Cisco Meeting Server

Cisco Meeting Server Release 3.0.4

Release Notes

March 10, 2021

Contents

What's changed

Version	Change
March 10, 2021	Fourth maintenance release (3.0.4). Hashes updated. See resolved issues.
January 12, 2020	Third maintenance release (3.0.3). Hashes updated.
January 07, 2020	Third maintenance release (3.0.3). Hashes updated.
December 03, 2020	Second maintenance release (3.0.2). Hashes updated. See resolved issues.
October 06, 2020	Minor correction.
October 01, 2020	First maintenance release (3.0.1). Hashes updated. See resolved issues.
September 16, 2020	Formatting errors corrected.
September 08, 2020	Minor edit to clarify new recorder/streamer can't be used as an external recording/streaming service.
September 02, 2020	Minor edit to clarify VM minimum requirement to 4 vCPU cores.
August 27, 2020	Included previously missing snippets.
August 12, 2020	Edit to web app call capacity figures.
July 29, 2020	First release of 3.0.

1 Introduction

These release notes describe the new features, improvements and changes in release 3.0 of the Cisco Meeting Server software.

The Cisco Meeting Server software can be hosted on:

- Cisco Meeting Server 2000, a UCS 5108 chassis with 8 B200 blades and the Meeting Server software pre-installed as the sole application.
- Cisco Meeting Server 1000, a Cisco UCS server preconfigured with VMware and the Cisco Meeting Server installed as a VM deployment.
- or on a specification-based VM server.

Version 3.0 introduces many component removals on Cisco Meeting Server. For a list of 3.0 changes, see Section 2.1.

Note: Meeting Server 3.0 introduces a mandatory requirement to have Cisco Meeting Management 3.0 (or later). Meeting Management handles the product registration and interaction with your Smart Account (if set up) for Smart Licensing support. For more details, see Section 2.2.

Note about Acano X-Series: Cisco Meeting Server version 3.0 and later does not support the X-Series servers. It's also not supported for older versions of Call Bridge to connect to the latest Meeting Server services such as Web Bridge 3, recorder, streamer available in Meeting Server 3.0.

Note: Cisco Meeting App for WebRTC (Web Bridge 2) is removed from Cisco Meeting Server version 3.0. If using software version 3.0 or later, you will need to use Cisco Meeting Server web app instead of Cisco Meeting App for WebRTC. To do this, you need to deploy Web Bridge 3 – for details on deploying and configuring Web Bridge 3, see the 3.0 or later Deployment Guides.

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 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. However, support for Skype for Business and Office 365 will continue.

1.1 Interoperability with other Cisco products

Interoperability test results for this product are posted to http://www.cisco.com/go/tp-interop, where you can also find interoperability test results for other Cisco conferencing products.

1.2 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.

1.2.1 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
- specification-based VM platforms.

1.2.2 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.

1.2.3 Call capacities

Table 1 provides a comparison of the call capacities across the platforms hosting Cisco Meeting Server software version 3.0.

Table 1: Call capacities across Meeting Server platforms

Type of calls	Cisco Meeting Server 1000 M4	Cisco Meeting Server 1000 M5	Cisco Meeting Server 2000
Full HD calls 1080p60 video 720p30 content	24	24	175
Full HD calls 1080p30 video 1080p30/4K7 content	24	24	175

Type of calls	Cisco Meeting Server 1000 M4	Cisco Meeting Server 1000 M5	Cisco Meeting Server 2000
Full HD calls 1080p30 video 720p30 content	48	48	350
HD calls 720p30 video 720p5 content	96	96	700
SD calls 448p30 video 720p5 content	192	192	1000
Audio calls (G.711)	1700	2200	3000

Table 2 below compares the call capacities for a single or cluster of Meeting Servers compared to load balancing calls within a Call Bridge Group.

Table 2: Meeting Server call capacity for software version 3.0

Cisco Meeting Server platform		Cisco Meeting Server 1000 M4	Cisco Meeting Server 1000 M5	Cisco Meeting Server 2000
Individual Meeting Servers or Meeting Servers in a cluster (notes 1,2 3 and 4)	1080p30 720p30 SD Audio calls	48 96 192 1700	48 96 192 2200	350 700 1000 3000
	HD participants per conference per server	96	96	450
	web app call capacities (internal calling): Full HD HD SD Audio calls	 48 96 192 500	 48 96 192 500	 350 700 1000 1000
Meeting Servers in a Call Bridge Group	Call type supported	Inbound SIP Outbound SIP		
	1080p30 720p30 SD Audio calls Load limit	48 96 192 1700 96,000	48 96 192 2200 96,000	350 700 1000 3000 700,000
	Number of HD participants per conference per server	96	96	450
	web app call capacities (internal calling): Full HD HD SD Audio calls	 48 96 192 500	 48 96 192 500	 350 700 1000 1000

Note 1: Maximum of 24 Call Bridge nodes per cluster; cluster designs of 8 or more nodes need to be approved by Cisco, contact Cisco Support for more information.

Note 2: Clustered Cisco Meeting Server 2000's without Call Bridge Groups configured, support integer multiples of maximum calls, for example integer multiples of 700 HD calls.

Note 3: Up to 16,800 HD concurrent calls per cluster (24 nodes x 700 HD calls) applies to SIP or web app calls.

Note 4: A maximum of 2600 participants per conference per cluster depending on the Meeting Servers platforms within the cluster.

Note 5: Table 2 assumes call rates up to 2.5 Mbps-720p5 content for video calls and G.711 for audio calls. Other codecs and higher content resolution/framerate will reduce capacity. When meetings span multiple call bridges, distribution links are automatically created and also count against a server's call count and capacity. Load limit numbers are for H.264 only.

Note 6: VMware have made changes in their recent versions (6.0 update 3, 6.5 update 2 and 6.7) that has reduced the throughput of audio calls on Cisco Meeting Server version 3.0 (video capacity is unaffected).

Note 7: The call setup rate supported for the cluster is up to 40 calls per second for SIP calls and 20 calls per second for Cisco Meeting Server web app calls.

1.2.4 Cisco Meeting Server web app call capacities

This section details call capacities for deployments using Web Bridge 3 and web app for external and mixed calling. (For internal calling capacities, see Table 2.)

1.2.4.1 Cisco Meeting Server web app call capacities – external calling

External calling is when clients use Cisco Expressway as a reverse proxy and TURN server to reach the Web Bridge and Call Bridge.

When using Expressway to proxy web app calls, the Expressway will impose maximum calls restrictions to your calls as shown in Table 3.

Note: If you are deploying Web Bridge 3 and web app you must use Expressway version X12.6 or later, earlier Expressway versions are not supported by Web Bridge 3.

Table 3: Cisco Meeting Server web app call capacities – external calling

Setup	Call Type	CE1200 Platform	Large OVA Expressway
Cisco Expressway Pair (X12.6 or later)	Full HD	150	150
	Other	200	200

The Expressway capacity can be increased by clustering the Expressway pairs. Expressway pairs clustering is possible up to 6 nodes (where 4 are used for scaling and 2 for redundancy), resulting in a total call capacity of four times the single pair capacity.

Note: The call setup rate for the Expressway cluster should not exceed 6 calls per second for Cisco Meeting Server web app calls.

1.2.4.2 Cisco Meeting Server web app capacities – mixed (internal + external) calling

Both standalone and clustered deployments can support combined internal and external call usage. When supporting a mix of internal and external participants the total web app capacity will follow Table 2 for Internal Calls, but the number of participants within the total that can connect from external is still bound by the limits in Table 3.

For example, a single standalone Meeting Server 2000 with a single Expressway pair supports a mix of 1000 audio-only web app calls but the number of participants that are external is limited to a maximum of 200 of the 1000 total.

1.3 Cisco Meeting Server web app Important information

Note: Cisco Meeting App for WebRTC (Web Bridge 2) is removed from Cisco Meeting Server version 3.0. If using software version 3.0 or later, you will need to use Cisco Meeting Server web app instead of Cisco Meeting App for WebRTC. To do this, you need to deploy Web Bridge 3 – for details on deploying and configuring Web Bridge 3, see the 3.0 or later Deployment Guides.

If you are using Cisco Meeting Server web app (i.e. you have deployed Web Bridge 3), see Cisco Meeting Server web app Important Information for details on when features are released and issues resolved for the web app.

All information relevant to the web app is contained in this separate document and is not included in the Meeting Server release notes.

The Important Information guide describes the following:

- Any new or changed feature in the web app, and details of fixed issues and open issues associated with the web app with an indication of the version of Meeting Server where this feature/fix is available.
- Any upcoming changes in browsers affecting the web app, and the affected versions of the web app with recommended workarounds.

1.4 End of Software Maintenance

On release of Cisco Meeting Server software version 3.0, Cisco announces the time line for the end of software maintenance for the software in Table 4.

Table 4: Time line for End of Software Maintenance for versions of Cisco Meeting Server

Cisco Meeting Server software version	End of Software Maintenance notice period
Cisco Meeting Server version 2.9.x	The last date that Cisco Engineering may release any final software maintenance releases or bug fixes for Cisco Meeting Server version 2.9.x is March 1, 2022.

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

2 New Features/Changes in version 3.0

Version 3.0 of the Meeting Server software, introduces the following new features and changes:

- removal of legacy Meeting Server components and other changes.
- support for Smart Licensing to improve the user experience of license purchasing, registration and software administration.
- security improvements using Image Signing when upgrading Cisco Meeting Server.
- administrator-configurable minimum passcode length for increased security across all methods of dialing into meetings.
- new internal SIP recorder and streamer components to replace the XMPP internal recorder and streamer components. The new recorder and streamer support changing layouts, and the new streamer supports up to 1080p resolution.
- Web Bridge configuration moved to Web Bridge profiles and settings in the API.
- Cisco Meeting Server web app introduces many new features in 3.0 to give feature parity with the now deprecated Cisco Meeting App for WebRTC. For a complete list of the web app features introduced in 3.0, see Cisco Meeting Server 3.0 web app Important Information. Web app features that require Meeting Server-side configuration are listed below:
  - Join a meeting using a video address (URI)
  - change in permissions for web app participants.
  - Web app participants now see name labels in their conference video.
  - web app controls for recording/streaming, lock/unlock a meeting, and Importance.
  - change to the C2W connection to now accept intermediate/end-entity (i.e. non-root) certificates.
  - ability to customize the web app sign-in page with your own branding
- Automatic Gain Control (AGC) is now enabled by default.
- support for ESXi7.0.
- viewable historical record of the assigned number of PMP licenses.

Note about Acano X-Series: Cisco Meeting Server version 3.0 and later does not support the X-Series servers. It's also not supported for older versions of Call Bridge to connect to the latest Meeting Server services such as Web Bridge 3, recorder, streamer available in Meeting Server 3.0.

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.

2.1 Meeting Server components removed and changed in 3.0

From 3.0 the following features and services are no longer available or supported in Meeting Server:

- ACU – ACU customers should migrate to SMP+ licensing. Contact your Cisco reseller for additional information.
- Web Bridge 2 – As Web Bridge 2 is removed in 3.0, Web Bridge 2 users will need to redeploy their Web Bridge to use Web Bridge 3 for web app support. There is no automatic upgrade migration from Web Bridge 2 to Web Bridge 3. If you have already deployed Web Bridge 3 in version 2.9, you should check your settings after upgrade because they will not be migrated across from the Web Admin or old settings in /webBridges/<webbridge id>.
    - Cisco Meeting App for desktop, iOS and WebRTC are no longer supported.
- XMPP – The old XMPP-dependent recorder and streamer are now replaced in 3.0 with new internal SIP Recorder and Streamer components. On upgrading to 3.0 you will need to re-deploy your recorder and streamer.
- H.323 Gateway
- Load balancer
- SIP edge
- Trunk
- X-Series servers

All related MMP commands and API objects and parameters are deprecated or removed. See Section 2.14 and Section 2.16 for specific information.

2.2 Smart Licensing

Version 3.0 introduces support for Smart Licensing on Cisco Meeting Server using Cisco Meeting Management version 3.0 (or later). This transition to the software licensing model, i.e. moving from traditional Product Activation Key (PAK) licenses to Smart Licensing, improves the user experience of license purchasing, registration and software administration. It also aligns Meeting Server with other Cisco products' approach to software licensing and utilizes Cisco Smart Account – a central repository where you can view, store, and manage licenses across your entire organization.

All new license purchases still receive a PAK code – retain for reference – as all licenses will be available in the Smart Account that Meeting Management will sync to.

For further information and to create a Smart Account, go to: https://software.cisco.com and choose Smart Licensing.

Note: The term "overage" is used to describe a situation where license usage is higher than the entitlement.

The Meeting Server licensing changes and behaviors in 3.0 are:

- Cisco Meeting Management version 3.0 (or later) is mandatory in version 3.0 – Meeting Management reads the Meeting Server license file, and can handle the product registration and interaction with your Smart Account (if set up).
- You can now license multiple clusters with one set of Meeting Server licenses in your Smart Account and you no longer need to load the license file onto each individual Meeting Server instance as was the case prior to 3.0.
- Meeting Management with Smart Licensing tracks how many Call Bridges per cluster, thereby eliminating the need for the R-CMS-K9 activation license.
- For a new deployment with no existing licenses:
  - Newly purchased licenses may be Smart-enabled by default and require a Smart Account – once you have entered the license details into Meeting Management, it will validate the license details against those held in the Smart Account.
- For an existing deployment with a local license file on each Call Bridge:
  - You can upgrade to 3.0 without a Smart Account, and Meeting Management will read the existing license file(s) as per the traditional licensing method.
  - You can move to a Smart Account using the Cisco Smart Software Manager (CSSM) portal and choose the option to convert your existing licenses to Smart.
- SMP and PMP license usage is combined to decide if a day is counted as overage (if either license is over, the whole day is regarded as usage higher than the entitlement). For other feature licenses (for example, recording or custom layout), they are assessed separately and enabled with entitlement via Meeting Management (assuming the license exists in your Smart account).

Note: As Meeting Management is required for all 3.0 deployments, for larger customer deployments, Meeting Management can be deployed in new licensing-only mode without active meeting management.

Smart Accounts can contain Virtual Accounts which allow you to organize your licenses by any designation of your choice, for example, by department. Here are some important points to note when using a Smart Virtual Account with Meeting Server and Meeting Management:

- Each Meeting Server cluster(s) to a single Meeting Management should be linked to a user-defined Smart Virtual Account.
- Each Virtual Account can only connect with a single Meeting Management server that is configured to handle Smart Licensing.
- Only configure a single Meeting Management to Smart – we recommend you do **not** configure a second redundant Meeting Management for Smart Licensing as double counting of license usage will occur.
- PMP, SMP and Recording/Streaming licenses can be shared across multiple clusters with a single Meeting Management instance and Smart Licensing in a single Virtual Account.
- ACU licensing is not available with the Meeting Management licensing dashboard – ACUs are not supported in 3.0 and later.

2.3 How Smart licenses work in Meeting Server – overview

2.3.1 How Smart licenses work in Meeting Server – overview

Note: For full details on using Cisco Meeting Management to administer Smart Licensing, see the Meeting Management 3.0 Administrator Guide.

Meeting Management is mandatory for licensing to work on Meeting Server 3.0 and later. Version 3.0 introduces a new trust and interaction between Meeting Server and Meeting Management to support the new licensing using Smart or for existing customers use of installed licensing files – it's this trusted link that enables Meeting Management to license Meeting Server. A high level work flow for implementing Smart Licensing is as follows:

1. Register your Meeting Management to Smart Licensing Virtual Account.
2. When a Meeting Server first starts up it will have no license status values defined.

   Note: You can use Trial Mode for a 90 day full featured period without licenses.

