



# **Automatize com o uso de APIS REST**

## **SnapCenter Software 6.0**

NetApp  
December 19, 2024

# Índice

- Automatize com o uso de APIS REST ..... 1
  - Visão geral das APIs REST ..... 1
  - Como acessar a API REST do SnapCenter nativamente ..... 1
  - Base de serviços web REST ..... 1
  - Caraterísticas operacionais básicas ..... 2
  - Variáveis de entrada que controlam uma solicitação de API ..... 4
  - Interpretação de uma resposta API ..... 7
  - APIs REST compatíveis com servidor SnapCenter e plug-ins ..... 10
  - Como acessar APIs REST usando a página da Web da API Swagger ..... 17
  - Comece a usar a API REST ..... 18

# Automatize com o uso de APIS REST

## Visão geral das APIs REST

As APIs REST podem ser usadas para executar várias operações de gerenciamento do SnapCenter. As APIs REST são expostas por meio da página da Web do Swagger.

Você pode acessar a página da Web do Swagger disponível em `https:// SnapCenter_IP_address_or_name> SnapCenter_port>/swagger/_` para exibir a documentação da API REST, bem como emitir manualmente uma chamada de API.

Os plug-ins compatíveis com APIs REST são:

- Plug-in para Microsoft SQL Server
- Plug-in para banco de dados SAP HANA
- Plug-ins personalizados
- Plug-in para Oracle Database

## Como acessar a API REST do SnapCenter nativamente

Você pode acessar a API REST do SnapCenter diretamente usando qualquer linguagem de programação compatível com um CLIENTE REST. As opções de linguagem populares incluem Python, PowerShell e Java.

## Base de serviços web REST

Representational State Transfer (REST) é um estilo para a criação de aplicações web distribuídas. Quando aplicada ao design de uma API de serviços da Web, ela estabelece um conjunto de tecnologias e melhores práticas para expor recursos baseados em servidor e gerenciar seus estados. Ele usa protocolos e padrões mainstream para fornecer uma base flexível para o gerenciamento de SnapCenter.

### Recursos e representação do Estado

Os recursos são os componentes básicos de um sistema baseado na Web. Ao criar um aplicativo REST de serviços da Web, as tarefas iniciais de design incluem:

#### Identificação de recursos baseados em sistema ou servidor

Cada sistema usa e mantém recursos. Um recurso pode ser um arquivo, transação comercial, processo ou entidade administrativa. Uma das primeiras tarefas no projeto de um aplicativo baseado em serviços web REST é identificar os recursos.

#### Definição de estados de recursos e operações de estado associadas

Os recursos estão sempre em um de um número finito de estados. Os estados, bem como as operações associadas usadas para afetar as mudanças de estado, devem ser claramente definidos.

## Pontos de extremidade URI

Todos os recursos REST devem ser definidos e disponibilizados usando um esquema de endereçamento bem definido. Os endpoints onde os recursos estão localizados e identificados usam um URI (Uniform Resource Identifier).

O URI fornece uma estrutura geral para criar um nome exclusivo para cada recurso na rede. O Uniform Resource Locator (URL) é um tipo de URI usado com serviços da Web para identificar e acessar recursos. Os recursos são normalmente expostos em uma estrutura hierárquica semelhante a um diretório de arquivos.

## Mensagens HTTP

O Hypertext Transfer Protocol (HTTP) é o protocolo usado pelo cliente e servidor de serviços da Web para trocar mensagens de solicitação e resposta sobre os recursos.

Como parte do projeto de um aplicativo de serviços da Web, os métodos HTTP são mapeados para os recursos e as ações de gerenciamento de estado correspondentes. HTTP está sem estado. Portanto, para associar um conjunto de solicitações e respostas relacionadas como parte de uma transação, informações adicionais devem ser incluídas nos cabeçalhos HTTP carregados com os fluxos de dados de solicitação e resposta.

## Formatação JSON

Embora as informações possam ser estruturadas e transferidas entre um cliente e um servidor de serviços da Web de várias maneiras, a opção mais popular é JavaScript Object Notation (JSON).

JSON é um padrão da indústria para representar estruturas de dados simples em texto simples e é usado para transferir informações de estado descrevendo os recursos. A API REST do SnapCenter usa JSON para formatar os dados transportados no corpo de cada solicitação e resposta HTTP.

