

Cisco Meeting Server の基本 API 関数

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[API 要求および応答](#)

[設定](#)

[CMS 2.9以降](#)

[APIオブジェクトの設定](#)

[すでに作成されているAPIオブジェクトの変更](#)

[APIメニューからのAPIオブジェクトの削除](#)

[説明されたAPI要求 \(サードパーティツールを使用 \)](#)

[HTTP POST](#)

[HTTP GET](#)

[HTTP PUT](#)

[HTTP DELETE](#)

[APIクライアント](#)

[POSTER](#)

[郵便局員](#)

[確認](#)

[トラブルシューティング](#)

概要

このドキュメントでは、CMS (Cisco Meeting Server) で使用される 4 つの基本的な API (アプリケーションプログラム インターフェイス) 関数である GET、POST、PUT、DELETE について説明します。CMS 2.9では、Web管理GUIの[Configuration]メニューにAPIメニューがあります。このドキュメントでは、この新しいメニューを確認し、2つの異なるAPIツールについて説明します。Poster および Postman と、それらを CMS の設定に使用方法についても説明します。

前提条件

要件

このドキュメントに特有の要件はありません。

使用するコンポーネント

このドキュメントの内容は、特定のソフトウェアやハードウェアのバージョンに限定されるものではありません。

このドキュメントの情報は、CMS 2.9以降、またはPostmanやPosterなどの異なるAPIクライアントで使用できます。これらのサードパーティツールについては、このドキュメントの「APIクライアント」セクションで説明します。

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、初期（デフォルト）設定の状態から起動しています。対象のネットワークが実稼働中である場合には、どのようなコマンドについても、その潜在的な影響について確実に理解しておく必要があります。

背景説明

CMS用のAPIは、その機能の多くを設定するための非常に柔軟な方法です。記憶する必要があるか、またはここで取り上げるAPI機能は非常に多いため、必ず最新のAPIリファレンスドキュメントを参照してください。本書の執筆時点で、最新のAPIリファレンスガイドは[ここ](#)で参照できます。

API 要求および応答

API通信は、クライアントとサーバ間の要求と応答の関係です。クライアントは、サーバに要求を行います。要求の処理（アクションの完了、またはその実行の拒否）後に、応答が返されます。

