概念
ONTAP Select

NetApp
November 08, 2026

This PDF was generated from https://docs.netapp.com/zh-cn/ontap-select-9191/concept_api_rest.html on November 08, 2026. Always check docs.netapp.com for the latest.

目录

概念 . . . 1
了解用于部署和管理 ONTAP Select 集群的 REST Web 服务 . . . 1
架构和经典约束 . . . 1
资源和状态表示 . . . 1
URI 端点 . . . 1
HTTP 消息 . . . 1
JSON 格式 . . . 1
了解有关访问 ONTAP Select Deploy API 的信息 . . . 2
部署实用程序本机用户界面 . . . 2
ONTAP Select Deploy 在线文档页面 . . . 2
自定义程序 . . . 2
ONTAP Select Deploy API 基本操作特性 . . . 2
Hypervisor 主机与 ONTAP Select 节点 . . . 2
对象标识符 . . . 2
请求标识符 . . . 3
同步和异步调用 . . . 3
确认长时间运行作业的完成 . . . 3
安全性 . . . 3
ONTAP Select 的请求和响应 API 事务 . . . 3
控制 API 请求的输入变量 . . . 3
解释 API 响应 . . . 5
使用 Job 对象对 ONTAP Select 进行异步处理 . . . 6
使用 Job 对象描述的异步请求 . . . 6
查询与 API 请求关联的作业对象 . . . 6
发出异步请求的一般过程 . . . 7

概念

了解用于部署和管理 ONTAP Select 集群的 REST Web 服务

表示状态传输 (REST) 是一种用于创建分布式 Web 应用程序的样式。当应用于 Web 服务 API 的设计时，它建立了一组技术和最佳实践，用于公开基于服务器的资源并管理其状态。它使用主流协议和标准，为部署和管理 ONTAP Select 集群提供灵活的基础。

架构和经典约束

REST 由 Roy Fielding 于 2000 年在 UC Irvine 的博士 "论文"中正式阐述。它通过一系列约束定义了一种架构风格，这些约束共同改进了基于 Web 的应用程序和底层协议。这些约束基于使用无状态通信协议的客户端/服务器架构建立了 RESTful Web 服务应用程序。

资源和状态表示

资源是基于 Web 的系统的基本组成部分。创建 REST Web 服务应用程序时，早期设计任务包括：

- 标识系统或基于服务器的资源 每个系统都使用和维护资源。资源可以是文件、业务事务、流程或管理实体。设计基于 REST Web 服务的应用程序的首要任务之一是识别资源。
- 资源状态和相关状态操作的定义 资源始终处于有限数量的状态之一。必须明确界定状态以及用于影响状态更改的相关操作。

根据通用 CRUD（创建、读取、更新和删除）模型，在客户端和服务器之间交换消息，以访问和更改资源的状态。

URI 端点

必须使用明确定义的编址方案来定义和提供每个 REST 资源。资源所在和标识的端点使用统一资源标识符 (URI)。URI 提供了一个通用框架，用于为网络中的每个资源创建唯一的名称。统一资源定位器 (URL) 是一种与 Web 服务一起使用的 URI 类型，用于标识和访问资源。资源通常以类似于文件目录的分层结构公开。

HTTP 消息

超文本传输协议 (HTTP) 是 Web 服务客户端和服务器用于交换有关资源的请求和响应消息的协议。作为设计 Web 服务应用程序的一部分，HTTP 谓词（如 GET 和 POST）映射到资源和相应的状态管理操作。

HTTP 是无状态的。因此，要在一个事务下关联一组相关的请求和响应，必须在请求/响应数据流所携带的 HTTP 标头中包含其他信息。

JSON 格式

虽然信息可以通过多种方式在客户端和服务器之间进行结构化和传输，但最受欢迎的选项（以及与 Deploy REST API 一起使用的选项）是 JavaScript 对象表示法 (JSON)。JSON 是以纯文本表示简单数据结构的行业标准，用于传输描述资源的状态信息。

