# Cisco Unified Contact Center Enterprise Features Guide, Release 12.6(1)

**First Published:** 2021-05-14

**Americas Headquarters**
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

# CONTENTS

# Preface



## Change History

This table lists changes made to this guide. Most recent changes appear at the top.

| Change | See | Date |
|---|---|---|
| Added a new chapter **Reverse-Proxy Configuration** | Appendix | **May, 2022** |
| Added information about reverse-proxy deployments that use L7 intermediaries | Mobile Agent>VPN-less Access to Finesse Desktop>Supported Reverse-Proxy Deployment Models>Authentication>Authenticate WebSocket Connections | |
| Added a new CLI to to view the content of the proxy map file | Mobile Agent>VPN-less Access to Finesse Desktop>VPN-less Finesse Configurations>Add Proxy IP by Using CLI | |
| Added a new section **Performance** | Mobile Agent>VPN-less Access to Finesse Desktop | |

| Change | See | Date |
|---|---|---|
| Added a note related to IdP | Mobile Agent>VPN-Less Access to Finesse Desktop>Supported Reverse-Proxy Deployment Models>Authentication>SSO | **December, 2021** |
| Updated IdP related information and the Hostname Mapping Example figure | Mobile Agent>VPN-Less Access to Finesse Desktop>VPN-less Finesse Configurations>Populate Network Translation Data | |
| New section **Configure Reverse-Proxy Host Verification** has been added | Mobile Agent>VPN-Less Access to Finesse Desktop>VPN-less Finesse Configurations> | |
| New section **Historical and Real Time Gadgets** has been added | Mobile Agent>VPN-Less Access to Finesse Desktop | **November, 2021** |
| New section **VPN-Less Access to Finesse Desktop** has been added | Mobile Agent | |
| **Initial Release of Document for Release 12.6(1)** | | **May, 2021** |
| Edge Chromium (Microsoft Edge) updates | Browser Settings and Single Sign-On | |
| New chapter has been added | Agent Answers | |
| New chapter has been added | Call Transcription | |
| Customer Virtual Assistant has been renamed Virtual Agent–Voice and additional information has been added | Virtual Agent–Voice | |

# About This Guide

This guide explains features you can use in conjunction with Cisco Unified Contact Center Enterprise. For each feature, there is a description, procedures for initial setup, and details on the functionality the feature provides.

# Audience

This guide is prepared for Contact Center administrators who configure and run the contact center, manage agents, and address operational issues.

# Related Documents

| Subject | Link |
|---|---|
| Design considerations and guidelines for deploying a Unified CCE solution, including its various components and subsystems. | *Solution Design Guide for Cisco Unified Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-implementation-design-guides-list.html |

# Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business impact you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

### Cisco Bug Search Tool

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.

# Field Notice

Cisco publishes Field Notices to notify customers and partners about significant issues in Cisco products that typically require an upgrade, workaround, or other user action. For more information, see *Product Field Notice Summary* at https://www.cisco.com/c/en/us/support/web/tsd-products-field-notice-summary.html.

You can create custom subscriptions for Cisco products, series, or software to receive email alerts or consume RSS feeds when new announcements are released for the following notices:

- Cisco Security Advisories
- Field Notices
- End-of-Sale or Support Announcements
- Software Updates
- Updates to Known Bugs

For more information on creating custom subscriptions, see *My Notifications* at https://cway.cisco.com/mynotifications.

# Documentation Feedback

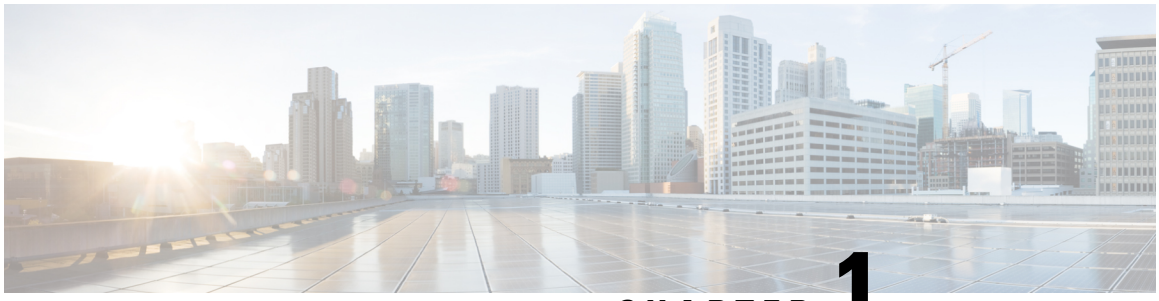
To provide comments about this document, send an email message to the following address: contactcenterproducts_docfeedback@cisco.com

We appreciate your comments.

# Conventions

This document uses the following conventions:

| Convention | Description |
|---|---|
| **boldface** font | Boldface font is used to indicate commands, such as user entries, keys, buttons, folder names, and submenu names.<br>For example:<br>• Choose **Edit** > **Find**.<br>• Click **Finish**. |
| *italic* font | Italic font is used to indicate the following:<br>• To introduce a new term. Example: A *skill group* is a collection of agents who share similar skills.<br>• A syntax value that the user must replace. Example: IF (*condition, true-value, false-value*)<br>• A book title. Example: See the *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide*. |
| `window font` | Window font, such as Courier, is used for the following:<br>• Text as it appears in code or that the window displays. Example:<br>`<html><title>Cisco Systems, Inc. </title></html>` |
| `< >` | Angle brackets are used to indicate the following:<br>• For arguments where the context does not allow italic, such as ASCII output.<br>• A character string that the user enters but that does not appear on the window such as a password. |



CHAPTER 1

# Agent Answers



## Introduction

Unified CCE leverages Artificial Intelligence (AI) and Natural Language Understanding (NLU) to provide services that assist agents. These services are available for the agents in the Cisco Finesse desktop gadgets.

Agent Answers feature provides relevant suggestions and recommendations in real time for the agent to consider. The suggestions and recommendations are based on the ongoing conversation between the caller and the agent.

More often than not, agents lack the depth of knowledge about the products and services of the business they serve. Agent Answers enhances the customer experience because the timely suggestions improve the ability of the agent to respond. Businesses can cut down on training costs and time.

## Prerequisites

The prerequisites for configuring Agent Answers are:

- Virtual CUBE (vCUBE) based on CSR8Kv platforms running the Cisco IOS XE 17.6 image.

  The Cisco IOS XE 17.6.1a image can be downloaded at https://software.cisco.com/download/home/286327102/type/282046477/release/Bengaluru-17.6.1a

  For more details, see the WebSocket-Based Media Forking for Cloud Speech Services chapter in the *Cisco Unified Border Element Configuration Guide - Cisco IOS XE 17.6 Onwards* at https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/voice/cube/ios-xe/config/ios-xe-book/websocket-forking-for-cube.html.

- The following components must be on release 12.6(1): CCE components (Router, Logger, AW, and PG), Cisco Finesse, Cisco Unified CVP, and Cloud Connect.
- Ensure that your Unified ICM AW server has 443/8443 ports opened and is able to access the following websites:
  - *.wbx2.com
  - *.ciscoccservice.com

# Important Considerations

Consider the following before configuring the Agent Answers services:

- Agent Answers services are supported on calls that originate from CVP routing clients. Calls originating from routing clients other than CVP or calls that are sent using the translation route to CVP do not support the Agent Answers services.
- The following failover scenarios don't support Agent Answers services:
  - CCE components running in maintenance modes switch to the peer side, passing the call context to the other side. If the call context (required to trigger the Agent Answers services) is lost, Agent Answers services may not work as expected.
  - Agent Answers services is supported during VRU PG failovers before and after the transfer. However, Agent Answers services aren't supported when the transfer is in progress.
  - Agent Answers services aren't supported during Agent PG failovers.
- Agent Answers services aren't supported in the following call scenarios:
  - Direct Extension calls
  - Outbound campaign calls and agent-initiated outbound calls.
  - Calls routed to agents on non-CUCM Peripheral Gateways such as the TDM PG and System PG
  - Transfer and conference calls
- Agent Answers services are supported only with G.711 μ law.
- A vCube instance can support either WebSocket-based forking or Network-based Recording (NBR) forking. However, you cannot enable both types of forking on the same instance of vCube.

# Contact Center AI Services Task Flow

Follow this procedure to enable the Contact Center AI (CCAI) Services that equips your Contact Center for Agent Answers Services.

### Procedure

**Step 1** Create a CCAI configuration in Cisco Webex Control Hub at https://admin.webex.com. A CCAI configuration leverages CCAI Connectors to invoke the CCAI services.

For details, see the *Create a Contact Center AI Configuration* article.

**Step 2** Ensure that the Cloud Connect publisher and subscriber are installed.

For more information, see the *Install Cloud Connect* section in *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-guides-list.html.

**Step 3** Configure Cloud Connect in the CVP Operations Console (OAMP). For details see the section *Configure CVP Devices for Cloud Connect* in the *Administration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

**Step 4** Register Cloud Connect in the Unified CCE Administration console to establish a secure and trusted communication channel between the Cisco Contact Center on-premises deployment and cloud services.

For details, see the *Cloud Connect Integration* section in the *Administration Guide for Cisco Unified Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-maintenance-guides-list.html.

**Step 5** Import the Cloud Connect certificate to the CVP Server.

For details, see the section *Import Cloud Connect Certificate to Unified CVP Keystore* in the *Configuration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

**Step 6** In the Unified CCE Administration console, do the following with the CCAI configuration (created in step 1):

a) To view and sync the Contact Center AI configuration which is associated with all call types as a global configuration, see Associate Contact Center AI Configuration with All Call Types, on page 4.

b) To view, update, or delete the Contact Center AI configuration associated with a specific call type, see Associate Contact Center AI Configuration with a Call Type, on page 5.

**Step 7** Provision Cloud Connect on Cisco Finesse.

For more information, see the *Cloud Connect Server Settings* topic in the *Cisco Finesse Administration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html.

**Step 8** To add the Agent Answers gadget to the Cisco Finesse desktop layout:

a) Enable the Agent Answers gadget in Cisco Finesse Administration.

For details, see the *Manage Desktop Layout* section in the Cisco Finesse Administration Guide.

b) Enable the Agent Answers service in Unified CCE Administration for an agent or multiple agents together.

For details, see Enable or Disable Contact Center AI Services for Agents, on page 6.

Once enabled, the Agent Answers gadget appears on the Home tab and displays relevant articles and suggestions during an incoming call. For details on how to use the gadget, see the *Contact Center AI Gadgets User Guide for Cisco Contact Center Enterprise*.

**Note** Gadget auto-hide/un-hide and notifications capability is available only if the gadget is configured as a multi-tab gadget in Cisco Finesse. For more details, see *Configure Multi-Tab Gadget Layout* section in the *Cisco Finesse Administration Guide*.

**Step 9** Perform the following steps to configure WebSocket-based forking in CUBE.

a) Create a SIP profile and associate it at the dial-peer level in CUBE. For details, see Create a SIP Profile at the Dial-Peer Level in CUBE, on page 8.

b) Import the WebSocket Connector certificate to CUBE. For details, see Import or Verify WebSocket Connector Certificate to CUBE, on page 8.

c) Configure WebSocket-based forking in CUBE. For details, see the *WebSocket-Based Media Forking for Cloud Speech Services* chapter in the *Cisco Unified Border Element Configuration Guide - Cisco IOS XE 17.6 Onwards*.

CUBE uses a WebSocket connection to fork the media streams of the agent and the caller towards the Webex CCAI Orchestrator service. For more details, see the Contact Center AI Services Considerations section in the *Solution Design Guide for Cisco Unified Contact Center Enterprise* .

# Contact Center AI Configuration

In the Unified CCE Administration console, the Contact Center AI (CCAI) feature tab allows administrators to associate the CCAI configuration (created in the Control Hub at https://admin.webex.com/) with all the call types (global configuration) or with a specific call type. Upon associating a CCAI configuration with the call type, the global configuration (if any) gets overridden for the specific call type.

**Note** To access this feature, add Cloud Connect to the inventory and register it in the Unified CCE Administration console.

# Associate Contact Center AI Configuration with All Call Types

You can view, update, or reset the Contact Center AI configuration, which is associated with all call types.

## View Contact Center AI Configuration

In the **Unified CCE Administration**, navigate to **Overview** > **Features** > **Contact Center AI**. The **Contact Center AI Configuration** search box displays the name of the CCAI configuration that was previously associated with all call types.

## Reset Contact Center AI Global Configuration

This procedure explains how to reset the Contact Center AI configuration. Upon reset, the previously associated configuration is cleared from the search box.

**Procedure**

**Step 1** In the **Unified CCE Administration**, navigate to **Overview** > **Features** > **Contact Center AI**.

**Step 2** In the **Contact Center AI Configuration** search box, next to the configuration name, click the **x** icon.

**Step 3** Click **Save**.

# Associate Contact Center AI Configuration with a Call Type

You can view, update, or delete the Contact Center AI configuration associated with a specific call type.

## View Contact Center AI Configuration

In the **Unified CCE Administration**, navigate to **Overview** > **Call Settings** > **Route Settings** > **Call Type**. The **Contact Center AI Configuration** search box displays the name of the CCAI configuration that was previously associated with the call type.

## Update  Associate Contact Center AI Configuration

You can create a Call Type using the **Configuration Manager** tool. However, you can use the **Unified CCE Administration** to associate a Contact Center AI configuration with a call type.  This procedure explains how to update the Contact Center AI configuration associated with a call type.

**Note** Only one configuration can be associated with a call type.

**Procedure**

**Step 1** In the **Unified CCE Administration**, navigate to **Overview** > **Call Settings** > **Route Settings**.

**Step 2** Click the **Call Type** tab and select the call type for which Contact Center AI configuration has to be associated.

**Step 3** Click the **Contact Center AI** tab.

**Step 4** In the **Contact Center AI Configuration** search box, click the search icon. A pop-up window displays a list of CCAI configurations.

**Step 5** Select the required configuration and click **Save**.

## Reset Contact Center AI Configuration

This procedure explains how to reset the Contact Center AI configuration. Upon reset, the previously associated configuration with the call type is cleared from the search box.

**Procedure**

**Step 1** In the **Unified CCE Administration**, navigate to **Overview** > **Call Settings** > **Route Settings**.

**Step 2** Click the **Call Type** tab.

**Step 3** In the **Contact Center AI Configuration** search box, next to the configuration name, click the **x** icon.

**Step 4** Click **Save**.

# Enable or Disable Contact Center AI Services for Agents

Contact Center AI Services can be configured for each agent. Administrators and supervisors can enable or disable the services for an agent or multiple agents together.

## Enable or Disable Contact Center AI Services for an Agent

This procedure explains how to enable or disable Contact Center AI Services for an agent.

### Procedure

**Step 1** In **Unified CCE Administration**, choose **Users** > **Agents**.

**Step 2** Click on the agent row whose services are to be modified.

**Step 3** Click the **Contact Center AI** tab.
Displays a list of services enabled or disabled for the agent.

**Step 4** To enable or disable the required Contact Center AI Services, check or uncheck the check boxes corresponding to the services.

**Step 5** Click **Save**.

## Enable or Disable Contact Center AI Services for Multiple Agents

Administrators and supervisors can enable or disable Contact Center AI Services for multiple agents.

All agents must belong to the same site and the same department, or all agents must be global agents. The **Edit** button is disabled if:

- Agents from different sites, departments, or peripheral sets are selected.
- A mix of global and departmental agents are selected.

### Procedure

**Step 1** In **Unified CCE Administration**, choose **Users** > **Agents**.

**Step 2** Check the check box corresponding to each agent whose services you want to edit.

**Step 3** Click **Edit** > **Contact Center AI**.
The Edit Services dialog displays a list of services that are  the service that is enabled or disabled.

- If the service is enabled for all the agents selected for editing, the check box is checked.
- If the service is disabled for all the agents selected for editing, the check box is unchecked.
- If the service is enabled for some agents and disabled for the others, the check box has a dash (—).

**Step 4** To enable or disable the Contact Center AI Services, check or uncheck the check boxes corresponding to the services.

**Step 5** Click **Save**, and then click **Yes** to confirm the changes.

## Enable or Disable AnswersContact Center AI Services for Agents using Bulk Job

### Procedure

**Step 1** Navigate to **Unified CCE Administration** > **Overview** > **Bulk Import**.

**Step 2** Click **Templates**.

The **Download Templates** popup window opens.

**Step 3** Click the **Download** icon for the Contact Center AI template you want to use.

**Step 4** Click **OK** to close the **Download Templates** popup window.

**Step 5** Open the `.csv` template in Microsoft Excel.

**Step 6** Populate the file as described in the Bulk Contact Center AI Services Content File, on page 7.

**Step 7** Save the populated file to the local machine.

**Step 8** Navigate to **Unified CCE Administration** > **Overview** > **Bulk Import**.

**Step 9** Click **New**.

**Step 10** In the optional **Description** field, enter up to 255 characters to describe the bulk job.

**Step 11** In the **Content file** field, choose the file to upload, and then click **Save**.

### Bulk Contact Center AI Services Content File

The content file for Contact Center AI bulk job contains the fields given in the following table. Enter the values appropriately in the given fields to enable or disable Contact Center AI Services for the agents.

**Note** Bulk job is available for administrators only when Cloud Connect is added in the inventory and registered on the Control Hub.

| Field | Required? | Description |
|---|---|---|
| agentId | Agent ID or Username | Existing agentId for which you want to enable or disable the Contact Center AI Services.<br>You must provide either an agentId or the userName. If both are provided, agentId takes precedence over the userName. If the agentId value is left blank, the userName will reference an existing agent. |

| Field | Required? | Description |
|---|---|---|
| userName | Username or Agent ID | Username of the agent for which you want to enable or disable the Contact Center AI Services.<br>If no agent is found with the given username, the Contact Center AI Services association fails. |
| agentServices | Yes (to enable Contact Center AI Services) | The type of Contact Center AI Services to be associated with the agent. Supported values are AgentAnswers and Transcript. To associate more than one services, seperate the values using semicolon (;).<br>If the value is updated, any existing enabled service gets overwritten. If the value is left empty, no service gets associated with the agent. |

# Create a SIP Profile at the Dial-Peer Level in CUBE

Run the following CLI commands on the CUBE terminal to create a SIP profile and associate that profile at the dial-peer level. These commands add a SIP header to the SIP profile configuration, allowing CVP to identify which CUBE device can receive the forking request.

```
voice class sip-profiles <SIP-profile-identifier-a>
request INVITE sip-header Call-Info add "X-Cisco-Forking: supported"
dial-peer voice <SIP-profile-identifier-b> voip
voice-class sip profiles <SIP-profile-identifier-a>
```

Example:

```
voice class sip-profiles 104
request INVITE sip-header Call-Info add "X-Cisco-Forking: supported"
dial-peer voice 4445 voip
voice-class sip profiles 104
```

# Import or Verify WebSocket Connector Certificate to CUBE

By default, the trust pool bundle includes the **IdenTrust Commercial** certificate. This certificate is required for validating the **WSConnector** certificate during the TLS connection establishment of the **WebSocket Connector**.

### Procedure

**Step 1** Run the command to verify if the certificate is included.

```
show crypto pki trustpool | include IdenTrust
cn=IdenTrust Commercial Root CA 1
o=IdenTrust Inc
cn=IdenTrust Commercial Root CA 1
o=IdenTrust Inc
```

**Step 2** If **IdenTrust** certificates are not present, add the certificates to CUBE.

a) Open the following URL https://www.cisco.com/security/pki/.

b) Locate the **Cisco Trusted Core Root Bundle** under the **Trusted Root Stores**.

c) Select the **Cisco Trusted Core Root Bundle**, right click, and then select **Copy link**. The URL for the bundle is copied to your clipboard.

d) Run the following command in CUBE terminal:

```
vCUBE# configure terminal
vCUBE(config)# crypto pki trustpool import clean URL <URL copied in step 2(c)>
```

Example:

```
vCUBE(config)# crypto pki trustpool import clean URL
http://www.cisco.com/security/pki/trs/ios_core.p7b
```

Output

```
Reading file from http://www.cisco.com/security/pki/trs/ios_core.p7b
Loading http://www.cisco.com/security/pki/trs/ios_core.p7b
% PEM files import succeeded.
```

The **IndenTrust** certificates are added to CUBE. To verify the addition, run the following command **show crypto pki trustpool | include IdenTrust**. The output will display the **IdenTrust** certificates as shown in Step 1.

# Reconfigure Agent Answers after Upgrade to Unified CCE 12.6

### Before you begin

In Unified CCE 12.6, the CCAI services include the following enhanced capabilities when compared to 12.5:

- **Reporting**: The Agent Answers Analytics report compares an agent's handle time when the Agent Answers service was enabled vs. when the service was disabled. The report helps you understand the impact of the Agent Answers services on an agent's performance.
- **Transfers and Conference Calls Support**: The Agent Answers and Call Transcript services continue during call transfers or call conferences.
- **Enable the CCAI Configuration for Specific Call Types**: The CCAI Configuration can be enabled for all or specific call types.
- **Enable the CCAI Services for Specific Agents**: The CCAI Services can be configured for each agent. Administrators and supervisors can enable or disable the services for an agent or multiple agents.

If you used the CCAI Services on Unified CCE 12.5, you've already completed most of the configurations that are required for the Agent Answers and Call Transcript services to work in Unified CCE 12.6. No changes are required to the existing Google CCAI, CUBE, or Cloud Connect configurations.

While configuring the CCAI services in Unified CCE 12.5, you enabled Cisco Finesse 12.6 to display the CCAI gadgets to all the agents by running the `enableCustomAgentServices` CLI command.

Once you complete the agent configurations (at Step 3 in the following procedure) and run the `enableCustomAgentServices` CLI command to disable the gadgets for all the agents (at Step 4), Cisco Finesse relies on agent-specific configuration in Unified CCE Administration to display the gadgets.

Follow these steps to complete the CCAI configuration in Unified CCE 12.6 and leverage the enhanced capabilities listed above:

### Procedure

**Step 1** In Control Hub, set a CCAI configuration as the default configuration for all calls. For more details, see Step 7a at https://help.webex.com/en-us/npbt02j/Configure-Contact-Center-AI.

**Step 2** In the Unified CCE Administration console, do one of the following:

- To view and sync the Contact Center AI configuration that is associated with all call types as a global configuration, seeAssociate Contact Center AI Configuration with All Call Types, on page 4.
- To view, update, or delete the Contact Center AI configuration that is associated with a specific call type, see Associate Contact Center AI Configuration with a Call Type, on page 5.

**Step 3** In the Unified CCE Administration console, enable or disable CCAI Services for an agent or multiple agents.

For more details, see Enable or Disable Contact Center AI Services for Agents, on page 6.

**Step 4** Undo these CCAI settings configured when Unified CCE was in 12.5:

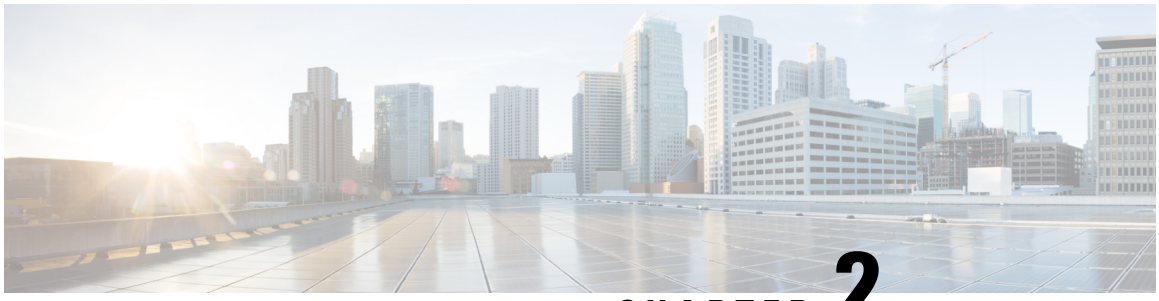
- Delete the "call.user.configid" ECC variable you created for the Answers feature and remove the association of this variable with the CCE script.

  For more details, see the *Contact Center AI Services Task Flow* section in the *Cisco Unified Contact Center Enterprise Features Guide, Release 12.5* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html .

- In Cisco Finesse, run the following CLI command to disable the CCAI services from all the Cisco Finesse clusters:

  ```
  utils finesse set_property webservices enableCustomAgentServices false
  ```

  For more details, see the *AI Services Configuration* topic in the *Cisco Finesse Administration Guide, Release 12.6(1)* at https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html.



CHAPTER 2

# Agent Greeting



## Capabilities

The Agent Greeting feature lets an agent record a message that plays automatically to callers when they connect to the agent. The greeting message can welcome the caller, identify the agent, and include other useful contextual information. With Agent Greeting, each caller can receive a clear, well-paced, language-appropriate, and enthusiastic introduction. Another benefit is that it saves the agent from having to repeat the same introductory phrase for each call. It also gives the agent a moment to review the desktop software screen popups while the greeting plays.

The process of recording a greeting is much the same as recording a message for voicemail. Depending on how the call center is set up, agents may be able to record different greetings that play for different types of callers. For example, agents can record an English greeting for English speakers or an Italian greeting for Italian speakers.

## Agent Greeting Phone Requirements (for Local Agents Only)

Agent Greeting is available to agents and supervisors who use IP Phones with Built-In Bridge (BIB). These agents are typically located within a contact center. Phones used with Agent Greeting must meet these requirements:

- The phones must have the BIB feature.

**Note** If you disable BIB, the system attempts to use a conference bridge for Agent Greeting call flow and raises a warning event.

- In an IPv6-enabled environment, Agent Greeting may require extra Media Termination Points (MTPs).

See the *Contact Center Enterprise Compatibility Matrix* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-device-support-tables-list.html for the list of supported Cisco Unified Call Center phone models.

## Agent Greeting Functional Limitations

Agent Greeting is subject to these limitations.

- Agent Greeting does not support outbound calls made by an agent. The announcement plays for inbound calls only.
- Only one Agent Greeting file plays per call.
- Supervisors cannot listen to agent recorded greetings.
- Agent Greetings do not play when the router selects the agent through a label node.
- Agent Greeting supports Unified CM based Silent Monitoring with this exception: Supervisors cannot hear the greetings themselves. If a supervisor tries to start a silent monitoring session while a greeting is playing, a message displays stating that a greeting is playing and to try again shortly.

## Whisper Announcement with Agent Greeting

You can use Agent Greeting with the Whisper Announcement feature. Here are some things to consider when using them together:

- On the call, the Whisper Announcement always plays first.
- To shorten your call-handling time, use shorter Whisper Announcements and Agent Greetings than if you were using either feature by itself. A long Whisper Announcement followed by a long Agent Greeting equals a long wait before an agent actively handles a call.
- If you use a Whisper Announcement, your agents probably handle different types of calls: for example, “English-Gold Member-Activate Card,” “English-Gold Member-Report Lost Card,” “English-Platinum Member-Account Inquiry.” Therefore, you may want to ensure that greetings your agents record are generic enough to cover the range of call types.

For more information about Whisper Announcement, see Whisper Announcement, on page 245

# Initial Setup

This section is intended for system administrators responsible for installing and configuring Unified CCE. It describes the one-time tasks required to set up Agent Greeting.

## Configuration Requirements

The following configuration components must be in place to deploy Agent Greeting.

| Where | What |
|---|---|
| Unified Communications Manager | For phones that use Agent Greeting, you must set the Built-in-Bridge option to On or De<br>Administration, select **Device** > **Phone** > **Built in Bridge**. |
| Unified CCE | Agent Greeting is supported with Type 10 Network VRUs only. (Type 10 is required to al<br>is not configured for a Type 10 VRU, you must modify it accordingly.<br>Agent Greeting requires at minimum three expanded call variables.<br>• user.microapp.ToExtVXML: This is used twice in an Agent Greeting record script:<br>application; the second time is to tell the recording application where to save greetir<br>Use the Unified CCE Administration tool to ensure this variable includes these setti<br>• user.microapp.app_media_lib:This is required in Agent Greeting record and play scr<br>greeting audio files are stored. Maximum Length - 100 and Enabled.<br>• user.microapp.input_type: This is required in Agent Greeting record scripts to limit th<br>No other ECC (Expanded Call Variable) are needed if you serve your files from the Unif<br>default locale directory ("<*web_server_root*>\en-us\app"). However, if you store your fil<br>the ECC in the next row in your scripts. |
| Unified CCE (optional variables, used to override defaults) | To make these variables available to your script authors, confirm that they are defined in<br>ECC variables for CVP, see the *Administration Guide for Cisco Unified Customer Voice Po*<br>unified-customer-voice-portal/tsd-products-support-series-home.html.<br>• user.microapp.media_server: Use to identify the Unified CVP media server if it is o<br>• user.microapp.locale: Use to specify the name of the locale directory on the media s<br>• user.microapp.UseVXMLParams: Required in your record script if you include the<br>recording script to use the name/value pair of the application that you pass in the us |
| Unified CVP | Unified CVP Server must be installed and configured, as described in the *Cisco Unified*<br>https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-en |

# Deploy Agent Greeting

## Agent Greeting Deployment Tasks

**Procedure**

**Step 1** Ensure that your system meets the baseline requirements for software, hardware, and configuration described in the System Requirements and Limitations section.

**Step 2** Configure one or more servers to act as media servers. Configuration requirements include IIS and FTP.

**Step 3** In Unified CVP, add media servers, configure FTP connection information, and deploy the media servers.

**Step 4** Configure a Unified CVP media server, if you have not already done so. See Configure Media Server for Agent Greeting, on page 14. .

**Step 5** In Unified CVP, republish the VXML Gateway.tcl scripts with updated Agent Greeting support. See Republish the tcl scripts to VXML Gateway, on page 18 for Agent Greeting support.

**Step 6** Set the cache size on the VXML Gateway. See Set Cache Size on VXML Gateway, on page 18.

**Step 7** Record the voice prompts to play to agents when they record a greeting and to deploy the audio files to your media server. See Create Voice Prompts for Recording Greetings, on page 18.

**Step 8** Configure call types to record and play agent greetings. See Configure Call Types, on page 19.

**Step 9** Configure dialed numbers to record and play agent greetings. See Configure Dialed Numbers, on page 20.

**Step 10** Schedule the Script, on page 20.

**Step 11** Define Network VRU Scripts for Agent Greeting, on page 20.

**Step 12** In Script Editor:

- To use the installed scripts to record and play agent greetings, see Import Example Agent Greeting Scripts, on page 22.
- To create your own scripts, see Agent Greeting Scripts, on page 25.

**Step 13** Modify the Unified CCE call routing scripts to use Play Agent Greeting script, on page 24.

## Configure Media Server for Agent Greeting

Agent Greeting uses the Unified CVP media server. If you previously configured and deployed one or more Unified CVP media servers for other features, you do not have to configure any additional servers for Agent Greeting. You can optionally add additional media servers.

Agent Greeting uses the Unified CVP media server to store and serve the following types of files:

- Prompt files, prepared by Administrators. These files supply the prompts that agents hear when they record their greetings. The Administrator must manually add the prompt files to all the media servers that their Agent Greeting scripts will query to retrieve those files.
- Greeting files, recorded by agents. These files are the actual greetings that play to callers. They are recorded by individual agents. The system handles the storage of these files as follows:
  - A greeting file is named using the convention *PersonID_AgentGreetingType*. For more about *AgentGreetingType*, see Specify AgentGreetingType Call Variable, on page 24.
  - When a greeting is first recorded, it is stored temporarily on the Unified CVP Server, where an agent can listen to it before confirming its use.
  - When the agent confirms the greeting, the file is transferred, using FTP, to all media servers that are deployed and are configured with FTP enabled. Make sure that an FTP server is installed and configured for the correct version of IIS on the media server. For instructions, consult your Microsoft documentation (http://microsoft.com).
  - To satisfy a request for the greeting to play to a caller, the greeting file is copied from the media server to the VXML Gateway, where it is cached. The cached copy is used to satisfy subsequent requests for the greeting. Content expires in the cache based on the cache timeout period defined on the media server.

The routing scripts look for the prompt and greeting files either on the configured default Unified CVP media server or on a specific server identified in the script. Some typical scripting scenarios for retrieving files for Agent Greeting include:

- All files are retrieved from the default server.
- All files are retrieved from the default server if available; otherwise, a redundant server is queried.
- For security, the prompt files are retrieved from one server and the greetings files are retrieved from a different server.
- For load balancing, the greetings files are dispersed among several servers and retrieved based on tests in the script.

### *Media Server Hardware and Network Requirements*

Ensure the server is accessible to CVP, Unified CCE, and your agent desktops.

### *Prepare a Media Server*

1. Ensure that IIS is properly configured and running on the server. It must be listening on port 80.
2. Ensure the server is accessible to CVP, Unified CCE, and your agent desktops.
3. Perform the following steps:
   a. On the taskbar, click **Start**, point to **Administrative Tools**, and then click **Server Manager**.
   b. In the **Server Manager** hierarchy pane, expand **Roles**, and then click **Web Server (IIS)**.
   c. In the **Web Server (IIS)** pane, scroll to the **Role Services** section, and then click **Add Role Services**.
   d. On the **Select Role Services** page of the **Add Role Services** wizard, expand **FTP Server**.
   e. Select **FTP Service**.

**Note** To support ASP.NET membership or IIS Manager authentication for the FTP service, you need to select **FTP Extensibility**.

   f. Click **Next.**
   g. On the **Confirm Installation Selections** page, click **Install.**
   h. On the **Results** page, click **Close.**
   i. In the sites section, click **Add FTP Site**. Provide a site name and path to the same location as the http directory c:\inetpub\wwwroot.
   j. Select your desired binding method, specify to start automatically, select **No SSL** and click **Next.**
   k. On the **Authentication and Authorization** section select the type of authentication required. If using basic, note the name and password of the account.
   l. Select the authorization; for anonymous select **Anonymous users**.
   m. Set the read and write permissions.

**Note** Make note of your FTP connection information -- connection type, user name, password, and port number.

4. Make sure that the FTP and the IIS share the same root directory, because the recording application writes the file to the media server directory structure, and the greeting playback call uses IIS to fetch the file. The `en-us/app` directory should be under the same root directory for FTP and IIS.
5. Create a dedicated directory on the server to store your greeting files. This lets you specify a lower cache timeout of 5 minutes for your agent greeting files that does not affect other more static files you may be serving from other directories. By default, the Record Greeting application posts the `.wav` file to the `en-us/app` directory under your web/ftp root directory. You may create a dedicated directory such as `ag_gr` under the `en-us/app` directory, and then indicate this in the Unified CCE script that invokes the recording application. Use the array for the ECC variable **call.user.microapp.ToExtVXML** to send the ftpPath parameter to the recording application. Make sure the ECC variable length is long enough, or it may get truncated and fail.
6. In IIS Manager, set the cache expiration for the dedicated directory to a value that allows re-recorded greetings to replace their predecessor in a reasonable amount of time, while minimizing requests for data to the media server from the VXML Gateway. The ideal value varies depending on the number of agents you support and how often they re-record their greetings. Two minutes may be a reasonable starting point.
7. Also find the site you are using, go to the agent greeting folder you created (ag_gr), and then select **HTTP Response Headers**.
8. Select **Add**, then **Set Common Headers**.
9. Select **Expire Web Content** and set your desired value.

**Note** After specifying the cache timeout, it is a good idea to clear the cache on the VXML Gateway. This ensures the gateway requests the latest files from the media server. You need only clear the gateway cache once. Open a command prompt on the CVP VXML Gateway, log into IOS, and enter the following commands:

```
my_server# conf t
my_server(config)# clear http client cache
my_server(config)# exit
my_server(config)# wr
```

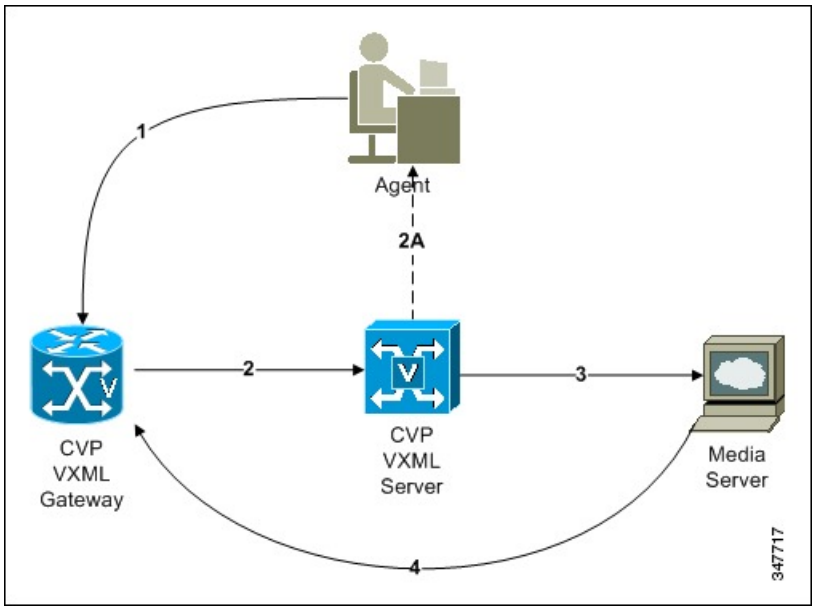
**Note** The HTTP client response timeout setting on the gateway must be greater than the time it takes to complete the largest anticipated FTP file transfer. If an FTP file transfer takes longer than the configured duration in seconds for HTTP client response timeout, the FTP transfer completes correctly, but the call drops as soon as the configured timeout duration is met. To change the HTTP client response timeout setting, open a command prompt on the CVP VXML Gateway, log into IOS, and enter the following commands:

```
my_server# conf t
my_server(config)# http client response timeout <new value in seconds>
my_server(config)# exit
my_server(config)# wr
```

By default, the HTTP client response timeout value for CVP is 30 seconds.

*How Greeting Files Are Recorded and Served*

Following is an illustration of how Greeting files are recorded and served, followed by a step by step description.




1. An agent initiates a greeting recording session and records a greeting.
2. The VXML Gateway passes the recorded (but unsaved) greeting file to the VXML Server.
3. The agent asks to listen to the greeting before saving it. The file is played from the VXML Server.
4. The agent saves the greeting. The file is named (based on the Person ID + AgentGreetingType) and stored on the media server.
5. Requests for the greeting file come in through the VXML Gateway. The VXML Gateway examines its web server cache for the file. If the file is present and not expired, the cached version is served. If the file is not present, or if its timestamp exceeds the cache expiration, the file is retrieved from the media server and cached again.

### Add and Configure Media Servers in CVP

You can add one or more servers to CVP to act as media servers. If you add multiple media servers, note the following:

- CVP automatically propagates files that are added to one media server out to all media servers in the list that have FTP enabled. To enable FTP on a media server, use the following procedure.
- You can designate one media server as the default. If a default media server is defined, requests for files are automatically sent to that server without your having to specify that server in your routing scripts.

1. Access the CVP Operations Console by typing **https://<OAMP_server_IP>:9443/oamp**.
2. At the CVP Operations Console, select **Device Management** > **Media Server**.
3. Add a server to the list of CVP media servers.
4. Select **FTP Enabled**.

5. Configure the credentials and port settings that will permit CVP to write files to the server using FTP.
6. Optionally, you can designate one of your media servers as the Default Media Server.
7. Click the **Deploy** button to deploy the list of media servers to your CVP Servers.

   **Note:** If you deploy the list of media servers and then designate a default, you must redeploy the list.

## Republish the tcl scripts to VXML Gateway

The .tcl script files that ship with Unified CVP include updates to support Agent Greeting. You must republish these updated files to your VXML Gateway.

Republishing scripts to the VXML Gateways is a standard task in CVP upgrades. You must republish the scripts before you can use Agent Greeting.

### Procedure

**Step 1** In the Unified CVP Operation Console, select **Bulk Administration** > **File Transfer** > **Scripts and Media**.

**Step 2** Set Device to Gateway.

**Step 3** Select the gateways you want to update. Typically you would select all of them unless you have a specific reason not to.

**Step 4** Select **Default Gateway Files**.

**Step 5** Click **Transfer**.

## Set Cache Size on VXML Gateway

To ensure adequate performance, set the size of the cache on the VXML Gateway to the maximum allowed. The maximum size is 100 megabytes; the default is 15 kilobytes. Failure to set the VXML Gateway cache to its maximum can result in slowed performance to increased traffic to the media server.

Use the following Cisco IOS commands on the VXML Gateway to reset the cache size:

```
conf t
http client cache memory pool 100000
exit
wr
```

For more information about configuring the cache size, see the *Configuration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/tsd-products-support-series-home.html.

## Create Voice Prompts for Recording Greetings

You must create audio files for each of the voice prompts that agents hear as they record a greeting. The number of prompts you require can vary, but a typical set can consist of:

- A welcome followed by a prompt to select which greeting to work with (this assumes you support multiple greetings per agent)
- A prompt to select whether they want to hear the current version, record a new one, or return to the main menu
- A prompt to play if a current greeting is not found.

To create voice prompts for recording greetings:

**Procedure**

**Step 1** Create the files using the recording tool of your choice. When you record your files:

- The media files must be in `.wav` format. Your `.wav` files must match Unified CVP encoding and format requirements (G.711, CCITT A-Law 8 kHz, 8 bit, mono).
- Test your audio files. Ensure that they are not clipped and that they are consistent in volume and tone.

**Step 2** After recording, deploy the files to your Unified CVP media server. The default deployment location is to the *<web_server_root>*`\en-us\app` directory.

**Step 3** Note the names of the files and the location where you deployed them on the media server. Your script authors need this information for the Agent Greeting scripts.

### *Built-In Recording Prompts*

The Unified CVP Get Speech micro-application used to record Agent Greetings includes the following built-in prompts:

- A prompt that agents can use to play back what they recorded
- A prompt to save the greeting, record it again, or return to the main menu
- A prompt that confirms the save, with an option to end the call or return to the main menu

You can replace these `.wav` files with files of your own. For more information, see the Unified Customer Voice Portal Call Studio documentation at https://www.cisco.com/c/en/us/support/unified-communications/unified-call-studio/tsd-products-support-series-home.html.

### *Example Record Greeting Prompts*

Unified CCE includes three example record greeting audio prompts. These are installed on each ICM server at <icm_root>\wav. These example files are referenced in the example recording script that are included with ICM. If you plan to deploy the example script, copy the audio prompts to the <web_server_root>en-us\app directory on your media server.

## Configure Call Types

To record and play agent greetings, create the following call types: RecordAgentGreeting and PlayAgentGreeting.

**Procedure**

**Step 1** From the Configuration Manager, select **Tools** > **List Tools**.

**Step 2** Select **Call Type List**.

**Step 3** Click **Retrieve**.

**Step 4** Click **Add**.

**Step 5** Create a call type to record agent greetings and use the name RecordAgentGreeting. Then click **Save**.

**Step 6** Create a call type to play agent greetings and use the name PlayAgentGreeting. Then click **Save**.

## Configure Dialed Numbers

To record and play agent greetings, create the following dialed numbers: RecordAgentGreeting and PlayAgentGreeting.

### Procedure

**Step 1** From the Configuration Manager, select **Tools** > **List Tools**.

**Step 2** Select **Dialed Number/Script Selector List**.

**Step 3** Click **Retrieve**.

**Step 4** Click **Add**.

**Step 5** On the **Attributes** tab, do the following:

a) Create a dialed number to record agent greetings; use the name RecordAgentGreeting. (The name must match exactly and is case-sensitive.) Set the **Media routing domain** to **Cisco_Voice**, and then click **Save**.

b) Create a dialed number to play agent greetings; use the name PlayAgentGreeting. (The name must match exactly and is case-sensitive.) Set the **Media routing domain** to **Cisco_Voice**, and then click **Save**.

**Step 6** On the **Dialed Number Mapping** tab, do the following:

a) Click **Add** and map the RecordAgentGreeting dialed number to its call type; click **OK**.

b) Click **Add** and map the PlayAgentGreeting dialed number and its call type; click **OK**.

## Schedule the Script

### Procedure

**Step 1** In the **Script Editor**, select **Script** > **Call Type Manager**.

**Step 2** From the Call Type Manager screen, select the **Schedules** tab.

**Step 3** From the Call type drop-down list, select the call type to associate with the script; for example, PlayAgentGreeting.

**Step 4** Click **Add** and select the script you want from the Scripts box.

**Step 5** Click **OK** twice to exit.

## Define Network VRU Scripts for Agent Greeting

For Agent Greeting record and play scripts to interact with Unified CVP, Network VRU scripts are required. The number of VRU scripts that you require and how you configure them depends on how you choose to script Agent Greeting.

To create these scripts, use the Network VRU Script List Tool found in Configuration Manager.

The following table lists an example set of Agent Greeting Network VRU scripts based on the example Agent Greeting scripts that are included with the software.

**Note** If you require the following example VRU scripts, you must manually create them.

- The Network VRU must be a Type10
- The default timeout 180 is acceptable
- Leave Overridable unchecked

***Table 2: Agent Greeting Network VRU Scripts***

| Name / VRU Script Name | Configuration Parameter | Interruptible (Y/N) | What it does |
|---|---|---|---|
| `AgentGreeting`<br>`PM,-a` | null | N | Causes a saved greeting audio file to play. The `-a` parameter automatically generates the file name by concatenating the Person ID with the AgentGreetingType variable value set in your routing scripts that target an agent. |
| `GreetingMenu_1_to_9`<br>`M,press_1_thru_9_greeting,A` | `1-9` | Y | During a recording session, play an audio file that presents a voice menu prompting the agent to press the number corresponding to the greeting he or she wants to record. The `1-9` configuration parameter defines the range of allowable keys. So this value also determines the number of concurrent greetings agents can have. The `A` parameter specifies that the file is in the (default) Application directory on the Unified CVP Server. |
| `GreetingSubMenu`<br>`M,press1-press2-press3,A` | `1-3` | Y | During a recording session, play an audio file that prompts the agent to press 1 to listen to a greeting, 2 to record, or 3 to go to the main menu. |
| `Greeting_Not_Found`<br>`PM,no_greeting_recorded,A` | `Y` | Y | During a recording session, if an agent tries to play back a greeting that does not exist, play the no_greeting_recorded audio file. The `Y` configuration parameter in this instance allows barge-in (digit entry to interrupt media playback). |
| `T10_GS_AUDIUM`<br>`GS,Server,V, FTP` | `,,,,,,,,,Y` | Y | This starts the external VXML application that records the greeting. The VRU script name must be specified exactly as shown and is case-sensitive.<br>The `Y` parameter in the eleventh position of the Configuration Parameter is required. It allows the script to pass FTP connection information to the VXML server. The VXML server then uses this information to make an FTP connection to the media server when saving greeting files. |

| Name / VRU Script Name | Configuration Parameter | Interruptible (Y/N) | What it does |
|---|---|---|---|
| `GreetingReview`<br>`PM,-a,A` | `Y` | Y | This script allows the agent to review the recorded greeting audio file. |

**Note** For descriptions of VRU Script Name parameters and detailed instructions on creating Network VRU scripts for CVP micro-applications, see the Configuration and Administration Guide for Cisco Unified Customer Voice Portal.

## Import Example Agent Greeting Scripts

To view or use the example Agent Greeting scripts, you must first import them into Script Editor. To import the scripts:

### Procedure

**Step 1** Launch Script Editor.

**Step 2** Select **File** > **Import Script** and select a script to import.

The scripts are located in the `icm\bin` directory on the Unified CCE AW-HDS-DDS.

**Note** When you import the example scripts, Script Editor maps objects that are referenced in the scripts. Some of the objects, such as the external Network VRU scripts, skill groups, route to skill group, or precision queue, do not map successfully. You must create these manually or change these references to point to existing scripts, skill groups, and precision queues in your system.

### What to do next

In addition to importing the scripts, you may need to modify the following items. For more information, see Agent Greeting Scripts, on page 25.

- If you do not use a default media server, you must modify the media server specification.
- If you do not use the default values for application and locale (`en-us/app`), you must modify the path name of greeting files.
- Using the Unified CCE Administration tool, enable all expanded call variables referenced by the following sample scripts.

### *Agent Greeting Example Routing Scripts*

The example routing script files in the `icm\bin` directory include:

- **AG.ICMS—**This script sets up an Agent Greeting by setting the greeting type to be used on the call and then queueing the call to a skill group or precision queue. Once an agent is selected from the skill group

or precision queue and the call routed to the agent, the PAG.ICMS script is invoked. It requires that you define an AgentGreeting VRU script (described in Define Network VRU Scripts for Agent Greeting, on page 20) and a skill group.

- **PAG.ICMS—**This script causes an Agent Greeting to play. It is invoked by the PlayAgentGreeting dialed number that you configured earlier in the configuration process. This number must be associated with a call type that then runs the script. It requires that you define an AgentGreeting VRU script, described in Define Network VRU Scripts for Agent Greeting, on page 20.
- **RECORD_AG.ICMS—**This script lets agents record a greeting. It is called from the agent desktop when an agent clicks the Record Agent Greeting button. It prompts the agent to select which greeting to play or record. This script is invoked by the RecordAgentGreeting dialed number that you configured earlier in this configuration process. It requires that you define all five VRU scripts described in Define Network VRU Scripts for Agent Greeting, on page 20.
- **WA_AG.ICMS—**This script plays a Whisper Announcement and an Agent Greeting together on the same call flow. It requires that you define an AgentGreeting VRU script (described in Define Network VRU Scripts for Agent Greeting, on page 20) and a skill group.

**Note** The PAG.ICMS and RECORD_AG.ICMS example scripts assume that a default media server is configured in Unified CVP, and the greeting files are stored in a dedicated directory named ag_gr directory. The WA_AG.ICMS script does not include a dedicated directory.

**Note** For greeting, the initial script sets up the call between caller and agent, and a different script plays the greeting to the agent after the caller is connected. If the initial Unified CCE script overrides the default media server with a SET node, the call context of expanded call variables is preserved on the greeting playback call as well, and the Default Media Server may be overridden. In this case, modify the greeting playback script to use a SET node with the correct media server.

## Test Agent Greeting File Path

When an agent records a greeting, the greeting file is saved with a system-generated name as follows:

- The Person ID number is prepended to the starting string. For example, an agent with a Person ID of 5050 would have greeting files named 5050_1 or 5050_French.
- The filename ends with the value of the Call.AgentGreetingType variable associated with the choice the agent made when recording the greeting. For example, if the agent selected the first option, and the Agent Greeting record script sets the first option to "1," then the greeting filename is appended with _1. As another example, if descriptive strings were implemented, and the first option is associated with the string "French," then the greeting filename is appended with _French.

The greeting file is saved in a directory whose path is determined by the following variables in the Agent Greeting record script:

- A specific media server, or the default media server. (The file is later pushed to all FTP-enabled media servers.)
- A specific application directory, or the default application directory.

- A specific locale directory, or the default locale directory.

To test the path you defined to the greeting file in your script variables, plug the complete URL into a browser. The `.wav` file should play. For example:

- If your script uses a default media server whose IP is *192.1.1.28 + the default locale + an application directory named greet + 5050_im1.wav*, then the generated URL should be `http://192.1.1.28/en-us/app/greet/5050_1.wav`. Entering this URL into a browser should cause this agent's greeting to play.
- If your script includes: *http://my_server.my_domain.com + the default locale + an application directory app/greet + 5050_1.wav*, then the path should be `http://my_server.my_domain.com/en-us/app/greet/5050_1.wav`.

## Modify the Unified CCE call routing scripts to use Play Agent Greeting script

For an Agent Greeting play script to run, you must add an AgentGreetingType Set Variable node to your existing Unified CCE call routing scripts: This variable's value is used to select the audio file to play for the greeting. Set the variable before the script node that queues the call to an agent (that is, the Queue [to Skill Group or Precision Queue], Queue Agent, Route Select, or Select node).

### *Specify AgentGreetingType Call Variable*

To include Agent Greeting in a script, insert a Set Variable node that references the AgentGreetingType call variable. The AgentGreetingType variable causes a greeting to play and specifies the audio file it should use. The variable value corresponds to the name of the greeting type for the skill group or Precision Queue. For example, if there is a skill group or Precision Queue for Sales agents and if the greeting type for Sales is '5', then the variable value should be 5.

You can use a single greeting prompt throughout a single call type. As a result, use one AgentGreetingType set node per script. However, as needed, you can set the variable at multiple places in your scripts to allow different greetings to play for different endpoints. For example, if you do skills-based routing, you can specify the variable at each decision point used to select a particular skill group or Precision Queue.

**Note** Only one greeting can play per call. If a script references and sets the AgentGreetingType variable more than once in any single path through a script, the last value to be set is the one that plays.

Use these settings in the Set Variable node for Agent Greeting:

- Object Type: Call.
- Variable: Must use the AgentGreetingType variable.
- Type: Must use the PersonID_AgentGreetingType type.
- Value: Specify the value that corresponds to the greeting type you want to play. For example: “2” or “French”
  - You must enclose the value in quotes.
  - The value is not case-sensitive.
  - The value cannot include spaces or characters that require URL encoding.

*Scripting Agent Greeting for Multiple Customers*

In the out-of-box method for deploying Agent Greeting, Unified CCE uses the customer information from the built-in "PlayAgentGreeting" dialed number to choose the correct network VRU to play the greeting. If your deployment has multiple customers configured within your Unified CCE instance and you want to use Agent Greeting with all of them, you must configure things differently to work around customer associations.

Configure Custom Dialed Number for Agent Greeting Play

To play Agent Greetings for multiple customer instances, configure the built-in PlayAgentGreeting dialed number for each Unified CM routing client, but do not associate it with a specific customer. The Unified CM peripheral uses this number to initiate Agent Greeting play. If you want your greetings to be played from a different network VRU, use the TranslationRouteToVRU node in your routing scripts to explicitly choose the network VRU.

Configure Custom Dialed Number for Agent Greeting Record

To record Agent Greetings when you have multiple customers, you must create your own custom dialed number for recording. You may want to create different dialed numbers for different customers. As with Agent Greeting play, if you want to use different network VRUs to record Agent Greetings for different customers, use the TranslationRouteToVRU node in your routing script to explicitly select the network VRU.

Create your own custom button or have your agents enter the record dialed number using the dial pad on their desktops.

## Agent Greeting Scripts

Agent Greeting requires two call routing scripts: one that agents can use to record greetings and one to play a greeting to callers. Examples of these scripts are included in your installation. This section describes the elements in the installed example scripts, including optional features and other modifications that you can make. To create scripts from scratch, use this section to understand the required elements in Agent Greeting scripts.

**Note** If you plan to use the installed example scripts out of the box, you can ignore this section.

### Agent Greeting Recording Script

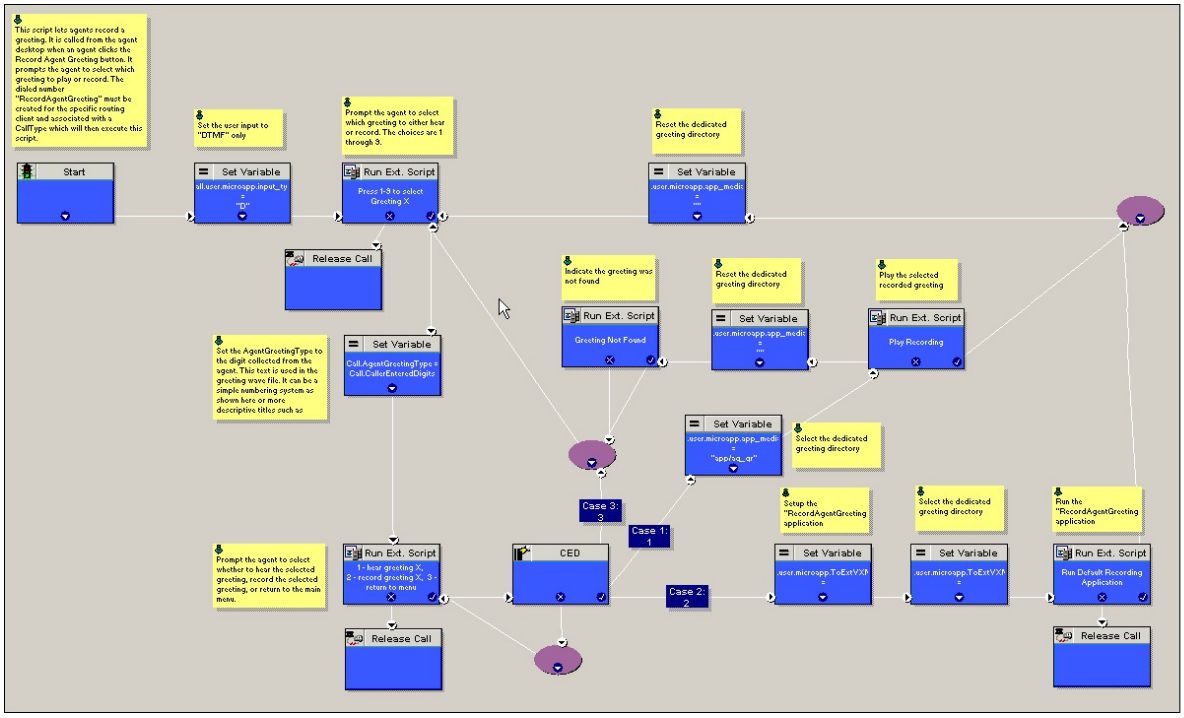
The Agent Greeting recording script is a dedicated routing script that allows agents to record greetings. You can use the installed example scripts or create your own.

In the example script shown here, the agent is first prompted to select one of nine possible greeting types. After selecting a greeting type, the agent chooses whether to 1) listen to the existing greeting for that type; 2) record a new greeting for that type, or 3) return to the main menu. If the agent selects the option to listen, the name of the application directory on the media server is set and the external VRU script that plays the greeting is triggered. Then the agent is returned to the main menu. If the agent selects the option to record, the Unified CVP recording application is called. The recording application contains its own built-in audio prompts that step the agent through the process of recording and saving a greeting. At the end, the agent is returned to the main menu.

There are several other behaviors in the script to note. An agent may select to listen to a greeting type for which no greeting exists. In that event, a VRU script that plays an error message is called. Also, in two places in the script, the path to the application directory is reset to the default. This is because (in this example) that

is where the files for the audio files reside. The only files that reside outside of the default directory are the greetings themselves.

***Figure 1: Agent Greeting Record Script***




## *RecordAgentGreeting Micro-application*

Unified CVP includes a dedicated micro-application -- RecordAgentGreeting -- for recording agent greetings. The application lets agents record, review, re-record, and confirm the save of a greeting. It includes audio files to support each of these functions. If an agent is not satisfied with a greeting, it can be re-recorded up to three times. Upon confirmation of a save, the application FTPs the saved file to the media server.

Built-in error checking includes checks for the data required to name the file (*Person ID* + *AgentGreetingType* variable value), media server specification, valid menu selections made by the agent, and successful FTP of the greeting file.

## *Agent Greeting Record Script Nodes*

Using the example script as a reference, here are descriptions of the functions its nodes perform.

***Table 3: Script Node Functions for Agent Greeting***

| Node | Value | What it does |
|---|---|---|
| Variable:Call:user.<br>microapp.input_type | D | Sets the allowable input type to DTMF (touch tone). |

| Node | Value | What it does |
|---|---|---|
| RunExtScript:Press 1-9 to Select Greeting X | `M,press_1_thru_9_greeting,A` | Runs the VRU script that defines which digits are valid to select an AgentGreetingType and plays a voice prompt describing the options. |
| Variable:Call:AgentGreetingType | `Call.CallerEnteredDigits` | Sets the AgentGreetingType to the digit the agent pressed. This text is used in the greeting wave file. It can be a simple numbering system or more descriptive titles such as “English.” |
| RunExtScript:<br>1 - hear greeting X,<br>2 - record greeting X,<br>3 - return to menu | `M,press1-press2-press3,A` | Runs the VRU script that defines which digits are valid to select a desired action and plays a voice prompt describing the options. |
| CED | `1,2,3` | Tells the script how to handle the caller entered digits in response to the 1,2,3 external script. |
| Variable:Call: user.microapp.app_media_lib | Set three times:<br>• Once to `"app/ag_gr"`<br>• Twice to `""` (an empty string; that is, the default) | Defines the path to the application directory on the Unified CVP media server. Prior to playing the greeting file, it is set to the dedicated greeting file directory (in this example, `app/ag_gr`). After the greeting file plays, it is reset to the default application directory where (in this example) the files for voice prompts are stored. If the voice prompts were stored in the same directory as the greeting files, there would be no need to reset the path. |
| RunExtScript: Play Recording | `PM,-a,A` | Runs the VRU script that plays the selected Agent Greeting. |
| RunExtScript:Greeting Not Found | `PM,no_greeting_recorded,A` | Runs the VRU script that plays an error message if the Agent Greeting selected to play does not exist. |

| Node | Value | What it does |
|---|---|---|
| Variable:<br>Call:user.microapp.<br>ToExtVXML[] | Array Index: 2<br>Value: "ftpPath=<*path_to_dedicated/directory*>"<br>For example: "ftpPath=en-us/app/ag_gr" | Specifies the FTP information that the VXML server uses to write greeting files to the media server. The information must match the FTP information configured for the media server in the Unified CVP Operations Console.<br>The value for array index must be 2.<br>The value consists of:<br>• ftpPath= to set the path to the dedicated directory for agent greeting files.<br>• The path must begin with the locale directory.<br>To view additional setting options, see CVP documentation. |
| Variable:<br>Call:user.microapp.<br>ToExtVXML[] | Array Index: 0<br>Value: "application=RecordAgentGreeting" | Identifies the external Unified CVP micro-application (RecordAgentGreeting) that is used to record the greeting.<br>The value for array index must be 0. |
| RunExtScript: Run Default Recording Application | GS,Server,V | Runs the VRU script that launches the Get Speech micro-application on the VXML server. |

### *Specify Media Server in Routing Scripts*

When you configure media servers in CVP, you can specify a default media server. The benefit to specifying a default media server is that your scripts do not need a Set Variable node to access the default media server. For this to work, you must make sure that the files a script requests are stored on the default server.

If you do not define a default media server, or if you define a default but the files that your script requires are not stored on the default, then the script must include a Set Variable node to identify a media server.

To specify a media server that stores the files required by your script, use the following settings in the Set Variable node:

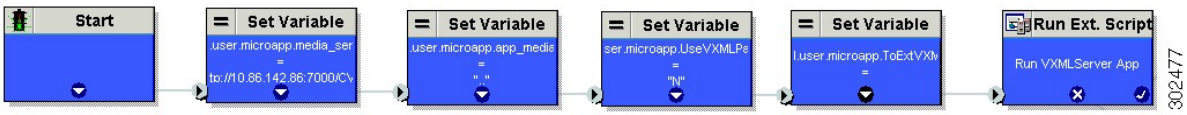
- Object Type: Call.
- Variable: Must use the user.microapp.media_server expanded call variable.
- Value: Specify the HTTP path to the server. For example: “http://myserver.mydomain.net.” You must enclose the path in quotes.
- Alternately you can specify an IP address in place of a hostname.

In scripts that invoke an external VXML application (as the Agent Greeting record script does), if you explicitly set a variable for the media server (user.microapp.media_server), then you must also set the following variables:

- The path to the media server application directory (user.microapp.app_media_lib)
- The CVP UseVXMLParams value to N.(user.microapp.UseVXMLParams)

See the following example.

***Figure 2: Additional Required Variables When Specifying a Media Server***




## *Specify Greeting File Locale and Application Directories in Routing Scripts*

CVP uses a default storage directory for media files: `<web_server_root>/en-us/app`. To take advantage of this, Unified CCE call routing scripts automatically add `en-us/app` to the server name when constructing HTTP requests for media files. For example:

- If the script node that defines the media server has a value of “http://myserver.mydomain.com,” and
- The script node that defines which audio file to play has a value of “5050_1.wav” (for an agent with a Person ID of 5050), then
- The HTTP request for the file is automatically constructed as `http://myserver.mydomain.com/en-us/app/5050_1.wav`

If your greeting audio files are stored in a different locale directory, you must add a Set Variable node to your script that identifies the locale directory. As you must store your greeting files in a dedicated subdirectory under the locale, you must always add a Set Variable node that identifies that directory.

Use these settings in the Set Variable node to specify your locale directory:

- Object Type: Call.
- Variable: Must use the user.microapp.locale expanded call variable.
- Value: Specify the directory name. For example: “pt-br” (Portuguese-Brazil). You must enclose the path in quotes.

Use these settings in the Set Variable node to specify your application directory:

- Object Type: Call.
- Variable: Must use the user.microapp.app_media_lib expanded call variable.
- Value: Specify the directory name. For example: to use a directory “greet” in place of the default directory “app”, enter “greet”. To use a sub-directory “greet” under “app” enter “app/greet”. You must enclose the path in quotes.

## *Verify Length for Media Server Locale and Application Directory Variables*

If you include Set Variable nodes for the media server, locale, and/or application directories, make sure that the values you set for them do not exceed the Maximum Length settings for their corresponding expanded call variables.

For example, if you include a Set Variable node for the media server with a value of "http://mysubdomain.mydomain.co.uk", the string is 33 characters long. Therefore, the Maximum Length setting for the user.microapp.media_server expanded call variable must be 33 or greater. Otherwise, the server name is truncated in the HTTP request for the file and the file is not found.

To configure ECC variables, use the Unified CCE Configuration Manager. Select **List Tools** > **Expanded Call Variables List**.

