

Secure Access APIを使用したCurlによる通知先リストの管理

内容

[はじめに](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[設定](#)

[1. APIキーの作成](#)

[2. APIアクセストークンの生成](#)

[3. 通知先リストの管理](#)

[すべての通知先リストの取得](#)

[通知先リスト内のすべての通知先を取得](#)

[新しい通知先リストの作成](#)

[通知先リストへの通知先の追加](#)

[通知先リストの削除](#)

[通知先リストからの通知先の削除](#)

[トラブルシューティング](#)

[関連情報](#)

はじめに

このドキュメントでは、セキュアアクセスAPI(SAPI)を使用してcurl経由で宛先リストを管理する方法について説明します。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- セキュアなアクセス
- セキュアアクセスAPI
- curl
- Json

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づいています。

- セキュアなアクセス

- セキュアアクセスAPI
- curl
- Json

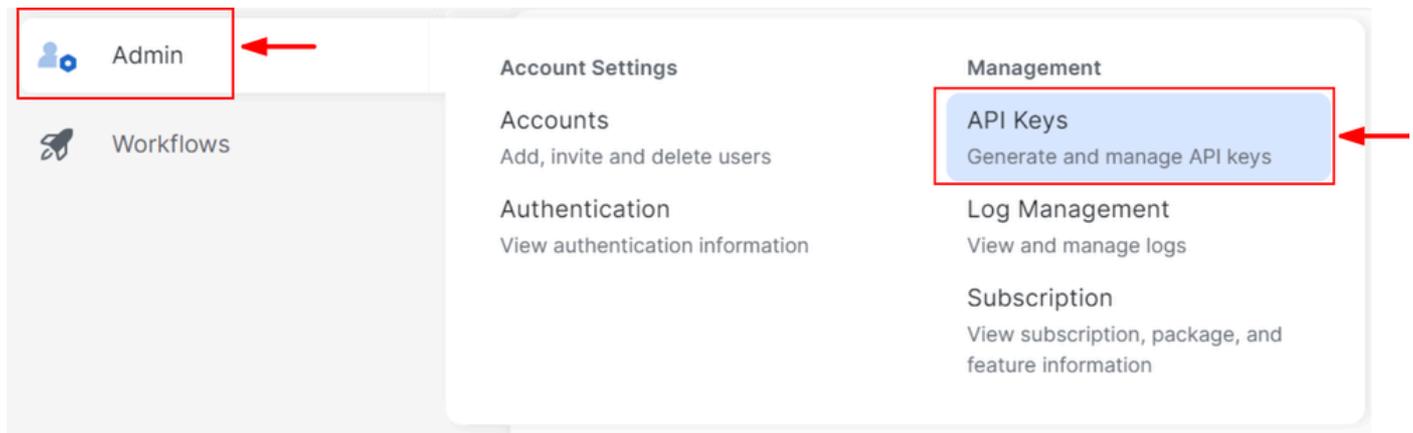
このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

設定

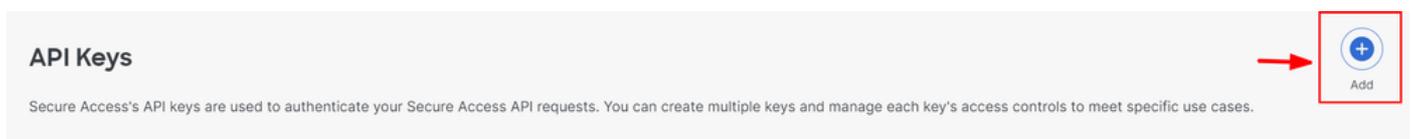
1. APIキーの作成

[Secure Access Dashboard](#)に移動します。

- Admin > Api Keys >をクリックします。 Add



APIキー1の作成



APIキーの作成2

- 必要に応じて、API Key Name、Description (Optional)、Expiry Dateを追加します。

Add New API Key

To add this unique API key to Secure Access, select its scope—what it can do—and set an expiry date. The key and secret created here are unique. Deleting, refreshing or modifying this API key may break or interrupt integrations that use this key.

