



Cisco Expressway REST API 概要ガイド (X14.2)

初版：2022年8月10日

シスコシステムズ合同会社

〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー

<http://www.cisco.com/jp>

お問い合わせ先：シスココンタクトセンター
0120-092-255（フリーコール、携帯・PHS含む）

電話受付時間：平日 10:00～12:00、13:00～17:00

<http://www.cisco.com/jp/go/contactcenter/>



目次

第 1 章

Expressway REST API の使用 1

変更履歴 1

Expressway REST API の使用 2



第 1 章

Expressway REST API の使用

- [変更履歴 \(1 ページ\)](#)
- [Expressway REST API の使用 \(2 ページ\)](#)

変更履歴

表 1: 変更履歴

日付	変更内容	理由
2022 年 8 月	X14.2 で最初に公開されました。 単一ノードをアップグレードするための API が導入されました。	X14.2 リリース
2020 年 4 月	ステータス API サポートがプロビジョニング API とともに追加されました。Expressway REST API にアクセスするためのベース URL を <code>http://<external_address>/api</code> に変更しました。	ドキュメントの訂正
2019 年 1 月	バージョン固有ではなくなったため、ドキュメントからソフトウェアバージョンを削除しました。	ドキュメントの訂正
2018 年 9 月	X8.11 が使用できなくなったため、ソフトウェアバージョンを X8.11 から X8.11.1 に変更しました。	ソフトウェアの取り消し

日付	変更内容	理由
2018年7月	APIは自己文書化されているため、個々のAPIコールに関する詳細を削除しました。	X8.11 リリース
2017年7月	REST APIのフェーズ3。ファイアウォールルール、SIP、およびドメイン証明書が含まれるようになりました。	X8.10 リリース
2017年1月	HTTPの許可リストのコールとフィルタオプションによる取得を更新しました。	X8.9.1 リリース
2016年12月	REST APIのフェーズ2。B2B機能と削除機能が含まれるようになりました。	X8.9 リリース
2016年6月	モバイルおよびリモートアクセス (MRA) を設定するためのREST APIの第1フェーズ。	X8.8 リリース

Expressway REST API の使用

Expressway REST API は、RAML バージョン 0.8 (raml.org/spec.html) に準拠しています。API は完全に準拠していますが、ネストされた API はサポートしていません。

API は、RESTful API モデリング言語 (RAML) を使用して自己文書化されています。システムの RAML 定義にアクセスするには、<https://<Expressway FQDN or IP address>/api/raml> を使用します。実験的なスキーマブラウザは Web ユーザーインターフェイスに組み込まれており、**実験的なメニュー**からアクセスできます。

スキーマ

Expressway REST API のすべてのリクエストおよび応答スキーマは、JSON スキーマバージョン 4 (json-schema.org/documentation.html) を使用します。リクエストパラメータはサポートされておらず、JSON スキーマのみが使用されます。

認証

API には HTTPS 経由でのみアクセスでき、認証が必要です。認証ログイン情報は、Expressway ノードの管理者ログイン情報です。

[ベース URL (Base URL)]

Expressway REST API にアクセスするためのベース URL は `http://<external_address>/api` です。たとえば、システム情報にアクセスするには、`https://10.0.0.1/api/provisioning/sysinfo` を使用します。

REST API は、次のカテゴリで公開されています。

- Cisco Expressway-E
/provisioning/edge/ <remaining path> (たとえば、
`https://10.0.0.1/api/provisioning/edge/zone/traversalserver`)
- Cisco Expressway-C
/provisioning/controller/ <remaining path> (たとえば、
`https://10.0.0.1/api/provisioning/controller/zone/traversalclient`)
- Cisco Expressway-E および Expressway-C の間で共通
/provisioning/common/<remaining path> (たとえば、
`https://10.0.0.1/api/provisioning/common/adminaccount/changepassword`)

