Cisco Meeting Server

Cisco Meeting Server Release 2.4.8

Release Notes

September 04, 2019

Contents

What's changed

Version	Change
2.4.8	Added section "Resolved in 2.4.8" (Sept 4, 2019) Hashes updated.
2.4.7	Added section "Resolved in 2.4.7". Hashes updated.
	Issues resolved in 2.4.0 updated. (May 23, 2019)
2.4.6	Additional browser support for WebRTC app introduced (April 25, 2019) Added section "Resolved in 2.4.6". Hashes updated.
2.4.5	Safari on iOS for iPhones now fully supported (April 4, 2019) Added section "Resolved in 2.4.5" Hashes updated.
2.4.4	Support for Google Chrome 73 introduced. (March 7, 2019) Added section "Resolved in 2.4.4". Hashes updated.
2.4.3	Note added on speaker source selection in browsers. (February 18, 2019)
2.4.3	Added: WebRTC App support using Safari on iOS (Documentation omission) (January 28, 2019) Recording indicator for dual homed confererences updated. Open issues updated.
2.4.3	Support for Google Chrome 72 introduced. (January 24, 2019) New Media Module status field introduced in the Web Admin. Added section "Resolved in 2.4.3". Hashes updated.
2.4.2	Added section "New features introduced in version 2.4.2" Added section "Resolved in 2.4.2". Hashes updated.
2.4.1	Added section "Resolved in 2.4.1". Hashes updated.
2.4.0	New release of Cisco Meeting Server software. (September 19, 2018)

1 Introduction

These release notes describe the new features, improvements and changes in release 2.4 of the Cisco Meeting Server software.

The Cisco Meeting Server software can be hosted on:

- the Cisco Meeting Server 2000, a UCS 5108 chassis with 8 B200 blades and the Meeting Server software pre-installed as the sole application.
- the Cisco Meeting Server 1000, a Cisco UCS server preconfigured with VMware and the Cisco Meeting Server installed as a VM deployment.
- the Acano X-Series hardware.
- or on a specification-based VM server. Note: From version 2.4 Meeting Server software no longer supports Microsoft Hyper-V.

Throughout the remainder of these release notes, the Cisco Meeting Server software is referred to as the Meeting Server.

If you are upgrading from a previous version, you are advised to take a configuration backup using the **`backup snapshot <filename>`** command, and save the backup safely on a different device. See the MMP Command Reference document for full details.

Note about certificate validation: From version 2.4, the Web Bridge correctly validates the XMPP Server's TLS certificate. If WebRTC app users have difficulty logging in after you upgrade the Meeting Server, then check that the uploaded XMPP certificate follows the advice in the Certificate Guidelines. Specifically, that the SAN field holds the domain name of the XMPP server. Prior to version 2.4 there were issues in XMPP certificate validation.

Note about Microsoft RTVideo: support for Microsoft RTVideo and consequently Lync 2010 on Windows and Lync 2011 on Mac OS, will be removed in a future version of the Meeting Server software.

Note about incoming calls: By default incoming calls are not allowed. To allow incoming calls to Cisco Meeting App users, set parameter **`canReceiveCalls=true`** for API object **`/user/profiles/<user profile id>`**.

Note about chat message board: For existing deployments that use chat message boards, chat will remain enabled when you upgrade to 2.4. Otherwise, you will need to use the API to create a callProfile with parameter **`messageBoardEnabled`** set to true.

1.1 Cisco Meeting Server platform maintenance

It is important that the platform that the Cisco Meeting Server software runs on is maintained and patched with the latest updates.

1.1.1 Cisco Meeting Server 1000 and other virtualized platforms

The Cisco Meeting Server software runs as a virtualized deployment on the following platforms:

- Cisco Meeting Server 1000
- Cisco Multiparty Media 400v, 410v and 410vb
- specification-based VM platforms.

Note: From version 2.4, Cisco Meeting Server software no longer supports Microsoft Hyper-V virtualized deployments.

CAUTION: Irrespective of which virtualized platform is running the Cisco Meeting Server software, ensure the platform is up to date with the latest patches. Failure to maintain the platform may compromise the security of your Cisco Meeting Server.

1.1.2 Cisco Meeting Server 2000

The Cisco Meeting Server 2000 is based on Cisco UCS technology running Cisco Meeting Server software as a physical deployment, not as a virtualized deployment.

CAUTION: Ensure the platform (UCS chassis and modules managed by UCS Manager) is up to date with the latest patches, follow the instructions in the Cisco UCS Manager Firmware Management Guide. Failure to maintain the platform may compromise the security of your Cisco Meeting Server.

1.2 Using the Cisco Expressway-E as the edge device in Meeting Server deployments

Over the previous few releases of Cisco Expressway software, edge features have been developed to enable the Cisco Expressway-E to be used as the edge device in Meeting Server deployments. Use the TURN server capabilities in Cisco Expressway-E to connect:

- participants using the WebRTC app to conferences hosted on the Meeting Server,
- remote Lync and Skype for Business clients to conferences hosted on the Meeting Server.

In addition, the Cisco Expressway-E can be used as a SIP Registrar to register SIP endpoints or to proxy registrations to the internal call control platform (Cisco Unified Communications Manager or Cisco Expressway-C).

Table 1 below indicates the configuration documentation that covers setting up Cisco Expressway-E to perform these functions. Table 2 below shows the introduction of the features by release.

Note: Cisco Expressway-E can not be used to connect remote Cisco Meeting App thick clients (Windows/Mac desktop or iOS) to conferences hosted on the Meeting Server. Nor can the Cisco Expressway-E be used between on premises Microsoft infrastructure and the Meeting Server. In deployments with on-premises Microsoft infrastructure and the Meeting Server, the Meeting Server must use the Microsoft Edge server to traverse Microsoft calls into and out of the organization.

Note: If you are configuring dual homed conferencing between on-premises Meeting Server and on-premises Microsoft Skype for Business infrastructure, then the Meeting Server automatically uses the TURN services of the Skype for Business Edge.

Table 1: Documentation covering Cisco Expressway as the edge device for the Meeting Server

Edge feature	Configuration covered in this guide
Connect remote WebRTC apps	Cisco Expressway Web Proxy for Cisco Meeting Server Deployment Guide
Connect remote Lync and Skype for Business clients	Cisco Meeting Server with Cisco Expressway Deployment Guide
SIP Registrar or to proxy registrations to the internal call control platform	Cisco Expressway-E and Expressway-C Basic Configuration (X8.11)

Table 2: Expressway edge support for the Meeting Server

Cisco Expressway-E version	Edge feature	Meeting Server version
X8.11	Supported: - load balancing of clustered Meeting Servers, - Microsoft clients on Lync or Skype for Business infrastructure in other organizations, or Skype for Business clients on Office 365 (not "consumer" versions of Skype). - interoperability between on-premise Microsoft infrastructure and on-premise Meeting Server, **where no Microsoft calls traverse into or out of the organization.** - standards based SIP endpoints. - standards based H.323 endpoints. - Cisco Meeting App thin client (Web RTC app) using TCP port 443. Not supported: - off premise Cisco Meeting App thick clients (Windows/Mac desktop or iOS). - interoperability between on-premise Microsoft infrastructure and on-premise Meeting Server **where Microsoft calls traverse into or out of the organization**, in this scenario, the Meeting Server must use the Microsoft Edge server to traverse Microsoft calls into and out of the organization. See Cisco Meeting Server with Cisco Expressway Deployment Guide (2.4/X8.11.4).	2.4
X8.10	Supported: - Microsoft clients on Lync or Skype for Business infrastructure in other organizations, or Skype for Business clients on Office 365 (not "consumer" versions of Skype), - standards based SIP endpoints, - Cisco Meeting App thin client (Web RTC app) using UDP port 3478 to connect to the Meeting Server via the Expressway reverse web proxy. Not supported: - load balancing of clustered Meeting Servers, - off premise Cisco Meeting App thick clients (Windows/Mac desktop or iOS) or Cisco Meeting App thin client (Web RTC app) using TCP port 443, - interoperability between on premises Microsoft infrastructure and Meeting Server; in this scenario, the Meeting Server must use the Microsoft Edge server to traverse Microsoft calls into and out of the organization. See Cisco Expressway Web Proxy for Cisco Meeting Server	2.3

Cisco Expressway-E version	Edge feature	Meeting Server version
X8.9	Supported: - Microsoft clients on Lync or Skype for Business infrastructure in other organizations, or Skype for Business clients on Office 365 (not " consumer" versions of Skype), - standards based SIP endpoints. Not supported: - load balancing of clustered Meeting Servers,, - off-premise Cisco Meeting App thick clients (Windows/Mac desktop or iOS) and Cisco Meeting App thin client (WebRTC app), - interoperability between on premises Microsoft infrastructure and Meeting Server; in this scenario, the Meeting Server must use the Microsoft Edge server to traverse Microsoft calls into and out of the organization See Cisco Expressway Options with Meeting Server and/or Microsoft Infrastructure	2.2

From version 2.4, you should start migrating your Meeting Server deployments from using the Meeting Server SIP edge component (SIP and Lync Call Traversal feature) and the Meeting Server TURN server, to using the Expressway X8.11 TURN server.

Figure 1 and Figure 2 illustrate recommended Meeting Server deployments. The deployments are discussed in the Cisco Meeting Server deployment guides version 2.4 and later.

Figure 1: Cisco Unified Communications Manager-centric deployment example

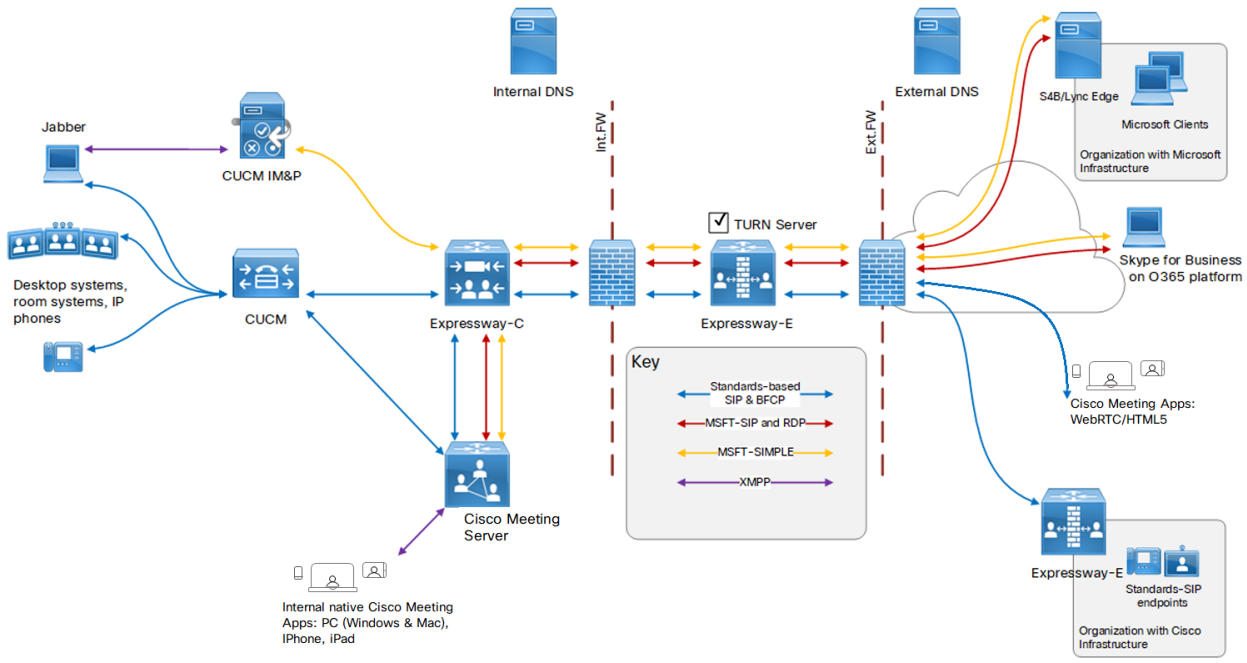
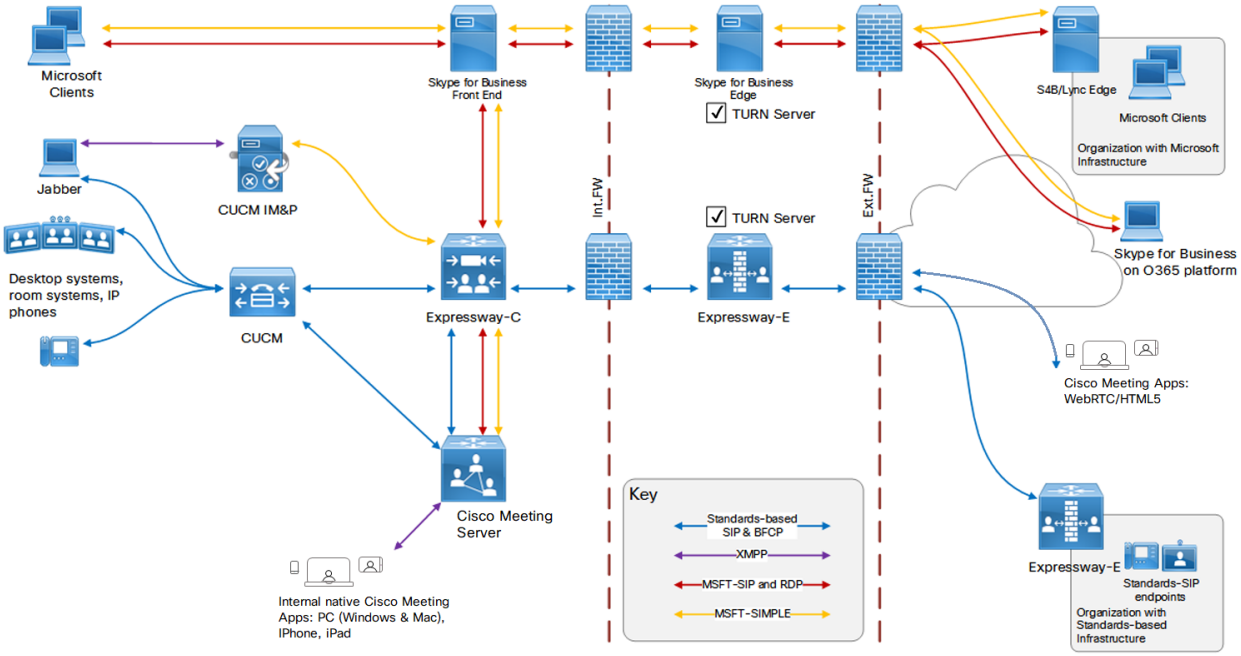


Figure 2: Cisco and Microsoft Infrastructure on-premises deployment example



1.2.1 Using the Cisco Expressway H.323 gateway component

In line with Cisco's goal of a single Edge solution across the Cisco Meeting Server and Cisco Expressway, Cisco plans to end of life the Meeting Server H.323 Gateway component. From version 2.4 of the Meeting Server software, there will be no further bug fixes for the H.323 Gateway component. The H.323 component will be removed from the Meeting Server software in a future release. Customers are encouraged to start evaluation of the more mature H.323 Gateway component in the Cisco Expressway, and plan their migration over.

Any H.323 endpoints registered to Expressway-E or Expressway-C will not consume Rich Media Session (RMS) licenses when calling into the Cisco Meeting Server from Expressway version X8.10 onwards.

1.3 End of Software Maintenance

On release of Cisco Meeting Server software version 2.4, Cisco announces the timeline for the end of software maintenance for version 2.2.

Table 3: Timeline for End Of Software Maintenance for version 2.2

Cisco Meeting Server software version	End of Software Maintenance notice period
2.2	3 months after first release of version 2.4

For more information on Cisco's End of Software Maintenance policy for Cisco Meeting Server click here.

