



Nutzen Sie die Swift REST API

StorageGRID

NetApp
October 03, 2025

Inhalt

Verwenden Sie Swift	1
Unterstützung für OpenStack Swift API in StorageGRID	1
Geschichte der Unterstützung von Swift API in StorageGRID	1
So implementiert StorageGRID die Swift REST API	2
Empfehlungen für die Implementierung der Swift REST API	3
Mandantenkonten und -Verbindungen werden konfiguriert	4
Erstellen und Konfigurieren von Swift Mandantenkonten	4
Wie Client-Verbindungen konfiguriert werden können	5
Testen der Verbindung in der Swift API-Konfiguration	8
Von Swift UNTERSTÜTZTE REST-API-Operationen	9
Von StorageGRID unterstützte Vorgänge	9
Gemeinsame Antwortheader für alle Vorgänge	10
Unterstützte Swift-API-Endpunkte	10
Konto-Operationen	12
Container-Operationen	13
Objekt-Operationen	16
OPTIONEN anfordern	20
Fehlerantworten bei Swift-API-Operationen	21
StorageGRID Swift REST-API-Operationen	22
ABRUFEN der Container-Konsistenzanforderung	22
PUT Container-Konsistenzanforderung	25
Sicherheit wird für DIE REST API konfiguriert	27
So bietet StorageGRID Sicherheit für DIE REST-API	27
Unterstützte Hashing- und Verschlüsselungsalgorithmen für TLS-Bibliotheken	29
Monitoring und Auditing von Vorgängen	30
Monitoring der Objekteinspeisung und -Abrufen	30
Aufrufen und Prüfen von Prüfprotokollen	32

Verwenden Sie Swift

Lesen Sie, wie Client-Applikationen die OpenStack Swift API zur Schnittstelle mit dem StorageGRID System nutzen.

- ["Unterstützung für OpenStack Swift API in StorageGRID"](#)
- ["Mandantenkonten und -Verbindungen werden konfiguriert"](#)
- ["Von Swift UNTERSTÜTZTE REST-API-Operationen"](#)
- ["StorageGRID Swift REST-API-Operationen"](#)
- ["Sicherheit wird für DIE REST API konfiguriert"](#)
- ["Monitoring und Auditing von Vorgängen"](#)

Unterstützung für OpenStack Swift API in StorageGRID

StorageGRID unterstützt die folgenden spezifischen Versionen von Swift und HTTP.

Element	Version
Swift-Spezifikation	OpenStack Swift Objekt Storage API v1 ab November 2015
HTTP	1.1 Weitere Informationen zu HTTP finden Sie unter HTTP/1.1 (RFCs 7230-35). Hinweis: StorageGRID unterstützt HTTP/1.1-Pipelining nicht.

Verwandte Informationen

["OpenStack: Objekt-Storage-API"](#)

Geschichte der Unterstützung von Swift API in StorageGRID

Bei Änderungen an der Unterstützung des StorageGRID-Systems für die Swift REST-API sollten Sie auf dieser hinweisen.

Freigabe	Kommentare
11.5	Schwache Konsistenzkontrolle entfernt. Stattdessen wird die verfügbare Konsistenzstufe verwendet.
11.4	Unterstützung für TLS 1.3 und aktualisierte Liste der unterstützten TLS-Chiffre-Suites hinzugefügt. CLB ist veraltet. Beschreibung der Wechselbeziehung zwischen ILM und KonsistenzEinstellung hinzugefügt

Freigabe	Kommentare
11.3	Aktualisierte PUT-Objektvorgänge zur Beschreibung der Auswirkungen von ILM-Regeln, die synchrone Platzierung bei der Aufnahme verwenden (die ausgewogenen und strengen Optionen für das Aufnahmeverhalten) Eine zusätzliche Beschreibung der Client-Verbindungen, die Load Balancer-Endpunkte oder Hochverfügbarkeitsgruppen verwenden. Aktualisierte Liste der unterstützten TLS-Cipher-Suites. TLS 1.1-Chiffren werden nicht mehr unterstützt.
11.2	Kleine redaktionelle Änderungen des Dokuments.
11.1	Zusätzlicher Support für die Verwendung von HTTP für Swift-Client-Verbindungen zu Grid-Nodes. Die Definitionen der Konsistenzkontrollen wurden aktualisiert.
11.0	Hinzugefügter Support für 1,000 Container für jedes Mandantenkonto.
10.3	Administrative Aktualisierungen und Korrekturen des Dokuments. Abschnitte zum Konfigurieren von benutzerdefinierten Serverzertifikaten entfernt.
10.2	Unterstützung der Swift API durch das StorageGRID System zu Beginn. Die derzeit unterstützte Version ist OpenStack Swift Object Storage API v1.

So implementiert StorageGRID die Swift REST API

Eine Client-Applikation kann mithilfe von Swift REST-API-Aufrufen eine Verbindung zu Storage-Nodes und Gateway-Nodes herstellen, um Container zu erstellen und Objekte zu speichern und abzurufen. Dadurch können serviceorientierte Applikationen, die für OpenStack Swift entwickelt wurden, mit lokalem Objekt-Storage des StorageGRID Systems verbunden werden.

Swift Objekt-Management

Nach der Aufnahme von Swift Objekten im StorageGRID System werden sie von den Regeln für Information Lifecycle Management (ILM) der aktiven ILM-Richtlinie des Systems gemanagt. Die ILM-Regeln und -Richtlinie bestimmen, wie StorageGRID Kopien von Objektdaten erstellt und verteilt und wie diese Kopien mit der Zeit gemanagt werden. Eine ILM-Regel kann beispielsweise für Objekte in bestimmten Swift Containern gelten und möglicherweise angeben, dass mehrere Objektkopien für eine bestimmte Anzahl von Jahren in mehreren Datacentern gespeichert werden.

Wenden Sie sich an Ihren StorageGRID-Administrator, wenn Sie wissen müssen, welche Auswirkungen die ILM-Regeln und Richtlinien des Grid auf die Objekte in Ihrem Swift-Mandantenkonto haben.

In Konflikt stehende Clientanforderungen

Widersprüchliche Clientanforderungen, wie z. B. zwei Clients, die in denselben Schlüssel schreiben, werden auf „latest-WINS“-Basis gelöst. Der Zeitpunkt für die Auswertung „latest-WINS“ basiert darauf, wann das StorageGRID System eine bestimmte Anfrage abschließt und nicht auf, wann Swift-Clients einen Vorgang starten.

Konsistenzgarantien und -Kontrollen

Standardmäßig bietet StorageGRID Lese-/Nachher-Konsistenz für neu erstellte Objekte und schließlich die Konsistenz von Objekt-Updates und HEAD-Operationen. 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.

StorageGRID ermöglicht Ihnen außerdem die Kontrolle der Konsistenz einzelner Container. Sie können die Consistency Control ändern, um je nach Anforderung zwischen der Verfügbarkeit der Objekte und der Konsistenz dieser Objekte in verschiedenen Storage Nodes und Standorten einen Kompromiss zwischen der Verfügbarkeit der Objekte herzustellen.

Verwandte Informationen

["Objektmanagement mit ILM"](#)

["ABRUFEN der Container-Konsistenzanforderung"](#)

["PUT Container-Konsistenzanforderung"](#)

Empfehlungen für die Implementierung der Swift REST API

Bei der Implementierung der Swift 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 in einem Pfad existiert, in dem Sie nicht erwarten, dass das Objekt tatsächlich vorhanden ist, sollten Sie die Konsistenzkontrolle „available“ verwenden. Sie sollten z. B. die Konsistenzkontrolle „Available“ verwenden, wenn Ihre Anwendung EINEN HEAD-Vorgang an einem Speicherort ausführt, bevor Sie einen PUT-Vorgang an diesen Ort ausführen.

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

Sie können die Konsistenzkontrolle „verfügbar“ für jeden Container mithilfe der PUT Container-Konsistenzanforderung festlegen.

Empfehlungen für Objektnamen

Als die ersten vier Zeichen von Objektnamen sollten keine Zufallswerte verwendet werden. Stattdessen sollten Sie nicht-zufällige, nicht-eindeutige Präfixe verwenden, wie z. B. Bild.

