

Configurar y solucionar problemas del acceso de invitado y de organizador en espacios CMS

Contenido

[Introducción](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Antecedentes](#)

[Configurar](#)

[1\) Configuración con diferentes URI](#)

[Verificación](#)

[2\) Configuración con los mismos URI pero con PIN/códigos de acceso de organizador y invitado no vacíos](#)

[Verificación](#)

[3\) Configuración con los mismos URI con combinación de PIN de invitado vacío y PIN de host no vacío](#)

[Verificación](#)

[4\) Un usuario organizador es miembro del espacio y autorizado a través de la conexión webRTC, los usuarios invitados se unen a la reunión con callID. El mismo URI y ID de llamada lo utilizan los participantes invitados y organizadores con PIN/contraseñas vacíos o no vacíos para los usuarios invitados](#)

[Verificación](#)

[Troubleshoot](#)

[Información Relacionada](#)

Introducción

En este documento se describe cómo configurar el acceso de invitado y de organizador en espacios de Cisco Meeting Server (CMS) mediante comandos de la API.

Prerequisites

Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

- Cisco Meeting Server (CMS) con espacios configurados y capaces de recibir llamadas
- Cliente API (como Poster, Postman) o
- [Guía de la API de CMS](#)

Componentes Utilizados

La información proporcionada en este documento se basa en la versión 2.1 de CMS:

La información en este documento se creó a partir de un dispositivo en un entorno de laboratorio específico. Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

Antecedentes

El documento describe los tipos de situaciones hipotéticas:

- Uso de diferentes URI o ID de llamada por parte de los participantes organizadores o invitados
- Uso del mismo URI por parte de los participantes organizadores o invitados donde la diferenciación se realiza en función de la entrada de PIN o código de acceso (ambos no vacíos)
- Uso del mismo URI por parte de los participantes organizadores o invitados donde la diferenciación se realiza en función de la entrada de PIN o código de acceso (combinación de un valor vacío y uno no vacío)
- Un usuario organizador es miembro del espacio y autorizado a través de la conexión webRTC, los usuarios invitados se unen a la reunión con callID. El mismo URI y ID de llamada lo utilizan los participantes invitados y organizadores con PIN/contraseñas vacíos o no vacíos para los usuarios invitados

Configurar

Hay cuatro posibilidades de diferenciación entre participantes de invitado y organizador en CMS, descritas en los siguientes cuatro ejemplos, y se basan principalmente en diferentes **callLegProfiles** que determinan el comportamiento de llamada entrante para aquellos participantes que se unen en el espacio.

Primero, se explica el método mediante un **URI** (o call-ID) diferente para los participantes invitados y los organizadores, y después se agrega usando diferentes códigos de acceso (o tiempo de espera) en el mismo **URI**, para hacer la diferenciación entre participantes invitados y organizadores. El tercer método de un tiempo de espera o una entrada PIN vacía para usuarios invitados se introdujo como una nueva función en CMS 2.1, como se muestra en la sección 2.4 de las [notas de la versión](#). El cuarto método explica cómo configurar el acceso de invitado y organizador en espacios con propietarios o miembros asignados y hacer que el miembro del espacio (propietario) sea el host del espacio.

1) Configuración con diferentes URI

Se trata de la configuración básica que ha estado disponible antes de la versión 2.1 CMS y es la mismo que para un call-ID diferente. Los siguientes pasos deben realizarse para obtener la diferenciación de acceso de invitado/organizador en el mismo espacio:

1. Crear un **callLegProfile** (**needsActivation = true**) de invitado.
2. Crear un **callLegProfile** (**needsActivation = false**) de organizador.
3. Asignar el **callLegProfile** de invitado a un espacio nuevo o existente (**método de acceso**

predeterminado).

4. Crear un nuevo `accessMethod` en ese mismo espacio con otro URI (e ID de llamada) y asignarle el `callLegProfile` de organizador.

Paso 1. Cree un `callLegProfile` de invitado (`necesitaActivation = true`).

Un `callLegProfile` determina el comportamiento de la llamada entrante y, de forma predeterminada, se asigna el comportamiento de la llamada entrante del invitado al espacio para que luego pueda tener un método de acceso diferente en ese mismo espacio, así como para que el host pueda unirse.

Nota: También puede asignar este comportamiento en el nivel de abonado (`/api/v1/tenants/<tenant-ID>`) o en el nivel de sistema (`/api/v1/system/profiles`) para, por ejemplo, aplicarlo a todos los espacios (o por abonado); sin embargo, aquí se muestra en el propio espacio. Tenga en cuenta que la asignación más específica del `callLegProfile` se tiene en cuenta para el comportamiento de la llamada entrante.

