

Probleemoplossing configureren en toegang tot host op CMS-ruimtes

Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Achtergrondinformatie](#)

[Configureren](#)

[1\) Configuratie met behulp van verschillende URIs](#)

[Verifiëren](#)

[2\) Configuratie waarbij dezelfde URIs worden gebruikt, maar niet-lege gast PIN's/wachtcodes](#)

[Verifiëren](#)

[3\) Configuratie waarbij dezelfde URIs worden gebruikt met een combinatie van lege uitgezonden PIN en niet-lege host PIN](#)

[Verifiëren](#)

[4\) Een gastgebruiker is lid van de ruimte en is geautoriseerd via webRTC loggen in, gastgebruikers voegen zich bij de vergadering met callID. Dezelfde URI en callID wordt gebruikt door gastdeelnemers met lege of niet-lege PIN/passcodes voor gastgebruikers](#)

[Verifiëren](#)

[Problemen oplossen](#)

[Gerelateerde informatie](#)

Inleiding

Dit document beschrijft hoe u Guest en Host Access op ruimtes van uw Cisco Meeting Server (CMS) kunt instellen door API-opdrachten te gebruiken.

Voorwaarden

Vereisten

Cisco raadt kennis van de volgende onderwerpen aan:

- Cisco Meeting Server (CMS) met ruimte die wordt opgezet en in staat is om te bellen
- API-client (zoals poster, Postman) of
- [CMS API-handleiding](#)

Gebruikte componenten

De informatie in dit document is gebaseerd op CMS versie 2.1

De informatie in dit document is gemaakt van een apparaat in een specifieke labomgeving. Als uw netwerk levend is, zorg er dan voor dat u de mogelijke impact van om het even welke opdracht begrijpt.

Achtergrondinformatie

Het document schetst verschillende scenario's:

- Verschillende URIs of call-IDs worden gebruikt door gastdeelnemers
- Dezelfde URI wordt gebruikt door gastdeelnemers wanneer differentiatie wordt gemaakt op basis van PIN of passcode (beide niet leeg)
- Dezelfde URI wordt gebruikt door gastdeelnemers wanneer differentiatie wordt gemaakt op basis van PIN of passcode (mix van leeg/niet leeg)
- Een host-gebruiker is lid van de ruimte en is geautoriseerd via een webRTC-inlogaccount, volgen gastgebruikers zich bij de vergadering aan met CallID. Dezelfde URI en callID wordt gebruikt door gastdeelnemers met lege of niet-lege PIN/passcodes voor gastgebruikers

Configureren

Er zijn vier mogelijkheden om onderscheid te maken tussen Guest- en hostdeelnemers in CMS, beschreven in de volgende vier voorbeelden, en die zijn voornamelijk gebaseerd op verschillende **callBeenProfiles** die het in-call gedrag bepalen van de deelnemers die deelnemen in de ruimte.

Eerst wordt de methode uitgelegd door een andere **URI** (of call-ID) te gebruiken voor gastdeelnemers en gastdeelnemers, en daarna wordt die toegevoegd door gebruik te maken van verschillende wachtcodes (of timeout) op dezelfde **URI**, om het onderscheid te maken tussen gastdeelnemers en gastdeelnemers. De derde methode van een tijdelijke of lege PIN-ingang voor gebruikers van Guest werd geïntroduceerd als een nieuwe optie op CMS 2.1 zoals weergegeven in paragraaf 2.4 van de [introdunctieaantekeningen](#). De vierde methode legt uit hoe de toegang van de Gast en de Host op ruimtes met toegewezen eigenaar/leden moet worden ingesteld en maakt het lid van de ruimte (eigenaar) de gastheer van de ruimte.

1) Configuratie met behulp van verschillende URIs

Dit is de basisconfiguratie die beschikbaar was vóór de release van CMS 2.1 en dezelfde is als voor een andere call-ID. De volgende stappen moeten worden uitgevoerd om het verschil in de Guest/Host-toegang op dezelfde ruimte te krijgen:

1. Maak een gast **callBeenProfile** (**needActivering = True**)
2. Een host-**aanroepBeenProfile** maken (**needActivering = vals**)
3. De gast **CallBeenProfile** toewijzen aan een bestaande of nieuwe ruimte (waarbij de standaard toegangsmethode wordt gebruikt)
4. Maak een nieuwe **accessMethode** op die zelfde ruimte met een verschillende **URI** (en vraag-ID) en verdeel de **host callBeenProfile** er aan

Stap 1. Maak een **gastgesprekBeenProfile** (**behoefteActivering = waar**).

Een **callBeenProfile** bepaalt het in-call gedrag en standaard wijst u het gast in-call gedrag aan de ruimte toe zodat u later een andere toegangsmethode op die zelfde ruimte kunt hebben, evenals

voor host om in te kunnen.

Opmerking: U kunt dit ook toewijzen op huurniveau (`/api/v1/huurders/<huurder-ID>`) of systeemniveau (`/api/v1/systeem/profielen`) om dit bijvoorbeeld toe te passen op alle ruimtes (of per huurder), maar hier wordt het getoond op de ruimte zelf. Houd er rekening mee dat de meest specifieke toewijzing van de **callBeenProfile** in aanmerking wordt genomen voor het in-call gedrag.

