



So implementiert StorageGRID die S3- REST-API

StorageGRID 11.7

NetApp
April 12, 2024

Inhalt

- So implementiert StorageGRID die S3-REST-API 1
- In Konflikt stehende Clientanforderungen 1
- Konsistenzkontrollen 1
- Managen von Objekten durch StorageGRID ILM-Regeln 4
- Objektversionierung 5
- Konfigurieren Sie die S3-Objektsperre über die S3-REST-API 6
- S3-Lebenszykluskonfiguration erstellen 12
- Empfehlungen für die Implementierung der S3-REST-API 16

So implementiert StorageGRID die S3-REST-API

In Konflikt stehende Clientanforderungen

Widersprüchliche Clientanforderungen, wie z. B. zwei Clients, die in denselben Schlüssel schreiben, werden auf der Grundlage der „neuesten Wins“ gelöst.

Der Zeitpunkt für die Bewertung „neuester Erfolge“ basiert auf dem Zeitpunkt, an dem das StorageGRID System eine bestimmte Anforderung abgeschlossen hat und nicht auf dem Zeitpunkt, an dem S3-Clients einen Vorgang starten.

Konsistenzkontrollen

Konsistenzkontrollen sorgen für ein Gleichgewicht zwischen der Verfügbarkeit der Objekte und der Konsistenz dieser Objekte über verschiedene Storage Nodes und Standorte hinweg, wie von Ihrer Anwendung gefordert.

Standardmäßig garantiert StorageGRID eine Lese-/Nachher-Konsistenz für neu erstellte Objekte. Jeder GET nach einem erfolgreich abgeschlossenen PUT wird in der Lage sein, die neu geschriebenen Daten zu lesen. Überschreibungen vorhandener Objekte, Metadatenaktualisierungen und -Löschungen sind schließlich konsistent. Überschreibungen dauern in der Regel nur wenige Sekunden oder Minuten, können jedoch bis zu 15 Tage in Anspruch nehmen.

Wenn Sie Objektvorgänge auf einer anderen Konsistenzstufe ausführen möchten, können Sie für jeden Bucket oder für jeden API-Vorgang eine Konsistenzkontrolle angeben.

Konsistenzkontrollen

Die Konsistenzkontrolle beeinflusst die Verteilung der Metadaten, die StorageGRID zum Verfolgen von Objekten zwischen Nodes verwendet, und somit die Verfügbarkeit von Objekten für Client-Anforderungen.

Sie können die Konsistenzkontrolle für einen Bucket- oder API-Vorgang auf einen der folgenden Werte festlegen:

- **All:** Alle Knoten erhalten die Daten sofort, oder die Anfrage schlägt fehl.
- **Strong-global:** Garantiert Lese-After-Write-Konsistenz für alle Kundenanfragen über alle Standorte hinweg.
- **Strong-site:** Garantiert Lese-After-Write Konsistenz für alle Kundenanfragen innerhalb einer Site.
- **Read-after-New-write:** (Standard) bietet Read-after-write-Konsistenz für neue Objekte und eventuelle Konsistenz für Objektaktualisierungen. Hochverfügbarkeit und garantierte Datensicherung Empfohlen für die meisten Fälle.
- **Verfügbar:** Bietet eventuelle Konsistenz für neue Objekte und Objekt-Updates. Verwenden Sie für S3-Buckets nur nach Bedarf (z. B. für einen Bucket mit Protokollwerten, die nur selten gelesen werden, oder für HEAD- oder GET-Vorgänge für nicht vorhandene Schlüssel). Nicht unterstützt für S3 FabricPool-Buckets.

Verwenden Sie die Consistency Controls „read-after-New-write“ und „available“

Wenn bei einem HEAD oder GET-Vorgang die Konsistenzkontrolle „read-after-New-write“ verwendet wird, führt StorageGRID die Suche in mehreren Schritten durch:

- Es sieht zunächst das Objekt mit einer niedrigen Konsistenz.
- Wenn diese Suche fehlschlägt, wiederholt sie die Suche auf der nächsten Konsistenzebene, bis sie eine Konsistenzstufe erreicht, die dem Verhalten für Strong-Global entspricht.

Wenn eine HEAD- oder GET-Operation das Konsistenzsteuerelement „read-after-New-write“ verwendet, das Objekt aber nicht existiert, erreicht die Objekt-Lookup immer eine Konsistenzstufe, die dem Verhalten für strong-global entspricht. Da für diese Konsistenzstufe mehrere Kopien der Objektmetadaten an jedem Standort verfügbar sein müssen, können Sie eine hohe Anzahl von 500 internen Serverfehlern erhalten, wenn zwei oder mehr Storage-Nodes am selben Standort nicht verfügbar sind.

Sofern Sie keine Konsistenzgarantien ähnlich wie Amazon S3 benötigen, können Sie diese Fehler bei DEN HEAD- und GET-Vorgängen vermeiden, indem Sie die Konsistenzkontrolle auf „Available“ setzen. Wenn bei einem HEAD oder GET-Vorgang die Konsistenzkontrolle „Available“ verwendet wird, bietet StorageGRID eventuell nur Konsistenz. Bei einem fehlgeschlagenen Vorgang wird nicht erneut versucht, die Konsistenzstufen zu erhöhen, daher müssen nicht mehrere Kopien der Objekt-Metadaten verfügbar sein.

Festlegen der Konsistenzkontrolle für den API-Betrieb

Um die Consistency Control für einen einzelnen API-Vorgang festzulegen, müssen für den Vorgang Konsistenzkontrollen unterstützt werden, und Sie müssen die Consistency Control in der Anforderungs-Kopfzeile angeben. In diesem Beispiel wird die Consistency Control auf „strong-site“ für EINE GET Object Operation gesetzt.

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



Sie müssen für DEN PUT-Objekt- und DEN GET-Objektbetrieb dasselbe Konsistenzsteuerelement verwenden.

Festlegen der Konsistenzkontrolle für Bucket

Zum Festlegen der Konsistenzkontrolle für Bucket können Sie die StorageGRID PUT Bucket-Konsistenzanforderung und DIE ANFORDERUNG FÜR GET-Bucket-Konsistenz verwenden. Alternativ können Sie den Tenant Manager oder die Mandantenmanagement-API verwenden.

