Conceitos

ONTAP Select

NetApp
November 08, 2026

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

Índice

Conceitos . . . 1
- Saiba mais sobre os serviços web REST para implantar e gerenciar clusters ONTAP Select . . . 1
  - Arquitetura e restrições clássicas . . . 1
  - Recursos e representação de estado . . . 1
  - Endpoints URI . . . 1
  - Mensagens HTTP . . . 1
  - Formatação JSON . . . 2
- Saiba como acessar a API de Deploy do ONTAP Select . . . 2
  - Implantar interface de usuário nativa da ferramenta . . . 2
  - Página de documentação online do ONTAP Select Deploy . . . 2
  - Programa personalizado . . . 2
- Características operacionais básicas da API ONTAP Select Deploy . . . 2
  - Host do hipervisor versus nó do ONTAP Select . . . 3
  - Identificadores de objetos . . . 3
  - Identificadores de solicitação . . . 3
  - Chamadas síncronas e assíncronas . . . 3
  - Confirme a conclusão de um trabalho de longa duração . . . 3
  - Segurança . . . 4
- Transação de API de solicitação e resposta para ONTAP Select . . . 4
  - Variáveis de entrada que controlam uma solicitação de API . . . 4
  - Interpretar uma resposta de API . . . 6
- Processamento assíncrono usando o objeto Job para ONTAP Select . . . 7
  - Solicitações assíncronas descritas usando o objeto Job . . . 7
  - Consultar o objeto Job associado a uma solicitação de API . . . 7
  - Procedimento geral para emitir uma solicitação assíncrona . . . 8

Conceitos

Saiba mais sobre os serviços web REST para implantar e gerenciar clusters ONTAP Select

A Transferência de Estado Representacional (REST) é um estilo para a criação de aplicações web distribuídas. Quando aplicada ao design de uma API de serviços web, ela estabelece um conjunto de tecnologias e boas práticas para expor recursos baseados em servidor e gerenciar seus estados. Utiliza protocolos e padrões consagrados para fornecer uma base flexível para a implantação e o gerenciamento de clusters ONTAP Select.

Arquitetura e restrições clássicas

O REST foi formalmente articulado por Roy Fielding em seu doutorado "dissertação" na UC Irvine em 2000. Ele define um estilo arquitetônico por meio de um conjunto de restrições que, coletivamente, aprimoraram aplicações baseadas na Web e os protocolos subjacentes. As restrições estabelecem uma aplicação de serviços web RESTful baseada em uma arquitetura cliente/servidor usando um protocolo de comunicação sem estado.

Recursos e representação de estado

Os recursos são os componentes básicos de um sistema baseado na Web. Ao criar um aplicativo de serviços web REST, as primeiras tarefas de projeto incluem:

- Identificação de recursos do sistema ou do servidor Todo sistema utiliza e mantém recursos. Um recurso pode ser um arquivo, uma transação comercial, um processo ou uma entidade administrativa. Uma das primeiras tarefas no projeto de uma aplicação baseada em serviços web REST é identificar os recursos.
- Definição dos estados dos recursos e operações associadas aos estados Os recursos estão sempre em um de um número finito de estados. Os estados, bem como as operações associadas usadas para efetuar as mudanças de estado, devem ser claramente definidos.

As mensagens são trocadas entre o cliente e o servidor para acessar e alterar o estado dos recursos de acordo com o modelo genérico CRUD (Create, Read, Update, and Delete).

Endpoints URI

Todo recurso REST deve ser definido e disponibilizado utilizando um esquema de endereçamento bem definido. Os endpoints onde os recursos estão localizados e identificados utilizam um Identificador Uniforme de Recursos (URI). O URI fornece uma estrutura geral para a criação de um nome único para cada recurso na rede. O Localizador Uniforme de Recursos (URL) é um tipo de URI utilizado com serviços baseados na Web para identificar e acessar recursos. Os recursos são normalmente expostos em uma estrutura hierárquica semelhante a um diretório de arquivos.

Mensagens HTTP

O Protocolo de Transferência de Hipertexto (HTTP) é o protocolo usado pelo cliente e servidor de serviços web para trocar mensagens de requisição e resposta sobre os recursos. Como parte do projeto de uma aplicação de serviços web, os verbos HTTP (como GET e POST) são mapeados para os recursos e as respectivas ações de gerenciamento de estado.

