

# Cisco Meeting Server basis-API-functies

## Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Achtergrondinformatie](#)

[API-aanvraag en -respons](#)

[Configureren](#)

[CMS 2.9 en hoger](#)

[Een API-object configureren](#)

[Een API-object wijzigen dat al is gemaakt](#)

[Een API-object uit het API-menu verwijderen](#)

[API-aanvragen toegelicht \(met behulp van gereedschappen van derden\)](#)

[HTTP POST](#)

[HTTP-BEGROTING](#)

[HTTP PUT](#)

[HTTP-VERWIJDERING](#)

[API-clients](#)

[POSTER](#)

[POSTMAN](#)

[Verifiëren](#)

[Problemen oplossen](#)

## Inleiding

In dit document worden de vier basisfuncties van API (Application Program Interface) beschreven: GET, POST, PUT en DELETE, gebruikt op CMS (Cisco Meeting Server). Vanaf CMS 2.9 heeft de web admin GUI een API-menu beschikbaar onder het menu Configuration. Dit document geeft een overzicht van het nieuwe menu en beschrijft ook twee verschillende API-tools: Poster en Postman en hoe ze te gebruiken voor CMS-configuratie.

## Voorwaarden

### Vereisten

Er zijn geen specifieke vereisten van toepassing op dit document.

### Gebruikte componenten

Dit document is niet beperkt tot specifieke software- en hardware-versies.

De informatie in dit document kan met CMS 2.9 en later of met verschillende API klanten zoals

Postman of Poster worden gebruikt. Deze derdengereedschappen worden beschreven in de API-clientsectie van het document.

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u de potentiële impact van elke opdracht begrijpen.

## Achtergrondinformatie

De API voor de CMS is een bijzonder flexibele manier om veel van zijn functies te configureren. Er zijn veel te veel API-functies om hier te memoriseren of te dekken, dus wees erop dat u de huidige API-referentie-documentatie niet hoeft te raadplegen. Op het moment van schrijven zijn de huidige API-handleidingen [hier](#) beschikbaar.

## API-aanvraag en -respons

API-communicatie is een verzoek-responsrelatie tussen klanten en servers. De client doet een verzoek van de server. Na de behandeling van het verzoek (het invullen van een vordering of het weigeren van een beroep) wordt een antwoord teruggezonden.

# API Request and Response



De vier in dit artikel beschreven verzoeken zijn:

1. Get - Bestaande informatie ophalen
2. POST - hiermee wordt nieuwe informatie gemaakt
3. PUT - Hiermee wijzigt u bestaande informatie
4. VERWIJDEREN - Bestaande informatie verwijderen.

Dit zijn de basis API verzoeken om CMS te configureren.

De meest algemene reactie is een 200 OK. Andere reacties zijn 4xx en 5xx, die foutreacties zijn.

