Cisco Meeting Server

Cisco Meeting Server 3.5

Customization Guidelines

May 17, 2024

Contents

Change History 5

1 Introduction 6
1.1 What's new or changed in Cisco Meeting Server 3.4 that affects customization 7
1.2 What are the ways to manage branding files? 8
1.3 What can be branded? 8
1.4 Web Server Requirements 8
1.5 Using this guide 9

2 Cisco Meeting Server web app customization 10
2.1 File Properties and Names 13
2.1.1 Sign-in page background image, icon and logo 14
2.1.2 text_strings.json file parameters 15
2.2 Overview of customization procedure (web server hosted branding) 15

3 Call customization 17
3.1 IVR Message customization 17
3.1.1 IVR Messages to be customized 17
3.1.2 Recording format for IVR messages 18
3.1.3 IVR background image properties 18
3.1.4 Overview of customization procedure (web server hosted branding) 18
3.2 SIP/Lync Call Message Customization 19
3.2.1 Audio messages to be customized 19
3.2.2 Recording format for audio messages 24
3.2.3 Background image properties 25
3.2.4 Overview of customization procedure (web server hosted branding) 25
3.2.5 Missing and invalid files 26

4 Invitation text customization 29
4.1 Cisco Meeting Server web app Custom Email Invites 29
4.1.1 Types of web app invitation templates 29
4.2 Creating the invitation template text file 31
4.2.1 Custom email invitation content created with new syntax 31
Header 31
Format 31
Variables 32

Loops 35
Conditions 35
Logical operations 35
Functions 35
Whitespace Control 37
4.2.2 Example invitation template 37
4.3 Overview of customization procedure (web server hosted branding) for web app using Web Bridge 3 39

5 Implementing locally hosted branding 40
5.1 Web app customizations 40
5.2 IVR Message, SIP/Lync Call Message and Invitation Text customization 41
5.3 Testing customized invitation_template*.txt 42
5.4 Removing locally hosted branding files 42

6 Implementing remote branding using API 43
6.1 Setting up the web server 43
6.2 Using the API for branding 45
6.2.1 Applying branding to web app 47
6.2.2 Applying branding to SIP calls and/or customized invitations 47
6.2.3 Applying branding to an IVR 48

7 Switching between branding methods 50
7.1 Changing from web server (or default) branding to locally hosted branding 50
7.2 Changing from locally hosted to web server branding 50
7.3 Mixing locally hosted and web server customization 51
7.4 Limitations of using locally hosted branding 51

Cisco Legal Information 52

Cisco Trademark 53

Figures:

Figure 1: Overview of guides covering the Meeting Server 7
Figure 2: web app assets with branding files 11
Figure 3: Creating zip file for web app assets 14
Figure 4: Example contents of text_strings.json 15

Figure 5: Accessing the API via the Meeting Server web interface 46

Change History

Date	Change Summary
April 20, 2022	New version for 3.5.
December 15, 2021	New version for 3.4.
August, 24, 2021	New version for 3.3.
May 25, 2021	Cisco Meeting Server web app customization updated with a note for brand_tag_line parameter.
April 09, 2021	New version for 3.2.
February 04, 2021	Cisco Meeting Server web app customization updated with recommended specifications for sign-in page background image.
November 30, 2020	New version for 3.1.
July 29, 2020	New version for 3.0. Cisco Meeting Server web app customization/branding introduced. WebRTC app support removed.

1 Introduction

The Cisco Meeting Server software can be hosted on specific servers based on Cisco Unified Computing Server (UCS) technology or on a specification-based VM server. Cisco Meeting Server is referred to as the Meeting Server throughout this document.

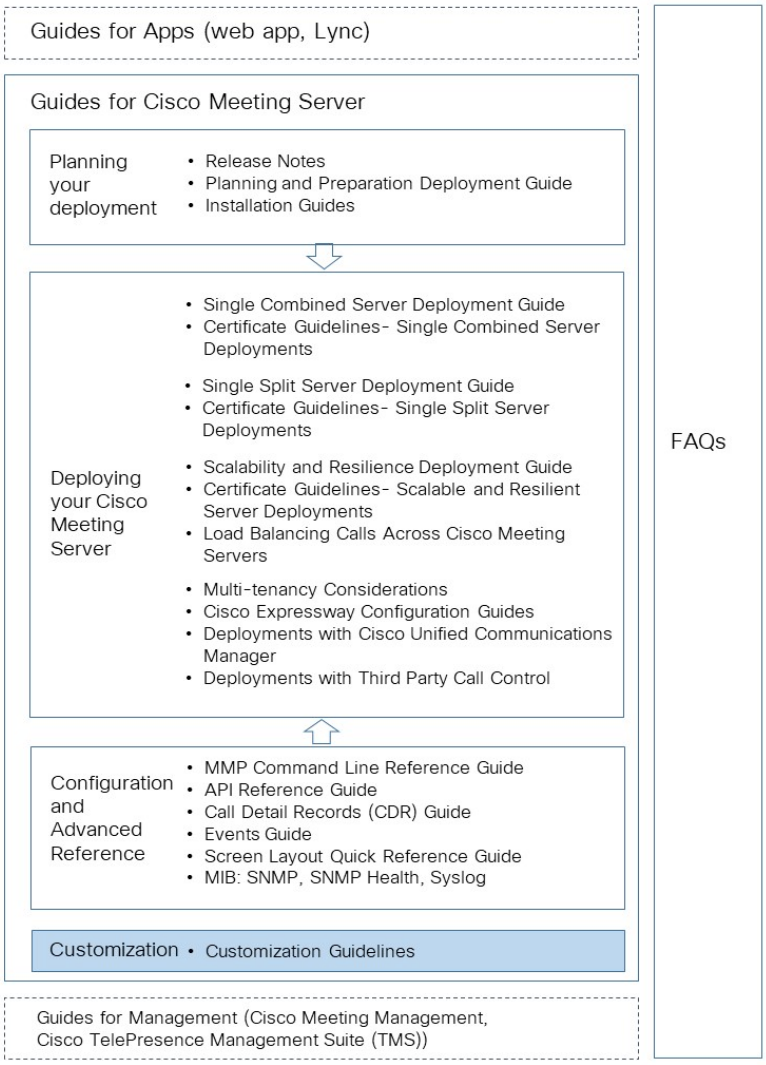
Note: Cisco Meeting Server software version 3.0 onwards does not support X-Series servers.

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.

This document covers the features for customizing release 3.5 of the Meeting Server software. These features allow modification of the voice prompts, background images and logos, and the text shown in invitations.

It follows on from the appropriate Deployment Guide (see Figure 1).

Figure 1: Overview of guides covering the Meeting Server



1.1 What's new or changed in Cisco Meeting Server 3.4 that affects customization

A new audio prompt file **cospace_blast_dial_join_confirmation.wav** is included to guide the participants with DTMF key options to accept or reject the call.

A new variable **`organizername`** is added in the email invitation template. This variable is added in the email invitation template included with Meeting Server. If you are using a customized template, the following variable should be added in the **invitation_template_xx_XX.txt** file to include organizer's name in the email invite.

```
#if organizername
Organisator: %organizername%
```

```
#endif
```

1.2 What are the ways to manage branding files?

There are two options for managing branding files:

- Store one set of branding files locally on the Meeting Server.
- Store branding files on a remote web server, see Web Server Requirements.

1.3 What can be branded?

If you are using a web server to store your branding files, then the list below can be branded. If you have chosen locally hosted branding, see Section 7.4 for limitations which apply.

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

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

If you apply a single brand with only a single set of resources specified (one web app sign-in page, one set of voice prompts, one invitation text), then these resources are used for all spaces, IVRs and Web Bridges in the deployment. Multiple brandings allow different resources to be used for different spaces, IVRs and Web Bridges. Resources can be assigned at the system, tenant, space or IVR level using the API.

1.4 Web Server Requirements

If you intend to use remote branding, you will require a directory on a web server on which .wav, jpg, png or archive (e.g. zip) files can be stored. The web server must be reachable by the Call Bridge, and must not require the Call Bridge to perform any form of HTTP authentication. For more information on setting up the web server, see Section .

The import occurs when the Call Bridge first needs to use the customized files.

Note: If you require multiple brandings, for example one per tenant, then you need to deploy a separate Web Bridge for each branding. Each can be a standalone Web Bridge on a VM server.

1.5 Using this guide

Chapter 2 covers branding the web app. It details the elements that can be branded on the web app and explains how to customize the app with your branding.

Chapter 3 covers branding the images and messages that participants see and hear when dialing into an IVR or dialing directly into a space using a SIP endpoint or Lync.

Chapter 4 describes customizing the text shown in invitations which is sent to participants inviting them to join a call or space.

Chapter 5 describes the procedure to implementing customization from locally hosted branding files.

Chapter 6 provides a step-by-step procedure example on customizing calls using the API.

Chapter 7 provides instructions to switch from one method of branding to another and also limitations

---

Note: All information applies to web server branding and locally hosted branding unless stated otherwise.

---

2 Cisco Meeting Server web app customization

From version 3.0, you can customize and brand for your Cisco Meeting Server web app sign-in page.