O HTTP não mantém estado. Portanto, para associar um conjunto de solicitações e respostas relacionadas em uma única transação, informações adicionais devem ser incluídas nos cabeçalhos HTTP transmitidos pelos fluxos de dados de solicitação/resposta.

Formatação JSON

Embora as informações possam ser estruturadas e transferidas entre um cliente e um servidor de diversas maneiras, a opção mais popular (e a utilizada com a API REST Deploy) é JavaScript Object Notation (JSON). JSON é um padrão do setor para representar estruturas de dados simples em texto sem formatação e é usado para transferir informações de estado que descrevem os recursos.

Saiba como acessar a API de Deploy do ONTAP Select

Devido à flexibilidade inerente dos serviços web REST, a API ONTAP Select Deploy pode ser acessada de diversas maneiras.

> A API REST incluída no ONTAP Select Deploy possui um número de versão. O número da versão da API é independente do número da versão do Deploy. O utilitário de administração do ONTAP Select 9.19.1 Deploy inclui a versão 3 da API REST.

Implantar interface de usuário nativa da ferramenta

A principal forma de acessar a API é através da interface de usuário baseada na Web do ONTAP Select Deploy. O navegador faz chamadas à API e reformata os dados de acordo com o design da interface de usuário. Você também acessa a API através da interface de linha de comando da Deploy utility.

Página de documentação online do ONTAP Select Deploy

A página de documentação online do ONTAP Select Deploy oferece um ponto de acesso alternativo para uso em navegadores. Além de permitir a execução direta de chamadas de API individuais, a página também inclui uma descrição detalhada da API, incluindo parâmetros de entrada e outras opções para cada chamada. As chamadas de API são organizadas em diversas áreas funcionais ou categorias.

Programa personalizado

Você pode acessar a API de Deploy usando diversas linguagens de programação e ferramentas. Opções populares incluem Python, Java e cURL. Um programa, script ou ferramenta que utiliza a API atua como um cliente de serviços web REST. Usar uma linguagem de programação permite que você compreenda melhor a API e oferece a oportunidade de automatizar as implantações do ONTAP Select.

Características operacionais básicas da API ONTAP Select Deploy

Embora o REST estabeleça um conjunto comum de tecnologias e boas práticas, os detalhes de cada API podem variar de acordo com as escolhas de design. Você deve estar ciente dos detalhes e das características operacionais da ONTAP Select Deploy API antes de usar a API.

Host do hipervisor versus nó do ONTAP Select

Um host de hipervisor é a plataforma de hardware central que hospeda uma máquina virtual ONTAP Select. Quando uma máquina virtual ONTAP Select é implantada e está ativa em um host de hipervisor, a máquina virtual é considerada um nó ONTAP Select. Com a versão 3 da API REST do Deploy, os objetos host e nó são separados e distintos. Isso permite um relacionamento um-para-muitos, em que um ou mais nós ONTAP Select podem ser executados no mesmo host de hipervisor.

Identificadores de objetos

Cada instância ou objeto de recurso recebe um identificador único no momento de sua criação. Esses identificadores são globalmente únicos dentro de uma instância específica do ONTAP Select Deploy. Após a emissão de uma chamada de API que cria uma nova instância de objeto, o valor do id associado é retornado ao solicitante no `location` header da resposta HTTP. Você pode extrair o identificador e usá-lo em chamadas subsequentes ao se referir à instância do recurso.

O conteúdo e a estrutura interna dos identificadores de objeto podem mudar a qualquer momento. Você deve usar os identificadores apenas nas chamadas de API aplicáveis, conforme necessário, ao se referir aos objetos associados.

Identificadores de solicitação

A cada requisição de API bem-sucedida é atribuído um identificador único. Esse identificador é retornado no `request-id` cabeçalho da resposta HTTP associada. Você pode usar um identificador de requisição para se referir coletivamente às atividades de uma única transação de requisição-resposta da API. Por exemplo, você pode recuperar todas as mensagens de evento de uma transação com base no id da requisição.

