



객체에 대한 연산 StorageGRID software

NetApp
July 23, 2026

목차

객체에 대한 연산	1
StorageGRID에서 객체에 대한 S3 REST API 작업을 구현하는 방법	1
StorageGRID에서 지원되는 Amazon S3 Select 작업	4
조항	5
데이터 유형	5
운영자	5
집계 함수	6
조건 함수	6
변환 함수	6
날짜 함수	6
문자열 함수	7
StorageGRID에서 서버 측 암호화를 사용하는 방법	7
\${post_edited_translations.segment}	7
SSE-C 사용	8
\${post_edited_translations.segment}	8
S3 CopyObject 요청을 사용하여 StorageGRID에 객체의 복사본을 생성합니다	9
충돌 해결	9
객체 크기	9
사용자 메타데이터의 UTF-8 문자	9
지원되는 요청 헤더	10
지원되지 않는 요청 헤더	11
스토리지 클래스 옵션	11
CopyObject에서 x-amz-copy-source 사용	12
서버 측 암호화에 대한 요청 헤더	12
버전 관리	13
GetObject 요청을 사용하여 StorageGRID의 버킷에서 객체 검색	13
GetObject 및 멀티파트 객체	13
사용자 메타데이터의 UTF-8 문자	13
지원되는 요청 헤더	13
지원되지 않는 요청 헤더	14
버전 관리	14
고객이 제공한 암호화 키를 사용한 서버 측 암호화(SSE-C) 요청 헤더	14
클라우드 스토리지 풀 객체에 대한 GetObject의 동작	14
GetObject 및 그리드 간 복제	15
S3 HeadObject 요청을 사용하여 StorageGRID의 객체에서 메타데이터를 검색합니다	16
HeadObject 및 멀티파트 객체	16
사용자 메타데이터의 UTF-8 문자	16
지원되는 요청 헤더	16
지원되지 않는 요청 헤더	16

버전 관리	16
고객이 제공한 암호화 키를 사용한 서버 측 암호화(SSE-C) 요청 헤더	17
HeadObject 클라우드 스토리지 풀 객체에 대한 응답	17
HeadObject 및 그리드 간 복제	19
S3 PutObject 요청을 사용하여 StorageGRID의 버킷에 객체 추가	19
충돌 해결	19
객체 크기	19
사용자 메타데이터 크기	19
사용자 메타데이터의 UTF-8 문자	20
객체 태그 제한	20
객체 소유권	20
지원되는 요청 헤더	20
지원되지 않는 요청 헤더	21
스토리지 클래스 옵션	22
서버 측 암호화에 대한 요청 헤더	23
버전 관리	24
Authorization 헤더에 대한 서명 계산	24
S3 RestoreObject 요청을 사용하여 StorageGRID에서 객체 복원	24
지원되는 요청 유형	24
버전 관리	24
클라우드 스토리지 풀 객체에 대한 RestoreObject의 동작	24
S3 SelectObjectContent 요청을 사용하여 StorageGRID에서 객체 데이터 필터링	25
CSV 요청 구문 예시	26
Parquet 요청 구문 예시	27
SQL 쿼리 예시	28
AWS-CLI 사용 예시 (CSV)	29
AWS-CLI 사용 예시 (Parquet)	30

객체에 대한 연산

StorageGRID에서 객체에 대한 S3 REST API 작업을 구현하는 방법

이 섹션에서는 StorageGRID 시스템이 객체에 대한 S3 REST API 작업을 구현하는 방법을 설명합니다.

다음 조건은 모든 객체 작업에 적용됩니다.

- StorageGRID "**일관성 값**"은(는) 다음을 제외한 모든 객체 작업에서 지원됩니다.
 - GetObjectAcl
 - OPTIONS /
 - PutObjectLegalHold
 - PutObjectRetention
 - SelectObjectContent
- 두 클라이언트가 동일한 키에 쓰기를 시도하는 등 클라이언트 요청이 충돌하는 경우 "최신 요청 우선" 원칙에 따라 처리됩니다. "최신 요청 우선" 평가 시점은 S3 클라이언트가 작업을 시작한 시점이 아니라 StorageGRID 시스템이 해당 요청을 완료한 시점을 기준으로 합니다.
- StorageGRID 버킷의 모든 객체는 버킷 소유자의 소유이며, 여기에는 익명 사용자 또는 다른 계정이 생성한 객체도 포함됩니다.

다음 표는 StorageGRID S3 REST API 객체 작업을 구현하는 방식을 설명합니다.

작업	구현
DeleteObject	DeleteObject 요청을 처리할 때 StorageGRID는 모든 저장 위치에서 해당 객체의 모든 복사본을 즉시 제거하려고 시도합니다. 제거에 성공하면 StorageGRID는 클라이언트에 즉시 응답을 반환합니다. 모든 복사본을 30초 이내에 제거할 수 없는 경우(예: 특정 위치를 일시적으로 사용할 수 없는 경우) StorageGRID는 제거를 위해 복사본을 대기열에 추가한 후 클라이언트에 성공을 알립니다. 제한 사항 - 다중 요소 인증(MFA) 및 응답 헤더 `x-amz-mfa`는 지원되지 않습니다. - If-None 및 If-None-Match 헤더는 허용되지만 기능하지 않습니다. 버전 관리 특정 버전을 삭제하려면 요청자가 버킷 소유자여야 하며 versionId 하위 리소스를 사용해야 합니다. 이 하위 리소스를 사용하면 해당 버전이 영구적으로 삭제됩니다. `versionId`가 삭제 마커에 해당하는 경우 응답 헤더 `x-amz-delete-marker`가 `true`로 설정되어 반환됩니다. - 버전 관리가 활성화된 버킷에서 versionId 하위 리소스 없이 객체를 삭제하면 삭제 마커가 생성됩니다. 삭제 마커에 대한 versionId는 `x-amz-version-id` 응답 헤더를 사용하여 반환되며, x-amz-delete-marker 응답 헤더는 `true`로 설정되어 반환됩니다. - 버전 관리가 일시 중단된 버킷에서 versionId 하위 리소스 없이 객체를 삭제하면 이미 존재하는 'null' 버전 또는 'null' 삭제 마커가 영구적으로 삭제되고 새로운 'null' 삭제 마커가 생성됩니다. x-amz-delete-marker 응답 헤더는 `true`로 설정되어 반환됩니다. 참고: 경우에 따라 하나의 객체에 여러 개의 삭제 표시가 존재할 수 있습니다. "S3 REST API를 사용하여 S3 Object Lock 구성"을 참조하여 GOVERNANCE 모드에서 객체 버전을 삭제하는 방법을 알아보십시오.
DeleteObjects (이전 명칭: DELETE Multiple Objects)	다중 요소 인증(MFA) 및 응답 헤더 `x-amz-mfa`는 지원되지 않습니다. 하나의 요청 메시지에서 여러 객체를 삭제할 수 있습니다. "S3 REST API를 사용하여 S3 Object Lock 구성"을 참조하여 GOVERNANCE 모드에서 객체 버전을 삭제하는 방법을 알아보십시오.

