aws

開發人員指南

AWS Key Management Service

AWS Key Management Service: 開發人員指南

Table of Contents

AWS Key Management Service

AWS Key Management Service (AWS KMS) 是一種受管服務，可讓您輕鬆地建立和控制用來保護資料的密碼編譯金鑰。AWS KMS 使用硬體安全模組 (HSM) 根據 FIPS 140-2 密碼模組驗證計畫來保護和驗證您的 AWS KMS keys。中國 (北京) 與中國 (寧夏) 區域不支援 FIPS 140-2 密碼編譯模組驗證計畫。AWS KMS 使用 OSCCA 認證之 HSM 來保護中國地區的 KMS 金鑰。

AWS KMS 已與大部分加密資料的其他 AWS 服務整合。AWS KMS 也與 AWS CloudTrail 整合，以針對稽核、法務和合規需求使用您的 KMS 金鑰。

您可以使用 AWS KMS API 來建立和管理 KMS 金鑰和特殊功能，例如自訂金鑰存放區，以及在密碼編譯操作中使用 KMS 金鑰。如需詳細資訊，請參閱 AWS Key Management Service API 參考。

您可以建立並管理您的 AWS KMS keys：

- 建立、編輯和檢視對稱和非對稱 KMS 金鑰 (包括 HMAC 金鑰)。
- 使用金鑰政策、IAM 政策和授予來控制對 KMS 金鑰的存取。AWS KMS 支援屬性型存取控制 (ABAC)。您也可以使用條件索引鍵調整政策。
- 建立、刪除、列出和更新別名，這是您 KMS 金鑰的易記名稱 您也可以使用別名控制對 KMS 金鑰的存取。
- 為您的 KMS 金鑰加上標籤以進行識別、自動化和成本追蹤。您也可以使用標籤控制對 KMS 金鑰的存取。
- 啟用和停用 KMS 金鑰。
- 啟用和停用 KMS 金鑰中密碼編譯資料的自動輪換。
- 刪除 KMS 金鑰以完成金鑰生命週期。

您可以在密碼編譯操作中使用 KMS 金鑰。如需範例，請參閱 對 AWS KMS API 進行程式設計。

- 使用對稱或非對稱 KMS 金鑰加密、解密和重新加密資料
- 使用非對稱 KMS 金鑰簽署和驗證訊息。
- 產生可匯出的對稱資料金鑰和非對稱資料金鑰對。
- 產生和驗證 HMAC 代碼。
- 產生適合加密應用程式的隨機數字。

您也可以使用 AWS KMS 的進階功能。

- 建立多區域金鑰，其相當於相同 KMS 金鑰在不同 AWS 區域 中的副本。
- 將密碼編譯材料匯入至 KMS 金鑰。
- 在 AWS CloudHSM 叢集提供技術的 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰。
- 在 AWS 之外的密碼編譯金鑰所支援的外部金鑰存放區中建立 KMS 金鑰。
- 透過 VPC 的私有端點直接連線到 AWS KMS。
- 使用混合式後量子 TLS 為您傳送至 AWS KMS 的資料提供前瞻性傳輸中加密。

透過 AWS KMS，您可對加密資料的存取擁有更多的掌控權。您可以直接在應用程式中使用金鑰管理和加密功能，或者透過與 AWS KMS 整合的 AWS 服務。不論您是撰寫 AWS 適用的應用程式或者使用 AWS 服務，AWS KMS 都可讓您維持控制誰能夠使用您的 AWS KMS keys，以及取得加密資料的存取權。

AWS KMS 與 AWS CloudTrail 整合，這項服務會將日誌檔案傳送到您指定的 Amazon S3 儲存貯體。透過使用， CloudTrail 您可以監控和調查 KMS 金鑰的使用方式和時間，以及使用者。

AWS 區域 中的 AWS KMS

支援 AWS KMS 的 AWS 區域 會列在 AWS Key Management Service 端點和配額中。如果 AWS KMS 支援的 AWS 區域 中不支援某個 AWS KMS 功能，該功能的相關主題中會描述區域性差異。

AWS KMS 定價

與其他 AWS 產品一樣，使用 AWS KMS 不需要合同或最低購買額。如需關於 AWS KMS 定價的詳細資訊，請參閱 AWS Key Management Service 定價。

服務水準協議

定義服務可用性政策的服務水準協議為 AWS Key Management Service 提供技術。

進一步了解

- 若要了解 AWS KMS 中的術語和概念，請參閱 AWS KMS 概念。
- 如需 AWS KMS API 的資訊，請參閱 AWS Key Management Service API 參考。如需不同程式設計語言的範例，請參閱 對 AWS KMS API 進行程式設計。
- 若要了解如何使用 AWS CloudFormation 範本建立和管理金鑰和別名，請參閱 透過 AWS CloudFormation 建立 AWS KMS 資源 和《AWS CloudFormation 使用者指南》中的AWS Key Management Service 資源類型參考。

- 如需 AWS KMS 如何使用密碼編譯和保護 KMS 金鑰的詳細技術資訊，請參閱 AWS Key Management Service 密碼編譯詳細資訊。密碼編譯詳細資訊文件並未描述 AWS KMS 在中國 (北京) 和中國 (寧夏) 區域的運作方式。
- 如需每個 AWS 區域 的 AWS KMS 端點清單 (包含 FIPS 端點)，請參閱《AWS 一般參考》AWS Key Management Service 主題的服務端點。
- 如需有關 AWS KMS 問題的說明，請參閱 AWS Key Management Service開發論壇。

AWS SDK 中的 AWS KMS

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP
- AWS SDK for Python (Boto3)
- AWS SDK for Ruby

AWS KMS 概念

了解 AWS Key Management Service (AWS KMS) 中使用的基本術語和概念，以及它們如何共同運作來保護您的資料。

主題

- AWS KMS keys
- 客戶金鑰和 AWS 金鑰
- 對稱加密 KMS 金鑰
- 非對稱 KMS 金鑰
- HMAC KMS 金鑰
- 資料金鑰
- 資料金鑰對
- Aliases

- 自訂金鑰存放區
- 密碼編譯操作
- 金鑰識別碼 (KeyId)
- 金鑰材料
- 金鑰資料來源
- 金鑰規格
- 金鑰用途
- 封套加密
- 加密內容
- 金鑰政策
- 授權
- 稽核 KMS 金鑰使用情形
- 金鑰管理基礎設施

AWS KMS keys

AWS KMS keys (KMS 金鑰) 是 AWS KMS 中的主要資源。您可以使用 KMS 金鑰來加密、解密和重新加密資料。它也可以產生在 AWS KMS 外部使用的資料金鑰。一般而言，您會使用對稱加密 KMS 金鑰，但您可以建立並使用非對稱 KMS 金鑰進行加密或簽署，以及建立並使用 HMAC KMS 金鑰來產生和驗證 HMAC 標籤。

> **Note**
>
> AWS KMS 正在將術語客戶主金鑰 (CMK) 取代為 AWS KMS key 和 KMS 金鑰。概念並沒有變更。為了防止重大變更，AWS KMS 會保留此術語的一些變化。

AWS KMS key 是密碼編譯金鑰的邏輯表示。KMS 金鑰包含中繼資料，如金鑰 ID、金鑰規格、金鑰使用方式、建立日期、描述和金鑰狀態。最重要的是，它包含對使用 KMS 金鑰執行密碼編譯操作時使用的金鑰材料的參考。

您可以使用在 AWS KMS FIPS 驗證的硬體安全模組中產生的密碼編譯金鑰材料來建立 KMS 金鑰。對稱 KMS 金鑰的金鑰材料和非對稱 KMS 金鑰的私有金鑰永遠不會讓 AWS KMS 處於未加密的狀態。若要使用或管理 KMS 金鑰，您必須使用 AWS KMS。如需建立和管理 KMS 金鑰的詳細資訊，請參閱 管

理金鑰。如需有關如何使用 KMS 金鑰的詳細資訊，請參閱《AWS Key Management Service API 參考》。

在預設情況下，AWS KMS 會為 KMS 金鑰建立金鑰材料。您無法擷取、匯出、檢視或管理此金鑰材料。唯一例外是非對稱金鑰對的公有金鑰，您可以將其匯出到 AWS 之外使用。而且，您也無法刪除金鑰材料，只能刪除 KMS 金鑰。不過，您可以將自己的金鑰材料匯入 KMS 金鑰，或使用自訂金鑰存放區來建立 KMS 金鑰，它們會使用 AWS CloudHSM 叢集中的金鑰材料，或者您在 AWS 之外擁有和管理的外部金鑰管理器中的金鑰材料。

AWS KMS 也支援多區域金鑰，它允許您在一個 AWS 區域 中加密資料，並在另一個 AWS 區域 中進行解密。

如需建立和管理 KMS 金鑰的詳細資訊，請參閱 管理金鑰。如需有關如何使用 KMS 金鑰的詳細資訊，請參閱《AWS Key Management Service API 參考》。

客戶金鑰和 AWS 金鑰

您建立的 KMS 金鑰是客戶受管金鑰。使用 KMS 金鑰來加密您的服務資源的 AWS 服務通常會為您建立金鑰。AWS 服務在您的 AWS 帳戶中為您建立的 KMS 金鑰是 AWS 受管金鑰。AWS 服務在服務帳戶中建立的 KMS 金鑰是 AWS 擁有的金鑰。

KMS 金鑰類型	可以檢視 KMS 金鑰中繼資料	可以管理 KMS 金鑰	僅適用於我的 AWS 帳戶	自動旋轉	定價
客戶受管金鑰	是	是	是	選用。每年 (大約 365 天)	月費 (按小時比例計算) 每次使用費
AWS 受管金鑰	是	否	是	必要。每年 (大約 365 天)	沒有月費 每次使用費 (有些 AWS 服務 為您支付此費用)
AWS 擁有的金鑰	否	否	否	各有不同	沒有費用

與 AWS KMS 整合的 AWS 服務所支援的 KMS 金鑰不盡相同。在預設情況下，有些 AWS 服務會使用 AWS 擁有的金鑰 或 AWS 受管金鑰 來加密資料。一些 AWS 服務支援客戶受管金鑰。另一些 AWS 服務則支援所有類型的 KMS 金鑰，讓您可輕鬆使用 AWS 擁有的金鑰、取得 AWS 受管金鑰 的可見度，或控制客戶受管金鑰。如需 AWS 服務提供之加密選項的詳細資訊，請參閱該服務之使用者指南或開發人員指南中的靜態加密主題。

客戶受管金鑰

您建立的 KMS 金鑰是客戶受管金鑰。客戶受管金鑰是您在 AWS 帳戶 中建立、擁有和管理的 KMS 金鑰。您可以完全控制這些 KMS 金鑰，包括建立和維護其金鑰政策、IAM 政策和授予、啟用和停用這些項目、輪換其密碼編譯材料、新增標籤、建立參考 KMS 金鑰的別名，以及排程 KMS 金鑰供刪除。

客戶受管金鑰會出現在 AWS KMS 的 AWS Management Console 客戶受管金鑰頁面。若要明確識別客戶管理的金鑰，請使用DescribeKey作業。對於客戶受管金鑰，`DescribeKey` 回應的 `KeyManager` 欄位值是 `CUSTOMER`。

您可以在密碼編譯操作中使用您的客戶受管金鑰，並在 AWS CloudTrail 日誌中稽核其使用情形。此外，許多與 AWS KMS 整合的 AWS 服務可讓您指定客戶受管金鑰，以保護為您存放和管理的資料。

客戶受管金鑰會衍生每月費用，以及超出免費方案部分的使用費用。它們都會計入您帳戶的AWS KMS配額中。如需詳細資訊，請參閱 AWS Key Management Service 定價和配額。

AWS 受管金鑰

AWS 受管金鑰 是您帳戶中的 KMS 金鑰，其由與 AWS KMS 整合的 AWS 服務代表您建立、管理和使用。

有些 AWS 服務讓您選擇 AWS 受管金鑰 或客戶受管金鑰來保護您在讓服務中的資源。一般而言，除非您需要控制保護資源的加密金鑰，否則 AWS 受管金鑰 是一個不錯的選擇。您不需要建立或維護金鑰或其金鑰政策，而且 AWS 受管金鑰 永遠沒有月費。

您具有許可，能在您的帳戶中檢視 AWS 受管金鑰，檢視其金鑰政策，以及在 AWS CloudTrail 日誌中稽核其使用方式。但是，您無法變更 AWS 受管金鑰 的任何屬性、進行輪換、變更其金鑰政策或進行排程以刪除。此外，您無法直接在密碼編譯操作中使用 AWS 受管金鑰；建立它們的服務會代表您加以使用。

AWS 受管金鑰 會出現在 AWS KMS 的 AWS Management Console AWS 受管金鑰 頁面。您也可以依別名識別 AWS 受管金鑰，其格式為 `aws/`*`service-name`*，例如 `aws/redshift`。要明確識別AWS受管金鑰，請使用DescribeKey操作。對於 AWS 受管金鑰，`DescribeKey` 回應的 `KeyManager` 欄位值是 `AWS`。

所有 AWS 受管金鑰 會每年自動輪換一次。您無法變更此輪換排程。

> **Note**
>
> 在 2022 年 5 月，AWS KMS 將 AWS 受管金鑰 的輪換頻率從每三年 (大約 1,095 天) 變更為每年 (大約 365 天)。
> 新的 AWS 受管金鑰 會在建立一年後自動輪換，此後大約每年輪換一次。
> 現有的 AWS 受管金鑰 會在最近一次輪換一年後自動輪換，此後每年自動輪換一次。

AWS 受管金鑰 沒有月費。必須對超過免費方案的使用支付費用，但有些 AWS 服務可為您涵蓋這些費用。如需詳細資訊，請參閱該服務使用者指南或開發人員指南中的靜態加密主題。如需詳細資訊，請參閱 AWS Key Management Service 定價。

AWS 受管金鑰不會計入您的帳戶在每個區域的 KMS 金鑰數目資源配額。但是，如果代表您帳戶中的委託人使用，則會將 KMS 金鑰計入請求配額中。如需詳細資訊，請參閱 配額。

AWS 擁有的金鑰

AWS 擁有的金鑰 是一組 KMS 金鑰，AWS 服務會擁有並管理這組 KMS 金鑰，以在多個 AWS 帳戶 中使用。雖然 AWS 擁有的金鑰 不在 AWS 帳戶 中，但 AWS 服務可以使用 AWS 擁有的金鑰，來保護您帳戶中的資源。

有些 AWS 服務會讓您選擇 AWS 擁有的金鑰 或客戶受管金鑰。一般而言，除非您需要稽核或控制保護您資源的加密金鑰，否則 AWS 擁有的金鑰 是一個不錯的選擇。AWS 擁有的金鑰 完全免費（沒有月費或使用費），不會計入您帳戶的 AWS KMS 配額，而且容易使用。您不需要建立或維護金鑰或其金鑰政策。

AWS 擁有的金鑰的輪換因服務而異。如需特定 AWS 擁有的金鑰 的輪換的詳細資訊，請參閱該服務之使用者指南或開發人員指南中的靜態加密主題。

對稱加密 KMS 金鑰

建立 AWS KMS key 時，根據預設，您會取得對稱加密的 KMS 金鑰。這是最基本和最常用的 KMS 金鑰類型。

在 AWS KMS 中，一個對稱加密 KMS 金鑰代表 256 位元 AES-GCM 加密金鑰，但中國區域除外，在該處代表 128 位元 SM4 加密金鑰。對稱金鑰資料絕不會讓 AWS KMS 出現未加密的情況。若要使用對稱加密 KMS 金鑰，您必須呼叫 AWS KMS。對稱加密會使用對稱加密金鑰，也會以相同的金鑰加

密和解密。除非您的任務明確要求非對稱加密，否則絕不會讓 AWS KMS 出現未加密狀況的對稱加密 KMS 金鑰是很好的選擇。

與 AWS KMS 整合的 AWS 服務僅會使用對稱加密 KMS 金鑰來加密您的資料。這些服務不支援使用非對稱 KMS 金鑰進行加密。如需有關如何判斷 KMS 金鑰是對稱還是不對稱的說明，請參閱識別非對稱 KMS 金鑰。

技術上來說，對稱金鑰的金鑰規格是 SYMMETRIC_DEFAULT，金鑰使用方式是 ENCRYPT_DECRYPT，加密演算法是 SYMMETRIC_DEFAULT。如需詳細資訊，請參閱 SYMMETRIC_DEFAULT 金鑰規格。

您可以在 AWS KMS 中使用對稱加密 KMS 金鑰來加密、解密和重新加密資料、產生資料金鑰和資料金鑰對。您可以建立多區域對稱加密 KMS 金鑰、將自己的金鑰資料匯入對稱加密 KMS 金鑰，並在自訂金鑰存放區中建立對稱加密 KMS 金鑰。如需不同類型的 KMS 金鑰執行之操作的比較表，請參閱金鑰類型參考。

非對稱 KMS 金鑰

您可以在 AWS KMS 中建立非對稱 KMS 金鑰。非對稱 KMS 金鑰表示以數學方式相關的公有金鑰和私有金鑰對。私有金鑰絕不會讓 AWS KMS 出現未加密的狀況。若要使用私有金鑰，您必須呼叫 AWS KMS。您可以呼叫 AWS KMS API 操作以使用 AWS KMS 內的公有金鑰，或者下載公有金鑰，並且在 AWS KMS 外面使用。您也可以建立多區域非對稱 KMS 金鑰。

您可以建立非對稱 KMS 金鑰，代表用於公有金鑰加密或簽署和驗證的 RSA 金鑰對或 SM2 金鑰對 (僅限中國區域)，或用於簽署和驗證的橢圓曲線金鑰對。

如需建立和使用非對稱 KMS 金鑰的詳細資訊，請參閱 AWS KMS 中的非對稱金鑰。

HMAC KMS 金鑰

HMAC KMS 金鑰是指不同長度的對稱金鑰，用於產生和驗證雜湊訊息驗證碼 (HMAC)。HMAC 金鑰中的金鑰資料絕不會讓 AWS KMS 出現未加密的狀況。若要使用 HMAC 金鑰，請呼叫 `GenerateMac` 或者 `VerifyMac` API 操作。

您也可以建立多區域 HMAC KMS 金鑰。

如需有關建立和使用 HMAC KMS 金鑰的詳細資訊，請參閱 AWS KMS 中的 HMAC 金鑰。

資料金鑰

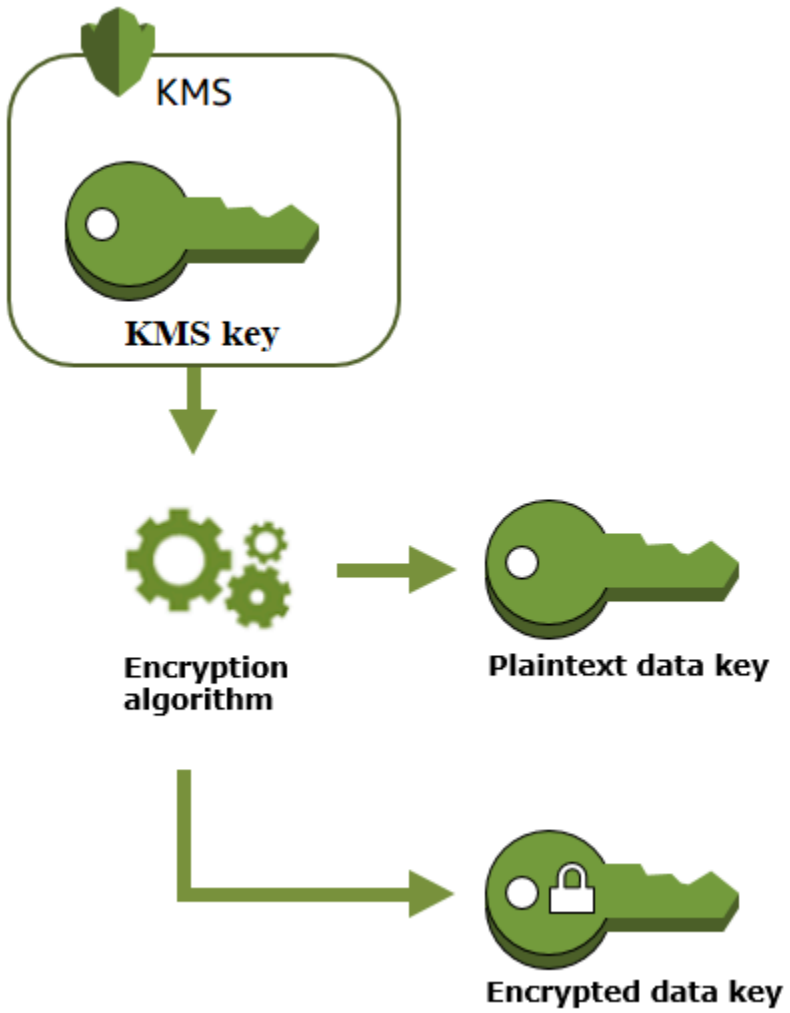
資料金鑰是對稱金鑰，可用於加密資料，包括大量資料和其他資料加密金鑰。與對稱 KMS 金鑰 (無法下載) 不同，資料金鑰可傳回給您，以便在 AWS KMS 外部使用。

AWS KMS 產生資料金鑰時，會傳回可立即使用 (選用) 的純文字資料金鑰，以及可與資料一起安全存放的資料金鑰加密複本。當您準備解密資料時，您必須先要求 AWS KMS 解密已加密的資料金鑰。

AWS KMS 會產生、加密及解密資料金鑰。不過，AWS KMS 不會存放、管理或追蹤您的資料金鑰，或使用資料金鑰來執行密碼編譯操作。您必須在 AWS KMS 外部使用和管理資料金鑰。如需安全使用資料金鑰的說明，請參閱 AWS Encryption SDK。

建立資料金鑰

若要建立資料索引鍵，請呼叫GenerateDataKey作業。 AWS KMS生成數據鍵。然後它會加密在您指定的對稱加密 KMS 金鑰下的資料金鑰複本。操作會傳回資料金鑰的純文字複本和以 KMS 金鑰加密的資料金鑰複本。下圖顯示此操作。

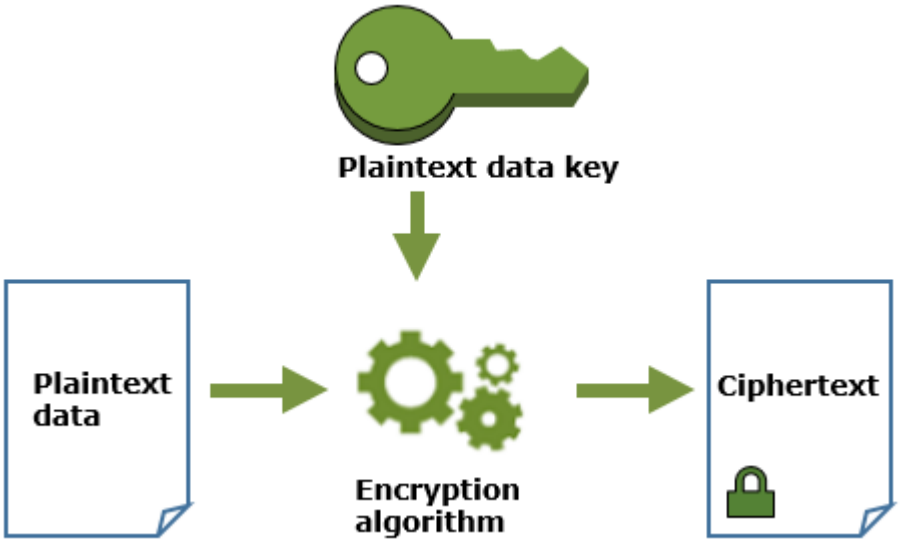


AWS KMS還支持GenerateDataKeyWithoutPlaintext操作，該操作僅返回加密的數據密鑰。當您需要使用資料金鑰，請要求 AWS KMS 解密它。

使用資料金鑰來加密資料

AWS KMS 無法使用資料金鑰來加密資料。但是，您可以在 AWS KMS 外部使用資料金鑰，例如使用 OpenSSL 或 AWS Encryption SDK 之類的密碼編譯程式庫。

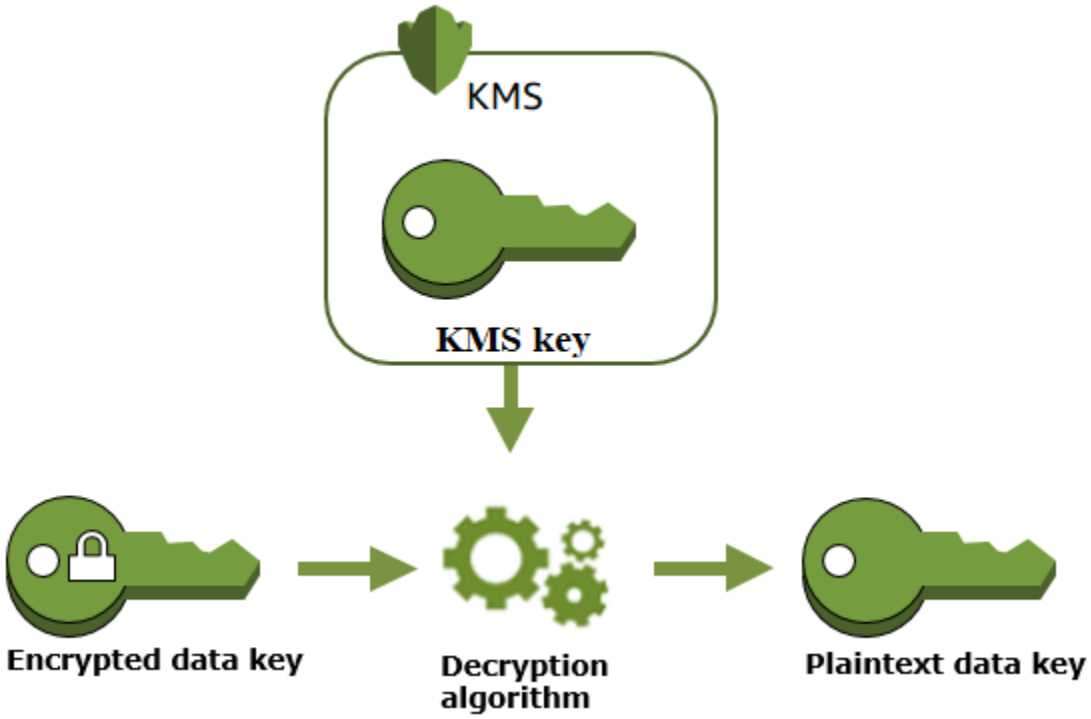
使用純文字資料金鑰加密資料後，請盡快從記憶體中移除該金鑰。您可將加密的資料金鑰與加密的資料安全地存放在一起，以便用來隨需解密資料。



使用資料金鑰來解密資料

若要解密資料，請傳遞加密的資料金鑰給 Decrypt 操作。AWS KMS 會使用您的 KMS 金鑰來解密資料金鑰，然後傳回純文字資料金鑰。使用純文字資料金鑰來解密您的資料，然後盡快從記憶體中移除純文字資料金鑰。

下圖說明如何使用 `Decrypt` 操作來解密加密的資料金鑰。



無法使用的 KMS 金鑰如何影響資料金鑰

當 KMS 金鑰變得無法使用時，效果幾乎是即時的 (視最終一致性而定)。KMS 金鑰的金鑰狀態變更反映了其最新狀況，所有在密碼編譯操作中使用 KMS 金鑰的請求都將失敗。

但是，對由 KMS 金鑰加密的資料金鑰的影響以及對由資料金鑰加密的資料的影響會延遲，除非再次使用 KMS 金鑰，例如解密資料金鑰。

KMS 金鑰變得無法使用的原因有很多，包括您可能執行的下列動作。

- 停用 KMS 金鑰
- 排程要刪除的 KMS 金鑰
- 從具有匯入金鑰材料的 KMS 金鑰中刪除金鑰材料，或允許匯入的金鑰材料過期。
- 中斷連接託管 KMS 金鑰的 AWS CloudHSM 金鑰存放區，或從用作 KMS 金鑰之金鑰材料的 AWS CloudHSM 叢集中刪除金鑰。
- 中斷連接託管 KMS 金鑰的外部金鑰存放區，或任何其他干擾外部金鑰存放區代理的加密和解密請求的動作，包括從其外部金鑰管理器刪除外部金鑰。

對於使用資料金鑰來保護服務所管理之資源的許多 AWS 服務 而言，此效果特別重要。下列範例使用 Amazon Elastic Block Store (Amazon EBS) 和 Amazon Elastic Compute Cloud (Amazon EC2)。不同

的 AWS 服務 以不同方式使用資料金鑰。如需詳細資訊，請參閱 AWS 服務 的「安全性」一章的「資料保護」一節。

例如，考量以下情境：

1. 您可以建立已加密的 EBS 磁碟區，並指定 KMS 金鑰來進行保護。Amazon EBS 會要求 AWS KMS 使用您的 KMS 金鑰來為磁碟區產生加密資料金鑰。Amazon EBS 會隨著磁碟區的中繼資料存放加密資料金鑰。
2. 當您將 EBS 磁碟區連接到 EC2 執行個體時，Amazon EC2 會使用您的 KMS 金鑰來解密 EBS 磁碟區的加密資料金鑰。Amazon EC2 使用 Nitro 硬體中的資料金鑰，該硬體負責將所有磁碟 I/O 加密到 EBS 磁碟區。只要 EBS 磁碟區連接 EC2 執行個體，資料金鑰就會存在 Nitro 硬體。
3. 您執行的動作使 KMS 金鑰無法使用。這不會立即影響 EC2 執行個體或 EBS 磁碟區。當磁碟區連接執行個體時，Amazon EC2 會採用資料金鑰 (而非 KMS 金鑰) 來加密所有磁碟 I/O。
4. 然而，當加密的 EBS 磁碟區從 EC2 執行個體中斷連接時，Amazon EBS 就會從 Nitro 硬體移除資料金鑰。下次再將加密的 EBS 磁碟區連接到 EC2 執行個體，連接會失敗，因為 Amazon EBS 無法使用 KMS 金鑰來解密磁碟區的加密資料金鑰。若要再次使用 EBS 磁碟區，您必須讓 KMS 金鑰變得再次可用。

資料金鑰對

資料金鑰對是非對稱的資料金鑰，由數學上相關的公有金鑰和私有金鑰組成。其旨在用於用戶端加密和解密，或是 AWS KMS 之外的簽署和驗證。

與 OpenSSL 等工具產生的資料金鑰對不同，AWS KMS 會保護您指定 AWS KMS 中對稱加密 KMS 金鑰下每個資料金鑰對中的私有金鑰。但是，AWS KMS 不會存放、管理或追蹤您的資料金鑰對，或使用資料金鑰對來執行密碼編譯操作。您必須在 AWS KMS 外部使用和管理資料金鑰對。

AWS KMS 支援下列類型的資料金鑰對：

- RSA 金鑰對：RSA_2048、RSA_3072 和 RSA_4096
- 橢圓曲線金鑰對：ECC_NIST_P256、ECC_NIST_P384、ECC_NIST_P521 及 ECC_SECG_P256K1
- SM 金鑰對 (僅限中國區域)：SM2

您選取的資料金鑰對類型通常取決於您的使用案例或法規需求。大多數的憑證都需要 RSA 金鑰。橢圓曲線金鑰通常用於數位簽章。ECC_SECG_P256K1 金鑰通常用於加密貨幣。AWS KMS 建議您使用

ECC 金鑰對進行簽署，並使用 RSA 金鑰對進行加密或簽署，但不能同時使用兩者。然而，AWS KMS 無法對 AWS KMS 外部資料金鑰對的使用強制執行任何限制。

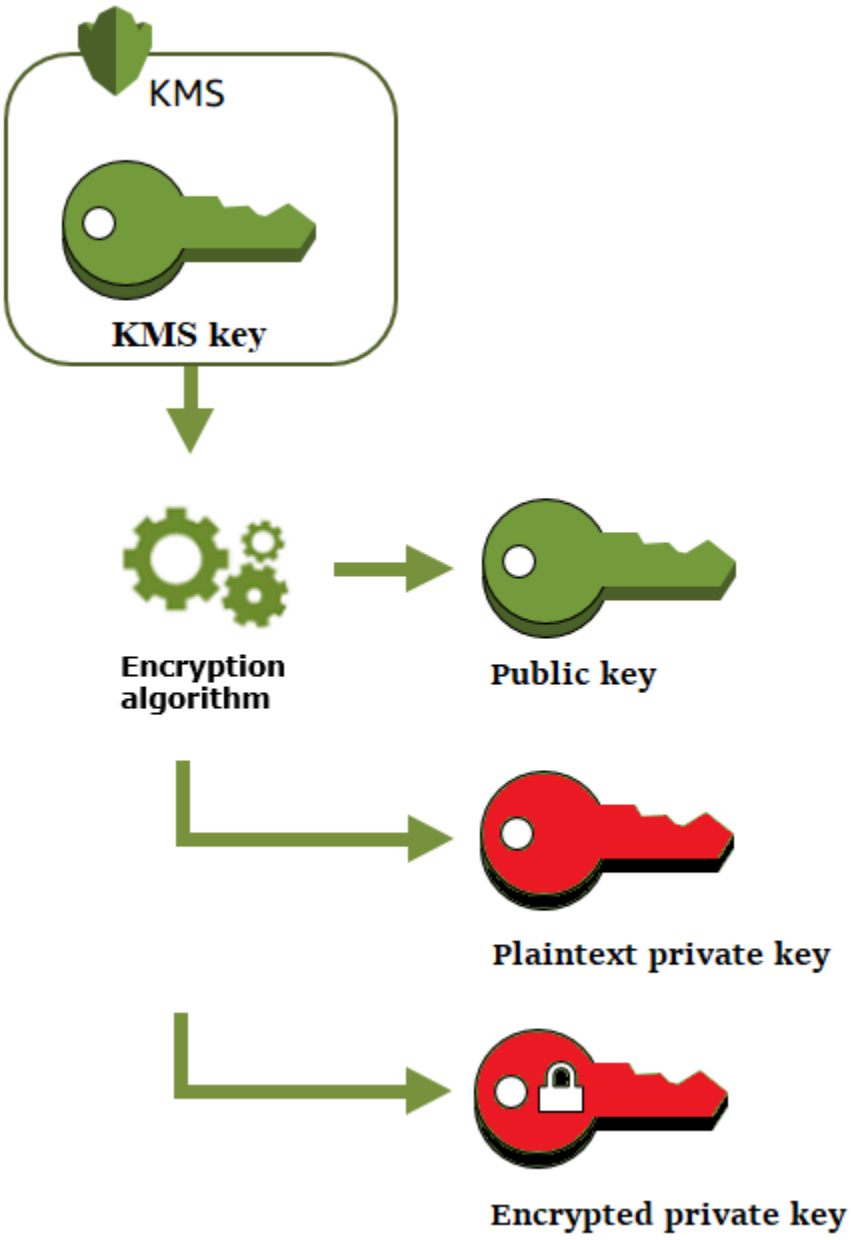
建立資料金鑰對

若要建立資料 key pair，請呼叫GenerateDataKeyPair或GenerateDataKeyPairWithoutPlaintext作業。指定您希望用來加密私有金鑰的對稱加密 KMS 金鑰。

`GenerateDataKeyPair` 會傳回純文字的公有金鑰、純文字的私有金鑰和加密後的私有金鑰。請在立即需要純文字私有金鑰的時候使用此操作 (例如產生數位簽章)。

`GenerateDataKeyPairWithoutPlaintext` 會傳回純文字的公有金鑰和加密後的私有金鑰，但其中不包括純文字的私有金鑰。請在您不立即需要純文字私有金鑰的時候使用此操作 (例如使用公有金鑰進行加密時)。稍後，當您需要純文字的私有金鑰來解密資料時，您可以呼叫 Decrypt 操作。

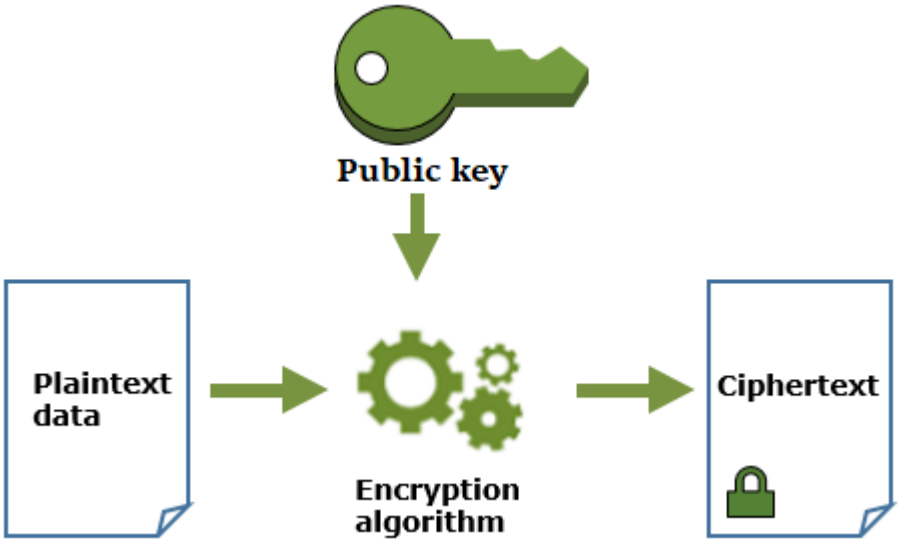
下圖顯示 `GenerateDataKeyPair` 操作。`GenerateDataKeyPairWithoutPlaintext` 操作會省略純文字私有金鑰。



使用資料金鑰對加密資料

當您使用資料金鑰對進行加密時，您可以使用金鑰隊的公有金鑰來加密資料，並使用相同金鑰對的私有金鑰來解密資料。通常，在多方需要加密僅持有私有金鑰的一方能夠解密的資料時，您會使用資料金鑰對。

持有公有金鑰的一方會使用該金鑰來加密資料，如下圖所示。

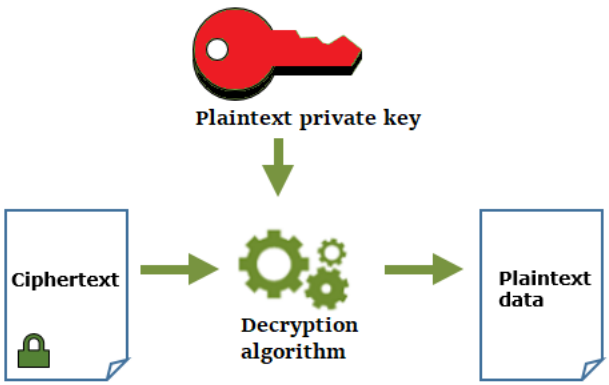


使用資料金鑰對解密資料

如要解密您的資料，請使用資料金鑰對中的私有金鑰。如要讓操作成功，公有和私有金鑰必須來自相同的資料金鑰對，且您必須使用相同的加密演算法。

如要解密經過加密的私有金鑰，請將該金鑰傳遞至 Decrypt 操作。使用純文字私有金鑰來解密資料。然後盡快從記憶體中移除純文字的私有金鑰。

下圖顯示如何使用資料金鑰對中的私有金鑰來解密加密文字。



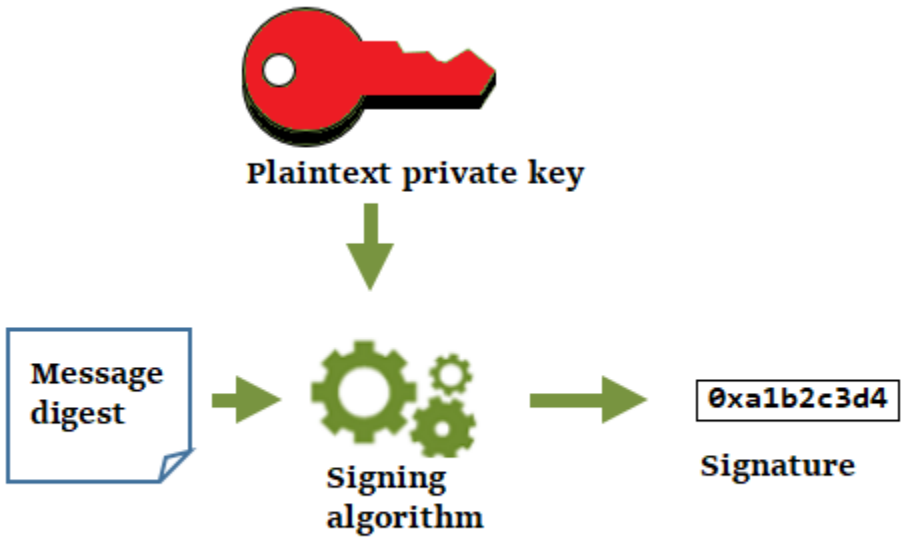
使用資料金鑰對簽署訊息

如要產生訊息的密碼編譯簽章，請使用資料金鑰對中的私有金鑰。任何具備公有金鑰的人員都能夠使用該金鑰來驗證訊息是使用您的私有金鑰進行簽署的，且在簽署之後並未發生任何變更。

如果加密您的私有金鑰，請將已加密的私有金鑰傳遞至 Decrypt 操作。AWS KMS 會使用您的 KMS 金鑰來解密資料金鑰，並傳回純文字的私有金鑰。使用純文字私有金鑰來產生簽章。然後盡快從記憶體中移除純文字的私有金鑰。

如要簽署訊息，請使用密碼編譯雜湊函數來建立訊息摘要，例如 OpenSSL 中的 `dgst` 命令。然後，將您的純文字私有金鑰傳遞至簽署演算法。結果便是代表訊息內容的簽章。(您可以在不先建立摘要的情況下簽署較短的訊息。訊息大小上限會因您使用的簽署工具而有所不同。)

下圖顯示如何使用資料金鑰對中的私有金鑰來簽署訊息。

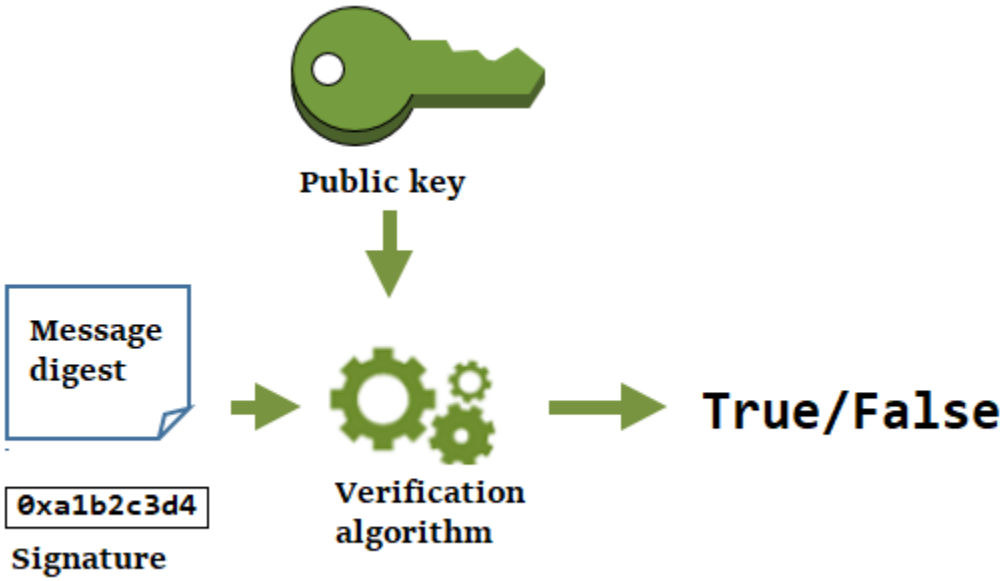


使用資料金鑰對驗證簽章

任何持有您資料金鑰對中公有金鑰的人員，都能使用該金鑰來驗證您使用私有金鑰產生的簽章。驗證會確認經過授權的使用者使用了指定的私有金鑰和簽署演算法簽署了訊息，且訊息自簽署以來並未產生變更。

如要成功，驗證簽章的一方必須產生相同的摘要類型、使用相同演算法，然後使用對應至用來簽署訊息私有金鑰的公有金鑰。

下圖顯示如何使用資料金鑰對中的公有金鑰來驗證訊息簽章。



Aliases

使用別名作為 KMS 金鑰的易記名稱。例如，您可以將 KMS 金鑰稱為 test-key，而不是 1234abcd-12ab-34cd-56ef-1234567890ab。

別名可讓您更輕鬆地識別 AWS Management Console 中的 KMS 金鑰。您可以使用別名來識別某些 AWS KMS 操作中的 KMS 金鑰，包括密碼編譯操作。在應用程式中，您可以使用單一別名來引用每個 AWS 區域 中的不同 KMS 金鑰。

您也可以根據其別名允許和拒絕存取 KMS 金鑰，而不需編輯政策或管理授權。這項功能是對屬性型存取控制 (ABAC) 的 AWS KMS 支援。如需詳細資訊，請參閱 AWS KMS 的 ABAC。

在 AWS KMS 中，別名是獨立的資源，而不是 KMS 金鑰的屬性。因此，您可以新增、變更和刪除別名，而不會影響相關聯的 KMS 金鑰。

> **Important**
>
> 請勿在別名名稱包含機密或敏感資訊。別名可能會以純文字顯示在 CloudTrail 記錄檔和其他輸出中。

進一步了解：

- 如需別名的詳細資訊，請參閱 使用別名。

- 如需金鑰識別符格式 (包括別名) 的相關資訊，請參閱 金鑰識別碼 (KeyId)。
- 如需尋找與 KMS 金鑰相關聯的別名，請參閱 尋找別名和別名 ARN
- 如需建立和管理別名之多種程式設計語言的範例，請參閱 處理別名。

自訂金鑰存放區

自訂金鑰存放區是您擁有和管理的 AWS KMS 之外的金鑰管理器所支援的 AWS KMS 資源。當您使用自訂金鑰存放區中的 KMS 金鑰進行密碼編譯操作時，密碼編譯操作實際上是在金鑰管理器中使用其密碼編譯金鑰執行的。

AWS KMS 支援由 AWS CloudHSM 叢集支援的 AWS CloudHSM 金鑰存放區，以及由 AWS 之外的外部金鑰管理器所支援的外部金鑰存放區。

如需詳細資訊，請參閱 自訂金鑰存放區。

密碼編譯操作

在 AWS KMS 中，密碼編譯操作是 API 操作，它會使用 KMS 金鑰來保護資料。因為 KMS 金鑰保留在 AWS KMS 內，您必須呼叫 AWS KMS 以在密碼編譯操作中使用 KMS 金鑰。

若要使用 KMS 金鑰執行密碼編譯操作，請使用 AWS 開發套件、AWS Command Line Interface (AWS CLI) 或 AWS Tools for PowerShell。您無法在 AWS KMS 主控台中執行密碼編譯操作。如需以數種程式設計語言呼叫密碼編譯操作的範例，請參閱對 AWS KMS API 進行程式設計。

下表列出 AWS KMS 密碼編譯操作。它也會顯示操作中使用之 KMS 金鑰的金鑰類型和金鑰使用情形需求。

操作	Key type	金鑰用途
解密	對稱或不對稱	ENCRYPT_DECRYPT
加密	對稱或不對稱	ENCRYPT_DECRYPT
GenerateDataKey	對稱	ENCRYPT_DECRYPT
GenerateDataKeyPair	對稱 [1] 不支援自訂金鑰存放區中的 KMS 金鑰。	ENCRYPT_DECRYPT

操作	Key type	金鑰用途
GenerateDataKeyPairWithoutPlaintext	對稱 [1] 不支援自訂金鑰存放區中的 KMS 金鑰。	ENCRYPT_DECRYPT
GenerateDataKeyWithoutPlaintext	對稱	ENCRYPT_DECRYPT
GenerateMac	HMAC	GENERATE_VERIFY_MAC
GenerateRandom	N/A。此操作不會使用 KMS 金鑰。	N/A
ReEncrypt	對稱或不對稱	ENCRYPT_DECRYPT
符號	非對稱	SIGN_VERIFY
確認	非對稱	SIGN_VERIFY
VerifyMac	HMAC	GENERATE_VERIFY_MAC

[1] 產生受對稱加密 KMS 金鑰保護的非對稱資料金鑰對。

如需密碼編譯操作許可的詳細資訊，請參閱the section called “許可參考”。

為了讓 AWS KMS 對所有使用者都能迅速回應以及有高效能，AWS KMS 針對可在每秒呼叫的密碼編譯操作數量建立配額。如需詳細資訊，請參閱 the section called “密碼編譯操作的共用配額”。

金鑰識別碼 (KeyId)

金鑰識別符形同 KMS 金鑰的名稱，可幫助您在主控台中識別您的 KMS 金鑰。您可以使用它們指出要在 AWS KMS API 操作、金鑰政策、IAM 政策和授予中使用的 KMS 金鑰。金鑰識別符的值與 KMS 金鑰相關聯的金鑰資料完全無關。

AWS KMS 定義了數個金鑰識別符。當您建立 KMS 金鑰時，AWS KMS 會產生金鑰 ARN 和金鑰 ID，它們是 KMS 金鑰的屬性。建立別名時，AWS KMS 會根據您定義的別名產生別名 ARN。您可以檢視 AWS Management Console 和 AWS KMS API 中的金鑰和別名識別符。

在 AWS KMS 主控台中，您可以依金鑰 ARN、金鑰 ID 或別名來檢視和篩選 KMS 金鑰，並依金鑰 ID 和別名進行排序。如需在主控台中尋找金鑰識別符的說明，請參閱the section called “尋找金鑰 ID 和金鑰 ARN”。

在 AWS KMS API 中，您用來識別 KMS 金鑰的參數命名為 `KeyId` 或變體，例如 `TargetKeyId` 或 `DestinationKeyId`。不過，這些參數的數值並不限於金鑰 ID。有些參數可以接受任何有效的金鑰識別符。如需每個參數值的相關資訊，請參閱《AWS Key Management Service API 參考》中的參數描述。

> **Note**
>
> 使用 AWS KMS API 時，請審慎使用金鑰識別符。不同的 API 需要不同的金鑰識別符。一般而言，請使用您任務適用之最完整且最實用的金鑰識別符。

AWS KMS 支援下列金鑰識別符。

金鑰 ARN

金鑰 ARN 是 KMS 金鑰的 Amazon Resource Name (ARN)。它是 KMS 金鑰唯一、完全合格的識別符。金鑰 ARN 包括 AWS 帳戶、區域和金鑰 ID。如需尋找 KMS 金鑰的金鑰 ARN 說明，請參閱 the section called “尋找金鑰 ID 和金鑰 ARN”。

金鑰 ARN 的格式如下：

```
arn:<partition>:kms:<region>:<account-id>:key/<key-id>
```

以下是單一區域 KMS 金鑰的範例金鑰 ARN。

```
arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
```

多區域金鑰之金鑰 ARN 的 *`key-id`* 元素以 `mrk-` 字首開頭。以下是多區域金鑰的範例金鑰 ARN。

```
arn:aws:kms:us-west-2:111122223333:key/mrk-1234abcd12ab34cd56ef1234567890ab
```

金鑰 ID

金鑰 ID 可唯一地識別帳戶和區域內的 KMS 金鑰。如需尋找 KMS 金鑰的金鑰 ID 說明，請參閱 the section called “尋找金鑰 ID 和金鑰 ARN”。

以下是單一區域 KMS 金鑰的範例金鑰 ID。

```
1234abcd-12ab-34cd-56ef-1234567890ab
```

多區域金鑰的金鑰 ID 以 `mrk-` 字首開頭。以下是多區域金鑰的範例金鑰 ID。

```
mrk-1234abcd12ab34cd56ef1234567890ab
```

別名 ARN

別名 ARN 是 AWS KMS 別名的 Amazon Resource Name (ARN)。它是別名及其代表的 KMS 金鑰唯一、完全合格的識別符。別名 ARN 包括 AWS 帳戶、區域和別名名稱。

別名 ARN 在任何時候可識別一個特定的 KMS 金鑰。不過，因為您可以變更與別名相關聯的 KMS 金鑰，所以別名 ARN 可以在不同的時間識別不同的 KMS 金鑰。如需尋找 KMS 金鑰的別名 ARN 說明，請參閱 尋找別名和別名 ARN。

別名 ARN 的格式如下：

```
arn:<partition>:kms:<region>:<account-id>:alias/<alias-name>
```

以下是虛構 `ExampleAlias` 的 ARN 別名。

```
arn:aws:kms:us-west-2:111122223333:alias/ExampleAlias
```

別名名稱

別名名稱是最多 256 個字元的字串。其可唯一地識別帳戶和區域內關聯的 KMS 金鑰。在 AWS KMS API 中，別名一律以 `alias/` 開頭。如需尋找 KMS 金鑰之別名名稱的說明，請參閱 尋找別名和別名 ARN。

別名的格式如下：

```
alias/<alias-name>
```

例如：

```
alias/ExampleAlias
```

別名名稱的 `aws/` 字首預留給 AWS 受管金鑰。您無法使用此字首建立別名。例如，Amazon Simple Storage Service (Amazon S3) 的 AWS 受管金鑰 別名名稱如下所示。

```
alias/aws/s3
```

金鑰材料

金鑰資料是密碼編譯演算法中使用的位元字串。私密金鑰資料必須保密，以保護使用它的密碼編譯操作。公有金鑰資料旨在共用。

每個 KMS 金鑰在中繼資料中都包含對其金鑰資料的引用。對稱加密 KMS 金鑰的金鑰資料來源可能會有所不同。您可以使用 AWS KMS 產生的金鑰材料，在自訂金鑰存放區的 AWS CloudHSM 叢集中產生的金鑰材料，或匯入您自己的金鑰材料。如果將 AWS KMS 金鑰資料用於對稱加密 KMS 金鑰，您可以啟用金鑰資料的自動輪換。

根據預設，每個 KMS 金鑰都有唯一的金鑰材料。不過，您可以建立一組具有相同金鑰材料的多區域金鑰。

金鑰資料來源

金鑰材料來源 是 KMS 金鑰屬性，用於識別 KMS 金鑰中金鑰材料的來源。您可以在建立 KMS 金鑰時選擇金鑰材料來源，之後便無法進行變更。金鑰材料的來源會影響 KMS 金鑰的安全性、耐久性、可用性、延遲和輸送量特性。

若要尋找 KMS 金鑰的金鑰材料來源，請使用DescribeKey作業，或在AWS KMS主控台中查看詳細資料頁面的 [加密組態] 索引標籤上的 [Origin] 值。如需相關說明，請參閱檢視金鑰。

KMS 金鑰可具有下列其中一個金鑰材料來源值。

`AWS_KMS`

AWS KMS 會在自己的金鑰存放區中建立和管理 KMS 金鑰的金鑰材料。這是大多數 KMS 金鑰的預設值，也是建議使用的數值。

如需使用來自 AWS KMS 的金鑰材料建立金鑰的說明，請參閱建立金鑰。

`EXTERNAL (Import key material)`

KMS 金鑰擁有匯入的金鑰材料。當您建立具有 `External` 金鑰材料來源的 KMS 金鑰時，KMS 金鑰不具有金鑰材料。您可以在稍後將金鑰材料匯入 KMS 金鑰。當使用匯入的金鑰材料時，您需要保護並管理 AWS KMS 以外的金鑰材料，包括在金鑰材料到期時更換金鑰材料。如需詳細資訊，請參閱 關於匯入的金鑰材料。

如需為匯入的金鑰材料建立 KMS 金鑰的說明，請參閱 步驟 1：建立不含金鑰材料的 KMS 金鑰。

AWS_CLOUDHSM

AWS KMS 在 AWS CloudHSM 叢集中為您的 AWS CloudHSM 金鑰存放區建立金鑰材料。

如需有關在 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰的說明，請參閱 在 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰。

EXTERNAL_KEY_STORE

金鑰材料是 AWS 之外的外部金鑰管理器中的密碼編譯金鑰。只有外部金鑰存放區中的 KMS 金鑰才支援此來源。

如需有關在外部金鑰存放區中建立 KMS 金鑰的說明，請參閱 在外部金鑰存放區中建立 KMS 金鑰。

金鑰規格

金鑰規格是一種代表金鑰密碼編譯組態的屬性。金鑰規格的含義與金鑰類型不同。

- AWS KMS 金鑰 – 金鑰規格會決定 KMS 金鑰為對稱還是非對稱。其也會決定其金鑰材料的類型，及其支援的演算法。您可以在建立 KMS 金鑰時選擇金鑰規格，之後便無法進行變更。預設金鑰規格 SYMMETRIC_DEFAULT 表示 256 位元對稱加密金鑰。

  > **Note**
  >
  > KMS 金鑰的 `KeySpec` 被稱為 `CustomerMasterKeySpec`。該CreateKey操作的`CustomerMasterKeySpec`參數已被棄用。改為使用 `KeySpec` 參數，其工作方式相同。為了防止中斷變更，`CreateKey`和DescribeKey作業的回應現在會包含具有相同值的`KeySpec`和`CustomerMasterKeySpec`成員。

  如需金鑰規格的清單和選擇金鑰規格的協助，請參閱 選取金鑰規格。若要尋找 KMS 金鑰的金鑰規格，請使用DescribeKey作業，或參閱AWS KMS主控台中 KMS 金鑰詳細資料頁面上的 [加密組態] 索引標籤。如需相關說明，請參閱檢視金鑰。

  若要限制主體在建立 KMS 金鑰時可使用的金鑰規格，請使用 kms: KeySpec 條件金鑰。您也可以使用 `kms:KeySpec` 條件金輪，允許主體僅針對具有特定金鑰規格的 KMS 金鑰呼叫 AWS KMS 操作。例如，您可以拒絕排程刪除任何具有 `RSA_4096` 金鑰規格之 KMS 金鑰的許可。
- 資料鍵 (GenerateDataKey) — 關鍵規格決定 AES 資料金鑰的長度。

- 資料鍵配對 (GenerateDataKeyPair) — key pair 規格決定資料 key pair 中金鑰材料的類型。

金鑰用途

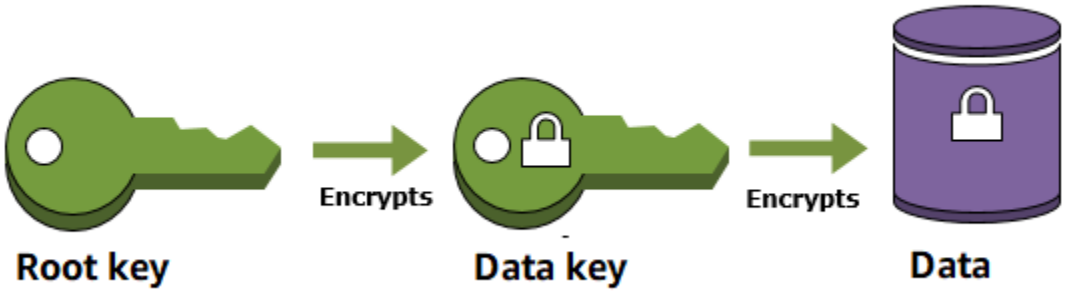
金鑰用途是決定該金鑰支援之密碼編譯操作的屬性。KMS 金鑰的金鑰用途可以是 ENCRYPT_DECRYPT、SIGN_VERIFY 或 GENERATE_VERIFY_MAC。每個 KMS 金鑰都僅有一種金鑰用途。使用 KMS 金鑰執行多種類型的操作，會使得兩種操作的產品更容易受到攻擊。

如需有關為 KMS 金鑰選擇金鑰用途的說明，請參閱選取金鑰用途。若要尋找 KMS 金鑰的金鑰用途，請使用DescribeKey作業，或在AWS KMS主控台中針對 KMS 金鑰選擇詳細資料頁面上的 [加密組態] 索引標籤。如需相關說明，請參閱檢視金鑰。

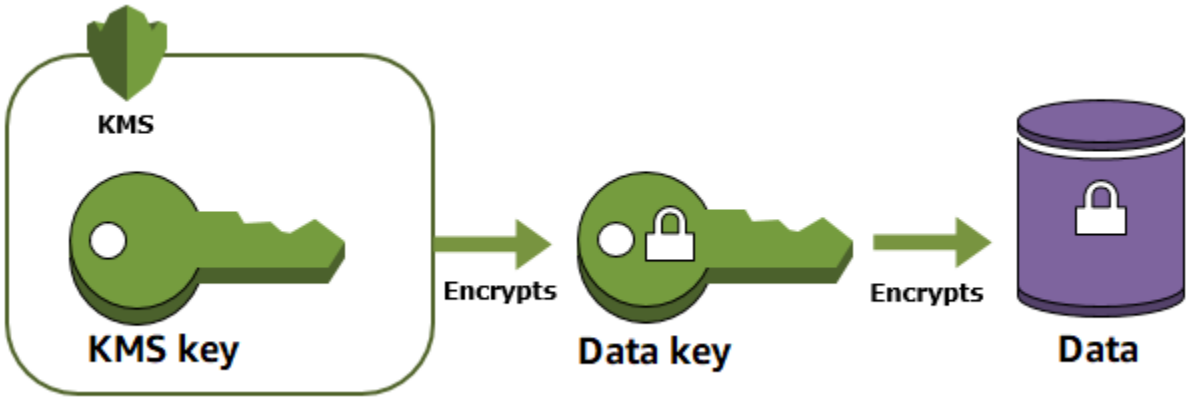
封套加密

加密資料後，您的資料會受到保護，但您需要保護您的加密金鑰。一個策略是把它加密。封套加密是使用資料金鑰來加密純文字資料，然後再透過另一個金鑰來加密資料金鑰的實務做法。

您甚至可以透過另一個加密金鑰來加密資料加密金鑰，並透過另一個加密金鑰來加密該加密金鑰。但是，最終必須有一個金鑰保留在純文字中，以便您可以解密金鑰和您的資料。這個最上層的純文字金鑰加密金鑰稱為根金鑰。



AWS KMS 透過安全地存放和管理加密金鑰，來協助您保護您的主金鑰。存放在 AWS KMS 中的根金鑰稱為 AWS KMS keys，永遠會對 AWS KMS FIPS 驗證的硬體安全模組進行加密。若要使用 KMS 金鑰，您必須呼叫 AWS KMS。



封套加密提供多種優勢：

- 保護資料金鑰

  加密資料金鑰時，您不需要擔心如何存放加密的資料金鑰，因為這份資料金鑰本質上已獲得加密保護。加密的資料金鑰可以安全地跟加密資料一起存放。

- 在多個金鑰下加密相同的資料

  加密操作可能極為耗時，特別是要加密的資料屬於大型物件時，更為明顯。這時您可以捨棄使用不同金鑰來多次重新加密原始資料的做法，改成只重新加密負責保護原始資料的資料金鑰。

- 結合多種演算法的優勢

  一般而言，比起公開金鑰演算法，對稱金鑰演算法速度較快，產生的加密文字較小。但是，公開金鑰演算法本質上就會區隔角色，因此金鑰管理較為方便。封套加密可讓您結合每個策略的優點。

加密內容

所有使用對稱加密 KMS 金鑰的 AWS KMS 密碼編譯操作都接受加密內容，這是一組選用的非私密金鑰值對，可包含資料的額外內容資訊。AWS KMS 會使用加密內容作為額外驗證資料 (AAD)，以便支援經過驗證的加密。

當您在加密請求中包含加密內容時，它會以密碼編譯方式繫結至加密文字，因此解密 (或解密並重新加密) 資料需要相同的加密內容。如果解密請求中提供的加密內容並非精確的區分大小寫相符，則解密請求將會失敗。加密內容中只有金鑰值對的順序可以不同。

> **Note**
>
> 您不能使用非對稱 KMS 金鑰或 HMAC KMS 金鑰在密碼編譯操作中指定加密內容。非對稱演算法和 MAC 演算法不支援加密內容。

加密內容不是秘密，也不被加密。它會以純文字出現在 AWS CloudTrail Logs 中，因此您可以使用它來識別和分類密碼編譯操作。您的加密內容不應包含敏感資訊。建議在您的加密內容中描述要加密或解密的資料。例如，當您加密檔案時，您可以使用檔案路徑的一部分作為加密內容。

```
"encryptionContext": {
    "department": "10103.0"
}
```

例如，在加密使用 Amazon 彈性區塊存放區 (Amazon EBS) CreateSnapshot操作建立的磁碟區和快照時，Amazon EBS 會使用該磁碟區 ID 做為加密內容值。

```
"encryptionContext": {
  "aws:ebs:id": "vol-abcde12345abc1234"
}
```

您也可以使用加密內容來進一步調整或限制對您帳戶中 AWS KMS keys 的存取。您可以使用加密內容做為授與中的限制，以及做為政策陳述式中的條件。

若要瞭解如何使用加密內容來保護加密資料的完整性，請參閱如何使用AWS Key Management Service和AWS安全部落格 EncryptionContext上的「如何保護加密資料的完整性」一文。

更多有關加密內容的資訊。

加密內容規則

AWS KMS 會對加密內容金鑰和值強制執行以下規則。

- 加密內容對中的金鑰和值必須是簡單的常值字串。如果您使用不同的類型，例如整數或浮點數，AWS KMS 會將它解譯為字串。
- 加密內容中的金鑰和值可以包含 Unicode 字元。如果加密內容包含金鑰政策或 IAM 政策中不允許的字元，則您將無法在政策條件金鑰中指定加密內容，例如 `kms:EncryptionContext:context-key` 和 `kms:EncryptionContextKeys`。如需有關金鑰政策文件規則的詳細資訊，請參閱金鑰政策格式。如需有關 IAM 政策文件規則的詳細資訊，請參閱《IAM 使用者指南》中的 IAM 名稱需求。

政策中的加密內容

加密內容主要用於驗證完整性和真確性。但是，您也可以在金鑰政策和 IAM 政策中，使用加密內容來控制對對稱加密 AWS KMS keys 的存取。

kms:: 和 kmsEncryptionContext: EncryptionContextKeys 條件金鑰只有在要求包含特定的加密內容金鑰或金鑰-值配對時，才允許 (或拒絕) 權限。

例如，下列金鑰政策陳述式允許 RoleForExampleApp 角色在 Decrypt 操作中使用 KMS 金鑰。它使用 kms:EncryptionContext:*context-key* 條件金鑰，只在請求中的加密內容包含 AppName:ExampleApp 加密內容對時，才允許此許可。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": "kms:Decrypt",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:EncryptionContext:AppName": "ExampleApp"
    }
  }
}
```

如需這些加密內容條件金鑰的詳細資訊，請參閱 條件鍵 AWS KMS。

授與中的加密內容

建立授權時，您可以包含為授予許可建立條件的授予限制條件。AWS KMS 支援兩個授予限制條件，EncryptionContextEquals 和 EncryptionContextSubset，它們都涉及密碼編譯操作請求中的加密內容。當您使用這些授予限制條件時，只有在密碼編譯操作請求中的加密內容滿足授予限制條件的需求時，授予中的許可才會生效。

例如，您可以將EncryptionContextEquals授與條件約束新增至允許GenerateDataKey作業的授與。藉由此限制條件，只有在請求中的加密內容與授予限制條件中的加密內容大小寫全部相符時，授予才會允許操作。

```
$ aws kms create-grant \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
```

```
    --grantee-principal arn:aws:iam::111122223333:user/exampleUser \
    --retiring-principal arn:aws:iam::111122223333:role/adminRole \
    --operations GenerateDataKey \
    --constraints EncryptionContextEquals={Purpose=Test}
```

來自承授者委託人的類似下列請求應當滿足 `EncryptionContextEquals` 限制條件。

```
$ aws kms generate-data-key \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --key-spec AES_256 \
    --encryption-context Purpose=Test
```

如需授予限制條件的詳細資訊，請參閱 使用授予限制條件。如需授予的詳細資訊，請參閱 the section called “授權”。

記錄加密內容

AWS KMS 使用 AWS CloudTrail 來記錄加密內容，讓您可以判斷哪些 KMS 金鑰和資料已被存取。日誌項目會準確顯示哪個 KMS 金鑰用來加密或解密由日誌項目中加密內容參考的特定資料。

> **Important**
>
> 由於加密內容會被記錄，因此不能包含敏感資訊。

儲存加密內容

如要簡化在您呼叫 `Decrypt` 或 `ReEncrypt` 操作時的任何加密內容使用，您可以將加密內容與加密的資料存放在一起。我們建議您只存放足夠的加密內容，以協助您在需要用它來加密或解密時建立完整的加密內容。

例如，如果加密內容是完整路徑檔案，只需將部分路徑與加密的檔案內容一起存放。接著，當您需要完整的加密內容，請從存放的片段中重新建構。如果有人擅自竄改檔案，例如重新命名或移到不同的位置，加密內容值會變更，解密請求會失敗。

金鑰政策

建立 KMS 金鑰時，您會決定誰能夠使用和管理該 KMS 金鑰。這些許可包含在稱為金鑰政策的文件中。您可以使用金鑰政策來隨時新增、移除或變更客戶受管金鑰的許可。但是，您不能編輯 AWS 受管金鑰 的金鑰政策。如需詳細資訊，請參閱 AWS KMS 中的金鑰政策。

授權

授予是一個政策工具，允許 AWS 委託人使用密碼編譯操作中的 AWS KMS keys。它也可以讓其檢視 KMS 金鑰 (DescribeKey)，並建立和管理授予。當授權存取 KMS 金鑰時，會考慮與金鑰政策和 IAM 政策一起授予。授予通常用於臨時許可，因為您可以建立授予、使用其許可並刪除授予，而無需變更金鑰政策或 IAM 政策。由於授與可能非常具體，而且易於建立和撤銷，因此通常用於提供臨時的許可或更精細的許可。

如需授予 (包括授予術語) 的詳細資訊，請參閱 AWS KMS 中的授權。

稽核 KMS 金鑰使用情形

您可以使用AWS CloudTrail稽核金鑰使用情況。 CloudTrail 會建立包含帳戶 AWS API 呼叫歷史記錄和相關事件的記錄檔。這些日誌檔包含使用 AWS 管理主控台、AWS 軟體開發套件和命令列工具提出的所有 AWS KMS API 請求。日誌檔也包含 AWS 服務代您向 AWS KMS 發出的請求。您可以使用這些日誌檔來尋找重要資訊，包括使用 KMS 金鑰的時間、所請求的操作、申請者的身分和來源 IP 地址。如需詳細資訊，請參閱 使用 AWS CloudTrail 進行記錄 及《AWS CloudTrail 使用者指南》。

金鑰管理基礎設施

密碼編譯的常見做法是使用公開且經同儕審查的演算法，例如 AES (進階加密標準) 和私密金鑰。密碼編譯的一個主要問題是金鑰很難保持私密。這通常是金鑰管理基礎設施 (KMI) 的任務。AWS KMS 會為您操作金鑰基礎設施。AWS KMS 會建立和安全地存放您的根金鑰，稱為 AWS KMS keys。如需 AWS KMS 如何操作的詳細資訊，請參閱 AWS Key Management Service 密碼編譯詳細資訊。

管理金鑰

若要開始使用 AWS KMS，請建立 AWS KMS key。

本節中的主題會說明如何管理基本 KMS 金鑰 (即對稱加密 KMS 金鑰)，包括從建立到刪除的完整過程。其中包括編輯和檢視金鑰、為金鑰加上標籤、啟用和停用金鑰、輪換金鑰材料，以及使用 AWS 工具和服務來監控 KMS 金鑰的使用情況等主題。它還包括有關使用 AWS CloudFormation 來建立和管理您的 KMS 金鑰的詳細資訊，以及顯示每個 AWS KMS 操作所需金鑰狀態的金鑰狀態參考。

如需有關建立、使用和管理其他類型 KMS 金鑰的詳細資訊，請參閱特殊用途金鑰。

主題

- 建立金鑰
- 使用別名
- 檢視金鑰
- 編輯金鑰
- 標記金鑰
- 啟用和停用金鑰
- 輪換 AWS KMS keys
- 監控 AWS KMS keys
- 透過 AWS CloudFormation 建立 AWS KMS 資源
- 刪除 AWS KMS keys
- AWS KMS 金鑰的金鑰狀態

建立金鑰

您可以AWS KMS keys在中建立AWS Management Console，也可以使用CreateKey作業或AWS CloudFormation範本來建立。在此過程中，您可選擇 KMS 金鑰的類型、其區域性 (單一區域或多區域)，以及金鑰資料的來源 (依預設，AWS KMS 會建立金鑰資料)。建立 KMS 金鑰之後即無法變更這些屬性。您也可以設定 KMS 金鑰的金鑰政策，您可以隨時變更這些政策。

本主題介紹如何使用 AWS KMS 的金鑰資料，為單一區域建立基本 KMS 金鑰 (即對稱加密 KMS 金鑰)。您可使用此 KMS 金鑰來保護 AWS 服務 中的資料。如需有關對稱加密 KMS 金鑰的詳細資訊，請

參閱 SYMMETRIC_DEFAULT 金鑰規格。如需有關如何建立其他類型金鑰的說明，請參閱特殊用途金鑰。

如果您要建立 KMS 金鑰來加密在 AWS 服務中存放或管理的資料，則請建立對稱加密 KMS 金鑰。與 AWS KMS 整合的 AWS 服務僅會使用對稱加密 KMS 金鑰來加密您的資料。這些服務不支援使用非對稱 KMS 金鑰進行加密。如需決定要建立的 KMS 金鑰類型說明，請參閱 選擇一個 KMS 金鑰類型。

> **Note**
>
> 對稱 KMS 金鑰現在稱為對稱加密 KMS 金鑰。AWS KMS 支援兩種對稱 KMS 金鑰，即對稱加密 KMS 金鑰 (預設類型) 和 HMAC KMS 金鑰 (也是對稱金鑰)。

在 AWS KMS 主控台建立 KMS 金鑰時，您需要為其提供別名 (易記名稱)。`CreateKey` 操作不會建立新 KMS 金鑰的別名。若要為新的或現有的 KMS 金鑰建立別名，請使用此CreateAlias作業。如需 AWS KMS 中別名的詳細資訊，請參閱 使用別名。

本主題介紹如何建立對稱加密 KMS 金鑰。請利用下表尋找指示，了解如何建立不同類型 KMS 金鑰。

建立 KMS 金鑰的指示

KMS 金鑰類型	指示
對稱加密金鑰 (SYMMETRIC_DEFAULT)	the section called “建立對稱加密 KMS 金鑰”
非對稱金鑰	the section called “建立非對稱 KMS 金鑰”
HMAC 金鑰	the section called “建立 HMAC 金鑰”
多區域金鑰 (任何類型)	the section called “建立具有匯入金鑰材料的主要金鑰” the section called “建立具有匯入金鑰材料的複本金鑰”
匯入的金鑰資料 (「使用自有金鑰 — BYOK」)	the section called “步驟 1：建立不含金鑰材料的 KMS 金鑰”
AWS CloudHSM 金鑰存放區	the section called “在 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰”

KMS 金鑰類型	指示
外部金鑰存放區 (「持有自有金鑰 — HYOK」)	the section called “在外部金鑰存放區中建立 KMS 金鑰”

進一步了解：

- 若要建立用戶端加密的資料金鑰，請使用此GenerateDataKey作業。
- 若要建立用於加密或簽署的非對稱 KMS 金鑰，請參閱 建立非對稱 KMS 金鑰。
- 若要建立 HMAC KMS 金鑰，請參閱建立 HMAC KMS 金鑰。
- 若要使用匯入的金鑰資料建立 KMS 金鑰 (「使用自有金鑰」)，請參閱匯入金鑰材料步驟 1：建立不含金鑰材料的 AWS KMS key。
- 若要建立多區域主要金鑰或複本金鑰，請參閱 建立多區域金鑰。
- 若要在自訂金鑰存放區 (金鑰資料來源為自訂金鑰存放區 (CloudHSM)) 中建立 KMS 金鑰，請參閱在 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰。
- 若要使用AWS CloudFormation範本建立 KMS 金鑰，請參閱使用AWS CloudFormation者指南AWS::KMS::Key中的〈〉。
- 若要判斷現有 KMS 金鑰是對稱或非對稱，請參閱 識別非對稱 KMS 金鑰。
- 若要透過編寫程式的方式以及在命令列介面操作中使用 KMS 金鑰，您需要有金鑰 ID 或金鑰 ARN。如需詳細說明，請參閱 尋找金鑰 ID 和金鑰 ARN。
- 如需 KMS 金鑰配額的詳細資訊，請參閱 配額。

主題

- 建立 KMS 金鑰的許可
- 建立對稱加密 KMS 金鑰

建立 KMS 金鑰的許可

若要在主控台或使用 API 建立 KMS 金鑰，您必須在 IAM 政策中具有下列許可。盡可能使用條件金鑰以限制許可。例如，您可以在 IAM 政策中使用 kms: KeySpec 條件金鑰，允許主體僅建立對稱加密金鑰。

如需建立金鑰之主體的 IAM 政策範例，請參閱 允許使用者建立 KMS 金鑰。

> **Note**
>
> 授予主體管理標籤和別名的許可時，請務必謹慎。變更標記或別名可允許或拒絕客戶受管金鑰的許可。如需詳細資訊，請參閱 AWS KMS 的 ABAC。

- 公里：CreateKey是必需的。
- kms：需要CreateAlias在主控台中建立 KMS 金鑰，其中每個新 KMS 金鑰都需要別名。
- kms：需要TagResource在建立 KMS 金鑰時新增標籤。
- iam：需CreateServiceLinkedRole要創建多區域主鍵。如需詳細資訊，請參閱 控制對多區域金鑰的存取。

建立 KMS 金鑰不需要公里數：PutKeyPolicy權限。`kms:CreateKey` 許可包含設定初始金鑰政策的許可。但是，您必須在建立 KMS 金鑰時將此許可新增至金鑰政策，以確保您可以控制對 KMS 金鑰的存取。另一種方法是使用該BypassLockoutSafetyCheck參數，這是不推薦的。

KMS 金鑰屬於建立所在的 AWS 帳戶。建立 KMS 金鑰的 IAM 使用者不會被視為金鑰擁有者，而且他們不會自動擁有使用或管理他們建立之 KMS 金鑰的許可。與任何其他主體一樣，金鑰建立者必須透過金鑰政策、IAM 政策或授予取得許可。不過，擁有 `kms:CreateKey` 許可的主體可以設定初始金鑰政策，並授予自己使用或管理金鑰的許可。

建立對稱加密 KMS 金鑰

您可以在 AWS Management Console 中或透過使用 AWS KMS API 建立 KMS 金鑰。

本主題介紹如何使用 AWS KMS 的金鑰資料，為單一區域建立基本 KMS 金鑰 (即對稱加密 KMS 金鑰)。您可使用此 KMS 金鑰來保護 AWS 服務 中的資料。如需有關如何建立其他類型金鑰的說明，請參閱特殊用途金鑰。

建立對稱加密 KMS 金鑰 (主控台)

您可以使用 AWS Management Console 來建立 AWS KMS keys (KMS 金鑰)。

> **Important**
>
> 請勿在別名、說明或標籤包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 選擇建立金鑰。
5. 若要建立對稱加密 KMS 金鑰，在 Key type (金鑰類型) 欄位中，選擇 Symmetric (對稱)。

   如需如何在 AWS KMS 主控台中建立非對稱 KMS 金鑰的資訊，請參閱建立非對稱 KMS 金鑰 (主控台)。
6. 在 Key usage (金鑰用途) 欄位中，系統會自動選取 Encrypt and decrypt (加密和解密) 選項。

   如需有關如何建立可以產生和驗證 MAC 代碼的 KMS 金鑰的詳細資訊，請參閱建立 HMAC KMS 金鑰。
7. 選擇下一步。

   如需有關進階選項的詳細資訊，請參閱特殊用途金鑰。
8. 輸入 KMS 金鑰的別名。別名名稱的開頭不可以是 **`aws/`**。**`aws/`** 字首由 Amazon Web Services 保留，以代表您帳戶中的 AWS 受管金鑰。

   > **Note**
   >
   > 新增、刪除或更新別名可允許或拒絕 KMS 金鑰的許可。如需詳細資訊，請參閱 AWS KMS 的 ABAC 和 使用別名來控制對 KMS 金鑰的存取。

   別名是您可用來識別 KMS 金鑰的顯示名稱。我們建議您選擇別名來表示您計劃保護的資料類型，或您計劃搭配 KMS 金鑰一起使用的應用程式。

   在 AWS Management Console 中建立 KMS 金鑰時需要別名。當您使用該CreateKey操作時，它們是可選的。
9. (選用) 輸入 KMS 金鑰的描述。

   您可以立即新增描述或在任意時間更新，除非金鑰狀態為 `Pending Deletion` 或 `Pending Replica Deletion`。若要加入、變更或刪除現有客戶管理金鑰的描述，請編輯中的描述AWS Management Console或使用UpdateKeyDescription作業。

10. (選用) 輸入標籤索引鍵和選用標籤值。若要將其他標籤新增至 KMS 金鑰，請選擇 Add tag (新增標籤)。

    > **Note**
    >
    > 標記或取消標記 KMS 金鑰可以允許或拒絕 KMS 金鑰的許可。如需詳細資訊，請參閱 AWS KMS 的 ABAC 和 使用標籤來控制對 KMS 金鑰的存取。

    將標籤新增到 AWS 資源時，AWS 會產生成本配置報告，內含按標籤彙總的用量與成本。標籤也可以用來控制 KMS 金鑰的存取。如需標記 KMS 金鑰的詳細資訊，請參閱 標記金鑰 和 AWS KMS 的 ABAC。

11. 選擇下一步。
12. 選取可管理 KMS 金鑰的 IAM 使用者和角色。

    > **Note**
    >
    > 此金鑰政策授予 AWS 帳戶 完全控制此 KMS 金鑰。它允許帳戶管理員使用 IAM 政策授予其他主體管理 KMS 金鑰的許可。如需詳細資訊，請參閱 the section called “預設金鑰政策”。
    >
    > IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

13. (選用) 為了防止選取的 IAM 使用者和角色刪除此 KMS 金鑰，請在頁面底部的 Key deletion (金鑰刪除) 區段中，清除 Allow key administrators to delete this key (允許金鑰管理員刪除此金鑰) 核取方塊。
14. 選擇下一步。
15. 選取可將金鑰用於密碼編譯操作的 IAM 使用者和角色

    > **Note**
    >
    > 此金鑰政策授予 AWS 帳戶 完全控制此 KMS 金鑰。它允許帳戶管理員使用 IAM 政策授予其他主體在密碼編譯操作中使用 KMS 金鑰的許可。如需詳細資訊，請參閱 the section called “預設金鑰政策”。

IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

16. (選用) 您可以允許其他 AWS 帳戶 將此 KMS 金鑰用於密碼編譯操作。若要這樣做，請在頁面底部的其他 AWS 帳戶 區段中，選擇新增另一個 AWS 帳戶，然後輸入外部帳戶的 AWS 帳戶 識別號碼。若要新增多個外部帳戶，請重複此步驟。

> **Note**
>
> 若要允許外部帳戶中的主體使用 KMS 金鑰，外部帳戶的管理員必須建立 IAM 政策來提供這些許可。如需詳細資訊，請參閱 允許其他帳戶中的使用者使用 KMS 金鑰。

17. 選擇下一步。
18. 檢閱您選擇的金鑰設定。您仍然可以返回並變更所有設定。
19. 選擇 Finish (完成) 來建立 KMS 金鑰。

建立對稱加密 KMS 金鑰 (AWS KMS API)

您可以使用該CreateKey操作來創建AWS KMS keys所有類型。以下範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

> **Important**
>
> 請勿在 `Description` 或 `Tags` 欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

以下操作建立了最常用的 KMS 金鑰，即單一區域中由 AWS KMS 產生的對稱加密金鑰。此操作沒有必要參數。但您可能還想要使用 `Policy` 參數指定金鑰政策。您可以隨時變更金鑰原則 (PutKeyPolicy) 並新增選擇性元素，例如說明和標籤。您還可以建立非對稱金鑰、多區域金鑰、匯入金鑰資料中的金鑰，以及自訂金鑰存放區中的金鑰。

此`CreateKey`作業不允許您指定別名，但您可以使用此CreateAlias作業為新 KMS 金鑰建立別名。

以下範例呼叫 `CreateKey` 操作，不帶參數。這個命令會使用所有的預設值。其會使用 AWS KMS 產生的金鑰資料來建立對稱加密 KMS 金鑰。

```
$ aws kms create-key
```

```
{
    "KeyMetadata": {
        "Origin": "AWS_KMS",
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "Description": "",
        "KeyManager": "CUSTOMER",
        "Enabled": true,
        "KeySpec": "SYMMETRIC_DEFAULT",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "KeyState": "Enabled",
        "CreationDate": 1502910355.475,
        "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "AWSAccountId": "111122223333",
        "MultiRegion": false
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ],
    }
}
```

如果您不為您的新 KMS 金鑰指定金鑰政策，則在用其建立新 KMS 金鑰時，CreateKey 套用的預設金鑰政策會不同於主控台套用的預設金鑰政策。

例如，此GetKeyPolicy作業呼叫會傳回套用的金鑰原CreateKey則。它為 AWS 帳戶 提供了 KMS 金鑰的存取權，並允許其建立 KMS 金鑰的 AWS Identity and Access Management (IAM) 政策。如需有關 KMS 金鑰之 IAM 政策和金鑰政策的詳細資訊，請參閱 AWS KMS 的身分驗證與存取控制。

```
$ aws kms get-key-policy --key-id 1234abcd-12ab-34cd-56ef-1234567890ab --policy-name
 default --output text
{
  "Version" : "2012-10-17",
  "Id" : "key-default-1",
  "Statement" : [ {
    "Sid" : "Enable IAM User Permissions",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "arn:aws:iam::111122223333:root"
    },
    "Action" : "kms:*",
    "Resource" : "*"
  } ]
```

```
}
```

使用別名

別名是 AWS KMS key 的易記名稱。例如，別名可讓您將 KMS 金鑰稱為 `test-key`，而不是 `1234abcd-12ab-34cd-56ef-1234567890ab`。

您可以使用別名來識別AWS KMS主控台、作業中和密碼編譯DescribeKey作業 (例如 Encrypt 和 GenerateDataKey. 別名也可以很容易地識別 AWS 受管金鑰。這些 KMS 金鑰的別名一律具有 `aws/<service-name>` 格式。例如，Amazon DynamoDB 的 AWS 受管金鑰 別名是 `aws/dynamodb`。您可以為專案建立類似的別名標準，例如在別名前面加上專案或品類的名稱。

您也可以根據其別名允許和拒絕存取 KMS 金鑰，而不需編輯政策或管理授權。這項功能是對屬性型存取控制 (ABAC) 的 AWS KMS 支援。如需詳細資訊，請參閱 使用別名來控制對 KMS 金鑰的存取。

別名的大部分功能來自於您隨時變更與別名相關聯之 KMS 金鑰的能力。別名可以讓您的程式碼更容易撰寫和維護。例如，假設您使用別名來指代特定的 KMS 金鑰，而且您想要變更 KMS 金鑰。在這種情況下，只要將別名與不同的 KMS 金鑰建立關聯即可。您不需要變更程式碼。

別名也可以更容易在不同的 AWS 區域 中重複使用相同的程式碼。在多個區域中建立具有相同名稱的別名，並將每個別名與其區域中的 KMS 金鑰產生關聯。當程式碼在每個區域中執行時，別名是指該區域中相關聯的 KMS 金鑰。如需範例，請參閱 在應用程式中使用別名。

您可以使用 CreateAliasAPI 或使用AWS CloudFormation範本，在AWS KMS主控台中為 KMS 金鑰建立別名。

AWS KMS API 可完全控制每個帳戶和區域中的別名。API 包括建立別名 (CreateAlias)、檢視別名和別名 ARN (ListAliases)、變更與別名 () 相關聯的 KMS 金鑰，以及刪除別名 (DeleteAlias) 的作業。UpdateAlias如需管理別名之多種程式設計語言的範例，請參閱 the section called “處理別名”。

下列資源可協助您進一步了解：

- 如需 KMS 金鑰識別符 (包括別名) 的相關資訊，請參閱 金鑰識別碼 (KeyId)。
- 如需使用AWS CloudFormation範本建立 KMS 金鑰別名的說明，請參閱使用AWS CloudFormation者指南AWS::KMS::Alias中的〈〉。
- 如需尋找與 KMS 金鑰相關聯的別名，請參閱 尋找別名和別名 ARN
- 如需別名資源配額和與別名相關聯 API 操作之費率配額的相關資訊，請參閱 配額。
- 如需建立和管理別名之多種程式設計語言的範例，請參閱 處理別名。

主題

關於別名

了解別名如何在 AWS KMS 運作。

別名是獨立的 AWS 資源

別名不是 KMS 金鑰的屬性。您對別名所採取的動作不會影響其關聯的 KMS 金鑰。您可以為 KMS 金鑰建立別名，然後更新別名，讓其與不同的 KMS 金鑰相關聯。您甚至可以刪除別名，而不會對相關聯的 KMS 金鑰造成任何影響。不過，如果您刪除 KMS 金鑰，則會刪除與該 KMS 金鑰相關聯的所有別名。

如果您在 IAM 政策中指定別名作為資源，則政策會指代別名，而不是關聯的 KMS 金鑰。

每個別名都有兩種格式

在建立別名時，請指定別名名稱。AWS KMS 為您建立別名 ARN。

- 別名 ARN 是唯一可識別別名的 Amazon Resource Name (ARN)。

```
# Alias ARN
arn:aws:kms:us-west-2:111122223333:alias/<alias-name>
```

- 在帳戶和區域中，別名名稱必須是唯一的。在 AWS KMS API 中，別名名稱一律以 `alias/` 為字首。字首在 AWS KMS 主控台中省略了。

```
# Alias name
alias/<alias-name>
```

別名並非秘密

別名可以在 CloudTrail 日誌和其他輸出中以明文顯示。請勿在別名名稱包含機密或敏感資訊。

每個別名一次都與一個 KMS 金鑰相關聯

別名及其 KMS 金鑰必須位於相同的帳戶和區域中。

您可以將別名與相同 AWS 帳戶 和區域中的任何客戶受管金鑰相關聯。不過，您沒有將別名與 AWS 受管金鑰 相關聯的許可。

例如，此ListAliases輸出顯示test-key別名只與一個目標 KMS 金鑰相關聯，該金鑰由TargetKeyId屬性表示。

```
{
     "AliasName": "alias/test-key",
     "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/test-key",
     "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
     "CreationDate": 1593622000.191,
     "LastUpdatedDate": 1593622000.191
}
```

多個別名可以與相同的 KMS 金鑰建立關聯

例如，您可以將 test-key 和 project-key 別名與相同的 KMS 金鑰相關聯。

```
{
     "AliasName": "alias/test-key",
     "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/test-key",
     "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
     "CreationDate": 1593622000.191,
     "LastUpdatedDate": 1593622000.191
},
{
     "AliasName": "alias/project-key",
     "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/project-key",
     "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
     "CreationDate": 1516435200.399,
     "LastUpdatedDate": 1516435200.399
}
```

別名在帳戶和區域中必須是唯一的。

例如，您在每個帳戶和區域只能有一個 test-key 別名。別名區分大小寫，但只有大小寫不同的別名很容易出錯。您無法變更別名名稱。但是，您可以刪除別名，並使用所需名稱建立新別名。

但是，您可以在不同區域中使用相同的名稱建立別名。

例如，您可以在美國東部 (維吉尼亞北部) 有一個 finance-key 別名和在歐洲 (法蘭克福) 有一個 finance-key 別名。每個別名都會與其區域中的 KMS 金鑰相關聯。如果您的程式碼引用了類似 alias/finance-key 的別名名稱，則可以在多個區域中執行。在每個區域中，它會使用不同的 KMS 金鑰。如需詳細資訊，請參閱 在應用程式中使用別名。

您可以變更與別名相關聯的 KMS 金鑰

您可以使用此UpdateAlias作業將別名與不同的 KMS 金鑰建立關聯。例如，如果 finance-key 別名與 1234abcd-12ab-34cd-56ef-1234567890ab KMS 金鑰相關聯，則您可以對其進行更新，使其與 0987dcba-09fe-87dc-65ba-ab0987654321 KMS 金鑰相關聯。

不過，目前的和新的 KMS 金鑰必須是相同類型 (兩個皆為對稱或兩個皆為非對稱或兩個皆為 HMAC)，並且必須具有相同的金鑰使用情形 (ENCRYPT_DECRYPT 或 SIGN_VERIFY 或 GENERATE_VERIFY_MAC)。此限制可避免使用別名之程式碼中的錯誤。如果您必須將別名與不同類型的金鑰建立關聯，且已降低風險，則可以刪除並重新建立別名。

某些 KMS 金鑰沒有別名

在 AWS KMS 主控台建立 KMS 金鑰時，您必須指定新的別名。但是當您使用CreateKey作業建立 KMS 金鑰時，不需要別名。此外，您可以使用此UpdateAlias作業變更與別名相關聯的 KMS 金鑰，以及刪除別名的DeleteAlias作業。因此，某些 KMS 金鑰可能有數個別名，有些金鑰可能沒有。

AWS 在您的帳戶中建立別名

AWS 在您的帳戶中建立 AWS 受管金鑰 別名。這些別名有 alias/aws/*<service-name>* 格式的名稱，例如 alias/aws/s3。

一些 AWS 別名沒有 KMS 金鑰。當您開始使用服務時，這些預先定義的別名通常與 AWS 受管金鑰相關聯。

使用別名來識別 KMS 金鑰

您可以使用別名或別名 ARN 來識別密碼編譯作業中的 KMS 金鑰DescribeKey、和。GetPublicKey(如果 KMS 金鑰位於不同的 AWS 帳戶，您必須使用其金鑰 ARN 或別名 ARN。) 別名在其他 AWS KMS 操作中不是 KMS 金鑰的有效識別符。如需每個 AWS KMS API 操作之有效金鑰識別符的詳細資訊，請參閱 AWS Key Management Service API 參考中的 KeyId 參數描述。

您不能使用別名名稱或別名 ARN 來識別 IAM 政策中的 KMS 金鑰。若要根據其別名控制 KMS 金鑰的存取權，請使用 kms: RequestAlias 或 kms: ResourceAliases 條件金鑰。如需詳細資訊，請參閱 AWS KMS 的 ABAC。

管理別名

授權的使用者可以建立、檢視和刪除別名。您也可以更新別名，將現有的別名關聯至不同的 KMS 金鑰。

主題

建立別名

您可以在 AWS KMS 主控台或使用 AWS KMS API 操作建立別名。

別名必須是 1-256 個字元的字串。它只能包含英數字元、斜線 (/)、底線 (_) 和破折號 (-)。客戶受管金鑰的別名名稱不能以 `alias/aws/` 開頭。此 `alias/aws/` 字首已保留供 AWS 受管金鑰 使用。

您可以為新的 KMS 金鑰或現有的 KMS 金鑰建立別名。您可以新增別名，以便在專案或應用程式中使用特定的 KMS 金鑰。

建立別名 (主控台)

在 AWS KMS 主控台中建立 KMS 金鑰時，您必須為新的 KMS 金鑰建立別名。若要為現有的 KMS 金鑰建立別名，請使用 KMS 金鑰詳細資訊頁面上的 Aliases (別名) 索引標籤。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。您無法管理 AWS 受管金鑰 或 AWS 擁有的金鑰 的別名。
4. 在資料表中，選擇 KMS 金鑰的金鑰 ID 或別名。然後，在 KMS 金鑰詳細資訊頁面上，選擇 Aliases (別名) 索引標籤。

如果 KMS 金鑰有多個別名，Aliases (別名) 資料行會顯示一個別名和別名摘要，例如 (+n 等)。選擇別名摘要會直接帶您前往 KMS 金鑰詳細資訊頁面上的 Aliases (別名) 索引標籤。

5. 在 Aliases (別名) 索引標籤中，選擇 Create alias (建立別名)。輸入別名名稱，然後選擇 Create alias (建立別名)。

> **Important**
>
> 請勿在此欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，此欄位可能會以純文字顯示。

> **Note**
>
> 請勿加入 `alias/` 字首。主控台會自動執行此操作。如果您輸入 `alias/ExampleAlias`，實際的別名名稱將是 `alias/alias/ExampleAlias`。

建立別名 (AWS KMS API)

若要建立別名，請使用此CreateAlias作業。與在主控台中建立 KMS 金鑰的程序不同，此CreateKey作業不會為新的 KMS 金鑰建立別名。

> **Important**
>
> 請勿在此欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，此欄位可能會以純文字顯示。

您可以使用 `CreateAlias` 操作為沒有別名的新 KMS 金鑰建立別名。您也可以使用 `CreateAlias` 操作將別名新增至任何現有的 KMS 金鑰，或重新建立意外刪除的別名。

在 AWS KMS API 操作中，別名必須以 `alias/` 開頭，其後跟隨名稱，例如 `alias/ExampleAlias`。別名在帳戶和區域中必須是唯一的。若要尋找已在使用中的別名，請使用此ListAliases作業。別名名稱區分大小寫。

`TargetKeyId` 可以為相同 AWS 區域 中的任一項客戶受管金鑰。若要識別 KMS 金鑰，請使用它的金鑰 ID 或金鑰 ARN。您無法使用另一個別名。

下列範例會建立 example-key 別名，並將其與指定的 KMS 金鑰建立關聯。這些範例皆是採用 AWS Command Line Interface (AWS CLI)。如需多種程式設計語言的範例，請參閱處理別名。

```
$ aws kms create-alias \
    --alias-name alias/example-key \
    --target-key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

CreateAlias 不會傳回任何輸出。若要查看新別名，請使用 ListAliases 操作。如需詳細資訊，請參閱 檢視別名 (AWS KMS API)。

檢視別名

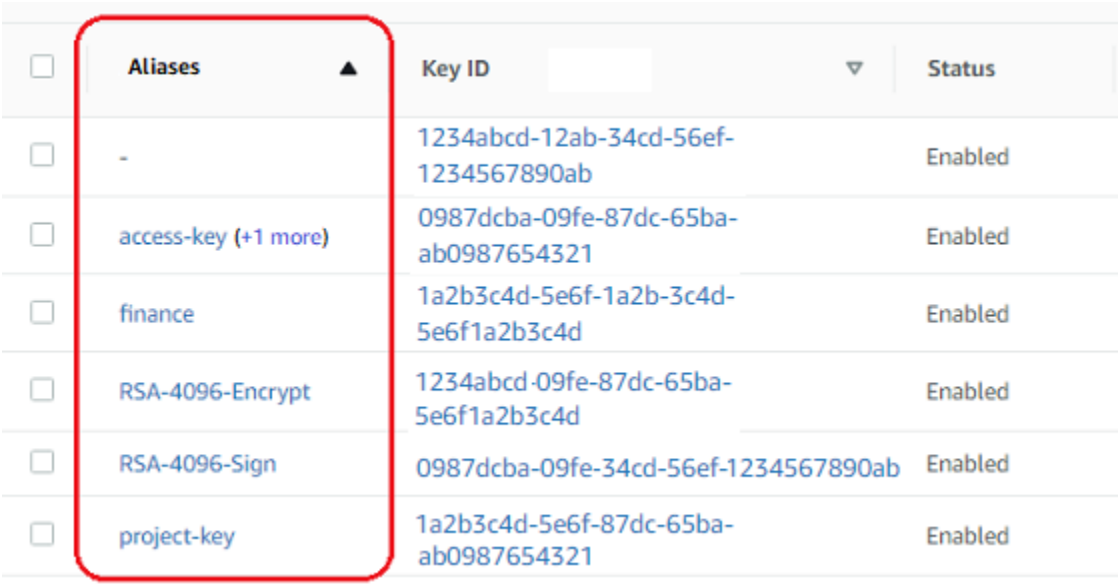
別名可讓您輕鬆辨識 AWS KMS 主控台中的 KMS 金鑰。您可以在AWS KMS主控台或使用ListAliases作業來檢視 KMS 金鑰的別名。傳回 KMS 金鑰屬性的DescribeKey作業不包含別名。

檢視別名 (主控台)

AWS KMS 主控台中的客戶受管金鑰和 AWS 受管金鑰 的頁面會顯示與每個 KMS 金鑰相關聯的別名。您也可依其別名搜尋、排序和篩選 KMS 金鑰。

AWS KMS 主控台的下圖會顯示範例帳戶客戶受管金鑰頁面上的別名。如圖所示，某些 KMS 金鑰無別名。

當 KMS 金鑰具有多個別名時，Aliases (別名) 資料行會顯示一個別名，而別名摘要 (+n 等)。別名摘要會顯示與 KMS 金鑰相關聯的其他別名數目和顯示 Aliases (別名) 索引標籤上 KMS 金鑰之所有別名的連結。

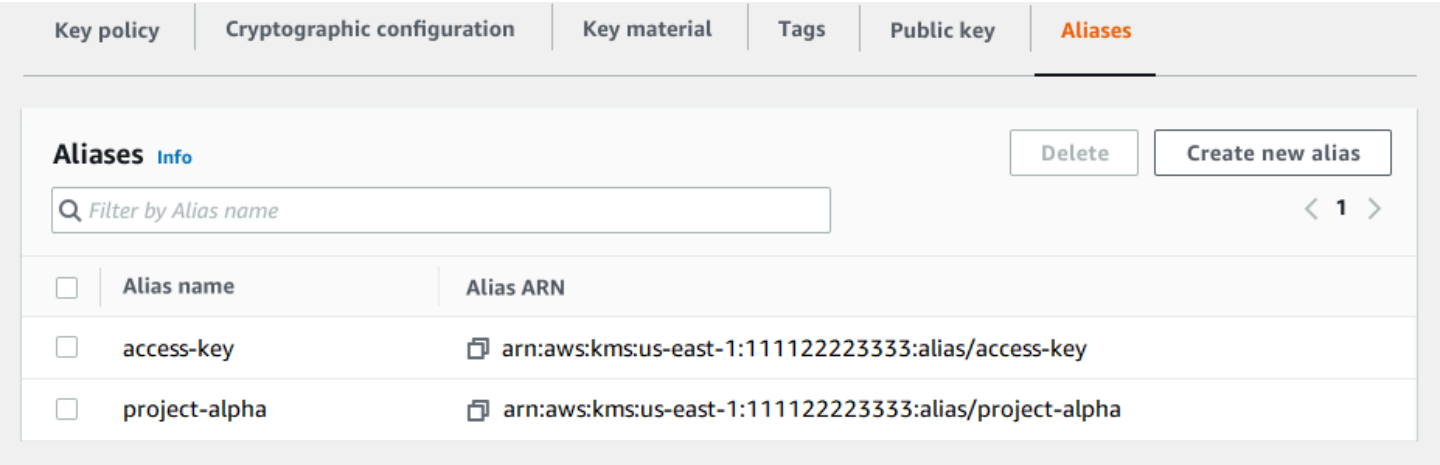


每個 KMS 金鑰的詳細資訊頁面 Aliases (別名) 索引標籤會顯示 AWS 帳戶 和區域中 KMS 金鑰之所有別名的別名名稱和別名 ARN。您也可以使用 Aliases (別名) 索引標籤建立別名和刪除別名。

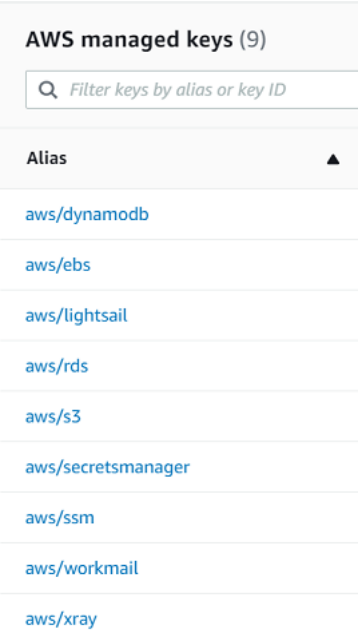
若要尋找 KMS 金鑰之所有別名的別名名稱和別名 ARN，請使用 Aliases (別名) 索引標籤。

- 若要直接移至 Aliases (別名) 欄中的 Aliases (別名) 索引標籤，請選擇別名摘要 (+n 等)。只有在 KMS 金鑰具有多個別名時，才會顯示別名摘要。
- 或者，選擇 KMS 金鑰的別名或金鑰 ID (這會開啟 KMS 金鑰的詳細資訊頁面)，然後選擇 Aliases (別名) 索引標籤。索引標籤位於 General configuration (一般組態) 區段。

下圖顯示範例 KMS 金鑰的 Aliases (別名) 索引標籤。



您可以使用別名來識別 AWS 受管金鑰，如此範例 AWS 受管金鑰 頁面中所示。AWS 受管金鑰 的別名一律具有以下格式：aws/*<service-name>*。例如，Amazon DynamoDB 的 AWS 受管金鑰 別名是 aws/dynamodb。



檢視別名 (AWS KMS API)

此ListAliases作業會傳回帳戶與區域中別名的別名和別名 ARN。輸出包含 AWS 受管金鑰 和客戶受管金鑰的別名。AWS 受管金鑰 別名具有格式 aws/*<service-name>* ，例如 aws/dynamodb。

回應可能也包含沒有 TargetKeyId 欄位的別名。這些是 AWS 已建立但尚未關聯至 KMS 金鑰的預先定義別名。

```
$ aws kms list-aliases
{
    "Aliases": [
        {
            "AliasName": "alias/access-key",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/access-key",
            "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
            "CreationDate": 1516435200.399,
            "LastUpdatedDate": 1516435200.399
        },
        {
            "AliasName": "alias/ECC-P521-Sign",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/ECC-P521-Sign",
            "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "CreationDate": 1693622000.704,
            "LastUpdatedDate": 1693622000.704
        },
        {
```

```
            "AliasName": "alias/ImportedKey",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/ImportedKey",
            "TargetKeyId": "1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d",
            "CreationDate": 1493622000.704,
            "LastUpdatedDate": 1521097200.235
        },
        {
            "AliasName": "alias/finance-project",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/finance-project",
            "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
            "CreationDate": 1604958290.014,
            "LastUpdatedDate": 1604958290.014
        },
        {
            "AliasName": "alias/aws/dynamodb",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/aws/dynamodb",
            "TargetKeyId": "0987ab65-43cd-21ef-09ab-87654321cdef",
            "CreationDate": 1521097200.454,
            "LastUpdatedDate": 1521097200.454
        },
        {
            "AliasName": "alias/aws/ebs",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/aws/ebs",
            "TargetKeyId": "abcd1234-09fe-ef90-09fe-ab0987654321",
            "CreationDate": 1466518990.200,
            "LastUpdatedDate": 1466518990.200
        }
    ]
}
```

若要取得與特定 KMS 金鑰關聯的所有別名，請使用 `ListAliases` 操作的選用 `KeyId` 參數。`KeyId` 參數採用 KMS 金鑰的金鑰 ID 或金鑰 ARN。

此範例會將所有別名與 `0987dcba-09fe-87dc-65ba-ab0987654321` KMS 金鑰關聯。

```
$ aws kms list-aliases --key-id 0987dcba-09fe-87dc-65ba-ab0987654321
{
    "Aliases": [
        {
            "AliasName": "alias/access-key",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/access-key",
            "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
            "CreationDate": "2018-01-20T15:23:10.194000-07:00",
```

```
            "LastUpdatedDate": "2018-01-20T15:23:10.194000-07:00"
        },
        {
            "AliasName": "alias/finance-project",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/finance-project",
            "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
            "CreationDate": 1604958290.014,
            "LastUpdatedDate": 1604958290.014
        }
    ]
}
```

KeyId 參數不會採用萬用字元，但您可以使用程式設計語言的功能來篩選回應。

例如，下列 AWS CLI 命令只會取得 AWS 受管金鑰 的別名。

```
$ aws kms list-aliases --query 'Aliases[?starts_with(AliasName, `alias/aws/`)]'
```

下列命令只會取得 access-key 別名。別名名稱區分大小寫。

```
$ aws kms list-aliases --query 'Aliases[?AliasName==`alias/access-key`]'
[
    {
        "AliasName": "alias/access-key",
        "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/access-key",
        "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
        "CreationDate": "2018-01-20T15:23:10.194000-07:00",
        "LastUpdatedDate": "2018-01-20T15:23:10.194000-07:00"
    }
]
```

更新別名

由於別名是獨立的資源，因此您可以變更與別名相關聯的 KMS 金鑰。例如，如果test-key別名與一個 KMS 金鑰相關聯，您可以使用該UpdateAlias作業將其與不同的 KMS 金鑰建立關聯。這是手動輪換 KMS 金鑰的幾種方式之一，不會變更其金鑰材料。您也可以更新 KMS 金鑰，以便針對新資源使用 KMS 金鑰的應用程式現在使用不同的 KMS 金鑰。

您無法更新 AWS KMS 主控台中的別名。您也無法使用 UpdateAlias (或任何其他操作) 來變更別名名稱。若要變更別名名稱，請刪除目前別名，然後為 KMS 金鑰建立新的別名。

更新別名時，當前 KMS 金鑰和新 KMS 金鑰必須為相同類型 (兩者皆為對稱或非對稱或 HMAC)。也必須有相同的金鑰用途 (ENCRYPT_DECRYPT 或 SIGN_VERIFY 或 GENERATE_VERIFY_MAC)。這項限制可防止使用別名的程式碼中出現密碼編譯錯誤。

下列範例會使用ListAliases作業開始，顯示test-key別名目前與 KMS 金鑰相關聯1234abcd-12ab-34cd-56ef-1234567890ab。

```
$ aws kms list-aliases --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
{
    "Aliases": [
        {
            "AliasName": "alias/test-key",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/test-key",
            "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "CreationDate": 1593622000.191,
            "LastUpdatedDate": 1593622000.191
        }
    ]
}
```

接下來，它使用 UpdateAlias 操作來將與 test-key 別名關聯的 KMS 金鑰變更為 KMS 金鑰 0987dcba-09fe-87dc-65ba-ab0987654321。您不需要指定目前關聯的 KMS 金鑰，只需指定新的 (「目標」) KMS 金鑰。別名名稱區分大小寫。

```
$ aws kms update-alias --alias-name 'alias/test-key' --target-key-id
 0987dcba-09fe-87dc-65ba-ab0987654321
```

若要驗證別名是否已與目標 KMS 金鑰關聯，請再次使用 ListAliases 操作。此 AWS CLI 命令會使用 --query 參數，以便只取得 test-key 別名。TargetKeyId 和 LastUpdatedDate 欄位會更新。

```
$ aws kms list-aliases --query 'Aliases[?AliasName==`alias/test-key`]'
[
    {
        "AliasName": "alias/test-key",
        "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/test-key",
        "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
        "CreationDate": 1593622000.191,
        "LastUpdatedDate": 1604958290.154
    }
]
```

刪除別名

您可以在AWS KMS控制台中刪除別名，也可以使用DeleteAlias操作來刪除別名。在刪除別名前，請確認別名未處於使用中狀態。雖然刪除別名並不會影響相關聯的 KMS 金鑰，但它可能會對使用別名的任何應用程式造成問題。如果不小心刪除了別名，您可以建立具有相同名稱的新別名，並將其與相同或不同的 KMS 金鑰產生關聯。

如果刪除 KMS 金鑰，則會刪除與該 KMS 金鑰相關聯的所有別名。

刪除別名 (主控台)

若要在 AWS KMS 主控台中刪除別名，請使用 KMS 金鑰之詳細資訊頁面上的 Aliases (別名) 索引標籤。您可以一次刪除 KMS 金鑰的多個別名。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。您無法管理 AWS 受管金鑰 或 AWS 擁有的金鑰 的別名。
4. 在資料表中，選擇 KMS 金鑰的金鑰 ID 或別名。然後，在 KMS 金鑰詳細資訊頁面上，選擇 Aliases (別名) 索引標籤。

   如果 KMS 金鑰有多個別名，Aliases (別名) 資料行會顯示一個別名和別名摘要，例如 (+n 等)。選擇別名摘要會直接帶您前往 KMS 金鑰詳細資訊頁面上的 Aliases (別名) 索引標籤。
5. 在 Aliases (別名) 索引標籤上，選取您要刪除之別名旁邊的核取方塊。然後選擇 Delete (刪除)。

刪除別名 (AWS KMS API)。

若要刪除別名，請使用此DeleteAlias作業。此操作一次會刪除一個別名。別名名稱區分大小寫，且前面必須加上 `alias/` 字首。

例如，以下命令會刪除 `test-key` 別名。此命令不會傳回任何輸出。

```
$ aws kms delete-alias --alias-name alias/test-key
```

若要確認已刪除別名，請使用此ListAliases作業。下列命令會使用 AWS CLI 中的 `--query` 參數以只取得 `test-key` 別名。回應中的空括號表示 `ListAliases` 回應中未包含 `test-key` 別名。若要消除括號，請使用 `--output text` 參數和值。

```
$ aws kms list-aliases --query 'Aliases[?AliasName==`alias/test-key`]'
[]
```

在應用程式中使用別名

您可以使用別名來代表應用程式程式碼中的 KMS 金鑰。AWS KMS加密操作中的KeyId參數 DescribeKey，並GetPublicKey接受別名或別名 ARN。

例如，下列 GenerateDataKey 命令會使用別名名稱 (alias/finance) 來識別 KMS 金鑰。別名名稱為 KeyId 參數的值。

```
$ aws kms generate-data-key --key-id alias/finance --key-spec AES_256
```

如果 KMS 金鑰位於不同 AWS 帳戶，則您必須在這些操作中使用金鑰 ARN 或別名 ARN。使用別名 ARN 時，請記住 KMS 金鑰的別名是在擁有 KMS 金鑰的帳戶中定義的，且在每個區域中可能會有所不同。如需尋找別名 ARN 的說明，請參閱 尋找別名和別名 ARN。

例如，下列 GenerateDataKey 命令會使用不位於呼叫者帳戶中的 KMS 金鑰。ExampleAlias 別名與指定帳戶和區域中的 KMS 金鑰相關聯。

```
$ aws kms generate-data-key --key-id arn:aws:kms:us-west-2:444455556666:alias/
ExampleAlias --key-spec AES_256
```

其中一個最強大的別名用途是在多個 AWS 區域 中執行之應用程式中使用。例如，您可能有使用 RSA 非對稱 KMS 金鑰的全域應用程式，以進行簽章和驗證。

- 在美國西部 (奧勒岡) (us-west-2) 中，您想要使用 arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab。
- 在歐洲 (法蘭克福) (eu-central-1) 中，您想要使用 arn:aws:kms:eu-central-1:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321
- 在亞太區域 (新加坡) (ap-southeast-1) 中，您想要使用 arn:aws:kms:ap-southeast-1:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d。

您可以在每個區域中建立不同版本的應用程式，或使用字典或 switch 陳述式為每個區域選取正確的 KMS 金鑰。但是，更輕鬆的方式是在每個區域中建立具有相同別名名稱的別名。請記住，別名名稱區分大小寫。

```
aws --region us-west-2 kms create-alias \
```

```
    --alias-name alias/new-app \
    --key-id arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab

aws --region eu-central-1 kms create-alias \
    --alias-name alias/new-app \
    --key-id arn:aws:kms:eu-central-1:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321

aws --region ap-southeast-1 kms create-alias \
    --alias-name alias/new-app \
    --key-id arn:aws:kms:ap-
southeast-1:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d
```

然後，在程式碼中使用別名。當程式碼在每個區域中執行時，別名會參照其在該區域中關聯的 KMS 金鑰。例如，此程式碼會使用別名名稱呼叫 Sign 操作。

```
aws kms sign --key-id alias/new-app \
    --message $message \
    --message-type RAW \
    --signing-algorithm RSASSA_PSS_SHA_384
```

不過，風險是有可能會刪除或更新別名，以與不同的 KMS 金鑰相關聯。在這種情況下，應用程式嘗試使用別名名稱驗證簽章將會失敗，您可能需要重新建立或更新別名。

若要降低此風險，請謹慎授予委託人管理您在應用程式中使用之別名的許可。如需詳細資訊，請參閱控制對別名的存取。

對於在多個 AWS 區域 中加密資料的應用程式，還有幾個其他解決方案，包括 AWS Encryption SDK。

控制對別名的存取

當您建立或變更別名時，會影響別名及其關聯的 KMS 金鑰。因此，管理別名的委託人必須具有對別名和所有受影響之 KMS 金鑰呼叫別名操作的許可。您可以使用金鑰政策、IAM 政策和授權來提供這些許可。

> **Note**
>
> 授予委託人管理標籤和別名的許可時，請務必謹慎。變更標記或別名可允許或拒絕客戶受管金鑰的許可。如需詳細資訊，請參閱 AWS KMS 的 ABAC 和 使用別名來控制對 KMS 金鑰的存取。

如需控制對所有 AWS KMS 操作之存取的詳細資訊，請參閱 許可參考。

建立和管理別名的許可如下所示。

公里：CreateAlias

若要建立別名，委託人需要別名和相關聯的 KMS 金鑰的下列許可。

- 別名的 kms:CreateAlias。在連接至允許建立別名之委託人的 IAM 政策中提供此許可。

  下列範例政策陳述式會指定 Resource 元素中的特定別名。但是，您可以列出多個別名 ARN 或指定別名模式，例如 "test*"。您也可以指定 "*" 的 Resource 值以允許委託人在帳戶和區域中建立任何別名。建立別名的許可也可以包含在帳戶和區域中所有資源的 kms:Create* 許可中。

```
{
  "Sid": "IAMPolicyForAnAlias",
  "Effect": "Allow",
  "Action": [
    "kms:CreateAlias",
    "kms:UpdateAlias",
    "kms:DeleteAlias"
  ],
  "Resource": "arn:aws:kms:us-west-2:111122223333:alias/test-key"
}
```

- KMS 金鑰的 kms:CreateAlias。必須在金鑰政策或在從金鑰政策委派的 IAM 政策中提供此許可。

```
{
  "Sid": "Key policy for 1234abcd-12ab-34cd-56ef-1234567890ab",
  "Effect": "Allow",
  "Principal": {"AWS": "arn:aws:iam::111122223333:user/KMSAdminUser"},
  "Action": [
    "kms:CreateAlias",
    "kms:DescribeKey"
  ],
  "Resource": "*"
}
```

您可以使用條件索引鍵來限制可與別名建立關聯的 KMS 金鑰。例如，您可以使用 kms: KeySpec 條件金鑰，允許主體僅在非對稱 KMS 金鑰上建立別名。如需您可用於限制 KMS 金鑰資料之 kms:CreateAlias 許可的條件索引鍵完整清單，請參閱 AWS KMS 權限。

公里：ListAliases

若要列出帳戶和區域中的別名，委託人必須具有 IAM 政策的 kms:ListAliases 許可。由於此政策與任何特定 KMS 金鑰或別名資源無關，因此政策中資源元素的值必須為 "*"。

例如，下列 IAM 政策陳述式會授予委託人許可，以列出帳戶和區域中的所有 KMS 金鑰和別名。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "kms:ListKeys",
      "kms:ListAliases"
    ],
    "Resource": "*"
  }
}
```

公里：UpdateAlias

若要變更與別名相關聯的 KMS 金鑰，委託人需要三個許可元素：一個用於別名、一個用於當前 KMS 金鑰，另一個用於新的 KMS 金鑰。

例如，假設您想要將 test-key 別名從金鑰 ID 為 1234abcd-12ab-34cd-56ef-1234567890ab 的 KMS 金鑰變更為金鑰 ID 為 0987dcba-09fe-87dc-65ba-ab0987654321 的 KMS 金鑰。在這種情況下，請包含類似於本節範例的政策陳述式。

- 別名的 kms:UpdateAlias。您可以在連接至委託人的 IAM 政策中提供此許可。下列 IAM 政策會指定特定別名。但是，您可以列出多個別名 ARN 或指定別名模式，例如 "test*"。您也可以指定 "*" 的 Resource 值以允許委託人更新帳戶和區域中的任何別名。

```
{
  "Sid": "IAMPolicyForAnAlias",
  "Effect": "Allow",
  "Action": [
    "kms:UpdateAlias",
    "kms:ListAliases",
```

```
    "kms:ListKeys"
  ],
  "Resource": "arn:aws:kms:us-west-2:111122223333:alias/test-key"
}
```

- 目前與別名相關聯之 KMS 金鑰的 kms:UpdateAlias。必須在金鑰政策或在從金鑰政策委派的 IAM 政策中提供此許可。

```
{
  "Sid": "Key policy for 1234abcd-12ab-34cd-56ef-1234567890ab",
  "Effect": "Allow",
  "Principal": {"AWS": "arn:aws:iam::111122223333:user/KMSAdminUser"},
  "Action": [
    "kms:UpdateAlias",
    "kms:DescribeKey"
  ],
  "Resource": "*"
}
```

- 操作與別名相關聯之 KMS 金鑰的 kms:UpdateAlias。必須在金鑰政策或在從金鑰政策委派的 IAM 政策中提供此許可。

```
{
  "Sid": "Key policy for 0987dcba-09fe-87dc-65ba-ab0987654321",
  "Effect": "Allow",
  "Principal": {"AWS": "arn:aws:iam::111122223333:user/KMSAdminUser"},
  "Action": [
    "kms:UpdateAlias",
    "kms:DescribeKey"
  ],
  "Resource": "*"
}
```

您可以使用條件索引鍵來限制 UpdateAlias 操作中的其中一個 KMS 金鑰，或者兩個金鑰皆限制。例如，您可以使用 kms: ResourceAliases 條件金鑰，讓主體只有在目標 KMS 金鑰已有特定別名時才更新別名。如需您可用於限制 KMS 金鑰資源之 kms:UpdateAlias 許可的條件索引鍵完整清單，請參閱 AWS KMS 權限。

公里：DeleteAlias

若要刪除別名，委託人需要別名和相關聯的 KMS 金鑰的許可。

一如往常，在授予委託人刪除資源的許可時，您應該小心謹慎。不過，刪除別名並不會影響相關聯的 KMS 金鑰。雖然這可能會在依賴別名的應用程式中造成故障，但是如果錯誤地刪除了別名，則您可以重新建立別名。

- 別名的 kms:DeleteAlias。在連接至允許刪除別名之委託人的 IAM 政策中提供此許可。

  下列範例政策陳述式會指定 Resource 元素中的別名。但是，您可以列出多個別名 ARN 或指定別名模式，例如 "test*"，您也可以指定 "*" 的 Resource 值以允許委託人刪除帳戶和區域中的任何別名。

```
{
  "Sid": "IAMPolicyForAnAlias",
  "Effect": "Allow",
  "Action": [
    "kms:CreateAlias",
    "kms:UpdateAlias",
    "kms:DeleteAlias"
  ],
  "Resource": "arn:aws:kms:us-west-2:111122223333:alias/test-key"
}
```

- 相關聯 KMS 金鑰的 kms:DeleteAlias。必須在金鑰政策或在從金鑰政策委派的 IAM 政策中提供此許可。

```
{
  "Sid": "Key policy for 1234abcd-12ab-34cd-56ef-1234567890ab",
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:user/KMSAdminUser"
  },
  "Action": [
    "kms:CreateAlias",
    "kms:UpdateAlias",
    "kms:DeleteAlias",
    "kms:DescribeKey"
  ],
  "Resource": "*"
}
```

限制別名許可

當資源為 KMS 金鑰時，您可以使用條件索引鍵來限制別名許可。例如，以下 IAM 政策允許在特定帳戶和區域中對 KMS 金鑰執行別名操作。不過，它會使用 kms: KeyOrigin 條件金鑰，進一步限制權限限制在具有金鑰材料的 KMS 金鑰AWS KMS。

如需可用來限制 KMS 金鑰資源別名許可的條件索引鍵完整清單，請參閱 AWS KMS 權限。

```
{
  "Sid": "IAMPolicyKeyPermissions",
  "Effect": "Allow",
  "Resource": "arn:aws:kms:us-west-2:111122223333:key/*",
  "Action": [
    "kms:CreateAlias",
    "kms:UpdateAlias",
    "kms:DeleteAlias"
  ],
  "Condition": {
    "StringEquals": {
      "kms:KeyOrigin": "AWS_KMS"
    }
  }
}
```

您無法在資源為別名的政策陳述式中使用條件索引鍵。若要限制委託人可以管理的別名，請使用 IAM 政策陳述式 (可控制對別名的存取) 之 Resource 元素的值。例如，下列政策陳述式允許委託人建立、更新或刪除 AWS 帳戶 和區域中的任何別名，除非別名以 Restricted 開頭。

```
{
  "Sid": "IAMPolicyForAnAliasAllow",
  "Effect": "Allow",
  "Action": [
    "kms:CreateAlias",
    "kms:UpdateAlias",
    "kms:DeleteAlias"
  ],
  "Resource": "arn:aws:kms:us-west-2:111122223333:alias/*"
},
{
  "Sid": "IAMPolicyForAnAliasDeny",
  "Effect": "Deny",
  "Action": [
```

```
    "kms:CreateAlias",
    "kms:UpdateAlias",
    "kms:DeleteAlias"
  ],
  "Resource": "arn:aws:kms:us-west-2:111122223333:alias/Restricted*"
}
```

使用別名來控制對 KMS 金鑰的存取

您可以根據與 KMS 金鑰相關聯的別名來控制對 KMS 金鑰的存取。若要這麼做，請使用 kms:RequestAlias 和 kms: ResourceAliases 條件金鑰。這項功能是對屬性型存取控制 (ABAC) 的 AWS KMS 支援。

`kms:RequestAlias` 條件索引鍵允許或拒絕根據請求中的別名存取 KMS 金鑰。`kms:ResourceAliases` 條件索引鍵會根據與 KMS 金鑰相關聯的別名，允許或拒絕存取 KMS 金鑰。

這些功能不允許您使用政策陳述式中 `resource` 元素的別名來識別 KMS 金鑰。當別名是 `resource` 元素的值時，政策會套用至別名資源，而不會套用至可能與其相關聯的任何 KMS 金鑰。

> **Note**
>
> 可能最多需要五分鐘才能將標籤和別名變更體現在 KMS 金鑰授權上。最近的變更可能會在 API 操作中可見，然後才會影響授權。

使用別名控制 KMS 金鑰的存取時，請考慮下列事項：

- 使用別名來強化最低權限存取的最佳實務。僅為 IAM 委託人提供他們必須使用或管理之 KMS 金鑰所需的許可。例如，使用別名來識別專案所使用的 KMS 金鑰。然後授予專案小組僅將 KMS 金鑰與專案別名搭配使用的許可。
- 要謹慎地授予委託人 `kms:CreateAlias`、`kms:UpdateAlias` 或 `kms:DeleteAlias` 許可，讓其新增、編輯和刪除別名。當您使用別名來控制對 KMS 金鑰的存取時，變更別名可授予委託人使用 KMS 金鑰 (否則其沒有使用的許可) 的許可。它也可以拒絕存取其他委託人執行任務所需的 KMS 金鑰。
- 檢閱您 AWS 帳戶 的委託人，目前具有管理別名和調整許可的許可 (如有必要)。沒有變更主要政策或建立授權之許可的重要管理員可以控制 KMS 金鑰的存取，如果他們擁有管理別名的許可。

例如，重要管理員的主控台預設重要政策會包括 kms:CreateAlias、kms:DeleteAlias 和 kms:UpdateAlias 許可。IAM 政策可能會為您的 AWS 帳戶 中的所有 KMS 金鑰提供別名許可。例如，AWSKeyManagementServicePowerUser受管理的原則允許主體建立、刪除和列出所有 KMS 金鑰的別名，但不更新它們。

- 在設定取決於別名的政策之前，請先檢閱 AWS 帳戶 中 KMS 金鑰的別名。請確定您的政策僅適用於您想要包含的別名。使用CloudTrail 記錄和CloudWatch 警示來警示您可能會影響 KMS 金鑰存取的別名變更。此外，ListAliases回應還包括每個別名的建立日期和上次更新日期。
- 別名政策條件使用模式比對；其不會繫結至別名的特定執行個體。使用別名型條件索引鍵的政策會影響所有符合模式的新別名和現有別名。如果您刪除並重新建立符合政策條件的別名，則條件會套用至新別名，就像舊別名一樣。

kms:RequestAlias 條件索引鍵依賴於操作請求中明確指定的別名。kms:ResourceAliases 條件索引鍵取決於與 KMS 金鑰相關聯的別名，即使其並沒有出現在請求中。

公里：RequestAlias

根據識別請求中 KMS 金鑰的別名，允許或拒絕存取 KMS 金鑰。您可以在金鑰政策或 IAM 政策中使用 kms: RequestAlias 條件金鑰。它適用於在請求中使用別名來識別 KMS 金鑰的作業，也就是密碼編譯作業和 GetPublicKey. DescribeKey 它對別名作業無效，例如CreateAlias或DeleteAlias。

在條件索引鍵中，指定別名名稱或別名名稱模式。您不能指定別名 ARN。

例如，下列金鑰政策陳述式可讓委託人對 KMS 金鑰使用指定的操作。僅在請求使用包含 alpha 來識別 KMS 金鑰的別名時，許可才有效。

```
{
  "Sid": "Key policy using a request alias condition",
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/alpha-developer"
  },
  "Action": [
    "kms:Decrypt",
    "kms:GenerateDataKey*",
    "kms:DescribeKey"
  ],
  "Resource": "*",
  "Condition": {
    "StringLike": {
```

```
      "kms:RequestAlias": "alias/*alpha*"
    }
  }
}
```

下列授權委託人提出的範例請求將滿足條件。不過，使用金鑰 ID 的請求、金鑰 ARN，或者不同的別名將無法滿足條件，即使這些值識別了相同的 KMS 索引鍵。

```
$ aws kms describe-key --key-id "arn:aws:kms:us-west-2:111122223333:alias/project-
alpha"
```

公里：ResourceAliases

根據與 KMS 金鑰相關聯的別名允許或拒絕存取 KMS 金鑰，即使請求中未使用別名。kms:ResourceAliases 條件金鑰可讓您指定別名或別名模式，例如`alias/test*`，您可以在 IAM 政策中使用它來控制對相同區域中數個 KMS 金鑰的存取。它適用於使用 KMS 金鑰的任何 AWS KMS 操作。

例如，下列 IAM 政策可讓委託人管理兩個 AWS 帳戶 中 KMS 金鑰的自動金鑰輪換。不過，此許可僅適用於與開頭為 `restricted` 之別名關聯的 KMS 金鑰。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AliasBasedIAMPolicy",
      "Effect": "Allow",
      "Action": [
        "kms:EnableKeyRotation",
        "kms:DisableKeyRotation",
        "kms:GetKeyRotationStatus"
      ],
      "Resource": [
        "arn:aws:kms:*:111122223333:key/*",
        "arn:aws:kms:*:444455556666:key/*"
      ],
      "Condition": {
        "ForAnyValue:StringLike": {
          "kms:ResourceAliases": "alias/restricted*"
        }
      }
    }
  ]
```

```
}
```

kms:ResourceAliases 條件是資源的條件，而不是請求。因此，未指定別名的請求仍然可以滿足條件。

下列範例請求 (指定相符別名) 會滿足條件。

```
$ aws kms enable-key-rotation --key-id "alias/restricted-project"
```

不過，下列範例請求也符合條件，前提是指定的 KMS 金鑰具有以 restricted 開頭的別名，即使別名未在請求中使用。

```
$ aws kms enable-key-rotation --key-id "1234abcd-12ab-34cd-56ef-1234567890ab"
```

尋找 AWS CloudTrail 日誌中的別名

您可以使用別名來表示 AWS KMS API 操作中的 AWS KMS key。當您執行時，KMS 金鑰的別名和金鑰 ARN 會記錄在事件的 AWS CloudTrail 日誌項目中。別名會顯示在 requestParameters 欄位。金鑰 ARN 會顯示在 resources 欄位。即使 AWS 服務使用您帳戶中的 AWS 受管金鑰，亦是如此。

例如，下列GenerateDataKey要求會使用project-key別名來代表 KMS 金鑰。

```
$ aws kms generate-data-key --key-id alias/project-key --key-spec AES_256
```

記錄 CloudTrail 檔中記錄此要求時，記錄項目會同時包含所使用的實際 KMS 金鑰的別名和金鑰 ARN。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "ABCDE",
        "arn": "arn:aws:iam::111122223333:role/ProjectDev",
        "accountId": "111122223333",
        "accessKeyId": "FFHIJ",
        "userName": "example-dev"
    },
    "eventTime": "2020-06-29T23:36:41Z",
    "eventSource": "kms.amazonaws.com",
```

```
    "eventName": "GenerateDataKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "205.205.123.000",
    "userAgent": "aws-cli/1.18.89 Python/3.6.10
 Linux/4.9.217-0.1.ac.205.84.332.metal1.x86_64 botocore/1.17.12",
    "requestParameters": {
        "keyId": "alias/project-key",
        "keySpec": "AES_256"
    },
    "responseElements": null,
    "requestID": "d93f57f5-d4c5-4bab-8139-5a1f7824a363",
    "eventID": "d63001e2-dbc6-4aae-90cb-e5370aca7125",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

如需記錄 CloudTrail 檔中記錄AWS KMS作業的詳細資訊，請參閱使用 AWS CloudTrail 記錄 AWS KMS API 呼叫。

檢視金鑰

您可以使用 AWS Management Console 或 AWS Key Management Service (AWS KMS) API 來檢視每個帳戶和區域中的 AWS KMS keys，包括您管理的 KMS 金鑰和由 AWS 管理的 KMS 金鑰。

主題

- 在主控台中檢視 KMS 金鑰
- 使用 API 檢視 KMS 金鑰
- 檢視 KMS 金鑰的密碼編譯組態
- 尋找金鑰 ID 和金鑰 ARN
- 尋找別名和別名 ARN

在主控台中檢視 KMS 金鑰

在 AWS Management Console 中，您可以檢視帳戶和區域中 KMS 金鑰的清單，以及每個 KMS 金鑰的詳細資訊。

> **Note**
>
> AWS KMS 主控台會顯示您在您的帳戶和區域中有權檢視的 KMS 金鑰。其他 AWS 帳戶 中的 KMS 金鑰不會出現在主控台中，即使您有權檢視、管理和使用它們。若要檢視其他帳戶中的 KMS 金鑰，請使用此DescribeKey作業。

主題

導覽至金鑰資料表

每個帳戶和每個區域中的 AWS KMS keys 會顯示在資料表中。您所建立的 KMS 金鑰和 AWS 服務為您建立的 KMS 金鑰會有不同的資料表。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 若要檢視您所建立及管理帳戶中的金鑰，請在導覽窗格中選擇Customer managed keys (客戶受管金鑰)。若要檢視 AWS 為您建立及管理之帳戶中的金鑰，請在導覽窗格中選擇 AWS 受管金鑰。如需不同類型 KMS 金鑰的詳細資訊，請參閱 AWS KMS keys。

> **Tip**
>
> 若要檢視遺失別名的 AWS 受管金鑰，請使用 Customer managed keys (客戶受管金鑰) 頁面。

AWS KMS 主控台也會顯示帳戶和區域中的自訂金鑰存放區。您在自訂金鑰存放區中建立的 KMS 金鑰會顯示在 Customer managed keys (客戶受管金鑰) 頁面上。如需自訂金鑰存放區的詳細資訊，請參閱 自訂金鑰存放區。

瀏覽至金鑰詳細資訊

有帳戶和區域中每個 AWS KMS key 的詳細資訊頁面。詳細資訊頁面會顯示 KMS 金鑰的 General configuration (一般組態) 區段，並包含可讓授權使用者檢視和管理金鑰的 Cryptographic configuration (密碼編譯組態) 和 Key policy (金鑰政策) 索引標籤。根據金鑰類型而定，詳細資訊頁面可能也包含 Aliases (別名)、Key material (金鑰材料)、Key rotation (金鑰輪換)、Public key (公有金鑰)、Regionality (區域性) 和 Tags (標籤) 索引標籤。

瀏覽 KMS 金鑰的金鑰詳細資訊頁面。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 若要檢視您所建立及管理帳戶中的金鑰，請在導覽窗格中選擇Customer managed keys (客戶受管金鑰)。若要檢視 AWS 為您建立及管理之帳戶中的金鑰，請在導覽窗格中選擇 AWS 受管金鑰。如需不同類型 KMS 金鑰的詳細資訊，請參閱 AWS KMS key。
4. 若要開啟金鑰詳細資訊頁面，請在金鑰資料表中選擇 KMS 金鑰的金鑰 ID 或別名。

   如果 KMS 金鑰具有多個別名，則別名摘要 (+n more (+n 等)) 會顯示在其中一個別名的名稱旁邊。選擇別名摘要會直接帶您前往金鑰詳細資訊頁面上的 Aliases (別名) 索引標籤。

排序和篩選 KMS 金鑰

若要更容易地在主控台中找到您的 KMS 金鑰，則您可以排序和篩選金鑰資料表。

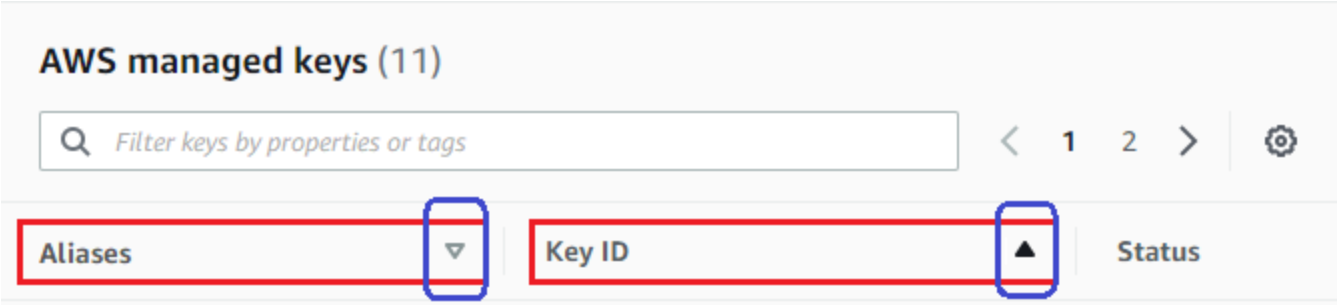
Sort

您可以根據資料欄的值，以遞增或遞減順序排序 KMS 金鑰。這項功能會排序資料表中的所有 KMS 金鑰，即使 KMS 金鑰沒有在目前的資料表頁面上顯示也一樣。

可排序的資料行在其資料行名稱的旁邊會有一個箭頭。在 AWS 受管金鑰 頁面上，您可以依 Aliases (別名) 或 Key ID (金鑰 ID) 排序。在 Customer managed keys (客戶受管金鑰) 頁面上，您可以根據 Aliases (別名)、Key ID (金鑰 ID) 或 Key type (金鑰類型) 排序。

如要依照遞增順序排序，請選擇資料行的標頭，直到箭頭指向上方。如要依照遞減順序排序，請選擇資料行的標頭，直到箭頭指向下方。您一次只能依據一欄來排序。

例如，您可以根據金鑰 ID，以遞增順序排序 KMS 金鑰，而非使用預設的別名。



當您在 Customer managed keys (客戶受管金鑰) 頁面上依 Key type (金鑰類型) 按遞增順序排列 KMS 金鑰時，所有非對稱金鑰會顯示在所有對稱金鑰之前。

篩選條件

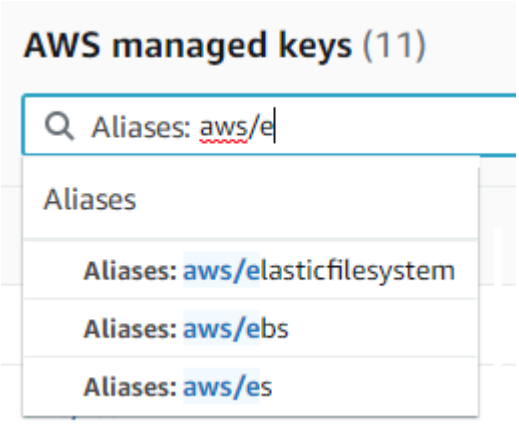
您可以依據 KMS 金鑰的屬性值或標籤來篩選 KMS 金鑰。篩選條件適用於所有資料表中的 KMS 金鑰，即使 CMK 沒有出現在目前的資料表頁面上也一樣。篩選條件不區分大小寫。

可篩選的屬性會列在篩選條件方塊中。在 AWS 受管金鑰 頁面上，您可以依別名和金鑰 ID 進行篩選。在 Customer managed keys (客戶受管金鑰) 頁面上，您可以根據別名、金鑰 ID 和金鑰類型以及根據標籤進行篩選。

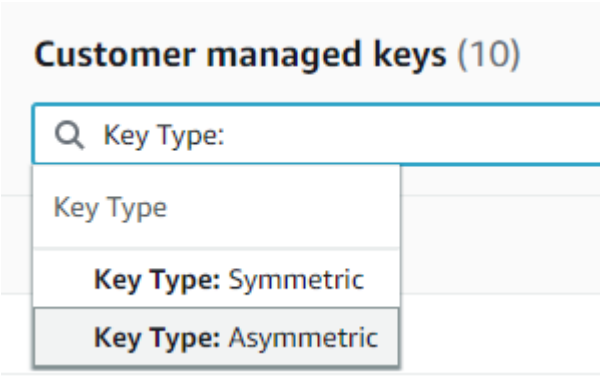
- 在 AWS 受管金鑰 頁面上，您可以依別名和金鑰 ID 進行篩選。
- 在 Customer managed keys (客戶受管金鑰) 頁面上，您可以根據標籤，或根據別名、金鑰 ID 和金鑰類型或區域性屬性進行篩選。

若要根據屬性值進行篩選，請選擇篩選條件、選擇屬性名稱，然後從實際屬性值的清單中選擇。若要依標籤進行篩選，請選擇標籤索引鍵，然後從實際標籤值清單中選擇。在選擇屬性或標籤索引鍵後，您也可以輸入全部或部分的屬性值或標籤值。在您做出選擇之前，您將會看到結果的預覽。

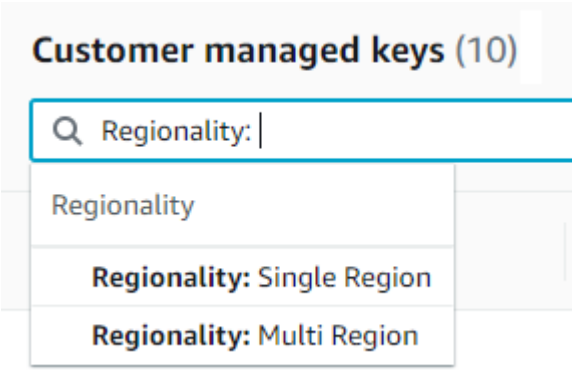
例如，如要顯示別名名稱包含 `aws/e` 的 KMS 金鑰，請選擇篩選條件方塊、選擇 Alias (別名)、輸入 `aws/e`，然後按 `Enter` 或 `Return` 來新增篩選條件。



若要在 Customer managed keys (客戶受管金鑰) 頁面上僅顯示非對稱 KMS 金鑰，請按一下篩選條件方塊，選擇 Key type (金鑰類型)，然後選擇 Key type: Asymmetric (金鑰類型：不對稱)。只有當資料表中有非對稱 KMS 金鑰時，才會顯示 Asymmetric (非對稱) 選項。如需識別非對稱 KMS 金鑰的詳細資訊，請參閱 識別非對稱 KMS 金鑰。



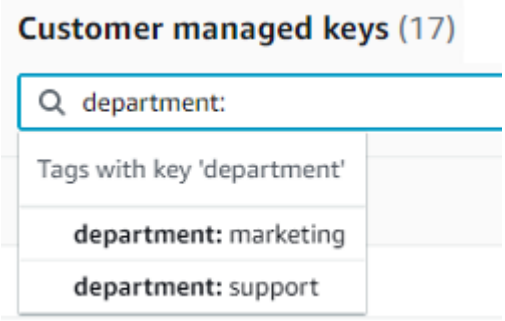
若只要顯示多區域金鑰，請在 Customer managed keys (客戶受管金鑰) 頁面上，選擇篩選條件方塊、選擇 Regionality (區域性)，然後選擇 Regionality: Multi-Region (區域性：多區域)。Multi-Region (多區域) 選項僅當資料表中有多區域金鑰時才會顯示。如需識別多區域金鑰的詳細資訊，請參閱 檢視多區域金鑰。



標籤篩選有一些不同。若只要顯示具有特定標籤的 KMS 金鑰，請選擇篩選方塊、選擇標籤索引鍵，然後從實際標籤值中選擇。您也可以輸入全部或部分的標籤值。

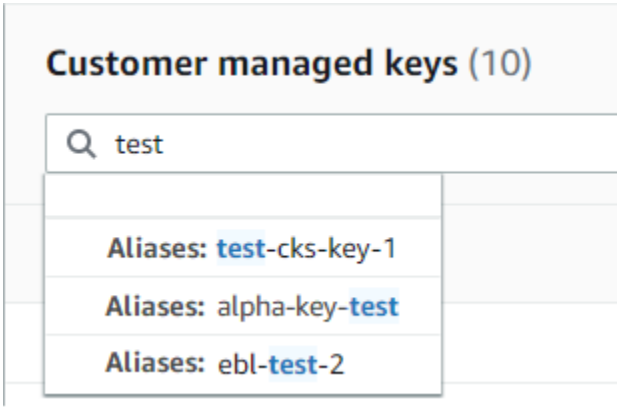
產生的資料表會顯示具有所選標籤的所有 KMS 金鑰。但不會顯示標籤。若要查看標籤，請選擇 KMS 金鑰的金鑰 ID 或別名，然後在其詳細資訊頁面上選擇 Tags (標籤) 索引標籤。索引標籤會顯示在 General Configuration (一般組態) 區段下。

此篩選條件需要標籤索引鍵和標籤值。只輸入標籤索引鍵或只輸入其值，就無法找到 KMS 金鑰。若要依標籤金鑰或值的全部或部分篩選標籤，請使用此ListResourceTags作業取得標記 KMS 金鑰，然後使用程式設計語言的篩選功能。如需範例，請參閱 ListResourceTags：取得 KMS 金鑰上的標籤。

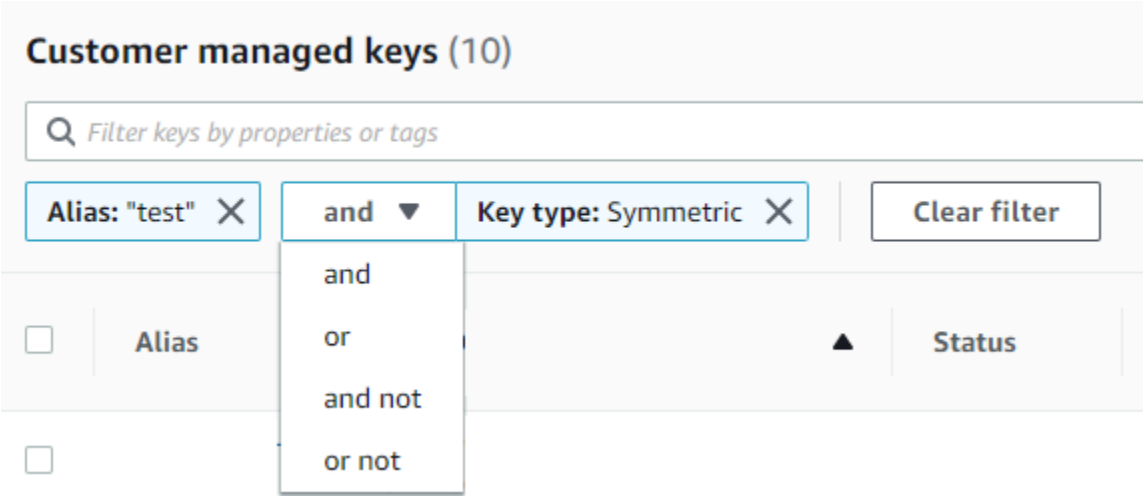


若要搜尋文字，請在篩選條件方塊中輸入全部或部分的別名、金鑰 ID、金鑰類型或標籤索引鍵。(選取標籤索引鍵後，您可以搜尋標籤值)。在您做出選擇之前，您將會看到結果的預覽。

例如，若要顯示在其標籤索引鍵中具有 `test` 的 KMS 金鑰或具有可篩選屬性的 KMS 金鑰，請輸入在篩選條件方塊中輸入 `test`。預覽會顯示篩選條件會選取的 KMS 金鑰。在此案例中，`test` 只會出現在 Alias (別名) 屬性中。



您可以同時使用多個篩選條件。在您新增其他篩選條件時，您也可以選取邏輯運算子。

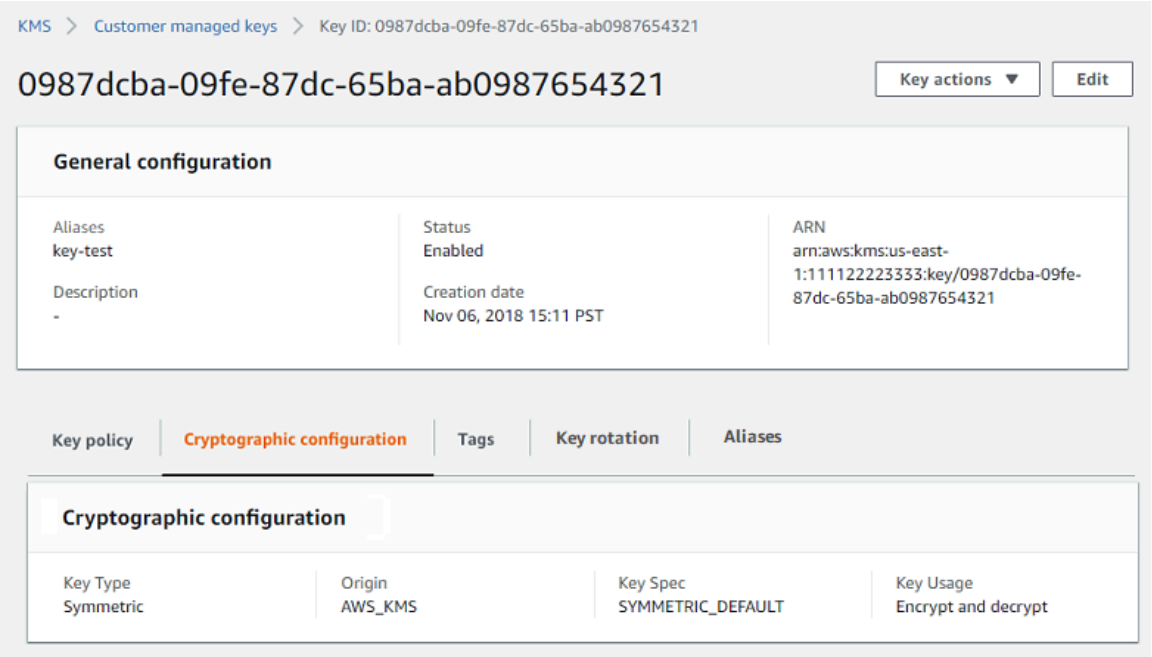


顯示 KMS 金鑰詳細資訊

每個 KMS 金鑰的詳細資訊頁面會顯示 KMS 金鑰的屬性。這會根據不同 KMS 金鑰類型而有些微不同。

若要顯示 KMS 金鑰的詳細資訊，請在 AWS 受管金鑰 或 Customer managed keys (客戶受管金鑰) 頁面上，選擇 KMS 金鑰的別名或金鑰 ID。

KMS 金鑰的詳細資訊頁面包含 General Configuration (一般組態) 區段，顯示 KMS 金鑰的基本屬性。其也包含可讓您檢視和編輯 KMS 金鑰屬性的索引標籤，例如 Key policy (金鑰政策)、Cryptographic configuration (密碼編譯組態)、Tags (標籤)、Key material (金鑰資料) (適用於含有匯入金鑰資料的 KMS 金鑰)、Key rotation (金鑰輪換) (適用於對稱加密 KMS 金鑰)、Regionality (區域性) (適用於多區域金鑰) 以及 Public key (公有金鑰) (適用於非對稱 KMS 金鑰)。



以下清單描述了詳細顯示中的欄位，包括索引標籤中的欄位。其中有些欄位也可以在資料表顯示中做為資料行使用。

Aliases

其中：別名索引標籤

KMS 金鑰的易記名稱。您可以使用別名來識別主控台和某些 AWS KMS API 中的 KMS 金鑰。如需詳細資訊，請參閱 使用別名。

Aliases (別名) 索引標籤會顯示與 AWS 帳戶 和區域中 KMS 金鑰相關聯的所有別名。

ARN

其中：一般組態區段

KMS 金鑰的 Amazon Resource Name (ARN)。該值會專屬識別 KMS 金鑰。您可以使用該值在 AWS KMS API 操作中識別 KMS 金鑰。

連線狀態

指示自訂金鑰存放區是否已連接至其備份金鑰存放區。當 KMS 金鑰已在自訂金鑰存放區中建立時，才會顯示此欄位。

如需有關此欄位中值的資訊，請參閱 AWS KMSAPI 參考ConnectionState中的。

建立日期

其中：一般組態區段

建立 KMS 金鑰的日期和時間。這個值會以裝置的本地時間顯示。時區不會依存區域。

與 Expiration (過期) 不同，建立指的只是 KMS 金鑰，而非其金鑰材料。

CloudHSM 叢集 ID

其中：密碼編譯組態索引標籤

AWS CloudHSM 叢集的叢集 ID，其中包含 KMS 金鑰的金鑰材料。當 KMS 金鑰已在自訂金鑰存放區中建立時，才會顯示此欄位。

如果您選擇 CloudHSM 叢集 ID，它會在 AWS CloudHSM 主控台開啟 Clusters (叢集) 頁面。

自訂金鑰存放區 ID

其中：密碼編譯組態索引標籤

包含 KMS 金鑰的自訂金鑰存放區 ID。當 KMS 金鑰已在自訂金鑰存放區中建立時，才會顯示此欄位。

如果您選擇自訂金鑰存放區 ID，它會在 AWS KMS 主控台開啟 Custom key stores (自訂金鑰存放區) 頁面。

自訂金鑰存放區名稱

其中：密碼編譯組態索引標籤

包含 KMS 金鑰的自訂金鑰存放區 名稱。當 KMS 金鑰已在自訂金鑰存放區中建立時，才會顯示此欄位。

自訂金鑰存放區類型

其中：密碼編譯組態索引標籤

指示自訂金鑰存放區是 AWS CloudHSM 金鑰存放區還是外部金鑰存放區。當 KMS 金鑰已在自訂金鑰存放區中建立時，才會顯示此欄位。

描述

其中：一般組態區段

KMS 金鑰的簡短、選用描述，您可以撰寫和編輯。如要新增或更新客戶受管金鑰的描述，請在 General Configuration (一般組態) 上方，選擇 Edit (編輯)。

加密演算法

其中：密碼編譯組態索引標籤

列出可搭配 AWS KMS 中 KMS 金鑰使用的加密演算法。此欄位只有在 Key type (金鑰類型) 是 Asymmetric (非對稱) 且 Key usage (金鑰使用方式) 是 Encrypt and decrypt (加密及解密) 時才會出現。如需 AWS KMS 支援之加密演算法的相關資訊，請參閱SYMMETRIC_DEFAULT 金鑰規格和用於加密和解密的 RSA 金鑰規格。

過期日期

其中：金鑰材料索引標籤

KMS 金鑰之金鑰材料過期的日期和時間。此欄位只會針對具備匯入金鑰材料的 KMS 金鑰顯示，即 Origin (來源) 是 External (外部) 且 KMS 金鑰具有會過期的金鑰材料時。

外部金鑰 ID

其中：密碼編譯組態索引標籤

與外部金鑰存放區中的 KMS 金鑰相關聯的外部金鑰 ID。此欄位只會針對外部金鑰存放區中的 KMS 金鑰顯示。

外部金鑰狀態

其中：密碼編譯組態索引標籤

外部金鑰存放區代理針對與 KMS 金鑰相關聯的外部金鑰所報告的最新狀態。此欄位只會針對外部金鑰存放區中的 KMS 金鑰顯示。

外部金鑰使用情況

其中：密碼編譯組態索引標籤

在與 KMS 金鑰相關聯的外部金鑰上啟用的密碼編譯操作。此欄位只會針對外部金鑰存放區中的 KMS 金鑰顯示。

金鑰政策

其中：金鑰政策索引標籤

與 IAM 政策和授予一同控制對 KMS 金鑰的存取。每個 KMS 金鑰都有一個金鑰政策。這是唯一的強制授權元素。如要變更客戶受管金鑰的金鑰政策，請在 Key policy (金鑰政策) 標籤上，選擇 Edit (編輯)。如需詳細資訊，請參閱 the section called “金鑰政策”。

金鑰輪換

其中：金鑰輪換索引標籤

啟用和停用客戶管理的 CMK 金鑰 中金鑰材料的自動輪換。如要變更客戶受管金鑰的金鑰輪換狀態，請使用 Key rotation (金鑰輪換) 標籤上的核取方塊。

您不能啟用或停用 AWS 受管金鑰 中金鑰資料的輪換。AWS 受管金鑰 每年會自動輪換一次。

金鑰規格

其中：密碼編譯組態索引標籤

KMS 金鑰中金鑰資料的類型。AWS KMS 支援對稱加密 KMS 金鑰 (SYMMETRIC_DEFAULT)、不同長度的 HMAC KMS 金鑰、不同長度 RSA 金鑰的 KMS 金鑰，以及具有不同曲線的橢圓曲線金鑰。如需詳細資訊，請參閱 金鑰規格。

Key type

其中：密碼編譯組態索引標籤

指出 KMS 金鑰是 Symmetric (對稱) 還是 Asymmetric (非對稱)。

金鑰用途

其中：密碼編譯組態索引標籤

指出 KMS 金鑰是用於 Encrypt and decrypt (加密及解密)、Sign and verify (簽署及驗證) 還是 Generate and verify MAC (產生及驗證 MAC)。如需詳細資訊，請參閱 金鑰用途。

Origin

其中：密碼編譯組態索引標籤

KMS 金鑰之金鑰材料的來源。有效的 值如下：

- AWS KMS 適用於 AWS KMS 產生的金鑰材料
- AWS CloudHSM 適用於 AWS CloudHSM 金鑰存放區中的 KMS 金鑰
- External (外部) 適用於匯入金鑰材料 (BYOK)
- External key store (外部金鑰存放區) 適用於外部金鑰存放區中的 KMS 金鑰

MAC 演算法

其中：密碼編譯組態索引標籤

列出可搭配 AWS KMS 中 HMAC KMS 金鑰使用的 MAC 演算法。僅在 Key spec (金鑰規格) 為 HMAC 金鑰規格 (HMAC_*) 時，才會顯示此欄位。如需有關 AWS KMS 支援的 MAC 演算法的詳細資訊，請參閱 HMAC KMS 金鑰的金鑰規格。

主索引鍵

其中：區域性索引標籤

表示此 KMS 金鑰是多區域主要金鑰。授權使用者可以使用此區段將主要金鑰變更為不同的相關多區域金鑰。僅在 KMS 金鑰為多區域主要金鑰時，才會顯示此欄位。

公有金鑰

其中：公有金鑰索引標籤

顯示非對稱 KMS 金鑰的公有金鑰。獲得授權的使用者可以使用此標籤來複製及下載公有金鑰。

區域性

其中：「一般組態」區段和「區域性」索引標籤

指出 KMS 金鑰是否為單一區域金鑰、多區域主要金鑰，或多區域複本金鑰。僅在 KMS 金鑰為多區域金鑰時，才會顯示此欄位。

相關的多區域金鑰

其中：區域性索引標籤

顯示所有相關多區域主要金鑰和複本金鑰，但目前的 KMS 金鑰除外。僅在 KMS 金鑰為多區域金鑰時，才會顯示此欄位。

在主要金鑰的相關多區域金鑰區段，授權使用者可以建立新的複本金鑰。

複本金鑰

其中：區域性索引標籤

表示此 KMS 金鑰是多區域複本金鑰。僅在 KMS 金鑰為多區域複本金鑰時，才會顯示此欄位。

簽署演算法

其中：密碼編譯組態索引標籤

列出可搭配 AWS KMS 中 KMS 金鑰使用的簽署演算法。此欄位只有在 Key type (金鑰類型) 是 Asymmetric (非對稱) 且 Key usage (金鑰使用方式) 是 Sign and verify (簽署及驗證) 時才會出現。如需 AWS KMS 支援之簽署演算法的相關資訊，請參閱用於簽署和驗證的 RSA 金鑰規格和橢圓曲線金鑰規格。

Status

其中：一般組態區段

KMS 金鑰的金鑰狀態。只有在狀態為 Enabled (啟用) 時，您才能在密碼編譯操作中使用 KMS 金鑰。如需每個 KMS 金鑰狀態的詳細說明及其對您可以在 KMS 金鑰上執行操作所帶來的影響，請參閱 AWS KMS 金鑰的金鑰狀態。

標籤

其中：標籤索引標籤

描述 KMS 金鑰的選用鍵值對。若要新增或變更 KMS 金鑰的標籤，請在 Tags (標籤) 索引標籤上，選擇 Edit (編輯)。

將標籤新增到 AWS 資源時，AWS 會產生成本配置報告，內含按標籤彙總的用量與成本。標籤也可以用來控制 KMS 金鑰的存取。如需標記 KMS 金鑰的詳細資訊，請參閱 標記金鑰 和 AWS KMS 的 ABAC。

自訂您的 KMS 金鑰資料表

您可以自訂呈現在 AWS 受管金鑰 和 AWS Management Console 中 Customer managed keys (客戶管理的金鑰) 頁面上的資料表，以符合您的需求。您可以選擇資料表資料欄、每個頁面的 AWS KMS keys 數量 (Page size (頁面大小))，以及文字換行。您選擇的組態會在您確認時儲存，並會在您開啟頁面時重新套用。

若要自訂您的 KMS 金鑰資料表

1. 在 AWS 受管金鑰 或 Customer managed keys (客戶受管金鑰) 頁面上，選擇位於頁面右上角的設定圖示 ()。
2. 在 Preferences (偏好設定) 頁面上，選擇您偏好的設定，然後選擇 Confirm (確認)。

請考慮使用 Page size (頁面大小) 設定來增加每個頁面上顯示的 KMS 金鑰數量，特別是如果您通常使用容易捲動的裝置的話。

您顯示的資料資料行可能會因資料表、您的任務角色和帳戶及區域中的 KMS 金鑰類型而有所不同。下表提供了一些建議的組態。如需資料行的說明，請參閱 顯示 KMS 金鑰詳細資訊。

建議的 KMS 金鑰資料表組態

您可以自訂 KMS 金鑰資料表中顯示的欄位，以顯示您需要的 KMS 金鑰相關資訊。

AWS 受管金鑰

根據預設，AWS 受管金鑰 資料表會顯示 Aliases (別名)、Key ID (金鑰 ID) 和 Status (狀態) 資料欄。這些資料行適合大多數的使用案例。

對稱加密 KMS 金鑰

如果只搭配 AWS KMS 產生的金鑰資料使用對稱加密 KMS 金鑰，Aliases (別名)、Key ID (金鑰 ID)、Status (狀態) 和 Creation date (建立日期) 資料欄可能最為有用。

非對稱 KMS 金鑰

如果使用非對稱 KMS 金鑰，則除了 Aliases (別名)、Key ID (金鑰 ID) 和 Status (狀態) 資料欄外，請考慮新增 Key type (金鑰類型)、Key spec (金鑰規格) 和 Key usage (金鑰使用情形) 資料欄。這些資料行會顯示 KMS 金鑰是對稱還是非對稱、金鑰材料的類型，以及 KMS 金鑰可用於加密還是簽署。

HMAC KMS 金鑰

如果使用 HMAC KMS 金鑰，則除了 Aliases (別名)、Key ID (金鑰 ID) 和 Status (狀態) 資料欄外，還可以考慮新增 Key spec (金鑰規格) 和 Key usage (金鑰用途) 資料欄。這些資料欄會顯示 KMS 金鑰是否為 HMAC 金鑰。由於您無法依據金鑰規格或金鑰用途來排序 KMS 金鑰，因此請使用別名和標籤來識別 HMAC 金鑰，然後使用 AWS KMS 主控台的篩選功能，依別名或標籤進行篩選。

匯入的金鑰資料

如果您的 KMS 金鑰具有匯入的金鑰材料，請考慮新增 Origin (來源)和 Expiration date (過期日期) 資料欄。這些資料行會顯示 KMS 金鑰中的金鑰材料是匯入的還是由 AWS KMS 產生的，以及金鑰過期的時間 (若有的話)。Creation date (建立日期) 欄位會顯示建立 KMS 金鑰的日期 (沒有金鑰材料時)。其不會反映任何金鑰材料的特性。

自訂金鑰存放區中的金鑰

如果您在自訂金鑰存放區中擁有 KMS 金鑰，則請考慮新增 Origin (來源) 和 Custom key store ID (自訂金鑰存放區 ID) 資料欄。這些資料欄會顯示 KMS 金鑰位於自訂金鑰存放區中、顯示自訂金鑰存放區類型，並識別自訂金鑰存放區。

多區域金鑰

如果您擁有多區域金鑰，請考慮新增 Regionality (區域性) 資料欄。這會顯示 KMS 金鑰是否為單一區域金鑰、多區域主要金鑰，或多區域複本金鑰。

使用 API 檢視 KMS 金鑰

您可以使用 AWS Key Management Service (AWS KMS) API 來檢視您的 KMS 金鑰。這個部分示範數個會傳回現有 KMS 金鑰之詳細資訊的操作。範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

主題

- ListKeys：取得所有 KMS 金鑰的識別碼和 ARN
- DescribeKey：取得 KMS 金鑰的詳細資訊
- GetKeyPolicy：取得附加至 KMS 金鑰的金鑰原則
- ListAliases：取得 KMS 金鑰的別名名稱和 ARN
- ListResourceTags：取得 KMS 金鑰上的標籤

ListKeys：取得所有 KMS 金鑰的識別碼和 ARN

此ListKeys作業會傳回帳戶和區域中所有 KMS 金鑰的 ID 和 Amazon 資源名稱 (ARN)。

例如，呼叫 `ListKeys` 操作會傳回這個虛構帳戶中每個 KMS 金鑰的 ID 和 ARN。如需多種程式設計語言的範例，請參閱取得 KMS 金鑰的金鑰 ID 和金鑰 ARN。

```
$ aws kms list-keys

{
  "Keys": [
    {
      "KeyArn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
      "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    {
      "KeyArn": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321",
      "KeyId": "0987dcba-09fe-87dc-65ba-ab0987654321"
    },
    {
      "KeyArn": "arn:aws:kms:us-
east-2:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d",
      "KeyId": "1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
    }
```

```
}
```

DescribeKey：取得 KMS 金鑰的詳細資訊

DescribeKey作業會傳回有關指定 KMS 金鑰的詳細資料。若要識別 KMS 金鑰，請使用金鑰 ID、金鑰 ARN、別名名稱或別名 ARN。

與僅在呼叫者帳戶和區域中顯示 KMS 金鑰的ListKeys作業不同，授權使用者可以使用此DescribeKey作業取得有關其他帳戶中 KMS 金鑰的詳細資料。

> **Note**
>
> DescribeKey 回應包括具有相同值的 KeySpec 和 CustomerMasterKeySpec 成員。已取代 CustomerMasterKeySpec 成員。

例如，呼叫 DescribeKey 會傳回對稱加密 KMS 金鑰的相關資訊。回應中的欄位會隨 AWS KMS key 規格、金鑰狀態和金鑰材料來源而有所不同。如需多種程式設計語言的範例，請參閱檢視 AWS KMS key。

```
$ aws kms describe-key --key-id 1234abcd-12ab-34cd-56ef-1234567890ab

{
    "KeyMetadata": {
        "Origin": "AWS_KMS",
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "Description": "",
        "KeyManager": "CUSTOMER",
        "Enabled": true,
        "KeySpec": "SYMMETRIC_DEFAULT",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "KeyState": "Enabled",
        "CreationDate": 1499988169.234,
        "MultiRegion": false,
        "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "AWSAccountId": "111122223333",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ]
```

```
    }
}
```

這個範例會在用於簽署和驗證的非對稱 KMS 金鑰上呼叫 `DescribeKey` 操作。回應包含 AWS KMS 針對此 KMS 金鑰支援的簽署演算法。

```
$ aws kms describe-key --key-id 0987dcba-09fe-87dc-65ba-ab0987654321

{
    "KeyMetadata": {
        "KeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
        "Origin": "AWS_KMS",
        "Arn": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321",
        "KeyState": "Enabled",
        "KeyUsage": "SIGN_VERIFY",
        "CreationDate": 1569973196.214,
        "Description": "",
        "KeySpec": "ECC_NIST_P521",
        "CustomerMasterKeySpec": "ECC_NIST_P521",
        "AWSAccountId": "111122223333",
        "Enabled": true,
        "MultiRegion": false,
        "KeyManager": "CUSTOMER",
        "SigningAlgorithms": [
            "ECDSA_SHA_512"
        ]
    }
}
```

GetKeyPolicy：取得附加至 KMS 金鑰的金鑰原則

作GetKeyPolicy業會取得附加至 KMS 金鑰的金鑰原則。若要識別 KMS 金鑰，請使用它的金鑰 ID 或金鑰 ARN。您還必須指定政策名稱，這一律是 `default`。(如果您的輸出難以讀取，請將 `--output text` 選項新增至您的命令。) `GetKeyPolicy` 僅適用於呼叫者帳戶和區域中的 KMS 金鑰。

如需多種程式設計語言的範例，請參閱取得金鑰政策。

```
$ aws kms get-key-policy --key-id 1234abcd-12ab-34cd-56ef-1234567890ab --policy-name
 default

{
```

```
  "Version" : "2012-10-17",
  "Id" : "key-default-1",
  "Statement" : [ {
    "Sid" : "Enable IAM User Permissions",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "arn:aws:iam::111122223333:root"
    },
    "Action" : "kms:*",
    "Resource" : "*"
  } ]
}
```

ListAliases：取得 KMS 金鑰的別名名稱和 ARN

作ListAliases業會傳回帳戶與區域中的別名。回應中的 TargetKeyId 會顯示別名 (如果有) 參考之 KMS 金鑰的金鑰 ID。

在預設情況下，ListAliases 命令會傳回在帳戶和區域中的所有別名。這包括您建立的別名，而且是與客戶管理的金鑰相關聯的別名，以及 AWS 建立且與您帳戶中 AWS 受管金鑰 相關聯的別名。您可以識別 AWS 別名，因為其名稱具有格式 aws/*<service-name>*，例如 aws/dynamodb。

回應可能還包括沒有 TargetKeyId 欄位的別名，例如此範例中的 aws/redshift 別名。這些是 AWS 已建立但尚未關聯至 KMS 金鑰的預先定義別名。

如需多種程式設計語言的範例，請參閱列出別名。

```
$ aws kms list-aliases

{
    "Aliases": [
        {
            "AliasName": "alias/access-key",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/access-key",
            "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
            "CreationDate": 1516435200.399,
            "LastUpdatedDate": 1516435200.399
        },
        {
            "AliasName": "alias/financeKey",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/financeKey",
            "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
```

```
            "CreationDate": 1604958290.014,
            "LastUpdatedDate": 1604958290.014
        },
        {
            "AliasName": "alias/ECC-P521-Sign",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/ECC-P521-Sign",
            "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "CreationDate": 1693622000.704,
            "LastUpdatedDate": 1693622000.704
        },
        {
            "AliasName": "alias/ImportedKey",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/ImportedKey",
            "TargetKeyId": "1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d",
            "CreationDate": 1493622000.704,
            "LastUpdatedDate": 1521097200.235
        },
        {
            "AliasName": "alias/aws/dynamodb",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/aws/dynamodb",
            "TargetKeyId": "0987ab65-43cd-21ef-09ab-87654321cdef",
            "CreationDate": 1521097200.454,
            "LastUpdatedDate": 1521097200.454
        },
        {
            "AliasName": "alias/aws/ebs",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/aws/ebs",
            "TargetKeyId": "abcd1234-09fe-ef90-09fe-ab0987654321",
            "CreationDate": 1466518990.200,
            "LastUpdatedDate": 1466518990.200
        },
        {
            "AliasName": "alias/aws/redshift",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/aws/redshift"
        },
    ]
}
```


若要取得參考特定 KMS 金鑰的別名，請使用 KeyId 參數。參數值可以是金鑰 ID 或金鑰 ARN。您不能指定別名名稱或別名 ARN。

以下範例中的命令會取得參考客戶受管金鑰的別名。但是，您也可以使用這類命令來尋找參考 AWS 受管金鑰 的別名。

```
$ aws kms list-aliases --key-id arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321
{
    "Aliases": [
        {
            "AliasName": "alias/access-key",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/access-key",
            "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
            "CreationDate": 1516435200.399,
            "LastUpdatedDate": 1516435200.399
        },
        {
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/financeKey",
            "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
            "AliasName": "alias/financeKey",
            "CreationDate": 1604958290.014,
            "LastUpdatedDate": 1604958290.014
        },
    ]
}
```

若只要取得 AWS 受管金鑰 的別名，請使用程式設計語言的功能來篩選回應。

```
$ aws kms list-aliases --query 'Aliases[?starts_with(AliasName, `alias/aws/`)]'
```

ListResourceTags：取得 KMS 金鑰上的標籤

此作ListResourceTags業會傳回指定 KMS 金鑰上的標籤。API 會傳回一個 KMS 金鑰的標籤，但您可以在迴圈中執行命令，以取得帳戶和區域中所有 KMS 金鑰的標籤，或您選取的一組 KMS 金鑰。此 API 一次傳回一個頁面，因此如果您在許多 KMS 金鑰上有許多標籤，您可能必須使用程式設計語言中的分頁器來取得所有您想要的標籤。

`ListResourceTags` 操作會傳回所有 KMS 金鑰的標籤，但是 AWS 受管金鑰 未加上標籤。它僅適用於呼叫者帳戶和區域中的 KMS 金鑰。

若要尋找 KMS 金鑰的標籤，請使用 `ListResourceTags` 操作。`KeyId` 參數是必要參數。它接受金鑰 ID 或金鑰 ARN。執行此範例之前，請使用有效的 ARN 取代範例金鑰 ARN。

```
$  aws kms list-resource-tags --key-id arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
      {
```

```
    "Tags": [
        {
            "TagKey": "Department",
            "TagValue": "IT"
        },
        {
            "TagKey": "Purpose",
            "TagValue": "Test"
        }
    ],
    "Truncated": false
}
```

您可能想要使用 ListResourceTags 操作，以取得帳戶和區域中具有特定標籤、標籤索引鍵或標籤值的所有 KMS 金鑰。若要執行這項操作，請使用程式設計語言的篩選功能。

例如，下列 Bash 指令碼會使用ListKeys和ListResourceTags作業，以Project標籤金鑰取得帳戶和區域中的所有 KMS 金鑰。這兩個操作都只取得結果的第一頁。如果您有許多 KMS 金鑰或許多標籤，請使用您語言的分頁功能，從每個操作中取得全部結果。執行此範例之前，請使用有效的 ID 取代範例金鑰 ID。

```
TARGET_TAG_KEY='Project'

for key in $(aws kms list-keys --query 'Keys[*].KeyId' --output text); do
  key_tags=$(aws kms list-resource-tags --key-id "$key" --query "Tags[?TagKey==\`
$TARGET_TAG_KEY\`]")
  if [ "$key_tags" != "[]" ]; then
    echo "Key: $key"
    echo "$key_tags"
  fi
done
```

系統會對輸出進行格式化，如下列範例輸出。

```
Key: 0987dcba-09fe-87dc-65ba-ab0987654321
[
    {
        "TagKey": "Project",
        "TagValue": "Gamma"
    }
]
Key: 1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d
```

```
[
    {
        "TagKey": "Project",
        "TagValue": "Alpha"
    }
]
Key: 0987ab65-43cd-21ef-09ab-87654321cdef
[
    {
        "TagKey": "Project",
        "TagValue": "Alpha"
    }
]
```

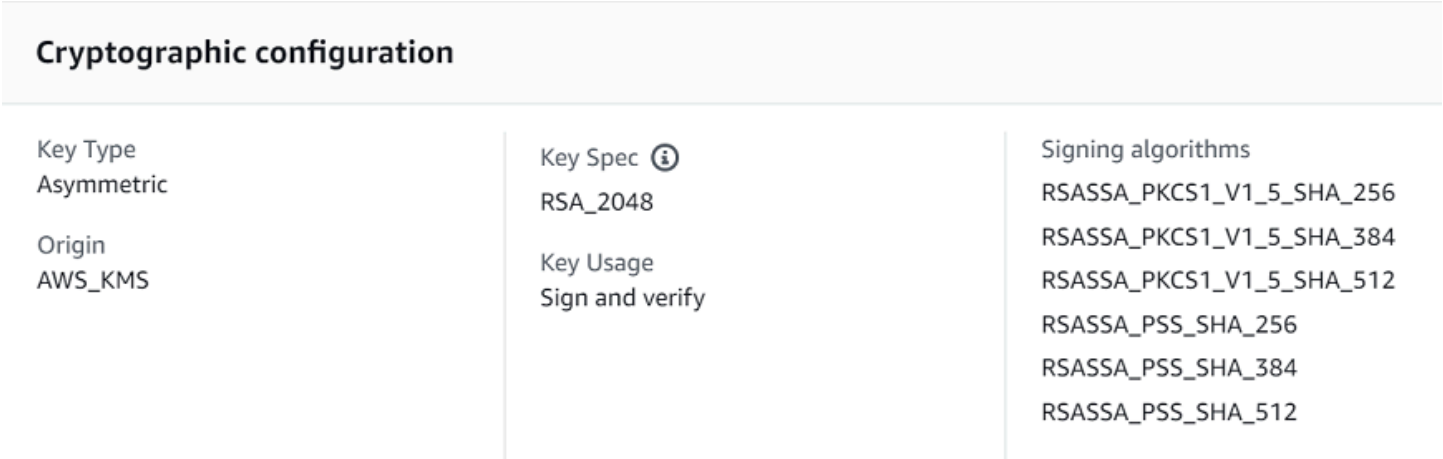
檢視 KMS 金鑰的密碼編譯組態

建立 KMS 金鑰之後，即可檢視其密碼編譯組態。KMS 金鑰建立之後即無法變更組態。如果您偏好不同的組態，請刪除 KMS 金鑰後再重新建立。

您可以在 AWS KMS 主控台中或使用 AWS KMS API 找到 KMS 金鑰的密碼編譯組態，包括金鑰規格、金鑰用途，以及支援的加密或簽署演算法。如需詳細資訊，請參閱 識別非對稱 KMS 金鑰。

在 AWS KMS 主控台中，每個 KMS 金鑰的詳細資訊頁面都有 Cryptographic configuration (密碼編譯組態) 索引標籤，顯示您 KMS 金鑰的密碼編譯詳細資訊。例如，下圖顯示用於簽署和驗證之 RSA KMS 金鑰的 Cryptographic configuration (密碼編譯組態) 索引標籤。

某些特殊用途 KMS 金鑰的 Cryptographic configuration (密碼編譯組態) 索引標籤還有其他專用區段。例如，自訂金鑰存放區中 KMS 金鑰的 Cryptographic configuration (密碼編譯組態) 索引標籤具有 Custom key stores (自訂金鑰存放區) 區段。外部金鑰存放區中 KMS 金鑰的 Cryptographic configuration (密碼編譯組態) 索引標籤具有 External key (外部金鑰) 區段。



在 AWS KMS API 中，使用該DescribeKey操作。回應中的 KeyMetadata 結構包括 KMS 金鑰的密碼編譯組態。例如，DescribeKey 會傳回以下用於簽署和驗證的 RSA KMS 金鑰回應。

```
{
  "KeyMetadata": {

    "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333",
    "CreationDate": 1571767572.317,
    "CustomerMasterKeySpec": "RSA_2048",
    "Description": "",
    "Enabled": true,
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeyState": "Enabled",
    "MultiRegion": false,
    "Origin": "AWS_KMS",
    "KeySpec": "RSA_2048",
    "KeyUsage": "SIGN_VERIFY",
    "SigningAlgorithms": [
        "RSASSA_PKCS1_V1_5_SHA_256",
        "RSASSA_PKCS1_V1_5_SHA_384",
        "RSASSA_PKCS1_V1_5_SHA_512",
        "RSASSA_PSS_SHA_256",
        "RSASSA_PSS_SHA_384",
        "RSASSA_PSS_SHA_512"
    ]
  }
}
```

尋找金鑰 ID 和金鑰 ARN

若要識別 AWS KMS key，您可以使用金鑰 ID 或 Amazon Resource Name (金鑰 ARN)。在密碼編譯操作中，您也可以使用別名名稱或別名 ARN。

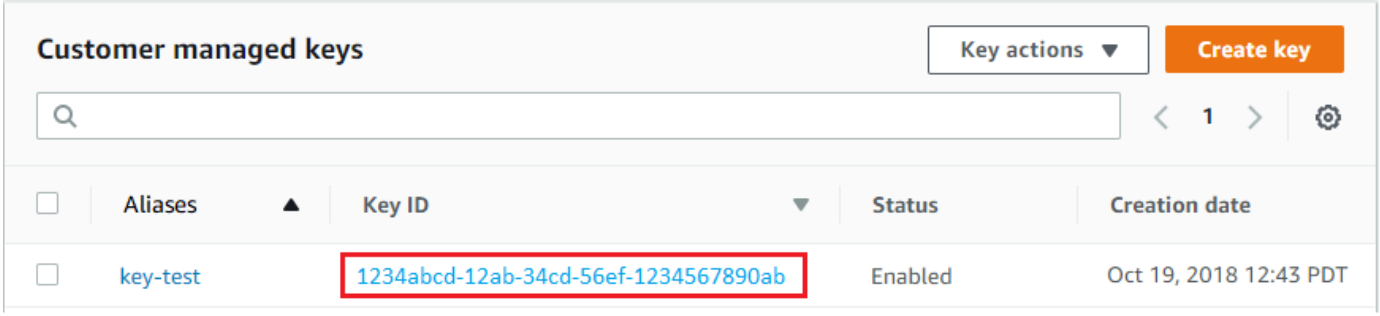
如需 AWS KMS 支援之 KMS 金鑰識別符的詳細資訊，請參閱 金鑰識別碼 (KeyId)。如需尋找別名名稱和別名 ARN 的說明，請參閱 尋找別名和別名 ARN。

若要尋找金鑰 ID 和 ARN (主控台)

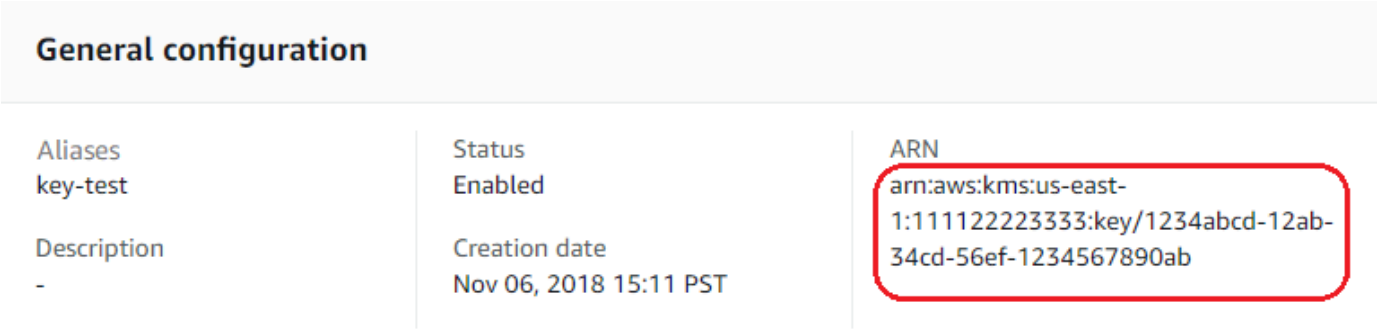
1. 開啟位於 https://console.aws.amazon.com/kms 的 AWS KMS 主控台。

2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 若要檢視您所建立及管理帳戶中的金鑰，請在導覽窗格中選擇Customer managed keys (客戶受管金鑰)。若要檢視 AWS 為您建立及管理之帳戶中的金鑰，請在導覽窗格中選擇 AWS 受管金鑰。
4. 若要尋找 KMS 金鑰的金鑰 ID，請查看以 KMS 金鑰別名開頭的資料列。

   根據預設，Key ID (金鑰 ID) 資料行會出現在資料表中。如果 Key ID (金鑰 ID) 資料行並未出現在您的資料表內，請使用 the section called “自訂您的 KMS 金鑰資料表” 中說明的程序來還原。您也可以在 KMS 金鑰的詳細資訊頁面上檢視 KMS 金鑰的金鑰 ID。



5. 若要尋找 KMS 金鑰的 Amazon Resource Name (ARN)，請選擇金鑰 ID 或別名。General Configuration (一般組態) 區段中會顯示的金鑰 ARN。



尋找金鑰 ID 和金鑰 ARN (AWS KMS API)

若要尋找的金鑰 ID 和金鑰 ARNAWS KMS key，請使用ListKeys作業。如需多種程式設計語言的範例，請參閱取得金鑰 ID 和 ARN和取得金鑰 ID 和 ARN。

`ListKeys` 回應包括帳戶和區域中每個 KMS 金鑰的金鑰 ID 和金鑰 ARN。

```
$ aws kms list-keys
{
    "Keys": [
        {
```

```
            "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "KeyArn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        },
        {
            "KeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
            "KeyArn": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321"
        }
    ]
}
```

尋找別名和別名 ARN

別名是 AWS KMS AWS KMS keys (KMS 金鑰) 的易記名稱。您可以在 AWS KMS 主控台或 AWS KMS API 中找到別名名稱和別名 ARN。

如需 AWS KMS 支援之 KMS 金鑰識別符的詳細資訊，請參閱 金鑰識別碼 (KeyId)。如需尋找金鑰 ID 和金鑰 ARN 的說明，請參閱 尋找金鑰 ID 和金鑰 ARN。

主題

- 尋找別名名稱和別名 ARN (主控台)
- 尋找別名名稱和別名 ARN (AWS KMS API)

尋找別名名稱和別名 ARN (主控台)

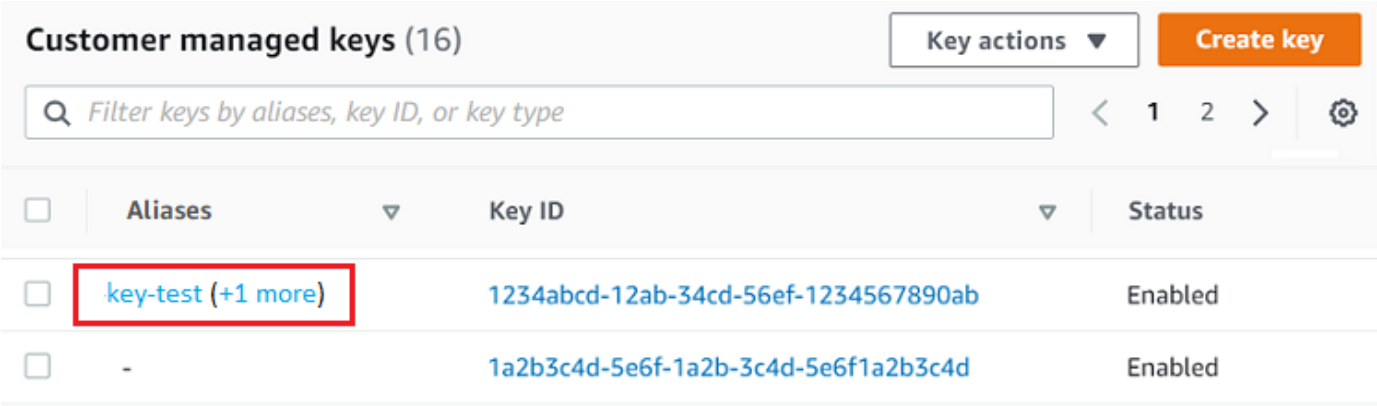
AWS KMS 主控台會顯示與 KMS 金鑰相關聯的別名。

1. 開啟位於 https://console.aws.amazon.com/kms 的 AWS KMS 主控台。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 若要檢視您所建立及管理帳戶中的金鑰，請在導覽窗格中選擇Customer managed keys (客戶受管金鑰)。若要檢視 AWS 為您建立及管理之帳戶中的金鑰，請在導覽窗格中選擇 AWS 受管金鑰。
4. Aliases (別名) 欄會顯示每個 KMS 金鑰的別名。如果 KMS 金鑰沒有別名，則會在 Aliases (別名) 資料欄中使用破折號 (-) 顯示。

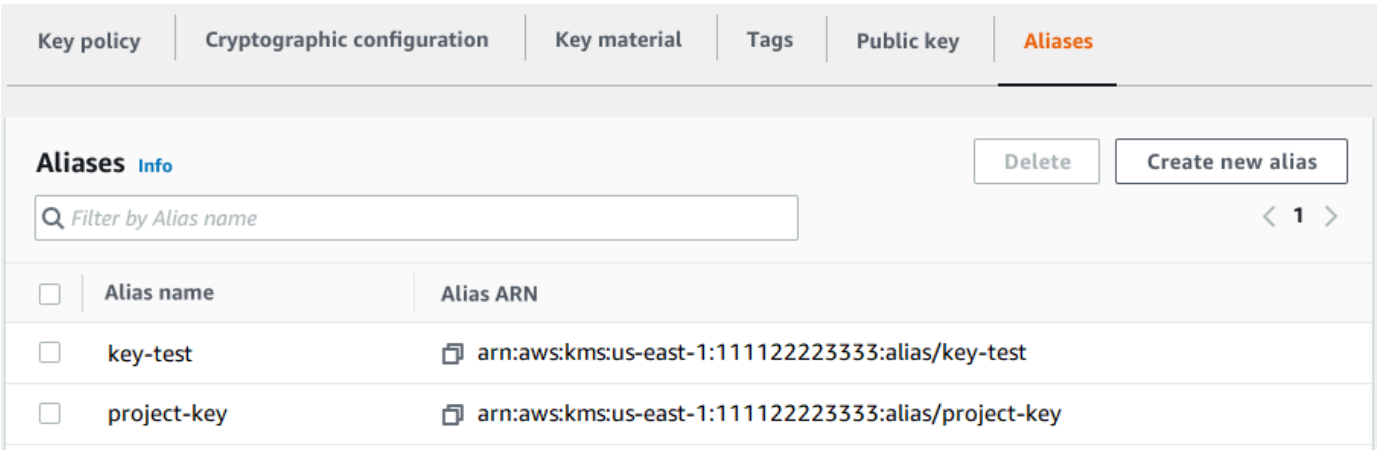
   如果 KMS 金鑰有多個別名，Aliases (別名) 資料欄也有別名摘要，例如 (+n more) (+n 等)。例如，下列 KMS 金鑰有兩個別名，其中一個是 `key-test`。

   若要尋找 KMS 金鑰所有別名的別名名稱和別名 ARN，請使用 Aliases (別名) 索引標籤。

- 若要直接移至 Aliases (別名) 欄中的 Aliases (別名) 索引標籤，請選擇別名摘要 (+n 等)。只有在 KMS 金鑰具有多個別名時，才會顯示別名摘要。
- 或者，選擇 KMS 金鑰的別名或金鑰 ID (這會開啟 KMS 金鑰的詳細資訊頁面)，然後選擇 Aliases (別名) 索引標籤。索引標籤位於 General configuration (一般組態) 區段。



5. Aliases (別名) 索引標籤會顯示 KMS 金鑰所有別名的別名名稱和別名 ARN。您也可以在此索引標籤上建立並刪除 KMS 金鑰的別名。



尋找別名名稱和別名 ARN (AWS KMS API)

若要尋找的別名和別名 ARNAWS KMS key，請使用此ListAliases作業。如需多種程式設計語言的範例，請參閱列出別名和取得別名和 ARN。

依預設，回應會包含帳戶和區域中每個別名的別名和別名 ARN。若只要取得特定 KMS 金鑰的別名，請使用 `KeyId` 參數。

例如，下列命令只會取得具有金鑰 ID 1234abcd-12ab-34cd-56ef-1234567890ab 的範例 KMS 金鑰的別名。

```
$ aws kms list-aliases --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
{
    "Aliases": [
        {
            "AliasName": "alias/key-test",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/key-test",
            "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "CreationDate": 1593622000.191,
            "LastUpdatedDate": 1593622000.191
        },
        {
            "AliasName": "alias/project-key",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/project-key",
            "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
            "CreationDate": 1516435200.399,
            "LastUpdatedDate": 1516435200.399
        }
    ]
}
```

編輯金鑰

您可以在 AWS KMS 主控台中和藉由使用 AWS KMS API 來變更您客戶受管金鑰的以下屬性。

您無法編輯 AWS 受管金鑰 或 AWS 擁有的金鑰 的任何屬性。這些金鑰是由建立金鑰的 AWS 服務來管理。

描述

您可以在 KMS 金鑰的詳細資料頁面上變更客戶受管金鑰的說明，或使用UpdateKeyDescription作業。

若要在主控台中編輯金鑰說明，請在 KMS 金鑰的詳細資訊頁面右上角，選擇 Edit (編輯)。

金鑰政策

您可以在客戶受管金鑰的詳細資料頁面的 [金鑰原則] 索引標籤上變更金鑰原則，或使用PutKeyPolicy作業變更金鑰原則。

如需詳細資訊，請參閱 變更金鑰政策。

標籤

您可以在 AWS KMS 主控台的客戶受管金鑰頁面上，或客戶受管金鑰詳細資訊頁面的Tags標籤 (標籤) 索引標籤中，建立和刪除標籤。或者，您可以使用TagResource和UntagResource操作。

如需詳細資訊，請參閱 標記金鑰。

啟用和停用

您可以在 AWS KMS 主控台的客戶受管金鑰頁面上，或客戶受管金鑰的詳細資訊頁面上，啟用和停用 KMS 金鑰。或者，您可以使用EnableKey和DisableKey操作。

如需詳細資訊，請參閱 啟用和停用金鑰。

自動金鑰輪換

您可以在客戶管理金鑰的詳細資料頁面的金鑰輪替標籤上啟用和停用自動金鑰輪換，或使用EnableKeyRotation和DisableKeyRotation作業。

如需詳細資訊，請參閱 輪換 AWS KMS keys。

另請參閱

更新別名

標記金鑰

在 AWS KMS 中，您可以在建立 KMS 金鑰時將標籤新增到客戶受管金鑰，並標記或取消標記現有的 KMS 金鑰，除非其處於待刪除狀態。您無法標記別名、自訂金鑰存放區、AWS 受管金鑰、AWS 擁有的金鑰，或其他 AWS 帳戶 中的 KMS 金鑰。標籤是選用的，但它們可以非常有用。

如需更多詳細資訊，請參閱 建立金鑰 和 編輯金鑰。如需標籤的一般資訊，包括最佳實務、標記策略以及標籤的格式與語法，請參閱《Amazon Web Services 一般參考》的標記 AWS 資源。

主題

- 關於 AWS KMS 中的標籤
- 在主控台中管理 KMS 金鑰標籤
- 使用 API 操作管理 KMS 金鑰標籤
- 控制對標籤的存取
- 使用標籤來控制對 KMS 金鑰的存取

關於 AWS KMS 中的標籤

標籤是您可以 (或 AWS 可以) 指派給 AWS 資源的選用中繼資料標籤。每個標籤皆包含標籤索引鍵和標籤值，它們都是區分大小寫的字串。此標籤值可以是空 (null) 字串。資源上的每個標籤都必須有不同的標籤索引鍵，但您可以將相同的標籤新增至多個 AWS 資源。每個資源最多可以有 50 個使用者建立的標籤。

請勿在標籤金鑰或標籤值包含機密或敏感資訊。許多 AWS 服務 都可以存取標籤，包括帳單。

在 AWS KMS 中，您可以在建立 KMS 金鑰時將標籤新增到客戶受管金鑰，並標記或取消標記現有的 KMS 金鑰，除非其處於待刪除狀態。您無法標記別名、自訂金鑰存放區、AWS 受管金鑰、AWS 擁有的金鑰，或其他 AWS 帳戶 中的 KMS 金鑰。標籤是選用的，但它們可以非常有用。

例如，您可以將 `"Project"="Alpha"` 標籤新增至您用於 Alpha 專案的所有 KMS 金鑰和 Amazon S3 儲存貯體。

```
TagKey   = "Project"
TagValue = "Alpha"
```

如需標籤的相關資訊 (包括格式與語法)，請參閱《Amazon Web Services 一般參考》的標記 AWS 資源。

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 服務支援標記，因此您可以對來自不同服務的資源指派相同的標籤，指出資源是相關的。例如，您可以將相同標籤指派給 KMS 金鑰和 Amazon Elastic Block Store (Amazon EBS) 磁碟區或 AWS Secrets Manager 機密。您也可以使用標籤來識別 KMS 金鑰以進行自動化。
- 追蹤您的 AWS 成本。將標籤新增到 AWS 資源時，AWS 會產生成本配置報告，內含按標籤彙總的用量與成本。您可以使用此功能來追蹤專案、應用程式或成本中心的 AWS KMS 成本。

  如需有關使用成本配置標籤的詳細資訊，請參閱《AWS Billing 使用者指南》中的使用成本分配標籤。如需標籤鍵和標籤值規則的相關資訊，請參閱《AWS Billing 使用者指南》中的使用者定義的標籤限制。
- 控制對 AWS 資源的存取。根據 KMS 金鑰的標籤允許和拒絕存取，屬於 AWS KMS 對屬性型存取控制 (ABAC) 的支援。如需有關根據標籤控制 AWS KMS keys 存取的更多資訊，請參閱 使用標籤來控制對 KMS 金鑰的存取。如需使用標籤以控制對 AWS 資源之存取的詳細資訊，請參閱《IAM 使用者指南》中的使用資源標籤控制對 AWS 資源的存取。

AWS KMS當您使用、或ListResourceTags作業時 TagResource，UntagResource會將項目寫入AWS CloudTrail記錄中。

在主控台中管理 KMS 金鑰標籤

當在 AWS KMS 主控台上建立 KMS 金鑰時，您可以將標籤新增至 KMS 金鑰。您也可以使用主控台的標籤索引標籤，以新增、編輯和刪除客戶受管金鑰上的標籤。若要新增、編輯、檢視和刪除 KMS 金鑰的標籤，您必須擁有必要的許可。如需詳細資訊，請參閱 控制對標籤的存取。

建立 KMS 金鑰時新增標籤

若要在主控台中建立 KMS 金鑰時新增標籤，您必須擁有 IAM 政策的 `kms:TagResource` 許可，以及在主控台中建立 KMS 金鑰和檢視 KMS 金鑰所需的許可。此許可至少必須涵蓋帳戶和區域中的所有 KMS 金鑰。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。(您無法管理 AWS 受管金鑰 的標籤)
4. 選擇金鑰類型，然後選擇 Next (下一頁)。
5. 輸入別名和選用描述。
6. 輸入標籤索引鍵和選用的標籤值。若要新增其他標籤，請選擇 Add tag (新增標籤)。若要移除標籤，請選擇 Remove (移除)。標記完新的 KMS 金鑰後，選擇 Next (下一頁)。
7. 完成建立 KMS 金鑰。

檢視和管理現有 KMS 金鑰上的標籤

若要在主控台中新增、檢視、編輯和刪除標籤，您需要 KMS 金鑰的標記許可。您可以從 KMS 金鑰的金鑰政策取得此許可，或者如果金鑰政策允許，則從包含 KMS 金鑰的 IAM 政策取得此許可。除了在主控台中檢視 KMS 金鑰的許可之外，您還需要這些許可。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。(您無法管理 AWS 受管金鑰 的標籤)

4. 您可以使用資料表篩選器，只顯示具有特定標籤的 KMS 金鑰。如需詳細資訊，請參閱 排序和篩選 KMS 金鑰。
5. 選取 KMS 金鑰別名旁的核取方塊。
6. 選擇 Key actions (金鑰動作)、Add or edit tags (新增或編輯標籤)。
7. 在 KMS 金鑰的詳細資訊頁面上，選擇 Tags (標籤) 索引標籤。

   - 若要建立您的第一個標籤，請選擇 Cretate tag (建立標籤)，輸入標籤索引鍵 (必要) 和標籤值 (選用)，然後選擇 Save (儲存)。

     如果您將標籤值保留空白，則實際標籤值為 null 或空字串。
   - 若要新增標籤，請選擇 Edit (編輯)，選擇 Add tag (新增標籤)，輸入標籤索引鍵和標籤值，然後選擇 Save (儲存)。
   - 若要變更標籤的名稱或值，請選擇 Edit (編輯)，完成您要的變更，然後選擇 Save (儲存)。
   - 若要刪除標籤，請選擇 Edit (編輯)。在標籤列，選擇 Remove (移除)，然後選擇 Save (儲存)。
8. 若要儲存您所做的變更，請選擇 Save changes (儲存變更)。

使用 API 操作管理 KMS 金鑰標籤

您可以使用 AWS Key Management Service (AWS KMS) API 來新增、刪除和列出您管理之 KMS 金鑰的標籤。以下範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。您無法標記 AWS 受管金鑰。

若要新增、編輯、檢視和刪除 KMS 金鑰的標籤，您必須擁有必要的許可。如需詳細資訊，請參閱 控制對標籤的存取。

主題

- CreateKey：將標籤新增至新的 KMS 金鑰
- TagResource：新增或變更 KMS 金鑰的標籤
- ListResourceTags：取得 KMS 金鑰的標籤
- UntagResource：從 KMS 金鑰刪除標籤

CreateKey：將標籤新增至新的 KMS 金鑰

您可以在建立客戶管理金鑰時新增標籤若要指定標籤，請使用CreateKey作業的Tags參數。

若要在建立 KMS 金鑰時新增標籤，呼叫者必須擁有 IAM 政策的 kms:TagResource 許可。此許可至少必須涵蓋帳戶和區域中的所有 KMS 金鑰。如需詳細資訊，請參閱 控制對標籤的存取。

CreateKey 的 Tags 參數值是區分大小寫的標籤鍵和標籤值對的集合。KMS 金鑰上的每個標籤必須有不同的標籤名稱。標籤值可以為 null 或空字串。

例如，下列 AWS CLI 命令會建立具有 Project:Alpha 標籤的對稱加密 KMS 金鑰。指定多個索引鍵/值組時，請使用空格來分隔每一組。

```
$ aws kms create-key --tags TagKey=Project,TagValue=Alpha
```

此命令成功時，會傳回包含新 KMS 金鑰相關資訊的 KeyMetadata 物件。但是，KeyMetadata 不包含標籤。若要取得標籤，請使用此ListResourceTags作業。

TagResource：新增或變更 KMS 金鑰的標籤

此TagResource作業會將一或多個標記新增至 KMS 金鑰。您無法使用此操作新增或編輯不同 AWS 帳戶 中的標籤。

若要新增標籤，請指定新標籤索引鍵和標籤值。若要編輯標籤，請指定現有標籤索引鍵和新標籤值。KMS 金鑰上的每個標籤必須有不同的標籤索引鍵。標籤值可以為 null 或空字串。

例如，下列命令會將 **Purpose** 和 **Department** 標籤新增至範例 KMS 金鑰。

```
$ aws kms tag-resource \
        --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
        --tags TagKey=Purpose,TagValue=Pretest TagKey=Department,TagValue=Finance
```

當此命令成功時，不會傳回任何輸出。若要檢視 KMS 金鑰上的標籤，請使用此ListResourceTags作業。

您也可以使用 TagResource 來變更現有標籤的標籤值。若要取代標籤值，請使用不同的值來指定相同的標籤索引鍵。

例如，這個命令會將 Purpose 標籤的值從 Pretest 變更為 Test。

```
$ aws kms tag-resource \
        --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
        --tags TagKey=Purpose,TagValue=Test
```

ListResourceTags：取得 KMS 金鑰的標籤

作ListResourceTags業會取得 KMS 金鑰的標籤。KeyId 參數是必要參數。您無法使用此操作檢視不同 AWS 帳戶 中 KMS 金鑰的標籤。

例如，下列命令會取得範例 KMS 金鑰的標籤。

```
$ aws kms list-resource-tags --key-id 1234abcd-12ab-34cd-56ef-1234567890ab

  "Truncated": false,
  "Tags": [
      {
        "TagKey": "Project",
        "TagValue": "Alpha"
     },
     {
       "TagKey": "Purpose",
       "TagValue": "Test"
     },
     {
       "TagKey": "Department",
       "TagValue": "Finance"
     }
  ]
}
```

UntagResource：從 KMS 金鑰刪除標籤

此作UntagResource業會從 KMS 金鑰刪除標籤。若要識別要刪除的標籤，請指定標籤索引鍵。您無法使用此操作刪除不同 AWS 帳戶 中 KMS 金鑰的標籤。

成功時，UntagResource 操作不會傳回任何輸出。此外，如果在 KMS 金鑰上找不到指定的標籤金鑰，則其不會擲回例外狀況或傳回回應。若要確認作業是否有效，請使用該ListResourceTags作業。

例如，此命令會從指定的 KMS 金鑰刪除 **Purpose** 標籤及其值。

```
$ aws kms untag-resource --key-id 1234abcd-12ab-34cd-56ef-1234567890ab --tag-keys
 Purpose
```

控制對標籤的存取

若要透過 AWS KMS 主控台或使用 API 新增、檢視和刪除標籤，則主體需要標記許可。您可以在金鑰政策中提供這些許可。您也可以在 IAM 政策 (包括 VPC 端點政策) 中提供，但僅當金鑰政策允許時。AWSKeyManagementServicePowerUser受管理的政策允許主體在帳戶可存取的所有 KMS 金鑰上標記、取消標記和列出標記。

您也可以使用標籤的 AWS 全域條件索引鍵來限制許可。在中AWS KMS，這些條件可以控制對標籤作業的存取，例如TagResource和UntagResource。

> **Note**
>
> 授予主體管理標籤和別名的許可時，請務必謹慎。變更標記或別名可允許或拒絕客戶受管金鑰的許可。如需詳細資訊，請參閱 AWS KMS 的 ABAC 和 使用標籤來控制對 KMS 金鑰的存取。