2 New Features/Changes in version 2.4

Release 2.4 of the Meeting Server software adds the following:

- improved mute and unmute meeting controls for Lync, Skype for Business and O365 meetings and a visual indicator on SIP endpoints to show when the endpoints have been muted/unmuted on the AVMCU server (dual homed and gateway calls).
- a facility to control which participant appears in which pane on an endpoint connected to the Meeting Server,
- an increase in call capacity on the Cisco Meeting Server 2000,
- load balancing in Expressway deployments,
- support for the WebRTC version of the Cisco Meeting App on more web browsers,
- an Uploader tool to simplify the work flow for uploading Meeting Server recordings to the video content manager, Vbrick, from a configured NFS. This is a fully released feature from version 2.4.0.
- ability to configure the recording resolution of the Recorder,
- activation key for unencrypted media,
- support for notifying " events clients" in real-time, of changes that are occuring on the Meeting Server,
- ability to use DTMF key sequences in clustered Call Bridge deployments,
- trust stores on the Call Bridge and Web Bridge. The trust stores can hold a certificate whitelist for XMPP server verification. In addition the Call Bridge trust store can hold a certificate white list of Call Bridges in a cluster, increasing the security of the cluster.
- support for more video streams over distribution links creating a more consistent video experience from remote single, dual and three screen end point systems. This is still a preview feature.
- call diagnostic information for active calls added to the Meeting Server log bundle. The log bundle includes the syslog and live.json files, these files will aid Cisco Support speed up their analysis of your issue. To obtain the log bundle, sftp to the Meeting Server and download the file logbundle.tar.gz, see section “Downloading the Log Bundle” in the MMP Command Line Reference guide.
- additional MMP commands,
- new API functionality,
- a new CDR indicates that a conference has been recorded by an endpoint such as a Lync client at any given time,

In addition, from version 2.4:

- you no longer need to purchase a branding license to apply branding to the WebRTC app login page, IVR messages, SIP or Lync call messages or invitation text.

There is also a new interactive API reference tool enabling you to see a high level view of the API objects and drill down to lower levels for the detail, see here for more information.

You are advised not to use beta (or preview) features in a production environment. Only use them in a test environment until they are fully released.

Note: Cisco does not guarantee that a beta or preview feature will become a fully supported feature in the future. Beta features are subject to change based on feedback, and functionality may change or be removed in the future.

Note: The term spaces is used throughout the documentation apart from the API guide which still uses the old terminology of coSpaces.

2.1 New features introduced in version 2.4.6

2.1.1 Additional browser support for WebRTC app

Version 2.4.6 introduces support for Cisco Meeting App for WebRTC using:

- Apple Safari on iOS 12.3
- Apple Safari 12.2 on macOS
- Mozilla Firefox 68

All these browsers are currently in beta (at time of first publication).

See Cisco Meeting App WebRTC Important Information for the latest browser support details.

2.2 New features introduced in version 2.4.5

2.2.1 Safari on iOS for iPhones now fully supported

Version 2.4.5 fully supports Safari on iOS for iPhones, previously it was beta quality. This assumes running the latest version of iOS (recommended). iOS 11.0 is the minimum supported release.

2.3 New features introduced in version 2.4.4

2.3.1 Additional browser support for WebRTC app

Version 2.4.4 introduces support for Cisco Meeting App for WebRTC using Google Chrome version 73.

The expected release date for this version of Chrome is March 12th, 2019. Meeting Server must be upgraded to version 2.4.4 otherwise sharing presentation on WebRTC calls on Meeting Server using Google Chrome, as described below, will not work after updating Chrome to version 73 or above, if the camera permissions are not granted.

For more information, see the Software Advisory notice here and the Bug Search details for CSCvo51143.

Table 4:Cisco Meeting Server support for Google Chrome

Cisco Meeting Server software version	Validated Google Chrome versions
2.4.4	72 and 73 beta

Impact of Chrome 73 on versions earlier than Meeting Server 2.4.4

- When using the WebRTC app on Chrome browser version 73, joining a meeting/call can fail if used in the 'Management and Presentation' mode, and
- If a user has previously blocked the Camera and Microphone permissions, or cannot grant them, they will be impacted if using Chrome 73.

However, if the user has previously granted permission to the browser whilst using the WebRTC app to use the Camera and Microphone, they will not be impacted by Chrome 73. The WebRTC app prompts for these permissions the first time a user tries to join a meeting except in cases where they chose to join using the 'Management and Presentation' mode.

2.4 New features introduced in version 2.4.3

2.4.1 Additional browser support for WebRTC app

Version 2.4.3 introduces support for the WebRTC app using Google Chrome version 72.

The expected release date for this version of Chrome is January 29, 2019. Meeting Server must be upgraded to version 2.4.3 otherwise Chrome users will not be able to use the WebRTC app once version 72 is released.

Table 5:Cisco Meeting Server support for Google Chrome

Cisco Meeting Server software version	Validated Google Chrome versions
2.4.3	71 and 72 beta

2.4.2 New Media Module Status field in the Web Admin interface

Version 2.4.3 introduces a new field for **Media module status** on the Status page (**System > General**). This field shows the number and status of media modules that are operational on the Meeting Server. For example:

- VM: 1/1
- X-series X1: 1/1
- X-series X2: 5/5
- X-series X3: 11/11
- CMS 2K: 7/7

2.4.3 WebRTC App support using Safari on iOS

Version 2.4.3 supports additional browsers, these are:

- Safari on iOS for iPads, running the latest version of iOS (recommended). iOS 11.0 is the minimum supported release.
- Safari on iOS for iPhones, running the latest version of iOS (recommended). iOS 11.0 is the minimum supported release. (This is beta quality in version 2.4.3 and 2.4.4 but is now fully supported in version 2.4.5).

Note: We have tested the WebRTC app using the Safari browser on iPad Air 2 and iPad Pro 12.9 inch (2nd generation) with iOS 11.4.1, iPad (6th generation) with iOS 12.0.1, iPhone 6 on iOS12, iPhone 7 on iOS 12 and 12.1, iPhone 8 Plus on iOS12 and 12.1, and iPhone X on iOS 11.4.1.

2.4.3.1 Important note about audio and video source selection

Camera and microphone selection in the browsers is not very reliable, so we recommend using the Operating System's audio source selection instead of the browser's. We also recommend changing options before a call rather than during the call. Speaker selection via the browser was removed from WebRTC app in Cisco Meeting Server version 2.4.3 and 2.5.1.

2.5 New features introduced in version 2.4.2

2.5.1 Additional browser support for WebRTC App

Version 2.4.2 includes a preview of support for the WebRTC App using the Microsoft Edge browser. This is a beta feature only and you are advised to only use in a test environment, and not to deploy in a production network.

- Microsoft Edge. Supported Edge versions are the latest version of Edge (Microsoft Edge 42/Microsoft EdgeHTML 17) on Microsoft Windows 10, allowing:
  - audio and video calls,
  - pairing with endpoints,
  - receiving a presentation.

  Note: Sharing a presentation using Cisco Meeting App with the Edge browser is now supported in 2.4.3. (Not supported in 2.4.2.)

  Note: In 2.4.2 it is still possible to cross launch the Windows App, but with the introduction of Microsoft Edge support for WebRTC it is no longer the default option.

  Note: Cisco does not guarantee that a beta or preview feature will become a fully supported feature in the future. Beta features are subject to change based on feedback, and functionality may change or be removed in the future.

- Added support for content sharing in Firefox 63.

2.5.2 Office 365 PSTN Audio support

Version 2.4.2 introduces support for PSTN participants in AVMCU conferences. Participants joining an AVMCU conference using the Skype Dial-in phone number will hear and be heard by participants on the Meeting Server side.

Note: Skype for Business doesn't share speaker information of PSTN participants, so Meeting Server cannot detect them and will not mark them as the active speaker. For any other participants (not PSTN) active speaker notifications will work correctly.

Version 2.4 of the Meeting Server software introduced improved mute/unmute meeting controls in dual homed conferences for:

- on-premise and Office 365 Lync/Skype for Business clients,
- end point users,
- Cisco Meeting App users.

Note: This section assumes that muting and unmuting is enabled using the API of the Meeting Server.

Muting/unmuting:

- Lync clients can mute and unmute anyone in the dual homed conference, this means themselves and others, and they can mute and unmute the audience too.
- All endpoint users can now mute Lync clients,
- Endpoint users on the Lync side of the AVMCU can now mute and unmute themselves (self) and other endpoints (either on the Lync clients/endpoints connected to the AVMCU or on the Meeting Server side). Prior to version 2.4, only endpoint users on the Meeting Server side of the AVMCU could mute and unmute themselves (self) and others.
  - For non-ActiveControl endpoints, the Meeting Server sends DTMF key sequences for each mute and unmute, and overlays an icon on the media stream to the endpoint to indicate whether the endpoint is muted or unmuted.
  - For ActiveControl endpoints running CE 9.2.1or later software, the endpoint handles the icons and messages (the Meeting Server does not overlay icons).
- Once an ActiveControl endpoint is muted it has to be unmuted locally so as to ensure the privacy of any local conversation. For example, when a remote participant mutes an ActiveControl endpoint and then tries to unmute it, the ActiveControl endpoint will mute itself again until it is locally unmuted.
- When a remote participant tries to unmute a non-ActiveControl endpoint, the non-ActiveControl endpoint will be unmuted.
- Cisco Meeting App users and Cisco Meeting Management users can mute and unmute Lync clients. They also see the correct mute state of all participants in the meeting.

Muting/unmuting Cisco Meeting App users:

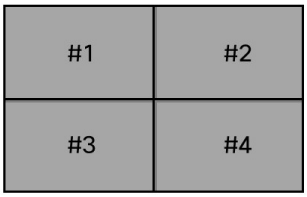
- Information on local muting and unmuting of a Cisco Meeting App user is not passed to Lync clients in dual homed conferences. However, if a Lync client remotely mutes a Cisco Meeting App user and the Cisco Meeting App unmutes itself, the Meeting Server tells the Lync clients about the unmuting.
- When a remote participant tries to unmute a Cisco Meeting App user, the Cisco Meeting App user will remain locally muted. Note: other participants will still see them as unmuted, although they are actually muted.
- The Cisco Meeting App shows the mute/unmute state using its own icons. Meeting Server icons are not overlaid on the Cisco Meeting App video pane.

2.6 Pane placement

From version 2.4, the Meeting Server API pane placement feature enables an administrator to control which participants appear in the panes on SIP endpoints dialed into a conference held within a space. It is common practice to combine pane placement with setting the screen layout, and turning off participants' permission to change their screen layout. The screen layout assigned to the endpoints will determine the panes displayed, and the pane position in the layout determines the order of being filled by the video of participants assigned an importance level.

Figure 3 shows the pane positions on an endpoint with the screen layout set to allEqualQuarters; this is a fixed layout so if there are more than four participants, the layout will not change to display more participants. The numbers indicate the pane position or order of being filled, not an importance value. The pane labeled #1 is filled first, followed by the pane labeled #2 etc. For the sequence of filling panes using other screen layouts, click here.

Figure 3: Pane positions using allEqualQuarters screen layout



Pane placement is controlled through the API using:

- the **`panePlacementHighestImportance`** parameter. This parameter is set on spaces and determines how many panes will be reserved for pane placement,
- the **`importance`** value assigned to participant(s) or Access Method for a specific space, and
- the screen layout selected for the endpoint.

In the **`allEqualQuarters`** layout shown in Figure 3, if **`panePlacementHighestImportance`** is set to 4, then:

- pane #1 will be reserved for a participant with importance set to 4
- pane #2 will be reserved for a participant with importance set to 3
- pane #3 will be reserved for a participant with importance set to 2
- pane #4 will be reserved for a participant with importance set to 1

Pane placement will display a blank pane on the endpoints used by other participants if the participant with the assigned importance has not yet joined the conference in the space (either by dialing in or being dialed out to). This has the effect of fixing the displayed position of the important participant on SIP endpoints other than their own. If importance is used without pane placement, then the pane displaying an important participant will move around as other

important participants join and leave the conference. The important participant is not displayed in a pane on their own endpoint.

Note: Pane placement on the Meeting Server applies conference wide, and is not configurable per participant.

Note: The way a participant joins a conference does not affect pane placement, for instance there is no affect whether a participant dials in to join a conference or is dialed out to.

Note: The pane placement feature is not currently supported by the Recorder or Streamer. Neither is the feature currently supported on Cisco Meeting Management or Cisco Meeting App or using DTMF on a SIP endpoint. If pane placement is selected through the Meeting Server API, then the ability to set importance is disabled in Cisco Meeting Management for meetings where pane placement is in use. Cisco Meeting App does not display blank panes.

2.6.1 Using pane placement

Pane placement works only on conferences occurring within spaces, it does not work on calls. To use pane placement:

1. Set a value for parameter **`panePlacementHighestImportance`** for the space, this defines the highest importance level to be used in the space:
   - for an existing specific space, PUT to **`/coSpaces/<coSpace id>`** the request parameter **`panePlacementHighestImportance`** set to the chosen value,
   - to create a new space with pane placement set, POST to **`/coSpaces`** with the request parameter **`panePlacementHighestImportance`** set to the chosen value.
2. Assign importance values to the important participants who will connect to the meeting. Setting the importance value for a participant has been available since version 2.2:
   - create a new participant with the assigned importance for the specified call, POST to **`/calls/<call id>/participants`** with the request parameter **`importance`** set to the chosen value, or
   - assign an importance to a specified participant in the conference, PUT to **`/participants/<participant id>`** with the request parameter **`importance`** set to the chosen value

   Alternatively, an importance value can be assigned to an Access Method for a specific coSpace (from version 2.4):
   - create a new accessMethod for the specified coSpace, POST to **`/coSpaces/<coSpace id>/accessMethods`** the request parameter **`importance`** set to the chosen value, or

- update an existing Access Method for the coSpace, PUT to **/coSpaces/<coSpace id>/accessMethods/<access method id>** the request parameter **importance** set to the chosen value.

Once pane placement is operational, the following rules are applied:

- Number of participants shown on an endpoint will depend on the layout selected for that endpoint.
- Participants are placed based on their importance value (highest importance is placed first). A blank pane is inserted if no participants match a specific importance level, for instance because the level was not assigned, the participant has yet to join the meeting or has already left the meeting. This has the effect of fixing the displayed position of the important participant on SIP endpoints other than their own. If importance is used without pane placement, then the pane displaying an important participant will move around as other important participants join and leave the conference. The important participant is not displayed in a pane on their own endpoint.
- Participants with a higher importance than specified in **panePlacementHighestImportance** for the space, appear "at the top of the layout", no blank panes are added for importance values between this participant's importance and the **panePlacementHighestImportance** value. Similarly, if an Access Method to a space is configured with a higher importance value for **panePlacementHighestImportance** than that set for the space, then participants with importance values greater than the **panePlacementHighestImportance** set for the space will be displayed first, but blank panes will not represent gaps in the importance levels between the value set for the Access Method compared to the space.
- A participant never sees themselves, and a blank pane is not shown for them on their endpoint.
- If a participant with no importance set is reached before the panes on the screen run out, the remaining layout panes will be filled with any participants not yet placed, but are in the active speaker history order (most recent speakers first). Blank panes are not added in between them and participants with assigned importance.
- If multiple participants are given the same importance value, then they are ordered according to who is the most frequent active speaker. In this mode, there is no reordering of participants to keep them in the same pane, so participants will move between panes.
- If the conference is split across several Meeting Servers, the participants that are shared across the distribution link appear in the panes according to their importance. If the participant is present, but the video stream isn't sent over the distribution link, an empty pane is inserted. As the most important participants are shared over the distribution link, the more important (higher importance value) panes will always be filled for all participants. However, it will depend on which Meeting Server a participant is hosted on as to whether they see a participant or an empty pane.

For more information on screen layouts and Pane placement see the Cisco Meeting Server Administrator Quick Reference Guide on Screen Layouts and Pane Placement.

2.6.2 Removing pane placement

To remove pane placement, leave the **panePlacementHighestImportance** parameter as unset (leave the parameter value as blank).

2.6.3 Example of using Pane Placement

In this example, the **panePlacementHighestImportance** has been set to 8 for the space, this implies that there are participants with importance of 8,7,6,5,4,3,2,1; 8 being the most important and 1 the least important of those assigned an importance level.

Participants have been assigned the following importance levels:

Table 6: Assigned importance levels to participants

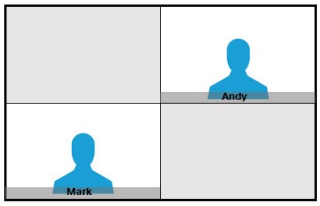
Participant	Importance
Kimberley	8
Andy	7
Mark	6
Sally	4
Bob	3
Julie	1

There are no participants configured with importance 5 or 2.

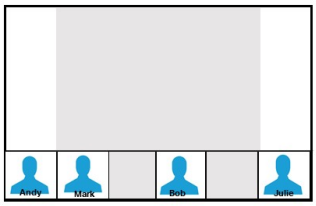
Bob's endpoint is configured for a 2x2 layout, and Sally and Julie's endpoints are configured for stacked layout. Before Kimberley joins the meeting, there is nobody in the meeting with importance 8. Bob's endpoint shows blank panes for importance 8 and 5, Sally and Julie's endpoints show blank panes for importance 8,5 and 2. Since Sally doesn't appear on her own endpoint, Bob appears in different panes in Sally's view compared to Julie's view, see Figure 4.