작업	구현
DeleteObjectTagging	`tagging` 하위 리소스를 사용하여 객체에서 모든 태그를 제거합니다. 버전 관리 요청에 <code>versionId</code> 쿼리 매개변수가 지정되지 않은 경우, 해당 작업은 버전 관리 버킷에서 객체의 최신 버전에 있는 모든 태그를 삭제합니다. 현재 객체 버전이 삭제 마커인 경우, "MethodNotAllowed" 상태가 <code>x-amz-delete-marker</code> 응답 헤더가 <code>true</code> 로 설정된 상태로 반환됩니다.
GetObject	"GetObject"
GetObjectAcl	계정에 필요한 액세스 자격 증명이 제공되면 작업은 긍정적인 응답과 함께 개체 소유자의 ID, DisplayName 및 권한을 반환하여 소유자가 해당 개체에 대한 모든 액세스 권한을 가지고 있음을 나타냅니다.
GetObjectLegalHold	"S3 REST API를 사용하여 S3 Object Lock 구성"
GetObjectRetention	"S3 REST API를 사용하여 S3 Object Lock 구성"
GetObjectTagging	`tagging` 하위 리소스를 사용하여 객체의 모든 태그를 반환합니다. 버전 관리 요청에 <code>versionId</code> 쿼리 매개변수가 지정되지 않은 경우, 해당 작업은 버전 관리 버킷에서 객체의 최신 버전의 모든 태그를 반환합니다. 현재 객체 버전이 삭제 마커인 경우, <code>x-amz-delete-marker</code> 응답 헤더가 <code>true</code> 로 설정된 "MethodNotAllowed" 상태가 반환됩니다.
HeadObject	"HeadObject"
RestoreObject	"RestoreObject"
PutObject	"PutObject"
CopyObject (이전 명칭: PUT Object - Copy)	"CopyObject"
PutObjectLegalHold	"S3 REST API를 사용하여 S3 Object Lock 구성"
PutObjectRetention	"S3 REST API를 사용하여 S3 Object Lock 구성"

작업	구현
PutObjectTagging	<code>`tagging`</code> 하위 리소스를 사용하여 기존 객체에 태그 세트를 추가합니다. 객체 태그 제한 새 객체를 업로드할 때 태그를 추가하거나 기존 객체에 태그를 추가할 수 있습니다. StorageGRID와 Amazon S3 모두 객체당 최대 10개의 태그를 지원합니다. 객체와 연결된 태그는 고유한 태그 키를 가져야 합니다. 태그 키는 최대 128자의 유니코드 문자, 태그 값은 최대 256자의 유니코드 문자까지 사용할 수 있습니다. 키와 값은 대소문자를 구분합니다. 태그 업데이트 및 수집 동작 PutObjectTagging을 사용하여 객체의 태그를 업데이트하면 StorageGRID는 해당 객체를 다시 수집하지 않습니다. 즉, 일치하는 ILM 규칙에 지정된 수집 동작 옵션이 사용되지 않습니다. 업데이트로 인해 발생하는 객체 배치 변경 사항은 일반적인 백그라운드 ILM 프로세스에 의해 ILM이 재평가될 때 적용됩니다. 즉, ILM 규칙에서 수집 동작에 대해 엄격(Strict) 옵션을 사용하는 경우, 필요한 객체 배치가 불가능한 경우(예: 새로 필요한 위치를 사용할 수 없는 경우) 아무런 조치도 취해지지 않습니다. 업데이트된 객체는 필요한 배치가 가능해질 때까지 현재 위치를 유지합니다. 충돌 해결 두 클라이언트가 동일한 키에 쓰기를 시도하는 등 클라이언트 요청이 충돌하는 경우 "최신 요청 우선" 원칙에 따라 처리됩니다. "최신 요청 우선" 평가 시점은 S3 클라이언트가 작업을 시작한 시점이 아니라 StorageGRID 시스템이 해당 요청을 완료한 시점을 기준으로 합니다. 버전 관리 요청에 <code>versionId</code> 쿼리 매개변수가 지정되지 않은 경우, 해당 작업은 버전 관리 버킷에서 객체의 최신 버전에 태그를 추가합니다. 객체의 현재 버전이 삭제 마커인 경우, "MethodNotAllowed" 상태가 반환되며 <code>x-amz-delete-marker</code> 응답 헤더가 <code>`true`</code>로 설정됩니다.
SelectObjectContent	"SelectObjectContent"

StorageGRID에서 지원되는 Amazon S3 Select 작업

StorageGRID는 "[SelectObjectContent 명령](#)"에 대해 다음과 같은 Amazon S3 Select 절, 데이터 유형 및 연산자를 지원합니다.



목록에 없는 항목은 지원되지 않습니다.

구문에 대해서는 "[SelectObjectContent](#)"을 참조하십시오. S3 Select에 대한 자세한 내용은 "[AWS S3 Select 설명서](#)"을 참조하십시오.

S3 Select가 활성화된 테넌트 계정만 SelectObjectContent 쿼리를 실행할 수 있습니다. 자세한 내용은 "[S3 Select 사용을 위한 고려 사항 및 요구 사항](#)"을 참조하십시오.

조항

- SELECT 목록
- FROM 절
- WHERE 절
- LIMIT 절

데이터 유형

- 불
- 정수
- 문자열
- float
- 십진수, 숫자
- 타임스탬프

운영자

논리 연산자

- AND
- NOT
- 또는

비교 연산자

- <
- >
- <=
- >=
- =
- =
- <>
- !=
- 사이
- IN

패턴 매칭 연산자

- LIKE
- _
- %

단일 연산자

- IS NULL
- NULL이 아닙니다

수학 연산자

- +
- -
- *
- /
- %

StorageGRID는 Amazon S3 Select 연산자 우선순위를 따릅니다.

집계 함수

- AVG()
- COUNT(*)
- MAX()
- MIN()
- SUM()

조건 함수

- 사례
- COALESCE
- NULLIF

변환 함수

- CAST(지원되는 데이터 형식의 경우)

날짜 함수

- DATE_ADD
- DATE_DIFF
- 발취

- TO_STRING
- TO_TIMESTAMP
- UTCNOW

문자열 함수

- `${post_edited_translations.segment}`
- `${post_edited_translations.segment}`
- `${post_edited_translations.segment}`
- TRIM
- `${post_edited_translations.segment}`

StorageGRID에서 서버 측 암호화를 사용하는 방법

서버 측 암호화를 사용하면 저장된 객체 데이터를 보호할 수 있습니다. StorageGRID는 객체에 데이터를 기록할 때 데이터를 암호화하고, 객체에 접근할 때 데이터를 복호화합니다.

`${post_edited_translations.segment}`

- **SSE(StorageGRID에서 관리하는 키를 사용한 서버 측 암호화)**: S3 요청을 통해 객체를 저장하면 StorageGRID가 고유한 키로 객체를 암호화합니다. S3 요청을 통해 객체를 검색하면 StorageGRID가 저장된 키를 사용하여 객체를 복호화합니다.
- **SSE-C(고객 제공 키를 사용한 서버 측 암호화)**: 객체를 저장하기 위해 S3 요청을 보낼 때 자체 암호화 키를 제공합니다. 객체를 검색할 때 요청의 일부로 동일한 암호화 키를 제공합니다. 두 암호화 키가 일치하면 객체가 복호화되고 객체 데이터가 반환됩니다.

`${post_edited_translations.segment}`



사용자가 제공하는 암호화 키는 절대 저장되지 않습니다. 암호화 키를 분실하면 해당 객체도 함께 분실하게 됩니다.



객체가 SSE 또는 SSE-C로 암호화된 경우 버킷 수준 또는 그리드 수준의 암호화 설정은 무시됩니다.

`${post_edited_translations.segment}`

`${post_edited_translations.segment}`

`x-amz-server-side-encryption`

SSE 요청 헤더는 다음 객체 작업에서 지원됩니다.