Aquí, el parámetro `needsActivation` es el más importante para el comportamiento de invitado/organizador, ya que si se establece en `true`, el participante no puede recibir ni aportar audio y video hasta que uno o varios participantes `full/activator` (organizadores) entren en el espacio. Otros parámetros en el `callLegProfile` se pueden encontrar en la sección 8.4.3 de la [Guía de API](#), bajo la cual los que se muestran pueden ser relevantes también en esta configuración (dependiendo de sus requisitos):

- `presentationContributionAllowed`
- `rxAudioMute`
- `rxVideoMute`
- `deactivationMode` (`deactivate` | `desconectar` | `keepActivated`) y `deactivationModeTime` [acción que se realizará cuando el último activador abandone la llamada]

Para crear el `callLegProfile` de invitado, haga una solicitud POST en `/api/v1/callLegProfiles` con los parámetros preferidos y el parámetro `needsActivation` establecido en `true` para que pueda realizar una solicitud GET en esa `callLegProfile-ID` después con este resultado, por ejemplo:

```
needsActivation>true
```

```
needsActivation>
```

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

Anote el `callLegProfile-ID` en negrita, ya que debe aplicarse en el espacio del paso 3 para el acceso de invitado (predeterminado).

Paso 2. Cree un `callLegProfile` de host (`necesitaActivation = false`).

De forma similar, cree el `callLegProfile` de organizador para el comportamiento de llamada entrante del organizador. Se aplican los mismos parámetros mencionados anteriormente, aunque los parámetros se pueden seleccionar según sus preferencias y requisitos. El paso principal aquí es establecer el parámetro `needsActivation` en `false` para darle la función de organizador.

Lo crea mediante una solicitud POST en /api/v1/callLegProfiles con los parámetros preferidos y con el parámetro needsActivation establecido en **false** para que pueda realizar una solicitud GET en esa callLegProfile-ID posteriormente con este resultado, por ejemplo:

```
needsActivation>
```

```
    false
```

```
needsActivation>
```

Anote el **callLegProfile-ID** en negrita, ya que esto se debe aplicar en el espacio **accessMethod** en el paso 4 para el acceso del host.

Paso 3. Asigne el **callLegProfile** de invitado a un espacio nuevo o existente (siendo el **accessMethod** predeterminado).

Ejecute un comando PUT en un espacio existente (/api/v1/coSpaces/<coSpace-ID>) para adaptar el espacio o un comando POS en /api/v1/coSpaces para crear uno nuevo con el parámetro **callLegProfile** de invitado creado en el paso 1 como el comportamiento de llamadas entrantes para ese espacio. También puede configurar los parámetros URI, [passcode](#) y call-ID para ese espacio de acuerdo con sus preferencias, como se describe en la sección 6.2 de la [Guía de la API](#).

Lleve a cabo una solicitud GET en ese espacio (/api/v1/coSpaces/<coSpace-ID>) para comprobar que el **callLegProfile** de invitado esté asociado con él, así como los valores de URI y call-ID. Un ejemplo de resultado con este ejemplo creado como invitado **callLegProfile** en el paso 1 es este con un valor URI de **guest.space** y call-ID de 628821815 (sin código de acceso establecido):

```
uriCallIdcallLegProfile>
```

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

```
callLegProfile>
```

Anote el space-ID en negrita, ya que se debe utilizar para crear el **accessMethod** en ese espacio concreto en el paso 4.

Paso 4. Cree un nuevo **accessMethod** en ese espacio con un URI diferente (y call-ID) y asígnele el **callLegProfile** del host.

Desea crear una forma diferente de acceder al espacio que el acceso de invitado que es actualmente el predeterminado. Esto se realiza especificando un accessMethod en el propio espacio mediante un comando POST en /api/v1/coSpaces/<coSpace-ID>/accessMethods con el

coSpace-ID como valor marcado en negrita en el paso 3 (7cc797c9-c0a8-47cf-b51998dc5a01f1ade) en el que se aplica el callLegProfile de organizador del paso 2, así como los diferentes campos URI y call-ID.