了解有关访问 ONTAP Select Deploy API 的信息

由于 REST Web 服务固有的灵活性，ONTAP Select Deploy API 可以通过几种不同的方式访问。

ONTAP Select Deploy 附带的 REST API 分配了一个版本号。API 版本号与 Deploy 发布版本号无关。ONTAP Select 9.19.1 Deploy 管理实用程序包括 REST API 版本 3。

部署实用程序本机用户界面

访问 API 的主要方式是通过 ONTAP Select Deploy Web 用户界面。浏览器调用 API，并根据用户界面的设计重新格式化数据。您还可以通过 Deploy 实用程序命令行界面访问 API。

ONTAP Select Deploy 在线文档页面

使用浏览器时，ONTAP Select Deploy 在线文档页面提供了另一个接入点。除了提供直接执行单个 API 调用的方法外，该页面还包括 API 的详细说明，包括每个调用的输入参数和其他选项。API 调用分为几个不同的功能区域或类别。

自定义程序

您可以使用多种不同的编程语言和工具中的任何一种来访问 Deploy API。热门选择包括 Python、Java 和 cURL。使用 API 的程序、脚本或工具充当 REST Web 服务客户端。使用编程语言可以让您更好地了解 API，并提供自动化 ONTAP Select 部署的机会。

ONTAP Select Deploy API 基本操作特性

虽然 REST 建立了一套通用的技术和最佳实践，但每个 API 的详细信息可能因设计选择而异。在使用 API 之前，您应该了解 ONTAP Select Deploy API 的详细信息和操作特性。

Hypervisor 主机与 ONTAP Select 节点

虚拟机监控程序主机 是托管 ONTAP Select 虚拟机的核心硬件平台。当 ONTAP Select 虚拟机部署并在 hypervisor host 上处于活动状态时，该虚拟机被视为 *ONTAP Select* 节点。使用 Deploy REST API 版本 3，主机和节点对象是分开且不同的。这允许一对多关系，其中一个或多个 ONTAP Select 节点可以在同一 hypervisor host 上运行。

对象标识符

每个资源实例或对象在创建时都分配有唯一的标识符。这些标识符在 ONTAP Select Deploy 的特定实例中是全局唯一的。在发出创建新对象实例的 API 调用后，关联的 id 值将在 HTTP 响应的 `location` 标头中返回给调用方。您可以提取标识符，并在引用资源实例时将其用于后续调用。

对象标识符的内容和内部结构可以随时更改。在引用关联对象时，仅应根据需要在适用的 API 调用中使用标识符。

请求标识符

每个成功的 API 请求都会分配一个唯一标识符。标识符在关联 HTTP 响应的 `request-id`标头中返回。您可以使用请求标识符来共同引用单个特定 API 请求-响应事务的活动。例如，您可以根据请求 ID 检索事务的所有事件消息。

同步和异步调用

服务器执行从客户端接收的 HTTP 请求有两种主要方式：

- 同步 服务器立即执行请求，并以状态代码 200、201 或 204 响应。
- 异步 服务器接受请求并以状态代码 202 响应。这表示服务器已接受客户端请求并启动后台任务以完成请求。最终的成功或失败无法立即获得，必须通过其他 API 调用来确定。

确认长时间运行作业的完成

一般来说，任何可能需要很长时间才能完成的操作都会在服务器上使用后台任务进行异步处理。使用 Deploy REST API，每个后台任务都由 Job 对象锚定，该对象跟踪任务并提供信息，例如当前状态。在创建后台任务后，在 HTTP 响应中返回 Job 对象，包括其唯一标识符。

您可以直接查询 Job 对象，以确定关联 API 调用的成功或失败。有关其他信息，请参阅_使用 Job 对象的异步处理_。

除了使用 Job 对象之外，还有其他方法可以确定请求的成功或失败，包括：

- 事件消息 您可以使用与原始响应一起返回的请求 ID 检索与特定 API 调用关联的所有事件消息。事件消息通常包含成功或失败的指示，并且在调试错误条件时也很有用。
- 资源状态或状态 一些资源维护一个状态或状态值，您可以查询该值以间接确定请求的成功或失败。

