

# 思科 Meeting Server 基本 API 函数

## 目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[背景信息](#)

[API 请求和响应](#)

[配置](#)

[CMS 2.9及更高版本](#)

[配置API对象](#)

[修改已创建的API对象](#)

[从API菜单中删除API对象](#)

[解释的API请求 \(使用第三方工具\)](#)

[HTTP POST](#)

[HTTP GET](#)

[HTTP PUT](#)

[HTTP DELETE](#)

[API 客户端](#)

[POSTER](#)

[POSTMAN](#)

[验证](#)

[故障排除](#)

## 简介

本文档介绍 CMS ( 思科 Meeting Server ) 上使用的四个基本 API ( 应用接口 ) 函数 GET、POST、PUT 和 DELETE。自 CMS 2.9 起，Web 管理 GUI 的“配置”(Configuration) 菜单下有一个 API 菜单。本文档将回顾新菜单，并介绍两种不同的 API 工具：Poster 和 Postman，以及如何使用这两个工具配置 CMS。

## 先决条件

### 要求

本文档没有任何特定的要求。

### 使用的组件

本文档不限于特定的软件和硬件版本。

本文档中的信息可与 CMS 2.9 及更高版本一起使用，也可与 Postman 或 Poster 等不同的 API 客户端一起使用。这些第三方工具在文档的 API 客户端部分进行了说明。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您使用的是真实网络，请确保您已经了解所有命令的潜在影响。

## 背景信息

CMS 的 API 是配置其多项功能的极其灵活的方式。本文中回顾或介绍了大量 API 功能，请务必参考当前的 API 参考文档。截止撰写本文时，当前的 API 参考指南在[此处](#)提供