Wenn Sie in Objektnamen-Präfixen zufällige und eindeutige Zeichen verwenden müssen, sollten Sie die Objektnamen mit einem Verzeichnisnamen vorschreiben. Verwenden Sie dieses Format:

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

Anstelle dieses Formats:

```
mycontainer/f8e3-image3132.jpg
```

Empfehlungen für „Range reads“

Wenn die Option **komprimiere gespeicherte Objekte** ausgewählt ist (**Konfiguration > Systemeinstellungen > Grid-Optionen**), sollten Swift-Client-Anwendungen vermeiden, GET-Objektoperationen durchzuführen, die einen Bereich von Bytes angeben. Diese Vorgänge „range Read“ sind ineffizient, da StorageGRID die Objekte effektiv dekomprimieren muss, um auf die angeforderten Bytes zugreifen zu können. GET-Objektvorgänge, die einen kleinen Byte-Bereich von einem sehr großen Objekt anfordern, sind besonders ineffizient, beispielsweise ist es sehr 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

["ABRUFEN der Container-Konsistenzanforderung"](#)

["PUT Container-Konsistenzanforderung"](#)

["StorageGRID verwalten"](#)

Mandantenkonten und -Verbindungen werden konfiguriert

Wenn StorageGRID konfiguriert wird, um Verbindungen von Client-Applikationen zu akzeptieren, müssen ein oder mehrere Mandantenkonten erstellt und die Verbindungen eingerichtet werden.

Erstellen und Konfigurieren von Swift Mandantenkonten

Bevor Swift API-Clients Objekte auf StorageGRID speichern und abrufen können, ist ein Swift-Mandantenkonto erforderlich. Jedes Mandantenkonto hat seine eigene Konto-ID, Gruppen und Benutzer sowie Container und Objekte.

Swift-Mandantenkonten werden von einem StorageGRID Grid-Administrator mit dem Grid Manager oder der Grid Management API erstellt.

Beim Erstellen eines Swift-Mandantenkontos gibt der Grid-Administrator folgende Informationen an:

- Anzeigename für den Mandanten (die Konto-ID des Mandanten wird automatisch zugewiesen und kann nicht geändert werden)

- Optional: Ein Storage-Kontingent für das Mandantenkonto – die maximale Anzahl der Gigabyte, Terabyte oder Petabyte, die für die Mandantenobjekte verfügbar sind. Das Storage-Kontingent eines Mandanten stellt eine logische Menge (Objektgröße) und keine physische Menge (Größe auf der Festplatte) dar.
- Wenn Single Sign-On (SSO) nicht für das StorageGRID-System verwendet wird, gibt das Mandantenkonto seine eigene Identitätsquelle an oder teilt die Identitätsquelle des Grid mit, und zwar mit dem anfänglichen Passwort für den lokalen Root-Benutzer des Mandanten.
- Wenn SSO aktiviert ist, hat die föderierte Gruppe Root-Zugriffsberechtigungen, um das Mandantenkonto zu konfigurieren.

Nach der Erstellung eines Swift-Mandantenkontos können Benutzer mit Root Access-Berechtigung auf den Mandanten-Manager zugreifen, um Aufgaben wie die folgenden durchzuführen:

- Einrichten von Identitätsföderation (es sei denn, die Identitätsquelle wird gemeinsam mit dem Grid verwendet) und Erstellen lokaler Gruppen und Benutzer
- Monitoring der Storage-Auslastung



Swift-Benutzer müssen über die Root-Zugriffsberechtigung für den Zugriff auf den Mandanten-Manager verfügen. Die Root-Zugriffsberechtigung ermöglicht Benutzern jedoch nicht, sich in der Swift REST-API zu authentifizieren, um Container zu erstellen und Objekte aufzunehmen. Benutzer müssen über die Swift-Administratorberechtigung verfügen, um sich bei der Swift-REST-API zu authentifizieren.

Verwandte Informationen

["StorageGRID verwalten"](#)

["Verwenden Sie ein Mandantenkonto"](#)

["Unterstützte Swift-API-Endpunkte"](#)

Wie Client-Verbindungen konfiguriert werden können

Ein Grid-Administrator trifft Konfigurationsmöglichkeiten, die Einfluss darauf haben, wie Swift-Clients sich mit StorageGRID verbinden, um Daten zu speichern und abzurufen. Die spezifischen Informationen, die benötigt werden, um eine Verbindung herzustellen, hängen von der gewählten Konfiguration ab.

Client-Applikationen können Objekte speichern oder abrufen, indem sie eine Verbindung mit folgenden Komponenten herstellen:

- Der Lastverteilungsservice an Admin-Nodes oder Gateway-Nodes oder optional die virtuelle IP-Adresse einer HA-Gruppe (High Availability, Hochverfügbarkeit) von Admin-Nodes oder Gateway-Nodes
- Der CLB-Dienst auf Gateway-Knoten oder optional die virtuelle IP-Adresse einer Hochverfügbarkeitsgruppe von Gateway-Knoten



Der CLB-Service ist veraltet. Clients, die vor der Version StorageGRID 11.3 konfiguriert wurden, können den CLB-Service auf Gateway-Knoten weiterhin verwenden. Alle anderen Client-Applikationen, die zum Lastausgleich vom StorageGRID abhängig sind, sollten über den Load Balancer Service eine Verbindung herstellen.

- Storage-Nodes mit oder ohne externen Load Balancer

Bei der Konfiguration von StorageGRID kann ein Grid-Administrator den Grid-Manager oder die Grid-Management-API verwenden, um die folgenden Schritte auszuführen, die alle optional sind:

1. Konfigurieren von Endpunkten für den Load Balancer Service.

Sie müssen Endpunkte konfigurieren, um den Load Balancer Service verwenden zu können. Der Lastverteilungsservice an Admin-Nodes oder Gateway-Nodes verteilt eingehende Netzwerkverbindungen von Client-Anwendungen auf Storage-Nodes. Beim Erstellen eines Load Balancer-Endpunkts gibt der StorageGRID-Administrator eine Portnummer an, ob der Endpunkt HTTP- oder HTTPS-Verbindungen akzeptiert, der Client-Typ (S3 oder Swift), der den Endpunkt verwendet, und das für HTTPS-Verbindungen zu verwendende Zertifikat (falls zutreffend).

2. Konfigurieren Sie Nicht Vertrauenswürdige Client-Netzwerke.

Wenn ein StorageGRID-Administrator das Clientnetzwerk eines Node so konfiguriert, dass es nicht vertrauenswürdig ist, akzeptiert der Knoten nur eingehende Verbindungen im Clientnetzwerk an Ports, die explizit als Load Balancer-Endpunkte konfiguriert sind.

3. Konfigurieren Sie Hochverfügbarkeitsgruppen.

Wenn ein Administrator eine HA-Gruppe erstellt, werden die Netzwerkschnittstellen mehrerer Admin-Nodes oder Gateway-Nodes in einer aktiv-Backup-Konfiguration platziert. Client-Verbindungen werden mithilfe der virtuellen IP-Adresse der HA-Gruppe hergestellt.

Weitere Informationen zu den einzelnen Optionen finden Sie in den Anweisungen zur Administration von StorageGRID.

Zusammenfassung: IP-Adressen und Ports für Client-Verbindungen

Client-Applikationen stellen mithilfe der IP-Adresse eines Grid-Node und der Port-Nummer eines Service auf diesem Node eine Verbindung zu StorageGRID her. Bei Konfiguration von Hochverfügbarkeitsgruppen (High Availability, HA) können Client-Applikationen eine Verbindung über die virtuelle IP-Adresse der HA-Gruppe herstellen.

Zum Erstellen von Client-Verbindungen erforderliche Informationen

Die Tabelle fasst die verschiedenen Möglichkeiten zusammen, wie Clients eine Verbindung zu StorageGRID sowie zu den für die einzelnen Verbindungstypen verwendeten IP-Adressen und Ports herstellen können. Wenden Sie sich an Ihren StorageGRID-Administrator, um weitere Informationen zu erhalten, oder lesen Sie die Anweisungen zur Administration von StorageGRID, um eine Beschreibung der Informationen im Grid-Manager zu erhalten.