如需政策和詳細資訊，請參閱《IAM 使用者指南》中的根據標籤索引鍵控制存取。

建立和管理標籤的許可如下所示。

公里：TagResource

允許主體新增或編輯標籤。若要在建立 KMS 金鑰時新增標籤，主體必須在 IAM 政策中具有不限於特定 KMS 金鑰的許可。

公里：ListResourceTags

允許主體檢視 KMS 金鑰上的標籤。

公里：UntagResource

允許主體從 KMS 金鑰刪除標籤。

標記政策中的許可

您可以在金鑰政策或 IAM 政策中提供標記許可。例如，以下範例金鑰政策會給予精選使用者標記 KMS 金鑰的許可。它為所有可以擔任範例管理員或開發人員角色的使用者提供檢視標籤的許可。

```
{
  "Version": "2012-10-17",
  "Id": "example-key-policy",
```

```
  "Statement": [
    {
      "Sid": "Enable IAM User Permissions",
      "Effect": "Allow",
      "Principal": {"AWS": "arn:aws:iam::111122223333:root"},
      "Action": "kms:*",
      "Resource": "*"
    },
    {
      "Sid": "Allow all tagging permissions",
      "Effect": "Allow",
      "Principal": {"AWS": [
        "arn:aws:iam::111122223333:user/LeadAdmin",
        "arn:aws:iam::111122223333:user/SupportLead"
      ]},
      "Action": [
          "kms:TagResource",
          "kms:ListResourceTags",
          "kms:UntagResource"
      ],
      "Resource": "*"
    },
    {
      "Sid": "Allow roles to view tags",
      "Effect": "Allow",
      "Principal": {"AWS": [
        "arn:aws:iam::111122223333:role/Administrator",
        "arn:aws:iam::111122223333:role/Developer"
      ]},
      "Action": "kms:ListResourceTags",
      "Resource": "*"
    }
  ]
}
```

若要授予主體在多個 KMS 金鑰上的標記許可，您可以使用 IAM 政策。若要讓此政策生效，每個 KMS 金鑰的金鑰政策必須允許帳戶使用 IAM 政策來控制對 KMS 金鑰的存取。

例如，以下 IAM 政策允許主體建立 KMS 金鑰。它也允許其在指定帳戶中的所有 KMS 金鑰上建立和管理標籤。這種組合可讓主體使用CreateKey作業的標籤參數，在建立 KMS 金鑰時將標籤新增至 KMS 金鑰。

```
{
```

```
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "IAMPolicyCreateKeys",
      "Effect": "Allow",
      "Action": "kms:CreateKey",
      "Resource": "*"
    },
    {
      "Sid": "IAMPolicyTags",
      "Effect": "Allow",
      "Action": [
        "kms:TagResource",
        "kms:UntagResource",
        "kms:ListResourceTags"
      ],
      "Resource": "arn:aws:kms:*:111122223333:key/*"
    }
  ]
}
```

限制標籤許可

您可以使用政策條件來限制標記許可。下列政策條件可套用至 kms:TagResource 和 kms:UntagResource 許可。例如，您可以使用 aws:RequestTag/*tag-key* 條件，允許主體僅新增特定標籤，或防止主體新增具有特定標籤索引鍵的標籤。或者，您可以使用 kms:KeyOrigin 條件，以防止主體使用匯入金鑰材料標記或取消標記 KMS 金鑰。

- AWS：RequestTag
- aws：ResourceTag/標籤密鑰（僅適用於 IAM 政策）
- AWS：TagKeys
- 公里：CallerAccount
- 公里：KeySpec
- 公里：KeyUsage
- 公里：KeyOrigin
- 公里：ViaService

最佳實務的做法是，當您使用標籤來控制 KMS 金鑰的存取時，請使用 aws:RequestTag/*tag-key* 或 aws:TagKeys 條件鍵來確定允許哪些標籤 (或標籤索引鍵)。

例如，下列 IAM 政策與前一個類似。不過，此政策允許主體建立標籤 (TagResource) 並僅為具有 Project 標籤索引鍵的標籤刪除標籤 UntagResource。

由於TagResource和UntagResource請求可以包含多個標籤，因此您必須使用 aws: TagKeys 條件指定ForAllValues或 ForAnyValue set 運算子。ForAnyValue 運算子會要求請求中的至少一個標籤索引鍵與政策中的標籤索引鍵相符。ForAllValues 運算子會要求請求中的所有標籤索引鍵與政策中的其中一個標籤索引鍵相符。true如果請求中沒有標籤，ForAllValues操作員也會返回 TagResource ，但沒有指定標籤時 UntagResource失敗。如需集合運算子的詳細資訊，請參閱《IAM 使用者指南》中的使用多個索引鍵和值。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "IAMPolicyCreateKey",
      "Effect": "Allow",
      "Action": "kms:CreateKey",
      "Resource": "*"
    },
    {
      "Sid": "IAMPolicyViewAllTags",
      "Effect": "Allow",
      "Action": "kms:ListResourceTags",
      "Resource": "arn:aws:kms:*:111122223333:key/*"
    },
    {
      "Sid": "IAMPolicyManageTags",
      "Effect": "Allow",
      "Action": [
        "kms:TagResource",
        "kms:UntagResource"
      ],
      "Resource": "arn:aws:kms:*:111122223333:key/*",
      "Condition": {
          "ForAllValues:StringEquals": {"aws:TagKeys": "Project"}
      }
    }
  ]
}
```

使用標籤來控制對 KMS 金鑰的存取

您可以根據 KMS 金鑰上的標籤控制對 AWS KMS keys 的存取。例如，您可以撰寫 IAM 政策，允許主體僅啟用和停用具有特定標籤的 KMS 金鑰。或者，您可以使用 IAM 政策來防止主體在密碼編譯操作中使用 KMS 金鑰，除非 KMS 金鑰有特定的標籤。

這項功能是對屬性型存取控制 (ABAC) 的 AWS KMS 支援。如需使用標籤以控制對 AWS 資源之存取的資訊，請參閱《IAM 使用者指南》中的什麼是適用於 AWS 的 ABAC？和使用資源標籤控制對 AWS 資源的存取。如需解決與 ABAC 相關之存取問題的說明，請參閱 對適用於 AWS KMS 的 ABAC 進行故障診斷。

> **Note**
>
> 可能最多需要五分鐘才能將標籤和別名變更體現在 KMS 金鑰授權上。最近的變更可能會在 API 操作中可見，然後才會影響授權。

AWS KMS支援 aws:ResourceTag/tag-key 全域條件內容金鑰，可讓您根據 KMS 金鑰上的標籤控制對 KMS 金鑰的存取。由於多個 KMS 金鑰可以具有相同的標籤，所以此功能可讓您將許可套用至一組精選 KMS 金鑰。您也可以透過變更其標籤，輕鬆變更集合中的 KMS 金鑰。

在 AWS KMS 中，僅在 IAM 政策中支援 `aws:ResourceTag/`*`tag-key`* 條件索引鍵。金鑰原則不支援，這些原則僅適用於一個 KMS 金鑰，或不使用特定 KMS 金鑰的作業 (例如ListKeys或ListAliases作業)。

使用標籤控制存取可提供一種簡單、可擴展且靈活的方式來管理許可。不過，如果沒有正確設計和管理，它可能會意外允許或拒絕存取您的 KMS 金鑰。如果您使用標籤來控制存取，請考慮下列實務。

- 使用標籤來強化最低權限存取的最佳實務。僅為 IAM 主體提供他們必須使用或管理之 KMS 金鑰所需的許可。例如，使用標籤來標註專案所使用的 KMS 金鑰。然後授予專案小組僅將 KMS 金鑰與專案標籤搭配使用的許可。
- 要謹慎地授予主體 `kms:TagResource` 和 `kms:UntagResource` 許可，讓其新增、編輯和刪除別名。當您使用標籤來控制對 KMS 金鑰的存取時，變更標籤可授予主體使用 KMS 金鑰 (否則其沒有使用的許可) 的許可。它也可以拒絕存取其他主體執行任務所需的 KMS 金鑰。如果他們有管理標籤的許可，則沒有變更主要政策或建立授予之許可的金鑰管理員可以控制對 KMS 金鑰的存取。

  如果可能，請使用政策條件，例如 `aws:RequestTag/`*`tag-key`* 或 `aws:TagKeys`，將主體的標記許可限制為特定 KMS 金鑰上的特定標籤或標籤模式。

- 檢閱您 AWS 帳戶 中目前具有標記和取消標記許可的主體，並視需要進行調整。例如，主控台金鑰管理員的預設金鑰政策包含該 KMS 金鑰的 kms:TagResource 和 kms:UntagResource 許可。IAM 政策可能允許所有 KMS 金鑰的標記和取消標記許可。例如，AWSKeyManagementServicePowerUser受管理的政策允許主體在所有 KMS 金鑰上標記、取消標記和列出標記。
- 在設定依賴於標籤的政策之前，請先檢閱您 AWS 帳戶 中 KMS 金鑰上的標籤。請確定您的政策僅適用於您想要包含的標籤。使用CloudTrail 記錄和CloudWatch 警示來提醒您標記可能會影響 KMS 金鑰存取權的變更。
- 標籤型政策條件使用模式比對；其不會繫結至標籤的特定執行個體。使用標籤型條件索引鍵的政策會影響所有符合模式的新標籤和現有標籤。如果您刪除並重新建立符合政策條件的標籤，則條件會套用至新標籤，就像舊標籤一樣。

例如，請考慮以下 IAM 政策。它允許主體僅對您帳戶中屬於亞太區域 (新加坡) 區域GenerateDataKeyWithoutPlaintext且具有"Project"="Alpha"標籤的 KMS 金鑰呼叫和解密作業。您可以將此政策連接至 Alpha 專案範例中的角色。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "IAMPolicyWithResourceTag",
      "Effect": "Allow",
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt"
      ],
      "Resource": "arn:aws:kms:ap-southeast-1:111122223333:key/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Project": "Alpha"
        }
      }
    }
  ]
}
```

以下 IAM 政策範例允許主體使用帳戶中的任何 KMS 金鑰來進行密碼編譯操作。但其禁止主體針對具有 "Type"="Reserved" 標籤或不具有 "Type" 標籤的 KMS 金鑰使用這些密碼編譯操作。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "IAMAllowCryptographicOperations",
      "Effect": "Allow",
      "Action": [
        "kms:Encrypt",
        "kms:GenerateDataKey*",
        "kms:Decrypt",
        "kms:ReEncrypt*"
      ],
      "Resource": "arn:aws:kms:*:111122223333:key/*"
    },
    {
      "Sid": "IAMDenyOnTag",
      "Effect": "Deny",
      "Action": [
        "kms:Encrypt",
        "kms:GenerateDataKey*",
        "kms:Decrypt",
        "kms:ReEncrypt*"
      ],
      "Resource": "arn:aws:kms:*:111122223333:key/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Type": "Reserved"
        }
      }
    },
    {
      "Sid": "IAMDenyNoTag",
      "Effect": "Deny",
      "Action": [
        "kms:Encrypt",
        "kms:GenerateDataKey*",
        "kms:Decrypt",
        "kms:ReEncrypt*"
      ],
      "Resource": "arn:aws:kms:*:111122223333:key/*",
      "Condition": {
        "Null": {
          "aws:ResourceTag/Type": "true"
```

```
        }
      }
    }
  ]
}
```

啟用和停用金鑰

您可以停用和重新啟用客戶受管金鑰。當您建立 KMS 金鑰時，它預設為啟用。如果您停用 KMS 金鑰，則在您重新啟用前，其無法在任何密碼編譯操作中使用。

因其為暫時性並且容易撤消，因此停用 KMS 金鑰是刪除 KMS 金鑰的安全替代方法，刪除是一項具破壞性且不可逆轉的動作。如果您考慮刪除 KMS 金鑰，請先將其停用，然後設定CloudWatch 警示或類似機制，以確定您永遠不需要使用該金鑰來解密加密的資料。

當您停用 KMS 金鑰時，該 KMS 金鑰會立即變為無法使用 (視最終一致性而定)。不過，使用受 KMS 金鑰保護之資料金鑰所加密的資源不會受影響，直到再次使用 KMS 金鑰為止 (例如解密資料金鑰)。此問題會影響 AWS 服務，其中許多服務會使用資料金鑰來保護您的資源。如需詳細資訊，請參閱 無法使用的 KMS 金鑰如何影響資料金鑰。

您無法啟用或停用 AWS 受管金鑰 或 AWS 擁有的金鑰。AWS 受管金鑰 已永久啟用，以供使用 AWS KMS 的服務使用。AWS 擁有的金鑰 完全由擁有它們的服務進行管理。

> **Note**
>
> 停用時，AWS KMS 不會輪換客戶受管金鑰的金鑰材料。如需詳細資訊，請參閱 自動金鑰輪換的運作方式。

主題

- 啟用和停用 KMS 金鑰 (主控台)
- 啟用和停用 KMS 金鑰 (AWS KMS API)

啟用和停用 KMS 金鑰 (主控台)

您可以使用 AWS KMS 主控台以啟用和停用客戶受管金鑰。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。

2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 選擇您要啟用或停用的 KMS 金鑰的核取方塊。
5. 若要啟用 KMS 金鑰，請選擇 Key actions (金鑰動作)、Enable (啟用)。若要停用 KMS 金鑰，請選擇 Key actions (金鑰動作)、Disable (停用)。

啟用和停用 KMS 金鑰 (AWS KMS API)

該EnableKey操作啟用禁用AWS KMS key。以下範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。`key-id` 參數是必要參數。

此操作不會傳回輸出。若要查看金鑰狀態，請使用DescribeKey作業。

```
$ aws kms enable-key --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

此作DisableKey業會停用已啟用的 KMS 金鑰。`key-id` 參數是必要參數。

```
$ aws kms disable-key --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

此操作不會傳回輸出。若要查看金鑰狀態，請使用DescribeKey作業並查看`Enabled`欄位。

```
$ aws kms describe-key --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
{
    "KeyMetadata": {
        "Origin": "AWS_KMS",
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "Description": "",
        "KeyManager": "CUSTOMER",
        "MultiRegion": false,
        "Enabled": false,
        "KeyState": "Disabled",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "CreationDate": 1502910355.475,
        "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "AWSAccountId": "111122223333"
        "KeySpec": "SYMMETRIC_DEFAULT",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
```

```
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ]
    }
}
```

輪換 AWS KMS keys

若要為您的客戶受管金鑰建立新的加密材料，您可以建立新的 KMS 金鑰，然後變更您的應用程式或別名來使用新的 KMS 金鑰。或者，您可以啟用現有 KMS 金鑰的自動金鑰輪換。

當您為 KMS 金鑰啟用自動金鑰輪換時，AWS KMS 每年都會為 KMS 金鑰產生新的密碼編譯資料。AWS KMS 會永久儲存所有之前版本的密碼編譯資料，以便您可以解密使用該 KMS 金鑰加密的任何資料。AWS KMS 在您刪除 KMS 金鑰之前，不會刪除任何輪換密鑰資料。您可以跟踪 Amazon CloudWatch 和 KMS 密鑰的密鑰材料的輪換AWS CloudTrail。

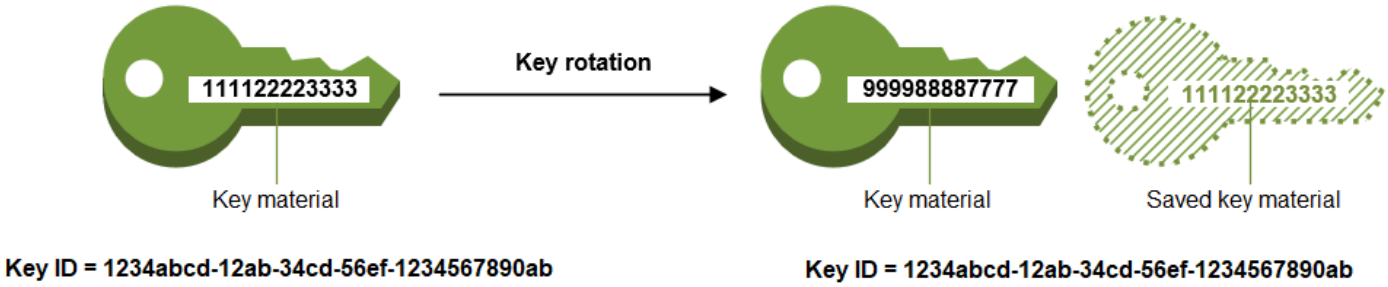
使用輪換的 KMS 金鑰來加密資料時，AWS KMS 會使用目前的金鑰資料。當您使用輪換的 KMS 金鑰來對加密文字進行解密時，AWS KMS 會使用原先用來加密的金鑰資料版本。您不能請求特定版本的金鑰資料。AWS KMS 會透明地使用適當的金鑰資料進行解密，所以您無須變更程式碼，就可以安全地在應用程式和 AWS 服務 中使用輪換的 KMS 金鑰。

不過，自動金鑰輪換對 KMS 金鑰保護的資料沒有影響。它不會輪換 KMS 金鑰產生的資料金鑰，或重新加密 KMS 金鑰保護的任何資料，也不會減輕受損資料金鑰的有效性。

AWS KMS 僅支援對稱加密 KMS 金鑰 (其含有 AWS KMS 建立的金鑰資料) 的自動金鑰輪換。對於客戶受管 KMS 金鑰而言，自動輪換為選用功能。AWS KMS 會一直每年輪換一次 AWS 受管 KMS 金鑰的金鑰資料。輪換 AWS 擁有的 KMS 金鑰有所差異。

> **Note**
>
> AWS 受管金鑰 的輪換間隔於 2022 年 5 月發生變更。如需詳細資訊，請參閱 AWS 受管金鑰。

金鑰輪換只會變更金鑰資料 (即加密操作中使用的密碼編譯私密)。無論其金鑰材料變更多少次，KMS 金鑰都是相同的邏輯資源。KMS 金鑰的屬性不會變更，如下圖所示。



自動金鑰輪換有下列好處：

- KMS 金鑰的屬性，包括其金鑰 ID、金鑰 ARN、區域、政策和許可，在金鑰輪換時不會變更。
- 您不需要變更參考 KMS 金鑰之金鑰 ID 或金鑰 ARN 的應用程式或別名。
- 輪換金鑰資料不會影響 KMS 金鑰在任何 AWS 服務 之中的使用。
- 當您啟用金鑰輪換，AWS KMS 會每年自動輪換 KMS 金鑰。您不必記住或排程更新。

您可以決定建立新的 KMS 金鑰，並使用它來代替原始 KMS 金鑰。這與輪換現有 KMS 金鑰的金鑰材料有相同效果，所以通常被視為手動輪換金鑰。當您想要控制金鑰輪換排程時，手動輪換是一個不錯的選擇。這一方法也可以用於輪換不符合自動金鑰輪換資格的 KMS 金鑰，包括非對稱 KMS 金鑰、HMAC KMS 金鑰、位於自訂金鑰存放區中的 KMS 金鑰，以及含有匯入金鑰材料的 KMS 金鑰。

金鑰輪換和定價

AWS KMS 會對 KMS 金鑰中維護的每個版本金鑰資料收取每月費用。如需詳細資訊，請參閱 AWS Key Management Service 定價。

> **Note**
>
> 您可利用 AWS Cost Explorer Service 來檢視金鑰儲存費用明細。例如，您可指定 `$REGION-KMS-Keys` 作為用量類型，然後按 API 操作將資料分組，以便篩選檢視並查看作為目前及輪換 KMS 金鑰計費的金鑰總費用。
> 您可能仍會看到歷史日期的舊版 Unknown API 操作執行個體。

金鑰輪換和配額

在計算金鑰資源配額時，每個 KMS 金鑰都被視為一個金鑰，而不考慮輪換的金鑰資料版本編號。

如需金鑰資料和輪換的詳細資訊，請參閱 AWS Key Management Service 密碼編譯詳細資訊。

主題

為什麼要輪換 KMS 金鑰？

密碼編譯最佳實務不鼓勵廣泛重複使用直接加密資料的金鑰，例如 AWS KMS 產生的資料金鑰。當 256 位元資料金鑰加密數百萬則訊息時，這些金鑰可能會用盡，並開始產生帶有詭祕模式的密文，狡猾分子可利用這些模式來發現金鑰中的位元。為避免此金鑰用盡，最好僅採用資料金鑰一次，或僅採用數次，這樣可有效輪換金鑰資料。

然而，KMS 金鑰最常用作包裝金鑰，也稱為金鑰加密的金鑰。包裝金鑰不會加密資料，其所加密的是用於加密資料的資料金鑰。因此，其使用頻率遠低於資料金鑰，且重複使用頻率之低，鮮少具金鑰額度用盡風險。

儘管額度用盡風險非常低，但由於業務或合約規則或政府法規，您仍可能需要輪換 KMS 金鑰。當您必須輪換 KMS 金鑰時，建議您採用自動金鑰輪換 (若支援)，並在不支援自動金鑰輪換時採用手動金鑰輪換。

自動金鑰輪換的運作方式

AWS KMS 的金鑰輪換設計不僅透明且易於使用。AWS KMS 僅支援客戶自管金鑰的選用自動金鑰輪換。

管理金鑰資料

AWS KMS 會保留 KMS 金鑰的所有金鑰資料，即使金鑰輪換已停用。AWS KMS 僅當您刪除 KMS 金鑰時才會刪除金鑰資料。

使用金鑰資料

使用輪換的 KMS 金鑰來加密資料時，AWS KMS 會使用目前的金鑰資料。當您使用輪換的 KMS 金鑰來對加密文字進行解密時，AWS KMS 會使用原先用來加密的相同版本的金鑰資料。您不能請求特定版本的金鑰資料。

輪換日期

AWS KMS在啟用輪換後一年 (約 365 天) 輪換金鑰資料，之後每年 (約 365 天) 輪換一次。

客戶受管金鑰

由於自動金鑰輪換在客戶自管金鑰是選擇性，且可隨時啟用及停用，因此輪換日期取決於最近啟用輪換的日期。在金鑰的使用壽命期間該日期可改變多次。

例如，如您於 2022 年 1 月 1 日建立客戶自管金鑰，並於 2022 年 3 月 15 日啟用自動輪換金鑰，則 AWS KMS 會在 2023 年 3 月 15 日、2024 年 3 月 15 日及之後每 365 天輪換金鑰資料。

下列為特殊案例：

- 停用金鑰輪換 — 如您在任何時候停用自動金鑰輪換，KMS 金鑰會繼續使用停用輪換時所用的金鑰資料版本。如您再次啟用自動金鑰輪換，AWS KMS 會在啟用輪換的新日期後一年以及之後每年 (約 365 天) 輪換金鑰資料。
- 停用 KMS 金鑰 — 當 KMS 金鑰停用時，AWS KMS 不會輪換金鑰。不過，金鑰輪換狀態不會變更，而且當 KMS 金鑰停用時您無法變更它。重新啟用 KMS 金鑰後，如果金鑰資料已存在超過一年，則 AWS KMS 會立即輪換且之後每年輪換。如果金鑰資料存在不超過一年，AWS KMS 會繼續原本的金鑰輪換排程。
- KMS 金鑰待刪除 — 當 KMS 金鑰待刪除時，AWS KMS 不會輪換金鑰。金鑰輪換狀態會設定為 `false`，且等待刪除時您無法變更它。如果刪除已取消，會恢復先前的金鑰輪換狀態。如金鑰資料已存在超過一年，AWS KMS 會立即輪換且之後每年 (自上次輪換起約 365 天) 輪換。如果金鑰資料存在不超過一年，AWS KMS 會繼續原本的金鑰輪換排程。

AWS 受管金鑰

AWS KMS 每年 (大約 365 天) 會自動輪換 AWS 受管金鑰。您無法啟用或停用 AWS 受管金鑰的金鑰輪換。

AWS 受管金鑰 的金鑰資料在建立日期後一年首次輪換，之後每年 (自上次輪換起約 365 天) 進行輪換。

> **Note**
>
> 在 2022 年 5 月，AWS KMS 將 AWS 受管金鑰 的輪換頻率從每三年 (大約 1,095 天) 變更為每年 (大約 365 天)。
> 新的 AWS 受管金鑰 會在建立一年後自動輪換，此後大約每年輪換一次。
> 現有的 AWS 受管金鑰 會在最近一次輪換一年後自動輪換，此後每年自動輪換一次。

AWS 擁有的金鑰

您無法啟用或停用 AWS 擁有的金鑰 的金鑰輪換。AWS 擁有的金鑰 的金鑰輪換策略會取決於建立和管理金鑰的 AWS 服務。如需詳細資訊，請參閱該服務使用者指南或開發人員指南中的靜態加密主題。

支援的 KMS 金鑰類型

僅對稱加密 KMS 金鑰 (其含有 AWS KMS 產生的金鑰資料 (來源 = AWS_KMS)) 支援自動金鑰輪換。

在下列 KMS 金鑰類型上不支援自動金鑰輪換，但是您可以手動輪換這些 KMS 金鑰。

- 非對稱 KMS 金鑰
- HMAC KMS 金鑰
- 自訂金鑰存放區中的 KMS 金鑰
- 包含匯入金鑰資料的 KMS 金鑰

多區域金鑰

您可以啟用和停用多區域金鑰的自動金鑰輪換。您只能在主要金鑰上設定屬性。AWS KMS 同步金鑰時，它會將主要金鑰的屬性設定複製到其複本金鑰。輪換主要金鑰的金鑰材料時，AWS KMS 會自動將該金鑰材料複製到其所有複本金鑰中。如需詳細資訊，請參閱 輪換多區域金鑰。

AWS 服務

您可以啟用您在 AWS 服務中用於伺服器端加密之客戶受管金鑰的自動金鑰輪換。年度輪換是透明的，而且與 AWS 服務相容。

監控金鑰輪換

當AWS KMS自動輪換AWS 受管金鑰或客戶受管金鑰的金鑰材料時，會將KMS CMK Rotation事件寫入 Amazon， EventBridge並將RotateKey 事件寫入AWS CloudTrail日誌。您可以使用這些記錄來驗證 KMS 金鑰是否已輪換。

最終一致性

自動金鑰輪換會受限於與其他 AWS KMS 管理操作相同的最終一致性效果。在新的金鑰材料可用於整個 AWS KMS 之前，可能會稍微延遲。然而，輪換金鑰材料不會造成密碼編譯操作中的任何中斷或延遲。在新的金鑰材料可用於整個 AWS KMS 之前，目前的金鑰材料會用於密碼編譯操作。當多區域金鑰的金鑰材料自動輪換時，在新的金鑰材料可用於所有具有相關多區域金鑰的區域之前，AWS KMS 會使用目前的金鑰材料。

如何啟用和停用自動金鑰輪換

授權使用者可以使用 AWS KMS 主控台或 AWS KMS API 來啟用和停用自動金鑰輪換並檢視金鑰的輪換狀態。

啟用自動金鑰輪換後，AWS KMS 會在啟用日期一年後輪換 KMS 金鑰，之後每年輪換 KMS 金鑰一次。

主題

啟用和停用金鑰輪換 (主控台)

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。(您不能啟用或停用 AWS 受管金鑰 的輪換。它們會每年自動輪換一次。)
4. 選擇 KMS 金鑰的別名或金鑰 ID。
5. 選擇 Key rotation (金鑰輪換) 標籤。

   僅對稱加密 KMS 金鑰 (其含有 AWS KMS 產生的金鑰資料 (Origin (來源) 是 AWS_KMS)) (包括多區域對稱加密 KMS 金鑰) 的詳細資訊頁面會顯示 Key rotation (金鑰輪換) 索引標籤。

   您無法自動輪換非對稱 KMS 金鑰、HMAC KMS 金鑰、含有匯入金鑰資料的 KMS 金鑰，或是位於自訂金鑰存放區中的 KMS 金鑰。但是，您可以手動進行輪換。
6. 選取或清除 Automatically rotate this KMS key every year (每年自動輪換此 KMS 金鑰) 核取方塊。

> **Note**
>
> 如果 KMS 金鑰已停用或待刪除，則會清除 Automatically rotate this KMS key every year (每年自動輪換此 KMS 金鑰)，而且您無法變更。當您啟用 KMS 金鑰或取消刪除時，金鑰輪換狀態會還原。如需詳細資訊，請參閱自動金鑰輪換的運作方式和AWS KMS 金鑰的金鑰狀態。

7. 選擇儲存。

啟用和停用金鑰輪換 (AWS KMS API)

您可以使用 AWS Key Management Service (AWS KMS) API 來啟用和停用自動金鑰輪換，並檢視任何客戶受管金鑰的目前輪換狀態。這些範例會使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

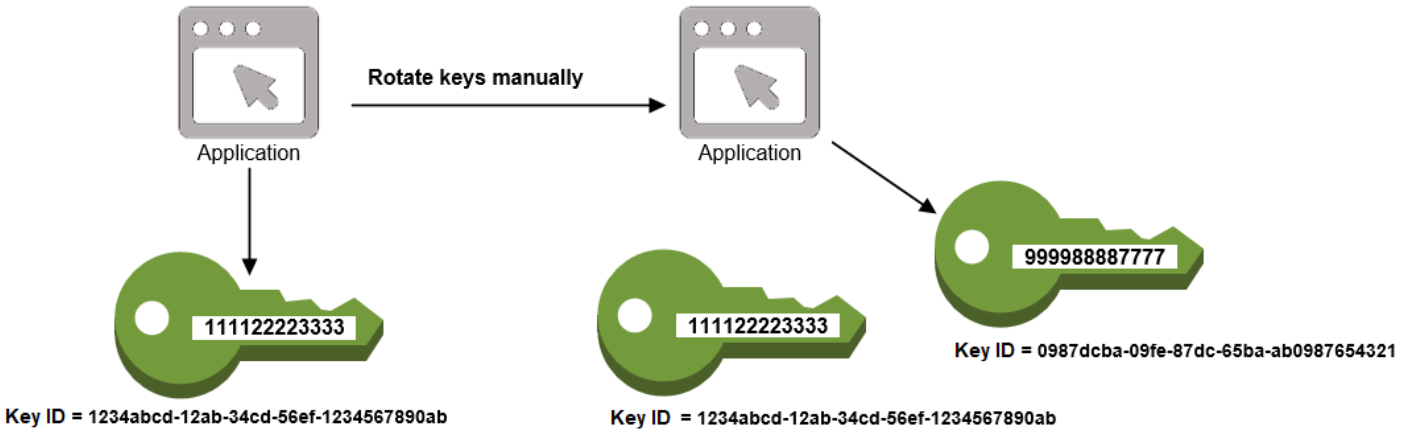
此EnableKeyRotation作業會為指定的 KMS 金鑰啟用自動金鑰輪替。該DisableKeyRotation操作將禁用它。若要在這些操作中識別 KMS 金鑰，請使用其金鑰 ID 或金鑰 ARN。依預設，客戶受管金鑰會停用金鑰輪換。

下列範例會針對指定的對稱加密 KMS 金鑰啟用金鑰輪替，並使用GetKeyRotationStatus作業查看結果。接著它停用金鑰輪換，並再次使用 GetKeyRotationStatus 來查看變更。

```
$ aws kms enable-key-rotation --key-id 1234abcd-12ab-34cd-56ef-1234567890ab

$ aws kms get-key-rotation-status --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
{
    "KeyRotationEnabled": true
}

$ aws kms disable-key-rotation --key-id 1234abcd-12ab-34cd-56ef-1234567890ab

$ aws kms get-key-rotation-status --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
{
    "KeyRotationEnabled": false
}
```

手動輪換金鑰

您可能會想要建立新的 KMS 金鑰，並使用它來取代目前的 KMS 金鑰，而不是啟用自動金鑰輪換。新的 KMS 金鑰擁有與目前 KMS 金鑰不同的密碼編譯資料，使用新的 KMS 金鑰與變更現有 KMS 金鑰中的金鑰材料具有相同的效果。將一個 KMS 金鑰更換為另一個 KMS 金鑰的程序稱為手動金鑰輪換。



您可能會想手動輪換金鑰，以便控制輪換頻率。對於不符合自動金鑰輪換資格的 KMS 金鑰，例如非對稱 KMS 金鑰、HMAC KMS 金鑰、位於自訂金鑰存放區中的 KMS 金鑰，以及帶有匯入金鑰材料的 KMS 金鑰，這也是一個很好的解決方案。

> **Note**
>
> 當您開始使用新的 KMS 金鑰，請務必保持原始 KMS 金鑰為啟用狀態，讓 AWS KMS 解密原始 KMS 金鑰加密的資料。

在手動輪換 KMS 金鑰時，您還需要更新應用程式中對 KMS 金鑰 ID 或金鑰 ARN 的參考。別名會將易記名稱關聯至 KMS 金鑰，讓此程序更容易進行。您可在應用程式中使用別名來參考 KMS 金鑰。而後，在您想要變更應用程式使用的 KMS 金鑰時，變更別名的目標 KMS 金鑰即可，而無需編輯應用程式代碼。如需詳細資訊，請參閱 在應用程式中使用別名。

> **Note**
>
> 指向手動輪替 KMS 金鑰的最新版本的別名對於「加密」DescribeKey、、、和「簽署」作業而言是很好的解決方案。GenerateDataKeyGenerateDataKeyPairGenerateMac管理 KMS 金鑰的作業 (例如DisableKey或) 不允許使用別名ScheduleKeyDeletion。
>
> 對手動輪換的對稱加密 KMS 金鑰呼叫 Decrypt 操作時，請省略命令中的 `KeyId` 參數。AWS KMS 會自動使用加密密文的 KMS 金鑰。
>
> 使用非對稱 KMS 金鑰呼叫`Decrypt`或驗證，或使用 HMAC KMS 金鑰呼叫VerifyMac時，需要此`KeyId`參數。當 `KeyId` 參數的值是不再指向執行密碼編譯操作的 KMS 金鑰的別名時 (例如手動輪換金鑰時)，這些請求將失敗。若要避免此錯誤，您必須追蹤每個操作並為其指定正確的 KMS 金鑰。

若要變更別名的目標 KMS 金鑰，請使用 AWS KMS API 中的UpdateAlias作業。例如，此命令會更新 alias/TestKey 別名以指向新的 KMS 金鑰。由於作業不會傳回任何輸出，因此範例會使用該ListAliases作業來顯示別名現在與不同的 KMS 金鑰相關聯，而且LastUpdatedDate欄位也會更新。這些 ListAliases 命令使用中的queryAWS CLI參數僅取得alias/TestKey別名。

```
$ aws kms list-aliases --query 'Aliases[?AliasName==`alias/TestKey`]'
{
    "Aliases": [
        {
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/TestKey",
            "AliasName": "alias/TestKey",
            "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "CreationDate": 1521097200.123,
            "LastUpdatedDate": 1521097200.123
        },
    ]
}


$ aws kms update-alias --alias-name alias/TestKey --target-key-id
 0987dcba-09fe-87dc-65ba-ab0987654321

$ aws kms list-aliases --query 'Aliases[?AliasName==`alias/TestKey`]'
{
    "Aliases": [
        {
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/TestKey",
            "AliasName": "alias/TestKey",
            "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
            "CreationDate": 1521097200.123,
            "LastUpdatedDate": 1604958290.722
        },
    ]
}
```

監控 AWS KMS keys

對於了解 AWS KMS 中 AWS KMS keys 的可用性、狀態和使用情形，以及維護 AWS 解決方案的可靠性、可用性和效能，監控是重要的一環。收集 AWS 解決方案全面的監控資料，可協助在出現多點故障時進行偵錯。不過，在您開始監控 KMS 金鑰之前，應先建立監控計畫，為下列問題提供解答：

- 監控目標是什麼？
- 要監控哪些資源？
- 監控這些資源的頻率為何？
- 要使用哪些監控工具？
- 誰將執行監控任務？
- 發生問題時應該通知誰？

下一個步驟是隨時間監控您的 KMS 金鑰，以建立您環境中 AWS KMS 正常使用和期望的基準。當您監控 KMS 金鑰時，請存放歷史記錄監控資料，如此才能與目前的資料做比較、辨識正常模式和異常狀況、規劃問題處理方式。

例如，您可以監控影響 KMS 金鑰的 AWS KMS API 活動和事件。當資料高於或低於既定常規，您可能需要調查或採取修正動作。

若要建立正常模式的基準，請監控下列項目：

- 資料平面操作的 AWS KMS API 活動。這些是使用 KMS 金鑰的密碼編譯作業，例如解密 ReEncrypthttps://docs.aws.amazon.com/kms/latest/APIReference/API_ReEncrypt.html、加密和 GenerateDataKey.
- 對您而言很重要之控制平面操作的 AWS KMS API 活動。這些作業會管理 KMS 金鑰，而且您可能想要監控變更 KMS 金鑰可用性的金鑰 (例如ScheduleKeyDeletionCancelKeyDeletion、DisableKey、EnableKey、ImportKeyMaterial、和 DeleteImportedKeyMaterial) 或變更 KMS 金鑰的存取控制 (例如PutKeyPolicy和 RevokeGrant) 的金鑰。
- 其他 AWS KMS 指標 (例如，匯入的金鑰材料過期之前剩餘的時間) 和事件 (例如匯入的金鑰材料已過期或刪除或 KMS 金鑰的金鑰輪換)。

監控工具

AWS 提供各種工具，可讓您監控 KMS 金鑰。您可以設定其中一些工具來進行監控，但有些工具需要手動介入。建議您盡可能自動化監控任務。

自動化監控工具

您可以使用下列自動化監控工具來監看 KMS 金鑰，並在發生變更時進行回報。

- AWS CloudTrail記錄監控 — 在帳戶之間共用記錄檔、透過將記 CloudTrail 錄檔傳送至 CloudWatch 記錄檔來即時監控記錄檔、使用處理程式庫寫入記錄CloudTrail 處理應用程式，以及驗證記錄檔在傳送之後是否未變更 CloudTrail。若要取得更多資訊，請參閱《使用指南》中的〈AWS CloudTrail使用 CloudTrail 記錄檔〉。
- Amazon CloudWatch 警示 — 觀看您指定期間內的單一指標，並根據指定臨界值在多個時段內相對於指定閾值的指標值執行一或多個動作。動作是傳送至亞馬遜簡單通知服務 (Amazon SNS) 主題或 Amazon EC2 Auto Scaling 政策的通知。 CloudWatch 警示不會僅因為處於特定狀態而叫用動作；狀態必須已變更並維持指定數目的期間。如需詳細資訊，請參閱 使用 Amazon 監控 CloudWatch。
- Amazon EventBridge — 匹配事件並將其路由到一個或多個目標函數或串流，以擷取狀態資訊，並在必要時進行變更或採取糾正措施。如需詳細資訊，請參閱使用 Amazon 監控 EventBridge和 Amazon EventBridge 使用者指南。
- Amazon CloudWatch 日誌 — 監控、存放和存取來自AWS CloudTrail或其他來源的日誌檔。如需詳細資訊，請參閱 Amazon CloudWatch 日誌使用者指南。

手動監控工具

監視 KMS 金鑰的另一個重要部分是手動監視 CloudWatch 警示和事件不涵蓋的項目。AWS KMS、CloudWatchAWS Trusted Advisor、和其他AWS儀表板可提供您AWS環境狀態的 at-a-glance 檢視。

您可以自訂 AWS KMS 主控台的 AWS 受管金鑰 和客戶受管金鑰頁面，以顯示有關每個 KMS 金鑰的下列資訊：

- 金鑰 ID
- Status
- 建立日期
- 過期日期 (適用於具有匯入金鑰材料的 KMS 金鑰)
- Origin
- 自訂金鑰存放區 ID (適用於自訂金鑰存放區中的 KMS 金鑰)

CloudWatch 主控台儀表板會顯示下列項目：

- 目前警示與狀態
- 警示與資源的圖表
- 服務運作狀態

此外，您可以使用執行 CloudWatch 以下操作：

- 建立自定儀表板來監控您注重的服務
- 繪製指標資料圖表，以對問題進行故障診斷並探索趨勢
- 搜尋與瀏覽您所有的 AWS 資源指標
- 建立與編輯要通知發生問題的警示

AWS Trusted Advisor 可協助您監控您的 AWS 資源，以改善效能、可靠性、安全與成本效益。所有使用者都可以使用四項 Trusted Advisor 檢查；具有商業或企業支援計畫的使用者可以使用超過 50 項以上的檢查。如需更多詳細資訊，請參閱 AWS Trusted Advisor。

使用 AWS CloudTrail 記錄 AWS KMS API 呼叫

AWS KMS與此服務整合 AWS CloudTrail，可記錄使用者、角色和其他AWS服務的所有呼叫。AWS KMS CloudTrail 擷取AWS KMS作為事件的所有 API 呼叫，包括來自AWS KMS主控台的呼叫、AWS KMS API、AWS CloudFormation範本、AWS Command Line Interface (AWS CLI) 和AWS Tools for PowerShell.

CloudTrail 記錄所有AWS KMS作業，包括唯讀作業 (例如ListAliases和 GetKeyRotationStatus) 管理 KMS 金鑰的作業 (例如CreateKey和) PutKeyPolicy，以及密碼編譯作業 (例如GenerateDataKey和解密)。它還會記錄AWS KMS呼叫您的內部操作 DeleteExpiredKeyMaterial，例如DeleteKey、SynchronizeMultiRegionKey、和RotateKey。

CloudTrail 記錄成功的作業，並嘗試失敗的呼叫，例如當呼叫者被拒絕存取資源時。對 KMS 金鑰的跨帳戶操作會同時記入呼叫者帳戶和 KMS 金鑰擁有者帳戶。不過，因存取遭拒而遭到拒絕的跨帳戶 AWS KMS 請求只會記錄在呼叫者的帳戶中。

基於安全理由，AWS KMS記錄項目會省略某些欄位，例如 En crypt 要求的`Plaintext`參數、回應GetKeyPolicy或任何密碼編譯作業。若要更輕鬆地搜尋特定 KMS 金鑰的 CloudTrail 記錄項目，請AWS KMS將受影響 KMS 金鑰的金鑰 ARN 新增至某些金AWS KMS鑰管理作業的記錄項目`responseElements`欄位，即使 API 作業未傳回金鑰 ARN 也一樣。

雖然依預設，所有AWS KMS動作都會記錄為 CloudTrail 事件，但您可以從 CloudTrail 追蹤中排除AWS KMS動作。如需詳細資訊，請參閱 從線索排除 AWS KMS 事件。

進一步了解：

- 如需AWS硝基飛地之AWS KMS作業的 CloudTrail 記錄範例，請參閱。對 Nitro Enclaves 的監空請求

主題

記錄事件 CloudTrail

CloudTrail 在您創建帳戶AWS 帳戶時啟用。當活動發生在中時AWS KMS，該活動會與事件歷史記錄中的其他AWS服務 CloudTrail 事件一起記錄在事件中。您可以檢視、搜尋和下載 AWS 帳戶 的最新事件。如需詳細資訊，請參閱檢視具有事 CloudTrail 件記錄的事件。

如需您 AWS 帳戶 帳戶中正在進行事件的記錄 (包含 AWS KMS 的事件)，請建立追蹤。追蹤可 CloudTrail 將日誌檔交付到 Amazon S3 儲存貯體。依預設，當您在主控台中建立追蹤時，該追蹤會套用至所有的 AWS 區域。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 CloudTrail 記錄中收集的事件資料並採取行動。如需詳細資訊，請參閱：

- 建立追蹤的概觀
- CloudTrail 支援的服務與整合
- 設定 Amazon SNS 通知 CloudTrail
- 從多個區域接收 CloudTrail 日誌文件和從多個帳戶接收 CloudTrail日誌文件

若要進一步了解 CloudTrail，請參閱使AWS CloudTrail用者指南。若要了解監控 KMS 金鑰使用之其他方式的詳細資訊，請參閱 監控 AWS KMS keys。

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：

- 該請求是否透過根憑證或 IAM 使用者憑證來提出。
- 提出該請求時，是否使用了角色或聯合身分使用者的暫時憑證。
- 該請求是否由其他 AWS 服務 提出。

如需詳細資訊，請參閱 CloudTrail 使用者身分元素。

搜尋事件 CloudTrail

若要搜尋 CloudTrail 記錄項目，請使用CloudTrail 主控台或CloudTrail LookupEvents作業。 CloudTrail 支援許多屬性值來篩選搜尋，包括事件名稱、使用者名稱和事件來源。

為了協助您在中搜尋AWS KMS記錄項目 CloudTrail，請AWS KMS填入下列 CloudTrail 記錄項目欄位。

> **Note**
>
> 從 2022 年 12 月開始，AWS KMS 會在可變更特定 KMS 金鑰的所有管理操作中填入 Resource type (資源類型) 和 Resource name (資源名稱) 屬性。對於下列操作，這些屬性值在較舊的 CloudTrail 項目中可能為 null：CreateAliasCreateGrantDeleteAliasDeleteImportedKeyMaterial、ImportKeyMaterial、ReplicateK 和UpdatePrimaryRegion。

屬性	值	日誌項目
事件來源 (`EventSource`)	`kms.amazonaws.com`	所有操作。
資源類型 (`ResourceType`)	`AWS::KMS::Key`	可變更特定 KMS 金鑰的管理操作，例如 `CreateKey` 和 `EnableKey` ，但不包括 `ListKeys`。
資源名稱 (`ResourceName`)	金鑰 ARN (或金鑰 ID 和金鑰 ARN)	可變更特定 KMS 金鑰的管理操作，例如 `CreateKey` 和 `EnableKey` ，但不包括 `ListKeys`。

為了協助您尋找特定 KMS 金鑰的管理操作的日誌項目，AWS KMS 會在日誌項目的 `responseElements.keyId` 元素中記錄受影響 KMS 金鑰的金鑰 ARN，即使 AWS KMS API 操作沒有傳回金鑰 ARN。

例如，成功呼叫DisableKey作業不會在回應中傳回任何值，而不是 null 值，DisableKey 記錄項目中的`responseElements.keyId`值包括停用的 KMS 金鑰的金鑰 ARN。

此功能於 2022 年 12 月新增，會影響下列 CloudTrail 記錄項目：CreateAliasCreateGrantDeleteAlias、DeleteKey、DisableKey、EnableKey、、EnableKeyRotation、Im和UpdatePrimaryRegion。

從線索排除 AWS KMS 事件

為了提供AWS KMS資源使用和管理的記錄，大多數使用AWS KMS者都依賴 CloudTrail 追蹤中的事件。追蹤可以是稽核重要事件的重要資料來源，例如建立、停用和刪除 AWS KMS keys，以及變更金鑰政策和代您依 AWS 服務使用 KMS 金鑰。在某些情況下， CloudTrail 記錄項目中的中繼資料 (例如加密作業中的加密內容) 可協助您避免或解決錯誤。

但是，因為 AWS KMS 會產生大量的事件，所以 AWS CloudTrail 讓您排除線索中的 AWS KMS 事件。此依線索設定會排除所有 AWS KMS 事件，所以無法排除特定的 AWS KMS 事件。

> **Warning**
>
> 從 CloudTrail 記錄中排除AWS KMS事件可能會隱藏使用 KMS 金鑰的動作。授予委託人執行此操作所需的 `cloudtrail:PutEventSelectors` 許可時時，請務必小心。

從線索中排除 AWS KMS 事件：

- 在 CloudTrail 主控台中，當您建立追蹤或更新追蹤時，請使用記錄金鑰管理服務事件設定。如需說明，請參閱《AWS CloudTrail 使用者指南》中的使用 AWS Management Console 記錄管理事件。
- 在 CloudTrail API 中，使用PutEventSelectors操作。將 `ExcludeManagementEventSources` 屬性新增加到您的事件選擇器，其值為 `kms.amazonaws.com`。如需範例，請參閱《AWS CloudTrail 使用者指南》中的範例：不記錄 AWS Key Management Service 事件的追蹤。

您可以變更主控台設定或線索的事件選擇器，以隨時停用此排除。然後，線索即會開始記錄 AWS KMS 事件。但無法復原排除有效期間發生過的 AWS KMS 事件。

當您使用主控台或 API 排除AWS KMS事件時，產生的 CloudTrail `PutEventSelectors` API 作業也會記錄在記 CloudTrail 錄中。如果AWS KMS事件未出現在記 CloudTrail 錄中，請尋找`ExcludeManagementEventSources`屬性設定為的`PutEventSelectors`事件`kms.amazonaws.com`。

AWS KMS 日誌項目的範例

AWS KMS當您呼叫作業，以及當AWS服務代表您呼叫AWS KMS作業時，會將項目寫入 CloudTrail 記錄。 AWS KMS當它為您調用操作時，也會寫入一個條目。例如，當刪除您排定刪除的 KMS 金鑰時，它會寫入一條項目。

下列主題顯示AWS KMS作業的 CloudTrail 記錄項目範例。

有關AWS KMS從 AWS Nitro Enclaves 請求的 CloudTrail 日誌條目的示例，請參閱。對 Nitro Enclaves 的監空請求

主題

- CancelKeyDeletion
- ConnectCustomKeyStore
- CreateAlias
- CreateCustomKeyStore
- CreateGrant
- CreateKey
- 解密
- DeleteAlias
- DeleteCustomKeyStore
- DeleteExpiredKeyMaterial
- DeleteImportedKeyMaterial
- DeleteKey
- DescribeCustomKeyStores
- DescribeKey
- DisableKey
- DisableKeyRotation
- DisconnectCustomKeyStore
- EnableKey
- EnableKeyRotation
- 加密
- GenerateDataKey

- GenerateDataKeyPair
- GenerateDataKeyPairWithoutPlaintext
- GenerateDataKeyWithoutPlaintext
- GenerateMac
- GenerateRandom
- GetKeyPolicy
- GetKeyRotationStatus
- GetParametersForImport
- ImportKeyMaterial
- ListAliases
- ListGrants
- PutKeyPolicy
- ReEncrypt
- ReplicateKey
- RetireGrant
- RevokeGrant
- RotateKey
- ScheduleKeyDeletion
- 符號
- SynchronizeMultiRegionKey
- TagResource
- UntagResource
- UpdateAlias
- UpdateCustomKeyStore
- UpdateKeyDescription
- UpdatePrimaryRegion
- VerifyMac
- 確認
- Amazon EC2 範例 1

- Amazon EC2 範例 2

CancelKeyDeletion

以下範例顯示藉由呼叫 AWS CloudTrail 操作而產生的 CancelKeyDeletion 日誌項目。如需刪除 AWS KMS keys 的資訊，請參閱 刪除 AWS KMS keys。


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2020-07-27T21:53:17Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CancelKeyDeletion",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": {
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "requestID": "e3452e68-d4b0-4ec7-a768-7ae96c23764f",
    "eventID": "d818bf03-6655-48e9-8b26-f279a07075fd",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
```

```
}
```

ConnectCustomKeyStore

以下範例顯示藉由呼叫 AWS CloudTrail 操作而產生的 ConnectCustomKeyStore 日誌項目。如需連線自訂金鑰存放區的詳細資訊，請參閱 連接和中斷連接 AWS CloudHSM 金鑰存放區。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2021-10-21T20:17:32Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "ConnectCustomKeyStore",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "customKeyStoreId": "cks-1234567890abcdef0"
    },
    "responseElements": null,
    "additionalEventData": {
        "customKeyStoreName": "ExampleKeyStore",
        "clusterId": "cluster-1a23b4cdefg"
    },
    "requestID": "abcde9e1-f1a3-4460-a423-577fb6e695c9",
    "eventID": "114b61b9-0ea6-47f5-a9d2-4f2bdd0017d5",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333"
}
```

CreateAlias

下列範例顯示作CreateAlias業的AWS CloudTrail記錄項目。resources 元素包含別名和 KMS 金鑰資源的欄位。如需有關在 AWS KMS 中建立別名的資訊，請參閱 建立別名。

CloudTrail 在 2022 年 12 月或之後記錄的此作業的記錄項目會在`responseElements.keyId`值中包含受影響 KMS 金鑰的金鑰 ARN，即使此作業不會傳回金鑰 ARN。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-08-14T23:08:31Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateAlias",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "aliasName": "alias/ExampleAlias",
        "targetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": {
        "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "requestID": "caec1e0c-ce03-419e-bdab-6ab1f7c57c01",
    "eventID": "2dd6e784-8286-46a6-befd-d64e5a02fb28",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        },
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-west-2:111122223333:alias/ExampleAlias"
        }
    ],
```

```
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

CreateCustomKeyStore

以下範例顯示藉由呼叫 AWS CloudHSM 金鑰存放區上的 `CreateCustomKeyStore` 操作而產生的 AWS CloudTrail 日誌項目。如需建立自訂金鑰存放區的詳細資訊，請參閱 建立 AWS CloudHSM 金鑰存放區。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2021-10-21T20:17:32Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateCustomKeyStore",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "customKeyStoreName": "ExampleKeyStore",
        "clusterId": "cluster-1a23b4cdefg"
    },
    "responseElements": {
        "customKeyStoreId": "cks-1234567890abcdef0"
    },
    "requestID": "abcde9e1-f1a3-4460-a423-577fb6e695c9",
    "eventID": "114b61b9-0ea6-47f5-a9d2-4f2bdd0017d5",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333"
}
```

CreateGrant

下列範例顯示作CreateGrant業的AWS CloudTrail記錄項目。如需有關在 AWS KMS 中建立授予的資訊，請參閱 AWS KMS 中的授權。

CloudTrail 在 2022 年 12 月或之後記錄的此作業的記錄項目會在`responseElements.keyId`值中包含受影響 KMS 金鑰的金鑰 ARN，即使此作業不會傳回金鑰 ARN。

```
{
  "eventVersion": "1.02",
  "userIdentity": {
      "type": "IAMUser",
      "principalId": "EX_PRINCIPAL_ID",
      "arn": "arn:aws:iam::111122223333:user/Alice",
      "accountId": "111122223333",
      "accessKeyId": "EXAMPLE_KEY_ID",
      "userName": "Alice"
  },
  "eventTime": "2014-11-04T00:53:12Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "CreateGrant",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "192.0.2.0",
  "userAgent": "AWS Internal",
  "requestParameters": {
      "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
      "constraints": {
          "encryptionContextSubset": {
              "ContextKey1": "Value1"
          }
      },
      "operations": ["Encrypt",
      "RetireGrant"],
      "granteePrincipal": "EX_PRINCIPAL_ID"
  },
  "responseElements": {
      "grantId": "f020fe75197b93991dc8491d6f19dd3cebb24ee62277a05914386724f3d48758",
      "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
  },
  "requestID": "f3c08808-63bc-11e4-bc2b-4198b6150d5c",
  "eventID": "5d529779-2d27-42b5-92da-91aaea1fc4b5",
```

```
  "readOnly": false,
  "resources": [{
      "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
      "accountId": "111122223333"
  }],
  "eventType": "AwsApiCall",
  "recipientAccountId": "111122223333"
}
```

CreateKey

這些範例顯示CreateKey作業的AWS CloudTrail記錄項目。

記CreateKey錄項目可能是由要CreateKey求或要求CreateKey作業所產生。ReplicateKey

下列範例顯示建立對稱加密 KMS 金鑰之CreateKey作業的 CloudTrail 記錄項目。如需建立 KMS 金鑰的相關資訊，請參閱 建立金鑰。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-08-10T22:38:27Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "description": "",
        "origin": "EXTERNAL",
        "bypassPolicyLockoutSafetyCheck": false,
        "customerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "keySpec": "SYMMETRIC_DEFAULT",
        "keyUsage": "ENCRYPT_DECRYPT"
    },
```

```
    "responseElements": {
        "keyMetadata": {
            "AWSAccountId": "111122223333",
            "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
            "creationDate": "Aug 10, 2022, 10:38:27 PM",
            "enabled": false,
            "description": "",
            "keyUsage": "ENCRYPT_DECRYPT",
            "keyState": "PendingImport",
            "origin": "EXTERNAL",
            "keyManager": "CUSTOMER",
            "customerMasterKeySpec": "SYMMETRIC_DEFAULT",
            "keySpec": "SYMMETRIC_DEFAULT",
            "encryptionAlgorithms": [
                "SYMMETRIC_DEFAULT"
            ],
            "multiRegion": false
        }
    },
    "requestID": "1aef6713-0223-4ff7-9a6d-781360521930",
    "eventID": "36327b37-f4f6-40a9-92ab-48064ec905a2",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


下列範例顯示在金鑰存放區中建立對稱加密 KMS 金鑰的`CreateKey`作AWS CloudHSM業 CloudTrail 記錄。


```
{
    "eventVersion": "1.08",
```

```
    "userIdentity": {
        "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
    },
    "eventTime": "2021-10-14T17:39:50Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyUsage": "ENCRYPT_DECRYPT",
        "bypassPolicyLockoutSafetyCheck": false,
        "origin": "AWS_CLOUDHSM",
        "keySpec": "SYMMETRIC_DEFAULT",
        "customerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "customKeyStoreId": "cks-1234567890abcdef0",
        "description": ""
    },
    "responseElements": {
        "keyMetadata": {
            "aWSAccountId": "111122223333",
            "keyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
            "arn": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321",
            "creationDate": "Oct 14, 2021, 5:39:50 PM",
            "enabled": true,
            "description": "",
            "keyUsage": "ENCRYPT_DECRYPT",
            "keyState": "Enabled",
            "origin": "AWS_CLOUDHSM",
            "customKeyStoreId": "cks-1234567890abcdef0",
            "cloudHsmClusterId": "cluster-1a23b4cdefg",
            "keyManager": "CUSTOMER",
            "customerMasterKeySpec": "SYMMETRIC_DEFAULT",
            "keySpec": "SYMMETRIC_DEFAULT",
            "encryptionAlgorithms": [
                "SYMMETRIC_DEFAULT"
            ],
            "multiRegion": false
```

```
        }
    },
    "additionalEventData": {
        "backingKey": "{\"keyHandle\":\"19\",\"backingKeyId\":\"backing-key-id\"}"
    },
    "requestID": "4f0b185c-588c-4767-9e90-c618f7e13cad",
    "eventID": "c73964b8-703d-49e4-bd9e-f773d0ee1e65",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

下列範例顯示在外部金鑰存放區中建立對稱加密 KMS 金鑰的CreateKey作業 CloudTrail 記錄。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
    },
    "eventTime": "2022-09-07T22:37:45Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "CreateKey",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "tags": [],
        "keyUsage": "ENCRYPT_DECRYPT",
```

```
        "description": "",
        "origin": "EXTERNAL_KEY_STORE",
        "multiRegion": false,
        "keySpec": "SYMMETRIC_DEFAULT",
        "customerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "bypassPolicyLockoutSafetyCheck": false,
        "customKeyStoreId": "cks-1234567890abcdef0",
        "xksKeyId": "bb8562717f809024"
    },
    "responseElements": {
        "keyMetadata": {
            "aWSAccountId": "111122223333",
            "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
            "creationDate": "Dec 7, 2022, 10:37:45 PM",
            "enabled": true,
            "description": "",
            "keyUsage": "ENCRYPT_DECRYPT",
            "keyState": "Enabled",
            "origin": "EXTERNAL_KEY_STORE",
            "customKeyStoreId": "cks-1234567890abcdef0",
            "keyManager": "CUSTOMER",
            "customerMasterKeySpec": "SYMMETRIC_DEFAULT",
            "keySpec": "SYMMETRIC_DEFAULT",
            "encryptionAlgorithms": [
                "SYMMETRIC_DEFAULT"
            ],
            "multiRegion": false,
            "xksKeyConfiguration": {
                "id": "bb8562717f809024"
            }
        }
    },
    "requestID": "ba197c82-3ac7-487a-8ff4-7736bbeb1316",
    "eventID": "838ad5f4-5fdd-4044-afd7-4dbd88c6af56",
    "readOnly": false,
    "resources": [
        {
            "accountId": "227179770375",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-east-1:227179770375:key/39c5eb22-
f37c-4956-92ca-89e8f8b57ab2"
        }
```

```
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

解密

這些範例顯示 Decrypt 操作的 AWS CloudTrail 的日誌項目。

requestParameters即使未在要求encryptionAlgorithm中指定加密演算法，Decrypt作業的 CloudTrail 記錄項目仍會包含中的。省略請求中的加密文字和回應中的純文字。

主題

- 使用標準對稱加密金鑰進行解密
- 使用標準對稱加密金鑰進行解密失敗
- 使用 AWS CloudHSM 金鑰存放區中的 KMS 金鑰進行解密
- 使用外部金鑰存放區中的 KMS 金鑰進行解密
- 使用外部金鑰存放區中的 KMS 金鑰進行解密失敗

使用標準對稱加密金鑰進行解密

以下是具有標準對稱加密金鑰之Decrypt作業的 CloudTrail 記錄項目範例。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2020-07-27T22:58:24Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
```

```
    "requestParameters": {
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "encryptionContext": {
            "Department": "Engineering",
            "Project": "Alpha"
        }
    },
    "responseElements": null,
    "requestID": "12345126-30d5-4b28-98b9-9153da559963",
    "eventID": "abcde202-ba1a-467c-b4ba-f729d45ae521",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

使用標準對稱加密金鑰進行解密失敗

下列範例記 CloudTrail 錄項目會使用標準對稱加密 KMS 金鑰記錄失敗的`Decrypt`作業。包括例外狀況 (`errorCode`) 和錯誤消息 (`errorMessage`)，可幫助您解決錯誤。

在這種情況下，在 `Decrypt` 請求中指定的對稱加密 KMS 金鑰不是用來加密資料的對稱加密 KMS 金鑰。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
```

```
    "eventTime": "2022-11-24T18:57:43Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "errorCode": "IncorrectKeyException"
    "errorMessage": "The key ID in the request does not identify a CMK that can perform
 this operation.",
    "requestParameters": {
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "encryptionContext": {
            "Department": "Engineering",
            "Project": "Alpha"
        }
    },
    "responseElements": null,
    "requestID": "22345126-30d5-4b28-98b9-9153da559963",
    "eventID": "abcde202-ba1a-467c-b4ba-f729d45ae521",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

使用 AWS CloudHSM 金鑰存放區中的 KMS 金鑰進行解密

下列範例記 CloudTrail 錄項目會在金鑰存放區中記錄具有 KMS 金AWS CloudHSM鑰的Decrypt作業。使用自訂金鑰存放區中的 KMS 金鑰進行密碼編譯操作的所有日誌項目都包含具有 customKeyStoreId 的 additionalEventData 欄位。請求中未指定 additionalEventData。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
```

```
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2021-10-26T23:41:27Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "requestParameters": {
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "encryptionContext": {
            "Department": "Development",
            "Purpose": "Test"
        }
    },
    "responseElements": null,
    "additionalEventData": {
        "customKeyStoreId": "cks-1234567890abcdef0"
    },
    "requestID": "e1b881f8-2048-41f8-b6cc-382b7857ec61",
    "eventID": "a79603d5-4cde-46fc-819c-a7cf547b9df4",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

使用外部金鑰存放區中的 KMS 金鑰進行解密

下列範例記 CloudTrail 錄項目會在外部金鑰存放區中記錄具有 KMS 金鑰的Decrypt作業。除了customKeyStoreId 之外，additionalEventData 欄位還包括外部金鑰 ID (XksKeyId)。請求中未指定 additionalEventData。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-11-24T00:26:58Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "AWS Internal",
    "requestParameters": {
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "keyId": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321",
        "encryptionContext": {
            "Department": "Engineering",
            "Purpose": "Test"
        }
    },
    "responseElements": null,
    "additionalEventData": {
        "customKeyStoreId": "cks-9876543210fedcba9",
        "xksKeyId": "abc01234567890fe"
    },
    "requestID": "f1b881f8-2048-41f8-b6cc-382b7857ec61",
    "eventID": "b79603d5-4cde-46fc-819c-a7cf547b9df4",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
```

```
            "ARN": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

使用外部金鑰存放區中的 KMS 金鑰進行解密失敗

下列範例記 CloudTrail 錄項目會在外部金鑰存放區中，針對具有 KMS 金鑰的`Decrypt`作業，記錄失敗的要求。 CloudWatch 除了成功的要求之外，還會記錄失敗的要求。記錄失敗時， CloudTrail 記錄項目會包含例外狀況 (errorCode) 和隨附的 errorMessage (錯誤訊息)。

如果失敗的請求到達您的外部金鑰存放區代理 (如本範例所示)，則您可以使用 `requestId` 值將失敗的請求與外部金鑰存放區代理所記錄的對應請求建立關聯 (如果代理提供的話)。

如需外部金鑰存放區中 `Decrypt` 請求的說明，請參閱 解密錯誤。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-11-24T00:26:58Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "errorCode": "KMSInvalidStateException",
    "errorMessage": "The external key store proxy rejected the request because the
 specified ciphertext or additional authenticated data is corrupted, missing, or
 otherwise invalid.",
    "requestParameters": {
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
```

```
        "keyId": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321",
        "encryptionContext": {
            "Department": "Engineering",
            "Purpose": "Test"
        }
    },
    "responseElements": null,
    "additionalEventData": {
        "customKeyStoreId": "cks-9876543210fedcba9",
        "xksKeyId": "abc01234567890fe"
    },
    "requestID": "f1b881f8-2048-41f8-b6cc-382b7857ec61",
    "eventID": "b79603d5-4cde-46fc-819c-a7cf547b9df4",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

DeleteAlias

下列範例顯示作DeleteAlias業的AWS CloudTrail記錄項目。如需刪除別名的詳細資訊，請參閱 刪除別名。

CloudTrail 在 2022 年 12 月或之後記錄的此作業的記錄項目會在`responseElements.keyId`值中包含受影響 KMS 金鑰的金鑰 ARN，即使此作業不會傳回金鑰 ARN。

```
{
    "eventVersion": "1.02",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
```

```
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2014-11-04T00:52:27Z"
            }
        }
    },
    "eventTime": "2014-11-04T00:52:27Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DeleteAlias",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "aliasName": "alias/my_alias"
    },
    "responseElements": {
        "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "requestID": "d9542792-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "12f48554-bb04-4991-9cfc-e7e85f68eda0",
    "readOnly": false,
    "resources": [{
        "ARN": "arn:aws:kms:us-east-1:111122223333:alias/my_alias",
        "accountId": "111122223333"
    },
    {
        "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "accountId": "111122223333"
    }],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

DeleteCustomKeyStore

以下範例顯示藉由呼叫 AWS CloudTrail 操作而產生的 DeleteCustomKeyStore 日誌項目。如需建立自訂金鑰存放區的詳細資訊，請參閱 刪除 AWS CloudHSM 金鑰存放區。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2021-10-21T20:17:32Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DeleteCustomKeyStore",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "customKeyStoreId": "cks-1234567890abcdef0"
    },
    "responseElements": null,
    "additionalEventData": {
        "customKeyStoreName": "ExampleKeyStore",
        "clusterId": "cluster-1a23b4cdefg"
    },
    "requestID": "abcde9e1-f1a3-4460-a423-577fb6e695c9",
    "eventID": "114b61b9-0ea6-47f5-a9d2-4f2bdd0017d5",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333"
}
```


DeleteExpiredKeyMaterial

將金鑰材料匯入 AWS KMS key (KMS 金鑰) 時，您可以為該金鑰材料設定到期日期和時間。 AWS KMS當您 CloudTrail 匯入金鑰材料 (使用到期設定) 以及何時AWS KMS刪除過期的金鑰材料時，會在記錄中記錄項目。如需有關建立具有匯入金鑰材料之 KMS 金鑰的資訊，請參閱 匯入 AWS KMS 金鑰的金鑰材料。

以下範例顯示 AWS KMS 刪除過期的金鑰材料時產生的 AWS CloudTrail 日誌項目。


```
{
    "eventVersion": "1.05",
```

```
    "userIdentity": {
        "accountId": "111122223333",
        "invokedBy": "AWS Internal"
    },
    "eventTime": "2021-01-01T16:00:00Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DeleteExpiredKeyMaterial",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": null,
    "responseElements": null,
    "eventID": "cfa932fd-0d3a-4a76-a8b8-616863a2b547",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsServiceEvent",
    "recipientAccountId": "111122223333",
    "serviceEventDetails": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    }
}
```

DeleteImportedKeyMaterial

如果您將金鑰材料匯入 KMS 金鑰，您可以隨時使用DeleteImportedKeyMaterial作業刪除匯入的金鑰材料。從 KMS 金鑰中刪除匯入金鑰材料後，KMS 金鑰的金鑰狀態變更為 PendingImport，並且不能在任何密碼編譯操作中使用 KMS 金鑰。如需詳細資訊，請參閱 刪除匯入的金鑰材料。

以下範例顯示針對 DeleteImportedKeyMaterial 操作產生的 AWS CloudTrail 日誌項目。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
```

```
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-10-04T21:43:33Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DeleteImportedKeyMaterial",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": {
        "keyId":"&example-key-arn-1;"
    },
    "requestID": "dcf0e82f-dad0-4622-a378-a5b964ad42c1",
    "eventID": "2afbb991-c668-4641-8a00-67d62e1fecbd",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

DeleteKey

這些範例顯示刪除 KMS 金鑰時產生的 AWS CloudTrail 日誌項目。若要刪除 KMS 金鑰，請使用ScheduleKeyDeletion作業。指定的等待期過期後，AWS KMS刪除 KMS 金鑰，並在記錄 CloudTrail 檔中記錄下列項目，以記錄該事件。

CloudTrail 在 2022 年 12 月或之後記錄的此作業的記錄項目會在`responseElements.keyId`值中包含受影響 KMS 金鑰的金鑰 ARN，即使此作業未傳回金鑰 ARN。

如需作ScheduleKeyDeletion業的 CloudTrail 記錄項目範例，請參閱ScheduleKeyDeletion。如需刪除 KMS 金鑰的相關資訊，請參閱 刪除 AWS KMS keys。

下列範例記 CloudTrail 錄項目會記錄包含金鑰材料的 KMS 金鑰DeleteKey作業AWS KMS。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "accountId": "111122223333",
        "invokedBy": "AWS Internal"
    },
    "eventTime": "2020-07-31T00:07:00Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DeleteKey",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": null,
    "responseElements": null,
    "eventID": "b25f9cda-74e1-4458-847b-4972a0bf9668",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsServiceEvent",
    "recipientAccountId": "111122223333",
     "managementEvent": true,
    "eventCategory": "Management"
}
```

下列記 CloudTrail 錄項目會記錄AWS CloudHSM自訂金鑰存放區中 KMS 金鑰的DeleteKey作業。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "accountId": "111122223333",
        "invokedBy": "AWS Internal"
    },
```

```
    "eventTime": "2021-10-26T23:41:27Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DeleteKey",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": null,
    "responseElements": {
        "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "additionalEventData": {
        "customKeyStoreId": "cks-1234567890abcdef0",
        "clusterId": "cluster-1a23b4cdefg",
        "backingKeys": "[{\"keyHandle\":\"01\",\"backingKeyId\":\"backing-key-id\"}]",
        "backingKeysDeletionStatus": "[{\"keyHandle\":\"01\",\"backingKeyId\":
\"backing-key-id\",\"deletionStatus\":\"SUCCESS\"}]"
    },
    "eventID": "1234585c-4b0c-4340-ab11-662414b79239",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsServiceEvent",
    "recipientAccountId": "111122223333",
    "managementEvent": true,
    "eventCategory": "Management"
}
```

DescribeCustomKeyStores

以下範例顯示藉由呼叫 AWS CloudTrail 操作而產生的 DescribeCustomKeyStores 日誌項目。如需檢視自訂金鑰存放區的詳細資訊，請參閱 檢視 AWS CloudHSM 金鑰存放區。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
```

```
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2021-10-21T20:17:32Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DescribeCustomKeyStores",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "customKeyStoreId": "cks-1234567890abcdef0"
    },
    "responseElements": null,
    "requestID": "abcde9e1-f1a3-4460-a423-577fb6e695c9",
    "eventID": "2ea1735f-628d-43e3-b2ee-486d02913a78",
    "readOnly": true,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333"
}
```

DescribeKey

下列範例顯示作DescribeKey業的AWS CloudTrail記錄項目。 AWS KMS當您呼叫AWS KMS主控台中的DescribeKey作業或檢視 KMS 金鑰時，會記錄如下所示的項目。此呼叫是在 AWS KMS 管理主控台中檢視金鑰的結果。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-09-26T18:01:36Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DescribeKey",
```

```
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": null,
    "requestID": "12345126-30d5-4b28-98b9-9153da559963",
    "eventID": "abcde202-ba1a-467c-b4ba-f729d45ae521",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

DisableKey

下列範例顯示作DisableKey業的AWS CloudTrail記錄項目。如需有關啟用和停用 AWS KMS 中 AWS KMS keys 的資訊，請參閱 啟用和停用金鑰。

CloudTrail 在 2022 年 12 月或之後記錄的此作業的記錄項目會在`responseElements.keyId`值中包含受影響 KMS 金鑰的金鑰 ARN，即使此作業不會傳回金鑰 ARN。

```
{
    "eventVersion": "1.02",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2014-11-04T00:52:43Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DisableKey",
```

```
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": {
        "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "requestID": "12345126-30d5-4b28-98b9-9153da559963",
    "eventID": "abcde202-ba1a-467c-b4ba-f729d45ae521",
    "readOnly": false,
    "resources": [{
        "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "accountId": "111122223333"
    }],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```


DisableKeyRotation

以下範例顯示藉由呼叫 AWS CloudTrail 操作而產生的 DisableKeyRotation 日誌項目。如需有關自動金鑰輪換的詳細資訊，請參閱 輪換 AWS KMS keys。


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-09-01T19:31:39Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DisableKeyRotation",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
```

```
    "requestParameters": {
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": null,
    "requestID": "d6a9351a-ed6e-4581-88d1-2a9a8a538497",
    "eventID": "6313164c-83aa-4cc3-9e1a-b7c426f7a5b1",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


DisconnectCustomKeyStore

以下範例顯示藉由呼叫 AWS CloudTrail 操作而產生的 `DisconnectCustomKeyStore` 日誌項目。如需中斷連線自訂金鑰存放區的詳細資訊，請參閱 連接和中斷連接 AWS CloudHSM 金鑰存放區。


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2021-10-21T20:17:32Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DisconnectCustomKeyStore",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
```

```
    "requestParameters": {
        "customKeyStoreId": "cks-1234567890abcdef0"
    },
    "responseElements": null,
    "additionalEventData": {
        "customKeyStoreName": "ExampleKeyStore",
        "clusterId": "cluster-1a23b4cdefg"
    },
    "requestID": "abcde9e1-f1a3-4460-a423-577fb6e695c9",
    "eventID": "114b61b9-0ea6-47f5-a9d2-4f2bdd0017d5",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333"
}
```

EnableKey

下列範例顯示作EnableKey業的AWS CloudTrail記錄項目。如需有關啟用和停用 AWS KMS 中 AWS KMS keys 的資訊，請參閱 啟用和停用金鑰。

CloudTrail 在 2022 年 12 月或之後記錄的此作業的記錄項目會在`responseElements.keyId`值中包含受影響 KMS 金鑰的金鑰 ARN，即使此作業未傳回金鑰 ARN。

```
{
    "eventVersion": "1.02",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2014-11-04T00:52:20Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "EnableKey",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
```

```
    "responseElements": {
        "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "requestID": "d528a6fb-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "be393928-3629-4370-9634-567f9274d52e",
    "readOnly": false,
    "resources": [{
        "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "accountId": "111122223333"
    }],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

EnableKeyRotation

下列範例顯示呼叫EnableKeyRotation作業的AWS CloudTrail記錄項目。如需旋轉金鑰時所寫入之 CloudTrail 記錄項目的範例，請參閱RotateKey。如需關於輪換 AWS KMS keys 的資訊，請參閱 輪換 AWS KMS keys。

CloudTrail 在 2022 年 12 月或之後記錄的此作業的記錄項目會在`responseElements.keyId`值中包含受影響 KMS 金鑰的金鑰 ARN，即使此作業未傳回金鑰 ARN。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2020-07-25T23:41:56Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "EnableKeyRotation",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
```

```
    },
    "responseElements": {
        "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "requestID": "81f5b794-452b-4d6a-932b-68c188165273",
    "eventID": "fefc43a7-8e06-419f-bcab-b3bf18d6a401",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

加密

以下範例顯示 Encrypt 操作的 AWS CloudTrail 日誌項目。

```
{
    "eventVersion": "1.02",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-07-14T20:17:42Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Encrypt",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "encryptionContext": {
            "Department": "Engineering"
        },
```

```
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
    },
    "responseElements": null,
    "requestID": "f3423043-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "91235988-eb87-476a-ac2c-0cdc244e6dca",
    "readOnly": true,
    "resources": [{
        "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "accountId": "111122223333"
    }],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

GenerateDataKey

下列範例顯示作GenerateDataKey業的AWS CloudTrail記錄項目。

```
{
    "eventVersion": "1.02",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2014-11-04T00:52:40Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKey",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "keySpec": "AES_256",
        "encryptionContext": {
            "Department": "Engineering",
            "Project": "Alpha"
        }
```

```
    },
    "responseElements": null,
    "requestID": "e0eb83e3-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "a9dea4f9-8395-46c0-942c-f509c02c2b71",
    "readOnly": true,
    "resources": [{
        "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "accountId": "111122223333"
    }],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

GenerateDataKeyPair

下列範例顯示作GenerateDataKeyPair業的AWS CloudTrail記錄項目。此範例會記錄產生 RSA 金鑰對的操作，該 RSA 金鑰對採用對稱加密 AWS KMS key 進行加密。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
    },
    "eventTime": "2020-07-27T18:57:57Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKeyPair",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyPairSpec": "RSA_3072",
        "encryptionContext": {
            "Project": "Alpha"
        },
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": null,
```

```
    "requestID": "52fb127b-0fe5-42bb-8e5e-f560febde6b0",
    "eventID": "9b6bd6d2-529d-4890-a949-593b13800ad7",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

GenerateDataKeyPairWithoutPlaintext

下列範例顯示作GenerateDataKeyPairWithoutPlaintext業的AWS CloudTrail記錄項目。此範例會記錄產生 RSA 金鑰對的操作，該 RSA 金鑰對採用對稱加密 AWS KMS key 進行加密。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
    },
    "eventTime": "2020-07-27T18:57:57Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKeyPairWithoutPlaintext",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyPairSpec": "RSA_4096",
        "encryptionContext": {
            "Index": "5"
        },
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
```

```
    "responseElements": null,
    "requestID": "52fb127b-0fe5-42bb-8e5e-f560febde6b0",
    "eventID": "9b6bd6d2-529d-4890-a949-593b13800ad7",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

GenerateDataKeyWithoutPlaintext

下列範例顯示作GenerateDataKeyWithoutPlaintext業的AWS CloudTrail記錄項目。

```
{
    "eventVersion": "1.02",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2014-11-04T00:52:23Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKeyWithoutPlaintext",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "errorCode": "InvalidKeyUsageException",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "keySpec": "AES_256",
        "encryptionContext": {
            "Project": "Alpha"
        }
    },
```

```
    "responseElements": null,
    "requestID": "d6b8e411-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "f7734272-9ec5-4c80-9f36-528ebbe35e4a",
    "readOnly": true,
    "resources": [{
        "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "accountId": "111122223333"
    }],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

GenerateMac

下列範例顯示作GenerateMac業的AWS CloudTrail記錄項目。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
     },
    "eventTime": "2022-12-23T19:26:54Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateMac",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "macAlgorithm": "HMAC_SHA_512",
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": null,
    "requestID": "e0eb83e3-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "a9dea4f9-8395-46c0-942c-f509c02c2b71",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
```

```
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

GenerateRandom

下列範例顯示作GenerateRandom業的AWS CloudTrail記錄項目。因為此操作不使用 AWS KMS key，所以 `resources` 欄位為空白。

```
{
    "eventVersion": "1.02",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2014-11-04T00:52:37Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateRandom",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": null,
    "responseElements": null,
    "requestID": "df1e3de6-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "239cb9f7-ae05-4c94-9221-6ea30eef0442",
    "readOnly": true,
    "resources": [],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

GetKeyPolicy

下列範例顯示作GetKeyPolicy業的AWS CloudTrail記錄項目。如需檢視 KMS 金鑰之金鑰政策的詳細資訊，請參閱 檢視金鑰政策。


```
{
    "eventVersion": "1.02",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2014-11-04T00:50:30Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GetKeyPolicy",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "policyName": "default"
    },
    "responseElements": null,
    "requestID": "93746dd6-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "4aa7e4d5-d047-452a-a5a6-2cce282a7e82",
    "readOnly": true,
    "resources": [{
        "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "accountId": "111122223333"
    }],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```


GetKeyRotationStatus

下列範例顯示作GetKeyRotationStatus業的AWS CloudTrail記錄項目。如需有關 KMS 金鑰之金鑰材料的自動輪換詳細資訊，請參閱 輪換 AWS KMS keys。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-09-01T19:32:11Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GetKeyRotationStatus",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": null,
    "requestID": "12f9b7e8-49b9-4c1c-a7e3-34ac0cdf0467",
    "eventID": "3d082126-9e7d-4167-8372-a6cfcbed4be6",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


GetParametersForImport

下列範例顯示使用GetParametersForImport作業時產生的AWS CloudTrail記錄項目。此操作會傳回您在將金鑰材料匯入 KMS 金鑰時使用的公有金鑰和匯入字符。當您使用GetParametersForImport操作或使用AWS KMS控制台下載公鑰和導入令牌時，會記錄相同的 CloudTrail 條目。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
    },
    "eventTime": "2020-07-25T23:58:23Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GetParametersForImport",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "wrappingAlgorithm": "RSAES_OAEP_SHA_256",
        "wrappingKeySpec": "RSA_2048"
    },
    "responseElements": null,
    "requestID": "b5786406-e3c7-43d6-8d3c-6d5ef96e2278",
    "eventID": "4023e622-0c3e-4324-bdef-7f58193bba87",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```


ImportKeyMaterial

下列範例顯示使用ImportKeyMaterial作業時產生的AWS CloudTrail記錄項目。當您使用`ImportKeyMaterial`操作或使用AWS KMS控制台將關鍵材料導入到AWS KMS key. CloudTrail

CloudTrail 在 2022 年 12 月或之後記錄的此作業的記錄項目會在`responseElements.keyId`值中包含受影響 KMS 金鑰的金鑰 ARN，即使此作業不會傳回金鑰 ARN。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
    },
    "eventTime": "2020-07-26T00:08:00Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "ImportKeyMaterial",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "validTo": "Jan 1, 2021 8:00:00 PM",
        "expirationModel": "KEY_MATERIAL_EXPIRES"
    },
    "responseElements": {
        "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "requestID": "89e10ee7-a612-414d-95a2-a128346969fd",
    "eventID": "c7abd205-a5a2-4430-bbfa-fc10f3e2d79f",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

ListAliases

下列範例顯示作ListAliases業的AWS CloudTrail記錄項目。因為此操作不使用任何特定別名或 AWS KMS key，所以 `resources` 欄位為空白。如需在 AWS KMS 中檢視別名的更多資訊，請參閱 檢視別名。


```
{
    "eventVersion": "1.02",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2014-11-04T00:51:45Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "ListAliases",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "limit": 5,
        "marker":
 "eyJiIjoiYWxpYXMvZTU0Y2MxOTMtYTMwNC00YzEwLTliZWItYTJjZjA3NjA2OTJhIiwiYSI6ImFsaWFzL2U1NGNjMTkzL
    },
    "responseElements": null,
    "requestID": "bfe6c190-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "a27dda7b-76f1-4ac3-8b40-42dfba77bcd6",
    "readOnly": true,
    "resources": [],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```


ListGrants

下列範例顯示作ListGrant業的AWS CloudTrail記錄項目。如需 AWS KMS 授予的詳細資訊，請參閱 AWS KMS 中的授權。


```
{
    "eventVersion": "1.02",
```

```
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2014-11-04T00:52:49Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "ListGrants",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "marker":
 "eyJncmFudElkIjoiMWY4M2U2ZmM0YTY2NDgxYjQ2Yzc4MTdhM2Y4YmQwMDFkZDNiYmQ1MGVlYTMyY2RmOWFiNWY1Nzc1N
\u003d\u003d",
        "limit": 10
    },
    "responseElements": null,
    "requestID": "e5c23960-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "d24380f5-1b20-4253-8e92-dd0492b3bd3d",
    "readOnly": true,
    "resources": [{
        "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "accountId": "111122223333"
    }],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```


PutKeyPolicy

以下範例顯示藉由呼叫 AWS CloudTrail 操作而產生的 PutKeyPolicy 日誌項目。如需有關更新金鑰政策的資訊，請參閱 "變更金鑰政策"。


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
```

```
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-09-01T20:06:16Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "PutKeyPolicy",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "policyName": "default",
        "policy": "{\n  \"Version\" : \"2012-10-17\",\n  \"Id\" : \"key-default-1\",\n
 \"Statement\" : [ {\n    \"Sid\" : \"Enable IAM User Permissions\",\n    \"Effect\" :
 \"Allow\",\n    \"Principal\" : {\n      \"AWS\" : \"arn:aws:iam::111122223333:root
\"\n    },\n    \"Action\" : \"kms:*\",\n    \"Resource\" : \"*\"\n  } ]\n}",
        "bypassPolicyLockoutSafetyCheck": false
    },
    "responseElements": null,
    "requestID": "7bb906fa-dc21-4350-b65c-808ff0f72f55",
    "eventID": "c217db1f-903f-4a2f-8f88-9580182d6313",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


ReEncrypt

下列範例顯示作ReEncrypt業的AWS CloudTrail記錄項目。此日誌項目的 `resources` 欄位會指定兩個 AWS KMS keys、來源 KMS 金鑰和目的地 KMS 金鑰，依此順序排列。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2020-07-27T23:09:13Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "ReEncrypt",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "sourceEncryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "sourceEncryptionContext": {
            "Project": "Alpha",
            "Department": "Engineering"
        },
        "destinationKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
        "destinationEncryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "destinationEncryptionContext": {
            "Level": "3A"
        }
    },
    "responseElements": null,
    "requestID": "03769fd4-acf9-4b33-adf3-2ab8ca73aadf",
    "eventID": "542d9e04-0e8d-4e05-bf4b-4bdeb032e6ec",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        },
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
```

```
            "ARN": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

ReplicateKey

以下範例顯示藉由呼叫 AWS CloudTrail 操作而產生的 ReplicateKey 日誌項目。要ReplicateKey求會產生ReplicateKey作業和作CreateKey業。

如需複寫多區域金鑰的相關資訊，請參閱 建立多區域複本金鑰。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2020-11-18T01:29:18Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "ReplicateKey",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "replicaRegion": "us-west-2",
        "bypassPolicyLockoutSafetyCheck": false,
        "description": ""
    },
    "responseElements": {
        "replicaKeyMetadata": {
            "aWSAccountId": "111122223333",
            "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
```

```
            "creationDate": "Nov 18, 2020, 1:29:18 AM",
            "enabled": false,
            "description": "",
            "keyUsage": "ENCRYPT_DECRYPT",
            "keyState": "Creating",
            "origin": "AWS_KMS",
            "keyManager": "CUSTOMER",
            "keySpec": "SYMMETRIC_DEFAULT",
            "customerMasterKeySpec": "SYMMETRIC_DEFAULT",
            "encryptionAlgorithms": [
                "SYMMETRIC_DEFAULT"
            ],
            "multiRegion": true,
            "multiRegionConfiguration": {
                "multiRegionKeyType": "REPLICA",
                "primaryKey": {
                    "arn": "arn:aws:kms:us-
east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
                    "region": "us-east-1"
                },
                "replicaKeys": [
                    {
                        "arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
                        "region": "us-west-2"
                    }
                ]
            }
        },
        "replicaPolicy": "{\n  \"Version\":\"2012-10-17\",\n  \"Statement\":[{\n
 \"Effect\":\"Allow\",\n    \"Principal\":{\"AWS\":\"arn:aws:iam::123456789012:user/
Alice\"},\n    \"Action\":\"kms:*\",\n    \"Resource\":\"*\"\n  }, {\n    \"Effect
\":\"Allow\",\n    \"Principal\":{\"AWS\":\"arn:aws:iam::012345678901:user/Bob\"},\n
    \"Action\":\"kms:CreateGrant\",\n    \"Resource\":\"*\"\n  }, {\n    \"Effect\":
\"Allow\",\n    \"Principal\":{\"AWS\":\"arn:aws:iam::012345678901:user/Charlie\"},\n
  \"Action\":\"kms:Encrypt\",\n    \"Resource\":\"*\"\n}]\n}",
    },
    "requestID": "abcdef68-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "fedcba44-6773-4f96-8763-1993aec9ae6a",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
```

```
            "ARN": "arn:aws:kms:us-
east-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

RetireGrant

以下範例顯示藉由呼叫 AWS CloudTrail 操作而產生的 RetireGrant 日誌項目。如需有關淘汰授予的詳細資訊，請參閱 淘汰和撤銷授予。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-09-01T19:39:33Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "RetireGrant",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": null,
    "responseElements": null,
    "additionalEventData": {
        "grantId": "abcde1237f76e4ba7987489ac329fbfba6ad343d6f7075dbd1ef191f0120514a"
    },
    "requestID": "1d274d57-5697-462c-a004-f25fcc29fa26",
    "eventID": "0771bcfb-3e24-4332-9ac8-e1c06563eecf",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
```

```
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

RevokeGrant

以下範例顯示藉由呼叫 AWS CloudTrail 操作而產生的 RevokeGrant 日誌項目。如需有關撤銷授予的詳細資訊，請參閱 淘汰和撤銷授予。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-09-01T19:35:17Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "RevokeGrant",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "grantId": "abcde1237f76e4ba7987489ac329fbfba6ad343d6f7075dbd1ef191f0120514a"
    },
    "responseElements": null,
    "requestID": "59d94c03-c5b7-428d-ae6e-f2c4b47d2917",
    "eventID": "07a23a39-6526-4ae2-b31e-d35fbe9e24ee",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
```

```
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

RotateKey

以下範例顯示輪換 AWS KMS key 之操作的 AWS CloudTrail 記錄項目。AWS KMS 會在需要輪換 KMS 金鑰，且已啟用自動金鑰輪換時呼叫此操作。啟用自動金鑰輪換 (EnableKeyRotation) 時，AWS KMS會在 365 天後輪換 KMS 金鑰，之後每 365 天輪換一次。

CloudTrail 在 2022 年 12 月或之後記錄的此作業的記錄項目會在`responseElements.keyId`值中包含受影響 KMS 金鑰的金鑰 ARN，即使此作業未傳回金鑰 ARN。

如需記錄`EnableKeyRotation`作業的記 CloudTrail 錄項目範例，請參閱EnableKeyRotation。如需輪換 KMS 金鑰的詳細資訊，請參閱 輪換 AWS KMS keys。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "accountId": "111122223333",
        "invokedBy": "AWS Internal"
    },
    "eventTime": "2021-01-14T01:41:59Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "RotateKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": null,
    "responseElements": {
        "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "eventID": "a24b3967-ddad-417f-9b22-2332b918db06",
    "readOnly": false,
    "resources": [
        {
```

```
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsServiceEvent",
    "recipientAccountId": "111122223333",
    "serviceEventDetails": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    }
}
```

ScheduleKeyDeletion

這些範例顯示ScheduleKeyDeletion作業的AWS CloudTrail記錄項目。

如需刪除金鑰時所寫入之 CloudTrail 記錄項目的範例，請參閱DeleteKey。如需刪除 AWS KMS keys 的資訊，請參閱 刪除 AWS KMS keys。

以下範例會記錄單一區域 KMS 金鑰的 `ScheduleKeyDeletion` 請求。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
    },
    "eventTime": "2021-03-23T18:58:30Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "ScheduleKeyDeletion",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "pendingWindowInDays": 20,
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": {
```

```
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "keyState": "PendingDeletion",
        "deletionDate": "Apr 12, 2021 18:58:30 PM"
    },
    "requestID": "ee408f36-ea01-422b-ac14-b0f147c68334",
    "eventID": "3c4226b0-1e81-48a8-a333-7fa5f3cbd118",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

以下範例會記錄具有複本金鑰之多區域 KMS 金鑰的 ScheduleKeyDeletion 請求。

因為 AWS KMS 在刪除其所有複本金鑰之前不會刪除多區域金鑰，所以在 responseElements 欄位中，keyState 是 PendingReplicaDeletion 且會省略 deletionDate 欄位。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
    },
    "eventTime": "2021-10-28T17:59:05Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "ScheduleKeyDeletion",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "pendingWindowInDays": 30,
```

```
        "keyId": "mrk-1234abcd12ab34cd56ef1234567890ab"
    },
    "responseElements": {
        "keyId": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
        "keyState": "PendingReplicaDeletion",
        "pendingWindowInDays": 30
    },
    "requestID": "12341411-d846-42a6-a476-b1cbe3011f89",
    "eventID": "abcda5f-396d-494c-9380-0c47860df5f1",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

以下範例會記錄 AWS CloudHSM 自訂金鑰存放區中之 KMS 金鑰的 `ScheduleKeyDeletion` 請求。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
    },
    "eventTime": "2021-10-26T23:25:25Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "ScheduleKeyDeletion",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
```

```
    "requestParameters": {
        "keyId": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321",
        "pendingWindowInDays": 30
    },
    "responseElements": {
        "keyId": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321",
        "deletionDate": "Nov 2, 2021, 11:25:25 PM",
        "keyState": "PendingDeletion",
        "pendingWindowInDays": 30
    },
    "additionalEventData": {
        "customKeyStoreId": "cks-1234567890abcdef0",
        "clusterId": "cluster-1a23b4cdefg",
        "backingKeys": "[{\"keyHandle\":\"01\",\"backingKeyId\":\"backing-key-id\"}]"
    },
    "requestID": "abcd9f60-2c9c-4a0b-a456-d5d998f7f321",
    "eventID": "ca01996a-01b0-4edd-bbbb-25d7b6d1a6fa",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

符號

這些範例顯示 Sign 操作的 AWS CloudTrail 的日誌項目。

下列範例顯示 S ign 作業的 CloudTrail 記錄項目，該作業使用非對稱 RSA KMS 金鑰產生檔案的數位簽章。

```
{
    "eventVersion": "1.08",
```

```
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-03-07T22:36:44Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Sign",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "messageType": "RAW",
        "keyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
        "signingAlgorithm": "RSASSA_PKCS1_V1_5_SHA_256"
    },
    "responseElements": null,
    "requestID": "8d0b35e0-46cf-48b9-be99-bf2ebc9ab9fb",
    "eventID": "107b3cac-b125-4556-9702-12a2b9afcff7",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```


SynchronizeMultiRegionKey

以下範例顯示 AWS KMS 同步多區域金鑰時產生的 AWS CloudTrail 日誌項目。同步涉及將多區域主要金鑰之共用屬性複製至其複本金鑰的跨區域呼叫。AWS KMS 會定期同步多區域金鑰，以確保所有相關的多區域金鑰具有相同的金鑰資料。

CloudTrail 記錄項目的resources元素包括多區域主索引鍵 (包括其在內) 的索引鍵 ARN。AWS 區域相關的多區域複本金鑰及其區域不會列在此日誌項目中。

CloudTrail 在 2022 年 12 月或之後記錄的此作業的記錄項目會在responseElements.keyId值中包含受影響 KMS 金鑰的金鑰 ARN，即使此作業不會傳回金鑰 ARN。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "accountId": "111122223333",
        "invokedBy": "AWS Internal"
    },
    "eventTime": "2020-11-18T02:04:37Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "SynchronizeMultiRegionKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": null,
    "responseElements": {
        "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "requestID": "12345681-de97-42e9-bed0-b02ae1abd8dc",
    "eventID": "abcdec99-2b5c-4670-9521-ddb8f031e146",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

TagResource

下列範例顯示呼叫的AWS CloudTrail記錄項目，該TagResource作業可新增標籤的標籤，其標籤鍵為，Department且標籤值為IT。

如需旋轉金鑰時寫入的UntagResource CloudTrail 記錄項目範例，請參閱UntagResource。如需標記 AWS KMS keys 的詳細資訊，請參閱 標記金鑰。


```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2020-07-01T21:19:25Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "TagResource",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "tags": [
            {
                "tagKey": "Department",
                "tagValue": "IT"
            }
        ]
    },
    "responseElements": null,
    "requestID": "b942584a-f77d-4787-9feb-b9c5be6e746d",
    "eventID": "0a091b9b-0df5-4cf9-b667-6f2879532b8f",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
```

```
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

UntagResource

下列範例顯示對刪除標籤鍵為之標籤之UntagResource作業的呼叫AWS CloudTrail記錄項目Dept。

CloudTrail 在 2022 年 12 月或之後記錄的此作業的記錄項目會在responseElements.keyId值中包含受影響 KMS 金鑰的金鑰 ARN，即使此作業不會傳回金鑰 ARN。

如需記TagResource CloudTrail 錄項目的範例，請參閱TagResource。如需標記 AWS KMS keys 的詳細資訊，請參閱 標記金鑰。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2020-07-01T21:19:19Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "UntagResource",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "tagKeys": [
            "Dept"
        ]
    },
    "responseElements": {
```

```
        "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "requestID": "cb1d507b-6015-47f4-812b-179713af8068",
    "eventID": "0b00f4b0-036e-411d-aa75-87eb4a35a4b3",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

UpdateAlias

下列範例顯示作UpdateAlias業的AWS CloudTrail記錄項目。`resources` 元素包含別名和 KMS 金鑰資源的欄位。如需有關在 AWS KMS 中建立別名的資訊，請參閱 建立別名。

CloudTrail 在 2022 年 12 月或之後記錄的此作業的記錄項目會在`responseElements.keyId`值中包含受影響 KMS 金鑰的金鑰 ARN，即使此作業未傳回金鑰 ARN。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
         "principalId": "EX_PRINCIPAL_ID",
         "arn": "arn:aws:iam::111122223333:user/Alice",
         "accountId": "111122223333",
         "accessKeyId": "EXAMPLE_KEY_ID",
         "userName": "Alice"
    },
    "eventTime": "2020-11-13T23:18:15Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "UpdateAlias",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
```

```
        "aliasName": "alias/my_alias",
        "targetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": {
        "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "requestID": "d9472f40-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "f72d3993-864f-48d6-8f16-e26e1ae8dff0",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-west-2:111122223333:alias/my_alias"
        },
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

UpdateCustomKeyStore

以下範例會顯示藉由呼叫 `UpdateCustomKeyStore` 操作產生的 AWS CloudTrail 日誌項目，以更新自訂金鑰存放區的叢集 ID。如需編輯自訂金鑰存放區的詳細資訊，請參閱 編輯 AWS CloudHSM 金鑰存放區設定。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
```

```
    "eventTime": "2021-10-21T20:17:32Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "UpdateCustomKeyStore",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "customKeyStoreId": "cks-1234567890abcdef0",
        "clusterId": "cluster-1a23b4cdefg"
    },
    "responseElements": null,
    "additionalEventData": {
        "customKeyStoreName": "ExampleKeyStore",
        "clusterId": "cluster-1a23b4cdefg"
    },
    "requestID": "abcde9e1-f1a3-4460-a423-577fb6e695c9",
    "eventID": "114b61b9-0ea6-47f5-a9d2-4f2bdd0017d5",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333"
}
```


UpdateKeyDescription

以下範例顯示藉由呼叫 AWS CloudTrail 操作而產生的 UpdateKeyDescription 日誌項目。


```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-09-01T19:22:40Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "UpdateKeyDescription",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
```

```
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "description": "New key description"
    },
    "responseElements": null,
    "requestID": "8c3c1f8b-336d-4896-b034-4eb9916bc9b3",
    "eventID": "f5f3d548-2e9e-4658-8427-9dcb5b1ea791",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

UpdatePrimaryRegion

下列範例顯示AWS CloudTrail透過呼叫多區域金鑰UpdatePrimaryRegion作業所產生的記錄項目。

此UpdatePrimaryRegion作業會寫入兩個 CloudTrail 記錄項目：一個在 [區域] 中具有轉換為複本金鑰的多區域主索引鍵，另一個在區域中具有轉換為主索引鍵的多區域複本金鑰。

CloudTrail 在 2022 年 12 月或之後記錄的此作業的記錄項目會在`responseElements.keyId`值中包含受影響 KMS 金鑰的金鑰 ARN，即使此作業不會傳回金鑰 ARN。

下列範例顯示多區域金鑰從主索引鍵變更為複本金鑰 (us-west-2) 的區域UpdatePrimaryRegion中的 CloudTrail 記錄項目。`primaryRegion` 欄位顯示現在託管主要金鑰的區域 (ap-northeast-1)。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
```

```
    },
    "eventTime": "2021-03-10T20:23:37Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "UpdatePrimaryRegion",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "mrk-1234abcd12ab34cd56ef1234567890ab",
        "primaryRegion": "ap-northeast-1"
    },
    "responseElements": {
        "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "requestID": "ee408f36-ea01-422b-ac14-b0f147c68334",
    "eventID": "3c4226b0-1e81-48a8-a333-7fa5f3cbd118",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333"
}
```

下列範例代表多區域索引鍵從複本索引鍵變更為主索引鍵 (ap-northeast-1) 的區域`UpdatePrimaryRegion`中的 CloudTrail 記錄項目。此日誌項目無法識別先前的主要區域。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
```

```
            "userName": "Alice",
            "invokedBy": "kms.amazonaws.com"
    },
    "eventTime": "2021-03-10T20:23:37Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "UpdatePrimaryRegion",
    "awsRegion": "ap-northeast-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "arn:aws:kms:ap-northeast-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
        "primaryRegion": "ap-northeast-1"
    },
    "responseElements": {
        "keyId":"arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "requestID": "ee408f36-ea01-422b-ac14-b0f147c68334",
    "eventID": "091e6be5-737f-43c6-8431-e3679d6d0619",
    "readOnly": false,
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "eventCategory": "Management",
    "recipientAccountId": "111122223333"
}
```

VerifyMac

下列範例顯示作VerifyMac業的AWS CloudTrail記錄項目。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
     },
    "eventTime": "2022-03-31T19:25:54Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "VerifyMac",
```

```
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "macAlgorithm": "HMAC_SHA_384",
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": null,
    "requestID": "f35da560-edff-4d6e-9b40-fb306fa9ef1e",
    "eventID": "6b464487-6dea-44cd-84ad-225d7450c975",
    "readOnly": true,
    "resources": [
        {
           "accountId": "111122223333",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

確認

這些範例顯示 Verify 操作的 AWS CloudTrail 的日誌項目。

下列範例顯示使用非對稱 RSA KMS 金鑰來驗證數位簽章的「驗證」作業的 CloudTrail 記錄項目。

```
{
    "eventVersion": "1.08",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2022-03-07T22:50:41Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Verify",
    "awsRegion": "us-west-2",
```

```
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "signingAlgorithm": "RSASSA_PKCS1_V1_5_SHA_256",
        "keyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
        "messageType": "RAW"
    },
    "responseElements": null,
    "requestID": "c73ab82a-af82-4750-ae2c-b6bb790e9c28",
    "eventID": "3b4331cd-5b7b-4de5-bf5f-82ec22f0dac0",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321"
        }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
}
```

Amazon EC2 範例 1

以下範例記錄 IAM 主體在 Amazon EC2 管理主控台中使用預設磁碟區金鑰來建立加密的磁碟區。

下列範例顯示使用者 Alice 在 Amazon EC2 管理主控台中使用預設磁碟區金鑰建立加密磁碟區的 CloudTrail 記錄項目。EC2 日誌檔記錄包含 volumeId 欄位，其值為 "vol-13439757"。AWS KMS 記錄包含一個 encryptionContext 欄位，其值為 "aws:ebs:id": "vol-13439757"。同樣地，兩個記錄之間的 principalId 和 accountId 相符。記錄反映建立加密的磁碟區會產生用來加密磁碟區內容的資料金鑰。

```
{
  "Records": [
     {
      "eventVersion": "1.02",
      "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
```

```
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
        },
      "eventTime": "2014-11-05T20:50:18Z",
      "eventSource": "ec2.amazonaws.com",
      "eventName": "CreateVolume",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "AWS Internal",
      "requestParameters": {
        "size": "10",
        "zone": "us-east-1a",
        "volumeType": "gp2",
        "encrypted": true
      },
      "responseElements": {
        "volumeId": "vol-13439757",
        "size": "10",
        "zone": "us-east-1a",
        "status": "creating",
        "createTime": 1415220618876,
        "volumeType": "gp2",
        "iops": 30,
        "encrypted": true
      },
      "requestID": "1565210e-73d0-4912-854c-b15ed349e526",
      "eventID": "a3447186-135f-4b00-8424-bc41f1a93b4f",
      "eventType": "AwsApiCall",
      "recipientAccountId": "123456789012"
    },
    {
      "eventVersion": "1.02",
      "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
      },
      "eventTime": "2014-11-05T20:50:19Z",
      "eventSource": "kms.amazonaws.com",
      "eventName": "GenerateDataKeyWithoutPlaintext",
```

```
      "awsRegion": "us-east-1",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "&AWS; Internal",
      "requestParameters": {
        "encryptionContext": {
          "aws:ebs:id": "vol-13439757"
        },
        "numberOfBytes": 64,
        "keyId": "alias/aws/ebs"
      },
      "responseElements": null,
      "requestID": "create-123456789012-758241111-1415220618",
      "eventID": "4bd2a696-d833-48cc-b72c-05e61b608399",
      "readOnly": true,
      "resources": [
        {
          "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
          "accountId": "111122223333"
        }
      ],
      "eventType": "AwsApiCall",
      "recipientAccountId": "111122223333"
    }
  ]
}
```

Amazon EC2 範例 2

在下列範例中，執行 Amazon EC2 執行個體的 IAM 主體會建立並掛載在 KMS 金鑰下加密的資料磁碟區。此動作會產生多個記錄 CloudTrail 檔記錄。

建立磁碟區時，Amazon EC2 會代表客戶，從 AWS KMS (`GenerateDataKeyWithoutPlaintext`) 取得加密的資料金鑰。然後它建立一個授予權限 (`CreateGrant`)，允許它解密資料密鑰。掛載磁碟區時，Amazon EC2 呼叫 AWS KMS 來解密資料金鑰 (`Decrypt`)。

Amazon EC2 執行個體的 `instanceId` (`"i-81e2f56c"`) 會顯示在 `RunInstances` 事件中。相同執行個體 ID 會使 `granteePrincipal` 授權限定所建立的授權 (`"111122223333:aws:ec2-infrastructure:i-81e2f56c"`) 和假設角色是 `Decrypt` 呼叫中的委託人 (`"arn:aws:sts::111122223333:assumed-role/aws:ec2-infrastructure/i-81e2f56c"`)。

保護資料磁碟區的 KMS 金鑰的金鑰 ARN `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab`，會出現在所有三個 AWS KMS 呼叫 (`CreateGrant`、`GenerateDataKeyWithoutPlaintext` 和 `Decrypt`) 中。


```
{
  "Records": [
    {
      "eventVersion": "1.02",
      "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
      },
      "eventTime": "2014-11-05T21:35:27Z",
      "eventSource": "ec2.amazonaws.com",
      "eventName": "RunInstances",
      "awsRegion": "us-west-2",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "AWS Internal",
      "requestParameters": {
        "instancesSet": {
          "items": [
            {
              "imageId": "ami-b66ed3de",
              "minCount": 1,
              "maxCount": 1
            }
          ]
        },
        "groupSet": {
          "items": [
            {
              "groupId": "sg-98b6e0f2"
            }
          ]
        },
        "instanceType": "m3.medium",
        "blockDeviceMapping": {
          "items": [
            {
```

```
            "deviceName": "/dev/xvda",
            "ebs": {
              "volumeSize": 8,
              "deleteOnTermination": true,
              "volumeType": "gp2"
            }
          },
          {
            "deviceName": "/dev/sdb",
            "ebs": {
              "volumeSize": 8,
              "deleteOnTermination": false,
              "volumeType": "gp2",
              "encrypted": true
            }
          }
        ]
      },
      "monitoring": {
        "enabled": false
      },
      "disableApiTermination": false,
      "instanceInitiatedShutdownBehavior": "stop",
      "clientToken": "XdKUT141516171819",
      "ebsOptimized": false
    },
    "responseElements": {
      "reservationId": "r-5ebc9f74",
      "ownerId": "111122223333",
      "groupSet": {
        "items": [
          {
            "groupId": "sg-98b6e0f2",
            "groupName": "launch-wizard-2"
          }
        ]
      },
      "instancesSet": {
        "items": [
          {
            "instanceId": "i-81e2f56c",
            "imageId": "ami-b66ed3de",
            "instanceState": {
              "code": 0,
```

```
            "name": "pending"
          },
          "amiLaunchIndex": 0,
          "productCodes": {

          },
          "instanceType": "m3.medium",
          "launchTime": 1415223328000,
          "placement": {
            "availabilityZone": "us-east-1a",
            "tenancy": "default"
          },
          "monitoring": {
            "state": "disabled"
          },
          "stateReason": {
            "code": "pending",
            "message": "pending"
          },
          "architecture": "x86_64",
          "rootDeviceType": "ebs",
          "rootDeviceName": "/dev/xvda",
          "blockDeviceMapping": {

          },
          "virtualizationType": "hvm",
          "hypervisor": "xen",
          "clientToken": "XdKUT1415223327917",
          "groupSet": {
            "items": [
              {
                "groupId": "sg-98b6e0f2",
                "groupName": "launch-wizard-2"
              }
            ]
          },
          "networkInterfaceSet": {

          },
          "ebsOptimized": false
        }
      ]
    }
  },
```

```
      "requestID": "41c4b4f7-8bce-4773-bf0e-5ae3bb5cbce2",
      "eventID": "cd75a605-2fee-4fda-b847-9c3d330ebaae",
      "eventType": "AwsApiCall",
      "recipientAccountId": "111122223333"
    },
    {
      "eventVersion": "1.02",
      "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
      },
      "eventTime": "2014-11-05T21:35:35Z",
      "eventSource": "kms.amazonaws.com",
      "eventName": "CreateGrant",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "AWS Internal",
      "requestParameters": {
        "constraints": {
          "encryptionContextSubset": {
            "aws:ebs:id": "vol-f67bafb2"
          }
        },
        "granteePrincipal": "111122223333:aws:ec2-infrastructure:i-81e2f56c",
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
      },
      "responseElements": {
        "grantId": "abcde1237f76e4ba7987489ac329fbfba6ad343d6f7075dbd1ef191f0120514a"
      },
      "requestID": "41c4b4f7-8bce-4773-bf0e-5ae3bb5cbce2",
      "eventID": "c1ad79e3-0d3f-402a-b119-d5c31d7c6a6c",
      "readOnly": false,
      "resources": [
        {
          "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
          "accountId": "111122223333"
        }
      ],
```

```
      "eventType": "AwsApiCall",
      "recipientAccountId": "111122223333"
    },
    {
      "eventVersion": "1.02",
      "userIdentity": {
          "type": "IAMUser",
          "principalId": "EX_PRINCIPAL_ID",
          "arn": "arn:aws:iam::111122223333:user/Alice",
          "accountId": "111122223333",
          "accessKeyId": "EXAMPLE_KEY_ID",
          "userName": "Alice"
      },
      "eventTime": "2014-11-05T21:35:32Z",
      "eventSource": "kms.amazonaws.com",
      "eventName": "GenerateDataKeyWithoutPlaintext",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "192.0.2.0",
      "userAgent": "AWS Internal",
      "requestParameters": {
        "encryptionContext": {
          "aws:ebs:id": "vol-f67bafb2"
        },
        "numberOfBytes": 64,
        "keyId": "alias/aws/ebs"
      },
      "responseElements": null,
      "requestID": "create-111122223333-758247346-1415223332",
      "eventID": "ac3cab10-ce93-4953-9d62-0b6e5cba651d",
      "readOnly": true,
      "resources": [
        {
          "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
          "accountId": "111122223333"
        }
      ],
      "eventType": "AwsApiCall",
      "recipientAccountId": "111122223333"
    },
    {
      "eventVersion": "1.02",
      "userIdentity": {
        "type": "AssumedRole",
```

```
        "principalId": "111122223333:aws:ec2-infrastructure:i-81e2f56c",
        "arn": "arn:aws:sts::111122223333:assumed-role/aws:ec2-infrastructure/
i-81e2f56c",
        "accountId": "111122223333",
        "accessKeyId": "",
        "sessionContext": {
          "attributes": {
            "mfaAuthenticated": "false",
            "creationDate": "2014-11-05T21:35:38Z"
          },
          "sessionIssuer": {
            "type": "Role",
            "principalId": "111122223333:aws:ec2-infrastructure",
            "arn": "arn:aws:iam::111122223333:role/aws:ec2-infrastructure",
            "accountId": "111122223333",
            "userName": "aws:ec2-infrastructure"
          }
        }
      },
      "eventTime": "2014-11-05T21:35:47Z",
      "eventSource": "kms.amazonaws.com",
      "eventName": "Decrypt",
      "awsRegion": "us-east-1",
      "sourceIPAddress": "192.0.2.0",
      "requestParameters": {
        "encryptionContext": {
          "aws:ebs:id": "vol-f67bafb2"
        }
      },
      "responseElements": null,
      "requestID": "b4b27883-6533-11e4-b4d9-751f1761e9e5",
      "eventID": "edb65380-0a3e-4123-bbc8-3d1b7cff49b0",
      "readOnly": true,
      "resources": [
        {
          "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
          "accountId": "111122223333"
        }
      ],
      "eventType": "AwsApiCall",
      "recipientAccountId": "111122223333"
    }
  ]
```

```
}
```

使用 Amazon 監控 CloudWatch

您可以AWS KMS keys使用 Amazon 監控您的使用情況 CloudWatch，這是一項AWS服務，可將原始資料收集並處理AWS KMS成可讀且近乎即時的指標。這些資料會保留 2 週，因此您可以存取歷史資訊，更加了解您 KMS 金鑰的使用情形及其隨時間的變更。

您可以使用 Amazon CloudWatch 提醒您重要事件，例如以下事件。

- KMS 金鑰中匯入的金鑰材料接近其過期日期。
- 仍然會使用等待刪除的 KMS 金鑰。
- KMS 金鑰中的金鑰材料會自動輪換。
- 已刪除 KMS 金鑰。

您也可以建立 Amazon CloudWatch 警示，以便在請求率達到配額值的特定百分比時提醒您。如需詳細資訊，請參閱AWS安全部落格 CloudWatch中的使用 Service Quotas 和 Amazon 管理 AWS KMS API 請求費率。

主題

- AWS KMS 指標與維度
- 檢視 AWS KMS 指標
- 建立 CloudWatch 警示以監控 KMS 金鑰

AWS KMS 指標與維度

AWS KMS預先定義 Amazon CloudWatch 指標，讓您更輕鬆地監控重要資料和建立警示。您可以使用 AWS Management Console和 Amazon CloudWatch API 查看AWS KMS指標。

本節列出每個AWS KMS量度和每個量度的維度，並提供一些根據這些量度和維度建立 CloudWatch 警示的基本指引。

> **Note**
>
> 維度群組名稱：

若要在 Amazon CloudWatch 主控台中檢視指標，請在「指標」區段中選取維度群組名稱。然後，您可以按 Metric name (指標名稱) 進行篩選。本主題包括每個 AWS KMS 指標的指標名稱和維度群組名稱。

主題

- SecondsUntilKeyMaterialExpiration
- ExternalKeyStoreThrottle
- XksProxyCertificateDaysToExpire
- XksProxyCredentialAge
- XksProxyErrors
- XksExternalKeyManagerStates
- XksProxyLatency

SecondsUntilKeyMaterialExpiration

KMS 金鑰中匯入金鑰材料過期之前所剩的秒數。此指標僅對具有匯入金鑰材料 (EXTERNAL 的金鑰材料來源) 和到期日的 KMS 金鑰有效。

使用此指標可追蹤匯入金鑰資料過期之前所剩的時間。當該時間低於您定義的閾值時，您可能會重新匯入具有新到期日期的金鑰材料。該 `SecondsUntilKeyMaterialExpiration` 指標專用於 KMS 金鑰。您無法使用此指標來監控多個 KMS 金鑰或您可能在將來建立的 KMS 金鑰。如需建立 CloudWatch 警示以監視此指標的說明，請參閱建立匯入金鑰材料到期的 CloudWatch 警示。

此指標最實用的統計數字是 `Minimum`，其會告訴您指定統計週期內所有資料點之剩餘時間的最小數量。此指標的唯一有效單位是 `Seconds`。

維度群組名稱：Per-Key Metrics (每個金鑰指標)

`SecondsUntilKeyMaterialExpiration` 的維度

維度	描述；與 AWS 相關
KeyId	每個 KMS 金鑰的值。

ExternalKeyStoreThrottle

AWS KMS 限流 (以 ThrottlingException 回應) 的每個外部金鑰存放區中 KMS 金鑰的密碼編譯操作請求數目。此指標只適用於外部金鑰存放區。

此指ExternalKeyStoreThrottle標僅適用於外部金鑰存放區中的 KMS 金鑰，而且僅適用於密碼編譯作業和DescribeKey作業的要求。 AWS KMS當要求率超過外部金鑰存放區的自訂金鑰存放區要求配額時，會調節這些要求。此指標不包括外部金鑰存放區代理或外部金鑰管理器的限流。

使用此指標可檢閱和調整自訂金鑰存放區請求配額的值。如果此指標指示 AWS KMS 經常對這些 KMS 金鑰的請求進行限流，您可能會考慮請求增加自訂金鑰存放區請求配額值。如需相關說明，請參閱《Service Quotas 使用者指南》中的請求提升配額。

如果您經常收到 KMSInvalidStateException 錯誤，並且訊息說明「因為請求率非常高」而拒絕請求，或「因為外部金鑰存放區代理未及時回應」而拒絕請求，則可能表示您的外部金鑰管理器或外部金鑰存放區代理無法跟上目前的請求率。如果可能，請降低您的請求率。您也可以考慮請求減少自訂金鑰存放區請求配額值。減少此配額值可能會增加限流 (以及 ExternalKeyStoreThrottle 指標值)，但表示在傳送至外部金鑰存放區代理或外部金鑰管理器之前，AWS KMS 會快速拒絕多餘的請求。若要請求減少配額，請造訪 AWS Support 中心並建立案例。

維度群組名稱：Keystore Throttle Metrics (金鑰存放區限流指標)

維度	描述
CustomKeyStoreId	每個外部金鑰存放區的值。
KmsOperation	每個 AWS KMS API 操作的值。此指標僅適用於密碼編譯操作和外部金鑰存放區中 KMS 金鑰的 DescribeKey 操作。
KeySpec	每個 KMS 金鑰類型的值。外部金鑰存放區中 KMS 金鑰的唯一受支援的金鑰規格為 SYMMETRIC_DEFAULT。

XksProxyCertificateDaysToExpire

外部金鑰存放區代理端點 (XksProxyUriEndpoint) 的 TLS 憑證到期前的天數。此指標只適用於外部金鑰存放區。

使用此指標建立 CloudWatch 警示，通知您 TLS 憑證即將到期。憑證到期時，AWS KMS 無法與外部金鑰存放區代理通訊。在您續約憑證之前，外部金鑰存放區中受 KMS 金鑰保護的所有資料都無法存取。

憑證警示可防止憑證過期，這可能使您無法存取已加密的資源。設定警示可讓組織有時間在憑證過期之前續約憑證。

維度群組名稱：XKS Proxy Certificate Metrics (XKS 代理憑證指標)

維度	描述
CustomKeyStoreId	每個外部金鑰存放區的值。
CertificateName	TLS 憑證中的主體名稱 (CN)。

XksProxyCredentialAge

自當前外部金鑰存放區代理身分驗證憑證 (`XksProxyAuthenticationCredential`) 與外部金鑰存放區相關聯之後的天數。當您在建立或更新外部金鑰存放區時輸入身分驗證憑證，此計數便會開始。此指標只適用於外部金鑰存放區。

此值旨在提醒您身分驗證憑證的有效期。但是，由於我們會在您將憑證與外部金鑰存放區建立關聯時開始計數，而不是在外部金鑰存放區代理上建立身分驗證憑證時，因此這可能不是代理上憑證有效期的準確指標。

使用此度量建立 CloudWatch 警示，提醒您輪換外部金鑰存放區 Proxy 驗證認證。

維度群組名稱：Per-Keystore Metrics (每個金鑰存放區指標)

維度	描述
CustomKeyStoreId	每個外部金鑰存放區的值。

XksProxyErrors

與外部金鑰存放區代理的 AWS KMS 請求相關的例外狀況數。此計數包括外部金鑰存放區代理傳回給 AWS KMS 的例外狀況，以及當外部金鑰存放區代理未在 250 毫秒逾時間隔內回應 AWS KMS 時發生的逾時錯誤。此指標只適用於外部金鑰存放區。

使用此指標可追蹤外部金鑰存放區中 KMS 金鑰的錯誤率。它顯示了最常見的錯誤，因此您可以優先考慮工程工作。例如，產生高速率不可重試錯誤的 KMS 金鑰可能表示外部金鑰存放區的組態有問題。若要檢視外部金鑰存放區組態，請參閱 檢視外部金鑰存放區。若要編輯外部金鑰存放區設定，請參閱 編輯外部金鑰存放區屬性。

維度群組名稱：XKS Proxy Error Metrics (XKS 代理錯誤指標)

維度	描述
CustomKeyStoreId	每個外部金鑰存放區的值。
KmsOperation	對 XKS 代理產生請求的每個 AWS KMS API 操作的值。
XksOperation	每個外部金鑰存放區代理 API 操作的值。
KeySpec	每個 KMS 金鑰類型的值。外部金鑰存放區中 KMS 金鑰的唯一受支援的金鑰規格為 SYMMETRIC_DEFAULT。
ErrorType	數值： • 可重試的錯誤：可能是暫時性的，例如網路錯誤。 • 不可重試的錯誤：可能表示自訂金鑰存放區組態或外部元件有問題。 • N/A：請求成功；沒有錯誤
ExceptionName	數值： • 例外狀況的名稱 • 無：請求成功；沒有錯誤

XksExternalKeyManagerStates

下列每個運作狀態中的外部金鑰管理器執行個體的數目計數：`Active`、`Degraded` 和 `Unavailable`。此指標的資訊來自與每個外部金鑰存放區關聯的外部金鑰存放區代理。此指標只適用於外部金鑰存放區。

以下是與外部金鑰存放區相關聯之外部金鑰管理器執行個體的運作狀態。每個外部金鑰存放區代理可能會使用不同的指標來測量外部金鑰管理器的運作狀態。如需詳細資訊，請參閱外部金鑰存放區代理的文件。

- `Active`：外部金鑰管理器運作正常。
- `Degraded`：外部金鑰管理器運作不正常，但仍可提供流量
- `Unavailable`：外部金鑰管理器無法提供流量。

使用此測量結果可建立警示， CloudWatch 警示您降級和無法使用的外部金鑰管理員執行個體。若要確定處於每個狀態的外部金鑰管理器執行個體，請參閱外部金鑰存放區代理日誌。

維度群組名稱：XKS External Key Manager Metrics (XKS 外部金鑰管理器指標)

維度	描述
CustomKeyStoreId	每個外部金鑰存放區的值。
XksExternalKeyManagerState	每個運作狀態的值。

XksProxyLatency

外部金鑰存放區代理回應 AWS KMS 請求所需的毫秒數。如果請求逾時，則記錄的值為 250 毫秒逾時限制。此指標只適用於外部金鑰存放區。

使用此指標可評估外部金鑰存放區代理和外部金鑰管理器的效能。例如，如果代理在加密和解密操作中經常逾時，請咨詢您的外部代理管理員。

回應緩慢也可能表示您的外部金鑰管理器無法處理當前的請求流量。AWS KMS 建議您的外部金鑰管理器每秒最多可處理 1800 個密碼編譯操作請求。如果您的外部金鑰管理器無法處理每秒 1800 個請求，請考慮請求減少自訂金鑰存放區中 KMS 金鑰的請求配額。使用外部金鑰存放區中的 KMS 金鑰進行密

碼編譯操作的請求會快速檢錯，並發生限流例外狀況，而不是被外部金鑰存放區代理或外部金鑰管理器處理並拒絕。

維度群組名稱：XKS Proxy Latency Metrics (XKS 代理延遲指標)

維度	描述
CustomKeyStoreId	每個外部金鑰存放區的值。
KmsOperation	對 XKS 代理產生請求的每個 AWS KMS API 操作的值。
XksOperation	每個外部金鑰存放區代理 API 操作的值。
KeySpec	每個 KMS 金鑰類型的值。外部金鑰存放區中 KMS 金鑰的唯一受支援的金鑰規格為 SYMMETRIC_DEFAULT。

檢視 AWS KMS 指標

您可以使用AWS Management Console和 Amazon CloudWatch API 查看AWS KMS指標。

使用 CloudWatch 主控台檢視指標

1. 請在以下位置開啟 CloudWatch 主控台。 https://console.aws.amazon.com/cloudwatch/
2. 如有必要請變更區域。請在導覽列中選擇您的 AWS 資源所在的區域。
3. 在導覽窗格中，選擇 Metrics (指標)、All metrics (所有指標)。
4. 在 Browse (瀏覽) 索引標籤上，搜尋 KMS，然後選擇 KMS。
5. 選擇要檢視的指標之維度群組名稱。

   例如，若為 SecondsUntilKeyMaterialExpiration 指標，請選擇 Per-Key Metrics (每個金鑰指標)。
6. 如需指標值的圖表，請選擇指標名稱，然後選擇 Add to graph。若要將折線圖轉換為值，請選擇 Line (線條)，然後選擇 Number (數字)。

若要使用 Amazon CloudWatch API 檢視指標

若要使用 AWS KMS CloudWatch API 檢視指標，請傳送Namespace設定為的ListMetrics要求AWS/KMS。以下範例顯示如何使用 AWS Command Line Interface (AWS CLI) 執行此作業。

```
$  aws cloudwatch list-metrics --namespace AWS/KMS

{
    "Metrics": [
        {
            "Namespace": "AWS/KMS",
            "MetricName": "SecondsUntilKeyMaterialExpiration",
            "Dimensions": [
                {
                    "Name": "KeyId",
                    "Value": "1234abcd-12ab-34cd-56ef-1234567890ab"
                }
            ]
        },
        {
            "Namespace": "AWS/KMS",
            "MetricName": "ExtenalKeyStoreThrottle",
            "Dimensions": [
                {
                    "Name": "CustomKeyStoreId",
                    "Value": "cks-1234567890abcdef0"
                },
                {
                    "Name": "KmsOperation",
                    "Value": "Encrypt"
                },
                {
                    "Name": "KeySpec",
                    "Value": "SYMMETRIC_DEFAULT"
                }
            ]
        },
        {
            "Namespace": "AWS/KMS",
            "MetricName": "XksProxyCertificateDaysToExpire",
            "Dimensions": [
                {
                    "Name": "CustomKeyStoreId",
                    "Value": "cks-1234567890abcdef0"
                },
```

```
                {
                    "Name": "CertificateName",
                    "Value": "myproxy.xks.example.com"
                }
            ]
        },
        {
            "Namespace": "AWS/KMS",
            "MetricName": "XksProxyCredentialAge",
            "Dimensions": [
                {
                    "Name": "CustomKeyStoreId",
                    "Value": "cks-1234567890abcdef0"
                }
            ]
        },
        {
            "Namespace": "AWS/KMS",
            "MetricName": "XksProxyErrors",
            "Dimensions": [
                {
                    "Name": "CustomKeyStoreId",
                    "Value": "cks-1234567890abcdef0"
                },
                {
                    "Name": "KmsOperation",
                    "Value": "Decrypt"
                },
                {
                    "Name": "XksOperation",
                    "Value": "Decrypt"
                },
                {
                     "Name": "KeySpec",
                     "Value": "SYMMETRIC_DEFAULT"
                },
                {
                    "Name": "ErrorType",
                    "Value": "Retryable errors"
                },
                {
                     "Name": "ExceptionName",
                     "Value": "KMSInvalidStateException"
                }
```

```
            ]
        },
        {
            "Namespace": "AWS/KMS",
            "MetricName": "XksProxyHsmStates",
            "Dimensions": [
                {
                    "Name": "CustomKeyStoreId",
                    "Value": "cks-1234567890abcdef0"
                },
                {
                    "Name": "XksProxyHsmState",
                    "Value": "Active"
                }
            ]
        },
        {
            "Namespace": "AWS/KMS",
            "MetricName": "XksProxyLatency",
            "Dimensions": [
                {
                    "Name": "CustomKeyStoreId",
                    "Value": "cks-1234567890abcdef0"
                },
                {
                    "Name": "KmsOperation",
                    "Value": "Decrypt"
                },
                {
                    "Name": "XksOperation",
                    "Value": "Decrypt"
                },
                {
                     "Name": "KeySpec",
                     "Value": "SYMMETRIC_DEFAULT"
                }
            ]
        }
    ]
}
```

建立 CloudWatch 警示以監控 KMS 金鑰

您可以根據AWS KMS指標建立 Amazon CloudWatch 警示。當指標值超過警示組態中指定的閾值時，警示會傳送電子郵件訊息。警示可將電子郵件訊息傳送至 Amazon Simple Notification Service (Amazon SNS) 主題或 Amazon EC2 Auto Scaling 政策。如需 CloudWatch 警示的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用 Amazon CloudWatch 警示

為即將到期的匯入金鑰資料建立警示

您可以使用指`SecondsUntilKeyMaterialExpiration`標建立 CloudWatch 警示，以便在 KMS 金鑰中匯入的金鑰材料即將到期時通知您。

當您匯入金鑰資料至 KMS 金鑰時，您可以選擇指定該金鑰資料的到期日期和時間。當金鑰材料過期時，AWS KMS 會刪除金鑰材料，讓 KMS 金鑰變成不可用。若要再次使用 KMS 金鑰，您必須重新匯入金鑰資料。

如需說明，請參閱建立匯入金鑰材料到期的 CloudWatch 警示。

建立等待刪除之 KMS 金鑰的使用情形的警示

當您排程刪除 KMS 金鑰時，AWS KMS 會在刪除 KMS 金鑰之前強制執行等待期間。您可以利用等待期間來確保您現在或未來都不需要該 KMS 金鑰。您也可以設定警 CloudWatch示，以便在等待期間的人員或應用程式嘗試在密碼編譯作業中使用 KMS 金鑰時發出警告。如果您收到這類警示的通知，您可能需要取消刪除 KMS 金鑰。

如需說明，請參閱建立警示，偵測正在等待刪除之 KMS 金鑰的使用情況。

建立警示以監控外部金鑰存放區

您可以根據外部金鑰存放區和外部金鑰存放區中的 KMS 金鑰指標建立 CloudWatch 警示。

例如，我們建議您設定 CloudWatch 警示，以便在外部金鑰存放區的 TLS 憑證即將到期 (XksProxyCertificateDaysToExpire)、您的外部金鑰存放區 Proxy 報告外部金鑰管理員執行個體處於降級或無法使用狀態時通知您 (XksProxyHsmStates)。

如需說明，請參閱 監控外部金鑰存放區。

使用 Amazon 監控 EventBridge

您可以使用 Amazon EventBridge (以前稱為 Amazon CloudWatch 活動) 提醒您 KMS 金鑰生命週期中的下列重要事件。

- KMS 金鑰中的金鑰材料會自動輪換。
- KMS 金鑰中所匯入的金鑰資料過期。
- 已排程刪除的 KMS 金鑰已被刪除。

AWS KMS與 Amazon 整合， EventBridge 以通知您影響 KMS 金鑰的重要事件。每個事件都以 JSON（JavaScript對象符號）表示，並包括事件名稱，事件發生的日期和時間以及受影響的事件。您可以收集這些事件及建立規則，將事件路由到一或多個目標，例如 AWS Lambda 函數、Amazon SNS 主題、Amazon SQS 佇列、Amazon Kinesis Data Streams 中的串流，或是內建的目標。

如需 EventBridge 與其他類型的事件搭配使用的詳細資訊，包括記錄讀取/寫入 API 請求AWS CloudTrail時所發出的事件，請參閱 Amazon 使用 EventBridge 者指南。

下列主題說明AWS KMS產生的 EventBridge 事件。

KMS CMK Rotation (KMS CMK 輪換)

AWS KMS 支援對稱加密 KMS 金鑰中金鑰資料的自動輪換。對於客戶受管金鑰而言，年度金鑰資料輪換是選用功能。AWS 受管金鑰 的金鑰資料每年會自動輪換。

每當AWS KMS旋轉關鍵材料時，它會將`KMS CMK Rotation`事件發送到 EventBridge。 AWS KMS在盡最大努力的基礎上產生此事件。

以下為此事件的範例。

```
{
  "version": "0",
  "id": "6a7e8feb-b491-4cf7-a9f1-bf3703467718",
  "detail-type": "KMS CMK Rotation",
  "source": "aws.kms",
  "account": "111122223333",
  "time": "2022-08-10T16:37:50Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
  ],
  "detail": {
    "key-id": "1234abcd-12ab-34cd-56ef-1234567890ab"
  }
}
```

KMS Imported Key Material Expiration (KMS 匯入的金鑰資料過期)

當您匯入金鑰材料至 KMS 金鑰時，您可以選擇指定該金鑰材料的過期時間。當金鑰材料到期時，會AWS KMS刪除金鑰材料並將對應的`KMS Imported Key Material Expiration`事件傳送至 EventBridge。 AWS KMS在盡最大努力的基礎上產生此事件。

以下為此事件的範例。

```
{
  "version": "0",
  "id": "9da9af57-9253-4406-87cb-7cc400e43465",
  "detail-type": "KMS Imported Key Material Expiration",
  "source": "aws.kms",
  "account": "111122223333",
  "time": "2022-08-10T16:37:50Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
  ],
  "detail": {
    "key-id": "1234abcd-12ab-34cd-56ef-1234567890ab"
  }
}
```

KMS CMK Deletion (KMS CMK 的刪除)

當您排程刪除 KMS 金鑰時，AWS KMS 會在刪除 KMS 金鑰之前強制執行等待期間。等待期結束後，AWS KMS刪除 KMS 金鑰並將`KMS CMK Deletion`事件傳送至 EventBridge。 AWS KMS保證此 EventBridge 事件。由於重試，它可能會在幾秒鐘內產生多個刪除相同 KMS 金鑰的事件。

以下為此事件的範例。

```
{
  "version": "0",
  "id": "e9ce3425-7d22-412a-a699-e7a5fc3fbc9a",
  "detail-type": "KMS CMK Deletion",
  "source": "aws.kms",
  "account": "111122223333",
  "time": "2022-08-10T16:37:50Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
```

```
  ],
  "detail": {
    "key-id": "1234abcd-12ab-34cd-56ef-1234567890ab"
  }
}
```

透過 AWS CloudFormation 建立 AWS KMS 資源

AWS Key Management Service 已與 AWS CloudFormation 整合，這項服務可協助您建立 AWS 資源的模型和設定，以減少建立和管理資源和基礎設施的時間。您可以建立一個範本，描述 KMS 金鑰和別名，然後 AWS CloudFormation 會為您佈建和設定那些資源。如需AWS KMS支援的相關資訊 CloudFormation，請參閱使AWS CloudFormation用者指南中的 KMS 資源類型參考。

當您使用 AWS CloudFormation 時，您可以重複使用您的範本，重複、一致的設定您的 AWS KMS 資源。只需描述一次您的資源，即可在多個 AWS 帳戶 與區域內重複佈建相同資源。

若要佈建和設定 AWS KMS 和其他 AWS 服務的資源，您必須了解 AWS CloudFormation 範本。範本是以 JSON 或 YAML 格式化的文本檔案。而您亦可以透過這些範本的說明，了解欲在 AWS CloudFormation 堆疊中佈建的資源。如果您不熟悉 JSON 或 YAML，您可以使用 AWS CloudFormation 設計器協助您開始使用 AWS CloudFormation 範本。如需更多詳細資訊，請參閱 AWS CloudFormation 使用者指南 中的 什麼是 AWS CloudFormation 設計器？。

區域

AWS KMS CloudFormation 支援的所有區域都支援AWS CloudFormation資源。

AWS CloudFormation 範本中的 AWS KMS 資源

AWS KMS 支援以下 AWS CloudFormation 資源。

- `AWS::KMS::Key` 會建立對稱或非對稱 KMS 金鑰。您可以使用此資源來建立對稱或非對稱多區域主要 KMS 金鑰。若要建立多區域複本金鑰，請使用 `AWS::KMS::ReplicaKey` 資源。您無法使用此資源建立具有匯入金鑰材料的 KMS 金鑰或自訂金鑰存放區的 KMS 金鑰。
- `AWS::KMS::Alias` 會建立別名，並且讓它與 KMS 金鑰建立關聯。可以在範本中定義 KMS 金鑰，也可以由其他機制建立。
- `AWS::KMS::ReplicaKey` 會建立多區域複本金鑰。若要建立多區域主要金鑰，請使用 `AWS::KMS::Key` 資源。您無法使用此資源來複寫具有匯入金鑰材料的多區域金鑰。如需多區域金鑰的詳細資訊，請參閱 AWS KMS 中的多區域金鑰。

> **Important**
>
> 如果您變更了現有 KMS 金鑰上 `KeyUsage`、`KeySpec` 或 `MultiRegion` 屬性的值，則會排定刪除現有 KMS 金鑰，並用指定的值建立新的 KMS 金鑰。
> 排定刪除後，現有的 KMS 金鑰會變成無法使用。如果您沒有在 AWS CloudFormation 之外取消現有 KMS 金鑰的排定刪除，則在刪除 KMS 金鑰時，所有在現有 KMS 金鑰下加密的資料都將無法恢復。

範本建立的 KMS 金鑰是您 AWS 帳戶 中的實際資源。授權的主體可以透過範本、AWS KMS 主控台或 AWS KMS API 使用和管理範本所建立的 KMS 金鑰。當您從範本刪除 KMS 金鑰時，系統會使用您預先指定的等待期間排程刪除 KMS 金鑰。

例如，您可以使用 AWS CloudFormation 範本來建立測試 KMS 金鑰，其中包含您偏好的主要政策、金鑰規格、金鑰使用方式、別名和標籤。您可以透過測試套件執行金鑰、檢閱結果，然後使用範本來排程要刪除的測試金鑰。稍後，您可以再次執行範本，以建立具有相同屬性的測試金鑰。

或者，您可以使用 AWS CloudFormation 範本來定義符合商務規則和安全標準的特定 KMS 金鑰組態。然後，您可以在需要建立 KMS 金鑰的任何時候使用該範本。您無需擔心金鑰設定錯誤。如果您偏好的組態出現變更，則您可以使用範本來更新 KMS 金鑰。例如，範本可讓您輕鬆地以程式設計方式啟用範本定義之所有 KMS 金鑰上的自動金鑰輪換。

如需更多關於 AWS KMS 的詳細資訊 (包括範例)，請參閱《AWS CloudFormation 使用者指南》中的 KMS 資源類型參考。

進一步了解 AWS CloudFormation

如需進一步了解 AWS CloudFormation，請參閱下列資源：

- AWS CloudFormation
- AWS CloudFormation使用者指南
- AWS CloudFormation API 參考
- AWS CloudFormation 命令列介面使用者指南

刪除 AWS KMS keys

刪除 AWS KMS key 具有破壞性，且具有潛在危險。這樣會刪除金鑰資料，還有與 KMS 金鑰相關聯的所有中繼資料，而且無法復原。刪除 KMS 金鑰之後，您就再也無法解密以該 KMS 金鑰加密的資

料，這表示該資料已無法復原。(唯一的例外是多區域複本金鑰，以及具有匯入金鑰資料的非對稱金鑰與 HMAC KMS 金鑰。) 此風險對於用於加密的非對稱 KMS 金鑰至關重要，因為在不發出警告或錯誤的情況，使用者可繼續使用公開金鑰產生密文，但在私有金鑰從 AWS KMS 刪除後無法解密。

只有當您確定不再需要使用 KMS 金鑰時，才應刪除 KMS 金鑰。如果您不確定，請考慮停用 KMS 金鑰，而不是刪除。您可以重新啟用已停用的 KMS 金鑰並取消排程刪除 KMS 金鑰，但您無法復原已刪除的 KMS 金鑰。

您只能排程刪除客戶受管金鑰。您無法刪除 AWS 受管金鑰 或 AWS 擁有的金鑰。

刪除 KMS 金鑰之前，您可能想要知道有多少加密文字在該 KMS 金鑰下加密。AWS KMS 不會存放此資訊，也不會存放任何加密文字。若要取得此資訊，您必須判斷 KMS 金鑰的過去使用情形。如需協助，請前往 判斷 KMS 金鑰的過去使用情形。

除非您明確排程刪除工作且強制的等待期間過期，否則 AWS KMS 絕對不會刪除 KMS 金鑰。

然而，您可能會因以下一或多個原因而選擇刪除 KMS 金鑰：

- 為了完成不再需要的 KMS 金鑰生命週期
- 為了避免管理開銷以及維護未使用 KMS 金鑰的相關成本
- 為了降低針對 KMS 金鑰資源配額列入計算的 KMS 金鑰數量

> **Note**
>
> 如您關閉 AWS 帳戶，則 KMS 金鑰就會變為無法存取，而您不再需要支付相關費用。

AWS KMS 會在您排程刪除 KMS 金鑰和實際刪除 KMS 金鑰時在 AWS CloudTrail 日誌中記錄一條項目。

如需刪除多區域主要金鑰和複本金鑰的相關資訊，請參閱 刪除多區域金鑰。

主題

- 關於等待期
- 刪除非對稱 KMS 金鑰
- 刪除多區域金鑰
- 刪除包含匯入金鑰資料的 KMS 金鑰

- 控制對金鑰刪除的存取
- 排程和取消金鑰刪除
- 建立警示，偵測正在等待刪除之 KMS 金鑰的使用情況
- 判斷 KMS 金鑰的過去使用情形

關於等待期

由於刪除 KMS 金鑰具有破壞性且可能很危險，所以 AWS KMS 要求您設定等待期為 7 至 30 天。預設等待期間為 30 天。

不過，實際等待期可能比您排定的等待期長最多 24 小時。若要取得刪除 KMS 金鑰的實際日期和時間，請使用此DescribeKey作業。或者在 AWS KMS 主控台，KMS 金鑰的詳細資訊頁面，在 General configuration (一般組態) 區段中，請參閱 Scheduled deletion date (排定的刪除日期)。請務必注意時區。

在等待期間，KMS 金鑰狀態和金鑰狀態為 Pending deletion (待刪除)。

- 正在等待刪除的 KMS 金鑰不能用於任何密碼編譯操作。
- AWS KMS 不會輪換待刪除 KMS 金鑰的金鑰材料。

等待期結束後，AWS KMS 會刪除 KMS 金鑰、其別名和所有相關的 AWS KMS 中繼資料。

排程刪除 KMS 金鑰可能不會立即影響 KMS 金鑰所加密的資料金鑰。如需詳細資訊，請參閱 無法使用的 KMS 金鑰如何影響資料金鑰。

使用等待期間可確保您現在或未來不需要 KMS 金鑰。您可以設定 Amazon CloudWatch 警示，以在人員或應用程式在等待期間嘗試使用 KMS 金鑰時發出警告。若要復原 KMS 金鑰，您可以在等待期間結束前取消金鑰刪除。等待期間結束後，您就無法取消金鑰刪除，且 AWS KMS 會刪除 KMS 金鑰。

刪除非對稱 KMS 金鑰

獲得授權的使用者可以刪除對稱或非對稱 KMS 金鑰。兩種類型金鑰排程刪除這些 KMS 金鑰的程序都是一樣的。不過，因為非對稱 KMS 金鑰的公有金鑰可以下載並在 AWS KMS 外部使用，所以此操作會帶來巨大的額外風險，尤其是用於加密的非對稱 KMS 金鑰 (金鑰用途為 ENCRYPT_DECRYPT)。

- 當您排程刪除 KMS 金鑰時，KMS 金鑰的金鑰狀態會變更為 Pending deletion (待刪除)，而 KMS 金鑰不能用於密碼編譯操作。不過，排程刪除對 AWS KMS 外的公有金鑰沒有影響。擁有公有金鑰的

使用者可以繼續使用這些金鑰加密訊息。他們不會收到金鑰狀態變更的任何通知。除非取消刪除，否則無法解密使用公有金鑰建立的加密文字。

- 可偵測到嘗試使用待刪除 KMS 金鑰的警示、日誌和其他策略，偵測不到在 AWS KMS 外部對公有金鑰的使用。
- KMS 金鑰刪除後，所有與該 KMS 金鑰有關的 AWS KMS 動作都會失敗。不過，擁有公有金鑰的使用者可以繼續使用這些金鑰加密訊息。無法解密這些加密文字。

如果您必須刪除金鑰用法為的非對稱 KMS 金鑰ENCRYPT_DECRYPT，請使用您的 CloudTrail 記錄項目來判斷公開金鑰是否已下載並共用。如果是，請驗證公有金鑰未用在 AWS KMS 之外。然後，考慮停用 KMS 金鑰而不是刪除 KMS 金鑰。

對於含匯入金鑰資料的非對稱 KMS 金鑰來說，可減輕刪除非對稱 KMS 金鑰所造成的風險。如需詳細資訊，請參閱 刪除包含匯入金鑰資料的 KMS 金鑰。

刪除多區域金鑰

取得授權的使用者可以排程刪除多區域主要金鑰和複本金鑰。然而，AWS KMS 不會刪除具有複本金鑰的多區域主要金鑰。此外，只要其主要金鑰存在，您就可以重新建立已刪除的多區域複本金鑰。如需詳細資訊，請參閱 刪除多區域金鑰。

刪除包含匯入金鑰資料的 KMS 金鑰

授權使用者可排程刪除包含匯入金鑰資料的 KMS 金鑰。此動作會永久刪除 KMS 金鑰、其金鑰資料，還有所有關聯 KMS 金鑰的中繼資料。

您無法建立新對稱加密 KMS 金鑰，來針對包含匯入金鑰資料且已刪除的對稱加密金鑰解密其密文，即使您擁有相同金鑰資料也一樣。不過，如您有金鑰資料，您可有效重新建立包含匯入金鑰資料的非對稱 KMS 金鑰或 HMAC KMS 金鑰。如需詳細資訊，請參閱 刪除包含匯入金鑰資料的 KMS 金鑰。

控制對金鑰刪除的存取

如果您使用 IAM 政策允許 AWS KMS 許可，則具有 AWS 管理員存取權限 (`"Action": "*"`) 或 AWS KMS 完整存取權 (`"Action": "kms:*"`) 的 IAM 身分都可以排程和取消 KMS 金鑰的金鑰刪除。若要允許金鑰管理員排程和取消金鑰政策中的金鑰刪除，請使用 AWS KMS 主控台或 AWS KMS API。

通常，只有金鑰管理員才有權排程或取消金鑰刪除。不過，您可以將 `kms:ScheduleKeyDeletion` 和 `kms:CancelKeyDeletion` 許可新增至金鑰政策或 IAM 政策，從而將這些許可授予給其他 IAM 身

分。您也可以使用kms:ScheduleKeyDeletionPendingWindowInDays條件鍵來進一步限制主參數可在ScheduleKeyDeletion請求PendingWindowInDays參數中指定的值。

允許金鑰管理員排程和取消金鑰刪除 (主控台)

將排程和取消金鑰刪除的許可授予給金鑰管理員。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 針對您要變更許可的 KMS 金鑰，選擇其別名或金鑰 ID。
5. 選擇 Key policy (金鑰政策) 標籤。
6. 金鑰政策的 default view (預設檢視) 和 policy view (政策檢視) 的下一步有所不同。只有當您使用預設的主控台金鑰政策時，才能使用預設檢視。否則，只能使用政策檢視。

   當預設檢視可用時，Switch to policy view (切換為政策檢視) 或 Switch to default view (切換為預設檢視) 按鈕顯示在 Key policy (金鑰政策) 索引標籤中。

   - 在預設檢視中：
     - 在 Key deletion (金鑰刪除) 下方，選擇 Allow key administrators to delete this key (允許金鑰管理員刪除此金鑰)。
   - 在政策檢視中：
     a. 選擇編輯。
     b. 在金鑰管理員的政策陳述式中，將 kms:ScheduleKeyDeletion 和 kms:CancelKeyDeletion 許可新增至 Action 元素。

```
{
  "Sid": "Allow access for Key Administrators",
  "Effect": "Allow",
  "Principal": {"AWS": "arn:aws:iam::111122223333:user/KMSKeyAdmin"},
  "Action": [
    "kms:Create*",
    "kms:Describe*",
    "kms:Enable*",
    "kms:List*",
    "kms:Put*",
    "kms:Update*",
```

```
    "kms:Revoke*",
    "kms:Disable*",
    "kms:Get*",
    "kms:Delete*",
    "kms:ScheduleKeyDeletion",
    "kms:CancelKeyDeletion"
  ],
  "Resource": "*"
}
```

c. 選擇儲存變更。

允許金鑰管理員許可，以排程和取消金鑰刪除 (AWS CLI)

您可以使用 AWS Command Line Interface來新增排程和取消金鑰刪除的許可。

若要新增排程和取消金鑰刪除的許可

1. 使用 aws kms get-key-policy 命令擷取現有的金鑰政策，然後將政策文件儲存至檔案。
2. 在您偏好的文字編輯器中開啟政策文件。在金鑰管理員的政策陳述式中，新增 kms:ScheduleKeyDeletion 和 kms:CancelKeyDeletion 許可。以下範例顯示具有這兩個許可的政策陳述式：

```
{
  "Sid": "Allow access for Key Administrators",
  "Effect": "Allow",
  "Principal": {"AWS": "arn:aws:iam::111122223333:user/KMSKeyAdmin"},
  "Action": [
    "kms:Create*",
    "kms:Describe*",
    "kms:Enable*",
    "kms:List*",
    "kms:Put*",
    "kms:Update*",
    "kms:Revoke*",
    "kms:Disable*",
    "kms:Get*",
    "kms:Delete*",
    "kms:ScheduleKeyDeletion",
    "kms:CancelKeyDeletion"
  ],
  "Resource": "*"
```

```
}
```

3. 使用 `aws kms put-key-policy` 命令將金鑰政策套用到 KMS 金鑰。

排程和取消金鑰刪除

下列處理程序描述如何在 AWS KMS 中，使用 AWS Management Console、AWS CLI 和 AWS SDK for Java 排程刪除單一區域 AWS KMS keys (KMS 金鑰) 金鑰和取消刪除金鑰。

如需排程刪除多區域金鑰的相關資訊，請參閱 刪除多區域金鑰。

> **Warning**
>
> 刪除 KMS 金鑰很具有破壞性，可能有潛在危險。只有當您確定不再需要使用 KMS 金鑰，將來也不需要使用 KMS 金鑰時，才應繼續進行。如果不確定，您應停用 KMS 金鑰，而不是刪除。

在可以刪除 KMS 金鑰之前，您必須擁有執行此操作的許可。如需有關將這些許可授予給金鑰管理員的資訊，請參閱 控制對金鑰刪除的存取。您也可利用 `kms:ScheduleKeyDeletionPendingWindowInDays` 條件金鑰來進一步限制等待期間，例如強制執行最短等待期。

AWS KMS 會在您排程刪除 KMS 金鑰和實際刪除 KMS 金鑰時在 AWS CloudTrail 日誌中記錄一條項目。

排程和取消金鑰刪除 (主控台)

在 AWS Management Console 中，您可以一次排程和取消刪除多個 KMS 金鑰。

排程金鑰刪除

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。

   您無法排程刪除 AWS 受管金鑰 或 AWS 擁有的金鑰。
4. 選擇您要刪除的 KMS 金鑰旁邊的核取方塊。
5. 選擇 Key actions (金鑰動作)、Schedule key deletion (排程金鑰刪除)。

6. 閱讀並考量於等待期間取消刪除的警告及資訊。如果決定取消刪除，請選擇頁面底部的 Cancel (取消)。
7. 對於 Waiting period (in days) (等候期間 (以天為單位))，輸入介於 7 和 30 之間的數字。
8. 檢閱您要刪除的 KMS 金鑰。
9. 選擇 Confirm you want to schedule this key for deletion in ***<number of days>*** days. (確認您想要排程在 <number of days> 天內刪除此金鑰。) 旁的核取方塊。
10. 選擇 Schedule deletion (排定刪除)。

KMS 金鑰狀態會變更為 Pending deletion (待刪除)。

若要取消金鑰刪除

1. 開啟位於 https://console.aws.amazon.com/kms 的 AWS KMS 主控台。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 選擇您要復原的 KMS 金鑰旁邊的核取方塊。
5. 選擇 Key actions (金鑰動作)、Cancel key deletion (取消金鑰刪除)。

KMS 金鑰狀態會從 Pending deletion (待刪除) 變更為 Disabled (已停用)。若要使用 KMS 金鑰，您必須先將其啟用。

排程和取消金鑰刪除 (AWS CLI)

使用 `aws kms schedule-key-deletion` 命令排程客戶受管金鑰的金鑰刪除，如以下範例所示。

您無法排程刪除 AWS 受管金鑰 或 AWS 擁有的金鑰。

```
$ aws kms schedule-key-deletion --key-id 1234abcd-12ab-34cd-56ef-1234567890ab --
pending-window-in-days 10
```

成功使用時，AWS CLI 會傳回像以下範例顯示的輸出：

```
{
    "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "DeletionDate": 1598304792.0,
    "KeyState": "PendingDeletion",
```

```
    "PendingWindowInDays": 10
}
```

使用 aws kms cancel-key-deletion 命令從 AWS CLI 取消刪除金鑰，如以下範例所示。

```
$ aws kms cancel-key-deletion --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

成功使用時，AWS CLI 會傳回像以下範例顯示的輸出：

```
{
    "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
}
```

KMS 金鑰狀態會從 Pending Deletion (待刪除) 變更為 Disabled (已停用)。若要使用 KMS 金鑰，您必須先將其啟用。

排程和取消金鑰刪除 (AWS SDK for Java)

以下範例示範如何使用 AWS SDK for Java 來排程客戶受管金鑰的刪除。此範例要求您之前先將 AWSKMSClient 執行個體化為 kms。

```
String KeyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

int PendingWindowInDays = 10;

ScheduleKeyDeletionRequest scheduleKeyDeletionRequest =
new
 ScheduleKeyDeletionRequest().withKeyId(KeyId).withPendingWindowInDays(PendingWindowInDays);
kms.scheduleKeyDeletion(scheduleKeyDeletionRequest);
```

以下範例示範如何使用 AWS SDK for Java 取消金鑰刪除。此範例要求您之前先將 AWSKMSClient 執行個體化為 kms。

```
String KeyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

CancelKeyDeletionRequest cancelKeyDeletionRequest =
new CancelKeyDeletionRequest().withKeyId(KeyId);
kms.cancelKeyDeletion(cancelKeyDeletionRequest);
```

KMS 金鑰狀態會從 Pending Deletion (待刪除) 變更為 Disabled (已停用)。若要使用 KMS 金鑰，您必須先將其啟用。

建立警示，偵測正在等待刪除之 KMS 金鑰的使用情況

您可以結合 Amazon CloudWatch 日誌和亞馬遜簡單通知服務 (Amazon SNS) 的功能，以建立 Amazon CloudWatch 警示，當您帳戶中有人嘗試使用待刪除的 KMS 金鑰時通知您。AWS CloudTrail 如果您收到此通知，您可能想要取消刪除 KMS 金鑰並重新考慮您的刪除決定。

下列程序會建立警示，在將 "*Key ARN* is pending deletion" 錯誤訊息寫 CloudTrail 入記錄檔時通知您。此錯誤訊息指示有人或應用程式嘗試在密碼編譯操作中使用 KMS 金鑰。由於通知是連結到錯誤訊息，因此當您使用在待刪除的 KMS 金鑰上獲允許的 API 操作時，例如 `ListKeys`、`CancelKeyDeletion` 和 `PutKeyPolicy`，不會觸發通知。若要查看傳回此錯誤訊息的 AWS KMS API 操作清單，請參閱AWS KMS 金鑰的金鑰狀態。

您收到的通知電子郵件不會列出 KMS 金鑰或密碼編譯操作。您可以在您的 CloudTrail 日誌中找到該資訊。反之，電子郵件會報告警示狀態從正常變更為警示。如需 CloudWatch警示和狀態變更的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用 Amazon CloudWatch 警示。

> **Warning**
>
> 此 Amazon CloudWatch 警示無法偵測到非對稱 KMS 金鑰之外的公開金鑰是否使用情況AWS KMS。如需刪除用於公有金鑰密碼編譯之非對稱 KMS 金鑰特殊風險的詳細資訊，包括建立無法解密的加密文字，請參閱 刪除非對稱 KMS 金鑰。

主題

- CloudWatch 警報的要求
- 建立 CloudWatch 鬧鐘

CloudWatch 警報的要求

在建立 CloudWatch 警示之前，您必須先建立AWS CloudTrail追蹤並進行設定， CloudTrail 以便將日 CloudTrail 誌檔傳送到 Amazon CloudWatch Logs。您也需要警示通知的 Amazon SNS 主題。

- 建立 CloudTrail 線索。

  CloudTrail 在您建立帳戶AWS 帳戶時會自動啟用。不過，若要持續記錄您帳戶的事件 (包括 AWS KMS 的事件)，請建立線索。

- 設定 CloudTrail 以傳送記錄檔 CloudWatch 記錄檔。

  設定將記 CloudTrail 錄檔傳送至 CloudWatch 記錄檔。這可讓 CloudWatch 記錄監控嘗試使用待刪除之 KMS 金鑰的 AWS KMS API 要求的記錄。
- 建立一個 Amazon SNS 主題。

  觸發警示時，會將電子郵件訊息傳送至 Amazon Simple Notification Service (Amazon SNS) 主題中的電子郵件地址來通知您。

建立 CloudWatch 鬧鐘

在此程序中，您會建立 CloudWatch 日誌群組測量結果篩選，以尋找擱置刪除例外狀況的執行個體。然後，您可以根據記錄群組指標建立 CloudWatch 警示。如需日誌群組指標篩選器的相關資訊，請參閱 Amazon CloudWatch Logs 使用者指南中的使用篩選器從日誌事件建立指標。

1. 建立剖析 CloudTrail 記錄檔的 CloudWatch 度量篩選器。

   遵循建立日誌群組的指標篩選條件中的指示，使用以下所需值。對於其他欄位，請接受預設值，並按要求提供名稱。

欄位	值
篩選條件模式	`{ $.eventSource = kms* && $.errorMessage = "* is pending deletion."}`
指標值	1

2. 根據您在步驟 1 中建立的量度篩選器建立 CloudWatch 警示。

   遵循使用下列必要值根據記錄群組度量篩選器建立 CloudWatch 警示中的指示。對於其他欄位，請接受預設值，並按要求提供名稱。

欄位	值
指標篩選條件	您在步驟 1 中建立的指標篩選條件的名稱。
閾值類型	靜態
條件	每當####大於 1 時

欄位	值
要警示的資料點	1 個 (共 1 個)
遺失資料處理	將遺失資料視為良好 (不違反閾值)

完成此程序後，每次新 CloudWatch警報進入ALARM狀態時，您都會收到通知。如果您收到此警示的通知，可能表示仍需要排程刪除的 KMS 金鑰來加密或解密資料。在這種情況下，請取消刪除 KMS 金鑰並重新考慮您的刪除決定。

判斷 KMS 金鑰的過去使用情形

刪除 KMS 金鑰之前，您可能想要知道有多少加密文字在該金鑰下加密。AWS KMS 不會存放此資訊，也不會存放任何加密文字。了解 KMS 金鑰的過去使用情形，可能有助您決定將來是否需要它。本主題會建議幾項策略，協助您判斷 KMS 金鑰的過去使用情形。

> **Warning**
>
> 這些決定過去和實際用量的策略僅對 AWS 使用者和 AWS KMS 操作有效。他們偵測不到在 AWS KMS 外部對非對稱 KMS 金鑰之公有金鑰的使用。如需刪除用於公有金鑰密碼編譯之非對稱 KMS 金鑰特殊風險的詳細資訊，包括建立無法解密的加密文字，請參閱 刪除非對稱 KMS 金鑰。

主題

- 檢查 KMS 金鑰許可，以判斷潛在使用情形的範圍
- 檢查 AWS CloudTrail 日誌來判斷實際使用量

檢查 KMS 金鑰許可，以判斷潛在使用情形的範圍

判斷何人或何物目前可存取 KMS 金鑰，可能有助您判斷 KMS 金鑰的使用範圍以及是否仍然需要它。若要了解如何判斷何人或何物目前可以存取 KMS 金鑰，請前往 判斷 AWS KMS keys 的存取權。

檢查 AWS CloudTrail 日誌來判斷實際使用量

您可以使用 KMS 金鑰使用歷史記錄來協助您判斷是否擁有在特定 KMS 金鑰下加密的加密文字。

全部 AWS KMS API 活動記錄在 AWS CloudTrail 日誌檔案中。如果您已在 KMS 金鑰所在的區域建立 CloudTrail 追蹤，您可以檢查 CloudTrail 記錄檔以檢視特定 KMS 金鑰的所有 AWS KMS API 活動歷史記錄。如果您沒有追蹤，您仍然可以在活動歷史記錄中檢視最近的CloudTrail 事件。如需AWS KMS使用方式的詳細資訊 CloudTrail，請參閱使用 AWS CloudTrail 記錄 AWS KMS API 呼叫。

下列範例顯示 CloudTrail 使用 KMS 金鑰保護存放在 Amazon 簡單儲存服務 (Amazon S3) 中的物件時所產生的日誌項目。在此範例中，物件會使用透過含有 KMS 金鑰的伺服器端加密 (SSE-KMS) 來保護資料 上傳至 Simple Storage Service (Amazon S3)。當您使用 SSE-KMS 將物件上傳到 Amazon S3 時，請指定用於保護物件的 KMS 金鑰。Amazon S3 會使用此AWS KMSGenerateDataKey操作為物件要求唯一的資料金鑰，且此請求事件會以類似下列內 CloudTrail 容的項目登入：

```
{
  "eventVersion": "1.02",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROACKCEVSQ6C2EXAMPLE:example-user",
    "arn": "arn:aws:sts::111122223333:assumed-role/Admins/example-user",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2015-09-10T23:12:48Z"
      },
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/Admins",
        "accountId": "111122223333",
        "userName": "Admins"
      }
    },
    "invokedBy": "internal.amazonaws.com"
  },
  "eventTime": "2015-09-10T23:58:18Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "GenerateDataKey",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "internal.amazonaws.com",
  "userAgent": "internal.amazonaws.com",
  "requestParameters": {
    "encryptionContext": {"aws:s3:arn": "arn:aws:s3:::example_bucket/example_object"},
```

```
    "keySpec": "AES_256",
    "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
  },
  "responseElements": null,
  "requestID": "cea04450-5817-11e5-85aa-97ce46071236",
  "eventID": "80721262-21a5-49b9-8b63-28740e7ce9c9",
  "readOnly": true,
  "resources": [{
    "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "accountId": "111122223333"
  }],
  "eventType": "AwsApiCall",
  "recipientAccountId": "111122223333"
}
```

稍後當您從 Amazon S3 下載此物件時，Amazon S3 會向 AWS KMS 傳送 `Decrypt` 請求，以使用指定的 KMS 金鑰解密物件的資料金鑰。執行此操作時， CloudTrail 記錄檔會包含類似下列內容的項目：

```
{
  "eventVersion": "1.02",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROACKCEVSQ6C2EXAMPLE:example-user",
    "arn": "arn:aws:sts::111122223333:assumed-role/Admins/example-user",
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2015-09-10T23:12:48Z"
      },
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/Admins",
        "accountId": "111122223333",
        "userName": "Admins"
      }
    },
    "invokedBy": "internal.amazonaws.com"
  },
```

```
  "eventTime": "2015-09-10T23:58:39Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "Decrypt",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "internal.amazonaws.com",
  "userAgent": "internal.amazonaws.com",
  "requestParameters": {
    "encryptionContext": {"aws:s3:arn": "arn:aws:s3:::example_bucket/example_object"}},
  "responseElements": null,
  "requestID": "db750745-5817-11e5-93a6-5b87e27d91a0",
  "eventID": "ae551b19-8a09-4cfc-a249-205ddba330e3",
  "readOnly": true,
  "resources": [{
    "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "accountId": "111122223333"
  }],
  "eventType": "AwsApiCall",
  "recipientAccountId": "111122223333"
}
```

CloudTrail 會記錄所有 AWS KMS API 活動。透過評估這些日誌項目，您或許可以確定特定 KMS 金鑰的過去使用情形，這可能有助於您確定是否要刪除它。

若要查看更多 AWS KMS API 活動如何在 CloudTrail 記錄檔中顯示的範例，請移至使用 AWS CloudTrail 記錄 AWS KMS API 呼叫。如需有關的詳細資訊， CloudTrail 請參閱使AWS CloudTrail用者指南。

AWS KMS 金鑰的金鑰狀態

AWS KMS key 始終具有金鑰狀態。KMS 金鑰及其環境上的操作可以暫時變更該金鑰狀態，或直到另一個操作變更其金鑰狀態為止。

本節中的資料表顯示金鑰狀態如何影響對 AWS KMS API 操作的呼叫。其金鑰狀態的結果，KMS 金鑰上的操作預期會成功 (#)，失敗 (X) ，或者只在某些條件下成功 (?)。結果對於含匯入金鑰材料的 KMS 金鑰來說，通常是不同的。

此資料表僅包含使用現有 KMS 金鑰的 API 操作。省略其他作業 CreateKeyListKeys，例如和。

主題

- 金鑰狀態和 KMS 金鑰類型

- 金鑰狀態資料表

金鑰狀態和 KMS 金鑰類型

KMS 金鑰的類型決定了其可以具有的金鑰狀態。

- 所有 KMS 金鑰都可以處於 Enabled、Disabled 和 PendingDeletion 狀態。
- 大部分的 KMS 金鑰都是在 Enabled 狀態中建立。具有匯入金鑰材料的金鑰會在 PendingImport 狀態中建立。
- PendingImport 狀態僅適用於具有匯入金鑰材料的 KMS 金鑰。
- Unavailable 狀態僅適用於自訂金鑰存放區中的 KMS 金鑰。當自訂金鑰存放區刻意中斷與其 AWS CloudHSM 叢集的連接時，AWS CloudHSM 金鑰存放區中的 KMS 金鑰為 Unavailable。當自訂金鑰存放區刻意中斷與其外部金鑰存放區代理的連接時，外部金鑰存放區中的 KMS 金鑰為 Unavailable。您可以檢視和管理無法使用的 KMS 金鑰，但無法在密碼編譯操作中使用它們。

  自訂金鑰存放區中 KMS 金鑰的金鑰狀態不會受其備份金鑰的變更影響。AWS CloudHSM 金鑰存放區中的 KMS 金鑰不會受 AWS CloudHSM 叢集中其關聯金鑰材料的變更影響。外部金鑰存放區中的 KMS 金鑰不會受外部金鑰管理器中外部金鑰的變更影響。如果停用或刪除備份金鑰，KMS 金鑰狀態不會變更，但使用 KMS 金鑰進行的密碼編譯操作會失敗。
- Creating、Updating 和PendingReplicaDeletion 金鑰狀態僅適用於多區域金鑰。
  - 多區域複本金鑰在建立時處於暫時 Creating 金鑰狀態。ReplicateKey作業完成時，此程序可能仍在進行中。複寫程序完成時，複本金鑰處於 Enabled 或 PendingImport 狀態。
  - 多區域金鑰在主要區域更新時處於暫時 Updating 金鑰狀態。UpdatePrimaryRegion作業完成時，此程序可能仍在進行中。當更新程序完成時，主要和複本金鑰會繼續保持 Enabled 金鑰狀態。
  - 當您排程刪除具有複本金鑰的多區域主要金鑰時，主要金鑰處於 PendingReplicaDeletion 狀態，直到刪除其所有複本金鑰為止。其金鑰狀態會變更為 PendingDeletion。如需詳細資訊，請參閱 刪除多區域金鑰。

金鑰狀態資料表

下表顯示 KMS 金鑰的金鑰狀態如何影響 AWS KMS 操作。

編號註腳的描述 ([n]) 在本主題的結尾處。

> **Note**
>
> 您可能需要水平或垂直捲動，才能查看此資料表中的所有資料。

API	已啟用	已停用	待刪除 待刪除複本	待匯入	Unavailable	正在建立	更新中
CancelKeyDeletion	 [4]	 [4]		 [4]	 [4], [13]	 [4]	 [4]
CreateAlias			 [3]				
CreateGrant		 [1]	 [2] 或 [3]	 [5]		 [14]	
解密		 [1]	 [2] 或 [3]	 [5]	 [11]	 [14]	
DeleteAlias							
DeleteImportedKeyMaterial	 [9]	 [9]	 [9]	 (沒有效果)	N/A	 [14]	[15]
DescribeKey							

API	已啟用	已停用	待刪除 待刪除複本	待匯入	Unavailable	正在建立	更新中
DisableKey	✓	✓	⊗ [3]	⊗ [5]	✓ [12]	⊗ [14]	⊗ [15]
DisableKeyRotation	? [7]	⊗ [1] 或 [7]	⊗ [3] 或 [7]	⊗ [6]	⊗ [7]	⊗ [14]	? [7]
EnableKey	✓	✓	⊗ [3]	⊗ [5]	✓ [12]	⊗ [14]	⊗ [15]
EnableKeyRotation	? [7]	⊗ [1] 或 [7]	⊗ [3] 或 [7]	⊗ [6]	⊗ [7]	⊗ [14]	? [7]
加密	✓	⊗ [1]	⊗ [2] 或 [3]	⊗ [5]	⊗ [11]	⊗ [14]	✓
GenerateDataKey	✓	⊗ [1]	⊗ [2] 或 [3]	⊗ [5]	⊗ [11]	⊗ [14]	✓
GenerateDataKeyPair	✓	⊗ [1]	⊗ [2] 或 [3]	⊗ [5]	⊗ [11]	⊗ [14]	✓

API	已啟用	已停用	待刪除 待刪除複本	待匯入	Unavailable	正在建立	更新中
GenerateDataKeyPairWithoutPlaintext	✓	⊗ [1]	⊗ [2] 或 [3]	⊗ [5]	⊗ [11]	⊗ [14]	✓
GenerateDataKeyWithoutPlaintext	✓	⊗ [1]	⊗ [2] 或 [3]	⊗ [5]	⊗ [11]	⊗ [14]	✓
GenerateMac	✓	⊗ [1]	⊗ [2] 或 [3]	N/A	N/A	⊗ [14]	✓
GetKeyPolicy	✓	✓	✓	✓	✓	✓	✓
GetKeyRotationStatus	? [7]	? [7]	? [7]	⊗ [6]	⊗ [7]	? [7]	? [7]
GetParametersForImport	? [9]	? [9]	⊗ [8] 或 [9]	✓	⊗ [9]	⊗ [14]	⊗ [15]
GetPublicKey	✓	⊗ [1]	⊗ [2] 或 [3]	N/A	N/A	⊗ [14]	✓

API	已啟用	已停用	待刪除 待刪除複本	待匯入	Unavailable	正在建立	更新中
ImportKeyMaterial	[9]	[9]	[8] 或 [9]	✓	[9]	[14]	✓
ListAliases	✓	✓	✓	✓	✓	✓	✓
ListGrants	✓	✓	✓	✓	✓	✓	✓
ListKeyPolicies	✓	✓	✓	✓	✓	✓	✓
ListResourceTags	✓	✓	✓	✓	✓	✓	✓
PutKeyPolicy	✓	✓	✓	✓	✓	✓	✓
ReEncrypt	✓	[1]	[2] 或 [3]	[5]	[11]	[14]	✓
ReplicateKey	✓	[1]	[2] 或 [3]	[5]	N/A	[14]	[15]
RetireGrant	✓	✓	✓	✓	✓	✓	✓
RevokeGrant	✓	✓	✓	✓	✓	✓	✓

API	已啟用	已停用	待刪除 待刪除複本	待匯入	Unavailable	正在建立	更新中
ScheduleKeyDeletion	✓	✓	⊗ [3]	✓	✓	✓	⊗ [15]
符號	✓	⊗ [1]	⊗ [2] 或 [3]	N/A	N/A	⊗ [14]	✓
TagResource	✓	✓	⊗ [3]	✓	✓	✓	✓
UntagResource	✓	✓	⊗ [3]	✓	✓	✓	✓
UpdateAlias	✓	✓	? [10]	✓	✓	✓	✓
UpdateKeyDescription	✓	✓	⊗ [3]	✓	✓	✓	✓
UpdatePrimaryRegion	✓	⊗ [1]	⊗ [2] 或 [3]	⊗ [5]	N/A	⊗ [14]	✓

API	已啟用	已停用	待刪除 待刪除複本	待匯入	Unavailable	正在建立	更新中
確認	✓	⊗ [1]	⊗ [2] 或 [3]	N/A	N/A	⊗ [14]	✓
VerifyMac	✓	⊗ [1]	⊗ [2] 或 [3]	N/A	N/A	⊗ [14]	✓

資料表詳細資訊

- [1] `DisabledException: <key ARN> is disabled.`
- [2] `DisabledException: <key ARN> is pending deletion (or pending replica deletion).`
- [3] `KMSInvalidStateException: <key ARN> is pending deletion (or pending replica deletion).`
- [4] `KMSInvalidStateException: <key ARN> is not pending deletion (or pending replica deletion).`
- [5] `KMSInvalidStateException: <key ARN> is pending import.`
- [6] `UnsupportedOperationException: <key ARN> origin is EXTERNAL which is not valid for this operation.`
- [7] 如果 KMS 金鑰已匯入金鑰材料或在自訂金鑰存放區中：`UnsupportedOperationException`。
- [8] 如果 KMS 金鑰已匯入金鑰材料：`KMSInvalidStateException`
- [9] 如果 KMS 金鑰無法或未匯入金鑰材料：`UnsupportedOperationException`。
- [10] 如果來源 KMS 金鑰正在等待刪除，則命令成功。如果目的地 KMS 金鑰正在等待刪除，則命令失敗，並出現錯誤：`KMSInvalidStateException : <key ARN> is pending deletion.`
- [11] `KMSInvalidStateException: <key ARN> is unavailable.` 對於無法使用的 KMS 金鑰，您無法執行此操作。

- [12] 操作成功，但 KMS 金鑰在變成可用之前，狀態不會變更。
- [13] 當自訂金鑰存放區中的 KMS 金鑰等待刪除時，即使 KMS 金鑰變成無法使用，其金鑰狀態仍為 PendingDeletion。這可讓您在等待期間隨時取消刪除 KMS 金鑰。
- [14] `KMSInvalidStateException: <key ARN> is creating.` AWS KMS 在複寫多區域金鑰 (ReplicateKey) 時，會擲回此例外狀況。
- [15] `KMSInvalidStateException: <key ARN> is updating.` AWS KMS 在更新多區域金鑰 (UpdatePrimaryRegion) 的主要區域時，會擲回此例外狀況。

AWS KMS 的身分驗證與存取控制

若要使用 AWS KMS，您必須擁有 AWS 可以用來驗證您的請求的憑證。憑證必須包含存取 AWS 資源：AWS KMS keys 和別名的許可。任何 AWS 主體均沒有 KMS 金鑰的任何許可，除非有明確提供該許可且永遠不會拒絕。沒有能使用或管理 KMS 金鑰的隱含或自動許可。

管理對 AWS KMS 資源之存取的主要方法是使用政策。政策是描述哪些主體可以存取哪些資源的文件。連接至 IAM 身分的政策稱為以身分為基礎的政策 (或 IAM 政策)，而連接至其他資源類型的政策稱為資源政策。KMS 金鑰的 AWS KMS 資源政策稱為金鑰政策。所有 KMS 金鑰都擁有金鑰政策。

若要控制對 AWS KMS 別名的存取，請使用 IAM 政策。若要允許主體建立別名，您必須提供 IAM 政策中的別名許可，以及金鑰政策中的金鑰許可。如需詳細資訊，請參閱 控制對別名的存取。

若要控制對 KMS 金鑰的存取，您可以使用以下政策機制。

- 金鑰政策 – 每個 KMS 金鑰都有一個金鑰政策。它是控制對 KMS 金鑰之存取的主要機制。您可以單獨使用金鑰政策來控制存取，這表示在單一文件 (金鑰政策) 中定義 KMS 金鑰的完整存取範圍。如需使用金鑰政策的詳細資訊，請參閱金鑰政策。
- IAM 政策 – 您可以使用 IAM 政策搭配金鑰政策和授權，來控制對 KMS 金鑰的存取。透過這種方式控制存取，可讓您管理 IAM 中 IAM 身分的所有許可。若要使用 IAM 政策來允許存取 KMS 金鑰，金鑰政策必須明確允許。如需使用 IAM 政策的詳細資訊，請參閱 IAM 政策。
- 授權 – 您可以使用授權搭配金鑰政策和 IAM 政策，來允許對 KMS 金鑰的存取。透過這種方式控制存取，可讓您在金鑰政策中允許存取 KMS 金鑰，並且允許身分將其存取權委派給其他人。如需使用授與的詳細資訊，請參閱AWS KMS 中的授權。

KMS 金鑰屬於建立所在的 AWS 帳戶。但是，身分或主體 (包括 AWS 帳戶根使用者)均沒有使用或管理 KMS 金鑰的許可，除非在金鑰政策、IAM 政策或授予中有明確提供該許可。建立 KMS 金鑰的 IAM 身分不會被視為金鑰擁有者，而且他們不會自動擁有使用或管理他們建立之 KMS 金鑰的許可。與任何其他身分一樣，金鑰建立者必須透過金鑰政策、IAM 政策或授予取得許可。不過，擁有 `kms:CreateKey` 許可的身分可以設定初始金鑰政策，並授予自己使用或管理金鑰的許可。

以下主題提供如何使用 AWS Identity and Access Management (IAM) 與 AWS KMS 許可的詳細資訊，藉由控制可存取的人員，協助確保您資源的安全。

主題

- AWS KMS 存取控制中的概念
- AWS KMS 中的金鑰政策

- 將 IAM 政策與 AWS KMS 搭配使用
- AWS KMS 中的授權
- 透過 VPC 端點連線至 AWS KMS
- 條件鍵 AWS KMS
- AWS KMS 的 ABAC
- 允許其他帳戶中的使用者使用 KMS 金鑰
- 使用 AWS KMS 的服務連結角色
- 搭配 AWS KMS 使用混合式後量子 TLS
- 判斷 AWS KMS keys 的存取權
- AWS KMS 權限
- 測試您的許可

AWS KMS 存取控制中的概念

了解 AWS KMS 中討論存取控制時使用的概念。

主題

- 身分驗證
- 授權
- 使用身分驗證
- 使用政策管理存取權
- AWS KMS 資源

身分驗證

身分驗證是驗證您身分的過程。若要傳送請求至 AWS KMS，您必須利用您的 AWS 憑證登入 AWS。

授權

授權提供傳送請求的許可，以建立、管理或使用 AWS KMS 資源。例如，您必須取得授權，才可以在密碼編譯操作中使用 KMS 金鑰。

使用金鑰政策、IAM 政策和授予，控制對 AWS KMS 資源的存取。每個 KMS 金鑰都必須有一個金鑰政策。如果金鑰政策允許，您也可以使用 IAM 政策和授予，以允許主體存取 KMS 金鑰。若要優化您

的授權，您可以使用條件金鑰，只有在請求或資源符合您指定的條件時，條件金鑰才允許或拒絕存取。您也可以允許存取其他 AWS 帳戶 中您信任的主體。

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》https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html#rotate-credentials中的為需要長期憑證的使用案例定期輪換存取金鑰。

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

使用者與角色不同。使用者只會與單一人員或應用程式建立關聯，但角色的目的是在由任何需要它的人員取得。使用者擁有永久的長期憑證，但角色僅提供暫時憑證。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》https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_create_for-idp.html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。

- 跨帳户存取權 – 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。角色是授予跨帳户存取權的主要方式。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》。
  - 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

AWS KMS 金鑰政策是資源型政策，可控制對 KMS 金鑰的存取。每個 KMS 金鑰都必須有一個金鑰政策。您可以使用其他授權機制來允許存取 KMS 金鑰，但前提是金鑰政策允許。(您可以使用 IAM 政策來拒絕對 KMS 金鑰的存取，即使金鑰政策未明確允許。)

資源型政策是連接到資源 (如 KMS 金鑰) 的 JSON 政策文件，以控制對特定資源的存取。資源型政策可定義指定的主體在何種條件下對該資源執行哪些動作。您不會在資源型政策中指定資源，但必須指定主體，例如帳戶、使用者、角色、聯合身分使用者或 AWS 服務。資源型政策是管理該資源的服務中的內嵌政策。您無法在資源型政策中使用來自 IAM 的 AWS 受管政策，例如 `AWSKeyManagementServicePowerUser` 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。

AWS KMS 不支援 ACL。

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 KMS 資源

AWS KMS 的主要資源是 AWS KMS key。AWS KMS 也支援別名，此為向 KMS 金鑰提供易記名稱的獨立資源。某些 AWS KMS 操作允許您使用別名來識別 KMS 金鑰。

KMS 金鑰或別名的每個執行個體都具有唯一的 Amazon Resource Name (ARN)，具有標準格式。在 AWS KMS 資源中，AWS 服務名稱為 `kms`。

- AWS KMS key

ARN 格式：

```
arn:AWS partition name:AWS service name:AWS ##:AWS ## ID:key/key ID
```

範例 ARN：

```
arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
```

- Alias (別名)

  ARN 格式：

  ```
  arn:AWS partition name:AWS service name:AWS ##:AWS ## ID:alias/alias name
  ```

  範例 ARN：

  ```
  arn:aws:kms:us-west-2:111122223333:alias/example-alias
  ```

AWS KMS 提供一組 API 操作，用於處理 AWS KMS 資源。如需在 AWS Management Console 和 AWS KMS API 操作中識別 KMS 金鑰的詳細資訊，請參閱金鑰識別碼 (KeyId)。對於 AWS KMS 操作的清單，請參閱 AWS Key Management Service API 參考。

AWS KMS 中的金鑰政策

金鑰政策是用於 AWS KMS key 的資源政策。金鑰政策是控制對 KMS 金鑰之存取的主要方式。每個 KMS 金鑰都必須只有一個金鑰政策。金鑰政策中的陳述式決定誰有使用 KMS 金鑰的許可以及可以使用它的方式。您也可以使用 IAM 政策和授予來控制對 KMS 金鑰的存取，但每個 KMS 金鑰都必須有金鑰政策。

任何 AWS 主體 (包括帳戶根使用者或金鑰建立者) 均沒有 KMS 金鑰的任何許可，除非在金鑰政策、IAM 政策或授予中明確允許，且永遠不會拒絕。

除非金鑰政策明確允許，否則您不能使用 IAM 政策來允許存取 KMS 金鑰。如果沒有金鑰政策的許可，允許許可的 IAM 政策將不起作用。(您可以使用 IAM 政策，在未經金鑰政策許可的情況下拒絕對 KMS 金鑰的許可。) 預設金鑰政策會啟用 IAM 政策。若要在金鑰政策中啟用 IAM 政策，請新增 允許存取 AWS 帳戶 並啟用 IAM 政策 中所述的政策陳述式。

與全域 IAM 政策不同，金鑰政策是區域性的。金鑰政策僅控制對同一區域中 KMS 金鑰的存取。它對其他區域中的 KMS 金鑰沒有影響。

主題

建立金鑰政策

您可以使用 AWS KMS API 操作，例如、和或使用AWS CloudFormation範本，在AWS KMS主控台中建立和PutKeyPolicy管理金鑰政策。CreateKeyReplicateKey

您在 AWS KMS 主控台中建立 KMS 金鑰時，主控台會指導您根據主控台的預設金鑰政策，來完成建立金鑰政策的步驟。使用 CreateKey 或 ReplicateKey API 時，若不指定金鑰政策，這些 API 便會為以程式設計方式建立之金鑰，套用預設金鑰政策。當您採用 PutKeyPolicy API 時，必須指定金鑰政策。

每個政策文件可以擁有一個或多個政策陳述式。以下範例顯示具有一個政策陳述式的有效金鑰政策文件。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Describe the policy statement",
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:user/Alice"
      },
      "Action": "kms:DescribeKey",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:KeySpec": "SYMMETRIC_DEFAULT"
        }
      }
    }
  ]
}
```

主題

金鑰政策格式

金鑰政策文件必須符合以下規則：

- 最大為 32 KB (32,768 個位元組)
- 金鑰政策陳述式中的 Sid 元素可包含空格。(IAM 政策文件的 Sid 元素中禁止使用空格。)

金鑰政策文件僅可包含以下字元：

- 可印出的 ASCII 字元
- 基本拉丁字母和拉丁字母補充-1 字元集中的可印出字元
- Tab (\u0009)、換行字元 (\u000A) 和歸位字元 (\u000D) 特殊字元

金鑰政策中的元素

金鑰政策文件必須擁有下列元素：

版本

指定金鑰政策文件版本。將版本設定為 2012-10-17 (最新版本)。

陳述式

包含政策陳述式。金鑰政策文件必須至少包含一個陳述式。

每個金鑰政策陳述式最多可包含六個元素。需要 Effect、Principal、Action 和 Resource 元素。

Sid

(選用) 陳述式識別符 (Sid) 為可用來描述陳述式的任意字串。金鑰政策中的 Sid 可包含空格。(不得在 IAM 政策 Sid 元素中包含空格。)

Effect

(必要) 決定是允許還是拒絕政策陳述式中的許可。有效值為 `Allow` 或 `Deny`。如果您未明確允許存取 KMS 金鑰，將會隱含拒絕存取。您也可以明確拒絕存取 KMS 金鑰。您可以這樣做以確保使用者無法存取 CMK，即使其他政策允許存取。

Principal

(必要) 主體是取得政策陳述式中指定許可的身分。您可以指定 AWS 帳戶、IAM 使用者、IAM 角色以及部分 AWS 服務作為金鑰政策中的主體。IAM 使用者群組在任何政策類型中都不是有效的主體。

星號值，例如 `"AWS": "*"` 代表所有帳戶的所有 AWS 身份。

> **Important**
>
> 除非採用條件來限制金鑰政策，否則請勿在任何允許許可的金鑰政策陳述式將主體設為星號 (*)。星號為每個 AWS 帳戶 許可提供每個身分來使用 KMS 金鑰，除非另一個政策陳述式明確拒絕。其他 AWS 帳戶 的使用者只要其本身帳戶有對應的許可，即可利用您的 KMS 金鑰。

> **Note**
>
> IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

若金鑰政策陳述式中的主體是以 `arn:aws:iam::111122223333:root"` 表達的 AWS 帳戶 主體，則政策陳述式不會向任何 IAM 主體授予許可。而是授予 AWS 帳戶 許可，以使用 IAM 政策，來委派金鑰政策中指定的許可。(雖然帳戶識別符中使用了「根」，但採用 `arn:aws:iam::111122223333:root"` 格式的主體不代表AWS 帳戶根使用者。然而，帳戶主體代表帳戶及其管理員，包括帳戶根用戶。)

當主體是另一個 AWS 帳戶 或其主體時，只有在使用 KMS 金鑰和金鑰政策在區域中啟用帳戶時，許可才會生效。如需相關資訊了解哪些區域預設為未啟用 (「選擇加入區域」)，請參閱《AWS 一般參考》 的管理AWS 區域。

若要允許不同的 AWS 帳戶 或其主體使用 KMS 金鑰，您必須在金鑰政策和其他帳戶的 IAM 政策中提供許可。如需詳細資訊，請參閱 允許其他帳戶中的使用者使用 KMS 金鑰。

動作

(必要) 指定要允許或拒絕的 API 操作。例如，kms:Encrypt 動作對應至 AWS KMS Encrypt 操作。您可以在政策陳述式中列出多個動作。如需詳細資訊，請參閱 許可參考。

資源

(必要) 在金鑰政策中，Resource 元素的值是 "*"，表示「此 KMS 金鑰」。星號 ("*") 會識別金鑰政策所連接的 KMS 金鑰。

> **Note**
>
> 如果必要的 Resource 元素在金鑰政策陳述式中遺失，則政策陳述式沒有任何效果。沒有 Resource 元素的金鑰政策不適用於任何 KMS 金鑰。
> 當金鑰原則陳述式遺失其Resource元素時，AWS KMS主控台會正確回報錯誤，但即使原則陳述式無效，CreateKey和 PutKeyPolicyAPI 也會成功。

條件

(選用) 條件則會指定金鑰政策生效必須符合的需求。使用條件，AWS 可以評估 API 請求的內容，以判斷是否套用政策陳述式。

若要指定條件，請使用預先定義的條件金鑰。AWS KMS 支援 AWS 全球條件金鑰和AWS KMS 條件金鑰。若要支援以屬性為基礎的存取控制 (ABAC)，AWS KMS 會提供根據標籤和別名控制對 KMS 金鑰之存取的條件索引鍵。如需詳細資訊，請參閱 AWS KMS 的 ABAC。

條件的格式為：

```
"Condition": {"condition operator": {"condition key": "condition value"}}
```

例如：

```
"Condition": {"StringEquals": {"kms:CallerAccount": "111122223333"}}
```

如需 AWS 政策語法的詳細資訊，請參閱《IAM 使用者指南》中的 AWS IAM 政策參考。

範例金鑰政策

下列範例顯示對稱加密 KMS 金鑰的完整金鑰政策。在閱讀本章中的關鍵政策概念時，您可以將其用作參考。此金鑰政策將先前預設金鑰政策一節的範例政策陳述式結合至單一金鑰政策，來完成以下任務：

- 允許範例 AWS 帳戶，111122223333，對 KMS 金鑰的完整存取。它允許帳戶及其管理員 (包括帳戶根使用者 (緊急情況)) 在帳戶中使用 IAM 政策，以允許對 KMS 金鑰的存取。
- 允許 ExampleAdminRole IAM 角色管理 KMS 金鑰。
- 允許 ExampleUserRole IAM 角色使用 KMS 金鑰。

```
{
    "Id": "key-consolepolicy",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Sid": "Allow access for Key Administrators",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/ExampleAdminRole"
            },
            "Action": [
                "kms:Create*",
                "kms:Describe*",
                "kms:Enable*",
                "kms:List*",
                "kms:Put*",
                "kms:Update*",
                "kms:Revoke*",
                "kms:Disable*",
                "kms:Get*",
                "kms:Delete*",
                "kms:TagResource",
                "kms:UntagResource",
                "kms:ScheduleKeyDeletion",
                "kms:CancelKeyDeletion"
            ],
            "Resource": "*"
```

```
        },
        {
            "Sid": "Allow use of the key",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/ExampleUserRole"
            },
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*",
                "kms:DescribeKey"
            ],
            "Resource": "*"
        },
        {
            "Sid": "Allow attachment of persistent resources",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/ExampleUserRole"
            },
            "Action": [
                "kms:CreateGrant",
                "kms:ListGrants",
                "kms:RevokeGrant"
            ],
            "Resource": "*",
            "Condition": {
                "Bool": {
                    "kms:GrantIsForAWSResource": "true"
                }
            }
        }
    ]
}
```

預設金鑰政策

在您建立 KMS 金鑰時，可以指定新 KMS 金鑰的金鑰政策。若未提供值，AWS KMS 會為您建立一個。AWS KMS 使用的預設金鑰政策會有所不同，具體取決於您是否在 AWS KMS 主控台中建立金鑰或使用 AWS KMS API。

當您以程式設計方式建立 KMS 金鑰時的預設金鑰政策

當您以程式設計方式使用 AWS KMS API (包括使用 AWS SDK、AWS Command Line Interface 或者 AWS Tools for PowerShell)，並且您不指定金鑰政策時，AWS KMS 會套用非常簡單的預設金鑰政策。此預設金鑰政策具有一個政策陳述式，授權擁有 KMS 金鑰許可的 AWS 帳戶 能夠使用 IAM 政策，以允許存取對 KMS 金鑰進行的所有 AWS KMS 操作。如需有關此政策陳述式的詳細資訊，請參閱允許存取 AWS 帳戶 並啟用 IAM 政策。

當您使用 AWS Management Console建立 KMS 金鑰時的預設金鑰政策

在您使用 AWS Management Console 建立建立 KMS 金鑰時，金鑰政策以政策陳述式開頭，允許對 AWS 帳戶 的存取並啟用 IAM 政策。然後，主控台會新增金鑰管理員陳述式，這是一項金鑰使用者陳述式，也是 (對於大多數金鑰類型) 允許主體將 KMS 金鑰與其他 AWS 服務搭配使用的陳述式。您可以使用 AWS KMS 主控台的功能來指定 IAM 使用者、IAM 角色以及 AWS 帳戶，這些帳戶是金鑰管理員和金鑰使用者 (或兩者兼有)。

許可

- 允許存取 AWS 帳戶 並啟用 IAM 政策
- 允許金鑰管理員來管理 KMS 金鑰
- 允許金鑰使用者使用 KMS 金鑰
  - 允許金鑰使用者在密碼編譯操作中使用 KMS 金鑰
  - 允許金鑰使用者使用 KMS 金鑰搭配 AWS 服務

允許存取 AWS 帳戶 並啟用 IAM 政策

以下預設金鑰政策陳述式至關重要。

- 它授權擁有 KMS 金鑰的 AWS 帳戶 能夠完整存取 KMS 金鑰。

  不像其他 AWS 資源政策，AWS KMS 金鑰政策不會自動向帳戶或其任何身分授予許可。若要向帳戶管理員授予許可，金鑰政策必須包含提供此許可的明確陳述式。
- 除金鑰政策外，它還允許帳戶使用 IAM 政策來允許對 KMS 金鑰的存取。

  如果沒有此許可，允許存取金鑰的 IAM 政策將無效，但拒絕存取金鑰的 IAM 政策仍然有效。
- 它透過向帳戶管理員 (包括帳戶根使用者) 授予無法刪除存取控制許可，來降低金鑰變得無法管理的風險。

以下金鑰政策陳述式是以程式設計方式建立的 KMS 金鑰的完整預設金鑰政策。這是在 AWS KMS 主控台中建立的 KMS 金鑰的預設金鑰政策中的第一個政策陳述式。

```
{
  "Sid": "Enable IAM User Permissions",
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:root"
   },
  "Action": "kms:*",
  "Resource": "*"
}
```

允許 IAM 政策允許對 KMS 金鑰的存取。

以上所示金鑰政策陳述式授予 AWS 帳戶 擁有使用 IAM 政策以及金鑰政策的金鑰許可，以允許 KMS 金鑰上的所有動作 (`kms:*`)。

此金鑰政策陳述式的主體為帳戶主體，由以下格式的 ARN 表示：`arn:aws:iam::account-id:root`。帳戶主體代表 AWS 帳戶及其管理員。

若金鑰政策陳述式中的主體為帳戶主體，則政策陳述式不會向任何 IAM 主體授予使用 KMS 金鑰的許可。而是允許帳戶使用 IAM 政策，來委派政策陳述式中指定的許可。此預設金鑰政策陳述式允許帳戶使用 IAM 政策，來委派 KMS 金鑰上的所有動作許可 (`kms:*`)。

降低 KMS 金鑰變得無法管理的風險。

不像其他 AWS 資源政策，AWS KMS 金鑰政策不會自動向帳戶或其任何主體授予許可。向包括帳戶主體在內的任何主體授予許可，必須使用明確授予許可的金鑰政策陳述式。您無需授予帳戶主體或任何主體存取 KMS 金鑰的許可。然而，向帳戶主體授予存取許可，能夠幫助您防止金鑰變得無法管理。

例如，假設您建立的金鑰政策僅向一個使用者授予存取 KMS 金鑰的許可。如果您隨後刪除該使用者，金鑰將變得無法管理，您必須聯絡 AWS 支援，以重新取得對 KMS 金鑰的存取。

以下所示金鑰政策陳述式向代表 AWS 帳戶 及其管理員的帳戶主體授予控制金鑰的許可，包括帳戶根使用者。帳戶根使用者是唯一不能刪除的主體，除非您刪除 AWS 帳戶。IAM 最佳實務不鼓勵代表帳戶根使用者採取行動，但在緊急情況下除外。然而，如果刪除具有 KMS 金鑰存取許可的所有其他使用者和角色，則可能需要充當帳戶根使用者。

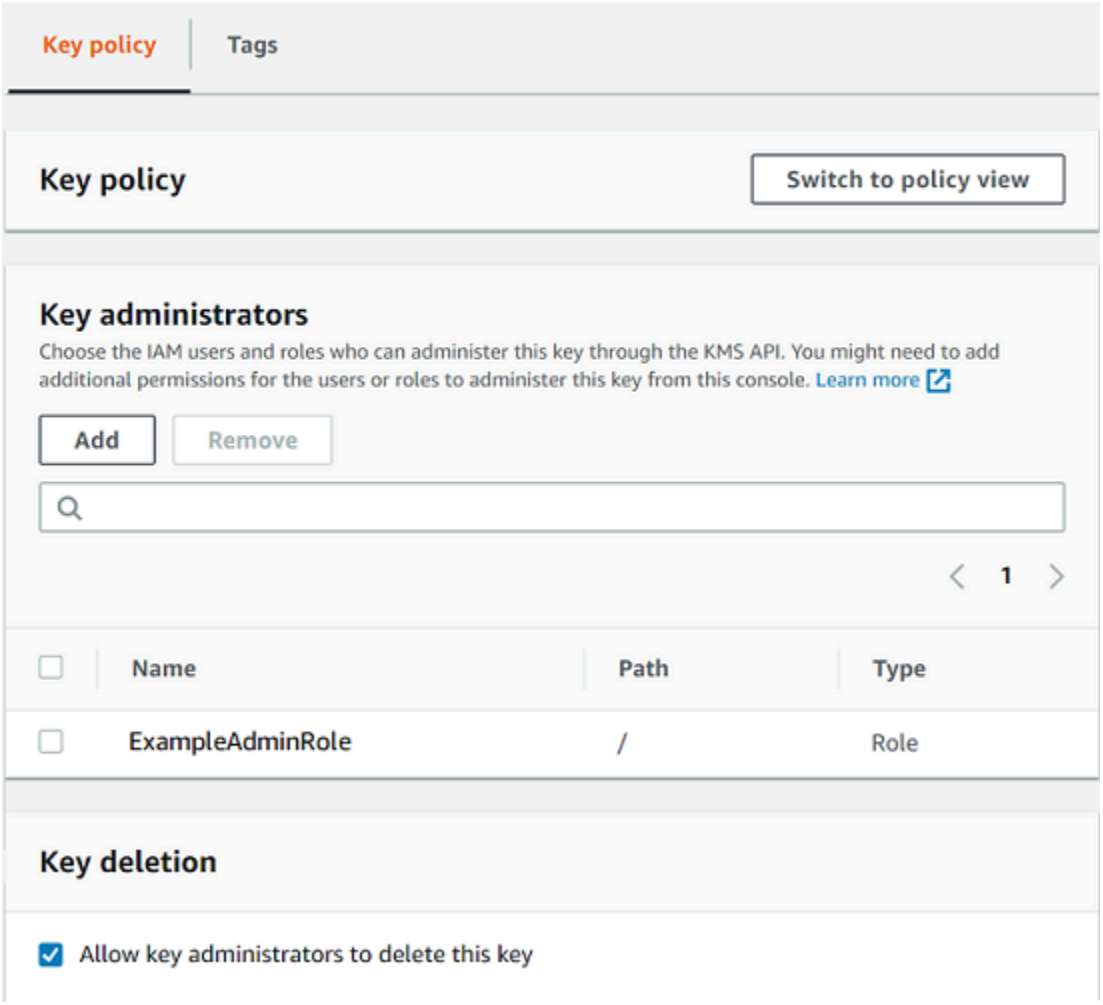
允許金鑰管理員來管理 KMS 金鑰

透過主控台建立的預設金鑰政策可讓您選擇帳戶中的 IAM 使用者和角色，並將其設為金鑰管理員。此陳述式稱為金鑰管理員陳述式。金鑰管理員有權管理 KMS 金鑰，但無權在密碼編譯操作中使用 KMS 金鑰。當您在預設檢視或政策檢視中建立 KMS 金鑰時，您可以新增 IAM 使用者和角色至金鑰管理員的清單。

> **Warning**
>
> 因為金鑰管理員具有變更金鑰政策和建立授予的許可，所以他們可以為自己和他人提供此政策中未指定的 AWS KMS 許可。
> 擁有管理標籤和別名許可的主體也可以控制對 KMS 金鑰的存取。如需詳細資訊，請參閱 AWS KMS 的 ABAC。

> **Note**
>
> IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

以下範例顯示 AWS KMS 主控台之預設檢視中的金鑰管理員陳述式。



以下是 AWS KMS 主控台之政策檢視中的範例金鑰管理員陳述式。此金鑰管理員陳述式適用於單一區域對稱加密 KMS 金鑰。

```
{
  "Sid": "Allow access for Key Administrators",
  "Effect": "Allow",
  "Principal": {"AWS":"arn:aws:iam::111122223333:role/ExampleAdminRole"},
  "Action": [
    "kms:Create*",
    "kms:Describe*",
    "kms:Enable*",
    "kms:List*",
    "kms:Put*",
    "kms:Update*",
    "kms:Revoke*",
    "kms:Disable*",
    "kms:Get*",
```