## API 请求和响应

API 通信是客户端和服务端之间的请求响应关系。客户端向服务器发出请求。处理请求（完成操作，或拒绝执行此操作）后，返回响应。

# API Request and Response



本文描述四个请求是：

1. GET — 检索现有信息
2. POST - 创建新信息
3. PUT — 修改现有信息
4. DELETE — 删除现有信息。

这些是用于配置CMS的基本API请求。

最常见的响应为 200 OK，其他响应为4xx和5xx，即错误响应。

## 配置

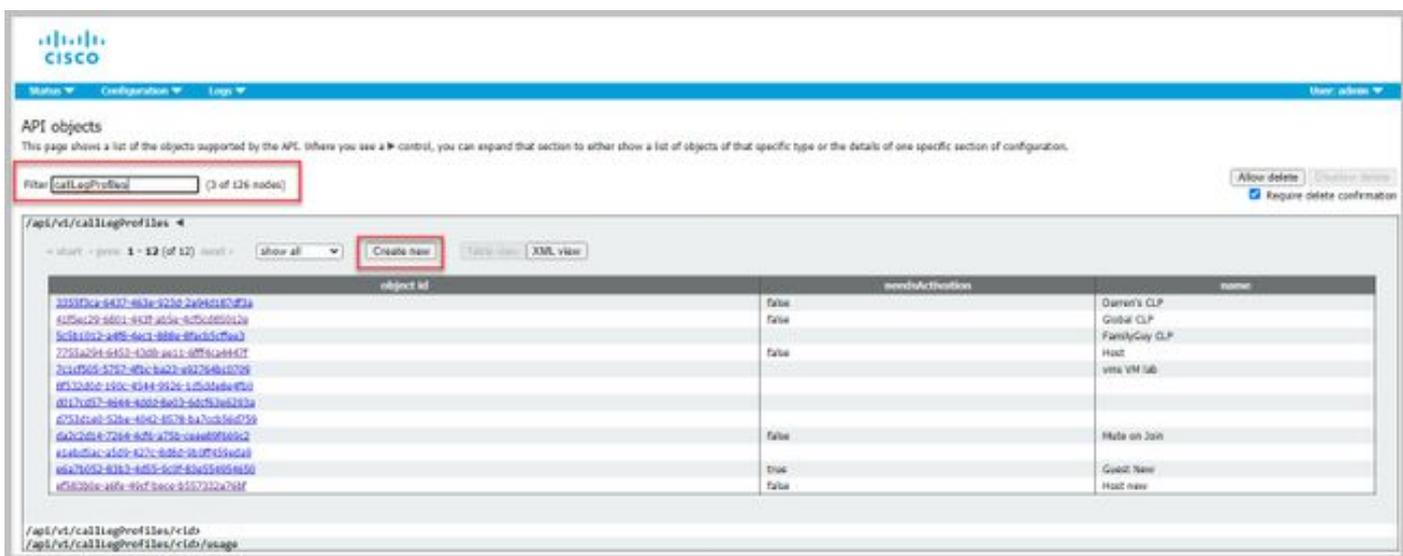
### CMS 2.9及更高版本

CMS 2.9引入了新的API菜单，使管理员能够更轻松地修改CMS中的设置和微调设置。使用菜单时，显示所有可用参数，这样可以快速轻松地更改设置和启用新功能。

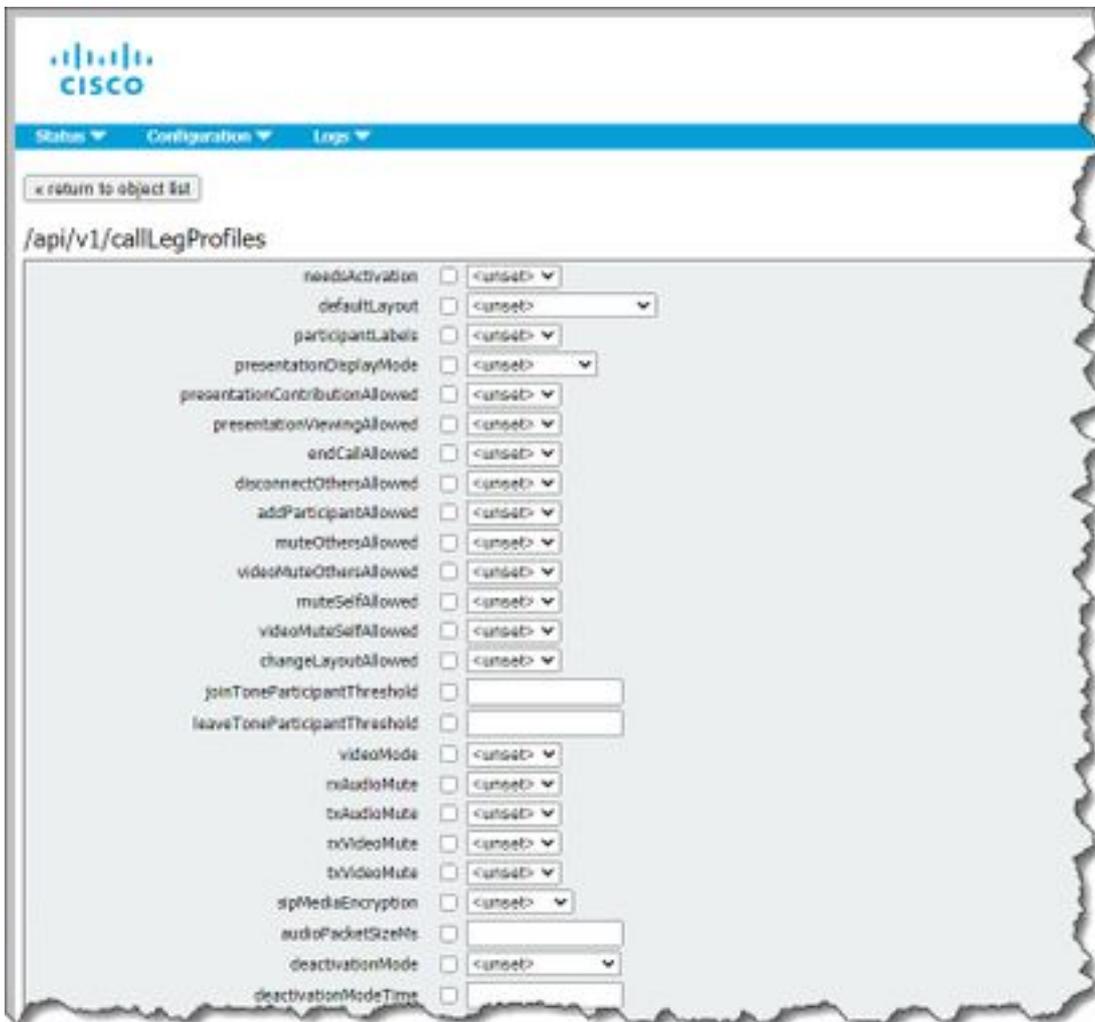


## 配置API对象

进入API菜单后，您可以将API对象过滤为要编辑/创建的对象，然后单击对象旁边的黑色箭头进行这些更改。此示例显示如何查找callLegProfiles并创建新的callLegProfile。