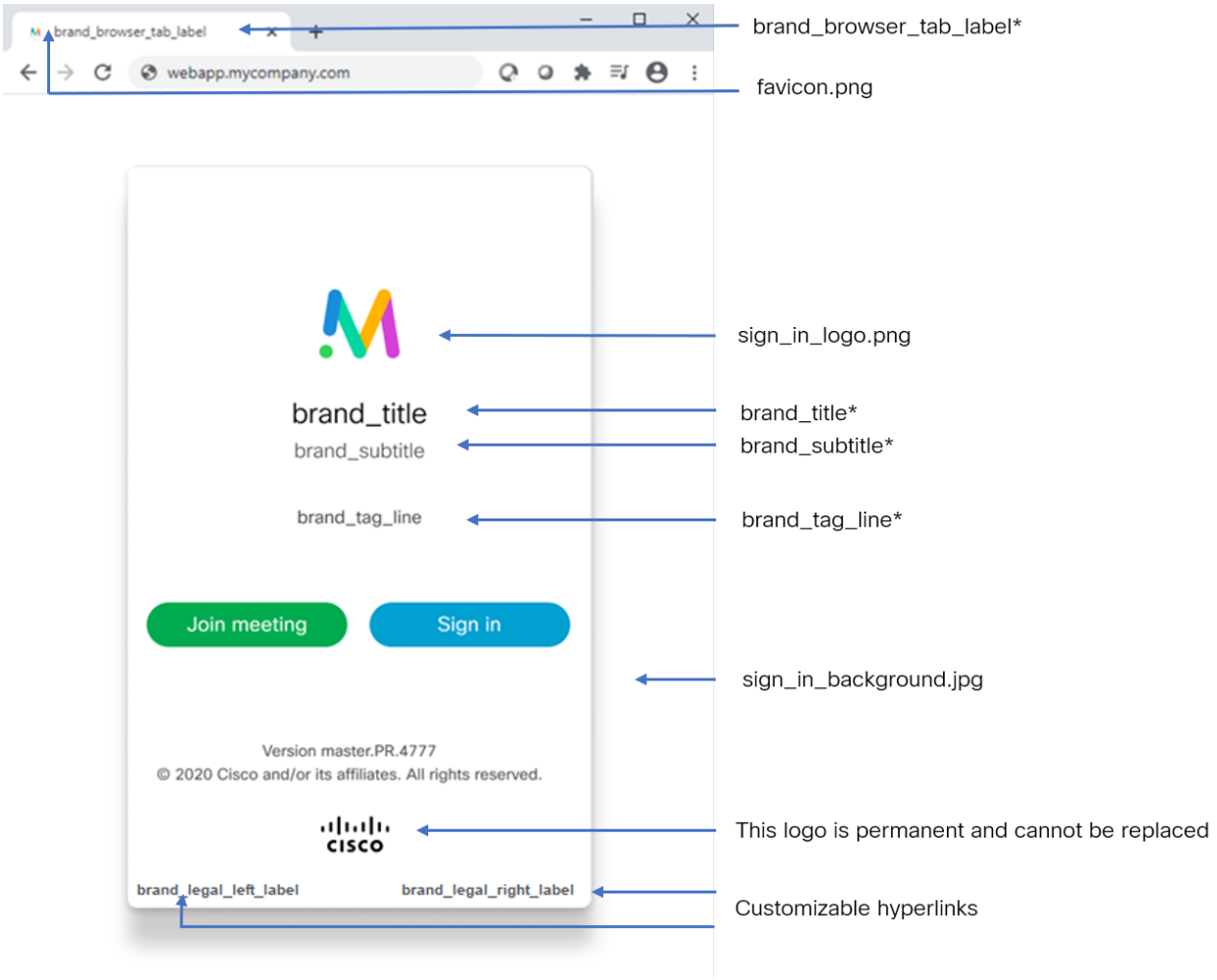
Note: You cannot use a previous Cisco Meeting App for WebRTC branding zip file, you will need to create and deploy a new branding zip file specifically for web app. However, the branding zip file is deployed for web app in the same way as previously for the WebRTC app. (Note that resourceArchive is now 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
- custom hyperlinks
- custom virtual background images

The positioning and location of these elements is shown in Figure 2 and 2.

Figure 2: web app assets with branding files



all these strings are contained within the single text_strings.json file (see Table 1).

Table 1 describes the files that can be uploaded to customize web app as shown in Figure 2 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 1.

The web app user interface is localized in 21 languages. The default web app user interface language is based upon the browser's default. If a web app user then wishes to select a different language, they can then do so before signing in to the app or joining a meeting. For more information on how web app users select their preferred language, see Cisco Meeting Server 3.1(or later) web app Important Information.

To support this new feature, Cisco Meeting Server has a new web app asset: " text_strings_ xx_XX.json" as described in Table 1. These languages are part of the software and require no additional configuration unless you want to customize a particular language file. If a specific language file doesn't exist, Meeting Server will default to using the text_strings.json file.

Table 1: 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
sign_in_ background.jpg	The background shown on the landing 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

File name	Description	Max file-size	Recommended sizes, formats and aspect ratios
text_strings_xx_XX.json (from version 3.1)	Text strings for a particular language; for example " text_strings_fr_CA.json" will offer the web app user interface in French Canadian. Supports the same as text_strings.json. Any text strings defined in this format will override those specified in " text_strings.json" for the specified language. 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 • brand_legal_left_label: left side hyperlink name • brand_legal_left_link: left side hyperlink to any legal information that the users need to know before signing in. • brand_legal_right_label: right hyperlink name • brand_legal_right_link: right side hyperlink to any legal information that the users need to know before signing in.	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 • brand_legal_left_label: up to 70 characters for English and up to 50 characters for other languages • brand_legal_right_label: up to 70 characters for English and up to 50 characters for other languages

For remote web server hosted branding, the branding files are held within an archive (zip) file stored on the web server. The location of this resourceArchive is set via the API – if using the Web Admin user interface see Section 6.2.1, (or if using a third-party application, set via a POST method to the /webBridgeProfiles node or a PUT to a /webBridgeProfiles/<web bridge profile id>.)

The Call Bridge retrieves the archive file from the web server and pushes it to the Web Bridge to be served. In the event of a failure (for example, if the configured URI can't be reached) an alarm is displayed in the Web Admin Interface and on the API /system/alarms node, but users can still log in using the web app.

For locally hosted branding information on where and how to locate the zipped branding files, see Chapter 5.

2.1 File Properties and Names

The branding files must be placed together in an archive file such as a zip file. The total file size of the zipped file must be less than 1 MB, and the maximum decompressed size of any one file

is 512 KB. For web server branding, the zip file is retrieved by the Call Bridge and then used by the Web Bridge to brand the web app. For locally hosted branding, the zip file is retrieved locally by the Web Bridge to brand the web app.

Note: When you zip the files, do not zip the folder containing the branded files. If this is done, this will create an extra layer of folder (zipped file > folder > branded files). Instead, highlight the branded files and right-click to zip them (or open a zip application and zip the files together). This will create a zipped file with the branded files without creating an extra layer of folder (e.g. zipped file > branded files).

For example, in Figure 3, webapp_client.zip folder contains the branded files so when configuring for Web Bridge customization for remote branding, you can use the path http://<webserver address>/Branding/webapp_client.zip.

Figure 3: Creating zip file for web app assets

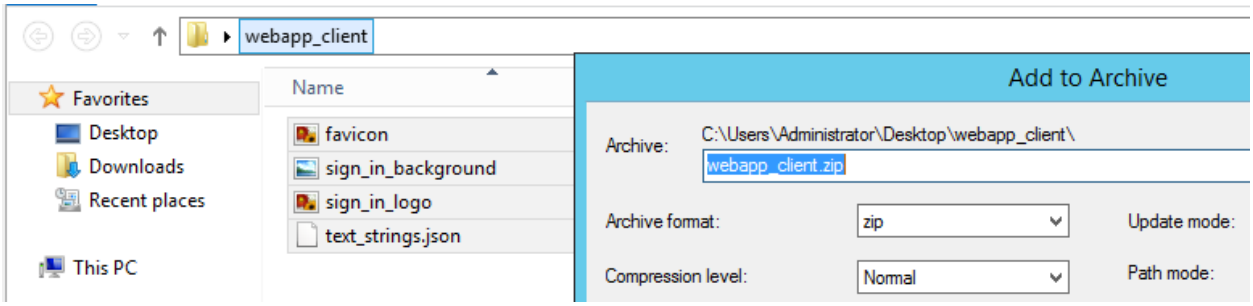


Table 2: web app assets

Use	Filename to use (filenames are case sensitive)
Sign in page background image	sign_in_background.jpg
Sign in page logo	sign_in_logo.png
Icon shown next to the browser tab label, and on any bookmarks / shortcuts	favicon.png
Text below sign in logo and text on browser tab	text_strings.json

2.1.1 Sign-in page background image, icon and logo

The background image must be in .jpg format and the logo file must be in .png format. Transparency in .png files is supported, and recommended, for the logo.

Maximum size for the background image is restricted by the 500kb size limit. It is recommended to use images up to 1920x1080 pixels in resolution, although other resolutions will work as long as the max size is not exceeded.

The recommended resolution for the logo image is 254 pixels wide by 64 pixels high; these dimensions fill the space to the edges of the Sign in box horizontally. This file must be less than 250 kB in size.

2.1.2 text_strings.json file parameters

The text_strings.json file is a JSON formatted file of text strings that can be overwritten to suit your own requirements. The supported strings are listed in Table 3.

Table 3: text_strings.json parameters

Parameter	Description
brand_title	Main brand name
brand_subtitle	Secondary text below
brand_tag_line	Tertiary text below
brand_browser_tab_label	The name of the tab in the browser

You can customize each of these text strings as shown in the example in Figure 4.