```
    "kms:Delete*",
    "kms:TagResource",
    "kms:UntagResource",
    "kms:ScheduleKeyDeletion",
    "kms:CancelKeyDeletion"
  ],
  "Resource": "*"
}
```

最常見的 KMS 金鑰 (即單一區域對稱加密 KMS 金鑰) 的預設金鑰管理員陳述式允許以下許可。如需有關每個許可的詳細資訊，請參閱 AWS KMS 權限。

當您使用 AWS KMS 主控台建立 KMS 金鑰時，主控台會將您指定的使用者和角色新增至金鑰管理員陳述式中的 `Principal` 元素。

這些許可中有許多都包含萬用字元 (*)，允許以指定動詞開頭的所有許可。因此，當 AWS KMS 新增新的 API 操作時，會自動允許金鑰管理員使用這些操作。您不需要更新您的主要政策，即可包含新的操作。如果您想要將金鑰管理員限制在一組固定的 API 操作中，您可以變更金鑰政策。

`kms:Create*`

允許 `kms:CreateAlias` 和 `kms:CreateGrant`。(`kms:CreateKey` 許可僅在 IAM 政策中有效。)

`kms:Describe*`

允許 `kms:DescribeKey` 需要 `kms:DescribeKey` 許可才能檢視 AWS Management Console中 KMS 金鑰的金鑰詳細資訊頁面。

`kms:Enable*`

允許 `kms:EnableKey` 如果是對稱加密 KMS 金鑰，其亦允許 `kms:EnableKeyRotation`。

`kms:List*`

允許 `kms:ListGrants`、`kms:ListKeyPolicies` 和 `kms:ListResourceTags`。(在 AWS Management Console 中檢視 KMS 金鑰所需的 `kms:ListAliases` 和 `kms:ListKeys` 許可，只在 IAM 政策中有效。)

`kms:Put*`

允許 `kms:PutKeyPolicy` 此許可允許金鑰管理員變更此 KMS 金鑰的金鑰政策。

kms:Update*

允許 kms:UpdateAlias 和 kms:UpdateKeyDescription。對於多區域金鑰，它允許此 KMS 金鑰上的 kms:UpdatePrimaryRegion。

kms:Revoke*

允許 kms:RevokeGrant，它允許金鑰管理員刪除授權，即使他們不是授權中的淘汰主體。

kms:Disable*

允許 kms:DisableKey 如果是對稱加密 KMS 金鑰，其亦允許 kms:DisableKeyRotation。

kms:Get*

允許 kms:GetKeyPolicy 和 kms:GetKeyRotationStatus。對於具有匯入金鑰材料的 KMS 金鑰，它允許 kms:GetParametersForImport。對於非對稱 KMS 金鑰，它允許 kms:GetPublicKey。需要 kms:GetKeyPolicy 許可才能檢視 AWS Management Console 中 KMS 金鑰的金鑰政策。

kms:Delete*

允許 kms:DeleteAlias 對於具有匯入金鑰材料的金鑰，它允許 kms:DeleteImportedKeyMaterial。kms:Delete* 許可不允許金鑰管理員刪除 KMS 金鑰 (ScheduleKeyDeletion)。

kms:TagResource

允許 kms:TagResource，可讓金鑰管理員將標籤新增至 KMS 金鑰。由於標籤也可用來控制對 KMS 金鑰的存取，此許可讓管理員可允許或拒絕對 KMS 金鑰的存取。如需詳細資訊，請參閱 AWS KMS 的 ABAC。

kms:UntagResource

允許 kms:UntagResource，可讓金鑰管理員從 KMS 金鑰刪除標籤。由於標籤可用來控制對金鑰的存取，因此此許可可讓管理員允許或拒絕對 KMS 金鑰的存取。如需詳細資訊，請參閱 AWS KMS 的 ABAC。

kms:ScheduleKeyDeletion

允許 kms:ScheduleKeyDeletion，可讓金鑰管理員刪除此 KMS 金鑰。若要刪除此許可，請清除 Allow key administrators to delete this key (允許金鑰管理員刪除此金鑰) 選項。

kms:CancelKeyDeletion

允許 kms:CancelKeyDeletion，可讓金鑰管理員取消刪除此 KMS 金鑰。若要刪除此許可，請清除 Allow key administrators to delete this key (允許金鑰管理員刪除此金鑰) 選項。

當您建立特殊用途金鑰時，AWS KMS 會將下列許可新增至預設金鑰管理員陳述式。

kms:ImportKeyMaterial

kms:ImportKeyMaterial 許可允許金鑰管理員將金鑰材料匯入 KMS 金鑰。只有當您建立不含金鑰資料的 KMS 金鑰時，此許可才會包含在金鑰政策中。

kms:ReplicateKey

kms:ReplicateKey 許可允許金鑰管理員在不同 AWS 區域中建立多區域主要金鑰的複本。只有當您建立多區域主要金鑰或複本金鑰時，此許可才會包含在金鑰政策中。

kms:UpdatePrimaryRegion

kms:UpdatePrimaryRegion 許可允許金鑰管理員將多區域複本金鑰變更為多區域主要金鑰。只有當您建立多區域主要金鑰或複本金鑰時，此許可才會包含在金鑰政策中。

允許金鑰使用者使用 KMS 金鑰

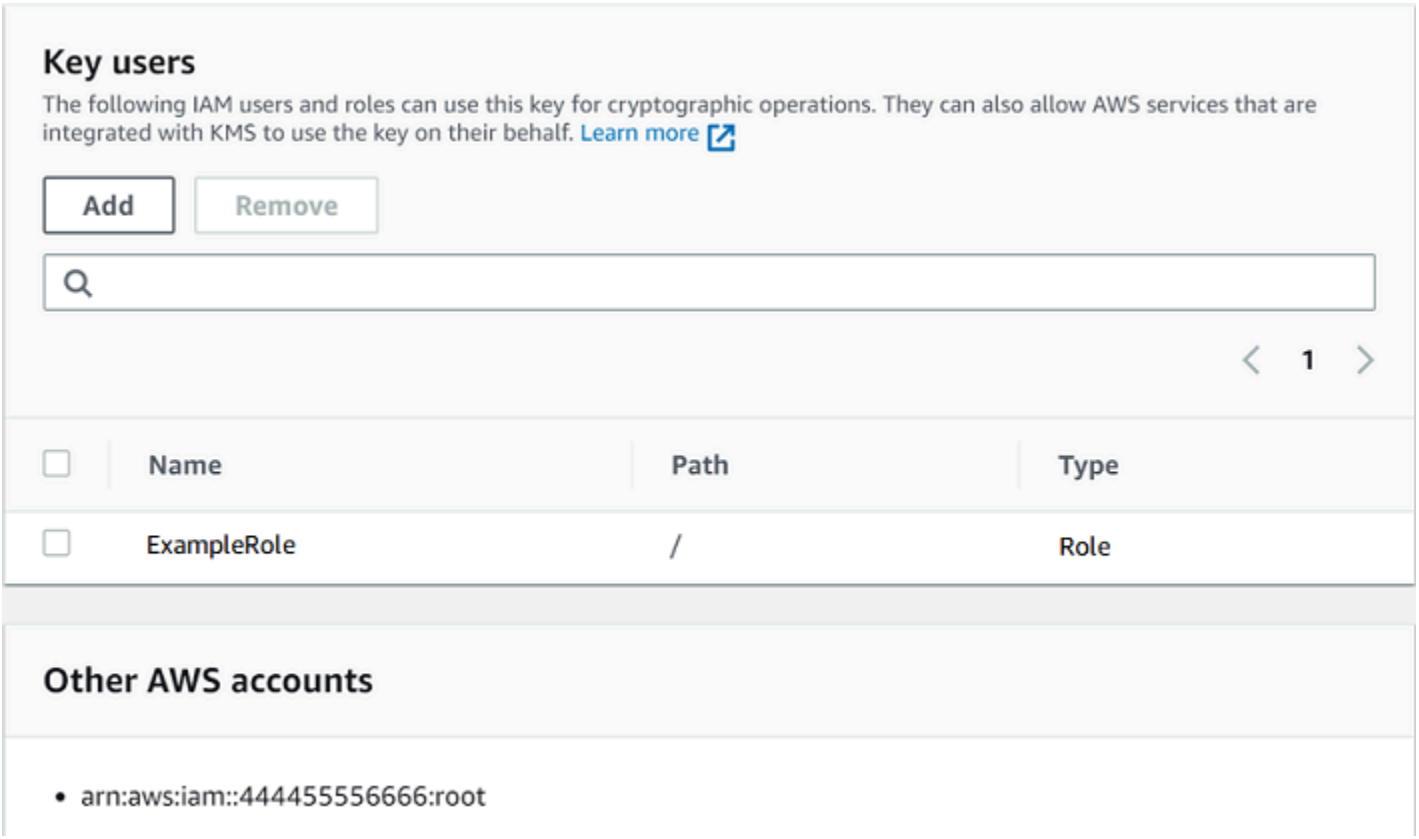
使用主控台為 KMS 金鑰建立的預設金鑰政策，您可以選擇帳戶中的 IAM 使用者和 IAM 角色以及外部 AWS 帳戶，並將其設為金鑰使用者。

主控台會將兩個政策陳述式新增至金鑰使用者的金鑰政策。

- 直接使用 KMS 金鑰 – 第一個金鑰政策陳述式給予金鑰使用者直接針對該類型 KMS 金鑰所有受支援的密碼編譯操作使用 KMS 金鑰的許可。
- 將 KMS 金鑰與 AWS 服務搭配使用 – 第二個政策陳述式為金鑰使用者提供許可，允許與 AWS KMS 整合的 AWS 服務代表他們使用 KMS 金鑰來保護資源，例如 Amazon S3 儲存貯體和 Amazon DynamoDB 資料表。

建立 KMS 金鑰時，您可以新增 IAM 使用者、IAM 角色和其他 AWS 帳戶 至金鑰使用者的清單。您也可以使用主控台的金鑰政策預設檢視來編輯清單，如下圖所示。金鑰政策的預設檢視位於金鑰詳細資訊頁面。如需允許其他 AWS 帳戶 使用者使用 KMS 金鑰的詳細資訊，請參閱 允許其他帳戶中的使用者使用 KMS 金鑰。

> **Note**
>
> IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。



單一區域對稱的預設金鑰使用者陳述式允許下列許可。如需有關每個許可的詳細資訊，請參閱 AWS KMS 權限。

當您使用 AWS KMS 主控台建立 KMS 金鑰時，主控台會將您指定的使用者和角色新增至每個金鑰使用者陳述式中的 `Principal` 元素。

```
{
  "Sid": "Allow use of the key",
  "Effect": "Allow",
  "Principal": {"AWS": [
    "arn:aws:iam::111122223333:role/ExampleRole",
    "arn:aws:iam::444455556666:root"
  ]},
  "Action": [
```

```
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*",
    "kms:DescribeKey"
  ],
  "Resource": "*"
},
{
  "Sid": "Allow attachment of persistent resources",
  "Effect": "Allow",
  "Principal": {"AWS": [
    "arn:aws:iam::111122223333:role/ExampleRole",
    "arn:aws:iam::444455556666:root"
  ]},
  "Action": [
    "kms:CreateGrant",
    "kms:ListGrants",
    "kms:RevokeGrant"
  ],
  "Resource": "*",
  "Condition": {"Bool": {"kms:GrantIsForAWSResource": true}}
}
```

允許金鑰使用者在密碼編譯操作中使用 KMS 金鑰

金鑰使用者有權直接在 KMS 金鑰支援的所有密碼編譯操作中使用 KMS 金鑰。他們也可以使用此DescribeKey作業在AWS KMS主控台中取得 KMS 金鑰的詳細資訊，或透過使用 AWS KMS API 作業取得關於 KMS 金鑰的詳細資訊。

根據預設，AWS KMS 主控台會將與下列範例內容類似的金鑰使用者陳述式，新增至預設的金鑰政策。由於政策陳述式支援不同的 API 操作，因此對稱加密 KMS 金鑰、HMAC KMS 金鑰、用於公有金鑰加密的非對稱 KMS 金鑰、用於簽署和驗證的非對稱 KMS 金鑰的政策陳述式中的動作都會略有不同。

對稱加密 KMS 金鑰

主控台會將下列陳述式新增至對稱加密 KMS 金鑰的金鑰政策。

```
{
  "Sid": "Allow use of the key",
  "Effect": "Allow",
  "Principal": {"AWS": "arn:aws:iam::111122223333:role/ExampleKeyUserRole"},
  "Action": [
```

```
    "kms:Decrypt",
    "kms:DescribeKey",
    "kms:Encrypt",
    "kms:GenerateDataKey*",
    "kms:ReEncrypt*"
  ],
  "Resource": "*"
}
```

HMAC KMS 金鑰

主控台會將下列陳述式新增至 HMAC KMS 金鑰的金鑰政策。

```
{
  "Sid": "Allow use of the key",
  "Effect": "Allow",
  "Principal": {"AWS": "arn:aws:iam::111122223333:role/ExampleKeyUserRole"},
  "Action": [
    "kms:DescribeKey",
    "kms:GenerateMac",
    "kms:VerifyMac"
  ],
  "Resource": "*"
}
```

公有金鑰加密的非對稱 KMS 金鑰

主控台會將下列陳述式新增至金鑰使用情形為 Encrypt and decrypt (加密和解密) 之非對稱 KMS 金鑰的金鑰政策。

```
{
  "Sid": "Allow use of the key",
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleKeyUserRole"
  },
  "Action": [
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:ReEncrypt*",
    "kms:DescribeKey",
    "kms:GetPublicKey"
  ],
```

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

用於簽署和驗證的非對稱 KMS 金鑰

主控台會將下列陳述式新增至金鑰使用情形為 Sign and verify (簽署和驗證) 之非對稱 KMS 金鑰的金鑰政策。

```
{
  "Sid": "Allow use of the key",
  "Effect": "Allow",
  "Principal": {"AWS": "arn:aws:iam::111122223333:role/ExampleKeyUserRole"},
  "Action": [
    "kms:DescribeKey",
    "kms:GetPublicKey",
    "kms:Sign",
    "kms:Verify"
  ],
  "Resource": "*"
}
```

這些陳述式中的動作會提供金鑰使用者下列許可。

kms:Encrypt

允許金鑰使用者使用此 KMS 金鑰加密資料。

kms:Decrypt

允許金鑰使用者使用此 KMS 金鑰解密資料。

kms:DescribeKey

允許金鑰使用者取得此 KMS 金鑰的資訊，包括其識別符、建立日期和金鑰狀態。也允許金鑰使用者在 AWS KMS 主控台中顯示 KMS 金鑰的詳細資訊。

kms:GenerateDataKey*

允許金鑰使用者為用戶端密碼編譯操作請求對稱資料金鑰或非對稱資料金鑰對。主控台使用 * 萬用字元來代表下列 API 作業的權限：GenerateDataKeyGenerateDataKeyWithoutPlaintext、GenerateDataKeyPair、和GenerateDataKeyPairWithoutPlaintext。這些許可只有在加密資料金鑰的對稱 KMS 金鑰上才有效。

公里：GenerateMac

允許金鑰使用者使用 HMAC KMS 金鑰來產生 HMAC 標籤。

公里：GetPublicKey

允許金鑰使用者下載非對稱 KMS 金鑰的公有金鑰。與您共用此公有金鑰的對象可以加密 AWS KMS 以外的資料。不過，這些加密文字只能透過呼叫 AWS KMS 中的 Decrypt 操作進行解密。

公里:ReEncrypt*

允許金鑰使用者重新加密原本使用此 KMS 金鑰加密的資料，或使用此 KMS 金鑰重新加密之前加密過的資料。此作ReEncrypt業需要存取來源和目的地 KMS 金鑰。若要完成此動作，您可以允許來源 KMS 金鑰的 `kms:ReEncryptFrom` 許可和目標 KMS 金鑰的 `kms:ReEncryptTo` 許可。但為簡單起見，主控台允許兩種 KMS 金鑰都使用 `kms:ReEncrypt*` (使用 * 萬用字元)。

kms:Sign

允許金鑰使用者使用此 KMS 金鑰簽署訊息。

kms:Verify

允許金鑰使用者使用此 KMS 金鑰驗證簽章。

公里：VerifyMac

允許金鑰使用者使用 HMAC KMS 金鑰來驗證 HMAC 標籤。

允許金鑰使用者使用 KMS 金鑰搭配 AWS 服務

主控台中的預設金鑰政策也會向金鑰使用者授予所需的許可，以保護使用授權的 AWS 服務中的資料。AWS 服務通常會使用授權來取得使用 KMS 金鑰的特定和有限權限。

此金鑰政策陳述式允許金鑰使用者建立、檢視和撤銷對 KMS 金鑰的授權，但只有在授權操作請求是來自與 AWS KMS 整合的 AWS 服務時才有效。kms: GrantIsFor AWSResource 原則條件不允許使用者直接呼叫這些授權作業。在金鑰使用者允許時，AWS 服務可以代表使用者建立授權，允許服務使用 KMS 金鑰來保護使用者的資料。

金鑰使用者需要這些授權許可，才能使用其 KMS 金鑰與整合服務，但這些許可仍有不足。金鑰使用者也需要有許可才能使用整合服務。如需給予使用者許可來存取與 AWS KMS 整合之 AWS 服務的詳細資訊，請參閱整合服務的文件。

```
{
  "Sid": "Allow attachment of persistent resources",
  "Effect": "Allow",
```

```
  "Principal": {"AWS": "arn:aws:iam::111122223333:role/ExampleKeyUserRole"},
  "Action": [
    "kms:CreateGrant",
    "kms:ListGrants",
    "kms:RevokeGrant"
  ],
  "Resource": "*",
  "Condition": {"Bool": {"kms:GrantIsForAWSResource": true}}
}
```

例如，金鑰使用者可以透過下列方式在 KMS 金鑰上使用這些許可。

- 將此 KMS 金鑰搭配 Amazon Elastic Block Store (Amazon EBS) 和 Amazon Elastic Compute Cloud (Amazon EC2) 使用，將加密的 EBS 磁碟區連接到 EC2 執行個體。金鑰使用者隱含提供 Amazon EC2 使用 KMS 金鑰將加密磁碟區連接到執行個體的許可。如需詳細資訊，請參閱 Amazon Elastic Block Store (Amazon EBS) 如何使用 AWS KMS。
- 將此 KMS 金鑰搭配 Amazon Redshift 使用，以啟動加密叢集。金鑰使用者隱含提供 Amazon Redshift 使用 KMS 金鑰啟動加密叢集並建立加密快照的許可。如需詳細資訊，請參閱 Amazon Redshift 如何使用 AWS KMS。
- 使用此 KMS 金鑰搭配其他與 AWS KMS 整合且使用授權的 AWS 服務，來建立、管理或使用加密資源搭配這些服務。

預設金鑰政策允許金鑰使用者將其許可授予所有使用授權的整合服務。然而，您可以建立自訂金鑰政策，將許可限制為指定的 AWS 服務。如需詳細資訊，請參閱 公里：ViaService 條件金鑰。

檢視金鑰政策

您可以使用 AWS KMS API AWS 受管金鑰中的或GetKeyPolicy作業，檢視AWS KMS客戶管理金鑰或您帳戶中某個金鑰的金鑰政策。AWS Management Console您無法使用這些技術檢視不同 AWS 帳戶中 KMS 金鑰的金鑰政策。

若要進一步了解 AWS KMS 金鑰政策，請參閱AWS KMS 中的金鑰政策。若要了解如何判斷哪些使用者和角色可以存取 KMS 金鑰，請參閱 the section called “判斷存取權”。

主題

- 檢視金鑰政策 (主控台)
- 檢視金鑰政策 (AWS KMS API)

檢視金鑰政策 (主控台)

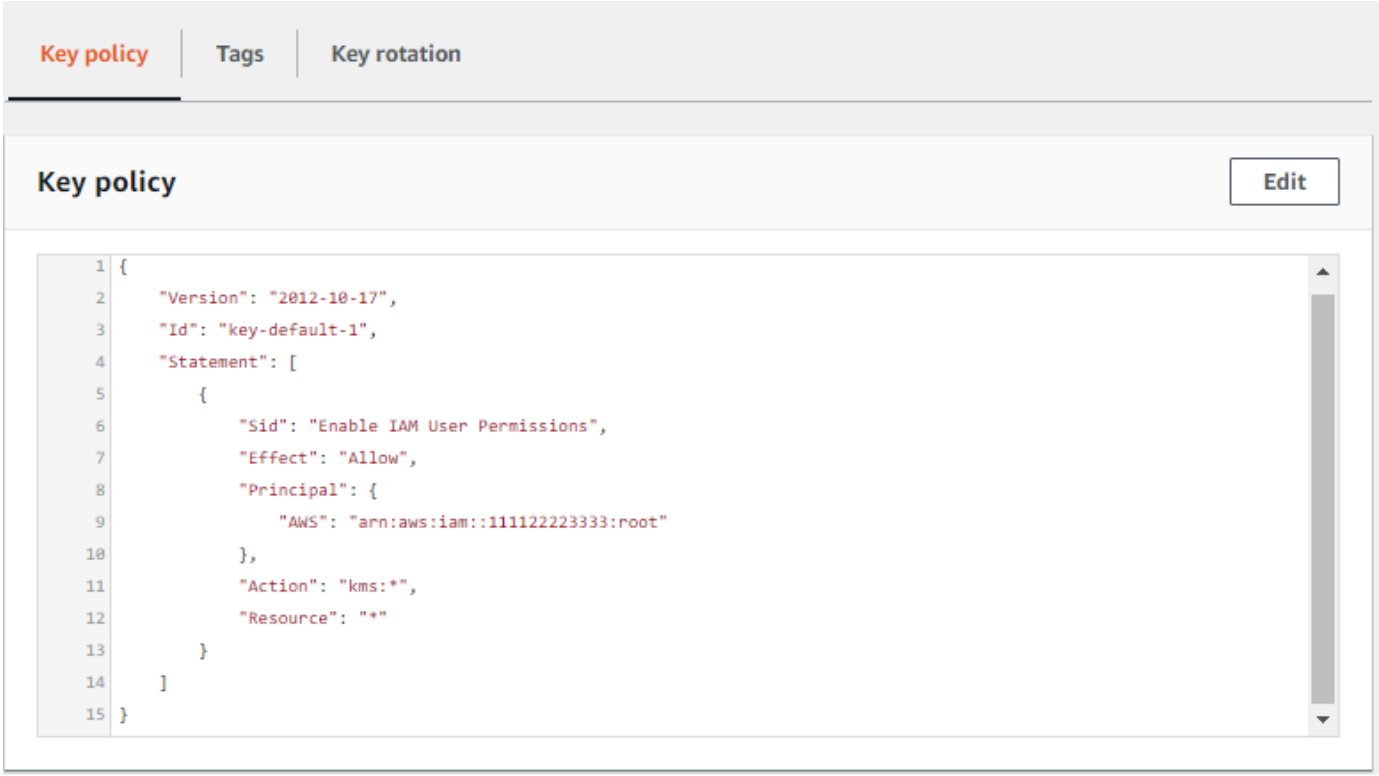
授權的使用者可以在 AWS Management Console 的 Key policy (金鑰政策) 索引標籤上檢視 AWS 受管金鑰 或客戶受管金鑰的金鑰政策。

若要檢視中 KMS 金鑰的金鑰原則AWS Management Console，您必須具有 kms: ListAliases、kms: DescribeKey 和 kms: GetKeyPolicy 權限。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 若要檢視 AWS 為您建立及管理之帳戶中的金鑰，請在導覽窗格中選擇 AWS 受管金鑰。若要檢視您所建立及管理帳戶中的金鑰，請在導覽窗格中選擇Customer managed keys (客戶受管金鑰)。
4. 在 KMS 金鑰清單中，選擇您要檢查之 KMS 金鑰的別名或金鑰 ID。
5. 選擇 Key policy (金鑰政策) 標籤。

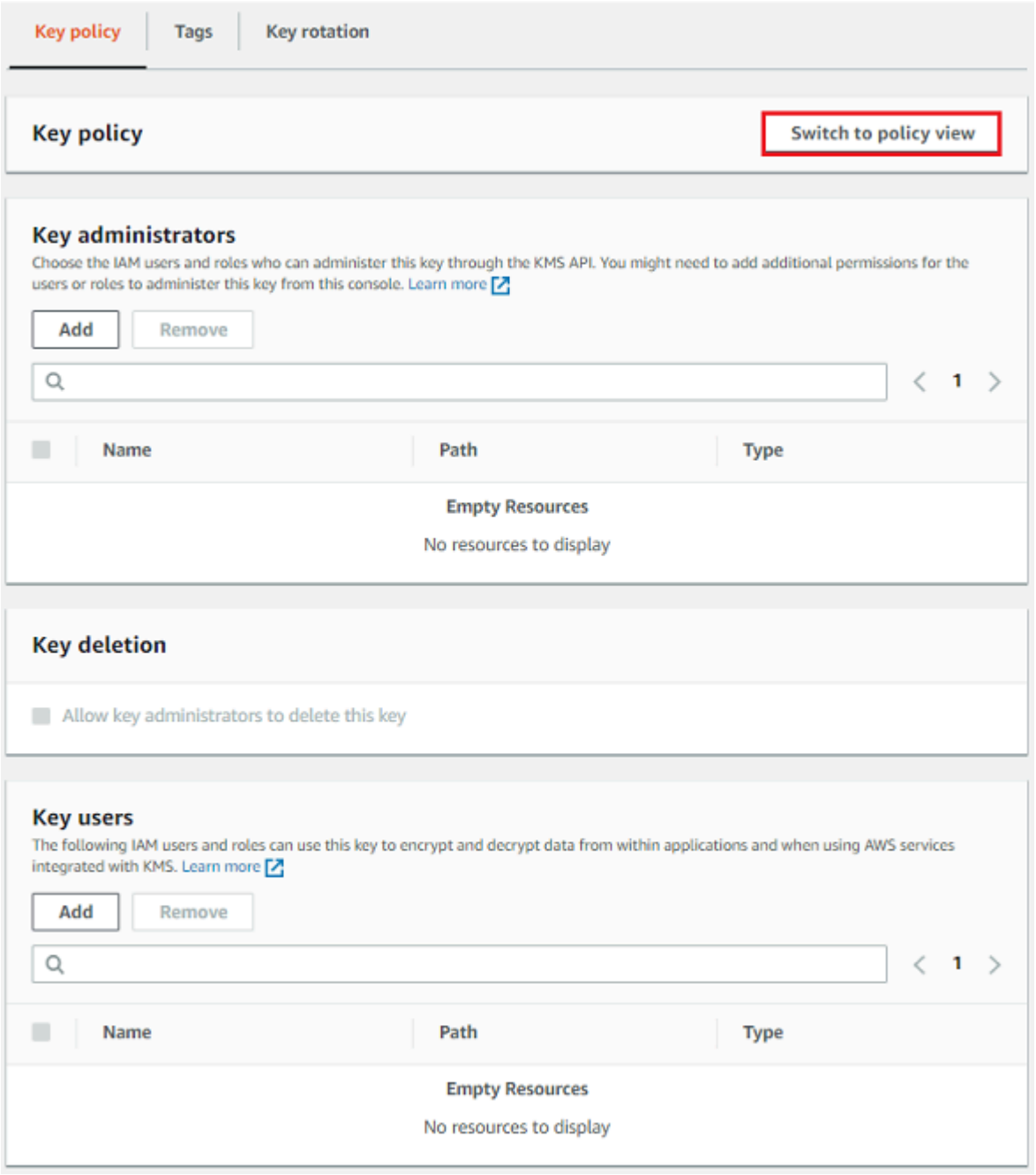
   在 Key policy (金鑰政策) 索引標籤中，您可能會看到金鑰政策文件。這是「政策檢視」。在金鑰政策陳述式中，您可以看到由金鑰政策授予 KMS 金鑰存取權的主體，也會看到他們可以執行的動作。

   下列範例顯示預設金鑰政策的政策檢視。



或者，如果在 AWS Management Console 中建立了 KMS 金鑰，您將在 Key administrators (金鑰管理員)、Key deletion (金鑰刪除) 和 Key Users (金鑰使用者) 區段中看到預設檢視。若要查看金鑰政策文件，請選擇 Switch to policy view (切換至政策檢視)。

下列範例顯示預設金鑰政策的預設檢視。



檢視金鑰政策 (AWS KMS API)

若要取得 KMS 金鑰的金鑰原則AWS 帳戶，請使用 AWS KMS API 中的GetKeyPolicy作業。您無法使用此操作檢視不同帳戶中的金鑰政策。

下列範例會使用 AWS Command Line Interface (AWS CLI) 中的get-key-policy命令，但您可以使用任何 AWS SDK 來發出此要求。

請注意，雖然 `default` 是唯一的有效值，但 `PolicyName` 參數是必要的。此外，這個命令請求使用較易檢視的文字輸出，而不是 JSON。

執行此命令之前，請將範例金鑰 ID 更換成您帳戶的有效 ID。

```
$ aws kms get-key-policy --key-id 1234abcd-12ab-34cd-56ef-1234567890ab --policy-name
 default --output text
```

回應應該類似下面其中一個項目，會傳回預設的金鑰政策。

```
{
  "Version" : "2012-10-17",
  "Id" : "key-consolepolicy-3",
  "Statement" : [ {
    "Sid" : "Enable IAM User Permissions",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "arn:aws:iam::111122223333:root"
    },
    "Action" : "kms:*",
    "Resource" : "*"
  } ]
}
```

變更金鑰政策

您可以使用AWS Management Console或PutKeyPolicy作業變更中 KMS 金鑰的AWS 帳戶金鑰原則。您無法使用這些技術變更不同 AWS 帳戶 中 KMS 金鑰的金鑰政策。

變更金鑰政策時，請注意以下規則：

- 您可以檢視 AWS 受管金鑰 或客戶受管金鑰的金鑰政策，但只能變更客戶受管金鑰的金鑰政策。AWS 受管金鑰 的政策是由您帳戶中建立 KMS 金鑰 的 AWS 服務所建立和管理。您無法檢視或變更 AWS 擁有的金鑰 的金鑰政策。
- 您可以在金鑰政策中新增或移除 IAM 使用者、IAM 角色和 AWS 帳戶，以及變更這些主體獲允許或拒絕的動作。如需在金鑰政策中指定主體和許可之方式的詳細資訊，請參閱金鑰政策。
- 您不能新增 IAM 群組到金鑰政策，但您可以新增多個 IAM 使用者和 IAM 角色。如需詳細資訊，請參閱 允許多個 IAM 主體存取 KMS 金鑰。

- 如果您新增外部 AWS 帳戶 到金鑰政策，您也必須在外部帳戶中使用 IAM 政策來提供許可給這些帳戶中的 IAM 使用者、群組或角色。如需詳細資訊，請參閱 允許其他帳戶中的使用者使用 KMS 金鑰。
- 產生的金鑰政策文件不能超過 32 KB (32,768 位元組)。

主題

如何變更金鑰政策

變更金鑰政策有三種方法，如以下章節所述。

主題

使用 AWS Management Console 預設檢視

您可以使用主控台，以稱為預設檢視的圖形界面來變更金鑰政策。

如果以下步驟不符合您在主控台中看到的步驟，可能表示此金鑰政策不是使用主控台所建立的。也可能表示主控台的預設檢視不支援修改後的金鑰政策。在這種情況下，請遵循使用 AWS Management Console 政策檢視或使用 AWS KMS API 中的步驟進行。

1. 請依 檢視金鑰政策 (主控台) 所述，檢視客戶受管金鑰的金鑰政策。(您無法變更 AWS 受管金鑰 的金鑰政策。)
2. 決定進行哪些變更。

   - 若要新增或移除金鑰管理員，以及允許或不允許金鑰管理員刪除 KMS 金鑰，請使用頁面上的 Key administrators (金鑰管理員) 區段中的控制項。金鑰管理員負責管理 KMS 金鑰，包括啟用和停用它、設定金鑰政策，以及啟用金鑰輪換。
   - 若要新增或移除金鑰使用者，以及允許或不允許 AWS 帳戶 使用 KMS 金鑰，請使用頁面上的 Key users (金鑰使用者) 區段中的控制項。金鑰使用者可以在密碼編譯操作中使用 KMS 金鑰，例如加密、解密、重新加密和產生資料金鑰。

使用 AWS Management Console 政策檢視

您可以使用主控台的「政策檢視」來變更金鑰政策文件。

1. 請依 檢視金鑰政策 (主控台) 所述，檢視客戶受管 KMS 金鑰的金鑰政策。(您無法變更 AWS 受管金鑰 的金鑰政策。)
2. 在 金鑰政策 區段中，選擇 切換為政策檢視。
3. 編輯金鑰政策文件，然後選擇 Save changes (儲存變更)。

使用 AWS KMS API

您可以使用此PutKeyPolicy作業變更您的 KMS 金鑰的金鑰原則AWS 帳戶。您無法對其他 AWS 帳戶中的 KMS 金鑰使用此 API。

1. 使用此GetKeyPolicy作業取得現有的金鑰原則文件，然後將金鑰原則文件儲存至檔案。如需多種程式設計語言的範例程式碼，請參閱取得金鑰政策。
2. 在您偏好的文字編輯器中開啟金鑰政策文件、編輯金鑰政策文件，然後儲存檔案。
3. 使用此PutKeyPolicy作業將更新的金鑰原則文件套用至 KMS 金鑰。如需多種程式設計語言的範例程式碼，請參閱設定金鑰政策。

如需將金鑰原則從一個 KMS 金鑰複製到另一個 KMS 金鑰的GetKeyPolicy 範例，請參閱AWS CLI命令參考中的範例。

允許多個 IAM 主體存取 KMS 金鑰

IAM 群組在金鑰政策中不是有效的主體。若要允許多個使用者和角色存取 KMS 金鑰，請執行下列其中一項操作：

- 使用 IAM 角色作為金鑰政策中的主體。多個授權使用者可以視需要擔任該角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 角色。

  雖然您可以在金鑰政策中列出多個 IAM 使用者，但不建議使用此做法，因為這會要求您在每次授權使用者清單變更時更新金鑰政策。此外，IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。
- 使用 IAM 政策，將許可授予給 IAM 群組。為此，請確保金鑰政策包含的陳述式可啟用 IAM 政策以允許存取 KMS 金鑰，建立一個允許存取 KMS 金鑰的 IAM 政策，然後將該政策連接到包含該授權 IAM 使用者的 IAM 群組。透過此方式，您不需要在授權的使用者清單變更時，隨之變更任何政策。

相反地，您只需要從適當的 IAM 群組新增或移除這些使用者。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 使用者群組

如需 AWS KMS 金鑰政策與 IAM 政策如何同時運作的相關資訊，請參閱 對金鑰存取進行故障診斷。

金鑰政策中 AWS 服務的許可

許多 AWS 服務使用 AWS KMS keys 來保護他們所管理的資源。當服務使用 AWS 擁有的金鑰 或 AWS 受管金鑰 時，服務會建立並維護這些 KMS 金鑰的金鑰政策。

但是，當您透過 AWS 服務使用客戶受管金鑰時，您需要設定並維護金鑰政策。該金鑰政策必須允許服務具有代表您保護資源所需的最低許可。建議您遵循最低權限原則：僅授予服務所需的許可。您可以藉由了解服務需要哪些權限並使用 AWS 全域條件金鑰和 AWS KMS 條件金鑰來調整許可，有效地完成這個操作。

若要尋找服務對客戶受管金鑰所需的許可，請參閱服務的加密文件。例如，對於 Amazon Elastic Block Store (Amazon EBS) 所需的許可，請參閱《Amazon EC2 Linux 執行個體使用者指南》和《Amazon EC2 Windows 執行個體使用者指南》中的 IAM 使用者的許可。對於 Secrets Manager 所需的許可，請參閱《AWS Secrets Manager 使用者指南》中的授權使用 KMS 金鑰。

實作最低權限的許可

當您給予 AWS 服務許可以便使用 KMS 金鑰時，請確定許可僅針對服務必須代表您存取的資源有效。此最低權限策略有助於防止當請求在 AWS 服務之間傳遞時，未經授權使用 KMS 金鑰。

若要實作最低權限策略，推薦使用 AWS KMS 加密內容條件金鑰和全域來源 ARN 或來源帳戶條件金鑰。

使用加密內容條件金鑰

使用 AWS KMS 資源時實作最低權限許可的最有效方法是將 kms:EncryptionContext:context-key 或 kms:EncryptionContextKeys 條件金鑰包含在政策中，該政策會允許主體呼叫 AWS KMS 密碼編譯操作。這些條件金鑰特別有效，因為它們會將許可與在資源加密時繫結至加密文字的加密內容建立關聯。

只有當策略陳述式中的動作為或需要EncryptionContext參數的AWS KMS對稱加密作業 (例如，CreateGrant或「解密」等作業) 時，才使用加密內容條件金鑰。GenerateDataKey(如需受支援操作的清單，請參閱 kms:EncryptionContext:context-key 或 kms:EncryptionContextKeys。) 如果您使用這些條件索引鍵來允許其他作業，例如 DescribeKey，將會拒絕權限。

將值設定為服務在加密資源時使用的加密內容。此資訊通常可在服務文件的「安全」一章中取得。例如，AWS Proton 的加密內容會識別 AWS Proton 資源及其相關聯範本。AWS Secrets Manager 加密內容會識別秘密及其版本。Amazon Location 的加密內容會識別追蹤器或收集。

下列範例金鑰政策陳述式允許 Amazon Location Service 代表授權使用者建立授予。此原則陳述式會使用 kms: ViaService、kms: 和kms:EncryptionContext:*context-key*條件金鑰將權限繫結至特定追蹤器資源CallerAccount，以限制權限。

```
{
  "Sid": "Allow Amazon Location to create grants on behalf of authorized users",
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/LocationTeam"
  },
  "Action": "kms:CreateGrant",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:ViaService": "geo.us-west-2.amazonaws.com",
      "kms:CallerAccount": "111122223333",
      "kms:EncryptionContext:aws:geo:arn": "arn:aws:geo:us-west-2:111122223333:tracker/
SAMPLE-Tracker"
    }
  }
}
```

使用 aws:SourceArn 或 aws:SourceAccount 條件金鑰

當主要政策陳述式中的主體是 AWS 服務主體時，強烈建議您除了 kms:EncryptionContext:*context-key* 條件金鑰之外，再使用 aws:SourceArn 或 aws:SourceAccount 全域條件金鑰。只有在請求從另一個 AWS 服務來到 AWS KMS 中時，ARN 和帳戶值會包含在授權內容中。這個條件組合會實作最低權限許可，並避免潛在的混淆代理人案例。服務主體通常不會做為金鑰政策中的主體，但是有些 AWS 服務 (例如 AWS CloudTrail) 需要它。

若要使用 aws:SourceArn 或 aws:SourceAccount 全域條件金鑰，請將值設定為要加密之資源的 Amazon Resource Name (ARN) 或帳戶。例如，在給予 AWS CloudTrail 許可以加密追蹤的金鑰政策陳述式中，將 aws:SourceArn 的值設定為追蹤的 ARN。盡可能使用 aws:SourceArn，這更為具體。將值設定為 ARN 或具有萬用字元的 ARN 模式。如果您不知道資源的 ARN，請改為使用 aws:SourceAccount。

> **Note**
>
> 若資源 ARN 包含 AWS KMS 金鑰政策中不允許的字元，則您便不能將該資源 ARN 用於 `aws:SourceArn` 條件金鑰的值中。請改用 `aws:SourceAccount` 條件金鑰。如需有關金鑰政策文件規則的詳細資訊，請參閱金鑰政策格式。

在下列範例金鑰政策中，取得許可的主體是 AWS CloudTrail 服務主體 `cloudtrail.amazonaws.com`。為了實作最低權限，此政策會使用 `aws:SourceArn` 和 `kms:EncryptionContext:`*`context-key`* 條件金鑰。原則陳述式 CloudTrail 允許使用 KMS 金鑰來產生用來加密追蹤的資料金鑰。獨立評估 `aws:SourceArn` 和 `kms:EncryptionContext:`*`context-key`* 條件。針對指定操作使用 KMS 金鑰的任何請求都必須滿足這兩個條件。

若要將服務的許可限制為範例帳戶 (111122223333) 和 `us-west-2` 區域中的 `finance` 追蹤，此政策陳述式會將 `aws:SourceArn` 條件金鑰設定為特定追蹤的 ARN。條件陳述式會使用ArnEquals運算子來確保相符時，ARN 中的每個項目都會獨立計算。此範例也會使用 `kms:EncryptionContext:`*`context-key`* 條件金鑰來限制特定帳戶和區域中追蹤的許可。

在使用此金鑰政策之前，請將範例帳戶 ID、區域和追蹤名稱取代為您帳戶的有效值。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow CloudTrail to encrypt logs",
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudtrail.amazonaws.com"
      },
      "Action": "kms:GenerateDataKey",
      "Resource": "*",
      "Condition": {
        "ArnEquals": {
          "aws:SourceArn": [
            "arn:aws:cloudtrail:us-west-2:111122223333:trail/finance"
          ]
        },
        "StringLike": {
          "kms:EncryptionContext:aws:cloudtrail:arn": [
            "arn:aws:cloudtrail:*:111122223333:trail/*"
```

```
            ]
          }
        }
      }
    ]
}
```

將 IAM 政策與 AWS KMS 搭配使用

您可以使用 IAM 政策以及金鑰政策、授予和 VPC 端點政策，以控制對 AWS KMS 中 AWS KMS keys 的存取。

> **Note**
>
> 若要使用 IAM 政策來控制 KMS 金鑰的存取，KMS 金鑰的金鑰政策必須授予帳戶使用 IAM 政策的許可。特別是，金鑰政策必須包含啟用 IAM 政策的政策陳述式。
> 本節說明如何使用 IAM 政策來控制 AWS KMS 操作的存取。如需 IAM 許可的一般資訊，請參閱《IAM 使用者指南》。

所有 KMS 金鑰都必須有金鑰政策。IAM 政策是選用的。若要使用 IAM 政策來控制 KMS 金鑰的存取，KMS 金鑰的金鑰政策必須授予帳戶使用 IAM 政策的許可。特別是，金鑰政策必須包含啟用 IAM 政策的政策陳述式。

IAM 政策可以控制對任何 AWS KMS 操作的存取。與主要政策不同，IAM 政策可以控制對多個 KMS 金鑰的存取，並提供數個相關 AWS 服務的操作許可。但 IAM 政策對於控制無法由金鑰政策控制的作業存取特別有用 CreateKey，因為它們不涉及任何特定的 KMS 金鑰。

如果透過 Amazon Virtual Private Cloud (Amazon VPC) 端點存取 AWS KMS，則您也可以使用 VPC 端點政策限制在使用端點時對 AWS KMS 資源的存取。例如，使用 VPC 端點時，您可能只允許 AWS 帳戶 中的主體來存取您的客戶受管金鑰。如需詳細資訊，請參閱 控制對 VPC 端點的存取。

如需撰寫及格式化 JSON 政策文件的說明，請參閱《IAM 使用者指南》中的 IAM JSON 政策參考。

主題

- IAM 政策概觀
- IAM 政策的最佳實務
- 在 IAM 政策陳述式中指定 KMS 金鑰

- 使用 AWS KMS 主控台所需的許可
- 進階使用者的 AWS 受管政策
- IAM 政策範例

IAM 政策概觀

您可以下方式來使用 IAM 政策：

- 連接許可政策到聯合或跨帳戶許可的角色 – 您可以連接 IAM 政策到 IAM 角色來啟用聯合身分、允許跨帳戶許可，或提供應用程式在 EC2 執行個體上執行的許可。如需 IAM 角色各種使用案例的詳細資訊，請參閱《IAM 使用者指南》中的 IAM 角色。
- 連接許可政策到使用者或群組 – 您可以連接允許使用者或使用者群組呼叫 AWS KMS 操作的政策。不過，IAM 最佳實務建議您盡可能使用具有臨時憑證的身分，例如 IAM 角色。

以下範例顯示具有 AWS KMS 許可的 IAM 政策。此政策允許其連接的 IAM 身分列出所有 KMS 金鑰和別名。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "kms:ListKeys",
      "kms:ListAliases"
    ],
    "Resource": "*"
  }
}
```

與所有 IAM 政策一樣，此政策沒有 `Principal` 元素。將 IAM 政策連接到 IAM 身分時，該身分會取得政策中指定的許可。

如需顯示所有 AWS KMS API 動作及其適用的各項資源表格，請參閱許可參考。

IAM 政策的最佳實務

保護對 AWS KMS keys 的存取對所有 AWS 資源的安全非常重要。KMS 金鑰可用來保護您的 AWS 帳戶 中許多最敏感的資源。花時間設計金鑰政策、IAM 政策、授予和 VPC 端點政策，可控制對 KMS 金鑰的存取。

在控制 KMS 金鑰存取的 IAM 政策陳述式中，使用最低權限政策。僅為 IAM 主體提供他們必須使用或管理之 KMS 金鑰所需的許可。

下列最佳實務適用於控制 AWS KMS 金鑰和別名存取的 IAM 政策。如需一般 IAM 政策最佳實務指南，請參閱《IAM 使用者指南》中的 IAM 安全最佳實務。

使用金鑰政策

盡可能在影響一個 KMS 金鑰的金鑰政策中提供許可，而不是在可套用至許多 KMS 金鑰的 IAM 政策中提供許可，包括其他 AWS 帳戶 中的這些政策。這對於像 kms: PutKeyPolicy 和 kms: 這樣的敏感權限尤為重要，ScheduleKeyDeletion但對於確定數據如何保護的加密操作也很重要。

限制 CreateKey 權限

僅授予建立金鑰 (kms: CreateKey) 權限給需要金鑰的主體。建立 KMS 金鑰的主體也會設定其金鑰政策，讓他們可以授予自己和其他人使用和管理其所建立之 KMS 金鑰的許可。當您允許此許可時，請考慮使用政策條件對其進行限制。例如，您可以使用 kms: KeySpec 條件來限制對稱加密 KMS 金鑰的權限。

在 IAM 政策中指定 KMS 金鑰

最佳實務是在政策陳述式的 `Resource` 元素中指定許可套用至其中之每個 KMS 金鑰的金鑰 ARN。此實務會限制主體所需之 KMS 金鑰的許可。例如，此 `Resource` 元素只會列出主體需要使用的 KMS 金鑰。

```
"Resource": [
    "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321"
]
```

不能指定 KMS 金鑰時，請使用 `Resource` 值，限制存取可信任 AWS 帳戶 和區域中的 KMS 金鑰，例如 `arn:aws:kms:`*`region`*`:`*`account`*`:key/*`。或在可信任 AWS 帳戶 中所有區域中限制對 KMS 金鑰的存取，例如 `arn:aws:kms:*:`*`account`*`:key/*`。

您無法使用金鑰 ID、別名名稱，或別名 ARN 來代表 IAM 政策 `Resource` 欄位中的 KMS 金鑰。如果您指定別名 ARN，政策會套用至別名，而不是 KMS 金鑰。如需別名 IAM 政策的資訊，請參閱控制對別名的存取。

避免 IAM 政策中的 "Resource": "*"

明智地使用萬用字元 (*)。在金鑰政策中，`Resource` 元素中的萬用字元代表金鑰政策所連接的 KMS 金鑰。但是在 IAM 政策中，`Resource` 元素 (`"Resource": "*"`) 中的單獨萬用字元會將許

可套用至所有主體帳戶擁有使用許可之 AWS 帳戶 中的所有 KMS 金鑰。這可能包括其他 AWS 帳戶 中的 KMS 金鑰，以及主體帳戶中的 KMS 金鑰。

例如，若要在另一個 AWS 帳戶 中使用 KMS 金鑰，則主體需要來自外部帳戶中 KMS 金鑰的金鑰政策以及其自己帳戶中的 IAM 政策的許可。假設某個任意帳戶針對其 KMS 金鑰向您的AWS 帳戶 kms:Decrypt 提供許可。如果是這樣，您帳戶中為所有 KMS 金鑰 (`"Resource": "*"`) 提供角色 `kms:Decrypt` 許可的 IAM 政策將滿足需求的 IAM 部分。因此，擔任該角色的主體現在可以使用不可信任帳戶中的 KMS 金鑰來解密加密文字。這兩個帳戶的 CloudTrail 記錄中都會顯示其作業項目。

特別是，避免在允許下列 API 操作的政策陳述式中使用 `"Resource": "*"`。可以對其他 AWS 帳戶 中的 KMS 金鑰呼叫這些操作。

- DescribeKey
- GetKeyRotationStatus
- 加密操作 (加密，解密，，GenerateDataKey，GenerateDataKeyPair，GenerateDataKeyWithoutPlaintext，，GenerateDa名 ReEncrypt，驗證)
- CreateGrant, ListGrants, ListRetirableGrants, RetireGrant, RevokeGrant

何時使用 "Resource": "*"

在 IAM 政策中，僅對需要的許可在 `Resource` 元素中使用萬用字元。只有下列許可需要 `"Resource": "*"` 元素。

- 公里：CreateKey
- 公里：GenerateRandom
- 公里：ListAliases
- 公里：ListKeys
- 自訂金鑰存放區的權限，例如 kms: CreateCustomKeyStore 和 kms: ConnectCustomKeyStore。

> **Note**
>
> 別名操作的權限 (公里：CreateAlias，公里：UpdateAlias，公里：DeleteAlias) 必須附加到別名和 KMS 密鑰。您可以使用 IAM 政策中的 `"Resource": "*"` 來表示別名和 KMS 金鑰，或在 `Resource` 元素中指定別名和 KMS 金鑰。如需範例，請參閱 控制對別名的存取。

本主題中的範例提供有關設計 KMS 金鑰之 IAM 政策的詳細資訊和指引。對於一般 AWS KMS 最佳實務指引，請參閱 AWS Key Management Service 最佳實務 (PDF)。對於所有 AWS 資源的 IAM 最佳實務，請參閱《IAM 使用者指南》中的 IAM 安全最佳實務。

在 IAM 政策陳述式中指定 KMS 金鑰

您可以使用 IAM 政策，允許主體使用或管理 KMS 金鑰。在政策陳述式的 `Resource` 元素中指定 KMS 金鑰。

- 若要在 IAM 政策陳述式中指定 KMS 金鑰，您必須使用其金鑰 ARN。您不能使用金鑰 ID、別名名稱或別名 ARN 來識別 IAM 政策陳述式中的 KMS 金鑰。

  例如："`Resource": "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab`"

  若要根據其別名控制 KMS 金鑰的存取權，請使用 kms: RequestAlias 或 kms: ResourceAliases 條件金鑰。如需詳細資訊，請參閱 AWS KMS 的 ABAC。

  只有在控制別名作業 (例如、或 DeleteAlias) 存取的原則陳述式中，才使用別名 ARN 作為CreateAlias資源。UpdateAlias如需詳細資訊，請參閱 控制對別名的存取。
- 若要在帳戶和區域中指定多個 KMS 金鑰，請在金鑰 ARN 的區域或資源 ID 位置中使用萬用字元 (*)。

  例如，若要在帳戶的美國西部 (奧勒岡) 區域中指定所有 KMS 金鑰，請使用 "`Resource": "arn:aws:kms:us-west-2:111122223333:key/*`"。若要在帳戶的所有區域中指定所有 KMS 金鑰，請使用 "`Resource": "arn:aws:kms:*:111122223333:key/*`"。
- 若要代表所有 KMS 金鑰，請單獨使用萬用字元 ("*")。對於不使用任何特定 KMS 金鑰 (亦即、、和) 的作業 CreateKeyGenerateRandomListAliases，請使用此格式ListKeys。

在撰寫政策陳述式時，最佳實務是僅指定主體需要使用的 KMS 金鑰，而不是為其授予所有 KMS 金鑰的存取權。

例如，下列 IAM 政策陳述式允許主體呼叫政策陳述式`Resource`元素中列出的 KMS 金鑰解密作業。DescribeKeyGenerateDataKey透過金鑰 ARN 指定 KMS 金鑰 (這是最佳實務) 可確保許可僅限於指定的 KMS 金鑰。

```
{
```

```
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "kms:DescribeKey",
      "kms:GenerateDataKey",
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
      "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321"
    ]
  }
}
```

若要將許可套用至特定可信任 AWS 帳戶 中的所有 KMS 金鑰，您可以在區域和金鑰 ID 位置中使用萬用字元 (*)。例如，下列政策陳述式可讓主體在兩個可信任範例帳戶中的所有 KMS 金鑰上呼叫指定的操作。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "kms:DescribeKey",
      "kms:GenerateDataKey",
      "kms:GenerateDataKeyPair"
    ],
    "Resource": [
      "arn:aws:kms:*:111122223333:key/*",
      "arn:aws:kms:*:444455556666:key/*"
    ]
  }
}
```

您也可以在 Resource 元素中單獨使用萬用字元 ("*")。因為它允許存取帳戶具有使用許可的所有 KMS 金鑰，所以建議主要用於沒有特定 KMS 金鑰的操作，以及 Deny 陳述式。您也可以在只允許較不敏感之唯讀操作的政策陳述式中使用它。若要判斷 AWS KMS 操作是否涉及特定 KMS 金鑰，請尋找 the section called “許可參考” 資料表資源資料欄中的 KMS 金鑰值。

例如，下列政策陳述式使用 Deny 效果，以禁止主體在任何 KMS 金鑰上使用指定的操作。它使用 Resource 元素中的萬用字元來代表所有 KMS 金鑰。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Deny",
    "Action": [
      "kms:CreateKey",
      "kms:PutKeyPolicy",
      "kms:CreateGrant",
      "kms:ScheduleKeyDeletion"
    ],
    "Resource": "*"
  }
}
```

下列政策陳述式僅使用萬用字元來代表所有 KMS 金鑰。但它只允許不太敏感的唯讀操作和不適用於任何特定 KMS 金鑰的操作。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "kms:CreateKey",
      "kms:ListKeys",
      "kms:ListAliases",
      "kms:ListResourceTags"
    ],
    "Resource": "*"
  }
}
```

使用 AWS KMS 主控台所需的許可

若想使用 AWS KMS 主控台，使用者必須擁有一組最基本的許可，以便允許他們使用其 AWS 帳戶中的 AWS KMS 資源。除了這些 AWS KMS 許可，使用者還必須具有列出 IAM 使用者和 IAM 角色清單的許可。如果您建立比最基本必要許可更嚴格的 IAM 政策，則對於採取該 IAM 政策的使用者而言，AWS KMS 主控台就無法如預期運作。

對於允許使用者唯讀存取 AWS KMS 主控台所需的最低許可，請參閱[允許使用者檢視 AWS KMS 主控台中的 KMS 金鑰]。

若要允許使用者使用AWS KMS主控台建立和管理 KMS 金鑰，請將AWSKeyManagementServicePowerUser受管理的原則附加至使用者，如下節所述。

您不需要允許使用者的最基本主控台許可，透過 AWS 開發套件、AWS Command Line Interface 或 AWS Tools for PowerShell 使用 AWS KMS API。但是，您需要授予這些使用者使用 API 的許可。如需詳細資訊，請參閱 許可參考。

進階使用者的 AWS 受管政策

您可以使用 AWSKeyManagementServicePowerUser 受管政策，為您帳戶中的 IAM 主體提供進階使用者的許可。進階使用者可以建立 KMS 金鑰、使用和管理其建立的 KMS 金鑰，以及檢視所有 KMS 金鑰和 IAM 身分識別。擁有 AWSKeyManagementServicePowerUser 受管政策的主體也可以從其他來源取得許可，包括金鑰政策、其他 IAM 政策和授予。

AWSKeyManagementServicePowerUser 是 AWS 受管 IAM 政策。如需 AWS 受管政策的更多相關資訊，請參閱 IAM 使用者指南中的 AWS 受管政策。

> **Note**
>
> 此政策中特定於 KMS 金鑰的許可，例如 kms:TagResource 和 kms:GetKeyRotationStatus，僅當該 KMS 金鑰的金鑰政策明確允許 AWS 帳戶 使用 IAM 政策來控制對金鑰的存取時才有效。若要確定許可是否特定於 KMS 金鑰，請參閱 AWS KMS 權限 並查找資源資料欄中的 KMS 金鑰。
>
> 這項政策為進階使用者提供針對任何 KMS 金鑰的許可，搭配允許操作的金鑰政策。對於跨帳戶許可，例如 kms:DescribeKey 和 kms:ListGrants，這可能包括不受信任的 AWS 帳戶中的 KMS 金鑰。如需詳細資訊，請參閱 IAM 政策的最佳實務 和 允許其他帳戶中的使用者使用 KMS 金鑰。若要確定許可是否對其他帳戶中的 KMS 金鑰有效，請參閱 AWS KMS 權限 並查找跨帳戶使用資料欄中的是。
>
> 若要允許主參與者檢視主AWS KMS控台而不會發生錯誤，主參與者需要標籤:GetResources權限，此權限不包含在AWSKeyManagementServicePowerUser原則中。您可以在單獨的 IAM 政策中允許此許可。

AWSKeyManagementServicePower 受管 IAM 政策包含下列許可。

- 允許主體建立 KMS 金鑰。由於此程序包含設定金鑰政策，進階使用者可以授予自己和其他人使用和管理其所建立之 KMS 金鑰的許可。
- 允許主體對所有 KMS 金鑰建立和刪除別名和標籤。變更標籤或別名可允許或拒絕使用和管理 KMS 金鑰的許可。如需詳細資訊，請參閱 AWS KMS 的 ABAC。

- 允許主體取得有關所有 KMS 金鑰的詳細資訊，包括金鑰 ARN、密碼編譯組態、金鑰政策、別名、標籤和輪換狀態。
- 允許主體列出 IAM 使用者、群組和角色。
- 此政策不允許主體使用或管理其未建立的 KMS 金鑰。然而，它們可以變更所有 KMS 金鑰上的別名和標籤，這可能允許或拒絕使用或管理 KMS 金鑰的許可。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:CreateAlias",
                "kms:CreateKey",
                "kms:DeleteAlias",
                "kms:Describe*",
                "kms:GenerateRandom",
                "kms:Get*",
                "kms:List*",
                "kms:TagResource",
                "kms:UntagResource",
                "iam:ListGroups",
                "iam:ListRoles",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

IAM 政策範例

在本節中，您可以找到允許各種 AWS KMS 動作之許可的 IAM 政策範例。

> **Important**
>
> 以下政策中的一些許可僅在 KMS 金鑰的金鑰政策也允許時，才會獲得允許。如需詳細資訊，請參閱 許可參考。

如需撰寫及格式化 JSON 政策文件的說明，請參閱《IAM 使用者指南》中的 IAM JSON 政策參考。

範例

- 允許使用者檢視 AWS KMS 主控台中的 KMS 金鑰
- 允許使用者建立 KMS 金鑰
- 允許使用者使用特定 AWS 帳戶 中的任何 KMS 金鑰來加密和解密
- 允許使用者使用特定 AWS 帳戶 和區域中的任何 KMS 金鑰來加密和解密
- 允許使用者使用特定 KMS 金鑰來加密和解密
- 防止使用者停用或刪除任何 KMS 金鑰

允許使用者檢視 AWS KMS 主控台中的 KMS 金鑰

以下 IAM 政策允許使用者唯讀存取 AWS KMS 主控台。具有這些許可的使用者可以檢視其 AWS 帳戶中的所有 KMS 金鑰，但它們無法建立或變更任何 KMS 金鑰。

若要檢視AWS 受管金鑰和客戶受管金鑰頁面上的 KMS 金鑰，主體需要 kms: ListKeys 、 kms: ListAliases 和 tag: GetResources 權限，即使金鑰沒有標籤或別名也一樣。若要在 KMS 金鑰詳細資料頁面上檢視選用的 KMS 金鑰資料表欄和資料DescribeKey，則需要其餘權限，特別是 kms:。需要 iam: ListUsers 和 iam: ListRoles 許可才能在默認視圖中顯示密鑰策略而不會出現錯誤。若要檢視 [自訂金鑰存放區] 頁面上的資料以及自訂金鑰存放區中 KMS 金鑰的詳細資料，主體也需要 kms: DescribeCustomKeyStores 權限。

如果您限制對特定 KMS 金鑰的使用者主控台存取，則主控台會針對不可見的每個 KMS 金鑰顯示錯誤。

此政策包含兩個政策陳述式。第一個政策陳述式中的 `Resource` 元素允許範例 AWS 帳戶 所有區域中所有 KMS 金鑰上的指定許可。主控台檢視器不需要額外的存取權，因為 AWS KMS 主控台只會顯示主體帳戶中的 KMS 金鑰。即使他們有在其他 AWS 帳戶 中檢視 KMS 金鑰的許可，亦是如此。其餘 AWS KMS 和 IAM 許可需要 `"Resource": "*"` 元素，因為其不適用於任何特定的 KMS 金鑰。

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

```
        "kms:GetPublicKey",
        "kms:GetKeyRotationStatus",
        "kms:GetKeyPolicy",
        "kms:DescribeKey",
        "kms:ListKeyPolicies",
        "kms:ListResourceTags",
        "tag:GetResources"
      ],
      "Resource": "arn:aws:kms:*:111122223333:key/*"
    },
    {
      "Sid": "ReadOnlyAccessForOperationsWithNoKMSKey",
      "Effect": "Allow",
      "Action": [
        "kms:ListKeys",
        "kms:ListAliases",
        "iam:ListRoles",
        "iam:ListUsers"
      ],
      "Resource": "*"
    }
  ]
}
```

允許使用者建立 KMS 金鑰

下列 IAM 政策可讓使用者建立各類型的 KMS 金鑰。Resource 元素的值是 *，因為 CreateKey 操作不會使用任何特定 AWS KMS 資源 (KMS 金鑰或別名)。

若要限制使用者使用特定類型的 KMS 金鑰，請使用 kms: KeyUsage、kms: 和 kms: KeyOrigin 條件金鑰。KeySpec

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": "kms:CreateKey",
    "Resource": "*"
  }
}
```

建立金鑰的主體可能需要一些相關許可。

- kms：PutKeyPolicy— 具有kms:CreateKey權限的主體可以設定 KMS 金鑰的初始金鑰原則。不過，CreateKey呼叫者必須具有 kms: PutKeyPolicy 權限，這可讓他們變更 KMS 金鑰原則，或者必須指定的BypassPolicyLockoutSafetyCheck參數CreateKey，不建議這麼做。CreateKey呼叫者可以從 IAM 政策為 KMS 金鑰取得 kms:PutKeyPolicy 許可，或者其可以在所建立之 KMS 金鑰的金鑰政策中包含此許可。
- kms：TagResource— 若要在CreateKey作業期間將標籤新增至 KMS 金鑰，CreateKey呼叫者必須在 IAM 政策中具有 kms: TagResource 權限。在新 KMS 金鑰的金鑰政策中包含此許可並不足夠。但是，如果 CreateKey 呼叫者在初始金鑰政策中包含 kms:TagResource，則他們可以在建立 KMS 金鑰之後，在個別呼叫中新增標籤。
- kms：CreateAlias— 在主控台中建立 KMS 金鑰的AWS KMS主體必須具有 kms：K MS 金鑰和別名的CreateAlias權限。(主控台進行兩個呼叫；一個至 CreateKey 和一個至 CreateAlias)。您必須在 IAM 政策中提供別名許可。您可以在金鑰政策、IAM 政策中提供 KMS 金鑰許可。如需詳細資訊，請參閱 控制對別名的存取。

除了 kms:CreateKey，下列 IAM 政策提供針對 AWS 帳戶 中所有 KMS 金鑰的 kms:TagResource 許可，以及針對該帳戶中所有別名的 kms:CreateAlias 許可。其中還包含一些只能在 IAM 政策中提供的有用唯讀許可。

此 IAM 政策並不包括 kms:PutKeyPolicy 許可或可在金鑰政策中設定的任何其他許可。最佳實務是在金鑰政策中設定這些許可，在其中它們會專門套用至一個 KMS 金鑰。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "IAMPermissionsForParticularKMSKeys",
      "Effect": "Allow",
      "Action": "kms:TagResource",
      "Resource": "arn:aws:kms:*:111122223333:key/*"
    },
    {
      "Sid": "IAMPermissionsForParticularAliases",
      "Effect": "Allow",
      "Action": "kms:CreateAlias",
      "Resource": "arn:aws:kms:*:111122223333:alias/*"
    },
    {
      "Sid": "IAMPermissionsForAllKMSKeys",
      "Effect": "Allow",
```

```
      "Action": [
        "kms:CreateKey",
        "kms:ListKeys",
        "kms:ListAliases"
      ],
      "Resource": "*"
    }
  ]
}
```

允許使用者使用特定 AWS 帳戶 中的任何 KMS 金鑰來加密和解密

下列 IAM 政策可讓使用者使用 AWS 帳戶 111122223333 的任何 KMS 金鑰來加密和解密資料。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "kms:Encrypt",
      "kms:Decrypt"
    ],
    "Resource": "arn:aws:kms:*:111122223333:key/*"
  }
}
```

允許使用者使用特定 AWS 帳戶 和區域中的任何 KMS 金鑰來加密和解密

下列 IAM 政策可讓使用者使用美國西部 (奧勒岡)區域 AWS 帳戶 111122223333 中的任何 KMS 金鑰來加密和解密資料。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "kms:Encrypt",
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:us-west-2:111122223333:key/*"
    ]
```

```
  }
}
```

允許使用者使用特定 KMS 金鑰來加密和解密

以下 IAM 政策允許使用者使用 Resource 元素中指定的兩個 KMS 金鑰來加密和解密資料。若要在 IAM 政策陳述式中指定 KMS 金鑰，您必須使用 KMS 金鑰的金鑰 ARN。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Action": [
      "kms:Encrypt",
      "kms:Decrypt"
    ],
    "Resource": [
      "arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
      "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321"
    ]
  }
}
```

防止使用者停用或刪除任何 KMS 金鑰

以下 IAM 政策能防止使用者停用或刪除任何 KMS 金鑰，即使另一個 IAM 政策或金鑰政策允許這些許可。明確拒絕許可的政策會覆寫所有其他政策，即使那些政策明確允許相同的許可。如需更多詳細資訊，請參閱 對金鑰存取進行故障診斷。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Deny",
    "Action": [
      "kms:DisableKey",
      "kms:ScheduleKeyDeletion"
    ],
    "Resource": "*"
  }
}
```

AWS KMS 中的授權

授權是一個政策工具，允許 AWS 主體使用密碼編譯操作中的 KMS 金鑰。它也可以讓其檢視 KMS 金鑰 (DescribeKey)，並建立和管理授予。當授權存取 KMS 金鑰時，會考慮與金鑰政策和 IAM 政策一起授予。授予通常用於臨時許可，因為您可以建立授予、使用其許可並刪除授予，而無需變更金鑰政策或 IAM 政策。

授予通常會由與 AWS KMS 整合的 AWS 服務使用，以加密您的靜態資料。服務會代表帳戶中的使用者建立授予、使用其許可，並在其任務完成後立即淘汰授予。如需 AWS 服務如何使用授予的詳細資訊，請參閱服務使用者指南或開發人員指南中的 AWS 服務使用 AWS KMS 的方式 或靜態加密主題。

有關示範如何以幾種程式設計語言使用授予的程式碼範例，請參閱 使用授與。

主題

關於授予

授予是非常靈活和有用的存取控制機制。當您建立 KMS 金鑰的授予時，只要授予中的所有指定條件都滿足，授予就會允許承授者主體呼叫 KMS 金鑰上指定的授予操作。

- 每個授予只允許存取一個 KMS 金鑰。您可以在不同的 AWS 帳戶 中為 KMS 金鑰建立授予。
- 授予可以允許存取 KMS 金鑰，但不能拒絕存取。
- 每個授予都有一個承授者主體。承授者主體可以在與 KMS 金鑰相同的 AWS 帳戶 中或不同的帳戶中代表一個或多個身分。
- 授予只能允許授予操作。授予操作必須由授予中的 KMS 金鑰支援。如果您指定不支援的作業，則CreateGrant要求會失敗，並出現ValidationError例外狀況。
- 承授者主體可以在不指定授予的情況下使用授予給予的許可，就如同許可來自金鑰政策或 IAM 政策一樣。然而，由於 AWS KMS API 遵循最終一致性模式，因此當您建立、淘汰或撤銷授權時，在該變更適用於整個 AWS KMS 之前，可能會有短暫延遲。若要立即使用授予中的許可，請使用授予字符。

- 授權主體可以刪除授予 (對其進行淘汰或撤銷）。刪除授予會消除授予允許的所有許可。您不需要弄清楚要新增或移除哪些政策，即可復原授予。
- AWS KMS 會限制每個 KMS 金鑰的授予數目。如需詳細資訊，請參閱 每個 KMS 金鑰的授予：50,000。

建立授予和給予他人建立授予的許可時，請小心謹慎。建立授權的權限有安全性隱患，就像允許 kms:PutKeyPolicy 權限設定原則一樣。

- 具有建立 KMS 金鑰 (kms:CreateGrant) 授予許可的使用者可以使用授予來允許使用者和角色，包括 AWS 服務，以使用 KMS 金鑰。主體可以是您自己 AWS 帳戶 中的身分或不同帳戶或組織中的身分。
- 授予可以只允許 AWS KMS 操作子集。您可以使用授予來允許主體檢視 KMS 金鑰、在密碼編譯操作中使用 KMS 金鑰，以及建立和淘汰授予。如需詳細資訊，請參閱授予操作。您也可以使用授予限制條件來限制授予對稱加密金鑰中的許可。
- 主體可以從金鑰政策或 IAM 政策取得建立授予的許可。針對 KMS 金鑰的任何授與操作，透過政策取得 kms:CreateGrant 許可的主體均可建立授與。這些主體不需擁有他們對金鑰授與的許可。當允許政策中的 kms:CreateGrant 許可時，您可以使用政策條件來限制此許可。
- 主體也可以從授予取得建立授予的許可。這些主體只能委派其所授予的許可，即使他們有來自政策的其他許可。如需詳細資訊，請參閱 授予 CreateGrant 權限。

如需授予相關概念的說明，請參閱授予術語。

授予概念

若要有效地使用授予，您需要了解 AWS KMS 使用的條款和概念。

授予限制條件

限制授予中許可的條件。目前，基於請求中的加密內容，AWS KMS 支援用於密碼編譯操作的授予限制條件。如需詳細資訊，請參閱 使用授予限制條件。

授予 ID

用於 KMS 金鑰的授予唯一識別符。您可以使用授權 ID 以及金鑰識別碼來識別RetireGrant或RevokeGrant請求中的授權。

授予操作

您可以在授予中允許的 AWS KMS 操作。如果您指定其他作業，則CreateGrant要求會失敗，並出現ValidationError例外狀況。這些也是接受授予字符的操作。如需有關這些許可的詳細資訊，請參閱 AWS KMS 權限。

這些授予操作實際上代表使用操作的許可。因此，對於 ReEncrypt 操作，您可以指定 ReEncryptFrom、ReEncryptTo，或兩者的 ReEncrypt*。

授予操作包括：

- 密碼編譯操作
  - 解密
  - 加密
  - GenerateDataKey
  - GenerateDataKeyPair
  - GenerateDataKeyPairWithoutPlaintext
  - GenerateDataKeyWithoutPlaintext
  - GenerateMac
  - ReEncryptFrom
  - ReEncryptTo
  - 符號
  - 確認
  - VerifyMac
- 其他操作
  - CreateGrant
  - DescribeKey
  - GetPublicKey
  - RetireGrant

您允許的授予作業必須由授予中的 KMS 金鑰支援。如果您指定不支援的作業，則CreateGrant要求會失敗，並出現ValidationError例外狀況。例如，對稱加密 KMS 金鑰的授予不允許 Sign、Verify、GenerateMac 或 VerifyMac 操作。非對稱 KMS 金鑰的授予不允許任何產生資料金鑰或資料金鑰對的操作。

授予字符

AWS KMS API 遵循最終一致性模式。當您建立授權時，在該變更適用於整個 AWS KMS 之前，可能會有短暫延遲。變更傳播到整個系統通常需要不到幾秒鐘的時間，但在某些情況下可能需要幾分鐘。如果您在整個系統中完全傳播授權之前嘗試使用授權，則可能會收到拒絕存取錯誤。授予字符可讓您參考授予並立即使用授予許可。

授予字符是唯一的，非秘密的，可變長度的，base64 編碼的字串，代表授予。您可以使用授予字符來識別任何授予操作中的授予。但是，由於字符值是雜湊摘要，因此它不會顯示有關授予的任何詳細資訊。

授權字符設計為只能在整個 AWS KMS 中完全傳播授權之後才能使用。在此之後，承授者主體可以使用授予中的許可，而不提供授予字符或任何其他授予的證據。您可以隨時使用授予令牌，但是一旦授予最終達成一致，AWS KMS 會使用授予來確定許可，而不是授予字符。

例如，下列命令會呼叫GenerateDataKey作業。它使用授予字符來表示授予呼叫者 (承授者主體) 許可來呼叫指定 KMS 金鑰上的 `GenerateDataKey`。

```
$ aws kms generate-data-key \
        --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
        --key-spec AES_256 \
        --grant-token $token
```

您也可以使用授予字符來識別管理授予之操作中的授予。例如，淘汰的主體可以在呼叫RetireGrant作業時使用授與 Token。

```
$ aws kms retire-grant \
        --grant-token $token
```

`CreateGrant` 是傳回授予字符的唯一操作。您無法從任何其他AWS KMS操作或操作的CloudTrail日誌事件中獲取 CreateGrant 授予令牌。ListGrants和作ListRetirableGrants業會傳回授權 ID，但不會傳回授權記號。

如需詳細資訊，請參閱 使用授予字符。

承授者主體

取得授予中所指定許可的身分。每個授予都有一個承授者主體，但承授者主體可以代表多個身分。

承授者主體可以是任何 AWS 主體，包括 AWS 帳戶 (根)、IAM 使用者、IAM 角色、同盟角色或使用者，或假定角色使用者。承授者主體可以在與 KMS 金鑰相同的帳戶中，也可以在不同的帳戶中。不過，承授者主體不能是服務主體、IAM 群組，或 AWS 組織。

> **Note**
>
> IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

淘汰授予

終止授予。當您完成使用許可時，就會淘汰授予。

撤銷和淘汰授予都會刪除授予。但是，由授予中指定的主體完成淘汰動作。撤銷通常是由金鑰管理員完成。如需詳細資訊，請參閱 淘汰和撤銷授予。

淘汰主體

可以淘汰授予的主體。您可以在授予中指定淘汰的主體，但不是必需的。淘汰的主體可以是任何 AWS 主體，包括 AWS 帳戶、IAM 使用者、IAM 角色、聯合身分使用者和假設的角色使用者。淘汰的主體可以在與 KMS 金鑰相同的帳戶中，也可以在不同的帳戶中。

> **Note**
>
> IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

除了授予內指定的淘汰主體外，還可以由在其中建立授予的 AWS 帳戶 淘汰授予。承授者主體可以淘汰授予 (如果授予允許 `RetireGrant` 操作)。另外，淘汰主體的 AWS 帳戶 或 AWS 帳戶 可以委派許可，以在相同 AWS 帳戶 中淘汰對 IAM 主體的授予。如需詳細資訊，請參閱 淘汰和撤銷授予。

撤銷 (授予)

終止授予。您撤銷授予，以主動拒絕授予允許的許可。

撤銷和淘汰授予都會刪除授予。但是，由授予中指定的主體完成淘汰動作。撤銷通常是由金鑰管理員完成。如需詳細資訊，請參閱 淘汰和撤銷授予。

最終一致性 (用於授予)

AWS KMS API 遵循最終一致性模式。當您建立、淘汰或撤銷授權時，在該變更適用於整個 AWS KMS 之前，可能會有短暫延遲。變更傳播到整個系統通常需要不到幾秒鐘的時間，但在某些情況下可能需要幾分鐘。

如果遇到非預期的錯誤，您可能會發現這個短暫的延遲。例如，如果在授予適用於整個 AWS KMS 之前嘗試管理新的授予，或在新的授予中使用許可，則您可能會遇到存取遭拒的錯誤。如果您淘汰或撤銷授予，承授者主體可能仍然可以在短期內使用其許可，直到授予完全刪除為止。典型的策略是重試請求，有些 AWS 開發套件包括自動退回和重試邏輯。

AWS KMS 具有緩解此短暫延遲的功能。

- 若要立即使用新授予中的許可，請使用授予字符。您可以使用授予字符來引用任何授予操作中的授予。如需說明，請參閱使用授予字符。
- CreateGrant作業具有可防止重試作業建立重複授權的Name參數。

> **Note**
>
> 授予許可會取代授予的有效性，直到服務中的所有端點都更新為新的授予狀態為止。在大多數情況下，最終一致性將在五分鐘內達成。

如需詳細資訊，請參閱《AWS KMS 最終一致性》。

AWS KMS 授予的最佳實務

建立、使用和管理授予時，AWS KMS 會推薦下列最佳實務。

- 將授予中的許可限制為承授者主體需要的許可。使用最低權限存取的原則。
- 使用特定承授者主體 (例如 IAM 角色)，並讓承授者主體僅使用其所需 API 操作的許可。
- 使用加密內容授予限制條件，以確保呼叫者使用 KMS 金鑰來達到預期目的。如需如何在要求中使用加密內容來保護資料的詳細資訊，請參閱如何使用AWS Key Management Service和AWS安全性部落格 EncryptionContext中的保護加密資料的完整性。

> **Tip**
>
> 盡可能使用EncryptionContextEqual授予約束。EncryptionContextSubset授予約束更難以正確使用。如果您需要使用，請仔細閱讀文件並測試授予限制條件，以確保其按預期工作。

- 刪除重複的授予。重複授予具有相同的金鑰 ARN、API 動作、承授者主體、加密內容和名稱。如果您淘汰或撤銷原始授予但保留重複項目，則剩餘的重複授予會構成非預期的權限升級。為了避免在重試 `CreateGrant` 請求時重複授予，請使用 Name 參數。若要偵測重複的授權，請使用此ListGrants作業。如果意外建立了重複授予，請儘快淘汰或撤銷該授予。

> **Note**
>
> AWS 受管金鑰授予可能看起來像重複項，但具有不同的承授者主體。`GranteePrincipal` 回應中的 `ListGrants` 欄位通常包含授與的承授者主體。不過，當授予中的承授者主體是 AWS 服務時，`GranteePrincipal` 欄位會包含服務主體，這可能代表數個不同的承授者主體。

- 請記住，授予不會自動過期。不再需要許可時，即刻淘汰或撤銷授予。未刪除的授予可能會對加密資源造成安全風險。

建立授予

建立授予之前，請先了解自訂授予的選項。您可以使用授予限制條件以限制授予中的許可。此外，了解授予 `CreateGrant` 許可。從授予取得建立授予之許可的主體，在其可以建立的授予中受到限制。

主題

- 建立授與
- 使用授予限制條件
- 授予 CreateGrant 權限

建立授與

若要建立授權，請呼叫CreateGrant作業。指定 KMS 金鑰、承授者主體和允許的授予操作清單。您也可以指定選用的淘汰主體。若要自訂授與，請使用選用 `Constraints` 參數來定義授與限制。

當您建立、淘汰或撤銷授予時，可能會有短暫的延遲 (通常不到五分鐘)，然後才會在整個 AWS KMS 中可用。如需詳細資訊，請參閱最終一致性模式 (授權)。

例如，下列 `CreateGrant` 命令會建立授予，以允許被授權擔任 `keyUserRole` 角色的使用者對指定的對稱 KMS 金鑰呼叫 Decrypt 操作。授權會使用 `RetiringPrincipal` 參數指定可淘汰授權的主體。同時也包含授權限制條件，僅當請求中的加密內容包含 `"Department": "IT"` 時，才允許許可。

```
$  aws kms create-grant \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --grantee-principal arn:aws:iam::111122223333:role/keyUserRole \
```

```
    --operations Decrypt \
    --retiring-principal arn:aws:iam::111122223333:role/adminRole \
    --constraints EncryptionContextSubset={Department=IT}
```

如果您的程式碼重試 CreateGrant 操作，或使用自動重試請求的 AWS 開發套件，則請使用選用名稱參數，以防止建立重複授予。如果 AWS KMS 取得具有與現有授予相同屬性之授予的 CreateGrant 請求，包括名稱，則它會將請求識別為重試，並且不會建立新授予。您無法使用 Name 值以識別任何 AWS KMS 操作中的授予。

> **Important**
>
> 請勿在授權名稱包含機密或敏感資訊。它可能會在 CloudTrail 日誌和其他輸出中以純文本形式出現。

```
$ aws kms create-grant \
    --name IT-1234abcd-keyUserRole-decrypt \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --grantee-principal arn:aws:iam::111122223333:role/keyUserRole \
    --operations Decrypt \
    --retiring-principal arn:aws:iam::111122223333:role/adminRole \
    --constraints EncryptionContextSubset={Department=IT}
```

有關示範如何以幾種程式設計語言使用授予的程式碼範例，請參閱 使用授與。

使用授予限制條件

授予限制條件會在為承授者主體提供授予的許可上設定條件。授予限制條件會取代金鑰政策或 IAM 政策中的條件索引鍵。每個授予限制條件值最多可以包含 8 個加密內容對。每個授予限制條件中的加密內容值不能超過 384 個字元。

> **Important**
>
> 請勿在此欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，此欄位可能會以純文字顯示。

AWS KMS 支援兩個授予限制條件，EncryptionContextEquals 和 EncryptionContextSubset，兩者皆為密碼編譯操作中請求的加密內容建立了要求。

加密內容授予限制條件是設計用於與擁有加密內容參數的授予操作搭配使用。

- 加密內容限制條件僅在對稱加密 KMS 金鑰的授予中有效。其他 KMS 金鑰的密碼編譯操作不支援加密內容。
- 加密內容限制條件會忽略 DescribeKey 和 RetireGrant 操作。DescribeKey 和 RetireGrant 沒有加密內容參數，但您可以將這些操作包含在具有加密內容限制條件的授予中。
- 您可以在 CreateGrant 操作的授予中使用加密內容限制條件。加密內容限制條件要求使用 CreateGrant 許可建立的任何授予具有同樣嚴格或更嚴格的加密內容限制條件。

AWS KMS 支援下列加密內容授予限制條件。

EncryptionContextEquals

使用 EncryptionContextEquals，以指定允許請求的確切加密內容。

EncryptionContextEquals 需要請求中的加密內容對完全符合 (包括大小寫) 授予限制條件中的加密內容。此對組可以任何順序顯示，但每個對組中的金鑰和值不能改變。

例如，如果 EncryptionContextEquals 授予限制條件需要 "Department": "IT" 加密內容對，則僅當請求中的加密內容正好是 "Department": "IT" 時，授予才會允許指定類型的請求。

EncryptionContextSubset

使用 EncryptionContextSubset 來要求請求包含特定的加密內容對。

EncryptionContextSubset 需要請求包含授予限制條件中的所有加密內容對 (完全符合，包括大小寫)，但請求也可以有其他加密內容對。此對組可以任何順序顯示，但每個對組中的金鑰和值不能改變。

例如，如果 EncryptionContextSubset 授予限制條件需要 Department=IT 加密內容對，則當請求中的加密內容為 "Department": "IT" 時，或包含 "Department": "IT" 與其他加密內容對 (例如 "Department": "IT","Purpose": "Test") 時，授予才會允許指定類型的請求。

若要在授與中指定對稱加密 KMS 金鑰的加密內容限制，請在CreateGrant作業中使用Constraints參數。此命令建立的授予會給予被授權擔任 keyUserRole 角色的使用者呼叫 Decrypt 操作的許可。但是，僅當 Decrypt 請求中的加密內容為 "Department": "IT" 加密內容對時，該許可才會生效。

```
$ aws kms create-grant \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
```

```
    --grantee-principal arn:aws:iam::111122223333:role/keyUserRole \
    --operations Decrypt \
    --retiring-principal arn:aws:iam::111122223333:role/adminRole \
    --constraints EncryptionContextEquals={Department=IT}
```

產生的授與看起來如下。請注意，授予給 keyUserRole 角色的許可只有在 Decrypt 請求使用授予限制條件中指定的相同加密內容對時才有效。若要尋找 KMS 金鑰的授權，請使用ListGrants作業。

```
$ aws kms list-grants --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
{
    "Grants": [
        {
            "Name": "",
            "IssuingAccount": "arn:aws:iam::111122223333:root",
            "GrantId":
 "abcde1237f76e4ba7987489ac329fbfba6ad343d6f7075dbd1ef191f0120514a",
            "Operations": [
                "Decrypt"
            ],
            "GranteePrincipal": "arn:aws:iam::111122223333:role/keyUserRole",
            "Constraints": {
                "EncryptionContextEquals": {
                    "Department": "IT"
                }
            },
            "CreationDate": 1568565290.0,
            "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
            "RetiringPrincipal": "arn:aws:iam::111122223333:role/adminRole"
        }
    ]
}
```

為了滿足 EncryptionContextEquals 授予限制條件，Decrypt 操作請求中的加密內容必須是 "Department": "IT" 對。來自承授者主體的類似下列請求應當滿足 EncryptionContextEquals 授予限制條件。

```
$ aws kms decrypt \
    --key-id arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab\
    --ciphertext-blob fileb://encrypted_msg \
    --encryption-context Department=IT
```

當授予限制條件為 EncryptionContextSubset，請求中的加密內容對必須包含授予限制條件中的加密內容對，但請求也可以包含其他加密內容對。下列授予限制條件需要請求中的其中一個加密內容對為 "Deparment": "IT"。

```
"Constraints": {
   "EncryptionContextSubset": {
       "Department": "IT"
   }
}
```

來自承授者主體的類似下列請求應當同時滿足此範例中的 EncryptionContextEqual 和 EncryptionContextSubset 授予限制條件。

```
$ aws kms decrypt \
    --key-id arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab \
    --ciphertext-blob fileb://encrypted_msg \
    --encryption-context Department=IT
```

然而，來自承授者主體的類似下列請求應當滿足 EncryptionContextSubset 授予限制條件，但它無法滿足 EncryptionContextEquals 授予限制條件。

```
$ aws kms decrypt \
    --key-id arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab \
    --ciphertext-blob fileb://encrypted_msg \
    --encryption-context Department=IT,Purpose=Test
```

AWS 服務通常會在提供其在您 AWS 帳戶 中使用 KMS 金鑰的許可授予中使用加密內容限制條件。例如，Amazon DynamoDB 使用類似下列授予來取得在您帳戶中為 DynamoDB 使用 AWS 受管金鑰 的許可。此 EncryptionContextSubset 授與中的授與限制只有在請求中的加密內容包含 "subscriberID": "111122223333" 和 "tableName": "Services" 對時，才會讓授與中的許可生效。此授予限制條件表示授予允許 DynamoDB 僅針對您 AWS 帳戶 中的特定資料表使用指定的 KMS 金鑰。

若要取得此輸出，請在帳戶中AWS 受管金鑰的 DynamoDB 上執行ListGrants作業。

```
$ aws kms list-grants --key-id 0987dcba-09fe-87dc-65ba-ab0987654321

{
```

```
    "Grants": [
        {
            "Operations": [
                "Decrypt",
                "Encrypt",
                "GenerateDataKey",
                "ReEncryptFrom",
                "ReEncryptTo",
                "RetireGrant",
                "DescribeKey"
            ],
            "IssuingAccount": "arn:aws:iam::111122223333:root",
            "Constraints": {
                "EncryptionContextSubset": {
                    "aws:dynamodb:tableName": "Services",
                    "aws:dynamodb:subscriberId": "111122223333"
                }
            },
            "CreationDate": 1518567315.0,
            "KeyId": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321",
            "GranteePrincipal": "dynamodb.us-west-2.amazonaws.com",
            "RetiringPrincipal": "dynamodb.us-west-2.amazonaws.com",
            "Name": "8276b9a6-6cf0-46f1-b2f0-7993a7f8c89a",
            "GrantId":
 "1667b97d27cf748cf05b487217dd4179526c949d14fb3903858e25193253fe59"
        }
    ]
}
```

授予 CreateGrant 權限

授予可以包含呼叫 CreateGrant 操作的許可。但是，當承授者主體從授予 (而非政策) 取得呼叫 CreateGrant 的許可時，該許可會受到限制。

- 承授者主體只能建立允許父項授予中部分或全部操作的授予。
- 其所建立授予中的授予限制條件必須至少與父項授予一樣嚴格。

這些限制不適用於從政策取得 CreateGrant 許可的主體，雖然其許可可以受到政策條件限制。

例如，假設有一個授與允許承授者主體呼叫 GenerateDataKey、Decrypt 和 CreateGrant 操作。我們稱之為授予，允許 CreateGrant 許可父項授予。

```
# The original grant in a ListGrants response.
{
    "Grants": [
        {
            "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
            "CreationDate": 1572216195.0,
            "GrantId":
 "abcde1237f76e4ba7987489ac329fbfba6ad343d6f7075dbd1ef191f0120514a",
            "Operations": [
                "GenerateDataKey",
                "Decrypt",
                "CreateGrant
            ]
            "RetiringPrincipal": "arn:aws:iam::111122223333:role/adminRole",
            "Name": "",
            "IssuingAccount": "arn:aws:iam::111122223333:root",
            "GranteePrincipal": "arn:aws:iam::111122223333:role/keyUserRole",
            "Constraints": {
                "EncryptionContextSubset": {
                    "Department": "IT"
                }
            },
        }
    ]
}
```

承授者主體 exampleUser 可以使用此許可來建立授予，其中包含原始授予中指定的任何操作子集，例如 CreateGrant 和 Decrypt。子項授予不能包含其他操作，例如 ScheduleKeyDeletion 或 ReEncrypt。

此外，子項授予中的授予限制條件必須與父項授予中的限制程度相等嚴格或比之更嚴格。例如，子授與可以新增對組到父授與的 EncryptionContextSubset 限制，但不能將其移除。子授與可將 EncryptionContextSubset 限制變更為 EncryptionContextEquals 限制，但不能反向。

例如，承授者主體可以使用其從父項授予中取得的 CreateGrant 許可，建立以下子項授予。子項授予的操作是父項授予操作的子集，授予限制條件更嚴格。

```
# The child grant in a ListGrants response.
{
    "Grants": [
        {
```

```
            "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
            "CreationDate": 1572249600.0,
            "GrantId":
 "fedcba9999c1e2e9876abcde6e9d6c9b6a1987650000abcee009abcdef40183f",
            "Operations": [
                "CreateGrant"
                "Decrypt"
            ]
            "RetiringPrincipal": "arn:aws:iam::111122223333:user/exampleUser",
            "Name": "",
            "IssuingAccount": "arn:aws:iam::111122223333:root",
            "GranteePrincipal": "arn:aws:iam::111122223333:user/anotherUser",
            "Constraints": {
IAM best practices discourage the use of IAM users with long-term credentials. Whenever
                      possible, use IAM roles, which provide temporary credentials. For
 details,
                      see Security best practices in IAM in the IAM User Guide.
        "EncryptionContextEquals": {
                    "Department": "IT"
                }
            },
        }
    ]
}
```

子項授予的承授者主體 anotherUser，可以使用其 CreateGrant 許可建立授予。然而，anotherUser 建立的授予必須在其父項授予或子集中包含操作，並且授予限制條件必須相同或更嚴格。

管理授予

具有必要許可的主體可以檢視、使用和刪除 (淘汰或撤銷) 授予。若要調整建立和管理授予的許可，AWS KMS 支援多個政策條件，可以用於金鑰政策和 IAM 政策。

主題

- 控制對授予的存取
- 檢視授予
- 使用授予字符
- 淘汰和撤銷授予

控制對授予的存取

您可以控制對在金鑰政策、IAM 政策和授予中建立和管理授予之操作的存取。從授予取得 `CreateGrant` 許可的主體具有更有限的授予許可。

API 操作	金鑰政策或 IAM 政策	授權
CreateGrant	✓	✓
ListGrants	✓	-
ListRetirableGrants	✓	-
淘汰授予	(有限。請參閱 淘汰和撤銷授予)	✓
RevokeGrant	✓	-

當使用金鑰政策或 IAM 政策控制對建立和管理授予之操作的存取時，您可以使用一或多個以下政策條件來限制許可。AWS KMS 支援所有以下授予相關條件索引鍵。如需詳細資訊和範例，請參閱 AWS KMS 條件鍵。

公里：GrantConstraintType

僅當授予包含指定的授予限制條件時，才允許主體建立授予。

公里：GrantIsForAWSResource

僅當與 AWS KMS 整合的 AWS 服務代表主體傳送請求時，才允許主體呼叫 `CreateGrant`、`ListGrants` 或 `RevokeGrant`。

公里：GrantOperations

允許主體建立授予，但將授予限制為指定的操作。

公里：GranteePrincipal

允許主體僅針對指定的承授者主體建立授予。

公里：RetiringPrincipal

僅當授予指定特定的淘汰主體時，才允許主體建立授予。

檢視授予

若要檢視授權，請使用ListGrants作業。您必須指定授予套用的 KMS 金鑰。您也可以透過授予 ID 或承授者主體篩選授予清單。如需更多範例，請參閱 檢視授與。

若要檢視與區域中AWS 帳戶具有特定退休主體的所有授權，請使用ListRetirableGrants。回應包括每項授予的詳細資訊。

> **Note**
>
> GranteePrincipal 回應中的 ListGrants 欄位通常包含授與的承授者主體。不過，當授予中的承授者主體是 AWS 服務時，GranteePrincipal 欄位會包含服務主體，這可能代表數個不同的承授者主體。

例如，下列命令會列出 KMS 金鑰的所有授予。

```
$  aws kms list-grants --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
{
    "Grants": [
        {
            "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
            "CreationDate": 1572216195.0,
            "GrantId":
 "abcde1237f76e4ba7987489ac329fbfba6ad343d6f7075dbd1ef191f0120514a",
            "Constraints": {
                "EncryptionContextSubset": {
                    "Department": "IT"
                }
            },
            "RetiringPrincipal": "arn:aws:iam::111122223333:role/adminRole",
            "Name": "",
            "IssuingAccount": "arn:aws:iam::111122223333:root",
            "GranteePrincipal": "arn:aws:iam::111122223333:user/exampleUser",
            "Operations": [
                "Decrypt"
            ]
        }
    ]
}
```

使用授予字符

AWS KMS API 遵循最終一致性模式。當建立授予時，授予可能無法立即生效。在該變更適用於整個 AWS KMS 之前，可能會有短暫延遲。變更傳播到整個系統通常需要不到幾秒鐘的時間，但在某些情況下可能需要幾分鐘。一旦授權完全人傳播至整個系統，承授者主體即可使用授權的許可，而不需指定授權字符或授權的任何證據。然而，如授權仍新，尚未適用於所有 AWS KMS，則請求可能失敗，並顯示 AccessDeniedException 錯誤。

若要立即使用新授予中的許可，請使用授予的授予字符。保存CreateGrant操作返回的授予令牌。然後在 AWS KMS 操作的請求中提交授予字符。您可以將授予字符提交給任何 AWS KMS 授予操作，並且您可以在同一次請求中提交多個授予字符。

下列範例會使用CreateGrant作業建立允許GenerateDataKey和 D ecryp t 作業的授權。它會儲存 CreateGrant 在 token 變數中傳回的授予字符。然後，在呼叫 GenerateDataKey 操作時，它會使用 token 變數中的授予字符。

```
# Create a grant; save the grant token
$ token=$(aws kms create-grant \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --grantee-principal arn:aws:iam::111122223333:user/appUser \
    --retiring-principal arn:aws:iam::111122223333:user/acctAdmin \
    --operations GenerateDataKey Decrypt \
    --query GrantToken \
    --output text)

# Use the grant token in a request
$ aws kms generate-data-key \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --key-spec AES_256 \
    --grant-tokens $token
```

即使在整個 AWS KMS 均可使用授權之前，具許可的主體也可利用授權字符來淘汰新的授權。(RevokeGrant 操作不接受授予字符。) 如需詳細資訊，請參閱 淘汰和撤銷授予。

```
# Retire the grant
$ aws kms retire-grant --grant-token $token
```

淘汰和撤銷授予

若要刪除授予，請淘汰或撤銷授予。

RetireGrant和RevokeGrant作業彼此非常相似。這兩項操作都會刪除授予，這會消除授予允許的許可。這些操作之間的主要區別在於它們是如何取得授權的。

RevokeGrant

與大多數 AWS KMS 操作相似，透過金鑰政策和 IAM 政策控制 RevokeGrant 操作的存取。任何具有kms:RevokeGrant權限的主體都可以呼叫 RevokeGrantAPI。此許可會納入提供給金鑰管理員的標準許可中。一般而言，管理員會撤銷授予，以拒絕授予允許的許可。

RetireGrant

授予決定誰可以將其淘汰。此設計可讓您控制授予的生命週期，而不需變更金鑰政策或 IAM 政策。一般而言，當您使用其許可完成時，就會淘汰授予。

可以透過授予中指定的選用淘汰主體淘汰授予。承授者主體亦可他淘汰授予，但前提是其也是淘汰主體或包含 RetireGrant 操作的授予。作為備份，在其中建立授予的 AWS 帳戶 可以淘汰授予。

有可以在 IAM 政策中使用的 kms:RetireGrant 許可，但其公用程式有限。在授予中指定的主體可以淘汰授予，無需 kms:RetireGrant 許可。單獨的 kms:RetireGrant 許可不允許主體淘汰授予。kms:RetireGrant 許可在金鑰政策中無效。

- 若要拒絕淘汰授予的許可，您可以使用具有 kms:RetireGrant 許可的 Deny 動作。
- 擁有 KMS 金鑰的 AWS 帳戶 可以將 kms:RetireGrant 許可委派給帳戶中的 IAM 主體。
- 如果淘汰的主體是不同的 AWS 帳戶，則其他帳戶的管理員可以使用 kms:RetireGrant，將淘汰授予的許可委派給該帳戶中的 IAM 主體。

AWS KMS API 遵循最終一致性模式。當您建立、淘汰或撤銷授權時，在該變更適用於整個 AWS KMS 之前，可能會有短暫延遲。變更傳播到整個系統通常需要不到幾秒鐘的時間，但在某些情況下可能需要幾分鐘。如果您需要立即刪除新的授予，則請在可用於整個 AWS KMS 之前，使用授予字符淘汰授予。您無法使用授予字符來撤銷授予。

透過 VPC 端點連線至 AWS KMS

您可以透過 Virtual Private Cloud (VPC) 內的私有端點直接連線至 AWS KMS。使用介面 VPC 端點時，VPC 和 AWS KMS 之間的通訊完全在 AWS 網路中執行。

AWS KMS 現在支援採用 AWS PrivateLink 技術的 Amazon Virtual Private Cloud (Amazon VPC) 端點。每個 VPC 端點皆會由一個或多個具私有 IP 地址彈性網路界面 (ENI) 來表示，而該界面位於 VPC 子網路中。

界面 VPC 端點可以直接將 VPC 連接至 AWS KMS，無需透過網際網路閘道、NAT 裝置、VPN 連接或 AWS Direct Connect 連線。VPC 中的執行個體不需要公有 IP 地址，即能與 AWS KMS 通訊。

區域

AWS KMS 在支援 AWS KMS 的所有 AWS 區域 中支援 VPC 端點和 VPC 端點政策。

主題

AWS KMS VPC 端點的考量事項

在設定 AWS KMS 的介面 VPC 端點之前，請務必檢閱《AWS PrivateLink 使用者指南》中的介面端點屬性和限制主題。

AWS KMS 支援的 VPC 端點包括下列項目。

- 您可以使用 VPC 端點從 VPC 中呼叫所有 AWS KMS API 操作。
- 您可建立連接至 AWS KMS 區域端點或 AWS KMSFIPS 端點的介面 VPC 端點。
- 您可以使用 AWS CloudTrail 日誌，來稽核透過 VPC 端點使用 KMS 金鑰的情況。如需詳細資訊，請參閱 記錄您的 VPC 端點。

為 AWS KMS 建立一個 VPC 端點

您可使用 Amazon VPC 主控台或 Amazon VPC API 來為 AWS KMS 建立 VPC 端點。如需詳細資訊，請參閱 《AWS PrivateLink 指南》中的建立介面端點。

- 若要建立 AWS KMS 的 VPC 端點，請使用下列服務名稱：

```
com.amazonaws.region.kms
```

例如，在美國西部 (奧勒岡) 區域 (us-west-2)，服務名稱為：

```
com.amazonaws.us-west-2.kms
```

- 若要建立連接 AWS KMS FIPS 端點 的 VPC 端點，請採用下列服務名稱：

```
com.amazonaws.region.kms-fips
```

例如，在美國西部 (奧勒岡) 區域 (us-west-2)，服務名稱為：

```
com.amazonaws.us-west-2.kms-fips
```

若要更輕鬆使用 VPC 端點，您可以為 VPC 端點啟用私有 DNS 名稱。如果您選擇 Enable DNS Name (啟用 DNS 名稱) 選項，標準 AWS KMS DNS 主機名稱會解析為您的 VPC 端點。例如，https://kms.us-west-2.amazonaws.com 會解析為連接至服務名稱 com.amazonaws.us-west-2.kms 的 VPC 端點。

此選項可讓您更輕鬆使用 VPC 端點。依預設，AWS 開發套件和 AWS CLI 會使用標準 AWS KMS DNS 主機名稱，因此您不需要在應用程式和命令中指定 VPC 端點 URL。

如需詳細資訊，請參閱《AWS PrivateLink 指南》中的透過介面端點存取服務。

連線到 AWS KMS VPC 端點

您可以使用 AWS、AWS CLI 或 AWS Tools for PowerShell 軟體開發套件，透過 VPC 端點連線至 AWS KMS。若要指定 VPC 端點，請使用它的 DNS 名稱。

例如，此 list-keys 命令會使用 endpoint-url 參數來指定 VPC 端點。若要使用如下的命令，請將範例 VPC 端點 ID 換成您帳戶中的 ID。

```
$ aws kms list-keys --endpoint-url https://vpce-1234abcdf5678c90a-09p7654s-us-east-1a.ec2.us-east-1.vpce.amazonaws.com
```

如果您在建立 VPC 端點時啟用私有主機名稱，則不需要在 CLI 命令或應用程式組態中指定 VPC 端點 URL。標準 AWS KMS DNS 主機名稱會解析為您的 VPC 端點。根據預設，AWS CLI 和軟體開發套件會使用此主機名稱，因此您不需要變更您的指令碼和應用程式，就可以開始使用該 VPC 端點連接到 AWS KMS 區域端點。

若要使用私有主機名稱，您 VPC 的 `enableDnsHostnames` 和 `enableDnsSupport` 屬性必須設為 `true`。若要設定這些屬性，請使用ModifyVpcAttribute作業。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的檢視和更新 VPC 的 DNS 屬性。

控制對 VPC 端點的存取

為此，控制 AWS KMS 對 VPC 端點的存取，將 VPC 端點政策連接到您的 VPC 端點。端點政策會決定委託人是否可以使用 VPC 端點呼叫 AWS KMS 資源上的 AWS KMS 操作。

您可以在建立端點時建立 VPC 端點政策，並且可以隨時變更 VPC 端點政策。使用 VPC 管理主控台或CreateVpcEndpoint或作ModifyVpcEndpoint業。您也可以建立和變更 VPC 端點政策，方法是使用 AWS CloudFormation 範本。如需有關如何使用 VPC 管理主控台的說明，請參閱《AWS PrivateLink 指南》中的建立介面端點和修改介面端點。

> **Note**
>
> AWS KMS 支援從 2020 年 7 月開始的 VPC 端點政策。在該日期之前建立的 AWS KMS VPC 端點擁有預設 VPC 端點政策，但您可隨時變更。

如需撰寫及格式化 JSON 政策文件的說明，請參閱《IAM 使用者指南》中的 IAM JSON 政策參考。

主題

- 關於 VPC 端點政策
- 預設 VPC 端點政策
- 建立 VPC 端點政策
- 檢視 VPC 端點政策

關於 VPC 端點政策

對於成功使用 VPC 端點的 AWS KMS 請求，委託人需要來自兩個來源的許可：

- 金鑰政策、IAM 政策，或授予必須為委託人授予對資源 (KMS 金鑰或別名) 呼叫操作的許可。
- VPC 端點政策必須授予委託人許可，才能使用端點提出請求。

例如，金鑰政策可能會授予委託人對特定 KMS 金鑰呼叫 Decrypt 的許可。不過，VPC 端點政策可能不允許該委託人透過使用端點對該 KMS 金鑰呼叫 `Decrypt`。

或者，VPC 端點原則可能會允許主體使用端點呼叫特DisableKey定 KMS 金鑰。但是，如果委託人沒有來自金鑰政策、IAM 政策或授予的許可，則請求會失敗。

預設 VPC 端點政策

每個 VPC 端點都有 VPC 端點政策，但您不需要指定政策。如果您未指定政策，則預設端點政策會允許端點上所有資源的所有委託人進行所有操作。

然而，對於 AWS KMS 資源，委託人也必須具有來自金鑰政策、IAM 政策或授予的呼叫操作許可。因此，實際上，預設政策指出，如果委託人具有對資源呼叫操作的許可，則其也可以使用端點來進行呼叫。

```
{
  "Statement": [
    {
      "Action": "*",
      "Effect": "Allow",
      "Principal": "*",
      "Resource": "*"
    }
  ]
}
```

若要僅允許主體將 VPC 端點用於其允許操作的子集，請建立或更新 VPC 端點政策。

建立 VPC 端點政策

VPC 端點政策決定委託人是否具有使用 VPC 端點對資源執行操作的許可。對於 AWS KMS 資源，委託人也必須具有來自金鑰政策、IAM 政策或授予的執行操作許可。

每個 VPC 端點政策陳述式都需要下列元素：

- 可執行動作的委託人
- 可執行的動作
- 可在其中執行動作的資源

政策陳述式不會指定 VPC 端點。相反地，它適用於連接政策的任何 VPC 端點。如需詳細資訊，請參閱《Amazon VPC 使用者指南》中的使用 VPC 端點控制對服務的存取。

以下是 AWS KMS VPC 端點政策的範例。當連接到 VPC 端點時，此政策允許 ExampleUser 使用 VPC 端點對指定的 KMS 金鑰呼叫指定的操作。在使用這類政策之前，請將範例委託人和金鑰 ARN 取代為您帳戶中的有效值。

```
{
   "Statement":[
      {
         "Sid": "AllowDecryptAndView",
         "Principal": {"AWS": "arn:aws:iam::111122223333:user/ExampleUser"},
         "Effect":"Allow",
         "Action": [
             "kms:Decrypt",
             "kms:DescribeKey",
             "kms:ListAliases",
             "kms:ListKeys"
          ],
         "Resource": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
      }
   ]
}
```

AWS CloudTrail 會記錄所有使用 VPC 端點的操作。不過，您的 CloudTrail 記錄不包含其他帳戶中主體要求的作業，或是其他帳戶中 KMS 金鑰的作業。

因此，您可能想要建立 VPC 端點政策，以防止外部帳戶中的委託人使用 VPC 端點呼叫本機帳戶中任何金鑰的任何 AWS KMS 操作。

下列範例使用 aws: PrincipalAccount 全域條件金鑰來拒絕存取所有 KMS 金鑰上所有作業的所有主體，除非主體位於本機帳戶中。使用這類政策之前，請將範例帳戶 ID 取代為有效值。

```
{
  "Statement": [
    {
      "Sid": "AccessForASpecificAccount",
      "Principal": {"AWS": "*"},
      "Action": "kms:*",
      "Effect": "Deny",
      "Resource": "arn:aws:kms:*:111122223333:key/*",
      "Condition": {
        "StringNotEquals": {
          "aws:PrincipalAccount": "111122223333"
```

```
        }
      }
    }
  ]
}
```

檢視 VPC 端點政策

若要檢視端點的 VPC 端點原則，請使用 VPC 管理主控台或作業。DescribeVpcEndpoints

以下 AWS CLI 命令會取得具有指定 VPC 端點 ID 的端點政策。

使用此命令之前，請將範例端點 ID 取代為您帳戶的有效 ID。

```
$ aws ec2 describe-vpc-endpoints \
--query 'VpcEndpoints[?VpcEndpointId==`vpce-1234abcdf5678c90a`].[PolicyDocument]'
--output text
```

在政策陳述式中使用 VPC 端點

請求來自 VPC 或使用 VPC 端點時，您可以控制對 AWS KMS 資源和操作的存取。為此，請使用金鑰政策或 IAM 政策的下列其中一個全域條件索引鍵。

- 使用 `aws:sourceVpce` 條件索引鍵，以根據 VPC 端點來授予或限制存取。
- 使用 `aws:sourceVpc` 條件索引鍵，以根據託管私有端點的 VPC 來授予或限制存取。

> **Note**
>
> 根據 VPC 端點建立金鑰政策和 IAM 政策時，請務必謹慎執行。如果政策陳述式要求請求必須來自特定 VPC 或 VPC 端點，則從代表您使用 AWS KMS 資源的整合 AWS 服務所送來的請求可能會失敗。如需協助，請參閱 在政策中使用 VPC 端點條件搭配 AWS KMS 許可。
> 此外，當請求來自 Amazon VPC 端點時，`aws:sourceIP` 條件索引鍵無效。若要限制對 VPC 端點的請求，請使用 `aws:sourceVpce` 或 `aws:sourceVpc` 條件金鑰。如需詳細資訊，請參閱《AWS PrivateLink 指南》中的VPC 端點和 VPC 端點服務的身分與存取管理

您可以使用這些全域條件金鑰來控制對 AWS KMS keys (KMS 金鑰)、別名的存取，以及不依賴於任何特定資源的作業。CreateKey

例如，以下範例金鑰政策只在請求使用指定 VPC 端點時，才允許使用者執行某些密碼編譯操作。使用者對 AWS KMS 提出請求時，系統會比較請求中的 VPC 端點 ID 與政策中的 aws:sourceVpce 條件金鑰值。如果不相符，則會拒絕請求。

若要使用如下的政策，請將 AWS 帳戶 ID 和 VPC 端點 ID 預留位置換成適用於您帳戶的有效值。

```
{
    "Id": "example-key-1",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Enable IAM policies",
            "Effect": "Allow",
            "Principal": {"AWS":["111122223333"]},
            "Action": ["kms:*"],
            "Resource": "*"
        },
        {
            "Sid": "Restrict usage to my VPC endpoint",
            "Effect": "Deny",
            "Principal": "*",
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*"
            ],
            "Resource": "*",
            "Condition": {
                "StringNotEquals": {
                    "aws:sourceVpce": "vpce-1234abcdf5678c90a"
                }
            }
        }

    ]
}
```

您也可以使用 aws:sourceVpc 條件索引鍵，以根據 VPC 端點所在的 VPC 來限制對 KMS 金鑰的存取權。

以下範例金鑰政策只在命令來自 vpc-12345678 時，才允許這些命令管理 KMS 金鑰。此外，它只在命令來自 vpc-2b2b2b2b 時，才允許命令將 KMS 金鑰用於密碼編譯操作。如果應用程式在一個 VPC 中執行，但您使用第二個隔離的 VPC 來執行管理功能，您可能會使用如下的政策。

若要使用如下的政策，請將 AWS 帳戶 ID 和 VPC 端點 ID 預留位置換成適用於您帳戶的有效值。

```
{
    "Id": "example-key-2",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Allow administrative actions from vpc-12345678",
            "Effect": "Allow",
            "Principal": {"AWS": "111122223333"},
            "Action": [
                "kms:Create*","kms:Enable*","kms:Put*","kms:Update*",
                "kms:Revoke*","kms:Disable*","kms:Delete*",
                "kms:TagResource", "kms:UntagResource"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:sourceVpc": "vpc-12345678"
                }
            }
        },
        {
            "Sid": "Allow key usage from vpc-2b2b2b2b",
            "Effect": "Allow",
            "Principal": {"AWS": "111122223333"},
            "Action": [
                "kms:Encrypt","kms:Decrypt","kms:GenerateDataKey*"
            ],
            "Resource": "*",
            "Condition": {
                "StringEquals": {
                    "aws:sourceVpc": "vpc-2b2b2b2b"
                }
            }
        },
        {
            "Sid": "Allow read actions from everywhere",
            "Effect": "Allow",
```

```
            "Principal": {"AWS": "111122223333"},
            "Action": [
                "kms:Describe*","kms:List*","kms:Get*"
            ],
            "Resource": "*",
        }
    ]
}
```

記錄您的 VPC 端點

AWS CloudTrail 會記錄所有使用 VPC 端點的操作。若使用 VPC 端點對 AWS KMS 提出請求，則用來記錄請求的 AWS CloudTrail 日誌項目即會顯示 VPC 端點 ID。您可以利用該端點 ID 來稽核 AWS KMS VPC 端點的使用情形。

不過，您的 CloudTrail 記錄不包含其他帳戶中主體要求的作業，也不會包含對其他帳戶中 KMS 金鑰和別名的AWS KMS操作要求。此外，為了保護您的 VPC，VPC 端點政策拒絕 (但否則會被允許) 的請求不會記錄在 AWS CloudTrail 中。

例如，此範例日誌項目中記錄使用 VPC 端點的 GenerateDataKey 請求。vpcEndpointId 欄位出現在日誌項目結尾。

```
{
  "eventVersion":"1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "EX_PRINCIPAL_ID",
    "arn": "arn:aws:iam::111122223333:user/Alice",
    "accessKeyId": "EXAMPLE_KEY_ID",
    "accountId": "111122223333",
    "userName": "Alice"
  },
  "eventTime":"2018-01-16T05:46:57Z",
  "eventSource":"kms.amazonaws.com",
  "eventName":"GenerateDataKey",
  "awsRegion":"eu-west-1",
  "sourceIPAddress":"172.01.01.001",
  "userAgent":"aws-cli/1.14.23 Python/2.7.12 Linux/4.9.75-25.55.amzn1.x86_64
 botocore/1.8.27",
  "requestParameters":{
    "keyId":"1234abcd-12ab-34cd-56ef-1234567890ab",
    "numberOfBytes":128
```

```
  },
  "responseElements":null,
  "requestID":"a9fff0bf-fa80-11e7-a13c-afcabff2f04c",
  "eventID":"77274901-88bc-4e3f-9bb6-acf1c16f6a7c",
  "readOnly":true,
  "resources":[{
    "ARN":"arn:aws:kms:eu-
west-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "accountId":"111122223333",
    "type":"AWS::KMS::Key"
  }],
  "eventType":"AwsApiCall",
  "recipientAccountId":"111122223333",
  "vpcEndpointId": "vpce-1234abcdf5678c90a"
}
```

條件鍵 AWS KMS

您可以在金鑰政策和 IAM 政策中指定控制 AWS KMS 資源存取的條件。政策陳述式只有在符合下列條件時才有效。例如，您可能希望在特定日期之後才套用政策陳述式。或者，您可能會希望政策陳述式只在 API 請求中出現特定的值時才控制存取。

若要指定條件，請在具有 IAM 條件運算子的政策陳述式 Condition 元素中使用條件鍵。某些條件鍵通常適用於 AWS；其他條件鍵則是特定於 AWS KMS。

條件索引鍵值必須符合金 AWS KMS 鑰政策和 IAM 政策的字元和編碼規則。如需有關金鑰政策文件規則的詳細資訊，請參閱金鑰政策格式。如需有關 IAM 政策文件規則的詳細資訊，請參閱《IAM 使用者指南》中的 IAM 名稱需求。

主題

- AWS 全域條件索引鍵
- AWS KMS 條件鍵
- AWS KMSAWS 硝基飛地的條件鍵

AWS 全域條件索引鍵

AWS 定義全域條件金鑰，這是使用 IAM 進行存取控制的所有 AWS 服務的一組政策條件金鑰。 AWS KMS 支持所有全局條件鍵。您可以在 AWS KMS 關鍵政策和 IAM 政策中使用它們。

例如，只有當請求中的主體由條件索引鍵值中的 Amazon 資源名稱 AWS KMS key (ARN) 表示時，您才可以使用 aws: PrincipalArn 全域條件金鑰來允許存取 (KMS 金鑰)。若要在中支援以屬性為基礎的存取控制 (ABAC) AWS KMS，您可以使用 IAM 政策中的 aws:ResourceTag/標籤金鑰全域條件金鑰，允許存取具有特定標記的 KMS 金鑰。

若要協助防止在主體為 AWS 服務主體的原則中將AWS 服務當做混淆的副作用，您可以使用aws:SourceArn或aws:SourceAccount全域條件索引鍵。如需詳細資訊，請參閱 使用 aws:SourceArn 或 aws:SourceAccount 條件金鑰。

如需 AWS 全域條件金鑰的相關資訊，包括可用的請求類型，請參閱 IAM 使用者指南中的AWS 全域條件內容金鑰。如需 IAM 政策中使用全域條件索引鍵的範例，請參閱《IAM 使用者指南》中的控制對請求的存取和控制標籤索引鍵。

下列主題提供的特殊指導，讓您能根據 IP 位址和 VPC 端點來使用條件金鑰。

主題

- 在政策中使用 IP 地址條件搭配 AWS KMS 許可
- 在政策中使用 VPC 端點條件搭配 AWS KMS 許可

在政策中使用 IP 地址條件搭配 AWS KMS 許可

您可以使 AWS KMS 用在整合式 AWS 服務中保護您的資料。但在允許或拒絕存取的相同原則陳述式中指定 IP 位址aws:SourceIp條件運算子或條件金鑰時，請務必 AWS KMS小心。例如，AWS 根據來源 IP 拒絕存取中AWS的原則會將 AWS 動作限制為來自指定 IP 範圍的要求。

考慮以下情形：

1. 您可以將政策附加到 IAM 身分，如下所示 AWS： AWS 根據來源 IP 拒絕存取。您將 aws:SourceIp 條件金鑰的值設定為使用者公司的 IP 地址範圍。這個 IAM 身分還有其他連接的政策，允許他使用 Amazon EBS、Amazon EC2 和 AWS KMS。
2. 該身分嘗試將加密的 EBS 磁碟區連接到 EC2 執行個體。即使使用者有權使用所有相關的服務，這個動作仍會失敗，發生授權錯誤。

步驟 2 失敗，因為 AWS KMS 解密磁碟區的加密資料金鑰的請求來自與 Amazon EC2 基礎設施關聯的 IP 地址。若要成功，請求必須來自原始使用者的 IP 地址。由於步驟 1 中的政策明確拒絕來自指定 IP 地址以外的所有請求，因此 Amazon EC2 被拒絕解密 EBS 磁碟區之加密資料金鑰的許可。

此外，當請求來自 Amazon VPC 端點時，aws:sourceIP 條件索引鍵無效。若要將請求限制為 VPC 端點 (包含 AWS KMS VPC 端點)，請使用 aws:sourceVpce 或 aws:sourceVpc 條件金鑰。如需詳細資訊，請參閱 Amazon VPC 使用者指南中的 VPC 端點 - 控制端點的使用。

在政策中使用 VPC 端點條件搭配 AWS KMS 許可

AWS KMS 支援由AWS PrivateLink提供支援的 Amazon Virtual Private Cloud 端 (Amazon VPC) 端點。當請求來自 VPC 或使用 VPC 端點時，您可以在金鑰政策和 IAM 政策中使用下列全域條件金鑰來控制對 AWS KMS 資源的存取。如需詳細資訊，請參閱 在政策陳述式中使用 VPC 端點。

- aws:SourceVpc 會限制對指定 VPC 所發出之請求的存取權。
- aws:SourceVpce 會限制對指定 VPC 端點所發出之請求的存取權。

如果您使用這些條件金鑰來控制 KMS 金鑰的存取權，可能會不小心拒絕存取代表您使用 AWS KMS 的 AWS 服務。

請注意避免 IP 地址條件金鑰範例這樣的狀況。如果您將 KMS 金鑰的請求限制為 VPC 或 VPC 端點，AWS KMS 從整合式服務 (例如 Amazon S3 或 Amazon EBS) 呼叫可能會失敗。即使來源請求最終源自 VPC 或來自 VPC 端點，也會發生這種情況。

AWS KMS 條件鍵

AWS KMS 提供一組條件金鑰，供您在金鑰政策和 IAM 政策中使用。這些條件索引鍵是特定於 AWS KMS。例如，在控制對稱加密 KMS 金鑰的存取權時，您可以使用 kms:EncryptionContext:*context-key* 條件金鑰來要求特定的加密內容。

API 操作請求的條件

許多 AWS KMS 條件金鑰會根據作業要求中的參數值來控制 KMS 金鑰的存 AWS KMS 取。例如，您可以在 IAM 政策中使用 kms: KeySpec 條件金鑰，只有在CreateKey請求中的KeySpec參數值為時，才允許使用該CreateKey作業RSA_4096。

即使請求中未出現參數 (例如使用參數的預設值時)，這種類型的條件仍會發生作用。例如SYMMETRIC_DEFAULT，您可以使用 kms: KeySpec 條件索引鍵，讓使用者只能在KeySpec參數值為 (預設值) 時使用此CreateKey作業。此條件允許 KeySpec 參數值為 SYMMETRIC_DEFAULT 的請求，以及沒有 KeySpec 參數的請求。

在 API 操作中使用 KMS 金鑰需滿足的條件

某些 AWS KMS 條件金鑰可以根據作業中使用的 KMS 金鑰屬性來控制作業的存取。例如，您可以使用 kms: KeyOrigin 條件，允許主體只有GenerateDataKey在 KMS 金鑰為時才呼叫 KMS 金鑰。Origin AWS_KMS若要確定可否以這種方式使用條件金鑰，請參閱條件金鑰的描述。

操作必須是 KMS 金鑰資源操作，也就是授權特定 KMS 金鑰的操作。若要識別 KMS 金鑰資源操作，請在動作與資源表中尋找操作之 Resources 資料欄的 KMS key 值。如果您將此類型的條件金鑰用於未授權特定 KMS 金鑰資源的作業 (例如) ListKeys，則權限無效，因為無法滿足條件。授權 ListKeys 操作中沒有相關的 KMS 金鑰資源，也沒有 KeySpec 屬性。

下列主題說明每個 AWS KMS 條件索引鍵，並包含示範原則語法的範例原則陳述式。

使用帶有條件索引鍵的集合運算子

當原則條件比較兩組值 (例如要求中的標籤集和原則中的標籤集) 時，您需要告訴 AWS 如何比較這些集合。IAM 為此定義了兩個集合的運算子，ForAnyValue 和 ForAllValues。使用僅具有所需多重值條件索引鍵的集合運算子。請勿將集合運算子與單一值條件索引鍵搭配使用。如往常一樣，在生產環境中使用之前，完整測試您的政策陳述式。

條件索引鍵是單一值或多重值。若要判斷 AWS KMS 條件索引鍵是單值還是多值，請參閱條件索引鍵描述中的「值類型」欄。

- 單一值條件索引鍵在授權內容 (請求或資源) 中最多有一個值。例如，由於每個 API 呼叫只能源自一個 AWS 帳戶，因此 kms: CallerAccount 是單一值的條件金鑰。請勿使用具有單一值條件索引鍵的集合運算子。
- 單一值條件索引鍵在授權內容 (請求或資源) 中有多個值。例如，由於每個 KMS 金鑰可以有多個別名，因此 kms: ResourceAliases 可以有多個值。多重值條件索引鍵需要集合運算子。

請注意，單一值和多重值條件索引鍵之間的差異取決於授權內容中值的數目；而非政策條件中值的數目。

> **Warning**
>
> 將集合運算子搭配單一值條件索引鍵使用，可以建立過度寬鬆 (或過度限制) 的政策陳述式。只能將集合運算子與多值條件索引鍵搭配使用。
> 如果您使用 kms:: 內容索引鍵或*aws:RequestTag/*tag-key條件索引鍵建立或更新包含 *ForAllValues* set 運算子的原則，則會 AWS KMS 傳回下列錯誤訊息EncryptionContext：
> `OverlyPermissiveCondition: Using the ForAllValues set operator with a single-valued condition key matches requests without the specified`

```
    [encryption context or tag] or with an unspecified [encryption
    context or tag]. To fix, remove ForAllValues.
```

如需 ForAnyValue 和 ForAllValues 集合運算子的詳細資訊，請參閱《IAM 使用者指南》中的使用多個金鑰和值。如需將ForAllValues集合運算子搭配單值條件使用風險的相關資訊，請參閱 IAM 使用者指南中 ForAllValues 含有單一值金鑰的安全性警告。

主題

- 公里：BypassPolicyLockoutSafetyCheck
- 公里：CallerAccount
- 公里：CustomerMasterKeySpec（已棄用）
- 公里：CustomerMasterKeyUsage（已棄用）
- 公里：DataKeyPairSpec
- 公里：EncryptionAlgorithm
- 公里:EncryptionContext: 上下文鍵
- 公里：EncryptionContextKeys
- 公里：ExpirationModel
- 公里：GrantConstraintType
- 公里：GrantIsForAWSResource
- 公里：GrantOperations
- 公里：GranteePrincipal
- 公里：KeyOrigin
- 公里：KeySpec
- 公里：KeyUsage
- 公里：MacAlgorithm
- 公里：MessageType
- 公里：MultiRegion
- 公里：MultiRegionKeyType
- 公里：PrimaryRegion
- 公里：ReEncryptOnSameKey

- 公里：RequestAlias
- 公里：ResourceAliases
- 公里：ReplicaRegion
- 公里：RetiringPrincipal
- 公里：ScheduleKeyDeletionPendingWindowInDays
- 公里：SigningAlgorithm
- 公里：ValidTo
- 公里：ViaService
- 公里：WrappingAlgorithm
- 公里：WrappingKeySpec

公里：BypassPolicyLockoutSafetyCheck

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
`kms:BypassPolicyLockoutSafetyCheck`	Boolean	單一值	`CreateKey` `PutKeyPolicy`	僅限 IAM 政策 金鑰政策和 IAM 政策

`kms:BypassPolicyLockoutSafetyCheck`條件索引鍵會根據要求中的`BypassPolicyLockoutSafetyCheck`參數值來控制CreateKey和PutKeyPolicy作業的存取。

以下範例 IAM 政策陳述式在 `CreateKey` 請求中的 `BypassPolicyLockoutSafetyCheck` 參數值是 `true`。時，拒絕使用者建立 KMS 金鑰的許可，來防止使用者繞過政策鎖定安全檢查。

```
{
  "Effect": "Deny",
  "Action": [
    "kms:CreateKey",
    "kms:PutKeyPolicy"
  ],
  "Resource": "*",
```

```
  "Condition": {
    "Bool": {
      "kms:BypassPolicyLockoutSafetyCheck": true
    }
  }
}
```

您也可以在 IAM 政策或金鑰政策中使用 kms:BypassPolicyLockoutSafetyCheck 條件索引鍵，以控制對 PutKeyPolicy 操作的存取。來自金鑰政策的以下範例政策陳述式防止使用者在變更 KMS 金鑰的政策時繞過政策鎖定安全檢查。

除了使用明確 Deny，此政策陳述式使用 Allow 搭配 Null 條件運算子，只在請求不包含 BypassPolicyLockoutSafetyCheck 參數時才允許存取。不使用參數時，預設值為 false。在極少數需要繞過的情況下，您可以覆寫這個比較弱的政策陳述式。

```
{
  "Effect": "Allow",
  "Action": "kms:PutKeyPolicy",
  "Resource": "*",
  "Condition": {
    "Null": {
      "kms:BypassPolicyLockoutSafetyCheck": true
    }
  }
}
```

另請參閱

- 公里：KeySpec
- 公里：KeyOrigin
- 公里：KeyUsage

公里：CallerAccount

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:CallerAccount	字串	單一值	KMS 金鑰資源操作	金鑰政策和 IAM 政策

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
			自訂金鑰存放操作	

您可以使用此條件索引鍵，來允許或拒絕 AWS 帳戶中所有身分 (使用者和角色) 的存取。在金鑰政策中，您可以使用 Principal 元素來指定政策陳述式套用的身分。Principal 元素的語法不提供用來指定 AWS 帳戶中所有身分的方式。但是，您可以通過將此條件鍵與指定所有 AWS 身份的Principal元素結合起來實現此效果。

您可以使用它來控制對任何 KMS 金鑰資源作業的存取，也就是使用特定 KMS 金鑰的任何 AWS KMS 作業。若要識別 KMS 金鑰資源操作，請在動作與資源表中尋找操作之 Resources 資料欄的 KMS key 值。其也適用於管理自訂金鑰存放的操作。

例如，以下金鑰政策陳述式示範了如何使用 kms:CallerAccount 條件索引鍵。此政策聲明列於適用於 Amazon EBS AWS 受管金鑰 的金鑰政策中。它將指定所有 AWS 身分識別的Principal元素與kms:CallerAccount條件索引鍵結合在一起，以便有效地允許存取 AWS 帳戶 111122223333 中的所有身分識別。它包含一個額外的 AWS KMS 條件金鑰 (kms:ViaService)，可透過僅允許透過 Amazon EBS 傳送的請求來進一步限制許可。如需詳細資訊，請參閱 公里：ViaService。

```
{
  "Sid": "Allow access through EBS for all principals in the account that are
 authorized to use EBS",
  "Effect": "Allow",
  "Principal": {"AWS": "*"},
  "Condition": {
    "StringEquals": {
      "kms:CallerAccount": "111122223333",
      "kms:ViaService": "ec2.us-west-2.amazonaws.com"
    }
  },
  "Action": [
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*",
    "kms:CreateGrant",
    "kms:DescribeKey"
  ],
```

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

公里：CustomerMasterKeySpec（已棄用）

kms:CustomerMasterKeySpec 條件索引鍵已被取代。而是使用 kms: KeySpec 條件金鑰。

kms:CustomerMasterKeySpec 和 kms:KeySpec 條件索引鍵的運作方式相同。只有名稱不同。建議您使用 kms:KeySpec。但是，為了避免中斷更改， AWS KMS 支持兩個條件鍵。

公里：CustomerMasterKeyUsage（已棄用）

kms:CustomerMasterKeyUsage 條件索引鍵已被取代。而是使用 kms: KeyUsage 條件金鑰。

kms:CustomerMasterKeyUsage 和 kms:KeyUsage 條件索引鍵的運作方式相同。只有名稱不同。建議您使用 kms:KeyUsage。但是，為了避免中斷更改， AWS KMS 支持兩個條件鍵。

公里：DataKeyPairSpec

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:DataKeyPairSpec	字串	單一值	GenerateDataKeyPair GenerateDataKeyPairWithoutPlaintext	金鑰政策和 IAM 政策

您可以使用此條件索引鍵，根據請求中的KeyPairSpec參數值來控制對GenerateDataKeyPair和GenerateDataKeyPairWithoutPlaintext作業的存取。例如，您可以允許使用者產生僅限特定類型的資料金鑰對。

下列範例金鑰政策陳述式使用 kms:DataKeyPairSpec 條件索引鍵，允許使用者使用 KMS 金鑰只產生 RSA 資料金鑰對。

```
{
```

```
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
  },
  "Action": [
    "kms:GenerateDataKeyPair",
    "kms:GenerateDataKeyPairWithoutPlaintext"
  ],
  "Resource": "*",
  "Condition": {
    "StringLike": {
      "kms:DataKeyPairSpec": "RSA*"
    }
  }
}
```

另請參閱

- 公里：KeySpec
- the section called “公里：EncryptionAlgorithm”
- the section called “公里:EncryptionContext: 上下文鍵”
- the section called “公里：EncryptionContextKeys”

公里：EncryptionAlgorithm

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:EncryptionAlgorithm	字串	單一值	Decrypt Encrypt GenerateDataKey GenerateDataKeyPair GenerateDataKeyPai	金鑰政策和 IAM 政策

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
			rWithoutPlaintext GenerateDataKeyWithoutPlaintext ReEncrypt	

您可以使用 kms:EncryptionAlgorithm 條件金鑰，根據操作中使用的加密演算法控制對密碼編譯操作的存取。對於「加密」、「解密」和ReEncrypt作業，它會根據要求中的EncryptionAlgorithm參數值來控制存取。對於產生資料金鑰和資料金鑰對的操作，根據加密資料金鑰所用之加密演算法控制存取。

此條件金鑰不會影響以外執行的作業 AWS KMS，例如使用非對稱 KMS 金鑰組中的公開金 key pair 加密。 AWS KMS

EncryptionAlgorithm 請求中的參數

若要允許使用者只使用特定加密演算法搭配 KMS 金鑰，請使用具有 Deny 效果的政策陳述式和 StringNotEquals 條件運算子。例如，下列範例金鑰政策陳述式禁止擔任 ExampleRole 角色的主體在指定的密碼編譯操作中使用此 KMS 金鑰，除非請求中的加密演算法是 RSAES_OAEP_SHA_256 (即與 RSA KMS 金鑰搭配使用的非對稱加密演算法)。

與允許使用者使用特定加密演算法的政策陳述式不同，具有像這樣的雙負數的政策陳述式可防止此 KMS 金鑰的其他政策和授權讓此角色使用其他加密演算法。此金鑰政策陳述式中的 Deny 優先於任何金鑰政策或具有 Allow 效果的 IAM 政策，也優先於此 KMS 金鑰及其主體的所有授權。

```
{
  "Sid": "Allow only one encryption algorithm with this asymmetric KMS key",
  "Effect": "Deny",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
  },
  "Action": [
    "kms:Encrypt",
```

```
    "kms:Decrypt",
    "kms:ReEncrypt*"
  ],
  "Resource": "*",
  "Condition": {
    "StringNotEquals": {
      "kms:EncryptionAlgorithm": "RSAES_OAEP_SHA_256"
    }
  }
}
```

用於操作的加密演算法

您也可以使用 kms:EncryptionAlgorithm 條件索引鍵，即使在請求中未指定演算法，也可根據操作中使用的加密演算法來控制對操作的存取。這可讓您需要或禁止 SYMMETRIC_DEFAULT 演算法，這可能不會在請求中指定，因為其是預設值。

此功能能讓您使用 kms:EncryptionAlgorithm 條件索引鍵控制對產生資料金鑰和資料金鑰對之操作的存取。這些操作只使用對稱加密 KMS 金鑰和 SYMMETRIC_DEFAULT 演算法。

例如，此 IAM 政策會將其主體限制在對稱加密。除非請求中指定或操作中使用的加密演算法是 SYMMETRIC_DEFAULT，否則拒絕存取範例帳戶中密碼編譯操作的任何 KMS 金鑰。包括GenerateDataKey*添加GenerateDataKeyGenerateDataKeyWithoutPlaintextGenerateDataKeyPair、、和GenerateDataKeyPairWithoutPlaintext權限。條件對這些操作沒有影響，因為其一律使用對稱式加密演算法。

```
{
  "Sid": "AllowOnlySymmetricAlgorithm",
  "Effect": "Deny",
  "Action": [
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*"
  ],
  "Resource": "arn:aws:kms:us-west-2:111122223333:key/*",
  "Condition": {
    "StringNotEquals": {
      "kms:EncryptionAlgorithm": "SYMMETRIC_DEFAULT"
    }
  }
```

```
}
```

另請參閱

- the section called “公里：MacAlgorithm”
- 公里：SigningAlgorithm

公里:EncryptionContext: 上下文鍵

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:EncryptionContext: *context-key*	字串	單一值	CreateGrant Encrypt Decrypt GenerateDataKey GenerateDataKeyPair GenerateDataKeyPairWithoutPlaintext GenerateDataKeyWithoutPlaintext ReEncrypt	金鑰政策和 IAM 政策

您可以使用 kms:EncryptionContext:*context-key* 條件金鑰，依據密碼編譯操作請求中的加密內容，控制對對稱加密 KMS 金鑰的存取。使用此條件索引鍵可同時評估加密內容對中

的金鑰和值。若只要評估加密內容金鑰，或不論金鑰或值為何，都需要加密內容，請使用 kms:EncryptionContextKeys 條件金鑰。

> **Note**
>
> 條件金鑰值必須符合金鑰政策和 IAM 政策的字元規則。某些在加密內容中有效的字元在政策中無效。您可能無法使用此條件金鑰來表示所有的有效加密內容值。如需有關金鑰政策文件規則的詳細資訊，請參閱金鑰政策格式。如需有關 IAM 政策文件規則的詳細資訊，請參閱《IAM 使用者指南》中的 IAM 名稱需求。

您不能使用非對稱 KMS 金鑰或 HMAC KMS 金鑰在密碼編譯操作中指定加密內容。非對稱演算法和 MAC 演算法不支援加密內容。

若要使用 kms:EncryptionContext: 內容金鑰條件金鑰，請將內容金鑰預留位置取代為加密*####*。將 *context-value* 預留位置取代為加密內容值。

```
"kms:EncryptionContext:context-key": "context-value"
```

例如，以下條件索引鍵指定加密內容，其中金鑰是 AppName，值是 ExampleApp (AppName = ExampleApp)。

```
"kms:EncryptionContext:AppName": "ExampleApp"
```

這是單一值條件索引鍵。條件索引鍵中的金鑰會指定特定的加密內容索引鍵 (context-key)。雖然您可以在每個 API 請求中包含多個加密內容對，但是與指定 context-key 對的加密內容只能有一個值。例如，kms:EncryptionContext:Department 條件索引鍵僅適用於與 Department 金鑰對的加密內容，以及任何與 Department 金鑰對的指定加密內容只能有一個值。

請勿將集合運算子與 kms:EncryptionContext:*context-key* 條件索引鍵搭配使用。如果您建立的政策陳述式包含 Allow 動作、kms:EncryptionContext:*context-key* 條件索引鍵和 ForAllValues 集合運算子，則條件會允許沒有加密內容的請求，以及未在政策條件中指定的加密內容對的請求。

> **Warning**
>
> 請勿將 ForAnyValue 或 ForAllValues 集合運算子與此單一值條件索引鍵搭配使用。這些集合運算子可以建立不需要您想要要求之值的政策條件，並允許您想要禁止的值。

如果您使用 kms:: 內容索引鍵建立或更新包含 ForAllValues set 運算子的原則，則會 AWS KMS 傳回下列錯誤訊息EncryptionContext：

```
OverlyPermissiveCondition:EncryptionContext: Using the ForAllValues
set operator with a single-valued condition key matches requests
without the specified encryption context or with an unspecified
encryption context. To fix, remove ForAllValues.
```

若要要求特定的加密內容對，請將 kms:EncryptionContext:*context-key* 條件索引鍵與 StringEquals 運算子搭配使用。

以下範例金鑰政策陳述式，僅當請求中的加密內容包含 AppName:ExampleApp 對時，才允許可以擔任角色的主體使用 GenerateDataKey 請求中的 KMS 金鑰。允許其他加密內容對。

金鑰名稱不會區分大小寫。值是否區分大小寫，取決於條件運算子，例如 StringEquals。如需詳細資訊，請參閱 加密內容條件需區分大小寫。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": "kms:GenerateDataKey",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:EncryptionContext:AppName": "ExampleApp"
    }
  }
}
```

若要要求加密內容配對並禁止所有其他加密內容配對，請同時使用 kms:EncryptionContext: 內容金鑰和原則陳述kms:EncryptionContextKeys式。以下範例政策陳述式使用 kms:EncryptionContext:AppName 條件索引鍵，要求請求中有 AppName=ExampleApp 加密內容對。它還會將 kms:EncryptionContextKeys 條件索引鍵與 ForAllValues 集合運算子搭配使用，以僅允許 AppName 加密內容索引鍵。

ForAllValues 集合運算子將請求中的加密內容索引鍵限制為 AppName。如果具有 ForAllValues 集合運算子的 kms:EncryptionContextKeys 條件在政策陳述式中單獨使用，則此集合運算子會允許沒有加密內容的請求。不過，如果請求沒有加密內容，則 kms:EncryptionContext:AppName 條

件將會失敗。如需 ForAllValues 集合運算子的詳細資訊，請參閱《IAM 使用者指南》中的使用多個索引鍵和值。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/KeyUsers"
  },
  "Action": "kms:GenerateDataKey",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:EncryptionContext:AppName": "ExampleApp"
    },
    "ForAllValues:StringEquals": {
      "kms:EncryptionContextKeys": [
        "AppName"
      ]
    }
  }
}
```

您也可以使用此條件索引鍵來拒絕存取特定操作的 KMS 金鑰。以下範例金鑰政策陳述式使用 Deny 效果，在請求中加密內容包含 Stage=Restricted 加密內容對的情形下，禁止主體使用 KMS 金鑰。此條件允許使用其他加密內容對的請求，包括具有 Stage 金鑰和其他值的加密內容對，例如 Stage=Test。

```
{
  "Effect": "Deny",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": "kms:GenerateDataKey",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:EncryptionContext:Stage": "Restricted"
    }
  }
}
```

使用多個加密內容對

您可以要求或禁止多個加密內容對。您也可以要求其中一個加密內容對。如需用來解譯這些條件之邏輯的詳細資訊，請參閱《IAM 使用者指南》中的建立具有多個索引鍵或值的條件。

> **Note**
>
> 本主題的舊版顯示使用ForAnyValue和ForAllValues設定運算子搭配 kms:EncryptionContext: 內容索引鍵條件金鑰的原則陳述式。使用具有單一值條件索引鍵的集合運算子可能會導致政策允許沒有加密內容和未指定加密內容對的請求。
> 例如，政策條件具有 Allow 效果、ForAllValues 集合運算子和 "kms:EncryptionContext:Department": "IT" 條件索引鍵不會將加密內容限制為 "Department=IT" 對。它允許沒有加密內容的請求和未指定加密內容對的請求，例如 Stage=Restricted。
> 請檢閱您的政策，並使用 kms:EncryptionContext: 上下文鍵從任何條件中消除 set 運算子。嘗試使用此格式建立或更新政策會失敗，OverlyPermissiveCondition 為例外狀況。若要解決錯誤，請刪除集合運算子。

若要要求多個加密內容對，請以相同的條件列出對。以下範例金鑰政策陳述式需要兩個加密內容對，Department=IT 和 Project=Alpha。因為條件有不同的索引鍵 (kms:EncryptionContext:Department 和 kms:EncryptionContext:Project)，所以它們由 AND 運算子隱式連線。允許其他加密內容對，但不是必需的。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": "kms:Decrypt",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:EncryptionContext:Department": "IT",
      "kms:EncryptionContext:Project": "Alpha"
    }
  }
}
```

若需要一個加密內容對 OR 另一個對，請將每個條件索引鍵放在個別的政策陳述式中。以下範例金鑰政策需要 Department=IT 或 Project=Alpha 對，或兩者。允許其他加密內容對，但不是必需的。

```
{
 "Effect": "Allow",
 "Principal": {
  "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
 },
 "Action": "kms:GenerateDataKey",
 "Resource": "*",
 "Condition": {
  "StringEquals": {
   "kms:EncryptionContext:Department": "IT"
  }
 }
},
{
 "Effect": "Allow",
 "Principal": {
  "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
 },
 "Action": "kms:GenerateDataKey",
 "Resource": "*",
 "Condition": {
  "StringEquals": {
   "kms:EncryptionContext:Project": "Alpha"
  }
 }
}
```

若要需要特定的加密配對並排除所有其他加密內容配對，請同時使用 kmsEncryptionContext:: 內容金鑰和原則陳述kms:EncryptionContextKeys式。下列金鑰原則陳述式會使用 kms:EncryptionContext: 內容金鑰條件來要求具有Department=IT和Project=Alpha配對的加密內容。它會將 kms:EncryptionContextKeys 條件索引鍵與 ForAllValues 集合運算子搭配使用，以僅允許 Department 和 Project 加密內容索引鍵。

ForAllValues 集合運算子將請求中的加密內容索引鍵限制為 Department 和 Project。如果在條件下單獨使用它，則此 set 運算符將允許沒有加密內容的請求，但在此配置中，此情況下的 kmsEncryptionContext:: 上下文密鑰將會失敗。

```
{
  "Effect": "Allow",
```

```
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": "kms:GenerateDataKey",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:EncryptionContext:Department": "IT",
      "kms:EncryptionContext:Project": "Alpha"
    },
    "ForAllValues:StringEquals": {
      "kms:EncryptionContextKeys": [
        "Department",
        "Project"
      ]
    }
  }
}
```

您也可以禁止多個加密內容對。以下範例金鑰政策陳述式使用 Deny 效果，在請求中的加密內容包含 Stage=Restricted 和 Stage=Production 對的情形下，禁止主體使用 KMS 金鑰。

相同索引鍵 (kms:EncryptionContext:Stage) 的多個值 (Restricted 和 Production) 由 OR 隱式連線。如需詳細資訊，請參閱《IAM 使用者指南》中的具有多個索引鍵或值之條件的評估邏輯。

```
{
  "Effect": "Deny",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": "kms:GenerateDataKey",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:EncryptionContext:Stage": [
         "Restricted",
         "Production"
      ]
    }
  }
}
```

加密內容條件需區分大小寫

解密操作中指定的加密內容和加密操作中指定的加密內容必須完全一樣，且大小寫相符。只有具有多對之加密內容中的配對順序可以改變。

不過，在政策條件中，條件金鑰不區分大小寫。條件值是否區分大小寫，取決於您使用的政策條件運算子，例如 StringEquals 或 StringEqualsIgnoreCase。

因此，由 kms:EncryptionContext: 字首和 *context-key* 替換組成的條件金鑰不區分大小寫。使用此條件的政策不會檢查條件金鑰任一元素的大小寫。值 (亦即 *context-value* 替換) 是否區分大小寫，取決於政策條件運算子。

例如，以下政策陳述式允許操作的加密內容包含 Appname 金鑰，無論其大小寫。StringEquals 條件要求 ExampleApp 符合指定的大寫形式。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": "kms:Decrypt",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:EncryptionContext:Appname": "ExampleApp"
    }
  }
}
```

若要需要區分大小寫的加密內容金鑰，請使用 kms: EncryptionContextKeys 原則條件搭配區分大小寫的條件運算子，例如。StringEquals在這個政策條件中，因為加密內容索引鍵是此政策條件的值，所以其是否區分大小寫取決於條件運算子。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": "kms:GenerateDataKey",
  "Resource": "*",
  "Condition": {
```

```
    "ForAnyValue:StringEquals": {
      "kms:EncryptionContextKeys": "AppName"
    }
  }
}
```

若要對加密內容金鑰和值進行區分大小寫的評估，請在相同的原則陳述式中同時使用kms:EncryptionContextKeys和 kmsEncryptionContext:: 內容索引鍵原則條件。區分大小寫的條件運算子 (例如 StringEquals) 一律適用於條件的值。加密內容索引鍵 (例如 AppName) 是 kms:EncryptionContextKeys 條件的值。加密內容值 (例如ExampleApp) 是 kms:EncryptionContext: 內容索引鍵條件的值。

例如，在以下範例政策陳述式中，因為 StringEquals 運算子區分大小寫，所以加密內容索引鍵和加密內容值會區分大小寫。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": "kms:GenerateDataKey",
  "Resource": "*",
  "Condition": {
    "ForAnyValue:StringEquals": {
      "kms:EncryptionContextKeys": "AppName"
    },
    "StringEquals": {
      "kms:EncryptionContext:AppName": "ExampleApp"
    }
  }
}
```

在加密內容條件中使用變數

加密內容對中的金鑰和值必須是簡單的常值字串。它們不能是整數或物件，或任何未完全解析的類型。如果您使用不同的類型 (例如整數或浮點數)，則會將其 AWS KMS 解譯為常值字串。

```
"encryptionContext": {
    "department": "10103.0"
}
```

不過，kms:EncryptionContext:*context-key* 條件索引鍵的值可以是 IAM 政策變數。這些政策變數會根據要求中的值在執行時間中解析。例如，aws:CurrentTime 解析為請求的時間，而 aws:username 解析為易記的發起人名稱。

您可以使用這些政策變數來建立政策陳述式，其條件加密內容中需要非常特定的資訊，例如發起人的使用者名稱。由於它包含變數，因此您可以對可擔任角色的所有使用者使用相同的政策陳述式。您不需要為每個使用者個別編寫政策陳述式。

請考慮以下情況，您想要所有可擔任角色的使用者使用相同的 KMS 金鑰來加密和解密其資料。不過，您只想要允許他們解密由他們加密的資料。首先要求每個請求都 AWS KMS 包含密鑰所在的加密上下文，user而該值是調 AWS 用者的用戶名，例如以下內容。

```
"encryptionContext": {
    "user": "bob"
}
```

然後，若要強制執行此需求，您可以使用類似以下範例中的政策陳述式。此政策陳述式提供 TestTeam 角色使用 KMS 金鑰加密和解密資料的許可。不過，許可只有在請求中的加密內容包含 "user": "*<username>*" 對時才有效。為了代表使用者名稱，條件使用 aws:username 政策變數。

評估請求時，發起人的使用者名稱會取代條件中的變數。因此，條件需要 "user": "bob" 的「bob」和 "user": "alice" 的「alice」加密內容。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/TestTeam"
  },
  "Action": [
    "kms:Decrypt",
    "kms:Encrypt"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:EncryptionContext:user": "${aws:username}"
    }
  }
}
```

您只能在 kms:EncryptionContext:*context-key* 條件索引鍵的值中使用 IAM 政策變數。您不能在金鑰中使用變數。

您也可以在變數中使用供應商特定的內容金鑰。這些內容索引鍵會唯一識別使用 Web 身分聯盟登入 AWS 的使用者。

如同所有變數，這些變數只能用於 kms:EncryptionContext:*context-key* 政策條件，不能用於實際的加密內容。而且只能用於條件的值，不能用於金鑰。

例如，下列金鑰政策陳述式與前一個陳述式類似。不過，條件需要加密內容，其金鑰是 sub 而值是可唯一識別登入 Amazon Cognito 使用者集區的使用者。如需有關在 Amazon Cognito 中識別使用者和角色的詳細資訊，請參閱《Amazon Cognito 開發人員指南》中的 IAM 角色。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/TestTeam"
  },
  "Action": [
    "kms:Decrypt",
    "kms:Encrypt"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
       "kms:EncryptionContext:sub": "${cognito-identity.amazonaws.com:sub}"
    }
  }
}
```

另請參閱

- the section called “公里：EncryptionContextKeys”
- the section called “公里：GrantConstraintType”

公里：EncryptionContextKeys

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)

`kms:EncryptionContextKeys`	字串 (清單)	多重值	`CreateGrant`
			`Decrypt`
			`Encrypt`
			`GenerateDataKey`
			`GenerateDataKeyPair`
			`GenerateDataKeyPairWithoutPlaintext`
			`GenerateDataKeyWithoutPlaintext`
			`ReEncrypt`
				金鑰政策和 IAM 政策

您可以使用 `kms:EncryptionContextKeys` 條件金鑰，依據密碼編譯操作請求中的加密內容，控制對對稱加密 KMS 金鑰的存取。使用此條件索引鍵，只評估每個加密內容對中的金鑰。若要同時評估加密內容中的金鑰和值，請使用 `kms:EncryptionContext:`*`context-key`* 條件索引鍵。

您不能使用非對稱 KMS 金鑰或 HMAC KMS 金鑰在密碼編譯操作中指定加密內容。非對稱演算法和 MAC 演算法不支援加密內容。

> **Note**
>
> 條件索引鍵值 (包括加密內容金鑰) 必須符合金 AWS KMS 鑰原則的字元和編碼規則。您可能無法使用此條件金鑰來表示所有的有效加密內容金鑰。如需有關金鑰政策文件規則的詳細資訊，請參閱金鑰政策格式。如需有關 IAM 政策文件規則的詳細資訊，請參閱《IAM 使用者指南》中的 IAM 名稱需求。

這是多重值條件索引鍵。您可以在每個 API 請求中指定多個加密內容對。`kms:EncryptionContextKeys` 會將請求中的加密內容索引鍵與政策中的加密內容索引鍵集合進行比較。若要判斷這些集合的比較方式，您必須提供政策條件中的 `ForAnyValue` 或 `ForAllValues` 集合運算子。如需集合運算子的詳細資訊，請參閱《IAM 使用者指南》中的使用多個索引鍵和值。

- `ForAnyValue`：請求中至少有一個加密內容索引鍵必須符合政策條件中的加密內容索引鍵。允許使用其他加密內容索引鍵。如果請求沒有加密內容，則不符合條件。
- `ForAllValues`：請求中每個加密內容索引鍵必須符合政策條件中的加密內容索引鍵。此集合運算子會將加密內容索引鍵限制為政策條件中的索引鍵。它不需要任何加密內容索引鍵，但它禁止未指定的加密內容索引鍵。

以下範例金鑰政策陳述式將 `kms:EncryptionContextKeys` 條件索引鍵與 `ForAnyValue` 集合運算子搭配使用。此政策陳述式僅在請求中的至少一個加密內容對包含 `AppName` 金鑰時 (無論其值為何)，才允許使用指定操作的 KMS 金鑰。

例如，此金鑰政策陳述式允許具有兩個加密內容對、`AppName=Helper` 和 `Project=Alpha` 的 `GenerateDataKey` 請求，因為第一個加密內容對符合條件。僅具有 `Project=Alpha` 或沒有加密內容的請求將失敗。

由於StringEquals條件作業區分大小寫，因此此原則陳述式需要加密內容金鑰的拼字和大小寫。但是，您也可以使用忽略金鑰大小寫的條件運算子，例如 `StringEqualsIgnoreCase`。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": [
    "kms:Encrypt",
```

```
    "kms:GenerateDataKey*"
  ],
  "Resource": "*",
  "Condition": {
    "ForAnyValue:StringEquals": {
      "kms:EncryptionContextKeys": "AppName"
    }
  }
}
```

您也可以使用 kms:EncryptionContextKeys 條件索引鍵來要求使用 KMS 金鑰之密碼編譯操作中的加密內容 (任何加密內容)。

以下範例金鑰政策陳述式使用 kms:EncryptionContextKeys 條件索引鍵搭配 Null 條件運算子，僅在 API 請求中存在加密內容 (不是 null) 時，才允許存取 KMS 金鑰。此情況不會檢查加密內容的索引鍵或值。它只會驗證加密內容是否存在。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": [
    "kms:Encrypt",
    "kms:GenerateDataKey*"
  ],
  "Resource": "*",
  "Condition": {
    "Null": {
      "kms:EncryptionContextKeys": false
    }
  }
}
```

另請參閱

- 公里:EncryptionContext: 上下文鍵
- 公里：GrantConstraintType

公里：ExpirationModel

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:ExpirationModel	字串	單一值	ImportKeyMaterial	金鑰政策和 IAM 政策

kms:ExpirationModel條件索引鍵會根據要求中的ExpirationModel參數值控制對ImportKeyMaterial作業的存取。

ExpirationModel 是選擇性參數，用來決定匯入的金鑰材料是否過期。有效值為 KEY_MATERIAL_EXPIRES 和 KEY_MATERIAL_DOES_NOT_EXPIRE。KEY_MATERIAL_EXPIRES 為預設值。

到期日期和時間由ValidTo參數的值決定。除非 ExpirationModel 參數的值是 KEY_MATERIAL_DOES_NOT_EXPIRE，否則 ValidTo 參數為必要。您也可以使用 kms: ValidTo condition 金鑰來要求特定的到期日作為存取條件。

以下範例政策陳述式使用 kms:ExpirationModel 條件索引鍵，只在請求包含 ExpirationModel 參數且其值為 KEY_MATERIAL_DOES_NOT_EXPIRE 時，才允許使用者將金鑰材料匯入 KMS 金鑰。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": "kms:ImportKeyMaterial",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:ExpirationModel": "KEY_MATERIAL_DOES_NOT_EXPIRE"
    }
  }
}
```

您也可以使用 kms:ExpirationModel 條件索引鍵，只在金鑰材料過期時才允許使用者匯入金鑰材料。以下範例金鑰政策陳述式使用 kms:ExpirationModel 條件索引鍵搭配 Null 條件運算子，僅

當請求沒有 ExpirationModel 參數時，才允許使用者匯入金鑰材料。的預設值 ExpirationModel 為KEY_MATERIAL_EXPIRES。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": "kms:ImportKeyMaterial",
  "Resource": "*",
  "Condition": {
    "Null": {
      "kms:ExpirationModel": true
    }
  }
}
```

另請參閱

- 公里：ValidTo
- 公里：WrappingAlgorithm
- 公里：WrappingKeySpec

公里：GrantConstraintType

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:GrantConstraintType	字串	單一值	CreateGrant	金鑰政策和 IAM 政策

您可以使用此條件索引鍵，根據要求中的 grant 條件約束類型來控制對CreateGrant作業的存取。

建立授與時，您可以選擇指定授與限制以允許操作，僅在特定加密內容存在時才授與許可。授與限制可以是兩種類型其中之一：EncryptionContextEquals 或 EncryptionContextSubset。您可以使用此條件金鑰來檢查請求是否包含其中一個類型。

> **Important**
>
> 請勿在此欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，此欄位可能會以純文字顯示。

以下範例金鑰政策陳述式使用 `kms:GrantConstraintType` 條件索引鍵，只在請求包含 `EncryptionContextEquals` 授予限制條件時才允許使用者建立授予。此範例顯示金鑰政策中的政策陳述式。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/RoleForExampleApp"
  },
  "Action": "kms:CreateGrant",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:GrantConstraintType": "EncryptionContextEquals"
    }
  }
}
```

另請參閱

- 公里:EncryptionContext: 上下文鍵
- 公里：EncryptionContextKeys
- 公里：GrantIsForAWSResource
- 公里：GrantOperations
- 公里：GranteePrincipal
- 公里：RetiringPrincipal

公里：GrantIsForAWSResource

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)

`kms:GrantIsForAWSResource`	Boolean	單一值	`CreateGrant`
			`ListGrants`
			`RevokeGrant`
				金鑰政策和 IAM 政策

只有在與整合的CreateGrantAWS 服務代表使用者 AWS KMS呼叫作RevokeGrant業時 ListGrants，才允許或拒絕、或作業的權限。此政策條件不允許使用者直接呼叫這些授予操作。

以下範例金鑰政策陳述式會使用 `kms:GrantIsForAWSResource` 條件索引鍵。它可讓整合的 AWS AWS KMS服務 (例如 Amazon EBS) 代表指定的主體在此 KMS 金鑰上建立授權。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
  },
  "Action": "kms:CreateGrant",
  "Resource": "*",
  "Condition": {
    "Bool": {
      "kms:GrantIsForAWSResource": true
    }
  }
}
```

另請參閱

- 公里：GrantConstraintType
- 公里：GrantOperations
- 公里：GranteePrincipal
- 公里：RetiringPrincipal

公里：GrantOperations

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:GrantOperations	字串	多重值	CreateGrant	金鑰政策和 IAM 政策

您可以使用此條件索引鍵，根據要求中的授與CreateGrant作業控制對作業的存取。例如，您可以允許使用者建立授予，以委派加密 (但不是解密) 的許可。如需授予的詳細資訊，請參閱使用授予。

這是多重值條件索引鍵。kms:GrantOperations 會將 CreateGrant 請求中的一組授予操作與政策中的一組授予操作進行比較。若要判斷這些集合的比較方式，您必須提供政策條件中的 ForAnyValue 或 ForAllValues 集合運算子。如需集合運算子的詳細資訊，請參閱《IAM 使用者指南》中的使用多個索引鍵和值。

- ForAnyValue：請求中至少有一個授予操作必須符合政策條件中的其中一個授予操作。允許其他授予操作。
- ForAllValues：要求中的每個授權作業都必須符合原則條件中的授權作業。此集合運算子會將授予操作限制為政策條件中指定的操作。它不需要任何授予操作，但它禁止未指定的授予操作。

  ForAllValues 當請求中沒有授權操作時，也返回 true，但CreateGrant不允許它。如果 Operations 參數遺失或具有空值時，則 CreateGrant 請求會失敗。

以下範例政策陳述式使用 kms:GrantOperations 條件索引鍵，僅當授予操作為 Encrypt、ReEncryptTo 或兩者時，才允許建立授予。如果授予包含任何其他操作，則 CreateGrant 請求會失敗。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
  },
  "Action": "kms:CreateGrant",
  "Resource": "*",
  "Condition": {
    "ForAllValues:StringEquals": {
      "kms:GrantOperations": [
```

```
        "Encrypt",
        "ReEncryptTo"
      ]
    }
  }
}
```

如果您將政策條件中的集合運算子變更為 `ForAnyValue`，則政策陳述式會要求授予中至少有一個授予操作為 `Encrypt` 或 `ReEncryptTo`，但它會允許其他授予操作，例如 `Decrypt` 或 `ReEncryptFrom`。

另請參閱

- 公里：GrantConstraintType
- 公里：GrantIsForAWSResource
- 公里：GranteePrincipal
- 公里：RetiringPrincipal

公里：GranteePrincipal

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
`kms:GranteePrincipal`	字串	單一值	`CreateGrant`	IAM 和金鑰政策

您可以使用此條件鍵，根據請求中的GranteePrincipal參數值來控制對CreateGrant作業的存取。例如，您可以只在 `CreateGrant` 請求中的承授者主體符合條件陳述式中指定的主體時，才可建立授予來使用 KMS 金鑰。

若要指定受權者主體，請使用主體的 Amazon 資源名稱 (ARN)。 AWS 有效的主體包括 IAM 使用者 AWS 帳戶、IAM 角色、聯合身分使用者和假定角色使用者。如需主體 ARN 語法的相關說明，請參閱 IAM 使用者指南中的 IAM ARN。

以下範例政策陳述式使用 `kms:GranteePrincipal` 條件索引鍵，僅當授予中的承授者主體是 `LimitedAdminRole` 時，才可建立 KMS 金鑰的授予。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
  },
  "Action": "kms:CreateGrant",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:GranteePrincipal": "arn:aws:iam::111122223333:role/LimitedAdminRole"
    }
  }
}
```

另請參閱

- 公里：GrantConstraintType
- 公里：GrantIsForAWSResource
- 公里：GrantOperations
- 公里：RetiringPrincipal

公里：KeyOrigin

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:KeyOrigin	字串	單一值	CreateKey KMS 金鑰資源操作	IAM 政策 金鑰政策和 IAM 政策

kms:KeyOrigin 條件索引鍵會根據此操作建立或使用之 KMS 金鑰的 Origin 屬性值，控制對操作的存取。以資源條件或請求條件運作。

您可以使用此條件鍵，根據請求中 Origin 參數的值來控制對CreateKey作業的存取。Origin 的有效值為 AWS_KMS、AWS_CLOUDHSM 和 EXTERNAL。

例如，您只能在 () 中產生金鑰材料時建立 KMS 金鑰，只有在與自訂金鑰存放區 AWS KMS (AWS_KMS) 相關聯的 AWS CloudHSM 叢集中產生金鑰材料時，或僅當金鑰材料是從外部來源 (EXTERNAL) 匯入時才建立 KMS 金鑰。AWS_CLOUDHSM

下列範例金鑰原則陳述式只 AWS KMS 會在建立金鑰材料時使用kms:KeyOrigin條件金鑰來建立 KMS 金鑰。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
      },
      "Action": "kms:CreateKey",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:KeyOrigin": "AWS_KMS"
        }
      }
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
      },
      "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:GenerateDataKey",
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:GenerateDataKeyPair",
        "kms:GenerateDataKeyPairWithoutPlaintext",
        "kms:ReEncrypt*"
      ],
      "Resource": "arn:aws:kms:us-west-2:111122223333:key/*",
      "Condition": {
        "StringEquals": {
          "kms:KeyOrigin": "AWS_CLOUDHSM"
        }
      }
```

```
    }
  ]
}
```

您也可以使用 kms:KeyOrigin 條件索引鍵，根據用於操作之 KMS 金鑰的 Origin 屬性控制對使用或管理 KMS 金鑰之操作的存取。操作必須是 KMS 金鑰資源操作，也就是授權特定 KMS 金鑰的操作。若要識別 KMS 金鑰資源操作，請在動作與資源表中尋找操作之 Resources 資料欄的 KMS key 值。

例如，下列 IAM 政策允許主體執行指定的 KMS 金鑰資源操作，但僅能使用帳戶中於自訂金鑰存放區中建立的 KMS 金鑰。

```
{
  "Effect": "Allow",
  "Action": [
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:GenerateDataKey",
    "kms:GenerateDataKeyWithoutPlaintext",
    "kms:GenerateDataKeyPair",
    "kms:GenerateDataKeyPairWithoutPlaintext",
    "kms:ReEncrypt*"
  ],
  "Resource": "arn:aws:kms:us-west-2:111122223333:key/*",
  "Condition": {
    "StringEquals": {
      "kms:KeyOrigin": "AWS_CLOUDHSM"
    }
  }
}
```

另請參閱

- 公里：BypassPolicyLockoutSafetyCheck
- 公里：KeySpec
- 公里：KeyUsage

公里：KeySpec

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:KeySpec	字串	單一值	CreateKey KMS 金鑰資源操作	IAM 政策 金鑰政策和 IAM 政策

kms:KeySpec 條件索引鍵會根據此操作建立或使用之 KMS 金鑰的 KeySpec 屬性值，控制對操作的存取。

您可以在 IAM 政策中使用此條件金鑰，根據請求中的KeySpec參數值控制對CreateKey作業的存CreateKey取。例如，您可以使用此條件，允許使用者僅建立對稱加密 KMS 金鑰，或僅建立 HMAC KMS 金鑰。

下列範例 IAM 政策陳述式會使用 kms:KeySpec 條件金鑰，允許主體僅建立 RSA 非對稱 KMS 金鑰。僅在請求中的 KeySpec 以 RSA_ 開頭時，許可才有效。

```
{
  "Effect": "Allow",
  "Action": "kms:CreateKey",
  "Resource": "*",
  "Condition": {
    "StringLike": {
      "kms:KeySpec": "RSA_*"
    }
  }
}
```

您也可以使用 kms:KeySpec 條件索引鍵，根據用於操作之 KMS 金鑰的 KeySpec 屬性控制對使用或管理 KMS 金鑰之操作的存取。操作必須是 KMS 金鑰資源操作，也就是授權特定 KMS 金鑰的操作。若要識別 KMS 金鑰資源操作，請在動作與資源表中尋找操作之 Resources 資料欄的 KMS key 值。

例如，下列 IAM 政策允許主體執行指定的 KMS 金鑰資源操作，但僅能使用帳戶中的對稱加密 KMS 金鑰。

```
{
```

```
  "Effect": "Allow",
  "Action": [
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:ReEncrypt*",
    "kms:DescribeKey"
  ],
  "Resource": "arn:aws:kms:us-west-2:111122223333:key/*",
  "Condition": {
    "StringEquals": {
      "kms:KeySpec": "SYMMETRIC_DEFAULT"
    }
  }
}
```

另請參閱

- 公里：BypassPolicyLockoutSafetyCheck
- 公里：CustomerMasterKeySpec（已棄用）
- 公里：DataKeyPairSpec
- 公里：KeyOrigin
- 公里：KeyUsage

公里：KeyUsage

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:KeyUsage	字串	單一值	CreateKey KMS 金鑰資源操作	IAM 政策 金鑰政策和 IAM 政策

kms:KeyUsage 條件索引鍵會根據此操作建立或使用之 KMS 金鑰的 KeyUsage 屬性值，控制對操作的存取。

您可以使用此條件鍵，根據請求中的KeyUsage參數值來控制對CreateKey作業的存取。KeyUsage 的有效值為 ENCRYPT_DECRYPT、SIGN_VERIFY 和 GENERATE_VERIFY_MAC。

例如，您只能在 KeyUsage 為 ENCRYPT_DECRYPT 時建立 KMS 金鑰，或在 KeyUsage 為 SIGN_VERIFY 時拒絕使用者許可。

下列範例 IAM 政策陳述式會使用 kms:KeyUsage 條件索引鍵，僅在 KeyUsage 為 ENCRYPT_DECRYPT 時才建立 KMS 金鑰。

```
{
  "Effect": "Allow",
  "Action": "kms:CreateKey",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:KeyUsage": "ENCRYPT_DECRYPT"
    }
  }
}
```

您也可以使用 kms:KeyUsage 條件索引鍵，根據操作中 KMS 金鑰的 KeyUsage 屬性控制對使用或管理 KMS 金鑰之操作的存取。操作必須是 KMS 金鑰資源操作，也就是授權特定 KMS 金鑰的操作。若要識別 KMS 金鑰資源操作，請在動作與資源表中尋找操作之 Resources 資料欄的 KMS key 值。

例如，下列 IAM 政策允許主體執行指定的 KMS 金鑰資源操作，但僅能使用帳戶中用於簽署和驗證的 KMS 金鑰。

```
{
  "Effect": "Allow",
  "Action": [
    "kms:CreateGrant",
    "kms:DescribeKey",
    "kms:GetPublicKey",
    "kms:ScheduleKeyDeletion"
  ],
  "Resource": "arn:aws:kms:us-west-2:111122223333:key/*",
  "Condition": {
    "StringEquals": {
      "kms:KeyUsage": "SIGN_VERIFY"
    }
  }
}
```

另請參閱

- 公里：BypassPolicyLockoutSafetyCheck
- 公里：CustomerMasterKeyUsage（已棄用）
- 公里：KeyOrigin
- 公里：KeySpec

公里：MacAlgorithm

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:MacAlgorithm	字串	單一值	GenerateMac VerifyMac	金鑰政策和 IAM 政策

您可以使用kms:MacAlgorithm條件索引鍵，根據請求中的MacAlgorithm參數值來控制對GenerateMac和VerifyMac作業的存取。

以下範例金鑰政策允許可擔任 testers 角色的使用者，僅可在請求中的 MAC 演算法為 HMAC_SHA_384 或 HMAC_SHA_512 時，使用 HMAC KMS 金鑰來產生及驗證 HMAC 標籤。此政策會使用兩個不同的政策陳述式，每個陳述式都有各自的條件。若您在單一條件陳述式中指定多個 MAC 演算法，則該條件會同時需要兩個演算法，而非僅任一個演算法。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/testers"
      },
      "Action": [
        "kms:GenerateMac",
        "kms:VerifyMac"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:MacAlgorithm": "HMAC_SHA_384"
        }
```

```
      }
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::111122223333:role/testers"
      },
      "Action": [
        "kms:GenerateMac",
        "kms:VerifyMac"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:MacAlgorithm": "HMAC_SHA_512"
        }
      }
    }
  ]
}
```

另請參閱

- the section called “公里：EncryptionAlgorithm”
- 公里：SigningAlgorithm

公里：MessageType

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:MessageType	字串	單一值	Sign Verify	金鑰政策和 IAM 政策

根據請求中的 MessageType 參數值，kms:MessageType 條件索引鍵會控制對 Sign 和 Verify 操作的存取。MessageType 的有效值為 RAW 和 DIGEST。

例如，下列金鑰政策陳述式使用 kms:MessageType 條件索引鍵，以使用非對稱 KMS 金鑰簽署訊息，而非訊息摘要。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
  },
  "Action": "kms:Sign",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:MessageType": "RAW"
    }
  }
}
```

另請參閱

- the section called “公里：SigningAlgorithm”

公里：MultiRegion

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:MultiRegion	Boolean	單一值	CreateKey KMS 金鑰資源操作	金鑰政策和 IAM 政策

您可以使用此條件索引鍵，僅允許對單一區域金鑰的操作，或僅允許對多區域金鑰的操作。kms:MultiRegion條件金鑰可控制 KMS 金鑰CreateKey作 AWS KMS 業的存取，以及根據 KMS 金鑰的MultiRegion屬性值存取作業。有效值為 true (多區域) 和 false (單一區域)。所有 KMS 金鑰都有 MultiRegion 屬性。

例如，下列 IAM 政策陳述式會使用 kms:MultiRegion 條件索引鍵，允許主體僅建立單一區域金鑰。

```
{
  "Effect": "Allow",
  "Action": "kms:CreateKey",
  "Resource": "*",
```

```
  "Condition": {
    "Bool": {
      "kms:MultiRegion": false
    }
  }
}
```

公里：MultiRegionKeyType

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:MultiRegionKeyType	字串	單一值	CreateKey KMS 金鑰資源操作	金鑰政策和 IAM 政策

您可以使用此條件索引鍵，僅允許對多區域主要金鑰的操作，或僅允許對多區域複本金鑰的操作。kms:MultiRegionKeyType條件金鑰控制對 KMS 金鑰CreateKey作 AWS KMS 業的存取，以及根據 KMS 金鑰的MultiRegionKeyType屬性控制作業的存取。有效值為 PRIMARY 和 REPLICA。只有多區域金鑰具有 MultiRegionKeyType 屬性。

一般而言，您需要使用 IAM 政策中的 kms:MultiRegionKeyType 條件索引鍵，以控制對多個 KMS 金鑰的存取。不過，由於指定的多區域金鑰可以變更為主要或複本金鑰，因此您可能想要在金鑰政策中使用此條件，僅當特定的多區域金鑰是主要金鑰或複本金鑰時才允許操作。

例如，以下 IAM 政策陳述式使用 kms:MultiRegionKeyType 條件索引鍵，以允許主體僅對指定 AWS 帳戶中的多區域複本金鑰排程和取消金鑰刪除。

```
{
  "Effect": "Allow",
  "Action": [
    "kms:ScheduleKeyDeletion",
    "kms:CancelKeyDeletion"
  ],
  "Resource": "arn:aws:kms:*:111122223333:key/*",
  "Condition": {
    "StringEquals": {
      "kms:MultiRegionKeyType": "REPLICA"
    }
```

```
  }
}
```

若要允許或拒絕存取所有多區域金鑰，您可以將兩個值或 Null 值與 kms:MultiRegionKeyType 搭配使用。不過，建議您使用 kms: MultiRegion 條件金鑰來達到此目的。

公里：PrimaryRegion

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:PrimaryRegion	字串 (清單)	單一值	UpdatePrimaryRegion	金鑰政策和 IAM 政策

您可以使用此條件鍵來限制UpdatePrimaryRegion作業中的目的地區域。這些是可 AWS 區域 以託管您的多區域主鍵。

kms:PrimaryRegion條件鍵會根據PrimaryRegion參數值控制對UpdatePrimaryRegion作業的存取。此PrimaryRegion參數指定要提升為主要 AWS 區域 的多區域複本金鑰。條件的值為一或多個 AWS 區域 名稱，例如us-east-1或ap-southeast-2，或區域名稱模式，例如 eu-*

例如，以下金鑰政策陳述式使用 kms:PrimaryRegion 條件索引鍵，以允許主體將多重區域金鑰的主要區域更新為四個指定區域之一。

```
{
  "Effect": "Allow",
  "Action": "kms:UpdatePrimaryRegion",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/Developer"
  },
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:PrimaryRegion": [
         "us-east-1",
         "us-west-2",
         "eu-west-3",
         "ap-southeast-2"
      ]
    }
```

```
  }
}
```

公里：ReEncryptOnSameKey

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:ReEncryptOnSameKey	Boolean	單一值	ReEncrypt	金鑰政策和 IAM 政策

您可以根據要求是否指定與原始加密相同的目的地 KMS 金鑰，使用此條件金鑰來控制對ReEncrypt作業的存取。

例如，下列金鑰政策陳述式使用 kms:ReEncryptOnSameKey 條件索引鍵，只在目的地 CMK 與原始加密所用的 KMS 金鑰相同時，才可重新加密。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
  },
  "Action": "kms:ReEncrypt*",
  "Resource": "*",
  "Condition": {
    "Bool": {
      "kms:ReEncryptOnSameKey": true
    }
  }
}
```

公里：RequestAlias

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:RequestAlias	字串 (清單)	單一值	密碼編譯操作	金鑰政策和 IAM 政策

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
			DescribeKey GetPublicKey	

您可以使用此條件索引鍵，僅當請求使用特定別名來識別 KMS 金鑰時，才允許操作。kms:RequestAlias 條件索引鍵可根據在請求中識別該 KMS 金鑰的別名，控制對密碼編譯操作、GetPublicKey 或 DescribeKey 中所使用 KMS 金鑰的存取。(此原則條件對作業沒有影響，因為GenerateRandom作業不使用 KMS 金鑰或別名。)

此條件支援中的屬性型存取控制 (ABAC) AWS KMS，可讓您根據 KMS 金鑰的標籤和別名控制 KMS 金鑰的存取。您可以使用標籤和別名，來允許或拒絕對 KMS 金鑰的存取，而無需變更政策或授予。如需詳細資訊，請參閱 AWS KMS 的 ABAC。

若要在此政策條件中指定別名，請使用別名名稱，例如 alias/project-alpha，或別名名稱模式，例如 alias/*test*。您不能指定此條件索引鍵值中的別名 ARN。

若要滿足此條件，請求中的 KeyId 參數值必須為相符的別名名稱或別名 ARN。如果請求使用不同的金鑰識別符，則其不滿足條件，即使識別為相同的 KMS 金鑰。

例如，下列金鑰原則陳述式允許主體呼叫 KMS 金鑰上的GenerateDataKey作業。但是，僅當請求中的 KeyId 參數值為 alias/finance-key 或具有該別名名稱的別名 ARN，例如 arn:aws:kms:us-west-2:111122223333:alias/finance-key。

```
{
  "Sid": "Key policy using a request alias condition",
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/developer"
  },
  "Action": "kms:GenerateDataKey",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:RequestAlias": "alias/finance-key"
    }
  }
}
```

您無法使用此條件鍵來控制對別名作業的存取，例如CreateAlias或DeleteAlias。如需控制對別名操作之存取的詳細資訊，請參閱 控制對別名的存取。

公里：ResourceAliases

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:ResourceAliases	字串 (清單)	多重值	KMS 金鑰資源操作	僅限 IAM 政策

使用此條件索引鍵，根據與 KMS 金鑰相關聯的別名，控制對 KMS 金鑰的存取。操作必須是 KMS 金鑰資源操作，也就是授權特定 KMS 金鑰的操作。若要識別 KMS 金鑰資源操作，請在動作與資源表中尋找操作之 Resources 資料欄的 KMS key 值。

此條件支援 AWS KMS中的屬性型存取控制 (ABAC)。透過 ABAC，您可以根據指派給 KMS 金鑰的標籤和與 KMS 金鑰相關聯的別名，來控制對 KMS 金鑰的存取。您可以使用標籤和別名，來允許或拒絕對 KMS 金鑰的存取，而無需變更政策或授予。如需詳細資訊，請參閱 AWS KMS 的 ABAC。

別名在 AWS 帳戶 和區域中必須是唯一的，但是此條件可讓您控制對相同區域中多個 KMS 金鑰的存取權 (使用StringLike比較運算子)，或控制每個帳戶不同的多個 KMS 金鑰 AWS 區域 的存取權。

> **Note**
>
> KMS：只有當 KMS 金鑰符合每個 KMS 金鑰配額的別名時，ResourceAliases條件才有效。如果 KMS 金鑰超過此配額，則會拒絕透過 kms:ResourceAliases 條件授權使用 KMS 金鑰的主體存取 KMS 金鑰。

若要在此政策條件中指定別名，請使用別名名稱，例如 alias/project-alpha，或別名名稱模式，例如 alias/*test*。您不能指定此條件索引鍵值中的別名 ARN。若要滿足條件，操作中使用的 KMS 金鑰必須具有指定的別名。在操作要求中是否識別或如何識別 KMS 金鑰並不重要。

這是多重值條件索引鍵，會將與 KMS 金鑰相關聯的一組別名與政策中的一組別名進行比較。若要判斷這些集合的比較方式，您必須提供政策條件中的 ForAnyValue 或 ForAllValues 集合運算子。如需集合運算子的詳細資訊，請參閱《IAM 使用者指南》中的使用多個索引鍵和值。

- ForAnyValue：至少有一個與 KMS 金鑰相關聯的別名必須符合原則條件中的別名。允許使用其他別名。如果 KMS 金鑰沒有別名，則不符合條件。

- ForAllValues：與 KMS 金鑰關聯的每個別名都必須符合原則中的別名。此集合運算子會將與 KMS 金鑰相關聯的別名限制為政策條件中的別名。它不需要任何別名，但會禁止未指定的別名。

例如，下列 IAM 政策陳述式允許主體呼叫與finance-key別名相關聯之指定 AWS 帳戶 之任何 KMS 金鑰上的GenerateDataKey作業。(受影響的 KMS 金鑰的金鑰政策也必須允許主體的帳戶使用這些金鑰來進行此操作。) 若要在可能與 KMS 金鑰相關聯的許多別名之一為 alias/finance-key 時滿足該條件，條件會使用 ForAnyValue 集合運算子。

由於 kms:ResourceAliases 條件是根據資源，而非請求，對於任何與 finance-key 別名相關聯的 KMS 金鑰的 GenerateDataKey 呼叫會成功，即使請求使用金鑰 ID 或金鑰 ARN 來識別 KMS 金鑰。

```
{
  "Sid": "AliasBasedIAMPolicy",
  "Effect": "Allow",
  "Action": "kms:GenerateDataKey",
  "Resource": [
    "arn:aws:kms:*:111122223333:key/*",
    "arn:aws:kms:*:444455556666:key/*"
  ],
  "Condition": {
    "ForAnyValue:StringEquals": {
      "kms:ResourceAliases": "alias/finance-key"
    }
  }
}
```

下列範例 IAM 政策陳述式允許主體啟用和停用 KMS 金鑰，但僅當 KMS 金鑰的所有別名都包含 "Test" 時。此政策陳述式使用兩個條件。具有 ForAllValues 集合運算子的條件需要所有與 KMS 金鑰相關聯的別名都包含「測試」。具有 ForAnyValue 集合運算子的條件需要 KMS 金鑰至少有一個具有「測試」的別名。沒有 ForAnyValue 條件時，此政策陳述式會允許主體使用沒有別名的 KMS 金鑰。

```
{
  "Sid": "AliasBasedIAMPolicy",
  "Effect": "Allow",
  "Action": [
    "kms:EnableKey",
    "kms:DisableKey"
  ],
```

```
  "Resource": "arn:aws:kms:*:111122223333:key/*",
  "Condition": {
    "ForAllValues:StringLike": {
      "kms:ResourceAliases": [
        "alias/*Test*"
      ]
    },
    "ForAnyValue:StringLike": {
      "kms:ResourceAliases": [
        "alias/*Test*"
      ]
    }
  }
}
```

公里：ReplicaRegion

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:ReplicaRegion	字串 (清單)	單一值	ReplicateKey	金鑰政策和 IAM 政策

您可以使用此條件索引鍵來限制主體可複寫多區域金鑰的 AWS 區域 內容。kms:ReplicaRegion條件索引鍵會根據要求中的ReplicaRegion參數值控制對ReplicateKey作業的存取。此參數指定為新複本金鑰指定 AWS 區域 。

條件的值是一或多個 AWS 區域 名稱，例如us-east-1或或ap-southeast-2名稱模式，例如eu-*。如需 AWS KMS 支援 AWS 區域 的名稱清單 AWS Key Management Service ，請參閱 AWS 一般參考.