Chamadas síncronas e assíncronas

Existem duas maneiras principais pelas quais um servidor processa uma solicitação HTTP recebida de um cliente:

- Síncrono O servidor executa a solicitação imediatamente e responde com um código de status 200, 201 ou 204.
- Assíncrono O servidor aceita a solicitação e responde com um código de status 202. Isso indica que o servidor aceitou a solicitação do cliente e iniciou uma tarefa em segundo plano para concluir a solicitação. O sucesso ou falha final não está disponível imediatamente e deve ser determinado por meio de chamadas de API REST adicionais.

Confirme a conclusão de um trabalho de longa duração

Geralmente, qualquer operação que possa levar muito tempo para ser concluída é processada de forma assíncrona usando uma tarefa em segundo plano no servidor. Com a Deploy API REST, cada tarefa em segundo plano é ancorada por um objeto Job que rastreia a tarefa e fornece informações, como o estado atual. Um objeto Job, incluindo seu identificador exclusivo, é retornado na resposta HTTP após a criação de uma tarefa em segundo plano.

Você pode consultar o objeto Job diretamente para determinar o sucesso ou a falha da chamada de API associada. Consulte *processamento assíncrono usando o objeto Job* para obter mais informações.

Além de usar o objeto Job, existem outras maneiras de determinar o sucesso ou a falha de uma solicitação, incluindo:

- Mensagens de evento Você pode recuperar todas as mensagens de evento associadas a uma chamada de API específica usando o ID da solicitação retornado na resposta original. As mensagens de evento geralmente contêm uma indicação de sucesso ou falha e também podem ser úteis na depuração de uma condição de erro.
- Estado ou status do recurso Vários recursos mantêm um valor de estado ou status que você pode consultar para determinar indiretamente o sucesso ou a falha de uma solicitação.

Segurança

A API de Implantação utiliza as seguintes tecnologias de segurança:

- Transport Layer Security Todo o tráfego enviado pela rede entre o servidor Deploy e o cliente é criptografado por meio do TLS. O uso do protocolo HTTP em um canal não criptografado não é compatível. A versão 1.2 do TLS é compatível.
- A autenticação HTTP Basic authentication é usada em todas as transações da API. Um cabeçalho HTTP, que inclui o nome de usuário e a senha em uma string base64, é adicionado a cada solicitação.

Transação de API de solicitação e resposta para ONTAP Select

Cada chamada à API Deploy é realizada como uma requisição HTTP para a máquina virtual Deploy, que gera uma resposta associada para o cliente. Esse par requisição/resposta é considerado uma transação de API. Antes de usar a API Deploy, você deve se familiarizar com as variáveis de entrada disponíveis para controlar uma requisição e com o conteúdo da saída da resposta.

Variáveis de entrada que controlam uma solicitação de API

Você pode controlar como uma chamada de API é processada por meio de parâmetros definidos na solicitação HTTP.

Cabeçalhos da solicitação

Você deve incluir vários cabeçalhos na solicitação HTTP, incluindo:

- content-type Se o corpo da requisição incluir JSON, este cabeçalho deve ser definido como application/json.
- accept Se o corpo da resposta incluir JSON, este cabeçalho deve ser definido como application/json.
- A autenticação Basic deve ser configurada com o nome de usuário e a senha codificados em uma string base64.

Corpo da solicitação

O conteúdo do corpo da requisição varia dependendo da chamada específica. O corpo da requisição HTTP consiste em um dos seguintes elementos:

- Objeto JSON com variáveis de entrada (como o nome de um novo cluster)
- Vazio

Filtrar objetos

Ao fazer uma chamada de API usando GET, você pode limitar ou filtrar os objetos retornados com base em qualquer atributo. Por exemplo, você pode especificar um valor exato para correspondência:

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

Além da correspondência exata, existem outros operadores disponíveis para retornar um conjunto de objetos em um intervalo de valores. ONTAP Select oferece suporte aos operadores de filtragem mostrados abaixo.

Operador	Descrição
=	Igual a
<	Menor que
>	Maior que
⇐	Menor ou igual a
>=	Maior ou igual a
	Ou
!	Não é igual a
*	Curinga ganancioso

Você também pode retornar um conjunto de objetos com base em se um campo específico está definido ou não, usando a palavra-chave null ou sua negação (!null) como parte da consulta.