Figure 4: 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"
}
```

Note: The text_strings.json file must use straight quotes. If your system uses smart quotes by default, you must change the setting.

Note: If you are testing your branding, you need to refresh the page twice (or use CTRL+F5 on Google Chrome) as the browser does not actually re-fetch images on a single refresh.

Note: If the user portal is disabled, the text added in the parameter brand_tag_line will not be displayed.

2.2 Overview of customization procedure (web server hosted branding)

These steps give an overview of the customization procedure, for a detailed procedure see Chapter 6.

1. Create a zip archive file containing these files:
   - favicon.png
   - sign_in_background.jpg
   - sign_in_logo.png
   - text_strings.json
2. On the web server create a directory for the customization files and place the zip file in it.
3. Using the API, create a webBridgeProfile node specifying the resourceArchive field and URL on the web server where the zip file is held and apply to a webBridge node. (See Chapter 6 for details). The Meeting Server uses these details to retrieve the zip file from the web server.

Note: You can only apply one archive file per Web Bridge.

Note: If you specify a port value in the path, this will be used in place of the default port values of :80 for http and :443 for https. If you do not specify a port value then the default value will be used. If you specify a port value make sure that the web server used in step 2 is listening on this port.

---

3 Call customization

There are two types of call customization:

- Messages heard and image seen when dialing into an IVR
- Messages heard and images seen when dialing directly as a SIP (including Lync) call into a space on a Meeting Server

3.1 IVR Message customization

The messages heard when calling an IVR can be customized via the API by using the ivrBrandingProfile. This profile can be applied at the system level, on a per-tenant basis or for individual IVRs.

Note: The call branding files are not retained after a Meeting Server upgrade and, hence, must be uploaded again after the upgrade.

3.1.1 IVR Messages to be customized

The following table lists all the IVR messages required for and the associated filenames to use for the recordings.

Table 4: IVR messages for customization

Text of message	Filename to use (filenames are case sensitive)	Played when .....
Please enter the call ID, followed by the '#'(pound) key.	ivr_id_entry.wav	dialling via IVR to enter a specific space
Unable to recognize that call ID. Please try again.	ivr_id_incorrect_try_again.wav	the incorrect call ID is entered to join the space
Please try again: this is your last attempt.	ivr_id_incorrect_final_attempt.wav	two incorrect pins/call ID's have been entered to join the space
Unable to recognize that call ID. Goodbye.	ivr_id_incorrect_goodbye.wav	entering three incorrect call ID's to join the space
Welcome to a Cisco meeting.	ivr_welcome.wav	joining a space
Unable to connect you. Goodbye.	ivr_timeout.wav	after dialling via IVR and not entering the call ID, the call times out

3.1.2 Recording format for IVR messages

Use audio files created with Audacity and saved as a single track PCM16, at 8, 16, 22.05, 32, 44.1 or 48 kHz sample rate, 16 bits per sample and mono. The file size for each recording must be less than 400kB.

Note: It is not necessary to include any additional periods of silence at the start or end of these prompts.

3.1.3 IVR background image properties

The background image file must be .jpg format, less than 500kB in size and a maximum of 1920 pixels wide by 1200 pixels. Images will be centered, scaled and padded with black to preserve their aspect ratio. Progressive JPEG is not supported.

Table 5: IVR background image for customization

Image use	Filename to use (filenames are case sensitive)
IVR background image	ivr_background.jpg

3.1.4 Overview of customization procedure (web server hosted branding)

The following steps provide an overview of the customization procedure, for a detailed procedure refer to Chapter .

Note: If using web server hosted branding, ensure that your web server is configured correctly with the proper MIME type for the WAV and JPEG file extensions (e.g. audio/wav and image/jpeg MIME type).

1. On the web server create a directory for the IVR customization files and place in it all the files listed in Section 3.1.1 and Section 3.1.3. Do not zip these files

   Cisco have created an archive with their audio and video files, the zip files are downloadable from the Customization subsection on this web page. The files can be individually used when a mix of custom and default files are required. The necessary files must be copied into the directory. If any files are missing then no alternative or default file will be used in their place.
2. Using the API, create an ivrBrandingProfile specifying the resourceLocation field and URL on the web server where the customized ivr files are held. (See Section for details). The Meeting Server uses these details to retrieve the files from the web server.
3. Use this ivrBrandingProfile to customize your calls on:
   a. a system level (that is, as part of the global systems profile)
   b. a per-tenant basis (as part of the tenant definition)

c. a per-IVR basis (as part of an IVR definition)

Note: When IVR branding is applied at more than one level, then the lowest level ivrBrandingProfile defined is the one that it used in a call leg. Only the ivrBrandingProfile applied at the system level is used for single branding. See the Using Profiles section in the API Reference.

3.2 SIP/Lync Call Message Customization

This section describes how to customize the in-call experience when dialing into a Meeting Server space from a SIP or Lync endpoint.

For web server hosted branding, the files are retained on the Call Bridge while they are in continuous use for one or more calls. If you change the background image or the audio messages, then the new images and audio files will not be utilized until any calls utilizing call branding have completed, and the files have been re-fetched.

3.2.1 Audio messages to be customized

The following list is all the audio messages required for and the associated filenames to use for the recordings.

Table 6: SIP call messages for customization

Text of message	Filename to use (filenames are case sensitive)	Repeats for audio calls	Played when .....
Welcome to a Cisco meeting	welcome.wav	No	joining a call
I haven't been able to connect you. Goodbye.	timeout.wav	No	after dialling via an IVR and not entering the call id, the call times out
Hello. You are invited to a Cisco call.	call_outgoing_welcome.wav	No	when **`confirmation`** is set to **`True`** in the API, " call_" prompts are played from a call that isn't associated with a cospace
Press '1' to join the call.	call_join_confirmation.wav	No	after the " Hello. You are invited to a Cisco call." prompt is played
You are joining the call now.	call_join.wav	No	after pressing '1' to join the call
You are invited to enter a Cisco meeting.	cospace_outgoing_welcome.wav	No	when **`confirmation`** is set to **`True`** in the API, " cospace_" prompts are played in configured meetings (e.g. cospace meetings)
Press '1' to enter the meeting.	cospace_join_confirmation.wav	No	after the " Hello. You are invited to a Cisco meeting." prompt is played
You are entering the meeting now.	cospace_join.wav	No	after dialling the URI of a space from a SIP endpoint, or after dialling the IVR and entering the call ID of the space (plays after PIN if space has a PIN)
Press '1' to enter the meeting or '*' to hang up	cospace_blast_dial_join_confirmation.wav	No	the prompt is played during a blast dial call
You have been disconnected from the meeting	disconnected.wav	No	the participant has been disconnected from the meeting.
This meeting is locked, you are waiting to be allowed in	locked_you_are_waiting.wav	No	the participant is waiting to be activated so they can enter the meeting (from version 2.9)
This meeting is being recorded	meeting_recorded.wav	No	recording starts or when joining a call that is being recorded

Text of message	Filename to use (filenames are case sensitive)	Repeats for audio calls	Played when .....
This meeting is no longer being recorded	meeting_recording_ended.wav	No	recording ends
This meeting is being streamed	meeting_streamed.wav	No	streaming starts (from version 2.1)
This meeting is no longer being streamed	meeting_streaming_ended.wav	No	streaming stops (from version 2.1)
Your meeting has ended	meeting_ended.wav	No	the meeting has ended.
The meeting is unlocked	meeting_unlocked.wav	No	the meeting state changes from locked to unlocked. This will play to all non-lobby participants (from version 2.8).
The meeting is locked	meeting_locked.wav	No	the meeting state changes from unlocked to locked. This will play to all non-lobby participants (from version 2.8).
You are the only participant	only_participant.wav	Yes	there is only one participant in the meeting.
Please enter the PIN, followed by the '#'(pound) key.	passcode_entry.wav	No	a PIN is required to enter the space
If you have a passcode, please enter it, followed by '#' pound. Otherwise please press '#' pound.	passcode_or_blank_required_entry.wav	No	a passcode is required for the host to enter the space as host, but guests only need to use the # (pound) key (from version 2.1)
If you have a passcode, please enter it, followed by '#' pound. Otherwise please wait.	passcode_or_blank_timeout_entry.wav	No	a passcode is required for the host to enter the space as host, but guests join after a short timeout (from version 2.1)

Text of message	Filename to use (filenames are case sensitive)	Repeats for audio calls	Played when .....
That PIN isn't correct. Please try again.	passcode_incorrect_try_ again.wav	No	the incorrect PIN is entered to join the space
Please try again: this is your last attempt.	passcode_incorrect_final_ attempt.wav	No	two incorrect PINs/call ID's have been entered to join the space
That PIN isn't correct. Goodbye.	passcode_incorrect_ goodbye.wav	No	after entering three incorrect PINs to join the space
There is one participant in the meeting.	total_participant_1.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are two participants in the meeting.	total_participant_2.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting(from version 2.8)
There are three participants in the meeting.	total_participant_3.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are four participants in the meeting.	total_participant_4.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are five participants in the meeting.	total_participant_5.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are six participants in the meeting.	total_participant_6.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are seven participants in the meeting.	total_participant_7.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)

Text of message	Filename to use (filenames are case sensitive)	Repeats for audio calls	Played when .....
There are eight participants in the meeting.	total_participant_8.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are nine participants in the meeting.	total_participant_9.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are ten participants in the meeting.	total_participant_10.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are eleven participants in the meeting.	total_participant_11.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are twelve participants in the meeting.	total_participant_12.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are thirteen participants in the meeting.	total_participant_13.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are fourteen participants in the meeting.	total_participant_14.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are fifteen participants in the meeting.	total_participant_15.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are sixteen participants in the meeting.	total_participant_16.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are seventeen participants in the meeting.	total_participant_17.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)

Text of message	Filename to use (filenames are case sensitive)	Repeats for audio calls	Played when .....
There are eighteen participants in the meeting.	total_participant_18.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are nineteen participants in the meeting.	total_participant_19.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are twenty participants in the meeting.	total_participant_20.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
There are more than twenty participants in the meeting.	total_participant_greater_than_20.wav	No	after entering the DTMF command configured to play the number of participants actively participating in the meeting (from version 2.8)
Waiting for your host to join	waiting_for_host.wav	Yes	while waiting for the meeting host to join

