Utilisez l'API REST S3

StorageGRID software

NetApp
July 23, 2026

Sommaire

Utilisez l'API REST S3

Versions et mises à jour prises en charge de l'API REST StorageGRID S3

StorageGRID prend en charge l'API Simple Storage Service (S3), qui est implémentée sous la forme d'un ensemble de services web REST (Representational State Transfer).

La prise en charge de l'API REST S3 vous permet de connecter des applications orientées services développées pour les services web S3 à un stockage d'objets sur site utilisant le système StorageGRID. Seules des modifications minimes sont nécessaires dans l'utilisation actuelle des appels à l'API REST S3 par l'application cliente.

Versions prises en charge

StorageGRID prend en charge les versions spécifiques suivantes de STS, S3 et HTTP :

Article	Version
Spécification de l'API STS AssumeRole	"Documentation Amazon Web Services (AWS) : Référence de l'API Amazon Assumerole" Pour plus d'informations sur AssumeRole, veuillez consulter "Configurez AssumeRole".
Spécification de l'API S3	"Documentation AWS : Référence de l'API Amazon Simple Storage Service"
HTTP	1,1 Pour plus d'informations sur HTTP, voir HTTP/1.1 (RFCs 7230-35). "IETF RFC 2616 : Protocole de transfert hypertexte (HTTP/1.1)" **Remarque** : StorageGRID ne prend pas en charge le pipelining HTTP/1.1.

Mises à jour de la prise en charge de l'API REST S3

Version	Commentaires
12,0	• Ajout de la prise en charge du partage de ressources entre origines (CORS) pour une interface de gestion, permettant à un autre domaine d'accéder aux données de StorageGRID à l'aide des API de gestion. "En savoir plus". • Ajout de la prise en charge de STS AssumeRole et des stratégies de session. Voir "un exemple de politique de session". Vous pouvez configurer AssumeRole dans les groupes de locataires.

Version	Commentaires
11,9	• Ajout de la prise en charge des valeurs de somme de contrôle SHA-256 précalculées pour les requêtes et en-têtes suivants. Vous pouvez utiliser cette fonctionnalité pour vérifier l'intégrité des objets téléchargés : ◦ CompleteMultipartUpload: `x-amz-checksum-sha256` ◦ CreateMultipartUpload: `x-amz-checksum-algorithm` ◦ GetObject: `x-amz-checksum-mode` ◦ HeadObject: `x-amz-checksum-mode` ◦ ListParts ◦ PutObject: `x-amz-checksum-sha256` ◦ UploadPart: `x-amz-checksum-sha256` • Ajout de la possibilité pour l'administrateur de la grille de contrôler les paramètres de rétention et de conformité au niveau du locataire. Ces paramètres affectent les paramètres S3 Object Lock. ◦ Mode de rétention par défaut du compartiment et mode de rétention des objets : Gouvernance ou Conformité, si autorisé par l'administrateur de la grille. ◦ Période de rétention par défaut du compartiment et date de conservation de l'objet : doit être inférieure ou égale à la période de rétention maximale autorisée par l'administrateur de la grille. • Amélioration de la prise en charge de `aws-chunked` l'encodage du contenu et de la diffusion en continu des valeurs `x-amz-content-sha256`. Limitations : ◦ Si présent, `chunk-signature` est facultatif et n'est pas validé ◦ Si présent, `x-amz-trailer` le contenu est ignoré
11,8	Les noms des opérations S3 ont été mis à jour pour correspondre aux noms utilisés dans le "Documentation Amazon Web Services (AWS) : Référence de l'API Amazon Simple Storage Service".
11,7	• Ajouté "Référence rapide : Requêtes API S3 prises en charge". • Ajout de la prise en charge de l'utilisation du mode GOVERNANCE avec S3 Object Lock. • Ajout de la prise en charge de l'en-tête de réponse spécifique à StorageGRID `x-ntap-sg-cgr-replication-status` pour les requêtes GET Object et HEAD Object. Cet en-tête indique l'état de réplication d'un objet pour la réplication inter-grilles. • SelectObjectContent prend désormais en charge les objets Parquet.

Version	Commentaires
11,6	• Ajout de la prise en charge de l'utilisation du `partNumber` paramètre de requête dans les requêtes GET Object et HEAD Object. • Ajout de la prise en charge d'un mode de rétention par défaut et d'une période de rétention par défaut au niveau du compartiment pour S3 Object Lock. • Ajout de la prise en charge de la `s3:object-lock-remaining-retention-days`clé de condition de stratégie pour définir la plage des périodes de conservation autorisées pour vos objets. • La taille maximale *recommandée* pour une opération PUT Object unique est désormais de 5 Gio (5 368 709 120 octets). Si vous avez des objets d'une taille supérieure à 5 Gio, utilisez plutôt le chargement partitionné.
11,5	• Ajout de la prise en charge de la gestion du chiffrement des compartiments. • Ajout de la prise en charge de S3 Object Lock et suppression des anciennes requêtes de conformité. • Ajout de la prise en charge de l'utilisation de DELETE Multiple Objects sur les compartiments versionnés. • L' `Content-MD5`en-tête de requête est désormais correctement pris en charge.
11,4	• Ajout de la prise en charge de DELETE Bucket tagging, GET Bucket tagging et PUT Bucket tagging. Les balises d'allocation de coûts ne sont pas prises en charge. • Pour les compartiments créés dans StorageGRID 11.4, il n'est plus nécessaire de limiter les noms de clés d'objet pour respecter les bonnes pratiques de performance. • Ajout de la prise en charge des notifications de compartiment sur le type d'événement `s3:ObjectRestore:Post`. • Les limites de taille AWS pour les chargements en plusieurs parties sont désormais appliquées. Chaque partie d'un chargement en plusieurs parties doit avoir une taille comprise entre 5 MiB et 5 GiB. La dernière partie peut être inférieure à 5 MiB. • Ajout de la prise en charge de TLS 1.3
11,3	• Ajout de la prise en charge du chiffrement côté serveur des données d'objet avec des clés fournies par le client (SSE-C). • Ajout de la prise en charge des opérations DELETE, GET et PUT du cycle de vie des compartiments (action d'expiration uniquement) et de l'en-tête de réponse `x-amz-expiration`. • Mise à jour de PUT Object, PUT Object - Copy et Multipart Upload pour décrire l'impact des règles ILM qui utilisent un placement synchrone à l'ingestion. • Les chiffrements TLS 1.1 ne sont plus pris en charge.

Version	Commentaires
11,2	Ajout de la prise en charge de la restauration d'objets POST pour une utilisation avec les Cloud Storage Pools. Ajout de la prise en charge de l'utilisation de la syntaxe AWS pour les ARN, les clés de condition de stratégie et les variables de stratégie dans les stratégies de groupe et de compartiment. Les stratégies de groupe et de compartiment existantes qui utilisent la syntaxe StorageGRID continueront d'être prises en charge. **Remarque :** Les utilisations des ARN/URN dans d'autres configurations JSON/XML, y compris celles utilisées dans les fonctionnalités personnalisées de StorageGRID, n'ont pas changé.
11,1	Ajout de la prise en charge du partage de ressources entre origines (CORS), du protocole HTTP pour les connexions client S3 aux nœuds de la grille et des paramètres de conformité sur les compartiments.
11,0	Ajout de la prise en charge de la configuration des services de plateforme (CloudMirror replication, notifications et intégration de la recherche Elasticsearch) pour les compartiments. Ajout également de la prise en charge des contraintes de localisation des balises d'objets pour les compartiments et de la cohérence Available.
10,4	Ajout de la prise en charge des modifications de versionnement de l'analyse ILM, des mises à jour de la page Noms de domaine des points de terminaison, des conditions et des variables dans les politiques, des exemples de politiques et de l'autorisation PutOverwriteObject.
10,3	Ajout de la prise en charge du versionnage.
10,2	Ajout de la prise en charge des politiques d'accès aux groupes et aux compartiments, ainsi que de la copie en plusieurs parties (Upload Part - Copy).
10,1	Ajout de la prise en charge du téléchargement en plusieurs parties, des requêtes de type hébergement virtuel et de l'authentification v4.
10,0	Prise en charge initiale de l'API REST S3 par le système StorageGRID. La version actuellement prise en charge de la *Référence de l'API Simple Storage Service* est 2006-03-01.

Requêtes S3 API prises en charge dans StorageGRID

Cette page résume comment StorageGRID prend en charge les API Amazon Simple Storage Service (S3).

Cette page ne comprend que les opérations S3 prises en charge par StorageGRID.

Pour consulter la documentation AWS relative à chaque opération, sélectionnez le lien dans l'en-tête.

Paramètres de requête URI et en-têtes de requête courants

Sauf indication contraire, les paramètres de requête URI courants suivants sont pris en charge :

- `versionId` (comme requis pour les opérations sur les objets)

${post_edited_translations.segment}

- `Authorization`
- `Connection`
- `Content-Length`
- `Content-MD5`
- `Content-Type`
- `Date`
- `Expect`
- `Host`
- `x-amz-date`

Informations connexes

- "Détails de l'implémentation de l'API REST S3"
- "Référence de l'API Amazon Simple Storage Service : En-têtes de requête courants"

"AbortMultipartUpload"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette requête, ainsi que ce paramètre de requête URI supplémentaire :

- `uploadId`

Corps de la requête

Aucune

Documentation StorageGRID

"${post_edited_translations.segment}"

"CompleteMultipartUpload"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette requête, ainsi que ce paramètre de requête URI supplémentaire :

- `uploadId`
- `x-amz-checksum-sha256`

Balises XML du corps de la requête

${post_edited_translations.segment}

- `ChecksumSHA256`
- `CompleteMultipartUpload`
- `ETag`
- `Part`
- `PartNumber`

Documentation StorageGRID

"CompleteMultipartUpload"

"CopyObject"

${post_edited_translations.segment}

StorageGRID prend en charge toutes paramètres et en-têtes communs pour cette requête, ainsi que les en-têtes supplémentaires suivants :

- `x-amz-copy-source`
- `x-amz-copy-source-if-match`
- `x-amz-copy-source-if-modified-since`
- `x-amz-copy-source-if-none-match`
- `x-amz-copy-source-if-unmodified-since`
- `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-metadata-directive`
- `x-amz-object-lock-legal-hold`
- `x-amz-object-lock-mode`
- `x-amz-object-lock-retain-until-date`
- `x-amz-server-side-encryption`
- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`
- `x-amz-storage-class`
- `x-amz-tagging`
- `x-amz-tagging-directive`
- `x-amz-meta-<metadata-name>`

Corps de la requête

Aucune

Documentation StorageGRID

"CopyObject"

"CreateBucket"

${post_edited_translations.segment}

StorageGRID prend en charge toutes paramètres et en-têtes communs pour cette requête, ainsi que les en-têtes supplémentaires suivants :

- `x-amz-bucket-object-lock-enabled`

Corps de la requête

StorageGRID prend en charge tous les paramètres du corps de requête définis par l'API REST Amazon S3 au moment de l'implémentation.

Documentation StorageGRID

"Opérations sur les compartiments"

"CreateMultipartUpload"

${post_edited_translations.segment}

StorageGRID prend en charge toutes paramètres et en-têtes communs pour cette requête, ainsi que les en-têtes supplémentaires suivants :

- `Cache-Control`
- `Content-Disposition`
- `Content-Encoding`
- `Content-Language`
- `Expires`
- `x-amz-checksum-algorithm`
- `x-amz-server-side-encryption`
- `x-amz-storage-class`
- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`
- `x-amz-tagging`
- `x-amz-object-lock-mode`
- `x-amz-object-lock-retain-until-date`
- `x-amz-object-lock-legal-hold`
- `x-amz-meta-<metadata-name>`

Corps de la requête

Aucune

Documentation StorageGRID

"CreateMultipartUpload"

"DeleteBucket"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Documentation StorageGRID

"Opérations sur les compartiments"

"DeleteBucketCors"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"DeleteBucketEncryption"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"DeleteBucketLifecycle"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

- "Opérations sur les compartiments"
- "Créer une configuration de cycle de vie S3"

"DeleteBucketPolicy"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"DeleteBucketReplication"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"DeleteBucketTagging"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"DeleteObject"

${post_edited_translations.segment}

StorageGRID prend en charge toutes paramètres et en-têtes communs pour cette requête, ainsi que cet en-tête de requête supplémentaire :

- `x-amz-bypass-governance-retention`

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les objets"

"DeleteObjects"

${post_edited_translations.segment}

StorageGRID prend en charge toutes paramètres et en-têtes communs pour cette requête, ainsi que cet en-tête de requête supplémentaire :

- `x-amz-bypass-governance-retention`

Corps de la requête

StorageGRID prend en charge tous les paramètres du corps de requête définis par l'API REST Amazon S3 au moment de l'implémentation.

Documentation StorageGRID

"Opérations sur les objets"

"DeleteObjectTagging"

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les objets"

"GetBucketAcl"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"GetBucketCors"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"GetBucketEncryption"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"GetBucketLifecycleConfiguration"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

- "Opérations sur les compartiments"
- "Créer une configuration de cycle de vie S3"

"GetBucketLocation"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"GetBucketNotificationConfiguration"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"GetBucketPolicy"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"GetBucketReplication"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"GetBucketTagging"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"GetBucketVersioning"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"GetObject"

${post_edited_translations.segment}

StorageGRID prend en charge tous les paramètres et en-têtes communs pour cette requête, ainsi que ces paramètres de requête URI supplémentaires :

- `x-amz-checksum-mode`
- `partNumber`
- `response-cache-control`
- `response-content-disposition`

- `response-content-encoding`
- `response-content-language`
- `response-content-type`
- `response-expires`

Et ces en-têtes de requête supplémentaires :

- `Range`
- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`
- `If-Match`
- `If-Modified-Since`
- `If-None-Match`
- `If-Unmodified-Since`

Corps de la requête

Aucune

Documentation StorageGRID

"GetObject"

"GetObjectAcl"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les objets"

"GetObjectLegalHold"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Utilisez l'API REST S3 pour configurer S3 Object Lock"

"GetObjectLockConfiguration"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Utilisez l'API REST S3 pour configurer S3 Object Lock"

"GetObjectRetention"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Utilisez l'API REST S3 pour configurer S3 Object Lock"

"GetObjectTagging"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les objets"

"HeadBucket"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"HeadObject"

${post_edited_translations.segment}

StorageGRID prend en charge toutes paramètres et en-têtes communs pour cette requête, ainsi que les en-têtes supplémentaires suivants :

- `x-amz-checksum-mode`
- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`
- `If-Match`
- `If-Modified-Since`
- `If-None-Match`
- `If-Unmodified-Since`
- `Range`

Corps de la requête

Aucune

Documentation StorageGRID

"HeadObject"

"ListBuckets"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Aucune

Documentation StorageGRID

Opérations sur le service › ListBuckets

"ListMultipartUploads"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette requête, ainsi que les paramètres supplémentaires suivants :

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

Corps de la requête

Aucune

Documentation StorageGRID

"ListMultipartUploads"

"ListObjects"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette requête, ainsi que les paramètres supplémentaires suivants :

- `delimiter`
- `encoding-type`
- `marker`
- `max-keys`
- `prefix`

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"ListObjectsV2"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette requête, ainsi que les paramètres supplémentaires suivants :

- `continuation-token`
- `delimiter`
- `encoding-type`
- `fetch-owner`
- `max-keys`
- `prefix`
- `start-after`

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"ListObjectVersions"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette requête, ainsi que les paramètres supplémentaires suivants :

- `delimiter`

- `encoding-type`
- `key-marker`
- `max-keys`
- `prefix`
- `version-id-marker`

Corps de la requête

Aucune

Documentation StorageGRID

"Opérations sur les compartiments"

"ListParts"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette requête, ainsi que les paramètres supplémentaires suivants :

- `max-parts`
- `part-number-marker`
- `uploadId`

Corps de la requête

Aucune

Documentation StorageGRID

"ListMultipartUploads"

"PutBucketCors"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

StorageGRID prend en charge tous les paramètres du corps de requête définis par l'API REST Amazon S3 au moment de l'implémentation.

Documentation StorageGRID

"Opérations sur les compartiments"

"PutBucketEncryption"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Balises XML du corps de la requête

${post_edited_translations.segment}

- `ApplyServerSideEncryptionByDefault`
- `Rule`
- `ServerSideEncryptionConfiguration`
- `SSEAlgorithm`

Documentation StorageGRID

"Opérations sur les compartiments"

"PutBucketLifecycleConfiguration"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Balises XML du corps de la requête

${post_edited_translations.segment}

- `And`
- `Days`
- `Expiration`
- `ExpiredObjectDeleteMarker`
- `Filter`
- `ID`
- `Key`
- `LifecycleConfiguration`
- `NewerNoncurrentVersions`
- `NoncurrentDays`
- `NoncurrentVersionExpiration`
- `Prefix`
- `Rule`
- `Status`
- `Tag`
- `Value`

Documentation StorageGRID

- "Opérations sur les compartiments"
- "Créer une configuration de cycle de vie S3"

"PutBucketNotificationConfiguration"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Balises XML du corps de la requête

${post_edited_translations.segment}

- `Event`
- `Filter`
- `FilterRule`
- `Id`
- `Name`
- `NotificationConfiguration`
- `Prefix`
- `S3Key`
- `Suffix`
- `Topic`
- `TopicConfiguration`
- `Value`

Documentation StorageGRID

"Opérations sur les compartiments"

"PutBucketPolicy"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Pour plus de détails sur les champs du corps JSON pris en charge, voir "Utilisez les stratégies d'accès aux compartiments et aux groupes".

"PutBucketReplication"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Balises XML du corps de la requête

- `Bucket`
- `Destination`
- `Prefix`
- `ReplicationConfiguration`

- `Rule`
- `Status`
- `StorageClass`

Documentation StorageGRID

"Opérations sur les compartiments"

"PutBucketTagging"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

StorageGRID prend en charge tous les paramètres du corps de requête définis par l'API REST Amazon S3 au moment de l'implémentation.

Documentation StorageGRID

"Opérations sur les compartiments"

"PutBucketVersioning"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Paramètres du corps de la requête

StorageGRID prend en charge les paramètres de corps de requête suivants :

- `VersioningConfiguration`
- `Status`

Documentation StorageGRID

"Opérations sur les compartiments"

"PutObject"

${post_edited_translations.segment}

StorageGRID prend en charge toutes paramètres et en-têtes communs pour cette requête, ainsi que les en-têtes supplémentaires suivants :

- `Cache-Control`
- `Content-Disposition`
- `Content-Encoding`
- `Content-Language`
- `Expires`
- `x-amz-checksum-sha256`
- `x-amz-server-side-encryption`

- `x-amz-storage-class`
- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`
- `x-amz-tagging`
- `x-amz-object-lock-mode`
- `x-amz-object-lock-retain-until-date`
- `x-amz-object-lock-legal-hold`
- `x-amz-meta-<metadata-name>`

Corps de la requête

- Données binaires de l'objet

Documentation StorageGRID

"PutObject"

"PutObjectLegalHold"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

StorageGRID prend en charge tous les paramètres du corps de requête définis par l'API REST Amazon S3 au moment de l'implémentation.

Documentation StorageGRID

"Utilisez l'API REST S3 pour configurer S3 Object Lock"

"PutObjectLockConfiguration"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

StorageGRID prend en charge tous les paramètres du corps de requête définis par l'API REST Amazon S3 au moment de l'implémentation.

Documentation StorageGRID

"Utilisez l'API REST S3 pour configurer S3 Object Lock"

"PutObjectRetention"

${post_edited_translations.segment}

StorageGRID prend en charge tous les paramètres et en-têtes communs pour cette requête, ainsi que cet en-tête supplémentaire :

- `x-amz-bypass-governance-retention`

Corps de la requête

StorageGRID prend en charge tous les paramètres du corps de requête définis par l'API REST Amazon S3 au moment de l'implémentation.

Documentation StorageGRID

"Utilisez l'API REST S3 pour configurer S3 Object Lock"

"PutObjectTagging"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

StorageGRID prend en charge tous les paramètres du corps de requête définis par l'API REST Amazon S3 au moment de l'implémentation.

Documentation StorageGRID

"Opérations sur les objets"

"RestoreObject"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Pour plus de détails sur les champs de corps pris en charge, voir "RestoreObject".

"SelectObjectContent"

${post_edited_translations.segment}

StorageGRID prend en charge toutes les paramètres et en-têtes communs pour cette demande.

Corps de la requête

Pour plus de détails sur les champs du corps pris en charge, consultez les informations suivantes :

- "Utilisez S3 Select"
- "SelectObjectContent"

"UploadPart"

${post_edited_translations.segment}

StorageGRID prend en charge tous les paramètres et en-têtes communs pour cette requête, ainsi que ces paramètres de requête URI supplémentaires :

- `partNumber`
- `uploadId`

Et ces en-têtes de requête supplémentaires :

- `x-amz-checksum-sha256`
- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`

Corps de la requête

- ${post_edited_translations.segment}

Documentation StorageGRID

"UploadPart"

"UploadPartCopy"

${post_edited_translations.segment}

StorageGRID prend en charge tous les paramètres et en-têtes communs pour cette requête, ainsi que ces paramètres de requête URI supplémentaires :

- `partNumber`
- `uploadId`

Et ces en-têtes de requête supplémentaires :

- `x-amz-copy-source`
- `x-amz-copy-source-if-match`
- `x-amz-copy-source-if-modified-since`
- `x-amz-copy-source-if-none-match`
- `x-amz-copy-source-if-unmodified-since`
- `x-amz-copy-source-range`
- `x-amz-server-side-encryption-customer-algorithm`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-key-MD5`
- `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`

Corps de la requête

Aucune

Documentation StorageGRID

"UploadPartCopy"

Tester la configuration de l'API REST S3 de StorageGRID

Vous pouvez utiliser l'interface de ligne de commande Amazon Web Services (AWS CLI) pour tester votre connexion au système et vérifier que vous pouvez lire et écrire des objets.

Avant de commencer

- Vous avez téléchargé et installé l'AWS CLI depuis "aws.amazon.com/cli".
- Vous pouvez éventuellement "a créé un point de terminaison d'équilibrage de charge". Sinon, vous connaissez l'adresse IP du Storage Node auquel vous souhaitez vous connecter ainsi que le numéro de port à utiliser. Voir "Adresses IP et ports pour les connexions client".
- Vous avez "a créé un compte locataire S3".
- Vous vous êtes connecté au locataire et "a créé une clé d'accès".

Pour plus de détails sur ces étapes, voir "Configurer les connexions client".

Étapes

1. Configurez les paramètres de l'interface de ligne de commande AWS (AWS CLI) pour utiliser le compte que vous avez créé dans le système StorageGRID :
   a. Entrez en mode de configuration : `aws configure`
   b. Saisissez l'identifiant de la clé d'accès du compte que vous avez créé.
   c. Saisissez la clé d'accès secrète du compte que vous avez créé.
   d. Saisissez la région par défaut à utiliser. Par exemple, `us-east-1`.
   e. Saisissez le format de sortie par défaut à utiliser, ou appuyez sur **Entrée** pour sélectionner JSON.
2. Créez un compartiment.

   Cet exemple suppose que vous avez configuré un point de terminaison d'équilibrage de charge pour utiliser l'adresse IP 10.96.101.17 et le port 10443.

   ```
   aws s3api --endpoint-url https://10.96.101.17:10443
   --no-verify-ssl create-bucket --bucket testbucket
   ```

   Si le compartiment est créé avec succès, son emplacement est renvoyé, comme illustré dans l'exemple suivant :

   ```
   "Location": "/testbucket"
   ```

3. Téléchargez un objet.

   ```
   aws s3api --endpoint-url https://10.96.101.17:10443 --no-verify-ssl
   put-object --bucket testbucket --key s3.pdf --body C:\s3-
   test\upload\s3.pdf
   ```

Si l'objet est téléchargé avec succès, un Etag est renvoyé, qui est un hachage des données de l'objet.

4. Répertoriez le contenu du bucket pour vérifier que l'objet a été téléchargé.

```
aws s3api --endpoint-url https://10.96.101.17:10443 --no-verify-ssl
list-objects --bucket testbucket
```

5. Supprimez l'objet.

```
aws s3api --endpoint-url https://10.96.101.17:10443 --no-verify-ssl
delete-object --bucket testbucket --key s3.pdf
```

6. Supprimez le compartiment.

```
aws s3api --endpoint-url https://10.96.101.17:10443 --no-verify-ssl
delete-bucket --bucket testbucket
```

Comment StorageGRID implémente l'API REST S3

Comment l'API REST StorageGRID S3 résout les requêtes client conflictuelles

Les requêtes conflictuelles des clients, telles que deux clients écrivant sur la même clé, sont résolues selon le principe du « dernier arrivé, dernier servi ».

Le moment de l'évaluation « latest-wins » est basé sur le moment où le système StorageGRID termine une requête donnée, et non sur le moment où les clients S3 commencent une opération.

Comment l'API REST StorageGRID S3 équilibre la disponibilité et la cohérence

La cohérence assure un équilibre entre la disponibilité des objets et leur cohérence sur différents nœuds de stockage et sites. Vous pouvez modifier la cohérence selon les besoins de votre application.

Par défaut, StorageGRID garantit la cohérence en lecture après écriture pour les objets nouvellement créés. Toute requête GET suivant une requête PUT réussie pourra lire les données nouvellement écrites. Les écrasements d'objets existants, les mises à jour de métadonnées et les suppressions sont finalement cohérents.

Si vous souhaitez effectuer des opérations sur les objets avec un niveau de cohérence différent, vous pouvez :

- Spécifiez une cohérence pour chaque compartiment.
- Spécifiez une consistance pour chaque opération API.
- Modifiez la cohérence globale par défaut de la grille en effectuant l'une des tâches suivantes :
  - Dans le Gestionnaire de grille, accédez à **Configuration** > **Système** > **Paramètres de stockage** > **Cohérence par défaut des compartiments**.

◦ .

> Une modification de la cohérence globale de la grille s'applique uniquement aux compartiments créés après la modification du paramètre. Pour connaître les détails d'une modification, consultez le journal des audits situé à `/var/local/log` (recherchez **consistencyLevel**).

Valeurs de cohérence

La cohérence influe sur la manière dont les métadonnées que StorageGRID utilise pour le suivi des objets sont distribuées entre les nœuds. La cohérence influe sur la disponibilité des objets pour les requêtes client.

Vous pouvez définir la cohérence d'un compartiment ou d'une opération d'API sur l'une des valeurs suivantes :

- **Tous** : Tous les nœuds reçoivent immédiatement les métadonnées de l'objet ou la requête échouera.
- **Forte-globale** : Garantit la cohérence de lecture après écriture pour toutes les requêtes client sur tous les sites. Lorsque la sémantique de quorum est configurée, les comportements suivants s'appliquent :
  - Permet une tolérance aux pannes de site pour les requêtes client lorsque les grilles comportent trois sites ou plus. Les grilles à deux sites n'ont pas de tolérance aux pannes de site.
  - Les opérations S3 suivantes ne pourront pas aboutir si l'un des sites est hors service :
    - DeleteBucketEncryption
    - PutBucketBranch
    - PutBucketEncryption
    - PutBucketVersioning
    - PutObjectLegalHold
    - PutObjectLockConfiguration
    - PutObjectRetention

  Si nécessaire, vous pouvez "Configurer la sémantique de quorum de StorageGRID pour une cohérence globale forte".
- **Strong-site** : les métadonnées des objets sont immédiatement distribuées aux autres nœuds du site. Garantit la cohérence lecture-après-écriture pour toutes les requêtes client au sein du site.
- **Lecture après nouvelle écriture** : Garantit la cohérence en lecture après écriture pour les nouveaux objets et la cohérence éventuelle pour les mises à jour d'objets. Offre une haute disponibilité et des garanties de protection des données. Recommandé dans la plupart des cas.
- **Disponible** : Garantit la cohérence éventuelle pour les nouveaux objets et les mises à jour d'objets. Pour les compartiments S3, utilisez cette fonctionnalité uniquement en cas de besoin (par exemple, pour un compartiment contenant des valeurs de journaux rarement lues, ou pour les opérations HEAD ou GET sur des clés qui n'existent pas). Non pris en charge pour les compartiments S3 FabricPool.

Utilisez la cohérence « Lecture après nouvelle écriture » et « Disponible »

Lorsqu'une opération HEAD ou GET utilise la cohérence « Read-after-new-write », StorageGRID effectue la recherche en plusieurs étapes, comme suit :

- Il commence par rechercher l'objet avec un niveau de cohérence faible.

- Si cette recherche échoue, elle est répétée à la valeur de cohérence suivante jusqu'à atteindre une cohérence équivalente au comportement pour strong-global.

Si une opération HEAD ou GET utilise la cohérence « Lecture après nouvelle écriture » mais que l'objet n'existe pas, la recherche d'objet atteindra toujours un niveau de cohérence équivalent à celui du comportement pour strong-global. Parce que cette cohérence nécessite que plusieurs copies des métadonnées de l'objet soient disponibles sur chaque site, vous pouvez recevoir un nombre élevé d'erreurs 500 Internal Server Error si deux nœuds de stockage ou plus du même site sont indisponibles.

Sauf si vous exigez des garanties de cohérence similaires à celles d'Amazon S3, vous pouvez éviter ces erreurs pour les opérations HEAD et GET en définissant la cohérence sur « Disponible ». Lorsqu'une opération HEAD ou GET utilise la cohérence « Disponible », StorageGRID assure uniquement une cohérence éventuelle. Il ne retente pas une opération ayant échoué en cas de cohérence croissante, il n'est donc pas nécessaire que plusieurs copies des métadonnées de l'objet soient disponibles.