再起動やシステム情報などのメンテナンス関連の項目は、スタンドアロンのコールであり、どのカテゴリにも適用されません。

Get リクエストをフィルタ処理して、特定のエントリを見つけることもできます。たとえば、`/controller/zone/traversalclient/name/myzone` は、「myzone」というトラバーサルクライアントゾーンを返します。

- 機能のステータスを `status/common/<remaining path>` (たとえば、
<http://10.0.0.1/api/status/common/smartlicensing/licensing>) として取得する REST API

リクエストと応答の例

このセクションでは、Expressway API メソッドの使用法の例を示します。この例は、DNS サーバーと NTP サーバーの API メソッドに関連しています。

例：API を使用して EXPRESSWAY をアップグレードする

SFTP 情報を追加する

この例では、JSON API を使用して SFTP 情報を追加します

URL	PUT <code>https://<Expressway FQDN or IP address>/api/v1/provisioning/common/sftpconfig</code>
リクエスト本文	<pre>{ "SftpServer": "10.0.0.0", "Username": "username", "Password": "password", "RsaKey": "*****" }</pre>
応答本文	<pre>{ "Message": "Update successful" }</pre>

「RsaKey」 値については、SFTP サーバー管理から RSA 公開キーを取得する方法をお勧めします。または、ネットワーク内のシステム（SFTP サーバーに到達できる）から以下のコマンドを使用して、公開キーを取得します。

```
ssh-keyscan <IP of SFTP server> | grep "ssh-rsa" --color
```

この例では、cURL を使用して SFTP 構成情報を追加します。

```
curl -X PUT -k -i 'https:// <Expressway FQDN or IP
address>/api/v1/provisioning/common/sftpconfig' --data '{SftpServer": "10.0.0.0",
"Username": "username", "Password": "password", "RsaKey": "*****"}'
```

GET リクエストもサポートされています。

```
curl -X GET -k -i 'https:// <Expressway FQDN or
IPAddress>/api/v1/provisioning/common/sftpconfig'
```

アップグレードをトリガーする

この例では、JSON API を使用してアップグレードをトリガーします

URL	POST https://<Expressway FQDN or IP address>/api/v1/provisioning/common/upgrade
リクエスト本文	{ "ImageDownloadMode": "sftp", "UpgradeFileLocationSftpPath": "/home/username/upgrade", "UpgradeFileName": "oak.tgz", "UpgradeFileSha512": "*****", "ClusterUpgrade": "no", "WaitForCallDisconnect": 0, "AutoReboot": "yes" }
応答本文	{ "Message": "Upgrade request created successfully" }



(注) 現在、「ImageDownloadMode」は値「sftp」のみをサポートし、「ClusterUpgrade」は「no」のみをサポートします。

この例では、cURL を使用してアップグレードをトリガーします。

```
curl -X POST -k -i 'https:// <Expressway FQDN or IP
address>/api/v1/provisioning/common/upgrade' --data '{"ImageDownloadMode": "sftp",
"UpgradeFileLocationSftpPath": "/home/username/upgrade", "UpgradeFileName": "oak.tgz",
"UpgradeFileSha512": "*****", "ClusterUpgrade": "no", "WaitForCallDisconnect": 0,
"AutoReboot": "yes"}'
```

アップグレードステータス情報を取得する

この例では、JSON API を使用してアップグレードステータス情報を取得します

URL	GET https://<Expressway FQDN or IP address>/api/v1/status/common/upgradestatus

リクエスト本文	この操作では、リクエストの本文は必要ありません。
応答本文	[{ "StatusUpdateTime": "22-05-18T13-23-36", "UpgradeStatus": "UPGD_STARTED" }]

この例では、cURL を使用してアップグレードステータス情報を取得します。

```
curl -X GET -k -i 'https:// <Expressway FQDN or  
IPaddress>/api/v1/status/common/upgradestatus'
```

「UpgradeStatus」 応答 :

- UPGD_INITIATED (すべての SFTP の詳細 - sftp サーバー (IP または FQDN) 、ユーザー名、パスワード、および rsakey が存在する必要があります)
- UPGD_FAILED (SFTP サーバーが構成されていない場合)
- UPGD_FILE_DOWNLOADING
- UPGD_FILE_DOWNLOADED (所要時間は、システムと SFTP サーバー間のファイル転送速度によって異なります)
- UPGD_FILE_DOWNLOADFAILED