- "PutObject"
- "CopyObject"
- "CreateMultipartUpload"

SSE-C 사용

`${post_edited_translations.segment}`

요청 헤더	설명
<code>x-amz-server-side-encryption-customer-algorithm</code>	암호화 알고리즘을 지정합니다. 헤더 값은 'AES256'이어야 합니다.
<code>x-amz-server-side-encryption-customer-key</code>	오브젝트를 암호화하거나 복호화하는 데 사용할 암호화 키를 지정합니다. 키 값은 base64로 인코딩된 256비트여야 합니다.
<code>x-amz-server-side-encryption-customer-key-MD5</code>	RFC 1321에 따라 암호화 키가 오류 없이 전송되었는지 확인하는 데 사용되는 암호화 키의 MD5 다이제스트를 지정합니다. MD5 다이제스트 값은 base64로 인코딩된 128비트여야 합니다.

`${post_edited_translations.segment}`

- `"GetObject"`
- `"HeadObject"`
- `"PutObject"`
- `"CopyObject"`
- `"CreateMultipartUpload"`
- `"UploadPart"`
- `"UploadPartCopy"`

`${post_edited_translations.segment}`

SSE-C를 사용하기 전에 다음 사항을 고려하십시오.

- https를 사용해야 합니다.



StorageGRID는 SSE-C를 사용할 때 http를 통해 전송된 모든 요청을 거부합니다. 보안상 이유로, 실수로 http를 통해 전송한 모든 키는 유출된 것으로 간주해야 합니다. 해당 키를 폐기하고 적절히 교체하십시오.

- `${post_edited_translations.segment}`
- 암호화 키와 오브젝트의 매핑을 관리해야 합니다. StorageGRID는 암호화 키를 저장하지 않습니다. 각 오브젝트에 대해 제공하는 암호화 키를 추적할 책임은 사용자에게 있습니다.
- 버킷의 버전 관리가 활성화된 경우, 각 오브젝트 버전에는 자체 암호화 키가 있어야 합니다. 각 오브젝트 버전에 사용된 암호화 키를 추적할 책임은 사용자에게 있습니다.
- 암호화 키를 클라이언트 측에서 관리하므로 키 순환과 같은 추가적인 보안 조치도 클라이언트 측에서 관리해야 합니다.



사용자가 제공하는 암호화 키는 절대 저장되지 않습니다. 암호화 키를 분실하면 해당 객체도 함께 분실하게 됩니다.

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

관련 정보

["Amazon S3 사용자 가이드: 고객 제공 키를 사용한 서버 측 암호화\(SSE-C\) 사용"](#)

S3 CopyObject 요청을 사용하여 StorageGRID에 객체의 복사본을 생성합니다

S3 CopyObject 요청을 사용하면 S3에 이미 저장된 객체의 복사본을 만들 수 있습니다. CopyObject 작업은 GetObject 다음에 PutObject를 수행하는 것과 같습니다.

충돌 해결

두 클라이언트가 동일한 키에 쓰기를 시도하는 등 클라이언트 요청이 충돌하는 경우 "최신 요청 우선" 원칙에 따라 처리됩니다. "최신 요청 우선" 평가 시점은 S3 클라이언트가 작업을 시작한 시점이 아니라 StorageGRID 시스템이 해당 요청을 완료한 시점을 기준으로 합니다.

객체 크기

단일 PutObject 작업에 권장되는 최대 크기는 5GiB(5,368,709,120바이트)입니다. 객체 크기가 5GiB보다 큰 경우, ["멀티파트 업로드"](#)를 사용하십시오.

단일 PutObject 작업에 대해 지원되는 최대 크기는 5TiB(5,497,558,138,880바이트)입니다.



StorageGRID 11.6 이하 버전에서 업그레이드한 경우, 5GiB를 초과하는 객체를 업로드하려고 하면 S3 PUT Object size too large 경고가 트리거됩니다. StorageGRID 11.7 또는 11.8을 새로 설치한 경우에는 이 경우 경고가 트리거되지 않습니다. 하지만 AWS S3 표준을 준수하기 위해 향후 StorageGRID 릴리스에서는 5GiB보다 큰 객체 업로드를 지원하지 않을 예정입니다.

사용자 메타데이터의 UTF-8 문자

요청에 사용자 정의 메타데이터의 키 이름 또는 값에 (이스케이프되지 않은) UTF-8 값이 포함된 경우 StorageGRID 동작은 정의되지 않습니다.

StorageGRID는 사용자 정의 메타데이터의 키 이름 또는 값에 포함된 이스케이프 처리된 UTF-8 문자를 구문 분석하거나 해석하지 않습니다. 이스케이프 처리된 UTF-8 문자는 ASCII 문자로 처리됩니다.

- 사용자 정의 메타데이터에 이스케이프 처리된 UTF-8 문자가 포함되어 있으면 요청이 성공합니다.
- StorageGRID는 키 이름이나 값의 해석된 값에 인쇄할 수 없는 문자가 포함된 경우 x-amz-missing-meta 헤더를 반환하지 않습니다.

지원되는 요청 헤더

다음과 같은 요청 헤더가 지원됩니다.

- 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-, 그 뒤에 사용자 정의 메타데이터를 포함하는 이름-값 쌍이 옵니다
- x-amz-metadata-directive: 기본값은 `COPY`이며, 이를 사용하면 객체와 관련 메타데이터를 복사할 수 있습니다.

객체를 복사할 때 기존 메타데이터를 덮어쓰거나 객체 메타데이터를 업데이트하도록 `REPLACE`를 지정할 수 있습니다.

- x-amz-storage-class
- x-amz-tagging-directive: 기본값은 `COPY`이며, 이 값을 사용하면 객체와 모든 태그를 복사할 수 있습니다.

객체를 복사할 때 기존 태그를 덮어쓸지, 아니면 태그를 업데이트할지 지정할 수 있습니다 REPLACE.

• S3 객체 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 버킷의 기본 보존 설정이 객체 버전 모드와 보존 만료일을 계산하는 데 사용됩니다. ["S3 REST API를 사용하여 S3 Object Lock 구성"](#)을 참조하십시오.

• 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

[서버 측 암호화에 대한 요청 헤더](#)을 참조하십시오

지원되지 않는 요청 헤더

다음 요청 헤더는 지원되지 않습니다.

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

`If-Match header`은 (는) 승인되었지만 작동하지 않습니다.

- If-None-Match

`If-None-Match header`은 (는) 승인되었지만 작동하지 않습니다.

- x-amz-checksum-algorithm

객체를 복사할 때 원본 객체에 체크섬이 있는 경우 StorageGRID는 해당 체크섬 값을 새 객체에 복사하지 않습니다. 이 동작은 객체 요청에서 `x-amz-checksum-algorithm`을(를) 사용하려고 시도하는지 여부와 관계없이 적용됩니다.

- x-amz-website-redirect-location

스토리지 클래스 옵션

`x-amz-storage-class` 요청 헤더는 지원되며, 일치하는 ILM 규칙에서 이중 커밋 또는 균형 xref:{relative_path}../ilm/data-protection-options-for-ingest.html["수집 옵션"]을 사용하는 경우 StorageGRID가 생성하는 객체 복사본 수에 영향을 미칩니다.

- STANDARD

(기본값) ILM 규칙에서 이중 커밋 옵션을 사용하거나 균형 옵션이 임시 복사본 생성으로 대체될 때 이중 커밋 수집 작업을 지정합니다.

- REDUCED_REDUNDANCY

ILM 규칙에서 이중 커밋 옵션을 사용하거나 균형 옵션이 임시 복사본 생성으로 대체될 때 단일 커밋 수집 작업을 지정합니다.