例如，下列索引鍵原則陳述式會使用kms:ReplicaRegion條件索引鍵，只有當ReplicaRegion參數值是其中一個指定的 Region 時，才允許主體呼叫ReplicateKey作業。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/Administrator"
  },
  "Action": "kms:ReplicateKey"
```

```
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:ReplicaRegion": [
         "us-east-1",
         "eu-west-3",
         "ap-southeast-2"
      ]
    }
  }
}
```

此條件鍵僅控制對ReplicateKey作業的存取。若要控制對UpdatePrimaryRegion作業的存取，請使用 kms: PrimaryRegion 條件金鑰。

公里：RetiringPrincipal

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:RetiringPrincipal	字串 (清單)	單一值	CreateGrant	金鑰政策和 IAM 政策

您可以使用此條件鍵，根據請求中的RetiringPrincipal參數值來控制對CreateGrant作業的存取。例如，您可以只在 CreateGrant 請求中的 RetiringPrincipal 符合條件陳述式中的 RetiringPrincipal 時，才可建立授予來使用 KMS 金鑰。

若要指定淘汰主體，請使用主體的 Amazon 資源名稱 (ARN)。 AWS 有效的主體包括 IAM 使用者 AWS 帳戶、IAM 角色、聯合身分使用者和假定角色使用者。如需主體 ARN 語法的相關說明，請參閱 IAM 使用者指南中的 IAM ARN。

下列金鑰原則陳述式範例可讓使用者建立 KMS 金鑰的授權。kms:RetiringPrincipal條件索引鍵會將權限限制在授權中已淘汰的主體為的CreateGrant要求。LimitedAdminRole

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
```

```
  },
  "Action": "kms:CreateGrant",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:RetiringPrincipal": "arn:aws:iam::111122223333:role/LimitedAdminRole"
    }
  }
}
```

另請參閱

- 公里：GrantConstraintType
- 公里：GrantIsForAWSResource
- 公里：GrantOperations
- 公里：GranteePrincipal

公里：ScheduleKeyDeletionPendingWindowInDays

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:ScheduleKeyDeletionPendingWindowInDays	數值	單一值	ScheduleKeyDeletion	金鑰政策和 IAM 政策

您可以使用此條件鍵來限制主參數可在ScheduleKeyDeletion請求PendingWindowInDays參數中指定的值。

PendingWindowInDays指定刪除金鑰之前 AWS KMS 要等待的天數。 AWS KMS 可讓您指定介於 7 到 30 天之間的等待期間，但是您可以使用kms:ScheduleKeyDeletionPendingWindowInDays條件鍵來進一步限制等候期間，例如強制執行有效範圍內的最短等待期。

例如，下列金鑰政策陳述式採用 kms:ScheduleKeyDeletionPendingWindowInDays 條件金鑰，以便防止主體在等待期間小於或等於 21 天時，排程刪除金鑰。

```
{
  "Effect": "Deny",
  "Action": "kms:ScheduleKeyDeletion",
  "Principal": "*",
  "Resource": "*",
  "Condition" : {
      "NumericLessThanEquals" : {
        "kms:ScheduleKeyDeletionPendingWindowInDays" : "21"
      }
  }
}
```

公里：SigningAlgorithm

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:SigningAlgorithm	字串	單一值	Sign Verify	金鑰政策和 IAM 政策

您可以使用kms:SigningAlgorithm條件索引鍵，根據請求中的SigningAlgorithm參數值來控制對 Sign 和 V erify 作業的存取。此條件金鑰不會影響以外執行的作業 AWS KMS，例如使用非對稱 KMS 金鑰組中的公開金 key pair 驗證簽章 AWS KMS。

下列範例金鑰政策允許可以擔任 testers 角色的使用者只有在請求所用的簽署演算法是 RSASSA_PSS 演算法 (例如 RSASSA_PSS_SHA512) 時，才能使用 KMS 金鑰簽署訊息。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/testers"
  },
  "Action": "kms:Sign",
  "Resource": "*",
  "Condition": {
    "StringLike": {
      "kms:SigningAlgorithm": "RSASSA_PSS*"
    }
  }
```

```
}
```

另請參閱

- 公里：EncryptionAlgorithm
- the section called “公里：MacAlgorithm”
- the section called “公里：MessageType”

公里：ValidTo

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:ValidTo	時間戳記	單一值	ImportKeyMaterial	金鑰政策和 IAM 政策

kms:ValidTo條件索引鍵會根據請求中的ValidTo參數值來控制ImportKeyMaterial作業的存取，這個值決定匯入的金鑰材料何時到期。這個值是以 Unix 時間表示。

在預設情況下，ImportKeyMaterial 請求需要 ValidTo 參數。但是，如果ExpirationModel參數的值為KEY_MATERIAL_DOES_NOT_EXPIRE，則ValidTo參數無效。您也可以使用 kms:ExpirationModel 條件索引鍵來要求ExpirationModel參數或特定參數值。

以下範例政策陳述式允許使用者將金鑰材料匯入 KMS 金鑰。kms:ValidTo 條件金鑰限制 ImportKeyMaterial 請求的許可，其中 ValidTo 值小於或等於 1546257599.0 (2018 年 12 月 31 日下午 11:59:59)。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
  },
  "Action": "kms:ImportKeyMaterial",
  "Resource": "*",
  "Condition": {
    "NumericLessThanEquals": {
      "kms:ValidTo": "1546257599.0"
    }
```

```
  }
}
```

另請參閱

- 公里：ExpirationModel
- 公里：WrappingAlgorithm
- 公里：WrappingKeySpec

公里：ViaService

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:ViaService	字串	單一值	KMS 金鑰資源操作	金鑰政策和 IAM 政策

kms:ViaService條件金鑰會限制使用 KMS 金鑰來自指定 AWS 服務的要求。您可以在每個 kms:ViaService 條件金鑰中指定一或多個服務。操作必須是 KMS 金鑰資源操作，也就是授權特定 KMS 金鑰的操作。若要識別 KMS 金鑰資源操作，請在動作與資源表中尋找操作之 Resources 資料欄的 KMS key 值。

例如，以下金鑰政策陳述式使用 kms:ViaService 條件索引鍵，僅當代表 ExampleRole 的請求來自美國西部 (奧勒岡) 區域的 Amazon EC2 或 Amazon RDS 時，才會允許客戶受管金鑰用於指定動作。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
  },
  "Action": [
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*",
    "kms:CreateGrant",
    "kms:ListGrants",
```

```
    "kms:DescribeKey"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:ViaService": [
        "ec2.us-west-2.amazonaws.com",
        "rds.us-west-2.amazonaws.com"
      ]
    }
  }
}
```

您也可以使用 kms:ViaService 條件索引鍵，在請求來自特定的服務時，拒絕使用 KMS 金鑰的許可。例如，以下來自金鑰政策的陳述式使用 kms:ViaService 條件索引鍵，當代表 ExampleRole 的請求來自 AWS Lambda 時，防止將客戶受管金鑰用於 Encrypt 操作。

```
{
  "Effect": "Deny",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
  },
  "Action": [
    "kms:Encrypt"
  ],
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:ViaService": [
          "lambda.us-west-2.amazonaws.com"
      ]
    }
  }
}
```

Important

當您使用 kms:ViaService 條件索引鍵時，該服務會代表 AWS 帳戶中的主體提出請求。這些主體必須擁有以下許可：

- 使用 KMS 金鑰的許可。主體需要授予這些許可給整合服務，以便該服務可以代表主體使用客戶受管金鑰。如需詳細資訊，請參閱 AWS 服務使用 AWS KMS 的方式。

- 使用整合服務的許可。如需有關提供使用者存取與整合之 AWS 服務的詳細資訊 AWS KMS，請參閱整合式服務的說明文件。

所有 AWS 受管金鑰 在其金鑰政策文件中使用 kms:ViaService 條件索引鍵。只在請求是來自建立 KMS 金鑰的服務時，這個條件才僅允許將 KMS 金鑰用於這些請求。若要查看的金鑰原則 AWS 受管金鑰，請使用GetKeyPolicy作業。

kms:ViaService 條件金鑰在 IAM 和金鑰政策陳述式中有效。您指定的服務必須與 AWS KMS整合，並支援 kms:ViaService 條件金鑰。

支援 **kms:ViaService** 條件金鑰的服務

下表列出整合 AWS KMS 並支援在客戶管理金鑰中使用kms:ViaService條件金鑰的 AWS 服務。此表格中的服務可能無法在所有地區提供。在所有 AWS 分區中使用 AWS KMS ViaService 名稱的.amazonaws.com後綴。

> **Note**
>
> 您可能需要水平或垂直捲動，才能查看此資料表中的所有資料。

服務名稱	AWS KMS ViaService 名稱
AWS App Runner	apprunner. *AWS_region* .amazonaws.com
AWS AppFabric	appfabric. *AWS_region* .amazonaws.com
Amazon AppFlow	appflow.*AWS_region* .amazonaws.com
AWS Application Migration Service	mgn.*AWS_region* .amazonaws.com
Amazon Athena	athena.*AWS_region* .amazonaws.com
AWS Audit Manager	auditmanager. *AWS_region* .amazonaws.com

服務名稱	AWS KMS ViaService 名稱
Amazon Aurora	rds.*AWS_region* .amazonaws.com
AWS Backup	backup.*AWS_region* .amazonaws.com
AWS Backup 閘道	backup-gateway. *AWS_region* .amazonaws.com
Amazon Chime SDK	chimevoiceconnector. *AWS_region* .amazonaws.com
AWS CodeArtifact	codeartifact. *AWS_region* .amazonaws.com
Amazon 評論 CodeGuru 家	codeguru-reviewer. *AWS_region* .amazonaws.com
Amazon Comprehend	comprehend. *AWS_region* .amazonaws.com
Amazon Connect	connect.*AWS_region* .amazonaws.com
Amazon Connect Customer Profiles	profile.*AWS_region* .amazonaws.com
Amazon Q in Connect	wisdom.*AWS_region* .amazonaws.com
AWS Database Migration Service (AWS DMS)	dms.*AWS_region* .amazonaws.com
AWS Directory Service	directoryservice. *AWS_region* .amazonaws.com
Amazon DynamoDB	dynamodb. *AWS_region* .amazonaws.com
Amazon DocumentDB	docdb-elastic. *AWS_region* .amazonaws.com
Amazon EC2 Systems Manager (SSM)	ssm.*AWS_region* .amazonaws.com

服務名稱	AWS KMS ViaService 名稱
Amazon Elastic Block Store (Amazon EBS)	ec2.*AWS_region* .amazonaws.com (僅 EBS)
Amazon Elastic Container Registry (Amazon ECR)	ecr.*AWS_region* .amazonaws.com
Amazon Elastic File System (Amazon EFS)	elasticfilesystem. *AWS_region* .amazonaws.com
Amazon ElastiCache	在條件索引鍵值中包含兩個 ViaService 名稱： • elasticache. *AWS_region* .amazonaws.com • dax.*AWS_region* .amazonaws.com
AWS Elemental MediaTailor	mediatailor. *AWS_region* .amazonaws.com
AWS 實體解析度	entityresolution. *AWS_region* .amazonaws.com
Amazon FinSpace	finspace. *AWS_region* .amazonaws.com
Amazon Forecast	forecast. *AWS_region* .amazonaws.com
Amazon FSx	fsx.*AWS_region* .amazonaws.com
AWS Glue	glue.*AWS_region* .amazonaws.com
AWS Ground Station	groundstation. *AWS_region* .amazonaws.com
Amazon GuardDuty	malware-protection. *AWS_region* .amazonaws.com

服務名稱	AWS KMS ViaService 名稱
AWS HealthLake	healthlake. *AWS_region* .amazonaws.com
AWS IoT SiteWise	iotsitewise. *AWS_region* .amazonaws.com
Amazon Kendra	kendra.*AWS_region* .amazonaws.com
Amazon Keyspaces (適用於 Apache Cassandra)	cassandra. *AWS_region* .amazonaws.com
Amazon Kinesis	kinesis.*AWS_region* .amazonaws.com
Amazon 數據 Firehose	firehose. *AWS_region* .amazonaws.com
Amazon Kinesis Video Streams	kinesisvideo. *AWS_region* .amazonaws.com
AWS Lambda	lambda.*AWS_region* .amazonaws.com
Amazon Lex	lex.*AWS_region* .amazonaws.com
AWS License Manager	license-manager. *AWS_region* .amazonaws.com
Amazon Location Service	geo.*AWS_region* .amazonaws.com
Amazon Lookout for Equipment	lookoutequipment. *AWS_region* .amazonaws.com
Amazon Lookout for Metrics	lookoutmetrics. *AWS_region* .amazonaws.com
Amazon Lookout for Vision	lookoutvision. *AWS_region* .amazonaws.com
Amazon Macie	macie.*AWS_region* .amazonaws.com

服務名稱	AWS KMS ViaService 名稱
AWS Mainframe Modernization	m2.*AWS_region* .amazonaws.com
Amazon Managed Blockchain	managedblockchain. *AWS_region* .amazonaws.com
Amazon Managed Streaming for Apache Kafka (Amazon MSK)	kafka.*AWS_region* .amazonaws.com
Amazon Managed Workflows for Apache Airflow (MWAA)	airflow.*AWS_region* .amazonaws.com
Amazon MemoryDB for Redis	memorydb. *AWS_region* .amazonaws.com
Amazon Monitron	monitron. *AWS_region* .amazonaws.com
Amazon MQ	mq.*AWS_region* .amazonaws.com
Amazon Neptune	rds.*AWS_region* .amazonaws.com
Amazon Nimble Studio	nimble.*AWS_region* .amazonaws.com
AWS HealthOmics	omics.*AWS_region* .amazonaws.com
Amazon OpenSearch 服務	es.*AWS_region* .amazonaws.com ，aoss.*AWS_region* .amazonaws.com
AWS Proton	proton.*AWS_region* .amazonaws.com
Amazon Quantum Ledger Database (Amazon QLDB)	qldb.*AWS_region* .amazonaws.com
Amazon RDS Performance Insights	rds.*AWS_region* .amazonaws.com
Amazon Redshift	redshift. *AWS_region* .amazonaws.com

服務名稱	AWS KMS ViaService 名稱
Amazon Redshift 查詢編輯器第 2 版	sqlworkbench. *AWS_region* .amazonaws.com
Amazon Redshift Serverless	redshift-serverless. *AWS_region* .amazonaws.com
Amazon Rekognition	rekognition. *AWS_region* .amazonaws.com
Amazon Relational Database Service (Amazon RDS)	rds.*AWS_region* .amazonaws.com
Amazon Replicated Data Store	ards.*AWS_region* .amazonaws.com
Amazon SageMaker	sagemaker. *AWS_region* .amazonaws.com
AWS Secrets Manager	secretsmanager. *AWS_region* .amazonaws.com
Amazon Security Lake	securitylake. *AWS_region* .amazonaws.com
Amazon Simple Email Service (Amazon SES)	ses.*AWS_region* .amazonaws.com
Amazon Simple Notification Service (Amazon SNS)	sns.*AWS_region* .amazonaws.com
Amazon Simple Queue Service (Amazon SQS)	sqs.*AWS_region* .amazonaws.com
Amazon Simple Storage Service (Amazon S3)	s3.*AWS_region* .amazonaws.com
AWS Snowball	importexport. *AWS_region* .amazonaws.com
AWS Storage Gateway	storagegateway. *AWS_region* .amazonaws.com

服務名稱	AWS KMS ViaService 名稱
AWS Systems Manager Incident Manager	ssm-incidents. *AWS_region* .amazonaws.com
AWS Systems Manager Incident Manager 联系人	ssm-contacts. *AWS_region* .amazonaws.com
Amazon Timestream	timestream. *AWS_region* .amazonaws.com
Amazon Translate	translate. *AWS_region* .amazonaws.com
AWS Verified Access	verified-access. *AWS_region* .amazonaws.com
Amazon WorkMail	workmail. *AWS_region* .amazonaws.com
Amazon WorkSpaces	workspaces. *AWS_region* .amazonaws.com
Amazon WorkSpaces 瘦客戶端	thinclient. *AWS_region* .amazonaws.com
Amazon WorkSpaces 網站	workspaces-web. *AWS_region* .amazonaws.com
AWS X-Ray	xray.*AWS_region* .amazonaws.com

公里：WrappingAlgorithm

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:WrappingAlgorithm	字串	單一值	GetParametersForImport	金鑰政策和 IAM 政策

此條件索引鍵會根據要求中的WrappingAlgorithm參數值控制對GetParametersForImport作業的存取。您可以使用此條件，要求主體在匯入過程中使用特定的演算法來加密金鑰材料。在指定不同的包裝演算法時，要求所需的公有金鑰和匯入符記會失敗。

下列範例金鑰政策陳述式會使用 kms:WrappingAlgorithm 條件索引鍵來提供範例使用者呼叫 GetParametersForImport 操作的許可，但會防止他們使用 RSAES_OAEP_SHA_1 包裝演算法。GetParametersForImport 請求中的 WrappingAlgorithm 是 RSAES_OAEP_SHA_1 時，操作會失敗。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
  },
  "Action": "kms:GetParametersForImport",
  "Resource": "*",
  "Condition": {
    "StringNotEquals": {
      "kms:WrappingAlgorithm": "RSAES_OAEP_SHA_1"
    }
  }
}
```

另請參閱

- 公里：ExpirationModel
- 公里：ValidTo
- 公里：WrappingKeySpec

公里：WrappingKeySpec

AWS KMS 條件鍵	條件類型	值類型	API 操作	Policy type (政策類型)
kms:WrappingKeySpec	字串	單一值	GetParametersForImport	金鑰政策和 IAM 政策

此條件索引鍵會根據要求中的WrappingKeySpec參數值控制對GetParametersForImport作業的存取。您可以使用此條件，要求主體在匯入過程中使用特定類型的公開金鑰。如果請求指定不同的金鑰類型，它會失敗。

由於 WrappingKeySpec 參數值的唯一有效值是 RSA_2048，防止使用者使用此值可以有效避免他們使用 GetParametersForImport 操作。

以下範例政策陳述式使用 kms:WrappingAlgorithm 條件金鑰，要求請求中的 WrappingKeySpec 必須是 RSA_4096。

```
{
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/ExampleRole"
  },
  "Action": "kms:GetParametersForImport",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:WrappingKeySpec": "RSA_4096"
    }
  }
}
```

另請參閱

- 公里：ExpirationModel
- 公里：ValidTo
- 公里：WrappingAlgorithm

AWS KMSAWS 硝基飛地的條件鍵

AWS Nitro Enclaves 是一種 Amazon EC2 功能，可讓您建立稱為隔離區的獨立運算環境，以保護和處理高度敏感的資料。 AWS KMS 提供條件鍵以支持 AWS 硝基飛地。這些條件金鑰僅適用於對硝基飛 AWS KMS 地的要求。

當您使用來自 Enclave 的已簽署驗證文件呼叫解密GenerateDataKeyPair、、或 GenerateRandomAPI 作業時，這些 API 會在驗證文件中公開金鑰下的回應中加密純文字，並傳回密文而非純文字。GenerateDataKey僅當使用 enclave 的私有金鑰時才能解密此密文。如需詳細資訊，請參閱 AWS Nitro Enclaves 如何使用 AWS KMS。

下列條件鍵可讓您根據已簽署的證明文件內容，限制這些操作的許可。允許作業之前，請先 AWS KMS 將 Enclave 中的驗證文件與這些條件索引鍵中的值進行比較。 AWS KMS

公里RecipientAttestationImageSha:

AWS KMS 條件鍵	條件類型	值類型	API 操作	政策類型

`kms:RecipientAttestation:ImageSha384`	字串	單一值	`Decrypt`
			`GenerateDataKey`
			`GenerateDataKeyPair`
			`GenerateRandom`
				金鑰政策和 IAM 政策

當請求中已簽署證明文件的影像摘要符合條件金鑰的值時，`kms:RecipientAttestation:ImageSha384` 條件金鑰可利用 KMS 金鑰控制對以下各項的存取：`Decrypt`、`GenerateDataKey`、`GenerateDataKeyPair` 與 `GenerateRandom`。`ImageSha384` 值對應證明文件的 PCR0。只有當請求中的`Recipient`參數指定了 AWS Nitro Enclave 的已簽署證明文件時，此條件金鑰才有效。

此值也包含在要求 Nitro 飛地 AWS KMS 的CloudTrail事件中。

> **Note**
>
> 此條件金鑰在金鑰政策陳述式和 IAM 政策陳述式中有效，即使它未出現在 IAM 主控台或 IAM 服務授權參考。

例如，下列金鑰原則陳述式允許`data-processing`角色使用 KMS 金鑰進行解密GenerateDataKey、GenerateDataKeyPair、和GenerateRandom作業。僅當請求中證明文件的影像摘要值 (PCR0) 符合條件的影像摘要值時，`kms:RecipientAttestation:ImageSha384` 條件金鑰才會允許操作。只有當請求中的`Recipient`參數指定了 AWS Nitro Enclave 的已簽署證明文件時，此條件金鑰才有效。

如果請求不包含來自 AWS Nitro Enclave 的有效證明文件，則會拒絕權限，因為不滿足此條件。

```
{
  "Sid" : "Enable enclave data processing",
  "Effect" : "Allow",
  "Principal" : {
    "AWS" : "arn:aws:iam::111122223333:role/data-processing"
  },
  "Action": [
    "kms:Decrypt",
    "kms:GenerateDataKey",
    "kms:GenerateDataKeyPair",
    "kms:GenerateRandom"
  ],
  "Resource" : "*",
  "Condition": {
    "StringEqualsIgnoreCase": {
      "kms:RecipientAttestation:ImageSha384":
 "9fedcba8abcdef7abcdef6abcdef5abcdef4abcdef3abcdef2abcdef1abcdef1abcdef0abcdef1abcdef2abcdef3a
    }
  }
}
```

公里:: 聚氯乙烯 RecipientAttestation <PCR_ID>

AWS KMS 條件鍵	條件類型	值類型	API 操作	政策類型
kms:RecipientAttestation:PCR<PCR_ID>	字串	單一值	Decrypt GenerateDataKey GenerateDataKeyPair GenerateRandom	金鑰政策和 IAM 政策

僅當請求中已簽署證明文件的平台組態註冊 (PCR) 符合條件金鑰的 PCR 時，kms:RecipientAttestation:PCR<PCR_ID> 條件金鑰才會利用 KMS 金鑰控制對以下各項的存取：Decrypt、GenerateDataKey、GenerateDataKeyPair 以及 GenerateRandom。只有當請求中的Recipient參數指定來自 AWS Nitro 飛地的已簽署證明文件時，此條件金鑰才有效。

此值也包含在代表 AWS KMS 對 Nitro 飛地的請求的CloudTrail事件中。

> **Note**
>
> 此條件金鑰在金鑰政策陳述式和 IAM 政策陳述式中有效，即使它未出現在 IAM 主控台或 IAM 服務授權參考。

若要指定 PCR 值，請使用以下格式。將 PCR ID 串連到條件索引鍵名稱。PCR 值必須是最多 96 個位元組的小寫十六進位字串。

```
"kms:RecipientAttestation:PCRPCR_ID": "PCR_value"
```

例如，下列條件金鑰指定 PCR1 的特定值，對應於用於 enclave 與啟動程序處理的核心雜湊。

```
kms:RecipientAttestation:PCR1:
 "0x1abcdef2abcdef3abcdef4abcdef5abcdef6abcdef7abcdef8abcdef9abcdef8abcdef7abcdef6abcdef5abcdef
```

下列範例金鑰政策陳述式允許 data-processing 角色利用 KMS 金鑰進行 Decrypt 操作。

僅當請求中已簽署證明文件中的 PCR1 值符合條件中的 kms:RecipientAttestation:PCR1 值時，本陳述式中的 kms:RecipientAttestation:PCR 條件索引鍵才會允許操作。使用 StringEqualsIgnoreCase 政策運算子要求 PCR 值不區分大小寫的比較。

如請求不含證明文件，則會因未滿足此條件而拒絕許可。

```
{
  "Sid" : "Enable enclave data processing",
  "Effect" : "Allow",
  "Principal" : {
    "AWS" : "arn:aws:iam::111122223333:role/data-processing"
  },
  "Action": "kms:Decrypt",
  "Resource" : "*",
  "Condition": {
    "StringEqualsIgnoreCase": {
      "kms:RecipientAttestation:PCR1":
 "0x1de4f2dcf774f6e3b679f62e5f120065b2e408dcea327bd1c9dddaea6664e7af79355814748447674530 82c6f15
    }
  }
}
```

AWS KMS 的 ABAC

以屬性為基礎的存取控制 (ABAC) 是一種授權策略，可根據屬性來定義許可。AWS KMS 支援 ABAC，讓您可以根據與 KMS 金鑰相關聯的標籤和別名來控制對客戶受管金鑰的存取。在 AWS KMS 中啟用 ABAC 的標籤和別名條件索引鍵，提供強大且靈活的方式來授權委託人使用 KMS 金鑰，而無需編輯政策或管理授予。但是，您應該謹慎使用這些功能，避免委託人意外被允許或拒絕存取。

如果您使用 ABAC，請注意管理標籤和別名的許可現在是存取控制許可。部署依存於標籤或別名的政策之前，請務必先知曉所有 KMS 金鑰上的現有標籤和別名。新增、刪除和更新別名，以及標記和取消標記金鑰時，請採取合理的預防措施。只將管理標籤和別名的許可授予需要的委託人，並限制他們可以管理的標籤和別名。

> **備註**
>
> 當您將 ABAC 用於 AWS KMS 時，請謹慎授予委託人管理標籤和別名的許可。變更標籤或別名可能會允許或拒絕 KMS 金鑰的許可。沒有變更主要政策或建立授權之許可的重要管理員可以控制 KMS 金鑰的存取，如果他們擁有管理標籤或別名的許可。

可能最多需要五分鐘才能將標籤和別名變更體現在 KMS 金鑰授權上。最近的變更可能會在 API 操作中可見，然後才會影響授權。
若要根據 KMS 金鑰的別名來控制對 KMS 金鑰的存取，您必須使用條件索引鍵。您不能使用別名來代表政策陳述式 `Resource` 元素中的 KMS 金鑰。當別名出現在 `Resource` 元素時，政策陳述式會套用至別名，而不是相關聯的 KMS 金鑰。

進一步了解

- 如需 AWS KMS 支援 ABAC 的詳細資訊 (包括範例)，請參閱 使用別名來控制對 KMS 金鑰的存取 和 使用標籤來控制對 KMS 金鑰的存取。
- 如需使用標籤以控制對 AWS 資源之存取的詳細資訊，請參閱《IAM 使用者指南》中的什麼是適用於 AWS 的 ABAC？和使用資源標籤控制對 AWS 資源的存取。

適用於 AWS KMS 的 ABAC 條件索引鍵

若要根據 KMS 金鑰的標籤和別名授權存取，請在金鑰政策或 IAM 政策中使用下列條件索引鍵。

ABAC 條件索引鍵	描述	Policy type (政策類型)	AWS KMS 操作
AWS：ResourceTag	KMS 金鑰上的標籤 (鍵和值) 與政策中的標籤 (鍵和值) 或標籤模式相符	僅限 IAM 政策	KMS 金鑰資源操作 [2]
AWS：RequestTag/標籤鍵	請求中的標籤 (鍵和值) 與政策中的標籤 (鍵和值) 或標籤模式相符	金鑰政策和 IAM 政策 [1]	TagResource, UntagResource
AWS：TagKeys	請求中的標籤索引鍵與政策中的標籤索引鍵相符	金鑰政策和 IAM 政策 [1]	TagResource, UntagResource
公里：ResourceAliases	與 KMS 金鑰相關聯的別名符合政策中的別名或別名模式	僅限 IAM 政策	KMS 金鑰資源操作 [2]

ABAC 條件索引鍵	描述	Policy type (政策類型)	AWS KMS 操作
公里：RequestAlias	代表請求中 KMS 金鑰的別名與政策中的別名或別名模式相符。	金鑰政策和 IAM 政策[1]	密碼編譯作業、DescribeKeyGetPublicKey

[1] 可在金鑰政策中使用的任何條件索引鍵也可以在 IAM 政策中使用，但只能在金鑰政策允許的情形下。

[2]KMS 金鑰資源操作是針對特定 KMS 金鑰授權的操作。若要識別 KMS 金鑰資源操作，請在 AWS KMS 許可表中尋找 Resources 欄 KMS 金鑰的值，以執行操作。

例如，您可以使用這些條件索引鍵來建立下列政策。

- 具有 kms:ResourceAliases 的 IAM 政策，允許使用具有特定別名或別名模式的 KMS 金鑰的許可。這與依賴標記的政策有點不同：雖然您可以在政策中使用別名模式，但每個別名都必須為 AWS 帳戶 和區域中的專屬別名。這可讓您將政策套用至一組精選 KMS 金鑰，而不會在政策陳述式中列出 KMS 金鑰的金鑰 ARN。若要為集合新增 KMS 金鑰或從中移除 KMS 金鑰，請變更 KMS 金鑰的別名。
- 具有 kms:RequestAlias 的金鑰政策允許委託人在 Encrypt 操作中使用 KMS 金鑰，但僅在 Encrypt 請求會使用該別名來識別 KMS 金鑰的情形下。
- 具有 aws:ResourceTag/*tag-key* 的 IAM 政策拒絕將 KMS 金鑰與特定標籤索引鍵和標籤值使用的許可。這可讓您將政策套用至一組精選 KMS 金鑰，而不會在政策陳述式中列出 KMS 金鑰的金鑰 ARN。若要為集合新增 KMS 金鑰或從中移除 KMS 金鑰，請標記或取消標記 KMS 金鑰。
- 具有 aws:RequestTag/*tag-key* 的 IAM 政策允許委託人只刪除 "Purpose"="Test" KMS 金鑰標籤。
- 具有 aws:TagKeys 的 IAM 政策拒絕使用 Restricted 標籤索引鍵標記或取消標記 KMS 金鑰的許可。

ABAC 讓存取管理具有靈活性和可擴展性。例如，您可以使用 aws:ResourceTag/*tag-key* 條件索引鍵來建立 IAM 政策，此政策只允許委託人在 KMS 金鑰具有 Purpose=Test 標籤時針對特定操作使用 KMS 金鑰。此政策適用於 AWS 帳戶 所有區域中的所有 KMS 金鑰。

當連接至使用者或角色時，下列 IAM 政策允許委託人將所有現有的 KMS 金鑰與 Purpose=Test 標籤搭配用於指定的操作。若要將此存取權提供給新的或現有的 KMS 金鑰，則您不需要變更政策。只需將

Purpose=Test 標籤連接至 KMS 金鑰。同樣地，若要從具有 Purpose=Test 標籤的 KMS 金鑰中移除此存取權，請編輯或刪除標籤。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AliasBasedIAMPolicy",
      "Effect": "Allow",
      "Action": [
        "kms:Decrypt",
        "kms:Encrypt",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
      ],
      "Resource": "arn:aws:kms:*:111122223333:key/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Purpose": "Test"
        }
      }
    }
  ]
}
```

但是，如果您使用此功能，則請謹慎管理標籤和別名。新增、變更或刪除標籤或別名可能會意外允許或拒絕對 KMS 金鑰的存取。如果他們有管理標籤和別名的許可，則沒有變更主要政策或建立授予之許可的金鑰管理員可以控制對 KMS 金鑰的存取。為了減輕這種風險，請考慮限制管理標籤的許可和別名。例如，您可能只允許選取的委託人管理 Purpose=Test 標籤。如需詳細資訊，請參閱 使用別名來控制對 KMS 金鑰的存取 和 使用標籤來控制對 KMS 金鑰的存取。

標籤或別名？

AWS KMS 使用標籤和別名支援 ABAC。這兩種選項都提供靈活、可擴展的存取控制策略，但彼此略有不同。

您可能會決定使用標籤或根據您的特定 AWS 使用模式使用別名。例如，如果您已將標記許可指定給大多數管理員，則根據別名來控制授權策略可能會比較容易。或者，如果接近每個 KMS 金鑰的別名配額，則您可能會偏好基於標籤的授權策略。

以下是一般利益的優勢。

標籤型存取控制的優勢

- 對於不同類型 AWS 資源的相同授權機制。

  您可以使用相同的標籤或標籤金鑰來控制對多種資源類型的存取，例如 Amazon Relational Database Service (Amazon RDS) 叢集、Amazon Elastic Block Store (Amazon EBS) 磁碟區和 KMS 金鑰。此功能可啟用數種不同的授權模式，這些模式比傳統的角色型存取控制更靈活。

- 授權一組 KMS 金鑰的存取權。

  您可以使用標籤來管理對相同 AWS 帳戶 和區域中一組 KMS 金鑰的存取。將相同的標籤或標籤金鑰指派給您選擇的 KMS 金鑰。然後建立以標籤或標籤鍵為基礎的簡單 easy-to-maintain 原則陳述式。若要為授權群組新增 KMS 金鑰或從中移除金鑰，請新增或移除標籤；您無需編輯政策。

別名型存取控制的優勢

- 根據別名授權存取密碼編譯操作。

  屬性的大多數以請求為基礎的政策條件 (包括 aws:RequestTag/tag-key) 只會影響新增、編輯或刪除屬性的作業。但是 kms: RequestAlias 條件金鑰會根據用來識別要求中 KMS 金鑰的別名來控制密碼編譯作業的存取。例如，您可以授予委託人在 `Encrypt` 操作中使用 KMS 金鑰的許可，但僅在 `KeyId` 參數值為 `alias/restricted-key-1` 的情形下。若要滿足此條件，需要下列所有項目：

  - KMS 金鑰必須與該別名相關聯。
  - 請求必須使用別名來識別 KMS 金鑰。
  - 委託人必須擁有許可才能使用受限於 `kms:RequestAlias` 條件的 KMS 金鑰。

  如果您的應用程式通常使用別名名稱或別名 ARN 來引用 KMS 金鑰，則這會特別有用。

- 提供非常有限的許可。

  別名在 AWS 帳戶 和區域中必須是唯一的。因此，授予委託人根據別名存取 KMS 金鑰的許可，可能比為委託人授予標籤型存取權更嚴格。與別名不同的是，標籤可指派給相同帳戶和區域中的多個 KMS 金鑰。如果選擇，則您可以使用別名模式，例如 `alias/test*`，讓委託人可以存取相同帳戶和區域中的一組 KMS 金鑰。不過，允許或拒絕存取特定別名，實現了對 KMS 金鑰非常嚴格的控制。

對適用於 AWS KMS 的 ABAC 進行故障診斷

根據 KMS 金鑰的標籤和別名來控制對 KMS 金鑰的存取非常方便且功能強大。但是，很容易出現一些您想要防止的可預測錯誤。

存取因標籤變更而變更

如果刪除標籤或變更其值，則只能根據該標籤存取 KMS 金鑰的委託人將被拒絕存取 KMS 金鑰。當拒絕政策陳述式中包含的標籤新增至 KMS 金鑰時，也可能會發生這種情況。將政策相關標籤新增至 KMS 金鑰可以允許存取應被拒絕存取 KMS 金鑰的委託人。

例如，假設委託人可以根據 `Project=Alpha` 標籤存取 KMS 金鑰，例如下列範例 IAM 政策陳述式所提供的許可。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "IAMPolicyWithResourceTag",
      "Effect": "Allow",
      "Action": [
        "kms:GenerateDataKeyWithoutPlaintext",
        "kms:Decrypt"
      ],
      "Resource": "arn:aws:kms:ap-southeast-1:111122223333:key/*",
      "Condition": {
        "StringEquals": {
          "aws:ResourceTag/Project": "Alpha"
        }
      }
    }
  ]
}
```

如果從該 KMS 金鑰刪除標籤或標籤值變更，則委託人就不再具有使用 KMS 金鑰進行指定操作的許可。當主體嘗試在使用客戶管理金鑰的AWS服務中讀取或寫入資料時，這可能會變得很明顯。若要追蹤標籤變更，請檢閱 CloudTrail記錄TagResource或UntagResource 項目。

若要在不更新政策的情況下還原存取權，請變更 KMS 金鑰上的標籤。這個動作的影響最小，除了很短的一段時間，該動作會在整個 AWS KMS 中有效。為了防止類似錯誤，請僅將標記和取消標記許可提供給需要的委託人，並將其標記許可限制為他們需要管理的標籤。變更標籤之前，搜尋政策可偵測依

存於標籤的存取權，並在具有該標籤的所有區域中取得 KMS 金鑰。您可以考慮在更改特定標籤時創建 Amazon CloudWatch 警報。

存取因別名變更而變更

如果別名遭到刪除或與不同的 KMS 金鑰相關聯，則只能以該別名為基礎存取 KMS 金鑰的委託人將會被拒絕存取 KMS 金鑰。當與 KMS 金鑰相關聯的別名包含在拒絕政策陳述式中時，也可能會發生這種情況。將政策相關別名新增至 KMS 金鑰也可以允許存取應被拒絕存取 KMS 金鑰的委託人。

例如，下列 IAM 政策陳述式使用 kms: ResourceAliases 條件金鑰，允許存取帳戶不同區域中具有任何指定別名的 KMS 金鑰。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AliasBasedIAMPolicy",
      "Effect": "Allow",
      "Action": [
        "kms:List*",
        "kms:Describe*",
        "kms:Decrypt"
      ],
      "Resource": "arn:aws:kms:*:111122223333:key/*",
      "Condition": {
        "ForAnyValue:StringEquals": {
          "kms:ResourceAliases": [
            "alias/ProjectAlpha",
            "alias/ProjectAlpha_Test",
            "alias/ProjectAlpha_Dev"
          ]
        }
      }
    }
  ]
}
```

若要追蹤別名變更，請檢閱CreateAliasUpdateAlias、和DeleteAlias項目的 CloudTrail 記錄。

若要在不更新政策的情況下還原存取權，請變更與 KMS 金鑰相關聯的別名。由於每個別名只能與帳戶和區域中的一個 KMS 金鑰相關聯，因此管理別名會比管理標籤困難。還原某個 KMS 金鑰上的某些委託人的存取，可以拒絕相同或其他委託人存取不同的 KMS 金鑰。

若要避免發生此錯誤，請僅將別名管理許可提供給需要的委託人，並將其別名管理許可限制為需要管理的別名。在更新或刪除別名之前，搜尋政策以偵測取決於別名的存取權，並在與別名相關聯的所有區域中尋找 KMS 金鑰。

因別名配額而拒絕存取

如果 KMS 金鑰超過該帳戶和區域每個 KMS 金鑰配額的預設別名，AccessDenied則獲得授權可以按公里:ResourceAliases條件使用 KMS 金鑰的使用者將會得到例外狀況。

若要還原存取權，請刪除與 KMS 金鑰相關聯的別名，使其符合配額。或者使用替代機制，讓使用者存取 KMS 金鑰。

延遲的授權變更

您對標籤和別名所做的變更可能需要最長五分鐘才會體現在 KMS 金鑰授權上。因此，標籤或別名變更可能會反映在 API 操作的回應中，然後才會影響授權。這種延遲可能會比影響大多數 AWS KMS 操作的最終一致性短暫延遲要長。

例如，您可能擁有 IAM 政策，允許特定委託人將任何 KMS 金鑰與 "Purpose"="Test" 標籤搭配使用。然後，您將 "Purpose"="Test" 標籤新增至 KMS 金鑰。雖然TagResource作業完成且ListResourceTags回應會確認標籤已指派給 KMS 金鑰，但主體最多可能無法存取 KMS 金鑰五分鐘。

若要防止錯誤，請將此預期延遲建置到您的程式碼中。

因別名更新而失敗的請求

當更新別名時，您會將現有的別名關聯至不同的 KMS 金鑰。

解密和指定別名或別名 ARN 的ReEncrypt請求可能會失敗，因為別名現在與未加密密文字的 KMS 金鑰相關聯。這種情況通常會傳回 IncorrectKeyException 或 NotFoundException。或者，如果請求沒有 KeyId 或 DestinationKeyId 參數，則操作可能會失敗，並顯示 AccessDenied 例外狀況，因為呼叫者不再具有加密文字之 KMS 金鑰的存取權。

您可以透過查看CreateAlias、UpdateAlias和 CloudTrail 記錄項目的記DeleteAlias錄來追蹤變更。您也可以在ListAliases回應中使用LastUpdatedDate欄位的值來偵測變更。

例如，下列ListAliases範例回應顯示kms:ResourceAliases條件中的ProjectAlpha_Test別名已更新。因此，具有別名型存取權的委託人會失去對先前關聯 KMS 金鑰的存取權。相反地，他們可以存取新關聯的 KMS 金鑰。

```
$ aws kms list-aliases --query 'Aliases[?starts_with(AliasName, `alias/ProjectAlpha`)]'
```

```
{
    "Aliases": [
        {
            "AliasName": "alias/ProjectAlpha_Test",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/ProjectAlpha_Test",
            "TargetKeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
            "CreationDate": 1566518783.394,
            "LastUpdatedDate": 1605308931.903
        },
        {
            "AliasName": "alias/ProjectAlpha_Restricted",
            "AliasArn": "arn:aws:kms:us-west-2:111122223333:alias/
ProjectAlpha_Restricted",
            "TargetKeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
            "CreationDate": 1553410800.010,
            "LastUpdatedDate": 1553410800.010
        }
    ]
}
```

這項變更的補救措施並不簡單。您可以再次更新別名，將別名與原始 KMS 金鑰相關聯。不過，在採取行動之前，您需要考慮該變更對目前相關聯 KMS 金鑰的影響。如果委託人在密碼編譯操作中使用後一個 KMS 金鑰，則他們可能需要繼續存取該金鑰。在此情形下，您可能想要更新政策，以確保委託人擁有使用這兩個 KMS 金鑰的許可。

您可以避免這樣的錯誤：在更新別名之前，搜尋政策以偵測取決於別名的存取權。然後在與別名相關聯的所有區域中取得 KMS 金鑰。請僅將別名管理許可提供給需要的委託人，並將其別名管理許可限制為需要管理的別名。

允許其他帳戶中的使用者使用 KMS 金鑰

您可以允許不同 AWS 帳戶 中的使用者或角色使用帳戶中的 KMS 金鑰。跨帳戶存取需要 KMS 金鑰的金鑰政策和外部使用者帳戶之 IAM 政策中的許可。

跨帳戶許可僅適用於下列操作：

- 密碼編譯操作
- CreateGrant
- DescribeKey
- GetKeyRotationStatus

- GetPublicKey
- ListGrants
- RetireGrant
- RevokeGrant

如果您提供許可給不同帳戶中的使用者，讓他們能夠進行其他操作，則這些許可沒有任何作用。例如，如果您在不同帳戶中授予主體 kms： IAM 政策中的ListKeys權限或 kms: 金鑰政策中 KMS 金鑰的ScheduleKeyDeletion權限，則使用者嘗試呼叫您的資源上的這些作業仍然失敗。

如需有關使用不同帳戶中的 KMS 金鑰進行 AWS KMS 操作的詳細資訊，請參閱AWS KMS 權限和 在其他帳戶中使用 KMS 金鑰 中的 Cross-account use (跨帳戶使用) 資料欄。還有 AWS Key Management Service API 參考中每個 API 描述的跨帳戶使用章節。

> **Warning**
>
> 請謹慎為委託人提供使用 KMS 金鑰的許可。只要有可能，請遵循最低權限原則。讓使用者只能存取其所需的 KMS 金鑰，以便只能存取所需的操作。
> 此外，請謹慎使用任何不熟悉的 KMS 金鑰，尤其是不同帳戶中的 KMS 金鑰。惡意使用者可能會授予您使用其 KMS 金鑰的許可，以取得關於您或您帳戶的資訊。
> 如需使用政策來保護帳戶中資源的詳細資訊，請參閱 IAM 政策的最佳實務。

若要提供 KMS 金鑰的使用許可給另一個帳戶中的使用者和角色，則需使用兩種不同類型的政策：

- KMS 金鑰的金鑰政策必須提供外部帳戶 (或外部帳戶中的使用者和角色) 使用 KMS 金鑰的許可。金鑰政策位在擁有 KMS 金鑰的帳戶中。
- 外部帳戶的 IAM 政策必須委派金鑰政策許可給其使用者和角色。這些政策會在外部帳戶中設定，並提供許可給該帳戶中的使用者和角色。

金鑰政策決定誰可以存取 KMS 金鑰。IAM 政策決定誰可以存取 KMS 金鑰。單獨的金鑰政策或 IAM 政策都不夠，您必須同時變更兩者。

若要編輯金鑰原則，您可以使用中的原則檢視，AWS Management Console或使用CreateKey或PutKeyPolicy作業。如需建立 KMS 金鑰時設定金鑰政策的說明，請參閱 建立其他帳戶可以使用的 KMS 金鑰。

如需編輯 IAM 政策的說明，請參閱 將 IAM 政策與 AWS KMS 搭配使用。

如需說明金鑰政策和 IAM 政策如何搭配運作，以允許在不同帳戶中使用 KMS 金鑰的範例，請參閱 範例 2：使用者採用的角色具有在不同 AWS 帳戶 中使用 KMS 金鑰的許可。

您可以檢視 AWS CloudTrail 日誌中對 KMS 金鑰產生的跨帳戶 AWS KMS 操作。對其他帳戶中使用 KMS 金鑰的操作會同時記入呼叫者帳戶和 KMS 金鑰擁有者帳戶。

主題

> **Note**
>
> 本主題中的範例旨在說明如何結合使用金鑰政策和 IAM 政策，來提供和限制存取 KMS 金鑰的權限。這些一般範例並非是指任何特定 AWS 服務 在 KMS 金鑰上所需的許可。如需有關 AWS 服務 所需許可的詳細資訊，請參閱服務文件中的加密主題。

步驟 1：在本機帳戶中新增金鑰政策陳述式

KMS 金鑰的金鑰政策是決定誰可以存取 KMS 金鑰及其可執行操作的主要因素。金鑰政策一律位在擁有 KMS 金鑰的帳戶中。與 IAM 政策不同，金鑰政策並不會指定資源。資源即是與該金鑰政策相關聯的 KMS 金鑰。當提供跨帳戶許可時，KMS 金鑰的金鑰政策必須提供外部帳戶 (或外部帳戶中的使用者和角色) 使用 KMS 金鑰的許可。

若要提供外部帳戶使用 KMS 金鑰的許可，請新增陳述式至指定外部帳戶的金鑰政策。在金鑰政策的 `Principal` 元素中，輸入外部帳戶的 Amazon Resource Name (ARN)。

當您在金鑰政策中指定外部帳戶時，該外部帳戶中的 IAM 管理員可以使用 IAM 政策來將這些許可委派給外部帳戶中的任何使用者和角色。他們也能決定使用者和角色可以執行金鑰政策中所指定的哪些動作。

只有在託管 KMS 金鑰及其金鑰政策的區域中啟用外部帳戶時，授予給外部帳戶及其委託人的許可才有效。如需相關資訊了解哪些區域預設為未啟用 (「選擇加入區域」)，請參閱《AWS 一般參考》 的管理 AWS 區域。

例如，假設您想允許帳戶 444455556666 使用帳戶 111122223333 中的對稱加密 KMS 金鑰。若要執行此操作，請將政策陳述式新增至帳戶 111122223333 中 KMS 金鑰的金鑰政策，如以下範例所示。此政策陳述式會授予外部帳戶 444455556666 許可，在對稱加密 KMS 金鑰的密碼編譯操作中使用 KMS 金鑰。

> **Note**
>
> 下列範例顯示與其他帳戶共用 KMS 金鑰的金鑰政策範例。將範例 `Sid`、`Principal` 和 `Action` 值取代為 KMS 金鑰之預定用途的有效值。

```
{
    "Sid": "Allow an external account to use this KMS key",
    "Effect": "Allow",
    "Principal": {
        "AWS": [
            "arn:aws:iam::444455556666:root"
        ]
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*"
}
```

您可以在金鑰政策中指定特定的外部使用者和角色，而不用提供許可給外部帳戶。不過，除非外部帳戶中的 IAM 管理員將適當的 IAM 政策連接至各自身分，否則這些使用者和角色無法使用 KMS 金鑰。IAM 政策可以將許可提供給金鑰政策中指定的所有或部分外部使用者和角色。而且，其可允許金鑰政策中指定的全部動作或部分動作。

在金鑰政策中指定身分可以限制外部帳戶 IAM 管理員能提供的許可。然而，這會讓兩個帳戶的政策管理作業更加複雜。舉例來說，假設您需要新增使用者或角色。您必須將該身分新增至擁有 KMS 金鑰之帳戶中的金鑰政策，並在該身分的帳戶中建立 IAM 政策。

若要在金鑰政策中指定特定的外部使用者或角色，則請在 `Principal` 元素中輸入外部帳戶使用者或角色的 Amazon Resource Name (ARN)。

例如，以下金鑰政策陳述式範例允許帳戶 444455556666 中的 ExampleRole 使用帳戶 111122223333 中的 KMS 金鑰。此金鑰政策陳述式會授予外部帳戶 444455556666 許可，在對稱加密 KMS 金鑰的密碼編譯操作中使用 KMS 金鑰。

> **Note**
>
> 下列範例顯示與其他帳戶共用 KMS 金鑰的金鑰政策範例。將範例 Sid、Principal 和 Action 值取代為 KMS 金鑰之預定用途的有效值。

```
{
    "Sid": "Allow an external account to use this KMS key",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::444455556666:role/ExampleRole"
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*"
}
```

> **Note**
>
> 除非採用條件來限制金鑰政策，否則請勿在任何允許許可的金鑰政策陳述式將主體設為星號 (*)。星號為每個 AWS 帳戶 許可提供每個身分來使用 KMS 金鑰，除非另一個政策陳述式明確拒絕。其他 AWS 帳戶 的使用者只要其本身帳戶有對應的許可，即可利用您的 KMS 金鑰。

您也需要決定要提供哪些許可給外部帳戶。如需 KMS 金鑰上的許可清單，請參閱 AWS KMS 權限。

您可以提供外部帳戶在密碼編譯操作中使用 KMS 金鑰的許可，並搭配與 AWS KMS 整合的 AWS 服務使用 KMS 金鑰。若要執行這項操作，請使用 AWS Management Console 的 Key Users (金鑰使用者) 區段。如需詳細資訊，請參閱 建立其他帳戶可以使用的 KMS 金鑰。

若要在金鑰政策中指定其他許可，則需編輯金鑰政策文件。例如，您可能想提供使用者能解密但無法加密的許可，或是能檢視但無法使用 KMS 金鑰的許可。若要編輯金鑰原則文件，您可以使用AWS Management ConsoleCreateKey或PutKeyPolicy作業中的原則檢視。

步驟 2：在外部帳戶中新增 IAM 政策

擁有 KMS 金鑰之帳戶中的金鑰政策能夠設定許可的有效範圍。但是，在您連接能夠委派這些許可或使用授予來管理 KMS 金鑰存取權限的 IAM 政策前，外部帳戶中的使用者和角色都無法使用 KMS 金鑰。IAM 政策是在外部帳戶中設定。

如果金鑰政策是將許可提供給外部帳戶，您就能將 IAM 政策連接至該帳戶中的任何角色或使用者。但若金鑰政策是將許可提供給指定的使用者或角色，則 IAM 政策僅可提供這些許可給所有或部分指定的使用者和角色。如果 IAM 政策將 KMS 金鑰存取權限提供給其他外部使用者或角色，並不會起任何作用。

金鑰政策也能限制 IAM 政策中的動作。IAM 政策可以委派金鑰政策中指定的全部動作或部分動作。如果 IAM 政策列出金鑰政策中未指定的動作，則這些許可不會生效。

以下 IAM 政策範例允許委託人使用帳戶 111122223333 中的 KMS 金鑰來進行密碼編譯操作。若要提供此許可給帳戶 444455556666 中的使用者和角色，請連接政策至帳戶 444455556666 中的使用者或角色。

> **Note**
>
> 下列範例顯示與其他帳戶共用 KMS 金鑰的 IAM 政策範例。將範例 `Sid`、`Resource` 和 `Action` 值取代為 KMS 金鑰之預定用途的有效值。

```
{
    "Sid": "AllowUseOfKeyInAccount111122223333",
    "Effect": "Allow",
    "Action": [
      "kms:Encrypt",
      "kms:Decrypt",
      "kms:ReEncrypt*",
      "kms:GenerateDataKey*",
      "kms:DescribeKey"
    ],
    "Resource": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
```

```
}
```

請注意有關此政策的下列詳細資訊：

- 與金鑰政策不同，IAM 政策陳述式不包含 `Principal` 元素。在 IAM 政策中，委託人即為該政策欲連接的身分。
- IAM 政策中的 `Resource` 元素會識別該委託人可以使用的 KMS 金鑰。若要指定 KMS 金鑰，請將其金鑰 ARN 新增至 `Resource` 元素。
- 您可以在 `Resource` 元素中指定多個 KMS 金鑰。不過，如果您沒有在 `Resource` 元素中指定特定的 KMS 金鑰，則提供的 KMS 金鑰存取權限可能會不小心超過預期。
- 若要允許外部使用者搭配與 AWS KMS 整合的 AWS 服務使用 KMS 金鑰，您可能需要新增許可至金鑰政策或 IAM 政策。如需詳細資訊，請參閱 允許透過 AWS 服務 使用外部 KMS 金鑰。

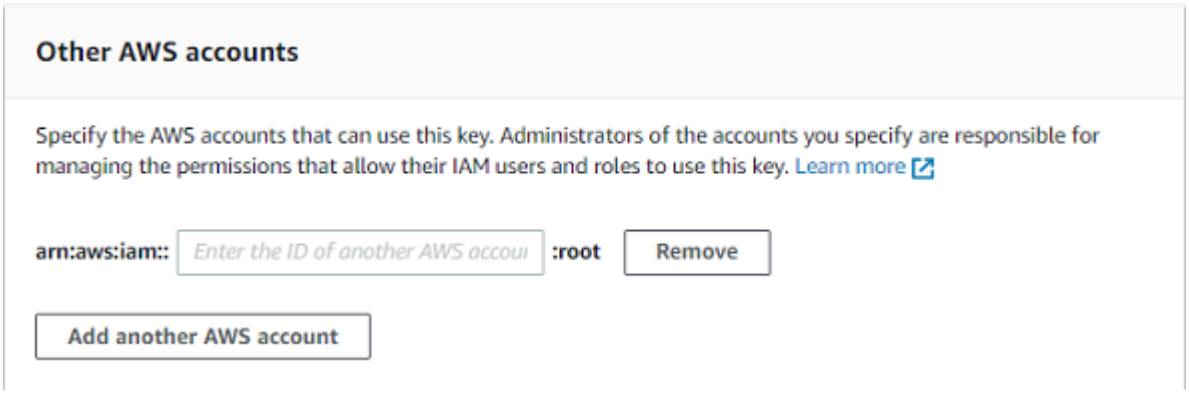
如需使用 IAM 政策的詳細資訊，請參閱 IAM 政策。

建立其他帳戶可以使用的 KMS 金鑰

當您使用此CreateKey作業建立 KMS 金鑰時，您可以使用其`Policy`參數來指定金鑰原則，以授予外部帳戶 (或外部使用者和角色) 使用 KMS 金鑰的權限。您還需要在外部帳戶中新增 IAM 政策，以便將這些許可委派給該帳戶的使用者和角色，即使是金鑰政策中指定的使用者和角色也一樣。您可以使用PutKeyPolicy作業隨時變更金鑰原則。

在 AWS Management Console 中建立 KMS 金鑰時，您可以一併建立其金鑰政策。當您在 Key Administrators (金鑰管理員) 和 Key Users (金鑰使用者) 區段中選取身分時，AWS KMS 會將這些身分的政策陳述式新增至 KMS 金鑰的金鑰政策。

Key Users (金鑰使用者) 區段也可讓您將外部帳戶做為金鑰使用者予以新增。



當您輸入外部帳戶的帳戶 ID 時，AWS KMS 會新增兩個陳述式至金鑰政策。此動作只會影響金鑰政策。在您連接 IAM 政策以提供部分或所有許可前，外部帳戶中的使用者和角色都無法使用 KMS 金鑰。

第一個金鑰政策陳述式會提供外部帳戶在密碼編譯操作中使用 KMS 金鑰的許可。

> **Note**
>
> 下列範例顯示與其他帳戶共用 KMS 金鑰的金鑰政策範例。將範例 `Sid`、`Principal` 和 `Action` 值取代為 KMS 金鑰之預定用途的有效值。

```
{
    "Sid": "Allow use of the key",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::444455556666:root"
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*"
}
```

第二個金鑰政策陳述式則允許外部帳戶建立、檢視和撤銷對 KMS 金鑰的授予，但只有在請求是來自與 AWS KMS 整合的 AWS 服務時才有效。這些許可允許其他加密使用者資料的 AWS 服務使用 KMS 金鑰。

這些權限專為 KMS 金鑰設計，用於加密AWS服務 (例如 Amazon) 中的使用者資料 WorkMail。這些服務通常會使用授予來取得所需的許可，以代表使用者使用 KMS 金鑰。如需詳細資訊，請參閱 允許透過 AWS 服務 使用外部 KMS 金鑰。

```
{
    "Sid": "Allow attachment of persistent resources",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::444455556666:root"
```

```
    },
    "Action": [
        "kms:CreateGrant",
        "kms:ListGrants",
        "kms:RevokeGrant"
    ],
    "Resource": "*",
    "Condition": {
        "Bool": {
            "kms:GrantIsForAWSResource": "true"
        }
    }
}
```

如果這些權限不符合您的需求，您可以在主控台原則檢視中或使用PutKeyPolicy作業來編輯這些權限。您可以指定特定的外部使用者和角色，而不用提供許可給外部帳戶。政策所指定的動作可加以變更。而且，您還可以使用全域和 AWS KMS 政策條件來調整許可。

允許透過 AWS 服務 使用外部 KMS 金鑰

您可以提供許可給不同帳戶中的使用者，讓他們能夠搭配與 AWS KMS 整合的服務使用 KMS 金鑰。舉例來說，外部帳戶中的使用者可以使用 KMS 金鑰來加密 Amazon S3 儲存貯體中的物件，或是加密存放在 AWS Secrets Manager 中的機密。

金鑰政策必須提供 KMS 金鑰使用許可給外部使用者或外部使用者的帳戶。此外，您還需要將 IAM 政策連接至能將 AWS 服務 使用許可提供給使用者的身分。服務也可能會要求使用者在金鑰政策或 IAM 政策中擁有額外許可。如需 AWS 服務 針對客戶受管金鑰所需的許可清單，請參閱該服務之使用者指南或開發人員指南中「安全性」一章中的「資料保護」主題。

在其他帳戶中使用 KMS 金鑰

如果您有權在不同 AWS 帳戶 中使用 KMS 金鑰，您可以在 AWS Management Console、AWS 開發套件、AWS CLI 和 AWS Tools for PowerShell 中使用 KMS 金鑰。

若要在 shell 命令或 API 請求中識別不同帳戶中的 KMS 金鑰，請使用下列金鑰識別符。

- 若要進行密碼編譯作業 DescribeKey，和 GetPublicKey，請使用 KMS 金鑰的金鑰 ARN 或別名 ARN。
- 對於CreateGrant、GetKeyRotationStatusListGrants、和 RevokeGrant，請使用 KMS 金鑰的金鑰 ARN。

如果您只輸入金鑰 ID 或別名名稱，則 AWS 會假設 KMS 金鑰在您的帳戶中。

AWS KMS 主控台不會在其他帳戶中顯示 KMS 金鑰，即使您有權使用這些金鑰。此外，在其他 AWS 服務的主控台中顯示的 KMS 金鑰清單不會在其他帳戶中包含 KMS 金鑰。

若要在 AWS 服務的主控台中指定不同帳戶中的 KMS 金鑰，您必須輸入 KMS 金鑰的金鑰 ARN 或別名 ARN。必要的金鑰識別符會隨服務而有所不同，服務主控台及其 API 操作可能會有所不同。如需詳細資訊，請參閱服務文件。

使用 AWS KMS 的服務連結角色

AWS Key Management Service 使用 AWS Identity and Access Management (IAM) 服務連結的角色。服務連結角色是直接連結至 AWS KMS 的一種特殊 IAM 角色類型。服務連結角色由 AWS KMS 所定義，且包含服務代替您呼叫其他 AWS 服務所需的所有許可。

服務連結的角色可讓設定 AWS KMS 更為簡單，因為您不必手動新增必要的許可。AWS KMS 定義其服務連結角色的許可，除非另有定義，否則僅有 AWS KMS 可以擔任其角色。定義的許可包括信任政策和許可政策，且該許可政策無法附加至其他 IAM 實體。

您必須先刪除相關的資源，才能刪除服務連結角色。如此可保護您 AWS KMS 的資源，避免您不小心移除資源的存取許可。

如需關於支援服務連結角色的其他服務的資訊，請參閱可搭配 IAM 運作的 AWS 服務，並尋找服務連結角色欄顯示為是的服務。選擇具有連結的是，以檢視該服務的服務連結角色文件。

AWS KMS 自訂金鑰存放區的服務連結角色許可

AWS KMS使用名為的服務連結角色AWSServiceRoleForKeyManagementServiceCustomKeyStores來支援自訂金鑰存放區。此服務連結角色為 AWS KMS 提供檢視 AWS CloudHSM 叢集和建立網路基礎設施的許可，以支援自訂金鑰存放區與其 AWS CloudHSM 叢集之間的連線。AWS KMS 只會在您建立自訂金鑰存放區時建立此角色。您無法直接建立此服務連結角色。

AWSServiceRoleForKeyManagementServiceCustomKeyStores 服務連結角色信任 `cks.kms.amazonaws.com` 擔任此角色。因此，只有 AWS KMS 可擔任此服務連結角色。

此角色的許可僅限於讓 AWS KMS 將自訂金鑰存放區連接到 AWS CloudHSM 叢集而執行的動作。並不授予 AWS KMS 任何額外的許可。例如，AWS KMS 沒有許可來建立、管理或刪除 AWS CloudHSM 叢集、HSM 或備份。

如需 AWSServiceRoleForKeyManagementServiceCustomKeyStores 角色的詳細資訊，包括許可清單，以及有關如何檢視角色、編輯角色描述、刪除角色和讓 AWS KMS 為您重新建立角色的指示，請參閱授權 AWS KMS 來管理 AWS CloudHSM 和 Amazon EC2 資源。

AWS KMS 多區域金鑰的服務連結角色許可

AWS KMS使用名為的服務連結角色AWSServiceRoleForKeyManagementServiceMultiRegionKeys來支援多區域金鑰。此服務連結角色為 AWS KMS 提供將多區域主要金鑰之金鑰材料的任何變更同步至其複本金鑰的許可。AWS KMS 只會在您建立多區域主要金鑰時建立此角色。您無法直接建立此服務連結角色。

AWSServiceRoleForKeyManagementServiceMultiRegionKeys 服務連結角色信任 `mrk.kms.amazonaws.com` 擔任此角色。因此，只有 AWS KMS 可擔任此服務連結角色。此角色的許可僅限於 AWS KMS 執行的動作，用於保持相關多區域金鑰中的金鑰材料同步。並不授予 AWS KMS 任何額外的許可。

如需 AWSServiceRoleForKeyManagementServiceMultiRegionKeys 角色的詳細資訊，包括許可清單，以及有關如何檢視角色、編輯角色描述、刪除角色和讓 AWS KMS 為您重新建立角色的指示，請參閱授權 AWS KMS 同步多區域金鑰。

AWS 管理的政策的 AWS KMS 更新項目

檢視自 AWS KMS 開始追蹤 AWS 管理的政策變更以來的更新詳細資訊。如需有關此頁面變更的自動提醒，請訂閱 AWS KMS 文件歷史紀錄 頁面的 RSS 摘要。

變更	描述	日期
AWSKeyManagementServiceCustomKeyStoresServiceRolePolicy – 更新現有政策	AWS KMS新增了`ec2:DescribeVpcs` `ec2:DescribeNetworkAcls` 、和`ec2:DescribeNetworkInterfaces` 權限來監視 VPC 中包含AWS CloudHSM叢集的變更，AWS KMS以便在發生故障時提供明確的錯誤訊息。	2023 年 11 月 10 日
AWS KMS 已開始追蹤變更	AWS KMS 已開始追蹤其 AWS 管理的政策的變更。	2023 年 11 月 10 日

搭配 AWS KMS 使用混合式後量子 TLS

AWS Key Management Service (AWS KMS) 支援適用於 Transport Layer Security (TLS) 網路加密通訊協定的混合式後量子金鑰交換選項。連線至 AWS KMS API 端點時，您可使用此 TLS 選項。在後量子演算法標準化前，我們會提供此功能，以便您可以開始測試這些金鑰交換通訊協定對於 AWS KMS 呼叫的影響。這些選用的混合式後量子金鑰交換功能至少與現今使用的 TLS 加密功能同樣安全，且還能提供其他長期安全優勢。不過，與現今使用的傳統金鑰交換通訊協定相較之下，這些功能會影響延遲和輸送量。

傳送至 AWS Key Management Service (AWS KMS) 資料會在傳輸中受到 Transport Layer Security (TLS) 連線提供的加密保護。AWS KMS 支援可用於 TLS 工作階段的傳統密碼套件會使得暴力破解攻擊在現今的科技下變得毫無用武之地。不過如果大規模量子運算在未來變得可行，那麼用於 TLS 金鑰交換機制的傳統密碼套件將容易遭受這些攻擊影響。如果您正在開發仰賴透過 TLS 連線傳遞之資料的長期機密性的應用程式，則應在大規模量子電腦問世可用之前，考慮遷移到後量子加密法的計畫。AWS 正在努力為此未來做好準備，而且我們也想要讓您做好充分準備。

若要保護現今加密的資料防範潛在的未來攻擊，AWS 正在與密碼編譯社群攜手合作開發抵禦量子或後量子演算法。我們已在 AWS KMS 中實作混合式後量子金鑰交換密碼套件，它們結合了傳統與後量子元素，能夠確保您的 TLS 連線至少與使用傳統密碼套件一樣堅強。

這些混合式加密套件可用於大部分 AWS 區域中的生產工作負載。然而，由於混合式密碼套件的效能特性與頻寬要求的緣故，我們建議您在不同的條件下對 AWS KMS API 呼叫測試這些套件。

意見回饋

我們總是歡迎您提供意見回饋，並參加我們的開放原始碼儲存庫。我們特別想要了解您的基礎設施如何與此新版的 TLS 流量互動。

- 若要提供有關此主題的意見回饋，請使用本頁右上角的意見回饋連結。
- 我們正在開放原始碼中開發這些混合密碼套件。s2n-tls GitHub若要提供密碼套件可用性的意見回饋，或分享新的測試條件或結果，請在 s2n-tls 儲存庫中建立問題。
- 我們正AWS KMS在撰寫程式碼範例，以便在aws-kms-pq-tls-example GitHub儲存庫中使用混合式後量子 TLS。若要提出問題或分享有關如何設定 HTTP 用戶端或 AWS KMS 用戶端以使用混合式加密套件的想法，請在 aws-kms-pq-tls-example 儲存庫中建立問題。

支援 AWS 區域

AWS KMS 的後量子 TLS 可用於所有 AWS KMS 支援的 AWS 區域，但中國 (北京) 與中國 (寧夏) 除外。

> **Note**
>
> AWS KMS 不支援 AWS GovCloud (US) 中 FIPS 端點的混合式後量子 TLS。

如需各 AWS 區域 的 AWS KMS 端點清單，請參閱《Amazon Web Services 一般參考》的AWS Key Management Service端點與配額。如需有關 FIPS 端點的詳細資訊，請參閱《Amazon Web Services 一般參考》的 FIPS 端點。

關於 TLS 中的混合式後量子金鑰交換

AWS KMS 支援混合式後量子金鑰交換密碼套件。您可以在 Linux 系統上使用 AWS SDK for Java 2.x 和 AWS 通用執行時間來設定使用這些加密套件的 HTTP 用戶端。然後每次將 AWS KMS 端點連接到 HTTP 用戶端時，則會使用混合式密碼套件。

此 HTTP 客戶端使用 s2n-tls，這是 TLS 協定的開源實作。s2n-tls 使用的混合式密碼套件僅針對金鑰交換實作，而非針對直接資料加密。在金鑰交換時，用戶端與伺服器會計算其用於加密與解密線路上資料的金鑰。

s2n-tls 使用的演算法是混合演算法，結合橢圓曲線 Diffie-Hellman (ECDH)，這是目前 TLS 中使用的經典金鑰交換演算法)，以及 Kyber，這是美國國家標準與技術研究院 (NIST) 已指定作為其第一個標準後量子金鑰協議的公有金鑰加密和金鑰建立演算法。此混合會獨立使用各演算法，以產生金鑰。然後以密碼編譯方式結合兩個金鑰。使用 s2n-tls，您可以將 HTTP 用戶端設定為偏好後量子 TLS，將具有 Kyber 的 ECDH 置於偏好清單的首位。為了確保相容性，傳統金鑰交換演算法仍包含在偏好設定清單中，但偏好設定順序較低。

如果持續的研究顯示，Kyber 演算法欠缺預期的後量子強度，則混合式金鑰至少仍根目前使用的單一 ECDH 金鑰同樣堅強。在後量子演算法的研究完成之前，我們建議使用混合型演算法，而不要單獨使用後量子演算法。

搭配 AWS KMS 使用混合式後量子 TLS

您可以將後量子演算法 TLS 用於對 AWS KMS 的呼叫上。設定 HTTP 用戶端測試環境時，請注意下列資訊：

傳輸中加密

s2n-tls 中的混合式密碼套件僅用於傳輸中加密。這些套件會在資料從用戶端傳輸至 AWS KMS 端點時保護資料。AWS KMS 不會使用這些加密套件，以 AWS KMS keys 來將資料加密。

而是當 AWS KMS 以 KMS 金鑰將資料加密時，它會採用對稱金鑰加密法，使用 256 位元金鑰和 Galois 計數器模式 (AES-GCM) 演算法中的進階加密標準，而此方法足以抵禦量子攻擊。在理論上，未來針對以 256 位元 AES-GCM 金鑰建立的加密文字所做的大規模量子運算攻擊，會將金鑰的有效安全性降低到 128 位元。這種安全等級足以讓針對 AWS KMS 加密文字所做的暴力破解攻擊變得不可行。

支援的系統

目前僅支援在 Linux 系統上使用 s2n-tls 中的混合式密碼套件。此外，這些加密套件僅在支援 AWS 常用執行時間 (如 AWS SDK for Java 2.x) 的軟體開發套件中獲得支援。如需範例，請參閱 如何設定混合式後量子 TLS。

AWS KMS 端點

使用混合式密碼套件，請使用標準 AWS KMS 端點。s2n-tls 中的混合式密碼套件不相容於 適用於 AWS KMS 的 FIPS 140-2 驗證端點。

將 HTTP 用戶端設定為偏好使用 s2n-tls 的後量子 TLS 連接時，後量子密碼在密碼偏好清單中排在首位。不過，為了確保相容性，偏好設定清單包括在偏好設定順序中較低的傳統、非混合式密碼。當您將 HTTP 用戶端設定為偏好使用 AWS KMS FIPS 140-2 驗證端點的後量子 TLS 時，s2n-tls 會協商傳統、非混合式金鑰交換密碼。

如需各 AWS 區域 的 AWS KMS 端點清單，請參閱《Amazon Web Services 一般參考》的AWS Key Management Service端點與配額。如需有關 FIPS 端點的詳細資訊，請參閱《Amazon Web Services 一般參考》的 FIPS 端點。

預期效能

我們的早期基準測試指出，在 s2n-tls 中的混合式密碼套件比傳統 TLS 密碼套件還慢。cipher suites。此影響會隨著網路設定檔、CPU 速度、核心數，以及您的通話費率而改變。如需效能測試結果，請參閱如何使用 Kyber 針對混合後量子密碼學調整 TLS。

如何設定混合式後量子 TLS

在此處理程序中，為 AWS 通用執行時間 HTTP 用戶端新增 Maven 相依性。接下來，設定偏好後量子 TLS 的 HTTP 用戶端。然後，建立使用 HTTP 用戶端的 AWS KMS 用戶端。

若要查看設定與搭配 AWS KMS 使用混合式後量子 TLS 的完整可用範例，請參閱 aws-kms-pq-tls-example 儲存庫。

> **Note**
>
> AWS 通用執行時間 HTTP 用戶端已作為預覽版提供，於 2023 年 2 月正式推出。在該版本中，`tlsCipherPreference` 類別和 `tlsCipherPreference()` 方法參數由 `postQuantumTlsEnabled()` 方法參數取代。如果您在預覽期間使用此範例，則需要更新程式碼。

1. 將 AWS 通用執行時間用戶端新增至 Maven 相依性。我們建議使用最新的可用版本。

   例如，這項陳述式將 AWS 通用執行時間用戶端的版本 `2.20.0` 新增至 Maven 相依性。

   ```
   <dependency>
       <groupId>software.amazon.awssdk</groupId>
       <artifactId>aws-crt-client</artifactId>
       <version>2.20.0</version>
   </dependency>
   ```

2. 若要啟用混合式後量子密碼套件，請將 AWS SDK for Java 2.x 新增至專案並初始化它。然後如下列範例所示在 HTTP 用戶端啟用混合式後量子密碼套件。

   此程式碼使用 `postQuantumTlsEnabled()` 方法參數來設定 AWS 通用執行時間 HTTP 用戶端，該用戶端偏好推薦的混合式後量子加密套件，具有 Kyber 的 ECDH。然後，其會使用已設定的 HTTP 用戶端來建置 AWS KMS 非同步用戶端的執行個體，`KmsAsyncClient`。完成此程式碼之後，`KmsAsyncClient` 執行個體上的所有 AWS KMS API 請求都會使用混合式後量子 TLS。

   ```
   // Configure HTTP client
   SdkAsyncHttpClient awsCrtHttpClient = AwsCrtAsyncHttpClient.builder()
              .postQuantumTlsEnabled(true)
              .build();

   // Create the AWS KMS async client
   KmsAsyncClient kmsAsync = KmsAsyncClient.builder()
             .httpClient(awsCrtHttpClient)
             .build();
   ```

3. 使用混合式後量子 TLS 測試您的 AWS KMS 呼叫。

   當您在已設定的 AWS KMS 用戶端呼叫 AWS KMS API 操作時，就會使用混合式後量子 TLS 將您的呼叫傳輸至 AWS KMS 端點。若要測試您的組態，請呼叫 AWS KMS API，例如 `ListKeys`。

```
ListKeysReponse keys = kmsAsync.listKeys().get();
```

搭配 AWS KMS 測試混合式後量子 TLS

請考慮在呼叫 AWS KMS 的應用程式上對混合式密碼套件直行下列測試。

- 執行負載測試和基準測試。混合式密碼套件的執行方式不同於傳統的金鑰交換演算法。您可能需要調整連線逾時，以允許較長的交握時間。如果您在 AWS Lambda 函式內執行，請延長執行逾時設定。
- 請嘗試從不同位置進行連線。視您請求佔用的網路路徑而定，您必須探索中繼主機、代理或採用深層封包檢查 (DPI) 的防火牆是否封鎖請求。這可能是因為在 TLS 握手中使用新的加密套ClientHello件，或是來自較大的金鑰交換訊息所導致。如果您解決這些問題的能力有限，請與您的安全團隊或 IT 管理員合作，以更新相關組態並解除對於新 TLS 密碼套件的封鎖。

進一步了解 AWS KMS 中的後量子 TLS

如需在 AWS KMS 中使用混合式後量子 TLS 的詳細資訊，請參閱下列資源。

- 要了解有關 AWS 的後量子密碼學的資訊，包括部落格文章和研究論文的連結，請參閱後量子密碼學。
- 如需 s2n-tls 的資訊，請參閱 全新開放原始碼 TLS 實作 s2n-tls 簡介和使用 s2n-tls。
- 如需有關 AWS 通用執行時間 HTTP 用戶端的資訊，請參閱《AWS SDK for Java 2.x 開發人員指南》中的設定 AWS CRT 型 HTTP 用戶端。
- 如需國家標準技術研究 (NIST) 的後量子加密法專案的資訊，請參閱後量子加密法。
- 如需有關 NIST 後量子密碼學標準化的資訊，請參閱後量子密碼學標準化。

判斷 AWS KMS keys 的存取權

若要完整判斷何人或何物目前可存取 AWS KMS key，您必須檢查 KMS 金鑰的金鑰政策、所有套用至 KMS 金鑰的授予，以及所有可能的 AWS Identity and Access Management (IAM) 政策。您可以這樣做來判斷 KMS 金鑰的潛在使用範圍，可協助您符合規範或稽核要求。您可以善用下列主題，藉此產生目前有權存取 KMS 金鑰的 AWS 主體 (身分) 完整清單。

主題

- 檢查金鑰政策

- 檢查 IAM 政策
- 檢查授與
- 對金鑰存取進行故障診斷

檢查金鑰政策

金鑰政策是控制對 KMS 金鑰之存取的主要方式。每個 KMS 金鑰只有一個金鑰政策。

當金鑰政策包含預設金鑰政策時，金鑰政策可讓帳戶中的 IAM 管理員使用 IAM 政策控制 KMS 金鑰的存取。此外，如果金鑰政策授權另一個 AWS 帳戶 使用 KMS 金鑰，則外部帳戶的 IAM 管理員可以使用 IAM 政策委派這些許可。請檢查 IAM 政策，以判斷可存取 KMS 金鑰之主體的完整清單。

若要檢視AWS KMS客戶管理金鑰或您帳戶AWS 受管金鑰中的金鑰政策，請使用 AWS KMS API 中的AWS Management Console或GetKeyPolicy作業。若要檢視金鑰政策，您必須具有 KMS 金鑰的 `kms:GetKeyPolicy` 許可。如需檢視 KMS 金鑰之金鑰政策的說明，請參閱 the section called “檢視金鑰政策”。

檢查金鑰政策文件並記下在各政策陳述式 `Principal` 元素中指定的所有主體。在具有 `Allow` 效用的政策陳述式中，IAM 使用者、IAM 角色以及 `Principal` 元素中的 AWS 帳戶 可以存取此 KMS 金鑰。

> **Note**
>
> 除非採用條件來限制金鑰政策，否則請勿在任何允許許可的金鑰政策陳述式將主體設為星號 (*)。星號為每個 AWS 帳戶 許可提供每個身分來使用 KMS 金鑰，除非另一個政策陳述式明確拒絕。其他 AWS 帳戶 的使用者只要其本身帳戶有對應的許可，即可利用您的 KMS 金鑰。

以下範例使用預設金鑰政策中的政策陳述式來示範如何執行此操作。

Example 政策陳述式 1

```
{
  "Sid": "Enable IAM User Permissions",
  "Effect": "Allow",
  "Principal": {"AWS": "arn:aws:iam::111122223333:root"},
  "Action": "kms:*",
  "Resource": "*"
}
```

在政策陳述式 1 中，arn:aws:iam::111122223333:root 是 AWS 帳戶主體，其指的是 AWS 帳戶 111122223333。(其不是帳戶根使用者。) 在預設情況下，當您使用 AWS Management Console 建立新的 KMS 金鑰，或者以程式設計方式建立新的 KMS 金鑰但不提供金鑰政策時，這類政策陳述式會包含在金鑰政策文件中。

具有允許 AWS 帳戶 存取權之陳述式的金鑰政策文件，會在帳戶中啟用 IAM 政策以允許存取 KMS 金鑰。這表示帳戶中的使用者和角色可能可以存取 KMS 金鑰，即使他們未在金鑰政策文件中被明確列為主體。仔細檢查所有列為主體之 AWS 帳戶 中的所有 IAM 政策，判斷其是否允許存取此 KMS 金鑰。

Example 政策陳述式 2

```
{
  "Sid": "Allow access for Key Administrators",
  "Effect": "Allow",
  "Principal": {"AWS": "arn:aws:iam::111122223333:role/KMSKeyAdmins"},
  "Action": [
    "kms:Describe*",
    "kms:Put*",
    "kms:Create*",
    "kms:Update*",
    "kms:Enable*",
    "kms:Revoke*",
    "kms:List*",
    "kms:Disable*",
    "kms:Get*",
    "kms:Delete*",
    "kms:ScheduleKeyDeletion",
    "kms:CancelKeyDeletion"
  ],
  "Resource": "*"
}
```

在政策聲明 2 中，arn:aws:iam::111122223333:role/KMSKeyAdmins指的是在 AWS 帳戶 111122223333 中名為 KMS KeyAdmins 的 IAM 角色。被授權擔任此角色的使用者可以執行政策陳述式中所列的動作，這些是用於管理 KMS 金鑰的管理動作。

Example 政策陳述式 3

```
{
  "Sid": "Allow use of the key",
  "Effect": "Allow",
```

```
  "Principal": {"AWS": "arn:aws:iam::111122223333:role/EncryptionApp"},
  "Action": [
    "kms:DescribeKey",
    "kms:GenerateDataKey*",
    "kms:Encrypt",
    "kms:ReEncrypt*",
    "kms:Decrypt"
  ],
  "Resource": "*"
}
```

在政策聲明 3 中，arn:aws:iam::111122223333:role/EncryptionApp指的是在 AWS 帳戶 111122223333 EncryptionApp 中指定的 IAM 角色。被授權擔任此角色的主體可以執行政策陳述式中所列的動作，包括對稱加密 KMS 金鑰的密碼編譯操作。

Example 政策陳述式 4

```
{
  "Sid": "Allow attachment of persistent resources",
  "Effect": "Allow",
  "Principal": {"AWS": "arn:aws:iam::111122223333:role/EncryptionApp"},
  "Action": [
    "kms:ListGrants",
    "kms:CreateGrant",
    "kms:RevokeGrant"
  ],
  "Resource": "*",
  "Condition": {"Bool": {"kms:GrantIsForAWSResource": true}}
}
```

在政策聲明 4 中，arn:aws:iam::111122223333:role/EncryptionApp指的是在 AWS 帳戶 111122223333 EncryptionApp 中指定的 IAM 角色。被授權擔任此角色的主體能夠執行政策陳述式中所列的動作。當這些動作結合範例政策陳述式 3 中允許的動作時，是委派 KMS 金鑰的使用給大部分與 AWS KMS 整合的 AWS 服務 (尤其是使用授予的服務) 的必要動作。「kms:」Condition 元素中的 GrantIsForAWSResource值可確保僅當委派是與授權整合AWS KMS並使用授權的AWS服務時，才允許委派。

若要了解在金鑰政策文件中指定主體的所有不同方法，請參閱《IAM 使用者指南》中的 Specifying a Principal (指定主體)。

若要進一步了解 AWS KMS 金鑰政策，請參閱AWS KMS 中的金鑰政策。

檢查 IAM 政策

除了金鑰政策和授予之外，您也可以使用 IAM 政策，以允許存取 KMS 金鑰。如需金鑰政策與 IAM 政策如何同時運作的相關資訊，請參閱 對金鑰存取進行故障診斷。

若要判斷哪些主體目前可透過 IAM 政策存取 KMS 金鑰，您可以使用以瀏覽器為基礎的 IAM 政策模擬器工具，或者可以提出請求至 IAM API。

檢查 IAM 政策的方式

- 使用 IAM 政策模擬器來檢查 IAM 政策
- 使用 IAM API 檢查 IAM 政策

使用 IAM 政策模擬器來檢查 IAM 政策

IAM 政策模擬器可協助您了解哪些原則有權限透過 IAM 政策來存取 KMS 金鑰。

若要使用 IAM 政策模擬器來以判斷是否可存取 KMS 金鑰

1. 登入 AWS Management Console，然後開啟 IAM 政策模擬器，網址為 https://policysim.aws.amazon.com/。
2. 在使用者、群組和角色窗格，選擇您想要模擬所屬政策的使用者、群組或角色。
3. (選用) 清除任何您想要從模擬中刪除的政策旁的核取方塊。若要模擬所有政策，請保持勾選所有政策。
4. 在 Policy Simulator (政策模擬器) 窗格中，執行下列動作：
   a. 在 Select service (選取服務) 中，選擇 Key Management Service (金鑰管理服務)。
   b. 若要模擬特定的 AWS KMS 動作，請在 Select actions (選取動作) 中選擇要模擬的動作；若要模擬所有 AWS KMS 動作，則請選擇 Select All (全選)。
5. (選用) 政策模擬器根據預設將模擬所有對 KMS 金鑰的存取。若要模擬存取特定 KMS 金鑰，請選擇 Simulation Settings (模擬設定)，接著輸入要模擬之 KMS 金鑰的 Amazon Resource Name (ARN)。
6. 選擇 Run Simulation (執行模擬)。

您可在 Results (結果) 部分查看模擬結果。對於 AWS 帳戶 中的每個 IAM 使用者、群組和角色，請重複步驟 2 到步驟 6。

使用 IAM API 檢查 IAM 政策

您可以使用 IAM API 來以程式設計方式檢查 IAM 政策。以下步驟提供執行此操作方法的一般概述：

1. 對於在金鑰政策中AWS 帳戶列為主體的每個主體 (亦即，以此格式指定的每個AWS帳戶主體:`"Principal": {"AWS": "arn:aws:iam::111122223333:root"}`)，請使用 IAM API 中的ListUsers和ListRoles操作來取得帳戶中的所有使用者和角色。
2. 針對清單中的每個使用者和角色，使用 IAM API 中的SimulatePrincipalPolicy作業，並傳入下列參數：
   - 針對 `PolicySourceArn`，請指定清單中使用者或角色的 Amazon Resource Name (ARN)。您只能為每個 `SimulatePrincipalPolicy` 請求指定一個 `PolicySourceArn`，因此您必須多次呼叫此操作，針對清單中的每個使用者與角色各呼叫一次。
   - 針對 `ActionNames` 清單，請指定要模擬的每個 AWS KMS API 動作。若要模擬所有 AWS KMS API 動作，請使用 `kms:*`；若要測試個別的 AWS KMS API 動作，請在每個 API 動作前加上「`kms:`」，例如「`kms:ListKeys`」。如需 AWS KMS API 動作的完整清單，請參閱 AWS Key Management Service API 參考中的動作。
   - (選用) 若要判斷使用者或角色是否可以存取特定 KMS 金鑰，請使用 `ResourceArns` 參數來指定 KMS 金鑰的 Amazon Resource Name (ARN) 清單。若要判斷使用者或角色是否可以存取任何 KMS 金鑰，請忽略 `ResourceArns` 參數。

IAM 將以下列評估決定來回應每個 `SimulatePrincipalPolicy` 請求：`allowed`、`explicitDeny` 或 `implicitDeny`。對於包含評估決策 `allowed` 的每個回應，回應中會包含允許的特定 AWS KMS API 操作的名稱。還會包含評估時所使用的 KMS 金鑰的 ARN (如果有)。

檢查授與

授予是一種進階機制，用來指定您或與 AWS KMS 整合的 AWS 服務可用來指定如何以及何時使用 KMS 金鑰的許可。授予會連接到 KMS 金鑰，而且每個授予包含獲得使用 KMS 金鑰之許可的主體以及允許的操作清單。授與是金鑰政策的替代選項，適用於特定的使用案例。如需詳細資訊，請參閱 AWS KMS 中的授權。

若要取得 KMS 金鑰的授權清單，請使用此AWS KMSListGrants作業。您可以檢查 KMS 金鑰的授予，以判斷何人或何物目前具有透過這些授予使用 KMS 金鑰的存取權。例如，以下是從 AWS CLI 中 list-grants 命令取得的 JSON 授予表示。

```
{"Grants": [{
  "Operations": ["Decrypt"],
```

```
  "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "Name": "0d8aa621-43ef-4657-b29c-3752c41dc132",
  "RetiringPrincipal": "arn:aws:iam::123456789012:root",
  "GranteePrincipal": "arn:aws:sts::111122223333:assumed-role/aws:ec2-infrastructure/
i-5d476fab",
  "GrantId": "dc716f53c93acacf291b1540de3e5a232b76256c83b2ecb22cdefa26576a2d3e",
  "IssuingAccount": "arn:aws:iam::111122223333:root",
  "CreationDate": 1.444151834E9,
  "Constraints": {"EncryptionContextSubset": {"aws:ebs:id": "vol-5cccfb4e"}}
}]}
```

若要了解何人或何物具有使用 KMS 金鑰的存取權，請查看 "GranteePrincipal" 元素。在上述範例中，承授者主體是與 EC2 執行個體 i-5d476fab 相關聯的擔任角色使用者。EC2 基礎設施使用此角色，將加密的 EBS 磁碟區 vol-5cccfb4e 連接到執行個體。在這種情況下，EC2 基礎設施角色有許可使用 KMS 金鑰，因為您之前建立受此 KMS 金鑰保護的加密 EBS 磁碟區。然後，您將此磁碟區連接到 EC2 執行個體。

以下是從 AWS CLI 中 list-grants 命令取得的另一個範例 JSON 授予表示。在下列範例中，承授者主體是另一個 AWS 帳戶。

```
{"Grants": [{
  "Operations": ["Encrypt"],
  "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
  "Name": "",
  "GranteePrincipal": "arn:aws:iam::444455556666:root",
  "GrantId": "f271e8328717f8bde5d03f4981f06a6b3fc18bcae2da12ac38bd9186e7925d11",
  "IssuingAccount": "arn:aws:iam::111122223333:root",
  "CreationDate": 1.444151269E9
}]}
```

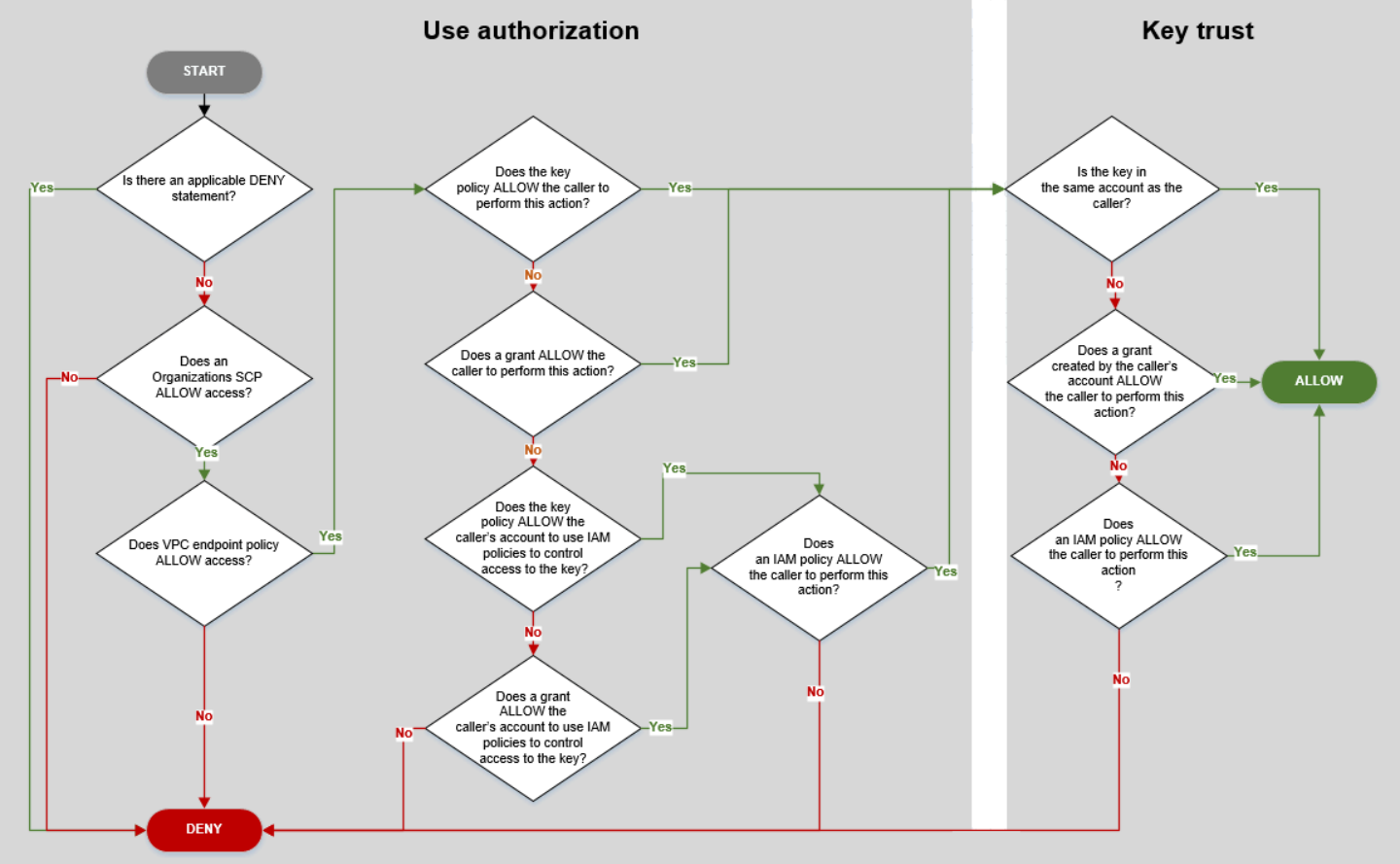
對金鑰存取進行故障診斷

當授權存取 KMS 金鑰時，AWS KMS 會評估下列項目：

- 連接到 KMS 金鑰的金鑰政策。金鑰政策一律在擁有 KMS 金鑰的 AWS 帳戶 和區域中定義。
- 連接到提出請求的使用者或角色的所有 IAM 政策。管理主體使用 KMS 金鑰的 IAM 政策一律在主體的 AWS 帳戶 中定義。
- 適用於 KMS 金鑰的所有授予。

- 其他可能套用至使用 KMS 金鑰之請求的政策類型，例如 AWS Organizations 服務控制政策和 VPC 端點政策。這些政策是選用的，依預設允許所有動作，但您可以使用其來限制許可，否則會將許可授予主體。

AWS KMS 會同時評估這些政策機制，以判定允許或拒絕存取 KMS 金鑰。若要這樣做，AWS KMS 會使用與以下流程圖中描述的類似程序。以下流程圖提供政策評估程序的視覺化呈現。



此流程圖分為兩個部分。這兩個部分應有其先後順序，但評估通常會同時進行。

- 使用授權可根據其金鑰政策、IAM 政策、授予和其他適用政策決定是否允許您使用 KMS 金鑰。
- 金鑰信任會決定您是否應信任您被允許使用的 KMS 金鑰。一般而言，您會信任 AWS 帳戶 中的資源。但是，如果您帳戶中的授予或 IAM 政策允許您使用 KMS 金鑰，您也可以放心在不同的 AWS 帳戶 中使用 KMS 金鑰。

您可以使用此流程圖來探索呼叫者被允許或拒絕使用 KMS 金鑰許可的原因。您也可以使用它來評估政策和授予。例如，流程圖會顯示呼叫者的存取被拒絕可能是由於明確 DENY 陳述式或金鑰政策、IAM 政策或授予中缺少明確 ALLOW 陳述式。

流程圖可以說明一些常見許可案例。

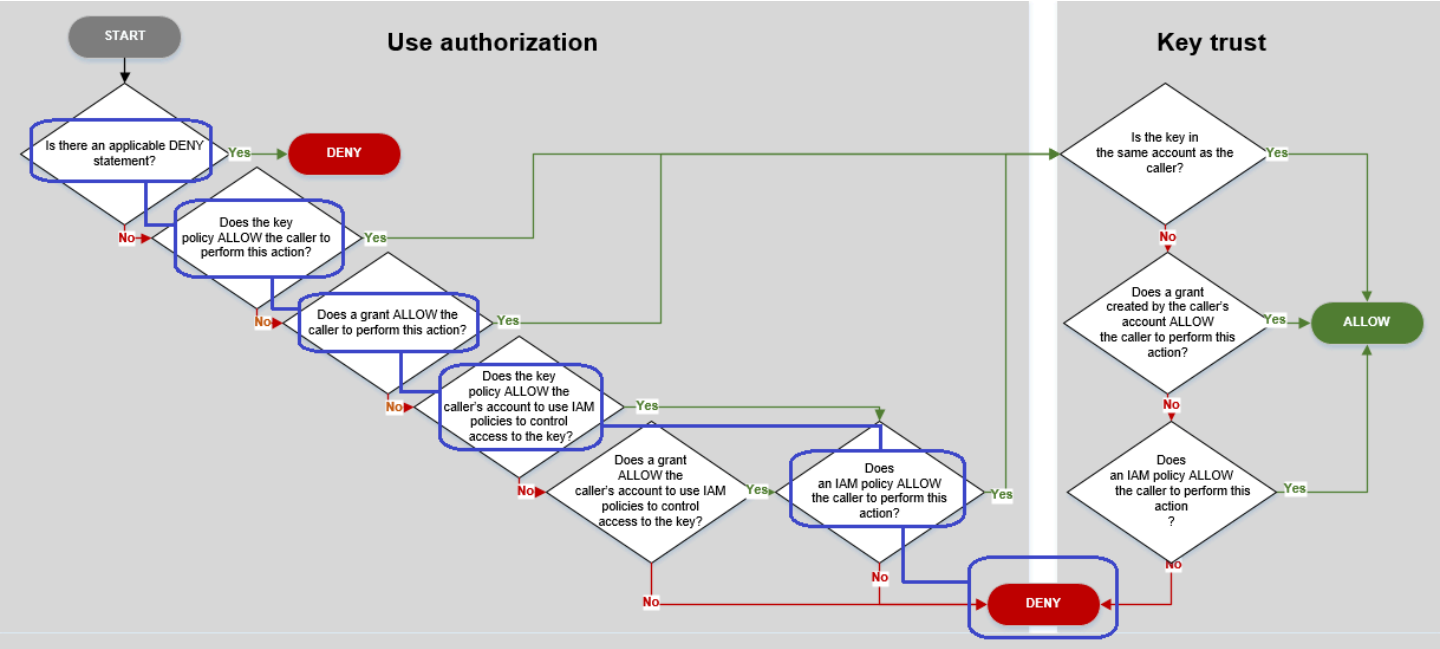
許可範例

- 範例 1：拒絕使用者存取其 AWS 帳戶 中的 KMS 金鑰
- 範例 2：使用者採用的角色具有在不同 AWS 帳戶 中使用 KMS 金鑰的許可

範例 1：拒絕使用者存取其 AWS 帳戶 中的 KMS 金鑰

Alice 是 111122223333 AWS 帳戶 中的 IAM 使用者。她被拒絕存取同一個 AWS 帳戶 中的 KMS 金鑰。為什麼 Alice 無法使用 KMS 金鑰？

在此案例中，Alice 對 KMS 金鑰的存取遭拒，因為沒有賦予她必要許可的金鑰政策、IAM 政策或授予。KMS 金鑰的金鑰政策允許 AWS 帳戶 使用 IAM 政策來控制對 KMS 金鑰的存取，但無 IAM 政策授予 Alice 使用 KMS 金鑰的許可。



請考量此範例的相關政策。

- Alice 想要使用的 KMS 金鑰具有預設金鑰政策。本政策允許擁有 KMS 金鑰的 AWS 帳戶，以使用 IAM 政策控制對 KMS 金鑰的存取。此金鑰政策符合流程圖中的金鑰政策是否允許呼叫者帳戶使用 IAM 政策來控制對金鑰的存取？條件。

```
{
  "Version" : "2012-10-17",
  "Id" : "key-test-1",
```

```
  "Statement" : [ {
    "Sid" : "Delegate to IAM policies",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "arn:aws:iam::111122223333:root"
    },
    "Action" : "kms:*",
    "Resource" : "*"
  } ]
}
```

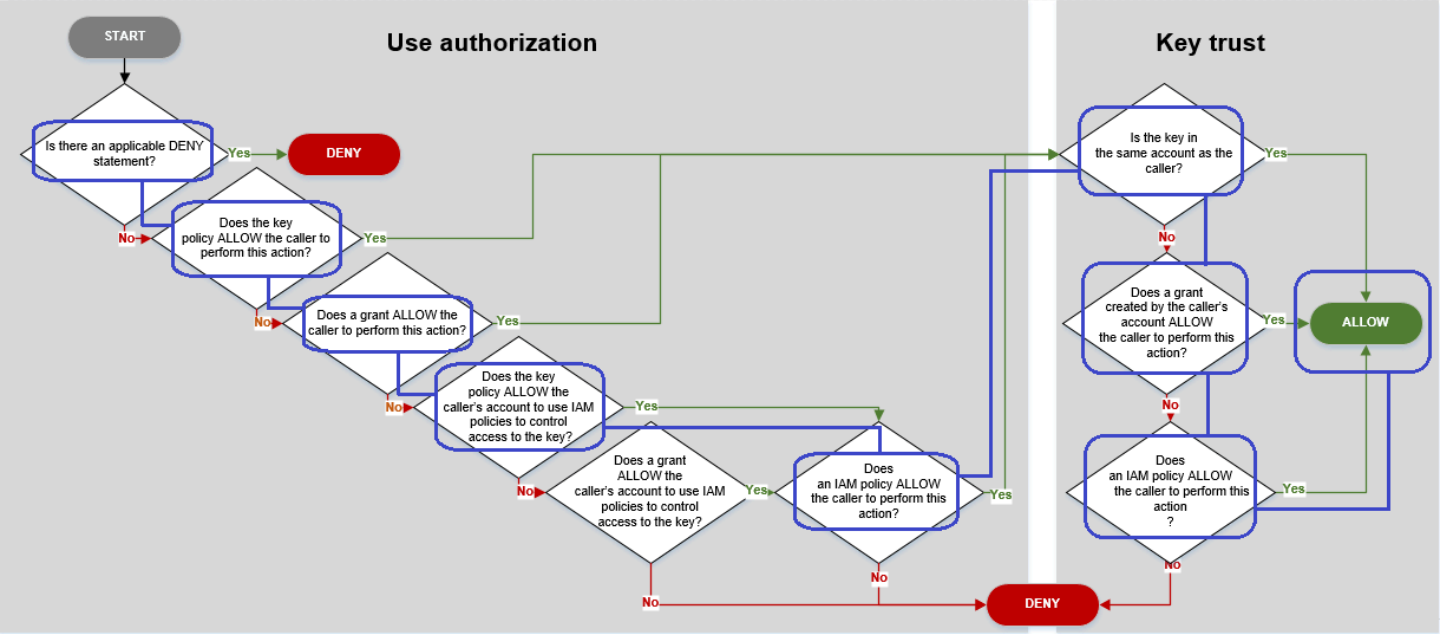
- 不過，沒有可賦予 Alice 使用 KMS 金鑰之許可的金鑰政策、IAM 政策或授予。因此，Alice 使用 KMS 金鑰的許可遭拒。

範例 2：使用者採用的角色具有在不同 AWS 帳戶 中使用 KMS 金鑰的許可

Bob 是帳戶 1 (111122223333) 的使用者。他獲准在密碼編譯操作中使用帳戶 2 (444455556666) 中的 KMS 金鑰。如何實現？

> **Tip**
>
> 評估跨帳戶許可時，請記住，金鑰政策已於 KMS 金鑰的帳戶中指定。在呼叫者帳戶中指定 IAM 政策，即使呼叫者位於不同帳戶中。如需提供跨帳戶存取 KMS 金鑰的詳細資訊，請參閱 允許其他帳戶中的使用者使用 KMS 金鑰。

- 帳戶 2 中 KMS 金鑰的金鑰政策允許帳戶 2 使用 IAM 政策來控制對 KMS 金鑰的存取。
- 帳戶 2 中 KMS 金鑰的金鑰政策允許帳戶 1 在密碼編譯操作中使用 KMS 金鑰。不過，帳戶 1 必須使用 IAM 政策賦予其主體存取 KMS 金鑰的權限。
- 帳戶 1 中的 IAM 政策允許 `Engineering` 角色使用帳戶 2 的 KMS 金鑰進行密碼編譯操作。
- Bob (帳戶 1 的使用者) 具有採用 `Engineering` 角色的許可。
- Bob 可以信任此 KMS 金鑰，因為即使它不在他的帳戶中，他帳戶中的 IAM 政策可為他提供此 KMS 金鑰的明確使用許可。



讓我們看看可讓 Bob (帳戶 1 的使用者) 使用帳戶 2 中 KMS 金鑰的政策。

- KMS 金鑰的金鑰政策允許帳戶 2 (444455556666，擁有 KMS 金鑰的帳戶) 使用 IAM 政策來控制對 KMS 金鑰的存取。此金鑰政策也允許帳戶 1 (111122223333) 在密碼編譯操作中使用 KMS 金鑰 (在政策陳述式的 `Action` 元素中指定)。不過，在帳戶 1 定義可賦予主體存取 KMS 金鑰的 IAM 政策前，帳戶 1 中沒有人可以使用帳戶 2 的 KMS 金鑰。

  在流程圖中，帳戶 2 的此金鑰政策符合金鑰政策是否允許呼叫者帳戶使用 IAM 政策來控制對金鑰的存取？條件。

```
{
    "Id": "key-policy-acct-2",
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Permission to use IAM policies",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::444455556666:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Sid": "Allow account 1 to use this KMS key",
            "Effect": "Allow",
```

```
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:ReEncryptFrom",
                "kms:ReEncryptTo",
                "kms:GenerateDataKey",
                "kms:GenerateDataKeyWithoutPlaintext",
                "kms:DescribeKey"
            ],
            "Resource": "*"
        }
    ]
}
```

- 呼叫者的 AWS 帳戶 (帳戶 1，111122223333) 的 IAM 政策賦予主體使用帳戶 2 (444455556666) 中的 KMS 金鑰執行密碼編譯操作的許可。Action 元素為主體指派了與帳戶 2 中金鑰政策提供給帳戶 1 相同的許可。若要將這些許可提供給帳戶 1 中的 Engineering 角色，此內嵌政策會內嵌於 Engineering 角色。

  只有在帳戶 2 中 KMS 金鑰的金鑰政策賦予帳戶 1 使用 KMS 金鑰的許可時，這類的跨帳戶 IAM 政策才有效。此外，帳戶 1 賦予其主體執行動作的許可僅限金鑰政策賦予該帳戶的許可。

  在流程圖中，這可符合 IAM 政策是否允許呼叫者執行此動作？條件。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "kms:Encrypt",
                "kms:Decrypt",
                "kms:ReEncryptFrom",
                "kms:ReEncryptTo",
                "kms:GenerateDataKey",
                "kms:GenerateDataKeyWithoutPlaintext",
                "kms:DescribeKey"
            ],
            "Resource": [
```

```
                "arn:aws:kms:us-
west-2:444455556666:key/1234abcd-12ab-34cd-56ef-1234567890ab"
            ]
        }
    ]
}
```

- 最後一個必要元素是帳戶 1 中 Engineering 角色的定義。該角色的 AssumeRolePolicyDocument 可讓 Bob採用 Engineering 角色。

```
{
    "Role": {
        "Arn": "arn:aws:iam::111122223333:role/Engineering",
        "CreateDate": "2019-05-16T00:09:25Z",
        "AssumeRolePolicyDocument": {
            "Version": "2012-10-17",
            "Statement": {
                "Principal": {
                    "AWS": "arn:aws:iam::111122223333:user/bob"
                },
                "Effect": "Allow",
                "Action": "sts:AssumeRole"
            }
        },
        "Path": "/",
        "RoleName": "Engineering",
        "RoleId": "AROA4KJY2TU23Y7NK62MV"
    }
}
```

AWS KMS 權限

此表格旨在協助您瞭解 AWS KMS 權限，以便您控制對 AWS KMS 資源的存取。欄標題的定義顯示在表格下方。

您也可以在「服務授 AWS KMS 權參考」 AWS Key Management Service主題的 「動作」、「資源」和「條件索引鍵」中瞭解權限。然而，該主題不會列出您可以用於細化每個許可的所有條件金鑰。

> **Note**
>
> 您可能需要水平或垂直捲動，才能查看資料表中的所有資料。

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
CancelKeyDeletion `kms:CancelKeyDeletion`	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService
ConnectCustomKeyStore `kms:ConnectCustomKeyStore`	IAM 政策	否	*	公里：CallerAccount
CreateAlias `kms:CreateAlias`	IAM 政策 (適用於別名)	否	別名	無 (控制對別名的存取時)

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
若要使用此操作時，呼叫者需要兩個資源上的 `kms:CreateAlias` 許可： • 別名 (在 IAM 政策中) • KMS 金鑰 (在金鑰政策中) 如需詳細資訊，請參閱 控制對別名的存取。	KMS 政策 (適用於 KMS 金鑰)	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService
CreateCustomKeyStore `kms:CreateCustomKeyStore`	IAM 政策	否	*	公里：CallerAccount

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
CreateGrant kms:CreateGrant	金鑰政策	是	KMS 金鑰	加密內容條件： 公里:EncryptionContext:上下文鍵 公里：EncryptionContextKeys 授予條件： 公里：GrantConstraintType 公里：GranteePrincipal 公里：GrantIsForAWSResource 公里：GrantOperations 公里：RetiringPrincipal KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
				aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService
CreateKey kms:CreateKey	IAM 政策	否	*	公里：BypassPolicyLockoutSafetyCheck 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ViaService aws：RequestTag/標籤鍵（AWS 全局條件鍵） aws：ResourceTag/標籤鍵（AWS 全局條件鍵） aws：TagKeys（AWS 全局條件鍵）

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
解密 kms:Decrypt	金鑰政策	是	KMS 金鑰	密碼編譯操作的條件 公里：EncryptionAlgorithm 公里：RequestAlias 加密內容條件： 公里:EncryptionContext:上下文鍵 公里：EncryptionContextKeys KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
DeleteAlias kms:DeleteAlias	IAM 政策 (適用於別名)	否	別名	無 (控制對別名的存取時)
若要使用此操作時，呼叫者需要兩個資源上的 kms:DeleteAlias 許可： • 別名 (在 IAM 政策中) • KMS 金鑰 (在金鑰政策中) 如需詳細資訊，請參閱 控制對別名的存取。	KMS 政策 (適用於 KMS 金鑰)	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵 (AWS 全局條件鍵) 公里：ViaService
DeleteCustomKeyStore kms:DeleteCustomKeyStore	IAM 政策	否	*	公里：CallerAccount

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
DeleteImportedKeyMaterial kms:DeleteImportedKeyMaterial	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService
DescribeCustomKeyStores kms:DescribeCustomKeyStores	IAM 政策	否	*	公里：CallerAccount

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
DescribeKey kms:DescribeKey	金鑰政策	是	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService 其他條件： 公里：RequestAlias

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
DisableKey kms:DisableKey	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
DisableKeyRotation kms:DisableKeyRotation	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService
DisconnectCustomKeyStore kms:DisconnectCustomKeyStore	IAM 政策	否	*	公里：CallerAccount

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
EnableKey kms:EnableKey	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
EnableKeyRotation kms:EnableKeyRotation	金鑰政策	否	KMS 金鑰 (僅對稱)	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
加密 kms:Encrypt	金鑰政策	是	KMS 金鑰	密碼編譯操作的條件 公里：EncryptionAlgorithm 公里：RequestAlias 加密內容條件： 公里:EncryptionContext:上下文鍵 公里：EncryptionContextKeys KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
GenerateDataKey kms:GenerateDataKey	金鑰政策	是	KMS 金鑰 (僅對稱)	密碼編譯操作的條件 公里：EncryptionAlgorithm 公里：RequestAlias 加密內容條件： 公里:EncryptionContext:上下文鍵 公里：EncryptionContextKeys KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
GenerateDataKeyPair kms:GenerateDataKeyPair	金鑰政策	是	KMS 金鑰 (僅對稱) 產生受對稱加密 KMS 金鑰保護的非對稱資料金鑰對。	資料金鑰對的條件： 公里：DataKeyPairSpec 密碼編譯操作的條件 公里：EncryptionAlgorithm 公里：RequestAlias 加密內容條件： 公里:EncryptionContext:上下文鍵 公里：EncryptionContextKeys KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵）

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
				公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
GenerateDataKeyPairWithoutPlaintext kms:GenerateDataKeyPairWithoutPlaintext	金鑰政策	是	KMS 金鑰 (僅對稱) 產生受對稱加密 KMS 金鑰保護的非對稱資料金鑰對。	資料金鑰對的條件： 公里：DataKeyPairSpec 密碼編譯操作的條件 公里：EncryptionAlgorithm 公里：RequestAlias 加密內容條件： 公里:EncryptionContext:上下文鍵 公里：EncryptionContextKeys KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵）

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
				公里：ViaService
GenerateDataKeyWithoutPlaintext `kms:GenerateDataKeyWithoutPlaintext`	金鑰政策	是	KMS 金鑰 (僅對稱)	密碼編譯操作的條件 公里：EncryptionAlgorithm 公里：RequestAlias 加密內容條件： 公里:EncryptionContext:上下文鍵 公里：EncryptionContextKeys KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
GenerateMac kms:GenerateMac	金鑰政策	是	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService 密碼編譯操作的條件： 公里：MacAlgorithm 公里：RequestAlias
GenerateRandom kms:GenerateRandom	IAM 政策	N/A	*	無

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
GetKeyPolicy kms:GetKeyPolicy	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
GetKeyRotationStatus kms:GetKeyRotationStatus	金鑰政策	是	KMS 金鑰 (僅對稱)	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
GetParametersForImport kms:GetParametersForImport	金鑰政策	否	KMS 金鑰	公里：WrappingAlgorithm 公里：WrappingKeySpec KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
GetPublicKey kms:GetPublicKey	金鑰政策	是	KMS 金鑰 (僅非對稱)	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService 其他條件： 公里：RequestAlias

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
ImportKeyMaterial kms:ImportKeyMaterial	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService 其他條件： 公里：ExpirationModel 公里：ValidTo
ListAliases kms:ListAliases	IAM 政策	否	*	無

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
ListGrants kms:ListGrants	金鑰政策	是	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService 其他條件： 公里：GrantIsForAWSResource

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
ListKeyPolicies kms:ListKeyPolicies	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService
ListKeys kms:ListKeys	IAM 政策	否	*	無

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
ListResourceTags kms:ListResourceTags	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService
ListRetirableGrants kms:ListRetirableGrants	IAM 政策	指定的主體必須位於本機帳戶中，但操作會在所有帳戶中傳回授予。	*	無

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
PutKeyPolicy kms:PutKeyPolicy	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService 其他條件： 公里：BypassPolicyLockoutSafetyCheck

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
ReEncrypt kms:ReEncryptFrom kms:ReEncryptTo 若要使用此操作時，呼叫者需要兩個 KMS 金鑰上的許可： • KMS 金鑰上的 kms:ReEncryptFrom ，用於解密 • KMS 金鑰上的 kms:ReEncryptTo ，用於加密	金鑰政策	是	KMS 金鑰	密碼編譯操作的條件 公里：EncryptionAlgorithm 公里：RequestAlias 加密內容條件： 公里:EncryptionContext:上下文鍵 公里：EncryptionContextKeys KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService 其他條件：

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
				公里：ReEncryptOnSameKey
ReplicateKey kms:ReplicateKey 若要使用此操作時，呼叫者需要以下許可： • 多區域主要金鑰上的 kms:ReplicateKey • 複本區域中 IAM 政策的 kms:CreateKey	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService 其他條件： 公里：ReplicaRegion

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
RetireGrant kms:RetireGrant 淘汰授予的許可主要取決於授予。單獨的政策無法允許存取此操作。如需詳細資訊，請參閱 淘汰和撤銷授予。	IAM 政策 (此許可在主要政策中無效。)	是	KMS 金鑰	公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵）
RevokeGrant kms:RevokeGrant	金鑰政策	是	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService 其他條件： 公里：GrantIsForAWSResource

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
ScheduleKeyDeletion kms:ScheduleKeyDeletion	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
符號 kms:Sign	金鑰政策	是	KMS 金鑰 (僅非對稱)	簽署和驗證的條件： 公里：MessageType 公里：RequestAlias 公里：SigningAlgorithm KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
TagResource kms:TagResource	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService 標記的條件： aws：RequestTag/標籤鍵（AWS 全局條件鍵） aws：TagKeys（AWS 全局條件鍵）

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
UntagResource kms:UntagResource	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService 標記的條件： aws：RequestTag/標籤鍵（AWS 全局條件鍵） aws：TagKeys（AWS 全局條件鍵）

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
UpdateAlias kms:UpdateAlias	IAM 政策 (適用於別名)	否	別名	無 (控制對別名的存取時)
若要使用此操作時，呼叫者需要三個資源上的 kms:UpdateAlias 許可： • 別名 • 目前關聯的 KMS 金鑰 • 新關聯的 KMS 金鑰 如需詳細資訊，請參閱 控制對別名的存取。	金鑰政策 (適用於 KMS 金鑰)	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService
UpdateCustomKeyStore kms:UpdateCustomKeyStore	IAM 政策	否	*	公里：CallerAccount

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
UpdateKeyDescription `kms:UpdateKeyDescription`	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
UpdatePrimaryRegion kms:UpdatePrimaryRegion 若要使用此操作時，呼叫者需要針對會成為複本金鑰之多區域主要金鑰和會成為主要金鑰之多區域複本金鑰的 kms:UpdatePrimaryRegion 許可。	金鑰政策	否	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService 其他條件 公里：PrimaryRegion

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
確認 kms:Verify	金鑰政策	是	KMS 金鑰 (僅非對稱)	簽署和驗證的條件： 公里：MessageType 公里：RequestAlias 公里：SigningAlgorithm KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService

動作與許可	Policy type (政策類型)	跨帳戶使用	資源 (適用於 IAM 政策)	AWS KMS 條件鍵
VerifyMac kms:VerifyMac	金鑰政策	是	KMS 金鑰	KMS 金鑰操作的條件： 公里：CallerAccount 公里：KeySpec 公里：KeyUsage 公里：KeyOrigin 公里：MultiRegion 公里：MultiRegionKeyType 公里：ResourceAliases aws：ResourceTag/標籤鍵（AWS 全局條件鍵） 公里：ViaService 密碼編譯操作的條件： 公里：MacAlgorithm 公里：RequestAlias

資料欄描述

此資料表中的資料欄提供下列資訊：

- 動作和權限會列出每個 AWS KMS API 作業以及允許該作業的權限。您可以在政策陳述式的 `Action` 元素中指定操作。
- 政策類型指出許可可用於金鑰政策還是 IAM 政策。

金鑰政策表示您可以在金鑰政策中指定許可。當金鑰政策包含啟用 IAM 政策的政策陳述式時，您可以在 IAM 政策中指定許可。

IAM 政策表示您只能在 IAM 政策中指定許可。

- 跨帳戶使用顯示授權使用者可以對不同 AWS 帳戶中資源執行的操作。

  值為是表示主體可以對不同 AWS 帳戶中的資源執行操作。

  值為否表示主體僅可對其自己 AWS 帳戶中的資源執行操作。

  如果您為不同帳戶中的主體授予無法在跨帳戶資源上使用的許可，則許可無效。例如，如果您在不同帳戶中授予主體 kms: 您帳戶中 KMS 金鑰的TagResource權限，則他們嘗試在您帳戶中標記 KMS 金鑰的嘗試將會失敗。

- 資 AWS KMS 源會列出權限套用的資源。 AWS KMS 支援兩種資源類型：KMS 金鑰和別名。在金鑰政策中，`Resource` 元素的值永遠是 `*`，這指的是 KMS 金鑰所連接的金鑰政策。

  使用下列值代表 IAM 政策中的 AWS KMS 資源。

  KMS 金鑰

  當資源為 KMS 金鑰時，請使用其金鑰 ARN。如需協助，請參閱 the section called “尋找金鑰 ID 和金鑰 ARN”。

  `arn:`*`AWS_partition_name`*`:kms:`*`AWS_Region`*`:`*`AWS_account_ID`*`:key/`*`key_ID`*

  例如：

  arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab

  別名

  當資源為別名時，請使用其別名 ARN。如需協助，請參閱 the section called “尋找別名和別名 ARN”。

  `arn:`*`AWS_partition_name`*`:kms:`*`AWS_region`*`:`*`AWS_account_ID`*`:alias/`*`alias_name`*

  例如：

  爪子:爪子:公里美國西部-2:11112222333: 別名/ExampleAlias

*** (星號)**

當許可不適用於特定資源 (KMS 金鑰或別名) 時，請使用星號 (*)。

在 AWS KMS 權限的 IAM 政策中，`Resource`元素中的星號表示所有 AWS KMS 資源 (KMS 金鑰和別名)。當 AWS KMS 權限不適用於任何特定 KMS 金鑰或別名時，您也可以在`Resource`元素中使用星號。例如，允許或拒絕 `kms:CreateKey` 或 `kms:ListKeys` 許可時，您可以將 `Resource` 元素設定為 `*` 或設定為帳戶特定的變數，例如 `arn:`*`AWS_partition_name`*`:kms:`*`AWS_region`*`:`*`AWS_account_ID`*`:*`。

- AWS KMS 條件索引鍵會列出您可用來控制對作業存取的 AWS KMS 條件索引鍵。您可以在政策的 `Condition` 元素中指定條件。如需詳細資訊，請參閱 AWS KMS 條件鍵。此資料行也包含AWS 全域條件索引鍵 AWS KMS，這些索引鍵受到所有服務支援 (但不是所有 AWS 服務)

測試您的許可

若要使用 AWS KMS，您必須擁有憑證以便 AWS 可用以驗證您的請求。憑證必須包含存取 KMS 金鑰與別名的許可。許可由金鑰政策、IAM 政策、授權以及跨帳戶存取控制決定。除控制 KMS 金鑰的存取權之外，您還可控制 CloudHSM 與自訂金鑰存放區的存取權。

您可指定 `DryRun` API 參數，驗證您擁有運用 AWS KMS 金鑰的必要許可。您也可利用 `DryRun` 來驗證 AWS KMS API 呼叫的請求參數是否已正確指定。

主題

- 什麼是 DryRun 參數？
- DryRun 使用 API 指定

什麼是 DryRun 參數？

`DryRun` 是選用的 API 參數，您可加以指定以便驗證 AWS KMS API 呼叫是否成功。在實際呼叫 AWS KMS 之前，利用 `DryRun` 來測試 API 呼叫。您可以驗證下列各項。

- 您擁有必要的許可，可運用 AWS KMS 金鑰。
- 您已正確指定呼叫的參數。

AWS KMS 支援在特定 API 動作採用 `DryRun` 參數：

- CreateGrant

- 解密
- 加密
- GenerateDataKey
- GenerateDataKeyPair
- GenerateDataKeyPairWithoutPlaintext
- GenerateDataKeyWithoutPlaintext
- GenerateMac
- ReEncrypt
- RetireGrant
- RevokeGrant
- 符號
- 確認
- VerifyMac

採用 DryRun 參數將產生費用，並作為標準 API 請求計費。如需關於 AWS KMS 定價的詳細資訊，請參閱 AWS Key Management Service 定價。

採用 DryRun 參數的所有 API 請求都會套用至 API 的請求配額，如您超出 API 請求配額，則可能導致限流例外狀況。例如，無論採用 DryRun 還是不採用 DryRun 來呼叫 Decrypt，都會根據相同的密碼編譯操作配額進行計數。如需進一步了解，請參閱《調節 AWS KMS 請求》。

系統會將每次對 AWS KMS API 操作的呼叫擷取並記錄為 AWS CloudTrail 日誌的事件。任何指定DryRun參數之作業的輸出都會顯示在 CloudTrail 記錄檔中。如需詳細資訊，請參閱 使用 AWS CloudTrail 記錄 AWS KMS API 呼叫。

DryRun 使用 API 指定

若要採用 DryRun，請在支援 –dry-run 參數的 AWS CLI 命令與 AWS KMS API 呼叫指定該參數。在這樣做之後，AWS KMS 將驗證您的呼叫是否會成功。採用 DryRun 的 AWS KMS 呼叫一律會失敗，並傳回訊息以提供資訊說明呼叫失敗的原因。訊息可能包含下列例外狀況：

- DryRunOperationException - 如未指定 DryRun，請求就會成功。
- ValidationException - 請求因指定不正確 API 參數而失敗。
- AccessDeniedException - 您不具權限，無法對 KMS 資源執行指定 API 動作。

例如，下列命令會使用CreateGrant作業並建立授權，允許有權承擔keyUserRole角色的使用者在指定的對稱 KMS 金鑰上呼叫「解密」作業。指定 DryRun 參數。

```
$  aws kms create-grant \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --grantee-principal arn:aws:iam::111122223333:role/keyUserRole \
    --operations Decrypt \
    --dry-run
```

特殊用途金鑰

AWS Key Management Service (AWS KMS) 支援多種不同類型的金鑰，以供不同用途使用。

建立 AWS KMS key 時，根據預設，您會取得對稱加密的 KMS 金鑰。在 AWS KMS 中，一個對稱加密 KMS 金鑰代表用來加密和解密的 256 位元 AES-GCM 加密金鑰，但中國區域除外，在該處代表一個使用 SM4 加密金鑰的 128 位元對稱金鑰。對稱金鑰資料絕不會讓 AWS KMS 出現未加密的情況。除非您的任務明確要求非對稱加密或 HMAC 金鑰，否則絕不會讓 AWS KMS 出現未加密情況的對稱 KMS 金鑰是很好的選擇。另外，與 AWS KMS 整合的 AWS 服務僅會使用對稱加密 KMS 金鑰來加密您的資料。這些服務不支援使用非對稱 KMS 金鑰進行加密。

您可以在 AWS KMS 中使用對稱加密 KMS 金鑰來加密、解密和重新加密資料、產生資料金鑰和資料金鑰對，以及產生隨機位元組字串。您可以將自己的金鑰資料匯入對稱加密 KMS 金鑰，並在自訂金鑰存放區中建立對稱加密 KMS 金鑰。如需對稱和非對稱 KMS 金鑰執行之操作的比較表，請參閱金鑰類型參考。

AWS KMS 亦支援以下特殊用途 KMS 金鑰類型：

- 適用於公有金鑰密碼編譯的非對稱 RSA 金鑰
- 適用於簽署和驗證的非對稱 RSA 和 ECC 金鑰
- 非對稱 SM2 金鑰 (僅限中國區域)，用於公有金鑰加密或簽署和驗證
- 產生並驗證雜湊訊息驗證碼的 HMAC 金鑰
- 運作方式類似不同 AWS 區域 中相同金鑰之複本的多區域金鑰 (對稱和非對稱)
- 具有匯入金鑰材料的金鑰，材料由您提供
- 由 AWS CloudHSM 叢集或 AWS 之外的外部金鑰管理器支援的自訂金鑰存放區中的金鑰。

選擇一個 KMS 金鑰類型

AWS KMS 可支援多種類型的 KMS 金鑰：對稱加密金鑰、對稱 HMAC 金鑰、非對稱加密金鑰和非對稱簽署金鑰。

KMS 金鑰內含不同的密碼編譯金鑰資料，因此各不相同。

- 對稱加密 KMS 金鑰：代表一個 256 位元 AES-GCM 加密金鑰，但中國區域除外，在該處代表 128 位元 SM4 加密金鑰。對稱金鑰資料絕不會讓 AWS KMS 出現未加密的情況。若要使用對稱加密 KMS 金鑰，必須呼叫 AWS KMS。

對稱加密金鑰 (此為預設 KMS 金鑰) 是多數用途的理想選擇。若您需要 KMS 金鑰來保護 AWS 服務中的資料，除非您收到指示需使用其他類型的金鑰，否則請使用對稱加密金鑰。

- 非對稱 KMS 金鑰：代表數學相關的公有金鑰和私有金鑰對，可用於加密和解密或簽署和驗證，但不能同時用於兩者。私有金鑰絕不會讓 AWS KMS 出現未加密的狀況。您可以呼叫 AWS KMS API 操作以使用 AWS KMS 內的公有金鑰，或者下載公有金鑰，在 AWS KMS 外面使用。
- HMAC KMS 金鑰 (對稱)：是指長度可變的對稱金鑰，用於產生和驗證雜湊訊息驗證碼。HMAC KMS 金鑰中的金鑰資料絕不會讓 AWS KMS 出現未加密的情況。若要使用 HMAC KMS 金鑰，必須呼叫 AWS KMS。

您建立的 KMS 金鑰類型，在很大程度上取決於您計劃如何使用 KMS 金鑰、安全需求，以及授權需求。建立 KMS 金鑰時，請記住，KMS 金鑰的加密編譯組態，包括其金鑰規格和金鑰用途，都是在您建立 KMS 金鑰時所建立，且無法變更。

根據您的使用案例，使用下列指導判斷您需要的 KMS 金鑰類型。

加密和解密資料

請使用對稱 KMS 金鑰處理對於大多數需要加密和解密資料的使用案例。AWS KMS 使用的對稱加密演算法快速又有效率，並可確保資料的機密性和真實性。其支援經定義為加密內容之額外的驗證資料 (AAD) 所驗證的加密。這種類型的 KMS 金鑰要求加密資料的寄件人和收件人都具有呼叫 AWS KMS 的有效 AWS 認證。

如果您的使用案例要求無法呼叫 AWS KMS 的使用者在 AWS 外部加密，非對稱 KMS 金鑰是很好的選擇。您可以分配非對稱 KMS 金鑰的公有金鑰，讓這些使用者加密資料。而需要解密該資料的應用程式，則可以使用 AWS KMS 內的非對稱 KMS 金鑰私有金鑰。

簽署訊息及驗證簽章

若要簽署訊息及驗證簽章，您必須使用非對稱 KMS 金鑰。您可以使用 KMS 金鑰搭配代表 RSA 金鑰對或橢圓曲線 (ECC) 金鑰對，或 SM2 金鑰對 (僅限中國區域) 的金鑰規格。您選擇的金鑰規格取決於您要使用的簽署演算法。ECC 金鑰對支援的 ECDSA 簽章演算法優於 RSA 簽章演算法。但是，您可能需要使用特定的金鑰規格和簽章演算法，以支援在 AWS 以外驗證簽章的使用者。

執行公有金鑰加密

若要執行公有金鑰加密，您必須使用非對稱 KMS 金鑰搭配 RSA 金鑰規格或 SM2 金鑰規格 (僅限中國區域) 。若要使用 KMS 金鑰的公有金鑰加密 AWS KMS 的資料，請使用 Encrypt 操作。您也可以下載公有金鑰，與 AWS KMS 外部需要加密資料的對象共用此金鑰。

當您下載非對稱 KMS 金鑰的公有金鑰時，您可以在 AWS KMS 之外使用。但此金鑰將不再受保護 AWS KMS 中 KMS 金鑰的安全控制項管控。例如，您無法使用 AWS KMS 金鑰政策，或授權控制公有金鑰的使用。也無法使用 AWS KMS 支援的加密演算法控制金鑰是否僅用於加密和解密。如需詳細資訊，請參閱下載公有金鑰的特殊考量。

若要解密以 AWS KMS 之外的公開金鑰加密的資料，請呼叫 Decrypt 操作。如果使用 KMS 金鑰的公有金鑰搭配 `SIGN_VERIFY` 的金鑰用途加密資料，則 `Decrypt` 操作會失敗。如果使用 AWS KMS 不支援您所選金鑰規格的演算法進行加密，也會失敗。如需主要規格和支援演算法的詳細資訊，請參閱《非對稱金鑰規格》。

為避免發生這些錯誤，使用 AWS KMS 外公有金鑰的所有人都必須儲存金鑰組態。主AWS KMS控台和GetPublicKey回應會提供您共用公開金鑰時必須包含的資訊。

產生和驗證 HMAC 代碼

若要產生和驗證雜湊訊息驗證碼，請使用 HMAC KMS 金鑰。在 AWS KMS 中建立 HMAC 金鑰時，AWS KMS 會建立和保護您的金鑰資料，並確保您為金鑰使用正確的 MAC 演算法。HMAC 代碼也可以用來作為虛擬亂數，並在特定情況下用於對稱簽署和字符化。

HMAC KMS 金鑰為對稱金鑰。在 AWS KMS 主控台中建立 HMAC KMS 金鑰時，請選擇 `Symmetric` 金鑰類型。

與 AWS 服務搭配使用

若要建立 KMS 金鑰以搭配使用與 AWS KMS 整合的 AWS 服務，請參閱服務文件。加密資料的 AWS 服務需要對稱加密 KMS 金鑰。。

除了這些考量外，具有不同金鑰規格的 KMS 金鑰的密碼編譯操作也有不同的定價和不同的請求配額。如需 AWS KMS 定價的資訊，請參閱 AWS Key Management Service 定價。如需請求配額的詳細資訊，請參閱 請求配額。

選取金鑰用途

KMS 金鑰的金鑰用途決定 KMS 金鑰是用於加密和解密，還是簽署和驗證簽名，或是產生和驗證 HMAC 標籤。每個 KMS 金鑰僅有一個金鑰用途。使用 KMS 金鑰執行多種類型的操作，會使得所有操作的產品更容易受到攻擊。

如下表所示，對稱加密 KMS 金鑰僅能用於加密和解密。HMAC KMS 金鑰僅能用於產生和驗證 HMAC 代碼。橢圓曲線 (ECC) KMS 金鑰只能用於簽署和驗證。您僅需要為 RSA KMS 金鑰決定金鑰用途。

KMS 金鑰類型的有效金鑰用途

KMS 金鑰類型	加密和解密 ENCRYPT_DECRYPT	簽署和驗證 SIGN_VERIFY	產生和驗證 MAC GENERATE_VERIFY_MAC
對稱加密 KMS 金鑰	✓	⊗	⊗
HMAC KMS 金鑰 (對稱)	⊗	⊗	✓
具有 RSA 金鑰對的非對稱 KMS 金鑰	✓	✓	⊗
具有 ECC 金鑰對的非對稱 KMS 金鑰	⊗	✓	⊗
非對稱 KMS 金鑰搭配 SM2 金鑰對 (僅限中國區域)	✓	✓	⊗

在 AWS KMS 主控台中，先選擇金鑰類型 (對稱或非對稱)，再選擇金鑰用途。您選擇的金鑰類型會決定系統顯示的金鑰用途。您選擇的金鑰用途會決定系統顯示的金鑰規格 (如有)。

在 AWS KMS 主控台中選擇金鑰用途：

- 如果是對稱加密 KMS 金鑰 (預設)，請選擇 Encrypt and decrypt (加密和解密)。
- 如果是 HMAC KMS 金鑰，請選擇 Generate and verify MAC (產生和驗證 MAC)。
- 如果是含有橢圓曲線 (ECC) 金鑰資料的非對稱 KMS 金鑰，請選擇 Sign and verify (簽署和驗證)。
- 如果是含有 RSA 金鑰資料的非對稱 KMS 金鑰，請選擇 Encrypt and decrypt (加密和解密) 或 Sign and verify (簽署和驗證)。
- 如果是搭配 SM2 金鑰材料的非對稱 KMS 金鑰，請選擇 Encrypt and decrypt (加密和解密) 或 Sign and verify (簽署和驗證)。SM2 金鑰規格僅在中國區域提供。

若要允許主體僅針對特定金鑰用途建立 KMS 金鑰，請使用 kms: KeyUsage 條件金鑰。您也可以使用 kms:KeyUsage 條件索引鍵來允許委託人根據金鑰使用情形呼叫 KMS 金鑰的 API 操作。例如，您可以允許只有在金鑰使用情形為 SIGN_VERIFY 時，才能停用 KMS 金鑰的許可。

選取金鑰規格

在建立非對稱 KMS 金鑰或 HMAC KMS 金鑰時，需選取其 key spec (金鑰規格)。金鑰規格是每個 AWS KMS key 的屬性，代表 KMS 金鑰的密碼編譯組態。您可以在建立 KMS 金鑰時選擇金鑰規格，之後便無法進行變更。如果選取了錯誤的金鑰規格，請刪除 KMS 金鑰，再建立新的金鑰規格。

> **Note**
>
> KMS 金鑰的金鑰規格稱為「客戶主要金鑰規格」。該CreateKey操作的CustomerMasterKeySpec參數已被棄用。請改用 KeySpec 參數。CreateKey和DescribeKey作業的回應包括具有相同值的KeySpec和CustomerMasterKeySpec成員。

金鑰規格會決定 KMS 金鑰為對稱或非對稱、KMS 金鑰中的金鑰資料類型，以及是加密演算法、簽署演算法還是訊息身分驗證代碼 (MAC) 演算法，這三種都是 AWS KMS 支援的 KMS 金鑰演算法。您選擇的金鑰規格通常會依您的使用案例和法規要求決定。不過，具有不同金鑰規格的 KMS 金鑰的密碼編譯操作定價也不同，而且有不同的配額。如需定價詳細資訊，請參閱 AWS Key Management Service 定價。如需請求配額的詳細資訊，請參閱 請求配額。

若要判斷您帳戶中的主體可用於 KMS 金鑰的金鑰規格，請使用 kms: KeySpec 條件金鑰。

AWS KMS 支援下列 KMS 金鑰的金鑰規格：

對稱加密金鑰規格 (預設)

- SYMMETRIC_DEFAULT

HMAC 金鑰規格

- HMAC_224
- HMAC_256
- HMAC_384
- HMAC_512

RSA 金鑰規格 (加密和解密或簽署和驗證)

- RSA_2048

- RSA_3072
- RSA_4096

[橢圓曲線金鑰規格]

- 非對稱 NIST 建議的[橢圓曲線金鑰對] (簽署和驗證)
  - ECC_NIST_P256 (secp256r1)
  - ECC_NIST_P384 (secp384r1)
  - ECC_NIST_P521 (secp521r1)
- 其他非對稱橢圓曲線金鑰對 (簽署和驗證)
  - ECC_SECG_P256K1 (secp256k1)，常用於加密貨幣。

[SM2 金鑰規格] (加密和解密或簽署和驗證)

- SM2 (僅限中國區域)

AWS KMS 中的非對稱金鑰

AWS KMS 支援代表數學上相關的 RSA、橢圓曲線 (ECC) 或 SM2 (僅限中國區域) 公有和私有金鑰對的非對稱 KMS 金鑰。這些金鑰對會在已通過 FIPS 140-2 密碼編譯模組驗證計劃驗證的 AWS KMS 硬體安全模組中產生，除了中國 (北京) 和中國 (寧夏) 區域。私有金鑰絕不會讓 AWS KMS HSM 出現未加密的狀況。您可以下載公有金鑰，在 AWS 以外分配和使用。您可以建立非對稱 KMS 金鑰以用於加密和解密操作，或用於簽署和驗證操作，但不能同時用於兩種操作。

您可以在自己的 AWS 帳戶 帳戶中建立及管理非對稱 KMS 金鑰，包括設定金鑰政策、IAM 政策和控制 KMS 金鑰存取權的授予，以及啟用和停用 KMS 金鑰、建立標籤和別名與刪除 KMS 金鑰。您可以在 AWS CloudTrail 日誌 中的 AWS 內稽核使用或管理非對稱 KMS 金鑰的所有操作。

AWS KMS 也提供非對稱資料金鑰對，旨在用於 AWS KMS 以外的用戶端密碼編譯。非對稱資料金鑰對的私有金鑰受 AWS KMS 的對稱加密 KMS 金鑰 保護。

本主題說明非對稱 KMS 金鑰的運作方式、與其他 KMS 金鑰的不同之處，以及如何決定採用何種 KMS 金鑰類型來保護資料。並且說明非對稱資料金鑰對的運作方式，以及在 AWS KMS 以外的使用方式。

區域

AWS KMS 支援的所有 AWS 區域 區域都支援非對稱 KMS 金鑰和非對稱資料金鑰對。

進一步了解

- 若要建立非對稱 KMS 金鑰，請參閱 建立非對稱 KMS 金鑰。若要建立對稱加密 KMS 金鑰，請參閱建立金鑰。
- 若要建立多區域非對稱 KMS 金鑰，請參閱建立多區域金鑰。
- 若要了解 KMS 金鑰是對稱還是非對稱，請參閱 識別非對稱 KMS 金鑰。
- 如需適用每種 KMS 金鑰類型的 AWS KMS API 操作比較表，請參閱 the section called “金鑰類型參考”。
- 若要控制對帳戶中主體可用於 KMS 金鑰和資料金鑰的金鑰規格、金鑰用途、加密演算法和簽署演算法的存取，請參閱 the section called “AWS KMS 條件鍵”。
- 若要了解適用不同 KMS 金鑰類型的請求配額，請參閱 the section called “請求配額”。
- 若要了解如何使用非對稱 KMS 金鑰簽署訊息並驗證簽章，請參閱 AWS 安全部落格中的使用 AWS KMS 新的非對稱金鑰功能進行數位簽署。

主題

非對稱 KMS 金鑰

您可以在 AWS KMS 中建立非對稱 KMS 金鑰。非對稱 KMS 金鑰表示以數學方式相關的公有金鑰和私有金鑰對。您可以將公有金鑰提供給任何人，即使不受信任的人也可以，但私有金鑰則一定要保管好。

在非對稱 KMS 金鑰中，私有金鑰是在 AWS KMS 中建立，且絕不會讓 AWS KMS 出現未加密狀況。若要使用私有金鑰，您必須呼叫 AWS KMS。您可以透過呼叫 AWS KMS API 操作，以在 AWS KMS 內使用公有金鑰。或者，您也可以下載公有金鑰，在 AWS KMS 以外使用。

如果您的使用案例要求無法呼叫 AWS KMS 的使用者在 AWS 外部加密，非對稱 KMS 金鑰是很好的選擇。不過，如果您要建立 KMS 金鑰來加密您在 AWS 服務中存放或管理的資料，則請使用對稱加密 KMS 金鑰。與 AWS KMS 整合的 AWS 服務僅會使用對稱加密 KMS 金鑰來加密您的資料。這些服務不支援使用非對稱 KMS 金鑰進行加密。

AWS KMS 支援三種類型的非對稱 KMS 金鑰。

- RSA KMS 金鑰：具有 RSA 金鑰對的 KMS 金鑰，可用於加密和解密或簽署和驗證 (但不能同時使用)。AWS KMS 支援數種金鑰長度，可滿足不同的安全需求。
- 橢圓曲線 (ECC) KMS 金鑰：具有橢圓曲線金鑰對的 KMS 金鑰對，可用於簽署和驗證。AWS KMS 支援數種常用的曲線。
- SM2 KMS 金鑰 (僅限中國區域)：具有 SM2 金鑰對的 KMS 金鑰，可用於加密和解密或簽署和驗證 (但不能同時使用)。

如需有關如何選擇非對稱金鑰組態的說明，請參閱選擇一個 KMS 金鑰類型。如需有關 AWS KMS 支援之 RSA KMS 金鑰的加密和簽署演算法技術詳細資訊，請參閱 RSA 金鑰規格。如需有關 AWS KMS 支援之 ECC KMS 金鑰的簽署演算法技術詳細資訊，請參閱橢圓曲線金鑰規格。如需有關 AWS KMS 支援之 SM2 KMS 金鑰 (僅限中國區域) 的加密和簽署演算法技術詳細資訊，請參閱 SM2 金鑰規格。

如需對稱和非對稱 KMS 金鑰執行之操作的比較表，請參閱比較對稱和非對稱 KMS 金鑰。如需判斷 KMS 金鑰是對稱還是不對稱的說明，請參閱 識別非對稱 KMS 金鑰。

區域

AWS KMS 支援的所有 AWS 區域 區域都支援非對稱 KMS 金鑰和非對稱資料金鑰對。

建立非對稱 KMS 金鑰

您可以使用 CreateKeyAPI 或使用AWS CloudFormation範本，在AWS KMS主控台中建立非對稱 KMS 金鑰。非對稱 KMS 金鑰代表可用於加密或簽署的公有和私有金鑰對。私有金鑰保留在 AWS KMS 範圍內。若要下載公有金鑰以供在外部使用 AWS KMS，請參閱 下載公開金鑰。

在建立 KMS 金鑰來加密在 AWS 服務中存放或管理的資料時，請使用對稱加密 KMS 金鑰。與 AWS KMS 整合的 AWS 服務不支援非對稱 KMS 金鑰。如需有關如何決定建立對稱還是非對稱 KMS 金鑰的說明，請參閱選擇一個 KMS 金鑰類型。

如需建立 KMS 金鑰所需之許可的詳細資訊，請參閱 建立 KMS 金鑰的許可。

主題

- 建立非對稱 KMS 金鑰 (主控台)
- 建立非對稱 KMS 金鑰 (AWS KMS API)

建立非對稱 KMS 金鑰 (主控台)

您可以使用 AWS Management Console 來建立非對稱 AWS KMS keys (KMS 金鑰)。每個非對稱 KMS 金鑰皆代表一對公有和私有金鑰對。

> **Important**
>
> 請勿在別名、說明或標籤包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 選擇建立金鑰。
5. 若要建立非對稱 KMS 金鑰，請在金鑰類型中選擇 Asymmetric (非對稱)。

   如需有關如何在 AWS KMS 主控台中建立對稱加密 KMS 金鑰的資訊，請參閱建立對稱加密 KMS 金鑰 (主控台)。
6. 若要建立用於公有金鑰加密的非對稱 KMS 金鑰，請在金鑰使用情形中選擇 Encrypt and decrypt (加密和解密)。或者，若要建立用於簽署訊息和驗證簽名的非對稱 KMS 金鑰，請在金鑰使用情形中選擇 Sign and verify (簽署和驗證)。

   如需選擇金鑰用途值的說明，請參閱選取金鑰用途。
7. 選取非對稱 KMS 金鑰的規格 (Key spec (金鑰規格))。

   通常，您選取的金鑰規格取決於法規、安全或業務需求。但也可能受需要加密或簽署之訊息的大小所左右。一般而言，較長的加密金鑰更能抵禦暴力破解攻擊。

   如需選擇金鑰規格的說明，請參閱選取金鑰規格。
8. 選擇下一步。
9. 輸入 KMS 金鑰的別名。別名名稱的開頭不可以是 **`aws/`**。**`aws/`** 字首由 Amazon Web Services 保留，以代表您帳戶中的 AWS 受管金鑰。

   別名是易記名稱，讓您可以使用該名稱來識別主控台和某些 AWS KMS API 中的 KMS 金鑰。我們建議您選擇別名來表示您計劃保護的資料類型，或您計劃搭配 KMS 金鑰一起使用的應用程式。

   在 AWS Management Console 中建立 KMS 金鑰時需要別名。使用CreateKey作業時無法指定別名，但可以使用主控台或CreateAlias作業為現有 KMS 金鑰建立別名。如需詳細資訊，請參閱 使用別名。
10. (選用) 輸入 KMS 金鑰的描述。

輸入描述來說明您計劃保護的資料類型，或您計劃搭配 KMS 金鑰一起使用的應用程式。

您可以立即新增描述或在任意時間更新，除非金鑰狀態為 `Pending Deletion` 或 `Pending Replica Deletion`。若要加入、變更或刪除現有客戶管理金鑰的描述，請編輯中的描述AWS Management Console或使用UpdateKeyDescription作業。

11. (選用) 輸入標籤索引鍵和選用標籤值。若要將其他標籤新增至 KMS 金鑰，請選擇 Add tag (新增標籤)。

    將標籤新增到 AWS 資源時，AWS 會產生成本配置報告，內含按標籤彙總的用量與成本。標籤也可以用來控制 KMS 金鑰的存取。如需標記 KMS 金鑰的詳細資訊，請參閱 標記金鑰 和 AWS KMS 的 ABAC。

12. 選擇下一步。
13. 選取可管理 KMS 金鑰的 IAM 使用者和角色。

> **Note**
>
> 此金鑰政策授予 AWS 帳戶 完全控制此 KMS 金鑰。它允許帳戶管理員使用 IAM 政策授予其他主體管理 KMS 金鑰的許可。如需詳細資訊，請參閱 the section called “預設金鑰政策”。
> IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

14. (選用) 為了防止選取的 IAM 使用者和角色刪除此 KMS 金鑰，請在頁面底部的 Key deletion (金鑰刪除) 區段中，清除 Allow key administrators to delete this key (允許金鑰管理員刪除此金鑰) 核取方塊。
15. 選擇下一步。
16. 選取可將 KMS 金鑰用於密碼編譯操作的 IAM 使用者和角色。

> **Note**
>
> 此金鑰政策授予 AWS 帳戶 完全控制此 KMS 金鑰。它允許帳戶管理員使用 IAM 政策授予其他主體在密碼編譯操作中使用 KMS 金鑰的許可。如需詳細資訊，請參閱 the section called “預設金鑰政策”。
> IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

17. (選用) 您可以允許其他 AWS 帳戶 將此 KMS 金鑰用於密碼編譯操作。若要這樣做，請在頁面底部的其他 AWS 帳戶 區段中，選擇新增另一個 AWS 帳戶，然後輸入外部帳戶的 AWS 帳戶 識別號碼。若要新增多個外部帳戶，請重複此步驟。

    > **Note**
    >
    > 若要允許外部帳戶中的主體使用 KMS 金鑰，外部帳戶的管理員必須建立 IAM 政策來提供這些許可。如需詳細資訊，請參閱 允許其他帳戶中的使用者使用 KMS 金鑰。

18. 選擇下一步。
19. 檢閱您選擇的金鑰設定。您仍然可以返回並變更所有設定。
20. 選擇 Finish (完成) 來建立 KMS 金鑰。

建立非對稱 KMS 金鑰 (AWS KMS API)

您可以使用該CreateKey操作來建立非對稱AWS KMS key。以下範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

當您建立非對稱 KMS 金鑰時，必須指定 KeySpec 參數，決定您建立的金鑰類型。此外，您也必須指定 ENCRYPT_DECRYPT 或 SIGN_VERIFY 的 KeyUsage 值。建立 KMS 金鑰之後即無法變更這些屬性。

此CreateKey作業不允許您指定別名，但您可以使用此CreateAlias作業為新 KMS 金鑰建立別名。

> **Important**
>
> 請勿在 Description 或 Tags 欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

下列範例會使用 CreateKey 操作建立專為公有金鑰加密設計之 4096 位元 RSA 金鑰的非對稱 KMS 金鑰。

```
$ aws kms create-key --key-spec RSA_4096 --key-usage ENCRYPT_DECRYPT
{
    "KeyMetadata": {
        "KeyState": "Enabled",
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "KeyManager": "CUSTOMER",
```

```
        "Description": "",
        "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "CreationDate": 1569973196.214,
        "MultiRegion": false,
        "KeySpec": "RSA_4096",
        "CustomerMasterKeySpec": "RSA_4096",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "EncryptionAlgorithms": [
            "RSAES_OAEP_SHA_1",
            "RSAES_OAEP_SHA_256"
        ],
        "AWSAccountId": "111122223333",
        "Origin": "AWS_KMS",
        "Enabled": true
    }
}
```

下列範例命令建立的非對稱 KMS 金鑰，代表用於簽署和驗證的一對 ECDSA 金鑰。您無法建立用於加密和解密的橢圓曲線金鑰對。

```
$ aws kms create-key --key-spec ECC_NIST_P521 --key-usage SIGN_VERIFY
{
    "KeyMetadata": {
        "KeyState": "Enabled",
        "KeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
        "CreationDate": 1570824817.837,
        "Origin": "AWS_KMS",
        "SigningAlgorithms": [
            "ECDSA_SHA_512"
        ],
        "Arn": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321",
        "AWSAccountId": "111122223333",
        "KeySpec": "ECC_NIST_P521",
        "CustomerMasterKeySpec": "ECC_NIST_P521",
        "KeyManager": "CUSTOMER",
        "Description": "",
        "Enabled": true,
        "MultiRegion": false,
        "KeyUsage": "SIGN_VERIFY"
    }
}
```

下載公開金鑰

您可以使用 AWS Management Console 或 AWS KMS API 檢視、複製和下載非對稱 KMS 金鑰對中的公有金鑰。您必須擁有非對稱 KMS 金鑰的 `kms:GetPublicKey` 許可。

每個非對稱 KMS 金鑰對都是由永遠不會在未加密狀態下離開 AWS KMS 的私有金鑰，以及您可以下載和共享的公有金鑰組成。

您可以共享公有金鑰，讓其他人員在 AWS KMS 的外部加密只有您可以使用私有金鑰進行解密的資料。或者，您可以允許其他人員在 AWS KMS 的外部驗證您使用私有金鑰產生的數位簽章。

在您使用 AWS KMS 中非對稱 KMS 金鑰內的公有金鑰時，您可以從身分驗證、授權和記錄日誌中獲益，因為這些都是每個 AWS KMS 操作的一部分。您也可以降低加密無法解密資料的風險。這些功能在 AWS KMS 外部無效。如需詳細資訊，請參閱 下載公開金鑰的特殊考量。

> **Tip**
>
> 正在尋找資料金鑰或 SSH 金鑰？ 本主題介紹如何在不可匯出私有金鑰的 AWS Key Management Service 中管理非對稱金鑰。如需私密金鑰受對稱加密 KMS 金鑰保護的可匯出資料金鑰配對，請參閱。GenerateDataKeyPair如需有關下載與 Amazon EC2 執行個體相關聯之公有金鑰的說明，請參閱《Amazon EC2 Linux 執行個體使用者指南》以及《Amazon EC2 Windows 執行個體使用者指南》中的擷取公有金鑰。

主題

下載公開金鑰的特殊考量

為了保護您的 KMS 金鑰，AWS KMS 提供存取控制驗證加密，以及每個操作的詳細日誌。AWS KMS 也可以讓您暫時或永久避免使用 KMS 金鑰。最後，AWS KMS 操作旨在將加密無法解密資料的風險降至最低。這些功能在您於 AWS KMS 外部使用下載的公有金鑰時，將無法提供使用。

授權

金鑰政策和 IAM 政策，可控制對 AWS KMS 內 KMS 金鑰的存取，不會影響在 AWS 之外執行的操作。任何能夠取得公有金鑰的使用者，皆可在 AWS KMS 之外使用公有金鑰，即使這些使用者沒有使用 KMS 金鑰加密資料或驗證簽章的許可也一樣。

金鑰使用方式限制

金鑰使用情形限制在 AWS KMS 之外無效。如果您使用具有 `SIGN_VERIFY` 之 `KeyUsage` 的 KMS 金鑰呼叫 Encrypt 操作，則 AWS KMS 操作會失敗。但是，如果您使用來自具有 `SIGN_VERIFY` 的 `KeyUsage` 之 KMS 金鑰的公有金鑰為 AWS KMS 之外的資料加密，則無法解密資料。

演算法限制

AWS KMS 支援的加密和簽署演算法限制在 AWS KMS 外部無效。如果您在 AWS KMS 的外部使用來自 KMS 金鑰的公有金鑰加密資料，並使用了 AWS KMS 不支援的加密演算法，便無法解密資料。

停用和刪除 KMS 金鑰

您在 AWS KMS 中可以採取以防止在密碼編譯操作內使用 KMS 金鑰的動作，無法防止任何人員從 AWS KMS 的外部使用公有金鑰。例如，停用 KMS 金鑰、排程刪除 KMS 金鑰、刪除 KMS 金鑰，或是從 KMS 金鑰刪除金鑰材料等，對 AWS KMS 外部的公有金鑰沒有任何影響。如果您刪除了非對稱 KMS 金鑰或刪除或遺失了其金鑰材料，您在 AWS KMS 外部使用公有金鑰加密的資料便無法復原。

日誌

記錄每個 AWS KMS 操作 (包括請求、回應、日期、時間和經過授權的使用者) 的 AWS CloudTrail 日誌無法記錄在 AWS KMS 外部使用公有金鑰的情況。

使用 SM2 金鑰對進行離線驗證 (僅限中國區域)

若要使用 SM2 公有金鑰驗證在 AWS KMS 外部的簽署，您必須指定辨別 ID。根據預設，AWS KMS 使用 1234567812345678 作為辨別 ID。如需更多資訊，請參閱使用 SM2 金鑰對進行離線驗證 (僅限中國區域)。

下載公有金鑰 (主控台)

您可以使用 AWS Management Console來檢視、複製和下載您 AWS 帳戶 中非對稱 KMS 金鑰的公有金鑰。如要下載位於不同 AWS 帳戶 中非對稱 KMS 金鑰的公有金鑰，請使用 AWS KMS API。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。

2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 選擇非對稱 KMS 金鑰的別名或金鑰 ID。
5. 選擇 Cryptographic configuration (密碼編譯組態) 索引標籤。記錄 Key spec (金鑰規格)、Key usage (金鑰使用情形) 和 Encryption algorithms (加密演算法) 或 Signing Algorithms (簽署演算法) 欄位的值。您將需要使用這些值來在 AWS KMS 外部使用公有金鑰。請務必在您共享公有金鑰時共享此資訊。
6. 選擇 Public key (公有金鑰) 標籤。
7. 如要將公有金鑰複製到您的剪貼簿，請使用 Copy (複製)。如要將公有金鑰下載至檔案，請選擇 Download (下載)。

下載公有金鑰 (AWS KMS API)

GetPublicKey作業會傳回非對稱 KMS 金鑰中的公開金鑰。這項操作也會傳回在 AWS KMS 外部正確使用公有金鑰時所需要的重要資訊，包括金鑰使用方式和加密演算法。請務必儲存這些值，並在您共享公有金鑰時也共享這些資訊。

本節中的範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

若要指定 KMS 金鑰，請使用它的金鑰 ID、金鑰 ARN、別名名稱或別名 ARN。使用別名時，請加上 alias/ 字首。如要指定不同 AWS 帳戶 中的 KMS 金鑰，您必須使用其金鑰 ARN 或別名 ARN。

執行此命令前，請將範例別名替換成 KMS 金鑰的有效識別符。如要執行此命令，您必須擁有 KMS 金鑰的 `kms:GetPublicKey` 許可。

```
$ aws kms get-public-key --key-id alias/example_RSA_3072

{
    "KeySpec": "RSA_3072",
    "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "EncryptionAlgorithms": [
        "RSAES_OAEP_SHA_1",
        "RSAES_OAEP_SHA_256"
    ],
    "PublicKey": "MIIBojANBgkqhkiG..."
```

```
}
```

識別非對稱 KMS 金鑰

若要判斷特定 KMS 金鑰是否為非對稱 KMS 金鑰，請查看 key type (金鑰類型) 或 key spec (金鑰規格)。您可使用 AWS KMS 主控台或 AWS KMS API。

其中一些方法也會顯示 KMS 金鑰密碼編譯組態的其他方面，包括金鑰使用情形和 KMS 金鑰支援的加密或簽署演算法。您可以檢視現有 KMS 金鑰的密碼編譯組態，但無法變更它。

如需檢視 KMS 金鑰的一般資訊，包括排序、篩選和選擇主控台顯示的欄位，請參閱 在主控台中檢視 KMS 金鑰。

主題

- 在 KMS 金鑰資料表中尋找金鑰類型
- 在詳細資訊頁面尋找金鑰類型
- 使用 AWS KMS API 尋找金鑰規格

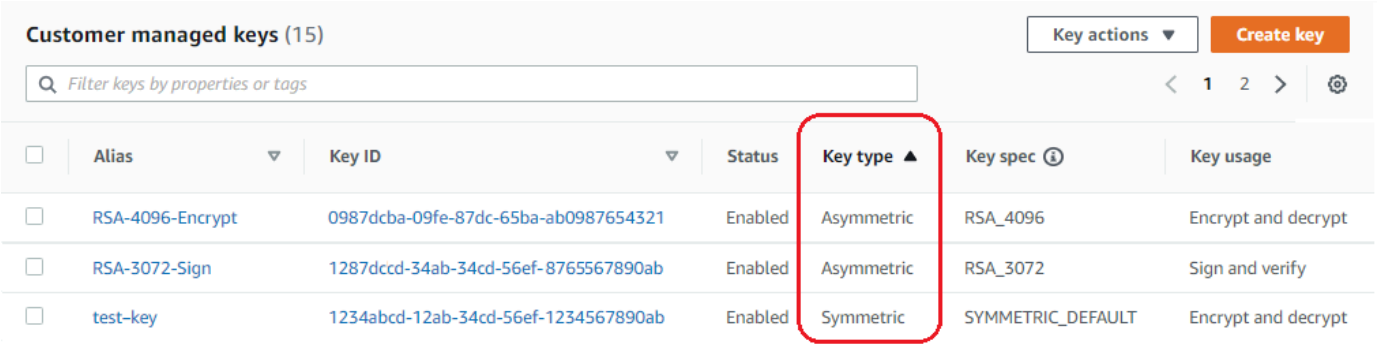
在 KMS 金鑰資料表中尋找金鑰類型

在 AWS KMS 主控台中，Key type (金鑰類型) 欄位會顯示每個 KMS 金鑰是對稱還是非對稱。您可以將 金鑰類型欄位新增至主控台中客戶受管金鑰或 AWS 受管金鑰 頁面上的 KMS 金鑰資料表。

若要識別 KMS 金鑰資料表中的對稱和非對稱 KMS 金鑰，請使用下列程序。

1. 開啟位於 https://console.aws.amazon.com/kms 的 AWS KMS 主控台。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 若要檢視您所建立及管理帳戶中的金鑰，請在導覽窗格中選擇Customer managed keys (客戶受管金鑰)。若要檢視 AWS 為您建立及管理之帳戶中的金鑰，請在導覽窗格中選擇 AWS 受管金鑰。
4. Key type (金鑰類型) 欄會顯示每個 KMS 金鑰是對稱還是非對稱。您也可以依 Key type (金鑰類型) 值進行排序和篩選。

   如果 KMS 金鑰資料表中沒有顯示 Key type (金鑰類型) 欄，請選擇頁面右上角的齒輪圖示，選擇 Key type (金鑰類型)，然後選擇 Confirm (確認)。您還可以新增 Key spec (金鑰規格) 和 Key usage (金鑰使用方式) 欄位。



在詳細資訊頁面尋找金鑰類型

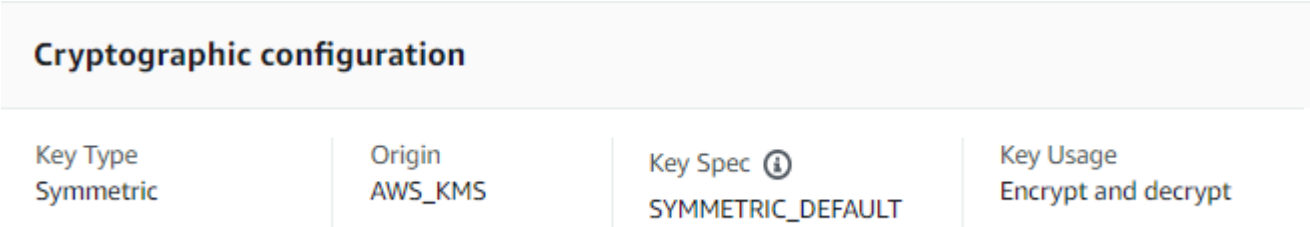
在 AWS KMS 主控台中，每個 KMS 金鑰的詳細資料頁面都有 Cryptographic Configuration (密碼編譯組態) 索引標籤，顯示 KMS 金鑰的金鑰類型 (對稱或非對稱) 和其他密碼編譯詳細資訊。

若要在 KMS 金鑰詳細資訊頁面上識別對稱和非對稱 KMS 金鑰，請使用下列程序。

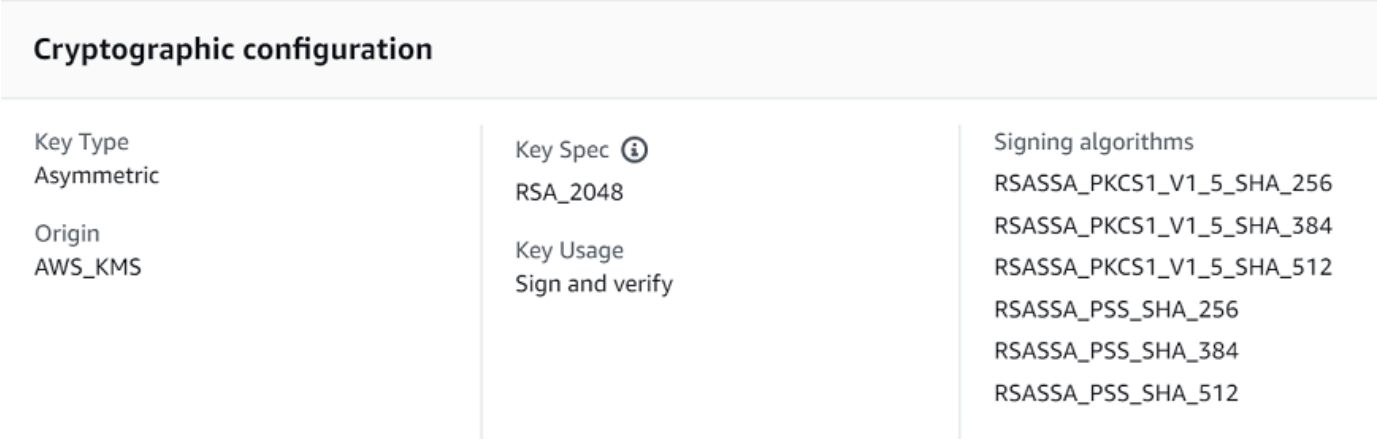
1. 開啟位於 https://console.aws.amazon.com/kms 的 AWS KMS 主控台。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 若要檢視您所建立及管理帳戶中的金鑰，請在導覽窗格中選擇Customer managed keys (客戶受管金鑰)。若要檢視 AWS 為您建立及管理之帳戶中的金鑰，請在導覽窗格中選擇 AWS 受管金鑰。
4. 選擇 KMS 金鑰的別名或金鑰 ID。
5. 選擇 Cryptographic configuration (密碼編譯組態) 索引標籤。索引標籤位於 General Configuration (一般組態) 區段下。

   Cryptographic configuration (密碼編譯組態) 區段會顯示 Key Type (金鑰類型)，指出它是對稱還是非對稱。它還會顯示 KMS 金鑰的其他詳細資訊，包括 Key Usage (金鑰使用情形)，表示 KMS 金鑰是否可用於加密和解密或簽署和驗證。對於非對稱 KMS 金鑰，它會顯示 KMS 金鑰支援的加密演算法或簽署演算法。

   例如，以下是對稱加密 KMS 金鑰的 Cryptographic configuration (密碼編譯組態) 索引標籤範例。



以下是非對稱 RSA KMS 金鑰 (用於簽署和驗證) 的範例 Cryptographic configuration (密碼編譯組態) 索引標籤。



使用 AWS KMS API 尋找金鑰規格

若要判斷 KMS 金鑰是對稱還是非對稱，請使用此DescribeKey作業。回應中的 KeySpec 欄位包含 KMS 金鑰的金鑰規格。若為對稱加密 KMS 金鑰，KeySpec 的值是 SYMMETRIC_DEFAULT。如果是其他值，則表示其為非對稱 KMS 金鑰或 HMAC KMS 金鑰。

> **Note**
>
> 已取代 CustomerMasterKeySpec 成員。請改用 KeySpec。若要防止中斷變更，則 DescribeKey 回應會包含具有相同值的 KeySpec 和 CustomerMasterKeySpec 成員。

例如，如果是對稱加密 KMS 金鑰，DescribeKey 會為傳回下列回應。KeySpec 值是 SYMMETRIC_DEFAULT。

```
{
  "KeyMetadata": {
    "AWSAccountId": "111122223333",
    "KeyId": "0987dcba-09fe-87dc-65ba-ab0987654321",
    "Arn": "arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321",
    "CreationDate": 1496966810.831,
    "Enabled": true,
    "Description": "",
    "KeyState": "Enabled",
```

```
    "Origin": "AWS_KMS",
    "KeyManager": "CUSTOMER",
    "MultiRegion": false,
    "KeySpec": "SYMMETRIC_DEFAULT",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "EncryptionAlgorithms": [
        "SYMMETRIC_DEFAULT"
    ]
  }
}
```

用於簽署和驗證之非對稱 RSA KMS 金鑰的 DescribeKey 回應看起來類似於此範例。KeySpec 值為 RSA_2048，KeyUsage 是 SIGN_VERIFY。SigningAlgorithms 元素會列出 KMS 金鑰的有效簽署演算法。

```
{
  "KeyMetadata": {
    "AWSAccountId": "111122223333",
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "CreationDate": 1571767572.317,
    "CustomerMasterKeySpec": "RSA_2048",
    "Enabled": false,
    "Description": "",
    "KeyState": "Disabled",
    "Origin": "AWS_KMS",
    "MultiRegion": false,
    "KeyManager": "CUSTOMER",
    "KeySpec": "RSA_2048",
    "KeyUsage": "SIGN_VERIFY",
    "SigningAlgorithms": [
        "RSASSA_PKCS1_V1_5_SHA_256",
        "RSASSA_PKCS1_V1_5_SHA_384",
        "RSASSA_PKCS1_V1_5_SHA_512",
        "RSASSA_PSS_SHA_256",
        "RSASSA_PSS_SHA_384",
        "RSASSA_PSS_SHA_512"
    ]
  }
}
```

非對稱金鑰規格

下列主題提供有關金鑰規格的技術資訊，這些都是 AWS KMS 支援的非對稱 KMS 金鑰規格。其中包括對稱加密金鑰的 SYMMETRIC_DEFAULT 金鑰規格資訊，以供比較之用。

主題

RSA 金鑰規格

當您使用 RSA 金鑰規格時，AWS KMS 會使用 RSA 金鑰對建立非對稱的 KMS 金鑰。私有金鑰絕不會讓 AWS KMS 出現未加密的狀況。您可以在 AWS KMS 中使用公有金鑰，或下載公有金鑰以供在 AWS KMS 外部使用。

> **Warning**
>
> 當在 AWS KMS 外加密資料時，請確定您可以解密加密文字。如果您使用已從 AWS KMS 刪除之 KMS 金鑰的公有金鑰、設定用於簽署和驗證之 KMS 金鑰的公有金鑰，或不受 KMS 金鑰支援的加密演算法，該資料無法復原。

在 AWS KMS 中，您可以使用非對稱 KMS 金鑰搭配 RSA 金鑰對執行加密和解密作業，或執行簽署和驗證操作，但不能同時用於兩種操作。此屬性稱為「金鑰用途」，與金鑰規格分開決定，但應該在選取金鑰規格前決定。

AWS KMS 支援下列 RSA 金鑰規格，處理加密和解密或簽署和驗證：

- RSA_2048
- RSA_3072
- RSA_4096

RSA 金鑰規格因 RSA 金鑰的位元長度而不同。您選擇的 RSA 金鑰規格可能隨安全標準或任務需求而定。任務一般會使用實用且可負擔的最長密鑰。具有不同 RSA 金鑰規格之 KMS 金鑰的密碼編譯操作

定價也不同。如需 AWS KMS 定價的資訊，請參閱 AWS 金鑰管理服務定價。如需請求配額的詳細資訊，請參閱 請求配額。

用於加密和解密的 RSA 金鑰規格

使用 RSA 非對稱 KMS 金鑰加密和解密時，您會使用公有金鑰加密，並使用私有金鑰解密。當您在 AWS KMS 中呼叫 `Encrypt` 操作以取得 RSA KMS 金鑰時，AWS KMS 會使用 RSA 金鑰對中的公有金鑰以及您指定的加密演算法，加密資料。若要解密加密文字，請呼叫 `Decrypt` 操作並指定相同的 KMS 金鑰和加密演算法。然後 AWS KMS 就會使用 RSA 金鑰對中的私有金鑰解密資料。

您也可以下載公有金鑰，用以加密 AWS KMS 外部的資料。請務必使用 AWS KMS 支援的 RSA KMS 金鑰加密演算法。若要解密加密文字，請使用相同的 KMS 金鑰和加密演算法呼叫 `Decrypt` 函數。

AWS KMS 支援兩種加密演算法，適用於具有 RSA 金鑰規格的 KMS 金鑰。這些演算法同在 PKCS #1 v2.2 中定義，但其內部使用的雜湊函數不同。在 AWS KMS 中，RSAES_OAEP 演算法對雜湊目的和遮罩產生函數 (MGF1) 一律使用相同的雜湊函數。當您呼叫 Encrypt 和 Decrypt 操作時，必須指定加密演算法。您可以為每個請求選擇不同的演算法。

支援 RSA 金鑰規格的加密演算法

加密演算法	演算法說明
RSAES_OAEP_SHA_1	PKCS #1 v2.2 第 7.1 節。RSA 加密與 OAEP 填補在雜湊和 MGF1 遮罩產生函數中都使用 SHA-1 加一個空標籤。
RSAES_OAEP_SHA_256	PKCS #1 第 7.1 節。RSA 加密與 OAEP 填補在雜湊和 MGF1 遮罩產生函數中都使用 SHA-256 加一個空標籤。

您無法將 KMS 金鑰設定為使用特定的加密演算法。不過，您可以使用 kms: EncryptionAlgorithm 原則條件來指定允許主體搭配 KMS 金鑰使用的加密演算法。

若要取得 KMS 金鑰的加密演算法，請在AWS KMS主控台中檢視 KMS 金鑰的密碼編譯組態或使用DescribeKey作業。 AWS KMS當您在AWS KMS主控台或使用GetPublicKey作業下載公開金鑰時，也會提供金鑰規格和加密演算法。

您可根據能在每個請求中加密的純文字資料長度，選擇 RSA 金鑰規格。下表顯示您在單次呼叫 Encrypt 操作中可以加密的純文字大小上限 (以位元組為單位)。這些值與金鑰規格和加密演算法不同。若要進行比較，您可以使用對稱加密 KMS 金鑰一次加密多達 4096 個位元組。

若要計算這些演算法的純文字長度上限 (以位元組為單位)，請使用以下公式：(*key_size_in_bits* / 8) - (2 * *hash_length_in_bits*/8) - 2。例如，具有 SHA-256 之 RSA_2048 的純文字大小上限 (位元組) 為 (2048/8) - (2 * 256/8) -2 = 190。

Encrypt 操作中的純文字大小上限 (位元組)

	加密演算法	
金鑰規格	RSAES_OAEP_SHA_1	RSAES_OAEP_SHA_256
RSA_2048	214	190
RSA_3072	342	318
RSA_4096	470	446

用於簽署和驗證的 RSA 金鑰規格

使用 RSA 非對稱 KMS 金鑰執行簽署和驗證作業時，您會使用私有金鑰產生訊息的簽章，並使用公有金鑰驗證該簽章。

當您在 AWS KMS 中呼叫 `Sign` 作業以取得非對稱 KMS 金鑰時，AWS KMS 會使用 RSA 金鑰對中的私有金鑰、訊息和您指定的簽署演算法，產生簽章。若要驗證簽章，請呼叫 Verify 操作。指定簽章，加上相同的 KMS 金鑰、訊息和簽署演算法。然後，AWS KMS 會使用 RSA 金鑰對中的公有金鑰驗證簽章。您也可以下載公有金鑰，用它驗證 AWS KMS 外部的簽章。

AWS KMS 支援下列具有 RSA 金鑰規格的所有 KMS 金鑰簽署演算法。當您呼叫 Sign 和 Verify 操作時，必須指定簽署演算法。您可以為每個請求選擇不同的演算法。使用 RSA 金鑰對進行簽署時，偏好使用 RSASSA-PSS 演算法。我們包含 RSASSA-PKCS1-v1_5 演算法，以便與現有應用程式相容。

支援 RSA 金鑰規格的簽署演算法

簽署演算法	演算法說明
RSASSA_PSS_SHA_256	PKCS #1 v2.2 第 8.1 節，具有 PSS 填補和使用 SHA-256 的 RSA 簽章，適用於訊息摘要和 MGF1 遮罩產生函數以及 256 位元的 salt

簽署演算法	演算法說明
RSASSA_PSS_SHA_384	PKCS #1 v2.2 第 8.1 節，具有 PSS 填補和使用 SHA-384 的 RSA 簽章，適用於訊息摘要和 MGF1 遮罩產生函數以及 384 位元的 salt
RSASSA_PSS_SHA_512	PKCS #1 v2.2 第 8.1 節，具有 PSS 填補和使用 SHA-512 的 RSA 簽章，適用於訊息摘要和 MGF1 遮罩產生函數以及 512 位元的 salt
RSASSA_PKCS1_V1_5_SHA_256	PKCS #1 v2.2 第 8.2 節，具有 PKCS #1v1.5 填補和 SHA-256 的 RSA 簽章
RSASSA_PKCS1_V1_5_SHA_384	PKCS #1 v2.2 第 8.2 節，具有 PKCS #1v1.5 填補和 SHA-384 的 RSA 簽章
RSASSA_PKCS1_V1_5_SHA_512	PKCS #1 v2.2 第 8.2 節，具有 PKCS #1v1.5 填補和 SHA-512 的 RSA 簽章

您無法將 KMS 金鑰設定為使用特定的簽署演算法。不過，您可以使用 kms: SigningAlgorithm 原則條件來指定允許主體搭配 KMS 金鑰使用的簽署演算法。

若要取得 KMS 金鑰的簽署演算法，請在AWS KMS主控台中檢視 KMS 金鑰的密碼編譯組態，或使用DescribeKey作業。 AWS KMS當您在AWS KMS主控台或使用GetPublicKey作業下載公開金鑰時，也會提供金鑰規格和簽章演算法。

橢圓曲線金鑰規格

當您使用橢圓曲線 (ECC) 金鑰規格時，AWS KMS 會建立具有 ECC 金鑰對的非對稱 KMS 金鑰，以供簽署和驗證之用。產生簽名的私有金鑰絕不會讓 AWS KMS 出現未加密狀況。您可以在 AWS KMS 中使用公有金鑰驗證簽章，或下載公有金鑰以供在 AWS KMS 外部使用。

AWS KMS 支援下列非對稱 KMS 金鑰的 ECC 金鑰規格。

- 非對稱 NIST 建議的橢圓曲線金鑰對 (簽署和驗證)
  - ECC_NIST_P256 (secp256r1)
  - ECC_NIST_P384 (secp384r1)
  - ECC_NIST_P521 (secp521r1)

- 其他非對稱橢圓曲線金鑰對 (簽署和驗證)
  - ECC_SECG_P256K1 (secp256k1)，常用於加密貨幣。

您選擇的 ECC 金鑰規格可能隨安全標準或任務需求而定。任務一般會使用實用且可負擔的最多點曲線。

如果您要建立非對稱 KMS 金鑰以搭配使用加密貨幣，請使用 ECC_SECG_P256K1 金鑰規格。此金鑰規格也可供其他用途使用，但對 Bitcoin 和其他加密貨幣而言是必要項目。

具有不同 ECC 金鑰規格的 KMS 金鑰定價也不同，而且有不同的請求配額。如需 AWS KMS 定價的資訊，請參閱 AWS Key Management Service 定價。如需請求配額的詳細資訊，請參閱 請求配額。

下表顯示 AWS KMS 支援的每種 ECC 金鑰規格簽署演算法。您無法將 KMS 金鑰設定為使用特定的簽署演算法。不過，您可以使用 kms: SigningAlgorithm 原則條件來指定允許主體搭配 KMS 金鑰使用的簽署演算法。

支援 ECC 金鑰規格的簽署演算法

金鑰規格	簽署演算法	演算法說明
ECC_NIST_P256	ECDSA_SHA_256	NIST FIPS 186-4 第 6.4 節，適用於訊息摘要之金鑰和 SHA-256 所指定的使用曲線的 ECDSA 簽章。
ECC_NIST_P384	ECDSA_SHA_384	NIST FIPS 186-4 第 6.4 節，適用於訊息摘要之金鑰和 SHA-384 所指定的使用曲線的 ECDSA 簽章。
ECC_NIST_P521	ECDSA_SHA_512	NIST FIPS 186-4 第 6.4 節，適用於訊息摘要之金鑰和 SHA-512 所指定的使用曲線的 ECDSA 簽章。
ECC_SECG_P256K1	ECDSA_SHA_256	NIST FIPS 186-4 第 6.4 節，適用於訊息摘要之金鑰和 SHA-256 所指定的使用曲線的 ECDSA 簽章。

SM2 金鑰規格 (僅限中國區域)

SM2 金鑰規格是在中國商用密碼管理辦公室 (OSCCA) 公佈的 GM/T 系列規格中定義的橢圓曲線金鑰規格。SM2 金鑰規格僅在中國區域提供。當您使用 SM2 金鑰規格時，AWS KMS 會使用 SM2 金鑰對建立非對稱的 KMS 金鑰。您可以在 AWS KMS 中使用您的 SM2 金鑰對，或下載公有金鑰以供在 AWS KMS 外部使用。

與 ECC 金鑰規格不同，您可以使用 SM2 KMS 金鑰進行簽署和驗證，或者進行加密和解密。在建立 KMS 金鑰時，您必須指定金鑰使用方式，且在金鑰建立之後即無法變更。

AWS KMS 支援以下 SM2 加密和簽署演算法：

- SM2PKE 加密演算法

  SM2PKE 是 OSCCA 在 GM/T 0003.4-2012 中定義的橢圓曲線型加密演算法。

- SM2DSA 簽署演算法

  SM2DSA 是 OSCCA 在 GM/T 0003.2-2012 中定義的橢圓曲線型簽署演算法。SM2DSA 需要使用 SM3 雜湊演算法進行雜湊處理的辨別 ID，然後與您傳遞給 AWS KMS 的訊息或訊息摘要組合使用。接著由 AWS KMS 將此串接值進行雜湊處理並簽署。

使用 SM2 進行離線操作 (僅限中國區域)

您可以下載您 SM2 金鑰對的公有金鑰以進行離線操作，即在 AWS KMS 的外部操作。但是，離線使用 SM2 公有金鑰時，您可能需要手動執行額外的轉換和計算。SM2DSA 操作可能會要求您提供辨別 ID 或計算訊息摘要。SM2PKE 加密操作可能需要您將原始密文輸出轉換為 AWS KMS 可以接受的格式。

為了協助您進行這些操作，Java 的 `SM2OfflineOperationHelper` 類別具有為您執行任務的方法。您可以使用此輔助程式類別作為針對其他密碼提供者的模型。

> **Important**
>
> `SM2OfflineOperationHelper` 參考程式碼設計為與 Bouncy Castle 1.68 版相容。如需其他版本的幫助，請聯絡 bouncycastle.org。

使用 SM2 金鑰對進行離線驗證 (僅限中國區域)

驗證簽章AWS KMS使用 SM2 公有金鑰，您必須指定辨別 ID。當您傳遞原始訊息 `MessageType:RAW` 至 《簽署》API 時，AWS KMS 使用 OSCCA 在 GM/T 0009-2012 中定義的預設辨別 ID，1234567812345678。您無法在 AWS KMS 中指定自己的辦別 ID。

但是，如果您在 AWS 之外產生訊息摘要，則您可以指定自己的辨別 ID，然後將訊息摘要 `MessageType:DIGEST` 傳遞至 AWS KMS 進行簽署。若要執行此操作，請變更 `SM2OfflineOperationHelper` 類別中的 `DEFAULT_DISTINGUISHING_ID` 值。您指定的辨別 ID 可以是最長 8,192 個字元的任何字串。AWS KMS 簽署訊息摘要後，您需要訊息摘要或訊息以及用來計算摘要的辦別 ID，以進行離線驗證。

`SM2OfflineOperationHelper` 類別

在 AWS KMS 之內，原始密文轉換和 SM2DSA 訊息摘要計算會自動執行。並非所有加密提供者都以相同的方式實施 SM2。有些程式庫，像是 OpenSSL 1.1.1 版及更高版本會自動執行這些動作。在使用 OpenSSL 3.0 版進行測試時，AWS KMS 會確認此行為。使用以下 `SM2OfflineOperationHelper` 類別和像是 Bouncy Castle 之類的程式庫，則需要您手動執行這些轉換和計算。

所以 `SM2OfflineOperationHelper` 類別提供了進行以下離線操作的方法：

- 訊息摘要計算

  若要離線產生可用於離線驗證或可傳遞至 AWS KMS 進行簽署的訊息摘要，請使用 `calculateSM2Digest` 方法。`calculateSM2Digest` 會使用 SM3 雜湊演算法產生訊息摘要。GetPublicKeyAPI 會以二進位格式傳回您的公開金鑰。您必須將二進制密鑰解析為 Java PublicKey。提供剖析好的公有金鑰以及訊息。該方法會自動將您的訊息與預設的辨別 ID (1234567812345678) 相結合，但您可以藉由變更 `DEFAULT_DISTINGUISHING_ID` 值來設置自己的辨別 ID。

- 確認

  若要離線驗證簽署，請使用 `offlineSM2DSAVerify` 方法。`offlineSM2DSAVerify` 方法使用從指定辨別 ID 計算出的訊息摘要，以及您提供的原始訊息來驗證數位簽署。GetPublicKeyAPI 會以二進位格式傳回您的公開金鑰。您必須將二進制密鑰解析為 Java PublicKey。提供已剖析的公開金鑰，以及您要驗證的原始訊息和簽章。如需詳細資訊，請參閱使用 SM2 金鑰對進行離線驗證。

- 加密

  若要離線加密明文，請使用 `offlineSM2PKEEncrypt` 方法。此方法可確保密文是 AWS KMS 能夠進行解密的格式。`offlineSM2PKEEncrypt` 方法加密明文，然後將由 SM2PKE 產生的原

始密文轉換為 ASN.1 格式。GetPublicKeyAPI 會以二進位格式傳回您的公開金鑰。您必須將二進制密鑰解析為 Java PublicKey。提供剖析好的公有金鑰以及以您要加密的明文。

如果您不確定是否需要執行轉換，請使用以下 OpenSSL 操作來測試您的密文格式。如果操作失敗，則需要將密文轉換為 ASN.1 格式。

```
openssl asn1parse -inform DER -in ciphertext.der
```

根據預設，當產生用於 SM2DSA 操作的訊息摘要時，SM2OfflineOperationHelper 類別使用預設的辨別 ID，即 1234567812345678。

```
package com.amazon.kms.utils;

import javax.crypto.BadPaddingException;
import javax.crypto.Cipher;
import javax.crypto.IllegalBlockSizeException;
import javax.crypto.NoSuchPaddingException;
import java.io.IOException;
import java.math.BigInteger;
import java.nio.ByteBuffer;
import java.nio.charset.StandardCharsets;
import java.security.InvalidKeyException;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.security.NoSuchProviderException;
import java.security.PrivateKey;
import java.security.PublicKey;

import org.bouncycastle.crypto.CryptoException;
import org.bouncycastle.jce.interfaces.ECPublicKey;

import java.util.Arrays;

import org.bouncycastle.asn1.ASN1EncodableVector;
import org.bouncycastle.asn1.ASN1Integer;
import org.bouncycastle.asn1.DEROctetString;
import org.bouncycastle.asn1.DERSequence;
import org.bouncycastle.asn1.gm.GMNamedCurves;
import org.bouncycastle.asn1.x9.X9ECParameters;
import org.bouncycastle.crypto.CipherParameters;
import org.bouncycastle.crypto.params.ParametersWithID;
```

```
import org.bouncycastle.crypto.params.ParametersWithRandom;
import org.bouncycastle.crypto.signers.SM2Signer;
import org.bouncycastle.jcajce.provider.asymmetric.util.ECUtil;

public class SM2OfflineOperationHelper {
    // You can change the DEFAULT_DISTINGUISHING_ID value to set your own
 distinguishing ID,
    // the DEFAULT_DISTINGUISHING_ID can be any string up to 8,192 characters long.
    private static final byte[] DEFAULT_DISTINGUISHING_ID =
 "1234567812345678".getBytes(StandardCharsets.UTF_8);
    private static final X9ECParameters SM2_X9EC_PARAMETERS =
 GMNamedCurves.getByName("sm2p256v1");

    // ***calculateSM2Digest***
    // Calculate message digest
    public static byte[] calculateSM2Digest(final PublicKey publicKey, final byte[]
 message) throws
            NoSuchProviderException, NoSuchAlgorithmException {
        final ECPublicKey ecPublicKey = (ECPublicKey) publicKey;

        // Generate SM3 hash of default distinguishing ID, 1234567812345678
        final int entlenA = DEFAULT_DISTINGUISHING_ID.length * 8;
        final byte [] entla = new byte[] { (byte) (entlenA & 0xFF00), (byte) (entlenA &
 0x00FF) };
        final byte [] a = SM2_X9EC_PARAMETERS.getCurve().getA().getEncoded();
        final byte [] b = SM2_X9EC_PARAMETERS.getCurve().getB().getEncoded();
        final byte [] xg = SM2_X9EC_PARAMETERS.getG().getXCoord().getEncoded();
        final byte [] yg = SM2_X9EC_PARAMETERS.getG().getYCoord().getEncoded();
        final byte[] xa = ecPublicKey.getQ().getXCoord().getEncoded();
        final byte[] ya = ecPublicKey.getQ().getYCoord().getEncoded();
        final byte[] za = MessageDigest.getInstance("SM3", "BC")
                .digest(ByteBuffer.allocate(entla.length +
 DEFAULT_DISTINGUISHING_ID.length + a.length + b.length + xg.length + yg.length +
                        xa.length +
 ya.length).put(entla).put(DEFAULT_DISTINGUISHING_ID).put(a).put(b).put(xg).put(yg).put(xa).put
                        .array());

        // Combine hashed distinguishing ID with original message to generate final
 digest
        return MessageDigest.getInstance("SM3", "BC")
                .digest(ByteBuffer.allocate(za.length +
 message.length).put(za).put(message)
                        .array());
    }
```

```
   // ***offlineSM2DSAVerify***
   // Verify digital signature with SM2 public key
   public static boolean offlineSM2DSAVerify(final PublicKey publicKey, final byte []
message,
           final byte [] signature) throws InvalidKeyException {
       final SM2Signer signer = new SM2Signer();
       CipherParameters cipherParameters =
ECUtil.generatePublicKeyParameter(publicKey);
       cipherParameters = new ParametersWithID(cipherParameters,
DEFAULT_DISTINGUISHING_ID);
       signer.init(false, cipherParameters);
       signer.update(message, 0, message.length);
       return signer.verifySignature(signature);
   }

   // ***offlineSM2PKEEncrypt***
   // Encrypt data with SM2 public key
   public static byte[] offlineSM2PKEEncrypt(final PublicKey publicKey, final byte []
plaintext) throws
           NoSuchPaddingException, NoSuchAlgorithmException, NoSuchProviderException,
InvalidKeyException,
           BadPaddingException, IllegalBlockSizeException, IOException {
       final Cipher sm2Cipher = Cipher.getInstance("SM2", "BC");
       sm2Cipher.init(Cipher.ENCRYPT_MODE, publicKey);

       // By default, Bouncy Castle returns raw ciphertext in the c1c2c3 format
       final byte [] cipherText = sm2Cipher.doFinal(plaintext);

       // Convert the raw ciphertext to the ASN.1 format before passing it to AWS KMS
       final ASN1EncodableVector asn1EncodableVector = new ASN1EncodableVector();
       final int coordinateLength = (SM2_X9EC_PARAMETERS.getCurve().getFieldSize() +
7) / 8 * 2 + 1;
       final int sm3HashLength = 32;
       final int xCoordinateInCipherText = 33;
       final int yCoordinateInCipherText = 65;
       byte[] coords = new byte[coordinateLength];
       byte[] sm3Hash = new byte[sm3HashLength];
       byte[] remainingCipherText = new byte[cipherText.length - coordinateLength -
sm3HashLength];

       // Split components out of the ciphertext
       System.arraycopy(cipherText, 0, coords, 0, coordinateLength);
```

```
        System.arraycopy(cipherText, cipherText.length - sm3HashLength, sm3Hash, 0,
 sm3HashLength);
        System.arraycopy(cipherText, coordinateLength, remainingCipherText,
 0,cipherText.length - coordinateLength - sm3HashLength);

        // Build standard SM2PKE ASN.1 ciphertext vector
        asn1EncodableVector.add(new ASN1Integer(new BigInteger(1,
 Arrays.copyOfRange(coords, 1, xCoordinateInCipherText))));
        asn1EncodableVector.add(new ASN1Integer(new BigInteger(1,
 Arrays.copyOfRange(coords, xCoordinateInCipherText, yCoordinateInCipherText))));
        asn1EncodableVector.add(new DEROctetString(sm3Hash));
        asn1EncodableVector.add(new DEROctetString(remainingCipherText));

        return new DERSequence(asn1EncodableVector).getEncoded("DER");
    }
}
```

SYMMETRIC_DEFAULT 金鑰規格

預設的金鑰規格 SYMMETRIC_DEFAULT 是對稱加密 KMS 金鑰的金鑰規格。在 AWS KMS 主控台中選取 Symmetric (對稱) 金鑰類型和 Encrypt and decrypt (加密和解密) 金鑰用途時，系統會選取 SYMMETRIC_DEFAULT 金鑰規格。在CreateKey作業中，如果您未指定KeySpec值，則會選取「對稱_預設值」。如果沒有使用不同金鑰規格的理由，SYMMETRIC_DEFAULT 是很好的選擇。

SYMMETRIC_DEFAULT 目前表示 AES-256-GCM，一種基於具有 256 位元金鑰之 Galois 計數器模式 (GCM) 中的進階加密標準 (AES) 的對稱演算法，是安全加密的業界標準。此演算法產生的加密文字支援額外的驗證資料 (AAD)，如加密內容，而 GCM 則提供對加密文字的額外完整性檢查。如需技術詳細資訊，請參閱 AWS Key Management Service 密碼編譯詳細資訊。

使用 AES-256-GCM 加密的資料現在和未來都受到保護。密碼學家認為此演算法具有「量子抗性」。在理論上，未來針對以 256 位元 AES-GCM 金鑰建立的加密文字所做的大規模量子運算攻擊，會將金鑰的有效安全性降低到 128 位元。但這種安全等級足以讓針對 AWS KMS 加密文字所做的暴力破解攻擊變得不可行。

中國區域是唯一的例外，在該處 SYMMETRIC_DEFAULT 代表使用 SM4 加密的 128 位元對稱金鑰。您只能在中國區域內建立 128 位元 SM4 金鑰。您無法在中國區域內建立 256 位元 AES-GCM KMS 金鑰。

您可在 AWS KMS 採用對稱加密 KMS 金鑰來加密、解密及重新加密資料，並保護產生的資料金鑰與資料金鑰對。整合 AWS 的 AWS KMS 服務會採用對稱加密 KMS 金鑰來加密靜態資料。您可以將自己的

金鑰資料匯入對稱加密 KMS 金鑰，並在自訂金鑰存放區中建立對稱加密 KMS 金鑰。如需對稱和非對稱 KMS 金鑰執行之操作的比較表，請參閱比較對稱和非對稱 KMS 金鑰。

如需有關 AWS KMS 和對稱加密金鑰的技術詳細資訊，請參閱 AWS Key Management Service 密碼編譯詳細資訊。

AWS KMS 中的 HMAC 金鑰

雜湊訊息驗證碼 (HMAC) KMS 金鑰是對稱金鑰，用於在 AWS KMS 中產生和驗證 HMAC。與每個 HMAC KMS 金鑰關聯的唯一金鑰資料提供了 HMAC 演算法所需的私密金鑰。您可以將 HMAC KMS 金鑰與 `GenerateMac` 和 `VerifyMac` 操作搭配使用來驗證 AWS KMS 之中的資料完整性和真實性。

HMAC 演算法結合了密碼編譯雜湊函數和共享私密金鑰。它們接受訊息和私密金鑰 (如 HMAC KMS 金鑰中的金鑰資料)，並傳回一個唯一且固定大小的代碼或標籤。如果訊息中即使只有一個字元更改，或者私密金鑰不相同，則產生的標籤將完全不同。透過要求私密金鑰，HMAC 還提供了真實性；如果沒有私密金鑰，就無法產生相同的 HMAC 標籤。HMAC 有時被稱為對稱簽章，因為它們的運作方式類似於數位簽章，但使用單一金鑰進行簽署和驗證。

AWS KMS 使用的 HMAC KMS 金鑰和 HMAC 演算法符合 RFC 2104 中定義的產業標準。此作AWS KMSGenerateMac業會產生標準 HMAC 標籤。HMAC KMS 金鑰會在已通過 FIPS 140-2 密碼編譯模組驗證計劃驗證的 AWS KMS 硬體安全模組中產生 (中國 (北京) 和中國 (寧夏) 區域除外)，且絕不讓 AWS KMS 出現未加密的狀況。若要使用 HMAC KMS 金鑰，必須呼叫 AWS KMS。

您可使用 HMAC KMS 金鑰來確定訊息的真實性，例如 JSON Web 字符 (JWT)、字符化的信用卡資訊或提交的密碼。它們還可以用作安全金鑰衍生函數 (KDF)，特別是在需要確定性金鑰的應用程式中。

HMAC KMS 金鑰比應用程式軟體的 HMAC 具有優勢，因為金鑰資料完全在 AWS KMS 之中產生並使用，完全受控於您在金鑰上設定的存取控制。

> **Tip**
>
> 最佳實務是建議您限制任何簽名機制 (包括 HMAC) 的有效時間。這可以阻止攻擊者使用簽署的訊息反覆建立有效性攻擊或在訊息被取代很長時間後反覆建立有效性攻擊。HMAC 標籤不包含時間戳記，但您可以在字符或訊息中包含時間戳記，以幫助您偵測何時重新整理 HMAC。

授權使用者可建立、管理及運用 AWS 帳戶中的 HMAC KMS 金鑰。其中包括啟用和停用金鑰、設定和變更別名和標籤，以及排定刪除 HMAC KMS 金鑰。您也可以使用金鑰政策、IAM 政策和授予，控制對 HMAC KMS 金鑰的存取。您可以在 AWS 日誌 中的 AWS CloudTrail 內稽核使用或管理 HMAC

KMS 金鑰的所有操作。您可利用匯入的金鑰資料來建立 HMAC KMS 金鑰。您也可以建立 HMAC 多區域 KMS 金鑰，這些金鑰在多個 AWS 區域中的運作方式會類似於相同的 HMAC KMS 金鑰副本。

HMAC KMS 金鑰僅支援 GenerateMac 和 VerifyMac 密碼編譯操作。您無法使用 HMAC KMS 金鑰來加密資料或簽署訊息，或在 HMAC 操作中使用任何其他類型的 KMS 金鑰。使用 GenerateMac 操作時，您可以提供最多 4,096 個位元組的訊息、一個 HMAC KMS 金鑰以及與 HMAC 金鑰規格相容的 MAC 演算法，GenerateMac 會運算 HMAC 標籤。若要驗證 HMAC 標籤，您必須提供 HMAC 標籤以及 GenerateMac 用來運算原始 HMAC 標籤的相同訊息、HMAC KMS 金鑰以及 MAC 演算法。VerifyMac 操作會運算 HMAC 標籤並驗證它是否與提供的 HMAC 標籤相同。如果輸入和運算得出的 HMAC 標籤不相同，驗證會失敗。

HMAC KMS 金鑰不支援金鑰自動輪換，而且您無法在自訂金鑰存放區建立 HMAC KMS 金鑰。

如果您要建立 KMS 金鑰來加密 AWS 服務中的資料，請使用對稱加密金鑰。您無法使用 HMAC KMS 金鑰。

區域

AWS KMS 支援的所有 AWS 區域 都支援 HMAC KMS 金鑰。

進一步了解

- 如需有關如何選擇 KMS 金鑰類型的說明，請參閱選擇一個 KMS 金鑰類型。
- 如需每種 KMS 金鑰類型支援之 AWS KMS API 操作的比較表，請參閱金鑰類型參考。
- 如需有關建立多區域 HMAC KMS 金鑰的詳細資訊，請參閱 AWS KMS 中的多區域金鑰。
- 若要檢查 AWS KMS 主控台為 HMAC KMS 金鑰設定之預設金鑰政策的差異，請參閱the section called “允許金鑰使用者使用 KMS 金鑰搭配 AWS 服務”。
- 如需有關 HMAC KMS 金鑰定價的詳細資訊，請參閱 AWS Key Management Service 定價。
- 如需有關適用於 HMAC KMS 金鑰配額的詳細資訊，請參閱資源配額和請求配額。
- 如需有關如何刪除 HMAC KMS 金鑰的詳細資訊，請參閱刪除 AWS KMS keys。
- 若要了解有關使用 HMAC 建立 JSON Web 字符的詳細資訊，請參閱《AWS 安全部落格》中的「如何在 AWS KMS 內保護 HMAC」。
- 在官方 AWS Podcast 上收聽 Podcast：AWS Key Management Service 的 HMAC 簡介。

主題

- HMAC KMS 金鑰的金鑰規格

- 建立 HMAC KMS 金鑰
- 控制對 HMAC KMS 金鑰的存取
- 檢視 HMAC KMS 金鑰

HMAC KMS 金鑰的金鑰規格

AWS KMS 支援不同長度的對稱 HMAC 金鑰。金鑰規格可根據您的安全、法規或業務需求進行選取。金鑰的長度決定了用於GenerateMac和VerifyMac作業的 MAC 演算法。通常，較長的金鑰更安全。使用適用於您的使用案例的最長金鑰。

HMAC 金鑰規格	MAC 演算法
HMAC_224	HMAC_SHA_224
HMAC_256	HMAC_SHA_256
HMAC_384	HMAC_SHA_384
HMAC_512	HMAC_SHA_512

建立 HMAC KMS 金鑰

您可以在 AWS KMS 主控台中，使用 CreateKey API 或 AWS CloudFormation 範本來建立 HMAC KMS 金鑰。

AWS KMS 支援 HMAC KMS 金鑰的多種金鑰規格。金鑰規格可根據您的法規、安全或業務需求進行選取。一般而言，較長的金鑰更能抵禦暴力破解攻擊。

> **Important**
>
> 請勿在別名、說明或標籤包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

若要建立 KMS 金鑰來加密在 AWS 服務中的資料，請使用對稱加密 KMS 金鑰。與 AWS KMS 整合的 AWS 服務不支援非對稱 KMS 金鑰或 HMAC KMS 金鑰。如需有關如何建立對稱加密 KMS 金鑰的說明，請參閱建立金鑰。

進一步了解

- 若要確定要建立哪種 KMS 金鑰，請參閱選擇一個 KMS 金鑰類型。
- 您可以使用本主題中所述的程序來建立多區域主要 HMAC KMS 金鑰。若要複製多區域 HMAC 金鑰，請參閱the section called “建立複本金鑰”。
- 如需建立 KMS 金鑰所需之許可的詳細資訊，請參閱 建立 KMS 金鑰的許可。
- 如需使用AWS CloudFormation範本建立 HMAC KMS 金鑰的相關資訊，請參閱使用AWS CloudFormation者指南AWS::KMS::Key中的〈〉。

主題

- 建立 HMAC KMS 金鑰 (主控台)
- 建立 HMAC KMS 金鑰 (AWS KMS API)

建立 HMAC KMS 金鑰 (主控台)

您可以使用 AWS Management Console 來建立 HMAC KMS 金鑰。HMAC KMS 金鑰是對稱金鑰，其金鑰用途為 Generate and verify MAC (產生和驗證 MAC)。您也可以建立多區域 HMAC 金鑰。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 選擇建立金鑰。
5. 針對 Key type (金鑰類型)，請選擇 Symmetric (對稱)。

   HMAC KMS 金鑰為對稱金鑰。您可以使用相同的金鑰產生和驗證 HMAC 標籤。
6. 在 Key usage (金鑰用途) 欄位中，請選擇 Generate and verify MAC (產生和驗證 MAC)。

   產生和驗證 MAC 是 HMAC KMS 金鑰的唯一有效金鑰用途。

   > **Note**
   >
   > 僅當所選區域支援 HMAC KMS 金鑰時，才會顯示對稱金鑰的 Key usage (金鑰用途)。
7. 選取 HMAC KMS 金鑰的規格 (Key spec (金鑰規格))。

金鑰規格可根據您的法規、安全或業務需求進行選取。通常，較長的金鑰更安全。

8. 若要建立多區域主要 HMAC 金鑰，請在 Advanced options (進階選項) 中選擇 Multi-Region key (多區域金鑰)。您為此 KMS 金鑰定義的共用屬性 (如金鑰類型和金鑰用途) 將與其複本金鑰共享。如需詳細資訊，請參閱 建立多區域金鑰。

   您無法使用此程序來建立複本金鑰。若要建立多區域複本 HMAC 金鑰，請遵循建立複本金鑰的指示。
9. 選擇下一步。
10. 輸入 KMS 金鑰的別名。別名名稱的開頭不可以是 **`aws/`**。**`aws/`** 字首由 Amazon Web Services 保留，以代表您帳戶中的 AWS 受管金鑰。

    建議您使用能將 KMS 金鑰識別為 HMAC 金鑰的別名，例如 `HMAC/test-key`。這可讓您更容易識別 AWS KMS 主控台中的 HMAC 金鑰，您可以依標籤和別名對金鑰進行排序和篩選，但不能依金鑰規格或金鑰用途進行排序和篩選。

    在 AWS Management Console 中建立 KMS 金鑰時需要別名。使用CreateKey作業時無法指定別名，但可以使用主控台或CreateAlias作業為現有 KMS 金鑰建立別名。如需詳細資訊，請參閱 使用別名。
11. (選用) 輸入 KMS 金鑰的描述。

    輸入描述來說明您計劃保護的資料類型，或您計劃搭配 KMS 金鑰一起使用的應用程式。

    您可以立即新增描述或在任意時間更新，除非金鑰狀態為 `Pending Deletion` 或 `Pending Replica Deletion`。若要加入、變更或刪除現有客戶管理金鑰的描述，請編輯中的描述AWS Management Console或使用UpdateKeyDescription作業。
12. (選用) 輸入標籤索引鍵和選用標籤值。若要將其他標籤新增至 KMS 金鑰，請選擇 Add tag (新增標籤)。

    考慮新增能將金鑰識別為 HMAC 金鑰的標籤，例如 `Type=HMAC`。這可讓您更容易識別 AWS KMS 主控台中的 HMAC 金鑰，您可以依標籤和別名對金鑰進行排序和篩選，但不能依金鑰規格或金鑰用途進行排序和篩選。

    將標籤新增到 AWS 資源時，AWS 會產生成本配置報告，內含按標籤彙總的用量與成本。標籤也可以用來控制 KMS 金鑰的存取。如需標記 KMS 金鑰的詳細資訊，請參閱 標記金鑰 和 AWS KMS 的 ABAC。
13. 選擇下一步。
14. 選取可管理 KMS 金鑰的 IAM 使用者和角色。

> **Note**
>
> 此金鑰政策授予 AWS 帳戶 完全控制此 KMS 金鑰。它允許帳戶管理員使用 IAM 政策授予其他主體管理 KMS 金鑰的許可。如需詳細資訊，請參閱 the section called “預設金鑰政策”。
> IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

15. (選用) 為了防止選取的 IAM 使用者和角色刪除此 KMS 金鑰，請在頁面底部的 Key deletion (金鑰刪除) 區段中，清除 Allow key administrators to delete this key (允許金鑰管理員刪除此金鑰) 核取方塊。
16. 選擇下一步。
17. 選取可將 KMS 金鑰用於密碼編譯操作的 IAM 使用者和角色。

> **Note**
>
> 此金鑰政策授予 AWS 帳戶 完全控制此 KMS 金鑰。它允許帳戶管理員使用 IAM 政策授予其他主體在密碼編譯操作中使用 KMS 金鑰的許可。如需詳細資訊，請參閱 the section called “預設金鑰政策”。
> IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

18. (選用) 您可以允許其他 AWS 帳戶 將此 KMS 金鑰用於密碼編譯操作。若要這樣做，請在頁面底部的其他 AWS 帳戶 區段中，選擇新增另一個 AWS 帳戶，然後輸入外部帳戶的 AWS 帳戶 識別號碼。若要新增多個外部帳戶，請重複此步驟。

> **Note**
>
> 若要允許外部帳戶中的主體使用 KMS 金鑰，外部帳戶的管理員必須建立 IAM 政策來提供這些許可。如需詳細資訊，請參閱 允許其他帳戶中的使用者使用 KMS 金鑰。

19. 選擇下一步。
20. 檢閱您選擇的金鑰設定。您仍然可以返回並變更所有設定。
21. 選擇 Finish (完成) 來建立 HMAC KMS 金鑰。

建立 HMAC KMS 金鑰 (AWS KMS API)

您可以使用此CreateKey作業來建立 HMAC KMS 金鑰。以下範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

建立 HMAC KMS 金鑰時，必須指定 KeySpec 參數，以決定 KMS 金鑰的類型。此外，您必須指定 GENERATE_VERIFY_MAC 的 KeyUsage 值，即使它是 HMAC 金鑰的唯一有效金鑰用途值。若要建立多區域 HMAC KMS 金鑰，請新增值為 true 的 MultiRegion 參數。建立 KMS 金鑰之後即無法變更這些屬性。

此CreateKey作業不允許您指定別名，但您可以使用此CreateAlias作業為新 KMS 金鑰建立別名。建議您使用能將 KMS 金鑰識別為 HMAC 金鑰的別名，例如 HMAC/test-key。這將讓您更容易識別 AWS KMS 主控台中的 HMAC 金鑰，您可以依別名對金鑰進行排序和篩選，但不能依金鑰規格或金鑰用途進行排序和篩選。

如果您嘗試在某個 AWS 區域 中建立 HMAC KMS 金鑰，而這個區域不支援 HMAC 金鑰時，則 CreateKey 操作會傳回 UnsupportedOperationException。

下列範例會使用 CreateKey 操作建立 512 位元的 HMAC KMS 金鑰。

```
$ aws kms create-key --key-spec HMAC_512 --key-usage GENERATE_VERIFY_MAC
{
    "KeyMetadata": {
        "KeyState": "Enabled",
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "KeyManager": "CUSTOMER",
        "Description": "",
        "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "CreationDate": 1669973196.214,
        "MultiRegion": false,
        "KeySpec": "HMAC_512",
        "CustomerMasterKeySpec": "HMAC_512",
        "KeyUsage": "GENERATE_VERIFY_MAC",
        "MacAlgorithms": [
            "HMAC_SHA_512"
        ],
        "AWSAccountId": "111122223333",
        "Origin": "AWS_KMS",
        "Enabled": true
    }
}
```

控制對 HMAC KMS 金鑰的存取

若要控制對 HMAC KMS 金鑰的存取，請使用金鑰政策，每個 KMS 金鑰都需要有此政策。您也可以使用 IAM 政策和授予工具。

在 AWS KMS 主控台中建立的 HMAC 金鑰預設金鑰政策，會授予金鑰使用者呼叫 GenerateMac 和 VerifyMac 操作的許可。但此政策中不包含專為使用 AWS 服務授予所設計的金鑰政策陳述式。若您使用 CreateKey 操作建立 HMAC 金鑰，則必須在金鑰政策或 IAM 政策中指定這些許可。

您可以使用 AWS 全域條件金鑰和 AWS KMS 條件金鑰來精簡及限制 HMAC 金鑰的許可。例如，您可以使用 kms:ResourceAliases 條件金鑰，依據與 HMAC 金鑰相關聯的別名，控制對 AWS KMS 操作的存取。以下 AWS KMS 政策條件對 HMAC 金鑰政策十分實用。

- 使用 kms:MacAlgorithm 條件金鑰來限制主體在呼叫 GenerateMac 和 VerifyMac 操作時可請求的演算法。例如，您可以允許主體呼叫 GenerateMac 操作，但僅限於請求中的 MAC 演算法為 HMAC_SHA_384 時。
- 使用 kms:KeySpec 條件金鑰允許或阻止主體建立特定類型的 HMAC 金鑰。例如，若要允許主參與者只建立 HMAC 金鑰，您可以允許CreateKey作業，但使用kms:KeySpec條件只允許具有金鑰規格的金HMAC_384鑰。

  您也可以使用 kms:KeySpec 條件金鑰，依據金鑰規格，控制對 KMS 金鑰上其他操作的存取。例如，您可以僅允許主體在具有 HMAC_256 金鑰規格的 KMS 金鑰上，排定和取消金鑰刪除。
- 使用 kms:KeyUsage 條件金鑰允許或阻止主體建立任何 HMAC 金鑰。例如，若要允許主參與者只建立 HMAC 金鑰，您可以允許CreateKey作業，但使用kms:KeyUsage條件只允許使用金鑰的GENERATE_VERIFY_MAC金鑰。

  您也可以使用 kms:KeyUsage 條件金鑰，依據金鑰用途，控制對 KMS 金鑰上其他操作的存取。例如，您可以僅允許主體在具有 GENERATE_VERIFY_MAC 金鑰用途的 KMS 金鑰上進行啟用和停用。

您也可以為 GenerateMac 和 VerifyMac 操作建立授予，這兩個操作是授予操作。然而，您不能在 HMAC 金鑰的授予中使用加密內容授予限制條件。HMAC 標籤格式不支援加密內容值。

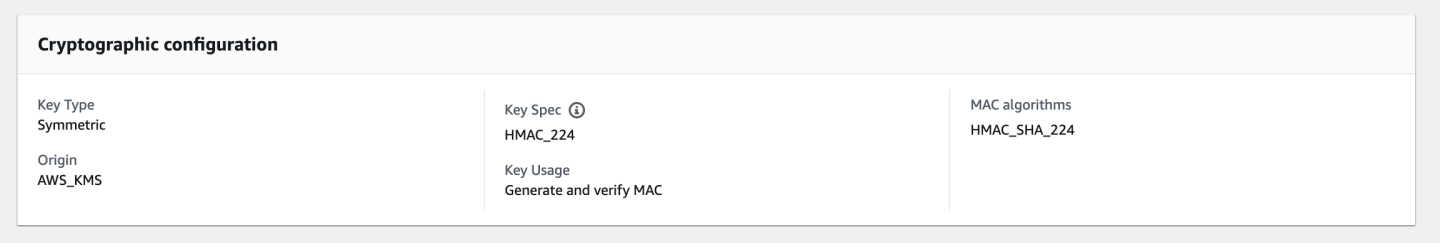
檢視 HMAC KMS 金鑰

您可以在 AWS KMS 主控台或使用 DescribeKey API 檢視 HMAC KMS 金鑰。您可以監控 HMAC KMS 金鑰在AWS CloudTrail日誌和 Amazon CloudWatch 中的使用情況。如需有關如何檢視 KMS 金鑰的基本指示，請參閱檢視金鑰。

您可以依據金鑰規格 (以 HMAC 開頭) 或是金鑰用途 (一律為 Generate and verify MAC (產生和驗證 MAC) (GENERATE_VERIFY_MAC))，來區分 HMAC KMS 金鑰與其他類型的 KMS 金鑰。

HMAC KMS 金鑰包含在 AWS KMS 主控台 Customer managed keys (客戶受管金鑰) 頁面上的資料表中。但是，您無法依金鑰規格或金鑰用途來排序或篩選 KMS 金鑰。若要更容易地找到您的 HMAC 金鑰，請為其指派一個獨特的別名或標籤。然後，您便可以依別名或標籤進行排序或篩選。

在 HMAC KMS 金鑰的金鑰詳細資訊頁面，查看 Cryptographic configuration (密碼編譯組態) 索引標籤便可找到組態詳細資訊。



AWS KMS 中的多區域金鑰

AWS KMS 支援多區域金鑰，這是在不同 AWS 區域 中的 AWS KMS keys，可以互換使用 – 如同在多個區域中有相同的金鑰。每組相關的多區域金鑰具有相同的金鑰材料和金鑰 ID，因此您可以在一個 AWS 區域 中加密資料並不同的 AWS 區域 中將其解密，而無需重新加密或跨區域呼叫 AWS KMS。

與所有 KMS 金鑰一樣，多區域金鑰永遠不會讓 AWS KMS 處於未加密狀態。您可以建立對稱或非對稱多區域金鑰以進行加密或簽署，或是建立 HMAC 多區域金鑰以產生和驗證 HMAC 標籤，或是建立含有匯入金鑰資料或 AWS KMS 產生之金鑰資料的多區域金鑰。您必須獨立管理每個多區域金鑰，包括建立別名和標籤、設定其金鑰政策和授予，以及選擇性地將其啟用和停用。您可以在所有可以使用單一區域金鑰執行的密碼編譯操作中使用多區域金鑰。

多區域金鑰是靈活且強大的解決方案，適用於許多常見的資料安全案例。

災難復原

在備份與復原架構中，多區域金鑰可讓您在不中斷的情況下處理加密資料，即使在 AWS 區域 中斷的情形下。備份區域中維護的資料可以在備份區域中解密，備份區域中新加密的資料可以在主要區域 (當該區域還原時) 中解密。

全域資料管理

全球營運的企業需要全球分散式資料，這些資料可一致地跨 AWS 區域 提供。您可以在資料所在的所有區域中建立多區域金鑰，然後將金鑰當作單一區域金鑰使用，以避免跨區域呼叫的延遲，或是在每個區域中不同金鑰下重新加密資料的成本。

分散式簽署應用程式

需要跨區域簽章功能的應用程式可以使用多區域非對稱簽署金鑰，在不同 AWS 區域 中一致且重複地產生相同的數位簽章。

如果您將憑證鏈結與單一全域信任存放區 (針對單一根憑證授權機構 (CA)) 和根 CA 簽署的區域中繼 CA，則不需要多區域金鑰。不過，如果您的系統不支援中繼 CA (例如應用程式簽署)，則可以使用多區域金鑰來為區域認證提供一致性。

跨越多個區域的主動-主動應用程式

某些工作負載和應用程式可以跨越主動-主動架構中的多個區域。對於這些應用程式，透過為針對可能跨區域邊界移動之資料的同時加密和解密操作提供相同的金鑰材料，多區域金鑰可以降低複雜性。

您可以將多區域金鑰與用戶端加密程式庫搭配使用，例如 AWS Encryption SDK、DynamoDB 加密用戶端和 Amazon S3 用戶端加密。如需將多區域金鑰搭配 Amazon DynamoDB 全域資料表和 DynamoDB 加密用戶端使用的範例，請參閱 AWS 安全部落格中的使用 AWS KMS 多區域金鑰加密全域資料用戶端中的安全部落格。

用於靜態加密或數位簽章之與 AWS KMS 整合的 AWS 服務，目前將多區域金鑰視為單一區域金鑰。其可能會重新包裝或重新加密在區域之間移動的資料。例如，Amazon S3 跨區域複寫會在目的地區域的 KMS 金鑰下解密和重新加密資料，即使複寫受多區域金鑰保護的物件也一樣。

多區域金鑰不適用於全域。您建立多區域主要金鑰，然後將其複寫到您在 AWS 分割區內選取的區域中。然後，您可以獨立管理每個區域中的多區域金鑰。AWS 或 AWS KMS 不會代表您自動建立或複寫多區域金鑰到任何區域。AWS 受管金鑰 (AWS 服務在您的帳戶中為您建立的 KMS 金鑰) 永遠是單一區域金鑰。

您無法將現有的單一區域金鑰轉換為多區域金鑰。此設計可確保所有受現有單一區域金鑰保護的資料都維持相同的資料落地和資料主權屬性。

對於大部分的資料安全需求，區域資源的區域隔離和容錯能力讓標準 AWS KMS 單一區域金鑰成為最適合的解決方案。不過，當您需要跨多個區域加密或簽署用戶端應用程式中的資料時，多區域金鑰可能就是解決方案。

區域

在所有 AWS KMS 支援的 AWS 區域 中，多區域金鑰均受支援，除了中國 (北京) 和中國 (寧夏) 以外。

定價和配額

一組相關多區域金鑰中的每個金鑰都會計為一個 KMS 金鑰，用於定價和配額。AWS KMS 配額會針對帳戶的每個區域個別計算。在每個區域中使用和管理多區域金鑰會計為該區域的配額。

支援的 KMS 金鑰類型

您可以建立以下類型的多區域 KMS 金鑰：

- 對稱加密 KMS 金鑰
- 非對稱 KMS 金鑰
- HMAC KMS 金鑰
- 包含匯入金鑰資料的 KMS 金鑰

您無法在自訂金鑰存放區建立多區域金鑰。

主題

- 控制對多區域金鑰的存取
- 建立多區域金鑰
- 檢視多區域金鑰
- 管理多區域金鑰
- 將金鑰資料匯入多區域金鑰
- 刪除多區域金鑰

多區域金鑰的安全考量

只有在需要時才能使用 AWS KMS 多區域金鑰。多區域金鑰為在 AWS 區域 之間移動加密資料的工作負載或需要跨區域存取的工作負載提供靈活且可擴展的解決方案。如果您必須跨區域共用、移動或備份受保護的資料，或者需要建立在不同區域中運作之應用程式的相同數位簽章，請考慮使用多區域金鑰。

不過，建立多區域金鑰的程序會在 AWS KMS 內跨 AWS 區域 邊界移動您的金鑰材料。由多區域金鑰產生的加密文字可能會由多個地理位置中的多個相關金鑰進行解密。區域隔離的服務和資源也有顯著的優勢。每個 AWS 區域 都是隔離的，且與其他區域各自獨立。區域提供容錯能力、穩定性和恢復能力，也可降低延遲。它們可讓您建立冗餘資源，這些資源會保持可用且不受其他區域運行中斷的影響。在 AWS KMS 中，它們還會確保每份加密文字只能透過一個金鑰進行解密。

多區域金鑰也會引發新的安全考量：

- 使用多區域金鑰，控制存取和強制執行資料安全政策會比較複雜。您需要確保會在多個隔離區域的金鑰上對政策進行一致地稽核。而且您需要使用政策來強制執行邊界，而不是依賴單獨的金鑰。

  例如，您需要針對資料設定政策條件，以防止一個區域的薪資團隊能夠讀取不同區域的薪資資料。此外，您必須使用存取控制來防止出現以下案例：一個區域中的多區域金鑰保護一個租用戶的資料，而另一個區域中的相關多區域金鑰保護不同租用戶的資料。
- 跨區域稽核金鑰也比較複雜。使用多區域索引鍵，您需要檢查和協調多個區域的稽核活動，以完全了解受保護資料的金鑰活動。
- 符合資料落地規定可能會更加複雜。透過隔離區域，您可以確保資料落地和資料主權的合規性。指定區域中的 KMS 金鑰只能解密該區域中的敏感資料。在一個區域中加密的資料可以在任何其他區域中保持處於完全保護且無法存取的狀態。

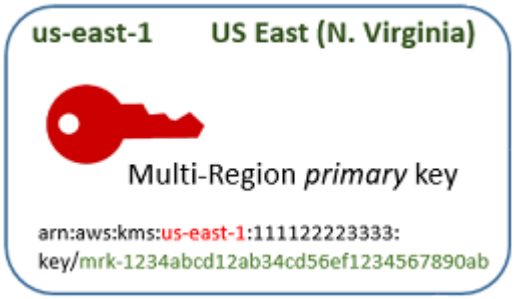
  若要使用多區域金鑰驗證資料落地和資料主權，您需要實作存取政策並編譯跨多個區域編譯 AWS CloudTrail 活動。

為了讓您更輕鬆地管理多區域金鑰的存取控制，複寫多區域金鑰 (kms: ReplicateKey) 的權限與建立金鑰的標準權限 (kms: CreateKey) 不同。另外，AWS KMS 支援多個區域金鑰的政策條件，包括 `kms:MultiRegion`，以允許或拒絕建立、使用或管理多區域金鑰和 `kms:ReplicaRegion` 的許可，將區域限制為可以複寫多區域金鑰的區域。如需詳細資訊，請參閱 控制對多區域金鑰的存取。

多區域金鑰的運作方式

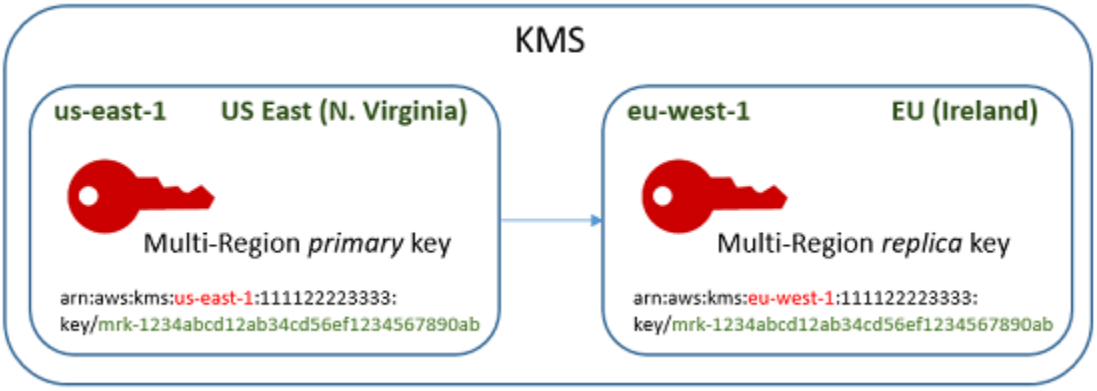
您首先在 AWS KMS 支援的 AWS 區域 中建立對稱或非對稱多區域主要金鑰，例如美國東部 (維吉尼亞北部)。只有在建立金鑰時，您才會決定是單一區域金鑰還是多區域金鑰；之後就無法變更此屬性。與任何 KMS 金鑰一樣，您可以設定多區域金鑰的金鑰政策，並建立授予，以及新增分類和授權的別名和標籤。(這些是獨立屬性，並未與其他金鑰共用或同步。) 您可以在密碼編譯操作中使用多區域主要金鑰進行加密或簽署。

您可以在主AWS KMS控台中建立多區域主索引鍵，或使用 CreateKeyAPI 並將`MultiRegion`參數設定為`true`。請注意，多區域金鑰具有開頭為 `mrk-` 的獨特金鑰 ID。您可以使用 `mrk-` 字首，以程式設計方式識別 MRK。



如果選擇，則您可以將多區域主要金鑰複寫為相同 AWS 分割區中的一個或多個不同 AWS 區域，例如歐洲 (愛爾蘭)。如此操作時，AWS KMS 會在具有與主要金鑰相同之金鑰 ID 和其他共用屬性的指定區域中建立複本金鑰。然後，它會安全地跨越區域邊界傳輸金鑰材料，並將其與目的地區域中的新 KMS 金鑰相關聯，全部在 AWS KMS 範圍內。其結果是兩個相關多區域金鑰 (主要金鑰和複本金鑰) 可以互換使用。

您可以在AWS KMS主控台或使用 ReplicateKeyAPI 建立多區域複本金鑰。



由此產生的多區域複本金鑰是功能齊全的 KMS 金鑰，具有與主要金鑰相同的共用屬性。在所有其他方面，它是獨立的 KMS 金鑰，具有自己的描述、金鑰政策、授予、別名和標籤。啟用或停用多區域金鑰不會影響相關的多區域金鑰。您可以在密碼編譯操作中獨立使用主要金鑰和複本金鑰，或協調其使用。例如，您可以使用美國東部 (維吉尼亞北部) 區域的主要金鑰來加密資料、將資料移至歐洲 (愛爾蘭) 區域，然後使用複本金鑰來解密資料。

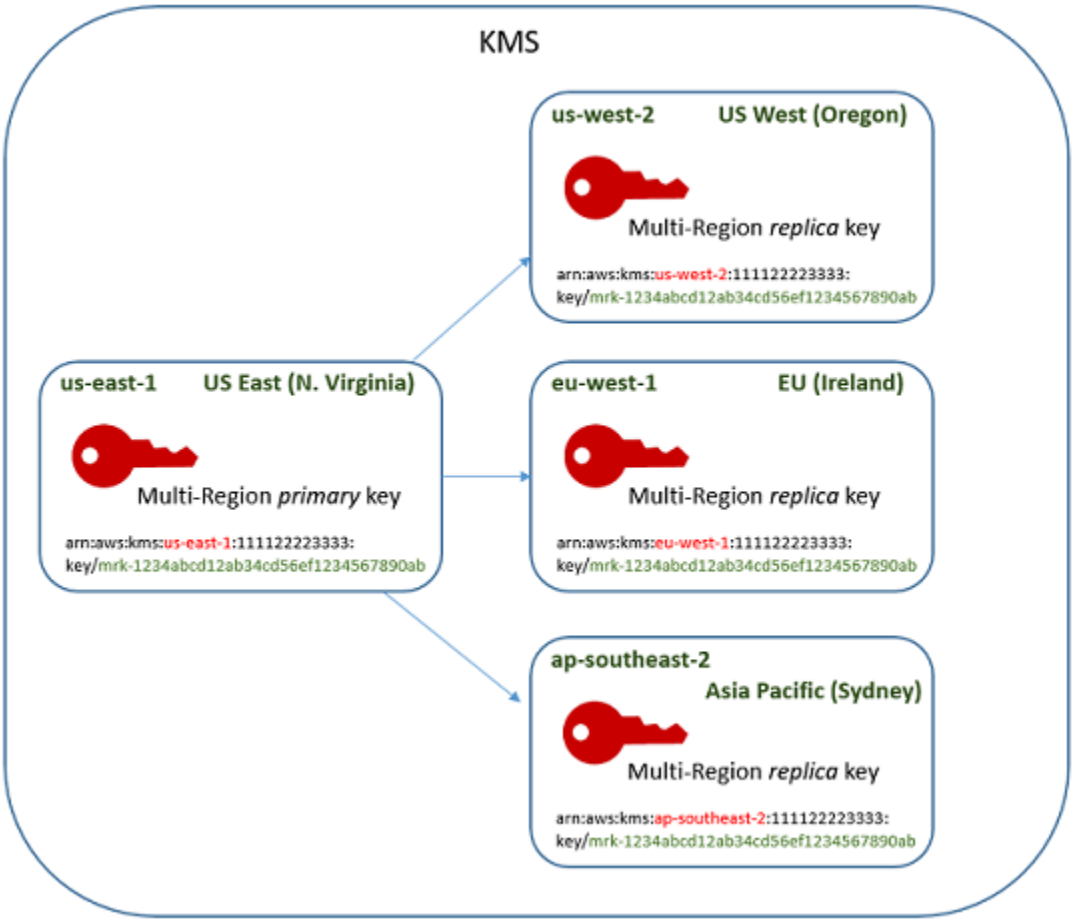
相關的多區域金鑰具有相同的金鑰 ID。其金鑰 ARN (Amazon Resource Name) 僅在區域字段中不同。例如，多區域主要金鑰和複本金鑰可能具有下列範例金鑰 ARN。金鑰 ID (金鑰 ARN 中的最後一個元素) 是相同的。兩個金鑰都有多區域金鑰的獨特金鑰 ID，開頭為 mrk-。

```
Primary key: arn:aws:kms:us-
east-1:111122223333:key/mrk-1234abcd12ab34cd56ef12345678990ab
Replica key: arn:aws:kms:eu-
west-1:111122223333:key/mrk-1234abcd12ab34cd56ef12345678990ab
```

具有相同的金鑰 ID 才能交互操作。加密時，AWS KMS 會將 KMS 金鑰的金鑰 ID 繫結至加密文字，因此只能使用該 KMS 金鑰或具有相同金鑰 ID 的 KMS 金鑰來解密加密文字。這項功能也讓相關的多區域金鑰易於識別，並且可以更輕鬆地對其進行互換使用。例如，在應用程式中使用時，您可以透過其共用金鑰 ID 來引用相關的多區域金鑰。然後，如有必要，請指定區域或 ARN 來區分這些金鑰。

您的資料需要變更時，您可以將主要金鑰複寫至位於相同分割區中的其他 AWS 區域，例如美國西部 (奧勒岡) 和亞太區域 (雪梨)。其結果是具有相同金鑰材料和金鑰 ID 的四個相關多區域金鑰，如下圖所

示。您可以獨立管理金鑰。您可以獨立或以協調的方式使用這些金鑰。例如，您可以使用亞太區域 (雪梨) 中的複本金鑰來加密資料、將資料移至美國西部 (奧勒岡)，然後使用美國西部 (奧勒岡) 的複本金鑰來解密資料。



多區域金鑰的其他考量包括下列各項。

同步共用屬性 – 如果多區域金鑰的共用屬性發生變更，AWS KMS 會自動將主要金鑰的變更同步至其所有複本金鑰。您無法請求或強制執行共用屬性同步。AWS KMS 會為您偵測並同步所有變更。不過，您可以使用 CloudTrail 記錄檔中的SynchronizeMultiRegionKey事件來稽核同步處理。

例如，如果您在對稱多區域主要金鑰上啟用自動金鑰輪換，則 AWS KMS 會將該設定複製到其所有複本金鑰。輪換金鑰材料時，所有相關多區域金鑰之間的輪換會同步，因此其仍然具有相同的當前金鑰材料，並存取所有舊版的金鑰材料。如果您建立新的複本金鑰，則該金鑰具有與所有相關多區域金鑰相同的當前金鑰材料，並可存取所有舊版金鑰材料。如需詳細資訊，請參閱 輪換多區域金鑰。

變更主要金鑰 – 每組多區域金鑰必須只有一個主要金鑰。主要金鑰是唯一可以複寫的金鑰。它也是其複本金鑰共用屬性的來源。但是您可以將主要金鑰變更為複本，並將其中一個複本金鑰升級為主要金

鑰。您可以這樣做，以便您可以從特定區域刪除多區域主要金鑰，或者在更接近專案管理員的區域找到主要金鑰。如需詳細資訊，請參閱 更新主要區域。

刪除多區域金鑰 – 與所有 KMS 金鑰一樣，您必須在 AWS KMS 將其刪除之前排程刪除多區域金鑰。當金鑰正在等待刪除時，您無法在任何密碼編譯操作中使用金鑰。然而，AWS KMS 不會刪除多區域主要金鑰，直到刪除其所有複本金鑰為止。如需詳細資訊，請參閱 刪除多區域金鑰。

概念

下列術語和概念與多區域金鑰搭配使用。

多區域金鑰

多區域金鑰是在不同 AWS 區域 中具有相同金鑰 ID 和金鑰材料 (以及其他共用屬性) 的一組 KMS 金鑰之一。每個多區域金鑰都是功能完善的 KMS 金鑰，可完全獨立於其相關的多區域金鑰之外使用。由於所有相關多區域金鑰都具有相同的金鑰 ID 和金鑰材料，所以這些金鑰是互通的，即任何 AWS 區域 中任何相關的多區域金鑰都可以透過任何其他相關的多區域金鑰解密已加密的文字。

您在建立 KMS 金鑰時會設定其多區域屬性。您無法在現有金鑰上變更多區域屬性。您無法將單一區域金鑰轉換為多區域金鑰，或將多區域金鑰轉換為單一區域金鑰。若要將現有的工作負載移至多區域案例，您必須重新加密資料，或使用新的多區域金鑰建立新的簽章。

多區域金鑰可以是對稱或非對稱的，其可以使用 AWS KMS 金鑰材料或匯入的金鑰材料。您無法在自訂金鑰存放區建立多區域金鑰。

在一組相關的多區域金鑰中，任何時候都只有一個主要金鑰。您可以在其他 AWS 區域 建立該主要金鑰的複本金鑰。您也可以更新主要區域，它會將主要金鑰變更為複本金鑰，並將指定的複本金鑰變更為主要金鑰。不過，您在每個 AWS 區域 只能維護一個主要金鑰或複本金鑰。所有區域必須在相同的 AWS 分割區中。

您可以在相同或不同的 AWS 區域 中擁有多組相關的多區域金鑰。雖然相關的多區域金鑰可互通操作，但不相關的多區域金鑰無法互通操作。

主索引鍵

多區域主要金鑰是 KMS 金鑰，可以複寫到相同分割區的其他 AWS 區域 中。每組多區域金鑰只有一個主要金鑰。

主要金鑰與複本金鑰有下列幾點不同之處：

- 只有主要金鑰可以複寫。

- 主要金鑰是其複本金鑰之共用屬性的來源，包括金鑰資料和金鑰 ID。
- 您可以僅針對主要金鑰啟用和停用自動金鑰輪換。
- 您可以隨時排程刪除主要金鑰。但 AWS KMS 不會刪除主要金鑰，直到刪除其所有複本金鑰為止。

不過，主要和複本金鑰在任何密碼編譯屬性中都沒有差異。您可以互換使用主要金鑰及其複本金鑰。

您不需要複寫主要金鑰。您可以像使用任何 KMS 金鑰一樣使用它，並在有用時對其進行複寫。不過，由於多區域金鑰與單一區域金鑰具有不同的安全屬性，建議您只在計劃複寫時建立多區域金鑰。

複本金鑰

多區域複本金鑰是 KMS 金鑰，具有與其主要金鑰和相關複本金鑰相同的金鑰 ID 和金鑰材料，但存在於不同的 AWS 區域 中。

複本金鑰是功能完善的 KMS 金鑰，具有自己的金鑰政策、授予、別名、標籤和其他屬性。它不是主要金鑰或任何其他金鑰的指標複本。您可以使用複本金鑰，即使其主要金鑰和所有相關的複本金鑰已停用。您也可以將複本金鑰轉換為主要金鑰，將主要金鑰轉換為複本金鑰。建立後，複本金鑰僅依賴其主要金鑰進行金鑰輪換和更新主要區域。

主要和複本金鑰在任何密碼編譯屬性中都沒有差異。您可以互換使用主要金鑰及其複本金鑰。透過主要金鑰或複本金鑰加密的資料可以透過相同的金鑰或任何相關的主要金鑰或複本金鑰進行解密。

複寫

您可以將多區域主要金鑰複寫至相同分割區的不同 AWS 區域 中。如此操作時，AWS KMS 會在具有與主要金鑰相同之金鑰 ID 和其他共用屬性的指定區域中建立多區域複本金鑰。然後，它會安全地跨越區域邊界傳輸金鑰材料，並將其與新複本金鑰相關聯，全部在 AWS KMS 範圍內。

共用屬性

共用屬性是與其複本金鑰共用之多區域主要金鑰的屬性。AWS KMS 會建立具有與這些主要金鑰相同之共用屬性值的複本金鑰。然後，它會定期將主要金鑰的共用屬性值同步至其複本金鑰。您無法在複本金鑰上設定這些屬性。

以下是多區域金鑰的共用屬性。

- 金鑰 ID – (金鑰 ARN 的 `Region` 元素有所不同。)
- 金鑰資料
- 金鑰資料來源

- 金鑰規格和加密演算法
- 金鑰用途
- 自動金鑰輪換 – 您可以僅針對主要金鑰啟用和停用自動金鑰輪換。使用共用金鑰材料的所有版本建立新的複本金鑰。如需詳細資訊，請參閱 輪換多區域金鑰。

您也可以將相關多區域金鑰的主要和複本指定視為共用屬性。當您建立新的複本金鑰或更新主要金鑰時，AWS KMS 會將變更同步至所有相關的多區域金鑰。完成這些變更後，所有相關的多區域金鑰都會準確地列出其主要金鑰和複本金鑰。

多區域金鑰的所有其他屬性都是獨立屬性，包括描述、金鑰政策、授予、已啟用和已停用的金鑰狀態、別名，以及標籤。您可以在所有相關的多區域金鑰上為這些屬性設定相同的值，但是如果您變更獨立屬性的值，則 AWS KMS 不會將其同步。

您可以追蹤多區域金鑰之共用屬性的同步。在記AWS CloudTrail錄檔中尋找SynchronizeMultiRegionKey事件。

控制對多區域金鑰的存取

您可以在合規、災難復原和備份案例中使用多區域金鑰，而單一區域金鑰會比較複雜。不過，由於多區域金鑰的安全屬性與單一區域金鑰的安全屬性有很大的不同，所以建議在授權建立、管理和使用多區域金鑰時謹慎使用。

> **Note**
>
> 在 `Resource` 欄位包含萬用字元的現有 IAM 政策陳述式，現在會同時套用至單一區域和多區域金鑰。若要將它們限制為單一區域 KMS 金鑰或多區域金鑰，請使用 kms: MultiRegion 條件金鑰。

使用您的授權工具來防止在任何單一區域足夠的情況下建立和使用多區域金鑰。允許委託人只將多區域金鑰複寫至需要的 AWS 區域 中。只將多區域金鑰的許可授予需要這些金鑰的委託人，並且僅為需要這些金鑰的任務授予。

您可以使用金鑰政策、IAM 政策和授權來允許 IAM 委託人管理和使用 AWS 帳戶 中的多區域金鑰。每個多區域金鑰都是具有唯一金鑰 ARN 和金鑰政策的獨立資源。您需要為每個金鑰建立和維護金鑰政策，並確保新的和現有的 IAM 政策會實作您的授權策略。

主題

- 多區域金鑰的授權基礎知識
- 授權多區域主要管理員和使用者
- 授權 AWS KMS 同步多區域金鑰

多區域金鑰的授權基礎知識

針對多區域金鑰設計金鑰政策和 IAM 政策時，請考慮下列政策。

- 金鑰政策 – 每個多區域金鑰都是獨立的 KMS 金鑰資源，具有其自己的金鑰政策。您可以將相同或不同的金鑰政策套用至相關多區域金鑰集合中的每個金鑰。金鑰政策不是多區域金鑰的共用屬性。AWS KMS 不會在相關的多區域金鑰之間複製或同步金鑰政策。

  當您在 AWS KMS 主控台建立複本金鑰時，主控台會顯示主要金鑰的目前金鑰政策，以方便使用。您可以使用此金鑰政策，加以編輯，或刪除並取代。但即使您接受未變更的主要金鑰政策，AWS KMS 也不會同步政策。例如，如果您變更主要金鑰的金鑰政策，複本金鑰的金鑰政策會保持不變。
- 預設金鑰原則 — 使用CreateKey和ReplicateKey作業建立多區域金鑰時，除非您在要求中指定金鑰原則，否則會套用預設金鑰原則。這與套用至單一區域金鑰的預設金鑰政策相同。
- IAM 政策 – 與所有 KMS 金鑰一樣，您可以使用 IAM 政策來控制多區域金鑰的存取，僅當金鑰政策允許時。IAM 政策預設會套用至所有 AWS 區域。不過，您可以使用諸如 aws: RequestedRegion 之類的條件金鑰來限制特定區域的許可。

  若要建立主要金鑰和複本金鑰，委託人必須在 IAM 政策中具有 `kms:CreateKey` 許可，該許可會套用至建立金鑰的區域。
- 授予 – AWS KMS 授予是區域性的。每個授予都允許一個 KMS 金鑰的許可。您可以使用授予來允許多區域主要金鑰或複本金鑰的許可。但是，您無法使用單一授予來允許多個 KMS 金鑰的許可，即使這些金鑰是相關的多區域金鑰。
- 金鑰 ARN – 每個多區域金鑰都有唯一金鑰 ARN。相關多區域金鑰的金鑰 ARN 具有相同的分割區、帳戶和金鑰 ID，但區域不同。

  若要將 IAM 政策陳述式套用至特定的多區域金鑰，請使用其金鑰 ARN 或包含區域的金鑰 ARN 模式。若要將 IAM 政策陳述式套用到所有相關的多區域金鑰，請在 ARN 的區域元素中使用萬用字元 (*)，如下列範例所示。

```
{
  "Effect": "Allow",
  "Action": [
    "kms:Describe*",
```

```
    "kms:List*"
  ],
  "Resource": {
      "arn:aws:kms:*::111122223333:key/mrk-1234abcd12ab34cd56ef1234567890ab"
  }
}
```

若要將原則陳述式套用至您中的所有多區域金鑰AWS 帳戶，您可以使用 kms: MultiRegion 原則條件或包含獨特mrk-前置詞的金鑰 ID 模式。

- 服務連結角色 — 建立多區域主索引鍵的主體必須具有 iam: 權限。CreateServiceLinkedRole

  若要同步相關多區域金鑰的共用屬性，AWS KMS 會擔任 IAM 服務連結角色。每當您建立多區域主要金鑰時，AWS KMS 會在 AWS 帳戶 中建立服務連結角色。(如果角色存在，AWS KMS 會重新建立角色，這沒有任何有害的影響。) 角色在所有區域都有效。若AWS KMS要允許建立 (或重新建立) 服務連結角色，建立多區域主要索引鍵的主體必須具有 iam: 權限。CreateServiceLinkedRole

授權多區域主要管理員和使用者

建立和管理多區域金鑰的委託人需要在主要區域和複本區域中的下列許可：

- kms:CreateKey
- kms:ReplicateKey
- kms:UpdatePrimaryRegion
- iam:CreateServiceLinkedRole

建立主要金鑰

若要建立多區域主要金鑰，主體需要在主金鑰區域中有效的 IAM 政策中的 kms: CreateKey 和 iam: CreateServiceLinkedRole 許可。擁有這些許可的委託人可以建立單一區域和多區域金鑰，除非您限制其許可。

此iam:CreateServiceLinkedRole權限允許AWS KMS建立AWSServiceRoleForKeyManagementServiceMultiRegionKeys角色，以同步處理相關多區域金鑰的共用內容。

例如，此 IAM 政策允許委託人建立任何類型的 KMS 金鑰。

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

```
  "Statement":{
      "Action": [
        "kms:CreateKey",
        "iam:CreateServiceLinkedRole"
      ],
      "Effect":"Allow",
      "Resource":"*"
  }
}
```

若要允許或拒絕建立多區域主索引鍵的權限，請使用 kms: MultiRegion 條件金鑰。有效值為 `true` (多區域金鑰) 或 `false` (單一區域金鑰)。例如，下列 IAM 政策陳述式會使用具有 `kms:MultiRegion` 條件索引鍵的 `Deny` 動作，以防止委託人建立多區域金鑰。

```
{
  "Version": "2012-10-17",
  "Statement":{
      "Action":"kms:CreateKey",
      "Effect":"Deny",
      "Resource":"*",
      "Condition": {
          "Bool": "kms:MultiRegion": true
      }
  }
}
```

複寫金鑰

若要建立多區域複本金鑰，則委託人需要下列許可：

- kms：主鍵的密鑰策略中的ReplicateKey權限。
- kms：在複本金鑰區域中有效的 IAM 政策中的CreateKey權限。

允許這些許可時請小心謹慎。其允許委託人建立 KMS 金鑰，以及授權其使用的金鑰政策。`kms:ReplicateKey` 許可還授權在 AWS KMS 範圍內跨區域邊界傳輸金鑰材料。

若要限制可複寫多區域金鑰的項目，請使用 kms: ReplicaRegion 條件金鑰。AWS 區域它只限制 `kms:ReplicateKey` 許可。否則，它沒有影響。例如，下列金鑰政策允許委託人複寫該主要金鑰，但僅限於指定區域中。

```
{
```

```
  "Effect": "Allow",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/Administrator"
  },
  "Action": "kms:ReplicateKey",
  "Resource": "*",
  "Condition": {
    "StringEquals": {
      "kms:ReplicaRegion": [
         "us-east-1",
         "eu-west-3",
         "ap-southeast-2"
      ]
    }
  }
}
```

更新主要區域

授權的委託人可以將複本金鑰轉換為主要金鑰，這會將先前的主要金鑰變更為複本。這個動作稱為更新主要區域。若要更新主要區域，主體需要兩個區域的 kms: UpdatePrimaryRegion 權限。您可以在金鑰政策或 IAM 政策中提供這些許可。

- 主要金鑰上的 kms:UpdatePrimaryRegion。此許可必須在主要金鑰區域有效。
- 複本金鑰上的 kms:UpdatePrimaryRegion。此許可必須在複本金鑰區域中生效。

例如，下列金鑰政策會為可以擔任管理員角色的使用者提供許可，以更新 KMS 金鑰的主要區域。此 KMS 金鑰可以是此操作中的主要金鑰或複本金鑰。

```
{
  "Effect": "Allow",
  "Resource": "*",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/Administrator"
  },
  "Action": "kms:UpdatePrimaryRegion"
}
```

若要限制AWS 區域可託管主索引鍵的項目，請使用 kms: PrimaryRegion 條件金鑰。例如，下列 IAM 政策陳述式允許委託人更新 AWS 帳戶 中多區域金鑰的主要區域，但僅當新的主要區域是指定區域之一時。

```
{
  "Effect": "Allow",
  "Action": "kms:UpdatePrimaryRegion",
  "Resource": {
      "arn:aws:kms:*:111122223333:key/*"
  },
  "Condition": {
    "StringEquals": {
      "kms:PrimaryRegion": [
         "us-west-2",
         "sa-east-1",
         "ap-southeast-1"
      ]
    }
  }
}
```

使用和管理多區域金鑰

根據預設，在 AWS 帳戶 和區域中擁有使用和管理 KMS 金鑰之許可的委託人也具有使用和管理多區域金鑰的許可。不過，您可以使用 kms: MultiRegion 條件金鑰僅允許單一區域金鑰或僅允許多區域金鑰。或者使用 kms: MultiRegionKeyType 條件鍵僅允許多區域主索引鍵或僅允許複本金鑰。這兩個條件金鑰都會控制對CreateKey作業的存取以及使用現有 KMS 金鑰的任何作業的存取，例如 Encrypt 或 EnableKey.