## Caraterísticas operacionais básicas

Embora O REST estabeleça um conjunto comum de tecnologias e práticas recomendadas, os detalhes de cada API podem variar com base nas escolhas de design.

## Transação de API de solicitação e resposta

Cada chamada de API REST é executada como uma solicitação HTTP para o sistema servidor SnapCenter que gera uma resposta associada ao cliente. Esse par de solicitação e resposta é considerado uma transação de API.

Antes de usar a API, você deve estar familiarizado com as variáveis de entrada disponíveis para controlar uma solicitação e o conteúdo da saída de resposta.

## Suporte para operações CRUD

Cada um dos recursos disponíveis por meio da API REST do SnapCenter é acessado com base no modelo CRUD:

- Criar
- Leia

- Atualização
- Eliminar

Para alguns dos recursos, apenas um subconjunto das operações é suportado.

## Identificadores de objeto

Cada instância ou objeto de recurso recebe um identificador exclusivo quando é criado. Na maioria dos casos, o identificador é um UUID de 128 bits. Esses identificadores são globalmente exclusivos dentro de um servidor SnapCenter específico.

Depois de emitir uma chamada de API que cria uma nova instância de objeto, um URL com a ID associada é retornado ao chamador no cabeçalho do local da resposta HTTP. Você pode extrair o identificador e usá-lo em chamadas subseqüentes quando se refere à instância de recurso.



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

## Instâncias e coleções de objetos

Dependendo do caminho do recurso e do método HTTP, uma chamada de API pode ser aplicada a uma instância de objeto específica ou a uma coleção de objetos.

## Operações síncronas e assíncronas

O SnapCenter executa uma solicitação HTTP recebida de um cliente de forma síncrona ou assíncrona.

### Processamento síncrono

O SnapCenter executa a solicitação imediatamente e responde com um código de status HTTP de 200 ou 201 se for bem-sucedido.

Cada solicitação usando o método GET é sempre realizada de forma síncrona. Além disso, as solicitações que usam POST são projetadas para serem executadas de forma síncrona, caso se espera que sejam concluídas em menos de dois segundos.

### Processamento assíncrono

Se uma solicitação assíncrona for válida, o SnapCenter criará uma tarefa em segundo plano para processar a solicitação e um objeto de tarefa para ancorar a tarefa. O código de status HTTP 202 é retornado ao chamador juntamente com o objeto de tarefa. Você deve recuperar o estado do trabalho para determinar o sucesso ou a falha.

As solicitações que usam os métodos POST e DELETE são projetadas para serem executadas de forma assíncrona se espera que levem mais de dois segundos para serem concluídas.

## Segurança

A segurança fornecida com a API REST é baseada principalmente nos recursos de segurança existentes disponíveis com o SnapCenter. A seguinte segurança é usada pela API:

## Segurança da camada de transporte

Todo o tráfego enviado pela rede entre o servidor SnapCenter e o cliente geralmente é criptografado usando TLS, com base nas configurações do SnapCenter.

## Autenticação HTTP

Em um nível HTTP, a autenticação básica é usada para as transações da API. Um cabeçalho HTTP com o nome de usuário e senha em uma cadeia de caracteres base64 é adicionado a cada solicitação.

# Variáveis de entrada que controlam uma solicitação de API

Você pode controlar como uma chamada de API é processada através de parâmetros e variáveis definidos na solicitação HTTP.

## Métodos HTTP

Os métodos HTTP suportados pela API REST do SnapCenter são mostrados na tabela a seguir.



Nem todos os métodos HTTP estão disponíveis em cada um dos pontos finais REST.

Método HTTP	Descrição
OBTER	Recupera propriedades de objeto em uma instância ou coleção de recursos.
POST	Cria uma nova instância de recurso com base na entrada fornecida.
ELIMINAR	Exclui uma instância de recurso existente.
COLOQUE	Modifica uma instância de recurso existente.

## Cabeçalhos de solicitação

Você deve incluir vários cabeçalhos na solicitação HTTP.

### Tipo de conteúdo

Se o corpo da solicitação incluir JSON, esse cabeçalho deve ser definido como *application/json*.