Beachten Sie beim Festlegen der Konsistenzkontrollen für einen Bucket Folgendes:

- Durch das Festlegen der Konsistenzkontrolle für einen Bucket wird festgelegt, welche Konsistenzkontrolle für S3-Operationen verwendet wird, die für Objekte im Bucket oder in der Bucket-Konfiguration durchgeführt werden. Er hat keine Auswirkungen auf die Vorgänge auf dem Bucket selbst.

- Die Konsistenzkontrolle für einen einzelnen API-Vorgang überschreibt die Konsistenzkontrolle für den Bucket.
- Im Allgemeinen sollten Buckets die Standardkonsistenzkontrolle „read-after-New-write.“ verwenden. Wenn die Anforderungen nicht korrekt funktionieren, ändern Sie das Client-Verhalten der Anwendung, wenn möglich. Oder konfigurieren Sie den Client so, dass für jede API-Anforderung das Consistency Control angegeben wird. Legen Sie die Consistency Control auf Bucket-Ebene nur als letztes Resort fest.

wie Konsistenzkontrollen und ILM-Regeln interagieren, um den Datenschutz zu beeinträchtigen

Die Wahl der Konsistenzkontrolle und der ILM-Regel haben Auswirkungen auf den Schutz von Objekten. Diese Einstellungen können interagieren.

Die beim Speichern eines Objekts verwendete Konsistenzkontrolle beeinflusst beispielsweise die anfängliche Platzierung von Objekt-Metadaten, während das für die ILM-Regel ausgewählte Aufnahmeverhalten sich auf die anfängliche Platzierung von Objektkopien auswirkt. Da StorageGRID Zugriff auf die Metadaten eines Objekts und seine Daten benötigt, um Kundenanforderungen zu erfüllen, kann die Auswahl der passenden Sicherungsstufen für Konsistenz und Aufnahme-Verhalten eine bessere Erstsicherung und zuverlässigere Systemantworten ermöglichen.

Die folgenden Aufnahmeverhalten stehen für ILM-Regeln zur Verfügung:

- **Dual Commit:** StorageGRID erstellt sofort Zwischenkopien des Objekts und gibt den Erfolg an den Client zurück. Kopien, die in der ILM-Regel angegeben sind, werden nach Möglichkeit erstellt.
- **Streng:** Alle in der ILM-Regel angegebenen Kopien müssen erstellt werden, bevor der Erfolg an den Client zurückgesendet wird.
- **Ausgewogen:** StorageGRID versucht bei der Aufnahme alle in der ILM-Regel festgelegten Kopien zu erstellen; wenn dies nicht möglich ist, werden Zwischenkopien erstellt und der Erfolg an den Client zurückgesendet. Die Kopien, die in der ILM-Regel angegeben sind, werden, wenn möglich gemacht.



Lesen Sie vor der Auswahl des Aufnahmeverhaltens für eine ILM-Regel die vollständige Beschreibung dieser Einstellungen in den Anweisungen zum Managen von Objekten mit Information Lifecycle Management.

Beispiel für die Interaktion zwischen Konsistenzkontrolle und ILM-Regel

Angenommen, Sie haben ein Grid mit zwei Standorten mit der folgenden ILM-Regel und der folgenden Einstellung für die Konsistenzstufe:

- **ILM-Regel:** Erstellen Sie zwei Objektkopien, eine am lokalen Standort und eine an einem entfernten Standort. Das strikte Aufnahmeverhalten wird ausgewählt.
- **Konsistenzstufe:** „strong-global“ (Objektmetadaten werden sofort auf alle Standorte verteilt.)

Wenn ein Client ein Objekt im Grid speichert, erstellt StorageGRID sowohl Objektkopien als auch verteilt Metadaten an beiden Standorten, bevor der Kunde zum Erfolg zurückkehrt.

Das Objekt ist zum Zeitpunkt der Aufnahme der Nachricht vollständig gegen Verlust geschützt. Wenn beispielsweise der lokale Standort kurz nach der Aufnahme verloren geht, befinden sich Kopien der Objektdaten und der Objektmetadaten am Remote-Standort weiterhin. Das Objekt kann vollständig abgerufen werden.

Falls Sie stattdessen dieselbe ILM-Regel und die Konsistenzstufe „strong-site“ verwendet haben, erhält

der Client möglicherweise eine Erfolgsmeldung, nachdem die Objektdaten an den Remote Standort repliziert wurden, aber bevor die Objektmetadaten dort verteilt werden. In diesem Fall entspricht die Sicherung von Objektmetadaten nicht dem Schutzniveau für Objektdaten. Falls der lokale Standort kurz nach der Aufnahme verloren geht, gehen Objektmetadaten verloren. Das Objekt kann nicht abgerufen werden.

Die Wechselbeziehung zwischen Konsistenzstufen und ILM-Regeln kann komplex sein. Wenden Sie sich an NetApp, wenn Sie Hilfe benötigen.

Verwandte Informationen

["Objektmanagement mit ILM"](#)

["Get Bucket-Konsistenz"](#)

["PUT Bucket-Konsistenz"](#)

Managen von Objekten durch StorageGRID ILM-Regeln

Der Grid-Administrator erstellt Informationen Lifecycle Management (ILM)-Regeln für das Management von Objektdaten, die von S3-REST-API-Client-Applikationen in das StorageGRID-System aufgenommen werden. Diese Regeln werden dann zur ILM-Richtlinie hinzugefügt, um zu bestimmen, wie und wo Objektdaten im Laufe der Zeit gespeichert werden.

ILM-Einstellungen bestimmen die folgenden Aspekte eines Objekts:

- **Geographie**

Der Speicherort der Objektdaten kann entweder im StorageGRID-System (Storage-Pool) oder in einem Cloud-Storage-Pool gespeichert werden.

- * Speicherklasse*

Storage-Typ zur Speicherung von Objektdaten, z. B. Flash oder rotierende Festplatte

- **Verlustschutz**

Wie viele Kopien erstellt werden und welche Arten von Kopien erstellt werden: Replizierung, Erasure Coding oder beides.