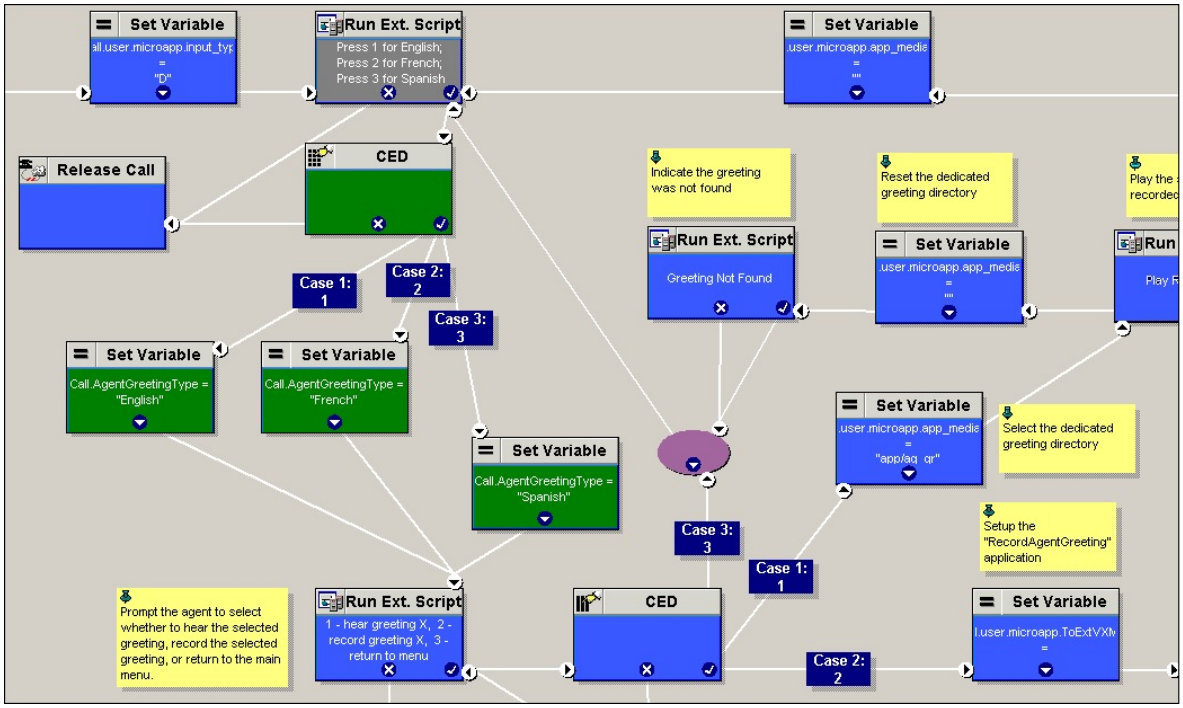
## Descriptive Agent Greeting Type Strings

The previous Agent Greeting record script example stores Agent Greeting Type values as numbers (although in string format). But suppose you prefer more descriptive string names. For example, "English," "French," and "Spanish." Or "Sales," "Billing," and "Tech Support."

Descriptive names can make it easier to understand at a glance what different numeric key selections in your scripts correspond to. Note that they also affect how greeting files are named (for example, for an agent whose Person ID is 5050, `5050_English.wav` as opposed to `5050_1.wav`).

The following script example is almost identical to the previous record script, except that it includes four additional nodes (highlighted in green). They consist of an additional CED node that maps the keys 1, 2, and 3 to language names. The Run Ext Script node (in gray) was modified for the new options. The rest of the script is the same with no other changes required. Note that your routing scripts require a corresponding mapping of numeric keys to language names.

***Figure 3: Script with Descriptive Greeting Type Strings***




## Agent Greeting Play Script

The Agent Greeting feature requires a dedicated routing script that causes the agent greeting to play. This script is invoked by the PlayAgentGreeting dialed number.

The Play script must contain at least two and possibly four specific nodes, depending on other factors.

You always need the following nodes:

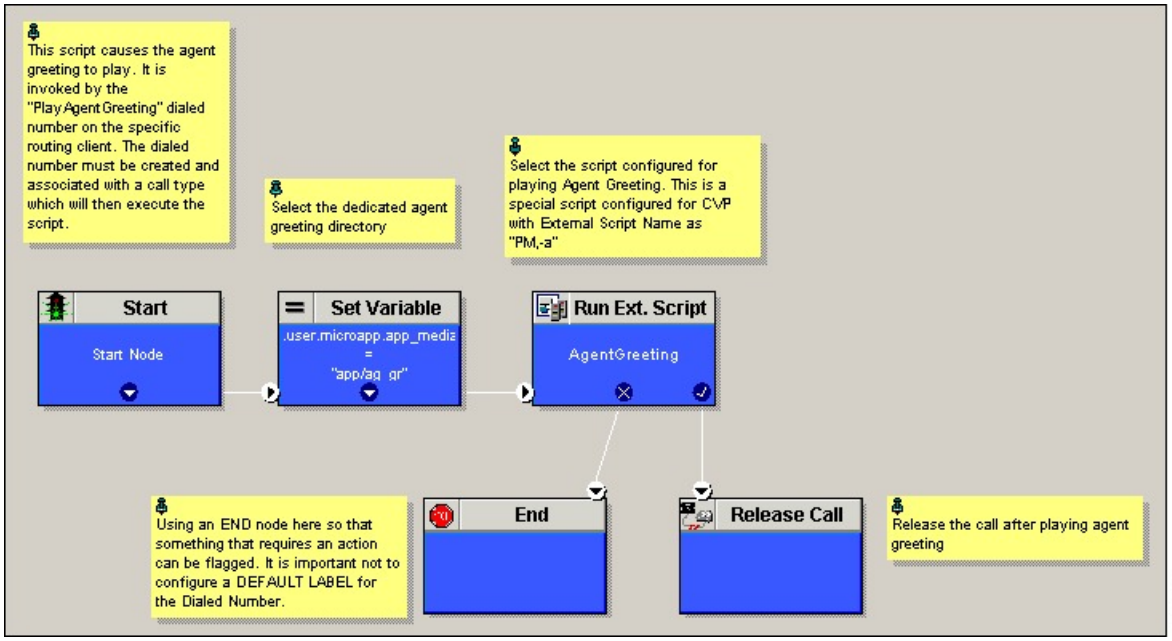
- A Run External Script node that calls the VRU script that plays the greeting.
- A Set Variable node that sets the directory path to your greeting files.

You may also need to include in your scripts Set Variable nodes that:

- Specify the Media Server: Unified CVP lets you specify a default media server. If you are not serving your audio files from the default media server, your scripts must include a variable that identifies the server where your audio files are stored.
- Specify the Locale Directory: Additionally, if you are not storing your files in the default locale directory `en-us` on the media server, you must include a variable that specifies the name of the locale directory where the files are stored.

**Note** The Locale Directory set variable node is optional. It is needed only if you decide to use a directory other than the default one.

***Figure 4: Agent Greeting Play Script Example***




On a Mobile Agent callflow, CUCM may return a 404 error due to the absence of Agent Greeting, leading to call failure. To fix this issue, do the following:

1. Add a new Run External Script node with its backup media mapped to the agent greeting.
2. Add the Run External Script node between the failure path of the AgentGreeting Run External Script node and the End node.
3. Connect the Run External Script node's success path to the existing Release call node and failure path to the existing end node.

Adding the Run External Script node may add a short delay of one to two seconds to the call flow.

# Reporting

In agent, skill group, and precision queue reports, greeting time is not specifically broken out. The period during which the greeting plays is reported as talk time. Record time is counted as an internal call by the default skill group.

Calls that involve Agent Greeting consist of two call legs: the inbound call from the customer and the call to Unified CVP for the greeting. Both of these legs have the same RouterCallKeyDay and RouterCallKey values in the TCD and RCD tables in the database. You can use these values to link the two legs together for reporting purposes.

## Greeting Call Statistics

To view greeting call statistics, create a separate call type and associate it with the routing script that plays agent greeting. New Cisco Unified Intelligence Center templates for the agent greeting call type are created based on the data in the existing Call_Type_Real_Time and Call_Type_Interval table in the database.

## Peripheral Call Types for Agent Greeting

There are two peripheral call types specific to Agent Greeting that you can use to track and report on the feature.

- Call Type 39: Play Agent Greeting. Route request to play an Agent Greeting.
- Call Type 40: Record Agent Greeting. Agent call for recording an Agent Greeting.

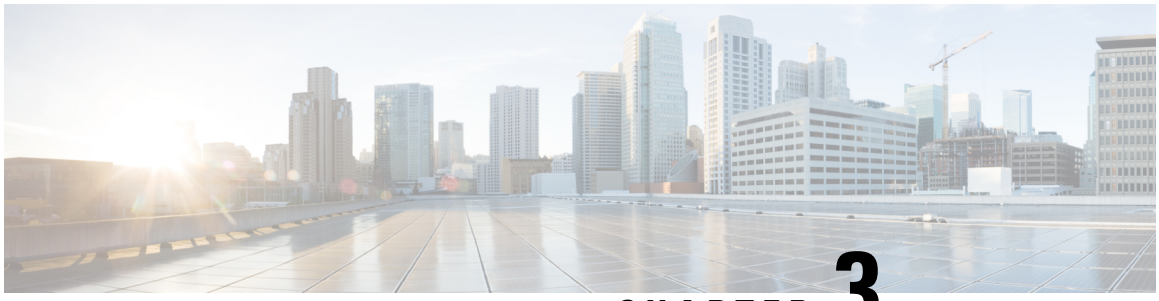
Extra TCDs and RCDs are generated for the agent greeting call leg, and they can be linked to the first call leg by the same RouterCallKeyDay and RouterCallKey.

# Serviceability

Serviceability for Agent Greeting includes SNMP events captured by your Network management software that indicate reasons for greeting failures and counters to track the number of failed greeting events.

**Note** There is no counter for the number of failed agent greeting calls.

When system components fail, Agent Greeting may be impacted. For example, if a requested greeting audio file cannot be found for any reason, the call proceeds without the Agent Greeting.



CHAPTER 3

# Agent Request



# Agent Request Feature Description

The Agent Request feature allows a customer to initiate a request on the web that results in a call from an agent.

To use Agent Request, your solution requires the Cisco Customer Collaboration Platform optional component. Cisco Customer Collaboration Platform works in a Contact Center Enterprise (Unified CCE) solution to process the request from its inception through the delivery of the callback.

**Important** The Agent Request feature can be used only if the customer or a partner develops a custom application. There is sample code on DevNet (formerly Cisco Developer Network) that you can use to understand how to start building your custom application to submit callback requests to Customer Collaboration Platform.

### Customer Collaboration Platform and Agent Request

Customer Collaboration Platform provides the Callback API used by a custom application to request a phone call from a contact center agent.

The API works in conjunction with Customer Collaboration Platform callback feeds, campaigns, and notifications to pass callback requests to the contact center for routing.

The Callback API:

- Allows custom applications to initiate a callback.
- Forwards the callback request and callback details to Unified CCE using a notification mechanism (the Connection to Unified CCE notification type) through a Media Routing (MR) connection.

- Allows custom applications to retrieve the state of the callback as well as the estimated wait time (EWT) until an agent becomes available.
- Allows custom applications to cancel a requested callback.

The Callback API supports the use of Call variables and ECC variables for callback requests. Call variables and ECC variables send customer-specific information with the request. When you create a callback contact, the social contact associated with the callback contact includes all of the specified variables as extension fields.

### Unified CCE and Agent Request

When it receives an Agent Request, Unified CCE performs these tasks:

- Process the callback request.
- Route the callback request to an agent and place a call from the agent's phone to the customer.
- Notify Customer Collaboration Platform that the agent has been selected.

### Agent Desktops and Agent Request

Cisco Finesse supports Agent Request.

### Enterprise Chat and Email and Agent Request

To configure prefixes and filters for dialed numbers in Enterprise Chat and Email, you must use the Unified CCE Script Editor.

### Unsupported Environments

Agent Request is not supported:

- In a Parent/Child deployment
- With Mobile Agents
- In a hybrid deployment

# Agent Request Prerequisites

Install and configure Customer Collaboration Platform before implementing Agent Request. Customer Collaboration Platform must be geographically colocated with one side of the Media Routing Peripheral Gateway (MR PG).

The customer or partner must build a custom application for the Agent Request feature. See Use the Sample Code to Create a Customer Callback Request, on page 42.

Customer Collaboration Platform is always deployed in a DMZ. Remember to open the port you have configured for the MR PG. See Set up the Media Routing PG and PIM, on page 38.

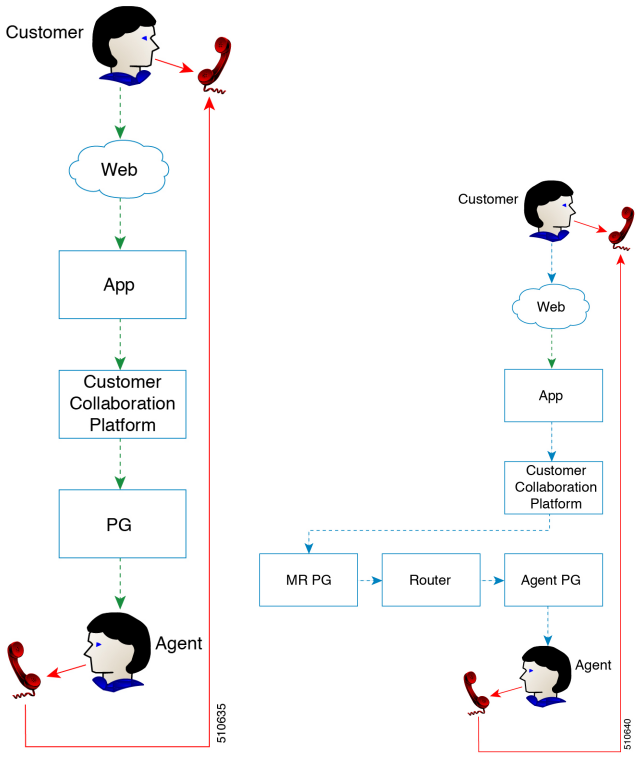
# Agent Request Call Flow

The flow proceeds as follows:

1. The customer application initiates an agent request by requesting a callback.
2. Customer Collaboration Platform sends the request to the Media Routing PG.
   a. The Media Routing PG sends the request to the Router.
   b. The Router sends the request to the Agent PG.
   c. The Agent PG sends the request to the agent.
3. A call is initiated from the agent's phone, on behalf of the agent, dialing the customer's phone number.

**Note** The agent does not control when the call is placed.

***Figure 5: Agent Request Call Flow***




# Agent Request Scenarios

1. From the web, the customer requests to speak to an agent.
2. The customer receives feedback that the request is accepted.
3. The customer receives feedback that the call is queued and the estimated wait time.
4. The customer receives feedback that a call is on its way.

5. The agent's phone places an outbound call.
6. The agent is presented with call context.

| If | Then |
|---|---|
| The customer is available | The customer receives and answers the call, and speaks to the agent |
| The customer is busy when the callback occurs | The agent receives a busy tone |
| The customer does not answer when the callback occurs | The agent hears ringing |
| The customer cancels the callback before an agent is selected | There is no impact on the agent |

# Configure Unified CCE for Agent Request

The following information describes how to configure Agent Request for a Unified CCE deployment.

**Important** Configure Unified CCE before you configure Customer Collaboration Platform.

## Configuration Manager

Use these Configuration Manager tools and procedures to configure Agent Request.

### Configure Network VRU and Network VRU Script

**Procedure**

**Step 1** In the Configuration Manager, use the Network VRU Explorer tool to configure and save a type 2 VRU. The Network VRU is used to queue voice callback tasks if an agent is not available to handle them.

**Step 2** In the Configuration Manager, use the Network VRU Script List tool to add a Network VRU Script that references the Network VRU that you configured in Step 1.

The Network VRU Script is used for Estimated Wait Time.

## Configure the Media Routing PG and PIM

**Procedure**

**Step 1** In Configuration Manager, open the PG Explorer tool to configure a media routing PG.

**Step 2** Create a media routing PIM and routing client for Customer Collaboration Platform.

Write down the Logical Controller ID and the Peripheral ID. You will use them when you set up the PG.

**Step 3** On the Peripheral tab in the PG Explorer tool, check the **Enable post routing** check box.

**Step 4** On the Routing Client tab in the PG Explorer tool, select the **Multichannel** option from the **Routing Type** drop-down list box.

**Step 5** On the Advanced tab in the PG Explorer tool, select the type 2 Network VRU that you created.

## Configure Call Type

**Procedure**

Open the Call Type List tool, and create a call type to handle calls from an agent request voice callback.

## Configure Dialed Number/Script Selector

**Procedure**

**Step 1** Open the Dialed Number/Script Selector List tool, and create a script selector on the routing client that you configured. Customer Collaboration Platform uses this script selector to request agents for voice callback. (The script selector configured here must be the same as the one entered in the Customer Collaboration Platform notification.)

**Step 2** On the Attributes tab, select **Cisco_Voice** from the **Media routing domain** drop-down list box.

**Step 3** On the **Dialed Number Mapping** tab, map the script selector to the call type you created.

## Configure ECC Variables

**Procedure**

**Step 1** Open the **Expanded Call Variable List** tool.

**Step 2** Add one or more ECC Variables for the callback request.

**Note** Arrays are not supported with the Agent Request feature.

CCE solutions support the Latin 1 character set only for Expanded Call Context variables and Call variables when used with CVP, Finesse, and Customer Collaboration Platform. CCE also supports the use of multi-byte character sets in limited usage for ECC and call variables when setting them in Script Editor using double quotes.

# Set up the Media Routing PG and PIM

**Procedure**

**Step 1** Navigate to **Unified CCE Administration** > **Overview** > **Infrastructure Settings** > **Peripheral Gateways**. Determine the Peripheral ID for a Multichannel peripheral for the customer collaboration platform, as mentioned in the *Cisco Packaged Contact Center Enterprise Administration and Configuration guide*.

**Step 2** From Cisco Unified CCE Tools, select **Peripheral Gateway Setup**.

**Step 3** On the Components Setup screen, in the Instance Components panel, select the PG Instance component. If the PG does not exist, click **Add**. If it exists, click **Edit**.

**Step 4** In the Peripheral Gateways Properties screen, click **Media Routing**. Click **Next**.

**Step 5** Click **Yes** at the prompt to stop the service.

**Step 6** From the Peripheral Gateway Component Properties screen, click **Add**, select the next PIM, and configure with the Client Type of Media Routing as follows.

a) Check **Enabled**.
b) In the **Peripheral Name** field, enter `MR`.
c) For **Application Hostname (1)**, enter the hostname or IP address of Customer Collaboration Platform.

**Note** The system does not support IP address change. Use the hostname if you foresee a change in IP address. This is applicable for all the **Hostname/ IP Address** fields.

d) By default, Customer Collaboration Platform accepts the MR connection on **Application Connection Port** 38001. The Application Connection Port setting on Customer Collaboration Platform must match the setting on the MR PG; if you change the port on one side of the connection, you must change it on the other side.
e) Leave the **Application Hostname (2)**, field blank.
f) Keep all other values.
g) Click **OK**.

**Step 7** On the Peripheral Gateway Component Properties screen, enter the Logical Controller ID that you recorded when you configured the Media Routing PG and PIM.

**Step 8** Accept defaults and click **Next** until the Setup Complete screen opens.

**Step 9** At the Setup Complete screen, check **Yes** to start the service. Click **Finish**.

**Step 10** Click **Exit Setup**.

**Step 11** Repeat from Step 1 for Side B.

**Step 12** Navigate to **Unified CCE Administration** > **Infrastructure Settings** > **Inventory**.

**Step 13** Add Customer Collaboration Platform as an external machine.

a) Click Add.
b) Select Customer Collaboration Platform from the drop-down list.
c) Enter the required information.
d) Click **Save**.

The system automatically enables and completes the **CCE Configuration for Multichannel Routing** settings in Customer Collaboration Platform Administration, including the **Application Connection Port** you specified.

# Configure Customer Collaboration Platform for a Voice Callback Agent Request

To support a callback request, Customer Collaboration Platform must be configured with:

- A callback feed
- A campaign
- A Connection to CCE notification configured for the campaign mentioned above that will be triggered by incoming callback requests with a matching tag.

## Create Feed

**Procedure**

**Step 1** Sign in to Customer Collaboration Platform.
**Step 2** Click **Configuration**.
**Step 3** On the **Manage Feeds** panel, click **New**.
**Step 4** For **Type**, select **Callback**.
**Step 5** Name the feed.
**Step 6** For **Reply Template**, retain the default, *No reply template*.
**Step 7** Configure the feed to automatically tag all callback requests that come in on that feed. For example, autotag with 'sendtocontactcenter'.
Make a note of the tag. It is used to trigger the notification to CCE.
**Step 8** Click **Save**.

## Create Campaign

**Procedure**

**Step 1** Sign in to Customer Collaboration Platform.

**Step 2** Click **Configuration**.
**Step 3** On the **Manage Campaigns** panel, click **New**.
**Step 4** Name the campaign.
**Step 5** Enter an optional description.
**Step 6** Make no selection in the **Chat Invitation Feed** drop-down list.
**Step 7** Locate the Callback feed in the **Available** panel and move it to **Selected**.
**Step 8** Click **Save**.

## Create Notification

**Procedure**

**Step 1** Sign in to Customer Collaboration Platform.
**Step 2** Click **Administration**.
**Step 3** On the **Manage Notifications** panel, click **New**.
**Step 4** For **Type**, select **Connection to CCE**.
**Step 5** Name the notification.
**Step 6** From the **Campaigns** drop-down list, select the campaign that you created for the callback.
**Step 7** In the **Tags** field, enter the tag that is automatically applied to callback requests by the feed. In our example 'sendtocontactcenter'.
**Step 8** For **Request Type**, select **Callback**.
**Step 9** In the **Dialed Number/Script Selector** field, enter the dialed number string that you have configured.

See Configure Dialed Number/Script Selector, on page 37.

**Step 10** Click **Save**.

# Agent Request Script

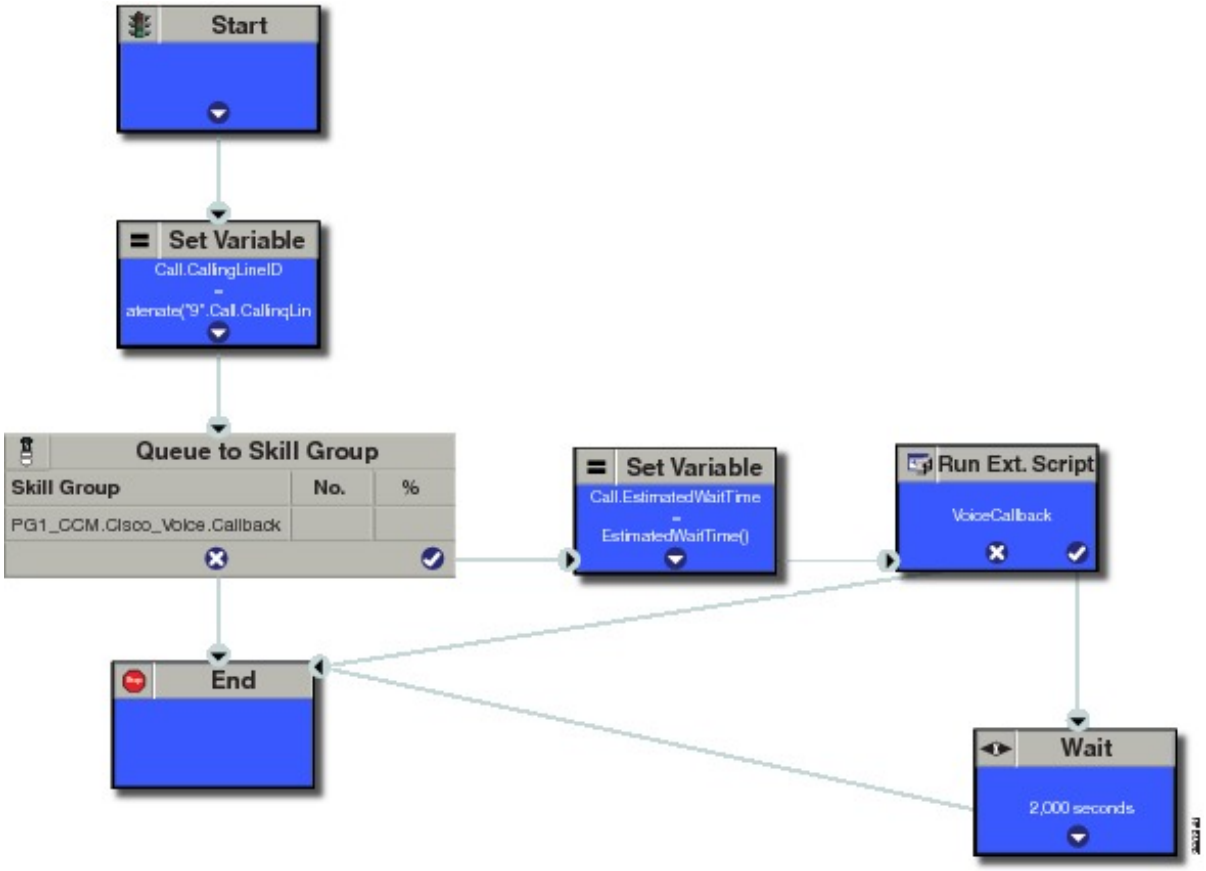
The Agent Request Script is used to implement an optional dial-plan, queue a call to the skill group node, and calculate the estimated wait time for a voice callback to a customer.

A customer who requests a voice callback might want to know approximately how long it will be before the call is returned. You can configure voice callback to provide an estimate of the wait time back to the customer. The estimated wait time is calculated once, when the call enters the queue. The time is not updated as the position in the queue changes.

The default estimated wait time algorithm is based on a running five minute window of the rate of calls leaving the queue. Any calls that are routed or abandoned during the first five minutes are taken into account as part of the rate leaving queue. For Precision Queues, the rate leaving queue represents the rate at which calls are delivered or abandoned from the entire Precision queue, not any individual Precision Queue steps. The algorithm computes the wait time for each of the queues against which the call is queued (Skill Groups or Precision Queues) and then returns the minimum estimated wait time. Queue to Agent is not supported.

While the queue builds, the small number of calls in the queue makes the estimated wait time less accurate and the value fluctuates rapidly. As the queue operates with more calls over time, the estimated wait time is more accurate and consistent.

**Note** The built-in function also applies to inbound calls that queue.




# Create Agent Request Script

To create a Agent Request script:

**Procedure**

**Step 1** **Start node:** Create the **Start** node by selecting a new Routing Script from the Script Editor.

**Step 2** **Set Variable (Call.Calling Line ID) node:** (optional). If required, you can set the CallingLineID (CLID/ANI) variable to implement a "dial-plan," pre-pending a set of digits to the phone number provided by the customer so that it can be correctly routed. For example, it is often necessary to add 9 to the phone number to reach an outside line. In other cases, more pre-pended digits may be required to reach the end customer.

You can also set up Unified Communications Manager Route Patterns to respond to a certain set of digits by routing the call to an outside line with a specified area code. To implement a dial-plan, add a Set Variable node before the queue, as shown in this example. In this case, a 9 is pre-pended to the customer phone number using the built-in concatenate function.

**Step 3** **Queue to Skill Group node:** The Agent Request call can be queued against one or more Skill Groups, Precision Queues, or a queue-to-agent node. In the example script, the call is queued against a single skill group.

**Step 4** **Set Variable (Call.Estimated Wait Time) node:** Set the Call Wait time as follows:

a) From the Set Variable node, select **Call** from the **Object type** drop-down menu.
b) From the **Variable** drop-down menu, choose **Estimated Wait Time()**. You can then work with the Formula Editor to use the default estimated wait value or create a formula and use your own value.
c) Click **Formula Editor**. You can either use the default estimated wait value, by clicking the **Built-In Functions** tab and choosing **EstimatedWaitTime()** . Or to create a formula and use your own estimated time value, click the **Variables** tab, and choose an entry in the **Object type** list and **Object** list. Then double-click a variable in the **Variable** list.

**Step 5** **Run Ext Script node:** Apply the **Network VRU** script as follows:

a) Click the **Queue** tab.
b) Click **Run External Script**.
c) Click inside the script. A Run External Script node appears.
d) Double-click the node and choose the Network VRU script from the list and then click **OK**. The call variable Estimated Wait Time now contains a value in the **EstimatedWaitTime** field and can be passed to peripherals.

Note that a Run External Script node is required to send the EstimatedWaitTime to Customer Collaboration Platform.

**Step 6** **Wait node:** The wait period for an agent becomes available.

**Step 7** **End node:** The script ends if no agent becomes available.

# Use the Sample Code to Create a Customer Callback Request

Cisco Systems has made sample callback application code available to use as a baseline in building your own application. This sample includes retrieving and displaying the estimated wait time, assuming it has been configured in Unified CCE. You can find the sample code on DevNet.

**Note** You cannot copy and paste this code to achieve a working application. It is a only a guideline.

For more information about how to use the Callback API, see the Cisco Customer Collaboration Platform Developer Guide.

### Procedure

**Step 1** Retrieve the feed id by entering this URL in a browser: **https://<Customer Collaboration Platform_Hostname_or_Ip>/ccp-webapp/ccp/feed**.

In the example output below, note that the value in the <name> field is "Callback." Look for the number of the feed id identified at the end of the refURL path (in this case, it is 100000) just before the </refURL> tag. Copy this number.

```
<feeds>
<Feed>
<changeStamp>0</changeStamp>
<name>Callback</name>
<pushFeedURL>https://128.107.81.27/ccp/callback/feed/100000</pushFeedURL>
<refURL>https://128.107.81.27/ccp-webapp/ccp/feed/100000</refURL>
<status>1</status>
<tags>
<tag>trial</tag>
</tags>
<type>10</type>
</Feed>
</feeds>
```

**Step 2** Access the sample application from DevNet: https://developer.cisco.com.

**Step 3** Enter values in the fields:

- Title: A title or subject for the callback request.
- Author: The name of the person submitting the callback request.
- Phone: The phone number to call back.
- Feed Id: The value from the refURL above.

**Step 4** Click **Call me back**.

# Agent Request Reporting

Cisco Unified Intelligence Center CCE reports include data for Agent Requests

**Note** Agent requests that fail before being routed to CCE will not be included in the CCE solution-level reports. The Customer Collaboration Platform search function can be used to identify these requests.

### Call Type and Call Type Skill Group Metrics

- **Calls Offered** — Incremented when Call Type is entered (through Script Selector or Call Type node).
- **Calls Abandoned in Queue** — Incremented when a Queued Callback request is canceled by the customer prior to when an Agent is selected to handle the Voice Callback call.

- **Calls Answered** — Incremented if the call is placed from the agent and represents work accepted by the agent.
- **Calls Handled** — Incremented if the customer answers the call. Calls Answered minus Calls Handled indicates how many calls failed to reach the intended customer.
- **Service Level Offered** — Incremented for all routed calls, including voice callback calls initiated through the agent request API.
- **ServiceLevelCalls** — Incremented if the call is presented to the agent within a service level.
- **Answer Intervals (1 - 10)** — The appropriate bucket is incremented based on how long the call was in the queue.

### Skill Group Metrics

Call Type Skill Group and Skill Group metrics are not counted in the same way. The skill group metric treats each call as agent-initiated; therefore, Calls Answered and Calls Handled are not incremented. AgentOutCallsTime, AgentOutCalls, AgentOutCallsTalkTime, AgentOutCallsOnHold, and AgentOutCallsOnHoldTime are incremented.

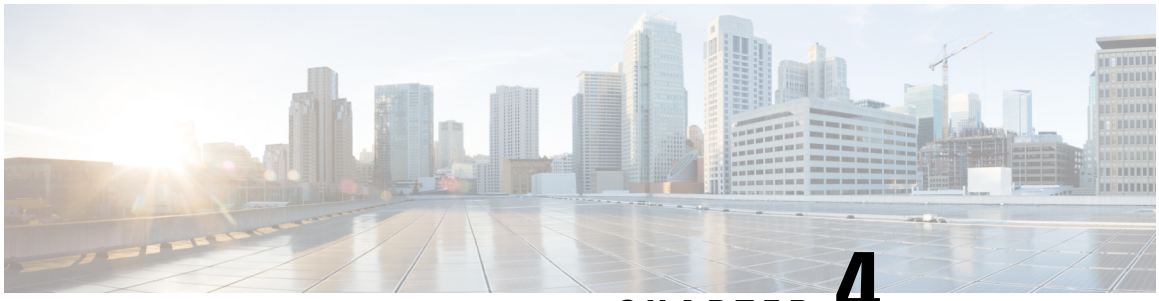
### Agent Real Time

The direction in the Agent Real Time table is listed as Outbound.

### Termination Call Detail

For custom reporting, the Termination Call Detail records contain a PeripheralCallType of 41 -Voice Callback.

Calls which do not successfully connect to a customer have a call disposition of **10 - Disconnect/Drop no answer**. This includes agent request calls to busy numbers.



CHAPTER 4

# Business Hours



## Business Hours Overview

The Business Hours feature lets you create schedules for regular working hours and extra working hours, and to close the contact center for holidays or emergencies. It provides the mechanism for routing these contacts to specific support teams based on the configured work hour schedules, holidays, emergency closures, or extra working hours. You can create Business Hour schedules for various scenarios for various contact center teams. This feature helps you create and apply several Business Hour schedules to the same team. On the other hand, you could apply the same Business Hour schedule to several support teams.

When a customer contacts the contact center, the response by the contact center is based on the status of the support team. This status is evaluated using the Business Hour configured for the team.

Use this feature to:

- Configure default working hours (regular hours) for contact center teams for each day of the week. This option is not applicable to 24x7 support teams.
- Configure the special hours for the contact center team or teams for any special days such as Sale days or holidays.
- Force Close the contact center for any emergency such as a natural calamity.
- Force Open the contact center on a holiday or a non-working day to cater to specific business requirements such as Sale days.
- Create and deploy customer notifications that are based on Business Hour status.

Define the status reasons for business hours and assign codes for each status reason. Status reason is required when you force open or force close a business hour, and when you add special hours and holidays.

### Contact Center Enterprise Reference Design Support for Business Hours

Unified CCE supports Business Hours for these reference designs:

- 2000 Agents

- 4000 Agents
- 12000 Agents
- 24000 Agents

# Business Hours Use Cases

Use the Business Hours feature to manage the incoming customer calls or digital channel contacts to your teams , by routing these contacts based on the Business Hours you configure.

Use the Business Hour status in an IF node in scripts to control the call and digital channel contacts, such as email and chat, and notify the customers accordingly.

You can have Business Hours scripts for the following treatments:

- When the business is open, route calls and digital contacts to the applicable skill groups and precision queues.
- When the business is closed, play the message for the closed status with the appropriate Status Reason and terminate the call. Route the digital contacts to the appropriate queues.
- When the business is not open 24x7, route the calls to skill groups and precision queues for after-hours support or play the after-hours message.
- When the business is open 24x7, at a predefined time before the end of each shift, route the calls and digital contacts to the appropriate queues for the next shift.
- When the business is closed for an emergency on a working day, notify the customers contacting your contact center appropriately about the emergency closing.

Based on reason code and status, the customers will hear appropriate prompts on the call.

# Set the Principal AW for Business Hours

You must specify and set the Principal AW before configuring the Business Hours.

The Principal AW (Admin Workstation) is responsible for managing background tasks that are run periodically to sync configuration with other solution components, such as SSO management, Smart Licensing, etc.

### Procedure

**Step 1** In Unified CCE Administration, go to **Infrastructure** > **Inventory**.

**Step 2** Click the AW that you want to set as **Principal AW**.
The Edit CCE AW window opens.

**Step 3** Select the **Principal AW** check box.

**Step 4** Enter the Unified CCE Diagnostic Framework Service domain, username, and password.

**Step 5** Click **Save**.

**Step 6** Restart Tomcat service on **Principal AW** machine.

# Business Hours Set Up Workflow

This section provides information necessary to set up the Business Hours feature.

***Table 4: Business Hours Set Up Workflow***

| Tasks | Documentation |
|---|---|
| Scripting for Business Hours | *Business Hours Variables* and *Dynamic Formula for Business Hours* in the Scripting and Media Routing Guide for Cisco Unified ICM/Contact Center Enterprise at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-user-guide-list.html |
| Business Hours Configuration | Administration Guide for Cisco Unified Contact Center Enterprise at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-maintenance-guides-list.html |

### Scripting for Business Hours

To enable scripting for Business Hours, use the following variables:

- Business Hour Status—The real time status based on the configured Business Hours.
- Reason Code— The reason code associated with the current status.

These variables must be used in the CCE script IF node to route the customer contacts.

### Business Hours Configuration

Business hours can be configured by defining the following:

- **Default Open/Close (as per Business Calendar)**: The status of the regular hours.
- **Force Open**: Force the status of Business Hours to Open with a reason code.
- **Force Closed**: Force the status of Business Hours to Closed with a reason code.
- **24x7** setting: Set the schedules for 24x7 working. Status is Always Open.
- **Special Hours & Holiday**: Configure a holiday or special day. On a holiday, you can close the Business Hours for the whole day or open it for specific hours. On a special day, you can configure extra working hours (in cases where 24x7 working is not set), the evaluation stops when the Special Hours & Holiday step is evaluated and the status is derived from this step. Further Business Hours setting for that day will not be evaluated.
- **Custom**: Configure the regular working hours for a weekday and, if necessary, specify the working hours for each day in a week.

The following variables control this feature:

- **Time Zone**: The line of Business or team's operational time zone.

- **Reason Code**: Reason code for special days and Force Open/Close status.

  Apply the appropriate reason codes when you configure a Business Hour with **Force Open**, **Force Close**, and **Special Hours & Holidays**. When these Business Hour schedules are in effect, the associated reason code is available in the scripting environment and the real time reports. **Reason Code** configuration is not available for regular weekdays. In such cases, the system reports the default open and default close reason codes.

### Configure Yearly Schedules

You can configure and maintain a Business hours calendar for the whole year.

- Configure the regular working hours for weekdays.
- Configure **Special Hours & Holidays** schedules for whole year by doing the following:
  - Add the **Special Hours & Holidays** details for all the special hours and holidays for the whole year into the CSV template file.
  - On the **Import Special Hours & Holidays** page, click **Choose File** and browse to the special hours and holidays file.

    Click **Import** to upload the file.

    After you import the configuration file, the configurations are loaded on the Business Hours page. Validate the configurations.
  - Click **Save**.

**Note** When you update the configured Business Hours, remove any elapsed schedules and then update the new schedules for any new special hours or holidays in a Business Hour configuration.

### Daylight Saving Time

The Business Hours feature uses **`Time Zone`** to determine the status based on the Business Hours configured. **`Time Zone`** is set based on the local time. When the daylight saving time (DST) settings are applicable to any **`Time Zone`**, the status is automatically adjusted for DST.

When time zones are added or updated, the DST information changes. Apply all the Operating System (OS) updates related to time zones and DST, to both sides A and B of the deployment.

### Business Hours Status Evaluation

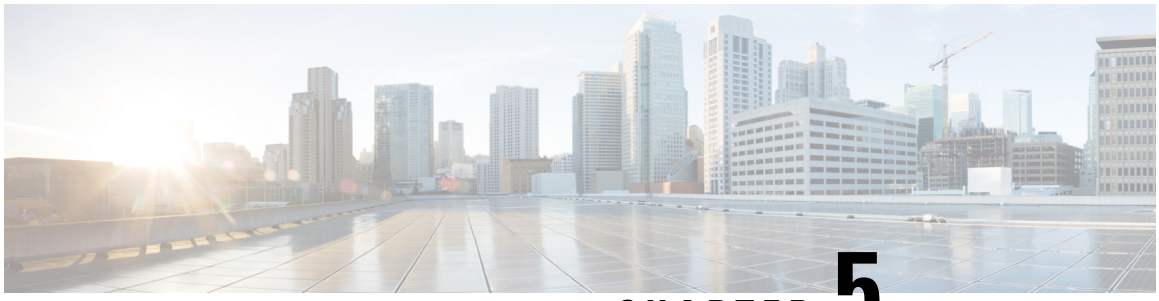
The status is evaluated using the following order of the configured Business Hour settings:

1. Open/Close as per Business Calendar
2. Force Open or Close
3. Special Hours and Holiday schedule
4. 24x7 setting
5. Regular hour schedules

The evaluation terminates at the step at which the status is determined.

For example, if **Special Hours & Holidays** is configured, the evaluation stops when the **Special Hours & Holiday** step is evaluated and the status is derived from this step.

If any configuration is changed, the status is re-evaluated and updated to reflect the change.



CHAPTER 5

# Call Transcription



## Introduction

Unified CCE leverages Artificial Intelligence (AI) and Natural Language Understanding (NLU) to provide services that assist agents. These services are available for the agents in the Cisco Finesse desktop gadgets.

In the **Transcript** gadget, you can view in real-time, the voice conversation that was dynamically converted to text.

## Prerequisites

The prerequisites for configuring Call Transcription are:

- Virtual CUBE (vCUBE) based on CSR8Kv platforms running the Cisco IOS XE 17.6 image.

  You can download the Cisco IOS XE 17.6 image from https://software.cisco.com/download/home/286327102/type/282046477/release/Bengaluru-17.6.1a.

  For more details on WebSockets support for media forking on Cisco 4431, 4451-X, and 4461 Integrated Services Routers, see the WebSocket-Based Media Forking for Cloud Speech Services chapter in the *Cisco Unified Border Element Configuration Guide - Cisco IOS XE 17.6 Onwards* at https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/voice/cube/ios-xe/config/ios-xe-book/websocket-forking-for-cube.html.

- The following components must be on release 12.6(1): CCE components (Router, Logger, AW, and PG), Cisco Finesse, Cisco Unified CVP, and Cloud Connect.

## Contact Center AI Services Task Flow

Follow this procedure to enable the Contact Center AI (CCAI) Services that equips your Contact Center for Call Transcription Services.

### Procedure

**Step 1** Create a CCAI configuration in Cisco Webex Control Hub at https://admin.webex.com. A CCAI configuration leverages CCAI Connectors to invoke the CCAI services.

For details, see the *Create a Contact Center AI Configuration* article.

**Step 2** Ensure that the Cloud Connect publisher and subscriber are installed.

For more information, see the *Install Cloud Connect* section in *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-guides-list.html.

**Step 3** Configure Cloud Connect in the CVP Operations Console (OAMP). For details see the section *Configure CVP Devices for Cloud Connect* in the *Administration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

**Step 4** Register Cloud Connect in the Unified CCE Administration console to establish a secure and trusted communication channel between the Cisco Contact Center on-premises deployment and cloud services.

For details, see the *Cloud Connect Integration* section in the *Administration Guide for Cisco Unified Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-maintenance-guides-list.html.

**Step 5** Import the Cloud Connect certificate to the CVP Server.

For details, see the section *Import Cloud Connect Certificate to Unified CVP Keystore* in the *Configuration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

**Step 6** In the Unified CCE Administration console, do the following with the CCAI configuration (created in step 1):

a) To view and sync the Contact Center AI configuration which is associated with all call types as a global configuration, see Associate Contact Center AI Configuration with All Call Types, on page 4.

b) To view, update, or delete the Contact Center AI configuration associated with a specific call type, see Associate Contact Center AI Configuration with a Call Type, on page 5.

**Step 7** Provision Cloud Connect on Cisco Finesse.

For more information, see the *Cloud Connect Server Settings* topic in the *Cisco Finesse Administration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html.

**Step 8** To add the Call Transcript gadget to the Cisco Finesse desktop layout:

a) Enable the Call Transcript gadget in Cisco Finesse Administration.

For details, see the *Manage Desktop Layout* section in the Cisco Finesse Administration Guide.

b) Enable the Call Transcription service in Unified CCE Administration for an agent or multiple agents together.

For details, see Enable or Disable Contact Center AI Services for Agents, on page 53.

Once enabled, the Call Transcript gadget appears on the Home tab. For details on how to use the gadget, see the *Contact Center AI Gadgets User Guide for Cisco Contact Center Enterprise*.

**Note** Gadget auto-hide/un-hide and notifications capability is available only if the gadget is configured as a multi-tab gadget in Cisco Finesse. For more details, see *Configure Multi-Tab Gadget Layout* section in the *Cisco Finesse Administration Guide*.

**Step 9** Perform the following steps to configure WebSocket-based forking in CUBE.

a) Create a SIP profile and associate it at the dial-peer level in CUBE. For details, see Create a SIP Profile at the Dial-Peer Level in CUBE, on page 8.

b) Import the WebSocket Connector certificate to CUBE. For details, see Import or Verify WebSocket Connector Certificate to CUBE, on page 8.

c) Configure WebSocket-based forking in CUBE. For details, see the *WebSocket-Based Media Forking for Cloud Speech Services* chapter in the *Cisco Unified Border Element Configuration Guide - Cisco IOS XE 17.6 Onwards*.

CUBE uses a WebSocket connection to fork the media streams of the agent and the caller towards the Webex CCAI Orchestrator service. For more details, see the Contact Center AI Services Considerations section in the *Solution Design Guide for Cisco Unified Contact Center Enterprise* .

# Enable or Disable Contact Center AI Services for Agents

Contact Center AI Services can be configured for each agent. Administrators and supervisors can enable or disable the services for an agent or multiple agents together.

## Enable or Disable Contact Center AI Services for an Agent

This procedure explains how to enable or disable Contact Center AI Services for an agent.

**Procedure**

**Step 1** In **Unified CCE Administration**, choose **Users** > **Agents**.

**Step 2** Click on the agent row whose services are to be modified.

**Step 3** Click the **Contact Center AI** tab.
Displays a list of services enabled or disabled for the agent.

**Step 4** To enable or disable the required Contact Center AI Services, check or uncheck the check boxes corresponding to the services.

**Step 5** Click **Save**.

## Enable or Disable Contact Center AI Services for Multiple Agents

Administrators and supervisors can enable or disable Contact Center AI Services for multiple agents.

All agents must belong to the same site and the same department, or all agents must be global agents. The **Edit** button is disabled if:

- Agents from different sites, departments, or peripheral sets are selected.

- A mix of global and departmental agents are selected.

**Procedure**

**Step 1** In **Unified CCE Administration**, choose **Users** > **Agents**.

**Step 2** Check the check box corresponding to each agent whose services you want to edit.

**Step 3** Click **Edit** > **Contact Center AI**.
The Edit Services dialog displays a list of services that are  the service that is enabled or disabled.

- If the service is enabled for all the agents selected for editing, the check box is checked.
- If the service is disabled for all the agents selected for editing, the check box is unchecked.
- If the service is enabled for some agents and disabled for the others, the check box has a dash (—).

**Step 4** To enable or disable the Contact Center AI Services, check or uncheck the check boxes corresponding to the services.

**Step 5** Click **Save**, and then click **Yes** to confirm the changes.

# Enable or Disable AnswersContact Center AI Services for Agents using Bulk Job

**Procedure**

**Step 1** Navigate to **Unified CCE Administration** > **Overview** > **Bulk Import**.

**Step 2** Click **Templates**.

The **Download Templates** popup window opens.

**Step 3** Click the **Download** icon for the Contact Center AI template you want to use.

**Step 4** Click **OK** to close the **Download Templates** popup window.

**Step 5** Open the `.csv` template in Microsoft Excel.

**Step 6** Populate the file as described in the Bulk Contact Center AI Services Content File, on page 7.

**Step 7** Save the populated file to the local machine.

**Step 8** Navigate to **Unified CCE Administration** > **Overview** > **Bulk Import**.

**Step 9** Click **New**.

**Step 10** In the optional **Description** field, enter up to 255 characters to describe the bulk job.

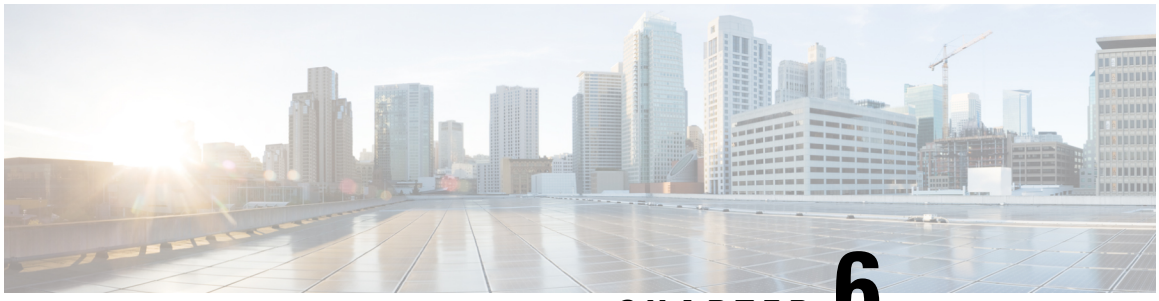
**Step 11** In the **Content file** field, choose the file to upload, and then click **Save**.

# Bulk Contact Center AI Services Content File

The content file for Contact Center AI bulk job contains the fields given in the following table. Enter the values appropriately in the given fields to enable or disable Contact Center AI Services for the agents.

**Note** Bulk job is available for administrators only when Cloud Connect is added in the inventory and registered on the Control Hub.

| Field | Required? | Description |
|---|---|---|
| agentId | Agent ID or Username | Existing agentId for which you want to enable or disable the Contact Center AI Services.<br>You must provide either an agentId or the userName. If both are provided, agentId takes precedence over the userName. If the agentId value is left blank, the userName will reference an existing agent. |
| userName | Username or Agent ID | Username of the agent for which you want to enable or disable the Contact Center AI Services.<br>If no agent is found with the given username, the Contact Center AI Services association fails. |
| agentServices | Yes (to enable Contact Center AI Services) | The type of Contact Center AI Services to be associated with the agent. Supported values are AgentAnswers and Transcript. To associate more than one services, seperate the values using semicolon (;).<br>If the value is updated, any existing enabled service gets overwritten. If the value is left empty, no service gets associated with the agent. |



CHAPTER 6

# Contact Sharing



## Contact Sharing Overview

The Contact Sharing feature uses a Contact Director to distribute incoming contacts to up to 3 Unified CCE instances. The 3 instances can support a total of 24,000 active agents.

Contact Sharing uses extrapolation to distribute calls and increase the overall agent and call handling capacity. Contact Sharing enables customers with multiple Unified Contact Center Enterprise (Unified CCE) systems to distribute calls across those systems. The Contact Director (sometimes called an IVR ICM) acts as an initial entry point for the call. If the call needs agent attention, Contact Sharing decides where to route the call based on Live Data real-time state information from the Unified CCE target systems. You can configure Contact Sharing to base routing decisions on factors such as the number of calls in queue, agent availability, average handle time, and custom calculations.

Use Unified CCE Administration to create and maintain the Contact Sharing groups and rules. A group is a collection of skill groups and precision queues across target systems. Each group has a rule that defines the logic for selecting the best skill group or precision queue in that group for a routing request. Each group has an `Accept Queue If` condition to include or exclude the individual skill groups and precision queues from the group for the routing decision. You can then route the call to the Unified CCE target system whose precision queue or skill group is the best match for the group's rule. The target system's routing scripts determine the final method for handling the request.

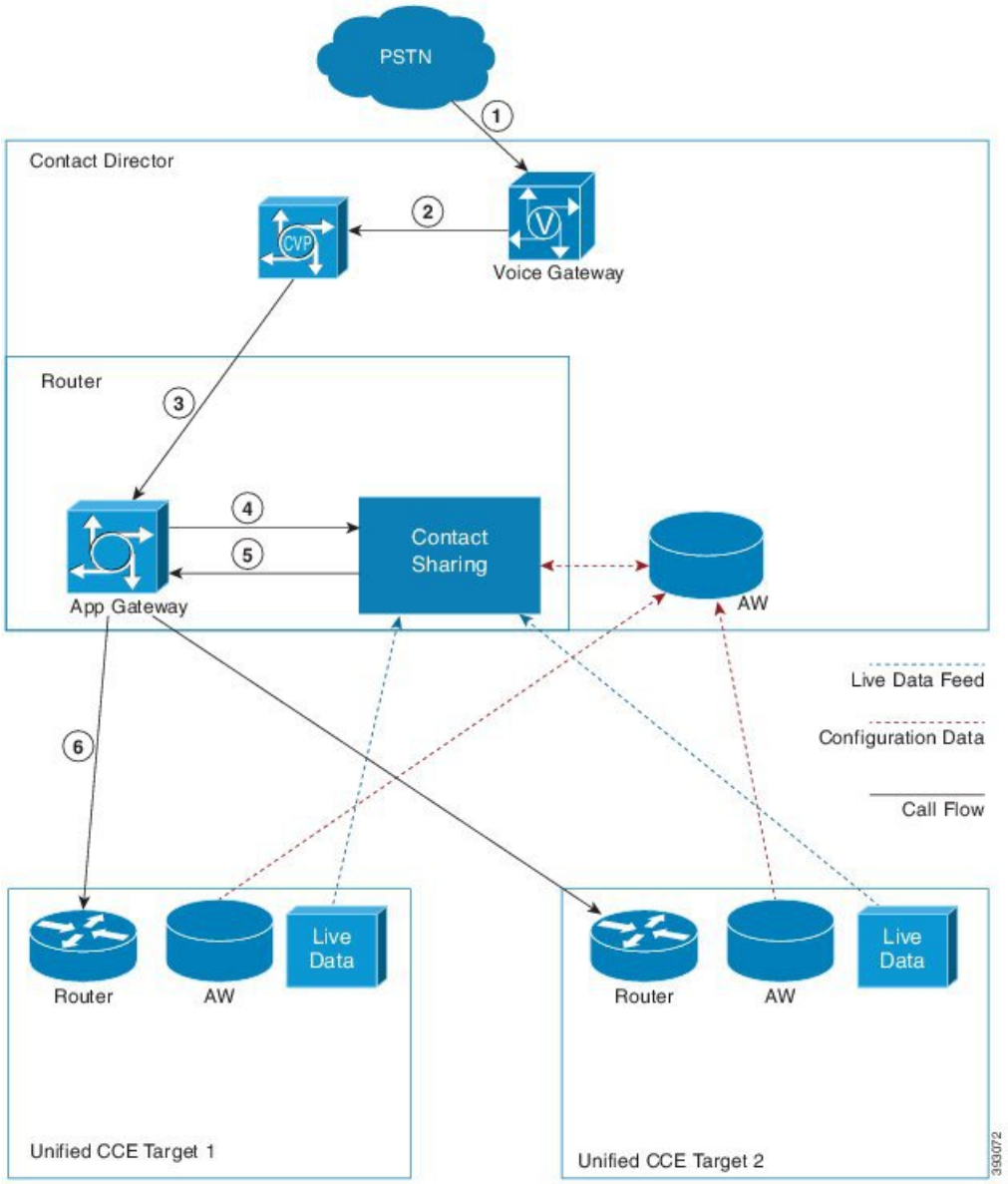
**Note** Contact Sharing gadgets are enabled only for the Contact Director deployment type.

For Contact Director configuration limits, see the chapter on configuration limits in the *Solution Design Guide for Cisco Unified Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-implementation-design-guides-list.html.

# Contact Sharing Call Flow

The basic Contact Sharing call flow runs as shown in this diagram:

***Figure 6: Contact Sharing Call Flow***




1. A call comes into the Voice Gateway on the Contact Director.
2. The Voice Gateway passes the call to CVP for VRU processing.
3. When the caller opts to speak to an agent, CVP passes the call data to the Router through the VRU PG.
4. The Router runs a script that assigns the call to a particular Contact Sharing Group. The Router sends the call data to the Application Gateway to pass to that Contact Sharing node.

5. The Contact Sharing node uses the Group Rule to determine which skill group or precision queue in its Queue should get the call. The node passes the selected target instance and its extrapolated guess of the best skill group or precision queue back to the Application Gateway.
6. The Application Gateway passes the information to the Router which routes the call to the selected target instance.

# Failover for Contact Sharing

Like all the main components in Unified CCE, Contact Sharing nodes run in redundant pairs. The redundant pair operates in hot-standby mode. Side B's data is kept in sync with Side A to ensure minimum failover time.

When the Side A process fails over, Side B takes over routing. Because the nodes operate in hot-standby mode, Side B does not reread Queues from the database. Side B requests a snapshot from Live Data. Until the snapshot arrives, Side B continues routing based on the last available Live Data modified by the current extrapolated data.

During failover, some route requests may receive an error. Error handling sends those requests to the default route. When Side A comes back online, it does not take over immediately. Side A remains in a ready state until Side B fails over.

The Contact Sharing process monitors the AW to see whether it has the latest configuration changes. If the AW configuration database does not have those changes or is not accessible, the Contact Sharing process switches to the alternate AW configuration data source.

When core components fail over on a target instance, reporting data can occasionally zero out. In that case, the Contact Sharing routing sends calls to the instance with reported resources. If Live Data does not zero out reporting data, then Contact Sharing continues to route on stale data until the snapshot information begins to arrive. If the active Contact Sharing side loses both Live Data connections, that side goes inactive and fails over to other side.

# Contact Director Installation and Setup

Contact Sharing runs on a Contact Director that you connect with up to three target Unified CCE deployments. You configure the Contact Director to monitor the Live Data feed from the targets. The task flow for installing a Contact Director is as follows:

| Task | See |
|---|---|
| Ensure that virtual machines are ready for installation. | *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide* |
| Install Unified Communications Manager. | *Installation Guide for Cisco Unified Communications Manager and IM and Presence Service* |
| Install Unified CCE components (Router, Logger, Administration & Data Servers, peripherals). | *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide* |
| Optionally, install Cisco Unified Intelligence Center. | *Installation and Upgrade Guide for Cisco Unified Intelligence Center* |

| Task | See |
|---|---|
| Install Unified CVP. | *Installation and Upgrade Guide for Cisco Unified Customer Voice Portal* |

# Install Unified CCE

This section expands the installation process outlined in the *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-guides-list.html.

After setting up the VMs for the Contact Director, you install the Unified CCE components. The install and configuration of a Contact Director varies slightly from a Unified CCE deployment. The following table lists the applicable parts of the Unified CCE install procedures with any necessary changes for the Contact Director:

| Task | Steps | Contact Director Notes |
|---|---|---|
| Install Unified CCE Component Software (running the ICM-CCEInstaller) | — | — |
| Set up Organizational Units | Add a Domain | — |
| | Add Organizational Units | — |
| | Add Users to Security Groups | — |
| Set Up Unified CCE Central Controller Components | Add Unified CCE Instance | — |
| | Create Logger Database | Select **NAM**. |
| | Create HDS Database | — |
| | Add Logger Component to Instance | Choose **Hosted** > **Network Application Manager (NAM)** for the **Logger Type**. |
| | Add Router Component to Instance | Set the following values for a Contact Director:<br>• Check **Enable Remote Network Routing**.<br>• Set the **NAM ID**.<br>• Check **Contact Sharing** when you create the Router. |
| | Add Administration & Data Server Component to Instance | For a Contact Director, choose **Hosted** > **Network Administration & Data Server for Network Application Manager (NAM)** for the **Deployment Type**. |

| Task | Steps | Contact Director Notes |
|---|---|---|
| Set up Peripheral Gateways | Configure Peripheral Gateways | — |
| | Add Peripherals to Peripheral Gateways | — |
| | Set Up Peripheral Gateways | — |
| After completing the standard installation, perform these Contact Director-specific setup procedures. | | |
| Application Gateway Access Between Systems | Create Unified ICM Application Gateway | — |
| | Create an INCRP on Each Target Instance | — |
| | Set Application Gateway Default Values | — |

**Related Topics**

Application Gateway Access Between Systems, on page 61
Create Unified ICM Application Gateway, on page 61
Create an INCRP on Each Target Instance, on page 63
Set Application Gateway Default Values, on page 64

## Application Gateway Access Between Systems

The Contact Director uses a type of Unified ICM Application Gateway to access a target instance. After adding the components to the target instance, set up a Unified ICM Application Gateway in the Configuration Manager on the Administration & Data Server.

After setting up the Unified ICM Application Gateway, you can reference it with a Unified ICM Remote ICM node in a routing script on the Contact Director.

### Create Unified ICM Application Gateway

This procedure creates a Unified ICM Application Gateway on the Contact Director. You also need an application gateway on each target Unified CCE system.

**Procedure**

**Step 1** Open the **Configuration Manager** on an Administration & Data Server that your Contact Director uses.

**Step 2** Select **Tools** > **List Tools** > **Application Gateway List**.
The **Application Gateway List** window appears.

**Step 3** Click **Retrieve**.

**Step 4** Click **Add**.
The **Attributes** tab appears.

**Step 5** Specify the following values on the **Attributes** tab:

| Field | Description |
|---|---|
| Name | A name for the Unified ICM Application Gateway |
| Type | Select **Remote ICM**. |
| Preferred Side | Indicates the preferred side of the Application Gateway to use when both are available. If only one side is available, that side is always used. This option applies only for Custom Gateways. For Remote ICM systems, a suffix on the connection address indicates the preference. |
| Encryption | Indicates whether requests to the Application Gateway are encrypted. Select **None**. |
| Fault Tolerance | Specify the fault-tolerance strategy that the Application Gateway uses. |
| Connection | Select **Duplex**. |
| Description | Any additional information about the Unified ICM Application Gateway. |

**Step 6** Save your changes to create the Unified ICM Application Gateway.

**Note** Copy down the Unified ICM Application Gateway ID value. You use the ID when you set up the INCRP NIC on the target instance.

**Step 7** Select either of the **Connection** tabs to set the connection information.

**Step 8** Click **Enter Address**.
The **Enter Contact Director Address** dialog appears.

**Step 9** Specify the following information:

| Field | Description |
|---|---|
| IP Address/Name | Enter the Public (high priority) IP address of the target instance. Alternatively, You can use the SAN with assistance from your Cisco certified partner or TAC. Use the *same address* that you specified for the INCRP NIC on the target instance. You can use the hostname in place of the address. |
| Instance Number | Enter the number of the customer ICM on the target instance (0 — 24). |
| Side | Indicate which side of the Contact Director prefers this connection:<br>• **Side A**—Contact Director Side A prefers to use this connection.<br>• **Side B**—Contact Director Side B prefers to use this connection.<br>• **None**—Neither side of the Contact Director prefers to use this connection.<br>• **Both Side A and B**—Both sides of the Contact Director prefer to use this connection.<br>**Note** Use this setting to avoid unnecessary WAN traffic. For example, if you collocate Contact Director Side A with target instance Side A, this correct choice avoids WAN traffic to the other side. |

**Note** The **Enter Contact Director Address** dialog displays different fields depending on the type of Application Gateway chosen.

**Step 10** Repeat this process for the other side of the redundant pair.

**Step 11** Save your work and exit the dialog.

## Create an INCRP on Each Target Instance

The Contact Director communicates with the target instance by an INCRP NIC. Perform this procedure on each target instance.

### Procedure

**Step 1** From the Configuration Manager menu on the target instance, select **Tools > Explorer Tools > NIC Explorer**. The **NIC Explorer** window appears.

**Step 2** In the **Select Filter Data** box, click **Retrieve**.

**Step 3** Select a NIC or click the **Add NIC** button to create a new NIC.

**Step 4** Specify the following values on the **Logical Interface Controller** tab:

| Field | Description |
|---|---|
| Name | An enterprise name that serves as the NIC name. |
| Client Type | Select **INCRP**. |

**Step 5** Click the **Add Physical Interface Controller** button.
The Physical Interface Controller dialog displays.

**Step 6** Specify an **Enterprise Name** and click **OK**.

**Step 7** In the NIC tree window, click the routing client for your newly created NIC.

**Step 8** Specify the following values on the **Routing Client** tab:

| Option | Description |
|---|---|
| Name | An enterprise name that serves as the Routing Client name. |
| Configuration Parameters | `/customerID <RCID>` where <RCID> is the Routing Client ID of the matching routing client on the Contact Director. |
| Network Routing Client | The same value as the **Name** field. |

**Step 9** Click **Save**.
The newly defined NIC is saved in the database. A **Physical Controller ID** is assigned, and the **To Be Inserted** icon is removed from the tree window.

### Set Application Gateway Default Values

If you see performance issues, the Cisco Technical Assistance Center might advise you to change some of the application gateway's default values. Use the following procedure to change these values.

**Procedure**

**Step 1** In the **Configuration Manager**, select **Enterprise** > **System Information** > **System Information**. The **System Information** dialog appears.

**Step 2** In the **Application Gateway** section, select **Remote ICM**.

**Step 3** Use the other tabs to set the default values for the Unified ICM Application Gateway connections. Take account of the Contact Director NIC settings for timeout, late, and so on as you set the Unified ICM Application Gateway timeout settings for a target Unified CCE system.

**Step 4** Click **OK** and close the dialog.

## Install Cisco Unified Intelligence Center (Optional)

To run the Contact Sharing reports at the Contact Director site, you can install Cisco Unified Intelligence Center there. For installation procedures, see the *Installation and Upgrade Guide for Cisco Unified Intelligence Center* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-intelligence-center/products-installation-guides-list.html.

## Install Unified CVP

The Contact Director uses Unified CVP for VRU processing of the incoming calls. For installation procedures, see the *Installation and Upgrade Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-guides-list.html.

**Note** The Unified CCE Reference Designs specify that you use Unified CVP. In a non-Reference Design deployment, you can alternately use Unified IP IVR or a third-party VRU.

# Set Up Contact Sharing

After installing the Contact Director and connecting the Contact Director to the target Unified CCE instances, you set up the contact sharing feature as follows:

| Task | See this Topic |
|---|---|
| Set up the contact sharing node on the Contact Director. | Set Up a Contact Sharing Node |
| Set up the machine inventory for the contact sharing node. | Set up Contact Sharing Machine Inventory |

| Task | See this Topic |
|---|---|
| Add contact sharing rules. | Add and Maintain Rules |
| Add contact sharing groups. | Add and Maintain Groups |

**Note** Your solution can only have one contact sharing node.

# Set Up a Contact Sharing Node

You set up a contact sharing node, as opposed to the target instances, with the same procedure for setting up an application gateway.