Figure 4: Pane placement at start of meeting

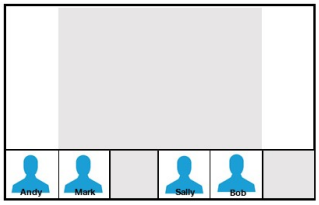
Participant	Importance
Andy	7
Mark	6
Sally	4
Bob	3
Julie	1



Bob's view



Sally's view

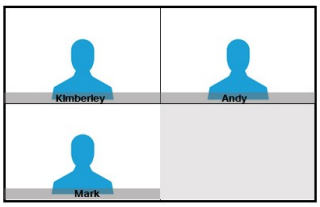


Julie's view

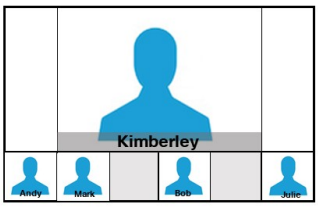
When Kimberley joins the meeting, Bob's endpoint only shows one blank pane for importance 5, Sally and Julie's endpoints show blank panes for importance 5 and 2. Again Bob appears in different panes in Sally's view compared to Julie's view due to Sally not appearing on her own endpoint, see Figure 5.

Figure 5: Pane placement when Kimberley joins the meeting

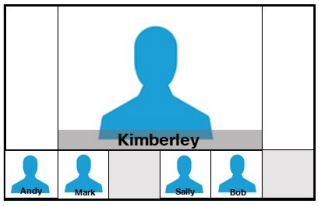
Participant	Importance
Kimberley	8
Andy	7
Mark	6
Sally	4
Bob	3
Julie	1



Bob's view



Sally's view



Julie's view

2.7 Cisco Meeting Server 2000 call capacity

From version 2.4 the Cisco Meeting Server 2000 has an increased call capacity to 700 HD (720p30) calls or 350 full HD (1080p30), on a single Meeting Server or cluster of Meeting

Servers, see 2.7 below. The HD call capacity of Meeting Servers within a Call Bridge group remains at 500, see Table 8.

Table 7: Call capacity on a single Cisco Meeting Server 2000 or clustered servers

Type of call	Call capacity from version 2.4	Call capacity, prior to version 2.4
Full HD (1080p30)	350	250
HD (720p30)	700	500
SD (448p30)	1000	1000
Audio	3000	3000

Table 8: Call capacity on a Cisco Meeting Server 2000 within a Call Bridge Group

Type of call	Call capacity
Full HD (1080p30)	250
HD (720p30)	500
SD (448p30)	1000
Audio	3000

In addition, this software release supports an increase in the number of participants in a conference hosted on the Cisco Meeting Server 2000, 450 is now the participant limit per conference per server across all Meeting Server platforms.

Note: The number of participants per conference across distributed servers remains at 2600.

2.8 Load balancing across Meeting Servers when using Expressway

A typical large scale deployment consists of several Meeting Servers deployed at multiple offices/data centers. For scalability and resilience of the conferencing service, the Call Bridges will typically be configured as a cluster.

Using Call Bridge Groups, a Meeting Server cluster can intelligently load balance calls across the Call Bridges within the same location or across nodes in different locations. The intelligent decision making behind where calls end up, is handled by the Meeting Servers. The call control system needs to be able to handle SIP messages from the Meeting Servers, in order to move calls to the correct location. Prior to version 2.4, this functionality had been tested using Cisco Unified Communications Manager as the call control system. From version 2.4, the Meeting Server also supports load balancing calls that arrive at the Meeting Server via Expressway. For load balancing with Cisco Expressway, use Cisco Expressway release X8.11 or later and Cisco Meeting Server release 2.4 or later.

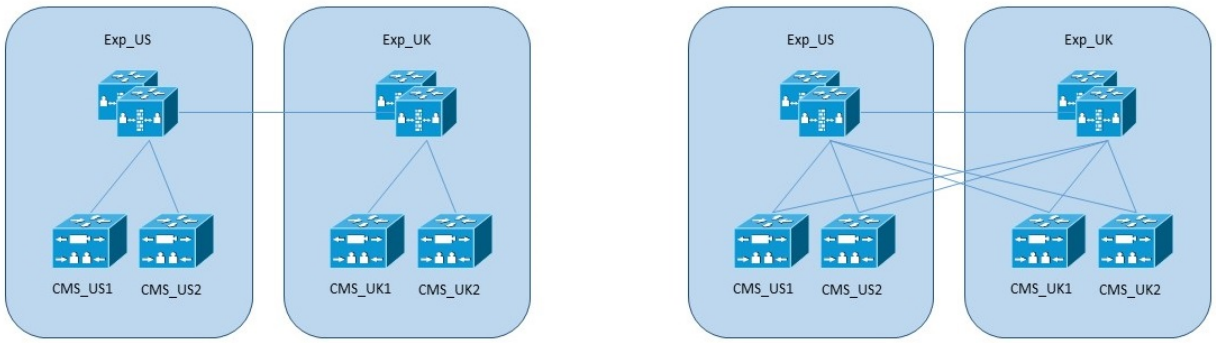
Note: Dual-homed conferences with a Meeting Server cluster are not currently supported with Expressway X8.11 as the edge for the Meeting Server, unless at least some of the Microsoft traffic flows directly between one of the Meeting Servers in the cluster and the Microsoft infrastructure (and not through Expressway). Dual-homing is supported with Expressway X8.11 as the edge for standalone Meeting Servers.

2.8.1 Supported Expressway scenarios

The white paper Load Balancing Calls Across Cisco Meeting Servers details two deployment examples for load balancing incoming calls when using Cisco Expressway:

- Example 1 has the Meeting Servers trunked to their local Cisco Expressway cluster. The Cisco Expressways connect to each other via a neighbor zone. Neighbor zones allow a Cisco Expressway to connect to a remote Meeting Server via another Cisco Expressway cluster. Meeting Servers are trunked to the local Cisco Expressway devices. Endpoints in the US all connect to the call control in the US (EXP_US), and similarly for the UK. Ideally calls originating in the US should use the US Call Bridges and similarly the UK endpoints should end up connected to the UK Call Bridges.
- Example 2 has trunks from each Cisco Expressway cluster to every Meeting Server. This deployment uses zones to allow a Cisco Expressway to directly connect to a remote Meeting Server. Meeting Servers are trunked to the local Cisco Expressway devices. Endpoints in the US all connect to the call control in the US (EXP_US), and similarly for the UK. Ideally calls originating in the US should use the US Call Bridges and similarly the UK endpoints should end up connected to the UK Call Bridges.

Figure 6: Two example deployments for load balancing incoming calls using Cisco Expressway



Load balancing incoming calls using Expressway in the deployment is achieved by:

- creating a Call Bridge Group of local Call Bridges in a location,
- configuring a zone , dial plan search rules and setting **Meeting Server load balancing** = **on**, on the Expressway,

- enabling the load balancing of calls across the Call Bridge Group by setting **`loadbalancingEnabled`** = **true** and **`loadbalanceIndirectCalls`** = **true** on the Meeting Server API object **`/callBridgeGroups`**.

2.8.2 Call Bridge Grouping

The load balancing of calls occurs between a group of Call Bridges that exist in the same location. To configure which Call Bridges are in each location, the concept of Call Bridge groups is used. A Call Bridge group defines the set of nodes in a location, and then load balancing can be enabled for different types of call to this group. A location could refer to a single data center, or a whole continent. The decision of how to group Call Bridges will depend on the specifics of network configuration and the desired behavior.

For the load balancing feature to work correctly, a Round Trip Time (RTT) of less than 100 ms is required for the servers in a Call Bridge group. The maximum RTT between any two nodes in the same cluster remains as 300 ms.

By default, if you have Call Bridge Groups configured, and load balancing activated, then calls from new participants are rejected at 80% load, and only distribution calls will be allowed.

To determine the media processing load on a Call Bridge, use the API object **`/system/load`**. You can change the load limit on each server in the cluster from the default 80%.

Note: If you are not using load balancing with Call Bridge Groups, then calls will not be rejected, but the quality of all calls will be reduced when the load limit is reached. If this happens often, we recommend that you buy additional hardware.

2.8.3 Configuring a Cisco Expressway dial plan

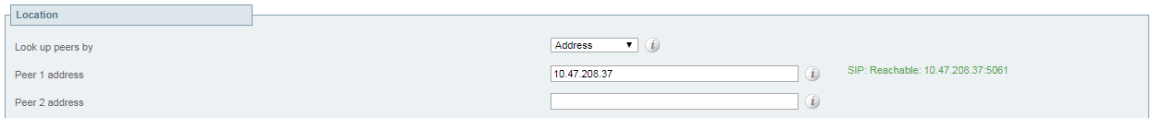
Call routing in Cisco Expressway deployments relies on the use of dial plans and zones, and it is assumed that these concepts are understood. For information on configuring dial plans and zones, please consult the Expressway documentation.

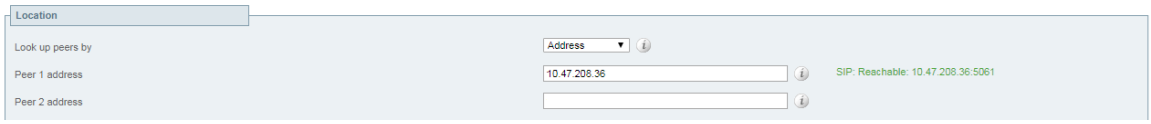
Dial plans are used by the call control system to favor sending calls to local Meeting Server resources, thereby reducing inter-office bandwidth.

On the call control device:

1. Configure a zone for each Call Bridge in the Call Bridge Group.
   a. Navigate to **Configuration** > **Zones** and create a **New** zone of type **Neighbor**, that this cluster will be directly communicating with. You need one zone per Call Bridge node.

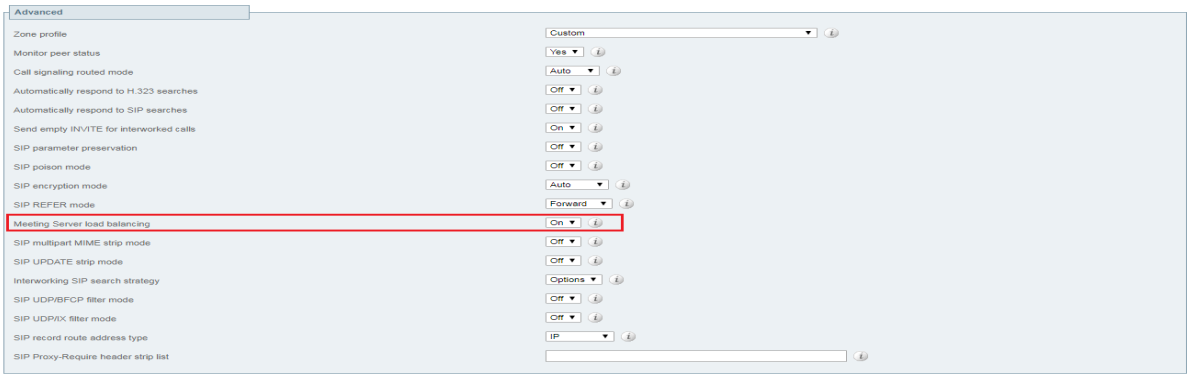
Figure 7: Creating new zones on Expressway





2. In the **Advanced** section, select the **Zone profile** as **Custom** and set the **Meeting Server load balancing** parameter to **On**.

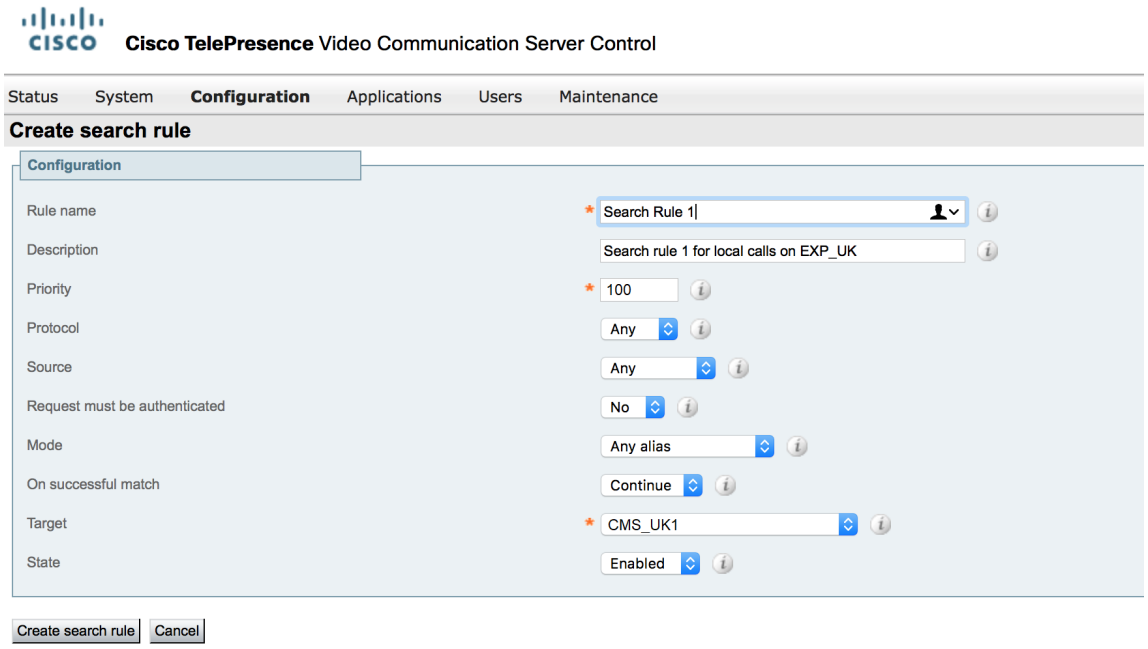
Figure 8: Setting the Meeting Server load balancing parameter



3. For each zone, create a search rule pointing to the zone. Navigate to **Configuration** > **Dial plan** > **Search rules** and create a **New** search rule. Each search rule must have a different priority, and **On successful match** must be set to **Continue**.

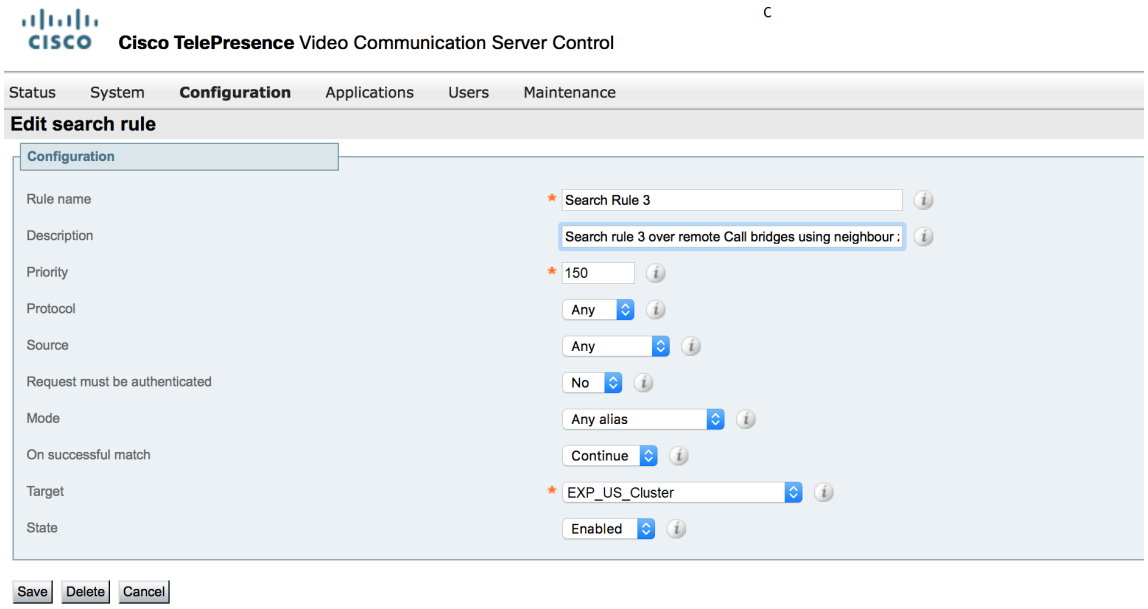
   When choosing priorities it is important to consider the order in which you want this Cisco Expressway to use resources. Local resources would typically have higher priority than remote resources.