Wo eine Verbindung hergestellt wird	Dienst, mit dem der Client verbunden ist	IP-Adresse	Port
HA-Gruppe	Lastausgleich	Virtuelle IP-Adresse einer HA-Gruppe	• Endpunkt-Port des Load Balancer
HA-Gruppe	CLB Hinweis: der CLB-Service ist veraltet.	Virtuelle IP-Adresse einer HA-Gruppe	Swift-Standardports: • HTTPS: 8083 • HTTP: 8085
Admin-Node	Lastausgleich	IP-Adresse des Admin-Knotens	• Endpunkt-Port des Load Balancer

Wo eine Verbindung hergestellt wird	Dienst, mit dem der Client verbunden ist	IP-Adresse	Port
Gateway-Node	Lastausgleich	IP-Adresse des Gateway-Node	<ul style="list-style-type: none"> • Endpunkt-Port des Load Balancer
Gateway-Node	CLB Hinweis: der CLB-Service ist veraltet.	IP-Adresse des Gateway-Node Hinweis: standardmäßig sind HTTP-Ports für CLB und LDR nicht aktiviert.	Swift-Standardports: <ul style="list-style-type: none"> • HTTPS: 8083 • HTTP: 8085
Storage-Node	LDR	IP-Adresse des Speicherknoten	Swift-Standardports: <ul style="list-style-type: none"> • HTTPS: 18083 • HTTP: 18085

Beispiel

Verwenden Sie eine strukturierte URL, wie unten gezeigt, um einen Swift-Client mit dem Load Balancer-Endpunkt einer HA-Gruppe von Gateway-Nodes zu verbinden:

- `https://VIP-of-HA-group:LB-endpoint-port`

Wenn beispielsweise die virtuelle IP-Adresse der HA-Gruppe 192.0.2.6 lautet und die Portnummer eines Swift Load Balancer Endpunkts 10444 ist, kann ein Swift-Client die folgende URL zur Verbindung mit StorageGRID verwenden:

- `https://192.0.2.6:10444`

Ein DNS-Name kann für die IP-Adresse konfiguriert werden, die Clients zum Herstellen der Verbindung mit StorageGRID verwenden. Wenden Sie sich an Ihren Netzwerkadministrator vor Ort.

Entscheidung über die Verwendung von HTTPS- oder HTTP-Verbindungen

Wenn Client-Verbindungen mit einem Load Balancer-Endpunkt hergestellt werden, müssen Verbindungen über das Protokoll (HTTP oder HTTPS) hergestellt werden, das für diesen Endpunkt angegeben wurde. Um HTTP für Client-Verbindungen zu Storage-Nodes oder zum CLB-Dienst auf Gateway-Knoten zu verwenden, müssen Sie dessen Verwendung aktivieren.

Wenn Client-Anwendungen eine Verbindung zu Speicherknoten oder zum CLB-Dienst auf Gateway-Knoten herstellen, müssen sie für alle Verbindungen verschlüsseltes HTTPS verwenden. Optional können Sie weniger sichere HTTP-Verbindungen aktivieren, indem Sie im Grid Manager die Option **HTTP-Verbindung** aktivieren auswählen. Eine Client-Anwendung kann beispielsweise HTTP verwenden, wenn die Verbindung zu einem Speicherknoten in einer nicht produktiven Umgebung getestet wird.



Achten Sie bei der Aktivierung von HTTP für ein Produktionsraster darauf, dass die Anforderungen unverschlüsselt gesendet werden.



Der CLB-Service ist veraltet.

Wenn die Option **HTTP-Verbindung aktivieren** ausgewählt ist, müssen Clients für HTTP unterschiedliche Ports verwenden als für HTTPS. Lesen Sie die Anweisungen zum Verwalten von StorageGRID.

Verwandte Informationen

["StorageGRID verwalten"](#)

Testen der Verbindung in der Swift API-Konfiguration

Mit der Swift CLI können Sie die Verbindung zum StorageGRID System testen und überprüfen, ob Sie Objekte lesen und in das System schreiben können.

Was Sie benötigen

- Sie müssen Python-swiftclient, den Swift-Befehlszeilen-Client, heruntergeladen und installiert haben.
- Im StorageGRID System müssen Sie ein Swift Mandantenkonto haben.

Über diese Aufgabe

Wenn Sie keine Sicherheit konfiguriert haben, müssen Sie die hinzufügen `--insecure` Flag auf jeden dieser Befehle.

Schritte

1. Fragen Sie die Info-URL für Ihre StorageGRID Swift Implementierung:

```
swift
-U <Tenant_Account_ID:Account_User_Name>
-K <User_Password>
-A https://<FQDN | IP>:<Port>/info
capabilities
```

Dies reicht aus, um zu testen, ob Ihre Swift-Implementierung funktionsfähig ist. Um die Kontenkonfiguration durch Speichern eines Objekts weiter zu testen, fahren Sie mit den zusätzlichen Schritten fort.

2. Legen Sie ein Objekt in den Container:

```
touch test_object
swift
-U <Tenant_Account_ID:Account_User_Name>
-K <User_Password>
-A https://<FQDN | IP>:<Port>/auth/v1.0
upload test_container test_object
--object-name test_object
```

3. Holen Sie sich den Container, um das Objekt zu überprüfen:

```
swift
-U <Tenant_Account_ID:Account_User_Name>
-K <User_Password>
-A https://<FQDN | IP>:<Port>/auth/v1.0
list test_container
```

4. Löschen Sie das Objekt:

```
swift
-U <Tenant_Account_ID:Account_User_Name>
-K <User_Password>
-A https://<FQDN | IP>:<Port>/auth/v1.0
delete test_container test_object
```

5. Löschen Sie den Container:

```
swift
-U `<_Tenant_Account_ID:Account_User_Name_>`
-K `<_User_Password_>`
-A `https://<_FQDN_ | _IP_>:<_Port_>/auth/v1.0`
delete test_container
```

Verwandte Informationen

["Erstellen und Konfigurieren von Swift Mandantenkonten"](#)

["Sicherheit wird für DIE REST API konfiguriert"](#)

Von Swift UNTERSTÜTZTE REST-API-Operationen

Das StorageGRID System unterstützt die meisten Operationen in der OpenStack Swift API. Informieren Sie sich vor der Integration von Swift REST API Clients mit StorageGRID über die Implementierungsdetails für Konto-, Container- und Objektvorgänge.

Von StorageGRID unterstützte Vorgänge

Die folgenden Swift-API-Operationen werden unterstützt:

- ["Konto-Operationen"](#)
- ["Container-Operationen"](#)
- ["Objekt-Operationen"](#)

Gemeinsame Answerheader für alle Vorgänge

Das StorageGRID-System implementiert alle gemeinsamen Header für unterstützte Vorgänge, wie sie von der OpenStack Swift Objekt-Storage-API v1 definiert wurden.

Verwandte Informationen

["OpenStack: Objekt-Storage-API"](#)

Unterstützte Swift-API-Endpunkte

StorageGRID unterstützt die folgenden Swift-API-Endpunkte: Die Info-URL, die auth-URL und die Storage-URL.

Info-URL

Sie können die Funktionen und Einschränkungen der StorageGRID-Swift-Implementierung bestimmen, indem Sie eine GET-Anfrage an die Swift-Basis-URL mit dem /info-Pfad senden.

```
https://FQDN | Node IP:Swift Port/info/
```

In der Anfrage:

- *FQDN* Ist der vollständig qualifizierte Domain-Name.
- *Node IP* Ist die IP-Adresse für den Storage-Node oder den Gateway-Node im StorageGRID-Netzwerk.
- *Swift Port* Ist die Portnummer, die für Swift-API-Verbindungen auf dem Storage-Node oder Gateway-Node verwendet wird.

Die folgende Info-URL würde beispielsweise Informationen von einem Storage-Node mit der IP-Adresse von 10.99.106.103 anfordern und Port 18083 verwenden.

```
https://10.99.106.103:18083/info/
```

Die Antwort umfasst die Funktionen der Swift-Implementierung als JSON-Wörterbuch. Ein Client-Tool kann die JSON-Antwort analysieren, um die Funktionen der Implementierung zu bestimmen und sie als Einschränkungen für nachfolgende Storage-Vorgänge zu verwenden.

Die StorageGRID-Implementierung von Swift ermöglicht nicht authentifizierten Zugriff auf die Info-URL.

Auth-URL

Ein Client kann die Swift auth URL verwenden, um sich als Benutzer eines Mandantenkontos zu authentifizieren.

```
https://FQDN | Node IP:Swift Port/auth/v1.0/
```

Sie müssen die Mandanten-Konto-ID, den Benutzernamen und das Passwort als Parameter in angeben x-Auth-User Und X-Auth-Key Anforderungs-Header wie folgt:

```
X-Auth-User: Tenant_Account_ID:Username
```

```
X-Auth-Key: Password
```

In den Kopfzeilen der Anfrage:

- *Tenant_Account_ID* Ist die Account-ID, die StorageGRID beim Erstellen des Swift-Mandanten zugewiesen hat. Dies ist die gleiche Mandantenkonto-ID, die auf der Anmeldeseite des Mandanten-Managers verwendet wird.
- *Username* Ist der Name eines im Mandanten-Manager erstellten Benutzers. Dieser Benutzer muss einer Gruppe angehören, die über die Swift Administrator-Berechtigung verfügt. Der Root-Benutzer des Mandanten kann nicht für die Verwendung der Swift REST API konfiguriert werden.