API Request and Response



この記事で説明する4つの要求は次のとおりです。

1. GET – 既存の情報を取得します
2. POST – 新しい情報を作成
3. PUT – 既存の情報を変更します。
4. DELETE – 既存の情報を削除します。

これらは、CMSの設定に使用される基本的なAPI要求です。

最も一般的な応答は、200 OKです。その他の応答は、エラー応答である4xxおよび5xxです。

設定

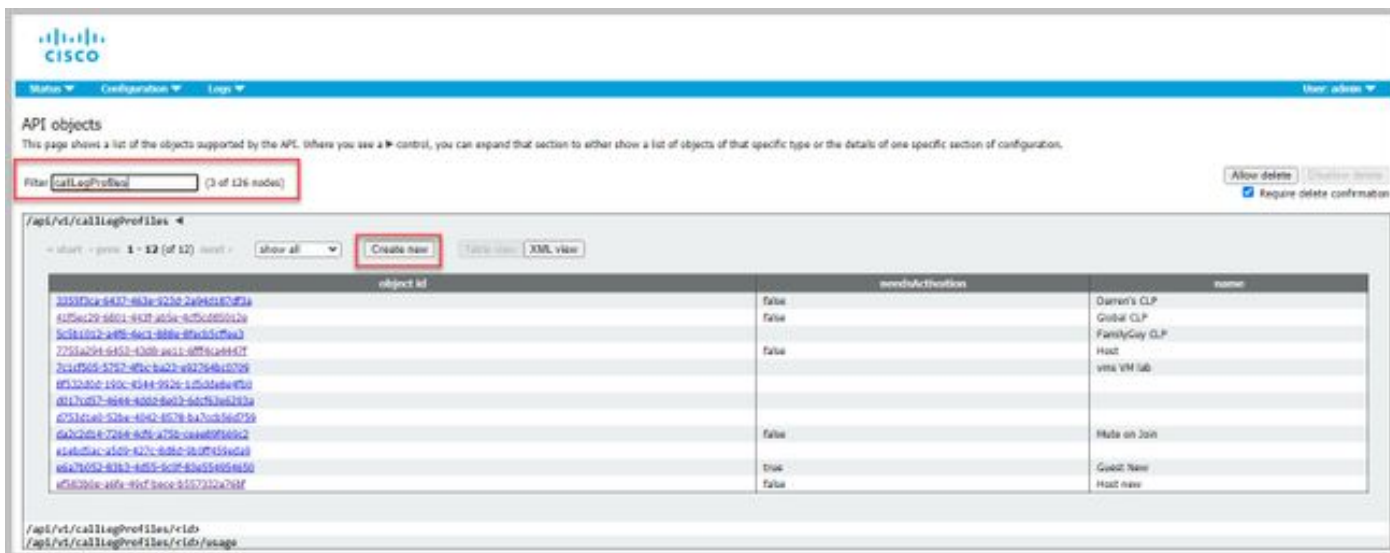
CMS 2.9以降

CMS 2.9では、管理者がCMSの設定を変更したり、設定を微調整したりしやすくする新しいAPIメニューが導入されました。メニューを使用すると、使用可能なすべてのパラメータが表示されるので、設定を簡単に変更したり、新機能を有効にしたりできます。

