



# 支援 **Amazon S3 REST API**

## StorageGRID software

NetApp  
May 29, 2026

# 目錄

支援 Amazon S3 REST API .....	1
S3 REST API 實作細節 .....	1
日期處理 .....	1
常見請求標頭 .....	1
常見響應頭 .....	1
驗證請求 .....	2
使用 HTTP 授權標頭 .....	2
使用查詢參數 .....	2
對服務的操作 .....	2
對 bucket 的操作 .....	2
對物件的操作 .....	9
對物件的操作 .....	9
使用 S3 Select .....	12
使用伺服器端加密 .....	15
複製對象 .....	17
取得對象 .....	20
頭部對象 .....	22
放置對象 .....	25
復原對象 .....	30
選擇對象內容 .....	31
分段上傳的操作 .....	35
分段上傳的操作 .....	35
完成多部分上傳 .....	36
建立多部分上傳 .....	38
列出多部分上傳 .....	40
上傳部分 .....	41
上傳部分複製 .....	42
錯誤回應 .....	43
支援的 S3 API 錯誤代碼 .....	43
StorageGRID 自訂錯誤代碼 .....	45

# 支援 Amazon S3 REST API

## S3 REST API 實作細節

StorageGRID系統實作了簡單儲存服務 API (API 版本 2006-03-01) ，支援大多數操作，但也有一些限制。在整合 S3 REST API 用戶端應用程式時，您需要了解實作細節。

StorageGRID系統支援虛擬託管式請求和路徑式請求。

### 日期處理

S3 REST API 的StorageGRID實作僅支援有效的 HTTP 日期格式。

StorageGRID系統僅支援接受日期值的任何標頭的有效 HTTP 日期格式。日期的時間部分可以採用格林威治標準時間 (GMT) 格式指定，也可以採用沒有時區偏移的協調世界時 (UTC) 格式指定 (必須指定 +0000)。如果你包括 `x-amz-date` 請求中的標頭，它會覆寫 Date 請求標頭中指定的任何值。使用 AWS 簽章版本 4 時，`x-amz-date` 標頭必須存在於簽署的請求中，因為不支援日期標頭。

### 常見請求標頭

StorageGRID系統支援由下列定義的通用請求標頭 "[Amazon Simple Storage Service API 參考：常見請求標頭](#)"，但有一個例外。

請求頭	執行
授權	完全支援 AWS 簽章版本 2  支援 AWS 簽章版本 4，但有以下例外： <ul style="list-style-type: none"><li>當您提供實際有效載荷校驗和值時 <code>x-amz-content-sha256</code>，該值無需驗證即可被接受，就像該值 `UNSIGNED-PAYLOAD` 已提供標題。當你提供 `x-amz-content-sha256` 標頭值暗示 `aws-chunked` 串流 (例如，STREAMING-AWS4-HMAC-SHA256-PAYLOAD)，區塊簽章不會根據區塊資料進行驗證。</li></ul>
x-amz-安全令牌	未實施。返回 <code>XNotImplemented</code> 。

### 常見響應頭

StorageGRID系統支援《簡單儲存服務 API 參考》定義的所有常見回應標頭，但有一個例外。

回應頭	執行
x-amz-id-2	未使用

## 驗證請求

StorageGRID系統支援使用 S3 API 對物件進行經過驗證和匿名存取。

S3 API 支援簽章版本 2 和簽章版本 4 用於驗證 S3 API 請求。

必須使用您的存取金鑰 ID 和秘密存取金鑰對經過驗證的請求進行簽署。

StorageGRID系統支援兩種身份驗證方法：HTTP `Authorization` 標題並使用查詢參數。

### 使用 HTTP 授權標頭

HTTP `Authorization` 除儲存桶策略允許的匿名請求外，所有 S3 API 操作都使用標頭。這 `Authorization` 標頭包含驗證請求所需的所有簽名資訊。

### 使用查詢參數

您可以使用查詢參數向 URL 新增身份驗證資訊。這稱為預簽名 URL，可用於授予對特定資源的臨時存取權限。具有預簽名 URL 的使用者不需要知道秘密存取金鑰即可存取資源，這使您能夠提供對資源的第三方受限存取。

## 對服務的操作

StorageGRID系統支援對服務進行下列操作。

手術	執行
列表桶  (以前稱為 GET 服務)	使用所有 Amazon S3 REST API 行為實作。如有變更，恕不另行通知。
取得儲存使用情況	StorageGRID"取得儲存使用情況"請求會告訴您帳戶使用的儲存總量以及與該帳戶關聯的每個儲存桶的儲存總量。這是對具有 / 路徑和自訂查詢參數的服務的操作(?x-ntap-sg-usage) 補充道。
選項 /	客戶端應用程式可以發出 `OPTIONS /` 向儲存節點上的 S3 連接埠發出請求，無需提供 S3 驗證憑證，以確定儲存節點是否可用。您可以使用此請求進行監控，或允許外部負載平衡器識別儲存節點何時關閉。

## 對 bucket 的操作

StorageGRID系統支援每個 S3 租用戶帳戶最多 5,000 個儲存桶。

每個網格最多可以有 100,000 個桶。

為了支援 5,000 個儲存桶，網格中的每個儲存節點必須至少具有 64 GB 的 RAM。

儲存桶名稱限制遵循 AWS 美國標準區域限制，但您應該進一步將它們限制為 DNS 命名約定，以支援 S3 虛擬

託管樣式請求。

有關詳細信息，請參閱以下內容：

- ["Amazon Simple Storage Service 使用者指南：儲存桶配額、限制和局限性"](#)
- ["配置 S3 端點域名"](#)

ListObjects（取得 Bucket）和 ListObjectVersions（取得 Bucket 物件版本）操作支援StorageGRID"[一致性值](#)"。

您可以檢查是否針對各個儲存桶啟用或停用了上次存取時間的更新。看"[取得 Bucket 上次造訪時間](#)"。

下表描述了StorageGRID如何實作 S3 REST API 儲存桶操作。要執行任何這些操作，必須為帳戶提供必要的存取憑證。

手術	執行
創建桶	<p>建立一個新的儲存桶。透過建立儲存桶，您就成為儲存桶的擁有者。</p> <ul style="list-style-type: none"><li>• Bucket 名稱必須遵循以下規則：<ul style="list-style-type: none"><li>◦ 每個StorageGRID系統中必須是唯一的（而不僅僅是在租用戶帳戶內是唯一的）。</li><li>◦ 必須符合 DNS 標準。</li><li>◦ 必須包含至少 3 個字元且不超過 63 個字元。</li><li>◦ 可以是一系列一個或多個標籤，相鄰的標籤以句點分隔。每個標籤必須以小寫字母或數字開頭和結尾，並且只能使用小寫字母、數字和連字符。</li><li>◦ 不能看起來像文字格式的 IP 位址。</li><li>◦ 不應在虛擬託管樣式請求中使用句點。句點將導致伺服器通配符憑證驗證出現問題。</li></ul></li><li>• 預設情況下，儲存桶在 `us-east-1` 區域；但是，您可以使用 `LocationConstraint` 請求主體中的請求元素來指定不同的區域。使用時 `LocationConstraint` 元素，您必須指定使用網格管理器或網格管理 API 定義的區域的確切名稱。如果您不知道應該使用的區域名稱，請聯絡您的系統管理員。</li></ul> <p>注意：如果您的 CreateBucket 要求使用StorageGRID中未定義的區域，則會發生錯誤。</p> <ul style="list-style-type: none"><li>• 您可以包括 `x-amz-bucket-object-lock-enabled` 請求標頭來建立啟用了 S3 物件鎖的儲存桶。看"<a href="#">使用 S3 REST API 設定 S3 物件鎖</a>"。</li></ul> <p>建立儲存桶時必須啟用 S3 物件鎖定。建立儲存桶後，您無法新增或停用 S3 物件鎖定。S3 物件鎖定需要儲存桶版本控制，該功能在您建立儲存桶時自動啟用。</p>
刪除桶	刪除儲存桶。

手術	執行
刪除BucketCors	刪除儲存桶的 CORS 配置。
刪除桶加密	從儲存桶中刪除預設加密。現有的加密物件仍保持加密狀態，但新增至儲存桶的任何新物件都不會加密。
DeleteBucket生命週期	從儲存桶中刪除生命週期配置。看" <a href="#">建立 S3 生命週期配置</a> "。
刪除桶策略	刪除附加到儲存桶的策略。
刪除桶複製	刪除附加到儲存桶的複製配置。
刪除桶標記	<p>使用 `tagging` 子資源從儲存桶中刪除所有標籤。</p> <p>警告：如果為此儲存桶設定了非預設的 ILM 策略標籤，則會出現 `NTAP-SG-ILM-BUCKET-TAG` 已指派值的 bucket 標籤。如果存在 `NTAP-SG-ILM-BUCKET-TAG` 桶標籤。相反，發出僅包含 `NTAP-SG-ILM-BUCKET-TAG` 標籤及其指派的值，以從儲存桶中刪除所有其他標籤。請勿修改或刪除 `NTAP-SG-ILM-BUCKET-TAG` 桶標籤。</p>
獲取BucketAcl	傳回肯定回應以及儲存桶擁有者的 ID、DisplayName 和 Permission，表示擁有者對該儲存桶具有完全存取權。
獲取BucketCors	返回 `cors` 儲存桶的配置。
取得桶加密	傳回儲存桶的預設加密配置。
取得BucketLifecycleConfiguration  (以前稱為 GET Bucket 生命週期)	傳回儲存桶的生命週期配置。看" <a href="#">建立 S3 生命週期配置</a> "。
取得儲存桶位置	返回使用 LocationConstraint `CreateBucket` 請求中的元素。如果 bucket 的區域是 `us-east-1`，則該區域傳回一個空字串。
取得儲存桶通知配置  (以前稱為 GET Bucket 通知)	傳回附加到儲存桶的通知配置。
取得BucketPolicy	傳回附加到儲存桶的策略。
獲取Bucket複製	傳回附加到儲存桶的複製配置。

手術	執行
取得桶標記	<p>使用 `tagging` 子資源傳回儲存桶的所有標籤。</p> <p>警告：如果為此儲存桶設定了非預設的 ILM 策略標籤，則會出現 `NTAP-SG-ILM-BUCKET-TAG` 已指派值的 bucket 標籤。請勿修改或刪除此標籤。</p>
取得Bucket版本	<p>此實作使用 `versioning` 子資源傳回儲存桶的版本控制狀態。</p> <ul style="list-style-type: none"> <li>• <i>blank</i>：從未啟用版本控制（儲存桶為「未版本控制」）</li> <li>• 已啟用：版本控制已啟用</li> <li>• 已暫停：版本控制之前已啟用，現已暫停</li> </ul>
取得物件鎖配置	<p>如果已配置，則傳回儲存桶預設保留模式和預設保留期。</p> <p>看<a href="#">"使用 S3 REST API 設定 S3 物件鎖"</a>。</p>
頭桶	<p>確定儲存桶是否存在以及您是否有權存取它。</p> <p>此操作返回：</p> <ul style="list-style-type: none"> <li>• <code>x-ntap-sg-bucket-id</code>：Bucket 的 UUID，格式為 UUID。</li> <li>• <code>x-ntap-sg-trace-id</code>：關聯請求的唯一追蹤 ID。</li> </ul>
ListObjects 和 ListObjectsV2  (以前稱為 GET Bucket)	<p>傳回儲存桶中的部分或全部（最多 1,000 個）物件。物件的儲存類別可以有兩個值，即使物件是透過 `REDUCED_REDUNDANCY` 儲存類別選項：</p> <ul style="list-style-type: none"> <li>• STANDARD，表示該物件儲存在由 Storage Node 組成的儲存池中。</li> <li>• GLACIER，表示該物件已移至雲端儲存池指定的外部儲存桶中。</li> </ul> <p>如果儲存桶包含大量具有相同前綴的已刪除鍵，則回應可能包含一些 `CommonPrefixes` 不包含密鑰。</p>
列出物件版本  (以前稱為 GET Bucket Object 版本)	<p>擁有儲存桶的讀取權限，使用此操作 `versions` 子資源列出了儲存桶中所有版本的物件的元資料。</p>
PutBucketCors	<p>設定儲存桶的 CORS 配置，以便儲存桶可以處理跨域請求。跨域資源共用 (CORS) 是一種安全機制，可讓一個網域中的用戶端 Web 應用程式存取不同網域中的資源。例如，假設您使用名為 <code>images</code> 儲存圖形。透過設定 CORS 配置 <code>images</code> bucket，您可以允許該 bucket 中的圖像顯示在網站上 <code>http://www.example.com</code>。</p>