Wenn Identity Federation für das Mandantenkonto aktiviert ist, geben Sie den Benutzernamen und das Passwort des föderierten Benutzers vom LDAP-Server an. Geben Sie alternativ den Domännennamen des LDAP-Benutzers an. Beispiel:

```
X-Auth-User: Tenant_Account_ID:Username@Domain_Name
```

- *Password* Ist das Passwort für den Mandantenbenutzer. Benutzerpasswörter werden im Mandanten-Manager erstellt und gemanagt.

Als Antwort auf eine erfolgreiche Authentifizierungsanforderung werden eine Storage-URL und ein auth-Token zurückgegeben:

```
X-Storage-Url: https://FQDN | Node_IP:Swift_Port/v1/Tenant_Account_ID
```

```
X-Auth-Token: token
```

```
X-Storage-Token: token
```

Das Token ist standardmäßig für 24 Stunden ab der Erzeugung gültig.

Token werden für ein bestimmtes Mandantenkonto generiert. Ein gültiges Token für ein Konto ermächtigt einen Benutzer nicht, auf ein anderes Konto zuzugreifen.

Storage-URL

Eine Client-Applikation kann Swift-REST-API-Aufrufe ausstellen, um unterstützte Konto-, Container- und Objektvorgänge mit einem Gateway-Node oder Storage-Node durchzuführen. Storage-Anforderungen werden an die in der Authentifizierungsantwort zurückgegebene Storage-URL adressiert. Die Anforderung muss auch die Kopfzeile von X-Auth-Token und den Wert enthalten, der von der auth-Anforderung zurückgegeben wurde.

```
https://FQDN | IP:Swift_Port/v1/Tenant_Account_ID
```

```
[/container] [/object]
```

```
X-Auth-Token: token
```

Einige Kopf für Speicherantwort, die Nutzungsstatistiken enthalten, geben möglicherweise keine genauen Zahlen für kürzlich geänderte Objekte wieder. Es kann einige Minuten dauern, bis genaue Zahlen in diesen Kopfzeilen angezeigt werden.

Die folgenden Antwortkopfzeilen für Konto- und Container-Vorgänge sind Beispiele für solche, die Nutzungsstatistiken enthalten:

- X-Account-Bytes-Used

- X-Account-Object-Count
- X-Container-Bytes-Used
- X-Container-Object-Count

Verwandte Informationen

["Wie Client-Verbindungen konfiguriert werden können"](#)

["Erstellen und Konfigurieren von Swift Mandantenkonten"](#)

["Konto-Operationen"](#)

["Container-Operationen"](#)

["Objekt-Operationen"](#)

Konto-Operationen

Die folgenden Swift-API-Vorgänge werden bei Accounts durchgeführt.

GET Konto

Dieser Vorgang ruft die Containerliste ab, die mit den Statistiken zur Konto- und Kontonutzung verknüpft ist.

Der folgende Parameter für die Anfrage ist erforderlich:

- Account

Die folgende Anfrageüberschrift ist erforderlich:

- X-Auth-Token

Die folgenden unterstützten Abfrageparameter sind optional:

- Delimiter
- End_marker
- Format
- Limit
- Marker
- Prefix

Eine erfolgreiche Ausführung gibt die folgenden Header mit einer „HTTP/1.1 204 No Content“-Antwort zurück, wenn das Konto gefunden wurde und keine Container oder die Containerliste leer ist; oder eine „HTTP/1.1 200 OK“-Antwort, wenn das Konto gefunden wurde und die Containerliste nicht leer ist:

- Accept-Ranges
- Content-Length
- Content-Type

- Date
- X-Account-Bytes-Used
- X-Account-Container-Count
- X-Account-Object-Count
- X-Timestamp
- X-Trans-Id

HEAD Konto

Mit dieser Operation werden Kontoinformationen und Statistiken von einem Swift-Konto abgerufen.

Der folgende Parameter für die Anfrage ist erforderlich:

- Account

Die folgende Anfrageüberschrift ist erforderlich:

- X-Auth-Token

Bei einer erfolgreichen Ausführung werden die folgenden Header mit einer „HTTP/1.1 204 No Content“-Antwort zurückgegeben:

- Accept-Ranges
- Content-Length
- Date
- X-Account-Bytes-Used
- X-Account-Container-Count
- X-Account-Object-Count
- X-Timestamp
- X-Trans-Id

Verwandte Informationen

["In den Audit-Protokollen werden Swift-Vorgänge nachverfolgt"](#)

Container-Operationen

StorageGRID unterstützt maximal 1,000 Container pro Swift Konto. Die folgenden Swift-API-Vorgänge werden auf Containern durchgeführt.

Container LÖSCHEN

Durch diesen Vorgang wird ein leerer Container aus einem Swift-Konto in einem StorageGRID-System entfernt.

Die folgenden Anfrageparameter sind erforderlich:

- Account
- Container

Die folgende Anfrageüberschrift ist erforderlich:

- X-Auth-Token

Eine erfolgreiche Ausführung gibt die folgenden Kopfzeilen mit einer HTTP/1.1 204 No Content-Antwort zurück:

- Content-Length
- Content-Type
- Date
- X-Trans-Id

GET Container

Dieser Vorgang ruft die dem Container zugeordnete Objektliste sowie die Containerstatistiken und Metadaten in einem StorageGRID System ab.

Die folgenden Anfrageparameter sind erforderlich:

- Account
- Container

Die folgende Anfrageüberschrift ist erforderlich:

- X-Auth-Token

Die folgenden unterstützten Abfrageparameter sind optional:

- Delimiter
- End_marker
- Format
- Limit
- Marker
- Path
- Prefix

Eine erfolgreiche Ausführung liefert die folgenden Header mit einer "HTTP/1.1 200 success" oder einer "HTTP/1.1 204 No Content"-Antwort:

- Accept-Ranges
- Content-Length
- Content-Type

- Date
- X-Container-Bytes-Used
- X-Container-Object-Count
- X-Timestamp
- X-Trans-Id

KOPF Behälter

Dieser Vorgang ruft Containerstatistiken und Metadaten aus einem StorageGRID System ab.

Die folgenden Anfrageparameter sind erforderlich:

- Account
- Container

Die folgende Anfrageüberschrift ist erforderlich:

- X-Auth-Token

Eine erfolgreiche Ausführung gibt die folgenden Kopfzeilen mit einer HTTP/1.1 204 No Content-Antwort zurück:

- Accept-Ranges
- Content-Length
- Date
- X-Container-Bytes-Used
- X-Container-Object-Count
- X-Timestamp
- X-Trans-Id

Legen Sie den Behälter

Durch diesen Vorgang wird ein Container für ein Konto in einem StorageGRID-System erstellt.

Die folgenden Anfrageparameter sind erforderlich:

- Account
- Container

Die folgende Anfrageüberschrift ist erforderlich:

- X-Auth-Token

Eine erfolgreiche Ausführung gibt die folgenden Header mit einer "HTTP/1.1 201 created" oder "HTTP/1.1 202 Accepted" (falls der Container bereits unter diesem Konto existiert) Antwort zurück:

- Content-Length
- Date
- X-Timestamp
- X-Trans-Id

Container-Name muss im StorageGRID-Namespace eindeutig sein. Wenn der Container unter einem anderen Konto vorhanden ist, wird der folgende Header zurückgegeben: „HTTP/1.1 409-Konflikt“.

Verwandte Informationen

["In den Audit-Protokollen werden Swift-Vorgänge nachverfolgt"](#)

Objekt-Operationen

Die folgenden Swift-API-Vorgänge werden an Objekten durchgeführt.

Delete Objekt

Durch diesen Vorgang werden der Inhalt und die Metadaten eines Objekts aus dem StorageGRID System gelöscht.

Die folgenden Anfrageparameter sind erforderlich:

- Account
- Container
- Object

Die folgende Anfrageüberschrift ist erforderlich:

- X-Auth-Token

Bei einer erfolgreichen Ausführung werden die folgenden Antwortheadern mit einem zurückgegeben
HTTP/1.1 204 No Content Antwort:

- Content-Length
- Content-Type
- Date
- X-Trans-Id

Bei der Verarbeitung einer LÖSCHOBJEKTANFORDERUNG versucht StorageGRID, alle Kopien des Objekts sofort von allen gespeicherten Speicherorten zu entfernen. Wenn erfolgreich, gibt StorageGRID sofort eine Antwort an den Client zurück. Falls nicht alle Kopien innerhalb von 30 Sekunden entfernt werden können (z. B. weil ein Standort vorübergehend nicht verfügbar ist), warteschlangen StorageGRID die Kopien zum Entfernen und zeigen dann den Erfolg des Clients an.