3. When Meeting Server first connects to a Meeting Management instance set up to administer Smart Licensing, it checks to see if the Meeting Server has previously had a license applied. If not, it will set the license expiry date to 90 days in the future.

   The expiry date for a license is shown in Meeting Management and also returned in the clusterLicensing API, as shown in Section 2.4.3.

   Note: The expiry date for any feature license will only ever be up to a maximum of 90 days in the future.

4. Meeting Management collates Meeting Server licensing usage for the cluster and reports to your Smart Account on a daily basis to check that it has the licenses required to ensure the Meeting Server is in compliance. The Smart Account responds to Meeting Management to indicate if the Meeting Server is compliant or not. Meeting Management then sets the expiry dates as appropriate as follows:

    a. If the Meeting Management identifies that a license exists and is below entitlement for a particular feature, the expiry date will be extended to 90 days in the future.

    Note: If Meeting Server doesn't connect to Meeting Management and send usage data for a period of 90 days then the Meeting Server's license won't get refreshed and will therefore expire. For information on the enforcement actions when a license expires, see Section 2.4.1.

    If a license usage is higher than the entitlement, or a license is not found, then enforcement occurs as follows.

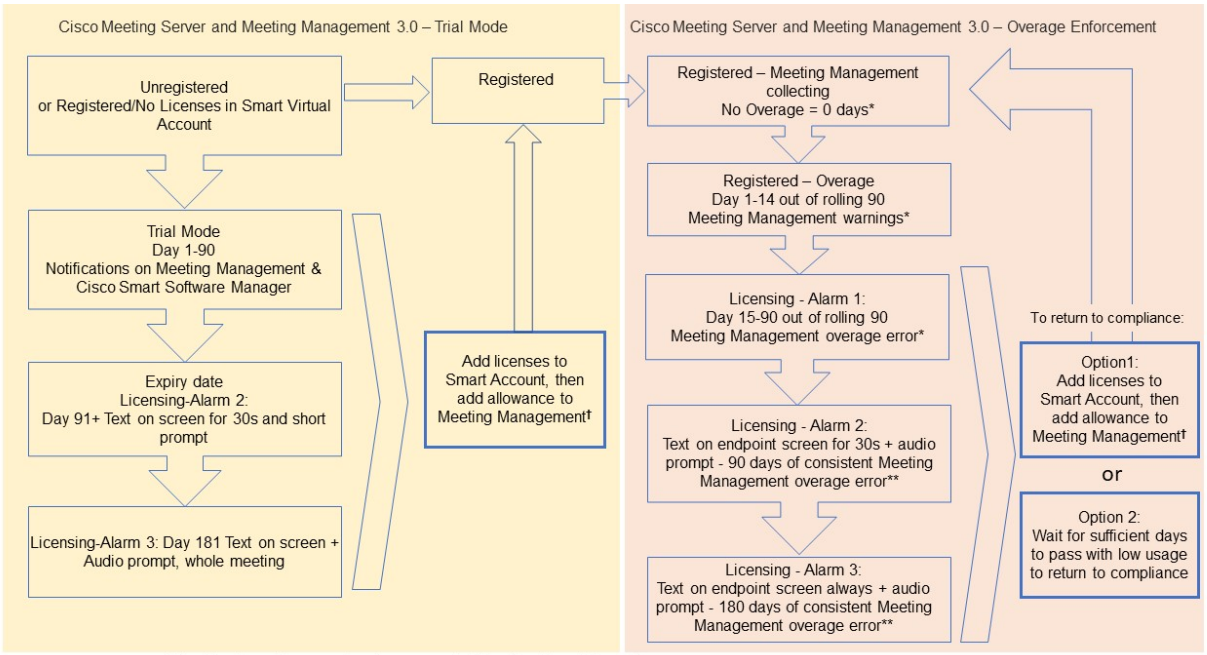
    b. If Meeting Management identifies that less than 15 out of the last 90 days are non-compliant, it will allow this and reset the Meeting Server expiry date to 90 days in the future from that point. The admin will get a visual warning to notify "Insufficient licenses".

    c. If Meeting Management identifies that more than 15 of the last 90 days are non-compliant, the first level of enforcement (Alarm 1) will occur, i.e. out of compliance notifications on the Meeting Management interface.

    d. If overage continues, Meeting Management does not reset the 90 day clock, it gives you a countdown in xx days in which to add new licenses otherwise Alarm levels 2 and 3 will be enabled for all participants joining a meeting as shown in Figure 1.

Figure 1 shows the enforcement flow from initial start up in trial mode on the left-hand side through to overage enforcement as shown on the right-hand side.

Figure 1: Cisco Meeting Server and Cisco Meeting Management Smart Licensing enforcement flow



Note: For detailed information on how to enable and administer licensing for all deployment types, see the Cisco Meeting Management 3.0 Release Notes.

2.4 Expired license feature enforcement actions

2.4.1 Expired license feature enforcement actions

Previously, Meeting Server would evaluate its license file on restart only. From 3.0 the current status of whether a feature is licensed or not can change dynamically, for example, because a feature license expires (previously this would not have been evident until a restart), or there has been an API change. Meeting Management will calculate enforcement actions with Smart Licensing or traditional license file mode.

Note: You can use the Smart Licensing portal to enable email notifications for "insufficient licenses".

When a license feature has expired the actions described in Table 5 will occur.

Table 5: Expired license enforcement actions

Feature	Action
callBridge	When expired: a visual text message displays on screen lasting 30 seconds and an audio prompt plays on joining a meeting for all participants/all meetings. (Alarm level 2)
callBridgeNoEncryption	When expired more than 90 days ago or no license present: the same as before but the visual message is permanent. The audio prompt plays "Your deployment is out of licensing compliance, please contact your administrator". (Alarm level 3)
PMP/SMP	Note: you only need callBridge or callBridgeNoEncryption to prevent the above action.
customizations	When expired or not present, customization features will not be active during a meeting.
recording	When expired or not present you will not be able to start a new recording (regardless of whether it is a 3rd party recorder or not). This license represents recording and streaming so the same restrictions also apply to streaming.

To turn off Alarms 2 and 3, simply add more licenses to your Smart Account.

2.4.2 Smart licensing API additions

This feature introduces the following API additions in version 3.0:

New API objects:

- **`/clusterLicensing`**
- **`/clusterLicensing/raw`**

New API response value:

- **`clusterId`** is added to /system/status response values. This is the ID that identifies the cluster that the Meeting Server is in – this ID remains constant throughout the lifetime of the cluster. As you assign new Call Bridges to a cluster each will take on the same clusterId. An unclustered Meeting Server is regarded as a cluster of 1 so this response parameter will still have a value for a single Meeting Server instance.

Previously, the existing **`/system/licensing`** API returned the contents of the license file, i.e. the feature components for a Meeting Server, together with each component's license status and expiry date (if applicable) shown. For example, whether the callbridge license was activated or not on that Meeting Server, and if licensed, the expiry date.

From 3.0, the existing **`/system/licensing`** API now only returns the contents of the license file (i.e. the feature components) on a per Meeting Server instance, and the newly introduced API object /clusterLicensing returns the license status and expiry date (if applicable) for a Meeting Server cluster.

Note:
The new **/clusterLicensing** API represents the cluster (a single Meeting Server deployment is regarded as a cluster of one). The **/system/licensing** API that represents the license file contents continues to be per Meeting Server instance.

Meeting Server supports a new **/clusterLicensing/raw** API object that writes licensing data in JSON format, and has two mandatory parameters: **data** and **signature** that are used by Meeting Management to create the trusted link and issue the Meeting Server licenses.

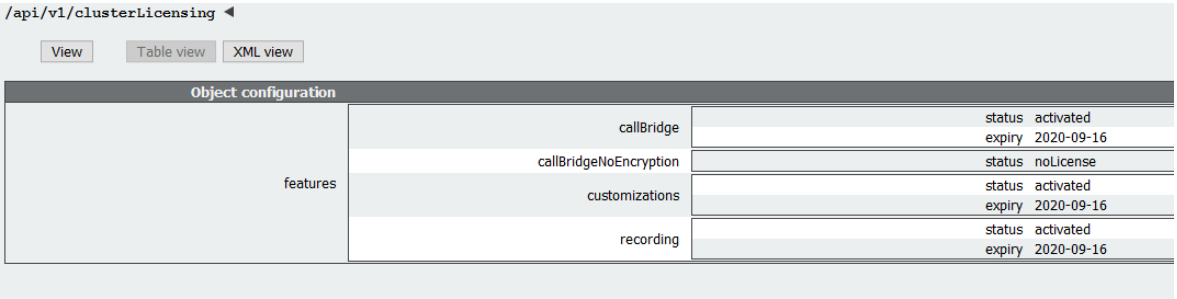
2.4.3 How to retrieve licensing information

2.5 How to retrieve licensing information (Smart Licensing)

To retrieve licensing information for a cluster using the Meeting Server Web Admin interface:

1. Log in to the Meeting Server Web Admin interface and select **Configuration > API**:
2. From the list of API objects, tap the ▶ after **/api/v1/clusterLicensing**
3. The current license status for the cluster is displayed as shown in this example:

Figure 2: clusterLicensing API – license status



2.5.1 MMP license command

The MMP **license** command shows the licenses in the local cms.lic file for the single Meeting Server instance rather than what the cluster is licensed to do (i.e. as implemented with the /clusterLicensing API).

2.6 Image Signing

Meeting Server version 3.0 introduces image signing to improve security when upgrading devices. This new feature introduces signatures to Meeting Server upgrade images, and performs verification of the upgrade images (signature and integrity). Meeting Server uses these signatures to verify the authenticity of upgrade images before each upgrade.

Note: Meeting Server does not support secure boot. The signature verification is only performed during an upgrade.

Previously, administrators were advised to verify the integrity of upgrade images by using the **`upgrade list`** MMP command which displays SHA-256 hashes of all upgrade images. Administrators would then manually verify the hashes against those published in the release notes before proceeding with the upgrade.

This new feature embeds a signature within the upgrade image, which Meeting Server uses to confirm whether the image is genuine. Tampered images are rejected by Meeting Server. This process is done automatically when administrators upgrade to a signed image and removes the need for manual verification. This gives assurance to administrators that the image they are installing / running on Meeting Server is a genuine Cisco image, and has not been tampered with.

Image signatures are only verified when upgrading from a signed image. So manual verification is still advised when upgrading from an unsigned image to a signed one. i.e. if you upgrade from 2.9 to 3.0, or downgrade to earlier versions, you are still advised to manually verify the hashes. This feature will be fully effective when upgrading from 3.0 and beyond.

Meeting Server now prompts the following and requires confirmation before upgrading to an unsigned image:

```
The integrity of the upgrade image cannot be verified.
Are you sure you wish to continue? (Y/n)
```

CAUTION: Upgrading to an untrusted image may compromise the security of your system. Only upgrade to an unsigned image after manually verifying the hashes.

2.6.1 How image signing works

Upgrade images include a signature generated by a secure internal Cisco server which restricts access to the private key. The public key is stored inside the image of the running Meeting Server and is used to validate signatures. The signature is then used to validate the authenticity of the whole image.

2.6.2 Differences introduced in the upgrade process

This new feature is largely transparent to administrators. The Meeting Server can be upgraded in the same way as before, with the following differences:

- on upgrading to an unsigned image, you are warned and asked to confirm whether you want to proceed (this behavior is required for downgrades).
- if the image has been tampered with, the upgrade is prevented.

- a new **upgrade <name> verify** MMP command verifies the status of an image (i.e. signed or not, tampered with or not) without proceeding with the upgrade. This command is optional as the verification is always done on upgrade.
- new **authenticity** MMP commands show the status of the running image, and manage image signature keys. Managing keys should only be done under TAC supervision if there is a need to run engineering special images.

Note: When you run the **authenticity** MMP command when you have upgraded to this software version, it will show "The integrity of the running image could not be verified". This is OK and merely indicates that the Meeting Server was upgraded from an image that doesn't support image signing and therefore could not verify the new image's signature.

2.6.3 Image validation process

The new upgrade process for an administrator is as follows:

1. Upload an upgrade image on Meeting Server via SFTP. Meeting Server does not verify image signatures at this stage.
2. Start an upgrade via MMP console, specifying the image name to use:

```
MMP> upgrade <upgrade_file.img>
```

Meeting Server then performs the following verification process:

1. Extracts the upgrade package.
2. Retrieves the key version/key type used to sign the image and ensures it holds the matching public key
3. Verifies the imported key's signature if the image is signed with a SPECIAL key. The MASTER key is used for this step. Note that this applies to EFT/engineering special images only.
4. Validates the integrity of the whole image using the signature.

If any of the above steps fail, Meeting Server rejects the image giving the reason for the rejection.

2.6.4 Signed image file naming convention

The following convention is used in the image filename:

- ***[release_name]_s<s/p><a/b/…>.img***

where:

- ***[release_name]***: is the release name
- ***_s***: indicates that the file is signed
- ***<s/p>***: indicates if the image is Special / Production
- ***<a/b/…>***: indicates the key version

Note: Upgrade images may be renamed before being uploaded to Meeting Server so their names should not be relied upon to determine the image type. The image type can be retrieved using the MMP **upgrade <name> verify** command.

2.6.5 Key file naming convention

The following convention is used in key filenames:

- **CMS_[key identifier]_key_<a/b/...>_SPECIAL.pem**

where

- ***[key identifier]***: information to identify which SPECIAL upgrade images were signed with this key
- ***<a/b/…>***: indicates the MASTER key version with which this key is signed

---

Note: Key files may not be renamed. Renamed keys will be rejected by the Meeting Server.

2.6.6 File naming examples

The examples below assume keys with version 'a':

Release images	Description
upgrade_spa.img	image signed with internal RELEASE key
vm_upgrade-3.0_spa.img	image signed with internal RELEASE key

Points to note:

- _spa suffix denotes a production image which will be verified with a key internal to Meeting Server.
- the key version may change if there is a need to rotate the keys.

Only beta or Engineering Special release builds will be signed with a SPECIAL key. Production builds will always be signed with a RELEASE key. Some useful information about builds signed with a SPECIAL key:

- a typical file name example is: upgrade_ssa.img
- before upgrading to one of these, the SPECIAL key will need to be uploaded to the Meeting Server – please contact Cisco Support for assistance.
- upgrades from a release signed with a SPECIAL key back to 3.0 (or any later release) will not require any special action from the administrator.

2.7 Minimum passcode length policy

Version 3.0 introduces the minimum passcode length feature which is configurable by an administrator in order to improve security and adhere to an individual company's security policies. The minimum passcode length can be applied to all the different methods of dialing in, for example IVRs, direct SIP dial, and web app.

The minimum passcode length is defined in the new API object **`/dialInSecurityProfiles`**. The newly defined security profile can then be assigned to the top level (global) profile, tenants, coSpaces, or accessMethods. The profile can also be assigned to coSpaceTemplates and /coSpaceTemplates/<coSpace template id>/accessMethodTemplates.

There is a hierarchy of profiles – values in the profiles lower in the hierarchy override those set above, and if a parameter is unset or no dial-in security profile is set then it inherits from the next profile up within the hierarchy.

The hierarchy for dialInSecurityProfile is:

- Top level (global) profile (**`/system/profiles`**)
- Tenants (**`/tenants/<tenant id>`**)
- coSpaces (**`/coSpaces/<cospace id>`**)
- accessMethod (**`/coSpaces/<cospace id>/accessMethods/<access method id>`**)

Dial-in security profiles can also be applied to cospace templates and cospace access method templates as follows:

- coSpaceTemplates (**`/coSpaceTemplates/<coSpace template id>`**)
- accessMethodTemplates (**`/coSpaceTemplates/<coSpace template id>/accessMethodTemplates/<access method template id>`**)

When coSpaces and their associated access methods get instantiated from templates, the dial-in security profiles from the templates get assigned to the corresponding instantiated objects.