### Aceitar

Esse cabeçalho deve ser definido como *application/json*.

### Autorização

A autenticação básica deve ser definida com o nome de usuário e senha codificados como uma string base64.

## Corpo do pedido

O conteúdo do corpo da solicitação varia de acordo com a chamada específica. O corpo da solicitação HTTP consiste em um dos seguintes:

- Objeto JSON com variáveis de entrada
- Vazio

## Filtrando objetos

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

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

Além de uma correspondência exata, outros operadores estão disponíveis para retornar um conjunto de objetos em uma faixa de valores. A API REST do SnapCenter suporta os operadores de filtragem mostrados na tabela abaixo.

Operador	Descrição
.	Igual a.
*	Menos de
>	Superior a.
O que é que eu tenho	Inferior ou igual a
>	Maior ou igual a
ATUALIZAÇÃO	Ou
!	Não é igual a
*	Wildcard ganancioso

Você também pode retornar uma coleção de objetos com base se um campo específico está definido ou não usando a palavra-chave **null** ou sua negação **!null** como parte da consulta.



Todos os campos que não estão definidos são geralmente excluídos de consultas correspondentes.

## Solicitando campos de objeto específicos

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

### Campos comuns ou padrão

Especifique **campos\*** para recuperar os campos de objeto mais comumente usados. Esses campos são normalmente mantidos na memória do servidor local ou requerem pouco processamento para acessar. Estas são as mesmas propriedades retornadas para um objeto depois de usar GET com uma chave de caminho de URL (UUID).

### Todos os campos

Especifique **campos\*\*** para recuperar todos os campos de objeto, incluindo aqueles que exigem processamento adicional de servidor para acessar.

## Seleção de campo personalizada

Use **fields** <field\_name> para especificar o campo exato desejado. Ao solicitar vários campos, os valores devem ser separados usando vírgulas sem espaços.



Como prática recomendada, você deve sempre identificar os campos específicos que deseja. Você só deve recuperar o conjunto de campos comuns ou todos os campos quando necessário. Quais campos são classificados como comuns e retornados usando *fields\**, são determinados pelo NetApp com base na análise interna de desempenho. A classificação de um campo pode mudar em versões futuras.

## 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 `order_by` parâmetro de consulta com o nome do campo e a direção de classificação da seguinte forma:

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

Por exemplo, você pode classificar o campo tipo em ordem decrescente seguido de id em ordem crescente:

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

- Se você especificar um campo de classificação, mas não fornecer uma direção, os valores serão classificados em ordem crescente.
- Ao incluir vários parâmetros, você deve separar os campos com uma vírgula.

## Paginação ao recuperar objetos em uma coleção

Ao emitir uma chamada de API usando GET para acessar uma coleção de objetos do mesmo tipo, o SnapCenter tenta retornar o maior número possível de objetos com base em duas restrições. Você pode controlar cada uma dessas restrições usando parâmetros de consulta adicionais na solicitação. A primeira restrição alcançada para uma SOLICITAÇÃO GET específica termina a solicitação e, portanto, limita o número de Registros retornados.



Se uma solicitação terminar antes de iterar todos os objetos, a resposta conterá o link necessário para recuperar o próximo lote de Registros.

## Limitando o número de objetos

Por padrão, o SnapCenter retorna um máximo de 10.000 objetos para uma SOLICITAÇÃO GET. Você pode alterar esse limite usando o parâmetro de consulta *Max\_Records*. Por exemplo:

```
max_records=20
```

O número de objetos realmente retornados pode ser menor do que o máximo em efeito, com base na restrição de tempo relacionada, bem como o número total de objetos no sistema.

## Limitar o tempo usado para recuperar os objetos

Por padrão, o SnapCenter retorna o maior número possível de objetos dentro do tempo permitido para a solicitação GET. O tempo limite padrão é de 15 segundos. Você pode alterar esse limite usando o parâmetro



de consulta `return_timeout`. Por exemplo:

```
return_timeout=5
```

O número de objetos realmente retornados pode ser menor que o máximo em efeito, com base na restrição relacionada ao número de objetos, bem como o número total de objetos no sistema.