## Configureren

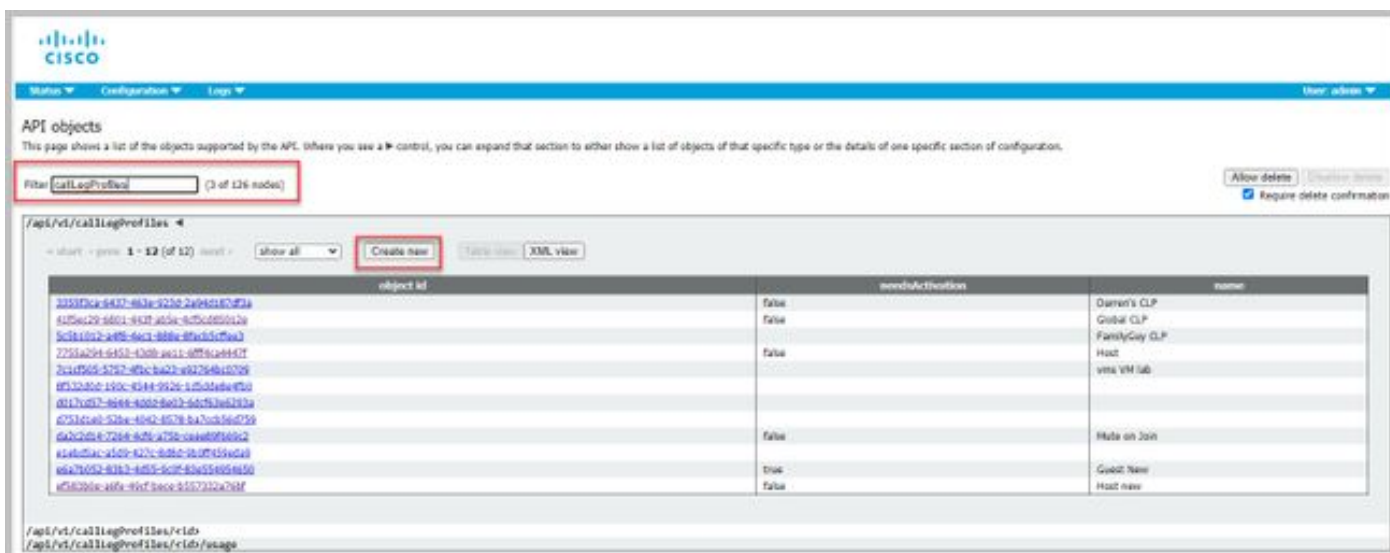
### CMS 2.9 en hoger

CMS 2.9 introduceerde een nieuw API-menu dat het voor beheerders veel gemakkelijker maakt om instellingen aan te passen en instellingen voor fijnafstemming in CMS te wijzigen. Wanneer u het menu gebruikt, worden alle beschikbare parameters weergegeven, waardoor het snel en gemakkelijk is om instellingen te wijzigen en nieuwe functies mogelijk te maken.