Note: The voice prompts labeled as repeating are played repeatedly for audio callers. They are played a single time for video callers. To prevent voice prompts that are repeated from sounding repetitive, record a quiet period (2 to 3 seconds, or longer if you want them to repeat less frequently) after the voice.

Note: participant_join.wav and participant_leave.wav are joining and leaving tones respectively, to indicate that a participant has joined or left the space; neither tone can be customized.

Note: If the format of a replacement audio file is invalid, then nothing will be played.

3.2.2 Recording format for audio messages

If you plan to record your own audio messages, then save each prompt as type “WAV (Microsoft) signed 16 bit PCM”. Audio files can be converted into the correct format using Audacity. The project rate should be one of 8000, 16000, 22050, 32000, 44100, 48000 Hz.

If creating files using another application, the output must be:

- single track PCM format
- 16 bits per sample
- Mono
- 8, 16, 22.05, 32, 44.1 or 48 kHz sample rate

For recordings that repeat, the file size must be less than 1000kB. This is sufficient for 32 seconds when using a 16 kHz sample rate. The file size for all other recording must be less than 400kB

Note: We have not tested other audio tools and, if used, they may cause problems.

Note: It is not necessary to include any additional periods of silence at the start or end of the prompts that do not repeat. For those that repeat, silence can be used to create a gap between the instances.

3.2.3 Background image properties

The background file must be in .jpg format, less than 500kB in size and a maximum of 1920 pixels wide and 1200 pixels high. Images will be centered, scaled and padded to preserve their aspect ratio. Progressive JPEG is not supported.

Table 7: SIP call background image for customization

Image use	Filename to use (filenames are case sensitive)
Call logo	background.jpg
Screen can be shown when a PIN needs to be entered.	passcode_background.jpg
Screen can be shown when a PIN is required for the host to enter the space as host, but guests only need to use the # (pound) key (from version 2.1)	passcode_or_blank_required_background.jpg
Screen can be shown when a PIN is required to enter the coSpace as host, but guests join after a short timeout (from version 2.1)	passcode_or_blank_timeout_background.jpg
Screen can be shown when awaiting activation, (waiting for host to join, meeting needs to be unlocked etc).	deactivated_background.jpg

Note: The behaviour and choice of background image for /callBrandingProfiles does not affect the /ivrBrandingProfiles

Note: In locally hosted branding, only **`background.jpg`** will be used for the call background and ivr background images; **`passcode_background.jpg`**, **`passcode_or_blank_required_background.jpg`**, **`passcode_or_blank_timeout_ background.jpg`**, **`deactivated_background.jpg`** and **`ivr_background.jpg`** are ignored.

3.2.4 Overview of customization procedure (web server hosted branding)

The following steps provide an overview of the customization procedure, for a detailed procedure example, see Chapter . For steps on implementing locally hosted branding, see Chapter .

Note: If using web server hosted branding, ensure that your web server is configured correctly with the proper MIME type for the WAV and JPEG file extensions (e.g. audio/wav and image/jpeg MIME type).

1. On the web server create a directory for the customization files and place in it all the files listed in Section 3.2.1 and Section 3.2.3. Do not zip these files.

   Cisco have created an archive with their audio and video files, the zip files are downloadable from the Customization subsection on this web page. The files can be individually used when a mix of custom and default files are required. The necessary files must be copied into the directory. If any files are missing then no alternative or default file will be used in their place.

2. Using the API, create a callBrandingProfile specifying the resourceLocation field and URL on the webserver where the customized files are held. (See Section for details). The Meeting Server uses these details to retrieve the files from the web server.

   Note: : If no resourceLocation is specified, then by default the Cisco logo will be used.

3. Use this callBrandingProfile to customize your SIP/Lync calls on:
   a. a system level (that is, as part of the global systems profile)
   b. a per-tenant basis (as part of the tenant definition)
   c. a per-space basis (as part of the space definition)

   Note: When call branding is applied at more than one level, then the lowest level callBrandingProfile defined is the one that it used in a call leg. Only the callBrandingProfile applied at the system level is used for single branding. See the Using Profiles section in the API Reference.

3.2.5 Missing and invalid files

Files missing from the web server and those with an invalid format are ignored unless the alternative or default file is available.

Table 8: Alternative and default files

Missing or invalid file	Alternative or default file
background.jpg	No alternative file, screen remains black.
passcode_background.jpg	background.jpg *
passcode_or_blank_required_background.jpg	background.jpg *
passcode_or_blank_timeout_ background.jpg	background.jpg *

Missing or invalid file	Alternative or default file
deactivated_background.jpg	background.jpg *
welcome.wav	No alternative or default file **
timeout.wav	No alternative or default file **
call_join_confirmation.wav	No alternative or default file **
call_join.wav	No alternative or default file **
call_outgoing_welcome.wav	No alternative or default file **
cospace_join_confirmation.wav	No alternative or default file **
cospace_join.wav	No alternative or default file **
cospace_outgoing_welcome.wav	No alternative or default file **
disconnected.wav	No alternative or default file **
meeting_recorded.wav	No alternative or default file **
meeting_recording_ended.wav	No alternative or default file **
meeting_streamed.wav	No alternative or default file **
meeting_streaming_ended.wav	No alternative or default file **
meeting_ended.wav	No alternative or default file **
meeting_locked.wav	No alternative or default file **
meeting_unlocked.wav	No alternative or default file **
only_participant.wav	No alternative or default file **
participant_join.wav	No alternative or default file **
participant_leave.wav	No alternative or default file **
passcode_entry.wav	No alternative or default file **
passcode_or_blank_required_entry.wav	No alternative or default file **
passcode_or_blank_timeout_entry.wav	No alternative or default file **
passcode_incorrect_try_again.wav	No alternative or default file **
passcode_incorrect_final_attempt.wav	No alternative or default file **
passcode_incorrect_goodbye.wav	No alternative or default file **
total_participant_1.wav	No alternative or default file **
total_participant_2.wav	No alternative or default file **
total_participant_3.wav	No alternative or default file **

Missing or invalid file	Alternative or default file
total_participant_4.wav	No alternative or default file **
total_participant_5.wav	No alternative or default file **
total_participant_6.wav	No alternative or default file **
total_participant_7.wav	No alternative or default file **
total_participant_8.wav	No alternative or default file **
total_participant_9.wav	No alternative or default file **
total_participant_10.wav	No alternative or default file **
total_participant_11.wav	No alternative or default file **
total_participant_12.wav	No alternative or default file **
total_participant_13.wav	No alternative or default file **
total_participant_14.wav	No alternative or default file **
total_participant_15.wav	No alternative or default file **
total_participant_16.wav	No alternative or default file **
total_participant_17.wav	No alternative or default file **
total_participant_18.wav	No alternative or default file **
total_participant_19.wav	No alternative or default file **
total_participant_20.wav	No alternative or default file **
total_participant_greater_than_20.wav	No alternative or default file **
waiting_for_host.wav	No alternative or default file **

Note: * You do not need to load multiple copies of background.jpg, the single file will be used automatically to replace the missing images.

Note: ** If this audio file is missing or format is invalid, then nothing will be played.

4 Invitation text customization

Customization of the invitation text is done via the use of a template.

Note: On upgrade to a new release, administrators must recreate all custom email invite templates as there is no migration path for existing custom email invite templates from a previous release to a new release.

4.1 Cisco Meeting Server web app Custom Email Invites

Meeting Server 2.9 introduced Custom Email Invites for use with the new Cisco Meeting Server web app.

This allows an administrator to create and upload different email invitation templates so that web app users can:

- send an email invite — in a language of their choice — to other people to join a future meeting.
- send email invites appropriate to different audiences, for example external and internal participants.

In Meeting Server 3.2 and later versions, administrators can also use the callBridges API to get the email invite text. For more information on using the API, see the API Reference Guide.

Note: We recommend using locally hosted branding for web app Custom Email invites. However, if you wish to use multiple languages on a more complex deployment (e.g. a multi-tenant deployment) then you will need to use a remote branding web server. For more information see " Overview of customization procedure (web server hosted branding) for web app using Web Bridge 3" on page 39 .

4.1.1 Types of web app invitation templates

Web app offers support for invitation templates for different languages and for different audiences.

4.1.1.1 Invitation templates for different languages

The default email invitation template is a .txt file available in 21 languages on web app.

- invitation_template_xx_XX.txt

Replace xx_XX with the language code. For example: invitation_template_fr_CA.txt

The table below lists the supported languages and their region codes:

Language	Code
Arabic	ar_SA
Danish	da_DK
German	de_DE
English (GB)	en_GB
English (US)	en_US
Spanish (Spain)	es_ES
Suomi (Finnish)	fi_FI
French (Canadian)	fr_CA
French (European)	fr_FR
Indonesian	id_ID
Italian	it_IT
Japanese	ja_JP
Korean	ko_KR
Norwegian	nb_NO
Dutch	nl_NL
Polish	pl_PL
Portuguese (Brazil)	pt_BR
Russian	ru_RU
Swedish	sv_SE
Turkish	tr_TR
Chinese (China)	zh_CN
Chinese (Taiwan)	zh_TW

If you want to overwrite the default templates, you can create your own language tagged template file and upload to locally hosted branding. The Meeting Server interprets these language tags to return the appropriate template option in the web app.

A web app user can only select a language template that is uploaded – a language option is not shown to the web app user in the drop-down list if it is not uploaded.

Note: You must upload the template files to all Meeting Servers in a cluster.

4.1.1.2 Invitation templates for different audiences

A user can generate different email invites appropriate for different audiences, for example, external and internal participants. To do this, you need to create an invitation template file with the following naming convention example: invitation_template_Internal.txt. This file will display in the email invite options as " Internal" as shown in 4.

For more information on using the email invite option, see Cisco Meeting Server web app Important Information.

4.2 Creating the invitation template text file

Create an invitation template text file in UTF-8 using the information in Section 4. Name the file using the naming convention appropriate to the file type, for example:

- invitation_template.txt (shows as **Default** in the web app drop-down list)
- invitation_template_xx_XX.txt
- invitation_template_*Internal*.txt (the italicized part of the filename shows in the web app drop-down list.)

4.2.1 Custom email invitation content created with new syntax

Version 3.1 introduces new syntax to create a custom email invitation.

Note: On upgrade to 3.1, administrators must recreate all custom email invite templates as there is no migration path for existing custom email invite templates from a previous release to 3.1.

Header

All custom email templates must now start with " Subject:" followed by an empty line to separate from the body text. The subject header is used to generate the email link that sets the text specified in the header as subject of the email. The header cannot contain new line characters. The syntax is:

`Subject: <subject….>` [followed by an empty line]

Format

The new custom email invitation template syntax allows variables, loops, conditions, includes, callbacks, and comments (nested and combined as required).

There are escape sequences that allow you to write conditional code, loops and so on, i.e. New line escape sequence (the same as prior to 3.1). Any new line starting with # introduces a template statement. Example:

```
#if name
I can use a name variable as it is set to %name%
```

```
#endif
```

You can specify an inline statement by using " {%" and " %}" . Example:

```
You have been invited to a meeting{% if name %}: %name%{% endif %}
```

You can also add comments to your templates using " {# my comment #}" . So any text between those escape sequences won't be rendered. Example:

```
Hello, {# This section is just an intro #}
```

Variables

The template can contain both variables and conditions. This allows a single template to be used for multiple spaces and gives a consistent feel to the invitations.

%<var name>% will be substituted with the content of a variable.

The variables that are currently defined are detailed below:

Table 9: Variables in invitation template

Variable name, description, and example

name
 	
This will be replaced by the name of the cospace. To see if it has content, check its length or whether it's not null. Example:
`#if name`
`The cospace name is %name%`
`#endif`
 	
`#if length(name)>0`
`The cospace name is %name%`
`#endif`

uri
 	
The cospace uri that can be used to 'join by uri' in the web app or dialed-in on endpoints.
It can be checked by length or if set to null.
`#if uri`
`The uri is: %uri%`
`#endif`
 	
`#if length(uri) > 0`
`The uri is: %uri%`
`#endif`

Variable name, description, and example

organizer name
 	
This will be replaced by the organizer's name in the email invite.
`#if organizername`
`Organizer: %organizername%`
`#endif`

numeric_id
 	
This is the callId of the cospace. It is normally a numeric identifier for the cospace.
It can be checked on length or if set to null.
Example:
`#if numeric_id`
`The meeting id is %numeric_id%`
`#endif`
 	
`#if length(numeric_id) > 0`
`The meeting id is %numeric_id%`
`#endif`

passcode
 	
The passcode assigned to the cospace.
It can be checked on length or if set to null.
Example:
`#if passcode`
`The meeting requires a passcode: %passcode%`
`#endif`
 	
`#if length(passcode) > 0`
`The meeting requires a passcode: %passcode%`
`#endif`

Variable name, description, and example

ivr_numbers
 	
This is an array of objects that contains label and number.
(It supersedes the dial_pstn variable that was used in custom email templates prior to 3.1.)
It can be an empty array and it needs to be looped over to access the internal variables: label and number.
It cannot be checked for null, only by length.
• label: The label assigned to this IVR number when registering it using the Meeting Server API.
• number: The IVR number to dial in.
 	
Example:
`#if length(ivr_numbers) > 0`
`You can use the following dial-in number: %ivr_numbers.0.number%`
`%`
`#endif`
 	
`#for ivr_number in ivr_numbers`
`%loop.index1% - %ivr_number.label%: %ivr_number.number%`
`#endfor`

web_bridge_addresses
 	
This is an array of objects that contain label, address and hyperlink.
(It supersedes the hyperlink and webbridge_url variables used in custom email templates prior to 3.1.)
It can be empty but it cannot be checked for null, only by length.
• label: The label assigned to this Web Bridge Address when registering it via the Meeting Server API. Example: web app address
• address: The HTTPS address that can be used to access the web app on this Web Bridge. Example: https://join.mydomain.com/
• hyperlink: When a cospace has a callId non-empty (aka, meeting id or number_id) and a non-empty address in this object, then a unique hyperlink will be generated that can be used to directly join a given meeting without requesting a passcode.
 	
Example:
`#if length(web_bridge_addresses) > 0`
`You can use the following web app address: %web_bridge_addresses.0.address%`
`#endif`
 	
`#for wba in web_bridge_addresses`
`Label: %wba.label%`
`Address: %wba.address%`
`Hyperlink: %wba.hyperlink%`
`#endfor`

Loops

You can use loops by using the " for" and " endfor" keywords. Example:

```
#for ivr_number in ivr_numbers
%loop.index1% - %ivr_number.label%: %ivr_number.number%
#endfor
```

In a loop, the special variables are:

- loop.index (number): Loop iteration number starting from 0.
- loop.index1 (number): Loop iteration number starting from 1.
- loop.is_first (boolean): If the iteration is the first iteration.
- loop.is_last (boolean): If the iteration is the last iteration.
- loop.parent.*: In nested loops, the parent loop variables are available using loop.parent.<name of var> example: loop.parent.is_first.

You can also iterate over objects like web_bridge_address or ivr_number. Example:

```
#for ivr_number in ivr_numbers
{%- for key, value in ivr_number -%}
%key%: %value%
{%- endfor -%}
#endfor
```

Conditions

Conditions support the typical " if" and " else" statements. Examples:

Print something different depending on the length of a passcode

```
{% if length(passcode) >= 3 %}…{% else if length(passcode) >= 10 %}…{% endif %}
```

Print something only if a particular field is set in a web_bridge_address

```
{% if web_bridge_address.hyperlink %}…{% endif %}
```

Logical operations

You can use " and" , " or" and " not" to generate complex conditions. Example:

```
{% if numeric_id and passcode %}…{% endif %}
{% if not name %}…{% endif %}
```

Functions

A few functions are implemented within the template syntax. These are detailed below.

Upper and lower function, for string cases. Example:

```
Join {% upper(name) %}
Join {% lower(name) %}
```

Range function, useful for loops. Example:

```
{% for i in range(4) %} %loop.index1% {% endfor %}
```

```
{# It will show only the first two web_bridge_addresses #}
{% for i in range(2) %} %at(web_bridge_addresses, i).address% {% endfor %}
```

Get first and last element in a list. Example:

```
The first ivr_number: % first(ivr_numbers).number %
The last ivr_number: % last(ivr_numbers).number %
```

Sort a list. . Example:

```
#for sort(ivr_numbers)
...
#endfor

{# Produces [1, 2, 3] #}
% sort([3,2,1]) %
```

Round numbers to a given precision. . Example:

```
{# returns 1 #}
% round(1.4142135, 0) %
{# returns 1.4 #}
% round(1.4142135, 1) %
```

Check if a value is odd, even or divisible by a number. Example:

```
{# returns true #}
% odd(1) %
{# returns true #}
% even(2) %
{# returns true #}
% divisibleBy(42, 7) %
```

Maximum and minimum values from a list. Example:

```
{# returns 3 #}
% max([1,2,3]) %
{# returns 1 #}
% min([1,2,3]) %
```

Convert strings to numbers. Example:

```
# if int(ivr_number) >= 123123
...
#endif
```

Set default values if variables are not defined. Otherwise, the template rendering will fail and show no result. Example:

```
{% if default(has_valid_ivr_number, false) %}... {% endif %}
```

Check if a key exists in an object. Example:

```
{# returns false #}
% exists("pstn_dial_in") %
{# returns true #}
% existsIn("a", myobj) %
{# returns false #}
```

```
% existsIn("c", myobj) %
```

Check if a key is a specific type. Example:

```
{# returns true #}
% isArray(ivr_numbers) %
{# returns true #}
% isString(ivr_numbers.0.number) %
{# returns false #}
% isString(int(ivr.numbers.0.number)) %
Implemented type checks are: isArray, isBoolean, isFloat, isInteger, isNumber,
isObject, isString,
```

Whitespace Control

Whitespaces are removed by default. To support a more readable template style you can also remove whitespaces for both statements and expressions by hand. If you add a minus sign (-) to the start or end, the whitespaces before or after that block will be removed. Example:

```
Cospace name: % name -%        .
{# Produces: Cospace name: blah. #}
```

Stripping behind a statement or expression also removes any new lines.

4.2.2 Example invitation template

Note:

- There is a 10000 byte limit* on the size of the invitation template .txt file.
- All invitation templates must be provided in UTF-8 format.
- Extended ASCII characters are not supported.
- UTF-8 format invitation template .txt files need to have Unix line endings, i.e. LF, not CRLF (as Windows uses). Omitting Unix line endings will result in the file not working.
- Language .txt files should have the appropriate language tags for its language variant (as defined by the IANA Language Subtag Registry) – where the two lower case characters indicate the language code and the two upper case characters the region code. For example, invitation_template_en_GB.txt, where " en" is the english language, and " GB" is the region (United Kingdom).
- The part of the file name between " invitation_template" and the .txt suffix can use alphanumerics and " _" (underscores) of up to 32 characters, i.e. any of the following regular expression: ^[a-zA-Z_]{1,32}$.

Note: *At the time of publication whilst using web app from Google Chrome on Windows, the **Invite > Send email** or **Open email** option fails if the file size of the invitation template exceeds 1491 bytes. This is a known issue with Google Chrome browsers, more information about this issue is available here: https://bugs.chromium.org/p/chromium/issues/detail?id=1034497

If this occurs on web app, Open email will be greyed out, however, you can select Copy which will allow you to paste the meeting join information into your preferred email client.

Use the example below and customize with your specific values in the placeholders. Save it using the appropriate file name format as defined above.

Note: The " Subject:" header must be followed by an empty line to separate from the body text.