### Estreitar o conjunto de resultados

Se necessário, você pode combinar esses dois parâmetros com parâmetros de consulta adicionais para restringir o conjunto de resultados. Por exemplo, o seguinte retorna até 10 eventos EMS gerados após o tempo especificado:

```
time⇒ 2018-04-04T15:41:29.140265Z&max_records=10
```

Você pode emitir várias solicitações para percorrer os objetos. Cada chamada de API subsequente deve usar um novo valor de tempo com base no evento mais recente no último conjunto de resultados.

### Propriedades de tamanho

Os valores de entrada usados com algumas chamadas de API, bem como certos parâmetros de consulta são numéricos. Em vez de fornecer um inteiro em bytes, você pode opcionalmente usar um sufixo como mostrado na tabela a seguir.

Sufixo	Descrição
KB	KB kilobytes (1024 bytes) ou kibibytes
MB	MB megabytes (KB x 1024 bytes) ou megabytes
GB	GB Gigabytes (MB x 1024 bytes) ou gibytes
TB	TB Terabytes (GB x 1024 bytes) ou tebibytes
PB	PB petabytes (TB x 1024 bytes) ou petabytes

## Interpretação de uma resposta API

Cada solicitação de API gera uma resposta de volta ao cliente. Você deve examinar a resposta para determinar se ela foi bem-sucedida e recuperar dados adicionais, conforme necessário.

### Código de status HTTP

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

Código	Descrição
200	OK indica sucesso para chamadas que não criam um novo objeto.
201	Criado um objeto foi criado com sucesso. O cabeçalho de localização na resposta inclui o identificador exclusivo para o objeto.

<b>Código</b>	<b>Descrição</b>
202	Aceite Um trabalho em segundo plano foi iniciado para executar a solicitação, mas ainda não foi concluído.
400	Solicitação incorreta a entrada de solicitação não é reconhecida ou é inadequada.
401	A autenticação de utilizador não autorizado falhou.
403	O acesso proibido é negado devido a um erro de autorização (RBAC).
404	Não encontrado o recurso referido na solicitação não existe.
405	Método não permitido o método HTTP na solicitação não é suportado para o recurso.
409	Conflito uma tentativa de criar um objeto falhou porque um objeto diferente deve ser criado primeiro ou o objeto solicitado já existe.
500	Erro interno ocorreu Um erro interno geral no servidor.

## Cabeçalhos de resposta

Vários cabeçalhos estão incluídos na resposta HTTP gerada pelo SnapCenter.

### Localização

Quando um objeto é criado, o cabeçalho do local inclui o URL completo para o novo objeto, incluindo o identificador exclusivo atribuído ao objeto.

### Tipo de conteúdo

Isso normalmente será `application/json`.

## Corpo de resposta

O conteúdo do corpo de resposta resultante de uma solicitação de API difere com base no objeto, no tipo de processamento e no sucesso ou falha da solicitação. A resposta é sempre renderizada em JSON.

### Um único objeto

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

Vários objetos de uma coleção de recursos podem ser retornados. Em todos os casos, há um formato consistente usado, com `num_records` a indicação do número de Registros e Registros contendo um array das instâncias do objeto. Por exemplo, você pode recuperar os nós definidos em um cluster específico.

## Objeto trabalho

Se uma chamada de API for processada de forma assíncrona, um objeto Job será retornado que ancora a tarefa em segundo plano. Por exemplo, a SOLICITAÇÃO DE PATCH usada para atualizar a configuração do cluster é processada de forma assíncrona e retorna um objeto Job.

## Objeto de erro

Se ocorrer um erro, um objeto de erro é sempre retornado. Por exemplo, você receberá um erro ao tentar alterar um campo não definido para um cluster.

## Vazio

Em certos casos, nenhum dado é retornado e o corpo de resposta inclui um objeto JSON vazio.

## Erros

Se ocorrer um erro, um objeto de erro é retornado no corpo de resposta.

## Formato

Um objeto de erro tem o seguinte formato:

```
"error": {  
  "message": "<string>",  
  "code": <integer>[,  
  "target": "<string>"]  
}
```

Você pode usar o valor do código para determinar o tipo ou categoria de erro geral e a mensagem para determinar o erro específico. Quando disponível, o campo de destino inclui a entrada específica do usuário associada ao erro.

