



# Cisco ASA REST API クイック スタート ガイド

初版 : 2014 年 12 月 24 日

## 目次

- 「概要」 (P.1)
- 「ASA REST API 要求および応答」 (P.2)
- 「ASA REST API エージェントおよびクライアントのインストールと設定」 (P.4)
- 「ドキュメント コンソールおよび API スクリプトのエクスポート」 (P.6)
- 「ASA での REST API デバッグの有効化」 (P.7)
- 「ASA REST API 関連の Syslog メッセージ」 (P.7)
- 「関連資料」 (P.8)

## 概要

それぞれの Cisco ASA の設定と管理には、次の複数のオプションを使用できます。

- CLI (コマンドライン インターフェイス) - 接続されたコンソールから直接 ASA に制御コマンドを送信します。
- Adaptive Security Device Manager (ASDM) - ASA の設定、管理および監視に使用できるグラフィカル ユーザ インターフェイスを備えた「オンボックス」の管理アプリケーション。
- Cisco Security Manager - 多くのセキュリティ デバイスの中規模から大規模までのネットワーク向けですが、このグラフィカル アプリケーションを使用して個々の ASA を設定、管理、監視できます。

シスコの ASA REST API のリリースによって、軽量で使いやすいオプションがさらに追加されました。これは「RESTful」原則に基づいたアプリケーション プログラミング インターフェイス (API) であり、迅速にダウンロードして、API が動作する任意の ASA 上でイネーブルにできます。



ブラウザに REST クライアントをインストールした後、特定の ASA の REST エージェントにコ  
ンタクトして、標準の HTTP メソッドを使用し、現在の設定情報にアクセスして、追加の設定  
パラメータを発行できます。



注意

REST API が ASA でイネーブルになっていると、他のセキュリティ管理プロトコルによる接続  
はブロックされません。つまり、CLI、ASDM または Security Manager を使用している他の  
ユーザも、ASA の設定を変更できるということです。

## ASA REST API 要求および応答

ASA REST API を使用すると、Representational State Transfer (REST) API を介して、個々の  
ASA の管理にプログラムでアクセスできます。API によって、外部クライアントは ASA リ  
ソースでの CRUD 操作（作成、読み取り、更新、削除）を実行できます。これは HTTPS プロ  
トコルと REST の方法論に基づいています。

すべての API 要求が HTTPS を介して ASA に送信され、応答が返されます。

この項では、要求の構造化の方法と予期される応答について概要を説明します。

### 要求構造

使用できる要求メソッドは次のとおりです。

- GET - 指定されたオブジェクトからデータを取得します。
- PUT - 指定されたオブジェクトが存在する場合、提供された情報をそのオブジェクトに追  
加します。存在しない場合は、提供された情報でオブジェクトを作成します。
- POST - 提供された情報を既存のオブジェクトに追加します。
- DELETE - 指定されたオブジェクトを削除します。
- PATCH - 指定されたオブジェクトに部分的な変更を適用します。

### 応答構造

各要求によって、標準ヘッダー、応答コンテンツおよびステータス コードを含んだ ASA から  
の HTTPS 応答が生成されます。

応答の構造は次のようになります。

- LOCATION - 新しく作成されたリソース ID。POST の場合のみ。新しいリソース ID を  
(URI 表現として) 保持します。
- CONTENT-TYPE - 応答メッセージ本文のメディア タイプ。応答メッセージ本文の表現と構  
文を示します。

各応答には HTTP ステータスまたはエラー コードが含まれます。使用可能なコードは、次のカ  
テゴリに分けられます。

- 20x - 200 番台のコードは次のような処理の成功を示します。
  - 200 OK - 成功した要求に対する標準応答。
  - 201 作成済み - 要求が完了し、新しいリソースが作成されました。

- 202 承認 - 要求が受け入れられましたが、処理は完了していません。
- 204 コンテンツなし - サーバが要求を正常に処理しましたが、コンテンツは返されていません。
- 4xx - 400 番台のコードは、次のようなクライアント側のエラーを示します。
  - 400 不正な要求 - 認識されないパラメータ、欠落しているパラメータ、無効な値などの不正なクエリー パラメータ。
  - 404 見つかりません - 指定された URL は既存のリソースに一致しません。たとえば、HTTP DELETE は、そのリソースがないために失敗する場合があります。
  - 405 メソッドは許可されません - リソースで許可されていない HTTP 要求が行われました。たとえば、読み取り専用リソースでの POST など。
- 5xx - 500 番台のコードはサーバ側のエラーを示します。

