Amazon S3 REST API 지원

StorageGRID

NetApp
November 04, 2025

This PDF was generated from https://docs.netapp.com/ko-kr/storagegrid-117/s3/s3-rest-api-supported-operations-and-limitations.html on November 04, 2025. Always check docs.netapp.com for the latest.

목차

Amazon S3 REST API 지원 . . . 1
S3 REST API 구현 세부 정보 . . . 1
날짜 처리 . . . 1
공통 요청 헤더 . . . 1
공통 응답 헤더 . . . 1
요청을 인증합니다 . . . 2
HTTP 인증 헤더를 사용합니다 . . . 2
쿼리 매개 변수를 사용합니다 . . . 2
서비스에 대한 작업 . . . 2
버킷 작업 . . . 3
버킷에 대한 사용자 지정 작업 . . . 10
객체에 대한 작업 . . . 11
S3 Select를 사용합니다 . . . 15
서버측 암호화를 사용합니다 . . . 17
객체 가져오기 . . . 19
헤드 개체 . . . 21
사후 개체 복원 . . . 24
개체 를 넣습니다 . . . 26
개체 - 복사 를 선택합니다 . . . 30
SelectObjectContent 를 선택합니다 . . . 34
멀티파트 업로드 작업 . . . 38
다중 파트 업로드 나열 . . . 39
멀티파트 업로드를 시작합니다 . . . 40
부품 업로드 . . . 42
업로드 부품 - 복사 . . . 43
멀티파트 업로드를 완료합니다 . . . 44
오류 응답 . . . 46
지원되는 S3 API 오류 코드입니다 . . . 46
StorageGRID 사용자 지정 오류 코드 . . . 48

Amazon S3 REST API 지원

S3 REST API 구현 세부 정보

StorageGRID 시스템은 대부분의 작업을 지원하고 몇 가지 제한 사항이 있는 간단한 스토리지 서비스 API(API 버전 2006-03-01)를 구현합니다. S3 REST API 클라이언트 애플리케이션을 통합할 때 구현 세부 정보를 이해해야 합니다.

StorageGRID 시스템은 가상 호스팅 방식의 요청과 경로 스타일 요청을 모두 지원합니다.

날짜 처리

S3 REST API의 StorageGRID 구현은 유효한 HTTP 날짜 형식만 지원합니다.

StorageGRID 시스템은 날짜 값을 허용하는 모든 헤더에 대해 유효한 HTTP 날짜 형식만 지원합니다. 날짜의 시간 부분은 그리니치 표준시(GMT) 형식 또는 표준 시간대 오프셋 없이 UTC(국제 표준시) 형식으로 지정할 수 있습니다(+0000을 지정해야 함). 을 포함하는 경우 `x-amz-date` 헤더 요청의 날짜 요청 헤더에 지정된 모든 값을 재정의합니다. AWS 서명 버전 4를 사용하는 경우 `x-amz-date` 날짜 헤더가 지원되지 않으므로 서명된 요청에 헤더가 있어야 합니다.

공통 요청 헤더

StorageGRID 시스템은 에 정의된 공통 요청 헤더를 지원합니다 "Amazon Simple Storage Service API 참조: 일반 요청 헤더"한 가지 예외가 있습니다.

요청 헤더	구축
권한 부여	AWS Signature 버전 2에 대한 전체 지원 다음 경우를 제외하고 AWS Signature 버전 4 지원: • 요청 본문에 대한 SHA256 값이 계산되지 않습니다. 사용자가 제출한 값은 마치 값이 있는 것처럼 유효성 검사 없이 승인됩니다 `UNSIGNED-PAYLOAD` 에 대한 정보가 제공되었습니다 `x-amz-content-sha256` 머리글.
X-amz-security-token	구현되지 않았습니다. 반환 `XNotImplemented`.

공통 응답 헤더

StorageGRID 시스템은 한 가지 예외를 제외하고 _Simple Storage Service API Reference_에 의해 정의된 모든 공통 응답 헤더를 지원합니다.

응답 헤더	구축
X-amz-id-2	사용 안 합니다

요청을 인증합니다

StorageGRID 시스템은 S3 API를 사용하여 오브젝트에 대한 인증된 액세스와 익명 액세스를 모두 지원합니다.

S3 API는 S3 API 요청을 인증하는 데 서명 버전 2 및 서명 버전 4를 지원합니다.

인증된 요청은 액세스 키 ID 및 비밀 액세스 키를 사용하여 서명해야 합니다.

StorageGRID 시스템은 HTTP라는 두 가지 인증 방법을 지원합니다 `Authorization` 머리글 및 쿼리 매개 변수 사용

HTTP 인증 헤더를 사용합니다

HTTP `Authorization` 헤더는 버킷 정책에서 허용하는 익명 요청을 제외한 모든 S3 API 작업에서 사용됩니다. 를 클릭합니다 `Authorization` Header 요청을 인증하는 데 필요한 모든 서명 정보를 포함합니다.

쿼리 매개 변수를 사용합니다

쿼리 매개 변수를 사용하여 URL에 인증 정보를 추가할 수 있습니다. 이를 URL 사전 서명 이라고 하며, 이 URL을 사용하여 특정 리소스에 대한 임시 액세스 권한을 부여할 수 있습니다. 미리 지정된 URL을 가진 사용자는 리소스에 액세스하기 위해 비밀 액세스 키를 알 필요가 없으며, 이를 통해 타사에 리소스에 대한 제한된 액세스를 제공할 수 있습니다.

서비스에 대한 작업

StorageGRID 시스템은 서비스에 대해 다음 작업을 지원합니다.

작동	구축
서비스 받기 `(ListBuckets)`	모든 Amazon S3 REST API 동작으로 구현됩니다. 예고 없이 변경될 수 있습니다.
스토리지 사용량을 가져옵니다	Get Storage Usage 요청은 계정이 사용하는 총 스토리지 양과 계정과 연결된 각 버킷에 대해 알려줍니다. 이 작업은 /path 및 사용자 지정 쿼리 매개 변수가 있는 서비스에 대한 작업입니다 (`?x-ntap-sg-usage`)가 추가되었습니다.
옵션 /	클라이언트 응용 프로그램을 실행할 수 있습니다 `OPTIONS /` 스토리지 노드의 사용 가능 여부를 결정하기 위해 S3 인증 자격 증명을 제공하지 않고 스토리지 노드의 S3 포트에 대한 요청입니다. 이 요청을 사용하여 모니터링을 수행하거나, 외부 로드 밸런서가 스토리지 노드가 다운된 시점을 식별하도록 할 수 있습니다.

관련 정보

"스토리지 사용량을 가져옵니다"

버킷 작업

StorageGRID 시스템은 각 S3 테넌트 계정에 대해 최대 1,000개의 버킷을 지원합니다.

버킷 이름 제한은 AWS US 표준 지역 제한을 따르지만, S3 가상 호스팅 스타일 요청을 지원하기 위해 DNS 명명 규칙으로 제한해야 합니다.

자세한 내용은 다음을 참조하십시오.

- "AWS(Amazon Web Services) 문서: 버킷 제한 및 제한 사항"
- "S3 끝점 도메인 이름을 구성합니다"

버킷 가져오기(개체 나열) 및 버킷 버전 가져오기 작업은 StorageGRID 정합성 보장 제어를 지원합니다.

개별 버킷에 대해 마지막 액세스 시간에 대한 업데이트가 설정되었는지 여부를 확인할 수 있습니다.

다음 표에서는 StorageGRID에서 S3 REST API 버킷 작업을 구축하는 방법을 설명합니다. 이러한 작업을 수행하려면 계정에 필요한 액세스 자격 증명을 제공해야 합니다.

작동	구축
버킷 삭제	이 작업은 버킷을 삭제합니다.
버킷 CORS를 삭제합니다	이 작업은 버킷에 대한 CORS 구성을 삭제합니다.
Bucket 암호화를 삭제합니다	이 작업은 버킷에서 기본 암호화를 삭제합니다. 암호화된 기존 개체는 암호화된 상태로 유지되지만 버킷에 추가된 새 개체는 암호화되지 않습니다.
버킷 수명 주기를 삭제합니다	이 작업은 버킷에서 라이프사이클 구성을 삭제합니다. 을 참조하십시오 "S3 라이프사이클 구성을 생성합니다".
버킷 정책을 삭제합니다	이 작업은 버킷에 연결된 정책을 삭제합니다.
버킷 복제를 삭제합니다	이 작업은 버킷에 연결된 복제 구성을 삭제합니다.
버킷 태그 지정을 삭제합니다	이 작업은 를 사용합니다 `tagging` 버킷에서 모든 태그를 제거하는 하위 리소스입니다.

작동	구축
버킷 가져오기 (목록 개체) (ListObjectsV2)	이 작업은 버킷에 있는 오브젝트의 일부 또는 전체(최대 1,000개)를 반환합니다. 오브젝트를 에 인제스트한 경우에도 오브젝트에 대한 스토리지 클래스는 두 값 중 하나를 가질 수 있습니다 `REDUCED_REDUNDANCY` 스토리지 클래스 옵션: • `STANDARD`는 객체가 스토리지 노드로 구성된 스토리지 풀에 저장되었음을 나타냅니다. • `GLACIER`가 표시됩니다. 이는 해당 객체가 Cloud Storage Pool에 지정된 외부 버킷으로 이동되었음을 나타냅니다. 버킷에 동일한 접두사가 있는 삭제된 키의 많은 수가 포함된 경우 응답에 몇 가지 항목이 포함될 수 있습니다 `CommonPrefixes` 여기에는 키가 포함되어 있지 않습니다.
버킷 객체 버전을 가져옵니다 (ListObjectVersions)	버킷에 대한 읽기 액세스에서 이 작업은 를 통해 수행됩니다 `versions` 하위 리소스는 버킷에 있는 모든 버전의 오브젝트의 메타데이터를 나열합니다.
버킷 ACL 가져오기	이 작업은 양수 응답 및 버킷 소유자의 ID, DisplayName 및 권한을 반환하며, 이는 소유자가 버킷에 대한 모든 액세스 권한을 가지고 있음을 나타냅니다.
버킷 CORS를 가져옵니다	이 작업은 를 반환합니다 `cors` 버킷에 대한 구성.
버킷 암호화 가져오기	이 작업은 버킷의 기본 암호화 구성을 반환합니다.
버킷 수명 주기 가져오기 (GetBuckLifecycleConfiguration)	이 작업은 버킷의 수명 주기 구성을 반환합니다. 을 참조하십시오 "S3 라이프사이클 구성을 생성합니다".
버킷 위치를 가져옵니다	이 작업은 를 사용하여 설정된 영역을 반환합니다 `LocationConstraint` PUT 버킷 요청에 있는 요소입니다. 버킷 영역이 인 경우 `us-east-1`영역에 대해 빈 문자열이 반환됩니다.
버킷 알림을 받습니다 (GetBuckNotificationConfiguration)	이 작업은 버킷에 연결된 알림 구성을 반환합니다.
버킷 정책 가져오기	이 작업은 버킷에 연결된 정책을 반환합니다.
버킷 복제를 가져옵니다	이 작업은 버킷에 연결된 복제 구성을 반환합니다.
버킷 태그 지정을 가져옵니다	이 작업은 를 사용합니다 `tagging` 버킷에 대한 모든 태그를 반환하는 하위 리소스입니다.

작동	구축
버킷 버전 관리 가져오기	이 구현에서는 을 사용합니다 `versioning` 버킷의 버전 관리 상태를 반환하는 하위 리소스입니다. • *blank*: 버전 관리가 활성화된 적이 없습니다(버킷을 ""버전 없음""). • 사용: 버전 관리가 활성화됩니다 • 일시 중단됨: 버전 관리가 이전에 활성화되었으며 일시 중단되었습니다
개체 잠금 구성을 가져옵니다	이 작업은 버킷 기본 보존 모드 및 기본 보존 기간(구성된 경우)을 반환합니다. 을 참조하십시오 "S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다".
헤드 버킷	이 작업은 버킷이 있는지 그리고 버킷에 액세스할 권한이 있는지 여부를 결정합니다. 이 작업은 다음을 반환합니다. • `x-ntap-sg-bucket-id`UUID 형식의 버킷 UUID입니다. • `x-ntap-sg-trace-id`: 연결된 요청의 고유한 추적 ID입니다.