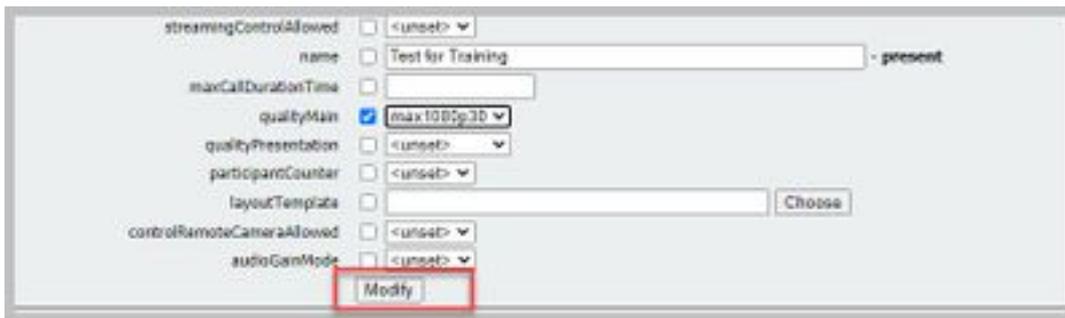


单击“新建”时，将显示此屏幕，该屏幕显示CallLegProfiles的所有可用参数。将鼠标悬停在特定参数上时，将弹出一个显示每个选项用途的窗口。



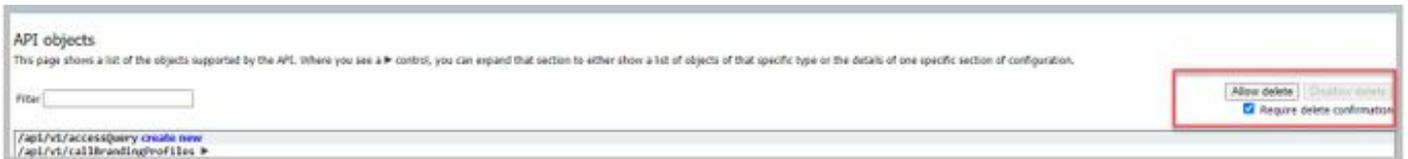
## 修改已创建的API对象

更改对象中的设置时，您将在底部看到“修改”按钮。这与来自第三方工具的PUT相同。



## 从API菜单中删除API对象

要删除对象，可在“对象列表”主页上启用删除项的功能。单击允许删除以启用要删除的选项，如本示例所示：



Status Configuration Logs User: admin

[◀ return to object list](#)

[/api/v1/callLegProfiles](#)

start prev 1 - 13 (of 13) next show all Table view XML view

object id	needsActivation	name	
33057ce-5437-403e-923c-2a1d43876f39	false	Darren's CLP	delete
4375e79-1871-443f-a07e-4c71c082012e	false	Global CLP	delete
328891f-8b24-4c2a-b679-d01e2726b120		Test for Training	delete
5e2b1012-e4f6-4ec3-d86e-8fec35f8e3		FamilyGuy CLP	delete
7795a794-8453-4368-ae11-d886c4442f		Host	delete
711c702-5757-4f8c-b323-e92714bc0109	false	vms VM lab	delete
8f532896-190c-4544-8926-3d54de8e8f0			delete
e012e57-4844-4d4d-8e03-6c453e3233e			delete
e793d1e0-52be-4040-8578-a71cd564799			delete
ca21214-7104-4cfd-a77c-cccc89505c2	false	Mute on Jan	delete
a1ab05ac-8508-4527-8696-8a0c95be8d8			delete
87a7b202-82b7-4d37-810f-83e39495493f	true	Guest New	delete
a70820e-af6e-49cf-bccc-8237332a796f	false	Host new	delete