- **Aufbewahrung**

Es ändert sich im Laufe der Zeit, wie Objektdaten verwaltet werden, wo sie gespeichert sind und wie sie vor Verlust geschützt sind.

- **Schutz während der Aufnahme**

Methode zum Schutz von Objektdaten bei der Aufnahme: Synchroner Platzierung (mit ausgeglichenen oder strengen Optionen für das Aufnahmeverhalten) oder Erstellung von vorläufigen Kopien (unter Verwendung der Option Dual-Commit)

ILM-Regeln können Objekte filtern und auswählen. Bei mit S3 aufgenommenen Objekten können ILM-Regeln Objekte auf Basis der folgenden Metadaten filtern:

- Mandantenkonto
- Bucket-Name
- Aufnahmezeit
- Taste
- Zeitpunkt des letzten Zugriffs



Standardmäßig werden Updates der letzten Zugriffszeit für alle S3 Buckets deaktiviert. Wenn Ihr StorageGRID System eine ILM-Regel enthält, die die Option „Letzte Zugriffszeit“ verwendet, müssen Sie für die in dieser Regel angegebenen S3 Buckets Updates für den letzten Zugriff aktivieren. Verwenden Sie die Anforderung ZUM letzten Zugriff auf Bucket, den Tenant Manager (siehe) "[Aktiviert bzw. deaktiviert Updates der letzten Zugriffszeit](#)") Oder die Mandanten-Management-API. Beachten Sie bei der Aktivierung von Updates der letzten Zugriffszeit, dass die Performance von StorageGRID möglicherweise reduziert wird, insbesondere bei Systemen mit kleinen Objekten.

- Positionsbeschränkung
- Objektgröße
- Benutzer-Metadaten
- Objekt-Tag

Verwandte Informationen

["Verwenden Sie ein Mandantenkonto"](#)

["Objektmanagement mit ILM"](#)

["PUT Bucket-Zeit für den letzten Zugriff"](#)

Objektversionierung

Sie können mithilfe der Versionierung mehrere Versionen eines Objekts aufbewahren, das vor versehentlichem Löschen von Objekten schützt und Ihnen das Abrufen und Wiederherstellen älterer Versionen eines Objekts ermöglicht.

Das StorageGRID System implementiert Versionierung mit Unterstützung für die meisten Funktionen und weist einige Einschränkungen auf. StorageGRID unterstützt bis zu 1,000 Versionen jedes Objekts.

Die Objektversionierung kann mit StorageGRID Information Lifecycle Management (ILM) oder mit der S3 Bucket Lifecycle-Konfiguration kombiniert werden. Sie müssen für jeden Bucket die Versionierung aktivieren, um diese Funktion für den Bucket zu aktivieren. Jedem Objekt im Bucket wird eine Version-ID zugewiesen, die vom StorageGRID-System generiert wird.

Die Verwendung von MFA (Multi-Faktor-Authentifizierung) Löschen wird nicht unterstützt.



Die Versionierung kann nur auf Buckets aktiviert werden, die mit StorageGRID Version 10.3 oder höher erstellt wurden.

ILM und Versionierung

ILM-Richtlinien werden auf jede Version eines Objekts angewendet. Ein ILM-Scanprozess scannt kontinuierlich alle Objekte und bewertet sie anhand der aktuellen ILM-Richtlinie neu. Alle Änderungen, die Sie an ILM-Richtlinien vornehmen, werden auf alle zuvor aufgenommenen Objekte angewendet. Dies umfasst bereits aufgenommene Versionen, wenn die Versionierung aktiviert ist. Beim ILM-Scannen werden neue ILM-Änderungen an zuvor aufgenommenen Objekten angewendet.

Bei S3-Objekten in versionierungsfähigen Buckets können Sie mithilfe der Versionierungsunterstützung ILM-Regeln erstellen, die „noncurrent time“ als Referenzzeit verwenden. Wählen Sie **Ja** für die Frage „Diese Regel nur auf ältere Objektversionen anwenden?“ aus. Zoll "[Schritt 1 des Assistenten zum Erstellen einer ILM-Regel](#)". Wenn ein Objekt aktualisiert wird, werden seine vorherigen Versionen nicht aktuell. Mit dem Filter „noncurrent time“ können Sie Richtlinien erstellen, die die Auswirkungen früherer Objektversionen auf den Storage verringern.



Wenn Sie eine neue Version eines Objekts über einen mehrteiligen Upload-Vorgang hochladen, wird der nicht aktuelle Zeitpunkt für die Originalversion des Objekts angezeigt, wenn der mehrteilige Upload für die neue Version erstellt wurde, nicht erst nach Abschluss des mehrteiligen Uploads. In begrenzten Fällen kann die nicht aktuelle Zeit der ursprünglichen Version Stunden oder Tage früher als die Zeit für die aktuelle Version sein.

Siehe "[ILM-Regeln und Richtlinien für versionierte S3-Objekte \(Beispiel 4\)](#)".

Konfigurieren Sie die S3-Objektsperre über die S3-REST-API

Wenn die globale S3-Objektsperre für Ihr StorageGRID-System aktiviert ist, können Sie Buckets mit aktivierter S3-Objektsperre erstellen. Sie können für jeden Bucket oder die Aufbewahrungseinstellungen für jede Objektversion die Standardaufbewahrung festlegen.

Aktivieren der S3-Objektsperre für einen Bucket

Wenn die globale S3-Objektsperreinstellung für Ihr StorageGRID-System aktiviert ist, können Sie bei der Erstellung jedes Buckets optional die S3-Objektsperre aktivieren.

S3 Object Lock ist eine permanente Einstellung, die nur beim Erstellen eines Buckets aktiviert werden kann. Sie können S3-Objektsperre nicht hinzufügen oder deaktivieren, nachdem ein Bucket erstellt wurde.

Verwenden Sie eine der folgenden Methoden, um S3 Object Lock für einen Bucket zu aktivieren:

- Erstellen Sie den Bucket mit Tenant Manager. Siehe "[S3-Bucket erstellen](#)".
- Erstellen Sie den Bucket mithilfe einer PUT-Bucket-Anforderung zusammen mit dem `x-amz-bucket-object-lock-enabled` Kopfzeile der Anfrage. Siehe "[Operationen auf Buckets](#)".