下列範例 IAM 政策陳述式使用 `kms:MultiRegion` 條件索引鍵，以防止委託人使用或管理任何多區域金鑰。

```
{
  "Effect": "Deny",
  "Action": "kms:*",
  "Resource": "*",
  "Condition": {
    "Bool": "kms:MultiRegion": true
  }
}
```

此範例 IAM 政策陳述式使用 `kms:MultiRegionKeyType` 條件，以允許委託人排程和取消金鑰刪除，但僅限於多區域複本金鑰。

```
{
```

```
  "Effect": "Allow",
  "Action": [
    "kms:ScheduleKeyDeletion",
    "kms:CancelKeyDeletion"
  ],
  "Resource": {
      "arn:aws:kms:us-west-2:111122223333:key/*"
  },
  "Condition": {
    "StringEquals": "kms:MultiRegionKeyType": "REPLICA"
  }
}
```

授權 AWS KMS 同步多區域金鑰

若要支援多區域金鑰，AWS KMS 會使用 IAM 服務連結的角色。這個角色為 AWS KMS 提供其所需的許可，以同步共用屬性。您可以檢視記錄在SynchronizeMultiRegionKey CloudTrail 記錄AWS CloudTrail檔中AWS KMS同步處理共用屬性的事件。

關於多區域金鑰的服務連結角色

服務連結角色是 IAM 角色，提供許可給一個 AWS 服務代表您呼叫其他 AWS 服務。目的是讓您輕鬆使用多個整合的 AWS 服務的功能，而不需要建立和維護複雜的 IAM 政策。

對於多區域金鑰，請使用原則AWS KMS建立AWSServiceRoleForKeyManagementServiceMultiRegionKeys服務連結角色。AWSKeyManagementServiceMultiRegionKeysServiceRolePolicy此政策為角色提供 `kms:SynchronizeMultiRegionKey` 許可，允許其同步多區域金鑰的共用屬性。

由於AWSServiceRoleForKeyManagementServiceMultiRegionKeys服務連結角色只信任`mrk.kms.amazonaws.com`，因此只AWS KMS能擔任此服務連結角色。此角色僅限於 AWS KMS 需要同步多區域共用屬性的操作。並不授予 AWS KMS 任何額外的許可。例如：AWS KMS 沒有建立、複寫或刪除任何 KMS 金鑰的許可。

如需 AWS 服務如何使用服務連結角色的詳細資訊，請參閱《IAM 使用者指南》中的使用服務連結角色。

建立服務連結角色

AWS KMS當您建立多地區金鑰AWS 帳戶時，如果角色尚未存在，則會自動在您的中建立AWSServiceRoleForKeyManagementServiceMultiRegionKeys服務連結角色。您無法直接建立或重新建立此服務連結角色。

編輯服務連結角色描述

您無法編輯AWSServiceRoleForKeyManagementServiceMultiRegionKeys服務連結角色中的角色名稱或原則陳述式，但可以編輯角色描述。如需相關說明，請參閱《IAM 使用者指南》中的編輯服務連線角色。

刪除服務連結角色

AWS KMS不會刪除您的AWSServiceRoleForKeyManagementServiceMultiRegionKeys服務連結角色 AWS 帳戶，也無法將其刪除。但是，除非您的AWS 帳戶和區域中有多區域金鑰，否則不AWS KMS會擔任該AWSServiceRoleForKeyManagementServiceMultiRegionKeys角色或使用其任何權限。

建立多區域金鑰

您可以在主控台或使用 AWS KMS API 建立多區域金鑰。

您在此程序中設定的多區域屬性是不可變的。您無法將單一區域金鑰轉換為多區域金鑰，或將多區域金鑰轉換為單一區域金鑰。

主題

- 建立多區域主要金鑰
- 建立多區域複本金鑰

建立多區域主要金鑰

您可以在 AWS KMS 主控台或使用 AWS KMS API 建立多區域主要金鑰。您可以在任何 AWS KMS 支援多區域金鑰的 AWS 區域 中建立主要金鑰。

若要建立多區域主要金鑰，主體需要與建立任何 KMS 金鑰所需的相同權限，包括 IAM 政策中的 kms: CreateKey 權限。主體還需要 iam: CreateServiceLinkedRole 許可。您可以使用 kms: MultiRegionKeyType 條件金鑰來允許或拒絕建立多區域主索引鍵的權限。

這些指示會建立具有 AWS KMS 產生之金鑰材料的多區域主要金鑰。若要建立具有匯入金鑰材料的多區域主要金鑰，請參閱 建立具有匯入金鑰材料的主要金鑰。

主題

- 建立多區域主要金鑰 (主控台)
- 建立多區域主要金鑰 (AWS KMS API)

建立多區域主要金鑰 (主控台)

若要在 AWS KMS 主控台建立多區域主要金鑰，請使用您要用來建立任何 KMS 金鑰的相同程序。您在 Advanced options (進階選項) 中選取多區域金鑰。如需完整說明，請參閱 建立金鑰。

> **Important**
>
> 請勿在別名、說明或標籤包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 選擇建立金鑰。
5. 選取對稱或非對稱金鑰類型。對稱金鑰為預設值。

   您可以建立多區域對稱和非對稱金鑰，包括多區域 HMAC KMS 金鑰，這些都是對稱金鑰。
6. 選擇您的金鑰用途。Encrypt and decrypt (加密和解密) 為預設值。

   如需相關說明，請參閱the section called “建立金鑰”、the section called “建立非對稱 KMS 金鑰”或the section called “建立 HMAC 金鑰”。
7. 展開 Advanced options (進階選項)。
8. 在 Key material origin (金鑰材料來源) 下，AWS KMS 會產生您的主要金鑰和複本金鑰共用的金鑰材料，請選擇 KMS。若您要將金鑰材料匯入主要金鑰和複本金鑰中，請選擇 External (Import key material) (外部 (匯入金鑰材料))。
9. 在 Multi-Region replication (多區域複寫) 下，選擇 Allow this key to be replicated into other Regions (允許此金鑰複寫至其他區域)。

   您無法在建立 KMS 金鑰之後變更此設定。
10. 輸入主要金鑰的別名。

    別名不是多區域金鑰的共用屬性。您可以為多區域主要金鑰及其複本提供相同的別名或不同的別名。AWS KMS 不會同步多區域金鑰的別名。

> **Note**
>
> 新增、刪除或更新別名可允許或拒絕 KMS 金鑰的許可。如需詳細資訊，請參閱 AWS KMS 的 ABAC 和 使用別名來控制對 KMS 金鑰的存取。

11. (選用) 輸入主要金鑰的描述。

    描述不是多區域金鑰的共用屬性。您可以為多區域主要金鑰及其複本提供相同的描述或不同的描述。AWS KMS 不會同步多區域金鑰的金鑰描述。

12. (選用) 輸入標籤索引鍵和選用標籤值。若要為主要金鑰指派超過一個標籤，請選擇 Add tag (新增標籤)。

    標籤不是多區域金鑰的共用屬性。您可以為多區域主要金鑰及其複本提供相同的標籤或不同的標籤。AWS KMS 不會同步多區域金鑰的標籤。您可以隨時變更 KMS 金鑰上的標籤。

> **Note**
>
> 標記或取消標記 KMS 金鑰可以允許或拒絕 KMS 金鑰的許可。如需詳細資訊，請參閱 AWS KMS 的 ABAC 和 使用標籤來控制對 KMS 金鑰的存取。

13. 選取可管理主要金鑰的 IAM 使用者和角色。

> **Note**
>
> IAM 政策可授權其他 IAM 使用者和角色來管理 KMS 金鑰。
> IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

    這個步驟會開始為主要金鑰建立金鑰政策的程序。金鑰政策不是多區域金鑰的共用屬性。您可以為多區域主要金鑰及其複本提供相同的金鑰政策或不同的金鑰政策。AWS KMS 不會同步多區域金鑰的金鑰政策。您可以隨時變更 KMS 金鑰的金鑰政策。

14. 完成建立金鑰政策的步驟，包括選取金鑰使用者。檢閱金鑰政策後，選擇 Finish (完成) 以建立 KMS 金鑰。

建立多區域主要金鑰 (AWS KMS API)

若要建立多區域主索引鍵，請使用此CreateKey作業。使用值為 `True` 的 `MultiRegion` 參數。

例如，以下命令會在呼叫者的 AWS 區域 (us-east-1) 中建立多區域主要金鑰。它會接受所有其他屬性 (包括金鑰政策) 的預設值。多區域主要金鑰的預設值與所有其他 KMS 金鑰的預設值相同，包括預設金鑰政策。此程序會建立一個對稱加密金鑰，即預設 KMS 金鑰。

回應包含具有一般子元素的 `MultiRegion` 元素和 `MultiRegionConfiguration` 元素，以及無複本金鑰的多區域主要金鑰值。多區域金鑰的金鑰 ID 始終以 `mrk-` 開頭。

> **Important**
>
> 請勿在 `Description` 或 `Tags` 欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

```
$ aws kms create-key --multi-region
{
    "KeyMetadata": {
        "Origin": "AWS_KMS",
        "KeyId": "mrk-1234abcd12ab34cd56ef1234567890ab",
        "Description": "",
        "KeyManager": "CUSTOMER",
        "Enabled": true,
        "KeySpec": "SYMMETRIC_DEFAULT",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "KeyState": "Enabled",
        "CreationDate": 1606329032.475,
        "Arn": "arn:aws:kms:us-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
        "AWSAccountId": "111122223333",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ],
        "MultiRegion": true,
        "MultiRegionConfiguration": {
            "MultiRegionKeyType": "PRIMARY",
            "PrimaryKey": {
                "Arn": "arn:aws:kms:us-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
```

```
                "Region": "us-east-1"
            },
            "ReplicaKeys": [ ]
        }
    }
}
```

建立多區域複本金鑰

您可以使用ReplicateKey作業或使用AWS CloudFormation範本，在主AWS KMS控台中建立多區域複本金鑰。您無法使用此CreateKey作業來建立複本金鑰。

您可以使用這些程序複製任何多區域主要金鑰，包括對稱加密 KMS 金鑰、非對稱 KMS 金鑰或 HMAC KMS 金鑰。

當這項操作完成時，新的複本金鑰會有短暫的 `Creating` 金鑰狀態。建立新複本金鑰的程序完成後，此金鑰狀態會在幾秒鐘後變更為 `Enabled` (或 PendingImport)。雖然金鑰狀態為 `Creating`，您可以管理金鑰，但還無法在密碼編譯操作中使用金鑰。如果您以程式設計方式建立並使用複本金鑰，請在使用前重試`KMSInvalidStateException`或呼叫DescribeKey以檢查其`KeyState`值。

如果錯誤地刪除了複本金鑰，則可使用此程序重新建立。如果您在同一個區域中複寫相同的主要金鑰，則您建立的新複本金鑰將具有相同的共用屬性作為原始複本金鑰。

> **Important**
>
> 請勿在別名、說明或標籤包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

進一步了解

- 若要建立具有匯入金鑰材料的多區域複本金鑰，請參閱 建立具有匯入金鑰材料的複本金鑰。
- 若要使用AWS CloudFormation範本建立複本金鑰，請參閱《使AWS CloudFormation用指南》AWS::KMS::ReplicaKey中的〈〉。

主題

- 複本區域
- 建立複本金鑰 (主控台)
- 建立複本金鑰 (AWS KMS API)

複本區域

您通常會根據您的業務模式和法規要求選擇將多區域金鑰複寫至 AWS 區域。例如，您可能會將金鑰複寫至您保留資源的區域。或者，為了符合災難復原要求，您可以將金鑰複寫至地理位置偏遠的區域。

如下是複本區域的 AWS KMS 要求。如果您選擇的區域不符合這些要求，則嘗試複寫金鑰會失敗。

- 每個區域一個相關的多區域金鑰 – 您無法在與其主要金鑰相同的區域中建立複本金鑰，或在與主要金鑰之另一個複本相同的區域中建立複本金鑰。

  如果嘗試在已有該主要金鑰的區域中複寫主要金鑰，則嘗試會失敗。如果區域中目前的複本金鑰處於 `PendingDeletion` 金鑰狀態，您可以取消複本金鑰刪除，或等到複本金鑰刪除完成。
- 同一區域中多個不相關的多個區域金鑰 – 您可以在同一區域中擁有多個不相關的多區域金鑰。例如，您可以在 `us-east-1` 區域中有兩個多區域主要金鑰。每個主要金鑰都可以在 `us-west-2` 區域中有複本金鑰。
- 相同分割區中的區域 – 複本金鑰區域必須位於與主要金鑰區域相同的 AWS 分割區中。
- 必須啟用區域 – 如果區域預設為停用，則您無法在該區域中建立任何資源，直到為您的 AWS 帳戶啟用。

建立複本金鑰 (主控台)

在 AWS KMS 主控台中，您可以在相同操作中建立多區域主要金鑰的一或多個複本。

此程序類似於在主控台中建立標準的單一區域 KMS 金鑰。不過，因為複本金鑰是以主要金鑰為基礎，所以您不會選取共用屬性的值，例如金鑰規格 (對稱或非對稱)、金鑰使用情形或金鑰來源。

您可以指定不共用的屬性，包括別名、標籤、描述和金鑰政策。為了方便起見，主控台會顯示主要金鑰的目前屬性值，但您可以對其進行變更。即使您保留主要鍵值，AWS KMS 也不會保持這些值同步。

> **Important**
>
> 請勿在別名、說明或標籤包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。

4. 選取多區域主要金鑰的金鑰 ID 或別名。如此會開啟 KMS 金鑰的金鑰詳細資訊頁面。

   若要識別多區域主要金鑰，請使用右上角的工具圖示新增 Regionality (區域性) 資料欄至資料表。

5. 選擇 Regionality (區域性) 索引標籤。

6. 在 Related multi-Region keys (相關的多區域金鑰) 區段中，選擇 Create new replica keys (建立新的複本金鑰)。

   Related multi-Region keys (相關的多區域金鑰) 區段會顯示主要金鑰及其複本金鑰的區域。您可以使用此顯示來協助您為新複本金鑰選擇區域。

7. 選擇一或多個 AWS 區域。此操作程序會在您選取的每個區域中建立複本金鑰。

   該選單只包括與主要金鑰所在相同 AWS 分割區中的區域。已有相關多區域金鑰的區域會顯示出來，但無法選取。您可能沒有許可將金鑰複寫至選單上的所有區域。

   完成選擇「區域」後，請關閉選單。隨即會顯示您選擇的區域。若要取消複寫至區域，請選擇區域名稱旁邊的 X。

8. 輸入複本金鑰的別名。

   主控台會顯示主要金鑰目前的其中一個別名，但您可以對其進行變更。您可以為多區域主要金鑰及其複本提供相同的別名或不同的別名。別名不是多區域金鑰的共用屬性。AWS KMS 不會同步多區域金鑰的別名。

   新增、刪除或更新別名可允許或拒絕 KMS 金鑰的許可。如需詳細資訊，請參閱 AWS KMS 的 ABAC 和 使用別名來控制對 KMS 金鑰的存取。

9. (選用) 輸入複本金鑰的描述。

   主控台會顯示主要金鑰的當前描述，但您可以對其進行變更。描述不是多區域金鑰的共用屬性。您可以為多區域主要金鑰及其複本提供相同的描述或不同的描述。AWS KMS 不會同步多區域金鑰的金鑰描述。

10. (選用) 輸入標籤索引鍵和選用標籤值。若要為複本金鑰指派超過一個標籤，請選擇 Add tag (新增標籤)。

    主控台會顯示目前連接至主要金鑰的標籤，但您可以對其進行變更。標籤不是多區域金鑰的共用屬性。您可以為多區域主要金鑰及其複本提供相同的標籤或不同的標籤。AWS KMS 不會同步多區域金鑰的標籤。

    標記或取消標記 KMS 金鑰可以允許或拒絕 KMS 金鑰的許可。如需詳細資訊，請參閱 AWS KMS 的 ABAC 和 使用標籤來控制對 KMS 金鑰的存取。

11. 選取可管理複本金鑰的 IAM 使用者和角色。

> **Note**
>
> IAM 政策可授權其他 IAM 使用者和角色來管理複本金鑰。
> IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

這個步驟會開始為複本金鑰建立金鑰政策的程序。主控台會顯示主要金鑰的當前金鑰政策，但您可以對其進行變更。金鑰政策不是多區域金鑰的共用屬性。您可以為多區域主要金鑰及其複本提供相同的金鑰政策或不同的金鑰政策。AWS KMS 不會同步金鑰政策。您可以隨時變更任何 KMS 金鑰的金鑰政策。

12. 完成建立金鑰政策的步驟，包括選取金鑰使用者。檢閱金鑰政策後，選擇 Finish (完成) 以建立複本金鑰。

建立複本金鑰 (AWS KMS API)

若要建立多區域複本金鑰，請使用作ReplicateKey業。您無法使用此CreateKey作業來建立複本金鑰。此操作一次會建立一個複本金鑰。您指定的區域必須符合複本金鑰的區域要求。

當使用 `ReplicateKey` 操作時，您不需要為多區域金鑰的任何共用屬性指定值。共用屬性值會從主要金鑰複製並保持同步。然而，您可以為不共用的屬性指定值。否則，AWS KMS 會套用 KMS 金鑰的標準預設值，而不是主要金鑰的值。

> **Note**
>
> 如不為 `Description`、`KeyPolicy` 或 `Tags` 參數指定值，則 AWS KMS 會建立複本金鑰 (具空字串描述、預設金鑰政策，且無標籤)。
> 請勿在 `Description` 或 `Tags` 欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

例如，以下命令會在亞太區域 (雪梨) 區域 (ap-southeast-2) 建立多區域複本金鑰。此複本金鑰是基於美國東部 (維吉尼亞北部) 區域 (us-east-1) 的主要金鑰進行建模，由 `KeyId` 參數值識別。此範例會接受所有其他屬性 (包括金鑰政策) 的預設值。

回應會描述新的複本金鑰。它包含共用屬性的欄位，例如 KeyId、KeySpec、KeyUsage，以及金鑰材料來源 (Origin)。它也包含獨立於主要金鑰的屬性，例如 Description、金鑰政策 (ReplicaKeyPolicy) 和標籤 (ReplicaTags)。

回應還包括金鑰 ARN 和主要金鑰的區域及其所有複本金鑰，包括剛剛在 ap-southeast-2 區域中建立的金鑰。在此範例中，ReplicaKey 元素顯示此主要金鑰已在歐洲 (愛爾蘭) 區域 (eu-west-1) 中複寫。

```
$ aws kms replicate-key \
    --key-id arn:aws:kms:us-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab \
    --replica-region ap-southeast-2
{
    "ReplicaKeyMetadata": {
        "MultiRegion": true,
        "MultiRegionConfiguration": {
            "MultiRegionKeyType": "REPLICA",
            "PrimaryKey": {
                "Arn": "arn:aws:kms:us-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                "Region": "us-east-1"
            },
            "ReplicaKeys": [
                {
                    "Arn": "arn:aws:kms:ap-southeast-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                    "Region": "ap-southeast-2"
                },
                {
                    "Arn": "arn:aws:kms:eu-west-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                    "Region": "eu-west-1"
                }
            ]
        },
        "AWSAccountId": "111122223333",
        "Arn": "arn:aws:kms:ap-southeast-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
        "CreationDate": 1607472987.918,
        "Description": "",
        "Enabled": true,
        "KeyId": "mrk-1234abcd12ab34cd56ef1234567890ab",
        "KeyManager": "CUSTOMER",
        "KeySpec": "SYMMETRIC_DEFAULT",
```

```
        "KeyState": "Enabled",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "Origin": "AWS_KMS",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ]
    },
    "ReplicaKeyPolicy": "{\n  \"Version\" : \"2012-10-17\",\n  \"Id\" : \"key-
default-1\",...,
    "ReplicaTags": []
}
```

檢視多區域金鑰

您可以在 AWS KMS 主控台和透過使用 AWS KMS API 操作，檢視單一區域和多區域金鑰。

主題

- 在主控台中檢視多區域金鑰
- 在 API 中檢視多區域金鑰

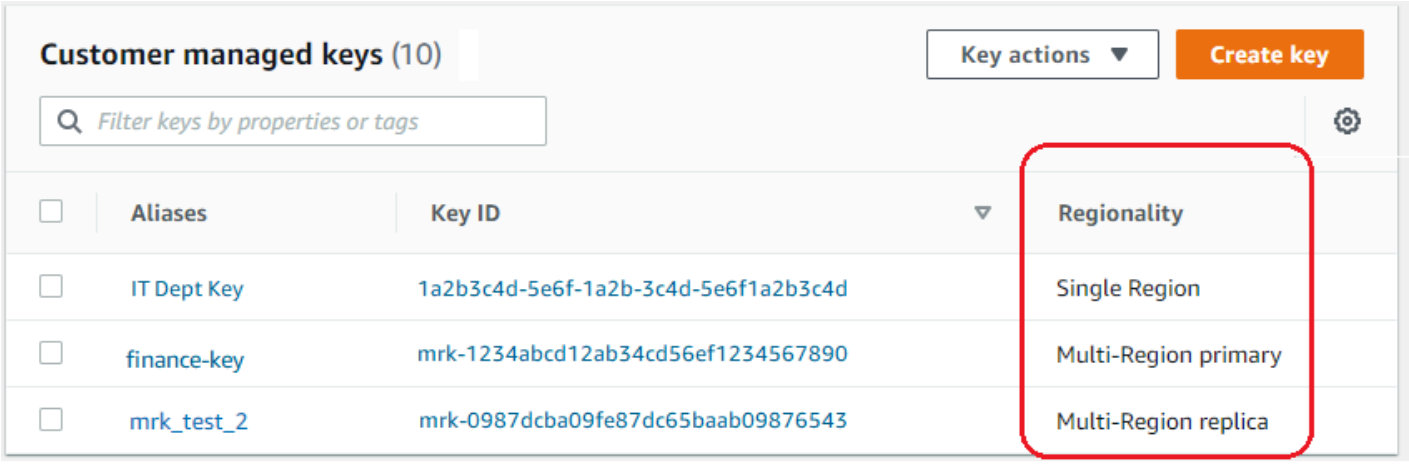
在主控台中檢視多區域金鑰

在 AWS KMS 主控台中，您可以檢視所選區域中的 KMS 金鑰。不過，如果您有多區域金鑰，您可以在其他 AWS 區域 中查看其相關的多區域金鑰。

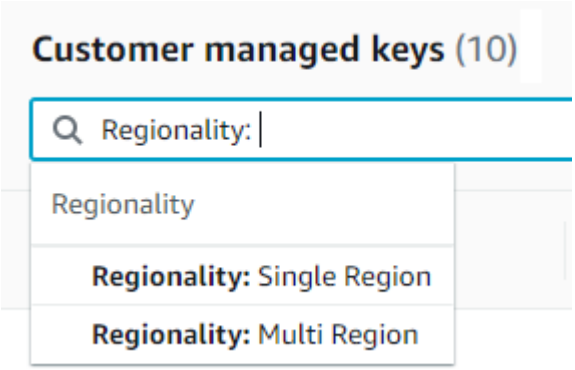
AWS KMS 主控台中的 Customer managed keys (客戶受管金鑰) 資料表僅顯示所選區域中的 KMS 金鑰。您可以檢視所選區域中的多區域主要金鑰和複本金鑰。若要變更 AWS 區域，請使用頁面右上角的區域選取器。

AWS 受管金鑰 資料表沒有區域性功能，因為 AWS 受管金鑰 一律是單一區域金鑰。

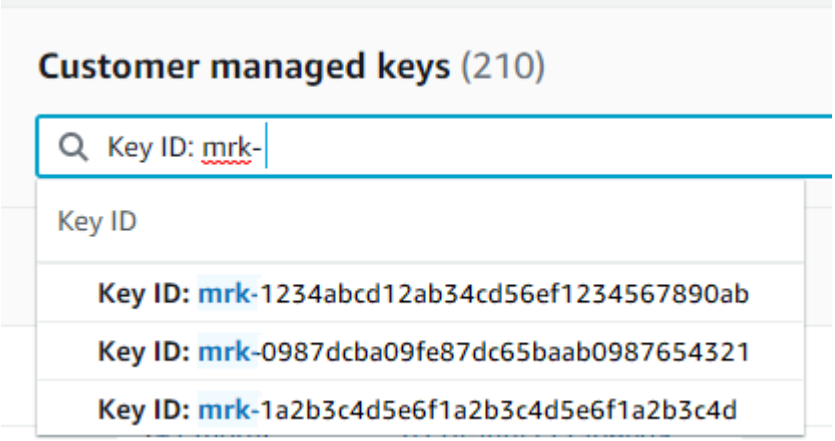
- 若要輕鬆識別您的多區域金鑰，請新增 Regionality (區域性) 欄至您的金鑰資料表。如需協助，請參閱 自訂您的 KMS 金鑰資料表。



- 若僅在金鑰資料表中顯示單一區域金鑰或僅顯示多區域金鑰，請依每個金鑰的 Regionality (區域性) 屬性篩選您的金鑰。如需協助，請參閱 排序和篩選 KMS 金鑰。



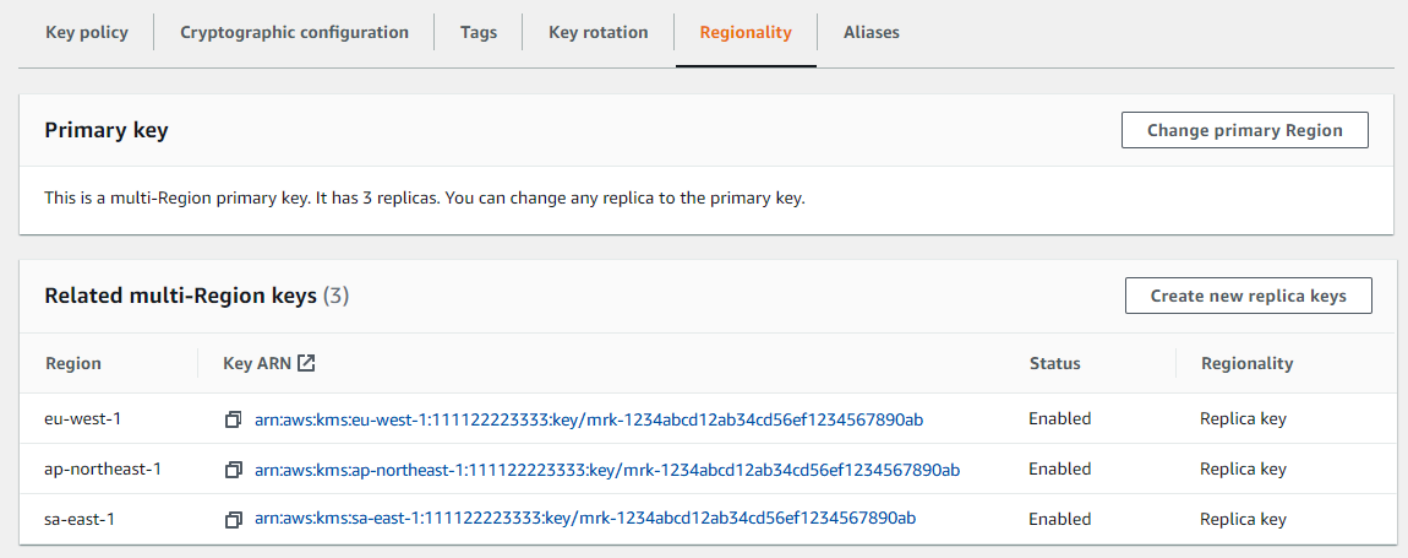
- 您也可以針對獨特 mrk- 金鑰 ID 字首，對 Customer managed keys (客戶受管金鑰) 資料表進行排序和篩選。



- 如需多區域主要金鑰或複本金鑰的詳細資訊，請前往金鑰的詳細資訊頁面，然後選擇 Regionality (區域性) 索引標籤。

Regionality (區域性) 索引標籤包括「變更主要區域」和「建立新的複本金鑰」按鈕。(複本金鑰的「區域性」索引標籤沒有按鈕)。Related multi-Region keys (相關的多區域金鑰) 區段會列出所有與目前區域相關的多區域金鑰。如果目前金鑰是複本金鑰，則此清單會包含主要金鑰。

如果您從 Related multi-Region keys (相關的多區域金鑰) 資料表選擇相關的多區域金鑰，則 AWS KMS 主控台會變更為所選取金鑰的區域，並開啟金鑰的詳細資訊頁面。例如，如果您從下列範例 Related multi-Region keys (相關多區域金鑰) 區段選擇 `sa-east-1` 區域的複本金鑰，則 AWS KMS 主控台變更為 `sa-east-1` 區域以顯示該複本金鑰的詳細資訊頁面。您可以執行這項操作來檢視複本金鑰的別名或金鑰政策。若要再次變更區域，請使用頁面右上角的區域選取器。



在 API 中檢視多區域金鑰

若要檢視 AWS KMS API 中的多區域金鑰，請使用此DescribeKey作業。它顯示指定金鑰及其所有相關的多區域金鑰。

就像 AWS KMS 主控台、AWS KMS API 操作是區域性的。例如，當您呼叫ListKeys或ListAliases作業時，它們只會傳回目前或指定區域中的資源。但是，當您針對多區域金鑰呼叫 `DescribeKey` 操作時，回應會包含其他 AWS 區域 中所有相關的多區域金鑰。

例如，下列 `DescribeKey` 請求會取得亞太區域 (東京) (`ap-northeast-1`) 區域中範例多區域複本金鑰的詳細資訊。

```
$ aws kms describe-key \
        --key-id arn:aws:kms:ap-northeast-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab \
```

```
        --region ap-northeast-1
```

回應中的大部分 KeyMetadata 描述了亞太區域 (東京) 區域中的複本金鑰，這是請求的主旨。不過，MultiRegionConfiguration 元素會描述美國西部 (奧勒岡) (us-west-2) 區域的主要金鑰及其其他 AWS 區域 的複本金鑰，包括亞太區域 (東京) 區域的複本。DescribeKey 會傳回相同的 MultiRegionConfiguration 值，用於所有相關的多區域金鑰。

```
{
    "KeyMetadata": {
        "MultiRegion": true,
        "AWSAccountId": "111122223333",
        "Arn": "arn:aws:kms:ap-northeast-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
        "CreationDate": 1586329200.918,
        "Description": "",
        "Enabled": true,
        "KeyId": "mrk-1234abcd12ab34cd56ef1234567890ab",
        "KeyManager": "CUSTOMER",
        "KeyState": "Enabled",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "Origin": "AWS_KMS",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ],
        "MultiRegionConfiguration": {
            "MultiRegionKeyType": "PRIMARY",
            "PrimaryKey": {
                "Arn": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                "Region": "us-west-2"
            },
            "ReplicaKeys": [
                {
                    "Arn": "arn:aws:kms:eu-west-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                    "Region": "eu-west-1"
                },
                {
                    "Arn": "arn:aws:kms:ap-northeast-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                    "Region": "ap-northeast-1"
                },
```

```
                {
                    "Arn": "arn:aws:kms:sa-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                    "Region": "sa-east-1"
                }
            ]
        }
    }
}
```

管理多區域金鑰

對於大多數動作，您可以採用與使用和管理單一區域金鑰相同的方式來管理多區域金鑰。您可以啟用和停用金鑰、設定和更新別名、金鑰政策、授予和標籤。不過，多區域金鑰的管理方式有下列不同。

- 您可以更新主要區域。這會將其中一個複本金鑰變更為主要金鑰，並將目前的主要金鑰變更為複本。
- 您僅在主要金鑰上管理自動金鑰輪換。
- 您可以從任何相關主要或複本金鑰取得非對稱多區域金鑰的公有金鑰。

您在建立 KMS 金鑰時所設定的多區域屬性是不可變的。您無法將單一區域金鑰轉換為多區域金鑰，或將多區域金鑰轉換為單一區域金鑰。

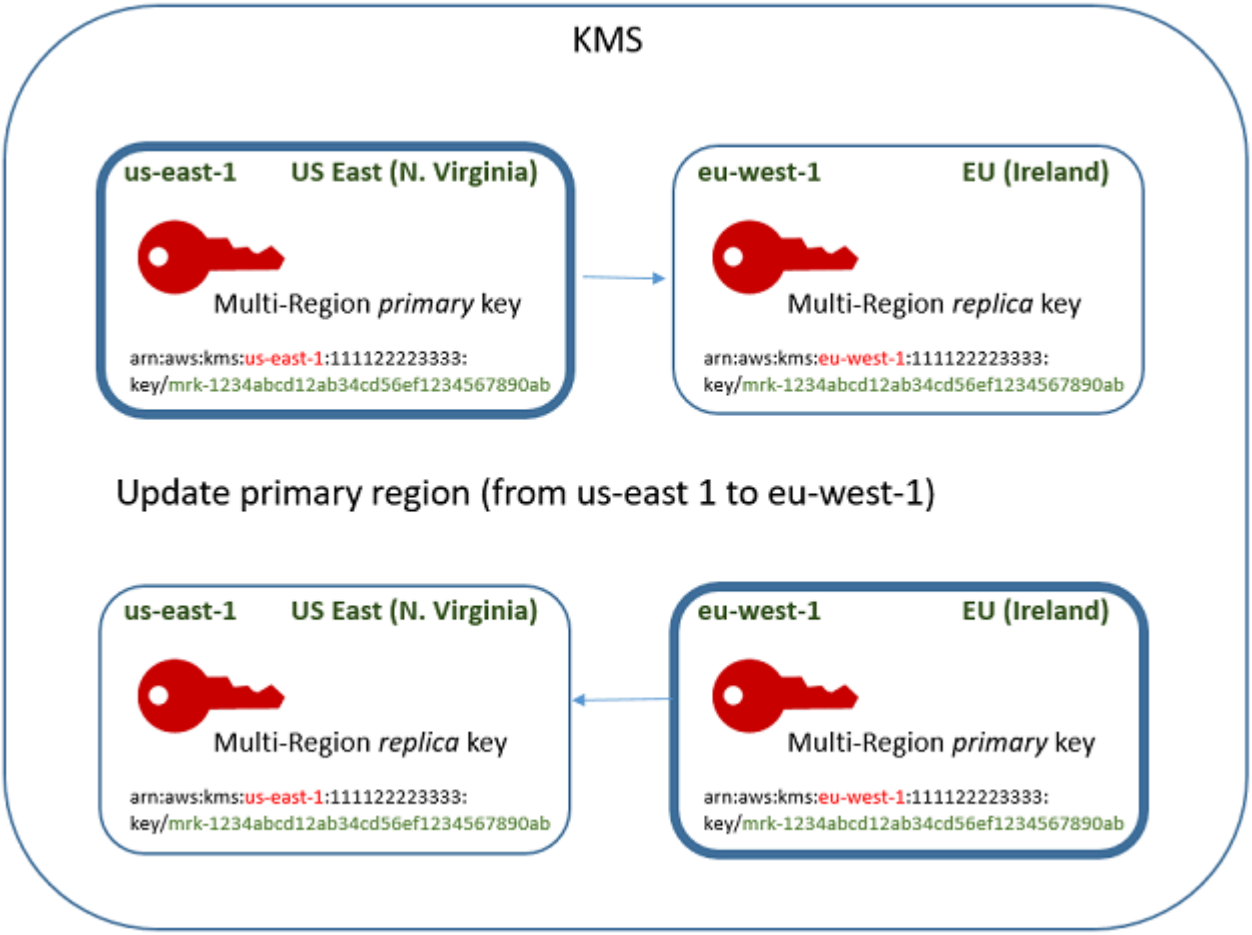
更新主要區域

每組相關的多區域金鑰都必須有主要金鑰。但您可以變更主要金鑰。這個動作，稱為更新主要區域，會將目前的主要金鑰轉換為複本金鑰，並將其中一個相關的複本金鑰轉換為主要金鑰。如果您需要在維護複本金鑰的同時刪除目前的主要金鑰，或是要在與金鑰管理員相同的區域中尋找主要金鑰，則可以執行此動作。

您可以選取任何相關的複本金鑰做為新的主要金鑰。主要金鑰和複本金鑰都必須在操作開始時處於 `Enabled` 金鑰狀態。

即使在此操作完成之後，主要區域的更新程序仍可能會繼續進行幾秒鐘。在此期間，舊的和新的主要金鑰具有更新的暫時性金鑰狀態。雖然金鑰狀態為 `Updating`，您可以在密碼編譯操作中使用金鑰，但無法複寫新的主要金鑰或執行特定的管理操作，例如啟用或停用這些金鑰。諸如此類的作業DescribeKey可能會將舊主索引鍵和新主索引鍵都顯示為複本。更新完成時，`Enabled` 金鑰狀態會還原。

假設您在美國東部 (維吉尼亞北部) (us-east-1) 中有主要金鑰和在歐洲 (愛爾蘭) (eu-west-1) 中有複本金鑰。您可以使用更新功能將美國東部 (維吉尼亞北部) (us-east-1) 的主要金鑰變更為複本金鑰，並將歐洲 (愛爾蘭) (eu-west-1) 的複本金鑰變更為主要金鑰。



當更新程序完成時，歐洲 (愛爾蘭) (eu-west-1) 區域的多區域金鑰是多區域主要金鑰，而美國東部 (維吉尼亞北部) (us-east-1) 區域的多區域主要金鑰是其複本金鑰。如果有其他相關的複本金鑰，它們會成為新主要金鑰的複本。AWS KMS 下一次會同步多區域金鑰的共用屬性，它將從新主要金鑰取得共用屬性，並將其複製到其複本金鑰，包括前主要金鑰。

更新操作並不會影響任何多區域金鑰的金鑰 ARN。它也不會影響共用屬性 (例如金鑰材料) 或獨立屬性 (例如金鑰政策)。不過，您可能想要更新新主要金鑰的金鑰政策。例如，您可能想要將 kms: 受信任主體的ReplicateKey權限新增至新主索引鍵，並將其從新的複本金鑰中移除。

`Updating` 金鑰狀態

更新主要區域的程序比影響大多數 AWS KMS 操作的短暫最終一致性延遲長。`UpdatePrimaryRegion` 操作傳回或您已在主控台完成更新處理程序後，程序可能仍在進行中。作業 (例如) DescribeKey可能會將舊的和新的主索引鍵都顯示為複本，直到處理程序完成為止。

在更新主要區域的過程中，舊主要金鑰和新主要金鑰處於 Updating 金鑰狀態。當更新程序順利完成時，兩個金鑰都會返回 Enabled 金鑰狀態。雖然處於 Updating 狀態時，某些管理操作 (例如啟用和停用金鑰) 無法使用。不過，您可以在不中斷的情況下在密碼編譯操作中繼續使用這兩個金鑰。如需有關 Updating 金鑰狀態之影響的資訊，請參閱 AWS KMS 金鑰的金鑰狀態。

更新主要區域 (主控台)

您可以在 AWS KMS 主控台更新主要金鑰。從目前主要金鑰的金鑰詳細資訊頁面開始。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 選取多區域主要金鑰的金鑰 ID 或別名。如此會開啟主要金鑰的金鑰詳細資訊頁面。

   若要識別多區域主要金鑰，請使用右上角的工具圖示新增 Regionality (區域性) 資料欄至資料表。
5. 選擇 Regionality (區域性) 索引標籤。
6. 在 Primary key (主要金鑰) 區段中，選擇 Change primary Region (變更主要區域)。
7. 選擇新主要金鑰的區域。您只可以從選單中選擇一個區域。

   Change primary Regions (變更主要區域) 選單只包含具有相關多區域金鑰的區域。您可能沒有更新選單上所有區域中主要區域的許可。
8. 選擇 Change primary Region (變更主要區域)。

更新主要區域 (AWS KMS API)

若要變更一組相關的多區域金鑰中的主索引鍵，請使用此UpdatePrimaryRegion作業。

使用 KeyId 參數來識別目前的主要金鑰。使用 PrimaryRegion 參數來指示新主要金鑰的 AWS 區域。如果主要金鑰尚未在新的主要區域中有複本，則操作會失敗。

下列範例會將 us-west-2 區域中多區域金鑰的主要金鑰變更為其在 eu-west-1 區域的複本。KeyId 參數會識別 us-west-2 區域中的目前主要金鑰。PrimaryRegion 參數會指定新主要金鑰的 AWS 區域，eu-west-1。

```
$ aws kms update-primary-region \
      --key-id arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab \
```

```
      --primary-region eu-west-1
```

成功時，此操作不會返回任何輸出；只是 HTTP 狀態碼。若要查看效果，請呼叫其中一個多區域鍵的DescribeKey作業。您可能需要等到金鑰狀態返回到 Enabled。金鑰狀態為正在更新時，金鑰的值可能仍在變化中。

例如，下列 DescribeKey 呼叫會取得有關 eu-west-1 區域中多區域金鑰的詳細資訊。輸出顯示 eu-west-1 區域中的多區域金鑰現在已成為主要金鑰。us-west-2 區域中相關的多區域金鑰 (相同金鑰 ID) 現在是複本金鑰。

```
$ aws kms describe-key \
      --key-id arn:aws:kms:eu-west-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab \

{
    "KeyMetadata": {
        "AWSAccountId": "111122223333",
        "KeyId": "mrk-1234abcd12ab34cd56ef1234567890ab",
        "Arn": "arn:aws:kms:eu-west-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
        "CreationDate": 1609193147.831,
        "Enabled": true,
        "Description": "multi-region-key",
        "KeySpec": "SYMMETRIC_DEFAULT",
        "KeyState": "Enabled",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "Origin": "AWS_KMS",
        "KeyManager": "CUSTOMER",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ],
        "MultiRegion": true,
        "MultiRegionConfiguration": {
           "MultiRegionKeyType": "PRIMARY",
           "PrimaryKey": {
              "Arn": "arn:aws:kms:eu-west-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
              "Region": "eu-west-1"
           },
           "ReplicaKeys": [
              {
```

```
                "Arn": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                "Region": "us-west-2"
            }
        ]
    }
}
```

輪換多區域金鑰

您可以啟用和停用多區域金鑰的金鑰材料自動輪換。自動金鑰輪換是多區域金鑰的共用屬性。

您只能在主要金鑰上啟用和停用自動金鑰輪換。

- AWS KMS 同步多區域金鑰時，它會將主要金鑰的金鑰輪換屬性設定複製到其所有相關複本金鑰。
- AWS KMS 輪換金鑰材料時，它會為主要金鑰建立新金鑰材料，然後將新金鑰材料跨區域邊界複製到所有相關的複本金鑰。金鑰材料絕不會讓 AWS KMS 出現未加密的狀況。這個步驟經過精心控制，以確保在密碼編譯操作中使用任何金鑰之前，金鑰材料已完全同步。
- 在金鑰材料可用於主要金鑰和每個其複本金鑰之前，AWS KMS 不會使用新金鑰材料加密任何資料。
- 當您複寫已輪換的主要金鑰時，新的複本金鑰會具有目前的金鑰材料，以及其相關多區域金鑰的所有舊版金鑰材料。

這種模式可確保相關的多區域金鑰完全可互通操作。任何多區域金鑰都可以解密由相關多區域金鑰加密的任何加密文字，即使加密文字在建立金鑰之前已加密。

非對稱 KMS 金鑰或具有匯入金鑰材料的 KMS 金鑰不支援自動金鑰輪換。如需自動金鑰輪換的相關資訊，以及啟用和停用此功能的指示，請參閱 輪換 AWS KMS keys。

下載公開金鑰

當您建立多區域非對稱 KMS 金鑰時，AWS KMS 會針對主要金鑰建立 RSA 或橢圓曲線 (ECC) 金鑰對。然後，其會將該金鑰對複製到主要金鑰的每個複本。因此，您可以從主鑰金鑰或其任何複本金鑰下載公有金鑰。一律會得到相同的金鑰材料。

如需下載和使用 AWS KMS 外部公有金鑰的詳細資訊，請參閱 下載公開金鑰的特殊考量。如需說明，請參閱 下載公開金鑰。

將金鑰材料匯入多區域金鑰

您可匯入自己的金鑰資料至多區域 KMS 金鑰。使用您自己的金鑰材料建立的多區域金鑰可互通操作。您可以在一個區域中加密資料，並使用相關的多區域金鑰在任何其他區域中解密資料。

但是，您必須管理金鑰材料。

- AWS KMS 不會將金鑰材料從具有匯入金鑰材料的主要金鑰複製或同步至其複本金鑰。您必須將相同的金鑰材料匯入相關的主要金鑰和複本金鑰。
- 當您匯入金鑰材料時，您可以獨立設定過期模型和每個金鑰的過期日期。您可以為相關的多區域金鑰設定相同或不同的過期模型和過期日期。如果金鑰材料即將過期，則您必須將金鑰材料重新匯入到受影響的多區域金鑰中。

  相關的多區域金鑰的金鑰狀態是相互獨立的。例如，如果主要金鑰中的金鑰材料過期，則其複本金鑰不會受到影響。

相同複本金鑰的區域需求適用於具有匯入金鑰材料的多區域金鑰。如果您將相同的金鑰材料匯入單一區域金鑰或不相關的多區域金鑰，則這些 KMS 金鑰不可互通操作。

您可利用匯入的對稱、非對稱或 HMAC 金鑰資料來建立多區域金鑰。AWS KMS 不支援自訂金鑰存放區中的匯入金鑰資料。此外，您無法為具有匯入金鑰材料的任何 KMS 金鑰啟用自動金鑰輪換。

除了多區域功能之外，具有匯入金鑰材料的多區域金鑰與具有匯入金鑰材料的其他 KMS 金鑰相同。如需建立和設定具有匯入金鑰資料之單一區域金鑰的詳細資訊，請參閱 關於匯入的金鑰材料。

主題

為什麼所有具有匯入金鑰材料的 KMS 金鑰都不可互通？

具有匯入金鑰材料的單一區域 KMS 金鑰無法互通，即使它們具有相同的金鑰材料。AWS KMS 使用 KMS 金鑰來加密資料時，它會以密碼編譯方式將某些金鑰中繼資料繫結至加密文字。這會保護加密文字，以便只有加密資料的 KMS 金鑰可以解密該資料。

多區域金鑰設計為可互通操作。除了具有相同的金鑰材料之外，它們還具有相同的金鑰 ID 和其他中繼資料。因此，其生成的加密文字可以通過任何相關的多區域金鑰進行解密。因此，多區域金鑰的信任屬

性與單一區域金鑰的信任屬性不同。但對於某些客戶來說，在多個區域中解密的好處超過了依賴於單一 AWS 區域 中單一 KMS 金鑰的加密文字安全值。

建立具有匯入金鑰材料的主要金鑰

若要建立包含匯入金鑰資料的主要金鑰，請先建立不含金鑰資料的 KMS 金鑰。當您建立不含金鑰資料的主要金鑰時，您必須根據您計畫匯入的金鑰資料來指定可反映其類型的金鑰規格。然後，您可匯入金鑰資料至主要金鑰。

建立不含金鑰材料之多區域主要金鑰的操作程序幾乎與建立不含金鑰材料的單一區域金鑰相同。唯一的差別是，您指定金鑰為多區域金鑰。

使用匯入的金鑰材料建立多區域主索引鍵的許可，與使用金鑰材料建立多區域主索引鍵 (包括 IAM 政策中的 kms: CreateKey 和 iam: CreateServiceLinkedRole 許可) 所需的權限相同。AWS KMS您可以使用 kms: MultiRegionKeyType 和 kms: KeyOrigin 條件金鑰來允許或拒絕使用匯入金鑰材料建立多區域主索引鍵的權限。

在 AWS KMS 主控台建立具有匯入金鑰材料的主索引鍵時，請使用 Advanced options (進階選項) 區段中的設定。建立 KMS 金鑰之後即無法變更這些屬性。

- 將 Key material origin (金鑰材料來源) 設定為 External (Import key material) (外部 (匯入金鑰材料))。
- 將 Multi-Region replication (多區域複寫) 設定為 Allow this key to be replicated into other Regions (允許此金鑰複寫至其他區域)。

使用CreateKey作業建立具有匯入關鍵字材料的主關鍵字時，請使用`Origin`和`MultiRegion`參數`KeySpec`並指定和`KeyUsage`。下列範例會建立可匯入 `ECC_NIST_P384` 金鑰資料的 `EXTERNAL` KMS 金鑰。

```
$ aws kms create-key --origin EXTERNAL --key-spec ECC_NIST_P384 --key-usage SIGN_VERIFY
 --multi-region
```

結果是不含金鑰材料的多區域金鑰，且金鑰狀態為 `PendingImport`。

若要啟用此 KMS 金鑰，您必須下載公有金鑰和匯入字符、使用公有金鑰來加密您的金鑰材料，然後匯入您的金鑰材料。如需相關指示，請參閱匯入 AWS KMS 金鑰的金鑰材料。

建立具有匯入金鑰材料的複本金鑰

您可以在 AWS KMS 主控台或使用 AWS KMS API 操作建立多區域複本金鑰。若要複寫具有匯入金鑰材料的多區域主要金鑰，請使用與建立具有 AWS KMS 金鑰材料的複本金鑰相同的操作程序。但是，結果是不同的。複寫程序會傳回不含金鑰材料且金鑰狀態為 `PendingImport` 的複本金鑰，並非傳回具有與主要金鑰相同之金鑰材料的複本金鑰。若要啟用複本金鑰，您必須將相同的金鑰材料匯入您已匯入到其主要金鑰的複本金鑰。

雖然不會複寫金鑰材料，但 AWS KMS 會建立與主要金鑰具有相同金鑰 ID、金鑰規格、金鑰使用情形和金鑰材料來源的複本金鑰。它也可確保您匯入至複本金鑰的金鑰材料與您已匯入至主要金鑰的金鑰材料相同。

建立具有匯入金鑰材料的複本金鑰：

1. 建立具有匯入金鑰材料的多區域主要金鑰。
2. 執行下列其中一項操作。

   在 AWS KMS 主控台中，選擇具有匯入金鑰材料的多區域主要金鑰。然後，在其 Regionality (區域性) 索引標籤上選擇 Create new replica keys (建立新的複本金鑰)。如需說明，請參閱建立複本金鑰 (主控台)。

   或者使用ReplicateKey操作。對於 `KeyId` 參數，輸入具有匯入金鑰材料之多區域主要金鑰的金鑰 ID 或金鑰 ARN。如需說明，請參閱建立複本金鑰 (AWS KMS API)。
3. 對於每個新的複本金鑰，請依照步驟下載公有金鑰和匯入字符。使用公有金鑰來加密主要金鑰的金鑰資料，然後在複本金鑰中匯入主要金鑰的金鑰材料。您需要每個複本金鑰的不同公有金鑰和匯入字符。

   如果您嘗試匯入複本金鑰的金鑰材料與其主要金鑰不相同，則操作會失敗。AWS KMS 不需要協調過期模型和過期日期，但您可以為多區域金鑰建立業務規則。如需說明，請參閱匯入 AWS KMS 金鑰的金鑰材料。

複寫具有匯入金鑰材料之金鑰的許可

若要建立具有匯入金鑰材料的複本金鑰，您必須具有下列許可。

在主要金鑰區域中：

- 公里：ReplicateKey在主鍵（在主鍵的區域）。在主要金鑰的金鑰政策或 IAM 政策中包含此許可。

在複本金鑰區域中：

- 公里：CreateKey在 IAM 政策中。
- 公里：GetParametersForImport。您可以在複本金鑰的金鑰政策或 IAM 政策中包含此許可。
- 公里：ImportKeyMaterial。您可以在複本金鑰的金鑰政策或 IAM 政策中包含此許可。
- kms：需TagResource要在複製時指派標籤。將此許可包含在複本區域的 IAM 政策中。
- 公里：需要CreateAlias在AWS KMS控制台中複製一個密鑰。如需詳細資訊，請參閱 控制對別名的存取。

刪除多區域金鑰

當不再使用多區域主要金鑰或複本金鑰時，您可以排程其刪除。

雖然刪除 KMS 金鑰應該一律謹慎執行，但刪除多區域金鑰的複本風險較小，前提是主要金鑰仍存在於 AWS KMS。如果您從其區域刪除複本金鑰，但在已刪除金鑰下發現已加密的加密文字，則可以使用任何相關的多區域金鑰來解密該加密文字。您也可以透過將主要金鑰再次複寫至複本金鑰區域來重新建立複本金鑰。

不過，刪除主要金鑰及其所有複本金鑰是非常危險的操作，相當於刪除單一區域金鑰。

> **Warning**
>
> 刪除 KMS 金鑰很具有破壞性，可能有潛在危險。只有當您確定不再需要使用 KMS 金鑰，將來也不需要使用 KMS 金鑰時，才應繼續進行。如果您不確定，您應停用 KMS 金鑰，而不是刪除。

若要刪除主要金鑰，您必須先刪除其所有複本金鑰。如果您必須從特定區域刪除主要金鑰而不刪除其複本金鑰，請透過更新主要區域將主要金鑰變更為複本金鑰。

在排程刪除任何 KMS 金鑰之前，請檢閱主刪除 AWS KMS keys題中的注意事項，以及說明如何判斷過去使用 KMS 金鑰以及如何設定 CloudWatch 警示的主題，以警示您在等待期間使用 KMS 金鑰。刪除非對稱多區域金鑰的主要金鑰之前，請先檢閱刪除非對稱金鑰主題。

主題

- 刪除多區域金鑰的許可
- 如何刪除複本金鑰

- 如何刪除主要金鑰

刪除多區域金鑰的許可

若要排程刪除多區域金鑰，您只需要下列許可。

- kms：ScheduleKeyDeletion— 安排刪除多區域密鑰並設置其等待時間。

我們也強烈建議您擁有下列相關許可。

- kms：CancelKeyDeletion— 取消多區域金鑰的排程刪除。
- kms: DescribeKey — 檢視多區域金鑰的金鑰狀態以及相關的多區域金鑰清單。
- kms：DisableKey— 為您提供停用多區域金鑰的選項，而不是刪除它。
- kms：EnableKey— 在取消刪除多區域金鑰後還原其功能。

您也可以包含複寫主要金鑰和變更主要金鑰的許可。

- 公里：ReplicateKey
- 公里：UpdateReplicaRegion

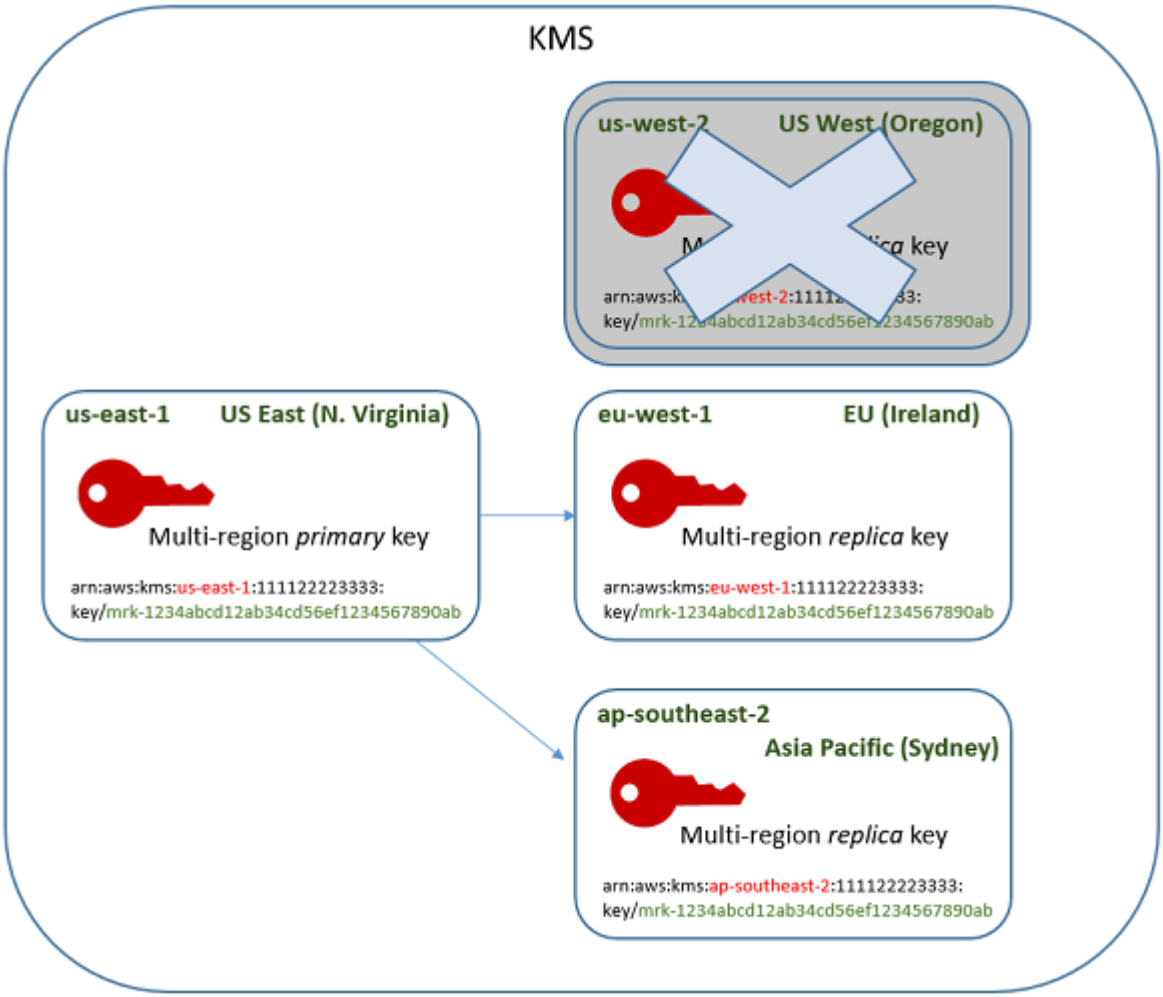
您可以將這些許可包含在 IAM 政策中，但最佳實務是將其放置在僅套用至您需要管理之 KMS 金鑰的金鑰政策中。

如何刪除複本金鑰

您可以使用 AWS KMS 主控台或 AWS KMS API 來刪除複本金鑰。您可以隨時刪除複本金鑰。它不取決於任何其他 KMS 金鑰的金鑰狀態。

如果錯誤地刪除了複本密鑰，則可透過在同一區域中複寫相同的主要金鑰來重新建立。您建立的新複本金鑰將具有相同的共用屬性作為原始複本金鑰。

刪除多區域複本金鑰的操作程序與刪除單一區域金鑰相同。



1. 排程刪除複本金鑰。選取 7-30 天的等候期。預設等待期間為 30 天。
2. 在等待期間，複本金鑰的金鑰狀態會變更為 `Pending deletion` (`PendingDeletion`)，並且您無法在密碼編譯操作中使用該金鑰。
3. 您可以在等待期間的任何時候取消排定的複本金鑰刪除。金鑰狀態會變更為 `Disabled`，但您可以重新啟用 KMS 金鑰。
4. 等待期間過期時，AWS KMS 會刪除複本金鑰。

您可以在 AWS CloudTrail 日誌中檢視動作記錄。AWS KMS 會記錄排程刪除 KMS 金鑰的操作和刪除 KMS 金鑰的動作。

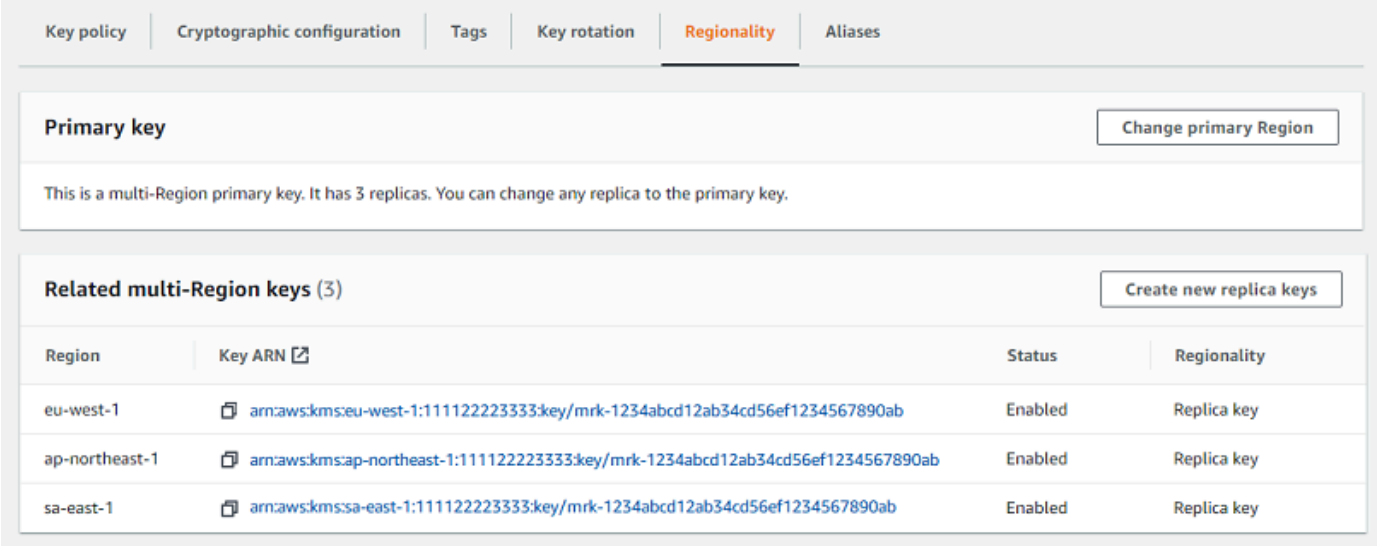
刪除複本金鑰 (主控台)

若要排程刪除多區域複本金鑰，請使用與您用於排程單一區域金鑰刪除相同的操作程序。

因為相關的複本金鑰位於不同 AWS 區域，您一次無法排程刪除一個以上的複本金鑰。若要刪除所有相關的複本金鑰，請使用以下模式。

排程刪除所有相關複本金鑰

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
3. 使用右上角的區域選取器選擇多區域主要金鑰的區域。
4. 選擇主要金演的別名或金鑰 ID。
5. 選擇 Regionality (區域性) 索引標籤。



6. 在 Related multi-Region keys (相關的多區域金鑰) 區段中，選擇複本金鑰的金鑰 ARN。

   此動作會在新的瀏覽器索引標籤中開啟複本金鑰的金鑰詳細資訊頁面。主控台設定為複本金鑰區域。

7. 從 Key actions (金鑰動作) 選單，選擇 Schedule key deletion (排程金鑰刪除)。

   此動作會啟動排程刪除金鑰的程序。完成排程金鑰刪除程序。如需詳細資訊，請參閱 排程和取消金鑰刪除 (主控台)。

8. 傳回至瀏覽器索引標籤，顯示主要金鑰的 Regionality (區域性) 索引標籤。(您可能需要重新整理頁面，才能看到複本金鑰的更新狀態。) 選擇另一個複本金鑰的金鑰 ARN，然後重複排程刪除複本金鑰的程序。

刪除複本金鑰 (AWS KMS API)

若要排程刪除多區域複本金鑰，請使用此ScheduleKeyDeletion作業。若要指定 KMS 金鑰，請使用其金鑰 ID 或金鑰 ARN。當使用多區域金鑰時，您可以透過將金鑰 ARN 與其明確的區域值搭配使用來減少錯誤的發生率。

例如，此命令會從 us-west-2 (美國西部 (奧勒岡) 區域刪除複本金鑰。由於命令未指定等待期間，因此等待期間會設定為預設值 30 天。

```
$ aws kms schedule-key-deletion \
    --region us-west-2 \
    --key-id arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab
```

當命令成功時，會傳回金鑰 ARN (KeyId)、等待期間 (PendingWindowInDays) 、刪除日期 (DeletionDate) 和當前金鑰狀態 (KeyState)，這預計是 PendingDeletion。

刪除多區域複本金鑰時，請務必確認金鑰 ARN 中的金鑰 ID 和區域值是您所預期的。

```
{
    "KeyId": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
    "DeletionDate": 1599523200.0,
    "KeyState": "PendingDeletion",
    "PendingWindowInDays": 30
}
```

若要以程式設計方式刪除多區域主要金鑰的所有複本，請建立包含複本金鑰的區域清單。然後，對於清單中的每個區域，呼叫 ScheduleKeyDeletion 操作，如上所示。

與永久刪除的單一區域金鑰不同，您可以透過將主要金鑰複寫至已刪除複本金鑰所在的區域還原複本金鑰。

若要檢查複本金鑰的狀態，並檢視多區域金鑰的主索引鍵和複本金鑰，請使用DescribeKey作業。

如何刪除主要金鑰

您可以隨時排程刪除多區域主要金鑰。然而，AWS KMS 不會刪除具有複本金鑰的多區域主要金鑰，即使這些主要金鑰已排定刪除。

若要刪除主要金鑰，您必須排程刪除其所有複本金鑰，然後等待刪除複本金鑰。刪除主要金鑰所需的等待期間在刪除其最後一個複本金鑰時開始。如果您必須從特定區域刪除主要金鑰而不刪除其複本金鑰，請透過更新主要區域將主要金鑰變更為複本金鑰。

如果主要金鑰沒有複本金鑰，則程序與刪除複本金鑰或刪除任何區域 KMS 金鑰相同。

排定刪除主要金鑰期間，您無法將其用於密碼編譯操作中，也無法對其進行複製。不過，除非它們也已排定刪除，否則其複本金鑰不會受到影響。

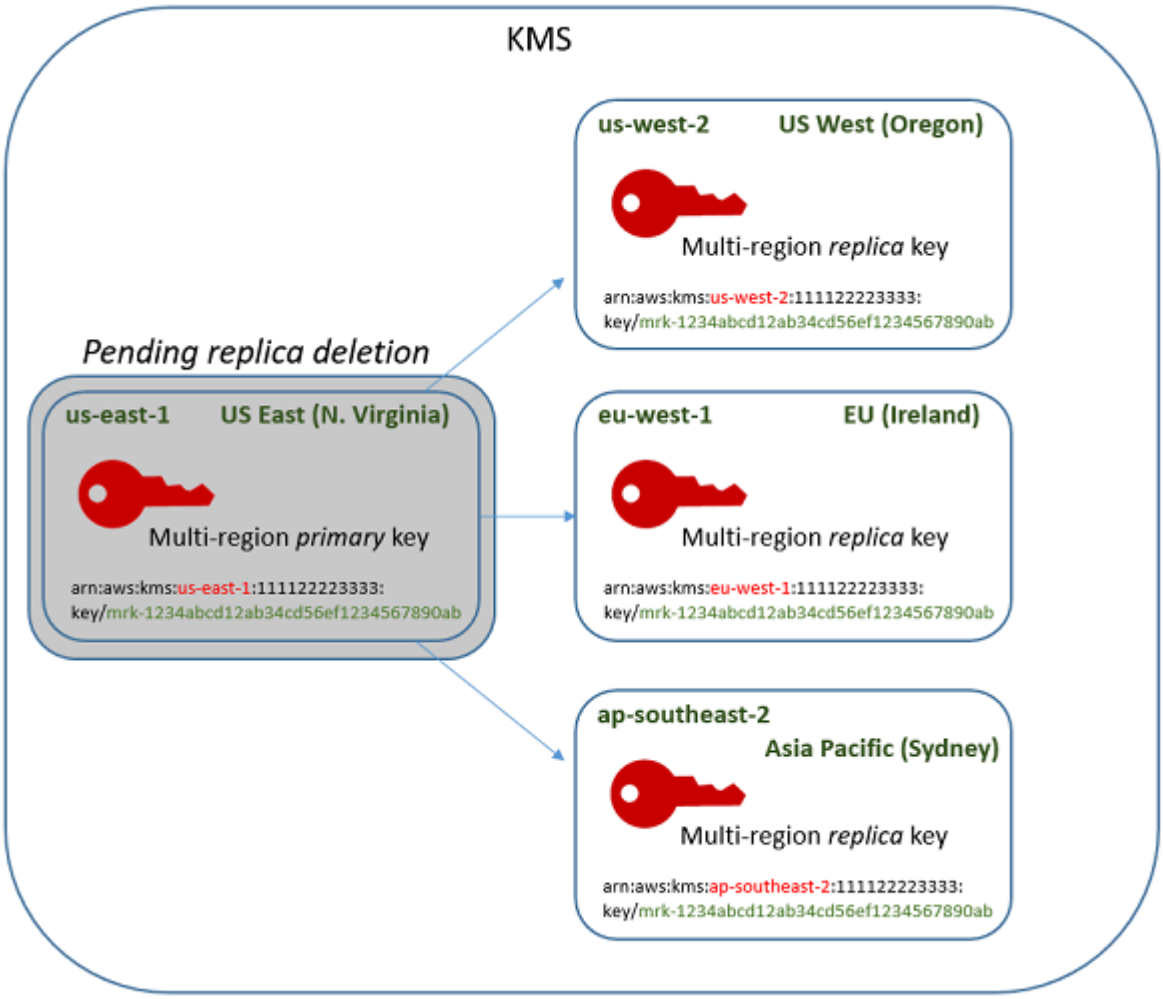
您可以使用 AWS KMS 主控台或 AWS KMS API 來排程刪除主要金鑰和複本金鑰。您可以在排程刪除主要金鑰之前、之後或同時排程刪除複本金鑰。該程序可能看起來類似如下內容。

1. 排程刪除主要金鑰。選取 7-30 天的等候期。預設等待期間為 30 天。不過，在刪除所有複本金鑰前，主要金鑰的等待期間不會開始。

   如果有任何複本金鑰仍然存在，則主要金鑰的金鑰狀態會變更為 `Pending replica deletion` (`PendingReplicaDeletion`)。否則，會變更為 `Pending deletion` (`PendingDeletion`)。在任何一種情況下，您都無法在密碼編譯操作中使用主要索引，而且無法進行複寫。

   排程主要金鑰的刪除不會影響複本金鑰。其金鑰狀態會保持啟用狀態，您可以在密碼編譯操作中使用。如果未刪除複本金鑰，主要金鑰的 `Pending replica deletion` 狀態可以無限期地持續存在。

```
KMS key:                    Key state:
Primary (us-east-1)           Pending replica deletion (waiting period 30 days -- not
 started)
Replica (us-west-2)           Enabled
Replica (eu-west-1)           Enabled
Replica (ap-southeast-2)      Enabled
```



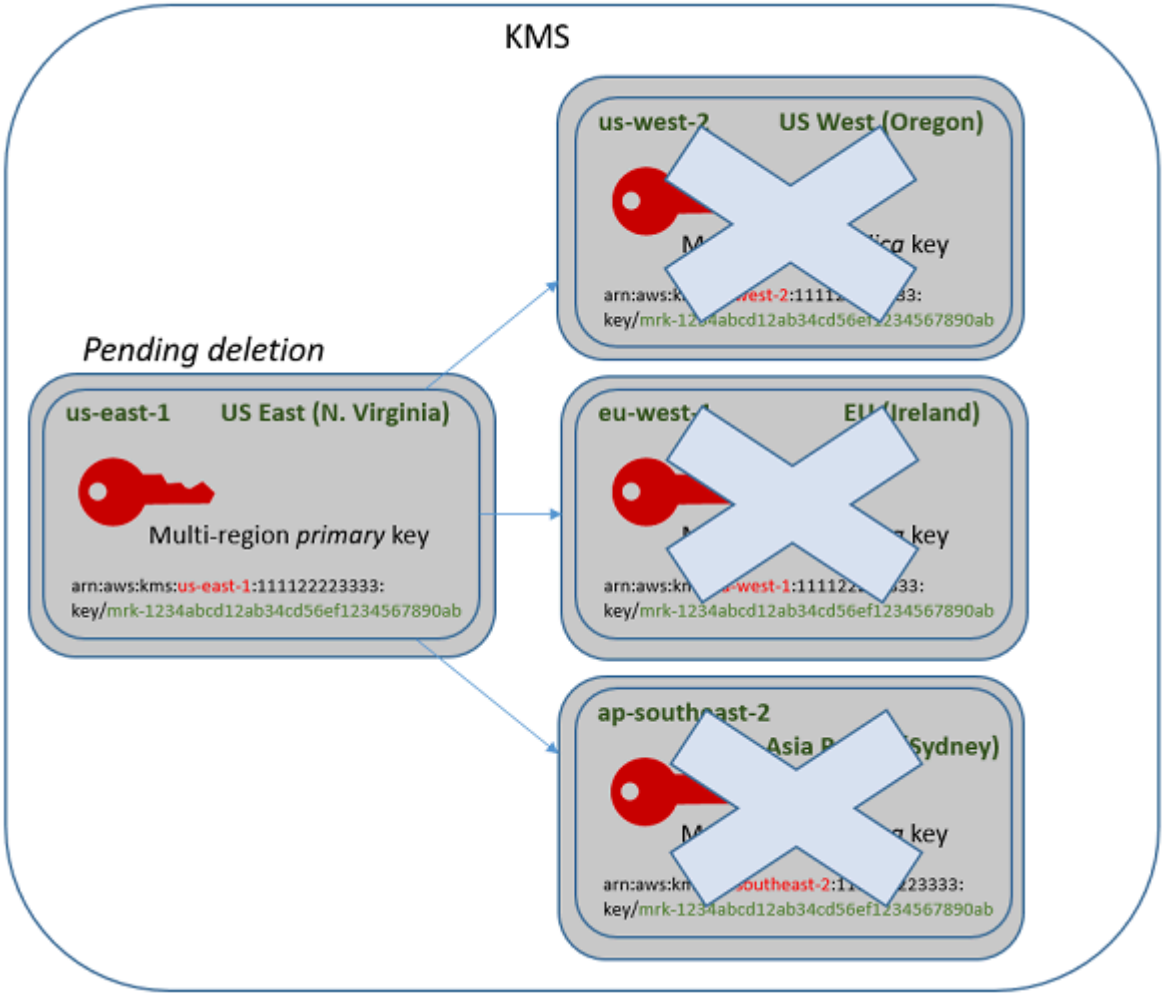
2. 排程刪除每個複本金鑰。選取 7-30 天的等候期。預設等待期間為 30 天。您可以同時刪除多個複本金鑰。其等待期會同時執行。在等待期間，複本金鑰的金鑰狀態會變更為 Pending deletion (PendingDeletion)，並且您無法在密碼編譯操作中使用這些 KMS 金鑰。

   例如，如果有三個複本金鑰，則您可以同時排程刪除所有三個金鑰。它們可以有相同或不同的等待期。請注意，主要金鑰的等待期尚未開始。其金鑰狀態為 PendingReplicaDeletion，因為它擁有現有的複本金鑰。

```
KMS key:                    Key state:
Primary key (us-east-1)       Pending replica deletion (waiting period 30 days -- not
 started)
Replica (us-west-2)           Pending deletion (7 days)
Replica (eu-west-1)           Pending deletion (7 days)
Replica (ap-southeast-2)      Pending deletion (30 days)
```

3. 您可以取消排程刪除主要金鑰或任何複本金鑰，直到它被刪除為止。金鑰狀態會變更為 `Disabled`，但您可以重新啟用 KMS 金鑰。
4. 當最後一個複本金鑰的等待期間過期時，AWS KMS 會刪除最後一個複本金鑰。主要金鑰的金鑰狀態會從 `Pending replica deletion` (PendingReplicaDeletion) 變更為 `Pending deletion` (PendingDeletion)，並且主要金鑰的 7-30 天等待期開始。

```
KMS key:                    Key state:
Primary key (us-east-1)       Pending deletion (waiting period 30 days)
```



5. 等待期間過期時，AWS KMS 會刪除主要金鑰。

刪除具有複本之主索引鍵的時間下限為 14 天。

如果排定主要金鑰和所有複本金鑰的金鑰刪除等待期間為 7 天，則複本金鑰會在 7 天後刪除。主要金鑰會在第 14 天刪除。

- 第 1 天：排程刪除最短等待期限為 7 天的主要金鑰和複本金鑰。複本金鑰的 7 天刪除等待期間開始。主要金鑰的刪除等待期間尚未開始。
- 第 7 天：複本金鑰的刪除等待期間結束。AWS KMS 會刪除所有複本金鑰。刪除最後一個複本金鑰時，主要金鑰的 7 天刪除等待期間會開始。
- 第 14 天：主要金鑰的刪除等待期間結束。AWS KMS 會刪除主要金鑰。

您可以在 AWS CloudTrail 日誌中檢視動作記錄。AWS KMS 會記錄排程刪除每個 KMS 金鑰的操作和刪除 KMS 金鑰的動作。

刪除主要金鑰 (主控台)

若要刪除多區域主要金鑰，請使用下列處理程序。

排程金鑰刪除

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 選取您要刪除的主要金鑰旁邊的核取方塊。您也可以選取一或多個 KMS 金鑰，包括此主要金鑰的複本。
5. 選擇 Key actions (金鑰動作)、Schedule key deletion (排程金鑰刪除)。
6. 閱讀並考量於等待期間取消刪除的警告及資訊。如果決定取消刪除，請選擇 Cancel (取消)。
7. 對於 Waiting period (in days) (等候期間 (以天為單位))，輸入介於 7 和 30 之間的數字。如果選取多個 KMS 金鑰，則您選擇的等待期間會套用至所有選取的 KMS 金鑰。複本金鑰的等待期間會同時執行，但主要金鑰的等待期間會在 AWS KMS 刪除最後一個複本金鑰後開始。
8. 選取 Confirm that you want to delete this key in ***<number of days>*** days (確認您想要在 <number of days> 天內刪除此金鑰) 旁的核取方塊。
9. 選擇 Schedule deletion (排定刪除)。

若要在主要金鑰的詳細資訊頁面上查看 KMS 金鑰的刪除狀態，請參閱 General configuration (一般組態) 區段。金鑰狀態會出現在 Status (狀態) 欄位。當主要金鑰的金鑰狀態變更為 `Pending deletion` 時，會顯示 Scheduled deletion date (排定刪除日期)。

在任何多區域金鑰之詳細資訊頁面上的 Regionality (區域性) 索引標籤上，您還可以查看所有主要金鑰和複本金鑰的金鑰狀態 (Status (狀態))。如需詳細資訊，請參閱 檢視多區域金鑰。

刪除主要金鑰 (AWS KMS API)

若要刪除多區域複本金鑰，請使用此ScheduleKeyDeletion作業。若要指定 KMS 金鑰，請使用其金鑰 ID 或金鑰 ARN。當使用多區域金鑰時，您可以透過將金鑰 ARN 與其明確的區域值搭配使用來減少錯誤的發生率。

例如，此命令會刪除 us-east-1 (美國東部 (維吉尼亞北部) 區域中的主要金鑰。由於命令未指定等待期間，因此等待期間會設定為預設值 30 天。

```
$ aws kms schedule-key-deletion \
    --key-id arn:aws:kms:us-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab
```

命令成功時，會傳回金鑰 ARN、產生的金鑰狀態和等待期間 (PendingWindowInDays)。

如果主要金鑰沒有複本，則主要金鑰的金鑰狀態為 PendingDeletion，並且輸出包含 DeletionDate 欄位。如果有任何複本金鑰仍然存在，則主要金鑰的金鑰狀態為 PendingReplicaDeletion，且會省略 DeletionDate，因為它是不確定的。即使也排定了刪除複本金鑰，您也可以取消排定的刪除。

刪除多區域主要金鑰時，請務必確認金鑰 ARN 中的金鑰 ID 和區域值是您所預期的。

```
{
    "KeyId": "arn:aws:kms:us-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
    "KeyState": "PendingReplicaDeletion",
    "PendingWindowInDays": 30
}
```

若要檢查 KMS 金鑰的刪除狀態，請使用主索引鍵或任何剩餘複本金鑰的DescribeKey作業。在刪除最後一個複本，並且金鑰狀態變更為 PendingDeletion 之前，主要金鑰的等待期間計時不會開始。

若要計算主要金鑰的預期刪除日期，在回應中對複本金鑰 ARN 執行迴圈，對每個執行 DescribeKey，取得最新的 DeletionDate 值，然後為主要金鑰新增 PendingDeletionWindowInDays 值。複本金鑰的等待期間會同時執行。

在下列範例中，KMS 金鑰是具有現有複本金鑰的多區域主要金鑰。因為金鑰狀態為 PendingReplicaDeletion，所以回應包括等待期間 (PendingWindowInDays)，但不是 DeletionDate。主要金鑰的實際刪除日期取決於刪除複本金鑰的時間。

```
$ aws kms describe-key \
```

```
    --key-id arn:aws:kms:us-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab

{
    "KeyMetadata": {
        "AWSAccountId": "111122223333",
        "KeyId": "mrk-1234abcd12ab34cd56ef1234567890ab",
        "Arn": "arn:aws:kms:us-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
        "CreationDate": 1597902361.481,
        "Enabled": false,
        "Description": "",
        "KeySpec": "SYMMETRIC_DEFAULT",
        "KeyState": "PendingReplicaDeletion",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "Origin": "AWS_KMS",
        "KeyManager": "CUSTOMER",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ],
        "MultiRegion": true,
        "MultiRegionConfiguration": {
            "MultiRegionKeyType": "PRIMARY",
            "PrimaryKey": {
                "Arn": "arn:aws:kms:us-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                "Region": "us-east-1"
            },
            "ReplicaKeys": [
                {
                    "Arn": "arn:aws:kms:us-west-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                    "Region": "us-west-2"
                },
{
                    "Arn": "arn:aws:kms:eu-west-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                    "Region": "eu-west-1"
                },
                {
                    "Arn": "arn:aws:kms:ap-southeast-2:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                    "Region": "ap-southeast-2"
```

```
                }
            ]
        },
        "PendingDeletionWindowInDays": 30
    }
}
```

刪除所有複本時，DescribeKey 輸出會顯示金鑰狀態為 PendingDeletion 的剩餘主要金鑰。雖然金鑰狀態為 PendingDeletion，但 DeletionDate 欄位會出現，而不是 PendingWindowInDays 欄位。

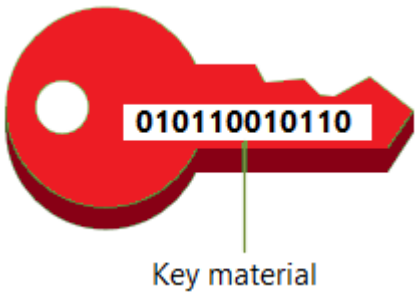
```
$ aws kms describe-key \
    --key-id arn:aws:kms:us-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab

{
    "KeyMetadata": {
        "AWSAccountId": "111122223333",
        "KeyId": "mrk-1234abcd12ab34cd56ef1234567890ab",
        "Arn": "",
        "CreationDate": 1597902361.481,
        "Enabled": false,
        "Description": "",
        "KeySpec": "SYMMETRIC_DEFAULT",
        "KeyState": "PendingDeletion",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "DeletionDate": 1597968000.0,
        "Origin": "AWS_KMS",
        "KeyManager": "CUSTOMER",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ],
        "MultiRegion": true,
        "MultiRegionConfiguration": {
            "MultiRegionKeyType": "PRIMARY",
            "PrimaryKey": {
                "Arn": "arn:aws:kms:us-east-1:111122223333:key/
mrk-1234abcd12ab34cd56ef1234567890ab",
                "Region": "us-east-1"
            },
            "ReplicaKeys": []
        }
```

```
    }
}
```

匯入 AWS KMS 金鑰的金鑰材料

您可利用您提供的金鑰資料來建立 AWS KMS keys (KMS 金鑰)。

KMS 金鑰是加密金鑰的邏輯表示。KMS 金鑰的中繼資料包括用於加密和解密資料的金鑰資料的 ID。當您建立 KMS 金鑰時， AWS KMS 預設會為該 KMS 金鑰產生金鑰材料。但是，您可以建立不含金鑰材料的 KMS 金鑰，然後將您自己的金鑰材料匯入該 KMS 金鑰。這項功能通常稱為「使用自有金鑰 (BYOK)」。



> **Note**
>
> AWS KMS 不支援解密以外的任何加 AWS KMS 密文字 AWS KMS，即使加密文字是使用匯入金鑰材料的 KMS 金鑰加密。 AWS KMS 不會發佈此工作所需的加密文字格式，且格式可能會變更，恕不另行通知。

所有類型的 KMS 金鑰都支援匯入的金鑰資料，但自訂金鑰存放區的 KMS 金鑰除外。然而，在中國區域，僅能匯入對稱加密金鑰資料至 KMS 金鑰。

使用匯入的關鍵材料時，您仍然對關鍵材料負責，同時允許 AWS KMS 使用該關鍵材料的複本。您可能因為以下一個或多個原因而選擇這樣做：

- 為證明您採用符合要求的熵來源產生金鑰資料。
- 將您自己基礎架構中的關鍵材料與 AWS 服務搭配使用，並用 AWS KMS 於管理其中關鍵材料的生命週期 AWS。
- 使用中現有且完善的金鑰 AWS KMS，例如用於程式碼簽署的金鑰、PKI 憑證簽署和憑證固定的應用程式

- 為中設定關鍵材料的到期時間 AWS 並手動刪除它，但也可以在 future 再次使用。反之，排程金鑰刪除需要等候 7 到 30 天，超過此期間將無法恢復已刪除的 KMS 金鑰。
- 擁有金鑰材料的原始副本，並在金鑰材料的 AWS 完整生命週期內將其保留在外面，以提高耐用性和災難復原。
- 對於非對稱金鑰和 HMAC 金鑰，匯入會建立在內外運作的相容且可互通的金鑰。 AWS

您可以使用匯入的金鑰材料稽核和監控 KMS 金鑰的使用和管理。 AWS KMS 當您建立 KMS 金鑰、下載包裝公開金鑰和匯入權杖，以及匯入金鑰材料時，會在記錄 AWS CloudTrail 檔中記錄事件。 AWS KMS 當您手動刪除匯入的金鑰材料或刪除過期的金鑰材料時，也 AWS KMS 會記錄事件。

如需有關 KMS 金鑰與匯入金鑰材料之間的重要差異資訊 AWS KMS，請參閱關於匯入的金鑰材料。

支援的 KMS 金鑰

AWS KMS 支援以下類型的 KMS 金鑰匯入金鑰材料。您無法匯入金鑰資料至自訂金鑰存放區的 KMS 金鑰。在中國區域，僅能匯入金鑰資料至對稱加密金鑰。

- 對稱加密 KMS 金鑰
- 非對稱 RSA KMS 金鑰 (用於加密或簽署，但不能同時使用)
- 非對稱橢圓曲線 (ECC) KMS 金鑰 (僅限簽署)
- HMAC KMS 金鑰
- 所有支援類型的多區域金鑰。

區域

所有 AWS KMS 支援的支援都支援匯入 AWS 區域 的關鍵材料。

在中國區域，僅能匯入金鑰資料至對稱加密 KMS 金鑰。此外，金鑰資料要求與其他區域不同。如需詳細資訊，請參閱 匯入金鑰材料步驟 3：加密金鑰材料。

主題

- 規劃匯入金鑰資料
- 受管匯入的金鑰資料
- 匯入金鑰材料步驟 1：建立不含金鑰材料的 AWS KMS key
- 匯入金鑰資料步驟 2：下載包裝公有金鑰及匯入字符
- 匯入金鑰材料步驟 3：加密金鑰材料

- 匯入金鑰材料步驟 4：匯入金鑰材料

規劃匯入金鑰資料

匯入的金鑰材料可讓您以產生的密碼編譯金鑰來保護 AWS 資源。您匯入的金鑰資料與特定 KMS 金鑰關聯。您可以將相同金鑰材料重新匯入相同的 KMS 金鑰，但是您無法將不同的金鑰材料匯入 KMS 金鑰，也無法將專為匯入金鑰材料設計的 KMS 金鑰轉換為包含金鑰材料的 KMS 金 AWS KMS 鑰。

進一步了解：

- the section called “選取包裝公有金鑰規格”
- the section called “選取包裝演算法”

主題

- 關於匯入的金鑰材料
- 保護匯入的金鑰資料
- 匯入金鑰材料的許可
- 匯入金鑰資料需求

關於匯入的金鑰材料

在決定將關鍵材料匯入至之前 AWS KMS，您應該瞭解匯入金鑰材料的下列特性。

您可以產生關鍵材料

您有責任使用符合您安全要求的隨機來源產生金鑰材料。

您可以刪除金鑰材料

您可以從 KMS 金鑰中刪除匯入的金鑰材料，立即使 KMS 金鑰無法使用。此外，當您將金鑰材料匯入 KMS 金鑰時，您可以決定金鑰是否會過期，並設定其過期時間。到期時間到達時， AWS KMS 會刪除關鍵材料。如果沒有金鑰材料，即無法在任何密碼編譯操作中使用 KMS 金鑰。若要還原金鑰，您必須將相同的金鑰材料重新匯入至金鑰。

您無法變更金鑰資料

當您將金鑰材料匯入 KMS 金鑰，KMS 金鑰將永久關聯到該金鑰材料。您可以重新匯入相同的金鑰材料，但您不能將不同的金鑰材料匯入 KMS 金鑰。此外，您無法為具有匯入金鑰材料的任何 KMS 金鑰啟用自動金鑰輪換。不過，您可以手動輪換具有匯入金鑰材料的 KMS 金鑰。

您無法變更金鑰資料來源

設計用於匯入金鑰材料的 KMS 金鑰具有無法變更的 EXTERNAL 來源值。您無法將匯入金鑰材料的 KMS 金鑰轉換為使用任何其他來源的金鑰材料，包括 AWS KMS. 同樣地，您無法將包含金鑰材料的 KMS 金 AWS KMS 鑰轉換為專為匯入金鑰材料所設計的金鑰。

您無法匯出金鑰資料

您無法匯出已匯入的任何關鍵材料。 AWS KMS 無法以任何形式將匯入的金鑰材料傳回給您。您必須在金鑰管理員 (例如硬體安全性模組 (HSM) 之外維護匯入金鑰材料的複本，如此一來，您才能在刪除金鑰材料或金鑰材料到期時重新匯入。 AWS

您可利用匯入金鑰資料來建立多區域金鑰。

具匯入金鑰資料的多區域擁有匯入金鑰資料的 KMS 金鑰功能，且可在 AWS 區域之間互通。若要利用匯入金鑰資料來建立多區域金鑰，您必須匯入相同金鑰資料至主要 KMS 金鑰以及每個複本金鑰。如需詳細資訊，請參閱 將金鑰材料匯入多區域金鑰。

非對稱金鑰與 HMAC 金鑰為可攜式且可互通

您可以在以外使用非對稱金鑰材料和 HMAC 金鑰材料，與具有相同匯入 AWS KMS 金鑰材料的 AWS 金鑰互操作。

與與算法中使用的 KMS 密鑰密不可分割的 AWS KMS 對稱密文不同，它 AWS KMS 使用標準 HMAC 和非對稱格式進行加密，簽名和 MAC 生成。因此，這些金鑰為可攜式，並且支援傳統委付金鑰案例。

當您的 KMS 金鑰匯入金鑰材料時，您可以在以外使用匯入的金鑰材料 AWS 來執行下列作業。

- HMAC 金鑰 — 您可利用匯入金鑰資料來驗證 HMAC KMS 金鑰所產生的 HMAC 標籤。您也可以將 HMAC KMS 金鑰與匯入的金鑰材料搭配使用，以驗證由外部金鑰材料所產生的 HMAC 標籤。 AWS
- 非對稱加密金鑰 — 您可以在以外使用私密非對稱加密金鑰， AWS 以對應的公開金鑰解密由 KMS 金鑰加密的加密文字。您也可以使用非對稱 KMS 金鑰來解密在外部產生的非對稱加密文字。 AWS
- 非對稱簽章金鑰 — 您可以將非對稱簽章 KMS 金鑰與匯入的金鑰材料搭配使用，驗證您在外部私人簽章金鑰所產生的數位簽章 AWS。您也可以在以外使用非對稱公開簽章金鑰 AWS 來驗證非對稱 KMS 金鑰所產生的簽章。

如您匯入相同金鑰資料至位於相同 AWS 區域的不同 KMS 金鑰，則這些金鑰也可互通。若要以不同的方式建立可互通的 KMS 金鑰 AWS 區域，請使用匯入的金鑰材料建立多區域金鑰。

對稱加密金鑰為不可攜式且不可互通

AWS KMS 產生的對稱密文不可移植或可互操作。 AWS KMS 不會發佈可攜性所需的對稱加密文字格式，且格式可能會變更，恕不另行通知。

- AWS KMS 無法解密您在以外加密的對稱密碼文本 AWS，即使您使用已導入的密鑰材料也是如此。
- AWS KMS 不支援解密以外的任何 AWS KMS 對稱加密文字 AWS KMS，即使加密文字是使用匯入金鑰材料的 KMS 金鑰加密。
- 具相同匯入金鑰資料的 KMS 金鑰無法互通。 AWS KMS 產生每個 KMS 金鑰專屬加密文字的對稱密文。此密文格式可保證僅加密該資料的 KMS 金鑰可解密。

此外，您無法使用任何 AWS 工具 (例如AWS Encryption SDK或 Amazon S3 用戶端加密) 來解密 AWS KMS 對稱加密文字。

因此，您無法將金鑰與匯入的金鑰材料搭配使用來支援金鑰委付安排，在此安排中，具有條件式存取金鑰資料的授權第三方可以解密以外的某些加密文字。 AWS KMS若要支援金鑰委付，請使用 AWS Encryption SDK 在獨立於 AWS KMS的金鑰下將您的訊息加密。

您必須為可用性和耐久性負責

AWS KMS 旨在保持進口關鍵材料的高可用性。但 AWS KMS 不會將進口關鍵材料的耐用性保持在與 AWS KMS 產生的關鍵材料相同的水平。如需詳細資訊，請參閱 保護匯入的金鑰資料。

保護匯入的金鑰資料

您匯入的金鑰資料在傳輸過程及靜態狀態都受到保護。在匯入金鑰材料之前，您必須使用在 FIPS 140-2 加密模組驗證程式下驗證的 AWS KMS 硬體安全模組 (HSM) 中產生的 RSA 金鑰組公開金鑰對公開金鑰材料進行加密 (或「包裝」)。您可利用包裝公有金鑰直接加密金鑰資料，或利用 AES 對稱金鑰來加密金鑰資料，然後利用 RSA 公有金鑰加密 AES 對稱金鑰。

收到後，使用 HSM 中對應的私 AWS KMS 密金鑰將金鑰材料解密，然後在僅存在於 AWS KMS HSM 揮發性記憶體中的 AES 對稱金鑰下重新加密金鑰。您的金鑰資料永遠不會以純文字形式離開 HSM。僅在使用中且僅在 AWS KMS HSM 中對其進行解密。

與匯入的金鑰資料搭配運用的 KMS 金鑰取決於您在 KMS 金鑰設定的存取控制政策。此外，您可利用別名與標籤來識別及控制 KMS 金鑰的存取權。您可啟用及停用金鑰、檢視及編輯其屬性，並利用類似 AWS CloudTrail的服務來加以監控。

然而，您要維護金鑰資料的唯一故障安全副本。為了回報這項額外的控制措施，您需要負責進口關鍵材料的耐用性和整體可用性。 AWS KMS 旨在保持進口關鍵材料的高可用性。但 AWS KMS 不會將進口關鍵材料的耐用性保持在與 AWS KMS 產生的關鍵材料相同的水平。

這種持久性差異在以下情況具有意義：

- 當您為匯入的金鑰材料設定到期時間時，請在金鑰材料到期後 AWS KMS 刪除該金鑰材料。 AWS KMS 不會刪除 KMS 金鑰或其中繼資料。您可以建立 Amazon CloudWatch 警示，在匯入的金鑰材料即將到期日時通知您。

  您無法刪除針對 KMS 金鑰 AWS KMS 產生的金鑰材料，也無法將金 AWS KMS 鑰材料設定為過期，但您可以旋轉它。
- 手動刪除匯入的金鑰材料時，會刪 AWS KMS 除金鑰材料，但不會刪除 KMS 金鑰或其中繼資料。反之，排程金鑰刪除需要 7 到 30 天的等待期間，超過此期限後 AWS KMS 會永久刪除 KMS 金鑰、其中繼資料及其金鑰材料。
- 在不太可能發生影響的區域範圍內故障 AWS KMS （例如完全喪失電源）時， AWS KMS 無法自動還原匯入的金鑰材料。但是， AWS KMS 可以還原 KMS 金鑰及其中繼資料。

您必須 AWS 在您控制的系統之外保留匯入關鍵材料的副本。建議您將匯入金鑰資料的可匯出複本儲存在金鑰管理系統，例如 HSM。如匯入的金鑰資料遭到刪除或到期，則在您重新匯入相同金鑰資料之前，其關聯的 KMS 金鑰將無法運用。如匯入的金鑰資料永久遺失，則利用 KMS 金鑰加密的任何密文均無法復原。

匯入金鑰材料的許可

若要使用匯入的金鑰材料建立和管理 KMS 金鑰，使用者需要此程序中的操作許可。您可以在建立 KMS 金鑰時提供金鑰政策中的 `kms:GetParametersForImport`、`kms:ImportKeyMaterial` 和 `kms:DeleteImportedKeyMaterial` 許可。在 AWS KMS 主控台中，當您使用外部金鑰材料來源建立金鑰時，會自動為金鑰管理員新增這些權限。

若要建立具有匯入金鑰材料的 KMS 金鑰，主體需要下列許可。

- 公里：CreateKey(IAM 政策)
  - 若要將此權限限制為具有匯入金鑰材料的 KMS 金鑰，請使用值為的 kms: KeyOrigin 原則條件`EXTERNAL`。

```
{
  "Sid": "CreateKMSKeysWithoutKeyMaterial",
```

```
  "Effect": "Allow",
  "Resource": "*",
  "Action": "kms:CreateKey",
  "Condition": {
    "StringEquals": {
      "kms:KeyOrigin": "EXTERNAL"
    }
  }
}
```

- 公里：GetParametersForImport(金鑰政策或 IAM 政策)
  - 若要將此權限限制為使用特定環繞演算法和包裝金鑰規格的要求，請使用 kms: WrappingAlgorithm 和 kms: WrappingKeySpec 原則條件。
- 公里：ImportKeyMaterial(金鑰政策或 IAM 政策)
  - 若要允許或禁止過期並控制到期日的金鑰材料，請使用 kms: ExpirationModel 和 kms: ValidTo 原則條件。

若要重新匯入匯入的金鑰材料，主體需要 kms: GetParametersForImport 和 kms: ImportKeyMaterial 權限。

若要刪除匯入的金鑰材料，主體需要 kms: DeleteImportedKeyMaterial 權限。

例如，若要准許範例 KMSAdminRole 管理具有匯入金鑰資料的 KMS 金鑰的所有方面，請在 KMS 金鑰的金鑰政策中包含如下所示的金鑰政策陳述式。

```
{
  "Sid": "Manage KMS keys with imported key material",
  "Effect": "Allow",
  "Resource": "*",
  "Principal": {
    "AWS": "arn:aws:iam::111122223333:role/KMSAdminRole"
  },
  "Action": [
    "kms:GetParametersForImport",
    "kms:ImportKeyMaterial",
    "kms:DeleteImportedKeyMaterial"
  ]
}
```

匯入金鑰資料需求

您匯入的金鑰資料必須與關聯 KMS 金鑰的金鑰規格相容。對於非對稱金鑰配對，只匯入配對的私密金鑰。 AWS KMS 衍生自私密金鑰的公開金鑰。

AWS KMS 支援以下包含匯入金鑰材料之 KMS 金鑰的金鑰規格。在中國區域，匯入的金鑰資料僅支援 SYMMETRIC_DEFAULT 金鑰規格。

KMS 金鑰之金鑰規格	金鑰資料要求
對稱加密金鑰 SYMMETRIC_DEFAULT	256 位元 (32 位元組) 的二進位資料 在中國區域，必須是 128 位元 (16 位元組) 的二進位資料。
HMAC 金鑰 HMAC_224 HMAC_256 HMAC_384 HMAC_512	HMAC 金鑰資料必須符合 RFC 2104。 金鑰長度必須符合金鑰規格所指定的長度。
RSA 非對稱私有金鑰 RSA_2048 RSA_3072 RSA_4096	您匯入的 RSA 非對稱私有金鑰必須是符合 RFC 3447 金鑰對的一部分。 模數：2048 位元、3072 位元或 4096 位元 素數：2 (不支持多素 RSA 金鑰) 非對稱金鑰資料必須是符合 RFC 5208 公有金鑰加密標準 (PKCS) #8 格式的 BER 編碼或 DER 編碼。
橢圓曲線非對稱私有金鑰 ECC_NIST_P256 (secp256r1) ECC_NIST_P384 (secp384r1)	您匯入的 ECC 非對稱私有金鑰必須是符合 RFC 5915 的一種金鑰對。 曲線：NIST P-256、NIST P-384、NIST P-521 或 Secp256k1

KMS 金鑰之金鑰規格	金鑰資料要求
ECC_NIST_P521 (secp521r1) ECC_SECG_P256K1 (secp256k1)	參數：僅限命名曲線 (拒絕具明確參數的 ECC 金鑰) 公有點座標：可壓縮、解壓縮或投影 非對稱金鑰資料必須是符合 RFC 5208 公有金鑰加密標準 (PKCS) #8 格式的 BER 編碼或 DER 編碼。

受管匯入的金鑰資料

這些主題將說明如何匯入及重新匯入金鑰資料至 KMS 金鑰，以及如何建立自動過期的匯入金鑰資料。

主題

- 匯入金鑰資料概觀
- 重新匯入金鑰資料
- 識別包含匯入金鑰資料的 KMS 金鑰
- 建立匯入金鑰材料到期的 CloudWatch 警示
- 刪除匯入的金鑰材料
- 刪除包含匯入金鑰資料的 KMS 金鑰

匯入金鑰資料概觀

以下概觀說明如何將金鑰材料匯入 AWS KMS。如需程序每個步驟的更多詳細資訊，請參閱對應的主題。

1. 建立不含金鑰資料的 KMS 金鑰 — 來源必須是 EXTERNAL。的金鑰來源EXTERNAL表示金鑰是針對匯入的金鑰材料所設計，而且無 AWS KMS 法產生 KMS 金鑰的金鑰材料。在後續步驟中，您會將您自己的金鑰材料匯入到這個 KMS 金鑰中。

   您匯入的金鑰材料必須與相關金鑰的金鑰規格相 AWS KMS 容。如需關於相容的詳細資訊，請參閱 the section called “匯入金鑰資料需求”。
2. 下載包裝公有金鑰及匯入字符 — 在完成步驟 1 之後，請下載包裝公有金鑰及匯入字符。這些項目會在匯入金鑰材料時保護您的金鑰材料 AWS KMS。

您可在此步驟選擇 RSA 包裝金鑰的類型 (「金鑰規格」)，以及用來加密資料的包裝演算法，以便將其傳輸至 AWS KMS。每次匯入或重新匯入相同金鑰資料時，您都可選擇不同包裝金鑰規格與包裝金鑰演算法。

3. 加密金鑰資料 — 利用您在步驟 2 下載的包裝公有金鑰來加密您在自己系統建立的金鑰資料。
4. 匯入金鑰材料 – 上傳您在步驟 3 建立的加密金鑰材料，以及您在步驟 2 下載的匯入字符。

   在此階段，您可以設定選用的到期時間。當匯入的金鑰材料到期時，將其 AWS KMS 刪除，KMS 金鑰就會變成無法使用。若要繼續運用 KMS 金鑰，您必須重新匯入相同的金鑰資料。

   當匯入操作成功完成後，KMS 金鑰的金鑰狀態會從 `PendingImport` 變成 `Enabled`。您現在可以在密碼編譯操作中使用 KMS 金鑰。

AWS KMS 當您 AWS CloudTrail 建立 KMS 金鑰、下載包裝公開金鑰和匯入權杖，以及匯入金鑰材料時，會在記錄中記錄項目。 AWS KMS 當您刪除匯入的金鑰材料或刪除過期的金鑰材料時，也 AWS KMS 會記錄項目。

重新匯入金鑰資料

如果您管理的 KMS 金鑰含有已匯入的金鑰資料，您可能需要重新匯入金鑰資料。您可能重新匯入金鑰材料來取代即將過期或已刪除的金鑰材料，或變更金鑰材料的過期模型或過期日期。

當您將金鑰材料匯入 KMS 金鑰，KMS 金鑰將永久關聯到該金鑰材料。您可以重新匯入相同的金鑰材料，但您不能將不同的金鑰材料匯入 KMS 金鑰。您無法輪換金鑰資料， AWS KMS 也無法為含有匯入金鑰資料的 KMS 金鑰建立金鑰資料。

您可以根據自己的安全性需求來排程或隨時重新匯入金鑰材料。不必等到金鑰材料到期或接近到期時間。

若要重新匯入金鑰材料，請使用首次匯入金鑰材料的相同程序，除了以下幾點例外。

- 使用現有的 KMS 金鑰，而不是建立新的 KMS 金鑰。您可以略過匯入程序的步驟 1。
- 重新匯入金鑰資料時，您可以變更過期模型和過期日期。

每次將金鑰材料匯入 KMS 金鑰時，您需要為 KMS 金鑰下載並使用新的包裝金鑰和匯入字符。包裝程序不會影響金鑰資料內容，因此您可採用不同包裝公有金鑰和不同包裝演算法來匯入相同金鑰資料。

識別包含匯入金鑰資料的 KMS 金鑰

當您建立不具有匯入金鑰材料的 KMS 金鑰時，KMS 金鑰 `Origin` 屬性的值為 `EXTERNAL`，而且無法變更。不同於金鑰狀態，`Origin` 值與是否有金鑰材料無關。

您可以使用 `EXTERNAL` 來源值來標識設計用於匯入金鑰材料的 KMS 金鑰。您可以在 AWS KMS 控制台中或使用DescribeKey操作查找密鑰來源。您也可以使用主控台或 API 來檢視金鑰材料的屬性，例如它是否過期以及何時過期。

識別具有匯入金鑰材料的 KMS 金鑰 (主控台)

1. 請在以下位置開啟 AWS KMS 主控台。 https://console.aws.amazon.com/kms
2. 若要變更 AWS 區域，請使用頁面右上角的「地區」選取器。
3. 使用下列其中一項技術來檢視 KMS 金鑰的 `Origin` 屬性。
   - 若要新增 Origin (來源) 欄到您的 KMS 金鑰資料表，請選擇右上角的 Settings (設定) 圖示。選擇 Origin (來源)，然後選擇 Confirm (確認)。來源欄可讓您輕鬆識別具外部 (匯入金鑰資料)來源屬性值的 KMS 金鑰。
   - 若要尋找特定 KMS 金鑰的 `Origin` 屬性值，請選擇 KMS 金鑰的金鑰 ID 或別名。然後選擇 Cryptographic configuration (密碼編譯組態) 索引標籤。索引標籤位於 General Configuration (一般組態) 區段下。
4. 若要檢視金鑰材料的詳細資訊，請選擇 Key material (金鑰材料) 索引標籤。此索引標籤只會顯示在具有匯入金鑰材料之 KMS 金鑰的詳細資料頁面上。

使用匯入的金鑰材料 (AWS KMS API) 識別 KMS 金鑰

使用DescribeKey操作。回應包含 KMS 金鑰的 `Origin` 屬性、過期模型和過期日期，如下列範例所示。

```
$  aws kms describe-key --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
{
    "KeyMetadata": {
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "Origin": "EXTERNAL",
        "ExpirationModel": "KEY_MATERIAL_EXPIRES"
        "ValidTo": 2023-06-05T12:00:00+00:00,
        "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "AWSAccountId": "111122223333",
        "CreationDate": 2018-06-09T00:06:50.831000+00:00,
```

```
        "Enabled": false,
        "MultiRegion": false,
        "Description": "",
        "KeyUsage": "ENCRYPT_DECRYPT",
        "KeyState": "PendingImport",
        "KeyManager": "CUSTOMER",
        "KeySpec": "SYMMETRIC_DEFAULT",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ]
    }
}
```

建立匯入金鑰材料到期的 CloudWatch 警示

您可以建立 CloudWatch 警示，在 KMS 金鑰中匯入的金鑰材料即將到期時通知您。例如，警示會在到期時間不到 30 天時通知您。

當您匯入金鑰資料至 KMS 金鑰時，您可以選擇指定該金鑰資料的到期日期和時間。當金鑰材料到期時， AWS KMS 會刪除金鑰材料，KMS 金鑰就會變成無法使用。若要再次使用 KMS 金鑰，您必須重新匯入金鑰資料。但是，如果您在金鑰資料到期前重新匯入，則可以避免中斷使用該 KMS 金鑰的程序。

此警示會使用 CloudWatch 針對 KMS 金`SecondsUntilKeyMaterialExpires`鑰發 AWS KMS 佈至的指標，其中包含匯入的金鑰材料到期。每個警示都會使用此指標來監控特定 KMS 金鑰的匯入金鑰資料。您無法為具有到期金鑰資料的所有 KMS 金鑰建立單一警示，也不能為您未來可能建立的 KMS 金鑰建立警示。

需求

監控匯入金鑰材料到期的 CloudWatch 警示需要下列資源。

- 具有到期匯入金鑰資料的 KMS 金鑰。如需協助，請參閱 識別包含匯入金鑰資料的 KMS 金鑰。
- Amazon SNS 主題。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的建立 Amazon SNS 主題。

建立警示

遵循使用下列必要值根據靜態臨界值建立 CloudWatch 警示中的指示進行。對於其他欄位，請接受預設值，並按要求提供名稱。

欄位	值
選取指標	選擇 KMS，然後選擇 Per-Key Metrics (每個金鑰指標)。 選擇包含 KMS 金鑰和 `SecondsUntilKeyMaterialExpires` 指標的資料列。然後選擇 Select metric (選取指標)。 對於具有會到期的匯入金鑰資料之 KMS 金鑰的指標，Metrics (指標) 清單只會顯示 `SecondsUntilKeyMaterialExpires` 指標。如果您在帳戶和區域中沒有具有這些屬性的 KMS 金鑰，則此清單為空白。
統計數字	下限
期間	1 分鐘
閾值類型	靜態
Whenever ...	每當####大於 1 時

刪除匯入的金鑰材料

您可以隨時刪除 KMS 金鑰中匯入的金鑰材料。此外，當匯入的金鑰材料有到期日期到期時， AWS KMS 會刪除金鑰材料。在兩者任一情況下，當金鑰資料遭刪除時，KMS 金鑰的金鑰狀態會變為待匯入，且在您重新匯入相同金鑰資料之前，無法在任何密碼編譯操作使用該 KMS 金鑰。(您無法匯入其他金鑰資料至 KMS 金鑰。)

除停用 KMS 金鑰及撤回權限外，刪除金鑰資料也可作為快速但暫時停止使用 KMS 金鑰的策略。反之，利用匯入的金鑰資料來排程刪除 KMS 金鑰也可快速停止運用 KMS 金鑰。然而，如在等待期間未取消刪除，則 KMS 金鑰、金鑰資料與所有金鑰中繼資料都會永久刪除。如需詳細資訊，請參閱 the section called “刪除包含匯入金鑰資料的 KMS 金鑰”。

若要刪除金鑰材料，您可以使用 AWS KMS 主控台或 DeleteImportedKeyMaterialAPI 作業。 AWS KMS 當您 AWS CloudTrail 刪除匯入的金鑰材料以及刪除過期的金鑰材料時，AWS KMS 會在記錄中記錄項目。

主題

- 刪除關鍵材料如何影響 AWS 服務
- 刪除金鑰材料 (主控台)

- 刪除金鑰材料AWS KMS

刪除關鍵材料如何影響 AWS 服務

當您刪除金鑰材料時，沒有金鑰材料的 KMS 金鑰立即變為無法使用 (視最終一致性而定)。不過，使用受 KMS 金鑰保護之資料金鑰所加密的資源不會受影響，除非再次使用 KMS 金鑰 (例如解密資料金鑰)。此問題會影響 AWS 服務，其中許多使用資料金鑰來保護您的資源。如需詳細資訊，請參閱 無法使用的 KMS 金鑰如何影響資料金鑰。

刪除金鑰材料 (主控台)

您可以使用刪 AWS Management Console 除關鍵材料。

1. 登入 AWS Management Console 並開啟 AWS Key Management Service (AWS KMS) 主控台，網址為 https://console.aws.amazon.com/kms。
2. 若要變更 AWS 區域，請使用頁面右上角的「地區」選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 執行以下任意一項：
   - 選取含有匯入金鑰資料的 KMS 金鑰的核取方塊。選擇 Key actions (金鑰動作)、Delete key material (刪除金鑰材料)。
   - 針對含有匯入金鑰資料的 KMS 金鑰，選擇其別名或金鑰 ID。選擇 Key material (金鑰材料) 索引標籤，然後選擇 Delete key material (刪除金鑰材料)。
5. 確認您要刪除金鑰材料，然後選擇 Delete key material (刪除金鑰材料)。KMS 金鑰的狀態 (對應其金鑰狀態) 會變更為 Pending import (待匯入)。

刪除金鑰材料AWS KMS

若要使用 AWS KMS API 刪除金鑰材料，請傳送DeleteImportedKeyMaterial要求。以下範例顯示如何使用 AWS CLI 執行此作業。

將 *1234abcd-12ab-34cd-56ef-1234567890ab* 替換為您要刪除其金鑰材料之 KMS 金鑰的金鑰 ID。您可以使用 KMS 金鑰的金鑰 ID 或 ARN，但不能對此操作使用別名。

```
$ aws kms delete-imported-key-material --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
```

刪除包含匯入金鑰資料的 KMS 金鑰

刪除包含匯入金鑰資料的 KMS 金鑰資料為暫時性且可復原。若要還原金鑰，請重新匯入其金鑰資料。

相對地，刪除 KMS 金鑰則無法復原。如果您排程金鑰刪除且所需的等待期到期，則會 AWS KMS 永久且不可逆轉地刪除 KMS 金鑰、其金鑰材料，以及與 KMS 金鑰相關聯的所有中繼資料。

然而，刪除包含匯入金鑰資料的 KMS 金鑰所產生的風險與後果取決於 KMS 金鑰的類型 (「金鑰規格」)。

- 對稱加密金鑰 — 如您刪除對稱加密 KMS 金鑰，則該金鑰加密的所有剩餘密文均無法復原。您無法建立新的對稱加密 KMS 金鑰來針對已刪除的對稱加密 KMS 金鑰解密其密文，即使您擁有相同金鑰資料也無法做到。每個 KMS 金鑰的唯一中繼資料會以密碼編譯方式繫結至每個對稱密文。此安全性功能可保證僅經加密對稱密文的 KMS 金鑰才能將其解密，但這會讓您無法重新建立對等 KMS 金鑰。
- 非對稱金鑰和 HMAC 金鑰 — 如果您擁有原始金鑰材料，您可以建立新的 KMS 金鑰，其密碼編譯內容與已刪除的非對稱或 HMAC KMS 金鑰相同。 AWS KMS 產生標準的 RSA 密文和簽名、ECC 簽名和 HMAC 標籤，這些標籤不包含任何唯一的安全性功能。此外，您也可在 AWS外部利用 HMAC 金鑰或非對稱金鑰對的私有金鑰。

  您利用相同非對稱或 HMAC 金鑰資料建立的新 KMS 金鑰將具有不同金鑰識別碼。您必須建立新的金鑰政策、重新建立任何別名，以及更新現有 IAM 政策與授權，以便參考新金鑰。

匯入金鑰材料步驟 1：建立不含金鑰材料的 AWS KMS key

在預設情況，當您建立 KMS 金鑰時，AWS KMS 會為您建立金鑰資料。若是要匯入您自己的金鑰材料，請從建立一個不含金鑰材料的 KMS 金鑰開始。然後再匯入金鑰材料。若要建立不含金鑰材料的 KMS 金鑰，請使用AWS KMS主控台或CreateKey作業。

若要建立不含金鑰資料的金鑰，請指定 `EXTERNAL` 的來源。KMS 金鑰的來源屬性不可變。建立後，即無法將設計用於匯入金鑰材料的 KMS 金鑰轉換為具有 AWS KMS 或任何其他來源之金鑰材料的 KMS 金鑰。

具有 `EXTERNAL` 來源且沒有金鑰材料的 KMS 金鑰的金鑰狀態為 `PendingImport`。KMS 金鑰可以無限期保持在 `PendingImport` 狀態。然而，您無法在密碼編譯操作採用 `PendingImport` 狀態的 KMS 金鑰。在匯入金鑰資料之後，KMS 金鑰的金鑰狀態會變為 `Enabled`，且您可在密碼編譯操作使用該 KMS 金鑰。

AWS KMS當您建立 KMS 金鑰、下載公開金鑰和匯入權杖，以及匯入金鑰材料時，會在記錄AWS CloudTrail檔中記錄事件。 AWS KMS當您刪除匯入的金鑰材料或刪除過期的金鑰材料時，也AWS KMS會記錄 CloudTrail 事件。

如需有關建立具有匯入金鑰材料之多區域金鑰的資訊，請參閱 將金鑰材料匯入多區域金鑰。

主題

- 建立不含金鑰材料的 KMS 金鑰 (主控台)
- 建立不含金鑰材料的 KMS 金鑰 (AWS KMS API)

建立不含金鑰材料的 KMS 金鑰 (主控台)

您只需為匯入的金鑰資料建立一次 KMS 金鑰。您可視需要無限次匯入及重新匯入相同金鑰資料至現有 KMS 金鑰，但您不能匯入不同金鑰資料至一個 KMS 金鑰。如需詳細資訊，請參閱 步驟 2：下載包裝公有金鑰及匯入字符。

若要在 Customer managed keys (客戶受管金鑰) 資料表中尋找具有匯入金鑰材料的現有 KMS 金鑰，請使用右上角的齒輪圖示在 KMS 金鑰清單中顯示 Origin (來源) 資料欄。匯入金鑰的來源值為外部 (匯入金鑰資料)。

若要建立包含匯入金鑰資料的 KMS 金鑰，請依照基本說明來開始建立偏好金鑰類型的 KMS 金鑰，但下列情況例外。

在選擇金鑰用途之後，請執行下列步驟：

1. 展開 Advanced options (進階選項)。
2. 對於 Key material origin (金鑰材料來源)，選擇 External (Import key material) (外部 (匯入金鑰材料))。
3. 然後選擇我了解採用已匯入金鑰的安全性及持久性隱憂旁邊的核取方塊，表示您了解使用匯入金鑰資料的隱憂。如要閱讀這些隱憂的相關資訊，請參閱 保護匯入的金鑰資料。
4. 返回基本說明。對於該類型的所有 KMS 金鑰，基本程序的其餘步驟均相同。

當您選擇完成時，您已建立不含金鑰資料的 KMS 金鑰，且狀態 (金鑰狀態) 為待匯入。

然而，主控台不會返回客戶自管金鑰表，而是顯示頁面讓您可在其中下載匯入金鑰資料所需的公有金鑰與匯入字符。您可立即繼續執行下載步驟，或選擇取消並在此時停止。您可隨時返回此下載步驟。

下一步：步驟 2：下載包裝公有金鑰及匯入字符。

建立不含金鑰材料的 KMS 金鑰 (AWS KMS API)

若要使用 AWS KMSAPI 建立不含金鑰材料的對稱加密 KMS 金鑰，請傳送Origin參數設定為EXTERNAL的CreateKey要求。以下範例顯示如何使用 AWS Command Line Interface (AWS CLI) 執行此作業。

```
$ aws kms create-key --origin EXTERNAL
```

如果命令成功執行，您會看到類似如下的輸出。AWS KMS 金鑰的 Origin 為 EXTERNAL 及其 KeyState 為 PendingImport。

> **Tip**
>
> 如果命令未成功，您可能看到 KMSInvalidStateException 或 NotFoundException。您可重試請求。

```
{
    "KeyMetadata": {
        "Origin": "EXTERNAL",
        "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "Description": "",
        "Enabled": false,
        "MultiRegion": false,
        "KeyUsage": "ENCRYPT_DECRYPT",
        "KeyState": "PendingImport",
        "CreationDate": 1568289600.0,
        "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "AWSAccountId": "111122223333",
        "KeyManager": "CUSTOMER",
        "KeySpec": "SYMMETRIC_DEFAULT",
        "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
        "EncryptionAlgorithms": [
            "SYMMETRIC_DEFAULT"
        ]
    }
}
```

從命令輸出複製 KeyId 值以供後續步驟使用，然後繼續進行 步驟 2：下載包裝公有金鑰及匯入字符。

> **Note**
>
> 此命令會利用 SYMMETRIC_DEFAULT 的 KeySpec 與 ENCRYPT_DECRYPT 的 KeyUsage 來建立對稱加密 KMS 金鑰。您可利用選用參數 --key-spec 與 --key-usage 來建立非對稱 HMAC KMS 金鑰。有關更多資訊，請參閱 CreateKey 操作。

匯入金鑰資料步驟 2：下載包裝公有金鑰及匯入字符

建立不AWS KMS key含金鑰材料的 RSA 後，請使用AWS KMS主控台或 GetParametersForImportAPI 下載 RSA 包裝公用金鑰和該 KMS 金鑰的匯入權杖。包裝公有金鑰與匯入字符是不可分割的集合，必須一起使用。

您將利用包裝公有金鑰來加密您的金鑰資料，以便進行傳輸。在下載之前，請在步驟 3 選取 RSA 包裝金鑰對的長度 (金鑰規格) 與包裝演算法，以便用來加密匯入的金鑰資料並進行傳輸。

每個包裝公有金鑰與匯入字符集合的有效期為 24 小時。如您未在下載後 24 小時內利用它們來匯入金鑰資料，則必須下載新的集合。您可隨時下載新的包裝公有金鑰與匯入字符集合。這可讓您更改 RSA 包裝金鑰長度 (「金鑰規格」) 或替取代遺失的集合。

您也可下載包裝公有金鑰與匯入字符集合，以便重新匯入相同金鑰資料至 KMS 金鑰。這樣做可設定或變更金鑰資料的過期時間，或者還原已過期或已刪除的金鑰資料。每次匯入金鑰資料至 AWS KMS 時，均須下載並重新加密金鑰資料。

運用包裝公有金鑰

下載內容包括一個公有金鑰 (也稱為包裝公有金鑰)，這是您 AWS 帳戶 的特有金鑰。

在匯入金鑰資料之前，您必須先利用公有包裝金鑰來加密金鑰資料，然後上傳已加密的金鑰資料至 AWS KMS。當 AWS KMS 收到您的加密金鑰資料時，其會利用對應的私有金鑰來解密金鑰資料，然後利用 AES 對稱金鑰重新加密金鑰資料，一切均在 AWS KMS 硬體安全模組 (HSM) 內進行。

使用匯入符記

該下載包括匯入字符，字符所帶的中繼資料可確保金鑰材料正確匯入。當您上傳加密的金鑰資料至 AWS KMS 時，您必須上傳在此步驟下載的相同匯入字符。

選取包裝公有金鑰規格

為在匯入期間保護金鑰資料，請利用從 AWS KMS 下載的包裝公有金鑰及支援的包裝演算法來加密金鑰資料。在下載包裝公有金鑰與匯入字符之前，請先選取金鑰規格。所有包裝金鑰對都會在 AWS KMS 硬體安全模組 (HSM) 產生。您的私有金鑰永遠不會以純文字形式離開 HSM。

包裝公有金鑰的金鑰規格決定 RSA 金鑰對的金鑰長度，該金鑰對在傳輸至 AWS KMS 的過程會保護您的金鑰資料。一般情況而言，建議採用符合實用的最長公有金鑰。我們提供數種包裝公有金鑰規格，以便支援各種 HSM 與金鑰管理器。

針對用於匯入所有類型金鑰資料的 RSA 包裝金鑰，AWS KMS 支援下列金鑰規格，除非另有說明。

- RSA_4096 (首選)
- RSA_3072
- RSA_2048

> **Note**
>
> 不支援下列組合：ECC_NIST_P521 金鑰資料、RSA_2048 公有包裝金鑰規格以及 RSAES_OAE_SHA_* 包裝演算法。
> 您無法使用 RSA_2048 公有包裝金鑰直接包裝 ECC_NIST_P521 金鑰材質。使用較大的包裝金鑰或 RSA_AES_KEY_WRAP_SHA_* 包裝演算法。

選取包裝演算法

為在匯入期間保護金鑰資料，請利用下載的包裝公有金鑰及支援的包裝演算法來加密金鑰資料。

AWS KMS 支援多種標準 RSA 包裝演算法與兩步驟混合包裝演算法。一般而言，建議採用相容匯入金鑰資料及包裝金鑰規格的最安全包裝演算法。您通常是選擇硬體安全模組 (HSM) 或保護金鑰資料的金鑰管理系統所支援的演算法。

下表顯示每種金鑰資料與 KMS 金鑰類型所支援的包裝演算法。演算法以偏好順序列出。

金鑰資料	支援的包裝演算法與規格
對稱加密金鑰	包裝演算法：
256 位元 AES 金鑰	RSAES_OAEP_SHA_256

金鑰資料	支援的包裝演算法與規格
128 位元 SM4 金鑰 (僅限中國區域)	RSAES_OAEP_SHA_1 已取代的包裝演算法： RSAES_PKCS1_V1 **Note** 截至 2023 年 10 月 10 日，AWS KMS不支援 RSAES_PK1_5 折繞演算法。 包裝金鑰規格： RSA_2048 RSA_3072 RSA_4096
非對稱 RSA 私有金鑰	包裝演算法： RSA_AES_KEY_WRAP_SHA_256 RSA_AES_KEY_WRAP_SHA_1 包裝金鑰規格： RSA_2048 RSA_3072 RSA_4096

金鑰資料	支援的包裝演算法與規格
非對稱橢圓曲線 (ECC) 私有金鑰 您不能搭配採用 RSAES_OAEP_SHA_* 包裝演算法與 RSA_2048 包裝金鑰規格來包裝 ECC_NIST_P521 金鑰資料。	包裝演算法： RSA_AES_KEY_WRAP_SHA_256 RSA_AES_KEY_WRAP_SHA_1 RSAES_OAEP_SHA_256 RSAES_OAEP_SHA_1 包裝金鑰規格： RSA_2048 RSA_3072 RSA_4096
HMAC 金鑰	包裝演算法： RSAES_OAEP_SHA_256 RSAES_OAEP_SHA_1 包裝金鑰規格： RSA_2048 RSA_3072 RSA_4096

- RSA_AES_KEY_WRAP_SHA_256 - 此為兩步驟混合包裝演算法，結合加密金鑰資料以及您產生的 AES 對稱金鑰，然後利用下載的 RSA 公有包裝金鑰與 RSAES_OAEP_SHA_256 包裝演算法來加密 AES 對稱金鑰。

  若要包裝 RSA 私有金鑰資料，需採用 RSA_AES_KEY_WRAP_SHA_* 包裝演算法。
- RSA_AES_KEY_WRAP_SHA_1 - 此為兩步驟混合包裝演算法，結合加密金鑰資料以及您產生的 AES 對稱金鑰，然後利用下載的 RSA 包裝公有金鑰與 RSAES_OAEP_SHA_1 包裝演算法來加密 AES 對稱金鑰。

若要包裝 RSA 私有金鑰資料，需採用 RSA_AES_KEY_WRAP_SHA_* 包裝演算法。

- RSAES_OAEP_SHA_256 – RSA 加密演算法搭配最佳非對稱加密填補 (OAEP) 以及 SHA-256 雜湊函數。
- RSAES_OAEP_SHA_1 – RSA 加密演算法搭配最佳非對稱加密填補 (OAEP) 以及 SHA-1 雜湊函數。
- RSAES_PKCS1_V1_5 (已取代；自 2023 年 10 月 10 日起，AWS KMS 不支援 RSAES_PKCS1_V1_V1_5 包裝演算法) - 此為 RSA 加密演算法搭配 PKCS #1 1.5 版定義的填補格式。

主題

下載包裝公有金鑰與匯入字符 (主控台)

您可利用 AWS KMS 主控台來下載包裝公有金鑰與匯入字符。

1. 如果您剛完成建立不含金鑰材料之 KMS 金鑰的步驟，且目前是在 Download wrapping key and import token (下載包裝金鑰和匯入字符) 頁面，請跳到 Step 9。
2. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
3. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
4. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。

   > **Tip**
   >
   > 您僅能匯入金鑰資料至來源是外部 (匯入金鑰資料) 的 KMS 金鑰。這表示建立的 KMS 金鑰不含金鑰資料。如要將 Origin (來源) 資料行新增至您的資料表，請在頁面的右上角選擇設定圖示 ()。開啟 Origin (來源)，然後選擇 Confirm (確認)。

5. 選擇待匯入的 KMS 金鑰的別名或金鑰 ID。
6. 選擇 Cryptographic configuration (密碼編譯組態) 索引標籤並檢視其值。索引標籤位於 General Configuration (一般組態) 區段下。

您僅能匯入金鑰資料至來源是外部 (匯入金鑰資料) 的 KMS 金鑰。如需有關建立具有匯入金鑰材料之 KMS 金鑰的資訊，請參閱 匯入 AWS KMS 金鑰的金鑰材料。

7. 選擇金鑰資料索引標籤，然後選擇匯入金鑰資料。

   僅來源值為外部 (匯入金鑰資料) 的 KMS 金鑰，才會顯示金鑰資料索引標籤。

8. 針對選取包裝金鑰規格，請選擇 KMS 金鑰的組態。在建立此金鑰後，您便無法變更金鑰規格。
9. 對於 Select wrapping algorithm (選取包裝演算法)，請選擇您將用來加密金鑰材料的選項。如需選項的詳細資訊，請參閱選取包裝演算法。
10. 選擇下載包裝公有金鑰與匯入字符，然後儲存檔案。

    如果您有 Next (下一步) 選項，現在請繼續程序，選擇 Next (下一步)。若要稍後再繼續，請選擇 Cancel (取消)。

11. 解壓縮您在上一個步驟中儲存的 `.zip` 檔案 (`Import_Parameters_<key_id>_<timestamp>`)。

    資料夾內含下列檔案：

    - RSA 包裝公有金鑰位於名為 `WrappingPublicKey.bin` 的檔案中。
    - 匯入字符位於名為 `ImportToken.bin` 的檔案。
    - 名為 README.txt 的文本檔案。此檔案包含包裝公有金鑰的相關資訊、用來加密金鑰資料的包裝演算法，以及包裝公有金鑰與匯入字符過期的日期及時間。

12. 若要繼續程序，請參閱加密金鑰材料。

下載包裝公有金鑰與匯入字符 (AWS KMS API)

若要下載公開金鑰和匯入權杖，請使用 GetParametersForImportAPI。指定將與匯入金鑰材料相關聯的 KMS 金鑰。此 KMS 金鑰的 Origin (來源) 值必須為 `EXTERNAL`。

此範例指定 `RSA_AES_KEY_WRAP_SHA_256` 包裝演算法、`RSA_3072` 包裝公有金鑰規格，以及金鑰 ID 範例。將這些範例值取代為下載的有效值。對於金鑰 ID，您可採用金鑰 ID 或金鑰 ARN，但您不能在此操作採用別名名稱或別名 ARN。

```
$ aws kms get-parameters-for-import \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --wrapping-algorithm RSA_AES_KEY_WRAP_SHA_256 \
    --wrapping-key-spec RSA_3072
```

如果命令成功執行，您會看到類似如下的輸出：

```
{
    "ParametersValidTo": 1568290320.0,
    "PublicKey": "public key (base64 encoded)",
    "KeyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "ImportToken": "import token (base64 encoded)"
}
```

為了準備下一步要使用的資料，將對公有金鑰和匯入字符進行 base64 解碼，然後將解碼的值保存在檔案中。

若要對公有金鑰和匯入字符進行 base64 解碼：

1. 複製 base64 編碼公有金鑰 (在範例輸出中表示為*#### (base64 ##)*)，將其貼入新檔案，然後儲存檔案。以描述性名稱命名檔案，例如 PublicKey.b64。
2. 使用 OpenSSL 來以 base64 解碼檔案的內容，並將解碼資料儲存到新的檔案。以下範例會解碼您在上一個步驟中所儲存檔案的資料 (PublicKey.b64)，並將輸出儲存到新的檔案，檔名為 WrappingPublicKey.bin。

   ```
   $ openssl enc -d -base64 -A -in PublicKey.b64 -out WrappingPublicKey.bin
   ```

3. 複製 base64 編碼匯入字符 (在範例輸出中表示為*#### (base64 ##)*)、將其貼入新檔案，然後儲存檔案。提供檔案描述性的名稱，例如 importtoken.b64。
4. 使用 OpenSSL 來以 base64 解碼檔案的內容，並將解碼資料儲存到新的檔案。以下範例會解碼您在上一個步驟中所儲存檔案的資料 (ImportToken.b64)，並將輸出儲存到新的檔案，檔名為 ImportToken.bin。

   ```
   $ openssl enc -d -base64 -A -in importtoken.b64 -out ImportToken.bin
   ```

繼續執行「步驟 3：加密金鑰材料」。

匯入金鑰材料步驟 3：加密金鑰材料

在您下載公有金鑰和匯入字符之後，請使用您下載的公有金鑰與您指定的包裝演算法來加密金鑰資料。如果您需要替換公有金鑰或匯入字符，或要變更包裝算法，則必須下載新的公有金鑰和匯入字符。如需 AWS KMS 支援之簽署演算法的相關資訊，請參閱 選取包裝公有金鑰規格 和 選取包裝演算法。

金鑰材料必須是二進位格式。如需詳細資訊，請參閱 匯入金鑰資料需求。

> **Note**
>
> 對於非對稱金鑰配對，請僅加密並匯入私有金鑰。 AWS KMS 從私有金鑰中衍生公有金鑰。
> 不支援下列組合：ECC_NIST_P521 金鑰資料、RSA_2048 公有包裝金鑰規格以及 RSAES_OAE_SHA_* 包裝演算法。
> 您無法使用 RSA_2048 公有包裝金鑰直接包裝 ECC_NIST_P521 金鑰材質。使用較大的包裝金鑰或 RSA_AES_KEY_WRAP_SHA_* 包裝演算法。

一般而言，當您從硬體安全模組 (HSM) 或金鑰管理系統匯出金鑰材料，您會將其加密。如需如何以二進位格式匯出金鑰材料的詳細資訊，請參閱您的 HSM 或金鑰管理系統的文件。您也可以參閱以下章節，其中提供使用 OpenSSL 的概念驗證示範。

當加密金鑰資料時，請使用您在下載包裝公有金鑰與匯入字符時所指定的包裝演算法。若要尋找您指定的環繞演算法，請參閱關聯GetParametersForImport要求的 CloudTrail 記錄事件。

產生用於測試的金鑰資料

下列 OpenSSL 指令會產生每種受支援類型的金鑰資料以供測試。這些範例僅供測試和 proof-of-concept 示範使用。對於生產系統，請使用更安全方法 (例如商業 HSM 或金鑰管理系統) 來產生並存放您的金鑰資料。

若要將非對稱金鑰配對的私有金鑰轉換為 DER 編碼格式，請將金鑰資料產生命令傳輸至下列 `openssl pkcs8` 命令。此 `topk8` 參數會指示 OpenSSL 取得私有金鑰做為匯入，並傳回 PKCS #8 格式化的金鑰。(預設行為是相反。)

```
openssl pkcs8 -topk8 -outform der -nocrypt
```

下列命令會為每個受支援的金鑰類型產生測試金鑰資料。

- 對稱加密金鑰 (32 位元組)

  此命令會產生 256 位元對稱金鑰 (32 位元組隨機字串)，並將其儲存在 `PlaintextKeyMaterial.bin` 檔案。您不需要對此金鑰資料進行編碼。

  ```
  openssl rand -out PlaintextKeyMaterial.bin 32
  ```

  僅在中國區域，您必須產生 128 位元對稱金鑰 (16 位元組隨機字串)。

```
openssl rand -out PlaintextKeyMaterial.bin 16
```

- HMAC 金鑰

  此命令產生指定大小的隨機位元組字串。您不需要對此金鑰資料進行編碼。

  HMAC 金鑰的長度必須符合 KMS 金鑰規格所定義的長度。例如，如果 KMS 金鑰是 HMAC_384，您必須匯入 384 位元 (48 位元組) 金鑰。

```
openssl rand -out HMAC_224_PlaintextKey.bin 28

openssl rand -out HMAC_256_PlaintextKey.bin 32

openssl rand -out HMAC_384_PlaintextKey.bin 48

openssl rand -out HMAC_512_PlaintextKey.bin 64
```

- RSA 私有金鑰

```
openssl genpkey -algorithm rsa -pkeyopt rsa_keygen_bits:2048 | openssl pkcs8 -topk8 -
outform der -nocrypt > RSA_2048_PrivateKey.der

openssl genpkey -algorithm rsa -pkeyopt rsa_keygen_bits:3072 | openssl pkcs8 -topk8 -
outform der -nocrypt > RSA_3072_PrivateKey.der

openssl genpkey -algorithm rsa -pkeyopt rsa_keygen_bits:4096 | openssl pkcs8 -topk8 -
outform der -nocrypt > RSA_4096_PrivateKey.der
```

- ECC 私有金鑰

```
openssl genpkey -algorithm ec -pkeyopt ec_paramgen_curve:P-256 | openssl pkcs8 -topk8
 -outform der -nocrypt > ECC_NIST_P256_PrivateKey.der

openssl genpkey -algorithm ec -pkeyopt ec_paramgen_curve:P-384 | openssl pkcs8 -topk8
 -outform der -nocrypt > ECC_NIST_P384_PrivateKey.der

openssl genpkey -algorithm ec -pkeyopt ec_paramgen_curve:P-521 | openssl pkcs8 -topk8
 -outform der -nocrypt > ECC_NIST_P521_PrivateKey.der

openssl genpkey -algorithm ec -pkeyopt ec_paramgen_curve:secp256k1 | openssl pkcs8 -
topk8 -outform der -nocrypt > ECC_SECG_P256K1_PrivateKey.der
```

範例：使用 OpenSSL 加密金鑰資料

下列範例說明如何使用 OpenSSL 以下載的公有金鑰來加密金鑰內容。

> **Important**
>
> 此範例僅為概念驗證示範。對於生產系統，請使用更安全方法 (例如商業 HSM 或金鑰管理系統) 來產生和存放您的金鑰材料。
> 不支援下列組合：ECC_NIST_P521 金鑰資料、RSA_2048 公有包裝金鑰規格以及 RSAES_OAE_SHA_* 包裝演算法。
> 您無法使用 RSA_2048 公有包裝金鑰直接包裝 ECC_NIST_P521 金鑰材質。使用較大的包裝金鑰或 RSA_AES_KEY_WRAP_SHA_* 包裝演算法。

RSAES_OAEP_SHA_1

AWS KMS 支援對稱式加密金鑰 (SYMMETRIC_DEFAULT)、橢圓曲線 (ECC) 私有金鑰與 HMAC 金鑰支援 RSAES_OAEP_SHA_1。

RSAES_OAEP_SHA_1 不支援 RSA 私有金鑰。此外，您不能將 RSA_2048 公有包裝金鑰與任何 RSAES_OAEP_SHA_* 包裝演算法搭配使用，來包裝 ECC_NIST_P521 (secp521r1) 私有金鑰。您必須使用較大的公有包裝金鑰或 RSA_AES_KEY_WRAP 包裝演算法。

下列範例會使用您下載的公有金鑰與 RSAES_OAEP_SHA_1 包裝演算法來加密金鑰資料，並將其儲存在 `EncryptedKeyMaterial.bin` 檔案。

在此範例中：

- *`WrappingPublicKey.bin`* 是包含下載的包裝公有金鑰的檔案。
- *`PlaintextKeyMaterial.bin`* 是包含您正在加密的金鑰資料的檔案，例如 `PlaintextKeyMaterial.bin`、`HMAC_384_PlaintextKey.bin` 或 `ECC_NIST_P521_PrivateKey.der`。

```
$ openssl pkeyutl \
    -encrypt \
    -in PlaintextKeyMaterial.bin \
    -out EncryptedKeyMaterial.bin \
    -inkey WrappingPublicKey.bin \
    -keyform DER \
```

```
    -pubin \
    -pkeyopt rsa_padding_mode:oaep \
    -pkeyopt rsa_oaep_md:sha1
```

RSAES_OAEP_SHA_256

AWS KMS 支援對稱式加密金鑰 (SYMMETRIC_DEFAULT)、橢圓曲線 (ECC) 私有金鑰與 HMAC 金鑰支援 RSAES_OAEP_SHA_256。

RSAES_OAEP_SHA_256 不支援 RSA 私有金鑰。此外，您不能將 RSA_2048 公有包裝金鑰與任何 RSAES_OAEP_SHA_* 包裝演算法搭配使用，來包裝 ECC_NIST_P521 (secp521r1) 私有金鑰。您必須使用較大的公有金鑰或 RSA_AES_KEY_WRAP 包裝演算法。

下列範例會使用您下載的公有金鑰與 RSAES_OAEP_SHA_256 包裝演算法來加密金鑰資料，並將其儲存在 EncryptedKeyMaterial.bin 檔案。

在此範例中：

- *WrappingPublicKey.bin* 是包含下載的公有包裝金鑰的檔案。如果從主控台下載公有金鑰，這個檔案名稱為 wrappingKey_*KMS key_key_ID_timestamp* (例如 wrappingKey_f44c4e20-f83c-48f4-adc6-a1ef38829760_0809092909)。
- *PlaintextKeyMaterial.bin* 是包含您正在加密金鑰資料的檔案，例如 PlaintextKeyMaterial.bin、HMAC_384_PlaintextKey.bin 或 ECC_NIST_P521_PrivateKey.der。

```
$ openssl pkeyutl \
    -encrypt \
    -in PlaintextKeyMaterial.bin \
    -out EncryptedKeyMaterial.bin \
    -inkey WrappingPublicKey.bin \
    -keyform DER \
    -pubin \
    -pkeyopt rsa_padding_mode:oaep \
    -pkeyopt rsa_oaep_md:sha256 \
    -pkeyopt rsa_mgf1_md:sha256
```

RSA_AES_KEY_WRAP_SHA_1

此 RSA_AES_KEY_WRAP_SHA_1 RSA 包裝演算法涉及兩個加密操作。

1. 使用您產生的 AES 對稱金鑰與 AES 對稱加密演算法來加密金鑰資料。

2. 將您所下載的公有金鑰與 RSAES_OAEP_SHA_1 包裝演算法搭配使用的 AES 對稱金鑰加密。

AWS KMS 支援 RSA_AES_KEY_WRAP_SHA_* 包裝演算法，適用於所有受支援的匯入金鑰資料類型及所有支援的公有金鑰規格。RSA_AES_KEY_WRAP_SHA_* 演算法是唯一支援包裝 RSA 金鑰材質的包裝演算法。

The RSA_AES_KEY_WRAP_SHA_1 包裝演算法需要 OpenSSL 版本 3。x 或更新版本。

1. 產生 256 位元 AES 對稱加密金鑰

   此命令會產生由 256 個隨機位元組成的 AES 對稱加密金鑰，並將其儲存在 aes-key.bin 檔案

   ```
   # Generate a 32-byte AES symmetric encryption key
   $ openssl rand -out aes-key.bin 32
   ```

2. 使用 AES 對稱加密金鑰加密您的金鑰資料

   此命令會使用 AES 對稱加密金鑰加密金鑰，並將加密的金鑰資料儲存在 key-material-wrapped.bin 檔案。

   以下是範例命令：

   - *PlaintextKeyMaterial.bin* 是包含您要匯入金鑰資料的檔案，例如 PlaintextKeyMaterial.bin、HMAC_384_PlaintextKey.bin、RSA_3072_PrivateKey.
     或 ECC_NIST_P521_PrivateKey.der。
   - *aes-key.bin* 是包含您在上一個命令中產生的 256 位元 AES 對稱加密金鑰的檔案。

   ```
   # Encrypt your key material with the AES symmetric encryption key
   $ openssl enc -id-aes256-wrap-pad \
           -K "$(xxd -p < aes-key.bin | tr -d '\n')" \
           -iv A65959A6 \
           -in PlaintextKeyMaterial.bin\
           -out key-material-wrapped.bin
   ```

3. 使用公有金鑰加密 AES 對稱加密金鑰

   此命令使用您下載的公有金鑰與 RSAES_OAEP_SHA_1 包裝演算法對您的 AES 對稱加密金鑰進行加密，並將其保存在 aes-key-wrapped.bin 檔案。

以下是範例命令：

- *WrappingPublicKey.bin* 是包含下載的公有包裝金鑰的檔案。如果從主控台下載公有金鑰，此檔案名稱為 wrappingKey_*KMS key_key_ID_timestamp* (例如 wrappingKey_f44c4e20-f83c-48f4-adc6-a1ef38829760_0809092909)。
- *aes-key.bin* 是包含您在此範例序列的第一個命令所產生的 256 位元 AES 對稱加密金鑰的檔案。

```
# Encrypt your AES symmetric encryption key with the downloaded public key
$ openssl pkeyutl \
    -encrypt \
    -in aes-key.bin \
    -out aes-key-wrapped.bin \
    -inkey WrappingPublicKey.bin \
    -keyform DER \
    -pubin \
    -pkeyopt rsa_padding_mode:oaep \
    -pkeyopt rsa_oaep_md:sha1 \
    -pkeyopt rsa_mgf1_md:sha1
```

4. 產生要匯入的檔案

將檔案與加密的金鑰資料及檔案與加密的 AES 金鑰連接起來。將它們儲存在 EncryptedKeyMaterial.bin 檔案，這是您將匯入在 步驟 4：匯入金鑰材料 檔案 。

以下是範例命令：

- *key-material-wrapped.bin* 是包含加密金鑰資料的檔案。
- *aes-key-wrapped.bin* 是包含加密 AES 加密金鑰的檔案。

```
# Combine the encrypted AES key and encrypted key material in a file
$ cat aes-key-wrapped.bin key-material-wrapped.bin > EncryptedKeyMaterial.bin
```

RSA_AES_KEY_WRAP_SHA_256

此 RSA_AES_KEY_WRAP_SHA_256 RSA 包裝演算法涉及兩個加密操作。

1. 使用您產生的 AES 對稱金鑰與 AES 對稱加密演算法來加密金鑰資料。

2. 將您所下載的公有金鑰與 RSAES_OAEP_SHA_256 包裝演算法搭配使用的 AES 對稱金鑰加密。

AWS KMS 支援 RSA_AES_KEY_WRAP_SHA_* 包裝演算法，適用於所有受支援的匯入金鑰資料類型及所有支援的公有金鑰規格。RSA_AES_KEY_WRAP_SHA_* 演算法是唯一支援包裝 RSA 金鑰材質的包裝演算法。

The RSA_AES_KEY_WRAP_SHA_256 包裝演算法需要 OpenSSL 版本 3。x 或更新版本。

1. 產生 256 位元 AES 對稱加密金鑰

   此命令會產生由 256 個隨機位元組成的 AES 對稱加密金鑰，並將其儲存在 aes-key.bin 檔案

```
# Generate a 32-byte AES symmetric encryption key
$ openssl rand -out aes-key.bin 32
```

2. 使用 AES 對稱加密金鑰加密您的金鑰資料

   此命令會使用 AES 對稱加密金鑰加密金鑰，並將加密的金鑰資料儲存在 key-material-wrapped.bin 檔案。

   以下是範例命令：

   - *PlaintextKeyMaterial.bin* 是包含您要匯入金鑰資料的檔案，例如 PlaintextKeyMaterial.bin、HMAC_384_PlaintextKey.bin、RSA_3072_PrivateKey.
     或 ECC_NIST_P521_PrivateKey.der。
   - *aes-key.bin* 是包含您在上一個命令中產生的 256 位元 AES 對稱加密金鑰的檔案。

```
# Encrypt your key material with the AES symmetric encryption key
$ openssl enc -id-aes256-wrap-pad \
        -K "$(xxd -p < aes-key.bin | tr -d '\n')" \
        -iv A65959A6 \
        -in PlaintextKeyMaterial.bin\
        -out key-material-wrapped.bin
```

3. 使用公有金鑰加密 AES 對稱加密金鑰

   此命令使用您下載的公有金鑰與 RSAES_OAEP_SHA_256 包裝演算法對您的 AES 對稱加密金鑰進行加密，並將其保存在 aes-key-wrapped.bin 檔案。

以下是範例命令：

- *WrappingPublicKey.bin* 是包含下載的公有包裝金鑰的檔案。如果從主控台下載公有金鑰，此檔案名稱為 wrappingKey_*KMS key_key_ID_timestamp* (例如 wrappingKey_f44c4e20-f83c-48f4-adc6-a1ef38829760_0809092909)。
- *aes-key.bin* 是包含您在此範例序列的第一個命令所產生的 256 位元 AES 對稱加密金鑰的檔案。

```
# Encrypt your AES symmetric encryption key with the downloaded public key
$ openssl pkeyutl \
    -encrypt \
    -in aes-key.bin \
    -out aes-key-wrapped.bin \
    -inkey WrappingPublicKey.bin \
    -keyform DER \
    -pubin \
    -pkeyopt rsa_padding_mode:oaep \
    -pkeyopt rsa_oaep_md:sha256 \
    -pkeyopt rsa_mgf1_md:sha256
```

4. 產生要匯入的檔案

將檔案與加密的金鑰資料及檔案與加密的 AES 金鑰連接起來。將它們儲存在 EncryptedKeyMaterial.bin 檔案，這是您將匯入在 步驟 4：匯入金鑰材料 檔案 。

以下是範例命令：

- *key-material-wrapped.bin* 是包含加密金鑰資料的檔案。
- *aes-key-wrapped.bin* 是包含加密 AES 加密金鑰的檔案。

```
# Combine the encrypted AES key and encrypted key material in a file
$ cat aes-key-wrapped.bin key-material-wrapped.bin > EncryptedKeyMaterial.bin
```

繼續執行「步驟 4：匯入金鑰材料」。

匯入金鑰材料步驟 4：匯入金鑰材料

加密您的金鑰材料之後，您可以匯入金鑰材料以便搭配使用 AWS KMS key。若要匯入金鑰材料，請上傳在步驟 3：加密金鑰材料加密的金鑰材料，以及在步驟 2：下載包裝公有金鑰及匯入字符下載的匯入符記。您必須將金鑰材料匯入至您在下載公有金鑰和匯入字符時指定的同一個 KMS 金鑰。在成功匯入金鑰資料之後，KMS 金鑰的金鑰狀態會變為 Enabled，這表示您可在密碼編譯操作使用 KMS 金鑰。

在匯入金鑰材料時，您可以為金鑰材料設定可選的過期時間。當金鑰材料過期時，AWS KMS 會刪除金鑰材料，讓 KMS 金鑰變成不可用。若要在密碼編譯操作中使用 KMS 金鑰，您必須重新匯入相同的金鑰材料。匯入金鑰材料後，您無法設定、變更或取消目前匯入材料的過期日期。若要變更這些值，您必須刪除並重新匯入相同的金鑰材料。

若要匯入金鑰材料，您可以使用AWS KMS主控台或 ImportKeyMaterialAPI。您可以透過提出 HTTP 請求，或透過使用 AWS 開發套件、AWS Command Line Interface 或 AWS Tools for PowerShell，直接使用 API。

當您匯入金鑰材料時，會在記錄AWS CloudTrail檔中加入一個ImportKeyMaterial項目以記錄ImportKeyMaterial操作。無論您使用AWS KMS控制台還是 AWS KMS API， CloudTrail 項目都是相同的。

設定到期時間 (選用)

匯入 KMS 金鑰的金鑰材料時，可以為金鑰材料設定選用的到期日期和時間，從匯入日期算起最多 365 天。匯入的金鑰材料到期時，AWS KMS 會刪除它。此動作會將 KMS 金鑰的金鑰狀態變更為 PendingImport，這會防止在任何密碼編譯操作中對其進行使用。若要使用 KMS 金鑰，您必須重新匯入原始金鑰材料的複本。

確保匯入的金鑰材料頻繁地到期，可協助您滿足法規要求，但這會對使用 KMS 金鑰加密的資料帶來額外的風險。在您重新匯入原始金鑰材料的複本之前，包含過期金鑰材料的 KMS 金鑰將無法使用，而且使用該 KMS 金鑰進行加密的任何資料都無法存取。如果您因任何原因而無法重新匯入金鑰材料，包括遺失原始金鑰材料的複本，則該 KMS 金鑰將永久無法使用，而且使用該 KMS 金鑰進行加密的資料將無法復原。

若要降低此風險，請確保已匯入金鑰材料的複本可供存取，並設計一個系統，以便在金鑰材料到期和中斷您的 AWS 工作負載之前刪除並重新將其匯入。建議您為已匯入的金鑰材料到期設定一個警示，讓您有足夠的時間在金鑰材料到期前重新匯入。您還可以使用 CloudTrail 日誌來審核導入（和重新導入）密鑰材料和刪除導入的密鑰材料的操作，以及刪除過期密鑰材料的AWS KMS操作。

您無法將不同的金鑰材料匯入 KMS 金鑰，AWS KMS 也無法還原、復原或重新產生已刪除的金鑰材料。您可以透過程式設計方式定期刪除和重新匯入已匯入的金鑰材料，而不是設定到期時間，但是保留原始金鑰材料複本的要求是相同的。

您可以在匯入金鑰材料時確定金鑰材料是否有到期時間以及何時到期。但是您可以開啟和關閉到期時間，或者透過刪除和重新匯入金鑰材料來設定新的到期時間。使用的`ExpirationModel`參數可ImportKeyMaterial開啟 (`KEY_MATERIAL_EXPIRES`) 和 off (`KEY_MATERIAL_DOES_NOT_EXPIRE`)，並使用`ValidTo`參數來設定到期時間。最長時間為匯入資料後 365 天；沒有最低限度，但時間必須為將來的時間。

匯入金鑰材料 (主控台)

您可以使用 AWS Management Console 來匯入金鑰材料。

1. 如您在 Upload your wrapped key material (上傳包裝的金鑰資料) 頁面，請跳至 Step 8。
2. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
3. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
4. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
5. 針對您已下載公開金鑰和匯入字符的 KMS 金鑰，選擇其金鑰 ID 或別名。
6. 選擇 Cryptographic configuration (密碼編譯組態) 索引標籤並檢視其值。這些索引標籤位於 General configuration (一般組態) 區段下 KMS 金鑰的詳細資訊頁面上。

   您僅能將金鑰資料匯入 Origin (來源) 是 EXTERNAL (外部 (匯入金鑰材料)) 的 KMS 金鑰。如需有關建立具有匯入金鑰材料之 KMS 金鑰的資訊，請參閱 匯入 AWS KMS 金鑰的金鑰材料。
7. 選擇金鑰資料索引標籤，然後選擇匯入金鑰資料。僅 Origin (來源) 值為External (Import key material) (外部 (匯入金鑰資料)) 的 KMS 金鑰，才會顯示金鑰資料索引標籤。

   如您已下載金鑰資料、匯入字符並加密金鑰資料，請選擇 Next (下一步)。
8. 在 Encrypted key material and import token (加密金鑰資料與匯入字符) 區段，執行下列動作。
   a. 在 Wrapped key material (包裝的金鑰資料)，選擇 Choose file (選擇檔案)。然後，上傳內含包裝 (加密) 金鑰資料的檔案。
   b. 在 Import token (匯入字符)，選擇 Choose file (上傳檔案)。上傳內含您下載之匯入符記的檔案。

9. 在 Expiration option (過期選項) 區段中，您可以決定金鑰材料是否會過期。若要設定過期日期和時間，請選擇 Key material expires (金鑰資料過期)，然後使用行事曆選取日期和時間。您可以指定自目前日期和時間起的日期，但最長不超過 365 天。
10. 選擇 Upload key material (上傳金鑰資料)。

匯入金鑰材料 (AWS KMS API)

若要匯入關鍵材料，請使用此ImportKeyMaterial作業。以下範例使用 AWS CLI，但您可以使用任何支援的程式設計語言。

若要使用此範例：

1. 將 *1234abcd-12ab-34cd-56ef-1234567890ab* 取代為您在下載公有金鑰和匯入字符時指定的 KMS 金鑰的金鑰 ID。若要識別 KMS 金鑰，請使用它的金鑰 ID 或金鑰 ARN。不能對此操作使用別名或別名 ARN。
2. 將 *EncryptedKeyMaterial.bin* 取代為包含加密金鑰材料的檔案名稱。
3. 將 *ImportToken.bin* 取代為包含匯入符記的檔案名稱。
4. 如果希望匯入的金鑰材料過期，請將 `expiration-model` 參數設定為其預設值 `KEY_MATERIAL_EXPIRES`，或省略 `expiration-model` 參數。然後，將 `valid-to` 參數的值取代為您希望金鑰材料到期的日期和時間。日期和時間最多可以是自提出請求之日起的 365 天內。

```
$ aws kms import-key-material --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --encrypted-key-material fileb://EncryptedKeyMaterial.bin \
    --import-token fileb://ImportToken.bin \
    --expiration-model KEY_MATERIAL_EXPIRES \
    --valid-to 2023-06-17T12:00:00-08:00
```

如果不希望匯入的金鑰材料過期，請將 `expiration-model` 參數的值設定為 `KEY_MATERIAL_DOES_NOT_EXPIRE`，並省略命令中的 `valid-to` 參數。

```
$ aws kms import-key-material --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --encrypted-key-material fileb://EncryptedKeyMaterial.bin \
    --import-token fileb://ImportToken.bin \
    --expiration-model KEY_MATERIAL_DOES_NOT_EXPIRE
```

> **Tip**
>
> 如果命令未成功，您可能看到 `KMSInvalidStateException` 或 `NotFoundException`。您可重試請求。

自訂金鑰存放區

金鑰存放區是用於存放密碼編譯金鑰的安全位置。AWS KMS 中的預設金鑰存放區也支援產生及管理其存放金鑰所需的方法。在預設情況下，您在 AWS KMS 中建立的 AWS KMS keys 的密碼編譯金鑰材料會由硬體安全模組 (HSM) (通過 FIPS 140-2 驗證的密碼編譯模組) 產生並保護。KMS 金鑰的金鑰材料永遠不會讓 HSM 處於未加密狀態。

然而，如果需要更多的 HSM 控制權，您可以建立自訂金鑰存放區。

自訂金鑰存放區是 AWS KMS 內的邏輯金鑰存放區，由您擁有和管理的 AWS KMS 之外的金鑰管理器支援。自訂金鑰存放區結合了 AWS KMS 的方便且完整的金鑰管理介面，以及擁有和控制金鑰材料和密碼編譯操作的能力。當您使用自訂金鑰存放區中的 KMS 金鑰時，您的金鑰管理器將使用您的密碼編譯金鑰執行密碼編譯操作。因此，您需要對密碼編譯金鑰的可用性和耐久性以及 HSM 的操作承擔更多責任。

AWS KMS 支援兩種自訂金鑰存放區。

- AWS CloudHSM 金鑰存放區是由 AWS CloudHSM 叢集支援的 AWS KMS 自訂金鑰存放區。在您的 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰時，AWS KMS 會在相關聯的 AWS CloudHSM 叢集中產生一個 256 位元、持久、不可匯出的進階加密標準 (AES) 對稱金鑰。此金鑰材料永遠不會讓您的 AWS CloudHSM 叢集處於未加密狀態。當您在 AWS CloudHSM 金鑰存放區中使用 KMS 金鑰時，將在叢集的 HSM 中執行密碼編譯操作。AWS CloudHSM 叢集由經過 FIPS 140-2 第 3 級認證的硬體安全模組 (HSM) 支援。
- 外部金鑰存放區是由您擁有和控制的 AWS 之外的外部金鑰管理器所支援的 AWS KMS 自訂金鑰存放區。當您在外部金鑰存放區中使用 KMS 金鑰時，外部金鑰管理器將使用密碼編譯金鑰執行所有加密和解密操作。外部金鑰存放區的設計是為了支援來自不同廠商的各種外部金鑰管理器。

  AWS KMS 不會直接檢視、存取外部金鑰管理器或密碼編譯金鑰，或與之互動。當您在外部金鑰存放區中使用 KMS 金鑰進行加密或解密時，外部金鑰管理器將使用外部金鑰執行操作。您保留對密碼編譯金鑰的完全控制權，包括拒絕或停止密碼編譯操作而不與 AWS 互動的能力。但是，由於距離和額外的處理，外部金鑰存放區中的 KMS 金鑰可能會有較差的延遲和效能，並且可能與 AWS KMS 中包含金鑰材料的 KMS 金鑰具有不同的可用性特性。如需詳細資訊了解與 AWS KMS 外部金鑰存放

區功能相容的金鑰管理程式，請參閱《AWS Key Management Service常見問題》的 Which external vendors support the XKS Proxy specification?

這兩種類型的自訂金鑰存放區與標準 AWS KMS 金鑰存放區以及彼此之間有很大不同。其安全模型、責任歸屬、效能、價格與使用案例也非常不同。在選擇自訂金鑰存放區之前，請閱讀相關文件，並確認為了獲得額外控制而承擔額外的設定和維護責任是否為明智的取捨。但是，如果您的操作所依據的規則和法規需要直接控制金鑰材料，則自訂金鑰存放區可能是您的理想選擇。

不支援的功能

AWS KMS 不支援自訂金鑰存放區中的以下功能。

- 非對稱 KMS 金鑰
- 非對稱資料金鑰對
- HMAC KMS 金鑰
- 含有匯入金鑰資料的 KMS 金鑰
- 自動金鑰輪換
- 多區域金鑰

主題

- AWS CloudHSM 主要商店
- 外部金鑰存放區

AWS CloudHSM 主要商店

AWS CloudHSM 金鑰存放區是由AWS CloudHSM 叢集支援的自訂金鑰存放區。當您在自訂金鑰存放區AWS KMS key中建立時， AWS KMS 會在您擁有和管理的 AWS CloudHSM 叢集中為 KMS 金鑰產生並儲存不可擷取的金鑰材料。當您使用自訂金鑰存放區中的 KMS 金鑰時，密碼編譯操作是在叢集的 HSM 中執行。此功能結合的 AWS KMS 便利性和廣泛整合，以 AWS CloudHSM 及您的 AWS 帳戶.