De **needActivering** parameter is de belangrijkste hier voor het gast/host-gedrag omdat de deelnemer, indien deze op **waarheid** is ingesteld, geen audio- en video kan ontvangen of bijdragen tot een of meer **volledige/activator** (host) deelnemers meedoen. Andere parameters op de **callBeenProfile** kunnen worden gevonden in sectie 8.4.3 van de [API-handleiding](#), waarbij ook de weergegeven parameters relevant kunnen zijn in deze instelling (afhankelijk van uw vereisten):

- Toegestaan bijdrage
- RXAudio-expressie
- RXVideoMute
- **deactiveringMode** (deactive) | loskoppelen | blijvendegeactiveerd) en **deactivatieTijd** [actie uit te voeren wanneer de laatste activator de oproep verlaat]

Om de gast **callBeenProfile** te maken, moet u een **POST**- verzoek op `/api/v1/callBeenProfiles` maken met de voorkeurparameters en **needActivering** parameter die op **True** is ingesteld, zodat u **nadien met deze uitkomst een GET aanvraag op die callBeenProfile-ID kunt uitvoeren:**

```
needsActivation>true
```

```
needsActivation>
```

```
< deactivationMode>deactivate deactivationMode>
```

Let op de **aanroepBeenProfile-ID** zoals vet weergegeven, aangezien dit op de ruimte in stap 3 moet worden toegepast voor de (standaard) gasttoegang.

Stap 2. Maak een host-**callBeenProfile** (**needActivering = vals**).

Creëer op dezelfde manier het host **callBeenProfile** voor het host in-call gedrag. Dezelfde parameters zijn van toepassing als hierboven vermeld, hoewel de parameters kunnen worden geselecteerd volgens uw eigen voorkeur en vereisten. Het belangrijkste element hier is om de parameter **needActivering** op **vals** in te stellen om hem de host-rol te geven.

U maakt het door een **POST**-aanvraag op `/api/v1/callBeenProfiles` met de voorkeurparameters en **needActivering** parameter die op **vals** is ingesteld, zodat u **een GET aanvraag op die CallBeenProfile-ID kan uitvoeren met deze uitkomst bijvoorbeeld:**

```
needsActivation>
```

```
false
```



```
uricallIdpasscodecallLegProfile>
```

```
7306d2c1-bc15-4dbf-ab4a-1cbdaabd1912
```

```
callLegProfile>
```

Verifiëren

U kunt nu op dezelfde vergadering inbellen:

- Als gast
 - door in te stellen op gastvrijheid.space URI (gevolgd door het domein zoals ingesteld op uw Call matching regels)
 - door de Bel-ID-waarde 628821815 via IVR of WebexRTC aan te sluiten (geen wachtcode)
- Als host
 - door in te stellen op host.space URI (gevolgd door het domein zoals ingesteld op uw call matching regels)
 - door de Bel-ID-waarde 888 via IVR of WebexRTC in te voeren (geen wachtcode)

Als er slechts gasten bij de ruimte horen, worden ze allemaal in een lobby gezet die wacht tot de gastheer toetreedt. Zodra een gastheer toetreedt, worden alle gasten en gastheren in conferentie gezet. Als er geen gastheren meer zijn die op de ruimte zijn aangesloten maar nog steeds gasten, keren ze terug naar het scherm van de lobby zoals in de configuratie van **deactiveren** op **deactivatieMode** parameter op de **vraagBeenProfile** zoals in Stap 1 wordt getoond.

2) Configuratie waarbij dezelfde URIs worden gebruikt, maar niet-lege gast PIN's/wachtcodes

Deze configuratie is vergelijkbaar met die in het vorige voorbeeld en is ook al beschikbaar voor CMS 2.1 release. Het vereist dat zowel de gast als de host een niet-lege PIN of passcode invoeren zodat differentiatie op dat moment kan worden gemaakt, aangezien ze naar hetzelfde **URI** draaien.

De configuratiestappen zijn vrijwel gelijk aan het vorige configuratie voorbeeld:

1. Maak een gast **callBeenProfile** (**needActivering = True**)
 2. Een host-**aanroepBeenProfile** maken (**needActivering = vals**)
 3. Toewijzen van de gast **callBeenProfile** aan een bestaande of nieuwe ruimte die een gastenwachtcode (PIN) (is de standaard toegangsmethode) specificeren
 4. Maak een nieuwe **accessMethode** op dezelfde ruimte met dezelfde **URI** (verschillende **call-ID**) en verdeel **callBeenProfile** aan het met inbegrip van een host passcode (PIN)
- Stap 1. Maak een **gastgesprekBeenProfile** (**behoefteActivering = waar**).

Dezelfde configuratie als in vorig voorbeeld 1 en zelfs dezelfde gast **callBeenProfile** (d4bfe12d-68cd-41c0-a671-48395ee170ab) kan worden gebruikt zoals aangetoond.

Stap 2. Maak een host-**aanroepBeenProfile** (**needActivering = vals**).

Dezelfde configuratie als in eerder voorbeeld 1 en zelfs dezelfde **host callBeenProfile** (7306d2c1-bc15-4dbf-ab4a-1cbdd1912) kan worden gebruikt zoals aangetoond.

Stap 3. Pas de **gastcallBeenProfile** aan een bestaande of nieuwe ruimte toe die een gastenwachtcode (PIN) (die de standaard **accessMethode** is) specificeren.

Evenzo kunt u ofwel een PUT-operatie uitvoeren op een bestaande ruimte (/api/v1/coruimtes/<cospace-ID>) ofwel een POST-operatie om een nieuwe ruimte (/api/v1/coSpaces) te creëren met de gewenste parameters voor de URI, passcode en call-ID bijvoorbeeld, evenals de **CallBeenBeenprofiel** (vanaf stap 1) dat u deze heeft toegewezen volgens sectie 6.2 van de [API-handleiding](#).