작동	구축
버킷 을 놓습니다	이 작업은 새 버킷을 생성합니다. 버킷을 만들면 버켓 소유자가 됩니다. • 버킷 이름은 다음 규칙을 준수해야 합니다. ◦ 각 StorageGRID 시스템에서 고유해야 합니다(테넌트 계정에서만 고유한 것은 아님). ◦ DNS를 준수해야 합니다. ◦ 3자 이상 63자 이하여야 합니다. ◦ 인접한 레이블이 마침표로 구분된 하나 이상의 레이블일 수 있습니다. 각 레이블은 소문자 또는 숫자로 시작하고 끝나야 하며 소문자, 숫자 및 하이픈만 사용할 수 있습니다. ◦ 텍스트 형식의 IP 주소처럼 보이지 않아야 합니다. ◦ 가상 호스팅 스타일 요청에서 기간을 사용하지 않아야 합니다. 마침표는 서버 와일드카드 인증서 확인에 문제를 일으킬 수 있습니다. • 기본적으로 버킷은 에서 생성됩니다 `us-east-1` 지역. 그러나 을 사용할 수 있습니다 `LocationConstraint` 다른 영역을 지정할 요청 본문의 요청 요소입니다. 를 사용할 때 `LocationConstraint` 요소, 그리드 관리자 또는 그리드 관리 API를 사용하여 정의된 영역의 정확한 이름을 지정해야 합니다. 사용할 지역 이름을 모르는 경우 시스템 관리자에게 문의하십시오. • 참고 *: PUT 버킷 요청이 StorageGRID에 정의되지 않은 지역을 사용하는 경우 오류가 발생합니다. • 을 포함할 수 있습니다 `x-amz-bucket-object-lock-enabled` S3 오브젝트 잠금이 활성화된 버킷을 생성하도록 헤더를 요청합니다. 을 참조하십시오 "S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다". 버킷을 생성할 때 S3 오브젝트 잠금을 활성화해야 합니다. 버킷을 생성한 후에는 S3 오브젝트 잠금을 추가하거나 비활성화할 수 없습니다. S3 오브젝트 잠금에는 버킷 버전 관리가 필요하며, 이 버전은 버킷을 생성할 때 자동으로 활성화됩니다.
버킷 CORS를 넣습니다	이 작업은 버킷이 오리진 간 요청을 처리할 수 있도록 버킷에 대한 CORS 구성을 설정합니다. CORS(Cross-origin Resource Sharing)는 한 도메인의 클라이언트 웹 애플리케이션이 다른 도메인의 리소스에 액세스할 수 있도록 하는 보안 메커니즘입니다. 예를 들어, 이라는 S3 버킷을 사용한다고 가정합니다 `images` 그래픽을 저장합니다. 에 대한 CORS 구성을 설정합니다 `images` 버킷을 사용하면 버켓의 이미지를 웹 사이트에 표시할 수 있습니다 `http://www.example.com`.
Bucket 암호화를 적용합니다	이 작업은 기존 버킷의 기본 암호화 상태를 설정합니다. 버킷 수준 암호화가 활성화된 경우 버킷에 추가된 모든 새 오브젝트는 암호화됩니다. StorageGRID는 StorageGRID 관리 키로 서버 측 암호화를 지원합니다. 서버 측 암호화 구성 규칙을 지정할 때 를 설정합니다 `SSEAlgorithm` 매개 변수 대상 `AES256`를 사용하지 마십시오 `KMSMasterKeyID` 매개 변수. 객체 업로드 요청이 이미 암호화를 지정한 경우(즉, 요청에 가 포함된 경우) 버킷 기본 암호화 구성은 무시됩니다 `x-amz-server-side-encryption-*` 요청 헤더 참조).

작동	구축
버킷 수명 주기를 놓습니다 (PutBucketLifecycleConfiguration)	이 작업은 버킷에 대한 새 수명 주기 구성을 생성하거나 기존 수명 주기 구성을 대체합니다. StorageGRID는 수명 주기 구성에서 최대 1,000개의 수명 주기 규칙을 지원합니다. 각 규칙에는 다음 XML 요소가 포함될 수 있습니다. • 만료(일, 날짜) • NoncurrentVersionExpiration(NoncurrentDays) • 필터(접두사, 태그) • 상태 • ID입니다 StorageGRID는 다음 작업을 지원하지 않습니다. • AbortIncompleteMultipartUpload를 중단합니다 • ExpiredObjectDeleteMarker 를 참조하십시오 • 전환 을 참조하십시오 "S3 라이프사이클 구성을 생성합니다". 버킷 수명 주기의 만료 작업이 ILM 배치 지침과 상호 작용하는 방법을 이해하려면 을 참조하십시오 "ILM이 개체 수명 전반에 걸쳐 작동하는 방식". • 참고 *: 버킷 수명 주기 구성은 S3 오브젝트 잠금이 활성화된 버킷과 함께 사용할 수 있지만 레거시 준수 버킷에서는 버킷 수명 주기 구성이 지원되지 않습니다.

작동	구축

버킷 통지를 보냅니다
 	
(PutBuckNotificationConfiguration)
	이 작업은 요청 본문에 포함된 알림 구성 XML을 사용하여 버킷에 대한 알림을 구성합니다. 다음과 같은 구현 세부 사항에 유의해야 합니다.
	
	• StorageGRID는 SNS(Simple Notification Service) 항목을 대상으로 지원합니다. SQS(Simple Queue Service) 또는 Amazon Lambda 엔드포인트는 지원되지 않습니다.
	• 알림 대상은 StorageGRID 엔드포인트의 URN으로 지정해야 합니다. 테넌트 관리자 또는 테넌트 관리 API를 사용하여 엔드포인트를 생성할 수 있습니다.
	
	알림 설정을 성공적으로 하려면 끝점이 있어야 합니다. 끝점이 없는 경우, 를 클릭합니다 `400 Bad Request` 코드와 함께 오류가 반환됩니다 `InvalidArgument`.
	
	• 다음 이벤트 유형에 대한 알림을 구성할 수 없습니다. 이러한 이벤트 유형은 * 지원되지 않습니다 *.
	◦ `s3:ReducedRedundancyLostObject`
	◦ `s3:ObjectRestore:Completed`
	• StorageGRID에서 보낸 이벤트 알림은 다음 목록에 표시된 것처럼 일부 키를 포함하지 않고 다른 키에 대해 특정 값을 사용한다는 점을 제외하고 표준 JSON 형식을 사용합니다.
	◦ * eventSource * 를 선택합니다
	
	`sgws:s3`
	
	◦ * awsRegion *
	
	포함되지 않음
	
	◦ x-amz-id-2 *
	
	포함되지 않음
	
	◦ * 표시 *
	
	`urn:sgws:s3:::bucket_name`

버킷 정책을 적용합니다	이 작업은 버킷에 연결된 정책을 설정합니다.

작동	구축
버킷 복제를 배치합니다	이 작업은 를 구성합니다 "StorageGRID CloudMirror 복제" 요청 본문에 제공된 복제 구성 XML을 사용하는 버킷의 경우 CloudMirror 복제의 경우 다음과 같은 구축 세부 정보를 알고 있어야 합니다. • StorageGRID는 복제 구성의 V1만 지원합니다. 즉, StorageGRID는 의 사용을 지원하지 않습니다 `Filter` 규칙에 대한 요소로, 개체 버전 삭제에 대한 V1 규칙을 따릅니다. 자세한 내용은 를 참조하십시오 "복제 구성에 대한 Amazon S3 문서". • 버킷 복제는 버전 관리되거나 버전이 지정되지 않은 버킷에서 구성할 수 있습니다. • 복제 구성 XML의 각 규칙에서 다른 대상 버킷을 지정할 수 있습니다. 소스 버킷은 둘 이상의 대상 버킷에 복제할 수 있습니다. • 대상 버킷은 테넌트 관리자 또는 테넌트 관리 API에 지정된 StorageGRID 엔드포인트의 URN으로 지정해야 합니다. 을 참조하십시오 "CloudMirror 복제를 구성합니다". 복제 구성이 성공하려면 엔드포인트가 있어야 합니다. 엔드포인트가 없으면 요청이 로 실패합니다 `400 Bad Request`. 오류 메시지는 다음과 같습니다. `Unable to save the replication policy. The specified endpoint URN does not exist: URN`. • 을 지정할 필요가 없습니다 `Role` 구성 XML에서. 이 값은 StorageGRID에서 사용되지 않으며 제출될 경우 무시됩니다. • 구성 XML에서 스토리지 클래스를 생략하면 StorageGRID에서 를 사용합니다 `STANDARD` 기본적으로 스토리지 클래스입니다. • 소스 버킷에서 객체를 삭제하거나 소스 버킷 자체를 삭제하는 경우 지역 간 복제 동작은 다음과 같습니다. ◦ 복제되기 전에 오브젝트 또는 버킷을 삭제하면 객체/버킷이 복제되지 않으므로 사용자에게 통지되지 않습니다. ◦ 복제된 후 오브젝트 또는 버킷을 삭제하면 StorageGRID는 지역 간 복제 V1에 대한 표준 Amazon S3 삭제 동작을 따릅니다.
Bucket 태그 달기	이 작업은 를 사용합니다 `tagging` 하위 리소스로서 버킷에 대한 태그 집합을 추가하거나 업데이트합니다. 버킷 태그를 추가할 때 다음과 같은 제한 사항을 숙지하십시오. • StorageGRID 및 Amazon S3 모두 각 버킷당 최대 50개의 태그를 지원합니다. • 버킷과 연결된 태그에는 고유한 태그 키가 있어야 합니다. 태그 키의 길이는 최대 128자의 유니코드 문자일 수 있습니다. • 태그 값의 길이는 최대 256자의 유니코드 문자일 수 있습니다. • 키와 값은 대/소문자를 구분합니다.

작동	구축
버킷 버전 관리	이 구현에서는 을 사용합니다 `versioning` 기존 버킷의 버전 관리 상태를 설정하는 하위 리소스입니다. 다음 값 중 하나를 사용하여 버전 관리 상태를 설정할 수 있습니다. • Enabled(사용): 버킷의 오브젝트에 대한 버전 관리를 활성화합니다. 버킷에 추가된 모든 오브젝트는 고유한 버전 ID를 받습니다. • Suspended(일시 중지됨): 버킷의 오브젝트에 대한 버전 관리를 비활성화합니다. 버킷에 추가된 모든 오브젝트는 버전 ID를 수신합니다 `null`.
개체 잠금 구성을 배치합니다	이 작업은 버킷 기본 보존 모드 및 기본 보존 기간을 구성하거나 제거합니다. 기본 보존 기간이 수정되면 기존 개체 버전의 보존 기한은 그대로 유지되며 새 기본 보존 기간을 사용하여 다시 계산되지 않습니다. 을 참조하십시오 "S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다" 을 참조하십시오.

관련 정보

"일관성 제어"

"버킷 최종 액세스 시간 가져오기"

"버킷 및 그룹 액세스 정책을 사용합니다"

"S3 작업이 감사 로그에서 추적되었습니다"

버킷에 대한 사용자 지정 작업

StorageGRID 시스템은 S3 REST API에 추가되고 시스템에 고유한 맞춤형 버킷 작업을 지원합니다.

다음 표에는 StorageGRID에서 지원하는 사용자 지정 버킷 작업이 나열되어 있습니다.

작동	설명	를 참조하십시오
버킷 일관성 확보	특정 버킷에 적용되는 정합성 보장 레벨을 반환합니다.	"버킷 일관성 확보"
버킷 일관성을 유지합니다	특정 버킷에 적용되는 정합성 수준을 설정합니다.	"버킷 일관성을 유지합니다"
버킷 최종 액세스 시간 가져오기	특정 버킷에 대해 마지막 액세스 시간 업데이트를 사용할 수 있는지 여부를 반환합니다.	"버킷 최종 액세스 시간 가져오기"

작동	설명	를 참조하십시오
버킷 최종 접근 시간	특정 버킷에 대한 마지막 액세스 시간 업데이트를 활성화하거나 비활성화할 수 있습니다.	"버킷 최종 접근 시간"
버킷 메타데이터 알림 구성을 삭제합니다	특정 버킷과 연결된 메타데이터 알림 구성 XML을 삭제합니다.	"버킷 메타데이터 알림 구성을 삭제합니다"
Bucket 메타데이터 알림 구성 가져오기	특정 버킷과 연결된 메타데이터 알림 구성 XML을 반환합니다.	"Bucket 메타데이터 알림 구성 가져오기"
Put Bucket 메타데이터 알림 구성	버킷에 대한 메타데이터 알림 서비스를 구성합니다.	"Put Bucket 메타데이터 알림 구성"
준수 설정이 적용된 버킷을 배치합니다	더 이상 사용되지 않으며 지원되지 않음: Compliance를 사용하는 새 버킷을 더 이상 생성할 수 없습니다.	"사용되지 않음: 준수 설정이 포함된 Bucket을 넣습니다"
버킷 규정 준수	더 이상 사용되지 않지만 지원됨: 기존 레거시 준수 버킷에 대해 현재 적용되는 규정 준수 설정을 반환합니다.	"사용되지 않음: 버킷 준수 가져오기"
버킷 규정 준수	사용되지 않지만 지원됨: 기존 레거시 준수 버킷의 준수 설정을 수정할 수 있습니다.	"사용되지 않음: 버킷 준수"

관련 정보

"감사 로그에서 S3 작업을 추적했습니다"

객체에 대한 작업

이 섹션에서는 StorageGRID 시스템이 객체에 대해 S3 REST API 작업을 구축하는 방법에 대해 설명합니다.

다음 조건은 모든 개체 작업에 적용됩니다.

- StorageGRID "일관성 제어" 는 다음과 같은 경우를 제외하고 모든 개체 작업에서 지원됩니다.
  - 객체 ACL을 가져옵니다
  - `OPTIONS /`
  - 개체를 법적 증거 자료 보관
  - 개체 보존
  - 개체 내용 을 선택합니다

- 동일한 키에 쓰는 두 클라이언트 등의 충돌하는 클라이언트 요청은 "최신 성공" 기준으로 해결됩니다. "Latest-WINS" 평가 시기는 S3 클라이언트가 작업을 시작할 때가 아니라 StorageGRID 시스템이 지정된 요청을 완료하는 시점을 기준으로 합니다.
- StorageGRID 버킷의 모든 오브젝트는 익명 사용자 또는 다른 계정에서 만든 오브젝트를 포함하여 버킷 소유자가 소유합니다.
- Swift를 통해 StorageGRID 시스템으로 수집된 데이터 오브젝트는 S3를 통해 액세스할 수 없습니다.

다음 표에서는 StorageGRID에서 S3 REST API 오브젝트 작업을 구현하는 방법을 설명합니다.