The screenshot shows the 'Add New API Key' form with several fields highlighted by red boxes and arrows:

- API Key Name:** A text input field containing 'New API Key'.
- Description (Optional):** An empty text input field.
- Key Scope:** A list of scopes with checkboxes and counts:
 - Auth (1 >)
 - Deployments (16 >)
 - Investigate (2 >)
 - Policies (4 >)
 - Reports (9 >)
- Expiry Date:** Radio buttons for 'Never expire' (selected) and 'Expire on' (with a date picker set to 'May 21 2024').
- Scope Selection Panel:** A panel titled '1 selected' with a 'Remove All' link. It shows a 'Scope' dropdown set to 'Policies' and a 'Read / Write' dropdown with a count of '4' and a close button.

Buttons at the bottom: 'CANCEL' and 'CREATE KEY'.

APIキーの作成3

- の下で Key Scope、「Policies」、「ポリシーの展開」の順に選択します
- 「Destination Lists」を選択し、Destinations
- 必要に応じてScope変更し、必要に応じて Read/Write
- クリック CREATE KEY

Add New API Key

To add this unique API key to Secure Access, select its scope—what it can do—and set an expiry date. The key and secret created here are unique. Deleting, refreshing or modifying this API key may break or interrupt integrations that use this key.

API Key Name

Description *(Optional)*

Key Scope / Policies

Select the appropriate access scopes to define what this API key can do.

 Destination Lists Destinations DLP Indexer Rules

2 selected

[Remove All](#)

Scope

Policies / Destination Lists

Read / Write



Policies / Destinations

Read / Write



Expiry Date

 Never expire Expire on

May 21 2024

[CANCEL](#)[CREATE KEY](#)

APIキーの作成4

- API KeyおよびKey Secretをコピーして、をクリックします。 ACCEPT AND CLOSE

Click Refresh to generate a new key and secret.

API Key

e2- [masked] [copy icon]

Key Secret

1e- [masked] [copy icon]

Copy the Key Secret. For security reasons, it is only displayed once. If lost, it cannot be retrieved.

[ACCEPT AND CLOSE](#)