For more information on using profiles, see Chapter 14 of the API Reference Guide.

Note: If you use TMS versions earlier than 15.12.0 for scheduled meetings – CUCM ad hoc conferencing calls – do not set a security profile at the system or tenant level.

Note: If the parameter **minPasscodeLength** is set to **0**, it will result in no passcode length enforcement.

This feature introduces the following API additions in version 3.0:

New API objects:

- **/dialInSecurityProfiles**
- **/dialInSecurityProfiles/<dial in security profile id>**

New API request and response parameter:

- **dialInSecurityProfile**

New error codes:

- **dialInSecurityProfileDoesNotExist**
- **passcodeTooShort**

2.7.1 How Meeting Server validates passcode changes

When changing the passcode of a coSpace or an accessMethod using the API or the Web Admin interface (**Configuration > Spaces**) or the web app, if the coSpace or accessMethod is callable, i.e. it has a URI or a call ID, or a secondary URI in the case of a coSpace, it validates whether the new passcode is in policy or not, i.e. whether the passcode length is greater or equal to the effective minPasscodeLength in the coSpace or accessMethod.

If the new passcode is within policy then the change will be accepted as normal. However, if the new passcode is too short then the change will be rejected. The API will reject a change under these circumstances using an HTTP response code "403 Forbidden" (with reason **passcodeTooShort**); web app users will see a message "Passcode requires at least n digits" (where n is the minimum passcode length).

Note: When generating a cospace from a coSpace template, the created coSpace has no URI, call ID or passcode, regardless of the dial-in security profiles set globally, at tenant level or at coSpace template level. Since the coSpace is not callable, Meeting Server does not require that it adheres to the existing dial-in security profiles set in place. If a URI or a call ID (or a secondary URI) gets assigned to the coSpace at a later stage and the passcode is not updated, attempts to join the coSpace will fail if the effective value of minPasscodeLength is greater than 0 and the effective value of allowOutOfPolicy is false. Access methods created from templates always have a URI so their passcodes are auto-generated with respect to the hierarchy of dial-in security profiles.

When joining a meeting using a passcode (SIP or web app), Meeting Server checks whether the supplied passcode is within policy but may also take additional action if the currently set

passcode is out of policy, e.g. if the passcode is too short given the minPasscodeLength settings. The behavior will be as follows:

- If **`allowOutOfPolicy=true`** then Meeting Server will not take additional action if the passcode is too short. You need the correct passcode to join the meeting though.
- If **`allowOutOfPolicy=false`** then you will not be able to join the meeting even if the passcode is correct. In that case, if the passcode is correct but out of policy, there will be a message in the syslog for administrators to take action.

Note: From a users perspective an out of policy rejection will appear the same as an incorrect passcode.

2.7.2 How to create and apply a minimum passcode length

1. To create a dialInSecurityProfile using the Meeting Server Web Admin interface:
   a. Log in to the Meeting Server Web Admin interface and select **Configuration > API**:
   b. From the list of API objects, tap the ► after **/api/v1/dialInSecurityProfiles**
   c. Click **Create new**.
   d. Set the **name** field to the name you wish to call this security profile.
   e. Set the **minPasscodeLength** field to the minimum passcode length you wish to allow – this can be between 0 and 200 (inclusive).
   f. Set the **allowOutOfPolicy** field to either, **true** or **false**. This field determines whether or not users are allowed to join a call using an old passcode that was set before the dial-in security profile was applied and which is no longer compliant with the newly defined passcode length. If this parameter is not supplied, it defaults to "true".
   g. Click **Create**.
2. Assign the ID of the newly created dialInSecurityProfile to any or all of the following, as required:
   - Top level (global) profile (**/api/v1/system/profiles**)
   - Tenants (**/api/v1/tenants/<*id*>**)
   - coSpaces (**/api/v1/coSpaces/<*id*>**)
   - accessMethod (**/api/v1/coSpaces/<*id*>/accessMethods/<*id*>**)
   - coSpaceTemplates (**/api/v1/coSpaceTemplates/<*id*>**)
   - accessMethodTemplates (**/api/v1/coSpaceTemplates/<*id*>/accessMethodTemplates/<*id*>**)

   In this example an updated dialInSecurityProfile is assigned to the top level (global) profile as follows:

a. From the list of API objects tap the ► after **/api/v1/system/profiles**
b. Click **View or edit**
c. Scroll down the parameters to **dialInSecurityProfile** and click **Choose**.
d. From the resulting "dialInSecurityProfile object selector window", click **Select** for the **object id** of the **dialInSecurityProfile** that you have just created in Step 1 that you wish to assign to the top level global profile.
e. Click **Modify**.
f. The newly assigned dialInSecurityProfile object id should now be listed under **Object configuration**.

2.8 SIP recorder and streamer

Previously, Meeting Server's internal recorder and streamer components were dependent upon the Meeting Server's internal XMPP server component – in 3.0 this XMPP server is removed. Version 3.0 introduces a new internal recorder and streamer, both SIP-based.

The new internal recorder and streamer components and dialing out to third-party SIP recorders are all configured using SIP URIs, so when recording or streaming is started the administrator-configured SIP URI is called.

2.8.1 Feature benefits of the new recorder and streamer

- The new recorder and streamer support changing layouts. The recorder/streamer get its layout in a similar way to other SIP calls, i.e. from the defaultLayout parameter on the callLegProfile hierarchy or coSpace object. You can also change the layout parameter in the callLeg.
- Custom layouts can be set using the layoutTemplate parameter (you will need a customizations license to implement custom layouts).
- You can control the maximum resolution on a per callLeg basis using the qualityMain parameter in callLegProfiles and callLegs.
- Previously the XMPP streamer only supported 720p resolution, however the new streamer supports up to 1080p resolution and 3.0 allows you to select the streamer resolution using the MMP comand **`streamer sip resolution`**.
- You can choose whether the streamer/recorder receives presentation by changing the presentationViewingAllowed parameter setting in the callLegProfile.
- Improved scalability with the introduction of the new MMP command **`recorder limit`** and **`streamer limit`**.

2.8.2 Points to note when implementing the new internal recorder and streamer:

- Support for the new internal streamer and recorder, and utilizing an external third-party SIP recorder still requires Meeting Server recording licenses.
- The recorder and streamer applications are only supported on virtualized deployments (including Meeting Server 1000).
- Running a recorder or streamer is not supported on Meeting Server 2000.
- You need to redeploy the recorder and streamer using the MMP interface when upgrading to 3.0 – note that the MMP commands for the new recorder and streamer are different to those used previously. (See the deploying recorder and streamer sections below.)
- We recommend that you do not run a recorder/streamer co-located with a Call Bridge – such a configuration is not supported other than in lab scenarios for test reasons. You should configure a recorder/streamer on a separate VM from the Call Bridge.
- We recommend that you do not run both a recorder and streamer on the same VM.
- The **`/recorders`** and **`/streamers`** API objects used for the recorder and streamer prior to 3.0 are now removed.

Note: The new internal SIP recorder and streamer service cannot be used as an External recording or streaming service as the services rely on specific SIP header parameters passed by the Meeting Server Call Bridge. When calls from any other source that is not Meeting Server Call Bridge connect, the recorder/streamer will reject the call as it won't locate the specific SIP headers expected.

2.8.2.1 VM sizing for the new internal SIP recorder component

The recommended deployment for production usage of the recorder is to run it on a dedicated VM with a minimum of 4 vCPU cores and 4GB of RAM. The following table provides an idea of performance and resource usage for each of the recording types.

Table 6: Internal SIP recorder performance and resource usage

Recording Setting	Recordings per vCPU	RAM required per recording	Disk budget per hour	Maximum concurrent recording
720p	2	0.5GB	1GB	40
1080p	1	1GB	2GB	20
audio	16	100MB	150MB	100

Key point to note (applies to new internal recorder component only):

- Performance scales linearly adding vCPUs up to the number of host physical cores.

2.8.2.2 VM sizing for the new internal SIP streamer component

The recommended deployment for production usage of the streamer is to run it on a dedicated VM with a minimum of 4 vCPU cores and 4GB of RAM. The following table gives an idea of 3 recommended minimum specifications and the number of streams they can handle.

Table 7: Internal SIP streamer recommended specifications

Number of vCPUs	RAM	Number of 720p streams	Number of 1080p streams	Number of audio-only streams
4	4GB	50	37	100
4	8GB	100	75	200
8	8GB	200	150	200

Key points to note (applies to new internal streamer component only):

- Number of vCPUs should not oversubscribe the number of physical cores.
- Maximum number of 720p streams supported is 200 regardless of adding more vCPUs.
- Maximum number of 1080p streams supported is 150 regardless of adding more vCPUs.
- Maximum number of audio-only streams supported is 200 regardless of adding more vCPUs.

2.8.3 New API command to specify the SIP streamer

A new API parameter for **`/callProfiles`** object is introduced that takes the value of a string to specify the URI for the SIP streamer. It supports GET, PUT and POST and is defined as follows:

- **`sipStreamerUri`** – If set, this URI is used to dial out to when streaming is enabled.

Note: **`sipRecorderUri`** API parameter was introduced in version 2.9 and is also specified in the API call profile object.

2.8.4 New MMP commands

Prior to 3.0, certificates and trust commands referred to the https link between the Call Bridge and the recorder/streamer components. These are no longer required for the new recorder/streamer components in 3.0. Instead Meeting Server allows you to configure a SIP certificate using the new MMP command **`recorder sip certs`** and **`streamer sip certs`**.

The **`listen`** command used prior to 3.0 referred to listening for https connections from the Call Bridge. The new recorder/streamer components do not need to listen for https connections, however, they do need to listen for SIP connections. To achieve this, the following new MMP command is introduced for setting both TCP and TLS: **`recorder sip listen <interface> <tcp-port|none> <tls-port|none>`**.

A new streamer and recorder **sip trace** command is introduced to turn on logging of all SIP messages on the streamer or recorder.

For full details of all the new MMP commands and deprecations, see Section .

2.8.5 Deploying the new recorder component on a VM server

This is a two stage process:

- Configuring a Meeting Server recorder via the MMP
- Configuring the recorder URI via the API

Task 1: Configuring a Meeting Server recorder via the MMP

1. Upgrade to version 3.0.
2. SSH into the MMP and login to configure the recorder (enter the MMP command, **recorder** to see a list of all available commands).
3. Enter **recorder nfs <hostname/IP>:<directory>** to configure the NFS location.
4. Enter **recorder resolution <audio|720p|1080p>** to configure the desired resolution (or to only record the audio of calls).
5. Configure the listening interface of the recorder and the SIP TCP and TLS ports to listen on using the MMP command **recorder sip listen <interface> <tcp-port|none> <tls-port|none>**. Set the respective port to **none** to disable the service:
   a. For example, if you want to only listen on the TLS port and not the TCP port, enter **recorder sip listen a none 6000**
   b. Make a note of the ports you've configured if they're not the default TCP/TLS ports (5060/5061) as they will be needed later.

   Note: If you want to listen on the default SIP TCP/TLS ports (5060/5061) you MUST ensure that the Call Bridge is not listening on the same interface, otherwise the ports will clash. You must disable the Call Bridge by removing the corresponding interface, by entering the MMP command **callbridge listen none**.
6. Optionally, if TLS is configured, configure the SIP TLS certificates you would like to use:
   a. Enter the MMP command **recorder sip certs <key-file> <crt-file> [<crt-bundle>]**

   Note: Note that if SIP TLS certificates are not configured with this option, the SIP TLS service will fail to start.

7. Optionally, if TLS is configured, you can perform TLS verification for SIP on the recorder as follows:
   a. Enter the MMP command **`tls sip trust [<crt-bundle>]`**
   b. Enter the MMP command **`tls sip verify enable`**

   Note: For the TLS connection to be secure we recommend enabling TLS verification.

8. Check the configuration is correct – enter the MMP command **`recorder`** to view the configuration.
9. Enter the MMP command **`recorder enable`** to enable the recorder service.

Task 2: Configuring the recorder URI via the API

Once the new SIP recorder is enabled, it can be configured and used in the Call Bridge in the same way as a third-party SIP recorder, using the **`sipRecorderUri`** API parameter specified in the API call profile object.

If you wish, you can also configure a custom URI that maps to an outboundDialPlan rule (the domain can be anything of your choice, e.g. "recording.com"). You will need to configure an outboundDialPlan rule which tells Meeting Server how to route the domain used in **`sipRecorderUri`** to the recorder. This will allow you to control priority values, encryption, etc. For more information on configuring outboundDialPlan rules, see the "Dial plan configuration – overview" chapter of your deployment guide.

Note: The user part of the configured URI (i.e. the part before the '@' symbol) has no special meaning, and for the new internal SIP recorder component, although required, it can usually be anything, e.g. "recording@recorder.com". However, this may not be the case for third-party SIP recorders which may use the user part of the URI for user credentials, for example. The important part of the URI is the domain part.

To configure the **`sipRecorderUri`** parameter using the Meeting Server Web Admin interface:

1. Log in to the Meeting Server Web Admin interface and select **Configuration > API**:
2. From the list of API objects, tap the ► after **/api/v1/callProfiles**
3. To configure or modify an existing call profile, select the object id of the required callProfile and fill in the **sipRecorderUri** field with your chosen URI.

   Note: When using the new SIP recorder you only need to use one SIP URI, e.g recording@recorder.com, you don't need to have different SIP URIs on different profiles (it makes no difference).

4. If you haven't done so already, set the **recordingMode** field to either, **manual** or **automatic** (depending on how you want meetings to be recorded).
5. Click **Modify**.

The updated callProfile can then be assigned to coSpaces, tenants or the top level (global) profile, as required. In this example an updated callProfile is assigned at the global level as follows:

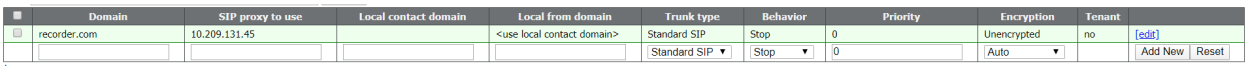
1. Using the Web Admin interface, select **Configuration > API**:
   a. From the list of API objects tap the ► after **/api/v1/system/profiles**
   b. Click **View or edit**
   c. Scroll down the parameters to **callProfile** and click **Choose**.
   d. From the resulting "callProfile object selector window", click **Select** for the **object id** of the **callProfile** you wish to assign to the top level global profile.
   e. Click **Modify**.
   f. The newly assigned callProfile object id should now be listed under **Object configuration**.

2.8.5.1 callProfile configuration example (if using a matching outbound dial plan rule):

In this example, **recordingMode** is set to **automatic** and **sipRecorderUri** to **recording@recorder.com** using the steps above.

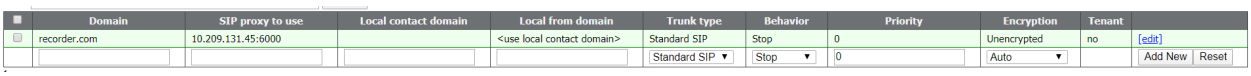
Object configuration	
recordingMode	automatic
sipRecorderUri	recording@recorder.com

From the Meeting Server Web Admin interface select **Configuration > Outbound calls** to see the matching outbound dial plan rule:



Domain	SIP proxy to use	Local contact domain	Local from domain	Trunk type	Behavior	Priority	Encryption	Tenant	
recorder.com	10.209.131.45		<use local contact domain>	Standard SIP	Stop	0	Unencrypted	no	[edit]
				Standard SIP	Stop	0	Auto		Add New Reset

If you configured the recorder in the MMP to use SIP TCP/TLS ports which are different from the default standard ports (5060/5061), you MUST specify the listening port in the **sipRecorderUri** field or in the matching outbound dial plan rule if you are using one, as shown below:



Domain	SIP proxy to use	Local contact domain	Local from domain	Trunk type	Behavior	Priority	Encryption	Tenant	
recorder.com	10.209.131.45:6000		<use local contact domain>	Standard SIP	Stop	0	Unencrypted	no	[edit]
				Standard SIP	Stop	0	Auto		Add New Reset

If using an outbound dial plan rule, make sure the service of the port specified matches the encryption type, for example, if using the SIP TLS port, set the **Encryption** mode to **Encrypted**.