작동	구축
개체 삭제	MFA(Multi-Factor Authentication) 및 응답 헤더입니다 `x-amz-mfa` 지원되지 않습니다. 오브젝트 삭제 요청을 처리할 때 StorageGRID는 저장된 모든 위치에서 오브젝트의 모든 복사본을 즉시 제거하려고 시도합니다. 성공하면 StorageGRID는 즉시 클라이언트에 응답을 반환합니다. 위치를 일시적으로 사용할 수 없기 때문에 30초 이내에 모든 복사본을 제거할 수 없는 경우 StorageGRID는 제거할 복사본을 대기시킨 다음 클라이언트에 성공 여부를 표시합니다. 버전 관리 특정 버전을 제거하려면 요청자가 버킷 소유자여야 하며 를 사용해야 합니다 `versionId` 하위 리소스. 이 하위 리소스를 사용하면 버전이 영구적으로 삭제됩니다. 를 누릅니다 `versionId` 삭제 마커인 응답 헤더에 해당합니다 `x-amz-delete-marker` 가 로 설정된 상태로 반환됩니다 `true`. • 를 사용하지 않고 개체를 삭제한 경우 `versionId` 버전 지원 버킷의 하위 리소스에서는 삭제 마커가 생성됩니다. 를 클릭합니다 `versionId` 삭제 마커는 를 사용하여 반환됩니다 `x-amz-version-id` 응답 헤더 및 `x-amz-delete-marker` 로 설정된 응답 헤더가 반환됩니다 `true`. • 를 사용하지 않고 개체를 삭제한 경우 `versionId` 버전 일시 중지된 버킷의 하위 리소스는 기존 'null' 버전 또는 'null' 삭제 표식을 영구적으로 삭제하고 새 'null' 삭제 표식을 생성합니다. 를 클릭합니다 `x-amz-delete-marker` 로 설정된 응답 헤더가 반환됩니다 `true`. • 참고 *: 경우에 따라 객체에 대해 여러 개의 삭제 마커가 존재할 수 있습니다. 을 참조하십시오 "S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다" 거버넌스 모드에서 오브젝트 버전을 삭제하는 방법을 알아보십시오.

작동	구축
여러 개체를 삭제합니다 (삭제 개체)	MFA(Multi-Factor Authentication) 및 응답 헤더입니다 `x-amz-mfa` 지원되지 않습니다. 동일한 요청 메시지에서 여러 객체를 삭제할 수 있습니다. 을 참조하십시오 "S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다" 거버넌스 모드에서 오브젝트 버전을 삭제하는 방법을 알아보십시오.
개체 태그 지정 삭제	를 사용합니다 `tagging` 개체에서 모든 태그를 제거하는 하위 리소스입니다. 버전 관리 를 누릅니다 `versionId` 쿼리 매개 변수가 요청에 지정되지 않았습니다. 이 작업은 버전이 지정된 버킷에 있는 개체의 최신 버전에서 모든 태그를 삭제합니다. 개체의 현재 버전이 삭제 표식이면 " MethodNotAllowed " 상태가 과 함께 반환됩니다 `x-amz-delete-marker` 응답 헤더가 로 설정되었습니다 `true`.
객체 가져오기	"객체 가져오기"
객체 ACL을 가져옵니다	계정에 필요한 액세스 자격 증명이 제공된 경우 이 작업은 개체 소유자의 ID, DisplayName 및 사용 권한과 함께 긍정적인 응답을 반환합니다. 이는 소유자가 개체에 대한 모든 액세스 권한을 가지고 있음을 나타냅니다.
객체 법적 증거 자료 보관	"S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"
개체 보존 가져오기	"S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"
객체 태그 지정 가져오기	를 사용합니다 `tagging` 개체의 모든 태그를 반환하는 하위 리소스입니다. 버전 관리 를 누릅니다 `versionId` 쿼리 매개 변수가 요청에 지정되지 않았습니다. 이 작업은 버전 관리되는 버킷에서 가장 최신 버전의 개체에 있는 모든 태그를 반환합니다. 개체의 현재 버전이 삭제 표식이면 " MethodNotAllowed " 상태가 과 함께 반환됩니다 `x-amz-delete-marker` 응답 헤더가 로 설정되었습니다 `true`.
헤드 개체	"헤드 개체"
사후 개체 복원	"사후 개체 복원"
개체 를 넣습니다	"개체 를 넣습니다"
개체 - 복사 를 선택합니다	"개체 - 복사 를 선택합니다"

작동	구축
개체를 법적 증거 자료 보관	"S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"
개체 보존	"S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"
개체 태그 지정	를 사용합니다 `tagging` 기존 개체에 태그 집합을 추가하는 하위 리소스입니다. 개체 태그 제한 새 개체를 업로드할 때 태그를 추가하거나 기존 개체에 태그를 추가할 수 있습니다. StorageGRID 및 Amazon S3 모두 각 오브젝트에 대해 최대 10개의 태그를 지원합니다. 개체와 관련된 태그에는 고유한 태그 키가 있어야 합니다. 태그 키의 길이는 최대 128자의 유니코드 문자이고 태그 값의 길이는 최대 256자의 유니코드 문자일 수 있습니다. 키와 값은 대/소문자를 구분합니다. 태그 업데이트 및 수집 동작 오브젝트 태그 지정을 사용하여 개체의 태그를 업데이트하는 경우 StorageGRID에서는 개체를 다시 수집하지 않습니다. 즉, 일치하는 ILM 규칙에 지정된 Ingest 동작 옵션이 사용되지 않습니다. ILM이 정상적인 백그라운드 ILM 프로세스에 의해 다시 평가될 때 업데이트로 인해 트리거되는 개체 배치에 대한 모든 변경 사항이 발생합니다. 즉, ILM 규칙이 수집 동작에 Strict 옵션을 사용하는 경우 필요한 개체 배치를 만들 수 없는 경우(예: 새로 필요한 위치를 사용할 수 없음) 작업이 수행되지 않습니다. 업데이트된 오브젝트는 필요한 배치가 가능할 때까지 현재 위치를 유지합니다. 충돌 해결 같은 키에 쓰는 두 클라이언트 등 충돌하는 클라이언트 요청은 "최근 성공" 기준으로 해결됩니다. "Latest-WINS" 평가 시기는 S3 클라이언트가 작업을 시작할 때가 아니라 StorageGRID 시스템이 지정된 요청을 완료하는 시점을 기준으로 합니다. 버전 관리 를 누릅니다 `versionId` 쿼리 매개 변수가 요청에 지정되지 않았습니다. 작업에서 버전 관리되는 버킷의 가장 최근 개체 버전에 태그를 추가합니다. 개체의 현재 버전이 삭제 표식이면 " MethodNotAllowed " 상태가 과 함께 반환됩니다 `x-amz-delete-marker` 응답 헤더가 로 설정되었습니다 `true`.
SelectObjectContent 를 선택합니다	"SelectObjectContent 를 선택합니다"

관련 정보

"S3 작업이 감사 로그에서 추적되었습니다"

S3 Select를 사용합니다

StorageGRID는 에 대해 다음과 같은 Amazon S3 Select 절, 데이터 유형 및 연산자를 지원합니다 "SelectObjectContent 명령".

목록에 없는 항목은 지원되지 않습니다.

구문은 을 참조하십시오 "SelectObjectContent 를 선택합니다". S3 Select에 대한 자세한 내용은 를 참조하십시오 "S3 Select용 AWS 문서".

S3 Select가 활성화된 테넌트 계정만 SelectObjectContent 쿼리를 실행할 수 있습니다. 를 참조하십시오 "S3 Select 사용에 대한 고려 사항 및 요구 사항".

절을 참조하십시오

- 목록을 선택합니다
- FROM 절입니다
- WHERE 절
- Limit 절

데이터 유형

- 불입니다
- 정수
- 문자열
- 부동
- 십진수, 숫자
- 타임 스탬프입니다

연산자

논리 연산자

- 및
- 아닙니다
- 또는

비교 연산자

- 를 누릅니다
- 를 누릅니다
- lt;=.(&L
- GT;=.(&T
- =

- =
- 를 누릅니다
- !=
- 사이
- 인치

패턴 일치 연산자

- 좋아요
- _
- %

단일 작업자

- NULL입니다
- NULL이 아닙니다

수학 연산자

- 를 누릅니다
- -
- *
- /
- %

StorageGRID는 Amazon S3 Select 운영자 우선권을 따릅니다.

집계 함수

- 평균()
- 개수(*)
- 최대()
- 최소()
- 합계()

조건부 함수

- 케이스
- 합체
- 노LIF

변환 함수

- 캐스트(지원되는 데이터 형식용)

날짜 함수

- date_add
- Date_DIFF(날짜/시간
- 압축 풀기
- to_string(대상 문자열)
- 를 _TIMESTAMP로 설정합니다
- UTCNOW

문자열 함수

- char_length, character_length
- 낮음
- 부분 문자열
- 잘라내기
- 위쪽

서버측 암호화를 사용합니다

서버측 암호화를 통해 유휴 개체 데이터를 보호할 수 있습니다. StorageGRID는 개체를 쓸 때 데이터를 암호화하고 개체에 액세스할 때 데이터를 해독합니다.

서버측 암호화를 사용하려면 암호화 키가 관리되는 방식에 따라 상호 배타적인 두 가지 옵션 중 하나를 선택할 수 있습니다.

- * SSE(StorageGRID 관리 키를 사용한 서버 측 암호화) *: S3 요청을 발행하여 오브젝트를 저장할 때 StorageGRID는 고유 키를 사용하여 오브젝트를 암호화합니다. S3 요청을 통해 오브젝트를 검색할 때 StorageGRID는 저장된 키를 사용하여 오브젝트를 해독합니다.
- * SSE-C(고객이 제공한 키를 사용한 서버측 암호화) *: S3 요청을 발행하여 오브젝트를 저장할 때 사용자는 자신만의 암호화 키를 제공합니다. 오브젝트를 검색할 때 요청의 일부로 동일한 암호화 키를 제공합니다. 두 암호화 키가 일치하면 해당 개체는 해독되고 개체 데이터는 반환됩니다.

  StorageGRID는 모든 개체 암호화 및 암호 해독 작업을 관리하지만 사용자가 제공하는 암호화 키를 관리해야 합니다.

  > 제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다.

  > 개체가 SSE 또는 SSE-C로 암호화된 경우 버킷 수준 또는 그리드 수준 암호화 설정은 무시됩니다.

SSE를 사용합니다

StorageGRID에서 관리하는 고유 키를 사용하여 개체를 암호화하려면 다음 요청 헤더를 사용합니다.

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

SSE 요청 헤더는 다음 오브젝트 작업에서 지원됩니다.

- "개체 를 넣습니다"
- "개체 - 복사 를 선택합니다"
- "멀티파트 업로드를 시작합니다"

SSE-C를 사용합니다

관리하는 고유 키로 개체를 암호화하려면 다음 세 가지 요청 헤더를 사용합니다.

요청 헤더	설명
`x-amz-server-side-encryption-customer-algorithm`	암호화 알고리즘을 지정합니다. 헤더 값은 이어야 합니다 `AES256`.
`x-amz-server-side-encryption-customer-key`	개체를 암호화하거나 해독하는 데 사용할 암호화 키를 지정합니다. 키의 값은 256비트 base64로 인코딩되어야 합니다.
`x-amz-server-side-encryption-customer-key-MD5`	RFC 1321에 따라 암호화 키의 MD5 다이제스트를 지정합니다. RFC 1321은 암호화 키가 오류 없이 전송되도록 하는 데 사용됩니다. MD5 다이제스트 값은 base64로 인코딩된 128비트여야 합니다.

SSE-C 요청 헤더는 다음 개체 작업에서 지원됩니다.

- "객체 가져오기"
- "헤드 개체"
- "개체 를 넣습니다"
- "개체 - 복사 를 선택합니다"
- "멀티파트 업로드를 시작합니다"
- "부품 업로드"
- "업로드 부품 - 복사"

고객이 제공한 키**(SSE-C)**와 함께 서버측 암호화 사용 시 고려 사항

SSE-C를 사용하기 전에 다음 사항을 고려하십시오.

- https를 사용해야 합니다.

> StorageGRID는 SSE-C를 사용할 때 http를 통해 이루어진 요청을 거부합니다 보안을 위해 실수로 http를 사용하여 보낸 모든 키가 손상되지 않도록 고려해야 합니다. 키를 폐기하고 필요에 따라 회전합니다.

- 응답의 ETag는 객체 데이터의 MD5가 아닙니다.
- 암호화 키를 개체에 매핑하는 작업을 관리해야 합니다. StorageGRID는 암호화 키를 저장하지 않습니다. 각 개체에

대해 제공하는 암호화 키를 추적할 책임은 사용자에게 있습니다.

- 버킷을 버전 관리가 활성화된 경우 각 오브젝트 버전에는 고유한 암호화 키가 있어야 합니다. 각 개체 버전에 사용되는 암호화 키를 추적할 책임은 사용자에게 있습니다.
- 클라이언트 측에서 암호화 키를 관리하기 때문에 클라이언트 측에서 키 회전과 같은 추가 보호 수단을 관리해야 합니다.

> ⓘ 제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다.

- 버킷에 대해 교차 그리드 복제 또는 CloudMirror 복제가 구성된 경우 SSE-C 객체를 수집할 수 없습니다. 수집 작업이 실패합니다.

관련 정보

"Amazon S3 개발자 가이드: 고객 제공 암호화 키(SSE-C)를 사용하여 서버측 암호화를 사용하여 데이터 보호"

객체 가져오기

S3 오브젝트 가져오기 요청을 사용하여 S3 버킷에서 오브젝트를 검색할 수 있습니다.

개체 및 다중 파트 개체를 가져옵니다

를 사용할 수 있습니다 `partNumber` multipart 또는 segmented object의 특정 부분을 검색하기 위한 request 매개 변수입니다. 를 클릭합니다 `x-amz-mp-parts-count` Response 요소는 개체에 있는 파트 수를 나타냅니다.

을 설정할 수 있습니다 `partNumber` 분할/다중 파트 개체 및 비분할/비다중 파트 개체 모두에 대해 1로, 그러나, 입니다 `x-amz-mp-parts-count` 응답 요소는 분할된 개체 또는 다중 파트 개체에 대해서만 반환됩니다.

사용자 메타데이터의 **UTF-8** 문자

StorageGRID는 사용자 정의 메타데이터에서 이스케이프된 UTF-8 문자를 구문 분석하거나 해석하지 않습니다. 사용자 정의 메타데이터에서 이스케이프된 UTF-8 문자가 있는 개체에 대한 요청을 가져오지 않습니다 `x-amz-missing-meta` 머리글 키 이름이나 값에 인쇄할 수 없는 문자가 포함된 경우.