## Códigos de erro comuns

Os códigos de erro comuns são descritos na tabela a seguir. As chamadas de API específicas podem incluir códigos de erro adicionais.

Código	Descrição
409	Já existe um objeto com o mesmo identificador.
400	O valor de um campo tem um valor inválido ou está em falta ou um campo extra foi fornecido.
400	A operação não é suportada.
405	Não é possível encontrar um objeto com o identificador especificado.
403	A permissão para executar a solicitação é negada.
409	O recurso está em uso.

# APIs REST compatíveis com servidor SnapCenter e plug-ins

Os recursos disponíveis por meio da API REST do SnapCenter são organizados em categorias, conforme exibido na página de documentação da API do SnapCenter. Uma breve descrição de cada um dos recursos com os caminhos de recursos básicos é apresentada abaixo, juntamente com considerações de uso adicionais, quando apropriado.

## Auth

Você pode usar essa API para fazer login no servidor SnapCenter. Esta API retorna um token de autorização de usuário que é usado para autenticar solicitações subsequentes.

## Domínios

Você pode usar APIs para executar operações diferentes.

- Recupere todos os domínios no SnapCenter
- recuperar detalhes de um domínio específico
- registrar ou cancelar o registro de um domínio
- modificar um domínio

## Trabalhos

Você pode usar APIs para executar operações diferentes.

- Recupere todos os trabalhos no SnapCenter
- recuperar o status de um trabalho
- cancelar ou parar um trabalho

## Definições

Você pode usar APIs para executar operações diferentes.

- registre, modifique ou remova uma credencial
- Exibe as informações de credenciais registradas no servidor SnapCenter
- configure as definições de notificação
- Recupera informações sobre o servidor SMTP atualmente configurado para enviar notificações por e-mail e exibe o nome do servidor SMTP, o nome dos destinatários e o nome do remetente
- Exibe a configuração de autenticação multifator (MFA) do login do servidor SnapCenter
- Ative ou desative e configure o MFA para o login do servidor SnapCenter
- Crie o arquivo de configuração necessário para configurar o MFA

## Hosts

Você pode usar APIs para executar operações diferentes.

- Consultar todos os hosts SnapCenter
- Remova um ou mais hosts do SnapCenter
- recuperar um host pelo nome
- recuperar todos os recursos em um host
- Recuperar um recurso usando o ID do recurso
- recupere os detalhes de configuração do plug-in
- configure o host do plug-in
- Recuperar todos os recursos do plug-in para o host Microsoft SQL Server
- Recuperar todos os recursos do plug-in para o host de banco de dados Oracle
- recupere todos os recursos do plug-in para o host de aplicativos personalizados
- Recuperar todos os recursos do plug-in para host SAP HANA
- recupere os plug-ins instalados
- instale plug-ins em um host existente
- atualize o pacote de host
- remova plug-ins de um host existente
- adicione plug-in em um host
- adicionar ou modificar host
- Obtenha a assinatura do host Linux
- Registre a assinatura do host Linux
- coloque o host no modo de manutenção ou produção
- inicie ou reinicie os serviços de plug-in no host
- renomeie um host

## Recursos

Você pode usar APIs para executar operações diferentes.

- recuperar todos os recursos
- Recuperar um recurso usando o ID do recurso
- Recuperar todos os recursos do plug-in para o host Microsoft SQL Server
- Recuperar todos os recursos do plug-in para o host de banco de dados Oracle
- recupere todos os recursos do plug-in para o host de aplicativos personalizados
- Recuperar todos os recursos do plug-in para host SAP HANA
- Recuperar um recurso do Microsoft SQL Server usando uma chave
- recuperar um recurso personalizado usando uma chave
- modifique um recurso do plug-in para o host de aplicativos personalizados