**Procedure**

**Step 1** In the Configuration Manager, select **Tools** > **List Tools** > **Application Gateway List**. The **Application Gateway List** window appears.

**Step 2** To enable **Add**, click **Retrieve**.

**Step 3** Click **Add**. The **Attributes** tab appears.

**Step 4** Fill out the **Attributes** tab as follows:

| Option | Description |
|---|---|
| **Name** | Name of your contact sharing node |
| **Type** | Contact Share |

**Note** The remaining fields are preset and cannot be modified.

**Step 5** On the **Connection Side A** tab, set the **Address** field to the router's IP address. The default port is 5070.

**Step 6** On the **Connection Side B** tab, set the **Address** field to the router's IP address. The default port is 5070.

**Step 7** Click **Save** to create the contact sharing node.

# Set up Contact Sharing Machine Inventory

You set up the machine inventory for contact sharing through the `icm/bin/csMachineInventory.csv` file. Changes to the machine inventory do not take effect until the router restarts.

**Procedure**

**Step 1** Open your machine inventory file for contact sharing, by default `csMachineInventory.csv`.

**Step 2** Follow the instructions for editing the `csMachineInventory.csv` file. The instructions are contained within the file.

**Step 3** Run the following command:

**csMachineInventory.bat** *[options] username password inputfile*
**options: /list, /help, /config**

/list - This option lists the contact sharing machine inventory.

/help - This option displays the usage of the tool.

/config - This option configures the contact sharing machine inventory based on the input file.

**username**

Username for the REST API request

**password**

Password for the REST API request.

**inputfile**

Machine inventory file, including the path, for contact sharing, by default `csMachineInventory.csv`

### What to do next

If the router has already started, restart the router after setting up the machine inventory.

# Add and Maintain Rules

**Note** Contact Sharing comes with a default rule that cannot be deleted or modified. The name of this rule is **DefaultRule**.

### Procedure

**Step 1** Navigate to **CCE Web Administration** > **Feature** > **Contact Share Rules**.

**Step 2** Click **New** to open the **New Rule** window, or click an existing rule to open the **Edit Rule** window.

**Step 3** Complete the following fields:

| Field | Required | Description |
|---|---|---|
| **Name** | Yes | Enter a name using up to 32 alphanumeric characters, periods (.), and underscores (_). The name must start with an alphanumeric character. |
| **Expression** | Yes | Enter a formula that the Contact Share server uses to select the skill group or precision queue from a Contact Sharing group for a routing request. |
| **Description** | No | Enter up to 255 characters to describe the rule. |

**Step 4** Click **Save** to return to the List window.

**Step 5** To delete a rule, do one of the following:

- To delete a single rule, hover over the row for that rule and click the **trash can** icon at the end of the row.
- To delete up to 50 rules, check the check box for each rule that you want to delete. To select all rules in a list, check the **Select All** check box in the list header. Click **Delete**.

Deleting a rule is permanent.

## Add a New Rule by Copying an Existing Rule

You can also create a new rule by copying an existing rule. The **Description** and **Expression** fields are copied to the new rule.

### Procedure

**Step 1** Navigate to **Unified CCE Administration** > **Feature** > **Contact Share Rules**.

**Step 2** Either:

- Click the rule you want to copy, and then click the **Copy** button in the **Edit Rule** window.
- Hover over the row for that rule, and click the **copy** icon that appears at the end of the row.

The **New Rule** window opens.

**Step 3** Enter a **Name** for the rule, using up to 32 alphanumeric characters, periods (.), and underscores (_). The name must start with an alphanumeric character.

**Step 4** Review **Description** and **Expression** fields that were copied from the original rule, and make any necessary changes.

**Step 5** Click **Save**.

# Add and Maintain Groups

### Before you begin

Ensure that the Live Data connection is active before you configure Groups.

### Procedure

**Step 1** Navigate to **CCE Web Administration** > **Feature** > **Contact Share Groups**.

**Step 2** Click **New** to open the **New Group** window, or click an existing group to open the **Edit Group** window.

**Step 3** Complete the following fields:

| Field | Required | Description |
|---|---|---|

| | | |
|---|---|---|
| **Name** | Yes | Enter a name using up to 32 alphanumeric characters, periods (.), and underscores (_). The name must start with an alphanumeric character. |
| **Description** | No | Enter up to 255 characters to describe the group. |
| **Rule** | Yes | Select a rule that defines the logic for selecting a skill group or precision queue in this group for a routing request:<br>**a.** Click the **magnifying glass** icon to display the **Select Rule** window.<br>**b.** Click the row to select a rule. |
| **Accept Queue If** | No | Enter a logical expression to determine if the individual skill groups and precision queues in the group can be included in the routing decision. |

**Step 4** Complete the Queues tab:

This tab shows the list of queues for this group.

a) Click **Add** to open **Add Queues**.
b) Click the queues you want to add to this group. The queues you chose appear on the **List of Queues**.
c) Close **Add Queues**.
d) Click **Save** on this tab to return to the List window.

**Note** The maximum number of queues is 100.

**Step 5** To delete a group, do one of the following:

- To delete a single group, hover over the row for that group and click the **trash can** icon at the end of the row.
- To select all groups in a list, check the **Select All** check box in the list header. Click **Delete**.

Deleting a group is permanent.

**Related Topics**

About Contact Sharing Expression Formula, on page 68

# Scripting for Contact Sharing

## Expression Formula for Contact Sharing

### About Contact Sharing Expression Formula

You can enter expressions for the following fields:

| Field | Description |
|---|---|
| **Accept Queue If** for a group | A logical expression to determine whether to include the individual skill groups or precision queues in the Contact Sharing group in the routing decision. The field is a freeform editor with a maximum length of 512 characters. Any result except zero evaluates as TRUE.<br>There is an implicit **Accept Queue If** of `Queue.*.LoggedOn > 0`. You cannot override this implicit check. |
| **Expression** for a rule | A formula that the Contact Share server uses to calculate the value to be considered against other queues in a Contact Sharing group. The expression always selects the queue with the minimum value. The field is a freeform editor with a maximum length of 512 characters. |

## Contact Sharing Expression Format

To evaluate all the skill groups and precision queues in a Contact Sharing group, use this syntax:

```
Queue.*.<FieldName>
```

Where *FieldName* is the name of the field that the expression evaluates, for example, `Ready`.

To evaluate a specific skill group or precision queue, use this syntax:

```
<ObjectType>.<InstanceName>/<TargetQueueName>.<FieldName>
```

- *ObjectType* must be SkillGroup or PrecisionQueue.
- *InstanceName* is the application gateway name.
- *TargetQueueName* is the enterprise name of the skill group or precision queue in the target system.
- *FieldName* is the name of the field that the expression evaluates.

## Contact Sharing Expression Examples

The following examples demonstrate some basic Contact Sharing expressions.

### Expression for a Group

A Contact Sharing group can take an Accept Queue If expression. The expression determines whether to include specific skill groups and precision queues in the group in the routing decision.

```
Queue.*.Avail > 5
```

This expression accepts all queues with more than five agents that are available.

### Expression for a Skill Group

```
SkillGroup.<InstanceName>/<TargetQueueName>.Avail > 5
```

This expression accepts the named skill group if it has more than five agents available.

### Expression for a Precision Queue

```
PrecisionQueue.<InstanceName>/<TargetQueueName>.Avail > 5
```

This expression accepts the named precision queue if it has more than five agents available.

### Expression for a Rule

A rule must take an expression. The expression selects a skill group or precision queue from a Contact Sharing group.

```
-1 * (Queue.*.Avail)
```

This expression selects the queue with the most available agents.

### Expression for MED Only

This expression calculates the Minimum Expected Delay (MED) to determine which target system receives the call for routing.

```
(Queue.*.QueuedNow+1)*(Queue.*.AvgHandledCallsTimeToInterval>0?
Queue.*.AvgHandledCallsTimeToInterval: 120) / (Queue.*.Ready>0?Queue.*.Ready:1)
```

### The Default Rule

Contact Sharing comes with a default rule. You cannot modify or delete the default rule. The default rule combines a MED calculation with an Agent Occupancy calculation to determine which target system receives the call for routing.

*If there are calls in queue,*

```
Queue.*.QueuedNow > 0?
```

*Then use the MED calculation:*

```
((Queue.*.QueuedNow+1)*(Queue.*.AvgHandledCallsTimeToInterval>0?
 Queue.*.AvgHandledCallsTimeToInterval: 120)/(Queue.*.Ready>0?Queue.*.Ready:1)):
```

*Otherwise, use the Agent Occupancy calculation:*

```
(((Queue.*.LoggedOnTimeToInterval - Queue.*.NotReadyTimeToInterval)==0
||(Queue.*.AvailTimeToInterval <= 10 * Queue.*.XAvail) )?
0: -1*(Queue.*.AvailTimeToInterval-10 * Queue.*.XAvail)/
(Queue.*.LoggedOnTimeToInterval - Queue.*.NotReadyTimeToInterval))
```

This expression chooses a queue on the target instance with the least occupied agents or the least queued calls.

**Note** The default rule is only an example. Customize the rule to match your needs or write your own rules.

## Contact Sharing Expression Reference

### Supported Operations

The following table lists the supported operations:

| Type of Operation | Operator | Description |
|---|---|---|
| Conditional | && | Conditional-AND |
| | \|\| | Conditional-OR |
| | ? : | Ternary (shorthand for if-then-else statement)<br>ex. A **?** B **:** C<br>If A, then B, otherwise C. |
| Relational | == | Equal to |
| | != | Not equal to |
| | > | Greater than |
| | >= | Greater than or equal to |
| | < | Less than |
| | <= | Less than or equal to |
| Bitwise and Bit Shift | ~ | Unary bitwise complement |
| | << | Signed left shift |
| | >> | Signed right shift |
| | & | Bitwise AND, for strings, also used for string concatenation |
| | ^ | Bitwise exclusive OR |
| | \| | Bitwise inclusive OR |
| Arithmetic | + | Addition |
| | - | Subtraction |
| | * | Multiplication |
| | / | Division |
| | % | Percentage |
| Prefix | + | Unary plus operator; indicates positive value |
| | - | Unary minus operator; negates an expression |
| | ! | Logical complement operator; inverts the value of a boolean |

| Type of Operation | Operator | Description |
|---|---|---|
| Wildcard | * | Wildcard support similar to the expression used in router. For example, in `SkillGroup.*.Ready`, the actual target replaces the asterisk when applying the expression. |

### Supported Objects and Fields

The following table lists the fields available from the Live Data feed or calculated by Contact Sharing for use in Contact Sharing expressions:

| Field Name | Description |
|---|---|
| ApplicationAvailable | The number of agents belonging to this Queue who are currently ApplicationAvailable for the MRD to which the Queue belongs. An agent is Application available if the agent is Not Routable and Available for the MRD. |
| Avail | The extrapolated number of agents in the READY state for this Queue. The extrapolation is as follows:<br>(The number of agents that Live Data reports in READY state) - XAvail<br>If the extrapolation results in a negative number, Contact Sharing sets this field to zero. |
| AvailTimeToInterval | Total seconds agents in the Queue have been in the READY state during the current interval. |
| AvgHandledCallsTimeToInterval | Average handle time in seconds for calls counted as handled by the Queue during the interval. |
| BusyOther | Number of agents currently in the BusyOther state for this Queue. |
| CallsHandledToInterval | Calls that by been answered and have completed wrap-up by the Queue during the interval. |
| Hold | The number of agents that have all active calls on hold. |
| ICMAvailable | The number of agents belonging to this Queue who are currently ICMAvailable for the MRD to which the Queue belongs. An agent is ICM available if the agent is Routable and Available for the MRD. |
| LoggedOn | Number of agents that are currently logged on to the Queue. |
| LoggedOnTimeToInterval | Total time, in seconds, agents were logged on to the Queue during the current interval. |
| NotReady | Number of agents in the Not Ready state for the Queue. |
| NotReadyTimeToInterval | Total seconds agents in the Queue have been in the Not Ready state during the interval. |

| Field Name | Description |
|---|---|
| QueuedNow | The extrapolated number of calls currently queued to this Queue. The extrapolation is as follows:<br>(The number of calls that Live Data reports queued to the Queue) + XQueuedNow |
| Ready | The number of agents who are Routable for the MRD associated with this Queue, and whose state for this Queue is not currently NOT_READY or WORK_NOT_READY. |
| ReseveredAgents | The number of agents for the Queue currently in the Reserved state. |
| TalkingAutoOut | The number of agents in the Queue currently talking on AutoOut (predictive) calls. |
| TalkingIn | The number of agents in the Queue currently talking on inbound calls. |
| TalkingOther | The number of agents in the Queue currently talking on internal calls, rather than inbound or outbound calls. Examples of other calls include agent-to-agent transfers and supervisor calls. |
| TalkingOut | The number of agents in the Queue currently talking on outbound calls. |
| TalkingPreview | The number of agents in the Queue currently talking on outbound Preview calls. |
| TalkingReserve | The number of agents in the Queue currently talking on agent reservation calls. |
| WorkNotReady | The number of agents in the Queue in the Work Not Ready state. |
| WorkReady | The number of agents in the Queue in the Work Ready state. |
| XAvail | The number of Contact Sharing requests assigned to the available agent count for this Queue during the extrapolation period. The extrapolation period defaults to 10 seconds.<br>Contact Sharing request increments this field when QueuedNow = 0 and Avail > 0. |
| XQueuedNow | The number of Contact Sharing requests assigned to the queued call count for this Queue during the extrapolation period. The extrapolation period defaults to 10 seconds.<br>Contact Sharing request increments this field when one of the following conditions apply:<br>• QueuedNow = 0 and Avail = 0<br>• QueuedNow > 0 and Avail = 0<br>• QueuedNow > 0 and Avail > 0 |

The following table lists the Call Variables that are available for use in Contact Sharing expressions:

| Call Variable Name | Description |
|---|---|
| CallerEnteredDigits | Digits caller entered in response to prompts. |
| CallingLineID | Billing phone number of the caller. (Commonly referred to as CLID). |
| CustomerProvidedDigits | Digits to be passed to the call recipient. |
| DialedNumberString | Phone number dialed by the caller. |
| PeripheralVariable1 through 10 | Value passed to and from the peripheral. |
| RouterCallDay | An encoded value that indicates the date on which the software processes the call. |
| RouterCallKey | A value that is unique among all calls the software has processed since midnight. RouterCallDay and RouterCallKey combine to form a unique call identifier. |
| RoutingClient | Name of the routing client making the route request. |

# Routing and Scripting for Contact Sharing

Contact Sharing uses two non-persistent call variables, `Call.ContactShareStatus` and `Call.ContactShareTarget`.

A successful route request returns `Call.ContactShareStatus` populated with the application gateway selected to receive the call. Use the call variable to route the call to the target Unified CCE instance.

`Call.ContactShareTarget` is populated only when the Gateway node takes a success path. The variable contains the target queue type and the target queue id. The target queue id is the Skill Group ID or Precision Queue ID on the target instance. The format is "Target Type, Target Queue ID". For example, "SG,5000 or PQ,5005". You can pass this data to the target instance in a call or ECC variable. Then, you have the Contact Sharing result available to use in scripting on the target instance.

# Error Handling for Contact Sharing

If a Contact Share route request fails, the router populates `Call.ContactShareStatus` with the following error codes. The call flow takes the failure branch out of the Gateway node. The error codes appear in the RCD table.

| Status Variable | Description |
|---|---|
| CS_NOT_CONNECTED (2) | No connection to the contact share process. |
| CS_TIMED_OUT (3) | Request to the contact share process failed. |
| CS_CONFIG_ERROR (4) | Contact share process encountered a configuration related error. |
| CS_EXECUTION_ERROR (5) | Contact share process encountered an expression execution related error. |

| Status Variable | Description |
|---|---|
| CS_APPGTW_ERROR (8) | Unable to lookup the application gateway with the code that the contact share process returned. |
| CS_UNKNOWN_ERROR (9) | Contact sharing encountered an unknown error. |

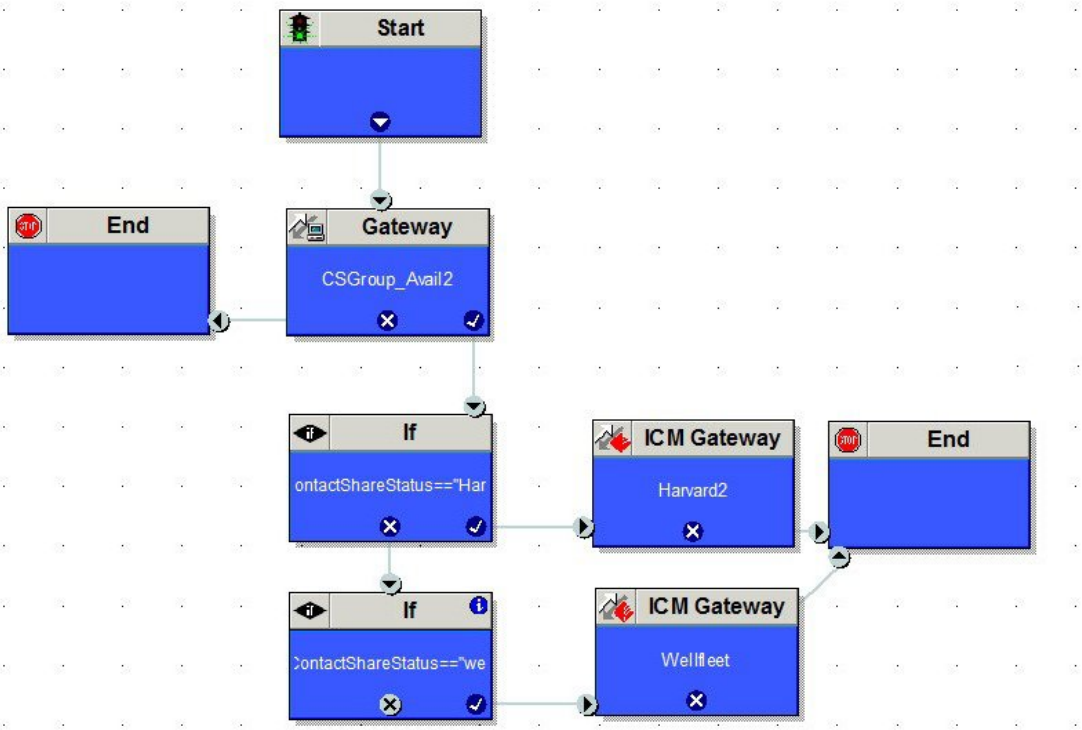
The following error messages can also appear:

| Status Variable | Description |
|---|---|
| ERROR_CONTACT_SHARE_GROUP_NOT_FOUND (1) | The Contact Share Group with the listed ID was not found. |
| ERROR_CONTACT_SHARE_RULE_NOT_FOUND (2) | The Contact Share Rule with the listed ID was not found. |
| ERROR_CONTACT_SHARE_RULE_EXPRESSION_INVALID (3) | The Contact Share Rule has an invalid rule expression. |
| ERROR_CONTACT_SHARE_GROUP_CONSIDERIF_EXPRESSION_INVALID (4) | The Contact Share Rule has an invalid AcceptQueueIf expression. |
| ERROR_CONTACT_SHARE_GROUP_NO_QUEUE_CONFIGURED (5) | There are no queues configured for the Contact Share Group. |
| ERROR_CONTACT_SHARE_ROUTING_EXCEPTION (6) | The route request failed for an unspecified reason. |
| ERROR_CONTACT_SHARE_ROUTING_NO_ELIGIBLE_TARGET (7) | No eligible queue was found for the route request. |
| ERROR_CONTACT_SHARE_ROUTING_TARGET_CONGESTED (8) | No eligible queue was found for the route request because of congestion control. |

# Other Scripting Considerations

A simple Contact Sharing script looks like the following:

*Figure 7: Sample Contact Sharing Script*




Consider the following points when you create Contact Sharing scripts:

- Always double check the logic in the routing to the Contact Sharing node.

**Tip** If you see all calls routing to one target system, check the IP Addresses in the machine inventory table and your script. The relationship between the Application Gateway ID and the IP Addresses might be wrong.

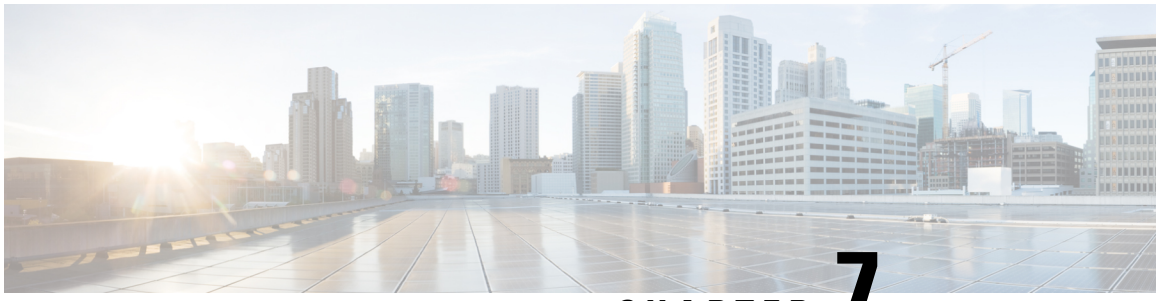
- Use Call Tracer to test your call flows.
- Never put two Contact Sharing nodes in the same path of your script.
- To search for a particular Contact Sharing node, use the string selection type to search for the Contact Sharing Group name.
- Contact Sharing returns the current Application Gateway name, not the ID. If you change the Application Gateway name for one of your target systems, change your scripts to match the new name.

### Script with Extrapolation in Mind

Contact Sharing's extrapolation assumes that the target systems route calls within the same Contact Sharing Group that the Contact Director used. If the target system's router does not follow this assumption, Contact Sharing's extrapolated data gets out of sync.

For Contact Sharing, have the target system route by one of the following methods:

- Route to the skill group or precision queue specified in `Call.ContactShareTarget`. You can pass the value from the Contact Director to the target system in a call or ECC variable.
- Route only among the same skill groups and precision queues that are part of the Contact Sharing Group that the Contact Director used.



CHAPTER 7

# Mobile Agent



# Mobile Agent

Deployments that need connectivity from the agent desktop over the internet without using a VPN is supported using the Reverse Proxy Automated Installer .

**Related Topics**

Reverse Proxy Automated Installer

## Capabilities

### Cisco Unified Mobile Agent Description

Unified Mobile Agent supports call center agents using phones that your contact center enterprise solution does not directly control. You can deploy a Mobile Agent as follows:

- Outside the contact center, by using an analog phone or a mobile phone in the home.
- On an IP phone connection that is not CTI-controlled by Unified CCE or by an associated Unified Communications Manager.
- On any voice endpoint of any ACD (including endpoints on other Unified Communication Managers) that the contact center Unified Communication Manager can reach by a SIP trunk.

A Mobile Agent can use different phone numbers at different times; the agent enters the phone number at login time. An agent can access the Mobile Agent functionality using any phone number that is included in the Unified Communications Manager dial plan.

With Cisco Unified Mobile Agent, contact centers can:

- Add or enable temporary staff during seasonal high call volume who can be brought on line with reduced startup costs
- Provide agents with the flexibility to work from home with similar quality, function, performance, convenience, and security as are available in the corporate headquarters contact center
- Allow agents to use the device they are most comfortable with, which improves agent productivity, helps to retain agents, and reduces training costs

- Hire skilled employees where they live and integrate remote workers into geographically dispersed teams with access to equivalent corporate applications

The sections that follow highlight some of the benefits of Unified Mobile Agent, and describe its features.

### Unified Mobile Agent Extends Unified CCE Capabilities

Before Mobile Agent, Unified CCE used a JTAPI interface to Unified CM to connect customer calls arriving on a voice gateway to an agent's IP phone. Mobile Agent enables the Unified CCE architecture to connect customer calls to an agent phone that Unified CCE does not directly control.

Mobile Agent uses a pair of CTI ports that function as proxies for the Mobile Agent phone and the caller phone. Every logged-in Mobile Agent requires two CTI ports (local and remote). The two CTI ports take the place of the Cisco IP Phone monitored and controlled by Unified CM JTAPI. The agent at login uses the local CTI port DN. When this agent is selected, the router transfers the caller to that CTI port. The remote CTI port calls the agent either at login for a nailed (permanent) connection or upon being selected for a call-by-call connection.

Cisco Unified Contact Center functionality remains intact whether an agent is mobile or local:

- Mobile Agents have the same capabilities and functionality that local agents have.
- Mobile Agents do not need any specialized equipment; they can receive calls on an analog or mobile phone.
- Unified Mobile Agent supports Cisco Finesse.
- Mobile Agent activity is recorded in the same contact center reports as local agent activity.
- Mobile Agent CTI and application data uses the same security mechanisms as local agent data.

### Unified Mobile Agent Provides Agent Sign-In Flexibility

Agents can be either local agents or Mobile Agents, depending on how they sign in at various times.

Regardless of whether agents sign in as local or Mobile Agents, their skill groups do not change. Because agents are chosen by existing selection rules and not by how they are connected, the same routing applies regardless of how the agents log in. If you want to control routing depending on whether agents are local or mobile, assign the agents to different skill groups and design your scripts accordingly.

### Connection Modes

Cisco Unified Mobile Agent allows system administrators to configure agents to use either call by call dialing or a nailed connection, or the administrator can configure agents to choose a connection mode at login time.

Mobile Agents are defined as agents using phones not directly controlled by Unified CC, irrespective of their physical location. (The term local agent refers to an agent who uses a phone that is under control of Unified CC, irrespective of physical location.)

You can configure Mobile Agents using either of two delivery modes:

- Call by Call—In this mode, the Mobile Agent's phone is dialed for each incoming call. When the call ends, the Mobile Agent's phone is disconnected before being made ready for the next call.
- Nailed Connection—In this mode, the agent is called at login time and the line stays connected through multiple customer calls.

**Note** The administrator can select the *Agent chooses* option, which allows an agent to select a call delivery mode at login.

### *Call by Call*

In a *call by call* delivery mode, the Mobile Agent's phone is dialed for each incoming call. When the call ends, the Mobile Agent's phone disconnects before is it made ready for the next call.

The *call by call* call flow works as follows:

1. At login, the agent specifies an assigned extension for a CTI port.
2. A customer call arrives in the system and, through Unified ICM configuration and scripting, is queued for a skill group or an agent. (This is no different than existing processing for local agents.)
3. The system assigns an agent to the call. If the agent's Desk Setting is Unified Mobile Agent-enabled and configured for either call by call or Agent chooses mode, the router uses the extension of the agent's CTI port as a label.
4. The incoming call rings at the agent's CTI port. The JTAPI Gateway and PIM notice this but do not answer the call.
5. A call to the agent is initiated on another CTI port chosen from a preconfigured pool. If this call fails, Redirect on No Answer processing is initiated.

**Note** In call by call mode, the Answer Wait Time is 3 to 15 seconds longer than in a local agent inbound call scenario. Specify a Redirect on No Answer setting large enough to accommodate the extra processing time.

6. When the agent takes the remote phone off-hook to answer the call, the system directs the customer call to the agent's call media address and the agent's call to the customer's call media address.
7. When the call ends, both connections are terminated and the agent is ready to accept another call.

**Note** To configure Mobile Agent in call by call delivery mode, you must set the wrap-up timer to at least one second using the Agent Desktop Settings List tool in the Configuration Manager.

In call by call delivery mode, callers often perceive a longer ring time compared to nailed connection delivery mode. This is because callers hear the ringtone during the call flow; ringing stops only after the agent answers. From the Unified CCE reporting perspective, a Mobile Agent in call by call delivery mode has a longer Answer Wait Time for the same reason.

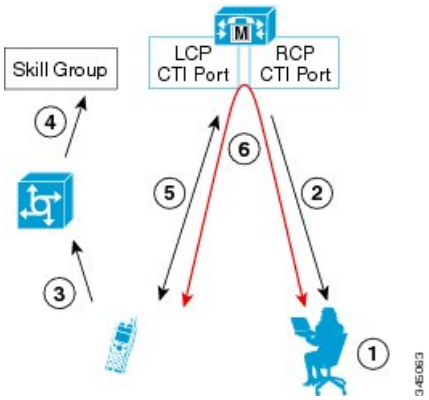
**Related Topics**

Configure Agent Desk Settings with Configuration Manager, on page 99

### *Nailed Connections*

In *nailed connection* delivery mode, the agent is called once, at login, and the phone line remains connected through multiple customer calls. See the following figure.

*Figure 8: Nailed Connection Call Flow*




The nailed connection call flow works as follows:

1. At login, the agent specifies an assigned extension for a CTI port from a pool.
2. A call to the agent is initiated on another CTI port chosen from a preconfigured pool. The agent answers the call. (The agent must answer this setup call to complete the connection and finalize the login procedure.)
3. A customer's call arrives in the system and, through Packaged CCE configuration and scripting, is queued for a skill group or an agent. (This is no different than existing processing for local agents.)
4. The system assigns an agent to the call. If the agent's Desk Setting is Unified Mobile Agent-enabled and configured for either nailed connection or Agent chooses mode, the router uses the extension of the agent's CTI port as a label.
5. The incoming call rings at the agent's CTI port. The JTAPI Gateway and PIM notice this but does not answer the call.
6. The agent desktop indicates a call is ringing and the agent clicks **Answer**.
7. When the agent indicates that they will answer the phone, the system directs the customer call to the agent's call media address and the agent call to the customer's call media address.
8. When the call ends, the customer connection is terminated and the agent state is set to Ready.

## Connect Tone

The *Connect Tone* feature in the nailed connection mode enables the system to play a tone to the Mobile Agent through the agent's headset to let the agent know when a new call is connected. In the nailed connection mode, you can configure an audible connect tone in addition to a call arrival notice (on the desktop only).

Connect Tone is particularly useful when Auto Answer is enabled or the agent is an Outbound agent. Here are its features:

- An audible tone (two beeps) is sent to the Mobile Agent headset when the call to the nailed connection Mobile Agent is connected. It is a DTMF tone played by Unified CM and cannot be modified.
- The Connect Tone plays only when the nailed connection Mobile Agent receives a call, as in the following examples:
  - The agent receives a consultation call.
  - The agent receives an outbound call.

- The Connect Tone does not play when the nailed connection Mobile Agent initiates a call, as in the following examples:
  - The agent makes a call.
  - The agent makes the consultation call.
  - Outbound direct preview call is made.
  - Supervisor barge-in call is made.

#### Related Topics

Enable Mobile Agent Connect Tone, on page 104

## Agent Greeting and Whisper Announcement

The Agent Greeting and Whisper Announcement features are available to Unified Mobile Agents. The following sections explain more about how these features apply to Unified Mobile Agents.

### *Agent Greeting*

You can use the Agent Greeting feature to record a message that plays automatically to callers when they connect to you. Your greeting message can welcome the caller, identify you, and include other useful information.

#### Limitations

The following limitations apply to the Agent Greeting feature for Mobile Agents.

- If a Mobile Agent ends the call when an Agent Greeting plays, the customer still hears the complete Agent Greeting before the call ends. This applies for both call by call and nailed-up calls.

**Note** In the Agent Greeting Call Type Report, this call does not appear as a failed agent greeting call.

- A supervisor cannot barge in when an Agent Greeting is playing.
- If a Peripheral Gateway (PG), JTAPI Gateway (JGW), or PIM failover occurs when an Agent Greeting plays for a Mobile Agent, the call fails.
- If a Mobile Agent ends the call when an Agent Greeting plays, the customer still hears the complete Agent Greeting before the call ends.

**Note** In the Agent Greeting Call Type Report, this call does not appear as a failed agent greeting call.

- If a Peripheral Gateway (PG), JTAPI Gateway (JGW), or PIM failover occurs when an Agent Greeting plays for a Mobile Agent, the call fails. This applies for both call-by-call and nailed-up calls.

**Note** You can use Agent Greeting for Mobile Agents only with parent/child deployments that are approved by Cisco Assessment-to-Quality (A2Q) with Design Mentoring Services (DMS).

For more information about Agent Greeting, see Capabilities, on page 11.

*Whisper Announcement*

With Whisper Announcement, agents can hear a brief prerecorded message just before they connect with each caller. The announcement plays only to the agent; the caller hears ringing (based on existing ringtone patterns) while the announcement plays. The announcement can contain information about the caller, such as language preference or customer status. This information helps the agent prepare for the call.

#### Configuration Requirement

For the Whisper Announcement feature for Unified Mobile Agents, you require a Media Termination Point (MTP) resource on an incoming SIP device.

## Feature Requirements

### Hardware and Software Requirements

Hardware and software requirements for the Unified Mobile Agent are identical to those of Unified CCE. For more information on feature requirements, consult these documents:

- *Solution Design Guide for Cisco Unified Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-implementation-design-guides-list.html
- *Virtualization for Unified Contact Center Enterprise* at https://www.cisco.com/c/dam/en/us/td/docs/voice_ip_comm/uc_system/virtualization/cisco-collaboration-virtualization.html
- *Contact Center Enterprise Compatibility Matrix* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-device-support-tables-list.html

### Phone Requirements

A Unified Mobile Agent can use an analog, digital, or IP phone to handle calls.

### Conference Requirements

To use Agent Greeting for Mobile Agents, you must configure external conference-bridge (hardware) resources. To estimate the number of required resources, you can use the following formula:

*Number of conference bridge resources = Mobile Agent call rate × Average greeting time (in seconds)*

For information about configuring external conference-bridge resources, see the `dspfarm profile 1 for conference` configuration section in the sample configuration gateway, listed in Media Termination Points Configuration, on page 100.

### CTI Port Requirements

You need two CTI ports (local and remote) for every logged-in Mobile Agent.

Unified Mobile Agent uses Unified CM CTI Port as a proxy for the agent's phone. When this proxy is set up, whenever a Mobile Agent is selected to handle a customer call, the following happens:

- The call is directed to the CTI port extension.
- Unified CCE, using the JTAPI Gateway, intercepts the call arriving on the CTI Port and directs Unified CM to connect the call to the Mobile Agent.

Unified Mobile Agent requires that maximum number of calls is set to 2 and busy trigger is set to 1.

For Unified Mobile Agent to work properly, you must configure two CTI ports:

- One port to serve as the agent's virtual extension.
- The other port to initiate calls to the agent.

You must assign these CTI ports to the Unified ICM application. The ports are recognized by Unified ICM when receiving the Unified CM configuration.

For these CTI ports in IPv6 enabled deployments, you have to set **IP Addressing Mode** to **IPv4 Only**. You do this by creating a **Common Device Configuration** and referencing it to these CTI ports.

## Supported Unified CCE Features

The following features are supported:

- Unified CCE supports temporary uninstallation while preserving Mobile Agent data.

  For more information about temporary uninstallation, see the *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide*.

- Mobile Agents can participate in outbound campaigns, but they must use a nailed-up connection for all outbound dialing modes.
- Unified Mobile Agent supports Redirect on No Answer (RONA). If the Mobile Agent fails to answer, the agent is made Not Ready, and the call is redirected to a RANA DN route point.
- Unified Mobile Agent supports G.711A-law, G.711u-law, and G.729 codecs.
- There is no direct interaction between Unified Mobile Agent and multichannel applications. Email and Chat are IP applications that continue to operate, assuming the Mobile Agent has a desktop with enough bandwidth on the broadband connection to support them.
- Unified Mobile Agent supports Cisco Unified Customer Voice Portal (Unified CVP) and Cisco Unified IP-IVR (Unified IP IVR).

### Related Topics

Silent Monitoring, on page 87

### Fault Tolerance Support

Fault tolerance for the Unified Mobile Agent follows the behavior of Unified CCE:

- The JTAPI Gateway, Unified CCE PIM, and CTI components record key events related to Unified Mobile Agent as part of the logging process.
- As with standard Unified CCE calls, if a Peripheral Gateway (PG) component such as the JTAPI Gateway fails, the phone call is not lost, but subsequent call control (transfer, conference, or hold) might not be

possible after a failover. The Mobile Agent is notified of a failure (on the desktop), but they must log in again after a Unified CM or Unified ICM failure occurs.

- Where CTI data is delivered for screen pops, CTI data is preserved.

Unified Mobile Agent can experience many of the same failure cases as Unified CCE:

- Side A/B failure
- VRU failure
- Unified CM failure
- CTI server failure

There are also some failure cases that are unique to Unified Mobile Agent:

- A situation where a Mobile Agent is using a cellular phone and the connection is dropped due to non-availability of a signal, is deemed as external failure. The agent must call back and log-in again.
- If a Mobile Agent's phone line disconnects while using nailed connection mode, the agent must log in again to receive new calls.

### Related Topics

Failover, on page 86

## Important Considerations

Before you proceed, consider the following Unified Mobile Agent limitations and considerations:

### Failover

- During a failover, if an agent in call by call mode answers an alerting call, the call can drop. This occurs because the media cannot be bridged when there is no active PG.
- During a prolonged Peripheral Gateway (PG) failover, if an agent takes call control action for a Unified Mobile Agent-to-Unified Mobile Agent call, the call can drop. This occurs because the activating PG may not have information for all agents and calls at that point.
- Unified Communications Manager failover causes a Mobile Agent call to be lost.
- If a call by call Mobile Agent initiates a call (including a supervisor call) and does not answer the remote leg of the call before PG failover, the call fails. The agent must disconnect the remote agent call leg and reinitiate the call.

### Performance

- Mobile Agent call processing uses more server resources and therefore reduces the maximum number of supported agents on both Unified CM and the Unified ICM Agent PG.

  For more information about sizing Mobile Agents, see the *Solution Design Guide for Cisco Unified Contact Center Enterprise*.
- Because Unified Mobile Agent adds processing steps to Unified CCE default functionality, Mobile Agents may experience some delay in screen popup windows.

- From a caller's perspective, the call by call delivery mode has a longer ring time compared with the nailed connection delivery mode. This is because Unified CCE does not start to dial the Mobile Agent's phone number until *after* the call information is routed to the agent desktop. In addition, the customer call media stream is not connected to the agent until after the agent answers the phone.

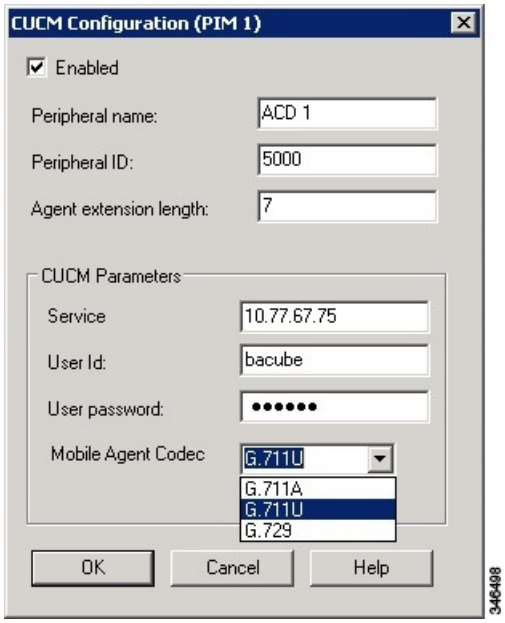
  The caller hears a repeated ringtone while Unified CCE makes these connections.

## Codec

The codec settings on the Peripheral Gateway and Voice Gateway must match. Perform the following procedure:

1. Launch the Peripheral Gateway Setup.
2. In the Peripheral Gateway Component Properties, select the UCM PIM and click **Edit.**
3. In the CallManager Parameters section, select the appropriate codec from the Mobile Agent Codec drop down list.

***Figure 9: Mobile Agent Codec Selection***




## Silent Monitoring

Unified Mobile Agent provides the following silent monitoring support:

- Unified Mobile Agent requires that caller and agent voice gateways be on separate devices if silent monitoring is to be used.
- Unified Mobile Agent does not support desktop monitoring.
- Whenever silent monitoring is used on Unified Mobile Agent, caller and agent voice gateways must be on separate devices. Similarly, if MTP is enabled when silent monitoring is used, MTP resources for caller and agent must also be on separate devices.

### Mobile Agent Scalability

Mobile Agent scalability may be contingent on specific Unified CM versions. For more information, see the *Solution Design Guide for Cisco Unified Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-implementation-design-guides-list.html.

### Unsupported Features

The following is a list of unsupported features for Mobile Agent:

- Web Callback
- Blended Collaboration
- Unified CM-based Silent Monitoring
- Agent Request

## Unified Mobile Agent Call Flows

This section provides sample Unified Mobile Agent call flows for:

- Inbound calls
- Local consultation calls
- Remote consultation calls
- Remote conference calls

In all Unified Mobile Agent call flows, the JTAPI Gateway maintains the signaling association between the inbound and outbound calls and, if necessary, performs further operations on the call. JTAPI Gateway, however, does not terminate media; it uses CTI to deliver the customer call from the inbound gateway port to the outbound gateway port.

This means that a Mobile Agent *must* use an agent desktop application to log in, change agent state, log out, send dual-tone multifrequency (DTMF) digits, and perform call control.

### About Figures in This Section

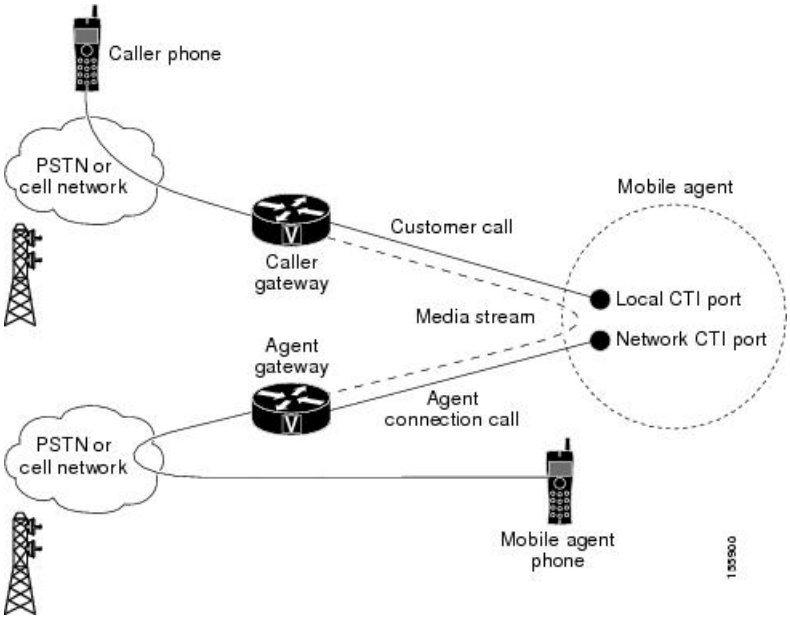
The figures in this section:

- Show a caller and a Mobile Agent in a cellular network. However, the same concepts apply whether the Mobile Agent is using an enterprise desk phone, an IP Phone spanning another Unified CM cluster, standard analog phone, or a third-party ACD phone.
- Focus solely on call media flow; a Mobile Agent must use a CTI Desktop with broadband access to perform agent state and call control.
- Show only a sampling of the call flows possible with Unified Mobile Agent.

### Inbound Call Flow

The following figure shows an inbound call flow.

*Figure 10: Mobile Agent Inbound Call Flow*




**Note** Caller and Agent voice gateways can coreside on one device, except in deployments where Silent Monitoring is required.

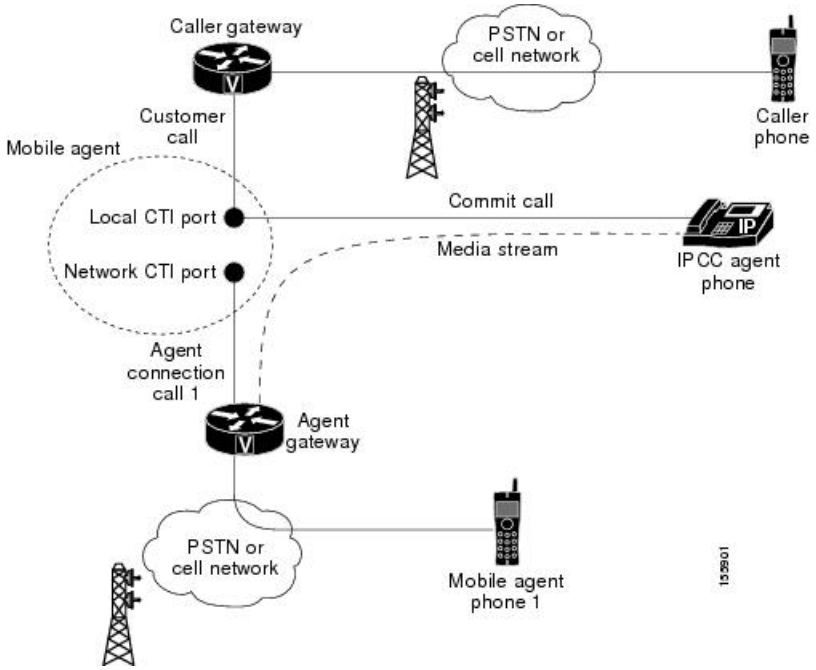
The following describes an inbound call flow:

1. The Mobile Agent becomes available to answer calls by:
   - Logging in to the corporate domain using VPN over the ADSL/Cable connection
   - Launching the agent desktop interface and logging in with their remote phone information
   - Entering the Ready mode
2. A customer call arrives at the Unified CC.
3. The JTAPI Gateway creates a Mobile Agent class to manage local and network CTI ports for a Mobile Agent.
4. The Router passes the call to the *local* CTI Port of a Mobile Agent.
5. The JTAPI Gateway places a call on a *network* CTI port to the agent's cell phone.
6. The JTAPI Gateway uses local and network CTI ports of the Mobile Agent to stream the media for the call from the inbound (caller) gateway port to the outbound (agent) gateway port.

## Local Consult Calls

The following figure shows a consult call flow between a Mobile Agent and a local agent.

*Figure 11: Mobile Agent Consult Call Flow*




**Note** Caller and Agent voice gateways can coreside on one device, except in deployments where Silent Monitoring is required.

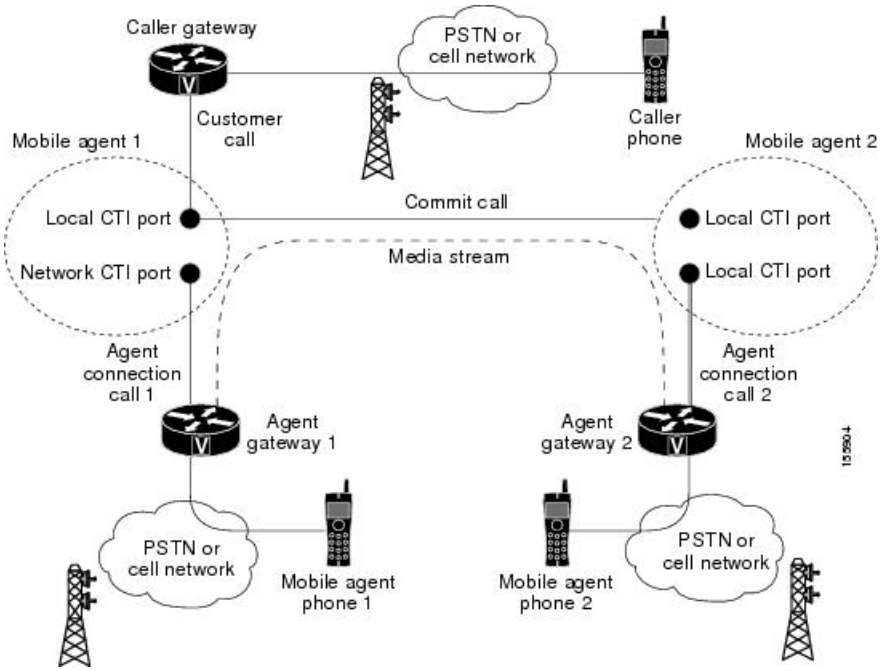
The following describes a local consult call flow:

1. The Mobile Agent becomes available to answer calls by:
   - Logging in to the corporate domain using VPN over the ADSL/Cable connection
   - Launching the agent desktop interface and logging in with their remote phone information
   - Entering the Ready mode
2. A customer call arrives at the Unified CC.
3. The JTAPI Gateway creates a Mobile Agent class to manage local and network CTI ports for a Mobile Agent.
4. The Router passes the call to the *local* CTI Port of a Mobile Agent.
5. The JTAPI Gateway places Agent Connection Call 1 on a *network* CTI port to the agent's cell phone.
6. The Mobile Agent places the customer call on hold and consults a local Unified CCE agent.
7. The JTAPI Gateway uses local and network CTI ports of the Mobile Agent to stream the media for the call from the IP hard phone to the outbound gateway port.

## Remote Consult Calls

The following figure shows a remote consult call flow between two Mobile Agents.

***Figure 12: Mobile Agent Remote Consult Call Flow***




**Note** Caller and Agent voice gateways can coreside on one device, except in deployments where Silent Monitoring is required.

The following describes a remote consult call flow:

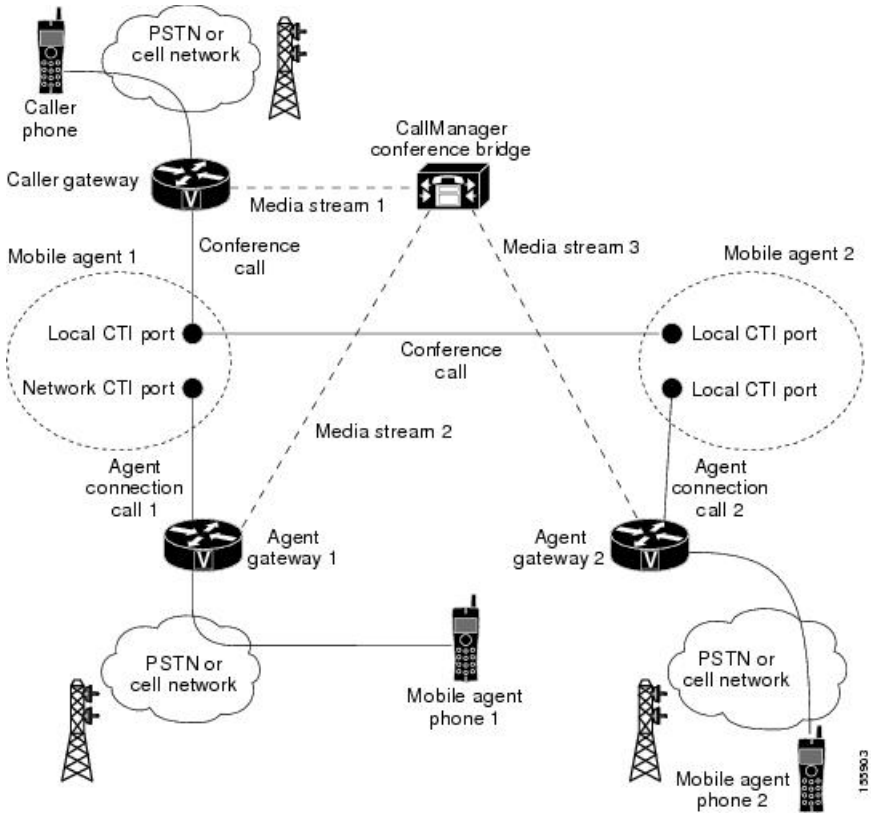
1. The Mobile Agent becomes available to answer calls by:
   - Logging in to the corporate domain using VPN over the ADSL/Cable connection
   - Launching the agent desktop interface and logging in with their remote phone information
   - Entering the Ready mode
2. A customer call arrives at the Unified CC.
3. The JTAPI Gateway creates a Mobile Agent class to manage local and network CTI ports for a Mobile Agent.
4. The Router passes the call to the *local* CTI Port of a Mobile Agent.
5. The JTAPI Gateway places Agent Connection Call 1 on a *network* CTI port to the agent's cell phone.
6. Mobile Agent 1 puts the customer call on hold and consults Mobile Agent 2.

7. The JTAPI Gateway uses the network CTI port of Mobile Agent 1 and the network CTI port of Mobile Agent 2 to stream the media for the call from the outbound gateway port on Agent Gateway 1 to the outbound gateway port on Agent Gateway 2.

## Remote Conference Calls

The following figure shows a remote conference call flow between two Mobile Agents.

***Figure 13: Mobile Agent Remote Conference Call Flow***




**Note** Caller and Agent voice gateways can coreside on one device, except in deployments where Silent Monitoring is required.

The following describes a remote conference call flow:

1. The Mobile Agent becomes available to answer calls by:
   - Logging in to the corporate domain using VPN over the ADSL/Cable connection
   - Launching the agent desktop interface and logging in with their remote phone information
   - Entering the Ready mode
2. A customer call arrives at the Unified CC.

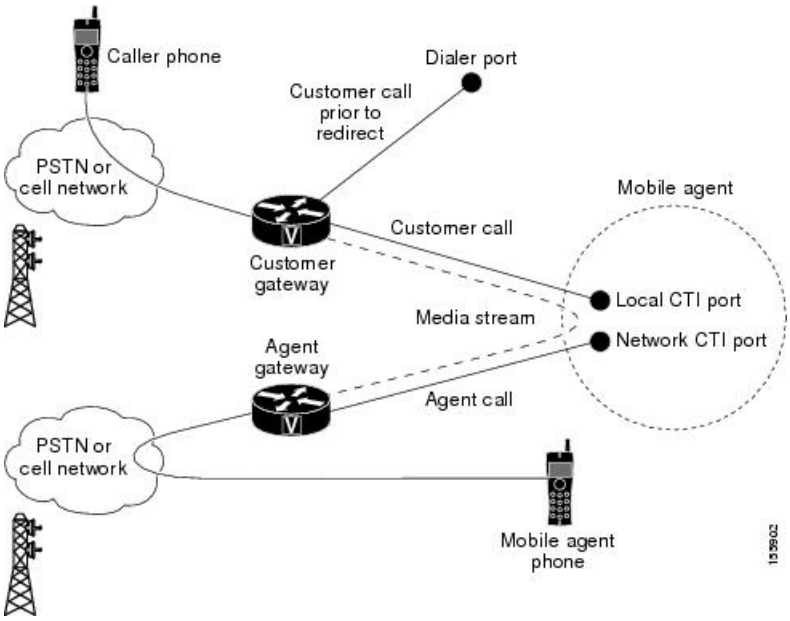
3. The JTAPI Gateway creates a Mobile Agent class to manage local and network CTI ports for a Mobile Agent.
4. The Router passes the call to the *local* CTI Port of a Mobile Agent.
5. Unified CM redirects the media stream 1 from inbound gateway on the Caller Gateway to the conference bridge during call merging process.
6. The JTAPI Gateway uses local and network CTI ports of Mobile Agent 1 to loop the Media Stream 2 for the call from the outbound gateway port on the Agent Gateway 1 to the conference bridge.
7. The JTAPI Gateway uses local and network CTI ports of Mobile Agent 2 to loop the Media Stream 3 for the call from the outbound gateway port on the Agent Gateway 2 to the conference bridge.

## Outbound Option Call Flow

The following figure shows a Outbound Option call flow between a customer and a Mobile Agent.

**Note** Unified Mobile Agent supports Outbound Option calls in nailed connection delivery mode *only*.

***Figure 14: Mobile Agent Outbound Call Flow***




**Note** Caller and Agent voice gateways can coreside on one device, except in deployments where Silent Monitoring is required.

The following describes an Outbound Option call flow:

1. The Mobile Agent becomes available to answer calls by:
   - Logging in to the corporate domain using VPN over the ADSL/Cable connection

- Launching the agent desktop interface and logging in with their remote phone information
- Entering the Ready mode

2. The JTAPI Gateway creates a Mobile Agent class to manage local and network CTI ports for a Mobile Agent.
3. Outbound Option dials the customer number and, after reaching a live customer, the Dialer redirects the customer call to the *local* CTI Port of an Outbound Option Mobile Agent.
4. The JTAPI Gateway places a call on a *network* CTI port to the agent's cell phone.
5. The JTAPI Gateway uses local and network CTI ports of the Mobile Agent to stream the media for the call from the inbound gateway port to the outbound gateway port.

## Unified Mobile Agent Reporting

Unified Mobile Agent-specific call data is contained in the following Cisco Unified Intelligence Center reports: Agent Team Historical, Agent Real Time, and Agent Skill Group Historical. These “All Field” reports contain information in multiple fields that show what kind of call the agent is on (nonmobile, call by call, nailed connection) and the Unified Mobile Agent phone number.

Notes about Mobile Agents and reporting:

- The Mobile Agent must be logged in through the agent desktop for call data to be recorded in Unified CC reports.
- Service level for Mobile Agent calls might be different than local agent calls, because it takes longer to connect the call to the agent.

  For example, a call by call Mobile Agent might have a longer Answer Wait Time Average than a local agent. This is because Unified CCE does not start to dial the Mobile Agent phone number until *after* the call information is routed to the agent desktop. In addition, the customer call media stream is not connected to the agent until after the agent answers the phone.

  For more information about Unified Mobile Agent fields in the database schema, see *Database Schema Handbook for Cisco Unified Contact Center Enterprise*.

# Initial Setup

## Summary of Unified Mobile Agent System Configuration Tasks

The following table describes system configuration tasks for Unified Mobile Agent.

***Table 5: Unified Mobile Agent System Configuration Tasks***

| Task | See |
|---|---|
| Configure Unified CM CTI Port pools | Unified CM CTI Port Configuration and Mapping for Unified Mobile Agent, on page 95 |
| Configure Unified CM Call Duration Timer | Maximum Call Duration Timer Configuration, on page 98 |

| Task | See |
|---|---|
| Configure Agent Desk Settings | Agent Desk Setting Configuration for Unified Mobile Agent, on page 99 |
| Configure Devices | Device Configuration for Unified Mobile Agent, on page 100 |
| Configure Media Termination Points | Media Termination Points Configuration, on page 100 |

## Unified CM CTI Port Configuration and Mapping for Unified Mobile Agent

This section describes the CTI Port Pool configuration tasks *specific* to Mobile Agent Option configuration. It does not discuss installation or configuration of Unified CCE.

**Note** For more information about installing and configuring Unified CM with Unified CCE, see the *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide*.

Unified Mobile Agent must have two CTI ports configured on Unified CM:

- A *local* CTI port, which Unified Mobile Agent uses as the agent's virtual extension.
- A *remote* CTI port, which Unified Mobile Agent uses to initiate a call to the Mobile Agent's phone.

### Naming Conventions for Local and Network Ports

- The local port *must* begin with the string LCP.
- The remote port *must* begin with the string RCP.
- The remaining characters in the device names for the LCP and RCP pair *must match*. For example an LCP port named LCP0000 has a corresponding RCP port named RCP0000.
- For example, you can use the following naming convention:
  - For a local CTI Port pool name, configure a name in the format LCP*xxxxFyyyy*, where LCP identifies a local CTI Port Pool, *xxxx* is the peripheral ID for the Unified CM PIM, and *yyyy* is the number of local CTI Port.

    Example: LCP5000F0000 represents CTI Port: 0 in a local CTI Port pool for the Unified CM PIM with the peripheral ID 5000.
  - For a network CTI Port pool name, use the same format, except substitute RCP as the first three characters.

**Note** While you do not require a naming convention, the substrings identifying the Unified CM PIM peripheral ID and the CTI Port *must* match for each local/network pair.

CTI Port configuration consists of the following steps:

1. Add the CTI port as you would for an IP Phone.
2. Use the naming convention described above to map the local and network CTI ports.

**Note** Each local CTI port must have a corresponding network CTI port.

3. Add a directory number for the local CTI port (that is, the agent's virtual extension).
4. Map the local and network CTI ports with the PG user.

## Music on Hold Design

If you want callers to hear music when a Mobile Agent places the caller on hold, you must assign Music on Hold (MoH) resources to the ingress voice gateway or trunk that is connected to the *caller* (as you do with traditional agents). In this case, the user or network audio source is specified on the local CTI port configuration. Similarly, if a Mobile Agent must hear music when the system puts the agent on hold, you must assign MoH resources to the ingress voice gateway or trunk that is connected to the *Mobile Agent.* In this case, the user or network audio source is specified on the remote CTI port configuration.

Do not assign MoH resources to local ports and remote CTI ports, because it might affect the system performance.

If a remote Mobile Agent calls over a nailed connection and if there is no active call to the agent, the agent is put on hold. Enable MoH to the Mobile Agent phone for nailed connection calls. If MoH resources are an issue, consider multicast MoH services.

If a remote Mobile Agent calls over a nailed connection, and if MoH is disabled, the hold tone plays to the agent phone during the hold time. This depends on the call processing agent that controls the Mobile Agent remote phone. For Unified CM, the hold tone is enabled by default (it is similar to the Mobile Agent connect tone). Because the hold tone is similar to the connect tone, it is difficult for the agent to identify if a call arrived from listening to the Mobile Agent connect tone. The hold tone prevents the agent from hearing the connect tone.

Therefore, disable the hold tone by changing the setting of the Tone on Hold Timer service parameter to 0. For more information about setting this parameter, see the Unified CM product documentation available at: https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/tsd-products-support-series-home.html.

## Configure Unified CM CTI Port Pools for Unified Mobile Agent

Perform the following steps to configure CTI Ports.

### Procedure

**Step 1** In Unified CM Administration, select **Device** > **Phone**.

**Step 2** Click **Add a New Phone**.

**Step 3** From Phone Type, select **CTI Port**.

**Step 4** Click **Next**.

**Step 5** In Device Name, enter a unique name for the local CTI Port pool name; click **OK** when finished.

Using the naming convention format LCP*xxxxyyyy*:

- LCP identifies the CTI Port as a local device.
- *xxxx* is the peripheral ID for the Unified CM PIM.
- *yyyy* is the local CTI Port.

The name LCP5000F0000 would represent CTI Port: 0 in a local CTI Port pool for the Unified CM PIM with the peripheral ID 5000.

The name LCP0000 represents the local port.

**Step 6** In Description, enter text that identifies the local CTI port.

**Step 7** Use the **Device Pool** drop-down list to choose the device pool to which you want to assign the network CTI port pool. Do not select Default. (The device pool defines sets of common characteristics for devices.)

**Step 8** Click **Save**.

**Step 9** Highlight a record and select **Add a New DN**.

**Step 10** Add a unique directory number for the CTI port you just created.

**Step 11** In Maximum Number of Calls, enter **2**.

**Step 12** In Busy Trigger, enter **1**.

**Step 13** When finished, click **Save**, and click **Close**.

**Step 14** Repeat the preceding steps to configure the network CTI port pool.

In Device Name, using the naming convention format RCP*xxxxyyyy*, where:

- RCP identifies the CTI port as the Remote CTI port where the call between the agent's remote device and the Unified CM Port is nailed up at agent login time.
- *xxxx* is the peripheral ID for the Unified CM PIM.
- *yyyy* is the network CTI port.

The name RCP5000F0000 represents CTI Port: 0 in a network CTI Port pool for the Unified CM PIM with the peripheral ID 5000.

**Step 15** In Description, enter text that identifies the network CTI port pool.

**Step 16** Use the **Device Pool** drop-down list to choose the device pool to which you want to assign the network CTI port. Do not select Default. pool. (The device pool defines sets of common characteristics for devices.)

**Step 17** Click **Save**.

**Step 18** Highlight a record and select **Add a New DN**.

**Step 19** Add a unique directory number for the CTI port you just created.

The extension length can be different from the extension length of the LCP Port if your dial plan requires it.

**Step 20** When finished, click **Save**, and click **Close**.

### Map Local and Remote CTI Ports with Peripheral Gateway User

After you define the CTI Port pool, you must associate the CTI Ports with PG users.

### Procedure

**Step 1** In Unified CM Administration, select **Application User**.

**Step 2** Select a username and associate ports with it.

**Step 3** When finished, click **Save**, and then click **Close**.

**Note** If CTI ports for Unified Mobile Agent are disassociated at the Unified CM while a Mobile Agent is on an active call, the call can drop.

## Maximum Call Duration Timer Configuration

By default, Mobile Agents in nailed connection mode log out after 12 hours. This happens because a Unified CM Service Parameter—the Maximum Call Duration Timer—determines the amount of time an agent phone can remain in the Connected state after login.

If you anticipate that nailed connection agents in your Unified Mobile Agent deployment will be logged on *longer than* 12 hours, use the following instructions to either:

- Increase the Maximum Call Duration Timer setting.
- Disable the timer entirely.

### Configure Maximum Call Duration Timer

**Note** This procedure applies only to Unified Mobile Agent deployments where agents logged in to nailed connection mode are to remain connected *longer than* 12 hours. Also, if your Mobile Agent deployment uses intercluster trunks, you must perform the following steps on both local and network Unified CM clusters.

### Procedure

**Step 1** In Unified CM Administration, choose **System** > **Service Parameters**.

**Step 2** In the Server drop-down list, choose a server.

**Step 3** In the Service drop-down list, choose a server .

The **Service Parameters Configuration** windowappears.

**Step 4** In the Cluster-wide Parameters section, specify a **Maximum Call Duration Timer** setting.

The default is 720 minutes (12 hours); the maximum setting allowed is 35791 minutes.

**Note** To disable the timer, enter **0**.

**Step 5** Click **Save**.

# Agent Desk Setting Configuration for Unified Mobile Agent

This section describes Agent Desk Settings that you must modify to accommodate Unified Mobile Agent features.

## Configure Agent Desk Settings with Configuration Manager

This section describes Agent Desk Settings configuration settings you should specify in Unified ICM Configuration Manager to accommodate Unified Mobile Agent features.

The following instructions describe how to configure *one* Agent Desk Setting. Repeat this process for each different Agent Desk Setting in your deployment.

### Procedure

**Step 1** From the Unified ICM Configuration Manager, choose **Configure ICM** > **Enterprise** > **Agent Desk Settings List**.

The Unified ICM Agent Desk Settings List dialog box opens.

**Step 2** Click **Retrieve**.

**Step 3** Click **Add**.

**Step 4** Fill in the following Attributes tab information, making sure to include settings for the following fields and check boxes:

- **Ring no answer time**. The system allows a call to ring at the agent's station before redirecting the call. This can be from 1 to 120 seconds.