2.8.6 Deploying the new streamer component on a VM server

This is a two stage process:

- Configuring a Meeting Server streamer via the MMP
- Configuring the streamer URI via the API

Task 1: Configuring a Meeting Server streamer via the MMP

1. Upgrade to version 3.0.
2. SSH into the MMP and login to configure the recorder (enter the MMP command, **`streamer help`** to see a list of all available commands).
3. Configure the listening interface of the streamer and the SIP TCP and TLS ports to listen on using the MMP command **`streamer sip listen <interface> <tcp-port|none> <tls-port|none>`**. Set the respective port to **`none`** to disable the service:
   a. For example, if you want to only listen on the TLS port and not the TCP port, enter **`streamer sip listen a none 6000`**
   b. Make a note of the ports you've configured if they're not the default TCP/TLS ports (5060/5061), as they will be needed later.
4. Optionally, you can set the maximum resolution that you want the streamer to do (or to only stream the audio of calls) using the MMP command **`streamer sip resolution <audio|720p|1080p>`**, if not specified, the default is 720p.
   a. For example, if you want to set it to 1080p, enter **`streamer sip resolution 1080p`**

   Note: If you want to use 1080p we recommend that you increase your transmit SIP call bandwidth to 3,500,000 bits per second to optimize the video quality. To do this, on the Web Admin UI go to **Configuration > Call settings > Bandwidth settings (SIP)** and set as required.
5. Optionally, if TLS is configured, configure the SIP TLS certificates you would like to use:
   a. Enter the MMP command **`streamer sip certs <key-file> <crt-file> [<crt-bundle>]`**

   Note: Note that if SIP TLS certificates are not configured with this option, the SIP TLS service will fail to start.
6. Optionally, if TLS is configured, you can perform TLS verification for SIP on the streamer as follows, for example:

a. Enter the MMP command **`tls sip trust [<crt-bundle>]`**

b. Enter the MMP command **`tls sip verify enable`**

Note: For the TLS connection to be secure we recommend enabling TLS verification.

7. Check the configuration is correct – enter the MMP command **`streamer`** to view the configuration.
8. Enter the MMP command **`streamer enable`** to enable the streamer service.

Task 2: Configuring the streamer URI via the API

Once the new SIP streamer is enabled, it can be configured and used in the Call Bridge using the **`sipStreamerUri`** API parameter specified in the API call profile object.

If you wish, you can also configure a custom URI that maps to an outboundDialPlan rule (the domain can be anything of your choice, e.g. "streaming.com"). You will need to configure an outboundDialPlan rule which tells Meeting Server how to route the domain used in **`sipStreamerUri`** to the streamer. This will allow you to control priority values, encryption, etc. For more information on configuring **`/outboundDialPlanRules`**, see the "Dial plan configuration - overview" chapter of your deployment guide.

Note: The user part of the configured URI (i.e. the part before the '@' symbol) has no special meaning, and for the new internal SIP streamer component, although required, it can usually be anything, e.g. "streaming@streamer.com". The important part of the URI is the domain part.

To configure the **`sipStreamerUri`** parameter using the Meeting Server Web Admin interface:

1. Log in to the Meeting Server Web Admin interface and select **Configuration > API**:
2. From the list of API objects, tap the ► after **/api/v1/callProfiles**
3. To configure or modify an existing call profile, select the object id of the required callProfile and fill in the **sipStreamerUri** field with your chosen URI.

   Note: When using the new SIP streamer you only need to use one SIP URI, e.g streaming@streamer.com, you don't need to have different SIP URIs on different profiles.

4. If you haven't done so already, set the **`streamingMode`** parameter to either, **`manual`** or **`automatic`** (depending on how you want meetings to be streamed).
5. Click **Modify**.

The updated callProfile can then be assigned to coSpaces, tenants or the top level (global) profile, as required. In this example an updated callProfile is assigned at the global level as follows:

1. Using the Web Admin interface, select **Configuration > API**:
    a. From the list of API objects tap the ► after **/api/v1/system/profiles**
    b. Click **View or edit**
    c. Scroll down the parameters to **callProfile** and click **Choose**.
    d. From the resulting "callProfile object selector window", click **Select** for the **object id** of the **callProfile** you wish to assign to the top level global profile.
    e. Click **Modify**.
    f. The newly assigned callProfile object id should now be listed under **Object configuration**.

For each coSpace in the API that you wish to enable streaming for, you must configure the **`streamUrl`** coSpace API field with the RTMP stream URL to stream to (e.g. "rtmp://mystream.com/live/app"). To configure this:

1. Log in to the Meeting Server Web Admin interface and select **Configuration > API**:
2. From the list of API objects, tap the ► after **/api/v1/coSpaces**
3. To configure or modify an existing coSpace, select the object id of the required coSpace and fill in the **streamUrl** field with the RTMP stream URL to stream to.
4. Click **Modify**.

2.8.7 Known Limitations

CAUTION: Be warned that the stream URL is sent via SIP headers, so any RTMP stream URLs containing login credentials could potentially be exposed to call control providers which may log them.

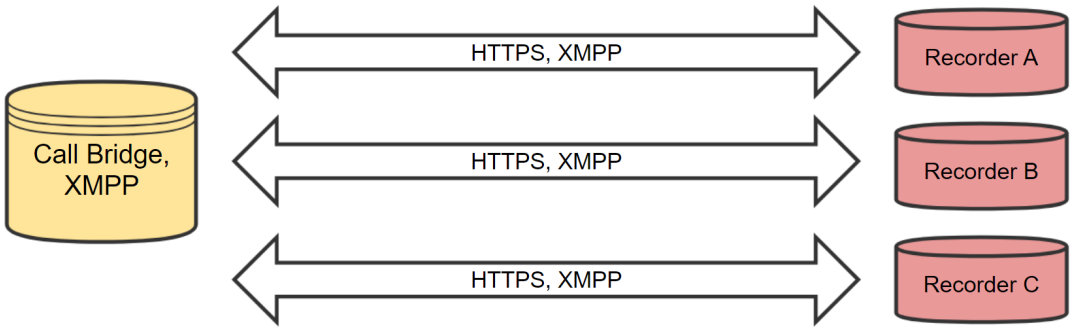
The new SIP Streamer component does not support RTMPS.

2.8.8 Deploying a Recorder and Streamer for Scalability and Resiliency

When deploying more than one recorder or streamer, we recommend that you deploy them behind a Call Control provider and allow the Call Control provider to give load-balancing and fail-over support. You will need to set the **`sipRecorderUri`** API parameter to point to a dial plan-rule, which can forward calls to the proxy.

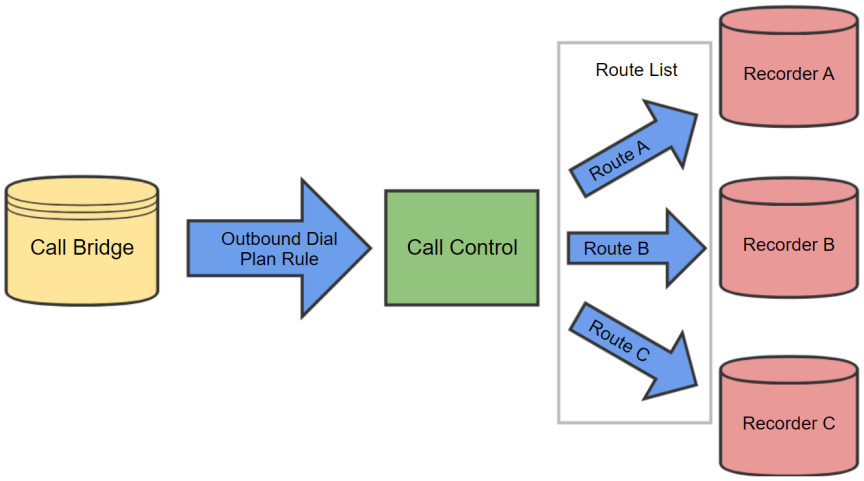
Call Control flow behavior for the new recorder is different to that for the old XMPP recorder. Previously, recorders would connect directly to the Call Bridge as an XMPP client, and send information of availability via an HTTPS link, as shown in Figure 3.

Figure 3: Old XMPP recorder Call Control flow



For the new internal SIP recorder component, Call Control flows through a Call Control provider while media will, in most cases, flow directly between the Call Bridge and the recorder. In some cases, media may also flow through the Call Control provider, depending on the call control provider configuration. This new recorder Call Control flow is shown in Figure 4.

Figure 4: New internal SIP recorder Call Control flow



2.8.8.1 Supported Call Control methods

- **Cisco Unified Communications Manager** – Each recorder in a cluster should be deployed behind a SIP trunk, with each recorder in a cluster associated with the same route list.
- **Cisco Expressway** – Each recorder should be configured to have their own zone, with a route pattern mapping to all recorders in the cluster.

- **Direct Flow** – You may set the dial plan rule's proxy to be the address/FQDN of the recorder / streamer, but this is only recommended when deploying one instance.

For more information, see Cisco Meeting Server 2.x, White Paper on Load Balancing Calls Across Cisco Meeting Servers.

2.9 Web Bridge profiles and settings in the API

Version 3.0 removes the **Web bridge settings** configuration option from the **Configuration > General** Web Admin user interface page. This Web Bridge configuration is now moved to the API and some redundant configuration options are removed. Web Bridge configuration is now done through the API; notably you can do this using the **Configuration > API** page of the Web Admin interface.

Note: The **Web Bridge URI** and **IVR telephone number** fields are still currently set under **External access** on the **Configuration > General** page in the Web Admin user interface. These configuration fields may be moved to web bridge profiles in a future release.

These newly introduced changes allow you to configure some Web Bridge configuration options in a common place rather than solely on a per Web Bridge basis – you can now apply the same settings for all, or a specified group of Web Bridges.

To support this change, the /webBridgeProfiles API object is introduced which contains the various Web Bridge configuration options. A newly defined Web Bridge profile can be assigned to the individual webBridge objects, or to the top level (global) profile or tenants.

There is a hierarchy of profiles – values in the profiles lower in the hierarchy override those set above, and if a parameter is unset or no web bridge profile is set then it inherits from the next profile up within the hierarchy.

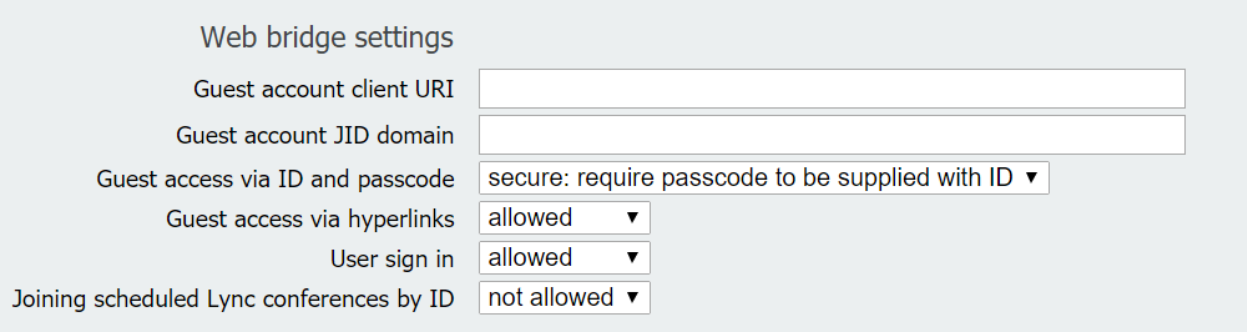
The hierarchy for webBridgeProfiles is:

- Top level (global) profile (**`/system/profiles`**)
- Tenants (**`/tenants/<tenant id>`**)
- webBridges (**`/webBridges/<webbridge id>`**)

2.9.1 Web Admin user interface changes

Previously, the **Configuration > General** page contained the Web Bridge settings options as shown in Figure 5 – these options are now removed from this page.

Figure 5: Web bridge settings removed from 3.0 Web Admin user interface



The fields under **Web bridge settings** are dealt with as follows in 3.0:

- **Guest account client URI**: removed, not required for deploying web app
- **Guest account JID domain**: removed, not required for deploying web app
- **Guest access via hyperlinks**: moved to webBridgeProfiles under **`allowSecrets`**
- **User sign in**: replaced with **`userPortalEnabled`** under webBridgeProfiles
- **Joining scheduled lync conferences by ID**: moved to webBridgeProfiles as **`resolveLyncConferenceIds`**

2.9.2 API additions and changes

This feature introduces the following API additions in version 3.0:

New API objects:

- **`/webBridgeProfiles`**
- **`/webBridgeProfiles/<web bridge profile id>`**
- **`/webBridges/<web bridge id>/effectiveWebBridgeProfile`**
- **`/tenants/<tenant id>/effectiveWebBridgeProfile`**
- **`/system/profiles/effectiveWebBridgeProfile`**

New API request and response parameter:

- **`webBridgeProfile`**

New error code:

- **`webBridgeProfileDoesNotExist`**

Moved API request and response parameter:

Previously the **`resolveCoSpaceUris`** parameter was on the **`/webBridges/<web bridge id>`** API object. From version 3.0 this parameter can now be found on the following API objects:

- **/webBridgeProfiles**
- **/webBridgeProfiles/<web bridge profile id>**
- **/webBridges/<web bridge id>/effectiveWebBridgeProfile**
- **/tenants/<tenant id>/effectiveWebBridgeProfile**
- **/system/profiles/effectiveWebBridgeProfile**

Other parameters that were on **/webBridges/<web bridge id>** API object that are now moved or removed in 3.0 are:

- **resourceArchive** – now in webBridgeProfiles
- **idEntryMode** – now deprecated
- **allowWeblinkAccess** – now in webBridgeProfiles as **allowSecrets**
- **showSignIn** – now in webBridgeProfiles as **userPortalEnabled**
- **resolveCoSpaceCallIds** – now in webBridgeProfiles
- **resolveLyncConferenceIds** – now in webBridgeProfiles

2.9.3 How to create and apply a web bridge profile

1. To create a webBridgeProfile using the Meeting Server Web Admin interface:
   a. Log in to the Meeting Server Web Admin interface and select **Configuration > API**:
   b. From the list of API objects, tap the ► after **/api/v1/webBridgeProfiles**
   c. Click **Create new**.
   d. Set the **name** field to the name you wish to call this web bridge profile.
   e. Set the **resourceArchive** field to the address of any customization archive file that the Meeting Server should use for web bridges using this web bridge profile.
   f. Set the **allowPasscodes** field to either **true** or **false**. This field determines whether or not web bridges using this web bridge profile should allow users to lookup coSpaces (and coSpace access methods) with passcodes in combination with an numeric ID/URI. If this parameter is not supplied, it defaults to **true**.
   g. Set the **allowSecrets** field to either **true** or **false**. This field determines whether or not web bridges using this web bridge profile should allow users to access coSpaces (and coSpace access methods) through a meeting join link with a numeric ID and secret. If this parameter is not supplied, it defaults to **true**.
   h. Set the **userPortalEnabled** field to either **true** or **false**. This field determines whether or not web bridges using this web bridge profile should display the sign-in tab on the index page. If this parameter is not supplied, it defaults to **true**.

i. Set the **allowUnauthenticatedGuests** field to either **true** or **false**. If set to **true**, guest access is allowed from the landing screen on web bridges using this web bridge profile. If set to false, visitor access is only allowed once users have logged into the User Portal. If this parameter is not supplied, it defaults to **true**.

j. Set the **resolveCoSpaceCallIds** field to either **true** or **false**. This field determines whether or not web bridges using this web bridge profile should accept coSpace and coSpace access method call IDs for the purpose of allowing visitors to join cospace meetings. If this parameter is not supplied, it defaults to **true**.

k. Set the **resolveLyncConferenceIds** field to either **true** or **false**. This field determines whether or not web bridges using this web bridge profile should accept IDs to be resolved to Lync scheduled conference IDs. If this parameter is not supplied, it defaults to **false**. (This field is visible but non-functional in 3.0.)

l. Set the **resolveCoSpaceUris** field to either **off**, **domainSuggestionDisabled** or **domainSuggestionEnabled**. This field determines whether or not this web bridge should accept coSpace and coSpace access method SIP URIs for the purpose of allowing visitors to join cospace meetings. When set to **off**, join by URI is disabled; when set to **domainSuggestionDisabled**, join by URI is enabled but the domain of the URI won't be auto-completed or verified on this web bridge; when set to **domainSuggestionEnabled** join by URI is enabled and the domain of the URI can be auto-completed and verified on this web bridge. If this parameter is not supplied, it defaults to **off**.

m. Click **Create**.

