使用Swift REST API (已弃用)

StorageGRID

NetApp
November 04, 2025

This PDF was generated from https://docs.netapp.com/zh-cn/storagegrid-118/swift/history-of-swift-api-support-in-storagegrid.html on November 04, 2025. Always check docs.netapp.com for the latest.

目录

使用Swift REST API (已弃用) 1
- 使用Swift REST API：概述 1
  - StorageGRID 中的 Swift API 支持历史记录 1
  - StorageGRID 如何实施 Swift REST API 2
  - 实施 Swift REST API 的建议 2
- 测试Swift REST API配置 3
- Swift REST API 支持的操作 5
  - StorageGRID 中支持的操作 5
  - 所有操作的通用响应标头 5
  - 支持的 Swift API 端点 5
  - 帐户操作 7
  - 容器操作 9
  - 对象操作 11
  - 选项请求 15
  - 对 Swift API 操作的错误响应 16
- StorageGRID Swift REST API 操作 17
  - 获取容器一致性请求 17
  - 提交容器一致性请求 18
- 审核日志中跟踪的 Swift 操作 20
  - 帐户操作 20
  - 容器操作 20
  - 对象操作 21

使用Swift REST API (已弃用)

使用Swift REST API：概述

客户端应用程序可以使用 OpenStack Swift API 与 StorageGRID 系统相连接。

对Swift客户端应用程序的支持已弃用、将在未来版本中删除。

StorageGRID 支持以下特定版本的 Swift 和 HTTP 。

项目	**version**
Swift 规格	截至 2015 年 11 月的 OpenStack Swift 对象存储 API v1
HTTP	1.1 有关 HTTP 的详细信息，请参见 HTTP/1.1 （ RFC 7230-35 ）。 • 注 * ： StorageGRID 不支持 HTTP/1.1 管道化。

相关信息

"OpenStack ： 对象存储 API"

StorageGRID 中的 Swift API 支持历史记录

您应了解 StorageGRID 系统对 Swift REST API 的支持发生了哪些变化。

版本。	注释
11.8.	
11.7	对Swift客户端应用程序的支持已弃用、将在未来版本中删除。
11.6.	少量编辑更改。
11.5	消除了较弱的一致性。此时将改用可用的一致性。
11.4	增加了对TLS 1.3的支持。增加了ILM与一致性之间相互关系的问题描述。
11.3	更新了 PUT 对象操作，以描述在载入时使用同步放置的 ILM 规则的影响（适用于载入行为的平衡而严格的选项）。添加了使用负载平衡器端点或高可用性组的客户端连接的问题描述 。不再支持 TLS 1.1 密码。
11.2.	对文档进行了少量编辑更改。

版本。	注释
11.1	增加了对使用 HTTP 与网格节点建立 Swift 客户端连接的支持。已更新一致性值的定义。
11.0	增加了对每个租户帐户 1 ， 000 个容器的支持。
10.3	对文档进行管理更新和更正。删除了用于配置自定义服务器证书的部分。
10.2	StorageGRID 系统对 Swift API 的初始支持。当前支持的版本为 OpenStack Swift 对象存储 API v1 。

StorageGRID 如何实施 Swift REST API

客户端应用程序可以使用 Swift REST API 调用连接到存储节点和网关节点以创建容器以及存储和检索对象。这样，为 OpenStack Swift 开发的面向服务的应用程序便可与 StorageGRID 系统提供的内部对象存储相连接。

Swift 对象管理

在StorageGRID系统中加入Swift对象后、这些对象将通过活动ILM策略中的信息生命周期管理(ILM)规则进行管理。 "ILM 规则" 和 "ILM策略" 确定StorageGRID 如何创建和分发对象数据的副本、以及如何在一段时间内管理这些副本。例如， ILM 规则可能适用于特定 Swift 容器中的对象，并可能指定在一定年数内将多个对象副本保存到多个数据中心。