Als u een **GET** aanvraag op die ruimte uitvoert moet u een zelfde soort output kunnen zien als deze waar u een **URI** van guestpin.space ziet, een **call-ID** van 189, onze eerder gemaakte **gastcallBeenProfile** en een **passcode** van 789:

```
uricallIdcallLegProfile>
```

```
d4bfe12d-68cd-41c0-a671-48395ee170ab
```

```
callLegProfile<<
```

```
passcode>789
```

```
passcode>
```

Let op de ruimte-ID zoals vet gemarkeerd, aangezien deze moet worden gebruikt om de **accessMethode** op die bepaalde ruimte in stap 4 te maken.

Stap 4. Maak een nieuwe **accessMethode** op die ruimte met dezelfde **URI** (verschillende **call-ID**) en wijs **callBeenProfile** aan het toe met inbegrip van een host passcode (PIN).

Op deze ruimte creëert u ook een andere toegangsmethode voor de hosts (aangezien de gast **callBeenProfile** wordt toegewezen op de ruimte zelf aangezien de standaard optie aansluit), net zoals op het eerste configuratievoorbeeld. Dit gebeurt met behulp van een **POST**-opdracht op /api/v1/coSpaces/<coSpace-ID>/accessMethods met de **coSpace-ID**-waarde voor onze ruimte 22d9f4ca-8b88-4d1-bba9-e2a2f742 c46 zoals in de vorige stap benadrukt. Op deze **POST**-opdracht kunt u de verschillende parameters opgeven, zoals de **URI** (guestpin.space, dezelfde als de oorspronkelijke), **call-ID** (889), **hostBeenProfile** zoals gedefinieerd in stap 2 en de host

passcode of PIN (1234 in dit geval).

Als u een **GET** verzoek op die **accessMethode** uitvoert, moet u een gelijkaardig soort output zien die dezelfde **URI** van `guestpin.space`, een **vraag-ID** van 889, de **host-callBeenProfile** referentie en de **host PIN** van 1234 toont:

```
uricallIdpasscode>1234
```

```
passcode><
```

```
callLegProfile>
```

```
7306d2c1-bc15-4dbf-ab4a-1cbdaabd1912
```

```
callLegProfile>
```

Verifiëren

U kunt nu op dezelfde vergadering inbellen:

- Als gast

- door in te stellen op `guestpin.space` **URI** (gevolgd door het domein zoals ingesteld op uw Call matching regels) en **PIN** 789 in te voeren

- door de **aanroep-ID** waarde 189 via IVR of WebexRTC in te voeren sluit u zich aan bij **PIN** 789

- Als host

- door in te stellen op `guestpin.space` **URI** (gevolgd door het domein zoals ingesteld op uw Call matching regels) en **PIN** 1234 in te voeren

- door de **call-ID** waarde 889 in te voeren via IVR of WebexRTC sluit zich aan bij **PIN** 1234

Als er slechts gasten bij de ruimte horen, worden ze allemaal in een lobby gezet die wacht tot de gastheer toetreedt. Zodra een gastheer toetreedt, worden alle gasten en gastheren in conferentie gezet. Als er geen gastheren meer zijn die op de ruimte zijn aangesloten maar nog steeds gasten, keren ze terug naar het scherm van de lobby zoals in de configuratie van **deactiveren** op **deactivatieMode** parameter op de **vraagBeenProfile** zoals in Stap 1 wordt getoond.

3) Configuratie waarbij dezelfde URIs worden gebruikt met een combinatie van lege uitgezonden PIN en niet-lege host PIN

Deze configuratie is alleen beschikbaar vanaf versie 2.1 van CMS door enkele nieuwe API-opdrachten van **passcodeMode** en **passcodeTime out** in het **CallProfile**-gedeelte. Dit staat voor

een lege PIN voor gasten toe te voegen (het ingaan # of tijd) terwijl de host een PIN heeft om de ruimte te bereiken en het te activeren. De **callProfile** beheerst de in-call ervaring voor SIP (inclusief Lync)-oproepen en is dus niet van toepassing voor CMA-klanten (zowel dikke client als Webex).

De configuratiestappen zijn gelijk aan die van voorbeeld 2, met de toevoeging van CallProfile:

1. Maak een gast **callBeenProfile** (**needActivering = True**)
2. Een **host-aanroepBeenProfile** maken (**needActivering = vals**)
3. Een **CallProfile** maken met de gewenste **passcodeMode**- en **passcodeTime-out**-configuratie
4. Toewijzen van de gast **callBeenProfile** en **callProfile** van stap 3 aan een bestaande of nieuwe ruimte die een gastenwachtcode (PIN) (de standaard toegangsmethode) specificeren) (die de standaard toegangsmethode is)
5. Maak een nieuwe **accessMethode** op dezelfde ruimte met dezelfde **URI** (verschillende **call-ID**) en verdeel **callBeenProfile** aan het met inbegrip van een host passcode (PIN)

Aangezien de configuraties volledig identiek zijn aan de configuratievoorbeelden 1 en 2, zijn er verwijzingen naar die. In feite werd voor de test dezelfde ruimte gebruikt als in voorbeeld 2, maar nu toegevoegd aan het callProfile.

Stap 1. Maak een **gastgesprekBeenProfile** (**behoefteActivering = waar**).