Figure 9: Creating a search rule for connections to local resources



If remote resources are going to be used via other Cisco Expressway clusters, configure search rules that resolve to these. These rules would typically have a lower priority than rules to local Meeting Servers, and typically have a source set to avoid call loops.

Figure 10: Creating search rules for connections to other Cisco Expressway clusters



2.8.4 Enabling load balancing on local Meeting Servers in Expressway deployments

To enable the load balancing of inbound and outbound SIP calls through Expressway X8.11 use the Meeting Server API to set **`loadBalanceIndirectCalls=true`** and **`loadBalancingEnabled=true`** using a POST to **`/callBridgeGroups`** for a new Call Bridge Group, or using a PUT to **`/callBridgeGroups/<call bridge group id>`** for an existing Call Bridge Group.

Note: By default, the **`loadBalanceIndirectCalls`** parameter is set to false, disabling load balancing in Expressway deployments.

2.9 Web browsers supporting the WebRTC app

Prior to version 2.4 of the Meeting Server software, the WebRTC version of the Cisco Meeting App was only supported on Chrome. From version 2.4 the WebRTC app is supported on the following browsers:

- Google Chrome for Windows, macOS and Android. Use Chrome version 66 or later. We strongly recommend using the most recent version of Chrome.
- Mozilla Firefox for Windows and macOS. Use Firefox is 59.0.2 or later. We strongly recommend using the most recent version of Firefox.
- Apple Safari for macOS. Use Safari 11.1 or later. We strongly recommend using the most recent version of Safari.

Note: Content cannot be sent from Safari on macOS, iOS or from Chrome on Android, this is a browser limitation.

For more information on browser support and supported devices, see the Cisco Meeting App WebRTC Important Information.

2.10 Recorder improvements

Version 2.4 features four improvements to the Recorder component of the Meeting Server. These are:

- the ability to set the recording resolution of the Meeting Server Recorder,
- a recording indicator to show when a Lync client is recording a dual homed conference,
- an addition to the API, enabling you to determine whether a Lync client is recording a dual homed conference,
- a new parameter **`endpointRecorded`** added to the **`callEnd`** CDR record, indicating whether or not the call was recorded by an endpoint such as a Lync client.

In addition, the Uploader component, first introduced in version 2.3 as a beta feature, is a fully released feature in version 2.4.0.

2.10.1 Setting the resolution of the Recorder

From version 2.4, you can configure the recording resolution of the Meeting Server Recorder. The resolution is configured on the Recorder component itself and is not passed to the Recorder by the Call Bridge. To configure the resolution use the MMP command: **`recorder resolution <audio|720p|1080p>`**.

If no resolution is configured, the default setting is 720p30. Audio recordings are stored in .mp4 file format.

Table 9 provides typical specifications for the different recorder settings, the recommendations are based on our internal testing.

Table 9: Recorder resolution specifications

Recorder setting	Percentage of physical core per recording	RAM required per recording	Typical disk usage per hour	Planned disk usage per hour (recommended)
1080p	100%	1GB	1GB to 1.6GB	2GB
720p	50%	0.5GB	300MB to 800MB	1GB
audio	20%	125MB	70MB	100MB

2.10.2 Example of setting the recording resolution

This example assumes you already have a working Recorder.

1. SSH into the MMP of the Meeting Server hosting the Recorder.
2. Disable the Recorder to change the configuration.
   `recorder disable`
3. Configure the Recorder to record meetings at the specified resolution using the MMP command:
   `recorder resolution <audio|720p|1080p>`

   For example:
   `recorder resolution 1080p`
4. Re-enable the Recorder so that it picks up the new configuration.
   `recorder enable`

2.10.3 Recording indicator for dual homed conferences

For dual homed conferences, recording should be done using the Microsoft recording method on the Lync/Skype endpoint. We do not recommend using Cisco Meeting Server to record dual homed conferences.

From version 2.4, a recording icon indicates to SIP participants connected to the Meeting Server that a Lync/Skype endpoint is recording the conference on the Lync/Skype side.

Meeting Server adds a recording icon to the video pane composed for non-ActiveControl endpoints. Table 10 below shows the icons that Meeting Server will display to indicate that a dual homed conference is being recorded.

Table 10: Recording indicators

Icon displayed	Description
	Meeting is being recorded via the Meeting Server.
	Meeting is being recorded by a Lync/Skype endpoint
	Meeting is being recorded via the Meeting Server and by a Lync/Skype endpoint.
	The meeting is not being recorded (no icon displayed).

Note: The Cisco Meeting App shows the recording state using its own icons, they do not distinguish between local and remote recording. Meeting Server icons are not overlaid on the Cisco Meeting App video pane.

2.11 Recording with Vbrick

Note: The Uploader component is a fully released feature in version 2.4.0.

The Uploader component simplifies the work flow for uploading Meeting Server recordings to the video content manager, Vbrick, from a configured NFS connected to a Meeting Server. No manual importing of recordings is required.

Once the Uploader component is configured and enabled, recordings are pushed from the NFS to Vbrick, and an owner is assigned to the recording. The Rev portal applies security configured by your administrator to your video content, only allowing a user to access the content that they are permitted to access. Vbrick emails the owner when the recording is available in the owner's Rev portal. Owners of a recording access the video content through their Rev portal, and can edit and distribute as necessary.

Note: If a file is added to the NFS share within a space directory, the file will be uploaded to Vbrick as though it were a valid recording. Take care to apply permissions to your NFS share so that only the Recorder can write to it.

Note: Depending on the mechanism you use to store the recordings you may need to open external firewall ports so that the recorder, uploader and storage system can communicate. For example: NFS running version 2 or 3 of the port mapper protocol uses TCP or UDP ports 2049 and 111.

Note: Do not use the Firewall component on the Meeting Server if using either the Recorder or Uploader.

2.11.1 Prerequisites for the Meeting Server

Uploader installation. The Uploader component can be installed on the same server as the Recorder component, or on a separate server. If installed on the same server as the Recorder, then add a couple of vCPUs for it to use. If run on a different server, then use the same server specification as for the Recorder: dedicated VM with a minimum of 4 physical cores and 4GB.

CAUTION: The Uploader must run on a different Meeting Server to the Call Bridge hosting the conferences.

Read and Write access to the NFS share. The Meeting Server running the Uploader will require Read and Write permissions for the NFS. Write permission is required to allow the Uploader to re-write the name of the mp4 file when upload is completed.

Note: If the NFS is set or becomes Read Only, then the Uploader component will continuously upload the same video recording to Vbrick. This is a result of the Uploader being unable to mark the file as upload complete. To avoid this, ensure that the NFS provides read/write access.

API Access to Vbrick Rev. Configure API access for a user on Vbrick Rev.

API Access to Call Bridge. Configure API access for a user on the Meeting Server running the Call Bridge.

Trust Store Store the certificate chains from the Vbrick Rev server, and the Meeting Server running the Web Admin interface for the Call Bridge. The Uploader needs to trust both the Vbrick Rev and the Call Bridge.

Decide who has access to the video recordings. Access to uploaded video recordings can be set to: All Users, Private, and for only space owners and members.

Default state of video recordings. Decide whether the video recordings are immediately available after upload (Active), or that the owner of the video recording needs to publish it to make the recording available (Inactive).

Table 11: Port Requirements

Component	Connecting to	Destination port to open
Call Bridge	NFS (version 3)	2049
Uploader	Web Admin of Call Bridge	443 or port specified in Uploader configuration
Uploader	Vbrick Rev server	443 for video uploads and API access to Vbrick Rev server

2.11.2 Configuring the Meeting Server to work with Vbrick

These steps assume that you have already setup the NFS to store recordings.

1. Establish an SSH connection to the MMP of the Meeting Server where you want to run the Uploader. Log in.
2. For new Vbrick installations, ignore this step. If you are reconfiguring a Vbrick installation then first disable Vbrick access to the Meeting Server.
   `uploader disable`
3. Specify the NFS that the Uploader will monitor.
   `uploader nfs <hostname/IP>:<directory>`
4. Specify the Meeting Server that the Uploader will query for recording information, for example the name of the Meeting Server hosting the space associated with the recording.
   `uploader cms host <hostname>`
5. Specify the Web Admin port on the Meeting Server running the Call Bridge. If a port is not specified, it defaults to port 443.
   `uploader cms port <port>`
6. Specify the user with API access on the Meeting Server running the Call Bridge. The password is entered separately.
   `uploader cms user <username>`
7. Set the password for the user specified in step 6. Type
   `uploader cms password`
   you will be prompted for the password.
8. Create a certificate bundle (crt-bundle) holding a copy of the Root CA's certificate and all intermediate certificates in the chain for the Web Admin on the Meeting Server running the Call Bridge.
9. Add the certificate bundle created in step 8 to the Meeting Server trust store.
   `uploader cms trust <crt-bundle>`
10. Configure the Vbrick host and the port to which the Uploader will connect.
    `uploader rev host <hostname>`
    `uploader rev port <port>`

    Note: The port defaults to 443 unless otherwise specified.

11. Add a Vbrick Rev user who has API permission to upload video recordings.
    uploader rev user <username>
12. Set the password for the user specified in step 11. Type
    uploader rev password
    you will be prompted for the password.
13. Create a certificate bundle (crt-bundle) holding a copy of the Root CA's certificate and all intermediate certificates in the chain for the Vbrick Rev server.
14. Add the certificate bundle created in step 13 to the Vbrick Rev trust store.
    uploader rev trust <crt-bundle>
15. Set access to the video recording.
    uploader access <Private|Public|AllUsers>
16. Give members of the space the ability to view or edit the recordings.
    uploader cospace_member_access <view|edit|none>

    Note: This step requires the listed members to have valid email addresses which are associated with accounts on Vbrick. For example **user1@example.com**

17. Decide whether the owner of the space is the single owner of the video recordings.
    uploader recording_owned_by_cospace_owner <true|false>

    Note: This step also requires the owner of the video recordings to have a valid email address which is associated with an account on Vbrick.

18. If the owner of the space is not listed in Vbrick Rev, then set the username of the fallback owner. If the fallback owner is not specified, then owner will default to the user configured on the MMP.
    uploader fallback_owner <vbrick-user>
19. Enable comments to the video recordings.
    uploader comments enable
20. Enable ratings for the video recordings.
    uploader ratings enable
21. Set the download permission for the video recordings.
    uploader downloads enable
22. Set the default state of the video recording when first uploaded to Vbrick Rev.
    uploader initial_state <active|inactive>
23. Decide whether to delete the video recording from the NFS after upload is complete
    uploader delete_after_upload <true|false>
24. Enable the Uploader to access the Meeting Server
    uploader enable

Note: Set **`messageBoardEnabled`** to **`true`** to see the messages being posted in the space indicating that the recording is available.

2.12 Changes to licensing for branding

Some aspects of the participant experience of meetings hosted on Meeting Servers can be branded, they include :

- the WebRTC app sign in background image, sign in logo, text below sign in logo and the text on the browser tab,
- IVR messages,
- SIP and Lync participant's splash screen images and all audio prompts/messages,
- some text on the meeting invitation.

From version 2.4, no license is required to apply single or multiple brands to these customizable features. If you apply a single brand with only a single set of resources specified (one WebRTC app sign-in page, one set of voice prompts, one invitation text), then these resources are used for all spaces, IVRs and Web Bridges in the deployment. Multiple brandings allow different resources to be used for different spaces, IVRs and Web Bridges. Resources can be assigned at the system, tenant, space or IVR level using the API.

2.13 Activation key for unencrypted SIP media

Prior to version 2.4, you could only purchase an activation key with SIP media encryption enabled. Media includes audio, video, content video and ActiveControl data. From version 2.4, you have the choice of purchasing an activation key with SIP media encryption enabled or SIP media encryption disabled (unencrypted SIP media) for the Cisco Meeting Server 1000, Cisco Meeting Server 2000 and the VM software image. Choose either encrypted or unencrypted options under the software pids R-CMS-K9 and R-CMS-2K-K9.

Note: Current Call Bridge activations are unaffected, unless an activation key is uploaded with SIP media encryption disabled.

2.13.1 Unencrypted SIP media mode

If the activation key for SIP media encryption disabled is uploaded to the Meeting Server, then the following occurs:

- media sent between the Meeting Server and SIP devices is unencrypted,
- media sent over distribution links between clustered Call Bridges is unencrypted,
- call signalling remains encrypted,

- media in calls between the Meeting Server and Cisco Meeting App, on any platform, remains encrypted,
- an error message is returned if the **sipMediaEncryption** parameter is set to anything other than **prohibited** on the following API objects:

  /calls/<call id>/participants

  /calls/<call id>/callLegs

  /callLegs/<call leg id>

  /callLegProfiles and **/callLegProfiles/<call leg profile id>**

  /callLegs/<call leg id>/callLegProfileTrace
- an error message is displayed if the SIP media encryption field on the the Configuration>Call settings web page of the Web Admin interface is set to anything other than disabled.

Note: If SIP media encryption is disabled, call signaling can still be encrypted on outbound calls, if required, by setting the **sipControlEncryption** parameter on **/outboundDialPlanRules**.

2.13.2 Determining the Call Bridge media mode

To determine whether the Call Bridge uses encrypted or unencrypted SIP media use a GET on API object **/system/licensing**. If the **features** response value has the **status** of **callBridgeNoEncryption** set to **activated** then an activation key for unencrypted media is loaded on theCall Bridge. Other valid settings for the **status** of **callBridgeNoEncryption** are **noLicense**, **grace** or **expired**.

callBridgeNoEncryption also has an **expiry** field in the form of a string.

2.14 Events

From version 2.4, the Meeting Server can notify an " events client" in real-time of changes that are occurring on the Meeting Server. The Meeting Server acts as a server for the events, and the events client could be for example, a web-based management application. Cisco Meeting Management acts as an events client.

Note: You can construct your own events client, which is similar to constructing an API client. The events client needs to support HTTP and WebSocket libraries, both are available in common scripting languages like Python. The events port on the Meeting Server is the same port as you configured for the Web Admin, typically TCP port 443 on interface A.

Rather than continually poll an API resource on the Meeting Server, an events client can subscribe to an event resource to receive updates. For example, after establishing a WebSocket connection between the events client and the Meeting Server, the events client can subscribe to the event resource **callRoster** and receive updates on the participant list of an active

conference to find out when a new participant joins, or an existing participant changes layout etc.

2.14.1 Subscription overview

When an events client makes a subscription to the Meeting Server, the subscription lists the set of resources the events client wants to subscribe to. This should be the complete list of resources that it wants to have an active subscription to, so if the events client needs to add a new resource to an existing set of subscriptions then it must include all the existing subscribed-to resources in the new request. To stop all active subscriptions, the events client should supply an empty request (this will also happen when the WebSocket connection is torn down).

Each resource being subscribed to within the subscription request is given a unique number by the events client, so that when the Meeting Server provides updates the client knows which subscription is being referred to. If the events client subscribes to the same resource as before, but with a different numeric identifier, this is treated as a new subscription by the Meeting Server, and the old subscription request is effectively torn down.

When the Meeting Server receives a new subscription request, it first replies with a simple " ack" saying it has received that subscription request and is processing it. A positive acknowledgement at this stage only means that the Meeting Server has received the request - there will be one or more follow ups giving an updated status of each element of the subscription request, and thereafter actual updates for that resource.

As an example, if the events client requests to subscribe to information on active conferences and the participant list for one specific conference, it will first get back an " ack" , and then a " subscriptionUpdate" message telling it that both subscriptions are " pending, this means that the Meeting Server is still in the process of setting up the subscriptions. A little while later the events client might get an update that says the subscription to the list of active conferences is " active" and the participant list subscription is still " pending" . The Meeting Server will also start providing updates for the active conferences list at this stage; no actual updates relating to a subscribed-to resource occur before the subscriber has been told that subscription is " active" . If the conference, whose participant list the events client subscribed to, exists (i.e. the supplied GUID still corresponds to an active conference) the events client will receive a subscription update indicating that both the active conference subscription and the participant list subscription are " active" , the events client will also start receiving participant list updates at this point. If the conference GUID for the participant list subscription isn't successfully resolved to an active conference, the subscription status will be " deactivated" . This is also the state that the subscription will change to in a subsequent " subscriptionUpdate" message when the conference ends or the events client unsubscribes from it.

For more information on the flow of messages when subscribing to event resources, see Section 2.14.5.

2.14.2 Event resources and subscribable elements

From version 2.4, the Meeting Server supports using events to provide real-time information to an " events client" on the following event resources:

- **`callInfo`** provides information about a specific conference,
- **`callRoster`** provides information about each participant in a conference,
- **`calls`** provides information on active conferences.