AWS KMS 為建立、使用和管理自訂金鑰存放區提供完整的主控台和 API 支援。您使用自訂金鑰存放區中 KMS 金鑰的方式，與您使用任何 KMS 金鑰的方式相同。例如，您可以使用 KMS 金鑰來產生資料金鑰並加密資料。您也可以將自訂金鑰存放區中的 KMS 金鑰與支援客戶管理金鑰的 AWS 服務搭配使用。

我是否需要自訂金鑰存放區？

對於大多數使用者而言，預設 AWS KMS 金鑰存放區受到 FIPS 140-2 驗證的加密模組保護，可滿足其安全性需求。您不需要多一層維護責任或依賴額外的服務。

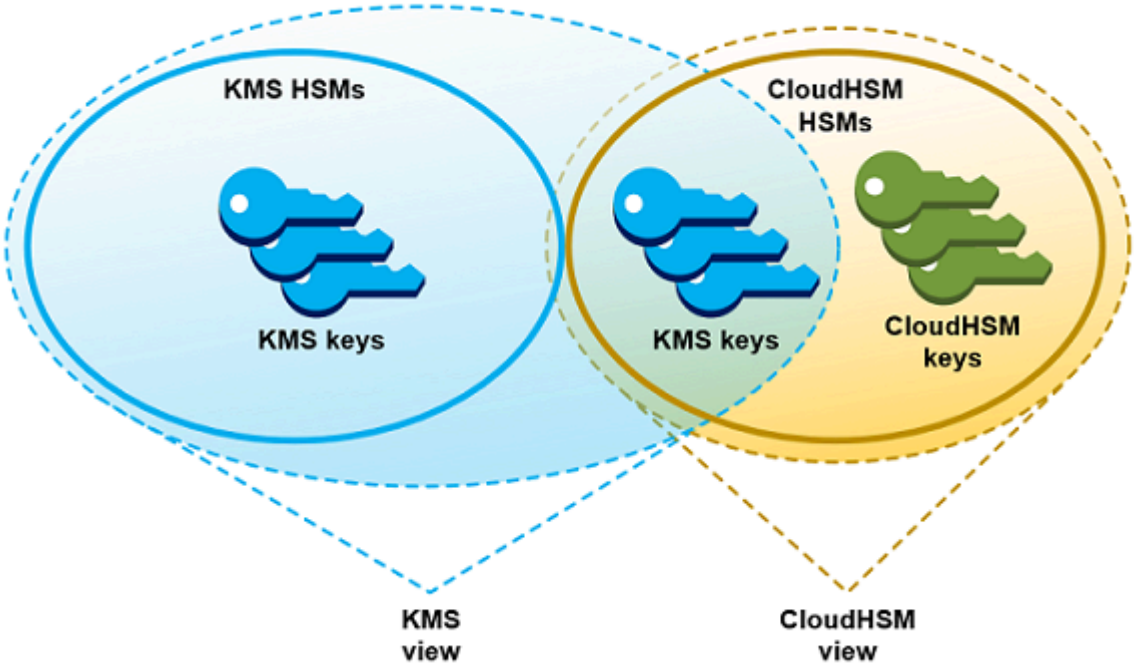
不過，如果您的組織有下列要求，您可能會考慮建立自訂金鑰存放區：

- 您在單一租用戶 HSM 或您可直接控制的 HSM 有明確需要受到保護的金鑰。
- 您需要能夠立即從中移除關鍵材料 AWS KMS。
- 您需要能夠獨立於或之外稽核金鑰的所有使 AWS KMS 用 AWS CloudTrail。

自訂金鑰存放區如何運作？

每個自訂金鑰存放區 AWS CloudHSM 都與您的 AWS 帳戶. 將自訂金鑰存放區連線至其叢集時， AWS KMS 會建立網路基礎結構以支援連線。然後，它會使用叢集中的專屬加密使用者的憑證登入叢集中的金鑰用 AWS CloudHSM 戶端。

您可以在中建立和管理自訂金鑰存放區， AWS KMS 並在中建立和管理 HSM 叢集。 AWS CloudHSM 在 AWS KMS 自訂金鑰存放區 AWS KMS keys 中建立時，您可以在中檢視和管理 KMS 金鑰 AWS KMS。但是您也可以在中檢視和管理其金鑰材料 AWS CloudHSM，就像您對叢集中的其他金鑰一樣。



您可以使用自訂金鑰存放區 AWS KMS 中產生的金鑰材料來建立對稱加密 KMS 金鑰。然後使用相同的技巧來檢視和管理您在金鑰存放區中用於 KMS 金鑰的自訂金鑰存放區中的 KMS 金 AWS KMS 鑰。您

可以使用 IAM 和金鑰政策控制存取、建立標籤和別名、啟用和停用 KMS 金鑰，以及排程金鑰刪除。您可以使用 KMS 金鑰進行密碼編譯作業，並將其與整合的 AWS 服務搭配 AWS KMS使用。

此外，您還可以完全控制 AWS CloudHSM 叢集，包括建立和刪除 HSM 以及管理備份。您可以使用用 AWS CloudHSM 戶端和支援的軟體程式庫來檢視、稽核和管理 KMS 金鑰的金鑰材料。自訂金鑰存放區中斷連線時， AWS KMS 無法存取，且使用者無法使用自訂金鑰存放區中的 KMS 金鑰進行密碼編譯作業。多一道這種控制讓自訂金鑰存放區成為有需要的組織的強大解決方案。

從何處開始？

若要建立和管理 AWS CloudHSM 金鑰存放區，請使用 AWS KMS 和的功能 AWS CloudHSM。

1. 開始於 AWS CloudHSM。建立作用中 AWS CloudHSM 叢集或選取現有的叢集。叢集必須有至少兩個作用中 HSM 在不同的可用區域。然後為 AWS KMS建立該叢集中的專用加密使用者 (CU) 帳戶。
2. 在中 AWS KMS，建立與所選 AWS CloudHSM 叢集相關聯的自訂金鑰存放區。 AWS KMS 提供完整的管理介面，可讓您建立、檢視、編輯和刪除自訂金鑰存放區。
3. 當您準備好使用自訂金鑰存放區時，請將其連接至其關聯的 AWS CloudHSM 叢集。 AWS KMS 建立支援連線所需的網路基礎結構。接著會使用專用加密使用者帳戶來登入叢集，以便能夠在叢集內產生和管理金鑰資料。
4. 現在，您可以在自訂金鑰存放區建立對稱加密 KMS 金鑰。只需要在建立 KMS 金鑰時指定自訂金鑰存放區。

如果您在任何時候受阻，您可以在對自訂金鑰存放區進行故障診斷主題中尋找說明。如果找不到問題的解答，請使用本指南每一頁底部的意見回饋連結，或將問題張貼到 AWS Key Management Service 開發論壇。

配額

AWS KMS 每個 AWS 帳戶 區域最多允許 10 個自訂金鑰存放區，包括AWS CloudHSM 金鑰存放區和外部金鑰存放區，無論其連線狀態為何。此外，還有在金鑰存放區中使用 KMS 金鑰的 AWS KMS要求配額。 AWS CloudHSM

定價

如需自訂金鑰存放區中 AWS KMS 自訂金鑰存放區和客戶管理金鑰的成本資訊，請參閱AWS Key Management Service 定價。如需 AWS CloudHSM 叢集和 HSM 成本的相關資訊，請參閱AWS CloudHSM 定價。

區域

AWS KMS 支持所有 AWS 區域 支持的 AWS CloudHSM 主要商店，亞太區域（墨爾本），中國（北京），中國（寧夏）和歐洲（西班牙）除外。 AWS KMS

不支援的功能

AWS KMS 在自訂金鑰存放區中不支援下列功能。

- 非對稱 KMS 金鑰
- 非對稱資料金鑰對
- HMAC KMS 金鑰
- 含有匯入金鑰資料的 KMS 金鑰
- 自動金鑰輪換
- 多區域金鑰

主題

- AWS CloudHSM 金鑰存放區概念
- 控制對 AWS CloudHSM 金鑰存放區的存取
- 管理 CloudHSM 自訂金鑰存放區
- 管理 CloudHSM 金鑰存放區中的 KMS 金鑰
- 對自訂金鑰存放區進行故障診斷

AWS CloudHSM 金鑰存放區概念

此主題說明 AWS CloudHSM 金鑰存放區中所使用的一些概念。

AWS CloudHSM 金鑰存放區

AWS CloudHSM 金鑰存放區是與您擁有和管理之 AWS CloudHSM 叢集相關聯的自訂金鑰存放區。AWS CloudHSM 叢集由經過 FIPS 140-2 第 3 級認證的硬體安全模組 (HSM) 支援。

在您的 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰時，AWS KMS 會在相關聯的 AWS CloudHSM 叢集中產生一個 256 位元、持久、不可匯出的進階加密標準 (AES) 對稱金鑰。金鑰材料離開您的 HSM 時一定會加密。當您使用 AWS CloudHSM 金鑰存放區中的 KMS 金鑰時，在叢集的 HSM 中執行密碼編譯操作。

AWS CloudHSM 金鑰存放區將 AWS KMS 的方便且完整的金鑰管理介面與 AWS 帳戶 中的 AWS CloudHSM 叢集提供的額外控制結合在一起。此整合功能可讓您在 AWS KMS 中建立、管理和使用

KMS 金鑰，同時保有存放其金鑰材料 (包括管理叢集、HSM 和備份) 的 HSM 的完整控制。您可以使用 AWS KMS 主控台和 API 來管理 AWS CloudHSM 金鑰存放區和其 KMS 金鑰。您也可以使用 AWS CloudHSM 主控台、API、用戶端軟體和相關聯的軟體程式庫來管理相關聯的叢集。

您可以檢視及管理 AWS CloudHSM 金鑰存放區、編輯其屬性，以及從其相關聯的 AWS CloudHSM 叢集中連接和中斷連接。如果您需要刪除 AWS CloudHSM 金鑰存放區，則必須先刪除 AWS CloudHSM 金鑰存放區中的 KMS 金鑰，方法是排程其刪除並等候直到寬限期過期。刪除 AWS CloudHSM 金鑰存放區會從 AWS KMS 移除資源，但不會影響您的 AWS CloudHSM 叢集。

AWS CloudHSM 叢集

每個 AWS CloudHSM 金鑰存放區會與一個 AWS CloudHSM 叢集相關聯。在您的 AWS CloudHSM 金鑰存放區中建立 AWS KMS key 時，AWS KMS 會在相關聯的叢集中建立金鑰材料。在您的 AWS CloudHSM 金鑰存放區中使用 KMS 金鑰時，密碼編譯操作會在關聯的叢集中執行。

每個 AWS CloudHSM 叢集僅可以與一個 AWS CloudHSM 金鑰存放區相關聯。您選擇的叢集不能與另一個 AWS CloudHSM 金鑰存放區相關聯，或與另一個 AWS CloudHSM 金鑰存放區相關聯的叢集共用備份歷史記錄。叢集必須是已初始化且作用中，而且必須位於與 AWS CloudHSM 金鑰存放區相同的 AWS 帳戶 和區域中。您可以建立新的叢集或使用現有的叢集。AWS KMS 不需要獨佔使用叢集。若要在 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰，其相關聯的叢集必須至少包含兩個作用中的 HSM。所有其他操作只需要一個 HSM。

您可在建立 AWS CloudHSM 金鑰存放區時指定 AWS CloudHSM 叢集，而且無法變更它。不過，您可以替代與原始叢集共用備份歷史記錄的任何叢集。這可讓您刪除叢集，並在必要時將它以從其中一個備份建立的叢集取代。您可保有相關聯 AWS CloudHSM 叢集的完整控制，讓您可以管理使用者和金鑰、建立和刪除 HSM，並使用和管理備份。

當您準備好使用 AWS CloudHSM 金鑰存放區時，您會將它連接至其相關聯的 AWS CloudHSM 叢集。您可以隨時連接和中斷連接您的自訂金鑰存放區。連接自訂金鑰存放區時，您可以建立和使用它的 KMS 金鑰。當裝置中斷連接時，您可以檢視和管理 AWS CloudHSM 金鑰存放區和其 KMS 金鑰。但您無法建立新的 KMS 金鑰或使用 AWS CloudHSM 金鑰存放區中的 KMS 金鑰進行密碼編譯操作。

`kmsuser` 加密使用者

若要代表您建立和管理關聯 AWS CloudHSM 叢集中的金鑰材料，AWS KMS 會使用名為 `kmsuser` 叢集中的專用 AWS CloudHSM 加密使用者 (CU)。`kmsuser` CU 是標準 CU 帳戶，會自動同步到叢集中的所有 HSM，並且儲存在叢集備份中。

在您建立 AWS CloudHSM 金鑰存放區之前，您可以使用 cloudhsm_mgmt_util 中的 createUser 命令在 AWS CloudHSM 叢集中建立 `kmsuser` CU 帳戶。然後，在建立 AWS CloudHSM 金鑰存放區時，向

AWS KMS 提供 kmsuser 帳戶密碼。連接自訂金鑰存放區時，AWS KMS 會以 kmsuser CU 的身分登入叢集並輪換其密碼。AWS KMS 會在安全存放您的 kmsuser 密碼之前進行加密。當密碼輪換時，新密碼會加密並以相同的方式存放。

只要 AWS CloudHSM 金鑰存放區已連接，AWS KMS 便會保持以 kmsuser 身分登入。您不應該將此 CU 帳戶用於其他用途。不過，您對於 kmsuser CU 帳戶會保有最終控制。在任何時候，您都可以找到 kmsuser 擁有之金鑰的金鑰控制代碼。如有需要，您可以中斷連線自訂金鑰存放區，變更 kmsuser 密碼，以 kmsuser 身分登入叢集，以及檢視和管理 kmsuser 擁有的金鑰。

如需建立您的 kmsuser CU 帳戶的相關指示，請參閱建立 kmsuser 加密使用者。

AWS CloudHSM 金鑰存放區中的 KMS 金鑰

您可以使用 AWS KMS 或 AWS KMS API，在 AWS CloudHSM 金鑰存放區中建立 AWS KMS keys。您可以使用與您在任何 KMS 金鑰上所使用的相同技術。唯一的差別是您必須識別 AWS CloudHSM 金鑰存放區，並指定金鑰材料的來源是 AWS CloudHSM 叢集。

當您在 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰時，AWS KMS 會在 AWS KMS 中建立 KMS 金鑰，並且會在其相關聯的叢集中產生一個 256 位元、持久、不可匯出的進階加密標準 (AES) 對稱金鑰材料。當您在密碼編譯操作中使用 AWS KMS 金鑰時，會使用叢集型 AES 金鑰在 AWS CloudHSM 叢集中執行此操作。雖然 AWS CloudHSM 支援不同類型的對稱和非對稱金鑰，但 AWS CloudHSM 金鑰存放區僅支援 AES 對稱加密金鑰。

您可以在 AWS KMS 主控台中檢視 AWS CloudHSM 金鑰存放區中的 KMS 金鑰，並使用主控台選項來顯示自訂金鑰存放區 ID。您也可以使用此DescribeKey作業來尋找AWS CloudHSM金鑰存放區 ID 和 AWS CloudHSM叢集 ID。

AWS CloudHSM 金鑰存放區中的 KMS 金鑰運作方式就像 AWS KMS 中的任何 KMS 金鑰。獲授權使用者需要相同權限才能使用和管理 KMS 金鑰。您可以使用相同的主控台程序和 API 操作，以檢視和管理 AWS CloudHSM 金鑰存放區中的 KMS 金鑰。這些包含啟用和停用 KMS 金鑰、建立和使用標籤和別名，以及設定和變更 IAM 和金鑰政策。您可以將 AWS CloudHSM 金鑰存放區中的 KMS 金鑰用於密碼編譯操作，並將其與支援使用受管金鑰的整合 AWS 服務搭配使用。但是，您無法啟用自動金鑰輪換或將金鑰材料匯入 AWS CloudHSM 金鑰存放區中的 KMS 金鑰。

您也可以使用相同的程序，在 AWS CloudHSM 金鑰存放區中排程刪除 KMS 金鑰。在等待期間過期之後，AWS KMS 會從 KMS 刪除 KMS 金鑰。然後，它會盡可能從相關聯的 AWS CloudHSM 叢集刪除 KMS 金鑰的金鑰材料。不過，您可能需要手動從叢集及其備份刪除遺棄的金鑰材料。

控制對 AWS CloudHSM 金鑰存放區的存取

您可以使用 IAM 政策來控制對 AWS CloudHSM 金鑰存放區和 AWS CloudHSM 叢集的存取。您可以使用金鑰政策、IAM 政策，並授予對 AWS CloudHSM 金鑰存放區中 AWS KMS keys 的存取。建議您只提供使用者、群組和角色可能執行的任務所需的許可給他們。

主題

- 授權 AWS CloudHSM 金鑰存放區管理員和使用者
- 授權 AWS KMS 來管理 AWS CloudHSM 和 Amazon EC2 資源

授權 AWS CloudHSM 金鑰存放區管理員和使用者

在設計 AWS CloudHSM 金鑰存放區時，請確定使用和管理它的主體只具有他們所需的許可。以下清單說明 AWS CloudHSM 金鑰存放區管理員和使用者所需的最低許可。

- 建立和管理 AWS CloudHSM 金鑰存放區的主體需要以下許可，才能使用 AWS CloudHSM 金鑰存放區 API 操作。
  - `cloudhsm:DescribeClusters`
  - `kms:CreateCustomKeyStore`
  - `kms:ConnectCustomKeyStore`
  - `kms:DeleteCustomKeyStore`
  - `kms:DescribeCustomKeyStores`
  - `kms:DisconnectCustomKeyStore`
  - `kms:UpdateCustomKeyStore`
  - `iam:CreateServiceLinkedRole`
- 建立和管理與 AWS CloudHSM 金鑰存放區相關聯的 AWS CloudHSM 叢集的主體，需要許可來建立和初始化 AWS CloudHSM 叢集。這包括要有許可來建立或使用 Amazon Virtual Private Cloud (VPC)、建立子網路，以及建立 Amazon EC2 執行個體。他們可能還需要建立和刪除 HSM，以及管理備份。如需必要許可的清單，請參閱《AWS CloudHSM 使用指南》中 AWS CloudHSM 的身分與存取管理。
- 在 AWS CloudHSM 金鑰存放區建立和管理 AWS KMS keys 的主體，與在 AWS KMS 中建立和管理任何 KMS 金鑰的主體，需要相同的許可。AWS CloudHSM 金鑰存放區中 KMS 金鑰的預設金鑰政策與 AWS KMS 中 KMS 金鑰的預設金鑰政策完全相同。屬性型存取控制 (ABAC) 使用標籤和別名來控制對 KMS 金鑰的存取，對 AWS CloudHSM 金鑰存放區中的 KMS 金鑰也有效。

- 將 AWS CloudHSM 金鑰存放區中的 KMS 金鑰用於密碼編譯操作的委託人，需要許可才能使用 KMS 金鑰執行密碼編譯操作，例如 kms:Decrypt。您可以在金鑰政策或 IAM 政策中提供這些許可。但是，他們使用 AWS CloudHSM 金鑰存放區中的 KMS 金鑰並不需要任何額外的許可。

授權 AWS KMS 來管理 AWS CloudHSM 和 Amazon EC2 資源

為了支援您的 AWS CloudHSM 金鑰存放區，AWS KMS 需要許可來取得 AWS CloudHSM 叢集的相關資訊。還需要許可來建立網路基礎設施，以便將 AWS CloudHSM 金鑰存放區連接到其 AWS CloudHSM 叢集。若要取得這些權限 AWSServiceRoleForKeyManagementServiceCustomKeyStores，AWS KMS請在您的AWS 帳戶. 建立 AWS CloudHSM 金鑰存放區的使用者必須具有 `iam:CreateServiceLinkedRole` 許可，讓他們能夠建立服務連結角色。

主題

- 關於 AWS KMS 服務連結角色
- 建立服務連結角色
- 編輯服務連結角色描述
- 刪除服務連結角色

關於 AWS KMS 服務連結角色

服務連結角色是 IAM 角色，提供許可給一個 AWS 服務代表您呼叫其他 AWS 服務。目的是讓您輕鬆使用多個整合的 AWS 服務的功能，而不需要建立和維護複雜的 IAM 政策。如需詳細資訊，請參閱 使用 AWS KMS 的服務連結角色。

針對AWS CloudHSM金鑰存放區，AWS KMS建立具有AWSKeyManagementServiceCustomKeyStoresServiceRolePolicy策略的AWSServiceRoleForKeyManagementServiceCustomKeyStores服務連結角色。此政策將下列許可授予此角色：

- CloudHSM: 說明 * — 偵測附加至自訂金鑰存放區之AWS CloudHSM叢集中的變更。
- ec2: CreateSecurityGroup — 當您連接AWS CloudHSM金鑰存放區以建立安全群組，以啟用AWS KMS與AWS CloudHSM叢集之間的網路流量流動時使用。
- ec2: AuthorizeSecurityGroupIngress — 當您連接AWS CloudHSM金鑰存放區時使用，以允許從包含 AWS CloudHSM叢集的 VPC AWS KMS 進行網路存取。

- ec2: CreateNetworkInterface — 當您連接AWS CloudHSM密鑰存儲區以創建用於AWS KMS與AWS CloudHSM集群之間通信的網絡界面時使用。
- ec2: RevokeSecurityGroupEgress — 當您連接AWS CloudHSM密鑰存儲區以從創建的安全組中刪除所有出站規則時使AWS KMS用。
- ec2: DeleteSecurityGroup — 當您斷開AWS CloudHSM密鑰存儲區以刪除連接密AWS CloudHSM鑰存儲時創建的安全組時使用。
- ec2: DescribeSecurityGroups — 用於監視在包含AWS CloudHSM叢集的 VPC 中AWS KMS建立的安全群組中的變更，AWS KMS以便在發生故障時提供明確的錯誤訊息。
- ec2: DescribeVpcs — 用於監控包含AWS CloudHSM叢集的 VPC 中的變更，AWS KMS以便在發生故障時提供明確的錯誤訊息。
- ec2: DescribeNetworkAcls — 用於監控包含AWS CloudHSM叢集的 VPC 網路 ACL 中的變更，AWS KMS以便在發生故障時提供明確的錯誤訊息。
- ec2: DescribeNetworkInterfaces — 用於監視在包含AWS CloudHSM叢集的 VPC 中AWS KMS建立的網路介面中的變更，AWS KMS以便在發生故障時提供明確的錯誤訊息。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "cloudhsm:Describe*",
        "ec2:CreateNetworkInterface",
        "ec2:AuthorizeSecurityGroupIngress",
        "ec2:CreateSecurityGroup",
        "ec2:DescribeSecurityGroups",
        "ec2:RevokeSecurityGroupEgress",
        "ec2:DeleteSecurityGroup",
        "ec2:DescribeVpcs",
        "ec2:DescribeNetworkAcls",
        "ec2:DescribeNetworkInterfaces"
      ],
      "Resource": "*"
    }
  ]
}
```

由於AWSServiceRoleForKeyManagementServiceCustomKeyStores服務連結角色只信任`cks.kms.amazonaws.com`，因此只AWS KMS能擔任此服務連結角色。此角色僅限用於讓 AWS KMS 檢視 AWS CloudHSM 叢集和將 AWS CloudHSM 金鑰存放區連接到其關聯的 AWS CloudHSM 叢集所需的操作。並不授予 AWS KMS 任何額外的許可。例如，AWS KMS 沒有許可來建立、管理或刪除 AWS CloudHSM 叢集、HSM 或備份。

區域

與AWS CloudHSM金鑰存放區功能一樣，AWSServiceRoleForKeyManagementServiceCustomKeyStores角色在所有可用的AWS 區域位置AWS CloudHSM都AWS KMS受到支援。如需每個服務支援的 AWS 區域 清單，請參閱《Amazon Web Services 一般參考》的AWS Key Management Service 端點及配額與AWS CloudHSM端點及配額。

如需 AWS 服務如何使用服務連結角色的詳細資訊，請參閱《IAM 使用者指南》中的使用服務連結角色。

建立服務連結角色

AWS KMS當您建立AWS CloudHSM金鑰存放區AWS 帳戶時，如果角色尚未存在，則會自動在您的中建立AWSServiceRoleForKeyManagementServiceCustomKeyStores服務連結角色。您無法直接建立或重新建立此服務連結角色。

編輯服務連結角色描述

您無法編輯 AWSServiceRoleForKeyManagementServiceCustomKeyStores 服務連結角色中的角色名稱或政策陳述式，但可以編輯角色描述。如需相關說明，請參閱《IAM 使用者指南》中的編輯服務連結角色。

刪除服務連結角色

AWS KMS即使您已刪除所有AWS CloudHSM金鑰存放區，AWS 帳戶也不會刪除您的AWSServiceRoleForKeyManagementServiceCustomKeyStores服務連結角色。雖然目前沒有刪除AWSServiceRoleForKeyManagementServiceCustomKeyStores服務連結角色的程序，但除非您擁有作用中的AWS CloudHSM金鑰存放區，否則不AWS KMS會擔任此角色或使用其權限。

管理 CloudHSM 自訂金鑰存放區

您可以使用 AWS Management Console 和 AWS KMS API 來管理自訂金鑰存放區。例如，您可以檢視自訂金鑰存放區、編輯其屬性、與其相關聯的 AWS CloudHSM 叢集連接和中斷連接，以及刪除自訂金鑰存放區。

主題

建立 AWS CloudHSM 金鑰存放區

您可以在帳戶中建立一個或多個 AWS CloudHSM 金鑰存放區。每個 AWS CloudHSM 金鑰存放區會與相同 AWS 帳戶 和區域中的一個 AWS CloudHSM 叢集相關聯。在您建立 AWS CloudHSM 金鑰存放區之前，您需要備妥先決條件。然後，在您使用 AWS CloudHSM 金鑰存放區之前，必須將其連接到其 AWS CloudHSM 叢集。

> **Note**
>
> 如果您嘗試建立一個 AWS CloudHSM 金鑰存放區，並將其所有屬性值設定為與現有中斷連接的 AWS CloudHSM 存放區相同，AWS KMS 不會建立新的 AWS CloudHSM 金鑰存放區，並且其不會擲回例外狀況或顯示錯誤。相反地，AWS KMS 會將這個重複項識別為重試的可能結果，並傳回現有 AWS CloudHSM 金鑰存放區的 ID。

> **Tip**
>
> 您不需要立即連接到 AWS CloudHSM 金鑰存放區。在準備使用之前可以維持在中斷連線狀態。不過，若要確認是否已正確設定，您可能需要將其連接、檢視其連接狀態，然後中斷連接。

主題

備妥先決條件

每個 AWS CloudHSM 金鑰存放區都由 AWS CloudHSM 叢集所支援。若要建立 AWS CloudHSM 金鑰存放區，您必須指定尚未與另一個金鑰存放區相關聯的作用中 AWS CloudHSM 叢集。您還需要在叢集的 HSM 中建立專用加密使用者 (CU)，供 AWS KMS 用來代替您建立和管理金鑰。

在建立 AWS CloudHSM 金鑰存放區之前，請執行下列操作：

選取 AWS CloudHSM 叢集

每個 AWS CloudHSM 金鑰存放區僅與一個 AWS CloudHSM 叢集相關聯。當您在 AWS CloudHSM 金鑰存放區建立 AWS KMS keys 時，AWS KMS 會在 AWS KMS 中建立 KMS 金鑰中繼資料，例如 ID 和 Amazon Resource Name (ARN)。接著，它會在相關聯叢集的 HSM 中建立金鑰材料。您可以建立新的 AWS CloudHSM 叢集或使用現有的叢集。AWS KMS 不需要叢集的獨佔存取權。

您選取的 AWS CloudHSM 叢集會與 AWS CloudHSM 金鑰存放區永久相關聯。建立 AWS CloudHSM 金鑰存放區之後，您可以對相關聯叢集變更叢集 ID，但您指定的叢集必須與原始叢集共用備份歷史記錄。若要使用不相關的叢集，您需要建立新的 AWS CloudHSM 金鑰存放區。

您選取的 AWS CloudHSM 叢集必須有下列特性：

- 叢集必須為作用中。

  您必須建立叢集、初始化叢集、為您的平台上安裝 AWS CloudHSM 用戶端軟體，然後啟動叢集。如需詳細說明，請參閱《AWS CloudHSM 使用者指南》中的 AWS CloudHSM 入門。
- 叢集必須位於與 AWS CloudHSM 金鑰存放區相同的帳戶和區域中。您不能將某個區域的 AWS CloudHSM 金鑰存放區與不同區域的叢集相關聯。若要在多個區域建立金鑰基礎設施，則必須在每個區域建立 AWS CloudHSM 金鑰存放區和叢集。
- 叢集不能與相同帳戶和區域中的其他自訂金鑰存放區關聯。帳戶和區域中的每個 AWS CloudHSM 金鑰存放區都必須與不同的 AWS CloudHSM 叢集關聯。您不能指定已經與自訂金鑰存放區相關聯的叢集，或與相關聯叢集共用備份歷史記錄的叢集。共用備份歷史記錄的叢集具有相同的叢集憑證。若要檢視叢集的叢集憑證，請使用AWS CloudHSM主控台或DescribeClusters作業。

  如果您將 AWS CloudHSM 叢集備份至不同區域，則此叢集會被視為不同的叢集，您可以將備份與該區域中的自訂金鑰存放區關聯。但是，兩個自訂金鑰存放區中的 KMS 金鑰不可互通，即使它們具有相同的備份金鑰。AWS KMS 會將中繼資料繫結至密文，以便只能透過加密它的 KMS 金鑰對其進行解密。

- 必須在區域中至少兩個可用區域中，為叢集設定私有子網路。由於並非所有可用區域都支援 AWS CloudHSM，建議您在區域中的所有可用區域建立私有子網路。您不能對現有叢集重新設定子網路，但可以從備份建立叢集，並於叢集組態中使用不同的子網路。

  > **Important**
  >
  > 建立 AWS CloudHSM 金鑰存放區後，請勿刪除為其 AWS CloudHSM 叢集設定的任何私有子網路。如果 AWS KMS 在叢集組態中找不到所有子網路，嘗試連線至自訂金鑰存放區會失敗，並顯示 SUBNET_NOT_FOUND 連線錯誤狀態。如需詳細資訊，請參閱 如何修正連線失敗。

- 叢集的安全群組 (cloudhsm-cluster-*<cluster-id>*-sg) 必須包含在連接埠 2223-2225 上允許 TCP 流量的傳入規則和傳出規則。傳入規則的 來源和傳出規則的目的地必須符合安全群組 ID。當您建立叢集時依預設會設定這些規則。請勿刪除或變更。
- 叢集必須包含至少兩個作用中 HSM 在不同的可用區域。若要驗證 HSM 的數目，請使用主AWS CloudHSM控台或DescribeClusters作業。如有需要，您可以新增 HSM。

尋找信任起點憑證

當您建立自訂金鑰存放區時，您必須將 AWS CloudHSM 叢集的信任錨憑證上傳至 AWS KMS。AWS KMS 需要信任錨憑證，才能將 AWS CloudHSM 金鑰存放區連接到其相關聯的 AWS CloudHSM 叢集。

每個作用中 AWS CloudHSM 叢集都有信任起點憑證。當您初始化叢集時，您需要產生此憑證，將它儲存在 customerCA.crt 檔案中，然後複製到連接至叢集的主機。

建立 AWS KMS 的 **kmsuser** 加密使用者

為了管理您的 AWS CloudHSM 金鑰存放區，AWS KMS 會登入所選叢集的kmsuser加密使用者 (CU) 帳戶。建立 AWS CloudHSM 金鑰存放區之前，您必須建立 kmsuser CU。然後，在建立 AWS CloudHSM 金鑰存放區時，向 AWS KMS 提供 kmsuser 帳戶密碼。將 AWS CloudHSM 金鑰存放區連接到其相關聯的 AWS CloudHSM 叢集時，AWS KMS 會以 kmsuser 身分登入並輪換 kmsuser 密碼。

> **Important**
>
> 當您建立 kmsuser CU 時，請勿指定 2FA 選項。否則，AWS KMS 無法登入，且 AWS CloudHSM 金鑰存放區無法連接到此 AWS CloudHSM 叢集。一旦指定 2FA 就無法復原。您必須刪除 CU 再重新建立。

若要建立 kmsuser 角色，請使用下列程序。

1. 如《AWS CloudHSM 使用者指南》中 CloudHSM Management Utility (CMU) 主題所述，啟動 cloudhsm_mgmt_util。
2. 在 loudhsm_mgmt_util 中使用 createUser 命令建立名為 kmsuser 的 CU。密碼必須包含 7 到 32 個英數字元。區分大小寫，且不能包含任何特殊字元。

   例如，以下範例命令會建立密碼為 kmsPswd 的 kmsuser CU。

   ```
   aws-cloudhsm> createUser CU kmsuser kmsPswd
   ```

建立 AWS CloudHSM 金鑰存放區 (主控台)

在 AWS Management Console 中建立 AWS CloudHSM 金鑰存放區時，您可以新增和建立先決條件，作為工作流程的一部分。不過，如果事先備妥，則程序會更快。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格依次選擇自訂金鑰存放區、AWS CloudHSM 金鑰存放區。
4. 選擇 Create a key store (建立金鑰存放區)。
5. 輸入自訂金鑰存放區的易記名稱。該名稱在帳戶的所有自訂金鑰存放區中必須是唯一的。

   > **Important**
   >
   > 請勿在此欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，此欄位可能會以純文字顯示。

6. 為 AWS CloudHSM 金鑰存放區選取 AWS CloudHSM叢集。或者，若要建立新的 AWS CloudHSM 叢集，請選擇 Create an AWS CloudHSM cluster (建立叢集) 連結。

   選單會顯示您的帳戶和區域中尚未與 AWS CloudHSM 金鑰存放區相關聯的 AWS CloudHSM 叢集。叢集必須滿足要求，才能與自訂金鑰存放區建立關聯。
7. 選擇 Choose file (選擇檔案)，然後針對所選的 AWS CloudHSM 叢集上傳信任錨憑證。這是 customerCA.crt 檔案，您在初始化叢集時所建立。
8. 輸入您在所選叢集中建立的 kmsuser加密使用者 (CU) 的密碼。

9. 選擇建立。

當程序成功時，新的 AWS CloudHSM 金鑰存放區會出現在帳戶和區域的 AWS CloudHSM 金鑰存放區清單中。如果不成功，則會出現錯誤訊息來描述問題，並提供如何修正的說明。如果您需要更多協助，請參閱對自訂金鑰存放區進行故障診斷。

如果您嘗試建立一個 AWS CloudHSM 金鑰存放區，並將其所有屬性值設定為與現有中斷連接的 AWS CloudHSM 存放區相同，AWS KMS 不會建立新的 AWS CloudHSM 金鑰存放區，並且其不會擲回例外狀況或顯示錯誤。相反地，AWS KMS 會將這個重複項識別為重試的可能結果，並傳回現有 AWS CloudHSM 金鑰存放區的 ID。

下一步：不會自動連接新的 AWS CloudHSM 金鑰存放區。在 AWS CloudHSM 金鑰存放區中建立 AWS KMS keys 之前，您必須將自訂金鑰存放區連接到其相關聯的 AWS CloudHSM 叢集。

建立 AWS CloudHSM 金鑰存放區 (API)

您可以使用此CreateCustomKeyStore作業建立與帳戶和區域中AWS CloudHSM叢集相關聯的新AWS CloudHSM金鑰存放區。以下範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

`CreateCustomKeyStore` 操作需要以下參數值。

- CustomKeyStoreName — 自訂金鑰存放區的易記名稱，在帳戶中是唯一的。

> **Important**
>
> 請勿在此欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，此欄位可能會以純文字顯示。

- CloudHsmClusterId — 符合AWS CloudHSM金鑰存放區需求之AWS CloudHSM叢集的叢集 ID。
- KeyStorePassword — 指定叢集中 `kmsuser` CU 帳戶的密碼。
- TrustAnchorCertificate — 初始化叢集時建立的`customerCA.crt`檔案內容。

以下範例使用虛構的叢集 ID。執行命令之前，請換成有效的叢集 ID。

```
$ aws kms create-custom-key-store
        --custom-key-store-name ExampleCloudHSMKeyStore \
        --cloud-hsm-cluster-id cluster-1a23b4cdefg \
        --key-store-password kmsPswd \
```

```
        --trust-anchor-certificate <certificate-goes-here>
```

如果您使用的是 AWS CLI，您可以指定信任起點憑證檔案，而不是其內容。在下列範例中，customerCA.crt 檔案位於根目錄。

```
$ aws kms create-custom-key-store
        --custom-key-store-name ExampleCloudHSMKeyStore \
        --cloud-hsm-cluster-id cluster-1a23b4cdefg \
        --key-store-password kmsPswd \
        --trust-anchor-certificate file://customerCA.crt
```

當操作成功時，CreateCustomKeyStore 會傳回自訂金鑰存放區 ID，如下回應範例所示。

```
{
    "CustomKeyStoreId": cks-1234567890abcdef0
}
```

如果操作失敗，請修正例外狀況所指出的錯誤，然後重試。如需其他說明，請參閱對自訂金鑰存放區進行故障診斷。

如果您嘗試建立一個 AWS CloudHSM 金鑰存放區，並將其所有屬性值設定為與現有中斷連接的 AWS CloudHSM 存放區相同，AWS KMS 不會建立新的 AWS CloudHSM 金鑰存放區，並且其不會擲回例外狀況或顯示錯誤。相反地，AWS KMS 會將這個重複項識別為重試的可能結果，並傳回現有 AWS CloudHSM 金鑰存放區的 ID。

下一步：若要使用 AWS CloudHSM 金鑰存放區，請將其連接到其 AWS CloudHSM 叢集。

檢視 AWS CloudHSM 金鑰存放區

您可以使用AWS KMS主控台或DescribeCustomKeyStores操作來檢視每個帳戶和區域中的AWS CloudHSM金鑰存放區。

另請參閱:

- 檢視外部金鑰存放區
- 檢視 AWS CloudHSM 金鑰存放區中的 KMS 金鑰
- 使用 AWS CloudTrail 記錄 AWS KMS API 呼叫

主題

- 檢視 AWS CloudHSM 金鑰存放區 (主控台)
- 檢視 AWS CloudHSM 金鑰存放區 (API)

檢視 AWS CloudHSM 金鑰存放區 (主控台)

在 AWS Management Console 中檢視 AWS CloudHSM 金鑰存放區時，您可以查看下列項目：

- 自訂金鑰存放區名稱和 ID
- 相關聯 AWS CloudHSM 叢集的 ID
- 叢集中 HSM 的數量
- 目前的連接狀態

連接狀態 (Status) 的 Disconnected (已中斷連接) 值指示自訂金鑰存放區是新的並且從未連接，或其是特意從其 AWS CloudHSM 叢集中斷連接。不過，如果您嘗試在連線的自訂金鑰存放區中使用 KMS 金鑰但失敗，可能表示自訂金鑰存放區或其 AWS CloudHSM 叢集中發生問題。如需協助，請參閱 如何修正失效的 KMS 金鑰。

若要檢視指定帳戶和區域中的 AWS CloudHSM 金鑰存放區，請使用下列程序。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格依次選擇自訂金鑰存放區、AWS CloudHSM 金鑰存放區。

若要自訂顯示，請按一下出現在 Create key store (建立金鑰存放區) 按鈕下的齒輪圖示。

檢視 AWS CloudHSM 金鑰存放區 (API)

若要檢視AWS CloudHSM金鑰存放區，請使用DescribeCustomKeyStores作業。在預設情況下，此操作會傳回帳戶和區域中的所有自訂金鑰存放區。但是，您可以使用 `CustomKeyStoreId` 或 `CustomKeyStoreName` 參數 (但不能同時使用) 來限制對特定自訂金鑰存放區的輸出。若為 AWS CloudHSM 金鑰存放區，則輸出包含自訂金鑰存放區 ID 和名稱、自訂金鑰存放區類型、相關聯的 AWS CloudHSM 叢集 ID 以及連接狀態。如果連接狀態指出錯誤，則輸出也會包含描述錯誤原因的錯誤碼。

本節中的範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

例如，以下命令會傳回帳戶和區域中的所有自訂金鑰存放區。您可以使用 Limit 和 Marker 參數來切換輸出中的自訂金鑰存放區頁面。

```
$ aws kms describe-custom-key-stores
```

以下範例命令使用 CustomKeyStoreName 參數來取得僅具有 ExampleCloudHSMKeyStore 易記名稱的自訂金鑰存放區。您可以在每個命令中使用 CustomKeyStoreName 或 CustomKeyStoreId 參數 (但不可同時使用)。

以下範例輸出表示已連接到其 AWS CloudHSM 叢集的 AWS CloudHSM 金鑰存放區。

> **Note**
>
> 此 CustomKeyStoreType 欄位已新增至 DescribeCustomKeyStores 回應，以區分 AWS CloudHSM 金鑰存放區與外部金鑰存放區。

```
$ aws kms describe-custom-key-stores --custom-key-store-name ExampleCloudHSMKeyStore
{
   "CustomKeyStores": [
      {
         "CloudHsmClusterId": "cluster-1a23b4cdefg",
         "ConnectionState": "CONNECTED",
         "CreationDate": "1.499288695918E9",
         "CustomKeyStoreId": "cks-1234567890abcdef0",
         "CustomKeyStoreName": "ExampleCloudHSMKeyStore",
         "CustomKeyStoreType": "AWS_CLOUDHSM",
         "TrustAnchorCertificate": "<certificate appears here>"
      }
   ]
}
```

Disconnected 的 ConnectionState 指出自訂金鑰存放區是新的並且從未連線，或它是特意與其 AWS CloudHSM 叢集中斷連接。不過，如果嘗試在已連接的 AWS CloudHSM 金鑰存放區中使用 KMS 金鑰失敗，可能表示 AWS CloudHSM 金鑰存放區或其 AWS CloudHSM 叢集中發生問題。如需協助，請參閱 如何修正失效的 KMS 金鑰。

如果自訂金鑰存放區的 ConnectionState 為 FAILED，則 DescribeCustomKeyStores 回應會包含一個 ConnectionErrorCode 元素，解釋錯誤的原因。

例如，在以下輸出中，INVALID_CREDENTIALS 值指出自訂金鑰存放區連接失敗，因為 kmsuser 密碼無效。如需此錯誤和其他連接錯誤失敗的協助，請參閱對自訂金鑰存放區進行故障診斷。

```
$ aws kms describe-custom-key-stores --custom-key-store-id cks-1234567890abcdef0
{
   "CustomKeyStores": [
      {
         "CloudHsmClusterId": "cluster-1a23b4cdefg",
         "ConnectionErrorCode": "INVALID_CREDENTIALS",
         "ConnectionState": "FAILED",
         "CustomKeyStoreId": "cks-1234567890abcdef0",
         "CustomKeyStoreName": "ExampleCloudHSMKeyStore",
         "CustomKeyStoreType": "AWS_CLOUDHSM",
         "CreationDate": "1.499288695918E9",
         "TrustAnchorCertificate": "<certificate appears here>"
      }
   ]
}
```

編輯 AWS CloudHSM 金鑰存放區設定

您可以變更現有 AWS CloudHSM 金鑰存放區的設定。必須將自訂金鑰存放區從 AWS CloudHSM 叢集中斷連接。

若要編輯 AWS CloudHSM 金鑰存放區設定：

1. 將自訂金鑰存放區從其 AWS CloudHSM 叢集中斷連線。當自訂金鑰存放區中斷連線時，您無法在自訂金鑰存放區中建立 AWS KMS keys (KMS 金鑰)，並且無法使用其包含的 KMS 金鑰進行密碼編譯操作。
2. 編輯一個或多個 AWS CloudHSM 金鑰存放區設定。
3. 將自訂金鑰存放區重新連接至其 AWS CloudHSM 叢集。

您可以編輯自訂金鑰存放區中的以下設定：

自訂金鑰存放區的易記名稱。

輸入新的易記名稱。該新名稱在 AWS 帳戶 的所有自訂金鑰存放區中必須唯一。

> **Important**
>
> 請勿在此欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，此欄位可能會以純文字顯示。

相關聯 AWS CloudHSM 叢集的叢集 ID。

編輯此值，以使用相關的 AWS CloudHSM 叢集來替代原始叢集。您可以使用此功能在 AWS CloudHSM 叢集損毀或遭到刪除時修復自訂金鑰存放區。

指定與原始叢集共用備份歷史記錄的 AWS CloudHSM 叢集，並且實現與自訂金鑰存放區建立關聯的要求，包含在不同的可用區域有兩個作用中 HSM。共用備份歷史記錄的叢集具有相同的叢集憑證。若要檢視叢集的叢集憑證，請使用DescribeClusters作業。您無法使用編輯功能，將自訂金鑰存放區與不相關的 AWS CloudHSM 叢集產生關聯。

`kmsuser` 加密使用者 (CU) 的目前密碼。

告知 AWS KMS AWS CloudHSM 叢集中 `kmsuser` CU 的目前密碼。此動作不會變更 AWS CloudHSM 叢集中 `kmsuser` CU 的密碼。

如果您變更 AWS CloudHSM 叢集中 `kmsuser` CU 的密碼，請使用此功能來告知 AWS KMS 相關的新 `kmsuser` 密碼。否則，AWS KMS 會無法登入叢集，並且將自訂金鑰存放區連接到叢集的所有嘗試都會失敗。

主題

- 編輯 AWS CloudHSM 金鑰存放區 (主控台)
- 編輯 AWS CloudHSM 金鑰存放區 (API)

編輯 AWS CloudHSM 金鑰存放區 (主控台)

編輯 AWS CloudHSM 金鑰存放區時，您可以變更任何可設定的值。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格依次選擇自訂金鑰存放區、AWS CloudHSM 金鑰存放區。

4. 選擇您想要編輯的 AWS CloudHSM 金鑰存放區的資料列。

   如 Connection state (連接狀態) 欄的值不是 DISCONNECTED (已中斷連接)，則在編輯之前，您必須先中斷連接自訂金鑰存放區。(從 Key store actions (金鑰存放區動作) 選單中，選擇 Disconnect (中斷連接)。)

   當 AWS CloudHSM 金鑰存放區中斷連接時，您可以管理 AWS CloudHSM 金鑰存放區和其 KMS 金鑰，但無法在 AWS CloudHSM 金鑰存放區中建立或使用 KMS 金鑰。

5. 從 Key store actions (金鑰存放區動作) 選單中，選擇 Edit (編輯)。
6. 執行下列其中一或多個動作。

   - 輸入自訂金鑰存放區的易記名稱。
   - 輸入相關 AWS CloudHSM 叢集的叢集 ID。
   - 輸入關聯的 AWS CloudHSM 叢集中 `kmsuser` 加密使用者目前的密碼。

7. 選擇儲存。

   當程序成功時，會出現訊息描述您編輯的設定。當操作失敗時，會出現錯誤訊息，其中描述問題並提供如何修正的協助。如果您需要更多協助，請參閱對自訂金鑰存放區進行故障診斷。

8. 重新連接自訂金鑰存放區。

   若要使用 AWS CloudHSM 金鑰存放區，您必須在編輯之後，將它重新連接。您可以將 AWS CloudHSM 金鑰存放區保持中斷連接。中斷連接時，您無法在 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰，並且無法在密碼編譯操作中使用 AWS CloudHSM 金鑰存放區中的 KMS 金鑰。

編輯 AWS CloudHSM 金鑰存放區 (API)

若要變更AWS CloudHSM金鑰存放區的屬性，請使用此UpdateCustomKeyStore作業。您可以在相同命令中變更自訂金鑰存放區的多個屬性。如果操作成功，則 AWS KMS 傳回 HTTP 200 回應和不帶屬性的 JSON 物件。若要驗證變更是否有效，請使用DescribeCustomKeyStores作業。

本節中的範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

首先使用中斷DisconnectCustomKeyStore自訂金鑰存放區與其AWS CloudHSM叢集的連線。將範例自訂金鑰存放區 ID cks-1234567890abcdef0 以實際 ID 取代。

```
$ aws kms disconnect-custom-key-store --custom-key-store-id cks-1234567890abcdef0
```

第一個範例使UpdateCustomKeyStore用將AWS CloudHSM金鑰存放區的易記名稱變更為DevelopmentKeys。此命令使用 CustomKeyStoreId 參數來識別 AWS CloudHSM 金鑰存放區和 CustomKeyStoreName，以指定自訂金鑰存放區的新名稱。

```
$ aws kms update-custom-key-store --custom-key-store-id cks-1234567890abcdef0 --new-custom-key-store-name DevelopmentKeys
```

以下範例會將與 AWS CloudHSM 金鑰存放區相關聯的叢集變更為相同叢集的另一個備份。此命令使用 CustomKeyStoreId 參數來識別 AWS CloudHSM 金鑰存放區和 CloudHsmClusterId 參數，以指定新叢集 ID。

```
$ aws kms update-custom-key-store --custom-key-store-id cks-1234567890abcdef0 --cloud-hsm-cluster-id cluster-1a23b4cdefg
```

以下範例會告知 AWS KMS 目前的 kmsuser 密碼為 ExamplePassword。此命令使用 CustomKeyStoreId 參數來識別 AWS CloudHSM 金鑰存放區和 KeyStorePassword 參數，以指定目前的密碼。

```
$ aws kms update-custom-key-store --custom-key-store-id cks-1234567890abcdef0 --key-store-password ExamplePassword
```

最後的命令會將 AWS CloudHSM 金鑰存放區重新連接至其 AWS CloudHSM 叢集。您可以將自訂金鑰存放區保持在中斷連線狀態，但必須先將其連線，才能建立新的 KMS 金鑰或將現有的 KMS 金鑰用於密碼編譯操作。將範例自訂金鑰存放區 ID 以實際 ID 取代。

```
$ aws kms connect-custom-key-store --custom-key-store-id cks-1234567890abcdef0
```

連接和中斷連接 AWS CloudHSM 金鑰存放區

新 AWS CloudHSM 金鑰存放區未連接。在 AWS CloudHSM 金鑰存放區中建立和使用 AWS KMS keys 之前，您需要將其連接到其關聯的 AWS CloudHSM 叢集。您可以隨時連接和中斷連接您的 AWS CloudHSM 金鑰存放區，並且檢視其連接狀態。

您不需要連接您的 AWS CloudHSM 金鑰存放區。您可以將 AWS CloudHSM 金鑰存放區無限期保留在中斷連接狀態，並只在您需要使用它時連接它。不過，您可能希望定期測試連接，以驗證設定正確並且可連接。

> **Note**
>
> 僅當金鑰存放區從未連接或明確中斷連接時，AWS CloudHSM 金鑰存放區才會有 DISCONNECTED 連接狀態。如果您的 AWS CloudHSM 金鑰存放區狀態為 CONNECTED，但您在使用時遇到問題，請確保其關聯的 AWS CloudHSM 叢集處於作用中狀態，且至少包含一個作用中 HSM。如需連線失敗的協助，請參閱 the section called “對自訂金鑰存放區進行故障診斷”。

主題

連接 AWS CloudHSM 金鑰存放區

當您連接 AWS CloudHSM 金鑰存放區時，AWS KMS 會尋找相關聯的 AWS CloudHSM 叢集、進行連接、以 kmsuser 加密使用者 (CU) 身分登入 AWS CloudHSM 用戶端，然後輪換 kmsuser 密碼。只要 AWS CloudHSM 金鑰存放區已連接，AWS KMS 會保持登入 AWS CloudHSM 用戶端。

若要建立連線，AWS KMS 會在叢集的 Virtual Private Cloud (VPC) 中建立名為 kms-*<custom key store ID>* 的安全群組。安全群組有一個規則，可允許來自叢集安全群組的傳入流量。AWS KMS 也會在叢集適用的私有子網路中的每個可用區域中建立彈性網路界面 (ENI)。AWS KMS 會將 ENI 新增到 kms-*<cluster ID>* 安全群組和叢集的安全群組。每個 ENI 的描述為 KMS managed ENI for cluster *<cluster-ID>*。

連接程序可能需要很長的時間才能完成；最多 20 分鐘。

連接 AWS CloudHSM 金鑰存放區之前，請驗證它符合需求。

- 它的相關聯 AWS CloudHSM 叢集必須至少包含一個作用中 HSM。若要尋找叢集中的 HSM 數目，請在AWS CloudHSM主控台中檢視叢集或使用DescribeClusters作業。如有需要，您可以新增HSM。

- 叢集必須有 kmsuser 加密使用者 (CU) 帳戶，但是當您連接 AWS CloudHSM 金鑰存放區時，該 CU 無法登入叢集。如需登出的說明，請參閱如何登出和重新連線。
- AWS CloudHSM 金鑰存放區的連接狀態不能是 DISCONNECTING 或 FAILED。若要檢視連線狀態，請使用AWS KMS主控台或DescribeCustomKeyStores回應。如果連接狀態為 FAILED，請中斷連接自訂金鑰存放區，解決問題，然後進行連接。

如需連線失敗的協助，請參閱 如何修正連線失敗。

連接您的 AWS CloudHSM 金鑰存放區時，您可以在其中建立 KMS 金鑰，並在密碼編譯操作中使用現有的 KMS 金鑰。

中斷連接 AWS CloudHSM 金鑰存放區

中斷連接 AWS CloudHSM 金鑰存放區時，AWS KMS 會從 AWS CloudHSM 用戶端登出，與相關聯的 AWS CloudHSM 叢集中斷連接，並移除它建立用於支援連接的網路基礎設施。

當 AWS CloudHSM 金鑰存放區中斷連接時，您可以管理 AWS CloudHSM 金鑰存放區和其 KMS 金鑰，但無法在 AWS CloudHSM 金鑰存放區中建立或使用 KMS 金鑰。金鑰存放區的連接狀態為 DISCONNECTED，並且自訂金鑰存放區中 KMS 金鑰的金鑰狀態為 Unavailable，除非其為 PendingDeletion。您可以隨時重新連接 AWS CloudHSM 金鑰存放區。

當您中斷連接自訂金鑰存放區時，該金鑰存放區中的 KMS 金鑰會立即變成無法使用 (視最終一致性而定)。不過，使用受 KMS 金鑰保護之資料金鑰所加密的資源不會受影響，除非再次使用 KMS 金鑰 (例如解密資料金鑰)。此問題會影響 AWS 服務，其中許多服務會使用資料金鑰來保護您的資源。如需詳細資訊，請參閱 無法使用的 KMS 金鑰如何影響資料金鑰。

> **Note**
>
> 當自訂金鑰存放區中斷連接時，所有在自訂金鑰存放區中建立 KMS 金鑰的嘗試，或在密碼編譯操作中使用現有 KMS 金鑰的嘗試，均會失敗。此動作可防止使用者存放和存取敏感資料。

若要更好地預估中斷連接您的自訂金鑰存放區的效果，請在自訂金鑰存放區中識別 KMS 金鑰，並判斷其過去的使用情形。

您可能中斷連接 AWS CloudHSM 金鑰存放區的原因如下所示：

- 為了輪換 **kmsuser** 密碼。AWS KMS 會在每次連接到 AWS CloudHSM 叢集時變更 kmsuser 密碼。若要強制密碼輪換，只需中斷連接並重新連接。

- 稽核 AWS CloudHSM 叢集中 KMS 金鑰的金鑰材料。當您中斷連線自訂金鑰存放區時，AWS KMS 會登出 AWS CloudHSM 用戶端的 kmsuser 加密使用者帳戶。這可讓您以 kmsuser CU 身分登入叢集，並稽核和管理 KMS 金鑰的金鑰材料。
- 立即停用 AWS CloudHSM 金鑰存放區中的所有 KMS 金鑰。您可以使用AWS Management Console或DisableKey作業停用和重新啟用金AWS CloudHSM鑰存放區中的 KMS 金鑰。這些操作會快速完成，但一次只能對一個 KMS 金鑰進行。立即中斷連接 AWS CloudHSM 金鑰存放區會將 AWS CloudHSM 金鑰存放區中所有 KMS 金鑰的金鑰狀態變更為 Unavailable，這會防止在任何密碼編譯操作中對其進行使用。
- 為了修復失敗的連接嘗試。如果嘗試連接 AWS CloudHSM 金鑰存放區失敗 (自訂金鑰存放區的連接狀態為 FAILED)，您必須先中斷連接 AWS CloudHSM 金鑰存放區，之後再嘗試重新連接。

連接 AWS CloudHSM 金鑰存放區 (主控台)

若要連接 AWS Management Console 中的 AWS CloudHSM 金鑰存放區，請從 Custom key stores (自訂金鑰存放區) 頁面選擇 AWS CloudHSM 金鑰存放區開始。此連接程序最長可能需要 20 分鐘的時間才能完成。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格依次選擇自訂金鑰存放區、AWS CloudHSM 金鑰存放區。
4. 選擇您想要連接的 AWS CloudHSM 金鑰存放區列。

   如 AWS CloudHSM 金鑰存放區的狀態為 Failed (失敗)，則在連線之前，您必須中斷連接自訂金鑰存放區。
5. 從 Key store actions (金鑰存放區動作) 選單中，選擇 Connect (連接)。

AWS KMS 會開始連接您的自訂金鑰存放區的程序。它會找到相關的 AWS CloudHSM 叢集，建置所需的網路基礎設施，連接它，以 kmsuser CU 身分登入 AWS CloudHSM 叢集，並輪換 kmsuser 密碼。當操作完成時，連接狀態會變為 CONNECTED (已連接)。

如果部署失敗，說明失敗原因的錯誤訊息會顯示。在您嘗試重新連接之前，請檢視 AWS CloudHSM 金鑰存放區的連接狀態。如狀態為 FAILED (失敗)，在重新連接之前，您必須先中斷連接自訂金鑰存放區。如果您需要協助，請參閱對自訂金鑰存放區進行故障診斷。

下一步：the section called “在 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰”。

連線自訂金鑰存放區 (API)

若要連線已中斷連線的AWS CloudHSM金鑰存放區，請使用ConnectCustomKeyStore作業。相關聯的 AWS CloudHSM 叢集必須至少包含一個作用中的 HSM，並且連接狀態不能是 FAILED。

連接程序可能需要很長的時間才能完成；最多 20 分鐘。除非其快速失敗，否則 操作會傳回 HTTP 200 回應和不帶屬性的 JSON 物件。不過，這個初始回應並不表示連接已成功。若要判斷自訂金鑰存放區的連線狀態，請參閱DescribeCustomKeyStores回應。

本節中的範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

若要識別 AWS CloudHSM 金鑰存放區，請使用其自訂金鑰存放區 ID。您可以在主控台的 [自訂金鑰存放區] 頁面上找到 ID，或使用不含參數的DescribeCustomKeyStores作業。執行此範例之前，請將範例 ID 以有效的 ID 取代。

```
$ aws kms connect-custom-key-store --custom-key-store-id cks-1234567890abcdef0
```

若要確認AWS CloudHSM金鑰存放區是否已連線，請使用DescribeCustomKeyStores作業。在預設情況下，此操作會傳回您帳戶和區域中的所有自訂金鑰存放區。但是，您可以使用 CustomKeyStoreId 或 CustomKeyStoreName 參數 (但不能同時使用) 來限制對特定自訂金鑰存放區的回應。此 ConnectionState 值 CONNECTED 指出自訂的金鑰存放區連接到它的 AWS CloudHSM 叢集。

> **Note**
>
> 此 CustomKeyStoreType 欄位已新增至 DescribeCustomKeyStores 回應，以區分 AWS CloudHSM 金鑰存放區與外部金鑰存放區。

```
$ aws kms describe-custom-key-stores --custom-key-store-id cks-1234567890abcdef0
{
   "CustomKeyStores": [
      "CustomKeyStoreId": "cks-1234567890abcdef0",
      "CustomKeyStoreName": "ExampleCloudHSMKeyStore",
      "CloudHsmClusterId": "cluster-1a23b4cdefg",
      "CustomKeyStoreType": "AWS_CLOUDHSM",
      "TrustAnchorCertificate": "<certificate string appears here>",
      "CreationDate": "1.499288695918E9",
      "ConnectionState": "CONNECTED"
   ],
```

```
}
```

如果 ConnectionState 值為 failed，ConnectionErrorCode 元素會指出失敗的原因。在此情況下，AWS KMS 無法在您的帳戶中找到叢集 ID 為 cluster-1a23b4cdefg 的 AWS CloudHSM 叢集。如果已刪除叢集，則可以從原始叢集的備份還原叢集，然後編輯自訂金鑰存放區的叢集 ID。如需回應連接錯誤代碼的說明，請參閱 如何修正連線失敗。

```
$ aws kms describe-custom-key-stores --custom-key-store-id cks-1234567890abcdef0
{
   "CustomKeyStores": [
      "CustomKeyStoreId": "cks-1234567890abcdef0",
      "CustomKeyStoreName": "ExampleKeyStore",
      "CloudHsmClusterId": "cluster-1a23b4cdefg",
      "CustomKeyStoreType": "AWS_CLOUDHSM",
      "TrustAnchorCertificate": "<certificate string appears here>",
      "CreationDate": "1.499288695918E9",
      "ConnectionState": "FAILED"
      "ConnectionErrorCode": "CLUSTER_NOT_FOUND"
   ],
}
```

下一步：在 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰。

中斷連接 AWS CloudHSM 金鑰存放區 (主控台)

若要中斷連接 AWS Management Console 中已連接的 AWS CloudHSM 金鑰存放區，請從 Custom Key Stores (自訂金鑰存放區) 頁面選擇 AWS CloudHSM 金鑰存放區開始。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格依次選擇自訂金鑰存放區、AWS CloudHSM 金鑰存放區。
4. 選擇您想要中斷連接的外部金鑰存放區列。
5. 從 Key store actions (金鑰存放區動作) 選單中，選擇 Disconnect (中斷連接)。

當操作完成時，連接狀態會從 Disconnecting (正在中斷連接) 變為 Disconnected (已中斷連接)。如果操作失敗，會出現錯誤訊息，其中描述問題並提供如何修正的協助。如果您需要更多協助，請參閱對自訂金鑰存放區進行故障診斷。

中斷連接 AWS CloudHSM 金鑰存放區 (API)

若要中斷連線的AWS CloudHSM金鑰存放區，請使用此DisconnectCustomKeyStore作業。如果操作成功，則 AWS KMS 傳回 HTTP 200 回應和不帶屬性的 JSON 物件。

本節中的範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

此範例會中斷連接 AWS CloudHSM 金鑰存放區。執行此範例之前，請將範例 ID 以有效的 ID 取代。

```
$ aws kms disconnect-custom-key-store --custom-key-store-id cks-1234567890abcdef0
```

若要確認AWS CloudHSM金鑰存放區已中斷連線，請使用DescribeCustomKeyStores作業。
在預設情況下，此操作會傳回您帳戶和區域中的所有自訂金鑰存放區。但是，您可以使用 CustomKeyStoreId 和 CustomKeyStoreName 參數 (但不能同時使用) 來限制對特定自訂金鑰存放區的回應。DISCONNECTED 的 ConnectionState 值指示此 AWS CloudHSM 金鑰存放區範例未連接到其 AWS CloudHSM 叢集。

```
$ aws kms describe-custom-key-stores --custom-key-store-id cks-1234567890abcdef0
{
   "CustomKeyStores": [
      "CloudHsmClusterId": "cluster-1a23b4cdefg",
      "ConnectionState": "DISCONNECTED",
      "CreationDate": "1.499288695918E9",
      "CustomKeyStoreId": "cks-1234567890abcdef0",
      "CustomKeyStoreName": "ExampleKeyStore",
      "CustomKeyStoreType": "AWS_CLOUDHSM",
      "TrustAnchorCertificate": "<certificate string appears here>"
   ],
}
```

刪除 AWS CloudHSM 金鑰存放區

刪除 AWS CloudHSM 金鑰存放區時，AWS KMS 會從 KMS 刪除有關 AWS CloudHSM 金鑰存放區的所有中繼資料，包括其與 AWS CloudHSM 叢集關聯的資訊。這個操作不會影響 AWS CloudHSM 叢集、其 HSM 或其使用者。您可以建立與相同 AWS CloudHSM 叢集相關聯的新 AWS CloudHSM 金鑰存放區，但無法復原刪除操作。

您只能刪除已與 AWS CloudHSM 叢集中斷連接並且不包含任何 AWS KMS keys 的 AWS CloudHSM 金鑰存放區。刪除自訂金鑰存放區之前，請執行下列動作。

- 驗證您將不再需要使用金鑰存放區中的任何 KMS 金鑰進行任何密碼編譯操作。然後，排程從金鑰存放區刪除所有 KMS 金鑰。如需有關在 AWS CloudHSM 金鑰存放區中尋找 KMS 金鑰的說明，請參閱 在 AWS CloudHSM 金鑰存放區中尋找 KMS 金鑰。
- 確認已刪除所有 KMS 金鑰。若要檢視 AWS CloudHSM 金鑰存放區中的 KMS 金鑰，請參閱 檢視 AWS CloudHSM 金鑰存放區中的 KMS 金鑰。
- 中斷 AWS CloudHSM 金鑰存放區與其 AWS CloudHSM叢集的連接。

不要刪除 AWS CloudHSM 金鑰存放區，而是考慮從其關聯的 AWS CloudHSM 叢集中斷連接。當 AWS CloudHSM 金鑰存放區中斷連接時，您可以管理 AWS CloudHSM 金鑰存放區和其 AWS KMS keys。但您無法在 AWS CloudHSM 金鑰存放區中建立或使用 KMS 金鑰。您可以隨時重新連接 AWS CloudHSM 金鑰存放區。

主題

- 刪除 AWS CloudHSM 金鑰存放區 (主控台)
- 刪除 AWS CloudHSM 金鑰存放區 (API)

刪除 AWS CloudHSM 金鑰存放區 (主控台)

若要刪除 AWS Management Console 中的 AWS CloudHSM 金鑰存放區，請從 Custom key stores (自訂金鑰存放區) 頁面選擇 AWS CloudHSM 金鑰存放區開始。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格依次選擇自訂金鑰存放區、AWS CloudHSM 金鑰存放區。
4. 尋找代表您要刪除之 AWS CloudHSM 金鑰存放區的資料列。如 AWS CloudHSM 金鑰存放區的連接狀態不是 Disconnected (中斷連接)，則在刪除之前，您必須中斷連接 AWS CloudHSM 金鑰存放區。
5. 從 Key store actions (金鑰存放區動作) 選單中，選擇 Delete (刪除)。

當操作完成時，就會出現成功訊息，並且 AWS CloudHSM 金鑰存放區將不再顯示在金鑰存放區清單中。如果操作失敗，就會出現錯誤訊息，其中描述問題並提供如何修正的協助。如果您需要更多協助，請參閱對自訂金鑰存放區進行故障診斷。

刪除 AWS CloudHSM 金鑰存放區 (API)

若要刪除AWS CloudHSM金鑰存放區，請使用此DeleteCustomKeyStore作業。如果操作成功，則 AWS KMS 傳回 HTTP 200 回應和不帶屬性的 JSON 物件。

若要開始，請確認 AWS CloudHSM 金鑰存放區不包含任何 AWS KMS keys。您無法刪除包含 KMS 金鑰的自訂金鑰存放區。第一個範例命令使用ListKeys和DescribeKey在金鑰存放區中搜尋，其AWS KMS keys中包含 *cks-1234567* 890abcdef0 自訂AWS CloudHSM金鑰存放區識別碼範例。在此情況下，命令不會傳回任何 KMS 金鑰。如果是這樣，請使用此ScheduleKeyDeletion作業來排程刪除每個 KMS 金鑰。

Bash

```
for key in $(aws kms list-keys --query 'Keys[*].KeyId' --output text) ;
do aws kms describe-key --key-id $key |
grep '"CustomKeyStoreId": "cks-1234567890abcdef0"' --context 100; done
```

PowerShell

```
PS C:\> Get-KMSKeyList | Get-KMSKey | where CustomKeyStoreId -eq
 'cks-1234567890abcdef0'
```

接著中斷連接 AWS CloudHSM 金鑰存放區 此範例命令使用此DisconnectCustomKeyStore作業來中斷AWS CloudHSM金鑰存放區與其AWS CloudHSM叢集的連線。執行此命令之前，請將範例自訂金鑰存放區 ID 以有效的 ID 取代。

Bash

```
$ aws kms disconnect-custom-key-store --custom-key-store-id cks-1234567890abcdef0
```

PowerShell

```
PS C:\> Disconnect-KMSCustomKeyStore -CustomKeyStoreId cks-1234567890abcdef0
```

中斷自訂金鑰存放區之後，您可以使用該DeleteCustomKeyStore作業將其刪除。

Bash

```
$ aws kms delete-custom-key-store --custom-key-store-id cks-1234567890abcdef0
```

PowerShell

```
PS C:\> Remove-KMSCustomKeyStore -CustomKeyStoreId cks-1234567890abcdef0
```

管理 CloudHSM 金鑰存放區中的 KMS 金鑰

您可以在 AWS CloudHSM 金鑰存放區中建立、檢視、管理、使用和排程刪除 AWS KMS keys。您使用的處理程序非常類似於對其他 KMS 金鑰所用的程序。唯一的差別是您在建立 KMS 金鑰時會指定 AWS CloudHSM 金鑰存放區。然後，AWS KMS 會在與 AWS CloudHSM 金鑰存放區相關聯的 AWS CloudHSM 叢集中，為 KMS 金鑰建立不可擷取的金鑰材料。當您使用 AWS CloudHSM 金鑰存放區中的 KMS 金鑰時，在叢集的 HSM 中執行密碼編譯操作。

支援的功能

除了本節所討論的程序，您還可以對 AWS CloudHSM 金鑰存放區中的 KMS 金鑰執行下列操作：

- 使用金鑰政策、IAM 政策和授予，以授權存取 KMS 金鑰。
- 啟用和停用 KMS 金鑰。
- 指派標籤並建立別名，使用屬性型存取控制 (ABAC) 授權對 KMS 金鑰的存取。
- 將 KMS 金鑰用於密碼編譯操作，包括加密、解密、重新加密和產生資料金鑰。
- 將 KMS 金鑰用於與 AWS KMS 整合的 AWS 服務，且這些服務支援客戶受管金鑰。
- 在AWS CloudTrail日誌和 Amazon CloudWatch 監控工具中追蹤 KMS 金鑰的使用情況。

不支援的功能

- AWS CloudHSM 金鑰存放區僅支援對稱加密 KMS 金鑰。您無法在 AWS CloudHSM 金鑰存放區中建立 HMAC KMS 金鑰、非對稱 KMS 金鑰或非對稱資料金鑰對。
- 您無法匯入金鑰材料至 AWS CloudHSM 金鑰存放區中的 KMS 金鑰。AWS KMS 會針對 AWS CloudHSM 叢集中的 KMS 金鑰產生金鑰材料。
- 您無法啟用或停用自動輪換 AWS CloudHSM 金鑰存放區中 KMS 金鑰的金鑰材料。

主題

- 在 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰
- 檢視 AWS CloudHSM 金鑰存放區中的 KMS 金鑰
- 使用 AWS CloudHSM 金鑰存放區中的 KMS 金鑰
- 尋找 KMS 金鑰和金鑰材料
- 排程從 AWS CloudHSM 金鑰存放區刪除 KMS 金鑰

在 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰

建立 AWS CloudHSM 金鑰存放區之後，您可以在金鑰存放區建立 AWS KMS keys。它們必須是含有 AWS KMS 產生之金鑰資料的對稱加密 KMS 金鑰。您無法在自訂金鑰存放區中建立非對稱 KMS 金鑰、HMAC KMS 金鑰，或是含有匯入金鑰資料的 KMS 金鑰。此外，您無法在自訂金鑰存放區中使用對稱加密 KMS 金鑰來產生非對稱資料金鑰對。

若要在 AWS CloudHSM 金鑰存放區建立 KMS 金鑰，AWS CloudHSM 金鑰存放區必須連接到相關聯的 AWS CloudHSM 叢集，且叢集必須在不同的可用區域包含至少兩個作用中 HSM。若要尋找 HSM 的連接狀態和數目，請檢視 AWS Management Console 中的 AWS CloudHSM 金鑰存放區頁面。使用 API 作業時，請使用DescribeCustomKeyStores作業驗證AWS CloudHSM金鑰存放區是否已連線。若要驗證叢集中作用中 HSM 的數目及其可用區域，請使用此AWS CloudHSMDescribeClusters作業。

當您在 AWS CloudHSM 金鑰存放區建立 KMS 金鑰時，AWS KMS 會在 AWS KMS 中建立 KMS 金鑰。但是，它會在相關聯的 AWS CloudHSM 叢集內建立 KMS 金鑰的金鑰材料。特別的是，AWS KMS 會以您建立的 `kmsuser` CU 登入叢集。然後，它會在叢集建立持久性、不可擷取的 256 位元進階加密標準 (AES) 對稱金鑰。AWS KMS 會將只在叢集內可見的金鑰標籤屬性的值，設定為 KMS 金鑰的 Amazon Resource Name (ARN)。

當命令成功時，新 KMS 金鑰的金鑰狀態是 `Enabled`，而其來源是 `AWS_CLOUDHSM`。建立任何 KMS 金鑰之後就無法變更其來源。當您在AWS KMS主控台的金AWS CloudHSM鑰存放區中檢視 KMS 金鑰或使用DescribeKey作業時，您可以看到一般屬性，例如其金鑰識別碼、金鑰狀態和建立日期。但是，您也可以查看自訂金鑰存放區 ID 和 (選擇性) AWS CloudHSM 叢集 ID。如需詳細資訊，請參閱 檢視 AWS CloudHSM 金鑰存放區中的 KMS 金鑰。

如果您嘗試在 AWS CloudHSM 金鑰存放區建立 KMS 金鑰失敗，請使用錯誤訊息來協助您判斷原因。它可能指出 AWS CloudHSM 金鑰存放區未連接 (`CustomKeyStoreInvalidStateException`)，或相關聯的 AWS CloudHSM 叢集沒有此操作所需的兩個作用中 HSM (`CloudHsmClusterInvalidConfigurationException`)。如需協助，請參閱對自訂金鑰存放區進行故障診斷。

如需會在 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰之操作的 AWS CloudTrail 日誌範例，請參閱 CreateKey。

主題

- 在 AWS CloudHSM 金鑰存放區建立 KMS 金鑰 (主控台)
- 在 AWS CloudHSM 金鑰存放區建立 KMS 金鑰 (API)

在 AWS CloudHSM 金鑰存放區建立 KMS 金鑰 (主控台)

請使用下列程序在 AWS CloudHSM 金鑰存放區建立對稱加密 KMS 金鑰。

> **Note**
>
> 請勿在別名、說明或標籤包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 選擇建立金鑰。
5. 選擇 Symmetric (對稱)。
6. 在 Key usage (金鑰用途) 欄位中，系統會自動選取 Encrypt and decrypt (加密和解密) 選項。請勿變更該欄位。
7. 選擇 Advanced options (進階選項)。
8. 對於金鑰資料來源，選擇 AWS CloudHSM 金鑰存放區。

   您無法在 AWS CloudHSM 金鑰存放區建立多區域金鑰。
9. 選擇下一步。
10. 為新的 KMS 金鑰選取 AWS CloudHSM 金鑰存放區。若要建立新的 AWS CloudHSM 金鑰存放區，請選擇 Create custom key store (建立自訂金鑰存放區)。

    您選取的 AWS CloudHSM 金鑰存放區必須為 CONNECTED (已連接) 狀態。其相關聯的 AWS CloudHSM 叢集必須處於作用中，且在不同的可用區域至少包含兩個作用中 HSM。

如需有關連接 AWS CloudHSM 金鑰存放區的說明，請參閱 連接和中斷連接 AWS CloudHSM 金鑰存放區。如需新增 HSM 的說明，請參閱《AWS CloudHSM 使用者指南》中的新增 HSM。

11. 選擇下一步。
12. 輸入 KMS 金鑰的別名和選用描述。
13. (選用)。在 Add Tags (新增標籤) 頁面，新增標籤來識別或分類 KMS 金鑰。

    將標籤新增到 AWS 資源時，AWS 會產生成本配置報告，內含按標籤彙總的用量與成本。標籤也可以用來控制 KMS 金鑰的存取。如需標記 KMS 金鑰的詳細資訊，請參閱 標記金鑰 和 AWS KMS 的 ABAC。
14. 選擇下一步。
15. 在 Key Administrators (金鑰管理員) 區段中，選取可管理 KMS 金鑰的 IAM 使用者和角色。如需詳細資訊，請參閱允許金鑰管理員來管理 KMS 金鑰。

    > **Note**
    >
    > IAM 政策可授權其他 IAM 使用者和角色來使用 KMS 金鑰。
    > IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

16. (選用) 若要防止這些金鑰管理員刪除此 KMS 金鑰，請清除頁面底部的 Allow key administrators to delete this key (允許金鑰管理員刪除此金鑰) 方塊。
17. 選擇下一步。
18. 在 This account (這個帳戶) 區段中，選取此 AWS 帳戶 中可在密碼編譯操作中使用 KMS 金鑰的 IAM 使用者和角色。如需詳細資訊，請參閱允許金鑰使用者使用 KMS 金鑰。

    > **Note**
    >
    > IAM 政策可授權其他 IAM 使用者和角色來使用 KMS 金鑰。
    > IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

19. (選用) 您可以允許其他 AWS 帳戶 將此 KMS 金鑰用於密碼編譯操作。若要這樣做，請在頁面底部的其他 AWS 帳戶 區段中，選擇新增另一個 AWS 帳戶，然後輸入外部帳戶的 AWS 帳戶 ID。若要新增多個外部帳戶，請重複此步驟。

> **Note**
>
> 其他 AWS 帳戶 的管理員也必須透過為其使用者建立 IAM 政策，來允許存取 KMS 金鑰。如需詳細資訊，請參閱 允許其他帳戶中的使用者使用 KMS 金鑰。

20. 選擇下一步。
21. 檢閱您選擇的金鑰設定。您仍然可以返回並變更所有設定。
22. 完成時，請選擇 Finish (完成) 以建立金鑰。

當處理程序成功時，畫面會在您選擇的 AWS CloudHSM 金鑰存放區中顯示新的 KMS 金鑰。當您選擇新 KMS 金鑰的名稱或別名時，它的詳細資訊頁面上的 Cryptographic configuration (密碼編譯組態) 標籤會顯示 KMS 金鑰的來源 (AWS CloudHSM)、名稱、ID、自訂金鑰存放區類型以及 AWS CloudHSM 叢集的 ID。如果程序失敗，則會出現錯誤訊息來描述失敗。

> **Tip**
>
> 為了更輕鬆識別自訂金鑰存放區中的 KMS 金鑰，請在 Customer managed keys (客戶受管金鑰) 頁面上，新增要顯示的 Custom key store ID (自訂金鑰存放區 ID) 資料欄。按一下右上方的齒輪圖示，然後選取 Custom key store ID (自訂金鑰存放區 ID)。如需詳細資訊，請參閱 自訂您的 KMS 金鑰資料表。

在 AWS CloudHSM 金鑰存放區建立 KMS 金鑰 (API)

若要在金鑰存放區中建立新的 AWS KMS key(KMS 金AWS CloudHSM鑰)，請使用此CreateKey作業。使用 `CustomKeyStoreId` 參數來識別您的自訂金鑰存放區，並指定 `Origin` 值為 `AWS_CLOUDHSM`。

您可能還想要使用 `Policy` 參數來指定金鑰政策。您可以隨時變更金鑰原則 (PutKeyPolicy) 並新增選擇性元素，例如說明和標籤。

本節中的範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

下列範例會從呼叫DescribeCustomKeyStores作業開始，以驗證AWS CloudHSM金鑰存放區是否已連線至其關聯的AWS CloudHSM叢集。在預設情況下，此操作會傳回您帳戶和區域中的所有自訂金鑰存放區。若要只描述特定的 AWS CloudHSM 金鑰存放區，請使用 `CustomKeyStoreId` 或 `CustomKeyStoreName` 參數 (但不是都使用)。

執行此命令之前，請將範例自訂金鑰存放區 ID 換成有效的 ID。

> **Note**
>
> 請勿在 `Description` 或 `Tags` 欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

```
$ aws kms describe-custom-key-stores --custom-key-store-id cks-1234567890abcdef0
{
   "CustomKeyStores": [
      "CustomKeyStoreId": "cks-1234567890abcdef0",
      "CustomKeyStoreName": "ExampleKeyStore",
      "CustomKeyStoreType": "AWS CloudHSM key store",
      "CloudHsmClusterId": "cluster-1a23b4cdefg",
      "TrustAnchorCertificate": "<certificate string appears here>",
      "CreationDate": "1.499288695918E9",
      "ConnectionState": "CONNECTED"
   ],
}
```

下一個範例命令會使用DescribeClusters作業來確認與 (AWS CloudHSM叢集 1a23b4cdefgExampleKeyStore) 相關聯的叢集至少有兩個作用中的 HSM。如果叢集的 HSM 少於兩個，`CreateKey` 操作會失敗。

```
$ aws cloudhsmv2 describe-clusters
{
    "Clusters": [
        {
            "SubnetMapping": {
               ...
            },
            "CreateTimestamp": 1507133412.351,
            "ClusterId": "cluster-1a23b4cdefg",
            "SecurityGroup": "sg-865af2fb",
            "HsmType": "hsm1.medium",
            "VpcId": "vpc-1a2b3c4d",
            "BackupPolicy": "DEFAULT",
            "Certificates": {
                "ClusterCertificate": "-----BEGIN CERTIFICATE-----\...\n-----END
 CERTIFICATE-----\n"
            },
```

```
            "Hsms": [
                {
                    "AvailabilityZone": "us-west-2a",
                    "EniIp": "10.0.1.11",
                    "ClusterId": "cluster-1a23b4cdefg",
                    "EniId": "eni-ea8647e1",
                    "StateMessage": "HSM created.",
                    "SubnetId": "subnet-a6b10bd1",
                    "HsmId": "hsm-abcdefghijk",
                    "State": "ACTIVE"
                },
                {
                    "AvailabilityZone": "us-west-2b",
                    "EniIp": "10.0.0.2",
                    "ClusterId": "cluster-1a23b4cdefg",
                    "EniId": "eni-ea8647e1",
                    "StateMessage": "HSM created.",
                    "SubnetId": "subnet-b6b10bd2",
                    "HsmId": "hsm-zyxwvutsrqp",
                    "State": "ACTIVE"
                },
            ],
            "State": "ACTIVE"
        }
    ]
}
```

此範例命令使用CreateKey作業在金鑰存放區中建立 KMS 金AWS CloudHSM鑰。若要在 AWS CloudHSM 金鑰存放區建立 KMS 金鑰，您必須提供 AWS CloudHSM 金鑰存放區的自訂金鑰存放區 ID，並指定 `Origin` 值為 `AWS_CLOUDHSM`。

回應包含自訂金鑰存放區和 AWS CloudHSM 叢集的 ID。

執行此命令之前，請將範例自訂金鑰存放區 ID 換成有效的 ID。

```
$ aws kms create-key --origin AWS_CLOUDHSM --custom-key-store-id cks-1234567890abcdef0
{
  "KeyMetadata": {
    "AWSAccountId": "111122223333",
    "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "CreationDate": 1.499288695918E9,
    "Description": "Example key",
```

```
    "Enabled": true,
    "MultiRegion": false,
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "Origin": "AWS_CLOUDHSM"
    "CloudHsmClusterId": "cluster-1a23b4cdefg",
    "CustomKeyStoreId": "cks-1234567890abcdef0"
    "KeySpec": "SYMMETRIC_DEFAULT",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "EncryptionAlgorithms": [
        "SYMMETRIC_DEFAULT"
    ]
  }
}
```

檢視 AWS CloudHSM 金鑰存放區中的 KMS 金鑰

若要檢視 AWS CloudHSM 金鑰存放區中的 AWS KMS keys，請使用您檢視任何 AWS KMS 客戶受管金鑰時同樣的技巧。若要學習基本操作，請參閱檢視金鑰。若要識別 AWS CloudHSM 叢集內的金鑰 (作為 KMS 金鑰的金鑰材料)，請參閱 尋找 KMS 金鑰和金鑰材料。如需檢視記錄自訂金鑰存放區上所有 API 操作之 AWS CloudTrail 日誌的詳細資訊，請參閱 使用 AWS CloudTrail 記錄 AWS KMS API 呼叫。

在 AWS KMS 主控台中，除了 AWS 帳戶 和區域中的其他所有客戶受管金鑰，自訂金鑰存放區中的 KMS 金鑰也會顯示在客戶受管金鑰頁面。

不過，以下是 AWS CloudHSM 金鑰存放區中的 KMS 金鑰所特有的值。

- 存放 KMS 金鑰的 AWS CloudHSM 金鑰存放區的名稱和 ID。
- 相關聯的 AWS CloudHSM 叢集的叢集 ID，其中包含其金鑰資料。
- AWS KMS 主控台中的 `Origin` 值為 `AWS CloudHSM`，或在 API 回應中為 `AWS_CLOUDHSM`。
- 金鑰狀態值可以是 `Unavailable`。如需解析狀態的說明，請參閱如何修正無法使用的 KMS 金鑰。

檢視 AWS CloudHSM 金鑰存放區中的 KMS 金鑰 (主控台)

1. 開啟位於 https://console.aws.amazon.com/kms 的 AWS KMS 主控台。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。

4. 在右上角，選擇齒輪圖示，選擇 Custom key store ID (自訂金鑰存放區 ID) 和 Origin (來源)，然後選擇 Confirm (確認)。
5. 若要在任何 AWS CloudHSM 金鑰存放區中識別 KMS 金鑰，請尋找 Origin (來源) 值為 AWS CloudHSM 的 KMS 金鑰。若要識別特定 AWS CloudHSM 金鑰存放區中的 KMS 金鑰，請檢視 Custom key store ID (自訂金鑰存放區 ID) 資料欄的值。
6. 選擇 AWS CloudHSM 金鑰存放區中 KMS 金鑰的別名或金鑰 ID。

   這個頁面會顯示 KMS 金鑰的詳細資訊，包括其 Amazon Resource Name (ARN)、金鑰政策和標籤。
7. 選擇 Cryptographic configuration (密碼編譯組態) 索引標籤。索引標籤位於 General Configuration (一般組態) 區段下。

   此區段包含 KMS 金鑰相關聯的 AWS CloudHSM 金鑰存放區和 AWS CloudHSM 叢集的資訊。

檢視自訂金鑰存放區中的 KMS 金鑰 (API)

您可以使用相同的 AWS KMS API 操作來檢視金AWS CloudHSM鑰存放區中要用於任何 KMS 金鑰 (包括ListKeysDescribeKey、和) 的 KMS 金鑰GetKeyPolicy。例如，AWS CLI 中的以下 `describe-key` 操作會顯示 AWS CloudHSM 金鑰存放區中 KMS 金鑰的特殊欄位。執行像這樣的命令之前，請將範例 KMS 金鑰 ID 換成有效值。

```
$ aws kms describe-key --key-id 1234abcd-12ab-34cd-56ef-1234567890ab

{
   "KeyMetadata": {
      "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
      "AWSAccountId": "111122223333",
      "CloudHsmClusterId": "cluster-1a23b4cdefg",
      "CreationDate": 1537582718.431,
      "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
      "CustomKeyStoreId": "cks-1234567890abcdef0",
      "Description": "Key in custom key store",
      "Enabled": true,
      "EncryptionAlgorithms": [
         "SYMMETRIC_DEFAULT"
      ],
      "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
      "KeyManager": "CUSTOMER",
      "KeySpec": "SYMMETRIC_DEFAULT",
```

```
      "KeyState": "Enabled",
      "KeyUsage": "ENCRYPT_DECRYPT",
      "MultiRegion": false,
      "Origin": "AWS_CLOUDHSM"
   }
}
```

如需在 AWS CloudHSM 金鑰存放區中尋找 KMS 金鑰，或識別 AWS CloudHSM 叢集內的金鑰以作為 KMS 金鑰的金鑰材料，相關說明請參閱 尋找 KMS 金鑰和金鑰材料。

使用 AWS CloudHSM 金鑰存放區中的 KMS 金鑰

在 AWS CloudHSM 金鑰存放區中建立對稱加密 KMS 金鑰之後，您可以將它用於下列密碼編譯操作：

- 加密
- 解密
- GenerateDataKey
- GenerateDataKeyWithoutPlaintext
- ReEncrypt

自訂金鑰存放區不支援產生非對稱資料金鑰配對GenerateDataKeyPair和GenerateDataKeyPairWithoutPlaintext的作業。

在請求中使用您的 KMS 金鑰時，請依其 ID 或別名來識別 KMS 金鑰；您不需要指定 AWS CloudHSM 金鑰存放區或 AWS CloudHSM 叢集。回應包含為針對任何對稱加密 KMS 金鑰所傳回的相同欄位。

不過，當您使用 AWS CloudHSM 金鑰存放區中的 KMS 金鑰時，密碼編譯操作完全是在與 AWS CloudHSM 金鑰存放區相關聯的 AWS CloudHSM 叢集內執行。操作會使用叢集內與您選擇的 KMS 金鑰相關聯的金鑰材料。

但必須符合以下條件，才有可能這樣做。

- KMS 金鑰的金鑰狀態必須是 `Enabled`。若要尋找金鑰狀態，請使用AWS KMS主控台中的 [狀態`KeyState`] 欄位或DescribeKey回應中的欄位。
- AWS CloudHSM 金鑰存放區必須連接到其 AWS CloudHSM 叢集。其在AWS KMS控制台或DescribeCustomKeyStores響應`ConnectionState`中的狀態必須是`CONNECTED`。
- 與自訂金鑰存放區相關聯的 AWS CloudHSM 叢集至少必須包含一個作用中的 HSM。若要尋找叢集中作用中 HSM 的數目，請使用AWS KMS主控台、AWS CloudHSM主控台或DescribeClusters作業。

- AWS CloudHSM 叢集必須包含 KMS 金鑰的金鑰材料。如果已從叢集刪除金鑰資料，或 HSM 是從不含金鑰資料的備份所建立，則密碼編譯操作會失敗。

如果不符合這些條件，密碼編譯操作會失敗，且 AWS KMS 會傳回 `KMSInvalidStateException` 例外狀況。一般而言，您只需要重新連接 AWS CloudHSM 金鑰存放區。如需其他說明，請參閱如何修正失效的 KMS 金鑰。

在 AWS CloudHSM 金鑰存放區中使用 KMS 金鑰時，請注意每個 AWS CloudHSM 金鑰存放區中的 KMS 金鑰會針對密碼編譯操作共用自訂金鑰存放區請求配額。如果您超過配額，則 AWS KMS 會傳回 `ThrottlingException`。如果與 AWS CloudHSM 金鑰存放區相關聯的 AWS CloudHSM 叢集正在處理許多命令，包括與 AWS CloudHSM 金鑰存放區無關的命令，您甚至會在較低的速率時收到 `ThrottlingException`。如果您收到任何請求的 `ThrottlingException`，請降低請求速率，然後再試一次命令。如需自訂金鑰存放區配額的詳細資訊，請參閱 自訂金鑰存放區請求配額。

尋找 KMS 金鑰和金鑰材料

如果管理 AWS CloudHSM 金鑰存放區，則您可能需要識別每個 AWS CloudHSM 金鑰存放區中的 KMS 金鑰。例如，您可能需要執行下列一些任務。

- 在 AWS CloudTrail 日誌中追蹤 AWS CloudHSM 金鑰存放區中的 KMS 金鑰。
- 預測中斷連接 AWS CloudHSM 金鑰存放區對 KMS 金鑰造成的影響。
- 在刪除 AWS CloudHSM 金鑰存放區之前排程刪除 KMS 金鑰。

此外，您可能需要識別 AWS CloudHSM 叢集內的金鑰，這些金鑰作為 KMS 金鑰的金鑰材料。雖然 AWS KMS 會管理 KMS 金鑰及其金鑰材料，但您需要控制和負責管理您的 AWS CloudHSM 叢集、HSM 和備份，以及在 HSM 中的金鑰。您可能需要識別金鑰，以稽核金鑰材料、避免意外刪除它，或在刪除 KMS 金鑰後從 HSM 和叢集備份中刪除它。

AWS CloudHSM 金鑰存放區中 KMS 金鑰的所有金鑰材料是由 `kmsuser` 加密使用者 (CU) 擁有。AWS KMS 會將只可在 AWS CloudHSM 中檢視的金鑰標籤屬性設定為 KMS 金鑰的 Amazon Resource Name (ARN)。

若要尋找 KMS 金鑰和金鑰材料，請使用下列任何技巧。

- 在 AWS CloudHSM 金鑰存放區中尋找 KMS 金鑰 – 如何在其中一個或所有 AWS CloudHSM 金鑰存放區中識別 KMS 金鑰。
- 尋找 AWS CloudHSM 金鑰存放區的所有金鑰 – 如何在叢集內尋找金鑰，這些金鑰全部作為 AWS CloudHSM 金鑰存放區中的 KMS 金鑰的金鑰材料。

- 尋找 KMS 金鑰的 AWS CloudHSM 金鑰 – 如何在叢集內尋找金鑰，此金鑰作為 AWS CloudHSM 金鑰存放區中特定 KMS 金鑰的金鑰材料。
- 尋找 AWS CloudHSM 金鑰的 KMS 金鑰 – 如何尋找叢集中特定金鑰的 KMS 金鑰。

在 AWS CloudHSM 金鑰存放區中尋找 KMS 金鑰

如果管理 AWS CloudHSM 金鑰存放區，則您可能需要識別每個 AWS CloudHSM 金鑰存放區中的 KMS 金鑰。您可以使用這項資訊在 AWS CloudTrail 日誌中追蹤 KMS 金鑰操作、預測中斷連接自訂金鑰存放區對 KMS 金鑰造成的影響，或在刪除 AWS CloudHSM 金鑰存放區之前排程刪除 KMS 金鑰。

尋找 AWS CloudHSM 金鑰存放區中的 KMS 金鑰 (主控台)

若要在特定的 AWS CloudHSM 金鑰存放區中尋找 KMS 金鑰，請在 Customer managed keys (客戶受管金鑰) 頁面上，檢視 Custom Key Store Name (自訂金鑰存放區名稱) 或 Custom Key Store ID (自訂金鑰存放區 ID) 欄位中的值。若要在任何 AWS CloudHSM 金鑰存放區中識別 KMS 金鑰，請尋找 Origin (來源) 值為 AWS CloudHSM 的 KMS 金鑰。若要將選用欄新增到畫面上，請選擇頁面右上角的齒輪圖示。

尋找 AWS CloudHSM 金鑰存放區中的 KMS 金鑰 (API)

若要尋找金鑰存放區中的 KMS 金AWS CloudHSM鑰，請使用ListKeys和DescribeKey作業，然後依`CustomKeyStoreId`值篩選。執行範例之前，請將虛構的自訂金鑰存放區 ID 值換成有效值。

Bash

若要在特定的 AWS CloudHSM 金鑰存放區中尋找 KMS 金鑰，請取得帳戶和區域中的所有 KMS 金鑰。然後按自訂金鑰存放區的 ID 進行篩選。

```
for key in $(aws kms list-keys --query 'Keys[*].KeyId' --output text) ;
do aws kms describe-key --key-id $key |
grep '"CustomKeyStoreId": "cks-1234567890abcdef0"' --context 100; done
```

若要取得帳戶和區域中任何 AWS CloudHSM 金鑰存放區中的 KMS 金鑰，請搜尋值為 `AWS_CloudHSM` 的 `CustomKeyStoreType`。

```
for key in $(aws kms list-keys --query 'Keys[*].KeyId' --output text) ;
do aws kms describe-key --key-id $key |
grep '"CustomKeyStoreType": "AWS_CloudHSM"' --context 100; done
```

PowerShell

若要尋找特定AWS CloudHSM金鑰存放區中的 KMS 金鑰，請使用 G et-KmsKeyList 和 G et-KmsKey 指令程式來取得帳戶和區域中的所有 KMS 金鑰。然後按自訂金鑰存放區的 ID 進行篩選。

```
PS C:\> Get-KMSKeyList | Get-KMSKey | where CustomKeyStoreId -eq
 'cks-1234567890abcdef0'
```

若要在帳戶和區域中的任何金AWS CloudHSM鑰存放區中取得 KMS 金鑰，請篩選的 CustomKeyStoreType 值AWS_CLOUDHSM。

```
PS C:\> Get-KMSKeyList | Get-KMSKey | where CustomKeyStoreType -eq 'AWS_CLOUDHSM'
```

尋找 AWS CloudHSM 金鑰存放區的所有金鑰

您可以在 AWS CloudHSM 叢集內識別金鑰，這些金鑰作為 AWS CloudHSM 金鑰存放區的金鑰材料。若要這麼做，請使用 cloudhsm_mgmt_util 中的findAllKeys命令，尋找擁有或共用之所有金鑰的金鑰控制代碼。kmsuser除非您以 kmsuser 身分登入，且在 AWS KMS 之外建立金鑰，否則 kmsuser 擁有的所有金鑰都代表 KMS 金鑰的金鑰材料。

在不中斷連接 AWS CloudHSM 金鑰存放區的情況下，叢集的任何加密主管都可以執行此命令。

1. 使用 CloudHSM Management Utility (CMU) 入門主題中所述的程序，啟動 cloudhsm_mgmt_util。
2. 使用加密主管 (CO) 帳戶登入 cloudhsm_mgmt_util。
3. 使用 listUsers 命令尋找 kmsuser 加密使用者的使用者 ID。

   在這個範例中，kmsuser 有使用者 ID 3。

```
aws-cloudhsm> listUsers
Users on server 0(10.0.0.1):
Number of users found:3

    User Id             User Type       User Name            MofnPubKey
 LoginFailureCnt         2FA
         1              PCO             admin                    NO
 0               NO
         2              AU              app_user                 NO
 0               NO
         3              CU              kmsuser                  NO
 0               NO
```

4. 使用命findAllKeys令尋找kmsuser擁有或共用之所有金鑰的金鑰控制代碼。將範例使用者 ID (3) 換成叢集內 kmsuser 的實際使用者 ID。

   範例輸出顯示 kmsuser 在叢集的兩個 HSM 上，擁有金鑰控制代碼為 8、9 和 262162 的金鑰。

```
aws-cloudhsm> findAllKeys 3 0
Keys on server 0(10.0.0.1):
Number of keys found 3
number of keys matched from start index 0::6
8,9,262162
findAllKeys success on server 0(10.0.0.1)

Keys on server 1(10.0.0.2):
Number of keys found 6
number of keys matched from start index 0::6
8,9,262162
findAllKeys success on server 1(10.0.0.2)
```

尋找 AWS CloudHSM 金鑰的 KMS 金鑰

如果您知道 kmsuser 在叢集內擁有的金鑰的金鑰控制代碼，則可以使用金鑰標籤來識別 AWS CloudHSM 金鑰存放區中相關聯的 KMS 金鑰。

當 AWS KMS 在您的 AWS CloudHSM 叢集內建立 KMS 金鑰的金鑰材料時，它會將 KMS 金鑰的 Amazon Resource Name (ARN) 寫入金鑰標籤中。除非您已變更標籤值，否則您可以在 key_mgmt_util 或 cloudhsm_mgmt_util 中使用 getAttribute 命令，將金鑰和其 KMS 金鑰建立關聯。

若要執行此程序，您需要暫時中斷連接 AWS CloudHSM 金鑰存放區，才能以 kmsuser CU 身分登入。

> **Note**
>
> 當自訂金鑰存放區中斷連接時，所有在自訂金鑰存放區中建立 KMS 金鑰的嘗試，或在密碼編譯操作中使用現有 KMS 金鑰的嘗試，均會失敗。此動作可防止使用者存放和存取敏感資料。

1. 中斷連接 AWS CloudHSM 金鑰存放區 (如果尚未中斷連接)，然後以 kmsuser 身分登入 key_mgmt_util，如 如何中斷連線和登入 中所述。
2. 使用 key_mgmt_util 或 cloudhsm_mgmt_util 中的 getAttribute 命令，來取得標籤屬性 (OBJ_ATTR_LABEL，屬性 3)，用於特定金鑰控制代碼。

例如，此命令在 cloudhsm_mgmt_util 中使用 `getAttribute`，以取得金鑰代碼為 262162 的金鑰的標籤屬性 (屬性 3)。輸出顯示對於 ARN 為 `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab` 的 KMS 金鑰，262162 作為 KMS 金鑰的金鑰材料。執行此命令之前，請將範例金鑰控制代碼換成有效的控制代碼。

如需金鑰屬性清單，請使用 listAttributes 命令或參閱《AWS CloudHSM 使用者指南》中的金鑰屬性參考。

```
aws-cloudhsm> getAttribute 262162 3

Attribute Value on server 0(10.0.1.10):
OBJ_ATTR_LABEL
arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
```

3. 登出 key_mgmt_util 或 cloudhsm_mgmt_util，然後如 如何登出和重新連線 中所述，重新連接 AWS CloudHSM 金鑰存放區。

尋找 KMS 金鑰的 AWS CloudHSM 金鑰

您可以使用 AWS CloudHSM 金鑰存放區中的 KMS 金鑰的 KMS 金鑰 ID，以識別 AWS CloudHSM 叢集內的金鑰，此金鑰作為其金鑰材料。然後，您可以使用其金鑰控制代碼，在 AWS CloudHSM 用戶端命令中識別金鑰。

當 AWS KMS 在您的 AWS CloudHSM 叢集內建立 KMS 金鑰的金鑰材料時，它會將 KMS 金鑰的 Amazon Resource Name (ARN) 寫入金鑰標籤中。除非您已變更標籤值，否則您可以在 key_mgmt_util 中使用 findKey 命令，以取得 KMS 金鑰的金鑰材料的金鑰控制代碼。若要執行此程序，您需要暫時中斷連接 AWS CloudHSM 金鑰存放區，才能以 `kmsuser` CU 身分登入。

> **Note**
>
> 當自訂金鑰存放區中斷連接時，所有在自訂金鑰存放區中建立 KMS 金鑰的嘗試，或在密碼編譯操作中使用現有 KMS 金鑰的嘗試，均會失敗。此動作可防止使用者存放和存取敏感資料。

1. 中斷連接 AWS CloudHSM 金鑰存放區 (如果尚未中斷連接)，然後以 `kmsuser` 身分登入 key_mgmt_util，如 如何中斷連線和登入 中所述。

2. 在 key_mgmt_util 中使用 findKey 命令來搜尋金鑰，此金鑰的標籤符合 AWS CloudHSM 金鑰存放區中 KMS 金鑰的 ARN。將 -l (小寫 L 代表 'label') 參數值中的範例 KMS 金鑰 ARN，取代為有效的 KMS 金鑰 ARN。

   例如，此命令會尋找標籤符合範例 KMS 金鑰 ARN `arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab` 的金鑰。範例輸出顯示金鑰控制代碼為 262162 的金鑰，在其標籤中具有指定的 KMS 金鑰 ARN。您現在可以在其他 key_mgmt_util 命令中使用此金鑰控制代碼。

```
Command: findKey -l arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
Total number of keys present 1

 number of keys matched from start index 0::1
262162

        Cluster Error Status
        Node id 1 and err state 0x00000000 : HSM Return: SUCCESS
        Node id 0 and err state 0x00000000 : HSM Return: SUCCESS

        Cfm3FindKey returned: 0x00 : HSM Return: SUCCESS
```

3. 登出 key_mgmt_util，然後如如何登出和重新連線所述，重新連接自訂金鑰存放區。

排程從 AWS CloudHSM 金鑰存放區刪除 KMS 金鑰

當您確定不再需要將 AWS KMS key 用於任何密碼編譯操作時，您可以排程刪除 KMS 金鑰。就像您排定從 AWS KMS 刪除任何 KMS 金鑰一樣，使用同樣的處理程序。此外，請讓 AWS CloudHSM 金鑰存放區保持連接，以便 AWS KMS 在等待期間到期時，能夠從相關聯的 AWS CloudHSM 叢集刪除對應的金鑰材料。

您可以監控 AWS CloudTrail 日誌中 KMS 金鑰的排程、取消和刪除。

> **Warning**
>
> 刪除 KMS 金鑰是一種破壞性和具有潛在危險的操作，您將無法復原以 KMS 金鑰加密的所有資料。在排程刪除 KMS 金鑰之前，請檢查 KMS 金鑰的過去使用情況，並建立 Amazon CloudWatch 警示，以便在有人嘗試使用 KMS 金鑰擱置刪除時向您發出警示。儘可能停用 KMS 金鑰，而不要刪除。

排程從 AWS CloudHSM 金鑰存放區刪除 KMS 金鑰時，其金鑰狀態會變更為 Pending deletion (等待刪除)。KMS 金鑰會在整個等待期保持在 Pending deletion (等待刪除) 狀態，即使 KMS 金鑰變為無法使用，因為您已中斷連接自訂金鑰存放區。這可讓您在等待期間隨時取消刪除 KMS 金鑰。

當等待期過期時，AWS KMS 會從 AWS KMS 刪除 KMS 金鑰。然後，AWS KMS 會盡可能從相關聯的 AWS CloudHSM 叢集刪除金鑰資料。如果 AWS KMS 無法刪除金鑰資料 (例如，當金鑰存放區與 AWS KMS 中斷連接時)，您可能需要手動從叢集刪除遺棄的金鑰資料。

AWS KMS 不會從叢集備份中刪除金鑰資料。即使您從 AWS KMS 刪除 KMS 金鑰並從 AWS CloudHSM 叢集刪除其金鑰材料，從備份建立的叢集仍可能包含已刪除的金鑰材料。若要永久刪除金鑰材料，請檢視 KMS 金鑰的建立日期。然後，刪除所有叢集備份，其中可能包含金鑰資料。

當您排程從 AWS CloudHSM 金鑰存放區刪除 KMS 金鑰時，該 KMS 金鑰會立即變為無法使用 (視最終一致性而定)。不過，使用受 KMS 金鑰保護之資料金鑰所加密的資源不會受影響，除非再次使用 KMS 金鑰 (例如解密資料金鑰)。此問題會影響 AWS 服務，其中許多服務會使用資料金鑰來保護您的資源。如需詳細資訊，請參閱 無法使用的 KMS 金鑰如何影響資料金鑰。

對自訂金鑰存放區進行故障診斷

AWS CloudHSM 金鑰存放區的設計是要具有可用性和彈性。不過，您可能需要修正一些錯誤情況，以讓您的 AWS CloudHSM 金鑰存放區能夠運作。

主題

- 如何修正無法使用的 KMS 金鑰
- 如何修正失效的 KMS 金鑰
- 如何修正連線失敗
- 如何回應密碼編譯操作失敗
- 如何修正無效的 kmsuser 登入資料
- 如何刪除遺棄的金鑰材料
- 如何復原 KMS 金鑰已刪除的金鑰材料
- 如何以 kmsuser 身分登入

如何修正無法使用的 KMS 金鑰

AWS CloudHSM 金鑰存放區中 AWS KMS keys 的金鑰狀態通常是 `Enabled`。如同所有 KMS 金鑰，當您停用 AWS CloudHSM 金鑰存放區中的 KMS 金鑰或排程進行刪除時，金鑰狀態會變更。不過，與其他 KMS 金鑰不同，自訂金鑰存放區中的 KMS 金鑰也可以有 `Unavailable` 的金鑰狀態。

Unavailable 的金鑰狀態表示自訂金鑰存放區中的 KMS 金鑰是特意中斷連接，並且會在失敗時 (如果有) 嘗試重新連接。當 KMS 金鑰無法使用時，您可以檢視和管理 KMS 金鑰，但無法將它用於密碼編譯操作。

若要尋找 KMS 金鑰的金鑰狀態，請在 Customer managed keys (客戶受管金鑰) 頁面上，檢視 KMS 金鑰的 Status (狀態) 欄位。或者，使用DescribeKey操作並查看響應中的KeyState元素。如需詳細資訊，請參閱 檢視金鑰。

中斷連線的自訂金鑰存放區中的 KMS 金鑰會有 Unavailable 或 PendingDeletion 的金鑰狀態。排程要從自訂金鑰存放區刪除的 KMS 金鑰會有 Pending Deletion 金鑰狀態，即使自訂金鑰存放區中斷連接時亦然。這可讓您取消排程的金鑰刪除，而無需重新連接自訂金鑰存放區。

若要修正無法使用的 KMS 金鑰，請重新連線自訂金鑰存放區。自訂金鑰存放區重新連接之後，自訂金鑰存放區中 KMS 金鑰的金鑰狀態會自動還原到其先前的狀態，例如 Enabled 或 Disabled。等待刪除的 KMS 金鑰會保持在 PendingDeletion 狀態。不過，當問題存在時，啟用和停用無法使用的 KMS 金鑰不會變更它的金鑰狀態。啟用或停用動作僅在金鑰可供使用時生效。

如需失敗的連接的協助，請參閱如何修正連線失敗。

如何修正失效的 KMS 金鑰

建立和使用 AWS CloudHSM 金鑰存放區中 KMS 金鑰的相關問題可能是由 AWS CloudHSM 金鑰存放區、其相關的 AWS CloudHSM 叢集、KMS 金鑰或其金鑰材料問題引起的。

當 AWS CloudHSM 金鑰存放區與其 AWS CloudHSM 叢集中斷連接時，自訂金鑰存放區中 KMS 金鑰的金鑰狀態會是 Unavailable。在已中斷連接的 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰的所有請求會傳回 CustomKeyStoreInvalidStateException 例外狀況。加密、解密、重新加密或產生資料金鑰的所有請求會傳回 KMSInvalidStateException 例外狀況。若要修正問題，請重新連接 AWS CloudHSM 金鑰存放區。

不過，嘗試將 AWS CloudHSM 金鑰存放區中的 KMS 金鑰用於密碼編譯操作可能會失敗，即使其金鑰狀態為 Enabled，並且 AWS CloudHSM 金鑰存放區的連接狀態為 Connected 亦然。這可能是因為以下任何情況所造成。

- 可能已從相關聯的 AWS CloudHSM 叢集刪除 KMS 金鑰的金鑰材料。若要調查，請尋找 KMS 金鑰之金鑰材料的金鑰控制代碼，並在必要時，嘗試還原金鑰材料。
- 已從與 AWS CloudHSM 金鑰存放區相關聯的 AWS CloudHSM 叢集中刪除所有 HSM。若要在密碼編譯操作中使用 AWS CloudHSM 金鑰存放區中的 KMS 金鑰，其 AWS CloudHSM 叢集必須至少包含一個作用中的 HSM。若要驗證AWS CloudHSM叢集中 HSM 的數目和狀態，請使用主AWS

CloudHSM控台或DescribeClusters作業。若要將 HSM 新增至叢集，請使用主AWS CloudHSM控台或CreateHsm作業。

- 已刪除與 AWS CloudHSM 金鑰存放區相關聯的 AWS CloudHSM 叢集。若要修正此問題，請從與原始叢集相關的備份 (例如，原始叢集的備份，或用於建立原始叢集的備份) 建立叢集。然後在自訂金鑰存放區設定中編輯叢集 ID。如需說明，請參閱如何復原 KMS 金鑰已刪除的金鑰材料。
- 與自訂金鑰存放區關聯的 AWS CloudHSM 叢集沒有任何可用的 PKCS #11 工作階段。這通常發生在高爆量流量期間，此時需要額外的工作階段來服務流量。若要回應帶有關於 PKCS #11 工作階段錯誤訊息的 `KMSInternalException`，請退回並重試請求。

如何修正連線失敗

如果您嘗試將 AWS CloudHSM 金鑰存放區連接到其 AWS CloudHSM 叢集，但操作失敗，AWS CloudHSM 金鑰存放區的連接狀態會變更為 `FAILED`。若要尋找AWS CloudHSM金鑰存放區的連線狀態，請使用AWS KMS主控台或DescribeCustomKeyStores作業。

或者，由於很容易偵測到叢集組態錯誤，一些連接嘗試會很快失敗。在這種情況下，連接狀態仍然是 `DISCONNECTED`。這些失敗會傳回錯誤訊息或 例外狀況來說明嘗試失敗的原因。檢閱例外狀況描述和叢集需求、修正問題、更新 AWS CloudHSM 金鑰存放區 (如有必要)，並嘗試重新連接。

當連接狀態為時`FAILED`，執行DescribeCustomKeyStores作業並查看回應中的`ConnectionErrorCode`元素。

> **Note**
>
> 如果 AWS CloudHSM 金鑰存放區的狀態為 `FAILED`，在嘗試重新連接之前，您必須中斷連接 AWS CloudHSM 金鑰存放區。您無法連接具有 `FAILED` 連接狀態的 AWS CloudHSM 金鑰存放區。

- `CLUSTER_NOT_FOUND` 表示 AWS KMS 找不到具有指定叢集 ID 的 AWS CloudHSM 叢集。發生此情況可能是因為提供給 API 操作的是錯誤的叢集 ID 或是已刪除叢集而無法取代。若要修正此錯誤，請驗證叢集 ID，例如使用AWS CloudHSM主控台或DescribeClusters作業。如果已刪除叢集，請從原始叢集的最近備份建立叢集。然後，中斷連接 AWS CloudHSM 金鑰存放區、編輯 AWS CloudHSM 金鑰存放區叢集 ID 設定，並將 AWS CloudHSM 金鑰存放區重新連接到叢集。
- `INSUFFICIENT_CLOUDHSM_HSMS` 指出相關聯的 AWS CloudHSM 叢集不包含任何 HSM。若要連接，叢集必須至少有一個 HSM。若要尋找叢集中的 HSM 數目，請使用此DescribeClusters作業。若

要解決此錯誤，請新增至少一個 HSM 到叢集。如果您新增多個 HSM，最好在不同的可用區域建立它們。

- INSUFFICIENT_FREE_ADDRESSES_IN_SUBNET 表示 AWS KMS 無法將 AWS CloudHSM 金鑰存放區連接至其 AWS CloudHSM 叢集，因為至少有一個與叢集相關聯的私有子網路沒有任何可用的 IP 地址。AWS CloudHSM 金鑰存放區連接需要每個相關聯的私有子網路中有一個可用的 IP 地址，但最好有兩個地址。

  您無法新增 IP 地址 (CIDR 區塊) 至現有的子網路。如果可能，請移動或刪除子網路中使用 IP 地址的其他資源，例如未使用的 EC2 執行個體或彈性網路介面。否則，您可從 AWS CloudHSM 叢集的最近備份建立叢集以及具有更多可用地址空間的新的或現有的私有子網路。然後，若要將新叢集與 AWS CloudHSM 金鑰存放區產生關聯，請中斷連接自訂金鑰存放區、將 AWS CloudHSM 金鑰存放區的叢集 ID 變更為新叢集的 ID，然後再次嘗試連接。

  > **Tip**
  >
  > 若要避免 kmsuser 密碼的重設，請使用 AWS CloudHSM 叢集的最新備份。

- INTERNAL_ERROR 表示因為內部錯誤，AWS KMS 無法完成請求。重試 請求。若為 ConnectCustomKeyStore 請求，請先中斷連接 AWS CloudHSM 金鑰存放區，再重試連接。
- INVALID_CREDENTIALS 指出 AWS KMS 無法登入相關聯的 AWS CloudHSM 叢集，因為它沒有正確的 kmsuser 帳戶密碼。如需此錯誤的協助，請參閱如何修正無效的 kmsuser 登入資料。
- NETWORK_ERRORS 通常指出暫時性的網路問題。中斷連接 AWS CloudHSM 金鑰存放區，等待幾分鐘，並嘗試重新連接。
- SUBNET_NOT_FOUND 表示 AWS CloudHSM 叢集組態中至少一個子網路已遭刪除。如果 AWS KMS 在叢集組態中找不到所有子網路，則嘗試將 AWS CloudHSM 金鑰存放區連接至 AWS CloudHSM 叢集會失敗。

  若要修正此錯誤，請從相同 AWS CloudHSM 叢集的最近備份建立叢集。(此程序會建立具有 VPC 和私有子網路的新叢集組態)。請確認新叢集符合自訂金鑰存放區的需求，並記下新的叢集 ID。然後，若要將新叢集與 AWS CloudHSM 金鑰存放區產生關聯，請中斷連接自訂金鑰存放區、將 AWS CloudHSM 金鑰存放區的叢集 ID 變更為新叢集的 ID，然後再次嘗試連接。

  > **Tip**
  >
  > 若要避免 kmsuser 密碼的重設，請使用 AWS CloudHSM 叢集的最新備份。

- USER_LOCKED_OUT 指出 kmsuser 加密使用者 (CU) 帳戶因為有太多失敗的密碼嘗試，已鎖定在相關聯的 AWS CloudHSM 叢集之外。如需此錯誤的協助，請參閱如何修正無效的 kmsuser 登入資料。

  若要修正此錯誤，請中斷連接 AWS CloudHSM 金鑰存放區並使用 cloudhsm_mgmt_util 中的 changePswd 命令來變更 kmsuser 帳戶密碼。然後編輯自訂金鑰存放區的kmsuser 密碼設定，並嘗試重新連接。如需協助，請使用如何修正無效的 kmsuser 登入資料主題中描述的程序。
- USER_LOGGED_IN 表示 kmsuser CU 帳戶已登入相關聯的 AWS CloudHSM 叢集。這樣可防止 AWS KMS 輪換 kmsuser 帳戶密碼以及登入叢集。若要修正這個錯誤，請將 kmsuser CU 登出叢集。如果您變更登入叢集的 kmsuser 密碼，則您也必須更新 AWS CloudHSM 金鑰存放區的金鑰存放區密碼值。如需協助，請參閱 如何登出和重新連線。
- USER_NOT_FOUND 表示 AWS KMS 在相關聯的 AWS CloudHSM 叢集中找不到 kmsuser CU 帳戶。若要修正此錯誤，請在叢集中建立 kmsuser CU 帳戶，然後針對 AWS CloudHSM 金鑰存放區更新金鑰存放區密碼值。如需協助，請參閱 如何修正無效的 kmsuser 登入資料。

如何回應密碼編譯操作失敗

在自訂金鑰存放區採用 KMS 金鑰的密碼編譯操作可能失敗，並顯示 KMSInvalidStateException。下列錯誤訊息可能會伴隨 KMSInvalidStateException。

KMS 無法與 CloudHSM 叢集通訊。這可能是暫時性的網路問題。如您重複看到此錯誤，請確認 AWS CloudHSM 叢集 VPC 的網路 ACL 與安全群組規則是否正確。

- 雖然這是 HTTPS 400 錯誤，但可能是因為暫時性網路問題所造成的。若要回應，請先重試請求。不過，如果繼續失敗，請檢查聯網元件的組態。此錯誤很可能是因為聯網元件設定錯誤所造成，例如防火牆規則或 VPC 安全群組規則封鎖傳出流量。

由於 kmsuser 已遭到鎖定，因此 KMS 無法與您的 AWS CloudHSM 叢集通訊。如您重複看到此錯誤，請中斷連接 AWS CloudHSM 金鑰存放區並重設 kmsuser 帳戶密碼。更新自訂金鑰存放區的 kmsuser 密碼並重試請求。

- 此錯誤訊息指出 kmsuser加密使用者 (CU) 帳戶因密碼嘗試多次失敗，已鎖定在關聯的 AWS CloudHSM 叢集之外。如需此錯誤的協助，請參閱如何中斷連線和登入。

如何修正無效的 **kmsuser** 登入資料

當您連接 AWS CloudHSM 金鑰存放區時，AWS KMS 會作為 kmsuser 加密使用者 (CU) 登入相關聯的 AWS CloudHSM 叢集。它會保持登入，直到 AWS CloudHSM 金鑰存放區中斷連接為止。DescribeCustomKeyStores 回應顯示 FAILED 的 ConnectionState 和 INVALID_CREDENTIALS 的 ConnectionErrorCode 值，如以下範例所示。

如果您中斷連接 AWS CloudHSM 金鑰存放區並變更 kmsuser 密碼，AWS KMS 即無法使用 kmsuser CU 帳戶的憑證來登入 AWS CloudHSM 叢集。因此，連接 AWS CloudHSM 金鑰存放區的所有嘗試會失敗。DescribeCustomKeyStores 回應顯示 FAILED 的 ConnectionState 和 INVALID_CREDENTIALS 的 ConnectionErrorCode 值，如以下範例所示。

```
$ aws kms describe-custom-key-stores --custom-key-store-name ExampleKeyStore
{
   "CustomKeyStores": [
      "CloudHsmClusterId": "cluster-1a23b4cdefg",
      "ConnectionErrorCode": "INVALID_CREDENTIALS"
      "CustomKeyStoreId": "cks-1234567890abcdef0",
      "CustomKeyStoreName": "ExampleKeyStore",
      "TrustAnchorCertificate": "<certificate string appears here>",
      "CreationDate": "1.499288695918E9",
      "ConnectionState": "FAILED"
   ],
}
```

此外，在以不正確的密碼嘗試登入叢集失敗五次之後，AWS CloudHSM 會鎖定該使用者帳戶。若要登入叢集，您必須變更帳戶的密碼。

如果 AWS KMS 在嘗試以 kmsuser CU 身分登入叢集時遇到鎖定回應，請求連接 AWS CloudHSM 金鑰存放區將會失敗。回DescribeCustomKeyStores應包含ConnectionState的FAILED和ConnectionErrorCode值USER_LOCKED_OUT，如下列範例所示。

```
$ aws kms describe-custom-key-stores --custom-key-store-name ExampleKeyStore
{
   "CustomKeyStores": [
      "CloudHsmClusterId": "cluster-1a23b4cdefg",
      "ConnectionErrorCode": "USER_LOCKED_OUT"
      "CustomKeyStoreId": "cks-1234567890abcdef0",
      "CustomKeyStoreName": "ExampleKeyStore",
      "TrustAnchorCertificate": "<certificate string appears here>",
```

```
      "CreationDate": "1.499288695918E9",
      "ConnectionState": "FAILED"
   ],
}
```

若要修正任一個情況，請使用下列程序。

1. 中斷連接 AWS CloudHSM 金鑰存放區。
2. 執行DescribeCustomKeyStores作業並檢視回應中ConnectionErrorCode元素的值。
   - 如果 ConnectionErrorCode 值為 INVALID_CREDENTIALS，請判斷 kmsuser 帳戶目前的密碼。如果需要，請使用 cloudhsm_mgmt_util 中的 changePswd 命令，將密碼設定為已知的值。
   - 如果 ConnectionErrorCode 值為 USER_LOCKED_OUT，您必須使用 cloudhsm_mgmt_util 中的 changePswd 命令來變更 kmsuser 密碼。
3. 編輯 kmsuser 密碼設定，讓它符合叢集中目前的 kmsuser 密碼。此動作可告知 AWS KMS 要使用哪個密碼來登入叢集。它不會變更叢集中的 kmsuser 密碼。
4. 連接自訂金鑰存放區。

如何刪除遺棄的金鑰材料

排程從 AWS CloudHSM 金鑰存放區刪除 KMS 金鑰之後，您可能需要手動從相關聯的 AWS CloudHSM 叢集刪除對應的金鑰材料。

在 AWS CloudHSM 金鑰存放區中建立 KMS 金鑰時，AWS KMS 會在 AWS KMS 中建立 KMS 金鑰中繼資料，並在相關聯的 AWS CloudHSM 叢集中產生金鑰材料。排程從 AWS CloudHSM 金鑰存放區刪除 KMS 金鑰時，在等待期間之後，AWS KMS 會刪除 KMS 金鑰中繼資料。然後，AWS KMS 會盡可能從 AWS CloudHSM 叢集刪除金鑰資料。如果 AWS KMS 無法存取叢集，則嘗試可能會失敗，例如當叢集與 AWS CloudHSM 金鑰存放區中斷連接時，或 kmsuser 密碼變更時。AWS KMS 不會嘗試從叢集備份中刪除資料。

AWS KMS 會報告其嘗試從 AWS CloudTrail 日誌的 DeleteKey 事件項目的叢集中刪除金鑰資料的結果。它會出現在 additionalEventData 元素的 backingKeysDeletionStatus 元素中，如以下範例項目所示。此項目還包含 KMS 金鑰 ARN、AWS CloudHSM 叢集 ID，以及金鑰資料的金鑰控制代碼 (backing-key-id)。

```
{
    "eventVersion": "1.08",
```

```
    "userIdentity": {
        "accountId": "111122223333",
        "invokedBy": "AWS Internal"
    },
    "eventTime": "2021-12-10T14:23:51Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DeleteKey",
    "awsRegion": "eu-west-1",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": null,
    "responseElements": null,
    "additionalEventData": {
        "customKeyStoreId": "cks-1234567890abcdef0",
        "clusterId": "cluster-1a23b4cdefg",
        "backingKeys": "[{\"keyHandle\":\"01\",\"backingKeyId\":\"backing-key-id\"}]",
        "backingKeysDeletionStatus": "[{\"keyHandle\":\"16\",\"backingKeyId\":
\"backing-key-id\",\"deletionStatus\":\"FAILURE\"}]"
    },
    "eventID": "c21f1f47-f52b-4ffe-bff0-6d994403cf40",
    "readOnly": false,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:eu-
west-1:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsServiceEvent",
    "recipientAccountId": "111122223333",
    "managementEvent": true,
    "eventCategory": "Management"
}
```


若要從相關聯的 AWS CloudHSM 叢集刪除金鑰材料，請使用類似以下的程序。此範例使用 AWS CLI 和 AWS CloudHSM 命令列工具，但您也可以使用 AWS Management Console 而非 CLI。

1. 中斷連接 AWS CloudHSM 金鑰存放區 (如果尚未中斷連接)，然後登入 key_mgmt_util，如 如何中斷連線和登入 中所述。
2. 使用 key_mgmt_util 中的 deleteKey 命令，從叢集中的 HSM 刪除金鑰。

例如，此命令會從叢集中的 HSM 刪除金鑰 262162。金鑰控制代碼會列在記 CloudTrail 錄項目中。

```
Command: deleteKey -k 262162

        Cfm3DeleteKey returned: 0x00 : HSM Return: SUCCESS

        Cluster Error Status
        Node id 0 and err state 0x00000000 : HSM Return: SUCCESS
        Node id 1 and err state 0x00000000 : HSM Return: SUCCESS
        Node id 2 and err state 0x00000000 : HSM Return: SUCCESS
```

3. 登出 key_mgmt_util，並如 如何登出和重新連線 中所述重新連接 AWS CloudHSM 金鑰存放區。

如何復原 KMS 金鑰已刪除的金鑰材料

如果已刪除 AWS KMS key 的金鑰材料，則該 KMS 金鑰會無法使用，並且使用該 KMS 金鑰加密的所有加密文字會無法解密。如果已從相關聯的 AWS CloudHSM 叢集刪除 AWS CloudHSM 金鑰存放區中 KMS 金鑰的金鑰材料，則可能發生此情況。不過，復原金鑰材料可能可行。

在 AWS CloudHSM 金鑰存放區中建立 AWS KMS key (KMS 金鑰) 時，AWS KMS 會記錄至相關聯的 AWS CloudHSM 叢集並為 KMS 金鑰建立金鑰材料。還會將密碼變更為只有它才知道的某個值，並且只要 AWS CloudHSM 金鑰存放區保持連接便保持登入。由於只有金鑰擁有者 (也就是建立金鑰的 CU) 可以刪除金鑰，幾乎不可能會從 HSM 中不慎刪除金鑰。

不過，如果某個 KMS 金鑰的金鑰材料已從叢集中的 HSM 刪除，KMS 金鑰之金鑰狀態最終會變更為 UNAVAILABLE。如果您嘗試將該 KMS 金鑰用於密碼編譯操作，則該操作會失敗，出現 `KMSInvalidStateException` 例外狀況。最重要的是，使用該 KMS 金鑰加密的任何資料會無法解密。

在某些情況下，您可以透過從包含金鑰材料的備份建立叢集來復原已刪除的金鑰材料。此策略僅在當該金鑰存在時並且在刪除它之前建立了至少一個備份的情況下才有用。

使用以下程序來復原金鑰材料。

1. 尋找包含金鑰材料的叢集備份。該備份也必須包含您需要支援叢集及其加密資料的所有使用者和金鑰。

   使用此DescribeBackups作業列出叢集的備份。然後使用備份時間戳記來協助您選取備份。若要限制輸出到與 AWS CloudHSM 金鑰存放區相關聯的叢集，請使用 `Filters` 參數，如下列範例所示。

```
$ aws cloudhsmv2 describe-backups --filters clusterIds=<cluster ID>
{
    "Backups": [
        {
            "ClusterId": "cluster-1a23b4cdefg",
            "BackupId": "backup-9g87f6edcba",
            "CreateTimestamp": 1536667238.328,
            "BackupState": "READY"
        },
             ...
    ]
}
```

2. 從選取的備份建立叢集。驗證備份包含已刪除的金鑰和叢集所需的其他使用者和金鑰。
3. 中斷連接 AWS CloudHSM 金鑰存放區，使得您可以編輯其屬性。
4. 編輯 AWS CloudHSM 金鑰存放區的叢集 ID。輸入您從備份建立之叢集的叢集 ID。由於該叢集與原始叢集共用備份歷史記錄，新叢集 ID 應該是有效的。
5. 重新連接 AWS CloudHSM 金鑰存放區。

如何以 **kmsuser** 身分登入

為了在您的 AWS CloudHSM 金鑰存放區的 AWS CloudHSM 叢集中建立和管理金鑰材料，AWS KMS 會使用 kmsuser 加密使用者 (CU) 帳戶。您會在您的叢集中建立 kmsuser CU 帳戶，並在建立您的 AWS CloudHSM 金鑰存放區時提供其密碼給 AWS KMS。

一般而言，AWS KMS 會管理 kmsuser 帳戶。不過，對於某些任務，您需要中斷連接 AWS CloudHSM 金鑰存放區，請以 kmsuser CU 身分登入叢集，並使用 cloudhsm_mgmt_util 和 key_mgmt_util 命令列工具。

> **Note**
>
> 當自訂金鑰存放區中斷連接時，所有在自訂金鑰存放區中建立 KMS 金鑰的嘗試，或在密碼編譯操作中使用現有 KMS 金鑰的嘗試，均會失敗。此動作可防止使用者存放和存取敏感資料。

此主題說明如何中斷連接您的 AWS CloudHSM 金鑰存放區並以 kmsuser 身分登入、執行 AWS CloudHSM 命令列工具，以及登出並重新連接您的 AWS CloudHSM 金鑰存放區。

主題

- 如何中斷連線和登入
- 如何登出和重新連線

如何中斷連線和登入

每當需要以 kmsuser CU 身分登入相關聯的叢集時，請使用以下程序。

1. 中斷連接 AWS CloudHSM 金鑰存放區 (如果尚未中斷連接)。您可使用 AWS KMS 主控台或 AWS KMS API。

   在您的 AWS CloudHSM 金鑰連接時，AWS KMS 會以 kmsuser 的身分登入。如此會防止您以 kmsuser 身分登入或變更 kmsuser 密碼。

   例如，此命令用DisconnectCustomKeyStore於中斷範例金鑰存放區的連線。將範例 AWS CloudHSM 金鑰存放區 ID 取代為有效 ID。

   ```
   $ aws kms disconnect-custom-key-store --custom-key-store-id cks-1234567890abcdef0
   ```

2. 啟動 cloudhsm_mgmt_util。使用《AWS CloudHSM 使用者指南》的準備執行 cloudhsm_mgmt_util 章節所述的處理程序。
3. 作為加密管理員 (CO) 登入 AWS CloudHSM 叢集上的 cloudhsm_mgmt_util。

   例如，此命令會以名為 admin 的 CO 身分登入。以有效的值取代範例 CO 使用者名稱和密碼。

   ```
   aws-cloudhsm>loginHSM CO admin <password>
   loginHSM success on server 0(10.0.2.9)
   loginHSM success on server 1(10.0.3.11)
   loginHSM success on server 2(10.0.1.12)
   ```

4. 使用 changePswd 命令，將 kmsuser 帳戶的密碼變更為您知道的密碼。(AWS KMS 會在您連接 AWS CloudHSM 金鑰存放區時輪換密碼。) 密碼必須包含 7 到 32 個英數字元。區分大小寫，且不能包含任何特殊字元。

   例如，此命令會將 kmsuser 密碼變更為 tempPassword。

   ```
   aws-cloudhsm>changePswd CU kmsuser tempPassword

   ************************CAUTION*********************************
   This is a CRITICAL operation, should be done on all nodes in the
   cluster. Cav server does NOT synchronize these changes with the
   ```

```
nodes on which this operation is not executed or failed, please
ensure this operation is executed on all nodes in the cluster.
*****************************************************************

Do you want to continue(y/n)?y
Changing password for kmsuser(CU) on 3 nodes
```

5. 以 kmsuser 的身分並使用您設定的密碼登入 key_mgmt_util 或 cloudhsm_mgmt_util。如需詳細的指示，請參閱cloudhsm_mgmt_util 入門和 key_mgmt_util 入門。您使用的工具取決於您的任務。

   例如，此命令會登入 key_mgmt_util。

   ```
   Command: loginHSM -u CU -s kmsuser -p tempPassword
   Cfm3LoginHSM returned: 0x00 : HSM Return: SUCCESS

   Cluster Error Status
   Node id 0 and err state 0x00000000 : HSM Return: SUCCESS
   Node id 1 and err state 0x00000000 : HSM Return: SUCCESS
   Node id 2 and err state 0x00000000 : HSM Return: SUCCESS
   ```

如何登出和重新連線

1. 執行任務，然後登出命令列工具。如果您不登出，重新連接您的 AWS CloudHSM 金鑰存放區的嘗試將會失敗。

   ```
   Command:  logoutHSM
   Cfm3LogoutHSM returned: 0x00 : HSM Return: SUCCESS

   Cluster Error Status
   Node id 0 and err state 0x00000000 : HSM Return: SUCCESS
   Node id 1 and err state 0x00000000 : HSM Return: SUCCESS
   ```

2. 編輯自訂金鑰存放區的 kmsuser 密碼設定。

   這會告知 AWS KMS 叢集中 kmsuser 的目前密碼。如果您省略此步驟，AWS KMS 將無法以 kmsuser 身分登入叢集，而且重新連接您的自訂金鑰存放區的所有嘗試將會失敗。您可以使用 AWS KMS控制台或UpdateCustomKeyStore操作的KeyStorePassword參數。

   例如，此命令會告知 AWS KMS 目前的密碼為 tempPassword。將範例密碼以實際密碼取代。

```
$ aws kms update-custom-key-store --custom-key-store-id cks-1234567890abcdef0 --
key-store-password tempPassword
```

3. 將 AWS KMS 金鑰存放區重新連接至其 AWS CloudHSM 叢集。將範例 AWS CloudHSM 金鑰存放區 ID 取代為有效 ID。連接程序期間，AWS KMS 會將 kmsuser 密碼變更為只有它知道的值。

   ConnectCustomKeyStore作業會快速傳回，但連線程序可能需要較長的時間。初始回應並不表示連接程序已成功。

   ```
   $ aws kms connect-custom-key-store --custom-key-store-id cks-1234567890abcdef0
   ```

4. 使用此DescribeCustomKeyStores作業驗證AWS CloudHSM金鑰存放區是否已連線。將範例 AWS CloudHSM 金鑰存放區 ID 取代為有效 ID。

   在此範例中，連接狀態欄位顯示現在已連接 AWS CloudHSM 金鑰存放區。

   ```
   $ aws kms describe-custom-key-stores --custom-key-store-
   id cks-1234567890abcdef0
   {
      "CustomKeyStores": [
         "CustomKeyStoreId": "cks-1234567890abcdef0",
         "CustomKeyStoreName": "ExampleKeyStore",
         "CloudHsmClusterId": "cluster-1a23b4cdefg",
         "TrustAnchorCertificate": "<certificate string appears here>",
         "CreationDate": "1.499288695918E9",
         "ConnectionState": "CONNECTED"
      ],
   }
   ```

外部金鑰存放區

外部金鑰存放區可讓您使用 AWS 之外的密碼編譯金鑰來保護 AWS 資源。此進階功能專為受管制的工作負載而設計，您必須使用儲存在您控制的外部金鑰管理系統中的加密金鑰來保護這些工作負載。外部金鑰存放區支援 AWS 數位主權承諾，讓您對 AWS 中的資料擁有主權控制權，包括能夠使用您在 AWS 之外擁有和控制的金鑰材料進行加密。

外部金鑰存放區是由您在 AWS 之外擁有和控制的外部金鑰管理器所支援的自訂金鑰存放區。您的外部金鑰管理器可以是實體或虛擬硬體安全模組 (HSM)，也可以是任何能夠產生和使用密碼編譯金鑰的硬

體型或軟體型系統。在外部金鑰存放區中使用 KMS 金鑰的加密和解密操作由外部金鑰管理器使用您的密碼編譯金鑰材料執行，這項功能稱為 Hold Your Own Keys (HYOK)。

AWS KMS 從不直接與外部金鑰管理器互動，也無法建立、檢視、管理或刪除您的金鑰。相反，AWS KMS 只會與您提供的外部金鑰存放區代理 (XKS 代理) 軟體互動。您的外部金鑰存放區代理會協調 AWS KMS 與外部金鑰管理器之間的所有通訊。它會將所有請求從 AWS KMS 傳送至外部金鑰管理器，並將外部金鑰管理器的回應傳送回 AWS KMS。外部金鑰存放區代理也會將來自 AWS KMS 的一般請求轉譯成外部金鑰管理器可以理解的特定於廠商的格式，讓您可以搭配使用外部金鑰存放區和各種廠商的金鑰管理器。

您可以在外部金鑰存放區中使用 KMS 金鑰進行用戶端加密，包括 AWS Encryption SDK。但是，外部金鑰存放區是伺服器端加密的重要資源，可讓您使用 AWS 外部的密碼編譯金鑰來保護多個 AWS 服務中的 AWS 資源。支援客戶受管金鑰以進行對稱加密的 AWS 服務 也支援外部金鑰存放區中的 KMS 金鑰。如需服務支援詳細資訊，請參閱 AWS 服務整合。

外部金鑰存放區允許您將 AWS KMS 用於受管制的工作負載，其中加密金鑰必須在 AWS 之外儲存和使用。但其與標準共同責任模式有很大不同，並且需要額外的運營負擔。對於大多數客戶而言，可用性和延遲的風險更大，將超過外部金鑰存放區的安全優勢。

外部金鑰存放區可讓您控制信任的根源。只有使用您控制的外部金鑰管理器，才能解密使用外部金鑰存放區中 KMS 金鑰加密的資料。如果您暫時撤銷對外部金鑰管理器的存取權，例如中斷連接外部金鑰存放區或中斷連接外部金鑰管理器與外部金鑰存放區代理，則 AWS 會失去對密碼編譯金鑰的所有存取權，直到您將其還原為止。在該間隔期間，不能解密使用 KMS 金鑰加密的密文。如果您永久撤銷對外部金鑰管理器的存取權，則使用外部金鑰存放區中的 KMS 金鑰加密的所有密文都將無法復原。唯一的例外情況是 AWS 服務，其會短暫快取受 KMS 金鑰保護的資料金鑰。這些資料金鑰會繼續運作，直到您停用資源或快取到期為止。如需詳細資訊，請參閱 無法使用的 KMS 金鑰如何影響資料金鑰。

外部金鑰存放區會針對受管制的工作負載解鎖少數使用案例，其中加密金鑰必須完全由您控制且無法由 AWS 存取。但是，這是雲端基礎架構操作方式的重大變化，也是共同責任模式的明顯轉變。對於大多數工作負載而言，額外的操作負擔以及可用性和效能的更大風險將超過外部金鑰存放區感知的安全優勢。

進一步了解：

- 在 AWS 新聞部落格中宣告 AWS KMS 外部金鑰存放區。

我需要外部金鑰存放區嗎？

對於多數使用者，預設 AWS KMS 金鑰存放區 (受到 FIPS 140-2 安全層級 3 驗證的硬體安全模組保護) 可滿足其安全性、控制及法規需求。外部金鑰存放區使用者會產生巨大的成本、維護和疑難排解負擔，以及延遲、可用性和可靠性風險。

考量外部金鑰存放區時，請花一些時間了解替代方案，包括您擁有和管理的 AWS CloudHSM 叢集支援的 AWS CloudHSM 金鑰存放區，以及在您自己的 HSM 中產生且可視需要從 KMS 金鑰中刪除的具有匯入金鑰材料的 KMS 金鑰。特別是，匯入有效期非常短的金鑰材料可能會提供類似的控制層級，而不會造成效能或可用性風險。

如果您有下列需求，外部金鑰存放區可能是您組織的正確解決方案：

- 您必須在內部部署金鑰管理器或您控制的 AWS 之外的金鑰管理器中使用密碼編譯金鑰。
- 必須證明您的密碼編譯金鑰在雲端之外完全由您控制。
- 您必須使用具有獨立授權的密碼編譯金鑰來進行加密和解密。
- 金鑰資料必須放在次要、獨立的稽核路徑。

如果您選擇外部金鑰存放區，則請將其使用限制於需要利用 AWS 之外的密碼編譯金鑰來保護的工作負載。

共同的責任模型

標準 KMS 金鑰使用在 AWS KMS 擁有和管理的 HSM 中產生和使用的金鑰材料。您可以在 KMS 金鑰上建立存取控制政策，並設定使用 KMS 金鑰來保護資源的 AWS 服務。AWS KMS 會承擔 KMS 金鑰中金鑰材料的安全性、可用性、延遲和耐久性責任。

外部金鑰存放區中的 KMS 金鑰依賴於外部金鑰管理器中的金鑰材料和操作。因此，責任的天平會朝著您的方向移動。您必須對外部金鑰管理器中密碼編譯金鑰的安全性、可靠性、耐久性和效能負責。AWS KMS 負責迅速回應請求並與外部金鑰存放區代理進行通訊，以及維護我們的安全標準。為了確保每個外部金鑰存放區密文至少和標準 AWS KMS 密文一樣強，AWS KMS 首先使用特定於 KMS 金鑰的 AWS KMS 金鑰材料來加密所有純文字，然後將其傳送至外部金鑰管理器，以便使用外部金鑰進行加密，這一過程稱為雙重加密。因此，無論是 AWS KMS 還是外部金鑰材料擁有者都不能單獨解密雙重加密的密文。

您有責任維護符合法規與效能標準的外部金鑰管理器、供應及維護符合AWS KMS外部金鑰存放區代理 API 規格的外部金鑰存放區代理，並確保金鑰資料的可用性與持久性。此外，您也必須建立、設定和維護外部金鑰存放區。當您維護的元件造成錯誤發生時，您必須準備好識別並解決錯誤，以便 AWS 服務

能夠存取您的資源，而不會造成不必要的中斷。AWS KMS 提供故障診斷指南，協助您確定問題的原因和最可能的解決方案。

查看記AWS KMS錄外部金鑰存放區的 Amazon CloudWatch 指標和維度。 AWS KMS強烈建議您建立 CloudWatch 警示來監控外部金鑰存放區，以便在效能和作業問題發生之前，早期偵測到效能和作業問題的跡象。

有什麼變化？

外部金鑰存放區僅支援對稱加密 KMS 金鑰。在 AWS KMS 中，您在外部金鑰存放區中使用和管理 KMS 金鑰的方法與管理其他客戶受管金鑰大致相同，包括設定存取控制政策和監控金鑰用途。您可以使用具有相同參數的相同 API，在用於任何 KMS 金鑰的外部金鑰存放區中使用 KMS 金鑰來請求密碼編譯操作。定價也與標準 KMS 金鑰相同。如需詳細資訊，請參閱 管理外部金鑰存放區中的 KMS 金鑰、使用外部金鑰存放區中的 KMS 金鑰 和 AWS Key Management Service 定價。

但是，對於外部金鑰存放區，以下原則發生了變化：

- 您負責金鑰操作的可用性、耐久性和延遲。
- 您負責開發、購買、操作和授權外部金鑰管理器系統的所有費用。
- 您可以實作從 AWS KMS 到外部金鑰存放區代理之所有請求的獨立授權。
- 您可以監控、稽核和記錄外部金鑰存放區代理的所有操作，以及與 AWS KMS 請求相關的外部金鑰管理器的所有操作。

從何處開始？

若要建立和管理外部金鑰存放區，您需要選擇外部金鑰存放區代理連接選項、備妥先決條件以及建立和設定外部金鑰存放區。若要開始，請參閱 規劃外部金鑰存放區。

配額

AWS KMS 在每個 AWS 帳戶 和區域最多允許 10 個自訂金鑰存放區，包括 AWS CloudHSM 金鑰存放區和外部金鑰存放區，無論其連接狀態為何。此外，在外部金鑰存放區中使用 KMS 金鑰有 AWS KMS 請求配額。

如果您為外部金鑰存放區代理選擇 VPC 代理連接，則必要元件 (例如 VPC、子網路和網路負載平衡器) 可能也會有配額。如需這些配額的相關資訊，請使用 Service Quotas 主控台。

區域

若要將網路延遲降到最低，請在離外部金鑰管理器最近的 AWS 區域 中建立外部金鑰存放區元件。如果可能，請選擇網路封包來回時間 (RTT) 為 35 毫秒或更短的區域。

在支援 AWS KMS 的所有 AWS 區域，外部金鑰存放區均受支援，除了中國 (北京) 與中國 (寧夏) 外。

不支援的功能

AWS KMS 不支援自訂金鑰存放區中的以下功能。

- 非對稱 KMS 金鑰
- 非對稱資料金鑰對
- HMAC KMS 金鑰
- 含有匯入金鑰資料的 KMS 金鑰
- 自動金鑰輪換
- 多區域金鑰

主題

- 外部金鑰存放區概念
- 外部金鑰存放區的運作方式
- 控制對外部金鑰存放區的存取
- 規劃外部金鑰存放區
- 管理外部金鑰存放區
- 管理外部金鑰存放區中的 KMS 金鑰
- 外部金鑰存放區故障診斷

外部金鑰存放區概念

此主題說明外部金鑰存放區中所使用的一些概念。

主題

- 外部金鑰存放區
- 外部金鑰管理器

- 外部金鑰
- 外部金鑰存放區代理
- 外部金鑰存放區代理連接
- 外部金鑰存放區代理身分驗證憑證
- 代理 API
- 雙重加密

外部金鑰存放區

外部金鑰存放區是由您擁有和管理的 AWS 之外的外部金鑰管理器所支援的 AWS KMS 自訂金鑰存放區。外部金鑰存放區中的每個 KMS 金鑰都與外部金鑰管理器中的外部金鑰相關聯。當您在外部金鑰存放區中使用 KMS 金鑰進行加密或解密時，外部金鑰管理器將使用外部金鑰執行操作，這種安排稱為 Hold your Own Keys (HYOK)。此功能專為需要在其外部金鑰管理器中維護密碼編譯金鑰的組織而設計。

外部金鑰存放區確保可保護 AWS 資源的密碼編譯金鑰和操作會保留在您控制的外部金鑰管理器中。AWS KMS 傳送請求給外部金鑰管理器以加密和解密資料，但 AWS KMS 無法建立、刪除或管理任何外部金鑰。從 AWS KMS 到外部金鑰管理器的所有請求都由您提供、擁有及管理的外部金鑰存放區代理軟體元件協調。

支援 AWS KMS 客戶受管金鑰的 AWS 服務可以使用外部金鑰存放區中的 KMS 金鑰來保護資料。因此，使用外部金鑰管理器中的加密操作，您的資料最終會受到金鑰保護。

與標準 KMS 金鑰相比，外部金鑰存放區中的 KMS 金鑰具有本質上不同的信任模型、共同責任安排和效能預期。透過外部金鑰存放區，您負責金鑰材料和密碼編譯操作的安全性和完整性。外部金鑰存放區中 KMS 金鑰的可用性和延遲會受到硬體、軟體、網路元件以及 AWS KMS 與外部金鑰管理器之間距離的影響。您也可能會為外部金鑰管理器以及外部金鑰管理器與 AWS KMS 通訊所需的聯網和負載平衡基礎設施產生額外費用。

您可以使用外部金鑰存放區作為更廣泛資料保護策略的一部分。對於您保護的每個 AWS 資源，您可以決定哪些需要外部金鑰存放區中的 KMS 金鑰，哪些可由標準 KMS 金鑰保護。這可讓您靈活地為特定資料分類、應用程式或專案選擇 KMS 金鑰。

外部金鑰管理器

外部金鑰管理器是 AWS 外部的元件，其可產生 256 位元 AES 對稱金鑰並執行對稱加密和解密。外部金鑰存放區的外部金鑰管理器可以是實體硬體安全模組 (HSM)、虛擬 HSM，或是含有或不含 HSM 元

件的軟體金鑰管理器。其可以位於 AWS 以外的任何地方，包括在您的內部部署、本機或遠端資料中心或任何雲端中。您的外部金鑰存放區可由單一外部金鑰管理器或多個共用密碼編譯金鑰的相關金鑰管理器執行個體 (例如 HSM 叢集) 進行支援。外部金鑰存放區的設計是為了支援來自不同廠商的各種外部管理器。如需有關外部金鑰管理器需求的詳細資訊，請參閱 規劃外部金鑰存放區。

外部金鑰

外部金鑰存放區中的每個 KMS 金鑰都與外部金鑰管理器中稱為外部金鑰的密碼編譯金鑰相關聯。當您在外部金鑰存放區中使用 KMS 金鑰進行加密或解密時，外部金鑰管理器將使用外部金鑰執行密碼編譯操作。

> **Warning**
>
> 外部金鑰對於 KMS 金鑰的操作至關重要。如果遺失或刪除外部金鑰，則使用關聯之 KMS 金鑰加密的密文將無法復原。

對於外部金鑰存放區，外部金鑰必須是已啟用且可執行加密和解密動作的 256 位元 AES 金鑰。如需外部金鑰需求的詳細資訊，請參閱 外部金鑰存放區中 KMS 金鑰的要求。

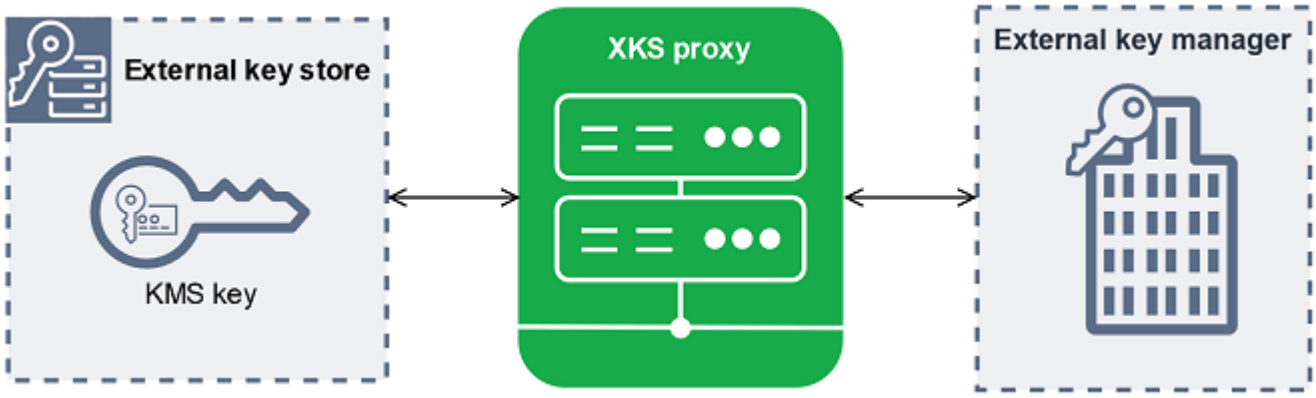
AWS KMS 無法建立、刪除或管理任何外部金鑰。您的密碼編譯金鑰材料永遠不會離開外部金鑰管理器。在外部金鑰存放區中建立 KMS 金鑰時，您需要提供外部金鑰的 ID (`XksKeyId`)。您無法變更與 KMS 金鑰相關聯的外部金鑰 ID，不過您的外部金鑰管理器可以輪換與外部金鑰 ID 相關聯的金鑰材料。

除了外部金鑰之外，外部金鑰存放區中的 KMS 金鑰也包含 AWS KMS 金鑰材料。受 KMS 金鑰保護的資料首先由 AWS KMS 使用 AWS KMS 金鑰材料進行加密，然後再由外部金鑰管理器使用外部金鑰加密。此雙重加密程序可確保受 KMS 金鑰保護的密文始終至少與僅受 AWS KMS 保護的密文一樣強。

許多密碼編譯金鑰具有不同類型的識別符。在外部金鑰存放區中建立 KMS 金鑰時，請提供外部金鑰存放區代理用來參照外部金鑰的外部金鑰 ID。如果使用錯誤的識別符，則您嘗試在外部金鑰存放區中建立 KMS 金鑰會失敗。

外部金鑰存放區代理

外部金鑰存放區代理 ("XKS proxy") 是客戶擁有且客戶管理的軟體應用程式，可協調 AWS KMS 與外部金鑰管理器之間的所有通訊。其還將一般 AWS KMS 請求轉換為特定於廠商的外部金鑰管理器能夠理解的格式。外部金鑰存放區需要外部金鑰存放區代理。每個外部金鑰存放區與一個外部金鑰存放區代理相關聯。



AWS KMS 無法建立、刪除或管理任何外部金鑰。您的密碼編譯金鑰材料永遠不會離開您的外部金鑰管理器。AWS KMS 與外部金鑰管理器之間的所有通訊都由外部金鑰存放區代理協調。AWS KMS 將請求傳送至外部金鑰存放區代理，並接收來自外部金鑰存放區代理的回應。外部金鑰存放區代理負責將請求從 AWS KMS 傳送至外部金鑰管理器，並將回應從外部金鑰管理器傳送回 AWS KMS。

您擁有並管理外部金鑰存放區的外部金鑰存放區代理，並負責其維護和操作。您可以根據 AWS KMS 發佈的開放原始碼外部金鑰存放區代理 API 規格來開發外部金鑰存放區代理，或購買廠商的代理應用程式。您的外部金鑰存放區代理可能包含在您的外部金鑰管理器中。為了支援 Proxy 開發，AWS KMS還提供範例外部金鑰存放區 Proxy (aws-kms-xks-proxy) 和測試用戶端 (xks-kms-xksproxy-test-client)，以驗證您的外部金鑰存放區 Proxy 是否符合規格。

若要對 AWS KMS 進行身分驗證，代理會使用伺服器端 TLS 憑證。若要對代理進行身分驗證，AWS KMS 會使用 Sigv4 代理身分驗證憑證將所有請求簽署至外部金鑰存放區代理。或者，您的代理可以啟用交互 TLS (mTLS)，以進一步確保其僅接受來自 AWS KMS 的請求。

您的外部金鑰存放區代理必須支援 HTTP/1.1 或更新版本，以及 TLS 1.2 或更新版本，且至少包含下列其中一個加密套件：

- TLS_AES_256_GCM_SHA384 (TLS 1.3)
- TLS_CHACHA20_POLY1305_SHA256 (TLS 1.3)

> **Note**
>
> AWS GovCloud (US) Region 不支援 TLS_CHACHA20_POLY1305_SHA256。

- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (TLS 1.2)
- TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 (TLS 1.2)

若要在外部金鑰存放區中建立和使用 KMS 金鑰，您必須首先將外部金鑰存放區連接至其外部金鑰存放區代理。您也可以視需求中斷外部金鑰存放區與其代理的連接。當您這麼做時，外部金鑰存放區中的所有 KMS 金鑰都變得無法使用；其無法用於任何加密操作。

外部金鑰存放區代理連接

外部金鑰存放區代理連接 (「XKS 代理連接」) 描述了 AWS KMS 用來與外部金鑰存放區代理通訊的方法。

您可以在建立外部金鑰存放區時指定代理連接選項，其會成為外部金鑰存放區的屬性。您可以透過更新自訂金鑰存放區屬性來變更代理連接選項，但您必須確定外部金鑰存放區代理仍可存取相同的外部金鑰。

AWS KMS 支援以下連接選項：

- 公有端點連接 – AWS KMS 透過網際網路將外部金鑰存放區代理的請求傳送至您控制的公有端點。此選項的建立和維護非常簡單，但可能無法滿足每個安裝的安全需求。
- VPC 端點服務連接 – AWS KMS 將請求傳送至您建立和維護的 Amazon Virtual Private Cloud (Amazon VPC) 端點服務。您可以在 Amazon VPC 內託管外部金鑰存放區代理，或在 AWS 外部託管外部金鑰存放區代理，並僅將 Amazon VPC 用於通訊。

如需有關外部金鑰存放區代理連接選項的詳細資訊，請參閱 選擇代理連接選項。

外部金鑰存放區代理身分驗證憑證

若要對外部金鑰存放區代理進行身分驗證，AWS KMS 會使用 Signature V4 (SigV4) 身分驗證憑證將所有請求簽署至外部金鑰存放區代理。您可以在代理上建立並維護身分驗證憑證，然後在建立外部存放區時將此憑證提供給 AWS KMS。

> **Note**
>
> AWS KMS 用於向 XKS 代理簽署請求的 SigV4 憑證與 AWS 帳戶 中的 AWS Identity and Access Management 主體相關聯的任何 SigV4 憑證無關。請勿針對外部金鑰存放區代理重複使用任何 IAM SigV4 憑證。

每個代理身分驗證憑證有兩部分。建立外部金鑰存放區或更新外部金鑰存放區的身分驗證憑證時，您必須同時提供這兩個部分。

- 存取金鑰 ID：識別私密存取金鑰。您可以提供純文字形式的 ID。
- 私密存取金鑰：憑證的秘密部分。AWS KMS 在儲存憑證之前，先加密其中的私密存取金鑰。

您可以隨時編輯憑證設定，例如當您輸入錯誤的值時、變更代理上的憑證時或者代理輪換憑證時。如需有關外部金鑰存放區代理之 AWS KMS 身分驗證的技術詳細資訊，請參閱「AWS KMS 外部金鑰存放區代理 API 規格」中的身分驗證。

若要允許輪換憑證，而不會中斷使用外部金鑰存放區之 KMS 金鑰的 AWS 服務，建議外部金鑰存放區代理至少支援 AWS KMS 的兩個有效身分驗證憑證。這可確保當您為 AWS KMS 提供新憑證時，先前的憑證可繼續運作。

為了協助您追蹤代理身份驗證登入資料的使用期AWS KMS限，請定義 Amazon CloudWatch 指標XksProxyCredentialAge。您可以使用此指標建立 CloudWatch 警示，以在認證的年齡達到您建立的臨界值時通知您。

為了更加確保外部金鑰存放區代理僅回應 AWS KMS，某些外部金鑰代理支援交互式 Transport Layer Security (mTLS)。如需詳細資訊，請參閱 mTLS 身份驗證 (選用)。

代理 API

若要支援 AWS KMS 外部金鑰存放區，外部金鑰存放區代理必須依照 AWS KMS 外部金鑰存放區代理 API 規格中所述實作必要的代理 API。這些代理 API 請求是 AWS KMS 傳送給代理的僅有請求。雖然您永遠不會直接傳送這些請求，但了解其可能有助於您解決外部金鑰存放區或其代理可能出現的任何問題。例如，在外部金鑰存放區的 Amazon CloudWatch 指標中AWS KMS包含這些 API 呼叫的延遲和成功率相關資訊。如需詳細資訊，請參閱 監控外部金鑰存放區。

下表列出並說明每個代理 API。其還包括觸發對代理 API 呼叫的 AWS KMS 操作以及與代理 API 相關的任何 AWS KMS 操作例外狀況。

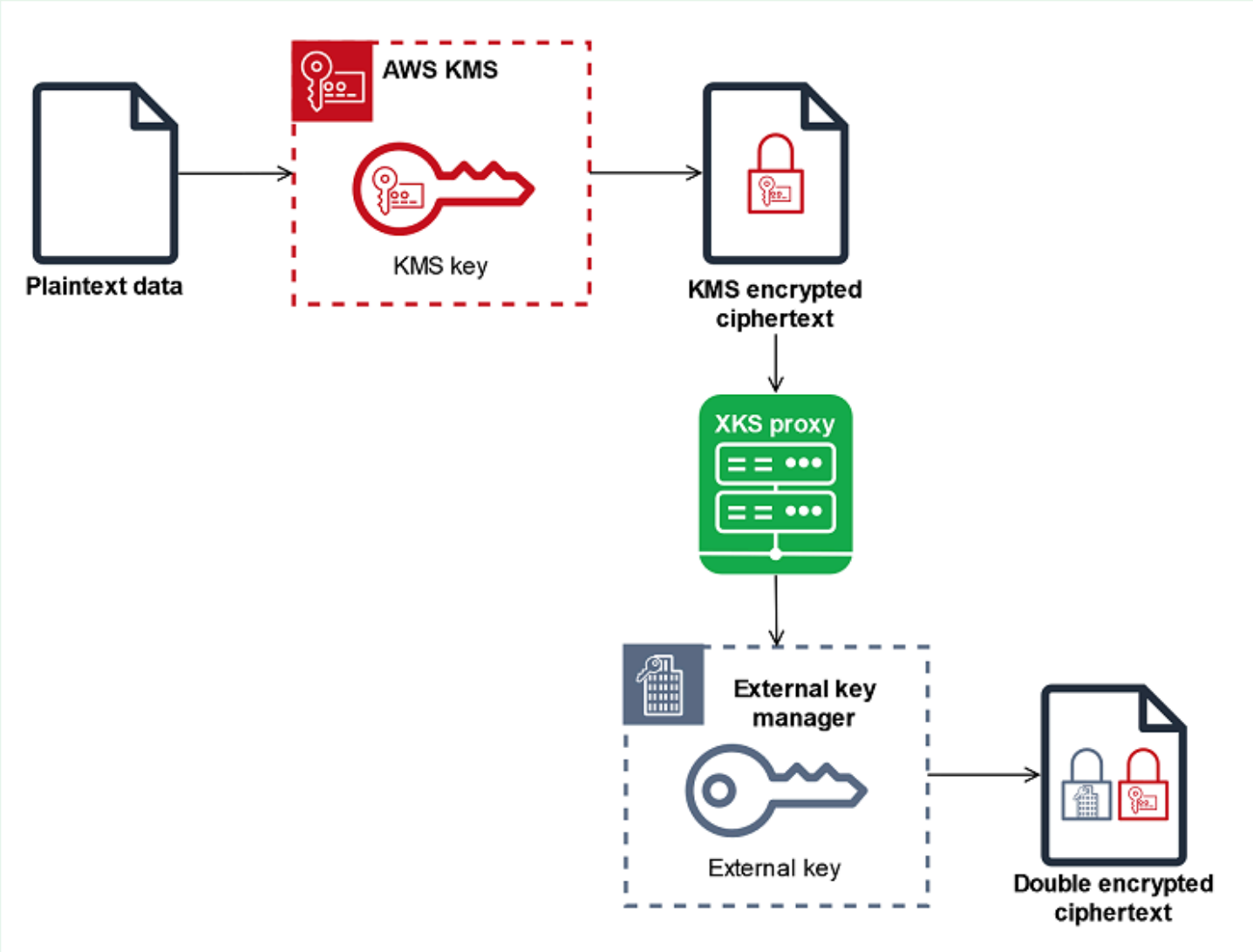
代理 API	描述	相關的 AWS KMS 操作
解密	AWS KMS 會傳送要解密的密文，以及要使用的外部金鑰的 ID。所需的加密演算法是 AES_GCM。	解密，ReEncrypt
加密	AWS KMS 會傳送要加密的資料，以及要使用的外部金鑰的 ID。所需的加密演算法是 AES_GCM。	加密、GenerateDataKey、GenerateDataKeyWithoutPlaintext、ReEncrypt

代理 API	描述	相關的 AWS KMS 操作
GetHealthStatus	有關代理和外部金鑰管理器狀態的 AWS KMS 請求資訊。 每個外部金鑰管理器的狀態可以為以下其中一種。 • Active：狀態良好；可以提供流量 • Degraded：狀態不良，但可提供流量 • Unavailable ：狀態不良；不能提供流量	CreateCustomKeyStore(適用於公用端點連線)，ConnectCustomKeyStore(適用於 VPC 端點服務連線) 如果所有外部金鑰管理器執行個體都是 Unavailable ，則嘗試建立或連接金鑰存放區會失敗，且出現 XksProxyUriUnreachableException 。
GetKeyMetadata	AWS KMS 會請求與外部金鑰存放區中 KMS 金鑰相關聯之外部金鑰的相關資訊。 回應包括金鑰規格 (AES_256)、金鑰用法 ([ENCRYPT, DECRYPT]) 以及外部金鑰是否為 ENABLED 或 DISABLED。	CreateKey 如果金鑰規格不是 AES_256，或金鑰用法不是 [ENCRYPT, DECRYPT]，或狀態為 DISABLED，則 CreateKey 操作會失敗，且會出現 XksKeyInvalidConfigurationException 。

雙重加密

透過外部金鑰存放區中 KMS 金鑰加密的資料會加密兩次。首先，AWS KMS 使用 KMS 金鑰特有的 AWS KMS 金鑰材料加密資料。然後，外部金鑰管理器使用外部金鑰對 AWS KMS 加密的密文進行加密。此過程稱為雙重加密。

雙重加密可確保由外部金鑰存放區中 KMS 金鑰加密的密文至少與使用標準 KMS 金鑰加密的密文一樣強。其還可以保護從 AWS KMS 傳輸到外部金鑰存放區代理的純文字。使用雙重加密，您可以完全控制您的密文。如果您透過外部代理永久撤銷對外部金鑰的 AWS 存取權，則 AWS 中剩餘的任何密文都會被有效地加密銷毀。



若要啟用雙重加密，外部金鑰存放區中的每個 KMS 金鑰都有兩個密碼編譯備份金鑰：

- KMS 金鑰特有的 AWS KMS 金鑰材料。僅可在 AWS KMS FIPS 140-2 安全層級 3 驗證的硬體安全模組 (HSM) 產生並運用此金鑰資料。
- 外部金鑰管理器中的外部金鑰。

雙重加密具有以下效果：

- AWS KMS 在沒有透過外部金鑰存放區代理來存取外部金鑰的情況下，無法解密由外部金鑰存放區中的 KMS 金鑰加密的任何密文。
- 您無法解密由 AWS 之外的外部金鑰存放區中的 KMS 金鑰加密的任何密文，即使您擁有其外部金鑰材料。

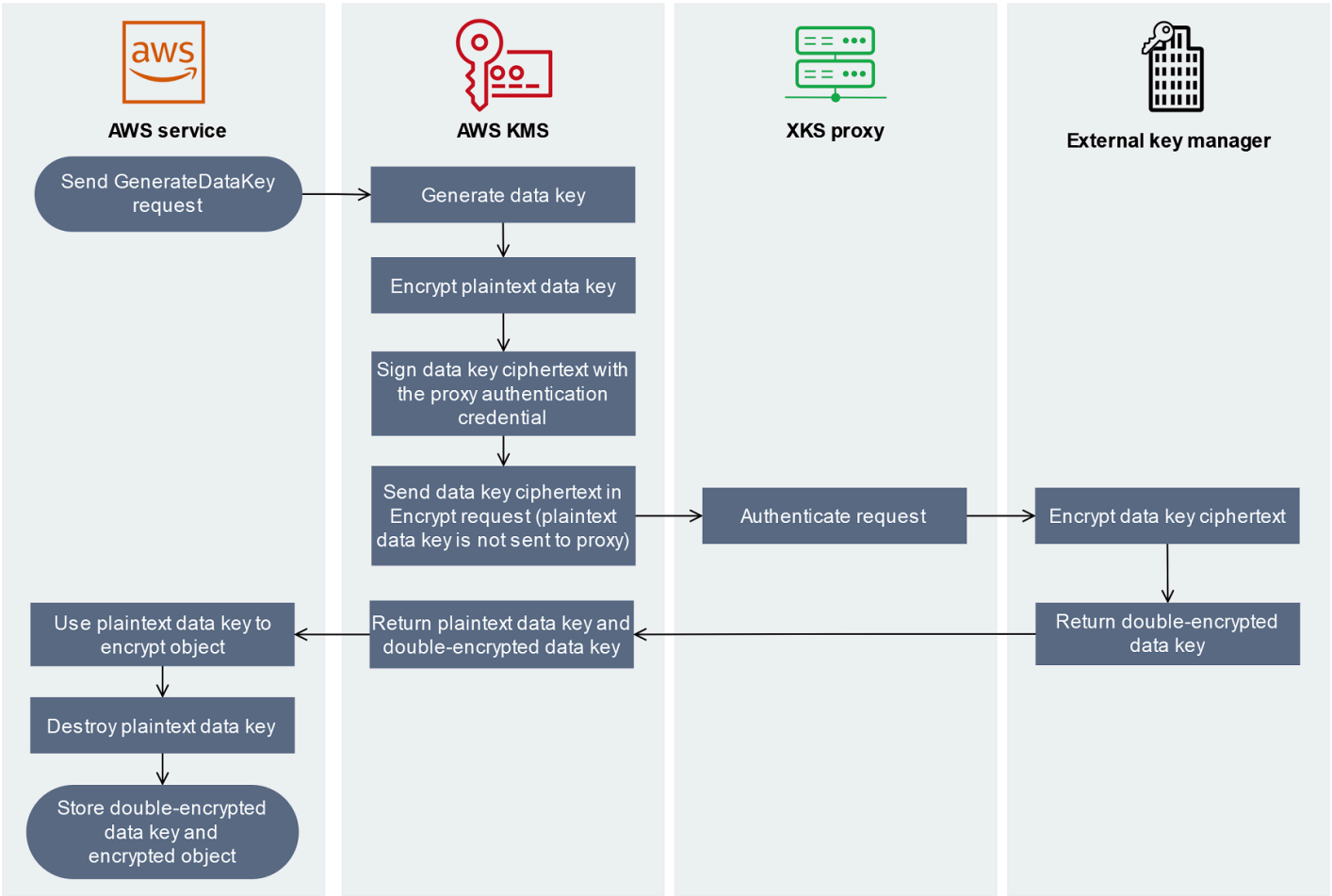
- 您無法重新建立從外部金鑰存放區刪除的 KMS 金鑰，即使您擁有其外部金鑰材料。每個 KMS 金鑰都擁有包含在對稱密文中的唯一中繼資料。新的 KMS 金鑰無法解密由原始金鑰加密的密文，即使其使用相同的外部金鑰材料。

如需做法中的雙重加密範例，請參閱 外部金鑰存放區的運作方式。

外部金鑰存放區的運作方式

您的外部金鑰存放區、外部金鑰存放區代理和外部金鑰管理器會共同運作，以保護您的 AWS 資源。以下程序描述了典型 AWS 服務 的加密工作流程，該服務使用由 KMS 金鑰保護的唯一資料金鑰對每個物件進行加密。在此情況下，您已選擇外部金鑰存放區中的 KMS 金鑰來保護物件。此範例顯示 AWS KMS 如何使用雙重加密來保護傳輸中的資料金鑰，並確保外部金鑰存放區中 KMS 金鑰所產生的密文始終至少與使用標準對稱 KMS 金鑰 (具有 AWS KMS 中的金鑰材料) 加密的密文一樣強。

與 AWS KMS 整合的每個實際 AWS 服務 使用的加密方法各不相同。如需詳細資訊，請參閱 AWS 服務 文件「安全性」一章中的「資料保護」主題。



1. 將新物件新增至 AWS 服務 資源。若要加密物件，會將GenerateDataKey要求AWS 服務傳送至外部金鑰存放區中AWS KMS使用 KMS 金鑰。
2. AWS KMS 會產生 256 位元對稱資料金鑰，並準備透過外部金鑰存放區代理將純文字資料金鑰複本傳送至外部金鑰管理器。AWS KMS 透過使用與外部金鑰存放區中 KMS 金鑰相關聯的 AWS KMS 金鑰材料加密純文字資料金鑰，以開始雙重加密程序。
3. AWS KMS 將加密請求傳送至與外部金鑰存放區相關聯的外部金鑰存放區代理。請求包含要加密的資料金鑰密文，以及與 KMS 金鑰相關聯的外部金鑰 ID。AWS KMS 使用外部金鑰存放區代理的代理身分驗證憑證來簽署請求。

   資料金鑰的純文字複本不會傳送至外部金鑰存放區代理。
4. 外部金鑰存放區代理會驗證請求，然後將加密請求傳送至外部金鑰管理器。

   某些外部金鑰存放區代理也會實作選用的授權政策，只允許選取的主體在特定條件下執行操作。
5. 您的外部金鑰管理器會使用指定的外部金鑰加密資料金鑰密文。外部金鑰管理器會將雙重加密的資料金鑰傳回至您的外部金鑰存放區代理，它會將其傳回 AWS KMS。
6. AWS KMS 會將純文字資料金鑰和該資料金鑰的雙重加密複本傳回給 AWS 服務。
7. AWS 服務 會使用純文字資料金鑰來加密資源物件，解密純文字資料金鑰，以及將加密的資料金鑰與加密的物件儲存在一起。

   一些 AWS 服務 可能會快取純文字資料金鑰以用於多個物件，或在使用資源時重複使用。如需詳細資訊，請參閱 無法使用的 KMS 金鑰如何影響資料金鑰。

若要解密加密物件，AWS 服務 必須在解密請求中將加密的資料金鑰傳回至 AWS KMS。若要解密已加密的資料金鑰，AWS KMS 必須使用外部金鑰 ID 將加密的資料金鑰傳送回外部金鑰存放區代理。如果對外部金鑰存放區代理的解密請求因任何原因而失敗，則 AWS KMS 無法解密已加密的資料金鑰，且 AWS 服務 無法解密已加密的物件。

控制對外部金鑰存放區的存取

對於外部金鑰存放區中的 KMS 金鑰而言，與標準 KMS 金鑰搭配使用的所有 AWS KMS 存取控制功能 (金鑰政策、IAM 政策和授予) 的運作方式相同。您可以使用 IAM 政策來控制對建立和管理外部金鑰存放區之 API 操作的存取。您可以使用 IAM 政策和金鑰政策，以控制對外部金鑰存放區中的 AWS KMS keys 的存取。您也可以使用 AWS 組織的服務控制政策和 VPC 端點政策來控制對外部金鑰存放區中 KMS 金鑰的存取。

建議您僅向使用者和角色提供執行任務所需的許可。

主題

- 授權外部金鑰存放區管理器
- 授權外部金鑰存放區中 KMS 金鑰的使用者
- 授權 AWS KMS 與外部金鑰存放區代理通訊
- 外部金鑰存放區代理授權 (選用)
- mTLS 身份驗證 (選用)

授權外部金鑰存放區管理器

建立及管理外部金鑰存放區的主體需要自訂金鑰存放區操作許可。以下清單說明了外部金鑰存放區管理器所需的最低許可。由於自訂金鑰存放區不是 AWS 資源，因此您無法為 AWS 帳戶 中的主體提供外部金鑰存放區許可。

- `kms:CreateCustomKeyStore`
- `kms:DescribeCustomKeyStores`
- `kms:ConnectCustomKeyStore`
- `kms:DisconnectCustomKeyStore`
- `kms:UpdateCustomKeyStore`
- `kms:DeleteCustomKeyStore`

建立外部金鑰存放區的主體需要許可，才能建立及設定外部金鑰存放區元件。主體只能在自己的帳戶中建立外部金鑰存放區。若要建立具有 VPC 端點服務連接的外部金鑰存放區，主體必須具有建立下列元件的許可：

- Amazon VPC
- 公有和私有子網路
- 網路負載平衡器和目標群組
- Amazon VPC 端點服務

如需詳細資訊，請參閱 Amazon VPC 的身分和存取管理、VPC 端點和 VPC 端點服務的身分和存取管理以及 Elastic Load Balancing API 許可。

授權外部金鑰存放區中 KMS 金鑰的使用者

在外部金鑰存放區建立和管理 AWS KMS keys 的主體，與在 AWS KMS 中建立和管理任何 KMS 金鑰的主體，需要相同的許可。外部金鑰存放區中 KMS 金鑰的預設金鑰政策與 AWS KMS 中 KMS 金鑰的

預設金鑰政策完全相同。屬性型存取控制 (ABAC) 使用標籤和別名來控制對 KMS 金鑰的存取，對外部金鑰存放區中的 KMS 金鑰也有效。

將自訂金鑰存放區中的 KMS 金鑰用於密碼編譯操作的委託人，需要許可對 KMS 金鑰執行密碼編譯操作，例如 kms:Decrypt。您可以在 IAM 或金鑰政策中提供這些許可。但是，他們使用自訂金鑰存放區中的 KMS 金鑰並不需要任何額外的許可。

若要設定僅適用於外部金鑰存放區中 KMS 金鑰的許可，請使用值為 EXTERNAL_KEY_STORE 的 kms:KeyOrigin 政策條件。您可以使用此條件來限制 kms：CreateKey權限或 KMS 金鑰資源專屬的任何權限。例如，下列 IAM 政策允許其所連接的身分，以便對帳戶中的所有 KMS 金鑰呼叫指定的操作，前提是 KMS 金鑰位於外部金鑰存放區中。請注意，您可以限制外部金鑰存放區中的 KMS 金鑰和 AWS 帳戶 中的 KMS 金鑰的許可，但不能限制帳戶中任何特定外部金鑰存放區的許可。

```
{
  "Sid": "AllowKeysInExternalKeyStores",
  "Effect": "Allow",
  "Action": [
    "kms:Encrypt",
    "kms:Decrypt",
    "kms:ReEncrypt*",
    "kms:GenerateDataKey*",
    "kms:DescribeKey"
  ],
  "Resource": "arn:aws:kms:us-west-2:111122223333:key/*",
  "Condition": {
    "StringEquals": {
      "kms:KeyOrigin": "EXTERNAL_KEY_STORE"
    }
  }
}
```

授權 AWS KMS 與外部金鑰存放區代理通訊

AWS KMS 只能透過您提供的外部金鑰存放區代理與外部金鑰管理器通訊。AWS KMS 透過搭配使用第 4 版簽署程序 (SigV4) 和您指定的外部金鑰存放區代理身分驗證憑證來簽署其請求，從而對代理進行身分驗證。如果您將公有端點連接用於外部金鑰存放區代理，則 AWS KMS 不需要任何其他許可。

不過，如果您使用的是 VPC 端點服務連接，則必須為 AWS KMS 授予許可，才能建立 Amazon VPC 端點服務的介面端點。無論外部金鑰存放區代理位於您的 VPC 中或者外部金鑰存放區代理位於其他位置，但使用 VPC 端點服務與 AWS KMS 通訊，則都需要此許可。

若AWS KMS要允許建立介面端點，請使用 Amazon VPC 主控台或ModifyVpcEndpointServicePermissions操作。允許下列主體的許可：cks.kms.*<region>*.amazonaws.com。

例如，下列 AWS CLI 命令允許 AWS KMS 連接到美國西部 (奧勒岡) (us-west-2) 區域中指定的 VPC 端點服務。使用此命令之前，請將 Amazon VPC 服務 ID 和 AWS 區域 取代為您組態的有效值。

```
modify-vpc-endpoint-service-permissions
--service-id vpce-svc-12abc34567def0987
--add-allowed-principals '["cks.kms.us-west-2.amazonaws.com"]'
```

若要移除此權限，請使用 Amazon VPC 主控台或與RemoveAllowedPrincipals參數ModifyVpcEndpointServicePermissions搭配使用。

外部金鑰存放區代理授權 (選用)

某些外部金鑰存放區代理會實作使用其外部金鑰的授權需求。允許外部金鑰存放區代理 (但不是必需的) 來設計和實作授權方案，該方案允許特定使用者僅在特定條件下請求特定操作。例如，代理可能會設定為允許使用者 A 使用特定的外部金鑰進行加密，但無法使用其進行解密。

代理授權獨立於 AWS KMS 對所有外部金鑰存放區代理所要求的基於 SigV4 的代理身分驗證。其也獨立於金鑰政策、IAM 政策和授予，其可授權存取會影響外部金鑰存放區或其 KMS 金鑰的操作。

若要透過外部金鑰存放區代理啟用授權，AWS KMS 會在每個代理 API 請求中包含中繼資料，這包括呼叫者、KMS 金鑰、AWS KMS 操作、AWS 服務 (如果有的話)。外部金鑰代理 API 第 1 版 (v1) 的請求中繼資料如下所示。

```
"requestMetadata": {
    "awsPrincipalArn": string,
    "awsSourceVpc": string, // optional
    "awsSourceVpce": string, // optional
    "kmsKeyArn": string,
    "kmsOperation": string,
    "kmsRequestId": string,
    "kmsViaService": string // optional
}
```

例如，您可以將代理設定為允許來自特定主體 (awsPrincipalArn) 的請求，但只有當特定 AWS 服務 (kmsViaService) 代表主體提出請求時才可以。

如果代理授權失敗，相關 AWS KMS 操作會失敗，並顯示說明錯誤的訊息。如需詳細資訊，請參閱 代理授權問題。

mTLS 身份驗證 (選用)

若要讓外部金鑰存放區代理對來自 AWS KMS 的請求進行身分驗證，AWS KMS 會使用外部金鑰存放區的 Signature V4 (SigV4) 代理身分驗證憑證將所有請求簽署至外部金鑰存放區代理。

為了進一步保證外部金鑰存放區代理僅回應 AWS KMS 請求，某些外部金鑰代理支援交互式 Transport Layer Security (mTLS)，在這種情況下，交易雙方都使用憑證來互相驗證。mTLS 會將用戶端身分驗證 (外部金鑰存放區代理伺服器會驗證 AWS KMS 用戶端) 新增至標準 TLS 提供的伺服器端身分驗證。在極少數情況下，您的代理身分驗證憑證會洩露，mTLS 會防止第三方對外部金鑰存放區代理發出成功的 API 請求。

若要實作 mTLS，請將外部金鑰存放區代理設定為僅接受具有下列屬性的用戶端 TLS 憑證：

- TLS 憑證上的主體通用名稱必須為 `cks.kms.<Region>.amazonaws.com`，例如 `cks.kms.eu-west-3.amazonaws.com`。
- 憑證必須鏈結至與 Amazon Trust Services 相關聯的憑證授權機構。

規劃外部金鑰存放區

在建立外部金鑰存放區之前，請選擇決定 AWS KMS 如何與外部金鑰存放區元件通訊的連接選項。您選擇的連接選項會決定規劃程序的剩餘部分。

進一步了解：

- 檢閱建立外部金鑰存放區的程序，包括備妥先決條件。它將幫助您確保在建立外部金鑰存放區時具有所需的所有元件。
- 了解如何控制對外部金鑰存放區的存取，包括外部金鑰存放區管理員和使用者所需的許可。
- 了解AWS KMS記錄外部金鑰存放區的 Amazon CloudWatch 指標和維度。強烈建議您建立警示來監控外部金鑰存放區，以便可偵測到效能和操作問題的早期跡象。

選擇代理連接選項

如果您要建立外部金鑰存放區，則需要決定 AWS KMS 如何與外部金鑰存放區代理通訊。此選擇將決定您需要哪些元件以及如何進行設定。AWS KMS 支援以下連接選項。選擇可滿足您的效能和安全目標的選項。

開始之前，請確認您需要外部金鑰存放區。大多數客戶都可以使用由 AWS KMS 金鑰材料支援的 KMS 金鑰。

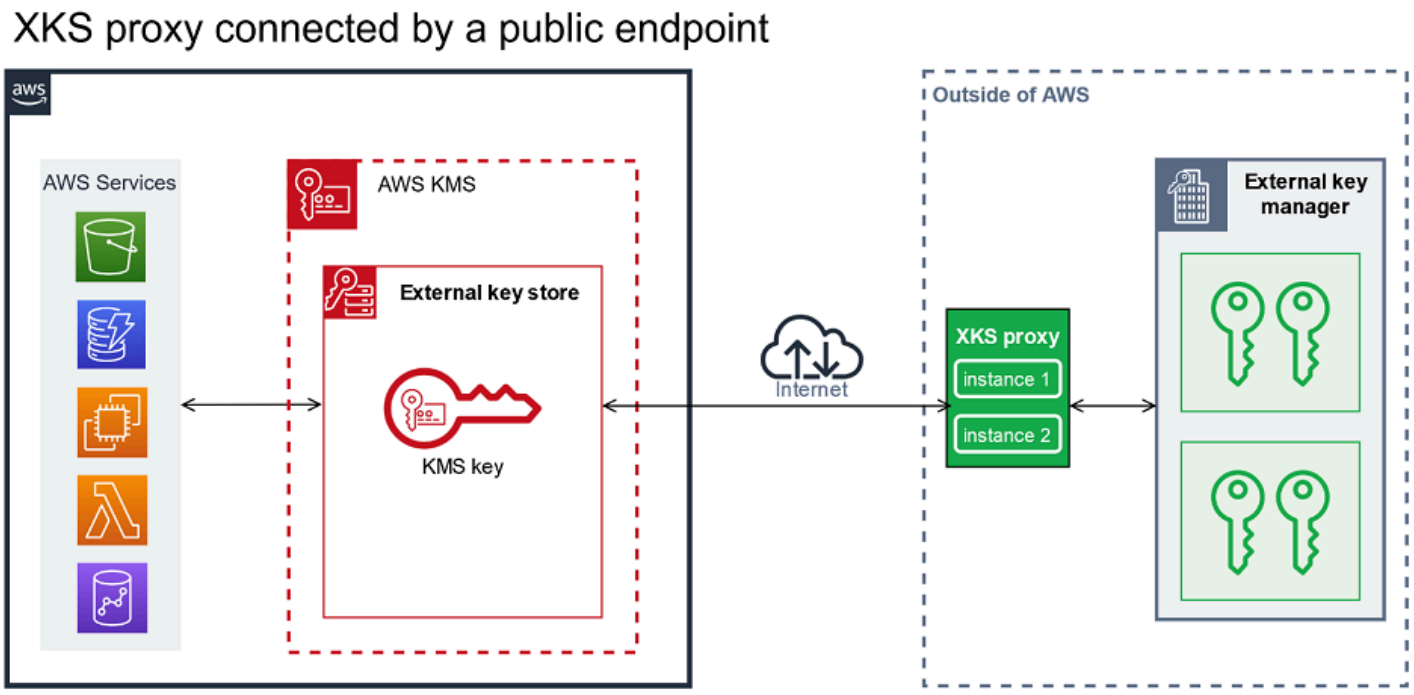
> **Note**
>
> 如果您的外部金鑰存放區代理內建於外部金鑰管理器中，則可能已預先決定您的連接。如需相關指導，請參閱外部金鑰管理器或外部金鑰存放區代理的文件。

即使在正操作的外部金鑰存放區中，您也可以變更外部金鑰存放區代理連接選項。但是，必須仔細規劃和執行此程序，以最大限度地減少中斷、避免錯誤並確保持續存取對您的資料進行加密的密碼編譯金鑰。

公有端點連接

AWS KMS 使用公有端點透過網際網路連接至外部金鑰存放區代理 (XKS 代理)。

此連接選項更易於設定和維護，並且可與某些金鑰管理模式完美配合。但是，其可能無法滿足某些組織的安全要求。



需求

如果您選擇公有端點連接，則需要下列項目。

- 您的外部金鑰存放區代理在公開可路由端點必須可存取。
- 您可對多個外部金鑰存放區使用相同的公有端點，前提是它們使用不同的代理 URI 路徑值。
- 對於相同 AWS 區域 中具有公有端點連接的外部金鑰存放區以及具有 VPC 端點服務連接的任何外部金鑰存放區，不能使用相同的端點，即使這些金鑰存放區位於不同的 AWS 帳戶 中。
- 您必須取得由外部金鑰存放區支援的公有憑證授權機構核發的 TLS 憑證。如需清單，請參閱受信任的憑證授權機構。

  TLS 憑證上的主體通用名稱 (CN) 必須與外部金鑰存放區代理之代理 URI 端點中的網域名稱相符。例如，如果公有端點為 `https://myproxy.xks.example.com`，則 TLS 憑證上的通用名稱必須為 `myproxy.xks.example.com` 或 `*.xks.example.com`。
- 確保 AWS KMS 與外部金鑰存放區代理之間的任何防火牆都允許代理上連接埠 443 的進出流量。AWS KMS 在連接埠 443 上進行通訊。此值不可設定。

如需外部金鑰存放區的所有要求，請參閱備妥先決條件。

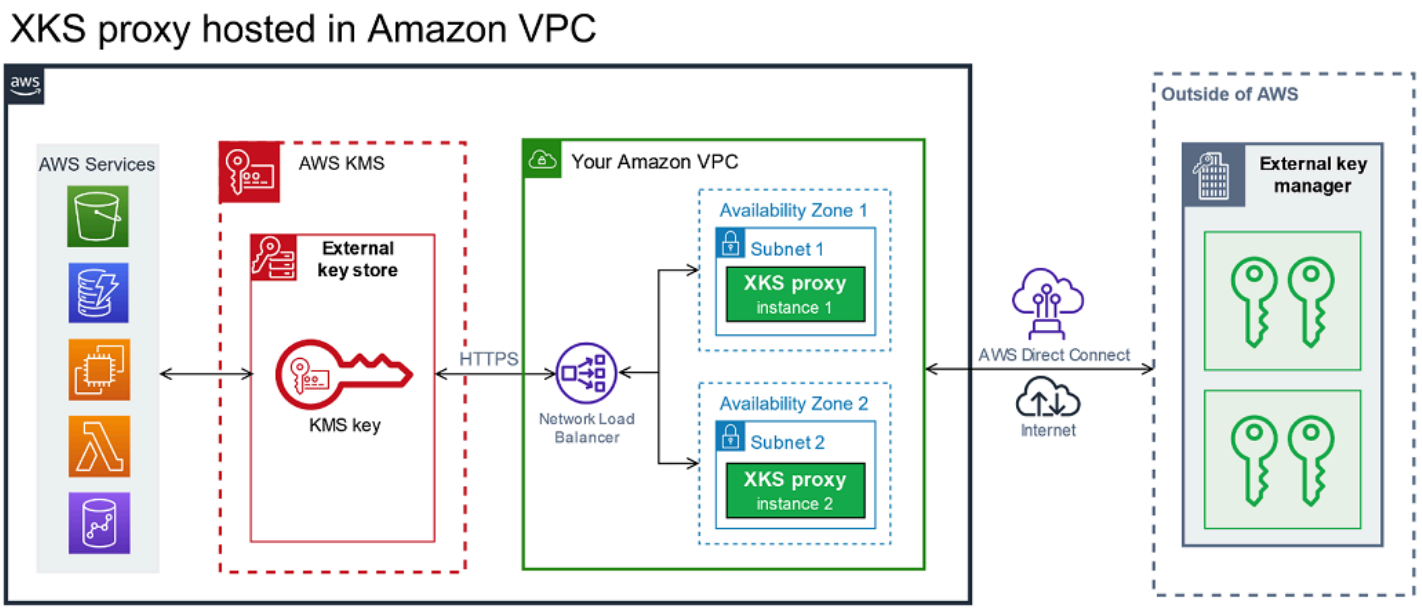
VPC 端點服務連接

透過為您建立和設定的 Amazon VPC 端點服務建立介面端點，AWS KMS 連接至外部金鑰存放區代理 (XKS 代理)。您必須負責建立 VPC 端點服務，並將 VPC 連接至外部金鑰管理器。

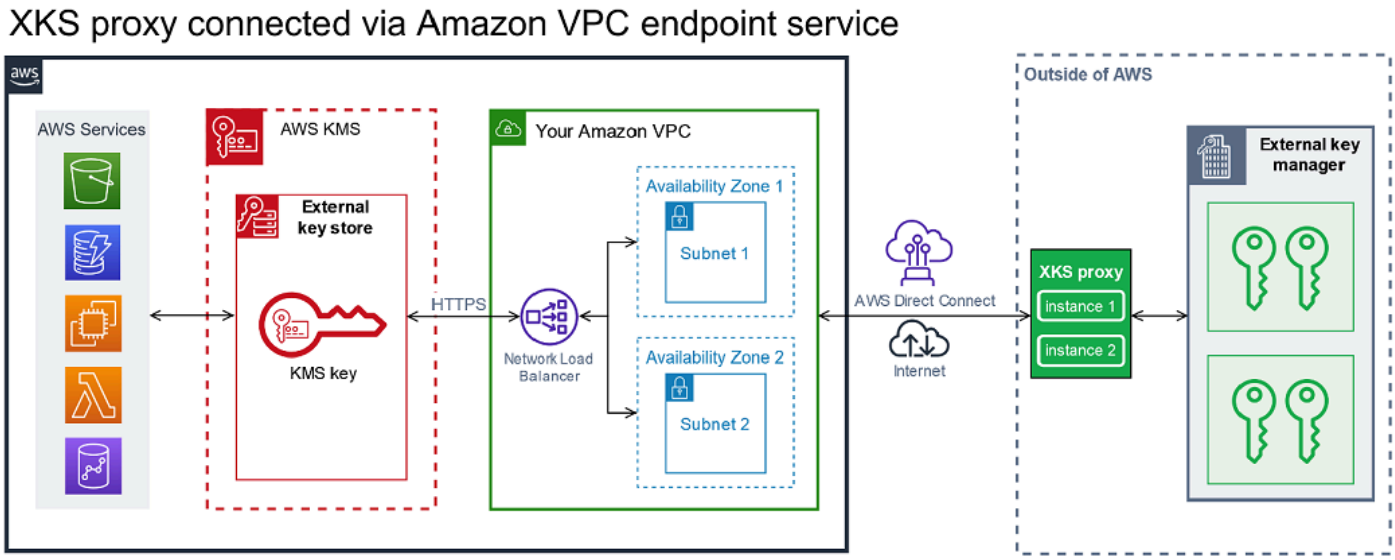
您的端點服務可以使用任何支援的網路到 Amazon VPC 選項進行通訊，包括 AWS Direct Connect。

此連接選項的設定和維護更為複雜。但它使用的是 AWS PrivateLink，使 AWS KMS 能夠私密連接到您的 Amazon VPC 和外部金鑰存放區代理，而無需使用公有網際網路。

您可以在 Amazon VPC 中找到外部金鑰存放區代理。



或者，找到 AWS 之外的外部金鑰存放區代理，並僅將 Amazon VPC 端點服務用於與 AWS KMS 之間的安全通訊。



設定 VPC 端點服務連接

使用本節中的指導來建立和設定使用 VPC 端點服務連接的外部金鑰存放區所需的 AWS 資源和相關元件。針對此連接選項列出的資源是所有外部金鑰存放區所需資源的補充。建立並設定所需資源後，您可以建立外部金鑰存放區。

您可以在 Amazon VPC 中找到外部金鑰存放區代理，或在 AWS 之外找到代理，並使用 VPC 端點服務進行通訊。

開始之前，請確認您需要外部金鑰存放區。大多數客戶都可以使用由 AWS KMS 金鑰材料支援的 KMS 金鑰。

> **Note**
>
> VPC 端點服務連接所需的一些元素可能包含在外部金鑰管理器中。此外，您的軟體可能還有其他組態要求。在建立和設定本節中的 AWS 資源之前，請參閱代理和金鑰管理器文件。

主題

- VPC 端點服務連接的要求
- 建立 Amazon VPC 和子網路
- 建立目標群組
- 建立網路負載平衡器
- 建立 VPC 端點服務
- 驗證您的私有 DNS 名稱網域
- 授權 AWS KMS 連接至 VPC 端點服務

VPC 端點服務連接的要求

如果您為外部金鑰存放區選擇 VPC 端點服務連接，則需要下列資源。

若要將網路延遲降到最低，請在離外部金鑰管理器最近的受支援的 AWS 區域 中建立 AWS 元件。如果可能，請選擇網路封包來回時間 (RTT) 為 35 毫秒或更短的區域。

- 連接到外部金鑰管理器的 Amazon VPC。其在兩個不同可用區域中必須擁有至少兩個私有子網路。

  您可以將現有的 Amazon VPC 用於外部金鑰存放區，前提是其符合與外部金鑰存放區搭配使用的要求。多個外部金鑰存放區可以共用 Amazon VPC，但是每個外部金鑰存放區都必須擁有自己的 VPC 端點服務和私有 DNS 名稱。
- 由 AWS PrivateLink 提供支援的 Amazon VPC 端點服務，具有網路負載平衡器和目標群組。

  端點服務無法要求接受。此外，您必須新增 AWS KMS 作為允許的主體。這允許 AWS KMS 建立介面端點，以便其可以與您的外部金鑰存放區代理進行通訊。
- VPC 端點服務的私有 DNS 名稱在其 AWS 區域 中是唯一的。

私有 DNS 名稱必須是較高層級公有網域的子網域。例如，如果私有 DNS 名稱為 `myproxy-private.xks.example.com`，則其必須是公有網域的子網域，例如 `xks.example.com` 或 `example.com`。

您必須驗證私有 DNS 名稱的 DNS 網域的擁有權。

- 支援的公有憑證授權機構為您的外部金鑰存放區代理核發的 TLS 憑證。

  TLS 憑證上的主體通用名稱 (CN) 必須與私有 DNS 名稱相符。例如，如果私有 DNS 名稱為 `myproxy-private.xks.example.com`，則 TLS 憑證上的 CN 必須為 `myproxy-private.xks.example.com` 或 `*.xks.example.com`。

如需外部金鑰存放區的所有要求，請參閱備妥先決條件。

建立 Amazon VPC 和子網路

VPC 端點服務連接需要一個 Amazon VPC，其使用至少兩個私有子網路連接到外部金鑰管理器。您可以建立 Amazon VPC，或使用符合外部金鑰存放區要求的現有 Amazon VPC。如需有關建立新 Amazon VPC 的說明，請參閱《Amazon 虛擬私有雲端使用者指南》中的建立 VPC。

Amazon VPC 的要求

若要透過 VPC 端點服務連接來搭配使用外部金鑰存放區，Amazon VPC 必須具有下列屬性：

- 必須與外部金鑰存放區位於相同的 AWS 帳戶 和受支援的區域中。
- 需要至少兩個私有子網路，每個處於不同的可用區域。
- Amazon VPC 的私有 IP 地址範圍不得與託管外部金鑰管理器之資料中心的私有 IP 地址範圍重疊。
- 所有元件都必須使用 IPv4。

您有許多選項可將 Amazon VPC 連接到外部金鑰存放區代理。選擇可滿足您的效能和安全需求的選項。如需清單，請參閱將您的 VPC 連接到其他網路和網路到 Amazon VPC 連接選項。如需詳細資訊，請參閱 AWS Direct Connect 和 AWS Site-to-Site VPN 使用者指南。

為您的外部金鑰存放區建立 Amazon VPC

遵循以下指示建立外部金鑰存放區的 Amazon VPC。只有在選擇 VPC 端點服務連接選項時，才需要 Amazon VPC。您可以使用符合外部金鑰存放區要求的現有 Amazon VPC。

遵循建立 VPC、子網路和其他 VPC 資源主題中的指示，使用以下所需值。對於其他欄位，請接受預設值，並按要求提供名稱。

欄位	值
IPv4 CIDR 區塊	輸入 VPC 的 IP 地址。Amazon VPC 的私有 IP 地址範圍不得與託管外部金鑰管理器之資料中心的私有 IP 地址範圍重疊。
可用區域 (AZ) 的數量	2 或以上
公有子網路數量	需要空值 (0)
私有子網路數量	每個 AZ 一個
NAT 閘道	需要空值。
VPC 端點	需要空值。
Enable DNS hostnames (啟用 DNS 主機名稱)	是
啟用 DNS 解析	是

請務必測試 VPC 通訊。例如，如果您的外部金鑰存放區代理不在 Amazon VPC 中，則請在 Amazon VPC 中建立 Amazon EC2 執行個體，確認 Amazon VPC 可以與您的外部金鑰存放區代理通訊。

將 VPC 連接到外部金鑰管理器

使用 Amazon VPC 支援的任何網路連接選項，將 VPC 連接到託管外部金鑰管理器的資料中心。確保 VPC 中的 Amazon EC2 執行個體 (或外部金鑰存放區代理 (如果位於 VPC 中) 可以與資料中心和外部金鑰管理器通訊。

建立目標群組

建立必要的 VPC 端點服務之前，請先建立其必要元件、網路負載平衡器 (NLB) 和目標群組。網路負載平衡器 (NLB) 會在多個狀態良好的目標之間分佈請求，其中任何一個都可以為請求提供服務。在此步驟中，您會為外部金鑰存放區代理建立至少具有兩個主機的目標群組，並向目標群組註冊 IP 地址。

遵循設定目標群組主題中的指示，使用以下所需值。對於其他欄位，請接受預設值，並按要求提供名稱。

欄位	值
Target type (目標類型)	IP 位址
通訊協定	TCP
連線埠	443
IP 地址類型	IPv4
VPC	選擇您要在其中為外部金鑰存放區建立 VPC 端點服務的 VPC。
運作狀態檢查通訊協定和路徑	您的運作狀態檢查通訊協定和路徑會因外部金鑰存放區代理組態而有所不同。請參閱外部金鑰管理器或外部金鑰存放區代理的文件。 如需有關針對目標群組設定運作狀態檢查的一般資訊，請參閱《Network Load Balancer 之 Elastic Load Balancing 使用者指南》中的目標群組運作狀態檢查。
網路	其他私有 IP 地址
IPv4 地址	外部金鑰存放區代理的私有地址
連接埠	443

建立網路負載平衡器

網路負載平衡器會將網路流量 (包括從 AWS KMS 到外部金鑰存放區代理的請求) 分佈至設定的目標。

遵循設定負載平衡器和接聽程式主題中的指示，設定和新增接聽程式，並使用下列必要值建立負載平衡器。對於其他欄位，請接受預設值，並按要求提供名稱。

欄位	值
Scheme	內部 (Internal)
IP 地址類型	IPv4
網路映射	選擇您要在其中為外部金鑰存放區建立 VPC 端點服務的 VPC。

欄位	值
映射	選擇您為 VPC 子網路設定的兩個可用區域 (至少兩個)。驗證子網路名稱和私有 IP 地址。
通訊協定	TCP
連線埠	443
預設動作：轉送至	選擇網路負載平衡器的目標群組。

建立 VPC 端點服務

通常，您會建立服務的端點。但是，當您建立 VPC 端點服務時，您就是提供者，AWS KMS 會為您的服務建立端點。對於外部金鑰存放區，請使用您在上一個步驟中建立的網路負載平衡器建立 VPC 端點服務。VPC 端點服務必須與外部金鑰存放區處於相同的 AWS 帳戶 和受支援的區域中。

多個外部金鑰存放區可以共用 Amazon VPC，但是每個外部金鑰存放區都必須擁有自己的 VPC 端點服務和私有 DNS 名稱。

遵循建立端點服務主題中的指示，使用下列必要值建立 VPC 端點服務。對於其他欄位，請接受預設值，並按要求提供名稱。

欄位	值
負載平衡器類型	網路
可用的負載平衡器	選擇您在上一步驟中建立的網路負載平衡器。 如果您的新負載平衡器未出現在清單中，請確認它處於作用中狀態。負載平衡器狀態可能需要幾分鐘才能從佈建變更為作用中。
需要接受	False。取消勾選核取方塊。 不需要接受。如果沒有手動接受，AWS KMS 無法連接至 VPC 端點服務。如果需要接受，嘗試建立外部金鑰存放區會失敗，並出現 `XksProxyInvalidConfigurationException` 例外狀況。

欄位	值
啟用私有 DNS 名稱	將私有 DNS 名稱與服務建立關聯
私有 DNS 名稱	輸入在其 AWS 區域 中唯一的私有 DNS 名稱。 私有 DNS 名稱必須是較高層級公有網域的子網域。例如，如果私有 DNS 名稱為 myproxy-private.xks.example.com ，則其必須是公有網域的子網域，例如 xks.example.com 或 example.com 。 此私有 DNS 名稱必須與外部金鑰存放區代理上設定的 TLS 憑證中的主體通用名稱 (CN) 相符。例如，如果私有 DNS 名稱為 myproxy-private.xks.example.com ，則 TLS 憑證上的 CN 必須為 myproxy-private.xks.example.com 或 *.xks.example.com 。 如果憑證和私有 DNS 名稱不相符，則嘗試將外部金鑰存放區連接至其外部金鑰存放區代理會失敗，並顯示 XKS_PROXY_INVALID_TLS_CONFIGURATION 的連接錯誤代碼。如需詳細資訊，請參閱 一般組態錯誤。
支援的 IP 地址類型	IPv4

驗證您的私有 DNS 名稱網域

建立 VPC 端點服務時，其網域驗證狀態為 pendingVerification。在使用 VPC 端點服務建立外部金鑰存放區之前，此狀態必須為 verified。若要確認您擁有與私有 DNS 名稱相關聯的網域，您必須在公有 DNS 伺服器中建立 TXT 記錄。

例如，如果 VPC 端點服務的私有 DNS 名稱為 myproxy-private.xks.example.com，則必須在公有網域 (例如 xks.example.com 或 example.com) 中建立 TXT 記錄，以公有網域為準。AWS PrivateLink 首先在 xks.example.com 上查找 TXT 記錄，然後在 example.com 上查找。

> **Tip**
>
> 新增 TXT 記錄之後，Domain verification status (網域驗證狀態) 值可能需要幾分鐘才能從 pendingVerification 變更為 verify。

若要開始，請使用下列其中一種方法查找網域的驗證狀態。有效值為 `verified`、`pendingVerification` 和 `failed`。

- 在 Amazon VPC 主控台中，選擇 Endpoint services (端點服務)，然後選擇您的端點服務。在詳細資訊窗格中，請查看 Domain verification status (網域驗證狀態)。
- 使用DescribeVpcEndpointServiceConfigurations操作。`State` 值位於 `ServiceConfigurations.PrivateDnsNameConfiguration.State` 欄位中。

如果驗證狀態不是 `verified`，請遵循網域擁有權驗證主題中的指示，將 TXT 記錄新增至網域的 DNS 伺服器，並確認 TXT 記錄已發佈。然後再次檢查您的驗證狀態。

您不需要為私有 DNS 網域名稱建立 A 記錄。當 AWS KMS 建立 VPC 端點服務的介面端點時，AWS PrivateLink 會自動建立託管區域，它具有 AWS KMS VPC 中私有網域名稱所需的 A 記錄。對於具有 VPC 端點服務連接的外部金鑰存放區，當您將外部金鑰存放區連接至其外部金鑰存放區代理時，會發生這種情況。

授權 AWS KMS 連接至 VPC 端點服務

您必須將 AWS KMS 新增至 VPC 端點服務的 Allow principals (允許主體) 清單。這允許 AWS KMS 建立 VPC 端點服務的介面端點。如果 AWS KMS 不是允許的主體，嘗試建立外部金鑰存放區將會失敗，並出現 `XksProxyVpcEndpointServiceNotFoundException` 例外狀況。

遵循《AWS PrivateLink 指南》中的管理許可主題。使用以下所需值。

欄位	值
ARN	`cks.kms.<region>.amazonaws.com` 例如：`cks.kms.us-east-1.amazonaws.com`

下一頁: 建立外部金鑰存放區

管理外部金鑰存放區

您可以使用 AWS KMS 主控台或 AWS KMS API 來管理外部金鑰存放區。您可以建立外部金鑰存放區，檢視並編輯其屬性、監控其效能，從其外部金鑰存放區代理中連接和中斷連接，以及刪除外部金鑰存放區。

主題

- 建立外部金鑰存放區
- 編輯外部金鑰存放區屬性
- 檢視外部金鑰存放區
- 監控外部金鑰存放區
- 連接和中斷連接外部金鑰存放區
- 刪除外部金鑰存放區

建立外部金鑰存放區

您可以在每個 AWS 帳戶 和區域中建立一個或許多外部金鑰存放區。每個外部金鑰存放區必須與 AWS 之外的外部金鑰管理器以及協調 AWS KMS 與外部金鑰管理器之間通訊的外部金鑰存放區代理 (XKS 代理) 相關聯。如需詳細資訊，請參閱 規劃外部金鑰存放區。開始之前，請確認您需要外部金鑰存放區。大多數客戶都可以使用由 AWS KMS 金鑰材料支援的 KMS 金鑰。

> **Tip**
>
> 有些外部金鑰管理器會提供更簡單的方法來建立外部金鑰存放區。如需詳細資訊，請參閱外部金鑰管理器文件。

在您建立外部金鑰存放區之前，您需要備妥先決條件。在建立過程中，您可以指定外部金鑰存放區的屬性。最重要的是，您可以指出 AWS KMS 中的外部金鑰存放區是否使用公有端點或 VPC 端點服務來連接到其外部金鑰存放區代理。您也可以指定連接詳細資料，包括代理的 URI 端點，以及 AWS KMS 將 API 請求傳送至代理的代理端點內的路徑。

- 如果您使用公有端點連接，則請確定 AWS KMS 可以使用 HTTPS 連接透過網際網路與代理通訊。這包括在外部金鑰存放區代理上設定 TLS，並確保 AWS KMS 和代理之間的任何防火牆在代理上都允許連接埠 443 的進出流量。建立具有公有端點連接的外部金鑰存放區時，AWS KMS 透過將狀態請求傳送至外部金鑰存放區代理來測試連接。此測試會驗證端點是否可連接，以及您的外部金鑰存放區代理是否將接受使用外部金鑰存放區代理身分驗證憑證簽署的請求。如果此測試請求失敗，則建立外部金鑰存放區的操作失敗。
- 如果您使用 VPC 端點服務連接，則請確定網路負載平衡器、私有 DNS 名稱和 VPC 端點服務設定正確且可運作。如果外部金鑰存放區代理不在 VPC 中，您需要確保 VPC 端點服務可以與外部金鑰存放區代理通訊。(當您將外部金鑰存放區連接至其外部金鑰存放區代理時，AWS KMS 會測試 VPC 端點服務連接。)

其他考量：

- AWS KMS記錄 Amazon CloudWatch 指標和維度，特別是對於外部金鑰存放區。以這些指標為依據的監控圖表會顯示在每個外部金鑰存放區的 AWS KMS 主控台中。強烈建議您使用這些指標來建立警示，以監控您的金鑰存放區。這些警示會在效能和操作問題的相關跡象發生之前提醒您。如需說明，請參閱監控外部金鑰存放區。
- 外部金鑰存放區受資源配額限制。在外部金鑰存放區中使用 KMS 金鑰會受到請求配額的限制。在設計外部金鑰存放區實作之前，請檢閱這些配額。

> **Note**
>
> 檢閱您的組態，了解可能導致其無法運作的循環相依性。
> 例如，如您利用 AWS 資源建立外部金鑰存放區代理，請確定在透過該代理存取的外部金鑰存放區操作代理不需可用的 KMS 金鑰。

在中斷連接的狀態下建立所有新的外部金鑰存放區。在外部金鑰存放區中建立 KMS 金鑰之前，必須先將其連接至外部金鑰存放區代理。若要變更外部金鑰存放區的屬性，請編輯外部金鑰存放區設定。

主題

備妥先決條件

在建立外部金鑰存放區之前，您需要組合必要的元件，包括您將用來支援外部金鑰存放區的外部金鑰管理器，以及將 AWS KMS 請求轉換為外部金鑰管理器可以理解之格式的外部金鑰存放區代理。

所有外部金鑰存放區都需要下列元件。除了這些元件之外，您還需要提供其他元件來支援您選擇的外部金鑰存放區代理連接選項。

> **Tip**
>
> 您的外部金鑰管理器可能包含其中一些元件，也可能會為您設定這些元件。如需詳細資訊，請參閱外部金鑰管理器文件。

如果您在 AWS KMS 主控台中建立外部金鑰存放區，則可以選擇上傳基於 JSON 的代理組態文件，該文件指定代理 URI 路徑和代理身分驗證憑證。某些外部金鑰存放區代理會為您產生此檔案。如需詳細資訊，請參閱外部金鑰存放區代理或外部金鑰管理器的文件。

外部金鑰管理器

每個外部金鑰存放區至少需要一個外部金鑰管理器執行個體。這可以是實體或虛擬硬體安全模組 (HSM)，也可以是金鑰管理軟體。

您可以使用單一金鑰管理器，但建議有至少兩個相關的金鑰管理器執行個體共用密碼編譯金鑰以進行備援。外部金鑰存放區不需要獨佔使用外部金鑰管理器。不過，外部金鑰管理器必須能夠處理來自 AWS 服務之加密和解密請求的預期頻率，這些服務使用外部金鑰存放區中的 KMS 金鑰來保護您的資源。您的外部金鑰管理器應設定為每秒最多可處理 1800 個請求，並在每個請求的 250 毫秒逾時內回應。建議您將外部金鑰管理器定位在靠近 AWS 區域 的位置，以便網路封包來回時間 (RTT) 為 35 毫秒或更短。

如果您的外部金鑰存放區代理允許，則您可以變更與外部金鑰存放區代理關聯的外部金鑰管理器，但新的外部金鑰管理器必須是具有相同金鑰材料的備份或快照。如果與 KMS 金鑰關聯的外部金鑰無法再供外部金鑰存放區代理使用，則 AWS KMS 無法解密使用 KMS 金鑰加密的密文。

外部金鑰管理器必須可供外部金鑰存放區代理存取。如果代理伺服器的GetHealthStatus回應報告所有外部金鑰管理員執行個體都是`Unavailable`，則建立外部金鑰存放區的所有嘗試都會失敗，並顯示`XksProxyUriUnreachableException`.

外部金鑰存放區代理

您必須指定外部金鑰存放區代理 (XKS 代理)，其符合 AWS KMS 外部金鑰存放區代理規格中的設計需求。您可以開發或購買外部金鑰存放區代理，或使用外部金鑰管理器所提供或內建的外部金鑰存放區代理。AWS KMS 建議您將外部金鑰存放區代理設定為每秒最多可處理 1800 個請求，並在每個請求的 250 毫秒逾時內回應。建議您將外部金鑰管理器定位在靠近 AWS 區域 的位置，以便網路封包來回時間 (RTT) 為 35 毫秒或更短。

您可以將外部金鑰存放區代理用於多個外部金鑰存放區，但是每個外部金鑰存放區在其請求的外部金鑰存放區代理內必須具有唯一的 URI 端點和路徑。

如果您使用的是 VPC 端點服務連接，則可以在 Amazon VPC 中找到外部金鑰存放區代理，但這不是必需的。您可以在 AWS 之外找到代理，例如在私有資料中心，並僅使用 VPC 端點服務與 Proxy 通訊。

代理身分驗證憑證

若要建立外部金鑰存放區，您必須指定外部金鑰存放區代理身分驗證憑證 (`XksProxyAuthenticationCredential`)。

您必須在外部金鑰存放區代理上為 AWS KMS 建立身分驗證憑證 (`XksProxyAuthenticationCredential`)。AWS KMS 透過搭配使用第 4 版簽署程序 (SigV4) 與外部金鑰存放區代理身分驗證憑證來簽署其請求，從而對代理進行身分驗證。您可以在建立外部金鑰存放區時指定身分驗證憑證，而且您可以隨時將其變更。如果代理輪換憑證，則請務必更新外部金鑰存放區的憑證值。

代理身分驗證憑證有兩部分。您必須為外部金鑰存放區提供這兩部分。

- 存取金鑰 ID：識別私密存取金鑰。您可以提供純文字形式的 ID。
- 私密存取金鑰：憑證的秘密部分。AWS KMS 在儲存憑證之前，先加密其中的私密存取金鑰。

AWS KMS 用於向外部金鑰存放區代理簽署請求的 SigV4 憑證與 AWS 帳戶中的任何 AWS Identity and Access Management 主體相關聯的任何 SigV4 憑證無關。請勿針對外部金鑰存放區代理重複使用任何 IAM SigV4 憑證。

代理連接

若要建立外部金鑰存放區，您必須指定外部金鑰存放區代理連接選項 (`XksProxyConnectivity`)。

AWS KMS 透過使用公有端點或 Amazon Virtual Private Cloud (Amazon VPC) 端點服務與外部金鑰存放區代理通訊。雖然公有端點的設定和維護較為簡單，但可能不符合每個安裝的安全需求。如果選擇 Amazon VPC 端點服務連接選項，則必須建立和維護所需元件，包括在兩個不同可用區域中至少具有兩個子網路的 Amazon VPC、具有網路負載平衡器和目標群組的 VPC 端點服務，以及 VPC 端點服務的私有 DNS 名稱。

您可以變更外部金鑰存放區的代理連接選項。但是，您必須確保與外部金鑰存放區中 KMS 金鑰相關聯的金鑰材料持續可用。否則，AWS KMS 無法解密使用這些 KMS 金鑰加密的任何密文。

如需決定哪個代理連接選項最適合您的外部金鑰存放區的說明，請參閱 選擇代理連接選項。如需建立和設定 VPC 端點服務連接的說明，請參閱 設定 VPC 端點服務連接。

代理 URI 端點

若要建立外部金鑰存放區，您必須指定 AWS KMS 用來將請求傳送至外部金鑰存放區代理的端點 (`XksProxyUriEndpoint`)。

通訊協定必須為 HTTPS。AWS KMS 在連接埠 443 上進行通訊。請勿在代理 URI 端點值中指定連接埠。

- 公有端點連接 – 指定外部金鑰存放區代理的公開可用端點。在建立外部金鑰存放區之前，必須可連接此端點。
- VPC 端點服務連接 – 指定 `https://`，後跟 VPC 端點服務的私有 DNS 名稱。

在外部金鑰存放區代理上設定的 TLS 伺服器憑證必須與外部金鑰存放區代理 URI 端點中的網域名稱相符，並由外部金鑰存放區支援的憑證授權機構發行。如需清單，請參閱受信任的憑證授權機構。在發行 TLS 憑證之前，您的憑證授權機構將要求提供網域擁有權證明。

TLS 憑證上的主體通用名稱 (CN) 必須與私有 DNS 名稱相符。例如，如果私有 DNS 名稱為 `myproxy-private.xks.example.com`，則 TLS 憑證上的 CN 必須為 `myproxy-private.xks.example.com` 或 `*.xks.example.com`。

您可以變更代理 URI 端點，但請確定外部金鑰存放區代理可存取與外部金鑰存放區中 KMS 金鑰相關聯的金鑰材料。否則，AWS KMS 無法解密使用這些 KMS 金鑰加密的任何密文。

唯一性要求

- 組合的代理 URI 端點 (`XksProxyUriEndpoint`) 和代理 URI 路徑 (`XksProxyUriPath`) 值在 AWS 帳戶 和區域中必須唯一。
- 具有公有端點連接的外部金鑰存放區可以共用相同的代理 URI 端點，前提是其具有不同的代理 URI 路徑值。
- 具有公有端點連接的外部金鑰存放區不能使用與同一 AWS 區域 中具有 VPC 端點服務連接之任何外部金鑰存放區相同的代理 URI 端點值，即使金鑰存放區位於不同的 AWS 帳戶 中。
- 具有 VPC 端點連接的每個外部金鑰存放區都必須有自己的私有 DNS 名稱。代理 URI 端點 (私有 DNS 名稱) 在 AWS 帳戶 和區域中必須唯一。

代理 URI 路徑

若要建立外部金鑰存放區，您必須在外部金鑰存放區代理中指定所需代理 API 的基本路徑。該值必須以 / 開頭，且必須以 /kms/xks/v1 結尾，其中 v1 表示外部金鑰存放區代理的 AWS KMS API 版本。此路徑可以在必要元素之間包含選用字首，例如 `/example-prefix/kms/xks/v1`。若要尋找此值，請參閱外部金鑰存放區代理的文件。

AWS KMS 將代理請求傳送到代理 URI 端點和代理 URI 路徑的組合所指定的地址。例如，如果代理 URI 端點為 `https://myproxy.xks.example.com`，而代理 URI 路徑為 `/kms/xks/v1`，則 AWS KMS 會將其代理 API 請求傳送到 `https://myproxy.xks.example.com/kms/xks/v1`。

您可以變更代理 URI 路徑，但請確定外部金鑰存放區代理可存取與外部金鑰存放區中 KMS 金鑰相關聯的金鑰材料。否則，AWS KMS 無法解密使用這些 KMS 金鑰加密的任何密文。

唯一性要求

- 組合的代理 URI 端點 (`XksProxyUriEndpoint`) 和代理 URI 路徑 (`XksProxyUriPath`) 值在 AWS 帳戶 和區域中必須唯一。

VPC 端點服務

指定用於與外部金鑰存放區代理通訊的 Amazon VPC 端點服務名稱。只有使用 VPC 端點服務連接的外部金鑰存放區才需要此元件。如需為外部金鑰存放區設定和配置 VPC 端點服務的說明，請參閱 設定 VPC 端點服務連接。

VPC 端點服務必須具有下列屬性：

- VPC 端點服務必須與外部金鑰存放區處於相同的 AWS 帳戶 和區域中。
- 其必須擁有已連接到至少兩個子網路的網路負載平衡器 (NLB)，每個位於不同的可用區域中。
- VPC 端點服務的允許主體清單必須包含該區域的 AWS KMS 服務主體：`cks.kms.<region>.amazonaws.com`，例如 `cks.kms.us-east-1.amazonaws.com`。
- 其不能要求接受連接請求。
- 其在較高級別的公有網域中必須具有私有 DNS 名稱。例如，在公有 `xks.example.com` 網域中擁有私有 DNS 名稱 myproxy-private.xks.example.com。

  具有 VPC 端點服務連接之外部金鑰存放區的私有 DNS 名稱在其 AWS 區域 中必須唯一。
- 私有 DNS 名稱網域的網域驗證狀態必須為 `verified`。
- 在外部金鑰存放區代理上設定的 TLS 伺服器憑證必須指定可連接端點的私有 DNS 主機名稱。

唯一性要求

- 具有 VPC 端點連接的外部金鑰存放區可以共用 Amazon VPC，但是每個外部金鑰存放區都必須有自己的 VPC 端點服務和私有 DNS 名稱。

代理組態檔案

代理組態檔案是一個可選的基於 JSON 的檔案，其中包含外部金鑰存放區的代理 URI 路徑和代理身分驗證憑證屬性的值。在 AWS KMS 主控台中建立或編輯外部金鑰存放區時，您可以上傳代理組態檔案，以提供外部金鑰存放區的組態值。使用此檔案可避免輸入和貼上錯誤，並確保外部金鑰存放區中的值與外部金鑰存放區代理中的值相符。

代理組態檔案由外部金鑰存放區代理產生。若要了解您的外部金鑰存放區代理是否提供代理組態檔案，請參閱您的外部金鑰存放區代理文件。

以下是具有虛構值的正確代理組態檔案範例。

```
{
  "XksProxyUriPath": "/example-prefix/kms/xks/v1",
  "XksProxyAuthenticationCredential": {
    "AccessKeyId": "ABCDE12345670EXAMPLE",
    "RawSecretAccessKey": "0000EXAMPLEFA5FT0mCc3DrGUe2sti527BitkQ0Zr9MO9+vE="
  }
}
```

只有在 AWS KMS 主控台中建立或編輯外部金鑰存放區時，您才能上傳代理組態檔案。您無法將其與CreateCustomKeyStore或UpdateCustomKeyStore作業搭配使用，但可以使用 Proxy 組態檔案中的值來確保參數值正確無誤。

建立外部金鑰存放區 (主控台)

在建立外部金鑰存放區之前，請檢閱 規劃外部金鑰存放區，選擇代理連接類型，並確保您已建立並設定所有必要元件。如果您需要尋找任何必要值的說明，請參閱外部金鑰存放區代理或金鑰管理軟體的文件。

> **Note**
>
> 當您在 AWS Management Console 中建立外部金鑰存放區時，您可以上傳以 JSON 為基礎的代理組態檔案，其中包含代理 URI 路徑和代理身分驗證憑證的值。某些代理會為您產生此檔案。這不是必要的。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。

3. 在導覽窗格中，依次選擇 Custom key stores (自訂金鑰存放區)、External key stores (外部金鑰存放區)。
4. 選擇 Create external key store (建立外部金鑰存放區)。
5. 輸入外部金鑰存放區的易記名稱。該名稱在帳戶的所有外部金鑰存放區中必須唯一。

   > **Important**
   >
   > 請勿在此欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，此欄位可能會以純文字顯示。

6. 選擇您的代理連接類型。

   您的代理連接選項會決定外部金鑰存放區代理所需的元件。如需進行此選擇的說明，請參閱 選擇代理連接選項。

7. 選擇或輸入此外部金鑰存放區的 VPC 端點服務名稱。只有當外部金鑰存放區代理連接類型為 VPC endpoint service (VPC 端點服務) 時，此步驟才會出現。

   VPC 端點服務及其 VPC 必須滿足外部金鑰存放區的要求。如需詳細資訊，請參閱 the section called “備妥先決條件”。

8. 輸入您的代理 URI 端點。通訊協定必須為 HTTPS。AWS KMS 在連接埠 443 上進行通訊。請勿在代理 URI 端點值中指定連接埠。

   如果 AWS KMS 識別出您在上一個步驟中指定的 VPC 端點服務，則其會為您完成此欄位。

   若為公有端點連接，請輸入公開可用的端點 URI。若為 VPC 端點連接，請輸入 `https://`，後跟 VPC 端點服務的私有 DNS 名稱。

9. 若要輸入代理 URI 路徑字首和代理身分驗證憑證的值，請上傳代理組態檔案，或手動輸入值。
   - 如果您有可選的代理組態檔案，其中包含代理 URI 路徑和代理身分驗證憑證的值，則請選擇 Upload configuration file (上傳組態檔案)。遵循步驟上傳檔案。

     上傳檔案時，主控台會在可編輯欄位中顯示檔案中的值。您可以立即變更值，或在建立外部金鑰存放區後編輯這些值。

     若要顯示私密存取金鑰的值，請選擇 Show secret access key (顯示私密存取金鑰)。
   - 如果您沒有代理組態檔案，則可以手動輸入代理 URI 路徑和代理身分驗證憑證。

     a. 如果您沒有代理組態檔案，則可以手動輸入代理 URI。主控台會提供必要的 /kms/xks/v1 值。

如果您的代理 URI 路徑包含可選字首，例如 /*example-prefix*/kms/xks/v1 中的 example-prefix，則請在 Proxy URI path prefix (代理 URI 路徑字首) 欄位中輸入該字首。否則，請將欄位保留空白。

b. 如果您沒有代理組態檔案，則可以手動輸入代理身分驗證憑證。存取金鑰 ID 和私密存取金鑰都是必要項目。

- 在 Proxy credential: Access key ID (代理憑證：存取金鑰 ID) 中，輸入代理身分驗證憑證的存取金鑰 ID。存取金鑰 ID 可識別私密存取金鑰。
- 在 Proxy credential: Secret access key (代理憑證：私密存取金鑰) 中，輸入代理身分驗證憑證的私密存取金鑰。

若要顯示私密存取金鑰的值，請選擇 Show secret access key (顯示私密存取金鑰)。

此程序不會設定或變更您在外部金鑰存放區代理上建立的身分驗證憑證。其只是將這些值與您的外部金鑰存放區相關聯。如需有關設定、變更及輪換代理身分驗證憑證的資訊，請參閱外部金鑰存放區代理或金鑰管理軟體的文件。

如果您的代理身分驗證憑證發生變更，則請編輯外部金鑰存放區的憑證設定。

10. 選擇 Create external key store (建立外部金鑰存放區)。

當程序成功時，新的外部金鑰存放區會出現在帳戶和區域的外部金鑰存放區清單中。如果不成功，則會出現錯誤訊息來描述問題，並提供如何修正的說明。如果您需要更多協助，請參閱CreateKey 外部鍵錯誤。

下一步：不會自動連接新的外部金鑰存放區。在外部金鑰存放區中建立 AWS KMS keys 之前，必須將外部金鑰存放區連接至外部金鑰存放區代理。

建立外部金鑰存放區 (API)

您可以使用此CreateCustomKeyStore作業建立新的外部金鑰存放區。如需尋找必要參數值的說明，請參閱外部金鑰存放區代理或金鑰管理軟體的文件。

> **Tip**
>
> 在使用 CreateCustomKeyStore 操作時，您無法上傳代理組態檔案。但是，您可以使用代理組態檔案中的值來確保參數值正確無誤。

若要建立外部金鑰存放區，CreateCustomKeyStore 操作需要下列參數值。

- CustomKeyStoreName – 外部金鑰存放區在帳戶中唯一的易用名稱。

> **Important**
>
> 請勿在此欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，此欄位可能會以純文字顯示。

- CustomKeyStoreType – 指定 EXTERNAL_KEY_STORE。
- XksProxyConnectivity – 指定 PUBLIC_ENDPOINT 或 VPC_ENDPOINT_SERVICE。
- XksProxyAuthenticationCredential – 指定存取金鑰 ID 和私密存取金鑰 ID。
- XksProxyUriEndpoint – AWS KMS 用於與外部金鑰存放區代理通訊的端點。
- XksProxyUriPath – 連接代理 API 的代理內路徑。
- XksProxyVpcEndpointServiceName – 僅當 XksProxyConnectivity 值為 VPC_ENDPOINT_SERVICE 時才需要。

> **Note**
>
> 如果您使用 AWS CLI 1.0 版，則請先執行下列命令，然後再指定具有 HTTP 或 HTTPS 值的參數，例如 XksProxyUriEndpoint 參數。
>
> ```
> aws configure set cli_follow_urlparam false
> ```
>
> 否則，AWS CLI 1.0 版會將參數值取代為在該 URI 地址找到的內容，這會導致下列錯誤：
>
> ```
> Error parsing parameter '--xks-proxy-uri-endpoint': Unable to retrieve
> https:// : received non 200 status code of 404
> ```

下列範例使用虛構值。在執行命令之前，請將其取代為外部金鑰存放區的有效值。

建立具有公有端點連接的外部金鑰存放區。

```
$ aws kms create-custom-key-store
        --custom-key-store-name ExampleExternalKeyStorePublic \
```

```
        --custom-key-store-type EXTERNAL_KEY_STORE \
        --xks-proxy-connectivity PUBLIC_ENDPOINT \
        --xks-proxy-uri-endpoint https://myproxy.xks.example.com \
        --xks-proxy-uri-path /kms/xks/v1 \
        --xks-proxy-authentication-credential
 AccessKeyId=<value>,RawSecretAccessKey=<value>
```

建立具有 VPC 端點服務連接的外部金鑰存放區。

```
$ aws kms create-custom-key-store
        --custom-key-store-name ExampleExternalKeyStoreVPC \
        --custom-key-store-type EXTERNAL_KEY_STORE \
        --xks-proxy-connectivity VPC_ENDPOINT_SERVICE \
        --xks-proxy-vpc-endpoint-service-name com.amazonaws.vpce.us-east-1.vpce-svc-
example \
        --xks-proxy-uri-endpoint https://myproxy-private.xks.example.com \
        --xks-proxy-uri-path /kms/xks/v1 \
        --xks-proxy-authentication-credential
 AccessKeyId=<value>,RawSecretAccessKey=<value>
```

當操作成功時，CreateCustomKeyStore 會傳回自訂金鑰存放區 ID，如下回應範例所示。

```
{
    "CustomKeyStoreId": cks-1234567890abcdef0
}
```

如果操作失敗，請修正例外狀況所指出的錯誤，然後重試。如需其他說明，請參閱外部金鑰存放區故障診斷。

下一步：若要使用外部金鑰存放區，請將其連接至其外部金鑰存放區代理。

編輯外部金鑰存放區屬性

您可以編輯現有外部金鑰存放區的所選屬性。

您可以在連接或中斷連接外部金鑰存放區時編輯某些屬性。對於其他屬性，您必須先中斷外部金鑰存放區與其外部金鑰存放區代理的連接。外部金鑰存放區的連接狀態必須是 DISCONNECTED。當外部金鑰存放區中斷連接時，您可以管理金鑰存放區和其 KMS 金鑰，但無法在外部金鑰存放區中建立或使用 KMS 金鑰。若要尋找外部金鑰存放區的連線狀態，請使用DescribeCustomKeyStores作業或參閱外部金鑰存放區詳細資料頁面上的 [一般組態] 區段。

在更新外部金鑰存放區的屬性之前，請使用新值將GetHealthStatus要求AWS KMS傳送至外部金鑰存放區 Proxy。如果請求成功，表示您可以使用更新後的屬性值連接至外部金鑰存放區代理並進行身分驗證。如果請求失敗，編輯操作將失敗，並顯示識別錯誤的例外狀況。

當編輯操作完成時，外部金鑰存放區的更新屬性值會顯示在 AWS KMS 主控台和 `DescribeCustomKeyStores` 回應中。但是，變更需要 5 分鐘才能完全生效。

如果您在 AWS KMS 主控台中編輯外部金鑰存放區，則可以選擇上傳基於 JSON 的代理組態文件，該文件指定代理 URI 路徑和代理身分驗證憑證。某些外部金鑰存放區代理會為您產生此檔案。如需詳細資訊，請參閱外部金鑰存放區代理或外部金鑰管理器的文件。

> **Warning**
>
> 更新的屬性值必須將外部金鑰存放區連接至與先前值相同的外部金鑰管理器的代理，或使用相同密碼編譯金鑰之外部金鑰管理器的備份或快照。如果您的外部金鑰存放區永久無法存取與其 KMS 金鑰相關聯的外部金鑰，則使用這些外部金鑰加密的密文將無法復原。特別是，變更外部金鑰存放區的代理連接可能會阻止 AWS KMS 存取外部金鑰。

> **Tip**
>
> 有些外部金鑰管理器會提供更簡單的方法來編輯外部金鑰存放區屬性。如需詳細資訊，請參閱外部金鑰管理器文件。

您可以變更外部金鑰存放區的下列屬性。

可編輯的外部金鑰存放區屬性	任何連接狀態	需要已中斷連接狀態
自訂金鑰存放區名稱 自訂金鑰存放區的必要易記名稱。 **Important** 請勿在此欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，此欄位可能會以純文字顯示。	✓	

可編輯的外部金鑰存放區屬性	任何連接狀態	需要已中斷連接狀態
代理伺服器驗證憑證 (XksProxyAuthenticationCredential) (您必須同時指定存取金鑰 ID 和私密存取金鑰，即使您只變更一個元素。)	✓	
代理伺服器 URI 路徑 (XksProxyUriPath)	✓	
代理伺服器連線 (XksProxyConnectivity) (您也必須更新代理 URI 端點。如果您要變更為 VPC 端點服務連接，則必須指定代理 VPC 端點服務名稱。)		✓
代理 URI 端點 (XksProxyUriEndpoint) 如果變更代理端點 URI，則您可能也需要變更相關的 TLS 憑證。		✓
代理 VPC 端端點服務名稱 () XksProxyVpcEndpointServiceName (VPC 端點服務連接需要此欄位)		✓

主題

- 編輯外部金鑰存放區 (主控台)
- 編輯外部金鑰存放區 (API)

編輯外部金鑰存放區 (主控台)

編輯金鑰存放區時，您可以變更任何可編輯的值。某些變更需要將外部金鑰存放區與其外部金鑰存放區代理中斷連接。

如果您正在編輯代理 URI 路徑或代理身分驗證憑證，則可以輸入新值或上傳包含新值的外部金鑰存放區代理組態檔案。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，依次選擇 Custom key stores (自訂金鑰存放區)、External key stores (外部金鑰存放區)。
4. 選擇您想要編輯的外部金鑰存放區的資料列。
5. 如果需要，將外部金鑰存放區與其外部金鑰存放區代理中斷連接。從 Key store actions (金鑰存放區動作) 選單中，選擇 Disconnect (中斷連接)。
6. 從 Key store actions (金鑰存放區動作) 選單中，選擇 Edit (編輯)。
7. 變更一個或多個可編輯的外部金鑰存放區屬性。您也可以上傳外部金鑰存放區代理組態檔案，其中包含代理 URI 路徑和代理身分驗證憑證的值。即使檔案中指定的某些值尚未變更，您仍可使用代理組態檔案。
8. 選擇 Update external key store (更新外部金鑰存放區)。
9. 檢閱警告，如果您決定繼續，請確認警告，然後選擇 Update external key store (更新外部金鑰存放區)。

   當程序成功時，會出現訊息描述您編輯的屬性。當操作失敗時，會出現錯誤訊息，其中描述問題並提供如何修正的協助。
10. 如有必要，請重新連接外部金鑰存放區。從 Key store actions (金鑰存放區動作) 選單中，選擇 Connect (連接)。

    您可以將外部金鑰存放區保持中斷連接。中斷連接時，您無法在外部金鑰存放區中建立 KMS 金鑰，並且無法在密碼編譯操作中使用外部金鑰存放區中的 KMS 金鑰。

編輯外部金鑰存放區 (API)

若要變更外部金鑰存放區的屬性，請使用此UpdateCustomKeyStore作業。您可以在相同操作中變更外部金鑰存放區的多個屬性。如果操作成功，則 AWS KMS 傳回 HTTP 200 回應和不帶屬性的 JSON 物件。

使用 `CustomKeyStoreId` 參數可識別外部金鑰存放區。使用其他參數來變更屬性。您無法將代理組態檔案用於 `UpdateCustomKeyStore` 操作。代理組態檔案僅受 AWS KMS 主控台支援。不過，您可以使用代理組態檔案來協助您判斷外部金鑰存放區代理的正確參數值。

本節中的範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

在開始之前，如果需要，將外部金鑰存放區與其外部金鑰存放區代理中斷連接。更新之後，如果需要，您可以將外部金鑰存放區重新連接至其外部金鑰存放區代理。您可以將外部金鑰存放區保持在中斷連接狀態，但必須先重新將其連接，才能在金鑰存放區中建立新的 KMS 金鑰或使用金鑰存放區中的現有 KMS 金鑰進行密碼編譯操作。

> **Note**
>
> 如果您使用 AWS CLI 1.0 版，則請先執行下列命令，然後再指定具有 HTTP 或 HTTPS 值的參數，例如 XksProxyUriEndpoint 參數。
>
> ```
> aws configure set cli_follow_urlparam false
> ```
>
> 否則，AWS CLI 1.0 版會將參數值取代為在該 URI 地址找到的內容，這會導致下列錯誤：
>
> ```
> Error parsing parameter '--xks-proxy-uri-endpoint': Unable to retrieve
> https:// : received non 200 status code of 404
> ```

變更外部金鑰存放區的名稱

第一個範例使用UpdateCustomKeyStore作業將外部金鑰存放區的易記名稱變更為XksKeyStore。此命令使用 CustomKeyStoreId 參數來識別自訂金鑰存放區和 CustomKeyStoreName 來指定自訂金鑰存放區的新名稱。將所有範例值取代為外部金鑰存放區的實際值。