## 解释的API请求 (使用第三方工具)

我们通过配置示例说明这四个基本请求。

### HTTP POST

步骤1.使用POST创建对象。

在本示例中，使用此请求创建 CMS 空间。要通过API创建空间，请查阅API文档。在本示例中，我使用了CMS 2.4 API指南，但您应使用此处找到的最新API指南

第6.2节包含有关如何创建和修改空格的信息。

<b>6.2 Creating and Modifying a coSpace</b> .....	37
6.2.1 Secondary coSpace URIs .....	40
6.2.2 Auto-generation of coSpace callId .....	40

第一句为，要创建空间，您需要向 /coSpaces 发送 post。后面接着说，空间 ID 位于 200 OK 的“Location”标头中。太好了，您现在知道如何创建空间了。您只需向 <https://<WebAdminIP>/api/v1/coSpaces>发送 POST。

- 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

指定 POST 的参数。

在文档的第6.2节中，您会看到一个表，其中列出了可以使用的所有参数。

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 )

例如：使用名称 **APITest** 并在 URI 的用户部分使用 **APITestURI** 来创建空间

内容类型为 **application/x-www-form-urlencoded**，内容为 **name=APITest&uri=APITestURI**

添加此参数后即完成请求，如下图所示。

```
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

图中显示了之前请求的响应。

```
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
```

请注意响应中的“Location”标头。

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

**70ca0ed7-4e50-428c-b9ab-4e68faeb86ae** 是新创建的空间的 ID。当您需要发出以相同空间为目标的未来API请求时，ID非常有用。

您可以在 CMS 的 WebAdmin 中看到该空间。导航至 **配置 > 空间**。

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

下图总结了 POST 请求。

# HTTP POST

- Creates new object



## HTTP GET

步骤2.创建空间后，请提取该空间的配置。

HTTP GET 用于实现这一目的。

使用“位置”信头中所创建的空间的 ID。URL 为 <https://<WebAdminIP>/api/v1/coSpaces/70ca0ed7-4e50-428c-b9ab-4e68faeb86ae>。在此页面上执行 **GET**。

GET 请求示例：

```
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 请求的响应：

```
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>
```

注意：响应是空间的XML编码配置。

图像汇总了请求GET。

## HTTP GET

- Retrieves existing information
- No Content in Body



## HTTP PUT

步骤3.对空间进行更改（如果需要）。

此示例显示如何修改创建的空间。设想一下向空间添加辅助用户部分。

请参阅 API 文档。其中介绍了需要使用的参数为：**secondaryUri**。

添加 asdf 的 URI。编写一个请求（看起来类似于为 POST 创建的请求）。

PUT 请求示例：

```
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 请求的响应：

```

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

```

您可以在 CMS 的 WebAdmin 中看到所做的更改。导航至 **配置 > 空间**。

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

通过 GET :

```

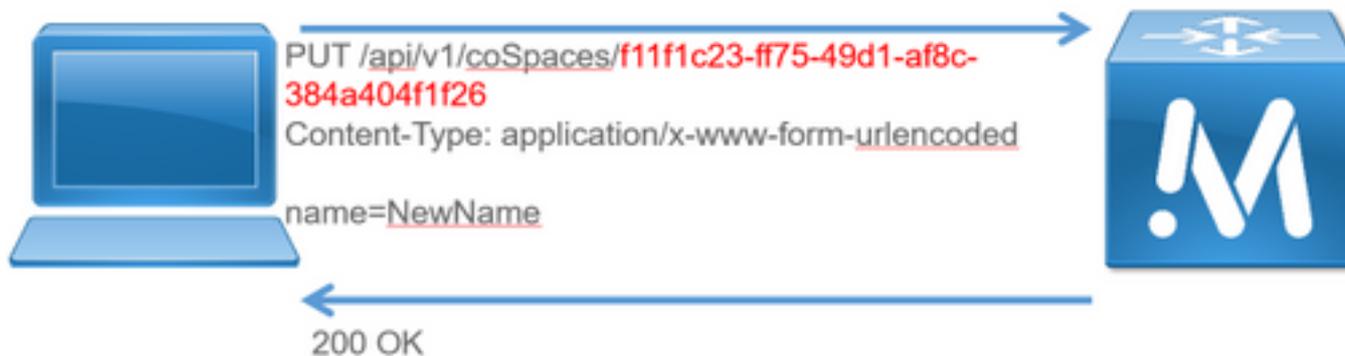
<?xml version="1.0"?><coSpace id="70ca0ed7-4e50-428c-b9ab-4e68faeb86ae"><name>APITest</name><autoGenerated>>false</autoGenerated><uri>apitesturi</uri><secondaryUri>asdf</secondaryUri></coSpace>

```

图像汇总了请求PUT。

## HTTP PUT

- Modifies existing object



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

## HTTP DELETE

第 4 步：删除空间（如果需要）。

DELETE 方法类似于 GET 方法。

DELETE 请求示例：