Table 12: Information about a specific conference (call) available using subscribable event resource `callInfo`

Name	Value	Description
Request parameters		
call	ID	The id of the conference to receive updated information on.
Subscribable elements		
name	String	The name of the conference.
participants	Numeric	The number of participants currently in the conference.
distributedInstances	Numeric	The number of distributed instances of this conference that exist across the Call Bridge cluster (0 for an unclustered conference).
recording	active| inactive	One of: *active* - this conference is currently being recorded *inactive* - this conference is currently not being recorded.
endpointRecording	active| inactive	One of: *active* - this conference is currently being externally recorded by an endpoint (Lync client) *inactive* - this conference is currently not being externally recorded by an endpoint (Lync client).
streaming	active| inactive	One of: *active* - this conference is currently being streamed *inactive* - this conference is currently not being streamed.
lockState	locked| unlocked	One of: *locked* - this conferenc. is currently locked *unlocked*- this conference is currently unlocked .

Name	Value	Description
callType	coSpace| adHoc| lyncConferencing| forwarding	One of: *coSpace* - this call is a coSpace instantiation *adHoc* - this is an ad hoc multi-party call *lyncConferencing* - this call is a Meeting Server connection to a Lync-hosted conference *forwarding* - this is a forwarded or "gateway" call.
callCorrelator	ID	This value can be used to identify call legs which may be distributed across multiple Call Bridges, but which are all in the same call either in the same coSpace or an ad hoc call. Note: For calls within a coSpace, the callCorrelator value will be the same for the life time of the coSpace. For every ad hoc call, the value will be dynamically generated.
joinAudioMuteOverride	true| false	One of: *true* - new participants will be muted when joining the conference *false* - new participants will not be muted when joining the conference. This is the default if not set.

Table 13: Information about each participant in a conference (call) available using subscribable event resource `callRoster`

Name	Value	Description
Request parameters		
call	ID	The id of the conference to receive participant updates from.
Subscribable elements		
name	String	The participant display name.
uri	URI user part	The URI associated with this participant.
state	initial| ringing| connected| onHold	Reflects the current signaling state of this participant.
direction	incoming| outgoing	One of: *incoming* – this participant dialled into the conference (the remote SIP device initiated the connection to the Meeting Server) *outgoing* – this participant was dialled out to inorder to join the conference (the call leg was established from the Meeting Server to the remote SIP device).
audioMuted	true| false	One of: *true* – the Meeting Server has muted this participant's audio *false* – the Meeting Server has not muted this participant's audio.
videoMuted	true| false	One of: *true* – the Meeting Server has muted this participant's video *false* – the Meeting Server has not muted this participant's video.
importance	Numeric	Value for this participant's importance. NULL if importance is not set or changes to unset.

Name	Value	Description
layout	allEqual| speakerOnly| telepresence| stacked| allEqualQuarters| allEqualNinths| allEqualSixteenths| allEqualTwentyFifths| onePlusFive| onePlusSeven| onePlusNine| automatic| onePlusN	The layout currently being used by this participant.
activeSpeaker	true| false	One of: *true* – this participant is currently considered an active speaker in this conference *false* – this participant is currently not considered an active speaker in this conference
presenter	true| false	One of: *true* – this participant is currently presenting (sharing their screen) in this conference *false* –this participant is currently not presenting in this conference.
endpointRecording	active| inactive	One of: *active* – this participant is currently recording the conference *inactive* – this participant is currently not recording the conference.

Table 14: Information on active conferences (calls) available using subscribable event resource `calls`

Name	Value	Description
Mandatory response elements		
call	ID	The id of the conference whose elements have been updated.
Subscribable elements		
name	String	The name of the conference.
participants	Numeric	The number of participants currently in the conference.
distributedInstances	Numeric	The number of distributed instances of this conference that exist across the Call Bridge cluster (0 for an unclustered conference).
recording	active| inactive	One of: *active* - this conference is currently being recorded *inactive* - this conference is currently not being recorded.
endpointRecording	active| inactive	One of: *active* - this conference is currently being externally recorded by an endpoint (Lync client) *inactive* - this conference is currently not being externally recorded by an endpoint (Lync client).
streaming	active| inactive	One of: *active* - this conference is currently being streamed *inactive* - this conference is currently not being streamed.
lockState	locked| notLocked	One of: *locked* - this conference is currently being locked *notLocked* - this conference is currently not being locked.
callType	coSpace| forwarding| adHoc| lyncConferencing	One of: *coSpace* - this conference is the instantiation of a coSpace *forwarding* - this is a forwarded/" gateway" call *adHoc* - this is an ad hoc multi-party call *lyncConferencing* - this call leg is participating in a Lync conference.
callCorrelator	ID	The correlator GUID which is the same across all distributed instances of the call.

2.14.3 Call Bridge Groups and clusters

When a conference is hosted across a Call Bridge Group (or cluster), messages are passed between the Call Bridges in the group to ensure that each Meeting Server knows the active

conferences and the roster list information for any conference for which it has one or more participants.

Subscriptions receive updates for local and remote participants to a conference, but when a remote participant leaves the conference, the parameter 'reason' won't be provided. Only subscriptions to the Call Bridge hosting the conference will receive the reason why the participant has left the conference. To receive information on all active conferences across the Call Bridge Group or cluster, the events client will need to subscribe to event resource **`calls`** on every Meeting Server, but to see event resources **`callInfo`** or **`callRoster`**, the events client only needs to subscribe to one of the Meeting Servers which reports on the conference.

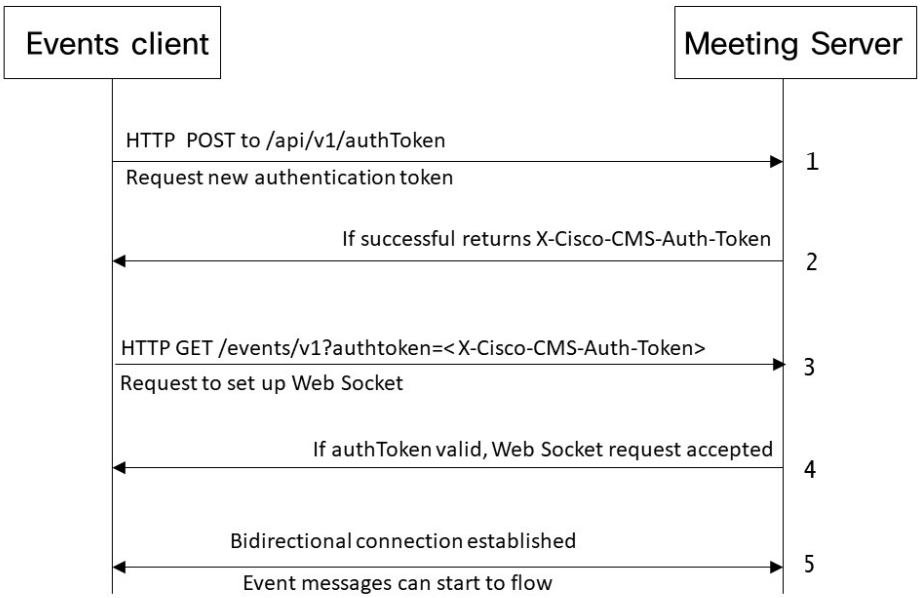
When a Meeting Server is no longer participating in the conference due to participants hosted on its Call Bridge leaving the conference, the Call Bridge will deactivate the subscription, and the events client will have to pick another Meeting Server.

2.14.4 Example authentication flow

Before being allowed to subscribe to events on the Meeting Server, the events client needs to be authenticated by the Meeting Server and a connection using the WebSocket Protocol established between the events client and the Meeting Server. This connection enables the events client to receive event information without needing to constantly poll the API of the Meeting Server.

Figure 11 illustrates the call flow between the events client and the Meeting Server required to establish a WebSocket. You will need to use Python or a similar coding language for the events client to use to send the HTTP POSTs and GETs to the Meeting Server.

Figure 11: Overview of call flows establishing a WebSocket between the events client and the Meeting Server



1. The events client POSTs to /api/v1/authTokens in order to provision a new authorization token; for instance the request might look something like:

```
POST /api/v1/authTokens HTTP/1.1\r\n
Origin: http://xx.xxx.xxx.xxx:8080\r\n
Content-length: 0\r\n
Host: xx.xxx.xxx.xxx:8080\r\n
Accept: */*\r\n
Connection: keep-alive\r\n
Authorization: Basic Ym9iOmJ1aWxkZXI=\r\n
\r\n
```

2. The successful response from the Meeting Server, assuming that " authorization" relates to a user account with sufficient privilege, will be of the form:

```
HTTP/1.1 200 OK\r\n
X-Cisco-CMS-Auth-Token: 7174c102-61b3-47a6-8ff2-b86256cca958\r\n
Connection: close\r\n
\r\n
```

   The returned " X-Cisco-CMS-Auth-Token" is used in the next stage, the WebSocket connection itself

3. The client makes another HTTP request to set up the WebSocket connection, of the form:

```
GET /events/v1?authToken=7174c102-61b3-47a6-8ff2-b86256cca958 HTTP/1.1\r\n/
Host: xx.xxx.xxx.xxx\r\n/
Connection: Upgrade\r\n/
Pragma: no-cache\r\n/
Cache-Control: no-cache\r\n/
Upgrade: websocket\r\n/
Origin: http://xx.xxx.xxx.xxx:8080\r\n/
Sec-WebSocket-Version: 13\r\n/
Accept-Encoding: gzip, deflate\r\n
Accept-Language: en-GB,en-US;q=0.8,en;q=0.6\r\n
Sec-WebSocket-Key: lGaahHe/KdA9lPdPxAIZfw==\r\n
Sec-WebSocket-Extensions: permessage-deflate; client_max_window_bits\r\n
\r\n
```

Note: The X-Cisco-CMS-Auth-Token value should be sent to the Meeting Server as an " authToken" URI parameter. The Sec-WebSocket-Key value (and follow up Sec-WebSocket-Accept) values are as per RFC6455.

4. Assuming that the Meeting Server accepts the WebSocket connection (for instance, the authToken is valid) the success response will look like:

```
HTTP/1.1 101 Switching Protocols\r\n
Upgrade: websocket\r\n
Connection: upgrade\r\n
Sec-WebSocket-Accept: ZISmDfOsp675RM7TQKa0LbQKCqk=\r\n
\r\n
```

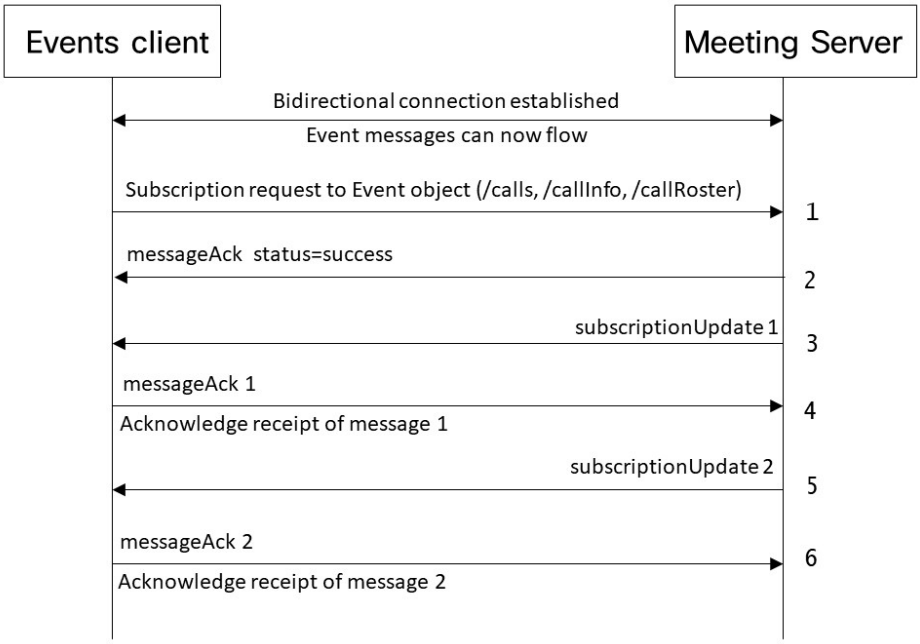
5. The bi-directional WebSocket connection is " ready to go" and the event messages can start to flow.

2.14.5 Example message flows

Once the WebSocket between the events client and the Meeting Server has been established, the events client can subscribe to the Meeting Server for updates on event types.

Figure 12 illustrates the call flow between the events client and the Meeting Server over the web socket connection. The subscriptions from the events client should be in the form of JSON files. For more detail on the call flow, see the explanations and examples following Figure 12.

Figure 12: Overview of event messages flowing between the events client and the Meeting Server



1. Initial subscription from the events client to the Meeting Server, requesting to subscribe to an event type, for example the calls list (active conference list).

```
{
     "type": "message",
     "message": {
          "messageId": 8,
          "type": "subscribeRequest",
          "subscriptions": [
               {
                    "index": 3,
                    "type": "calls",
                    "elements": [
                    "name",
                    "participants"
                    ]
               }
          ]
     }
}
```

   In the above example the client is sending the subscribe request as its message with index 8, and is supplying a single resource to subscribe to the set (array) called subscriptions. It has used the index (tag) " 3" for this subscription, and its type is " calls" , which refers to the

active calls list. Within this subscription, it has specified the set of elements in which it is interested, " name" (the conference name) and " participants" (the number of active participants). This set of elements has 2 main implications:

- it determines which changes trigger a Meeting Server → client update (in this case if the name or the participant count changes an update will be sent),
- it determines which elements are included in the Meeting Server → client update (in this case the record that will be sent to the client will include just the name and the participant count).

2. The Meeting Server responds with an ACK to the subscribe request:

```
{
     "type": "messageAck",
     "messageAck": {
          "messageId": 8,
          "status": "success"
     }
}
```

This ACK has a " messageId" of 8 to indicate to the client which message is being acknowledged, and a " status" code to show if the message has been successfully understood and acted upon.

3. The Meeting Server also keeps the requesting client updated as to the status of its active subscriptions. This is in the form of a subscriptionUpdate sent to the client, for instance:

```
{
     "type": "message",
     "message":
          "messageId": 1,
          "type": "subscriptionUpdate",
          "subscriptions": [
               {
                    "index": 3,
                    "state": "pending"
               }
          ]
     }
}
```

This tells the client that its subscription with index " 3" is pending – it is in the process of being set up but is not yet active. The Meeting Server has tagged this message with a messageId of 1, and the client should ACK this message in order for the Meeting Server to send later updates.

4. The corresponding messageAck from the client to the Meeting Server looks like:

```
{
     "type": "messageAck",
     "messageAck": {
          "messageId": 1,
          "status": "success"
     }
}
```

The Meeting Server and the client use their own individual number spaces for messageId values, which are not required to be distinct; neither are they required to be sequential.

5. Once the Meeting Server has successfully set up the requested subscription, it may send a further subscriptionUpdate, for example:

```
{
     "type": "message",
     "message": {
          "messageId": 2,
          "type": "subscriptionUpdate",
          "subscriptions": [
               {
                    "index": 3,
                    "state": "active"
               }
          ]
     }
}
```

This tells the client that the subscription (with index 3) is now active - the client can now expect to receive updates specific to that subscription. As before, this message from the Meeting Server needs to be acknowledged by the client.

6. Client acknowledges the subscription update for messageId 2. The client supplies messageId 2 in the ack, as that is the value used by the Meeting Server in the subscriptionUpdate.

```
{
     "type": "messageAck",
          "messageAck": {
          "messageId": 2,
          "status": "success"
     }
}
```

7. At some point after the client has subscribed to the calls list, the Meeting Server sends information on a conference just starting:

```
{
     "type": "message",
```

```
    "message": {
        "messageId": 3,
        "type": "callListUpdate",
        "subscriptionIndex": 3,
        "updates": [
            {
                "call": "97c771ae-fc2e-4257-b129-30ee818e034b",
                "updateType": "add",
                "name": "Andy's coSpace",
                "participants": 0
            }
        ]
    }
}
```

As before, this message, messageId 3, needs to be acked by the client.

Note: To avoid duplication, messageAcks are no longer shown in this example.

This update is tagged with subscriptionIndex " 3", indicating to the client which of its subscriptions the update refers to. The update includes a " type" to aid parsing of the data. The " updates" array contains the new information that the Meeting Server is supplying, in this example the " updateType" is an " add" which means that this is the first notification of this conference, and the conference in question has its GUID supplied in the " call" field. The " call" and " updateType" values appear in all updates, but the remaining fields are determined by the " elements" value supplied in the subscription request from the client. If the no " elements" node is supplied in the subscribe message (or it is empty) then no additional fields will be included. However, as " name" and " participants" were included in the example then those are present in the update.

