



# **REST APIを使用したSnapCenterの自動化**

## SnapCenter software

NetApp  
November 06, 2025

# 目次

REST APIを使用したSnapCenterの自動化	1
REST APIを使用したSnapCenterの自動化	1
SnapCenter REST APIヘネイティブにアクセスする方法	1
基盤としてのREST Webサービス	1
リソースと状態の表示	1
URIエンドポイント	2
HTTPメッセージ	2
JSONの形式	2
基本的な動作の特性	2
要求と応答のAPIトランザクション	2
CRUD操作のサポート	2
オブジェクトID	3
オブジェクトのインスタンスとコレクション	3
同期操作と非同期操作	3
セキュリティ	3
API要求を制御する入力変数	4
HTTPメソッド	4
要求ヘッダー	4
リクエスト本文	5
オブジェクトのフィルタ	5
特定のオブジェクトフィールドの要求	5
出力セット内のオブジェクトのソート	6
オブジェクトのコレクションを取得するときのページ付け	6
サイズのプロパティ	7
API応答の解釈	7
HTTPステータス コード	7
応答ヘッダー	8
応答の本文	8
エラー	9
SnapCenter ServerとプラグインでサポートされているREST API	10
認証	10
ドメイン	10
ジョブ	10
設定	10
ホスト	11
リソース	11
バックアップ	13
クローン	13
クローン スプリット	14

リソース グループ	14
ポリシー	14
ストレージ	15
共有	15
プラグイン	15
レポート	16
アラート	16
RBAC	17
構成	17
証明書設定	17
リポジトリ	17
version	17
Swagger API WebページからREST APIにアクセスする方法	18
REST APIの使用の開始	18
Hello world	18

# REST APIを使用したSnapCenterの自動化

## REST APIを使用したSnapCenterの自動化

REST API を使用して、いくつかのSnapCenter管理操作を実行できます。REST API はSwagger Webページから利用できます。

[https://<SnapCenter\\_IP\\_address\\_or\\_name>:<SnapCenter\\_port>/swagger/](https://<SnapCenter_IP_address_or_name>:<SnapCenter_port>/swagger/) にある Swagger Web ページにアクセスして、REST API ドキュメントを表示したり、API 呼び出しを手動で発行したりできます。

REST APIをサポートするプラグインは次のとおりです。

- Plug-in for Microsoft SQL Server
- Plug-in for SAP HANA Database
- Plug-in for Oracle Database

SnapCenter Plug-in for VMware vSphereの詳細については、以下を参照してください。 "["SnapCenter Plug-in for VMware vSphere"](#)"

## SnapCenter REST APIへネイティブにアクセスする方法

SnapCenter REST APIには、RESTクライアントをサポートする任意のプログラミング言語を使用して直接アクセスできます。よく使われる言語には、Python、PowerShell、Javaなどがあります。

## 基盤としてのREST Webサービス

Representational State Transfer (REST) は、分散Webアプリケーションの作成に使用される形式です。WebサービスAPIの設計においては、これによってサーバベースのリソースの公開とその状態の管理に関する一連のテクノロジとベストプラクティスが確立されます。主流のプロトコルと標準が使用されており、SnapCenterを管理するための柔軟な基盤が提供されます。

### リソースと状態の表示

リソースはWebベース システムの基本コンポーネントです。REST Webサービス アプリケーションを作成する場合、設計の早い段階で次の作業を行います。

#### システムまたはサーバベースのリソースの識別

すべてのシステムは、リソースを使用し、管理します。リソースには、ファイル、ビジネス トランザクション、プロセス、管理エンティティなどがあります。REST Webサービスに基づいてアプリケーションを設計する際に行う最初の作業の1つは、リソースを識別することです。

## リソースの状態の定義および関連する状態処理

リソースは必ず複数ある状態のいずれかに該当します。状態は明確に定義する必要があり、状態の変化に作用する操作も明確に定義する必要があります。

## URIエンドポイント

すべてのRESTリソースは、明確に定義されたアドレス指定方式を使用して定義、提供される必要があります。リソースが置かれているエンドポイントは、Uniform Resource Identifier (URI) で識別されます。

URIは、ネットワークの各リソースに一意の名前を作成するための一般的なフレームワークです。Uniform Resource Locator (URL) は、リソースを識別してアクセスするためにWebサービスで使用されるURIの一種です。リソースは、通常、ファイルディレクトリに似た階層構造で公開されます。

## HTTPメッセージ

Hypertext Transfer Protocol (HTTP) は、Webサービスのクライアントとサーバがリソースに関する要求と応答のメッセージを交換する際に使用するプロトコルです。