Spécifiez la cohérence pour l'opération API

Pour définir la cohérence d'une opération API individuelle, les valeurs de cohérence doivent être prises en charge pour l'opération, et vous devez spécifier la cohérence dans l'en-tête de la requête. Cet exemple définit la cohérence sur « Strong-site » pour une opération GetObject.

```
GET /bucket/object HTTP/1.1
Date: date
Authorization: authorization name
Host: host
Consistency-Control: strong-site
```

> Vous devez utiliser la même cohérence pour les opérations PutObject et GetObject.

Spécifiez la cohérence du compartiment

Pour définir la cohérence d'un compartiment, vous pouvez utiliser la requête "Cohérence PUT Bucket" StorageGRID. Ou vous pouvez "modifier la cohérence d'un bucket" depuis le Tenant Manager.

Lors du réglage de la consistance d'un bucket, tenez compte des points suivants :

- La configuration de la cohérence d'un compartiment détermine le type de cohérence utilisé pour les opérations S3 effectuées sur les objets contenus dans le compartiment ou sur la configuration du compartiment. Elle n'affecte pas les opérations sur le compartiment lui-même.
- La cohérence d'une opération API individuelle prime sur la cohérence du compartiment.
- En règle générale, les buckets doivent utiliser la cohérence par défaut « Lecture après nouvelle écriture ». Si les requêtes ne fonctionnent pas correctement, modifiez le comportement du client applicatif si possible. Ou, configurez le client pour spécifier la cohérence pour chaque requête API. Définissez la cohérence au niveau du bucket uniquement en dernier recours.

Comment les règles de cohérence et les règles ILM interagissent pour affecter la protection des données

Le niveau de cohérence choisi et la règle ILM influent tous deux sur la manière dont les objets sont protégés. Ces paramètres peuvent interagir.

Par exemple, la cohérence utilisée lors du stockage d'un objet influe sur l'emplacement initial de ses métadonnées, tandis que le comportement d'ingestion sélectionné pour la règle ILM influe sur l'emplacement initial des copies de l'objet. Parce que StorageGRID nécessite l'accès à la fois aux métadonnées d'un objet et à ses données pour répondre aux requêtes client, le choix de niveaux de protection identiques pour la cohérence et le comportement d'ingestion peut offrir une meilleure protection des données initiale et des réponses système plus prévisibles.

Les "options d'ingestion"règles ILM suivantes sont disponibles :

Double engagement
: StorageGRID crée immédiatement des copies intermédiaires de l'objet et renvoie la confirmation au client. Les copies spécifiées dans la règle ILM sont effectuées lorsque cela est possible.

Strict
: Toutes les copies spécifiées dans la règle ILM doivent être effectuées avant que la réussite ne soit renvoyée au client.

Équilibré
: StorageGRID tente de créer toutes les copies spécifiées dans la règle ILM lors de l'ingestion ; si cela s'avère impossible, des copies intermédiaires sont créées et un message de succès est renvoyé au client. Les copies spécifiées dans la règle ILM sont créées dès que possible.

Exemple de la façon dont la cohérence et la règle ILM peuvent interagir

Supposons que vous ayez une grille à trois sites avec la règle ILM suivante et la cohérence suivante :

- **Règle ILM** : Créez trois copies de l'objet, une sur le site local et une sur chaque site distant. Utilisez le comportement d'ingestion strict.
- **Cohérence** : Forte et globale (les métadonnées de l'objet sont immédiatement distribuées à plusieurs sites).

Lorsqu'un client stocke un objet dans la grille, StorageGRID effectue trois copies de l'objet et distribue les métadonnées à plusieurs sites avant de renvoyer un message de succès au client.

L'objet est intégralement protégé contre toute perte dès la confirmation de son ingestion. Par exemple, si le site local est perdu peu après l'ingestion, des copies des données de l'objet et des métadonnées de l'objet existent toujours sur les sites distants. L'objet est entièrement récupérable depuis les autres sites.

Si vous utilisiez la même règle ILM et la cohérence forte des sites, le client pourrait recevoir un message de succès après la réplication des données d'objet sur les sites distants, mais avant la distribution des métadonnées d'objet. Dans ce cas, le niveau de protection des métadonnées d'objet ne correspond pas au niveau de protection des données d'objet. Si le site local est perdu peu après l'ingestion, les métadonnées d'objet sont perdues. L'objet ne peut pas être récupéré.

L'interaction entre la cohérence et les règles ILM peut être complexe. Contactez NetApp si vous avez besoin d'assistance.

Comment StorageGRID utilise le versionnage des objets

Vous pouvez configurer le versionnage d'un compartiment si vous souhaitez conserver plusieurs versions de chaque objet. L'activation du versionnage pour un compartiment peut vous aider à vous prémunir contre la suppression accidentelle d'objets et vous permet de récupérer et de restaurer des versions antérieures d'un objet.

Le système StorageGRID implémente le versionnage et prend en charge la plupart des fonctionnalités, avec toutefois certaines limitations. StorageGRID prend en charge jusqu'à 10 000 versions de chaque objet.

Le versionnage des objets peut être combiné avec la gestion du cycle de vie des informations (ILM) de StorageGRID ou avec la configuration du cycle de vie des compartiments S3. Vous devez activer explicitement le versionnage pour chaque compartiment. Lorsque le versionnage est activé pour un compartiment, chaque objet ajouté au compartiment se voit attribuer un ID de version, qui est généré par le système StorageGRID.

L'utilisation de la suppression MFA (authentification multifacteur) n'est pas prise en charge.

> Le versionnage ne peut être activé que sur les compartiments créés avec StorageGRID version 10.3 ou ultérieure.

ILM et gestion des versions

Les politiques ILM sont appliquées à chaque version d'un objet. Un processus d'analyse ILM examine en continu tous les objets et les réévalue par rapport à la politique ILM en vigueur. Toute modification apportée aux politiques ILM est appliquée à tous les objets précédemment importés. Cela inclut les versions précédemment importées si le versionnage est activé. L'analyse ILM applique les nouvelles modifications ILM aux objets précédemment importés.

Pour les objets S3 dans des compartiments avec versionnage activé, la prise en charge du versionnage vous permet de créer des règles ILM qui utilisent « Heure non actuelle » comme heure de référence (sélectionnez **Oui** à la question « Appliquer cette règle uniquement aux anciennes versions de l'objet ? » dans "Étape 1 de l'assistant de création d'une règle ILM"). Lorsqu'un objet est mis à jour, ses versions précédentes deviennent non actuelles. L'utilisation d'un filtre « Heure non actuelle » vous permet de créer des stratégies qui réduisent l'impact sur le stockage des versions précédentes des objets.

> Lorsque vous chargez une nouvelle version d'un objet à l'aide d'une opération de chargement en plusieurs parties, l'horodatage de non-actualité de la version originale de l'objet correspond au moment où le chargement en plusieurs parties a été créé pour la nouvelle version, et non au moment où le chargement en plusieurs parties a été terminé. Dans certains cas limités, l'horodatage de non-actualité de la version originale peut être antérieur de plusieurs heures, voire de plusieurs jours, à celui de la version actuelle.

Informations connexes

- "${post_edited_translations.segment}"
- "Règles et politiques ILM pour les objets versionnés S3 (Exemple 4)".

Utilisez l'API REST S3 pour configurer StorageGRID S3 Object Lock

Si le paramètre global S3 Object Lock est activé pour votre système StorageGRID, vous pouvez créer des compartiments avec S3 Object Lock activé. Vous pouvez spécifier une rétention par défaut pour chaque compartiment ou des paramètres de rétention pour chaque version d'objet.

${post_edited_translations.segment}

Si le paramètre de verrouillage global des objets S3 est activé pour votre système StorageGRID, vous pouvez éventuellement activer le verrouillage des objets S3 lors de la création de chaque compartiment.

Le verrouillage d'objet S3 est un paramètre permanent qui ne peut être activé que lors de la création d'un

compartiment. Vous ne pouvez pas ajouter ni désactiver le verrouillage d'objet S3 après la création d'un compartiment.

${post_edited_translations.segment}

- Créez le compartiment à l'aide du Tenant Manager. Voir "Créer un compartiment S3".
- Créez le compartiment à l'aide d'une requête CreateBucket avec l' `x-amz-bucket-object-lock-enabled` en-tête de requête. Voir "Opérations sur les compartiments".

S3 Object Lock requiert le versionnement du compartiment, qui est activé automatiquement lors de la création du compartiment. Vous ne pouvez pas suspendre le versionnement pour le compartiment. Voir "Versionnage d'objets".

${post_edited_translations.segment}

Lorsque le verrouillage d'objet S3 est activé pour un compartiment, vous pouvez éventuellement activer la rétention par défaut pour le compartiment et spécifier un mode de rétention par défaut et une période de rétention par défaut.

Mode de rétention par défaut

- En mode CONFORMITÉ :
  - L'objet ne peut pas être supprimé tant que sa date de conservation n'est pas atteinte.
  - La date de conservation de l'objet peut être augmentée, mais elle ne peut pas être diminuée.
  - La date de conservation de l'objet ne peut pas être supprimée avant que cette date ne soit atteinte.
- En mode GOUVERNANCE :
  - Les utilisateurs disposant de l' `s3:BypassGovernanceRetention`autorisation peuvent utiliser l' `x-amz-bypass-governance-retention: true`en-tête de requête pour contourner les paramètres de rétention.
  - Ces utilisateurs peuvent supprimer une version d'objet avant que sa date de conservation ne soit atteinte.
  - Ces utilisateurs peuvent augmenter, diminuer ou supprimer la date de conservation d'un objet.

Période de conservation par défaut

${post_edited_translations.segment}

Comment définir la rétention par défaut pour un compartiment

${post_edited_translations.segment}

- Gérez les paramètres du compartiment depuis le Gestionnaire de locataires. Voir "Créez un compartiment S3" et "Mettre à jour la rétention par défaut du verrouillage d'objet S3".
- Envoyez une requête PutObjectLockConfiguration pour le compartiment afin de spécifier le mode par défaut et le nombre de jours ou d'années par défaut.

PutObjectLockConfiguration

La requête PutObjectLockConfiguration vous permet de définir et de modifier le mode de rétention par défaut et la période de rétention par défaut pour un compartiment ayant le verrouillage d'objets S3 activé. Vous pouvez également supprimer les paramètres de rétention par défaut précédemment configurés.

Lors de l'ingestion de nouvelles versions d'objets dans le compartiment, le mode de rétention par défaut est appliqué si `x-amz-object-lock-mode` et `x-amz-object-lock-retain-until-date` ne sont pas spécifiés. La période de rétention par défaut est utilisée pour calculer la date limite de rétention si `x-amz-object-lock-retain-until-date` n'est pas spécifié.

Si la période de conservation par défaut est modifiée après l'ingestion d'une version d'objet, la date de fin de conservation de la version d'objet reste la même et n'est pas recalculée en utilisant la nouvelle période de conservation par défaut.

Vous devez disposer de l' `s3:PutBucketObjectLockConfiguration`autorisation, ou être account root, pour effectuer cette opération.

L' `Content-MD5`en-tête de requête doit être spécifié dans la requête PUT.

Exemple de requête

Cet exemple active le verrouillage d'objet S3 pour un compartiment et définit le mode de rétention par défaut sur COMPLIANCE et la période de rétention par défaut sur 6 ans.

```
PUT /bucket?object-lock HTTP/1.1
Accept-Encoding: identity
Content-Length: 308
Host: host
Content-MD5: request header
User-Agent: s3sign/1.0.0 requests/2.24.0 python/3.8.2
X-Amz-Date: date
X-Amz-Content-SHA256: authorization-string
Authorization: authorization-string

<ObjectLockConfiguration>
    <ObjectLockEnabled>Enabled</ObjectLockEnabled>
    <Rule>
        <DefaultRetention>
            <Mode>COMPLIANCE</Mode>
            <Years>6</Years>
        </DefaultRetention>
    </Rule>
</ObjectLockConfiguration>
```

${post_edited_translations.segment}

Pour déterminer si le verrouillage d'objet S3 est activé pour un compartiment et pour afficher le mode de rétention et la période de rétention par défaut, utilisez l'une de ces méthodes :

- Consultez le compartiment dans le Gestionnaire de locataires. Voir "Afficher les compartiments S3".
- Envoyez une demande GetObjectLockConfiguration.

GetObjectLockConfiguration

La requête GetObjectLockConfiguration vous permet de déterminer si S3 Object Lock est activé pour un compartiment et, s'il est activé, de voir si un mode de rétention et une période de rétention par défaut sont configurés pour le compartiment.

Lorsque de nouvelles versions d'objets sont ingérées dans le compartiment, le mode de rétention par défaut est appliqué si `x-amz-object-lock-mode` n'est pas spécifié. La période de rétention par défaut est utilisée pour calculer la date limite de conservation si `x-amz-object-lock-retain-until-date` n'est pas spécifié.

Vous devez disposer de l `s3:GetBucketObjectLockConfiguration`autorisation, ou être account root, pour effectuer cette opération.

Exemple de requête

```
GET /bucket?object-lock HTTP/1.1
Host: host
Accept-Encoding: identity
User-Agent: aws-cli/1.18.106 Python/3.8.2 Linux/4.4.0-18362-Microsoft
botocore/1.17.29
x-amz-date: date
x-amz-content-sha256: authorization-string
Authorization: authorization-string
```

Exemple de réponse

```
HTTP/1.1 200 OK
x-amz-id-2:
iVmcB7OXXJRkRH1FiVq1151/T24gRfpwpuZrEG11Bb9ImOMAAe98oxSpXlknabA0LTvBYJpSIX
k=
x-amz-request-id: B34E94CACB2CEF6D
Date: Fri, 04 Sep 2020 22:47:09 GMT
Transfer-Encoding: chunked
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>
<ObjectLockConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
    <ObjectLockEnabled>Enabled</ObjectLockEnabled>
    <Rule>
        <DefaultRetention>
            <Mode>COMPLIANCE</Mode>
            <Years>6</Years>
        </DefaultRetention>
    </Rule>
</ObjectLockConfiguration>
```

${post_edited_translations.segment}

Un compartiment avec le verrouillage d'objets S3 activé peut contenir une combinaison d'objets avec et sans paramètres de rétention de verrouillage d'objets S3.

Les paramètres de rétention au niveau de l'objet sont définis à l'aide de l'API REST S3. Les paramètres de rétention d'un objet remplacent tous les paramètres de rétention par défaut du compartiment.

Vous pouvez spécifier les paramètres suivants pour chaque objet :

- **Mode de rétention** : soit COMPLIANCE, soit GOVERNANCE.
- **Retain-until-date** : une date spécifiant la durée pendant laquelle la version de l'objet doit être conservée par StorageGRID.
  - En mode COMPLIANCE, si la date de conservation est dans le futur, l'objet peut être récupéré, mais il ne peut pas être modifié ou supprimé. La date de conservation peut être augmentée, mais cette date ne peut pas être diminuée ou supprimée.
  - En mode GOVERNANCE, les utilisateurs disposant d'une autorisation spéciale peuvent contourner le paramètre retain-until-date. Ils peuvent supprimer une version d'objet avant l'expiration de sa période de conservation. Ils peuvent également augmenter, diminuer ou même supprimer le retain-until-date.
- **Conservation légale** : L'application d'une conservation légale à une version d'objet verrouille immédiatement cet objet. Par exemple, vous pourriez avoir besoin d'appliquer une conservation légale à un objet lié à une enquête ou à un litige. Une conservation légale n'a pas de date d'expiration, mais reste en place jusqu'à ce qu'elle soit explicitement levée.

  Le paramètre de conservation légale d'un objet est indépendant du mode de conservation et de la date limite de conservation. Si une version d'un objet est soumise à une conservation légale, personne ne peut supprimer cette version.

Pour spécifier les paramètres de S3 Object Lock lors de l'ajout d'une version d'objet à un compartiment, émettez une requête "PutObject", "CopyObject" ou "CreateMultipartUpload".

${post_edited_translations.segment}

- `x-amz-object-lock-mode`, qui peut être COMPLIANCE ou GOVERNANCE (sensible à la casse).

  > Si vous spécifiez `x-amz-object-lock-mode`, vous devez également spécifier `x-amz-object-lock-retain-until-date`.

- `x-amz-object-lock-retain-until-date`
  - La valeur de conservation jusqu'à la date doit être au format `2020-08-10T21:46:00Z`. Les fractions de seconde sont autorisées, mais seules trois décimales sont conservées (précision à la milliseconde). Les autres formats ISO 8601 ne sont pas autorisés.
  - ${post_edited_translations.segment}
- `x-amz-object-lock-legal-hold`

  Si la conservation légale est ON (sensible à la casse), l'objet est placé sous conservation légale. Si la conservation légale est OFF, aucune conservation légale n'est appliquée. Toute autre valeur entraîne une erreur 400 Bad Request (InvalidArgument).

Si vous utilisez l'un de ces en-têtes de requête, tenez compte des restrictions suivantes :

- L' `Content-MD5`en-tête de requête est requis si un `x-amz-object-lock-*`en-tête de requête est présent dans la requête PutObject. `Content-MD5`n'est pas requis pour CopyObject ou CreateMultipartUpload.
- Si le compartiment n'a pas activé S3 Object Lock et qu'un en-tête de requête `x-amz-object-lock-*` est présent, une erreur 400 Bad Request (InvalidRequest) est renvoyée.
- La requête PutObject prend en charge l'utilisation de `x-amz-storage-class: REDUCED_REDUNDANCY` pour correspondre au comportement d'AWS. Toutefois, lorsqu'un objet est ingéré dans un compartiment avec S3 Object Lock activé, StorageGRID effectue toujours une ingestion dual-commit.
- Une réponse GET ou HeadObject de version ultérieure inclura les en-têtes `x-amz-object-lock-mode`, `x-amz-object-lock-retain-until-date` et `x-amz-object-lock-legal-hold`, s'ils sont configurés et si l'expéditeur de la requête dispose des autorisations `s3:Get*` correctes.

Vous pouvez utiliser la clé de condition de politique `s3:object-lock-remaining-retention-days` pour limiter les périodes de rétention minimales et maximales autorisées pour vos objets.

${post_edited_translations.segment}

${post_edited_translations.segment}

- `PutObjectLegalHold`

  Si la nouvelle valeur de conservation légale est ON, l'objet est placé sous conservation légale. Si la valeur de conservation légale est OFF, la conservation légale est levée.

- `PutObjectRetention`
  - La valeur du mode peut être COMPLIANCE ou GOVERNANCE (sensible à la casse).
  - La valeur de conservation jusqu'à la date doit être au format `2020-08-10T21:46:00Z`. Les fractions de seconde sont autorisées, mais seules trois décimales sont conservées (précision à la milliseconde). Les autres formats ISO 8601 ne sont pas autorisés.
  - Si une version d'objet possède déjà une date de conservation (retain-until), vous pouvez uniquement l'augmenter. La nouvelle valeur doit être dans le futur.

Comment utiliser le mode GOVERNANCE

Les utilisateurs qui disposent de l'autorisation `s3:BypassGovernanceRetention` peuvent contourner les paramètres de rétention active d'un objet qui utilise le mode GOVERNANCE. Toutes les opérations DELETE ou PutObjectRetention doivent inclure l'en-tête de requête `x-amz-bypass-governance-retention:true`. Ces utilisateurs peuvent effectuer ces opérations supplémentaires :

- Exécutez les opérations DeleteObject ou DeleteObjects pour supprimer une version d'objet avant que sa période de rétention ne soit écoulée.

  Les objets soumis à une conservation légale ne peuvent pas être supprimés. La conservation légale doit être désactivée.

- Effectuez des opérations PutObjectRetention qui modifient le mode d'une version d'objet de GOVERNANCE à COMPLIANCE avant que la période de rétention de l'objet ne soit écoulée.

  Il est strictement interdit de passer du mode COMPLIANCE au mode GOVERNANCE.

- Effectuez des opérations PutObjectRetention pour augmenter, diminuer ou supprimer la période de conservation d'une version d'objet.

Informations connexes

- "Gérez les objets avec S3 Object Lock"
- "${post_edited_translations.segment}"
- "Guide de l'utilisateur Amazon Simple Storage Service : Verrouillage des objets"

Découvrez la configuration du cycle de vie S3 pour StorageGRID

Vous pouvez créer une configuration de cycle de vie S3 pour contrôler le moment où des objets spécifiques sont supprimés du système StorageGRID.

L'exemple simple présenté dans cette section illustre comment une configuration de cycle de vie S3 peut contrôler la suppression (expiration) de certains objets dans des compartiments S3 spécifiques. L'exemple de cette section est fourni uniquement à des fins d'illustration. Pour des informations complètes sur la création de configurations de cycle de vie S3, consultez "Guide de l'utilisateur Amazon Simple Storage Service : gestion du cycle de vie des objets". Notez que StorageGRID prend uniquement en charge les actions d'expiration ; il ne prend pas en charge les actions de transition.

Ce qu'est la configuration du cycle de vie

Une configuration de cycle de vie est un ensemble de règles appliquées aux objets dans des compartiments S3 spécifiques. Chaque règle spécifie quels objets sont concernés et quand ces objets expireront (à une date précise ou après un certain nombre de jours).

StorageGRID prend en charge jusqu'à 1 000 règles de cycle de vie dans une configuration de cycle de vie. Chaque règle peut inclure les éléments XML suivants :

- Expiration : Supprimer un objet lorsqu'une date spécifiée est atteinte ou lorsqu'un nombre de jours spécifié est atteint, à compter de la date d'ingestion de l'objet.
- NoncurrentVersionExpiration : Supprimez un objet lorsqu'un nombre de jours spécifié est atteint, à compter du moment où l'objet est devenu non actuel.
- Filtrer (Préfixe, Tag)
- Statut
- ID

Chaque objet suit les paramètres de rétention définis soit par le cycle de vie d'un compartiment S3, soit par une stratégie ILM. Lorsqu'un cycle de vie de compartiment S3 est configuré, les actions d'expiration de ce cycle de vie prévalent sur la stratégie ILM pour les objets correspondant au filtre de cycle de vie du compartiment. Les objets qui ne correspondent pas au filtre de cycle de vie du compartiment utilisent les paramètres de rétention de la stratégie ILM. Si un objet correspond à un filtre de cycle de vie de compartiment et qu'aucune action d'expiration n'est explicitement spécifiée, les paramètres de rétention de la stratégie ILM ne sont pas utilisés et il est sous-entendu que les versions de l'objet sont conservées indéfiniment. Voir "Exemples de priorités pour le cycle de vie d'un compartiment S3 et la politique ILM".

Par conséquent, un objet peut être retiré de la grille même si les instructions de placement d'une règle ILM s'appliquent toujours à cet objet. Ou, un objet peut être conservé sur la grille même après l'expiration de toutes les instructions de placement ILM le concernant. Pour plus de détails, voir "Comment l'ILM fonctionne tout au long du cycle de vie d'un objet".

La configuration du cycle de vie des compartiments peut être utilisée avec les compartiments dont le S3 Object Lock est activé, mais la configuration du cycle de vie des compartiments n'est pas prise en charge pour les anciens compartiments Compliant.

StorageGRID prend en charge l'utilisation des opérations de compartiment suivantes pour gérer les configurations de cycle de vie :

- DeleteBucketLifecycle
- GetBucketLifecycleConfiguration
- PutBucketLifecycleConfiguration

Créer une configuration de cycle de vie

La première étape de la création d'une configuration de cycle de vie consiste à créer un fichier JSON contenant une ou plusieurs règles. Par exemple, ce fichier JSON contient trois règles, comme suit :

1. La règle 1 s'applique uniquement aux objets qui correspondent au préfixe `category1/` et qui ont une valeur de `key2 tag2`. Le paramètre `Expiration` spécifie que les objets correspondant au filtre expireront à minuit le 22 août 2020.
2. La règle 2 s'applique uniquement aux objets qui correspondent au préfixe `category2/`. Le `Expiration` paramètre spécifie que les objets correspondant au filtre expireront 100 jours après leur ingestion.

   Les règles qui spécifient un nombre de jours sont relatives à la date d'ingestion de l'objet. Si la date actuelle est postérieure à la date d'ingestion plus le nombre de jours, certains objets peuvent être supprimés du compartiment dès l'application de la configuration du cycle de vie.

3. La règle 3 s'applique uniquement aux objets qui correspondent au préfixe `category3/`. Le `Expiration` paramètre spécifie que toutes les versions non actuelles des objets correspondants expireront 50 jours après qu'elles soient devenues non actuelles.

```
{
    "Rules": [
        {
            "ID": "rule1",
            "Filter": {
                "And": {
                    "Prefix": "category1/",
                    "Tags": [
                        {
                            "Key": "key2",
                            "Value": "tag2"
                        }
                    ]
                }
            },
            "Expiration": {
                "Date": "2020-08-22T00:00:00Z"
            },
            "Status": "Enabled"
        },
        {
            "ID": "rule2",
            "Filter": {
                "Prefix": "category2/"
            },
            "Expiration": {
                "Days": 100
            },
            "Status": "Enabled"
        },
        {
            "ID": "rule3",
            "Filter": {
                "Prefix": "category3/"
            },
            "NoncurrentVersionExpiration": {
                "NoncurrentDays": 50
            },
            "Status": "Enabled"
        }
    ]
}
```

Appliquer la configuration du cycle de vie au compartiment

Après avoir créé le fichier de configuration du cycle de vie, vous l'appliquez à un compartiment en émettant une requête PutBucketLifecycleConfiguration.

Cette requête applique la configuration de cycle de vie du fichier d'exemple aux objets d'un compartiment nommé `testbucket`.

```
aws s3api --endpoint-url <StorageGRID endpoint> put-bucket-lifecycle-
configuration
--bucket testbucket --lifecycle-configuration file://bktjson.json
```

Pour vérifier qu'une configuration de cycle de vie a bien été appliquée au compartiment, envoyez une requête GetBucketLifecycleConfiguration. Par exemple :

```
aws s3api --endpoint-url <StorageGRID endpoint> get-bucket-lifecycle-
configuration
 --bucket testbucket
```

Une réponse réussie répertorie la configuration du cycle de vie que vous venez d'appliquer.

Vérifiez que l'expiration du cycle de vie du compartiment s'applique à l'objet

Vous pouvez déterminer si une règle d'expiration dans la configuration du cycle de vie s'applique à un objet spécifique lors de l'émission d'une requête PutObject, HeadObject ou GetObject. Si une règle s'applique, la réponse inclut un `Expiration` paramètre qui indique quand l'objet expire et quelle règle d'expiration a été appliquée.

> Étant donné que le cycle de vie du bucket remplace ILM, la `expiry-date`date affichée correspond à la date réelle à laquelle l'objet sera supprimé. Pour plus de détails, consultez "Comment la rétention d'objet est déterminée".

Par exemple, cette requête PutObject a été émise le 22 juin 2020 et place un objet dans le `testbucket` bucket.

```
aws s3api --endpoint-url <StorageGRID endpoint> put-object
--bucket testbucket --key obj2test2 --body bktjson.json
```

La réponse positive indique que l'objet expirera dans 100 jours (01 Oct 2020) et qu'il correspond à la règle 2 de la configuration du cycle de vie.

```
{
      *"Expiration": "expiry-date=\"Thu, 01 Oct 2020 09:07:49 GMT\", rule-
id=\"rule2\"",
      "ETag": "\"9762f8a803bc34f5340579d4446076f7\""
}
```

Par exemple, cette requête HeadObject a été utilisée pour obtenir les métadonnées du même objet dans le compartiment testbucket.

```
aws s3api --endpoint-url <StorageGRID endpoint> head-object
--bucket testbucket --key obj2test2
```

La réponse de succès inclut les métadonnées de l'objet et indique que l'objet expirera dans 100 jours et qu'il correspond à la Rule 2.

```
{
      "AcceptRanges": "bytes",
      *"Expiration": "expiry-date=\"Thu, 01 Oct 2020 09:07:48 GMT\", rule-
id=\"rule2\"",
      "LastModified": "2020-06-23T09:07:48+00:00",
      "ContentLength": 921,
      "ETag": "\"9762f8a803bc34f5340579d4446076f7\""
      "ContentType": "binary/octet-stream",
      "Metadata": {}
}
```

Pour les compartiments compatibles avec le versionnage, l' `x-amz-expiration`en-tête de réponse s'applique uniquement aux versions actuelles des objets.

Recommandations pour la mise en œuvre de l'API REST S3 avec StorageGRID

${post_edited_translations.segment}

${post_edited_translations.segment}

Si votre application vérifie régulièrement si un objet existe dans un chemin où vous ne vous attendez pas à ce que l'objet existe réellement, vous devez utiliser le niveau « Available » "cohérence". Par exemple, vous devez utiliser la cohérence « Available » si votre application effectue une requête HEAD sur un emplacement avant d'y effectuer un PUT.

Sinon, si l'opération HEAD ne trouve pas l'objet, vous pourriez recevoir un grand nombre d'erreurs 500 Internal Server errors si deux ou plusieurs Storage Nodes du même site sont indisponibles ou si un site distant est inaccessible.

Vous pouvez définir la cohérence « Disponible » pour chaque compartiment à l'aide de la requête "Cohérence PUT Bucket", ou vous pouvez spécifier la cohérence dans l'en-tête de requête pour une opération API

individuelle.

Recommandations concernant les clés d'objet

Suivez ces recommandations pour les noms de clés d'objet, en fonction du moment où le compartiment a été créé pour la première fois.

Compartiments créés dans StorageGRID 11.4 ou une version antérieure

- N'utilisez pas de valeurs aléatoires comme premiers quatre caractères des clés d'objet. Ceci contraste avec l'ancienne recommandation d'AWS concernant les préfixes de clés. Utilisez plutôt des préfixes non aléatoires et non uniques, tels que `image`.
- Si vous suivez l'ancienne recommandation d'AWS d'utiliser des caractères aléatoires et uniques dans les préfixes de clés, préfixez les clés d'objet avec un nom de répertoire. C'est-à-dire, utilisez ce format :

  `mybucket/mydir/f8e3-image3132.jpg`

  Au lieu de ce format :

  `mybucket/f8e3-image3132.jpg`

${post_edited_translations.segment}

Il n'est pas nécessaire de restreindre les noms de clés d'objets pour respecter les bonnes pratiques en matière de performances. Dans la plupart des cas, vous pouvez utiliser des valeurs aléatoires pour les quatre premiers caractères des noms de clés d'objets.