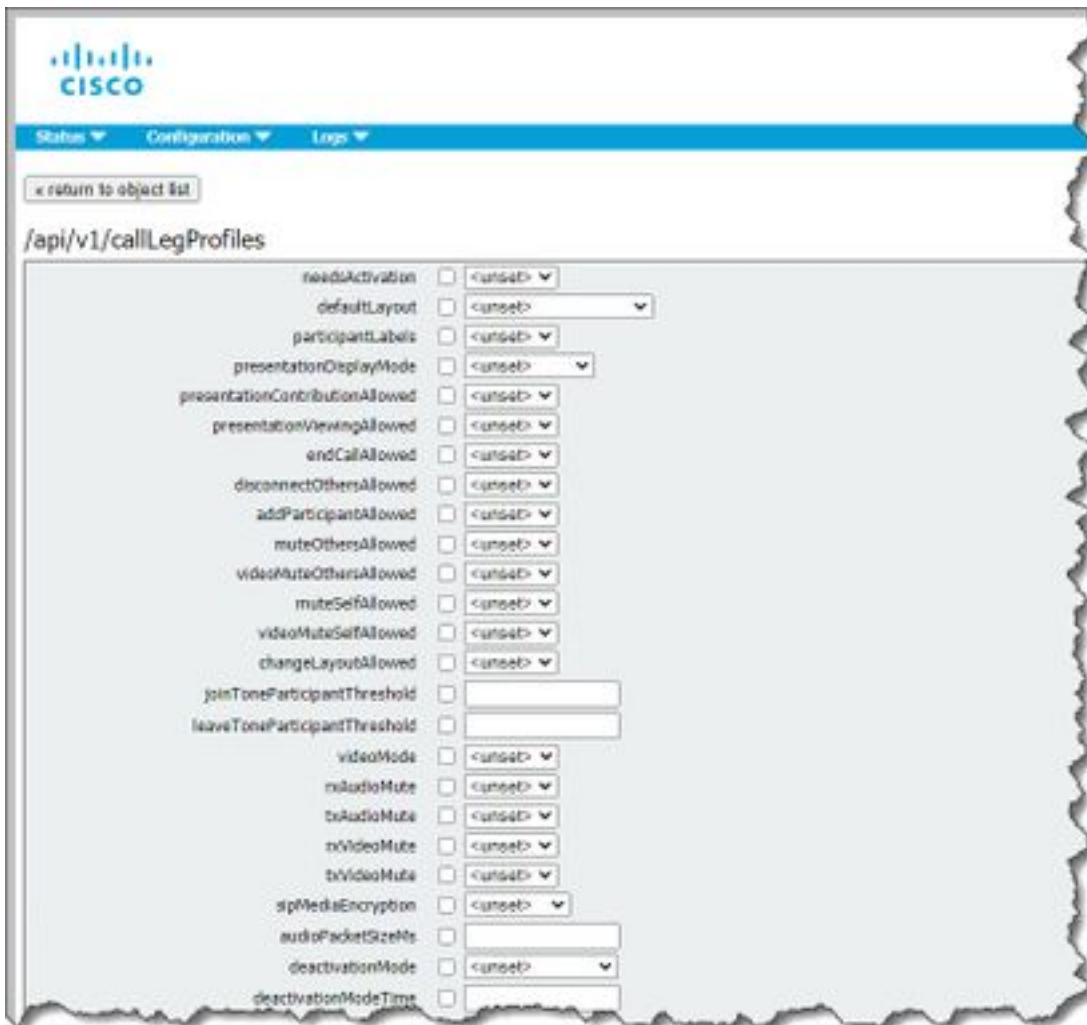


## Een API-object configureren

Enmaal in het API-menu kunt u de API-objecten filteren op wat u wilt bewerken/maken en vervolgens op de zwarte pijl naast het object klikken om deze wijzigingen aan te brengen. Dit voorbeeld toont hoe te om callBeenProfiles op te kijken en een nieuwe callBeenProfile te creëren.