S3 Object Lock erfordert eine Bucket-Versionierung, die beim Erstellen des Buckets automatisch aktiviert wird. Die Versionierung für den Bucket kann nicht unterbrochen werden. Siehe "[Objektversionierung](#)".

Standardeinstellungen für die Aufbewahrung eines Buckets

Wenn S3 Object Lock für einen Bucket aktiviert ist, können Sie optional die Standardaufbewahrung für den

Bucket aktivieren und einen Standardaufbewahrungsmodus und die Standardaufbewahrungsdauer festlegen.

Standardaufbewahrungsmodus

- Im COMPLIANCE-Modus:
 - Das Objekt kann erst gelöscht werden, wenn das Aufbewahrungsdatum erreicht ist.
 - Das Aufbewahrungsdatum des Objekts kann erhöht, aber nicht verringert werden.
 - Das Aufbewahrungsdatum des Objekts kann erst entfernt werden, wenn dieses Datum erreicht ist.
- Im GOVERNANCE-Modus:
 - Benutzer mit `s3:BypassGovernanceRetention` Berechtigung kann den verwenden `x-amz-bypass-governance-retention: true` Kopfzeile anfordern, um Aufbewahrungseinstellungen zu umgehen.
 - Diese Benutzer können eine Objektversion löschen, bevor das Aufbewahrungsdatum erreicht ist.
 - Diese Benutzer können das Aufbewahrungsdatum eines Objekts erhöhen, verringern oder entfernen.

Standardaufbewahrungszeitraum

Für jeden Bucket kann ein Standardaufbewahrungszeitraum in Jahren oder Tagen angegeben werden.

Festlegen der Standardaufbewahrung für einen Bucket

Verwenden Sie eine der folgenden Methoden, um die Standardaufbewahrung für einen Bucket festzulegen:

- Managen Sie die Bucket-Einstellungen über den Tenant Manager. Siehe ["Erstellen eines S3-Buckets"](#) Und ["Aktualisieren Sie die S3 Object Lock-Standardaufbewahrung"](#).
- Stellen Sie eine ANFORDERUNG ZUR OBJEKTSPERRKONFIGURATION für den Bucket AUS, um den Standardmodus und die Standardanzahl von Tagen oder Jahren festzulegen.

PUT Objekt Lock-Konfiguration

Mit DER ANFORDERUNG „OBJEKTSPERRKONFIGURATION“ KÖNNEN Sie den Standardaufbewahrungsmodus und den Standardaufbewahrungszeitraum für einen Bucket festlegen und ändern, für den S3 Object Lock aktiviert ist. Sie können auch zuvor konfigurierte Standardeinstellungen entfernen.

Wenn neue Objektversionen in den Bucket aufgenommen werden, wird der standardmäßige Aufbewahrungsmodus angewendet, wenn `x-amz-object-lock-mode` Und `x-amz-object-lock-retain-until-date` Sind nicht angegeben. Der Standardaufbewahrungszeitraum wird verwendet, um das Aufbewahrungsdatum von IF zu berechnen `x-amz-object-lock-retain-until-date` Ist nicht angegeben.

Wenn der Standardaufbewahrungszeitraum nach der Aufnahme einer Objektversion geändert wird, bleibt das „bis-Aufbewahrung“-Datum der Objektversion identisch und wird im neuen Standardaufbewahrungszeitraum nicht neu berechnet.

Sie müssen die haben `s3:PutBucketObjectLockConfiguration` Berechtigung, oder Konto root, um diesen Vorgang abzuschließen.

Der `Content-MD5` Der Anforderungskopf muss in der PUT-Anforderung angegeben werden.

Anforderungsbeispiel

In diesem Beispiel wird S3 Object Lock für einen Bucket aktiviert und der Standardaufbewahrungsmodus auf COMPLIANCE und der Standardaufbewahrungszeitraum auf 6 Jahre festgelegt.

```
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>
```

Bestimmen der Standardaufbewahrung für einen Bucket

Verwenden Sie eine der folgenden Methoden, um zu ermitteln, ob S3 Object Lock für einen Bucket aktiviert ist und den Standardaufbewahrungsmodus und den Standardaufbewahrungszeitraum anzuzeigen:

- Zeigen Sie den Bucket im Tenant Manager an. Siehe "[S3 Buckets anzeigen](#)".
- Geben Sie eine Anforderung ZUM ABRUFEN der Objektsperrenkonfiguration aus.

Konfiguration der Objektsperre ABRUFEN

Mit der Anforderung OBJEKTSPERRKONFIGURATION ABRUFEN können Sie festlegen, ob S3 Object Lock für einen Bucket aktiviert ist. Ist diese Option aktiviert, können Sie prüfen, ob für den Bucket ein Standardaufbewahrungsmodus und eine Aufbewahrungsfrist konfiguriert sind.

Wenn neue Objektversionen in den Bucket aufgenommen werden, wird der standardmäßige Aufbewahrungsmodus angewendet, wenn `x-amz-object-lock-mode` ist nicht angegeben. Der Standardaufbewahrungszeitraum wird verwendet, um das Aufbewahrungsdatum von IF zu berechnen `x-amz-object-lock-retain-until-date` ist nicht angegeben.

Sie müssen die haben `s3:GetBucketObjectLockConfiguration` Berechtigung, oder Konto root, um diesen Vorgang abzuschließen.

Anforderungsbeispiel

```
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
```

Antwortbeispiel

```
HTTP/1.1 200 OK
x-amz-id-2:
iVmcB7OXXJRkRH1FiVq1151/T24gRfpwpuZrEG11Bb9ImOMAAe98oxSpX1knabA0LTvBYJpSIX
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>
```

Festlegen von Aufbewahrungseinstellungen für ein Objekt

Ein Bucket mit aktivierter S3-Objektsperre kann eine Kombination von Objekten mit und ohne Aufbewahrungseinstellungen für S3-Objektsperre enthalten.

Aufbewahrungseinstellungen auf Objektebene werden über die S3-REST-API angegeben. Die Aufbewahrungseinstellungen für ein Objekt überschreiben alle Standardaufbewahrungseinstellungen für den Bucket.