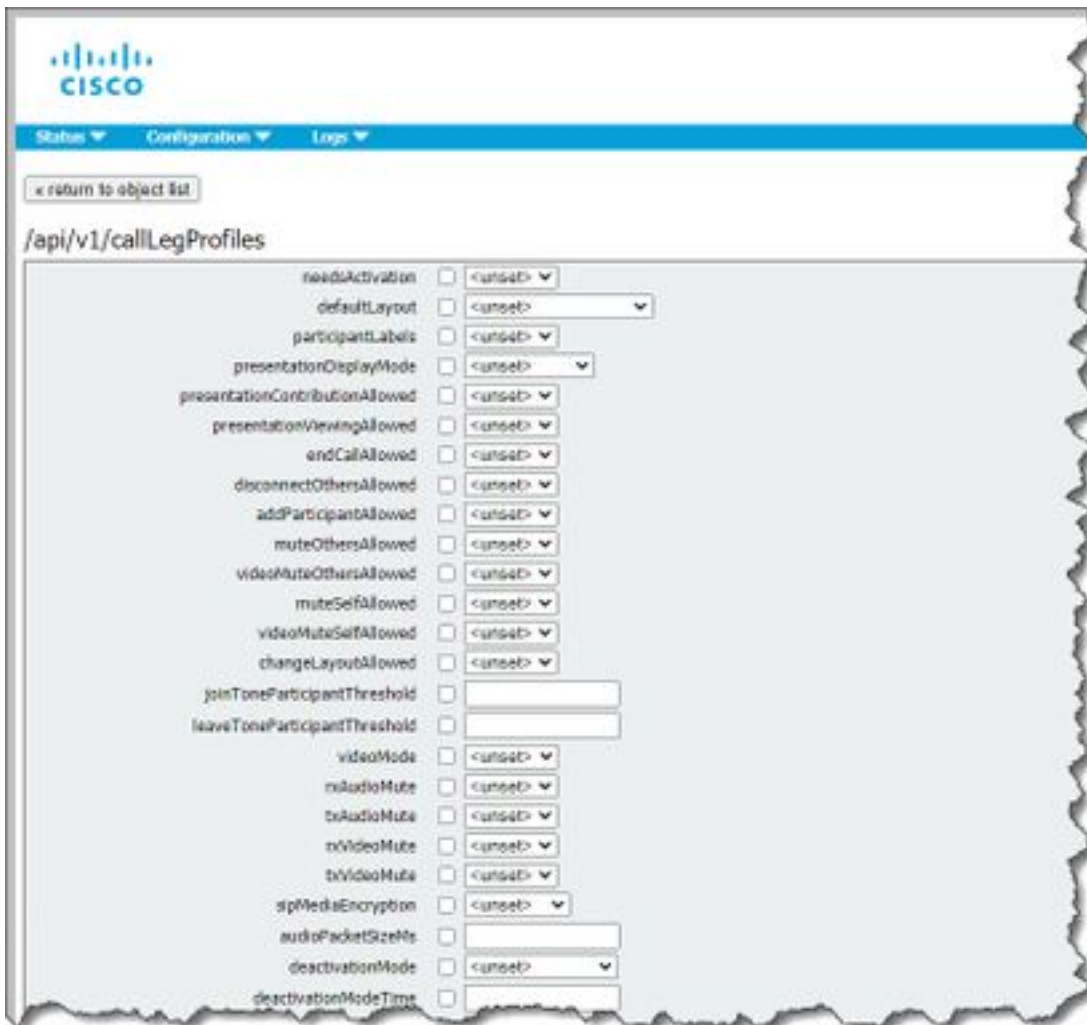


APIオブジェクトの設定

[API]メニューで、APIオブジェクトを編集または作成したいオブジェクトにフィルタし、オブジェクトの横にある黒い矢印をクリックして変更できます。この例では、callLegProfilesを検索し、新しいcallLegProfileを作成する方法を示します。



[Create New]をクリックすると、次の画面が表示され、CallLegProfilesで使用可能なすべてのパラメータが表示されます。特定のパラメータにカーソルを合わせると、各オプションの目的を示すポップアップが表示されます。