8. After the initial " add" message for this conference, a participant joins the conference, and so a further update is received from the Meeting Server:

```
{
    "type": "message",
    "message": {
        "messageId": 4,
        "type": "callListUpdate",
        "subscriptionIndex": 3,
        "updates": [
            {
                "call": "97c771ae-fc2e-4257-b129-30ee818e034b",
                "updateType": "update",
                "participants": 1
            }
```

```
            ]
        }
}
```

The participants count is now " 1" , and the updateType is now " update" to show that this isn't the first message for the call in question (97c771ae-fc2e-4257-b129-30ee818e034b), but an update to the previous notification. As the " name" value for this call hasn't changed it isn't included in this update.

9. Once a client application has learnt about the presence of an active call in which it is interested (either via the events mechanism or through an API query or " callStart" CDR) the client application may then subscribe to resources specific to that call. For instance, it may re-configure its subscription with the Meeting Server with a new message such as:

```
{
    "type": "message",
    "message": {
        "messageId": 9,
        "type": "subscribeRequest",
        "subscriptions": [
            {
                "index": 1,
                "type": "callRoster",
                "call": "97c771ae-fc2e-4257-b129-30ee818e034b",
                "elements": [
                    "name",
                    "uri",
                    "state",
                    "importance"
                ]
            },
            {
                "index": 2,
                "type": "callInfo",
                "call": "97c771ae-fc2e-4257-b129-30ee818e034b",
                "elements": [
                    "name",
                    "participants",
                    "streaming"
                ]
            },
            {
                "index": 3,
                "type": "calls",
                "elements": [
```

```
                    "name",
                    "participants"
                ]
            }
        ]
    }
}
```

The client remains subscribed to " calls" (the active conference list) by keeping a subscription with the same " index" 3, in its set of " subscriptions" . However, it has now added subscriptions to " callRoster" and " callInfo" to the set. For these subscriptions, a specific " call" GUID needs to be supplied, in this case " 97c771ae-fc2e-4257-b129-30ee818e034b" , which the client was notified of in an earlier " callListUpdate" message from the Meeting Server.

2.15 Using DTMF sequences in clustered Call Bridge deployments

Note: This feature does not support sending DTMF sequences to Lync participants in dual homed conferences.

Prior to version 2.4, DTMF sequences could only be configured for call legs on a local Call Bridge. DTMF sequences could not be configured via the participant API, which meant that DTMF could not be sent to participants on a remote Call Bridge, or when calling out from a cluster of Call Bridges.

From version 2.4, DTMF sequences can be configured for participants. This enables DTMF sequences to be sent to any participant in the conference regardless of which Call Bridge they are connected to. Similarly, DTMF can now be sent when calling out from a cluster of Call Bridges using the participants API to call out. This applies to cases where the Call Bridge for the outbound call is either implicitly or explicitly chosen via load balancing, dial plan rules, or selection of Call Bridge Group or Call Bridge.

- To send DTMF key sequences to the far end in clustered and load balanced deployments:

  POST to **`/calls/<call id>/participants`** with the **`dtmfSequence`** parameter containing a string composed of: digits 0 to 9, # and a “,” which adds a pause between digits.

  This will send the DTMF sequence to the far end when a participant is initially created or during the call.

- To set a DTMF sequence to get played to a specific participant already in a call:

  PUT to **`/participants/<participant id>`** the parameter **`dtmfSequence`** .

2.16 XMPP server certificate validation

From version 2.4.0, the Call Bridge and Web Bridge have trust stores to hold the certificates for the XMPP servers in the deployment. If configured, these trust stores enable the Call Bridge and Web Bridge to check the identity of the XMPP servers when making connections to them. Validating the certificate files of the XMPP servers ensures that XMPP servers are legitimate, and removes the risk that an attacker could redirect traffic to an insecure XMPP server. In addition, validation can be used to prevent the WebRTC app from being used to connect to meetings hosted by other Meeting Server deployments.

Note: By default, neither the Web Bridge nor the Call Bridge validate the certificate files of the XMPP servers in a deployment. If you choose to use this feature then we recommend that you configure validation by both the Web Bridge and the Call Bridge, see below.

CAUTION: If you enable XMPP server validation, but have added a certificate bundle to the Web Bridge and Call Bridge trust stores that is not properly configured, then participants using Cisco Meeting App will be unable to join the Meeting Server meetings.

2.16.1 Single XMPP server deployment

If you are using a single XMPP server, .

1. Use the MMP command **xmpp** or **xmpp status** to identify the name of the XMPP server's certificate file:

   ```
   CMS-SERVER> xmpp
   Enabled : true
   Clustered : false
   Domain : cms.example.com
   Listening interfaces : a
   Key file : private.key
   Certificate file : xmpp-certificate.crt
   Max sessions per user : unlimited
   STATUS : XMPP server running
   ```

   Note: To avoid certificate errors, ensure that the XMPP server certificate specifies:
   - the XMPP server's domain name (FQDN) in the subjectAltName field. DO NOT use an IP address in the subjectAltName field.

2. Disable the Web Bridge before uploading the certificate bundle.

   ```
   webbridge disable
   ```

3. Use the MMP command **webbridge trust xmpp <xmpp certificate bundle>** to upload the XMPP server's certificate file to the Web Bridge trust store.

   ```
   webbridge trust xmpp xmpp-certificate.crt
   ```

4. Re-enable the Web Bridge.

   webbridge enable

5. Use the MMP command **callbridge trust xmpp <xmpp certificate bundle>** to upload the XMPP server's certificate file to the Call Bridge trust store.

   callbridge trust xmpp xmpp-certificate.crt

   Note: It is not mandatory to upload the XMPP server certificate file to both the Web Bridge and the Call Bridge, however it is advised for maximum security.

After uploading the certificate file of the XMPP server to the trust stores of the Web Bridge and the Call Bridge, both the Web Bridge and the Call Bridge will validate the certificate received from the XMPP server to ensure that they only establish a connection to the XMPP server whose certificate they hold in their trust store.

2.16.2 Resilient XMPP server deployment

If you have deployed more than one XMPP server for failover and resiliency, then you will have already created a bundle of XMPP server certificates; see section "Example of deploying XMPP resiliency" in the Scalability and Resilient Deployment guide. If you choose to configure XMPP server validation then the certificate bundle needs to be copied onto every Meeting Server running either the Web Bridge or Call Bridge.

1. Use the MMP command **xmpp cluster status** to identify the name of the trust bundle holding the certificates for the cluster of XMPP servers:

```
CMS-SERVER> xmpp cluster status
Last state change: 2018-Aug-17 13:42:12
Key file : private.key
Certificate file : xmpp-certificate.crt
Trust bundle : cmstrust.crt
```

Note about split Meeting Server deployments: In a split deployment, the XMPP server is located on the core server with the XMPP Load Balancer on an edge server. The XMPP Load Balancer acts as a gateway for the current XMPP leader at that time. When the Web Bridge connects to the XMPP Load Balancer, the Web Bridge needs to receive the certificate of the XMPP server that resides on the core server, rather than the certificate of the edge server. Consequently, the trust bundle used in **`webbridge trust xmpp <xmpp certificate bundle>`** must contain the certificates of the core XMPP server(s).

```
EDGE-0> webbridge
Enabled                    : true
Interface whitelist        : a:443
Key file                   : private.key
Certificate file           : edge-0cms.crt
CA Bundle file             : CA.crt
XMPP Trust bundle          : cmscluster.crt
HTTP Trust bundle          : cmscluster.crt
HTTP redirect              : Enabled
HTTP URL redirect          : true
.....
```

Note: To avoid certificate errors, ensure each XMPP server certificate specifies:
- the XMPP server's domain name in the subjectAltName field.

2. Use the MMP command **`webbridge trust xmpp <xmpp cluster certificate bundle>`** to upload to the Web Bridge trust store the bundle of certificates for the cluster of XMPP servers.

   `webbridge trust xmpp cmstrust.crt`

3. Use the MMP command **`callbridge trust xmpp <xmpp cluster certificate bundle>`** to upload to the Call Bridge trust store the bundle of certificates for the cluster of XMPP servers.

   `callbridge trust xmpp cmstrust.crt`

2.16.3 Removing certificate validation

To stop the Web Bridge validating the XMPP server's certificate, use the MMP command:

`webbridge trust xmpp none`

To stop the Call Bridge validating the XMPP server's certificate, use the MMP command:

`callbridge trust xmpp none`

Note: Currently, if the trust stores are not used then the Web Bridge and the Call Bridge will continue to make connections to the XMPP server(s) without verifying the server(s) identity.

2.17 Call Bridge cluster validation

From version 2.4.0, you can improve the security of a Call Bridge cluster by using the Call Bridge trust store to validate Call Bridges within the cluster. As Call Bridges connect to each other over HTTPS, which is fronted by the Web Admin, you need to create a certificate bundle holding the Web Admin certificates of the clustered Call Bridges, and upload the certificate bundle to the trust store of each Call Bridge in the cluster. Use the MMP command:

```
callbridge trust cluster <bundle name>
```

When a Call Bridge connects to another Call Bridge in a cluster, it checks the whitelist of certificates in its trust store to validate the identity of the Call Bridge that it is connecting to. This removes the risk that the Call Bridge is connecting to an insecure Meeting Server.

If the trust store is not used, then there will be no certificate validation between clustered Call Bridges, and a Call Bridge will continue to make the connection to a remote Call Bridge, but without verifying its identity.

To remove the Call Bridge cluster certificate whitelist from the Call Bridge trust store, use the MMP command:

```
callbridge trust cluster none
```

2.18 More video streams over distribution links between clustered Call Bridges (preview feature)

Note: This remains a beta feature.

Prior to version 2.3, video from a maximum of four remote participants could be sent over each distribution link between clustered Call Bridges. From version 2.3, the Meeting Server supports up to nine video streams over the distribution links. Participants using single, dual and three screen endpoint systems can now have a more consistent conference experience whether conferences are hosted on clustered Call Bridges or on a single Call Bridge.

To configure the maximum number of video streams sent over each distribution link between clustered Call Bridges, set the **`maxPeerVideoStreams`** parameter on API object **`/system/configuration/cluster`** to a value of 1, 4 or 9; the parameter defaults to 4 if not set.

Note: The API parameter **`maxPeerVideoStreams`** parameter can take any value between 1 and 9. However, the screen resolution sent is optimized for 1, 4 or 9, so if you set the variable to 2, 3, 5, 6, 7 or 8 then not all of the screen will be used. For example, if set to " 5" then each of the 5 participants will be 1/9th of the screen, similarly if set to " 2" then the two participants will be 1/4 of the screen.

To support more than four video streams across a distribution link, it is recommended that the bandwidth of the link be set to greater than 2Mbps. Use the API or the Web Admin Interface to

set the bandwidth. If using the API, PUT a value for the **peerLinkBitRate** parameter to the API object **/system/configuration/cluster**; the value will be the maximum media bit rate to use on distribution links between Call Bridges in the cluster. Alternatively, using the Web Admin Interface, go to **Configuration>Cluster>Call Bridge identity** and enter the **Peer link bit rate**.

2.19 Summary of MMP changes

Version 2.4 supports these additional MMP commands:

Command	Description
callbridge trust cluster <trusted cluster certificate bundle>	Configures the Call Bridge to use a particular whitelist of certificates to validate the identity of the Call Bridges in the cluster. (From version 2.4).
callbridge trust cluster none	Removes the certificate whitelist for the Call Bridge cluster from the Call Bridge trust store. (From version 2.4).
callbridge trust xmpp <trusted xmpp certificate whitelist>	Configures the Call Bridge to use a particular whitelist of certificates to validate the identity of the XMPP servers. (From version 2.4).
callbridge trust xmpp none	Removes the XMPP certificate whitelist from the Call Bridge trust store. (From version 2.4)
webbridge trust xmpp <trusted xmpp certificate whitelist>	Configures the Web Bridge to use a particular whitelist of certificates to validate the identity of the XMPP servers. (From version 2.4)
webbridge trust xmpp none	Removes the XMPP certificate whitelist from the Web Bridge trust store. (From version 2.4)
recorder resolution <audio|720p|1080p>	Sets the resolution that the recorder will record meetings. The default is 720p30. (From version 2.4)

The Uploader component previously introduced as a beta feature in version 2.3, is now fully released in version 2.4.0.

Commands	Description
uploader (enable|disable)	Enables or disables the uploader component. Before configuring the Uploader, ensure the component is disabled.
uploader nfs <hostname/IP>:<directory>	Specify the NFS that the Uploader will monitor.

Commands	Description
uploader (cms|rev) host <host-name>	Configure the Uploader with the name of the host for the Meeting Server (cms) and the host for the Vbrick Rev server. Default port is 443.
uploader (cms|rev) port <port>	Configure the Uploader with the port to use to connect to the Meeting Server (cms) and the port for the Vbrick Rev server. Default port is 443.
uploader (cms|rev) user <user-name>	Configure the Uploader with the user that has access to the API of the Meeting Server and the user with access to the Vbrick Rev server.
uploader (cms|rev) password	Configure the Uploader with the password for the specified Meeting Server user and the Vbrick Rev user.
uploader (cms|rev) trust (<crt-bundle>|none)	Upload the specified certificate bundle to the trust store on the Meeting Server or the Vbrick Rev server. **none** removes the certificate bundle from the specified trust store. Note: the Uploader will not work without a certificate bundle in the Meeting Server trust store and the Vbrick Rev trust store.
uploader edit (<uploader-team name>|none)	Not supported in version 2.4.0.
uploader view (<uploader-team name>|none)	Not supported in version 2.4.0.
uploader access <Private|Public|AllUsers>	Set access permission to the video recordings
uploader cospace_member_access <view|edit|none>	Allows members of the space to view or edit the video recordings.**none** removes view or edit permissions for members of the space.
uploader recording_owned_by_cospace_owner <true|false>	**true** selects the owner of the space as the single owner of these video recordings.
uploader fallback_owner (<user-name>|none)	Use the named user as the fallback owner of the video recordings, if the owner of the space is not listed in VbrickRev.**none** removes the fallback owner.
uploader comments (enable|disable)	Enables or disables commenting on video recordings. Default is disabled.
uploader ratings (enable|disable)	Enables or disables video recording ratings. Default is disabled.
uploader downloads (enable|disable)	Sets the download permission, enables or disables downloading the video recordings.
uploader initial_state (<active|inactive>)	Set the initial state of the video recording when first uploaded to Vbrick Rev. Default is active.

Commands	Description
`uploader delete_after_upload (<true|false>)`	Selects whether to delete the video recording from the NFS after upload is complete. Default is false.
`uploader debug (<true|false>)`	Set to true to log, via syslog, additional debugging information for the Uploader.

2.20 Summary of API Additions & Changes

New API functionality for the Meeting Server 2.4 includes:

- setting the highest importance value for pane placement on endpoints connecting to the Meeting Server,
- setting the importance value on an AccessMethod for a coSpace,
- using load balancing in Expressway deployments,
- recorder improvements, including the ability to determine whether a conference is being recorded externally by a Skype or Lync client,
- determining whether the audience was muted by a Skype or Lync client,
- identifying whether a Lync participant is a presenter or an attendee,
- retrieving the Call Bridge media mode,
- using DTMF sequences in clustered Call Bridge deployments,
- setting maximum number of video streams over a distribution link,
- the ability to ensure the member must configure non-member access and set a passcode for every space created as part of the LDAP sync,
- overriding display name labels and setting to a specific name,
- setting a unique identifier for each Call Bridge,
- ability to reduce parts of the H.264 Constrained High Profile video codec from being applied to outgoing calls to SIP endpoints,
- bulk operation on participants in a specific call.

In addition, there is the following change to licenses:

- a license is no longer required to apply single or multiple brands to your Meeting Servers. This includes no longer needing a licence to use `/callBrandingProfiles`.

2.20.1 Setting the highest importance value for pane placement on endpoints connecting to the Meeting Server

To set the highest importance value for pane placement for all spaces:

- POST to **/coSpaces** the request parameter **panePlacementHighestImportance** set to the chosen value.

To set the highest importance value for pane placement for a particular space:

- PUT to **/coSpaces/<coSpace id>** the request parameter **panePlacementHighestImportance** set to the chosen value.

To retrieve the **panePlacementHighestImportance** value for a space, use GET on **/coSpaces/<coSpace id>**.

2.20.2 Setting the importance value on an Access Method for a specific coSpace

To set the importance value for a new Access Method for a specific coSpace:

- POST to **/coSpaces/<coSpace id>/accessMethods** the request parameter **importance** set to the chosen value.