```
$ aws kms update-custom-key-store --custom-key-store-id cks-1234567890abcdef0 --new-
custom-key-store-name XksKeyStore
```

變更代理身分驗證憑證

下列範例會更新 AWS KMS 用來驗證外部金鑰存放區代理的代理身分驗證憑證。如果在代理上輪換憑證，則可以使用類似這樣的命令來更新憑證。

請先更新外部金鑰存放區代理上的憑證。然後使用此功能將變更報告給 AWS KMS。(您的代理將短暫支援舊憑證和新憑證，因此您有時間在 AWS KMS 中更新憑證。)

您必須在憑證中同時指定存取金鑰 ID 和私密存取金鑰，即使只變更一個值。

前兩個命令設定變數來保留憑證值。UpdateCustomKeyStore 操作使用 CustomKeyStoreId 參數識別外部金鑰存放區。其會使用 XksProxyAuthenticationCredential 參數及其 AccessKeyId 和 RawSecretAccessKey 欄位來指定新的憑證。將所有範例值取代為外部金鑰存放區的實際值。

```
$ accessKeyID=access key id
$ secretAccessKey=secret access key

$ aws kms update-custom-key-store --custom-key-store-id cks-1234567890abcdef0 \
       --xks-proxy-authentication-credential \
           AccessKeyId=$accessKeyId,RawSecretAccessKey=$secretAccessKey
```

變更代理 URI 路徑

下列範例會更新代理 URI 路徑 (XksProxyUriPath)。代理 URI 端點和代理 URI 路徑的組合在 AWS 帳戶 和區域中必須是唯一的。將所有範例值取代為外部金鑰存放區的實際值。

```
$ aws kms update-custom-key-store --custom-key-store-id cks-1234567890abcdef0 \
           --xks-proxy-uri-path /kms/xks/v1
```

變更為 VPC 端點服務連接

下列範例會使用此UpdateCustomKeyStore作業將外部金鑰存放區 Proxy 連線類型變更為VPC_ENDPOINT_SERVICE。若要進行此變更，您必須指定 VPC 端點服務連接的必要值，包括 VPC 端點服務名稱 (XksProxyVpcEndpointServiceName) 和包含 VPC 端點服務之私有 DNS 名稱的代理 URI 端點 (XksProxyUriEndpoint) 值。將所有範例值取代為外部金鑰存放區的實際值。

```
$ aws kms update-custom-key-store --custom-key-store-id cks-1234567890abcdef0 \
           --xks-proxy-connectivity "VPC_ENDPOINT_SERVICE" \
           --xks-proxy-uri-endpoint https://myproxy-private.xks.example.com \
           --xks-proxy-vpc-endpoint-service-name com.amazonaws.vpce.us-east-1.vpce-
svc-example
```

變更為公有端點連接

下列範例會將外部金鑰存放區代理連接類型變更為 PUBLIC_ENDPOINT。當您進行此變更時，必須更新代理 URI 端點 (XksProxyUriEndpoint) 值。將所有範例值取代為外部金鑰存放區的實際值。

> **Note**
>
> 與公有端點連接相比，VPC 端點連接可提供更高的安全性。在變更為公有端點連接之前，請考慮其他選項，包括在內部部署中尋找外部金鑰存放區代理，以及僅將 VPC 用於通訊。

```
$ aws kms update-custom-key-store --custom-key-store-id cks-1234567890abcdef0 \
```

```
        --xks-proxy-connectivity "PUBLIC_ENDPOINT" \
        --xks-proxy-uri-endpoint https://myproxy.xks.example.com
```

檢視外部金鑰存放區

您可以使用AWS KMS控制台或使用DescribeCustomKeyStores操作來檢視每個帳戶和區域中的外部金鑰存放區。

檢視外部金鑰存放區時，您可以查看下列項目：

- 有關金鑰存放區的基本資訊，包括其易記名稱、ID、金鑰存放區類型和建立日期。
- 外部金鑰存放區代理的組態資訊，包括連接類型、代理 URI 端點和路徑，以及當前代理身分驗證憑證的存取金鑰 ID。
- 如果外部金鑰存放區代理使用 VPC 端點服務連接，則主控台會顯示 VPC 端點服務的名稱。
- 當前的連接狀態。

> **Note**
>
> 連接狀態值 Disconnected (已中斷連接) 指示外部金鑰存放區從未連接，或它是特意從其外部金鑰存放區代理中斷連接。不過，如果嘗試在已連接的外部金鑰存放區中使用 KMS 金鑰失敗，可能表示外部金鑰存放區或其代理發生問題。如需協助，請參閱 外部金鑰存放區連接錯誤。

- 帶有 Amazon CloudWatch 指標圖形的「監控」部分，旨在協助您偵測和解決外部金鑰存放區的問題。如需解譯圖形、在規劃和疑難排解中使用這些圖形，以及根據圖表中的量度建立 CloudWatch 警示的說明，請參閱監控外部金鑰存放區。

另請參閱:

- 檢視外部金鑰存放區中的 KMS 金鑰
- 使用 AWS CloudTrail 記錄 AWS KMS API 呼叫

主題

- 外部金鑰存放區屬性
- 檢視外部金鑰存放區 (主控台)
- 檢視外部金鑰存放區 (API)

外部金鑰存放區屬性

外部金鑰存放區的下列屬性會顯示在AWS KMS主控台和DescribeCustomKeyStores回應中。

自訂金鑰存放區屬性

下列值會顯示在每個自訂金鑰存放區之詳細資料頁面的 General configuration (一般組態) 區段中。這些屬性適用於所有自訂金鑰存放區，包括 AWS CloudHSM 金鑰存放區和外部金鑰存放區。

自訂金鑰存放區 ID

AWS KMS 指派給自訂金鑰存放區的唯一 ID。

自訂金鑰存放區名稱

您在建立自訂金鑰存放區時指派給它的易記名稱。您可隨時變更此值。

自訂金鑰存放區類型

自訂金鑰存放區的類型。有效值為 AWS CloudHSM (`AWS_CLOUDHSM`) 或外部金鑰存放區 (`EXTERNAL_KEY_STORE`)。您無法在建立自訂金鑰存放區之後變更類型。

建立日期

建立自訂金鑰存放區的日期。此日期顯示為 AWS 區域 的本地時間。

連線狀態

指示自訂金鑰存放區是否已連接至其備份金鑰存放區。只有當自訂金鑰存放區從未連接至其備份金鑰存放區，或已故意中斷連接時，連接狀態才為 `DISCONNECTED`。如需詳細資訊，請參閱 the section called “連線狀態”。

外部金鑰存放區組態屬性

下列值會顯示在每個外部金鑰儲存庫詳細資訊頁面的「外部金鑰儲存區代理主機組態」段落中，以及DescribeCustomKeyStores回應的`XksProxyConfiguration`元素中。如需每個欄位的詳細說明，包括唯一性要求以及決定每個欄位正確值的說明，請參閱「建立外部金鑰存放區」主題中的 the section called “備妥先決條件”。

代理連接

指示外部金鑰存放區是使用公有端點連接還是 VPC 端點服務連接。

代理 URI 端點

AWS KMS 用於連接至外部金鑰存放區代理的端點。

代理 URI 路徑

來自代理 URI 端點的路徑，AWS KMS 在其中傳送代理 API 請求。

代理憑證：存取金鑰 ID

您在外部金鑰存放區代理上建立的代理身分驗證憑證的一部分。存取金鑰 ID 可識別憑證中的私密存取金鑰。

AWS KMS 使用 Sigv4 簽署程序和代理身分驗證憑證，將其請求簽署到外部金鑰存放區代理。簽章中的憑證允許外部金鑰存放區代理代表您從 AWS KMS 中驗證請求。

VPC 端點服務名稱

支援外部金鑰存放區的 Amazon VPC 端點服務名稱。只有當外部金鑰存放區使用 VPC 端點服務連接時，才會顯示此值。您可以在 VPC 中找到外部金鑰存放區代理，或使用 VPC 端點服務與外部金鑰存放區代理進行安全通訊。

檢視外部金鑰存放區 (主控台)

若要檢視指定帳戶和區域中的外部金鑰存放區，請使用下列程序。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，依次選擇 Custom key stores (自訂金鑰存放區)、External key stores (外部金鑰存放區)。
4. 若要檢視有關外部金鑰存放區的詳細資訊，請選擇金鑰存放區名稱。

檢視外部金鑰存放區 (API)

若要檢視外部金鑰存放區，請使用此DescribeCustomKeyStores作業。在預設情況下，此操作會傳回帳戶和區域中的所有自訂金鑰存放區。但是，您可以使用 `CustomKeyStoreId` 或 `CustomKeyStoreName` 參數 (但不能同時使用) 來限制對特定自訂金鑰存放區的輸出。

若為自訂金鑰存放區，則輸出包含自訂金鑰存放區 ID、名稱和類型以及金鑰存放區的連接狀態。如果連接狀態為 `FAILED`，則輸出也會包含描述錯誤原因的 `ConnectionErrorCode`。如需解譯外部金鑰存放區的 `ConnectionErrorCode` 說明，請參閱 外部金鑰存放區的連接錯誤代碼。

若為外部金鑰存放區，輸出還包括 XksProxyConfiguration 元素。此元素包括連接類型、代理 URI 端點、代理 URI 路徑以及代理身分驗證憑證的存取金鑰 ID。

本節中的範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

例如，以下命令會傳回帳戶和區域中的所有自訂金鑰存放區。您可以使用 Limit 和 Marker 參數來切換輸出中的自訂金鑰存放區頁面。

```
$ aws kms describe-custom-key-stores
```

以下命令使用 CustomKeyStoreName 參數來取得僅具有 ExampleXksPublic 易記名稱的範例外部金鑰存放區。此範例金鑰存放區使用公有端點連接。它連接到其外部金鑰存放區代理。

```
$ aws kms describe-custom-key-stores --custom-key-store-name ExampleXksPublic
{
    "CustomKeyStores": [
    {
      "CustomKeyStoreId": "cks-1234567890abcdef0",
      "CustomKeyStoreName": "ExampleXksPublic",
      "ConnectionState": "CONNECTED",
      "CreationDate": "2022-12-14T20:17:36.419000+00:00",
      "CustomKeyStoreType": "EXTERNAL_KEY_STORE",
      "XksProxyConfiguration": {
        "AccessKeyId": "ABCDE12345670EXAMPLE",
        "Connectivity": "PUBLIC_ENDPOINT",
        "UriEndpoint": "https://xks.example.com:6443",
        "UriPath": "/example/prefix/kms/xks/v1"
      }
    }
  ]
}
```

下列命令取得具有 VPC 端點服務連接的範例外部金鑰存放區。在此範例中，外部金鑰存放區會連接至其外部金鑰存放區代理。

```
$ aws kms describe-custom-key-stores --custom-key-store-name ExampleXksVpc
{
    "CustomKeyStores": [
    {
      "CustomKeyStoreId": "cks-9876543210fedcba9",
```

```
      "CustomKeyStoreName": "ExampleXksVpc",
      "ConnectionState": "CONNECTED",
      "CreationDate": "2022-12-13T18:34:10.675000+00:00",
      "CustomKeyStoreType": "EXTERNAL_KEY_STORE",
      "XksProxyConfiguration": {
        "AccessKeyId": "ABCDE98765432EXAMPLE",
        "Connectivity": "VPC_ENDPOINT_SERVICE",
        "UriEndpoint": "https://example-proxy-uri-endpoint-vpc",
        "UriPath": "/example/prefix/kms/xks/v1",
        "VpcEndpointServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-example"
      }
    }
  ]
}
```

ConnectionState 的 Disconnected 值指示外部金鑰存放區從未連接，或它是特意從其外部金鑰存放區代理中斷連接。不過，如果嘗試在已連接的外部金鑰存放區中使用 KMS 金鑰失敗，可能表示外部金鑰存放區代理或其他外部元件發生問題。

如果外部金鑰存放區的 ConnectionState 為 FAILED，則 DescribeCustomKeyStores 回應會包含一個 ConnectionErrorCode 元素，解釋錯誤的原因。

例如，在以下輸出中，XKS_PROXY_TIMED_OUT 值表示 AWS KMS 可以連接到外部金鑰存放區代理，但連接失敗，因為外部金鑰存放區代理在指定的時間內沒有回應 AWS KMS。如果您重複看到此連接錯誤代碼，則請通知您的外部金鑰存放區代理廠商。如需此錯誤和其他連接錯誤失敗的協助，請參閱外部金鑰存放區故障診斷。

```
$ aws kms describe-custom-key-stores --custom-key-store-name ExampleXksVpc
{
    "CustomKeyStores": [
    {
      "CustomKeyStoreId": "cks-9876543210fedcba9",
      "CustomKeyStoreName": "ExampleXksVpc",
      "ConnectionState": "FAILED",
      "ConnectionErrorCode": "XKS_PROXY_TIMED_OUT",
      "CreationDate": "2022-12-13T18:34:10.675000+00:00",
      "CustomKeyStoreType": "EXTERNAL_KEY_STORE",
      "XksProxyConfiguration": {
        "AccessKeyId": "ABCDE98765432EXAMPLE",
        "Connectivity": "VPC_ENDPOINT_SERVICE",
        "UriEndpoint": "https://example-proxy-uri-endpoint-vpc",
        "UriPath": "/example/prefix/kms/xks/v1",
```

```
        "VpcEndpointServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-example"
      }
    }
  ]
}
```

監控外部金鑰存放區

AWS KMS收集與外部金鑰存放區的每次互動的指標，並將其發佈到您的 CloudWatch 帳戶中。這些指標用於在每個外部金鑰存放區之詳細資訊頁面的監控區段中產生圖表。下列主題詳細說明如何使用圖表來識別會影響外部金鑰存放區的操作和組態問題，並進行疑難排解。我們建議您使用 CloudWatch 指標來設定警示，以便在外部金鑰存放區未如預期般執行時通知您。如需詳細資訊，請參閱使用 Amazon 進行監控 CloudWatch。

主題

- 檢視圖表
- 解釋圖表
- 設定警示

檢視圖表

您可以用不同的詳細資料層級檢視圖表。依預設，每個圖表使用三小時的時間範圍和五分鐘的彙總期間。您可以在主控台內調整圖表檢視，但是當外部金鑰存放區詳細資訊頁面關閉或瀏覽器重新整理時，您的變更將還原為預設設定。如需 Amazon CloudWatch 術語的說明，請參閱 Amazon CloudWatch 概念。

檢視資料點詳細資訊

每個圖表中的資料是依 AWS KMS 指標收集的。若要檢視有關特定資料點的詳細資訊，請將滑鼠暫停在折線圖上的資料點上。這會顯示一個彈出式視窗，其中包含有關衍生出該圖表之指標的詳細資訊。每個清單項目都會顯示在該資料點記錄的維度值。如果該資料點的維度值沒有可用的指標資料，彈出式視窗會顯示空值 (–)。有些圖表會記錄單一資料點的多個維度和值。其他圖表 (例如可靠性圖表) 會使用指標收集的資料來計算唯一值。每個清單項目都與不同的折線圖顏色相關聯。

修改時間範圍

若要修改時間範圍，請選取監控區段右上角其中一種預先定義的時間範圍。預先定義的時間範圍為 1 小時到 1 週 (1h (1 小時)、3h (3 小時)、12h (12 小時)、1d (1 天)、3d (3 天) 或 1w (1 週))。這會調整

所有圖表的時間範圍。如果您想要檢視不同時間範圍內的特定圖形，或想要設定自訂時間範圍，請放大圖形或在 Amazon CloudWatch 主控台中檢視圖表。

放大圖表

您可以使用迷你地圖縮放功能，將焦點放在折線圖的某個區段以及圖表的某些部分上，而無需在放大和縮小檢視之間進行變更。例如，您可以使用迷你地圖縮放功能，將焦點放在圖表中的峰值上，以便在同一時間軸中將尖峰與監控區段中的其他圖表進行比較。

1. 選擇並拖曳要聚焦的圖表區域，然後放開拖曳。
2. 若要重設縮放，請選擇 Reset zoom (重設縮放) 圖示，這看起來像一個帶有減號 (-) 符號的放大鏡。

放大圖表

若要放大圖表，請選取單個圖表右上角的選單圖示，然後選擇 Enlarge (放大)。您也可以選取當您將游標停留在圖表上時顯示在選單圖示旁邊的放大圖示。

放大圖表可讓您透過指定不同的時段、自訂時間範圍或重新整理間隔來進一步修改圖表檢視。當您關閉放大的檢視時，這些變更將恢復為預設設定。

修改期間

1. 選擇 Period options (期間選項) 功能表。依預設，此選單會顯示值：5 分鐘。
2. 選擇一個期間，預先定義的期間從 1 秒到 30 天不等。

   例如，您可以選擇一分鐘檢視，這在疑難排解時非常有用。或者，選擇較不精細的一小時檢視。當檢視更廣泛的時間範圍 (例如 3 天) 時，您可以看到隨時間變化的趨勢。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的期間。

修改時間範圍或時區

1. 選取其中一個預先定義的時間範圍，範圍從 1 小時到 1 週不等 (1h (1 小時)、3h (3 小時)、12h (12 小時)、1d (1 天)、3d (3 天) 1w (1 週))。或者，您也可以選擇 Custom (自訂) 來設定您自己的時間範圍。
2. 選擇 Custom (自訂)。

   a. Time range: (時間範圍：) 選取方塊左上角的 Absolute (絕對值) 索引標籤。使用行事曆選擇器或文字欄位方塊來指定時間範圍。

   b. Time zone: (時區：) 選擇方塊右上角的下拉式選單。您可以將時區變更為 UTC 或者 Local time zone (本機時區)。

3. 指定時間範圍後，選擇 Apply (套用)。

修改圖表中資料重新整理的頻率

1. 選擇右上角的 Refresh options (重新整理選項) 選單。
2. 選擇重新整理間隔 (Off (關閉)、10 Seconds (10 秒)、1 Minute (1 分鐘)、2 Minutes (2 分鐘)、5 Minutes (5 分鐘) 或 15 Minutes (15 分鐘))。

在 Amazon CloudWatch 控制台中查看圖形

監控部分中的圖形衍生自AWS KMS發佈到 Amazon 的預先定義指標 CloudWatch。您可以在 CloudWatch 控制台中打開它們並將其保存到 CloudWatch 儀表板中。如果您有多個外部金鑰存放區，您可以在中開啟各自的圖形， CloudWatch 並將其儲存到單一儀表板，以比較其健康狀態和使用情況。

添加到 CloudWatch 儀表板

選取右上角的「新增至儀表板」，將所有圖形新增至 Amazon CloudWatch 儀表板。您可以選取現有的儀表板或建立新的儀表板。如需使用此儀表板建立圖形和警示的自訂檢視的相關資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用 Amazon CloudWatch 儀表板。

在 CloudWatch 量度中檢視

選取個別圖表右上角的功能表圖示，然後選擇在指標中檢視以在 Amazon CloudWatch 主控台中檢視此圖表。在 CloudWatch 主控台中，您可以將此單一圖形新增至儀表板，並修改時間範圍、期間和重新整理間隔。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的繪圖指標。

解釋圖表

AWS KMS 提供數個圖表來監控 AWS KMS 主控台內外部金鑰存放區的運作狀態。這些圖表會自動設定並衍生自 AWS KMS 指標。

作為您對外部金鑰存放區和外部金鑰進行呼叫的一部分來收集圖表資料。您可能會在進行任何呼叫的時間範圍內看到資料填入圖表，此資料來自 AWS KMS 代表您進行的定期 `GetHealthStatus` 呼叫，以檢查外部金鑰存放區代理和外部金鑰管理器的狀態。如果您的圖表顯示 No data available (沒有資料可用) 訊息，則在該時間範圍內沒有記錄任何呼叫，或者您的外部金鑰存放區處於 `DISCONNECTED` 狀態。透過將檢視調整為更寬的時間範圍，可確定外部金鑰存放區中斷連接的時間。

主題

- 請求總數
- 可靠性

- Latency (延遲)
- 前 5 個例外狀況
- 憑證到期天數

請求總數

在指定時間範圍內，針對特定外部金鑰存放區接收的 AWS KMS 請求總數。使用此圖表來確定是否有限流風險。

AWS KMS 建議您的外部金鑰管理器每秒最多處理 1800 個密碼編譯操作請求。如果五分鐘內的呼叫達到 540,000 個，則有限流風險。

您可以監控 AWS KMS 使用 ExternalKeyStoreThrottle 指標來限流的外部金鑰存放區中 KMS 金鑰的密碼編譯操作請求數目。

如果您經常收到 KMSInvalidStateException 錯誤，並且訊息說明「因為請求率非常高」而拒絕請求，則可能表示您的外部金鑰管理器或外部金鑰存放區代理無法跟上目前的請求率。如果可能，請降低您的請求率。您也可以考慮請求減少自訂金鑰存放區請求配額值。減少此配額值可能會增加限流，但表示在傳送至外部金鑰存放區代理或外部金鑰管理器之前，AWS KMS 會快速拒絕多餘的請求。若要請求減少配額，請造訪 AWS Support 中心並建立案例。

總請求圖表衍生自 XksProxyErrors 指標，其會收集 AWS KMS 從外部金鑰存放區代理接收的成功與失敗回應的相關資料。當您檢視特定資料點時，彈出式視窗會顯示 CustomKeyStoreId 維度值，以及在該資料點記錄的 AWS KMS 請求總數。CustomKeyStoreId 將永遠是相同的。

可靠性

外部金鑰存放區代理傳回成功回應或不可重試錯誤的 AWS KMS 請求百分比。使用此圖表來評估外部金鑰存放區代理的操作運作狀態。

當圖表顯示的值小於 100% 時，表示代理未回應或回應時出現可重試錯誤。這可能表示網路出現問題、外部金鑰存放區代理或外部金鑰管理器運行緩慢或實作錯誤。

如果請求包含錯誤的憑證並且您的代理回應時出現 AuthenticationFailedException，則圖表仍會顯示 100% 的可靠性，因為代理在外部金鑰存放區代理請求中發現錯誤值，因此預期會發生失敗。如果可靠性圖表的百分比為 100%，則外部金鑰存放區代理會按預期回應。如果圖表顯示的值小於 100%，則代理回應時會出現可重試錯誤或逾時。例如，如果由於請求率非常高，代理回應時出現 ThrottlingException，則會顯示較低的可靠性百分比，因為代理無法識別導致它失敗的請求中的特定問題。這是因為可重試的錯誤很可能是暫時性的問題，可以透過重試請求來解決。

下列錯誤回應會降低可靠性百分比。您可以使用 前 5 個例外狀況 圖表和 XksProxyErrors 指標進一步監控代理傳回每個可重試錯誤的頻率。

- `InternalException`
- `DependencyTimeoutException`
- `ThrottlingException`
- `XksProxyUnreachableException`

可靠性圖表衍生自 XksProxyErrors 指標，它會收集 AWS KMS 從外部金鑰存放區代理接收的成功與失敗回應的相關資料。只有當回應的 `ErrorType` 值為 `Retryable` 時，才會降低可靠性百分比。當您檢視特定資料點時，彈出式視窗會顯示 `CustomKeyStoreId` 維度值，以及在該資料點記錄的 AWS KMS 請求可靠性百分比。`CustomKeyStoreId` 將永遠是相同的。

我們建議您使用XksProxyErrors指標建立 CloudWatch 警示，在一分鐘內記錄超過五個可重試的錯誤時，通知您潛在的網路問題。如需詳細資訊，請參閱 為可重試的錯誤創建 Amazon CloudWatch 警報。

Latency (延遲)

外部金鑰存放區代理回應 AWS KMS 請求所需的毫秒數。使用此圖表可評估外部金鑰存放區代理和外部金鑰管理器的效能。

AWS KMS 預期外部金鑰存放區代理會在 250 毫秒內回應每個請求。如果網路逾時，AWS KMS 將重試一次請求。如果代理第二次失敗，則記錄的延遲是兩次請求嘗試的合併逾時限制，圖表將顯示約 500 毫秒。在代理未在 250 毫秒逾時限制內進行回應的所有其他情況下，記錄的延遲為 250 毫秒。如果代理在加密和解密操作中經常逾時，請咨詢您的外部代理管理員。如需解決延遲問題的說明，請參閱 延遲和逾時錯誤。

回應緩慢也可能表示您的外部金鑰管理器無法處理當前的請求流量。AWS KMS 建議您的外部金鑰管理器每秒最多可處理 1800 個密碼編譯操作請求。如果您的外部金鑰管理器無法處理每秒 1800 個請求，請考慮請求減少自訂金鑰存放區中 KMS 金鑰的請求配額。使用外部金鑰存放區中的 KMS 金鑰進行密碼編譯操作的請求會快速檢錯，並發生限流例外狀況，而不是被外部金鑰存放區代理或外部金鑰管理器處理並拒絕。

延遲圖表衍生自 XksProxyLatency 指標。當您檢視特定資料點時，彈出式視窗會顯示相應的 `KmsOperation` 和 `XksOperation` 維度值，以及在該資料點記錄的平均操作延遲。清單項目會從最高延遲到最低延遲排序。

我們建議您使用XksProxyLatency指標來建立 CloudWatch 警示，以便在延遲接近逾時限制時通知您。如需詳細資訊，請參閱 為響應超時創建 Amazon CloudWatch 警報。

前 5 個例外狀況

指定時間範圍內失敗的密碼編譯和管理操作的前五個例外狀況。使用此圖表可追蹤最常見的錯誤，因此您可以優先考慮工程工作。

此計數包括 AWS KMS 從外部金鑰存放區代理接收的例外狀況，以及當 AWS KMS 無法與外部金鑰存放區代理建立通訊時在內部傳回的 `XksProxyUnreachableException` 例外狀況。

高比率的可重試錯誤可能表示存在網路錯誤，而高比率的不可重試錯誤可能表示外部金鑰存放區的組態有問題。例如，`AuthenticationFailedExceptions` 中的峰值表示在 AWS KMS 中設定的身分驗證憑證與外部金鑰存放區代理之間存在差異。若要檢視外部金鑰存放區組態，請參閱 檢視外部金鑰存放區。若要編輯外部金鑰存放區設定，請參閱 編輯外部金鑰存放區屬性。

AWS KMS 從外部金鑰存放區代理接收的例外狀況與操作失敗時 AWS KMS 傳回的例外狀況不同。對於與外部金鑰存放區的外部組態或連接狀態相關的所有失敗，AWS KMS 密碼編譯操作會傳回 `KMSInvalidStateException`。若要識別問題，請使用隨附的錯誤訊息文字。

下表顯示前 5 個例外狀況圖表中可能出現的例外狀況，以及 AWS KMS 傳回給您的相應例外狀況。

錯誤類型	圖表中顯示的例外狀況	AWS KMS 傳回給您的例外狀況
不可重試	**AccessDeniedException** 如需故障診斷協助，請參閱代理授權問題。	**CustomKeyStoreInvalidStateException** 回應 `CreateKey` 操作。 **KMSInvalidStateException** 回應密碼編譯操作。
不可重試	**AuthenticationFailedException** 如需故障診斷協助，請參閱身分驗證憑證錯誤。	**XksProxyIncorrectAuthenticationCredentialException** 回應 `CreateCustomKeyStore` 和 `UpdateCustomKeyStore` 操作。

錯誤類型	圖表中顯示的例外狀況	AWS KMS 傳回給您的例外狀況
		CustomKeyStoreInvalidStateException 回應 CreateKey 操作。 **KMSInvalidStateException** 回應密碼編譯操作。
可重試	**DependencyTimeoutException** 如需故障診斷協助，請參閱延遲和逾時錯誤。	**XksProxyUriUnreachableException** 回應 CreateCustomKeyStore 和 UpdateCustomKeyStore 操作。 **CustomKeyStoreInvalidStateException** 回應 CreateKey 操作。 **KMSInvalidStateException** 回應密碼編譯操作。
可重試	**InternalException** 外部金鑰存放區代理拒絕了請求，因為其無法與外部金鑰管理器通訊。確認外部金鑰存放區代理組態正確，以及外部金鑰管理器可使用。	**XksProxyInvalidResponseException** 回應 CreateCustomKeyStore 和 UpdateCustomKeyStore 操作。 **CustomKeyStoreInvalidStateException** 回應 CreateKey 操作。 **KMSInvalidStateException** 回應密碼編譯操作。

錯誤類型	圖表中顯示的例外狀況	AWS KMS 傳回給您的例外狀況
不可重試	**InvalidCiphertextException** 如需故障診斷協助，請參閱解密錯誤。	**KMSInvalidStateException** 回應密碼編譯操作。
不可重試	**InvalidKeyUsageException** 如需故障診斷協助，請參閱外部金鑰的密碼編譯操作錯誤。	**XksKeyInvalidConfigurationException** 回應 CreateKey 操作。 **KMSInvalidStateException** 回應密碼編譯操作。
不可重試	**InvalidStateException** 如需故障診斷協助，請參閱外部金鑰的密碼編譯操作錯誤。	**XksKeyInvalidConfigurationException** 回應 CreateKey 操作。 **KMSInvalidStateException** 回應密碼編譯操作。
不可重試	**InvalidUriPathException** 如需故障診斷協助，請參閱一般組態錯誤。	**XksProxyInvalidConfigurationException** 回應 CreateCustomKeyStore 和 UpdateCustomKeyStore 操作。 **CustomKeyStoreInvalidStateException** 回應 CreateKey 操作。 **KMSInvalidStateException** 回應密碼編譯操作。

錯誤類型	圖表中顯示的例外狀況	AWS KMS 傳回給您的例外狀況
不可重試	**KeyNotFoundException** 如需故障診斷協助，請參閱外部金鑰錯誤。	**XksKeyNotFoundException** 回應 CreateKey 操作。 **KMSInvalidStateException** 回應密碼編譯操作。
可重試	**ThrottlingException** 由於請求率非常高，所以外部金鑰存放區代理拒絕了請求。減少使用此外部金鑰存放區中的 KMS 金鑰進行呼叫的頻率。	**XksProxyUriUnreachableException** 回應 CreateCustomKeyStore 和 UpdateCustomKeyStore 操作。 **CustomKeyStoreInvalidStateException** 回應 CreateKey 操作。 **KMSInvalidStateException** 回應密碼編譯操作。
不可重試	**UnsupportedOperationException** 如需故障診斷協助，請參閱外部金鑰的密碼編譯操作錯誤。	**XksKeyInvalidResponseException** 回應 CreateKey 操作。 **KMSInvalidStateException** 回應密碼編譯操作。

錯誤類型	圖表中顯示的例外狀況	AWS KMS 傳回給您的例外狀況
不可重試	**ValidationException** 如需故障診斷協助，請參閱代理問題。	**XksProxyInvalidResponseException** 回應 CreateCustomKeyStore 和 UpdateCustomKeyStore 操作。 **CustomKeyStoreInvalidStateException** 回應 CreateKey 操作。 **KMSInvalidStateException** 回應密碼編譯操作。
可重試	**XksProxyUnreachableException** 如果您重複看到此錯誤，則請確認外部金鑰存放區代理處於作用中狀態且已連接至網路，以及外部金鑰存放區中的 URI 路徑和端點 URI 或 VPC 服務名稱正確無誤。	**XksProxyUriUnreachableException** 回應 CreateCustomKeyStore 和 UpdateCustomKeyStore 操作。 **CustomKeyStoreInvalidStateException** 回應 CreateKey 操作。 **KMSInvalidStateException** 回應密碼編譯操作。

前 5 個例外狀況圖衍生自 XksProxyErrors 指標。當您檢視特定資料點時，彈出式視窗會顯示 ExceptionName 維度值，以及在該資料點記錄的例外狀況次數。這五個清單項目從最常見的例外狀況到最不常見的例外狀況排序。

我們建議您使用XksProxyErrors指標建立 CloudWatch 警示，在一分鐘內記錄超過五個不可重試的錯誤時，通知您潛在的組態問題。如需詳細資訊，請參閱 為不可重試的錯誤創建 Amazon CloudWatch 警報。

憑證到期天數

外部金鑰存放區代理端點 (`XksProxyUriEndpoint`) 的 TLS 憑證到期前的天數。使用此圖表來監控 TLS 憑證即將到期。

憑證到期時，AWS KMS 無法與外部金鑰存放區代理通訊。在您續約憑證之前，外部金鑰存放區中受 KMS 金鑰保護的所有資料都無法存取。

憑證到期天數圖表衍生自 XksProxyCertificateDaysToExpire 指標。我們強烈建議您使用此指標來建立 CloudWatch 警示，通知您即將到期。憑證到期可能會阻礙您存取加密資源。設定警示可讓組織有時間在憑證過期之前續約憑證。如需詳細資訊，請參閱 為憑證到期建立 Amazon CloudWatch 警示。

設定警示

監控區段中的圖表提供指定時間段內外部金鑰存放區和外部金鑰存放區中 KMS 金鑰的運作狀態概觀。不過，您可以根據外部金鑰存放區指標建立 Amazon CloudWatch 警示，以在指標值超過您指定的閾值時通知您。警示可將訊息傳送至 Amazon Simple Notification Service (Amazon SNS) 主題或 Amazon EC2 Auto Scaling 政策。如需 CloudWatch 警示的詳細資訊，請參閱 Amazon 使用 CloudWatch 者指南中的使用 Amazon CloudWatch 警示。

在創建 Amazon CloudWatch 警報之前，您需要一個 Amazon SNS 主題。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南中的建立 Amazon SNS 主題。

主題

- 為憑證到期建立 Amazon CloudWatch 警示
- 為響應超時創建 Amazon CloudWatch 警報
- 為可重試的錯誤創建 Amazon CloudWatch 警報
- 為不可重試的錯誤創建 Amazon CloudWatch 警報

為憑證到期建立 Amazon CloudWatch 警示

此警示會使用AWS KMS發佈至的XksProxyCertificateDaysToExpire指標 CloudWatch 來記錄與外部金鑰存放區 Proxy 端點相關聯之 TLS 憑證的預期到期日。您無法為您帳戶中的所有外部金鑰存放區建立單一警示，也不能為您將來可能建立的外部金鑰存放區建立警示。

我們建議設定警示，以便在憑證設定到期的前 10 天提醒您，但您應該設定最符合您需求的閾值。

建立警示

遵循使用下列必要值根據靜態臨界值建立 CloudWatch 警示中的指示進行。對於其他欄位，請接受預設值，並按要求提供名稱。

欄位	值
選取指標	選擇 KMS，然後選擇 XKS Proxy Certificate Metrics (XKS 代理憑證指標)。 選取您想要監控的 `XksProxyCertificateName` 旁的核取方塊。 然後選擇 Select metric (選取指標)。
統計數字	下限
期間	5 分鐘
閾值類型	靜態
Whenever ...	每`XksProxyCertificateDaysToExpire`當`Lower`比`10`。

為響應超時創建 Amazon CloudWatch 警報

此警示會使用發AWS KMS佈 CloudWatch至的`XksProxyLatency`測量結果，記錄外部金鑰儲存區 Proxy 回應要AWS KMS求所需的毫秒數。您無法為您帳戶中的所有外部金鑰存放區建立單一警示，也不能為您將來可能建立的外部金鑰存放區建立警示。

AWS KMS 預期外部金鑰存放區代理會在 250 毫秒內回應每個請求。我們建議您設定警示，以便在外部金鑰存放區代理的回應時間超過 200 毫秒時提醒您，但您應該設定最符合您需求的閾值。

建立警示

遵循使用下列必要值根據靜態臨界值建立 CloudWatch 警示中的指示進行。對於其他欄位，請接受預設值，並按要求提供名稱。

欄位	值
選取指標	選擇 KMS，然後選擇 XKS Proxy Latency Metrics (XKS 代理延遲指標)。 選取您想要監控的 `KmsOperation` 旁的核取方塊。 然後選擇 Select metric (選取指標)。

欄位	值
統計數字	平均數
期間	5 分鐘
閾值類型	靜態
Whenever ...	每XksProxyLatency當`Greater`比`200`。

為可重試的錯誤創建 Amazon CloudWatch 警報

此警示會使用AWS KMS發佈 CloudWatch 至的XksProxyErrors指標，記錄與外部金鑰存放區 Proxy 要 AWS KMS求相關的例外狀況數目。您無法為您帳戶中的所有外部金鑰存放區建立單一警示，也不能為您將來可能建立的外部金鑰存放區建立警示。

可重試的錯誤會降低您的可靠性百分比，並可指示網路錯誤。我們建議您設定警示，以便在一分鐘內記錄五個以上的可重試錯誤時提醒您，但您應該設定最符合您需求的閾值。

遵循使用下列必要值根據靜態臨界值建立 CloudWatch 警示中的指示進行。對於其他欄位，請接受預設值，並按要求提供名稱。

欄位	值
選取指標	選擇 Queries (查詢) 索引標籤。 將 Namespace (命名空間) 選為 `AWS/KMS`。 針對 Metric name (指標名稱)，輸入 `SUM(XksProxyErrors)` 。 針對 Filter by (篩選依據)，輸入 `ErrorType = Retryable` 。 選擇執行。然後選擇 Select metric (選取指標)。
標籤	`######`
期間	1 分鐘
閾值類型	靜態

欄位	值
Whenever ...	每當 q1 `Greater` 5 時。

為不可重試的錯誤創建 Amazon CloudWatch 警報

此警示會使用AWS KMS發佈 CloudWatch 至的XksProxyErrors指標，記錄與外部金鑰存放區 Proxy 要AWS KMS求相關的例外狀況數目。您無法為您帳戶中的所有外部金鑰存放區建立單一警示，也不能為您將來可能建立的外部金鑰存放區建立警示。

不可重試的錯誤表示外部金鑰存放區的組態有問題。我們建議您設定警示，以便在一分鐘內記錄五個以上的不可重試錯誤時提醒您，但您應該設定最符合您需求的閾值。

遵循使用下列必要值根據靜態臨界值建立 CloudWatch 警示中的指示進行。對於其他欄位，請接受預設值，並按要求提供名稱。

欄位	值
選取指標	選擇 Queries (查詢) 索引標籤。 將 Namespace (命名空間) 選為 `AWS/KMS`。 針對 Metric name (指標名稱)，輸入 `SUM(XksProxyErrors)` 。 針對 Filter by (篩選依據)，輸入 `ErrorType = Non-retryable` 。 選擇執行。然後選擇 Select metric (選取指標)。
標籤	`#######`
期間	1 分鐘
閾值類型	靜態
Whenever ...	每當 q1 `Greater` 5 時。

連接和中斷連接外部金鑰存放區

新的外部金鑰存放區未連接。若要在外部金鑰存放區中建立和使用 AWS KMS keys，您需要將外部金鑰存放區連接至其外部金鑰存放區代理。您可以隨時連接和中斷連接您的外部金鑰存放區，並且檢視其連接狀態。

當您的外部金鑰存放區中斷連接時，AWS KMS 無法與外部金鑰存放區代理進行通訊。因此，您可以檢視和管理外部金鑰存放區和其現有 KMS 金鑰。但是，您無法在外部金鑰存放區中建立 KMS 金鑰，或在密碼編譯操作中使用其 KMS 金鑰。您可能需要在某些時候中斷連接外部金鑰存放區，例如在編輯其屬性時，但需要進行相應的規劃。中斷連接金鑰存放區可能會中斷使用其 KMS 金鑰之 AWS 服務的操作。

您不需要連接您的外部金鑰存放區。您可以將外部金鑰存放區無限期保留在中斷連接狀態，並只在您需要使用它時連接它。不過，您可能希望定期測試連接，以驗證設定正確並且可連接。

當您中斷連接自訂金鑰存放區時，該金鑰存放區中的 KMS 金鑰會立即變成無法使用 (視最終一致性而定)。不過，使用受 KMS 金鑰保護之資料金鑰所加密的資源不會受影響，除非再次使用 KMS 金鑰 (例如解密資料金鑰)。此問題會影響 AWS 服務，其中許多服務會使用資料金鑰來保護您的資源。如需詳細資訊，請參閱 無法使用的 KMS 金鑰如何影響資料金鑰。

> **Note**
>
> 僅當金鑰存放區從未連接或明確中斷連接時，外部金鑰存放區才會處於 DISCONNECTED 狀態。CONNECTED 狀態並不表示外部金鑰存放區或其支援元件正在高效運作。如需外部金鑰存放區元件效能的相關資訊，請參閱每個外部金鑰存放區詳細資訊頁面之 Monitoring (監控) 區段中的圖表。如需詳細資訊，請參閱 監控外部金鑰存放區。
>
> 您的外部金鑰管理器可能會提供其他方法來停止和重新啟動 AWS KMS 外部金鑰存放區與外部金鑰存放區代理之間的通訊，或者外部金鑰存放區代理與外部金鑰管理器之間的通訊。如需詳細資訊，請參閱外部金鑰管理器文件。

主題

- 連接外部金鑰存放區
- 中斷連接外部金鑰存放區
- 連線狀態
- 連接外部金鑰存放區 (主控台)
- 連接外部金鑰存放區 (API)

- 中斷連接外部金鑰存放區 (主控台)
- 中斷連接外部金鑰存放區 (API)

連接外部金鑰存放區

外部金鑰存放區連接到其外部金鑰存放區代理時，您可以在外部金鑰存放區中建立 KMS 金鑰，並在密碼編譯操作中使用其現有的 KMS 金鑰。

將外部金鑰存放區連接至其外部金鑰存放區代理的程序會有所不同，這取決於外部金鑰存放區的連接。

- 當您連線具有公用端點連線的外部金鑰存放區時，AWS KMS會將GetHealthStatus 要求傳送至外部金鑰存放區 Proxy，以驗證 Proxy URI 端點、Proxy URI 路徑和 Proxy 驗證認證。來自代理的成功回應可確認代理 URI 端點和代理 URI 路徑是正確且可存取的，而且代理驗證了使用外部金鑰存放區之代理身分驗證憑證所簽署的請求。
- 當您使用 VPC 端點服務連接將外部金鑰存放區連接至其外部金鑰存放區代理時，AWS KMS 會執行下列動作：
  - 確認在代理 URI 端點中指定之私有 DNS 名稱的網域已經過驗證。
  - 建立從 AWS KMS VPC 到 VPC 端點服務的介面端點。
  - 為代理 URI 端點中指定的私有 DNS 名稱建立私有託管區域
  - 傳送要GetHealthStatus求至外部金鑰存放區 Proxy。來自代理的成功回應可確認代理 URI 端點和代理 URI 路徑是正確且可存取的，而且代理驗證了使用外部金鑰存放區之代理身分驗證憑證所簽署的請求。

連接操作開始了連接自訂金鑰存放區的過程，但將外部金鑰存放區連接到其外部代理大約需要五分鐘。來自連接操作的成功回應並不表示外部金鑰存放區已連接。若要確認連線成功，請使用AWS KMS主控台或DescribeCustomKeyStores作業來檢視金鑰存放區外部的連線狀態。

當連接狀態為 `FAILED` 時，AWS KMS 主控台中會顯示連接錯誤代碼，並會新增到 `DescribeCustomKeyStore` 回應中。如需有關解譯連接錯誤代碼的說明，請參閱 外部金鑰存放區的連接錯誤代碼。

中斷連接外部金鑰存放區

當您中斷具有 VPC 端點服務連接的外部金鑰存放區與其外部金鑰存放區代理的連接時，AWS KMS 會刪除其與 VPC 端點服務的介面端點，並移除其為支援連接而建立的網路基礎設施。具有公有端點連接的外部金鑰存放區不需要同等程序。此動作不會影響 VPC 端點服務或其任何支援元件，也不會影響外部金鑰存放區代理或任何外部元件。

當外部金鑰存放區中斷連接時，AWS KMS 不會傳送任何請求至外部金鑰存放區代理。外部金鑰存放區的連接狀態為 DISCONNECTED。中斷連接的外部金鑰存放區中的 KMS 金鑰處於 UNAVAILABLE 金鑰狀態 (除非其為待刪除)，這表示其無法用於密碼編譯操作。但是，您仍可以檢視和管理外部金鑰存放區和其現有 KMS 金鑰。

中斷連接狀態被設計為暫時且可還原的。您可以隨時重新連接外部金鑰存放區。通常，不需要重新設定。但是，如果相關聯的外部金鑰存放區代理的任何屬性在中斷連接時發生變更，例如其代理身分驗證憑證的輪換，則必須先編輯外部金鑰存放區設定，才能重新連接。

> **Note**
>
> 當自訂金鑰存放區中斷連接時，所有在自訂金鑰存放區中建立 KMS 金鑰的嘗試，或在密碼編譯操作中使用現有 KMS 金鑰的嘗試，均會失敗。此動作可防止使用者存放和存取敏感資料。

若要更好地預估中斷連接您的外部金鑰存放區的效果，請在外部金鑰存放區中識別 KMS 金鑰，並判斷其過去的使用情形。

您可能中斷連接外部金鑰存放區的原因如下所示：

- 編輯其屬性。在連接外部金鑰存放區時，您可以編輯自訂金鑰存放區名稱、代理 URI 路徑和代理身分驗證憑證。但是，若要編輯代理連接類型、代理 URI 端點或 VPC 端點服務名稱，您必須先中斷連接外部金鑰存放區。如需詳細資訊，請參閱 編輯外部金鑰存放區屬性。
- 停止 AWS KMS 與外部金鑰存放區代理之間的所有通訊。也可以透過停用您的端點或 VPC 端點服務來停止 AWS KMS 與代理之間的通訊。此外，您的外部金鑰存放區代理或金鑰管理軟體可能會提供額外的機制，以防止 AWS KMS 與代理通訊，或防止代理存取您的外部金鑰管理器。
- 停用外部金鑰存放區中的所有 KMS 金鑰。您可以使用AWS KMS主控台或DisableKey作業停用和重新啟用外部金鑰存放區中的 KMS 金鑰。這些操作會快速完成 (視最終一致性而定)，但一次只能對一個 KMS 金鑰進行。中斷連接外部金鑰存放區會將外部金鑰存放區中所有 KMS 金鑰的金鑰狀態變更為 Unavailable，這會防止在任何密碼編譯操作中對其進行使用。
- 為了修復失敗的連接嘗試。如果嘗試連接外部金鑰存放區失敗 (自訂金鑰存放區的連接狀態為 FAILED)，您必須先中斷連接外部金鑰存放區，之後再嘗試重新連接。

連線狀態

連接和中斷連接會變更自訂金鑰存放區的連接狀態。AWS CloudHSM 金鑰存放區和外部金鑰存放區的連接狀態值相同。

若要檢視自訂金鑰存放區的連線狀態，請使用DescribeCustomKeyStores作業或AWS KMS主控台。Connection state (連接狀態) 會顯示在每個自訂金鑰存放區資料表、每個自訂金鑰存放區詳細資料頁面的 General configuration (一般組態) 區段以及自訂金鑰存放區中 KMS 金鑰的 Cryptographic configuration (密碼編譯組態) 索引標籤中。如需詳細資訊，請參閱 檢視 AWS CloudHSM 金鑰存放區 和 檢視外部金鑰存放區。

自訂金鑰存放區可以有下列其中一個連接狀態：

- CONNECTED：自訂金鑰存放區已連接至其備份金鑰存放區。您可在自訂金鑰存放區中建立和使用 KMS 金鑰。

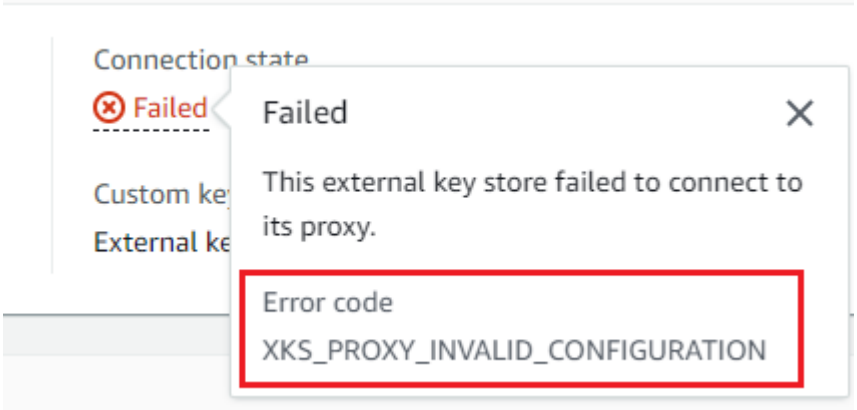
  AWS CloudHSM 金鑰存放區的備份金鑰存放區是其相關聯的 AWS CloudHSM 叢集。外部金鑰存放區的備份金鑰存放區是外部金鑰存放區代理及其支援的外部金鑰管理器。

  CONNECTED 狀態表示連接成功，且自訂金鑰存放區尚未有意中斷連接。這並不表示連接運作正常。如需有關與AWS CloudHSM金鑰存放區相關聯之AWS CloudHSM叢集狀態的資訊，請參閱《AWS CloudHSM使用指南》AWS CloudHSM中的〈取得的指 CloudWatch 標〉。如需有關外部金鑰存放區狀態和操作的相關資訊，請參閱每個外部金鑰存放區詳細資訊頁面的監控區段中的圖表。如需詳細資訊，請參閱 監控外部金鑰存放區。
- CONNECTING：連接自訂金鑰存放區的程序正在進行中。這是暫時的狀態。
- DISCONNECTED：自訂金鑰存放區從未連線至其支援，或是使用AWS KMS主控台或DisconnectCustomKeyStore作業故意中斷連線。
- DISCONNECTING：中斷連接自訂金鑰存放區的程序正在進行中。這是暫時的狀態。
- FAILED：嘗試連接自訂金鑰存放區失敗。ConnectionErrorCode在DescribeCustomKeyStores響應中表示問題。

若要連接自訂金鑰存放區，其連接狀態必須為 DISCONNECTED。如果連接狀態為 FAILED，則請使用 ConnectionErrorCode 來識別並解決問題。請先中斷連接自訂金鑰存放區，然後再重試連接。如需連線失敗的協助，請參閱 外部金鑰存放區連接錯誤。如需有關回應連接錯誤代碼的說明，請參閱 外部金鑰存放區的連接錯誤代碼。

若要檢視連接錯誤代碼：

- 在DescribeCustomKeyStores回應中，檢視ConnectionErrorCode元素的值。只有當 ConnectionState 為 FAILED 時，此元素才會出現在 DescribeCustomKeyStores 回應中。
- 若要在 AWS KMS 主控台中檢視連接錯誤代碼，請在外部金鑰存放區的詳細資訊頁面上，將滑鼠游標移至 Failed (失敗) 值上。



連接外部金鑰存放區 (主控台)

您可以使用 AWS KMS 主控台將外部金鑰存放區連接至其外部金鑰存放區代理。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，依次選擇 Custom key stores (自訂金鑰存放區)、External key stores (外部金鑰存放區)。
4. 選擇您想要連接的外部金鑰存放區資料列。

   如果外部金鑰存放區的連接狀態為 FAILED (失敗)，則在連接之前，必須中斷連接外部金鑰存放區。
5. 從 Key store actions (金鑰存放區動作) 選單中，選擇 Connect (連接)。

連接程序通常需要大約五分鐘才能完成。操作完成時，連接狀態會變更為 CONNECTED (已連接)。

如果連接狀態為 Failed (失敗)，則請將滑鼠游標移至連接狀態上，以查看連接錯誤代碼，其中會說明錯誤原因。如需有關回應連接錯誤代碼的說明，請參閱 外部金鑰存放區的連接錯誤代碼。若要連接具有 Failed (失敗) 連接狀態的外部金鑰存放區，您必須先中斷連接自訂金鑰存放區。

連接外部金鑰存放區 (API)

若要連線已中斷連線的外部金鑰存放區，請使用此ConnectCustomKeyStore作業。

在連接之前，外部金鑰存放區的連接狀態必須為 DISCONNECTED。如果當前連接狀態為 FAILED，則請中斷連接外部金鑰存放區，然後將其連接。

連接程序需要大約五分鐘才能完成。除非它快速失敗，否則 ConnectCustomKeyStore 會傳回 HTTP 200 回應和不帶屬性的 JSON 物件。不過，這個初始回應並不表示連接已成功。若要判斷外部金鑰存放區是否已連線，請參閱DescribeCustomKeyStores回應中的連線狀態。

本節中的範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

若要識別外部金鑰存放區，請使用其自訂金鑰存放區 ID。您可以在主控台的 [自訂金鑰存放區] 頁面上或使用DescribeCustomKeyStores作業來尋找 ID。執行此範例之前，請將範例 ID 以有效的 ID 取代。

```
$ aws kms connect-custom-key-store --custom-key-store-id cks-1234567890abcdef0
```

ConnectCustomKeyStore 操作不會傳回其回應中的 ConnectionState。若要確認外部金鑰存放區已連線，請使用DescribeCustomKeyStores作業。在預設情況下，此操作會傳回您帳戶和區域中的所有自訂金鑰存放區。但是，您可以使用 CustomKeyStoreId 或 CustomKeyStoreName 參數 (但不能同時使用) 來限制對特定自訂金鑰存放區的回應。CONNECTED 的 ConnectionState 值表示外部金鑰存放區已連接至其外部金鑰存放區代理。

```
$ aws kms describe-custom-key-stores --custom-key-store-name ExampleXksVpc
{
    "CustomKeyStores": [
    {
      "CustomKeyStoreId": "cks-9876543210fedcba9",
      "CustomKeyStoreName": "ExampleXksVpc",
      "ConnectionState": "CONNECTED",
      "CreationDate": "2022-12-13T18:34:10.675000+00:00",
      "CustomKeyStoreType": "EXTERNAL_KEY_STORE",
      "XksProxyConfiguration": {
        "AccessKeyId": "ABCDE98765432EXAMPLE",
        "Connectivity": "VPC_ENDPOINT_SERVICE",
        "UriEndpoint": "https://example-proxy-uri-endpoint-vpc",
        "UriPath": "/example/prefix/kms/xks/v1",
        "VpcEndpointServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-example"
      }
    }
  ]
}
```

如果 DescribeCustomKeyStores 回應中的 ConnectionState 值為 FAILED，則 ConnectionErrorCode 元素會指出失敗的原因。

在下列範例中，ConnectionErrorCode 的 XKS_VPC_ENDPOINT_SERVICE_NOT_FOUND 值表示 AWS KMS 找不到它用來與外部金鑰存放區代理通訊的 VPC 端點服務。確認 XksProxyVpcEndpointServiceName 正確無誤、AWS KMS 服務主體是 Amazon VPC 端點服務中允許的主體以及 VPC 端點服務不需要接受連接請求。如需有關回應連接錯誤代碼的說明，請參閱 外部金鑰存放區的連接錯誤代碼。

```
$ aws kms describe-custom-key-stores --custom-key-store-name ExampleXksVpc
{
    "CustomKeyStores": [
    {
      "CustomKeyStoreId": "cks-9876543210fedcba9",
      "CustomKeyStoreName": "ExampleXksVpc",
      "ConnectionState": "FAILED",
      "ConnectionErrorCode": "XKS_VPC_ENDPOINT_SERVICE_NOT_FOUND",
      "CreationDate": "2022-12-13T18:34:10.675000+00:00",
      "CustomKeyStoreType": "EXTERNAL_KEY_STORE",
      "XksProxyConfiguration": {
        "AccessKeyId": "ABCDE98765432EXAMPLE",
        "Connectivity": "VPC_ENDPOINT_SERVICE",
        "UriEndpoint": "https://example-proxy-uri-endpoint-vpc",
        "UriPath": "/example/prefix/kms/xks/v1",
        "VpcEndpointServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-example"
      }
    }
  ]
}
```

中斷連接外部金鑰存放區 (主控台)

您可以使用 AWS KMS 主控台將外部金鑰存放區連接至其外部金鑰存放區代理。此程序約需 5 分鐘才能完成。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，依次選擇 Custom key stores (自訂金鑰存放區)、External key stores (外部金鑰存放區)。
4. 選擇您想要中斷連接之外部金鑰存放區的資料列。
5. 從 Key store actions (金鑰存放區動作) 選單中，選擇 Disconnect (中斷連接)。

操作完成時，連接狀態會從 DISCONNECTING (中斷連接中) 變更為 DISCONNECTED (已中斷連接)。如果操作失敗，會出現錯誤訊息，其中描述問題並提供如何修正的協助。如果您需要更多協助，請參閱外部金鑰存放區連接錯誤。

中斷連接外部金鑰存放區 (API)

若要中斷連接的外部金鑰存放區，請使用此DisconnectCustomKeyStore作業。如果操作成功，則 AWS KMS 傳回 HTTP 200 回應和不帶屬性的 JSON 物件。該程序需要大約五分鐘才能完成。若要尋找外部金鑰存放區的連線狀態，請使用DescribeCustomKeyStores作業。

本節中的範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

此範例中斷連接具有 VPC 端點服務連接的外部金鑰存放區。執行此範例之前，請將範例自訂金鑰存放區 ID 取代為有效的 ID。

```
$ aws kms disconnect-custom-key-store --custom-key-store-id cks-1234567890abcdef0
```

若要確認外部金鑰存放區已中斷連線，請使用此DescribeCustomKeyStores作業。在預設情況下，此操作會傳回您帳戶和區域中的所有自訂金鑰存放區。但是，您可以使用 CustomKeyStoreId 和 CustomKeyStoreName 參數 (但不能同時使用) 來限制對特定自訂金鑰存放區的回應。DISCONNECTED 的 ConnectionState 值表示此範例外部金鑰存放區不再連接至其外部金鑰存放區代理。

```
$ aws kms describe-custom-key-stores --custom-key-store-name ExampleXksVpc
{
    "CustomKeyStores": [
    {
      "CustomKeyStoreId": "cks-9876543210fedcba9",
      "CustomKeyStoreName": "ExampleXksVpc",
      "ConnectionState": "DISCONNECTED",
      "CreationDate": "2022-12-13T18:34:10.675000+00:00",
      "CustomKeyStoreType": "EXTERNAL_KEY_STORE",
      "XksProxyConfiguration": {
        "AccessKeyId": "ABCDE98765432EXAMPLE",
        "Connectivity": "VPC_ENDPOINT_SERVICE",
        "UriEndpoint": "https://example-proxy-uri-endpoint-vpc",
        "UriPath": "/example/prefix/kms/xks/v1",
        "VpcEndpointServiceName": "com.amazonaws.vpce.us-east-1.vpce-svc-example"
      }
    }
```

```
  ]
}
```

刪除外部金鑰存放區

刪除外部金鑰存放區時，AWS KMS 會從 AWS KMS 中刪除有關外部金鑰存放區的所有中繼資料，包括其外部金鑰存放區代理的相關資訊。此操作不會影響外部金鑰存放區代理、外部金鑰管理器、外部金鑰或您為支援外部金鑰存放區而建立的任何 AWS 資源，例如 Amazon VPC 或 VPC 端點服務。

刪除外部金鑰存放區之前，您必須刪除金鑰存放區中的所有 KMS 金鑰，並從其外部金鑰存放區代理中中斷連接金鑰存放區。否則，嘗試刪除金鑰存放區會失敗。

刪除外部金鑰存放區是不可復原的，但您可以建立新的外部金鑰存放區，並將其與相同的外部金鑰存放區代理和外部金鑰管理器產生關聯。但是，即使您可以存取相同的外部金鑰材料，也無法在外部金鑰存放區中重新建立對稱加密 KMS 金鑰。AWS KMS 包含每個 KMS 金鑰唯一的對稱密文中的中繼資料。此安全功能可以確保僅已加密資料的 KMS 金鑰可以解密。

請考慮將其中斷連接，而不是刪除外部金鑰存放區。當外部金鑰存放區中斷連接時，您可以管理外部金鑰存放區和其 AWS KMS keys，但無法在外部金鑰存放區中建立或使用 KMS 金鑰。您可以隨時重新連接外部金鑰存放區，並繼續使用其 KMS 金鑰來加密和解密資料。中斷連接的外部金鑰存放區代理或其無法使用的 KMS 金鑰無需支付任何費用。

主題

- 刪除外部金鑰存放區 (主控台)
- 刪除外部金鑰存放區 (API)

刪除外部金鑰存放區 (主控台)

您可以使用 AWS KMS 主控台刪除外部金鑰存放區。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，依次選擇 Custom key stores (自訂金鑰存放區)、External key stores (外部金鑰存放區)。
4. 尋找代表您要刪除之外部金鑰存放區的資料列。如果外部金鑰存放區的連接狀態不是 DISCONNECTED (已中斷連接)，則在刪除之前，必須中斷連接外部金鑰存放區。
5. 從 Key store actions (金鑰存放區動作) 選單中，選擇 Delete (刪除)。

當操作完成時，就會出現成功訊息，並且外部金鑰存放區將不再顯示在金鑰存放區清單中。如果操作失敗，就會出現錯誤訊息，其中描述問題並提供如何修正的協助。如果您需要更多協助，請參閱外部金鑰存放區故障診斷。

刪除外部金鑰存放區 (API)

若要刪除外部金鑰存放區，請使用此DeleteCustomKeyStore作業。如果操作成功，則 AWS KMS 傳回 HTTP 200 回應和不帶屬性的 JSON 物件。

若要開始，請中斷連接外部金鑰存放區。執行此命令之前，請將範例自訂金鑰存放區 ID 以有效的 ID 取代。

```
$ aws kms disconnect-custom-key-store --custom-key-store-id cks-1234567890abcdef0
```

中斷外部金鑰存放區之後，您可以使用該DeleteCustomKeyStore作業將其刪除。

```
$ aws kms delete-custom-key-store --custom-key-store-id cks-1234567890abcdef0
```

若要確認已刪除外部金鑰存放區，請使用此DescribeCustomKeyStores作業。

```
$ aws kms describe-custom-key-stores

{
    "CustomKeyStores": []
}
```

如果您指定不再存在的自訂金鑰存放區名稱或 ID，則 AWS KMS 會傳回 `CustomKeyStoreNotFoundException` 例外狀況。

```
$ aws kms describe-custom-key-stores --custom-key-store-id cks-1234567890abcdef0

An error occurred (CustomKeyStoreNotFoundException) when calling the
 DescribeCustomKeyStore operation:
```

管理外部金鑰存放區中的 KMS 金鑰

若要在外部金鑰存放區中建立、檢視、管理、使用和排程 KMS 金鑰的刪除，您可以使用與用於其他 KMS 金鑰非常相似的程序。但是，當您在外部金鑰存放區中建立 KMS 金鑰時，您會指定外部金鑰存放區和外部金鑰。當您在外部金鑰存放區中使用 KMS 金鑰時，外部金鑰管理器將使用指定的外部金鑰執行加密和解密操作。

AWS KMS 無法建立、檢視、更新或刪除外部金鑰管理器中的任何密碼編譯金鑰。AWS KMS 從不會直接存取您的外部金鑰管理器或任何外部金鑰。所有密碼編譯操作的請求都會由您的外部金鑰存放區代理進行協調。若要在外部金鑰存放區中使用 KMS 金鑰，託管 KMS 金鑰的外部金鑰存放區必須連接至其外部金鑰存放區代理。

支援的功能

除了本節所討論的程序，您還可以對外部金鑰存放區中的 KMS 金鑰執行下列操作：

- 使用金鑰政策、IAM 政策和授予，以控制對 KMS 金鑰的存取。
- 啟用和停用 KMS 金鑰。這些動作不會影響外部金鑰管理器中的外部金鑰。
- 指派標籤並建立別名，使用屬性型存取控制 (ABAC) 授權對 KMS 金鑰的存取。
- 將 KMS 金鑰用於與 AWS KMS 整合的 AWS 服務，且這些服務支援客戶受管金鑰。

不支援的功能

- 外部金鑰存放區僅支援對稱加密 KMS 金鑰。您無法在外部金鑰存放區中建立 HMAC KMS 金鑰或非對稱 KMS 金鑰。
- GenerateDataKeyPair和GenerateDataKeyPairWithoutPlaintext不支援外部金鑰存放區中的 KMS 金鑰。
- 您無法使用 AWS CloudFormation 範本在外部金鑰存放區中建立外部金鑰存放區或 KMS 金鑰。
- 外部金鑰存放區不支援多區域金鑰。
- 外部金鑰存放區不支援具有匯入金鑰材料的 KMS 金鑰。
- 外部金鑰存放區中的 KMS 金鑰不支援自動金鑰輪換。

主題

- 在外部金鑰存放區中建立 KMS 金鑰
- 檢視外部金鑰存放區中的 KMS 金鑰
- 使用外部金鑰存放區中的 KMS 金鑰
- 排程從外部金鑰存放區刪除 KMS 金鑰

在外部金鑰存放區中建立 KMS 金鑰

建立並連接外部金鑰存放區之後，您可以在金鑰存放區建立 AWS KMS keys。其必須是具有 External key store (外部金鑰存放區) (EXTERNAL_KEY_STORE) 原始值的對稱加密 KMS 金鑰。您無法在自訂金

鑰存放區中建立非對稱 KMS 金鑰、HMAC KMS 金鑰，或是含有匯入金鑰資料的 KMS 金鑰。此外，您無法在自訂金鑰存放區中使用對稱加密 KMS 金鑰來產生非對稱資料金鑰對。

與標準 KMS 金鑰相比，外部金鑰存放區中的 KMS 金鑰可能具有較差的延遲、耐久性和可用性，因為其依賴於 AWS 之外的元件。在外部金鑰存放區中建立或使用 KMS 金鑰之前，請確認您需要具有外部金鑰存放區屬性的金鑰。

> **Note**
>
> 有些外部金鑰管理器會提供更簡單的方法來建立外部金鑰存放區的 KMS 金鑰。如需詳細資訊，請參閱外部金鑰管理器文件。

若要在外部金鑰存放區建立 KMS 金鑰，您必須指定下列項目：

- 外部金鑰存放區的 ID。
- 外部金鑰存放區 (EXTERNAL_KEY_STORE) 的金鑰材料來源。
- 與外部金鑰存放區相關聯的外部金鑰管理器中的現有外部金鑰 ID。此外部金鑰作為 KMS 金鑰的金鑰材料。您無法在建立 KMS 金鑰之後變更外部金鑰 ID。

  AWS KMS 在加密和解密操作請求中向外部金鑰存放區代理提供外部金鑰 ID。AWS KMS 無法直接存取您的外部金鑰管理器或其任何密碼編譯金鑰。

除了外部金鑰之外，外部金鑰存放區中的 KMS 金鑰也包含 AWS KMS 金鑰材料。使用 KMS 金鑰進行加密的所有資料首先在 AWS KMS 中使用該金鑰的 AWS KMS 金鑰材料進行加密，然後再由外部金鑰管理器使用外部金鑰加密。此雙重加密程序可確保受外部金鑰存放區中 KMS 金鑰保護的密文至少與僅受 AWS KMS 保護的密文一樣強。如需詳細資訊，請參閱 外部金鑰存放區的運作方式。

當 CreateKey 操作成功時，新 KMS 金鑰的金鑰狀態為 Enabled。當您在外部金鑰存放區檢視 KMS 金鑰時，您會看到一般屬性，例如其金鑰 ID、金鑰規格、金鑰用途、金鑰狀態和建立日期。但是您也可以看到外部金鑰存放區的 ID 和連接狀態以及外部金鑰的 ID。

如果您嘗試在外部金鑰存放區建立 KMS 金鑰失敗，請使用錯誤訊息來判斷原因。這可能表示外部金鑰存放區未連接 (CustomKeyStoreInvalidStateException)、外部金鑰存放區代理找不到具有指定外部金鑰 ID 的外部金鑰 (XksKeyNotFoundException)，或外部金鑰已與相同外部金鑰存放區中的 KMS 金鑰相關聯 XksKeyAlreadyInUseException。

.

如需會在外部金鑰存放區中建立 KMS 金鑰之操作的 AWS CloudTrail 日誌範例，請參閱 CreateKey。

主題

外部金鑰存放區中 KMS 金鑰的要求

若要在外部金鑰存放區中建立 KMS 金鑰，外部金鑰存放區、KMS 金鑰以及作為 KMS 金鑰之外部密碼編譯金鑰材料的外部金鑰需要下列屬性。

外部金鑰存放區要求

- 必須連接到其外部金鑰存放區代理。

  若要檢視外部金鑰存放區的連接狀態，請參閱 檢視外部金鑰存放區。若要連接外部金鑰存放區，請參閱 連接和中斷連接外部金鑰存放區。

KMS 金鑰要求

您無法在建立 KMS 金鑰之後變更這些屬性。

- 金鑰規格：SYMMETRIC_DEFAULT
- 金鑰用途：ENCRYPT_DECRYPT
- 金鑰材料來源：EXTERNAL_KEY_STORE
- 多區域：FALSE

外部金鑰要求

- 256 位元 AES 密碼編譯金鑰 (256 個隨機位元)。外部金鑰的 `KeySpec` 必須為 `AES_256`。
- 已啟用並可供使用。外部金鑰的 `Status` 必須為 `ENABLED`。
- 設定用於加密和解密。外部金鑰的 `KeyUsage` 必須包含 `ENCRYPT` 和 `DECRYPT`。
- 僅與此 KMS 金鑰搭配使用。外部金鑰存放區中的每個 KMS key 都必須與不同的外部金鑰關聯。

  AWS KMS 還建議僅將外部金鑰用於外部金鑰存放區。此限制可讓您更輕鬆地識別並解決金鑰的問題。

- 可供外部金鑰存放區的外部金鑰存放區代理存取。

  如果外部金鑰存放區代理找不到使用指定外部金鑰 ID 的金鑰，CreateKey 操作就會失敗。
- 可以處理您使用 AWS 服務 產生的預期流量。AWS KMS 建議外部金鑰準備好處理每秒多達 1800 個請求。

在外部金鑰存放區建立 KMS 金鑰 (主控台)

有兩種方法可以在外部金鑰存放區中建立 KMS 金鑰。

- 方法 1 (建議選擇)：選擇外部金鑰存放區，然後在該外部金鑰存放區建立 KMS 金鑰。
- 方法 2：建立 KMS 金鑰，然後指出金鑰位於外部金鑰存放區。

如您採用方法 1 (在建立金鑰之前選擇外部金鑰存放區)，AWS KMS 會為您選擇所有必要的 KMS 金鑰屬性，並填寫外部金鑰存放區的 ID。此方法可避免您在建立 KMS 金鑰時可能發生的錯誤。

> **Note**
>
> 請勿在別名、說明或標籤包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

方法 1 (建議選擇)：在外部金鑰存放區開始

若要使用此方法，請選擇外部金鑰存放區，然後建立 KMS 金鑰。AWS KMS 主控台會為您選擇所有必需的屬性，並填寫外部金鑰存放區的 ID。此方法可避免您在建立 KMS 金鑰時可能發生的許多錯誤。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，依次選擇 Custom key stores (自訂金鑰存放區)、External key stores (外部金鑰存放區)。
4. 選擇外部金鑰存放區的名稱。
5. 在右上角，選擇 Create a KMS key in this key store (在此金鑰存放區建立 KMS 金鑰)。

   如果未連接外部金鑰存放區，系統會提示您進行連接。如果連接嘗試失敗，則您需要先解決問題並連接外部金鑰存放區，才能在其中建立新的 KMS 金鑰。

如果外部金鑰存放區已連接，則系統會將您重新導向至 Customer managed keys (客戶受管金鑰) 頁面以建立金鑰。已為您選擇所需的 Key configuration (金鑰組態) 值。此外，會填寫外部金鑰存放區的自訂金鑰存放區 ID，不過您可以變更它。

6. 在外部金鑰管理器中輸入外部金鑰的金鑰 ID。此外部金鑰必須滿足要求，才能與 KMS 金鑰搭配使用。建立金鑰之後，便無法變更此值。

   如果外部金鑰有多個 ID，請輸入外部金鑰存放區代理用來識別外部金鑰的金鑰 ID。

7. 確認您要在指定的外部金鑰存放區中建立 KMS 金鑰。

8. 選擇下一步。

   此程序的其餘部分與建立標準 KMS 金鑰相同。

9. 輸入 KMS 金鑰的別名 (必需) 和描述 (選用)。

10. (選用)。在 Add Tags (新增標籤) 頁面，新增標籤來識別或分類 KMS 金鑰。

    將標籤新增到 AWS 資源時，AWS 會產生成本配置報告，內含按標籤彙總的用量與成本。標籤也可以用來控制 KMS 金鑰的存取。如需標記 KMS 金鑰的詳細資訊，請參閱 標記金鑰 和 AWS KMS 的 ABAC。

11. 選擇下一步。

12. 在 Key Administrators (金鑰管理員) 區段中，選取可管理 KMS 金鑰的 IAM 使用者和角色。如需詳細資訊，請參閱允許金鑰管理員來管理 KMS 金鑰。

    > **Note**
    >
    > IAM 政策可授權其他 IAM 使用者和角色來使用 KMS 金鑰。
    > IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

13. (選用) 若要防止這些金鑰管理員刪除此 KMS 金鑰，請清除 Allow key administrators to delete this key (允許金鑰管理員刪除此金鑰) 核取方塊。

    刪除 KMS 金鑰是一種破壞性和不可復原的操作，可可能導致密文無法復原。即使您擁有外部金鑰材料，也無法在外部金鑰存放區中重新建立對稱 KMS 金鑰。不過，刪除 KMS 金鑰不會影響其相關聯的外部金鑰。如需從外部金鑰存放區刪除 KMS 金鑰的相關資訊，請參閱 排程從外部金鑰存放區刪除 KMS 金鑰。

14. 選擇下一步。

15. 在 This account (這個帳戶) 區段中，選取此 AWS 帳戶 中可在密碼編譯操作中使用 KMS 金鑰的 IAM 使用者和角色。如需詳細資訊，請參閱允許金鑰使用者使用 KMS 金鑰。

    > **Note**
    >
    > IAM 政策可授權其他 IAM 使用者和角色來使用 KMS 金鑰。
    > IAM 最佳實務不建議使用具有長期憑證的 IAM 使用者。盡可能使用提供臨時憑證的 IAM 角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 IAM 中的安全性最佳實務。

16. (選用) 您可以允許其他 AWS 帳戶 將此 KMS 金鑰用於密碼編譯操作。若要這樣做，請在頁面底部的其他 AWS 帳戶 區段中，選擇新增另一個 AWS 帳戶，然後輸入外部帳戶的 AWS 帳戶 ID。若要新增多個外部帳戶，請重複此步驟。

    > **Note**
    >
    > 其他 AWS 帳戶 的管理員也必須透過為其使用者建立 IAM 政策，來允許存取 KMS 金鑰。如需詳細資訊，請參閱 允許其他帳戶中的使用者使用 KMS 金鑰。

17. 選擇下一步。
18. 檢閱您選擇的金鑰設定。您仍然可以返回並變更所有設定。
19. 完成時，請選擇 Finish (完成) 以建立金鑰。

方法 2：在客戶受管金鑰中開始

此程序與使用 AWS KMS 金鑰材料建立對稱加密金鑰的程序相同。但是，在此程序中，您可指定外部金鑰存放區的自訂金鑰存放區 ID 和外部金鑰的金鑰 ID。您也必須為外部金鑰存放區中的 KMS 金鑰指定必要的屬性值，例如金鑰規格和金鑰用途。

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 選擇建立金鑰。
5. 選擇 Symmetric (對稱)。
6. 在 Key usage (金鑰用途) 欄位中，系統會自動選取 Encrypt and decrypt (加密和解密) 選項。請勿變更該欄位。

7. 選擇 Advanced options (進階選項)。
8. 對於 Key material origin (金鑰材料來源)，選擇 External key store (外部金鑰存放區)。
9. 確認您要在指定的外部金鑰存放區中建立 KMS 金鑰。
10. 選擇下一步。
11. 選擇代表新 KMS 金鑰的外部金鑰存放區的資料列。

    您無法選擇已中斷連接的外部金鑰存放區。若要連接已中斷連接的金鑰存放區，請選擇金鑰存放區名稱，然後在 Key store actions (金鑰存放區動作) 中選擇 Connect (連接)。如需詳細資訊，請參閱 連接外部金鑰存放區 (主控台)。
12. 在外部金鑰管理器中輸入外部金鑰的金鑰 ID。此外部金鑰必須滿足要求，才能與 KMS 金鑰搭配使用。建立金鑰之後，便無法變更此值。

    如果外部金鑰有多個 ID，請輸入外部金鑰存放區代理用來識別外部金鑰的金鑰 ID。
13. 選擇下一步。

    此程序的其餘部分與建立標準 KMS 金鑰相同。
14. 輸入 KMS 金鑰的別名和選用描述。
15. (選用)。在 Add Tags (新增標籤) 頁面，新增標籤來識別或分類 KMS 金鑰。

    將標籤新增到 AWS 資源時，AWS 會產生成本配置報告，內含按標籤彙總的用量與成本。標籤也可以用來控制 KMS 金鑰的存取。如需標記 KMS 金鑰的詳細資訊，請參閱 標記金鑰 和 AWS KMS 的 ABAC。
16. 選擇下一步。
17. 在 Key Administrators (金鑰管理員) 區段中，選取可管理 KMS 金鑰的 IAM 使用者和角色。如需詳細資訊，請參閱允許金鑰管理員來管理 KMS 金鑰。

    > **Note**
    >
    > IAM 政策可授權其他 IAM 使用者和角色來使用 KMS 金鑰。
18. (選用) 若要防止這些金鑰管理員刪除此 KMS 金鑰，請清除 Allow key administrators to delete this key (允許金鑰管理員刪除此金鑰) 核取方塊。

    刪除 KMS 金鑰是一種破壞性和不可復原的操作，可可能導致密文無法復原。即使您擁有外部金鑰材料，也無法在外部金鑰存放區中重新建立對稱 KMS 金鑰。不過，刪除 KMS 金鑰不會影響其相關聯的外部金鑰。如需從外部金鑰存放區刪除 KMS 金鑰的相關資訊，請參閱 排程從外部金鑰存放區刪除 KMS 金鑰。

19. 選擇下一步。
20. 在 This account (這個帳戶) 區段中，選取此 AWS 帳戶 中可在密碼編譯操作中使用 KMS 金鑰的 IAM 使用者和角色。如需詳細資訊，請參閱允許金鑰使用者使用 KMS 金鑰。

    > **Note**
    >
    > IAM 政策可授權其他 IAM 使用者和角色來使用 KMS 金鑰。

21. (選用) 您可以允許其他 AWS 帳戶 將此 KMS 金鑰用於密碼編譯操作。若要這樣做，請在頁面底部的其他 AWS 帳戶 區段中，選擇新增另一個 AWS 帳戶，然後輸入外部帳戶的 AWS 帳戶 ID。若要新增多個外部帳戶，請重複此步驟。

    > **Note**
    >
    > 其他 AWS 帳戶 的管理員也必須透過為其使用者建立 IAM 政策，來允許存取 KMS 金鑰。如需詳細資訊，請參閱 允許其他帳戶中的使用者使用 KMS 金鑰。

22. 選擇下一步。
23. 檢閱您選擇的金鑰設定。您仍然可以返回並變更所有設定。
24. 完成時，請選擇 Finish (完成) 以建立金鑰。

當處理程序成功時，畫面會在您選擇的外部金鑰存放區中顯示新的 KMS 金鑰。當您選擇新 KMS 金鑰的名稱或別名時，其詳細資訊頁面上的 Cryptographic configuration (密碼編譯組態) 標籤會顯示 KMS 金鑰的來源 (External key store (外部金鑰存放區))、名稱、ID、自訂金鑰存放區類型，以及外部金鑰的 ID、金鑰用途和狀態。如果程序失敗，則會出現錯誤訊息來描述失敗。若為 ，請參閱 外部金鑰存放區故障診斷。

> **Tip**
>
> 為了更輕鬆識別自訂金鑰存放區中的 KMS 金鑰，請在 Customer managed keys (客戶受管金鑰) 頁面上，新增要顯示的 Origin (來源) 和 Custom key store ID (自訂金鑰存放區 ID) 資料欄。若要變更資料表欄位，請選擇頁面右上角的齒輪圖示。如需詳細資訊，請參閱 自訂您的 KMS 金鑰資料表。

在外部金鑰存放區建立 KMS 金鑰 (AWS KMS API)

若要在外部金鑰存放區中建立新的 KMS 金鑰，請使用此CreateKey作業。下列是必要參數：

- Origin 值必須為 EXTERNAL_KEY_STORE。
- CustomKeyStoreId 參數可識別您的外部金鑰存放區。指定的外部金鑰存放區的 ConnectionState 必須為 CONNECTED。若要尋找 CustomKeyStoreId 和 ConnectionState，請使用 DescribeCustomKeyStores 操作。
- XksKeyId 參數可識別外部金鑰。此外部金鑰必須滿足要求，才能與 KMS 金鑰相關聯。

您也可以使用 CreateKey 操作的任何選用參數，例如使用 Policy 或 Tags (標籤) 參數。

> **Note**
>
> 請勿在 Description 或 Tags 欄位包含機密或敏感資訊。在 CloudTrail 記錄檔和其他輸出中，這些欄位可能會以純文字顯示。

本節中的範例使用 AWS Command Line Interface (AWS CLI)，但您可以使用任何支援的程式設計語言。

此範例命令使用此CreateKey作業在外部金鑰存放區中建立 KMS 金鑰。回應包括 KMS 金鑰的屬性、外部金鑰存放區 ID 以及外部金鑰的 ID、用途和狀態。如需有關這些欄位的詳細資訊，請參閱 檢視外部金鑰存放區中的 KMS 金鑰。

執行此命令之前，請將範例自訂金鑰存放區 ID 換成有效的 ID。

```
$ aws kms create-key --origin EXTERNAL_KEY_STORE --custom-key-store-
id cks-1234567890abcdef0 --xks-key-id bb8562717f809024
{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333",
    "CreationDate": "2022-12-02T07:48:55-07:00",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "CustomKeyStoreId": "cks-1234567890abcdef0",
    "Description": "",
    "Enabled": true,
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ],
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
```

```
    "KeyManager": "CUSTOMER",
    "KeySpec": "SYMMETRIC_DEFAULT",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": false,
    "Origin": "EXTERNAL_KEY_STORE",
    "XksKeyConfiguration": {
      "Id": "bb8562717f809024"
    }
  }
}
```

檢視外部金鑰存放區中的 KMS 金鑰

若要檢視外部金鑰存放區中的 KMS 金鑰，請使用AWS KMS主控台或DescribeKey作業。您可以使用與檢視任何 AWS KMS 客戶受管金鑰相同的技巧。若要學習基本操作，請參閱檢視金鑰。

在 AWS KMS 主控台中，除了 AWS 帳戶 和區域中的其他所有客戶受管金鑰，外部金鑰存放區中的 KMS 金鑰也會顯示在客戶受管金鑰頁面。若要識別外部金鑰存放區中的 KMS 金鑰，請按照獨特的來源值、External key store (外部金鑰存放區) 和自訂金鑰存放區 ID 進行篩選。

如需詳細資訊，請參閱 檢視外部金鑰存放區、監控外部金鑰存放區 及 使用 AWS CloudTrail 記錄 AWS KMS API 呼叫。

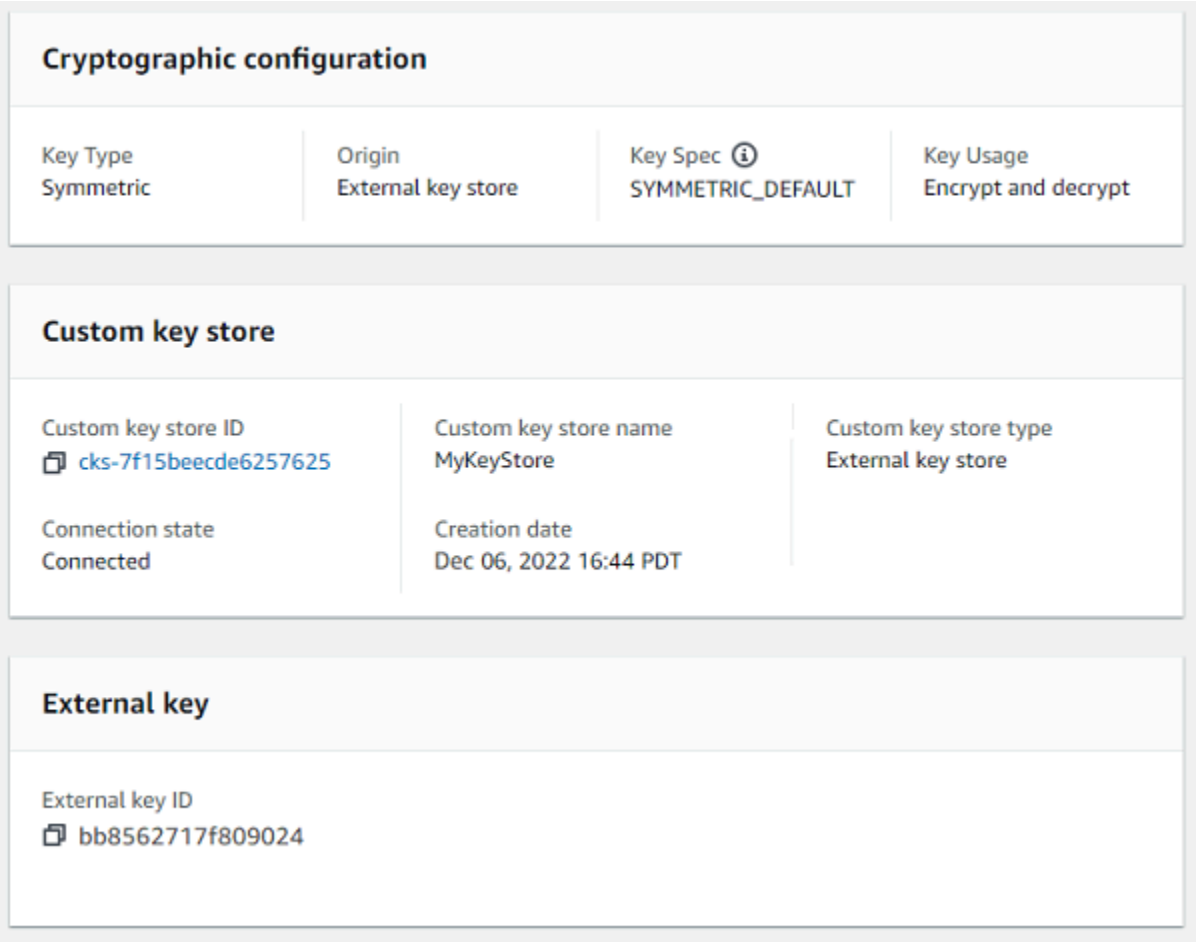
主題

- 外部金鑰存放區中 KMS 金鑰的屬性
- 檢視外部金鑰存放區中的 KMS 金鑰 (主控台)
- 檢視外部金鑰存放區中的 KMS 金鑰 (AWS KMS API)

外部金鑰存放區中 KMS 金鑰的屬性

與所有 KMS 金鑰一樣，外部金鑰存放區中的 KMS 金鑰具有金鑰 ARN、金鑰規格和金鑰用途值，但它們也具有外部金鑰存放區中 KMS 金鑰特有的屬性和屬性值。例如，外部金鑰存放區中所有 KMS 金鑰的 Origin (來源) 值為 External key store (外部金鑰存放區)。

對於外部金鑰存放區中的 KMS 金鑰，AWS KMS 主控台中的 Cryptographic configuration (密碼編譯組態) 索引標籤包含兩個其他區段：Custom key store (自訂金鑰存放區) 和 External key (外部金鑰)。



自訂金鑰存放區屬性

下列值會顯示在 [密碼編譯組態] 索引標籤的 [自訂金鑰存放區] 區段中，以及DescribeKey回應中。這些屬性適用於所有自訂金鑰存放區，包括 AWS CloudHSM 金鑰存放區和外部金鑰存放區。

自訂金鑰存放區 ID

AWS KMS 指派給自訂金鑰存放區的唯一 ID。

自訂金鑰存放區名稱

您在建立自訂金鑰存放區時指派給它的易記名稱。您可隨時變更此值。

自訂金鑰存放區類型

自訂金鑰存放區的類型。有效值為 AWS CloudHSM (`AWS_CLOUDHSM`) 或外部金鑰存放區 (`EXTERNAL_KEY_STORE`)。您無法在建立自訂金鑰存放區之後變更類型。

建立日期

建立自訂金鑰存放區的日期。此日期顯示為 AWS 區域 的本地時間。

連線狀態

指示自訂金鑰存放區是否已連接至其備份金鑰存放區。只有當自訂金鑰存放區從未連接至其備份金鑰存放區，或已故意中斷連接時，連接狀態才為 DISCONNECTED。如需詳細資訊，請參閱 the section called “連線狀態”。

外部金鑰屬性

外部索引鍵內容會顯示在 [密碼編譯組態] 索引標籤的 [外部索引鍵] 區段中，以及DescribeKey回應的XksKeyConfiguration項目中。

External key (外部金鑰) 區段只會針對外部金鑰存放區中的 KMS 金鑰顯示在 AWS KMS 主控台中。它提供與 KMS 金鑰相關聯的外部金鑰的資訊。外部金鑰是 AWS 之外的密碼編譯金鑰，可作為外部金鑰存放區中 KMS 金鑰的金鑰材料。當您使用 KMS 金鑰進行加密或解密時，外部金鑰管理器將使用指定的外部金鑰執行操作。

下列值會顯示在 External key (外部金鑰) 區段中。

外部金鑰 ID

外部金鑰在其外部金鑰管理器中的識別符。這是外部金鑰存放區代理用來識別外部金鑰的值。您可以在建立 KMS 金鑰時指定外部金鑰 ID，並且無法變更它。如果用來建立 KMS 金鑰的外部金鑰 ID 值變更或變得無效，您必須排程要刪除的 KMS 金鑰，並使用正確的外部金鑰 ID 值建立新的 KMS 金鑰。

檢視外部金鑰存放區中的 KMS 金鑰 (主控台)

檢視外部金鑰存放區中的 KMS 金鑰 (主控台)

1. 開啟位於 https://console.aws.amazon.com/kms 的 AWS KMS 主控台。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 若要識別外部金鑰存放區中的 KMS 金鑰，請將 Origin (來源) 和 Custom key store ID (自訂金鑰存放區 ID) 欄位新增至金鑰資料表。任何外部金鑰存放區中的 KMS 金鑰都具有 External key store (外部金鑰存放區) 的 Origin (來源) 值。

   在右上角，選擇齒輪圖示，選擇 Origin (來源) 和 Custom key store ID (自訂金鑰存放區 ID)，然後選擇 Confirm (確認)。

5. 選擇外部金鑰存放區中 KMS 金鑰的別名或金鑰 ID。
6. 若要檢視外部金鑰存放區中 KMS 金鑰的特定屬性，請選擇 Cryptographic configuration (密碼編譯組態) 索引標籤。外部金鑰存放區中 KMS 金鑰的特殊值會顯示在 Custom key store (自訂金鑰存放區) 和 External key (外部金鑰) 區段中。

檢視外部金鑰存放區中的 KMS 金鑰 (AWS KMS API)

檢視外部金鑰存放區中的 KMS 金鑰 (API)

您可以使用相同的 AWS KMS API 作業來檢視外部金鑰存放區中的 KMS 金鑰，以供任何 KMS 金鑰使用 ListKeys，包括DescribeKey、和GetKeyPolicy。例如，AWS CLI 中的以下 `describe-key` 操作會顯示外部金鑰存放區中 KMS 金鑰的特殊欄位。執行像這樣的命令之前，請將範例 KMS 金鑰 ID 換成有效值。

```
$ aws kms describe-key --key-id 1234abcd-12ab-34cd-56ef-1234567890ab
{
  "KeyMetadata": {
    "Arn": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "AWSAccountId": "111122223333",
    "CreationDate": "2022-12-02T07:48:55-07:00",
    "CustomerMasterKeySpec": "SYMMETRIC_DEFAULT",
    "CustomKeyStoreId": "cks-1234567890abcdef0",
    "Description": "",
    "Enabled": true,
    "EncryptionAlgorithms": [
      "SYMMETRIC_DEFAULT"
    ],
    "KeyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
    "KeyManager": "CUSTOMER",
    "KeySpec": "SYMMETRIC_DEFAULT",
    "KeyState": "Enabled",
    "KeyUsage": "ENCRYPT_DECRYPT",
    "MultiRegion": false,
    "Origin": "EXTERNAL_KEY_STORE",
    "XksKeyConfiguration": {
      "Id": "bb8562717f809024"
    }
  }
}
```

使用外部金鑰存放區中的 KMS 金鑰

在外部金鑰存放區中建立對稱加密 KMS 金鑰之後，您可以將其用於下列密碼編譯操作：

- 加密
- 解密
- GenerateDataKey
- GenerateDataKeyWithoutPlaintext
- ReEncrypt

自訂金鑰存放區不支援產生非對稱資料金鑰配對GenerateDataKeyPair和GenerateDataKeyPairWithoutPlaintext的對稱加密作業。

在外部金鑰存放區中使用 KMS 金鑰的所有密碼編譯操作都支援加密內容。與往常一樣，使用加密內容是 AWS KMS 建議的安全最佳實務。

在請求中使用 KMS 金鑰時，請按照其金鑰 ID、金鑰 ARN、別名或別名 ARN 來識別 KMS 金鑰。您不需要指定外部金鑰存放區。回應包含為針對任何對稱加密 KMS 金鑰所傳回的相同欄位。不過，當您在外部金鑰存放區中使用 KMS 金鑰時，外部金鑰管理器將使用與 KMS 金鑰關聯的外部金鑰來執行加密和解密操作。

為了確保由外部金鑰存放區中的 KMS 金鑰加密的密文至少與使用標準 KMS 金鑰加密的任何密文一樣安全，AWS KMS 會使用雙重加密。首先在 AWS KMS 中使用 AWS KMS 金鑰材料加密資料。然後，您的外部金鑰管理器會使用 KMS 金鑰的外部金鑰對其進行加密。若要解密雙重加密的密文，您的外部金鑰管理器會先使用 KMS 金鑰的外部金鑰對密文進行解密。然後，在 AWS KMS 中使用 KMS 金鑰的 AWS KMS 金鑰材料對它進行解密。

但必須符合以下條件，才有可能這樣做。

- KMS 金鑰的金鑰狀態必須是 `Enabled`。若要尋找金鑰狀態，請參閱客戶管理金鑰AWS KMS主控台的 [狀態`KeyState`] 欄位或DescribeKey回應中的欄位。
- 託管 KMS 金鑰的外部金鑰存放區必須連接至其外部金鑰存放區代理，也就是說，外部金鑰存放區的連接狀態必須為 `CONNECTED`。

  您可以在AWS KMS主控台或DescribeCustomKeyStores回應中的 [外部金鑰存放區] 頁面上檢視連線狀態。外部金鑰存放區的連接狀態也顯示在 AWS KMS 主控台 KMS 金鑰的詳細資訊頁面上。在詳細資訊頁面上，選擇 Cryptographic configuration (密碼編譯組態) 索引標籤，並查看 Custom key store (自訂金鑰存放區) 區段中的 Connection state (連接狀態) 欄位。

如果連接狀態為 DISCONNECTED，則必須先將其連接。如果連接狀態為 FAILED，則您必須解決問題，中斷連接外部金鑰存放區，然後進行連接。如需說明，請參閱 連接和中斷連接外部金鑰存放區。

- 外部金鑰存放區代理必須能夠找到外部金鑰。
- 必須啟用外部金鑰，而且必須執行加密和解密。

外部金鑰的狀態獨立於 KMS 金鑰的金鑰狀態變更，並不受其影響，包括啟用和停用 KMS 金鑰。同樣，停用或刪除外部金鑰不會變更 KMS 金鑰的金鑰狀態，但使用關聯 KMS 金鑰的密碼編譯操作將會失敗。

如果不符合這些條件，密碼編譯操作會失敗，且 AWS KMS 會傳回 KMSInvalidStateException 例外狀況。您可能需要重新連接外部金鑰存放區，或使用外部金鑰管理器工具來重新設定或修復外部金鑰。如需其他說明，請參閱the section called “外部金鑰存放區故障診斷”。

在外部金鑰存放區中使用 KMS 金鑰時，請注意每個外部金鑰存放區中的 KMS 金鑰會針對密碼編譯操作共用自訂金鑰存放區請求配額。如果您超過配額，則 AWS KMS 會傳回 ThrottlingException。如需自訂金鑰存放區配額的詳細資訊，請參閱 自訂金鑰存放區請求配額。

排程從外部金鑰存放區刪除 KMS 金鑰

當您確定不再需要將 AWS KMS key 用於任何密碼編譯操作時，您可以排程刪除 KMS 金鑰。就像您排定從 AWS KMS 刪除任何 KMS 金鑰一樣，使用同樣的處理程序。從外部金鑰存放區刪除 KMS 金鑰不會影響作為其金鑰材料的外部金鑰。

您可以在其強制等待期間，取消 KMS 金鑰的排程刪除。不過，已刪除的 KMS 金鑰無法復原。即使您使用相同的外部金鑰，也無法在外部金鑰存放區中重新建立對稱加密 KMS 金鑰。由於外部金鑰存放區中的每個對稱 KMS 金鑰都有唯一的 AWS KMS 金鑰材料和中繼資料，因此只有已加密對稱密文的 AWS KMS 金鑰才能將其解密。

> **Warning**
>
> 刪除 KMS 金鑰是一種破壞性和具有潛在危險的操作，您將無法復原以 KMS 金鑰加密的所有資料。在排程刪除 KMS 金鑰之前，請檢查 KMS 金鑰的過去使用情況，並建立 Amazon CloudWatch 警示，以便在有人嘗試使用 KMS 金鑰擱置刪除時向您發出警示。儘可能停用 KMS 金鑰，而不要刪除。

排程從外部金鑰存放區刪除 KMS 金鑰時，其金鑰狀態會變更為 Pending deletion (等待刪除)。KMS 金鑰會在整個等待期保持在 Pending deletion (等待刪除) 狀態，即使 KMS 金鑰變為無法使用，因為您已中斷連接外部金鑰存放區。這可讓您在等待期間隨時取消刪除 KMS 金鑰。當等待期過期時，AWS KMS 會從 AWS KMS 刪除 KMS 金鑰。

當您排程從外部金鑰存放區刪除 KMS 金鑰時，KMS 金鑰會立即變為無法使用 (視最終一致性而定)。不過，使用受 KMS 金鑰保護之資料金鑰所加密的資源不會受影響，除非再次使用 KMS 金鑰 (例如解密資料金鑰)。此問題會影響 AWS 服務，其中許多服務會使用資料金鑰來保護您的資源。如需詳細資訊，請參閱 無法使用的 KMS 金鑰如何影響資料金鑰。

您可以監控 AWS CloudTrail 日誌中 KMS 金鑰的排程、取消和刪除。

外部金鑰存放區故障診斷

大多數外部金鑰存放區問題的解決方案由 AWS KMS 顯示的每個例外狀況的錯誤訊息表示，或由 AWS KMS 在嘗試將外部金鑰存放區連接至其外部金鑰存放區代理失敗時傳回的連接錯誤代碼表示。但是，有些問題有點複雜。

診斷外部金鑰存放區的問題時，請先找出原因。這將縮小補救措施的範圍，並使您更有效地進行故障排除。

- AWS KMS – 問題可能在 AWS KMS 中，例如外部金鑰存放區組態中的值不正確。
- 外部 – 問題可能源於 AWS KMS 外部，包括外部金鑰存放區代理、外部金鑰管理器、外部金鑰或 VPC 端點服務的組態或操作發生問題。
- 聯網 – 這可能是連接或聯網問題，例如代理端點、連接埠或私有 DNS 名稱或網域發生問題。

> **Note**
>
> 當外部金鑰存放區上的管理操作失敗時，其會產生數個不同的例外狀況。但是，對於與外部金鑰存放區的外部組態或連接狀態相關的所有失敗，AWS KMS 密碼編譯操作會傳回 `KMSInvalidStateException`。若要識別問題，請使用隨附的錯誤訊息文字。
> 在連線程序完成之前，ConnectCustomKeyStore作業會迅速成功。若要確定連接程序是否成功，請檢視外部金鑰存放區的連接狀態。如果連接程序失敗，則 AWS KMS 會傳回連接錯誤代碼，說明原因並建議解決方法。

主題

- 外部金鑰存放區的故障診斷工具

- 組態錯誤
- 外部金鑰存放區連接錯誤
- 延遲和逾時錯誤
- 身分驗證憑證錯誤
- 金鑰狀態錯誤
- 解密錯誤
- 外部金鑰錯誤
- 代理問題
- 代理授權問題

外部金鑰存放區的故障診斷工具

AWS KMS 提供數種工具，協助您識別並解決外部金鑰存放區及其金鑰的問題。將這些工具與外部金鑰存放區代理和外部金鑰管理器隨附的工具搭配使用。

> **Note**
>
> 您的外部金鑰存放區代理和外部金鑰管理器可能會提供更簡單的方法來建立和維護外部金鑰存放區及其 KMS 金鑰。如需詳細資訊，請參閱外部工具的文件。

AWS KMS 例外狀況和錯誤訊息

AWS KMS 提供有關它遇到的任何問題的詳細錯誤訊息。您可以在 AWS Key Management ServiceAPI 參考和 AWS SDK 中找到有關 AWS KMS 例外狀況的其他資訊。即使您正在使用 AWS KMS 主控台，您也可能會發現這些參考很有幫助。例如，請參閱 `CreateCustomKeyStores` 操作的錯誤清單。

如果問題出現在不同的 AWS 服務中，例如當您在外部金鑰存放區中使用 KMS 金鑰來保護其他 AWS 服務中的資源時，AWS 服務可能會提供額外資訊來協助您識別問題。如果AWS服務未提供訊息，您可以在記錄 KMS 金鑰使用的記錄中檢視錯誤訊息。CloudTrail

CloudTrail 日誌

每個 AWS KMS API 操作 (包括 AWS KMS 主控台中的操作) 都記錄在 AWS CloudTrail 日誌中。AWS KMS 會記錄成功和失敗操作的日誌項目。對於失敗的操作，日誌項目會包含 AWS KMS 例外狀況名稱 (`errorCode`) 和錯誤訊息 (`errorMessage`)。您可以使用此資訊來協助您識別和解決錯誤。如需範例，請參閱 使用外部金鑰存放區中的 KMS 金鑰進行解密失敗。

日誌項目也包含請求 ID。如果請求到達外部金鑰存放區代理，則您可以使用日誌項目中的請求 ID，在代理日誌中尋找相應的請求 (如果代理提供的話)。

CloudWatch 度量

AWS KMS記錄有關外部金鑰存放區的操作和效能的詳細 Amazon CloudWatch 指標，包括延遲、節流、Proxy 錯誤、外部金鑰管理員狀態、TLS 憑證到期前的天數，以及代理身份驗證登入資料的報告保留時間。您可以使用這些指標為外部金鑰存放區的操作開發資料模型，並在問題發生之前提醒您即將發生的問題的 CloudWatch 警示。

> **Important**
>
> AWS KMS建議您建立 CloudWatch 警示以監視外部金鑰存放區指標。這些警報會在問題發生之前提醒您早期跡象。

監控圖表

AWS KMS在AWS KMS主控台中每個外部金鑰存放區的詳細資料頁面上，顯示外部金鑰存放區 CloudWatch 測量結果的圖形。您可以使用圖表中的資料來協助尋找錯誤來源、偵測即將發生的問題、建立基準線，以及調整 CloudWatch 警示臨界值。如需有關解釋監控圖表及使用其資料的詳細資訊，請參閱 監控外部金鑰存放區。

顯示外部金鑰存放區和 KMS 金鑰

AWS KMS會在主控台的外部金鑰存放區中顯示外部金鑰存放區和 KMS 金鑰的詳細資AWS KMS訊，以及對DescribeCustomKeyStores和DescribeKey作業的回應。這些顯示包括外部金鑰存放區和 KMS 金鑰的特殊欄位，以及可用於進行故障診斷的資訊，例如外部金鑰存放區的連接狀態，以及與 KMS 金鑰相關聯的外部金鑰 ID。如需詳細資訊，請參閱 檢視外部金鑰存放區 和 檢視外部金鑰存放區中的 KMS 金鑰。

XKS 代理測試用戶端

AWS KMS 提供開放原始碼測試用戶端，以驗證您的外部金鑰存放區代理是否符合 AWS KMS 外部金鑰存放區代理 API 規格。您可以使用此測試用戶端來識別並解決外部金鑰存放區代理的問題。

組態錯誤

建立外部金鑰存放區時，您可以指定組成外部金鑰存放區組態的屬性值，例如代理身分驗證憑證、代理 URI 端點、代理 URI 路徑以及 VPC 端點服務名稱。當 AWS KMS 偵測到屬性值中的錯誤時，操作會失敗，並傳回指出錯誤值的錯誤。

許多組態問題可以透過修正不正確的值來解決。您可以修正無效的代理 URI 路徑或代理身分驗證憑證，而無需中斷連接外部金鑰存放區。如需這些值的定義，包括唯一性要求，請參閱 備妥先決條件。如需有關更新這些值的指示，請參閱 編輯外部金鑰存放區屬性。

若要避免代理 URI 路徑和代理身分驗證憑證值發生錯誤，在建立或更新外部金鑰存放區時，請將代理組態檔案上傳至 AWS KMS 主控台。這是一個基於 JSON 的檔案，其中包含代理 URI 路徑和代理身分驗證憑證值，由外部金鑰存放區代理或外部金鑰管理器提供。您無法將代理組態檔案用於 AWS KMS API 操作，但可以使用檔案中的值來協助您為 API 請求提供與代理中的值相符的參數值。

一般組態錯誤

例外狀況：`CustomKeyStoreInvalidStateException` (`CreateKey`)、`KMSInvalidStateException` (密碼編譯操作)、`XksProxyInvalidConfigurationException` (管理操作，`CreateKey` 除外)

連接錯誤代碼：`XKS_PROXY_INVALID_CONFIGURATION`、`XKS_PROXY_INVALID_TLS_CONFIGURATION`

對於具有公有端點連接的外部金鑰存放區，AWS KMS 會在建立和更新外部金鑰存放區時測試屬性值。對於具有 VPC 端點服務連接的外部金鑰存放區，AWS KMS 會在連接和更新外部金鑰存放區時測試屬性值。

> **Note**
>
> 即使嘗試將外部金鑰存放區連接至其外部金鑰存放區代理失敗，非同步的 `ConnectCustomKeyStore` 操作仍可能成功。在這種情況下，沒有例外狀況，但是外部金鑰存放區的連接狀態為 Failed (失敗)，連接錯誤代碼會說明錯誤訊息。如需詳細資訊，請參閱 外部金鑰存放區連接錯誤。

如果 AWS KMS 偵測到屬性值中的錯誤，則操作會失敗並傳回 `XksProxyInvalidConfigurationException` 及下列其中一個錯誤訊息。

由於 URI 路徑無效，外部金鑰存放區代理拒絕請求。驗證外部金鑰存放區的 URI 路徑，並在必要時更新。

- 代理 URI 路徑是代理 API 的 AWS KMS 請求的基本路徑。如果此路徑不正確，則對代理的所有請求都會失敗。若要檢視外部金鑰存放區的當前代理 URI 路徑，請使用 AWS KMS 主控台或

DescribeCustomKeyStores 操作。若要尋找正確的代理 URI 路徑，請參閱您的外部金鑰存放區代理文件。如需有關修正代理 URI 路徑值的說明，請參閱 編輯外部金鑰存放區屬性。

- 外部金鑰存放區代理的代理 URI 路徑可能會隨著外部金鑰存放區代理或外部金鑰管理器的更新而變更。如需有關這些變更的資訊，請參閱外部金鑰存放區代理或外部金鑰管理器的文件。

XKS_PROXY_INVALID_TLS_CONFIGURATION
AWS KMS 無法建立與外部金鑰存放區代理的 TLS 連接。驗證 TLS 組態，包括其憑證。

- 所有外部金鑰存放區代理都需要 TLS 憑證。TLS 憑證必須由外部金鑰存放區支援的公有憑證授權機構核發。如需支援的 CA 清單，請參閱「AWS KMS 外部金鑰存放區代理 API 規格」中的受信任憑證授權機構。
- 若為公有端點連接，TLS 憑證上的主體通用名稱 (CN) 必須與外部金鑰存放區代理之代理 URI 端點中的網域名稱相符。例如，如果公有端點為 https://myproxy.xks.example.com，則 TLS 憑證上的通用名稱必須為 myproxy.xks.example.com 或 *.xks.example.com。
- 若為 VPC 端點服務連接，TLS 憑證上的主體通用名稱 (CN) 必須與 VPC 端點服務的私有 DNS 名稱相符。例如，如果私有 DNS 名稱為 myproxy-private.xks.example.com，則 TLS 憑證上的通用名稱必須為 myproxy-private.xks.example.com 或 *.xks.example.com。
- TLS 憑證不能過期。若要取得 TLS 憑證的到期日，請使用 SSL 工具，例如 OpenSSL。若要監視與外部金鑰存放區關聯之 TLS 憑證的到期日，請使用XksProxyCertificateDaysToExpire CloudWatch 指標。TLS 憑證到期日的天數也會顯示在 AWS KMS 主控台的 Monitoring (監控) 區段中。
- 如果您使用的是公有端點連接，則請使用 SSL 測試工具來測試您的 SSL 組態。TLS 連接錯誤可能是因為不正確的憑證鏈結所導致。

VPC 端點服務連接組態錯誤

例外狀
況：XksProxyVpcEndpointServiceNotFoundException、XksProxyVpcEndpointServiceInvali

除了一般的連接問題之外，在使用 VPC 端點服務連接來建立、連接或更新外部金鑰存放區時，您可能會遇到下列問題。在建立、連接及更新外部金鑰存放區時，AWS KMS 會測試具有 VPC 端點服務連接的外部金鑰存放區的屬性值。當管理操作因組態錯誤而失敗時，會產生下列例外狀況：

XksProxyVpcEndpointServiceNotFound例外

原因可能為下列之一：

- 不正確的 VPC 端點服務名稱。請確認外部金鑰存放區的 VPC 端點服務名稱正確且符合外部金鑰存放區的代理 URI 端點值。若要尋找虛擬私人雲端端點服務名稱，請使用 Amazon VPC 主控台或操作DescribeVpcEndpointServices。若要尋找現有外部金鑰存放區的 VPC 端點服務名稱和 Proxy URI 端點，請使用主AWS KMS控台或DescribeCustomKeyStores作業。如需詳細資訊，請參閱 檢視外部金鑰存放區。
- VPC 端點服務可能位於不同於外部金鑰存放區的 AWS 區域 中。請確認 VPC 端點服務與外部金鑰存放區處於相同區域中。區域名稱的外部名稱，例如`us-east-1`，是虛擬私人 VPC 端點服務名稱的一部分，例如 vpce.us-東部 1。 vpce-svc-example。 ）如需外部金鑰存放區之 VPC 端點服務的要求清單，請參閱 VPC 端點服務。您無法將 VPC 端點服務或外部金鑰存放區移至不同區域。不過，您可以在與 VPC 端點服務相同的區域中建立新的外部金鑰存放區。如需詳細資訊，請參閱 設定 VPC 端點服務連接 和 建立外部金鑰存放區。
- AWS KMS 不是 VPC 端點服務允許的主體。VPC 端點服務的 Allow principals (允許主體) 清單必須包含 `cks.kms.<region>.amazonaws.com` 值，例如 `cks.kms.eu-west-3.amazonaws.com`。如需有關新增此值的指示，請參閱《AWS PrivateLink 指南》中的管理許可。

XksProxyVpcEndpointServiceInvalidConfiguration例外

當 VPC 端點服務無法滿足下列其中一項要求時，就會發生此錯誤：

- VPC 需要至少兩個私有子網路，每個處於不同的可用區域。如需有關將子網路新增至 VPC 的說明，請參閱《Amazon VPC 使用者指南》中的在 VPC 中建立子網路。
- 您的 VPC 端點服務類型必須使用網路負載平衡器，而非閘道負載平衡器。
- VPC 端點服務不要求接受 (Acceptance required (要求接受) 必須為 False)。如果需要手動接受每個連接請求，則 AWS KMS 無法使用 VPC 端點服務連接至外部金鑰存放區代理。如需詳細資訊，請參閱《AWS PrivateLink 指南》中的接受或拒絕連接請求。
- VPC 端點服務必須具有私有 DNS 名稱，它是公有網域的子網域。例如，如果私有 DNS 名稱為 `https://myproxy-private.xks.example.com`，則 `xks.example.com` 或 `example.com` 網域必須具有公有 DNS 伺服器。若要檢視或變更 VPC 端點服務的私有 DNS 名稱，請參閱《AWS PrivateLink 指南》中的管理 VPC 端點服務的 DNS 名稱。

- 私有 DNS 名稱網域的 Domain verification status (網域驗證狀態) 必須為 `verified`。若要檢視和更新私有 DNS 名稱網域的驗證狀態，請參閱 驗證您的私有 DNS 名稱網域。新增必要的文字記錄後，可能需要幾分鐘的時間才會顯示更新的驗證狀態。

  > **Note**
  >
  > 只有當私有 DNS 網域是公有網域的子網域時，才能驗證它。否則，即使新增所需的 TXT 記錄後，私有 DNS 網域的驗證狀態也不會變更。

- VPC 端點服務的私有 DNS 名稱必須與外部金鑰存放區的代理 URI 端點值相符。對於具有 VPC 端點服務連接的外部金鑰存放區，代理 URI 端點必須為 `https://`，後面為 VPC 端點服務的私有 DNS 名稱。若要檢視代理 URI 端點值，請參閱 檢視外部金鑰存放區。若要變更代理 URI 端點值，請參閱 編輯外部金鑰存放區屬性。

外部金鑰存放區連接錯誤

將外部金鑰存放區連接至其外部金鑰存放區代理的程序大約需要五分鐘才能完成。除非其快速失敗，否則 `ConnectCustomKeyStore` 操作會傳回 HTTP 200 回應和不帶屬性的 JSON 物件。不過，這個初始回應並不表示連接已成功。若要判斷外部金鑰存放區是否已連接，請參閱其連接狀態。如果連接失敗，外部金鑰存放區的連接狀態會變更為 `FAILED`，並且 AWS KMS 會傳回解釋失敗原因的連接錯誤代碼。

> **Note**
>
> 如果自訂金鑰存放區的狀態為 `FAILED`，在嘗試重新連接之前，您必須中斷連接自訂金鑰存放區。您無法連接具有 `FAILED` 連接狀態的自訂金鑰存放區。

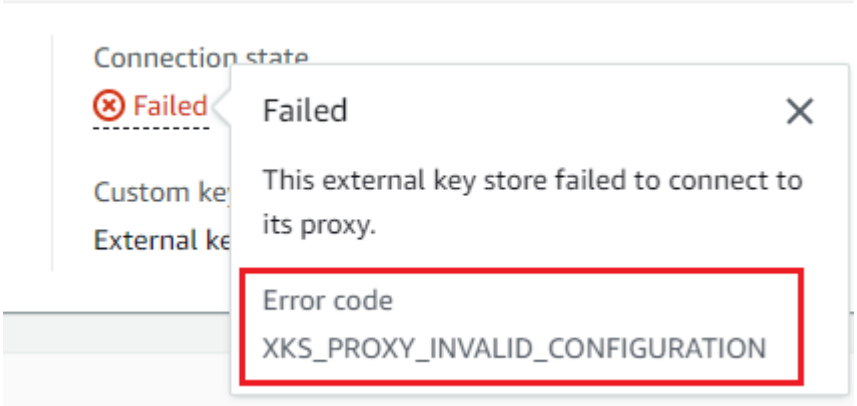
若要檢視外部金鑰存放區的連接狀態：

- 在DescribeCustomKeyStores回應中，檢視`ConnectionState`元素的值。
- 在 AWS KMS 主控台中，Connection state (連接狀態) 會顯示在外部金鑰存放區資料表中。此外，在每個外部金鑰存放區的詳細資訊頁面上，Connection state (連接狀態) 會顯示在 General configuration (一般組態) 區段中。

當連接狀態為 `FAILED` 時，連接錯誤代碼有助於解釋錯誤。

若要檢視連接錯誤代碼：

- 在DescribeCustomKeyStores回應中，檢視ConnectionErrorCode元素的值。只有當 ConnectionState 為 FAILED 時，此元素才會出現在 DescribeCustomKeyStores 回應中。
- 若要在 AWS KMS 主控台中檢視連接錯誤代碼，請在外部金鑰存放區的詳細資訊頁面上，將滑鼠游標移至 Failed (失敗) 值上。



外部金鑰存放區的連接錯誤代碼

下列連接錯誤代碼適用於外部金鑰存放區

INTERNAL_ERROR

由於內部錯誤，AWS KMS 無法完成請求。重試 請求。若為 ConnectCustomKeyStore 請求，請先中斷連接自訂金鑰存放區，再重試連接。

INVALID_CREDENTIALS

在指定的外部金鑰存放區代理上，一個或兩個 XksProxyAuthenticationCredential 值無效。

NETWORK_ERRORS

網路錯誤導致 AWS KMS 無法將自訂金鑰存放區連接至其備份金鑰存放區。

XKS_PROXY_ACCESS_DENIED

AWS KMS 請求被拒絕存取外部金鑰存放區代理。如果外部金鑰存放區代理有授權規則，請確認其允許 AWS KMS 代表您與代理通訊。

XKS_PROXY_INVALID_CONFIGURATION

組態錯誤導致外部金鑰存放區無法連接到其代理。驗證 XksProxyUriPath 的值。

`XKS_PROXY_INVALID_RESPONSE`

AWS KMS 無法解譯來自外部金鑰存放區代理的回應。如果您重複看到此連接錯誤代碼，則請通知您的外部金鑰存放區代理廠商。

`XKS_PROXY_INVALID_TLS_CONFIGURATION`

由於 TLS 組態無效，AWS KMS 無法連接至外部金鑰存放區代理。確認外部金鑰存放區代理支援 TLS 1.2 或 1.3。此外，請確認 TLS 憑證尚未過期，與 `XksProxyUriEndpoint` 值中的主機名稱相符，並且由受信任憑證授權機構清單中所包含的受信任憑證授權機構簽署。

`XKS_PROXY_NOT_REACHABLE`

AWS KMS 無法與外部金鑰存放區代理通訊。確認 `XksProxyUriEndpoint` 和 `XksProxyUriPath` 正確無誤。使用外部金鑰存放區代理的工具來驗證代理處於作用中狀態且可在其網路上使用。此外，請確認您的外部金鑰管理器執行個體運作正常。如果代理報告所有外部金鑰管理器執行個體都無法使用，則連接嘗試失敗，並顯示此連接錯誤代碼。

`XKS_PROXY_TIMED_OUT`

AWS KMS 可以連接至外部金鑰存放區代理，但代理在規定時間內不會回應 AWS KMS。如果您重複看到此連接錯誤代碼，則請通知您的外部金鑰存放區代理廠商。

`XKS_VPC_ENDPOINT_SERVICE_INVALID_CONFIGURATION`

Amazon VPC 端點服務組態不符合 AWS KMS 外部金鑰存放區的要求。

- VPC 端點服務必須是呼叫者 AWS 帳戶 中的介面端點的端點服務。
- 其必須擁有已連接到至少兩個子網路的網路負載平衡器 (NLB)，每個位於不同的可用區域中。
- `Allow principals` 清單必須包含區域 (`cks.kms.<region>.amazonaws.com`) 的 AWS KMS 服務主體，例如 `cks.kms.us-east-1.amazonaws.com`。
- 其不得要求接受連接請求。
- 其必須擁有私有 DNS 名稱。具有 `VPC_ENDPOINT_SERVICE` 連接的外部金鑰存放區的私有 DNS 名稱在其 AWS 區域 中必須唯一。
- 私有 DNS 名稱網域的驗證狀態必須為 `verified`。
- TLS 憑證會指定可連接端點的私有 DNS 主機名稱。

`XKS_VPC_ENDPOINT_SERVICE_NOT_FOUND`

AWS KMS 找不到其用來與外部金鑰存放區代理通訊的 VPC 端點服務。確認 `XksProxyVpcEndpointServiceName` 正確無誤，且 AWS KMS 服務主體在 Amazon VPC 端點服務上具有服務消費者許可。

延遲和逾時錯誤

例外狀況：CustomKeyStoreInvalidStateException (CreateKey)、KMSInvalidStateException (密碼編譯操作)、XksProxyUriUnreachableException (管理操作)

連接錯誤代碼：XKS_PROXY_NOT_REACHABLE、XKS_PROXY_TIMED_OUT

當 AWS KMS 無法在 250 毫秒逾時間隔內聯絡代理時，它會傳回例外狀況。CreateCustomKeyStore 和 UpdateCustomKeyStore 會傳回 XksProxyUriUnreachableException。密碼編譯操作會傳回標準 KMSInvalidStateException，並顯示描述問題的錯誤訊息。如果 ConnectCustomKeyStore 失敗，AWS KMS 會傳回描述問題的連接錯誤代碼。

逾時錯誤可能是暫時性的問題，可透過重試請求來解決。如果問題依然存在，請確認外部金鑰存放區代理處於作用中狀態且已連接至網路，並且其代理 URI 端點、代理 URI 路徑以及 VPC 端點服務名稱 (如果有的話) 在外部金鑰存放區中正確無誤。此外，請確認您的外部金鑰管理器是否靠近 AWS 區域 外部金鑰存放區。如果需要更新任何這些值，請參閱 編輯外部金鑰存放區屬性。

若要追蹤延遲模式，請使用主控台 「監AWS KMS控」區段中的指標和「平均延遲」圖表 (根據該指標)。XksProxyLatency CloudWatch 您的外部金鑰存放區代理也可能會產生追蹤延遲和逾時的日誌和指標。

XksProxyUriUnreachableException
AWS KMS 無法與外部金鑰存放區代理通訊。這可能是暫時性的網路問題。如果您重複看到此錯誤，請確認外部金鑰存放區代理處於作用中狀態且已連接至網路，並且其端點 URI 在外部金鑰存放區中正確無誤。

- 外部金鑰存放區代理在 250 毫秒逾時間隔內未回應 AWS KMS 代理 API 請求。這可能表示代理發生暫時性的網路問題或者操作或效能問題。如果重試無法解決問題，請通知您的外部金鑰存放區代理管理員。

延遲和逾時錯誤通常顯示為連接失敗。當ConnectCustomKeyStore作業失敗時，外部金鑰存放區的連線狀態會變更為，FAILED並AWS KMS傳回說明錯誤的連線錯誤碼。如需連接錯誤代碼和解決錯誤的建議清單，請參閱 外部金鑰存放區的連接錯誤代碼。All custom key stores (所有自訂金鑰存放區) 和 External key stores (外部金鑰存放區) 的連接代碼清單適用於外部金鑰存放區。下列連接錯誤與延遲和逾時有關。

`XKS_PROXY_NOT_REACHABLE`

-或-

`CustomKeyStoreInvalidStateException` ，`KMSInvalidStateException` ，`XksProxyUriUnreachableException`
AWS KMS 無法與外部金鑰存放區代理通訊。請確認外部金鑰存放區代理處於作用中狀態且已連接至網路，以及外部金鑰存放區中的 URI 路徑和端點 URI 或 VPC 服務名稱正確無誤。

此錯誤可能發生的原因如下：

- 外部金鑰存放區代理處於非作用中狀態，或未連接至網路。
- 外部金鑰存放區組態中的代理 URI 端點、代理 URI 路徑或 VPC 端點服務名稱 (如果適用) 值中有錯誤。若要檢視外部金鑰存放區組態，請在主控台中使用DescribeCustomKeyStores作業或檢視外部金鑰存放區的詳細資料頁AWS KMS面。
- AWS KMS 與外部金鑰存放區代理之間的網路路徑上可能存在網路組態錯誤，例如連接埠錯誤。AWS KMS 與連接埠 443 上的外部金鑰存放區代理進行通訊。此值不可設定。
- 當外部密鑰存儲代理報告（在GetHealthStatus響應中）所有外部密鑰管理器實例都是時`UNAVAILABLE`，ConnectCustomKeyStore操作失敗，並顯示為`ConnectionErrorCode``XKS_PROXY_NOT_REACHABLE`。如需說明，請參閱外部金鑰管理器文件。
- 此錯誤可能是由於外部金鑰管理器與具有外部金鑰存放區的 AWS 區域 之間有很長的實際距離。AWS 區域 與外部金鑰管理器之間的 Ping 延遲 (網路封包來回時間 (RTT)) 不得超過 35 毫秒。您可能必須在更接近外部金鑰管理器的 AWS 區域 中建立外部金鑰存放區，或將外部金鑰管理器移至更接近 AWS 區域 的資料中心。

`XKS_PROXY_TIMED_OUT`

-或-

`CustomKeyStoreInvalidStateException` ，`KMSInvalidStateException` ，`XksProxyUriUnreachableException`
AWS KMS 拒絕請求，因為外部金鑰存放區代理沒有及時回應。重試 請求。如果您重複看到此錯誤，請向外部金鑰存放區代理管理員報告。

此錯誤可能發生的原因如下：

- 此錯誤可能是由於外部金鑰管理器與外部金鑰存放區代理之間有很長的實際距離。如果可能，請將外部金鑰存放區代理移至更接近外部金鑰管理器的位置。
- 當代理不是設計用來處理來自 AWS KMS 的請求數量和頻率時，就會發生逾時錯誤。如果您的指 CloudWatch 標指出存在持續性問題，請通知您的外部金鑰存放區 Proxy 管理員。
- 當外部金鑰管理器與外部金鑰存放區的 Amazon VPC 之間的連接未正常運作時，可能會發生逾時錯誤。如果您正在使用 AWS Direct Connect，則請確認您的 VPC 和外部金鑰管理器可以有效通訊。如需有關解決任何問題的說明，請參閱《AWS Direct Connect 使用者指南》中的 AWS Direct Connect 故障診斷。

`XKS_PROXY_TIMED_OUT`

-或-

`CustomKeyStoreInvalidStateException` ，`KMSInvalidStateException` ，`XksProxyUriUnreachableException`
外部金鑰存放區代理沒有在規定時間內回應請求。重試 請求。如果您重複看到此錯誤，請向外部金鑰存放區代理管理員報告。

- 此錯誤可能是由於外部金鑰管理器與外部金鑰存放區代理之間有很長的實際距離。如果可能，請將外部金鑰存放區代理移至更接近外部金鑰管理器的位置。

身分驗證憑證錯誤

例外狀況：`CustomKeyStoreInvalidStateException` (`CreateKey`)、`KMSInvalidStateException` (密碼編譯操作)、`XksProxyIncorrectAuthenticationCredentialException` (`CreateKey` 以外的管理操作)

您可以在外部金鑰存放區代理上建立並維護 AWS KMS 的身分驗證憑證。然後，在建立外部金鑰存放區時，將憑證值告知 AWS KMS。若要變更身分驗證憑證，請在外部金鑰存放區代理上進行變更。然後，更新外部金鑰存放區的憑證。如果代理輪換憑證，則必須更新外部金鑰存放區的憑證。

如果外部金鑰存放區代理無法驗證使用外部金鑰存放區的代理身分驗證憑證簽署的請求，則效果取決於請求：

- CreateCustomKeyStore 和 UpdateCustomKeyStore 因 XksProxyIncorrectAuthenticationCredentialException 失敗。
- ConnectCustomKeyStore 成功，但連接失敗。連接狀態為 FAILED，連接錯誤代碼為 INVALID_CREDENTIALS。如需詳細資訊，請參閱 外部金鑰存放區連接錯誤。
- 密碼編譯操作會針對外部金鑰存放區中的所有外部組態錯誤和連接狀態錯誤傳回 KMSInvalidStateException。隨附的錯誤訊息描述了問題。

外部金鑰存放區代理拒絕了請求，因為它無法對 AWS KMS 進行身分驗證。驗證外部金鑰存放區的憑證，並在必要時更新。

此錯誤可能發生的原因如下：

- 外部金鑰存放區的存取金鑰 ID 或私密存取金鑰與外部金鑰存放區代理上建立的值不符。

  若要修正此錯誤，請更新外部金鑰存放區的代理身分驗證憑證。您可以在不中斷外部金鑰存放區連接的情況下進行此變更。
- AWS KMS 和外部金鑰存放區代理之間的反向代理能夠以使 Sigv4 簽章無效的方式操作 HTTP 標頭。若要修正此錯誤，請通知代理管理員。

金鑰狀態錯誤

例外狀況：KMSInvalidStateException

KMSInvalidStateException 用於自訂金鑰存放區中 KMS 金鑰的兩種不同用途。

- 當管理操作 (例如 CancelKeyDeletion) 失敗並傳回此例外狀況時，表示 KMS 金鑰的金鑰狀態與操作不相容。
- 當自訂金鑰存放區中 KMS 金鑰的密碼編譯操作因 KMSInvalidStateException 失敗時，可能表示 KMS 金鑰的金鑰狀態有問題。但是，AWS KMS 密碼編譯操作會針對外部金鑰存放區中的所有外部組態錯誤和連接狀態錯誤傳回 KMSInvalidStateException。若要識別問題，請使用例外狀況隨附的錯誤訊息。

若要查找 AWS KMS API 操作所需的金鑰狀態，請參閱 AWS KMS 金鑰的金鑰狀態。若要尋找 KMS 金鑰的金鑰狀態，請在 Customer managed keys (客戶受管金鑰) 頁面上，檢視 KMS 金鑰的 Status

(狀態) 欄位。或者，使用DescribeKey操作並查看響應中的KeyState元素。如需詳細資訊，請參閱 檢視金鑰。

> **Note**
>
> 外部金鑰存放區中 KMS 金鑰的金鑰狀態不會指示與其關聯的外部金鑰的任何狀態資訊。如需有關外部金鑰狀態的資訊，請使用外部金鑰管理器和外部金鑰存放區代理工具。CustomKeyStoreInvalidStateException 指的是外部金鑰存放區的連接狀態，而不是 KMS 金鑰的金鑰狀態。

自訂存放區中 KMS 金鑰的密碼編譯操作可能會失敗，因為 KMS 金鑰的金鑰狀態為 Unavailable 或 PendingDeletion。(已停用的金鑰會傳回 DisabledException。)

- 只有在您故意停用DisabledAWS KMS主控台中的 KMS 金鑰或使用DisableKey作業時，KMS 金鑰才會有金鑰狀態。當 KMS 金鑰被停用時，您可以檢視和管理金鑰，但無法在密碼編譯操作中使用它。若要修正此問題，請啟用金鑰。如需詳細資訊，請參閱 啟用和停用金鑰。
- 當外部金鑰存放區與其外部金鑰存放區代理中斷連接時，KMS 金鑰的金鑰狀態為 Unavailable。若要修正無法使用的 KMS 金鑰，請重新連接外部金鑰存放區。外部金鑰存放區重新連接之後，外部金鑰存放區中 KMS 金鑰的金鑰狀態會自動還原到其先前的狀態，例如 Enabled 或 Disabled。

  當 KMS 金鑰已排程刪除且處於等待期間時，KMS 金鑰的金鑰狀態為 PendingDeletion。正在等待刪除的 KMS 金鑰的金鑰狀態錯誤表示不應該刪除該金鑰，因為它正用於加密，或者需要它進行解密。若要重新啟用 KMS 金鑰，請取消已排程的刪除，然後啟用金鑰。如需詳細資訊，請參閱 排程和取消金鑰刪除。

解密錯誤

例外狀況：KMSInvalidStateException

使用外部金鑰存放區中 KMS 金鑰的解密操作失敗時，AWS KMS 會傳回標準的 KMSInvalidStateException，密碼編譯操作將該例外狀況用於外部金鑰存放區上的所有外部組態錯誤和連接狀態錯誤。指出問題的錯誤訊息。

若要解密使用雙重加密來加密的密文，外部金鑰管理器會首先使用外部金鑰來解密密文外層。然後，AWS KMS 使用 AWS KMS KMS 金鑰中的金鑰材料來解密密文的內層。外部金鑰管理器或 AWS KMS 可拒絕無效或損毀的密文。

解密失敗時，KMSInvalidStateException 會伴隨下列錯誤訊息。它指出請求中的密文或選用的加密內容有問題。

外部金鑰存放區代理拒絕請求，因為指定的密文或其他已驗證的資料已損毀、遺失或無效。

- 當外部金鑰存放區代理或外部金鑰管理器報告密文或其加密內容無效時，通常表示傳送至 AWS KMS 的 Decrypt 請求中的密文或加密內容有問題。對於 Decrypt 操作，AWS KMS 會向代理傳送其在 Decrypt 請求中接收的相同密文和加密內容。

  此錯誤可能是由傳輸過程中的網路問題所引起，例如翻轉位。重試 Decrypt 請求。如果問題仍然存在，請確認密文沒有被更改或損壞。此外，請確認針對 AWS KMS 的 Decrypt 請求中的加密內容是否符合請求中加密資料的加密內容。

外部金鑰存放區代理提交用於解密的密文或加密內容已損毀、遺失或無效。

- 當 AWS KMS 拒絕從代理中接收到的密文時，表示外部金鑰管理器或代理向 AWS KMS 傳回無效或損毀的密文。

  此錯誤可能是由傳輸過程中的網路問題所引起，例如翻轉位。重試 Decrypt 請求。如果問題仍然存在，請確認外部金鑰管理器是否正常運作，而且外部金鑰存放區代理在將從外部金鑰管理器接收的密文傳回到 AWS KMS 之前不會對其進行變更。

外部金鑰錯誤

外部金鑰是外部金鑰管理器中的密碼編譯金鑰，可作為 KMS 金鑰的外部金鑰材料。AWS KMS 無法直接存取外部金鑰。它必須要求外部金鑰管理器 (透過外部金鑰存放區代理) 使用外部金鑰來加密資料或解密密文。

在外部金鑰存放區中建立 KMS 金鑰時，您可以在其外部金鑰管理器中指定外部金鑰的 ID。您無法在建立 KMS 金鑰之後變更外部金鑰 ID。為了避免 KMS 金鑰發生問題，CreateKey 操作會要求外部金鑰存放區代理驗證外部金鑰的 ID 和組態。如果外部金鑰不滿足與 KMS 金鑰搭配使用的要求，則 CreateKey 操作會失敗，並顯示可識別問題的例外狀況和錯誤訊息。

不過，建立 KMS 金鑰之後可能會發生問題。如果密碼編譯操作由於外部金鑰問題而失敗，則操作會失敗並傳回 KMSInvalidStateException 和指出問題的錯誤訊息。

CreateKey 外部鍵錯誤

例外狀
況：XksKeyAlreadyInUseException、XksKeyNotFoundException、XksKeyInvalidConfigurati

此CreateKey作業會嘗試驗證您在外部金鑰 ID (主控台) 或 XksKeyId (API) 參數中提供之外部金鑰的 ID 和屬性。此做法的設計目的在於在您嘗試搭配使用外部金鑰和 KMS 金鑰之前，提前偵測錯誤。

使用中的外部金鑰

外部金鑰存放區中的每個 KMS 金鑰都必須使用不同的外部金鑰。當CreateKey識別出 KMS 金鑰的外部金鑰 ID (XksKeyId) 在外部金鑰存放區中不是唯一的時候，就會失敗並顯示XksKeyAlreadyInUseException.

如果您為相同外部金鑰使用多個 ID，CreateKey 將無法識別重複內容。但是，具有相同外部金鑰的 KMS 金鑰無法互通，因為它們具有不同的 AWS KMS 金鑰材料和中繼資料。

找不到外部金鑰

當外部金鑰存放區 Proxy 報告找不到使用 KMS 金鑰的外部金鑰 ID (XksKeyId) 的外部金鑰時，CreateKey作業會失敗，並傳回XksKeyNotFoundException下列錯誤訊息。

```
外部金鑰存放區代理拒絕了請求，因為它找不到外部金鑰。
```

此錯誤可能發生的原因如下：

- KMS 金鑰的外部金鑰 ID (XksKeyId) 可能無效。若要查找外部金鑰代理用來識別外部金鑰的 ID，請參閱外部金鑰存放區代理或外部金鑰管理器文件。
- 可能已從外部金鑰管理器刪除外部金鑰。若要進行調查，請使用外部金鑰管理器工具。如果外部金鑰已永久刪除，請搭配 KMS 金鑰使用其他外部金鑰。如需外部金鑰的清單或要求，請參閱 外部金鑰存放區中 KMS 金鑰的要求。

未滿足外部金鑰要求

當外部金鑰存放區代理報告外部金鑰不符合與 KMS 金鑰搭配使用的要求時，CreateKey 操作會失敗，並傳回 XksKeyInvalidConfigurationException 及下列其中一個錯誤訊息。

```
外部金鑰的金鑰規格必須為 AES_256。指定外部金鑰的金鑰規格為 <key-spec> 。
```

- 外部金鑰必須是 256 位元對稱加密金鑰，金鑰規格為 AES_256。如果指定的外部金鑰是不同的類型，請指定符合此要求的外部金鑰 ID。

外部金鑰的狀態必須為 ENABLED。指定外部金鑰的狀態為 *<status>*。

- 必須在外部金鑰管理器中啟用外部金鑰。如果指定的外部金鑰未啟用，請使用外部金鑰管理器工具來啟用它，或指定已啟用的外部金鑰。

外部金鑰的金鑰用途必須包括 ENCRYPT 和 DECRYPT。指定外部金鑰的金鑰用途是 **<***key-usage* **>**。

- 外部金鑰必須設定為在外部金鑰管理器中進行加密和解密。如果指定的外部金鑰不包含這些操作，請使用外部金鑰管理器工具變更操作，或指定其他外部金鑰。

外部金鑰的密碼編譯操作錯誤

例外狀況：`KMSInvalidStateException`

當外部金鑰存放區代理找不到與 KMS 金鑰相關聯的外部金鑰時，或外部金鑰不滿足與 KMS 金鑰搭配使用的要求時，密碼編譯操作會失敗。

與建立 KMS 金鑰之前偵測到的外部金鑰問題相比，在密碼編譯操作期間偵測到的外部金鑰問題更難解決。您無法在建立 KMS 金鑰之後變更外部金鑰 ID。如果 KMS 金鑰尚未加密任何資料，您可以刪除 KMS 金鑰，然後使用不同的外部金鑰 ID 建立新金鑰。但是，使用 KMS 金鑰產生的密文無法由任何其他 KMS 金鑰解密，即使是具有相同外部金鑰的金鑰，因為金鑰會有不同的金鑰中繼資料和不同的 AWS KMS 金鑰材料。相反，盡可能使用外部金鑰管理器工具來解決外部金鑰問題。

當外部金鑰存放區代理報告外部金鑰問題時，密碼編譯操作會傳回 `KMSInvalidStateException` 以及可識別問題的錯誤訊息。

找不到外部金鑰

當外部金鑰存放區 Proxy 回報無法使用 KMS 金鑰的外部金鑰識別碼 (XksKeyId) 找到外部金鑰時，密碼編譯作業會傳回`KMSInvalidStateException`含下列錯誤訊息的錯誤訊息。

外部金鑰存放區代理拒絕了請求，因為它找不到外部金鑰。

此錯誤可能發生的原因如下：

- KMS 金鑰的外部金鑰 ID (`XksKeyId`) 已不再有效。

  若要查找與 KMS 金鑰相關聯的外部金鑰 ID，請檢視 KMS 金鑰的詳細資訊。若要查找外部金鑰代理用來識別外部金鑰的 ID，請參閱外部金鑰存放區代理或外部金鑰管理器文件。

  AWS KMS 在外部金鑰存放區中建立 KMS 金鑰時，會驗證外部金鑰 ID。不過，ID 可能會變得無效，特別是如果外部金鑰 ID 值是別名或可變名稱。您不能變更與現有 KMS 金鑰相關聯的外部金鑰 ID。若要解密使用 KMS 金鑰加密的任何密文，您必須將外部金鑰與現有的外部金鑰 ID 重新關聯。

  如果您尚未使用 KMS 金鑰加密資料，您可以使用有效的外部金鑰 ID 建立新的 KMS 金鑰。但是，如果您已使用 KMS 金鑰產生密文，即使使用相同的外部金鑰，也無法使用任何其他 KMS 金鑰來解密密文。
- 可能已從外部金鑰管理器刪除外部金鑰。若要進行調查，請使用外部金鑰管理器工具。如果可能，請嘗試從外部金鑰管理器的複本或備份中復原金鑰材料。如果永久刪除外部金鑰，則使用關聯的 KMS 金鑰加密的任何密文都無法復原。

外部金鑰組態錯誤

當外部金鑰存放區代理報告外部金鑰不符合與 KMS 金鑰搭配使用的要求時，密碼編譯操作會傳回 `KMSInvalidStateException` 及下列其中一個錯誤訊息。

外部金鑰存放區代理拒絕了請求，因為外部金鑰不支持請求的操作。

- 外部金鑰必須同時支援加密和解密。如果金鑰用途不包含加密和解密，請使用外部金鑰管理器工具來變更金鑰用途。

外部金鑰存放區代理拒絕了請求，因為外部金鑰管理器中未啟用外部金鑰。

- 外部金鑰必須已啟用並且可用於外部金鑰管理器。如果外部金鑰的狀態不是 `Enabled`，請使用外部金鑰管理器工具來啟用它。

代理問題

例外狀況：

CustomKeyStoreInvalidStateException (CreateKey)、KMSInvalidStateException (密碼編譯操
作)、UnsupportedOperationException、XksProxyUriUnreachableException、XksProxyInval
(CreateKey 以外的管理操作))

外部金鑰存放區代理會協調 AWS KMS 與外部金鑰管理器之間的所有通訊。它可將一般 AWS KMS 請求轉換為外部金鑰管理器能夠理解的格式。如果外部金鑰存放區代理不符合 AWS KMS 外部金鑰存放區代理 API 規格，或者如果無法正常運作，或無法與 AWS KMS 通訊，您將無法在外部金鑰存放區中建立或使用 KMS 金鑰。

雖然許多錯誤提到外部金鑰存放區代理，因為它在外部金鑰存放區架構中起著關鍵作用，但這些問題可能源於外部金鑰管理器或外部金鑰。

本節中的問題與外部金鑰存放區代理的設計或操作問題有關。解決這些問題可能需要變更代理軟體。請諮詢您的代理管理員。為了協助診斷代理問題，AWS KMS 會提供 XKS Proxy Text Client，它是一個開放原始碼測試用戶端，可驗證您的外部金鑰存放區代理是否符合 AWS KMS 外部金鑰存放區代理 API 規格。

CustomKeyStoreInvalidStateException 、KMSInvalidStateException 或 XksProxyUriUnreachableException
外部金鑰存放區代理處於運作不佳狀態。如果您重複看到此訊息，請通知外部金鑰存放區代理管理員。

- 此錯誤表示外部金鑰存放區代理中存在操作問題或軟體錯誤。您可以找到產生每個錯誤之 AWS KMS API 作業的 CloudTrail 記錄項目。重試操作可能會解決此錯誤。但是，如果問題仍然存在，請通知您的外部金鑰存放區代理管理員。
- 當外部金鑰存放區 Proxy 報告 (在GetHealthStatus回應中) 所有外部金鑰管理員執行個體為時UNAVAILABLE，嘗試建立或更新外部金鑰存放區失敗，並出現此例外狀況。如果此錯誤仍然存在，請參閱外部金鑰管理器文件。

CustomKeyStoreInvalidStateException 、KMSInvalidStateException 或 XksProxyInvalidResponseException

AWS KMS 無法解譯來自外部金鑰存放區代理的回應。如果您重複看到此錯誤，請咨詢您的外部金鑰存放區代理管理員。

- 當代理傳回 AWS KMS 無法解析或解釋的未定義回應時，AWS KMS 操作會產生此例外狀況。由於暫時的外部問題或偶發的網絡錯誤，偶爾會發生此錯誤。但是，如果它仍然存在，則可能表示外部金鑰存放區代理不符合 AWS KMS 外部金鑰存放區代理 API 規格。通知您的外部金鑰存放區管理員或廠商。

`CustomKeyStoreInvalidStateException` 、`KMSInvalidStateException` 或 `UnsupportedOperationException`

外部金鑰存放區代理拒絕了請求，因為其不支援請求的密碼編譯操作。

- 外部金鑰存放區代理應支援 AWS KMS 外部金鑰存放區代理 API 規格中定義的所有代理 API。此錯誤表示代理不支援與請求相關的操作。通知您的外部金鑰存放區管理員或廠商。

代理授權問題

例外狀況：`CustomKeyStoreInvalidStateException`、`KMSInvalidStateException`

某些外部金鑰存放區代理會實作使用其外部金鑰的授權需求。允許外部金鑰存放區代理 (但不是必需的) 來設計和實作授權方案，該方案允許特定使用者在特定條件下請求特定操作。例如，代理可能允許使用者使用特定的外部金鑰進行加密，但無法使用它進行解密。如需詳細資訊，請參閱 外部金鑰存放區代理授權 (選用)。

代理授權是基於中繼資料，AWS KMS 將其包含在其對代理的請求中。只有當請求來自 VPC 端點且呼叫者與 KMS 金鑰位於相同帳戶時，`awsSourceVpc` 和 `awsSourceVpce` 欄位才會包含在中繼資料中。

```
"requestMetadata": {
    "awsPrincipalArn": string,
    "awsSourceVpc": string, // optional
    "awsSourceVpce": string, // optional
    "kmsKeyArn": string,
    "kmsOperation": string,
    "kmsRequestId": string,
    "kmsViaService": string // optional
```

```
}
```

當代理因授權失敗而拒絕請求時，相關 AWS KMS 操作會失敗。`CreateKey` 會傳回 `CustomKeyStoreInvalidStateException`。AWS KMS 密碼編譯操作會傳回 `KMSInvalidStateException`。兩者都使用下列錯誤訊息：

> 外部金鑰存放區代理拒絕存取該操作。確認使用者和外部金鑰都已獲得針對此操作的授權，然後再試一次請求。

- 若要解決錯誤，請使用外部金鑰管理器或外部金鑰存放區代理工具來確定授權失敗的原因。然後，更新導致未授權請求的程序，或使用外部金鑰存放區代理工具來更新授權政策。您無法在 AWS KMS 中解決此錯誤。

金鑰類型參考

AWS KMS 可支援不同類型 KMS 金鑰的不同功能。例如，您只能使用對稱加密 KMS 金鑰產生對稱資料金鑰和非對稱資料金鑰對。此外，僅對稱加密 KMS 金鑰支援金鑰資料匯入和金鑰自動輪換，而且只能在自訂金鑰存放區中建立對稱加密 KMS 金鑰。

此參考包括兩個資料表。

- 金鑰類型資料表列出對稱加密 KMS 金鑰、非對稱 KMS 金鑰和 HMAC KMS 金鑰的有效 AWS KMS 操作。
- 特殊功能資料表列出多區域 KMS 金鑰、含有匯入金鑰材料的 KMS 金鑰以及自訂金鑰存放區中的 KMS 金鑰的有效 AWS KMS 操作。

金鑰類型資料表

您可能需要水平或垂直捲動，才能查看此資料表中的所有資料。

AWS KMSAPI 操作	對稱加密 KMS 金鑰	HMAC KMS 金鑰	非對稱 KMS 金鑰 (ENCRYPT_DECRYPT)	非對稱 KMS 金鑰 (SIGN_VERIFY)
CancelKeyDeletion	✓	✓	✓	✓
CreateAlias	✓	✓	✓	✓
CreateGrant	✓	✓	✓	✓
CreateKey	✓	✓	✓	✓
解密	✓	⊗	✓	⊗
DeleteAlias	✓	✓	✓	✓
DeleteImportedKeyMaterial 僅對含有匯入金鑰材料的 KMS 金鑰有效 (`Origin` 為 `EXTERNAL`)。	✓	✓	✓	✓
DescribeKey	✓	✓	✓	✓
DisableKey	✓	✓	✓	✓
DisableKeyRotation	✓ 僅對含有 AWS KMS 金鑰材料的 KMS	⊗	⊗	⊗

AWS KMSAPI 操作	對稱加密 KMS 金鑰	HMAC KMS 金鑰	非對稱 KMS 金鑰 (ENCRYPT_DECRYPT)	非對稱 KMS 金鑰 (SIGN_VERIFY)
	金鑰有效 (`Origin` 為 `AWS_KMS`)。			
EnableKey	✓	✓	✓	✓
EnableKeyRotation	✓ 僅對含有 AWS KMS 金鑰材料的 KMS 金鑰有效 (`Origin` 為 `AWS_KMS`)。	⊗	⊗	⊗
加密	✓	⊗	✓	⊗
GenerateDataKey	✓	⊗	⊗	⊗
GenerateDataKeyPair 產生受對稱加密 KMS 金鑰保護的非對稱資料金鑰對。	✓ 對於自訂金鑰存放區中的 KMS 金鑰無效。	⊗	⊗	⊗

AWS KMSAPI 操作	對稱加密 KMS 金鑰	HMAC KMS 金鑰	非對稱 KMS 金鑰 (ENCRYPT_DECRYPT)	非對稱 KMS 金鑰 (SIGN_VERIFY)
GenerateDataKeyPairWithoutPlaintext 產生受對稱加密 KMS 金鑰保護的非對稱資料金鑰對。	✓ 對於自訂金鑰存放區中的 KMS 金鑰無效。	⊗	⊗	⊗
GenerateDataKeyWithoutPlaintext	✓	⊗	⊗	⊗
GenerateMac	⊗	✓	⊗	⊗
GetKeyPolicy	✓	✓	✓	✓
GetKeyRotationStatus	✓	✓ (KeyRotationEnabled 一律為 false.)	✓ (KeyRotationEnabled 一律為 false.)	✓ (KeyRotationEnabled 一律為 false.)
GetParametersForImport 僅對含有匯入金鑰材料的 KMS 金鑰有效 (Origin 為 EXTERNAL)。	✓	✓	✓	✓
GetPublicKey	⊗	⊗	✓	✓

AWS KMSAPI 操作	對稱加密 KMS 金鑰	HMAC KMS 金鑰	非對稱 KMS 金鑰 (ENCRYPT_DECRYPT)	非對稱 KMS 金鑰 (SIGN_VERIFY)
ImportKeyMaterial 僅對含有匯入金鑰材料的 KMS 金鑰有效 (`Origin` 為 `EXTERNAL`)。	✓	✓	✓	✓
ListAliases	✓	✓	✓	✓
ListGrants	✓	✓	✓	✓
ListKeyPolicies	✓	✓	✓	✓
ListResourceTags	✓	✓	✓	✓
ListRetirableGrants	✓	✓	✓	✓
PutKeyPolicy	✓	✓	✓	✓
ReEncrypt	✓	⊗	✓	⊗
ReplicateKey - 僅適用於多區域金鑰	✓	✓	✓	✓
RetireGrant	✓	✓	✓	✓
RevokeGrant	✓	✓	✓	✓
ScheduleKeyDeletion	✓	✓	✓	✓

AWS KMSAPI 操作	對稱加密 KMS 金鑰	HMAC KMS 金鑰	非對稱 KMS 金鑰 (ENCRYPT_DECRYPT)	非對稱 KMS 金鑰 (SIGN_VERIFY)
符號	⊗	⊗	⊗	✓
TagResource	✓	✓	✓	✓
UntagResource	✓	✓	✓	✓
UpdateAlias 目前的 KMS 金鑰和新的 KMS 金鑰必須是相同類型 (兩者皆為對稱或皆為非對稱或皆為 HMAC)，且必須具有相同的金鑰用途。	✓	✓	✓	✓
UpdateKeyDescription	✓	✓	✓	✓
UpdateReplicaRegion - 僅適用於多區域金鑰	✓	✓	✓	✓
確認	⊗	⊗	⊗	✓
VerifyMac	⊗	✓	⊗	⊗

特殊功能資料表

此資料表顯示每種特殊用途金鑰類型支援的 AWS KMS API 操作。

閱讀此資料表時，請注意下列互動：

- 多區域金鑰：

  - 多區域金鑰可以是對稱加密 KMS 金鑰、非對稱 KMS 金鑰、HMAC KMS 金鑰、含有匯入金鑰材料的 KMS 金鑰。
  - 您無法在自訂金鑰存放區建立多區域金鑰。
- 匯入的金鑰資料
  - 您可針對對稱加密 KMS 金鑰、非對稱 KMS 金鑰，以及 HMAC KMS 金鑰匯入金鑰資料。
  - 您可利用匯入金鑰資料來建立多區域金鑰。
  - 您無法在自訂金鑰存放區中建立含有匯入金鑰材料的金鑰。
  - 具有匯入金鑰資料的 KMS 金鑰不支援自動金鑰輪換 (`EnableKeyRotation`、`DisableKeyRotation`)。
- 自訂金鑰存放區
  - 自訂金鑰存放區僅支援對稱加密 KMS 金鑰。
  - 自訂金鑰存放區中的 KMS 金鑰不支援非對稱金鑰對 (`GenerateDataKeyPair`、`GenerateDataKeyPairWithoutPlaintext`) 的對稱操作。
  - 自訂金鑰存放區中的 KMS 金鑰不支援自動金鑰輪換 (`EnableKeyRotation`、`DisableKeyRotation`)。
  - 您無法在自訂金鑰存放區建立多區域金鑰。

您可能需要水平或垂直捲動，才能查看此資料表中的所有資料。

AWS KMSAPI 操作	多區域金鑰	匯入的金鑰資料	在自訂金鑰存放區中的 KMS 金鑰
CancelKeyDeletion	✓	✓	✓
CreateAlias	✓	✓	✓
CreateGrant	✓	✓	✓
CreateKey 您可以使用 `CreateKey` 來建立多區域主要金鑰、含有匯入金鑰材料的	✓	✓	✓

AWS KMSAPI 操作	多區域金鑰	匯入的金鑰資料	在自訂金鑰存放區中的 KMS 金鑰
KMS 金鑰，或者自訂金鑰存放區中的 KMS 金鑰。若要建立多區域複本金鑰，請使用 ReplicateKey 。			
解密	✓ 只有在 KeyUsage 為 ENCRYPT_DECRYPT 時有效	✓	✓
DeleteAlias	✓	✓	✓
DeleteImportedKeyMaterial	✓ 僅對含有匯入金鑰材料的金鑰有效 (Origin 為 EXTERNAL)	✓	⊗
DescribeKey	✓	✓	✓
DisableKey	✓	✓	✓
DisableKeyRotation	✓ 僅對含有 AWS KMS 金鑰材料的對稱加密金鑰有效 (Origin 為 AWS_KMS)。	⊗	⊗

AWS KMSAPI 操作	多區域金鑰	匯入的金鑰資料	在自訂金鑰存放區中的 KMS 金鑰
EnableKey	✓ 僅對對稱加密 KMS 金鑰有效	✓	✓
EnableKeyRotation	✓ 僅對含有 AWS KMS 金鑰材料的對稱加密金鑰有效 (`Origin` 為 `AWS_KMS`)。	⊗	⊗
加密	✓ 只有在 `KeyUsage` 為 `ENCRYPT_DECRYPT` 時有效	✓	✓
GenerateDataKey	✓ 僅對對稱加密 KMS 金鑰有效	✓	✓
GenerateDataKeyPair	✓ 僅對對稱加密 KMS 金鑰有效	✓	⊗

AWS KMSAPI 操作	多區域金鑰	匯入的金鑰資料	在自訂金鑰存放區中的 KMS 金鑰
GenerateDataKeyPairWithoutPlaintext	✓ 僅對對稱加密 KMS 金鑰有效	✓	⊗
GenerateDataKeyWithoutPlaintext	✓ 僅對對稱加密 KMS 金鑰有效	✓	✓
GenerateMac 僅對 HMAC KMS 金鑰有效	✓	✓	⊗
GetKeyPolicy	✓	✓	✓
GetKeyRotationStatus	✓	✓ (KeyRotationEnabled 一律為 false.)	⊗
GetParametersForImport	✓ 僅對含有匯入金鑰材料的金鑰有效 (Origin 為 EXTERNAL)。	✓	⊗
GetPublicKey 僅對非對稱 KMS 金鑰有效。	✓	✓	⊗

AWS KMSAPI 操作	多區域金鑰	匯入的金鑰資料	在自訂金鑰存放區中的 KMS 金鑰
ImportKeyMaterial	✓ 僅對含有匯入金鑰材料的金鑰有效 (`Origin` 為 `EXTERNAL`)。	✓	⊗
ListAliases	✓	✓	✓
ListGrants	✓	✓	✓
ListKeyPolicies	✓	✓	✓
ListResourceTags	✓	✓	✓
ListRetirableGrants	✓	✓	✓
PutKeyPolicy	✓	✓	✓
ReEncrypt	✓ 只有在 `KeyUsage` 為 `ENCRYPT_DECRYPT` 時有效	✓	✓
ReplicateKey	✓ 僅對多區域主金鑰有效。	✓ 僅對多區域主金鑰有效。	⊗

AWS KMSAPI 操作	多區域金鑰	匯入的金鑰資料	在自訂金鑰存放區中的 KMS 金鑰
RetireGrant	✓	✓	✓
RevokeGrant	✓	✓	✓
ScheduleKeyDeletion	✓	✓	✓
符號 只有在 KeyUsage 為 SIGN_VERIFY 時有效。	✓	✓	⊗
TagResource	✓	✓	✓
UntagResource	✓	✓	✓
UpdateAlias - 目前的 KMS 金鑰和新的 KMS 金鑰必須是相同類型 (兩者皆為對稱或皆為非對稱或皆為 HMAC)，且必須具有相同的金鑰用途。	✓	✓	✓
UpdateKeyDescription	✓	✓	✓
UpdateReplicaRegion	✓	✓ 僅對多區域金鑰有效。	⊗

AWS KMSAPI 操作	多區域金鑰	匯入的金鑰資料	在自訂金鑰存放區中的 KMS 金鑰
確認 只有在 KeyUsage 為 SIGN_VERIFY 時有效。	✓	✓	⊗
VerifyMac 僅對 HMAC KMS 金鑰有效	✓	✓	⊗

AWS Key Management Service 的安全

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。

安全是 AWS 與您共同肩負的責任。共同責任模型將其描述為雲端的安全性和雲端中的安全性：

- 雲端本身的安全 – AWS 負責保護在 AWS Cloud 中執行 AWS 服務的基礎設施。AWS 也提供您可安全使用的服務。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。若要了解適用於 AWS Key Management Service (AWS KMS) 的合規計劃，請參閱合規計劃的 AWS 服務範圍。
- 雲端內部的安全 – 您的責任取決於所使用的 AWS 服務。在 AWS KMS 中，除了 AWS KMS keys 的組態和使用之外，您也要對其他因素負責，包括資料的敏感性、您公司的要求和適用法律和法規。

本文件有助於您了解如何在使用 AWS Key Management Service 時套用共同責任模型。它會示範如何設定 AWS KMS 以符合您的安全性和合規目標。

主題

AWS Key Management Service 中的資料保護

AWS Key Management Service 存放並保護您的加密金鑰，使其具有高可用性，同時為您提供強大且靈活的存取控制。

主題

- 網際網路流量隱私權

保護金鑰資料

根據預設，AWS KMS 會產生並保護 KMS 金鑰的密碼編譯金鑰資料。此外，AWS KMS 還提供在 AWS KMS 外部建立並受保護的金鑰資料選項。如需有關 KMS 金鑰與金鑰資料的技術詳細資訊，請參閱《AWS Key Management Service 密碼編譯詳細資訊》。

保護在 AWS KMS 產生的金鑰資料

當您建立 KMS 金鑰時，AWS KMS 預設會為該 KMS 金鑰產生並保護密碼編譯材料。

為保護 KMS 金鑰的金鑰資料，AWS KMS 依賴 FIPS 140-2 安全層級 3 驗證的硬體安全模組 (HSM) 分散式機群。每個 AWS KMS HSM 都是獨立的專用硬體設備，旨在提供專用的密碼編譯功能，以便滿足 AWS KMS 的安全性及可擴可性需求。(AWS KMS 在中國區域採用的 HSM 由 OSCCA 認證，並符合所有相關中國法規，但並未根據 FIPS 140-2 密碼編譯模組驗證計畫進行驗證。)

根據預設，KMS 金鑰的金鑰資料會在 HSM 產生時加密。金鑰資料僅在 HSM 揮發性記憶體內進行解密，且解密時間僅限密碼編譯操作運用金鑰資料所需的數毫秒。每當金鑰資料未處於作用中使用狀態時，就會在 HSM 加密，並傳輸至高持性 (99.999999999%) 且低延遲持續性儲存體，其會在此儲存體保持獨立並與 HSM 隔離。純文字金鑰資料永遠不會離開 HSM 安全界限；永遠不會寫入磁碟或保存於任何儲存媒體。(唯一例外是非對稱金鑰對的公有金鑰，其非秘密。)

AWS 宣告作為基本安全政策，在任何 AWS 服務 的任何類型純文字密碼編譯金鑰資料均未與人類互動。包括 AWS 服務 操作員在內，任何人均無任何機制可檢視、存取或匯出純文字金鑰資料。即使在災難性故障及災難復原事件，此政策也適用。AWS KMS 的純文字客戶金鑰資料僅在回應客戶或其委派對服務的授權要求時，才會在 AWS KMS FIPS 驗證的 HSM 用於密碼編譯操作。

對於客戶自管金鑰，建立金鑰的 AWS 帳戶 是金鑰的唯一且不可轉讓擁有者。擁有帳戶對控制金鑰存取權的授權政策具完整且獨佔控制權。對於AWS 受管金鑰，AWS 帳戶 可完全控制 IAM 政策，並授權向 AWS 服務 提出請求。

保護 AWS KMS 外部產生的金鑰資料

AWS KMS 提供在 AWS KMS 產生的金鑰資料替代方案。

自訂金鑰存放區為選用 AWS KMS 功能，可讓您建立 KMS 金鑰，該金鑰由 AWS KMS 外部產生及使用的金鑰資料提供支援。在 AWS CloudHSM 金鑰存放區的 KMS 金鑰由 AWS CloudHSM 硬體安全模組 (由您控制) 的金鑰支援。這些 HSM 取得 FIPS 140-2 層級 3 認證。外部金鑰存放區的 KMS 金鑰由外部金鑰管理器 (由您在 AWS 外部控制及管理) 的金鑰支援，例如私有資料中心的實體 HSM。

另一個選用功能可讓您為 KMS 金鑰匯入金鑰材料。為在傳輸至 AWS KMS 時保護匯入的金鑰資料，您可利用 AWS KMS HSM 所產生 RSA 金鑰對的公有金鑰來加密金鑰資料。匯入的金鑰資料會在 AWS KMS HSM 解密，並在 HSM 採用對稱金鑰重新加密。如同所有 AWS KMS 金鑰資料，純文字匯入金鑰資料永遠會對 HSM 進行加密。然而，提供金鑰資料的客戶需負責安全使用、持久性及 AWS KMS 外部的金鑰資料維護。

資料加密

AWS KMS 中的資料包括 AWS KMS keys 及其所代表的加密金鑰材料。此金鑰材料僅以純文字形式存在於 AWS KMS 硬體安全模組 (HSM)，且僅在使用時存在。否則，金鑰材料會加密並存放在耐久的持久性儲存裝置中。

AWS KMS 為 KMS 金鑰產生的金鑰材料永遠會對 AWS KMS HSM 邊界加密。它不會在任何 AWS KMS API 操作中匯出或傳輸。但多區域金鑰是例外：AWS KMS 會使用跨區域複寫機制，將多區域金鑰的金鑰材料從一個 AWS 區域 的 HSM 複製到另一個 AWS 區域 的 HSM。如需詳細資訊，請參閱《AWS Key Management Service 密碼編譯詳細資訊》中的多區域金鑰的複寫程序。

主題

- 靜態加密
- 傳輸中加密

靜態加密

AWS KMS 會在符合 FIPS 140-2 安全層級 3 之硬體安全模組 (HSM) 為 AWS KMS keys 產生金鑰資料。唯一的例外是中國區域，在該處 AWS KMS 用來產生 KMS 金鑰的 HSM 符合所有相關的中國法規，但並未根據 FIPS 140-2 密碼模組驗證計畫進行驗證。不使用時，金鑰資料會由 HSM 金鑰加密，並寫入耐久的持久性儲存裝置。KMS 金鑰的金鑰材料和保護金鑰材料的加密金鑰永遠不會讓 HSM 以純文字形式出現。

KMS 金鑰的金鑰材料加密和管理完全由 AWS KMS 處理。

如需詳細資訊，請參閱 AWS Key Management Service 密碼編譯詳細資訊中的使用 AWS KMS keys。

傳輸中加密

AWS KMS 為 KMS 金鑰產生的金鑰材料永遠不會在 AWS KMS API 操作中匯出或傳輸。AWS KMS 使用金鑰識別符來代表 API 操作中的 KMS 金鑰。同樣地，AWS KMS 自訂金鑰存放區中 KMS 金鑰的金鑰材料是不可匯出的，永遠不會在 AWS KMS 或 AWS CloudHSM API 操作中傳輸。

然而，一些 AWS KMS API 操作會傳回資料金鑰。此外，客戶可以使用 API 操作為所選的 KMS 金鑰匯入金鑰材料。

所有 AWS KMS API 呼叫必須經過簽署，並利用 Transport Layer Security (TLS) 進行傳輸。AWS KMS 需要 TLS 1.2，並建議在所有區域採用 TLS 1.3。AWS KMS 也支援混合式後量子 TLS，適用所有區域 (中國區域除外) 的 AWS KMS 服務端點。AWS KMS 不支援 AWS GovCloud (US) FIPS 端點的混合式後量子 TLS。呼叫 AWS KMS 還需要支援完整轉寄密碼的現代加密套件，這表示任何機密 (例如私有金鑰) 的洩露也不會影響工作階段金鑰。

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。若要採用標準 AWS KMS 端點或 AWS KMS FIPS 端點，用戶端必須支援 TLS 1.2 或更新版本。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。如需 AWS KMS FIPS 端點清單，請參閱《AWS 一般參考》的 AWS Key Management Service 端點與配額。

AWS KMS 服務主機和 HSM 之間的通訊會使用經過驗證之加密配置中的橢圓曲線密碼編譯 (ECC) 和進階加密標準 (AES) 進行保護。如需詳細資訊，請參閱 AWS Key Management Service 密碼編譯詳細資訊中的內部通訊安全。

網際網路流量隱私權

AWS KMS 支援 AWS Management Console 和一組 API 操作，可讓您建立和管理 AWS KMS keys 並在密碼編譯操作中使用它們。

AWS KMS 支援兩種從私有網路至 AWS 的網路連線選項。

- 網際網路上的 IPSec VPN 連接
- AWS Direct Connect 透過標準乙太網路光纖纜線將您的內部網路連結至 AWS Direct Connect 位置。

所有 AWS KMS API 呼叫必須經過簽署，並使用 Transport Layer Security (TLS) 進行傳輸。這些呼叫還需要支援完整轉寄密碼的現代加密套件。存放 KMS 金鑰之金鑰材料的硬體安全模組 (HSM) 流量只能從已知的 AWS KMS API 主機透過 AWS 內部網路傳輸。

若要直接從 Virtual Private Cloud (VPC) 連線至 AWS KMS，而不透過公有網際網路傳送流量，請使用由 AWS PrivateLink 提供技術的 VPC 端點。如需詳細資訊，請參閱 透過 VPC 端點連線至 AWS KMS。

AWS KMS 也支援適用於 Transport Layer Security (TLS) 網路加密通訊協定的混合式後量子金鑰交換選項。連線到 AWS KMS API 端點時，您可將此選項與 TLS 搭配使用。

適用於 AWS Key Management Service 的 Identity and Access Management

AWS Identity and Access Management (IAM) 可協助您安全地控管對 AWS 資源的存取。管理員可以控制對誰進行身分驗證 (已登入) 和授權 (具有許可)，以使用 AWS KMS 資源。如需詳細資訊，請參閱將 IAM 政策與 AWS KMS 搭配使用。

金鑰政策是控制對 AWS KMS 中 KMS 金鑰之存取的主要機制。每個 KMS 金鑰都必須有一個金鑰政策。您也可以使用 IAM 政策和授予，以及金鑰政策，以控制對 KMS 金鑰的存取。如需詳細資訊，請參閱 AWS KMS 的身分驗證與存取控制。

如果正在使用 Amazon Virtual Private Cloud (Amazon VPC)，則您可以建立介面 VPC 端點至採用 AWS PrivateLink 技術的 AWS KMS。您也可以使用 VPC 端點政策來判斷哪些委託人可以存取 AWS KMS 端點，他們可以進行哪些 API 呼叫，以及他們可以存取哪些 KMS 金鑰。如需詳細資訊，請參閱控制對 VPC 端點的存取。

AWS Key Management Service 中的日誌記錄和監控

監控主要用於了解 AWS KMS 中 AWS KMS keys 的可用性、狀態和使用情形。監控是維持 AWS 解決方案之安全性、可靠性、可用性和效能的重要環節。AWS 會提供數種工具，用於監控您的 KMS 金鑰。

AWS CloudTrail 日誌

系統會將每次對 AWS KMS API 操作的呼叫擷取為 AWS CloudTrail 日誌中的事件。這些日誌會記錄來自 AWS KMS 主控台的全部 API 呼叫，以及 AWS KMS 和其他 AWS 服務的呼叫。跨帳戶 API 呼叫 (例如在不同AWS 帳戶帳戶中使用 KMS 金鑰的呼叫) 會記錄在兩個帳戶的記 CloudTrail 錄中。

故障診斷或稽核時，您可以使用日誌來重建 KMS 金鑰的生命週期。您也可以在密碼編譯操作中檢視其 KMS 金鑰的管理和使用。如需詳細資訊，請參閱 the section called “使用 AWS CloudTrail 進行記錄”。

Amazon CloudWatch 日誌

監控、存放及存取來自 AWS CloudTrail 和其他來源的日誌檔案。如需詳細資訊，請參閱 Amazon CloudWatch 使用者指南。

對於AWS KMS， CloudWatch 儲存有用的資訊，以協助您避免 KMS 金鑰及其保護的資源發生問題。如需詳細資訊，請參閱 the section called “使用監控 CloudWatch”。

Amazon EventBridge

AWS KMS當 KMS 金鑰已輪替或刪除，或 KMS 金鑰中匯入的金鑰材料到期時，就會產生 EventBridge 事件。搜尋 AWS KMS 事件 (API 操作)，並將這些事件路由至一或多個目標函數或串流，以擷取狀態資訊。如需詳細資訊，請參閱the section called “使用 Amazon 監控 EventBridge”和 Amazon EventBridge 使用者指南。

Amazon CloudWatch 指標

您可以使用 CloudWatch 指標來監控 KMS 金鑰，這些指標會從原始資料收集並處理AWS KMS成效能指標。資料會以兩週的時間間隔記錄，以便您可以檢視目前和歷史記錄資訊的趨勢。這可協助您了解 KMS 金鑰的使用方式，及其使用方式隨著時間的變化。如需使用 CloudWatch 指標監控 KMS 金鑰的相關資訊，請參閱AWS KMS 指標與維度。

Amazon CloudWatch 警報

監看您指定期間的單一指標變更。然後，根據在數個期間與閾值相關的指標值，執行動作。例如，您可以建立 CloudWatch 警示，該警示會在有人嘗試使用排定在加密作業中刪除的 KMS 金鑰時觸發。這表示 KMS 金鑰仍在使用中，可能不應該刪除。如需詳細資訊，請參閱 the section called “建立警示”。

AWS Security Hub

您可利用 AWS Security Hub 來監控 AWS KMS 的使用方式是否符合安全業界標準及最佳實務合規。Security Hub 會透過安全控制來評估資源組態和安全標準，協助您遵守各種合規架構。如需詳細資訊，請參閱《AWS Security Hub使用者指南》的AWS Key Management Service控制。

AWS Key Management Service 的合規驗證

在多個 AWS 合規計劃中，第三方稽核人員會評估 AWS Key Management Service 的安全與合規。這些計劃包括 SOC、PCI、FedRAMP、HIPAA 等等。

主題

- 合規和安全文件
- 進一步了解

合規和安全文件

下列合規和安全文件涵蓋 AWS KMS。若要檢視，請使用 AWS Artifact。

- Cloud Computing Compliance Controls Catalogue (C5)
- ISO 27001:2013 適用性聲明 (SoA)
- ISO 27001:2013 認證
- ISO 27017:2015 適用性聲明 (SoA)
- ISO 27017:2015 認證
- ISO 27018:2015 適用性聲明 (SoA)
- ISO 27018:2014 認證
- ISO 9001:2015 認證
- PCI DSS 合規聲明文件 (AOC) 與責任摘要
- 服務組織控制 (SOC) 1 報告
- 服務組織控制 (SOC) 2 報告
- 服務組織控制 (SOC) 2 報告，針對機密性
- FedRAMP-High

如需使用 AWS Artifact 的說明，請參閱 AWS Artifact 中的下載報告。

進一步了解

您使用 AWS KMS 的合規責任，取決於資料的機密性、您公司的合規目標及適用法律和法規。如果使用 AWS KMS 必須遵循已發行標準，則 AWS 會提供資源予以協助：

- 合規計劃範圍內的 AWS 服務 – 此頁面列出了特定合規計劃範圍內的 AWS 服務。如需一般資訊，請參閱 AWS 合規計劃。
- 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基準環境的架構考量和步驟。
- 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 Config – 此 AWS 服務可評定資源組態與內部實務、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 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 Key Management Service 中的恢復能力

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，您所設計與操作的應用程式和資料庫，就能夠在可用區域之間自動容錯移轉，而不會發生中斷。可用區域的可用性、容錯能力和擴充能力，均較單一或多個資料中心的傳統基礎設施還高。

除了 AWS 全球基礎設施，AWS KMS 還提供數種功能，可協助支援資料的彈性和備份需求。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。

區域隔離

AWS Key Management Service (AWS KMS) 是一種可自我維持的區域服務，可用於所有 AWS 區域。AWS KMS 的區域隔離設計可確保一個 AWS 區域 中的可用性問題不會影響任何其他區域中的 AWS KMS 操作。AWS KMS 旨在確保零計劃停機，所有軟體更新和擴展操作都無縫且不知不覺地執行。

AWS KMS 服務水準協議 (SLA) 包括對所有 KMS API 的 99.999% 的服務承諾。為了履行此承諾，AWS KMS 確保執行 API 請求所需的所有資料和授權資訊在接收請求的所有區域主機上都可用。

AWS KMS 基礎設施會複寫在每個區域的至少三個可用區域 (AZ) 中。AWS KMS 旨在為一個區域中任何可用區域的客戶流量提供服務，以便確定多個主機失敗不會影響 AWS KMS 效能。

您對 KMS 金鑰的屬性或許可所做的變更將複製到區域中的所有主機，以確保區域中的任何主機都可以正確處理後續請求。使用您的 KMS 金鑰進行密碼編譯操作的請求會轉送至 AWS KMS 硬體安全模組 (HSM) 的機群，其中任何一個模組都可以使用 KMS 金鑰執行操作。

多租用戶設計

AWS KMS 的多租用戶設計使其能夠實現 99.999% 的可用性 SLA，並保持較高的請求率，同時保護金鑰和資料的機密性。

部署多個完整性強制執行機制，以確保您為密碼編譯操作指定的 KMS 金鑰始終是使用的金鑰。

您的 KMS 金鑰的純文字金鑰資料受到廣泛保護。金鑰資料一經建立就會在 HSM 中加密，並且加密的金鑰資料會立即移至安全、低延遲的儲存中。系統會在 HSM 內擷取並解密已加密的金鑰，以便及時使用。純文字金鑰僅在完成密碼編譯操作所需的時間內保留在 HSM 記憶體中。然後在 HSM 中對其進行重新加密，並將加密的金鑰傳回至儲存體。純文字金鑰資料永遠不會離開 HSM；它永遠不會寫入持久性儲存。

如需 AWS KMS 用於保護您的金鑰的機制的詳細資訊，請參閱 AWS Key Management Service 密碼編譯詳細資訊。

AWS KMS 中的復原功能最佳實務

若要對您的 AWS KMS 資源最佳化復原功能，請考慮以下策略。

- 若要支援備份和災難復原策略，請考慮多區域金鑰，它們是在一個 AWS 區域 中建立的 KMS 金鑰，並僅複寫至您指定的區域。使用多區域金鑰，您可以在 AWS 區域 (在同一分割區內) 間移動已加密資源，而不會暴露純文字，並在需要時在其任何目的地區域中解密資源。相關的多區域金鑰是可相互操作的，因為它們共用相同的金鑰資料和金鑰 ID，但它們具有獨立的金鑰策略來實現高解析度存取控制。如需詳細資訊，請參閱 AWS KMS 中的多區域金鑰。
- 若要在多租用戶服務 (如 AWS KMS) 中保護您的金鑰，請務必使用存取控制，包括金鑰政策和 IAM 政策。此外，您可以使用採用 AWS PrivateLink 技術的 VPC 界面端點將請求傳送至 AWS KMS。執行此操作時，Amazon VPC 與 AWS KMS 之間的所有通訊都會使用僅限於您的 VPC 的專用 AWS KMS 端點完全在 AWS 網路內執行。您可以透過使用 VPC 端點政策建立額外授權層來進一步保護這些請求。如需詳細資訊，請參閱透過 VPC 端點連接至 AWS KMS。

AWS Key Management Service 中的基礎設施安全

作為受管服務，AWS Key Management Service (AWS KMS) 受到 Amazon Web Services：安全程序概觀所述的 AWS 全球網路安全處理程序保護。

若要透過網路存取 AWS KMS，您可呼叫 AWS Key Management ServiceAPI 參考所述的 AWS KMS API 操作。AWS KMS 在所有區域要求 TLS 1.2 並建議採用 TLS 1.3。AWS KMS 也支援混合式後量子 TLS，適用所有區域 (中國區域除外) 的 AWS KMS 服務端點。AWS KMS 不支援 AWS GovCloud (US) FIPS 端點的混合式後量子 TLS。若要採用標準 AWS KMS 端點或 AWS KMS FIPS 端點，用戶端必須支援 TLS 1.2 或更新版本。用戶端也必須支援具備完美轉送私密 (PFS) 的密碼套件，例如臨時 Diffie-Hellman (DHE) 或橢圓曲線臨時 Diffie-Hellman (ECDHE)。現代系統 (如 Java 7 和更新版本) 大多會支援這些模式。

此外，請求必須使用存取金鑰 ID 和與 IAM 主體相關聯的私密存取金鑰來簽署。或者，您可以透過 AWS Security Token Service (AWS STS) 來產生暫時安全憑證來簽署請求。

您可以從任何網路位置呼叫這些 API 操作，但 AWS KMS 支援全域政策條件，可讓您根據來源 IP 地址、VPC 和 VPC 端點控制對 KMS 金鑰的存取。您可以在金鑰政策和 IAM 政策中使用這些條件索引鍵。但是，這些條件可以防止 AWS 代表您使用 KMS 金鑰。如需詳細資訊，請參閱 AWS 全域條件索引鍵。

例如，下列金鑰政策陳述式允許擔任 KMSTestRole 角色的使用者使用該 AWS KMS key 進行指定的密碼編譯操作，除非來源 IP 地址是政策中指定的其中一個 IP 地址。

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Effect": "Allow",
    "Principal": {"AWS":
    "arn:aws:iam::111122223333:role/KMSTestRole"},
    "Action": [
      "kms:Encrypt",
      "kms:Decrypt",
      "kms:ReEncrypt*",
      "kms:GenerateDataKey*",
      "kms:DescribeKey"
    ],
    "Resource": "*",
    "Condition": {
      "NotIpAddress": {
        "aws:SourceIp": [
          "192.0.2.0/24",
          "203.0.113.0/24"
        ]
      }
    }
  }
}
```

實體主機的隔離

AWS KMS 使用的實體基礎設施的安全受限於 Amazon Web Services：安全程序概觀的實體和環境安全章節中所述的控制項。您可以在上一節所列的合規報告和第三方稽核問題清單中找到更多詳細資訊。

AWS KMS 由專用強化硬體安全模組 (HSM) 支援，該模組具備特定控制項，可抵禦實體攻擊。HSM 是不具有虛擬化層 (例如 Hypervisor) 的實體裝置，可在數個邏輯租用戶之間共用實體裝置。AWS KMS keys 金鑰材料只會存放在 HSM 的揮發性記憶體中，而且只有在使用 KMS 金鑰時才會存放。當 HSM 移出操作狀態時，此記憶體會被清除，包括預期和非預期的關機和重設。如需 AWS KMS HSM 操作的詳細資訊，請參閱 AWS Key Management Service 密碼編譯詳細資訊。

AWS Key Management Service 的安全最佳實務

AWS Key Management Service (AWS KMS) 支援多項安全功能，您可以實作這些功能來增強對加密金鑰的保護，包括金鑰政策和 IAM 政策；一個加密內容選項，用於對稱加密金鑰的加密操作；廣泛的條件金鑰集，用於優化金鑰政策和 IAM 政策；以及用於限制授權授與限制。

如需這些安全功能的詳細說明，請參閱 AWS Key Management Service 最佳實務 (PDF)。此技術白皮書中的一般準則不代表完整的安全解決方案。由於並非所有的最佳實務都適用於所有情形，因此這些實務並非為規範性的。

另請參閱

- IAM 政策的最佳實務
- AWS KMS 授予的最佳實務
- 《IAM 使用者指南》中的 IAM 安全最佳實務。

配額

為了為所有使用者提高 AWS KMS 的回應能力和效能，AWS KMS 會套用兩種配額類型，資源配額和請求配額。每一種配額都會針對每個 AWS 帳戶 的每個區域獨立計算。

所有 AWS KMS 配額皆為可調整，但金鑰政策文件大小資源配額與 AWS CloudHSM金鑰存放區請求配額除外。若要請求提高配額，請參閱《Service Quotas 使用者指南》中的請求提高配額。若要請求減少配額、變更 Service Quotas 中未列出的配額、或變更 AWS 區域 (在其中無法針對 AWS KMS 使用 Service Quotas) 中的配額，請造訪 AWS Support 中心並建立案例。

主題

資源配額

AWS KMS 會設立資源配額，確保能夠為所有客戶提供快速且有彈性的服務。有些資源配額只適用於您建立的資源，但不適用於 AWS 服務為您建立的資源。您使用但並非位於您 AWS 帳戶 中的資源 (例如 AWS 擁有的金鑰) 不會計入這些配額中。

如果您已超出資源限制，請求建立該類型的其他資源會產生 `LimitExceededException` 錯誤訊息。

所有 AWS KMS 資源配額皆為可調整，但金鑰政策文件大小配額除外。若要請求提高配額，請參閱《Service Quotas 使用者指南》中的請求提高配額。若要請求減少配額、變更 Service Quotas 中未列出的配額、或變更 AWS 區域 (在其中無法針對 AWS KMS 使用 Service Quotas) 中的配額，請造訪 AWS Support 中心並建立案例。

下表列出及說明每個 AWS 帳戶 和區域中的 AWS KMS 資源配額。

配額名稱	預設值	適用對象	可調整
AWS KMS keys	100,000	客戶受管金鑰	是
每個 KMS 金鑰的別名	50	客戶建立的別名	是
每個 KMS 金鑰的授予	50,000	客戶受管金鑰	是

配額名稱	預設值	適用對象	可調整
金鑰政策文件大小	32 KB (32,768 位元組)	客戶受管金鑰 AWS 受管金鑰	否
自訂金鑰存放區資源配額	10	AWS 帳戶 和區域	是

除了資源配額外，AWS KMS 還使用請求配額來確保服務的回應速度。如需詳細資訊，請參閱 the section called “請求配額”。

AWS KMS keys：100,000 個

在您 AWS 帳戶 的每個區域，您最多可有 100,000 個客戶受管金鑰。此配額適用於所有 AWS 區域 中所有的客戶受管金鑰，與其金鑰規格或金鑰狀態無關。每個 KMS 金鑰都會被視為一個資源。AWS 受管金鑰 和 AWS 擁有的金鑰 不會計入此配額。

每個 KMS 金鑰的別名：50

您最多可以將 50 個別名與每個客戶受管金鑰關聯。AWS 與 AWS 受管金鑰 關聯的別名不會計入此配額。建立或更新別名時，您可能會遇到此配額。

> **Note**
>
> kms: ResourceAliases 條件只有在 KMS 金鑰符合此配額時才有效。如果 KMS 金鑰超過此配額，則會拒絕透過 `kms:ResourceAliases` 條件授權使用 KMS 金鑰的委託人存取 KMS 金鑰。如需詳細資訊，請參閱 因別名配額而拒絕存取。

每個 KMS 金鑰的別名配額會取代每個區域的別名配額，這些配額限制了 AWS 帳戶 的每個區域中別名的總數。AWS KMS 已經消除了每個區域的別名配額。

每個 KMS 金鑰的授予：50,000

每個客戶受管金鑰最多可有 50,000 個授予，包括與 AWS KMS 整合的 AWS 服務所建立的授予。此配額不適用於 AWS 受管金鑰 或 AWS 擁有的金鑰。

此配額的其中一個效果是，您無法同時執行超過 50,000 個使用相同 KMS 金鑰的授予授權操作。達到配額後，只有在作用中的授予已淘汰或撤銷時，您才能在 KMS 金鑰上建立新的授予。

例如，當您將 Amazon Elastic Block Store (Amazon EBS) 磁碟區連接到 Amazon Elastic Compute Cloud (Amazon EC2) 執行個體時，磁碟區便會解密，讓您可以讀取。為了取得解密資料的許可，Amazon EBS 會為每個磁碟區建立授予。因此，如果您所有的 Amazon EBS 磁碟區都使用相同的 KMS 金鑰，您無法一次連接超過 50,000 個磁碟區。

金鑰政策文件大小：32 KB

每個金鑰政策文件的最大長度為 32 KB (32,768 位元組)。如果您使用較大的政策文件來建立或更新 KMS 金鑰的金鑰政策，操作便會失敗。

此配額不可調整。您無法藉由在 AWS Support 中使用 Service Quotas 或建立案例來增加它。如果您的金鑰政策即將超出限制，請考慮使用授予而非政策陳述式。授予特別適合暫時或非常特定的許可。

每當您使用或作業中的預設檢視或原則檢視來建立或變更金鑰原則時AWS Management Console，都可以使用金鑰原則文PutKeyPolicy件。此配額適用於您的金鑰政策文件，即使您使用 AWS KMS 主控台中的預設檢視，而在主控台您不會直接編輯 JSON 陳述式。

自訂金鑰存放區資源配額：10

您可以在每個 AWS 帳戶 和區域中建立最多 10 個自訂金鑰存放區。如果您嘗試建立更多，作CreateCustomKeyStore業會失敗。

此配額適用於每個帳戶和區域中自訂金鑰存放區的總數，包括 AWS CloudHSM 金鑰存放區和外部金鑰存放區，無論其連接狀態為何。

請求配額

AWS KMS 建立每秒請求的 API 操作數配額。請求配額會因 API 操作、AWS 區域 和其他因素 (例如 KMS 金鑰類型) 而不同。當您超過 API 請求配額時，AWS KMS 會調節請求。

所有 AWS KMS 請求配額皆可調整，但 AWS CloudHSM 金鑰存放區請求配額除外。若要請求提高配額，請參閱《Service Quotas 使用者指南》中的請求提高配額。若要請求減少配額、變更 Service Quotas 中未列出的配額、或變更 AWS 區域 (在其中無法針對 AWS KMS 使用 Service Quotas) 中的配額，請造訪 AWS Support 中心並建立案例。

如果您超出GenerateDataKey作業的要求配額，請考慮使用的資料金鑰快取功能AWS Encryption SDK。重複使用資料金鑰可降低您向 AWS KMS 提出請求的頻率。

除了請求配額之外，AWS KMS 會使用資源配額來確保所有使用者的容量。如需詳細資訊，請參閱 資源配額。

若要檢視請求率的趨勢，請使用 Service Quotas 主控台。您也可以建立 Amazon CloudWatch 警示，以便在請求率達到配額值的特定百分比時提醒您。如需詳細資訊，請參閱AWS安全部落格 CloudWatch 中的使用 Service Quotas 和 Amazon 管理 AWS KMS API 請求費率。

主題

每個 AWS KMS API 操作的請求配額

此表格列出「Service Quotas」配額代碼，以及每個 AWS KMS 請求配額的預設值。所有 AWS KMS 請求配額皆可調整，但 AWS CloudHSM 金鑰存放區請求配額除外。

> **Note**
>
> 您可能需要水平或垂直捲動，才能查看此資料表中的所有資料。

配額名稱	預設值 (每秒請求數)

`Cryptographic operations (symmetric) request rate`
 	
適用對象：
 	
• `Decrypt`
• `Encrypt`
• `GenerateDataKey`
	這些共用配額會因 AWS 區域 和請求中所使用的 KMS 金鑰類型而不同。每個配額會分別計算。
	
	• 5,500 (共用)
	• 下列區域中為 10,000 (共享)：
	  • 美國東部 (俄亥俄)，us-east-2
	  • 亞太區域 (新加坡) ap-southeast-1
	  • 亞太區域 (雪梨)，ap-southeast-2

配額名稱	預設值 (每秒請求數)
• GenerateDataKeyWithoutPlaintext • GenerateMac • GenerateRandom • ReEncrypt • VerifyMac	• 亞太區域 (東京)，ap-northeast-1 • 歐洲 (法蘭克福)，eu-central-1 • 歐洲 (倫敦)，eu-west-2 • 下列區域中為 50,000 (共用)： • 美國東部 (維吉尼亞北部)，us-east-1 • 美國西部 (奧勒岡)，us-west-2 • 歐洲 (愛爾蘭)，eu-west-1
Cryptographic operations (RSA) request rate 適用對象： • Decrypt • Encrypt • ReEncrypt • Sign • Verify	RSA KMS 金鑰為 500 (共用)
Cryptographic operations (ECC) request rate 適用對象： • Sign • Verify	橢圓曲線 (ECC) KMS 金鑰為 300 (共用)

配額名稱	預設值 (每秒請求數)
Cryptographic operations (SM) request rate 適用對象： • Decrypt • Encrypt • ReEncrypt • Sign • Verify	SM2 (僅限中國區域) KMS 金鑰為 300 (共用)
Custom key store request quotas 適用對象： • Decrypt • Encrypt • GenerateDataKey • GenerateDataKeyWithoutPlaintext • GenerateRandom • ReEncrypt	會針對每個自訂金鑰存放區分別計算自訂金鑰存放區請求配額 • 每個 AWS CloudHSM 金鑰存放區 1800 個 (共用) • 每個外部金鑰存放區 1800 個 (共用)
CancelKeyDeletion request rate	5
ConnectCustomKeyStore request rate	5
CreateAlias request rate	5
CreateCustomKeyStore request rate	5
CreateGrant request rate	50
CreateKey request rate	5

配額名稱	預設值 (每秒請求數)
DeleteAlias request rate	15
DeleteCustomKeyStore request rate	5
DeleteImportedKeyMaterial request rate	5
DescribeCustomKeyStores request rate	5
DescribeKey request rate	2000
DisableKey request rate	5
DisableKeyRotation request rate	5
DisconnectCustomKeyStore request rate	5
EnableKey request rate	5
EnableKeyRotation request rate	15
GenerateDataKeyPair (ECC_NIST_P256) request rate 適用對象： • GenerateDataKeyPair • GenerateDataKeyPairWithoutPlaintext	100

配額名稱	預設值 (每秒請求數)
GenerateDataKeyPair (ECC_NIST_P384) request rate 適用對象： • GenerateDataKeyPair • GenerateDataKeyPairWithoutPlaintext	100
GenerateDataKeyPair (ECC_NIST_P521) request rate 適用對象： • GenerateDataKeyPair • GenerateDataKeyPairWithoutPlaintext	100
GenerateDataKeyPair (ECC_SECG_P256K1) request rate 適用對象： • GenerateDataKeyPair • GenerateDataKeyPairWithoutPlaintext	100
GenerateDataKeyPair (RSA_2048) request rate 適用對象： • GenerateDataKeyPair • GenerateDataKeyPairWithoutPlaintext	1

配額名稱	預設值 (每秒請求數)
GenerateDataKeyPair (RSA_3072) request rate 適用對象： • GenerateDataKeyPair • GenerateDataKeyPairWithoutPlaintext	0.5 (每 2 秒間隔 1 個)
GenerateDataKeyPair (RSA_4096) request rate 適用對象： • GenerateDataKeyPair • GenerateDataKeyPairWithoutPlaintext	0.1 (每 10 秒間隔 1 個)
GenerateDataKeyPair (SM2 – China Regions only) request rate 適用對象： • GenerateDataKeyPair • GenerateDataKeyPairWithoutPlaintext	25
GetKeyPolicy request rate	1000
GetKeyRotationStatus request rate	1000
GetParametersForImport request rate	0.25 (每 4 秒間隔 1 個)
GetPublicKey request rate	2000
ImportKeyMaterial request rate	5

配額名稱	預設值 (每秒請求數)
ListAliases request rate	500
ListGrants request rate	100
ListKeyPolicies request rate	100
ListKeys request rate	500
ListResourceTags request rate	2000
ListRetirableGrants request rate	100
PutKeyPolicy request rate	15
ReplicateKey request rate ReplicateKey 操作在主要金鑰的區域中計為一個 ReplicateKey 請求，在複本區域中計為兩個 CreateKey 請求。其中一個 CreateKey 請求是試執行，以在建立金鑰之前偵測潛在的問題。	5
RetireGrant request rate	30
RevokeGrant request rate	30
ScheduleKeyDeletion request rate	15
TagResource request rate	10
UntagResource request rate	5
UpdateAlias request rate	5
UpdateCustomKeyStore request rate	5
UpdateKeyDescription request rate	5

配額名稱	預設值 (每秒請求數)
UpdatePrimaryRegion request rate UpdatePrimaryRegion 操作會計為兩個 UpdatePrimaryRegion 請求；每兩個受影響的區域中一個請求。	5

套用請求配額

檢閱請求配額時，請注意以下資訊。

- 請求配額同時適用於客戶受管金鑰和 AWS 受管金鑰。使用 AWS 擁有的金鑰 不會計入您 AWS 帳戶的請求配額，即使使用其來保護您帳戶中的資源也一樣。
- 請求配額適用於傳送至 FIPS 端點和非 FIPS 端點的請求。如需 AWS KMS 服務端點清單，請參閱《AWS 一般參考》的AWS Key Management Service端點與配額。
- 調節是以區域中所有類型 KMS 金鑰的所有請求為基礎。此總和包括 AWS 帳戶 中所有委託人的請求，包括 AWS 服務代您發出的請求。
- 每個請求配額都是獨立計算的。例如，CreateKey作業的要求不會影響作CreateAlias業的要求配額。如果您的 `CreateAlias` 請求遭到調節，您的 `CreateKey` 請求仍然可以成功完成。
- 雖然密碼編譯操作會共用配額，但是共用的配額會獨立於其他操作配額之外進行計算。例如，對「加密」和「解密」作業的呼叫共用要求配額，但該配額與管理作業的配額無關，例如EnableKey。例如，在歐洲 (倫敦) 區域，您可以在對稱 KMS 金鑰上執行 10,000 次密碼編譯操作，加上每秒 5 次 `EnableKey` 操作，而仍不會遭到調節。

密碼編譯操作的共用配額

AWS KMS 密碼編譯操作共用請求配額。您可以請求 KMS 金鑰所支援密碼編譯操作的任何組合，只要密碼編譯操作的總數不會超過該 KMS 金鑰類型的請求配額即可。例外是GenerateDataKeyPair和GenerateDataKeyPairWithoutPlaintext，它們共享一個單獨的配額。

不同 KMS 金鑰類型的配額也會獨立計算。每個配額在每一秒間隔的中均適用於 AWS 帳戶 和區域中具有指定金鑰類型的這些操作的所有請求。

- 密碼編譯操作 (對稱) 請求率是帳戶和區域中使用對稱 KMS 金鑰進行加密編譯操作的共用請求配額。此配額適用於具有對稱加密金鑰和 HMAC 金鑰 (兩者皆為對稱金鑰) 的密碼編譯操作。

例如，您可能使用 AWS 區域 中的對稱 KMS 金鑰，共用配額為每秒 10,000 個請求。當您每秒發出 7,000 個GenerateDataKey請求和每秒 2,000 個解密請求時，AWS KMS不會限制您的請求。但是，當您每秒發出 9,500 次 `GenerateDataKey` 請求及 1,000 次 Encrypt 請求時，AWS KMS 便會進行調節，因為其超過了共用配額。

自訂金鑰存放區中對稱加密 KMS 金鑰的密碼編譯操作會計入帳戶的密碼編譯操作 (對稱) 請求率以及自訂金鑰存放區的自訂金鑰存放區請求配額。

- 密碼編譯操作 (RSA) 請求率是使用 RSA 非對稱 KMS 金鑰進行加密編譯操作的共用請求配額。

  例如，如果請求配額為每秒 500 個操作，您可以使用 RSA KMS 金鑰 (可加密和解密) 製作 200 個加密請求和 100 個解密請求，加上使用 RSA KMS 金鑰 (可簽署和驗證) 製作 50 個簽署請求和 150 個驗證請求。

- 密碼編譯操作 (ECC) 請求率是使用橢圓曲線 (ECC) 非對稱 KMS 金鑰進行加密編譯操作的共用請求配額。

  例如，如果請求配額為每秒 300 個操作，您可以使用 RSA KMS 金鑰 (可簽署和驗證) 製作 100 個「簽署」請求和 200 個「驗證」請求。

- 密碼編譯操作 (SM — 僅限中國區域) 請求率是使用 SM 非對稱 KMS 金鑰進行加密編譯操作的共用請求配額。

  例如，如果請求配額為每秒 300 個操作，您可以使用 SM2 KMS 金鑰 (可加密和解密) 製作 100 個加密請求和 100 個解密請求，加上使用 SM2 KMS 金鑰 (可簽署和驗證) 製作 50 個簽署請求和 50 個驗證請求。

- 自訂金鑰存放區請求配額是自訂金鑰存放區中 KMS 金鑰之密碼編譯操作的共用請求配額。此配額會針對每個自訂金鑰存放區分別計算。

  自訂金鑰存放區中對稱加密 KMS 金鑰的密碼編譯操作會計入帳戶的密碼編譯操作 (對稱) 請求率以及自訂金鑰存放區的自訂金鑰存放區請求配額。

不同金鑰類型的配額也會獨立計算。例如，在亞太區域 (新加坡) 區域中，如果使用對稱和非對稱 KMS 金鑰，您最多可以使用對稱 KMS 金鑰 (包括 HMAC 金鑰) 每秒進行 10,000 次呼叫，加上使用 RSA 非對稱 KMS 金鑰每秒進行最多 500 次額外的呼叫，再加上使用 ECC 型 KMS 金鑰每秒進行最多 300 次額外的請求。

代表您進行的 API 請求

您可以直接提出 API 請求，或使用整合的 AWS 服務來代表您向 AWS KMS 提出 API 請求。此配額同時適用於這兩種請求。

例如，您可能將資料存放在使用伺服器端加密搭配 KMS 金鑰 (SSE-KMS) 的 Amazon S3 上。每次上傳或下載使用 SSE-KMS 加密的 S3 物件時，Amazon S3 都會代表您向 AWS KMS 提出 `GenerateDataKey` (適用於上傳) 或 `Decrypt` (適用於下載) 請求。這些請求會計入您的配額，因此如果您總計每秒上傳或下載超過 5,500 (或 10,000 或 50,000，取決於您的 AWS 區域) 個以 SSE-KMS 加密的 S3 物件，AWS KMS 就會調節請求。

跨帳戶請求

當 AWS 帳戶 中的應用程式使用由不同帳戶擁有的 KMS 金鑰時，這就稱為跨帳戶請求。對於跨帳戶請求，AWS KMS 會調節發出請求的帳戶，而不是擁有 KMS 金鑰的帳戶。例如，如果帳戶 A 中的應用程式使用帳戶 B 中的 KMS 金鑰，使用該 KMS 金鑰就會套用帳戶 A 中的配額。

自訂金鑰存放區請求配額

AWS KMS 會維持自訂金鑰存放區中 KMS 金鑰的密碼編譯操作的請求配額。會針對每個自訂金鑰存放區分別計算這些請求配額。

自訂金鑰存放區請求配額	每個自訂金鑰存放區的預設值 (每秒請求數)	可調整
AWS CloudHSM 金鑰存放區請求配額	1800	否
外部金鑰存放區請求配額	1800	是

> **Note**
>
> AWS KMS 自訂金鑰存放區請求配額不會顯示在 Service Quotas 主控台中。您無法使用 Service Quotas API 操作來檢視或管理這些配額。若要請求變更外部金鑰存放區配額，請造訪 AWS Support 中心並建立案例。
>
> 如果與AWS CloudHSM金鑰存放區關聯的AWS CloudHSM叢集正AWS KMS`ThrottlingException`在處理許多命令，包括與自訂金鑰存放區無關的命令，您可能會以一定的 lower-than-expected 速率取得。如果發生這種情況，請降低您對 AWS KMS 的請

求速率、降低不相關的負載，或對 AWS CloudHSM 金鑰存放區使用專用的 AWS CloudHSM 叢集。
AWS KMS報告指ExternalKeyStoreThrottle CloudWatch 標中外部金鑰存放區要求的限制。您可以使用此指標來檢視限流模式、建立警示以及調整外部金鑰存放區請求配額。

自訂金鑰存放區 KMS 金鑰的密碼編譯操作請求會計入兩個配額：

- 密碼編譯操作 (對稱) 請求率配額 (每個帳戶)

  自訂金鑰存放區中 KMS 金鑰的密碼編譯操作請求會計入每個 AWS 帳戶 和區域的 Cryptographic operations (symmetric) request rate 配額。例如，在美國東部 (維吉尼亞北部) (us-east-1)，每個 AWS 帳戶 對於對稱加密 KMS 金鑰每秒最多可有 50,000 個請求，包括在自訂金鑰存放區中使用 KMS 金鑰的請求。
- 自訂金鑰存放區請求配額 (每個自訂金鑰存放區)

  自訂金鑰存放區中 KMS 金鑰的密碼編譯操作請求也會計入每秒 1,800 次 Custom key store request quota 操作中。會針對每個自訂金鑰存放區分別計算這些配額。它們可能包括來自多個 AWS 帳戶 的請求，它們在自訂金鑰存放區中使用 KMS 金鑰。

例如，對美國東部 (維吉尼亞北部) (us-east-1) 區域的自訂金鑰存放區 (任一類型) 中 KMS 金鑰的加密操作會計入其帳戶和區域的 Cryptographic operations (symmetric) request rate 帳戶層級配額 (每秒 50,000 個請求)，並計入其自訂金鑰存放區的 Custom key store request quota (每秒 1,800 個請求) 。不過，管理作業 (例如 PutKeyPolicy，自訂金鑰存放區中的 KMS 金鑰) 的要求只會套用至其帳戶層級配額 (每秒 15 個要求)。

調節 AWS KMS 請求

為了確保 AWS KMS 可以針對來自所有客戶的 API 請求提供快速且可靠的回應，它會調節超出特定界限的 API 請求。

AWS KMS 拒絕可能有效的請求並傳回類似如下的 ThrottlingException 錯誤，則會發生限流。

```
You have exceeded the rate at which you may call KMS. Reduce the frequency of your
 calls.
(Service: AWSKMS; Status Code: 400; Error Code: ThrottlingException; Request ID: <ID>
```

下列條件的 AWS KMS 調節請求。

- 每秒的請求率超過帳戶和區域的 AWS KMS 請求配額。

  例如，如果您帳戶中的使用者提交 1000 個 `DescribeKey` 請求，則 AWS KMS 會調節該秒之中所有後續 `DescribeKey` 請求。

  如要回應調節，請使用退避與重試策略。針對某些 AWS 開發套件中的 HTTP 400 錯誤，這個策略會自動實作。
- 變更相同 KMS 金鑰狀態之高載或持續高水準的請求率。這種情況通常稱為「快速鍵」。

  例如，如果您帳戶中的應用程式針對相同 KMS 金鑰同時且持續傳送 `EnableKey` 和 `DisableKey` 請求，則 AWS KMS 會調節請求。即使要求未超過`EnableKey`和`DisableKey`作業的要 request-per-second 求限制，也會發生此節流。

  若要回應調節，請調整您的應用程式邏輯，讓其只提出必要的請求，或合併多個函數的請求。
- 當與金鑰存放區關聯的AWS CloudHSM叢集正在處理大量命令 (包括與金AWS CloudHSM鑰存放區無關的命令) 時，金AWS CloudHSM鑰存放區中 KMS 金鑰的作業要求可能會以一定 lower-than-expected 速率限制。AWS CloudHSM

  (當 AWS CloudHSM 叢集無可用的 PKCS #11 工作階段時，AWS KMS 不會再對 AWS CloudHSM 金鑰存放區 KMS 金鑰的操作請求進行限制。反之，會擲出 `KMSInternalException` 並建議重試請求。)