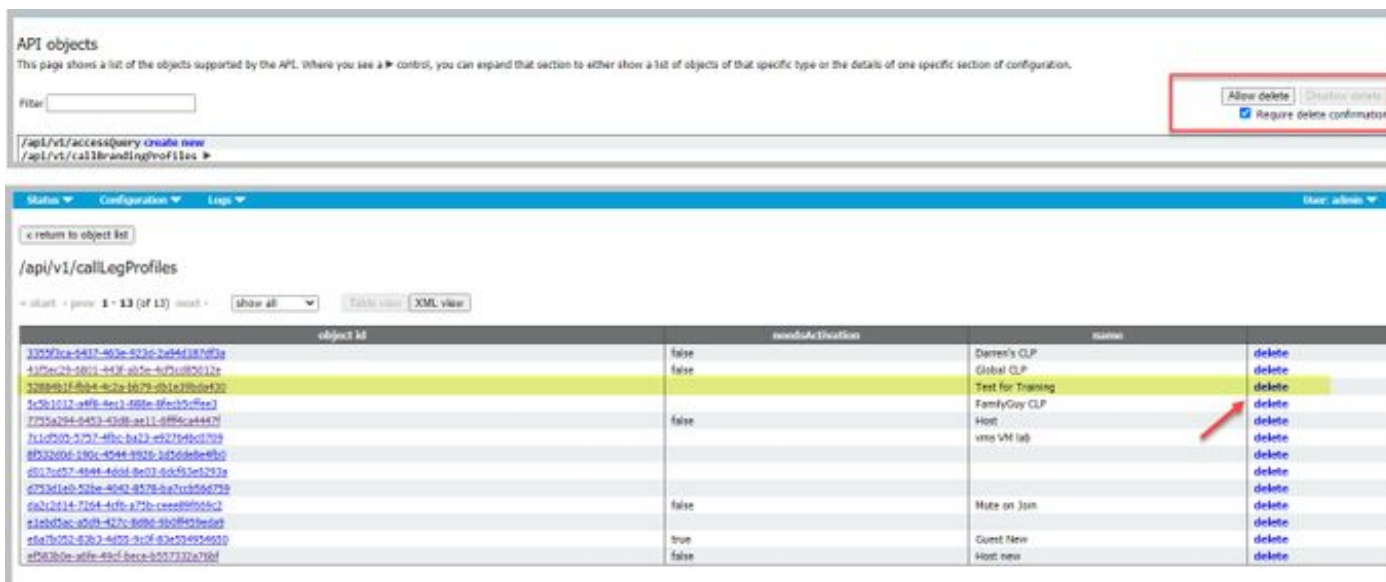
すでに作成されているAPIオブジェクトの変更

オブジェクトの設定を変更すると、下部に[修正]ボタンが表示されます。これは、サードパーティツールのPUTと同じです。



APIメニューからのAPIオブジェクトの削除

オブジェクトを削除するには、メインの[Object List]ページで、アイテムを削除できます。次の例に示すように、削除するオプションを有効にするには、[Allow delete]をクリックします。



説明されたAPI要求 (サードパーティツールを使用)

設定例では、4 つの基本的な要求について説明します。

HTTP POST

ステップ1:POSTを使用してオブジェクトを作成します。

次の例では、この要求を使用して CMS スペースが作成されます。API経由でスペースを作成するには、APIのドキュメントを参照してください。この例ではCMS 2.4 APIガイドを使用しましたが、ここに記載されている最新のAPIガイドを使用する必要があり^{ます}

セクション6.2には、コスペースを作成および修正する方法に関する情報が記載されています。

6.2 Creating and Modifying a coSpace	37
6.2.1 Secondary coSpace URIs	40
6.2.2 Auto-generation of coSpace callId	40

最初の文には、スペースを作成するには /coSpaces に POST を送信する必要があると記載されています。また、スペースの ID は、200 OK のロケーション ヘッダーにあると記載されています。これで、スペースを作成する方法がわかりました。POST を <https://<WebAdminIP>/api/v1/coSpaces> に送信するだけです。

- Creating: POST method to the "/coSpaces" node. If the coSpace was created successfully, a "200 OK" response is received, and the "Location" header contains the ID for the new coSpace

POST のパラメータを指定します。

ドキュメントのセクション6.2には、使用可能なすべてのパラメータをリストした表が表示されています。

Parameters	Type/Value	Description/Notes
name	String	The human-readable name that will be shown on clients' UI for this coSpace
uri	String (URI user part)	The URI that a SIP system would use to dial in to this coSpace. (The URI "user part" is the part before any '@' character in a full URI.)
secondaryUri	String (URI user part)	The secondary URI for this coSpace - this provide the same functionality as the "uri" parameter, but allows more than one URI to be configured for a coSpace. (The URI "user part" is the part before any '@' character in a full URI.)

以下に、いくつかの例を示します。APITestという名前のスペースと、APITestURI の URI ユーザ部分を作成します。

コンテンツ タイプは application/x-www-form-urlencoded で、コンテンツは name=APITest&uri=APITestURI です。

このパラメータを追加すると、次の図のように要求が完了します。

```
POST https://<WebAdminIP>/api/v1/coSpaces HTTP/1.1
Host: <WebAdminIP>
Content-Type: application/x-www-form-urlencoded
Content-Length: 27
Authorization: Basic YWRtaW46QzFzYzBDMXNjMA==
Connection: keep-alive
```

```
name=APITest&uri=APITestURI
```

次の図に、以前の要求に対する応答を示します。

```
HTTP/1.1 200 OK
Server: Apache
X-Frame-Options: DENY
Strict-Transport-Security: max-age=31536000; includeSubDomains
Location: /api/v1/coSpaces/70ca0ed7-4e50-428c-b9ab-4e68faeb86ae
Vary: Accept-Encoding
Content-Encoding: gzip
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Transfer-Encoding: chunked
```

応答内のロケーション ヘッダーに注意してください。

```
Location: /api/v1/coSpaces/70ca0ed7-4e50-428c-b9ab-4e68faeb86ae
```

70ca0ed7-4e50-428c-b9ab-4e68faeb86ae は、新しく作成されたスペースの ID です。このIDは、同じスペースを対象とする将来のAPI要求を行う必要がある場合に便利です。