```
Subject: {% if name %}You are invited to join a meeting: %name%{% else %}You
are invited to join a meeting{% endif %}

#if numeric_id
Meeting ID: %numeric_id%
#if passcode
Meeting passcode: %passcode%
#endif
#endif

#for wba in sort(web_bridge_addresses)
#if wba.address or wba.hyperlink
#if loop.index == 0
Join from a computer, mobile phone or tablet
#endif
{% if wba.label %} %wba.label%:{% endif %} {% if wba.hyperlink
%}%wba.hyperlink%{% else %}%wba.address%{% endif %}
#endif
#endfor

#for ivrn in sort(ivr_numbers)
#if ivrn.number
#if loop.index == 0
Join by phone
#endif
{% if ivrn.label %} %ivrn.label%:{% endif %} %ivrn.number%
#endif
#endfor

#if uri
Join from a video conferencing system or application
Dial %uri%
#endif
```

4.3 Overview of customization procedure (web server hosted branding) for web app using Web Bridge 3

We recommend using locally hosted branding for Custom Email Invite templates with web app. For more information, see Implementing locally hosted branding.

You only need to use remote web server hosted branding if you want to have a different set of customized invitation templates per coSpace. For example, if you have multiple tenants and you want them to have their own custom invitation template.

The following steps provide an overview of the customization procedure for remote web server hosted branding.

Note: All invitation templates must be provided in UTF-8 format. Extended ASCII characters are not supported. UTF-8 format invitation template .txt files need to have Unix line endings, i.e. LF, not CRLF (as Windows uses). Omitting Unix line endings will result in the file not working.

---

1. Configure the remote web server by creating a server-side script that can process the query string correctly to ensure it returns the correct requested invitation template.

   For more information on how to do this, see this example: https://github.com/ciscocms/example-http-invites . (Meeting Server uses this method to ensure backwards compatibility with existing deployments.)

2. Using the API, create a **callBrandingProfile** specifying the **invitationTemplate** field and URL on the web server where the template is held, you need to specify the full path to the server side script, for example: http://192.0.2.0/branding/invitation.cgi. The Meeting Server uses these details to retrieve the template from the web server.

   When using web app, it will try to add the requested language or string (e.g. " Internal") as a query string to the URL depending on the selected option in the drop-down list of the web app, for example:

   - http://192.0.2.0/branding/invitation.cgi?language=es_ES

   or

   - http://192.0.2.0/branding/invitation.cgi?language=Internal

3. Use this callBrandingProfile to customize invitations at:
   - a system level (that is, as part of the global systems profile)
   - a per-tenant basis (as part of the tenant definition)
   - a per-space basis (as part of the space definition)

5 Implementing locally hosted branding

If you are using locally hosted branding files, the following steps provide an overview of the customization procedure.

Note: If you are changing from using a web server to hosting the files locally then follow the guidance in Section 7.1 before following the steps below.

Note: For limitations in using locally hosted branding, see Limitations of using locally hosted branding.

5.1 Web app customizations

The branding files for web app are held within an archive (zip) file. If you are changing from using a web server to hosting the files locally then follow the guidance in Section 7.1 before following the steps below.

The following steps provide an overview of the customization procedure, for a detailed procedure refer to the Customization Guidelines.

Note: The commands in the following steps are for console/terminal environments (i.e. command prompt or terminal) and not for SFTP clients such as WinSCP.

1. Create a zip archive file containing these files:
   - favicon.png
   - sign_in_background.jpg
   - sign_in_logo.png
   - text_strings.json

   Note: When you zip the files, do not zip the folder containing the branded files. If this is done, this will create an extra layer of folder (zipped file > folder > branded files). Instead, highlight the branded files and right-click to zip them (or open a zip application and zip the files together). This will create a zipped file with the branded files without creating an extra layer of folder (e.g. zipped file > branded files).

   Note: This zip file must be named **`web_branding.zip`**, it cannot have a different file name.

2. For each Meeting Server with an enabled Web Bridge which will locally host this zip archive:

a. Connect your SFTP client to the IP address of the MMP.
b. Log in using the credentials of the MMP admin user.
c. Upload the zip file **web_branding.zip**. For example:
   ```
   PUT web_branding.zip
   ```
d. Connect your SSH client to the IP address of the MMP.
e. Log in using the credentials of the MMP admin user.
f. Restart the Web Bridge
   ```
   webbridge restart
   ```

The new branding will be picked up after the restart. The Web Bridge retrieves the locally hosted branding file for the WebRTC app, rather than relying on the Call Bridge to pass the file.

5.2 IVR Message, SIP/Lync Call Message and Invitation Text customization

To locally host the IVR messages , SIP/Lync call messages and invitation text you need to create a Call Bridge branding zip file.

The following steps provide an overview of the customization procedure, for a detailed procedure refer to the Customization Guidelines.

1. Create the call branding zip file, this file must be named **call_branding.zip** to ensure it is processed correctly.
   a. Create a single folder with the files listed in Chapter 3 of the Customization Guidelines, these are the same files that are used if a web server is deployed.

      Note: In locally hosted branding, only **background.jpg** will be used for the call background and ivr background images; **passcode_background.jpg**, **passcode_or_blank_required_background.jpg**, **passcode_or_blank_timeout_ background.jpg**, **deactivated_background.jpg** and **ivr_background.jpg** are ignored.

   b. Add the file **invitation_template*.txt** containing the invitation text to the folder, as described in Chapter 4 of the guidelines .

      Note: For this call branding zip file, you must use the filename convention **invitation_template*.txt** even if you are already using a different filename on a web server.

   c. Zip up the files in the folder, all files should be at the top level of the zip file (no folders nested in the zip file), the filename must be **call_branding.zip**.

2. Install the IVR, call and invitation customization on every Call Bridge. For each Meeting Server:
   a. Connect your SFTP client to the IP address of the MMP.