APIキーの作成5

注:APIシークレットをコピーする機会は1回だけです。セキュアアクセスではAPIシークレットは保存されず、最初の作成後は取得できません。

2. APIアクセストークンの生成

APIアクセストークンを生成するには、トークン認証要求を作成します。

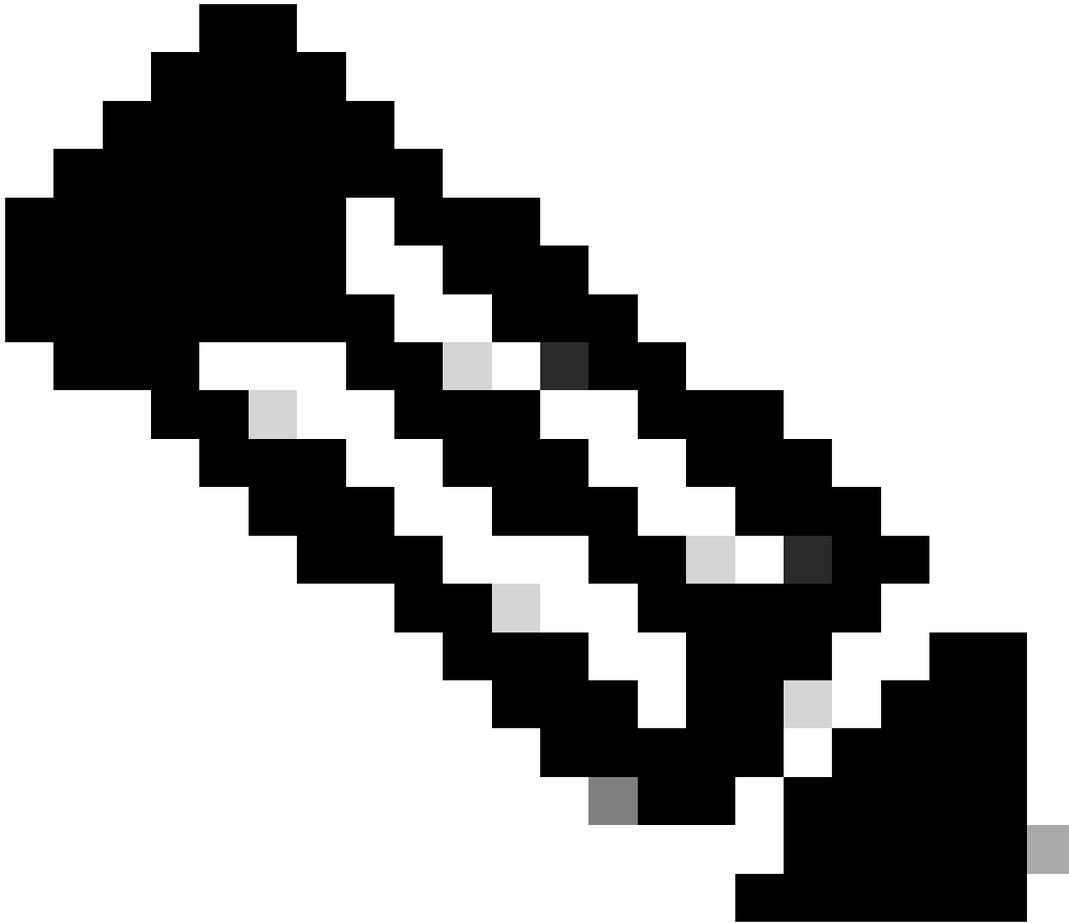
トークン許可要求

APIアクセストークンを生成するには、組織で作成したSecure Access API資格情報を使用します。

- curlサンプルで、Secure Access APIのキーとシークレットを置き換えます

```
curl --user key:secret --request POST --url https://api.sse.cisco.com/auth/v2/token -H Content-Type: ap
```

- 生成されたBearer APIトークンをコピーして保存する
-



注：セキュアアクセスOAuth 2.0アクセストークンは、1時間（3600秒）で有効期限が切れます。アクセストークンの有効期限が近づくまでは、アクセストークンを更新しないことをお勧めします。

3. 通知先リストの管理

通知先リストを管理するには、次の方法があります。

すべての通知先リストの取得

WindowsコマンドプロンプトまたはMacターミナルを開いてコマンドを実行します。

```
curl -L --location-trusted --request GET --url https://api.sse.cisco.com/policies/v2/destinationlists -
```

サンプル出力のスニペット：

```
{"id":23456789,"organizationId":1234567,"access":"none","isGlobal":false,"name":" Test Block list","thi
{"destinationCount":2,"domainCount":2,"urlCount":0,"ipv4Count":0,"applicationCount":0}
```

出力の「id」フィールドの下にリストされているdestinationListIdをメモします。このフィールドは、この宛先リストに固有のGET、POST、またはDELETE要求に対してさらに使用されます。

通知先リスト内のすべての通知先を取得

- 前述のこのステップを使用したdestinationListId、すべての宛先リストの取得を取得します。

WindowsコマンドプロンプトまたはMacターミナルを開いてコマンドを実行します。

```
curl -L --location-trusted --request GET --url https://api.sse.cisco.com/policies/v2/destinationlists/d
```

出力例：

```
{"status":{"code":200,"text":"OK"},"meta":{"page":1,"limit":100,"total":3},"data":
[
{"id":"415214","destination":"cisco.com","type":"domain","comment":null,"createdAt":"2024-02-20 09:15:4
}]
```