Después de una solicitud GET en ese espacio accessMethod (/api/v1/coSpaces/<coSpace-ID>/accessMethods/<accessMethod-ID>), debe poder ver un tipo de resultado similar al siguiente, en el que puede ver un **URI** (host.space) y **call-ID** (888) distinto al default accessMethod del espacio así como el host especialmente asociado **callLegProfile** como se configuró en el paso 2:

```
uricallIdpasscodecallLegProfile>
```

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

```
callLegProfile>
```

Verificación

Ahora puede marcar para entrar en la misma reunión:

- Como invitado

- marcando el URI guest.space (seguido por el dominio tal como se configuró en las reglas de coincidencia de llamadas)

- introduciendo el valor de call-ID 628821815 a través de la entrada en IVR o WebRTC (sin código de acceso)

- Como organizador

- marcando el URI host.space (seguido por el dominio tal como se configuró en las reglas de coincidencia de llamadas)

- introduciendo el valor de call-ID 888 a través de la entrada en IVR o WebRTC (sin código de acceso)

Cuando solo hay invitados que entraron en el espacio, todos se colocan en una sala de espera de recepción hasta que entre el organizador. Una vez que un organizador se une, todos los invitados y los organizadores se ponen en conferencia. Si ya no hay hosts unidos en el espacio pero todavía hay algunos invitados, vuelven a la pantalla del vestíbulo según la configuración de **desactivar** en el parámetro **deactivationMode** del invitado **callLegProfile**, como se muestra en el Paso 1.

2) Configuración con los mismos URI pero con PIN/códigos de acceso de organizador y invitado no vacíos

Esta configuración es similar a la del ejemplo anterior y también está disponible antes de la versión CMS 2.1. Requiere que tanto los invitados como los organizadores introduzcan un PIN o código de acceso no vacío para que se pueda realizar una diferenciación posteriormente, ya que

marcan el mismo **URI**.

Los pasos de configuración son bastante similares al ejemplo de configuración anterior:

1. Crear un **callLegProfile** (**needsActivation = true**) de invitado.
2. Crear un **callLegProfile** (**needsActivation = false**) de organizador.
3. Asignar el **callLegProfile** de invitado a un espacio nuevo o existente y especificar un **código de acceso (PIN)** de invitado (método de acceso predeterminado).
4. Crear un nuevo **accessMethod** en ese mismo espacio con el mismo **URI (call-ID)** diferente) y asignarle el **callLegProfile** de organizador, incluido un **código de acceso (PIN)** de organizador.

Paso 1. Cree un **callLegProfile** de invitado (**necesitaActivation = true**).

Se puede utilizar la misma configuración que en el ejemplo anterior 1 e incluso el mismo **callLegProfile** (**d4bfe12d-68cd-41c0-a671-48395ee170ab**) como se ha demostrado.

Paso 2. Crear un **callLegProfile** (**needsActivation = false**) de organizador.

Se puede utilizar la misma configuración que en el ejemplo anterior 1 e incluso el mismo **callLegProfile** de host (**7306d2c1-bc15-4dbf-ab4a-1cbdaabd1912**) como se ha demostrado.

Paso 3. Asigne el **callLegProfile** de invitado a un espacio nuevo o existente que especifique un código de acceso de invitado (PIN) (que es el **accessMethod** predeterminado).

Del mismo modo que antes, puede realizar una operación PUT en un espacio existente (**/api/v1/coSpaces/<cospace-ID>**) o una operación POST para crear un nuevo espacio (**/api/v1/coSpaces**) con los parámetros deseados para el URI, el código de acceso y el call-ID, por ejemplo (desde el paso 1) que le asignó según la sección 6.2 de la guía de API.

Si realiza una solicitud GET en ese espacio, debe poder ver un resultado similar al que se muestra en este caso, en el que se ve un **URI** de **guestpin.space**, un **call-ID** de 189, nuestro **callLegProfile** invitado creado anteriormente y un **código de pasaje** de 789:

```
uricallIdcallLegProfile>
```

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

```
callLegProfile><
```

```
passcode>789
```

```
passcode>
```

Anote el space-ID en negrita, ya que se debe utilizar para crear el **accessMethod** en ese espacio concreto en el paso 4.

Paso 4. Cree un nuevo **accessMethod** en ese espacio con el mismo **URI** (diferentes **call-ID**) y asígnele el **callLegProfile** del host, incluido un código de acceso del host (PIN).

En este espacio también crea un método de acceso diferente para los organizadores (ya que el **callLegProfile** de invitado está asignado en el espacio como la opción de entrada predeterminada), tal como el primer ejemplo de configuración. Para esto, se utiliza el comando **POST** en **/api/v1/coSpaces/<coSpace-ID>/accessMethods** con un **coSpace-ID** para nuestro espacio de **22d9f4ca-8b88-4d11-bba9-e2a2f7428c46** , como se resaltó en el paso anterior. En este comando **POST**, puede proporcionar los diferentes parámetros como el **URI** (guestpin.space, el mismo que el original), **call-ID** (889), **host callLegProfile** según se define en el paso 2 y el **PIN** del host (1234 en este caso).