- remova um recurso do plug-in para o host de aplicativos personalizados usando uma chave
- Recuperar um recurso do SAP HANA usando uma chave
- Modificar um recurso do plug-in para host SAP HANA
- Remover um recurso do plug-in para host SAP HANA usando uma chave
- Recuperar um recurso Oracle usando uma chave
- Criar um recurso de volume de aplicativos Oracle
- Modificar um recurso de volume de aplicativos Oracle
- Remova um recurso de volume de aplicativos Oracle usando uma chave
- Recuperar os detalhes secundários do recurso Oracle
- Faça backup do recurso Microsoft SQL Server usando plug-in para Microsoft SQL Server
- Faça backup do recurso Oracle usando plug-in para banco de dados Oracle
- faça backup do recurso personalizado usando o plug-in para aplicativo personalizado
- Configurar o banco de dados SAP HANA
- Configure o banco de dados Oracle
- Restaurar um backup de banco de dados SQL
- Restaurar um backup de banco de dados Oracle
- restaure um backup de aplicativo personalizado
- crie um recurso de plug-in personalizado
- Criar um recurso do SAP HANA
- proteja um recurso personalizado usando o plug-in para aplicativos personalizados
- Proteja um recurso do Microsoft SQL Server usando o plug-in para Microsoft SQL Server
- Modifique um recurso protegido do Microsoft SQL Server
- Remova a proteção do recurso Microsoft SQL Server
- Proteger um recurso Oracle usando plug-in para banco de dados Oracle
- Modificar um recurso Oracle protegido
- Remova a proteção do recurso Oracle
- clonar um recurso do backup usando o plug-in para aplicação personalizada
- Clonar um volume de aplicações Oracle a partir do backup usando o plug-in para banco de dados Oracle
- Clonar um recurso do Microsoft SQL Server a partir do backup usando o plug-in para Microsoft SQL Server
- Crie um ciclo de vida clone de um recurso do Microsoft SQL Server
- Modifique o ciclo de vida do clone de um recurso do Microsoft SQL Server
- Excluir ciclo de vida de clone de um recurso do Microsoft SQL Server
- Mova um banco de dados existente do Microsoft SQL Server de um disco local para um LUN NetApp
- Crie um arquivo de especificação clone para um banco de dados Oracle
- Inicie um trabalho de atualização de clone sob demanda de um recurso Oracle
- Crie um recurso Oracle a partir do backup usando o arquivo de especificação clone

- restaura o banco de dados para a réplica secundária e junta o banco de dados de volta ao grupo de disponibilidade
- Criar um recurso de volume de aplicativos Oracle

## Backups

Você pode usar APIs para executar operações diferentes.

- recuperar detalhes da cópia de segurança por nome, tipo, plug-in, recurso ou data da cópia de segurança
- recuperar todos os backups
- recuperar detalhes da cópia de segurança
- renomear ou excluir backups
- Montar um backup Oracle
- Desmontar um backup Oracle
- catalogue um backup Oracle
- Descatalogar um backup Oracle
- obtenha todos os backups necessários para serem montados para executar a recuperação pontual

## Clones

Você pode usar APIs para executar operações diferentes.

- Crie, exiba, modifique e exclua o arquivo de especificação do clone do banco de dados Oracle
- Exibir hierarquia de clones de banco de dados Oracle
- recuperar detalhes do clone
- recuperar todos os clones
- excluir clones
- Recuperar detalhes do clone por ID
- Inicie um trabalho de atualização de clone sob demanda de um recurso Oracle
- Clonar um recurso Oracle a partir do backup usando o arquivo de especificação clone

## Divisão de clones

Você pode usar APIs para executar operações diferentes.

- estime a operação de divisão do clone do recurso clonado
- recuperar o status de uma operação de divisão de clones
- inicie ou pare uma operação de divisão de clones

## Grupos de recursos

Você pode usar APIs para executar operações diferentes.

- recuperar detalhes de todos os grupos de recursos
- recuperar o grupo de recursos por nome

- crie um grupo de recursos para plug-in para aplicativos personalizados
- Crie um grupo de recursos para plug-in para Microsoft SQL Server
- Criar um grupo de recursos para plug-in para banco de dados Oracle
- modifique um grupo de recursos para plug-in para aplicativos personalizados
- Modificar um grupo de recursos para plug-in para Microsoft SQL Server
- Modificar um grupo de recursos para plug-in para banco de dados Oracle
- Crie, modifique ou exclua o ciclo de vida do clone de um grupo de recursos para plug-in para Microsoft SQL Server
- faça backup de um grupo de recursos
- coloque o grupo de recursos no modo de manutenção ou produção
- remover um grupo de recursos

## Políticas

Você pode usar APIs para executar operações diferentes.