  **Note** If you use call by call mode, the answer wait time will be longer than in a local agent inbound call scenario, so specify a value in this field to accommodate the extra processing time.

- **Logout non-activity time**. The number of seconds of agent inactivity while in the not ready state before the system logs out the agent. A blank entry disables the timer.
- **Cisco Unified Mobile Agent** (check box). Enables the Mobile Agent feature so that the agent can log in remotely and take calls from any phone.
- **Mobile Agent mode**. Select how call connections are made to the Mobile Agent's phone:
  - **Agent chooses**. Agent selects call by call or nailed connection at login.
  - **Call by call**. Agent's phone is dialed for each incoming call. When a call ends, the connection is terminated before the agent is made ready for next call.
  - **Nailed connection**. Agent is called once, at login. The line stays connected through multiple customer calls.

**Step 5** Click **Save**.

**Note** For more information about configuring Agent Desk Settings in Unified CCE, see *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide*.

# Device Configuration for Unified Mobile Agent

Use the Agent Targeting Rules (ATR) mechanism described in the *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide* to configure a device as you would for a phone, but using the LCP Port in place of the agent's phone extension.

# Media Termination Points Configuration

If you use SIP trunks, you must configure Media Termination Points (MTPs). You must also configure MTPs if you use TDM trunks to create an interface with service providers.

Additionally, MTPs are required for Mobile Agent call flows that involve a Cisco Unified Customer Voice Portal (CVP) solution. Because in DTMF signaling mode the Mobile Agent uses out-of-band signaling, whereas Unified CVP supports in-band signaling, the conversion from out-of-band to in-band signaling requires an MTP resource.

MTPs may be allocated as required in deployments that use a mix of IPv4 and IPv6 connections. MTP resources are allocated provided that the Media Resource Group List is configured on the IPV4 endpoint.

MTPs are available in the following forms, but not all are supported in Mobile Agent environments:

- Software-based MTPs in Cisco IOS gateways—use these MTPs for Mobile Agent as they provide codec flexibility and improved scalability compared with other MTP options. The following is a sample configuration on a gateway.

```
sccp local GigabitEthernet0/0
sccp ccm 10.10.10.31 identifier 1 priority 1 version 7.0
sccp ccm 10.10.10.131 identifier 2 priority 2 version 7.0
sccp
!
sccp ccm group 1
 associate ccm 1 priority 1
 associate ccm 2 priority 2
 associate profile 3 register gw84xcode
 associate profile 1 register gw84conf
 associate profile 2 register gw84mtp
!
dspfarm profile 3 transcode
 codec g729abr8
 codec g729ar8
 codec g711alaw
 codec g711ulaw
 codec g729r8
 codec g729br8
 maximum sessions 52
 associate application SCCP
!
dspfarm profile 1 conference
 codec g729br8
 codec g729r8
 codec g729abr8
 codec g729ar8
 codec g711alaw
 codec g711ulaw
 maximum sessions 24
 associate application SCCP
!
dspfarm profile 2 mtp
 codec g711ulaw
 maximum sessions software 500
 associate application SCCP
```

- Hardware-based MTPs in Cisco IOS gateways—These MTPs are supported. If you choose these, consider the extra cost, codec restrictions, and scalability constraints.
- Software-based MTPs using the Cisco IP Voice Media Streaming Application—These MTPs are not supported with Mobile Agents.

**Note** Because Unified CM-based software MTPs are used implicitly, you must add a special configuration to avoid using thcce-in10360-01-pcceucceipv6support-1101em. Create a new Media Resource Group (MRG) as a place holder, and place the software MTPs in that MRG. For instructions, refer to the Unified CM help documentation.

**Note** Ensure the `sccp ccm` configuration matches the Cisco Unified CM Group order used in the Device Pool assigned to the Media Termination Point in **CUCM** > **Media Resource**.

## Configure Media Termination Points in Unified CM

### *Add MTP Resources to Unified CM*

Perform these steps to add media termination points (MTPs) to Unified CM.

**Procedure**

**Step 1** In Unified CM Administration click **Media Resources** > **Media Termination Point**.

**Step 2** Click Add New.

**Step 3** Choose **Cisco IOS Enhanced Software Media Termination Point** from the **Media Termination Point Type** drop-down list.

**Step 4** Enter an MTP name. This name must match the device name you chose in IOS. In the example in the previous section, the MTP was called gw84mtp, as from the config line: **`associate profile 2 gw84mtp`**.

**Step 5** Choose the appropriate device pool.

**Step 6** Click **Save** and then click **Apply config**.

**Step 7** Navigate back to **Media Termination Point** and ensure the newly added MTP is listed as being registered with <*Unified CM subscriber IP address*> in the Status column.

**Step 8** Repeat steps 1 through 7 for each sccp ccm group you configured on each of your gateways.

### *Configure Media Termination Point Resources in Unified CM*

This section explains how to create media resource groups and media resource group lists.

**Procedure**

**Step 1** Navigate to **Media Resources** > **Media Resource Group** in Unified CM Administration.

**Step 2** Click **Add New**.

**Step 3** Specify a name and description.

**Step 4** From the Available Media Resources that you just created, move the those devices from the Available to the Selected list by clicking the down arrow. Ensure that you do *not* include Unified CM Software resources. For example, type anything that starts with ANN_, MTP_, or MOH_ .

**Step 5** Navigate to **Media Resources** > **Media Resource Group List**.

**Step 6** Click **Add New**.

**Step 7** Move the Media Resource Group you just created from the Available Media Resource Groups to the Selected Media Resource Groups.

**Step 8** Click **Save**.

### *Associate Media Resource Group List with Device Pools*

**Procedure**

**Step 1** Navigate to **System** > **Device Pool** and click on the device pool that contains the CTI ports for Mobile Agent. If there are multiple pools, perform the next step for each device pool that applies.

**Step 2** In the Media Resource Group List drop-down list, select the Media Resource Group List that you just created, click **Save** and then click **Apply config**.

### *Quarantine Unified CM Software-Based Resources*

Unified CM-based software MTPs are used by default. However, Cisco contact center deployments do not support these resources because they may cause performance problems in call processing. You must quarantine them with a special configuration. Perform the following steps:

**Procedure**

**Step 1** Create a new Media Resource Group (MRG) as a place holder.

**Step 2** Place the software MTPs in that MRG.

For further instructions, refer to the Unified CM help documentation.

### *Insert MTPs*

If you use SIP trunks, you must configure MTPs. This also applies if you use TDM trunks to interact with service providers. Mobile Agent cannot use an MTP with codec pass through. When you configure the MTP, you must select No pass through. KPML is not supported with Mobile Agent.

**Procedure**

**Step 1** Log in to Unified CM Administration and select **Device** > **Trunk**.

**Step 2** Select the trunk on which you want to configure MTPs.

**Step 3** Depending on the scenario listed below, perform the corresponding step listed in the Description column. Note that if you configure Trunk Groups to dynamically insert MTPs, only the calls that require MTPs use them.

- If you want to always insert MTPs for inbound and outbound calls through a given trunk: In the Trunk Configuration settings, select the **Media Termination Point Required** check box.
- If you want to dynamically insert MTPs when Unified ICM detects media or signaling incompatibility between the caller and called endpoints: In the Trunk Group Configuration settings, in DTMF Signaling Method, select **RFC2833**.

### *Enable Call Progress Tones for Agent-Initiated Calls*

**Procedure**

When **MTP Required** is not enabled, extra configuration is required to enable an agent to hear call progress tones for agent initiated calls. If instead you have dynamic MTP allocation by forcing mismatched DTMF settings, then configure the Unified Communications Manager to enable Early Offer.

For information on configuring the Unified Communications Manager, see the Unified Communications Manager product documentation at https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/tsd-products-support-series-home.html. The Cisco Annunciator does not generate ringback and other call progress tones, as it does for regular phones and softphones. Instead, Mobile Agent relies on the called party generating these tones (and the early offer setting triggers sending these tones to the agent).

**Note** This selection does not affect MTP sizing for IP Phones and other endpoints that support RFC2833 signaling, as is the case for many Cisco phones. For more information about supported phones, see the Contact Center Enterprise Compatibility Matrix at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-device-support-tables-list.html.

### *Verify MTP Resource Utilization*

Since Unified CM comes preconfigured with Software MTP resources, these resources may sometimes be used to provide MTP for Mobile Agent calls without proper configuration. Because we don't support the use of Unified CM based software MTPs, we explicitly quarantined them in the above section, Disabling Unified CM Based Software MTPs. To ensure that the new IOS-based MTPs are the ones being used for Mobile Agents, perform the following steps to verify that correct MTPs are used.

**Procedure**

**Step 1** Install the Unified CM Realtime monitoring tool. This tool can be downloaded under **Application** > **Plugins** within Unified CM Administration.

**Step 2** Place a call to a logged-in Mobile Agent.

**Step 3** Open the Unified CM Realtime monitoring tool and navigate to **System** > **Performance** > **Open Performance Monitoring**.

**Step 4** Expand the node(s) that are associated with your IOS-based MTP resources and choose **Cisco MTP Device**.

**Step 5** Double-click **Resources Active** and choose all of the available resources to monitor. This includes both IOS and Unified CM-based resources. Ensure that the only resources that are active during the Mobile Agent phone call are the IOS-based resources. Also, ensure that all UCM-based MTP resources are *not* active.

**Step 6** Repeat the previous step for each node that has MTP resources associated with it.

## Enabled Connect Tone Feature

In a nailed connection, the system can play a tone to the Unified Mobile Agent through the agent headset to let the agent know when a new call is connected. In the default Installation, the Mobile Agent Connect Tone feature is disabled.

## Enable Mobile Agent Connect Tone

If you require Unified Mobile Agent Connect Tone, you must make the following change in the Windows Registry for the key PlayMAConnectTone under the JTAPI GW PG registry entries.

Perform the following procedure to allow a Mobile Agent in the nailed connection mode to hear a tone when a new call is connected.

### Before you begin

MTP resources must be associated with the CUCM trunk that connects to the Agent Gateway.

### Procedure

**Step 1** On the PG machine, open the Registry Editor (regedit.exe).

**Step 2** Navigate to `HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\ICM\<InstanceName>\PG1A\PG\CurrentVersion\JGWS\jgw1\JGWData\Config\PlayMAConnectTone`.

The Edit DWORD Value dialog box appears.

**Step 3** In the Value data: field, enter **1** to enable Mobile Agent Connect Tone and click **OK**.

**Step 4** Exit the Registry Editor to save the change, and cycle the PG service.

# Administration and Usage

## Cisco Finesse

Finesse provides a browser-based desktop for agents and supervisors. Mobile agents can perform the same call control functions as Unified CCE agents. Mobile supervisors can perform all call control functions except for silent monitoring.

## Sign in to Cisco Finesse Desktop

### Procedure

**Step 1** Enter the hostname of the Finesse server in the fully qualified domain name (FQDN) format: https://<FQDN of Finesse server>, where FQDN is the fully qualified domain name of the Finesse server.

In an IPv6-enabled environment, you must include the port number in the URL (https://*FQDN of Finesse server*:8082/desktop).

**Step 2** In the ID field, enter your agent ID.

**Step 3** In the Password field, enter your password.

**Step 4** In the Extension field, enter your extension.

For a mobile agent, the extension represents the virtual extension for the agent, also known as the local CTI port (LCP).

**Step 5** Check the **Sign in as a Mobile Agent** check box.

The Mode and Dial Number fields appear.

**Step 6** From the Mode drop-down list, choose the mode you want to use.

In **Call by Call** mode, your phone is dialed for each incoming call and disconnected when the call ends.

In **Nailed Connection** mode, your phone is called when you sign in and the line stays connected through multiple customer calls.

**Step 7** In the Dial Number field, enter the number for the phone you are using.


| Option | Description |
|---|---|
| ID | The agent ID. |
| Password | Your supervisor assigns this password. |
| Extension | The agent's extension. |
| Sign in as Unified Mobile Agent | Select to sign in as a Unified Mobile Agent. |

| Option | Description |
|---|---|
| Mode | Call by Call or Nailed Connection |
| Dial Number | The number of the phone being used. |

**Step 8** Click **Sign In**.

**Note** In Nailed Connection mode, the desktop must receive and answer a setup call before sign-in is complete.

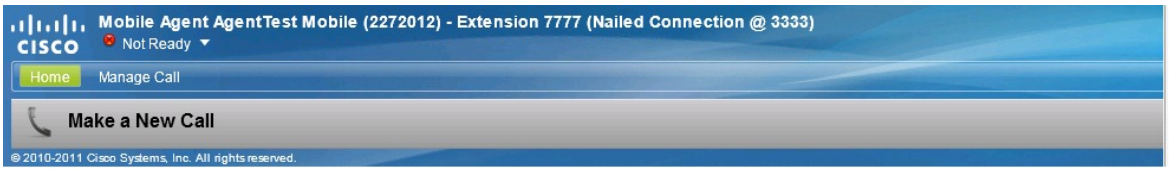
In Call by Call mode, the dial number provided is not verified. To ensure that the number is correct, verify the number in the header on the Agent Desktop after sign-in is complete.

## Verify Sign-In to Cisco Finesse

### Procedure

Check to be sure the Finesse Agent Desktop displays the following in the header:

- *Mobile Agent* before your agent name
- The mode used (Call by Call or Nailed Connection)
- The dial number you provided




## Enable Ready State

You must be in Ready state to process incoming calls.

### Procedure

Choose **Ready** from the drop-down list below the agent name.

**Note** If you are in call-by-call mode, you must answer and end each incoming call on your physical phone. After you answer a call, you must perform all other call control functions (such as Conference, Transfer, Hold, Retrieve) using the desktop.

With call-by-call connection, an agent cannot end one leg of a transfer without terminating it at the other end. The transfer must either be fully completed or both legs completely dropped.

If you are in Nailed Connection mode, after you answer the initial setup call, you must perform all other call control functions using the desktop.

### Make a Call

**Procedure**

**Step 1** From the drop-down list below the agent name, choose **Not Ready**.

**Note** You must be in Not Ready state to make a call.

**Step 2** Click **Make a New Call**.

**Step 3** Enter the number you want to call on the keypad, and then click **Call**.

If you are in Call by Call mode, the CTI server sends a setup call to your phone. A message appears on the keypad that states the following:

```
A call will be initiated to your phone which must be answered before an
outbound call to your destination can be made.
```

After the setup call is answered, the system establishes the outbound call to the destination specified.

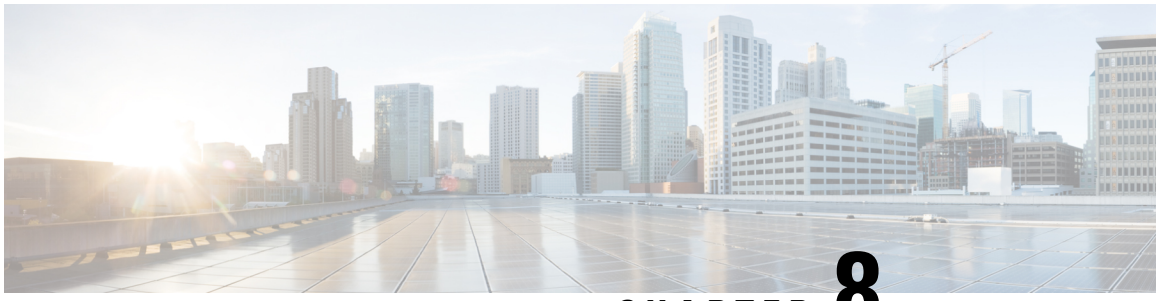
## Serviceability

On a Mobile Agent call flow, CUCM may return a 404 error due to the absence of a agent greeting, leading to call failure. To fix this issue, do the following:

1. Create a new Run External Script node. Map the backup media of the script to the agent greeting recording (media file).
2. Add the Run External Script node between the failure path of the AgentGreeting Run External Script node and the End node.
3. Connect the Run External Script node's success path to the existing Release Call node and failure path to the existing End node.

**Note** This fix may add a short delay of one to two seconds to the call flow.

For information about Agent Greeting Play Script, on page 30.



CHAPTER 8

# Post Call Survey



# Post Call Survey

A Post Call Survey takes place after the call. Typically, you use the survey to determine whether a customer was satisfied with the call experience. You configure a call flow that sends the call to a DNIS for the Post Call Survey after the agent disconnects from the caller.

Your VRU asks callers whether they want to participate in a Post Call Survey. If they choose to do so, they are automatically transferred to the Post Call Survey after the call flow completes.

## Post Call Survey Use Case

The caller is typically asked if they want to participate in a survey after the call. Your solution can determine based on dialed numbers to invoke the post call survey at the end of a call. When the customer completes the conversation with an agent, the customer is automatically redirected to a survey. When the agent ends the call, it initiates the Post Call Survey.

A customer can use the keypad on a touch tone phone and voice with ASR/TTS to respond to questions asked during the survey. For the solution, the post call survey call is just like another regular call. The Post Call Survey retrieves the call context information from the original customer call.

## Post Call Survey Design Impacts

Observe the following conditions when designing a Post Call Survey:

- A Post Call Survey initiates when the last agent ends the call. The call routing script launches a survey script.
- The mapping of a dialed number pattern to a Post Call Survey number enables the Post Call Survey feature for the call.
- The value of the expanded call variable **user.microapp.isPostCallSurvey** controls whether the call transfers to the Post Call Survey number.

- If **user.microapp.isPostCallSurvey** is set to **y** (the implied default), the call transfers to the mapped post call survey number.
- If **user.microapp.isPostCallSurvey** is set to **n**, the call ends.
- To route all calls in the dialed number pattern to the survey, your script does not have to set the **user.microapp.isPostCallSurvey** variable. The variable is set to **y** by default.

- You cannot have a REFER call flow with Post Call Survey. REFER call flows remove Unified CVP from the call. But, Post Call Survey needs Unified CVP because the agent has already disconnected.
- For Unified CCE reporting purposes, the Post Call Survey call inherits the call context for the initial call. When a survey starts, the call context of the customer call that was transferred to the agent replicates into the call context of the Post Call Survey call.
- The expanded call variable **isPostCallSurvey** will be cached only when the UCCE router generates a label for CVP.

# Configure Post Call Survey in CVP

Complete the following procedure to configure Post Call Survey in Unified CVP.

**Procedure**

**Step 1** Log in to the Unified CVP Operations Console and choose **System** > **Dialed Number Pattern**.

**Step 2** Enter the following configuration settings to associate incoming dialed numbers with survey numbers:

- **Dialed Number Pattern** - Enter the appropriate dialed number.

  The incoming Dialed Number for calls being directed to a Post Call Survey Dialed. This is the Dialed Number you want to redirect to the survey.

- **Enable Post Call Survey for Incoming Calls** - Select to enable post call survey for incoming calls.
- **Survey Dialed Number Pattern** - Enter the dialed number of the Post Call Survey. This is the dialed number to which the calls should be transferred to after the normal call flow completes.
- Click **Save** to save the Dialed Number Pattern.

**Step 3** Click **Deploy** to deploy the configuration to all Unified CVP Call Server devices.

# Configure Unified CCE

## Configure ECC Variable

You need not configure Unified CCE to use Post Call Survey, however, you can turn the feature off (and then on again) within an ICM script by using the ECC variable **user.microapp.isPostCallSurvey** and a value of n or y (value is case insensitive) to disable and re-enable the feature.

Configure the ECC variable to a value of n or y before the label node or before the Queue to Skillgroup node. This sends the correct value to Unified CVP before the agent transfer. This ECC variable is not needed to initiate a Post Call Survey call, but you can use it to control the feature when the Post Call Survey is configured using the Operations Console.

When the DN is mapped in the Operations Console for Post Call Survey, the call automatically transfers to the configured Post Call Survey DN.

Complete the following procedure to enable or disable the Post Call Survey:

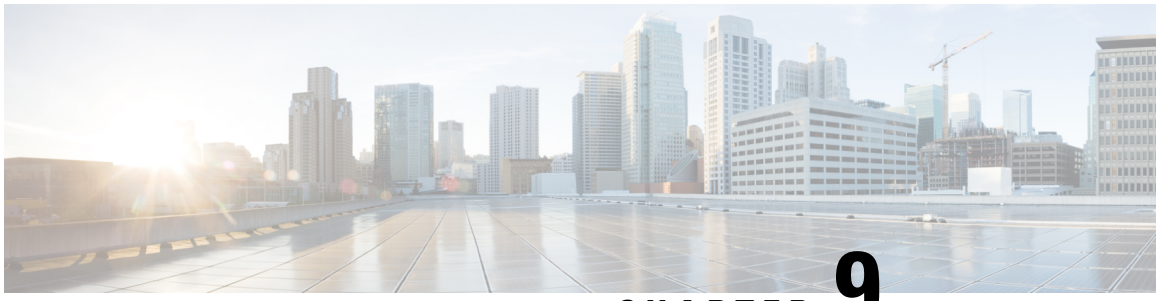
### Procedure

**Step 1** On the Unified CCE Administration Workstation, using configuration manager, select the **Expanded Call Variable List tool**.

**Step 2** Create a new ECC Variable with **Name:user.microapp.isPostCallSurvey**.

**Step 3** Set **Maximum Length** to 1.

**Step 4** Select the **Enabled** check box then click **Save**.



CHAPTER 9

# Precision Queue



# Capabilities

## Precision Queues

Precision routing offers a multidimensional alternative to skill group routing: using Unified CCE scripting, you can dynamically map the precision queues to direct a call to the agent who best matches the caller's precise needs. Precision queues are the key components of precision routing.

To configure Precision Routing, you must do the following:

1. Create attributes. Attributes are characteristics that can be assigned a True | False value or a Proficiency rating from 1 to 10.
2. Assign attributes to agents.
3. Create precision queues.
4. Create routing scripts.

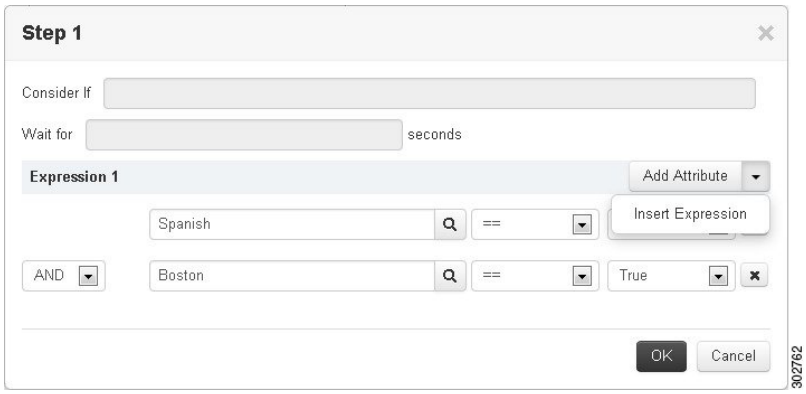
There is no need to add an agent to a precision queue; agents become members of precision queues automatically based on their attributes. If a precision queue requires an agent who lives in Boston, who speaks fluent Spanish, and who is proficient in troubleshooting a specific piece of equipment, an agent with the attributes *Boston = True*, *Spanish = True*, and *Repair = 10* is automatically part of the precision queue. A Spanish caller in Boston who needs help with equipment is routed to that agent.

A precision queue includes:

- **Terms:** A term compares an attribute against a value. For example, you can create the following term: Spanish == 10. The term of the attribute is the highest proficiency in Spanish.

  Each precision queue can have multiple attributes, and these attributes can be used in multiple terms. For example, to select an agent with a Spanish proficiency value between 5 and 10, you would create one term for Spanish > 5 and another for Spanish < 10.

- **Expressions:** An expression is a collection of one or more terms. The terms in an expression must share the same operator—they must all be AND or must all be OR relationships.




- **Steps:** A precision queue step is a time-based routing point within the precision queue. A step is a collection of one or more expressions.

  A step may also include wait time and a Consider If formula. Use wait time to assign a maximum amount of time to wait for an available agent. Use a Consider If formula to evaluate the step against predefined criteria, for example, another queue.


Administrators can see and manage attributes. Supervisors can configure attributes for their supervised agents on the Attributes tab of the Agents tool.

# Skill Groups or Precision Queues?

Should you use skill groups or precision queues for the routing needs of your organization? This section distinguishes the two methods.

### Use a Skill Group

A skill group represents a competency or responsibility. For example, it could be a predefined collection of traits, such as salespeople who are in charge of selling to England. The skill group could be called “English sales”. If you wanted to divide the agents in this group into two types of proficiencies (perhaps based on experience), you would need to set up two separate skill groups; for example, English Sales 1 and English Sales 2. You would then associate an agent with one of them, based on the agent's proficiency. Do this by accessing the skill group and locating the agent that you want to add to it (or add that skill group to the agent). To summarize, creating a skill group involves first building a concept of what combinations of traits you want for each agent, like English Sales 2.

### Use a Precision Queue

In contrast to skill groups, a precision queue breaks down attribute definitions to form a collection of agents at an *attribute* level. The agents that match the attribute level of the precision queue become associated with that precision queue.

With precision queues, the preceding English sales example involves defining the attributes English and Sales, and associating agents that have those traits to them. The precision queue English Sales would dynamically

map all those agents that had those traits to the precision queue. In addition, you can define more complex proficiency attributes to associate with those agents. This would allow you to build, in a single precision queue, multiple proficiency searches like English language proficiency 10 and sales proficiency 5.

To break down the precision queue example into skill groups, you would need to set up two separate skill groups: English language proficiency 10 and sales proficiency 5. With precision queues, you can refine agents by attributes. With skill groups, you define a skill group and then assign agents to it.

### Decide on Skill Groups or a Precision Queue

Precision routing enhances and can replace traditional routing. Traditional routing looks at all of the skill groups to which an agent belongs and defines the hierarchy of skills to map business needs. However, traditional routing is restricted by its single-dimensional nature.

Precision routing provides multidimensional routing with simple configuration, scripting, and reporting. Agents are represented through multiple attributes with proficiencies so that the capabilities of each agent are accurately exposed, bringing more value to the business.

If your routing needs are not too complex, consider using one or two skill groups. However, if you want to conduct a search involving as many as ten different proficiency levels in one easily managed queue, use precision queues.

## Attributes

Attributes identify a call routing requirement, such as language, location, or agent expertise. You can create two types of attributes: Boolean or Proficiency.

When you create a precision queue, you identify which attributes are part of that queue and then implement the queue in a script. When you assign a new attribute to an agent and the attribute value matches the precision queue criteria, the agent is automatically associated with the precision queue.

You must take system limits into account when you assign attributes to agents, and satisfy both of the following conditions:

- A) an agent can have a maximum of 50 attributes

and

- B) an agent can belong to a maximum total of 50 combined precision queues and skill groups

Failure to meet both of these conditions will result in an unsuccessful configuration operation.

For example, if a particular attribute is used in many precision queues, and that attribute is assigned to an agent, that agent belongs to all of those precision queues. It is therefore possible to exceed condition B by assigning just a few attributes to an agent, if those attributes are used in many precision queues.

It is therefore prudent to plan carefully and to keep system limits in mind when creating attributes and adding them to precision queues.

Navigate to **Unified CCE Administration** > **Organization** > **Skills** > **Attributes** to configure attributes.

Administrators can see and manage attributes. Supervisors can configure attributes for their supervised agents on the Attributes tab of the Agents tool.

# Precision Queue Call Flow Example

At a high level, consider a 5-step precision queue with a Consider If formula for *Caller is Premium Member* attached to the Step 1:

- Step 1 - Attribute: Skill > 8 - Consider If: Caller is Premium Member
- Step 2 - Attribute: Skill > 6
- Step 3 - Attribute: Skill > 4
- Step 4 - Attribute: Skill > 3
- Step 5 - Attribute: Skill >= 1

Caller John, who is not a premium customer, calls 1-800-repairs. John's call is routed to this precision queue.

- Since John is not a premium customer, John is immediately routed out of Step 1 (because of the Consider If on Step 1) and into Step 2 where John waits for the call to be answered.
- After the Step 2 wait time has expired, John's call moves to Step 3 to wait for an agent.
- After the Step 3 wait time has expired, John's call moves to Step 4 to wait for an agent.
- When it arrives at Step 5, John's call will wait indefinitely for an available agent. This step cannot be avoided by any call because there is no routing logic past this.

The overarching idea is that customer will use each successive step to expand the pool of available agents. Eventually, when you reach the "last" step (the step with the highest number), the call is waiting in a potentially very large pool of agents. With each extra step, the chances of the call being handled increase. This also puts the most valuable and skilled agents in the earlier precision queue steps. Calls come to them first before moving on the less appropriate agents in later steps.

**Note** When two or more agents have the same proficiently level for the attributes the PQ step leverages the Longest Available Agent (LLA).

# Scripts for Precision Queues

To implement Precision Routing in your contact center, you must create scripts.

You can create and use configured (static) and dynamic precision queue nodes in your scripts.

- Static precision queue nodes target a single, configured precision queue. When the script utilizes a single precision queue, use static precision queues.
- Dynamic precision queue nodes are used to target one or more previously configured precision queues. Use dynamic precision queues when you want a single routing script for multiple precision queues (for example, when the overall call treatment does not vary from one precision queue to another). Dynamic precision queues can simplify and reduce the overall number of routing scripts in the system.

# Precision Queue Script Node

Use the Precision Queue script node to queue a call based on caller requirements until an agent with desired proficiency become available. This node contains multiple agent selection criteria which are separated into steps.

A single call can be queued on multiple precision queues. If an agent becomes available in one of the precision queues, the call is routed to that resource. You cannot reference multiple precision queues with a single Precision Queue node. However, you can run multiple Precision Queue nodes sequentially to achieve this.

The Precision Queue node includes a Priority field, which sets the initial queuing priority for the calls processed through this node versus other calls queued to the other targets using different nodes. The priority is expressed as an integer from 1 (top priority) to 10 (least priority). The default value is 5.

If more than one call is queued to a precision queue when an agent becomes available, the queued call with the lowest priority number is routed to the target first. For example, assume an agent in a precision queue becomes available and two calls are queued to that precision queue. If one call has priority 3 and the other has priority 5, the call with priority 3, the lower value, is routed to the precision queue while the other call continues to wait. If the priorities of the two calls are same, then the call queued first is routed first.

VRU (voice response unit) script instructions are not sent to the VRU. If a call enters the precision queue node and no resource is available, the call is queued to the precision queue and the node transfers the call to the default VRU, if the call is not already on a VRU. The script flow then exits immediately through the success branch. The script should then continue with a run external script node to instruct the VRU what to do while holding the call until an agent becomes available. Typically, this invokes a network VRU script that plays music-on-hold, possibly interrupted on a regular basis with an announcement. The script flow can also use other queuing nodes to queue the same call to other targets, for example, Queue to Skill Group and Queue to Agent.

**Note** Non-voice tasks can also be picked or pulled out of turn from queues, not necessarily based on the priority of the call. Such non-voice tasks that are picked or pulled by a specific agent, require a Pick/Pull node to be used in the ICM script. However, the agents belonging to other skill groups or precision queues can also pick tasks that may be queued in Skill Groups or Precision Queues other than their own. These are denoted by **Picked by another Skillgroup/PQ** or **Pulled by another Skillgroup/PQ** monitor labels, when viewing the scripts in monitor mode.

# Queuing Behavior of the Precision Queue Node

Precision queues internally are configured with one or more time-based steps, each with a configured wait time. After a call is queued, the first step begins and the timer starts. This occurs although the path of the script exited the success node and a new node may be targeted (for example, Run Ext. Script).

If the timer for the first step expires, control moves to the second step (assuming one exists), and so on. As long as the call remains in queue and there are steps left to perform, the call internally continues to move between steps regardless of the path the call takes after it leaves the precision queue node. If a call is queued to two or more precision queues, the call internally walks through the steps for each precision queue in parallel. After the call reaches the last step on a precision queue, it remains queued on that step until the call is routed, abandoned, or ended.

If there is an update to the precision queue definition, then all queued calls in the precision queue are re-evaluated and are re-run from the first step.

For example, consider the wait time for an ongoing call at step 1 to be 1080 seconds, of which 1000 seconds has already elapsed. Now, suppose the wait time is changed to 900 seconds, then the wait time for this call is also reset to 900 seconds, even though only 80 more seconds are left to move to the next step.

# Dynamic Limits for Skill Groups and Precision Queues Per Agent

The number of skill groups and precision queues per agent significantly affects the following subcomponents of Unified CCE:

- Cisco Finesse servers
- Agent PGs
- Router
- Logger

**Note** We use *queue* as a common term for skill groups and precision queues.

To maintain the performance of your solution, periodically remove unused queues.

The Reference Designs set a standard limit for the average queues per agent on each PG. On a particular PG, some agents can have more queues than other agents. As long as the average across all the agents on the PG is within the limit, you can still have the maximum active agents on that PG.

For example, assume that you have three groups of agents on a PG in a 4000 Agent Reference Design:

- Group A has 500 agents with five queues each.
- Group B has 1000 agents with 15 queues each.
- Group C has 500 agents with 25 queues each.

These three groups average to 15 queues per agent, so you can have them all on a single PG under the standard limits.

You can also exceed that standard limit if you reduce the number of agents on each PG and on the whole system.

**Note** See the configuration tables in the configuration limits chapter for the standard limits.

The Cisco Finesse server doesn't display statistics for unused queues. So, the active queues affect the performance of the Cisco Finesse server more than the total configured queues.

The Cisco Finesse desktop updates queue (skill group) statistics at 10-second intervals. The Cisco Finesse Desktop also supports a fixed number of queue statistics fields. You can't change these fields.

This table shows the approximate reduction in the number of agents your solution can support with more queues per agent:

*Table 6: Dynamic Agents and Queues Limits*

| Average Queues per Agent | Maximum Agents per PG | Maximum Agents for 2000 Agent Reference Design | Maximum Agents for 4000 Agent Reference Design[1] | Maximum Agents for 12000 Agent Reference Design | Maximum Agents for 24000 Agent Reference Design |
|---|---|---|---|---|---|
| 10 | 2000 | 2000 | 4000 | 12000 | 24000 |
| 15 | 2000 | 2000 | 4000 | 12000 | 16000 |
| 20 | 1500 | 1500 | 3000 | 9000 | 12000 |
| 30 | 1000 | 1000 | 2000 | 6000 | 8000 |
| 40 | 750 | 750 | 1500 | 4500 | 6000 |
| 50 | 600 | 600 | 1200 | 3600 | 4800 |

[1] You can't have more than 4000 Agents on a Rogger deployment.

Unified CCE supports a maximum of 50 unique skill groups across all agents on a supervisor's team, including the supervisor's own skill groups. If this number is exceeded, all skill groups that are monitored by the supervisor still appear on the supervisor desktop. However, exceeding this number can cause performance issues and isn't supported.

**Note** Each precision queue that you configure creates a skill group for each Agent PG and counts toward the supported number of skill groups per PG. The skill groups are created in the same Media Routing Domain as the precision queue.

# Initial Setup

When you configure precision queues associated with a large number of agents, the system avoids potential overload conditions by updating the agent associations as system resources allow. Updates may take a few minutes. If you submit multiple configuration updates, the system has a threshold of five concurrent configuration updates, and will reject any updates that exceed the threshold.

## Add Attributes

**Procedure**

**Step 1** Navigate to **Unified CCE Administration** > **Organization** > **Skills** > **Attributes**.

**Step 2** In the **List of Attributes** window, click **New**. The **New Attributes** window has two tabs: General and Member.

**Step 3** Complete the following fields on the **General** tab:

| Field | Required | Description |
|---|---|---|
| **Name** | yes | Type a unique attribute name. For example, to create an attribute for mortgage insurance, type *mortgage*. |
| **Description** | no | Enter a maximum of 255 characters to describe the attribute. |
| **Type** | no | Select the type: Boolean or Proficiency. |
| **Default** | no | Select the default (True or False for Boolean, or a number from 1 to 10 for Proficiency). |

**Step 4** Click **Save**.

# Search for Agents

The Search field in the Agents tool offers an advanced and flexible search.

Click the + icon at the far right of the **Search** field to open a popup window, where you can:

- Select to search for agents only, supervisors only, or both.
- Select to search for all agents or only ECE enabled agents.
- Enter a username, agent ID, first or last name, or description to search for that string.
- Enter one or more site names separated by spaces. (Site is an OR search.)
- Enter one or more peripheral set names separated by spaces (Peripheral Set is an OR search). The search is case-insensitive and does not support partial matches.

**Note** Search by department is available only when departments are configured.

# Assign Attributes to Agents

**Procedure**

**Step 1** With the selected agent displayed, click the **Attributes** tab.

**Step 2** Complete the **Attributes** tab:

This tab shows the attributes associated with this agent and their current values.

Click **Add** to open a popup list of all attributes, showing the name and current default value for each.

a) Click the attributes you want to add for this agent.

b) Set the attribute value as appropriate for this agent.

# Add Precision Queue

**Procedure**

**Step 1** Navigate to **Unified CCE Administration** > **Organization** > **Skills** > **Precision Queues**.

This opens a **List of Precision Queues** window showing all precision queues that are currently configured.

**Step 2** Click **New** to open the **New Precision Queue** window. Complete the fields.

| Name | Required | Description |
|---|---|---|
| **Description** | no | Enter up to 255 characters to describe the precision queue. |
| **Media Routing Domain** | no | MRDs organize how requests for media are routed. The system routes calls to skill groups or precision queues that are associated with a particular communication medium; for example, voice or email. This field defaults to *Cisco_Voice*. |

| Name | Required | Description |
|---|---|---|
| **Service Level Type** | yes | Select the service level type used for reporting on your service level agreement.<br><br>Service level type indicates how calls that are abandoned before the service level threshold affect the service level calculation.<br><br>• **Ignore Abandoned Calls** (the default): Select this option if you want to exclude abandoned calls from the service level calculation.<br><br>• **Abandoned Calls have Negative Impact:** Select this option if you want only those calls that are answered within the service level threshold time to be counted as treated calls. The service level is negatively affected by calls that abandon within the service level threshold time.<br><br>• **Abandoned Calls have Positive Impact:** Select this option if you consider a call that is abandoned within the service level threshold time as a treated call. With this configuration, abandoned calls have a positive impact on the service level. |
| **Service Level Threshold** | yes | Enter the time in seconds that calls are to be answered based on your service level agreement, from 0 to 2,147,483,647.<br><br>The time that you enter in this field is used to report on service level agreements and does not affect how long a call remains in a precision queue. The length of time a call remains in a step is determined by the wait time for each individual step. |

| Name | Required | Description |
|---|---|---|
| **Agent Order** | yes | Select an option to determine which agents receive calls from this queue.<br>The ordering of agents does not dictate the agents who are selected into a Precision Queue step. Agents are included or excluded based on the conditions specified for the step.<br>• **Longest Available Agent** (the default): The default method of agent ordering for a precision queue. The call is delivered to the agent who has been in the available (or ready) state the longest.<br>• **Most Skilled Agent:** The call is delivered to the agent who has the highest competency sum from all the attributes pertinent to the Precision Queue step. In an agent-rich environment, this can mean that more competent agents would be utilized more than less competent agents.<br>• **Least Skilled Agent:** The call is delivered to the agent who has the lowest competency sum from all the attributes pertinent to the Precision Queue step. |
| **Bucket Intervals** | no | Select the bucket interval whose bounds are to be used to measure the time slot in which calls are answered.<br>The field defaults to the system default.<br>To select a different bucket interval: |

**Step 3** Click the numbered Step Builder link (Step 1, Step 2, and so on) to build a precision queue step in the **Step Builder** popup window.

**Step 4** When you have finished adding, click **Save**.

# Consider If Formula for Precision Queue

If you are not on the last step of the precision queue, then you can enter a *Consider If* formula for that step. A Consider If formula evaluates a call (within a step) against additional criteria. Each time a call reaches a step with a Consider If expression, the expression is evaluated. If the value for the expression returns as true, the call is considered for the step. If the value returns as false, the call moves to the next step. If no expression is provided for a step, the step is always considered for calls.

To add a Consider If formula, type the formula into the **Consider If** box. Alternatively, you can use the Script Editor to build the formula and then copy and paste it into the **Consider If** box. Objects used in Consider If formulas are case-sensitive. All Consider If formulas that you add to a precision queue must be valid. If you add an invalid formula, you cannot save the precision queue. To ensure that the formula is valid, use Script Editor to build and validate the formula.

Only the following scripting objects are valid in a Consider If formula:

- Call
- PQ
- Skillgroup
- ECC
- PQ Step
- Call Type
- Custom Functions (You can create custom functions in Script Editor.)

It is possible that a valid Consider If formula can become invalid. For example, if you delete an object used in the formula after you create or update the precision queue, the formula is no longer valid.

**Consider If Formula Examples**

- **PQ.PQ1.LoggedOn > 1**--Evaluates whether there is more than one agent logged in to this queue.
- **CallType.CallType1.CallsRoutedToday > 100**--Evaluates whether more than 100 calls of this call type were routed today.
- **PQStep.PQ1.1.RouterAgentsLoggedIn > 1**--Evaluates whether there is more than one router agent logged in to this queue for Step 1.
- **CustomFunction(Call.PeripheralVariable1) > 10**--Evaluates whether this formula using a custom function returns a value greater than 10.

# Build Precision Queue Steps

Every precision queue must have a step, and every step must have an Expression. An Expression is a collection of attribute terms.

**Procedure**

**Step 1** Click the numbered step link in the **Steps** panel (Step 1, Step 2, and so on).

The step number popup window opens.

**Step 2** Build the first step as follows.

a) Click the **magnifying glass** icon to the right of the Select Attribute field in the Expression 1 panel.
b) Select an attribute from the list.
c) Use the two **Select** fields to establish the terms of the attribute. Click the first **Select** field to choose an operator.

- For Boolean attributes, choices are the operators for Equal and Not Equal.
- For Proficiency attributes, choices are the operators for True, False, Less Than, Less Than or Equal To, Greater Than, and Greater Than or Equal To.

d) Click the second **Select** field to choose a value.

- For Boolean attributes, values are True and False.
- For Proficiency attributes, values are numbers from 1 to 10.

Your selection creates an attribute term for the Expression.

**Step 3** To add a second attribute to the first Expression, click **Add Attribute** in the **Expression 1** row.

a) Select **AND** or **OR** to establish the relationship between the first and second attributes.
b) Repeat steps 2b, 2c, and 2d.

**Step 4** Continue to add attributes to Expression 1.

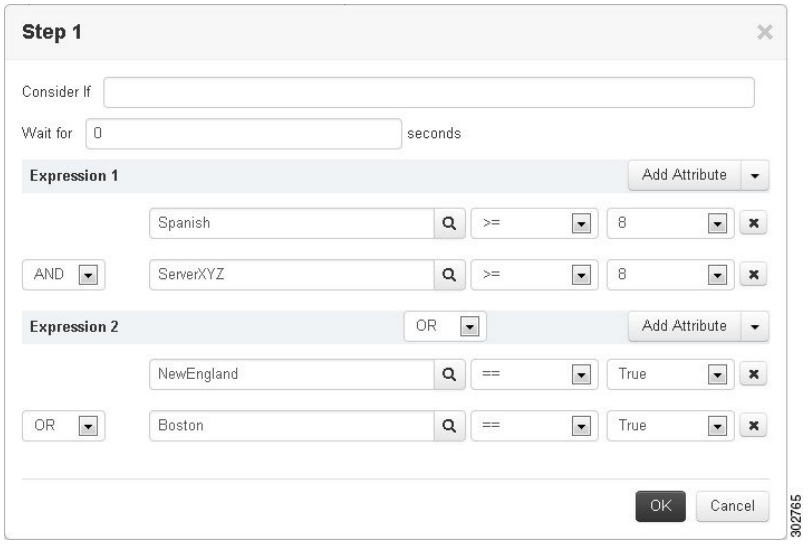
All attributes within an expression must be joined by the same logical operator. They must all be ANDs, or they must all be ORs.

**Step 5** To add a second Expression, click the **Add Attribute** drop-down in the **Expression 1** row and select **Add Expression**.

**Step 6** Select **AND** or **OR** to establish the relationship between the first and second Expressions.

**Step 7** Add attributes to Expression 2.

**Step 8** Continue to add Expressions as needed.



In this example, a Spanish caller located in the Boston area needs an onsite visit from a technician to repair his ServerXYZ. An ideal agent should be fluent in Spanish and have the highest proficiency in ServerXYZ. This can be seen in Expression 1. Expression 2 allows us to specify that the selected agent must also be from either Boston or the New England area.

**Step 9** When you have completed the step, click **OK** to add it to the precision queue.

**Step 10** To build the next step, click **Add Step**.

Each successive step is prepopulated with the Expressions and attributes of its predecessor. Decrease the attribute qualifications and competencies in successive steps to lower the bar such that the pool of acceptable agents increases.

**Step 11** When you have created all steps, you can open any step *except the last* and enter values in the **Consider if** and **Wait for** fields.

- **Consider if** is a formula that evaluates a call within a step against additional criteria. (See Consider If Formula for Precision Queue, on page 124 for more information about Consider If.)
- **Wait for** is a value in seconds to wait for an available agent. A call will queue at a particular step and wait for an available agent matching that step criteria until the number of seconds specified. A blank wait time indicates that the call will proceed immediately to the next step if no available agents match the step criteria. Wait time defaults to 0 and can take a value up to 2147483647.

# Configure a Static Precision Queue

**Procedure**

**Step 1** In the **Precision Queue Properties** dialog box, select the **Statically** option.

**Step 2** From the list, select a precision queue to which to route all calls that enter this node.

**Step 3** In the **Priority selection** box, select the initial queuing priority for calls processed through this node. You can select from 1 - 10. The default is 5.

**Step 4** Check the **Enable target requery** check box to enable the requery feature for calls processed through this node.

**Step 5** Check the **Wait if Agents Not Logged In** check box.

If this check box is selected and the agents associated with this step are not logged in, then the router waits for the time that is configured for that step. Whereas, if this check box is not selected, the router does not wait on any step.

**Note** The router waits indefinitely on the last step, irrespective of the selection of this check box.

**Step 6** To edit a precision queue, select a precision queue from the list, and then click **Edit Precision Queue**.

# Configure a Dynamic Precision Queue

**Procedure**

**Step 1** In the **Precision Queue Properties** dialog box, select the **Dynamically** option.

**Step 2** In the **Priority selection** section, select the initial queuing priority for calls processed through this node. You can select from 1 - 10. The default is 5.

**Step 3** Check the **Enable target requery** check box to enable the requery feature for calls processed through this node.

**Step 4** Check the **Wait if Agents Not Logged In** check box.

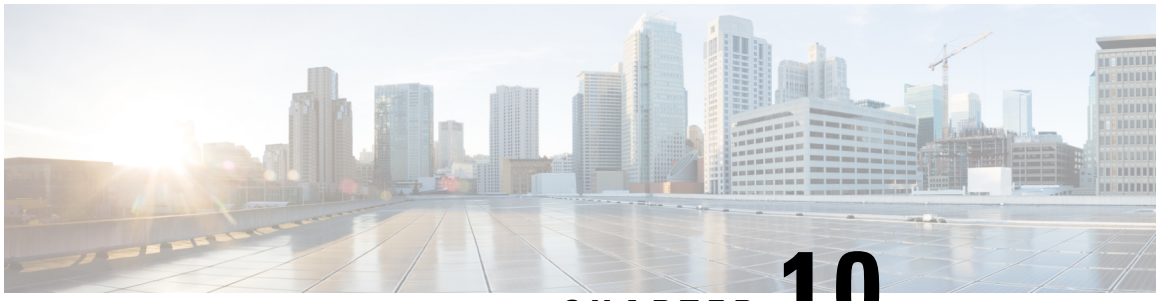
If this check box is selected and the agents associated with this step are not logged in, then the router waits for the time that is configured for that step. Whereas, if this check box is not selected, the router does not wait on any step.

**Note** The router waits indefinitely on the last step, irrespective of the selection of this check box.

**Step 5** Select a queue option:

- To dynamically route calls that enter this node to a precision queue name, select the **Precision Queue Name** option.
- To dynamically route calls that enter this node to a precision queue ID, select the **Precision Queue ID** option.

**Step 6** Click **Formula Editor** to create a formula that determines the precision queue name or ID to which to route calls.



CHAPTER 10

# Single Sign-On



## Single Sign-On

Single sign-on (SSO) is an authentication and authorization process. (Authentication proves that you are the user you say that you are, and authorization verifies that you are allowed to do what you want to do.) SSO allows you to sign in to one application and then securely access other authorized applications without a prompt to resupply user credentials. SSO permits Cisco supervisors or agents to sign on only once with a username and password. Supervisors and agents gain access to all of their Cisco browser-based applications and services within a single browser instance. By using SSO, Cisco administrators can manage all users from a common user directory and enforce password policies for all users consistently.

**Note** Before enabling SSO in Unified CCE, ensure to sign in to the Cisco Unified Intelligence Center OAMP interface and perform the Unified CCE User Integration operation (Cluster Configuration > UCCE User Integration) once manually to import the Supervisors with the required roles.

SSO is an optional feature whose implementation requires you to enable the HTTPS protocol across the enterprise solution.

You can implement single sign-on in one of these modes:

- **SSO** - Enable *all* agents and supervisors in the deployment for SSO.
- **Hybrid** - Enable agents and supervisors *selectively* in the deployment for SSO. Hybrid mode allows you to phase in the migration of agents from a non-SSO deployment to an SSO deployment and enable SSO for local PGs. Hybrid mode is useful if you have third-party applications that don't support SSO, and some agents and supervisors must be SSO-disabled to sign in to those applications.
- **Non-SSO** - Continue to use existing Active Directory-based and local authentication, without SSO.

SSO uses Security Assertion Markup Language (SAML) to exchange authentication and authorization details between an identity provider (IdP) and an identity service (IdS). The IdP authenticates based on user credentials, and the IdS provides authorization between the IdP and applications. The IdP issues SAML assertions, which are packages of security information transferred from the IdP to the service provider for user authentication. Each assertion is an XML document that contains trusted statements about a subject including, for example, username and privileges. SAML assertions are digitally signed to ensure their authenticity.

The IdS generates an authentication request (also known as a SAML request) and directs it to the IdP. SAML does not specify the method of authentication at the IdP. It may use a username and password or other form of authentication, including multi-factor authentication. A directory service such as LDAP or AD that allows you to sign in with a username and a password is a typical source of authentication tokens at an IdP.

### Prerequisites

The Identity Provider must support Security Assertion Markup Language (SAML) 2.0. See the *Compatibility Matrix* for your solution at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-device-support-tables-list.htmlhttps://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-device-support-tables-list.html for details.

# Contact Center Enterprise Reference Design Support for Single Sign-On

Unified CCE supports single sign-on for these reference designs:

- 2000 Agents
- 4000 Agents
- 12000 Agents
- 24000 Agents
- Contact Director (Maximum of 24000 agents, Each target system must include a dedicated Cisco IdS deployment.)

# Coresidency of Cisco Identity Service by Reference Design

| Reference Design | Unified CCE |
|---|---|
| 2000 Agent | Cisco IdS is coresident with Unified Intelligence Center and Live Data on a single VM. |
| 4000 Agent | Standalone Cisco IdS VM |

| Reference Design | Unified CCE |
|---|---|
| 12000 Agent | Standalone Cisco IdS VM |
| 24000 Agent | Standalone Cisco IdS VM |

## Single Sign-On Support and Limitations

Note the following points that are related to SSO support:

- To support SSO, enable the HTTPS protocol across the enterprise solution.
- SSO supports agents and supervisors only. SSO support is not available for administrators in this release.
- SSO supports multiple domains with federated trusts.
- SSO supports only contact center enterprise peripherals.
- SSO support is available for Agents and Supervisors that are registered to remote or main site PG in global deployments.

Note the following limitations that are related to SSO support:

- SSO support is not available for third-party Automatic Call Distributors (ACDs).
- The SSO feature does not support Cisco Finesse IP Phone Agent (FIPPA).
- The SSO feature does not support Cisco Finesse Desktop Chat.
- In Hybrid mode,
  - When an agent in SSO mode tries to log in to CUIC, and if the agent does not exist in CUIC, the agent cannot log in to CUIC.
  - When a Supervisor in SSO mode tries to log in to CUIC, and if the Supervisor user does not exist in CUIC, the Supervisor cannot log in to CUIC. For the Supervisor to log in to CUIC, perform Unified CCE User Integration. For more information on Unified CCE User Integration, see *Administration Console User Guide for Cisco Unified Intelligence Center* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-intelligence-center/products-maintenance-guides-list.html.

# Single Sign-On Configuration Flow

**Note** To ensure that token validations based on token lifetimes are correctly applied, it is mandatory that you synchronize the time in Cisco IdS, IdP, and all IdS clients, including VPN-Less reverse proxy hosts, to the same NTP source (preferred) or to the same NTP stratum.

**Note** It is recommended that the Administrator configures SSO from the IdS publisher node.

1. Install the appropriate release of the CCE solution. For more information, see Cisco Unified Contact Center Enterprise Installation and Upgrade Guide at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-guides-list.html
2. Install the Cisco Identity Service (Cisco IdS). For more information, see Cisco Unified Contact Center Enterprise Installation and Upgrade Guide at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-guides-list.html
3. Configure an Identity Provider (IdP).
4. Configure System Inventory.
5. Configure the Cisco IdS.
6. Register and test SSO-compatible components with the Cisco IdS.
7. Choose the SSO mode.
8. Enable multiple users at once for SSO by using the SSO migration tool, or enable users one at time by using the configuration tools.

### Related Topics

Configure the Cisco Identity Service, on page 140
Configure an Identity Provider (IdP), on page 132
Migrate Agents and Supervisors to Single Sign-On Accounts, on page 146
Register Components and Set Single Sign-On Mode, on page 143

# Configure an Identity Provider (IdP)

To support SSO for the contact center solution, configure an Identity Provider (IdP) that is compliant with the Security Assertion Markup Language 2.0 (SAML v2) Oasis standard. The IdP stores user profiles and provides authentication services to the contact center solution.

**Note** For a current list of supported Identity Provider products and versions, see the *Contact Center Enterprise Compatibility Matrix*.

This section provides sample configuration information for Microsoft AD FS.

Follow this sequence of tasks to configure the Identity Provider.

| Sequence | Task |
|---|---|
| 1 | Install and Configure Active Directory Federation Services, on page 133 |
| 2 | Set Authentication Type. See Authentication Types, on page 133. |
| 4 | Enable Signed SAML Assertions, on page 136 |
| 5 | Optionally Customize the AD FS Sign-In Page in Windows Server to Allow User ID, on page 138 |

# Install and Configure Active Directory Federation Services

Follow Microsoft instructions and guidelines to install Microsoft Active Directory Federation Services (AD FS).

For example, see *Active Directory Federation Services Overview* at https://technet.microsoft.com/en-us/library/hh831502(v=ws.11).aspx

- For AD FS in Windows Server, see *AD FS Technical Reference* at https://docs.microsoft.com/en-us/windows-server/identity/active-directory-federation-services.

**Note** SSO for Unified CCE supports IdPs other than MS, and AD FS. For the list of supported IdPs see the Compatibility matrix https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-device-support-tables-list.html

**Note** Cisco IdS does not support AD FS Automatic Certificate Rollover. If the AD FS certificate gets rolled over, then re-establish the trust relationship between the IdS and AD FS.

# Authentication Types

Cisco Identity Service supports form-based authentication and Kerberos windows authentication of the Identity Provider.

For information on enabling form-based authentication in ADFS, see Microsoft documentation:

- For ADFS 3.0 see https://blogs.msdn.microsoft.com/josrod/2014/10/15/enabled-forms-based-authentication-in-adfs-3-0/

For Kerberos authentication to work, ensure to disable the form-based authentication.

# Integrate Cisco IdS with AD FS

Follow these steps to integrate the Cisco IdS with AD FS.

**Procedure**

**Step 1** In the Server Manager, open **AD FS Management**.

**Step 2** For form-based authentication, set the **Authentication Methods** for **Intranet** to **Forms Authentication**.

**Step 3** Download LAN SP metadata and reverse-proxy cluster SP metadata from the Cisco IdS publisher.

- Open the **Identity Service Management** console at `https://<CiscoIdS server address>:8553/idsadmin`

  From the menu on the left, select **Settings**. In the **IdS Trust** tab, download the XML file.

- In Unified CCE Administration, go to **Infrastructure Settings**> **Device Configuration** > **Identity Service** > **Identity Service Settings**.

  In the **IdS Trust** tab, download the XML file.

**Note** Ensure your browser's security settings allow downloads from the Cisco IdS site.

**Step 4** Download the IdP metadata file, `federationmetadata.xml`, from the following location:

`https://<ADFS Server FQDN>/federationmetadata/2007-06/federationmetadata.xml`

**Step 5** Do one of the following to upload the IdP metadata file, you downloaded at step 4, to the Cisco Ids server:

- In the **Identity Service Management** console, select **Settings** > **IdS Trust**.

  Click **Next** and then click **Upload Idp Metadata**.

- In the **Unified CCE Administration** console, navigate to **Infrastructure Settings** > **Device Configuration** > **Identity Service** > **Identity Service Settings** > **Ids Trust**.

  Click **Next** and then click **Upload Idp Metadata**.

**Note** Cisco IdS supports SAML self-signed certificates for authorization and authentication.

**Step 6** Follow these steps to create a Relying Party Trust.

a) In the Server Manager, open **AD FS Management**.
b) Select the **Add Relying Party Trust** option from the AD FS menu.
c) In the **Add Relying Party Trust** wizard, click **Select Data Source**.
d) Select the **Import data about the relying party from a file** option and then click **Browse** to the open the SAML SP metadata XML file you downloaded at Step 3 and click **Next**.
e) In the **Display name** field, enter a unique name for the relying party and click **Next**.
f) *This step is applicable only for Windows Server 2012 R2*. In the **Configure Multi-factor Authentication Now** step, select **I do not want to configure multi-factor authentication settings for the relying party at this time.**
g) Select the option that permits all users and click **Next**.
h) Skip the option to edit the Rule/Claim Issuance Policy for now (you edit the policy from Step 8 onwards) and click **Close** to complete adding the relying party trust.

**Step 7** Do the following to set the properties for the Relying Party Trust created at Step 5. Right-click on the Relying Party Trust and click **Properties**. In the **Properties** window:

- Configure the following under the **Identifiers** tab:

| Field | Description |
|---|---|
| Display name | The unique name of the identifier. |
| Relying party identifier | FQDN of the publisher node of Cisco Identity Server from which you downloaded the Cisco IdS metadata file at step 3. |
| | FQDN of the subscriber node of Cisco Identity Server. |

- Under the **Advanced** tab, choose **SHA-256** from the **Secure hash algorithm** field.

**Step 8** In the list of Relying Party Trusts, right-click on your Relying Party Trust, and select the option to edit the Rules/Claim Issuance Policy from the menu.

**Step 9** In the **Edit Claim Rules**/**Edit Claim Issuance Policy** window that opens, click **Add Rule** and then click **OK**.

**Step 10** In the **Add Transform Claim Rule Wizard** that opens, follow these steps to create the first claim:

a) In the **Choose Rule Type** step, select **Send LDAP Attributes as Claims** from the **Claim rule template** drop-down list and click **Next**.

b) In the **Configure Claim Rule** step, configure the following:

| Field | Description |
|---|---|
| Claim Rule Name | Enter "NameID" |
| Attribute Store | Select **Active Directory**. |
| Mapping of LDAP Attributes to Outgoing Claims | If the identifier is a Security Account Name (SAM), do the following:<br>• Select **SAM-Account-Name** as one of the LDAP attributes and set the **Outgoing Claim Type** to "uid."<br>• Select **User-Principal-Name** as one of the LDAP attributes and set the **Outgoing Claim Type** to "user_principal".<br><br>If the identifier is a User Principal Name (UPN), do the following:<br>• Select **User-Principal-Name** as one of the LDAP attributes and set the **Outgoing Claim Type** to "uid."<br>• Select **User-Principal-Name** again as the LDAP attribute and set the **Outgoing Claim Type** to "user_principal."<br><br>The "uid" identifies the authenticated user in the claim sent to the applications.<br>The "user_principal" identifies the authentication realm of the user in the assertion sent to Cisco Identity Service. |

c) Click **Finish**.

**Step 11** Repeat steps 8 and 9 to open the **Add Transform Claim Rule** wizard. In the **Add Transform Claim Rule** wizard, follow these steps to create the second claim:

a) In the **Choose Rule Type** step, select **Send Claims Using a Custom Rule** from the **Claim rule template** drop-down list and click **Next**.

b) In the **Configure Claim Rule** step, configure the following:

1. In the **Claim rule name** field, enter the FQDN of the Cisco Identity Server publisher's primary node.
2. Add the following to the **Custom Rule** field:

```
c:[Type ==
"http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname"] =>
```

```
issue(Type = "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier",
Issuer = c.Issuer, OriginalIssuer = c.OriginalIssuer, Value = c.Value, ValueType =
c.ValueType,
Properties["http://schemas.xmlsoap.org/ws/2005/05/identity/claimproperties/format"]
 =
"urn:oasis:names:tc:SAML:2.0:nameid-format:transient",
Properties["http://schemas.xmlsoap.org/ws/2005/05/identity/claimproperties/namequalifier"]
 =
"http://<AD FS Server FQDN>/adfs/services/trust",
Properties["http://schemas.xmlsoap.org/ws/2005/05/identity/claimproperties/spnamequalifier"]
 =
"<fully qualified domain name of Cisco IdS>");
```

c) Edit the script as follows:

- Replace **<ADFS Server FQDN>** to match exactly (including case) the ADFS server FQDN (fully qualified domain name.)
- Replace **<Cisco IdS server FQDN>** to match exactly (including case) the Cisco Identity Server FQDN.

**Step 12** Click **OK**.

# Enable Signed SAML Assertions

Enable Signed SAML Assertions for the Relying Party Trust (Cisco Identity Service).

**Procedure**

**Step 1** Click **Start** and type **powershell** in the Search field to display the Windows Powershell icon.

**Step 2** Right-click on the Windows Powershell program icon and select **Run as administrator**

**Note** All PowerShell commands in this procedure must be run in Administrator mode.

**Step 3** Run the command, **Set-ADFSRelyingPartyTrust -TargetName <Relying Party Trust Display Name> -SamlResponseSignature "MessageAndAssertion"**.

**Note** Set <Relying Party Trust Display Name> to exactly match (including case) the Identifier tab of the Relying Party Trust properties.

For example:

```
Set-ADFSRelyingPartyTrust -TargetName CUICPub.PCCERCDN.cisco.com
-SamlResponseSignature "MessageAndAssertion".
```

**Step 4** Navigate back to the Cisco Identity Service Management window.

**Step 5** Click **Settings**.
By default **IdS Trust** tab is displayed.

**Step 6** Click **Next** as you have already downloaded the required metadata.

**Step 7** Click **Next** as you have already established trust relationship between IdP and IdS.

The configured IdP Entity ID is listed.

**Note** If reverse-proxy is configured for IdP, the IdP proxy url is listed at the bottom of the page.

**Step 8** Click **Test SSO Setup** to test the required entity where the **SSO Status** displays **Needs Validation**. **SSO Status** can be **Successful**, **Unsuccessful**, or **Needs Validation**.

**Note** If **Unsuccessful**, ensure that the claim you created on the AD FS is enabled or the rule has the correct names for IdS and AD FS.

Administrator client machine requires connectivity to reverse-proxy nodes for validating SSO connection with reverse-proxy.

# Multi-Domain Configuration for Federated ADFS

In Multi-Domain Federation in ADFS, an ADFS in one domain provides federated SAML authentication for users in other configured domains. In such cases, additional configuration is required:

- Primary ADFS Configuration that refers to the ADFS to be used in IdS.
- Federated ADFS Configuration that refers to the ADFS, whose users can log in via IdS, thus is the primary ADFS.

## Federated ADFS Configuration

In each federated ADFS, create the relying party trust for primary ADFS and the claim rules configured.

## Primary ADFS Configuration

### Before you begin

In the Claim Provider Trust, ensure that the **Pass through or Filter an Incoming Claim** rules are configured with pass through all claim values as the option

### Procedure

**Step 1** Name ID

**Step 2** Choose Name ID from Incoming Claim Type drop box

**Step 3** Choose **Transient** as the option for Incoming NameID format

**Step 4** uid: This is a custom claim. Enter the value uid in the **Incoming Claim Type** drop box.

**Step 5** user_principal: This is a custom claim. Type the value user_principal in the **Incoming Claim Type** drop box.

In the relying party trust for IdS, add **Pass though or Filter an Incoming Claim** rules with pass through all claim values as the option.

**Step 6** NameIDFromSubdomain

**Step 7** Choose Name ID from Incoming Claim Type drop box

**Step 8** Choose Transient as the option for Incoming NameID format

**Step 9** uid: This is a custom claim. Type the value uid in the Incoming Claim Type drop box

**Step 10** user_principal: This is a custom claim. Type the value user_principal in the Incoming Claim Type drop box

# Kerberos Authentication (Integrated Windows Authentication)

### Before you begin

The CCE solution supports Kerberos authentication.

# Optionally Customize the AD FS Sign-In Page in Windows Server to Allow User ID

By default, the sign-in page presented to SSO users by AD FS in Windows Server requires a username that is a UPN. Usually this is an email format, for example, user@cisco.com. If your contact center solution is in a single domain, you can modify the sign-in page to allow your users to provide a simple User ID that does not include a domain name as part of the user name.

There are several methods you can use to customize the AD FS sign-in page. Look in the Microsoft AD FS in Windows Server documentation for details and procedures to configure alternate login IDs and customize the AD FS sign-in pages.

The following procedure is an example of one solution.