Webサービスアプリケーションの設計の一環として、リソースおよび対応する状態管理アクションにHTTPメソッドがマッピングされます。HTTPはステートレスです。したがって、関連する一連の要求と応答を1つのトランザクションの一部として関連付けるには、要求と応答のデータフローで伝送されるHTTPヘッダーに追加情報を含める必要があります。

## JSONの形式

Webサービスのクライアントとサーバの間で情報を構造化したり転送したりする方法は複数ありますが、最も広く使用されているのはJavaScript Object Notation (JSON) です。

JSONは、単純なデータ構造をプレーンテキストで表現するための業界標準であり、リソースについての状態情報の転送に使用されます。SnapCenter REST APIでは、JSONを使用して、各HTTP要求と応答の本文で伝送されるデータをフォーマットします。

## 基本的な動作の特性

RESTで共通のテクノロジとベストプラクティスは確立されますが、各APIの詳細は設計内容に応じて異なる場合があります。

## 要求と応答のAPIトランザクション

すべてのREST API呼び出しは、SnapCenter ServerシステムへのHTTP要求として実行され、クライアントへの関連する応答が生成されます。この要求と応答のペアでAPIトランザクションが構成されます。

APIを使用する前に、要求の制御に使用できる入力変数と応答出力の内容を理解しておく必要があります。

## CRUD操作のサポート

SnapCenter REST APIで使用できる各リソースへのアクセスはCRUDモデルに基づきます。

- 作成

- ・読み取り
- ・更新
- ・削除

リソースによっては、一部の操作のみがサポートされるものもあります。

## オブジェクトID

リソース インスタンスまたはオブジェクトには、作成時に一意の識別子がそれぞれ割り当てられます。ほとんどの場合、識別子は128ビットのUUIDです。これらの識別子は、特定のSnapCenter Server内でグローバルに一意です。

新しいオブジェクト インスタンスを作成するAPI呼び出しを発行すると、関連付けられたIDを含むURLがHTTP応答のlocationヘッダーに記載されて呼び出し元に返されます。リソース インスタンスを以降の呼び出しで参照する際は、この識別子を抽出して使用できます。



オブジェクト識別子の内容と内部構造は変更になることがあります。識別子を使用するのは、該当するAPI呼び出しで関連付けられているオブジェクトを参照するために必要な場合だけにしてください。

## オブジェクトのインスタンスとコレクション

リソース パスとHTTPメソッドに応じて、API呼び出しを特定のオブジェクト インスタンスまたはオブジェクトのコレクションに適用できます。

### 同期操作と非同期操作

SnapCenterは、クライアントから受信したHTTP要求を同期または非同期で実行します。

#### 同期処理

SnapCenterは要求をすぐに実行し、HTTPステータス コード200または201（成功した場合）を返します。

GETメソッドを使用する要求はすべて、常に同期的に実行されます。また、POSTを使用する要求は、完了までに2秒かからないと予想される場合は同期的に実行するように設計されています。

#### 非同期処理

非同期要求が有効な場合、SnapCenterは、要求を処理するバックグラウンド タスクとそのタスクのアンカーを設定するジョブ オブジェクトを作成します。HTTPステータス コード202がジョブ オブジェクトとともに呼び出し元に返されます。成功したかどうかを判断するには、ジョブの状態を取得する必要があります。

POSTとDELETEの各メソッドを使用する要求は、完了までに2秒以上かかると予想される場合は非同期で実行するように設計されています。

## セキュリティ

REST APIのセキュリティは、主にSnapCenterで利用可能な既存のセキュリティ機能に基づいています。APIで使用されるセキュリティは次のとおりです。

## Transport Layer Security

SnapCenter Serverとクライアントの間でネットワークを介して送信されるトラフィックは、SnapCenter構成の設定に基づいて、通常はTLSを使用してすべて暗号化されます。

### HTTP認証

HTTPレベルでは、APIトランザクションにベーシック認証が使用されます。base64文字列のユーザ名とパスワードを含むHTTPヘッダーが各要求に追加されます。

## API要求を制御する入力変数

API呼び出しの処理方法は、HTTP要求で設定されたパラメータと変数を使用して制御できます。

### HTTPメソッド

次の表に、SnapCenter REST APIでサポートされるHTTPメソッドを示します。



RESTエンドポイントのそれぞれですべてのHTTPメソッドを使用できるわけではありません。

HTTPメソッド	説明
GET	リソース インスタンスまたはコレクションのオブジェクトのプロパティを取得します。
POST	指定された入力に基づいて新しいリソース インスタンスを作成します。
DELETE	既存のリソース インスタンスを削除します。
PUT	既存のリソース インスタンスを変更します。