지원되지 않는 요청 헤더입니다

다음 요청 헤더는 지원되지 않으며 반환됩니다 `XNotImplemented`:

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

버전 관리

가 있는 경우 `versionId` 하위 리소스가 지정되지 않았습니다. 작업이 버전 관리되는 버킷에서 개체의 최신 버전을 가져옵니다. 객체의 현재 버전이 삭제 마커인 경우 와 함께 ""찾을 수 없음" 상태가 반환됩니다 `x-amz-delete-marker` 응답 헤더가 로 설정되었습니다 `true`.

고객이 제공한 암호화 키**(SSE-C)**를 사용하여 서버측 암호화를 위한 요청 헤더

사용자가 제공한 고유 키로 개체를 암호화한 경우 머리글 3개를 모두 사용합니다.

- `x-amz-server-side-encryption-customer-algorithm`을 지정합니다 `AES256`.

- `x-amz-server-side-encryption-customer-key`: 오브젝트의 암호화 키를 지정합니다.
- `x-amz-server-side-encryption-customer-key-MD5`: 오브젝트의 암호화 키에 대한 MD5 다이제스트를 지정합니다.

> 제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다. 고객이 제공한 키를 사용하여 오브젝트 데이터를 보호하기 전에 의 고려 사항을 검토하십시오 "서버측 암호화를 사용합니다".

Get Object for Cloud Storage Pool 개체의 동작

개체가 에 저장된 경우 "클라우드 스토리지 풀"Get Object 요청의 동작은 개체의 상태에 따라 달라집니다. 을 참조하십시오 "헤드 개체" 를 참조하십시오.

> 객체가 클라우드 스토리지 풀에 저장되고 오브젝트 복사본이 하나 이상 그리드에 존재하는 경우, 객체 가져오기 요청은 클라우드 스토리지 풀에서 데이터를 검색하기 전에 그리드에서 데이터를 검색하려고 시도합니다.

개체의 상태입니다	**Get Object**의 동작입니다
StorageGRID로 수집되었지만 아직 ILM에 의해 평가되지 않은 오브젝트 또는 기존 스토리지 풀에 저장된 오브젝트 또는 삭제 코딩 사용	`200 OK` 개체의 복사본이 검색됩니다.
Cloud Storage Pool의 개체이지만 아직 검색할 수 없는 상태로 전환되지 않았습니다	`200 OK` 개체의 복사본이 검색됩니다.
개체가 검색할 수 없는 상태로 전환되었습니다	`403 Forbidden, InvalidObjectState` 를 사용합니다 "사후 개체 복원" 객체를 복구할 수 있는 상태로 복구하도록 요청합니다.
복구할 수 없는 상태에서 복원 중인 개체입니다	`403 Forbidden, InvalidObjectState` POST 개체 복원 요청이 완료될 때까지 기다립니다.
객체가 클라우드 스토리지 풀에 완전히 복구되었습니다	`200 OK` 개체의 복사본이 검색됩니다.

클라우드 스토리지 풀에서 다중 또는 분할 오브젝트

여러 부분 개체를 업로드했거나 StorageGRID가 큰 개체를 세그먼트로 분할한 경우 StorageGRID는 개체의 부분 또는 세그먼트의 하위 집합을 샘플링하여 클라우드 저장소 풀에서 개체를 사용할 수 있는지 여부를 결정합니다. 경우에 따라 Get Object 요청이 잘못 반환될 수 있습니다 `200 OK` 개체의 일부 부분이 이미 복구할 수 없는 상태로 전환되었거나 개체의 일부 부분이 아직 복원되지 않은 경우

다음과 같은 경우:

- Get Object 요청이 일부 데이터를 반환하지만 전송 도중에 중지될 수 있습니다.
- 후속 Get Object 요청이 반환될 수 있습니다 `403 Forbidden`.

객체 및 교차 그리드 복제 가져오기

를 사용하는 경우 "그리드 통합" 및 "교차 그리드 복제" 버킷에 대해 이 활성화된 경우 S3 클라이언트는 오브젝트 가져오기 요청을 실행하여 오브젝트의 복제 상태를 확인할 수 있습니다. 응답에는 StorageGRID에만 해당하는 것이 포함됩니다 `x-ntap-sg-cgr-replication-status` 다음 값 중 하나를 갖는 응답 헤더:

그리드	복제 상태입니다
출처	• * 성공 *: 복제가 성공했습니다. • * 보류 중 *: 객체가 아직 복제되지 않았습니다. • * 실패 *: 영구적인 장애로 인해 복제에 실패했습니다. 사용자가 오류를 해결해야 합니다.
목적지	• replica *: 객체가 소스 그리드에서 복제되었습니다.

StorageGRID는 을 지원하지 않습니다 `x-amz-replication-status` 머리글.

관련 정보

"S3 작업이 감사 로그에서 추적되었습니다"

헤드 개체

S3 헤드 오브젝트 요청을 사용하여 오브젝트 자체를 반환하지 않고 오브젝트에서 메타데이터를 검색할 수 있습니다. 객체가 클라우드 스토리지 풀에 저장된 경우 헤드 객체를 사용하여 객체의 전환 상태를 확인할 수 있습니다.

헤드 개체 및 다중 파트 개체

를 사용할 수 있습니다 `partNumber` multipart 또는 segmented 객체의 특정 부분에 대한 메타데이터를 검색하는 request 매개 변수입니다. 를 클릭합니다 `x-amz-mp-parts-count` Response 요소는 개체에 있는 파트 수를 나타냅니다.

을 설정할 수 있습니다 `partNumber` 분할/다중 파트 개체 및 비분할/비다중 파트 개체 모두에 대해 1로, 그러나, 입니다 `x-amz-mp-parts-count` 응답 요소는 분할된 개체 또는 다중 파트 개체에 대해서만 반환됩니다.

사용자 메타데이터의 **UTF-8** 문자

StorageGRID는 사용자 정의 메타데이터에서 이스케이프된 UTF-8 문자를 구문 분석하거나 해석하지 않습니다. 사용자 정의 메타데이터에서 이스케이프된 UTF-8 문자가 있는 개체에 대한 head 요청은 을 반환하지 않습니다 `x-amz-missing-meta` 머리글 키 이름이나 값에 인쇄할 수 없는 문자가 포함된 경우.

지원되지 않는 요청 헤더입니다

다음 요청 헤더는 지원되지 않으며 반환됩니다 `XNotImplemented`:

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

버전 관리

가 있는 경우 `versionId` 하위 리소스가 지정되지 않았습니다. 작업이 버전 관리되는 버킷에서 개체의 최신 버전을 가져옵니다. 객체의 현재 버전이 삭제 마커인 경우 와 함께 ""찾을 수 없음" 상태가 반환됩니다 `x-amz-delete-marker` 응답 헤더가 로 설정되었습니다 `true`.

고객이 제공한 암호화 키**(SSE-C)**를 사용하여 서버측 암호화를 위한 요청 헤더

사용자가 제공한 고유 키로 개체를 암호화한 경우 이 헤더 3개를 모두 사용합니다.

- `x-amz-server-side-encryption-customer-algorithm`을 지정합니다 `AES256`.
- `x-amz-server-side-encryption-customer-key`: 오브젝트의 암호화 키를 지정합니다.
- `x-amz-server-side-encryption-customer-key-MD5`: 오브젝트의 암호화 키에 대한 MD5 다이제스트를 지정합니다.

> 제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다. 고객이 제공한 키를 사용하여 오브젝트 데이터를 보호하기 전에 의 고려 사항을 검토하십시오 "서버측 암호화를 사용합니다".

Cloud Storage Pool 개체에 대한 헤드 개체 응답

개체가 에 저장된 경우 "클라우드 스토리지 풀"다음과 같은 응답 헤더가 반환됩니다.

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

응답 헤더는 클라우드 스토리지 풀로 이동되는 오브젝트의 상태에 대한 정보를 제공하며, 선택적으로 검색할 수 없는 상태로 전환된 후 복구됩니다.

개체의 상태입니다	헤드 객체에 대한 응답
StorageGRID로 수집되었지만 아직 ILM에 의해 평가되지 않은 오브젝트 또는 기존 스토리지 풀에 저장된 오브젝트 또는 삭제 코딩 사용	`200 OK` (특별한 응답 헤더가 반환되지 않습니다.)

개체의 상태입니다	헤드 객체에 대한 응답

Cloud Storage Pool의 개체이지만 아직 검색할 수 없는 상태로 전환되지 않았습니다	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"`
	
	개체가 검색할 수 없는 상태로 전환될 때까지 의 값은 입니다 `expiry-date` 앞으로 어느 정도 먼 시간으로 설정됩니다. 정확한 전환 시간은 StorageGRID 시스템에 의해 제어되지 않습니다.

개체가 검색할 수 없는 상태로 전환되었지만 하나 이상의 복사본이 그리드에 있습니다	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"`
	
	의 값 `expiry-date` 앞으로 어느 정도 먼 시간으로 설정됩니다.
	
	• 참고 *: 그리드의 복제본을 사용할 수 없는 경우(예: 스토리지 노드가 다운된 경우) 를 실행해야 합니다 "사후 개체 복원" 오브젝트를 성공적으로 검색하기 전에 클라우드 스토리지 풀에서 복사본을 복구하도록 요청합니다.

개체가 검색할 수 없는 상태로 전환되었으며 그리드에 복사본이 없습니다	`200 OK`
	
	`x-amz-storage-class: GLACIER`

복구할 수 없는 상태에서 복원 중인 개체입니다	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="true"`

개체의 상태입니다	헤드 객체에 대한 응답

객체가 클라우드 스토리지 풀에 완전히 복구되었습니다	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2018 00:00:00 GMT"`
	
	를 클릭합니다 `expiry-date` 클라우드 스토리지 풀의 객체가 검색 불가능한 상태로 반환되는 시점을 나타냅니다.

Cloud Storage Pool에서 다중 또는 분할 오브젝트 지원

여러 부분 개체를 업로드했거나 StorageGRID가 큰 개체를 세그먼트로 분할한 경우 StorageGRID는 개체의 부분 또는 세그먼트의 하위 집합을 샘플링하여 클라우드 저장소 풀에서 개체를 사용할 수 있는지 여부를 결정합니다. 경우에 따라 헤드 객체 요청이 잘못 반환될 수 있습니다 `x-amz-restore: ongoing-request="false"` 개체의 일부 부분이 이미 복구할 수 없는 상태로 전환되었거나 개체의 일부 부분이 아직 복원되지 않은 경우

Head Object 및 **Cross-Grid** 복제

를 사용하는 경우 "그리드 통합" 및 "교차 그리드 복제" 버킷에 대해 이 활성화된 경우 S3 클라이언트는 헤드 오브젝트 요청을 실행하여 오브젝트의 복제 상태를 확인할 수 있습니다. 응답에는 StorageGRID에만 해당하는 것이 포함됩니다 `x-ntap-sg-cgr-replication-status` 다음 값 중 하나를 갖는 응답 헤더:

그리드	복제 상태입니다
출처	• * 성공 *: 복제가 성공했습니다. • * 보류 중 *: 객체가 아직 복제되지 않았습니다. • * 실패 *: 영구적인 장애로 인해 복제에 실패했습니다. 사용자가 오류를 해결해야 합니다.
목적지	• replica *: 객체가 소스 그리드에서 복제되었습니다.

StorageGRID는 을 지원하지 않습니다 `x-amz-replication-status` 머리글.

관련 정보

"S3 작업이 감사 로그에서 추적되었습니다"

사후 개체 복원

S3 POST 오브젝트 복원 요청을 사용하여 클라우드 스토리지 풀에 저장된 오브젝트를 복원할 수 있습니다.

지원되는 요청 유형입니다

StorageGRID는 개체 복원을 위한 POST 개체 복원 요청만 지원합니다. 는 지원하지 않습니다 `SELECT` 복원 유형. 반품 요청을 선택합니다 `XNotImplemented`.

버전 관리

필요에 따라 를 지정합니다 `versionId` 버전 관리되는 버킷에서 특정 버전의 오브젝트 복원 를 지정하지 않은 경우 `versionId`, 개체의 최신 버전이 복원됩니다

클라우드 스토리지 풀 객체에 대한 **POST** 객체 복구의 동작

개체가 클라우드 스토리지 풀에 저장된 경우(정보 수명 주기 관리를 통해 개체 관리 지침 참조) POST 개체 복원 요청은 개체의 상태에 따라 다음과 같은 동작을 수행합니다. 자세한 내용은 " 헤드 개체 "를 참조하십시오.

> 객체가 클라우드 스토리지 풀에 저장되어 있고 하나 이상의 오브젝트 복제본도 그리드에 있는 경우 POST 객체 복원 요청을 실행하여 객체를 복원할 필요가 없습니다. 대신 Get Object 요청을 사용하여 로컬 복사본을 직접 검색할 수 있습니다.