S3 객체 잠금이 활성화된 버킷에 객체를 수집하는 경우 REDUCED_REDUNDANCY 옵션은 무시됩니다. 레거시 규정 준수 버킷에 객체를 수집하는 경우 REDUCED_REDUNDANCY 옵션은 오류를 반환합니다. StorageGRID는 규정 준수 요구 사항을 충족하기 위해 항상 이중 커밋 수집을 수행합니다.

CopyObject에서 x-amz-copy-source 사용

`x-amz-copy-source` 헤더에 지정된 소스 버킷 및 키가 대상 버킷 및 키와 다른 경우, 소스 객체 데이터의 복사본이 대상에 기록됩니다.

소스와 대상이 일치하고 `x-amz-metadata-directive` 헤더가 `REPLACE`로 지정된 경우, 객체의 메타데이터는 요청에 제공된 메타데이터 값으로 업데이트됩니다. 이 경우 StorageGRID는 객체를 다시 수집하지 않습니다. 이는 두 가지 중요한 결과를 가져옵니다.

- CopyObject를 사용하여 기존 객체를 제자리에서 암호화하거나 기존 객체의 암호화를 제자리에서 변경할 수 없습니다. x-amz-server-side-encryption 헤더 또는 x-amz-server-side-encryption-customer-algorithm 헤더를 제공하면 StorageGRID가 요청을 거부하고 `XNotImplemented`을 반환합니다.
- 일치하는 ILM 규칙에 지정된 수집 동작 옵션은 사용되지 않습니다. 업데이트로 인해 발생하는 객체 배치 변경 사항은 일반적인 백그라운드 ILM 프로세스에 의해 ILM이 재평가될 때 적용됩니다.

즉, ILM 규칙에서 수집 동작에 대해 엄격(Strict) 옵션을 사용하는 경우, 필요한 객체 배치가 불가능한 경우(예: 새로 필요한 위치를 사용할 수 없는 경우) 아무런 조치도 취해지지 않습니다. 업데이트된 객체는 필요한 배치가 가능해질 때까지 현재 위치를 유지합니다.

서버 측 암호화에 대한 요청 헤더

"[서버 측 암호화 사용](#)"을 수행하는 경우, 제공하는 요청 헤더는 소스 객체가 암호화되었는지 여부와 대상 객체를 암호화할 계획인지 여부에 따라 달라집니다.

- 원본 객체가 고객 제공 키(SSE-C)를 사용하여 암호화된 경우, 객체를 복호화한 후 복사할 수 있도록 CopyObject 요청에 다음 세 가지 헤더를 포함해야 합니다.
 - x-amz-copy-source-server-side-encryption-customer-algorithm: `AES256`을(를) 지정합니다.
 - x-amz-copy-source-server-side-encryption-customer-key: 소스 객체를 생성할 때 제공한 암호화 키를 지정하십시오.
 - x-amz-copy-source-server-side-encryption-customer-key-MD5: 소스 객체를 생성할 때 제공한 MD5 다이제스트를 지정하십시오.
- 대상 객체(복사본)를 사용자가 제공하고 관리하는 고유 키로 암호화하려면 다음 세 가지 헤더를 포함하십시오.
 - x-amz-server-side-encryption-customer-algorithm: `AES256`을(를) 지정합니다.
 - x-amz-server-side-encryption-customer-key: 대상 객체에 사용할 새 암호화 키를 지정합니다.
 - x-amz-server-side-encryption-customer-key-MD5: 새 암호화 키의 MD5 다이제스트를 지정합니다.



고객이 제공하는 암호화 키는 절대 저장되지 않습니다. 암호화 키를 분실하면 해당 객체도 손실됩니다. 고객이 제공한 키를 사용하여 객체 데이터를 보호하기 전에 "[서버 측 암호화 사용](#)"에 대한 고려 사항을 검토하십시오.

- StorageGRID(SSE)에서 관리하는 고유 키로 대상 객체(복사본)를 암호화하려면 CopyObject 요청에 다음 헤더를 포함하십시오.

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



The `server-side-encryption` 객체의 값은 업데이트할 수 없습니다. 대신, 새 `server-side-encryption` 값으로 복사본을 만들려면 `x-amz-metadata-directive: 'REPLACE'`을 사용하세요.

버전 관리

원본 버킷에 버전 관리 기능이 있는 경우 `'x-amz-copy-source'` 헤더를 사용하여 객체의 최신 버전을 복사할 수 있습니다. 특정 버전의 객체를 복사하려면 `'versionId'` 하위 리소스를 사용하여 복사할 버전을 명시적으로 지정해야 합니다. 대상 버킷에 버전 관리 기능이 있는 경우 생성된 버전이 `'x-amz-version-id'` 응답 헤더에 반환됩니다. 대상 버킷의 버전 관리가 일시 중단된 경우 `'x-amz-version-id' "null"` 값이 반환됩니다.

GetObject 요청을 사용하여 StorageGRID의 버킷에서 객체 검색

S3 GetObject 요청을 사용하여 S3 버킷에서 객체를 검색할 수 있습니다.

GetObject 및 멀티파트 객체

You can use the `partNumber` 요청 매개변수를 사용하여 여러 부분으로 구성된 객체 또는 분할된 객체의 특정 부분을 검색할 수 있습니다. `'x-amz-mp-parts-count'` 응답 요소는 객체가 몇 부분으로 구성되어 있는지를 나타냅니다.

`'partNumber'`을 분할/멀티파트 객체와 비분할/비멀티파트 객체 모두에 대해 1로 설정할 수 있지만, `'x-amz-mp-parts-count'` 응답 요소는 분할 또는 멀티파트 객체에 대해서만 반환됩니다.

사용자 메타데이터의 UTF-8 문자

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

지원되는 요청 헤더

다음 요청 헤더가 지원됩니다.

- `x-amz-checksum-mode`: 지정 ENABLED

`Range` 헤더는 `GetObject`에 대해 ``x-amz-checksum-mode``와 (과) 함께 지원되지 않습니다. 요청에 ``Range``을 (를) 포함하고 ``x-amz-checksum-mode``이 (가) 활성화된 경우, `StorageGRID`는 응답에서 체크섬 값을 반환하지 않습니다.

지원되지 않는 요청 헤더

다음 요청 헤더는 지원되지 않으며 ``XNotImplemented``을(를) 반환합니다.

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

버전 관리

If a `versionId` 하위 리소스가 지정되지 않은 경우, 해당 작업은 버전이 지정된 버킷에서 객체의 최신 버전을 가져옵니다. 현재 객체 버전이 삭제 마커인 경우, "찾을 수 없음" 상태가 반환되며 `x-amz-delete-marker` 응답 헤더가 ``true``로 설정됩니다.

고객이 제공한 암호화 키를 사용한 서버 측 암호화(SSE-C) 요청 헤더

제공된 고유 키로 객체가 암호화된 경우 세 가지 헤더를 모두 사용하십시오.

- `x-amz-server-side-encryption-customer-algorithm: `AES256``을(를) 지정합니다.
- `x-amz-server-side-encryption-customer-key`: 객체에 사용할 암호화 키를 지정하십시오.
- `x-amz-server-side-encryption-customer-key-MD5`: 객체의 암호화 키의 MD5 다이제스트를 지정합니다.