Sie können für jedes Objekt die folgenden Einstellungen festlegen:

- **Retention Mode:** Entweder COMPLIANCE oder GOVERNANCE.
- **Bis-Datum behalten:** Ein Datum, das angibt, wie lange die Objektversion von StorageGRID beibehalten

werden muss.

- Wenn im COMPLIANCE-Modus das Aufbewahrungsdatum in der Zukunft liegt, kann das Objekt abgerufen, aber nicht geändert oder gelöscht werden. Das Aufbewahrungsdatum kann erhöht werden, aber dieses Datum kann nicht verringert oder entfernt werden.
- Im GOVERNANCE-Modus können Benutzer mit besonderer Berechtigung die Einstellung „bis zum Datum behalten“ umgehen. Sie können eine Objektversion löschen, bevor der Aufbewahrungszeitraum abgelaufen ist. Außerdem können sie das Aufbewahrungsdatum erhöhen, verringern oder sogar entfernen.
- **Legal Hold:** Die Anwendung eines gesetzlichen Hold auf eine Objektversion sperrt diesen Gegenstand sofort. Beispielsweise müssen Sie ein Objekt, das mit einer Untersuchung oder einem Rechtsstreit zusammenhängt, rechtlich festhalten. Eine gesetzliche Aufbewahrungspflichten haben kein Ablaufdatum, bleiben aber bis zur ausdrücklichen Entfernung erhalten.

Die Legal Hold-Einstellung für ein Objekt ist unabhängig vom Aufbewahrungsmodus und dem Aufbewahrungsdatum. Befindet sich eine Objektversion unter einem Legal Hold, kann diese Version nicht gelöscht werden.

Wenn Sie beim Hinzufügen einer Objektversion zu einem Bucket S3-Objektsperreinstellungen angeben möchten, geben Sie ein "PUT Objekt", "PUT Objekt - Kopieren", Oder "Initiieren Von Mehrteiligen Uploads" Anfrage.

Sie können Folgendes verwenden:

- `x-amz-object-lock-mode`, Die COMPLIANCE oder GOVERNANCE sein können (Groß-/Kleinschreibung beachten).



Wenn Sie angeben `x-amz-object-lock-mode`, Sie müssen auch angeben `x-amz-object-lock-retain-until-date`.

- `x-amz-object-lock-retain-until-date`
 - Der Wert für „bis-Datum beibehalten“ muss das Format aufweisen `2020-08-10T21:46:00Z`. Fraktionale Sekunden sind zulässig, aber nur 3 Dezimalstellen bleiben erhalten (Präzision in Millisekunden). Andere ISO 8601-Formate sind nicht zulässig.
 - Das „Aufbewahrung bis“-Datum muss in der Zukunft liegen.
- `x-amz-object-lock-legal-hold`

Wenn die gesetzliche Aufbewahrungspflichten LIEGEN (Groß-/Kleinschreibung muss beachtet werden), wird das Objekt unter einer gesetzlichen Aufbewahrungspflichten platziert. Wenn die gesetzliche Aufbewahrungspflichten AUS DEM WEG gehen, wird keine gesetzliche Aufbewahrungspflichten platziert. Jeder andere Wert führt zu einem 400-Fehler (InvalidArgument).

Wenn Sie eine dieser Anfrageheadern verwenden, beachten Sie die folgenden Einschränkungen:

- Der `Content-MD5` Der Anforderungskopf ist erforderlich `x-amz-object-lock-*` In DER PUT-Objektanforderung ist eine Anforderungsüberschrift vorhanden. `Content-MD5` Ist für PUT Object – Copy oder Initiierung von mehrteiligen Uploads nicht erforderlich.
- Wenn für den Bucket die S3-Objektsperre nicht aktiviert ist und ein `x-amz-object-lock-*` Der Anforderungskopf ist vorhanden, es wird ein 400-Fehler (InvalidRequest) zurückgegeben.
- Die PUT-Objektanforderung unterstützt die Verwendung von `x-amz-storage-class`:

REDUCED_REDUNDANCY Passend zum Verhalten von AWS. Wird ein Objekt jedoch mit aktivierter S3-Objektsperre in einen Bucket aufgenommen, führt StorageGRID immer eine Dual-Commit-Aufnahme durch.

- Eine nachfolgende ANTWORT AUF GET- oder HEAD Object-Version enthält die Kopfzeilen `x-amz-object-lock-mode`, `x-amz-object-lock-retain-until-date`, und `x-amz-object-lock-legal-hold`, Wenn konfiguriert und wenn der Anforderungssender die richtige hat `s3:Get*` Berechtigungen.

Sie können das verwenden `s3:object-lock-remaining-retention-days` Policy Condition Key zur Begrenzung der minimalen und maximalen zulässigen Aufbewahrungsfristen für Ihre Objekte.

Aktualisieren von Aufbewahrungseinstellungen für ein Objekt

Wenn Sie die Einstellungen für die gesetzliche Aufbewahrungs- oder Aufbewahrungseinstellung einer vorhandenen Objektversion aktualisieren müssen, können Sie die folgenden Vorgänge der Unterressource des Objekts ausführen:

- `PUT Object legal-hold`

Wenn der neue Legal-Hold-Wert AKTIVIERT ist, wird das Objekt unter einer gesetzlichen Aufbewahrungspflichten platziert. Wenn DER Rechtsvorenthalten-Wert DEAKTIVIERT ist, wird die gesetzliche Aufbewahrungspflichten aufgehoben.

- `PUT Object retention`
 - Der Wert des Modus kann COMPLIANCE oder GOVERNANCE sein (Groß-/Kleinschreibung muss beachtet werden).
 - Der Wert für „bis-Datum beibehalten“ muss das Format aufweisen `2020-08-10T21:46:00Z`. Fraktionale Sekunden sind zulässig, aber nur 3 Dezimalstellen bleiben erhalten (Präzision in Millisekunden). Andere ISO 8601-Formate sind nicht zulässig.
 - Wenn eine Objektversion über ein vorhandenes Aufbewahrungsdatum verfügt, können Sie sie nur erhöhen. Der neue Wert muss in der Zukunft liegen.