Selecionando campos de objeto

Por padrão, uma chamada de API usando o método GET retorna apenas os atributos que identificam exclusivamente o objeto ou objetos. Esse conjunto mínimo de campos funciona como uma chave para cada objeto e varia de acordo com o tipo de objeto. Você pode selecionar propriedades adicionais do objeto usando o parâmetro de consulta fields das seguintes maneiras:

- Campos de baixo custo Especifique `fields=*` para recuperar os campos do objeto que são mantidos na memória do servidor local ou que exigem pouco processamento para acesso.
- Campos dispendiosos Especifique `fields=**` para recuperar todos os campos do objeto, incluindo aqueles que exigem processamento adicional do servidor para serem acessados.
- Seleção de campo personalizado Use `fields=FIELDNAME` para especificar o campo exato que você deseja. Ao solicitar vários campos, os valores devem ser separados por vírgulas sem espaços.



Como prática recomendada, você deve sempre identificar os campos específicos que deseja. Você só deve recuperar o conjunto de campos de baixo ou alto custo quando necessário. A classificação de baixo e alto custo é determinada pela NetApp com base em análises internas de desempenho. A classificação de um determinado campo pode mudar a qualquer momento.

Ordenar objetos no conjunto de saída

Os registros em uma coleção de recursos são retornados na ordem padrão definida pelo objeto. Você pode alterar a ordem usando o parâmetro de consulta order_by com o nome do campo e a direção da classificação, conforme mostrado a seguir:

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

Por exemplo, você pode classificar o campo type em ordem decrescente, seguido pelo campo id em ordem crescente:
`order_by=type desc, id asc`

Ao incluir vários parâmetros, você deve separar os campos com uma vírgula.

Paginação

Ao fazer uma chamada de API usando o método GET para acessar uma coleção de objetos do mesmo tipo, todos os objetos correspondentes são retornados por padrão. Se necessário, você pode limitar o número de registros retornados usando o parâmetro de consulta max_records na solicitação. Por exemplo:
`max_records=20`

Se necessário, você pode combinar esse parâmetro com outros parâmetros de consulta para restringir o conjunto de resultados. Por exemplo, o seguinte retorna até 10 eventos do sistema gerados após o horário especificado:
`time⇒ 2019-04-04T15:41:29.140265Z&max_records=10`

Você pode enviar várias solicitações para percorrer os eventos (ou qualquer tipo de objeto). Cada chamada de API subsequente deve usar um novo valor de tempo com base no evento mais recente no último conjunto de resultados.

Interpretar uma resposta de API

Cada solicitação à API gera uma resposta para o cliente. Você pode examinar a resposta para determinar se ela foi bem-sucedida e recuperar dados adicionais, se necessário.

Código de status HTTP

Os códigos de status HTTP usados pela API REST Deploy são descritos abaixo.

Código	Significado	Descrição
200	OK	Indica sucesso para chamadas que não criam um novo objeto.
201	Criado	Um objeto foi criado com sucesso; o cabeçalho de resposta de localização inclui o identificador exclusivo do objeto.
202	Aceito	Uma tarefa em segundo plano de longa duração foi iniciada para executar a solicitação, mas a operação ainda não foi concluída.
400	Pedido ruim	A entrada solicitada não foi reconhecida ou é inadequada.
403	Proibido	O acesso foi negado devido a um erro de autorização.
404	Não encontrado	O recurso mencionado na solicitação não existe.
405	Método não permitido	O verbo HTTP na requisição não é suportado para o recurso.
409	Conflito	A tentativa de criar um objeto falhou porque o objeto já existe.
500	Erro interno	Ocorreu um erro interno geral no servidor.
501	Não implementado	O URI é conhecido, mas não é capaz de executar a solicitação.

Cabeçalhos de resposta

Diversos cabeçalhos estão incluídos na resposta HTTP gerada pelo servidor Deploy, incluindo:

- request-id Cada solicitação de API bem-sucedida recebe um identificador de solicitação exclusivo.
- location Quando um objeto é criado, o cabeçalho de location inclui a URL completa do novo objeto, incluindo o identificador exclusivo do objeto.