Si realiza una solicitud **GET** en ese **accessMethod**, debe poder ver un resultado similar que muestre el mismo **URI** de guestpin.space, un **call-ID** de 889, la referencia **callLegProfile** del host y el **PIN** del host de 1234:

```
uricallId>passcode>1234
```

```
passcode><
```

```
callLegProfile>
```

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

```
callLegProfile>
```

Verificación

Ahora puede marcar para entrar en la misma reunión:

- Como invitado

- marcando el **URI** de guestpin.space (seguido del dominio configurado en las reglas de coincidencia de llamadas) e introduciendo el **PIN** 789

- introduciendo el valor de **call-ID 189** a través de la entrada en **IVR** o **WebRTC** con el **PIN 789**

- Como organizador

- marcando el **URI** de guestpin.space (seguido del dominio configurado en las reglas de coincidencia de llamadas) e introduciendo el **PIN** 1234

- introduciendo el valor de **call-ID 889** a través de la entrada en **IVR** o **WebRTC** con el **PIN 1234**

Cuando solo hay invitados que entraron en el espacio, todos se colocan en una sala de espera de recepción hasta que entre el organizador. Una vez que un organizador se une, todos los invitados y los organizadores se ponen en conferencia. Si ya no hay hosts unidos en el espacio pero todavía hay algunos invitados, vuelven a la pantalla del vestíbulo según la configuración de **desactivar** en el parámetro **deactivationMode** del invitado **callLegProfile** , como se muestra en el Paso 1.

3) Configuración con los mismos URI con combinación de PIN de invitado vacío y PIN de host no vacío

Esta configuración solo está disponible a partir de la versión 2.1 de CMS en adelante debido a algunos comandos de la API recientemente agregados de **passcodeMode** y **passcodeTimeout** en la **sección callProfile**. Esto permite un PIN vacío para invitados para unirse a (introducir # o tiempo de espera), mientras que el organizador tiene un PIN para acceder al espacio y activarlo. El parámetro **callProfile** **controla la experiencia de la llamada entrante para las llamadas de SIP (incluido Lync) y, por lo tanto, no se puede aplicar para los clientes de CMA (cliente pesado y WebRTC).**

Los pasos de configuración son similares a los del ejemplo 2, con la adición de **callProfile**:

1. Crear un **callLegProfile** (**needsActivation = true**) de invitado.
2. Crear un **callLegProfile** (**needsActivation = false**) de organizador.
3. Crear un **callProfile** con la configuración deseada de **passcodeMode** y **passcodeTimeout**.
4. Asignar el **callLegProfile** y el **callProfile** de invitado del paso 3 a un espacio nuevo o existente, y especificar un código de acceso (PIN) de invitado (método de acceso predeterminado).
5. Crear un nuevo **accessMethod** en ese mismo espacio con el mismo URI (call-ID diferente) y asignarle el **callLegProfile** de organizador, incluido un código de acceso (PIN) de organizador.

Como las configuraciones son bastante idénticas a los ejemplos de configuración 1 y 2, hay referencias a esas. De hecho, para la prueba, se utilizó el mismo espacio que en el ejemplo 2, pero ahora se agregó con **callProfile**.

Paso 1. Cree un **callLegProfile** de invitado (**necesitaActivation = true**).

Se puede utilizar la misma configuración que en el ejemplo anterior 1 e incluso el mismo **callLegProfile** de invitado (**d4bfe12d-68cd-41c0-a671-48395ee170ab**) como se ha demostrado.

Paso 2. Cree un **callLegProfile** de host (**necesitaActivation = false**).

Se puede utilizar la misma configuración que en el ejemplo anterior 1 e incluso el mismo **callLegProfile** de host (**7306d2c1-bc15-4dbf-ab4a-1cbdaabd1912**) como se ha demostrado.

Paso 3. Cree un **callProfile** con la configuración **passcodeMode** y **passcodeTimeout** deseada.

Puede crear un **callProfile** que determine la experiencia de llamada entrante para las llamadas de SIP (incluido Lync). Se pueden realizar algunas configuraciones aquí, como permitir la grabación, permitir la transmisión o definir el límite máximo de participantes, pero queremos centrarnos en las novedades de la API de CMS 2.1 relacionadas con la gestión del código de acceso. Los demás parámetros pueden encontrarse en la sección 8.2 de la [Guía de la API](#).