Dezelfde configuratie als in eerder voorbeeld 1 en zelfs dezelfde **gastroepBeenProfile** (**d4bfe12d-68cd-41c0-a671-48395ee170ab**) kan worden gebruikt zoals aangetoond.

Stap 2. Maak een **host-callBeenProfile** (**needActivering = vals**).

Dezelfde configuratie als in eerder voorbeeld 1 en zelfs dezelfde **host callBeenProfile** (**7306d2c1-bc15-4dbf-ab4a-1cbdd1912**) kan worden gebruikt zoals aangetoond.

Stap 3. Maak een **CallProfile** met de gewenste **passcodeMode** en **passcodeTime-out** configuratie.

U kunt een CallProfile maken dat de in-call ervaring voor SIP (inclusief Lync)-oproepen bepaalt. Hier zijn een paar mogelijke configuraties mogelijk, zoals het toestaan van opname of streaming of de maximale deelnemerslimiet, maar de nadruk ligt hier op de nieuwe API-toevoegingen van CMS 2.1 met betrekking tot de passcode verwerking. De andere parameters zijn te vinden in rubriek 8.2 van de [API-handleiding](#).

Er zijn twee parameters die het passacogedrag hier bepalen:

- **passcodeMode**

- **vereist** : IVR wacht tot een gebruiker de PIN of # voor een lege PIN (voor gasten) invoert

- **tijd** : de IVR wacht op een **wachtcodeTime out** hoeveelheid seconden voor de deelnemer om de PIN in te voeren, en indien er niet binnen die tijd is ingevoerd, wordt ervan uitgegaan dat een lege PIN (#) is ingevoerd

- **WachtwoordTime-out**: hoeft alleen te worden ingesteld wanneer **passcodeMode** is ingesteld op timeout en de hoeveelheid tijd controleert voordat u passcode als lege code interpreteert

Voer een **POST**-opdracht uit om het **CallProfile** te maken op **/api/v1/callProfiles** (of **PUT** op **/api/v1/callProfiles/<callProfile-ID>** als u een bestaande opdracht wilt wijzigen) met de gewenste parameters voor **passcodeTime**. Als u een **GET** opdracht op dat specifieke **callProfile** uitvoert,

moet u een zelfde soort resultaat krijgen bijvoorbeeld waar u de modus als timeout en een timeout waarde van 5 seconden hebt ingesteld:

```
passcodeMode>timeout
```

```
passcodeMode><
```

```
passcodeTimeout>5
```

```
passcodeTimeout>
```

Merk de **callProfile-ID** op zoals duidelijk in vet aangezien dit moet worden gebruikt om aan de ruimte toe te wijzen om dit in-call gedrag in stap 4 te hebben.

Stap 4. Pas de **gastcallBeenProfile** en de **callProfile** van stap 3 aan een bestaande of nieuwe ruimte toe die een gastenwachtcode (PIN) (de standaard toegangsmethode) specificeren).

Evenzo kunt u een **PUT**-operatie uitvoeren op een bestaande ruimte (**/api/v1/coSpaces/<cospace-ID>**) of een **POST**-operatie om een nieuwe ruimte (**/api/v1/coSpaces**) te creëren met de gewenste parameters voor **URI** en **call-ID** bijvoorbeeld, evenals de **CallProfile (BeenProfile uit stap 1)**. Het verschil met de vorige voorbeelden is de **callProfile** van stap 3 en het feit dat er geen passcode is toegewezen.

Als u een **GET** aanvraag op die ruimte uitvoert, moet u een soortgelijk soort output kunnen zien als dit voorbeeld, waar u de **URI** van `guestpin.space`, een **call-ID** van 189 ziet, de eerder gemaakte **oproepBeenProfile** en de **callProfile** zoals die in stap 3 is ingesteld:

```
uricallIdcallLegProfile>
```

```
d4bfe12d-68cd-41c0-a671-48395ee170ab
```

```
callLegProfile><
```

```
callProfile>
```

```
4b0eff60-e4aa-4303-8646-a7e800a4eac6
```

```
callProfile>
```

Let op de ruimte-ID zoals vet aangegeven, aangezien deze moet worden gebruikt om de

accessMethode op die bepaalde ruimte in stap 5 te maken.

Stap 5. Maak een nieuwe **accessMethode** op dezelfde ruimte met dezelfde **URI** (verschillende **call-ID**) en wijs **callBeenProfile** aan het toe met inbegrip van een host passcode (PIN).

Op deze ruimte creëert u ook een andere toegangsmethode voor de hosts (aangezien de gast **callBeenProfile** wordt toegewezen op de ruimte zelf aangezien de standaard optie aansluit), net zoals op het eerste configuratievoorbeeld. Dit gebeurt met behulp van een **POST**-opdracht op **/api/v1/coSpaces/<coSpace-ID>/accessMethods** waar de **coSpace-ID**-waarde wordt vervangen door **22d9f4ca-8b88-4d1-bba9-e2a2f7 428c46** zoals in de vorige stap voor deze case beschreven. Op deze **POST**-opdracht kunt u de verschillende parameters opgeven, zoals de **URI** (guestpin.space, dezelfde als de oorspronkelijke), **call-ID** (889), host **callBeenProfile** zoals gedefinieerd in Stap 2 en de host passcode of PIN (1234 in dit geval).