安全性

Deploy API 使用以下安全技术：

- 传输层安全性 Deploy 服务器和客户端之间通过网络发送的所有流量都通过 TLS 加密。不支持在未加密的通道上使用 HTTP 协议。支持 TLS 1.2 版本。
- HTTP 身份验证 基本身份验证用于每个 API 事务。每个请求都会添加一个 HTTP 标头，其中包括 base64 字符串中的用户名和密码。

ONTAP Select 的请求和响应 API 事务

每个 Deploy API 调用都作为对 Deploy 虚拟机的 HTTP 请求执行，该虚拟机生成对客户端的关联响应。此请求/响应对被视为 API 事务。在使用 Deploy API 之前，您应该熟悉可用于控制请求的输入变量和响应输出的内容。

控制 API 请求的输入变量

您可以通过在 HTTP 请求中设置的参数来控制如何处理 API 调用。

请求标头

您必须在 HTTP 请求中包含多个标头，包括：

- content-type 如果请求正文包含 JSON，则必须将此标头设置为 application/json。
- accept 如果响应正文将包含 JSON，则必须将此标头设置为 application/json。
- authorization 必须使用 base64 字符串编码的用户名和密码设置基本身份验证。

请求正文

请求正文的内容因特定调用而异。HTTP 请求正文由以下内容之一组成：

- 具有输入变量的 JSON 对象（例如，新集群的名称）
- 空

筛选对象

在发布使用 GET 的 API 调用时，可以根据任何属性来限制或筛选返回的对象。例如，您可以指定要匹配的确切值：

```
<field>=<query value>
```

除了精确匹配之外，还有其他运算符可用于返回一系列值范围内的一组对象。ONTAP Select 支持以下所示的过滤运算符。

运算符	问题描述
=	等于
<	小于
>	大于
⇐	小于或等于
>=	大于或等于
	或
!	不等于
*	贪婪通配符

您还可以使用 null 关键字或其否定（!null）作为查询的一部分，根据是否设置特定字段来返回一组对象。

选择对象字段

默认情况下，使用 GET 发出 API 调用仅返回唯一标识对象的属性。此最小字段集充当每个对象的键，并根据对象类型而变化。可以通过以下方式使用 fields 查询参数选择其他对象属性：

- 廉价字段 指定 `fields=*`检索维护在本地服务器内存中或需要很少处理才能访问的对象字段。
- 昂贵字段 指定 `fields=**`检索所有对象字段，包括需要额外服务器处理才能访问的字段。
- 自定义字段选择 使用 `fields=FIELDNAME`指定所需的确切字段。请求多个字段时，必须使用不带空格的逗

号分隔值。

作为最佳做法，您应始终确定所需的特定字段。您只应在需要时检索一组便宜或昂贵的字段。便宜和昂贵的分类由 NetApp 根据内部性能分析确定。给定字段的分类可以随时更改。

对输出集中的对象进行排序

资源集合中的记录按对象定义的默认顺序返回。您可以使用 order_by 查询参数以及字段名称和排序方向更改顺序，如下所示：

```
order_by=<field name> asc|desc
```

例如，您可以按降序对类型字段进行排序，然后按升序对 id 进行排序：

```
order_by=type desc, id asc
```

当包含多个参数时，必须使用逗号分隔字段。

分页

使用 GET 发出 API 调用以访问相同类型的对象集合时，默认情况下会返回所有匹配的对象。如果需要，您可以使用带有请求的 max_records 查询参数来限制返回的记录数。例如：

```
max_records=20
```

如果需要，您可以将此参数与其他查询参数相结合，以缩小结果集。例如，以下内容最多返回在指定时间之后生成的 10 个系统事件：

```
time⇒ 2019-04-04T15:41:29.140265Z&max_records=10
```