### 要求ヘッダー

HTTP要求には複数のヘッダーを含める必要があります。

#### Content-type

リクエスト本文に JSON が含まれている場合、このヘッダーは *application/json* に設定する必要があります。

#### 受け入れる

このヘッダーは *application/json* に設定する必要があります。

#### 承認

base64文字列としてエンコードされたユーザ名とパスワードを使用するベーシック認証を設定する必要があります。

## リクエスト本文

要求の本文の内容は、それぞれの呼び出しに応じて異なります。HTTP要求の本文は、次のいずれかで構成されます。

- JSONオブジェクトと入力変数
- 空の

## オブジェクトのフィルタ

GETを使用するAPI呼び出しを発行する際、返されるオブジェクトを任意の属性に基づいて制限またはフィルタできます。たとえば、一致する完全な値を指定できます。

<field>=<query value>

完全一致に加えて、他の演算子を使用して、特定の範囲の値の一連のオブジェクトを返すことができます。次の表に、SnapCenter REST APIでサポートされるフィルタ演算子を示します。

オペレーター	説明
=	等しい
<	小なり
>	より大きい
≤	以下
≥	より大きいか等しい
UPDATE	または
!	等しくない
*	すべてに一致するワイルドカード

クエリの一部として `null` キーワードまたはその否定 `!null` を使用することで、特定のフィールドが設定されているかどうかに基づいてオブジェクトのコレクションを返すこともできます。



通常、設定されていないフィールドはクエリの照合から除外されます。

## 特定のオブジェクト フィールドの要求

デフォルトでは、GETを使用してAPI呼び出しを実行すると、1つまたは複数のオブジェクトを一意に識別する属性のみが返されます。この最小のフィールド セットは各オブジェクトのキーとして機能し、オブジェクト タイプによって異なります。追加のオブジェクトプロパティを選択するには、`fields` クエリパラメータを次の方法で指定します。

### 共通または標準のフィールド

最もよく使用されるオブジェクト フィールドを取得するには、`fields=*` を指定します。通常、これらのフィールドはローカル サーバ メモリに保持されているか、ほとんど処理を必要とせずにアクセスできます。これらのプロパティは、URLパス キー (UUID) を指定してGETを使用した場合にオブジェクトに対して返されるプロパティと同じです。

## すべてのフィールド

**fields=\*\*** を指定すると、アクセスするために追加のサーバー処理が必要なものも含め、すべてのオブジェクト フィールドが取得されます。

## カスタム フィールド選択

必要なフィールドを正確に指定するには、**fields=<field\_name>** を使用します。複数のフィールドを要求する場合は、各値の間にスペースを入れずにカンマで区切る必要があります。

 ベストプラクティスとして、必要なフィールドを常に個別に指定することを推奨します。一連の共通のフィールドやすべてのフィールドを取得するのは、必要な場合だけにしてください。どのフィールドが共通として分類され、**fields=\*** を使用して返されるかは、NetAppによって内部パフォーマンス分析に基づいて決定されます。フィールドの分類は、今後のリリースで変更される可能性があります。

## 出力セット内のオブジェクトのソート

リソース コレクション内のレコードは、オブジェクトで定義されたデフォルトの順序で返されます。順序を変更するには、`order\_by` フィールド名と並べ替え方向を指定したクエリ パラメータは次のとおりです。

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

たとえば、次のように指定すると、type フィールドで降順にソートしてから、id で昇順にソートできます。

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

- ソートするフィールドを指定してソート順序を指定しなかった場合、値は昇順でソートされます。
- 複数のパラメータを指定する場合は、各フィールドをカンマで区切る必要があります。

## オブジェクトのコレクションを取得するときのページ付け

GETを使用するAPI呼び出しを発行して同じタイプのオブジェクトのコレクションにアクセスする場合、SnapCenterでは2つの制約に基づいて可能なかぎり多くのオブジェクトを返します。これらの各制約は、要求に対する追加のクエリ パラメータを使用して制御できます。それぞれのGET要求に対する最初の制約に達した時点で要求が終了され、結果として返されるレコードの数が制限されます。

 すべてのオブジェクトについての処理が完了する前に要求が終了した場合、次のレコードのバッチを取得するために必要なリンクが応答に含まれます。

## オブジェクト数の制限

デフォルトでは、SnapCenter はGET リクエストに対して最大 10,000 個のオブジェクトを返します。この制限は、**max\_records** クエリ パラメータを使用して変更できます。例えば：

```
max_records=20
```

実際に返されるオブジェクトの数は、関連する時間の制約やシステム内のオブジェクトの総数に基づいて、有効な最大数よりも少なくなることがあります。

## オブジェクトを取得する時間の制限

デフォルトでは、SnapCenterはGET要求に許可された時間の範囲内でできるだけ多くのオブジェクトを返します。デフォルトのタイムアウトは15秒です。この制限は、*return\_timeout* クエリ パラメータを使用して変更できます。例えば：

```
return_timeout=5
```