手術	執行
PutBucket加密	<p>設定現有儲存桶的預設加密狀態。啟用儲存桶級加密後，新增至儲存桶的任何新物件都將被加密。StorageGRID 支援使用StorageGRID管理的金鑰進行伺服器端加密。指定伺服器端加密配置規則時，設定 SSEAlgorithm`參數 `AES256，並且不要使用 `KMSMasterKeyID`範圍。</p> <p>如果物件上傳請求已經指定了加密（即，如果請求包含 `x-amz-server-side-encryption-*` 請求標頭）。</p>
PutBucket生命週期配置 (以前稱為 PUT Bucket 生命週期)	<p>為儲存桶建立新的生命週期配置或取代現有的生命週期配置。StorageGRID在生命週期配置中支援最多 1,000 條生命週期規則。每個規則可以包含以下 XML 元素：</p> <ul style="list-style-type: none"> <li>• 到期日（天數、日期、ExpiredObjectDeleteMarker）</li> <li>• NoncurrentVersionExpiration（NewerNoncurrentVersions、NoncurrentDays）</li> <li>• 過濾器（前綴、標籤）</li> <li>• 地位</li> <li>• ID</li> </ul> <p>StorageGRID不支援以下操作：</p> <ul style="list-style-type: none"> <li>• 中止未完成的分段上傳</li> <li>• 過渡</li> </ul> <p>看"<a href="#">建立 S3 生命週期配置</a>"。若要了解儲存桶生命週期中的到期操作如何與 ILM 放置指令交互，請參閱"<a href="#">ILM 如何在物件的整個生命週期中運作</a>"。</p> <p>注意：儲存桶生命週期配置可與啟用了 S3 物件鎖的儲存桶一起使用，但儲存桶生命週期配置不支援舊版相容儲存桶。</p>

手術	執行
PutBucketNotification配置 (以前稱為 PUT Bucket 通知)	<p>使用請求正文中包含的通知配置 XML 為儲存桶配置通知。您應該了解以下實施細節：</p> <ul style="list-style-type: none"> <li>• StorageGRID支援 Amazon Simple Notification Service (Amazon SNS) 或 Kafka 主題作為目的地。不支援簡單佇列服務 (SQS) 或 Amazon Lambda 終端節點。</li> <li>• 通知的目的地必須指定為StorageGRID端點的 URN。可以使用租用戶管理器或租用戶管理 API 建立端點。</li> </ul> <p>端點必須存在才能成功配置通知。如果端點不存在，400 Bad Request 錯誤與代碼一起返回 InvalidArgument。</p> <ul style="list-style-type: none"> <li>• 您無法為以下事件類型設定通知。這些事件類型不受支援。               <ul style="list-style-type: none"> <li>◦ s3:ReducedRedundancyLostObject</li> <li>◦ s3:ObjectRestore:Completed</li> </ul> </li> <li>• 從StorageGRID傳送的事件通知使用標準 JSON 格式，但它們不包含某些鍵，而對其他鍵使用特定值，如下列清單所示：               <ul style="list-style-type: none"> <li>◦ 事件源                   <ul style="list-style-type: none"> <li>sgws:s3</li> </ul> </li> <li>◦ <b>aws</b>區域                   <ul style="list-style-type: none"> <li>不包括</li> </ul> </li> <li>◦ <b>x-amz-id-2</b> <ul style="list-style-type: none"> <li>不包括</li> </ul> </li> <li>◦ 阿恩                   <ul style="list-style-type: none"> <li>urn:sgws:s3:::bucket_name</li> </ul> </li> </ul> </li> </ul>
PutBucketPolicy	設定附加到儲存桶的策略。看 <a href="#">"使用儲存桶和群組存取策略"</a> 。

手術	執行
PutBucket複製	<p>配置"<a href="#">StorageGRID CloudMirror 複製</a>"對於儲存桶，使用請求正文中提供的複製配置 XML。對於 CloudMirror 複製，您應該注意以下實作細節：</p> <ul style="list-style-type: none"> <li>StorageGRID僅支援複製配置的 V1。這意味著StorageGRID不支援使用 `Filter` 元素作為規則，並遵循 V1 約定來刪除物件版本。有關詳細信息，請參閱 "<a href="#">Amazon Simple Storage Service 使用者指南：複製配置</a>"。</li> <li>可以在版本化或非版本化的儲存桶上配置儲存桶複製。</li> <li>您可以在複製配置 XML 的每個規則中指定不同的目標儲存桶。一個來源儲存桶可以複製到多個目標儲存桶。</li> <li>目標儲存桶必須指定為租用戶管理器或租用戶管理 API 中指定的StorageGRID 端點的 URN。看"<a href="#">配置 CloudMirror 複製</a>"。</li> </ul> <p>端點必須存在才能成功進行複製配置。如果端點不存在，請求將會失敗，因為 400 Bad Request。錯誤訊息指出：Unable to save the replication policy. The specified endpoint URN does not exist: URN.</p> <ul style="list-style-type: none"> <li>您不需要指定 `Role` 在配置 XML 中。StorageGRID不使用此值，如果提交將被忽略。</li> <li>如果您從配置 XML 中省略儲存類，StorageGRID將使用 `STANDARD` 預設儲存類別。</li> <li>如果您從來源儲存桶中刪除物件或刪除來源儲存桶本身，則跨區域複製行為如下： <ul style="list-style-type: none"> <li>如果您在複製之前刪除物件或儲存桶，則該物件/儲存桶不會被複製，並且您不會收到通知。</li> <li>如果在複製後刪除物件或儲存桶，StorageGRID將遵循跨區域複製 V1 的標準 Amazon S3 刪除行為。</li> </ul> </li> </ul>
PutBucketTagging	<p>使用 `tagging` 子資源來新增或更新儲存桶的一組標籤。在新增儲存桶標籤時，請注意以下限制：</p> <ul style="list-style-type: none"> <li>StorageGRID和 Amazon S3 都支援每個儲存桶最多 50 個標籤。</li> <li>與儲存桶關聯的標籤必須具有唯一的標籤鍵。標籤鍵的長度最多可達 128 個 Unicode 字元。</li> <li>標籤值的長度最多可達 256 個 Unicode 字元。</li> <li>鍵和值區分大小寫。</li> </ul> <p>警告：如果為此儲存桶設定了非預設的 ILM 策略標籤，則會出現 `NTAP-SG-ILM-BUCKET-TAG` 已指派值的 bucket 標籤。確保 `NTAP-SG-ILM-BUCKET-TAG` 所有 PutBucketTagging 請求中都包含儲存桶標籤和已指派的值。請勿修改或刪除此標籤。</p> <p>注意：此操作將覆蓋儲存桶已有的任何目前標籤。如果集合中省略了任何現有標籤，則將從儲存桶中刪除這些標籤。</p>

手術	執行
PutBucket版本控制	<p>使用 `versioning` 子資源來設定現有儲存桶的版本控制狀態。您可以使用下列值之一設定版本控制狀態：</p> <ul style="list-style-type: none"> <li>• 已啟用：為儲存桶中的物件啟用版本控制。新增到儲存桶的所有物件都會收到唯一的版本 ID。</li> <li>• 暫停：停用儲存桶中物件的版本控制。新增到儲存桶的所有物件都會收到版本 ID null。</li> </ul>
PutObjectLock配置	<p>配置或刪除儲存桶預設保留模式和預設保留期。</p> <p>如果修改了預設保留期，則現有物件版本的保留截止日期保持不變，並且不會使用新的預設保留期重新計算。</p> <p>看"<a href="#">使用 S3 REST API 設定 S3 物件鎖</a>"了解詳細資訊。</p>

## 對物件的操作

### 對物件的操作

本節介紹StorageGRID系統如何為物件實作 S3 REST API 操作。

以下條件適用於所有物件操作：

- StorageGRID"[一致性值](#)"所有物件操作均支持，但以下操作除外：
  - 取得對象Acl
  - OPTIONS /
  - 放置對象合法保留
  - PutObjectRetention
  - 選擇對象內容
- 衝突的客戶端請求（例如兩個客戶端寫入同一個金鑰）將根據「最新勝利」的原則解決。「最新勝利」評估的時間取決於StorageGRID系統完成給定請求的時間，而不是 S3 用戶端開始操作的時間。
- StorageGRID儲存桶中的所有物件均歸儲存體桶擁有者所有，包括由匿名使用者或其他帳號建立的物件。
- 透過 Swift 提取到StorageGRID系統的資料物件無法透過 S3 存取。

下表描述了StorageGRID如何實作 S3 REST API 物件操作。

手術	執行
刪除對象	<p>在處理 DeleteObject 請求時，StorageGRID 會嘗試立即從所有儲存位置刪除該物件的所有副本。如果成功，StorageGRID 會立即向用戶端回傳回應。如果無法在 30 秒內刪除所有副本（例如，由於某個位置暫時無法使用），StorageGRID 會將副本排隊等待刪除，然後向用戶端指示成功。</p> <p>限制</p> <ul style="list-style-type: none"> <li>• 多重身份驗證 (MFA) 和回應標頭 `x-amz-mfa` 不受支援。</li> <li>• The If-None 和 If-None-Match 標頭已被接受，但無法正常工作。</li> </ul> <p>版本控制</p> <p>要刪除特定版本，請求者必須是儲存桶擁有者並使用 versionId 子資源。使用此子資源將永久刪除該版本。如果 `versionId` 對應刪除標記，響應頭 `x-amz-delete-marker` 返回設定為 `true`。</p> <ul style="list-style-type: none"> <li>• 如果刪除物件時沒有 versionId 啟用版本控制的儲存桶上的子資源，會導致產生刪除標記。這 `versionId` 刪除標記的回傳方式是使用 `x-amz-version-id` 響應頭，以及 `x-amz-delete-marker` 響應頭返回設定為 `true`。</li> <li>• 如果刪除物件時沒有 versionId 在暫停版本控制的儲存桶上刪除子資源，將導致永久刪除已經存在的「空白」版本或「空」刪除標記，並產生新的「空」刪除標記。這 `x-amz-delete-marker` 響應頭返回設定為 `true`。</li> </ul> <p>注意：在某些情況下，一個物件可能存在多個刪除標記。</p> <p>看<a href="#">"使用 S3 REST API 設定 S3 物件鎖"</a>了解如何在治理模式下刪除物件版本。</p>
刪除對象  (以前稱為 DELETE 多個物件)	<p>多重身份驗證 (MFA) 和回應標頭 `x-amz-mfa` 不受支援。</p> <p>可以在同一個請求訊息中刪除多個物件。</p> <p>看<a href="#">"使用 S3 REST API 設定 S3 物件鎖"</a>了解如何在治理模式下刪除物件版本。</p>
刪除物件標記	<p>使用 `tagging` 子資源會從物件中刪除所有標籤。</p> <p>版本控制</p> <p>如果 versionId 如果請求中未指定查詢參數，則該操作將從版本化儲存桶中物件的最新版本中刪除所有標籤。如果物件的目前版本是刪除標記，則傳回「MethodNotAllowed」狀態，並 `x-amz-delete-marker` 響應頭設定為 `true`。</p>
取得對象	<a href="#">"取得對象"</a>