> Une exception à cela est une charge de travail S3 qui supprime continuellement tous les objets après une courte période. Pour minimiser l'impact sur les performances pour ce cas d'utilisation, faites varier la partie initiale du nom de la clé tous les quelques milliers d'objets avec un élément tel que la date. Par exemple, supposez qu'un client S3 écrive généralement 2 000 objets/seconde et que la politique ILM ou de cycle de vie du compartiment supprime tous les objets après trois jours. Pour minimiser l'impact sur les performances, vous pouvez nommer les clés en utilisant un modèle comme celui-ci : `/mybucket/mydir/`*`yyyymmddhhmmss-random_UUID`*`.jpg`

${post_edited_translations.segment}

Si la "option globale pour compresser les objets stockés" est activée, les applications clientes S3 doivent éviter d'effectuer des opérations GetObject qui spécifient le retour d'une plage d'octets. Ces opérations de « lecture par plage » sont inefficaces car StorageGRID doit effectivement décompresser les objets pour accéder aux octets demandés. Les opérations GetObject qui demandent une petite plage d'octets à partir d'un objet très volumineux sont particulièrement inefficaces ; par exemple, il est inefficace de lire une plage de 10 Mo à partir d'un objet compressé de 50 Go.

Si les plages sont lues à partir d'objets compressés, les requêtes client peuvent expirer.

> Si vous devez compresser des objets et que votre application cliente doit utiliser des lectures par plage, augmentez le délai d'expiration de lecture pour l'application.

${post_edited_translations.segment}

Opérations prises en charge et limitations de l'API REST S3 dans StorageGRID

Le système StorageGRID implémente l'API Simple Storage Service (version 2006-03-01) et prend en charge la plupart des opérations, avec toutefois certaines limitations. Vous devez comprendre les détails d'implémentation lorsque vous intégrez des applications clientes de l'API REST S3.

${post_edited_translations.segment}

${post_edited_translations.segment}

${post_edited_translations.segment}

Le système StorageGRID prend en charge uniquement les formats de date HTTP valides pour tous les en-têtes qui acceptent des valeurs de date. La partie heure de la date peut être spécifiée au format GMT (Greenwich Mean Time) ou au format UTC (Universal Coordinated Time) sans décalage de fuseau horaire (+0000 doit être spécifié). Si vous incluez l'en-tête `x-amz-date` dans votre requête, il remplace toute valeur spécifiée dans l'en-tête de requête Date. Lors de l'utilisation d'AWS Signature Version 4, l'en-tête `x-amz-date` doit être présent dans la requête signée car l'en-tête de date n'est pas pris en charge.

En-têtes de requête courants

Le système StorageGRID prend en charge les en-têtes de requête communs définis par "Référence de l'API Amazon Simple Storage Service : En-têtes de requête courants", à une exception près.

En-tête de requête	Mise en œuvre
Autorisation	${post_edited_translations.segment} ${post_edited_translations.segment} • Lorsque vous fournissez la valeur réelle de la somme de contrôle de la charge utile dans `x-amz-content-sha256`, la valeur est acceptée sans validation, comme si la valeur `UNSIGNED-PAYLOAD` avait été fournie pour l'en-tête. Lorsque vous fournissez une valeur d'en-tête `x-amz-content-sha256` qui implique `aws-chunked` le streaming (par exemple, STREAMING-AWS4-HMAC-SHA256-PAYLOAD), les signatures de tronçon ne sont pas vérifiées par rapport aux données de tronçon.

${post_edited_translations.segment}

Le système StorageGRID prend en charge tous les en-têtes de réponse courants définis par la *Référence de l'API Simple Storage Service*, à une exception près.

${post_edited_translations.segment}	Mise en œuvre
${post_edited_translations.segment}	${post_edited_translations.segment}

Comment l'API REST StorageGRID S3 authentifie les requêtes

Le système StorageGRID prend en charge l'accès authentifié et l'accès anonyme aux objets via l'API S3.

L'API S3 prend en charge la Signature version 2 et la Signature version 4 pour l'authentification des requêtes API S3.

Les requêtes authentifiées doivent être signées à l'aide de votre identifiant de clé d'accès et de votre clé d'accès secrète.

Le système StorageGRID prend en charge deux méthodes d'authentification : l'en-tête HTTP `Authorization` et l'utilisation des paramètres de requête.

Utilisez l'en-tête d'autorisation HTTP

L' `Authorization`en-tête HTTP est utilisé par toutes les opérations de l'API S3, à l'exception des requêtes anonymes autorisées par la politique du compartiment. L' `Authorization`en-tête contient toutes les informations de signature requises pour authentifier une requête.

Utilisez les paramètres de requête

Vous pouvez utiliser des paramètres de requête pour ajouter des informations d'authentification à une URL. Cela s'appelle la pré-signature de l'URL, ce qui peut être utilisé pour accorder un accès temporaire à des ressources spécifiques. Les utilisateurs disposant de l'URL pré-signée n'ont pas besoin de connaître la clé d'accès secrète pour accéder à la ressource, ce qui vous permet de fournir à des tiers un accès restreint à une ressource.

Opérations de service prises en charge dans StorageGRID

Le système StorageGRID prend en charge les opérations suivantes sur le service.

Opération	Mise en œuvre
ListBuckets (anciennement nommé GET Service)	Implémenté avec l'ensemble du comportement de l'API REST Amazon S3. Sous réserve de modifications sans préavis.
Obtenir l'utilisation du stockage	La requête "Obtenir l'utilisation du stockage" StorageGRID indique la capacité totale de stockage utilisée par un compte, ainsi que pour chaque compartiment associé au compte. Il s'agit d'une opération sur le service avec un chemin de / et un paramètre de requête personnalisé (`?x-ntap-sg-usage` ajouté.
OPTIONS /	Les applications clientes peuvent envoyer `OPTIONS /`des requêtes au port S3 d'un nœud de stockage, sans fournir de paramètres d'authentification S3, afin de déterminer si le nœud de stockage est disponible. Vous pouvez utiliser cette requête pour la surveillance ou pour permettre aux équilibreurs de charge externes d'identifier lorsqu'un nœud de stockage est hors service.

Opérations sur les compartiments S3 et détails d'implémentation dans StorageGRID

Le système StorageGRID prend en charge un maximum de 5 000 compartiments pour chaque compte locataire S3.

Chaque grille peut comporter un maximum de 100 000 compartiments.

Pour prendre en charge 5 000 compartiments, chaque nœud de stockage de la grille doit disposer d'un minimum de 64 Go de RAM.

Les restrictions relatives aux noms de compartiments suivent les restrictions de la région standard AWS US, mais vous devez les restreindre davantage aux conventions de nommage DNS pour prendre en charge les requêtes de type hébergement virtuel S3.

Pour plus d'informations, veuillez consulter les documents suivants :

- "Guide de l'utilisateur d'Amazon Simple Storage Service : Quotas, restrictions et limitations des compartiments"
- "Configurer les noms de domaine des points de terminaison S3"

Les opérations ListObjects (GET Bucket) et ListObjectVersions (GET Bucket object versions) prennent en charge StorageGRID "valeurs de cohérence".

Vous pouvez vérifier si les mises à jour du temps d'accès sont activées ou désactivées pour chaque compartiment. Voir "Obtenir le temps d'accès du compartiment".

Le tableau suivant décrit comment StorageGRID implémente les opérations sur les compartiments de l'API REST S3. Pour effectuer l'une de ces opérations, les identifiants d'accès nécessaires doivent être fournis pour le compte.

Opération	Mise en œuvre
CreateBucket	Crée un nouveau compartiment. En créant le compartiment, vous en devenez le propriétaire. • Les noms des compartiments doivent respecter les règles suivantes : ◦ Doit être unique dans chaque système StorageGRID (et pas seulement unique au sein du compte du locataire). ◦ Doit être conforme au DNS. ◦ Doit contenir au moins 3 et au plus 63 caractères. ◦ Peut être une série d'une ou plusieurs étiquettes, les étiquettes adjacentes étant séparées par un point. Chaque étiquette doit commencer et se terminer par une lettre minuscule ou un chiffre et ne peut utiliser que des lettres minuscules, des chiffres et des traits d'union. ◦ Ne doit pas ressembler à une adresse IP au format texte. ◦ Il est déconseillé d'utiliser des points dans les requêtes hébergées virtuellement. Les points peuvent entraîner des problèmes lors de la vérification du certificat générique du serveur. • Par défaut, les compartiments sont créés dans la `us-east-1` région ; cependant, vous pouvez utiliser l' `LocationConstraint` élément dans le corps de la requête pour spécifier une autre région. Lorsque vous utilisez l' `LocationConstraint` élément, vous devez indiquer le nom exact d'une région qui a été définie à l'aide de Grid Manager ou de l'API Grid Management. Contactez votre administrateur système si vous ne connaissez pas le nom de la région à utiliser. **Remarque** : Une erreur se produira si votre requête CreateBucket utilise une région qui n'a pas été définie dans StorageGRID. • Vous pouvez inclure l' `x-amz-bucket-object-lock-enabled`en-tête de requête pour créer un compartiment avec S3 Object Lock activé. Voir "Utilisez l'API REST S3 pour configurer S3 Object Lock". Vous devez activer S3 Object Lock lors de la création du compartiment. Vous ne pouvez pas ajouter ni désactiver S3 Object Lock après la création d'un compartiment. S3 Object Lock nécessite le versionnage du compartiment, qui est activé automatiquement lors de la création du compartiment.
DeleteBucket	Supprime le compartiment.
DeleteBucketCors	Supprime la configuration CORS du compartiment.
DeleteBucketEncryption	Supprime le chiffrement par défaut du compartiment. Les objets chiffrés existants restent chiffrés, mais les nouveaux objets ajoutés au compartiment ne sont pas chiffrés.
DeleteBucketLifecycle	Supprime la configuration du cycle de vie du compartiment. Voir "Créer une configuration de cycle de vie S3".

Opération	Mise en œuvre
DeleteBucketPolicy	Supprime la stratégie associée au compartiment.
DeleteBucketReplication	Supprime la configuration de réplication associée au compartiment.
DeleteBucketTagging	Utilise la sous-ressource `tagging` pour supprimer toutes les balises d'un compartiment. **Attention** : Si une balise de règle de configuration ILM non par défaut est définie pour ce compartiment, il y aura une balise `NTAP-SG-ILM-BUCKET-TAG` de compartiment avec une valeur qui lui est attribuée. N'émettez pas de requête DeleteBucketTagging s'il existe une balise `NTAP-SG-ILM-BUCKET-TAG` de compartiment. Émettez plutôt une requête PutBucketTagging contenant uniquement la balise `NTAP-SG-ILM-BUCKET-TAG` et sa valeur attribuée afin de supprimer toutes les autres balises du compartiment. Ne modifiez ni ne supprimez la balise `NTAP-SG-ILM-BUCKET-TAG` de compartiment.
GetBucketAcl	Renvoie une réponse positive ainsi que l'ID, DisplayName et la permission du propriétaire du compartiment, indiquant que ce dernier dispose d'un accès complet au compartiment.
GetBucketCors	Renvoie la `cors`configuration du compartiment.
GetBucketEncryption	Renvoie la configuration de chiffrement par défaut du compartiment.
GetBucketLifecycleConfiguration (anciennement nommé GET Bucket lifecycle)	Renvoie la configuration du cycle de vie du compartiment. Voir "Créer une configuration de cycle de vie S3".
GetBucketLocation	Renvoie la région qui a été définie à l'aide de l'élément `LocationConstraint` dans la requête CreateBucket. Si la région du compartiment est `us-east-1`, une chaîne vide est renvoyée pour la région.
GetBucketNotificationConfiguration (anciennement nommé GET Bucket notification)	Renvoie la configuration de notification associée au compartiment.
GetBucketPolicy	Renvoie la règle de configuration associée au compartiment.
GetBucketReplication	Renvoie la configuration de réplication associée au compartiment.

Opération	Mise en œuvre
GetBucketTagging	Utilise la `tagging` sous-ressource pour renvoyer toutes les balises d'un compartiment. **Attention** : Si une étiquette de règle ILM non par défaut est définie pour ce compartiment, il y aura une étiquette de compartiment `NTAP-SG-ILM-BUCKET-TAG` avec une valeur qui lui est attribuée. Ne modifiez ni ne supprimez cette étiquette.
GetBucketVersioning	Cette implémentation utilise la `versioning` sous-ressource pour renvoyer l'état de versionnage d'un compartiment. • *blank*: Le versionnage n'a jamais été activé (bucket est "Non versionné") • Activé : Le versionnage est activé • Suspendu : le versionnage était précédemment activé et est suspendu
GetObjectLockConfiguration	Renvoie le mode de rétention par défaut et la période de rétention par défaut du compartiment, si configurés. Voir "Utilisez l'API REST S3 pour configurer S3 Object Lock".
HeadBucket	Détermine si un compartiment existe et si vous avez l'autorisation d'y accéder. Cette opération renvoie : • `x-ntap-sg-bucket-id`: L'UUID du compartiment au format UUID. • `x-ntap-sg-trace-id` : L'identifiant de suivi unique de la requête associée.

Opération	Mise en œuvre
ListObjects et ListObjectsV2 (anciennement nommé GET Bucket)	Renvoie tout ou partie (jusqu'à 1 000) des objets d'un compartiment. La classe de stockage des objets peut prendre l'une ou l'autre de deux valeurs, même si l'objet a été ingéré avec l'option de classe de stockage `REDUCED_REDUNDANCY` : • `STANDARD`, ce qui indique que l'objet est stocké dans un pool de stockage composé de Storage Nodes. • `GLACIER`, ce qui indique que l'objet a été déplacé vers le compartiment externe spécifié par le Cloud Storage Pool. Si le compartiment contient un grand nombre de clés supprimées ayant le même préfixe, la réponse peut inclure certaines `CommonPrefixes` qui ne contiennent pas de clés. Pour les requêtes HeadObject et ListObject, StorageGRID renvoie les horodatages LastModified avec une précision différente, tandis qu'AWS renvoie les horodatages avec la même précision, comme le montrent les exemples suivants : • StorageGRID HeadObject: "LastModified": "2024-09-26T16:43:24+00:00" • StorageGRID ListObject : « LastModified » : « 2024-09-26T16:43:24.931000+00:00 » • AWS HeadObject: "LastModified": "2023-10-17T00:19:54+00:00" • AWS ListObject: "LastModified": "2023-10-17T00:19:54+00:00"
ListObjectVersions (anciennement nommé GET Bucket Object versions)	Avec un accès en lecture sur un compartiment, l'utilisation de cette opération avec la `versions` sous-ressource répertorie les métadonnées de toutes les versions des objets dans le compartiment.
PutBucketCors	Définit la configuration CORS d'un compartiment afin que celui-ci puisse traiter des requêtes inter-origines. Le partage de ressources inter-origines (CORS) est un mécanisme de sécurité qui permet aux applications web clientes d'un domaine d'accéder aux ressources d'un domaine différent. Par exemple, supposons que vous utilisiez un compartiment S3 nommé `images` pour stocker des graphiques. En définissant la configuration CORS pour le compartiment `images`, vous pouvez autoriser l'affichage des images de ce compartiment sur le site web `http://www.example.com`.

Opération	Mise en œuvre
PutBucketEncryption	Définit l'état de chiffrement par défaut d'un compartiment existant. Lorsque le chiffrement au niveau du compartiment est activé, tout nouvel objet ajouté au compartiment est chiffré. StorageGRID prend en charge le chiffrement côté serveur avec des clés gérées par StorageGRID. Lors de la spécification de la règle de configuration du chiffrement côté serveur, définissez le paramètre `SSEAlgorithm` sur `AES256`, et n'utilisez pas le paramètre `KMSMasterKeyID`. La configuration de chiffrement par défaut du compartiment est ignorée si la requête de téléchargement d'objet spécifie déjà un chiffrement (c'est-à-dire si la requête inclut l' `x-amz-server-side-encryption-*`en-tête de requête).
PutBucketLifecycleConfiguration (anciennement nommé PUT Bucket lifecycle)	Crée une nouvelle configuration de cycle de vie pour le compartiment ou remplace une configuration de cycle de vie existante. StorageGRID prend en charge jusqu'à 1 000 règles de cycle de vie dans une règle de configuration. Chaque règle peut inclure les éléments XML suivants : • Expiration (Jours, Date, ExpiredObjectDeleteMarker) • NoncurrentVersionExpiration (NewerNoncurrentVersions, NoncurrentDays) • Filtrer (Préfixe, Tag) • Statut • ID StorageGRID ne prend pas en charge ces actions : • AbortIncompleteMultipartUpload • Transition Voir "Créer une configuration de cycle de vie S3". Pour comprendre comment l'action d'expiration dans un cycle de vie de compartiment interagit avec les instructions de placement ILM, voir "Comment l'ILM fonctionne tout au long du cycle de vie d'un objet". **Remarque** : La configuration du cycle de vie des compartiments peut être utilisée avec les compartiments dont le S3 Object Lock est activé, mais la configuration du cycle de vie des compartiments n'est pas prise en charge pour les anciens compartiments Compliant.

Opération	Mise en œuvre

PutBucketNotificationConfiguration
 	
(anciennement nommé PUT Bucket notification)
	Configure les notifications pour le compartiment à l'aide du fichier XML de configuration des notifications inclus dans le corps de la requête. Veuillez prendre en compte les détails d'implémentation suivants :
	
	• StorageGRID prend en charge les sujets Amazon Simple Notification Service (Amazon SNS), les sujets Kafka ou les points de terminaison webhook comme destinations. Les points de terminaison Simple Queue Service (SQS) ou AWS Lambda ne sont pas pris en charge.
	• La destination des notifications doit être spécifiée comme l'URN d'un point de terminaison StorageGRID. Les points de terminaison peuvent être créés à l'aide du Tenant Manager ou de l'API Tenant Management.
	
	Le point de terminaison doit exister pour que la configuration des notifications réussisse. Si le point de terminaison n'existe pas, une `400 Bad Request` erreur est renvoyée avec le code `InvalidArgument`.
	
	• Il est impossible de configurer une notification pour les types d'événements suivants. Ces types d'événements ne sont **pas** pris en charge.
	◦ `s3:ReducedRedundancyLostObject`
	◦ `s3:ObjectRestore:Completed`
	• Les notifications d'événements envoyées par StorageGRID utilisent le format JSON standard, à l'exception de certaines clés qui ne sont pas incluses et d'autres qui utilisent des valeurs spécifiques, comme indiqué dans la liste suivante :
	◦ **eventSource**
	
	`sgws:s3`
	
	◦ **awsRegion**
	
	non inclus
	
	◦ **x-amz-id-2**
	
	non inclus
	
	◦ **arn**
	
	`urn:sgws:s3:::bucket_name`

PutBucketPolicy	Définit la politique associée au compartiment. Voir "Utilisez les stratégies d'accès aux compartiments et aux groupes".

Opération	Mise en œuvre
PutBucketReplication	Configure "Réplication CloudMirror StorageGRID" pour le compartiment à l'aide du fichier XML de configuration de réplication fourni dans le corps de la requête. Pour la réplication CloudMirror, veuillez prendre en compte les détails d'implémentation suivants : • StorageGRID ne prend en charge que la version 1 de la configuration de réplication. Cela signifie que StorageGRID ne prend pas en charge l'utilisation de l' `Filter`élément pour les règles et suit les conventions de la version 1 pour la suppression des versions d'objets. Pour plus de détails, consultez "Guide de l'utilisateur Amazon Simple Storage Service : configuration de la réplication". • La réplication des compartiments peut être configurée sur des compartiments versionnés ou non versionnés. • Vous pouvez spécifier un compartiment de destination différent dans chaque règle du fichier XML de configuration de réplication. Un compartiment source peut répliquer vers plusieurs compartiments de destination. • Les compartiments de destination doivent être spécifiés comme l'URN des points de terminaison StorageGRID, tels que définis dans le Tenant Manager ou l'API Tenant Management. Voir "Configurer la réplication CloudMirror". Le point de terminaison doit exister pour que la configuration de la réplication réussisse. Si le point de terminaison n'existe pas, la requête échoue en tant que `400 Bad Request`. Le message d'erreur indique : `Unable to save the replication policy. The specified endpoint URN does not exist: URN`. • Il n'est pas nécessaire de spécifier un `Role` dans le fichier XML de configuration. Cette valeur n'est pas utilisée par StorageGRID et sera ignorée si elle est soumise. • Si vous omettez la classe de stockage dans le fichier XML de configuration, StorageGRID utilise la classe de stockage `STANDARD` par défaut. • Si vous supprimez un objet du compartiment source ou si vous supprimez le compartiment source lui-même, le comportement de la réplication interrégionale est le suivant : ◦ Si vous supprimez l'objet ou le compartiment avant qu'il n'ait été répliqué, l'objet/compartiment n'est pas répliqué et vous n'êtes pas notifié. ◦ Si vous supprimez l'objet ou le compartiment après sa réplication, StorageGRID suit le comportement de suppression standard d'Amazon S3 pour la version 1 de la réplication interrégionale.

Opération	Mise en œuvre
PutBucketTagging	Utilise la `tagging` sous-ressource pour ajouter ou mettre à jour un ensemble d'étiquettes pour un compartiment. Lors de l'ajout d'étiquettes de compartiment, tenez compte des limitations suivantes : • StorageGRID et Amazon S3 prennent tous deux en charge jusqu'à 50 balises par compartiment. • Les étiquettes associées à un compartiment doivent avoir des clés d'étiquette uniques. Une clé d'étiquette peut comporter jusqu'à 128 caractères Unicode. • Les valeurs des balises peuvent comporter jusqu'à 256 caractères Unicode. • Les clés et les valeurs sont sensibles à la casse. **Attention** : Si une balise de stratégie ILM non standard est définie pour ce compartiment, il y aura une balise `NTAP-SG-ILM-BUCKET-TAG` de compartiment avec une valeur qui lui est attribuée. Assurez-vous que la balise `NTAP-SG-ILM-BUCKET-TAG` de compartiment est incluse avec la valeur attribuée dans toutes les requêtes PutBucketTagging. Ne modifiez ni ne supprimez cette balise. **Remarque** : Cette opération écrasera toutes les étiquettes existantes du compartiment. Si des étiquettes existantes sont omises de l'ensemble, ces étiquettes seront supprimées du compartiment.
PutBucketVersioning	Utilise la `versioning` sous-ressource pour définir l'état de versionnage d'un compartiment existant. Vous pouvez définir l'état de versionnage avec l'une des valeurs suivantes : • Activé : active le versionnage des objets du compartiment. Tous les objets ajoutés au compartiment reçoivent un identifiant de version unique. • Suspendu : désactive le versionnage des objets du compartiment. Tous les objets ajoutés au compartiment reçoivent l'ID de version `null`.
PutObjectLockConfiguration	Configure ou supprime le mode de rétention par défaut et la période de rétention par défaut du compartiment. Si la période de conservation par défaut est modifiée, la date limite de conservation des versions d'objets existantes reste la même et n'est pas recalculée en fonction de la nouvelle période de conservation par défaut. Voir "Utilisez l'API REST S3 pour configurer S3 Object Lock" pour plus d'informations.

Opérations sur les objets

Comment StorageGRID implémente les opérations de l'API REST S3 pour les objets

Cette section décrit comment le système StorageGRID implémente les opérations de l'API REST S3 pour les objets.

Les conditions suivantes s'appliquent à toutes les opérations sur les objets :

- StorageGRID "valeurs de cohérence" est pris en charge par toutes les opérations sur les objets, à l'exception des suivantes :
  - GetObjectAcl
  - `OPTIONS /`
  - PutObjectLegalHold
  - PutObjectRetention
  - SelectObjectContent
- Les requêtes clients conflictuelles, comme par exemple deux clients écrivant sur la même clé, sont résolues selon le principe du « latest-wins ». Le moment de l'évaluation du « latest-wins » est déterminé par la date à laquelle le système StorageGRID termine une requête donnée, et non par la date à laquelle les clients S3 commencent une opération.
- Tous les objets dans un StorageGRID bucket sont la propriété du propriétaire du bucket, y compris les objets créés par un utilisateur anonyme ou par un autre compte.

Le tableau suivant décrit comment StorageGRID implémente les opérations d'objet de l'API REST S3.

Opération	Mise en œuvre
DeleteObject	Lors du traitement d'une requête DeleteObject, StorageGRID tente de supprimer immédiatement toutes les copies de l'objet de tous les emplacements de stockage. En cas de succès, StorageGRID renvoie immédiatement une réponse au client. Si toutes les copies ne peuvent pas être supprimées dans un délai de 30 secondes (par exemple, si un emplacement est temporairement indisponible), StorageGRID met les copies en file d'attente pour suppression, puis indique la réussite de l'opération au client. **Restrictions** • L'authentification multifacteur (MFA) et l'en-tête de réponse `x-amz-mfa` ne sont pas pris en charge. • Les `If-None` et `If-None-Match` en-têtes sont acceptés mais non fonctionnels. **Versionnage** Pour supprimer une version spécifique, le demandeur doit être le propriétaire du compartiment et utiliser la sous-ressource `versionId`. L'utilisation de cette sous-ressource supprime définitivement la version. Si la `versionId` correspond à un marqueur de suppression, l'en-tête de réponse `x-amz-delete-marker` est renvoyé avec la valeur `true`. • Si un objet est supprimé sans la `versionId` sous-ressource sur un compartiment avec le versionnage activé, cela entraîne la génération d'un marqueur de suppression. L' `versionId` du marqueur de suppression est renvoyé via l'en-tête de réponse `x-amz-version-id`, et l'en-tête de réponse `x-amz-delete-marker` est renvoyé avec la valeur `true`. • Si un objet est supprimé sans la `versionId` sous-ressource sur un compartiment dont le versionnage est suspendu, cela entraîne la suppression définitive d'une version « null » existante ou d'un marqueur de suppression « null », ainsi que la génération d'un nouveau marqueur de suppression « null ». L'en-tête de réponse `x-amz-delete-marker` est renvoyé avec la valeur `true`. **Remarque** : Dans certains cas, plusieurs marqueurs de suppression peuvent exister pour un objet. Consultez "Utilisez l'API REST S3 pour configurer S3 Object Lock" pour savoir comment supprimer les versions d'objets en mode GOUVERNANCE.

Opération	Mise en œuvre
DeleteObjects (anciennement nommé DELETE Multiple Objects)	L'authentification multifacteur (MFA) et l'en-tête de réponse `x-amz-mfa` ne sont pas pris en charge. Plusieurs objets peuvent être supprimés dans le même message de requête. Consultez "Utilisez l'API REST S3 pour configurer S3 Object Lock" pour savoir comment supprimer les versions d'objets en mode GOUVERNANCE.
DeleteObjectTagging	Utilise la `tagging` sous-ressource pour supprimer toutes les balises d'un objet. **Versionnage** Si le `versionId` paramètre de requête n'est pas spécifié dans la requête, l'opération supprime toutes les balises de la version la plus récente de l'objet dans un compartiment versionné. Si la version actuelle de l'objet est un marqueur de suppression, un statut "MethodNotAllowed" est renvoyé avec l' `x-amz-delete-marker` en-tête de réponse défini sur `true`.
GetObject	"GetObject"
GetObjectAcl	Si les informations d'identification nécessaires sont fournies pour le compte, l'opération renvoie une réponse positive ainsi que l'ID, DisplayName et l'autorisation du propriétaire de l'objet, indiquant que ce dernier dispose d'un accès complet à l'objet.
GetObjectLegalHold	"Utilisez l'API REST S3 pour configurer S3 Object Lock"
GetObjectRetention	"Utilisez l'API REST S3 pour configurer S3 Object Lock"
GetObjectTagging	Utilise la `tagging` sous-ressource pour renvoyer toutes les balises d'un objet. **Versionnage** Si le `versionId` paramètre de requête n'est pas spécifié dans la requête, l'opération renvoie toutes les balises de la version la plus récente de l'objet dans un compartiment versionné. Si la version actuelle de l'objet est un marqueur de suppression, un statut "MethodNotAllowed" est renvoyé avec l' `x-amz-delete-marker` en-tête de réponse défini sur `true`.
HeadObject	"HeadObject"
RestoreObject	"RestoreObject"

Opération	Mise en œuvre
PutObject	"PutObject"
CopyObject (anciennement nommé PUT Object - Copy)	"CopyObject"
PutObjectLegalHold	"Utilisez l'API REST S3 pour configurer S3 Object Lock"
PutObjectRetention	"Utilisez l'API REST S3 pour configurer S3 Object Lock"

Opération	Mise en œuvre
PutObjectTagging	Utilise la `tagging` sous-ressource pour ajouter un ensemble de balises à un objet existant. **Limites des balises d'objet** Vous pouvez ajouter des balises aux nouveaux objets lors de leur chargement, ou vous pouvez les ajouter à des objets existants. StorageGRID et Amazon S3 prennent en charge jusqu'à 10 balises par objet. Les balises associées à un objet doivent avoir des clés de balise uniques. Une clé de balise peut comporter jusqu'à 128 caractères Unicode et les valeurs de balise peuvent comporter jusqu'à 256 caractères Unicode. Les clés et les valeurs sont sensibles à la casse. **Mises à jour des balises et comportement d'ingestion** Lorsque vous utilisez PutObjectTagging pour mettre à jour les balises d'un objet, StorageGRID ne réingère pas l'objet. Cela signifie que l'option de comportement d'ingestion spécifiée dans la règle ILM correspondante n'est pas utilisée. Toute modification du placement de l'objet déclenchée par la mise à jour est effectuée lorsque l'ILM est réévalué par les processus ILM d'arrière-plan habituels. Cela signifie que si la règle ILM utilise l'option Strict pour le comportement d'ingestion, aucune action n'est entreprise si les emplacements d'objets requis ne peuvent pas être effectués (par exemple, parce qu'un emplacement nouvellement requis est indisponible). L'objet mis à jour conserve son emplacement actuel jusqu'à ce que l'emplacement requis soit possible. **Résolution des conflits** Les requêtes clients conflictuelles, comme par exemple deux clients écrivant sur la même clé, sont résolues selon le principe du « latest-wins ». Le moment de l'évaluation du « latest-wins » est déterminé par la date à laquelle le système StorageGRID termine une requête donnée, et non par la date à laquelle les clients S3 commencent une opération. **Versionnage** Si le `versionId` paramètre de requête n'est pas spécifié dans la requête, l'opération ajoute des balises à la version la plus récente de l'objet dans un compartiment versionné. Si la version actuelle de l'objet est un marqueur de suppression, un statut "MethodNotAllowed" est renvoyé avec l' `x-amz-delete-marker` en-tête de réponse défini sur `true`.
SelectObjectContent	"SelectObjectContent"

Opérations Amazon S3 Select prises en charge dans StorageGRID

StorageGRID prend en charge les clauses, les types de données et les opérateurs Amazon S3 Select suivants pour le "commande SelectObjectContent".