### Procedure

**Step 1** In the AD FS **Relying Party Trust**, change the NameID claim rule to map the chosen LDAP attribute to **uid**.

**Step 2** Click the Windows **Start** control and type **powershell** in the Search field to display the Windows Powershell icon.

**Step 3** Right-click on the Windows Powershell program icon and select **Run as administrator**

All PowerShell commands in this procedure must be run in Administrator mode.

**Step 4** To allow sign-ins to AD FS using the sAMAccountName, run the following Powershell command:

```
Set-AdfsClaimsProviderTrust -TargetIdentifier "AD AUTHORITY" -AlternateLoginID sAMAccountName
-LookupForests myDomain.com
```

In the LookupForests parameter, replace `myDomain.com` with the forest DNS that your users belong to.

**Step 5** Run the following commands to export a theme:

```
mkdir C:\themeExport-AdfsWebTheme -Name default -DirectoryPath c:\theme
```

**Step 6** Edit `onload.js` in `C:\theme\script` and add the following code at the bottom of the file. This code changes the theme so that the AD FS sign-in page does not require a domain name or an ampersand, "@", in the username.

```
// Update the placeholder text to not include the domain
var userNameInput = document.getElementById("userNameInput");
if (userNameInput) {
 userNameInput.setAttribute("placeholder", "Username");
}
```

```
// Override submitLoginRequest to not have the "@" check
Login.submitLoginRequest = function () {
 var u = new InputUtil();
 var e = new LoginErrors();
 var userName = document.getElementById(Login.userNameInput);
 var password = document.getElementById(Login.passwordInput);
 if (!userName.value) {
  u.setError(userName, e.userNameFormatError);
  return false;
 }
 if (!password.value) {
  u.setError(password, e.passwordEmpty);
  return false;
 }
 document.forms['loginForm'].submit();
 return false;
};
```

**Step 7** In Windows PowerShell, run the following commands to update the theme and make it active:

```
Set-AdfsWebTheme -TargetName custom -AdditionalFileResource
@{Uri='/adfs/portal/script/onload.js';path="c:\theme\script\onload.js"}

Set-AdfsWebConfig -ActiveThemeName custom
```

# Set the Principal AW for Single Sign On

**Note** This procedure is applicable only for Packaged CCE 4K or 12K agent reference design.

During deployment, the first SideA AW machine in the CSV file is the Principal AW.

The Principal AW (Admin Workstation) is responsible for managing background tasks that are run periodically to sync configuration with other solution components, such as SSO management, Smart Licensing, etc.

After deployment, you can change the Principal AW by selecting a different AW on the Inventory page. Set the AW on which you make most of your configuration changes as the Principal AW.

### Procedure

**Step 1** In Unified CCE Administration, choose **Inventory** to open the **Inventory** page.

**Step 2** Set the Principal AW:

a) Click the AW that you want to be the Principal AW.

**Note** You can only specify one Principal AW for each Unified CCE system.

The Edit CCE AW window opens.

b) Check the **PrincipalAW** check box.

c) Enter the Unified CCE Diagnostic Framework Service domain, username, and password.

d) Click **Save**.

# Set Up the System Inventory for Single Sign-On

Packaged CCE deployment automatically associates the Unified CCE AW, Unified Intelligence Center, and Finesse with a default Cisco Identity Service (Cisco IdS). However, if you have an external HDS in your deployment, you must manually associate it with a default Cisco IdS.

### Procedure

**Step 1** In **Unified CCE Administration**, click **Infrastructure** > **Inventory** to open the **Inventory** page.

**Step 2** Click the pencil icon for the External HDS to open the edit machine popup window.

**Step 3** Click the Search icon next to **Default Identity Service**.
The **Select Identity Service** popup window opens.

**Step 4** Enter the machine name for the Cisco IdS in the **Search** field or choose the Cisco IdS from the list.

**Step 5** Click **Save**.

**Note** If you are working with a Cisco IdS cluster, perform these steps on the Cisco IdS primary publisher node. For CCE 4000, 12000, and 24000 Agents deployment, ensure that the Principal AW is configured and functional before using the Single Sign-On tool in Unified CCE Administration. Also, add the SSO-capable machines to the Inventory, and select the default Cisco IdS for each of the SSO-capable machines.

# Configure the Cisco Identity Service

The Cisco Identity Service (Cisco IdS) provides authorization between the Identity Provider (IdP) and applications.

When you configure the Cisco IdS, you set up a metadata exchange between the Cisco IdS and the IdP. This exchange establishes a trust relationship that then allows applications to use the Cisco IdS for single sign-on. You establish the trust relationship by downloading a metadata file from the Cisco IdS and uploading it to the IdP. You can then select settings that are related to security, identify clients of the Cisco IdS service, and set log levels. If desired, enable Syslog format.

**Note**

- Unified CCE AW, Unified Intelligence Center, Finesse, and external HDS gets automatically associated with a default Cisco Identity Service (Cisco IdS).
- Make sure that the Principal AW is configured, and is functional before using the Single Sign-On tool in the Unified CCE Administration. Also, add the SSO-capable machines to the Inventory.

If you are working with a Cisco IdS cluster, perform these steps on the Cisco IdS primary publisher node.

### Procedure

**Step 1** In the Unified CCE Administration, choose **Overview** > **Infrastructure Settings** > **Device Configuration**.

**Note** Use a log in name in the format *username@FQDN* to log in to the Unified CCE Administration.

The **Identity Service Nodes**, **Identity Service Settings**, and **Identity Service Clients** tabs appear.

**Step 2** Click **Identity Service Nodes**.
You can view the overall Node level and identify which nodes are in service. You can also view the **SAML Certificate Expiry** details for each node, indicating when the certificate is due to expire. The node **Status** options are **Not Configured**, **In Service**, **Partial Service**, and **Out of Service**. Click a status to see more information. The star to the right of one of the Node names identifies the node that is the primary publisher.

**Step 3** Click **Identity Service Settings**.

**Step 4** Click **Security**.

**Step 5** Click **Tokens**.
Enter the duration for the following settings:

- **Refresh Token Expiry** -- Refresh token is used to get new Access tokens. This parameter specifies the duration after which the Refresh token expires. The default value is 10 hours. The minimum value is 2 hours. The maximum is 24 hours.
- **Authorization Code Expiry** -- Authorization code is used to get Access tokens from Cisco IdS. This parameter specifies the duration after which the Authorization code expires. The default value is 1 minute, which is also the minimum. The maximum is 10 minutes.
- **Access Token Expiry** -- Access token contains security credentials used to authorize clients for accessing resource server. This parameter specifies the duration after which the Access token expires. The default value is 60 minutes. The minimum value is 5 minutes. The maximum is 120 minutes.

**Step 6** Set the **Encrypt Token** (optional); the default setting is **On**. Use this configuration to secure the tokens as Cisco IdS issues tokens in both plain text or encrypted formats.

**Step 7** Click **Save**.

**Step 8** Click **Keys and Certificates**.
The **Generate Keys and SAML Certificate** page opens and allows you to:

- Regenerate the **Encryption/Signature key** by clicking **Regenerate**. A message appears to say that the Token Registration is successful and advises you to restart the system to complete the configuration. An Administrator regenerates the Encryption/Signature key when it is exposed or compromised.
- Regenerate the **SAML Certificate** by clicking **Regenerate**. A message appears to say that the SAML certificate regeneration is successful. SAML certificate is regenerated when it expires or when IdS relying party trust configuration on IdP is deleted.

  **Note** Establish the trust relationship again whenever the Encryption keys or SAML certificates are regenerated.

**Step 9** Click **Save**.

**Step 10** Click **Identity Service Clients**.
On the **Identity Service Clients** tab, you can view the existing Cisco IdS clients, with the client name, client ID, and a redirect URL. To search for a particular client, click the Search icon above the list of names and type the name of client.

**Step 11** To add a client on the **Identity Service Clients** tab:

a) Click **New**.
b) Enter the name of client.
c) Enter the Redirect URL. To add more than one URL, click the plus icon.
d) Click **Add** (or click **Clear** and then click the X to close the page without adding the client).

**Step 12** To edit or delete a client, highlight the client row and click the ellipses under **Actions**. Then:

- Click **Edit** to edit the client's name, ID, or redirect URL. On the **Edit Client** page, make changes and click **Save** (or click **Clear** and then click the X to close the page without saving edits).
- Click **Delete** to delete the client.

**Step 13** Click **Identity Service Settings**.

**Step 14** Click **Troubleshooting** to perform some optional troubleshooting.

**Step 15** From the **Log Level** drop-down list, set the local log level by choosing **Error**, **Warning**, **Info** (the default), **Debug**, or **Trace**.

**Step 16** To receive errors in Syslog format, enter the name of the Remote Syslog Server in the **Host** (Optional) field.

**Step 17** Click **Save**.

You can now:

- Register components with the Cisco IdS.
- Enable (or disable) SSO for the entire deployment.

**Note** If SSO is enabled in the deployment, then import all the IdS server nodes certificate into Cisco Finesse, CUIC, and LiveData component trust store.

# Install Certification Authority (CA) Certificate

To eliminate browser security warnings each time you sign in, obtain an application and root certificate signed by a CA. To obtain the application and a root certificate, use the Certificate Management utility from Cisco Unified Operating System Administration.

To open Cisco Unified Operating System Administration in your browser, enter: https://fqdn of IdS server:8443/cmplatform.

Sign in using the username and password for the Application User account created during IdS installation.

If you want to install CA signed certificates for IdS instead of out of the box self-signed certificates, you need to install the certificates on both the Cisco IdS nodes.

### Procedure

**Step 1** Log in to Cisco Unified Operating System Administration.

**Step 2** Navigate to **Security** > **Certificate Management** and then click **Find** to list all the certificates. The **Certificate List** page is displayed.

**Step 3** Click **Generate CSR**. The **Generate Certificate Signing Request** window is displayed.

**Step 4** Select **tomcat** from the **Certificate Purpose** list. By default, **tomcat** is selected.

**Step 5** Ensure that the default values are retained in the **Distribution**, **Common Name**, **Parent Domain**, **Key Type**, **Key Length**, and **Hash Algorithm** fields.

**Step 6** Click **Generate** to generate the CSR.

**Step 7** Click **Close**. The **Certificate List** page is displayed.

**Step 8** Click **Download CSR**. The **Download Certificate Signing Request** window is displayed.

**Step 9** Select **tomcat** from the **Certificate Purpose** list and click **Download CSR**.

**Step 10** Use the downloaded CSR and share it with the certificate authority to obtain the public certificate.

**Step 11** Log in to Cisco Unified Operating System Administration again and navigate to **Security** > **Certificate Management**, and then click **Find** to list all the certificates. The **Certificate List** page is displayed.

**Step 12** Click **Upload Certificate/Certificate chain**. The **Upload Certificate/Certificate chain** window is displayed.

**Step 13** Select **tomcat** from the **Certificate Purpose** list.

**Step 14** Click the **Choose File** button and navigate to select the certificate chain that includes the certificated obtained from the certiciate authority and then click **Open**.

**Step 15** Click **Upload** to upload the certificate.

**Step 16** Click **Close**. The **Certificate List** page is displayed.

**Step 17** After successfully uploading the certificate, navigate to **Security** > **Certificate Management**.

**Step 18** Click **Find** to open the list of certificates. The **Certificate List** page is displayed. Verify that the uploaded certificates are listed.

**Step 19** Restart the nodes using the CLI command *utils system restart*.

**Note**

- To avoid the certificate exception warning, you must access the servers using the Fully Qualified Domain Name (FQDN). Ensure that the Distribution field in the CSR is the FQDN of the server
- Ensure that the Certificate Authority (CA) certificate is RSA-signed.

# Register Components and Set Single Sign-On Mode

If you add any SSO-compatible machines to the System Inventory after you register components with the Cisco IdS, those machines are registered automatically.

### Before you begin

- Configure the Cisco Identity Service (Cisco IdS).
- Disable popup blockers. It enables viewing all test results correctly.

### Procedure

**Step 1** In the Unified CCE Administration, navigate to **Features** > **Single Sign-On**.

**Step 2** Click the **Register** button to register all SSO-compatible components with the Cisco IdS.

The component status table displays the registration status of each component.

If a component fails to register, correct the error and click **Retry**.

**Step 3** Click the **Test** button. When the new browser tab opens, you may be prompted to accept a certificate. In order for the page to load, accept any certificates. Then, when presented with a log in dialog, log in as a user with SSO credentials.

The test process verifies that each component has been configured correctly to reach the Identity Provider, and that the Cisco IdS successfully generates access tokens. Each component that you are setting up for SSO is tested.

The component status table displays the status of testing each component.

If a test is unsuccessful, correct the error, and then click **Test** again.

Test results are not saved. If you refresh the page, run the test again before enabling SSO.

**Step 4** Select the SSO mode for the system from the **Set Mode** drop-down menu:

- Non-SSO: This mode disables SSO for all agents and supervisors. Users log in using existing Active Directory-based local authentication.
- Hybrid: This mode allows you to enable agents and supervisors selectively for SSO.
- SSO: This mode enables SSO for all agents and supervisors.

The component status table displays the status of setting the SSO mode on each component.

If the SSO mode fails to be set on a component, correct the error, and then select the mode again.

# Hostname or IP Address Change

If you change the Hostname or IP Address of the Cisco IdS server, then perform the following:

- Re-generate the SAML certificate.
- Re-establish trust relationship between IdP and IdS.
- If the components are registered earlier, then
  - Re-register all the SSO components.
  - Perform the SSO Test to check if all the SSO components are registered. Verify that the test is successful for each component.

# Single Sign-On and the Agent Tool

When the global SSO-enabled setting is Hybrid, you can use the Unified CCE Administration Agent Tool to enable agents individually for single sign-on.

In the tool, check the **Single Sign-On** check box to require a selected agent to sign in with SSO authentication. For supervisors and for agents with single sign-on (SSO) enabled, the username is the user's Active Directory or SSO account username.

**Note** The check box is disabled when the global SSO mode is set to SSO or non-SSO.

To update agent records in bulk, use the Bulk Jobs Agent content file.

# Migration Considerations Before Enabling Single Sign-On

## Administrator User and Single Sign-On in Unified Intelligence Center

During installation, Cisco Unified Intelligence Center creates an administrator user. This user is not enabled for SSO, as the user is known only to Unified Intelligence Center.

When you enable SSO, this administrator user is no longer able to log in to the Unified Intelligence Center and perform administrative tasks. These tasks include configuring datasources and setting permissions for other users, for example. To avoid this situation, perform the following steps before enabling SSO.

1. Create a new SSO user who has the same roles and permissions as those of the administrator user.
2. Log in to the CLI.
3. Run the following command:

   **utils cuic user make-admin** *username*

in which the user name is the complete name of the new user, including the authenticator prefix as shown on the Unified Intelligence Center User List page.

The command, when performed, provides all the roles to the new user and copies all permissions from the administrator user to this new user.

**Note**

- The administrator's group memberships are not copied to the new user by this CLI command and must be manually updated. The new user, now a Security Administrator, can set up the group memberships.
- For any entity (for example, reports or report definitions), if this new user’s permissions provide higher privileges than the administrator, the privileges are left intact. The privileges are not overwritten by this CLI command.

# Browser Settings and Single Sign-On

If you have enabled single sign-on and are using Chrome, or Firefox, verify that the browser options are set as shown in the following table. These settings specify that you do not want a new session of the browser to reopen tabs from a previous session.

| Browser | Browser options to verify when using SSO |
|---|---|
| Chrome | 1. Open Chrome.<br>2. Click the **Customize and control Google Chrome** icon.<br>3. Click **Settings**.<br>4. In the **On startup** section of the **Settings** page, verify that the **Open the New Tab page** option is selected. |
| Firefox | 1. Open Firefox.<br>2. Click the **Open menu** icon.<br>3. Click **Options**.<br>4. In the **Startup** section of the **General** page, verify that either the home page or a blank page is chosen in the **When Firefox starts** drop-down list. |

# Migrate Agents and Supervisors to Single Sign-On Accounts

If you are enabling SSO in an existing deployment, you can set the SSO state to hybrid to support a mix of SSO and non-SSO users. In hybrid mode, you can enable agents and supervisors selectively for SSO making it possible for you to transition your system to SSO in phases.

Use the procedures in this section to migrate groups of agents and supervisors to SSO accounts using the SSO Migration content file in the Unified CCE Administration Bulk Jobs tool. You use the Administration Bulk Jobs tool to download a content file containing records for agents and supervisors who have not migrated to SSO accounts. You modify the content file locally to specify SSO usernames for the existing agents and supervisors. Using the Administration Bulk Jobs tool again, you upload the content file to update the agents and supervisors usernames; the users are also automatically enabled for SSO.

If you do not want to migrate a user, delete the row for that user.

**Important** While the Finesse agent is logged in, changing the login name prevents the agent from answering or placing calls. In this situation, the agent can still change between *ready* and *not_ready* state. This affects all active agents, independent of whether SSO is enabled or disabled. Should you need to modify a login name, do so only after the corresponding agent is logged out. Note too that SSO migration (moving a non-SSO agent to be SSO-enabled, by either hybrid mode or global SSO mode) should not be done when the agent is logged in.

### Procedure

**Step 1** In Unified CCE Administration, navigate to **Manage** > **Bulk Jobs**.

**Step 2** Download the SSO Migration bulk job content file.

a) Click **Templates.**

The **Download Templates** popup window opens.

b) Click the **Download** icon for the SSO Migration template.

c) Click **OK** to close the **Download Templates** popup window.

**Step 3** Enter the SSO usernames in the SSO Migration content file.

a) Open the template in Microsoft Excel. Update the **newUserName** field for the agents and supervisors whom you want to migrate to SSO accounts.

The content file for the SSO migration bulk job contains these fields:

| Field | Required? | Description |
|---|---|---|
| userName | Yes | The user's non-SSO username. |
| firstName | No | The user's first name. |
| lastName | No | The user's last name. |
| newUserName | No | The user's new SSO username. Enter up to 255 ASCII characters.<br>If you want to enable a user for SSO, but keep the current username, leave **newUserName** blank, or copy the value of **userName** into **newUserName**. |

b) Save the populated file locally.

**Step 4** Create a bulk job to update the usernames in the database.

a) Click **New** to open the **New Bulk Job** window.

b) Enter an optional **Description** for the job.

c) In the **Content File** field, browse to the SSO Migration content file you completed.

The content file is validated before the bulk job is created.

d) Click **Save**.

The new bulk job appears in the list of bulk jobs. Optionally, click the bulk job to review the details and status for the bulk job. You can also download the log file for a bulk job.

### What to do next

After all of the agents and supervisors in your deployment are migrated to SSO accounts, you can enable SSO globally in your deployment.

# Allowed Operations by Node Type

The Cisco IdS cluster contains a publisher and a subscriber node. A publisher node can perform any configuration and access token operations. The operations that a subscriber node can perform depends on whether the publisher is connected to the cluster.

This table lists which operations each type of node can perform.

***Table 7: Single Sign-On Allowed Operations***

| Operation | Allowed on Publisher | Allowed on Subscriber |
|---|---|---|
| Upload IdP metadata | Always | Never |
| Download SAML SP metadata | Always | Never |
| Regenerate SAML Certificate | Always | Never |
| Regenerate Token Encryption/Signing Key | Always | Never |
| Update AuthCode/Token Expiry | Always | Only when publisher is connected |
| Enable/Disable Token Encryption | Always | Only when publisher is connected |
| Add/Update/Delete Cisco IdS client configuration | Always | Only when publisher is connected |
| View Cisco IdS client configuration | Always | Always |
| View Cisco IdS status | Always | Always |
| Set Troubleshooting Log Level | Always | Always |
| Set Remote Syslog server | Always | Always |

# Configure SAML Certificate Secure Hash Algorithm from SHA-1 to SHA-256

This procedure is useful for upgrades from version 11.x where the only Secure Hash Algorithm supported was SHA-1.

.

Perform this procedure after the upgrade has completed successfully.

**Procedure**

**Step 1** From browser in AD FS Server, login to Cisco IdS admin interface `https://<Cisco IdS server address>:8553/idsadmin`.

**Step 2** Click **Settings**.

**Step 3** Click **Security** tab.

**Step 4** Click **Keys and Certificates**.

**Note** After this step, Single Sign On will stop working until you complete Step 8.

**Step 5** Regenerate SAML Certificate with SHA-256 Secure Hash Algorithm. In the SAML Certificate section, change Secure Hash algorithm dropdown menu to SHA-256 and then click **Regenerate** button

**Step 6** Download new metadata file. Click on **IdS Trust** tab and then click download button.

**Step 7** Change Secure Hash Algorithm in AD FS Relaying Party Trust configuration. In AD FS server, open AD FS Management. Go to **ADFS** ->**Trust Relationships**->**Relying Party Trusts**, right click on existing Relying Party Trust for Cisco IdS and then click on Properties. In the Advanced Tab, change the Secure Hash Algorithm to **SHA-256.** Click **Apply**.

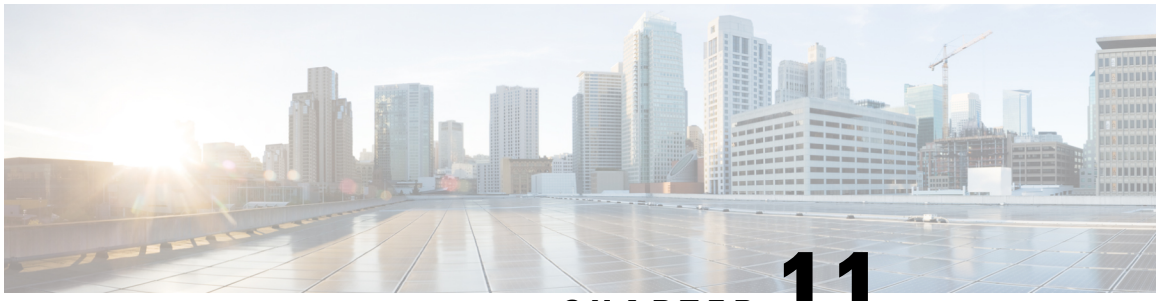
**Step 8** Update Relying party trust on AD FS. From AD FS Server, run the following Powershell command:

```
Update-AdfsRelyingPartyTrust -MetadataFile <path to Step 6 new MetaData File> -TargetName
        <Relying Party Trust Display Name>
```

# Single Sign-On Log Out

For a complete logout from all applications, sign out of the applications and close the browser window. In a Windows desktop, log out of the Windows account. In a Mac desktop, quit the browser application.

**Note** Users enabled for single sign-on are at risk of having their accounts misused by others if the browser is not closed completely. If the browser is left open, a different user can access the application from the browser page without entering credentials.



CHAPTER 11

# Task Routing



## Task Routing

Task Routing describes the system's ability to route requests from different media channels to any agents in a contact center.

You can configure agents to handle a combination of voice calls, emails, chats, and so on. For example, you can configure an agent as a member of skill groups or precision queues in three different Media Routing Domains (MRD) if the agent handles voice, e-mail, and chat. You can design routing scripts to send requests to these agents based on business rules, regardless of the media. Agents signed into multiple MRDs may switch media on a task-by-task basis.

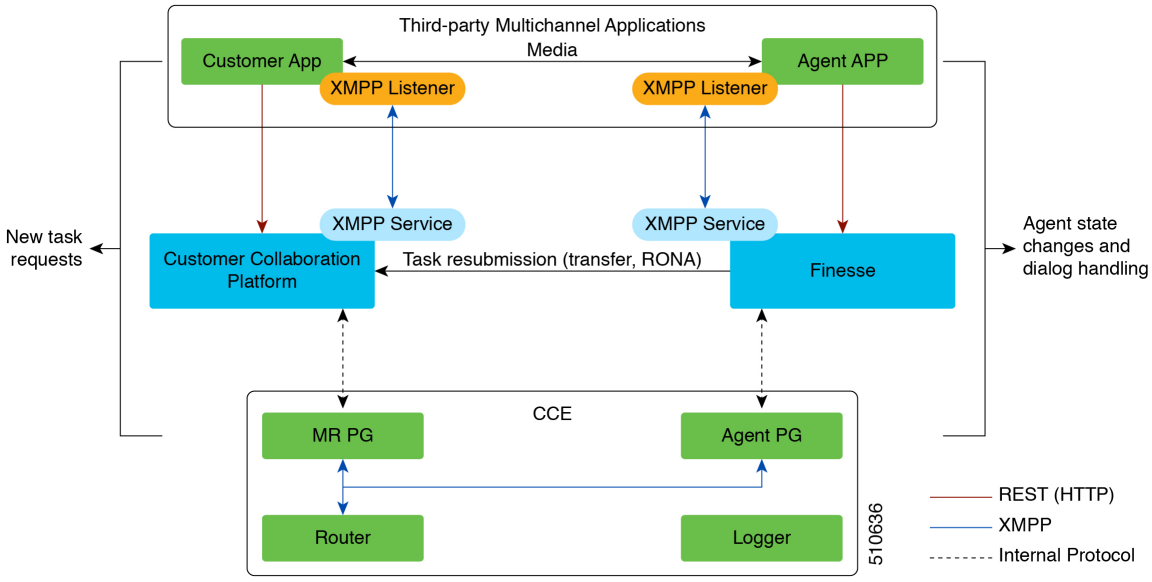
Enterprise Chat and Email provides universal queue out of the box. Third-party multichannel applications can use the universal queue by integrating with CCE through the Task Routing APIs.

Task Routing APIs provide a standard way to request, queue, route, and handle third-party multichannel tasks in CCE.

Contact Center customers or partners can develop applications using Customer Collaboration Platform and Finesse APIs in order to use Task Routing. The Customer Collaboration Platform Task API enables applications to submit nonvoice task requests to CCE. The Finesse APIs enable agents to sign into different types of media and handle the tasks. Agents sign into and manage their state in each media independently.

Cisco partners can use the sample code available on Cisco DevNet as a guide for building these applications (https://developer.cisco.com/site/task-routing/).

*Figure 15: Task Routing for Third-party Multichannel Applications Solution Components*




### Customer Collaboration Platform and Task Routing

Third-party multichannel applications use Customer Collaboration Platform'sTask API to submit nonvoice tasks to CCE.

The API works in conjunction with Customer Collaboration Platform task feeds, campaigns, and notifications to pass task requests to the contact center for routing.

The Task API supports the use of Call variables and ECC variables for task requests. Use these variables to send customer-specific information with the request, including attributes of the media such as the chat room URL or the email handle.

**Note** CCE solutions support only the Latin 1 character set for Expanded Call Context variables and Call variables when used with Finesse and Customer Collaboration Platform. Arrays are not supported.

### CCE and Task Routing

CCE provides the following functionality as part of Task Routing:

- Processes the task request.
- Provides estimated wait time for the task request.
- Notifies Customer Collaboration Platform when an agent has been selected.
- Routes the task request to an agent, using either skill group or precision queue based routing.
- Reports on contact center activity across media.

### Finesse and Task Routing

Finesse provides Task Routing functionality via the Media API and Dialog API.

With the Media API, agents using third-party multichannel applications can:

- Sign into different MRDs.
- Change state in different MRDs.

With the Dialog API, agents using third-party multichannel applications can handle tasks from different MRDs.

# Task Routing Deployment Requirements

Task Routing for third-party multichannel applications deployment requirements:

- Finesse and Customer Collaboration Platform are required. Install and configure Finesse and Customer Collaboration Platform before configuring the system for Task Routing.

  See the Finesse documentation and Customer Collaboration Platform documentation.

  By default, access to the Customer Collaboration Platform administration user interface is restricted. Administrator can provide access by unblocking the IP addresses of the clients. For more details, see the *Control Customer Collaboration Platform Application Access* topic in the *Cisco Customer Collaboration Platform Installation and Upgrade Guide* guide.

- You can install only one Customer Collaboration Platform machine in the deployment.
- Customer Collaboration Platform must be geographically colocated with one side of the Media Routing Peripheral Gateway (MR PG).
- Install Customer Collaboration Platform in a location from which CCE, Finesse, and the third-party multichannel Customer Collaboration Platform Task Routing application can access it over the network.

  If you install Customer Collaboration Platform in the DMZ, open a port for CCE and Finesse to connect to it. The default port for CCE to connect to Customer Collaboration Platform is port 38001. Finesse connects to Customer Collaboration Platform over HTTPS, port 443.

  Install the third-party multichannel application locally with Customer Collaboration Platform, or open a port on the Customer Collaboration Platform server for the application to connect to it.

### Related Topics

Control Customer Collaboration Platform Application Access, on page 160
utils permitlist admin_ui list, on page 161
utils permitlist admin_ui add, on page 161
utils permitlist admin_ui delete, on page 161

# Supported Functionality for Third-Party Multichannel Tasks

Blind transfer is supported for third-party multichannel tasks submitted through the Task Routing APIs.

We do not support the following functionality for these types of tasks:

- Agent-initiated tasks.

- Direct transfer.
- Consult and conference.

# Plan Task Routing Media Routing Domains

Media Routing Domains (MRDs) organize how requests for each communication medium, such as voice and email, are routed to agents. You configure an MRD for each media channel in your deployment.

Finesse agents can sign in to any of the multichannel MRDs you create for Task Routing.

Important factors to consider when planning your MRDs include the following:

- Whether the MRD is interactive.
- The maximum number of concurrent tasks that an agent can handle in an MRD.
- Whether the MRDs are interruptible.
- For interruptible MRDs, whether Finesse accepts or ignores interrupt events.

To configure the settings and parameters described in the following sections, see the following documents:

- Cisco Customer Collaboration Platform Developer Guide.
- Cisco Finesse Web Services Developer and JavaScript Guide
- Unified CCE Administration and Configuration Manager Tools, on page 178

### Interactive and Non-interactive MRDs

Interactive tasks are tasks in which an agent and customer communicate in real time with each other, such as chats and SMS messages. The customer usually engages with the agent through an application, like a chat window, and leaves this application open while waiting to be connected to an agent. Non-interactive tasks are asynchronous, such as email. The customer submits the request and then may close the application, checking later for a response from an agent.

| API Parameter or Setting | API/Tool | Possible Values | |
|---|---|---|---|
| | | Interactive Task/MRD | Non-interactive Task/MRD |
| **requeueOnRecovery**<br>Whether Customer Collaboration Platform re-queues or discards the task when Customer Collaboration Platform recovers from a failure.<br>Set this parameter when submitting a task request. | Customer Collaboration Platform Task Submission API | **False** - customers are waiting at an interface for an agent and can be notified if there is a problem. You don't need to resubmit these tasks. | **True** - customers are not waiting at an interface for an agent, and there is no way to alert them that there was a problem. You need to resubmit these tasks. |

| API Parameter or Setting | API/Tool | Possible Values | |
|---|---|---|---|
| | | Interactive Task/MRD | Non-interactive Task/MRD |
| **dialogLogoutAction**<br>Whether active tasks are closed or transferred when an agent signs out or loses presence.<br>Set this parameter when an agent signs in to a Media Routing Domain. | Finesse Media Sign In API | **Close**- customers are engaged with an agent, and can be notified that the task has ended. | **Transfer** - customers are not engaged with an agent, and there is no way to alert them that the task has ended. |
| **Start Timeout**<br>The amount of time that the system waits for an agent to accept an offered task. When this time is reached, the system makes the agent not routable and re-queues the task.<br>Set this parameter when configuring an MRD. | Media Routing Domains tool in Unified CCE Administration | **Shorter duration** - customer is waiting at an interface for the agent | **Longer duration** - customer is not waiting at an interface for an agent |
| **Monitoring status of submitted tasks**<br>You can monitor status of submitted and queued tasks using either the Customer Collaboration Platform Task API to poll for status or Customer Collaboration Platform XMPP BOSH eventing. | Customer Collaboration Platform Task API or XMPP BOSH eventing | Use Customer Collaboration Platform Task API status polling for MRDs when you want to monitor the status of a single contact/task. | Use Customer Collaboration Platform XMPP BOSH eventing to receive updates on all contacts/tasks in the campaign supporting Universal Queue over one channel. |

### Maximum Concurrent Tasks Per Agent

Specify the maximum number of concurrent tasks for an agent in an MRD when an agent signs into the Finesse application, using the **maxDialogLimit** parameter in the **Finesse Media - Sign In API**.

See the *Solution Design Guide for Cisco Unified Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-implementation-design-guides-list.html for the maximum number of tasks supported within an MRD and across MRDs for a single agent.

For agents handling interactive tasks, consider how many concurrent tasks an agent can handle reasonably. How many simultaneous chat sessions, for example, can an agent handle and provide good customer care? If you are using precision queue routing, keep in mind that CCE assigns tasks to agents who match attributes for step one, **up to their task limit**, until all of those agents are busy. CCE then assigns tasks to agents who match attributes for step two, up to their task limit, and so on.

### Interruptible and Non-Interruptible MRDs

When you create an MRD in the Unified CCE Administration Media Routing Domains tool, you select whether the MRD is interruptible.

- **Interruptible:** Agents handling tasks in the MRD can be interrupted by tasks from other MRDs. Non-interactive MRDs, such as an email MRD, are typically interruptible.
- **Non-interruptible:** Agents handling tasks in the MRD cannot be interrupted by tasks from other MRDs. The agents can be assigned tasks in the same MRD, up to their maximum task limits. For example, an agent can handle up to three non-interruptible chat tasks; if the agent is currently handling two chat tasks, CCE can assign the agent another chat, but cannot interrupt the agent with a voice call. Interactive MRDs, such as a chat MRD, are typically non-interruptible. Voice is non-interruptible.

When an agent is working on a non-interruptible task, CCE does not assign a task in any other MRD to the agent. Any application handling the non-voice MRDs must follow the same rule. In certain cases, it is possible that a task from another media routing domain gets assigned to an agent who is working on a non-interruptible task in an MRD.

For example, if an agent is working on a non-interruptible chat MRD and makes an outbound call (internal or external) using the desktop or phone, CCE cannot prevent the agent from making that call. Instead, the system handles this situation differently. CCE marks the agent temp not routable across all media domains until the agent has completed all non-interruptible tasks the agent is currently working on. Because of this designation, the agent is not assigned any new tasks from any MRDs until finishing all current tasks. Even if the agent tries to go ready or routable, the agent's temp not routable status is cleared only after all tasks are complete.

**Note** If you change the MRD from interruptible to non-interruptible or vice versa, the change takes effect once the agent logs out and then logs back in on that MRD.

### Accept and Ignore Interrupts

Specify whether an MRD accepts or ignores interrupt events when an agent signs into the Finesse application, using the **interruptAction** parameter in the **Finesse Media - Sign In API**. This setting controls the agent's state in an interrupted MRD and ability to work on interrupted tasks. The setting applies only when a task from a non-interruptible MRD interrupts the agent.

- **Accept:** When an agent is interrupted by a task from a non-interruptible MRD while working on a task in an interruptible MRD, Finesse accepts the interrupt event.

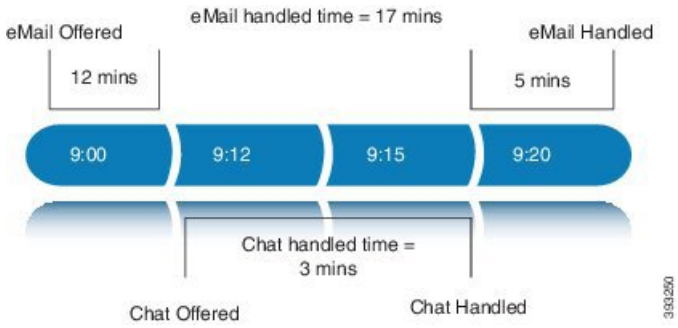
  The agent, CCE task, and Finesse dialog state in the interrupted MRD change to INTERRUPTED.

  The agent cannot perform dialog actions while a task is interrupted.

**Important** The application is responsible for disabling all dialog-related activities in the interface when an agent's state changes to INTERRUPTED.

  The agent's time on task stops while the agent is interrupted.

  Example: An agent has an email task for 20 minutes, and is interrupted for 3 of those minutes with a chat task. The handled time for the email task is 17 minutes, and the handled time for the chat task is 3 minutes.

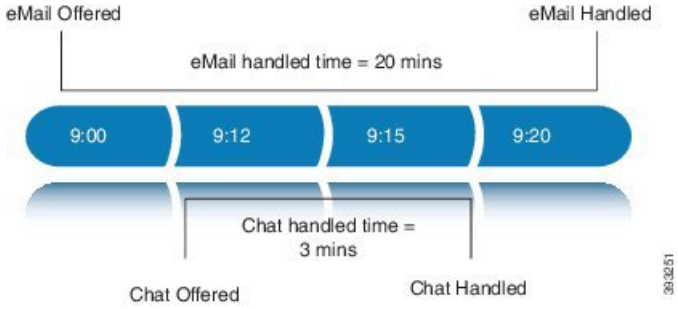



- **Ignore:** When an agent is interrupted by another task while working on a task in an interruptible MRD, Finesse ignores the interrupt event.

  The new task does not affect any of the agent's other assigned tasks. The agent, CCE task, and Finesse dialog state in the interrupted MRDs do not change.

  The agent can perform dialog actions on original task and the interrupting task at the same time. The agent's time on the original task does not stop while the agent is handling the interrupting task.

  Example: An agent has an email task for 20 minutes, and is interrupted for 3 of those minutes with a chat task. The handled time for the email task is 20 minutes, and the handled time for the chat task is 3 minutes. This means that during a 20-minute interval, the agent handled tasks for 23 minutes.




If an agent is working on a task in an interruptible MRD and is routed a task in another interruptible MRD, CCE does not send an interrupt event. Therefore, interruptAction setting does not apply.

# Plan Dialed Numbers

Dialed numbers, also called script selectors, are the strings or numbers submitted with Task Routing task requests through Customer Collaboration Platform. Each dialed number is associated with a call type, and determines which routing script CCE uses to route the request to an agent.

Dialed numbers are media-specific; you associate each one with a Media Routing Domain.

For Task Routing, plan which dialed numbers the custom Customer Collaboration Platform application will use when submitting new task requests. Consider whether you will use the same dialed numbers for transfer and tasks that are requeued on RONA, or if you need more dialed numbers.

**Important** You must associate each Task Routing dialed number with a call type. The default call type is not supported for Task Routing.

# Skill Group and Precision Queue Routing for Nonvoice Tasks

Routing to skill groups and precision queues is largely the same for voice calls and nonvoice tasks. However, the way that contact center enterprise distributes tasks has the following implications for agents who can handle multiple concurrent tasks:

- **Precision queues**—In precision queue routing, Unified CCE assigns tasks to agents in order of the precision queue steps. Unified CCE assigns tasks to agents who match the attributes for step one, up to their task limit, until all those agents are busy. Unified CCE then assigns tasks to agents who match attributes for step two, and so on. If you configure agents to handle three concurrent tasks, Unified CCE assigns three tasks to each agent in the first step. It then moves on to the second step and assigns any remaining tasks to those agents.
- **Overflow skill groups**—Routing scripts can specify a preferred skill group and an overflow skill group. Unified CCE assigns tasks to all agents in the preferred skill group, up to their task limit, before assigning any tasks in the overflow skill group. If you configure agents to handle three concurrent tasks, Unified CCE assigns three tasks to each agent in the preferred skill group. It then moves on to the overflow skill group and assigns any remaining tasks to those agents.

**Note** The number of available slots is an important factor in the Longest Available Agent (LAA) calculation.

The number of available slots = The maximum concurrent task limit for the MRD that an Agent has logged into - Current tasks being handled by the Agent or routed to the Agent.

If there are multiple skill groups that are part of the queue node, then the skill group that has the higher LAA is picked. Then, the agents within the picked skill group (or the Precision Queue) who have the highest number of available slots for non-voice tasks get prioritised.

Agents with the same number of available slots get prioritized based on the time in the available state or the LAA mechanism.

# Agent State and Agent Mode

An agent's state and routable mode in an MRD work together to determine whether CCE routes tasks to the agent in that MRD.

### Agent Routable Mode

The agent's routable mode controls whether CCE can assign the agent tasks in that MRD. If the agent is routable, CCE can assign tasks to the agent. If the agent is not routable, CCE cannot assign tasks to the agent.

The agent changes to routable/not routable through Finesse Media - Change Agent to Routable/Not Routable API calls.

### Agent State

The agent's state in an MRD indicates the agent's current status and whether the agent is available to handle a task:

- Ready: The agent is available to handle a task.
- Reserved/Active/Paused/Work Ready/Interrupted: The agent is available to handle a task if the agent has not reached their maximum task limit in the MRD.
- Not Ready: The agent is not available to handle a task.

The agent changes to Ready and Not Ready through calls to the Finesse Media - Change Agent State API. The agent's state while working on a task depends on the actions the agent performs on the Finesse dialog related to the task, through calls to the Finesse Dialog - Take Action on Participant API.

### How Mode and State Work Together to Determine if an Agent Receives Tasks

CCE will route an agent a task in the MRD if ALL of the following are true:

- The agent's mode is routable, and
- The agent is in any state other than NOT_READY, and
- The agent has not reached the maximum task limit in the MRD, and
- The agent is not working on a task in a different and non-interruptible MRD.

CCE will NOT route an agent a task in the MRD if ANY of the following are true:

- The agent's mode is not routable, or
- The agent is NOT_READY, or
- The agent has reached the maximum task limit in the MRD, or
- The agent is working on a task in a different and non-interruptible MRD.

### Why Change the Agent's Mode to Not Routable?

By changing the agent's mode to not routable, you stop sending tasks to the agent without changing the agent's state to Not Ready. You may want to make an agent not routable if the agent is close to ending the shift, and needs to complete in progress tasks before signing out.

If an agent changes to Not Ready state while still working on tasks, CCE reports show those tasks as ended; time spent working on the tasks after going Not Ready is not counted. By making the agent not routable instead of Not Ready, the agent's time on task continues to be counted.

In RONA situations, in which agents do not accept tasks within the Start Timeout threshold for the MRD, Finesse automatically makes agents not routable. Finesse resubmits the tasks through for routing through Customer Collaboration Platform. The application must make the agent routable in order for the agent to receive tasks again.

# Customer Collaboration Platform and Finesse Task States

In most cases, Customer Collaboration Platform social contact states do not map directly to Finesse dialog states. For Customer Collaboration Platform, social contacts are created when the customer submits a task request. For Finesse, the dialog with which the agent engages with the customer is created when the task is routed to the agent.

This table shows the relationships between Customer Collaboration Platform social contact task states and Finesse dialog states.

| Customer Collaboration Platform Social Contact Task State | Finesse Dialog State |
|---|---|
| **Unread:** The task request has not been submitted to the contact center. | None |
| **Queued:** The task request is successfully submitted to the contact center as a result of creating a new task or resubmitting a task due to agent transfer, automatic transfer on agent logout, or automatic transfer for RONA. | None |
| **Reserved:** The task is assigned to an agent. This state includes all work on a task. | **Offered:** The dialog is being offered to the agent. |
| | **Accepted:** The agent accepted the dialog but has not started working on it. |
| | **Active:** The agent is working on the dialog. |
| | **Paused:** The agent paused the dialog. |
| | **Wrapping Up:** The agent is performing wrap up activity on the dialog. |
| | **Interrupted:** The agent is interrupted with a task from a non-interruptible Media Routing Domain. The agent cannot work on this task until the interrupting task is complete. |
| **Handled:** Customer Collaboration Platform receives a handled notification from Finesse indicating that the task ended. | **Closed:** The agent ended the task. Finesse sends a handled notification to Customer Collaboration Platform. |

# Control Customer Collaboration Platform Application Access

By default, access to Customer Collaboration Platform administration user interface is restricted. Administrator can provide access by allowing clients IP addresses and revoke by removing the client's IP from the allowed list. For any modification to the allowed list to take effect, Cisco Tomcat must be restarted.

**Note** IP address range and subnet masks are not supported.

### Related Topics

Task Routing Deployment Requirements, on page 153

# utils permitlist admin_ui list

This command displays all the allowed IP addresses. This list is used to authorize the source of the incoming requests.

### Syntax

**utils admin_ui list**

### Example

```
admin: utils permitlist admin_ui list

Admin UI permitlist is:
10.232.20.31
10.232.20.32
10.232.20.33
10.232.20.34
```

### Related Topics

Task Routing Deployment Requirements, on page 153

# utils permitlist admin_ui add

This command adds the provided IP address to the allowed list of addresses.

### Syntax

**utils permitlist admin_ui add**

```
admin:utils whitelist admin_ui add 10.232.20.33

Successfully added IP: 10.232.20.33 to the whitelist permitlist

Restart Cisco Tomcat for the changes to take effect
```

### Related Topics

Task Routing Deployment Requirements, on page 153

# utils permitlist admin_ui delete

This command deletes the provided IP address from the allowed list.

### Syntax

**utils permitlist admin_ui delete**

### Example

```
admin:utils permitlist admin_ui delete 10.232.20.34

Successfully deleted IP: 10.232.20.34 from the permitlist

Restart Cisco Tomcat for the changes to take effect
```

### Related Topics

Task Routing Deployment Requirements, on page 153

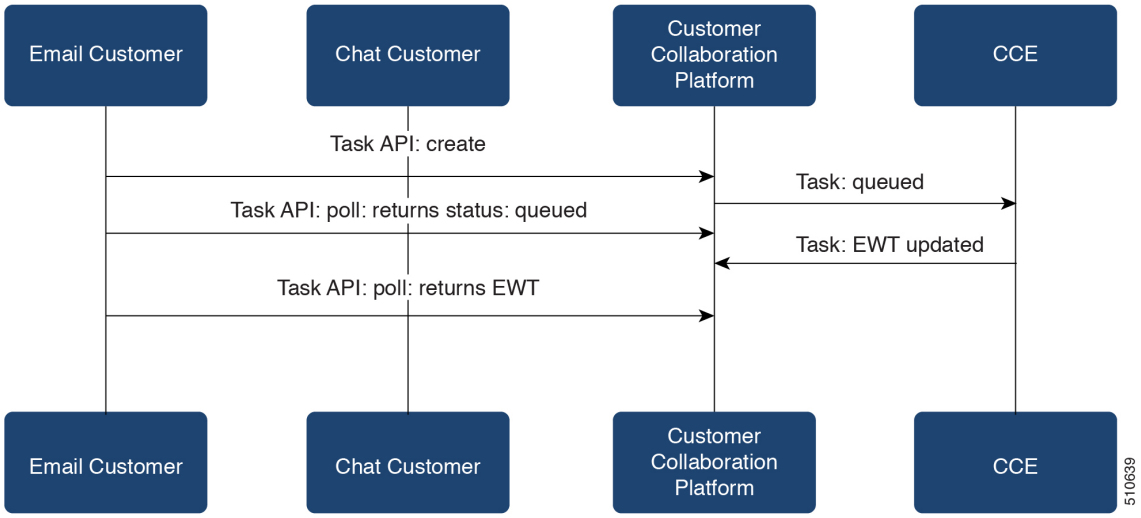
# Task Routing API Request Flows

## Task Routing API Basic Task Flow

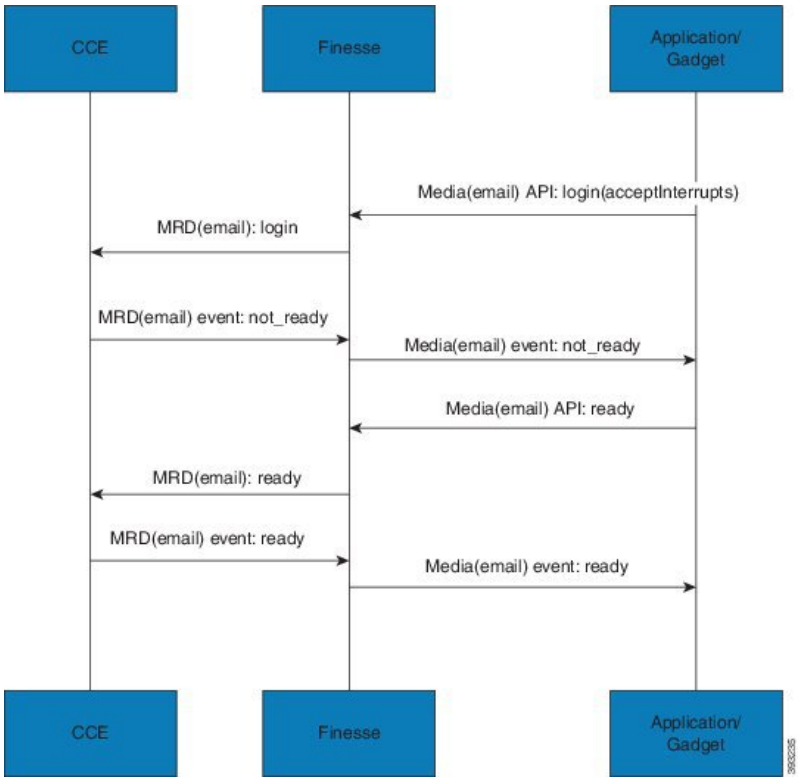
This topic provides the Customer Collaboration Platform and Finesse API calls and events when an active email task is interrupted by a chat request.

In this scenario, the email MRD is interruptible. When the agent signs into the email MRD, the application uses the Finesse Media API to accept interrupts. The chat MRD is non-interruptible.

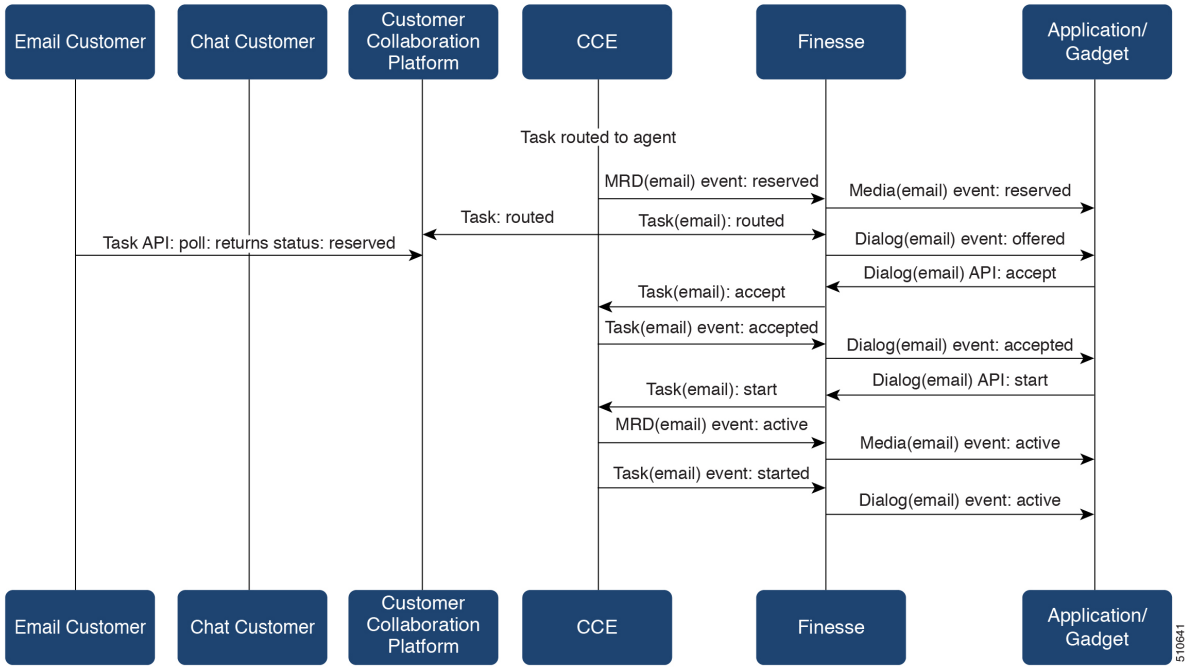
1. The email application submits a new email task request to CCE, and polls for status and Estimated Wait Time (EWT).



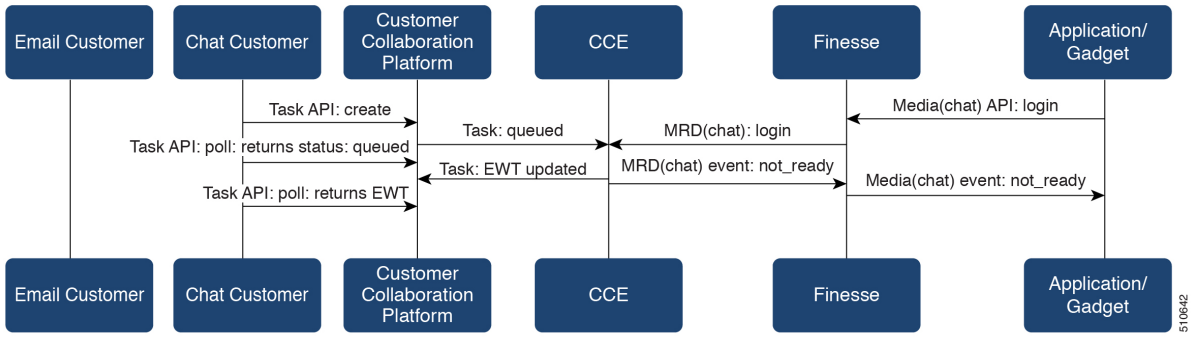

2. An agent signs in to the email MRD and changes state to Ready.



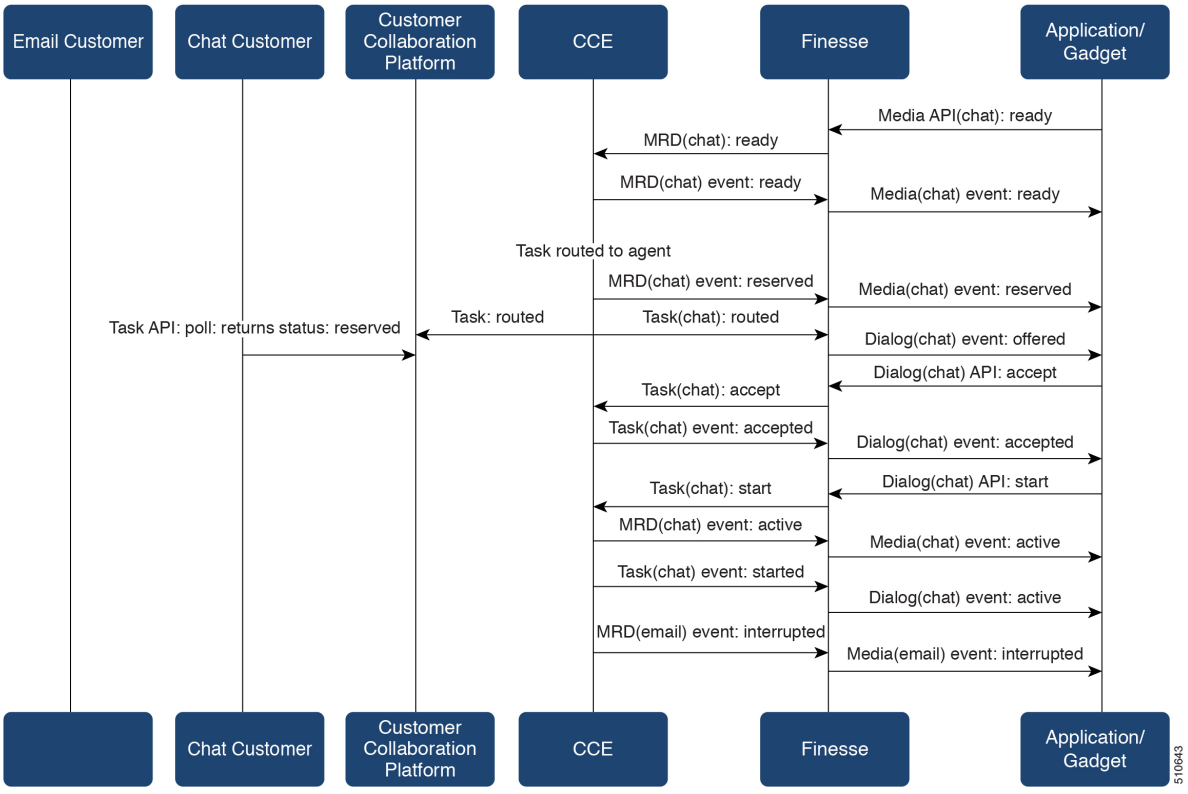

3. CCE assigns the agent the email task. The Call and ECC variables used to create the task are included in the dialog's media properties, and contain information such as the handle to the email. The variables can be used to reply to the email. The agent starts work on the email dialog in Finesse.



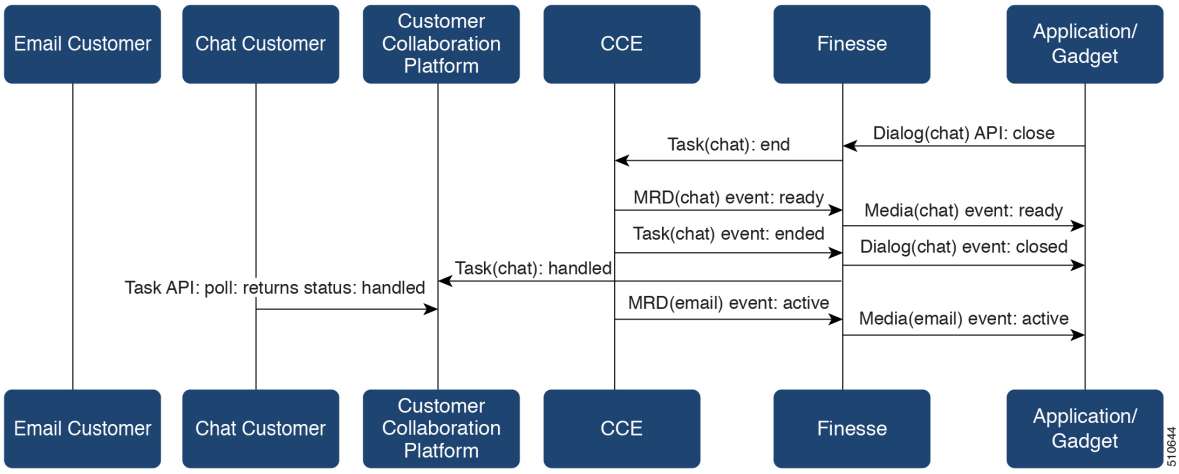
4. The chat application submits a new chat request, and polls for status and EWT. The same agent logs into the chat MRD.



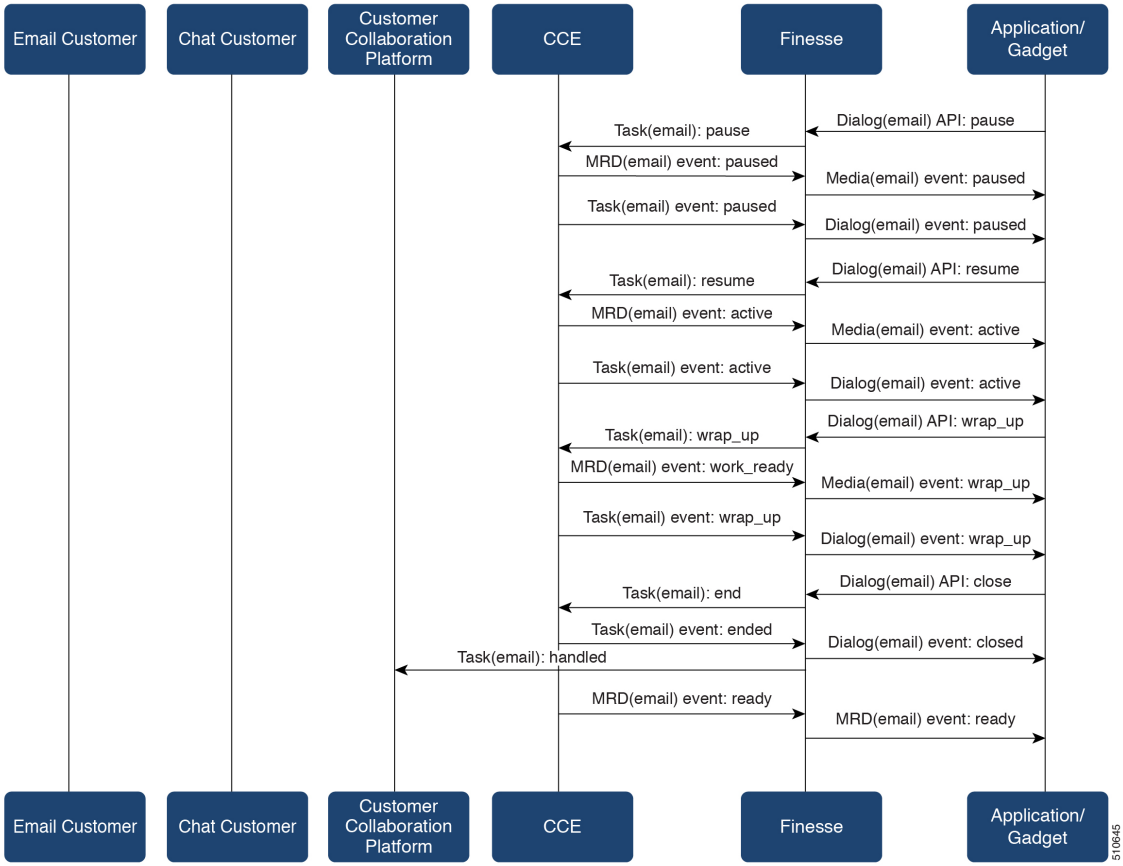

5. The agent changes state to Ready in the chat MRD. CCE assigns the chat task to the agent. The Call and ECC variables used to create the task are included in the dialog's media properties, and contain information such as the chat room URL. The variables can be used to join the chat room with the customer. The agent starts the chat dialog in Finesse. The Email dialog is interrupted.




6. The agent completes work on the chat dialog and closes the dialog. Finesse sends a handled event to Customer Collaboration Platform for the chat task. The application is responsible for closing the chat room. The agent is not handling other non-interruptible dialogs, and the email dialog becomes active.

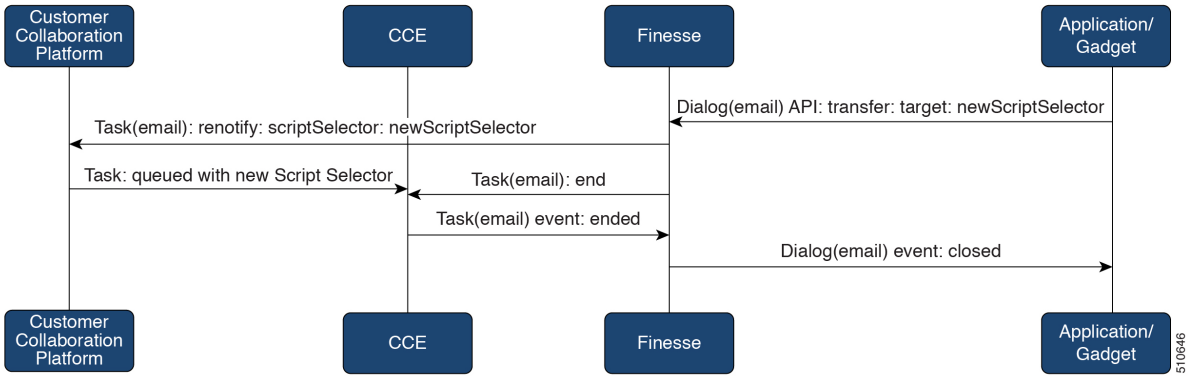



7. The agent continues working on the email dialog, including pausing, resuming, and wrapping up the dialog. The agent closes the dialog. Finesse sends a handle event to Customer Collaboration Platform for the email task. The application is responsible for sending the email reply to the customer.



# Task Routing API Agent Transfer Flow

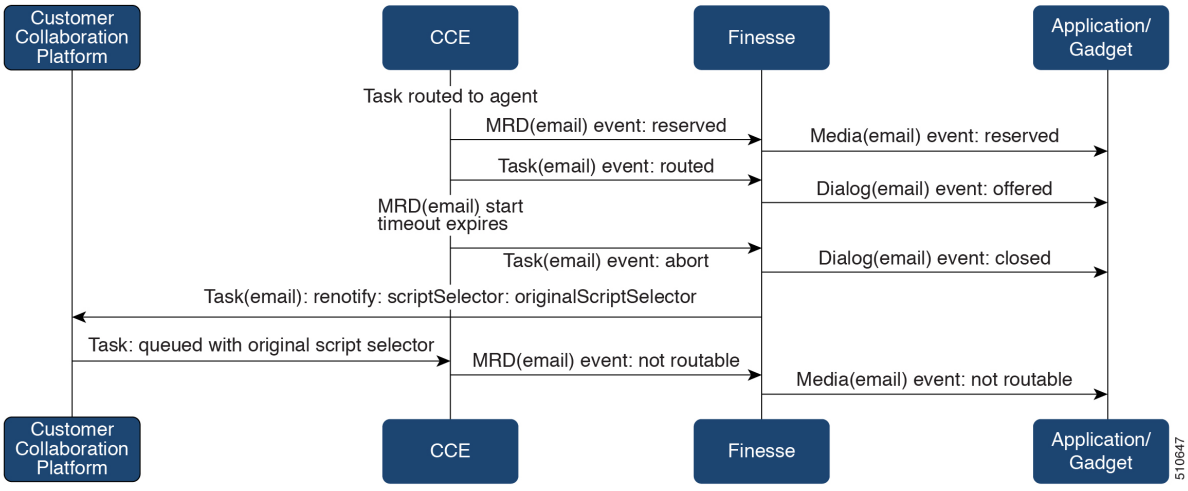
This illustration provides the Customer Collaboration Platform and Finesse API calls and events when an agent transfers a task.




1. The agent transfers the dialog from the Finesse application, selecting the script selector to which to transfer the task.
2. Finesse resubmits the task to Customer Collaboration Platform, and the task is queued to the script selector as a new task.
3. Finesse puts the original dialog in the CLOSED state, with the disposition code CD_TASK_TRANSFERRED. Finesse does not send a handled notification to Customer Collaboration Platform.