手術	執行
取得對象Acl	如果為帳戶提供了必要的存取憑證，則操作將傳回肯定回應以及物件擁有者的 ID、DisplayName 和 Permission，表示擁有人對該物件具有完全存取權。
獲取對象合法持有	"使用 S3 REST API 設定 S3 物件鎖"
取得對象保留	"使用 S3 REST API 設定 S3 物件鎖"
取得物件標記	<p>使用 `tagging` 子資源傳回物件的所有標籤。</p> <p>版本控制</p> <p>如果 `versionId` 如果請求中未指定查詢參數，則操作將傳回版本化儲存桶中物件最新版本的所有標籤。如果物件的目前版本是刪除標記，則傳回「MethodNotAllowed」狀態，並 `x-amz-delete-marker` 響應頭設定為 `true`。</p>
頭部對象	"頭部對象"
復原對象	"復原對象"
放置對象	"放置對象"
複製對象 (之前名為 PUT 物件 - 複製)	"複製對象"
放置對象合法保留	"使用 S3 REST API 設定 S3 物件鎖"
PutObjectRetention	"使用 S3 REST API 設定 S3 物件鎖"

手術	執行
PutObjectTagging	<p>使用 `tagging` 子資源會為現有物件新增一組標籤。</p> <p><b>對象標籤限制</b></p> <p>您可以在上傳新物件時為其新增標籤，也可以將其新增至現有物件中。StorageGRID和 Amazon S3 都支援每個物件最多 10 個標籤。與物件關聯的標籤必須具有唯一的標籤鍵。標籤鍵的長度最多為 128 個 Unicode 字符，標籤值的長度最多為 256 個 Unicode 字元。鍵和值區分大小寫。</p> <p><b>標籤更新和攝取行為</b></p> <p>當您使用 PutObjectTagging 更新物件的標籤時，StorageGRID不會重新擷取該物件。這表示符合的 ILM 規則中指定的 Ingest Behavior 選項未被使用。當 ILM 由正常後台 ILM 程序重新評估時，將對更新觸發的物件位置進行任何變更。</p> <p>這意味著，如果 ILM 規則對攝取行為使用嚴格選項，則當無法進行所需的物件放置時（例如，因為新需要的位置不可用），不會採取任何措施。更新後的物件將保留其目前位置，直到可以實現所需的位置。</p> <p><b>解決衝突</b></p> <p>衝突的客戶端請求（例如兩個客戶端寫入同一個金鑰）將根據「最新勝利」的原則解決。「最新勝利」評估的時間取決於StorageGRID系統完成給定請求的時間，而不是 S3 用戶端開始操作的時間。</p> <p><b>版本控制</b></p> <p>如果 versionId`如果請求中未指定查詢參數，則操作會將標籤新增至版本化儲存桶中物件的最新版本。如果物件的目前版本是刪除標記，則傳回「MethodNotAllowed」狀態，並 `x-amz-delete-marker`響應頭設定為 `true`。</p>
選擇對象內容	" <a href="#">選擇對象內容</a> "

## 使用 S3 Select

StorageGRID支援以下 Amazon S3 Select 子句、資料類型和運算符"[SelectObjectContent 指令](#)"。



任何未列出的項目均不受支援。

有關語法，請參閱"[選擇對象內容](#)"。有關 S3 Select 的更多信息，請參閱 "[S3 Select 的 AWS 文檔](#)"。

只有啟用了 S3 Select 的租用戶帳戶才能發出 SelectObjectContent 查詢。查看"[使用 S3 Select 的注意事項和要求](#)"。

### 條款

- [選擇清單](#)

- FROM 子句
- WHERE 子句
- LIMIT 子句

#### 資料類型

- 布林值
- 整數
- 細繩
- 漂浮
- 十進制，數字
- 時間戳

#### 運算符

##### 邏輯運算符

- 和
- 不是
- 或者

##### 比較運算符

- <
- >
- <=
- >=
- =
- =
- <>
- !=
- 之間
- 在

##### 模式匹配運算符

- 喜歡
- \_
- %

## 廬正算子

- 為空
- 不為空

## 數學運算符

- +
- -
- \*
- /
- %

StorageGRID遵循 Amazon S3 Select 運算子優先權。

## 聚合函數

- 平均值()
- 數數 (\*)
- 最大限度 ()
- MIN()
- 和 ()

## 條件函數

- 案件
- 合併
- NULLIF

## 轉換函數

- CAST (針對支援的資料類型)

## 日期函數

- DATE\_ADD
- 日期差異
- 提煉
- 至字串
- 到時間戳
- UTCNOW

## 字串函數

- 字元長度，字元長度
- 降低
- 子字串
- 修剪
- 上

## 使用伺服器端加密

伺服器端加密可讓您保護靜態的物件資料。StorageGRID在寫入物件時會對資料進行加密，並在您存取物件時解密資料。

如果您想使用伺服器端加密，您可以根據加密金鑰的管理方式選擇兩個互斥選項之一：

- **SSE**（使用StorageGRID管理金鑰的伺服器端加密）：當您發出 S3 請求來儲存物件時，StorageGRID會使用唯一金鑰加密該物件。當您發出 S3 請求來擷取物件時，StorageGRID會使用儲存的金鑰來解密該物件。
- **SSE-C**（使用客戶提供的金鑰進行伺服器端加密）：當您發出 S3 請求來儲存物件時，您需要提供自己的加密金鑰。當您檢索物件時，您會提供相同的加密金鑰作為請求的一部分。如果兩個加密金鑰匹配，則解密物件並傳回物件資料。

雖然StorageGRID管理所有物件加密和解密操作，但您必須管理您提供的加密金鑰。



您提供的加密金鑰永遠不會被儲存。如果遺失了加密金鑰，您就會遺失對應的物件。



如果使用 SSE 或 SSE-C 加密對象，則會忽略任何儲存桶層級或網格層級的加密設定。

## 使用 SSE

若要使用StorageGRID管理的唯一金鑰加密對象，請使用下列請求標頭：

```
x-amz-server-side-encryption
```

以下物件操作支援 SSE 請求頭：

- "放置對象"
- "複製對象"
- "建立多部分上傳"

## 使用 SSE-C

若要使用您管理的唯一密鑰加密對象，請使用三個請求標頭：

請求頭	描述
x-amz-server-side-encryption-customer-algorithm	指定加密算法。標頭值必須是 AES256。
x-amz-server-side-encryption-customer-key	指定用於加密或解密物件的加密金鑰。密鑰的值必須是 256 位元、base64 編碼的。
x-amz-server-side-encryption-customer-key-MD5	根據RFC 1321指定加密金鑰的MD5摘要，用於確保加密金鑰傳輸無誤。MD5 摘要的值必須是 base64 編碼的 128 位元。

以下物件操作支援 SSE-C 請求標頭：

- "取得對象"
- "頭部對象"
- "放置對象"
- "複製對象"
- "建立多部分上傳"
- "上傳部分"
- "上傳部分複製"

使用客戶提供的金鑰進行伺服器端加密 (SSE-C) 的注意事項

在使用 SSE-C 之前，請注意以下事項：

- 您必須使用 https。



使用 SSE-C 時，StorageGRID 會拒絕任何透過 http 發出的請求。出於安全考慮，您應將任何意外使用 http 發送的密鑰視為已洩露。丟棄鑰匙，並進行適當的旋轉。

- 響應中的 ETag 不是物件資料的 MD5。
- 您必須管理加密金鑰到物件的對應。StorageGRID 不會儲存加密金鑰。您有責任追蹤為每個物件提供的加密金鑰。
- 如果您的儲存桶啟用了版本控制，則每個物件版本都應該有自己的加密金鑰。您有責任追蹤每個物件版本所使用的加密金鑰。
- 因為您在客戶端管理加密金鑰，所以您還必須在客戶端管理任何其他保護措施，例如金鑰輪換。



您提供的加密金鑰永遠不會被儲存。如果遺失了加密金鑰，您就會遺失對應的物件。

- 如果為儲存桶配置了跨網格複製或 CloudMirror 複製，則無法擷取 SSE-C 物件。攝取操作將會失敗。

相關資訊

## 複製對象

您可以使用 S3 CopyObject 請求來建立已儲存在 S3 中的物件的副本。CopyObject 操作與執行 GetObject 後再執行 PutObject 操作相同。

### 解決衝突

衝突的客戶端請求（例如兩個客戶端寫入同一個金鑰）將根據「最新勝利」的原則解決。「最新勝利」評估的時間取決於StorageGRID系統完成給定請求的時間，而不是 S3 用戶端開始操作的時間。

### 物件大小

單一 PutObject 操作的最大\_建議\_大小為 5 GiB（5,368,709,120 位元組）。如果您有大於 5 GiB 的對象，請使用"分段上傳"反而。

單一 PutObject 操作支援的最大大小為 5 TiB（5,497,558,138,880 位元組）。



如果您從StorageGRID 11.6 或更早版本升級，則當您嘗試上傳超過 5 GiB 的物件時，將觸發 S3 PUT 物件大小過大警報。如果您新安裝了StorageGRID 11.7 或 11.8，則在這種情況下不會觸發警報。但是，為了與 AWS S3 標準保持一致，StorageGRID的未來版本將不支援上傳大於 5 GiB 的物件。

### 用戶元資料中的 UTF-8 字符

如果請求在使用者定義元資料的鍵名或值中包含（未轉義的）UTF-8 值，則StorageGRID行為未定義。

StorageGRID不會解析或解釋使用者定義元資料的鍵名或值中包含的轉義 UTF-8 字元。轉義的 UTF-8 字元將被視為 ASCII 字元：

- 如果使用者定義的元資料包含轉義的 UTF-8 字符，則請求成功。
- StorageGRID不回傳 `x-amz-missing-meta` 如果鍵名或值的解釋值包含不可列印的字符，則為標頭。

### 支援的請求標頭

支援以下請求標頭：

- Content-Type
- x-amz-copy-source
- x-amz-copy-source-if-match
- x-amz-copy-source-if-none-match
- x-amz-copy-source-if-unmodified-since
- x-amz-copy-source-if-modified-since
- x-amz-meta-，後面跟著包含使用者定義元資料的名稱-值對
- x-amz-metadata-directive：預設值為 COPY，它使您能夠複製物件和相關元資料。

您可以指定 `REPLACE` 在複製物件時覆寫現有的元數據，或更新物件元數據。

- `x-amz-storage-class`
- `x-amz-tagging-directive`：預設值為 `COPY`，它使您能夠複製物件和所有標籤。

您可以指定 `REPLACE` 在複製物件時覆寫現有標籤，或更新標籤。

- S3 物件鎖定請求標頭：

- `x-amz-object-lock-mode`
- `x-amz-object-lock-retain-until-date`
- `x-amz-object-lock-legal-hold`

如果發出的請求沒有這些標頭，則使用儲存桶預設保留設定來計算物件版本模式和保留截止日期。看"[使用 S3 REST API 設定 S3 物件鎖](#)"。

- SSE 請求標頭：

- `x-amz-copy-source-server-side-encryption-customer-algorithm`
- `x-amz-copy-source-server-side-encryption-customer-key`
- `x-amz-copy-source-server-side-encryption-customer-key-MD5`
- `x-amz-server-side-encryption`
- `x-amz-server-side-encryption-customer-key-MD5`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-algorithm`

看[\[伺服器端加密的請求標頭\]](#)