- recuperar detalhes da política
- recuperar detalhes da política por nome
- eliminar uma política
- crie uma cópia de uma política existente
- criar ou modificar política para plug-in para aplicação personalizada
- Criar ou modificar política para plug-in para Microsoft SQL Server
- Criar ou modificar política para plug-in para banco de dados Oracle
- Criar ou modificar política de plug-in para banco de dados SAP HANA

## Armazenamento

Você pode usar APIs para executar operações diferentes.

- recuperar todos os compartilhamentos
- recuperar um compartilhamento pelo nome
- criar ou excluir um compartilhamento
- recuperar detalhes de armazenamento
- recuperar detalhes de armazenamento por nome
- criar, modificar ou excluir um armazenamento
- descubra recursos em um cluster de storage
- recuperar recursos em um cluster de armazenamento

## Partilhar

Você pode usar APIs para executar operações diferentes.

- recuperar os detalhes de um compartilhamento



- recuperar detalhes de todas as ações
- crie ou exclua um compartilhamento no armazenamento
- recuperar um compartilhamento pelo nome

## Plugins

Você pode usar APIs para executar operações diferentes.

- listar todos os plug-ins de um host
- Recuperar um recurso do Microsoft SQL Server usando uma chave
- modifique um recurso personalizado usando uma chave
- remova um recurso personalizado usando uma chave
- Recuperar um recurso do SAP HANA usando uma chave
- Modificar um recurso do SAP HANA usando uma chave
- Remover um recurso do SAP HANA usando uma chave
- Recuperar um recurso Oracle usando uma chave
- Modificar um recurso de volume de aplicativos Oracle usando uma chave
- Remova um recurso de volume de aplicativos Oracle usando uma chave
- Faça backup do recurso Microsoft SQL Server usando plug-in para Microsoft SQL Server e uma chave
- Faça backup do recurso Oracle usando plug-in para banco de dados Oracle e uma chave
- faça backup do recurso de aplicativo personalizado usando o plug-in para aplicativo personalizado e uma chave
- Configurar o banco de dados SAP HANA usando uma chave
- Configure o banco de dados Oracle usando uma chave
- restaure um backup de aplicativo personalizado usando uma chave
- crie um recurso de plug-in personalizado
- Criar um recurso do SAP HANA
- Criar um recurso de volume de aplicativos Oracle
- proteja um recurso personalizado usando o plug-in para aplicativos personalizados
- Proteja um recurso do Microsoft SQL Server usando o plug-in para Microsoft SQL Server
- Modifique um recurso protegido do Microsoft SQL Server
- Remova a proteção do recurso Microsoft SQL Server
- Proteger um recurso Oracle usando plug-in para banco de dados Oracle
- Modificar um recurso Oracle protegido
- Remova a proteção do recurso Oracle
- clonar um recurso do backup usando o plug-in para aplicação personalizada
- Clonar um volume de aplicações Oracle a partir do backup usando o plug-in para banco de dados Oracle
- Clonar um recurso do Microsoft SQL Server a partir do backup usando o plug-in para Microsoft SQL Server
- Crie um ciclo de vida clone de um recurso do Microsoft SQL Server

- Modifique o ciclo de vida do clone de um recurso do Microsoft SQL Server
- Excluir ciclo de vida de clone de um recurso do Microsoft SQL Server
- Crie um arquivo de especificação clone para um banco de dados Oracle
- Inicie um ciclo de vida do clone sob demanda de um recurso Oracle
- Clonar um recurso Oracle a partir do backup usando o arquivo de especificação clone

## Relatórios

Você pode usar APIs para executar operações diferentes.

- recupere relatórios de operações de backup, restauração e clone para os respectivos plug-ins
- adicionar, executar, excluir ou modificar programações
- recuperar dados para os relatórios programados

## Alertas

Você pode usar APIs para executar operações diferentes.

- recuperar todos os alertas
- Recuperar alertas por IDs
- Excluir vários alertas ou excluir um alerta por ID

## RBAC

Você pode usar APIs para executar operações diferentes.