개체의 상태입니다	**POST** 개체 복원 동작
StorageGRID로 수집되었지만 ILM에서 아직 평가되지 않은 오브젝트 또는 클라우드 스토리지 풀에 없는 오브젝트	`403 Forbidden`, `InvalidObjectState`
Cloud Storage Pool의 개체이지만 아직 검색할 수 없는 상태로 전환되지 않았습니다	`200 OK` 변경 사항이 없습니다. • 참고 *: 개체가 검색할 수 없는 상태로 전환되기 전에는 개체를 변경할 수 없습니다 `expiry-date`.
개체가 검색할 수 없는 상태로 전환되었습니다	`202 Accepted` 요청 본문에서 지정한 일 수에 대해 검색할 수 있는 객체 복제본을 클라우드 스토리지 풀에 복구합니다. 이 기간이 끝나면 객체는 복구할 수 없는 상태로 돌아갑니다. 필요에 따라 를 사용합니다 `Tier` 복원 작업을 완료하는 데 걸리는 시간을 결정하는 요청 요소입니다 (`Expedited`, `Standard`, 또는 `Bulk`)를 클릭합니다. 를 지정하지 않은 경우 `Tier`, `Standard` 계층이 사용됩니다. • 중요 *: 오브젝트가 S3 Glacier Deep Archive로 전환된 경우 또는 Cloud Storage Pool에서 Azure Blob 스토리지를 사용하는 경우 를 사용하여 복원할 수 없습니다 `Expedited` 계층. 다음 오류가 반환됩니다 `403 Forbidden`, `InvalidTier`: `Retrieval option is not supported by this storage class`.
복구할 수 없는 상태에서 복원 중인 개체입니다	`409 Conflict`, `RestoreAlreadyInProgress`

개체의 상태입니다	**POST** 개체 복원 동작
객체가 클라우드 스토리지 풀에 완전히 복구되었습니다	`200 OK` • 참고: * 개체가 검색 가능한 상태로 복원되면 이를 변경할 수 있습니다 `expiry-date` 에 대한 새 값을 사용하여 POST 개체 복원 요청을 다시 발행합니다 `Days`. 복원 날짜는 요청 시간을 기준으로 업데이트됩니다.

관련 정보

"ILM을 사용하여 개체를 관리합니다"

"헤드 개체"

"S3 작업이 감사 로그에서 추적되었습니다"

개체 를 넣습니다

S3 PUT 오브젝트 요청을 사용하여 오브젝트를 버킷에 추가할 수 있습니다.

충돌 해결

동일한 키에 쓰는 두 클라이언트 등의 충돌하는 클라이언트 요청은 "최신 성공" 기준으로 해결됩니다. "Latest-WINS" 평가 시기는 S3 클라이언트가 작업을 시작할 때가 아니라 StorageGRID 시스템이 지정된 요청을 완료하는 시점을 기준으로 합니다.

개체 크기

단일 PUT 오브젝트 작업의 maximum_recommended_size는 5GiB(5,368,709,120바이트)입니다. 5GiB보다 큰 객체가 있는 경우 대신 멀티파트 업로드를 사용합니다.

단일 PUT 객체 작업의 Maximum_supported_size는 5TiB(5,497,558,138,880바이트)입니다. 그러나 5GiB를 초과하는 개체를 업로드하려고 하면 * S3 PUT 오브젝트 크기가 너무 큼 * 경고가 트리거됩니다.

사용자 메타데이터 크기입니다

Amazon S3는 각 PUT 요청 헤더 내의 사용자 정의 메타데이터 크기를 2KB로 제한합니다. StorageGRID는 사용자 메타데이터를 24KiB로 제한합니다. 사용자 정의 메타데이터의 크기는 각 키와 값의 UTF-8 인코딩에서 바이트 수의 합계를 구하여 측정됩니다.

사용자 메타데이터의 **UTF-8** 문자

요청에 사용자 정의 메타데이터의 키 이름이나 값에 UTF-8 값이 포함되어 있으면 StorageGRID 동작이 정의되지 않습니다.

StorageGRID는 사용자 정의 메타데이터의 키 이름이나 값에 포함된 이스케이프된 UTF-8 문자를 구문 분석하거나 해석하지 않습니다. 이스케이프된 UTF-8 문자는 ASCII 문자로 처리됩니다.

- 사용자 정의 메타데이터에 이스케이프된 UTF-8 문자가 포함된 경우 Put, Put Object-Copy, Get 및 head 요청이 성공합니다.
- StorageGRID는 을 반환하지 않습니다 `x-amz-missing-meta` 머리글 키 이름이나 값의 해석된 값에 인쇄할 수

없는 문자가 포함된 경우.

개체 태그 제한

새 개체를 업로드할 때 태그를 추가하거나 기존 개체에 태그를 추가할 수 있습니다. StorageGRID 및 Amazon S3 모두 각 오브젝트에 대해 최대 10개의 태그를 지원합니다. 개체와 관련된 태그에는 고유한 태그 키가 있어야 합니다. 태그 키의 길이는 최대 128자의 유니코드 문자이고 태그 값의 길이는 최대 256자의 유니코드 문자일 수 있습니다. 키와 값은 대/소문자를 구분합니다.

개체 소유권

StorageGRID에서는 소유자가 아닌 계정 또는 익명 사용자가 만든 개체를 포함하여 모든 개체가 버킷 소유자 계정에 의해 소유됩니다.

지원되는 요청 헤더입니다

지원되는 요청 헤더는 다음과 같습니다.

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

  를 지정할 때 `aws-chunked` 용 `Content-Encoding`StorageGRID는 다음 항목을 확인하지 않습니다.

  - StorageGRID에서 를 확인하지 않습니다 `chunk-signature` 청크 데이터를 기준으로 합니다.
  - StorageGRID는 사용자가 제공하는 값을 확인하지 않습니다 `x-amz-decoded-content-length` 반대.
- `Content-Language`
- `Content-Length`
- `Content-MD5`
- `Content-Type`
- `Expires`
- `Transfer-Encoding`

  청크된 전송 인코딩이 지원되는 경우 `aws-chunked` 페이로드 서명도 사용됩니다.

- `x-amz-meta-`사용자 정의 메타데이터를 포함하는 이름-값 쌍이 뒤에 옵니다.

  사용자 정의 메타데이터에 대한 이름 값 쌍을 지정할 때 다음 일반 형식을 사용합니다.

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

  ILM 규칙의 참조 시간으로 * 사용자 정의 생성 시간 * 옵션을 사용하려면 을 사용해야 합니다 `creation-time` 오브젝트를 만들 때 기록하는 메타데이터의 이름입니다. 예를 들면 다음과 같습니다.

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

의 값 `creation-time` 1970년 1월 1일 이후 초 단위로 평가됩니다.

> ILM 규칙은 참조 시간에 대해 * 사용자 정의 작성 시간 * 과 수집 동작에 대한 균형 또는 엄격 옵션을 모두 사용할 수 없습니다. ILM 규칙을 만들면 오류가 반환됩니다.

- `x-amz-tagging`
- S3 오브젝트 잠금 요청 헤더
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    이러한 헤더 없이 요청이 이루어지면 버킷 기본 보존 설정을 사용하여 오브젝트 버전 모드와 보존 기간을 계산합니다. 을 참조하십시오 "S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다".
- SSE 요청 헤더:
  - `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`

    을 참조하십시오 서버측 암호화에 대한 요청 헤더

지원되지 않는 요청 헤더입니다

다음 요청 헤더는 지원되지 않습니다.

- 를 클릭합니다 `x-amz-acl` 요청 헤더가 지원되지 않습니다.
- 를 클릭합니다 `x-amz-website-redirect-location` 요청 헤더가 지원되지 않으며 반환됩니다 `XNotImplemented`.

스토리지 클래스 옵션

를 클릭합니다 `x-amz-storage-class` 요청 헤더가 지원됩니다. 에 대해 제출된 값입니다 `x-amz-storage-class` ILM을 통해 결정되는 StorageGRID 시스템에 저장된 개체의 영구 복사본 수가 아닌 수집 중에 StorageGRID이 오브젝트 데이터를 보호하는 방법에 영향을 미칩니다.

인제스트 개체와 일치하는 ILM 규칙이 Ingest 동작에 대해 Strict 옵션을 사용하는 경우, 를 참조하십시오 `x-amz-storage-class` 머리글은 효과가 없습니다.

에 사용할 수 있는 값은 다음과 같습니다 `x-amz-storage-class`:

- `STANDARD` (기본값)

- * 이중 커밋 *: ILM 규칙이 Ingest 동작에 대한 이중 커밋 옵션을 지정하는 경우, 개체가 수집되는 즉시 해당 개체의 두 번째 복사본이 생성되어 다른 스토리지 노드(이중 커밋)에 배포됩니다. ILM을 평가할 때 StorageGRID는 이러한 초기 중간 복사본이 규칙의 배치 지침을 충족하는지 여부를 결정합니다. 그렇지 않으면 새 오브젝트 복사본을 다른 위치에 만들어야 하고 초기 중간 복사본을 삭제해야 할 수 있습니다.
- * 균형 *: ILM 규칙이 균형 옵션을 지정하고 StorageGRID이 규칙에 지정된 모든 복사본을 즉시 만들 수 없는 경우 StorageGRID은 다른 스토리지 노드에 두 개의 임시 복사본을 만듭니다.

  StorageGRID에서 ILM 규칙(동기식 배치)에 지정된 모든 오브젝트 복사본을 즉시 생성할 수 있는 경우 를 참조하십시오 `x-amz-storage-class` 머리글은 효과가 없습니다.

- `REDUCED_REDUNDANCY`
  - * 이중 커밋 *: ILM 규칙이 Ingest 동작에 대한 이중 커밋 옵션을 지정하는 경우 StorageGRID는 오브젝트가 수집될 때(단일 커밋) 단일 임시 복사본을 만듭니다.
  - * 균형 *: ILM 규칙이 균형 옵션을 지정하는 경우 StorageGRID은 시스템에서 규칙에 지정된 모든 사본을 즉시 만들 수 없는 경우에만 단일 중간 복사본을 만듭니다. StorageGRID에서 동기 배치를 수행할 수 있는 경우 이 머리글은 영향을 주지 않습니다. 를 클릭합니다 `REDUCED_REDUNDANCY` 옵션은 개체와 일치하는 ILM 규칙이 복제된 단일 복사본을 만들 때 가장 적합합니다. 이 경우 를 사용합니다 `REDUCED_REDUNDANCY` 모든 수집 작업에 대해 불필요한 오브젝트 복사본을 생성 및 삭제할 필요가 없습니다.

  를 사용합니다 `REDUCED_REDUNDANCY` 다른 상황에서는 옵션을 사용하지 않는 것이 좋습니다. `REDUCED_REDUNDANCY` 수집 중에 오브젝트 데이터가 손실될 위험이 증가합니다. 예를 들어, ILM 평가가 발생하기 전에 실패한 스토리지 노드에 단일 복사본이 처음 저장되는 경우 데이터가 손실될 수 있습니다.

> 복제된 복사본이 항상 하나만 있으면 데이터가 영구적으로 손실될 위험이 있습니다. 복제된 객체 복제본이 하나만 있는 경우 스토리지 노드에 장애가 발생하거나 심각한 오류가 발생한 경우 해당 객체가 손실됩니다. 또한 업그레이드와 같은 유지보수 절차 중에는 개체에 대한 액세스가 일시적으로 중단됩니다.

지정 `REDUCED_REDUNDANCY` 오브젝트를 처음 인제스트할 때 생성되는 복사본 수에만 영향을 줍니다. 활성 ILM 정책에 따라 개체를 평가할 때 개체의 복사본 수에 영향을 주지 않으며 StorageGRID 시스템에서 낮은 수준의 중복성에 데이터가 저장되지 않습니다.

> S3 오브젝트 잠금이 설정된 버킷으로 오브젝트를 밀어넣는 경우, 를 참조하십시오 `REDUCED_REDUNDANCY` 옵션이 무시됩니다. 오브젝트를 레거시 준수 버킷으로 인스팅하는 경우, 를 참조하십시오 `REDUCED_REDUNDANCY` 옵션을 사용하면 오류가 반환됩니다. StorageGRID은 규정 준수 요구 사항이 충족될 수 있도록 항상 이중 커밋 수집을 수행합니다.

서버측 암호화에 대한 요청 헤더

다음 요청 헤더를 사용하여 서버측 암호화를 사용하여 개체를 암호화할 수 있습니다. SSE 및 SSE-C 옵션은 상호 배타적입니다.

- * SSE *: StorageGRID에서 관리하는 고유 키를 사용하여 오브젝트를 암호화하려면 다음 헤더를 사용하십시오.
  - `x-amz-server-side-encryption`
- * SSE-C *: 사용자가 제공 및 관리하는 고유 키로 객체를 암호화하려면 이 헤더 세 개를 모두 사용합니다.
  - `x-amz-server-side-encryption-customer-algorithm`을 지정합니다 `AES256`.
  - `x-amz-server-side-encryption-customer-key`새 오브젝트의 암호화 키를 지정합니다.

- `x-amz-server-side-encryption-customer-key-MD5`: 새 개체의 암호화 키에 대한 MD5 다이제스트를 지정합니다.

> 제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다. 고객이 제공한 키를 사용하여 오브젝트 데이터를 보호하기 전에 의 고려 사항을 검토하십시오 "서버 측 암호화 사용".

> 개체가 SSE 또는 SSE-C로 암호화된 경우 버킷 수준 또는 그리드 수준 암호화 설정은 무시됩니다.

버전 관리

버켓에 대해 버전 관리가 활성화된 경우 고유한 가 사용됩니다 `versionId` 는 저장 중인 개체의 버전에 대해 자동으로 생성됩니다. 여기 `versionId` 를 사용하여 응답에서도 반환됩니다 `x-amz-version-id` 응답 헤더.

버전 관리가 일시 중단된 경우 개체 버전은 null로 저장됩니다 `versionId` null 버전이 이미 있는 경우 덮어쓰기가 됩니다.

승인 헤더의 서명 계산

를 사용할 때 `Authorization` 헤더 요청 인증, StorageGRID는 다음과 같은 방식으로 AWS와 다릅니다.

- StorageGRID은 필요하지 않습니다 `host` 에 포함할 헤더 `CanonicalHeaders`.
- StorageGRID은 필요하지 않습니다 `Content-Type` 에 포함될 수 있습니다 `CanonicalHeaders`.
- StorageGRID은 필요하지 않습니다 `x-amz-*` 에 포함할 헤더 `CanonicalHeaders`.

> 일반적인 모범 사례로서 항상 이 머리글을 안에 포함합니다 `CanonicalHeaders` 그러나 이러한 헤더를 제외하는 경우 StorageGRID에서 오류를 반환하지 않습니다.

자세한 내용은 을 참조하십시오 "승인 헤더에 대한 서명 계산:단일 청크(AWS 서명 버전 4)로 페이로드 전송".