Les articles non répertoriés ne sont pas pris en charge.

Pour la syntaxe, consultez "SelectObjectContent". Pour plus d'informations sur S3 Select, consultez le "Documentation AWS pour S3 Select".

Seuls les comptes locataires pour lesquels S3 Select est activé peuvent émettre des requêtes SelectObjectContent. Consultez "Considérations et exigences relatives à l'utilisation de S3 Select".

${post_edited_translations.segment}

- ${post_edited_translations.segment}
- ${post_edited_translations.segment}
- clause WHERE
- ${post_edited_translations.segment}

${post_edited_translations.segment}

- bool
- ${post_edited_translations.segment}
- chaîne
- flottant
- ${post_edited_translations.segment}
- horodatage

${post_edited_translations.segment}

${post_edited_translations.segment}

- ET
- PAS
- ${post_edited_translations.segment}

Opérateurs de comparaison

- <
- >
- ⇐
- ${post_edited_translations.segment}
- =
- =
- <>
- !=
- ${post_edited_translations.segment}
- DANS

${post_edited_translations.segment}

- ${post_edited_translations.segment}
- ${post_edited_translations.segment}
- ${post_edited_translations.segment}

Opérateurs unitaires

- IS NULL
- ${post_edited_translations.segment}

${post_edited_translations.segment}

- +
- -
- *
- /
- ${post_edited_translations.segment}

${post_edited_translations.segment}

${post_edited_translations.segment}

- ${post_edited_translations.segment}
- COUNT(*)
- ${post_edited_translations.segment}
- ${post_edited_translations.segment}
- ${post_edited_translations.segment}

${post_edited_translations.segment}

- ${post_edited_translations.segment}
- ${post_edited_translations.segment}
- ${post_edited_translations.segment}

${post_edited_translations.segment}

- ${post_edited_translations.segment}

Fonctions de date

- ${post_edited_translations.segment}
- DATE_DIFF
- EXTRAIT
- ${post_edited_translations.segment}
- ${post_edited_translations.segment}

- ${post_edited_translations.segment}

Fonctions de chaînes de caractères

- CHAR_LENGTH, CHARACTER_LENGTH
- INFÉRIEUR
- ${post_edited_translations.segment}
- TRIM
- ${post_edited_translations.segment}

Comment StorageGRID utilise le chiffrement côté serveur

Le chiffrement côté serveur vous permet de protéger vos données au repos. StorageGRID chiffre les données lors de l'écriture de l'objet et les déchiffre lorsque vous y accédez.

Si vous souhaitez utiliser le chiffrement côté serveur, vous pouvez choisir l'une des deux options mutuellement exclusives, selon la manière dont les clés de chiffrement sont gérées :

- **SSE (chiffrement côté serveur avec des clés gérées par StorageGRID)** : Lorsque vous effectuez une requête S3 pour stocker un objet, StorageGRID chiffre cet objet à l'aide d'une clé unique. Lorsque vous effectuez une requête S3 pour récupérer l'objet, StorageGRID utilise la clé stockée pour le déchiffrer.
- **SSE-C (chiffrement côté serveur avec clés fournies par le client)** : Lorsque vous envoyez une requête S3 pour stocker un objet, vous fournissez votre propre clé de chiffrement. Lorsque vous récupérez un objet, vous fournissez la même clé de chiffrement dans votre requête. Si les deux clés de chiffrement correspondent, l'objet est déchiffré et vos données d'objet sont renvoyées.

  Bien que StorageGRID gère toutes les opérations de chiffrement et de déchiffrement des objets, vous devez gérer les clés de chiffrement que vous fournissez.

> Les clés de chiffrement que vous fournissez ne sont jamais stockées. Si vous perdez une clé de chiffrement, vous perdez l'objet correspondant.

> Si un objet est chiffré avec SSE ou SSE-C, tous les paramètres de chiffrement au niveau du compartiment ou de la grille sont ignorés.

Utilisez SSE

Pour chiffrer un objet avec une clé unique gérée par StorageGRID, vous utilisez l'en-tête de requête suivant :

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

${post_edited_translations.segment}

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

Utilisez SSE-C

${post_edited_translations.segment}

En-tête de requête	Description
`x-amz-server-side-encryption-customer-algorithm`	Spécifiez l'algorithme de chiffrement. La valeur de l'en-tête doit être `AES256`.
`x-amz-server-side-encryption-customer-key`	Spécifiez la clé de chiffrement qui sera utilisée pour chiffrer ou déchiffrer l'objet. La valeur de la clé doit être de 256 bits, encodée en base64.
`x-amz-server-side-encryption-customer-key-MD5`	Spécifiez le condensé MD5 de la clé de chiffrement conformément à la norme RFC 1321, qui est utilisé pour s'assurer que la clé de chiffrement a été transmise sans erreur. La valeur du condensé MD5 doit être encodée en base64 sur 128 bits.

${post_edited_translations.segment}

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

Considérations relatives à l'utilisation du chiffrement côté serveur avec des clés fournies par le client (SSE-C)

Avant d'utiliser SSE-C, veuillez prendre en compte les points suivants :

- ${post_edited_translations.segment}

> StorageGRID rejette toutes les requêtes effectuées via http lors de l'utilisation de SSE-C. Pour des raisons de sécurité, vous devez considérer comme compromise toute clé que vous auriez envoyée accidentellement via http. Mettez la clé au rebut et procédez à sa rotation selon les besoins.

- L'ETag dans la réponse n'est pas le MD5 des données de l'objet.
- Vous devez gérer le mappage des clés de chiffrement aux objets. StorageGRID ne stocke pas les clés de chiffrement. Vous êtes responsable du suivi de la clé de chiffrement que vous fournissez pour chaque objet.
- Si le versionnement est activé pour votre compartiment, chaque version d'objet doit disposer de sa propre clé de chiffrement. Vous êtes responsable du suivi de la clé de chiffrement utilisée pour chaque version d'objet.
- Étant donné que vous gérez les clés de chiffrement côté client, vous devez également gérer toutes les mesures de sécurité supplémentaires, telles que la rotation des clés, côté client.

Les clés de chiffrement que vous fournissez ne sont jamais stockées. Si vous perdez une clé de chiffrement, vous perdez l'objet correspondant.

- Si la réplication inter-grilles ou la réplication CloudMirror est configurée pour le compartiment, vous ne pouvez pas ingérer d'objets SSE-C. L'opération d'ingestion échouera.

Informations connexes

"${post_edited_translations.segment}"

Créer une copie d'un objet dans StorageGRID avec la requête S3 CopyObject

Vous pouvez utiliser la requête S3 CopyObject pour créer une copie d'un objet déjà stocké dans S3. Une opération CopyObject équivaut à effectuer GetObject suivi de PutObject.

Résoudre les conflits

Les requêtes clients conflictuelles, comme par exemple deux clients écrivant sur la même clé, sont résolues selon le principe du « latest-wins ». Le moment de l'évaluation du « latest-wins » est déterminé par la date à laquelle le système StorageGRID termine une requête donnée, et non par la date à laquelle les clients S3 commencent une opération.

Taille de l'objet

La taille maximale *recommandée* pour une seule opération PutObject est de 5 Gio (5 368 709 120 octets). Si vous avez des objets de plus de 5 Gio, utilisez plutôt "téléchargement en plusieurs parties".

La taille maximale *prise en charge* pour une seule opération PutObject est de 5 Tio (5 497 558 138 880 octets).

Si vous avez effectué une mise à niveau depuis StorageGRID 11.6 ou une version antérieure, l'alerte « Taille de l'objet S3 trop volumineux lors de la commande PUT » sera déclenchée si vous tentez de télécharger un objet dépassant 5 Gio. Si vous disposez d'une nouvelle installation de StorageGRID 11.7 ou 11.8, l'alerte ne sera pas déclenchée dans ce cas. Cependant, afin de s'aligner sur la norme AWS S3, les prochaines versions de StorageGRID ne prendront plus en charge le téléchargement d'objets de plus de 5 Gio.

Caractères UTF-8 dans les métadonnées utilisateur

Si une requête inclut des valeurs UTF-8 (non échappées) dans le nom de clé ou la valeur des métadonnées définies par l'utilisateur, le comportement de StorageGRID est indéfini.

StorageGRID n'analyse ni n'interprète les caractères UTF-8 échappés inclus dans le nom ou la valeur de la clé des métadonnées définies par l'utilisateur. Les caractères UTF-8 échappés sont traités comme des caractères ASCII :

- Les requêtes aboutissent si les métadonnées définies par l'utilisateur incluent des caractères UTF-8 échappés.
- StorageGRID ne renvoie pas l' `x-amz-missing-meta`en-tête si la valeur interprétée du nom ou de la valeur de la clé contient des caractères non imprimables.

En-têtes de requête pris en charge

Les en-têtes de requête suivants sont pris en charge :

- `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-`, suivi d'une paire nom-valeur contenant des métadonnées définies par l'utilisateur
- `x-amz-metadata-directive` : La valeur par défaut est `COPY`, ce qui vous permet de copier l'objet et les métadonnées associées.

  Vous pouvez spécifier `REPLACE` pour écraser les métadonnées existantes lors de la copie de l'objet, ou pour mettre à jour les métadonnées de l'objet.
- `x-amz-storage-class`
- `x-amz-tagging-directive` : La valeur par défaut est `COPY`, ce qui vous permet de copier l'objet et toutes ses balises.

  Vous pouvez spécifier `REPLACE` d'écraser les balises existantes lors de la copie de l'objet, ou de mettre à jour les balises.
- En-têtes de requête S3 Object Lock :
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    Si une requête est effectuée sans ces en-têtes, les paramètres de rétention par défaut du compartiment sont utilisés pour calculer le mode de version de l'objet et la date de rétention jusqu'à la date limite. Voir "Utilisez l'API REST S3 pour configurer S3 Object Lock".
- En-têtes de requête SSE :
  - `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`

    Voir En-têtes de requête pour le chiffrement côté serveur

En-têtes de requête non pris en charge

Les en-têtes de requête suivants ne sont pas pris en charge :

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

  Le `If-Match header` est accepté mais non fonctionnel.

- `If-None-Match`

  Le `If-None-Match header` est accepté mais non fonctionnel.

- `x-amz-checksum-algorithm`

  Lors de la copie d'un objet, si l'objet source possède une somme de contrôle, StorageGRID ne copie pas cette valeur de somme de contrôle dans le nouvel objet. Ce comportement s'applique que vous tentiez ou non d'utiliser `x-amz-checksum-algorithm` dans la requête d'objet.

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

Options de classe de stockage

L' `x-amz-storage-class`en-tête de requête est pris en charge et influe sur le nombre de copies d'objets que StorageGRID crée si la règle ILM correspondante utilise la validation double ou équilibrée"option d'ingestion".

- `STANDARD`

  (Par défaut) Spécifie une opération d'ingestion à double validation lorsque la règle ILM utilise l'option Dual commit, ou lorsque l'option Balanced revient à créer des copies intermédiaires.

- `REDUCED_REDUNDANCY`

  Spécifie une opération d'ingestion à validation unique lorsque la règle ILM utilise l'option de double validation, ou lorsque l'option équilibrée revient à créer des copies intermédiaires.

  > Si vous ingérez un objet dans un compartiment avec S3 Object Lock activé, l' `REDUCED_REDUNDANCY`option est ignorée. Si vous ingérez un objet dans un compartiment Compliant hérité, l' `REDUCED_REDUNDANCY`option renvoie une erreur. StorageGRID effectuera toujours une double validation à l'ingestion pour garantir le respect des exigences de conformité.

Utilisation de x-amz-copy-source dans CopyObject

Si le compartiment et la clé source, spécifiés dans l `x-amz-copy-source`en-tête, sont différents du compartiment et de la clé de destination, une copie des données de l'objet source est écrite dans la

destination.

Si la source et la destination correspondent, et que l'en-tête `x-amz-metadata-directive` est spécifié comme `REPLACE`, les métadonnées de l'objet sont mises à jour avec les valeurs de métadonnées fournies dans la requête. Dans ce cas, StorageGRID ne réingère pas l'objet. Cela a deux conséquences importantes :

- Vous ne pouvez pas utiliser CopyObject pour chiffrer un objet existant sur place, ni pour modifier le chiffrement d'un objet existant sur place. Si vous fournissez l' `x-amz-server-side-encryption` en-tête ou l' `x-amz-server-side-encryption-customer-algorithm` en-tête, StorageGRID rejette la requête et renvoie `XNotImplemented`.
- L'option de comportement d'ingestion spécifiée dans la règle ILM correspondante n'est pas utilisée. Toute modification du placement des objets déclenchée par la mise à jour est effectuée lorsque l'ILM est réévalué par les processus ILM d'arrière-plan normaux.

  Cela signifie que si la règle ILM utilise l'option Strict pour le comportement d'ingestion, aucune action n'est entreprise si les emplacements d'objets requis ne peuvent pas être effectués (par exemple, parce qu'un emplacement nouvellement requis est indisponible). L'objet mis à jour conserve son emplacement actuel jusqu'à ce que l'emplacement requis soit possible.

En-têtes de requête pour le chiffrement côté serveur

Si vous "utiliser le chiffrement côté serveur", les en-têtes de requête que vous fournissez dépendent du chiffrement de l'objet source et de votre intention de chiffrer l'objet cible.

- Si l'objet source est chiffré à l'aide d'une clé fournie par le client (SSE-C), vous devez inclure les trois en-têtes suivants dans la requête CopyObject, afin que l'objet puisse être déchiffré puis copié :
  - `x-amz-copy-source-server-side-encryption-customer-algorithm`: Précisez `AES256`.
  - `x-amz-copy-source-server-side-encryption-customer-key`: Spécifiez la clé de chiffrement que vous avez fournie lors de la création de l'objet source.
  - `x-amz-copy-source-server-side-encryption-customer-key-MD5`: Spécifiez le condensé MD5 que vous avez fourni lors de la création de l'objet source.
- Si vous souhaitez chiffrer l'objet cible (la copie) avec une clé unique que vous fournissez et gérez, incluez les trois en-têtes suivants :
  - `x-amz-server-side-encryption-customer-algorithm`: Précisez `AES256`.
  - `x-amz-server-side-encryption-customer-key`: Spécifiez une nouvelle clé de chiffrement pour l'objet cible.
  - `x-amz-server-side-encryption-customer-key-MD5`: Spécifiez le condensé MD5 de la nouvelle clé de chiffrement.

  > Les clés de chiffrement que vous fournissez ne sont jamais stockées. Si vous perdez une clé de chiffrement, vous perdez l'objet correspondant. Avant d'utiliser des clés fournies par le client pour sécuriser les données d'objet, examinez les considérations pour "utilisation du chiffrement côté serveur".

- Si vous souhaitez chiffrer l'objet cible (la copie) avec une clé unique gérée par StorageGRID (SSE), incluez cet en-tête dans la requête CopyObject :
  - `x-amz-server-side-encryption`

La `server-side-encryption` valeur de l'objet ne peut pas être mise à jour. Créez plutôt une copie avec une nouvelle `server-side-encryption` valeur en utilisant `x-amz-metadata-directive` : `REPLACE`.

Versionnage

Si le compartiment source est versionné, vous pouvez utiliser l'en-tête `x-amz-copy-source` pour copier la dernière version d'un objet. Pour copier une version spécifique d'un objet, vous devez spécifier explicitement la version à copier à l'aide de la sous-ressource `versionId`. Si le compartiment de destination est versionné, la version générée est renvoyée dans l'en-tête de réponse `x-amz-version-id`. Si le versionnement est suspendu pour le compartiment cible, alors `x-amz-version-id` renvoie une valeur "null".

Récupérer un objet depuis un bucket dans StorageGRID avec la requête GetObject

Vous pouvez utiliser la requête S3 GetObject pour récupérer un objet depuis un compartiment S3.

GetObject et objets multipart

Vous pouvez utiliser le `partNumber`paramètre de requête pour récupérer une partie spécifique d'un objet multipart ou segmenté. L' `x-amz-mp-parts-count`élément de réponse indique combien de parties l'objet possède.

Vous pouvez définir `partNumber` à 1 pour les objets segmentés/multipart et les objets non segmentés/non multipart ; cependant, l'élément de réponse `x-amz-mp-parts-count` n'est renvoyé que pour les objets segmentés ou multipart.

Caractères UTF-8 dans les métadonnées utilisateur

StorageGRID n'analyse ni n'interprète les caractères UTF-8 échappés dans les métadonnées définies par l'utilisateur. Les requêtes GET concernant un objet dont les métadonnées définies par l'utilisateur contiennent des caractères UTF-8 échappés ne renvoient pas l `x-amz-missing-meta`en-tête si le nom ou la valeur de la clé inclut des caractères non imprimables.

En-tête de requête pris en charge

L'en-tête de requête suivant est pris en charge :

- `x-amz-checksum-mode`: Spécifiez `ENABLED`

  L' `Range`en-tête n'est pas pris en charge avec `x-amz-checksum-mode` pour GetObject. Lorsque vous incluez `Range` dans la requête avec `x-amz-checksum-mode` activé, StorageGRID ne renvoie pas de valeur de somme de contrôle dans la réponse.

En-tête de requête non pris en charge

L'en-tête de requête suivant n'est pas pris en charge et renvoie `XNotImplemented` :

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

Versionnage

Si aucune `versionId` sous-ressource n'est spécifiée, l'opération récupère la version la plus récente de l'objet dans un compartiment versionné. Si la version actuelle de l'objet est un marqueur de suppression, un statut « Not Found » est renvoyé avec l' `x-amz-delete-marker` en-tête de réponse défini sur `true`.

En-têtes de requête pour le chiffrement côté serveur avec des clés de chiffrement fournies par le client (SSE-C)

Utilisez les trois en-têtes si l'objet est chiffré avec une clé unique que vous avez fournie.

- `x-amz-server-side-encryption-customer-algorithm`: Précisez `AES256`.
- `x-amz-server-side-encryption-customer-key`: Spécifiez votre clé de chiffrement pour l'objet.
- `x-amz-server-side-encryption-customer-key-MD5`: Spécifiez le condensé MD5 de la clé de chiffrement de l'objet.

> Les clés de chiffrement que vous fournissez ne sont jamais stockées. Si vous perdez une clé de chiffrement, vous perdez l'objet correspondant. Avant d'utiliser des clés fournies par le client pour sécuriser les données d'objet, examinez les considérations dans "Utilisez le chiffrement côté serveur".

Comportement de GetObject pour les objets Cloud Storage Pool

Si un objet a été stocké dans un "Pool de stockage cloud", le comportement d'une requête GetObject dépend de l'état de l'objet. Voir "HeadObject" pour plus de détails.

> Si un objet est stocké dans un Cloud Storage Pool et qu'une ou plusieurs copies de cet objet existent également sur la grille, les requêtes GetObject tenteront de récupérer les données à partir de la grille avant de les récupérer à partir du Cloud Storage Pool.

État de l'objet	Comportement de GetObject
Objet ingéré dans StorageGRID mais pas encore évalué par ILM, ou objet stocké dans un pool de stockage traditionnel ou utilisant le code d'effacement	`200 OK` Une copie de l'objet est récupérée.
Objet présent dans le pool de stockage cloud mais n'ayant pas encore été transféré dans un état non récupérable	`200 OK` Une copie de l'objet est récupérée.
L'objet est passé dans un état non récupérable	`403 Forbidden`, `InvalidObjectState` Utilisez une "RestoreObject" demande pour restaurer l'objet à un état récupérable.
Objet en cours de restauration à partir d'un état non récupérable	`403 Forbidden`, `InvalidObjectState` Attendez que la requête RestoreObject soit terminée.

État de l'objet	Comportement de GetObject
Objet entièrement restauré dans le Cloud Storage Pool	`200 OK` Une copie de l'objet est récupérée.

Objets multipartites ou segmentés dans un Cloud Storage Pool

Si vous avez importé un objet multipart ou si StorageGRID a divisé un objet volumineux en segments, StorageGRID détermine si l'objet est disponible dans le Cloud Storage Pool en échantillonnant un sous-ensemble des parties ou segments de l'objet. Dans certains cas, une requête GetObject peut renvoyer `200 OK` de façon incorrecte lorsque certaines parties de l'objet ont déjà été transférées dans un état non récupérable ou lorsque certaines parties de l'objet n'ont pas encore été restaurées.

Dans ces cas :

- La requête GetObject peut renvoyer des données mais s'interrompre en cours de transfert.
- Une requête GetObject ultérieure pourrait renvoyer `403 Forbidden`.

GetObject et réplication inter-grilles

Si vous utilisez "fédération de grid" et "réplication inter-grilles" est activé pour un compartiment, le client S3 peut vérifier l'état de réplication d'un objet en envoyant une requête GetObject. La réponse inclut l'en-tête de réponse spécifique à StorageGRID `x-ntap-sg-cgr-replication-status`, qui aura l'une des valeurs suivantes :

Grid	État de la réplication
Source	• **TERMINÉ** : La réplication a réussi. • **EN ATTENTE** : L'objet n'a pas encore été répliqué. • **ÉCHEC** : La réplication a échoué de manière permanente. Un utilisateur doit résoudre l'erreur.
Destination	**REPLICA** : L'objet a été répliqué à partir de la grille source.

> StorageGRID ne prend pas en charge l' `x-amz-replication-status`en-tête.

Récupérer les métadonnées d'un objet dans StorageGRID avec la requête S3 HeadObject

Vous pouvez utiliser la requête S3 HeadObject pour récupérer les métadonnées d'un objet sans renvoyer l'objet lui-même. Si l'objet est stocké dans un Cloud Storage Pool, vous pouvez utiliser HeadObject pour déterminer l'état de transition de l'objet.

HeadObject et objets multipart

Vous pouvez utiliser le `partNumber`paramètre de requête pour récupérer les métadonnées d'une partie spécifique d'un objet multipart ou segmenté. L' `x-amz-mp-parts-count`élément de réponse indique combien de parties l'objet possède.

Vous pouvez définir `partNumber` à 1 pour les objets segmentés/multipart et les objets non segmentés/non multipart ; cependant, l'élément de réponse `x-amz-mp-parts-count` n'est renvoyé que pour les objets segmentés ou multipart.

Caractères UTF-8 dans les métadonnées utilisateur

StorageGRID n'analyse ni n'interprète les caractères UTF-8 échappés dans les métadonnées définies par l'utilisateur. Les requêtes HEAD concernant un objet dont les métadonnées définies par l'utilisateur contiennent des caractères UTF-8 échappés ne renvoient pas l'en-tête `x-amz-missing-meta` si le nom ou la valeur de la clé inclut des caractères non imprimables.

En-tête de requête pris en charge

L'en-tête de requête suivant est pris en charge :

- `x-amz-checksum-mode`

  Le `partNumber`paramètre et `Range`l'en-tête ne sont pas pris en charge avec `x-amz-checksum-mode` pour HeadObject. Lorsque vous les incluez dans la requête avec `x-amz-checksum-mode` activé, StorageGRID ne renvoie pas de valeur de somme de contrôle dans la réponse.

En-tête de requête non pris en charge

L'en-tête de requête suivant n'est pas pris en charge et renvoie `XNotImplemented` :

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

Versionnage

Si aucune `versionId` sous-ressource n'est spécifiée, l'opération récupère la version la plus récente de l'objet dans un compartiment versionné. Si la version actuelle de l'objet est un marqueur de suppression, un statut « Not Found » est renvoyé avec l' `x-amz-delete-marker` en-tête de réponse défini sur `true`.

En-têtes de requête pour le chiffrement côté serveur avec des clés de chiffrement fournies par le client (SSE-C)

Utilisez ces trois en-têtes si l'objet est chiffré avec une clé unique que vous avez fournie.

- `x-amz-server-side-encryption-customer-algorithm`: Précisez `AES256`.
- `x-amz-server-side-encryption-customer-key`: Spécifiez votre clé de chiffrement pour l'objet.
- `x-amz-server-side-encryption-customer-key-MD5`: Spécifiez le condensé MD5 de la clé de chiffrement de l'objet.

> ⚠ Les clés de chiffrement que vous fournissez ne sont jamais stockées. Si vous perdez une clé de chiffrement, vous perdez l'objet correspondant. Avant d'utiliser des clés fournies par le client pour sécuriser les données d'objet, examinez les considérations dans "Utilisez le chiffrement côté serveur".

HeadObject réponses pour les objets Cloud Storage Pool

Si l'objet est stocké dans un "Pool de stockage cloud", les en-têtes de réponse suivants sont renvoyés :

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

- `x-amz-restore`

Les en-têtes de réponse fournissent des informations sur l'état d'un objet lorsqu'il est déplacé vers un Cloud Storage Pool, éventuellement transféré vers un état non récupérable, puis restauré.

État de l'objet	Réponse à HeadObject
Objet ingéré dans StorageGRID mais pas encore évalué par ILM, ou objet stocké dans un pool de stockage traditionnel ou utilisant le code d'effacement	`200 OK` (Aucun en-tête de réponse spécial n'est renvoyé.)

Objet présent dans le pool de stockage cloud mais n'ayant pas encore été transféré dans un état non récupérable	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"`
	
	Jusqu'à ce que l'objet soit passé à un état non récupérable, la valeur pour `expiry-date` est définie à une date lointaine dans le futur. L'heure exacte de la transition n'est pas contrôlée par le système StorageGRID.

L'objet est passé à un état non récupérable, mais au moins une copie existe également sur la grille	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"`
	
	La valeur de `expiry-date` est fixée à une date future lointaine.
	
	Remarque : Si la copie sur la grille n'est pas disponible (par exemple, si un nœud de stockage est hors service), vous devez émettre une demande "RestoreObject" pour restaurer la copie à partir du Cloud Storage Pool avant de pouvoir récupérer l'objet.

L'objet est passé à un état irrécupérable et aucune copie n'existe sur la grille	`200 OK`
	
	`x-amz-storage-class: GLACIER`

État de l'objet	Réponse à HeadObject

Objet en cours de restauration à partir d'un état non récupérable	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="true"`

Objet entièrement restauré dans le Cloud Storage Pool	`200 OK`
	
	`x-amz-storage-class: GLACIER`
	
	`x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2018 00:00:00 GMT"`
	
	Le `expiry-date` indique quand l'objet du Cloud Storage Pool sera renvoyé à un état non récupérable.

Objets multipartites ou segmentés dans le Cloud Storage Pool

Si vous avez importé un objet multipart ou si StorageGRID a divisé un objet volumineux en segments, StorageGRID détermine si l'objet est disponible dans le Cloud Storage Pool en échantillonnant un sous-ensemble des parties ou segments de l'objet. Dans certains cas, une requête HeadObject peut renvoyer `x-amz-restore: ongoing-request="false"` de façon incorrecte lorsque certaines parties de l'objet ont déjà été transférées dans un état non récupérable ou lorsque certaines parties de l'objet n'ont pas encore été restaurées.

HeadObject et réplication inter-grilles

Si vous utilisez "fédération de grid" et "réplication inter-grilles" est activé pour un compartiment, le client S3 peut vérifier l'état de réplication d'un objet en envoyant une requête HeadObject. La réponse inclut l'en-tête de réponse spécifique à StorageGRID `x-ntap-sg-cgr-replication-status`, qui aura l'une des valeurs suivantes :

Grid	État de la réplication
Source	• **TERMINÉ** : La réplication a réussi. • **EN ATTENTE** : L'objet n'a pas encore été répliqué. • **ÉCHEC** : La réplication a échoué de manière permanente. Un utilisateur doit résoudre l'erreur.
Destination	**REPLICA** : L'objet a été répliqué à partir de la grille source.

> StorageGRID ne prend pas en charge l' `x-amz-replication-status`en-tête.

Ajouter un objet à un compartiment dans StorageGRID avec la requête S3 PutObject

Vous pouvez utiliser la requête S3 PutObject pour ajouter un objet à un compartiment.

Résoudre les conflits

Les requêtes clients conflictuelles, comme par exemple deux clients écrivant sur la même clé, sont résolues selon le principe du « latest-wins ». Le moment de l'évaluation du « latest-wins » est déterminé par la date à laquelle le système StorageGRID termine une requête donnée, et non par la date à laquelle les clients S3 commencent une opération.

Taille de l'objet

La taille maximale *recommandée* pour une seule opération PutObject est de 5 Gio (5 368 709 120 octets). Si vous avez des objets de plus de 5 Gio, utilisez plutôt "téléchargement en plusieurs parties".

La taille maximale *prise en charge* pour une seule opération PutObject est de 5 Tio (5 497 558 138 880 octets).