To update an existing Access Method for a specific coSpace:

- PUT to **/coSpaces/<coSpace id>/accessMethods/<access method id>** the request parameter **importance** set to the chosen value.

To retrieve the **importance** value for an Access Method, use GET on **/coSpaces/<coSpace id>/accessMethods/<access method id>**.

2.20.3 Using load balancing in Expressway deployments

To enable load balancing across Call Bridge Groups in deployments with Expressway:

- Either POST to **/callBridgeGroups** or PUT to **/callBridgeGroups/<call bridge group id>** the request parameter **loadBalanceIndirectCalls** set to **true**.

By default **loadBalanceIndirectCalls** is set to **false**.

Note: **loadBalanceIndirectCalls** set to **true**, load balances incoming calls that have Record-Route SIP headers.

- To retrieve the **loadBalanceIndirectCalls** setting, use GET on**/callBridgeGroups/<call bridge group id>**.

2.20.4 Determining whether a conference is being recorded externally

If the **endpointRecording** parameter is set to true for a call, then one of the call's participants is recording the conference externally.

- To retrieve the **endpointRecording** setting for a call, use GET on**/calls/<call id>**.

Note: Currently, a response value of **true** only indicates that a Skype or Lync client is recording the conference.

2.20.5 Determining whether the audience was muted by a Skype or Lync client

If the **lyncAudienceMute** parameter is set to true, then the audience was muted by a Skype or Lync client. Only present if this call is a Skype/Lync conference.

To retrieve the **lyncAudienceMute** setting for a call, use GET on **/calls/<call id>**

2.20.6 Identifying whether a Lync participant is a presenter or an attendee

To retrieve the **lyncRole** setting, use GET on **/callLegs/<call leg id>**. Only present if the participant associated with this call leg is in a Lync conference.

presenter - The participant associated with this call leg is a presenter in the Lync conference.
attendee - The participant associated with this call leg is an attendee in the Lync conference.

2.20.7 Retrieving the Call Bridge media mode

The media activation key determines whether the Call Bridge will encrypt media. To determine whether an unencrypted media license or an encrypted media license has been applied to the Meeting Server via an uploaded media activation key:

- Use GET on **/system/licensing**. The **features** response value will contain either the new **callBridgeNoEncryption** parameter or the original **callBridge** parameter.
  Both parameters, **callBridge** and **callBridgeNoEncryption**, have a **status** value of **noLicense**, **activated**, **grace** or **expired**, and an **expiry** in the form of a string.

nolicense indicates that a media activation key has not been uploaded and the Call Bridge cannot make any calls.

2.20.8 Using DTMF sequences in clustered Call Bridge deployments

To send to the far end a sequence of DTMF key press commands when a participant is initially created or during the call:

- POST to **/calls/<call id>/participants** the request parameter **dtmfSequence** set to a sequence of DTMF key sequences to send to the far end either when the participant is initially created or during the call. The DTMF sequence is played out from the Call Bridge where the call for this participant is placed. In the supplied sequence, you can use the digits 0 to 9, * and #, as well as one or more comma characters (" ,") which add a pause between digits.

To set a DTMF sequence to get played to a specific participant already in a call:

- PUT to **/participants/<participant id>** the request parameter **dtmfSequence** set to a string of DTMF key sequences to be played to this participant.

2.20.9 Setting the maximum number of video streams over a distribution link

To set the maximum number of video streams over a distribution link:

- PUT to **/system/configuration/cluster** the request parameter **maxPeerVideoStreams** set to the maximum number of video streams over a distribution link.
- To retrieve the **maxPeerVideoStreams** setting, use GET on **/system/configuration/cluster**.

Note: For a consistent meeting experience, **maxPeerVideoStreams** needs to be set to the same value on all of the Call Bridges in the cluster.

2.20.10 Creating spaces with nonMemberAccess set to false

When spaces are auto-generated via an LDAP sync, they are all created without a passcode. By default **nonMemberAccess** is set to **true** so that the existing behavior remains unchanged, no passcode is required to access the space and non-members are able to access the created spaces.

Setting **nonMemberAccess** to **false** allows a company to enforce passcode protection for non-member access to all user spaces.

To ensure the member must configure non-member access and set a passcode as part of the LDAP sync:

- Either POST to **/ldapSources** or PUT to **/ldapSources/<ldap source id>** the request parameter **nonMemberAccess** set to **false**. Existing spaces are unaffected.
- To retrieve the **nonMemberAccess** setting, use GET on **/ldapSources/<ldap source id>**.

2.20.11 Overriding display name labels and setting to a specific name

Set a personal name for a participant when they are in a meeting room. The name is also set on cascaded calls. It changes the name of the participant in the following:

- on-screen name label viewed by other conference participants,
- ActiveControl roster list,
- any place that the Meeting App sees the name of the participant in a call,
- CDR records,
- where the name appears in the web interface.

To create a new name label:

- POST to **/calls/<call id>/participants** when the participant is created, or during the call via PUT to **/participants/<participant id>**, the request parameter **nameLabelOverride** set to the chosen name string (maximum of 50 bytes of UTF-8) to override the current name label of this participant.

It can also be supplied via:

- POST to **/calls/<call id>/callLegs** when the call leg is created, or later during the call via a PUT to **/callLegs/<call leg id>**, the request parameter **nameLabelOverride** set to the chosen name string (maximum of 50 bytes of UTF-8) to override the name for this call leg.

Setting an empty string clears the value and restores the original name.

Note: Overriding the name of a participant and its associated call leg(s) is interchangeable and affects both, the latest change takes presedence.

To retrieve **nameLabelOverride** set for a participant, use GET on **/participants/<participant id>**, the assigned string will be returned in the response for **configuration**.

To retrieve **nameLabelOverride** set for a call leg, use GET on **/callLegs/<call leg id>**, the assigned string will be returned in the response for **configuration**.

2.20.12 Setting a unique identitier for each Call Bridge

If communicating with more than one Call Bridge, setting a unique identifier for each Call Bridge with the parameter **hostId** under **api/v1/system/status** allows you to identify which Call Bridge you are talking to.

2.20.13 Refinement in using H.264 Constrained High Profile

Some older third party endpoints do not fully support the H.264 Constrained High Profile video codec. Version 2.4 of the Meeting Server allows some control over only using parts of the codec.

Set **h264CHPMode** on API object **/compatibilityProfile** to either:

auto appropriate parts of H.264 Constrained High Profile are used based on endpoint identification. This is the default behaviour.

basic only uses a minimal subset of parts of H.264 Constrained High Profile.

POST the selected **h264CHPMode** setting to **/compatibilityProfiles** to create a new compatibility profile which will be applied to all outgoing calls to SIP endpoints.

PUT the selected **h264CHPMode** setting to **/compatibilityProfiles/<compatibility profile id>** to modify an existing compatibility profile.

To retrieve the **h264CHPMode** setting for a Compatibility Profile, use GET on **/compatibilityProfile/<compatibility profile id>**.

2.20.14 Bulk operation on participants

From version 2.4, bulk operation of an action can be performed on all of the participants in a call using filters and a mode on the API object **/calls/<call id>/participants/***.

PUT to **/calls/<call id>/participants/*?filterIds=<id1>,<id2>&mode=(exclude|selected)**

Mode	filterIds	Notes
exclude	empty (no ids)	This is the default settings, equivalent to the behavior prior to version 2.4. The operation will act on all of the participants in the selected call.
exclude	one or more ids	The operation will act on all of the participants in the selected call, except those listed.
selected	empty (no ids)	This will have no impact, as no ids are suppllied.
selected	one or more ids	The operation will only act on the selected participants in the call.

The maximum size of a list is fixed at 20, trying to include more ids than this will generate an error.

Return values are for acceptance of operation, failure or success of individual participants will not be returned.

For example:

PUT to **/calls/<call id>/participants/*?filterIds=<smith>,<green>&mode=exclude**

will have the effect that all participants that match filter ids of smith or green will be excluded from the bulk operation.

Errors:

callDoesNotExist call ID does not exist,

If more than 20 filterIds are included in the filter id list, then. a parameterError is generated with error attribute equal to " valueTooLong" .

2.20.15 New interactive API reference tool

We recently introduced a new interactive API reference tool enabling you to see a high level view of the API objects and drill down to lower levels for the detail. There are also learning labs to help you get started, these will be added to over time. We encourage you to try out this tool; sometime in the future we will discontinue publishing the pdf version of the API Reference Guide.

https://developer.cisco.com/cisco-meeting-server/

Steps to use the tool:

1. Click **View the docs**

2. Select a category from the list in the left pane. For example: Call Related Methods.

3. Click on any method to see URI: GET/POST/PUT. Refer to the table of parameters and response elements with descriptions. For example: GET https://ciscocms.docs.apiary.io/api/v1/calls?

Note: If you are using a POST/PUT methods, the related 'Attributes' with descriptions appear on the right-hand pane when you select the method.

Learning labs

https://learninglabs.cisco.com/modules/cisco-meeting-server

The learning labs are intended as a starting point, covering a broad cross-section of what is possible with the Cisco Meeting Server API. Every learning lab is a step-by-step tutorial which takes you through the steps to complete the task from start to finish.

Example: The 'Setting up host and guest access with Cisco Meeting Server API' provides instructions to configure ways in which users can join meetings in a space with different options.

2.21 Summary of CDR Changes

Version 2.4 has the following addition to the Call Detail Records of the Meeting Server:

- new parameter **`endpointRecorded`** in the **`callEnd`** record, indicates whether or not the call was recorded by an endpoint such as a Lync client.

3 Upgrading, downgrading and deploying Cisco Meeting Server software version 2.4

This section assumes that you are upgrading from Cisco Meeting Server software version 2.3. If you are upgrading from an earlier version, then Cisco recommends that you upgrade to 2.3 first following the instructions in the 2.3.x release notes, before following any instructions in these Cisco Meeting Server 2.4 Release Notes. This is particularly important if you have a Cisco Expressway connected to the Meeting Server.

Note: Cisco has not tested upgrading from a software release earlier than 2.3.

To check which version of Cisco Meeting Server software is installed on a Cisco Meeting Server 2000, Cisco Meeting Server 1000, or previously configured VM deployment, use the MMP command **`version`**.

If you are configuring a VM for the first time then follow the instructions in the Cisco Meeting Server Installation Guide for Virtualized Deployments.

3.1 Upgrading to Release 2.4

The instructions in this section apply to Meeting Server deployments that are not clustered. For deployments with clustered databases read the instructions in this FAQ, before upgrading clustered servers.

CAUTION: Before upgrading or downgrading Meeting Server you must take a configuration backup using the **`backup snapshot <filename>`** command and save the backup file safely on a different device. See the MMP Command Reference document for full details. Do **not** rely on the automatic backup file generated by the upgrade/downgrade process as it may be inaccessible in the event of a failed upgrade/downgrade.

CAUTION: If you have a Cisco Expressway connected to the Meeting Server, check that you have run version 2.2.10 or later on your Meeting Server for at least seven days before upgrading to release 2.4. This is required to resolve a cache issue which prevents the Meeting Server WebRTC from working with Cisco Expressway.

Upgrading the firmware is a two-stage process: first, upload the upgraded firmware image; then issue the upgrade command. This restarts the server: the restart process interrupts all active calls running on the server; therefore, this stage should be done at a suitable time so as not to impact users, or users should be warned in advance.

To install the latest firmware on the server follow these steps:

1. Obtain the appropriate upgrade file from the software download pages of the Cisco website:

   Cisco_Meeting_Server_2_4_8_CMS2000.zip

   This file requires unzipping to a single upgrade.img file before uploading to the server. Use this file to upgrade Cisco Meeting Server 2000 servers.

   Hash (SHA-256) for upgrade.img file:
   11f9a5c2d611a70d1a98a51fc297b4b0051129f31f32bfdd6d622661ae1c892d

   Cisco_Meeting_Server_2_4_8_vm-upgrade.zip

   This file requires unzipping to a single upgrade.img file before uploading to the server. Use this file to upgrade a Cisco Meeting Server virtual machine deployment.

   Hash (SHA-256) for upgrade.img file:
   4399a34319da126dd7969076b9978148ec826dc59ec9b8b4f56d0f29f6a6c75c

   Cisco_Meeting_Server_2_4_8_x-series.zip

   This file requires unzipping to a single upgrade.img file before uploading to the server. Use this file to upgrade Acano X-series servers.

   Hash (SHA-256) for upgrade.img file:
   cce2a13e7d81bac8329adeeb1248d359753c48f1e08767e6f1e2ef98050168fa

   Cisco_Meeting_Server_2_4_8.ova

   Use this file to deploy a new virtual machine via VMware.

   For vSphere6, hash (SHA-512) for Cisco_Meeting_Server_2_4_8_vSphere-6_0.ova file:
   a0037eae18aa6b38e2d7ae6e1e2fb5ef1b636347795be40de8996b746f32e76ee928b84b86c5e18f7f287482c9ec969e29c69962868ccb2c15fb756accc7d41d

   For vSphere6.5 and higher, hash (SHA-512) for Cisco_Meeting_Server_2_4_8_vSphere-6_5.ova file:
   68972fae507a0fa8f7c63fd9e0a99dfe732c24cf5e18390a3cb10d1af72f4d6939cf9ac4512f0e5e26cf52a77e8ff17e7f30d806566865567922b76a3e15cab5

2. To validate the OVA file, the checksum for the 2.4.8 release is shown in a pop up box that appears when you hover over the description for the download. In addition, you can check the integrity of the download using the SHA-512 hash value listed above.
3. Using an SFTP client, log into the MMP using its IP address. The login credentials will be the ones set for the MMP admin account. If you are using Windows, we recommend using the WinSCP tool.

   Note: If you are using WinSCP for the file transfer, ensure that the Transfer Settings option is 'binary' not 'text'. Using the incorrect setting results in the transferred file being slightly smaller than the original and this prevents successful upgrade.

Note:
a) You can find the IP address of the MMP's interface with the **`iface a`** MMP command.
b) The SFTP server runs on the standard port, 22.

4. Copy the software to the Server/ virtualized server.
5. To validate the upgrade file, issue the **`upgrade list`** command.
   a. Establish an SSH connection to the MMP and log in.
   b. Output the available upgrade images and their checksums by executing the upgrade list command.

      `upgrade list`
   c. Check that this checksum matches the checksum shown above.
6. To apply the upgrade, use the SSH connection to the MMP from the previous step and initiate the upgrade by executing the **`upgrade`** command.
7. Verify that the Meeting Server is running the upgraded image by re-establishing the SSH connection to the MMP and typing:
   `version`
8. Update the customization archive file when available.
9. If you are deploying a scaled or resilient deployment read the Scalability and Resilience Deployment Guide and plan the rest of your deployment order and configuration.
10. If you have deployed a database cluster, be sure to run the **`database cluster upgrade_schema`** command after upgrading. For instructions on upgrading the database schema refer to the Scalability and Resilience Deployment Guide.
11. You have completed the upgrade.

3.2 Downgrading

If anything unexpected occurs during or after the upgrade process you can return to the previous version of the Meeting Server software. Use the regular upgrade procedure to "downgrade" the Meeting Server to the required version using the MMP **`upgrade`** command.

1. Copy the software to the Server/ virtualized server.
2. To apply the downgrade, use the SSH connection to the MMP and start the downgrade by executing the **`upgrade <filename>`** command.

   The Server/ virtualized server will restart automatically – allow 10-12 minutes for the process to complete and for the Web Admin to be available after downgrading the server.
3. Log in to the Web Admin and go to **Status > General** and verify the new version is showing under **System status**.

4. Use the MMP command **`factory_reset app`** on the server and wait for it to reboot from the factory reset.
5. Restore the configuration backup for the older version, using the MMP command **`backup rollback <name>`** command.

   Note: The **`backup rollback`** command overwrites the existing configuration as well as the license.dat file and all certificates and private keys on the system, and reboots the Meeting Server. Therefore it should be used with caution. Make sure you copy your existing cms.lic file and certificates beforehand because they will be overwritten during the backup rollback process. The .JSON file will not be overwritten and does not need to be re-uploaded.

   The Meeting Server will reboot to apply the backup file.

   For a clustered deployment, repeat steps 1-5 for each node in the cluster.
6. In the case of XMPP clustering, you need to re-cluster XMPP:
   a. Pick one node as the XMPP master, initialize XMPP on this node
   b. Once the XMPP master has been enabled, joining any other XMPP nodes to it.
   c. Providing you restore using the backup file that was created from the same server, the XMPP license files and certificates will match and continue to function.
7. Finally, check that:
   - the Web Admin interface on each Call Bridge can display the list of coSpaces.
   - dial plans are intact,
   - XMPP service is connected
   - no fault conditions are reported on the Web Admin and log files.
   - you can connect using SIP and Cisco Meeting Apps (as well as Web Bridge if that is supported).