若要檢視請求率的趨勢，請使用 Service Quotas 主控台。您也可以建立 Amazon CloudWatch 警示，以便在請求率達到配額值的特定百分比時提醒您。如需詳細資訊，請參閱AWS安全部落格 CloudWatch 中的使用 Service Quotas 和 Amazon 管理 AWS KMS API 請求費率。

所有 AWS KMS 配額皆為可調整，但金鑰政策文件大小資源配額與 AWS CloudHSM金鑰存放區請求配額除外。若要請求提高配額，請參閱《Service Quotas 使用者指南》中的請求提高配額。若要請求減少配額、變更 Service Quotas 中未列出的配額、或變更 AWS 區域 (在其中無法針對 AWS KMS 使用 Service Quotas) 中的配額，請造訪 AWS Support 中心並建立案例。

> **Note**
>
> AWS KMS 自訂金鑰存放區請求配額不會顯示在 Service Quotas 主控台中。您無法使用 Service Quotas API 操作來檢視或管理這些配額。若要請求變更外部金鑰存放區配額，請造訪 AWS Support 中心並建立案例。

AWS 服務使用 AWS KMS 的方式

許多 AWS 服務使用 AWS KMS 以支援加密您的資料。AWS 服務已與 AWS KMS 整合，因此您可以使用您帳戶中的 AWS KMS keys 來保護服務為您接收、存放及管理的資料。如需已與 AWS KMS 整合之 AWS 服務的完整清單，請參閱 AWS 服務整合。

下列主題詳細討論特定服務如何使用 AWS KMS，包括它們支援的 KMS 金鑰、如何管理資料金鑰、所需的許可，以及如何追蹤每個服務對您帳戶中 KMS 金鑰的使用情形。

> **Important**
>
> 與 AWS KMS 整合的 AWS 服務僅會使用對稱加密 KMS 金鑰來加密您的資料。這些服務不支援使用非對稱 KMS 金鑰進行加密。如需有關如何判斷 KMS 金鑰是對稱還是不對稱的說明，請參閱識別非對稱 KMS 金鑰。

主題

AWS CloudTrail 使用 AWS KMS 的方式

您可以使用 AWS CloudTrail 記錄您 AWS 帳戶 的 AWS API 呼叫和其他活動，並將記錄的資訊儲存在您所選 Amazon Simple Storage Service (Amazon S3) 的日誌檔案中。根據預設， CloudTrail 放入 S3 儲存貯體的日誌檔會使用伺服器端加密搭配 Amazon S3 受管加密金鑰 (SSE-S3) 進行加密。但是，您可以選擇改為使用伺服器端加密搭配 KMS 金鑰 (SSE-KMS)。若要了解如何使用加密 CloudTrail 記錄檔AWS KMS，請參閱使用指南中的使用 AWS KMS keys (SSE-KMS) 加密 CloudTrail 記錄檔。AWS CloudTrail

> **Important**
>
> AWS CloudTrail 和 Amazon S3 只支援對稱 AWS KMS keys。您無法使用非對稱 KMS 金鑰來加密 CloudTrail 記錄。如需有關如何判斷 KMS 金鑰是對稱還是不對稱的說明，請參閱識別非對稱 KMS 金鑰。

CloudTrail 讀取或寫入使用 SSE-KMS 金鑰加密的記錄檔時，不需支付金鑰使用費。不過，當您存取使用 SSE-KMS 金鑰加密的 CloudTrail 記錄檔時，需要支付金鑰使用費。如需 AWS KMS 定價的資訊，請參閱 AWS Key Management Service 定價。如需有關 CloudTrail 定價的資訊，請參閱AWS CloudTrail使用者指南中的AWS CloudTrail定價和管理成本。

主題

- 了解 KMS 金鑰的使用時機

了解 KMS 金鑰的使用時機

使用 Amazon S3 上的AWS KMS建置功能加密 CloudTrail 日誌檔，稱為伺服器端加密 AWS KMS key (SSE-KMS)。若要進一步了解 SSE-KMS，請參閱本指南中的 Amazon Simple Storage Service (Amazon S3) 如何使用 AWS KMS，或《Amazon Simple Storage Service 使用者指南》中的使用伺服器端加密搭配 KMS 金鑰 (SSE-KMS) 保護資料。

當您設定AWS CloudTrail為使用 SSE-KMS 加密記錄檔時， CloudTrail Amazon S3 會在您對這些服務執行特定動作AWS KMS keys時使用您的。以下章節說明這些服務使用您的 KMS 金鑰的時機和方法，並提供您可用來驗證此說明的額外資訊。

導致 CloudTrail 和 Amazon S3 使用您的 KMS 金鑰的動作

- 您配置 CloudTrail 為使用您的加密日誌文件 AWS KMS key

- CloudTrail 將日誌文件放入您的 S3 存儲桶
- 您從 S3 儲存貯體取得加密的日誌檔

您配置 CloudTrail 為使用您的加密日誌文件 AWS KMS key

當您更新 CloudTrail組態以使用 KMS 金鑰時， CloudTrail 會傳送GenerateDataKeyAWS KMS要求以確認 KMS 金鑰是否存在，且 CloudTrail 具有將其用於加密的權限。 CloudTrail 不使用產生的資料金鑰。

GenerateDataKey 請求包含加密內容的下列資訊：

- 追踪的 Amazon 資源名稱（ARN） CloudTrail
- S3 儲存貯體的 ARN 以及傳送 CloudTrail 日誌檔案的路徑

要GenerateDataKey求會在 CloudTrail 記錄檔中產生類似下列範例的項目。當您看到類似這樣的記錄項目時，您可以判斷 CloudTrail (1) 呼叫 AWS KMS (2) GenerateDataKey 操作 (3) 的特定追蹤 (4)。AWS KMS在特定 KMS 金鑰 (5) 下建立資料金鑰。

> **Note**
>
> 您可能需要捲動至右側，才能看見下列範例日誌項目中的一些圖說文字。

```
{
  "eventVersion": "1.02",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
```

```
    "arn": "arn:aws:iam::086441151436:user/
AWSCloudTrail",
    "accountId": "086441151436",
    "accessKeyId": "AKIAI44QH8DHBEXAMPLE",
    "userName": "AWSCloudTrail",
    "sessionContext": {"attributes": {
      "mfaAuthenticated": "false",
      "creationDate": "2015-11-11T21:15:33Z"
    }},
    "invokedBy": "internal.amazonaws.com"
  },
  "eventTime": "2015-11-11T21:15:33Z",
  "eventSource":
 "kms.amazonaws.com",
  "eventName":
 "GenerateDataKey",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "internal.amazonaws.com",
  "userAgent": "internal.amazonaws.com",
  "requestParameters": {
    "keyId": "arn:aws:kms:us-west-2:111122223333:alias/ExampleAliasForCloudTrailKMS
 key",
    "encryptionContext": {
      "aws:cloudtrail:arn": "arn:aws:cloudtrail:us-west-2:111122223333:trail/
Default",
      "aws:s3:arn": "arn:aws:s3:::example-bucket-for-CT-logs/AWSLogs/111122223333/"
    },
    "keySpec": "AES_256"
  },
  "responseElements": null,
  "requestID": "581f1f11-88b9-11e5-9c9c-595a1fb59ac0",
  "eventID": "3cdb2457-c035-4890-93b6-181832b9e766",
  "readOnly": true,
  "resources": [{
    "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "accountId": "111122223333"
  }],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "111122223333"
}
```

CloudTrail 將日誌文件放入您的 S3 存儲桶

每次將日誌檔 CloudTrail 放入 S3 儲存貯體時，Amazon S3 都會代表向AWS KMS其傳送GenerateDataKey請求 CloudTrail。為了回應該請求，AWS KMS 會產生唯一資料金鑰，接著再傳送兩個資料金鑰複本至 Amazon S3，一個是純文字複本，另一個是使用指定 KMS 金鑰加密的複本。Amazon S3 使用純文字資料金鑰加密 CloudTrail 日誌檔，然後在使用後儘快從記憶體中移除純文字資料金鑰。Amazon S3 會將加密的資料金鑰存放為中繼資料，其中包含加密的 CloudTrail 日誌檔。

GenerateDataKey 請求包含加密內容的下列資訊：

- 追踪的 Amazon 資源名稱（ARN） CloudTrail
- S3 物件 (CloudTrail 記錄檔) 的 ARN

每個GenerateDataKey要求都會在 CloudTrail 記錄檔中產生一個項目，類似於下列範例。當您看到類似這樣的記錄項目時，您可以針對特定的追蹤 AWS KMS
(2)
判斷
(3)
呼叫 () GenerateDataKey 操作
(4)
以保護特定的記錄檔
(5)
CloudTrail
1
AWS KMS在指定的 KMS 金鑰
(6)
下建立資料金鑰，在相同的記錄項目中顯示兩次。

> **Note**
>
> 您可能需要捲動至右側，才能看見下列範例日誌項目中的一些圖說文字。

```
{
  "eventVersion": "1.02",
  "userIdentity": {
    "type": "AssumedRole",
```

```
    "principalId": "AROACKCEVSQ6C2EXAMPLE:i-34755b85",
    "arn": "arn:aws:sts::086441151436:assumed-role/AWSCloudTrail/
i-34755b85",(1)
    "accountId": "086441151436",
    "accessKeyId": "AKIAI44QH8DHBEXAMPLE",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2015-11-11T20:45:25Z"
      },
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROACKCEVSQ6C2EXAMPLE",
        "arn": "arn:aws:iam::086441151436:role/AWSCloudTrail",
        "accountId": "086441151436",
        "userName": "AWSCloudTrail"
      }
    },
    "invokedBy": "internal.amazonaws.com"
  },
  "eventTime": "2015-11-11T21:15:58Z",
  "eventSource":
 "kms.amazonaws.com",(2)
  "eventName":
 "GenerateDataKey",(3)
  "awsRegion": "us-west-2",
  "sourceIPAddress": "internal.amazonaws.com",
  "userAgent": "internal.amazonaws.com",
  "requestParameters": {
    "encryptionContext": {
      "aws:cloudtrail:arn": "arn:aws:cloudtrail:us-west-2:111122223333:trail/
Default",(4)
      "aws:s3:arn": "arn:aws:s3:::example-bucket-for-CT-logs/
AWSLogs/111122223333/CloudTrail/us-west-2/2015/11/11/111122223333_CloudTrail_us-
west-2_20151111T2115Z_7JREEBimdK8d2nC9.json.gz"(5)
    },
    "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",(6)
    "keySpec": "AES_256"
  },
  "responseElements": null,
  "requestID": "66f3f74a-88b9-11e5-b7fb-63d925c72ffe",
```

```
  "eventID": "7738554f-92ab-4e27-83e3-03354b1aa898",
  "readOnly": true,
  "resources": [{
    "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",(6)
    "accountId": "111122223333"
  }],
  "eventType": "AwsServiceEvent",
  "recipientAccountId": "111122223333"
}
```

您從 S3 儲存貯體取得加密的日誌檔

每次您從 S3 儲存貯體取得加密的 CloudTrail 日誌檔時，Amazon S3 都會代表您傳送Decrypt請求，以解密日誌檔的加密資料金鑰。AWS KMS為了回應該請求，AWS KMS 使用您的 KMS 金鑰來解密資料金鑰，接著再將純文字資料金鑰傳送至 Amazon S3。Amazon S3 使用純文字資料金鑰解密 CloudTrail 日誌檔，然後在使用後儘快從記憶體中移除純文字資料金鑰。

Decrypt 請求包含加密內容的下列資訊：

- 追踪的 Amazon 資源名稱（ARN） CloudTrail
- S3 物件 (CloudTrail 記錄檔) 的 ARN

每個Decrypt要求都會在 CloudTrail 記錄檔中產生一個項目，類似於下列範例。當看到類似這個項目的日誌項目時，您可以判斷 AWS 帳戶 (1) 中的使用者針對特定追蹤 (4) 和特定日誌檔案 (5) 呼叫 AWS KMS (2) Decrypt 操作 (3)。AWS KMS 在指定的 KMS 金鑰 (6) 下解密了資料金鑰。

> **Note**
>
> 您可能需要捲動至右側，才能看見下列範例日誌項目中的一些圖說文字。

```
{
  "eventVersion": "1.02",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "AIDACKCEVSQ6C2EXAMPLE",
    "arn": "arn:aws:iam::111122223333:role/cloudtrail-
admin",(1)
    "accountId": "111122223333",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "userName": "cloudtrail-admin",
    "sessionContext": {"attributes": {
      "mfaAuthenticated": "false",
      "creationDate": "2015-11-11T20:48:04Z"
    }},
    "invokedBy": "signin.amazonaws.com"
  },
  "eventTime": "2015-11-11T21:20:52Z",
  "eventSource":
 "kms.amazonaws.com",(2)
  "eventName":
 "Decrypt",(3)
  "awsRegion": "us-west-2",
  "sourceIPAddress": "internal.amazonaws.com",
  "userAgent": "internal.amazonaws.com",
  "requestParameters": {
    "encryptionContext": {
      "aws:cloudtrail:arn": "arn:aws:cloudtrail:us-west-2:111122223333:trail/
Default",(4)
      "aws:s3:arn": "arn:aws:s3:::example-bucket-for-CT-logs/
AWSLogs/111122223333/CloudTrail/us-west-2/2015/11/11/111122223333_CloudTrail_us-
west-2_20151111T2115Z_7JREEBimdK8d2nC9.json.gz"(5)
    }
  },
  "responseElements": null,
  "requestID": "16a0590a-88ba-11e5-b406-436f15c3ac01",
  "eventID": "9525bee7-5145-42b0-bed5-ab7196a16daa",
```

```
  "readOnly": true,
  "resources": [{
    "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
    "accountId": "111122223333"
  }],
  "eventType": "AwsApiCall",
  "recipientAccountId": "111122223333"
}
```

Amazon DynamoDB 如何使用 AWS KMS

Amazon DynamoDB 是全受管、可擴充的 NoSQL 資料庫服務。DynamoDB 與 AWS Key Management Service (AWS KMS) 整合，以支援靜態加密伺服器端加密功能。

如果使用靜態加密，每當資料表保存到磁碟時，DynamoDB 就會以透明方式加密 DynamoDB 資料表中的所有客戶資料，包括其主要金鑰和本機及全域次要索引。(如果您的資料表有排序索引鍵，則某些標示範圍界限的排序索引鍵將會以純文字的格式存放在資料表中繼資料中)。當您存取資料表，DynamoDB 會以透明方式解密資料表資料。您不需要變更應用程式來使用或管理加密的資料表。

每當 DynamoDB 串流、全域資料表和備份時儲存到持久性媒體時，加密靜態也會保護這些物件。這個主題中關於資料表的陳述式也適用於這些物件。

所有 DynamoDB 資料表都會加密。沒有可以為新的或現有資料表啟用或停用加密的選項。根據預設，所有資料表都會在 DynamoDB 服務帳戶中使用 AWS 擁有的金鑰 加密。但是，您可以選取選項使用客戶受管金鑰或帳戶中 DynamoDB AWS 受管金鑰 加密部分或所有您的資料表。

如需有關針對 KMS 金鑰的 Amazon DynamoDB 支援的詳細資訊，請參閱《Amazon DynamoDB 開發人員指南》中的 DynamoDB 靜態加密。

Amazon Elastic Block Store (Amazon EBS) 如何使用 AWS KMS

本主題將詳細討論 Amazon Elastic Block Store (Amazon EBS) 如何使用 AWS KMS 加密磁碟區和快照。對於加密 Amazon EBS 磁碟區的基本說明，請參閱 Amazon EBS 加密。

主題

- Amazon EBS 加密
- 使用 KMS 金鑰和資料金鑰
- Amazon EBS 加密內容

- 偵測 Amazon EBS 失敗
- 使用 AWS CloudFormation 建立加密的 Amazon EBS 磁碟區

Amazon EBS 加密

當您連接加密 Amazon EBS 磁碟區到支援的 Amazon Elastic Compute Cloud (Amazon EC2) 執行個體類型，存放在磁碟區的靜態資料、磁碟輸入/輸出，以及從磁碟區建立的快照全部都會加密。加密在託管 Amazon EC2 執行個體的伺服器上進行。

所有 Amazon EBS 磁碟區類型都支援此功能。您存取加密磁碟區方式與存取其他磁碟區的方式相同；加密和解密的處理過程皆相當透明，不需要您、您的 EC2 執行個體或您的應用程式進行任何額外動作。加密磁碟區的快照會自動加密，而從加密快照建立的磁碟區也會自動加密。

EBS 磁碟區的加密狀態取決於您建立磁碟區時。您不能改變現有磁碟區的加密狀態。不過，您可以在加密和未加密的磁碟區之間遷移資料，並在複製快照時套用新的加密狀態。

Amazon EBS 預設支援可選的加密。您可以自動對所有新的 EBS 磁碟區以及 AWS 帳戶 和 區域中的快照複本啟用加密。此組態設定不會影響現有的磁碟區或快照。如需詳細資訊，請參閱《Amazon EC2 Linux 執行個體使用者指南》或《Amazon EC2 Windows 執行個體使用者指南》中的預設加密。

使用 KMS 金鑰和資料金鑰

建立加密的 Amazon EBS 磁碟區時，您可以指定 AWS KMS key。根據預設，Amazon EBS 會使用您帳戶 (aws/ebs) 中的 Amazon EBS AWS 受管金鑰。但是，您可以指定您建立和管理的客戶受管金鑰。

若要使用客戶受管金鑰，您必須授予 Amazon EBS 許可，才能代表您使用 KMS 金鑰。如需必要許可的清單，請參閱《Amazon EC2 Linux 執行個體使用者指南》或《Amazon EC2 Windows 執行個體使用者指南》中的 IAM 使用者許可。

> **Important**
>
> Amazon EBS 只支援對稱 KMS 金鑰。您無法使用非對稱 KMS 金鑰來加密 Amazon EBS 磁碟區。如需判斷 KMS 金鑰是對稱還是不對稱的說明，請參閱 識別非對稱 KMS 金鑰。

對於每個磁碟區，Amazon EBS 會要求 AWS KMS 產生使用您指定的 KMS 金鑰加密的唯一資料金鑰。Amazon EBS 會隨著磁碟區存放加密的資料金鑰。然後，當您將磁碟區連接到 Amazon EC2 執行個體時，Amazon EBS 會呼叫 AWS KMS 來解密資料金鑰。Amazon EBS 使用存放在 Hypervisor

記憶體的純文字資料金鑰來加密所有磁碟區的磁碟輸入/輸出。如需詳細資訊，請參閱《Amazon EC2 Linux 執行個體使用者指南》或《Amazon EC2 Windows 執行個體使用者指南》中的 EBS 加密運作方式。

Amazon EBS 加密內容

Amazon EBS 在其請求GenerateDataKeyWithoutPlaintext和解密請求中AWS KMS，使用具有名稱-值組的加密內容，用於識別請求中的磁碟區或快照。加密內容中的名稱會維持不變。

加密內容是一組金鑰/值對，其中包含任意非私密資料。在加密資料的請求中包含加密內容時，AWS KMS 會以密碼編譯方式將加密內容繫結至加密的資料。若要解密資料，您必須傳遞相同的加密內容。

對於使用 Amazon EBS CreateSnapshot操作建立的所有磁碟區和加密快照，Amazon EBS 會使用磁碟區 ID 作為加密內容值。在 CloudTrail 日誌項目的 `requestParameters` 欄位中，加密內容看起來如下：

```
"encryptionContext": {
  "aws:ebs:id": "vol-0cfb133e847d28be9"
}
```

對於透過 Amazon EC2 CopySnapshot作業建立的加密快照，Amazon EBS 會使用快照 ID 做為加密內容值。在 CloudTrail 日誌項目的 `requestParameters` 欄位中，加密內容看起來如下：

```
"encryptionContext": {
  "aws:ebs:id": "snap-069a655b568de654f"
}
```

偵測 Amazon EBS 失敗

為了建立加密的 EBS 磁碟區或將磁碟區連接到 EC2 執行個體，Amazon EBS 和 Amazon EC2 基礎設施必須能夠使用您為 EBS 磁碟區加密指定的 KMS 金鑰。當 KMS 金鑰無法使用，例如，當其金鑰狀態不是 `Enabled` 時，磁碟區建立或磁碟區連接會失敗。

在這種情況下，Amazon EBS 會將事件傳送至 Amazon EventBridge (之前稱為「CloudWatch 活動」)，以通知您有關失敗的資訊。在中 EventBridge，您可以建立規則來觸發自動動作以回應這些事件。如需詳細資訊，請參閱 Amazon EBS 適用於 Amazon EBS 的 Amazon CloudWatch 事件 (特別是下列各節)：

- 磁碟區連接或重新連接時的無效加密金鑰
- 建立磁碟區時的無效加密金鑰

若要修正這些問題，請確保您為 EBS 磁碟區加密指定的 KMS 金鑰已啟用。若要進行此操作，首先檢視 KMS 金鑰來確定其當前金鑰狀態 (AWS Management Console 中的 Status (狀態) 欄)。接著，查看下列其中一個連結的相關資訊：

- 如果 KMS 金鑰的金鑰狀態為已停用，則請啟用。
- 如果 KMS 金鑰的金鑰狀態為待匯入，則請匯入金鑰材料。
- 如果 KMS 金鑰的金鑰狀態為待刪除，則請取消金鑰刪除。

使用 AWS CloudFormation 建立加密的 Amazon EBS 磁碟區

您可以使用 AWS CloudFormation 來建立加密的 Amazon EBS 磁碟區。如需詳細資訊，請參閱《AWS CloudFormation 使用者指南》中的 AWS::EC2::Volume。

Amazon Elastic Transcoder 如何使用 AWS KMS

您可以使用 Amazon Elastic Transcoder 將存放在 Amazon S3 儲存貯體的媒體檔案轉換為消費者播放裝置所需的格式。輸入和輸出檔案都能加密和解密。以下章節討論如何將 AWS KMS 用於兩個程序。

主題

加密輸入檔案

在可以使用 Elastic Transcoder 之前，您必須先建立 Amazon S3 儲存貯體並將您的媒體檔案上傳到其中。您可以在上傳之前使用 AES 用戶端加密來加密檔案，或在上傳之後使用 Amazon S3 伺服器端加密來加密檔案。

如果您選擇使用 AES 的用戶端加密，您需負責在上傳到 Amazon S3 之前加密檔案，您也必須提供 Elastic Transcoder 對加密金鑰的存取權。您可透過使用對稱 AWS KMS AWS KMS key 保護用於加密媒體檔案的 AES 加密金鑰以達成此目的。

如果選擇伺服器端加密，您允許 Amazon S3 代表您執行所有檔案的加密和解密。您可以設定 Amazon S3 以使用三種不同類型加密金鑰的其中之一來保護用於加密檔案的唯一資料金鑰：

- Amazon S3 金鑰，Amazon S3 擁有和管理的加密金鑰。它不屬於 AWS 帳戶。
- 適用於 Amazon S3 的 AWS 受管金鑰，屬於您帳戶的 KMS 金鑰，但由 AWS 建立和管理
- 您使用 AWS KMS 建立的任何對稱客戶受管金鑰

> **Important**
>
> 對於用戶端和伺服器端加密，Elastic Transcoder 僅支援對稱 KMS 金鑰。您無法使用非對稱 KMS 金鑰來加密您的 Elastic Transcoder 檔案。如需判斷 KMS 金鑰是對稱還是不對稱的說明，請參閱 識別非對稱 KMS 金鑰。

您可以使用 Amazon S3 主控台或適當的 Amazon S3 API 來啟用加密和指定金鑰。如需 Amazon S3 如何執行加密的詳細資訊，請參閱《Amazon Simple Storage Service 使用者指南》中的使用伺服器端加密搭配 KMS 金鑰 (SSE-KMS) 來保護資料。

當您使用帳戶中適用於 Amazon S3 的 AWS 受管金鑰 或客戶受管金鑰保護您的輸入檔案時，Amazon S3 和 AWS KMS 的互動如下：

1. Amazon S3 請求純文字資料金鑰和加密金鑰和在指定的 KMS 金鑰下加密的資料金鑰複本。
2. AWS KMS 會建立資料金鑰、使用指定的 KMS 金鑰為其加密，然後同時傳送純文字資料金鑰和加密的資料金鑰給 Amazon S3。
3. Amazon S3 使用純文字資料金鑰來加密媒體檔案，然後將檔案儲存到指定的 Amazon S3 儲存貯體。
4. Amazon S3 將加密的資料金鑰和加密的媒體檔案一起存放。

解密輸入檔案

如果您選擇 Amazon S3 伺服器端加密來加密輸入檔案，Elastic Transcoder 不會解密檔案。反之，Elastic Transcoder 依賴 Amazon S3 根據您建立任務時所指定的設定和管道來執行解密。

可以使用下列設定組合。

加密模式	AWS KMS 金鑰	意義
S3	預設	Amazon S3 建立和管理用來加密和解密媒體檔案的金鑰。此程序對使用者來說是不透明的。
S3-AWS-KMS	預設	Amazon S3 使用由您帳戶中適用於 Amazon S3 之預設 AWS 受管金鑰 所加密的資料金鑰來加密媒體檔案。
S3-AWS-KMS	自訂 (使用 ARN)	Amazon S3 使用由指定之客戶受管金鑰所加密的資料金鑰來加密媒體檔案。

指定 S3-AWS-KMS 時，Amazon S3 和 AWS KMS 使用以下方式搭配運作來執行解密。

1. Amazon S3 傳送加密的資料金鑰給 AWS KMS。
2. AWS KMS 使用適當的 KMS 金鑰解密資料金鑰，然後將純文字資料金鑰傳送回 Amazon S3。
3. Amazon S3 使用純文字資料金鑰來解密加密文字。

如果您選擇使用 AES 金鑰的用戶端加密，Elastic Transcoder 會從 Amazon S3 儲存貯體擷取加密的檔案，然後將其解密。Elastic Transcoder 使用您建立管道時指定的 KMS 金鑰來解密 AES 金鑰，然後使用 AES 金鑰來解密媒體檔案。

加密輸出檔案

Elastic Transcoder 根據當您建立任務和管道時所指定的加密設定來加密輸出檔案。以下是可用的選項。

加密模式	AWS KMS 金鑰	意義
S3	預設	Amazon S3 建立和管理用來加密輸出檔案的金鑰。

加密模式	AWS KMS 金鑰	意義
S3-AWS-KMS	預設	Amazon S3 使用您帳戶中由 AWS KMS 建立並由適用於 Amazon S3 之 AWS 受管金鑰加密的資料金鑰。
S3-AWS-KMS	自訂 (使用 ARN)	Amazon S3 使用由 ARN 指定之客戶受管金鑰所加密的資料金鑰來加密媒體檔案。
AES-	預設	Elastic Transcoder 使用您帳戶中適用於 Amazon S3 的 AWS 受管金鑰 來解密您提供的指定 AES 金鑰，並使用該金鑰來加密輸出檔案。
AES-	自訂 (使用 ARN)	Elastic Transcoder 使用由 ARN 指定之客戶受管金鑰來解密您提供的指定 AES 金鑰，並使用該金鑰來加密輸出檔案。

當您指定使用帳戶中適用於 Amazon S3 的 AWS 受管金鑰 或客戶受管金鑰來加密輸出檔案時，Amazon S3 和 AWS KMS 的互動如下：

1. Amazon S3 請求純文字資料金鑰和加密金鑰和在指定的 KMS 金鑰下加密的資料金鑰複本。
2. AWS KMS 會建立資料金鑰、使用指定的 KMS 金鑰為其加密，然後同時傳送純文字資料金鑰和加密的資料金鑰給 Amazon S3。
3. Amazon S3 會使用資料金鑰來加密媒體，然後將其存放在指定的 Amazon S3 儲存貯體中。
4. Amazon S3 將加密的資料金鑰和加密的媒體檔案一起存放。

當您指定使用您所提供的 AES 金鑰來加密輸出檔案時，AES 金鑰必須使用 AWS KMS 中的 KMS 金鑰進行加密。Elastic Transcoder (AWS KMS)，您會以下列方式進行互動：

1. 您可以透過呼叫 AWS KMS API 中的 Encrypt 操作加密 AES 金鑰。AWS KMS 會使用指定的 KMS 金鑰來加密金鑰。您在建立管道時指定要使用的 KMS 金鑰。

2. 您在建立 Elastic Transcoder 任務時指定包含加密 AES 金鑰的檔案。
3. Elastic Transcoder 透過呼叫 AWS KMS API 中的 Decrypt 操作，將加密的金鑰作為加密文字傳遞。
4. Elastic Transcoder 使用解密的 AES 金鑰來加密輸出媒體檔案，然後從記憶體刪除解密的 AES 金鑰。只有您最初在任務中定義的加密副本會儲存到磁碟。
5. 您可以下載加密的輸出檔案，並使用您定義的原始 AES 金鑰在本機解密。

> **Important**
>
> AWS 永遠不會儲存您的私有加密金鑰。因此，您一定要安全地管理您的金鑰。如果遺失這些金鑰，就無法解密資料。

HLS 內容保護

HTTP 即時串流 (HLS) 是一種調適型串流通訊協定。Elastic Transcoder 支援 HLS，方法是將您的輸入檔案分成較小的個別檔案 (稱為媒體分段)。一組相對應的個別媒體分段包含以不同位元率編碼的相同材料，讓播放器能夠選擇最適合可用頻寬的串流。Elastic Transcoder 也會建立播放清單，其中包含可串流播放的各個分段的中繼資料。

當您啟用 HLS 內容保護時，將會使用 128 位元 AES 加密金鑰來加密每個媒體分段。檢視內容時，在播放過程中，播放器會下載金鑰並解密媒體分段。

使用兩種金鑰類型：KMS 金鑰和資料金鑰。您必須建立 KMS 金鑰以用於加密和解密資料金鑰。Elastic Transcoder 使用資料金鑰來加密和解密媒體分段。資料金鑰必須是 AES-128。相同內容的所有變化和分段都使用相同的資料金鑰來加密。您可以提供資料金鑰或讓 Elastic Transcoder 為您建立。

KMS 金鑰可用於在以下幾點加密資料金鑰：

- 如果您提供自己的資料金鑰，您必須先將其加密再傳送到 Elastic Transcoder。
- 如果您請求 Elastic Transcoder 產生資料金鑰，Elastic Transder 會為您加密資料金鑰。

KMS 金鑰可用於在以下幾點解密資料金鑰：

- 需要使用資料金鑰來加密輸出檔案或解密輸入檔案時，Elastic Transcoder 會解密您提供的資料金鑰。
- 您解密由 Elastic Transcoder 產生的資料金鑰，並使用它來解密輸出檔案。

如需詳細資訊，請參閱《Amazon Elastic Transcoder 開發人員指南》中的 HLS 內容保護。

Elastic Transcoder 加密內容

加密內容是一組金鑰/值對，其中包含任意非私密資料。在加密資料的請求中包含加密內容時，AWS KMS 會以密碼編譯方式將加密內容繫結至加密的資料。若要解密資料，您必須傳遞相同的加密內容。

Elastic Transcoder 在所有 AWS KMS API 請求中使用相同的加密內容來產生資料金鑰、加密及解密。

```
"service" : "elastictranscoder.amazonaws.com"
```

加密內容會寫入 CloudTrail 記錄檔，以協助您瞭解指定 AWS KMS KMS 金鑰的使用方式。在 CloudTrail 記錄檔的`requestParameters`欄位中，加密內容看起來類似下列內容：

```
"encryptionContext": {
  "service" : "elastictranscoder.amazonaws.com"
}
```

如需如何將 Elastic Transcoder 任務設定為使用其中一個支援加密選項的詳細資訊，請參閱《Amazon Elastic Transcoder 開發人員指南》中的資料加密選項。

Amazon EMR 如何使用 AWS KMS

當使用 Amazon EMR 叢集時，您可以在將資料儲存至持久性儲存位置之前，設定叢集加密靜態資料。您可以加密 EMR 檔案系統 (EMRFS)、叢集節點之儲存磁碟區，或兩者上的靜態資料。若要靜態加密資料，您可以使用 AWS KMS key。下列主題說明 Amazon EMR 叢集如何使用 KMS 金鑰來加密靜態資料。

> **Important**
>
> Amazon EMR 僅支援對稱 KMS 金鑰。您無法使用非對稱 KMS 金鑰來加密 Amazon EMR 叢集中的靜態資料。如需判斷 KMS 金鑰是對稱還是不對稱的說明，請參閱 識別非對稱 KMS 金鑰。

Amazon EMR 叢集也會加密傳輸中的資料，這表示叢集會在透過網路傳送資料之前進行加密。您不能使用 KMS 金鑰加密傳輸中的資料。如需詳細資訊，請參閱《Amazon EMR 管理指南》中的傳輸資料加密。

如需 Amazon EMR 中所有可用加密選項的詳細資訊，請參閱《Amazon EMR 管理指南》中的加密選項。

主題

- 加密 EMR 檔案系統 (EMRFS) 上的資料
- 加密叢集節點之儲存磁碟區上的資料
- 加密內容

加密 EMR 檔案系統 (EMRFS) 上的資料

Amazon EMR 叢集使用兩個分散式檔案系統：

- Hadoop 分散式檔案系統 (HDFS)。HDFS 加密不使用 AWS KMS 中的 KMS 金鑰。
- EMR 檔案系統 (EMRFS)。EMRFS 是 HDFS 實作，可讓 Amazon EMR 叢集將資料存放在 Amazon Simple Storage Service (Amazon S3) 中。EMRFS 支援四種加密選項，其中兩種使用 AWS KMS 中的 KMS 金鑰。如需所有四個 EMRFS 加密選項的詳細資訊，請參閱《Amazon EMR 管理指南》中的加密選項。

使用 KMS 金鑰的兩個 EMRFS 加密選項使用 Amazon S3 提供的以下加密功能：

- 使用伺服器端加密搭配 AWS Key Management Service (SSE-KMS) 來保護資料 Amazon EMR 叢集會將資料傳送至 Simple Storage Service (Amazon S3)。Simple Storage Service (Amazon S3) 使用 KMS 金鑰來加密資料，然後將其儲存至 S3 儲存貯體。如需此操作如何進行的詳細資訊，請參閱使用 SSE-KMS 加密 EMRFS 上資料的程序。
- 使用用戶端加密保護資料 (CSE-KMS)。Amazon EMR 中的資料在 AWS KMS key 下加密，然後將其傳送至 Simple Storage Service (Amazon S3) 進行儲存。如需此操作如何進行的詳細資訊，請參閱使用 CSE-KMS 加密 EMRFS 上資料的程序。

當您設定 Amazon EMR 叢集在 EMRFS 上使用 KMS 金鑰來加密資料時，您可以選擇您想要 Simple Storage Service (Amazon S3) 或 Amazon EMR 叢集使用的 KMS 金鑰。使用 SSE-KMS，您可以選擇具有別名 aws/s3 的 Amazon S3 AWS 受管金鑰，或您建立的對稱客戶受管金鑰。使用用戶端加密時，您必須選擇您建立的對稱客戶受管金鑰。選擇客戶受管金鑰時，您必須確保 Amazon EMR 叢集具有使用 KMS 金鑰的許可。如需詳細資訊，請參閱《Amazon EMR 管理指南》中的將 AWS KMS keys 用於加密。

對於伺服器端和用戶端加密，您選擇的 KMS 金鑰是信封加密工作流程中的根金鑰。資料使用唯一的資料金鑰加密，該金鑰在 AWS KMS 中的 KMS 金鑰下加密。加密的資料及其資料金鑰的加密副本會以單一加密物件一起存放在 S3 儲存貯體中。如需其如何運作的詳細資訊，請參閱下列主題。

主題

- 使用 SSE-KMS 加密 EMRFS 上資料的程序
- 使用 CSE-KMS 加密 EMRFS 上資料的程序

使用 SSE-KMS 加密 EMRFS 上資料的程序

設定 Amazon EMR 叢集使用 SSE-KMS 時，加密程序的運作方式如下：

1. 叢集將資料傳送至 Amazon S3 以便儲存在 S3 儲存貯體中。
2. Amazon S3 會將GenerateDataKey請求傳送到AWS KMS，並指定您在將叢集設定為使用 SSE-KMS 時選擇的 KMS 金鑰金鑰的金鑰識別碼。請求包含加密內容；如需詳細資訊，請參閱加密內容。
3. AWS KMS 會產生唯一的資料加密金鑰 (資料金鑰)，然後傳送此資料金鑰的兩個複本給 Amazon S3。一個複本是未加密的 (純文字)，另一個複本則透過 KMS 金鑰加密。
4. Amazon S3 使用純文字資料金鑰來加密它在步驟 1 收到的資料，然後在使用後盡快從記憶體移除純文字資料金鑰。
5. Amazon S3 將加密的資料及資料金鑰的加密複本以單一加密物件一起存放在 S3 儲存貯體中。

解密程序的運作方式如下：

1. 叢集向 S3 儲存貯體請求加密的資料物件。
2. Amazon S3 會從 S3 物件中擷取加密的資料金鑰，然後將加密的資料金鑰傳送給具有 Decrypt 請求的 AWS KMS。此請求包含加密內容。
3. AWS KMS 使用用來加密的相同 KMS 金鑰來解密加密的資料金鑰，然後將解密後的 (純文字) 資料金鑰傳送給 Amazon S3。
4. Amazon S3 使用純文字資料金鑰來解密加密的資料，然後在使用後盡快從記憶體移除純文字資料金鑰。
5. Amazon S3 將解密的資料傳送給叢集。

使用 CSE-KMS 加密 EMRFS 上資料的程序

設定 Amazon EMR 叢集使用 CSE-KMS 時，加密程序的運作方式如下：

1. 準備好將資料存放在 Amazon S3 時，叢集會向其傳送GenerateDataKey請求AWS KMS，並指定您在將叢集設定為使用 CSE-KMS 時選擇的 KMS 金鑰金鑰的金鑰識別碼。請求包含加密內容；如需詳細資訊，請參閱加密內容。
2. AWS KMS 會產生唯一的資料加密金鑰 (資料金鑰)，然後傳送此資料金鑰的兩個副本給叢集。一個複本是未加密的 (純文字)，另一個複本則透過 KMS 金鑰加密。
3. 叢集使用純文字資料金鑰來加密資料，然後在使用後盡快從記憶體移除純文字資料金鑰。
4. 叢集將加密的資料及資料金鑰的加密副本合併為單一加密物件。
5. 叢集將加密的物件傳送到 Amazon S3 以便儲存。

解密程序的運作方式如下：

1. 叢集向 S3 儲存貯體請求加密的資料物件。
2. Amazon S3 傳送加密的物件給叢集。
3. 叢集會從加密的物件中擷取加密的資料金鑰，然後將加密的資料金鑰傳送給具有 Decrypt 請求的 AWS KMS。此請求包含加密內容。
4. AWS KMS 使用用來加密的相同 KMS 金鑰來解密加密的資料金鑰，然後將解密後的 (純文字) 資料金鑰傳送給叢集。
5. 叢集使用純文字資料金鑰來解密加密的資料，然後在使用後盡快從記憶體移除純文字資料金鑰。

加密叢集節點之儲存磁碟區上的資料

Amazon EMR 叢集是 Amazon Elastic Compute Cloud (Amazon EC2) 執行個體的集合。叢集中的每個執行個體稱為叢集節點或節點。每個節點可以有兩種類型的儲存磁碟區：執行個體存放磁碟區和 Amazon Elastic Block Store (Amazon EBS) 磁碟區。您可以設定叢集為使用 Linux 統一金鑰設定 (LUKS) 來加密節點上的這兩種儲存磁碟區類型 (但不包含每個節點的開機磁碟區)。這稱為本機磁碟加密。

當您啟用叢集的本機磁碟加密時，可以選擇使用 AWS KMS 中的 KMS 金鑰來加密 LUKS 金鑰。您必須選擇您建立的客戶受管金鑰，您無法使用 AWS 受管金鑰。如果選擇客戶受管金鑰，則您必須確保 Amazon EMR 叢集具有使用 KMS 金鑰的許可。如需詳細資訊，請參閱《Amazon EMR 管理指南》中的將 AWS KMS keys 用於加密。

當您啟用使用 KMS 金鑰的本機磁碟加密，加密程序的運作方式如下：

1. 每個叢集節點啟動時，會向其傳送GenerateDataKey要求AWS KMS，指定您在為叢集啟用本機磁碟加密時選擇的 KMS 金鑰金鑰的金鑰識別碼。
2. AWS KMS 會產生唯一的資料加密金鑰 (資料金鑰)，然後傳送此資料金鑰的兩個副本給節點。一個複本是未加密的 (純文字)，另一個複本則透過 KMS 金鑰加密。
3. 節點使用純文字資料金鑰的 base64 編碼版本做為保護 LUKS 金鑰的密碼。節點將資料金鑰的加密副本儲存在開機磁碟區。
4. 如果節點重新啟動，重新啟動的節點會傳送加密的資料金鑰給具有 Decrypt 請求的 AWS KMS。
5. AWS KMS 使用用來加密的相同 KMS 金鑰來解密加密的資料金鑰，然後將解密後的 (純文字) 資料金鑰傳送給節點。
6. 節點使用純文字資料金鑰的 base64 編碼版本做為解鎖 LUKS 金鑰的密碼。

加密內容

與 AWS KMS 整合的每個 AWS 服務可以在服務使用 AWS KMS 產生資料金鑰或加密或解密資料時指定加密內容。加密內容是 AWS KMS 用來檢查資料完整性的額外驗證資訊。當服務指定加密操作的加密內容，它必須為相對應的解密操作指定相同的加密內容，否則解密將失敗。加密內容也會寫入 AWS CloudTrail 日誌檔案，這可協助您了解為什麼使用指定的 KMS 金鑰。

以下章節說明使用 KMS 金鑰之每個 Amazon EMR 加密案例所用的加密內容。

使用 SSE-KMS 進行 EMRFS 加密的加密內容

透過 SSE-KMS，Amazon EMR 叢集會傳送資料至 Amazon S3，接著 Amazon S3 使用 KMS 金鑰加密資料，然後將其儲存到 S3 儲存貯體。在這種情況下，Amazon S3 會使用 S3 物件的 Amazon 資源名稱 (ARN) 做為每個物件的加密內容，GenerateDataKey並將其傳送到AWS KMS的目標解密請求。以下範例顯示 Amazon S3 所用加密內容的 JSON 顯示方式。

```
{ "aws:s3:arn" : "arn:aws:s3:::S3_bucket_name/S3_object_key" }
```

使用 CSE-KMS 進行 EMRFS 加密的加密內容

透過 CSE-KMS，Amazon EMR 叢集使用 KMS 金鑰加密資料，然後將其傳送到 Amazon S3 儲存貯體儲存。在這種情況下，叢集會使用 KMS 金鑰的 Amazon 資源名稱 (ARN) 做為加密內容，每個GenerateDataKey叢集都會使用其傳送的「解密」請求。AWS KMS以下範例顯示叢集所用加密內容的 JSON 顯示方式。

```
{ "kms_cmk_id" : "arn:aws:kms:us-
east-2:111122223333:key/0987ab65-43cd-21ef-09ab-87654321cdef" }
```

使用 LUKS 進行本機磁碟加密的加密內容

當 Amazon EMR 叢集使用本機磁碟加密搭配 LUKS 時，叢集節點不會使用它們傳送的GenerateDataKey和解密請求指定加密內容。AWS KMS

AWS Nitro Enclaves 如何使用 AWS KMS

AWS KMS支援 AWS Nitro Enclaves 的密碼編譯證明。支援 AWS Nitro Enclaves 的應用程式會使用 enclave 的已簽署證明文件來呼叫下列 AWS KMS 密碼編譯操作。這些 AWS KMS API 會驗證證明文字件來自 Nitro enclave。然後，在回應時，這些 API 不會傳回純文字資料，而是利用證明文件的公有金鑰來加密純文字，並傳回僅能透過 enclave 的相應私有金鑰進行解密的密文。

- 解密
- GenerateDataKey
- GenerateDataKeyPair
- GenerateRandom

下表顯示對 Nitro enclave 請求的回應與每個 API 操作的標準回應有何不同。

AWS KMS 操作	標準回應	對 AWS Nitro Enclaves 的回應
`Decrypt`	傳回純文字資料	傳回純文字資料 (經來自證明文件的公有金鑰加密)
`GenerateDataKey`	傳回資料金鑰的純文字複本 (也會傳回 KMS 金鑰加密的資料金鑰複本)	傳回資料金鑰複本 (經來自證明文件的公有金鑰加密) (也會傳回 KMS 金鑰加密的資料金鑰複本)
`GenerateDataKeyPair`	傳回私有金鑰的純文字副本 (也會傳回公有金鑰以及由 KMS 金鑰加密的私有金鑰副本)	傳回私有金鑰複本 (經來自證明文件的公有金鑰加密)

AWS KMS 操作	標準回應	對 AWS Nitro Enclaves 的回應
		(也會傳回公有金鑰以及由 KMS 金鑰加密的私有金鑰副本)
`GenerateRandom`	傳回隨機位元組字串	傳回隨機位元組字串 (經來自證明文件的公有金鑰加密)

AWS KMS 支援政策條件金鑰，可讓您根據證明文件內容利用 AWS KMS 來允許或拒絕 enclave 操作。您還可在 AWS CloudTrail 日誌監控針對 Nitro enclave 向 AWS KMS 提出的請求。

主題

- 如何呼叫 Nitro Enclaves 的 AWS KMS API
- AWS Nitro Enclaves 的 AWS KMS 條件索引鍵
- 對 Nitro Enclaves 的監空請求

如何呼叫 Nitro Enclaves 的 AWS KMS API

若要呼叫 Nitro Enclave 的 AWS KMS API，請於請求時採用 `Recipient` 參數為 enclave 提供已簽署的證明文件，以及要與 enclave 公有金鑰搭配使用的加密演算法。當請求包含已簽署證明文件的 `Recipient` 參數時，回應會包含 `CiphertextForRecipient` 欄位，以及由公有金鑰加密的密文。純文字欄位為空值或空白。

`Recipient` 參數必須從 AWS Nitro enclave 指定已簽署的證明文件。AWS KMS 依賴 enclave 證明文件的數位簽章來證明請求的公有金鑰來自有效 enclave。您無法提供自己的憑證來數位簽署證明文件。

若要指定 `Recipient` 參數，請採用 AWS Nitro Enclaves SDK 或任何 AWS SDK。AWS Nitro Enclaves SDK 僅在 Nitro enclave 受支援，會自動新增 `Recipient` 參數及其值至每個 AWS KMS 請求。若要在 AWS SDK 提出 Nitro Enclaves 請求，您必須指定 `Recipient` 參數及其值。於 2023 年 3 月針對 AWS SDK 推出 Nitro enclave 密碼編譯證明支援。

AWS KMS 支援政策條件金鑰，可讓您根據證明文件內容利用 AWS KMS 來允許或拒絕 enclave 操作。您還可在 AWS CloudTrail 日誌監控針對 Nitro enclave 向 AWS KMS 提出的請求。

如需有關`Recipient`參數和 AWS `CiphertextForRecipient` 回應欄位的詳細資訊，請參閱 AWS Key Management ServiceAPI 參考GenerateDataKeyGenerateDataKeyPair、AWSNitro Enclaves 開

發套件或任何開發套件中的解密、、和GenerateRandom主題。AWS如需設定加密之資料和資料金鑰的相關資訊，請參閱將 AWS KMS 與密碼編譯證明搭配使用。

AWS Nitro Enclaves 的 AWS KMS 條件索引鍵

您可針對控制 AWS KMS 資源存取的金鑰政策與 IAM 政策指定條件金鑰。包含條件金鑰的政策陳述式僅在符合條件時才有效。

AWS KMS提供條件金鑰，GenerateDataKey可根據請求中已簽署的驗證文件內容限制解密GenerateDataKeyPair、、和GenerateRandom作業的權限。這些條件金鑰僅在 AWS KMS 操作請求包含 `Recipient` 參數，以及來自 AWS Nitro enclave 的有效證明文件時才有效。若要指定 `Recipient` 參數，請採用 AWS Nitro Enclaves SDK 或任何 AWS SDK。

Enclave 特定的 AWS KMS 條件金鑰在金鑰政策陳述式與 IAM 政策陳述式有效，即使其未出現在 IAM 主控台或 IAM 服務授權參考。

公RecipientAttestation里 ImageSha

AWS KMS 條件金鑰	條件類型	值類型	API 操作	政策類型

`kms:RecipientAttestation:ImageSha384`	字串	單一值	`Decrypt`
			`GenerateDataKey`
			`GenerateDataKeyPair`
			`GenerateRandom`
				金鑰政策和 IAM 政策

當請求中已簽署證明文件的影像摘要符合條件金鑰的值時，`kms:RecipientAttestation:ImageSha384` 條件金鑰可利用 KMS 金鑰控制對以下各項的存取：`Decrypt`、`GenerateDataKey`、`GenerateDataKeyPair` 與 `GenerateRandom`。`ImageSha384` 值對應證明文件的 PCR0。僅當請求的 `Recipient` 參數指定 AWS Nitro Enclave 的已簽署證明文件時，此條件金鑰才有效。

此值也包含在要求 Nitro 飛地AWS KMS的CloudTrail事件中。

> **Note**
>
> 此條件金鑰在金鑰政策陳述式和 IAM 政策陳述式中有效，即使它未出現在 IAM 主控台或 IAM 服務授權參考。

例如，下列金鑰原則陳述式允許`data-processing`角色使用 KMS 金鑰進行解密GenerateDataKey、GenerateDataKeyPair、和GenerateRandom作業。僅當請求中證明文件的影像摘要值 (PCR0) 符合條件的影像摘要值時，`kms:RecipientAttestation:ImageSha384` 條件金鑰才會允許操作。僅當請求的 `Recipient` 參數指定 AWS Nitro Enclave 的已簽署證明文件時，此條件金鑰才有效。

如請求不含來自 AWS Nitro enclave 的有效證明文件，則會因未滿足此條件而拒絕許可。

```
{
  "Sid" : "Enable enclave data processing",
  "Effect" : "Allow",
  "Principal" : {
    "AWS" : "arn:aws:iam::111122223333:role/data-processing"
  },
  "Action": [
    "kms:Decrypt",
    "kms:GenerateDataKey",
    "kms:GenerateDataKeyPair",
    "kms:GenerateRandom"
  ],
  "Resource" : "*",
  "Condition": {
    "StringEqualsIgnoreCase": {
      "kms:RecipientAttestation:ImageSha384":
 "9fedcba8abcdef7abcdef6abcdef5abcdef4abcdef3abcdef2abcdef1abcdef1abcdef0abcdef1abcdef2abcdef3a
    }
  }
}
```

公里:: 聚氯乙烯 RecipientAttestation <PCR_ID>

AWS KMS 條件金鑰	條件類型	值類型	API 操作	政策類型
kms:RecipientAttestation:PCR<PCR_ID>	字串	單一值	Decrypt GenerateDataKey GenerateDataKeyPair GenerateRandom	金鑰政策和 IAM 政策

僅當請求中已簽署證明文件的平台組態註冊 (PCR) 符合條件金鑰的 PCR 時，kms:RecipientAttestation:PCR<PCR_ID> 條件金鑰才會利用 KMS 金鑰控制對以下各項的存取：Decrypt、GenerateDataKey、GenerateDataKeyPair 以及 GenerateRandom。僅當請求的 Recipient 參數指定來自 AWS Nitro Enclave 的已簽署證明文件時，此條件金鑰才有效。

此值也包含在代表AWS KMS對 Nitro 飛地的請求的CloudTrail事件中。

> **Note**
>
> 此條件金鑰在金鑰政策陳述式和 IAM 政策陳述式中有效，即使它未出現在 IAM 主控台或 IAM 服務授權參考。

若要指定 PCR 值，請使用以下格式。將 PCR ID 串連到條件索引鍵名稱。PCR 值必須是最多 96 個位元組的小寫十六進位字串。

```
"kms:RecipientAttestation:PCRPCR_ID": "PCR_value"
```

例如，下列條件金鑰指定 PCR1 的特定值，對應於用於 enclave 與啟動程序處理的核心雜湊。

```
kms:RecipientAttestation:PCR1:
 "0x1abcdef2abcdef3abcdef4abcdef5abcdef6abcdef7abcdef8abcdef9abcdef8abcdef7abcdef6abcdef5abcdef
```

下列範例金鑰政策陳述式允許 data-processing 角色利用 KMS 金鑰進行 Decrypt 操作。

僅當請求中已簽署證明文件中的 PCR1 值符合條件中的 kms:RecipientAttestation:PCR1 值時，本陳述式中的 kms:RecipientAttestation:PCR 條件索引鍵才會允許操作。使用 StringEqualsIgnoreCase 政策運算子要求 PCR 值不區分大小寫的比較。

如請求不含證明文件，則會因未滿足此條件而拒絕許可。

```
{
  "Sid" : "Enable enclave data processing",
  "Effect" : "Allow",
  "Principal" : {
    "AWS" : "arn:aws:iam::111122223333:role/data-processing"
  },
  "Action": "kms:Decrypt",
  "Resource" : "*",
  "Condition": {
    "StringEqualsIgnoreCase": {
      "kms:RecipientAttestation:PCR1":
 "0x1de4f2dcf774f6e3b679f62e5f120065b2e408dcea327bd1c9dddaea6664e7af79355814748447674530 82c6f15
    }
  }
}
```

對 Nitro Enclaves 的監空請求

您可以使用AWS CloudTrail記錄來監視 AWS Nitro 飛地的解密GenerateDataKeyGenerateDataKeyPair、、和GenerateRandom作業。在這些日誌項目，additionalEventData 欄位具 recipient 欄位，其中包含請求證明文件的模組 ID (attestationDocumentModuleId)、影像摘要 (attestationDocumentEnclaveImageDigest)，以及平台組態註冊 (PCR)。僅當請求的 Recipient 參數指定來自 AWS Nitro Enclave 的已簽署證明文件時，才會包含這些欄位。

模組 ID 是 Nitro enclave 的 enclave ID。影像摘要是 enclave 影像的 SHA384 雜湊。您可在金鑰政策與 IAM 政策的條件運用影像摘要及 PCR 值。如需 PCR 的相關資訊，請參閱《AWSNitro Enclaves使用者指南》的Where to get an enclave's measurements。

本節顯示每個受支援的 Nitro enclave 要求的範例 CloudTrail 記錄項目。AWS KMS

Decrypt (針對 enclave)

以下 AWS CloudTrail 日誌項目範例顯示 AWS Nitro enclave 的 Decrypt 操作。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2020-07-27T22:58:24Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "encryptionAlgorithm": "SYMMETRIC_DEFAULT",
        "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": null,
    "additionalEventData": {
        "recipient": {
            "attestationDocumentModuleId": "i-123456789abcde123-enc123456789abcde12",
            "attestationDocumentEnclaveImageDigest": "<AttestationDocument.PCR0>",
            "attestationDocumentEnclavePCR1": "<AttestationDocument.PCR1>",
            "attestationDocumentEnclavePCR2": "<AttestationDocument.PCR2>",
            "attestationDocumentEnclavePCR3": "<AttestationDocument.PCR3>",
            "attestationDocumentEnclavePCR4": "<AttestationDocument.PCR4>",
            "attestationDocumentEnclavePCR8": "<AttestationDocument.PCR8>"
        }
    },
    "requestID": "b4a65126-30d5-4b28-98b9-9153da559963",
    "eventID": "e5a2f202-ba1a-467c-b4ba-f729d45ae521",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
```

```
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

GenerateDataKey (適用於飛地)

下列範例顯示AWS硝基飛地之GenerateDataKey作業的AWS CloudTrail記錄項目。

```
{
    "eventVersion": "1.02",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2014-11-04T00:52:40Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKey",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab",
        "numberOfBytes": 32
    },
    "responseElements": null,
    "additionalEventData": {
        "recipient": {
            "attestationDocumentModuleId": "i-123456789abcde123-enc123456789abcde12",
            "attestationDocumentEnclaveImageDigest": "<AttestationDocument.PCR0>",
            "attestationDocumentEnclavePCR1": "<AttestationDocument.PCR1>",
            "attestationDocumentEnclavePCR2": "<AttestationDocument.PCR2>",
            "attestationDocumentEnclavePCR3": "<AttestationDocument.PCR3>",
            "attestationDocumentEnclavePCR4": "<AttestationDocument.PCR4>",
            "attestationDocumentEnclavePCR8": "<AttestationDocument.PCR8>"
        }
    },
    "requestID": "e0eb83e3-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "a9dea4f9-8395-46c0-942c-f509c02c2b71",
```

```
    "readOnly": true,
    "resources": [{
        "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
        "accountId": "111122223333"
    }],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

GenerateDataKeyPair (適用於飛地)

下列範例顯示AWS硝基飛地之GenerateDataKeyPair作業的AWS CloudTrail記錄項目。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
            "type": "IAMUser",
            "principalId": "EX_PRINCIPAL_ID",
            "arn": "arn:aws:iam::111122223333:user/Alice",
            "accountId": "111122223333",
            "accessKeyId": "EXAMPLE_KEY_ID",
            "userName": "Alice"
    },
    "eventTime": "2020-07-27T18:57:57Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateDataKeyPair",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": {
        "keyPairSpec": "RSA_3072",
        "encryptionContext": {
            "Project": "Alpha"
        },
        "keyId": "1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    "responseElements": null,
    "additionalEventData": {
        "recipient": {
            "attestationDocumentModuleId": "i-123456789abcde123-enc123456789abcde12",
            "attestationDocumentEnclaveImageDigest": "<AttestationDocument.PCR0>",
            "attestationDocumentEnclavePCR1": "<AttestationDocument.PCR1>",
```

```
            "attestationDocumentEnclavePCR2": "<AttestationDocument.PCR2>",
            "attestationDocumentEnclavePCR3": "<AttestationDocument.PCR3>",
            "attestationDocumentEnclavePCR4": "<AttestationDocument.PCR4>",
            "attestationDocumentEnclavePCR8": "<AttestationDocument.PCR8>"
        }
    },
    "requestID": "52fb127b-0fe5-42bb-8e5e-f560febde6b0",
    "eventID": "9b6bd6d2-529d-4890-a949-593b13800ad7",
    "readOnly": true,
    "resources": [
        {
            "accountId": "111122223333",
            "type": "AWS::KMS::Key",
            "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

GenerateRandom (適用於飛地)

下列範例顯示AWS硝基飛地之GenerateRandom作業的AWS CloudTrail記錄項目。

```
{
    "eventVersion": "1.02",
    "userIdentity": {
        "type": "IAMUser",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::111122223333:user/Alice",
        "accountId": "111122223333",
        "accessKeyId": "EXAMPLE_KEY_ID",
        "userName": "Alice"
    },
    "eventTime": "2014-11-04T00:52:37Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "GenerateRandom",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "AWS Internal",
    "requestParameters": null,
    "responseElements": null,
```

```
    "additionalEventData": {
        "recipient": {
            "attestationDocumentModuleId": "i-123456789abcde123-enc123456789abcde12",
            "attestationDocumentEnclaveImageDigest": "<AttestationDocument.PCR0>",
            "attestationDocumentEnclavePCR1": "<AttestationDocument.PCR1>",
            "attestationDocumentEnclavePCR2": "<AttestationDocument.PCR2>",
            "attestationDocumentEnclavePCR3": "<AttestationDocument.PCR3>",
            "attestationDocumentEnclavePCR4": "<AttestationDocument.PCR4>",
            "attestationDocumentEnclavePCR8": "<AttestationDocument.PCR8>"
        }
    },
    "requestID": "df1e3de6-63bc-11e4-bc2b-4198b6150d5c",
    "eventID": "239cb9f7-ae05-4c94-9221-6ea30eef0442",
    "readOnly": true,
    "resources": [],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333"
}
```

Amazon Redshift 如何使用 AWS KMS

本主題討論 Amazon Redshift 如何使用 AWS KMS 來加密資料。

主題

- Amazon Redshift 加密
- 加密內容

Amazon Redshift 加密

Amazon Redshift 資料倉儲是稱為節點的運算資源的集合，組織成稱為叢集的群組。每個叢集皆執行 Amazon Redshift 引擎並包含一或多個資料庫。

Amazon Redshift 使用四個階層的金鑰架構來加密。此架構包含資料加密金鑰、資料庫金鑰、叢集金鑰和根金鑰。您可以使用 AWS KMS key 做為根金鑰。

資料加密金鑰會加密叢集中的資料區塊。每個資料區塊都會獲指派一個隨機產生的 AES-256 金鑰。這些金鑰使用叢集的資料庫金鑰來加密。

資料庫金鑰會加密叢集中的資料加密金鑰。資料庫金鑰是隨機產生的 AES-256 金鑰。它會存放在與 Amazon Redshift 叢集不同之網路的磁碟上，並透過安全通道傳送給叢集。

叢集金鑰會加密 Amazon Redshift 叢集的資料庫金鑰。您可以使用 AWS KMS、AWS CloudHSM 或外部硬體安全模組 (HSM) 來管理叢集金鑰。請參閱 Amazon Redshift Database 加密文件以取得更多詳細資訊。

您可以在 Amazon Redshift 主控台中勾選適當的方塊來請求加密。您可以從加密方塊下方的清單中選擇一個項目，指定要客戶受管金鑰。如果您未指定客戶受管金鑰，Amazon Redshift 會在您的帳戶下使用 Amazon Redshift 的 AWS 受管金鑰。

> **Important**
>
> Amazon Redshift 只支援對稱加密 KMS 金鑰。您無法在 Amazon Redshift 加密工作流程中使用非對稱 KMS 金鑰。如需判斷 KMS 金鑰是對稱還是不對稱的說明，請參閱 識別非對稱 KMS 金鑰。

加密內容

與 AWS KMS 整合的每個服務在請求資料金鑰、加密及解密時，會指定加密內容。加密內容是 AWS KMS 用來檢查資料完整性的額外驗證資料 (AAD)。也就是說，為加密操作指定加密內容時，服務也必須為解密操作指定相同的加密內容，否則解密將無法成功。Amazon Redshift 會使用叢集 ID 和加密內容的建立時間。在 CloudTrail 記錄檔的`requestParameters`欄位中，加密內容看起來與此類似。

```
"encryptionContext": {
    "aws:redshift:arn": "arn:aws:redshift:region:account_ID:cluster:cluster_name",
    "aws:redshift:createtime": "20150206T1832Z"
},
```

您可以搜尋 CloudTrail 記錄中的叢集名稱，以瞭解使用 AWS KMS key (KMS 金鑰) 執行的作業。操作包括叢集加密、叢集解密和產生資料金鑰。

Amazon Relational Database Service (Amazon RDS) 如何使用 AWS KMS

您可以使用 Amazon Relational Database Service (Amazon RDS) 在雲端中設定、操作和擴展關聯式資料庫。您可以使用 AWS 受管金鑰 或客戶受管金鑰加密 Amazon RDS 資源。Amazon RDS 建置於 Amazon Elastic Block Store (Amazon EBS) 加密以提供資料庫磁碟區的完整磁碟加密。

如需 Amazon RDS 如何使用 KMS 金鑰保護資源的詳細資訊，請參閱《Amazon RDS 使用者指南》中的加密 Amazon RDS 資源和 AWS KMS 金鑰管理。

AWS Secrets Manager 使用 AWS KMS 的方式

AWS Secrets Manager 是一項 AWS 服務，用於加密並存放您的秘密，並以透明方式解密並以純文字方式將它們傳回給您。它專門用於存放定期變更且不應硬式編碼或以純文字形式存放在應用程式中的應用程式秘密，例如登入資料。取代硬式編碼登入資料或資料表查閱，您的應用程式改為呼叫 Secrets Manager。

Secrets Manager 也支援定期輪換與常用資料庫相關之秘密的功能。在存放新輪換的秘密之前一律會對其進行加密。

Secrets Manager 與 AWS Key Management Service (AWS KMS) 整合，使用受 AWS KMS key 保護的唯一資料金鑰來加密每個機密值的每個版本。這種整合可以使用加密金鑰來保護您的秘密，絕不會讓 AWS KMS 處於未加密狀態。它還可讓您在 KMS 金鑰上設定自訂許可，並稽核產生、加密及解密用來保護您秘密之資料金鑰的操作。

如需 Secrets Manager 如何使用 KMS 金鑰來保護機密的相關資訊，請參閱《AWS Secrets Manager 使用者指南》中的加密和解密機密。

Amazon Simple Email Service (Amazon SES) 如何使用 AWS KMS

您可以使用 Amazon Simple Email Service (Amazon SES) 接收電子郵件，以及 (選用) 加密收到的電子郵件訊息，再將其存放在您選擇的 Amazon Simple Storage Service (Amazon S3) 儲存貯體中。當您設定 Amazon SES 來加密電子郵件訊息時，您必須選擇 AWS KMS AWS KMS key，Amazon SES 會使用其加密訊息。您可以選擇 Amazon SES 的 AWS 受管金鑰 (其別名為 `aws/ses`)，或者您可以選擇 AWS KMS 中建立的對稱客戶受管金鑰。

> **Important**
>
> Amazon SES 只支援對稱 KMS 金鑰。您無法使用非對稱 KMS 金鑰來加密您的 Amazon SES 電子郵件訊息。如需判斷 KMS 金鑰是對稱還是不對稱的說明，請參閱 識別非對稱 KMS 金鑰。

如需使用 Amazon SES 接收電子郵件的詳細資訊，請參閱《Amazon Simple Email Service 開發人員指南》中的使用 Amazon SES 接收電子郵件。

主題

Amazon SES 使用 AWS KMS 加密的概觀

當您設定 Amazon SES 來接收電子郵件以及在將電子郵件訊息儲存到 S3 儲存貯體之前進行加密，此程序的運作方式如下：

1. 您為 Amazon SES 建立接收規則，指定 S3 動作、用於儲存的 S3 儲存貯體，以及用於加密的 AWS KMS key。
2. Amazon SES 接收符合您接收規則的電子郵件訊息。
3. Amazon SES 請求使用您在適用之接收規則中所指定 KMS 金鑰來加密的唯一資料金鑰。
4. AWS KMS 建立新的資料金鑰、使用指定的 KMS 金鑰將它加密，然後傳送加密的和純文字資料金鑰副本給 Amazon SES。
5. Amazon SES 使用純文字資料金鑰來加密電子郵件訊息，然後在使用後盡快從記憶體移除純文字資料金鑰。
6. Amazon SES 會將加密的電子郵件訊息和加密的資料金鑰放在指定的 S3 儲存貯體中。加密的資料金鑰以中繼資料形式與加密的電子郵件訊息一起儲存。

為了透過 Step 6 完成 Step 3，Amazon SES 使用 AWS 提供的 Amazon S3 加密用戶端。使用相同的用戶端來從 Amazon S3 擷取您的加密電子郵件訊息，然後將其解密。如需詳細資訊，請參閱 取得和解密電子郵件訊息。

Amazon SES 加密內容

當 Amazon SES 請求資料金鑰來加密您收到的電子郵件訊息時 (Amazon SES 使用 AWS KMS 加密的概觀 中的 Step 3)，它會在請求中包含加密內容。加密內容提供額外驗證資料 (AAD)，供 AWS KMS 用來確保資料完整性。加密內容也會寫入 AWS CloudTrail 日誌檔案，以協助您了解為何使用指定的 AWS KMS key (KMS 金鑰)。Amazon SES 使用下列加密內容：

- 您已在其中設定 Amazon SES 來接收電子郵件訊息之 AWS 帳戶 的 ID
- 在電子郵件訊息上呼叫 S3 動作之 Amazon SES 接收規則的規則名稱

- 電子郵件訊息的 Amazon SES 訊息 ID

以下範例顯示 Amazon SES 所用加密內容的 JSON 顯示方式：

```
{
  "aws:ses:source-account": "111122223333",
  "aws:ses:rule-name": "example-receipt-rule-name",
  "aws:ses:message-id": "d6iitobk75ur44p8kdnnp7g2n800"
}
```

授予 Amazon SES 使用 AWS KMS key 的許可

若要加密電子郵件訊息，您可以使用 Amazon SES (aws/ses) 帳戶中的 AWS 受管金鑰，或者您可以使用您建立的客戶受管金鑰。Amazon SES 已經有代表您使用 AWS 受管金鑰 的許可。不過，當您新增 S3 動作到 Amazon SES 接收規則時，若要指定客戶受管金鑰，您必須給予 Amazon SES 許可，以使用 KMS 金鑰來加密您的電子郵件訊息。

若要提供 Amazon SES 使用您客戶受管金鑰的許可，請將以下陳述式新增到 KMS 金鑰的金鑰政策：

```
{
  "Sid": "Allow SES to encrypt messages using this KMS key",
  "Effect": "Allow",
  "Principal": {"Service": "ses.amazonaws.com"},
  "Action": [
    "kms:Encrypt",
    "kms:GenerateDataKey*"
  ],
  "Resource": "*",
  "Condition": {
    "Null": {
      "kms:EncryptionContext:aws:ses:rule-name": false,
      "kms:EncryptionContext:aws:ses:message-id": false
    },
    "StringEquals": {"kms:EncryptionContext:aws:ses:source-account": "ACCOUNT-ID-
WITHOUT-HYPHENS"}
  }
}
```

將 *ACCOUNT-ID-WITHOUT-HYPHENS* 替換為您已在其中設定 Amazon SES 來接收電子郵件訊息之 AWS 帳戶 的 12 位數 ID。此政策陳述式僅允許 Amazon SES 在以下條件下使用此 KMS 金鑰來加密資料：

- Amazon SES 必須在其 AWS KMS API 請求的 `EncryptionContext` 中指定 `aws:ses:rule-name` 和 `aws:ses:message-id`。
- Amazon SES 必須在其 AWS KMS API 請求的 `EncryptionContext` 中指定 `aws:ses:source-account`，且 `aws:ses:source-account` 的值必須符合金鑰政策中指定的 AWS 帳戶 ID。

如需 Amazon SES 用來加密您電子郵件訊息之加密內容的詳細資訊，請參閱 Amazon SES 加密內容。如需有關 AWS KMS 如何使用加密內容的詳細資訊，請參閱加密內容。

取得和解密電子郵件訊息

Amazon SES 沒有許可來解密您加密的電子郵件訊息，也無法為您解密。您必須編寫程式碼以從 Amazon S3 取得您的電子郵件訊息並將其解密。為了更容易這樣做，請使用 Amazon S3 加密用戶端。下列 AWS 開發套件包含 Amazon S3 加密用戶端：

- AWS SDK for Java – 請參閱《AWS SDK for Java API 參考》中的 AmazonS3EncryptionClient 和 AmazonS3EncryptionClientV2。
- AWS SDK for Ruby – 請參閱《AWS SDK for Ruby API 參考》中的 Aws::S3::Encryption::Client。
- AWS SDK for .NET – 請參閱《AWS SDK for .NET API 參考》中的 AmazonS3EncryptionClient。
- AWS SDK for Go – 請參閱《AWS SDK for Go API 參考》中的 s3crypto。

Amazon S3 加密用戶端可簡化建構必要請求的工作來要求 Amazon S3 擷取加密電子郵件訊息、要求 AWS KMS 解密訊息的加密資料金鑰，以及解密電子郵件訊息。例如，若要成功解密加密的資料金鑰，您必須傳遞從 AWS KMS (Amazon SES 使用 AWS KMS 加密的概觀 中的 Step 3) 請求資料金鑰時 Amazon SES 傳遞的相同加密內容。Amazon S3 加密用戶端會為您處理這件事，以及很多其他工作。

如需在 AWS SDK for Java 中使用 Amazon S3 加密用戶端進行用戶端解密的範本程式碼，請參閱下列各項：

- Amazon Storage Service 使用者指南中的使用存放在 AWS KMS 的 KMS 金鑰。
- AWS 開發人員部落格的使用 AWS Key Management Service 的 Amazon S3 加密。

Amazon Simple Storage Service (Amazon S3) 如何使用 AWS KMS

Amazon Simple Storage Service (Amazon S3) 是將資料當做物件存放在儲存貯體中的物件儲存服務。儲存貯體和它們中的物件是私有的，只有在您明確授予存取許可時才能進行存取。

Amazon S3 與 AWS Key Management Service (AWS KMS) 整合，以提供 Amazon S3 物件的伺服器端加密。Amazon S3 使用 AWS KMS 金鑰來加密 Simple Storage Service (Amazon S3) 物件。保護物件的加密金鑰永遠不會讓 AWS KMS 處於未加密的狀態。這項整合還可讓您在 AWS KMS 金鑰上設定許可，並稽核產生、加密及解密用來保護您秘密之資料金鑰的操作。

若要減少 Amazon S3 呼叫的數量AWS KMS，請使用 Amazon S3 儲存貯體金鑰，這些金鑰是受 KMS 金鑰保護的金鑰 key-encryption-keys 保護，可在 Amazon S3 中有限的時間內重複使用。儲存貯體金鑰可以降低 AWS KMS 請求的成本，高達 99%。您可以設定一個針對 Amazon S3 儲存貯體中所有物件或針對 Amazon S3 儲存貯體中特定物件的儲存貯體金鑰。

如需 Simple Storage Service (Amazon S3) 平台使用 AWS KMS 加密的詳細資訊，請參閱 Amazon Simple Storage Service 使用者指南中的搭配使用伺服器端加密與 KMS 金鑰 (SSE-KMS) 來保護資料。

AWS Systems Manager 參數存放區如何使用 AWS KMS

使用 AWS Systems Manager 參數存放區，您可以建立安全字串參數，這些參數具有純文字參數名稱和加密的參數值。參數存放區使用 AWS KMS 來加密和解密安全字串參數的參數值。

您可以使用參數存放區將資料作為具有值的參數來建立、存放和管理。您可以在參數存放區中建立參數，並將它用在遵守您設計之政策和許可的多個應用程式和服務。需要變更參數值時，您只需變更一個執行個體，而不用管理容易出錯的多個來源變更。參數存放區支援參數名稱的階層結構，所以您可以限定參數用於特定用途。

若要管理敏感資料，您可以建立安全字串參數。參數存放區使用 AWS KMS keys 在您建立或變更安全字串參數的參數值時對其進行加密。當您存取這些參數時，它也使用 KMS 金鑰來解密參數值。您可以使用參數存放區為您帳戶建立的 AWS 受管金鑰，或指定您自己的客戶受管金鑰。

> **Important**
>
> 參數存放區僅支援對稱 KMS 金鑰。您無法使用非對稱 KMS 金鑰來加密您的參數。如需判斷 KMS 金鑰是對稱還是不對稱的說明，請參閱 識別非對稱 KMS 金鑰。

參數存放區支援兩層級的安全字串參數：標準型和進階型。標準參數，不能超過 4096 個位元組，直接在您指定的 KMS 金鑰下加密和解密。為了加密和解密進階安全字串參數，參數存放區會使用信封加密搭配 AWS Encryption SDK。您可以將標準安全字串參數轉換成進階參數，但無法將進階參數轉換成標準參數。如需標準和進階安全字串參數之間差異的相關資訊，請參閱《AWS Systems Manager 使用者指南》中的關於 Systems Manager 進階參數。

主題

保護標準安全字串參數

參數存放區不會執行任何密碼編譯操作。而是依賴 AWS KMS 來加密和解密安全字串參數值。當您建立或變更標準安全字串參數值時，參數存放區會呼叫 AWS KMS Encrypt 操作。這個操作會直接使用對稱加密 KMS 金鑰來加密參數值，而不使用 KMS 金鑰產生資料金鑰。

您可以選擇參數存放區用於加密參數值的 KMS 金鑰。如果您未指定 KMS 金鑰，參數存放區會使用 Systems Manager 在您帳戶中自動建立的 AWS 受管金鑰。此 KMS 金鑰具有 `aws/ssm` 別名。

若要檢視您帳戶的預設 `aws/ssm` KMS 金鑰，請使用 AWS KMS API 中的DescribeKey作業。以下範例在 AWS Command Line Interface (AWS CLI) 中使用 `describe-key` 命令搭配 `aws/ssm` 別名名稱。

```
aws kms describe-key --key-id alias/aws/ssm
```

若要建立標準安全字串參數，請使用 Systems Manager API 中的PutParameter作業。省略 `Tier` 參數，或指定值為 `Standard` (預設值)。加入值為 `SecureString` 的 `Type` 參數。若要指定 KMS 金鑰，請使用 `KeyId` 參數。預設是您帳戶的 AWS 受管金鑰，即 `aws/ssm`。

參數存放區接著會使用 KMS 金鑰和純文字參數值呼叫 AWS KMS `Encrypt` 操作。AWS KMS 會傳回參數存放區使用參數名稱存放的加密參數值。

下列範例會使用 Systems Manager put-parameter 命令及其在 AWS CLI 中的 `--type` 參數來建立安全字串參數。由於命令省略了選用 `--tier` 和 `--key-id` 參數，參數存放區會建立標準安全字串參數，並使用 AWS 受管金鑰 加密此參數。

```
aws ssm put-parameter --name MyParameter --value "secret_value" --type SecureString
```

以下類似範例使用 `--key-id` 參數來指定客戶受管金鑰。此範例使用 KMS 金鑰 ID 來識別 KMS 金鑰，但您可以使用任何有效的 KMS 金鑰識別符。由於命令省略了 `Tier` 參數 (`--tier`)，參數存放區會建立標準安全字串參數，而不是進階參數。

```
aws ssm put-parameter --name param1 --value "secret" --type SecureString --key-id
 1234abcd-12ab-34cd-56ef-1234567890ab
```

當您從參數存放區取得安全字串參數時，它的值已經過加密。若要取得參數，請使用 Systems Manager API 中的GetParameter 作業。

下列範例會使用 AWS CLI 中的 Systems Manager get-parameter 命令，以取得參數存放區的 MyParameter 參數，無需解密其值。

```
$  aws ssm get-parameter --name MyParameter

{
    "Parameter": {
        "Type": "SecureString",
        "Name": "MyParameter",
        "Value":
 "AQECAHgnOkMROh5LaLXkA4j0+vYi6tmM17Lg/9E464VRo68cvwAAAG8wbQYJKoZIhvcNAQcGoGAwXgIBADBZBgkqhkiG9
    }
}
```

若要在傳回參數值之前先解密，請將 GetParameter 的 WithDecryption 參數設定為 true。當您使用 WithDecryption 時，參數存放區會代您呼叫 AWS KMS Decrypt 操作來解密參數值。因此，GetParameter 請求會傳回具有純文字參數值的參數，如以下範例所示。

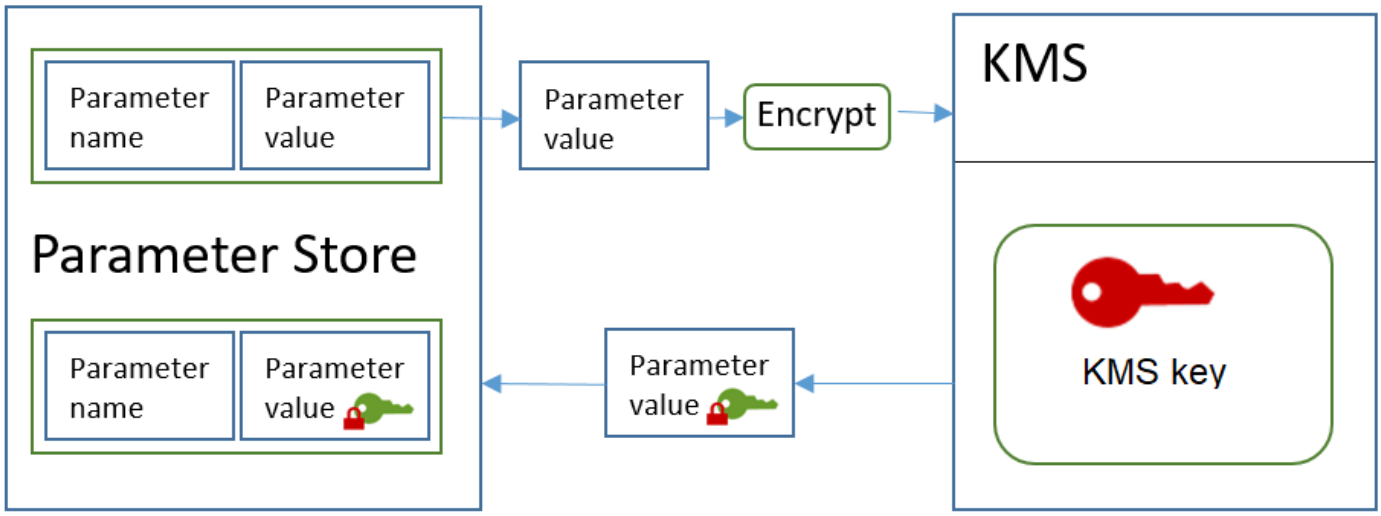
```
$  aws ssm get-parameter --name MyParameter --with-decryption

{
    "Parameter": {
        "Type": "SecureString",
        "Name": "MyParameter",
        "Value": "secret_value"
    }
}
```

以下工作流程說明參數存放區如何使用 KMS 金鑰來加密和解密標準安全字串參數。

加密標準參數

1. 當您使用 PutParameter 建立安全字串參數時，參數存放區會傳送 Encrypt 請求至 AWS KMS。該請求包含純文字參數值、您選擇的 KMS 金鑰，以及參數存放區加密內容。在傳輸到 AWS KMS 的過程中，安全字串參數中的純文字值受到 Transport Layer Security (TLS) 的保護。

2. AWS KMS 以指定的 KMS 金鑰和加密內容來加密參數值。它將加密文字傳回給參數存放區，以儲存參數名稱及其加密的值。



解密標準參數

1. 當您在 `GetParameter` 請求中包含 `WithDecryption` 參數時，參數存放區會傳送 `Decrypt` 請求至 AWS KMS，其中包含加密的安全字串參數值和參數存放區加密內容。
2. AWS KMS 使用相同的 KMS 金鑰和提供的加密內容來解密加密的值。它將純文字 (解密的) 參數值傳回至參數存放區。在傳輸過程中，純文字資料受到 TLS 保護。
3. 在 `GetParameter` 回應中，參數存放區會傳回純文字參數值給您。

保護進階安全字串參數

使用 `PutParameter` 建立進階安全字串參數時，參數存放區會使用信封加密搭配 AWS Encryption SDK 和對稱加密 AWS KMS key 來保護參數值。每個進階參數值都在唯一的資料金鑰下加密，而資料金鑰是在 KMS 金鑰下加密。您可以使用帳戶 (`aws/ssm`) 的 AWS 受管金鑰，或任何客戶受管金鑰。

AWS Encryption SDK 是開放原始碼的用戶端程式庫，可協助您使用產業標準和最佳實務來加密和解密資料。它支援多種平台和多種程式設計語言，包括命令列界面。您可以在中檢視原始程式碼並為其開發做出貢獻 GitHub。

對於每個安全字串參數值，參數存放區呼叫AWS Encryption SDK以使用 AWS KMS generates (GenerateDataKey) 的唯一資料金鑰來加密參數值。AWS Encryption SDK 為參數存放區傳回已加密訊息，其中包含加密的參數值和唯一資料金鑰的已加密複本。參數存放區會將整個加密的訊息存放在安全

字串參數值中。接著，當您取到進階安全字串參數值時，參數存放區會使用 AWS Encryption SDK 來解密參數值。這需要呼叫 AWS KMS 來解密加密的資料金鑰。

若要建立進階安全字串參數，請使用 Systems Manager API 中的PutParameter作業。將 `Tier` 參數的值設為 `Advanced`。加入值為 `SecureString` 的 `Type` 參數。若要指定 KMS 金鑰，請使用 `KeyId` 參數。預設是您帳戶的 AWS 受管金鑰，即 `aws/ssm`。

```
aws ssm put-parameter --name MyParameter --value "secret_value" --type SecureString --
tier Advanced
```

以下類似範例使用 `--key-id` 參數來指定客戶受管金鑰。此範例使用 KMS 金鑰的 Amazon Resource Name (ARN)，但您可以使用任何有效的 KMS 金鑰識別符。

```
aws ssm put-parameter --name MyParameter --value "secret_value"
 --type SecureString --tier Advanced --key-id arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab
```

當您從參數存放區取得安全字串參數時，它的值是 AWS Encryption SDK 傳回的加密訊息。若要取得參數，請使用 Systems Manager API 中的GetParameter 作業。

以下範例使用 Systems Manager `GetParameter` 操作從參數存放區取得 `MyParameter` 參數，無需解密其值。

```
$  aws ssm get-parameter --name MyParameter

{
    "Parameter": {
        "Type": "SecureString",
        "Name": "MyParameter",
        "Value":
 "AQECAHgnOkMROh5LaLXkA4j0+vYi6tmM17Lg/9E464VRo68cvwAAAG8wbQYJKoZIhvcNAQcGoGAwXgIBADBZBgkqhkiG9
    }
}
```

若要在傳回參數值之前先解密，請將 `GetParameter` 的 `WithDecryption` 參數設定為 `true`。當您使用 `WithDecryption` 時，參數存放區會代您呼叫 AWS KMS Decrypt 操作來解密參數值。因此，`GetParameter` 請求會傳回具有純文字參數值的參數，如以下範例所示。

```
$  aws ssm get-parameter --name MyParameter --with-decryption
```

```
{
    "Parameter": {
        "Type": "SecureString",
        "Name": "MyParameter",
        "Value": "secret_value"
    }
}
```

您無法將進階安全字串參數轉換成標準參數，但可以將標準安全字串轉換成進階參數。若要將標準安全字串參數轉換成進階安全字串，請使用 PutParameter 操作搭配 Overwrite 參數。Type 必須是 SecureString，Tier 值必須是 Advanced。識別客戶受管金鑰的 KeyId 參數是選用的。如果您將其省略，則參數存放區會將 AWS 受管金鑰 用於帳戶。即使您使用不同的 KMS 金鑰來加密標準參數，您也可以指定委託人有權使用的任何 KMS 金鑰。

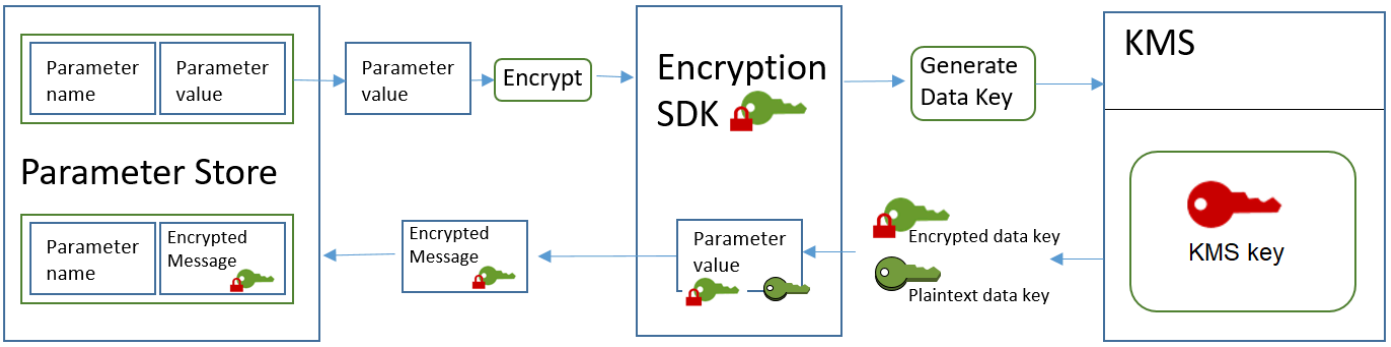
當您使用 Overwrite 參數時，參數存放區會使用 AWS Encryption SDK 來加密參數值。然後，它會將新加密的訊息存放在參數存放區中。

```
$ aws ssm put-parameter --name myStdParameter --value "secret_value"  --type
 SecureString --tier Advanced --key-id 1234abcd-12ab-34cd-56ef-1234567890ab --overwrite
```

以下工作流程說明參數存放區如何使用 KMS 金鑰來加密和解密進階安全字串參數。

加密進階參數

1. 當您使用 PutParameter 建立進階安全字串參數時，參數存放區會使用 AWS Encryption SDK 和 AWS KMS 來加密參數值。參數存放區呼叫 AWS Encryption SDK，其中包含參數值、您指定的 KMS 金鑰，以及參數存放區加密內容。
2. 會將GenerateDataKey要求AWS Encryption SDK傳送至，其中AWS KMS包含您指定的 KMS 金鑰識別碼和參數存放區加密內容。 AWS KMS傳回唯一資料金鑰的兩個複本：一個是純文字，另一個在 KMS 金鑰下加密。(加密資料金鑰時會使用加密內容。)
3. AWS Encryption SDK 使用純文字資料金鑰來加密參數值。它會傳回加密的訊息，其中包含加密的參數值、加密的資料金鑰和其他資料 (包括參數存放區加密內容)。
4. 參數存放區將加密的訊息存放為參數值。



解密進階參數

1. 您可以在 `GetParameter` 請求中包含 `WithDecryption` 參數，以取得進階安全字串參數。當您這麼做時，參數存放區會從參數值中將加密的訊息傳遞到 AWS Encryption SDK 的解密方法。
2. AWS Encryption SDK 呼叫 AWS KMS Decrypt 操作。它從加密的訊息傳入加密的資料金鑰和參數存放區加密內容。
3. AWS KMS 會使用 KMS 金鑰和參數存放區加密內容來解密已加密的資料金鑰。然後，它將純文字 (解密的) 資料金鑰傳回給 AWS Encryption SDK。
4. AWS Encryption SDK 使用純文字資料金鑰來解密參數值。它將純文字參數值傳回至參數存放區。
5. 參數存放區驗證加密內容，並在 `GetParameter` 回應中將純文字參數值傳回給您。

設定許可來加密和解密參數值

若要加密標準安全字串參數值，使用者需要 `kms:Encrypt` 許可。若要加密進階安全字串參數值，使用者需要 `kms:GenerateDataKey` 許可。若要解密任一類型的安全字串參數值，使用者需要 `kms:Decrypt` 許可。

您可以使用 IAM 政策來允許或拒絕使用者呼叫 Systems Manager `PutParameter` 和 `GetParameter` 操作的許可。

如果您使用客戶受管金鑰來加密安全字串參數值，您可以使用 IAM 政策和金鑰政策來管理加密和解密許可。不過，您無法為預設 `aws/ssm` KMS 金鑰建立存取控制政策。如需控制對客戶受管金鑰之存取權的詳細資訊，請參閱 AWS KMS 的身分驗證與存取控制。

以下範例顯示專為標準安全字串參數設計的 IAM 政策。它可讓使用者在 `FinancialParameters` 路徑中的所有參數上呼叫 Systems Manager `PutParameter` 操作。此政策也可讓使用者在範例客戶受管金鑰上呼叫 AWS KMS `Encrypt` 操作。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:PutParameter"
            ],
            "Resource": "arn:aws:ssm:us-west-2:111122223333:parameter/
FinancialParameters/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Encrypt"
            ],
            "Resource": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ]
}
```

以下範例顯示專為進階安全字串參數設計的 IAM 政策。它可讓使用者在 ReservedParameters 路徑中的所有參數上呼叫 Systems Manager PutParameter 操作。此政策也可讓使用者在範例客戶受管金鑰上呼叫 AWS KMS GenerateDataKey 操作。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:PutParameter"
            ],
            "Resource": "arn:aws:ssm:us-west-2:111122223333:parameter/
ReservedParameters/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:GenerateDataKey"
            ],
```

```
            "Resource": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ]
}
```

最後一個範例還顯示可用於標準或進階安全字串參數的 IAM 政策。它可讓使用者在 ITParameters 路徑中的所有參數上呼叫 Systems Manager GetParameter 操作 (以及相關操作)。此政策也可讓使用者在範例客戶受管金鑰上呼叫 AWS KMS Decrypt 操作。

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetParameter*"
            ],
            "Resource": "arn:aws:ssm:us-west-2:111122223333:parameter/ITParameters/*"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
        }
    ]
}
```

參數存放區加密內容

加密內容是一組金鑰/值對，其中包含任意非私密資料。在加密資料的請求中包含加密內容時，AWS KMS 會以密碼編譯方式將加密內容繫結至加密的資料。若要解密資料，您必須傳遞相同的加密內容。

您也可以使用加密內容來識別稽核記錄和日誌中的密碼編譯操作。加密內容會以純文字形式出現在日誌中，例如 AWS CloudTrail 日誌。

AWS Encryption SDK 也接受加密內容，但處理方式不同。參數存放區提供加密內容給加密方法。AWS Encryption SDK 以密碼編譯方式將加密內容繫結至加密的資料。它在傳回的已加密訊息的標頭中，還會包含純文字形式的加密內容。不過，與 AWS KMS 不同，AWS Encryption SDK 解密方法

不接受加密內容做為輸入。相反地，當其解密資料時，AWS Encryption SDK 會從加密的訊息中取得加密內容。參數存放區驗證加密內容是否包含在為您傳回純文字參數值之前其所預期的值。

參數存放區在其密碼編譯操作中使用下列加密內容：

- 索引鍵：PARAMETER_ARN
- 值：要加密之參數的 Amazon Resource Name (ARN)。

加密內容的格式如下：

```
"PARAMETER_ARN":"arn:aws:ssm:<REGION_NAME>:<ACCOUNT_ID>:parameter/<parameter-name>"
```

例如，參數存放區在呼叫中包含此加密內容，用來加密和解密範例 AWS 帳戶 和區域中的 `MyParameter` 參數。

```
"PARAMETER_ARN":"arn:aws:ssm:us-west-2:111122223333:parameter/MyParameter"
```

如果參數位於參數存放區階層路徑，則路徑和名稱會包含在加密內容中。例如，在範例 AWS 帳戶 和區域的 `/ReadableParameters` 路徑中加密和解密 `MyParameter` 參數時，會使用此加密內容。

```
"PARAMETER_ARN":"arn:aws:ssm:us-west-2:111122223333:parameter/ReadableParameters/
MyParameter"
```

您可以透過使用正確的加密內容和 Systems Manager `GetParameter` 操作傳回的加密參數值來呼叫 AWS KMS `Decrypt` 操作，解密已加密的安全字串參數值。不過，我們建議您使用 `GetParameter` 操作搭配 `WithDecryption` 參數來解密參數存放區參數值。

您也可以在 IAM 政策包含加密內容。例如，您可以允許使用者只解密一個特定的參數值或一組參數值。

以下範例 IAM 政策陳述式允許使用者取得 `MyParameter` 參數的值和使用指定的 KMS 金鑰來解密其值。不過，只有在加密內容符合指定的字串時，才會套用許可。這些許可不適用於任何其他參數或 KMS 金鑰，如果加密內容不符合字串，則呼叫 `GetParameter` 會失敗。

使用這類政策陳述式之前，請將範例 ARN 換成有效的值。

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

```
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "ssm:GetParameter*"
            ],
            "Resource": "arn:aws:ssm:us-west-2:111122223333:parameter/MyParameter"
        },
        {
            "Effect": "Allow",
            "Action": [
                "kms:Decrypt"
            ],
            "Resource": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab",
            "Condition": {
                "StringEquals": {
                    "kms:EncryptionContext:PARAMETER_ARN":"arn:aws:ssm:us-
west-2:111122223333:parameter/MyParameter"
                }
            }
        }
    ]
}
```

疑難排解參數存放區中的 KMS 金鑰問題

若要在安全字串參數上執行任何操作，參數存放區必須能夠使用您為預期操作指定的 AWS KMS KMS 金鑰。KMS 金鑰相關的大部分參數存放區故障是由下列問題引起：

- 應用程式使用的登入資料無權在 KMS 金鑰上執行指定的動作。

  若要修正此錯誤，請使用不同的登入資料來執行應用程式，或修改防止操作的 IAM 或金鑰政策。如需 AWS KMS IAM 和金鑰政策的相關協助，請參閱AWS KMS 的身分驗證與存取控制。

- 找不到 KMS 金鑰。

  這種情況通常發生在您為 KMS 金鑰使用不正確的識別符。為 KMS 金鑰尋找正確的識別符，再試一次命令。

- 未啟用 KMS 金鑰。發生這種情況時，參數存放區會傳回InvalidKeyId例外狀況，其中包含詳細的錯誤訊息AWS KMS。如果 KMS 金鑰狀態為 `Disabled`，請啟用。如果是 `Pending Import`，請完成匯入程序。如果金鑰狀態為 `Pending Deletion`，請取消金鑰刪除或使用不同的 KMS 金鑰。

若要尋找 AWS KMS 主控台上 KMS 金鑰的金鑰狀態，在 Customer managed keys (客戶受管金鑰) 或 AWS 受管金鑰 頁面上，請參閱「狀態」欄。若要使用 AWS KMS API 尋找 KMS 金鑰的狀態，請使用DescribeKey作業。

Amazon 如何 WorkMail 使用 AWS KMS

本主題討論 Amazon 如何 WorkMail 用AWS KMS來加密電子郵件訊息。

主題

Amazon WorkMail 概述

Amazon WorkMail 是安全的受管商業電子郵件和行事曆服務，可支援現有桌面和行動電子郵件用戶端。您可以建立 Amazon WorkMail 組織，並為其指派一或多個您擁有的電子郵件網域。然後，您可以為組織中的電子郵件使用者和通訊群組建立信箱。

Amazon 會在將訊息寫入磁碟之前， WorkMail 透明地加密所有 Amazon WorkMail 組織信箱中的所有訊息，並在使用者存取訊息時以透明方式解密訊息。沒有可停用加密的選項。為了保護訊息的加密金鑰，Amazon WorkMail 已與 AWS Key Management Service (AWS KMS) 整合。

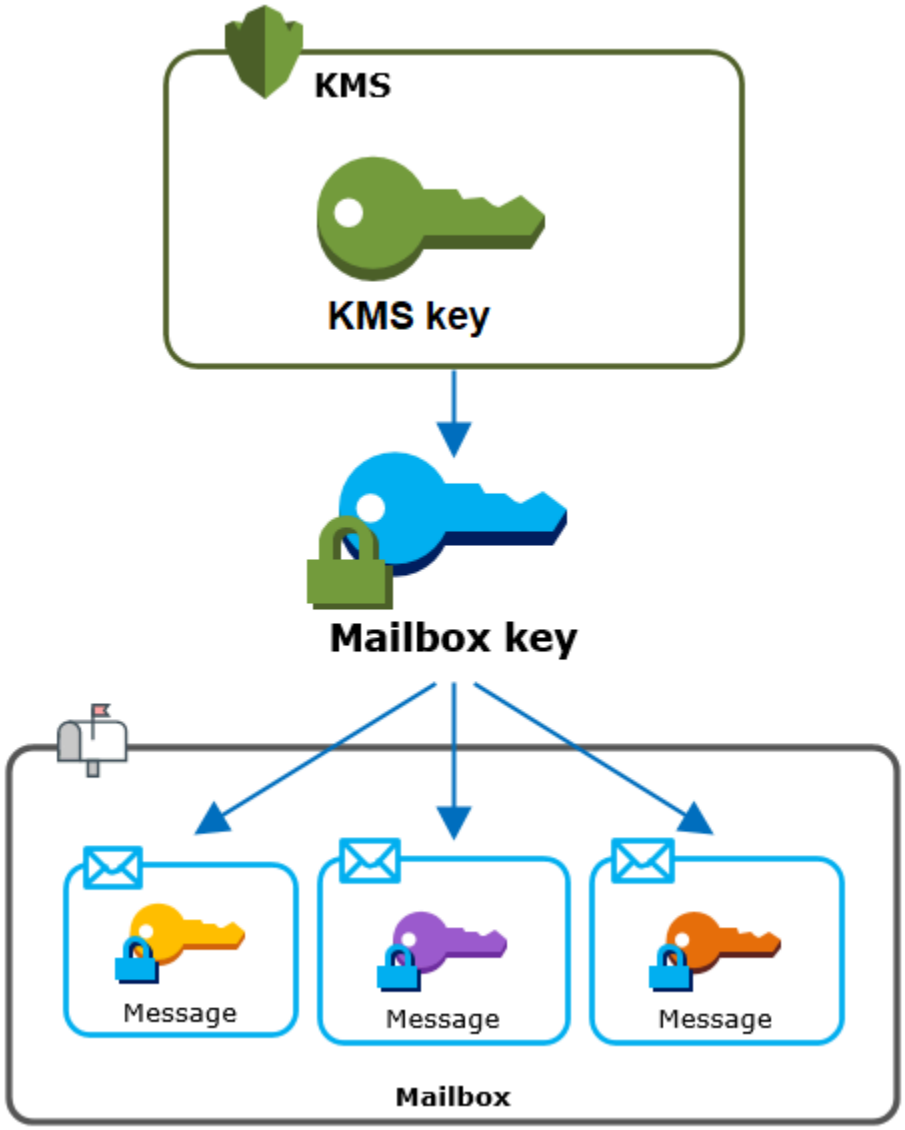
Amazon WorkMail 也提供一個選項，讓使用者能夠傳送已簽署或加密的電子郵件。此加密功能不使用 AWS KMS。

Amazon WorkMail 加密

在 Amazon 中 WorkMail，每個組織都可以包含多個信箱，組織中的每個使用者一個信箱。所有訊息 (包括電子郵件和行事曆項目) 都存放在使用者的信箱中。

為了保護 Amazon WorkMail 組織中信箱的內容，Amazon 會在所有信箱訊息寫入磁碟之前先 WorkMail 加密。客戶提供的資訊都不以純文字形式儲存。

每個訊息都在唯一的資料加密金鑰下加密。訊息金鑰由信箱金鑰加密，這是該信箱專用的唯一加密金鑰。信箱金鑰在組織的 AWS KMS key 下加密，永遠不會放任 AWS KMS 未加密。下圖顯示 AWS KMS 中在加密訊息、加密訊息金鑰、加密信箱金鑰及組織 KMS 金鑰之間的關係。



組織的 KMS 金鑰

建立 Amazon 組 WorkMail 織時，您可以為組織選取AWS KMS key一個組織。這個 KMS 金鑰保護該組織中的所有信箱金鑰。

如果您使用快速設置過程來創建您的組織，Amazon WorkMail 使AWS 受管金鑰用 Amazon WorkMail（`aws/workmail`）在你的AWS 帳戶. 如果您使用標準設定，您可以選AWS 受管金鑰取 Amazon WorkMail 或您擁有和管理的客戶管理金鑰。您可以為每個組織選取相同的 KMS 金鑰或不同的 KMS 金鑰，但一旦選取 KMS 金鑰之後就不能變更。

> **Important**
>
> Amazon 僅 WorkMail 支持對稱加密 KMS 密鑰。您無法使用非對稱 KMS 金鑰來加密 Amazon 中的資料 WorkMail。如需判斷 KMS 金鑰是對稱還是不對稱的說明，請參閱 識別非對稱 KMS 金鑰。

若要為您的組織尋找 KMS 金鑰，請使用 AWS CloudTrail 日誌項目，該項目記錄對 AWS KMS 的呼叫。

每個信箱的唯一加密金鑰

當您建立新信箱時，Amazon WorkMail 會為信箱產生唯一的 256 位元進階加密標準 (AES) 對稱加密金鑰 (稱為其信箱金鑰)。AWS KMSAmazon WorkMail 使用信箱金鑰來保護信箱中每個訊息的加密金鑰。

為了保護信箱金鑰，Amazon 會 WorkMail 呼叫AWS KMS在組織的 KMS 金鑰下加密信箱金鑰。然後，它會將加密的信箱金鑰存放在信箱中繼資料。

> **Note**
>
> Amazon WorkMail 使用對稱信箱加密金鑰來保護訊息金鑰。之前，Amazon 會使用非對稱 key pair 來 WorkMail 保護每個信箱。它使用公有金鑰來加密每個訊息金鑰，並使用私有金鑰來解密金鑰。私有信箱金鑰由組織的 KMS 金鑰保護。現有的信箱可能仍使用非對稱信箱金鑰對。此變更不會影響信箱或其訊息的安全性。

每個訊息的唯一加密金鑰

將訊息新增至信箱時，Amazon WorkMail 會為其外的訊息產生唯一的 256 位元 AES 對稱加密金鑰。AWS KMS它會使用此訊息金鑰來加密訊息。Amazon WorkMail 會加密信箱金鑰下的訊息金鑰，並將加密的訊息金鑰與訊息一起儲存。然後，它在組織的 KMS 金鑰下加密信箱金鑰。

建立新信箱

Amazon WorkMail 建立新信箱時，會使用下列程序準備信箱以保存加密訊息。

- Amazon WorkMail 會為外部的信箱產生唯一的 256 位元 AES 對稱加密金鑰。AWS KMS
- Amazon WorkMail 調用加AWS KMS密操作。它傳入信箱金鑰和組織的 AWS KMS key 識別符。AWS KMS 傳回在 KMS 金鑰下加密之信箱金鑰的加密文字。

- Amazon 會將加密的信箱金鑰與信箱中繼資料一起 WorkMail 儲存。

加密信箱訊息

要加密消息，Amazon WorkMail 使用以下過程。

1. Amazon 為消息 WorkMail 生成一個唯一的 256 位 AES 對稱密鑰。它使用純文字訊息金鑰和進階加密標準 (AES) 演算法，在 AWS KMS 外部加密訊息。
2. 為了保護信箱金鑰下的訊息金鑰，Amazon WorkMail 需要解密信箱金鑰，信箱金鑰一律以其加密形式儲存。

   Amazon WorkMail 呼叫「AWS KMS解密」作業，並傳入加密的信箱金鑰。 AWS KMS使用組織的 KMS 金鑰解密信箱金鑰，並將純文字信箱金鑰傳回給 Amazon。 WorkMail
3. Amazon WorkMail 使用純文字信箱金鑰和進階加密標準 (AES) 演算法來加密外部的訊息金鑰。AWS KMS
4. Amazon 會將加密的訊息金鑰 WorkMail 儲存在加密訊息的中繼資料中，以便對其進行解密。

解密信箱訊息

要解密消息，Amazon WorkMail 使用以下過程。

1. Amazon WorkMail 呼叫「AWS KMS解密」作業，並傳入加密的信箱金鑰。 AWS KMS使用組織的 KMS 金鑰解密信箱金鑰，並將純文字信箱金鑰傳回給 Amazon。 WorkMail
2. Amazon WorkMail 使用純文字信箱金鑰和進階加密標準 (AES) 演算法來解密外部的加密訊息金鑰。AWS KMS
3. Amazon WorkMail 使用純文字訊息金鑰來解密加密的訊息。

快取信箱金鑰

為了提升效能並將呼叫減至最少AWS KMS，Amazon 會在本機 WorkMail 快取每個用戶端的每個純文字信箱金鑰，最長可達一分鐘。在快取期間結束時，就會移除信箱金鑰。如果在快取期間需要該用戶端的信箱金鑰，Amazon WorkMail 可以從快取取取得金鑰，而不必呼叫AWS KMS。信箱金鑰放在快取中保護，絕對不會以純文字形式寫入磁碟。

授權使用 KMS 金鑰

當 Amazon WorkMail 使用密碼編譯作業時，它會代表信箱管理員執行動作。AWS KMS key

若要代表您對機密使用 AWS KMS key，管理員必須擁有以下許可。您可以在 IAM 政策或金鑰策略中指定這些必要的許可。

- `kms:Encrypt`
- `kms:Decrypt`
- `kms:CreateGrant`

若要允許 KMS 金鑰僅用於源自 Amazon 的請求 WorkMail，您可以使用帶有`workmail.<region>.amazonaws.com`值的 kms: ViaService 條件金鑰。

您也可以使用加密內容中的金鑰或值作為條件，以便將 KMS 金鑰用於密碼編譯操作。例如，您可以在 IAM 或金鑰政策文件中使用字串條件運算子，或在授權中使用授權限制。

AWS 受管金鑰 的金鑰政策

適用於 Amazon 的金鑰政策授 WorkMail 予使用者僅在 Amazon 代表使用者發出請求時，才允許 WorkMail 使用者將 KMS 金鑰用於指定的作業。AWS 受管金鑰金鑰政策不允許任何使用者直接使用 KMS 金鑰。

此金鑰政策與所有 AWS 受管金鑰 的政策一樣，都是由服務建立。您無法變更金鑰政策，但可以隨時進行檢視。如需詳細資訊，請參閱 檢視金鑰政策。

金鑰政策中的政策陳述式具有下列效果：

- 允許帳戶和區域中的使用者使用 KMS 金鑰進行加密操作和建立授權，但只有當請求來自 Amazon WorkMail 代表他們。`kms:ViaService` 條件金鑰會強制實施此限制。
- 允許 AWS 帳戶 建立 IAM 政策，以便使用者檢視 KMS 金鑰屬性和撤銷授予。

以下是 Amazon 示例AWS 受管金鑰的關鍵政策 WorkMail。

```
{
  "Version" : "2012-10-17",
  "Id" : "auto-workmail-1",
  "Statement" : [ {
    "Sid" : "Allow access through WorkMail for all principals in the account that are
 authorized to use WorkMail",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "*"
```

```
    },
    "Action" : [ "kms:Decrypt", "kms:CreateGrant", "kms:ReEncrypt*", "kms:DescribeKey",
 "kms:Encrypt" ],
    "Resource" : "*",
    "Condition" : {
      "StringEquals" : {
        "kms:ViaService" : "workmail.us-east-1.amazonaws.com",
        "kms:CallerAccount" : "111122223333"
      }
    }
  }, {
    "Sid" : "Allow direct access to key metadata to the account",
    "Effect" : "Allow",
    "Principal" : {
      "AWS" : "arn:aws:iam::111122223333:root"
    },
    "Action" : [ "kms:Describe*", "kms:List*", "kms:Get*", "kms:RevokeGrant" ],
    "Resource" : "*"
  } ]
}
```

使用贈款授權 Amazon WorkMail

除了金鑰政策之外，Amazon 還 WorkMail 使用授權將許可新增至每個組織的 KMS 金鑰。若要檢視您帳戶中 KMS 金鑰的授權，請使用ListGrants作業。

Amazon WorkMail 使用授權將下列許可新增至組織的 KMS 金鑰。

- 新增允許 Amazon WorkMail 加密信箱金鑰的`kms:Encrypt`權限。
- 新增允許 Amazon 使 WorkMail 用 KMS 金鑰解密信箱金鑰的`kms:Decrypt`權限。Amazon 在授權中 WorkMail 需要此權限，因為讀取信箱訊息的請求會使用讀取訊息之使用者的安全內容。請求不會使用 AWS 帳戶 的憑證。Amazon WorkMail 會在您為組織選取 KMS 金鑰時建立此授權。

若要建立贈款，Amazon 會代表建立組織的使用者 WorkMail 呼叫CreateGrant。建立授與的許可來自金鑰政策。此政策允許帳戶使用者`CreateGrant`在 Amazon 代表授權使用者提出要求時， WorkMail 使用組織的 KMS 金鑰呼叫。

金鑰政策也允許帳戶根使用者在 AWS 受管金鑰 上撤銷授予。但是，如果您撤銷授權，Amazon 將 WorkMail 無法解密信箱中的加密資料。

Amazon WorkMail 加密環境

加密內容是一組金鑰/值對，其中包含任意非私密資料。在加密資料的請求中包含加密內容時，AWS KMS 會以密碼編譯方式將加密內容繫結至加密的資料。若要解密資料，您必須傳遞相同的加密內容。

Amazon 在所AWS KMS有加密操作中 WorkMail 使用相同的加密內容格式。您可以使用加密內容來識別稽核記錄和日誌 (例如 AWS CloudTrail) 中的這些密碼編譯操作，以及在政策和授與中做為授權的條件。

在其加密和解密請求中AWS KMS，Amazon WorkMail 使用密鑰所在的加密上下文，值是組織的 Amazon 資源名稱 (ARN) 。`aws:workmail:arn`

```
"aws:workmail:arn":"arn:aws:workmail:region:account ID:organization/organization ID"
```

例如，以下加密內容包含美國東部 (俄亥俄) (`us-east-2`) 區域中的範例組織 ARN。

```
"aws:workmail:arn":"arn:aws:workmail:us-east-2:111122223333:organization/
m-68755160c4cb4e29a2b2f8fb58f359d7"
```

監控 Amazon WorkMail 互動 AWS KMS

您可以使用AWS CloudTrail和 Amazon CloudWatch 日誌來跟踪 Amazon 代表您 WorkMail 發送到AWS KMS的請求。

加密

當您建立新信箱時，Amazon WorkMail 會產生信箱金鑰並呼叫AWS KMS以加密信箱金鑰。Amazon WorkMail 會將加密請求傳送至，其中AWS KMS包含純文字信箱金鑰和 Amazon WorkMail 組織 KMS 金鑰的識別碼。

記錄 `Encrypt` 操作的事件類似於以下範例事件。用戶是 Amazon WorkMail 服務。這些參數包括 Amazon WorkMail 組織的 KMS 金鑰 ID (`keyId`) 和加密內容。Amazon WorkMail 也通過在郵箱密鑰，但不會記錄在 CloudTrail 日誌中。

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "workmail.eu-west-1.amazonaws.com"
```

```
    },
    "eventTime": "2019-02-19T10:01:09Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Encrypt",
    "awsRegion": "eu-west-1",
    "sourceIPAddress": "workmail.eu-west-1.amazonaws.com",
    "userAgent": "workmail.eu-west-1.amazonaws.com",
    "requestParameters": {
        "encryptionContext": {
            "aws:workmail:arn": "arn:aws:workmail:eu-west-1:111122223333:organization/
m-c6981ff7642446fa8772ba99c690e455"
        },
        "keyId": "arn:aws:kms:eu-
west-1:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d"
    },
    "responseElements": null,
    "requestID": "76e96b96-7e24-4faf-a2d6-08ded2eaf63c",
    "eventID": "d5a59c18-128a-4082-aa5b-729f7734626a",
    "readOnly": true,
    "resources": [
        {
            "ARN": "arn:aws:kms:eu-
west-1:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d",
            "accountId": "111122223333",
            "type": "AWS::KMS::Key"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333",
    "sharedEventID": "d08e60f1-097e-4a00-b7e9-10bc3872d50c"
}
```

解密

當您新增、檢視或刪除信箱訊息時，Amazon WorkMail 會AWS KMS要求您解密信箱金鑰。Amazon WorkMail 會將解密請求傳送至，其中AWS KMS包含加密信箱金鑰和 Amazon WorkMail 組織 KMS 金鑰的識別碼。

記錄 `Decrypt` 操作的事件類似於以下範例事件。用戶是 Amazon WorkMail 服務。這些參數包括未記錄在記錄中的加密信箱金鑰 (做為密文 Blob)，以及 Amazon WorkMail 組織的加密內容。 AWS KMS從加密文字衍生出 KMS 金鑰的識別碼。

```
{
```

```
    "eventVersion": "1.05",
    "userIdentity": {
        "type": "AWSService",
        "invokedBy": "workmail.eu-west-1.amazonaws.com"
    },
    "eventTime": "2019-02-20T11:51:10Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "Decrypt",
    "awsRegion": "eu-west-1",
    "sourceIPAddress": "workmail.eu-west-1.amazonaws.com",
    "userAgent": "workmail.eu-west-1.amazonaws.com",
    "requestParameters": {
        "encryptionContext": {
            "aws:workmail:arn": "arn:aws:workmail:eu-west-1:111122223333:organization/
m-c6981ff7642446fa8772ba99c690e455"
        }
    },
    "responseElements": null,
    "requestID": "4a32dda1-34d9-4100-9718-674b8e0782c9",
    "eventID": "ea9fd966-98e9-4b7b-b377-6e5a397a71de",
    "readOnly": true,
    "resources": [
        {
            "ARN": "arn:aws:kms:eu-
west-1:111122223333:key/1a2b3c4d-5e6f-1a2b-3c4d-5e6f1a2b3c4d",
            "accountId": "111122223333",
            "type": "AWS::KMS::Key"
        }
    ],
    "eventType": "AwsApiCall",
    "recipientAccountId": "111122223333",
    "sharedEventID": "241e1e5b-ff64-427a-a5b3-7949164d0214"
}
```


如何 WorkSpaces 使用 AWS KMS

您可以使WorkSpaces用為每個使用者佈建雲端桌面 (a WorkSpace)。當您啟動新的磁碟區時 WorkSpace，您可以選擇加密其磁碟區，並決定AWS KMS key要使用哪一個加密。您可以選擇 AWS 受管金鑰for WorkSpaces (aws/Workspace) 或對稱的客戶管理金鑰。

> **Important**
>
> WorkSpaces 僅支援對稱式加密 KMS 金鑰。您無法使用非對稱 KMS 金鑰來加密 WorkSpaces. 如需有關如何判斷 KMS 金鑰是對稱還是不對稱的說明，請參閱識別非對稱 KMS 金鑰。

如需有關 WorkSpaces 使用加密磁碟區建立的詳細資訊，請參閱 Amazon WorkSpaces 管理指南 WorkSpace中的加密 a。

主題

- 使用的 WorkSpaces 加密概述 AWS KMS
- WorkSpaces 加密上下文
- WorkSpaces 授予代表您使用 KMS 金鑰的權限

使用的 WorkSpaces 加密概述 AWS KMS

使 WorkSpaces 用加密磁碟區建立時，請 WorkSpaces 使用 Amazon Elastic Block Store (Amazon EBS) 來建立和管理這些磁碟區。這兩項服務都透過您的 AWS KMS key 使用已加密磁碟區。如需 EBS 磁碟區加密的詳細資訊，請參閱下列文件：

- 本指南中的 Amazon Elastic Block Store (Amazon EBS) 如何使用 AWS KMS
- 《Amazon EC2 Windows 執行個體使用者指南》中的 Amazon EBS 加密

當您 WorkSpaces 使用加密磁碟區啟動時， end-to-end 程序的運作方式如下：

1. 您可以指定用於加密的 KMS 金鑰，以及使用 WorkSpace者和目錄。此動作會建立僅允許針對此使 WorkSpaces 用 KMS 金鑰的授權，也就是 WorkSpace說，僅適用於與指定的使用者和目錄 WorkSpace 相關聯的 KMS 金鑰。
2. WorkSpaces 會建立加密的 EBS 磁碟區， WorkSpace 並指定要使用的 KMS 金鑰以及磁碟區的使用者和目錄 (與您在指定的資訊相同Step 1)。此動作會建立授權，讓 Amazon EBS 僅針對此 WorkSpace 和磁碟區使用您的 KMS 金鑰 — 也就是說，僅針對與指定的使用者和目錄 WorkSpace 相關聯的使用者和目錄，而且僅針對指定的磁碟區使用您的 KMS 金鑰。
3. Amazon EBS 請求使用 KMS 金鑰加密的磁碟區資料金鑰，並將 WorkSpace 使用者`Sid`和目錄 ID 以及磁碟區 ID 指定為加密內容。

4. AWS KMS 建立新的資料加密金鑰，在 KMS 金鑰下將其加密，然後將加密的資料金鑰傳送給 Amazon EBS。
5. WorkSpaces 使用 Amazon EBS 將加密磁碟區附加到您的 WorkSpace. Amazon EBS 會隨`Decrypt`請求一AWS KMS起傳送加密的資料金鑰`Sid`，並指定使用 WorkSpace 者的目錄 ID 和磁碟區 ID (用作加密內容)。
6. AWS KMS 使用您的 KMS 金鑰來解密資料金鑰，然後將純文字資料金鑰傳送到 Amazon EBS。
7. Amazon EBS 使用純文字資料金鑰來加密進出已加密磁碟區的所有資料。Amazon EBS 會在磁碟區連接到記憶體中保留純文字資料金鑰。 WorkSpace
8. Amazon EBS 會將加密的資料金鑰 (接收於Step 4) 與磁碟區中繼資料一起儲存，以備 future 在重新啟動或重建時使用。 WorkSpace
9. 當您使用移AWS Management Console除 WorkSpace (或使用 WorkSpaces API 中的`TerminateWorkspaces`動作) 時， WorkSpaces Amazon EBS 會淘汰允許他們為此使用 KMS 金鑰的授權。 WorkSpace

WorkSpaces 加密上下文

WorkSpaces 不會AWS KMS key直接使用您的加密操作 (例如 `Encrypt`，等) `DecryptGenerateDataKey`，這意味著 WorkSpaces 不會將請求發送到包含加密上下文AWS KMS的請求。但是，當 Amazon EBS 為 WorkSpaces (Step 3在中使用的 WorkSpaces 加密概述 AWS KMS) 的加密磁碟區請求加密的資料金鑰時，當它要求該資料金鑰的純文字複本 (Step 5) 時，它會在請求中包含加密內容。加密內容提供額外驗證資料 (AAD)，供 AWS KMS 用來確保資料完整性。加密內容也會寫入 AWS CloudTrail 日誌檔案，以協助您了解為何使用指定的 AWS KMS key。Amazon EBS 將以下項目用於加密內容：

- 與`sid`之相關聯的AWS Directory Service使用者的 WorkSpace
- 與之相關聯之AWS Directory Service目錄的目錄識別碼 WorkSpace
- 加密磁碟區的磁碟區 ID

以下範例顯示 Amazon EBS 所用加密內容的 JSON 顯示方式：

```
{
  "aws:workspaces:sid-directoryid":
 "[S-1-5-21-277731876-1789304096-451871588-1107]@[d-1234abcd01]",
  "aws:ebs:id": "vol-1234abcd"
}
```

WorkSpaces 授予代表您使用 KMS 金鑰的權限

您可以在 AWS 受管金鑰 for WorkSpaces (aws/workspace) 或客戶受管金鑰下保護工作區資料。如果您使用客戶受管金鑰，則需要 WorkSpaces 授予權限，才能代表帳戶中的系統管理 WorkSpaces員使用 KMS 金鑰。依預設，AWS 受管金鑰對 WorkSpaces 具有必要的權限。

若要準備您的客戶管理金鑰以供搭配使用 WorkSpaces，請遵循下列步驟。

1. 將 WorkSpaces 管理員新增至 KMS 金鑰金鑰原則中的金鑰使用者清單
2. 使用 IAM 政策為 WorkSpaces 管理員提供其他許可

WorkSpaces 管理員還需要使用權限 WorkSpaces。如需有關這些許可的詳細資訊，請參閱 Amazon WorkSpaces 管理指南中的控制 WorkSpaces資源存取。

第 1 部分：將 WorkSpaces 管理員新增至 KMS 金鑰的金鑰使用者

若要提供 WorkSpaces 管理員所需的權限，您可以使用AWS Management Console或 AWS KMS API。

將 WorkSpaces 管理員新增為 KMS 金鑰的金鑰使用者 (主控台)

1. 請登入 AWS Management Console，並開啟 AWS Key Management Service (AWS KMS) 主控台 (網站：https://console.aws.amazon.com/kms)。
2. 若要變更 AWS 區域，請使用頁面右上角的區域選取器。
3. 在導覽窗格中，選擇 Customer managed keys (客戶受管金鑰)。
4. 選擇您偏好的客戶受管金鑰的金鑰 ID 或別名。
5. 選擇 Key policy (金鑰政策) 標籤。在 Key users (金鑰使用者) 中，選擇 Add (新增)。
6. 在 IAM 使用者和角色清單中，選取對應至 WorkSpaces 管理員的使用者和角色，然後選擇 [連接]。

將 WorkSpaces 管理員新增為 KMS 金鑰 (AWS KMSAPI) 的金鑰使用者

1. 使用此GetKeyPolicy作業取得現有金鑰原則，然後將原則文件儲存至檔案。
2. 在您偏好的文字編輯器中開啟政策文件。將與您的 WorkSpaces 管理員對應的 IAM 使用者和角色新增至授予關鍵使用者權限的政策陳述式。接著儲存檔案。
3. 使用此PutKeyPolicy作業將金鑰原則套用至 KMS 金鑰。

第 2 部分：賦予 WorkSpaces 管理員額外權限

如果您使用客戶管理的金鑰來保護您的 WorkSpaces 資料，除了預設金鑰原則的金鑰使用者區段中的權限外，管理 WorkSpaces 員還需要對 KMS 金鑰建立授權的權限。此外，如果他們使用建立AWS Management Console WorkSpaces 加密磁碟區，則 WorkSpaces 管理員需要列出別名和列出金鑰的權限。如需建立和編輯 IAM 使用者政策的詳細資訊，請參閱《IAM 使用者指南》中的受管政策和內嵌政策。

若要將這些權限授予您的 WorkSpaces 管理員，請使用 IAM 政策。將類似下列範例的政策陳述式新增至每個 WorkSpaces管理員的 IAM 政策。將範例 KMS 金鑰 ARN (*arn:aws:kms:us-west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab*) 以有效的 ARN 取代。如果您的 WorkSpaces 系統管理員只使用 WorkSpaces API (而非主控台)，您可以省略具有"kms:ListAliases"和"kms:ListKeys"權限的第二個原則陳述式。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "kms:CreateGrant",
      "Resource": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab"
    },
    {
      "Effect": "Allow",
      "Action": [
        "kms:ListAliases",
        "kms:ListKeys"
      ],
      "Resource": "*"
    }
  ]
}
```

對 AWS KMS API 進行程式設計

您可以使用 AWS KMS API 來建立和管理 KMS 金鑰和特殊功能，例如自訂金鑰存放區，以及在密碼編譯操作中使用 KMS 金鑰。如需詳細資訊，請參閱 AWS Key Management Service API 參考。

以下主題的範本程式碼示範如何使用 AWS 開發套件呼叫 AWS KMS API。

如需使用 AWS KMS 主控台執行其中一些任務的資訊，請參閱管理金鑰。

主題

建立用戶端

若要 JavaScript 在 Node.js 中使用AWS SDK for .NETAWS SDK for Python (Boto3)、AWS SDK for Ruby、AWS SDK for PHP、或 AWS SDK 來撰寫使用 AWS Key Management Service(AWS KMS) API 的程式碼，請先建立用AWS KMS戶端。AWS SDK for Java

您建立的用戶端物件將用於隨後主題中的範例程式碼。

Java

若要在 Java 中建立 AWS KMS 用戶端，請使用用戶端建置器。

```
AWSKMS kmsClient = AWSKMSClientBuilder.standard().build();
```

如需使用 Java 用戶端建置器的更多資訊，請參閱下列資源。

- AWS 開發人員部落格上的流暢用戶端建置器

- 《AWS SDK for Java 開發人員指南》中的建立服務用戶端
- AWS SDK for Java API 參考中的 AWSKMSClientBuilder

C#

```
AmazonKeyManagementServiceClient kmsClient = new AmazonKeyManagementServiceClient();
```

Python

```
kms_client = boto3.client('kms')
```

Ruby

```
require 'aws-sdk-kms'  # in v2: require 'aws-sdk'

kmsClient = Aws::KMS::Client.new
```

PHP

若要在 PHP 中建立 AWS KMS 用戶端，請使用 AWS KMS 用戶端物件，並指定版本 2014-11-01。如需詳細資訊，請參閱《AWS SDK for PHP API 參考》中的 KMSClient 類別。

```
// Create a KMSClient
$KmsClient = new Aws\Kms\KmsClient([
    'profile' => 'default',
    'version' => '2014-11-01',
    'region'  => 'us-east-1'
]);
```

Node.js

```
const kmsClient = new AWS.KMS();
```

處理金鑰

此主題中的範例使用 AWS KMS API 建立、檢視、啟用和停用的 AWS KMS AWS KMS keys，並產生資料金鑰。

主題

建立 KMS 金鑰

若要建立 AWS KMS key(KMS 金鑰)，請使用此CreateKey作業。本區段中的範例會建立對稱加密 KMS 金鑰。這些範例中使用的 `Description` 參數是選用的。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

如需在 AWS KMS 主控台中建立 KMS 金鑰的說明，請參閱 建立金鑰。

Java

如需詳細資訊，請參閱《AWS SDK for Java API 參考》中的 createKey 方法。

```
// Create a KMS key
//
String desc = "Key for protecting critical data";

CreateKeyRequest req = new CreateKeyRequest().withDescription(desc);
CreateKeyResult result = kmsClient.createKey(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 CreateKey 方法。

```
// Create a KMS key
//
String desc = "Key for protecting critical data";

CreateKeyRequest req = new CreateKeyRequest()
{
    Description = desc
```

```
};
CreateKeyResponse response = kmsClient.CreateKey(req);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 create_key 方法。

```
# Create a KMS key

desc = 'Key for protecting critical data'

response = kms_client.create_key(
    Description=desc
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 create_key 執行個體方法。

```
# Create a KMS key

desc = 'Key for protecting critical data'

response = kmsClient.create_key({
  description: desc
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 CreateKey 方法。

```
// Create a KMS key
//
$desc = "Key for protecting critical data";

$result = $KmsClient->createKey([
    'Description' => $desc
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中 AWSSDK 中的 createKey 屬性。 JavaScript

```
// Create a KMS key
//
const Description = 'Key for protecting critical data';

kmsClient.createKey({ Description }, (err, data) => {
  ...
});
```

PowerShell

若要在中建立 KMS 金鑰 PowerShell，請使用新增KmsKey指令程式。

```
# Create a KMS key

$desc = 'Key for protecting critical data'
New-KmsKey -Description $desc
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《AWS Tools for Windows PowerShell 使用者指南》。

產生資料金鑰

若要產生對稱資料金鑰，請使用此作GenerateDataKey業。此操作會傳回純文字資料金鑰和在您指定的對稱加密 KMS 金鑰下加密的資料金鑰複本。您必須在每個命令中指定 KeySpec 或 NumberOfBytes (但不能同時指定兩者)。

如需使用資料金鑰來加密資料的說明，請參閱 AWS Encryption SDK。您也可以在 HMAC 操作中使用該資料金鑰。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

有關詳細信息，請參閱 AWS SDK for JavaAPI 參考中的generateDataKey 方法。

```
// Generate a data key
//
// Replace the following example key ARN with any valid key identfier
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
```

```
GenerateDataKeyRequest dataKeyRequest = new GenerateDataKeyRequest();
dataKeyRequest.setKeyId(keyId);
dataKeyRequest.setKeySpec("AES_256");

GenerateDataKeyResult dataKeyResult = kmsClient.generateDataKey(dataKeyRequest);

ByteBuffer plaintextKey = dataKeyResult.getPlaintext();

ByteBuffer encryptedKey = dataKeyResult.getCiphertextBlob();
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 GenerateDataKey 方法。

```
// Generate a data key
//
// Replace the following example key ARN with any valid key identfier
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
GenerateDataKeyRequest dataKeyRequest = new GenerateDataKeyRequest()
{
    KeyId = keyId,
    KeySpec = DataKeySpec.AES_256
};

GenerateDataKeyResponse dataKeyResponse = kmsClient.GenerateDataKey(dataKeyRequest);

MemoryStream plaintextKey = dataKeyResponse.Plaintext;

MemoryStream encryptedKey = dataKeyResponse.CiphertextBlob;
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 generate_data_key 方法。

```
# Generate a data key

# Replace the following example key ARN with any valid key identfier
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kms_client.generate_data_key(
```

```
    KeyId=key_id,
    KeySpec='AES_256'
)

plaintext_key = response['Plaintext']

encrypted_key = response['CiphertextBlob']
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 generate_data_key 執行個體方法。

```
# Generate a data key

# Replace the following example key ARN with any valid key identfier
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kmsClient.generate_data_key({
  key_id: key_id,
  key_spec: 'AES_256'
})

plaintext_key = response.plaintext

encrypted_key = response.ciphertext_blob
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 GenerateDataKey 方法。

```
// Generate a data key
//
// Replace the following example key ARN with any valid key identfier
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
$keySpec = 'AES_256';

$result = $KmsClient->generateDataKey([
    'KeyId' => $keyId,
    'KeySpec' => $keySpec,
]);
```

```
$plaintextKey = $result['Plaintext'];

$encryptedKey = $result['CiphertextBlob'];
```

Node.js

如需詳細資訊，請參閱 Node.js 中的 AWS SDK JavaScript 中的generateDataKey 屬性。

```
// Generate a data key
//
// Replace the following example key ARN with any valid key identfier
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
const KeySpec = 'AES_256';
kmsClient.generateDataKey({ KeyId, KeySpec }, (err, data) => {
  if (err) console.log(err, err.stack);
  else {
    const { CiphertextBlob, Plaintext } = data;
      ...
  }
});
```

PowerShell

若要產生對稱資料金鑰，請使用 New-KMS DataKey 指令程式。

在輸出中，純文本密鑰（在Plaintext屬性中）和加密密鑰（在CiphertextBlob屬性中）是MemoryStream對象。若要將它們轉換為字串，請使用MemoryStream類別的方法，或是將MemoryStream物件轉換為字串的指令程式或函數，例如 Convert 模組中的 ConvertFrom-MemoryStream 和 ConvertFrom-Base64 函數。

```
# Generate a data key

# Replace the following example key ARN with any valid key identfier
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
$keySpec = 'AES_256'

$response = New-KmsDataKey -KeyId $keyId -KeySpec $keySpec
$plaintextKey = $response.Plaintext
$encryptedKey = $response.CiphertextBlob
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

檢視 AWS KMS key

若要取得有關的詳細資訊AWS KMS key，包括 KMS 金鑰 ARN 和金鑰狀態，請使用DescribeKey作業。

DescribeKey 不會取得別名。若要取得別名，請使用ListAliases作業。如需範例，請參閱 處理別名。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

如需在 AWS KMS 主控台中檢視 KMS 金鑰的說明，請參閱 檢視金鑰。

Java

如需詳細資訊，請參閱《AWS SDK for Java API 參考》中的 describeKey 方法。

```
// Describe a KMS key
//
// Replace the following example key ARN with any valid key identfier
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

DescribeKeyRequest req = new DescribeKeyRequest().withKeyId(keyId);
DescribeKeyResult result = kmsClient.describeKey(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 DescribeKey 方法。

```
// Describe a KMS key
//
// Replace the following example key ARN with any valid key identfier
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

DescribeKeyRequest describeKeyRequest = new DescribeKeyRequest()
{
    KeyId = keyId
};
```

```
DescribeKeyResponse describeKeyResponse = kmsClient.DescribeKey(describeKeyRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 describe_key 方法。

```
# Describe a KMS key

# Replace the following example key ARN with any valid key identfier
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kms_client.describe_key(
    KeyId=key_id
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 describe_key 執行個體方法。

```
# Describe a KMS key

# Replace the following example key ARN with any valid key identfier
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kmsClient.describe_key({
  key_id: key_id
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 DescribeKey 方法。

```
// Describe a KMS key
//
// Replace the following example key ARN with any valid key identfier
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';

$result = $KmsClient->describeKey([
    'KeyId' => $keyId,
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中的 AWSSDK 中的 describeKey 屬性。 JavaScript

```
// Describe a KMS key
//
// Replace the following example key ARN with any valid key identfier
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
kmsClient.describeKey({ KeyId }, (err, data) => {
  ...
});
```

PowerShell

若要取得 KMS 金鑰的詳細資訊，請使用 G et-KmsKey 指令程式。

```
# Describe a KMS key

# Replace the following example key ARN with any valid key identfier
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
Get-KmsKey -KeyId $keyId
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

取得 KMS 金鑰的金鑰 ID 和金鑰 ARN

若要取得的金鑰 ID 和金鑰 ARNAWS KMS keys，請使用ListKeys作業。這些範例會使用選用的 `Limit` 參數，此參數會設定每次呼叫中傳回的 KMS 金鑰數目上限。如需識別 AWS KMS 操作中 KMS 金鑰的說明，請參閱 金鑰識別碼 (KeyId)。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

如需在 AWS KMS 主控台中尋找金鑰 ID 和金鑰 ARN 的協助，請參閱尋找金鑰 ID 和金鑰 ARN。

Java

如需詳細資訊，請參閱《AWS SDK for Java API 參考》中的 listKeys 方法。

```
// List KMS keys in this account
```

```
//
Integer limit = 10;

ListKeysRequest req = new ListKeysRequest().withLimit(limit);
ListKeysResult result = kmsClient.listKeys(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 ListKeys 方法。

```
// List KMS keys in this account
//
int limit = 10;

ListKeysRequest listKeysRequest = new ListKeysRequest()
{
    Limit = limit
};
ListKeysResponse listKeysResponse = kmsClient.ListKeys(listKeysRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 `list_keys` 方法。

```
# List KMS keys in this account

response = kms_client.list_keys(
    Limit=10
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 `list_keys` 執行個體方法。

```
# List KMS keys in this account

response = kmsClient.list_keys({
  limit: 10
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 ListKeys 方法。

```
// List KMS keys in this account
//
$limit = 10;

$result = $KmsClient->listKeys([
    'Limit' => $limit,
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中 AWS SDK 中的 listKeys 屬性。 JavaScript

```
// List KMS keys in this account
//
const Limit = 10;
kmsClient.listKeys({ Limit }, (err, data) => {
  ...
});
```

PowerShell

若要取得帳戶和區域中所有 KMS 金鑰的金鑰識別碼和金鑰 ARN，請使用 G et-KmsKeyList 指令程式。

若要限制輸出物件的數量，此範例使用 Select-Object Cmdlet，而不是在清單 Cmdlet 中已取代的 `Limit` 參數。如需 AWS Tools for PowerShell 中分頁輸出的說明，請參閱使用 AWS Tools for PowerShell 輸出分頁。

```
# List KMS keys in this account

$limit = 10
Get-KmsKeyList | Select-Object -First $limit
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

啟用 AWS KMS keys

要啟用禁用AWS KMS key，請使用該EnableKey操作。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

如需在 AWS KMS 主控台中啟用和停用 KMS 金鑰的說明，請參閱 啟用和停用金鑰。

Java

如需 Java 實作的詳細資訊，請參閱《AWS SDK for Java API 參考》中的 enableKey 方法。

```
// Enable a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

EnableKeyRequest req = new EnableKeyRequest().withKeyId(keyId);
kmsClient.enableKey(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 EnableKey 方法。

```
// Enable a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

EnableKeyRequest enableKeyRequest = new EnableKeyRequest()
{
    KeyId = keyId
};
kmsClient.EnableKey(enableKeyRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 enable_key 方法。

```
# Enable a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kms_client.enable_key(
    KeyId=key_id
```

```
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 enable_key 執行個體方法。

```
# Enable a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kmsClient.enable_key({
  key_id: key_id
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 EnableKey 方法。

```
// Enable a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';

$result = $KmsClient->enableKey([
    'KeyId' => $keyId,
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中的 AWSSDK 中的 enableKey 屬性。 JavaScript

```
// Enable a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
kmsClient.enableKey({ KeyId }, (err, data) => {
  ...
});
```

PowerShell

若要啟用 KMS 金鑰，請使用啟用KmsKey指令程式。

```
# Enable a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
Enable-KmsKey -KeyId $keyId
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

停用 AWS KMS key

若要停用 KMS 金鑰，請使用此DisableKey作業。停用 KMS 金鑰可防止其在密碼編譯操作中被使用。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

如需在 AWS KMS 主控台中啟用和停用 KMS 金鑰的說明，請參閱 啟用和停用金鑰。

Java

如需詳細資訊，請參閱《AWS SDK for Java API 參考》中的 disableKey 方法。

```
// Disable a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

DisableKeyRequest req = new DisableKeyRequest().withKeyId(keyId);
kmsClient.disableKey(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 DisableKey 方法。

```
// Disable a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
```

```
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

DisableKeyRequest disableKeyRequest = new DisableKeyRequest()
{
    KeyId = keyId
};
kmsClient.DisableKey(disableKeyRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 disable_key 方法。

```
# Disable a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kms_client.disable_key(
    KeyId=key_id
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 disable_key 執行個體方法。

```
# Disable a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kmsClient.disable_key({
  key_id: key_id
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 DisableKey 方法。

```
// Disable a KMS key
//
```

```
// Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';

$result = $KmsClient->disableKey([
    'KeyId' => $keyId,
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中的 AWSSDK 中的 disableKey 屬性。 JavaScript

```
// Disable a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
kmsClient.disableKey({ KeyId }, (err, data) => {
  ...
});
```

PowerShell

若要停用 KMS 金鑰，請使用停用KmsKey指令程式。

```
# Disable a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
Disable-KmsKey -KeyId $keyId
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《AWS Tools for Windows PowerShell 使用者指南》。

處理別名

此主題中的範例使用 AWS KMS API 建立、檢視、更新和刪除別名。如需關於別名的資訊，請參閱 the section called “使用別名”。

主題

- 建立別名
- 列出別名
- 更新別名
- 刪除別名

建立別名

在 AWS Management Console 中建立 AWS KMS key 時，您必須為其建立別名。不過，建立 KMS 金鑰的CreateKey作業不會建立別名。

若要建立別名，請使用此CreateAlias作業。別名在帳戶和區域中必須是唯一的。您無法建立開頭為 aws/ 的別名。Amazon Web Services 會為 AWS 受管金鑰 保留 aws/ 字首。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

如需詳細資訊，請參閱 AWS SDK for Java API 參考中的 createAlias 方法。

```
// Create an alias for a KMS key
//
String aliasName = "alias/projectKey1";
// Replace the following example key ARN with a valid key ID or key ARN
String targetKeyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

CreateAliasRequest req = new
 CreateAliasRequest().withAliasName(aliasName).withTargetKeyId(targetKeyId);
kmsClient.createAlias(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 CreateAlias 方法。

```
// Create an alias for a KMS key
//
String aliasName = "alias/projectKey1";
// Replace the following example key ARN with a valid key ID or key ARN
String targetKeyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
```

```
CreateAliasRequest createAliasRequest = new CreateAliasRequest()
{
    AliasName = aliasName,
    TargetKeyId = targetKeyId
};
kmsClient.CreateAlias(createAliasRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 create_alias 方法。

```
# Create an alias for a KMS key

alias_name = 'alias/projectKey1'
# Replace the following example key ARN with a valid key ID or key ARN
target_key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kms_client.create_alias(
    AliasName=alias_name,
    TargetKeyId=key_id
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 create_alias 執行個體方法。

```
# Create an alias for a KMS key

alias_name = 'alias/projectKey1'
# Replace the following example key ARN with a valid key ID or key ARN
target_key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kmsClient.create_alias({
  alias_name: alias_name,
  target_key_id: target_key_id
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 CreateAlias 方法。

```
// Create an alias for a KMS key
//
$aliasName = "alias/projectKey1";
// Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';


$result = $KmsClient->createAlias([
    'AliasName' => $aliasName,
    'TargetKeyId' =>  $keyId,
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中 AWSSDK 中的 createAlias 屬性。 JavaScript

```
// Create an alias for a KMS key
//
const AliasName = 'alias/projectKey1';

// Replace the following example key ARN with a valid key ID or key ARN
const TargetKeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
kmsClient.createAlias({ AliasName, TargetKeyId }, (err, data) => {
  ...
});
```

PowerShell

若要建立別名，請使用 New-KMSAlias Cmdlet。別名名稱區分大小寫。

```
# Create an alias for a KMS key

$aliasName = 'alias/projectKey1'
# Replace the following example key ARN with a valid key ID or key ARN
$targetKeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

New-KMSAlias -TargetKeyId $targetKeyId -AliasName $aliasName
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

列出別名

若要列出帳戶和區域中的別名，請使用此ListAliases作業。

在預設情況下，ListAliases 命令會傳回在帳戶和區域中的所有別名。這包括您建立的別名，而且是與客戶受管金鑰相關聯的別名，以及 AWS 建立且與您 AWS 受管金鑰 相關聯的別名。回應可能也包含沒有 `TargetKeyId` 欄位的別名。這些是 AWS 已建立但尚未關聯至 KMS 金鑰的預先定義別名。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

如需 Java 實作的詳細資訊，請參閱《AWS SDK for Java API 參考》中的 listAliases 方法。

```
// List the aliases in this AWS ##
//
Integer limit = 10;

ListAliasesRequest req = new ListAliasesRequest().withLimit(limit);
ListAliasesResult result = kmsClient.listAliases(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 ListAliases 方法。

```
// List the aliases in this AWS ##
//
int limit = 10;

ListAliasesRequest listAliasesRequest = new ListAliasesRequest()
{
    Limit = limit
};
ListAliasesResponse listAliasesResponse = kmsClient.ListAliases(listAliasesRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 `list_aliases` 方法。

```
# List the aliases in this AWS ##

response = kms_client.list_aliases(
    Limit=10
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 `list_aliases` 執行個體方法。

```
# List the aliases in this AWS ##

response = kmsClient.list_aliases({
  limit: 10
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 List Aliases 方法。

```
// List the aliases in this AWS ##
//
$limit = 10;

$result = $KmsClient->listAliases([
    'Limit' => $limit,
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中 AWS SDK 中的 [listAliases] 屬性。 JavaScript

```
// List the aliases in this AWS ##
//
const Limit = 10;
kmsClient.listAliases({ Limit }, (err, data) => {
  ...
});
```

PowerShell

若要列出帳戶和區域中的別名，請使用 Get-KMS AliasList 指令程式。

若要限制輸出物件的數量，此範例使用 Select-Object Cmdlet，而不是在清單 Cmdlet 中已取代的 `Limit` 參數。如需 AWS Tools for PowerShell 中分頁輸出的說明，請參閱使用 AWS Tools for PowerShell 輸出分頁。

```
# List the aliases in this AWS ##
$limit = 10

$result = Get-KMSAliasList | Select-Object -First $limit
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

若要列出與特定 KMS 金鑰關聯的別名，請使用 `KeyId` 參數。它的值可以是區域中任何 KMS 金鑰的金鑰 ID 或金鑰 ARN。您不能指定別名名稱或別名 ARN。

Java

如需 Java 實作的詳細資訊，請參閱《AWS SDK for Java API 參考》中的 listAliases 方法。

```
// List the aliases for one KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

ListAliasesRequest req = new ListAliasesRequest().withKeyId(keyId);
ListAliasesResult result = kmsClient.listAliases(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 ListAliases 方法。

```
// List the aliases for one KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

ListAliasesRequest listAliasesRequest = new ListAliasesRequest()
{
    KeyId = keyId
```

```
};
ListAliasesResponse listAliasesResponse = kmsClient.ListAliases(listAliasesRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 list_aliases 方法。

```
# List the aliases for one KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kms_client.list_aliases(
    KeyId=key_id
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 list_aliases 執行個體方法。

```
# List the aliases for one KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kmsClient.list_aliases({
  key_id: key_id
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 List Aliases 方法。

```
// List the aliases for one KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';

$result = $KmsClient->listAliases([
```

```
    'KeyId' => $keyId,
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中 AWS SDK 中的 [listAliases] 屬性。 JavaScript

```
// List the aliases for one KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
kmsClient.listAliases({ KeyId }, (err, data) => {
  ...
});
```

PowerShell

若要列出 KMS 金鑰的別名，請使用 Get-KMS AliasList 指令程式的KeyId參數。

```
# List the aliases for one KMS key

# Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

$response = Get-KmsAliasList -KeyId $keyId
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

更新別名

若要將現有別名與不同的 KMS 金鑰建立關聯，請使用此UpdateAlias作業。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

如需 Java 實作的詳細資訊，請參閱《AWS SDK for Java API 參考》中的 updateAlias 方法。

```
// Updating an alias
//
String aliasName = "alias/projectKey1";
// Replace the following example key ARN with a valid key ID or key ARN
String targetKeyId = "arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321";

UpdateAliasRequest req = new UpdateAliasRequest()
     .withAliasName(aliasName)
     .withTargetKeyId(targetKeyId);

kmsClient.updateAlias(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 UpdateAlias 方法。

```
// Updating an alias
//
String aliasName = "alias/projectKey1";
// Replace the following example key ARN with a valid key ID or key ARN
String targetKeyId = "arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321";

UpdateAliasRequest updateAliasRequest = new UpdateAliasRequest()
{
    AliasName = aliasName,
    TargetKeyId = targetKeyId
};

kmsClient.UpdateAlias(updateAliasRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 update_alias 方法。

```
# Updating an alias

alias_name = 'alias/projectKey1'
# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321'
```

```
response = kms_client.update_alias(
    AliasName=alias_name,
    TargetKeyID=key_id
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 update_alias 執行個體方法。

```
# Updating an alias

alias_name = 'alias/projectKey1'
# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321'

response = kmsClient.update_alias({
  alias_name: alias_name,
  target_key_id: key_id
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 UpdateAlias 方法。

```
// Updating an alias
//
$aliasName = "alias/projectKey1";

// Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321';

$result = $KmsClient->updateAlias([
    'AliasName' => $aliasName,
    'TargetKeyId' =>  $keyId,
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中 AWS SDK 中的 updateAlias 屬性。 JavaScript

```
// Updating an alias
//
const AliasName = 'alias/projectKey1';

// Replace the following example key ARN with a valid key ID or key ARN
const TargetKeyId = 'arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321';
kmsClient.updateAlias({ AliasName, TargetKeyId }, (err, data) => {
  ...
});
```

PowerShell

若要變更與別名相關聯的 KMS 金鑰，請使用 Update-KMSAlias cmdlet。別名名稱區分大小寫。

此 `Update-KMSAlias` Cmdlet 不會傳回任何輸出。若要確認命令是否有效，請使用 Get-KMS AliasList 指令程式。

```
# Updating an alias

$aliasName = 'alias/projectKey1'
# Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321'

Update-KMSAlias -AliasName $aliasName -TargetKeyID $keyId
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

刪除別名

若要刪除別名，請使用此DeleteAlias作業。刪除別名並不會影響相關聯的 KMS 金鑰。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

如需詳細資訊，請參閱 AWS SDK for Java API 參考中的 deleteAlias 方法。

```
// Delete an alias for a KMS key
```

```
//
String aliasName = "alias/projectKey1";

DeleteAliasRequest req  = new DeleteAliasRequest().withAliasName(aliasName);
kmsClient.deleteAlias(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 DeleteAlias 方法。

```
// Delete an alias for a KMS key
//
String aliasName = "alias/projectKey1";

DeleteAliasRequest deleteAliasRequest = new DeleteAliasRequest()
{
    AliasName = aliasName
};
kmsClient.DeleteAlias(deleteAliasRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 delete_alias 方法。

```
# Delete an alias for a KMS key

alias_name = 'alias/projectKey1'

response = kms_client.delete_alias(
    AliasName=alias_name
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 delete_alias 執行個體方法。

```
# Delete an alias for a KMS key

alias_name = 'alias/projectKey1'

response = kmsClient.delete_alias({
  alias_name: alias_name
```

```
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 DeleteAlias 方法。

```
// Delete an alias for a KMS key
//
$aliasName = "alias/projectKey1";

$result = $KmsClient->deleteAlias([
    'AliasName' => $aliasName,
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中的 AWSSDK 中的 deleteAlias 屬性。 JavaScript

```
// Delete an alias for a KMS key
//
const AliasName = 'alias/projectKey1';
kmsClient.deleteAlias({ AliasName }, (err, data) => {
  ...
});
```

PowerShell

若要刪除別名，請使用 Remove-KMSAlias Cmdlet。別名名稱區分大小寫。

由於此指令程式會永久刪除別名，因此 PowerShell 會提示您確認命令。`ConfirmImpact` 是 `High`，因此您無法使用 `ConfirmPreference` 來抑制此提示。如果您必須抑制確認提示，請新增具有 `$false` 值的 `Confirm` 常用參數，例如：`-Confirm:$false`。

此 `Remove-KMSAlias` Cmdlet 不會傳回任何輸出。若要確認命令是否有效，請使用 Get-KMSAliasList 指令程式。

```
# Delete an alias for a KMS key

$aliasName = 'alias/projectKey1'
Remove-KMSAlias -AliasName $aliasName
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《AWS Tools for Windows PowerShell 使用者指南》。

加密和解密資料金鑰

本主題中的範例使用 AWS KMS API 中的「加密」、「解密」和「ReEncrypt作業」。

這些操作旨在加密和解密資料金鑰。它們在加密操作中使用 AWS KMS keys，且無法接受超過 4 KB (4096 個位元組) 的資料。雖然您可以使用它們來加密少量資料，例如密碼或 RSA 金鑰，但它們不是為加密應用程式資料而設計的。

若要加密應用程式資料，請使用 AWS 服務的伺服器端加密功能或用戶端加密程式庫，例如 AWS Encryption SDK 或 Amazon S3 加密用戶端。

主題

- 加密資料金鑰
- 解密資料金鑰
- 以不同的 AWS KMS key 重新加密資料金鑰

加密資料金鑰

加密操作旨在加密資料金鑰，但不常使用。GenerateDataKey和作GenerateDataKeyWithoutPlaintext業會傳回加密的資料金鑰。當您要將加密的資料移到不同區域或是想使用新區域的 KMS 金鑰加密其資料金鑰，可以使用此方法。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

如需詳細資訊，請參閱《AWS SDK for Java API 參考》中的 encrypt 方法。

```
// Encrypt a data key
//
// Replace the following example key ARN with any valid key identfier
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
ByteBuffer plaintext = ByteBuffer.wrap(new byte[]{1,2,3,4,5,6,7,8,9,0});
```

```
EncryptRequest req = new EncryptRequest().withKeyId(keyId).withPlaintext(plaintext);
ByteBuffer ciphertext = kmsClient.encrypt(req).getCiphertextBlob();
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 Encrypt 方法。

```
// Encrypt a data key
//
// Replace the following example key ARN with any valid key identfier
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
MemoryStream plaintext = new MemoryStream();
plaintext.Write(new byte[] { 1, 2, 3, 4, 5, 6, 7, 8, 9, 0 }, 0, 10);

EncryptRequest encryptRequest = new EncryptRequest()
{
    KeyId = keyId,
    Plaintext = plaintext
};
MemoryStream ciphertext = kmsClient.Encrypt(encryptRequest).CiphertextBlob;
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 encrypt 方法。

```
# Encrypt a data key

# Replace the following example key ARN with any valid key identfier
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
plaintext = b'\x01\x02\x03\x04\x05\x06\x07\x08\x09\x00'

response = kms_client.encrypt(
    KeyId=key_id,
    Plaintext=plaintext
)

ciphertext = response['CiphertextBlob']
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 encrypt 執行個體方法。

```
# Encrypt a data key

# Replace the following example key ARN with any valid key identfier
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
plaintext = "\x01\x02\x03\x04\x05\x06\x07\x08\x09\x00"

response = kmsClient.encrypt({
  key_id: key_id,
  plaintext: plaintext
})

ciphertext = response.ciphertext_blob
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 Encrypt 方法。

```
// Encrypt a data key
//
// Replace the following example key ARN with any valid key identfier
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
$message = pack('c*',1,2,3,4,5,6,7,8,9,0);

$result = $KmsClient->encrypt([
    'KeyId' => $keyId,
    'Plaintext' => $message,
]);

$ciphertext = $result['CiphertextBlob'];
```

Node.js

如需詳細資訊，請參閱 Node.js 中的 AWS SDK JavaScript 中的加密屬性。

```
// Encrypt a data key
//
// Replace the following example key ARN with any valid key identfier
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
const Plaintext = Buffer.from([1, 2, 3, 4, 5, 6, 7, 8, 9, 0]);
kmsClient.encrypt({ KeyId, Plaintext }, (err, data) => {
```

```
  if (err) console.log(err, err.stack); // an error occurred
  else {
    const { CiphertextBlob } = data;
      ...
  }
});
```

PowerShell

若要加密 KMS 金鑰下的資料金鑰，請使用 Invoke-KMSEncrypt cmdlet。它返回密文作為一個MemoryStream (系統。IO。 MemoryStream) 物件。您可以使用 MemoryStream 物件做為 Invoke-KMSDecrypt Cmdlet 的輸入。

AWS KMS 也傳回資料金鑰作為 MemoryStream 物件。在這個例子中，為了模擬純文字資料金鑰，我們建立一個位元組陣列，並將其寫入 MemoryStream 物件。

請注意，Invoke-KMSEncrypt 的 Plaintext 參數採用位元組陣列 (byte[])；此參數不需要 MemoryStream 物件。從 4.0 AWSPowerShell 版開始，採用字節數組和MemoryStream對象的所有 AWSPowerShell 模塊中的參數都接受字節數組，MemoryStream對象，字符串，字符串數組和FileInfo (System.io。 FileInfo) 物件。您可以將這些類型中的任何一項傳遞給 Invoke-KMSEncrypt。

```
# Encrypt a data key

# Replace the following example key ARN with any valid key identfier
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

# Simulate a data key
 # Create a byte array
[byte[]] $bytes = 1, 2, 3, 4, 5, 6, 7, 8, 9, 0

 # Create a MemoryStream
$plaintext = [System.IO.MemoryStream]::new()

 # Add the byte array to the MemoryStream
$plaintext.Write($bytes, 0, $bytes.length)

# Encrypt the simulated data key
$response = Invoke-KMSEncrypt -KeyId $keyId -Plaintext $plaintext

# Get the ciphertext from the response
```

```
$ciphertext = $response.CiphertextBlob
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

解密資料金鑰

若要解密資料金鑰，請使用 Decrypt 操作。

您指定的必須ciphertextBlob是GenerateDataKey、GenerateDataKeyWithoutPlaintext或「加密」回應中的CiphertextBlob欄位值，或來自或回GenerateDataKeyPairWithoutPlaintext應的PrivateKeyCiphertextBlobGenerateDataKeyPair欄位值。您也可以使用此 Decrypt 操作來解密非對稱 KMS 金鑰中 AWS KMS 以外由公有金鑰加密的資料。

使用對稱加密 KMS 金鑰進行解密時，不需要 KeyId 參數。AWS KMS 可以從加密文字 Blob 的中繼資料取得用來加密資料的 KMS 金鑰。但是指定您正在使用的 KMS 金鑰永遠是最佳實務。此實務可確保您使用預定的 KMS 金鑰，並防止您意外使用您不信任的 KMS 金鑰來解密加密文字。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

如需詳細資訊，請參閱《AWS SDK for Java API 參考》中的 decrypt 方法。

```
// Decrypt a data key
//
// Replace the following example key ARN with any valid key identfier
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

ByteBuffer ciphertextBlob = Place your ciphertext here;

DecryptRequest req = new
 DecryptRequest().withCiphertextBlob(ciphertextBlob).withKeyId(keyId);
ByteBuffer plainText = kmsClient.decrypt(req).getPlaintext();
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 Decrypt 方法。

```
// Decrypt a data key
//
```

```
// Replace the following example key ARN with any valid key identfier
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

MemoryStream ciphertextBlob = new MemoryStream();
// Write ciphertext to memory stream

DecryptRequest decryptRequest = new DecryptRequest()
{
    CiphertextBlob = ciphertextBlob,
    KeyId = keyId
};
MemoryStream plainText = kmsClient.Decrypt(decryptRequest).Plaintext;
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 decrypt 方法。

```
# Decrypt a data key

# Replace the following example key ARN with any valid key identfier
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
ciphertext = 'Place your ciphertext here'

response = kms_client.decrypt(
    CiphertextBlob=ciphertext,
    KeyId=key_id
)

plaintext = response['Plaintext']
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 decrypt 執行個體方法。

```
# Decrypt a data key

# Replace the following example key ARN with any valid key identfier
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

ciphertext = 'Place your ciphertext here'
```

```
ciphertext_packed = [ciphertext].pack("H*")

response = kmsClient.decrypt({
  ciphertext_blob: ciphertext_packed,
  key_id: key_id
})

plaintext = response.plaintext
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 Decrypt 方法。

```
// Decrypt a data key
//
// Replace the following example key ARN with any valid key identfier
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
$ciphertext = 'Place your cipher text blob here';

$result = $KmsClient->decrypt([
     'CiphertextBlob' => $ciphertext,
     'KeyId' => $keyId,
]);

$plaintext = $result['Plaintext'];
```

Node.js

如需詳細資訊，請參閱 Node.js 中的 AWS SDK JavaScript 中的解密屬性。

```
// Decrypt a data key
//
// Replace the following example key ARN with any valid key identfier
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
const CiphertextBlob = 'Place your cipher text blob here';
kmsClient.decrypt({ CiphertextBlob, KeyId }, (err, data) => {
  if (err) console.log(err, err.stack); // an error occurred
  else {
    const { Plaintext } = data;
    ...
  }
```

```
});
```

PowerShell

若要解密資料金鑰，請使用 Invoke-KMSEncrypt Cmdlet。

此指令程式會傳回純文字為 (System.io. MemoryStream MemoryStream) 物件。若要將其轉換為位元組陣列，請使用 Cmdlet 或將 MemoryStream 物件轉換為位元組陣列的函數，例如 Convert 模組中的函數。

因為此範例會使用 AWS KMS 加密 cmdlet 傳回的加密文字，所以會使用 MemoryStream 物件做為 CiphertextBlob 參數的值。但是，Invoke-KMSDecrypt 的 CiphertextBlob 參數會採用位元組陣列 (byte[])；此參數不需要 MemoryStream 物件。從 4.0 AWSPowerShell 版開始，採用字節數組和MemoryStream對象的所有 AWSPowerShell 模塊中的參數都接受字節數組，MemoryStream對象，字符串，字符串數組和FileInfo (System.io。 FileInfo) 物件。您可以將這些類型中的任何一項傳遞給 Invoke-KMSDecrypt。

```
# Decrypt a data key
# Replace the following example key ARN with any valid key identfier
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

[System.IO.MemoryStream]$ciphertext = Read-Host 'Place your cipher text blob here'

$response = Invoke-KMSDecrypt -CiphertextBlob $ciphertext -KeyId $keyId
$plaintext = $response.Plaintext
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

以不同的 AWS KMS key 重新加密資料金鑰

若要解密加密的資料金鑰，然後立即在其他金鑰下重新加密資料金鑰AWS KMS key，請使用此ReEncrypt作業。此操作完全在伺服器端的 AWS KMS 中執行，所以絕不會在 AWS KMS 外部公開您的純文字。

您指定的必須ciphertextBlob是GenerateDataKey、GenerateDataKeyWithoutPlaintext或「加密」回應中的CiphertextBlob欄位值，或來自或回GenerateDataKeyPairWithoutPlaintext應的PrivateKeyCiphertextBlobGenerateDataKeyPair欄位值。您也可以使用此 ReEncrypt 操作來重新加密非對稱 KMS 金鑰中 AWS KMS 以外由公有金鑰加密的資料。

使用對稱加密 KMS 金鑰進行重新加密時，不需要 SourceKeyId 參數。AWS KMS 可以從加密文字 Blob 的中繼資料取得用來加密資料的 KMS 金鑰。但是指定您正在使用的 KMS 金鑰永遠是最佳實務。此實務可確保您使用預定的 KMS 金鑰，並防止您意外使用您不信任的 KMS 金鑰來解密加密文字。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

如需詳細資訊，請參閱《AWS SDK for Java API 參考》中的 reEncrypt 方法。

```
// Re-encrypt a data key

ByteBuffer sourceCiphertextBlob = Place your ciphertext here;

// Replace the following example key ARNs with valid key identfiers
String sourceKeyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
String destinationKeyId = "arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321";

ReEncryptRequest req = new ReEncryptRequest();
req.setCiphertextBlob(sourceCiphertextBlob);
req.setSourceKeyId(sourceKeyId);
req.setDestinationKeyId(destinationKeyId);
ByteBuffer destinationCipherTextBlob = kmsClient.reEncrypt(req).getCiphertextBlob();
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 ReEncrypt 方法。

```
// Re-encrypt a data key

MemoryStream sourceCiphertextBlob = new MemoryStream();
// Write ciphertext to memory stream

// Replace the following example key ARNs with valid key identfiers
String sourceKeyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
String destinationKeyId = "arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321";

ReEncryptRequest reEncryptRequest = new ReEncryptRequest()
{
```

```
    CiphertextBlob = sourceCiphertextBlob,
    SourceKeyId = sourceKeyId,
    DestinationKeyId = destinationKeyId
};
MemoryStream destinationCipherTextBlob =
 kmsClient.ReEncrypt(reEncryptRequest).CiphertextBlob;
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 re_encrypt 方法。

```
# Re-encrypt a data key
ciphertext = 'Place your ciphertext here'

# Replace the following example key ARNs with valid key identfiers
source_key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
destination_key_id = 'arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321'

response = kms_client.re_encrypt(
    CiphertextBlob=ciphertext,
    SourceKeyId=source_key_id,
    DestinationKeyId=destination_key_id
)

destination_ciphertext_blob = response['CiphertextBlob']
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 re_encrypt 執行個體方法。

```
# Re-encrypt a data key

ciphertext = 'Place your ciphertext here'
ciphertext_packed = [ciphertext].pack("H*")

# Replace the following example key ARNs with valid key identfiers
source_key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
destination_key_id = 'arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321'
```

```
response = kmsClient.re_encrypt({
  ciphertext_blob: ciphertext_packed,
  source_key_id: source_key_id,
  destination_key_id: destination_key_id
})

destination_ciphertext_blob = response.ciphertext_blob.unpack('H*')
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 ReEncrypt 方法。

```
// Re-encrypt a data key

$ciphertextBlob = 'Place your ciphertext here';

// Replace the following example key ARNs with valid key identfiers
$sourceKeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
$destinationKeyId = 'arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321';

$result = $KmsClient->reEncrypt([
    'CiphertextBlob' => $ciphertextBlob,
    'SourceKeyId' => $sourceKeyId,
    'DestinationKeyId' => $destinationKeyId,
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中 AWS SDK 中的 reEncrypt 屬性。 JavaScript

```
// Re-encrypt a data key
const CiphertextBlob = 'Place your cipher text blob here';
// Replace the following example key ARNs with valid key identfiers
const SourceKeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
const DestinationKeyId = 'arn:aws:kms:us-
west-2:111122223333:key/0987dcba-09fe-87dc-65ba-ab0987654321';

kmsClient.reEncrypt({ CiphertextBlob, SourceKeyId, DestinationKeyId }, (err, data)
 => {
```

```
  ...
});
```

PowerShell

若要重新加密相同或不同 KMS 金鑰下的密文，請使用叫用 KMS 指令程式。ReEncrypt

因為此範例會使用 AWS KMS 加密 cmdlet 傳回的加密文字，所以會使用 MemoryStream 物件做為 CiphertextBlob 參數的值。但是，Invoke-KMSReEncrypt 的 CiphertextBlob 參數會採用位元組陣列 (byte[])；此參數不需要 MemoryStream 物件。從 4.0 AWSPowerShell 版開始，採用字節數組和MemoryStream對象的所有 AWSPowerShell 模塊中的參數都接受字節數組，MemoryStream對象，字符串，字符串數組和FileInfo (System.io。 FileInfo) 物件。您可以將這些類型中的任何一項傳遞給 Invoke-KMSReEncrypt。

```
# Re-encrypt a data key

[System.IO.MemoryStream]$ciphertextBlob = Read-Host 'Place your cipher text blob
 here'

# Replace the following example key ARNs with valid key identfiers
$sourceKeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
$destinationKeyId = 'arn:aws:kms:us-west-2:111122223333:key/0987dcba-09fe-87dc-65ba-
ab0987654321'

$response = Invoke-KMSReEncrypt -Ciphertext $ciphertextBlob -SourceKeyId
 $sourceKeyId -DestinationKeyId $destinationKeyId
$reEncryptedCiphertext = $response.CiphertextBlob
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《AWS Tools for Windows PowerShell 使用者指南》。

處理金鑰政策

此主題中的範例使用 AWS KMS API 來檢視和變更 AWS KMS keys 的金鑰政策。

如需有關如何使用金鑰政策、IAM 政策和授予以管理 KMS 金鑰存取的詳細資訊，請參閱 AWS KMS 的身分驗證與存取控制。如需撰寫及格式化 JSON 政策文件的說明，請參閱《IAM 使用者指南》中的 IAM JSON 政策參考。

主題

- 列出金鑰政策名稱
- 取得金鑰政策
- 設定金鑰政策

列出金鑰政策名稱

若要取得的金鑰原則名稱AWS KMS key，請使用ListKeyPolicies作業。它傳回的唯一金鑰政策名稱為 default。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

有關 Java 實現的詳細信息，請參閱 AWS SDK for JavaAPI 參考中的listKeyPolicies 方法。

```
// List key policies
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

ListKeyPoliciesRequest req = new ListKeyPoliciesRequest().withKeyId(keyId);
ListKeyPoliciesResult result = kmsClient.listKeyPolicies(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 ListKeyPolicies 方法。

```
// List key policies
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

ListKeyPoliciesRequest listKeyPoliciesRequest = new ListKeyPoliciesRequest()
{
    KeyId = keyId
};
ListKeyPoliciesResponse listKeyPoliciesResponse =
 kmsClient.ListKeyPolicies(listKeyPoliciesRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 list_key_policies 方法。

```
# List key policies

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kms_client.list_key_policies(
    KeyId=key_id
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 list_key_policies 執行個體方法。

```
# List key policies

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kmsClient.list_key_policies({
  key_id: key_id
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 ListKeyPolicies 方法。

```
// List key policies
//
// Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';

$result = $KmsClient->listKeyPolicies([
    'KeyId' => $keyId
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中適用之 AWS SDK JavaScript 中的listKeyPolicies 屬性。

```
// List key policies
//
// Replace the following example key ARN with a valid key ID or key ARN
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';

kmsClient.listKeyPolicies({ KeyId }, (err, data) => {
  ...
});
```

PowerShell

若要列出預設金鑰原則的名稱，請使用 Get-KMS 指令KeyPolicyList程式。

```
# List key policies

# Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
$response = Get-KMSKeyPolicyList -KeyId $keyId
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

取得金鑰政策

若要取得的金鑰原則AWS KMS key，請使用GetKeyPolicy作業。

GetKeyPolicy 需要策略名稱。唯一的有效政策名稱為 default。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

有關詳細信息，請參閱 AWS SDK for JavaAPI 參考中的getKeyPolicy 方法。

```
// Get the policy for a KMS key
//
```

```
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
String policyName = "default";

GetKeyPolicyRequest req = new
 GetKeyPolicyRequest().withKeyId(keyId).withPolicyName(policyName);
GetKeyPolicyResult result = kmsClient.getKeyPolicy(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 GetKeyPolicy 方法。

```
// Get the policy for a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
String policyName = "default";

GetKeyPolicyRequest getKeyPolicyRequest = new GetKeyPolicyRequest()
{
    KeyId = keyId,
    PolicyName = policyName
};
GetKeyPolicyResponse getKeyPolicyResponse =
 kmsClient.GetKeyPolicy(getKeyPolicyRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 get_key_policy 方法。

```
# Get the policy for a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
policy_name = 'default'

response = kms_client.get_key_policy(
    KeyId=key_id,
    PolicyName=policy_name
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 get_key_policy 執行個體方法。

```
# Get the policy for a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
policy_name = 'default'

response = kmsClient.get_key_policy({
  key_id: key_id,
  policy_name: policy_name
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 GetKeyPolicy 方法。

```
// Get the policy for a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
$policyName = "default";

$result = $KmsClient->getKeyPolicy([
    'KeyId' => $keyId,
    'PolicyName' => $policyName
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中適用之 AWS SDK JavaScript 中的getKeyPolicy 屬性。

```
// Get the policy for a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
const PolicyName = 'default';
kmsClient.getKeyPolicy({ KeyId, PolicyName }, (err, data) => {
```

```
  ...
});
```

PowerShell

若要取得 KMS 金鑰的金鑰原則，請使用 Get-KMS 指令KeyPolicy程式。此指令程式會以字串 (System.String) 的形式傳回金鑰原則，您可以在 Write-KeyPolicy KMS () 命令中使用。PutKeyPolicy若要將 JSON 字串中的原則轉換為PSCustomObject物件，請使用 ConvertFrom-JSON 指令程式。

```
# Get the policy for a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
$policyName = 'default'

$response = Get-KMSKeyPolicy -KeyId $keyId -PolicyName $policyName
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

設定金鑰政策

若要建立或取代 KMS 金鑰的金鑰原則，請使用此PutKeyPolicy作業。

PutKeyPolicy 需要政策名稱。唯一的有效政策名稱為 default。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

有關詳細信息，請參閱 AWS SDK for JavaAPI 參考中的putKeyPolicy 方法。

```
// Set a key policy for a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
String policyName = "default";
String policy = "{" +
```

```
                "  \"Version\": \"2012-10-17\"," +
                "  \"Statement\": [{" +
                "    \"Sid\": \"Allow access for ExampleRole\"," +
                "    \"Effect\": \"Allow\"," +
                // Replace the following example user ARN with a valid one
                "    \"Principal\": {\"AWS\": \"arn:aws:iam::111122223333:role/
ExampleKeyUserRole\"}," +
                "    \"Action\": [" +
                "      \"kms:Encrypt\"," +
                "      \"kms:GenerateDataKey*\"," +
                "      \"kms:Decrypt\"," +
                "      \"kms:DescribeKey\"," +
                "      \"kms:ReEncrypt*\"" +
                "    ]," +
                "    \"Resource\": \"*\"" +
                "  }]" +
                "}";

PutKeyPolicyRequest req = new
 PutKeyPolicyRequest().withKeyId(keyId).withPolicy(policy).withPolicyName(policyName);
kmsClient.putKeyPolicy(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 PutKeyPolicy 方法。

```
// Set a key policy for a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
String policyName = "default";
String policy = "{" +
                "  \"Version\": \"2012-10-17\"," +
                "  \"Statement\": [{" +
                "    \"Sid\": \"Allow access for ExampleUser\"," +
                "    \"Effect\": \"Allow\"," +
                // Replace the following example user ARN with a valid one
                "    \"Principal\": {\"AWS\": \"arn:aws:iam::111122223333:role/
ExampleKeyUserRole\"}," +
                "    \"Action\": [" +
                "      \"kms:Encrypt\"," +
                "      \"kms:GenerateDataKey*\"," +
                "      \"kms:Decrypt\"," +
```

```
                "      \"kms:DescribeKey\"," +
                "      \"kms:ReEncrypt*\"" +
                "    ]," +
                "    \"Resource\": \"*\"" +
                "  }]" +
                "}";

PutKeyPolicyRequest putKeyPolicyRequest = new PutKeyPolicyRequest()
{
    KeyId = keyId,
    Policy = policy,
    PolicyName = policyName
};
kmsClient.PutKeyPolicy(putKeyPolicyRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 [put_key_policy 方法](#)。

```
# Set a key policy for a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
policy_name = 'default'
policy = """
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "Allow access for ExampleUser",
        "Effect": "Allow",
        "Principal": {"AWS": "arn:aws:iam::111122223333:role/ExampleKeyUserRole"},
        "Action": [
            "kms:Encrypt",
            "kms:GenerateDataKey*",
            "kms:Decrypt",
            "kms:DescribeKey",
            "kms:ReEncrypt*"
        ],
        "Resource": "*"
    }]
}"""

response = kms_client.put_key_policy(
```

```
    KeyId=key_id,
    Policy=policy,
    PolicyName=policy_name
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 put_key_policy 執行個體方法。

```
# Set a key policy for a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
policy_name = 'default'
policy = "{" +
  "  \"Version\": \"2012-10-17\"," +
  "  \"Statement\": [{" +
  "    \"Sid\": \"Allow access for ExampleUser\"," +
  "    \"Effect\": \"Allow\"," +
  # Replace the following example user ARN with a valid one
  "    \"Principal\": {\"AWS\": \"arn:aws:iam::111122223333:role/ExampleKeyUserRole
\"}," +
  "    \"Action\": [" +
  "      \"kms:Encrypt\"," +
  "      \"kms:GenerateDataKey*\"," +
  "      \"kms:Decrypt\"," +
  "      \"kms:DescribeKey\"," +
  "      \"kms:ReEncrypt*\"" +
  "    ]," +
  "    \"Resource\": \"*\"" +
  "  }]" +
  "}"

response = kmsClient.put_key_policy({
  key_id: key_id,
  policy: policy,
  policy_name: policy_name
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 PutKeyPolicy 方法。

```
// Set a key policy for a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
$policyName = "default";

$result = $KmsClient->putKeyPolicy([
    'KeyId' => $keyId,
    'PolicyName' => $policyName,
    'Policy' => '{
        "Version": "2012-10-17",
        "Id": "custom-policy-2016-12-07",
        "Statement": [
            { "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal":
               { "AWS": "arn:aws:iam::111122223333:user/root" },
            "Action": [ "kms:*" ],
            "Resource": "*" },
            { "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal":
               { "AWS": "arn:aws:iam::111122223333:role/ExampleKeyUserRole" },
            "Action": [
                "kms:Encrypt*",
                "kms:GenerateDataKey*",
                "kms:Decrypt*",
                "kms:DescribeKey*",
                "kms:ReEncrypt*"
            ],
            "Resource": "*" }
        ]
    } '
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中適用之 AWS SDK JavaScript 中的putKeyPolicy 屬性。

```
// Set a key policy for a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
```

```
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
const PolicyName = 'default';
const Policy = `{
    "Version": "2012-10-17",
    "Id": "custom-policy-2016-12-07",
    "Statement": [
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/ExampleKeyUserRole"
            },
            "Action": [
                "kms:Encrypt*",
                "kms:GenerateDataKey*",
                "kms:Decrypt*",
                "kms:DescribeKey*",
                "kms:ReEncrypt*"
            ],
            "Resource": "*"
        }
    ]
}`; // The key policy document

kmsClient.putKeyPolicy({ KeyId, Policy, PolicyName }, (err, data) => {
  ...
});
```

PowerShell

若要設定 KMS 金鑰的金鑰原則，請使用寫 KMS 指令程式KeyPolicy。此 Cmdlet 不會傳回任何輸出。若要確認命令是否有效，請使用 Get-KMS KeyPolicy 指令程式。

Policy 參數採用字串。將字串包含在單引號中，使其成為常值字串。您不必在常值字串中使用連續字元或逸出字元。

```
# Set a key policy for a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
$policyName = 'default'
$policy = '{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:root"
            },
            "Action": "kms:*",
            "Resource": "*"
        },
        {
            "Sid": "Enable IAM User Permissions",
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::111122223333:role/ExampleKeyUserRole"
            },
            "Action": [
                "kms:Encrypt*",
                "kms:GenerateDataKey*",
                "kms:Decrypt*",
                "kms:DescribeKey*",
                "kms:ReEncrypt*"
            ],
            "Resource": "*"
        }]
    }'

Write-KMSKeyPolicy -KeyId $keyId -PolicyName $policyName -Policy $policy
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《AWS Tools for Windows PowerShell 使用者指南》。

使用授與

此主題中的範例使用 AWS KMS API 來建立、檢視、淘汰和撤銷 AWS KMS keys 的授予。如需在 AWS KMS 中使用授與的詳細資訊，請參閱AWS KMS 中的授權。

主題

建立授與

若要建立授權AWS KMS key，請使用CreateGrant作業。回應只包含授與 ID 和授與字符。若要取得授權的詳細資訊，請使用ListGrants作業，如中所示檢視授與。

這些範例會建立授權，讓可以擔任該`ExampleKeyUser`角色的使用者呼叫`KeyId`參數所識別之 KMS 金鑰上的GenerateDataKey作業。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

如需詳細資訊，請參閱《AWS SDK for Java API 參考》中的 createGrant 方法。

```
// Create a grant
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
String granteePrincipal = "arn:aws:iam::111122223333:role/ExampleKeyUser";
String operation = GrantOperation.GenerateDataKey.toString();

CreateGrantRequest request = new CreateGrantRequest()
    .withKeyId(keyId)
    .withGranteePrincipal(granteePrincipal)
    .withOperations(operation);

CreateGrantResult result = kmsClient.createGrant(request);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 CreateGrant 方法。

```
// Create a grant
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
String granteePrincipal = "arn:aws:iam::111122223333:role/ExampleKeyUser";
String operation = GrantOperation.GenerateDataKey;

CreateGrantRequest createGrantRequest = new CreateGrantRequest()
{
    KeyId = keyId,
    GranteePrincipal = granteePrincipal,
    Operations = new List<string>() { operation }
};

CreateGrantResponse createGrantResult = kmsClient.CreateGrant(createGrantRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 create_grant 方法。

```
# Create a grant

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
grantee_principal = 'arn:aws:iam::111122223333:role/ExampleKeyUser'
operation = ['GenerateDataKey']

response = kms_client.create_grant(
    KeyId=key_id,
    GranteePrincipal=grantee_principal,
    Operations=operation
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 create_grant 執行個體方法。

```
# Create a grant

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
grantee_principal = 'arn:aws:iam::111122223333:role/ExampleKeyUser'
operation = ['GenerateDataKey']

response = kmsClient.create_grant({
  key_id: key_id,
  grantee_principal: grantee_principal,
  operations: operation
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 CreateGrant 方法。

```
// Create a grant
//
// Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
$granteePrincipal = "arn:aws:iam::111122223333:role/ExampleKeyUser";
$operation = ['GenerateDataKey']

$result = $KmsClient->createGrant([
    'GranteePrincipal' => $granteePrincipal,
    'KeyId' => $keyId,
    'Operations' => $operation
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中 AWSSDK 中的 [createGrant] 屬性。 JavaScript

```
// Create a grant
//
// Replace the following example key ARN with a valid key ID or key ARN
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
const GranteePrincipal = 'arn:aws:iam::111122223333:role/ExampleKeyUser';
const Operations: ["GenerateDataKey"];
```

```
kmsClient.createGrant({ KeyId, GranteePrincipal, Operations }, (err, data) => {
  ...
});
```

PowerShell

若要建立授與，請使用 New-KMSGrant Cmdlet。

```
# Create a grant

# Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
$granteePrincipal = 'arn:aws:iam::111122223333:role/ExampleKeyUser'
$operation = 'GenerateDataKey'

$response = New-KMSGrant -GranteePrincipal $granteePrincipal -KeyId $keyId -
Operation $operation
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

檢視授與

若要取得 KMS 金鑰授權的詳細資訊，請使用ListGrants作業。

> **Note**
>
> GranteePrincipal 回應中的 ListGrants 欄位通常包含授與的承授者主體。不過，當授予中的承授者委託人是 AWS 服務時，GranteePrincipal 欄位會包含服務委託人，這可能代表數個不同的承授者委託人。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

這些範例會使用選用的 Limits 參數，此參數會決定操作傳回的授與次數。

Java

如需 Java 實作的詳細資訊，請參閱《AWS SDK for Java API 參考》中的 listGrants 方法。

```
// Listing grants on a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
Integer limit = 10;

ListGrantsRequest req = new ListGrantsRequest().withKeyId(keyId).withLimit(limit);
ListGrantsResult result = kmsClient.listGrants(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 ListGrants 方法。

```
// Listing grants on a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
int limit = 10;

ListGrantsRequest listGrantsRequest = new ListGrantsRequest()
{
    KeyId = keyId,
    Limit = limit
};
ListGrantsResponse listGrantsResponse = kmsClient.ListGrants(listGrantsRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 list_grants 方法。

```
# Listing grants on a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kms_client.list_grants(
    KeyId=key_id,
    Limit=10
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 list_grants 執行個體方法。

```
# Listing grants on a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

response = kmsClient.list_grants({
  key_id: key_id,
  limit: 10
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 ListGrants 方法。

```
// Listing grants on a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
$limit = 10;

$result = $KmsClient->listGrants([
    'KeyId' => $keyId,
    'Limit' => $limit,
]);
```

Node.js

有關詳細信息，請參閱 Node.js 中的 AWS SDK 中的 listGrants 屬性。 JavaScript

```
// Listing grants on a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
const Limit = 10;
kmsClient.listGrants({ KeyId, Limit }, (err, data) => {
  ...
```

```
});
```

PowerShell

若要檢視 KMS 金鑰的所有AWS KMS授權詳細資料，請使用 Get-KMS GrantList 指令程式。

若要限制輸出物件的數量，此範例使用 Select-Object Cmdlet，而不是在清單 Cmdlet 中已取代的 `Limit` 參數。如需 AWS Tools for PowerShell 中分頁輸出的說明，請參閱使用 AWS Tools for PowerShell 輸出分頁。

```
# Listing grants on a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
$limit = 10

$response = Get-KMSGrantList -KeyId $keyId | Select-Object -First $limit
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

您必須在每個 `ListGrants` 操作中指定 KMS 金鑰。不過，您可以指定授予 ID 或承授者委託人，進一步篩選授予清單。下列範例只會取得 KMS 金鑰的授予，其中 `test-engineer` 角色是承授者委託人。

Java

如需 Java 實作的詳細資訊，請參閱《AWS SDK for Java API 參考》中的 listGrants 方法。

```
// Listing grants on a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
String grantee = "arn:aws:iam::111122223333:role/test-engineer";

ListGrantsRequest req = new
 ListGrantsRequest().withKeyId(keyId).withGranteePrincipal(grantee);
ListGrantsResult result = kmsClient.listGrants(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 ListGrants 方法。

```
// Listing grants on a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";
String grantee = "arn:aws:iam::111122223333:role/test-engineer";

ListGrantsRequest listGrantsRequest = new ListGrantsRequest()
{
    KeyId = keyId,
    GranteePrincipal = grantee
};
ListGrantsResponse listGrantsResponse = kmsClient.ListGrants(listGrantsRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 list_grants 方法。

```
# Listing grants on a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
grantee = 'arn:aws:iam::111122223333:role/test-engineer'

response = kms_client.list_grants(
    KeyId=key_id,
    GranteePrincipal=grantee
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 list_grants 執行個體方法。

```
# Listing grants on a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
```

```
grantee = 'arn:aws:iam::111122223333:role/test-engineer'

response = kmsClient.list_grants({
  key_id: keyId,
  grantee_principal: grantee
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 ListGrants 方法。

```
// Listing grants on a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
$grantee = 'arn:aws:iam::111122223333:role/test-engineer';

$result = $KmsClient->listGrants([
    'KeyId' => $keyId,
    'GranteePrincipal' => $grantee,
]);
```

Node.js

有關詳細信息，請參閱 Node.js 中的 AWS SDK 中的 listGrants 屬性。 JavaScript

```
// Listing grants on a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';
const Grantee = 'arn:aws:iam::111122223333:role/test-engineer';

kmsClient.listGrants({ KeyId, Grantee }, (err, data) => {
  ...
});
```

PowerShell

若要檢視 KMS 金鑰的所有AWS KMS授權詳細資料，請使用 Get-KMS GrantList 指令程式。

```
# Listing grants on a KMS key
```

```
# Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'
$grantee = 'arn:aws:iam::111122223333:role/test-engineer'
$response = Get-KMSGrantList -KeyId $keyId -GranteePrincipal $grantee
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

淘汰授與

若要淘汰 KMS 金鑰的授權，請使用此RetireGrant作業。授與使用完畢後，您應該將其淘汰以進行清理。

若要淘汰授予，請提供授予字符，或同時提供授予 ID 和 KMS 金鑰 ID。若要進行此操作，KMS 金鑰 ID 必須是 KMS 金鑰的 Amazon Resource Name (ARN)。授予令牌由CreateGrant操作返回。授權識別碼由 CreateGrant 和作ListGrants業傳回。

RetireGrant 不返回響應。若要驗證其是否有效，請使用該ListGrants作業。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

如需詳細資訊，請參閱《AWS SDK for Java API 參考》中的 retireGrant 方法。

```
// Retire a grant
//
String grantToken = Place your grant token here;

RetireGrantRequest req = new RetireGrantRequest().withGrantToken(grantToken);
kmsClient.retireGrant(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 RetireGrant 方法。

```
// Retire a grant
//
String grantToken = "Place your grant token here";
```

```
RetireGrantRequest retireGrantRequest = new RetireGrantRequest()
{
    GrantToken = grantToken
};
kmsClient.RetireGrant(retireGrantRequest);
```

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 retire_grant 方法。

```
# Retire a grant

grant_token = Place your grant token here

response = kms_client.retire_grant(
    GrantToken=grant_token
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 retire_grant 執行個體方法。

```
# Retire a grant

grant_token = Place your grant token here

response = kmsClient.retire_grant({
  grant_token: grant_token
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 RetireGrant 方法。

```
// Retire a grant
//
$grantToken = 'Place your grant token here';

$result = $KmsClient->retireGrant([
    'GrantToken' => $grantToken,
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中 AWSSDK 中的 [retireGrant] 屬性。 JavaScript

```
// Retire a grant
//
const GrantToken = 'Place your grant token here';
kmsClient.retireGrant({ GrantToken }, (err, data) => {
  ...
});
```

PowerShell

若要淘汰授與，請使用 Disable-KMSGrant Cmdlet。若要取得授與字符，請使用 New-KMSGrant Cmdlet。`GrantToken` 參數會採用字串，因此您不需要轉換 Read-Host Cmdlet 傳回的輸出。

```
# Retire a grant

$grantToken = Read-Host -Message Place your grant token here
Disable-KMSGrant -GrantToken $grantToken
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

撤銷授與

若要撤銷 KMS 金鑰的授權，請使用此RevokeGrant作業。您可以撤銷授與以明確拒絕依賴它的操作。

在需要用戶端物件的語言中，這些範例會使用您在 建立用戶端 中建立的 AWS KMS 用戶端物件。

Java

如需詳細資訊，請參閱《AWS SDK for Java API 參考》中的 revokeGrant 方法。

```
// Revoke a grant on a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

// Replace the following example grant ID with a valid one
```

```
String grantId = "grant1";

RevokeGrantRequest req = new
 RevokeGrantRequest().withKeyId(keyId).withGrantId(grantId);
kmsClient.revokeGrant(req);
```

C#

如需詳細資訊，請參閱 AWS SDK for .NET 中的 RevokeGrant 方法。

```
// Revoke a grant on a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
String keyId = "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab";

// Replace the following example grant ID with a valid one
String grantId = "grant1";

RevokeGrantRequest revokeGrantRequest = new RevokeGrantRequest()
{
    KeyId = keyId,
    GrantId = grantId
};
kmsClient.RevokeGrant(revokeGrantRequest);
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《 使用者指南》AWS Tools for Windows PowerShell。

Python

如需詳細資訊，請參閱 AWS SDK for Python (Boto3) 中的 revoke_grant 方法。

```
# Revoke a grant on a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

# Replace the following example grant ID with a valid one
grant_id = 'grant1'

response = kms_client.revoke_grant(
```

```
    KeyId=key_id,
    GrantId=grant_id
)
```

Ruby

如需詳細資訊，請參閱 AWS SDK for Ruby 中的 revoke_grant 執行個體方法。

```
# Revoke a grant on a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
key_id = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

# Replace the following example grant ID with a valid one
grant_id = 'grant1'

response = kmsClient.revoke_grant({
  key_id: key_id,
  grant_id: grant_id
})
```

PHP

如需詳細資訊，請參閱 AWS SDK for PHP 中的 RevokeGrant 方法。

```
// Revoke a grant on a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';

// Replace the following example grant ID with a valid one
$grantId = "grant1";

$result = $KmsClient->revokeGrant([
    'KeyId' => $keyId,
    'GrantId' => $grantId,
]);
```

Node.js

如需詳細資訊，請參閱 Node.js 中 AWSSDK 中的 [revokeGrant] 屬性。 JavaScript

```
// Revoke a grant on a KMS key
//
// Replace the following example key ARN with a valid key ID or key ARN
const KeyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab';

// Replace the following example grant ID with a valid one
const GrantId = 'grant1';
kmsClient.revokeGrant({ GrantId, KeyId }, (err, data) => {
  ...
});
```

PowerShell

若要撤銷授與，請使用 Revoke-KMSGrant Cmdlet。

```
# Revoke a grant on a KMS key

# Replace the following example key ARN with a valid key ID or key ARN
$keyId = 'arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-1234567890ab'

# Replace the following example grant ID with a valid one
$grantId = 'grant1'

Revoke-KMSGrant -KeyId $keyId -GrantId $grantId
```

若要使用AWS KMS PowerShell 指令程式，請安裝 AWS.Tools。 KeyManagementService模塊。如需詳細資訊，請參閱《AWS Tools for Windows PowerShell 使用者指南》。

測試您的 AWS KMS API 呼叫

若要使用 AWS KMS，您必須擁有憑證以便 AWS 可用以驗證您的請求。憑證必須包含存取 KMS 金鑰與別名的許可。許可由金鑰政策、IAM 政策、授權以及跨帳戶存取控制決定。除控制 KMS 金鑰的存取權之外，您還可控制 CloudHSM 與自訂金鑰存放區的存取權。

您可指定 DryRun API 參數，驗證您擁有運用 AWS KMS 金鑰的必要許可。您也可利用 DryRun 來驗證 AWS KMS API 呼叫的請求參數是否已正確指定。

主題

什麼是 DryRun 參數？

DryRun 是選用的 API 參數，您可加以指定以便驗證 AWS KMS API 呼叫是否成功。在實際呼叫 AWS KMS 之前，利用 DryRun 來測試 API 呼叫。您可以驗證下列各項。

- 您擁有必要的許可，可運用 AWS KMS 金鑰。
- 您已正確指定呼叫的參數。

AWS KMS 支援在特定 API 動作採用 DryRun 參數：

- CreateGrant
- 解密
- 加密
- GenerateDataKey
- GenerateDataKeyPair
- GenerateDataKeyPairWithoutPlaintext
- GenerateDataKeyWithoutPlaintext
- GenerateMac
- ReEncrypt
- RetireGrant
- RevokeGrant
- 符號
- 確認
- VerifyMac

採用 DryRun 參數將產生費用，並作為標準 API 請求計費。如需關於 AWS KMS 定價的詳細資訊，請參閱 AWS Key Management Service 定價。

採用 DryRun 參數的所有 API 請求都會套用至 API 的請求配額，如您超出 API 請求配額，則可能導致限流例外狀況。例如，無論採用 DryRun 還是不採用 DryRun 來呼叫 Decrypt，都會根據相同的密碼編譯操作配額進行計數。如需進一步了解，請參閱《調節 AWS KMS 請求》。

系統會將每次對 AWS KMS API 操作的呼叫擷取並記錄為 AWS CloudTrail 日誌的事件。任何指定DryRun參數之作業的輸出都會顯示在 CloudTrail 記錄檔中。如需詳細資訊，請參閱 使用 AWS CloudTrail 記錄 AWS KMS API 呼叫。

DryRun 使用 API 指定

若要採用 DryRun，請在支援 –dry-run 參數的 AWS CLI 命令與 AWS KMS API 呼叫指定該參數。在這樣做之後，AWS KMS 將驗證您的呼叫是否會成功。採用 DryRun 的 AWS KMS 呼叫一律會失敗，並傳回訊息以提供資訊說明呼叫失敗的原因。訊息可能包含下列例外狀況：

- DryRunOperationException - 如未指定 DryRun，請求就會成功。
- ValidationException - 請求因指定不正確 API 參數而失敗。
- AccessDeniedException - 您不具權限，無法對 KMS 資源執行指定 API 動作。

例如，下列命令會使用CreateGrant作業並建立授權，允許有權承擔keyUserRole角色的使用者在指定的對稱 KMS 金鑰上呼叫「解密」作業。指定 DryRun 參數。

```
$  aws kms create-grant \
    --key-id 1234abcd-12ab-34cd-56ef-1234567890ab \
    --grantee-principal arn:aws:iam::111122223333:role/keyUserRole \
    --operations Decrypt \
    --dry-run
```

AWS KMS 最終一致性

由於系統的分散式性質，AWS KMS API 遵循最終一致性模式。因此，您執行的後續命令可能不會立即注意到對 AWS KMS 資源的變更。

當您執行 AWS KMS API 呼叫時，在該變更適用於整個 AWS KMS 之前，可能會有短暫延遲。變更傳播到整個系統通常需要不到幾秒鐘的時間，但在某些情況下可能需要幾分鐘。在此期間，您可能會收到非預期的錯誤，例如 NotFoundException 或 InvalidStateException。例如，如果您在呼叫 CreateKey 後立即呼叫 GetParametersForImport，AWS KMS 可能會返回 NotFoundException。

我們建議您在 AWS KMS 用戶端設定重試策略，以便在短暫的等待期間後自動重試作業。如需詳細資訊，請參閱與叫在 AWS SDK 與工具參考指南的重試操作 。

對於授權相關的 API 呼叫，您可以使用授權權杖來避免任何潛在的延遲，並立即在授權中使用權限。如需詳細資訊，請參閱最終一致性模式 (授權)。

參考

下列參考提供有關使用和管理 KMS 金鑰的實用資訊。

- 金鑰類型參考。列出支援每個 AWS KMS API 操作的 KMS 金鑰類型。

  若要尋找：我可以啟用和停用 RSA 簽署 KMS 金鑰嗎？
- 金鑰狀態資料表。顯示 KMS 金鑰的金鑰狀態會如何影響其在 AWS KMS API 操作的使用。

  若要尋找：我可以變更等待刪除的 KMS 金鑰的別名嗎？
- AWS KMS API 許可參考。提供每個 AWS KMS API 操作所需許可的詳細資訊。

  要查找：我可以在不同AWS帳戶中的密鑰GetKeyPolicy上運行嗎？ 我可以允許 IAM 政策中的 `kms:Decrypt` 許可嗎？
  - ViaService 參考。支援 `kms:ViaService` 條件金鑰的 AWS 服務清單。

    要查找：我可以使用`kms:ViaService`條件密鑰僅在來自 Amazon 時允許許可 ElastiCache嗎？ Amazon Neptune 呢？
- AWS KMS 定價。列出並說明 KMS 金鑰的價格。

  若要尋找：使用我的非對稱金鑰的費用多少？
- AWS KMS 請求配額。列出每個帳戶和區域中 AWS KMS API 請求的每秒配額。

  若要尋找：我每秒可以執行多少 Decrypt 請求？ 我可以在自訂金鑰存放區的 KMS 金鑰上執行多少 Decrypt 請求？
- AWS KMS 資源配額。列出 AWS KMS 資源的配額。

  若要尋找：我的帳戶每個區域可以有多少 KMS 金鑰？ 每個 KMS 金鑰可以有多少個別名？
- 與 AWS KMS 整合的 AWS 服務。列出使用 KMS 金鑰來保護其建立、存放和管理的資源的 AWS 服務。

  若要尋找：Amazon Connect 是否使用 KMS 金鑰來保護我的 Connect 資源？

文件歷史紀錄

此主題描述《AWS Key Management Service 開發人員指南》的重大更新。

主題

最近更新

下表說明此文件自 2018 年 1 月後的重大變更。除了這裡所列的主要變更外，我們也會經常更新文件以改進說明內容和範例，並且反映您傳送給我們的意見回饋。如要接收重大變更的通知，請訂閱 RSS 摘要。

您可能需要水平或垂直捲動，才能查看此資料表中的所有資料。

變更	描述	日期
受管理策略的更新	增加了新的權限`AWSKeyManagementServiceCustomKeyStoresServiceRolePolicy`，AWS KMS 允許監視 VPC 中包含您的 AWS CloudHSM叢集的變更，AWS KMS以便在發生故障時提供明確的錯誤訊息。	2023 年 11 月 10 日
功能更新	已新增對 `DryRun` API 參數的支援。	2023 年 7 月 5 日
功能更新	針對所有類型 AWS KMS 金鑰的匯入金鑰資料新增支援，除自訂金鑰存放區外。	2023 年 6 月 5 日
功能更新	更新 Nitro Enclaves 的 AWS KMS API	2023 年 3 月 10 日

功能更新	已取代 RSAES_PKCS1_V1_5 包裝演算法。根據國家標準技術研究所 (NIST) 的加密金鑰管理指引，AWS KMS 將於 2023 年 10 月 1 日前針對 RSAES_PKCS1_V1_5 結束所有支援。建議您立即開始採用不同包裝演算法。	2023 年 2 月 28 日
功能更新	已新增對外部金鑰存放區的支援，這項功能可讓您使用 AWS 之外的密碼編譯金鑰來保護 AWS 資源。	2022 年 11 月 29 日
配額變更	增加每個帳戶和區域中的 AWS KMS keys 資源配額至 100,000 個 KMS 金鑰。	2022 年 7 月 8 日
功能更新	在多個 AWS 區域 新增了對 HMAC KMS 金鑰的支援	2022 年 7 月 8 日
新主題	已將 AWS Key Management Service 主題中的復原功能新增至《AWS KMS 開發人員指南》的「安全」章節。	2022 年 6 月 14 日
新功能	已新增對 AWS KMS 金鑰和 API 操作 (可產生和驗證 HMAC 代碼) 的支援。	2022 年 4 月 19 日
文件變更	使用 AWS KMS key 和 KMS 金鑰 取代術語客戶主要金鑰 (CMK)。	2021 年 8 月 30 日

新功能	新增對多區域金鑰的支援，這是一組在不同區域中具有相同金鑰 ID 和金鑰材料的可互通 KMS 金鑰。您可以使用多區域金鑰在一個區域中加密資料，並在其他區域中解密資料。	2021 年 6 月 8 日
新功能	已新增對屬性型存取控制 (ABAC) 的支援。您可以使用標籤和別名來控制對 AWS KMS keys 的存取。	2020 年 12 月 17 日
新功能	已新增對 VPC 端點政策的支援。	2020 年 7 月 9 日
新內容	說明 AWS KMS 的安全屬性。	2020 年 6 月 18 日
新功能	已新增對非對稱 AWS KMS keys 和非對稱資料金鑰的支援。	2019 年 11 月 25 日
更新的功能	您可以在 AWS KMS 主控台中檢視 AWS 受管金鑰 的金鑰政策。此功能過去僅限於客戶受管金鑰。	2019 年 11 月 15 日
新功能	說明如何將 TLS 中的混合式後量子金鑰交換演算法用於對 AWS KMS 的呼叫。	2019 年 11 月 4 日
配額變更	提高管理 KMS 金鑰之部分 API 的資源配額。	2019 年 9 月 18 日
配額變更	變更每個 KMS 金鑰之 KMS 金鑰、別名和授予的資源配額。	2019 年 3 月 27 日
配額變更	對於在自訂金鑰存放區中使用 AWS KMS keys 的加密操作，已變更共用的每秒請求配額。	2019 年 3 月 7 日

新功能	說明如何建立和管理 AWS KMS 自訂金鑰存放區。每個金鑰存放區由您擁有和控制的 AWS CloudHSM 叢集所支援。	2018 年 11 月 26 日
新主控台	說明如何使用新的 AWS KMS 主控台 (與 IAM 主控台無關)。短期間內仍會保留原始主控台及其使用指示，以利於您熟悉新的主控台。	2018 年 11 月 7 日
配額變更	已變更共用的請求配額，供 AWS KMS keys 使用。	2018 年 8 月 21 日
新內容	說明 AWS Secrets Manager 如何使用 AWS KMS 金鑰來加密機密中的機密值。	2018 年 7 月 13 日
新內容	說明 DynamoDB 如何使用 AWS KMS AWS KMS keys 來支援其伺服器端加密選項。	2018 年 5 月 23 日
新功能	說明如何在 VPC 中使用私有端點直接連接至 AWS KMS，而無需連接至網際網路。	2018 年 1 月 22 日

舊版更新

下表描述了 2018 年後《AWS Key Management Service 開發人員指南》的重要變更。

您可能需要水平或垂直捲動，才能查看此資料表中的所有資料。

變更	描述	日期
新內容	已新增 標記金鑰 的相關文件。	2017 年 2 月 15 日

變更	描述	日期
新內容	已新增監控 AWS KMS keys和使用 Amazon 監控 CloudWatch的相關文件。	2016 年 8 月 31 日
新內容	已新增 匯入的金鑰資料 的相關文件。	2016 年 8 月 11 日
新內容	已新增下列文件：IAM 政策、許可參考 和 條件索引鍵。	2016 年 7 月 5 日
更新	更新身分驗證與存取控制章節的文件部分。	2016 年 7 月 5 日
更新	更新配額頁面，以反映新的預設配額。	2016 年 5 月 31 日
更新	已更新 配額 頁面，以反映新的預設配額，並且已更新授予字符文件，以改善清晰度和準確性。	2016 年 4 月 11 日
新內容	已新增允許多個 IAM 主體存取 KMS 金鑰和使用 IP 地址條件的相關文件。	2016 年 2 月 17 日
更新	更新AWS KMS 中的金鑰政策和變更金鑰政策頁面，以改善清晰度和準確性。	2016 年 2 月 17 日
更新	更新管理金鑰主題頁面，以改善清晰度。	2016 年 1 月 5 日
新內容	已新增 AWS CloudTrail 使用 AWS KMS 的方式 的相關文件。	2015 年 11 月 18 日
新內容	新增變更金鑰政策的說明。	2015 年 11 月 18 日

變更	描述	日期
更新	已更新 Amazon Relational Database Service (Amazon RDS) 如何使用 AWS KMS 的相關文件。	2015 年 11 月 18 日
新內容	已新增 如何 WorkSpaces 使用 AWS KMS 的相關文件。	2015 年 11 月 6 日
更新	更新AWS KMS 中的金鑰政策頁面，以改善清晰度。	2015 年 10 月 22 日
新內容	新增刪除 AWS KMS keys的相關文件，包含建立警示和判斷 KMS 金鑰的過去使用情形的相關支援文件。	2015 年 10 月 15 日
新內容	已新增 判斷 AWS KMS keys 的存取權 的相關文件。	2015 年 10 月 15 日
新內容	已新增 AWS KMS 金鑰的金鑰狀態 的相關文件。	2015 年 10 月 15 日
新內容	已新增 Amazon Simple Email Service (Amazon SES) 如何使用 AWS KMS 的相關文件。	2015 年 10 月 1 日
更新	已更新 配額 頁面以說明新的請求配額。	2015 年 8 月 31 日
新內容	新增關於使用 AWS KMS 的費用資訊。請參閱 AWS KMS 定價。	2015 年 8 月 14 日
新內容	已新增請求配額至 AWS KMS 配額。	2015 年 6 月 11 日

變更	描述	日期
新內容	已新增示範使用 `UpdateAlias` 操作的新 Java 程式碼範例。請參閱 更新別名。	2015 年 6 月 1 日
更新	將 AWS Key Management Service 區域表格移動到 AWS 一般參考。	2015 年 5 月 29 日
新內容	已新增 Amazon EMR 如何使用 AWS KMS 的相關文件。	2015 年 1 月 28 日
新內容	已新增 Amazon 如何 WorkMail 使用 AWS KMS 的相關文件。	2015 年 1 月 28 日
新內容	已新增 Amazon Relational Database Service (Amazon RDS) 如何使用 AWS KMS 的相關文件。	2015 年 1 月 6 日
新內容	已新增 Amazon Elastic Transcoder 如何使用 AWS KMS 的相關文件。	2014 年 11 月 24 日
新指南	已介紹《AWS Key Management Service 開發人員指南》。	2014 年 11 月 12 日

本文為英文版的機器翻譯版本，如內容有任何歧義或不一致之處，概以英文版為準。