> Si vous avez effectué une mise à niveau depuis StorageGRID 11.6 ou une version antérieure, l'alerte « Taille de l'objet S3 trop volumineux lors de la commande PUT » sera déclenchée si vous tentez de télécharger un objet dépassant 5 Gio. Si vous disposez d'une nouvelle installation de StorageGRID 11.7 ou 11.8, l'alerte ne sera pas déclenchée dans ce cas. Cependant, afin de s'aligner sur la norme AWS S3, les prochaines versions de StorageGRID ne prendront plus en charge le téléchargement d'objets de plus de 5 Gio.

Taille des métadonnées utilisateur

Amazon S3 limite la taille des métadonnées définies par l'utilisateur au sein de chaque en-tête de requête PUT à 2 Ko. StorageGRID limite les métadonnées utilisateur à 24 Kio. La taille des métadonnées définies par l'utilisateur est mesurée en faisant la somme du nombre d'octets dans l'encodage UTF-8 de chaque clé et valeur.

Caractères UTF-8 dans les métadonnées utilisateur

Si une requête inclut des valeurs UTF-8 (non échappées) dans le nom de clé ou la valeur des métadonnées définies par l'utilisateur, le comportement de StorageGRID est indéfini.

StorageGRID n'analyse ni n'interprète les caractères UTF-8 échappés inclus dans le nom ou la valeur de la clé des métadonnées définies par l'utilisateur. Les caractères UTF-8 échappés sont traités comme des caractères ASCII :

- PutObject, CopyObject, GetObject, et HeadObject réussissent si les métadonnées définies par l'utilisateur incluent des caractères UTF-8 échappés.
- StorageGRID ne renvoie pas l' `x-amz-missing-meta`en-tête si la valeur interprétée du nom ou de la valeur de la clé contient des caractères non imprimables.

Limites des balises d'objet

Vous pouvez ajouter des balises aux nouveaux objets lors de leur chargement, ou vous pouvez les ajouter à des objets existants. StorageGRID et Amazon S3 prennent en charge jusqu'à 10 balises par objet. Les balises associées à un objet doivent avoir des clés de balise uniques. Une clé de balise peut comporter jusqu'à 128 caractères Unicode et les valeurs de balise peuvent comporter jusqu'à 256 caractères Unicode. Les clés et les valeurs sont sensibles à la casse.

Propriété de l'objet

Dans StorageGRID, tous les objets appartiennent au compte propriétaire du compartiment, y compris les objets créés par un compte autre que celui du propriétaire ou par un utilisateur anonyme.

En-têtes de requête pris en charge

Les en-têtes de requête suivants sont pris en charge :

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

  Lorsque vous spécifiez `aws-chunked` pour `Content-Encoding`StorageGRID, ce dernier ne vérifie pas les éléments suivants :

  - StorageGRID ne vérifie pas les `chunk-signature` par rapport aux données du bloc.
  - StorageGRID ne vérifie pas la valeur que vous fournissez `x-amz-decoded-content-length` par rapport à l'objet.
- `Content-Language`
- `Content-Length`
- `Content-MD5`
- `Content-Type`
- `Expires`
- `Transfer-Encoding`

  Le codage par transfert segmenté est pris en charge si `aws-chunked` la signature de la charge utile est également utilisée.
- `x-amz-checksum-sha256`
- `x-amz-meta-`, suivie d'une paire nom-valeur contenant des métadonnées définies par l'utilisateur.

  Lors de la spécification de la paire nom-valeur pour les métadonnées définies par l'utilisateur, utilisez ce format général :

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

  Si vous souhaitez utiliser l'option **Heure de création définie par l'utilisateur** comme heure de référence pour une règle ILM, vous devez utiliser `creation-time` comme nom des métadonnées qui enregistrent la date de création de l'objet. Par exemple :

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

  La valeur pour `creation-time` est évaluée en secondes depuis le 1er janvier 1970.

  > Une règle ILM ne peut pas utiliser simultanément une **date de création définie par l'utilisateur** comme date de référence et l'option d'ingestion Équilibrée ou Stricte. Une erreur est renvoyée lors de la création de la règle ILM.

- `x-amz-tagging`
- En-têtes de requête S3 Object Lock
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    Si une requête est effectuée sans ces en-têtes, les paramètres de rétention par défaut du compartiment sont utilisés pour calculer le mode de version de l'objet et la date de rétention jusqu'à la date limite. Voir "Utilisez l'API REST S3 pour configurer S3 Object Lock".
- En-têtes de requête SSE :
  - `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`

    Voir En-têtes de requête pour le chiffrement côté serveur

En-têtes de requête non pris en charge

Les en-têtes de requête suivants ne sont pas pris en charge :

- `If-Match`

  Le `If-Match header` est accepté mais non fonctionnel.
- `If-None-Match`

  Le `If-None-Match header` est accepté mais non fonctionnel.
- `x-amz-acl`
- `x-amz-sdk-checksum-algorithm`
- `x-amz-trailer`
- `x-amz-website-redirect-location`

  L'`x-amz-website-redirect-location`en-tête renvoie `XNotImplemented`.

Options de classe de stockage

L' `x-amz-storage-class`en-tête de requête est pris en charge. La valeur soumise pour `x-amz-storage-class`influe sur la manière dont StorageGRID protège les données d'objet lors de l'ingestion, et non sur le nombre de copies persistantes de l'objet stockées dans le système StorageGRID (ce nombre étant déterminé par ILM).

Si la règle ILM correspondant à un objet ingéré utilise l'option d'ingestion stricte, l' `x-amz-storage-class`en-tête n'a aucun effet.

Les valeurs suivantes peuvent être utilisées pour `x-amz-storage-class` :

- `STANDARD` (Défaut)
  - **Double validation** : Si la règle ILM spécifie l'option « Double validation » pour le comportement d'ingestion, dès qu'un objet est ingéré, une seconde copie de cet objet est créée et distribuée sur un nœud de stockage différent (double validation). Lors de l'évaluation de l'ILM, StorageGRID détermine si ces copies intermédiaires initiales respectent les instructions de placement de la règle. Si ce n'est pas le cas, il peut être nécessaire de créer de nouvelles copies de l'objet à d'autres emplacements et de supprimer les copies intermédiaires initiales.
  - **Équilibré** : Si la règle ILM spécifie l'option Équilibré et StorageGRID ne peut pas effectuer immédiatement toutes les copies spécifiées dans la règle, StorageGRID effectue deux copies intermédiaires sur différents nœuds de stockage.

    Si StorageGRID peut créer immédiatement toutes les copies d'objets spécifiées dans la règle ILM (placement synchrone), l `x-amz-storage-class` en-tête n'a aucun effet.
- `REDUCED_REDUNDANCY`
  - **Double validation** : Si la règle ILM spécifie l'option de double validation pour le comportement d'ingestion, StorageGRID crée une seule copie intermédiaire lors de l'ingestion de l'objet (validation unique).
  - **Équilibré** : Si la règle ILM spécifie l'option Équilibré, StorageGRID crée une seule copie intermédiaire uniquement si le système ne peut pas créer immédiatement toutes les copies spécifiées dans la règle. Si StorageGRID peut effectuer un placement synchrone, cet en-tête est sans effet. L' `REDUCED_REDUNDANCY`option est particulièrement utile lorsque la règle ILM correspondant à l'objet crée une seule copie répliquée. Dans ce cas, utiliser `REDUCED_REDUNDANCY`évite la création et la suppression inutiles d'une copie supplémentaire de l'objet pour chaque opération d'ingestion.

  L'utilisation de l' `REDUCED_REDUNDANCY`option est déconseillée dans d'autres circonstances. `REDUCED_REDUNDANCY` Elle augmente le risque de perte de données lors de l'ingestion. Par exemple, vous pourriez perdre des données si la copie unique est initialement stockée sur un Storage Node qui tombe en panne avant que l'évaluation ILM puisse avoir lieu.

> Le fait de ne disposer que d'une seule copie répliquée pour une période donnée expose les données à un risque de perte définitive. Si une seule copie répliquée d'un objet existe, cet objet est perdu en cas de panne ou d'erreur grave d'un Storage Node. Vous perdez également temporairement l'accès à l'objet lors des procédures de maintenance telles que les mises à niveau.

La spécification de `REDUCED_REDUNDANCY` n'affecte que le nombre de copies créées lors de la première ingestion d'un objet. Elle n'affecte pas le nombre de copies de l'objet créées lors de l'évaluation de l'objet par les politiques ILM actives et n'entraîne pas le stockage des données à des niveaux de redondance inférieurs dans le système StorageGRID.

> Si vous ingérez un objet dans un compartiment avec S3 Object Lock activé, l' `REDUCED_REDUNDANCY`option est ignorée. Si vous ingérez un objet dans un compartiment Compliant hérité, l' `REDUCED_REDUNDANCY`option renvoie une erreur. StorageGRID effectuera toujours une double validation à l'ingestion pour garantir le respect des exigences de conformité.

En-têtes de requête pour le chiffrement côté serveur

Vous pouvez utiliser les en-têtes de requête suivants pour chiffrer un objet avec le chiffrement côté serveur. Les options SSE et SSE-C sont incompatibles.

- **SSE** : Utilisez l'en-tête suivant si vous souhaitez chiffrer l'objet avec une clé unique gérée par StorageGRID.
  - `x-amz-server-side-encryption`

    Lorsque l'en-tête `x-amz-server-side-encryption` n'est pas inclus dans la requête PutObject, le "paramètre de chiffrement des objets stockés" à l'échelle de la grille est omis de la réponse PutObject.

- ${post_edited_translations.segment}
  - `x-amz-server-side-encryption-customer-algorithm`: Précisez `AES256`.
  - `x-amz-server-side-encryption-customer-key`: Spécifiez votre clé de chiffrement pour le nouvel objet.
  - `x-amz-server-side-encryption-customer-key-MD5`: Spécifiez le condensé MD5 de la clé de chiffrement du nouvel objet.

> Les clés de chiffrement que vous fournissez ne sont jamais stockées. Si vous perdez une clé de chiffrement, vous perdez l'objet correspondant. Avant d'utiliser des clés fournies par le client pour sécuriser les données d'objet, examinez les considérations pour "utilisation du chiffrement côté serveur".

> Si un objet est chiffré avec SSE ou SSE-C, tous les paramètres de chiffrement au niveau du compartiment ou de la grille sont ignorés.

Versionnage

Si le versionnage est activé pour un compartiment, un identifiant unique `versionId` est automatiquement généré pour la version de l'objet stocké. Cet identifiant `versionId` est également renvoyé dans la réponse via l' `x-amz-version-id` en-tête de réponse.

Si le versionnage est suspendu, la version de l'objet est stockée avec une valeur nulle `versionId` et si une version nulle existe déjà, elle sera écrasée.

${post_edited_translations.segment}

Lors de l'utilisation de l' `Authorization`en-tête pour authentifier les requêtes, StorageGRID diffère d'AWS de la manière suivante :

- StorageGRID n'exige pas que les `host` en-têtes soient inclus dans `CanonicalHeaders`.
- StorageGRID n'a pas besoin que `Content-Type` soit inclus dans `CanonicalHeaders`.
- StorageGRID n'exige pas que les `x-amz-*` en-têtes soient inclus dans `CanonicalHeaders`.

> En bonne pratique, incluez toujours ces en-têtes `CanonicalHeaders` afin de garantir leur vérification ; toutefois, si vous les excluez, StorageGRID ne renvoie pas d'erreur.

Pour plus de détails, consultez "Calculs de signature pour l'en-tête d'autorisation : transfert de la charge utile en un seul bloc (AWS Signature Version 4)".

Informations connexes

- "Gérez les objets avec ILM"
- "Référence de l'API Amazon Simple Storage Service : PutObject"

Restaurez un objet dans StorageGRID avec la requête S3 RestoreObject

Vous pouvez utiliser la requête S3 RestoreObject pour restaurer un objet stocké dans un Cloud Storage Pool.

Type de requête pris en charge

StorageGRID prend uniquement en charge les requêtes RestoreObject pour restaurer un objet. Il ne prend pas en charge le type de restauration `SELECT`. Les requêtes Select renvoient `XNotImplemented`.

Versionnage

Vous pouvez, si vous le souhaitez, spécifier `versionId` une version particulière d'un objet dans un compartiment versionné. Si vous ne spécifiez `versionId`, la version la plus récente de l'objet est restaurée.

Comportement de RestoreObject sur les objets du Cloud Storage Pool

Si un objet a été stocké dans un "Pool de stockage cloud", une requête RestoreObject présente le comportement suivant, en fonction de l'état de l'objet. Consultez "HeadObject" pour plus de détails.

> Si un objet est stocké dans un Cloud Storage Pool et qu'une ou plusieurs copies de l'objet existent également sur la grille, il n'est pas nécessaire de restaurer l'objet en émettant une requête RestoreObject. À la place, la copie locale peut être récupérée directement à l'aide d'une requête GetObject.

État de l'objet	Comportement de RestoreObject
Objet ingéré dans StorageGRID mais pas encore évalué par ILM, ou l'objet ne se trouve pas dans un Cloud Storage Pool	`403 Forbidden`, `InvalidObjectState`
Objet présent dans le pool de stockage cloud mais n'ayant pas encore été transféré dans un état non récupérable	`200 OK` Aucune modification n'est apportée. **Remarque** : Avant qu'un objet ne soit passé à un état non récupérable, vous ne pouvez pas modifier son `expiry-date`.

État de l'objet	Comportement de RestoreObject
L'objet est passé dans un état non récupérable	`202 Accepted` Restaure une copie récupérable de l'objet dans le Cloud Storage Pool pour le nombre de jours spécifié dans le corps de la requête. À l'issue de cette période, l'objet est renvoyé à un état non récupérable. Vous pouvez éventuellement utiliser l' `Tier`élément de requête pour déterminer combien de temps la tâche de restauration prendra pour se terminer (`Expedited`, `Standard`, ou `Bulk`). Si vous ne spécifiez pas `Tier`, le `Standard`niveau est utilisé. **Important** : Si un objet a été transféré vers S3 Glacier Deep Archive ou si le Cloud Storage Pool utilise le stockage Azure Blob, vous ne pouvez pas le restaurer à l'aide du niveau `Expedited`. L'erreur suivante est renvoyée `403 Forbidden`, `InvalidTier`: `Retrieval option is not supported by this storage class`.
Objet en cours de restauration à partir d'un état non récupérable	`409 Conflict`, `RestoreAlreadyInProgress`
Objet entièrement restauré dans le Cloud Storage Pool	`200 OK` **Remarque :** Si un objet a été restauré à un état récupérable, vous pouvez modifier son `expiry-date` en réémettant la requête RestoreObject avec une nouvelle valeur pour `Days`. La date de restauration est mise à jour par rapport à l'heure de la requête.

Filtrer les données d'objets dans StorageGRID avec la requête S3 SelectObjectContent

Vous pouvez utiliser la requête S3 SelectObjectContent pour filtrer le contenu d'un objet S3 en fonction d'une instruction SQL simple.

Pour plus d'informations, consultez "Référence de l'API Amazon Simple Storage Service : SelectObjectContent".

Avant de commencer

- ${post_edited_translations.segment}
- Vous disposez `s3:GetObject` des autorisations nécessaires pour l'objet que vous souhaitez interroger.
- L'objet que vous souhaitez interroger doit être dans l'un des formats suivants :
  - **CSV**. Peut être utilisé tel quel ou compressé dans des archives GZIP ou BZIP2.
  - **Parquet**. Exigences supplémentaires pour les objets Parquet :
    - S3 Select prend uniquement en charge la compression par colonnes via GZIP ou Snappy. S3 Select ne prend pas en charge la compression de l'objet entier pour les objets Parquet.
    - S3 Select ne prend pas en charge la sortie Parquet. Vous devez spécifier le format de sortie comme CSV ou JSON.
    - La taille maximale d'un groupe de lignes non compressé est de 512 Mo.

        - Vous devez utiliser les types de données spécifiés dans le schéma de l'objet.
        - Vous ne pouvez pas utiliser les types logiques INTERVAL, JSON, LIST, TIME ou UUID.
- Votre expression SQL a une longueur maximale de 256 Ko.
- ${post_edited_translations.segment}

Exemple de syntaxe de requête CSV

```
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>
```

Exemple de syntaxe de requête Parquet

```
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>
```

Exemple de requête SQL

Cette requête récupère le nom de l'État, la population de 2010, la population estimée de 2015 et le pourcentage de variation à partir des données de recensement des États-Unis. Les enregistrements du fichier qui ne sont pas des États sont ignorés.

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

Les premières lignes du fichier à interroger, `SUB-EST2020_ALL.csv`, se présentent ainsi :

```
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
```

${post_edited_translations.segment}

```
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
```

Les premières lignes du fichier de sortie, `changes.csv`, ressemblent à ceci :

```
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
```

${post_edited_translations.segment}

```
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
```

Les premières lignes du fichier de sortie, changes.csv, ressemblent à ceci :

```
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
```

${post_edited_translations.segment}

Opérations de chargement multiparties prises en charge dans StorageGRID

Cette section décrit comment StorageGRID prend en charge les opérations de chargement en plusieurs parties.

Les conditions et remarques suivantes s'appliquent à toutes les opérations de téléchargement en plusieurs parties :

- Vous ne devez pas dépasser 1 000 chargements multiparties simultanés vers un seul compartiment, car les résultats des requêtes ListMultipartUploads pour ce compartiment pourraient être incomplets.
- StorageGRID applique les limites de taille AWS pour les parties multipart. Les clients S3 doivent respecter les consignes suivantes :
  - Chaque partie d'un téléchargement en plusieurs parties doit être comprise entre 5 Mio (5 242 880 octets) et 5 Gio (5 368 709 120 octets).
  - La dernière partie peut être inférieure à 5 Mio (5,242,880 octets).
  - En règle générale, la taille des parties doit être aussi grande que possible. Par exemple, utilisez des parties de 5 Gio pour un objet de 100 Gio. Comme chaque partie est considérée comme un objet unique, l'utilisation de grandes tailles de parties réduit la surcharge des métadonnées StorageGRID.
  - Pour les objets de moins de 5 Gio, envisagez d'utiliser le téléchargement non multipart à la place.
- L'ILM est évalué pour chaque partie d'un objet multipartie lors de son ingestion, puis pour l'objet dans son ensemble une fois le chargement multipartie terminé, si la règle ILM utilise le mode Balanced ou Strict "option d'ingestion". Vous devez être conscient de la façon dont cela affecte le placement des objets et des parties :
  - Si les paramètres ILM sont modifiés pendant un chargement multipartie vers S3, certaines parties de

l'objet risquent de ne plus respecter les exigences ILM en vigueur une fois le chargement terminé. Toute partie qui n'est pas placée correctement est mise en file d'attente pour une réévaluation ILM et déplacée ultérieurement vers l'emplacement correct.

  - Lors de l'évaluation ILM d'une partie, StorageGRID filtre sur la taille de la partie, et non sur celle de l'objet. Cela signifie que des parties d'un objet peuvent être stockées dans des emplacements ne répondant pas aux exigences ILM de l'objet dans son ensemble. Par exemple, si une règle spécifie que tous les objets de 10 Go ou plus sont stockés à DC1 tandis que tous les objets plus petits sont stockés à DC2, chaque partie de 1 Go d'un chargement multipartie en 10 parties est stockée à DC2 lors de l'ingestion. Cependant, lors de l'évaluation ILM de l'objet dans son ensemble, toutes les parties de l'objet sont déplacées vers DC1.

- Toutes les opérations de téléchargement en plusieurs parties prennent en charge StorageGRID "valeurs de cohérence".
- Lorsqu'un objet est ingéré à l'aide d'un téléchargement multipart, le "seuil de segmentation d'objet (1 Gio)" n'est pas appliqué.
- Selon les besoins, vous pouvez utiliser "chiffrement côté serveur" avec les chargements en plusieurs parties. Pour utiliser SSE (chiffrement côté serveur avec des clés gérées par StorageGRID), vous devez inclure l' `x-amz-server-side-encryption` en-tête de requête uniquement dans la requête CreateMultipartUpload. Pour utiliser SSE-C (chiffrement côté serveur avec des clés fournies par le client), vous devez spécifier les mêmes trois en-têtes de requête de clé de chiffrement dans la requête CreateMultipartUpload et dans chaque requête UploadPart suivante.
- Un objet téléchargé en plusieurs parties est inclus dans un "bucket de branche" si l'ingestion a été initiée avant l'horodatage Before du compartiment de base, quelle que soit la date de fin du téléchargement.

Opération	Mise en œuvre
AbortMultipartUpload	Implémenté avec l'ensemble du comportement de l'API REST Amazon S3. Sous réserve de modifications sans préavis.
CompleteMultipartUpload	Voir "CompleteMultipartUpload"
CreateMultipartUpload (anciennement nommé Initiate Multipart Upload)	Voir "CreateMultipartUpload"
ListMultipartUploads	Voir "ListMultipartUploads"
ListParts	Implémenté avec l'ensemble du comportement de l'API REST Amazon S3. Sous réserve de modifications sans préavis.
UploadPart	Voir "UploadPart"
UploadPartCopy	Voir "UploadPartCopy"

Comment l'API REST S3 de StorageGRID effectue les chargements en plusieurs parties

L'opération CompleteMultipartUpload finalise le chargement en plusieurs parties d'un objet en assemblant les parties précédemment chargées.

StorageGRID prend en charge les valeurs non consécutives par ordre croissant pour le paramètre de requête `partNumber` avec CompleteMultipartUpload. Le paramètre peut commencer par n'importe quelle valeur.

Résoudre les conflits

Les requêtes clients conflictuelles, comme par exemple deux clients écrivant sur la même clé, sont résolues selon le principe du « latest-wins ». Le moment de l'évaluation du « latest-wins » est déterminé par la date à laquelle le système StorageGRID termine une requête donnée, et non par la date à laquelle les clients S3 commencent une opération.

En-têtes de requête pris en charge

Les en-têtes de requête suivants sont pris en charge :

- `x-amz-checksum-sha256`
- `x-amz-storage-class`

  L' `x-amz-storage-class`en-tête affecte le nombre de copies d'objets que StorageGRID crée si la règle ILM correspondante spécifie le "Option de double validation ou d'ingestion équilibrée".

- `STANDARD`

  (Par défaut) Spécifie une opération d'ingestion à double validation lorsque la règle ILM utilise l'option Dual commit, ou lorsque l'option Balanced revient à créer des copies intermédiaires.

- `REDUCED_REDUNDANCY`

  Spécifie une opération d'ingestion à validation unique lorsque la règle ILM utilise l'option de double validation, ou lorsque l'option équilibrée revient à créer des copies intermédiaires.

  Si vous ingérez un objet dans un compartiment avec S3 Object Lock activé, l' `REDUCED_REDUNDANCY`option est ignorée. Si vous ingérez un objet dans un compartiment Compliant hérité, l' `REDUCED_REDUNDANCY`option renvoie une erreur. StorageGRID effectuera toujours une double validation à l'ingestion pour garantir le respect des exigences de conformité.

Si un chargement en plusieurs parties n'est pas terminé dans un délai de 15 jours, l'opération est marquée comme inactive et toutes les données associées sont supprimées du système.

La `ETag`valeur renvoyée n'est pas une somme MD5 des données, mais suit l'implémentation de l'API Amazon S3 de la `ETag`valeur pour les objets multipart.

En-têtes de requête non pris en charge

Les en-têtes de requête suivants ne sont pas pris en charge :

- `If-Match`

  Le `If-Match header` est accepté mais non fonctionnel.

- `If-None-Match`

  Le `If-None-Match header` est accepté mais non fonctionnel.

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

Versionnage

Cette opération finalise un chargement en plusieurs parties. Si le versionnage est activé pour un compartiment, la version de l'objet est créée après la finalisation du chargement en plusieurs parties.

Si le versionnage est activé pour un compartiment, un identifiant unique `versionId` est automatiquement généré pour la version de l'objet stocké. Cet identifiant `versionId` est également renvoyé dans la réponse via l' `x-amz-version-id` en-tête de réponse.

Si le versionnage est suspendu, la version de l'objet est stockée avec une valeur nulle `versionId` et si une version nulle existe déjà, elle sera écrasée.

> Lorsque le versionnage est activé pour un bucket, la finalisation d'un chargement multipart crée toujours une nouvelle version, même si des chargements multipart simultanés sont finalisés sur la même clé d'objet. Lorsque le versionnage n'est pas activé pour un bucket, il est possible d'initier un chargement multipart, puis qu'un autre chargement multipart soit initié et finalisé en premier sur la même clé d'objet. Sur les buckets non versionnés, le chargement multipart finalisé en dernier prévaut.

Échec de la réplication, de la notification ou de la notification des métadonnées

Si le compartiment dans lequel s'effectue le chargement en plusieurs parties est configuré pour un service de plateforme, le chargement en plusieurs parties réussit même si l'action de réplication ou de notification associée échoue.

Un locataire peut déclencher l'échec de la réplication ou la notification en mettant à jour les métadonnées ou les balises de l'objet. Un locataire peut soumettre à nouveau les valeurs existantes afin d'éviter toute modification indésirable.

Consultez "Dépannage des services de plateforme".

En-têtes et options des requêtes S3 CreateMultipartUpload dans StorageGRID

L'opération CreateMultipartUpload (anciennement appelée Initiate Multipart Upload) lance un téléchargement en plusieurs parties pour un objet et renvoie un ID de téléchargement.

L' `x-amz-storage-class`en-tête de requête est pris en charge. La valeur soumise pour `x-amz-storage-class`influe sur la manière dont StorageGRID protège les données d'objet lors de l'ingestion, et non sur le nombre de copies persistantes de l'objet stockées dans le système StorageGRID (ce nombre étant déterminé par ILM).

Si la règle ILM correspondant à un objet ingéré utilise l'option Strict "option d'ingestion", l' `x-amz-storage-class`en-tête n'a aucun effet.

Les valeurs suivantes peuvent être utilisées pour `x-amz-storage-class` :

- `STANDARD` (Défaut)
  - **Double validation** : Si la règle ILM spécifie l'option d'ingestion « Double validation », dès qu'un objet est ingéré, une seconde copie de cet objet est créée et distribuée sur un nœud de stockage différent (double validation). Lors de l'évaluation de l'ILM, StorageGRID détermine si ces copies intermédiaires initiales respectent les instructions de placement de la règle. Si ce n'est pas le cas, il peut être nécessaire de créer de nouvelles copies de l'objet à d'autres emplacements et de supprimer les copies intermédiaires initiales.
  - **Équilibré** : Si la règle ILM spécifie l'option Équilibré et StorageGRID ne peut pas effectuer immédiatement toutes les copies spécifiées dans la règle, StorageGRID effectue deux copies intermédiaires sur différents nœuds de stockage.

    Si StorageGRID peut créer immédiatement toutes les copies d'objets spécifiées dans la règle ILM (placement synchrone), l `x-amz-storage-class` en-tête n'a aucun effet.
- `REDUCED_REDUNDANCY`
  - **Double validation** : Si la règle ILM spécifie l'option Double validation, StorageGRID crée une seule copie intermédiaire lors de l'ingestion de l'objet (validation unique).
  - **Équilibré** : Si la règle ILM spécifie l'option Équilibré, StorageGRID crée une seule copie intermédiaire uniquement si le système ne peut pas créer immédiatement toutes les copies spécifiées dans la règle. Si StorageGRID peut effectuer un placement synchrone, cet en-tête est sans effet. L' `REDUCED_REDUNDANCY`option est particulièrement utile lorsque la règle ILM correspondant à l'objet crée une seule copie répliquée. Dans ce cas, utiliser `REDUCED_REDUNDANCY`évite la création et la suppression inutiles d'une copie supplémentaire de l'objet pour chaque opération d'ingestion.

  L'utilisation de l' `REDUCED_REDUNDANCY`option est déconseillée dans d'autres circonstances. `REDUCED_REDUNDANCY` Elle augmente le risque de perte de données lors de l'ingestion. Par exemple, vous pourriez perdre des données si la copie unique est initialement stockée sur un Storage Node qui tombe en panne avant que l'évaluation ILM puisse avoir lieu.

> Le fait de ne disposer que d'une seule copie répliquée pour une période donnée expose les données à un risque de perte définitive. Si une seule copie répliquée d'un objet existe, cet objet est perdu en cas de panne ou d'erreur grave d'un Storage Node. Vous perdez également temporairement l'accès à l'objet lors des procédures de maintenance telles que les mises à niveau.

La spécification de `REDUCED_REDUNDANCY` n'affecte que le nombre de copies créées lors de la première ingestion d'un objet. Elle n'affecte pas le nombre de copies de l'objet créées lors de l'évaluation de l'objet par les politiques ILM actives et n'entraîne pas le stockage des données à des niveaux de redondance inférieurs dans le système StorageGRID.

> Si vous ingérez un objet dans un compartiment avec S3 Object Lock activé, l' `REDUCED_REDUNDANCY`option est ignorée. Si vous ingérez un objet dans un compartiment Compliant hérité, l' `REDUCED_REDUNDANCY`option renvoie une erreur. StorageGRID effectuera toujours une double validation à l'ingestion pour garantir le respect des exigences de conformité.

En-têtes de requête pris en charge

Les en-têtes de requête suivants sont pris en charge :

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

  Actuellement, seule la valeur SHA256 pour `x-amz-checksum-algorithm` est prise en charge.

- `x-amz-meta-`, suivi d'une paire nom-valeur contenant des métadonnées définies par l'utilisateur

  Lors de la spécification de la paire nom-valeur pour les métadonnées définies par l'utilisateur, utilisez ce format général :

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

  Si vous souhaitez utiliser l'option **Heure de création définie par l'utilisateur** comme heure de référence pour une règle ILM, vous devez utiliser `creation-time` comme nom des métadonnées qui enregistrent la date de création de l'objet. Par exemple :

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

  La valeur pour `creation-time` est évaluée en secondes depuis le 1er janvier 1970.

  > L'ajout `creation-time` en tant que métadonnées définies par l'utilisateur n'est pas autorisé si vous ajoutez un objet à un compartiment avec la conformité héritée activée. Une erreur sera renvoyée.