エラーの理由

- 構成が正しくない場合
 - SFTP サーバー接続がタイムアウトした場合 (30 秒)
 - SHA512sum チェックに失敗した場合
 - UPGD_WAIT_FOR_CALLDISCONNECT (アップグレード API の構成値に基づく待機時間)
 - UPGD_STARTED
 - UPGD_REBOOT (タイムアウトは UPGD_STARTED から 15 分後)
 - UPGD_FAILED
- 失敗理由。
- フレームワークのインストールの失敗
 - フレームワークのインストールの失敗 - アップグレードフラグが作成されていません
 - UPGD_SUCCESSFUL (再起動後)

例 : DNS サーバーの API の使用

DNS サーバー情報の取得

この例では、JSON API を使用して DNS サーバー情報を取得します。

URL	GET https://<Expressway FQDN or IP address>/api/v1/provisioning/common/dns/dnsserver
リクエスト本文	この操作では、リクエストの本文は必要ありません。
応答本文	{ "DefaultDNSServers": { "index": 2, "address": "10.0.0.2" } }

この例では、cURL を使用して DNS サーバー情報を取得します。

```
curl -X GET -k -i 'https://<Expressway FQDN or IP address>/api/v1/provisioning/common/dns/dnsserver'
```

DNS サーバーの追加

この例では、JSON API を使用して、IP アドレスが 10.0.0.2 で指数値が 2 の DNS サーバーを追加します。

URL	POST https://<Expressway FQDN or IP address>/api/v1/provisioning/common/dns/dnsserver'
リクエスト本文	{ "DefaultDNSServers": { "index": 2, "address": "10.0.0.2" } }
応答本文	{ "Message": "The operation was successful" }

この例では、cURL を使用して IP アドレス 10.0.0.2 で指数値が 2 の DNS サーバーを追加します。

```
curl -X POST -k -i 'https://<Expressway FQDN or IP address>/api/v1/provisioning/common/dns/dnsserver' --data '{"DefaultDNSServers": {"index": 2, "address": "10.0.0.2"}}'
```

DNS サーバーの変更

この例では、JSON API を使用して、DNS サーバーの IP アドレスを指数値 2 で変更します。

URL	PUT https://<Expressway FQDN or IP address>/api/v1/provisioning/common/dns/dnsserver
リクエスト本文	{ "DefaultDNSServers":

	<pre>{ "index": 2, "address": "10.0.0.3" }</pre>
応答本文	<pre>{ "Message": "The operation was successful" }</pre>

この例では、cURL を使用して、指数値 2 の DNS サーバーの IP アドレスを変更します。

```
curl -X PUT -k -i 'https://<Expressway FQDN or IP
address>/api/v1/provisioning/v1/common/dns/dnsserver' --data '{"DefaultDNSServers":
{"index": 2, "address": "10.0.0.3"}}'
```

DNS サーバーの削除

この例では、JSON API を使用して指数値 2 の DNS サーバーを削除します。

URL	DELETE https://<Expressway FQDN or IP address>/api/v1/provisioning/common/dns/dnsserver
リクエスト本文	<pre>{ "index": 2 }</pre>
応答本文	<pre>{ "Message": "The operation was successful" }</pre>

この例では、cURL を使用して指数値 2 の DNS サーバーを削除します。

```
curl -X DELETE -k -i 'https://<Expressway FQDN or IP
address>/api/v1/provisioning/common/dns/dnsserver' --data '{"index": 2}'
```

例：NTP サーバーの API の使用

NTP サーバー情報の取得

この例では、JSON API を使用して NTP サーバー情報を取得します。

URL	GET https://<Expressway FQDN or IP address>/api/v1/provisioning/common/time/ntpserver
リクエスト本文	この操作では、リクエストの本文は必要ありません。
応答本文	<pre>{ "index": 5, "KeyId": 1, "Hash": "sha1", "Authentication": "disabled", "Address": "10.0.0.1" }</pre>