Dos parámetros determinan el comportamiento de código de acceso:

- ***passcodeMode***

- **required (obligatorio)**: la IVR espera para siempre que un usuario introduzca el PIN o el # para un PIN vacío (para invitados)

- **timeout (tiempo de espera)**: la IVR espera una cantidad de segundos ***passcodeTimeout*** para que el participante introduzca el PIN y, si no se ha introducido ninguna entrada en ese tiempo, supone que se ha introducido un PIN en blanco (#)

- ***passcodeTimeout*** : solo debe establecerse cuando ***passcodeMode*** está establecido en tiempo de espera y controla la cantidad de tiempo que debe transcurrir antes de que el código de acceso se interprete como valor en blanco

Para crear el **callProfile**, ejecute un comando **POST** en ***/api/v1/callProfiles*** (o **PUT** en ***/api/v1/callProfiles/<callProfile-ID>*** si desea modificar uno existente) con los parámetros deseados para ***passcodeMode*** y ***passcodeTimeout***. Si ejecuta un comando **GET** en ese **callProfile** específico, recibirá un resultado con el modo configurado en **timeout** (tiempo de espera) y un valor de tiempo de espera de 5 segundos:

```
passcodeMode>timeout
```

```
passcodeMode><
```

```
passcodeTimeout>5
```

```
passcodeTimeout>
```

Anote el **callProfile-ID** en negrita, ya que se debe utilizar para asignar al espacio este comportamiento de llamada entrante en el paso 4.

Paso 4. Asigne el **callLegProfile** y el **callProfile** de invitado del paso 3 a un espacio nuevo o existente y especifique un código de acceso de invitado (PIN) (que es el método de acceso predeterminado).

De un modo similar al anterior, puede realizar una operación **PUT** en un espacio existente (***/api/v1/coSpaces/<cospace-ID>***) o una operación **POST** para crear un nuevo espacio (***/api/v1/coSpaces***) con los parámetros que desee para **URI** y **call-ID** por ejemplo y el **callLegProfile** de invitado (del paso 1). La diferencia con los ejemplos anteriores es el **callProfile** del paso 3 y el hecho de que no se le asigna ningún código de acceso.

Si realiza una solicitud **GET** en ese espacio, debe poder ver un resultado similar al de este ejemplo, donde verá el **URI** de **guestpin.space**, un **call-ID** de 189, el invitado previamente creado **callLegProfile** y el **callProfile** configurado en el paso 3:

```
uricallIdcallLegProfile>
```

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

```
callLegProfile><
```

```
callProfile>
```

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

```
callProfile>
```

Anote el ID de espacio en negrita, ya que se debe utilizar para crear `accessMethod` en ese espacio concreto en el paso 5.

Paso 5. Cree un nuevo `accessMethod` en ese mismo espacio con el mismo `URI` (diferentes `call-ID`) y asígnele el `callLegProfile` del host, incluido un código de acceso del host (PIN).

En este espacio también crea un método de acceso diferente para los organizadores (ya que el `callLegProfile` de invitado está asignado en el espacio como la opción de entrada predeterminada), tal como el primer ejemplo de configuración. Para esto, se utiliza el comando `POST` en `/api/v1/coSpaces/<coSpace-ID>/accessMethods` donde el valor `coSpace-ID` es reemplazado con el valor de su espacio `22d9f4ca-8b88-4d11-bba9-e2a2f7428c46`, como se resaltó en el paso para este caso. En este comando `POST`, puede proporcionar los diferentes parámetros como el `URI` (`guestpin.space`, el mismo que el original), `call-ID` (889), `callLegProfile` de organizador, como se define en el paso 2, y el código de acceso o PIN (1234 en este caso).

Si realiza una solicitud `GET` en ese `accessMethod`, debe poder ver un resultado similar que muestre el mismo `URI` de `guestpin.space`, un `call-ID` de 889, la referencia `callLegProfile` del host y el PIN del host de 1234:

```
uricallIdpasscode>1234
```

```
passcode><
```

```
callLegProfile>
```

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

```
callLegProfile>
```

Verificación

Ahora puede marcar para entrar en la misma reunión:

- Como invitado
 - marcando el **URI** de `guestpin.space` (seguido por el dominio configurado en las reglas de coincidencia de llamadas) e introduciendo **#** como **PIN** o dejándolo expirar 5 segundos
 - introduciendo el valor de **call-ID 189 a través de la entrada en IVR o WebRTC**
- Como organizador
 - marcando el identificador URI de `guestpin.space` (seguido del dominio configurado en las reglas de coincidencia de llamadas) e introduciendo el **PIN 1234**
 - **introduciendo el valor de call-ID 889 a través de la entrada en IVR o WebRTC con el PIN 1234**

4) Un usuario organizador es miembro del espacio y autorizado a través de la conexión webRTC, los usuarios invitados se unen a la reunión con callID. El mismo URI y ID de llamada lo utilizan los participantes invitados y organizadores con PIN/contraseñas vacíos o no vacíos para los usuarios invitados