実際に返されるオブジェクトの数は、関連するオブジェクト数の制約やシステム内のオブジェクトの総数に基づいて、有効な最大数よりも少なくなることがあります。

## 結果セットの絞り込み

必要に応じて、これらの2つのパラメータを追加のクエリ パラメータと組み合わせて、結果セットを絞り込むことができます。たとえば、次の例では、指定した時間のあとに生成されたEMSイベントを最大10件まで返します。

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

複数の要求を発行してオブジェクトをページングすることができます。以降のAPI呼び出しでは、前回の結果セットの最後のイベントに基づいて新しい時間の値を使用する必要があります。

## サイズのプロパティ

一部のAPI呼び出しおよびクエリ パラメータでは、入力値として数値が使用されます。バイト単位で整数を指定する代わりに、必要に応じて次の表に示すサフィックスを使用することができます。

サフィックス	説明
KB	KB : キロバイト (1,024バイト) またはキビバイト
MB	MB : メガバイト (KB x 1,024バイト) またはメビバイト
GB	GB : ギガバイト (MB x 1,024バイト) またはギビバイト
TB	TB : テラバイト (GB x 1,024バイト) またはテビバイト
PB	PB : ペタバイト (TB x 1,024バイト) またはペビバイト

## API応答の解釈

各API要求でクライアントへの応答が生成されます。要求が成功したかどうかを確認したり、必要に応じて追加データを取得したりするには、応答を調べる必要があります。

## HTTPステータス コード

SnapCenter REST APIで使用されるHTTPステータス コードを次に示します。

コード	説明
200	OK 新しいオブジェクトを作成しない呼び出しが成功したことを示します。
201	作成されました オブジェクトが正常に作成されました。応答のlocationヘッダーにオブジェクトの一意の識別子が含まれます。
202	承認済み 要求を実行するためにバックグラウンドジョブが開始されましたが、まだ完了していません。
400	不正なリクエスト リクエストの入力が認識されないか不適切です。
401	不正なユーザー認証に失敗しました。
403	承認 (RBAC) エラーのため、禁止アクセスは拒否されました。
404	見つかりません。リクエストで参照されたリソースが存在しません。
405	メソッドは許可されていません。リクエスト内のHTTP メソッドはリソースではサポートされていません。
409	競合 別のオブジェクトを最初に作成する必要があるか、要求されたオブジェクトがすでに存在するため、オブジェクトの作成に失敗しました。
500	内部エラー サーバーで一般的な内部エラーが発生しました。

## 応答ヘッダー

SnapCenterによって生成されるHTTP応答には、いくつかのヘッダーが含まれています。

### Location

オブジェクトが作成されたときに、オブジェクトに割り当てられた一意の識別子を含む、新しいオブジェクトへの完全なURLが含まれます。

### Content-type

これは通常 application/json。

## 応答の本文

API要求の結果として返される応答の本文の内容は、オブジェクト、処理タイプ、および要求の成否によって異なります。応答は常にJSON形式になります。

### 単一のオブジェクト

単一のオブジェクトを要求に基づいて一連のフィールドとともに返すことができます。たとえば、GETで一意の識別子を使用して、クラスタの選択したプロパティを取得できます。

## 複数のオブジェクト

リソース コレクションから複数のオブジェクトを返すことができます。いずれの場合も、一貫した形式が使用されます。`num\_records` レコード数と、オブジェクトインスタンスの配列を含むレコード数を示します。たとえば、特定のクラスタで定義されているノードを取得できます。

## ジョブ オブジェクト

API呼び出しが非同期で処理されると、バックグラウンド タスクのアンカーを設定するジョブ オブジェクトが返されます。たとえば、クラスタ構成の更新に使用されるPATCH要求は非同期で処理され、ジョブ オブジェクトが返されます。

## エラー オブジェクト

エラーが発生した場合は、常にエラー オブジェクトが返されます。たとえば、クラスタに定義されていないフィールドを変更しようとするとエラーが返されます。

## 空の

場合によっては、データが返されず、応答の本文に空のJSONオブジェクトが含まれることがあります。

## エラー

エラーが発生した場合は、応答の本文でエラー オブジェクトが返されます。

## フォーマット

エラー オブジェクトの形式は次のとおりです。

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

codeの値で一般的なエラーの種類やカテゴリを特定し、messageで具体的なエラーの内容を確認できます。特定可能な場合は、エラーに関連するユーザ入力がtargetフィールドに含まれます。