Weitere Informationen zum Löschen von Objekten finden Sie in den Anweisungen zum Verwalten von Objekten mit Information Lifecycle Management.

GET Objekt

Dieser Vorgang ruft den Objekteinhalt ab und ruft die Objektmetadaten von einem StorageGRID System ab.

Die folgenden Anfrageparameter sind erforderlich:

- Account
- Container
- Object

Die folgende Anfrageüberschrift ist erforderlich:

- X-Auth-Token

Die folgenden Anfragezeilen sind optional:

- Accept-Encoding
- If-Match
- If-Modified-Since
- If-None-Match
- If-Unmodified-Since
- Range

Bei einer erfolgreichen Ausführung werden die folgenden Kopfzeilen mit einem zurückgegeben HTTP/1.1 200 OK Antwort:

- Accept-Ranges
- Content-Disposition, Nur wenn zurückgegeben Content-Disposition Es wurden Metadaten festgelegt
- Content-Encoding, Nur wenn zurückgegeben Content-Encoding Es wurden Metadaten festgelegt
- Content-Length
- Content-Type
- Date
- ETag
- Last-Modified
- X-Timestamp
- X-Trans-Id

HEAD Objekt

Dieser Vorgang ruft Metadaten und Eigenschaften eines aufgenommenen Objekts von einem StorageGRID System ab.

Die folgenden Anfrageparameter sind erforderlich:

- Account
- Container
- Object

Die folgende Anfrageüberschrift ist erforderlich:

- X-Auth-Token

Eine erfolgreiche Ausführung gibt die folgenden Header mit einer HTTP/1.1 200 OK-Antwort zurück:

- Accept-Ranges
- Content-Disposition, Nur wenn zurückgegeben Content-Disposition Es wurden Metadaten festgelegt
- Content-Encoding, Nur wenn zurückgegeben Content-Encoding Es wurden Metadaten festgelegt
- Content-Length
- Content-Type
- Date
- ETag
- Last-Modified
- X-Timestamp
- X-Trans-Id

PUT Objekt

Durch diesen Vorgang wird ein neues Objekt mit Daten und Metadaten erstellt oder ein vorhandenes Objekt durch Daten und Metadaten in einem StorageGRID System ersetzt.

StorageGRID unterstützt Objekte mit einer Größe von bis zu 5 TB.



Widersprüchliche Clientanforderungen, wie z. B. zwei Clients, die in denselben Schlüssel schreiben, werden auf „latest-WINS“-Basis gelöst. Der Zeitpunkt für die Auswertung „latest-WINS“ basiert darauf, wann das StorageGRID System eine bestimmte Anfrage abschließt und nicht auf, wenn Swift-Clients einen Vorgang starten.

Die folgenden Anfrageparameter sind erforderlich:

- Account
- Container
- Object

Die folgende Anfrageüberschrift ist erforderlich:

- X-Auth-Token

Die folgenden Anfragezeilen sind optional:

- Content-Disposition
- Content-Encoding

Verwenden Sie keine Punkte Content-Encoding Wenn die ILM-Regel für ein Objekt Objekte nach der Größe filtert und synchrone Platzierung bei der Aufnahme verwendet wird (die ausgewogenen oder strengen Optionen für das Aufnahmeverhalten).

- Transfer-Encoding

Verwenden Sie keine komprimierten oder chunked Transfer-Encoding Wenn die ILM-Regel für ein Objekt Objekte nach der Größe filtert und synchrone Platzierung bei der Aufnahme verwendet wird (die ausgewogenen oder strengen Optionen für das Aufnahmeverhalten).

- Content-Length

Wenn eine ILM-Regel Objekte nach Größe filtert und bei der Aufnahme synchrone Platzierung verwendet, müssen Sie angeben Content-Length.



Wenn Sie diese Richtlinien für nicht befolgen Content-Encoding, Transfer-Encoding, und Content-Length, StorageGRID muss das Objekt speichern, bevor es die Objektgröße bestimmen kann und die ILM-Regel anwenden kann. Das heißt, StorageGRID muss standardmäßig vorläufige Kopien eines Objekts bei der Aufnahme erstellen. Das heißt, StorageGRID muss die Dual-Commit-Option für das Ingest-Verhalten verwenden.

Weitere Informationen zur synchronen Platzierung und zu ILM-Regeln finden Sie in den Anweisungen zum Managen von Objekten mit Information Lifecycle Management.

- Content-Type
- ETag
- X-Object-Meta-<name\> (Objektbezogene Metadaten)

Wenn Sie die Option **Benutzerdefinierte Erstellungszeit** als Referenzzeit für eine ILM-Regel verwenden möchten, müssen Sie den Wert in einem benutzerdefinierten Header mit dem Namen speichern X-Object-Meta-Creation-Time. Beispiel:

```
X-Object-Meta-Creation-Time: 1443399726
```

Dieses Feld wird seit dem 1. Januar 1970 als Sekunden ausgewertet.

- X-Storage-Class: reduced_redundancy

Diese Kopfzeile wirkt sich darauf aus, wie viele Objektkopien StorageGRID erstellt werden, wenn die ILM-Regel, die mit einem aufgenommenen Objekt übereinstimmt, ein Aufnahmeverhalten der Dual-Commit oder Balance angibt.

- **Dual Commit:** Wenn die ILM-Regel die Dual Commit-Option für das Aufnahmeverhalten angibt, erstellt StorageGRID bei Aufnahme des Objekts eine einzelne Interimskopie (Single Commit).
- **Ausgewogen:** Wenn die ILM-Regel die ausgewogene Option angibt, erstellt StorageGRID nur eine einzige Zwischenkopie, wenn das System nicht sofort alle in der Regel festgelegten Kopien erstellen

kann. Wenn StorageGRID eine synchrone Platzierung durchführen kann, hat diese Kopfzeile keine Auswirkung.

Der `reduced_redundancy` Kopfzeile eignet sich am besten, wenn die ILM-Regel, die dem Objekt entspricht, eine einzige replizierte Kopie erstellt. In diesem Fall verwenden `reduced_redundancy` Eine zusätzliche Objektkopie kann bei jedem Aufnahmevergang nicht mehr erstellt und gelöscht werden.

Verwenden der `reduced_redundancy` Header wird unter anderen Umständen nicht empfohlen, da dies das Risiko für den Verlust von Objektdaten während der Aufnahme erhöht. Beispielsweise können Sie Daten verlieren, wenn die einzelne Kopie zunächst auf einem Storage Node gespeichert wird, der ausfällt, bevor eine ILM-Evaluierung erfolgen kann.



Da nur eine Kopie zu einem beliebigen Zeitpunkt repliziert werden kann, sind Daten einem ständigen Verlust ausgesetzt. Wenn nur eine replizierte Kopie eines Objekts vorhanden ist, geht dieses Objekt verloren, wenn ein Speicherknoten ausfällt oder einen beträchtlichen Fehler hat. Während Wartungsarbeiten wie Upgrades verlieren Sie auch vorübergehend den Zugriff auf das Objekt.

Beachten Sie, dass Sie angeben `reduced_redundancy` Wirkt sich nur darauf aus, wie viele Kopien erstellt werden, wenn ein Objekt zum ersten Mal aufgenommen wird. Er hat keine Auswirkungen auf die Anzahl der Kopien des Objekts, wenn das Objekt von der aktiven ILM-Richtlinie geprüft wird, und führt nicht dazu, dass Daten auf einer niedrigeren Redundanzebene im StorageGRID System gespeichert werden.

Eine erfolgreiche Ausführung gibt die folgenden Header mit einer "HTTP/1.1 201 created"-Antwort zurück:

- `Content-Length`
- `Content-Type`
- `Date`
- `ETag`
- `Last-Modified`
- `X-Trans-Id`

Verwandte Informationen

["Objektmanagement mit ILM"](#)

["In den Audit-Protokollen werden Swift-Vorgänge nachverfolgt"](#)

OPTIONEN anfordern

Die `OPTIONEN` Request überprüft die Verfügbarkeit eines einzelnen Swift Service. Die `OPTIONSANFORDERUNG` wird vom in der URL angegebenen Speicherknoten oder Gateway-Node verarbeitet.