この例では、cURL を使用して NTP サーバー情報を取得します。

```
curl -X GET -k -i '<Expressway FQDN or IP
address>/api/v1/provisioning/common/time/ntpserver'
```

NTP サーバーの追加

この例では、JSON API を使用して IP アドレス 10.0.0.2 の NTP サーバーを追加します。

URL	POST https://<Expressway FQDN or IP Address>/api/v1/provisioning/common/time/ntpserver
リクエスト本文	{ "index": 6, "Address": "10.0.0.2", "KeyId": 1, "Hash": "sha1", "Authentication": "disabled" }
応答本文	{ "Message": "The operation was successful" }

この例では、cURL を使用して IP アドレス 10.0.0.2 の NTP サーバーを追加します。

```
curl -X POST -k -i 'https://<Expressway FQDN or IP address>/api/v1/provisioning/common/time/ntpserver' --data '{"index": 6, "Address": "10.0.0.2", "KeyId": 1, "Hash": "sha1", "Authentication": "disabled"}'
```

NTP サーバー情報の変更

この例では、JSON API を使用して、指数値 6 の NTP サーバーの IP アドレスを変更します。

URL	PUT https://<Expressway FQDN or IP address>/api/v1/provisioning/common/time/ntpserver
リクエスト本文	{ "index": 6, "Address": "10.0.0.3", "KeyId": 1, "Hash": "sha1", "Authentication": "disabled" }
応答本文	{ "Message": "The operation was successful" }

この例では、cURL を使用して指数値 6 の NTP サーバーの IP アドレスを変更します。

```
curl -X POST -k -i 'https://<Expressway FQDN or IP address>/api/v1/provisioning/common/time/ntpserver' --data '{"index": 6, "Address": "10.0.0.3", "KeyId": 1, "Hash": "sha1", "Authentication": "disabled"}'
```

NTP サーバーの削除

この例では、JSON API を使用して指数値 6 の NTP サーバーを削除します。

URL	DELETE https://<Expressway FQDN or IP address>/api/v1/provisioning/common/time/ntpserver
リクエスト本文	{ "index": 6 }

応答本文	{ "Message": "The operation was successful" }
------	---

この例では、cURL を使用して指数値 6 の DNS サーバーを削除します。

```
curl -X DELETE -k -i 'https://<Expressway FQDN or IP
address>/api/v1/provisioning/common/time/ntpserver' --data '{"index": 6}'
```

例：スマート ライセンス ステータスを取得するための API の使用

スマート ライセンス ステータスの取得

この例では、スマート ライセンス ステータスを取得します。

URL	GET https://<Expressway FQDN or IP address>/api/v1/status/common/smartlicensing/licensing
リクエスト本文	この操作では、リクエストの本文は必要ありません。
応答本文	{ "ExportControlledFunctionality": "True", "VirtualAccount": "Expressway", "SmartAccount": "testaccount.cisco.com", "Authorization": { "LicenseAuthorizationStatus": "OUT OF COMPLIANCE", "AuthorizationExpires": "May, 05 May 2020 06:13:06 GMT", "NextAuthorizationAttempt": "February, 05 Feb 2020 18:18:07 GMT", "LastAuthorizationAttempt": "February, 05 Feb 2020 06:18:07 GMT" }, "Registration": { "RegistrationStatus": "REGISTERED", "InitialRegistration": "February, 05 Feb 2020 05:59:04 GMT", "RegistrationExpires": "February, 04 Feb 2021 05:54:03 GMT", "NextRenewalAttempt": "August, 03 Aug 2020 05:59:04 GMT" } }

この例では、cURL を使用してスマート ライセンス ステータス情報を取得します。

```
curl -X GET -k -i 'https://<Expressway FQDN or IP
address>/api/v1/status/common/smartlicensing/licensing'
```

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。
リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。
あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。

翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。