## 不支援的請求標頭

不支援以下請求標頭：

- `Cache-Control`
- `Content-Disposition`
- `Content-Encoding`
- `Content-Language`
- `Expires`
- `If-Match`

``If-Match header` 已被接受，但無法正常使用。`

- `If-None-Match`

`If-None-Match` header`已被接受，但無法正常使用。

- `x-amz-checksum-algorithm`

當您複製物件時，如果來源物件具有校驗和，StorageGRID不會將該校驗和值複製到新物件。無論您是否嘗試使用`x-amz-checksum-algorithm`在物件請求中。

- `x-amz-website-redirect-location`

## 儲存類別選項

這`x-amz-storage-class`支援請求標頭，並且會影響StorageGRID在符合的 ILM 規則使用雙重提交或平衡時建立的物件副本數量"[攝取選項](#)"。

- STANDARD

(預設) 當 ILM 規則使用雙重提交選項時，或當平衡選項回退到建立臨時副本時，指定雙重提交接收操作。

- REDUCED\_REDUNDANCY

當 ILM 規則使用雙重提交選項時，或當平衡選項回退到建立臨時副本時，指定單一提交攝取操作。



如果您將物件提取到啟用了 S3 物件鎖定的儲存桶中，則`REDUCED\_REDUNDANCY`選項被忽略。如果您將物件提取到舊版相容儲存桶中，`REDUCED\_REDUNDANCY`選項傳回錯誤。StorageGRID將始終執行雙重提交攝取以確保滿足合規性要求。

## 在 CopyObject 中使用 x-amz-copy-source

如果來源儲存桶和金鑰在`x-amz-copy-source`標頭與目標儲存桶和鍵不同，來源物件資料的副本將寫入目標。

如果來源和目標匹配，並且`x-amz-metadata-directive`標題指定為`REPLACE`，物件的元資料將使用請求中提供的元資料值進行更新。在這種情況下，StorageGRID不會重新擷取該物件。這有兩個重要後果：

- 您不能使用 CopyObject 來加密現有對象，或變更現有物件的加密。如果你提供`x-amz-server-side-encryption`標題或`x-amz-server-side-encryption-customer-algorithm`標題，StorageGRID拒絕請求並返回`XNotImplemented`。
- 不使用符合的 ILM 規則中指定的 Ingest Behavior 選項。當 ILM 由正常後台 ILM 程序重新評估時，將對更新觸發的物件位置進行任何變更。

這意味著，如果 ILM 規則對攝取行為使用嚴格選項，則當無法進行所需的物件放置時（例如，因為新需要的位置不可用），不會採取任何措施。更新後的物件將保留其目前位置，直到可以實現所需的位置。

## 伺服器端加密的請求標頭

如果你"[使用伺服器端加密](#)"，您提供的請求標頭取決於來源物件是否加密以及您是否計劃加密目標物件。

- 如果來源物件使用客戶提供的金鑰 (SSE-C) 加密，則必須在 CopyObject 請求中包含以下三個標頭，以便可以解密然後複製物件：

- `x-amz-copy-source-server-side-encryption-customer-algorithm`：指定 AES256。
- `x-amz-copy-source-server-side-encryption-customer-key`：指定您在建立來源物件時提供的加密金鑰。
- `x-amz-copy-source-server-side-encryption-customer-key-MD5`：指定您在建立來源物件時提供的 MD5 摘要。
- 如果要使用您提供和管理的唯一密鑰加密目標物件（副本），請包含以下三個標頭：
  - `x-amz-server-side-encryption-customer-algorithm`：指定 AES256。
  - `x-amz-server-side-encryption-customer-key`：為目標對象指定新的加密金鑰。
  - `x-amz-server-side-encryption-customer-key-MD5`：指定新加密金鑰的 MD5 摘要。



您提供的加密金鑰永遠不會被儲存。如果遺失了加密金鑰，您就會遺失對應的物件。在使用客戶提供的金鑰保護物件資料之前，請先查看以下注意事項"使用伺服器端加密"。

- 如果要使用StorageGRID (SSE) 管理的唯一金鑰加密目標物件（副本），請在 `CopyObject` 請求中包含此標頭：
  - `x-amz-server-side-encryption`



這 `server-side-encryption` 物件的值無法更新。相反，用新的 `server-side-encryption` 價值使用 `x-amz-metadata-directive: REPLACE`。

## 版本控制

如果來源儲存桶已版本化，則可以使用 `x-amz-copy-source` 標頭來複製物件的最新版本。要複製物件的特定版本，必須使用 `versionId` 子資源。如果目標儲存桶有版本控制，則產生的版本將傳回 `x-amz-version-id` 響應標頭。如果目標儲存桶的版本控制已暫停，則 `x-amz-version-id` 傳回 "null" 值。

## 取得對象

您可以使用 S3 `GetObject` 請求從 S3 儲存桶中擷取物件。

### GetObject 和多部分對象

您可以使用 `partNumber` 請求參數來檢索多部分或分段物件的特定部分。這 `x-amz-mp-parts-count` 響應元素指示物件有多少個部分。

您可以設定 `partNumber` 對於分段/多部分對象和非分段/非多部分對象，皆為 1；但是，`x-amz-mp-parts-count` 響應元素僅針對分段或多部分物件傳回。

### 用戶元資料中的 UTF-8 字符

StorageGRID 不會解析或解釋使用者定義元資料中的轉義 UTF-8 字元。對使用者定義元資料中含有轉義 UTF-8 字元的物件進行 GET 請求不會傳回 `x-amz-missing-meta` 如果鍵名或值包含不可列印的字符，則為標頭。

### 支援的請求標頭

支援以下請求標頭：

- `x-amz-checksum-mode`：指定 `ENABLED`

這 `Range` 標頭不支援 `x-amz-checksum-mode` 用於取得物件。當你包括 `Range` 在請求中 `x-amz-checksum-mode` 啟用後，StorageGRID 不會在回應中傳回校驗和值。

### 不支援的請求標頭

不支援以下請求標頭並返回 `XNotImplemented`：

- `x-amz-website-redirect-location`

### 版本控制

如果 `versionId` 如果未指定子資源，則操作將取得版本化儲存桶中物件的最新版本。如果物件的目前版本是刪除標記，則傳回「未找到」狀態，並 `x-amz-delete-marker` 響應頭設定為 `true`。

### 使用客戶提供的加密金鑰進行伺服器端加密的請求標頭 (SSE-C)

如果物件使用您提供的唯一密鑰加密，請使用所有三個標頭。

- `x-amz-server-side-encryption-customer-algorithm`：指定 `AES256`。
- `x-amz-server-side-encryption-customer-key`：指定物件的加密金鑰。
- `x-amz-server-side-encryption-customer-key-MD5`：指定物件加密金鑰的 MD5 摘要。



您提供的加密金鑰永遠不會被儲存。如果遺失了加密金鑰，您就會遺失對應的物件。在使用客戶提供的金鑰保護物件資料之前，請先查看["使用伺服器端加密"](#)。

### 雲端儲存池物件的 `GetObject` 行為

如果一個物件被儲存在"[雲端儲存池](#)"，`GetObject` 請求的行為取決於物件的狀態。看"[頭部對象](#)"了解更多詳情。



如果物件儲存在雲端儲存池中，並且該物件的一個或多個副本也存在於網格中，則 `GetObject` 請求將嘗試從網格中檢索數據，然後再從雲端儲存池中檢索資料。

物體狀態	<code>GetObject</code> 的行為
物件已匯入StorageGRID但尚未經過 ILM 評估，或儲存在傳統儲存池中或使用擦除編碼的對象	200 OK 檢索物件的副本。
物件位於雲端儲存池中，但尚未轉換為不可檢索狀態	200 OK 檢索物件的副本。
物件已轉換為不可檢索狀態	403 Forbidden , InvalidObjectState 使用" <a href="#">復原對象</a> "請求將物件恢復到可檢索狀態。

物體狀態	GetObject 的行為
物件正在從不可恢復狀態恢復	403 Forbidden , InvalidObjectState 等待 RestoreObject 請求完成。
物件已完全恢復到雲端儲存池	200 OK 檢索物件的副本。

雲端儲存池中的多部分或分段對象

如果您上傳了多部分對象，或者StorageGRID將大型對象拆分為多個段，StorageGRID會透過對該對象的部分或段的子集進行採樣來確定該對象是否在雲端儲存池中可用。在某些情況下，GetObject 請求可能會錯誤地傳回 `200 OK` 當物件的某些部分已轉換為不可檢索狀態或物件的某些部分尚未復原時。

在這些情況下：

- GetObject 請求可能會傳回一些數據，但在傳輸過程中停止。
- 後續的 GetObject 請求可能會傳回 403 Forbidden。

### GetObject 和跨網格複製

如果您正在使用"網格聯盟"和"跨網格複製"為儲存桶啟用後，S3 用戶端可以透過發出 GetObject 請求來驗證物件的複製狀態。響應包括StorageGRID特定的 `x-ntap-sg-cgr-replication-status` 響應標頭，它將具有以下值之一：

網格	複製狀態
來源	<ul style="list-style-type: none"> <li>• 已完成：複製成功。</li> <li>• 待定：物件尚未被複製。</li> <li>• 失敗：複製失敗，並發生永久性故障。使用者必須解決該錯誤。</li> </ul>
目的地	<b>REPLICA</b> ：物件已從來源網格複製。



StorageGRID不支援 `x-amz-replication-status` 標頭。

### 頭部對象

您可以使用 S3 HeadObject 請求從物件中檢索元數據，而不回傳物件本身。如果物件儲存在雲端儲存池中，則可以使用 HeadObject 來確定物件的過渡狀態。

### HeadObject 和多部分對象

您可以使用 `partNumber` 請求參數來檢索多部分或分段物件的特定部分的元資料。這 `x-amz-mp-parts-count` 響應元素指示物件有多少個部分。

您可以設定 `partNumber` 對於分段/多部分對象和非分段/非多部分對象，皆為 1；但是，`x-amz-mp-parts-

count` 響應元素僅針對分段或多部分物件傳回。

用戶元資料中的 **UTF-8** 字符

StorageGRID 不會解析或解釋使用者定義元資料中的轉義 UTF-8 字元。對使用者定義元資料中含有轉義 UTF-8 字元的物件進行 HEAD 請求不會傳回 `x-amz-missing-meta` 如果鍵名或值包含不可列印的字符，則為標頭。

支援的請求標頭

支援以下請求標頭：

- `x-amz-checksum-mode`

這 `partNumber` 參數和 `Range` 標頭不支援 `x-amz-checksum-mode` 對於 HeadObject。當您將它們包含在請求中時 `x-amz-checksum-mode` 啟用後，StorageGRID 不會在回應中傳回校驗和值。

不支援的請求標頭

以下請求標頭不受支援並返回 XNotImplemented：

- `x-amz-website-redirect-location`

版本控制

如果 `versionId` 如果未指定子資源，則操作將取得版本化儲存桶中物件的最新版本。如果物件的目前版本是刪除標記，則傳回「未找到」狀態，並 `x-amz-delete-marker` 響應頭設定為 `true`。

使用客戶提供的加密金鑰進行伺服器端加密的請求標頭 (**SSE-C**)

如果物件使用您提供的唯一密鑰加密，請使用所有這三個標頭。

- `x-amz-server-side-encryption-customer-algorithm`：指定 AES256。
- `x-amz-server-side-encryption-customer-key`：指定物件的加密金鑰。
- `x-amz-server-side-encryption-customer-key-MD5`：指定物件加密金鑰的 MD5 摘要。