관련 정보

"ILM을 사용하여 개체를 관리합니다"

"버킷 작업"

"S3 작업이 감사 로그에서 추적되었습니다"

"클라이언트 연결 구성 방법"

개체 - 복사 를 선택합니다

S3 PUT 오브젝트 복사 요청을 사용하여 S3에 이미 저장된 오브젝트 복사본을 생성할 수 있습니다. Put Object - Copy 작업은 GET 및 PUT를 수행하는 작업과 동일합니다.

충돌 해결

동일한 키에 쓰는 두 클라이언트 등의 충돌하는 클라이언트 요청은 "최신 성공" 기준으로 해결됩니다. "Latest-WINS" 평가 시기는 S3 클라이언트가 작업을 시작할 때가 아니라 StorageGRID 시스템이 지정된 요청을 완료하는 시점을 기준으로 합니다.

개체 크기

단일 PUT 오브젝트 작업의 maximum_recommended_size는 5GiB(5,368,709,120바이트)입니다. 5GiB보다 큰 객체가 있는 경우 대신 멀티파트 업로드를 사용합니다.

단일 PUT 객체 작업의 Maximum_supported_size는 5TiB(5,497,558,138,880바이트)입니다. 그러나 5GiB를 초과하는 개체를 업로드하려고 하면 * S3 PUT 오브젝트 크기가 너무 큼 * 경고가 트리거됩니다.

사용자 메타데이터의 **UTF-8** 문자

요청에 사용자 정의 메타데이터의 키 이름이나 값에 UTF-8 값이 포함되어 있으면 StorageGRID 동작이 정의되지 않습니다.

StorageGRID는 사용자 정의 메타데이터의 키 이름이나 값에 포함된 이스케이프된 UTF-8 문자를 구문 분석하거나 해석하지 않습니다. 이스케이프된 UTF-8 문자는 ASCII 문자로 처리됩니다.

- 사용자 정의 메타데이터에 이스케이프된 UTF-8 문자가 포함된 경우 요청이 성공합니다.
- StorageGRID는 을 반환하지 않습니다 `x-amz-missing-meta` 머리글 키 이름이나 값의 해석된 값에 인쇄할 수 없는 문자가 포함된 경우.

지원되는 요청 헤더입니다

지원되는 요청 헤더는 다음과 같습니다.

- `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-`사용자 정의 메타데이터를 포함하는 이름-값 쌍이 뒤에 옵니다
- `x-amz-metadata-directive`기본값은 입니다 `COPY`, 개체 및 관련 메타데이터를 복사할 수 있습니다.

  지정할 수 있습니다 `REPLACE` 오브젝트를 복사할 때 기존 메타데이터를 덮어쓰거나 오브젝트 메타데이터를 업데이트합니다.

- `x-amz-storage-class`
- `x-amz-tagging-directive`기본값은 입니다 `COPY`, 개체 및 모든 태그를 복사할 수 있습니다.

  지정할 수 있습니다 `REPLACE` 개체를 복사할 때 기존 태그를 덮어쓰거나 태그를 업데이트합니다.

- S3 오브젝트 잠금 요청 헤더:
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

이러한 헤더 없이 요청이 이루어지면 버킷 기본 보존 설정을 사용하여 오브젝트 버전 모드와 보존 기간을 계산합니다. 을 참조하십시오 "S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다".

- SSE 요청 헤더:
  - `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`

  을 참조하십시오 서버측 암호화에 대한 요청 헤더

지원되지 않는 요청 헤더입니다

다음 요청 헤더는 지원되지 않습니다.

- `Cache-Control`
- `Content-Disposition`
- `Content-Encoding`
- `Content-Language`
- `Expires`
- `x-amz-website-redirect-location`

스토리지 클래스 옵션

를 클릭합니다 `x-amz-storage-class` 요청 헤더가 지원되며 일치하는 ILM 규칙에서 이중 커밋 또는 균형 조정의 수집 동작을 지정하는 경우 StorageGRID에서 만드는 개체 복사본 수에 영향을 줍니다.

- `STANDARD`

  (기본값) ILM 규칙이 이중 커밋 옵션을 사용하거나 균형 옵션이 중간 복사본 생성으로 돌아올 때 이중 커밋 수집 작업을 지정합니다.

- `REDUCED_REDUNDANCY`

  ILM 규칙이 이중 커밋 옵션을 사용하거나 균형 옵션이 중간 복사본 생성으로 돌아올 때 단일 커밋 수집 작업을 지정합니다.

  S3 오브젝트 잠금이 설정된 버킷으로 오브젝트를 밀어넣는 경우, 를 참조하십시오 `REDUCED_REDUNDANCY` 옵션이 무시됩니다. 오브젝트를 레거시 준수 버킷으로 인스팅하는 경우, 를 참조하십시오 `REDUCED_REDUNDANCY` 옵션을 사용하면 오류가 반환됩니다. StorageGRID은 규정 준수 요구 사항이 충족될 수 있도록 항상 이중 커밋 수집을 수행합니다.

Put Object - Copy에서 x-amz-copy-source 사용

소스 버킷과 키가 에 지정된 경우 `x-amz-copy-source` 헤더 는 대상 버킷 및 키와 다르며 소스 오브젝트 데이터의 복제본이 대상에 기록됩니다.

소스 및 대상이 일치하면, 및 입니다 `x-amz-metadata-directive` 머리글은 로 지정됩니다 `REPLACE`오브젝트의 메타데이터는 요청에 제공된 메타데이터 값으로 업데이트됩니다. 이 경우 StorageGRID는 오브젝트를 다시 수집하지 않습니다. 여기에는 두 가지 중요한 결과가 있습니다.

- Put Object-Copy를 사용하여 기존 개체를 현재 위치에서 암호화하거나 기존 개체의 암호화를 변경할 수 없습니다. 를 공급하는 경우 `x-amz-server-side-encryption` 머리글 또는 을 선택합니다 `x-amz-server-side-encryption-customer-algorithm` header, StorageGRID가 요청을 거부하고 반환합니다 `XNotImplemented`.
- 일치하는 ILM 규칙에 지정된 Ingest 동작 옵션은 사용되지 않습니다. ILM이 정상적인 백그라운드 ILM 프로세스에 의해 다시 평가될 때 업데이트로 인해 트리거되는 개체 배치에 대한 모든 변경 사항이 발생합니다.

  즉, ILM 규칙이 수집 동작에 Strict 옵션을 사용하는 경우 필요한 개체 배치를 만들 수 없는 경우(예: 새로 필요한 위치를 사용할 수 없음) 작업이 수행되지 않습니다. 업데이트된 오브젝트는 필요한 배치가 가능할 때까지 현재 위치를 유지합니다.

서버측 암호화에 대한 요청 헤더

서버 측 암호화를 사용하는 경우 소스 개체가 암호화되었는지 여부 및 대상 개체를 암호화할 계획인지에 따라 요청 헤더가 제공됩니다.

- 소스 객체가 SSE-C(customer-provided key)를 사용하여 암호화된 경우, 객체를 해독한 다음 복사할 수 있도록 객체 복사 요청(Put Object-Copy request)에 다음 세 개의 헤더를 포함해야 합니다.
  - `x-amz-copy-source-server-side-encryption-customer-algorithm`을 지정합니다 `AES256`.
  - `x-amz-copy-source-server-side-encryption-customer-key`: 소스 객체를 만들 때 제공한 암호화 키를 지정합니다.
  - `x-amz-copy-source-server-side-encryption-customer-key-MD5`: 소스 개체를 만들 때 제공한 MD5 다이제스트를 지정합니다.
- 제공 및 관리하는 고유 키를 사용하여 대상 개체(복사본)를 암호화하려면 다음 세 개의 머리글을 포함합니다.
  - `x-amz-server-side-encryption-customer-algorithm`을 지정합니다 `AES256`.
  - `x-amz-server-side-encryption-customer-key`: 대상 오브젝트의 새 암호화 키를 지정합니다.
  - `x-amz-server-side-encryption-customer-key-MD5`: 새 암호화 키의 MD5 다이제스트를 지정합니다.

  > 제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다. 고객이 제공한 키를 사용하여 오브젝트 데이터를 보호하기 전에 의 고려 사항을 검토하십시오 "서버 측 암호화 사용".

- SSE(StorageGRID)에서 관리되는 고유 키로 대상 객체(사본)를 암호화하려면 객체 복사 요청(Put Object-Copy request)에 이 헤더를 포함시킵니다.
  - `x-amz-server-side-encryption`

를 클릭합니다 `server-side-encryption` 개체의 값을 업데이트할 수 없습니다. 대신 새로 복사본을 만듭니다 `server-side-encryption` 값 사용 `x-amz-metadata-directive: REPLACE`.

버전 관리

소스 버킷의 버전이 있는 경우 를 사용할 수 있습니다 `x-amz-copy-source` Header - 개체의 최신 버전을 복사합니다. 특정 버전의 개체를 복사하려면 을 사용하여 복사할 버전을 명시적으로 지정해야 합니다 `versionId` 하위 리소스. 대상 버킷의 버전이 지정된 경우 생성된 버전이 에서 반환됩니다 `x-amz-version-id` 응답 헤더. 타겟 버킷에 대한 버전 관리가 일시 중지된 경우 `x-amz-version-id` ""null"" 값을 반환합니다.

관련 정보

"ILM을 사용하여 개체를 관리합니다"

"S3 작업이 감사 로그에서 추적되었습니다"

"개체 를 넣습니다"

SelectObjectContent 를 선택합니다

S3 SelectObjectContent 요청을 사용하여 간단한 SQL 문을 기반으로 S3 개체의 내용을 필터링할 수 있습니다.

자세한 내용은 를 참조하십시오 "SelectObjectContent에 대한 AWS 문서".

시작하기 전에

- 테넌트 계정에 S3 Select 권한이 있습니다.
- 있습니다 `s3:GetObject` 쿼리할 객체에 대한 권한입니다.
- 쿼리할 객체는 다음 형식 중 하나여야 합니다.
  - CSV *. GZIP 또는 BZIP2 보관 파일로 압축하거나 그대로 사용할 수 있습니다.
  - * 파케 *. Parquet 객체에 대한 추가 요구 사항:
    - S3 Select는 GZIP 또는 Snappy를 사용한 컬럼 압축만 지원합니다. S3 Select는 Parquet 오브젝트에 대한 전체 오브젝트 압축을 지원하지 않습니다.
    - S3 Select는 Parquet 출력을 지원하지 않습니다. 출력 형식을 CSV 또는 JSON으로 지정해야 합니다.
    - 압축되지 않은 최대 행 그룹 크기는 512MB입니다.
    - 개체의 스키마에 지정된 데이터 형식을 사용해야 합니다.
    - 간격, JSON, 목록, 시간 또는 UUID 논리적 유형은 사용할 수 없습니다.
- SQL 식의 최대 길이는 256KB입니다.
- 입력 또는 결과에 있는 모든 레코드의 최대 길이는 1MiB입니다.

ScanRange 사용이 지원되지 않습니다.

CSV 요청 구문 예

```
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>
```

쪽모이 세공 요청 구문 예

```
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 쿼리의 예

이 쿼리는 시/도 이름, 2010년 인구, 2015년 예상 인구, 미국 인구 조사 데이터의 변경 비율을 가져옵니다. 상태가 아닌 파일의 레코드는 무시됩니다.

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

질의할 파일의 처음 몇 줄, `SUB-EST2020_ALL.csv`다음과 같이 보십시오.

```
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 사용 예(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
```

출력 파일의 처음 몇 줄, `changes.csv`다음과 같이 보십시오.

```
Alabama,4779736,4854803,1.570526070895965802295356898372629785 4
Alaska,710231,738430,3.970398363349389142405780654463125377 5
Arizona,6392017,6832810,6.895992297892824753125656580700583243 1
Arkansas,2915918,2979732,2.188470320495981025529524492801237894 9
California,37253956,38904296,4.429972483996062055798852610444914897 1
Colorado,5029196,5454328,8.453279609703022113276157859029554624 6
```

AWS-CLI 사용 예(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
```

출력 파일의 처음 몇 줄인 changes.csv는 다음과 같습니다.

```
Alabama,4779736,4854803,1.570526070895965802295356898372629785 4
Alaska,710231,738430,3.970398363349389142405780654463125377 5
Arizona,6392017,6832810,6.895992297892824753125656580700583243 1
Arkansas,2915918,2979732,2.188470320495981025529524492801237894 9
California,37253956,38904296,4.429972483996062055798852610444914897 1
Colorado,5029196,5454328,8.453279609703022113276157859029554624 6
```

멀티파트 업로드 작업

이 섹션에서는 StorageGRID가 멀티파트 업로드 작업을 지원하는 방법에 대해 설명합니다.

다음 조건 및 참고 사항은 모든 다중 파트 업로드 작업에 적용됩니다.