고객이 제공하는 암호화 키는 절대 저장되지 않습니다. 암호화 키를 분실하면 해당 객체도 손실됩니다. 고객이 제공한 키를 사용하여 객체 데이터를 보호하기 전에 "[서버 측 암호화 사용](#)"의 고려 사항을 검토하십시오.

클라우드 스토리지 풀 객체에 대한 `GetObject`의 동작

객체가 "[클라우드 스토리지 풀](#)"에 저장된 경우, `GetObject` 요청의 동작은 객체의 상태에 따라 달라집니다. 자세한 내용은 "[HeadObject](#)"을 참조하십시오.



객체가 클라우드 스토리지 풀에 저장되어 있고 해당 객체의 복사본이 그리드에도 하나 이상 존재하는 경우, `GetObject` 요청은 클라우드 스토리지 풀에서 데이터를 가져오기 전에 그리드에서 데이터를 가져오려고 시도합니다.

객체의 상태	<code>GetObject</code> 의 동작
<code>StorageGRID</code> 에 수집되었지만 ILM에서 아직 평가되지 않은 객체이거나, 기존 스토리지 풀에 저장되었거나 이레이저 코딩을 사용하여 저장된 객체입니다.	200 OK 객체의 복사본이 검색됩니다.

객체의 상태	GetObject의 동작
클라우드 스토리지 풀에 있는 객체이지만 아직 검색 불가능한 상태로 전환되지 않았습니다.	200 OK 객체의 복사본이 검색됩니다.
객체가 복구 불가능한 상태로 전환되었습니다.	403 Forbidden, InvalidObjectState "RestoreObject" 요청을 사용하여 객체를 검색 가능한 상태로 복원하십시오.
복구 불가능한 상태에서 복원 중인 객체	403 Forbidden, InvalidObjectState RestoreObject 요청이 완료될 때까지 기다립니다.
객체가 클라우드 스토리지 풀에 완전히 복원되었습니다.	200 OK 객체의 복사본이 검색됩니다.

클라우드 스토리지 풀의 멀티파트 또는 분할된 객체

여러 부분으로 나뉜 객체를 업로드했거나 StorageGRID가 큰 객체를 여러 세그먼트로 분할한 경우, StorageGRID는 객체의 일부 또는 세그먼트를 샘플링하여 클라우드 스토리지 풀에서 해당 객체를 사용할 수 있는지 여부를 확인합니다. 경우에 따라 GetObject 요청이 객체의 일부가 이미 복구 불가능한 상태로 전환되었거나 아직 복원되지 않은 경우 잘못된 결과를 반환할 수 있습니다. 200 OK

이러한 경우:

- GetObject 요청은 일부 데이터를 반환할 수 있지만 전송 도중에 중단될 수 있습니다.
- 이후의 GetObject 요청은 `403 Forbidden`을 반환할 수 있습니다.

GetObject 및 그리드 간 복제

"[그리드 페더레이션](#)" 및 "[교차 그리드 복제](#)"가 버킷에 대해 활성화된 경우, S3 클라이언트는 GetObject 요청을 통해 객체의 복제 상태를 확인할 수 있습니다. 응답에는 StorageGRID 관련 x-ntap-sg-cgr-replication-status 응답 헤더가 포함되며, 이 헤더에는 다음 값 중 하나가 있습니다.

그리드	복제 상태
소스	• 완료됨: 복제가 성공적으로 완료되었습니다. • PENDING: 해당 개체가 아직 복제되지 않았습니다. • FAILURE: 복제가 영구적으로 실패했습니다. 사용자가 오류를 해결해야 합니다.
목적지	REPLICA : 해당 객체는 원본 그리드에서 복제되었습니다.



StorageGRID는 x-amz-replication-status 헤더를 지원하지 않습니다.

S3 HeadObject 요청을 사용하여 StorageGRID의 객체에서 메타데이터를 검색합니다

S3 HeadObject 요청을 사용하면 객체 자체를 반환하지 않고도 객체의 메타데이터를 검색할 수 있습니다. 객체가 클라우드 스토리지 풀에 저장된 경우 HeadObject를 사용하여 객체의 전환 상태를 확인할 수 있습니다.

HeadObject 및 멀티파트 객체

멀티파트 또는 분할된 객체의 특정 부분에 대한 partNumber 요청 매개변수를 사용하여 메타데이터를 검색할 수 있습니다. x-amz-mp-parts-count 응답 요소는 객체가 몇 부분으로 구성되어 있는지를 나타냅니다.

`partNumber`을 분할/멀티파트 객체와 비분할/비멀티파트 객체 모두에 대해 1로 설정할 수 있지만, `x-amz-mp-parts-count` 응답 요소는 분할 또는 멀티파트 객체에 대해서만 반환됩니다.

사용자 메타데이터의 UTF-8 문자

StorageGRID는 사용자 정의 메타데이터에서 이스케이프된 UTF-8 문자를 구문 분석하거나 해석하지 않습니다. 사용자 정의 메타데이터에 이스케이프된 UTF-8 문자가 있는 객체에 대한 HEAD 요청은 키 이름이나 값에 인쇄할 수 없는 문자가 포함된 경우 x-amz-missing-meta 헤더를 반환하지 않습니다.

지원되는 요청 헤더

다음 요청 헤더가 지원됩니다.

- x-amz-checksum-mode

`partNumber` 매개변수와 `Range` 헤더는 HeadObject에 대한 `x-amz-checksum-mode`에서 지원되지 않습니다. 해당 매개변수와 헤더를 `x-amz-checksum-mode`가 활성화된 요청에 포함하면 StorageGRID는 응답에 체크섬 값을 반환하지 않습니다.

지원되지 않는 요청 헤더

다음 요청 헤더는 지원되지 않으며 `XNotImplemented`을(를) 반환합니다.

- x-amz-website-redirect-location

버전 관리

If a versionId 하위 리소스가 지정되지 않은 경우, 해당 작업은 버전이 지정된 버킷에서 객체의 최신 버전을

가져옵니다. 현재 객체 버전이 삭제 마커인 경우, "찾을 수 없음" 상태가 반환되며 x-amz-delete-marker 응답 헤더가 `true`로 설정됩니다.

고객이 제공한 암호화 키를 사용한 서버 측 암호화(SSE-C) 요청 헤더

제공된 고유 키로 객체가 암호화된 경우 이 세 가지 헤더를 모두 사용하십시오.

- x-amz-server-side-encryption-customer-algorithm: `AES256`을(를) 지정합니다.
- x-amz-server-side-encryption-customer-key: 객체에 사용할 암호화 키를 지정하십시오.
- x-amz-server-side-encryption-customer-key-MD5: 객체의 암호화 키의 MD5 다이제스트를 지정합니다.



고객이 제공하는 암호화 키는 절대 저장되지 않습니다. 암호화 키를 분실하면 해당 객체도 손실됩니다. 고객이 제공한 키를 사용하여 객체 데이터를 보호하기 전에 ["서버 측 암호화 사용"](#)의 고려 사항을 검토하십시오.

HeadObject 클라우드 스토리지 풀 객체에 대한 응답

객체가 ["클라우드 스토리지 풀"](#)에 저장된 경우 다음과 같은 응답 헤더가 반환됩니다.

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

응답 헤더는 객체가 클라우드 스토리지 풀로 이동될 때, 선택적으로 복구 불가능한 상태로 전환될 때, 그리고 복원될 때 객체의 상태에 대한 정보를 제공합니다.