新しい通知先リストの作成

WindowsコマンドプロンプトまたはMacターミナルを開いてコマンドを実行します。

```
curl -L --location-trusted --request POST --url https://api.sse.cisco.com/policies/v2/destinationlists
```

注: 「Destination List Name」は任意の名前に置き換えてください。

出力例 :

```
{"id":23456789,"organizationId":1234567,"access":"none","isGlobal":false,"name":"API List 1","thirdpart
```

通知先リストへの通知先の追加

- 前述のこのステップを使用したdestinationListId、すべての宛先リストの取得を取得します。

WindowsコマンドプロンプトまたはMacターミナルを開いてコマンドを実行します。

```
curl -L --location-trusted --request POST --url https://api.sse.cisco.com/policies/v2/destinationlists/
```

出力例：

```
{"status":{"code":200,"text":"OK"},"data":{"id":17804929,"organizationId":1234567,"access":"none","isGlobal":true,"destinationCount":3}}
```

通知先リストの削除

- 前述のこのステップを使用したdestinationListId、すべての宛先リストの取得を取得します。

WindowsコマンドプロンプトまたはMacターミナルを開いてコマンドを実行します。

```
curl -L --location-trusted --request DELETE --url https://api.sse.cisco.com/policies/v2/destinationlist
```

出力例：

```
{"status":{"code":200,"text":"OK"},"data":[]}
```

通知先リストからの通知先の削除

- 前述のこのステップを使用したdestinationListId、すべての宛先リストの取得を取得します。
- 前述のこの手順「[宛先リスト内のすべての宛先を取得する](#)」を使用して、削除する必要があるリスト内の特定の宛先の「id」を取得します。

WindowsコマンドプロンプトまたはMacターミナルを開いてコマンドを実行します。

```
curl -L --location-trusted --request DELETE --url https://api.sse.cisco.com/policies/v2/destinationlist
```

出力例：

```
{"status":{"code":200,"text":"OK"},"data":{"id":17804929,"organizationId":1234567,"access":"none","isGlobal":true}}
```

トラブルシューティング

Secure Access APIエンドポイントは、HTTP応答コードを使用してAPI要求の成功または失敗を示します。一般に、2xxの範囲のコードは成功を示し、4xxの範囲のコードは提供された情報に起因するエラーを示し、5xxの範囲のコードはサーバエラーを示します。問題を解決するアプローチは、受信した応答コードによって異なります。

200	OK	Success. Everything worked as expected.
201	Created	New resource created.
202	Accepted	Success. Action is queued.
204	No Content	Success. Response with no message body.
400	Bad Request	Likely missing a required parameter or malformed JSON. The syntax of your query may need to be revised. Check for any spaces preceding, trailing, or in the domain name of the domain you are trying to query.
401	Unauthorized	The authorization header is missing or the key and secret pair is invalid. Ensure your API token is valid.
403	Forbidden	The client is unauthorized to access the content.
404	Not Found	The requested resource doesn't exist. Check the syntax of your query or ensure the IP and domain are valid.
409	Conflict	The client requests that the server create the resource, but the resource already exists in the collection.
429	Exceeded Limit	Too many requests received in a given amount of time. You may have exceeded the rate limits for your organization or package.
413	Content Too Large	The request payload is larger than the limits defined by the server.

REST API : 応答コード1

500	Internal Server Error	Something wrong with the server.
503	Service Unavailable	Server is unable to complete request.

REST API : 応答コード2

また、APIに関連するエラーや問題のトラブルシューティングを行う際には、次のレート制限に注意する必要があります。

- [セキュアアクセスAPIの制限](#)

関連情報

- [Cisco Secure Accessユーザガイド](#)
- [シスコテクニカルサポートおよびダウンロード](#)
- [セキュアアクセスAPIキーの追加](#)
- [開発者ユーザガイド](#)

翻訳について

シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。