OPTIONEN

Client-Anwendungen können zum Beispiel eine `OPTIONSANFORDERUNG` an den Swift-Port auf einem Storage Node stellen, ohne Swift-Authentifizierungsdaten bereitzustellen, um zu ermitteln, ob der Storage-

Node verfügbar ist. Sie können diese Anforderung zum Monitoring verwenden oder um externen Lastausgleich zu ermöglichen, wenn ein Storage-Node ausfällt.

Bei Verwendung mit der Info-URL oder der Speicher-URL gibt die OPTIONSMETHODE eine Liste der unterstützten Verben für die angegebene URL zurück (z. B. KOPF, GET, OPTIONEN und PUT). DIE OPTIONSMETHODE kann nicht mit der auth URL verwendet werden.

Der folgende Parameter für die Anfrage ist erforderlich:

- Account

Die folgenden Anfrageparameter sind optional:

- Container
- Object

Bei einer erfolgreichen Ausführung werden die folgenden Header mit einer „HTTP/1.1 204 No Content“-Antwort zurückgegeben. Für die ANFORDERUNG VON OPTIONEN an die Speicher-URL ist nicht erforderlich, dass das Ziel vorhanden ist.

- Allow (Eine Liste der unterstützten Verben für die angegebene URL, z. B. „KOPF“, „ABRUFEN“, „OPTIONEN“, Und PUT)
- Content-Length
- Content-Type
- Date
- X-Trans-Id

Verwandte Informationen

["Unterstützte Swift-API-Endpunkte"](#)

Fehlerantworten bei Swift-API-Operationen

Das Verständnis möglicher Fehlerantworten kann Ihnen bei der Fehlerbehebung helfen.

Wenn während eines Vorgangs Fehler auftreten, werden möglicherweise die folgenden HTTP-Statuscodes zurückgegeben:

Swift-Fehlername	HTTP-Status
AccountNameTooLong, ContainerNameTooLong, HeaderTooBig, InvalidContainerName, InvalidRequest, InvalidURI, MetadataNameTooLong, MetadaValueTooBig, MissingSecurityHeader, ObjectNameTooLong, TooManyContainers, TooManyMetadataItems, TotalMetadaTooLarge	400 Fehlerhafte Anfrage
AccessDenied	403 Verbotene
ContainerNotEmpty, ContainerAlreadyExists	409 Konflikt

Swift-Fehlername	HTTP-Status
Interner Fehler	500 Fehler Des Internen Servers
InvalidRange	416 Angeforderter Bereich Nicht Zu Unterprüfbar
MethodenAlled	405 Methode Nicht Zulässig
MissingContentLänge	411 Länge Erforderlich
Nicht gefunden	404 Nicht Gefunden
NotImplemsted	501 Nicht Implementiert
Vorbedingungen nicht möglich	412 Voraussetzung Fehlgeschlagen
ResourceNotFound	404 Nicht Gefunden
Nicht Autorisiert	401 Nicht Autorisiert
Nicht verarbeitbarEntity	422 Nicht Verarbeitbare Einheit

StorageGRID Swift REST-API-Operationen

Speziell für das StorageGRID System wurden Vorgänge zur Swift REST API hinzugefügt.

ABRUFEN der Container-Konsistenzanforderung

Die Konsistenzstufe sorgt für einen Kompromiss zwischen der Verfügbarkeit der Objekte und der Konsistenz dieser Objekte über verschiedene Speicherknoten und Standorte hinweg. Die GET Container-Konsistenzanforderung ermöglicht es Ihnen, die auf einen bestimmten Container angewendete Konsistenzstufe zu bestimmen.

Anfrage

HTTP-Header anfordern	Beschreibung
X-Auth-Token	Gibt das Swift-Authentifizierungs-Token für das Konto an, das für die Anforderung verwendet werden soll.
x-ntap-sg-consistency	Gibt den Anforderungstyp an, wobei <code>true</code> = GET Containerkonsistenz, und <code>false</code> = get Container.
Host	Der Hostname, auf den die Anforderung gerichtet ist.

Anforderungsbeispiel

```
GET /v1/28544923908243208806/Swift container
X-Auth-Token: SGRD_3a877009a2d24cb1801587bfa9050f29
x-ntap-sg-consistency: true
Host: test.com
```

Antwort

HTTP-Kopfzeile für Antwort	Beschreibung
Date	Datum und Uhrzeit der Antwort.
Connection	Ob die Verbindung zum Server offen oder geschlossen ist.
X-Trans-Id	Die eindeutige Transaktions-ID für die Anforderung.
Content-Length	Die Länge des Reaktionskörpers.

HTTP-Kopfzeile für Antwort	Beschreibung
x-ntap-sg-consistency	<p>Die auf den Container angewendete Konsistenzkontrollebene. Folgende Werte werden unterstützt:</p> <ul style="list-style-type: none"> • Alle: 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: Sorgt für die Konsistenz von Read-after-write für neue Objekte und eventuelle Konsistenz von Objekt-Updates. Hochverfügbarkeit und garantierte Datensicherung <p>Hinweis: Wenn Ihre Anwendung HEAD Requests für Objekte verwendet, die nicht existieren, erhalten Sie möglicherweise eine hohe Anzahl von 500 internen Serverfehlern, wenn ein oder mehrere Speicherknoten nicht verfügbar sind. Um diese Fehler zu vermeiden, verwenden Sie die Ebene „Available“.</p> <ul style="list-style-type: none"> • Verfügbar (eventuelle Konsistenz für KOPFOPERATIONEN): Verhält sich wie das „read-after-New-write“ Konsistenzniveau, bietet aber nur eventuelle Konsistenz für DEN KOPFBETRIEB. Bietet höhere Verfügbarkeit FÜR HEAD-Operationen als „read-after-New-write“, wenn Storage Nodes nicht verfügbar sind.

Antwortbeispiel

```

HTTP/1.1 204 No Content
Date: Sat, 29 Nov 2015 01:02:18 GMT
Connection: CLOSE
X-Trans-Id: 1936575373
Content-Length: 0
x-ntap-sg-consistency: strong-site

```

Verwandte Informationen

["Verwenden Sie ein Mandantenkonto"](#)

PUT Container-Konsistenzanforderung

Die PUT Container-Konsistenzanforderung ermöglicht es Ihnen, die Konsistenzstufe für die Operationen anzugeben, die auf einem Container ausgeführt werden. Standardmäßig werden neue Container mithilfe der Konsistenzstufe „read-after-New-write“ erstellt.

Anfrage

HTTP-Header anfordern	Beschreibung
X-Auth-Token	Swift Authentifizierungs-Token für das Konto zur Verwendung für die Anforderung.
x-ntap-sg-consistency	<p>Die Konsistenzkontrollebene gilt für Container-Operationen. Folgende Werte werden unterstützt:</p> <ul style="list-style-type: none">• Alle: 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: Sorgt für die Konsistenz von Read-after-write für neue Objekte und eventuelle Konsistenz von Objekt-Updates. Hochverfügbarkeit und garantierte Datensicherung <p>Hinweis: Wenn Ihre Anwendung HEAD Requests für Objekte verwendet, die nicht existieren, erhalten Sie möglicherweise eine hohe Anzahl von 500 internen Serverfehlern, wenn ein oder mehrere Speicherknoten nicht verfügbar sind. Um diese Fehler zu vermeiden, verwenden Sie die Ebene „Available“.</p> <ul style="list-style-type: none">• Verfügbar (eventuelle Konsistenz für KOPFOPERATIONEN): Verhält sich wie das „read-after-New-write“ Konsistenzniveau, bietet aber nur eventuelle Konsistenz für DEN KOPFBETRIEB. Bietet höhere Verfügbarkeit FÜR HEAD-Operationen als „read-after-New-write“, wenn Storage Nodes nicht verfügbar sind.
Host	Der Hostname, auf den die Anforderung gerichtet ist.

Konsistenzkontrollen und ILM-Regeln interagieren, um die Datensicherung 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:

- **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.
- **Dual Commit:** StorageGRID erstellt sofort Zwischenkopien des Objekts und gibt den Erfolg an den Kunden zurück. Kopien, die in der ILM-Regel angegeben sind, werden nach Möglichkeit erstellt.



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.

Anforderungsbeispiel

```
PUT /v1/28544923908243208806/_Swift_container_  
X-Auth-Token: SGRD_3a877009a2d24cb1801587bfa9050f29  
x-ntap-sg-consistency: strong-site  
Host: test.com
```

Antwort

HTTP-Kopfzeile für Antwort	Beschreibung
Date	Datum und Uhrzeit der Antwort.
Connection	Ob die Verbindung zum Server offen oder geschlossen ist.
X-Trans-Id	Die eindeutige Transaktions-ID für die Anforderung.
Content-Length	Die Länge des Reaktionskörpers.