So verwenden Sie DEN GOVERNANCE-Modus

Benutzer, die über das verfügen `s3:BypassGovernanceRetention` Berechtigung kann die aktiven Aufbewahrungseinstellungen eines Objekts umgehen, das DEN GOVERNANCE-Modus verwendet. Alle LÖSCHVORGÄNGE für die Objektaufbewahrung müssen den enthalten `x-amz-bypass-governance-retention:true` Kopfzeile der Anfrage. Diese Benutzer können die folgenden zusätzlichen Vorgänge ausführen:

- Führen SIE VORGÄNGE ZUM LÖSCHEN von Objekten aus oder LÖSCHEN Sie mehrere Objekte, um eine Objektversion vor Ablauf des Aufbewahrungszeitraums zu löschen.

Objekte, die sich unter einem Legal Hold befinden, können nicht gelöscht werden. Legal Hold muss DEAKTIVIERT sein.

- Führen SIE PUT Objektaufbewahrungsvorgänge durch, bei denen der Modus einer Objektversion von GOVERNANCE zu COMPLIANCE geändert wird, bevor der Aufbewahrungszeitraum des Objekts abgelaufen ist.

Die Änderung des Modus von COMPLIANCE zu GOVERNANCE ist niemals zulässig.

- Führen SIE PUT Objektaufbewahrungsvorgänge durch, um den Aufbewahrungszeitraum einer Objektversion zu erhöhen, zu verringern oder zu entfernen.

Verwandte Informationen

- ["Objekte managen mit S3 Object Lock"](#)
- ["Verwenden Sie S3 Objektsperre, um Objekte beizubehalten"](#)
- ["Amazon Simple Storage Service Benutzerhandbuch: S3 Object Lock verwenden"](#)

S3-Lebenszykluskonfiguration erstellen

Sie können eine S3-Lebenszykluskonfiguration erstellen, um zu steuern, wann bestimmte Objekte aus dem StorageGRID System gelöscht werden.

Das einfache Beispiel in diesem Abschnitt veranschaulicht, wie eine S3-Lebenszykluskonfiguration das Löschen bestimmter Objekte aus bestimmten S3-Buckets kontrollieren kann. Das Beispiel in diesem Abschnitt dient nur zu Illustrationszwecken. Weitere Informationen zum Erstellen von S3-Lebenszykluskonfigurationen finden Sie unter ["Amazon Simple Storage Service Developer Guide: Lifecycle Management von Objekten"](#). Beachten Sie, dass StorageGRID nur Aktionen nach Ablauf unterstützt. Es werden keine Aktionen zur Transition unterstützt.

Welche Lifecycle-Konfiguration ist

Eine Lifecycle-Konfiguration ist ein Satz von Regeln, die auf die Objekte in bestimmten S3-Buckets angewendet werden. Jede Regel gibt an, welche Objekte betroffen sind und wann diese Objekte ablaufen (an einem bestimmten Datum oder nach einigen Tagen).

StorageGRID unterstützt in einer Lebenszykluskonfiguration bis zu 1,000 Lebenszyklusregeln. Jede Regel kann die folgenden XML-Elemente enthalten:

- Ablauf: Löschen eines Objekts, wenn ein bestimmtes Datum erreicht wird oder wenn eine bestimmte Anzahl von Tagen erreicht wird, beginnend mit dem Zeitpunkt der Aufnahme des Objekts.
- NoncurrentVersionExpiration: Löschen Sie ein Objekt, wenn eine bestimmte Anzahl von Tagen erreicht wird, beginnend ab dem Zeitpunkt, an dem das Objekt nicht mehr aktuell wurde.
- Filter (Präfix, Tag)
- Status
- ID

Wenn Sie eine Lifecycle-Konfiguration auf einen Bucket anwenden, überschreiben die Lifecycle-Einstellungen für den Bucket immer die StorageGRID-ILM-Einstellungen. StorageGRID verwendet die Verfallseinstellungen für den Bucket und nicht ILM, um zu bestimmen, ob bestimmte Objekte gelöscht oder aufbewahrt werden sollen.

Aus diesem Grund kann ein Objekt aus dem Grid entfernt werden, obwohl die Speicheranweisungen in einer ILM-Regel noch auf das Objekt gelten. Alternativ kann ein Objekt auch dann im Grid aufbewahrt werden, wenn eine ILM-Platzierungsanleitung für das Objekt abgelaufen ist. Weitere Informationen finden Sie unter ["Funktionsweise von ILM während der gesamten Nutzungsdauer eines Objekts"](#).



Die Bucket-Lifecycle-Konfiguration kann für Buckets verwendet werden, für die S3-Objektsperre aktiviert ist. Die Bucket-Lifecycle-Konfiguration wird jedoch für ältere Buckets, die Compliance verwenden, nicht unterstützt.

StorageGRID unterstützt den Einsatz der folgenden Bucket-Operationen zum Management der Lebenszykluskonfigurationen:

- Bucket-Lebenszyklus LÖSCHEN
- BUCKET-Lebenszyklus ABRUFEN
- PUT Bucket-Lebenszyklus

Lebenszyklukonfiguration erstellen

Als erster Schritt beim Erstellen einer Lebenszykluskonfiguration erstellen Sie eine JSON-Datei mit einem oder mehreren Regeln. Diese JSON-Datei enthält beispielsweise drei Regeln:

1. Regel 1 gilt nur für Objekte, die mit dem Präfix übereinstimmen `category1/` Und das hat ein `key2` Der Wert von `tag2`. Der `Expiration` Der Parameter gibt an, dass Objekte, die dem Filter entsprechen, um Mitternacht am 22. August 2020 ablaufen.
2. Regel 2 gilt nur für Objekte, die mit dem Präfix übereinstimmen `category2/`. Der `Expiration` Parameter gibt an, dass Objekte, die dem Filter entsprechen, 100 Tage nach der Aufnahme ablaufen.



Regeln, die eine Anzahl von Tagen angeben, sind relativ zu dem Zeitpunkt, an dem das Objekt aufgenommen wurde. Wenn das aktuelle Datum das Aufnahmedatum plus die Anzahl der Tage überschreitet, werden einige Objekte möglicherweise aus dem Bucket entfernt, sobald die Lebenszykluskonfiguration angewendet wird.