Corpo de resposta

O conteúdo da resposta associada a uma solicitação de API varia de acordo com o objeto, o tipo de processamento e o sucesso ou falha da solicitação. O corpo da resposta é apresentado em JSON.

- Um único objeto pode ser retornado com um conjunto de campos com base na solicitação. Por exemplo, você pode usar GET para recuperar propriedades selecionadas de um cluster usando o identificador exclusivo.
- Vários objetos de uma coleção de recursos podem ser retornados. Em todos os casos, um formato consistente é utilizado, com `num_records` indicando o número de registros e os registros contendo uma matriz de instâncias do objeto. Por exemplo, você pode recuperar todos os nós definidos em um cluster específico.
- Objeto Job Se uma chamada de API for processada de forma assíncrona, um objeto Job será retornado, o qual representa a tarefa em segundo plano. Por exemplo, a solicitação POST usada para implantar um cluster é processada de forma assíncrona e retorna um objeto Job.
- Objeto de erro Se ocorrer um erro, um objeto de erro será sempre retornado. Por exemplo, você receberá um erro ao tentar criar um cluster com um nome que já existe.
- Em certos casos, nenhum dado é retornado e o corpo da resposta fica vazio. Por exemplo, o corpo da resposta fica vazio após usar DELETE para excluir um host existente.

Processamento assíncrono usando o objeto Job para ONTAP Select

Algumas chamadas da API Deploy, principalmente aquelas que criam ou modificam um recurso, podem demorar mais para serem concluídas do que outras chamadas. ONTAP Select Deploy processa essas solicitações de longa duração de forma assíncrona.

Solicitações assíncronas descritas usando o objeto Job

Após uma chamada de API assíncrona, o código de resposta HTTP 202 indica que a solicitação foi validada e aceita com sucesso, mas ainda não foi concluída. A solicitação é processada como uma tarefa em segundo plano que continua a ser executada após a resposta HTTP inicial ao cliente. A resposta inclui o objeto Job que ancora a solicitação, incluindo seu identificador exclusivo.

Você deve consultar a página de documentação online do ONTAP Select Deploy para determinar quais chamadas de API operam de forma assíncrona.

Consultar o objeto Job associado a uma solicitação de API

O objeto Job retornado na resposta HTTP contém diversas propriedades. Você pode consultar a propriedade state para determinar se a solicitação foi concluída com sucesso. Um objeto Job pode estar em um dos seguintes estados:

- Em fila
- Em execução
- Sucesso
- Falha

Existem duas técnicas que você pode usar ao consultar um objeto Job para detectar o estado final da tarefa, seja sucesso ou falha:

- A solicitação de sondagem padrão retorna imediatamente o estado atual do job
- A solicitação de long polling retorna o estado da tarefa somente quando uma das seguintes situações ocorre:
  - O estado mudou mais recentemente do que o valor de data e hora fornecido na solicitação da pesquisa
  - O tempo limite expirou (1 a 120 segundos)

A pesquisa padrão e a pesquisa longa usam a mesma chamada de API para consultar um objeto Job. No entanto, uma solicitação de pesquisa longa inclui dois parâmetros de consulta: `poll_timeout` e `last_modified`.



Você deve sempre usar o método de polling longo para reduzir a carga de trabalho na máquina virtual Deploy.

Procedimento geral para emitir uma solicitação assíncrona

Você pode usar o seguinte procedimento de alto nível para concluir uma chamada de API assíncrona:

1. Emita a chamada de API assíncrona.
2. Receba uma resposta HTTP 202 indicando aceitação bem-sucedida da solicitação.
3. Extraia o identificador do objeto Job do corpo da resposta.
4. Dentro de um laço, execute o seguinte em cada ciclo:
   a. Obtenha o estado atual do Job com uma solicitação de longa duração
   b. Se a tarefa estiver em um estado não terminal (enfileirada, em execução), execute o loop novamente.
5. Pare quando a tarefa atingir um estado terminal (sucesso, falha).