如果您需要了解网格的ILM规则和策略如何影响Swift租户帐户中的对象、请联系您的NetApp专业服务顾问或StorageGRID 管理员。

客户端请求冲突

冲突的客户端请求（例如，两个客户端写入同一密钥）将以 " 最新成功 " 为基础进行解决。" 最新赢单 " 评估的时间取决于 StorageGRID 系统何时完成给定请求，而不是 Swift 客户端何时开始操作。

一致性保证和控制

默认情况下， StorageGRID 为新创建的对象提供读写一致性，并为对象更新和机头操作提供最终一致性。任意 "获取" 成功完成后执行 "PUT" 将能够读取新写入的数据。对现有对象的覆盖，元数据更新和删除最终保持一致。覆盖通常需要几秒钟或几分钟才能传播，但可能需要长达 15 天的时间。

此外，您还可以通过 StorageGRID 控制每个容器的一致性。一致性值可根据应用程序的要求、在不同存储节点和站点之间的对象可用性与这些对象的一致性之间实现平衡。

实施 Swift REST API 的建议

在实施用于 StorageGRID 的 Swift REST API 时，应遵循以下建议。

针对不存在的对象的建议

如果应用程序定期检查某个对象是否位于您不希望该对象实际存在的路径上、则应使用"可用"一致性。例如、如果应用程序在对某个位置执行Put操作之前对该位置执行机头操作、则应使用"可用"一致性。

否则，如果 head 操作未找到对象，则在一个或多个存储节点不可用时，可能会收到大量 500 个内部服务器错误。

您可以使用为每个容器设置"可用"一致性 "提交容器一致性请求"。您可以查看使用为每个容器设置"可用"一致性 "获取容器一致性请求"。

对象名称建议

对于在 StorageGRID 11.4 或更高版本中创建的容器，不再需要限制对象名称以满足性能最佳实践。例如，现在可以对对象名称的前四个字符使用随机值。

对于在 StorageGRID 11.4 之前的版本中创建的容器，请继续对对象名称遵循以下建议：

- 不应使用随机值作为对象名称的前四个字符。这与 AWS 以前针对名称前缀的建议不同。而应使用非随机、非唯一前缀、例如 `image`。
- 如果您按照以前的 AWS 建议在名称前缀中使用随机和唯一字符，则应在对象名称前添加目录名称。也就是说，请使用以下格式：

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

  而不是以下格式：

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

"范围读取"建议

如果 "用于压缩存储对象的全局选项" 已启用、则Swift客户端应用程序应避免执行指定要返回的字节数范围的GET对象操作。这些"范围读取"操作效率低下、因为StorageGRID必须有效地解压缩对象才能访问请求的字节。从非常大的对象请求少量字节的 GET 对象操作效率尤其低下；例如，从 50 GB 压缩对象读取 10 MB 范围的操作效率非常低。

如果从压缩对象读取范围，则客户端请求可能会超时。

> 如果需要压缩对象，并且客户端应用程序必须使用范围读取，请增加应用程序的读取超时时间。

测试Swift REST API配置

您可以使用Swift命令行界面测试与StorageGRID系统的连接、并验证是否可以读取和写入对象。

开始之前

- 您已下载并安装Swift命令行客户端： "SwiftStack： Python-swiftclient"
- 您也可以选择 "已创建负载平衡器端点"。否则、您知道要连接到的存储节点的IP地址以及要使用的端口号。请参见 "客户端连接的IP地址和端口"。
- 您已拥有 "已创建Swift租户帐户"。
- 您已登录到租户帐户并至少创建了一个组和用户。请参见 "为 Swift 租户创建组"。

> Swift租户用户必须具有管理员组权限才能向Swift REST API进行身份验证。

关于此任务

如果尚未配置安全性、则必须添加 `--insecure` 标记这些命令中的每个命令。

步骤

1. 查询 StorageGRID Swift 部署的信息 URL：

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

这足以测试您的 Swift 部署是否正常运行。要通过存储对象进一步测试帐户配置，请继续执行其他步骤。

2. 将对象放入容器中：

```
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. 获取用于验证对象的容器：

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

4. 删除对象：