## 一般的なエラー コード

次の表に一般的なエラー コードを示します。特定のAPI呼び出しについては、追加のエラー コードが含まれる場合があります。

コード	説明
409	同じ識別子のオブジェクトがすでに存在します。
400	フィールドの値が無効であるか、値が指定されていないか、不要なフィールドが指定されています。
400	この操作はサポートされていません。

コード	説明
405	指定した識別子を持つオブジェクトが見つかりません。
403	要求を実行する権限が拒否されました。
409	リソースが使用中です。

## SnapCenter ServerとプラグインでサポートされているREST API

SnapCenter REST APIで使用できるリソースは、SnapCenter APIドキュメント ページに表示されるカテゴリ別に分類されています。ここでは、各リソースの簡単な説明と基本リソース パスを示し、使用に際しての追加の考慮事項がある場合はその情報も示します。

### 認証

このAPIを使用すると、SnapCenter Serverにログインできます。このAPIは、以降の要求を認証するために使用されるユーザ認証トークンを返します。

### ドメイン

APIを使用して、次のようなさまざまな処理を実行できます。

- SnapCenterのすべてのドメインを取得する
- 特定のドメインの詳細を取得する
- ドメインの登録や登録解除を行う
- ドメインを変更する

### ジョブ

APIを使用して、次のようなさまざまな処理を実行できます。

- SnapCenterのすべてのジョブを取得する
- ジョブのステータスを取得する
- ジョブをキャンセルまたは停止する

### 設定

APIを使用して、次のようなさまざまな処理を実行できます。

- クレデンシャルを登録、変更、削除する
- SnapCenter Serverに登録されているクレデンシャル情報を表示する
- 通知を設定する

- ・現在Eメール通知を送信するように設定されているSMTPサーバに関する情報を取得し、SMTPサーバ名、受信者名、送信者名を表示する
- ・SnapCenter Serverへのログインの多要素認証（MFA）設定を表示する
- ・SnapCenter ServerへのログインのMFAの有効 / 無効の切り替えや設定を行う
- ・MFAのセットアップに必要な構成ファイルを作成する

## ホスト

APIを使用して、次のようなさまざまな処理を実行できます。

- ・すべてのSnapCenterホストを照会する
- ・SnapCenterから1つ以上のホストを削除する
- ・名前を指定してホストを取得する
- ・ホスト上のすべてのリソースを取得する
- ・リソースIDを使用してリソースを取得する
- ・プラグイン設定の詳細を取得する
- ・プラグイン ホストを設定する
- ・Microsoft SQL Serverホスト用プラグインのすべてのリソースを取得する
- ・Oracleデータベース ホスト用プラグインのすべてのリソースを取得する
- ・カスタム アプリケーション ホスト用プラグインのすべてのリソースを取得する
- ・SAP HANAホスト用プラグインのすべてのリソースを取得する
- ・インストールされているプラグインを取得する
- ・既存のホストにプラグインをインストールする
- ・ホスト パッケージをアップグレードする
- ・既存のホストからプラグインを削除する
- ・ホストにプラグインを追加する
- ・ホストを追加または変更する
- ・Linuxホストのシグネチャを取得する
- ・Linuxホストのシグネチャを登録する
- ・ホストをメンテナンス モードまたは本番モードにする
- ・ホストのプラグイン サービスを開始または再起動する
- ・ホストの名前を変更する

## リソース

APIを使用して、次のようなさまざまな処理を実行できます。

- ・すべてのリソースを取得する

- ・リソースIDを使用してリソースを取得する
- ・Microsoft SQL Serverホスト用プラグインのすべてのリソースを取得する
- ・Oracleデータベース ホスト用プラグインのすべてのリソースを取得する
- ・カスタム アプリケーション ホスト用プラグインのすべてのリソースを取得する
- ・SAP HANAホスト用プラグインのすべてのリソースを取得する
- ・キーを使用してMicrosoft SQL Serverのリソースを取得する
- ・キーを使用してカスタム リソースを取得する
- ・カスタム アプリケーション ホスト用プラグインのリソースを変更する
- ・キーを使用してカスタム アプリケーション ホスト用プラグインのリソースを削除する
- ・キーを使用してSAP HANAのリソースを取得する
- ・SAP HANAホスト用プラグインのリソースを変更する
- ・キーを使用してSAP HANAホスト用プラグインのリソースを削除する
- ・キーを使用してOracleのリソースを取得する
- ・Oracleアプリケーション ボリュームのリソースを作成する
- ・Oracleアプリケーション ボリュームのリソースを変更する
- ・キーを使用してOracleアプリケーション ボリュームのリソースを削除する
- ・Oracleのリソースに関するセカンダリの詳細を取得する
- ・Plug-in for Microsoft SQL Serverを使用してMicrosoft SQL Serverのリソースをバックアップする
- ・Plug-in for Oracle Databaseを使用してOracleのリソースをバックアップする
- ・カスタム アプリケーション用プラグインを使用してカスタム リソースをバックアップする
- ・SAP HANAデータベースを設定する
- ・Oracleデータベースを設定する
- ・SQLデータベースのバックアップをリストアする
- ・Oracleデータベースのバックアップをリストアする
- ・カスタム アプリケーションのバックアップをリストアする
- ・SAP HANAのリソースを作成する
- ・カスタム アプリケーション用プラグインを使用してカスタム リソースを保護する
- ・Plug-in for Microsoft SQL Serverを使用してMicrosoft SQL Serverのリソースを保護する
- ・保護対象のMicrosoft SQL Serverのリソースを変更する
- ・Microsoft SQL Serverのリソースの保護を解除する
- ・Plug-in for Oracle Databaseを使用してOracleのリソースを保護する
- ・保護対象のOracleリソースを変更する
- ・Oracleのリソースの保護を解除する
- ・カスタム アプリケーション用プラグインを使用してバックアップからリソースをクローニングする