Los siguientes pasos deben realizarse para obtener la diferenciación de acceso de invitado/organizador en el mismo espacio para los miembros y los no miembros del espacio:

1. Crear un `callLegProfile` (**`needsActivation = true`**) de invitado.
2. Crear un `callLegProfile` (**`needsActivation = false`**) de organizador.
3. Asignar el `callLegProfile` de invitado a un espacio nuevo o existente (método de acceso predeterminado).
4. Cree un nuevo **`accessMethod`** en ese mismo espacio con el mismo **URI** (y **call-ID**) y asígnele el **`callLegProfile`** del host
5. Asigne el propietarioJID del usuario al mismo espacio. (si no está asignado)
6. Agregue `thatownerID` como usuario miembro al mismo espacio y asigne `hostcallLegProfile` a ese usuario miembro

Paso 1. Cree un `callLegProfile` de invitado (`needsActivation = true`).

En este ejemplo se utiliza la misma configuración que en el ejemplo anterior 1 y el `callLegProfile` de invitado (**`bfe7d07f-c7cb-4e90-a46e-4811bbaf6978`**).

Anote el **`callLegProfile-ID`** en negrita, ya que debe aplicarse en el espacio del paso 3 para el acceso de invitado.

Paso 2. Crear un `callLegProfile` (`needsActivation = false`) de organizador.

En este ejemplo se utiliza la misma configuración que en el ejemplo 1 anterior y el `callLegProfile` de host (**`0e76e943-6d90-43df-9f23-7f1985a74639`**).

Anote el **`callLegProfile-ID`** en negrita, ya que debe aplicarse en el espacio **`accessMethod`** en el paso 4 para el acceso del host y en el miembro de `coSpace` en el paso 6.

Paso 3. Asigne el **callLegProfile** de invitado a un espacio nuevo o existente (siendo **accessMethod** predeterminado).

Realice un comando PUT en un espacio existente (**/api/v1/coSpaces/<coSpace-ID>**) para adaptar el espacio o un comando POST en **/api/v1/coSpaces** para crear uno nuevo con el parámetro **callLegProfile** de invitado creado en el paso 1 como comportamiento de llamada entrante para ese espacio. También puede establecer los parámetros **URI** y **call-ID** para ese espacio, así como su deseo según la sección 6.2 de la [Guía de API](#).

Realice una solicitud GET en ese espacio (**/api/v1/coSpaces/<coSpace-ID>**) para verificar que el **callLegProfile** invitado esté asociado a él, así como el valor **URI** y **call-ID**. Un ejemplo de resultado con este ejemplo de **callLegProfile** de invitado creado en el paso 1 es este con un valor **URI global** y **call-ID** de 1234 (sin código de acceso establecido), **nonMemberAccess** to true:

```
<?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>
```

Anote el **space-ID** en negrita, ya que se debe utilizar para crear el **accessMethod** en ese espacio concreto en el paso 4.

Paso 4. Cree un nuevo **accessMethod** en ese espacio con el mismo **URI** (y **call-ID**) y asígnele el **callLegProfile** del host.

Desea crear una forma diferente de acceder al espacio que el acceso de invitado que es actualmente el predeterminado. Esto se hace especificando un **accessMethod** en el propio espacio mediante un comando POST **command** en **/api/v1/coSpaces/<coSpace-ID>/accessMethods**, con **coSpace-ID** como valor marcado en negrita en el paso 3 (**96d28acb-86c6-478d-b81a-a37ffb0adafc**) en el que se aplica el **callLegProfile** del host del paso 2, así como el mismo campo **URI** y **call-ID**. Puede agregar un código de acceso no vacío para los hosts que se conectan mediante **callID** (sin haber iniciado sesión como usuario a través de **webRTC**).

Después de una solicitud GET en ese espacio **accessMethod** (**/api/v1/coSpaces/<coSpace-ID>/accessMethods/<accessMethod-ID>**), debe poder ver un tipo de resultado similar al que aparece aquí, donde puede ver el mismo **URI** (**global**) y el **call-ID** (1234 asociado especialmente) **host callLegProfile** tal como se configuró en el paso 2 y el código de acceso del host(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>kff0lzTTE0feL4fsdf43w_B </secret>
</accessMethod>
```

Paso 5. Asigne el propietario del usuario **Jid**to al espacio. (si no está asignado). Agregue **ownerJID** al espacio especificando **ownerJid** (user1@evacanoalone.net) en el espacio mediante

unPUTcommand on </api/v1/coSpaces/<coSpace-ID>>

Después de una solicitud GET en ese espacio, debe poder ver que el propietarioId y el propietarioJid han sido asignados al espacio:

```
<?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>0w402zTTF0WdL4ymF8D0_A</secret>
  <numAccessMethods>1</numAccessMethods>
  <defaultLayout>allEqual</defaultLayout>
</coSpace>
```

Anote el ownerId (1d942281-413e-4a2a-b776-91a674c3a5a9).

Paso 6. Agregue ese ownerId (1d942281-413e-4a2a-b776-91a674c3a5a9) del paso 5 como usuario miembro al espacio y asigne hostcallLegProfile a ese usuario miembro. Esto se realiza especificando userJid and host callLegProfile en el propio espacio (especificando coSpaceID) mediante un comando POST (</api/v1/coSpaces/<coSpaceID>/coSpaceUsers>). Otros parámetros en el coSpaceLos usuarios pueden ser se encuentra en la sección 6.4.2 de la guía de la API, en la que las guías mostradas también pueden ser relevantes en esta configuración:

```
<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>
```

Verifique que el usuario miembro se haya agregado al espacio mediante un comando GET (</api/v1/coSpaces/<coSpaceID>/coSpaceUsers?>)

```
<?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>
```

Anote el ID de usuario (si es diferente ID de propietario del formulario, paso 5). Verifique que una solicitud GET haya asignado el callLegProfile a coSpaceUser especificando coSpaceIDanduserID (/api/v1/coSpaces/<coSpaceID>/coSpaceUsers/<userID>)