```
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. 删除容器：

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

Swift REST API 支持的操作

StorageGRID 系统在 OpenStack Swift API 中支持大多数操作。在将 Swift REST API 客户端与 StorageGRID 集成之前，请查看帐户，容器和对象操作的实施详细信息。

StorageGRID 中支持的操作

支持以下 Swift API 操作：

- "帐户操作"
- "容器操作"
- "对象操作"

所有操作的通用响应标头

StorageGRID 系统可为 OpenStack Swift 对象存储 API v1 定义的受支持操作实施所有通用标头。

相关信息

"OpenStack： 对象存储 API"

支持的 Swift API 端点

StorageGRID 支持以下 Swift API 端点：信息 URL ，身份验证 URL 和存储 URL 。

信息 URL

您可以通过向带有 /info 路径的 Swift 基础 URL 发出 GET 请求来确定 StorageGRID Swift 实施的功能和限制。

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

在请求中：

- *FQDN* 是完全限定域名。
- *Node IP* 是StorageGRID 网络上存储节点或网关节点的IP地址。
- *Swift Port* 是用于存储节点或网关节点上的Swift API连接的端口号。

例如，以下信息 URL 将从 IP 地址为 10.99.106.103 且使用端口 18083 的存储节点请求信息。

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

此响应包括 Swift 实施的功能，可用作 JSON 词典。客户端工具可以解析 JSON 响应以确定实施的功能，并将其用作后续存储操作的约束。

通过实施 StorageGRID ， Swift 可以对信息 URL 进行未经身份验证的访问。

身份验证 **URL**

客户端可以使用 Swift 身份验证 URL 作为租户帐户用户进行身份验证。

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

您必须在中提供租户帐户ID、用户名和密码作为参数 `X-Auth-User` 和 `X-Auth-Key` 请求标题、如下所示：

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

X-Auth-Key: Password
```

在请求标题中：

- *Tenant_Account_ID* 是StorageGRID 在创建Swift租户时分配的帐户ID。这与租户管理器登录页面上使用的租户帐户 ID 相同。
- *Username* 是已在租户管理器中创建的租户用户的名称。此用户必须属于具有 Swift 管理员权限的组。无法将租户的root用户配置为使用Swift REST API。

  如果为租户帐户启用了身份联合，请从 LDAP 服务器提供联合用户的用户名和密码。或者，也可以提供 LDAP 用户的域名。例如：

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

- *Password* 是租户用户的密码。用户密码在租户管理器中创建和管理。

对成功的身份验证请求的响应将返回存储 URL 和身份验证令牌，如下所示：

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

X-Auth-Token: token

X-Storage-Token: token
```

默认情况下，令牌在生成后 24 小时内有效。

为特定租户帐户生成令牌。一个帐户的有效令牌不会授权用户访问另一个帐户。

存储 **URL**

客户端应用程序可以通过问题描述 Swift REST API 调用对网关节点或存储节点执行支持的帐户，容器和对象操作。存储请求会发送到身份验证响应中返回的存储 URL 。此请求还必须包含从身份验证请求返回的 X-Auth-Token 标头和值。

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

[/container][/object]