- Plug-in for Oracle Databaseを使用してバックアップからOracleアプリケーション ボリュームをクローニングする
- Plug-in for Microsoft SQL Serverを使用してバックアップからMicrosoft SQL Serverのリソースをクローニングする
- Microsoft SQL Serverのリソースのクローン ライフサイクルを作成する
- Microsoft SQL Serverのリソースのクローン ライフサイクルを変更する
- Microsoft SQL Serverのリソースのクローン ライフサイクルを削除する
- 既存のMicrosoft SQL Serverデータベースをローカル ディスクからNetApp LUNに移動する
- Oracleデータベースのクローン仕様ファイルを作成する
- Oracleのリソースのクローン更新ジョブをオンデマンドで開始する
- クローン仕様ファイルを使用してバックアップからOracleのリソースを作成する
- データベースをセカンダリ レプリカにリストアし、データベースを可用性グループに再参加させる
- Oracleアプリケーション ボリュームのリソースを作成する

## バックアップ

APIを使用して、次のようなさまざまな処理を実行できます。

- バックアップの名前、タイプ、プラグイン、リソース、日付を指定してバックアップの詳細を取得する
- すべてのバックアップを取得する
- バックアップの詳細を取得する
- バックアップの名前変更または削除を行う
- Oracleのバックアップをマウントする
- Oracleのバックアップをアンマウントする
- Oracleのバックアップをカタログに登録する
- Oracleのバックアップをカタログから削除する
- ポイントインタイム リカバリを実行するためにマウントする必要があるバックアップをすべて取得する

## クローン

APIを使用して、次のようなさまざまな処理を実行できます。

- Oracleデータベースのクローン仕様ファイルを作成、表示、変更、削除する
- Oracleデータベースのクローン階層を表示する
- クローンの詳細を取得する
- すべてのクローンを取得する
- クローンを削除する
- IDを指定してクローンの詳細を取得する
- Oracleのリソースのクローン更新ジョブをオンデマンドで開始する

- ・ クローン仕様ファイルを使用してバックアップからOracleのリソースをクローニングする

## クローン スプリット

APIを使用して、次のようなさまざまな処理を実行できます。

- ・ クローン リソースのクローン スプリット処理を概算する
- ・ クローン スプリット処理のステータスを取得する
- ・ クローン スプリット処理を開始または停止する

## リソース グループ

APIを使用して、次のようなさまざまな処理を実行できます。

- ・ すべてのリソース グループの詳細を取得する
- ・ 名前を指定してリソース グループを取得する
- ・ カスタム アプリケーション用プラグインのリソース グループを作成する
- ・ Plug-in for Microsoft SQL Serverのリソース グループを作成する
- ・ Plug-in for Oracle Databaseのリソース グループを作成する
- ・ カスタム アプリケーション用プラグインのリソース グループを変更する
- ・ Plug-in for Microsoft SQL Serverのリソース グループを変更する
- ・ Plug-in for Oracle Databaseのリソース グループを変更する
- ・ Plug-in for Microsoft SQL Serverのリソース グループのクローン ライフサイクルを作成、変更、削除する
- ・ リソース グループをバックアップする
- ・ リソース グループをメンテナンス モードまたは本番モードにする
- ・ リソース グループを削除する

## ポリシー

APIを使用して、次のようなさまざまな処理を実行できます。

- ・ ポリシーの詳細を取得する
- ・ 名前を指定してポリシーの詳細を取得する
- ・ ポリシーを削除する
- ・ 既存のポリシーのコピーを作成する
- ・ カスタム アプリケーション用プラグインのポリシーを作成または変更する
- ・ Plug-in for Microsoft SQL Serverのポリシーを作成または変更する
- ・ Plug-in for Oracle Databaseのポリシーを作成または変更する
- ・ Plug-in for SAP HANA Databaseのポリシーを作成または変更する