```

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 请求的响应：

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

您可以在 CMS 的 WebAdmin 中看到所做的更改。导航至配置 > 空间。

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"/>

通过 GET：

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

图像汇总了请求DELETE。

## HTTP DELETE

- Destroys an object



Name	URI user part	Call ID

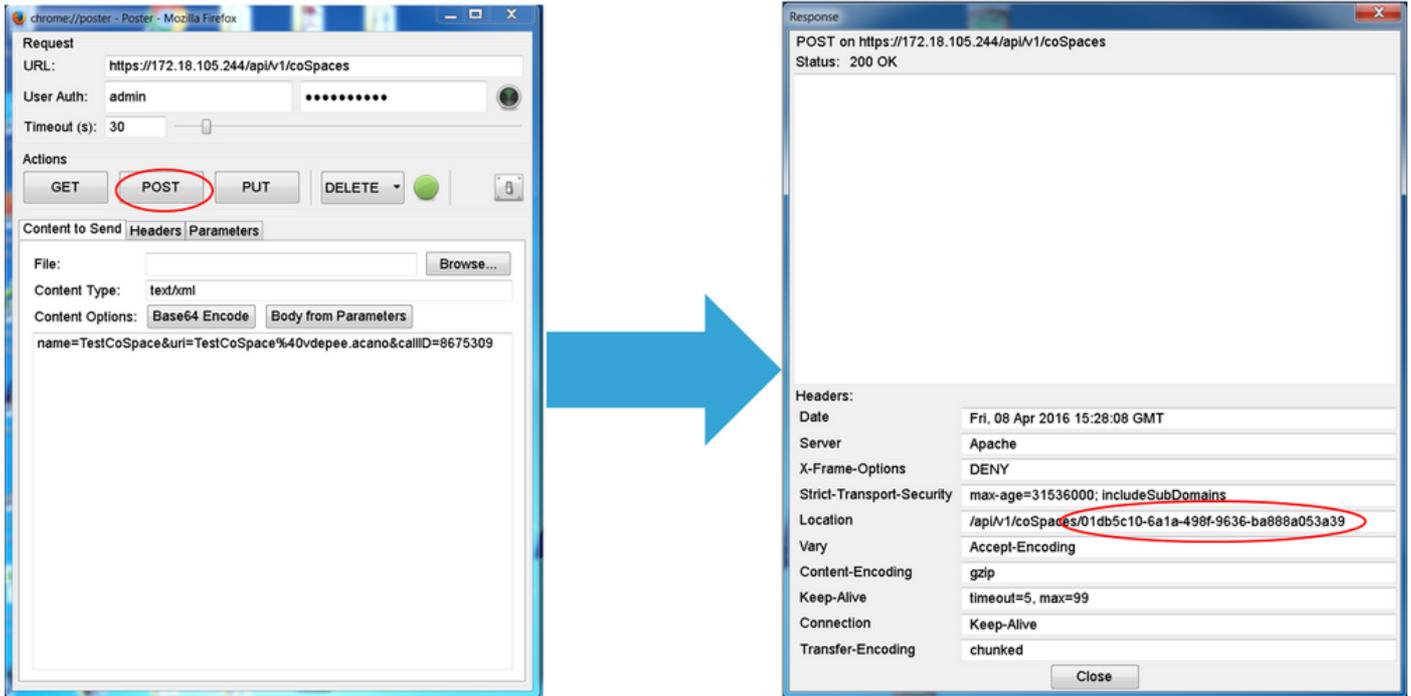
## API 客户端

### POSTER

您可以在 Poster 顶部的框中输入请求的 URL。