2. Assign the ID of the newly created webBridgeProfile to any or all of the following, as required:

   - Top level (global) profile (**/api/v1/system/profiles**)
   - Tenants (**/api/v1/tenants/<*id*>**)
   - WebBridges (**/api/v1/webBridges/<*id*>**)

   In this example an updated webBridgeProfile is assigned to the top level (global) profile as follows:

   a. From the list of API objects tap the ► after **/api/v1/system/profiles**

   b. Click **View or edit**

   c. Scroll down the parameters to **webBridgeProfile** and click **Choose**.

   d. From the resulting "webBridgeProfile object selector window", click **Select** for the **object id** of the **webBridgeProfile** that you have just created in Step 1 that you wish to assign to the top level global profile.

e. Click **Modify**.

f. The newly assigned webBridgeProfile object id should now be listed under **Object configuration**.

2.10 Cisco Meeting Server web app new features and changes

Version 3.0 introduces some new features and changes on Cisco Meeting Server web app. Additionally, web app scale has been increased – for call capacity details see Table 2.

Note: For details of all new 3.0 web app features, see Cisco Meeting Server 3.0 web app Important Information. The new web app features listed below are those that may require server-side configuration.

2.10.1 Join a meeting using a video address (URI) on Cisco Meeting Server web app

Version 3.0 allows a participant to join a meeting on web app by entering a video address (URI).

In 3.0 this feature is operational and does not require any administrator configuration providing the inbound dial plan rules are configured appropriately – the domains are any domain configured in the inbound dial plan rules (under the same tenant as the Web Bridge 3 in question) that allow calling into coSpaces, i.e. that have **Targets spaces** set to **yes**.

Domain names are configured on the Meeting Server Web Admin interface under: **Configuration > Incoming Calls > Call Matching**.

2.10.2 Change in permissions for web app participants

3.0 introduces a change in permissions for web app (Web Bridge 3) participants from previous Meeting App for WebRTC (Web Bridge 2) behavior.

Previously, the ability to add/remove participants was based on whether that participant was a member of the space. From 3.0, for web app this is controlled by the CallLegProfile associated with the CallLeg, as you would expect for SIP participants, for example.

The **`/CallLeg/CallLegProfile`** properties that are now implemented for web app participants are: **`disconnectOthersAllowed`**, **`endCallAllowed`**, **`addParticipantAllowed`**

For more information, see the API Reference Guide.

2.10.3 Name label behavior change seen by web app participants in their conference video

The name label behavior that a web app participant sees in their conference video is now the same as you would see for a SIP call – i.e. they will now appear or not as specified by the prevailing callLegProfile **`participantLabels`** setting, as per SIP name labels.

2.10.4 Other web app feature additions

Version 3.0 introduces controls for the following features on the web app interface:

- Recording/streaming
- Lock/unlock meeting
- Importance

All permissions to use these features on Web Bridge 3 are defined in the CallLegProfile settings on the API and are therefore unchanged to the implementation of these features on Web Bridge 2.

2.10.5 C2W connection certificate change

3.0 introduces a change to the C2W connection certificate for Web Bridge 3 so that the trust store no longer needs root certificates. This gives administrators more flexibility on which certificates are trusted. For example, if an admin needs to use a public certificate to protect the C2W connection due to a company's internal policies, now they can still choose not to trust all the certificates signed by that public CA but trust only the client or server C2W certificate used in the other end. This is called certificate pinning.

2.10.6 Customizing the web app sign-in page

Version 3.0 introducesd customization and branding for your Cisco Meeting Server web app sign-in page.

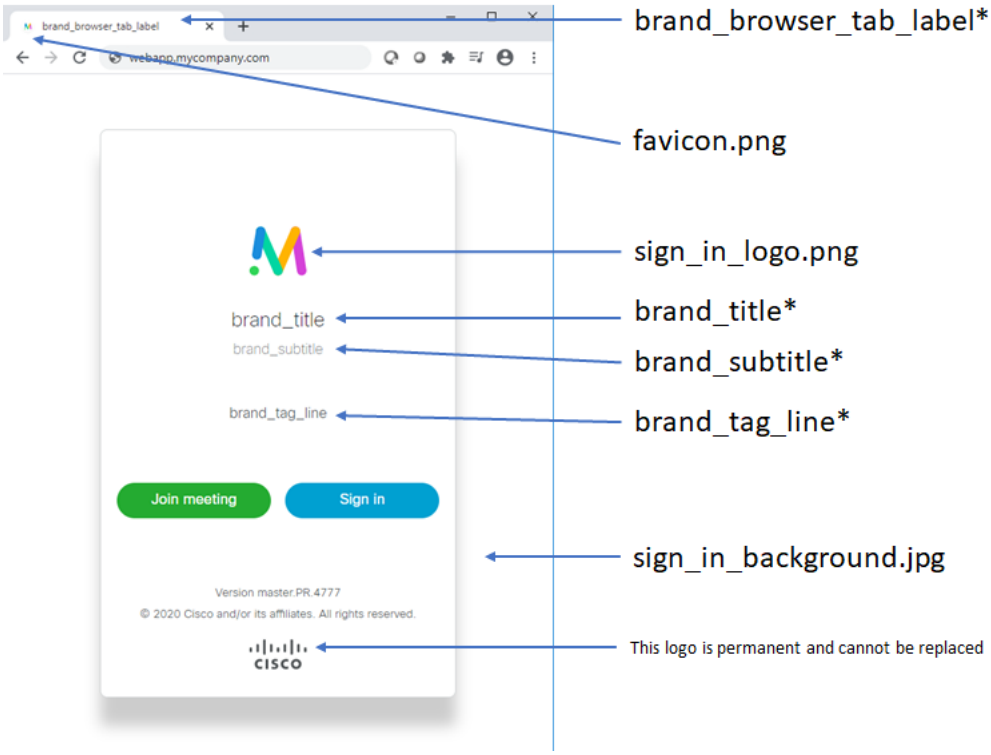
Note: You cannot use a previous Cisco Meeting App for WebRTC branding zip file, you will need to create and deploy a new branding zip file specifically for web app. However, the branding zip file is deployed for web app in the same way as previously for the WebRTC app. (Note that resourceArchive is now located under the webBridgeProfiles API.)

You can use the API to customize these elements of the web app:

- icon shown next to the browser tab, and on any bookmarks / shortcuts
- text on browser tab
- sign-in background image,
- sign-in dialog box – logo displayed,
- sign-in dialog box – text below logo

The positioning and location of these elements is shown in Figure 6.

Figure 6: web app assets



all these strings are contained within the single text_strings.json file (see Table 8).

Table 8 describes the files that can be uploaded to customize web app as shown in Figure 6 and their recommended sizes.

Note: All files must be in the specified file format, e.g. .png, .jpg, or .json. All file names are case sensitive and must adhere to the file name conventions used in Table 8.

Table 8: web app asset descriptions and specifications

File name	Description	Max file-size	Recommended sizes, formats and aspect ratios
favicon.png	The icon shown next to the browser tab label, and on any bookmarks / shortcuts	128 kb	• Recommended resolution: 16x16 pixels or 32x32 pixels • Recommended aspect ratio: 1:1 (square)
sign_in_logo.png	The logo shown on the landing page, the splash screen and the user portal	250 kb	• Recommended resolution: 128x128 pixels • Recommended aspect ratio: Preferably 1:1 (square) • Other recommendations: Transparent background

File name	Description	Max file-size	Recommended sizes, formats and aspect ratios
sign_in_ background.jpg	The background shown on the land-ing page	500 kb	• Recommended resolution: 1920x1080 pixels • Recommended aspect ratio: Preferably 16:9
text_strings.json	A JSON formatted file of text strings which can be overwritten. Supported strings: • brand_title: Main brand name • brand_subtitle: Secondary text below • brand_title brand_tag_line: Tertiary text below • brand_subtitle brand_ browser_tab_label: The name of the tab in the browser	16 kb	Recommended lengths: • brand_title: up to 24 characters (displays on 1 line), or up to 48 characters (displays on 2 lines). • brand_subtitle: up to 24 characters (displays on 1 line), or up to 48 characters (displays on 2 lines). • brand_tag_line: up to 100 characters • brand_browser_tab_label: up to 64 characters

You can customize each of these text strings as shown in the example in Figure 7.

Figure 7: Example contents of text_strings.json

```
{
 "brand_title": "Cisco Meeting Server",
 "brand_subtitle": "web app",
 "brand_tag_line": "Join meetings anywhere, anytime",
 "brand_browser_tab_label": "Cisco Meeting Server web app"
}
```

For full details on implementing this level of customization, see the Cisco Meeting Server 3.0 Customization Guidelines.

2.11 Automatic Gain Control (AGC) enabled by default

Note: This feature was introduced in version 2.8 as a beta feature and was fully supported in version 2.9. In both 2.8 and 2.9 it was disabled by default. From version 3.0 AGC is enabled by default.

Due to different audio levels being set by third-party clients and the variation in audio levels from different headsets, conferences can often have participants that sound too loud or too quiet. Meeting Server uses Automatic Gain Control (AGC) to adjust audio level that it receives from individual participants in order to deliver as consistent an audio level across the conference as possible.

From 2.8, Meeting Server introduces Automatic Gain Control (AGC) on audio received by the Meeting Server. (It is not on audio transmitted by the Meeting Server.)

AGC will be applied to any endpoint (physical endpoints or soft clients) connected directly to the Meeting Server. It will not be applied to TIP calls or AVMCU (because this is a mixed audio stream).

Note:

- Skype participants connected to AVMCU will not be subject to any AGC as the AVMCU controls the audio.
- AGC is not applied to distribution links between Meeting Servers because this is a mixed audio stream.

AGC is now enabled by default and can only be disabled via the parameter **`audioGainMode`** which has two possible values, **`agc`** (default) and **`disabled`**. The **`audioGainMode`** parameter is supported on these APIs:

- GET and PUT operations on **`/callLegProfiles/<call leg profile id>`** and also POST on **`/callLegProfiles`**
- GET and PUT operations on **`/callLegs/<call leg id>`** and also POST on **`/callLegs`**
- GET and PUT operations on **`/calls/<call id>/callLegs`**

When AGC is enabled, the gain applied is visible on the **Status > Calls** Web Admin user interface page. The API parameter **`gainApplied`** is also returned in response to a GET operation on **`/callLegs/<call leg id>`** under the **`rxAudio`** section.

2.12 ESXi support

Version 3.0 adds support on Meeting Server 1000 M4, M5, and specs-based servers for:

- ESXi7.0 with Virtual Hardware version 17

Previous ESXi versions also supported by version 3.0 include ESXi6.0, 6.5u2, and 6.7.

2.13 Historical record of PMP license assignment

Meeting Server 3.0 now allows you to view a historical record of the assigned number of PMP licenses at regular intervals. To support this, Meeting Server introduces a new **`pmpAssigned`** response parameter for each license usage event available upon GET on **`/system/MPLicenseUsage`**. This response value shows the number of users assigned a PMP license.

As previously, PMP licenses are assigned to users via LDAP sync. Meeting Server shows how many users have been assigned a PMP license in the **`/system/multipartyLicensing`** API using the **`personalLicenses`** parameter response value.

As per previous releases, Meeting Server takes snapshots of license usage at regular intervals in time and historical records are available upon GET on **`/system/MPLicenseUsage`**. However, for each license usage event, the new parameter **`pmpAssigned`** provides the number of users in the cluster who have been assigned a PMP license, while the existing parameter **`pmp`** tracks how many of the available PMP licenses are currently in use.

2.14 Summary of 3.0 API Additions and Changes

New API functionality for the Meeting Server 3.0 includes:

- New API objects and parameters to support Smart Licensing
- New API parameters to support the new SIP streamer
- New API objects and parameters to support dial-in security profiles
- New API objects and parameters to support web bridge profiles

2.14.1 API additions

The following new API objects are introduced in version 3.0:

- **`/clusterLicensing`**
- **`/clusterLicensing/raw`**
- **`/dialInSecurityProfiles`**
- **`/dialInSecurityProfiles/<dial in security profile id>`**
- **`/webBridgeProfiles`**
- **`/webBridgeProfiles/<web bridge profile id>`**
- **`/system/profiles/effectiveWebBridgeProfile`**
- **`/tenants/<tenant id>/effectiveWebBridgeProfile`**
- **`/webBridges/<web bridge id>/effectiveWebBridgeProfile`**

New API parameters in version 3.0:

- **`sipStreamerUri`** added for /callProfiles API objects
- **`dialInSecurityProfile`** added for the following API objects:
  - **`/system/profiles`**
  - **`/tenants`**
  - **`/coSpaces`**
  - **`/coSpaces/<cospace id>/accessMethods`**
  - **`/coSpaceTemplates`**
  - **`/coSpaceTemplates/<coSpace template id>/accessMethodTemplates`**
- **`webBridgeProfile`** added for the following API objects:
  - **`/webBridges`**
  - **`/system/profiles`**
  - **`/tenants`**

New API response parameters in version 3.0:

- **clusterId** – added on existing **/system/status** API
- **dnsFailure** – added on existing **/webBridges/<web bridge id>/status** API
- **pmpAssigned** – added on existing **/system/MPLicenseUsage**

New API failure reasons introduced in 3.0:

- **dialInSecurityProfileDoesNotExist**
- **passcodeTooShort**
- **webBridgeProfileDoesNotExist**

Updated API failure reasons introduced in 3.0:

- **recordingNotAllowedByLicensing** (Previously **recordingLimitReached**)
- **streamingNotAllowedByLicensing** (Previously **streamingLimitReached**)

New alarm type introduced in 3.0 as enumeration of /system/alarms:

- **c2wConnectionFailure**

2.14.2 API removals

API objects removed in version 3.0 due to the removal of these components or functionality:

- **/coSpaces/<coSpace id>/messages**
- **/recorders**
- **/streamers**
- **/system/configuration/xmpp**

API system alarm type removals in version 3.0:

- **webBridgeXmppCertificatePushFailure**
- **xmppAuthenticationRegistrationFailure**
- **xmppRegistrationFailure**
- **recorderLowDiskSpace**
- **guestAccountConnectionFailure**
- **webBridgeBackgroundImageRetrievalFailure**
- **webBridgeBackgroundImagePushFailure**
- **webBridgeLoginLogoImageRetrievalFailure**