您提供的加密金鑰永遠不會被儲存。如果遺失了加密金鑰，您就會遺失對應的物件。在使用客戶提供的金鑰保護物件資料之前，請先查看["使用伺服器端加密"](#)。

**Cloud Storage Pool** 物件的 HeadObject 回應

如果物件儲存在["雲端儲存池"](#)，將返回以下響應標頭：

- `x-amz-storage-class`: GLACIER
- `x-amz-restore`

回應標頭提供有關物件狀態的信息，因為它被移動到雲端儲存池，可選地轉換為不可檢索狀態，並恢復。

物體狀態	對 <b>HeadObject</b> 的回應
物件已匯入StorageGRID但尚未經過 ILM 評估，或儲存在傳統儲存池中或使用擦除編碼的對象	200 OK (沒有回傳特殊的回應頭。)
物件位於雲端儲存池中，但尚未轉換為不可檢索狀態	200 OK  x-amz-storage-class: GLACIER  x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"  在物件轉換為不可檢索狀態之前，`expiry-date`設定為未來某個遙遠的時間。轉換的確切時間不受StorageGRID系統控制。
物件已轉換為不可檢索狀態，但網格上還存在至少一個副本	200 OK  x-amz-storage-class: GLACIER  x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"  價值 `expiry-date` 設定為未來某個遙遠的時間。  注意：如果網格上的副本不可用（例如，儲存節點關閉），則必須發出"復原對象"請求從雲端儲存池恢復副本，然後才能成功檢索物件。
物件已轉換為不可檢索狀態，且網格上不存在副本	200 OK  x-amz-storage-class: GLACIER
物件正在從不可恢復狀態恢復	200 OK  x-amz-storage-class: GLACIER  x-amz-restore: ongoing-request="true"

物體狀態	對 <b>HeadObject</b> 的回應
物件已完全恢復到雲端儲存池	<pre>200 OK  x-amz-storage-class: GLACIER  x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2018 00:00:00 GMT"  這 `expiry-date` 指示雲端儲存池中的物件何時將返回不可檢索狀態。</pre>

#### 雲端儲存池中的多部分或分段對象

如果您上傳了多部分對象，或者StorageGRID將大型對象拆分為多個段，StorageGRID會透過對該對象的部分或段的子集進行採樣來確定該對象是否在雲端儲存池中可用。在某些情況下，HeadObject 請求可能會錯誤地傳回 `x-amz-restore: ongoing-request="false"` 當物件的某些部分已轉換為不可檢索狀態或物件的某些部分尚未復原時。

#### HeadObject 和跨網格複製

如果您正在使用"網格聯盟"和"跨網格複製"為儲存桶啟用後，S3 用戶端可以透過發出 HeadObject 請求來驗證物件的複製狀態。響應包括StorageGRID特定的 `x-ntap-sg-cgr-replication-status` 響應標頭，它將具有以下值之一：

網格	複製狀態
來源	<ul style="list-style-type: none"> <li>• 已完成：複製成功。</li> <li>• 待定：物件尚未被複製。</li> <li>• 失敗：複製失敗，並發生永久性故障。使用者必須解決該錯誤。</li> </ul>
目的地	<b>REPLICA</b> ：物件已從來源網格複製。



StorageGRID不支援 `x-amz-replication-status` 標頭。

#### 放置對象

您可以使用 S3 PutObject 請求將物件新增至儲存桶。

#### 解決衝突

衝突的客戶端請求（例如兩個客戶端寫入同一個金鑰）將根據「最新勝利」的原則解決。「最新勝利」評估的時間取決於StorageGRID系統完成給定請求的時間，而不是 S3 用戶端開始操作的時間。

#### 物件大小

單一 PutObject 操作的最大\_建議\_大小為 5 GiB (5,368,709,120 位元組)。如果您有大於 5 GiB 的對象，請使

用"分段上傳"反而。

單一 PutObject 操作支援的最大大小為 5 TiB (5,497,558,138,880 位元組)。



如果您從StorageGRID 11.6 或更早版本升級，則當您嘗試上傳超過 5 GiB 的物件時，將觸發 S3 PUT 物件大小過大警報。如果您新安裝了StorageGRID 11.7 或 11.8，則在這種情況下不會觸發警報。但是，為了與 AWS S3 標準保持一致，StorageGRID的未來版本將不支援上傳大於 5 GiB 的物件。

## 用戶元資料大小

Amazon S3 將每個 PUT 請求標頭內的使用者定義元資料的大小限制為 2 KB。StorageGRID將用戶元資料限制為 24 KiB。使用者定義元資料的大小是透過計算每個鍵和值的 UTF-8 編碼的位元組數總和來衡量的。

## 用戶元資料中的 UTF-8 字符

如果請求在使用者定義元資料的鍵名或值中包含（未轉義的）UTF-8 值，則StorageGRID行為未定義。

StorageGRID不會解析或解釋使用者定義元資料的鍵名或值中包含的轉義 UTF-8 字元。轉義的 UTF-8 字元將被視為 ASCII 字元：

- 如果使用者定義的元資料包含轉義的 UTF-8 字符，則 PutObject、CopyObject、GetObject 和 HeadObject 請求會成功。
- StorageGRID不回傳 `x-amz-missing-meta` 如果鍵名或值的解釋值包含不可列印的字符，則為標頭。

## 對象標籤限制

您可以在上傳新物件時為其新增標籤，也可以將其新增至現有物件中。StorageGRID和 Amazon S3 都支援每個物件最多 10 個標籤。與物件關聯的標籤必須具有唯一的標籤鍵。標籤鍵的長度最多為 128 個 Unicode 字符，標籤值的長度最多為 256 個 Unicode 字元。鍵和值區分大小寫。

## 物件所有權

在StorageGRID中，所有物件都歸儲存桶擁有者帳戶所有，包括由非擁有者帳戶或匿名使用者建立的物件。

## 支援的請求標頭

支援以下請求標頭：

- Cache-Control
- Content-Disposition
- Content-Encoding

當您指定 `aws-chunked` 為了Content-EncodingStorageGRID不驗證下列項目：

- StorageGRID不驗證 `chunk-signature` 針對區塊資料。
- StorageGRID不會驗證您提供的值 `x-amz-decoded-content-length` 針對該對象。
- Content-Language

- Content-Length
- Content-MD5
- Content-Type
- Expires
- Transfer-Encoding

如果支援分塊傳輸編碼 `aws-chunked` 也使用了有效載荷簽名。

- x-amz-checksum-sha256
- x-amz-meta-，後面跟著包含使用者定義元資料的名稱-值對。

為使用者定義的元資料指定名稱-值對時，請使用下列通用格式：

```
x-amz-meta-name: value
```

如果要使用 使用者定義建立時間 選項作為 ILM 規則的參考時間，則必須使用 `creation-time` 作為記錄物件建立時間的元資料的名稱。例如：

```
x-amz-meta-creation-time: 1443399726
```

價值 `creation-time` 以 1970 年 1 月 1 日以後的秒數計算。



ILM 規則不能同時使用 使用者定義的創建時間 作為參考時間以及平衡或嚴格的攝取選項。建立 ILM 規則時傳回錯誤。

- x-amz-tagging
- S3 物件鎖定請求標頭
  - x-amz-object-lock-mode
  - x-amz-object-lock-retain-until-date
  - x-amz-object-lock-legal-hold

如果發出的請求沒有這些標頭，則使用儲存桶預設保留設定來計算物件版本模式和保留截止日期。看["使用 S3 REST API 設定 S3 物件鎖"](#)。

- SSE 請求標頭：
  - x-amz-server-side-encryption
  - x-amz-server-side-encryption-customer-key-MD5
  - x-amz-server-side-encryption-customer-key
  - x-amz-server-side-encryption-customer-algorithm

看[\[伺服器端加密的請求標頭\]](#)

## 不支援的請求標頭

不支援以下請求標頭：

- If-Match

``If-Match header`` 已被接受，但無法正常使用。

- If-None-Match

``If-None-Match header`` 已被接受，但無法正常使用。

- x-amz-acl
- x-amz-sdk-checksum-algorithm
- x-amz-trailer
- x-amz-website-redirect-location

這 `x-amz-website-redirect-location`` 標題返回 ``XNotImplemented``。

## 儲存類別選項

這 ``x-amz-storage-class`` 支援請求標頭。提交的價值 ``x-amz-storage-class`` 影響 StorageGRID 在攝取期間如何保護物件數據，而不是影響 StorageGRID 系統中儲存了多少個物件的持久副本（由 ILM 決定）。

如果與已攝取物件相符的 ILM 規則使用嚴格攝取選項，``x-amz-storage-class`` 標頭無效。

以下值可用於 `x-amz-storage-class``：

- STANDARD(預設)
  - 雙重提交：如果 ILM 規則為攝取行為指定了雙重提交選項，則一旦攝取對象，就會建立該對象的第二個副本並分發到不同的儲存節點（雙重提交）。在評估 ILM 時，StorageGRID 會決定這些初始臨時副本是否符合規則中的放置說明。如果沒有，則可能需要在不同位置製作新的物件副本，並且可能需要刪除初始臨時副本。
  - 平衡：如果 ILM 規則指定了平衡選項，且 StorageGRID 無法立即製作規則中指定的所有副本，StorageGRID 會在不同的儲存節點上製作兩個暫存副本。

如果 StorageGRID 可以立即建立 ILM 規則中指定的所有物件副本（同步放置），則 ``x-amz-storage-class`` 標頭無效。

- REDUCED\_REDUNDANCY
  - 雙重提交：如果 ILM 規則為攝取行為指定了雙重提交選項，則 StorageGRID 會在攝取物件時建立單一暫存副本（單次提交）。
  - 平衡：如果 ILM 規則指定了平衡選項，則僅當系統無法立即製作規則中指定的所有副本時，StorageGRID 才會製作單一暫存副本。如果 StorageGRID 可以執行同步放置，則此標頭無效。這

`REDUCED\_REDUNDANCY` 當與物件相符的 ILM 規則建立單一複製副本時，最好使用此選項。在這種情況下使用 `REDUCED\_REDUNDANCY` 消除了每次攝取操作時不必要的額外物件副本的建立和刪除。

使用 `REDUCED\_REDUNDANCY` 在其他情況下不建議選擇此選項。`REDUCED\_REDUNDANCY` 增加了攝取過程中物件資料遺失的風險。例如，如果單一副本最初儲存在儲存節點上，而該儲存節點在 ILM 評估發生之前發生故障，則您可能會遺失資料。



任何時間段內只有一個複製副本會使資料面臨永久遺失的風險。如果某個物件的副本只有一個，則當儲存節點發生故障或發生重大錯誤時，該物件將會遺失。在升級等維護過程中，您也會暫時失去對該物件的存取權限。

指定 `REDUCED\_REDUNDANCY` 僅影響首次攝取物件時所建立的副本數量。它不會影響活動 ILM 策略評估物件時產生的物件副本數量，也不會導致資料在 StorageGRID 系統中以較低的冗餘層級進行儲存。



如果您將物件提取到啟用了 S3 物件鎖定的儲存桶中，則 `REDUCED\_REDUNDANCY` 選項被忽略。如果您將物件提取到舊版相容儲存桶中，`REDUCED\_REDUNDANCY` 選項傳回錯誤。StorageGRID 將始終執行雙重提交攝取以確保滿足合規性要求。

## 伺服器端加密的請求標頭

您可以使用下列請求標頭來使用伺服器端加密來加密物件。SSE 和 SSE-C 選項是互斥的。

- **SSE**：如果您想要使用由 StorageGRID 管理的唯一金鑰加密對象，請使用下列標頭。

- `x-amz-server-side-encryption`

當 `x-amz-server-side-encryption` 標頭未包含在 PutObject 請求中，網格範圍的"儲存物件加密設定"從 PutObject 回應中省略。