- En-têtes de requête S3 Object Lock :
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

    Si une requête est effectuée sans ces en-têtes, les paramètres de rétention par défaut du compartiment sont utilisés pour calculer la date de rétention jusqu'à la version de l'objet.

  "Utilisez l'API REST S3 pour configurer S3 Object Lock"

- En-têtes de requête SSE :
  - `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`

    En-têtes de requête pour le chiffrement côté serveur

Pour plus d'informations sur la façon dont StorageGRID gère les caractères UTF-8, consultez "PutObject".

En-têtes de requête pour le chiffrement côté serveur

Vous pouvez utiliser les en-têtes de requête suivants pour chiffrer un objet multipartite avec le chiffrement côté serveur. Les options SSE et SSE-C sont mutuellement exclusives.

- **SSE** : Utilisez l'en-tête suivant dans la requête CreateMultipartUpload si vous souhaitez chiffrer l'objet avec une clé unique gérée par StorageGRID. Ne spécifiez pas cet en-tête dans aucune des requêtes UploadPart.
  - `x-amz-server-side-encryption`
- **SSE-C** : Utilisez ces trois en-têtes dans la requête CreateMultipartUpload (et dans chaque requête UploadPart suivante) si vous souhaitez chiffrer l'objet avec une clé unique que vous fournissez et gérez.
  - `x-amz-server-side-encryption-customer-algorithm`: Précisez `AES256`.
  - `x-amz-server-side-encryption-customer-key`: Spécifiez votre clé de chiffrement pour le nouvel objet.
  - `x-amz-server-side-encryption-customer-key-MD5`: Spécifiez le condensé MD5 de la clé de chiffrement du nouvel objet.

Les clés de chiffrement que vous fournissez ne sont jamais stockées. Si vous perdez une clé de chiffrement, vous perdez l'objet correspondant. Avant d'utiliser des clés fournies par le client pour sécuriser les données d'objet, examinez les considérations pour "utilisation du chiffrement côté serveur".

En-têtes de requête non pris en charge

L'en-tête de requête suivant n'est pas pris en charge :

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

  L'`x-amz-website-redirect-location`en-tête renvoie `XNotImplemented`.

Versionnage

Le chargement en plusieurs parties comprend des opérations distinctes pour lancer le chargement, lister les chargements, charger les différentes parties, assembler les parties chargées et finaliser le chargement. Les objets sont créés (et versionnés le cas échéant) lorsque l'opération CompleteMultipartUpload est effectuée.

Fonctionnement S3 ListMultipartUploads et paramètres pris en charge dans StorageGRID

L'opération ListMultipartUploads répertorie les chargements multipart en cours pour un compartiment.

Les paramètres de requête suivants sont pris en charge :

- `encoding-type`
- `key-marker`

- `max-uploads`
- `prefix`
- `upload-id-marker`
- `Host`
- `Date`
- `Authorization`

Versionnage

Le chargement en plusieurs parties comprend des opérations distinctes pour lancer le chargement, lister les chargements, charger les différentes parties, assembler les parties chargées et finaliser le chargement. Les objets sont créés (et versionnés le cas échéant) lorsque l'opération CompleteMultipartUpload est effectuée.

En-têtes de requête pris en charge et non pris en charge pour les opérations S3 UploadPart dans StorageGRID

L'opération UploadPart télécharge une partie dans un chargement multipart pour un objet.

En-têtes de requête pris en charge

Les en-têtes de requête suivants sont pris en charge :

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

En-têtes de requête pour le chiffrement côté serveur

Si vous avez spécifié le chiffrement SSE-C pour la requête CreateMultipartUpload, vous devez également inclure les en-têtes de requête suivants dans chaque requête UploadPart :

- `x-amz-server-side-encryption-customer-algorithm`: Précisez `AES256`.
- `x-amz-server-side-encryption-customer-key`: Spécifiez la même clé de chiffrement que celle que vous avez fournie dans la requête CreateMultipartUpload.
- `x-amz-server-side-encryption-customer-key-MD5` : Veuillez indiquer le même condensé MD5 que celui que vous avez fourni dans la requête CreateMultipartUpload.

> Les clés de chiffrement que vous fournissez ne sont jamais stockées. Si vous perdez une clé de chiffrement, vous perdez l'objet correspondant. Avant d'utiliser des clés fournies par le client pour sécuriser les données d'objet, examinez les considérations dans "Utilisez le chiffrement côté serveur".

Si vous avez spécifié une somme de contrôle SHA-256 lors de la requête CreateMultipartUpload, vous devez également inclure l'en-tête de requête suivant dans chaque requête UploadPart :

- `x-amz-checksum-sha256`: Spécifiez la somme de contrôle SHA-256 pour cette partie.

En-têtes de requête non pris en charge

Les en-têtes de requête suivants ne sont pas pris en charge :

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

Versionnage

Le chargement en plusieurs parties comprend des opérations distinctes pour lancer le chargement, lister les chargements, charger les différentes parties, assembler les parties chargées et finaliser le chargement. Les objets sont créés (et versionnés le cas échéant) lorsque l'opération CompleteMultipartUpload est effectuée.

Chargez une partie d'un objet dans StorageGRID avec l'opération S3 UploadPartCopy

L'opération UploadPartCopy téléverse une partie d'un objet en copiant les données d'un objet existant comme source de données.

L'opération UploadPartCopy est implémentée avec l'ensemble du comportement de l'API REST Amazon S3. Sous réserve de modifications sans préavis.

Cette requête lit et écrit les données de l'objet spécifié dans `x-amz-copy-source-range` le système StorageGRID.

Les en-têtes de requête suivants sont pris en charge :

- `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`

En-têtes de requête pour le chiffrement côté serveur

Si vous avez spécifié le chiffrement SSE-C pour la requête CreateMultipartUpload, vous devez également inclure les en-têtes de requête suivants dans chaque requête UploadPartCopy :

- `x-amz-server-side-encryption-customer-algorithm`: Précisez `AES256`.
- `x-amz-server-side-encryption-customer-key`: Spécifiez la même clé de chiffrement que celle que vous avez fournie dans la requête CreateMultipartUpload.
- `x-amz-server-side-encryption-customer-key-MD5` : Veuillez indiquer le même condensé MD5 que celui que vous avez fourni dans la requête CreateMultipartUpload.

Si l'objet source est chiffré à l'aide d'une clé fournie par le client (SSE-C), vous devez inclure les trois en-têtes suivants dans la requête UploadPartCopy, afin que l'objet puisse être déchiffré puis copié :

- `x-amz-copy-source-server-side-encryption-customer-algorithm`: Précisez `AES256`.
- `x-amz-copy-source-server-side-encryption-customer-key`: Spécifiez la clé de chiffrement que vous avez fournie lors de la création de l'objet source.
- `x-amz-copy-source-server-side-encryption-customer-key-MD5`: Spécifiez le condensé MD5 que vous avez fourni lors de la création de l'objet source.

Les clés de chiffrement que vous fournissez ne sont jamais stockées. Si vous perdez une clé de chiffrement, vous perdez l'objet correspondant. Avant d'utiliser des clés fournies par le client pour sécuriser les données d'objet, examinez les considérations dans "Utilisez le chiffrement côté serveur".

Versionnage

Le chargement en plusieurs parties comprend des opérations distinctes pour lancer le chargement, lister les chargements, charger les différentes parties, assembler les parties chargées et finaliser le chargement. Les objets sont créés (et versionnés le cas échéant) lorsque l'opération CompleteMultipartUpload est effectuée.

Réponses d'erreur de l'API REST StorageGRID S3

Le système StorageGRID prend en charge toutes les réponses d'erreur standard de l'API REST S3 qui s'appliquent. De plus, l'implémentation StorageGRID ajoute plusieurs réponses personnalisées.

Codes d'erreur de l'API S3 pris en charge

Nom	statut HTTP
AccessDenied	403 Interdit
BadDigest	400 Mauvaise requête
BucketAlreadyExists	409 Conflit
BucketNotEmpty	409 Conflit
IncompleteBody	400 Mauvaise requête
InternalError	Erreur interne du serveur 500
InvalidAccessKeyId	403 Interdit
InvalidArgument	400 Mauvaise requête
InvalidBucketName	400 Mauvaise requête
InvalidBucketState	409 Conflit
InvalidDigest	400 Mauvaise requête
InvalidEncryptionAlgorithmError	400 Mauvaise requête
InvalidPart	400 Mauvaise requête

Nom	statut HTTP
InvalidPartOrder	400 Mauvaise requête
InvalidRange	416 Plage demandée non satisfaisable
InvalidRequest	400 Mauvaise requête
InvalidStorageClass	400 Mauvaise requête
InvalidTag	400 Mauvaise requête
URI invalide	400 Mauvaise requête
KeyTooLong	400 Mauvaise requête
XML malformé	400 Mauvaise requête
MetadataTooLarge	400 Mauvaise requête
MethodNotAllowed	405 Méthode non autorisée
MissingContentLength	411 Longueur requise
MissingRequestBodyError	400 Mauvaise requête
MissingSecurityHeader	400 Mauvaise requête
NoSuchBucket	404 Introuvable
NoSuchKey	404 Introuvable
NoSuchUpload	404 Introuvable
NotImplemented	501 Non implémenté
NoSuchBucketPolicy	404 Introuvable
ObjectLockConfigurationNotFoundError	404 Introuvable
PreconditionFailed	412 Précondition non remplie
RequestTimeTooSkewed	403 Interdit
ServiceUnavailable	503 Service indisponible

Nom	statut HTTP
SignatureDoesNotMatch	403 Interdit
TooManyBuckets	400 Mauvaise requête
UserKeyMustBeSpecified	400 Mauvaise requête

Codes d'erreur personnalisés StorageGRID

Nom	Description	statut HTTP
XBucketLifecycleNotAllowed	La configuration du cycle de vie du bucket n'est pas autorisée dans un bucket Compliant hérité	400 Mauvaise requête
Exception d'analyse de la politique de compartiment	Échec de l'analyse du JSON de la politique de compartiment reçue.	400 Mauvaise requête
XComplianceConflict	Opération refusée en raison de paramètres de conformité hérités.	403 Interdit
XComplianceReducedRedundancyForbidden	La redondance réduite n'est pas autorisée dans le compartiment Compliant hérité	400 Mauvaise requête
XMaxBucketPolicyLengthExceeded	Votre stratégie dépasse la longueur maximale autorisée pour une stratégie de compartiment.	400 Mauvaise requête
XMissingInternalRequestHeader	En-tête manquant d'une requête interne.	400 Mauvaise requête
XNoSuchBucketCompliance	Le compartiment spécifié n'a pas la conformité héritée activée.	404 Introuvable
XNotAcceptable	La requête contient un ou plusieurs en-têtes Accept qui n'ont pas pu être satisfaits.	406 Non acceptable
XNotImplemented	La demande que vous avez fournie implique une fonctionnalité qui n'est pas implémentée.	501 Non implémenté

Opérations personnalisées StorageGRID

Opérations personnalisées de compartiment prises en charge dans StorageGRID

Le système StorageGRID prend en charge les opérations personnalisées qui sont ajoutées à l'API REST S3.

Le tableau suivant répertorie les opérations personnalisées prises en charge par StorageGRID.

Opération	Description
"Cohérence du bucket GET"	Renvoie la cohérence appliquée à un compartiment particulier.
"Cohérence PUT Bucket"	Définit la cohérence appliquée à un compartiment particulier.
"Obtenir le temps d'accès du compartiment"	Indique si les mises à jour du temps d'accès sont activées ou désactivées pour un compartiment donné.
"PUT temps d'accès du dernier accès au compartiment"	Permet d'activer ou de désactiver les mises à jour du temps d'accès pour un compartiment particulier.
"SUPPRIMER la configuration de notification des métadonnées du compartiment"	Supprime le fichier XML de configuration des notifications de métadonnées associé à un compartiment particulier.
"Obtenir la configuration de notification des métadonnées du bucket"	Renvoie le fichier XML de configuration des notifications de métadonnées associé à un compartiment particulier.
"Configuration de notification des métadonnées du compartiment PUT"	Configure le service de notification des métadonnées pour un compartiment.
"Obtenir l'utilisation du stockage"	Indique la capacité totale de stockage utilisée par un compte et pour chaque compartiment associé au compte.
"Déprécié : CreateBucket avec paramètres de conformité"	Obsolète et non pris en charge : vous ne pouvez plus créer de nouveaux compartiments avec la conformité activée.
"Obsolète : GET Bucket compliance"	Déprécié mais toujours pris en charge : renvoie les paramètres de conformité actuellement en vigueur pour un compartiment Compliant hérité existant.
"Obsolète : PUT Bucket compliance"	Obsolète mais pris en charge : permet de modifier les paramètres de conformité d'un compartiment Compliant hérité existant.

Récupérez le niveau de cohérence du compartiment dans StorageGRID avec la requête GET Bucket consistency

La requête GET Bucket consistency vous permet de déterminer la cohérence appliquée à

un bucket particulier.

La cohérence par défaut est définie pour garantir la lecture après écriture des objets nouvellement créés.

Vous devez disposer de l'autorisation s3:GetBucketConsistency, ou être administrateur du compte, pour effectuer cette opération.

Exemple de requête

```
GET /bucket?x-ntap-sg-consistency HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

Réponse

Dans la réponse XML, `<Consistency>` renverra l'une des valeurs suivantes :

Cohérence	Description
tous	Tous les nœuds reçoivent les données immédiatement, ou la requête échouera.
fort-mondial	Garantit la cohérence en lecture après écriture pour toutes les requêtes client sur tous les sites.
site fort	Garantit la cohérence des données en lecture après écriture pour toutes les requêtes client au sein d'un site.
lecture après une nouvelle écriture	(Par défaut) Assure la cohérence en lecture après écriture pour les nouveaux objets et la cohérence éventuelle pour les mises à jour d'objets. Offre une haute disponibilité et des garanties de protection des données. Recommandé dans la plupart des cas.
disponible	Garantit la cohérence éventuelle des nouveaux objets et des mises à jour d'objets. Pour les compartiments S3, utilisez cette fonctionnalité uniquement lorsque cela est nécessaire (par exemple, pour un compartiment contenant des valeurs de journal rarement lues, ou pour les opérations HEAD ou GET sur des clés qui n'existent pas). Non pris en charge pour les compartiments S3 FabricPool.

Exemple de réponse

```
HTTP/1.1 200 OK
Date: Fri, 18 Sep 2020 01:02:18 GMT
Connection: CLOSE
Server: StorageGRID/11.5.0
x-amz-request-id: 12345
Content-Length: 127
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<Consistency xmlns="http://s3.storagegrid.com/doc/2015-02-01/">read-after-
new-write</Consistency>
```

Informations connexes

"Cohérence"

Spécifiez les niveaux de cohérence dans StorageGRID avec la requête PUT Bucket consistency

${post_edited_translations.segment}

La cohérence par défaut est définie pour garantir la lecture après écriture des objets nouvellement créés.

Avant de commencer

Vous devez disposer de l'autorisation s3:PutBucketConsistency, ou être le compte racine, pour effectuer cette opération.

Demande

Le paramètre `x-ntap-sg-consistency` doit contenir l'une des valeurs suivantes :

Cohérence	Description
tous	Tous les nœuds reçoivent les données immédiatement, ou la requête échouera.
fort-mondial	Garantit la cohérence en lecture après écriture pour toutes les requêtes client sur tous les sites.
site fort	Garantit la cohérence des données en lecture après écriture pour toutes les requêtes client au sein d'un site.
lecture après une nouvelle écriture	(Par défaut) Assure la cohérence en lecture après écriture pour les nouveaux objets et la cohérence éventuelle pour les mises à jour d'objets. Offre une haute disponibilité et des garanties de protection des données. Recommandé dans la plupart des cas.

Cohérence	Description
disponible	Garantit la cohérence éventuelle des nouveaux objets et des mises à jour d'objets. Pour les compartiments S3, utilisez cette fonctionnalité uniquement lorsque cela est nécessaire (par exemple, pour un compartiment contenant des valeurs de journal rarement lues, ou pour les opérations HEAD ou GET sur des clés qui n'existent pas). Non pris en charge pour les compartiments S3 FabricPool.

Remarque : En général, vous devez utiliser la cohérence « Lecture après nouvelle écriture ». Si les requêtes ne fonctionnent pas correctement, modifiez le comportement du client applicatif si possible. Ou configurez le client pour spécifier la cohérence pour chaque requête API. Définissez la cohérence au niveau du compartiment uniquement en dernier recours.

Exemple de requête

```
PUT /bucket?x-ntap-sg-consistency=strong-global HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

Informations connexes

"Cohérence"

Récupérez l'état de la dernière date d'accès au bucket dans StorageGRID avec la requête GET Bucket last access time

La requête GET Bucket last access time vous permet de déterminer si les mises à jour du temps d'accès sont activées ou désactivées pour chaque compartiment.

Vous devez disposer de l'autorisation s3:GetBucketLastAccessTime, ou être administrateur du compte, pour effectuer cette opération.

Exemple de requête

```
GET /bucket?x-ntap-sg-lastaccesstime HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

Exemple de réponse

Cet exemple montre que les mises à jour du dernier temps d'accès sont activées pour le compartiment.

```
HTTP/1.1 200 OK
Date: Sat, 29 Nov 2015 01:02:18 GMT
Connection: CLOSE
Server: StorageGRID/10.3.0
x-amz-request-id: 12345
Content-Length: 127
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<LastAccessTime xmlns="http://s3.storagegrid.com/doc/2015-02-01/">enabled
</LastAccessTime>
```

Activer et désactiver les mises à jour du temps d'accès dans StorageGRID avec la requête PUT Bucket last access time

La requête PUT Bucket last access time vous permet d'activer ou de désactiver les mises à jour du temps d'accès pour les compartiments individuels. La désactivation des mises à jour du temps d'accès améliore les performances et constitue le paramètre par défaut pour tous les compartiments créés avec la version 10.3.0 ou ultérieure.

Vous devez disposer de l'autorisation s3:PutBucketLastAccessTime pour un compartiment, ou être root du compte, pour effectuer cette opération.

> ⓘ À partir de la version 10.3 de StorageGRID, les mises à jour du temps d'accès sont désactivées par défaut pour tous les nouveaux compartiments. Si vous avez des compartiments créés avec une version antérieure de StorageGRID et que vous souhaitez adopter le nouveau comportement par défaut, vous devez explicitement désactiver les mises à jour du temps d'accès pour chacun de ces compartiments. Vous pouvez activer ou désactiver les mises à jour du temps d'accès à l'aide de la requête PUT Bucket last access time ou depuis la page de détails d'un compartiment dans le Tenant Manager. Voir "Activer ou désactiver la mise à jour du temps d'accès".

${post_edited_translations.segment}

- GetObject, GetObjectAcl, GetObjectTagging, et HeadObject ne mettent pas à jour le temps d'accès. L'objet n'est pas ajouté aux files d'attente pour l'évaluation de la gestion du cycle de vie de l'information (ILM).
- Les requêtes CopyObject et PutObjectTagging qui mettent à jour uniquement les métadonnées mettent également à jour le temps d'accès. L'objet est ajouté aux files d'attente pour l'évaluation ILM.
- Si les mises à jour du temps d'accès sont désactivées pour le compartiment source, les requêtes CopyObject ne mettent pas à jour le temps d'accès pour le compartiment source. L'objet copié n'est pas ajouté aux files d'attente pour l'évaluation ILM du compartiment source. Cependant, pour la destination, les requêtes CopyObject mettent toujours à jour le temps d'accès. La copie de l'objet est ajoutée aux files d'attente pour l'évaluation ILM.
- CompleteMultipartUpload les requêtes mettent à jour le temps d'accès. L'objet finalisé est ajouté aux files d'attente pour l'évaluation ILM.

Exemples de requêtes

Cet exemple active le dernier temps d'accès pour un compartiment.

```
PUT /bucket?x-ntap-sg-lastaccesstime=enabled HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

Cet exemple désactive le temps d'accès pour un compartiment.

```
PUT /bucket?x-ntap-sg-lastaccesstime=disabled HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

Désactivez le service d'intégration de recherche avec la requête DELETE Bucket metadata notification configuration dans StorageGRID

La requête de configuration de notification de métadonnées DELETE Bucket vous permet de désactiver le service d'intégration de recherche pour des compartiments individuels en supprimant le fichier XML de configuration.

Vous devez disposer de l'autorisation s3:DeleteBucketMetadataNotification pour un compartiment, ou être root du compte, pour effectuer cette opération.

Exemple de requête

Cet exemple montre comment désactiver le service d'intégration de recherche pour un compartiment.

```
DELETE /test1?x-ntap-sg-metadata-notification HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

Configurez l'intégration de la recherche dans StorageGRID avec la requête de configuration de notification des métadonnées du compartiment GET

La requête de configuration de notification des métadonnées GET Bucket vous permet de récupérer le fichier XML de configuration utilisé pour configurer l'intégration de la recherche pour des compartiments individuels.

Vous devez disposer de l'autorisation s3:GetBucketMetadataNotification, ou être administrateur du compte, pour effectuer cette opération.

Exemple de requête

Cette requête récupère la configuration de notification des métadonnées pour le compartiment nommé `bucket`.

```
GET /bucket?x-ntap-sg-metadata-notification HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

Réponse

Le corps de la réponse inclut la configuration de notification des métadonnées du compartiment. La configuration de notification des métadonnées vous permet de déterminer comment le compartiment est configuré pour l'intégration à la recherche. Autrement dit, elle vous permet de déterminer quels objets sont indexés et vers quels points de terminaison les métadonnées de leurs objets sont envoyées.

```
<MetadataNotificationConfiguration>
    <Rule>
        <ID>Rule-1</ID>
        <Status>rule-status</Status>
        <Prefix>key-prefix</Prefix>
        <Destination>
           <Urn>arn:aws:es:_region:account-
ID_:domain/_mydomain/myindex/mytype_</Urn>
        </Destination>
    </Rule>
    <Rule>
        <ID>Rule-2</ID>
         ...
    </Rule>
     ...
</MetadataNotificationConfiguration>
```

Chaque configuration de notification de métadonnées comprend une ou plusieurs règles. Chaque règle spécifie les objets auxquels elle s'applique et la destination où StorageGRID doit envoyer les métadonnées des objets. Les destinations doivent être spécifiées à l'aide de l'URN d'un point de terminaison StorageGRID.

Nom	Description	Obligatoire
MetadataNotificationConfiguration	Balise conteneur pour les règles utilisées pour spécifier les objets et la destination des notifications de métadonnées. Contient un ou plusieurs éléments de règle.	Oui

Nom	Description	Obligatoire
Règle	Balise conteneur pour une règle qui identifie les objets dont les métadonnées doivent être ajoutées à un index spécifié. Les règles comportant des préfixes qui se chevauchent sont rejetées. Inclus dans l'élément MetadataNotificationConfiguration.	Oui
ID	Identifiant unique de la règle. Inclus dans l'élément Rule.	Non
Statut	Le statut peut être « Activé » ou « Désactivé ». Aucune action n'est entreprise pour les règles désactivées. Inclus dans l'élément Rule.	Oui
Préfixe	Les objets correspondant au préfixe sont concernés par la règle et leurs métadonnées sont envoyées à la destination spécifiée. Pour faire correspondre tous les objets, spécifiez un préfixe vide. Inclus dans l'élément Rule.	Oui
Destination	Étiquette conteneur pour la destination d'une règle. Inclus dans l'élément Rule.	Oui

Nom	Description	Obligatoire
Urn	URN de la destination où les métadonnées de l'objet sont envoyées. Doit être l'URN d'un point de terminaison StorageGRID possédant les propriétés suivantes : • `es` doit être le troisième élément. • L'URN doit se terminer par l'index et le type où les métadonnées sont stockées, sous la forme `domain-name/myindex/mytype`. Les points de terminaison sont configurés à l'aide du Gestionnaire de locataires ou de l'API de gestion des locataires. Ils se présentent sous la forme suivante : • `arn:aws:es:_region:account-ID_:domain/mydomain/myindex/mytype` • `urn:mysite:es:::mydomain/myindex/mytype` Le point de terminaison doit être configuré avant la soumission du fichier XML de configuration, sinon la configuration échouera avec une erreur 404. Urn est inclus dans l'élément Destination.	Oui

Exemple de réponse

Le code XML inclus entre les `<MetadataNotificationConfiguration></MetadataNotificationConfiguration>` balises montre comment l'intégration avec un point de terminaison d'intégration de recherche est configurée pour le compartiment. Dans cet exemple, les métadonnées de l'objet sont envoyées à un index Elasticsearch nommé `current` et de type nommé `2017` qui est hébergé dans un domaine AWS nommé `records`.

```
HTTP/1.1 200 OK
Date: Thu, 20 Jul 2017 18:24:05 GMT
Connection: KEEP-ALIVE
Server: StorageGRID/11.0.0
x-amz-request-id: 3832973499
Content-Length: 264
Content-Type: application/xml

<MetadataNotificationConfiguration>
    <Rule>
        <ID>Rule-1</ID>
        <Status>Enabled</Status>
        <Prefix>2017</Prefix>
        <Destination>
           <Urn>arn:aws:es:us-east-
1:3333333:domain/records/current/2017</Urn>
        </Destination>
    </Rule>
</MetadataNotificationConfiguration>
```

Informations connexes

"Utilisez un compte locataire"

Activez le service d'intégration de recherche dans StorageGRID avec la requête de configuration de notification des métadonnées du compartiment PUT

La requête PUT de configuration de notification des métadonnées de compartiment permet d'activer le service d'intégration de recherche pour des compartiments individuels. Le fichier XML de configuration de notification des métadonnées fourni dans le corps de la requête spécifie les objets dont les métadonnées sont envoyées à l'index de recherche de destination.

Vous devez disposer de l'autorisation s3:PutBucketMetadataNotification pour un compartiment, ou être root du compte, pour effectuer cette opération.

Demande

La requête doit inclure la configuration de notification des métadonnées dans son corps. Chaque configuration de notification des métadonnées comprend une ou plusieurs règles. Chaque règle spécifie les objets auxquels elle s'applique et la destination où StorageGRID doit envoyer les métadonnées des objets.

Il est possible de filtrer les objets en fonction du préfixe de leur nom. Par exemple, vous pouvez envoyer les métadonnées des objets avec le préfixe `/images` vers une destination, et les objets avec le préfixe `/videos` vers une autre.

Les configurations comportant des préfixes qui se chevauchent ne sont pas valides et sont rejetées lors de leur soumission. Par exemple, une configuration incluant une règle pour les objets ayant le préfixe `test` et une autre règle pour les objets ayant le préfixe `test2` ne serait pas autorisée.

Les destinations doivent être spécifiées à l'aide de l'URN d'un point de terminaison StorageGRID. Le point de terminaison doit exister lors de la soumission de la configuration de notification des métadonnées, sinon la requête échoue en tant que `400 Bad Request`. Le message d'erreur indique : `Unable to save the metadata notification (search) policy. The specified endpoint URN does not exist: URN`.

```
<MetadataNotificationConfiguration>
    <Rule>
        <ID>Rule-1</ID>
        <Status>rule-status</Status>
        <Prefix>key-prefix</Prefix>
        <Destination>
           <Urn>arn:aws:es:region:account-
ID:domain/mydomain/myindex/mytype</Urn>
        </Destination>
    </Rule>
    <Rule>
        <ID>Rule-2</ID>
         ...
    </Rule>
     ...
</MetadataNotificationConfiguration>
```

Le tableau décrit les éléments du fichier XML de configuration des notifications de métadonnées.

Nom	Description	Obligatoire
MetadataNotificationConfiguration	Balise conteneur pour les règles utilisées pour spécifier les objets et la destination des notifications de métadonnées. Contient un ou plusieurs éléments de règle.	Oui
Règle	Balise conteneur pour une règle qui identifie les objets dont les métadonnées doivent être ajoutées à un index spécifié. Les règles comportant des préfixes qui se chevauchent sont rejetées. Inclus dans l'élément MetadataNotificationConfiguration.	Oui
ID	Identifiant unique de la règle. Inclus dans l'élément Rule.	Non

Nom	Description	Obligatoire
Statut	Le statut peut être « Activé » ou « Désactivé ». Aucune action n'est entreprise pour les règles désactivées. Inclus dans l'élément Rule.	Oui
Préfixe	Les objets correspondant au préfixe sont concernés par la règle et leurs métadonnées sont envoyées à la destination spécifiée. Pour faire correspondre tous les objets, spécifiez un préfixe vide. Inclus dans l'élément Rule.	Oui
Destination	Étiquette conteneur pour la destination d'une règle. Inclus dans l'élément Rule.	Oui
Urn	URN de la destination où les métadonnées de l'objet sont envoyées. Doit être l'URN d'un point de terminaison StorageGRID possédant les propriétés suivantes : • `es` doit être le troisième élément. • L'URN doit se terminer par l'index et le type où les métadonnées sont stockées, sous la forme `domain-name/myindex/mytype`. Les points de terminaison sont configurés à l'aide du Gestionnaire de locataires ou de l'API de gestion des locataires. Ils se présentent sous la forme suivante : • `arn:aws:es:region:account-ID:domain/mydomain/myindex/mytype` • `urn:mysite:es:::mydomain/myindex/mytype` Le point de terminaison doit être configuré avant la soumission du fichier XML de configuration, sinon la configuration échouera avec une erreur 404. Urn est inclus dans l'élément Destination.	Oui

Exemples de requêtes

Cet exemple illustre l'activation de l'intégration de la recherche pour un compartiment. Dans cet exemple, les métadonnées de tous les objets sont envoyées à la même destination.

```
PUT /test1?x-ntap-sg-metadata-notification HTTP/1.1
Date: date
Authorization: authorization string
Host: host

<MetadataNotificationConfiguration>
    <Rule>
        <ID>Rule-1</ID>
        <Status>Enabled</Status>
        <Prefix></Prefix>
        <Destination>
           <Urn>urn:sgws:es:::sgws-notifications/test1/all</Urn>
        </Destination>
    </Rule>
</MetadataNotificationConfiguration>
```

Dans cet exemple, les métadonnées d'objet pour les objets qui correspondent au préfixe `/images` sont envoyées à une destination, tandis que les métadonnées d'objet pour les objets qui correspondent au préfixe `/videos` sont envoyées à une seconde destination.