- 해당 버킷에 대한 다중 파트 업로드 나열 쿼리 결과는 불완전한 결과를 반환할 수 있으므로 단일 버킷에 대한 동시 다중 파트 업로드 1,000개를 초과할 수 없습니다.
- StorageGRID는 여러 파트에 대해 AWS 크기 제한을 적용합니다. S3 클라이언트는 다음 지침을 따라야 합니다.
  - 멀티파트 업로드의 각 파트는 5MiB(5,242,880바이트)와 5GiB(5,368,709,120바이트) 사이여야 합니다.
  - 마지막 부분은 5MiB(5,242,880바이트)보다 작을 수 있습니다.
  - 일반적으로 파트 크기는 가능한 한 커야합니다. 예를 들어, 100GiB 개체의 경우 5GiB의 파트 크기를 사용합니다. 각 파트는 고유한 개체로 간주되므로 큰 파트 크기를 사용하면 StorageGRID 메타데이터 오버헤드가 줄어듭니다.
  - 5GiB보다 작은 오브젝트의 경우 대신 비다중 파트 업로드를 사용하는 것이 좋습니다.
- ILM 규칙이 Balanced 또는 Strict 수집 동작을 사용하는 경우 ILM은 다중 파트 개체의 각 부분을 인제스트할 때 계산되고 다중 파트 업로드가 완료될 때 전체 개체에 대해 평가됩니다. 이 사항이 개체 및 파트 배치에 미치는 영향에 대해 알고 있어야 합니다.
  - S3 멀티파트 업로드가 진행 중인 동안 ILM이 변경되면 멀티파트 업로드가 완료될 때 개체의 일부 부분이 현재 ILM 요구 사항을 충족하지 못할 수 있습니다. 올바르게 배치되지 않은 모든 부품은 ILM 재평가를 위해 대기 중이며 나중에 올바른 위치로 이동됩니다.
  - 파트에 대한 ILM을 평가할 때 StorageGRID은 개체의 크기가 아닌 파트 크기를 필터링합니다. 즉, 개체의 일부를 개체에 대한 ILM 요구 사항을 전체가 충족하지 않는 위치에 저장할 수 있습니다. 예를 들어, 규칙이 모든 오브젝트 10GB 이상이 DC1에 저장되는 반면 모든 작은 오브젝트는 DC2에 저장되는 것으로 지정하는 경우

10개 부분 멀티파트 업로드의 각 1GB 부분은 DC2에 저장됩니다. 개체에 대한 ILM을 전체적으로 평가할 때 개체의 모든 부분이 DC1로 이동합니다.

- 모든 멀티파트 업로드 작업은 StorageGRID 정합성 제어를 지원합니다.
- 필요한 경우 다중 파트 업로드와 함께 서버측 암호화를 사용할 수 있습니다. SSE(StorageGRID 관리 키 사용 시 서버 측 암호화)를 사용하려면 를 포함합니다 `x-amz-server-side-encryption` 다중 파트 업로드 시작 요청의 요청 헤더만 SSE-C(고객이 제공한 키와 함께 서버측 암호화)를 사용하려면 다중 파트 업로드 시작 요청 및 각 후속 업로드 파트 요청에서 동일한 세 가지 암호화 키 요청 헤더를 지정합니다.

작동	구축
다중 파트 업로드 나열	을 참조하십시오 "다중 파트 업로드 나열"
멀티파트 업로드를 시작합니다	을 참조하십시오 "멀티파트 업로드를 시작합니다"
부품 업로드	을 참조하십시오 "부품 업로드"
업로드 부품 - 복사	을 참조하십시오 "업로드 부품 - 복사"
멀티파트 업로드를 완료합니다	을 참조하십시오 "멀티파트 업로드를 완료합니다"
멀티파트 업로드를 중단합니다	모든 Amazon S3 REST API 동작으로 구현됩니다. 예고 없이 변경될 수 있습니다.
파트 목록	모든 Amazon S3 REST API 동작으로 구현됩니다. 예고 없이 변경될 수 있습니다.

관련 정보

- "일관성 제어"
- "서버측 암호화를 사용합니다"

다중 파트 업로드 나열

다중 파트 업로드 나열 작업은 버킷에 대해 진행 중인 다중 파트 업로드를 나열합니다.

지원되는 요청 매개 변수는 다음과 같습니다.

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

- `Authorization`

버전 관리

멀티파트 업로드는 업로드 시작, 리스팅 업로드, 파트 업로드, 업로드된 파트 조립 및 업로드 완료를 위한 별도의 작업으로 구성됩니다. 전체 다중 파트 업로드 작업이 수행되는 경우, 즉 개체가 작성되는 시점(해당되는 경우 버전)입니다.

멀티파트 업로드를 시작합니다

다중 파트 업로드 시작(CreateMultipartUpload) 작업은 개체에 대해 다중 파트 업로드를 시작하고 업로드 ID를 반환합니다.

를 클릭합니다 `x-amz-storage-class` 요청 헤더가 지원됩니다. 에 대해 제출된 값입니다 `x-amz-storage-class` ILM을 통해 결정되는 StorageGRID 시스템에 저장된 개체의 영구 복사본 수가 아닌 수집 중에 StorageGRID이 오브젝트 데이터를 보호하는 방법에 영향을 미칩니다.

인제스트 개체와 일치하는 ILM 규칙이 Ingest 동작에 대해 Strict 옵션을 사용하는 경우, 를 참조하십시오 `x-amz-storage-class` 머리글은 효과가 없습니다.

에 사용할 수 있는 값은 다음과 같습니다 `x-amz-storage-class`:

- `STANDARD` (기본값)
  - * 이중 커밋 *: ILM 규칙이 Ingest 동작에 대한 이중 커밋 옵션을 지정하는 경우, 개체가 수집되는 즉시 해당 개체의 두 번째 복사본이 생성되어 다른 스토리지 노드(이중 커밋)에 배포됩니다. ILM을 평가할 때 StorageGRID는 이러한 초기 중간 복사본이 규칙의 배치 지침을 충족하는지 여부를 결정합니다. 그렇지 않으면 새 오브젝트 복사본을 다른 위치에 만들어야 하고 초기 중간 복사본을 삭제해야 할 수 있습니다.
  - * 균형 *: ILM 규칙이 균형 옵션을 지정하고 StorageGRID이 규칙에 지정된 모든 복사본을 즉시 만들 수 없는 경우 StorageGRID은 다른 스토리지 노드에 두 개의 임시 복사본을 만듭니다.

    StorageGRID에서 ILM 규칙(동기식 배치)에 지정된 모든 오브젝트 복사본을 즉시 생성할 수 있는 경우 를 참조하십시오 `x-amz-storage-class` 머리글은 효과가 없습니다.
- `REDUCED_REDUNDANCY`
  - * 이중 커밋 *: ILM 규칙이 Ingest 동작에 대한 이중 커밋 옵션을 지정하는 경우 StorageGRID는 오브젝트가 수집될 때(단일 커밋) 단일 임시 복사본을 만듭니다.
  - * 균형 *: ILM 규칙이 균형 옵션을 지정하는 경우 StorageGRID은 시스템에서 규칙에 지정된 모든 사본을 즉시 만들 수 없는 경우에만 단일 중간 복사본을 만듭니다. StorageGRID에서 동기 배치를 수행할 수 있는 경우 이 머리글은 영향을 주지 않습니다. 를 클릭합니다 `REDUCED_REDUNDANCY` 옵션은 개체와 일치하는 ILM 규칙이 복제된 단일 복사본을 만들 때 가장 적합합니다. 이 경우 를 사용합니다 `REDUCED_REDUNDANCY` 모든 수집 작업에 대해 불필요한 오브젝트 복사본을 생성 및 삭제할 필요가 없습니다.

  를 사용합니다 `REDUCED_REDUNDANCY` 다른 상황에서는 옵션을 사용하지 않는 것이 좋습니다. `REDUCED_REDUNDANCY` 수집 중에 오브젝트 데이터가 손실될 위험이 증가합니다. 예를 들어, ILM 평가가 발생하기 전에 실패한 스토리지 노드에 단일 복사본이 처음 저장되는 경우 데이터가 손실될 수 있습니다.

복제된 복사본이 항상 하나만 있으면 데이터가 영구적으로 손실될 위험이 있습니다. 복제된 객체 복제본이 하나만 있는 경우 스토리지 노드에 장애가 발생하거나 심각한 오류가 발생한 경우 해당 객체가 손실됩니다. 또한 업그레이드와 같은 유지보수 절차 중에는 개체에 대한 액세스가 일시적으로 중단됩니다.

지정 `REDUCED_REDUNDANCY` 오브젝트를 처음 인제스트할 때 생성되는 복사본 수에만 영향을 줍니다. 활성 ILM 정책에 따라 개체를 평가할 때 개체의 복사본 수에 영향을 주지 않으며 StorageGRID 시스템에서 낮은 수준의 중복성에 데이터가 저장되지 않습니다.

S3 오브젝트 잠금이 설정된 버킷으로 오브젝트를 밀어넣는 경우, 를 참조하십시오 `REDUCED_REDUNDANCY` 옵션이 무시됩니다. 오브젝트를 레거시 준수 버킷으로 인스팅하는 경우, 를 참조하십시오 `REDUCED_REDUNDANCY` 옵션을 사용하면 오류가 반환됩니다. StorageGRID은 규정 준수 요구 사항이 충족될 수 있도록 항상 이중 커밋 수집을 수행합니다.

지원되는 요청 헤더는 다음과 같습니다.

- `Content-Type`
- `x-amz-meta-`사용자 정의 메타데이터를 포함하는 이름-값 쌍이 뒤에 옵니다

  사용자 정의 메타데이터에 대한 이름 값 쌍을 지정할 때 다음 일반 형식을 사용합니다.

  ```
  x-amz-meta-_name_: `value`
  ```

  ILM 규칙의 참조 시간으로 * 사용자 정의 생성 시간 * 옵션을 사용하려면 을 사용해야 합니다 `creation-time` 오브젝트를 만들 때 기록하는 메타데이터의 이름입니다. 예를 들면 다음과 같습니다.

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

  의 값 `creation-time` 1970년 1월 1일 이후 초 단위로 평가됩니다.

  추가 중 `creation-time` 레거시 규정 준수 기능이 설정된 버킷에 오브젝트를 추가할 경우 사용자 정의 메타데이터가 허용되지 않습니다. 오류가 반환됩니다.

- S3 오브젝트 잠금 요청 헤더:
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    이러한 헤더 없이 요청이 이루어지면 버킷 기본 보존 설정을 사용하여 개체 버전 보존 기간을 계산합니다.

  "S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"

- SSE 요청 헤더:
  - `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

서버측 암호화에 대한 요청 헤더

StorageGRID에서 UTF-8 문자를 처리하는 방법에 대한 자세한 내용은 Put Object 설명서를 참조하십시오.

서버측 암호화에 대한 요청 헤더

다음 요청 헤더를 사용하여 서버측 암호화를 사용하여 다중 파트 개체를 암호화할 수 있습니다. SSE 및 SSE-C 옵션은 상호 배타적입니다.

- * SSE *: StorageGRID에서 관리하는 고유 키로 객체를 암호화하려면 다중 파트 업로드 시작 요청에서 다음 헤더를 사용하십시오. 업로드 부품 요청에 이 헤더를 지정하지 마십시오.
  - x-amz-server-side-encryption
- * SSE-C *: 사용자가 제공 및 관리하는 고유 키를 사용하여 개체를 암호화하려는 경우 다중 파트 업로드 시작 요청(및 각 후속 업로드 파트 요청)에서 이 헤더 세 개를 모두 사용합니다.
  - x-amz-server-side-encryption-customer-algorithm`을 지정합니다 `AES256.
  - `x-amz-server-side-encryption-customer-key`새 오브젝트의 암호화 키를 지정합니다.
  - x-amz-server-side-encryption-customer-key-MD5: 새 개체의 암호화 키에 대한 MD5 다이제스트를 지정합니다.

제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다. 고객이 제공한 키를 사용하여 오브젝트 데이터를 보호하기 전에 의 고려 사항을 검토하십시오 "서버 측 암호화 사용".

지원되지 않는 요청 헤더입니다

다음 요청 헤더는 지원되지 않으며 반환됩니다 XNotImplemented

- x-amz-website-redirect-location

버전 관리

멀티파트 업로드는 업로드 시작, 리스팅 업로드, 파트 업로드, 업로드된 파트 조립 및 업로드 완료를 위한 별도의 작업으로 구성됩니다. MultiPart Upload 작업이 완료되면 개체가 만들어지고 버전이 적용됩니다(해당하는 경우).

관련 정보

"ILM을 사용하여 개체를 관리합니다"

"개체 를 넣습니다"

부품 업로드

파트 업로드 작업은 개체에 대해 여러 부분으로 업로드되는 파트를 업로드합니다.

지원되는 요청 헤더입니다

지원되는 요청 헤더는 다음과 같습니다.

- `Content-Length`
- `Content-MD5`

서버측 암호화에 대한 요청 헤더

다중 파트 업로드 시작 요청에 대해 SSE-C 암호화를 지정한 경우 각 업로드 파트 요청에 다음 요청 헤더를 포함해야 합니다.

- `x-amz-server-side-encryption-customer-algorithm`을 지정합니다 `AES256`.
- `x-amz-server-side-encryption-customer-key`[다중 파트 업로드 시작] 요청에서 제공한 암호화 키와 동일한 암호화 키를 지정합니다.
- `x-amz-server-side-encryption-customer-key-MD5`: 다중 파트 업로드 시작 요청에서 제공한 것과 동일한 MD5 다이제스트를 지정합니다.

> 제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다. 고객이 제공한 키를 사용하여 오브젝트 데이터를 보호하기 전에 "'서버측 암호화 사용'의 고려 사항을 검토하십시오.

버전 관리

멀티파트 업로드는 업로드 시작, 리스팅 업로드, 파트 업로드, 업로드된 파트 조립 및 업로드 완료를 위한 별도의 작업으로 구성됩니다. MultiPart Upload 작업이 완료되면 개체가 만들어지고 버전이 적용됩니다(해당하는 경우).

관련 정보

"서버측 암호화를 사용합니다"

업로드 부품 - 복사

파트 업로드 - 복사 작업은 기존 개체의 데이터를 데이터 소스로 복사하여 개체의 일부를 업로드합니다.

Part-Copy 업로드 작업은 모든 Amazon S3 REST API 동작으로 구현됩니다. 예고 없이 변경될 수 있습니다.

이 요청은 에 지정된 오브젝트 데이터를 읽고 씁니다 `x-amz-copy-source-range` StorageGRID 시스템 내에서

지원되는 요청 헤더는 다음과 같습니다.

- `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`

서버측 암호화에 대한 요청 헤더

다중 파트 업로드 시작 요청에 대해 SSE-C 암호화를 지정한 경우 각 업로드 파트 - 복사 요청에 다음 요청 헤더를 포함해야 합니다.