您可以按顺序在“用户身份验证”字段中输入用户名和密码。然后，如果执行GET或DELETE操作，请选择相应的按钮。例如：单击GET，系统将弹出一个弹出窗口，显示您的响应。对于DELETE，确保选中DELETE，然后单击绿色按钮。

# Poster (Firefox)



对于 **POST** 和 **PUT**，需要添加内容。选择“参数”选项卡并添加参数的名称和值。然后，返回“要发送的内容”按钮并选择“参数中的正文”。

发送您的 **POST** 和/或 **PUT**。

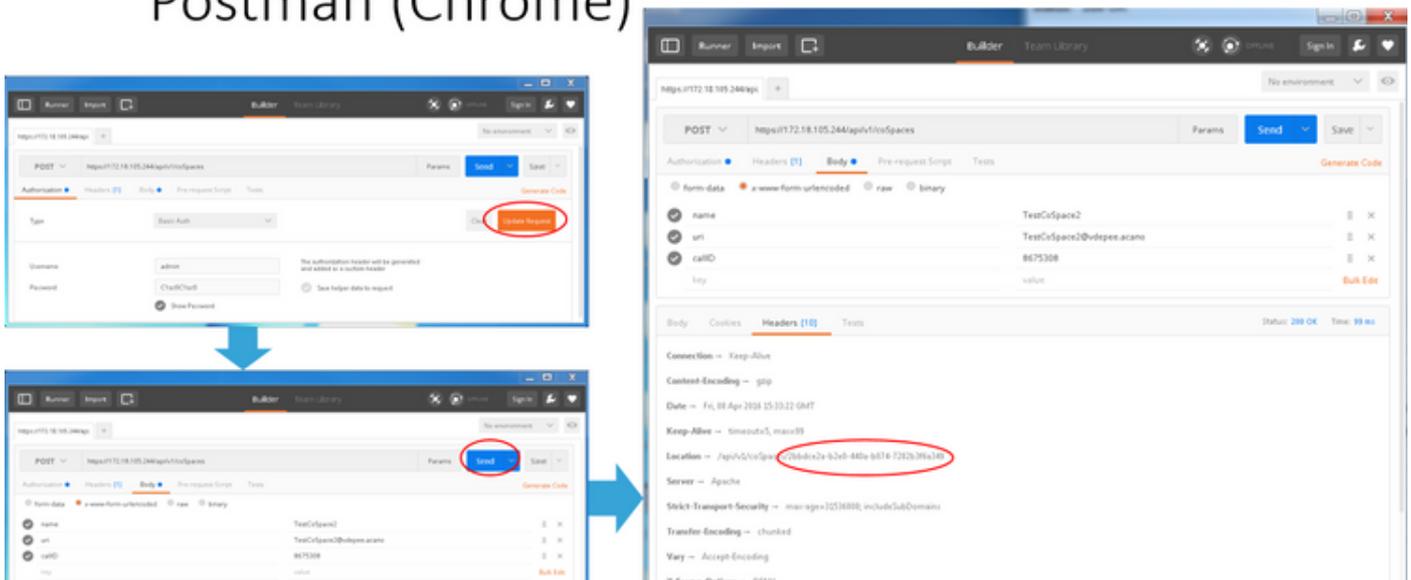
## POSTMAN

在邮递员中，在左上角的下拉框中，选择要使用的方法并输入请求URL。

对于授权，选择**基本身份验证**并输入您的用户名和密码。然后，选择“更新请求”。您可以在“标头”选项卡中看到“授权标头”。

如果您的请求是**POST/PUT**，请导航至“正文”选项卡，选择**x-www-form-urlencoded**并输入您的参数和值。完成后，选择**Send**。

# Postman (Chrome)



## 验证

在每个请求中都说明了验证方法。

## 故障排除

目前没有针对此配置的故障排除信息。