Als u een **GET** verzoek op die **accessMethode** uitvoert, moet u een gelijkaardig soort output zien die dezelfde **URI** van guestpin.space, een **vraag-ID** van 889, de **host-callBeenProfile referentie** en de **host PIN van 1234** toont:

```
uricallId>passcode>1234
```

```
passcode><
```

```
callLegProfile>
```

```
7306d2c1-bc15-4dbf-ab4a-1cbdaabd1912
```

```
callLegProfile>
```

Verifiëren

U kunt nu op dezelfde vergadering inbellen:

- Als gast

- door in te stellen op guestpin.space **URI** (gevolgd door het domein zoals ingesteld op uw aanroep-aanpasingsregels) en # als **PIN** in te voeren of door het gebied na 5 seconden te laten uitwijken

- door de **aanroep-ID** waarde 189 via IVR of WebexRTC in te voeren

- Als host

- door in te stellen op guestpin.space **URI** (gevolgd door het domein zoals ingesteld op uw Call matching regels) en **PIN** 1234 in te voeren

- door de **call-ID** waarde 889 in te voeren via IVR of WebexRTC voegen zich bij de **PIN**

1234

4) Een gastgebruiker is lid van de ruimte en is geautoriseerd via webRTC loggen in, gastgebruikers voegen zich bij de vergadering met callID. Dezelfde URI en callID wordt gebruikt door gastdeelnemers met lege of niet-lege PIN/passcodes voor gastgebruikers

De volgende stappen moeten worden uitgevoerd om het verschil in toegang tot de Gast/de Host te bereiken op dezelfde ruimte voor leden en niet-leden van de ruimte:

1. Maak een gast **callBeenProfile** (**needActivering = True**)
2. Een host-**aanroepBeenProfile** maken (**needActivering = vals**)
3. Toewijzen van de gast **callBeenProfile** aan een bestaande of nieuwe ruimte (die de standaard toegangsmethode is)
4. Maak een nieuwe **accessMethode** op die zelfde ruimte met de zelfde **URI** (en vraag-ID) en verdeel de host **callBeenProfile** er aan
5. Pas de gebruiker inzageJID toe aan dezelfde ruimte. (indien niet toegewezen)
6. Voeg die eigenaar/Das de lid gebruiker toe aan dezelfde ruimte en geefBeenProfile toe aan die lid gebruiker

Stap 1. Maak een gastgesprekBeenProfile (**needActivering = True**).

Dezelfde configuratie als in vorig voorbeeld 1 en de gastcallBeenProfile (**bfe7d07f-c7cb-4e90-a46e-4811bbaf6978**) wordt in dit voorbeeld gebruikt.

Noteer de **callBeenProfile-ID** zoals vet weergegeven, aangezien dit op de ruimte in stap 3 moet worden toegepast voor de gasttoegang.

Stap 2 . Maak een host-aanroepBeenProfile (**needActivering = vals**)

Dezelfde configuratie als in eerder voorbeeld 1 en de host callBeenProfile (**0e76e943-6d90-43df-9f23-7f1985a74639**) wordt in dit voorbeeld gebruikt.

Noteer de **callBeenProfile-ID** zoals vet weergegeven, aangezien deze in stap 4 op de **ruimtetoeegangsmethode** moet worden toegepast voor de host-toegang en op het coSpace lid in stap 6.

Stap 3. Pas de **gastarozeBeenProfile** aan een bestaande of nieuwe ruimte (die de standaard **accessMethode** is) toe.

Voer een **PUT**-opdracht op een bestaande ruimte (**/api/v1/coSpaces/<coSpace-ID>**) uit om de ruimte aan te passen of een **POST**-opdracht op **/api/v1/coSpatieplaatsen** om een nieuwe aan te passen met de gast **CallBEENProfile-parameter** zoals deze in stap 1 als het in-belgedrag voor die ruimte wordt gemaakt. U kunt de parameters **URI** en **call-ID** ook voor die ruimte instellen op uw wens volgens sectie 6.2 van de [API-handleiding](#).

Voer een **GET** aanvraag op die ruimte uit (**/api/v1/coSpaces/<coSpace-ID>**) om te controleren of de gast **callBeenProfile** er aan gekoppeld is, evenals de **URI** en **call-ID waarde**. Een voorbeeldoutput met dit voorbeeld creëerde gastcallBeenProfile in stap 1 is deze met een **URI-waarde van global** en **call-ID** van 1234 (geen passcode set), totaal van nietMemberAccess:

```
<?xml version="1.0" ?>
<coSpace id="96d28acb-86c6-478d-b81a-a37ffb0adafc">
```

```

<name>Global</name>
<autoGenerated>>false</autoGenerated>
<uri>global</uri>
<callId>1234</callId>
<callLegProfile>bfe7d07f-c7cb-4e90-a46e-4811bbaf6978</callLegProfile>
<nonMemberAccess>>true</nonMemberAccess>
<secret>0w4O2zTTF0WdL4ymF8D0_A</secret>
<defaultLayout>allEqual</defaultLayout>
</coSpace>

```

Let op de ruimte-ID zoals vet gemarkeerd, aangezien deze moet worden gebruikt om de **accessMethode** op die bepaalde ruimte in stap 4 te maken.

Stap 4. Maak een nieuwe **accessMethode** op die ruimte met dezelfde **URI** (en **call-ID**) en verdeel de **host callBeenProfile** aan het.