3. Regel 3 gilt nur für Objekte, die dem Präfix entsprechen `category3/`. Der `Expiration` Parameter gibt an, dass nicht aktuelle Versionen übereinstimmender Objekte 50 Tage nach deren Nichtstrom ablaufen.

```

{
  "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"
    }
  ]
}

```


Lifecycle-Konfiguration auf Bucket anwenden

Nachdem Sie die Lifecycle-Konfigurationsdatei erstellt haben, wenden Sie sie durch Ausgabe einer PUT Bucket Lifecycle-Anforderung auf einen Bucket an.

Diese Anforderung wendet die Lebenszykluskonfiguration in der Beispieldatei auf Objekte in einem Bucket mit dem Namen `testbucket`.

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

Um zu überprüfen, ob eine Lifecycle-Konfiguration erfolgreich auf den Bucket angewendet wurde, geben Sie eine ANFORDERUNG FÜR DEN GET Bucket-Lebenszyklus aus. Beispiel:

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

Eine erfolgreiche Antwort zeigt die Konfiguration des Lebenszyklus, die Sie gerade angewendet haben.

Überprüfen, ob der Bucket-Lebenszyklus für das Objekt gilt

Sie können feststellen, ob eine Ablaufregel in der Lebenszykluskonfiguration auf ein bestimmtes Objekt angewendet wird, wenn Sie eine PUT-Objekt-, HEAD-Objekt- oder GET-Objektanforderung ausgeben. Wenn eine Regel zutrifft, enthält die Antwort ein `Expiration` Parameter, der angibt, wann das Objekt abläuft und welche Ablaufregel übereinstimmt.



Da der Bucket-Lebenszyklus ILM überschreibt, wird der `expiry-date` Hier wird das tatsächliche Datum angezeigt, an dem das Objekt gelöscht wird. Weitere Informationen finden Sie unter "[Wie die Aufbewahrung von Objekten bestimmt wird](#)".

Zum Beispiel, diese PUT Objekt Anfrage wurde am 22. Juni 2020 und platziert ein Objekt in der `testbucket` Eimer.

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

Die Erfolgsreaktion zeigt an, dass das Objekt in 100 Tagen (01. Oktober 2020) abläuft und dass es mit Regel 2 der Lebenszykluskonfiguration übereinstimmt.

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

Diese HEAD Object-Anfrage wurde beispielsweise verwendet, um Metadaten für dasselbe Objekt im Testbucket zu erhalten.

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

Die Erfolgsreaktion umfasst die Metadaten des Objekts und gibt an, dass das Objekt in 100 Tagen abläuft und dass es mit Regel 2 übereinstimmt.

```
{
  "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": {}
}
```

Empfehlungen für die Implementierung der S3-REST-API

Bei der Implementierung der S3-REST-API zur Verwendung mit StorageGRID sollten Sie diese Empfehlungen beachten.

Empfehlungen für Köpfe zu nicht vorhandenen Objekten

Wenn Ihre Anwendung regelmäßig prüft, ob ein Objekt an einem Pfad vorhanden ist, wo Sie nicht erwarten, dass das Objekt tatsächlich existiert, sollten Sie die Konsistenzkontrolle „available“ verwenden. Verwenden Sie zum Beispiel die Konsistenzkontrolle „Available“, wenn Ihre Anwendung einen Speicherort vor DEM ANSETZEN an sie leitet.

Andernfalls werden möglicherweise 500 Fehler des internen Servers angezeigt, wenn ein oder mehrere Speicherknoten nicht verfügbar sind.

Sie können die Konsistenzkontrolle „Available“ für jeden Bucket mithilfe der PUT Bucket-Konsistenzanforderung festlegen oder Sie können die Konsistenzkontrolle in der Anforderungs-Kopfzeile für einen einzelnen API-Vorgang festlegen.

Empfehlungen für Objektschlüssel

Befolgen Sie diese Empfehlungen für Objektschlüsselnamen auf Basis des ersten Erstellens des Buckets.

Buckets, die in StorageGRID 11.4 oder früher erstellt wurden

- Verwenden Sie keine Zufallswerte als die ersten vier Zeichen von Objektschlüsseln. Dies steht im Gegensatz zu der früheren AWS Empfehlung für wichtige Präfixe. Verwenden Sie stattdessen nicht zufällige, nicht eindeutige Präfixe, z. B. `image`.
- Wenn Sie der früheren AWS-Empfehlung folgen, zufällige und eindeutige Zeichen in Schlüsselpräfixen zu verwenden, setzen Sie den Objektschlüsseln einen Verzeichnisnamen vor. Verwenden Sie dieses Format:

```
mybucket/mydir/f8e3-image3132.jpg
```

Anstelle dieses Formats:

```
mybucket/f8e3-image3132.jpg
```

Buckets, die in StorageGRID 11.4 oder höher erstellt wurden

Es ist nicht erforderlich, Objektschlüsselnamen auf die Best Practices für die Performance zu beschränken. In den meisten Fällen können Sie zufällige Werte für die ersten vier Zeichen von Objektschlüsselnamen verwenden.



Eine Ausnahme ist ein S3-Workload, der nach kurzer Zeit kontinuierlich alle Objekte entfernt. Um die Auswirkungen auf die Performance in diesem Anwendungsfall zu minimieren, variieren Sie alle tausend Objekte mit einem ähnlichen Datum einen führenden Teil des Schlüsselnamens. Angenommen, ein S3-Client schreibt in der Regel 2,000 Objekte/Sekunde, und die ILM- oder Bucket-Lifecycle-Richtlinie entfernt alle Objekte nach drei Tagen. Um die Auswirkungen auf die Performance zu minimieren, können Sie Schlüssel anhand eines Musters wie folgt benennen: `/mybucket/mydir/yyyymddhhmmss-random_UUID.jpg`

Empfehlungen für „Range reads“

Wenn der ["Globale Option zum Komprimieren gespeicherter Objekte"](#) Ist aktiviert, sollten S3-Client-Applikationen die Ausführung VON OPERATIONEN FÜR DAS ABRUFEN von Objekten verhindern, die einen Bereich von zurückgegebenen Bytes angeben. Diese Vorgänge „range Read“ sind ineffizient, da StorageGRID die Objekte effektiv dekomprimieren muss, um auf die angeforderten Bytes zugreifen zu können. VORGÄNGE ZUM ABRUFEN von Objekten, die einen kleinen Byte-Bereich von einem sehr großen Objekt anfordern, sind besonders ineffizient, beispielsweise ist es ineffizient, einen Bereich von 10 MB von einem komprimierten 50-GB-Objekt zu lesen.