   b. Log in using the credentials of the MMP admin user.
   c. Upload the zip file **`call_branding.zip`**. For example:
      ```
      PUT call_branding.zip
      ```
   d. Connect your SSH client to the IP address of the MMP.
   e. Log in using the credentials of the MMP admin user.
   f. Restart the Call Bridge
      ```
      callbridge restart
      ```

The new branding will be picked up after the restart.

5.3 Testing customized `invitation_template*.txt`

For web app

Meeting Server caches the invitation templates retrieved from the remote server for 2 minutes before requesting them again. So once you close the invitation pop-up in the web app the invitation will be requested again and regenerated for each requested language after that 2 minute caching period.

5.4 Removing locally hosted branding files

Follow these steps for each Meeting Server hosting local branding files.

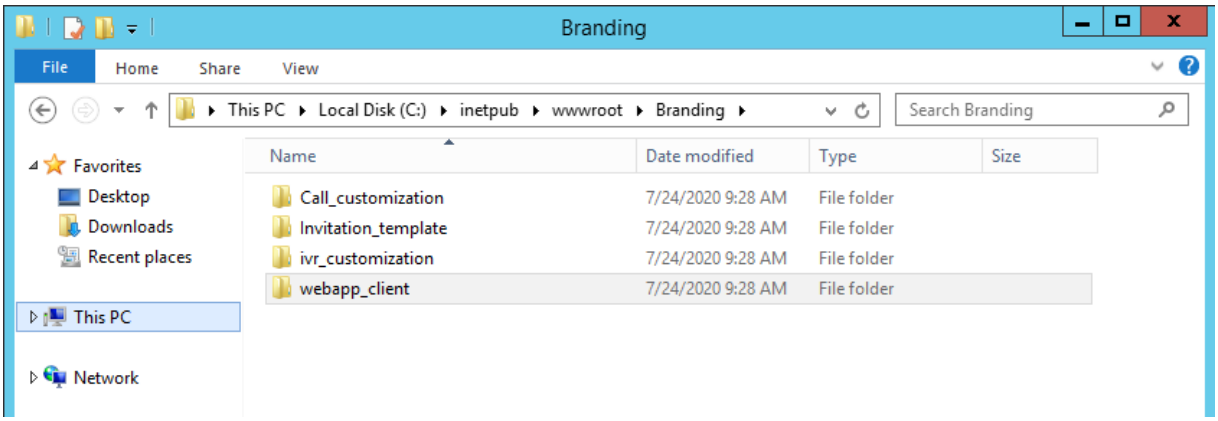
1. Connect your SFTP client to the IP address of the MMP.
2. Log in using the credentials of the MMP admin user.
3. Remove the locally hosted branding files from the Web Bridge
   ```
   RM web_branding.zip
   ```
4. Remove the locally hosted branding files from the Call Bridge
   ```
   RM call_branding.zip
   ```
5. Connect your SSH client to the IP address of the MMP.
6. Log in using the credentials of the MMP admin user.
7. Restart the Web Bridge
   ```
   webbridge3 restart
   ```
8. Restart the Call Bridge
   ```
   callbridge restart
   ```

6 Implementing remote branding using API

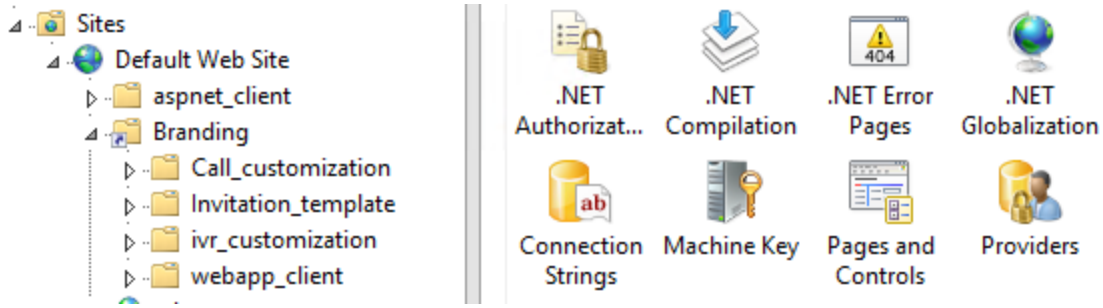
This section provides a step-by-step example of call branding using Microsoft Windows in a web server hosted branding scenario. It assumes that Microsoft Internet Information Services (IIS) Manager is installed with the latest patches on the web server that will host your call branding files.

6.1 Setting up the web server

1. On the web server create a directory structure under the default IIS Manager location c:\inetpub\wwwroot to hold the branding/custom files. The example below shows a Branding folder with four subfolders: Call_customization, Invitation_template, ivr_ customization, webapp_client.



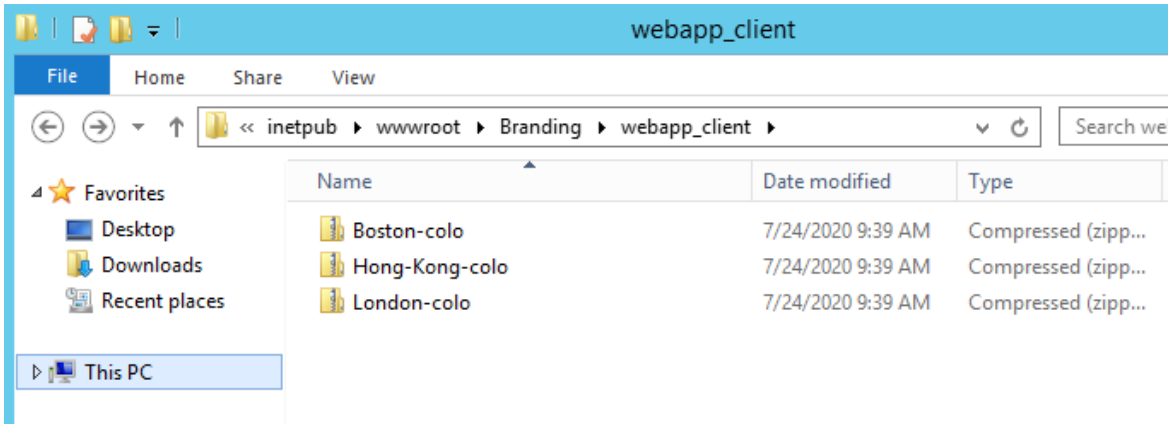
2. Open IIS Manager and expand the Default Web Site. The newly created folders should show within the list.



Note: If the newly created folders do not show within the list or your folders are outside of the standard IIS Manager location, you will need to create a new Virtual Directory. Within IIS,

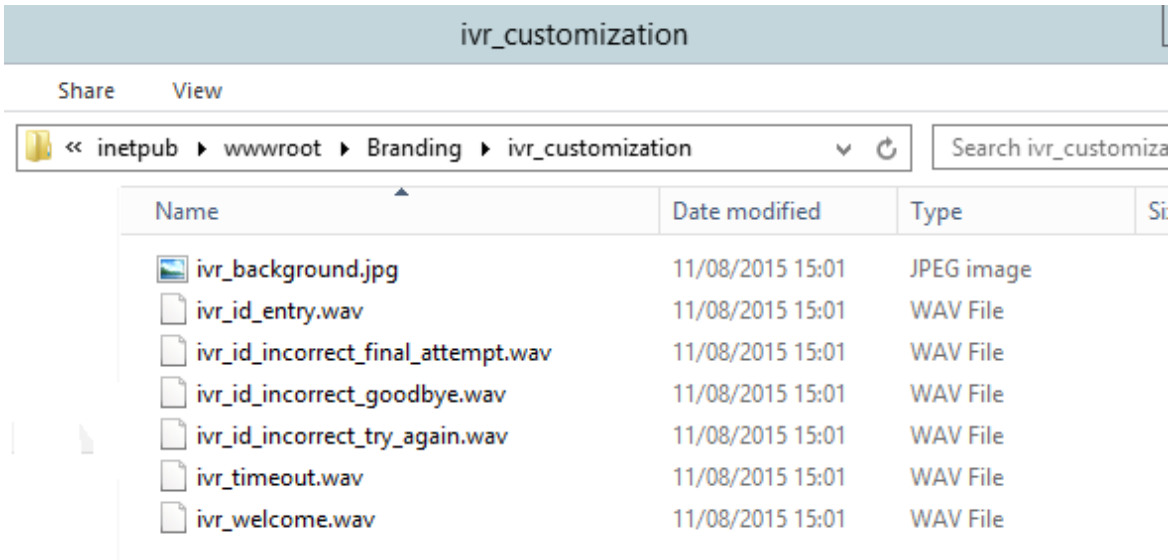
navigate to Default Web Site, right click and select Virtual Directory. Provide an Alias for this folder and browse for the physical path where the files will reside.

1. Place all the files into the folders, for the appropriate to the level of branding you intend to do.
   a. For web app customization these will be the zip files from Section 2.2.



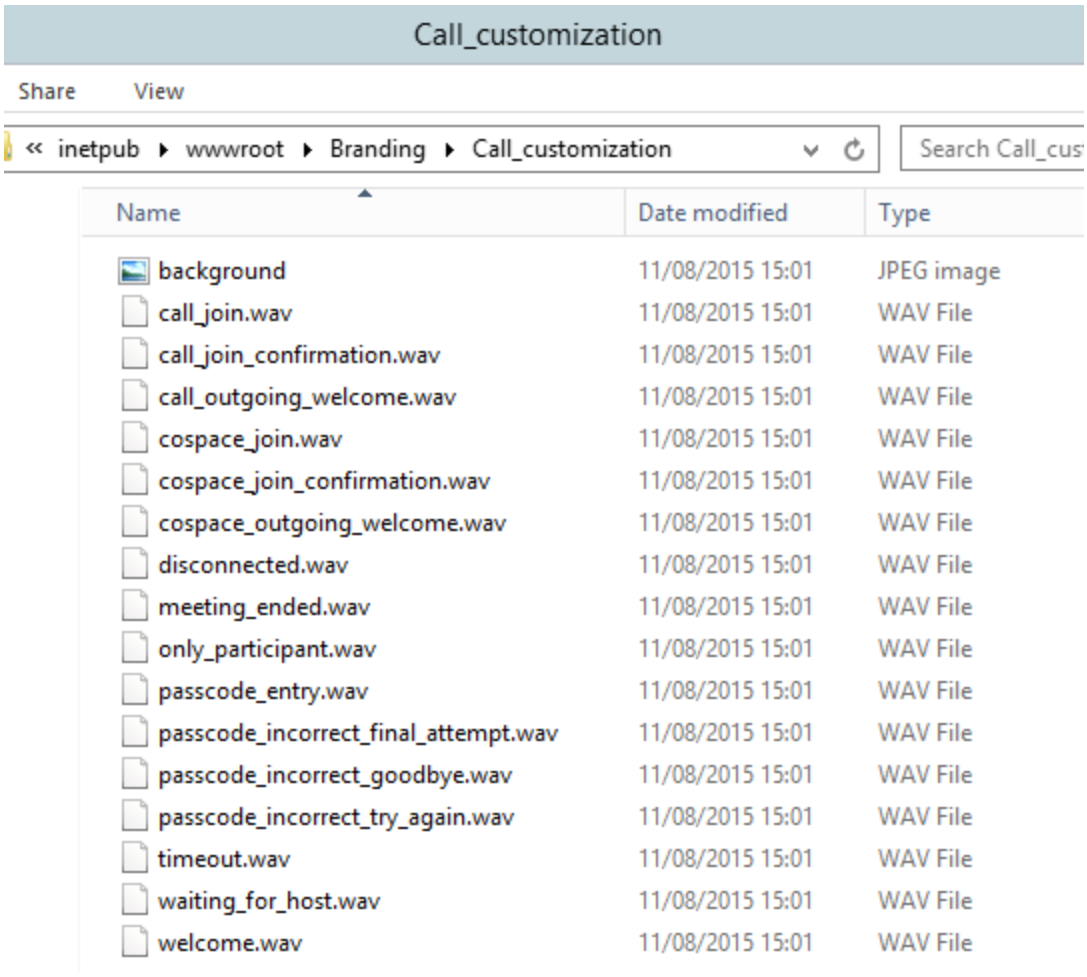
Note: in the above screenshot the webapp_client customization files are zipped together by location.

   b. For ivr customization these will be the files from Section 3.1



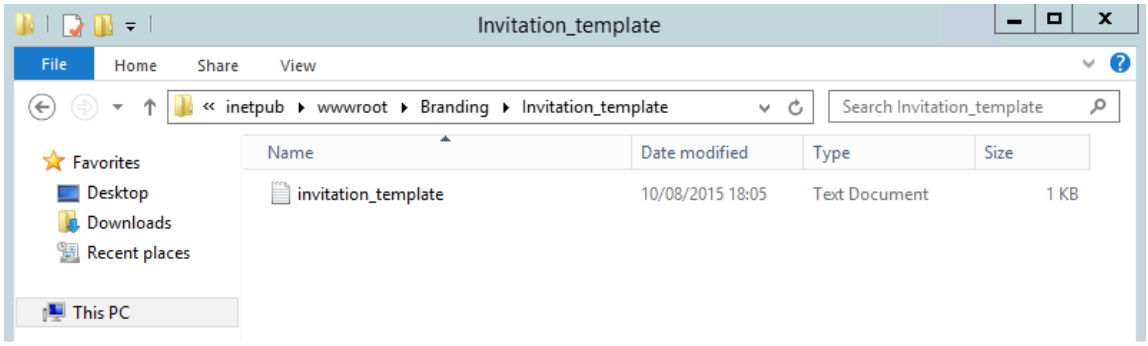
Note: the files for ivr customization are not zipped together.

   c. For call customization these will be the files from Section 3.



Note: the files for call customization are not zipped together.

d. For the invitation template, this will be the file you created from Chapter 4.

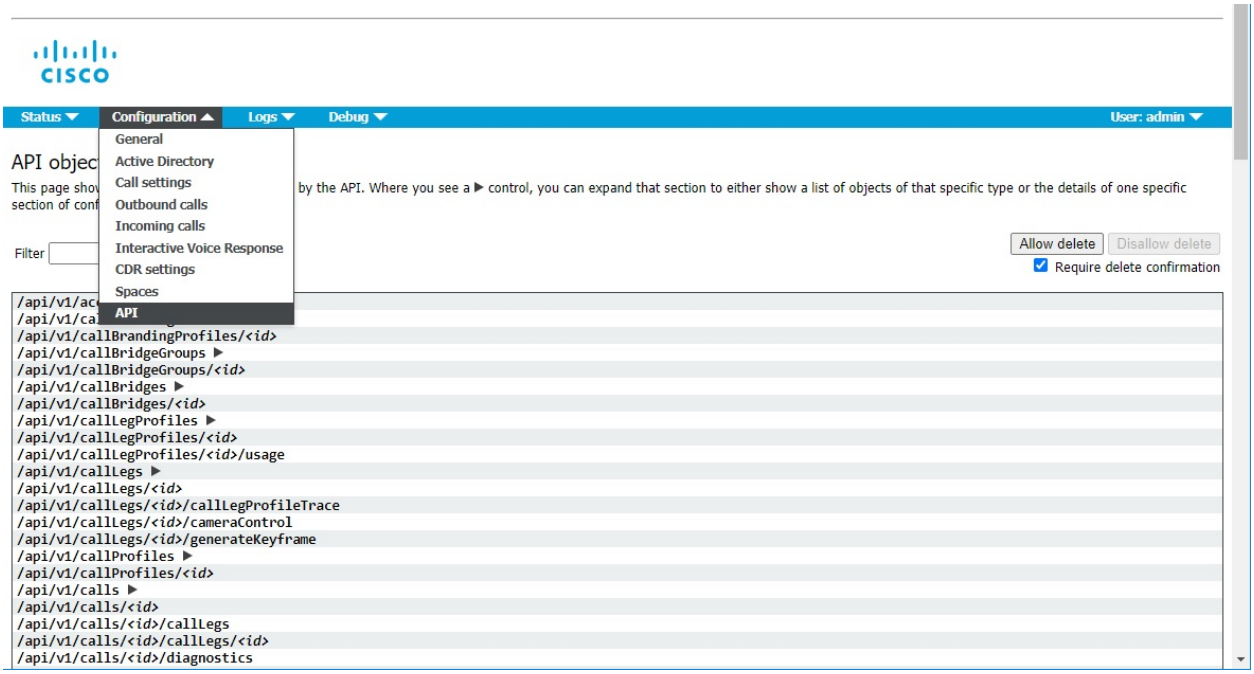


6.2 Using the API for branding

The remainder of this chapter shows an example using the API accessed via the Meeting Server Web Admin interface to brand the web app, IVR, SIP calls and apply custom invitations.

The API can be accessed using the Meeting Server Web Admin Interface rather than using API Methods and third-party applications. After logging in to the Web Admin interface, navigate to the **Configuration** tab and select **API** from the pull-down list. See Figure 5.

Figure 5: Accessing the API via the Meeting Server web interface



For more information on using the API, see the API Reference Guide.

6.2.1 Applying branding to web app

Branding files for web app are applied to a webBridgeProfile which is then associated with a webBridge node. If you need different web app branding for different tenants, then you need to set up a Web Bridge for each branded web app.

Note: If branding server hosted over HTTPS, the Meeting Server and branding server certificates must be signed by valid or same CA. Meeting Server may fail to communicate with the branding server if the branding server certificates are not signed by valid CA.

---

1. Using the Web Admin interface of one of the servers in the cluster, select **Configuration > API**:
   a. From the list of API objects, tap the ► after **/api/v1/webBridgeProfiles**
   b. Click **Create new**
   c. Enter the **resourceArchive** parameter and **url** where the web app branding files are held on your web server.

      At this point you can either apply the resourceArchive:

   d. at an individual Web Bridge level – from the list of API objects, tap the ► after **/api/v1/webBridges**, scroll down the parameters to **webBridgeProfile** and click **Choose**. Select the **object id** of the webBridgeProfile to use from the resulting dialog, return to the Web Admin page and click **Create**.

      or

   e. globally to all Web Bridges – from the list of API objects, tap the ► after **/api/v1/system/profiles**, click **View or edit**, scroll down the parameters to **webBridgeProfile** and click **Choose**. Select the **object id** of the webBridgeProfile to use from the resulting dialog, return to the Web Admin page and click **Modify**.
   f. Your web app branded files should now be used for web app calls to the Web Bridge, make a test call to check that the branded files are applied.

6.2.2 Applying branding to SIP calls and/or customized invitations

Branding files for SIP calls and customized invitation templates are applied to the callBrandingProfiles node.

Note: If branding server hosted over HTTPS, the Meeting Server and branding server certificates must be signed by valid or same CA. Meeting Server may fail to communicate with the branding server if the branding server certificates are not signed by valid CA.

---

1. Using the Web Admin interface of one of the servers in the cluster, select **Configuration > API**:
   a. From the list of API objects, tap the ► after **/api/v1/callBrandingProfiles**
   b. Click **Create new**
   c. Scroll down the parameters to **resourceLocation** and **invitationTemplate** and enter the URLs where the branding files are held on your web server. Be sure to specify the full filename including the path to the file for the invitationTemplate URL only, for example: https://192.0.2.23/branding/invitation_template.txt. You can also use http.