U wilt een andere manier om toegang te krijgen tot de ruimte dan de gasttoegang creëren die momenteel de standaard is. Dit gebeurt door een toegangsmethode op de ruimte zelf te specificeren door een POSTopdracht op **/api/v1/coSpaces/<coSpace-ID>/accessMethods** waarbij de **coSpace-ID** de vet gemarkeerde waarde is in stap 3 (**96d28acb-86c6-478d-b81a - a37ffb0adafc**) waarop de **host callBeenProfile** van stap 2 wordt toegepast evenals het zelfde **URI** en **call-ID** veld. U kunt een lege wachtcode toevoegen voor de hosts die via callID verbinding maken (zonder dat u via webRTC inlogt).

Nadat u een aanvraag hebt ingediend voor die ruimtetoegangsmethode (**/api/v1/coSpaces/<coSpace-ID>/accessMethods/<accessMethode-ID>**) moet u een vergelijkbaar soort uitvoer kunnen zien als deze, waarbij u **dezelfdeURI** (global) en **call-ID** (1234) als de speciaal aangesloten **host-callBeenProfile** zoals ingesteld op stap 2 en **host-passcode(12345)**:

```

<?xml version="1.0" ?>
<accessMethod id="c4ecc16e-945f-4e35-ba03-d9b69107b32c">
  <uri>global</uri>
  <callId>1234</callId>
  <passcode>12345</passcode>
  <callLegProfile>0e76e943-6d90-43df-9f23-7f1985a74639</callLegProfile>
  <secret>kff01zTTE0feL4fsdf43w_B </secret>
</accessMethod>

```

Stap 5 . Pas de eigenaar van de gebruiker Jidto aan de ruimte toe. (indien niet toegewezen). Voeg eigenaarJID aan de ruimte toe door **eigenaarJid (user1@evacanoalone.net)** op de ruimte in te stellen met een PUTopdracht op **/API/v1/coSpaces/<coSpace-ID>**

Na een GET aanvraag op die ruimte moet je kunnen zien dat eigenaar Idand eigenaarJidare is toegewezen aan de ruimte:

```

<?xml version="1.0" ?>
<coSpace id="96d28acb-86c6-478d-b81a-a37ffb0adafc">
  <name>Global</name>
  <autoGenerated>>false</autoGenerated>
  <uri>global</uri>
  <callId>1234</callId>
  <callLegProfile>bfe7d07f-c7cb-4e90-a46e-4811bbaf6978</callLegProfile>
  <nonMemberAccess>>true</nonMemberAccess>
  <ownerId>1d942281-413e-4a2a-b776-91a674c3a5a9</ownerId>
  <ownerJid>user1@evacanoalone.net</ownerJid>
  <secret>0w4O2zTTF0WdL4ymF8D0_A</secret>
  <numAccessMethods>1</numAccessMethods>
  <defaultLayout>allEqual</defaultLayout>

```

</coSpace>

Noteer de eigenaarID (1d942281-413e-4a2a-b776-91a674c3a5a9).

Stap 6. Voeg die eigenaarID (1d942281-413e-4a2a-b776-91a674c3a5a9) van stap 5 toe als lid naar de ruimte en geef **BeenProfile** aan die gebruiker toe. Dit gebeurt door het specificeren van **een specifieke gebruikerJid** and **host-callBeenProfile** voor de ruimte zelf (**specifieke coSpaceID**) door een POST opdracht (`/api/v1/coSpaces/<coSpaceID>/coSpaceGebruikers`). Andere parameters op de **coSpaceGebruikers** zie rubriek 6.4.2 van de API-handleiding, waarin ook de tonen van belang kunnen zijn voor deze installatie:

```
<kan vernietigen>trouw</kan vernietigen>

<canAddRemoveLid>True</canAddRemoveMember>

<canChangeName>True</canChangeName>

<canChangeUri>vals</canChangeUri>

<canChangeCallID>vals</canChangeCallID>

<canChangePasscode>True</canChangePasscode>

<canPostMessage>ware</canPostMessage>

<canDeleteAllMessaging>valse</canDeleteAllMessages>

<canRemoveSelf>vals</canRemoveSelf>
```

Controleer of de gebruiker van het lid aan de ruimte is toegevoegd door een GET opdracht (`/api/v1/coSpaces/<coSpaceID>/coSpaceGebruikers?`)

```
<?xml version="1.0" ?>
<coSpaceUsers total="1">
<coSpaceUser id="1d942281-413e-4a2a-b776-91a674c3a5a9">
<userId>1d942281-413e-4a2a-b776-91a674c3a5a9</userId>
<userJid>user1@evacanoalone.net</userJid>
<autoGenerated>>false</autoGenerated>
</coSpaceUser>
</coSpaceUsers>
```

Opmerking over de gebruikerID (indien andere formuliereigenaarID stap 5). Controleer of de hoofdaanroep **BeenProfile** aan **coSpaceGebruiker** is toegewezen door een **GET-aanvraag** waarin u **een specifieke coSpaceID** en **een gebruikersID** (`/api/v1/coruimtes/<coSpaceID>/coSpaceSpaceID`) kunt instellen

```
<?xml version="1.0" ?>
<coSpaceUser id="1d942281-413e-4a2a-b776-91a674c3a5a9">
```

```
<autoGenerated>>false</autoGenerated>
<canDestroy>>true</canDestroy>
```

```
<canAddRemoveMember>>true</canAddRemoveMember>
<canChangeName>>true</canChangeName>
<canChangeUri>>false</canChangeUri>
<canChangeCallId>>false</canChangeCallId>
<canChangePasscode>>true</canChangePasscode>
<canPostMessage>>true</canPostMessage>
<canDeleteAllMessages>>false</canDeleteAllMessages>
<canRemoveSelf>>false</canRemoveSelf>
<canChangeNonMemberAccessAllowed>>true</canChangeNonMemberAccessAllowed>
```

```
0e76e943-6d90-43df-9f23-7f1985a74639
```

```
</coSpaceUser>
```