- **webBridgeLoginLogoImagePushFailure**
- **webBridgeArchivePushFailure**

Other API removals introduced in 3.0:

Parameters removed on /userProfiles and /userProfiles/<user profile id>:

- **canCreateCoSpaces**
- **canCreateCalls**
- **canUseExternalDevices**
- **canMakePhoneCalls**
- **userToUserMessagingAllowed**
- **canReceiveCalls**
- **canSendEmailInvite**

Parameter removed on /callProfiles

- **messageBoardEnabled**

Parameters removed on /coSpaces/<coSpace id>/coSpaceUsers

- **canPostMessage**
- **canDeleteAllMessages**

Parameter removed on /inboundDialPlanRules

- **resolveToUsers**

2.14.3 API deprecations

Response parameters deprecated in 3.0:

- **activated** on /system/status
- **personalLicenseLimit** on /system/multipartyLicensing
- **sharedLicenseLimit** on /system/multipartyLicensing
- **capacityUnitLimit** on /system/multipartyLicensing

2.14.4 API changes/relocations

Previously the **resolveCoSpaceUris** parameter was on the **/webBridges/<web bridge id>** API object. From version 3.0 this parameter can now be found on the following API objects:

- **/webBridgeProfiles**
- **/webBridgeProfiles/<web bridge profile id>**

- **/webBridges/<web bridge id>/effectiveWebBridgeProfile**
- **/tenants/<tenant id>/effectiveWebBridgeProfile**
- **/system/profiles/effectiveWebBridgeProfile**

Other parameters that were on **/webBridges/<web bridge id>** API object that are now moved or removed in 3.0 are:

- **resourceArchive** – now in webBridgeProfiles
- **idEntryMode** – now deprecated
- **allowWeblinkAccess** – now in webBridgeProfiles as **allowSecrets**
- **showSignIn** – now in webBridgeProfiles as **userPortalEnabled**
- **resolveCoSpaceCallIds** – now in webBridgeProfiles
- **resolveLyncConferenceIds** – now in webBridgeProfiles

2.14.5 Using the new SIP streamer

To set the new SIP streamer, a new API request parameter **sipStreamerUri** that takes the value of a string is added for:

- POST to **/callProfiles**
- PUT to **/callProfiles/<call profile id>**

The **sipStreamerUri** parameter is the SIP streamer dial out URI string.

To find out the SIP streamer URI:

- GET on **/callProfiles/<call profile id>**. The response is structured as a top-level <callProfiles total="N"> tag with potentially multiple <callProfile> elements within it. Each <callProfile> tag may include **sipStreamerUri**.

Note: For a call to be streamable/recordable, there needs to be a call profile set at an appropriate level in the hierarchy and then the effective value of **sipStreamerUri** will be used. As per other fields in the call profile, this parameter can also be overridden.

2.14.6 Using dial-in security profiles to implement minimum passcode length

All **dialInSecurityProfile** parameters are optional. If these parameters are not specified at any level, the default settings will be minPasscodeLength=0 and allowOutOfPolicy=true. The default setting in dialInSecurityProfiles is <unset>.

2.14.6.1 Creating, modifying, and retrieving dial-in security profiles

The new **/dialInSecurityProfiles** object is used to implement dial-in security profiles, with the following request parameters:

Parameters	Type/Value	Description/Notes
name	String	The human-readable name associated with this dial in security profile
minPasscodeLength	Number	Minimum allowed passcode length, can be between 0 and 200 (inclusive)
allowOutOfPolicy	true|false	Whether or not users are allowed to join a call using an old passcode that was set before the dial-in security profile was applied and which is not compliant with the newly defined passcode length. If this parameter is not supplied in a create (POST) operation, it defaults to "true".

This new API node supports the following operations:

- POST to **`/dialInSecurityProfiles`** to create a new dial-in security profile
- PUT on individual profiles with **`/dialInSecurityProfiles/<dial in security profile id>`**
- Enumeration of **`/dialInSecurityProfiles`** accepts the following URI parameters:

URI parameters	Type/Value	Description/Notes
offset		an offset and limit can be supplied to retrieve dial-in security profiles other than the first page in the notional list
limit		
usageFilter	unreferenced|referenced	Supply "usageFilter=unreferenced" in the request to retrieve only those dial-in security profiles that are not referenced by global settings or any other object. This is a useful check before deleting the profile. To retrieve just those dial-in security profiles which are referenced in at least one place, you can supply "usageFilter=referenced"

Response is structured as a top-level <dialInSecurityProfiles total="N"> tag with potentially multiple <dialInSecurityProfile> elements within it.

Each <dialInSecurityProfile> tag may include the following elements:

Response elements	Type/Value	Description/Notes
name	String	The human-readable name associated with this dial-in security profile
minPasscodeLength	Number	Minimum allowed password length, can be between 0 and 200 (inclusive)
allowOutOfPolicy	true|false	Whether or not users are allowed to join a call using an old passcode that was set before the dial-in security profile was applied and which is not compliant with the newly defined passcode length. If this parameter is not supplied in a create (POST) operation, it defaults to "true".

- GET on individual profiles with **/dialInSecurityProfiles/<dial in security profile id>** gives the following response:

Response elements	Type/Value	Description/Notes
name	String	The human-readable name associated with this dial-in security profile
minPasscodeLength	Number	Minimum allowed password length, can be between 0 and 200 (inclusive)
allowOutOfPolicy	true|false	Whether or not users are allowed to join a call using an old passcode that was set before the dial-in security profile was applied and which is not compliant with the newly defined passcode length. If this parameter is not supplied in a create (POST) operation, it defaults to "true".

2.14.6.2 Setting the top level (global) dial-in security profile

The new API parameter **dialInSecurityProfile** allows you to set the top level (global) dial-in security profile to the one specified. This parameter takes the value of an ID and is added for:

- PUT to **/system/profiles**

This operation can take the following request parameter:

Parameter	Type/Value	Description/Notes
dialInSecurityProfile	ID	Sets the top level dial-in security profile to the one specified. Can be unset by supplying "".

- GET on **/system/profiles** gives the following response:

Response value	Type/Value	Description/Notes
dialInSecurityProfile	ID	If present, the configured top level dial-in security profile.

2.14.6.3 Applying a dial-in security profile to a tenant

The new API parameter **dialInSecurityProfile** allows you to apply a specific dial-in security profile to a tenant. This parameter takes the value of an ID and is added for:

- POST to **/tenants**
- PUT to **/tenants/<tenant id>**

  This operation can take the following request parameter:

Parameter	Type/Value	Description/Notes
dialInSecurityProfile	ID	If specified, associates the specified dial-in security profile with this tenant. Can be unset by supplying "".

- GET on **/tenants/<tenant id>** gives the following response:

Parameter	Type/Value	Description/Notes
dialInSecurityProfile	ID	If specified, the specified dial-in security profile associated with this tenant.

2.14.6.4 Applying a dial-in security profile to a coSpace

The new API parameter **dialInSecurityProfile** allows you to apply a dial-in security profile on a coSpace. This parameter takes the value of an ID and is added for:

- POST to **/coSpaces**
- PUT to **/coSpaces/<cospace id>**

This operation can take the following request parameter:

Parameter	Type/Value	Description/Notes
dialInSecurityProfile	ID	If provided, associates the specified dial-in security profile with this coSpace. Can be unset by supplying "".

- GET on **/coSpaces/<cospace id>** gives the following response:

Parameter	Type/Value	Description/Notes
dialInSecurityProfile	ID	If provided, the specified dial-in security profile associated with this coSpace.

2.14.6.5 Applying a dial-in security profile to an access method

The new API parameter **dialInSecurityProfile** allows you to apply a dial-in security profile to an access method. This parameter takes the value of an ID and is added for:

- POST to **/coSpaces/<cospace id>/accessMethods**
- PUT to **/coSpaces/<cospace id>/accessMethods/<access method id>**

This operation can take the following request parameter:

Parameter	Type/Value	Description/Notes
dialInSecurityProfile	ID	If provided, associates the specified dial-in security profile with this coSpace access method. Can be unset by supplying "".

- GET on **/coSpaces/<cospace id>/accessMethods/<access method id>** gives the following response:

Parameter	Type/Value	Description/Notes
dialInSecurityProfile	ID	If provided, the specified dial-in security profile associated with this coSpace access method.

2.14.6.6 Applying a dial-in security profile to a coSpace template

The new API parameter **`dialInSecurityProfile`** allows you to apply a dial-in security profile to a coSpace Template. This parameter takes the value of an ID and is added for:

The **`/coSpaceTemplates`** node supports the following operations:

- POST to **`/coSpaceTemplates`**
- PUT to **`/coSpaceTemplates/<coSpace template id>`**

Parameter	Type/Value	Description/Notes
dialInSecurityProfile	ID	If provided, associates the specified dial-in security profile with this coSpaceTemplate. Can be unset by supplying "".

- Enumeration of **`/coSpaceTemplates`**

  Response is structured as a top-level <coSpaceTemplates total="N"> tag with potentially multiple <coSpaceTemplate> elements within it.

  Each <coSpaceTemplate> tag may include the element: **`dialInSecurityProfile`** which, if provided, displays the dial-in security profile associated with this coSpaceTemplate.

Element	Type/Value	Description/Notes
dialInSecurityProfile	ID	If provided, displays the dial-in security profile associated with this coSpaceTemplate.

- GET on **`/coSpaceTemplates/<coSpace template id>`** gives the following response:

Response value	Type/Value	Description/Notes
dialInSecurityProfile	ID	If provided, the specified dial-in security profile associated with this coSpaceTemplate.

2.14.6.7 Applying a dial-in security profile to an access method template

The new API parameter **`dialInSecurityProfile`** allows you to apply a dial-in security profile to an access method template. This parameter takes the value of an ID and is added for:

- POST to **`/coSpaceTemplates/<coSpace template id>/accessMethodTemplates`**
- PUT to **`/coSpaceTemplates/<coSpace template id>/accessMethodTemplates/<access method template id>`**

This operation can take the following request parameter:

Parameter	Type/Value	Description/Notes
dialInSecurityProfile	ID	If provided, associates the specified dial-in security profile with this access method template. Can be unset by supplying "".

Response is structured as a top-level <accessMethodTemplates total="N"> tag with potentially multiple <accessMethodTemplate> elements within it.

Each <accessMethodTemplate> tag may include the following element:

Response elements	Type/Value	Description/Notes
dialInSecurityProfile	ID	if provided, the specified dial-in security profile associated with this access method template

- GET on **`/coSpaceTemplates/<coSpace template id>/accessMethodTemplates/<access method template id>`** gives the following response:

Response value	Type/Value	Description/Notes
dialInSecurityProfile	ID	If provided, the specified dial-in security profile associated with this access method template.

2.14.7 Using web bridge profiles

Version 3.0 allows you to configure Web Bridge options in a common place within the API and with web bridge profiles you can now apply the same settings for all, or a specified group of Web Bridges, not just on a per Web Bridge basis.

2.14.7.1 Creating, modifying, and retrieving web bridge profiles

The new **`/webBridgeProfiles`** object is used to implement web bridge profiles with the following request parameters:

Parameters	Type/Value	Description/Notes
name	String	The human-readable name associated with this web bridge profile.
resourceArchive	url	The address of any customization archive file that the Meeting Server should use for web bridges using this web bridge profile.

Parameters	Type/Value	Description/Notes
allowPasscodes	true | false	Whether or not web bridges using this web bridge profile should allow users to lookup coSpaces (and coSpace access methods) with passcodes in combination with an numeric ID/URI. If this parameter is not supplied in a create (POST) operation, it defaults to "true".
allowSecrets	true | false	Whether or not web bridges using this web bridge profile should allow users to access coSpaces (and coSpace access methods) through a meeting join link with a numeric id and secret. If this parameter is not supplied in a create (POST) operation, it defaults to "true".
userPortalEnabled	true | false	Whether or not web bridges using this web bridge profile should display the sign in tab on the index page. If this parameter is not supplied in a create (POST) operation, it defaults to "true".
allowUnauthenticatedGuests	true | false	Whether to allow guest access from the landing screen on web bridges using this web bridge profile, or only allow visitor access once users have logged into the User Portal. If false, links work only for logged in users. If this parameter is not supplied in a create (POST) operation, it defaults to "true".
resolveCoSpaceCallIds	true | false	Whether or not web bridges using this web bridge profile should accept coSpace and coSpace access method call IDs for the purpose of allowing visitors to join cospace meetings. If this parameter is not supplied in a create (POST) operation, it defaults to "true".
resolveLyncConferenceIds	true | false	(Currently visible but non-functional.) Whether or not web bridges using this web bridge profile should accept IDs to be resolved to Lync scheduled conference IDs. If this parameter is not supplied in a create (POST) operation, it defaults to "false".

Parameters	Type/Value	Description/Notes
resolveCoSpaceUris	off | domainSugges-tionDisabled | domainSugges-tionEnabled	Whether or not this web bridge should accept coSpace and coSpace access method SIP URIs for the purpose of allowing visitors to join cospace meetings. • when set to 'off' join by URI is disabled • when set to 'domainSuggestionDisabled' join by URI is enabled but the domain of the URI won't be autocompleted or verified on web bridges using this web bridge profile • when set to 'domainSuggestionEnabled' join by URI is enabled and the domain of the URI can be autocompleted and verified on web bridges using this web bridge profile If this parameter is not supplied in a create (POST) operation, it defaults to "off".

This new API node supports the following operations:

- POST to **`/webBridgeProfiles`** to create a new web bridge profile.
- PUT on individual profiles with **`/webBridgeProfiles/<web bridge profile id>`**
- Enumeration of **`/webBridgeProfiles`** accepts the following URI parameters:

URI parameters	Type/Value	Description/Notes
offset		an offset and limit can be supplied to retrieve web bridge profiles other than the first page in the notional list
limit		
usageFilter	unreferenced |referenced	Supply "usageFilter=unreferenced" in the request to retrieve only those web bridge profiles that are not referenced by global settings or any other object. This is a useful check before deleting the profile. To retrieve just those web bridge profiles that are referenced in at least one place, you can supply "usageFilter=referenced"

Response is structured as a top-level <webBridgeProfiles total="N"> tag with potentially multiple <webBridgeProfile> elements within it.

Each <webBridgeProfile> tag may include the following element:

Response elements	Type/Value	Description/Notes
name	String	The human-readable name associated with this web bridge profile

- GET on **`/webBridgeProfiles/<web bridge profile id>`** gives the following responses:

Response values	Type/Value	Description/Notes
name	String	The human-readable name associated with this web bridge profile.
resourceArchive	url	Whether or not web bridges using this web bridge profile should allow users to lookup coSpaces (and coSpace access methods) with passcodes in combination with an numeric ID/URI.
allowPasscodes	true | false	Whether or not web bridges using this web bridge profile should allow users to lookup coSpaces (and coSpace access methods) with passcodes in combination with an numeric ID/URI.
allowSecrets	true | false	Whether or not web bridges using this web bridge profile should allow users to access coSpaces (and coSpace access methods) through a meeting join link with a numeric id and secret.
userPortalEnabled	true | false	Whether or not web bridges using this web bridge profile should display the sign in tab on the index page.
allowUnauthenticatedGuests	true | false	Whether to allow guest access from the landing screen on web bridges using this web bridge profile, or only allow visitor access once users have logged into the User Portal. If false, links work only for logged in users.
resolveCoSpaceCallIds	true | false	Whether or not web bridges using this web bridge profile should accept coSpace and coSpace access method call IDs for the purpose of allowing visitors to join cospace meetings.
resolveLyncConferenceIds	true | false	(Currently visible but non-functional.) Whether or not web bridges using this web bridge profile should accept IDs to be resolved to Lync scheduled conference IDs.