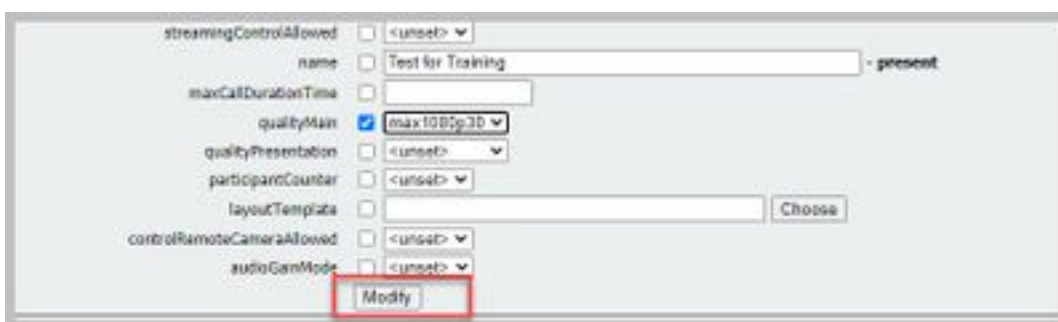


Wanneer u op **Maken New** klikt, wordt u met dit scherm getoond dat alle beschikbare parameters voor CallBeenProfiles toont. Wanneer u over een bepaalde parameter heen zweeft, krijgt u een pop-up die het doel van elke optie toont.



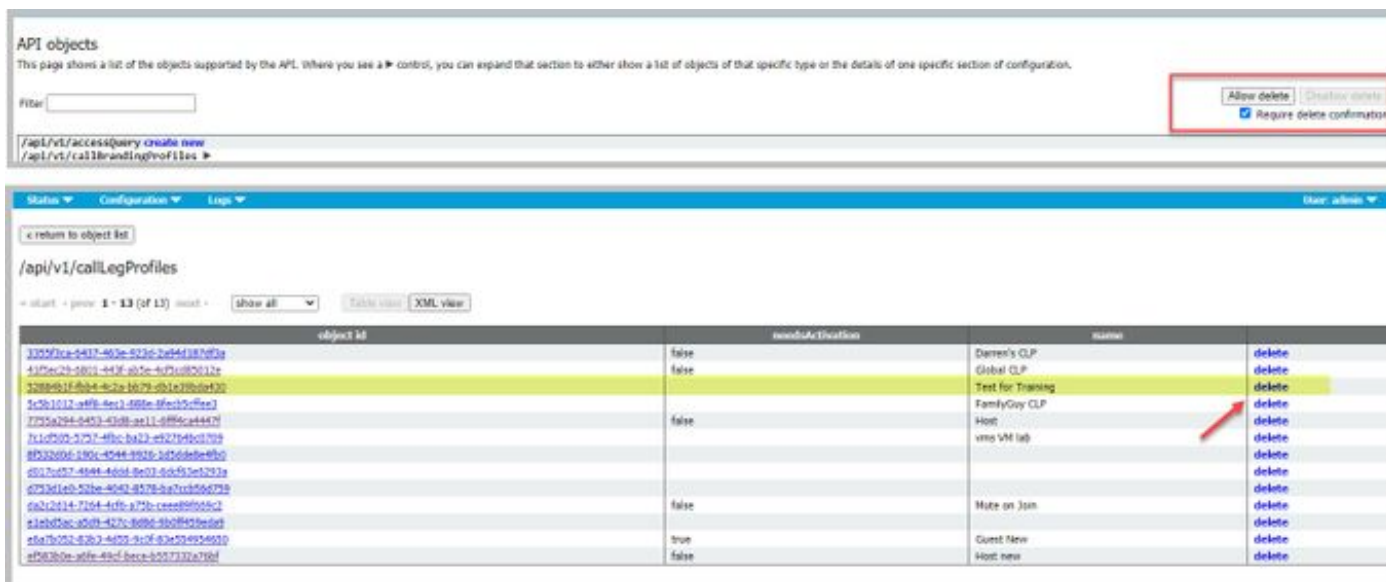
## Een API-object wijzigen dat al is gemaakt