Verifiëren

U kunt nu op dezelfde vergadering inbellen:

- Als gast

- door in te stellen op URI (gevolgd door het domein zoals ingesteld op uw call matching regels)

- door de waarde 1234 via IVR of WebexRTC in te voeren (geen wachtcode)

- Als host

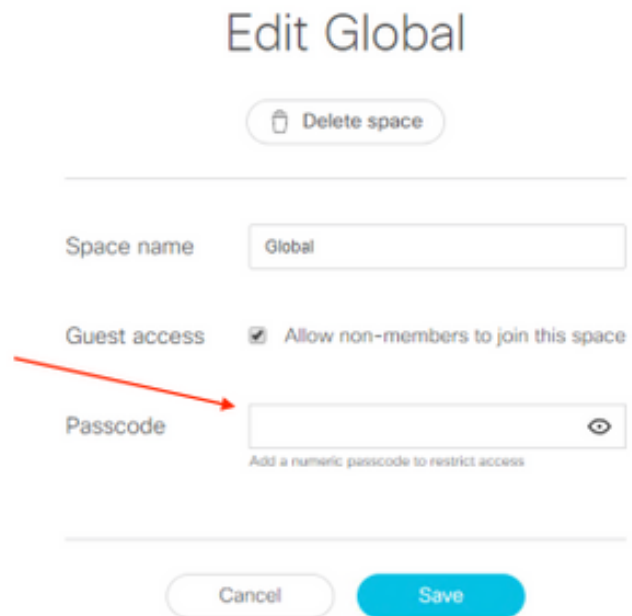
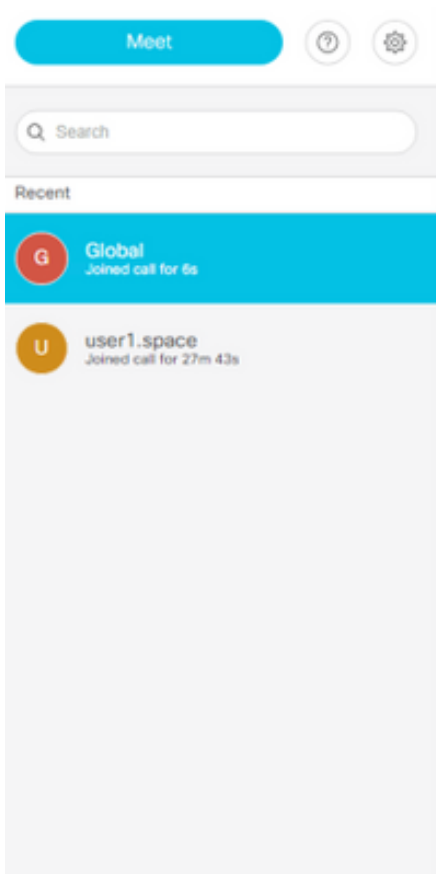
Door in te loggen als gebruiker (een lid van de ruimte met toegewezen "host" callBeenProfile, met user1@evacanoalone.net in dit scenario) via webRTC en zich aan te sluiten bij de ruimte ("global" URI).

- door in te stellen op "global" URI (gevolgd door het domein zoals ingesteld op uw aanroep-aanpassingsregels) en passcode 12345.

- door de waarde 1234 via IVR of WebexRTC in te voeren (met een host-code 12345)

Als er slechts gasten bij de ruimte horen, worden ze allemaal in een lobby gezet die wacht tot de gastheer toetreedt. Zodra een gastheer toetreedt, worden alle gasten en gastheren in de conferentie gezet. Als er geen gastheren meer zijn die op de ruimte zijn aangesloten maar nog steeds gasten, keren ze terug naar het scherm van de lobby zoals in de configuratie van **deactiveren** op **deactivatieMode** parameter op de **vraagBeenProfile** zoals in Stap 1 wordt getoond.

Host (eigenaar/lid) kan een wachtwoord instellen (bewerken/verwijderen) voor gasten direct in een webRTC-app of een niet-aangesloten (gast) toegang voor de ruimte volledig uitschakelen:



Problemen oplossen

Deze sectie verschaft informatie die u kunt gebruiken om problemen met uw configuratie op te lossen.

De registratie van CMS toont ons wel even wanneer u als gast toetreedt of wanneer de eerste gastvrouw toetreedt, maar het is het best te verifiëren met de wensen van GET, de callProfile, en de definities van de gastcallBeenProfile en de toewijzing ervan op de respectieve toegangsmethoden (of de standaard toegangsmethode) of op hoger niveau (mondiaal niveau of huurniveau).