X-Auth-Token: token
```

某些包含使用情况统计信息的存储响应标头可能无法反映最近修改的对象的准确数字。准确的数字可能需要几分钟才能显示在这些标题中。

帐户和容器操作的以下响应标头是包含使用情况统计信息的示例：

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

相关信息

"配置租户帐户和连接"

"帐户操作"

"容器操作"

"对象操作"

帐户操作

对帐户执行以下 Swift API 操作。

获取帐户

此操作将检索与帐户和帐户使用情况统计信息关联的容器列表。

以下请求参数为必填项：

- `Account`

以下请求标头为必填项：

- `X-Auth-Token`

以下支持的请求查询参数是可选的：

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

如果找到帐户且没有容器或容器列表为空、则成功执行将返回以下标头并返回"HTTP/1.1 204 No Content"响应；如果找到帐户且容器列表不为空、则返回"HTTP/1.1 200 Ok"响应：

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

主帐户

此操作将从 Swift 帐户检索帐户信息和统计信息。

以下请求参数为必填项：

- Account

以下请求标头为必填项：

- X-Auth-Token

成功执行将返回以下标头并显示 "HTTP/1.1 204 No Content" 响应：

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

- `X-Trans-Id`

相关信息

"审核日志中跟踪的 Swift 操作"

容器操作

每个 Swift 帐户最多支持 1 ， 000 个容器。 StorageGRID对容器执行以下 Swift API 操作。

删除容器

此操作会从 StorageGRID 系统中的 Swift 帐户中删除一个空容器。

需要以下请求参数：

- `Account`
- `Container`

以下请求标头为必填项：

- `X-Auth-Token`

成功执行将返回以下标头并显示 "HTTP/1.1 204 No Content" 响应：

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

获取容器

此操作将检索与此容器关联的对象列表以及 StorageGRID 系统中的容器统计信息和元数据。

需要以下请求参数：

- `Account`
- `Container`

以下请求标头为必填项：

- `X-Auth-Token`

以下支持的请求查询参数是可选的：

- `Delimiter`
- `End_marker`

- Format
- Limit
- Marker
- Path
- Prefix

成功执行将返回以下标头，并显示 "HTTP/1.1 200 successful" 或 "HTTP/1.1 204 No Content" 响应：

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

机头容器

此操作将从 StorageGRID 系统检索容器统计信息和元数据。

需要以下请求参数：

- Account
- Container

以下请求标头为必填项：

- X-Auth-Token

成功执行将返回以下标头并显示 "HTTP/1.1 204 No Content" 响应：

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

放入容器

此操作将为 StorageGRID 系统中的帐户创建一个容器。

需要以下请求参数：

- `Account`
- `Container`

以下请求标头为必填项：

- `X-Auth-Token`

成功执行将返回以下标头，并显示 "HTTP/1.1 201 Created " 或 "HTTP/1.1 202 Accept" （如果此帐户下已存在此容器）响应：

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

容器名称在 StorageGRID 命名空间中必须是唯一的。如果容器位于其他帐户下，则返回以下标头： "HTTP/1.1 409 conflict 。 "

相关信息

"监控和审核操作"

对象操作

对对象执行以下 Swift API 操作。可以在中跟踪这些操作 "StorageGRID 审核日志"。

删除对象

此操作将从 StorageGRID 系统中删除对象的内容和元数据。

需要以下请求参数：

- `Account`
- `Container`
- `Object`

以下请求标头为必填项：

- `X-Auth-Token`

成功执行将返回以下响应标头和 `HTTP/1.1 204 No Content` 响应：

- `Content-Length`

- `Content-Type`
- `Date`
- `X-Trans-Id`

在处理删除对象请求时， StorageGRID 会尝试立即从所有存储位置删除此对象的所有副本。如果成功， StorageGRID 会立即向客户端返回响应。如果无法在30秒内删除所有副本(例如、由于某个位置暂时不可用)、则StorageGRID 会将这些副本排队等待删除、然后向客户端指示删除成功。

有关详细信息，请参见 "如何删除对象"。