```
PUT /graphics?x-ntap-sg-metadata-notification HTTP/1.1
Date: date
Authorization: authorization string
Host: host

<MetadataNotificationConfiguration>
    <Rule>
        <ID>Images-rule</ID>
        <Status>Enabled</Status>
        <Prefix>/images</Prefix>
        <Destination>
           <Urn>arn:aws:es:us-east-1:3333333:domain/es-
domain/graphics/imagetype</Urn>
        </Destination>
    </Rule>
    <Rule>
        <ID>Videos-rule</ID>
        <Status>Enabled</Status>
        <Prefix>/videos</Prefix>
        <Destination>
           <Urn>arn:aws:es:us-west-1:22222222:domain/es-
domain/graphics/videotype</Urn>
        </Destination>
    </Rule>
</MetadataNotificationConfiguration>
```

JSON généré par le service d'intégration de recherche

${post_edited_translations.segment}

Cet exemple montre un exemple du JSON qui pourrait être généré lorsqu'un objet avec la clé `SGWS/Tagging.txt` est créé dans un compartiment nommé `test`. Le compartiment `test` n'est pas versionné, la balise `versionId` est donc vide.

```
{
  "bucket": "test",
  "key": "SGWS/Tagging.txt",
  "versionId": "",
  "accountId": "86928401983529626822",
  "size": 38,
  "md5": "3d6c7634a85436eee06d43415012855",
  "region":"us-east-1",
  "metadata": {
    "age": "25"
  },
  "tags": {
    "color": "yellow"
  }
}
```

Métadonnées d'objet incluses dans les notifications de métadonnées

Le tableau répertorie tous les champs inclus dans le document JSON qui est envoyé au point de terminaison de destination lorsque l'intégration de la recherche est activée.

Le nom du document comprend le nom du compartiment, le nom de l'objet et l'identifiant de version, le cas échéant.

Type	Nom de l'article	Description
Informations sur le bucket et l'objet	bucket	Nom du compartiment
Informations sur le bucket et l'objet	${post_edited_translations.segment}	Nom de la clé de l'objet
Informations sur le bucket et l'objet	versionID	Version de l'objet, pour les objets dans des compartiments versionnés
Informations sur le bucket et l'objet	région	Région de compartiment, par exemple `us-east-1`
Métadonnées système	taille	${post_edited_translations.segment}
Métadonnées système	md5	${post_edited_translations.segment}
Métadonnées utilisateur	métadonnées *`key:value`*	Toutes les métadonnées utilisateur de l'objet, sous forme de paires clé-valeur

Type	Nom de l'article	Description
Étiquettes	balises *key:value*	Toutes les étiquettes d'objet définies pour l'objet, sous forme de paires clé-valeur

> Pour les balises et les métadonnées utilisateur, StorageGRID transmet les dates et les nombres à Elasticsearch sous forme de chaînes de caractères ou de notifications d'événements S3. Pour configurer Elasticsearch afin qu'il interprète ces chaînes comme des dates ou des nombres, suivez les instructions Elasticsearch pour le mappage dynamique des champs et le mappage des formats de date. Vous devez activer le mappage dynamique des champs sur l'index avant de configurer le service d'intégration de recherche. Après l'indexation d'un document, vous ne pouvez plus modifier les types de champs du document dans l'index.

Informations connexes

"Utilisez un compte locataire"

Récupérez l'utilisation du stockage du compte et du bucket dans StorageGRID avec la requête GET Storage Usage

La requête GET Storage Usage vous indique la quantité totale de stockage utilisée par un compte, ainsi que pour chaque compartiment associé à ce compte.

La quantité de stockage utilisée par un compte et ses buckets peut être obtenue par une requête ListBuckets modifiée avec le paramètre de requête `x-ntap-sg-usage`. L'utilisation du stockage des buckets est suivie séparément des requêtes PUT et DELETE traitées par le système. Il peut y avoir un certain délai avant que les valeurs d'utilisation ne correspondent aux valeurs attendues en fonction du traitement des requêtes, en particulier si le système est fortement sollicité.

Par défaut, StorageGRID tente de récupérer les informations d'utilisation en utilisant une cohérence globale forte. Si une cohérence globale forte ne peut pas être atteinte, StorageGRID tente de récupérer les informations d'utilisation avec une cohérence site forte.

Vous devez disposer de l'autorisation s3:ListAllMyBuckets, ou être root du compte, pour effectuer cette opération.

Exemple de requête

```
GET /?x-ntap-sg-usage HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

Exemple de réponse

Cet exemple illustre un compte contenant quatre objets et 12 octets de données répartis dans deux compartiments. Chaque compartiment contient deux objets et six octets de données.

```
HTTP/1.1 200 OK
Date: Sat, 29 Nov 2015 00:49:05 GMT
Connection: KEEP-ALIVE
Server: StorageGRID/10.2.0
x-amz-request-id: 727237123
Content-Length: 427
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<UsageResult xmlns="http://s3.storagegrid.com/doc/2015-02-01">
<CalculationTime>2014-11-19T05:30:11.000000Z</CalculationTime>
<ObjectCount>4</ObjectCount>
<DataBytes>12</DataBytes>
<Buckets>
<Bucket>
<Name>bucket1</Name>
<ObjectCount>2</ObjectCount>
<DataBytes>6</DataBytes>
</Bucket>
<Bucket>
<Name>bucket2</Name>
<ObjectCount>2</ObjectCount>
<DataBytes>6</DataBytes>
</Bucket>
</Buckets>
</UsageResult>
```

Versionnage

Chaque version d'objet stockée contribuera aux valeurs `ObjectCount` et `DataBytes` dans la réponse. Les marqueurs de suppression ne sont pas ajoutés au total `ObjectCount`.

Informations connexes

"Cohérence"

${post_edited_translations.segment}

Requêtes de compartiment obsolètes pour la conformité StorageGRID héritée

Vous pourriez avoir besoin d'utiliser l'API REST S3 de StorageGRID pour gérer les compartiments créés à l'aide de l'ancienne fonctionnalité de conformité.

Fonctionnalité de conformité obsolète

La fonctionnalité de conformité StorageGRID, disponible dans les versions précédentes de StorageGRID, est obsolète et a été remplacée par S3 Object Lock.

Si vous aviez précédemment activé le paramètre de conformité global, le paramètre de verrouillage global des objets S3 est activé dans StorageGRID 11.6. Vous ne pouvez plus créer de nouveaux compartiments avec la conformité activée ; cependant, si nécessaire, vous pouvez utiliser l'API REST S3 de StorageGRID pour gérer tout compartiment conforme existant.

- "Utilisez l'API REST S3 pour configurer S3 Object Lock"
- "Gérez les objets avec ILM"
- "Base de connaissances NetApp : Comment gérer les compartiments conformes hérités dans StorageGRID 11.5"

Demandes de conformité obsolètes :

- "Déprécié - Modifications des requêtes PUT Bucket pour la conformité"

  L'élément XML SGCompliance est obsolète. Auparavant, vous pouviez inclure cet élément personnalisé StorageGRID dans le corps de la requête XML facultatif des requêtes PUT Bucket pour créer un bucket conforme.

- "Obsolète - GET Bucket compliance"

  La requête GET Bucket compliance est obsolète. Cependant, vous pouvez continuer à utiliser cette requête pour déterminer les paramètres de conformité actuellement en vigueur pour un bucket Compliant existant.

- "Obsolète - PUT Bucket compliance"

  La requête PUT Bucket compliance est obsolète. Cependant, vous pouvez continuer à utiliser cette requête pour modifier les paramètres de conformité d'un compartiment Compliant existant. Par exemple, vous pouvez placer un compartiment existant en rétention légale ou augmenter sa période de conservation.

Élément SGCompliance obsolète dans StorageGRID

L'élément XML SGCompliance est obsolète. Auparavant, vous pouviez inclure cet élément personnalisé StorageGRID dans le corps XML facultatif des requêtes CreateBucket pour créer un compartiment conforme.

> La fonctionnalité de conformité StorageGRID, disponible dans les versions précédentes de StorageGRID, est obsolète et a été remplacée par S3 Object Lock. Pour plus de détails, consultez ce qui suit :
>
> - "Utilisez l'API REST S3 pour configurer S3 Object Lock"
> - "Base de connaissances NetApp : Comment gérer les compartiments conformes hérités dans StorageGRID 11.5"

Vous ne pouvez plus créer de nouveaux compartiments avec la conformité activée. Le message d'erreur suivant est renvoyé si vous tentez d'utiliser les modifications de la requête CreateBucket pour la conformité afin de créer un nouveau compartiment conforme :

```
The Compliance feature is deprecated.
Contact your StorageGRID administrator if you need to create new Compliant
buckets.
```

Requête GET Bucket compliance obsolète dans StorageGRID

La requête GET Bucket compliance est obsolète. Cependant, vous pouvez continuer à utiliser cette requête pour déterminer les paramètres de conformité actuellement en vigueur pour un bucket Compliant existant.

> La fonctionnalité de conformité StorageGRID, disponible dans les versions précédentes de StorageGRID, est obsolète et a été remplacée par S3 Object Lock. Pour plus de détails, consultez ce qui suit :
>
> - "Utilisez l'API REST S3 pour configurer S3 Object Lock"
> - "Base de connaissances NetApp : Comment gérer les compartiments conformes hérités dans StorageGRID 11.5"

Vous devez disposer de l'autorisation s3:GetBucketCompliance, ou être root du compte, pour effectuer cette opération.

Exemple de requête

Cette requête d'exemple vous permet de déterminer les paramètres de conformité du compartiment nommé `mybucket`.

```
GET /mybucket/?x-ntap-sg-compliance HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

Exemple de réponse

Dans la réponse XML, `<SGCompliance>` répertorie les paramètres de conformité en vigueur pour le compartiment. Cet exemple de réponse montre les paramètres de conformité d'un compartiment dans lequel chaque objet sera conservé pendant un an (525 600 minutes), à partir de son ingestion dans la grille. Il n'y a actuellement aucune restriction légale sur ce compartiment. Chaque objet sera automatiquement supprimé après un an.

```
HTTP/1.1 200 OK
Date: date
Connection: connection
Server: StorageGRID/11.1.0
x-amz-request-id: request ID
Content-Length: length
Content-Type: application/xml

<SGCompliance>
  <RetentionPeriodMinutes>525600</RetentionPeriodMinutes>
  <LegalHold>false</LegalHold>
  <AutoDelete>true</AutoDelete>
</SGCompliance>
```

Nom	Description
RetentionPeriodMinutes	La durée de la période de conservation des objets ajoutés à ce compartiment, en minutes. La période de conservation commence lorsque l'objet est ingérée dans la grille.
LegalHold	• Vrai : Ce compartiment fait actuellement l'objet d'une rétention légale. Les objets dans ce compartiment ne peuvent pas être supprimés tant que la rétention légale n'est pas levée, même si leur période de conservation est expirée. • Faux : ce compartiment n'est actuellement soumis à aucune restriction légale. Les objets de ce compartiment peuvent être supprimés à l'expiration de leur période de conservation.
AutoDelete	• Vrai : les objets dans ce compartiment seront automatiquement supprimés à l'expiration de leur période de conservation, sauf si le compartiment fait l'objet d'une rétention légale. • Faux : les objets de ce compartiment ne seront pas supprimés automatiquement à l'expiration de la période de rétention. Vous devez supprimer ces objets manuellement si vous devez les supprimer.

Réponses d'erreur

Si le compartiment n'a pas été créé pour être conforme, le code d'état HTTP de la réponse est `404 Not Found`, avec un code d'erreur S3 de `XNoSuchBucketCompliance`.

Requête PUT Bucket compliance obsolète dans StorageGRID

La requête PUT Bucket compliance est obsolète. Cependant, vous pouvez continuer à utiliser cette requête pour modifier les paramètres de conformité d'un compartiment Compliant existant. Par exemple, vous pouvez placer un compartiment existant en rétention légale ou augmenter sa période de conservation.

La fonctionnalité de conformité StorageGRID, disponible dans les versions précédentes de StorageGRID, est obsolète et a été remplacée par S3 Object Lock. Pour plus de détails, consultez ce qui suit :

- "Utilisez l'API REST S3 pour configurer S3 Object Lock"
- "Base de connaissances NetApp : Comment gérer les compartiments conformes hérités dans StorageGRID 11.5"

Vous devez disposer de l'autorisation s3:PutBucketCompliance, ou être root du compte, pour effectuer cette opération.

Vous devez spécifier une valeur pour chaque champ des paramètres de conformité lors de l'émission d'une requête de conformité PUT Bucket.

Exemple de requête

Cette requête d'exemple modifie les paramètres de conformité du compartiment nommé `mybucket`. Dans cet exemple, les objets dans `mybucket` seront désormais conservés pendant deux ans (1 051 200 minutes) au lieu d'un an, à compter de leur ingestion dans la grille. Ce compartiment n'est soumis à aucune restriction légale. Chaque objet sera automatiquement supprimé après deux ans.

```
PUT /mybucket/?x-ntap-sg-compliance HTTP/1.1
Date: date
Authorization: authorization name
Host: host
Content-Length: 152

<SGCompliance>
  <RetentionPeriodMinutes>1051200</RetentionPeriodMinutes>
  <LegalHold>false</LegalHold>
  <AutoDelete>true</AutoDelete>
</SGCompliance>
```

Nom	Description
RetentionPeriodMinutes	La durée de la période de conservation des objets ajoutés à ce compartiment, en minutes. La période de conservation commence lorsque l'objet est ingérée dans la grille. **Important** Lorsque vous définissez une nouvelle valeur pour RetentionPeriodMinutes, vous devez indiquer une valeur supérieure ou égale à la période de rétention actuelle du compartiment. Une fois la période de rétention du compartiment définie, vous ne pouvez plus diminuer cette valeur ; vous pouvez seulement l'augmenter.

Nom	Description
LegalHold	• Vrai : Ce compartiment fait actuellement l'objet d'une rétention légale. Les objets dans ce compartiment ne peuvent pas être supprimés tant que la rétention légale n'est pas levée, même si leur période de conservation est expirée. • Faux : ce compartiment n'est actuellement soumis à aucune restriction légale. Les objets de ce compartiment peuvent être supprimés à l'expiration de leur période de conservation.
AutoDelete	• Vrai : les objets dans ce compartiment seront automatiquement supprimés à l'expiration de leur période de conservation, sauf si le compartiment fait l'objet d'une rétention légale. • Faux : les objets de ce compartiment ne seront pas supprimés automatiquement à l'expiration de la période de rétention. Vous devez supprimer ces objets manuellement si vous devez les supprimer.

Cohérence des paramètres de conformité

Lorsque vous mettez à jour les paramètres de conformité d'un compartiment S3 à l'aide d'une requête PUT Bucket compliance, StorageGRID tente de mettre à jour les métadonnées du compartiment sur l'ensemble de la grille. Par défaut, StorageGRID utilise la cohérence **Strong-global** pour garantir que tous les sites de centres de données et tous les nœuds de stockage contenant des métadonnées de compartiment bénéficient d'une cohérence en lecture après écriture pour les paramètres de conformité modifiés.

Si StorageGRID ne peut pas atteindre la cohérence **Strong-global** parce qu'un site de centre de données ou plusieurs Storage Nodes d'un site sont indisponibles, le code d'état HTTP de la réponse est `503 Service Unavailable.`

Si vous recevez cette réponse, vous devez contacter l'administrateur du grid pour vous assurer que les services de stockage requis sont disponibles dans les plus brefs délais. Si l'administrateur du grid ne parvient pas à rendre disponibles suffisamment de Storage Nodes sur chaque site, le support technique pourrait vous demander de relancer la requête ayant échoué en forçant la cohérence **Strong-site**.

> ! N'imposez jamais la cohérence **Strong-site** pour la conformité du compartiment PUT, sauf si le support technique vous l'a demandé et si vous comprenez les conséquences potentielles de l'utilisation de ce niveau.

Lorsque la cohérence est réduite à **Strong-site**, StorageGRID garantit que les paramètres de conformité mis à jour offriront une cohérence de lecture après écriture uniquement pour les requêtes client au sein d'un site. Cela signifie que le système StorageGRID peut temporairement avoir plusieurs paramètres incohérents pour ce bucket, jusqu'à ce que tous les sites et nœuds de stockage soient disponibles. Les paramètres incohérents peuvent entraîner un comportement inattendu et indésirable. Par exemple, si vous placez un bucket sous conservation légale et que vous forcez une cohérence inférieure, les paramètres de conformité précédents du bucket (c'est-à-dire conservation légale désactivée) peuvent continuer à être en vigueur sur certains sites de centres de données. En conséquence, des objets que vous pensez être sous conservation légale peuvent être supprimés à l'expiration de leur période de rétention, soit par l'utilisateur, soit par AutoDelete, si cette option est activée.

Pour forcer l'utilisation de la cohérence **Strong-site**, réémettez la requête de conformité PUT Bucket et incluez l' `Consistency-Control`en-tête de requête HTTP, comme suit :

```
PUT /mybucket/?x-ntap-sg-compliance HTTP/1.1
Consistency-Control: strong-site
```

Réponses d'erreur

- Si le compartiment n'a pas été créé pour être conforme, le code d'état HTTP de la réponse est `404 Not Found`.
- Si `RetentionPeriodMinutes` dans la requête est inférieur à la période de rétention actuelle du compartiment, le code d'état HTTP est `400 Bad Request`.

Informations connexes

"Déprécié : Modifications des requêtes PUT Bucket pour conformité"

${post_edited_translations.segment}

Comment StorageGRID utilise les politiques AWS pour contrôler l'accès aux compartiments et objets S3

StorageGRID utilise le langage de stratégie Amazon Web Services (AWS) pour permettre aux locataires S3 de contrôler l'accès aux compartiments et aux objets au sein de ces compartiments. Le système StorageGRID implémente un sous-ensemble du langage de stratégie de l'API REST S3. Les stratégies d'accès pour l'API S3 sont écrites en JSON.

Aperçu de la politique d'accès

StorageGRID prend en charge trois types de politiques d'accès :

- Les **stratégies de compartiment**, qui sont gérées à l'aide des opérations d'API S3 GetBucketPolicy, PutBucketPolicy et DeleteBucketPolicy, ou du Gestionnaire de locataires ou de l'API de gestion des locataires. Les stratégies de compartiment sont rattachées aux compartiments. Elles sont donc configurées pour contrôler l'accès au compartiment et aux objets qu'il contient par les utilisateurs du compte propriétaire du compartiment ou d'autres comptes. Une stratégie de compartiment s'applique à un seul compartiment et potentiellement à plusieurs groupes.
- **Politiques de groupe**, qui sont configurées à l'aide du Tenant Manager ou de l'API Tenant Management. Les politiques de groupe sont associées à un groupe du compte, elles sont donc configurées pour autoriser ce groupe à accéder à des ressources spécifiques détenues par ce compte. Une politique de groupe s'applique à un seul groupe et éventuellement à plusieurs compartiments.
- **Politiques de session**, qui sont incluses dans une demande AssumeRole. Les politiques de session ne s'appliquent qu'à la session donnée, définissant plus précisément les autorisations dont l'utilisateur dispose, en plus de celles accordées par la politique de groupe et de compartiment.

${post_edited_translations.segment}

Les politiques de compartiment et de groupe StorageGRID suivent une grammaire spécifique définie par Amazon. Chaque politique contient un tableau d'instructions de stratégie, et chaque instruction contient les éléments suivants :

- ${post_edited_translations.segment}

- ${post_edited_translations.segment}
- Principal/NotPrincipal
- Ressource/NotResource
- Action/NotAction
- Condition (facultative)

Les instructions de stratégie sont construites à l'aide de cette structure pour spécifier les autorisations : Accorder <Effect> pour autoriser/refuser à <Principal> d'effectuer <Action> sur <Resource> lorsque <Condition> s'applique.

${post_edited_translations.segment}

Élément	Description
Sid	L'élément Sid est facultatif. Le Sid est uniquement destiné à servir de description pour l'utilisateur. Il est stocké mais n'est pas interprété par le système StorageGRID.
${post_edited_translations.segment}	Utilisez l'élément Effect pour déterminer si les opérations spécifiées sont autorisées ou refusées. Vous devez identifier les opérations que vous autorisez (ou refusez) sur les compartiments ou les objets à l'aide des mots-clés de l'élément Action pris en charge.
Principal/NotPrincipal	Vous pouvez autoriser des utilisateurs, des groupes et des comptes à accéder à des ressources spécifiques et à effectuer des actions spécifiques. Si aucune signature S3 n'est incluse dans la requête, l'accès anonyme est autorisé en spécifiant le caractère générique (*) comme principal. Par défaut, seul le compte racine a accès aux ressources dont il est propriétaire. Il vous suffit de spécifier l'élément Principal dans une stratégie de compartiment. Pour les stratégies de groupe, le groupe auquel la stratégie est rattachée constitue l'élément Principal implicite.
Ressource/NotResource	L'élément Resource identifie les compartiments et les objets. Vous pouvez autoriser ou refuser des autorisations aux compartiments et aux objets en utilisant l'ARN (Amazon Resource Name) pour identifier la ressource.
Action/NotAction	Les éléments Action et Effect sont les deux composants des autorisations. Lorsqu'un groupe demande une ressource, l'accès à celle-ci lui est accordé ou refusé. L'accès est refusé à moins que vous n'attribuiez spécifiquement des autorisations, mais vous pouvez utiliser un refus explicite pour remplacer une autorisation accordée par une autre règle.
Condition	L'élément Condition est facultatif. Les conditions permettent de créer des expressions pour déterminer quand une règle doit être appliquée.

Dans l'élément Action, vous pouvez utiliser le caractère générique (*) pour spécifier toutes les opérations ou

un sous-ensemble d'opérations. Par exemple, cette Action correspond à des permissions telles que s3:GetObject, s3:PutObject et s3:DeleteObject.

```
s3:*Object
```

Dans l'élément Resource, vous pouvez utiliser les caractères génériques (**) et (?). L'astérisque (**) correspond à zéro ou plusieurs caractères, tandis que le point d'interrogation (?) correspond à n'importe quel caractère.

Dans l'élément Principal, les caractères génériques ne sont pas pris en charge, sauf pour définir l'accès anonyme, qui accorde l'autorisation à tous. Par exemple, vous définissez le caractère générique (*) comme valeur de Principal.

```
"Principal":"*"
```

```
"Principal":{"AWS":"*"}
```

Dans l'exemple suivant, l'instruction utilise les éléments Effect, Principal, Action et Resource. Cet exemple présente une instruction de stratégie de compartiment complète qui utilise l'effet « Allow » pour accorder aux Principals, le groupe admin `federated-group/admin` et le groupe finance `federated-group/finance`, les autorisations d'effectuer l'Action `s3:ListBucket` sur le compartiment nommé `mybucket` et l'Action `s3:GetObject` sur tous les objets à l'intérieur de ce compartiment.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::27233906934684427525:federated-group/admin",
          "arn:aws:iam::27233906934684427525:federated-group/finance"
        ]
      },
      "Action": [
        "s3:ListBucket",
        "s3:GetObject"
      ],
      "Resource": [
        "arn:aws:s3:::mybucket",
        "arn:aws:s3:::mybucket/*"
      ]
    }
  ]
}
```

${post_edited_translations.segment}

${post_edited_translations.segment}

Par défaut, toutes les mises à jour que vous apportez aux règles de groupe sont cohérentes à terme. Lorsqu'une règle de groupe devient cohérente, l'application des modifications peut prendre 15 minutes supplémentaires en raison de la mise en cache des règles. Par défaut, toutes les mises à jour que vous apportez aux règles de compartiment sont fortement cohérentes.

Vous pouvez, si nécessaire, modifier les garanties de cohérence pour les mises à jour des règles d'accès du compartiment. Par exemple, vous pouvez souhaiter qu'une modification d'une règle d'accès de compartiment soit disponible lors d'une panne de site.

Dans ce cas, vous pouvez soit définir l'en-tête `Consistency-Control` dans la requête PutBucketPolicy, soit utiliser la requête de cohérence PUT Bucket. Lorsqu'une politique de compartiment devient cohérente, la prise d'effet des modifications peut prendre 8 secondes supplémentaires en raison de la mise en cache des politiques.

> Si vous modifiez la cohérence pour contourner un problème temporaire, veillez à rétablir la valeur d'origine du paramètre au niveau du compartiment une fois la modification terminée. Sinon, toutes les requêtes ultérieures au compartiment utiliseront le paramètre modifié.

${post_edited_translations.segment}

Une règle de session est une règle d'accès qui restreint temporairement les autorisations disponibles lors d'une session spécifique, par exemple lorsqu'un utilisateur assume un groupe. Une règle de session ne peut autoriser qu'un sous-ensemble d'autorisations et ne peut pas accorder d'autorisations supplémentaires. Le groupe lui-même peut disposer d'autorisations plus larges.

${post_edited_translations.segment}

${post_edited_translations.segment}

- ${post_edited_translations.segment}

```
arn:aws:s3:::bucket-name
arn:aws:s3:::bucket-name/object_key
```

- Utilisez cette syntaxe pour spécifier l'ARN de la ressource d'identité (utilisateurs et groupes) :

```
arn:aws:iam::account_id:root
arn:aws:iam::account_id:user/user_name
arn:aws:iam::account_id:group/group_name
arn:aws:iam::account_id:federated-user/user_name
arn:aws:iam::account_id:federated-group/group_name
```

${post_edited_translations.segment}

- ${post_edited_translations.segment}

- Les caractères internationaux, qui peuvent être spécifiés dans la clé d'objet, doivent être encodés à l'aide de JSON UTF-8 ou de séquences d'échappement JSON \u. L'encodage en pourcentage n'est pas pris en charge.

  "${post_edited_translations.segment}"

  Le corps de la requête HTTP pour l'opération PutBucketPolicy doit être encodé avec charset=UTF-8.

Spécifiez les ressources dans une règle

${post_edited_translations.segment}

- Chaque énoncé de politique requiert un élément Resource. Dans une politique, les ressources sont désignées par l'élément `Resource`, ou alternativement, `NotResource` pour l'exclusion.
- Vous spécifiez les ressources avec un ARN de ressource S3. Par exemple :

```
"Resource": "arn:aws:s3:::mybucket/*"
```

- Vous pouvez également utiliser des variables de stratégie dans la clé de l'objet. Par exemple :

```
"Resource": "arn:aws:s3:::mybucket/home/${aws:username}/*"
```

- La valeur de la ressource peut spécifier un compartiment qui n'existe pas encore lors de la création d'une stratégie de groupe.

${post_edited_translations.segment}

${post_edited_translations.segment}

- Chaque déclaration de stratégie dans une stratégie de compartiment doit inclure un élément Principal. Les déclarations de stratégie dans une stratégie de groupe n'ont pas besoin de l'élément Principal car le groupe est considéré comme le principal.
- Dans une politique, les principaux sont désignés par l'élément "Principal" ou, alternativement, "NotPrincipal" pour l'exclusion.
- Les identités basées sur un compte doivent être spécifiées à l'aide d'un ID ou d'un ARN :

```
"Principal": { "AWS": "account_id"}
"Principal": { "AWS": "identity_arn" }
```

- ${post_edited_translations.segment}

```
 "Principal": { "AWS": "27233906934684427525" }
```

- Vous pouvez spécifier uniquement le compte racine :

```
"Principal": { "AWS": "arn:aws:iam::27233906934684427525:root" }
```

- ${post_edited_translations.segment}

```
"Principal": { "AWS": "arn:aws:iam::27233906934684427525:federated-
user/Alex" }
```

- Vous pouvez spécifier un groupe fédéré particulier (« Managers ») :

```
"Principal": { "AWS": "arn:aws:iam::27233906934684427525:federated-
group/Managers"  }
```

- ${post_edited_translations.segment}

```
"Principal": "*"
```

- Pour éviter toute ambiguïté, vous pouvez utiliser l'UUID de l'utilisateur au lieu du nom d'utilisateur :

```
arn:aws:iam::27233906934684427525:user-uuid/de305d54-75b4-431b-adb2-
eb6b9e546013
```

  Par exemple, supposons qu'Alex quitte l'organisation et que le nom d'utilisateur `Alex` soit supprimé. Si un nouvel Alex rejoint l'organisation et qu'on lui attribue le même `Alex` nom d'utilisateur, le nouvel utilisateur pourrait hériter involontairement des autorisations accordées à l'utilisateur d'origine.

- La valeur principale peut spécifier un nom de groupe/utilisateur qui n'existe pas encore lors de la création d'une règle de compartiment.

${post_edited_translations.segment}

Dans une stratégie, l'élément Action est utilisé pour autoriser ou refuser des autorisations sur une ressource. Il existe un ensemble d'autorisations que vous pouvez spécifier dans une stratégie, qui sont désignées par l'élément "Action" ou, alternativement, "NotAction" pour l'exclusion. Chacun de ces éléments correspond à des opérations d'API REST S3 spécifiques.

Les tableaux répertorient les autorisations qui s'appliquent aux compartiments et les autorisations qui s'appliquent aux objets.

> Amazon S3 utilise désormais l'autorisation s3:PutReplicationConfiguration pour les actions PutBucketReplication et DeleteBucketReplication. StorageGRID utilise des autorisations distinctes pour chaque action, ce qui correspond à la spécification Amazon S3 d'origine.