- `x-amz-server-side-encryption-customer-algorithm`을 지정합니다 `AES256`.
- `x-amz-server-side-encryption-customer-key`[다중 파트 업로드 시작] 요청에서 제공한 암호화 키와 동일한 암호화 키를 지정합니다.
- `x-amz-server-side-encryption-customer-key-MD5`: 다중 파트 업로드 시작 요청에서 제공한 것과 동일한 MD5 다이제스트를 지정합니다.

소스 객체가 SSE-C(customer-provided key)를 사용하여 암호화된 경우, 객체가 해독되고 복사될 수 있도록 업로드 파트 - 복사 요청에 다음 세 개의 헤더를 포함해야 합니다.

- `x-amz-copy-source-server-side-encryption-customer-algorithm`을 지정합니다 `AES256`.
- `x-amz-copy-source-server-side-encryption-customer-key`: 소스 객체를 만들 때 제공한 암호화 키를 지정합니다.
- `x-amz-copy-source-server-side-encryption-customer-key-MD5`: 소스 개체를 만들 때 제공한 MD5 다이제스트를 지정합니다.

> 제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다. 고객이 제공한 키를 사용하여 오브젝트 데이터를 보호하기 전에 "'서버측 암호화 사용'의 고려 사항을 검토하십시오.

버전 관리

멀티파트 업로드는 업로드 시작, 리스팅 업로드, 파트 업로드, 업로드된 파트 조립 및 업로드 완료를 위한 별도의 작업으로 구성됩니다. MultiPart Upload 작업이 완료되면 개체가 만들어지고 버전이 적용됩니다(해당하는 경우).

멀티파트 업로드를 완료합니다

전체 다중 파트 업로드 작업은 이전에 업로드한 파트를 조립하여 개체의 여러 부분 업로드를 완료합니다.

충돌 해결

동일한 키에 쓰는 두 클라이언트 등의 충돌하는 클라이언트 요청은 "최신 성공" 기준으로 해결됩니다. "Latest-WINS" 평가 시기는 S3 클라이언트가 작업을 시작할 때가 아니라 StorageGRID 시스템이 지정된 요청을 완료하는 시점을 기준으로 합니다.

요청 헤더

를 클릭합니다 `x-amz-storage-class` 요청 헤더가 지원되며 일치하는 ILM 규칙에서 이중 커밋 또는 균형 조정의 수집 동작을 지정하는 경우 StorageGRID에서 만드는 개체 복사본 수에 영향을 줍니다.

- `STANDARD`

  (기본값) ILM 규칙이 이중 커밋 옵션을 사용하거나 균형 옵션이 중간 복사본 생성으로 돌아올 때 이중 커밋 수집 작업을 지정합니다.

- `REDUCED_REDUNDANCY`

  ILM 규칙이 이중 커밋 옵션을 사용하거나 균형 옵션이 중간 복사본 생성으로 돌아올 때 단일 커밋 수집 작업을 지정합니다.

  > S3 오브젝트 잠금이 설정된 버킷으로 오브젝트를 밀어넣는 경우, 를 참조하십시오 `REDUCED_REDUNDANCY` 옵션이 무시됩니다. 오브젝트를 레거시 준수 버킷으로 인스팅하는 경우, 를 참조하십시오 `REDUCED_REDUNDANCY` 옵션을 사용하면 오류가 반환됩니다. StorageGRID은 규정 준수 요구 사항이 충족될 수 있도록 항상 이중 커밋 수집을 수행합니다.

> 15일 이내에 여러 부분 업로드가 완료되지 않으면 작업이 비활성으로 표시되고 모든 관련 데이터가 시스템에서 삭제됩니다.

> 를 클릭합니다 `ETag` 반환된 값은 MD5 합계가 아니라 의 Amazon S3 API 구현을 따릅니다 `ETag` 다중 파트 개체에 대한 값입니다.

버전 관리

이 작업은 여러 부분 업로드를 완료합니다. 버켓에 대해 버전 관리가 활성화된 경우 다중 파트 업로드가 완료된 후 개체 버전이 생성됩니다.

버켓에 대해 버전 관리가 활성화된 경우 고유한 가 사용됩니다 `versionId` 는 저장 중인 개체의 버전에 대해 자동으로 생성됩니다. 여기 `versionId` 를 사용하여 응답에서도 반환됩니다 `x-amz-version-id` 응답 헤더.

버전 관리가 일시 중단된 경우 개체 버전은 null로 저장됩니다 `versionId` null 버전이 이미 있는 경우 덮어쓰기가 됩니다.

> 버켓에 대해 버전 관리가 활성화된 경우, 같은 개체 키에서 동시 다중 파트 업로드가 완료된 경우에도 다중 파트 업로드를 완료하면 항상 새 버전이 생성됩니다. 버켓에 대해 버전 관리를 사용하지 않으면 다중 파트 업로드를 시작한 다음 다른 다중 파트 업로드를 시작하여 동일한 개체 키에서 먼저 완료할 수 있습니다. 비버전 버킷에서는 마지막으로 완료한 다중 파트 업로드가 우선 적용됩니다.

복제, 알림 또는 메타데이터 알림에 실패했습니다

플랫폼 서비스에 대해 다중 파트 업로드가 발생하는 버킷이 구성된 경우 연결된 복제 또는 알림 작업이 실패한 경우에도 다중 파트 업로드가 성공합니다.

이 경우 SMTT(Grid Manager on Total Events)에서 경보가 발생합니다. 마지막 이벤트 메시지는 알림이 실패한 마지막 객체에 대해 "버킷 이름 오브젝트 키에 대한 알림을 게시하지 못했습니다"라고 표시됩니다. (이 메시지를 보려면 * nodes * > **Storage Node* * > * Events * 를 선택합니다. 테이블 상단의 마지막 이벤트 보기) 이벤트 메시지는 에도 나열됩니다 `/var/local/log/bycast-err.log`.

테넌트는 개체의 메타데이터 또는 태그를 업데이트하여 실패한 복제 또는 알림을 트리거할 수 있습니다. 테넌트는 불필요한 변경을 방지하기 위해 기존 값을 다시 제출할 수 있습니다.

관련 정보

"ILM을 사용하여 개체를 관리합니다"

오류 응답

StorageGRID 시스템은 적용되는 모든 표준 S3 REST API 오류 응답을 지원합니다. 또한 StorageGRID 구현에는 여러 개의 사용자 지정 응답이 추가됩니다.

지원되는 **S3 API** 오류 코드입니다

이름	**HTTP** 상태입니다
액세스가 거부되었습니다	403 사용 금지
배다이제스트	400 잘못된 요청
BuckketAlreadyExists를 참조하십시오	409 충돌
BuckketNotEmpty	409 충돌
IncompleteBody	400 잘못된 요청
내부 오류입니다	500 내부 서버 오류입니다
InvalidAccessKeyId 입니다	403 사용 금지
InvalidArgument 를 선택합니다	400 잘못된 요청
InvalidBuckName입니다	400 잘못된 요청
InvalidBuckketState입니다	409 충돌
InvalidDigest 를 선택합니다	400 잘못된 요청
InvalidEncryptionAlgorithmError 가 발생합니다	400 잘못된 요청
InvalidPart 를 선택합니다	400 잘못된 요청
InvalidPartOrder를 선택합니다	400 잘못된 요청
InvalidRange 를 선택합니다	416 요청된 범위가 충분하지 않습니다
InvalidRequest 입니다	400 잘못된 요청
InvalidStorageClass 의 값을 반환합니다	400 잘못된 요청

이름	**HTTP** 상태입니다
InvalidTag 를 선택합니다	400 잘못된 요청
InvalidURI입니다	400 잘못된 요청
키투롱	400 잘못된 요청
MalformedXML을 참조하십시오	400 잘못된 요청
MetadataTooLarge를 참조하십시오	400 잘못된 요청
MethodNotAllowed 를 참조하십시오	405 메서드를 사용할 수 없습니다
MissingContentLength를 참조하십시오	411 길이 필요
MissingRequestBodyError가 발생합니다	400 잘못된 요청
MissingSecurityHeader 를 참조하십시오	400 잘못된 요청
NoSuchBucket	404를 찾을 수 없습니다
NoSuchKey를 클릭합니다	404를 찾을 수 없습니다
NoSuchUpload 를 클릭합니다	404를 찾을 수 없습니다
구현되지 않았습니다	501 구현되지 않음
NoSuchBuckketPolicy를 참조하십시오	404를 찾을 수 없습니다
ObjectLockConfigurationNotFoundError 가 발생합니다	404를 찾을 수 없습니다
사전 조건에 실패했습니다	412 전제 조건 실패
RequestTimeTooSkewed 를 참조하십시오	403 사용 금지
서비스를 사용할 수 없습니다	503 서비스를 사용할 수 없습니다
SignatureDoesNotMatch 를 참조하십시오	403 사용 금지
투만이버킷	400 잘못된 요청
UserKeyMustBeSpecified 를 선택합니다	400 잘못된 요청

StorageGRID 사용자 지정 오류 코드

이름	설명	HTTP 상태입니다
XBuckketLifecycleNotAllowed를 참조하십시오	버킷 수명 주기 구성은 레거시 준수 버킷에서 허용되지 않습니다	400 잘못된 요청
XBuckketPolicyParseException 을 참조하십시오	수신된 버킷 정책 JSON을 구문 분석하지 못했습니다.	400 잘못된 요청
XComplianceConflictt	레거시 준수 설정으로 인해 작업이 거부되었습니다.	403 사용 금지
XComplianceRedundancyForbidden을 선택합니다	레거시 준수 버킷에서는 감소된 중복성이 허용되지 않습니다	400 잘못된 요청
XMaxBuckketPolicyLengthExceeded 를 참조하십시오	정책이 허용되는 최대 버킷 정책 길이를 초과합니다.	400 잘못된 요청
XMissingInternalRequestHeader를 참조하십시오	내부 요청의 헤더가 누락되었습니다.	400 잘못된 요청
XNoSuchBuckketCompliance	지정된 버킷에 레거시 준법 기능이 설정되어 있지 않습니다.	404를 찾을 수 없습니다
XNotAcceptable(X 허용 가능)	요청에 충족되지 않은 하나 이상의 수락 헤더가 있습니다.	406 허용되지 않습니다
XNotImplemented(XNotImplemented)	제공한 요청은 구현되지 않은 기능을 의미합니다.	501 구현되지 않음

저작권 정보

Copyright © 2025 NetApp, Inc. All Rights Reserved. 미국에서 인쇄됨 본 문서의 어떠한 부분도 저작권 소유자의 사전 서면 승인 없이는 어떠한 형식이나 수단(복사, 녹음, 녹화 또는 전자 검색 시스템에 저장하는 것을 비롯한 그래픽, 전자적 또는 기계적 방법)으로도 복제될 수 없습니다.

NetApp이 저작권을 가진 자료에 있는 소프트웨어에는 아래의 라이센스와 고지사항이 적용됩니다.

본 소프트웨어는 NetApp에 의해 '있는 그대로' 제공되며 상품성 및 특정 목적에의 적합성에 대한 명시적 또는 묵시적 보증을 포함하여(이에 제한되지 않음) 어떠한 보증도 하지 않습니다. NetApp은 대체품 또는 대체 서비스의 조달, 사용 불능, 데이터 손실, 이익 손실, 영업 중단을 포함하여(이에 국한되지 않음), 이 소프트웨어의 사용으로 인해 발생하는 모든 직접 및 간접 손해, 우발적 손해, 특별 손해, 징벌적 손해, 결과적 손해의 발생에 대하여 그 발생 이유, 책임론, 계약 여부, 엄격한 책임, 불법 행위(과실 또는 그렇지 않은 경우)와 관계없이 어떠한 책임도 지지 않으며, 이와 같은 손실의 발생 가능성이 통지되었다 하더라도 마찬가지입니다.

NetApp은 본 문서에 설명된 제품을 언제든지 예고 없이 변경할 권리를 보유합니다. NetApp은 NetApp의 명시적인 서면 동의를 받은 경우를 제외하고 본 문서에 설명된 제품을 사용하여 발생하는 어떠한 문제에도 책임을 지지 않습니다. 본 제품의 사용 또는 구매의 경우 NetApp에서는 어떠한 특허권, 상표권 또는 기타 지적 재산권이 적용되는 라이센스도 제공하지 않습니다.

본 설명서에 설명된 제품은 하나 이상의 미국 특허, 해외 특허 또는 출원 중인 특허로 보호됩니다.

제한적 권리 표시: 정부에 의한 사용, 복제 또는 공개에는 DFARS 252.227-7013(2014년 2월) 및 FAR 52.227-19(2007년 12월)의 기술 데이터-비상업적 품목에 대한 권리(Rights in Technical Data -Noncommercial Items) 조항의 하위 조항 (b)(3)에 설명된 제한사항이 적용됩니다.

여기에 포함된 데이터는 상업용 제품 및/또는 상업용 서비스(FAR 2.101에 정의)에 해당하며 NetApp, Inc.의 독점 자산입니다. 본 계약에 따라 제공되는 모든 NetApp 기술 데이터 및 컴퓨터 소프트웨어는 본질적으로 상업용이며 개인 비용만으로 개발되었습니다. 미국 정부는 데이터가 제공된 미국 계약과 관련하여 해당 계약을 지원하는 데에만 데이터에 대한 전 세계적으로 비독점적이고 양도할 수 없으며 재사용이 불가능하며 취소 불가능한 라이센스를 제한적으로 가집니다. 여기에 제공된 경우를 제외하고 NetApp, Inc.의 사전 서면 승인 없이는 이 데이터를 사용, 공개, 재생산, 수정, 수행 또는 표시할 수 없습니다. 미국 국방부에 대한 정부 라이센스는 DFARS 조항 252.227-7015(b)(2014년 2월)에 명시된 권한으로 제한됩니다.

상표 정보

NETAPP, NETAPP 로고 및 http://www.netapp.com/TM에 나열된 마크는 NetApp, Inc.의 상표입니다. 기타 회사 및 제품 이름은 해당 소유자의 상표일 수 있습니다.