获取对象

此操作将检索对象内容并从 StorageGRID 系统获取对象元数据。

需要以下请求参数：

- `Account`
- `Container`
- `Object`

以下请求标头为必填项：

- `X-Auth-Token`

以下请求标头是可选的：

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

成功执行将返回以下标头和 `HTTP/1.1 200 OK` 响应：

- `Accept-Ranges`
- `Content-Disposition`、只有在出现此情况时才返回 `Content-Disposition` 已设置元数据
- `Content-Encoding`、只有在出现此情况时才返回 `Content-Encoding` 已设置元数据
- `Content-Length`
- `Content-Type`
- `Date`
- `ETag`
- `Last-Modified`

- `X-Timestamp`
- `X-Trans-Id`

head 对象

此操作将从 StorageGRID 系统检索所载入对象的元数据和属性。

需要以下请求参数：

- `Account`
- `Container`
- `Object`

以下请求标头为必填项：

- `X-Auth-Token`

成功执行将返回以下标头并显示 "HTTP/1.1 200 OK" 响应：

- `Accept-Ranges`
- `Content-Disposition`、只有在出现此情况时才返回 `Content-Disposition` 已设置元数据
- `Content-Encoding`、只有在出现此情况时才返回 `Content-Encoding` 已设置元数据
- `Content-Length`
- `Content-Type`
- `Date`
- `ETag`
- `Last-Modified`
- `X-Timestamp`
- `X-Trans-Id`

PUT 对象

此操作将使用数据和元数据创建新对象，或者将现有对象替换为 StorageGRID 系统中的数据和元数据。

StorageGRID 支持大小高达 5 TiB （ 5 ， 497 ， 556 ， 138 ， 880 字节）的对象。

> 冲突的客户端请求（例如，两个客户端写入同一密钥）将以 " 最新成功 " 为基础进行解决。" 最新赢单 " 评估的时间取决于 StorageGRID 系统何时完成给定请求，而不是 Swift 客户端何时开始操作。

需要以下请求参数：

- `Account`
- `Container`

- `Object`

以下请求标头为必填项：

- `X-Auth-Token`

以下请求标头是可选的：

- `Content-Disposition`
- `Content-Encoding`

  请勿使用chunked `Content-Encoding` 如果ILM规则要求对对象进行适用场景 筛选、并在载入时使用同步放置(用于载入行为的平衡或严格选项)。

- `Transfer-Encoding`

  请勿使用压缩或分块 `Transfer-Encoding` 如果ILM规则要求对对象进行适用场景 筛选、并在载入时使用同步放置(用于载入行为的平衡或严格选项)。

- `Content-Length`

  如果ILM规则按大小筛选对象并在载入时使用同步放置、则必须指定 `Content-Length`。

  > 如果您未遵循这些准则、请执行以下操作 `Content-Encoding`， `Transfer-Encoding`，和 `Content-Length`、StorageGRID 必须先保存该对象、然后才能确定对象大小并应用ILM规则。换言之， StorageGRID 必须默认为在载入时创建对象的临时副本。也就是说，StorageGRID 必须对载入行为使用双提交选项。

  有关同步放置和ILM规则的详细信息、请参见 "用于载入的数据保护选项"。

- `Content-Type`
- `ETag`
- `X-Object-Meta-<name\>` (与对象相关的元数据)

  如果要使用*用户定义的创建时间*选项作为ILM规则的参考时间，则必须将该值存储在名为的用户定义标头中 `X-Object-Meta-Creation-Time`。例如：

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

  自 1970 年 1 月 1 日以来，此字段的评估值为秒。

- `X-Storage-Class: reduced_redundancy`

  如果与所载入对象匹配的 ILM 规则指定了双重提交或平衡的载入行为，则此标头会影响 StorageGRID 创建的对象副本数。

  - * 双提交 * ：如果 ILM 规则为载入行为指定了双提交选项，则 StorageGRID 会在载入对象时创建一个临时副本（单个提交）。

- 均衡：如果ILM规则指定了均衡选项，则只有当系统无法立即创建规则中指定的所有副本时，StorageGRID 才会创建一个临时副本。如果 StorageGRID 可以执行同步放置，则此标头不起作用。

  。 `reduced_redundancy` 如果与对象匹配的ILM规则创建一个复制副本、则最好使用标题。在这种情况下、使用 `reduced_redundancy` 无需在每次载入操作中创建和删除额外的对象副本。