Antwortbeispiel

```
HTTP/1.1 204 No Content  
Date: Sat, 29 Nov 2015 01:02:18 GMT  
Connection: CLOSE  
X-Trans-Id: 1936575373  
Content-Length: 0
```

Verwandte Informationen

["Verwenden Sie ein Mandantenkonto"](#)

Sicherheit wird für DIE REST API konfiguriert

Sie sollten die für DIE REST API implementierten Sicherheitsmaßnahmen überprüfen und verstehen, wie Sie Ihr System sichern können.

So bietet StorageGRID Sicherheit für DIE REST-API

Sie sollten verstehen, wie das StorageGRID System die Sicherheit, Authentifizierung und Autorisierung für DIE REST-API implementiert.

StorageGRID setzt die folgenden Sicherheitsmaßnahmen ein.

- Die Client-Kommunikation mit dem Load Balancer-Service erfolgt über HTTPS, wenn HTTPS für den Load Balancer-Endpunkt konfiguriert ist.

Wenn Sie einen Endpunkt für den Load Balancer konfigurieren, kann HTTP optional aktiviert werden. Möglicherweise möchten Sie beispielsweise HTTP für Tests oder andere Zwecke verwenden, die nicht aus der Produktion stammen. Weitere Informationen finden Sie in den Anweisungen zum Verwalten von StorageGRID.

- Standardmäßig verwendet StorageGRID HTTPS für die Client-Kommunikation mit Speicherknoten und den CLB-Service auf Gateway-Knoten.

HTTP kann optional für diese Verbindungen aktiviert werden. Möglicherweise möchten Sie beispielsweise HTTP für Tests oder andere Zwecke verwenden, die nicht aus der Produktion stammen. Weitere Informationen finden Sie in den Anweisungen zum Verwalten von StorageGRID.



Der CLB-Service ist veraltet.

- Die Kommunikation zwischen StorageGRID und dem Client wird über TLS verschlüsselt.
- Die Kommunikation zwischen dem Load Balancer-Service und den Speicherknoten innerhalb des Grid wird verschlüsselt, ob der Load Balancer-Endpunkt für die Annahme von HTTP- oder HTTPS-Verbindungen konfiguriert ist.
- Clients müssen HTTP-Authentifizierungskopfzeilen an StorageGRID bereitstellen, um REST-API-Vorgänge durchzuführen.

Sicherheitszertifikate und Clientanwendungen

Clients können eine Verbindung zum Lastverteilungsservice auf Gateway-Knoten oder Admin-Nodes, direkt zu Storage-Nodes oder zum CLB-Dienst auf Gateway-Nodes herstellen.

Clientanwendungen können in jedem Fall TLS-Verbindungen herstellen, indem sie entweder ein vom Grid-Administrator hochgeladenes benutzerdefiniertes Serverzertifikat oder ein vom StorageGRID-System generiertes Zertifikat verwenden:

- Wenn Client-Anwendungen eine Verbindung zum Load Balancer-Service herstellen, verwenden sie dazu das Zertifikat, das für den spezifischen Load Balancer-Endpunkt konfiguriert wurde, der für die Verbindung verwendet wurde. Jeder Endpunkt verfügt über ein eigenes Zertifikat, entweder ein vom Grid-Administrator hochgeladenes benutzerdefiniertes Serverzertifikat oder ein Zertifikat, das der Grid-Administrator bei der Konfiguration des Endpunkts in StorageGRID generiert hat.
- Wenn Client-Anwendungen eine direkte Verbindung zu einem Speicherknoten oder zum CLB-Dienst auf Gateway-Knoten herstellen, verwenden sie entweder die vom System generierten Serverzertifikate, die bei der Installation des StorageGRID-Systems (die von der Systemzertifikatbehörde signiert sind) für Speicherknoten generiert wurden. Oder ein einzelnes benutzerdefiniertes Serverzertifikat, das von einem Grid-Administrator für das Grid bereitgestellt wird.

Die Clients sollten so konfiguriert werden, dass sie der Zertifizierungsstelle vertrauen, die unabhängig davon, welches Zertifikat sie zum Erstellen von TLS-Verbindungen verwenden, unterzeichnet hat.

Informationen StorageGRID zum Konfigurieren von Load Balancer-Endpunkten finden Sie in den Anweisungen zum Hinzufügen eines einzelnen benutzerdefinierten Serverzertifikats für TLS-Verbindungen direkt zu Storage-Nodes oder zum CLB-Dienst auf Gateway-Nodes.

Zusammenfassung

Die folgende Tabelle zeigt, wie Sicherheitsprobleme in den S3 und Swift REST-APIs implementiert werden:

Sicherheitsproblem	Implementierung für REST-API
Verbindungssicherheit	TLS
Serverauthentifizierung	X.509-Serverzertifikat, das von der System-CA oder vom Administrator zur Verfügung gestellten benutzerdefinierten Serverzertifikat unterzeichnet wurde
Client-Authentifizierung	<ul style="list-style-type: none"> • S3: S3-Konto (Zugriffsschlüssel-ID und geheimer Zugriffsschlüssel) • Swift: Swift-Konto (Benutzername und Passwort)
Client-Autorisierung	<ul style="list-style-type: none"> • S3: Bucket-Eigentümerschaft und alle anwendbaren Richtlinien für die Zugriffssteuerung • Swift: Administratorrollenzugriff

Verwandte Informationen

["StorageGRID verwalten"](#)

Unterstützte Hashing- und Verschlüsselungsalgorithmen für TLS-Bibliotheken

Das StorageGRID System unterstützt eine begrenzte Anzahl von Chiffren-Suites, die Client-Anwendungen beim Einrichten einer TLS-Sitzung (Transport Layer Security) verwenden können.

Unterstützte Versionen von TLS

StorageGRID unterstützt TLS 1.2 und TLS 1.3.



SSLv3 und TLS 1.1 (oder frühere Versionen) werden nicht mehr unterstützt.

Unterstützte Chiffren-Suiten

TLS-Version	IANA Name der Chiffre Suite
1.2	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
1.3	TLS_AES_256_GCM_SHA384
TLS_CHACHA20_POLY1305_SHA256	TLS_AES_128_GCM_SHA256

Veraltete Chiffre-Suiten

Die folgenden Chiffren Suites sind veraltet. Die Unterstützung für diese Chiffren wird in einer zukünftigen Version entfernt.

IANA-Name
TLS_RSA_WITH_AES_128_GCM_SHA256
TLS_RSA_WITH_AES_256_GCM_SHA384

Verwandte Informationen

["Wie Client-Verbindungen konfiguriert werden können"](#)

Monitoring und Auditing von Vorgängen

Kunden können Workloads und die Effizienz für Client-Vorgänge überwachen, indem sie Transaktionstrends für das gesamte Grid oder bestimmte Nodes anzeigen. Sie können Audit-Meldungen zur Überwachung von Client-Vorgängen und -Transaktionen verwenden.

Monitoring der Objekteinspeisung und -Abrufzeiten

Die Überwachung von Objektaufnahmeraten und -Abrufzeiten sowie von Metriken für Objektanzahl, -Abfragen und -Verifizierung. Sie können die Anzahl der erfolgreichen und fehlgeschlagenen Versuche von Client-Applikationen anzeigen, Objekte in StorageGRID zu lesen, zu schreiben und zu ändern.

Schritte

1. Melden Sie sich über einen unterstützten Browser beim Grid Manager an.
2. Suchen Sie im Dashboard den Abschnitt Protokollvorgänge.

In diesem Abschnitt wird die Anzahl der Client-Vorgänge zusammengefasst, die vom StorageGRID System durchgeführt werden. Die Protokollraten werden über die letzten zwei Minuten Durchschnitt.

3. Wählen Sie **Knoten**.
4. Klicken Sie auf der Startseite Knoten (Bereitstellungsebene) auf die Registerkarte **Load Balancer**.