Response values	Type/Value	Description/Notes
resolveCoSpaceUris	off | domainSuggestionDisabled | domainSuggestionEnabled	Whether or not web bridges using this web bridge profile should accept coSpace and coSpace access method SIP URIs for the purpose of allowing visitors to join cospace meetings. • when set to 'off' join by URI is disabled • when set to 'domainSuggestionDisabled' join by URI is enabled but the domain of the URI won't be autocompleted or verified on web bridges using this web bridge profile • when set to 'domainSuggestionEnabled' join by URI is enabled and the domain of the URI can be autocompleted and verified on web bridges using this web bridge profile

2.14.7.2 Creating and modifying a web bridge profile

The new API parameter **`webBridgeProfile`** allows you to associate a Web Bridge with a specified web bridge profile. This parameter takes the value of an ID and is added for:

- POST to **`/webBridges`**
- PUT to **`/webBridges/<web bridge id>`**

This operation can take the following request parameter:

Parameter	Type/Value	Description/Notes
webBridgeProfile	ID	If specified, associates this web bridge with the specified web bridge profile.

- GET on **`/webBridges/<web bridge id>`** gives the following response:

Response value	Type/Value	Description/Notes
webBridgeProfile	ID	If provided, the specified web bridge associated with this web bridge profile.

2.14.7.3 Finding out the web bridge profile currently in effect on a specified web bridge

The new API object **`/webBridges/<web bridge id>/effectiveWebBridgeProfile`** allows you to find out the web bridge profile and its associated values that are currently effective on a

specified web bridge. It supports the following operation:

- GET on **`/webBridges/<web bridge id>/effectiveWebBridgeProfile`** gives the following responses:

Response values	Type/Value	Description/Notes
resourceArchive	url	The address of any customization archive file that Meeting Server should use for this web bridge.
allowPasscodes	true | false	Whether or not this web bridge should allow users to lookup coSpaces (and coSpace access methods) with passcodes in combination with an numeric ID/URI.
allowSecrets	true | false	Whether or not this web bridge should allow users to access coSpaces (and coSpace access methods) through a meeting join link with a numeric id and secret.
userPortalEnabled	true | false	Whether or not this web bridge should display the sign in tab on the index page.
allowUnauthenticatedGuests	true | false	Whether to allow guest access from the landing screen on this web bridge, or only allow visitor access once users have logged into the User Portal. If false, links work only for logged in users.
resolveCoSpaceCallIds	true | false	Whether or not this web bridge should accept coSpace and coSpace access method call IDs for the purpose of allowing visitors to join cospace meetings.
resolveLyncConferenceIds	true | false	(Currently visible but non-functional.) Whether or not this web bridge should accept IDs to be resolved to Lync scheduled conference IDs.

Response values	Type/Value	Description/Notes
resolveCoSpaceUris	off | domainSuggestionDisabled | domainSuggestionEnabled	Whether or not this web bridge should accept coSpace and coSpace access method SIP URIs for the purpose of allowing visitors to join cospace meetings. • when set to 'off' join by URI is disabled • when set to 'domainSuggestionDisabled' join by URI is enabled but the domain of the URI won't be autocompleted or verified on this web bridge • when set to 'domainSuggestionEnabled' join by URI is enabled and the domain of the URI can be autocompleted and verified on this web bridge

2.14.7.4 Applying a web bridge profile to a tenant

The new API parameter **`webBridgeProfile`** allows you to apply a specific web bridge profile to a tenant. This parameter takes the value of an ID and is added for:

- POST to **`/tenants`**
- PUT to **`/tenants/<tenant id>`**

  This operation can take the following request parameter:

Parameter	Type/Value	Description/Notes
webBridgeProfile	ID	If specified, associates the specified web bridge profile with this tenant.

- GET on **`/tenants/<tenant id>`** gives the following response:

Parameter	Type/Value	Description/Notes
webBridgeProfile	ID	If specified, the specified web bridge profile associated with this tenant.

2.14.7.5 Finding out the web bridge profile currently in effect on a specified tenant

The new API object **`/tenants/<tenant id>/effectiveWebBridgeProfile`** allows you to find out the web bridge profile and its associated values that are currently effective on a specified tenant. It supports the following operation:

- GET on **`/tenants/<tenant id>/effectiveWebBridgeProfile`** gives the following responses:

Response values	Type/Value	Description/Notes
resourceArchive	url	The address of any customization archive file that Meeting Server should use as default for web bridges for this tenant.
allowPasscodes	true | false	Whether or not web bridges for this tenant should allow users to lookup coSpaces (and coSpace access methods) with passcodes in combination with an numeric ID/URI.
allowSecrets	true | false	Whether or not web bridges for this tenant should allow users to access coSpaces (and coSpace access methods) through a meeting join link with a numeric id and secret.
userPortalEnabled	true | false	Whether or not web bridges for this tenant should display the sign in tab on the index page.
allowUnauthenticatedGuests	true | false	Whether to allow guest access from the landing screen on web bridges for this tenant, or only allow visitor access once users have logged into the User Portal. If false, links work only for logged in users.
resolveCoSpaceCallIds	true | false	Whether or not web bridges for this tenant should accept coSpace and coSpace access method call IDs for the purpose of allowing visitors to join cospace meetings.
resolveLyncConferenceIds	true | false	(Currently visible but non-functional.) Whether or not web bridges for this tenant should accept IDs to be resolved to Lync scheduled conference IDs.

Response values	Type/Value	Description/Notes
resolveCoSpaceUris	off | domainSuggestionDisabled | domainSuggestionEnabled	Whether or not web bridges for this tenant should accept coSpace and coSpace access method SIP URIs for the purpose of allowing visitors to join cospace meetings. • when set to 'off' join by URI is disabled • when set to 'domainSuggestionDisabled' join by URI is enabled but the domain of the URI won't be autocompleted or verified on web bridges for this tenant • when set to 'domainSuggestionEnabled' join by URI is enabled and the domain of the URI can be autocompleted and verified on web bridges for this tenant

2.14.7.6 Setting the top level (global) web bridge profile

The new API parameter **`webBridgeProfile`** allows you to specify and set the top level (global) web bridge profile. This parameter takes the value of an ID and is added for:

- PUT to **`/system/profiles`**

This operation can take the following request parameter:

Parameter	Type/Value	Description/Notes
webBridgeProfile	ID	Sets the top level web bridge profile to the one specified.

- GET on **`/system/profiles`** gives the following response:

Response value	Type/Value	Description/Notes
webBridgeProfile	ID	If present, the configured top level web bridge profile.

2.14.7.7 Finding out the web bridge profile currently in effect at the top level (global) system level

The new API object **`/system/profiles/effectiveWebBridgeProfile`** allows you to find out the web bridge profile and its associated values that are currently effective on this system. It supports the following operation:

- GET on **`/system/profiles/effectiveWebBridgeProfile`** gives the following responses:

Response values	Type/Value	Description/Notes
resourceArchive	url	The address of any customization archive file that the Meeting Server uses as default for web bridges on this system.
allowPasscodes	true | false	Whether or not web bridges on this system should allow users to lookup coSpaces (and coSpace access methods) with passcodes in combination with an numeric ID/URI.
allowSecrets	true | false	Whether or not web bridges on this system should allow users to access coSpaces (and coSpace access methods) through a meeting join link with a numeric id and secret.
userPortalEnabled	true | false	Whether or not web bridges on this system should display the sign in tab on the index page.
allowUnauthenticatedGuests	true | false	Whether to allow guest access from the landing screen on web bridges for this system, or only allow visitor access once users have logged into the User Portal. If false, links work only for logged in users.
resolveCoSpaceCallIds	true | false	Whether or not web bridges on this system should accept coSpace and coSpace access method call IDs for the purpose of allowing visitors to join cospace meetings.
resolveLyncConferenceIds	true | false	(Currently visible but non-functional.) Whether or not web bridges on this system should accept IDs to be resolved to Lync scheduled conference IDs.

Response values	Type/Value	Description/Notes
resolveCoSpaceUris	off | domainSuggestionDisabled | domainSuggestionEnabled	Whether or not web bridges on this system should accept coSpace and coSpace access method SIP URIs for the purpose of allowing visitors to join cospace meetings. • when set to 'off' join by URI is disabled • when set to 'domainSuggestionDisabled' join by URI is enabled but the domain of the URI won't be autocompleted or verified on web bridges on this system • when set to 'domainSuggestionEnabled' join by URI is enabled and the domain of the URI can be autocompleted and verified on web bridges on this system

2.14.7.8 Retrieving the status of a Web Bridge

The new **`status`** response value type **`dnsFailure`** is introduced for:

- GET on **`/webBridges/<web bridge id>/status`** gives the following response:

Response value	Type/Value	Description/Notes
status	dnsFailure	dnsFailure – the configured web bridge url could not be resolved.

2.14.8 Viewing a historical record of PMP license assignment

A new response parameter **`pmpAssigned`** is introduced for each license usage event:

- GET on **`/system/MPLicenseUsage`** gives the following response:

Response value	Type/Value	Description/Notes
pmpAssigned	numeric	The number of personal licenses assigned to users in the cluster.

2.14.9 Retrieving cluster licensing information

From 3.0, a GET operation on the existing **`/system/licensing`** API now only returns the contents of the license file (i.e. the feature components) on a per Meeting Server instance. The newly introduced API object **`/clusterLicensing`** returns the license status and expiry date (if applicable) for a Meeting Server cluster.

Note: The expiry date field returned for `/clusterLicensing` will only ever be up to a maximum of 90 days in the future.

To retrieve the current license information for your Meeting Server or cluster:

GET method performed on `/clusterLicensing` gives the following:

Response elements	Type/Value	Description/Notes
features		If licensing is enabled then the <features> element includes the elements below.

Response elements	Type/Value	Description/Notes

Name	Type/Value	Description
callBridge		

Name	Type/Value	Description
status	noLicense| activated| expired	Status of the license: • noLicense – no license is available for this feature • activated – the feature is licensed and within its expiry date • expired – the license for this feature is past its expiry date
expiry	String	Date of expiry

Name	Type/Value	Description
callBridgeNoEncryption		

Name	Type/Value	Description
status	noLicense| activated| expired	Status of the license: • noLicense – no license is available for this feature • activated – the feature is licensed and within its expiry date • expired – the license for this feature is past its expiry date
expiry	String	Date of expiry

Name	Type/Value	Description
customizations		

Name	Type/Value	Description
status	noLicense| activated| expired	Status of the license: • noLicense – no license is available for this feature • activated – the feature is licensed and within its expiry date • expired – the license for this feature is past its expiry date
expiry	String	Date of expiry

<table>
<tr><th>Response elements</th><th>Type/Value</th><th>Description/Notes</th></tr>
<tr><td></td><td colspan="2">
<table>
<tr><th>Name</th><th>Type/Value</th><th>Description</th></tr>
<tr><td>recording</td><td colspan="2">
<table>
<tr><th>Name</th><th>Type/Value</th><th>Description</th></tr>
<tr><td>status</td><td>noLicense| activated| expired</td><td>Status of the license: noLicense – no license is available for this feature activated – the feature is licensed and within its expiry date expired – the license for this feature is past its expiry date</td></tr>
<tr><td>expiry</td><td>String</td><td>Date of expiry</td></tr>
</table>
</td></tr>
</table>
</td></tr>
</table>

2.15 Summary of CDR Changes

Version 3.0 introduces the following additions to the Call Detail Records of the Meeting Server:

- **`recorderUrl`** and **`streamerUrl`** are now removed from recordingStart and streamingStart Records, respectively. They are not required by the new SIP recorder and streamer components.
- new parameter **`streamerUri`** added in the streamingStart Record. This is a string and is the URI of the streamer device. (Previously, both **`path`** and **`streamerUrl`** would always be provided however these are not sent for a SIP streamer. There is no change to the recordingEnd record.)

2.16 Summary of MMP additions and changes

Version 3.0 supports these MMP changes:

2.16.1 Image signing

The following MMP commands are introduced in version 3.0.

Table 9: Version 3.0 Image signing MMP command changes and additions

Command	Description
upgrade [<name>]	Existing MMP command – however, it now performs signature and integrity checks before proceeding with upgrading Meeting Server with the specified image. The checks will be carried out even if the **upgrade <name> verify** command has been previously run on that image. Updated from version 3.0.
upgrade <name> verify	Carries out all the integrity and signature checks normally done during upgrade, but does not proceed with the upgrade. This command can also be used to display the image type. Added from version 3.0.
authenticity	Displays all information relating to software authenticity: how the running image was validated (key type and name), and the public keys currently loaded along with their details (type, name and source). It also displays whether the keys are trusted: if a SPECIAL key is installed, whether its signature has been verified with the MASTER key (other keys are internal and always trusted). Added from version 3.0.
authenticity key add <key-file>	Installs a SPECIAL key. Only one SPECIAL key may be installed at a time. Added from version 3.0.
authenticity key none	Removes the SPECIAL key currently installed. This command must be used to remove a key before installing another, or when the key is no longer in use. Added from version 3.0.

2.16.2 SIP Recorder

Table 10: Version 3.0 Recorder MMP command changes, additions, and removals

Command	Description
recorder sip certs	Allows you to configure a SIP certificate. Added from version 3.0.
recorder sip listen <interface> <tcp-port|none> <tls-port|none>	The SIP recorder/streamer components do not need to listen for https connections, however, they do need to listen for SIP connections. This new MMP command is introduced for setting both TCP and TLS. Added from version 3.0.
recorder sip trace <1m|10m|30m|24h|on|off>	Turns on logging of all SIP messages. All SIP messages will be logged on the recorder. Default is "off". You can enable it permanently with "on" or for a fixed time period. Added from version 3.0.

Command	Description
`recorder limit <value|none>`	Sets the recorder limit to allow scalability. This is the limit above which calls are rejected so that call control can fail over to another device. Added from version 3.0.
`recorder listen <a|b|c|d|lo|none [:<port>] allowed list>` `recorder listen a b`	Removed from version 3.0. Sets up the interface(s) and port(s) for the Recorder to listen on. You must enable the service to start listening with the command recorder enable. The default for the optional port argument is 443.
`recorder listen none`	Removed from version 3.0. Stops the Recorder listening.
`recorder certs <keyfile-name> <crt filename> [<crt-bundle>]`	Removed from version 3.0. Provides the name of the key file and .crt file for the Recorder and, optionally, a CA certificate bundle as provided by your CA
`recorder certs none`	Removed from version 3.0. Removes certificate configuration
`recorder trust <crt-bundle|crt-file>`	Removed from version 3.0. Controls which Call Bridge instances are allowed to connect to the Recorder. If the trusted Call Bridge is running on the same server as the Recorder, then issuing the recorder trust command with the name of the Call Bridge public certificate/certificate bundle is sufficient. If the Call Bridge is running on another server, the public certificate/certificate bundle of the Call Bridge must first be copied to the server with the enabled Recorder using SFTP.
`recorder trust none`	Removed from version 3.0. Deconfigures any trust settings.

2.16.3 SIP Streamer

Table 11: Version 3.0 Streamer MMP command changes, additions, and removals

Command	Description
`streamer sip certs`	Allows you to configure a SIP certificate. Added from version 3.0.
`streamer sip listen <interface> <tcp-port|none> <tls-port|none>`	The SIP recorder/streamer components do not need to listen for https connections, however, they do need to listen for SIP connections. This new MMP command is introduced for setting both TCP and TLS. Added from version 3.0.