  使用 `reduced_redundancy` 在其他情况下不建议使用标头、因为它会增加载入期间丢失对象数据的风险。例如，如果最初将单个副本存储在发生故障的存储节点上，而此存储节点未能进行 ILM 评估，则可能会丢失数据。

  > 在任何一段时间内只复制一个副本会使数据面临永久丢失的风险。如果某个对象只存在一个复制副本，则在存储节点出现故障或出现严重错误时，该对象将丢失。在升级等维护过程中，您还会暂时失去对对象的访问权限。

请注意、请指定 `reduced_redundancy` 仅影响首次载入对象时创建的副本数。它不会影响通过活动ILM策略评估对象时为对象创建的副本数、也不会导致数据在StorageGRID系统中以较低的冗余级别进行存储。

成功执行将返回以下标头，并显示 "HTTP/1.1 201 Created " 响应：

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

选项请求

选项请求会检查单个 Swift 服务的可用性。选项请求由 URL 中指定的存储节点或网关节点处理。

options 方法

例如，客户端应用程序可以在不提供 Swift 身份验证凭据的情况下将选项请求问题描述 到存储节点上的 Swift 端口，以确定存储节点是否可用。您可以使用此请求进行监控，也可以允许外部负载平衡器确定存储节点何时关闭。

与信息 URL 或存储 URL 结合使用时， options 方法将返回给定 URL 支持的动词列表（例如 head ， get ， options 和 put ）。选项方法不能与身份验证URL结合使用。

以下请求参数为必填项：

- `Account`

以下请求参数是可选的：

- `Container`

- `Object`

成功执行将返回以下标头、并返回"HTTP/1.1 204 No Content"响应。对存储 URL 的选项请求不要求目标存在。

- `Allow` (给定URL支持的动词列表、例如head、get、options、和PUT)
- `Content-Length`
- `Content-Type`
- `Date`
- `X-Trans-Id`

相关信息

"支持的 Swift API 端点"

对 **Swift API** 操作的错误响应

了解可能的错误响应有助于您对操作进行故障排除。

如果操作期间发生错误，可能会返回以下 HTTP 状态代码：

Swift 错误名称	**HTTP** 状态
AccountNameTooLong， ContainerNameTooLong， HeaderTooBig， InContainvaliderName， InvalidRequest， InvalidURI， MetadataNameTooLong， MetadataValueTooBig， MissingSecurityHeader， ObjectNameTooLong， TooManyContainers， TooManyMetadataItems， TotalMetadataTooLarge	400 个错误请求
ACCESSDENIED	403 已禁用
ContainerNotEmpty， ContainerAlreadyExists	409 冲突
内部错误	500 内部服务器错误
InvalidRange	416 无法满足请求的范围
方法未使用	不允许使用 405 方法
MissingContent长度	411 需要长度
未找到	未找到 404
未实施	501 未实施

Swift 错误名称	HTTP 状态
预条件已启用	412- 前提条件失败
ResourceNotFound	未找到 404
未授权	401 未授权
UnprocessableEntity	422 不可处理实体

StorageGRID Swift REST API 操作

在 Swift REST API 中添加了特定于 StorageGRID 系统的操作。

获取容器一致性请求

"一致性值" 在不同存储节点和站点之间的对象可用性与这些对象的一致性之间取得平衡。通过GET容器一致性请求、您可以确定应用于特定容器的一致性。

请求

请求 HTTP 标头	Description
X-Auth-Token	指定要用于请求的帐户的 Swift 身份验证令牌。
X-nta-sg-sg-sg-一致性	指定请求的类型、其中 `true` =获取容器一致性、和 `false` =获取容器。
主机	请求所定向到的主机名。

请求示例

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

响应

响应 HTTP 标头	Description
Date	响应的日期和时间。
连接	与服务器的连接是打开还是关闭。

响应 **HTTP** 标头	**Description**
X-Trans-ID	请求的唯一事务标识符。
内容长度	响应正文的长度。
X-nta-sg-sg-sg-一致性	应用于容器的一致性。支持以下值： **all**：所有节点都会立即接收数据、否则请求将失败。 强-全局：保证所有站点中所有客户端请求的写入后读一致性。 强站点：保证站点内所有客户端请求的写入后读一致性。 **read-after-new-write**：(默认)为新对象提供写后读一致性、并最终为对象更新提供一致性。提供高可用性和数据保护保证。建议用于大多数情况。 可用：为新对象和对象更新提供最终一致性。对于S3存储分段、请仅在需要时使用(例如、对于包含很少读取的日志值的存储分段、或者对于不存在的密钥执行HEAD或GET操作)。S3 FabricPool 存储分段不支持。

响应示例