Wanneer u instellingen in een object wijzigt, ziet u de knop Wijzigen onder aan. Dit is hetzelfde als een PUT van de tools van derden.



## Een API-object uit het API-menu verwijderen

U kunt een object op de hoofdpagina van de lijst van objecten verwijderen als u wilt dat een object kan worden verwijderd. Klik op **Verwijderen toestaan** om de optie te verwijderen, zoals in dit voorbeeld wordt getoond:



## API-aanvragen toegelicht (met behulp van gereedschappen van derden)

De vier basisverzoeken worden uitgelegd in een configuratievoorbeeld.

### HTTP POST

Step 1. Gebruik **POST** om een object te maken.

In dit voorbeeld wordt er een CMS Space gemaakt op basis van dit verzoek. Raadpleeg de API-documentatie om de spatie via API te maken. Ik heb bijvoorbeeld de CMS 2.4 API-handleiding gebruikt, maar u moet de meest recente API-gidsen gebruiken, die [hier](#) zijn gevonden

Sectie 6.2 bevat informatie over het maken en wijzigen van een ruimte.

<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

De eerste zin zegt dat om een Ruimte te creëren, je een post moet sturen naar /coSpace. Dan staat er dat het ID van de Space in de locatieheader van 200 OK staat. Geweldig, je weet nu hoe je een ruimte moet creëren. U stuurt alleen een **POST** naar **https://<WebAdminIP>/API/v1/coruimtes**.

- **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

Specificeer de parameters voor de **POST**.

In paragraaf 6.2 van de documentatie ziet u een tabel die alle parameters bevat die u kunt gebruiken.

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 )

Bijvoorbeeld: Een ruimte maken met de naam **APITest** en een URI-gebruikersgedeelte van **APITestURI**

Het type inhoud is **application/x-www-form-urlencoded** en de inhoud is **name=APITest&uri=APITestURI**

Wanneer u deze parameters toevoegt, is het verzoek volledig, zoals in de afbeelding weergegeven.

```
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
```

De afbeelding geeft een antwoord op het vorige verzoek.

```
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
```

Let op de locatie-header in de reactie.

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

**70ca0ed7-4e50-428c-b9ab-4e68faeb86ae** is de ID van de nieuwe gemaakte ruimte. De ID is nuttig wanneer u toekomstige API-verzoeken moet indienen die op dezelfde ruimte zijn gericht.