# Task Routing API RONA Flow

This illustration provides the Customer Collaboration Platform and Finesse API calls and events in a RONA scenario, in which an agent does not accept an offered task within the Start Timeout threshold for the MRD.




1. The task is routed to an agent, and the dialog is offered to the agent.

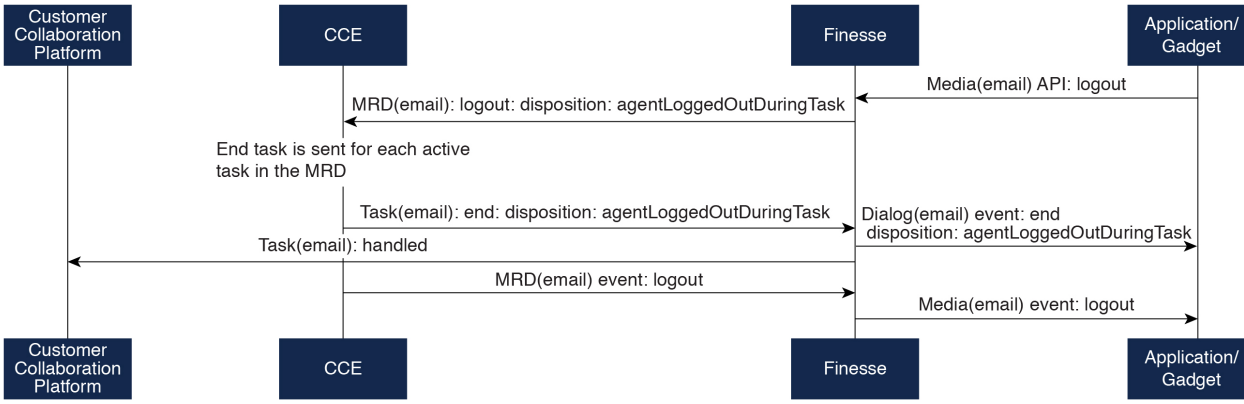
2. The Media Routing Domain's Start Timeout threshold expires.
3. CCE instructs Finesse to end the dialog. Finesse puts the dialog in the CLOSED state, with the disposition code CD_RING_NO_ANSWER. Finesse does not send a handled notification to Customer Collaboration Platform.
4. The Finesse server on which the agent was last signed in resubmits the task to Customer Collaboration Platform with the original script selector. The task is queued to the script selector as a new task.
5. CCE instructs Finesse to make the agent not routable in that Media Routing Domain, so that the agent is not routed more tasks.

# Task Routing API Agent Sign Out with Tasks Flows

The Finesse Media - Sign Out API allows agents to sign out with assigned tasks. The dialogLogoutAction parameter set by the Media - Sign In API determines whether those tasks are closed or transferred when the agent signs out.

### Close Tasks on Sign Out

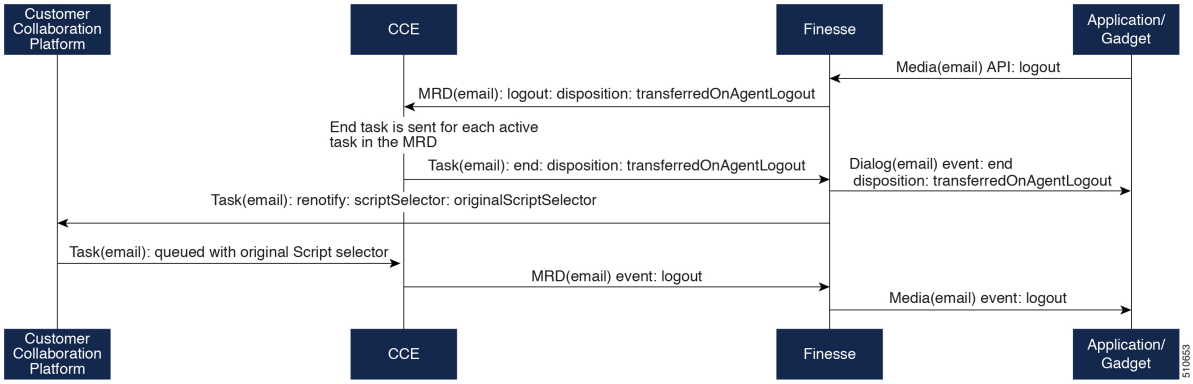
This illustration provides the Customer Collaboration Platform and Finesse API calls and events when agents are set to have assigned tasks closed on sign out.




1. The agent requests to sign out of the MRD with an active task.
2. CCE instructs Finesse to end the task. Finesse puts the dialog in CLOSED state, with the disposition code CD_AGENT_LOGGED_OUT_DURING_DIALOG.
3. The agent is signed out of the MRD.

### Transfer Tasks on Sign Out

This illustration provides the Customer Collaboration Platform and Finesse API calls and events when agents are set to have assigned tasks transferred on sign out.




1. The agent requests to sign out of the MRD with an active task.
2. CCE instructs Finesse to end the dialog. Finesse puts the dialog in the CLOSED state, with the disposition code CD_TASK_TRANSFERRED_ON_AGENT_LOGOUT. Finesse does not send a handled notification to Customer Collaboration Platform.
3. The Finesse server on which the agent was signed in resubmits the task to Customer Collaboration Platform with the original script selector. The task is queued to the script selector as a new task.
4. The agent is signed out of the MRD.

# Failover and Failure Recovery

| Component | Failover/Failure Scenario | New Task Request Impact | Queued, Offered, and Active Task Impact |
|---|---|---|---|
| Customer Collaboration Platform | MR connection fails. For example, there is a networking problem, the PG loses connection, or Customer Collaboration Platform loses connection.<br><br>Finesse loses connection with Customer Collaboration Platform. | **New task requests from Customer Collaboration Platform application**: New task requests fail, and the failures are delivered back to the application. Details of these failures are described in the next column.<br><br>**Automatic transfer request from Finesse** (for transfer on sign out or RONA): Results in a lost transfer request.<br><br>**Agent transfer request:** The request fails, and Finesse sends an error back to the application. Finesse retains the task. | **Queued tasks:** When tasks are submitted, they can be set to requeue on recovery. Typically, non-interactive tasks, such as email, are set to requeue on recovery because there is not a way to alert the customer that there was a problem while in queue. Interactive tasks, such as chat, are set not to requeue on recovery because the customer is waiting at an interface for an agent, and there is a way to alert the customer that there is a problem.<br><br>If tasks are set to requeue on recovery, the task is resubmitted when the MR connection is reestablished. The status and statusReason of the contact does not change.<br><br>If tasks are set NOT to requeue on recovery, the task's contact's status is marked discarded. The task's contact's statusReason is marked as follows:<br><br>**Customer Collaboration Platform failure:**<br><br>NOTIFICATION_CCE_<br><br>CUSTOMERCOLLABORATIONPLATFORM_SYSTEM_FAILURE<br><br>**MR connection failure:** NOTIFICATION_CCE_CONNECTION_LOST<br><br>**Offered and active tasks:** No impact. |
| Customer Collaboration Platform | Customer Collaboration Platform overruns the new task queue limit.<br><br>For the limit, see the Cisco Customer Collaboration Platform Developer Guide. | **New task requests from Customer Collaboration Platform application:** New task requests are discarded with the statusReason NOTIFICATION_RATE_LIMITED.<br><br>**Automatic or agent transfer requests:** No impact | **Queued, offered, and active tasks:** No impact. |

| Component | Failover/Failure Scenario | New Task Request Impact | Queued, Offered, and Active Task Impact |
|---|---|---|---|
| Finesse | Finesse loses connection with Agent PG or CTI Server | **New task request from Customer Collaboration Platform application:** No impact<br><br>**Automatic transfer requests from Finesse** (for transfer on logout or RONA): Automatic transfers are initiated on the Finesse server on which the agent was signed in. Any outage on that Finesse server can result in lost transfer requests.<br><br>**Agent transfer request:** The request fails because Finesse is out of service, and Finesse retains the task. | Agents signed into media on the failed Finesse server are put into WORK_NOT_READY state and made not routable. Tasks on that server are preserved in their current state, and time continues to accrue towards the maximum task lifetime. The agent fails over to the secondary Finesse server, and must sign in to the media again. The agent is put into the previous state. If the agent doesn't have tasks, the agent is put in NOT_READY state.<br><br>**Queued tasks:** No impact.<br><br>**Offered tasks:** These tasks RONA because the agent cannot accept them.<br><br>**Active tasks:** These tasks fail over to the other Finesse server and are recovered on that server.<br><br>**Note** Any active tasks that were in INTERRUPTED state at the time of the lost connection change are recovered. However, these tasks change to the UNKNOWN state when the task is no longer INTERRUPTED. The agent can only close tasks when they are in the UNKNOWN state. |

| Component | Failover/Failure Scenario | New Task Request Impact | Queued, Offered, and Active Task Impact |
|---|---|---|---|
| Finesse | Agent logs out, or presence is lost while agent has active tasks | **New task request from Customer Collaboration Platform application:** No impact<br><br>**Automatic or agent transfer requests:** No impact | **Queued tasks:** No impact.<br><br>**Offered tasks:** These tasks fail over to the other Finesse server and are recovered on that server. If a task's Start Timeout threshold is exceeded during failover, the task RONAs.<br><br>**Active tasks:** If an agent logs out with active tasks, or agent presence is lost with active tasks, the tasks are either closed or transferred to the original script selector depending on how the agent was configured when signing into the MRD.<br><br>If the tasks are transferred, the disposition code is CD_TASK_TRANSFERRED_AGENT_LOGOUT.<br><br>If the tasks are closed, the disposition code is CD_AGENT_LOGGED_OUT_DURING_DIALOG. |
| Finesse application | Finesse application fails | **New task request from Customer Collaboration Platform application:** No impact<br><br>**Automatic or agent transfer requests:** No impact | **Queued tasks:** No impact.<br><br>**Offered tasks:** These tasks may RONA depending on how the application is structured. A Task Routing application may prevent an agent from accepting a dialog when the application down because the agent cannot handle the dialog while the application is down. In this case, the dialog RONAs.<br><br>**Active tasks:** Varies by application. Applications are responsible for managing the tasks while the application is down. Finesse retains the tasks, and the tasks are recovered once the application is restored. |

| Component | Failover/Failure Scenario | New Task Request Impact | Queued, Offered, and Active Task Impact |
|---|---|---|---|
| CTI Server or OPC | One CTI Server or one OPC fails | **New task request from Customer Collaboration Platform application:** No impact<br>**Automatic transfer requests from Finesse** (for transfer on logout or RONA): Results in lost transfer requests.<br>**Agent transfer request:** The request fails, and Finesse retains the task. | **Queued tasks:** No impact.<br>**Offered tasks:** These tasks fail over to the other Finesse server and are recovered on that server. If a task's Start Timeout threshold is exceeded during failover, the task RONAs.<br>**Active tasks:** These tasks fail over to the other Finesse server and are recovered on that server.<br>**Note** Any active tasks that were in INTERRUPTED state at the time of the lost connection change are also recovered. However, these tasks change to the UNKNOWN state when the task is no longer INTERRUPTED. The agent only can only close tasks when they are in the UNKNOWN state. |
| OPC | Both OPCs fail | **New task request from Customer Collaboration Platform application:** No impact<br>**Automatic or agent transfer requests:** Results in lost transfers. | **Queued tasks:** No impact<br>**Offered and active tasks:** These tasks are lost |

# Task Routing Setup

## Initial Setup

| Step | Task | Notes |
|---|---|---|
| **Set up CCE** | | |
| 1 | Configure Finesse with the AW, so that Finesse can access Customer Collaboration Platform connection information.<br>See Configure Finesse with the AW, on page 174. | |

| Step | Task | Notes |
|---|---|---|
| 2 | Configure a Network VRU and Network VRU scripts.<br>See Configure Network VRU and Network VRU Scripts, on page 175. | |
| 3 | Configure the MR PG and PIM<br>See Configure the Media Routing PG and PIM, on page 175. | |
| 4 | Set up the MR PG and PIM for Customer Collaboration Platform.<br>See Set up the Media Routing PG and PIM, on page 176. | |
| 5 | Add Customer Collaboration Platform as an External Machine in the System Inventory.<br>See Add Customer Collaboration Platform as an External Machine, on page 177. | The system configures the following settings automatically in Customer Collaboration Platform Administration:<br>• Enables and configures the **CCE Multichannel Routing settings**.<br>• Configures the Task feed and the associated campaign and Connection to CCE notification needed for the Task Routing feature. |
| 6 | Configure the following in Unified CCE Administration or Configuration Manager:<br>• Media Routing Domains<br>• Call types<br>• Dialed numbers<br>• Skill groups or precision queues<br>• ECC variables<br>• Agent desk settings<br>See Unified CCE Administration and Configuration Manager Tools, on page 178. | |
| 7 | Increase the TCDTimeout registry key value, if you are using precision queues and will be submitting potentially long tasks, like email.<br>See Increase TCDTimeout Value, on page 179. | |
| 9 | Create routing scripts<br>See Create Routing Scripts for Task Routing, on page 180. | |

| Step | Task | Notes |
|---|---|---|
| **Create Customer Collaboration Platform and Finesse Applications** | | |
| 10 | Create the Customer Collaboration Platform multichannel application to begin task requests.<br>See Sample Customer Collaboration Platform HTML Task Application, on page 180. | |
| 11 | Create the Finesse applications to manage nonvoice agent and dialog states.<br>See Sample Finesse Code for Task Routing, on page 180. | |
| **Set up Finesse** | | |
| 12 | Upload the Finesse desktop gadgets to the desktop layout (optional).<br>See the *Cisco Finesse Administration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-user-guide-list.html. | |

# Configure Finesse with the AW

Finesse connects to Customer Collaboration Platform to transfer Task Routing tasks and resubmit tasks for RONA. The Finesse AWDB user requires special database permissions to access Customer Collaboration Platform connection information. Map the user to the Side A, AWDB, and primary databases. In these databases, give the user the db_datareader and public roles.

### Before you begin

Configure the Contact Center Administration and Data Server Connection Settings on Finesse. You need the Finesse AWDB username to complete this procedure.

See the *Cisco Finesse Administration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-user-guide-list.html.

### Procedure

**Step 1** Determine whether the Finesse AWDB user is a domain user or a SQL user. If the user is a domain user, proceed to the last step in the procedure (step 7). Otherwise, complete all of the steps.

**Step 2** Launch Microsoft SQL Server Management Studio on the Unified CCE Administration Client workstation.

**Step 3** Connect to the Side A Logger using the default credentials.

**Step 4** Navigate to **Security** > **Logins**. Right-click the Finesse AWDB username.
The Login Properties screen opens.

**Step 5** Select the **User Mapping** page, and perform the following:

a) Verify that the databases associated with Side A and AWdb are checked.

b) Check the primary database.

c) Select the Side A database. In the **Database role membership for** section, check the **db_datareader** and **public** roles.

Repeat this step for the AWdb and primary databases.

d) Click **OK**.

**Step 6** Repeat these steps on the Side B Logger.

**Step 7** Run the following SQL queries as the SQL administrative user "sa" or as a user with sysadmin privileges.

For <user>, enter the Finesse AWDB username. If the Finesse AWDB user is a domain user, rather than a SQL user, use the <domain\user> format.

```
USE master
GO
GRANT CONTROL ON CERTIFICATE :: UCCESymmetricKeyCertificate TO "<user>"
GRANT VIEW DEFINITION ON SYMMETRIC KEY :: UCCESymmetricKey TO "<user>"
```

# Configure Network VRU and Network VRU Scripts

The Network VRU is used to queue nonvoice tasks if an agent is not available to handle them. The Network VRU Script is used to return estimated wait time to customers. For more information on writing routing scripts that return estimated wait time, see the *Scripting and Media Routing Guide for Cisco Unified ICM/Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-user-guide-list.html.

When you configure the Network VRU Script, you specify whether it is interruptible. The **Interruptible** setting for the Network VRU Script controls whether the script can be interrupted (for example if an agent becomes available). This setting is not related to the Media Routing Domain **Interruptible** setting, which controls whether an agent working on a task in that MRD can be interrupted by a task from a non-interruptible MRD.

### Procedure

**Step 1** In Configuration Manager, use the **Network VRU Explorer** tool to configure and save a type 2 VRU.

**Step 2** Use the **Network VRU Script List** tool to add a Network VRU Script that references this Network VRU.

You can accept the default values.

# Configure the Media Routing PG and PIM

### Procedure

**Step 1** In Configuration Manager, open the PG Explorer tool to configure a media routing PG.

**Step 2** Create a media routing PIM and routing client for Customer Collaboration Platform.

Write down the Logical Controller ID and the Peripheral ID. You will use them when you set up the PG.

**Step 3** On the Peripheral tab in the PG Explorer tool, check the **Enable post routing** check box.

**Step 4** On the Routing Client tab in the PG Explorer tool, select the **Multichannel** option from the **Routing Type** drop-down list box.

**Note** The **Default call type** setting is not supported for tasks submitted through the Task Routing APIs.

**Step 5** On the Advanced tab in the PG Explorer tool, select the type 2 Network VRU that you created.

# Set up the Media Routing PG and PIM

**Caution** Before performing the step to enable the secured connection between the components, ensure that the security certificate management process is completed.

Set up the Media Routing PG and PIM

### Procedure

**Step 1** Navigate to **Unified CCE Administration** > **Overview** > **Infrastructure Settings** > **Peripheral Gateways**. Determine the Peripheral ID for a Multichannel peripheral that is unused.

**Step 2** From Cisco Unified CCE Tools, select **Peripheral Gateway Setup**.

**Step 3** On the Components Setup screen, in the Instance Components panel, select the PG Instance component. If the PG does not exist, click **Add**. If it exists, click **Edit**.

**Step 4** In the Peripheral Gateways Properties screen, click **Media Routing**. Click **Next**.

**Step 5** Click **Yes** at the prompt to stop the service.

**Step 6** From the Peripheral Gateway Component Properties screen, click **Add**, select the next PIM, and configure with the Client Type of Media Routing as follows.

a) Check **Enabled**.
b) In the **Peripheral Name** field, enter `MR`.
c) In the **Peripheral ID** field, enter the Peripheral ID that you recorded when you configured the Media Routing PG and PIM.
d) For **Application Hostname (1)**, enter the hostname or IP address of Customer Collaboration Platform.
e) By default, Customer Collaboration Platform accepts the MR connection on **Application Connection Port** 38001. The Application Connection Port setting on Customer Collaboration Platform must match the setting on the MR PG. If you change the port on one side of the connection, you must change it on the other side.
f) Leave the **Application Hostname (2)**, field blank.
g) Keep all other values.
h) Check the **Enable Secured Connection** option.

   This establishes a secured connectionbetween MR PIM and Application Server.

   Ensure that you provide the correct information in the Application Hostname (1) and Application Connection Port (1) fields.

i) Click **OK**.

**Step 7** On the Peripheral Gateway Component Properties screen, enter the Logical Controller ID that you recorded when you configured the Media Routing PG and PIM.

**Step 8** Accept defaults and click **Next** until the Setup Complete screen opens.

**Step 9** At the Setup Complete screen, check **Yes** to start the service. Click **Finish**.

**Step 10** Click **Exit Setup**.

**Step 11** Repeat this procedure for Side B.

# Add Customer Collaboration Platform as an External Machine

When you add Customer Collaboration Platform as an External Machine in the Unified CCE Administration System Inventory, the system automatically performs the following Customer Collaboration Platform configuration:

- Enables and completes the **CCE Configuration for Multichannel Routing** settings in Customer Collaboration Platform Administration.

  These settings include the hostnames of the MR PGs and the Application Connection Port you specified when setting up the MR PG and PIM.

- Configures the Task feed and the associated campaign and Connection to CCE notification needed for the Task Routing feature, with the following names:
  - **Task feed:** Cisco_Default_Task_Feed
  - **Campaign:** Cisco_Default_Task_Campaign
  - **Notification:** Cisco_Default_Task_Notification
  - **Tag:** cisco_task_tag

**Note** If the Task feed has been configured to use a different tag, the Connection to CCE notification is configured to use that tag.

### Procedure

**Step 1** In **Unified CCE Administration**, click **Inventory** from the left navigation.

**Step 2** Select the main site or remote site and in the **External Machines** section, click the + icon.

**Step 3** Click Add.

**Step 4** Select Customer Collaboration Platform from the drop-down list.

**Step 5** Enter the fully qualified domain name (FQDN), hostname or IP address in the **Hostname** field.

**Note** The system attempts to convert the value you enter to FQDN.

**Step 6** Enter the Customer Collaboration Platform Administration username and password.

**Step 7** Select the **Side A** and **Side B** Media Routing PGs.

**Step 8** Enter the Application Port you specified when setting up the MR PG and PIM. The default value is 38001.

**Step 9** Click **Save**.

# Unified CCE Administration and Configuration Manager Tools

This topic explains the Unified CCE Administration and Configuration Manager tools you need to configure Task Routing.

### Before you begin

For details on the procedures for these steps, refer to the Unified CCE Administration online help and the Configuration Manager online help.

### Procedure

**Step 1** Sign in to Unified CCE Administration.

**Step 2** From the **Manage** menu, configure the following:

| Item to Configure | Details |
|---|---|
| Media Routing Domains | Create an MRD for each type of task that the third-party multichannel application submits to CCE (email, chat, and so on). |
| Precision Queues | Configure either skill groups or precision queues.<br>If you configure precision queues:<br>• For **Media Routing Domain**, select one of the Task Routing MRDs you created.<br>• Associate agents with attributes that are part of the precision queue steps. |

**Step 3** Launch Configuration Manager.

**Step 4** Configure the following:

| Item to Configure | Details |
|---|---|
| Call Types | Create call types for Task Routing. |
| Dialed Numbers | Create dialed numbers for Task Routing. Add the numbers or strings that the custom application will use when submitting task requests.<br>• On the **Attributes** tab, select a Task Routing MRD from the **Media routing domain** drop-down list box.<br>• On the **Dialed Number Mapping** tab, map the script selector to a call type you created for Task Routing.<br>**Important** Each dialed number must be associated with a call type. Default call type is not supported for tasks submitted with Task Routing APIs. |

| Item to Configure | Details |
|---|---|
| Skill Groups | Configure either skill groups or precision queues.<br>If you configure skill groups:<br>• For **Media Routing Domain**, select one of the Task Routing MRDs you created.<br>• Assign agents to the skill group. |
| Expanded Call Variable | You can use an existing Expanded Call Variable, or you can create an Expanded Call Variable for Task Routing, depending on the needs of your third-party multichannel application.<br>**Note** Arrays are not supported with the Task Routing feature.<br>CCE solutions support the Latin 1 character set only for Expanded Call Context variables and Call variables when used with Finesse and Customer Collaboration Platform. |
| Agent Desk Settings | If agents will use a Task Routing gadget in the Finesse desktop, leave the **Logout inactivity time** setting for those agents blank, or remove the existing value.<br>Otherwise, if the agent exceeds the **Logout inactivity time** in the voice MRD, the agent is logged out of the Cisco Finesse desktop, even if the agent is actively working on tasks from nonvoice MRDs. The agent needs log into the desktop again to continue working on the nonvoice tasks. |

# Increase TCDTimeout Value

Complete this procedure only if you are using precision queues and routing tasks with potentially long durations, like emails.

Several precision queue fields in the Termination_Call_Detail record are not completed until the end of a task. These precision queue fields are blank for tasks whose durations exceed the TCDTimeout registry key value. The default value of theTCDTimeout registry key is 9,000 seconds (2.5 hours).

If you are configuring a system to handle email or other long tasks, you can increase the TCDTimeout registry key value to a maximum of 86,400 seconds (24 hours).

Change the registry key on either the Side A or B Router.

### Procedure

Modify the following registry key: `HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\Icm\<instance name>\Router<A/B>\Router\CurrentVersion\Configuration\Global\TCDTimeout`.

# Create Routing Scripts for Task Routing

For complete multichannel scripting information, see the *Scripting and Media Routing Guide for Cisco Unified ICM/Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-user-guide-list.html.

**Important** Ensure that the routing scripts include skill groups or precision queues from the appropriate Media Routing Domains to handle all of the types of tasks that can be routed with the scripts. For example, if a script is used to route email tasks, be sure that the script includes skill groups or precision queues from an email MRD.

# Sample Code for Task Routing

Cisco Systems has made sample Task Routing application code for Customer Collaboration Platform and Finesse available to use as baselines in building your own applications.

## Sample Customer Collaboration Platform HTML Task Application

The sample Customer Collaboration Platform HTML Task application:

- Submits task requests to CCE.
- Retrieves and displays the estimated wait time, if it has been configured in CCE.

**Note** You cannot copy and paste this code to achieve a working application. It is only a guideline.

The sample application uses the Task API. For more information about how to use the Task API, see the Cisco Customer Collaboration Platform Developer Guide.

### Procedure

**Step 1** Download the sample HTML Task application from DevNet: https://developer.cisco.com/site/task-routing/.

**Step 2** Read the sample application's **readme.txt** file to complete the prerequisites and use the sample application.

## Sample Finesse Code for Task Routing

The Finesse sample Task Management Gadget application lets agents perform the following actions in individual nonvoice Media Routing Domains:

- Sign in and out.
- Change state.
- Handle tasks.

The sample gadget also signals the Customer Context gadget to display a customer record.

**Note** You cannot copy and paste this code to achieve a working application. It is only a guideline.

For more information about how to use the APIs available for Task Routing, see the *Cisco Finesse Web Services Developer Guide* at https://developer.cisco.com/site/finesse/.

**Procedure**

**Step 1** Download the sample Task Management Gadget application (TaskManagementGadget-x.x.zip) from DevNet: https://developer.cisco.com/site/task-routing/.

**Step 2** Read the sample application's **readme.txt** file to complete the prerequisites and use the sample application.

For more information about uploading third-party gadgets to the Finesse server, see the "Third Party Gadgets" chapter in the *Cisco Finesse Web Services Developer Guide* at https://developer.cisco.com/site/finesse/.

For more information about adding third-party gadgets to the Finesse desktop, see the "Manage Third-Party Gadgets" chapter in the *Cisco Finesse Administration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/tsd-products-support-series-home.html.

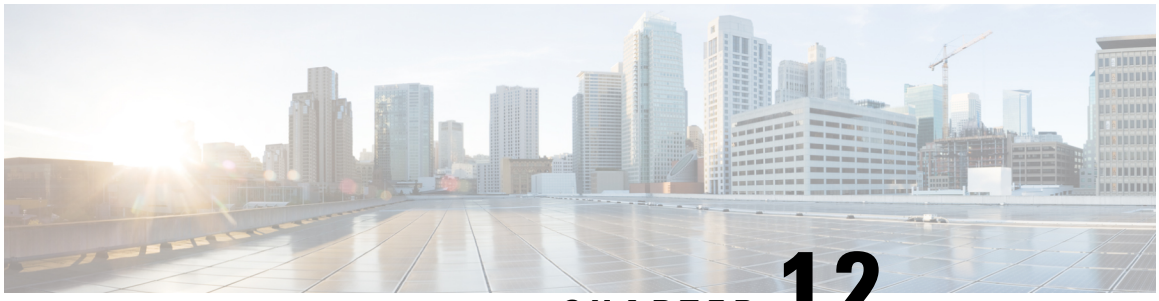
# Task Routing Reporting

Cisco Unified Intelligence Center CCE reports include data for voice calls and nonvoice Task Routing tasks.

You can filter these All Fields and Live Data report templates by Media Routing Domain:

- Agent Real Time
- Agent Skill Group Real Time
- Enterprise Skill Group Real Time
- Peripheral Skill Group Real Time All Fields
- Precision Queue Real Time All Fields
- Agent Precision Queue Historical All Fields
- Agent Skill Group Historical All Fields
- Peripheral Skill Group Historical All Fields
- Precision Queue Abandon Answer Distribution Historical
- Precision Queue Interval All Fields
- Skill Group Abandon-Answer Distribution Historical
- Precision Queue - Live Data
- Skill Group - Live Data

See the *Reporting Concepts for Cisco Unified ICM/Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-user-guide-list.html for information about multichannel reporting data.



CHAPTER 12

# Unified Communications Manager Extension Mobility



## Capabilities

Extension Mobility is a Unified Communications Manager feature that you can use in Unified CCE. The feature enables users to temporarily configure a phone as their own by logging in to that phone. Once a user logs in, the phone adopts the individual user device profile information, including line numbers, speed dials, services links, and other user-specific properties of a phone.

Cisco Extension Mobility (EM) works on phones that are located within the same Cisco Unified Communications Manager cluster. Cisco Extension Mobility Cross Cluster (EMCC) works on phones that are located in different Cisco Unified Communications Manager clusters.

The main documentation on this feature is in the Unified Communications Manager documentation. For more information, see the following sources:

| Information Type | Sources |
|---|---|
| Design considerations | *Cisco Collaboration System Solution Reference Network Designs* at https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/products-implementation-design-guides-list.html |
| Feature description and configuration | *Feature Configuration Guide for Cisco Unified Communications Manager* at https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/products-maintenance-guides-list.html |
| Extension Mobility API | https://developer.cisco.com/site/extension-mobility/ |

# Configuration

You configure EM and EMCC in the Cisco Unified Communications Manager. Take into account the following interactions between Unified CCE and Unified Communications Manager for successful implementation of EM and EMCC within a Unified CCE solution:

- For Unified CCE configurations with multiline agent phone line control on the PG, configure all directory numbers for the user profile in Cisco Unified Communications Manager as follows:

| Setting | Value |
|---|---|
| Maximum Number of Calls | 2 |
| Busy Trigger | 1 |

- For Unified CCE configurations with single-line agent phone line control on the PG, configure the secondary lines (but not the primary ACD line) for the directory number of the user profile in Cisco Unified Communications Manager as follows:

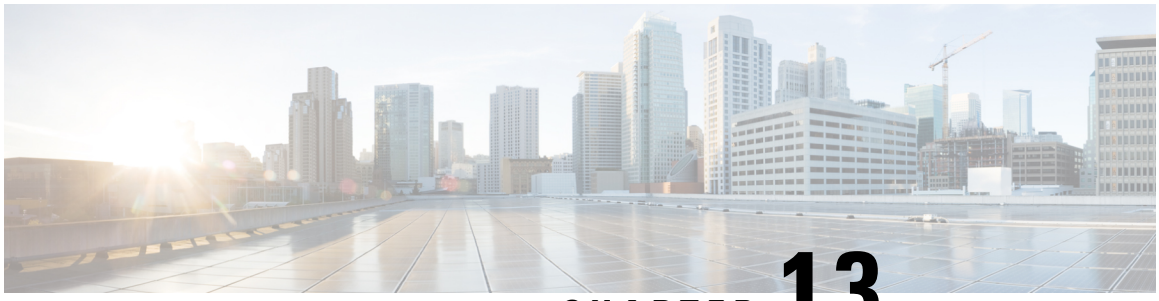
| Setting | Value |
|---|---|
| Maximum Number of Calls | 4 |
| Busy Trigger | 2 |

- You cannot use phones with an **IP Addressing Mode** of **IPv6 Only** for Cisco Extension Mobility. If you want to use Cisco Extension Mobility with the phone, you must configure the phone with an **IP Addressing Mode** of **IPv4 Only** or **IPv4 and IPv6**.
- Agents can log in to multiple devices, depending on the **Intra-cluster Multiple Login Behavior** service parameter. You can set this parameter for EM implementations. EMCC implementations require that you set this parameter for multiple logins.

  If an agent fails to log out of a device, another agent who attempts to access that device gets a "shared line" error. Follow these Unified Communications Manager configuration guidelines to avoid shared line errors:

  - For EM implementations with hard phones, set the **Intra-cluster Multiple Login Behavior** Extension Mobility service parameter to "Auto Logout".
  - For EM implementations with a mix of hard and IP phones and for all EMCC implementations, limit the time that an agent can remain logged in to a device. Set the **Intra-cluster Maximum LoginTime** service parameter to the typical time that an agent remains logged in to a device during a shift.

For more information on Extension Mobility with Unified CCE, see *UCCE Integration with CM Configuration Example* at https://www.cisco.com/c/en/us/support/docs/customer-collaboration/unified-contact-center-enterprise/117777-config-ucce-00.html.



CHAPTER 13

# Virtual Agent–Voice



## Feature Overview

Virtual Agent–Voice (VAV) feature, which was referred to as Customer Virtual Assistant (CVA) in 12.5(1) release, enables the IVR platform to integrate with cloud-based speech services. This feature supports human-like interactions that enable customers to resolve issues quickly and more efficiently within the IVR, thereby, reducing the calls directed towards agents.

**Note** The Dialogflow ES GCP project trial version should not be used in a production environment.

In a traditional IVR, customers can interact with the IVR in the following ways:

- **VVB Media Services-Based Interaction**: Prompts are played locally by VVB by downloading WAV files. User inputs are captured using DTMF grammar.
- **ASR-Based and TTS-Based Interactions**: Prompts are played by the external media server over MRCP Synthesis command for Text-to-Speech (TTS) functionality. The responses are recognized by external media server based on predefined grammar provided by Asynchronous Speech Recognition (ASR).

VAV-based IVR enables a new mechanism to leverage cloud-based-AI-enabled speech services. VAV provides the following speech services:

- **Text-to-Speech**: Integration with cloud-based TTS services in your application for Speech Synthesis operations. VAV currently supports Google Text to Speech service.
- **Speech-to-Text**: Integration with cloud-based ASR services in your application for Speech Recognition operations. VAV currently supports Google Speech to Text service.
- **Speech-to-Intent**: VAV provides capability of identifying the intent of customer utterances by processing the text received from Speech-to-Text operations. VAV offers this service by using cloud-based Natural Language Understanding (NLU) services. VAV currently supports Google Dialogflow service.

# Onboarding Experience

In 12.6(1) release, the VAV feature provides different onboarding experience for OEM users (who use Cisco's contract, billing, and support for Google's speech services) via Webex Control Hub. For details, see the Create a Contact Center AI configuration article. Non-OEM users use NOAMP in Unified CCE solution and CCE Administration Portal in Packaged CCE solution for onboarding.

The following table lists the onboarding channel and Google APIs used for OEM and non-OEM customers:

| Release | Services Billing | Onboarding | Google APIs |
|---|---|---|---|
| 12.6(1) | Cisco billed (OEM) | Webex Control Hub | AnalyzeContent (V2) |
| 12.6(1) | Vendor billed (non-OEM) | NOAMP/CCEAdmin | DetectIntent (V2) |

# VAV Onboarding for OEM Users

Virtual Agent–Voice (VAV) feature provides an enhanced onboarding experience to OEM customers via Webex Control Hub. All contract, billing, and support are managed through Cisco for OEM customers and they can use Cisco services coupled with Google's cloud-based-AI-enabled speech services.

A single config ID generated via Control Hub can be leveraged across all CVP/VVB instances as compared to the earlier experience where each instance was required to be configured individually.

## Prerequisites

The prerequisites for configuring Virtual Agent–Voice for OEM users are:

- OEM users must provision Google Contact Center AI (CCAI) for Cisco Contact Center Enterprise. For details, see the *Create a Contact Center AI configuration* article.
- CVP/VVB configuration:
  - Enable access to cloud-based services from CVP and VVB directly or via proxy.

    For details, see the *Cisco Unified Customer Voice Portal > Operations Console (NOAMP) > Integration > Cloud Connect > Configure CVP or VVB Devices for Cloud Connect* section in the *Administration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.
  - Synchronize the date/time in CVP, VVB, and proxy with a common NTP server.
  - Configure access to DNS server in CVP/VVB.

For more information on NTP and DNS server configurations in CVP, refer to the Microsoft Windows platform documentation.

For more information on NTP and DNS server configurations in VVB, refer to the *Command Line Interface Reference Guide for Cisco Unified Communications Solutions* at https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/products-maintenance-guides-list.html.

- The following table lists the required versions for Cisco Unified CVP, Cisco VVB, Cloud Connect, CCE components (Router, Logger, AW, and PG), and ICM.

| CVP | VVB | Cloud Connect | CCE Components | ICM |
|---|---|---|---|---|
| 12.6 | 12.6 | 12.6 | 12.0 and higher | 12.0 and higher |

# VAV Onboarding for OEM Users Task Flow

Follow this procedure to provision Google CCAI with Cisco Unified Contact Center Enterprise. A single configuration created via Control Hub can be used for accessing multiple AI services, such as VAV and Agent Answers on multiple devices.

### Procedure

**Step 1** Create a CCAI configuration in Cisco Webex Control Hub at https://admin.webex.com. A CCAI configuration leverages CCAI Connectors to invoke the CCAI services.

For details, see the *Create a Contact Center AI configuration* article. This default config can be used for accessing multiple AI services on multiple devices.

**Note** While creating a CCAI configuration (explained in the article *Create a Contact Center AI configuration*):

- Skip the creation of conversation profile.
- Select the **Apply as default for Virtual Agent** configuration, because it is mandatory for ASR and TTS.

**Step 2** Ensure that the Cloud Connect publisher and subscriber are installed.

For more information, see the *Install Cloud Connect* section in *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-guides-list.html.

**Step 3** Import the Cloud Connect certificate to the CVP server.

For details, see the *Unified CVP Security > Import Cloud Connect Certificate to Unified CVP Keystore* section in the *Configuration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

**Step 4** Configure Cloud Connect with CVP in the Operations Console (NOAMP).

**Step 5** Configure Cloud Connect with VVBs in the Operations Console (NOAMP).

For details, see the *Cisco Unified Customer Voice Portal > Operations Console (NOAMP) > Integration > Cloud Connect > Configure CVP or VVB Devices for Cloud Connect* section in the *Administration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

**Step 6** View the default CCAI configuration (created in step 1). If required, synchronize the configuration (using *Sync* option) in the CVP Operations Console (NOAMP).

For details, see the *Cisco Unified Customer Voice Portal > Operations Console (NOAMP) > Contact Center AI > Configuration for Cisco-Billed AI Services* section in the *Administration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

**Note** To change or override the default config (created in step 1), configure the `CCAI.configId` property of the `Dialogflow` element in Call Studio.

For details, see the *Dialogflow Element > Custom VoiceXML Properties* section in the *Element Specifications for Cisco Unified CVP VXML Server and Call Studio* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-programming-reference-guides-list.html.

## Migration for OEM Users

OEM users migrating from 12.5(1) or 12.5(1a) to 12.6(1) must onboard via Webex Control Hub for better experience and enhanced security. They can continue to use the old method of key generation by retaining their old configurations until their onboarding via Webex Control Hub is complete.

## Important Considerations

VVB periodically refreshes the CCAI configurations. Whenever these configurations are changed in the Control Hub, it may take upto 10 minutes for the changes to reflect in VVB. For applying the changes instantly, you need to restart the VVB speech server service.

# VAV Onboarding for Non-OEM Users

## Prerequisites

- CVP/VVB configuration:
  - Enable access to cloud-based services from CVP and VVB directly or via proxy.

    For details, see the *Cisco Unified Customer Voice Portal > Operations Console (NOAMP) > Integration > Cloud Connect > Configure CVP or VVB Devices for Cloud Connect* section in the *Administration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.
  - Synchronize the date/time in CVP, VVB, and proxy with a common NTP server.
  - Configure access to DNS server in CVP/VVB.

For more information on NTP and DNS server configurations in CVP, refer to the Microsoft Windows platform documentation.

For more information on NTP and DNS server configurations in VVB, refer to the *Command Line Interface Reference Guide for Cisco Unified Communications Solutions* at https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/products-maintenance-guides-list.html.

- Non-OEM users must enable speech services and generate JSON key. To know more about enabling speech services, see Enable Speech Services (For Non-OEM Users), on page 189. To know more about generating JSON key, see Generate JSON Key (for Non-OEM Users), on page 190.
- The following table lists the required versions for Cisco Unified CVP, Cisco VVB, Cloud Connect, CCE components (Router, Logger, AW, and PG), and ICM.

| CVP | VVB | Cloud Connect | CCE Components | ICM |
|---|---|---|---|---|
| 12.5 and higher | 12.5 and higher | 12.5 and higher | 11.6 and higher | 11.6 and higher |

# VAV Onboarding for Non-OEM Users Task Flow

**Procedure**

**Step 1** Enable speech services for your account.

To know more about enabling speech services, see Enable Speech Services (For Non-OEM Users), on page 189.

**Step 2** Generate JSON key for your account.

To know more about generating JSON key, see Generate JSON Key (for Non-OEM Users), on page 190.

**Step 3** Configure VVB devices for speech services.

For details, see the *Cisco Unified Customer Voice Portal > Operations Console (NOAMP) > Contact Center AI > Configuration for Vendor-Billed AI Services* section in the *Administration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

# Enable Speech Services (For Non-OEM Users)

To enable speech services, follow these steps:

1. Log in to your Dialogflow account at https://dialogflow.cloud.google.com/.
2. Scroll down on the homepage and click **Project ID** of your Dialogflow agent.

   This takes you to the Google Cloud Platform (GCP) homepage.
3. Select **APIs & Services** from the left pane (through the hamburger menu).
4. Select the API services (such as Cloud Text-to-Speech, Cloud Speech-to-Text, and Dialogflow) to be enabled.
5. Click **Enable** to enable the selected API for the given Project ID.

# Generate JSON Key (for Non-OEM Users)

To generate the JSON key, follow these steps:

1. In the GCP homepage, select **IAM & Admin** from the left pane (through the hamburger menu).
2. Select **Service accounts** which shows the list of your enabled services.
3. Select the service for which the JSON key is to be generated.
4. Click the ellipsis menu on the right and click **+Create Key**.
5. Select JSON as **Key type** and then click **Create**.

   The key is downloaded.

For best results:

- Migrate your Dialogflow Agent to Enterprise Essential (**Console Left Bar** > **Migrate from Standard to Enterprise Essential**).
- Enable the enhanced Speech Model in Dialogflow console (**Settings** > **Speech** > **Enable Enhanced Speech Model and Data Logging**).

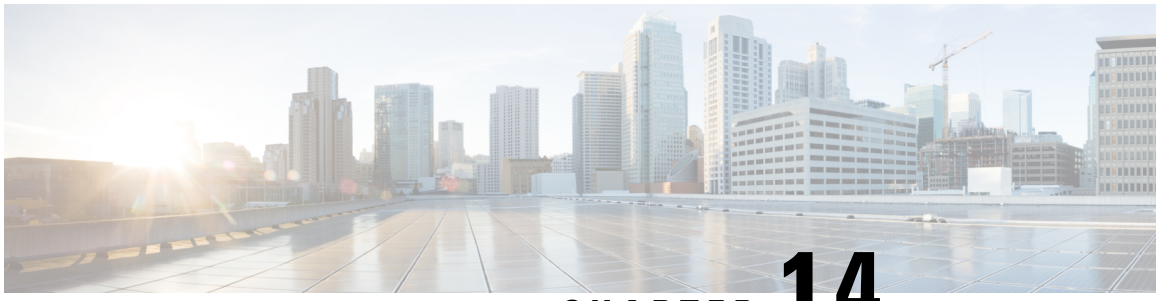
If this option is enabled, speech recognition data is shared with Google. For more information see https://cloud.google.com/speech-to-text/docs/enhanced-models.

# Documentation Resources

The following table lists the reference documents for VAV.

| Information | Resource |
|---|---|
| Sample VAV Application | See https://github.com/CiscoDevNet/cvp-sample-code/tree/master/CustomerVirtualAssistant. |
| Design Considerations | *Design Considerations for Integrated Features* > *Virtual Agent–Voice Considerations* section in *Solution Design Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-implementation-design-guides-list.html. |
| VAV configuration in Unified CCE Deployment | *Operations Console (NOAMP)* section in *CVP Administration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html. |
| TTS Prompt Cache Management and proxy setting for Speech Server | *VVB Operations Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/virtualized-voice-browser/products-maintenance-guides-list.html. |
| Proxy settings for VXML Server | See the *VXML Server Configuration* > *Proxy Settings in VXML Server for Virtual Agent–Voice* section in *CVP Configuration Guide* at https://www.cisco.com/c/en/us/support/ |

| Information | Resource |
|---|---|
| | customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html. |
| Configuration of Call Studio elements for VAV | The following chapters in *CVP Element Specification Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-programming-reference-guides-list.html:<br>• *Dialogflow Element*<br>• *DialogflowIntent Element*<br>• *DialogflowParam Element*<br>• *Transcribe Element* |
| VAV Speech Configuration APIs | See *VAV Speech Configuration* section in *VVB Developer Guide* at https://developer.cisco.com/site/customer-voice-portal/documents/virtual-voice-browser/. |



CHAPTER 14

# Virtual Agent–Voice for Dialogflow CX



## Overview

Virtual Agent–Voice for Dialogflow CX leverages Google's Dialogflow CX service that allows designing virtual voice agents and creating and connecting complex IVR call flows.

Using Google Dialogflow CX, multiple agents can be created under the same Project ID and can be accessed and managed for different lines of business with a single Google account. For more information, refer to the Google Dialogflow CX documentation at https://cloud.google.com/dialogflow/cx/docs.

The DialogflowCX element of Cisco Unified Call Studio is used to engage Google's Dialogflow CX services.

**Note**

- Only U Law codec is supported.
- This feature is available with Cisco subscription services only.
- The DialogflowCX element works both with Cisco DTMF and Nuance ASR adaptors.

## Prerequisites

For supported minimum versions, refer to the *Cisco Unified Customer Voice Portal > New Features > Virtual Agent–Voice for Dialogflow CX* section in the *Release Notes for Cisco Contact Center Enterprise Solutions Release 12.6(1)*.

To configure Google Dialogflow CX, you should have completed the following procedures:

- The customer's CX Agent ID and GCP Project ID is created. For more information, refer to Google documentation at https://dialogflow.cloud.google.com/cx/projects.
- Assessment to Quality (A2Q) process for Contact Center AI (CCAI) is completed and Cisco subscription Flex SKU for CX is procured.

- Customer's GCP project ID and Cisco's GCP partner projects are mapped.
- Control Hub credentials and Hybrid Org are generated.

For assistance, you can contact the Cisco TAC team.

- CX conversational profile ID is created using credentials provided by Cisco via email. For details, see Create a Conversation Profile using Google Cloud SDK, on page 195.

  This conversation profile URL is to be used for creating the CCAI configuration.

- Cisco Unified CVP and Cisco VVB are configured:
  - Date and time in CVP, VVB, and proxy are synchronized with a common NTP server.
  - Access to DNS server is configured in CVP and VVB.

  For more information on NTP and DNS server configurations in CVP, refer to the Microsoft Windows platform documentation.

  For more information on NTP and DNS server configurations in VVB, refer to the *Command Line Interface Reference Guide for Cisco Unified Communications Solutions* at https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/products-maintenance-guides-list.html.

- Port 443 and HTTP/2 are enabled in the proxy and firewall.
- The new Cisco Unified Call Studio application with the *DialogflowCX* element is deployed. You can download and install the latest patch from https://software.cisco.com/download/specialrelease/c359e375005563ceec2081c9151b482e.

  For details of the *DialogflowCX* element , refer to the *Element Specifications for Cisco Unified CVP VXML Server and Call Studio, Release 12.6(1)* guide at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-programming-reference-guides-list.html.

### What to do next

- Configure Google Dialogflow CX Agent with Cisco Unified CCE solution.

# Configuration Task Flow

Task flow for configuring Google Dialogflow CX Agent with Cisco Unified CCE solution.

### Procedure

**Step 1** Ensure that the Cloud Connect publisher and subscriber nodes are installed.

For more information, see the *Install Cloud Connect* section in *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-guides-list.html.

**Step 2** Configure Cloud Connect with VVBs in the Operations Console (NOAMP for Cisco Unified CCE and CCE Admin for Packaged CCE).

For details, see the *Cisco Unified Customer Voice Portal* > *Operations Console (NOAMP)* > *Integration* > *Cloud Connect* > *Configure CVP or VVB Devices for Cloud Connect* section in the *Administration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

**Step 3** Create Contact Center AI (CCAI) configuration in Cisco Webex Control Hub at https://admin.webex.com. A CCAI configuration leverages CCAI Connectors to invoke the CCAI services.

For detailed steps to set up Integration Connectors, see the *Set Up Integration Connectors for Contact Center solutions* article.

For detailed steps to create CCAI configuration, see the *Create a Contact Center AI configuration* article. This default config can be used for accessing multiple AI services on multiple devices.

**Step 4** View the default CCAI configuration (created in step 1). Sync, if required, in the CVP Operations Console (NOAMP for Cisco Unified CCE and CCE Admin for Packaged CCE).

For details, see the *Cisco Unified Customer Voice Portal* > *Operations Console (NOAMP)* > *Contact Center AI* > *Configuration for Cisco-Billed AI Services* section in the *Administration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

**Step 5** Design Google Dialogflow CX Agents. For more information, refer to Google documentation at https://dialogflow.cloud.google.com/cx/projects.

**Step 6** Configure a welcome event for all Google Dialogflow CX Agents. For details, see Create a Welcome Event, on page 197.

**Note**
- To override the default welcome event, provide the element data *event_name* in the DialogflowCX element. The same name must be configured at the CX Agent to start the flow.
- For further assistance, contact the Cisco TAC team.

# Create a Conversation Profile using Google Cloud SDK

**Note** The steps in the following procedure are for reference only. For more details, refer to Google Documentation.

**Procedure**

**Step 1** Get the agent ID:

**a.** Open https://dialogflow.cloud.google.com/cx/projects.

**b.** Select the appropriate project. The list of configured agents is displayed.

**c.** Note the agent ID to be configured. If no new agents are created, you can select a preconfigured agent.

**Step 2** Create a user via Google IAM (Identity and Access Management) and add the following roles: Dialogflow API admin, Service Account Token Creator, and Service Account user .

For more information, see https://cloud.google.com/iam/docs/creating-managing-service-accounts.

**Step 3** Install and configure the Google SDK on your system.

For more information, see https://cloud.google.com/sdk/docs/quickstart.

**Step 4** When you are asked to log in during the installation of Google SDK, log in using the ID of the agent for whom you want to create the conversation profile.

**Note** You can also log in using the same ID to the GCP CLI with the command `gcloud auth login` after installing the Google SDK.

**Step 5** Run the following command: `gcloud auth print-access-token --impersonate-service-account=Service Account ID`

For more information, see https://cloud.google.com/iam/docs/impersonating-service-accounts.

**Step 6** Create the conversation profile using REST API for Dialogflow by using Postman:

a) In the Postman workspace, select the method as **POST**.

b) In the URL field, add the address in the following format after replacing the *regionId* and *projectId* appropriately:

`https://<regionId>-dialogflow.googleapis.com/v2beta1/projects/<projectId>/locations/<regionId>/conversationProfiles`

c) Under the **Headers** section, add the following key values for Authorization and Content-type:

- Authorization: *Bearer <token generated with the Google command>*
- Content-type: *application/json*

d) Under the **Body** section, select **raw**. From the last dropdown, select **JSON**.

e) In the code space, enter the following code (after replacing the *regionId*, *projectId*, and *agentId* in the **agent** tag with actual values):

```
{
"name": "TACCXTest",
"automatedAgentConfig": {
    "agent": "projects/<projectId>/locations/<regionId>/agents/<agentId>”
    },
"displayName": "TACCXTest",
"humanAgentAssistantConfig": {
    "messageAnalysisConfig": {
        "enableEntityExtraction": true,
        "enableSentimentAnalysis": true
        }
    }
}
```

f) Click **Send** to run the command.

Example response:

```
{
"name":
"projects/projectrtp2020/locations/us-central1/conversationProfiles/QlO36mwSUa3cjg",
"displayName": "TACCXTest",
"automatedAgentConfig": {
"agent":
"projects/projectrtp2020/locations/us-central1/agents/40d0-aa2a-1bf453d9bf5c/environments/draft"
```

```
    },
"humanAgentAssistantConfig": {
"notificationConfig": {},
"messageAnalysisConfig": {
"enableEntityExtraction": true,
"enableSentimentAnalysis": true
        }
    },
"languageCode": "en-US"
}
```

In the above example response, the following conversation profile is obtained: *projects/projectrtp2020/locations/us-central1/conversationProfiles/dQlO36mwSUa3cjg.*

You can use this profile while creating the Control Hub configuration.

For more information, see https://cloud.google.com/sdk/gcloud/reference.

**Step 7** In a new browser tab, open https://agentassist.cloud.google.com/ and select the appropriate project. The list of profiles is displayed.

**Step 8** Click the copy icon next to the profile ID to be used. Copy the profile URL in the following format: *projects/<project_ID>/locations/<location>/conversationProfiles/<profile ID>*.

You can use this profile URL while creating the Control Hub configuration.

# Create a Welcome Event

Create a welcome event to be played to the caller when a call is initiated.

### Procedure

**Step 1** Open https://dialogflow.cloud.google.com/cx/projects.

**Step 2** Select the project and agent for which the welcome event is to be configured.

**Step 3** In the Google Dialogflow CX Agent screen, click **Default Start Flow** in the left panel.

**Step 4** Click **Start** > **Event handlers**.

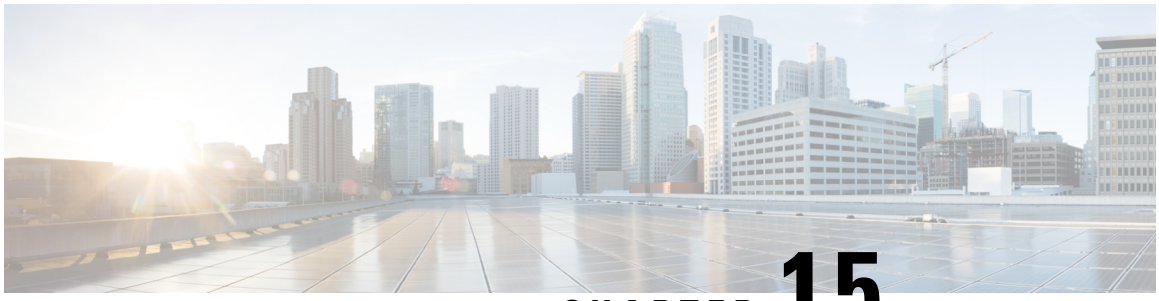
**Step 5** In the right panel, click **Add event handler**.

**Step 6** Check the **Use custom event** checkbox.

**Step 7** In the **Custom Event** textbox, type **welcome_event**.

**Step 8** In the **Agent says** textbox, type the welcome message to be played.

**Step 9** Save the changes.



CHAPTER 15

# VPN-less Access to Finesse Desktop



## Introduction

This section describes how to configure a reverse-proxy and access the Cisco Finesse desktop without connecting to a VPN based on 12.6 ES03 and above versions of Cisco Finesse, Cisco Unified Intelligence Center (CUIC), and Cisco Identity Service (IdS).

**Note**

- The content in this chapter is provided as a guidance for customers to install and configure reverse-proxy. Cisco does not support requests for reverse-proxy installation and configuration issues. Queries that are related to this subject can be discussed onCisco community forums.
- For ES04 deployments of VPN-less Finesse, see the Cisco Finesse 12.6 ES04 Readme.

**Note** The OpenResty® Nginx configurations provided as part of Release 12.5(1) SU2 need to be manually edited and applied to match your deployment, along with requiring a manual install of the OpenResty® Nginx.

## Prerequisites

### Requirements

Cisco recommends that you have knowledge of the following:

- Cisco Unified Contact Center Enterprise (Unified CCE) Release
- Cisco Finesse
- Linux administration
- Network administration and Linux network administration

## Components Used

The information in this section is based on the following software and hardware versions:

- Cisco Finesse - 12.6 ES03
- Cisco Unified Intelligence Center - 12.6 ES03
- IdS - 12.6 ES03
- Unified CCE - 11.6 or later
- Packaged CCCE - 12.0 or later

### Related Topics

Performance

# Background Information

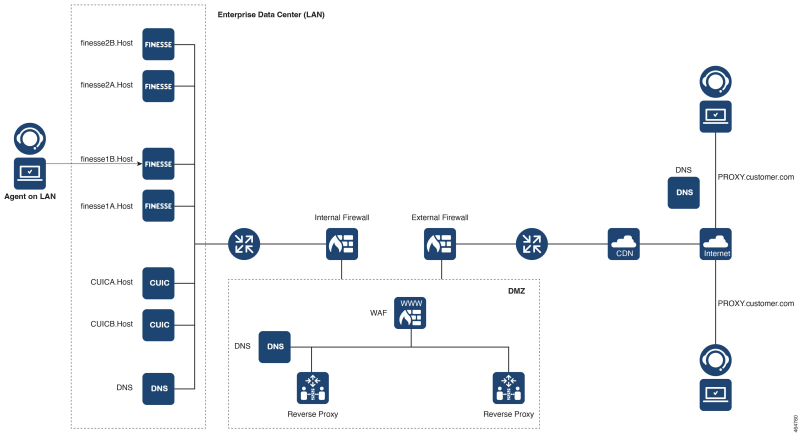
This deployment model is supported for the Unified CCE and Packaged CCE solutions.

Deployment of a reverse-proxy is supported (available from 12.6 ES01) as an option to access the Cisco Finesse desktop without connecting to a VPN.

To enable this feature, a reverse-proxy pair must be deployed in the Demilitarized Zone (DMZ).

Media access remains unchanged in reverse-proxy deployments. To connect to the media, agents can use Cisco Jabber over MRA solution or the Mobile Agent capability of Unified CCE with a Public Switched Telephone Network (PSTN) or mobile endpoint. This diagram shows how the network deployment will look like when you access two Finesse clusters and two Cisco Unified Intelligence Center nodes through a single HA pair of reverse-proxy nodes.

Concurrent access from agents on the Internet and agents who connect from LAN is supported as shown in the following image:




**Note** For more information on how to select an appropriate reverse-proxy that supports this deployment, see the section Reverse-Proxy Selection Criteria at *Security Guide for Cisco Unified ICM/Contact Center Enterprise, Release 12.6(1).*

Before you read this section, it is suggested to refer to VPN-less Access to Finesse Desktop. Also, see the *Security Considerations for Mobile Agent Deployments* section in *Security Guide for Cisco Unified ICM/Contact Center Enterprise, Release 12.6(1)*.

### Related Topics

Components Used, on page 200

# Upgrade Notes for ES01-Based VPN-less Configurations

- Nginx ES03-based configuration requires Nginx installation with Lua support.
- Certificate Requirements
  - Cisco Finesse, Cisco Unified Intelligence Center, and IdS require the Nginx or OpenResty host certificate to be added to the Tomcat trust store and restart the system. This enables Nginx ES02-based configuration to successfully connect to the component servers.
  - Cisco Finesse, Cisco Unified Intelligence Center, and IdS upstream server certificates need to be configured in the Nginx server to use the ES03-based configuration.

**Note** It is recommended to remove the existing ES01-based Nginx configuration before you install the ES03-based Nginx configurations.

**Note** Nginx ES03-based configuration scripts require the corresponding ES03 COP installation in Cisco Finesse, Cisco Unified Intelligence Center, and IdS.

## Validating Unauthenticated Static Resources

All valid endpoints that can be accessed without any authentication are actively tracked in the ES04 scripts . Requests to these unauthenticated paths are rejected without sending these requests to the components servers, if an invalid URI is requested.

## Brute Force Attack Prevention

Finesse 12.6 ES02 and above authentication scripts actively prevent brute force attacks that can be used to guess the user password. The scripts do this by blocking the IP address used to access the service, after a certain number of failed attempts in a short time. These requests will be rejected by **418 client error**. The number of failed requests, time interval, and blocking duration are configurable.

For more information, see the Brute Force Attack Prevention Configuration section.

## Caching CORS Headers

When the first options request is successful, the response headers **access-control-allow-headers, access-control-allow-origin, access-control-allow-methods, access-control-expose-headers,** and **access-control-allow-credentials** are cached at the proxy for five minutes. These headers are cached for each respective upstream server.

# Reverse-Proxy Configuration

## Install OpenResty as a Reverse-Proxy in DMZ

This section details the OpenResty-based proxy installation steps. The reverse-proxy is typically configured as a dedicated device in the network demilitarized zone (DMZ) as shown in the deployment diagram in *Background Information*.

1. Install the OS of your choice with the required hardware specification. Kernel and IPv4 parameter tweaks might differ depending on the OS selected. Users are advised to reverify these aspects if the chosen OS version is different from CentOS 8.0.
2. Configure two network interfaces. One interface will be required for public access from the Internet clients and another to communicate with the servers in the internal network.
3. Install OpenResty.

Any of the following Nginx versions can be used for this purpose, as long as they are based on Nginx 1.19+ and support Lua:

- Nginx Plus
- Nginx Open Source (Nginx open source must be compiled along with OpenResty-based Lua modules)
- OpenResty
- GetPageSpeed Extras

**Note** The configuration provided has been tested with OpenResty 1.19 and is expected to work with other distributions with only minor updates, if any.

# Install OpenResty

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | Install OpenResty. See OpenResty Linux Packages. | As part of the OpenResty installation, Nginx will be installed in this location. Add the OpenResty path to the ***PATH*** variable by adding the following line in the ***~/.bashrc*** file.<br>`export PATH=/usr/local/openresty/bin:$PATH` |
| **Step 2** | Start or stop OpenResty Nginx | • To start OpenResty Nginx, enter **`openresty`**.<br>• To stop OpenResty Nginx, enter **`openresty -s stop`**. |

# Configure OpenResty Nginx

The configuration is explained for an OpenResty-based Nginx installation. The default directories for OpenResty are:

- <nginx-install-directory> = /usr/local/openresty/nginx
- <Openresty-install-directory> = /usr/local/openresty

1. Download and extract the 12.6-ES04-reverse-proxy-config.zip file that contains the reverse-proxy configuration for Nginx. This file is available on the Finesse Release 12.6(1)ES04 software download page.
2. Copy `nginx.conf, nginx/conf.d/`, and `nginx/html/` from the extracted reverse-proxy configuration directory to `<nginx-install-directory>/conf, <nginx-install-directory>/conf/conf.d/`, and `<nginx-install-directory>/html/` respectively.
3. Copy the `nginx/lua` directory from the extracted reverse-proxy configuration directory inside the `<nginx-install-directory>`.
4. Copy the contents of lualib to `<Openresty-install-directory>/lualib/resty`.
5. Configure OpenResty Nginx log rotation by copying the `nginx/logrotate/saproxy` file to the `<nginx-install-directory>/logrotate/` folder. Modify the file contents to point to the correct log directories if OpenResty Nginx defaults are not used.

6. OpenResty Nginx must be run with a dedicated non-privileged service account, which must be locked and have an invalid shell (or as applicable for the chosen OS).
7. Find the **Must-change** string in the files under the extracted folders named `html` and `conf.d` and replace the indicated values with the appropriate entries.
8. Ensure that all mandatory replacements are done, which are described with the **Must-change** comments in the config files.
9. Make sure that the cache directories configured for Cisco Unified Intelligence Center and Finesse are created under `<nginx-install-directory>/cache` along with these temporary directories.
   - <nginx-install-directory>/cache/client_temp
   - <nginx-install-directory>/cache/proxy_temp

**Note** The configuration provided is for a sample 2000 Agent deployment and has to be expanded appropriately for a larger deployment.

## Configure the OpenResty Nginx Cache

By default, the proxy cache paths are stored in the file system. We recommend changing them to in-memory drives by creating a cache location in tmpfs as shown here.

1. Create directories for the different proxy cache paths under /home.

   As an example, these directories must be created for the primary Finesse server. The same steps should be followed for the secondary Finesse and Cisco Unified Intelligence Center servers.

```
mkdir -p /home/primaryFinesse/rest
mkdir -p /home/primaryFinesse/desktop
mkdir -p /home/primaryFinesse/shindig
mkdir -p /home/primaryFinesse/openfire
mkdir -p /home/primaryCUIC/cuic
mkdir -p /home/primaryCUIC/cuicdoc
mkdir -p /home/client_temp
mkdir -p /home/proxy_temp

echo "tmpfs /home/primaryFinesse/rest tmpfs
size=1510M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryFinesse/desktop tmpfs
size=20M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryFinesse/shindig tmpfs
size=500M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryFinesse/openfire tmpfs
size=10M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryCUIC/cuic tmpfs
size=100M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryCUIC/cuicdoc tmpfs
size=100M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/client_temp tmpfs
size=2048M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/proxy_temp tmpfs
size=2048M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >> /etc/fstab
```

**Note** Increase the client and proxy_temp caches by 1 GB for each new Finesse cluster added to the configuration.

2. Mount the new mount points with the `mount -av` command.
3. Use the `df -h` command to validate if the file system has mounted the new mount points.
4. Change the proxy_cache_path locations in the Finesse and Cisco Unified Intelligence Center cache configuration files. For example, to change the paths for the Finesse primary, go to `<nginx-install-directory>conf/conf.d/finesse/caches` and change the existing cache location `/usr/local/openresty/nginx/cache/finesse25/` to the newly created filesystem location `/home/primaryFinesse.`

```
##Must-change /usr/local/openresty/nginx/cache/finesse25 location would change depending
 on folder extraction
proxy_cache_path /home/primaryFinesse/desktop levels=1:2 use_temp_path=on
keys_zone=desktop_cache_fin25:10m max_size=15m inactive=3y use_temp_path=off;
proxy_cache_path /home/primaryFinesse/shindig levels=1:2 use_temp_path=on
keys_zone=shindig_cache_fin25:10m max_size=500m inactive=3y use_temp_path=off;
proxy_cache_path /home/primaryFinesse/openfire levels=1:2 use_temp_path=on
keys_zone=openfire_cache_fin25:10m max_size=10m inactive=3y use_temp_path=off;
proxy_cache_path /home/primaryFinesse/rest levels=1:2 use_temp_path=on
keys_zone=rest_cache_fin25:10m max_size=1500m inactive=40m use_temp_path=off;
```

5. Follow the same steps for the Finesse secondary server and Cisco Unified Intelligence Center server.

**Note** Ensure that the sum of all the **tmpfs** drive sizings created in all the previous steps are added to the final memory sizing for the deployment. This is because these drives are memory blocks that are configured to look like disks to the application and they consume memory.