U kunt deze structuur volgen om alle informatie te verkrijgen:

1. **Ga weg op /api/v1/callProfiles** (indien dit met de **passcodeMode** gebruikt)
>Controleer in detail de gewenste CallProfile-ID met **GET** op **/API/v1/callProfiles/<callProfile-ID>**
2. **RAAK OP /API/v1/callBeenProfiles**
>Controleer in detail de gewenste CallBeenProfile-IDs van gast en host met behulp van **GET** op **/api/v1/callProfiles/<callProfile-ID>**
3. **AAN/API/v1/co-ruimtes**
>Controleer in detail de gewenste ruimte-ID met behulp van **GET** op **/api/v1/co-ruimtes/<coSpace-ID>**
>Controleer of de gewenste callProfile-ID (stap 1) en de gastCallBeenProfile (stap 2) aan deze ruimte zijn gekoppeld
[indien dit niet het geval is, controleert u de minder specifieke elementen zoals huurder

(/api/v1/huurders/<huurder-ID>) of het algemene (/api/v1/systeem/profielen) niveau]

4. RAAK OP /API/v1/co-ruimtes/<coSpace-ID>/accessMethods

>Controleer in detail de gewenste toegangsmethode met GET op

/api/v1/coSpaces/<coSpace-ID>/accessMethods/<accessMethode-ID> om te controleren of de **hostBeenProfile** is toegewezen

In de CMS logging die in dit voorbeeld wordt getoond, heb je eerste twee gastdeelnemers die binnenkwamen (*Bel 38* vanaf 2000@steven.lab en *bel 39* vanaf 1060@steven.lab) die de `guestpin.space@acano.steven.lab` ruimte in en dan doet de host mee. Uit het fragment kunt u zien dat het voor gasten ons informeert over wat er met hem gedaan moet worden (**wordt gedeactiveerd**) en je kunt dit gedrag voor die gesprekken zien veranderen wanneer de host (`stejanss.movi@steven.lab`) zich in de ruimte begeeft (**wordt niet meer gedeactiveerd**). Op dezelfde manier kunt u dezelfde houtkap weer zien wanneer de gasten weer naar de lobby verhuizen zodra er geen gastheren meer zijn op de ruimte (**deze moeten worden gedeactiveerd**).

```
2017-02-21 17:48:54.809 Info call 38: incoming encrypted SIP call from
"sip:2000@steven.lab" to local URI "sip:guestpin.space@acano.steven.lab" 2017-02-21 17:48:54.822
Info call 38: setting up UDT RTP session for DTLS (combined media and control) 2017-02-21
17:48:54.837 Info call 38: compensating for far end not matching payload types 2017-02-21
17:48:54.847 Info sending prompt response (2) to BFCP message 2017-02-21 17:48:54.847 Info call
38: sending BFCP hello as client following receipt of hello when BFCP not active 2017-02-21
17:48:54.883 Warning call 38: replacing pending BFCP message "PrimitiveHelloAck" with
"PrimitiveHelloAck" 2017-02-21 17:48:54.883 Info call 38: BFCP (client role) now active 2017-02-
21 17:48:59.294 Info call 39: incoming encrypted SIP call from "sip:1060@steven.lab" to local
URI "sip:guestpin.space@acano.steven.lab" 2017-02-21 17:48:59.310 Info call 39: setting up UDT
RTP session for DTLS (combined media and control) 2017-02-21 17:48:59.323 Info call 39:
compensating for far end not matching payload types 2017-02-21 17:48:59.569 Info sending prompt
response (2) to BFCP message 2017-02-21 17:48:59.569 Info call 39: sending BFCP hello as client
following receipt of hello when BFCP not active 2017-02-21 17:48:59.746 Info call 39: BFCP
(client role) now active 2017-02-21 17:49:07.971 Info configuring call e2264fb0-483f-45bc-a4f3-
5a4ce326e72c to be deactivated
2017-02-21 17:49:07.972 Info participant "2000@steven.lab" joined space 22d9f4ca-8b88-
4d11-bba9-e2a2f7428c46 (Guest/Host PIN)
2017-02-21 17:49:12.463 Info configuring call b1b5d433-5ab5-49e1-9ae3-3f4f71703d1b to be
deactivated
2017-02-21 17:49:12.463 Info participant "1060@steven.lab" joined space 22d9f4ca-8b88-
4d11-bba9-e2a2f7428c46 (Guest/Host PIN)
2017-02-21 17:49:12.463 Info conference "Guest/Host PIN": unencrypted call legs now
present
2017-02-21 17:49:16.872 Info call 40: incoming encrypted SIP call from
"sip:stejanss.movi@steven.lab" to local URI "sip:guestpin.space@acano.steven.lab" 2017-02-21
17:49:16.885 Info call 40: setting up UDT RTP session for DTLS (combined media and control)
2017-02-21 17:49:24.260 Info call 40: audio prompt play time out 2017-02-21 17:49:26.670 Info
participant "stejanss.movi@steven.lab" joined space 22d9f4ca-8b88-4d11-bba9-e2a2f7428c46
(Guest/Host PIN)
2017-02-21 17:49:26.670 Info call e2264fb0-483f-45bc-a4f3-5a4ce326e72c ceasing to be
deactivated
2017-02-21 17:49:26.670 Info call b1b5d433-5ab5-49e1-9ae3-3f4f71703d1b ceasing to be
deactivated
2017-02-21 17:49:30.832 Info call 40: ending; remote SIP teardown - connected for 0:14
2017-02-21 17:49:30.833 Info participant "stejanss.movi@steven.lab" left space 22d9f4ca-
8b88-4d11-bba9-e2a2f7428c46 (Guest/Host PIN)
2017-02-21 17:49:30.833 Info configuring call e2264fb0-483f-45bc-a4f3-5a4ce326e72c to be
deactivated
2017-02-21 17:49:30.833 Info configuring call b1b5d433-5ab5-49e1-9ae3-3f4f71703d1b to be
deactivated
```

Gerelateerde informatie

- [Technische ondersteuning en documentatie – Cisco Systems](#)
- [CMS-documentatie](#)