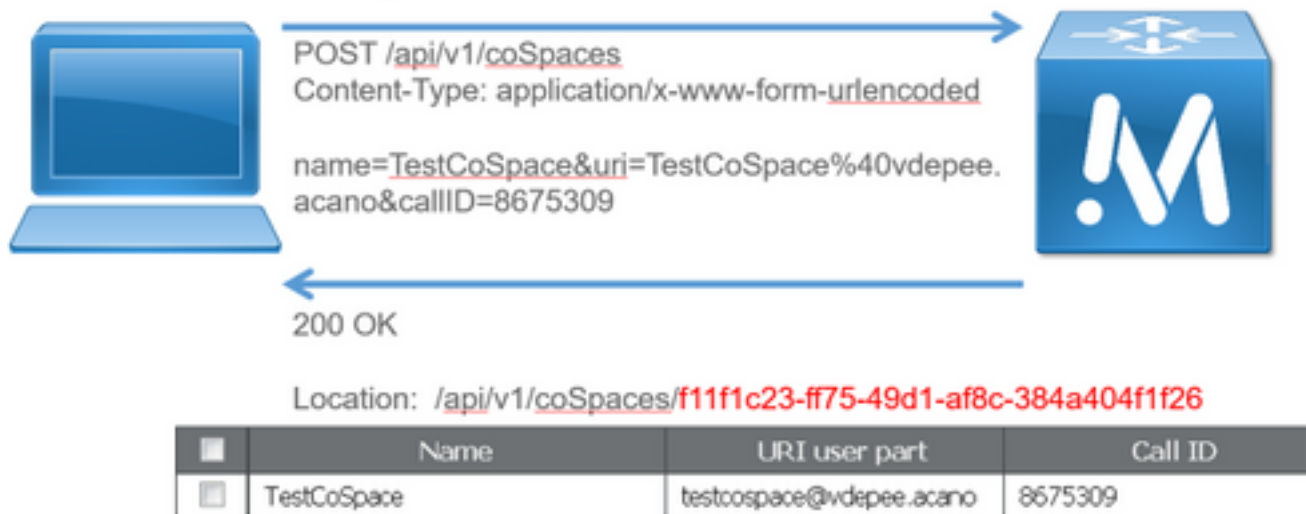
De ruimte kan in WebAdmin van CMS worden gezien. Navigeer naar **Configuratie > ruimtes**.

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

De afbeelding vat de **POST**-aanvraag samen.

## HTTP POST

- Creates new object



## HTTP-BEGROTING

Stap 2. Nadat de ruimte is gemaakt, trekt u de configuratie ervoor.

Gebruik hiervoor de HTTP GET methode.

Gebruik de ID voor de ruimte die is gemaakt vanaf de locatie-header. De URL is <https://<WebadminIP>/api/v1/coSpaces/70ca0ed7-4e50-428c-b9ab-4e68faeb86ae>. Ga op deze pagina een **GET** uitvoeren.

Bijvoorbeeld ONTVANG aanvraag:

```
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
```

Antwoorden op het verzoek van het TARGET:

```
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>
```

**Opmerking:** De respons is een XML gecodeerde configuratie van de Ruimte.

De afbeelding geeft een samenvatting van het verzoek **GET**.

## HTTP GET

- Retrieves existing information
- No Content in Body



## HTTP PUT

Stap 3. Wijzig de ruimte (indien nodig).

Dit voorbeeld toont hoe u de gemaakte ruimte kunt wijzigen. Stel je voor dat er een tweede gebruikersgedeelte aan de ruimte moet worden toegevoegd.

Raadpleeg het API-document. Dit vertelt de parameter die gebruikt moet worden: **secundair Uri**.

Voeg een URI van asdf toe. Schrijf een aanvraag die er ongeveer zo uitziet als het verzoek dat voor de POST is gemaakt.

Voorbeeld PUT-aanvraag:

```
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

## Antwoord op het PUT-verzoek:

```
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
```

U kunt de wijzigingen zien in Webemin van CMS. Navigeer naar **Configuratie > ruimtes**.

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

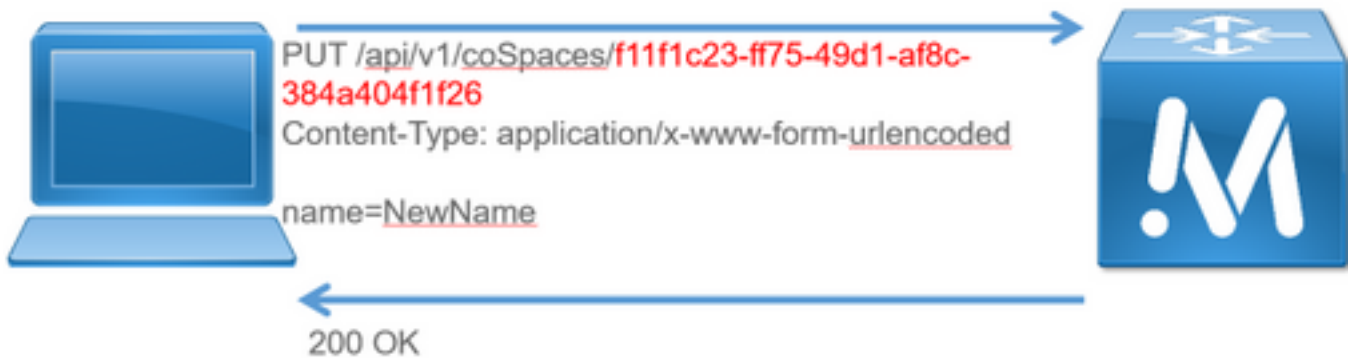
en via een **GET** :

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

De afbeelding vat het **PUT**-verzoek samen.

## HTTP PUT

- Modifies existing object



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

## HTTP-VERWIJDERING

Stap 4. Verwijder de ruimte (indien nodig).

De methode **VERWIJDEREN** lijkt op de methode **GET**.

Voorbeeld verzoek verwijderen:

```
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
```

### Antwoord op het verzoek van de DELETE:

```
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
```

U kunt de wijzigingen zien in Webemin van CMS. Navigeer naar **Configuratie > ruimtes**.

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"/>

en via een **GET** :

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

De afbeelding geeft een samenvatting van het verzoek **VERWIJDEREN**.

## HTTP DELETE

- Destroys an object



Name	URI user part	Call ID

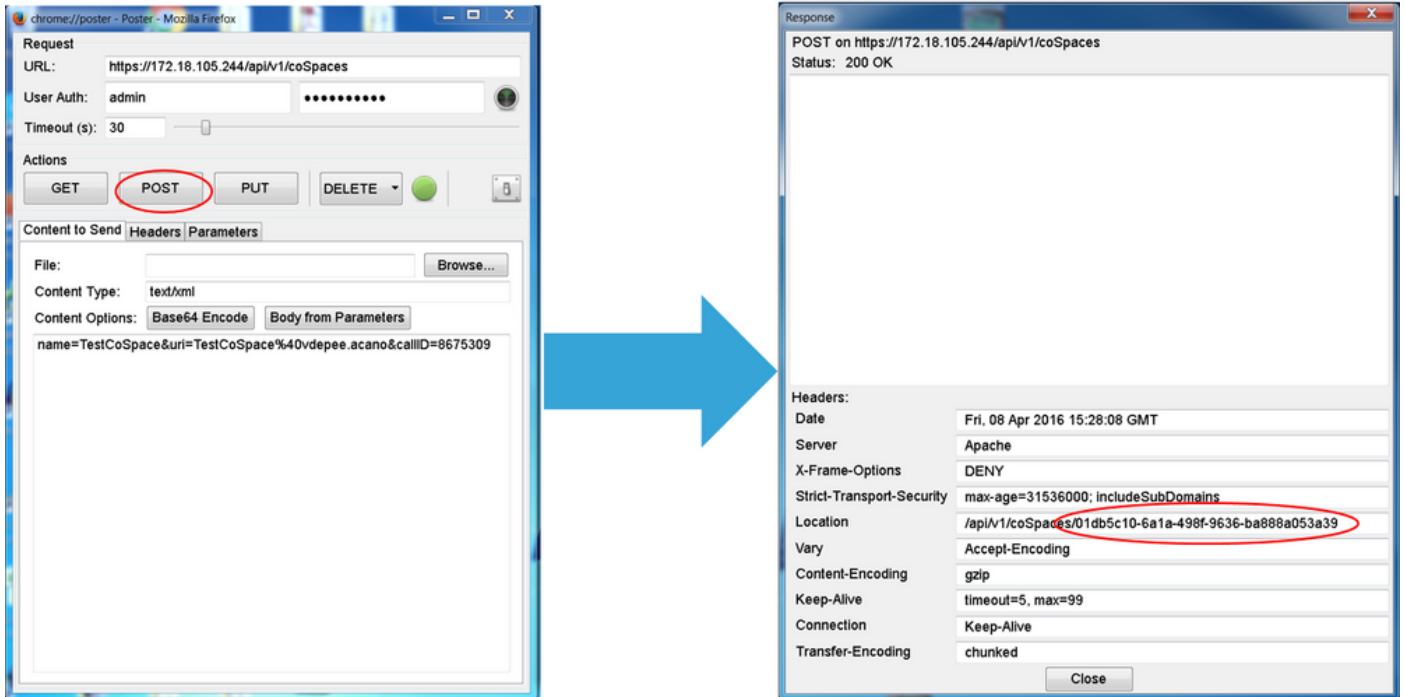
## API-clients

### POSTER

Het bovenste vakje in Poster is waar u de URL voor de verzoeken invoert.

De velden Gebruikershandleiding zijn waar u de gebruikersnaam en het wachtwoord in die volgorde invoert. Kies vervolgens, als u een **KRAAN** of een **VERWIJDERING** doet, de betreffende knoppen. Bijvoorbeeld: klik op **GET** en er verschijnt een pop-up met uw antwoord. Zorg er voor dat **VERWIJDERING** is geselecteerd en klik vervolgens op de groene knop.

# Poster (Firefox)



Voor **POST** en **PUT** moet er inhoud worden toegevoegd. Selecteer het tabblad parameters en voeg de namen en waarden voor de parameters toe. Ga dan terug naar de Content om knop te verzenden en kies **Tekst uit parameters**.

Stuur je post en/of PUT.

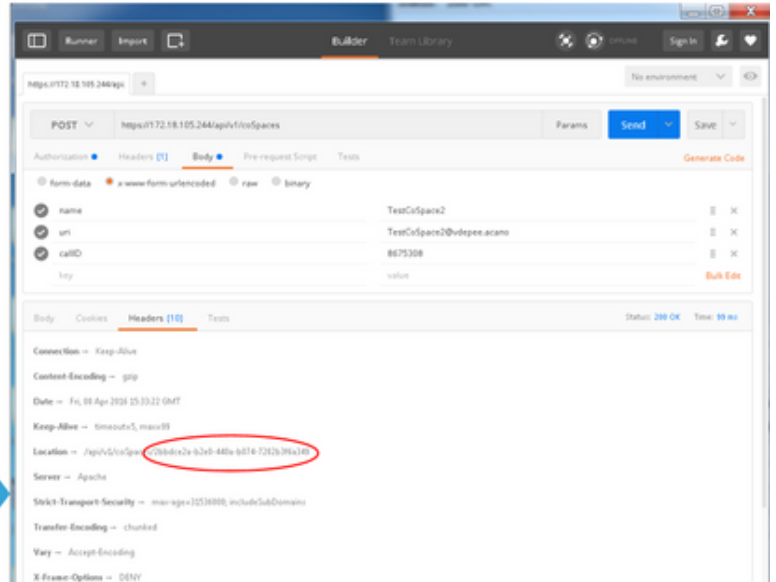
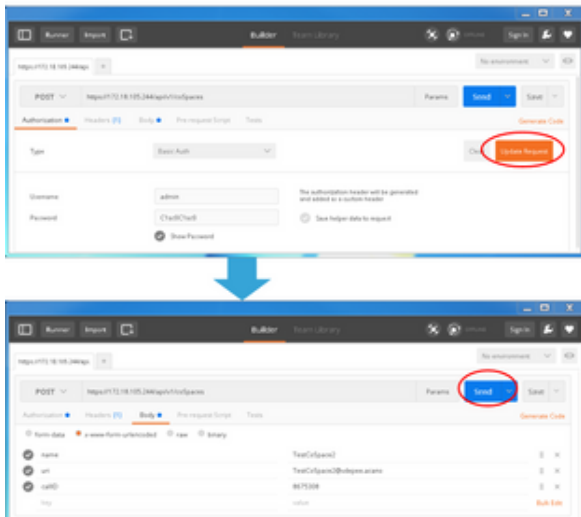
## POSTMAN

In Postman, in de bovenkant-links, kies de Methode die u van het vervolgkeuzevenster wilt gebruiken en voer het verzoek URL in.

Kies voor autorisatie de optie **Basic-Auth** en voer uw gebruikersnaam en wachtwoord in. Selecteer vervolgens de **aanvraag voor bijwerken**. In het tabblad Koppen ziet u een machtigingsheader.

Als uw verzoek een POST/PUT is, navigeer dan naar het tabblad Tekst, kies **x-www-form-urlencoded** en voer uw parameters en waarden in. Selecteer **Verzenden** als u klaar bent.

# Postman (Chrome)



## Verifiëren

De verificatiemethode wordt in elk verzoek toegelicht.

## Problemen oplossen

Er is momenteel geen specifieke troubleshooting-informatie beschikbaar voor deze configuratie.