## Configure Log Rotation

Nginx reverse-proxy produces many logs. Configure log rotation to ensure optimum use of disk space, else the logs fill up the disk. The steps to configure log rotation is as follows:

1. Copy the configuration file from `/usr/local/openresty/nginx/logrotate/saproxy` to `/etc/logrotate.d/reverseproxy`.
2. Ensure that there's a file in the `/etc/cron.daily/logrotate`. This does the log rotation daily based on the configuration in the `/etc/logrotate.d/reverseproxy`.

**Note** If log rotation has to be done more frequently, such as every hour, copy the configuration file from `/etc/cron.daily/logrotate` to `/etc/cron.hourly/logrotate`.

3. Log files under `/usr/local/openresty/nginx/logs/` are rotated based on the configuration in `/etc/logrotate.d/reverseproxy`. That is, rotate the log when the file size exceeds 100 MB and keep no more than 20 most recently rotated files.

The status of the log rotation is available in the `/var/lib/logrotate/logrotate.status` log file.

## Use Self-Signed Certificates—Test Deployments

Use self-signed certificates until the reverse-proxy is ready to be rolled out into production. On a production deployment, use only a Certificate Authority-signed (CA-signed) certificate.

1. Generate OpenResty Nginx certificates for SSL folder content. Before you generate certificates, you must create a folder called **ssl** under `/usr/local/openresty/nginx`. Generate two certificates (one for `<reverseproxy_primary_fqdn>` and another for `<reverseproxy_secondary_fqdn>`) with the help of the following commands:
   a. `sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /usr/local/openresty/nginx/ssl/nginx.key -out /usr/local/openresty/nginx/ssl/nginx.crt` (pass hostname as: <reverseproxy_primary_fqdn>)
   b. `sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /usr/local/openresty/nginx/ssl/nginxnode2.key -out /usr/local/openresty/nginx/ssl/nginxnode2.crt` (pass hostname as: <reverseproxy_secondary_fqdn>)
   c. Ensure that the certificate path is `/usr/local/openresty/nginx/ssl/nginx.crt` and `/usr/local/openresty/nginx/ssl/nginxnode2.crt`, because these are already configured in Finesse Nginx configuration files.
2. Change the permission of the private key **400 (r--------)**.
3. Configure the firewall and iptables on the reverse-proxy to enable the firewall to communicate with the ports that are configured to listen to the OpenResty Nginx server.
4. Add the IP address and hostname of all the configured servers in the `/etc/hosts` file of the reverse-proxy server.

**Note** The provided configuration is for a sample 2000 Agent deployment and must be expanded appropriately for larger deployments.

## Use CA-Signed Certificate—Production Deployments

A CA-signed certificate can be installed on the reverse-proxy with these steps:

1. Generate the certificate signing request (CSR).

   To generate the CSR and private key, enter `openssl req -new -newkey rsa:4096 -keyout nginx.key -out nginx.csr` after you log in to the proxy. Follow the prompt, and provide the details. This generates the CSR (nginx.csr in the example) and the RSA private key (nginx.key in the example) of 4096 bits.

   For example:

```
[root@reverseproxyhost.companyname.com ssl]# openssl req -new -newkey rsa:4096 -keyout
nginx.key -out nginx.csr
Generating a RSA private key
.....+++++
.....+++++
writing new private key to 'nginx.key'
Enter PEM pass phrase:passphrase
Verifying - Enter PEM pass phrase:passphrase
```

```
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:US
State or Province Name (full name) []:CA
Locality Name (eg, city) [Default City]:Orange County
Organization Name (eg, company) [Default Company Ltd]:CompanyName
Organizational Unit Name (eg, section) []:BusinessUnit
Common Name (eg, your name or your server's hostname)
[]:reverseproxyhostname.companydomain.com
Email Address []:john.doe@comapnydomain.com
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:challengePWD
An optional company name []:CompanyName
```

Write down the PEM passphrase. This is used to decrypt the private key during the deployment.

2. Obtain the signed certificate from the CA.

   Send the CSR to the certificate authority and obtain the signed certificate.

**Note** If the certificate received from the CA is not a certificate chain containing all the respective certificates, compose all the relevant certificates into a single certificate chain file.

3. Deploy the certificate and key.

   Decrypt the key generated in the first step with the `openssl rsa -in nginx.key -out nginx_decrypted.key` command. Place the CA-signed certificate and the decrypted key inside the folder `/usr/local/openresty/nginx/ssl` in the reverse-proxy machine. Update or add the following SSL configurations related to the certificate in the OpenResty Nginx configurations in the following file: `/usr/local/openresty/nginx/conf/conf.d/ssl/ssl.conf`:

   ```
   ssl_certificate /usr/local/openresty/nginx/ssl/ca_signed_cert.crt;
   ssl_certificate_key/usr/local/openresty/nginx/ssl/nginx_decrypted.key;
   ```

4. Configure permissions for the certificates.

   Enter `chmod 400 /usr/local/openresty/nginx/ssl/ca_signed_cert.crt` and `chmod 400 /usr/local/openresty/nginx/ssl/nginx_decrypted.key`, so that the certificate has read-only permission and is restricted to the owner.

5. Reload OpenResty Nginx.

## Create Custom Diffie-Hellman Parameter

1. Create a custom Diffie-Hellman parameter by using the following commands:

   ```
   openssl dhparam -out /usr/local/openresty/nginx/ssl/dhparam.pem 2048
   chmod 400 /usr/local/openresty/nginx/ssl/dhparam.pem
   ```

2. Modify the server configuration to use the new parameters in the file `/usr/local/openresty/nginx/conf/conf.d/ssl/ssl.conf` by using the following command:

```
ssl_dhparam /usr/local/openresty/nginx/ssl/dhparam.pem;
```

## Enable OCSP Stapling

**Note** In order to enable the Online Certificate Status Protocol (OCSP) stapling, the server should be using a CA-signed certificate and the server should have access to the CA which signed the certificate.

Add or update this configuration in the file: `/usr/local/openresty/nginx/conf/conf.d/ssl/ssl.conf`:

- `ssl_stapling on;`
- `ssl_stapling_verify on;`

## Modify the Common OpenResty Nginx Configuration

The default OpenResty Nginx configuration file `(/usr/local/openresty/nginx/conf/nginx.conf)` has to be modified to contain the following entries to enforce security and enhance performance. The following content should be used to modify the default configuration file (`nginx.conf`) which is created during the OpenResty Nginx installation:

```
# Increasing number of worker processes will not increase the processing the request. The
number of worker process will be same as number of cores
# in system CPU. OpenResty Nginx provides "auto" option to automate this, which will spawn
 one worker for each CPU core.
worker_processes  auto;

# Process id file location
pid /usr/local/openresty/nginx/logs/nginx.pid;

# Binds each worker process to a separate CPU
worker_cpu_affinity auto;

#Defines the scheduling priority for worker processes. This should be calculated by "nice"
 command. In our proxy set up the value is 0
worker_priority 0;


error_log  /usr/local/openresty/nginx/logs/error.log info;

#user root root;

# current limit on the maximum number of open files by worker processes, keeping 10 times
of worker_connections

worker_rlimit_nofile 102400;

events {
    multi_accept on;


    # Sets the maximum number of simultaneous connections that can be opened by a worker
process.
```

```
    # This should not be more the current limit on the maximum number of open files i.e.
hard limit of the maximum number of open files for the user (ulimit -Hn)
    # The appropriate setting depends on the size of the server and the nature of the
traffic, and can be discovered through testing.
    worker_connections  10240;
    #debug_connection 10.78.95.21
}


http {

    include       mime.types;


    default_type  text/plain;


    ## Must-change Change with DNS resolver ip in deployment
    resolver 192.168.1.3;


    ## Must-change change lua package path to load lua libraries
    lua_package_path
"/usr/local/openresty/lualib/resty/?.lua;/usr/local/openresty/nginx/lua/?.lua;;"


    ## Must-change change proxy_temp folder as per cache directory configurations
    proxy_temp_path /usr/local/openresty/nginx/cache/proxy_temp 1 2 ;
    ## Must-change change client_temp folder as per cache directory configurations
    client_body_temp_path /usr/local/openresty/nginx/cache/client_temp 1 2 ;


    lua_shared_dict userlist 50m;
    lua_shared_dict credentialsstore 100m;
    lua_shared_dict userscount 100k;
    lua_shared_dict clientstorage 100m;
    lua_shared_dict blockingresources 100m;
    lua_shared_dict tokencache_saproxy 10M;
    lua_shared_dict tokencache_saproxy125 10M;
    lua_shared_dict ipstore 10m;
    lua_shared_dict desktopurllist 10m;
    lua_shared_dict desktopurlcount 100k;
    lua_shared_dict thirdpartygadgeturllist 10m;
    lua_shared_dict thirdpartygadgeturlcount 100k;
    lua_shared_dict corsheadersstore 100k;


    init_worker_by_lua_block {
        local UsersListManager = require('users_list_manager')
        local UnauthenticatedDesktopResourcesManager =
require("unauthenticated_desktopresources_manager")
        local UnauthenticatedResourcesManager =
require("unauthenticated_thirdpartyresources_manager")
        --  Must-change Replace saproxy.cisco.com with reverseproxy fqdn

        if  ngx.worker.id() == 0 then
            UsersListManager.getUserList("saproxy.cisco.com",
"https://saproxy.cisco.com:8445/finesse/api/Users")
           UnauthenticatedDesktopResourcesManager.getDesktopResources("saproxy.cisco.com",
 "https://saproxy.cisco.com:8445/desktop/api/urls?type=desktop")
          UnauthenticatedResourcesManager.getThirdPartyGadgetResources("saproxy.cisco.com",
 "https://saproxy.cisco.com:8445/desktop/api/urls?type=3rdParty")
        end
    }
```

```
include conf.d/*.conf;

sendfile        on;

tcp_nopush     on;

server_names_hash_bucket_size 512;
```

## Configure Reverse-Proxy Port

By default, the Nginx configuration for Finesse requests on port 8445. At a time, only one port can be enabled from a reverse-proxy to support Finesse requests. If port 443 needs to be supported, edit the `<nginx-install-directory>conf/conf.d/finesse.conf` file to enable listening on 443 and disable listening on 8445.

## Configure Mutual TLS Authentication Between Reverse-Proxy and Components

Client SSL certificate authentication for connections from reverse-proxy hosts can be enabled on Cisco Unified Intelligence Center, Finesse, IdS, and LiveData by using the new CVOS CLI option `utils system reverse-proxy client-auth enable/disable/status`.

By default this is disabled and has to be enabled by the administrator by executing the CLI on each upstream server independently. After this option is enabled, Cisco Web proxy Service running on upstream host will start authenticating client certificates in TLS handshake for connections originating from trusted reverse-proxy hosts added by using the CLI `utils system reverse-proxy allowed-hosts add <proxy-host>`.

The following is the configuration block for the same in proxy config files named **ssl.conf** and **ssl2.conf**.

```
#Must-change /usr/local/openresty/nginx/ssl/nginx.crt change this location accordingly
proxy_ssl_certificate /usr/local/openresty/nginx/ssl/nginx.crt;
#Must-change /usr/local/openresty/nginx/ssl/nginx.key change this location accordingly
proxy_ssl_certificate_key /usr/local/openresty/nginx/ssl/nginx.key;
```

The SSL certificate used for outbound traffic (proxy to upstream) can be the same as the SSL certificate that is configured for inbound traffic (SSL connector for component server blocks). If self-signed certificate is used as **proxy_ssl_certificate**, it has to be uploaded to the tomcat trust store of the upstream components (Finesse/IdS/Cisco Unified Intelligence Center/Livedata) for it to be authenticated successfully.

Upstream server certificate validation by reverse-proxy is optional and disabled by default. If you wish to achieve full TLS mutual auth between reverse-proxy and upstream hosts, the following configuration has to be uncommented in the **ssl.conf** and **ssl2.conf** files.

```
#Enforce upstream server certificate validation at proxy ->
#this is not mandated as per CIS buit definitely adds to security.
#It requires the administrator to upload all upstream server certificates to the proxy
certificate store
#Must-Change Uncomment below lines IF need to enforce upstream server certificate validation
 at proxy
#proxy_ssl_verify on;
#proxy_ssl_trusted_certificate /usr/local/openresty/nginx/ssl/finesse25.crt;
proxy_ssl_trusted_certificate: This file should contain the all upstream certificate enteries
 concatenated together
```

mutual TLS (mTLS) is a standard security requirement for connections established from DMZ into the data center. For more information, see Nginx CIS behcmarks-https://www.cisecurity.org/benchmark/nginx

mTLS requires that both the server and client be pre-configured with mutual information about each other, as well as that the mutual certificates be properly verified. Hence the term Mutual TLS. A properly configured proxy server will be able to circumvent TCP rate limits and provide the client IP to the server for logging

purposes. As a result, it is critical that the proxy identity be verified before connecting as a reverse-proxy. For security reasons, it is therefore recommended that this feature be used and turned on.

This requires the upstream component certificates to be made available to the proxy and vice-versa. Reverse-proxy by default establishes verified TLS connections to the upstream server and it is the proxy verification at the client which is optional. Therefore this needs to be enabled at the upstream client server.

### Enabling mutual TLS

The mutual TLS needs to be enabled at the upstream component servers using the provided CLI.

Use the **utils system reverse-proxy client-auth enable** CLI to enable proxy certificate verification at the upstream component server.

After running the CLI, upload the proxy SSL certificate corresponding to the reverse-proxy hostname used to connect to the same server. This can be used to verify TLS connections when the reverse-proxy attempts to establish an upstream connection.

# Clear Cache

The reverse-proxy cache can be cleared with the `<NGINX_HOME>/clearCache.sh` command.

# Standard Guidelines

This section briefly describes the standard guidelines that must be followed when you set up OpenResty Nginx as a proxy server. The guidelines for the OpenResty Nginx server software is derived from the Center for Internet Security.

1. Use the latest stable versions of OpenResty and OpenSSL version.
2. Install OpenResty Nginx in a separate disk mount.
3. The OpenResty Nginx process id must be owned by the root user (or as applicable for the chosen OS) and must have permission **644 (rw-------)** or stricter.
4. OpenResty Nginx must block requests for unknown hosts. Ensure that each server block contains the server_name directive explicitly defined. To verify, search all server blocks in the `nginx.conf` and `nginx/conf.d` files and verify that all server blocks contain the server_name.
5. OpenResty Nginx must listen only on the authorized ports. Search all server blocks in the `nginx.conf` and `nginx/conf.d` files and check for the `listen` directives to verify that only the authorized ports are open for requests.
6. Block the proxy server HTTP port, because Cisco Finesse does not support HTTP.
7. The OpenResty Nginx SSL protocol must be TLS 1.2. Remove support for legacy SSL protocols. Disable weak SSL ciphers.
8. Send the OpenResty Nginx error and access logs to the remote syslog server.
9. Install the **mod_security** module that works as a web application firewall. See the ModSecurity manual for more information. Note that OpenResty Nginx load has not been verified within the **mod_security** module in place.

# Configure the Mapping File

Refer to Host Mapping File for Network Translation.

**Related Topics**

Host Mapping File for Network Translation

## Use Reverse-Proxy as the Mapping File Server

**Note** This appendix has the configuration details, for more information about the pre-requisites, refer to Use Reverse-Proxy as the Mapping File Server, on page 212.

These steps are required only if the reverse-proxy is also used as the proxy mapping file host.

1. Configure the reverse-proxy hostname in the domain controller used by the Finesse, Cisco Unified Intelligence Center and IdS hosts such that its IP address can be resolved.
2. Upload the generated OpenResty® Nginx signed certificates on both the nodes under tomcat-trust of cmplatform and restart the server.
3. Update the **Must-change** values in `<NGINX_HOME>/html/proxymap.txt`.
4. Reload OpenResty® Nginx configurations with the `nginx -s reload` command.
5. Use the `curl` command to validate if the configuration file is accessible from another network host.

## CentOS 8 Kernel Hardening

If the operating system is Cent OS 8 and the installations use a dedicated server for hosting the proxy, harden the kernel by using these `sysctl` configurations:

```
## Configurations for kernel hardening - CentOS8. The file path is /etc/sysctl.conf
## Note that the commented configurations denote that CentOS 8's default value matches
## the recommended/tested value, and are not security related configurations.

# Avoid a smurf attack
net.ipv4.icmp_echo_ignore_broadcasts = 1
# Turn on protection for bad icmp error messages
net.ipv4.icmp_ignore_bogus_error_responses = 1

# Turn on syncookies for SYN flood attack protection
net.ipv4.tcp_syncookies = 1

# Turn on and log spoofed, source routed, and redirect packets
net.ipv4.conf.all.log_martians = 1
net.ipv4.conf.default.log_martians = 1

# Turn off routing
net.ipv4.ip_forward = 0
net.ipv4.conf.all.forwarding = 0
net.ipv6.conf.all.forwarding = 0

net.ipv4.conf.all.mc_forwarding = 0
net.ipv6.conf.all.mc_forwarding = 0

# Block routed packets
net.ipv4.conf.all.accept_source_route = 0
```

```
net.ipv4.conf.default.accept_source_route = 0
net.ipv6.conf.all.accept_source_route = 0
net.ipv6.conf.default.accept_source_route = 0

# Block ICMP redirects
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.default.accept_redirects = 0
net.ipv6.conf.all.accept_redirects = 0
net.ipv6.conf.default.accept_redirects = 0
net.ipv4.conf.all.secure_redirects = 0
net.ipv4.conf.default.secure_redirects = 0
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.send_redirects = 0

# Filter routing packets with inward-outward path mismatch(reverse path filtering)
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.default.rp_filter = 1

# Router solicitations & advertisements related.
net.ipv6.conf.default.router_solicitations = 0
net.ipv6.conf.default.accept_ra_rtr_pref = 0
net.ipv6.conf.default.accept_ra_pinfo = 0
net.ipv6.conf.default.accept_ra_defrtr = 0
net.ipv6.conf.default.autoconf = 0
net.ipv6.conf.default.dad_transmits = 0
net.ipv6.conf.default.max_addresses = 1
net.ipv6.conf.all.accept_ra = 0
net.ipv6.conf.default.accept_ra = 0

# Backlog - increased from default 1000 to 5000.
net.core.netdev_max_backlog = 5000

# Setting syn/syn-ack retries to zero, so that they don't stay in the queue.
net.ipv4.tcp_syn_retries = 0
net.ipv4.tcp_synack_retries = 0

# Max tcp listen backlog. Setting it to 511 to match nginx config
net.core.somaxconn = 511

# Reduce the duration of connections held in TIME_WAIT(seconds)
net.ipv4.tcp_fin_timeout = 6

# Maximum resources allotted
# fs.file-max = 2019273
# kernel.pid_max = 4194304
# net.ipv4.ip_local_port_range = 32768 60999

# TCP window size tuning
# net.ipv4.tcp_window_scaling = 1
# net.core.rmem_default = 212992
# net.core.rmem_max = 212992
# net.ipv4.tcp_rmem = 4096 87380 6291456
# net.ipv4.udp_rmem_min = 4096
# net.core.wmem_default = 212992
# net.core.wmem_max = 212992
# net.ipv4.tcp_wmem = 4096 16384 4194304
# net.ipv4.udp_wmem_min = 4096
# vm.lowmem_reserve_ratio = 256 256 32 0 0
# net.ipv4.tcp_mem = 236373 315167 472746

# Randomize virtual address space
kernel.randomize_va_space = 2

# Congestion control
```

```
# net.core.default_qdisc = fq_codel
# net.ipv4.tcp_congestion_control = cubic

# Disable SysReq
kernel.sysrq = 0

# Controls the maximum size of a message, in bytes
kernel.msgmnb = 65536

# Controls the default maximum size of a message queue
kernel.msgmax = 65536

# Controls the eagerness of the kernel to swap.
vm.swappiness = 1
```

Reboot after you make the recommended changes.

## IPtables Hardening

IPtables is an application that allows a system administrator to configure the IPv4 and IPv6 tables, chains, and rules provided by the Linux kernel firewall.

The IPtables rules are configured to secure the proxy application from brute force attacks by restricting the access in the Linux kernel firewall.

The comments in the configuration indicate which service is being rate-limited by using the rules.

**Note** If administrators use a different port or expand access to multiple servers using the same ports, they must do appropriate sizing for these ports accordingly.

A sample IPtable is as follows:

```
## Configuration for iptables service
## The file path is /etc/sysconfig/iptables
## Make a note for must-change values to be replaced.
## Restart of the iptable service is required after applying following rules

*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]

# Ensure loopback traffic is configured
-A INPUT -i lo -j ACCEPT
-A OUTPUT -o lo -j ACCEPT
-A INPUT -s 127.0.0.0/8 -j DROP

# Ensure ping openeded only for the particular source and blocked for rest
# Must-Change: Replace the x.x.x.x with valid ip address
-A INPUT -p ICMP --icmp-type 8 -s x.x.x.x -j ACCEPT

# Ensure outbound and established connections are configured
-A INPUT -p tcp -m state --state RELATED,ESTABLISHED -j ACCEPT
-A OUTPUT -p tcp -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT

# Block ssh for external interface
# Must-Change: Replace the ens224 with valid ethernet interface
-A INPUT -p tcp -i ens224 --dport 22 -j DROP
# Open inbound ssh(tcp port 22) connections
-A INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
```

```
# Configuration for ccp 8445 port
-A INPUT -p tcp -m tcp --dport 8445 --tcp-flags SYN SYN -m connlimit --connlimit-above 10
--connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " Connections to 8445 exceeded connlimit "
-A INPUT -p tcp -m tcp --dport 8445 --tcp-flags SYN SYN -m connlimit --connlimit-above 10
--connlimit-mask 32 --connlimit-saddr -j DROP
 -A INPUT -p tcp -m tcp --dport 8445 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 6/sec
 --hashlimit-burst 8 --hashlimit-mode srcip,dstport --hashlimit-name TCP_8445_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8445 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 8445 hashlimit "
-A INPUT -p tcp -m tcp --dport 8445 --tcp-flags SYN SYN -j DROP

# Configuration for IdS 8553 port
-A INPUT -p tcp -m tcp --dport 8553 --tcp-flags SYN SYN -m connlimit --connlimit-above 6
--connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " IdS connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 8553 --tcp-flags SYN SYN -m connlimit --connlimit-above 6
--connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 8553 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 2/sec
 --hashlimit-burst 4 --hashlimit-mode srcip,dstport --hashlimit-name TCP_8553_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8553 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 8553 hashlimit "
-A INPUT -p tcp -m tcp --dport 8553 --tcp-flags SYN SYN -j DROP

# Configuration for IdP 443 port
-A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -m connlimit --connlimit-above 8
--connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " IdP connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -m connlimit --connlimit-above 8
--connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 4/sec
 --hashlimit-burst 6 --hashlimit-mode srcip,dstport --hashlimit-name TCP_443_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 443 hashlimit "
-A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -j DROP


# Must-Change: A2A file transfer has not been considered for below IMNP configuration.
# For A2A for support, these configuration must be recalculated to cater different file
transfer scenarios.

# Configuration for IMNP 5280 port
-A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30
--connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " IMNP connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30
--connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 20/sec
 --hashlimit-burst 25 --hashlimit-mode srcip,dstport --hashlimit-name TCP_5280_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 5280 hashlimit "
-A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -j DROP

# Configuration for IMNP 15280 port
-A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30
 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " IMNP connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30
 --connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto
20/sec --hashlimit-burst 25 --hashlimit-mode srcip,dstport --hashlimit-name TCP_15280_DOS
-j ACCEPT
```

```
-A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 15280 hashlimit "
-A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -j DROP


# Configuration for IMNP 25280 port
-A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30
 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " IMNP connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30
 --connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto
20/sec --hashlimit-burst 25 --hashlimit-mode srcip,dstport --hashlimit-name TCP_25280_DOS
-j ACCEPT
-A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 25280 hashlimit "
-A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -j DROP


# Configuration for CUIC 8444 port
-A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -m connlimit --connlimit-above 6
--connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " CUIC connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -m connlimit --connlimit-above 6
--connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 2/sec
 --hashlimit-burst 4 --hashlimit-mode srcip,dstport --hashlimit-name TCP_8444_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 8444 hashlimit "
-A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -j DROP

# Configuration for CUIC 8447 port
-A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -m connlimit --connlimit-above 6
--connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " CUIC connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -m connlimit --connlimit-above 6
--connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 2/sec
 --hashlimit-burst 4 --hashlimit-mode srcip,dstport --hashlimit-name  TCP_8447_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 8447 hashlimit "
-A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -j DROP

# Configuration for LiveData 12005  port
-A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -m connlimit --connlimit-above 10
 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " LD connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -m connlimit --connlimit-above 10
 --connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 6/sec
 --hashlimit-burst 8 --hashlimit-mode srcip,dstport --hashlimit-name TCP_12005_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 12005 hashlimit "
-A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -j DROP

# Configuration for LiveData 12008  port
-A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -m connlimit --connlimit-above 10
 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " LD connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -m connlimit --connlimit-above 10
 --connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 6/sec
 --hashlimit-burst 8 --hashlimit-mode srcip,dstport --hashlimit-name  TCP_12008_DOS -j
ACCEPT
```

```
-A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 12008 hashlimit "
-A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -j DROP

# Block all other ports
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited

COMMIT
```

**Note** The rules that are provided block the DNS resolution at the proxy. So, all the hostnames of the components that are configured in the proxy must be explicitly added to the host resolution file `/etc/hosts`.

Interface level rules must be added to restrict access to only users accessing via LAN and to block public access to port 10000, which is used for accessing the proxy map file. For example,

```
-A INPUT -p tcp -m tcp -i <PRIVATE_INTERFACE> --dport 10000 --tcp-flags SYN SYN -m hashlimit
--hashlimit-upto 35/sec --hashlimit-burst 2000 --hashlimit-mode srcip,dstport --hashlimit-name
TCP_10000_DOS -j ACCEPT -A INPUT -p tcp -m tcp -i <PRIVATE_INTERFACE> --dport 10000
--tcp-flags SYN SYN -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " Exceeded
hashlimit " -A INPUT -p tcp -m tcp -i <PRIVATE_INTERFACE> --dport 10000 --tcp-flags SYN SYN
-j DROP
```

These rules could be applied directly by editing the `/etc/sysconfig/iptables` file manually. Alternatively, save the configuration into a file such as `iptables.conf` and run cat `iptables.conf >>/etc/sysconfig/iptables` to apply the rules.

Restart the IPtables service after you apply the rules. To restart the IPtables service, enter `systemctl restart iptables`.

# Restrict Client Connections

In addition to the previous IPtables configuration, installations that know the address range for clients who use the proxy must use this knowledge to secure the proxy access rules. This helps to secure the proxy from malicious botnets which are often created in the IP address range of countries that have more lax rules with regards to online security. Restrict the IP address ranges to country-based, state-based, or ISP-based IP ranges if you are sure of the access patterns.

# Block Client Connections

Block the specific range of addresses when an attack is identified to be made from an IP address or a range of IP addresses. In such cases, the requests from those IP addresses can be blocked with **iptable** rules.

### Block Distinct IP Addresses

To block multiple distinct IP addresses, add a line to the **IPTables** configuration file for each IP address.

For example, to block the addresses 192.0.2.3 and 192.0.2.4, enter:

```
iptables -A INPUT -s 192.0.2.3 -j DROP iptables -A INPUT -s 192.0.2.4 - j DROP.
```

### Block a Range of IP Addresses

Block multiple IP addresses in a range and add a single line to the **IPTables** configuration file with the IP address range.

For example, to block the addresses from 192.0.2.3 to 192.0.2.35, enter:

```
iptables -A INPUT -m iprange --src-range 192.0.2.3-192.0.2.35 -j DROP.
```

### Block All IP Addresses in a Subnet

Block all IP addresses in an entire subnet by adding a single line to the **IPTables** configuration file by using the classless inter-domain routing notation for the IP address range. For example, to block all class **C addresses**, enter:

```
iptables -A INPUT -s 192.0.0.0/16 -j DROP.
```

# SELinux

Security-Enhanced Linux (SELinux) is a platform security framework integrated as an enhancement into the Linux OS. The procedure to install and add SELinux policies to run OpenResty as the reverse-proxy is provided next.

1. Stop the process with the `openresty -s stop` command.
2. Configure and start or stop OpenResty Nginx server with the `systemctl` command so that during boot up the OpenResty process will start automatically. Enter these commands as root user.
   a. Go to `/usr/lib/systemd/system`.
   b. Open the file called `openresty.service`.
   c. Update the content of the file as per `PIDFile` location.

```
[Unit]
Description=The OpenResty Application Platform
After=syslog.target network-online.target remote-fs.target nss-lookup.target
Wants=network-online.target

[Service]
Type=forking
PIDFile=/usr/local/openresty/nginx/logs/nginx.pid
ExecStartPre=/usr/local/openresty/nginx/sbin/nginx -t
ExecStart=/usr/local/openresty/nginx/sbin/nginx
ExecReload=/bin/kill -s HUP $MAINPID
ExecStop=/bin/kill -s QUIT $MAINPID
PrivateTmp=true

[Install]
WantedBy=multi-user.target
```

   d. As root user, enter `sudo systemctl enable openresty`.
   e. Start or stop the OpenResty service with the `systemctl start openresty / systemctl stop openresty` command and ensure that the process starts or stops as root user.

1. **Install SElinux**
   - By default, only some SELinux packages will be installed in CentOS.

- The **policycoreutils-devel** package and its dependencies must be installed in order to generate the SELinux policy.
- Enter the following command to install **policycoreutils-devel**

```
yum install policycoreutils-devel
```

- Ensure that after you install the package, the `sepolicy` command works.

```
usage: sepolicy [-h] [-P POLICY]


{booleans,communicate,generate,gui,interface,manpage,network,transition}
                ...

SELinux Policy Inspection Tool
```

2. **Create a New Linux User and Map with SElinux User**

   **a.** Enter `semanage login -l` to view the mapping between Linux users and SELinux users.

```
[root@loadproxy-cisco-com ~]# semanage login -l

Login Name           SELinux User         MLS/MCS Range        Service

__default__          unconfined_u         s0-s0:c0.c1023       *              *
root                 unconfined_u         s0-s0:c0.c1023       *
```

   **b.** As root, create a new Linux user (**nginx** user) that is mapped to the SELinux **user_u** user.

```
useradd -Z user_u nginxuser
[root@loadproxy-cisco-com ~]# passwd nginxuser
Changing password for user nginxuser.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

   **c.** In order to view the mapping between **nginxuser** and **user_u**, enter this command as root:

```
[root@loadproxy-cisco-com ~]# semanage login -l

Login Name           SELinux User         MLS/MCS Range        Service

__default__          unconfined_u         s0-s0:c0.c1023       *
nginxuser            user_u               s0                   *
root                 unconfined_u         s0-s0:c0.c1023       *
```

   **d.** SELinux **__default__** login is by default mapped to the SELinux **unconfined_u** user. By default, it is required to confine **user_u** by using the following command:

```
semanage login -m -s user_u -r s0 __default__
```

   In order to check if the command worked properly, enter `semanage login -l`. It should produce this output:

```
Login Name           SELinux User         MLS/MCS Range        Service

__default__          user_u               s0                   *
nginxuser            user_u               s0                   *
root                 unconfined_u         s0-s0:c0.c1023       *
```

464783

   **e.** Modify nginx.conf and perform change ownership for nginxuser.

1. Enter `chown -R nginxuser:nginxuser *` in the `<Openresty-install-directory>` directory.
2. Modify the **nginx.conf** file to include nginxuser as the user for running worker processes.

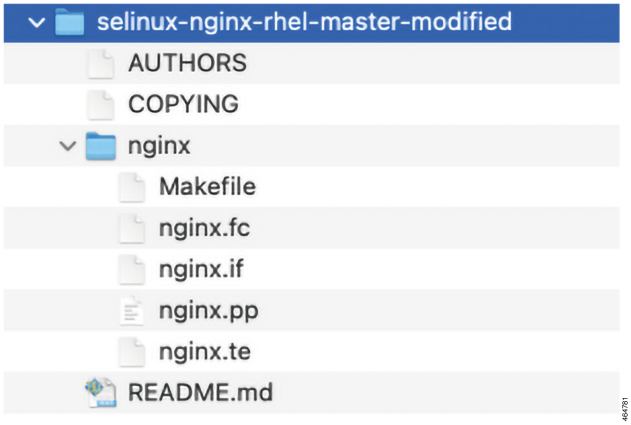
```
........

user nginxuser nginxuser;

..........
```

3. **Write the SELinux Policy for OpenResty Nginx**
   a. Instead of generating a new default custom policy for OpenResty Nginx with the `sepolicy generate --init /usr/bin/nginx` command, start with an existing policy.

   The **nginx.fc** file (File Contexts file) and **nginx.te** (Type Enforcement file) files, that are downloaded from the following location, are modified for reverse-proxy usage:




   This modified version can be used as a reference because it is updated for a particular use case.

   b. Download the **selinux-nginx-rhel-master-modified.tar** file from the Software Download .

   c. Extract the **.tar** file and navigate to the **nginx** directory within it.

   d. Open the **.fc** file and verify the required file paths of **Nginx installer**, **cache**, and **pid** files.

   e. Compile the configuration with the `make` command.

   f. The **nginx.pp** file is generated.

   g. Load the policy with the `semodule` command.

```
semodule -i nginx.pp
```

   h. Go to **/root** and create an empty file called `touch /.autorelabel`.

   i. Reboot the system.

   j. Enter the following command to verify that the policy is loaded successfully:

```
semodule --list-modules=full
```

```
[root@loadproxy-cisco-com ~]# semodule --list-modules=full
400 nginx               pp
200 container           pp
200 flatpak             pp
100 abrt                pp
100 accountsd           pp
100 acct                pp
100 afs                 pp
100 aiccu               pp
100 aide                pp
100 ajaxterm            pp
100 alsa                pp
```

464782

**k.** OpenResty Nginx should run without any violation. (Violation logs will be available in **/var/log/messages** and **/var/log/audit/audit.log**).

**l.** Enter the following command to check the status of OpenResty Nginx:

```
ps -aefZ | grep nginx
```

```
[root@loadproxy-cisco-com ~]# ps -aefZ |grep nginx
system_u:system_r:nginx_t:s0    root        1686       1  0 16:14 ?        00:00:00 nginx: master process /usr/bin/nginx
system_u:system_r:nginx_t:s0    nginxus+    1687    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1688    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1689    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1690    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1691    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1692    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1693    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1694    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1695    1686  0 16:14 ?        00:00:00 nginx: cache manager process
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 2543 2252  0 16:17 pts/0 00:00:00 grep --color=auto nginx
```

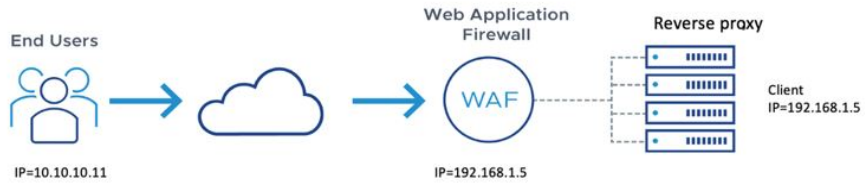
464784

**m.** Now the Finesse agent desktop or supervisor desktop should be accessible.

# Load Balancer, WAF, and Proxy support for reverse-proxy deployments

The reverse-proxy configurations have security features that are dependent on the information of the actual client IP. Requesting rate limits, logging of client activity, blocking the users due to multiple wrong credentials require the configuration to track the client IP to appropriately rate-limit, block, or log the actual users' access.

No specific configurations are required for deployments which directly terminate the agent connections on the reverse-proxy. The reverse-proxy has the information of the client IP due to the connections directly reaching the reverse-proxy. However, when other network devices are used to terminate the client connections, before forwarding them as fresh requests to the reverse-proxy, the client IPs are no longer visible to the reverse-proxy.



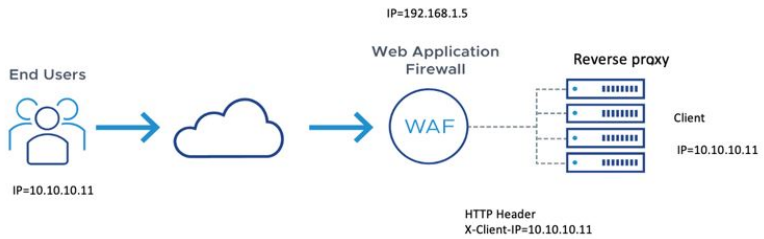
This happens when there are Load Balancers, Web Application Firewall (WAF), or other proxies deployed in front of a reverse-proxy. The CDN deployments work as an intermediary reverse-proxy/WAF and fall into the same deployment category.

Such deployments **MUST** add certain reverse-proxy configurations to enable the reverse-proxy to identify the actual client IP. The configurations that are required for such deployments are as follows:

1. The public IPs and private IPs of the devices which will forward the requests to the reverse-proxy must be to be added in **nginx.conf** http block and **maps.conf** geo block as mentioned in the **##Must-change** comment.
2. The new requests originating from the intermediary devices, **MUST** populate HTTP request header fields with the end-client IP to communicate the same to the reverse-proxy.

   The name of the request header field is configured in **nginx.conf** file using `real_ip_header` directive. For example: `real_ip_header X-REAL-IP;`

**Note** All CDN deployments provide a mechanism to extract the client IP as a HTTP header containing a single-client IP as part of the request payload. A custom header is often recommended to avoid conflict with the standard `X-FORWARDED-FOR` header. The VPN-less reverse-proxy deployments are also recommended to provide the client IP using a custom header for similar reasons.




3. For security purpose, the devices which are front ending the reverse-proxy **MUST** replace `X-FORWARDED-FOR` and `X-REAL-IP` headers provided by the client with the actual client IP or drop them if the deployment does not need these headers.
4. If the deployment is using a custom HTTP header for communicating to the client IP, the particular field **MUST** be replaced with the client IP before forwarding them upstream to the reverse-proxy.
5. Verify the configuration by transmitting a high rate of requests to a Finesse API such as `SystemInfo/DesktopConfig` from an external client. Verify through the Load Balancer or WAF to ensure that the client is blocked while the Load Balancer or intermediate devices are not blocked or rate limited. Ensure that the configurations are working as expected before going live.

   Refer to the Logging section for instructions on how to check whether a client is blocked or rate limited.
6. Deployments that employ WAF or other security devices must ensure that the desktop API traffic patterns are compatible with them before going live with the deployment. Certain WAF rules can be too restrictive and may need some modifications before they can be deployed.

**Note** The reverse-proxy configurations provided have no protection against layer-3 attacks such as IP address spoofing or flooding. The proxy provides only rate limiting, brute force attack detection, and restricting of requests to the allowed destinations. The operating system IP configurations are hardened to a certain level but there are no further protections that are available. It is assumed that the relevant operating system hardening and traffic protection devices are employed to secure the deployment Cisco Contact Center.

For more details refer to the *Security Guidelines for Reverse-Proxy Deployment* section in the *Security Guide for Cisco Unified ICM/Contact Center Enterprise, Release 12.6(1)* guide.

Load Balancers and other devices which does not have the HTTP header support can skip second and third points that are mentioned above. However, this causes a sub-optimal deployment which will be functional but loses certain features such as client IP logging for debugging purposes and blocking users attempting to brute force guess passwords.

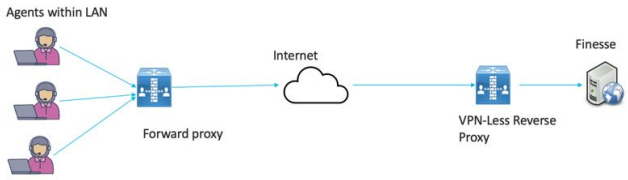
The websocket authentications will also be not effective at the reverse-proxy, which will not cause any loss in functionality but will allow all websocket request to reach the upstream component before authentication can be enforced.

### Related Topics

Logging, on page 226
Troubleshooting

## Access VPN-Less proxy through Forward proxy and NAT

The VPN-less configuration assumes that the proxy is accessed by clients/agents from the internet, who have separate individual IP's which can be used for enforcing security features. However, not all deployments dedicatedly use agents from the internet with their own unique IP addresses. Most deployments will have agents accessing the reverse-proxy deployments both from the internet as well as from LAN using the same reverse-proxy access URI.




So, if you have a deployment which uses agents behind a proxy or a NAT that looks like what is shown above, certain configuration changes have to be made to ensure that the end-user IP's are correctly communicated to the reverse-proxy. The steps to configure are as follows:

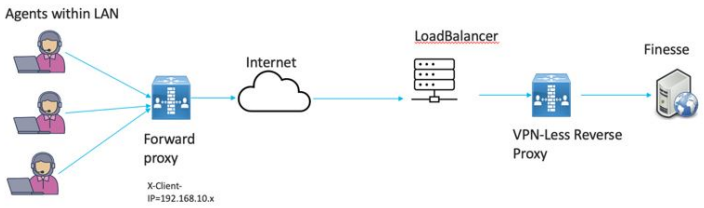
1. The Forward proxy (device A in the diagram above) has to be well-known in advance.
2. The Forward proxy device has to transmit the agent IP’s in a predefined header. For example, `X-REAL-IP` as shown above.
3. If there are other intermediary devices such as a Load Balancer or WAF at the network where Finesse is deployed, before the requests reach the reverse-proxy, these devices must be able to allow the Forward proxy by its IP address and then transmit the HTTP header without any changes.

**Note** It is crucial that the Forward proxy IP address is identified and only requests from this IP should be allowed to have the `X-REAL-IP` transmitted.

4. The **nginx.conf** http block and **maps.conf** geo block files should be updated with the list of Forward proxies' private and public IPs as mentioned in the **##Must-change** comments.

**Note** Deployments that do not have the HTTP header support can skip the steps 2 to 4. However, this causes a sub-optimal deployment which will be functional but loses certain security features listed above, which are dependent on client IP knowledge and these deployments are therefore not suggested.

Ensure that the deployment cannot support multiple HTTP header names to transmit the client IP corresponding to different Forward proxies that the network is interacting with.




Deployments such as these, should transmit or detect the final client IP of the users who are connecting from behind the **Forward proxy A** and this would be an agreement between Load Balancer and the Forward Proxy.

The Load Balancer or the final intermediary devices that forward requests to the VPN-less reverse-proxy should transmit the required headers and will need configuration as described in the section above. The Forward proxy information is not required to be added to the VPN-less configuration, if the intermediary device is able to identify the correct client IPs and transmit them to the reverse-proxy using the steps mentioned above.

However, if the actual client IP resolution is not setup between the Forward proxy and the Load Balancer, the reverse-proxy considers the IP of the Forward proxy as the actual client IP.

In this case, to avoid rate limiting to block the Forward proxy, its private and public IP addresses must be configured in **nginx.conf** http block and **maps.conf** geo block files. Both the files must be updated with the list of Forward-proxies' IP as mentioned in the **##Must-change** comments, so that the proxy is not blocked or rate limited. This would be a sub-optimal deployment and transmitting the actual client IP is recommended for a more effective deployment.

# Verifying Reverse-Proxy Configuration

## Finesse

**Procedure**

**Step 1** From the DMZ, open *https://<reverseproxy:port>/finesse/api/SystemInfo* and check if it's reachable.

**Step 2** Check if the **<host>** values in both **<primaryNode>** and **<secondaryNode>** are valid in reverse-proxy hostnames. It shouldn't be Finesse hostnames.

**Note**
- If CORS status is "enabled", you must explicitly add the reverse-proxy domain name to the list of CORS trusted domain names.
- Reverse-proxy supports a maximum of 8000 folders (including subdirectories) in the `finesse/3rdpartygadget` folder.

## Cisco Unified Intelligence Center and LiveData

**Procedure**

**Step 1** If the Finesse hostnames are seen in the response instead of reverse-proxy hostnames, validate the proxy-mapping configurations and check if the allowed hosts are properly added in Finesse servers as described in the section *Populate Network Translation Data*.

**Step 2** If LiveData gadgets load properly in Finesse Desktop, then CUIC and LiveData proxy configurations are proper.

**Step 3** To validate the Cisco Unified Intelligence Center and LiveData configurations, make the HTTP requests from the DMZ to the following URLs and check if they are reachable:

- https://<reverseproxy:cuic_port>/cuic/rest/about
- https://<reverseproxy:ldweb_port>/livedata/security
- https://<reverseproxy:ldsocketio_port>/security

## Cisco Identity Service

To validate Cisco IdS configuration, perform the following steps:

**Procedure**

**Step 1** Log in to the IdSAdmin interface at **https://<ids_LAN_host:ids_port>:8553/idsadmin** from the LAN because the admin interface is not exposed over reverse-proxy.

**Step 2** Choose **Settings** > **IdS Trust**.

**Step 3** Validate that the proxy cluster publisher node is listed on Download SP metadata page, and click **Next**.

**Step 4** Validate that the IDP proxy is correctly displayed if configured on Upload IDP metadata page, and click **Next**.

**Step 5** Initiate test SSO via all proxy cluster nodes from the Test SSO page and validate that all are successful. This requires client machine connectivity to reverse-proxy nodes.

# Brute Force Attack Prevention Configuration

Finesse 12.6 ES02 and above authentication scripts actively prevent brute force attacks that can be used to guess the user password. The scripts do this by blocking the IP address used to access the service, after a certain number of failed attempts in a short time. These requests will be rejected by **418 client error**. The number of failed requests, time interval, and blocking duration are configurable.

## Attack Detection Parameters

Configurations are present in the **<nginx-install-directory>/conf/conf.d/maps.conf** file.

```
## These two constants indicate five auth failures from a client can be allowed in thirty
seconds.
## if the threshold is crossed,client ip will be blocked.
map $host $auth_failure_threshold_for_lock {
## Must-change Replace below two parameters as per requirement
default 5 ;
}
map $host $auth_failure_counting_window_secs {
## Must-change Replace below two parameters as per requirement
default 30;
}
## This indicates duration of blocking a client to avoid brute force attack
map $host $ip_blocking_duration {
## Must-change Replace below parameter as per requirement
default 1800;
}
```

## Logging

The details of the blocked IP addresses can be accessed from the files **<nginx-install-directory>/logs/blocking.log** and **<nginx-install-directory>/logs/error.log**. To find the IP addresses that are blocked, run the following commands from the directory **<nginx-install-directory>/logs.**

```
grep "will be blocked for" blocking.log
grep "IP is already blocked." error.log
2021/10/29 17:30:59 [emerg] 1181750#1181750: *19 [lua] block_unauthorized_users.lua:153:
_redirectAndSendError(): 10.68.218.190
will be blocked for 30 minutes for exceeding retry limit., client: 10.68.218.190, server:
saproxy.cisco.com, request: "GET
```

```
/finesse/api/SystemInfo?nocache=1636456574482 HTTP/2.0", host: "saproxy.cisco.com:8445",
referrer:
"https://saproxy.cisco.com:8445/desktop/container/?locale=en_US&"

2021/10/29 19:21:00 [error] 943068#943068: *43 [lua] block_unauthorized_users.lua:53:
10.70.235.30 :: IP is already blocked...,
client: 10.70.235.30, server: saproxy.cisco.com, request: "GET
/finesse/api/SystemInfo?nocache=1635591686497 HTTP/2.0", host:
"saproxy.cisco.com:8445", referrer:
"https://saproxy.cisco.com:8445/desktop/container/?locale=en_US"
```

It is recommended that customers integrate with **Fail2ban** or a similar intrusion prevention system to add the blocked IP addresses to the IPtable or firewall rules.

# Install and Configure Fail2ban

Fail2ban can be configured to monitor the `blocking.log` to identify the IP addresses that are blocked by OpenResty Nginx on detecting brute force attacks, and ban the IP addresses for a configurable duration. Do the following to install and configure Fail2ban on a CentOS reverse-proxy:

### Procedure

**Step 1** Install Fail2ban using yum

```
yum update && yum install epel-release yum install fail2ban
```

**Step 2** Create a local jail

Jail configurations allow the administrator to configure various properties such as the ports that are to be banned from being accessed by any blocked IP address, the duration for which the IP address stays blocked, the filter configuration used for identifying the blocked IP address from the log file monitored, and so on. Steps to add a custom configuration for banning IP addresses that are blocked from accessing the upstream servers are as follows:

**a.** Go to Fail2ban installation directory (in this example /etc/fail2ban)

```
cd /etc/fail2ban
```

**b.** Make a copy of jail.conf into jail.local to keep the local changes isolated.

```
cp jail.conf jail.local
```

**c.** Add the following jail configurations to the end of the file jail.local, and substitute the ports in the template with the actual ones. Update ban time configurations as required.

```
# Jail configurations for HTTP connections.
[finesse-http-auth]
enabled = true
# The ports to be blocked. Add any additional ports.
port = http,https,<finesse-ports>,<cuic-ports>,<any-other-ports-to-be-blocked>
# Path to nginx blocking logs.
logpath = /usr/local/openresty/nginx/logs/blocking.log
# The filter configuration.
filter = finesseban
# Block the IP from accessing the port, once the IP is blocked by lua.
maxretry= 1
# Duration for retry set to 3 mins. Doesn't count as the maxretry is 1
findtime= 180
```

```
# Lock time is set to 3 mins. Change as per requirements.
bantime = 180
```

**Step 3** Configure a filter

A filter tells Fail2ban what to look for in the logs to identify the host to be banned. The steps to create a filter is as follows:

**a.** Create filter.d/finesseban.conf

```
touch filter.d/finesseban.conf
```

**b.** Add the following lines into the file filter.d/finesseban.conf

```
[Definition] # The regex match that would cause blocking of the host. failregex = <HOST>
will be blocked for
```

**Step 4** Start Fail2ban

Run the following command to start fail2ban:

```
fail2ban-client start
```

Open fail2ban log files and verify that there are no errors. By default, logs for fail2ban go into the file `/var/log/fail2ban.log`.

# Troubleshoot

## Troubleshoot SELinux

### Procedure

**Step 1** If OpenResty Nginx is not started by default or the Finesse Agent Desktop is not accessible, set SELinux to **permissive** mode with this command:

```
setenforce 0
```

**Step 2** Try to restart OpenResty Nginx with the **`systemctl restart nginx`** command.

**Step 3** All the violations will be available in **/var/log/messages** and **/var/log/audit/audit.log**.

**Step 4** You are required to regenerate the **.te** file with allow rules for addressing those violations by executing any one of the following commands:

- `cat /var/log/audit/audit.log | audit2allow -m nginx1 > nginx1.te. # this will create nginx1.te file`
- `ausearch -c 'nginx' --raw | audit2allow -M  my-nginx  # this will create my-nginx.te file`

**Step 5** Update the original **nginx.te** file present in the **selinux-nginx-rhel-master-modified/nginx** directory with the newly generated allow rules.

**Step 6** Compile the **nginx.te** file by using the **make** command.

**Step 7** The **nginx.pp** file is regenerated.

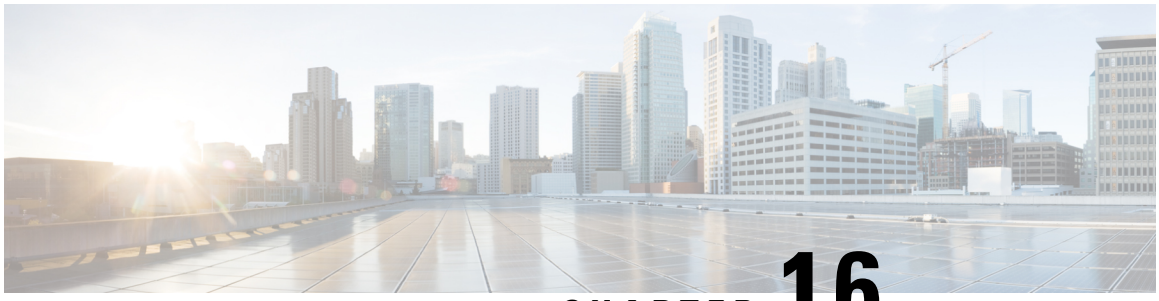
**Step 8** Load the policy by using the **semodule** command.

```
semodule -i nginx.pp
```

**Step 9** Change SELinux to **enforce** mode by using the **setenforce** command.

**Step 10** Reboot the system.

**Step 11** Repeat this procedure until all the violations are fixed.



CHAPTER 16

# Webex Experience Management Integration



## Experience Management Overview

Cisco Webex Experience Management is a Customer Experience Management (CEM) platform that allows you to see your business from your customers' perspective. To know more about Webex Expereince Management, see https://xm.webex.com/docs/ccoverview/.

With Webex Experience Management, Unified CCE supports:

- Customer experience surveys - Set up and send surveys to customers, after an interaction, to collect feedback about their interaction.
- Customer Experience Journey (CEJ) gadget - Displays all the past survey responses from a customer in a chronological list. The agent and supervisor use this gadget to gain context about the customers past experiences with the business and engage with them appropriately.
- Customer Experience Analytics (CEA) gadget - Displays the overall experience of the customer interaction with agents using industry-standard metrics such as NPS, CSAT, and CES or other KPIs tracked within Experience Management. This gadget is available for agents and supervisors.

## Experience Management Survey

Experience Management post-call survey is used to determine whether the customers are satisfied with their voice call experiences. You can configure Experience Management to initiate this survey when an agent disconnects from the caller. The survey can be done in three modes—voice, SMS, or email.

The CCE script enables or disables voice call survey for each call by testing for conditions and setting an expanded call variable that controls Experience Management. For example, the script can invoke a prompt

that asks callers whether they want to participate in a survey. Based on the caller's response, the script sets the expanded call variable that controls whether the call gets transferred to the voice call survey Dialed Number.

You can send post call survey links through email or SMS also. After every call, the customer is provided with a choice to participate in the survey and answer few questions over email or their phone. For more information on how to configure or to associate the survey, refer to the section Configure Unified CCE for Experience Management Voice, SMS and Email Survey, on page 235 .

**Note** Experience Management supports G.711 u-law and G.711 a-law codecs.

# Experience Management Task Flow

To enable Experience Management Post Call Survey in Cisco Unified CCE, follow this task flow:

### Procedure

**Step 1** Contact your Cisco representative to purchase Experience Management license. After the purchase, you need to provide relevant information about your organization to Experience Management Activation Team. To know more about the information that will be collected, see Prerequisites.

**Step 2** Experience Management Activation Team performs the following actions:

**a.** Creates accounts and provisions the same.

**b.** Creates default spaces and metric groups for your accounts. To know more about creating spaces, see Space Creation.

**c.** Creates standard questionnaires for Experience Management Post Call Survey and publishes the same. To know more about creating questionnaires, see Questionnaires.

**Step 3** After creating and provisioning the account, you will receive handover emails from the Experience Management Activation Team. The email contains credentials and other essential information for your account. To know more about provisioning details, see Handover.

**Step 4** Initially, Spaces and Widgets are created by the Experience Management provisioning team. To know more about the different default Widgets, how to export and derive meaningful insights from them, see Experience Management Gadgets.

To know how to configure additional Widgets in Experience Management, see Experience Management Gadgets.

**Step 5** Ensure that the Cloud Connect publisher and subscriber are installed. For more information, see the *Install Cloud Connect* section in *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-guides-list.html .

**Step 6** Provision Experience Management service using CLI on Cloud Connect. For more information, see Provision Experience Management Service on Cloud Connect, on page 233.

**Step 7** Configure Cloud Connect in Operations Console (NOAMP). For details see the section *Configure CVP Devices for Cloud Connect* in *Administration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/

c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

**Step 8** Configure Post Call Survey in CVP. For more information, see Configure Post Call Survey in CVP, on page 110.

**Step 9** Import the following certificates to the CVP Server:

- Cloud Connect certificate
- Experience Management certificate

For details, see the sections *Import Cloud Connect Certificate to Unified CVP Keystore* and *Import Experience Management Certificate to Unified CVP Call Server* in *Configuration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

**Step 10** Ensure that the threshold properties (in *ivr.properties* and *sip.properties* files) and proxy settings are configured in CVP for Experience Management. For details, see the section *Webex Experience Management Configuration* in *Configuration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

**Step 11** Configure Unified CCE Experience Management. For more information, see the topic Configure Unified CCE for Experience Management Voice, SMS and Email Survey, on page 235.

**Step 12** Configure Dialed Number and Call Type for Incoming Call and Experience Management post call survey routing script. For more information, see Configure Dialed Number and Call Type, on page 237.

**Step 13** Modify CCE scripts. For more information, see *Modify CCE Scripts for Experience Management Voice, SMS and Email Surveys* in *Scripting and Media Routing Guide for Cisco Unified ICM/Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-user-guide-list.html.

Associate the CCE script with the Call Type created in the previous step.

**Step 14** Provision Cloud Connect on Cisco Finesse. For more information, see the *Cloud Connect Server Settings* topic at the *Cisco Finesse Administration Guide*.

**Step 15** Add Experience Management gadgets into Finesse desktop layout. For more information, see Cisco Webex Experience Management Gadgets.

# Provision Experience Management Service on Cloud Connect

Before provisioning Experience Management service on Cloud Connect, ensure to setup and enable Cloud Connect. For more information, see the *Cloud Connect Administration* section in *Administration Guide for Cisco Unified Contact Center Enterprise*

Provision Experience Management service using the following CLI on Cloud Connect.

```
set cloudconnect cherrypoint config
```

For more commands related to Experience Management service, see the *Cloud Connect CLI* section in the *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide*.

The partner hosted module which is a part of Experience Management Invitations solution is required to send surveys to customers over emails and SMS.

For information about *Partner Hosted Module Architecture* refer to https://xm.webex.com/docs/cxsetup/guides/partnerarchitecture/

For information about how to provision the infrastructure required to deploy the partner hosted components of the Experience Management Invitations module, see https://xm.webex.com/docs/cxsetup/guides/partnerinfra/ .

For information about how to deploy the partner hosted components on the Experience Management Invitations module once the infrastructure is provisioned, see https://xm.webex.com/docs/cxsetup/guides/partnerdeployment/.

# Configuration Changes in Webex Experience Management

Any configuration changes done in the Webex Experience Management administration interface gets reflected in the on-premise Contact Center integrations after a maximum caching delay.

You observe the caching delay when:

- Modify the questionnaires.
  - Adding or removing questions.
  - Adding new stock prefill question.
  - Updating the tags associated with stock prefill questions. The specific tags that Contact Center integrations use.
- System configuration changes are made to the dispatch.

The maximum caching delay depends on the kind of configuration change. Because the on-premise systems cache, Webex Experience Management configuration, and the maximum delay depend on the cache invalidation period. In cache invalidation, the system invalidates or removes the data from the cache after a specific time interval.

### Types of Caches and Cache Invalidation

The different types of caches that are used in the on-premise systems are:

- **Prefill Cache** if any new tag is not in the cache it's updated. However, if any new questions get added with the tag it reflects only after a cache invalidation period of 24 hours. For example, `tag -> List of Questions`.

**Note** As the new questions are staff prefill questions, we do not expect it to change often. Hence a delay of 24 hours is acceptable.

- **All other caches** will be reflected after a cache invalidation period of 1 hour. For example, `questionnaire -> hashScheme dispatchTemplate -> questionnaireDispatchSettings (As obtained from settings API)`

The partner hosted module in the Experience Management Invitations solution maintains a cache with maximum delay of 1 hour for the Dispatches list, Settings, Delivery Policy, Active Questions, and Questionnaire APIs. For more information, see https://xm.webex.com/docs/cxsetup/guides/partnerarchitecture/#43-caching-mechanism.

# Configure Unified CCE for Experience Management Voice, SMS and Email Survey

Refer to the following procedures to enable the Experience Management voice, SMS and email survey:


- Configure Expanded Call Variables, on page 235
- Upload Audio Files for Questions in Experience Management, on page 237
- Configure Dialed Number and Call Type, on page 237
- Associate Survey to Call Type in Unified CCE Admin, on page 238


# Configure Expanded Call Variables

**Procedure**

**Step 1** In the **Configuration Manager** menu, select **Tools** > **List Tools** > **Expanded Call Variables List**. The **Expanded Call Variables List** window appears.

**Step 2** Click **Retrieve** to view the list of existing ECC variables.

**Step 3** Check if the *user.microapp.isPostCallSurvey* variable exists. If the variable does not exist, do the following to create a new variable:

a) Click **Add**.
b) In the **Attributes** tab that appears, enter *user.microapp.isPostCallSurvey* in the **Name** field.
c) Set **Max Length** to 1.
d) Check the **Enabled** check box.
e) Click **Save**.

When your CCE routing scripts starts, the Post Call Survey field is **enabled** by default even if *user.microapp.isPostCallSurvey* has not yet been set in the script. You can turn off Post Call Survey field in the script by setting *user.microapp.isPostCallSurvey* to *n*. You can later enable Post Call Survey in the same path of the script by setting this variable to *y*.

**Note** To enable Experience Management, *user.microapp.isPostCallSurvey* must be set to *y*.

**Step 4** Check if the `user.CxSurveyInfo` variable exists. If the variable does not exist, do the following to create a new variable:

a) Click **Add**.
b) In the **Attributes** tab that appears, enter `user.CxSurveyInfo` in the **Name** field.
c) Set the **Max Length** to 133 for Type 10 VRUs. For all other routing clients, set **Max Length** to 80.
d) Check the **Enabled** check box.

**Step 5** Click **Save**.

**Note** The newly created ECC variables are added to the default payload list. If you want to save the ECC variables to a different payload list, in the **Configuration Manager**, navigate to **Tools** > **List Tools** > **Expanded Call Variable Payload List** and add the ECC variables to the payload list of your choice.

**Step 6** Populate the **POD. ID** variable.

For more information on populating this variable, refer to the topic Configure POD.ID.

**Step 7** Restart the active VRU PG (side A or B) to register the new ECC variable.

If the ECC variable already exists, you can skip this step.

**Note** The **user.microapp.isPostCallSurvey** setting takes effect on Unified CVP only when it receives a **connect** or temporary connect message. If you do not want the survey to run, without first reaching an agent (such as 'after hours of treatment'), set the isPostCallSurvey to **n** before the initial 'Run script request'.

# Configure POD.ID

Cisco provided variables are predefined, but for POD.ID, the maximum length should be set to 120.

You can modify the variables only if you have the edit access.

Populate the value in the script with multiple attributes in a key-value pair format. Each key-value pair is seperated with a semi-colon. The following table displays the supported attributes:

***Table 8: Variables and their descriptions***

| Attribute | Description | Applicable |
|---|---|---|
| cc_CustomerId | Unique ID for a customer across multiple channels | Chat and Email surveys for Digital Channels |
| Email | Email ID of the caller for Email surveys | Email survey for voice channel |
| Mobile | Phone number for SMS surveys | SMS survey for voice channel |
| cc_language | Language of the survey<br>For the list of supported languages, refer to the Webex Experience Management documentation at https://xm.webex.com/docs/user/getting-help/#cloudcherry-language-support | Email, SMS, and Voice surveys for voice channel |
| Optin | Whether to opt in or opt out of the survey | Email, SMS, and Voice surveys for voice channel |

Example: `cc_CustomerId=xxx;Email=xx;Mobile=xxx;cc_langauge=xxx;Optin=yes/no`

For more information on **Expanded Call Context Variables**, see the chapter *Configure Variables* in the Configuration Guide for Cisco Unified ICM/Contact Center Enterprise at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-and-configuration-guides-list.html.

You can also configure POD.ID from CVP Call Studio. For more information, refer to the topic *Configure Call Studio App Data Format* in the Scripting and Media Routing Guide for Cisco Unified ICM/Contact Center Enterprise at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-user-guide-list.html.

# Upload Audio Files for Questions in Experience Management

Experience Management allows you to upload the audio files for post call survey.

**Note** To run post-call voice survey, you must either configure *Text-To-Speech(TTS)* in the voice browser or upload audio prompts in Experience Management.

Create and configure the questionnaires in Experience Management for sending IVR surveys to the customer. For more information on Experience Management, refer to https://xm.webex.com/docs/ccoverview/

For more information on how to create and modify the questionnaires, refer to https://xm.webex.com/docs/cxsetup/questionnaires/.

# Configure Dialed Number and Call Type

**Procedure**

**Step 1** In **Configuration Manager**, navigate to **Tools** > **List Tools** > **DialedNumber/Script Selector List**.

**Step 2** In **DialedNumber/Script Selector List** create **Incoming Dialed Number** and **Post Call Survey Dialed Number**.

For more information on how to create a PCS Dialed Number, refere to the section *Configure ICM for Post Call Survey* in *Configuration Guide for Cisco Unified Customer Voice Portal* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-installation-and-configuration-guides-list.html.

**Step 3** Click **Save**. You will be re-directed to the **Configuration Manager** window.

**Step 4** In **Configuration Manager**, navigate to **Tools** > **List Tools** >**Call Type List**.

**Step 5** In **Call Type List** create the **Call Type**. You should associate to the incoming call script and PCS script.

You should associate the incoming call type with the survey from CCEAdmin.

For more information, refer to the topic Associate Survey to Call Type in Unified CCE Admin, on page 238.

# Associate Survey to Call Type in Unified CCE Admin

You can associate the Call Type to the survey only if you have added **Cloud Connect** in the **Inventory** page and configured the survey in **Webex Experience Management** portal.

**Note** Only one survey can be associated to a Call Type.

Call Types are created and managed in **Configuration Manager** tool and the survey is associated using the **CCE Admin** tool.

### Procedure

**Step 1** In **Unified CCE Administration**, navigate to **Overview** > **Call Settings** > **Route Settings** > **Call Types**.

The list of all the **Call Types** are displayed.

For more information on Call Types, refer to the **Help** in **ConfigManager** > **List Tools** > **Call Types**.

**Step 2** Click on the **Call Type** which you want to associate to the Survey. Associate the survey with the last call type before the call is first connected to an agent.

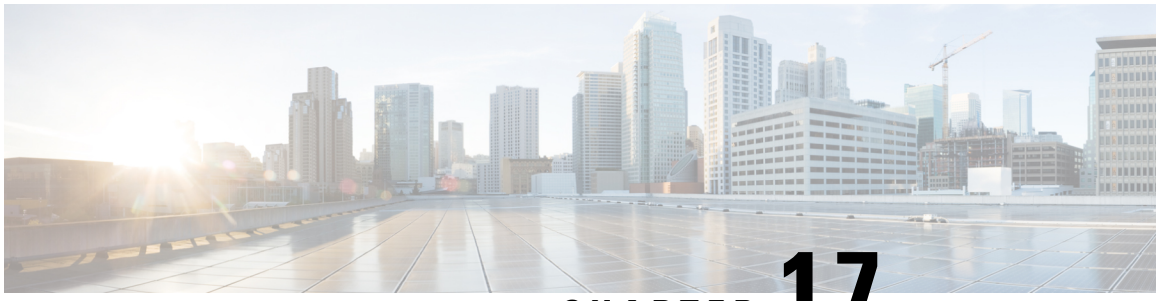
**Step 3** Select the **Enable Experience Management** check box to associate the **Webex Experience Management** survey.

The **Experience Management** tab is enabled with the following options:

- **Inline Survey** (post-call voice survey)
- **Deferred Survey** (post-call Email and SMS survey)

**Step 4** Click on the **magnifying glass** icon, and the configured surveys will be populated in the pop-up window.

**Step 5** Select the survey from the pop-up window and click **Save**.



CHAPTER 17

# Webex Experience Management Digital Channel Survey



## Overview

Digital Channel Survey is initiated when the agent responds to an email/chat from a customer using the Enterprise Chat and Email gadget. Cisco Webex Experience Management is a Customer Experience Management (CEM) platform that allows you to see the business from your customers perspective. It provides customer journey experience using the CEJ omni-channel gadget. To learn more about Webex Experience Management, see https://xm.webex.com/docs/ccoverview/.

With Webex Experience Management, Unified CCE supports:

- Customer experience surveys - Set up and send surveys to customers, after an interaction, to collect feedback about their interaction.
- Customer Experience Journey (CEJ) gadget - Displays all the past survey responses from a customer in a chronological list. The agent and supervisor use this gadget to gain context about the customers past experiences with the business and engage with them appropriately.
- Customer Experience Analytics (CEA) gadget - Displays the overall experience of the customer interaction with agents using industry-standard metrics such as NPS, CSAT, and CES or other KPIs tracked within Experience Management. This gadget is available for agents and supervisors.

## Digital Channel Survey

Email and chat inline surveys are used to determine whether customers are satisfied with their interaction with the agent in resolving their query over an email or chat. The feedback collected through the survey is used by the agents to gain context about the customer in their subsequent interactions and to also improve

their own performance. You can configure Enterprise Chat and Email to initiate this survey when the agent sends an email or terminates a chat conversation with a customer. The survey is sent inline in the agents email response to customers who contact them via email, and within the chat window for customers who contact them via chat.

# Digital Channel Survey Task Flow (Email/Chat)

To enable Experience Management inline surveys with Enterprise Email and Chat in Cisco Unified CCE, perform the following procedure:

### Procedure

**Step 1** Contact your Cisco representative to purchase Experience Management license. Provide relevant information about your organization to Experience Management Activation Team. To know more about the information that will be collected, see Prerequisites.

**Step 2** Experience Management Activation Team performs the following actions:

a) Creates account and provisions the same.

b) Creates default spaces and metric groups for your accounts. To know more about creating spaces, see Space Creation.

c) Creates default questionnaires in Expereince Management suited for inline email and chat survey. To know more about creating your own questionnaires or editing the default ones, see Questionnaires.

**Step 3** After creating and provisioning the account, you will receive handover emails from the Experience Management Activation Team. The email contains credentials and other essential information for your account. To know more about provisioning details, see Handover.

**Step 4** Initially, Spaces and Widgets are created by the Experience Management provisioning team. To know more about the different default Widgets and how to export and derive meaningful insights from them, see Cisco Webex Experience Management Gadgets.

To know how to configure other Widgets in Experience Management, see Basic Widget and Composite Widgets.

**Step 5** Ensure that the Cloud Connect publisher and subscriber are installed. For more information, see the *Install Cloud Connect* section in *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-guides-list.html

**Step 6** Provision Experience Management service using CLI on Cloud Connect. For more information, see Provision Cloud Connect for Digital Channel Survey, on page 241.

**Step 7** Ensure that the Enterprise Chat and Email (ECE) is installed and configured, see the *Webex Experience Manager Integration* in *Enterprise Chat and Email Administrator's Guide to Administration Console* at https://www.cisco.com/c/en/us/support/contact-center/enterprise-chat-email-12-5-1/model.html.

**Step 8** Configure Unified CCE Experience Management integration. For more information, see Configure Unified CCE for Digital Channel Survey , on page 241.

**Step 9** Configure Call Type and Dialed Number. For more information, see Configure Call Type, Dialed Number, and Survey Association, on page 243.

**Step 10** Modify CCE Scripts. For more information, see *Modify CCE Scripts for Experience Management Digital Channel Surveys* in *Scripting and Media Routing Guide for Cisco Unified ICM/Contact Center Enterprise*

at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-user-guide-list.html.

Associate the CCE script with the Call Type created in the previous step.

**Step 11** Provision Cloud Connect on Cisco Finesse. For more information, see the *Cloud Connect Server Settings* topic at the *Cisco Finesse Administration Guide*.

**Step 12** Add Experience Management gadgets into Finesse desktop layout. For more information, see Cisco Webex Experience Management Gadgets.

# Provision Cloud Connect for Digital Channel Survey

Before provisioning Cloud Connect for Experience Management service, ensure to setup and enable Cloud Connect. For more information, see the *Cloud Connect Administration* section in *Administration Guide for Cisco Unified Contact Center Enterprise*

**Note** Ensure that you have installed the self-signed certificates for Cloud Connect. For more information, see the *Self-Signed Certificates* section in the *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide*.

Provision Experience Management service using the following CLI on Cloud Connect.