Wenn Bereiche von komprimierten Objekten gelesen werden, können Client-Anforderungen eine Zeitdauer haben.



Wenn Sie Objekte komprimieren müssen und Ihre Client-Applikation Bereichslesevorgänge verwenden muss, erhöhen Sie die Zeitüberschreitung beim Lesen der Anwendung.

Verwandte Informationen

- ["Konsistenzkontrollen"](#)
- ["PUT Bucket-Konsistenz"](#)
- ["StorageGRID verwalten"](#)

Copyright-Informationen

Copyright © 2024 NetApp. Alle Rechte vorbehalten. Gedruckt in den USA. Dieses urheberrechtlich geschützte Dokument darf ohne die vorherige schriftliche Genehmigung des Urheberrechtinhabers in keiner Form und durch keine Mittel – weder grafische noch elektronische oder mechanische, einschließlich Fotokopieren, Aufnehmen oder Speichern in einem elektronischen Abrufsystem – auch nicht in Teilen, vervielfältigt werden.

Software, die von urheberrechtlich geschütztem NetApp Material abgeleitet wird, unterliegt der folgenden Lizenz und dem folgenden Haftungsausschluss:

DIE VORLIEGENDE SOFTWARE WIRD IN DER VORLIEGENDEN FORM VON NETAPP ZUR VERFÜGUNG GESTELLT, D. H. OHNE JEGLICHE EXPLIZITE ODER IMPLIZITE GEWÄHRLEISTUNG, EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE STILLSCHWEIGENDE GEWÄHRLEISTUNG DER MARKTGÄNGIGKEIT UND EIGNUNG FÜR EINEN BESTIMMTEN ZWECK, DIE HIERMIT AUSGESCHLOSSEN WERDEN. NETAPP ÜBERNIMMT KEINERLEI HAFTUNG FÜR DIREKTE, INDIREKTE, ZUFÄLLIGE, BESONDERE, BEISPIELHAFT SCHÄDEN ODER FOLGESCHÄDEN (EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE BESCHAFFUNG VON ERSATZWAREN ODER -DIENSTLEISTUNGEN, NUTZUNGS-, DATEN- ODER GEWINNVERLUSTE ODER UNTERBRECHUNG DES GESCHÄFTSBETRIEBS), UNABHÄNGIG DAVON, WIE SIE VERURSACHT WURDEN UND AUF WELCHER HAFTUNGSTHEORIE SIE BERUHEN, OB AUS VERTRAGLICH FESTGELEGTER HAFTUNG, VERSCHULDENSUNABHÄNGIGER HAFTUNG ODER DELIKTSHAFTUNG (EINSCHLIESSLICH FAHRLÄSSIGKEIT ODER AUF ANDEREM WEGE), DIE IN IRGEND EINER WEISE AUS DER NUTZUNG DIESER SOFTWARE RESULTIEREN, SELBST WENN AUF DIE MÖGLICHKEIT DERARTIGER SCHÄDEN HINGEWIESEN WURDE.

NetApp behält sich das Recht vor, die hierin beschriebenen Produkte jederzeit und ohne Vorankündigung zu ändern. NetApp übernimmt keine Verantwortung oder Haftung, die sich aus der Verwendung der hier beschriebenen Produkte ergibt, es sei denn, NetApp hat dem ausdrücklich in schriftlicher Form zugestimmt. Die Verwendung oder der Erwerb dieses Produkts stellt keine Lizenzierung im Rahmen eines Patentrechts, Markenrechts oder eines anderen Rechts an geistigem Eigentum von NetApp dar.

Das in diesem Dokument beschriebene Produkt kann durch ein oder mehrere US-amerikanische Patente, ausländische Patente oder anhängige Patentanmeldungen geschützt sein.

ERLÄUTERUNG ZU „RESTRICTED RIGHTS“: Nutzung, Vervielfältigung oder Offenlegung durch die US-Regierung unterliegt den Einschränkungen gemäß Unterabschnitt (b)(3) der Klausel „Rights in Technical Data – Noncommercial Items“ in DFARS 252.227-7013 (Februar 2014) und FAR 52.227-19 (Dezember 2007).

Die hierin enthaltenen Daten beziehen sich auf ein kommerzielles Produkt und/oder einen kommerziellen Service (wie in FAR 2.101 definiert) und sind Eigentum von NetApp, Inc. Alle technischen Daten und die Computersoftware von NetApp, die unter diesem Vertrag bereitgestellt werden, sind gewerblicher Natur und wurden ausschließlich unter Verwendung privater Mittel entwickelt. Die US-Regierung besitzt eine nicht ausschließliche, nicht übertragbare, nicht unterlizenzierbare, weltweite, limitierte unwiderrufliche Lizenz zur Nutzung der Daten nur in Verbindung mit und zur Unterstützung des Vertrags der US-Regierung, unter dem die Daten bereitgestellt wurden. Sofern in den vorliegenden Bedingungen nicht anders angegeben, dürfen die Daten ohne vorherige schriftliche Genehmigung von NetApp, Inc. nicht verwendet, offengelegt, vervielfältigt, geändert, aufgeführt oder angezeigt werden. Die Lizenzrechte der US-Regierung für das US-Verteidigungsministerium sind auf die in DFARS-Klausel 252.227-7015(b) (Februar 2014) genannten Rechte beschränkt.

Markeninformationen

NETAPP, das NETAPP Logo und die unter <http://www.netapp.com/TM> aufgeführten Marken sind Marken von NetApp, Inc. Andere Firmen und Produktnamen können Marken der jeweiligen Eigentümer sein.