- recupere detalhes de usuários, grupos e funções
- adicionar ou excluir usuários
- atribuir utilizador à função
- anular a atribuição do utilizador da função
- criar, modificar ou excluir funções
- atribuir grupo a uma função
- anular a atribuição de um grupo de uma função
- adicionar ou excluir grupos
- crie uma cópia de uma função existente
- atribuir ou anular a atribuição de recursos ao utilizador ou grupo

## Configuração

Você pode usar APIs para executar operações diferentes.

- ver as definições de configuração
- modifique as definições de configuração

## CertificateSettings

Você pode usar APIs para executar operações diferentes.

- Exibir o status do certificado para o servidor SnapCenter ou host de plug-in
- Modifique as configurações de certificado para o servidor SnapCenter ou host de plug-in

## Repositório

Você pode usar APIs para executar operações diferentes.

- recupere os backups do repositório
- veja as informações de configuração sobre o repositório
- Proteja e restaure o repositório do SnapCenter
- Desproteger o repositório SnapCenter
- reconstruir e fazer failover no repositório

## Versão

Você pode usar essa API para exibir a versão do SnapCenter.

## Como acessar APIs REST usando a página da Web da API Swagger

As APIs REST são expostas por meio da página da Web do Swagger. Você pode acessar a página da Web do Swagger para exibir as APIs REST do servidor SnapCenter, bem como emitir manualmente uma chamada de API. Você pode usar APIs REST para ajudar a gerenciar seu servidor SnapCenter ou para executar operações de proteção de dados.

Você deve saber o endereço IP de gerenciamento ou o nome de domínio do servidor SnapCenter no qual deseja executar as APIs REST.

Você não precisa de permissões especiais para executar o cliente API REST. Qualquer usuário pode acessar a página da Web do Swagger. As respectivas permissões nos objetos que são acessados através da API REST são baseadas no usuário que gera o token para fazer login na API REST.

### Passos

1. Em um navegador, digite o URL para acessar a página da Web do Swagger no formato `_ https:// SnapCenter_IP_address_or_name>: SnapCenter_port>/swagger/_`.



Certifique-se de que o URL da API REST não tem os seguintes caracteres: `.`, `%` e `&`.

2. No campo **exploração do Swagger**, se a documentação da API Swagger não for exibida automaticamente, digite: `_ https:// SnapCenter_IP_address_or_name>: SnapCenter_port>/content/swagger/SnapCenter.yaml_`
3. Clique em **explorar**.

Uma lista de tipos ou categorias de recursos da API é exibida.

4. Clique em um tipo de recurso de API para exibir as APIs nesse tipo de recurso.

Se você encontrar um comportamento inesperado ao executar APIs REST do SnapCenter, poderá usar os arquivos de log para identificar a causa e resolver o problema. Pode transferir os ficheiros de registo a partir da interface de utilizador do SnapCenter clicando em **Monitor > Registos > Download**.

## Comece a usar a API REST

Você pode começar rapidamente a usar a API REST do SnapCenter. Acessar a API fornece alguma perspectiva antes de começar a usá-la com os processos de fluxo de trabalho mais complexos em uma configuração ao vivo.

### Olá mundo

Você pode executar um comando simples em seu sistema para começar a usar a API REST do SnapCenter e confirmar sua disponibilidade.

#### Antes de começar

- Certifique-se de que o utilitário Curl está disponível no seu sistema.
- Endereço IP ou nome de host do servidor SnapCenter
- Nome de usuário e senha de uma conta com autoridade para acessar a API REST do SnapCenter.



Se suas credenciais incluem caracteres especiais, você precisa formatá-los de uma forma aceitável para Curl com base no shell que você está usando. Por exemplo, você pode inserir uma barra invertida antes de cada caractere especial ou envolver toda `username:password` a cadeia em aspas simples.

#### Passo

Na interface da linha de comando, execute o seguinte para recuperar as informações do plug-in:

```
curl -X GET -u username:password -k  
"https://<ip_address>/api/hosts?fields=IncludePluginInfo"
```

Exemplo:

```
curl -X GET -u admin:password -k  
"'https://10.225.87.97/api/hosts?fields=IncludePluginInfo'"
```

## Informações sobre direitos autorais

Copyright © 2024 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 DESTES DOCUMENTOS. 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 SOBRESSALIENTES; 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 DESTES SOFTWARES, 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.