      Note: If you have created subfolders under /Branding then you need to POST the url of each subfolder to callBrandingProfiles.

   d. Click **Create**.
2. Assign this newly created object id to the branding level required:
   - the system level (that is, as part of the global systems profile),
   - a per-tenant basis (as part of the tenant definition),
   - a per-space basis (as part of the space definition).

   The example below applies the branding to a previously created tenant:
   a. From the list of API objects, tap the ► after **/api/v1/tenant/**.
   b. Click on the **object id** of the individual tenant.
   c. Scroll down the parameters to **callBrandingProfile** and click **Choose**. Select the **object id** of the **callBrandingProfile** to use for this tenant.
   d. Click **Modify**
3. Your Call Branding files should now be used for calls for that tenant. Place a call from a SIP endpoint to a space setup for the tenant. Make sure the call displays the background picture, and that the voice prompt is audible.

6.2.3 Applying branding to an IVR

Branding files for IVR are applied to the ivrBrandingProfiles node. You can only configure one Web Bridge on a Meeting Server. If you need different IVR branding for different tenants, then you need to set up a Web Bridge for each branded IVR profile.

Note: If branding server hosted over HTTPS, the Meeting Server and branding server certificates must be signed by valid or same CA. Meeting Server may fail to communicate with the branding server if the branding server certificates are not signed by valid CA.

1. Using the Web Admin interface of one of the servers in the cluster, select **Configuration > API**:
   a. From the list of API objects tap the ► after **/api/v1/ivrBrandingProfiles**
   b. Click **Create new**
   c. Enter the **resourceLocation** URL where the ivr branding files are held on your web server.

      Note: If you have created subfolders under /Branding then you need to POST the url of the ivr subfolder to ivrBrandingProfiles.

   d. Click **Create**

2. Assign this newly created object id to the branding level required:
   - the system level (that is, as part of the global systems profile),
   - a per-ivr basis (as part of the ivr definition).
   - enter the ivrBrandingProfile field and <ivrBrandingProfile id>

   The example below applies the ivr branding to a previously created ivr:
   a. From the list of API objects, tap the ► after **/api/v1/ivrs/**.
   b. Click on the **object id** of the individual ivr.
   c. Scroll down the parameters to **ivrBrandingProfile** and click **Choose**. Select the **object id** of the **ivrBrandingProfile** to use for this ivr.
   d. Click **Modify**

3. Your IVR Branding files should now be used for calls to the specified ivr, make test calls to the ivr to check that the branded ivr files are applied.

7 Switching between branding methods

You can change from one method to another, however we recommend NOT mixing the two methods. For more information, refer to the steps below:

7.1 Changing from web server (or default) branding to locally hosted branding

If changing from web server (or default) branding to locally hosted branding, follow these recommendations:

- for every Call Bridge ensure:
  - the **`resourceLocation`** parameter for the IVR messages is not set; use a PUT method to set the **`resourceLocation`** parameter as blank on **`/ivrBrandingProfile/<ivr branding profile id>`**,
  - the **`resourceLocation`** parameter for the call messages is not set; use a PUT method to set the **`resourceLocation`** parameter as blank on **`/callBrandingProfile/<call branding profile id>`**,
  - the **`invitationTemplate`** parameter for the call messages is not set; use a PUT method to set the **`invitationTemplate`** parameter as blank on **`/callBrandingProfile/<call branding profile id>`**.
- for every Call Bridge which has a configured Web Bridge, make sure the **`resourceArchive`** field is not set in that configuration; use the PUT method to set the **`resourceArchive`** parameter as blank on API object **`/webBridgeProfiles`**.
- stop using web server URLs in scripts and configurations of call leg profiles,
- configure **`call_branding.zip`** files on every Call Bridge, as described above,
- configure **`web_branding.zip`** files on every Web Bridge, as described above.

7.2 Changing from locally hosted to web server branding

If changing from locally hosted to web server branding, then follow these recommendations:

- remove locally hosted **`call_branding.zip`** files from every Call Bridge as described above,
- remove locally hosted **`web_branding.zip`** files from every Web Bridge as described above,
- change all your scripts and configurations to use http://mywebserver/... as documented in the Customization Guidelines.

7.3 Mixing locally hosted and web server customization

If you install branding zip files on your Meeting Servers, but also deploy a web server and use it to serve branding resource files, then note the following:

For IVR, call and invitation customization:

- customization using the remote web server will override the locally hosted files,
- leaving the API fields blank or unset will cause the locally hosted files to be used.

For Cisco Meeting Server web app customization:

- customization using the remote web server will override the locally hosted files
- in a cluster environment the same invitation templates need to be uploaded to all the servers in a cluster for the web app users to have a consistent experience.

7.4 Limitations of using locally hosted branding

- Only one background image file, **`background.jpg`**, is used in locally hosted branding, other image files will be ignored.
- If you want different image backgrounds in different situations, for example during pass code entry or IVR, the only way is to use a web server for customization as described in the Customization Guidelines.
- To use multiple sets of branding files, you still need to use an external web server.
- If files are too large, missing or otherwise invalid then they will be treated in the same way as their web server equivalents and will not be used. In this instance the following behavior will occur:
  - For web app users, default files are provided: invitation_template_en_US.txt and invitation_template_es_ES.txt – if the locally hosted branding doesn't have them and there is no general invitation_template.txt, these default files will be used.

Note: For each Web Bridge 3 you can only get one brand, and there's no difference between local and remote hosted branding for Web Bridge 3.

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.

© 2022 Cisco Systems, Inc. All rights reserved.

Cisco Trademark

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)