```
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
```

提交容器一致性请求

通过Put容器一致性请求、您可以指定要应用于对容器执行的操作的一致性。默认情况下、新容器是使用"新写入后读取"一致性创建的。

请求

请求 **HTTP** 标头	**Description**
X-Auth-Token	要用于请求的帐户的 Swift 身份验证令牌。

请求 **HTTP** 标头	**Description**
X-nta-sg-sg-sg-一致性	要应用于容器上的操作的一致性。支持以下值： **all**：所有节点都会立即接收数据、否则请求将失败。 强-全局：保证所有站点中所有客户端请求的写入后读一致性。 强站点：保证站点内所有客户端请求的写入后读一致性。 **read-after-new-write**：(默认)为新对象提供写后读一致性、并最终为对象更新提供一致性。提供高可用性和数据保护保证。建议用于大多数情况。 可用：为新对象和对象更新提供最终一致性。对于S3存储分段、请仅在需要时使用(例如、对于包含很少读取的日志值的存储分段、或者对于不存在的密钥执行HEAD或GET操作)。S3 FabricPool 存储分段不支持。
`Host`	请求所定向到的主机名。

一致性和**ILM**规则如何相互作用以影响数据保护

您可以自由选择 "一致性值" ILM规则会影响对象的保护方式。这些设置可以进行交互。

例如、存储对象时使用的一致性会影响对象元数据的初始放置、而会影响 "加热行为" 为ILM规则选择此选项会影响对象副本的初始放置。由于StorageGRID需要同时访问对象的元数据及其数据才能满足客户端请求、因此为一致性和载入行为选择匹配的保护级别可以提供更好的初始数据保护、并提高系统响应的可预测性。

示例：如何交互一致性和**ILM**规则

假设您有一个双站点网格、该网格具有以下ILM规则、并且具有以下一致性：

- * ILM 规则 * ：创建两个对象副本，一个在本地站点，一个在远程站点。此时将选择严格的载入行为。
- **：“强全局”(对象元数据立即分发到所有站点。)

当客户端将对象存储到网格时， StorageGRID 会创建两个对象副本并将元数据分发到两个站点，然后再向客户端返回成功。

在载入成功消息时，此对象将受到完全保护，不会丢失。例如，如果本地站点在载入后不久丢失，则远程站点上仍存在对象数据和对象元数据的副本。此对象完全可检索。

如果您改用相同的ILM规则和"强站点"一致性、则在将对象数据复制到远程站点后、在远程站点分发对象元数据之前、客户端可能会收到一条成功消息。在这种情况下，对象元数据的保护级别与对象数据的保护级别不匹配。如果本地站点在载入后不久丢失，则对象元数据将丢失。无法检索此对象。

一致性和ILM规则之间的相互关系可能很复杂。如需帮助，请联系 NetApp 。

请求示例

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

响应

响应 **HTTP** 标头	**Description**
`Date`	响应的日期和时间。
`Connection`	与服务器的连接是打开还是关闭。
`X-Trans-Id`	请求的唯一事务标识符。
`Content-Length`	响应正文的长度。

响应示例

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

审核日志中跟踪的 **Swift** 操作

所有成功的存储删除， GET ， HEAD ， POST 和 PUT 操作都会在 StorageGRID 审核日志中进行跟踪。失败以及信息、身份验证或选项请求未记录。

帐户操作

- "获取帐户"
- "主帐户"

容器操作

- "删除容器"
- "获取容器"
- "机头容器"
- "放入容器"

对象操作

- "删除对象"
- "获取对象"
- "head 对象"
- "PUT 对象"

相关信息

- "访问审核日志文件"
- "客户端写入审核消息"
- "客户端读取审核消息"

版权信息

版权所有 © 2025 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。 美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

商标信息

NetApp、NetApp 标识和 http://www.netapp.com/TM 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。