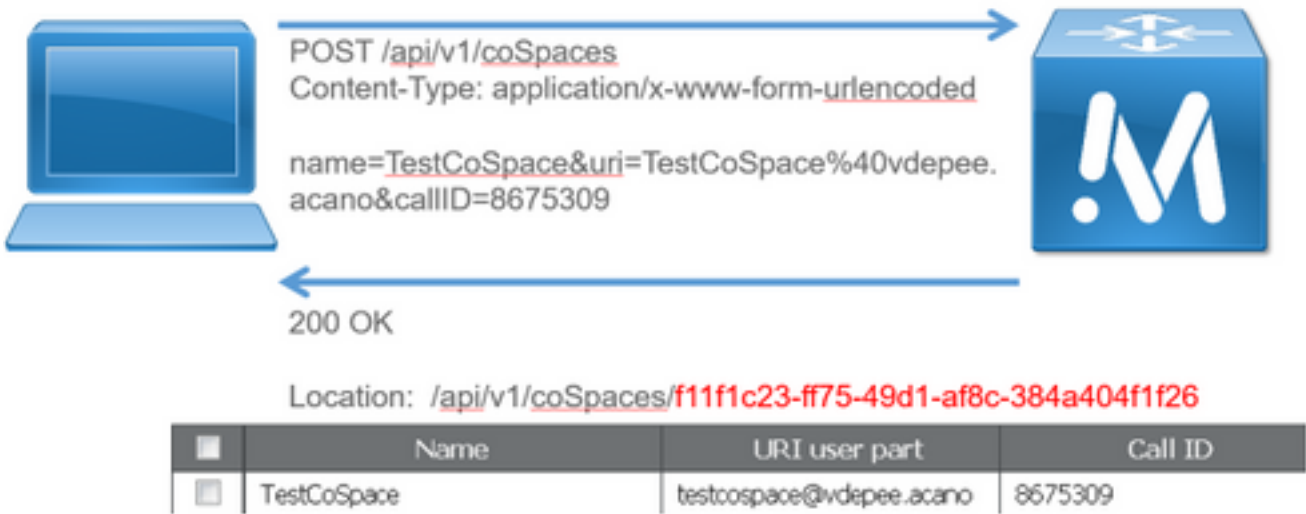
スペースは、CMS の WebAdmin で確認できます。[設定 (Configuration)] > [スペース (Spaces)] に移動します。

	Name	URI user part	Secondary URI user part	Additional access methods	Call ID	Passcode	Default layout
<input type="checkbox"/>	APITest	apitesturi					not set

次の図は、POST 要求をまとめたものです。

HTTP POST

- Creates new object



HTTP GET

ステップ2 : スペースが作成されたら、その設定をプルします。

そのためには、HTTP GET メソッドを使用します。

ロケーション ヘッダーにある、作成されたスペースの ID を使用します。URL は、<https://<WebAdminIP>/api/v1/coSpaces/70ca0ed7-4e50-428c-b9ab-4e68faeb86ae> です。このページで、GET を実行します。

GET 要求の例

```
GET https://<WebAdminIP>/api/v1/coSpaces/70ca0ed7-4e50-428c-b9ab-4e68faeb86ae HTTP/1.1
Host: <WebAdminIP>
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:45.0) Gecko/20100101 Firefox/45.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Cookie: session=logout
Authorization: Basic YWRtaW46QzFzYzBDMXNjMA==
Connection: keep-alive
```

GET 要求に対する応答

```
HTTP/1.1 200 OK
Server: Apache
X-Frame-Options: DENY
Strict-Transport-Security: max-age=31536000; includeSubDomains
Content-Type: text/xml
Vary: Accept-Encoding
Content-Length: 159
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
```

```
<?xml version="1.0"?><coSpace id="70ca0ed7-4e50-428c-b9ab-4e68faeb86ae"><name>APITest</name><autoGenerated>>false</autoGenerated><uri>apitesturi</uri></coSpace>
```