객체의 상태	HeadObject에 대한 응답
StorageGRID에 수집되었지만 ILM에서 아직 평가되지 않은 객체이거나, 기존 스토리지 풀에 저장되었거나 이레이저 코딩을 사용하여 저장된 객체입니다.	200 OK (특별한 응답 헤더는 반환되지 않습니다.)
클라우드 스토리지 풀에 있는 객체이지만 아직 검색 불가능한 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" 객체가 검색 불가능한 상태로 전환될 때까지 `expiry-date`의 값은 먼 미래의 시간으로 설정됩니다. 전환 시점은 StorageGRID 시스템에서 제어하지 않습니다.

객체의 상태	HeadObject에 대한 응답
객체가 검색할 수 없는 상태로 전환되었지만, 그리드에 적어도 하나의 복사본이 존재합니다.	200 OK x-amz-storage-class: GLACIER x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT" 해당 값은 expiry-date 먼 미래의 어느 시점으로 설정되어 있습니다. 참고: 그리드의 복사본을 사용할 수 없는 경우(예: 스토리지 노드가 다운된 경우) 객체를 성공적으로 검색하려면 클라우드 스토리지 풀에서 복사본을 복원하기 위한 "RestoreObject" 요청을 실행해야 합니다.
객체가 검색할 수 없는 상태로 전환되었으며, 그리드에 복사본이 존재하지 않습니다.	200 OK x-amz-storage-class: GLACIER
복구 불가능한 상태에서 복원 중인 객체	200 OK x-amz-storage-class: GLACIER x-amz-restore: ongoing-request="true"
객체가 클라우드 스토리지 풀에 완전히 복원되었습니다.	200 OK x-amz-storage-class: GLACIER x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2018 00:00:00 GMT" `expiry-date`는 Cloud Storage Pool의 객체가 검색 불가능한 상태로 반환되는 시점을 나타냅니다.

클라우드 스토리지 풀의 멀티파트 또는 분할된 객체

여러 부분으로 나뉜 객체를 업로드했거나 StorageGRID가 큰 객체를 여러 세그먼트로 분할한 경우, StorageGRID는 객체의 일부 또는 세그먼트를 샘플링하여 클라우드 스토리지 풀에서 해당 객체를 사용할 수 있는지 여부를 확인합니다. 경우에 따라 HeadObject 요청이 객체의 일부가 이미 복구 불가능한 상태로 전환되었거나 아직 복원되지 않은 경우에도 잘못된 결과를 x-amz-restore: ongoing-request="false" 반환할 수 있습니다.

HeadObject 및 그리드 간 복제

"[그리드 페더레이션](#)"을 사용하고 버킷에 대해 "[교차 그리드 복제](#)"가 활성화된 경우, S3 클라이언트는 HeadObject 요청을 통해 객체의 복제 상태를 확인할 수 있습니다. 응답에는 StorageGRID 관련 `x-ntap-sg-cgr-replication-status` 응답 헤더가 포함되며, 이 헤더에는 다음 값 중 하나가 있습니다.

그리드	복제 상태
소스	- 완료됨: 복제가 성공적으로 완료되었습니다. PENDING: 해당 개체가 아직 복제되지 않았습니다. FAILURE: 복제가 영구적으로 실패했습니다. 사용자가 오류를 해결해야 합니다.
목적지	REPLICA : 해당 객체는 원본 그리드에서 복제되었습니다.



StorageGRID는 `x-amz-replication-status` 헤더를 지원하지 않습니다.

S3 PutObject 요청을 사용하여 StorageGRID의 버킷에 객체 추가

S3 PutObject 요청을 사용하여 버킷에 객체를 추가할 수 있습니다.

충돌 해결

두 클라이언트가 동일한 키에 쓰기를 시도하는 등 클라이언트 요청이 충돌하는 경우 "최신 요청 우선" 원칙에 따라 처리됩니다. "최신 요청 우선" 평가 시점은 S3 클라이언트가 작업을 시작한 시점이 아니라 StorageGRID 시스템이 해당 요청을 완료한 시점을 기준으로 합니다.

객체 크기

단일 PutObject 작업에 권장되는 최대 크기는 5GiB(5,368,709,120바이트)입니다. 객체 크기가 5GiB보다 큰 경우, "[멀티파트 업로드](#)"를 사용하십시오.

단일 PutObject 작업에 대해 지원되는 최대 크기는 5TiB(5,497,558,138,880바이트)입니다.



StorageGRID 11.6 이하 버전에서 업그레이드한 경우, 5GiB를 초과하는 객체를 업로드하려고 하면 S3 PUT Object size too large 경고가 트리거됩니다. StorageGRID 11.7 또는 11.8을 새로 설치한 경우에는 이 경우 경고가 트리거되지 않습니다. 하지만 AWS S3 표준을 준수하기 위해 향후 StorageGRID 릴리스에서는 5GiB보다 큰 객체 업로드를 지원하지 않을 예정입니다.

사용자 메타데이터 크기

Amazon S3는 각 PUT 요청 헤더 내 사용자 정의 메타데이터의 크기를 2KB로 제한합니다. StorageGRID는 사용자 메타데이터를 24KiB로 제한합니다. 사용자 정의 메타데이터의 크기는 각 키와 값의 UTF-8 인코딩 바이트 수를 합산하여 측정합니다.

사용자 메타데이터의 UTF-8 문자

요청에 사용자 정의 메타데이터의 키 이름 또는 값에 (이스케이프되지 않은) UTF-8 값이 포함된 경우 StorageGRID 동작은 정의되지 않습니다.

StorageGRID는 사용자 정의 메타데이터의 키 이름 또는 값에 포함된 이스케이프 처리된 UTF-8 문자를 구문 분석하거나 해석하지 않습니다. 이스케이프 처리된 UTF-8 문자는 ASCII 문자로 처리됩니다.

- PutObject, CopyObject, GetObject, and HeadObject 요청은 사용자 정의 메타데이터에 이스케이프된 UTF-8 문자가 포함된 경우 성공합니다.
- StorageGRID는 키 이름이나 값의 해석된 값에 인쇄할 수 없는 문자가 포함된 경우 x-amz-missing-meta 헤더를 반환하지 않습니다.

객체 태그 제한

새 객체를 업로드할 때 태그를 추가하거나 기존 객체에 태그를 추가할 수 있습니다. StorageGRID와 Amazon S3 모두 객체당 최대 10개의 태그를 지원합니다. 객체와 연결된 태그는 고유한 태그 키를 가져야 합니다. 태그 키는 최대 128자의 유니코드 문자, 태그 값은 최대 256자의 유니코드 문자까지 사용할 수 있습니다. 키와 값은 대소문자를 구분합니다.

객체 소유권

StorageGRID에서 모든 객체는 버킷 소유자 계정이 소유합니다. 여기에는 소유자가 아닌 계정이나 익명 사용자가 생성한 객체도 포함됩니다.

지원되는 요청 헤더

다음과 같은 요청 헤더가 지원됩니다.

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

``aws-chunked``을 (를) ``Content-Encoding``에 대해 지정하면 StorageGRID 는 다음 항목을 확인하지 않습니다.

- StorageGRID는 청크 데이터에 대해 ``chunk-signature``의 유효성을 검사하지 않습니다.
- StorageGRID는 사용자가 제공하는 값을 `x-amz-decoded-content-length` 객체와 비교하여 검증하지 않습니다.

- Content-Language
- Content-Length
- Content-MD5
- Content-Type
- Expires

- Transfer-Encoding

청크 전송 인코딩은 `aws-chunked` 페이로드 서명도 함께 사용되는 경우에 지원됩니다.

- `x-amz-checksum-sha256`
- `x-amz-meta-`, 그 뒤에 사용자 정의 메타데이터를 포함하는 이름-값 쌍이 옵니다.