- **SSE-C**：如果您想使用您提供和管理的唯一金鑰加密對象，請使用所有這三個標頭。

- `x-amz-server-side-encryption-customer-algorithm`：指定 AES256。

- `x-amz-server-side-encryption-customer-key`：為新物件指定加密金鑰。

- `x-amz-server-side-encryption-customer-key-MD5`：指定新物件的加密金鑰的 MD5 摘要。



您提供的加密金鑰永遠不會被儲存。如果遺失了加密金鑰，您就會遺失對應的物件。在使用客戶提供的金鑰保護物件資料之前，請先查看以下注意事項"使用伺服器端加密"。



如果使用 SSE 或 SSE-C 加密對象，則會忽略任何儲存桶層級或網格層級的加密設定。

## 版本控制

如果為儲存桶啟用了版本控制，則唯一的 `versionId` 針對所儲存物件的版本會自動產生。這 `versionId` 也會在回應中返回 `x-amz-version-id` 響應頭。

如果版本控制暫停，則物件版本將以空值儲存 `versionId` 如果空版本已經存在，它將被覆蓋。

## 授權標頭的簽章計算

使用時 `Authorization` 標頭來驗證請求，StorageGRID 與 AWS 有以下不同：

- StorageGRID 不需要 `host` 標題包含在 ``CanonicalHeaders``。
- StorageGRID 不需要 `Content-Type` 被納入 ``CanonicalHeaders``。
- StorageGRID 不需要 `x-amz-*` 標題包含在 ``CanonicalHeaders``。



作為一般的最佳做法，請始終將這些標題包含在 ``CanonicalHeaders`` 以確保它們已被驗證；但是，如果排除這些標題，StorageGRID 不會傳回錯誤。

有關詳細信息，請參閱 ["授權標頭的簽章計算：在單一區塊中傳輸有效負載 \(AWS 簽章版本 4\)"](#)。

### 相關資訊

- ["使用 ILM 管理對象"](#)
- ["Amazon Simple Storage Service API 參考：PutObject"](#)

## 復原對象

您可以使用 S3 `RestoreObject` 請求來還原儲存在雲端儲存池中的物件。

### 支援的請求類型

StorageGRID 僅支援 `RestoreObject` 請求來還原物件。它不支援 `SELECT` 修復類型。選擇請求返回 ``XNotImplemented``。

### 版本控制

(可選) 指定 `versionId` 還原受版本控制的儲存桶中物件的特定版本。如果你沒有指定 ``versionId``，該物件的最新版本已恢復。

### `RestoreObject` 在雲端儲存池物件上的行為

如果一個物件被儲存在 ["雲端儲存池"](#)，根據物件的狀態，`RestoreObject` 請求具有以下行為。看 ["頭部對象"](#) 了解更多詳情。



如果物件儲存在雲端儲存池中，且該物件的一個或多個副本也存在於網格上，則無需透過發出 `RestoreObject` 請求來還原該物件。相反，可以使用 `GetObject` 請求直接檢索本機副本。

物體狀態	<code>RestoreObject</code> 的行為
物件已匯入 StorageGRID 但尚未由 ILM 評估，或物件不在雲端儲存池中	403 Forbidden, InvalidObjectState

物體狀態	RestoreObject 的行為
物件位於雲端儲存池中，但尚未轉換為不可檢索狀態	<p>`200 OK` 沒有做出任何改變。</p> <p>注意：在物件轉換為不可檢索狀態之前，您無法變更其 <code>expiry-date</code>。</p>
物件已轉換為不可檢索狀態	<p>`202 Accepted` 將物件的可檢索副本還原到雲端儲存池，保留天數為請求正文中指定的天數。在此期限結束時，物件將返回到不可檢索的狀態。</p> <p>或者，使用 <code>Tier</code> 請求元素來決定還原作業需要多長時間才能完成（`Expedited`，`Standard`，或者 `Bulk`）。如果你沒有指定 <code>Tier</code>，這 `Standard` 層被使用。</p> <p>重要：如果物件已轉換到 S3 Glacier Deep Archive 或雲端儲存池使用 Azure Blob 存儲，則無法使用 <code>Expedited</code> 層。傳回以下錯誤 <code>403 Forbidden, InvalidTier: Retrieval option is not supported by this storage class</code>。</p>
物件正在從不可恢復狀態恢復	409 Conflict, RestoreAlreadyInProgress
物件已完全恢復到雲端儲存池	<p>200 OK</p> <p>*注意：*如果物件已恢復到可檢索狀態，您可以變更其 <code>expiry-date</code> 透過重新發出 <code>RestoreObject</code> 請求並使用新值 <code>Days</code>。恢復日期根據請求的時間進行更新。</p>

## 選擇對象內容

您可以使用 S3 `SelectObjectContent` 請求根據簡單的 SQL 語句過濾 S3 物件的內容。

有關詳細信息，請參閱 ["Amazon Simple Storage Service API 參考：SelectObjectContent"](#)。

### 開始之前

- 租用戶帳戶具有 S3 `Select` 權限。
- 你有 `s3:GetObject` 您要查詢的物件的權限。
- 您要查詢的物件必須採用以下格式之一：
  - **CSV**。可原樣使用或壓縮為 GZIP 或 BZIP2 檔案。
  - 鑲木地板。Parquet 物件的附加要求：
    - S3 Select 僅支援使用 GZIP 或 Snappy 進行列壓縮。S3 Select 不支援 Parquet 物件的整個物件壓縮。
    - S3 Select 不支援 Parquet 輸出。您必須將輸出格式指定為 CSV 或 JSON。
    - 未壓縮的行組最大大小為 512 MB。
    - 您必須使用物件模式中指定的資料類型。
    - 您不能使用 INTERVAL、JSON、LIST、TIME 或 UUID 邏輯類型。

- 您的 SQL 表達式的最大長度為 256 KB。
- 輸入或結果中的任何記錄的最大長度為 1 MiB。

### CSV 請求語法範例

```
POST /{Key+}?select&select-type=2 HTTP/1.1
Host: Bucket.s3.abc-company.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<SelectObjectContentRequest xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Expression>string</Expression>
  <ExpressionType>string</ExpressionType>
  <RequestProgress>
    <Enabled>boolean</Enabled>
  </RequestProgress>
  <InputSerialization>
    <CompressionType>GZIP</CompressionType>
    <CSV>
      <AllowQuotedRecordDelimiter>boolean</AllowQuotedRecordDelimiter>
      <Comments>#</Comments>
      <FieldDelimiter>\t</FieldDelimiter>
      <FileHeaderInfo>USE</FileHeaderInfo>
      <QuoteCharacter>'</QuoteCharacter>
      <QuoteEscapeCharacter>\\</QuoteEscapeCharacter>
      <RecordDelimiter>\n</RecordDelimiter>
    </CSV>
  </InputSerialization>
  <OutputSerialization>
    <CSV>
      <FieldDelimiter>string</FieldDelimiter>
      <QuoteCharacter>string</QuoteCharacter>
      <QuoteEscapeCharacter>string</QuoteEscapeCharacter>
      <QuoteFields>string</QuoteFields>
      <RecordDelimiter>string</RecordDelimiter>
    </CSV>
  </OutputSerialization>
  <ScanRange>
    <End>long</End>
    <Start>long</Start>
  </ScanRange>
</SelectObjectContentRequest>
```

## Parquet 請求語法範例

```
POST /{Key+}?select&select-type=2 HTTP/1.1
Host: Bucket.s3.abc-company.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<SelectObjectContentRequest xmlns=http://s3.amazonaws.com/doc/2006-03-01/>
  <Expression>string</Expression>
  <ExpressionType>string</ExpressionType>
  <RequestProgress>
    <Enabled>boolean</Enabled>
  </RequestProgress>
  <InputSerialization>
    <CompressionType>GZIP</CompressionType>
    <PARQUET>
    </PARQUET>
  </InputSerialization>
  <OutputSerialization>
    <CSV>
      <FieldDelimiter>string</FieldDelimiter>
      <QuoteCharacter>string</QuoteCharacter>
      <QuoteEscapeCharacter>string</QuoteEscapeCharacter>
      <QuoteFields>string</QuoteFields>
      <RecordDelimiter>string</RecordDelimiter>
    </CSV>
  </OutputSerialization>
  <ScanRange>
    <End>long</End>
    <Start>long</Start>
  </ScanRange>
</SelectObjectContentRequest>
```

## SQL 查詢範例

此查詢取得州名稱、2010 年人口、估計 2015 年人口以及美國人口普查資料的變化百分比。文件中非狀態的記錄將被忽略。

```
SELECT STNAME, CENSUS2010POP, POPESTIMATE2015, CAST((POPESTIMATE2015 -
CENSUS2010POP) AS DECIMAL) / CENSUS2010POP * 100.0 FROM S3Object WHERE
NAME = STNAME
```

要查詢的文件的前幾行，SUB-EST2020\_ALL.csv，看起來像這樣：

```
SUMLEV, STATE, COUNTY, PLACE, COUSUB, CONCIT, PRIMGEO_FLAG, FUNCSTAT, NAME, STNAME,
CENSUS2010POP,
ESTIMATESBASE2010, POPESTIMATE2010, POPESTIMATE2011, POPESTIMATE2012, POPESTIM
ATE2013, POPESTIMATE2014,
POPESTIMATE2015, POPESTIMATE2016, POPESTIMATE2017, POPESTIMATE2018, POPESTIMAT
E2019, POPESTIMATE042020,
POPESTIMATE2020
040, 01, 000, 00000, 00000, 00000, 0, A, Alabama, Alabama, 4779736, 4780118, 4785514, 4
799642, 4816632, 4831586,
4843737, 4854803, 4866824, 4877989, 4891628, 4907965, 4920706, 4921532
162, 01, 000, 00124, 00000, 00000, 0, A, Abbeville
city, Alabama, 2688, 2705, 2699, 2694, 2645, 2629, 2610, 2602,
2587, 2578, 2565, 2555, 2555, 2553
162, 01, 000, 00460, 00000, 00000, 0, A, Adamsville
city, Alabama, 4522, 4487, 4481, 4474, 4453, 4430, 4399, 4371,
4335, 4304, 4285, 4254, 4224, 4211
162, 01, 000, 00484, 00000, 00000, 0, A, Addison
town, Alabama, 758, 754, 751, 750, 745, 744, 742, 734, 734, 728,
725, 723, 719, 717
```

## AWS-CLI 使用範例 (CSV)

```
aws s3api select-object-content --endpoint-url https://10.224.7.44:10443
--no-verify-ssl --bucket 619c0755-9e38-42e0-a614-05064f74126d --key SUB-
EST2020_ALL.csv --expression-type SQL --input-serialization '{"CSV":
{"FileHeaderInfo": "USE", "Comments": "#", "QuoteEscapeCharacter": "\"",
"RecordDelimiter": "\n", "FieldDelimiter": ",", "QuoteCharacter": "\"",
"AllowQuotedRecordDelimiter": false}, "CompressionType": "NONE"}' --output
-serialization '{"CSV": {"QuoteFields": "ASNEEDED",
"QuoteEscapeCharacter": "#", "RecordDelimiter": "\n", "FieldDelimiter":
",", "QuoteCharacter": "\""}}' --expression "SELECT STNAME, CENSUS2010POP,
POPESTIMATE2015, CAST((POPESTIMATE2015 - CENSUS2010POP) AS DECIMAL) /
CENSUS2010POP * 100.0 FROM S3Object WHERE NAME = STNAME" changes.csv
```

輸出文件的前幾行，changes.csv，看起來像這樣：

```
Alabama, 4779736, 4854803, 1.5705260708959658022953568983726297854
Alaska, 710231, 738430, 3.9703983633493891424057806544631253775
Arizona, 6392017, 6832810, 6.8959922978928247531256565807005832431
Arkansas, 2915918, 2979732, 2.1884703204959810255295244928012378949
California, 37253956, 38904296, 4.4299724839960620557988526104449148971
Colorado, 5029196, 5454328, 8.4532796097030221132761578590295546246
```