注：応答は、スペースのXMLエンコード設定です。

次の図は、要求GETを要約しています。

HTTP GET

- Retrieves existing information
- No Content in Body



HTTP PUT

ステップ3：スペースを変更します（必要に応じて）。

次の例では、作成されたスペースを変更する方法を示します。たとえば、スペースにセカンダリユーザ部分を追加する必要があるとします。

API のドキュメントを参照してください。このドキュメントによれば、そのために使用する必要があるパラメータは、**secondaryUri** です。

asdf の URI を追加します。POST 用に作成した要求と同じように要求を記述します。

PUT 要求の例

```
PUT https://172.18.105.244/api/v1/coSpaces/70ca0ed7-4e50-428c-b9ab-4e68faeb86ae HTTP/1.1  
Host: 172.18.105.244  
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:45.0) Gecko/20100101 Firefox/45.0  
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8  
Accept-Language: en-US,en;q=0.5  
Accept-Encoding: gzip, deflate, br  
Content-Type: application/x-www-form-urlencoded  
Content-Length: 17  
Cookie: session=b810c447daaeab6cdc6e019c  
Authorization: Basic YWRtaW46QzFzYzBDMXNjMA==  
Connection: keep-alive
```


secondaryUri=asdf

PUT 要求に対する応答

```
HTTP/1.1 200 OK
Date: Tue, 12 Apr 2016 19:11:02 GMT
Server: Apache
X-Frame-Options: DENY
Strict-Transport-Security: max-age=31536000; includeSubDomains
Vary: Accept-Encoding
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Length: 0
```

変更は、CMS の WebAdmin で確認できます。[設定 (Configuration)] > [スペース (Spaces)] に移動します。

	Name	URI user part	Secondary URI user part	Additional access methods	Call ID	Passcode	Default layout
<input type="checkbox"/>	APITest	apitesturi	asdf				not set

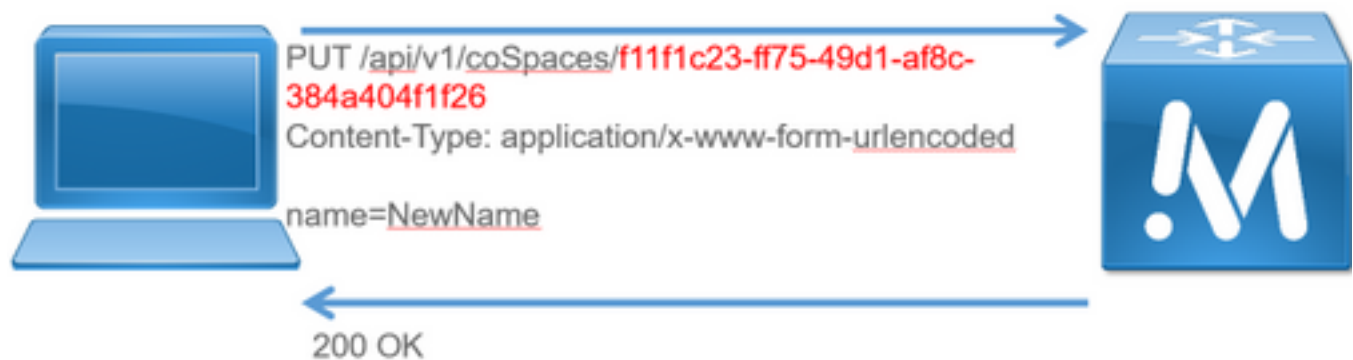
また、GET によっても確認できます。

```
<?xml version="1.0"?><coSpace id="70ca0ed7-4e50-428c-b9ab-4e68faeb86ae"><name>APITest</name><autoGenerated>>false</autoGenerated><uri>apitesturi</uri><secondaryUri>asdf</secondaryUri></coSpace>
```

次の図は、要求PUTを要約しています。

HTTP PUT

- Modifies existing object



	Name	URI user part	Call ID
<input type="checkbox"/>	NewName	testcospace@vdepee.acano	8675309