エラーが発生した場合は、エラー コードに加えて、エラーに関する詳細を示すエラー オブジェクトがリターン応答に含まれる場合があります。JSON のエラーおよび警告の応答スキーマは次のとおりです。

```
[
  { "code" : "string",
    "details": "string",
    "context": attribute name,
    "level"  : <Error/Warning/Info>
  },
  ...
]
```

ここで、オブジェクトのプロパティは次のとおりです。

プロパティ	タイプ	説明
messages	ディクショナリのリスト	エラーまたは警告メッセージのリスト
code	文字列	Error/Warning/Info コード
details	文字列	Error/Warning/Info に対応する詳細メッセージ



(注)

REST API 呼び出しによって ASA の設定に加えられた変更は、起動設定には反映されません。つまり、変更は実行時設定だけに割り当てられます。したがって、変更内容を起動設定に保存するには、Generic CLI Command Executer API 要求を介して明示的に、write memory コマンドを発行する必要があります。Generic CLI Command Executer API の使用方法については、『About the ASA REST API』ガイドを参照してください。

# ASA REST API エージェントおよびクライアントのインストールと設定

ここでは、ASA での REST API エージェントのインストールと有効化、およびローカル ホストでの REST API クライアントの追加と設定について説明します。

## REST API エージェントのダウンロードとインストール

CLI を使用して、特定の ASA に ASA REST API エージェントをダウンロードおよびインストールするには、次の手順に従ってください。

- ステップ 1** 目的の ASA で、`copy <package> disk0` コマンドを発行して、`cisco.com` から ASA のフラッシュメモリに現在の ASA REST API パッケージをダウンロードします。次に例を示します。

```
copy tftp://10.7.0.80/asa-restapi-101-1fbff-k8.SPA disk0:
```

- ステップ 2** `rest-api image disk0:<package>` コマンドを発行して、パッケージを確認しインストールします。次に例を示します。

```
rest-api image disk0:/asa-restapi-101-1fbff-k8.SPA
```

インストーラは互換性と有効性の確認を行ってからパッケージをインストールします。ASA は再起動されません。

## REST API エージェントのイネーブル化と設定

特定の ASA 上の REST API エージェントをイネーブルにして設定するには、次の手順に従います。

- ステップ 1** 正しいソフトウェアイメージが ASA にインストールされていることを確認します。

ASA 55 *nn-X* の場合、これは `asa100 12 16 15 smp k8.bin` 以降です。ASA*v* の場合は、`asav100-12-16-15.ova` です。`.ova` イメージがすでに ASA*v* にインストールされている場合は、`asa100 12 16 15 smp k8.bin` を使用して VM イメージを更新できます。

- ステップ 2** CLI を使用して、HTTP サーバが ASA でイネーブルになっていることと、API クライアントが管理インターフェイスに接続できることを確認します。次に例を示します。

```
http server enable
http 0.0.0.0 0.0.0.0 <management interface nameif>
```

- ステップ 3** CLI を使用して、API 接続用の HTTP 認証を定義します。次に例を示します。

```
aaa authentication http console LOCAL
```

- ステップ 4** CLI を使用して、読み取り/書き込み権限を持つローカル ユーザ（この場合は、特権レベル 15）を作成します。次に例を示します。

```
username <user name> password <password> encrypted privilege 15
```

モニタリング要求を呼び出すには特権レベル 3 以上が必要で、GET 要求を呼び出すには特権レベル 5 以上が必要です。また、PUT/POST/DELETE 操作を呼び出すには特権レベル 15 が必要です。

**ステップ 5** CLI を使用して、API トラフィック用に ASA 上でスタティック ルートを作成します。次に例を示します。

```
route <management interface nameif> 0.0.0.0 0.0.0.0 <gwip> 1
```