사용자 정의 메타데이터의 이름-값 쌍을 지정할 때 다음 일반 형식을 사용하십시오.

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

ILM 규칙의 참조 시간으로 사용자 정의 생성 시간 옵션을 사용하려면 ``creation-time``을(를) 객체가 생성된 시점을 기록하는 메타데이터의 이름으로 사용해야 합니다. 예를 들면 다음과 같습니다.

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

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



ILM 규칙은 참조 시간에 대해 *사용자 정의 생성 시간*을 사용하면서 동시에 균형 또는 엄격 수집 옵션을 사용할 수 없습니다. ILM 규칙을 생성할 때 오류가 반환됩니다.

- `x-amz-tagging`
- S3 객체 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 버킷의 기본 보존 설정이 객체 버전 모드와 보존 만료일을 계산하는 데 사용됩니다. ["S3 REST API를 사용하여 S3 Object Lock 구성"](#)을 참조하십시오.

- 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`

[서버 측 암호화에 대한 요청 헤더](#)을 참조하십시오

지원되지 않는 요청 헤더

다음 요청 헤더는 지원되지 않습니다.

- If-Match

`If-Match header`은 (는) 승인되었지만 작동하지 않습니다.

- If-None-Match

`If-None-Match header`은 (는) 승인되었지만 작동하지 않습니다.

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

The x-amz-website-redirect-location 헤더는 `XNotImplemented`을 반환합니다.

스토리지 클래스 옵션

`x-amz-storage-class` 요청 헤더가 지원됩니다. `x-amz-storage-class`에 제출된 값은 StorageGRID 수집 중에 객체 데이터를 보호하는 방식에 영향을 미치며, StorageGRID 시스템에 객체의 영구 복사본이 몇 개 저장되는지(ILM에 의해 결정됨)에는 영향을 미치지 않습니다.

수집된 객체와 일치하는 ILM 규칙이 엄격한 수집 옵션을 사용하는 경우 x-amz-storage-class 헤더는 아무런 효과가 없습니다.

다음 값을 사용할 수 있습니다 x-amz-storage-class:

- STANDARD (기본값)
 - 듀얼 커밋: ILM 규칙에서 수집 동작에 대해 듀얼 커밋 옵션을 지정한 경우, 객체가 수집되는 즉시 해당 객체의 두 번째 복사본이 생성되어 다른 스토리지 노드에 배포됩니다(듀얼 커밋). ILM을 평가할 때 StorageGRID는 이러한 초기 임시 복사본이 규칙의 배치 지침을 충족하는지 확인합니다. 충족하지 않는 경우, 다른 위치에 새 객체 복사본을 생성하고 초기 임시 복사본을 삭제해야 할 수 있습니다.
 - 균형: ILM 규칙에 균형 옵션이 지정되어 있고 StorageGRID가 규칙에 지정된 모든 복사본을 즉시 생성할 수 없는 경우 StorageGRID는 서로 다른 스토리지 노드에 두 개의 임시 복사본을 생성합니다.

StorageGRID가 ILM 규칙에 지정된 모든 객체 복사본을 즉시 생성할 수 있는 경우(동기 배치), x-amz-storage-class 헤더는 아무런 효과가 없습니다.

- REDUCED_REDUNDANCY
 - 듀얼 커밋: ILM 규칙에서 수집 동작에 대해 듀얼 커밋 옵션을 지정한 경우, StorageGRID는 객체가 수집될 때 단일 임시 복사본을 생성합니다(싱글 커밋).
 - 균형: ILM 규칙에 균형 옵션이 지정된 경우, StorageGRID 시스템이 규칙에 지정된 모든 복사본을 즉시 생성할 수 없는 경우에만 임시 복사본을 하나만 생성합니다. StorageGRID 동기 배치를 수행할 수 있는 경우에는 이

헤더가 아무런 효과가 없습니다. 이 `REDUCED_REDUNDANCY` 옵션은 객체와 일치하는 ILM 규칙이 단일 복제본을 생성하는 경우에 가장 적합합니다. 이 경우 `'REDUCED_REDUNDANCY'`를 사용하면 모든 수집 작업마다 불필요한 객체 복사본 생성 및 삭제를 방지할 수 있습니다.

다른 상황에서는 `REDUCED_REDUNDANCY` 옵션을 사용하는 것을 권장하지 않습니다.

`'REDUCED_REDUNDANCY'`은(는) 수집 중 오브젝트 데이터 손실 위험을 증가시킵니다. 예를 들어, ILM 평가가 수행되기 전에 장애가 발생한 Storage Node에 단일 복사본이 처음에 저장되는 경우 데이터를 손실할 수 있습니다.



특정 기간 동안 복제본이 하나만 존재하는 경우 데이터가 영구적으로 손실될 위험이 있습니다. 객체의 복제본이 하나만 있는 경우 스토리지 노드에 장애가 발생하거나 중대한 오류가 발생하면 해당 객체가 손실됩니다. 또한 업그레이드와 같은 유지 관리 절차 중에도 해당 객체에 대한 액세스가 일시적으로 제한됩니다.

이 설정을 지정하면 `REDUCED_REDUNDANCY` 객체가 처음 수집될 때 생성되는 복사본 수에만 영향을 미칩니다. 활성 ILM 정책에 따라 객체를 평가할 때 생성되는 객체 복사본 수에는 영향을 미치지 않으며, StorageGRID 시스템에서 데이터가 더 낮은 수준의 중복성으로 저장되는 결과를 초래하지도 않습니다.



S3 객체 잠금이 활성화된 버킷에 객체를 수집하는 경우 `REDUCED_REDUNDANCY` 옵션은 무시됩니다. 레거시 규정 준수 버킷에 객체를 수집하는 경우 `REDUCED_REDUNDANCY` 옵션은 오류를 반환합니다. StorageGRID는 규정 준수 요구 사항을 충족하기 위해 항상 이중 커밋 수집을 수행합니다.

서버 측 암호화에 대한 요청 헤더

다음 요청 헤더를 사용하여 서버 측 암호화로 객체를 암호화할 수 있습니다. SSE와 SSE-C 옵션은 상호 배타적입니다.

- **SSE:** StorageGRID에서 관리하는 고유 키로 객체를 암호화하려면 다음 헤더를 사용하십시오.

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

`'x-amz-server-side-encryption'` 헤더가 `PutObject` 요청에 포함되지 않으면 `grid-wide xref:{relative_path}../admin/changing-network-options-object-encryption.html["저장된 객체 암호화 설정"]`가 `PutObject` 응답에서 생략됩니다.

- **SSE-C:** 직접 제공하고 관리하는 고유 키로 객체를 암호화하려면 이 세 가지 헤더를 모두 사용하십시오.

- `x-amz-server-side-encryption-customer-algorithm:` `'AES256'`을(를) 지정합니다.

- `x-amz-server-side-encryption-customer-key:` 새 객체에 사용할 암호화 키를 지정하십시오.

- `x-amz-server-side-encryption-customer-key-MD5:` 새 객체의 암호화 키의 MD5 다이제스트를 지정합니다.



고객이 제공하는 암호화 키는 절대 저장되지 않습니다. 암호화 키를 분실하면 해당 객체도 손실됩니다. 고객이 제공한 키를 사용하여 객체 데이터를 보호하기 전에 ["서버 측 암호화 사용"](#)에 대한 고려 사항을 검토하십시오.