HTTP DELETE

ステップ 4 : スペースを削除します (必要な場合) 。

DELETE メソッドは、GET メソッドに似ています。

DELETE 要求の例

```
DELETE https://172.18.105.244/api/v1/coSpaces/70ca0ed7-4e50-428c-b9ab-4e68faeb86ae HTTP/1.1
Host: 172.18.105.244
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:45.0) Gecko/20100101 Firefox/45.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Cookie: session=4d13c7ebe739b662dc6e019c
Authorization: Basic YWRtaW46QzFzYzBDMXNjMA==
Connection: keep-alive
```

DELETE 要求に対する応答

```
HTTP/1.1 200 OK
Date: Tue, 12 Apr 2016 19:16:37 GMT
Server: Apache
X-Frame-Options: DENY
Strict-Transport-Security: max-age=31536000; includeSubDomains
Vary: Accept-Encoding
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Length: 0
```

変更は、CMS の WebAdmin で確認できます。[設定 (Configuration)] > [スペース (Spaces)] に移動します。

Name	URI user part	Secondary URI user part	Additional access methods	Call ID	Passcode	Default layout	
						not set	<input type="button" value="Add New"/> <input type="button" value="Reset"/>

また、GET によっても確認できます。

```
<?xml version="1.0"?><failureDetails><coSpaceDoesNotExist /></failureDetails>
```