## ストレージ

APIを使用して、次のようなさまざまな処理を実行できます。

- ・すべての共有を取得する
- ・名前を指定して共有を取得する
- ・共有を作成または削除する
- ・ストレージの詳細を取得する
- ・名前を指定してストレージの詳細を取得する
- ・ストレージを作成、変更、削除する
- ・ストレージ クラスタ上のリソースを検出する
- ・ストレージ クラスタ上のリソースを取得する

## 共有

APIを使用して、次のようなさまざまな処理を実行できます。

- ・共有の詳細を取得する
- ・すべての共有の詳細を取得する
- ・ストレージ上の共有を作成または削除する
- ・名前を指定して共有を取得する

## プラグイン

APIを使用して、次のようなさまざまな処理を実行できます。

- ・ホストのすべてのプラグインを一覧表示する
- ・キーを使用してMicrosoft SQL Serverのリソースを取得する
- ・キーを使用してカスタム リソースを変更する
- ・キーを使用してカスタム リソースを削除する
- ・キーを使用してSAP HANAのリソースを取得する
- ・キーを使用してSAP HANAのリソースを変更する
- ・キーを使用してSAP HANAのリソースを削除する
- ・キーを使用してOracleのリソースを取得する
- ・キーを使用してOracleアプリケーション ボリュームのリソースを変更する
- ・キーを使用してOracleアプリケーション ボリュームのリソースを削除する
- ・Plug-in for Microsoft SQL Serverとキーを使用してMicrosoft SQL Serverのリソースをバックアップする
- ・Plug-in for Oracle Databaseとキーを使用してOracleのリソースをバックアップする
- ・カスタム アプリケーション用のプラグインとキーを使用してカスタム アプリケーションのリソースをバックアップする

- ・キーを使用してSAP HANAデータベースを設定する
- ・キーを使用してOracleデータベースを設定する
- ・キーを使用してカスタム アプリケーションのバックアップをリストアする
- ・SAP HANAのリソースを作成する
- ・Oracleアプリケーション ボリュームのリソースを作成する
- ・カスタム アプリケーション用プラグインを使用してカスタム リソースを保護する
- ・Plug-in for Microsoft SQL Serverを使用してMicrosoft SQL Serverのリソースを保護する
- ・保護対象のMicrosoft SQL Serverのリソースを変更する
- ・Microsoft SQL Serverのリソースの保護を解除する
- ・Plug-in for Oracle Databaseを使用してOracleのリソースを保護する
- ・保護対象のOracleリソースを変更する
- ・Oracleのリソースの保護を解除する
- ・カスタム アプリケーション用プラグインを使用してバックアップからリソースをクローニングする
- ・Plug-in for Oracle Databaseを使用してバックアップからOracleアプリケーション ボリュームをクローニングする
- ・Plug-in for Microsoft SQL Serverを使用してバックアップからMicrosoft SQL Serverのリソースをクローニングする
- ・Microsoft SQL Serverのリソースのクローン ライフサイクルを作成する
- ・Microsoft SQL Serverのリソースのクローン ライフサイクルを変更する
- ・Microsoft SQL Serverのリソースのクローン ライフサイクルを削除する
- ・Oracleデータベースのクローン仕様ファイルを作成する
- ・Oracleリソースのクローン ライフサイクルをオンデマンドで開始する
- ・クローン仕様ファイルを使用してバックアップからOracleのリソースをクローニングする

## レポート

APIを使用して、次のようなさまざまな処理を実行できます。

- ・各プラグインのバックアップ、リストア、クローニング処理のレポートを取得する
- ・スケジュールを追加、実行、削除、変更する
- ・スケジュール済みレポートのデータを取得する

## アラート

APIを使用して、次のようなさまざまな処理を実行できます。

- ・すべてのアラートを取得する
- ・IDを指定してアラートを取得する
- ・複数のアラートを削除する、またはIDを指定してアラートを削除する

## RBAC

APIを使用して、次のようなさまざまな処理を実行できます。

- ・ユーザ、グループ、ロールの詳細を取得する
- ・ユーザを追加または削除する
- ・ロールにユーザを割り当てる
- ・ロールへのユーザの割り当てを解除する
- ・ロールを作成、変更、削除する
- ・グループをロールに割り当てる
- ・ロールへのグループの割り当てを解除する
- ・グループを追加または削除する
- ・既存のロールのコピーを作成する
- ・ユーザまたはグループへのリソースの割り当てや割り当ての解除を行う