## AWS-CLI 使用範例 (Parquet)

```
aws s3api select-object-content --endpoint-url https://10.224.7.44:10443
--bucket 619c0755-9e38-42e0-a614-05064f74126d --key SUB-
EST2020_ALL.parquet --expression "SELECT STNAME, CENSUS2010POP,
POPESTIMATE2015, CAST((POPESTIMATE2015 - CENSUS2010POP) AS DECIMAL) /
CENSUS2010POP * 100.0 FROM S3Object WHERE NAME = STNAME" --expression-type
'SQL' --input-serialization '{"Parquet":{}}' --output-serialization
'{"CSV":{}}' changes.csv
```

輸出檔 changes.csv 的前幾行如下：

```
Alabama,4779736,4854803,1.5705260708959658022953568983726297854
Alaska,710231,738430,3.9703983633493891424057806544631253775
Arizona,6392017,6832810,6.8959922978928247531256565807005832431
Arkansas,2915918,2979732,2.1884703204959810255295244928012378949
California,37253956,38904296,4.4299724839960620557988526104449148971
Colorado,5029196,5454328,8.4532796097030221132761578590295546246
```

## 分段上傳的操作

### 分段上傳的操作

本節介紹StorageGRID如何支援分段上傳操作。

以下條件和注意事項適用於所有分段上傳操作：

- 單一儲存桶的並發分段上傳不應超過 1,000 個，因為該儲存桶的 ListMultipartUploads 查詢結果可能會傳回不完整的結果。
- StorageGRID對多部分元件強制實施 AWS 大小限制。S3 用戶端必須遵循以下準則：
  - 分段上傳中的每個部分必須介於 5 MiB (5,242,880 位元組) 和 5 GiB (5,368,709,120 位元組) 之間。
  - 最後一部分可以小於 5 MiB (5,242,880 位元組)。
  - 一般來說，零件尺寸應盡可能大。例如，對於 100 GiB 的對象，使用 5 GiB 的部分大小。由於每個部分都被視為唯一對象，因此使用較大部分大小可以減少StorageGRID元資料開銷。
  - 對於小於 5 GiB 的對象，請考慮使用非分段上傳。
- 如果 ILM 規則使用「平衡」或「嚴格」模式，則在分段上傳完成時，將對分段物件的每個部分進行 ILM 評估；如果 ILM 規則使用「平衡」或「嚴格」模式，則將對整個物件進行 ILM 評估。["攝取選項"](#)。您應該了解這會影響物件和部件的放置：
  - 如果在 S3 分段上傳過程中 ILM 發生變化，則分段上傳完成時物件的某些部分可能不符合目前的 ILM 要求。任何未正確放置的部件都會排隊等待 ILM 重新評估，然後移動到正確的位置。
  - 在評估某個零件的 ILM 時，StorageGRID會根據該零件的大小進行過濾，而不是物件的大小。這意味著物件的各個部分可以儲存在不滿足物件整體的 ILM 要求的位置。例如，如果規則指定所有 10 GB 或更大

的物件都儲存在 DC1，而所有較小的物件都儲存在 DC2，則 10 部分分段上傳的每個 1 GB 部分在攝取時都儲存在 DC2。但是，當對整個物件進行 ILM 評估時，物件的所有部分都會移至 DC1。

- 所有分段上傳操作都支援StorageGRID"一致性值"。
- 當使用分段上傳提取物件時，"物件分割閾值（1 GiB）"不適用。
- 根據需要，您可以使用"伺服器端加密"使用分段上傳。若要使用 SSE（使用StorageGRID管理金鑰的伺服器端加密），您需要包含 `x-amz-server-side-encryption` 僅在 CreateMultipartUpload 請求中的請求標頭。若要使用 SSE-C（使用客戶提供的金鑰的伺服器端加密），您需要在 CreateMultipartUpload 請求和每個後續 UploadPart 請求中指定相同的三個加密金鑰請求標頭。

手術	執行
中止分段上傳	使用所有 Amazon S3 REST API 行為實作。如有變更，恕不另行通知。
完成多部分上傳	看"完成多部分上傳"
建立多部分上傳 (之前名為「啟動分段上傳」)	看"建立多部分上傳"
列出多部分上傳	看"列出多部分上傳"
列出零件	使用所有 Amazon S3 REST API 行為實作。如有變更，恕不另行通知。
上傳部分	看"上傳部分"
上傳部分複製	看"上傳部分複製"

## 完成多部分上傳

CompleteMultipartUpload 作業透過組裝先前上傳的部分來完成物件的分段上傳。



StorageGRID支援按升序排列非連續值 `partNumber` 請求參數與CompleteMultipartUpload一致。此參數可以以任意值開頭。

### 解決衝突

衝突的客戶端請求（例如兩個客戶端寫入同一個金鑰）將根據「最新勝利」的原則解決。「最新勝利」評估的時間取決於StorageGRID系統完成給定請求的時間，而不是 S3 用戶端開始操作的時間。

### 支援的請求標頭

支援以下請求標頭：

- x-amz-checksum-sha256

- `x-amz-storage-class`

這 ``x-amz-storage-class`` StorageGRID 符合的 ILM 規則指定了"雙重提交或平衡攝取選項"。

- STANDARD

(預設) 當 ILM 規則使用雙重提交選項時，或當平衡選項回退到建立臨時副本時，指定雙重提交接收操作。

- REDUCED\_REDUNDANCY

當 ILM 規則使用雙重提交選項時，或當平衡選項回退到建立臨時副本時，指定單一提交攝取操作。



如果您將物件提取到啟用了 S3 物件鎖定的儲存桶中，則 ``REDUCED_REDUNDANCY`` 選項被忽略。如果您將物件提取到舊版相容儲存桶中，``REDUCED_REDUNDANCY`` 選項傳回錯誤。StorageGRID 將始終執行雙重提交攝取以確保滿足合規性要求。



如果分段上傳未在 15 天內完成，則該操作將標記為非活動狀態，並且所有相關資料將從系統中刪除。



這 ``ETag`` 傳回的值不是資料的 MD5 和，而是遵循 Amazon S3 API 實現的 ``ETag`` 多部分物件的值。

## 不支援的請求標頭

不支援以下請求標頭：

- `If-Match`

``If-Match header`` 已被接受，但無法正常使用。

- `If-None-Match`

``If-None-Match header`` 已被接受，但無法正常使用。

- `x-amz-sdk-checksum-algorithm`

- `x-amz-trailer`

## 版本控制

此操作完成分段上傳。如果儲存桶啟用了版本控制，則在分段上傳完成後會建立物件版本。

如果為儲存桶啟用了版本控制，則唯一的 ``versionId`` 針對所儲存物件的版本會自動產生。這 ``versionId`` 也會在回應中返回 ``x-amz-version-id`` 響應頭。

如果版本控制暫停，則物件版本將以空值儲存 ``versionId`` 如果空版本已經存在，它將被覆蓋。



當為儲存桶啟用版本控制時，完成分段上傳總是會建立一個新版本，即使在同一個物件鍵上完成了並發分段上傳。當儲存桶未啟用版本控制時，可以啟動分段上傳，然後讓另一個分段上傳先在同一個物件鍵上啟動並完成。在非版本化儲存桶上，最後完成的分段上傳具有優先權。

## 複製、通知或元資料通知失敗

如果發生分段上傳的儲存桶配置了平台服務，即使相關的複製或通知操作失敗，分段上傳也會成功。

租用戶可以透過更新物件的元資料或標籤來觸發失敗的複製或通知。租戶可以重新提交現有值以避免做出不必要的更改。

請參閱["平台服務故障排除"](#)。

## 建立多部分上傳

CreateMultipartUpload（以前稱為 Initiate Multipart Upload）操作為物件啟動分段上傳，並傳回上傳 ID。

這 `x-amz-storage-class` 支援請求標頭。提交的價值 `x-amz-storage-class` 影響 StorageGRID 在攝取期間如何保護物件數據，而不是影響 StorageGRID 系統中儲存了多少個物件的持久副本（由 ILM 決定）。

如果與已攝取物件相符的 ILM 規則使用嚴格"[攝取選項](#)"，這 `x-amz-storage-class` 標頭無效。

以下值可用於 `x-amz-storage-class`：

- STANDARD(預設)
  - 雙重提交：如果 ILM 規則指定了雙重提交攝取選項，則一旦攝取對象，就會建立該對象的第二個副本並將其分發到不同的儲存節點（雙重提交）。在評估 ILM 時，StorageGRID 會決定這些初始臨時副本是否符合規則中的放置說明。如果沒有，則可能需要在不同位置製作新的物件副本，並且可能需要刪除初始臨時副本。
  - 平衡：如果 ILM 規則指定了平衡選項，且 StorageGRID 無法立即製作規則中指定的所有副本，StorageGRID 會在不同的儲存節點上製作兩個暫存副本。

如果 StorageGRID 可以立即建立 ILM 規則中指定的所有物件副本（同步放置），則 `x-amz-storage-class` 標頭無效。

- REDUCED\_REDUNDANCY
  - 雙重提交：如果 ILM 規則指定了雙重提交選項，StorageGRID 會在物件被攝取時建立一個臨時副本（單次提交）。
  - 平衡：如果 ILM 規則指定了平衡選項，則僅當系統無法立即製作規則中指定的所有副本時，StorageGRID 才會製作單一暫存副本。如果 StorageGRID 可以執行同步放置，則此標頭無效。這 `REDUCED\_REDUNDANCY` 當與物件相符的 ILM 規則建立單一複製副本時，最好使用此選項。在這種情況下使用 `REDUCED\_REDUNDANCY` 消除了每次攝取操作時不必要的額外物件副本的建立和刪除。

使用 `REDUCED\_REDUNDANCY` 在其他情況下不建議選擇此選項。`REDUCED\_REDUNDANCY` 增加了攝取過程中物件資料遺失的風險。例如，如果單一副本最初儲存在儲存節點上，而該儲存節點在 ILM 評估發生之前發生故障，則您可能會遺失資料。



任何時間段內只有一個複製副本會使資料面臨永久遺失的風險。如果某個物件的副本只有一個，則當儲存節點發生故障或發生重大錯誤時，該物件將會遺失。在升級等維護過程中，您也會暫時失去對該物件的存取權限。

指定 `REDUCED\_REDUNDANCY` 僅影響首次攝取物件時所建立的副本數量。它不會影響活動 ILM 策略評估物件時產生的物件副本數量，也不會導致資料在 StorageGRID 系統中以較低的冗餘層級進行儲存。



如果您將物件提取到啟用了 S3 物件鎖定的儲存桶中，則 `REDUCED\_REDUNDANCY` 選項被忽略。如果您將物件提取到舊版相容儲存桶中，`REDUCED\_REDUNDANCY` 選項傳回錯誤。StorageGRID 將始終執行雙重提交攝取以確保滿足合規性要求。

## 支援的請求標頭

支援以下請求標頭：

- Content-Type
- x-amz-checksum-algorithm

目前，只有 `x-amz-checksum-algorithm` 受支持。

- x-amz-meta-，後面跟著包含使用者定義元資料的名稱-值對

為使用者定義的元資料指定名稱-值對時，請使用下列通用格式：

```
x-amz-meta-__name__: `value`
```

如果要使用 使用者定義建立時間 選項作為 ILM 規則的參考時間，則必須使用 `creation-time` 作為記錄物件建立時間的元資料的名稱。例如：

```
x-amz-meta-creation-time: 1443399726
```