```
<?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>
```

Verificación

Ahora puede marcar para entrar en la misma reunión:

- Como invitado

- marcando el URI (seguido del dominio configurado en las reglas de coincidencia de llamadas)

- introduciendo el valor de call-ID 1234 a través de la entrada en IVR o WebRTC (sin código de acceso)

- Como organizador

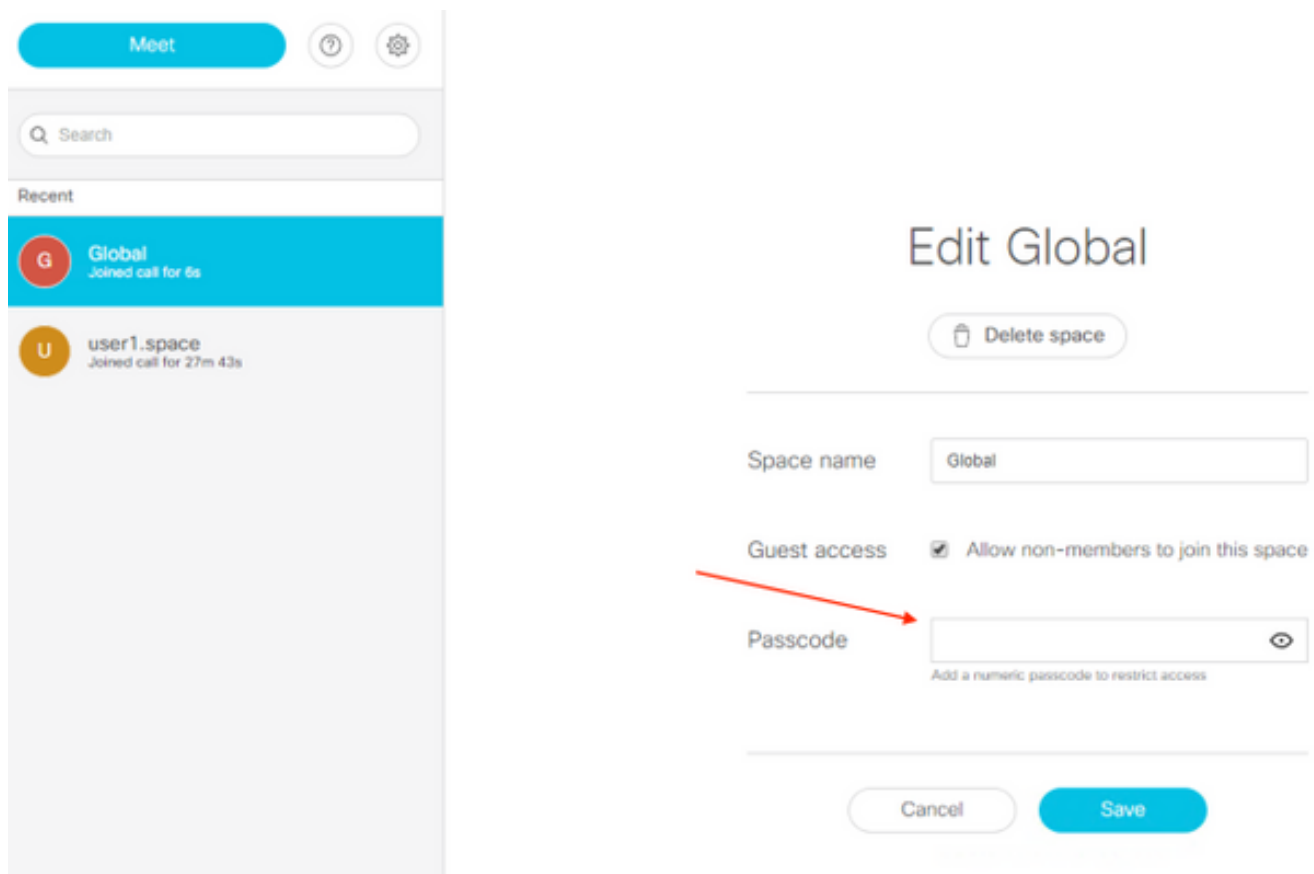
Al iniciar sesión como usuario (un miembro del espacio con callLegProfile de "host" asignado, con user1@evacanoalone.net en este escenario) a través de webRTC y unirse al espacio (URI "global").

- marcando el URI "global" (seguido del dominio configurado en las reglas de coincidencia de llamadas) y el código de acceso 12345.

- introduciendo el valor de call-ID 1234 a través de la entrada en IVR o WebRTC (con un código de acceso de host 12345)

Cuando solo hay invitados que entraron en el espacio, todos se colocan en una sala de espera de recepción hasta que entre el organizador. Una vez que un organizador se une, todos los invitados y los organizadores se ponen en la conferencia. Si ya no hay hosts unidos en el espacio pero todavía hay algunos invitados, vuelven a la pantalla del vestíbulo según la configuración de **desactivar** en el parámetro **deactivationMode** del invitado **callLegProfile** , como se muestra en el Paso 1.

El organizador (propietario/miembro) puede establecer(editar/quitar) una contraseña para los invitados directamente en la aplicación webRTC o deshabilitar completamente el acceso de un no miembro (invitado) para el espacio:



Troubleshoot

En esta sección se brinda información que puede utilizar para resolver problemas en su configuración.

El registro en CMS nos muestra brevemente cuando entra como invitado o cuando entra el primer organizador, pero es recomendable realizar la verificación con solicitudes GET del callProfile, así como de las definiciones de callLegProfile de organizador y de invitado, y la asignación de ellos en sus respectivos accessMethods (o el método de acceso de predeterminado) o en un nivel superior (nivel global o nivel de abonado).

Puede seguir esta estructura para obtener toda la información:

1. **GET en /api/v1/callProfiles** (si se utiliza con **passcodeMode**)
>verifique en detalle el callProfile-ID deseado usando **GET on /api/v1/callProfiles/<callProfile-**

ID>

2. GET en /api/v1/callLegProfiles

>verifique en detalle los callLegProfile-ID deseados de invitado y organizador usando GET en /api/v1/callProfiles/<callProfile-ID>

3. GET en /api/v1/coSpaces

>verifique en detalle el ID de espacio deseado usando GET on /api/v1/coSpaces/<coSpace-ID>

>compruebe si el callProfile-ID (paso 1) y el callLegProfile de invitado (paso 2) están asociados a este espacio

(Si no encuentra estos valores, realice la verificación elementos menos específicos, como nivel de abonado [/api/v1/tenants/<tenant-ID> o nivel global [/api/v1/system/profiles])

4. GET en /api/v1/coSpaces/<coSpace-ID>/accessMethods

>verifique en detalle el accessMethod deseado mediante GET on /api/v1/coSpaces/<coSpace-ID>/accessMethods/<accessMethod-ID> para comprobar si se ha asignado el callLegProfile de organizador

En el registro de CMS que se muestra en este ejemplo, primero hay dos participantes invitados que entran (*llame al 38* desde 2000@steven.lab y *llame al 39* desde 1060@steven.lab) que pasan el tiempo de espera al espacio guestpin.space@acano.steven.lab y luego el host se une. En el fragmento se puede observar que en el caso de invitados se informa lo que debe realizarse con la llamada (**to be deactivated** [desactivar]) y que este comportamiento cambia cuando el organizador (stejanss.movi@steven.lab) entra en el espacio (**ceasing to be deactivated** [activar]). De un modo similar, puede ver el mismo registro cuando los invitados pasan nuevamente a la recepción tan pronto como deje de haber organizadores en el espacio (**to be deactivated** [desactivar]).

```
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

Información Relacionada

- [Soporte Técnico y Documentación - Cisco Systems](#)
- [documentación de CMS](#)