> ${post_edited_translations.segment}

Autorisations applicables aux compartiments

Autorisations	${post_edited_translations.segment}	${post_edited_translations.segment}
s3:CreateBucket	CreateBucket	Oui. **Remarque** : À utiliser uniquement dans la stratégie de groupe.
s3:DeleteBucket	DeleteBucket	
s3:DeleteBucketMetadataNotification	SUPPRIMER la configuration de notification des métadonnées du compartiment	Oui
s3:DeleteBucketPolicy	DeleteBucketPolicy	
s3:DeleteReplicationConfiguration	DeleteBucketReplication	Oui, des autorisations distinctes pour PUT et DELETE
s3:GetBucketAcl	GetBucketAcl	
s3:GetBucketCompliance	GET Bucket compliance (obsolète)	Oui
s3:GetBucketConsistency	Cohérence du bucket GET	Oui
${post_edited_translations.segment}	GetBucketCors	
s3:GetEncryptionConfiguration	GetBucketEncryption	
s3:GetBucketLastAccessTime	Obtenir le temps d’accès du compartiment	Oui
s3:GetBucketLocation	GetBucketLocation	
s3:GetBucketMetadataNotification	Obtenir la configuration de notification des métadonnées du bucket	Oui
s3:GetBucketNotification	GetBucketNotificationConfiguration	
s3:GetBucketObjectLockConfiguration	GetObjectLockConfiguration	
s3:GetBucketPolicy	GetBucketPolicy	

Autorisations	${post_edited_translations.segment}	${post_edited_translations.segment}
s3:GetBucketTagging	GetBucketTagging	
s3:GetBucketVersioning	GetBucketVersioning	
s3:GetLifecycleConfiguration	GetBucketLifecycleConfiguration	
s3:GetReplicationConfiguration	GetBucketReplication	
s3:ListAllMyBuckets	• ListBuckets • Obtenir l'utilisation du stockage	${post_edited_translations.segment} **Remarque** : À utiliser uniquement dans la stratégie de groupe.
s3:ListBucket	• ListObjects • HeadBucket • RestoreObject	
s3:ListBucketMultipartUploads	• ListMultipartUploads • RestoreObject	
s3:ListBucketVersions	${post_edited_translations.segment}	
s3:PutBucketCompliance	${post_edited_translations.segment}	Oui
s3:PutBucketConsistency	Cohérence PUT Bucket	Oui
s3:PutBucketCORS	• DeleteBucketCors† • PutBucketCors	
s3:PutEncryptionConfiguration	• DeleteBucketEncryption • PutBucketEncryption	
s3:PutBucketLastAccessTime	PUT temps d'accès du dernier accès au compartiment	Oui
s3:PutBucketMetadataNotification	Configuration de notification des métadonnées du compartiment PUT	Oui
s3:PutBucketNotification	PutBucketNotificationConfiguration	

Autorisations	${post_edited_translations.segment}	${post_edited_translations.segment}
s3:PutBucketObjectLockConfiguration	• CreateBucket avec l'en-tête de requête `x-amz-bucket-object-lock-enabled: true` (requiert également l'autorisation s3:CreateBucket) • PutObjectLockConfiguration	
s3:PutBucketPolicy	PutBucketPolicy	
s3:PutBucketTagging	• DeleteBucketTagging† • PutBucketTagging	
s3:PutBucketVersioning	PutBucketVersioning	
s3:PutLifecycleConfiguration	• DeleteBucketLifecycle† • PutBucketLifecycleConfiguration	
s3:PutReplicationConfiguration	PutBucketReplication	Oui, des autorisations distinctes pour PUT et DELETE

${post_edited_translations.segment}

Autorisations	${post_edited_translations.segment}	${post_edited_translations.segment}
s3:AbortMultipartUpload	• AbortMultipartUpload • RestoreObject	
s3:BypassGovernanceRetention	• DeleteObject • DeleteObjects • PutObjectRetention	
s3:DeleteObject	• DeleteObject • DeleteObjects • RestoreObject	
s3:DeleteObjectTagging	DeleteObjectTagging	
s3:DeleteObjectVersionTagging	DeleteObjectTagging (une version spécifique de l'objet)	

Autorisations	${post_edited_translations.segment}	${post_edited_translations.segment}
s3:DeleteObjectVersion	DeleteObject (une version spécifique de l'objet)	
s3:GetObject	• GetObject • HeadObject • RestoreObject • SelectObjectContent	
s3:GetObjectAcl	GetObjectAcl	
s3:GetObjectLegalHold	GetObjectLegalHold	
s3:GetObjectRetention	GetObjectRetention	
s3:GetObjectTagging	GetObjectTagging	
s3:GetObjectVersionTagging	GetObjectTagging (une version spécifique de l'objet)	
s3:GetObjectVersion	GetObject (une version spécifique de l'objet)	
s3:ListMultipartUploadParts	ListParts, RestoreObject	
s3:PutObject	• PutObject • CopyObject • RestoreObject • CreateMultipartUpload • CompleteMultipartUpload • UploadPart • UploadPartCopy	
s3:PutObjectLegalHold	PutObjectLegalHold	
s3:PutObjectRetention	PutObjectRetention	
s3:PutObjectTagging	PutObjectTagging	
s3:PutObjectVersionTagging	PutObjectTagging (une version spécifique de l'objet)	

Autorisations	${post_edited_translations.segment}	${post_edited_translations.segment}
s3:PutOverwriteObject	• PutObject • CopyObject • PutObjectTagging • DeleteObjectTagging • CompleteMultipartUpload	Oui
s3:RestoreObject	RestoreObject	

Utilisez l'autorisation PutOverwriteObject

L'autorisation s3:PutOverwriteObject est une autorisation StorageGRID personnalisée qui s'applique aux opérations de création ou de mise à jour d'objets. Le paramétrage de cette autorisation détermine si le client peut écraser les données d'un objet, les métadonnées définies par l'utilisateur ou le balisage d'objet S3.

${post_edited_translations.segment}

- **Autoriser** : le client peut écraser un objet. Il s'agit du paramètre par défaut.
- **Deny** : le client ne peut pas écraser un objet. Lorsqu'elle est définie sur Deny, l'autorisation PutOverwriteObject fonctionne comme suit :
  - Si un objet existant est trouvé au même chemin :
    - Les données de l'objet, les métadonnées définies par l'utilisateur ou le balisage de l'objet S3 ne peuvent pas être écrasés.
    - Toute opération d'ingestion en cours est annulée et une erreur est renvoyée.
    - Si le versionnement S3 est activé, le paramètre Deny empêche les opérations PutObjectTagging ou DeleteObjectTagging de modifier le TagSet pour un objet et ses versions non actuelles.
  - ${post_edited_translations.segment}
- Lorsque cette autorisation est absente, l'effet est le même que si l'option Allow était activée.

> Si la politique S3 actuelle autorise l'écrasement et que l'autorisation PutOverwriteObject est définie sur Deny, le client ne peut pas écraser les données d'un objet, ses métadonnées définies par l'utilisateur ou son balisage d'objet. De plus, si la case à cocher **Empêcher la modification par le client** est sélectionnée (**Configuration** > **Paramètres de sécurité** > **Réseau et objets**), ce paramètre remplace le paramètre de l'autorisation PutOverwriteObject.

${post_edited_translations.segment}

Les conditions définissent quand une politique sera en vigueur. Les conditions se composent d'opérateurs et de paires clé-valeur.

Les conditions utilisent des paires clé-valeur pour l'évaluation. Un élément Condition peut contenir plusieurs conditions, et chaque condition peut contenir plusieurs paires clé-valeur. Le bloc de condition utilise le format suivant :

```
Condition: {
     condition_type: {
          condition_key: condition_values
```

Dans l'exemple suivant, la condition IpAddress utilise la clé de condition SourceIp.

```
"Condition": {
    "IpAddress": {
      "aws:SourceIp": "54.240.143.0/24"
        ...
},
        ...
```

${post_edited_translations.segment}

${post_edited_translations.segment}

- Chaîne
- Numérique
- ${post_edited_translations.segment}
- Adresse IP
- ${post_edited_translations.segment}

${post_edited_translations.segment}	Description
StringEquals	${post_edited_translations.segment}
StringNotEquals	Compare une clé à une valeur de chaîne en fonction d'une correspondance négative (sensible à la casse).
StringEqualsIgnoreCase	Compare une clé à une valeur de chaîne en fonction d'une correspondance exacte (ignore la casse).
StringNotEqualsIgnoreCase	${post_edited_translations.segment}
StringLike	Compare une clé à une valeur de chaîne de caractères en fonction d'une correspondance exacte (sensible à la casse). Peut inclure les caractères génériques * et ?.
StringNotLike	Compare une clé à une valeur de chaîne sur la base d'une correspondance inversée (sensible à la casse). Peut inclure les caractères génériques * et ?.

${post_edited_translations.segment}	Description
NumericEquals	Compare une clé à une valeur numérique en fonction d'une correspondance exacte.
NumericNotEquals	${post_edited_translations.segment}
NumericGreaterThan	${post_edited_translations.segment}
NumericGreaterThanEquals	Compare une clé à une valeur numérique en fonction d'une correspondance « supérieur ou égal à ».
NumericLessThan	Compare une clé à une valeur numérique en fonction de la correspondance « inférieur à ».
NumericLessThanEquals	${post_edited_translations.segment}
${post_edited_translations.segment}	${post_edited_translations.segment}
IpAddress	${post_edited_translations.segment}
NotIpAddress	Compare une clé à une adresse IP ou à une plage d'adresses IP en se basant sur une correspondance négative.
${post_edited_translations.segment}	Vérifie si une clé de condition est présente dans le contexte de la requête actuelle.
IfExists	Ajouté à tout opérateur de condition, à l'exception de la condition Null, pour vérifier l'absence de cette clé de condition. Renvoie TRUE si la clé de condition n'est pas présente.

Clés de condition prises en charge

Clés de condition	Actions	Description
aws:SourceIp	opérateurs IP	Sera comparée à l'adresse IP à partir de laquelle la requête a été envoyée. Peut être utilisé pour les opérations sur les compartiments ou les objets. ${post_edited_translations.segment} **Remarque** : Si un équilibreur de charge tiers non transparent est utilisé, la comparaison portera sur l'adresse IP de cet équilibreur. Tout en-tête `X-Forwarded-For` sera ignoré, car sa validité ne peut être vérifiée.

Clés de condition	Actions	Description
${post_edited_translations.segment}	${post_edited_translations.segment}	Sera comparé au nom d'utilisateur de l'expéditeur à partir duquel la requête a été envoyée. Peut être utilisé pour les opérations sur les compartiments ou les objets.
s3: délimiteur	s3:ListBucket et Autorisations s3 : ListBucketVersions	Sera comparé au paramètre délimiteur spécifié dans une requête ListObjects ou ListObjectVersions.
s3:ExistingObjectTag/<tag-key>	s3:DeleteObjectTagging s3:DeleteObjectVersionTagging s3:GetObject s3:GetObjectAcl 3:GetObjectTagging s3:GetObjectVersion s3:GetObjectVersionAcl s3:GetObjectVersionTagging s3:PutObjectAcl s3:PutObjectTagging s3:PutObjectVersionAcl s3:PutObjectVersionTagging	Nécessitera que l'objet existant possède la clé et la valeur de balise spécifiques.
s3:max-keys	s3:ListBucket et Autorisations s3 : ListBucketVersions	Sera comparé au paramètre max-keys spécifié dans une requête ListObjects ou ListObjectVersions.
s3:object-lock-mode	s3:PutObject	Compare avec le `object-lock-mode` développé à partir de l'en-tête de requête dans la requête PutObject, CopyObject et CreateMultipartUpload.
s3:object-lock-mode	s3:PutObjectRetention	Comparé à l' `object-lock-mode`élément développé à partir du corps XML dans la requête PutObjectRetention.

Clés de condition	Actions	Description
${post_edited_translations.segment}	s3:PutObject	Compare à la date de conservation obligatoire spécifiée dans l'en-tête de requête `x-amz-object-lock-retain-until-date` ou calculée à partir de la période de rétention par défaut du compartiment pour s'assurer que ces valeurs se situent dans la plage autorisée pour les requêtes suivantes : • PutObject • CopyObject • CreateMultipartUpload
${post_edited_translations.segment}	s3:PutObjectRetention	Compare à la date de conservation limite spécifiée dans la requête PutObjectRetention pour s'assurer qu'elle s'inscrit dans la plage autorisée.
${post_edited_translations.segment}	s3:ListBucket et Autorisations s3 : ListBucketVersions	Sera comparé au paramètre de préfixe spécifié dans une requête ListObjects ou ListObjectVersions.
s3:RequestObjectTag/<tag-key>	s3:PutObject s3:PutObjectTagging s3:PutObjectVersionTagging	Une clé et une valeur d'étiquette spécifiques seront requises lorsque la requête d'objet inclut un balisage.
s3:x-amz-server-side-encryption-customer-algorithm	s3:PutObject	Compare à la `sse-customer-algorithm` ou à la `copy-source-sse-customer-algorithm` valeur développée à partir de l'en-tête de la requête dans la demande PutObject, CopyObject, CreateMultipartUpload, UploadPart, UploadPartCopy et CompleteMultipartUpload.

Spécifiez les variables dans une politique

Vous pouvez utiliser des variables dans les politiques pour renseigner les informations de politique lorsqu'elles sont disponibles. Vous pouvez utiliser des variables de politique dans l' `Resource`élément et dans les comparaisons de chaînes dans l' `Condition`élément.

Dans cet exemple, la variable `${aws:username}` fait partie de l'élément Resource :

```
"Resource": "arn:aws:s3:::bucket-name/home/${aws:username}/*"
```

Dans cet exemple, la variable `${aws:username}` fait partie de la valeur de la condition dans le bloc de condition :

```
"Condition": {
    "StringLike": {
      "s3:prefix": "${aws:username}/*"
        ...
},
        ...
```

Variable	Description
`${aws:SourceIp}`	Utilise la clé SourceIp comme variable fournie.
`${aws:username}`	${post_edited_translations.segment}
`${s3:prefix}`	Utilise la clé de préfixe spécifique au service comme variable fournie.
`${s3:max-keys}`	Utilise la clé max-keys spécifique au service comme variable fournie.
`${*}`	Caractère spécial. Utilise le caractère comme un caractère * littéral.
`${?}`	Caractère spécial. Utilise le caractère en tant que caractère ? littéral.
`${$}`	Caractère spécial. Utilise le caractère comme un symbole $ littéral.

${post_edited_translations.segment}

Il arrive qu'une politique accorde des autorisations dangereuses pour la sécurité ou la continuité des opérations, comme le blocage de l'utilisateur racine du compte. L'implémentation de l'API REST S3 de StorageGRID est moins restrictive lors de la validation des politiques qu'Amazon, mais tout aussi stricte lors de leur évaluation.

Description de la politique	Type de stratégie	Comportement d'Amazon	Comportement de StorageGRID
Refusez-vous toute autorisation sur le compte racine	Bucket	Valide et appliquée, mais le compte utilisateur root conserve l'autorisation pour toutes les opérations de stratégie de compartiment S3	Même
Refuser à soi-même toutes les autorisations à l'utilisateur/groupe	Groupe	${post_edited_translations.segment}	Même
${post_edited_translations.segment}	Bucket	Principal invalide	${post_edited_translations.segment}

Description de la politique	Type de stratégie	Comportement d'Amazon	Comportement de StorageGRID
${post_edited_translations.segment}	Bucket	${post_edited_translations.segment}	Même
Autoriser à tous les autorisations pour toutes les actions	Bucket	Valide, mais les autorisations pour toutes les opérations de stratégie de compartiment S3 renvoient une erreur 405 Method Not Allowed pour le compte racine et les utilisateurs du compte externe.	Même
Refuser à tous l'autorisation d'effectuer toutes les actions	Bucket	Valide et appliquée, mais le compte utilisateur root conserve l'autorisation pour toutes les opérations de stratégie de compartiment S3	Même
${post_edited_translations.segment}	Bucket	Principal invalide	${post_edited_translations.segment}
La ressource est un compartiment S3 inexistant	Groupe	${post_edited_translations.segment}	Même
Le principal est un groupe local	Bucket	Principal invalide	${post_edited_translations.segment}
Cette politique accorde à un compte non propriétaire (y compris les comptes anonymes) l'autorisation de placer des objets.	Bucket	Valide. Les objets appartiennent au compte créateur, et la politique de compartiment ne s'applique pas. Le compte créateur doit accorder des autorisations d'accès pour l'objet à l'aide des ACL d'objet.	Valide. Les objets appartiennent au compte du propriétaire du compartiment. La politique de compartiment s'applique.

${post_edited_translations.segment}

Vous pouvez créer des compartiments WORM (write-once-read-many) afin de protéger les données, les métadonnées d'objet définies par l'utilisateur et le balisage d'objet S3. Vous configurez les compartiments WORM pour autoriser la création de nouveaux objets et empêcher la surécriture ou la suppression du contenu existant. Utilisez l'une des approches décrites ici.

Pour garantir que les écrasements soient toujours refusés, vous pouvez :

- Dans le Grid Manager, accédez à **Configuration** > **Sécurité** > **Paramètres de sécurité** > **Réseau et objets**, et cochez la case **Prevent client modification**.

- Appliquez les règles et politiques S3 suivantes :
  - Ajoutez une opération DENY PutOverwriteObject à la stratégie S3.
  - Ajoutez une opération DENY DeleteObject à la stratégie S3.
  - Ajoutez une opération PutObject ALLOW à la stratégie S3.

> Le paramètre DeleteObject sur REFUSER dans une stratégie S3 n'empêche pas ILM de supprimer des objets lorsqu'une règle telle que « zéro copie après 30 jours » existe.

> Même en appliquant toutes ces règles et politiques, elles ne protègent pas contre les écritures simultanées (voir Situation A). Elles protègent en revanche contre les écrasements séquentiels complets (voir Situation B).

${post_edited_translations.segment}

```
/mybucket/important.doc
PUT#1 ---> OK
PUT#2 -------> OK
```

Situation B : Écrasements séquentiels terminés (protégés)

```
/mybucket/important.doc
PUT#1 -------> PUT#2 ---X (denied)
```

Informations connexes

- "Comment les règles ILM de StorageGRID gèrent les objets"
- "${post_edited_translations.segment}"
- "Exemples de stratégies de groupe"
- "${post_edited_translations.segment}"
- "Gérez les objets avec ILM"
- "Utilisez un compte locataire"

Exemple de stratégie de session de l'API REST StorageGRID S3

Utilisez l'exemple suivant pour créer une stratégie de session StorageGRID.

Exemple : Configurer une stratégie de session autorisant la récupération d'objets

Dans cet exemple, le principal de la session est uniquement autorisé à récupérer des objets du bucket1. Toutes les autres actions sont implicitement refusées, sauf pour les actions spécifiques à StorageGRID, telles que l'utilisation de la permission "s3:PutOverwriteObject". La politique de session peut être fournie sous forme de fichier JSON lors de l'appel à l'API AssumeRole.

```
{
  "Statement": [
    {
      "Action": "s3:GetObject",
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::bucket1/*"
    }
  ]
}
```

Exemples de stratégie de compartiment de l'API REST StorageGRID S3

Utilisez les exemples de cette section pour créer des politiques d'accès StorageGRID pour les compartiments.

Les stratégies de compartiment définissent les autorisations d'accès pour le compartiment auquel la stratégie est associée. Vous configurez une stratégie de compartiment à l'aide de l'API S3 PutBucketPolicy via l'un de ces outils :

- "Gestionnaire de locataires".
- AWS CLI en utilisant cette commande (voir "Opérations sur les compartiments") :

```
> aws s3api put-bucket-policy --bucket examplebucket --policy
file://policy.json
```

Exemple : Autoriser tout le monde à accéder en lecture seule à un compartiment

Dans cet exemple, tout le monde, y compris les utilisateurs anonymes, est autorisé à lister les objets du compartiment et à effectuer des opérations GetObject sur tous les objets du compartiment. Toutes les autres opérations seront refusées. Notez que cette politique peut s'avérer peu utile, car seul le compte racine dispose des autorisations d'écriture dans le compartiment.

```
{
  "Statement": [
    {
      "Sid": "AllowEveryoneReadOnlyAccess",
      "Effect": "Allow",
      "Principal": "*",
      "Action": [ "s3:GetObject", "s3:ListBucket" ],
      "Resource":
["arn:aws:s3:::examplebucket","arn:aws:s3:::examplebucket/*"]
    }
  ]
}
```

Exemple : Autoriser à tous les utilisateurs d'un compte un accès complet, et à tous les utilisateurs d'un autre compte un accès en lecture seule à un compartiment

Dans cet exemple, tous les membres d'un compte spécifié bénéficient d'un accès complet à un compartiment, tandis que tous les membres d'un autre compte spécifié sont uniquement autorisés à lister le compartiment et à effectuer des opérations GetObject sur les objets du compartiment commençant par le préfixe de clé de l'objet `shared/`.

> Dans StorageGRID, les objets créés par un compte autre que le propriétaire (y compris les comptes anonymes) appartiennent au compte propriétaire du compartiment. La politique du compartiment s'applique à ces objets.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "95390887230002558202"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "31181711887329436680"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::examplebucket/shared/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "31181711887329436680"
      },
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::examplebucket",
      "Condition": {
        "StringLike": {
          "s3:prefix": "shared/*"
        }
      }
    }
  ]
}
```

Exemple : Autoriser tout le monde à accéder en lecture seule à un compartiment et à accorder un accès complet à un groupe spécifique

Dans cet exemple, tout le monde, y compris les utilisateurs anonymes, est autorisé à lister le compartiment et à effectuer des opérations GetObject sur tous les objets du compartiment, tandis que seuls les utilisateurs appartenant au groupe `Marketing` dans le compte spécifié bénéficient d'un accès complet.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-
group/Marketing"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": ["s3:ListBucket","s3:GetObject"],
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    }
  ]
}
```

Exemple : Autoriser à tous la lecture et l'écriture dans un bucket si le client se trouve dans la plage d'adresses IP

Dans cet exemple, toute personne, y compris anonyme, est autorisée à lister le bucket et à effectuer des opérations sur tous les objets du bucket, à condition que les requêtes proviennent d'une plage d'adresses IP spécifiée (54.240.143.0 à 54.240.143.255, sauf 54.240.143.188). Toutes les autres opérations seront refusées et toutes les requêtes en dehors de la plage d'adresses IP seront refusées.

```
{
  "Statement": [
    {
      "Sid": "AllowEveryoneReadWriteAccessIfInSourceIpRange",
      "Effect": "Allow",
      "Principal": "*",
      "Action": [ "s3:*Object", "s3:ListBucket" ],
      "Resource":
["arn:aws:s3:::examplebucket","arn:aws:s3:::examplebucket/*"],
      "Condition": {
        "IpAddress": {"aws:SourceIp": "54.240.143.0/24"},
        "NotIpAddress": {"aws:SourceIp": "54.240.143.188"}
      }
    }
  ]
}
```

Exemple : Autoriser l'accès complet à un compartiment exclusivement à un utilisateur fédéré spécifié

Dans cet exemple, l'utilisateur fédéré Alex bénéficie d'un accès complet au `examplebucket` bucket et à ses objets. Tous les autres utilisateurs, y compris 'root', se voient explicitement refuser toutes les opérations. Notez cependant que 'root' ne se voit jamais refuser les autorisations de Put/Get/DeleteBucketPolicy.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-user/Alex"
      },
      "Action": [
        "s3:*"
      ],
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Deny",
      "NotPrincipal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-user/Alex"
      },
      "Action": [
        "s3:*"
      ],
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    }
  ]
}
```

Exemple : autorisation PutOverwriteObject

Dans cet exemple, l' `Deny`effet pour PutOverwriteObject et DeleteObject garantit que personne ne peut écraser ou supprimer les données de l'objet, les métadonnées définies par l'utilisateur et le balisage de l'objet S3.

```
{
  "Statement": [
    {
      "Effect": "Deny",
      "Principal": "*",
      "Action": [
        "s3:PutOverwriteObject",
        "s3:DeleteObject",
        "s3:DeleteObjectVersion"
      ],
      "Resource": "arn:aws:s3:::wormbucket/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-
group/SomeGroup"

},
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::wormbucket"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-
group/SomeGroup"

},
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::wormbucket/*"
    }
  ]
}
```

Exemples de stratégie de groupe pour l'API REST S3 de StorageGRID

Utilisez les exemples de cette section pour créer des politiques d'accès StorageGRID pour les groupes.

Les stratégies de groupe définissent les autorisations d'accès du groupe auquel la stratégie est associée. Il n'y a pas d' `Principal` élément dans la stratégie car il est implicite. Les stratégies de groupe sont configurées à l'aide du Tenant Manager ou de l'API.

Exemple : Définir une stratégie de groupe à l'aide de Tenant Manager

Lorsque vous ajoutez ou modifiez un groupe dans le Tenant Manager, vous pouvez sélectionner une stratégie de groupe pour déterminer les autorisations d'accès S3 dont disposeront les membres de ce groupe. Voir "Créer des groupes pour un locataire S3".

- **Accès S3 interdit** : Option par défaut. Les utilisateurs de ce groupe n'ont pas accès aux ressources S3, sauf si l'accès est accordé par une stratégie de compartiment. Si vous sélectionnez cette option, seul l'utilisateur racine aura accès aux ressources S3 par défaut.
- **Accès en lecture seule** : Les utilisateurs de ce groupe disposent d'un accès en lecture seule aux ressources S3. Par exemple, les utilisateurs de ce groupe peuvent lister les objets et lire les données des objets, les métadonnées et les balises. Lorsque vous sélectionnez cette option, la chaîne JSON pour une stratégie de groupe en lecture seule apparaît dans la zone de texte. Vous ne pouvez pas modifier cette chaîne.
- **Accès complet** : Les utilisateurs de ce groupe disposent d'un accès complet aux ressources S3, y compris les compartiments. Lorsque vous sélectionnez cette option, la chaîne JSON de la stratégie de groupe d'accès complet s'affiche dans la zone de texte. Vous ne pouvez pas modifier cette chaîne.
- **Mesures de protection contre les ransomwares** : Cette stratégie s'applique à tous les compartiments de ce locataire. Les utilisateurs de ce groupe peuvent effectuer des actions courantes, mais ne peuvent pas supprimer définitivement d'objets des compartiments pour lesquels le versionnage des objets est activé.

  Les utilisateurs de Tenant Manager disposant de l'autorisation « Manage all buckets » peuvent ignorer cette stratégie de groupe. Limitez l'autorisation « Manage all buckets » aux utilisateurs de confiance et utilisez l'authentification multifacteur (MFA) lorsque cela est possible.

- **Personnalisé** : Les utilisateurs du groupe reçoivent les autorisations que vous spécifiez dans la zone de texte.

Exemple : Accorder au groupe un accès complet à tous les compartiments

Dans cet exemple, tous les membres du groupe sont autorisés à accéder pleinement à tous les compartiments appartenant au compte du locataire, sauf interdiction explicite par la politique du compartiment.

```
{
  "Statement": [
    {
      "Action": "s3:*",
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::*"
    }
  ]
}
```

Exemple : Autoriser l'accès en lecture seule du groupe à tous les compartiments

Dans cet exemple, tous les membres du groupe disposent d'un accès en lecture seule aux ressources S3, sauf interdiction explicite dans la stratégie du compartiment. Par exemple, les utilisateurs de ce groupe peuvent lister les objets et consulter les données, métadonnées et étiquettes des objets.

```
{
  "Statement": [
    {
      "Sid": "AllowGroupReadOnlyAccess",
      "Effect": "Allow",
      "Action": [
        "s3:ListAllMyBuckets",
        "s3:ListBucket",
        "s3:ListBucketVersions",
        "s3:GetObject",
        "s3:GetObjectTagging",
        "s3:GetObjectVersion",
        "s3:GetObjectVersionTagging"
      ],
      "Resource": "arn:aws:s3:::*"
    }
  ]
}
```

Exemple : Accorder aux membres d'un groupe un accès complet uniquement à leur « dossier » dans un compartiment

Dans cet exemple, les membres du groupe sont uniquement autorisés à consulter et à accéder à leur dossier spécifique (préfixe de clé) dans le compartiment indiqué. Notez que les autorisations d'accès provenant d'autres stratégies de groupe et de la stratégie du compartiment doivent être prises en compte pour déterminer la confidentialité de ces dossiers.

```
{
  "Statement": [
    {
      "Sid": "AllowListBucketOfASpecificUserPrefix",
      "Effect": "Allow",
      "Action": "s3:ListBucket",
      "Resource": "arn:aws:s3:::department-bucket",
      "Condition": {
        "StringLike": {
          "s3:prefix": "${aws:username}/*"
        }
      }
    },
    {
      "Sid": "AllowUserSpecificActionsOnlyInTheSpecificUserPrefix",
      "Effect": "Allow",
      "Action": "s3:*Object",
      "Resource": "arn:aws:s3:::department-bucket/${aws:username}/*"
    }
  ]
}
```

Opérations S3 suivies dans les journaux d'audit de StorageGRID

Les services StorageGRID génèrent des messages d'audit et les enregistrent dans des fichiers journaux texte. Vous pouvez consulter les messages d'audit spécifiques à S3 dans le journal des audits pour obtenir des détails sur les opérations de compartiment et d'objet.

${post_edited_translations.segment}

- CreateBucket
- DeleteBucket
- DeleteBucketTagging
- DeleteObjects
- GetBucketTagging
- HeadBucket
- ListObjects
- ListObjectVersions
- Conformité PUT Bucket
- PutBucketTagging

- PutBucketVersioning

Opérations sur les objets suivies dans les journaux des audits

- CompleteMultipartUpload
- CopyObject
- DeleteObject
- GetObject
- HeadObject
- PutObject
- RestoreObject
- SelectObject
- UploadPart (lorsqu'une règle ILM utilise l'ingestion équilibrée ou stricte)
- UploadPartCopy (lorsqu'une règle ILM utilise l'ingestion équilibrée ou stricte)

Informations connexes

- "Accéder au fichier journal des audits"
- "${post_edited_translations.segment}"
- "Messages de journal des audits lus par le client"

Informations sur le copyright

Informations sur les marques commerciales