객체가 SSE 또는 SSE-C로 암호화된 경우 버킷 수준 또는 그리드 수준의 암호화 설정은 무시됩니다.

버전 관리

버킷에 버전 관리가 활성화된 경우, 저장된 객체의 버전에 대한 고유 `versionId`가 자동으로 생성됩니다. 이 `versionId`는 `x-amz-version-id` 응답 헤더를 사용하여 응답에도 반환됩니다.

버전 관리가 일시 중단되면 객체 버전은 null `versionId`으로 저장되며, 이미 null 버전이 존재하는 경우 덮어쓰여집니다.

Authorization 헤더에 대한 서명 계산

`Authorization` 헤더를 사용하여 요청을 인증할 때 StorageGRID는 다음과 같은 점에서 AWS와 다릅니다.

- StorageGRID는 헤더를 host 내에 포함할 필요가 없습니다 CanonicalHeaders.
- StorageGRID는 Content-Type `이(가)` CanonicalHeaders 내에 포함될 필요가 없습니다.
- StorageGRID는 x-amz-* 헤더가 CanonicalHeaders 안에 포함될 필요가 없습니다.



일반적으로 이러한 헤더를 항상 CanonicalHeaders 내에 포함하여 검증하는 것이 좋습니다. 하지만 이러한 헤더를 제외하더라도 StorageGRID는 오류를 반환하지 않습니다.

자세한 내용은 ["권한 부여 헤더에 대한 서명 계산: 단일 청크로 페이로드 전송\(AWS Signature Version 4\)"](#)를 참조하십시오.

관련 정보

- ["ILM을 사용하여 객체 관리"](#)
- ["Amazon Simple Storage Service API 참조: PutObject"](#)

S3 RestoreObject 요청을 사용하여 StorageGRID에서 객체 복원

S3 RestoreObject 요청을 사용하면 클라우드 스토리지 풀에 저장된 객체를 복원할 수 있습니다.

지원되는 요청 유형

StorageGRID는 객체를 복원하기 위해 RestoreObject 요청만 지원합니다. SELECT 복원 유형은 지원하지 않습니다. Select 요청은 `XNotImplemented`을 반환합니다.

버전 관리

선택적으로, `versionId` 버전이 지정된 버킷에서 객체의 특정 버전을 복원하도록 지정할 수 있습니다. 지정하지 않으면 `versionId` 객체의 최신 버전이 복원됩니다.

클라우드 스토리지 풀 객체에 대한 RestoreObject의 동작

객체가 ["클라우드 스토리지 풀"](#)에 저장된 경우, RestoreObject 요청은 객체의 상태에 따라 다음과 같은 동작을 합니다. 자세한 내용은 ["HeadObject"](#)를 참조하십시오.



객체가 클라우드 스토리지 풀에 저장되어 있고 해당 객체의 복사본이 그리드에도 하나 이상 존재하는 경우, RestoreObject 요청을 실행하여 객체를 복원할 필요가 없습니다. 대신 GetObject 요청을 사용하여 로컬 복사본을 직접 검색할 수 있습니다.

객체의 상태	RestoreObject의 동작
StorageGRID에 수집되었지만 ILM에서 아직 평가되지 않은 객체이거나, 클라우드 스토리지 풀에 없는 객체입니다.	403 Forbidden, InvalidObjectState
클라우드 스토리지 풀에 있는 객체이지만 아직 검색 불가능한 상태로 전환되지 않았습니다.	200 OK 변경 사항이 없습니다. 참고: 객체가 검색 불가능한 상태로 전환되기 전에는 해당 객체의 `expiry-date`을(를) 변경할 수 없습니다.
객체가 복구 불가능한 상태로 전환되었습니다.	202 Accepted 요청 본문에 지정된 일수 동안 객체의 검색 가능한 복사본을 클라우드 스토리지 풀에 복원합니다. 이 기간이 종료되면 객체는 검색 불가능한 상태로 돌아갑니다. 선택적으로, Tier 요청 요소를 사용하여 복원 작업이 완료되는 데 걸리는 시간((Expedited, Standard, 또는 Bulk)을 결정할 수 있습니다. Tier`를 지정하지 않으면 `Standard 티어가 사용됩니다. 중요: 개체가 S3 Glacier Deep Archive로 전환되었거나 클라우드 스토리지 풀이 Azure Blob 스토리지를 사용하는 경우, 해당 Expedited 계층을 사용하여 복원할 수 없습니다. 다음 오류가 반환됩니다 403 Forbidden, InvalidTier:Retrieval option is not supported by this storage class.
복구 불가능한 상태에서 복원 중인 객체	409 Conflict, RestoreAlreadyInProgress
객체가 클라우드 스토리지 풀에 완전히 복원되었습니다.	200 OK 참고: 객체가 검색 가능한 상태로 복원된 경우, expiry-date`에 대한 새 값으로 RestoreObject 요청을 다시 실행하여 복원 날짜를 변경할 수 있습니다. `Days 복원 날짜는 요청 시간을 기준으로 업데이트됩니다.

S3 SelectObjectContent 요청을 사용하여 StorageGRID에서 객체 데이터 필터링

S3 SelectObjectContent 요청을 사용하면 간단한 SQL 문을 기반으로 S3 객체의 내용을 필터링할 수 있습니다.

자세한 내용은 "[Amazon Simple Storage Service API 참조: SelectObjectContent](#)"을 참조하십시오.

시작하기 전에

- 테넌트 계정에는 S3 Select 권한이 있습니다.
- 조회하려는 객체에 대한 `s3:GetObject` 권한이 있습니다.
- 조회하려는 객체는 다음 형식 중 하나여야 합니다.
 - **CSV.** 있는 그대로 사용하거나 GZIP 또는 BZIP2 아카이브로 압축할 수 있습니다.
 - **Parquet.** Parquet 객체에 대한 추가 요구사항:
 - S3 Select는 GZIP 또는 Snappy를 사용한 컬럼형 압축만 지원합니다. S3 Select는 Parquet 객체에 대한 전체 객체 압축을 지원하지 않습니다.
 - S3 Select는 Parquet 출력을 지원하지 않습니다. 출력 형식을 CSV 또는 JSON으로 지정해야 합니다.
 - 압축되지 않은 행 그룹의 최대 크기는 512MB입니다.
 - 객체의 스키마에 명시된 데이터 형식을 사용해야 합니다.
 - INTERVAL, JSON, LIST, TIME 또는 UUID 논리 유형은 사용할 수 없습니다.
- SQL 표현식의 최대 길이는 256 KB입니다.
- 입력 또는 결과의 모든 레코드는 최대 길이가 1MiB입니다.

CSV 요청 구문 예시

```

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 요청 구문 예시

```

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 쿼리 예시

이 쿼리는 주 이름, 2010년 인구, 2015년 추정 인구 및 미국 인구 조사 데이터 대비 변화율을 가져옵니다. 파일에서 주가 아닌 레코드는 무시됩니다.

```

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

```

조회할 파일의 처음 몇 줄인 `SUB-EST2020\_ALL.csv`은 다음과 같습니다.

```
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 예시 (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 줄인 `changes.csv`은 다음과 같습니다.

```
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 사용 예시 (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 파일인 changes.csv의 처음 몇 줄은 다음과 같습니다.

```
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
```

저작권 정보

Copyright © 2026 NetApp, Inc. All Rights Reserved. 미국에서 인쇄된 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 라이선스와 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 라이선스도 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 라이선스를 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 라이선스는 DFARS 조항 252.227-7015(b)(2014년 2월)에 명시된 권한으로 제한됩니다.

상표 정보

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