Command	Description
streamer sip trace <1m|10m|30m|24h|on|off>	Turns on logging of all SIP messages. All SIP messages will be logged on the streamer. Default is "off". You can enable it permanently with "on" or for a fixed time period. Added from version 3.0.
streamer limit <value|none>	Sets the streamer limit to allow scalability. This is the limit above which calls are rejected so that call control can fail over to another device. Added from version 3.0.
streamer sip resolution <audio|720p|1080p>	Sets the maximum resolution that the streamer will do. The default is 720p. If you want to use 1080p we recommend that you increase your transmit SIP call bandwidth to 3,500,000 bits per second to optimize the video quality. Added from version 3.0.
streamer listen <a|b|c|d|lo|none [:<port>] allowed list> **recorder listen a b**	Removed from version 3.0. Sets up the interface(s) and port(s) for the Streamer to listen on. You must enable the service to start listening with the command streamer enable. The default for the optional port argument is 443.
streamer certs none	Removed from version 3.0. Removes certificate configuration
streamer certs <keyfile-name> <crt filename> [<crt-bundle>]	Removed from version 3.0. Provides the name of the key file and .crt file for the Streamer and, optionally, a CA certificate bundle as provided by your CA
streamer trust <crt-bundle|crt-file>	Removed from version 3.0. Controls which Call Bridge instances are allowed to connect to the streamer. If the trusted Call Bridge is running on the same server as the streamer, then issuing the streamer trust command with the name of the Call Bridge public certificate/certificate bundle is sufficient. If the Call Bridge is running on another server, the public certificate/certificate bundle of the Call Bridge must first be copied to the server with the enabled streamer using SFTP.
streamer trust none	Removed from version 3.0. Deconfigures any trust settings.

2.16.4 Removed component MMP commands

All MMP commands associated with the features and components that are removed from Meeting Server in 3.0 are removed as follows:

- H.323 gateway commands (**`h323_gateway`**)
- Web Bridge 2 commands (**`webbridge`**)
- XMPP server commands (**`xmpp`**)
- XMPP multi-domains commands (**`xmpp multi_domain`**)
- XMPP resiliency commands (**`xmpp cluster`**)
- Load Balancer commands (**`loadbalancer`**)
- Trunk commands (**`trunk`**)
- SIP edge commands (**`sipedge`** and edge-related **`callbridge`**)
- Recorder and Streamer commands dependent upon XMPP
- MMP commands applicable to X-series server

For full details of all the removed commands in 3.0, see the MMP Command Guide.

2.16.5 Other MMP changes

All master/slave references in MMP responses are now changed to primary/replica.

2.17 Summary of Event Changes

There are no new Events for version 3.0.

3 Upgrading, downgrading and deploying Cisco Meeting Server software version 3.0.4

This section assumes that you are upgrading from Cisco Meeting Server software version 2.9. If you are upgrading from an earlier version, then Cisco recommends that you upgrade to 2.9 first following the instructions in the 2.9.x release notes, before following any instructions in these Cisco Meeting Server 3.0.4 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 2.9.

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.

Note: Due to the removal of Web Bridge 2 in 3.0, on upgrading to 3.0 you will need to redeploy your Web Bridge to use Web Bridge 3. Likewise, as the old XMPP-dependent recorder and streamer are now replaced in 3.0 with new internal SIP Recorder and Streamer components, you will need to re-deploy your recorder and streamer on upgrade.

3.1 Upgrading to Release 3.0.4

The instructions in this section apply to Meeting Server deployments which 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.

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.

Note:
Meeting Server 3.0 introduces a mandatory requirement to have Cisco Meeting Management 3.0 (or later). Meeting Management handles the product registration and interaction with your Smart Account (if set up) for Smart Licensing support. For more details, see Section 2.2.
As Web Bridge 2 is removed in 3.0, Web Bridge 2 users will need to redeploy their Web Bridge to use Web Bridge 3 for web app support. There is no automatic upgrade migration from Web Bridge 2 to Web Bridge 3. If you have already deployed Web Bridge 3 in version 2.9, you should check your settings after upgrade because they will not be migrated across from the Web Admin or old settings in /webBridges/<webbridge id>.
Additionally, the old XMPP-dependent recorder and streamer are now replaced in 3.0 with new internal SIP Recorder and Streamer components. On upgrading to 3.0 you will need to re-deploy your recorder and streamer.

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_3_0_4_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:
   c1ef736ad4af2dfe83a3855410e742de2a5c8bb2e61db7e4d3ab6595b025074f

   Cisco_Meeting_Server_3_0_4_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:
   1991cfee3f085da830b78734ae772674569d2c0b3113dd675e6120a26fd987cc

   Cisco_Meeting_Server_3_0_4.ova

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

   For vSphere6, hash (SHA-512) for Cisco_Meeting_Server_3_0_4_vSphere-6_0.ova file:
   516c68a19705ebcf56b9f6a4a33696f39038e1b845e48ddc0590254b471304750dc2178a80606af223f85bcb da590a0f156991cdf6433eeb0e9165b3fcfd8c1c

   For vSphere6.5 and higher, hash (SHA-512) for Cisco_Meeting_Server_3_0_4_vSphere-6_5.ova file:
   0bb6727735d9bc8eb6c3a6217d94621948627f0c8d3f086db19f5cf8f77ae803a915e4108f9721553486d673 adf3930057536f0bca51b1d54b5c8025c5f52847

2. To validate the OVA file, the checksum for the 3.0.4 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.
   a. Initiate the upgrade by executing the upgrade command.
      `upgrade`
   b. The Server/ virtualized server restarts automatically: allow 10 minutes for the process to complete.
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, if applicable, you need to re-cluster XMPP:
   a. Pick one node as the XMPP primary, initialize XMPP on this node
   b. Once the XMPP primary 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, if applicable,
   - 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 3.0.4 Deployments

To simplify explaining how to deploy the Meeting Server, deployments are described in terms of three models:

- single combined Meeting Server – all Meeting Server components (Call Bridge, Web Bridge 3, Database, Recorder, Uploader, Streamer and TURN server) are available, the Call Bridge and Database are automatically enabled but the other components can be individually enabled depending upon the requirements of the deployment. All enabled components reside on a single host server.
- single split Meeting Server – in this model the TURN server and Web Bridge 3 are enabled on a Meeting Server located at the network edge in the DMZ, while the other components are enabled on another Meeting Server located in the internal (core) network.
- the third model covers deploying multiple Meeting Servers clustered together to provide greater scale and resilience in the deployment.

Deployment guides covering all three models are available here. Each deployment guide is accompanied by a separate Certificate Guidelines document.

Points to note:

The Cisco Meeting Server 2000 only has the Call Bridge, Web Bridge 3, and database components. It is suited for deployment on an internal network, either as a single server or a cascade of multiple servers. The Cisco Meeting Server 2000 should not be deployed in a DMZ network. Instead if a deployment requires firewall traversal support for external Cisco Meeting Server web app users, then you will need to also deploy either:

- a Cisco Expressway-C in the internal network and an Expressway-E in the DMZ, or
- a separate Cisco Meeting Server 1000 or specification-based VM server deployed in the DMZ with the TURN server enabled.

The Cisco Meeting Server 1000 and specification-based VM servers have lower call capacities than the Cisco Meeting Server 2000, but all components (Call Bridge, Web Bridge 3, Database, Recorder, Uploader, Streamer and TURN server) are available on each host server. The Web Bridge, Recorder, Uploader, Streamer and TURN server require enabling before they are operational.

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 `3.0.4`.
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

Note: Refer to the Cisco Meeting Server web app Important information guide for information on resolved issues affecting web app.

Issues seen in previous versions that are fixed in 3.0.4.

Cisco identifier	Summary
CSCvx04125	In case of an unexpected call drop, automatic reconnect was failing. Users would see an '**unable to reconnect media**' error and a button to rejoin the meeting.
CSCvw98069	User would see the screen flickering severely for some time right after one participant shares content. The flickering is caused by an unexpected restart of media on H264 high profile.
CSCvx19320	An unexpected restart occurs when an activator leaves the space and disconnects all non-activators.
CSCvx47165	ActiveControl is broken 15 minutes after the SIP call has been replaced.

Issues seen in previous versions that are fixed in 3.0.3.

Cisco identifier	Summary
CSCvw69705	Users are unable to login to a web app meeting and get an "Your session has expired" error as Cisco Meeting Serverdoes not properly encode the JWT which causes the web app to not decode it.
CSCvw61466	Web Bridge 3 running on Cisco Meeting Server 2000 does not redirect incoming HTTP connections to HTTPS when`http-redirect` is enabled. This only affects Cisco Meeting Server 2000, not the VM or Cisco Meeting Server 1000 platforms.
CSCvw84107	Web Bridge 3 running on Cisco Meeting Server 2000 software versions 2.9.5, 3.0.2 and 3.1.0 may fail to start if the MMP configuration `http-redirect` is enabled.

Issues seen in previous versions that are fixed in 3.0.2.

Cisco identifier	Summary
CSCvw18292	Cisco Meeting Server experiences packet loss, which recovers after a restart.
CSCvw61501	On Meeting Server 2000 the local time display shows UTC time and does not reflect correct timezone setting.
CSCvw03087	The streamer will tear down the streaming call leg once an internal RTMP queue grows too large. The call will be flow controlled and once it reaches 200 kbps, it will be torn down by the streamer.

Issues seen in previous versions that are fixed in 3.0.1.

Cisco identifier	Summary
CSCvv45836	Meeting Server 3.0.0 recorded mp4 files are not showing total length of video clip and users can't jump forward or back using the seek bar in Microsoft Windows built-in players.
CSCvv84147	Meeting Server 3.0.0 recording fails to start in Space under Tenant.
CSCvv53755	Meeting Server Call object (api/v1/calls/id) is not torn down correctly for DualHomed calls resulting in calls still showing in the /calls API when all participants leave.
CSCvv34916	Meeting Server returns a 400 error when a GET is done on /api/v1/system/MPLicenseUsage. Meeting Management uses this API and if it receives a 400 error it raises an "Error seen updating license usage for cluster CMS" alarm.
CSCvv17118	On Cisco Meeting Server, for audio only calls, the join tone may be cached and can play in a delayed manner, i.e. prompted by other operations, e.g. another participant joins / leaves the conference.

Cisco identifier	Summary
CSCvv02453	When a Skype client loses network connection in the middle of a call to Meeting Server, the call should be disconnected in 60 sec after Meeting Server stops receiving media from the Skype client side. However, the call remains connected on the Meeting Server until the SIP session timer is reached.
CSCvu60026	When a call transfer occurs to a Cisco Meeting Server space, the display name may not update on the Web Admin to the correct name/number of the person who was transferred in. The display name may still show as the person who conducted the transfer.
CSCvu48740	When a participant is added to a meeting with the call parameter confirmation=true, they need to press 1 to confirm joining the meeting. However, when the meeting is being recorded when the participant is initially added, they are prompted to press 1, but they are being recorded before they actually join the meeting.

Issues seen in previous versions that are fixed in 3.0.

Cisco identifier	Summary
CSCvu70860	Cisco Meeting Server may exit and drop calls if ICE and Multistream are used together in a call.
CSCvt92631	The qualityMain/qualityPresentation assigned at cospace level may cause the Meeting Server to send incorrect SDP in reINVITE if the cospace is passcode protected.
CSCvt23261	Downloading or uploading certificate related files via SFTP on Meeting Server may fail.
CSCvu45771	When Meeting Server is under heavy load the audio prompt fails to play for the end user (IVR / Passcode entry / only participant / etc.).
CSCvu30182	The Recorder is listed as a participant in the participant list on Cisco Meeting Server web app.
CSCvu83901	RTMP stream ends after 2.5 hours (around 2GB size) due to "Send buffer limit reached".
CSCvt29547	On occasions, Meeting Server may fail to: create backups / start pcaps / generate debug files / collect logbundles, as the /secure partition is reporting at 100%.
CSCvm17422	IVR does not play any prompt if a participant dials into a TMS inactive space via IVR.
CSCvt74060	Web Bridge 3 issues the following warnings on call join: "sendRequest() failure - cannot find WB3 websocket connection". This log message doesn't have any serious implications and can be ignored. [TBC]
CSCvt74047	The API **`/api/v1/webbridges/<webbridge id>/status`** always returns **`connectionFailure`**, even when its connection to a Call Bridge is working correctly.

Cisco identifier	Summary
CSCvt74045	If you explicitly activate a participant into a locked meeting by posting deactivated=false to the participant API node and then unlock the meeting, that participant doesn't hear the expected prompt "this meeting is now unlocked".
CSCvt74035	If Web Bridge 3 is not started, it is not shown up in either the "Recent errors and warnings" or "Fault conditions" sections.
CSCvv19066	In some cases Meeting Server incorrectly reads the far end's maximum H.264 video bit rate limit which can result in a slightly reduced video rate to that remote system.

4.2 Open issues

Note: Refer to the Cisco Meeting Server web app Important information guide for information on open issues affecting web app.

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

Cisco identifier	Summary
CSCvv19087	The detailed tracing page option "Web Bridge connection tracing" on the Web Admin UI is still visible but now non-functional – it was used for the Web Bridge 2 component which is now removed from Meeting Server.
CSCvt11301	Web Bridge 3 cannot start if Web Bridge 2 or Webadmin are listening on the same https port number even if on different interfaces.
CSCvt74033	When content is being shared and an event happens to trigger a Webex Room Panorama to drop from sending two video streams to one, the video frame rate being received by a remote endpoint from the Room Panorama can drop noticeably.
CSCvt52420	The mediaProcessingLoad parameter returned in the system/load API on Meeting Server does not correctly account for calls using VP8 codec. When using VP8, there may be a higher actual media load on the Meeting Server than the API reports.
CSCvn65112	For locally hosted branding, if the audio prompt files are omitted then the default built-in prompts are used instead. To suppress all audio prompts use a zero-byte file, rather than no file at all.
CSCvm56734	In a dual homed conference, the video does not restart after the attendee unmutes the video.
CSCvj49594	ActiveControl does not work after a hold/resume when a call traverses Cisco Unified Communications Manager and Cisco Expressway.

Cisco identifier	Summary
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`.
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.
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.

5 Related user documentation

The following sites contain documents covering installation, planning and deployment, initial configuration, operation of the product, and more:

- Release notes (latest and previous releases): https://www.cisco.com/c/en/us/support/conferencing/meeting-server/products-release-notes-list.html
- Install guides (including VM installation, Meeting Server 2000, and using Installation Assistant): https://www.cisco.com/c/en/us/support/conferencing/meeting-server/products-installation-guides-list.html
- Configuration guides (including deployment planning and deployment, certificate guidelines, simplified setup, load balancing white papers, and quick reference guides for admins): https://www.cisco.com/c/en/us/support/conferencing/meeting-server/products-installation-and-configuration-guides-list.html
- Programming guides (including API, CDR, Events, and MMP reference guides and customization guidelines): https://www.cisco.com/c/en/us/support/conferencing/meeting-server/products-programming-reference-guides-list.html
- Open source licensing information: https://www.cisco.com/c/en/us/support/conferencing/meeting-server/products-licensing-information-listing.html
- Cisco Meeting Server FAQs: https://meeting-infohub.cisco.com/faq/category/25/cisco-meeting-server.html
- Cisco Meeting Server interoperability database: https://tp-tools-web01.cisco.com/interop/d459/s1718

6 Accessibility Notice

Cisco is committed to designing and delivering accessible products and technologies.

The Voluntary Product Accessibility Template (VPAT) for Cisco Meeting Server is available here:

http://www.cisco.com/web/about/responsibility/accessibility/legal_regulatory/vpats.html#telepresence

You can find more information about accessibility here:

www.cisco.com/web/about/responsibility/accessibility/index.html

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