價值 `creation-time` 以 1970 年 1 月 1 日以後的秒數計算。



添加 `creation-time` 因為如果您將物件新增至已啟用舊版合規性的儲存桶，則不允許使用者定義的元資料。將返回錯誤。

- S3 物件鎖定請求標頭：
  - x-amz-object-lock-mode
  - x-amz-object-lock-retain-until-date
  - x-amz-object-lock-legal-hold

如果發出的請求沒有這些標頭，則使用儲存桶預設保留設定來計算物件版本的保留截止日期。

["使用 S3 REST API 設定 S3 物件鎖"](#)

- SSE 請求標頭：

- x-amz-server-side-encryption
- x-amz-server-side-encryption-customer-key-MD5
- x-amz-server-side-encryption-customer-key
- x-amz-server-side-encryption-customer-algorithm

[伺服器端加密的請求標頭]



有關StorageGRID如何處理 UTF-8 字元的信息，請參閱["放置對象"](#)。

## 伺服器端加密的請求標頭

您可以使用下列請求標頭透過伺服器端加密來加密多部分物件。SSE 和 SSE-C 選項是互斥的。

- **SSE**：如果您想使用由StorageGRID管理的唯一金鑰加密對象，請在 CreateMultipartUpload 請求中使用下列標頭。不要在任何 UploadPart 請求中指定此標頭。
  - x-amz-server-side-encryption
- **SSE-C**：如果您想使用您提供和管理的唯一密鑰加密對象，請在 CreateMultipartUpload 請求（以及每個後續 UploadPart 請求）中使用所有這三個標頭。
  - x-amz-server-side-encryption-customer-algorithm：指定 AES256。
  - x-amz-server-side-encryption-customer-key：為新物件指定加密金鑰。
  - x-amz-server-side-encryption-customer-key-MD5：指定新物件的加密金鑰的 MD5 摘要。



您提供的加密金鑰永遠不會被儲存。如果遺失了加密金鑰，您就會遺失對應的物件。在使用客戶提供的金鑰保護物件資料之前，請先查看以下注意事項["使用伺服器端加密"](#)。

## 不支援的請求標頭

不支援以下請求標頭：

- x-amz-website-redirect-location

這 x-amz-website-redirect-location 標題返回 `XNotImplemented`。

## 版本控制

分段上傳包括啟動上傳、列出上傳、上傳部分、組裝上傳部分和完成上傳的單獨操作。執行 CompleteMultipartUpload 作業時會建立物件（如果適用，也會進行版本控制）。

## 列出多部分上傳

ListMultipartUploads 操作列出儲存桶正在進行的分段上傳。

支援以下請求參數：

- encoding-type
- key-marker
- max-uploads
- prefix
- upload-id-marker
- Host
- Date
- Authorization

## 版本控制

分段上傳包括啟動上傳、列出上傳、上傳部分、組裝上傳部分和完成上傳的單獨操作。執行 CompleteMultipartUpload 作業時會建立物件（如果適用，也會進行版本控制）。

## 上傳部分

UploadPart 作業用於在物件的分段上傳中上傳某個部分。

### 支援的請求標頭

支援以下請求標頭：

- x-amz-checksum-sha256
- Content-Length
- Content-MD5

### 伺服器端加密的請求標頭

如果您為 CreateMultipartUpload 要求指定了 SSE-C 加密，則也必須在每個 UploadPart 請求中包含下列請求標頭：

- x-amz-server-side-encryption-customer-algorithm：指定 AES256。
- x-amz-server-side-encryption-customer-key：指定您在 CreateMultipartUpload 請求中提供的相同加密金鑰。
- x-amz-server-side-encryption-customer-key-MD5：指定您在 CreateMultipartUpload 請求中提供的相同 MD5 摘要。



您提供的加密金鑰永遠不會被儲存。如果遺失了加密金鑰，您就會遺失對應的物件。在使用客戶提供的金鑰保護物件資料之前，請先查看["使用伺服器端加密"](#)。

如果您在 CreateMultipartUpload 請求期間指定了 SHA-256 校驗和，則還必須在每個 UploadPart 請求中包含下列請求標頭：

- x-amz-checksum-sha256：指定此部分的 SHA-256 校驗和。

## 不支援的請求標頭

不支援以下請求標頭：

- x-amz-sdk-checksum-algorithm
- x-amz-trailer

## 版本控制

分段上傳包括啟動上傳、列出上傳、上傳部分、組裝上傳部分和完成上傳的單獨操作。執行 CompleteMultipartUpload 作業時會建立物件（如果適用，也會進行版本控制）。

## 上傳部分複製

UploadPartCopy 操作透過從現有物件作為資料來源複製資料來上傳物件的一部分。

UploadPartCopy 操作是透過所有 Amazon S3 REST API 行為實現的。如有變更，恕不另行通知。

此請求讀取並寫入指定的物件數據 `x-amz-copy-source-range` 在 StorageGRID 系統內。

支援以下請求標頭：

- x-amz-copy-source-if-match
- x-amz-copy-source-if-none-match
- x-amz-copy-source-if-unmodified-since
- x-amz-copy-source-if-modified-since

## 伺服器端加密的請求標頭

如果您為 CreateMultipartUpload 要求指定了 SSE-C 加密，則也必須在每個 UploadPartCopy 請求中包含以下請求標頭：

- x-amz-server-side-encryption-customer-algorithm：指定 AES256。
- x-amz-server-side-encryption-customer-key：指定您在 CreateMultipartUpload 請求中提供的相同加密金鑰。
- x-amz-server-side-encryption-customer-key-MD5：指定您在 CreateMultipartUpload 請求中提供的相同 MD5 摘要。

如果來源物件使用客戶提供的金鑰（SSE-C）加密，則必須在 UploadPartCopy 請求中包含以下三個標頭，以便可以解密然後複製物件：

- x-amz-copy-source-server-side-encryption-customer-algorithm：指定 AES256。
- x-amz-copy-source-server-side-encryption-customer-key：指定您在建立來源物件時提供的加密金鑰。
- x-amz-copy-source-server-side-encryption-customer-key-MD5：指定您在建立來源物件時提供的 MD5 摘要。



您提供的加密金鑰永遠不會被儲存。如果遺失了加密金鑰，您就會遺失對應的物件。在使用客戶提供的金鑰保護物件資料之前，請先查看["使用伺服器端加密"](#)。

## 版本控制

分段上傳包括啟動上傳、列出上傳、上傳部分、組裝上傳部分和完成上傳的單獨操作。執行 CompleteMultipartUpload 作業時會建立物件（如果適用，也會進行版本控制）。

## 錯誤回應

StorageGRID系統支援所有適用的標準 S3 REST API 錯誤回應。此外，StorageGRID實作還新增了幾個自訂回應。

### 支援的 S3 API 錯誤代碼

Name	HTTP 狀態
拒絕訪問	403 禁止
壞文摘	400 錯誤請求
BucketAlreadyExists	409衝突
BucketNotEmpty	409衝突
不完整的身體	400 錯誤請求
內部錯誤	500 內部伺服器錯誤
無效存取金鑰	403 禁止
無效參數	400 錯誤請求
無效的儲存桶名稱	400 錯誤請求
無效的BucketState	409衝突
無效摘要	400 錯誤請求
無效加密演算法錯誤	400 錯誤請求
無效部分	400 錯誤請求
無效零件訂單	400 錯誤請求

Name	HTTP 狀態
無效範圍	416 請求範圍不滿足
無效請求	400 錯誤請求
無效的儲存類	400 錯誤請求
無效標籤	400 錯誤請求
無效URI	400 錯誤請求
鍵太長	400 錯誤請求
格式錯誤的XML	400 錯誤請求
元數據太大	400 錯誤請求
方法不允許	405 方法不允許
缺失內容長度	411 長度要求
MissingRequestBody錯誤	400 錯誤請求
缺少安全標頭	400 錯誤請求
沒有這樣的桶	404 未找到
NoSuchKey	404 未找到
NoSuchUpload	404 未找到
未實施	501 未實現
NoSuchBucketPolicy	404 未找到
ObjectLockConfigurationNotFoundError	404 未找到
前提條件不成立	412 先決條件失敗
請求時間偏差過大	403 禁止
暫停服務	503 服務不可用

Name	HTTP 狀態
簽名不匹配	403 禁止
桶太多	400 錯誤請求
必須指定使用者金鑰	400 錯誤請求

## StorageGRID 自訂錯誤代碼

Name	描述	HTTP 狀態
XBucketLifecycleNotAllowed	舊版相容儲存桶中不允許配置儲存桶生命週期	400 錯誤請求
XBucketPolicyParseException	無法解析收到的儲存桶策略 JSON。	400 錯誤請求
XComplianceConflict	由於舊版合規性設置，操作被拒絕。	403 禁止
XComplianceReducedRedundancyForbidden	舊版相容存儲桶中不允許減少冗餘	400 錯誤請求
XMaxBucketPolicyLengthExceeded	您的策略超出了允許的最大儲存桶策略長度。	400 錯誤請求
XMissingInternalRequestHeader	缺少內部請求的標頭。	400 錯誤請求
XNoSuchBucketCompliance	指定的儲存桶未啟用舊版合規性。	404 未找到
X不可接受	此請求包含一個或多個無法滿足的接受標頭。	406 不可接受
XNotImplemented	您提供的請求暗示了尚未實現的功能。	501 未實現

## 版權資訊

Copyright © 2026 NetApp, Inc. 版權所有。台灣印製。非經版權所有人事先書面同意，不得將本受版權保護文件的任何部分以任何形式或任何方法（圖形、電子或機械）重製，包括影印、錄影、錄音或儲存至電子檢索系統中。

由 NetApp 版權資料衍伸之軟體必須遵守下列授權和免責聲明：

此軟體以 NETAPP「原樣」提供，不含任何明示或暗示的擔保，包括但不限於有關適售性或特定目的適用性之擔保，特此聲明。於任何情況下，就任何已造成或基於任何理論上責任之直接性、間接性、附隨性、特殊性、懲罰性或衍生性損害（包括但不限於替代商品或服務之採購；使用、資料或利潤上的損失；或企業營運中斷），無論是在使用此軟體時以任何方式所產生的契約、嚴格責任或侵權行為（包括疏忽或其他）等方面，NetApp 概不負責，即使已被告知有前述損害存在之可能性亦然。

NetApp 保留隨時變更本文所述之任何產品的權利，恕不另行通知。NetApp 不承擔因使用本文所述之產品而產生的責任或義務，除非明確經過 NetApp 書面同意。使用或購買此產品並不會在依據任何專利權、商標權或任何其他 NetApp 智慧財產權的情況下轉讓授權。

本手冊所述之產品受到一項（含）以上的美國專利、國外專利或申請中專利所保障。

有限權利說明：政府機關的使用、複製或公開揭露須受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中的「技術資料權利 - 非商業項目」條款 (b)(3) 小段所述之限制。

此處所含屬於商業產品和 / 或商業服務（如 FAR 2.101 所定義）的資料均為 NetApp, Inc. 所有。根據本協議提供的所有 NetApp 技術資料和電腦軟體皆屬於商業性質，並且完全由私人出資開發。美國政府對於該資料具有非專屬、非轉讓、非轉授權、全球性、有限且不可撤銷的使用權限，僅限於美國政府為傳輸此資料所訂合約所允許之範圍，並基於履行該合約之目的方可使用。除非本文另有規定，否則未經 NetApp Inc. 事前書面許可，不得逕行使用、揭露、重製、修改、履行或展示該資料。美國政府授予國防部之許可權利，僅適用於 DFARS 條款 252.227-7015(b)（2014 年 2 月）所述權利。

## 商標資訊

NETAPP、NETAPP 標誌及 <http://www.netapp.com/TM> 所列之標章均為 NetApp, Inc. 的商標。文中所涉及的所有其他公司或產品名稱，均為其各自所有者的商標，不得侵犯。