Informações sobre direitos autorais

Copyright © 2026 NetApp, Inc. Todos os direitos reservados. Impresso nos EUA. Nenhuma parte deste documento protegida por direitos autorais pode ser reproduzida de qualquer forma ou por qualquer meio — gráfico, eletrônico ou mecânico, incluindo fotocópia, gravação, gravação em fita ou storage em um sistema de recuperação eletrônica — sem permissão prévia, por escrito, do proprietário dos direitos autorais.

O software derivado do material da NetApp protegido por direitos autorais está sujeito à seguinte licença e isenção de responsabilidade:

ESTE SOFTWARE É FORNECIDO PELA NETAPP "NO PRESENTE ESTADO" E SEM QUAISQUER GARANTIAS EXPRESSAS OU IMPLÍCITAS, INCLUINDO, SEM LIMITAÇÕES, GARANTIAS IMPLÍCITAS DE COMERCIALIZAÇÃO E ADEQUAÇÃO A UM DETERMINADO PROPÓSITO, CONFORME A ISENÇÃO DE RESPONSABILIDADE DESTE DOCUMENTO. EM HIPÓTESE ALGUMA A NETAPP SERÁ RESPONSÁVEL POR QUALQUER DANO DIRETO, INDIRETO, INCIDENTAL, ESPECIAL, EXEMPLAR OU CONSEQUENCIAL (INCLUINDO, SEM LIMITAÇÕES, AQUISIÇÃO DE PRODUTOS OU SERVIÇOS SOBRESSALENTES; PERDA DE USO, DADOS OU LUCROS; OU INTERRUPÇÃO DOS NEGÓCIOS), INDEPENDENTEMENTE DA CAUSA E DO PRINCÍPIO DE RESPONSABILIDADE, SEJA EM CONTRATO, POR RESPONSABILIDADE OBJETIVA OU PREJUÍZO (INCLUINDO NEGLIGÊNCIA OU DE OUTRO MODO), RESULTANTE DO USO DESTE SOFTWARE, MESMO SE ADVERTIDA DA RESPONSABILIDADE DE TAL DANO.

A NetApp reserva-se o direito de alterar quaisquer produtos descritos neste documento, a qualquer momento e sem aviso. A NetApp não assume nenhuma responsabilidade nem obrigação decorrentes do uso dos produtos descritos neste documento, exceto conforme expressamente acordado por escrito pela NetApp. O uso ou a compra deste produto não representam uma licença sob quaisquer direitos de patente, direitos de marca comercial ou quaisquer outros direitos de propriedade intelectual da NetApp.

O produto descrito neste manual pode estar protegido por uma ou mais patentes dos EUA, patentes estrangeiras ou pedidos pendentes.

LEGENDA DE DIREITOS LIMITADOS: o uso, a duplicação ou a divulgação pelo governo estão sujeitos a restrições conforme estabelecido no subparágrafo (b)(3) dos Direitos em Dados Técnicos - Itens Não Comerciais no DFARS 252.227-7013 (fevereiro de 2014) e no FAR 52.227- 19 (dezembro de 2007).

Os dados aqui contidos pertencem a um produto comercial e/ou serviço comercial (conforme definido no FAR 2.101) e são de propriedade da NetApp, Inc. Todos os dados técnicos e software de computador da NetApp fornecidos sob este Contrato são de natureza comercial e desenvolvidos exclusivamente com despesas privadas. O Governo dos EUA tem uma licença mundial limitada, irrevogável, não exclusiva, intransferível e não sublicenciável para usar os Dados que estão relacionados apenas com o suporte e para cumprir os contratos governamentais desse país que determinam o fornecimento de tais Dados. Salvo disposição em contrário no presente documento, não é permitido usar, divulgar, reproduzir, modificar, executar ou exibir os dados sem a aprovação prévia por escrito da NetApp, Inc. Os direitos de licença pertencentes ao governo dos Estados Unidos para o Departamento de Defesa estão limitados aos direitos identificados na cláusula 252.227-7015(b) (fevereiro de 2014) do DFARS.

Informações sobre marcas comerciais

NETAPP, o logotipo NETAPP e as marcas listadas em http://www.netapp.com/TM são marcas comerciais da NetApp, Inc. Outros nomes de produtos e empresas podem ser marcas comerciais de seus respectivos proprietários.