**ステップ 6** CLI を使用して、ASA 上の REST API エージェントをイネーブルにします。次に例を示します。

```
rest-api agent
```

## REST API クライアントの設定

ローカルホストのブラウザ上に REST API クライアントをインストールして設定するには、次の手順に従います。

**ステップ 1** REST API クライアントを取得して、使用しているブラウザ用にインストールします。

Chrome の場合は、Google から REST クライアントをインストールします。Firefox の場合は、RESTClient アドオンをインストールします。Internet Explorer はサポートされません。

**ステップ 2** ブラウザを使用して、次の要求を開始します。

```
https:<asa management ip address>/api/objects/networkobjects
```

エラーでない応答を受信した場合は、ASA で機能している REST API エージェントに到達しています。

エージェントの要求で問題が発生した場合は、「[ASA での REST API デバッグの有効化](#)」(P.7) の説明に従って CLI コンソールでデバッグ情報の表示を有効にできます。

**ステップ 3** 任意に、POST 操作を実行して、ASA への接続をテストできます。

次に例を示します。

基本認証クレデンシャルの指定 : <username><password>

ターゲット要求アドレス : https://<asa management ipaddress>/api/objects/networkobjects

本文コンテンツタイプ : application/json

処理の raw 本文 :

```
{
  "kind": "object#NetworkObj",
  "name": "TestNetworkRangeObj",
  "host": {
    "kind": "IPv4Network",
    "value": "12.12.12.0/24"
  }
}
```

ASA REST API を使用して、ASA を設定および監視できるようになりました。呼び出しの説明と例については、API マニュアルを参照してください。

# ドキュメント コンソールおよび API スクリプトのエクスポート

API 呼び出しについて把握し、ASA で直接試行するための「サンドボックス」として、*host:port/doc/*にある REST API オンライン ドキュメント コンソール（「ドキュメント UI」と呼ばれる）を使用することもできます。

さらに、ドキュメント UI の [Export Operation] ボタンを使用して、表示されたメソッドの例を JavaScript、Python、Perl のスクリプト ファイルとしてローカル ホストに保存できます。その後は、そのスクリプトを ASA に適用し、他の ASA や他のネットワーク デバイスに適用するために編集できます。これは、主に学習用ブートストラップ ツールとして使用するためのものです。

## JavaScript

JavaScript ファイルを使用するには、<http://nodejs.org/>にある `node.js` をインストールする必要があります。JavaScript ファイルは、通常はブラウザ用に記述されていますが、`node.js` を使用すればコマンドライン スクリプトのように実行できます。インストール手順に従った後、`node script.js` で目的のスクリプトを実行します。

## Python

Python スクリプトでは、<https://www.python.org/> から利用できる Python をインストールする必要があります。Python をインストールしたら、`python script.py username password` で目的のスクリプトを実行できます。

## Perl

Perl スクリプトを使用するには、追加のセットアップが必要です。5 つのコンポーネントとして、Perl 自体と 4 つの Perl ライブラリが必要になります。

- Perl パッケージは <http://www.perl.org/> にあります。
- `Bundle::CPAN` は、<http://search.cpan.org/~andk/Bundle-CPAN-1.861/CPAN.pm> にあります。
- `REST::Client` は、<http://search.cpan.org/~mcrawfor/REST-Client-88/lib/REST/Client.pm> にあります。
- `MIME::Base64` は、<http://perldoc.perl.org/MIME/Base64.html> にあります。
- `JSON` は、<http://search.cpan.org/~makamaka/JSON-2.90/lib/JSON.pm> にあります。

次に示すのは、Macintosh に Perl をブートストラップする例です。

```
$ sudo perl -MCPAN e shell
cpan> install Bundle::CPAN
cpan> install REST:: Client
cpan> install MIME::Base64
cpan> install JSON
```

依存関係をインストールしたら、`perl script.pl username password` を使用してスクリプトを実行できます。

# ASA での REST API デバッグの有効化

ASA での REST API の設定または接続に問題がある場合は、次の CLI コマンドを使用して、コンソールのデバッグ メッセージの表示を有効化できます。デバッグ メッセージをディセーブルにするには、このコマンドの **no** 形式を使用します。

```
debug rest-api [agent | daemon | cli | http | process] [event | error]
```

```
no debug rest-api
```