## 構成

APIを使用して、次のようなさまざまな処理を実行できます。

- ・構成設定を表示する
- ・構成設定を変更する

## 証明書設定

APIを使用して、次のようなさまざまな処理を実行できます。

- ・SnapCenter Serverまたはプラグインホストの証明書ステータスを表示する
- ・SnapCenter Serverまたはプラグインホストの証明書設定を変更する

## リポジトリ

APIを使用して、次のようなさまざまな処理を実行できます。

- ・リポジトリ バックアップを取得する
- ・リポジトリに関する設定情報を表示する
- ・SnapCenterリポジトリを保護、リストアする
- ・SnapCenterリポジトリの保護を解除する
- ・リポジトリを再構築したりフェイルオーバーしたりする

## version

このAPIを使用すると、SnapCenterのバージョンを表示できます。

# Swagger API WebページからREST APIにアクセスする方法

REST APIはSwagger Webページから利用できます。Swagger Webページにアクセスして、SnapCenter Server REST APIを表示したり、APIを手動で呼び出したりすることができます。REST APIを使用して、SnapCenter Serverの管理やデータ保護処理を行うことができます。

REST APIを実行するSnapCenter Serverの管理IPアドレスまたはドメイン名を確認しておく必要があります。

REST APIクライアントを実行するために特別な権限は必要ありません。すべてのユーザがSwagger Webページにアクセスできます。REST APIでアクセスするオブジェクトに対する権限は、REST APIへのログインに使用するトークンを生成したユーザに基づきます。

## 手順

1. ブラウザから、*https://<SnapCenter\_IP\_address\_or\_name>:<SnapCenter\_port>/swagger/* の形式で Swagger Web ページにアクセスするための URL を入力します。



REST API URL に +、..、%、& の文字が含まれていないことを確認します。

2. **Swagger Explore** フィールドに、Swagger API ドキュメントが自動的に表示されない場合は、次のように入力します: *https://<SnapCenter\_IP\_address\_or\_name>:<SnapCenter\_port>/Content/swagger/SnapCenter.yaml*
3. \*探索\*をクリックします。

APIリソース タイプ（カテゴリ）の一覧が表示されます。

4. APIリソース タイプをクリックすると、そのリソース タイプのAPIが表示されます。

SnapCenter REST APIの実行時に予期しない動作が発生した場合は、ログ ファイルで原因を特定して問題を解決することができます。モニター > ログ > ダウンロード をクリックすると、SnapCenterユーザー インターフェイスからログ ファイルをダウンロードできます。

# REST APIの使用の開始

SnapCenter REST APIはすぐに使用を開始できます。APIにアクセスすると、稼働中の設定で複雑なワークフロー プロセスを使用する前にいくつかの情報を確認できます。

## Hello world

システムで簡単なコマンドを実行して、SnapCenter REST APIの使用を開始し、使用できるかどうかを確認できます。

### 開始する前に

- システムでcURLユーティリティが使用可能なことの確認
- SnapCenter ServerのIPアドレスまたはホスト名
- SnapCenter REST APIにアクセスする権限を持つアカウントのユーザ名とパスワード



クレデンシャルに特殊文字が含まれている場合は、使用するシェルに基づいてCurlで受け入れられる形式で指定する必要があります。たとえば、各特殊文字の前にバックスラッシュを挿入したり、全体を囲んだりすることができます。`username:password`一重引用符で囲んだ文字列。

## 手順

コマンドライン インターフェイスで、次のコマンドを実行してプラグイン情報を取得します。

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

例：

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

## 著作権に関する情報

Copyright © 2025 NetApp, Inc. All Rights Reserved. Printed in the U.S.このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

ネットアップの著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、ネットアップによって「現状のまま」提供されています。ネットアップは明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。ネットアップは、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

ネットアップは、ここに記載されているすべての製品に対する変更を隨時、予告なく行う権利を保有します。ネットアップによる明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、ネットアップは責任を負いません。この製品の使用または購入は、ネットアップの特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許によって保護されている場合があります。

権利の制限について：政府による使用、複製、開示は、DFARS 252.227-7013（2014年2月）およびFAR 5225.227-19（2007年12月）のRights in Technical Data -Noncommercial Items（技術データ - 非商用品目に関する諸権利）条項の(b)(3)項、に規定された制限が適用されます。

本書に含まれるデータは商用製品および / または商用サービス（FAR 2.101の定義に基づく）に関係し、データの所有権はNetApp, Inc.にあります。本契約に基づき提供されるすべてのネットアップの技術データおよびコンピュータソフトウェアは、商用目的であり、私費のみで開発されたものです。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用権を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc.の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用権については、DFARS 252.227-7015(b)項（2014年2月）で定められた権利のみが認められます。

## 商標に関する情報

NetApp、NetAppのロゴ、<http://www.netapp.com/TM>に記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。