次の図は、DELETE 要求を要約しています。

HTTP DELETE

- Destroys an object



Name	URI user part	Call ID

API クライアント

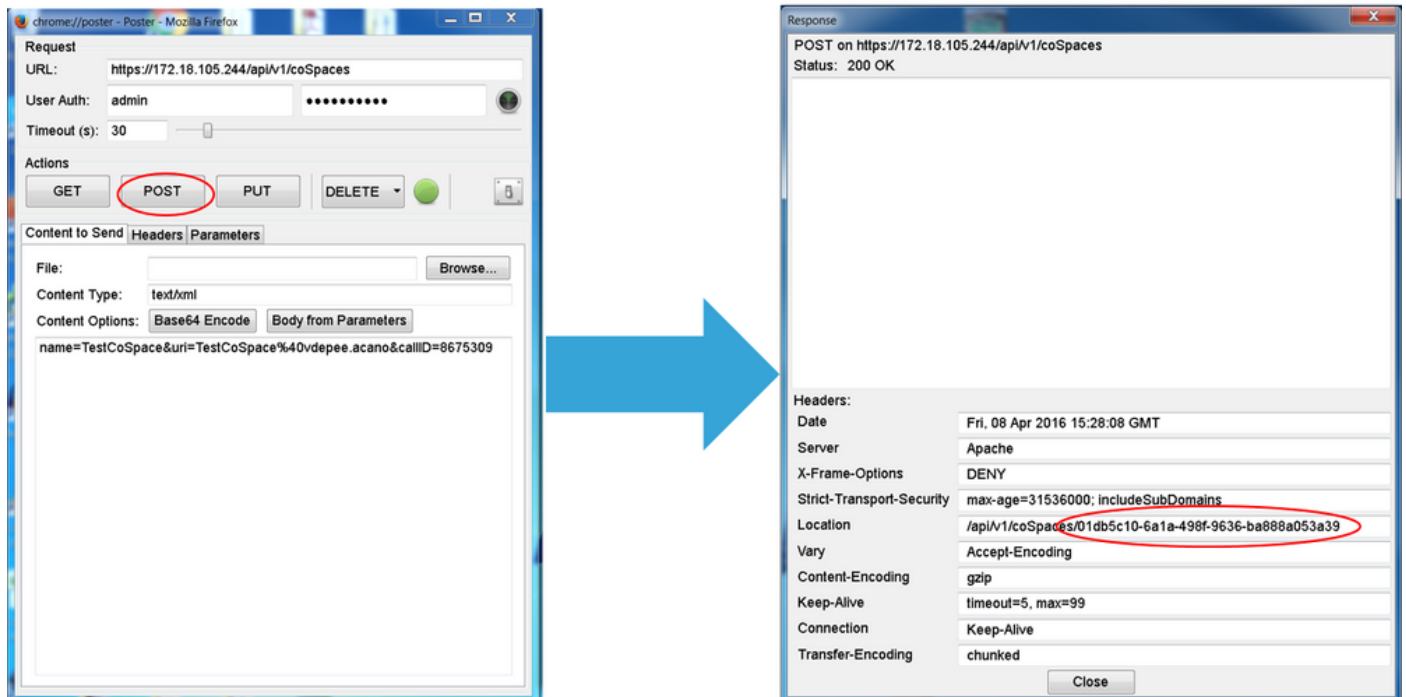
POSTER

POSTER の最上部のボックスには、要求の URL を入力します。

[ユーザ認証 (User Auth)] フィールドには、ユーザ名とパスワードを (この順序で) 入力します。次に、GET または DELETE を実行する場合は、それぞれのボタンを選択します。以下に、いく

つかの例を示します。[GET]をクリックするとポップアップが表示され、応答が表示されます。DELETE の場合は、[DELETE] を選択して緑色のボタンをクリックします。

Poster (Firefox)



POST と PUT の場合は、コンテンツを追加する必要があります。[パラメータ (Parameters)] タブを選択し、パラメータの名前と値を追加します。次に、[送信するコンテンツ]ボタンに戻り、[パラメータからボディ]を選択します。

POST または PUT を送信します。

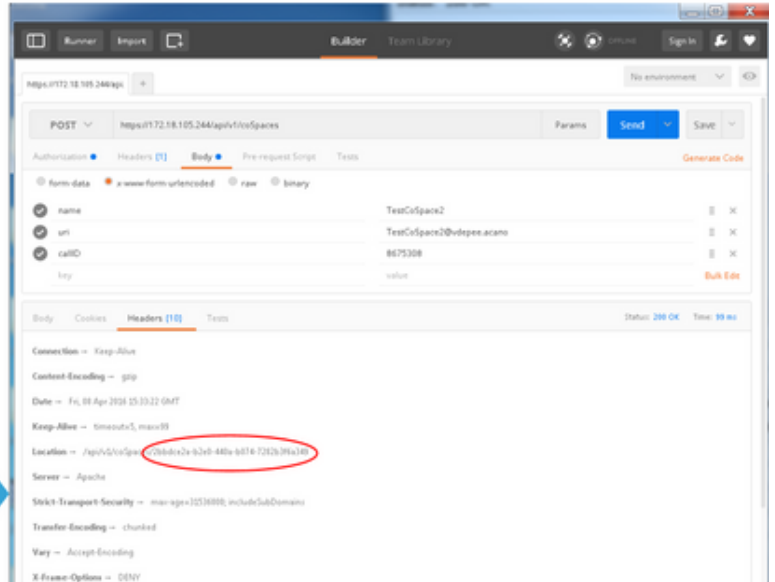
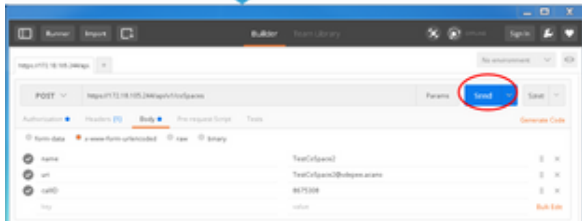
郵便局員

Postmanの左上にあるドロップダウンボックスから使用する方法を選択し、要求URLを入力します。

[Authorization]で[Basic-Auth]を選択し、ユーザ名とパスワードを入力します。次に、[Update Request]を選択します。[ヘッダー (Headers)] タブでは、認証ヘッダーが表示されます。

要求がPOST/PUTの場合は、[本文]タブに移動し、[x-www-form-urlencoded]を選択してパラメータと値を入力します。完了したら、[送信]を選択します。

Postman (Chrome)



確認

検証方法は、要求ごとに説明されます。

トラブルシューティング

現在、この設定に関する特定のトラブルシューティング情報はありません。