## 構文の説明

<b>agent</b>	(任意) REST API エージェントのデバッグ情報をイネーブルにします。
<b>daemon</b>	(任意) REST API デーモンからエージェントへの通信のデバッグ メッセージをイネーブルにします。
<b>cli</b>	(任意) REST API CLI デーモンからエージェントへの通信のデバッグ メッセージをイネーブルにします。
<b>client</b>	(任意) REST API クライアントと REST API エージェント間でメッセージのルーティングのデバッグ情報をイネーブルにします。
<b>process</b>	(任意) REST API エージェントプロセスの開始および停止のデバッグ情報をイネーブルにします。
<b>event</b>	(任意) API によって記録されたイベントだけにデバッグ メッセージを制限するには、このキーワードを使用します。
<b>error</b>	(任意) API によって記録されたエラーのみにデバッグ メッセージを制限するには、このキーワードを使用します。

## 使用上のガイドライン

特定のコンポーネントのキーワードを指定しない場合（つまり、単に **debug rest-api** コマンドを発行した場合）、デバッグ メッセージは、すべてのコンポーネントのタイプに対して表示されます。**event** または **error** キーワードを指定しない場合、イベントとエラーの両方のメッセージが、指定したコンポーネントに対して表示されます。たとえば、**debug rest-api daemon event** の場合は、API デーモンからエージェントへの通信についてのイベント デバッグ メッセージだけが表示されます。

## ASA REST API 関連の Syslog メッセージ

ASA REST API 関連のシステム ログ メッセージについて、この項で説明します。

### 342001

```
エラー メッセージ %ASA-7-342001: REST API Agent started successfully.
```

**説明** REST API クライアントで ASA を設定するには、その前に REST API エージェントを正常に起動する必要があります。

**推奨処置** なし。

## 342002

エラーメッセージ %ASA-3-342002: REST API Agent failed, reason: reason

**説明** REST API エージェントが、さまざまな理由で起動に失敗したかクラッシュした可能性があり、理由が示されます。

- *reason* : REST API エラーの原因

**推奨処置** 問題を解決するためのアクションは、記録された理由によって異なります。たとえば、REST API エージェントは Java プロセスでメモリが不足するとクラッシュします。この場合は、REST API エージェントを再起動する必要があります。正常に再起動できない場合は、根本原因の修正を特定するために Cisco TAC にお問い合わせください。

## 342003

エラーメッセージ %ASA-3-342003: REST API Agent failure notification received.Agent will be restarted automatically.

**説明** REST API エージェントからのエラー通知が受信され、エージェントの再起動が試みられます。

**推奨処置** なし。

## 342004

エラーメッセージ %ASA-3-342004: Failed to automatically restart the REST API Agent after 5 unsuccessful attempts. Use the 'no rest-api agent' and 'rest-api agent' commands to manually restart the Agent.

**説明** REST API エージェントは複数の試行後、起動に失敗しました。

**推奨処置** 失敗の理由をさらに把握するには、syslog %ASA-3-342002（記録されている場合）を参照してください。REST API エージェントの無効化を試行するために **no rest-api agent** コマンドを入力し、REST API エージェントを再有効化するために、**rest-api agent** コマンドを使用します。

## 関連資料

ASA とその設定および管理に関する詳細な情報を検索するには、次のリンクを使用してください。

- Cisco ASA シリーズ マニュアルのナビゲーション: <http://www.cisco.com/go/asadocs>

ASAv でサポートされない ASA の機能の一覧を表示するには、次のリンクを使用してください。

- <http://www.cisco.com/c/en/us/td/docs/security/asa/asa92/configuration/general/asa-general-cli/intro-asav.html#pgfId-1156883>



このマニュアルは、「関連資料」の項から利用できるマニュアルと併せて使用してください。

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: [www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

このマニュアルで使用している IP アドレスおよび電話番号は、実際のアドレスおよび電話番号を示すものではありません。マニュアル内の例、コマンド出力、ネットワークトポロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

© 2014-2015 Cisco Systems, Inc. All rights reserved.