   The downgrade of your Meeting Server deployment is now complete.

3.3 Cisco Meeting Server 2.4 Deployments

To simplify explaining how to deploy the Meeting Server, deployments are described in terms of three models: the single combined Meeting Server, the single split Meeting Server and the deployment for scalability and resilience. All three different models may well be used in different parts of a production network.

3.3.1 Deployments using a single host server

If you are deploying the Meeting Server as a single host server (a “combined” deployment), we recommend that you read and follow the documentation in the following order:

1. Appropriate Installation Guide for your Cisco Meeting Server (Cisco Meeting Server 2000, Cisco Meeting Server 1000 and virtualized deployments, or the installation guide for Acano X-Series Server).
2. The Single Combined Meeting Server Deployment Guide enabling all the solution components on the single host. This guide refers to the Certificate Guidelines for Single Combined Server Deployments for details on obtaining and installing certificates for this deployment.

Note: The Cisco Meeting Server 2000 only has the Call Bridge, Web Bridge, XMPP server and database components. It can be deployed as a single server on an internal network, but if a deployment requires firewall traversal support for external Cisco Meeting App clients, then TURN server and Load Balancer edge components need to be deployed on a separate Cisco Meeting Server 1000 or specification-based VM server - see the” single split” deployment below.

3.3.2 Deployments using a single split server hosted on a Core server and an Edge server

If you are deploying the Meeting Server in a split server model, we recommend that you deploy the XMPP server on the Core server, and deploy the Load Balancer on the Edge server.

Read and follow the documentation in the following order:

1. Appropriate Installation Guide for your Cisco Meeting Server
2. The Single Split Meeting Server Deployment Guide. This guide refers to the Certificate Guidelines for Single Split Server Deployments for details on obtaining and installing certificates for this deployment.

3.3.3 Deployments for scalability and resilience

If you are installing the Meeting Server for scalability and resilience using multiple host servers, we recommend that you deploy the XMPP server on Core servers, and deploy Load Balancers on the Edge server.

Read and follow the documentation in the following order:

1. Appropriate Installation Guide for your Cisco Meeting Server
2. The Scalability and Resilience Deployment Guide. This guide refers to the Certificate Guidelines for Scalable and Resilient Server Deployments for details on obtaining and installing certificates for this deployment.

4 Bug search tool, resolved and open issues

You can now use the Cisco Bug Search Tool to find information on open and resolved issues for the Cisco Meeting Server, including descriptions of the problems and available workarounds. The identifiers listed in these release notes will take you directly to a description of each issue.

1. Using a web browser, go to the Bug Search Tool.
2. Sign in with a cisco.com registered username and password.

To look for information about a specific problem mentioned in this document:

1. Enter the bug identifier in the **Search** field and click **Search**.

To look for information when you do not know the identifier:

1. Type the product name in the **Search** field and click **Search**

   or,

   in the **Product** field select **Series/Model** and start typing `Cisco Meeting Server`, then in the **Releases** field select **Fixed in these Releases** and type the releases to search for example `2.4.8`.
2. From the list of bugs that appears, filter the list using the *Modified Date*, *Status*, *Severity*, *Rating* drop down lists.

The Bug Search Tool help pages have further information on using the Bug Search Tool.

4.1 Resolved issues

Issues seen in previous versions that are fixed in 2.4.8

Cisco identifier	Summary
CSCvq19622	This issue has been filed to evaluate the product against the vulnerability released by the Netflix on June 17th affecting FreeBSD and Linux kernels, identified by CVE IDs: - CVE-2019-11477: SACK Panic - CVE-2019-11478: SACK Slowness or Excess Resource Usage - CVE-2019-11479: Excess Resource Consumption Due to Low MSS Values Cisco has reviewed this product and concluded that it is affected by this vulnerability as it contains a vulnerable version of Linux Kernel.

Issues seen in previous versions that are fixed in 2.4.7

Cisco identifier	Summary
CSCvp96569	In rare circumstances, a participant's content stream may not be seen by other conference participants after content is restarted.
CSCvo91844	Degraded audio experienced on a fully loaded Meeting Server with many audio participants.
CSCvp96694	On rare occasions, temporary poor video quality may be seen on immersive systems connected to a Meeting Server under conditions of high load.

Issues seen in previous versions that are fixed in 2.4.6

Cisco identifier	Summary
CSCvp29391	WebRTC calls on Meeting Server using Apple Safari will not work after updating to Apple Safari on iOS 12.3 or later, and Apple Safari 12.2 on macOS and later.
CSCvp37201	WebRTC calls on Meeting Server using Mozilla Firefox will not work after updating to version 68.
CSCvp33496	On WebRTC app, Input Method Editors (IME) not submitting fields correctly on Internet Explorer.
CSCvp38323	Frozen video seen on some Skype for Business participants when Meeting Server sends dual video streams to an AVMCU conference.
CSCvo82633	Occasionally the Recorder does not record for SIP calls when the CallProfile is set to Automatic.
CSCvk22499	In rare circumstances, Meeting Server's Callbridge component may restart unexpectedly when a participant joins a meeting.

Issues seen in previous versions that are fixed in 2.4.5

Cisco identifier	Summary
CSCvp12123	Cisco Meeting Server's Callbridge component may restart unexpectedly when a SIP participant joins a conference whilst Skype for Business content share is in progress.
CSCvp12120	On rare occasions, Media modules may report " presumed lost: no response to ping" or " media modules now not responding" under heavy call load.
CSCvp12118	Frozen video seen on some Skype for Business participants when Cisco Meeting Server sends dual video streams to an AVMCU conference.
CSCvn63172	On rare occasions, temporary poor video quality can be seen on immersive systems (TX9000 or IX5000) connected to a Cisco Meeting Server.

Issues seen in previous versions that are fixed in 2.4.4

Cisco identifier	Summary
CSCvo51143	Support for the WebRTC app using Google Chrome version 73. See New features introduced in 2.4.4 for further information.
CSCvo60648	MMP commands not functioning, resulting in inability to perform a PKI Inspect as well as inability to gather log bundles etc.
CSCvo17329	Cisco Meeting Server can become overloaded if audio-only calls are used. This is due to increased media processing load on distributed peer link calls for coSpaces with audio-only participants.
CSCvo56197	Guest participants joining a Meeting Server conference may be left at the lobby screen even after a host participant has already joined the conference.
CSCvn63372	The security padlock icon disappears indicating the security status changes from Encrypted to Non-Encrypted when a second Encrypted endpoint/Jabber joins a meeting. However, the call is still encrypted despite the lack of the padlock icon.
CSCvm93493	Chat does not work on one webbridge part of a cluster when using the guest WebRTC join path.
CSCvo13844	WebRTC client occasionally fails to login when one server in a cluster is offline/unavailable.
CSCvo11654	When joining a call using the WebRTC app, no video is received after a network interruption occurs.
CSCvo37253	Cisco Meeting Server's Callbridge component may restart unexpectedly in a dual homed meeting when it is unmuted by the Skype for Business host.
CSCvo37254	On very rare occasions when a Media Module restarts it could reconnect twice resulting the module status reporting that the Media Module is presumed lost with no response to ping for long a time.
CSCvo51337	When cross launching from a browser to Meeting App 1.11.13 the browser displays " You are no longer in the meeting" .
CSCvo07951	When using the WebRTC client, no search results are displayed when trying to add a participant by using the directory search (DirectorySearchLocation).
CSCvo11426	On Meeting Server, the Webbridge is spamming the logs " ... Session <GUID> is pending destroy. Performing partial detach... "

Issues seen in previous versions that are fixed in 2.4.3

Cisco identifier	Summary
CSCvn81865	Support for the WebRTC app using Google Chrome version 72. See New features introduced in 2.4.3 for further information.
CSCvo02066	Cisco Meeting App users experience intermittent failures when authenticating with Cisco Meeting Server.

Cisco identifier	Summary
CSCvn46679	Uploader fails with " runtime error: index out of range" after upgrading from version 2.3.4 to 2.4.2.
CSCvn37841	No video/audio received from a remote participant in a distributed call on clustered Cisco Meeting Server 2000.
CSCvn26366	When the uploader is enabled, the session timeout is not extended properly and it may cause login failures from Meeting Server to VBrick Rev.
CSCvn14138	The " media module status" line in the logs is not followed by the usual numbers to indicate the health of the media framework on Cisco Meeting Server 2000.
CSCvn16684	XMPP component disconnects due to invalid-xml causing XMPP users to get logged out of meetings.
CSCvm95156	When running a trunk debug on Meeting Server 2000 it returns an error that the file is not found.
CSCvk67078	When a Jabber for Windows user (in Desk Phone mode) places a call to a Meeting Server, the video quality received is fine, but after a Hold and Resume the receiving Jabber video quality is downgraded to low resolution.
CSCvk67533	When recording a session it stops after 1-3 hours of recording due to a recorder " keepalive failure" .
CSCvj13390	In clustered environments with multiple TURN Servers configured, if one TURN server is becoming unavailable, the webbridge does not correctly detect this in order to fail-over to the other available TURN servers - it continues to advertise the non available one to WebRTC clients.
CSCvn04352	When cross launching Meeting App client through IE browser it connects as " guest" when using the host passcode for a meeting.

Issues seen in previous versions that are fixed in 2.4.2

Cisco identifier	Summary
CSCvn01698	The callLegEnd txVideo maxSizeWidth and maxSizeHeight values are swapped in CDR callLegEnd records.
CSCvm72658	Lync client receives poor resolution 320x180 when a SIP endpoint shares content via peer link.
CSCvm40725	When a Skype client calls to a space on Meeting Server with two participants and the window is re-sized, the receiving video freeze. When the user drops the call and calls again the video then appears fine.
CSCvh58793	On Cisco Meeting App the participant status in the space is shown as active although that participant left.
CSCve08058	Load balancers stop working after a network issue. After disabling and re-enabling the load balancers start working again.

Cisco identifier	Summary
CSCvk12210	The syslog and audit log files on a Cisco Meeting Server 2000 may unexpectedly become truncated below their expected 100Mb file maximum.

Issues seen in previous versions that are fixed in 2.4.1

Cisco identifier	Summary
CSCvm73261	An unexpected restart after call failures can occur.
CSCvk56605	When a presentation is shared from WebRTC on Windows, the negotiated bandwidth for the presentation stream is low. This means that if sharing at 1080p30 the presentation quality seen by other participants is poor.
CSCvk77779	CDR receiver cannot receive CDR messages from Meeting Server.
CSCvm58644	Unable to access WebRTC page to launch Meeting App on Internet Explorer.
CSCvk77776	After a user logs in to the WebRTC app and joins a Space, if they click " add participants" and search by someone's name, and then click the name to dial, it fails.
CSCvk10971	Meeting Server Web Bridge shows a blank screen when a user copies an invitation to a space and then clicks on a previous call on the recent calls list.
CSCvk66053	Cisco Meeting App (WebRTC app) previously showed the duration of the whole meeting in the information panel rather than the duration of the local participant in the meeting. In 2.4.1 the information panel now shows the duration of the local participant in the meeting and not the duration of the whole meeting.
CSCvm56708	The callbridge trust **xmpp <xmpp certificate bundle>** will not work if an IP address is specified in the SAN field of the certificate for the XMPP server, rather than the domain name (FQDN).
CSCvm56706	The Uploader commands **uploader view <team name>** and **uploader edit <team name>** are not supported in version 2.4.0.

Issues seen in previous versions that are fixed in 2.4.0

Cisco identifier	Summary
CSCvh22816	Logging in using the WebRTC app may fail even when correct credentials are supplied. This occurs when a particular cookie string is supplied by the web browser to the Web Bridge. To avoid this happening either open an incognito tab to use the WebRTC app or clear all cookies for the domain used by the Web Bridge, for example for the WebRTC app at https://join.example.com, clear all example.com cookies.
CSCvm56705	Shared video content appears fuzzy on WebRTC app.
CSCvm56703	Meeting Server version number returned in HTTP header.
CSCvk00058	In a business to business deployment where a Cisco Meeting App user authenticates in one deployment but then dials into another deployment, the Cisco Meeting App does not receive the All Equal screen layout assigned to the space.

Cisco identifier	Summary
CSCvj64142	In a Meeting Server cluster, running the GET method on API object /participants and then applying the “callBridgeFilter” with a Call Bridge id returns 0 participants.
CSCvj05192	If the Meeting Server is configured as a Lync gateway and a PIN is used for Meeting Server spaces, the first participant using a SIP endpoint and the first Skype participant joining the same space only see the splash screen, they do not see each other’s video.
CSCvj04915	If Cisco TMS schedules conferences with the Meeting Server using spaces already created, it takes the Meeting Serversometime to unlock the space after being instructed by Cisco TMS to do so.
CSCvh32697	When using the Meeting Server Uploader with VBrick Rev, all videos show as being uploaded by the main API user account, regardless of any fallback owner configured.
CSCvh03814	In a clustered deployment, an excessive number of database connections may cause a database outage.
CSCvg87404	The acanoAccountLock SNMP trap will not be triggered in the correct circumstances. The trap should be sent when an MMP user account becomes locked after repeated login failures.
CSCvg57974	The setting for **`qualityMain`** is lost when calling from one Call Bridge to another Call Bridge in the same cluster, with outbound load balancing enabled. **`qualityMain`** restricts the maximum negotiated main video call quality for a call leg.
CSCvf98054	When spaces are auto-generated via an LDAP sync, they are all created without a passcode. See Section 2.20 in this release note.
CSCvf88625	In some rare cases when Meeting Serversends content in H.263 to SIP endpoints, the RTP packets sequence will be disordered and affect video quality.
CSCvf78579	In some deployments, Web Admin time stamps and cdrTime may be out of sync with time from the MMP. MMP **`date`** and **`timezone`** commands report the time correctly.

4.2 Open issues

The following are known issues in this release. If you require more details enter the Cisco identifier into the Search field of the Bug Search Tool.

Cisco identifier	Summary
CSCvo66473	Microphone selection on Cisco Meeting App for WebRTC doesn't work on Safari on Mac. Use Google Chrome if you need to use microphone selection or disable the extra microphones prior to joining a Meeting App meeting.
CSCvm56734	In a dual homed conference, the video does not restart after the attendee unmutes the video.
CSCvm56730	Using IPv6 a database cluster may fail after upgrading from a previous release.

Cisco identifier	Summary
CSCvj49594	ActiveControl does not work after a hold/resume when a call traverses Cisco Unified Communications Manager and Cisco Expressway.
CSCvh23039	The Uploader component does not work on tenanted recordings held on the NFS.
CSCvh23036	DTLS1.2, which is the default DTLS setting for Meeting Server 2.4, is not supported by Cisco endpoints running CE 9.1.x. ActiveControl will only be established between Meeting Server 2.4 and the endpoints, if DTLS is changed to 1.1 using the MMP command `tls-min-dtls-version 1.0`.
CSCvh23028	Changing the interface that the Web Bridge listens on or receiving a DHCP lease expire, will cause the Web Bridge to restart. WebRTC App users may have to log in again.
CSCvg62497	If the NFS is set or becomes Read Only, then the Uploader component will continuously upload the same video recording to Vbrick. This is a result of the Uploader being unable to mark the file as upload complete. To avoid this, ensure that the NFS has read/write access.
CSCve64225	Cisco UCS Manager for Cisco Meeting Server 2000 should be updated to 3.1(3a) to fix OpenSSL CVE issues.
CSCve60309	Cisco UCS Manager 3.1(3a) reports 'DIMM A1 on server 1/1 has an invalid FRU' as the CMS 2000 DIMMs are not listed in the 3.2(3e)T catalog.
CSCve37087 but related to CSCvd91302	One of the media blades of the Cisco Meeting Server 2000 occasionally fails to boot correctly. Workaround: Reboot the Fabric Interconnect modules.

In addition there is the following limitation:

CAUTION: The maximum number of concurrent XMPP clients supported by the current Meeting Server software is 500.This maximum is a total number of all different clients (Cisco Meeting App, WebRTC Sign-in and WebRTC Guest clients) registered at the same time to clustered Meeting Servers. If the number of concurrent XMPP registrations exceeds 500 sessions, some unexpected problems with sign in may occur or it may lead to a situation where all currently registered users need to re-sign in, this can cause a denial of service when all users try to sign in at the same time.

Cisco Legal Information

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco Trademark