您可以发出多个请求来浏览事件（或任何对象类型）。每个后续 API 调用应根据最后结果集中的最新事件使用新的时间值。

解释 API 响应

每个 API 请求都会生成一个回复给客户端的响应。您可以检查响应以确定它是否成功，并根据需要检索其他数据。

HTTP 状态代码

下面介绍 Deploy REST API 使用的 HTTP 状态代码。

代码	含义	问题描述
200	确定	表示未创建新对象的调用成功。
201	已创建	已成功创建对象；位置响应标头包括此对象的唯一标识符。
202	已接受	已启动长时间运行的后台作业以执行请求，但操作尚未完成。
400	错误请求	请求输入无法识别或不合适。
403	禁止	由于授权错误，访问被拒绝。
404	未找到	此请求中引用的资源不存在。
405	不允许使用此方法	此资源不支持请求中的 HTTP 谓词。
409	冲突	尝试创建对象失败，因为此对象已存在。

代码	含义	问题描述
500	内部错误	服务器出现常规内部错误。
501	未实现	URI 已知，但无法执行请求。

响应标头

Deploy 服务器生成的 HTTP 响应中包括几个标头，包括：

- request-id 为每个成功的 API 请求分配唯一的请求标识符。
- 位置 创建对象时，位置标头包括新对象的完整 URL，包括唯一对象标识符。

响应正文

与 API 请求相关联的响应的内容因对象、处理类型以及请求的成功或失败而异。响应正文以 JSON 呈现。

- 单个对象 单个对象可以根据请求返回一组字段。例如，您可以使用 GET 来使用唯一标识符检索集群的选定属性。
- 多个对象 可以从资源集合中返回多个对象。在所有情况下，都使用一致的格式， `num_records` 指示记录数量，records 包含对象实例数组。例如，您可以检索特定集群中定义的所有节点。
- Job 对象如果异步处理 API 调用，则返回一个 Job 对象，该对象锚定后台任务。例如，用于部署集群的 POST 请求是异步处理的，并返回 Job 对象。
- Error 对象如果发生错误，则始终返回 Error 对象。例如，当您尝试使用已存在的名称创建集群时，您将收到一个错误。
- 空 在某些情况下，不会返回任何数据，并且响应正文为空。例如，使用 DELETE 删除现有主机后，响应正文为空。

使用 Job 对象对 ONTAP Select 进行异步处理

某些 Deploy API 调用（尤其是创建或修改资源的调用）可能需要比其他调用更长的时间才能完成。ONTAP Select Deploy 异步处理这些长时间运行的请求。

使用 Job 对象描述的异步请求

在进行异步运行的 API 调用后，HTTP 响应代码 202 表示请求已成功验证和接受，但尚未完成。此请求作为后台任务处理，此后台任务在对客户端的初始 HTTP 响应后继续运行。响应包括锚定请求的 Job 对象，包括其唯一标识符。



您应参阅 ONTAP Select Deploy 在线文档页面，以确定哪些 API 调用异步运行。

查询与 API 请求关联的作业对象

HTTP 响应中返回的 Job 对象包含多个属性。您可以查询 state 属性以确定请求是否成功完成。Job 对象可以处于以下状态之一：

- 已排队

- 正在运行
- 成功
- 失败

在轮询 Job 对象以检测任务的终端状态时，您可以使用两种技术，无论是成功还是失败：

- 标准轮询请求 立即返回当前作业状态
- 仅当出现以下情况之一时，才会返回长轮询请求作业状态：
  - 状态的更改时间比轮询请求中提供的日期时间值更近
  - 超时值已过期（1 到 120 秒）

标准轮询和长轮询使用相同的 API 调用来查询作业对象。但是，长轮询请求包含两个查询参数：`poll_timeout` 和 `last_modified`。



应始终使用长轮询来减少 Deploy 虚拟机上的工作负载。

发出异步请求的一般过程

您可以使用以下高级过程完成异步 API 调用：

1. 发出异步 API 调用。
2. 接收 HTTP 响应 202，指示成功接受请求。
3. 从响应正文中提取 Job 对象的标识符。
4. 在循环中，在每个周期中执行以下操作：
   a. 使用长轮询请求获取 Job 的当前状态
   b. 如果作业处于非终端状态（已排队、正在运行），请再次执行循环。
5. 当作业达到终止状态（成功、失败）时停止。

版权信息

版权所有 © 2026 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。