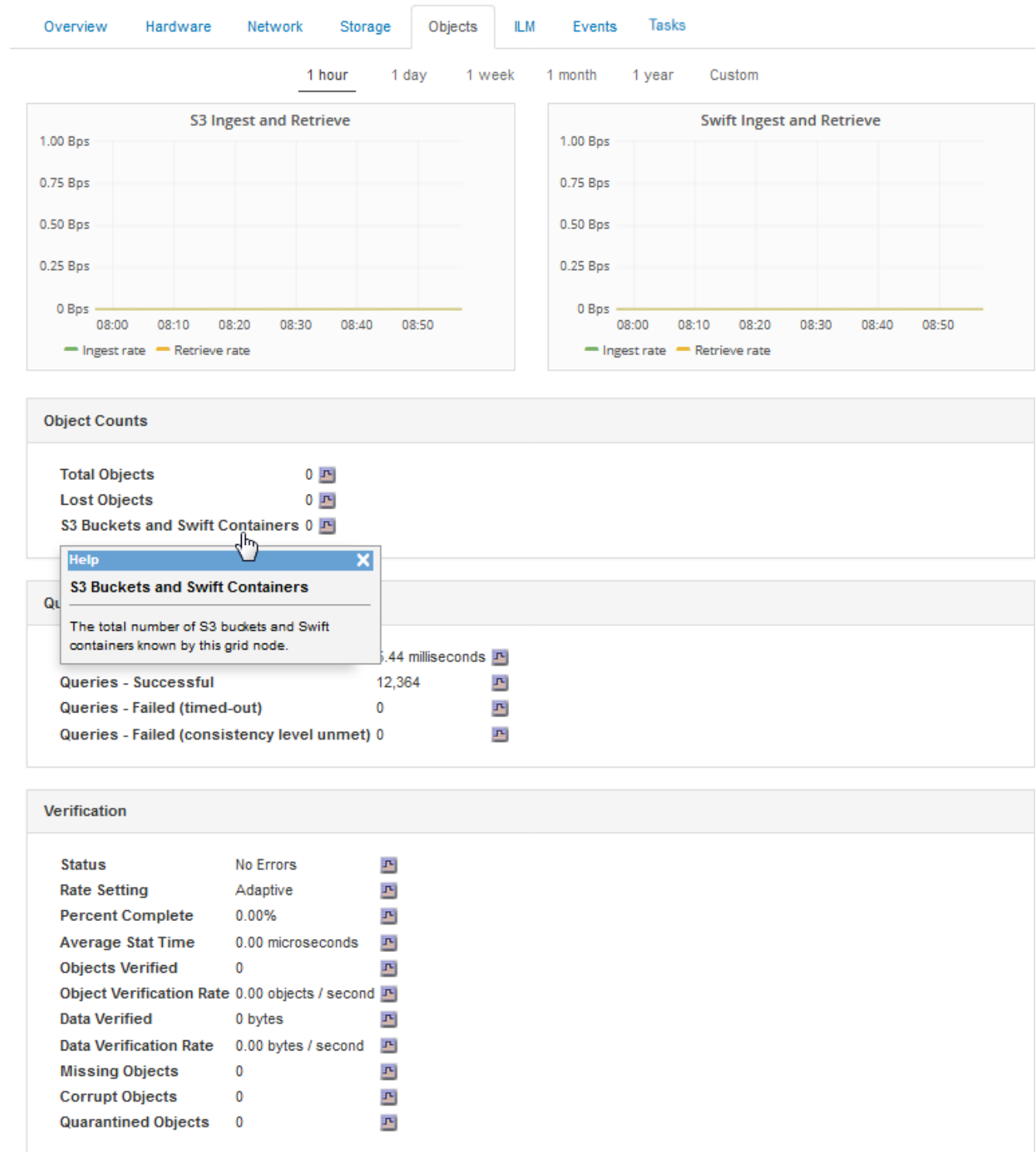
Die Diagramme zeigen Trends für den gesamten Client-Datenverkehr an Load Balancer-Endpunkte im Raster. Sie können ein Zeitintervall in Stunden, Tagen, Wochen, Monaten oder Jahren auswählen. Oder Sie können ein benutzerdefiniertes Intervall anwenden.

5. Klicken Sie auf der Startseite Knoten (Bereitstellungsebene) auf die Registerkarte **Objekte**.

Das Diagramm zeigt die Aufnahme- und Abrufzeiten Ihres gesamten StorageGRID Systems in Byte pro Sekunde sowie insgesamt Bytes. Sie können ein Zeitintervall in Stunden, Tagen, Wochen, Monaten oder Jahren auswählen. Oder Sie können ein benutzerdefiniertes Intervall anwenden.

6. Um Informationen zu einem bestimmten Speicherknoten anzuzeigen, wählen Sie den Knoten aus der Liste auf der linken Seite aus, und klicken Sie auf die Registerkarte **Objekte**.

Das Diagramm zeigt die Aufnahme- und Abrufzeiten des Objekts für diesen Speicherknoten. Die Registerkarte enthält außerdem Kennzahlen für Objektanzahl, Abfragen und Verifizierung. Sie können auf die Beschriftungen klicken, um die Definitionen dieser Metriken anzuzeigen.



7. Wenn Sie noch mehr Details wünschen:

- Wählen Sie **Support > Tools > Grid Topology** Aus.
- Wählen Sie **site > Übersicht > Haupt**.

Im Abschnitt API-Vorgänge werden zusammenfassende Informationen für das gesamte Raster angezeigt.

c. Wählen Sie **Storage Node > LDR > Client-Anwendung > Übersicht > Main** aus

Im Abschnitt „Vorgänge“ werden zusammenfassende Informationen für den ausgewählten Speicherknoten angezeigt.

Aufrufen und Prüfen von Prüfprotokollen

Audit-Meldungen werden von StorageGRID-Diensten generiert und in Text-Log-Dateien gespeichert. API-spezifische Audit-Meldungen in den Audit-Protokollen stellen kritische Daten zum Monitoring von Sicherheit, Betrieb und Performance bereit, die Ihnen bei der Bewertung des Systemzustands helfen können.

Was Sie benötigen

- Sie müssen über spezifische Zugriffsberechtigungen verfügen.
- Sie müssen die `Passwords.txt` Datei:
- Sie müssen die IP-Adresse eines Admin-Knotens kennen.

Über diese Aufgabe

Der Name der aktiven Audit-Log-Datei `audit.log`, Und es wird auf Admin-Knoten gespeichert.

Einmal am Tag wird die aktive `audit.log`-Datei gespeichert und eine neue `audit.log`-Datei gestartet. Der Name der gespeicherten Datei gibt an, wann sie gespeichert wurde, im Format `yyyy-mm-dd.txt`.

Nach einem Tag wird die gespeicherte Datei komprimiert und im Format umbenannt `yyyy-mm-dd.txt.gz`, Die das ursprüngliche Datum bewahrt.

Dieses Beispiel zeigt die aktive `audit.log`-Datei, die Datei des Vortags (`2018-04-15.txt`) und die komprimierte Datei für den Vortag (`2018-04-14.txt.gz`).

```
audit.log
2018-04-15.txt
2018-04-14.txt.gz
```

Schritte

1. Melden Sie sich bei einem Admin-Knoten an:
 - a. Geben Sie den folgenden Befehl ein: `ssh admin@primary_Admin_Node_IP`
 - b. Geben Sie das im aufgeführte Passwort ein `Passwords.txt` Datei:
2. Gehen Sie zu dem Verzeichnis, das die Audit-Log-Dateien enthält: `cd /var/local/audit/export`
3. Sehen Sie sich die aktuelle oder gespeicherte Audit-Protokolldatei nach Bedarf an.

Verwandte Informationen

["Prüfung von Audit-Protokollen"](#)

In den Audit-Protokollen werden Swift-Vorgänge nachverfolgt

Alle erfolgreichen Vorgänge zum LÖSCHEN, ABRUFEN, NACHFÜHREN, POSTEN und PUT werden im StorageGRID Audit-Protokoll verfolgt. Fehler werden weder protokolliert, noch sind Info-, Auth- oder OPTIONSANFORDERUNGEN.

Weitere Informationen zu den für die folgenden Swift-Vorgänge erfassten Informationen finden Sie unter *Audit-Meldungen*.

Konto-Operationen

- GET Konto
- HEAD Konto

Container-Operationen

- Container LÖSCHEN
- GET Container
- KOPF Behälter
- Legen Sie den Behälter

Objekt-Operationen

- Delete Objekt
- GET Objekt
- HEAD Objekt
- PUT Objekt

Verwandte Informationen

["Prüfung von Audit-Protokollen"](#)

["Konto-Operationen"](#)

["Container-Operationen"](#)

["Objekt-Operationen"](#)

Copyright-Informationen

Copyright © 2025 NetApp. Alle Rechte vorbehalten. Gedruckt in den USA. Dieses urheberrechtlich geschützte Dokument darf ohne die vorherige schriftliche Genehmigung des Urheberrechtsinhabers 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, BEISPIELHAFTE 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.