```
set cloudconnect cherrypoint config
```

For more commands related to Experience Management service, see the *Cloud Connect CLI* section in the *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide*.

# Configure Unified CCE for Digital Channel Survey

Refer to the following procedures to enable the Experience Management email and chat survey:

- Configure Expanded Call Variables , on page 241
- Configure Call Type, Dialed Number, and Survey Association, on page 243
- Associate Survey to Call Type in Unified CCE Admin, on page 244

# Configure Expanded Call Variables

**Procedure**

**Step 1** In the **Configuration Manager** menu, select **Tools** > **List Tools** > **Expanded Call Variable List**. The **Expanded Call Variable List** window appears.

**Step 2** Click **Retrieve** to view the list of existing ECC variables.

**Step 3** Check if the *user.microapp.isPostCallSurvey* variable exists. If the variable does not exist, do the following to create a new variable:

a) Click **Add**.
b) In the **Attributes** tab that appears, enter *user.microapp.isPostCallSurvey* in the **Name** field.
c) Set **Max Length** to 1.
d) Check the **Enabled** check box.
e) Click **Save**.

When your CCE routing scripts starts, you can turn off Post Call Survey field in the script by setting *user.microapp.isPostCallSurvey* to *n*. You can later enable Post Call Survey in the same path of the script by setting this variable to *y*.

**Note** In the script, set the *user.microapp.isPostCallSurvey* before routing it to the agent.

**Note** To enable Experience Management, *user.microapp.isPostCallSurvey* must be set to *y*.

**Step 4** Check if the *user.CxSurveyInfo* variable exists. If the variable does not exist, do the following to create a new variable:

a) Click **Add**.
b) In the **Attributes** tab that appears, enter `user.CxSurveyInfo` in the **Name** field.
c) Set **Max Length** to 80.
d) Check the **Enabled** check box.

**Step 5** Click **Save**.

**Note** The newly created ECC variables are added to the default payload list. If you want to save the ECC variables to a different payload list, in the **Configuration Manager**, navigate to **Tools** > **List Tools** > **Expanded Call Variable Payload List** and add the ECC variables to the payload list of your choice.

**Note** You can use several ECC payloads in the same call flow, but only one ECC payload has scope at a given moment. For more information, see *ECC Payloads* sections in *Configuration Guide for Cisco Unified ICM/Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-and-configuration-guides-list.html

**Step 6** Populate the **POD. ID** variable.

For more information on populating this variable, refer to the topic Configure POD.ID , on page 243.

**Step 7** Restart the active VRU PG (side A or B) to register the new ECC variable.

If the ECC variable exists, you can skip this step.

**Note** The **user.microapp.isPostCallSurvey** setting takes effect on Unified CVP only when it receives a **connect** or temporary connect message. If you don't want the survey to run, without first reaching an agent (such as 'after hours of treatment'), set the isPostCallSurvey to **`n`** before the initial 'Run script request'.

# Configure POD.ID

Cisco provided variables are predefined, but for POD.ID, the maximum length should be set to 120. Enable the POD.ID variable to edit its length.

You can modify the variables only if you have the edit access.

Populate the value in the script with multiple attributes in a key-value pair format. Each key-value pair is seperated with a semi-colon. These attributes are sent to the Webex Experience Management as prefills when ECE initiates the survey. The following table displays the supported attributes:

***Table 9: Variables and their descriptions***

| Attribute | Description | Applicable |
|---|---|---|
| cc_CustomerId | Unique ID for a customer across multiple channels | Chat and Email surveys for Digital Channels |
| Email | Email ID of the customer for Email survey across multiple channels | Chat and Email surveys for Digital Channels |
| Mobile | Phone number for Chat surveys | Chat and Email surveys for Digital Channels |

Example: `cc_CustomerId=xxx;Email=xx;Mobile=xxx;`

For more information on setting the ECC variables used in the example, see *Modify CCE Scripts for Experience Management Digital Channel Surveys* in *Scripting and Media Routing Guide for Cisco Unified ICM/Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-user-guide-list.html.

For more information on **Expanded Call Context Variables**, see the chapter *Configuring Variables* in the Configuration Guide for Cisco Unified ICM/Contact Center Enterprise at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-and-configuration-guides-list.html.

# Configure Call Type, Dialed Number, and Survey Association

**Procedure**

**Step 1** In ICM Configuration Manager, navigate to **Tools** > **List Tools** > **Call Type List** to create a Call Type.

**Step 2** Associate the survey with the last call type before the email/chat is handled by the agent.

For more information, refer to the topic Associate Survey to Call Type in Unified CCE Admin, on page 244.

**Step 3** Create a Dial Number in ICM Configuration Manager and associate it with Call type (created in Step 1) for email and chat.

# Associate Survey to Call Type in Unified CCE Admin

You can associate the survey to the Call Type only if you have added **Cloud Connect** to the **Inventory** page in **CCE Admin** and configured the survey in **Webex Experience Management** portal.

**Note** Only inline surveys can be associated to a Call Type associated with digital channels.

Call Types are created and managed in **Configuration Manager** tool and the survey is associated using the **CCE Admin** tool.

### Procedure

**Step 1** In **Unified CCE Administration**, navigate to **Overview** > **Call Settings** > **Route Settings** > **Call Type**.

The list of all the **Call Type** are displayed.

For more information on Call Types, refer to the **Help** in **Configuration Manager** > **Tools** > **List Tool** > **Call Type List**.

**Step 2** Click on the **Call Type** which you want to associate to the Survey.

**Step 3** Select the **Enable Experience Management** check box to associate the **Webex Experience Management** survey.

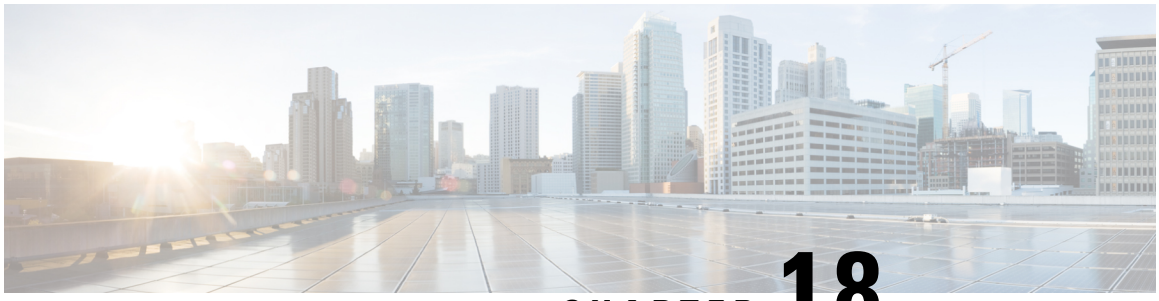
a) The Experience Management tab is enabled with the following options:

- Inline Survey
- Deferred Survey

**Step 4** Select **Inline Survey** for email and chat.

- Click on the **magnifying glass** icon, and the configured surveys will be populated in the pop-up window.

**Step 5** Select the survey from the pop-up window and click **Save**.



CHAPTER 18

# Whisper Announcement



## Capabilities

Whisper Announcement plays a brief, prerecorded message to an agent just before the agent connects with each caller. The announcement plays only to the agent; the caller hears ringing (based on existing ring tone patterns) while the announcement plays.

The content of the announcement can contain information about the caller that helps prepare the agent to handle the call. The information can include caller language preference, choices the caller made from a menu (Sales, Service), customer status (Platinum, Gold, Regular), and so on.

After Whisper Announcement is enabled, the played announcements are specified in the call routing scripts. The determination of which announcement to play is controlled in the script and is based on various inputs, such as the dialed number, a customer ID look up in your customer database, or selections you made from a VRU menu.

### Functional Limitations

Whisper Announcement is subject to these limitations:

- Announcements do not play for outbound calls made by an agent. The announcement plays for inbound calls only.
- For Whisper Announcement to work with agent-to-agent calls, use the SendToVRU or TranslationRouteToVRU node before you transfer the call to the agent. Transfer the call to Unified CVP before you transfer the call to another agent. Then, Unified CVP can control the call and play the announcement, regardless of which node transfers the call to Unified CVP.
- Announcements do not play when the router selects the agent through a label node.
- CVP Refer Transfers do not support Whisper Announcement.
- Whisper Announcement supports Silent Monitoring with this exception: For Unified Communications Manager-based Silent Monitoring, supervisors cannot hear the announcements themselves. The supervisor desktop dims the Silent Monitor button while an announcement plays.

- Only one announcement can play for each call. While an announcement plays, you cannot put the call on hold, transfer, or conference; release the call; or request supervisor assistance. These features become available again after the announcement completes.
- The codec settings for Whisper Announcement recording and the agent's phone must match. For example, if Whisper Announcement is recorded in G.711 ALAW, the phone must also be at G.711 ALAW. If Whisper Announcement is recorded in G.729, the phone must support or connect using G.729.
- Forking happens in Gateway in NBR only when a caller is connected to the agent (with two-way audio). Whisper announcement is played only with one way audio with agent (before connecting to the caller).
- In an IPv6-enabled environment, Whisper Announcement might require extra Media Termination Points (MTPs).

# Deployment Tasks

The following list shows the high-level tasks that are required to deploy Whisper Announcement. Individual steps are covered in more detail in later sections.

1. Ensure your deployment meets the baseline requirements for software, hardware, and configuration described in the System Requirements and Limitations section. See the *Solution Design Guide for Cisco Unified Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-implementation-design-guides-list.html.
2. Create Whisper Announcement Audio Files, on page 246.
3. Deploy Whisper Announcement Audio Files to Media Server, on page 247.
4. Configure Whisper Service Dialed Numbers, on page 247.
5. Add Whisper Announcement to Routing Scripts, on page 249.
6. Fail-Safe Timeout for Whisper Announcement in Unified CCE, on page 251.

Example scripts that enable Whisper Announcement are installed with your system. For information about these scripts and how to access them, see Whisper Announcement Sample Scripts, on page 252.

# Create Whisper Announcement Audio Files

You must create audio files for each different Whisper Announcement you want to use on your system; for example, “Sales, English” or “Soporte Técnico en Español.” Create the files using the recording tool of your choice.

When recording your files, follow these rules:

- The media files must be in wave (.wav) format. Your wave files must match Unified CVP encoding and format requirements (G729, CCITT G.711 A-Law and U-law 8 kHz, 8 bit, mono).
- To avoid cutting off files when they are played, make sure they do not exceed the Whisper Announcement play limit (15 seconds).
- Test your audio files. Ensure that they are not cut off and that they are consistent in volume and tone.

- To reduce the likelihood of scripting errors, decide ahead of time on a file-naming convention that is easy for you and others to remember. For example, en_sales.wav, sp_support.wav.

# Deploy Whisper Announcement Audio Files to Media Server

Deploy your whisper audio files to your Unified CVP media server using whatever file-transfer method you prefer. The most important consideration is where on the server to place the files. HTTP requests for media server audio files are constructed as

`http://<media_server>/<locale_directory>/<application_directory>/<file_name>.`

The CVP defaults for the locale and application directories are `en-us/app`. Unified CCE automatically adds `en-us/app` to the server name when making HTTP requests for media files.

For example, if:

- The script node that defines the media server has a value of `http://myserver.mydomain.com` and
- The script node that defines the audio file to play has a value of `en_sales.wav`

Then the HTTP request for the file is automatically constructed as

`http://myserver.mydomain.com/en-us/app/en_sales.wav`

If you store your files in a different locale and application directory, your routing scripts must include variable nodes that define those alternate locations. Make note of the directories in which you place your files and communicate the locations to your script authors.

Make sure that the directories in which you deploy your files have the appropriate permissions to allow Read access.

**CVP with the Streaming Audio (Helix) and Whisper Announcement**

You must set the **user.microapp.media_server** variable, to point to the whisper announcement .wav file, for the CVP Whisper Announcement feature to work while Streaming Audio feature (using Helix) is also on. This is achieved by setting the **Call.WhisperAnnouncement** variable to the complete URL of the whisper announcement wav file. The **Call.WhisperAnnouncement** variable should be put in using the `http://<VXMLserverip>:7000/CVP/audio/XXX.wav` URL format.

## Using a Default Media Server

Optionally, CVP lets you define a default media server. (You do this in the CVP Operations Console; see your CVP documentation for more information.) If a default media server is defined in CVP, script authors need not identify the media server in their call routing scripts provided the files that they request are available from that server.

# Configure Whisper Service Dialed Numbers

For Whisper Announcement, Unified CVP uses two different dialed numbers when transferring a call to an agent:

- The first number calls the ringtone service that the caller hears while the whisper plays to the agent. The CVP default for this number is 91919191.
- The second number calls the whisper itself. The Unified CVP default for this number is 9191919100.

**Note** Whisper Announcement dialed number is always an extension of the Ringtone dialed number with an extra two zeros at the end.

For Whisper Announcement to work, your dial plan must include both of these numbers. The easiest way to ensure coverage is through the use of wild cards such as 9191*.

## Configure Dialed Numbers

You configure the dialed numbers for Whisper Announcement in the Unified CVP Operations Console at **System** > **Dialed Number Pattern** > **Add new**. For the Dialed Number Pattern Types, select **Enable Local Static Route**. Once **Enable Local Static Route** is checked, select either **Route to Device** or **Route to SIP Server Group** for VXML gateways. Then save and deploy the dialed number.

It may be necessary to override the dialed number plan for the default Whisper DN, if the default DN conflicts with the overall dial number plan.

### Change the Whisper Announcement Default Dialed Number

To override the DN pattern from the SIP subsystem level in CVP OAMP:

**Procedure**

**Step 1** Select **Device Management > Unified CVP Server**.

**Step 2** Select the Call Server on which to override the default whisper DN.

**Step 3** Select the SIP tab.

**Step 4** Override the default value of 91919191 configured under the **DN on the Gateway to play the ringtone** field.

**Step 5** Click **Save & Deploy**.

## Configure Ringtone Dialed Number

To configure the Ringtone dialed number in the CVP Operations Console:

1. Select **Device Management** > **Unified CVP Server**.
2. Select the Call Server on which you want to configure the settings.
3. Select the SIP tab.
4. In the **DN on the Gateway to play the ringtone** field, configure the default Ringtone dialed number Pattern.

### Dialed Number in the Dial-Peer

In addition to configuring the dial plan in Unified CVP, examine your IOS dial-peer. Make sure that the dialed number setting in your dial-peer configuration accommodates both of the whisper service dialed numbers.

# Add Whisper Announcement to Routing Scripts

To enable Whisper Announcements, use the Script Editor to modify your routing scripts as follows:

- Specify the WhisperAnnouncement call variable
- Specify the Unified CVP media server and location of whisper audio files
- Specify other required variables

For more information, see Whisper Announcement Sample Scripts, on page 252.

## Specify WhisperAnnouncement Call Variable

To include Whisper Announcement in a script, insert a Set Variable node that references the WhisperAnnouncement call variable. The WhisperAnnouncement variable causes a whisper to play and specifies the audio file it should use. Typically, you use a single whisper prompt for a single call type. As a result, you use only one WhisperAnnouncement set node for each script. However, as needed, you can set the variable at multiple places in your scripts to allow different announcements to play for different endpoints. For example, for skills-based routing, you can specify the variable at each decision point used to select a particular skill group or Precision Queue.

**Note** Only one Whisper Announcement can play for each call. If a script references and sets the WhisperAnnouncement variable more than once in a single path through a script, the last value to be set is the one that plays.

Use these settings in the Set Variable node for Whisper Announcement:

- Object Type: Call.
- Variable: Must use the WhisperAnnouncement variable.
- Value: Specify the filename of the whisper file. For example: “my_whisper.wav” or “my_whisper”.
  - Specify the filename only, not its path.
  - You must enclose the filename in quotation marks.
  - The filename is not case sensitive.
  - The filename cannot include spaces or characters that require URL encoding.
  - The .wav extension is optional. If you omit it, Unified CVP adds it automatically in the HTTP request.

## Specify Unified CVP Media Server Information

If you define a default media server in your CVP Operations Console and it is the server from which you serve your whisper files, then you need not specify the media server in your routing scripts. However, if you do not define a default media server, or if you store your whisper file on a server other than the default, then your scripts must include a Set Variable node that identifies that server.

To specify your media server, use the following settings in the Set Variable node:

- Object Type: Call.
- Variable: Must use the user.microapp.media_server ECC variable.
- Value: Specify the HTTP path to the server. For example: “http://myserver.mydomain.net.” You must enclose the path in quotes.
- Alternately you can specify an IP address in place of a DNS. Include the listening port number if the media server web server listens on a port other than 80 (for HTTP) or 443 (for HTTPS).

## Specify Whisper File Locale and Application Directories

CVP uses a default storage directory for media files: <web_server_root>/en-us/app. To take advantage of this, Unified CCE call routing scripts automatically add “en-us/app,” to the server name when constructing HTTP requests for media files. For example:

- If the script node that defines the media server has a value of “http://myserver.mydomain.com” and...
- The script node that defines which audio file to play has a value of “en_sales.wav,” then...
- The HTTP request for the file is automatically constructed as

http://myserver.mydomain.com/en-us/app/en_sales.wav

If your whisper audio files are stored in a different locale directory, you must add a Set Variable node to your script that identifies the locale directory. Similarly, if your whisper files are stored in a different application directory, you must add a Set Variable node that identifies that directory.

### *Specify Locale Directory*

Use these settings in the Set Variable node to specify your locale directory:

- Object Type: Call.
- Variable: Must use the user.microapp.locale ECC variable.
- Value: Specify the directory name. For example: “pt-br,” You must enclose the path in quotes.

### *Specify Application Directory*

Use these settings in the Set Variable node to specify your application directory:

- Object Type: Call.
- Variable: Must use the user.microapp.app_media_lib ECC variable.
- Value: Specify the directory name. For example: to use a directory “wav_files” in place of the default directory “app,” enter “wav_files.” To use a sub-directory “wav_files” “app,” enter “app/wav_files.” You must enclose the path in quotes.

### *Variable Length for Media Server Locale and Application Directory Variables*

If you do include Set Variable nodes for the media server, locale, or application directories, ensure that the values you set for them do not exceed the Maximum Length settings for their corresponding ECC variables.

For example, if you include a Set Variable node for the media server with a value of “http://mysubdomain.mydomain.co.uk,” the string is 33 characters long. Therefore, the Maximum Length setting for the user.microapp.media_server ECC variable must be 33 or greater. If it is not, you must increase

the Maximum Length setting. Otherwise, the server name is truncated in the HTTP request for the file and the file is not found. You configure ECC variables in the Unified CCE Configuration Manager at List Tools > Expanded Call Variables List.

## Test Whisper Announcement File Path

To test the path to the whisper file that you defined in you script variables, enter the complete URL into a browser. The .wav file should play. For example:

- If your script includes: default media server + default locale + default application directory + whisper.wav, then the path is “http://<default_media_server>/en-us/app/whisper.wav”
- If your script includes: http://my_server.my_domain.com + default locale + “app/wav_files” + whisper.wav, then the path is “http://my_server.my_domain.com/en-us/app/wav_files/whisper.wav”

## Other Script Settings That Are Required for Whisper Announcement

These additional settings are required for Whisper Announcement to work:

- Enable Target Requery on all script nodes that follow the WhisperAnnouncement variable and target an agent. These include Queue (to Skill Group or Precision Queue), Queue Agent, Route Select, and Select. If Target Requery is not enabled, the Whisper Announcement does not play.
- When you run an agent transfer or a conference script, use a SendToVRU, a TranslationToVRU, or a Run Script Request node before you target an agent.

# Fail-Safe Timeout for Whisper Announcement in Unified CCE

Unified CVP sends one message to Unified CCE each time a Whisper Announcement begins and a second message when the announcement ends. The time stamps from these messages are used to calculate Whisper Announcement data in Unified CCE reports.

If Unified CVP fails to send a Whisper Announcement end message toUnified CCE, the following occurs:

- Unified CCE cannot accurately calculate the whisper length, thus skewing report data.
- The agent cannot control the call (for example, put it on hold or transfer it) because these controls are disabled while a Whisper Announcement is playing.

To prevent this, Unified CCE has a Whisper Announcement timeout setting. The value for this setting represents the maximum Whisper Announcement play time that Unified CCE uses to calculate its report data.

The default is 20 seconds. This default is based on the default Whisper Announcement play time (specified in Unified CVP) of 15 seconds. The extra 5 seconds in the Unified CCE fail-safe timeout is a buffer against latency. If you modify the maximum Whisper Announcement play time in Unified CVP, modify the Unified CCE Whisper Announcement fail-safe timeout accordingly.

The Unified CCE Whisper Announcement fail-safe timeout value should be equal to or greater than the maximum Whisper Announcement play time setting in Unified CVP. Otherwise, Whisper Announcement play time in Unified CCE reports are under-reported.

To change the fail-safe timeout value, complete the following steps for theUnified CCE peripheral by using the PG explorer tool:

**Procedure**

**Step 1** In Unified CCE Configuration, select **Tools** > **Explorer Tools** > **PG Explorer**.

**Step 2** Click **Retrieve** to return a list of PGs (Peripheral Gateways).

**Step 3** Double-click the agent PG to expand it, and select the peripheral with client type **CUCM** or **UCCE system**.

**Step 4** On the **Peripheral** tab, enter the following text in the **Configuration Parameters** field:

```
/WHSTMOUT <value in seconds>
```

**Step 5** Once you are finished, click **Save**.

# Whisper Announcement Sample Scripts

Unified CCE includes sample routing scripts that demonstrate Whisper Announcement. You can use them as learning tools and as models for your own Whisper Announcement scripts. They are the following:

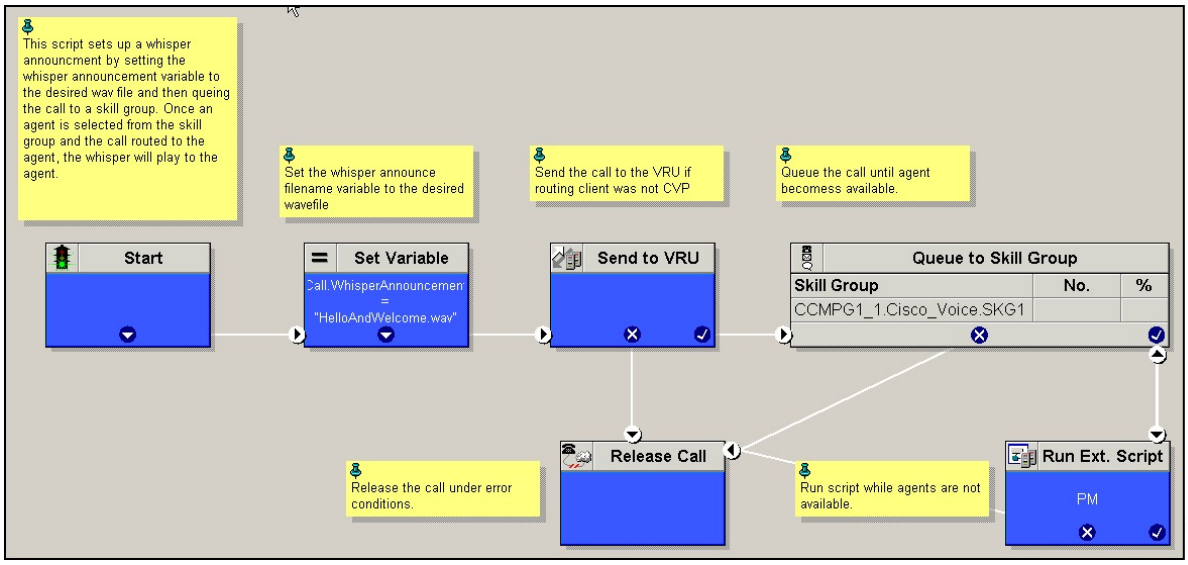
- **WA.ICMS**—This script plays a Whisper Announcement.
- **WA_AG.ICMS—**This script plays both a Whisper Announcement and an Agent Greeting to play on the same call flow.

The script files are located in the `c:\icm\bin` directory. In Unified CCE Script Editor, they are installed to the application root directory.

**Note** To use these scripts you must have a default media server configured in Unified CVP, and have the Whisper file stored in the default location on the media server. For that reason, they do not include variables that specify the media server, locale, or application directories.

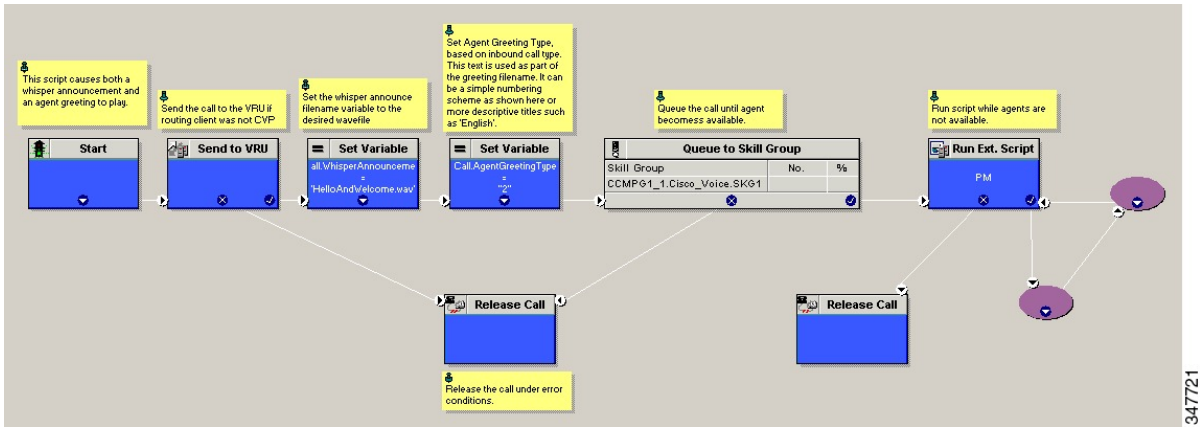
## WA.ICMS Script

This script sets up a Whisper Announcement by setting the Whisper Announcement variable to the desired wave file and then queuing the call to a skill group or Precision Queue. After an agent is selected from the skill group or Precision Queue and the call routed to the agent, the whisper plays to the agent.




## WA_AG.ICMS Script

This script causes both a Whisper Announcement and an Agent Greeting to play.




## Import Sample Whisper Announcement Scripts

To view or use the sample Whisper Announcement scripts, you must first import them into Unified CCE Script Editor. Follow this procedure to import the scripts:

### Procedure

**Step 1** Open Script Editor.

**Step 2** Select **File** > **Import Script** and select the first of the two scripts to import.

In addition to importing the script, Script Editor tries to map imported objects. Some objects that are referenced in the sample scripts, such as the external Network VRU scripts or the skill groups or Precision Queues, do not map successfully. You must create these maps manually or change these references to point to existing Network VRU scripts, skill groups, and Precision Queues in your system.

**Step 3** Repeat steps 2 and 3 for the remaining script.

# How Whisper Announcement Works

## Whisper Announcement Audio File

You store and serve your Whisper Announcement audio files from the Cisco Unified Contact Center Enterprise (Unified CCE) media server. This feature supports only the wave (`.wav`) file type. The maximum play time for a Whisper Announcement is subject to a timeout. Playback terminates at the timeout regardless of the actual length of the audio file. The default timeout is 15 seconds. In practice, you may want your messages to be much shorter than that, 5 seconds or less, to shorten your call-handling time.

### While a Whisper Announcement Is Playing

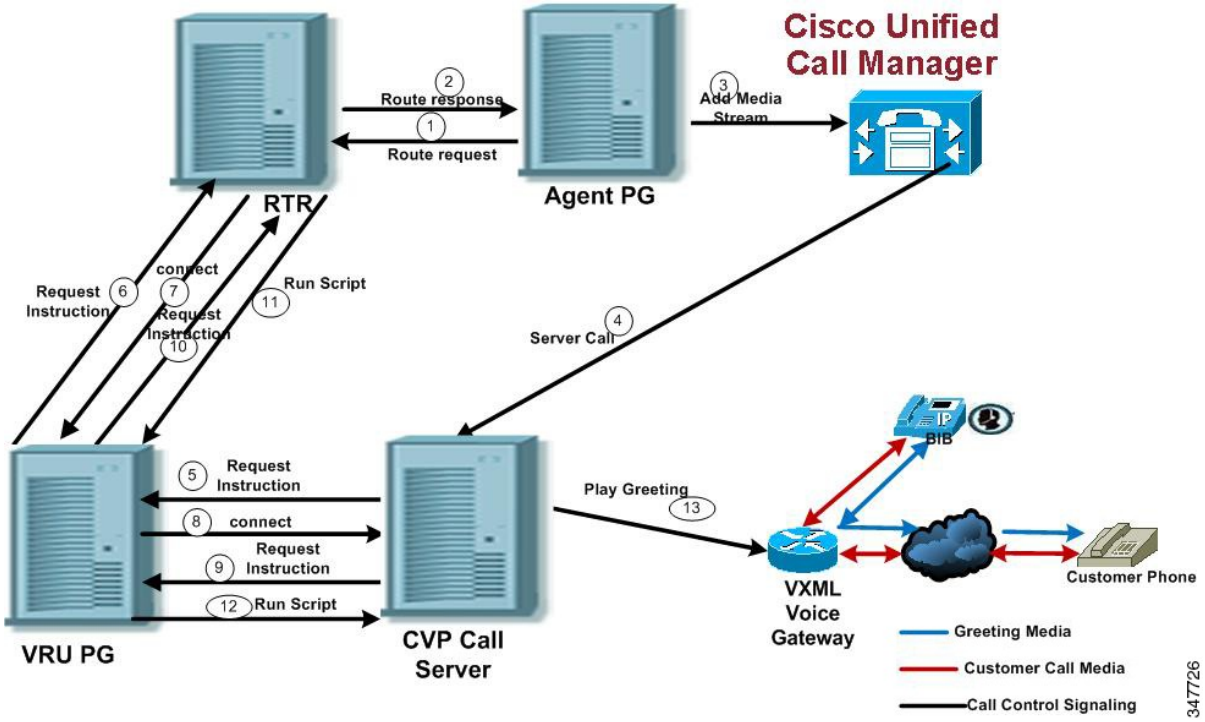
Only one Whisper Announcement can play for each call. While a Whisper Announcement is playing, you cannot put the call on hold, transfer, conference, or release the call, or request supervisor assistance. These features become available again after the whisper is complete.

### Whisper Announcement with Transfers and Conference Calls

When an agent transfers or initiates a conference call to another agent, the second agent hears an announcement if the second agent's number supports Whisper Announcement. In the case of consultative transfers or conferences, while the whisper plays, the caller hears whatever generally plays during hold. The first agent hears ringing. In the case of blind transfers, the caller hears ringing while the whisper announcement plays.

### Whisper Announcement Call Flow

Following is a Whisper Announcement call flow diagram accompanied by a description of the steps.




1. CVP receives a new call from the PSTN.

2 - 3. CVP sends the new call to the VRU PIM and the VRU PIM sends the new call to the Unified CCE router.

4. If an agent is available, the router reserves the agent.

5 - 6. The router sends a label with a whisper prompt to CVP.

7. CVP sends the call to Unified CM.

8 - 9. The agent receives and answers the call.

10. Unified CM sends the established event to the agent PIM. The agent PIM holds the event until the Whisper Announcement is done playing.

11. CVP tells the VXML gateway to play ringback to the caller and the Whisper Announcement to the agent. After the Whisper Announcement plays, CVP connects the agent to the customer and notifies Unified CCE.

12. The agent PIM gets notification that Whisper Announcement is complete and sends the established event to the agent desktop.

## Reporting and Serviceability

Whisper time is not specifically broken out in Unified CCE reports. In agent, skill group, and Precision Queue reports, the period during which the announcement plays is reported as Reserved agent state time. In the Termination Call Detail records, it is treated as Ring Time.

Serviceability for Whisper Announcement includes system events to indicate reasons for Whisper Announcement failures and counters to track the number of failed whisper events.

## Component Failure and Whisper Announcement

*Failure to Access CVP Media Server*

If the connection to the CVP media server fails, or if a requested whisper audio file cannot be found, the call proceeds without Whisper Announcement.

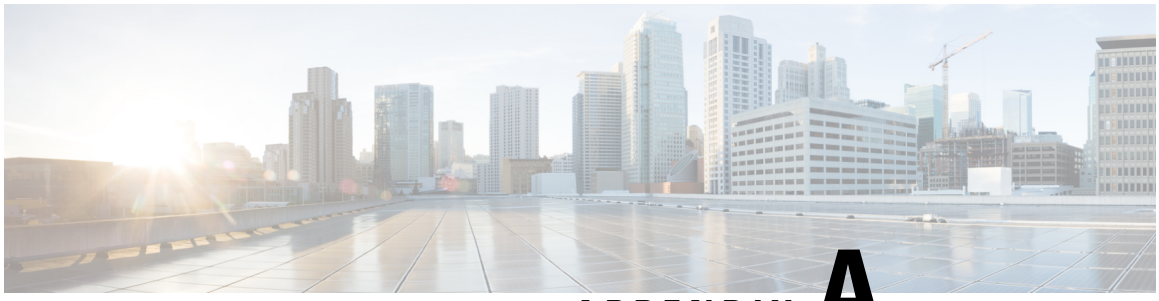
## Whisper Announcement in Agent Desktop Software

No configuration is needed to integrate Whisper Announcement with agent desktop software. While a whisper is playing, software on the agent desktop shows the call in the Ring state. Desk phones show the call in the Talking state.

## Using Agent Greeting with Whisper Announcement

You can use Agent Greeting along with the Whisper Announcement feature. Consider the following when you use them together:

- On the call, the Whisper Announcement always plays first before the greeting.
- To shorten your call-handling time, you may want to use shorter whispers and greetings than you might if you were using either feature by itself. A long whisper followed by a long greeting means a long wait before an agent handles a call.
- Usually, agents that use Whisper Announcement handle different types of calls: for example, "English, Gold Member, Activate Card, Spanish, Gold Member, Report Lost Card, English, Platinum Member, Account Inquiry." Ensure the greetings your agents record are generic enough to cover the range of customer calls they handle.



APPENDIX A

# Reverse-Proxy Configuration



## Introduction

This section describes how to configure a reverse-proxy and access the Cisco Finesse desktop without connecting to a VPN based on 12.6 ES03 and above versions of Cisco Finesse, Cisco Unified Intelligence Center (CUIC), and Cisco Identity Service (IdS).

**Note**

- The content in this chapter is provided as a guidance for customers to install and configure reverse-proxy. Cisco does not support requests for reverse-proxy installation and configuration issues. Queries that are related to this subject can be discussed onCisco community forums.
- For ES04 deployments of VPN-less Finesse, see the Cisco Finesse 12.6 ES04 Readme.

**Note** The OpenResty® Nginx configurations provided as part of Release 12.5(1) SU2 need to be manually edited and applied to match your deployment, along with requiring a manual install of the OpenResty® Nginx.

## Prerequisites

### Requirements

Cisco recommends that you have knowledge of the following:

- Cisco Unified Contact Center Enterprise (Unified CCE) Release
- Cisco Finesse
- Linux administration
- Network administration and Linux network administration

## Components Used

The information in this section is based on the following software and hardware versions:

- Cisco Finesse - 12.6 ES03
- Cisco Unified Intelligence Center - 12.6 ES03
- IdS - 12.6 ES03
- Unified CCE - 11.6 or later
- Packaged CCCE - 12.0 or later

### Related Topics

Performance

# Background Information

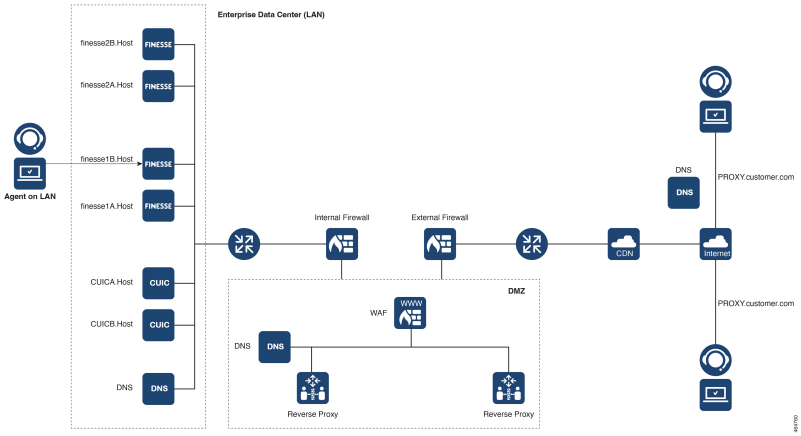
This deployment model is supported for the Unified CCE and Packaged CCE solutions.

Deployment of a reverse-proxy is supported (available from 12.6 ES01) as an option to access the Cisco Finesse desktop without connecting to a VPN.

To enable this feature, a reverse-proxy pair must be deployed in the Demilitarized Zone (DMZ).

Media access remains unchanged in reverse-proxy deployments. To connect to the media, agents can use Cisco Jabber over MRA solution or the Mobile Agent capability of Unified CCE with a Public Switched Telephone Network (PSTN) or mobile endpoint. This diagram shows how the network deployment will look like when you access two Finesse clusters and two Cisco Unified Intelligence Center nodes through a single HA pair of reverse-proxy nodes.

Concurrent access from agents on the Internet and agents who connect from LAN is supported as shown in the following image:




**Note** For more information on how to select an appropriate reverse-proxy that supports this deployment, see the section Reverse-Proxy Selection Criteria at *Security Guide for Cisco Unified ICM/Contact Center Enterprise, Release 12.6(1).*

Before you read this section, it is suggested to refer to VPN-less Access to Finesse Desktop. Also, see the *Security Considerations for Mobile Agent Deployments* section in *Security Guide for Cisco Unified ICM/Contact Center Enterprise, Release 12.6(1)*.

### Related Topics

Components Used, on page 200

# Upgrade Notes for ES01-Based VPN-less Configurations

- Nginx ES03-based configuration requires Nginx installation with Lua support.
- Certificate Requirements
  - Cisco Finesse, Cisco Unified Intelligence Center, and IdS require the Nginx or OpenResty host certificate to be added to the Tomcat trust store and restart the system. This enables Nginx ES02-based configuration to successfully connect to the component servers.
  - Cisco Finesse, Cisco Unified Intelligence Center, and IdS upstream server certificates need to be configured in the Nginx server to use the ES03-based configuration.

**Note** It is recommended to remove the existing ES01-based Nginx configuration before you install the ES03-based Nginx configurations.

**Note** Nginx ES03-based configuration scripts require the corresponding ES03 COP installation in Cisco Finesse, Cisco Unified Intelligence Center, and IdS.

## Validating Unauthenticated Static Resources

All valid endpoints that can be accessed without any authentication are actively tracked in the ES04 scripts . Requests to these unauthenticated paths are rejected without sending these requests to the components servers, if an invalid URI is requested.

## Brute Force Attack Prevention

Finesse 12.6 ES02 and above authentication scripts actively prevent brute force attacks that can be used to guess the user password. The scripts do this by blocking the IP address used to access the service, after a certain number of failed attempts in a short time. These requests will be rejected by **418 client error**. The number of failed requests, time interval, and blocking duration are configurable.

For more information, see the Brute Force Attack Prevention Configuration section.

## Caching CORS Headers

When the first options request is successful, the response headers **access-control-allow-headers, access-control-allow-origin, access-control-allow-methods, access-control-expose-headers,** and **access-control-allow-credentials** are cached at the proxy for five minutes. These headers are cached for each respective upstream server.

# Reverse-Proxy Configuration

## Install OpenResty as a Reverse-Proxy in DMZ

This section details the OpenResty-based proxy installation steps. The reverse-proxy is typically configured as a dedicated device in the network demilitarized zone (DMZ) as shown in the deployment diagram in *Background Information*.

1. Install the OS of your choice with the required hardware specification. Kernel and IPv4 parameter tweaks might differ depending on the OS selected. Users are advised to reverify these aspects if the chosen OS version is different from CentOS 8.0.
2. Configure two network interfaces. One interface will be required for public access from the Internet clients and another to communicate with the servers in the internal network.
3. Install OpenResty.

Any of the following Nginx versions can be used for this purpose, as long as they are based on Nginx 1.19+ and support Lua:

- Nginx Plus
- Nginx Open Source (Nginx open source must be compiled along with OpenResty-based Lua modules)
- OpenResty
- GetPageSpeed Extras

**Note** The configuration provided has been tested with OpenResty 1.19 and is expected to work with other distributions with only minor updates, if any.

# Install OpenResty

**Procedure**

| | Command or Action | Purpose |
|---|---|---|
| **Step 1** | Install OpenResty. See OpenResty Linux Packages. | As part of the OpenResty installation, Nginx will be installed in this location. Add the OpenResty path to the ***PATH*** variable by adding the following line in the ***~/.bashrc*** file.<br>`export PATH=/usr/local/openresty/bin:$PATH` |
| **Step 2** | Start or stop OpenResty Nginx | • To start OpenResty Nginx, enter **`openresty`**.<br>• To stop OpenResty Nginx, enter **`openresty -s stop`**. |

# Configure OpenResty Nginx

The configuration is explained for an OpenResty-based Nginx installation. The default directories for OpenResty are:

- <nginx-install-directory> = /usr/local/openresty/nginx
- <Openresty-install-directory> = /usr/local/openresty

1. Download and extract the 12.6-ES04-reverse-proxy-config.zip file that contains the reverse-proxy configuration for Nginx. This file is available on the Finesse Release 12.6(1)ES04 software download page.
2. Copy `nginx.conf`, `nginx/conf.d/`, and `nginx/html/` from the extracted reverse-proxy configuration directory to `<nginx-install-directory>/conf`, `<nginx-install-directory>/conf/conf.d/`, and `<nginx-install-directory>/html/` respectively.
3. Copy the `nginx/lua` directory from the extracted reverse-proxy configuration directory inside the `<nginx-install-directory>`.
4. Copy the contents of lualib to `<Openresty-install-directory>/lualib/resty`.
5. Configure OpenResty Nginx log rotation by copying the `nginx/logrotate/saproxy` file to the `<nginx-install-directory>/logrotate/` folder. Modify the file contents to point to the correct log directories if OpenResty Nginx defaults are not used.

6. OpenResty Nginx must be run with a dedicated non-privileged service account, which must be locked and have an invalid shell (or as applicable for the chosen OS).
7. Find the **Must-change** string in the files under the extracted folders named `html` and `conf.d` and replace the indicated values with the appropriate entries.
8. Ensure that all mandatory replacements are done, which are described with the **Must-change** comments in the config files.
9. Make sure that the cache directories configured for Cisco Unified Intelligence Center and Finesse are created under `<nginx-install-directory>/cache` along with these temporary directories.
   - <nginx-install-directory>/cache/client_temp
   - <nginx-install-directory>/cache/proxy_temp

**Note** The configuration provided is for a sample 2000 Agent deployment and has to be expanded appropriately for a larger deployment.

## Configure the OpenResty Nginx Cache

By default, the proxy cache paths are stored in the file system. We recommend changing them to in-memory drives by creating a cache location in tmpfs as shown here.

1. Create directories for the different proxy cache paths under /home.

   As an example, these directories must be created for the primary Finesse server. The same steps should be followed for the secondary Finesse and Cisco Unified Intelligence Center servers.

```
mkdir -p /home/primaryFinesse/rest
mkdir -p /home/primaryFinesse/desktop
mkdir -p /home/primaryFinesse/shindig
mkdir -p /home/primaryFinesse/openfire
mkdir -p /home/primaryCUIC/cuic
mkdir -p /home/primaryCUIC/cuicdoc
mkdir -p /home/client_temp
mkdir -p /home/proxy_temp

echo "tmpfs /home/primaryFinesse/rest tmpfs
size=1510M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryFinesse/desktop tmpfs
size=20M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryFinesse/shindig tmpfs
size=500M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryFinesse/openfire tmpfs
size=10M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryCUIC/cuic tmpfs
size=100M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/primaryCUIC/cuicdoc tmpfs
size=100M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/client_temp tmpfs
size=2048M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >>
/etc/fstab echo "tmpfs /home/proxy_temp tmpfs
size=2048M,rw,auto,noexec,nodev,nosuid,gid=root,uid=root,mode=1700 0 0" >> /etc/fstab
```

**Note** Increase the client and proxy_temp caches by 1 GB for each new Finesse cluster added to the configuration.

2. Mount the new mount points with the `mount -av` command.
3. Use the `df -h` command to validate if the file system has mounted the new mount points.
4. Change the proxy_cache_path locations in the Finesse and Cisco Unified Intelligence Center cache configuration files. For example, to change the paths for the Finesse primary, go to `<nginx-install-directory>conf/conf.d/finesse/caches` and change the existing cache location `/usr/local/openresty/nginx/cache/finesse25/` to the newly created filesystem location `/home/primaryFinesse.`

```
##Must-change /usr/local/openresty/nginx/cache/finesse25 location would change depending
 on folder extraction
proxy_cache_path /home/primaryFinesse/desktop levels=1:2 use_temp_path=on
keys_zone=desktop_cache_fin25:10m max_size=15m inactive=3y use_temp_path=off;
proxy_cache_path /home/primaryFinesse/shindig levels=1:2 use_temp_path=on
keys_zone=shindig_cache_fin25:10m max_size=500m inactive=3y use_temp_path=off;
proxy_cache_path /home/primaryFinesse/openfire levels=1:2 use_temp_path=on
keys_zone=openfire_cache_fin25:10m max_size=10m inactive=3y use_temp_path=off;
proxy_cache_path /home/primaryFinesse/rest levels=1:2 use_temp_path=on
keys_zone=rest_cache_fin25:10m max_size=1500m inactive=40m use_temp_path=off;
```

5. Follow the same steps for the Finesse secondary server and Cisco Unified Intelligence Center server.

**Note** Ensure that the sum of all the **tmpfs** drive sizings created in all the previous steps are added to the final memory sizing for the deployment. This is because these drives are memory blocks that are configured to look like disks to the application and they consume memory.

## Configure Log Rotation

Nginx reverse-proxy produces many logs. Configure log rotation to ensure optimum use of disk space, else the logs fill up the disk. The steps to configure log rotation is as follows:

1. Copy the configuration file from `/usr/local/openresty/nginx/logrotate/saproxy` to `/etc/logrotate.d/reverseproxy`.
2. Ensure that there's a file in the `/etc/cron.daily/logrotate`. This does the log rotation daily based on the configuration in the `/etc/logrotate.d/reverseproxy`.

**Note** If log rotation has to be done more frequently, such as every hour, copy the configuration file from `/etc/cron.daily/logrotate` to `/etc/cron.hourly/logrotate`.

3. Log files under `/usr/local/openresty/nginx/logs/` are rotated based on the configuration in `/etc/logrotate.d/reverseproxy`. That is, rotate the log when the file size exceeds 100 MB and keep no more than 20 most recently rotated files.

The status of the log rotation is available in the `/var/lib/logrotate/logrotate.status` log file.

## Use Self-Signed Certificates—Test Deployments

Use self-signed certificates until the reverse-proxy is ready to be rolled out into production. On a production deployment, use only a Certificate Authority-signed (CA-signed) certificate.

1. Generate OpenResty Nginx certificates for SSL folder content. Before you generate certificates, you must create a folder called **ssl** under `/usr/local/openresty/nginx`. Generate two certificates (one for `<reverseproxy_primary_fqdn>` and another for <`reverseproxy_secondary_fqdn`>) with the help of the following commands:

   a. `sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /usr/local/openresty/nginx/ssl/nginx.key -out /usr/local/openresty/nginx/ssl/nginx.crt` (pass hostname as: <reverseproxy_primary_fqdn>)

   b. `sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /usr/local/openresty/nginx/ssl/nginxnode2.key -out /usr/local/openresty/nginx/ssl/nginxnode2.crt` (pass hostname as: <reverseproxy_secondary_fqdn>)

   c. Ensure that the certificate path is `/usr/local/openresty/nginx/ssl/nginx.crt` and `/usr/local/openresty/nginx/ssl/nginxnode2.crt`, because these are already configured in Finesse Nginx configuration files.

2. Change the permission of the private key **400 (r--------)**.

3. Configure the firewall and iptables on the reverse-proxy to enable the firewall to communicate with the ports that are configured to listen to the OpenResty Nginx server.

4. Add the IP address and hostname of all the configured servers in the `/etc/hosts` file of the reverse-proxy server.

**Note** The provided configuration is for a sample 2000 Agent deployment and must be expanded appropriately for larger deployments.

## Use CA-Signed Certificate—Production Deployments

A CA-signed certificate can be installed on the reverse-proxy with these steps:

1. Generate the certificate signing request (CSR).

   To generate the CSR and private key, enter `openssl req -new -newkey rsa:4096 -keyout nginx.key -out nginx.csr` after you log in to the proxy. Follow the prompt, and provide the details. This generates the CSR (nginx.csr in the example) and the RSA private key (nginx.key in the example) of 4096 bits.

   For example:

```
[root@reverseproxyhost.companyname.com ssl]# openssl req -new -newkey rsa:4096 -keyout
nginx.key -out nginx.csr
Generating a RSA private key
.....+++++
.....+++++
writing new private key to 'nginx.key'
Enter PEM pass phrase:passphrase
Verifying - Enter PEM pass phrase:passphrase
```

```
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:US
State or Province Name (full name) []:CA
Locality Name (eg, city) [Default City]:Orange County
Organization Name (eg, company) [Default Company Ltd]:CompanyName
Organizational Unit Name (eg, section) []:BusinessUnit
Common Name (eg, your name or your server's hostname)
[]:reverseproxyhostname.companydomain.com
Email Address []:john.doe@comapnydomain.com
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:challengePWD
An optional company name []:CompanyName
```

Write down the PEM passphrase. This is used to decrypt the private key during the deployment.

2. Obtain the signed certificate from the CA.

   Send the CSR to the certificate authority and obtain the signed certificate.

**Note** If the certificate received from the CA is not a certificate chain containing all the respective certificates, compose all the relevant certificates into a single certificate chain file.

3. Deploy the certificate and key.

   Decrypt the key generated in the first step with the `openssl rsa -in nginx.key -out nginx_decrypted.key` command. Place the CA-signed certificate and the decrypted key inside the folder `/usr/local/openresty/nginx/ssl` in the reverse-proxy machine. Update or add the following SSL configurations related to the certificate in the OpenResty Nginx configurations in the following file: `/usr/local/openresty/nginx/conf/conf.d/ssl/ssl.conf`:

   ```
   ssl_certificate /usr/local/openresty/nginx/ssl/ca_signed_cert.crt;
   ssl_certificate_key/usr/local/openresty/nginx/ssl/nginx_decrypted.key;
   ```

4. Configure permissions for the certificates.

   Enter `chmod 400 /usr/local/openresty/nginx/ssl/ca_signed_cert.crt` and `chmod 400 /usr/local/openresty/nginx/ssl/nginx_decrypted.key`, so that the certificate has read-only permission and is restricted to the owner.

5. Reload OpenResty Nginx.

## Create Custom Diffie-Hellman Parameter

1. Create a custom Diffie-Hellman parameter by using the following commands:

   ```
   openssl dhparam -out /usr/local/openresty/nginx/ssl/dhparam.pem 2048
   chmod 400 /usr/local/openresty/nginx/ssl/dhparam.pem
   ```

2. Modify the server configuration to use the new parameters in the file `/usr/local/openresty/nginx/conf/conf.d/ssl/ssl.conf` by using the following command:

```
ssl_dhparam /usr/local/openresty/nginx/ssl/dhparam.pem;
```

## Enable OCSP Stapling

**Note** In order to enable the Online Certificate Status Protocol (OCSP) stapling, the server should be using a CA-signed certificate and the server should have access to the CA which signed the certificate.

Add or update this configuration in the file: `/usr/local/openresty/nginx/conf/conf.d/ssl/ssl.conf`:

- `ssl_stapling on;`
- `ssl_stapling_verify on;`

## Modify the Common OpenResty Nginx Configuration

The default OpenResty Nginx configuration file `(/usr/local/openresty/nginx/conf/nginx.conf)` has to be modified to contain the following entries to enforce security and enhance performance. The following content should be used to modify the default configuration file (`nginx.conf`) which is created during the OpenResty Nginx installation:

```
# Increasing number of worker processes will not increase the processing the request. The
number of worker process will be same as number of cores
# in system CPU. OpenResty Nginx provides "auto" option to automate this, which will spawn
 one worker for each CPU core.
worker_processes  auto;

# Process id file location
pid /usr/local/openresty/nginx/logs/nginx.pid;

# Binds each worker process to a separate CPU
worker_cpu_affinity auto;

#Defines the scheduling priority for worker processes. This should be calculated by "nice"
 command. In our proxy set up the value is 0
worker_priority 0;


error_log  /usr/local/openresty/nginx/logs/error.log info;

#user root root;

# current limit on the maximum number of open files by worker processes, keeping 10 times
of worker_connections

worker_rlimit_nofile 102400;

events {
    multi_accept on;


    # Sets the maximum number of simultaneous connections that can be opened by a worker
process.
```

```
    # This should not be more the current limit on the maximum number of open files i.e.
hard limit of the maximum number of open files for the user (ulimit -Hn)
    # The appropriate setting depends on the size of the server and the nature of the
traffic, and can be discovered through testing.
    worker_connections  10240;
    #debug_connection 10.78.95.21
}


http {

    include       mime.types;


    default_type  text/plain;


    ## Must-change Change with DNS resolver ip in deployment
    resolver 192.168.1.3;


    ## Must-change change lua package path to load lua libraries
    lua_package_path
"/usr/local/openresty/lualib/resty/?.lua;/usr/local/openresty/nginx/lua/?.lua;;"


    ## Must-change change proxy_temp folder as per cache directory configurations
    proxy_temp_path /usr/local/openresty/nginx/cache/proxy_temp 1 2 ;
    ## Must-change change client_temp folder as per cache directory configurations
    client_body_temp_path /usr/local/openresty/nginx/cache/client_temp 1 2 ;


    lua_shared_dict userlist 50m;
    lua_shared_dict credentialsstore 100m;
    lua_shared_dict userscount 100k;
    lua_shared_dict clientstorage 100m;
    lua_shared_dict blockingresources 100m;
    lua_shared_dict tokencache_saproxy 10M;
    lua_shared_dict tokencache_saproxy125 10M;
    lua_shared_dict ipstore 10m;
    lua_shared_dict desktopurllist 10m;
    lua_shared_dict desktopurlcount 100k;
    lua_shared_dict thirdpartygadgeturllist 10m;
    lua_shared_dict thirdpartygadgeturlcount 100k;
    lua_shared_dict corsheadersstore 100k;


    init_worker_by_lua_block {
        local UsersListManager = require('users_list_manager')
        local UnauthenticatedDesktopResourcesManager =
require("unauthenticated_desktopresources_manager")
        local UnauthenticatedResourcesManager =
require("unauthenticated_thirdpartyresources_manager")
        --  Must-change Replace saproxy.cisco.com with reverseproxy fqdn

        if  ngx.worker.id() == 0 then
            UsersListManager.getUserList("saproxy.cisco.com",
"https://saproxy.cisco.com:8445/finesse/api/Users")
           UnauthenticatedDesktopResourcesManager.getDesktopResources("saproxy.cisco.com",
 "https://saproxy.cisco.com:8445/desktop/api/urls?type=desktop")
          UnauthenticatedResourcesManager.getThirdPartyGadgetResources("saproxy.cisco.com",
 "https://saproxy.cisco.com:8445/desktop/api/urls?type=3rdParty")
        end
    }
```

```
include conf.d/*.conf;

sendfile        on;

tcp_nopush     on;

server_names_hash_bucket_size 512;
```

## Configure Reverse-Proxy Port

By default, the Nginx configuration for Finesse requests on port 8445. At a time, only one port can be enabled from a reverse-proxy to support Finesse requests. If port 443 needs to be supported, edit the `<nginx-install-directory>conf/conf.d/finesse.conf` file to enable listening on 443 and disable listening on 8445.

## Configure Mutual TLS Authentication Between Reverse-Proxy and Components

Client SSL certificate authentication for connections from reverse-proxy hosts can be enabled on Cisco Unified Intelligence Center, Finesse, IdS, and LiveData by using the new CVOS CLI option `utils system reverse-proxy client-auth enable/disable/status`.

By default this is disabled and has to be enabled by the administrator by executing the CLI on each upstream server independently. After this option is enabled, Cisco Web proxy Service running on upstream host will start authenticating client certificates in TLS handshake for connections originating from trusted reverse-proxy hosts added by using the CLI `utils system reverse-proxy allowed-hosts add <proxy-host>`.

The following is the configuration block for the same in proxy config files named **ssl.conf** and **ssl2.conf**.

```
#Must-change /usr/local/openresty/nginx/ssl/nginx.crt change this location accordingly
proxy_ssl_certificate /usr/local/openresty/nginx/ssl/nginx.crt;
#Must-change /usr/local/openresty/nginx/ssl/nginx.key change this location accordingly
proxy_ssl_certificate_key /usr/local/openresty/nginx/ssl/nginx.key;
```

The SSL certificate used for outbound traffic (proxy to upstream) can be the same as the SSL certificate that is configured for inbound traffic (SSL connector for component server blocks). If self-signed certificate is used as **proxy_ssl_certificate**, it has to be uploaded to the tomcat trust store of the upstream components (Finesse/IdS/Cisco Unified Intelligence Center/Livedata) for it to be authenticated successfully.

Upstream server certificate validation by reverse-proxy is optional and disabled by default. If you wish to achieve full TLS mutual auth between reverse-proxy and upstream hosts, the following configuration has to be uncommented in the **ssl.conf** and **ssl2.conf** files.

```
#Enforce upstream server certificate validation at proxy ->
#this is not mandated as per CIS buit definitely adds to security.
#It requires the administrator to upload all upstream server certificates to the proxy
certificate store
#Must-Change Uncomment below lines IF need to enforce upstream server certificate validation
 at proxy
#proxy_ssl_verify on;
#proxy_ssl_trusted_certificate /usr/local/openresty/nginx/ssl/finesse25.crt;
proxy_ssl_trusted_certificate: This file should contain the all upstream certificate enteries
 concatenated together
```

mutual TLS (mTLS) is a standard security requirement for connections established from DMZ into the data center. For more information, see Nginx CIS behcmarks-https://www.cisecurity.org/benchmark/nginx

mTLS requires that both the server and client be pre-configured with mutual information about each other, as well as that the mutual certificates be properly verified. Hence the term Mutual TLS. A properly configured proxy server will be able to circumvent TCP rate limits and provide the client IP to the server for logging

purposes. As a result, it is critical that the proxy identity be verified before connecting as a reverse-proxy. For security reasons, it is therefore recommended that this feature be used and turned on.

This requires the upstream component certificates to be made available to the proxy and vice-versa. Reverse-proxy by default establishes verified TLS connections to the upstream server and it is the proxy verification at the client which is optional. Therefore this needs to be enabled at the upstream client server.

### Enabling mutual TLS

The mutual TLS needs to be enabled at the upstream component servers using the provided CLI.

Use the **utils system reverse-proxy client-auth enable** CLI to enable proxy certificate verification at the upstream component server.

After running the CLI, upload the proxy SSL certificate corresponding to the reverse-proxy hostname used to connect to the same server. This can be used to verify TLS connections when the reverse-proxy attempts to establish an upstream connection.

# Clear Cache

The reverse-proxy cache can be cleared with the `<NGINX_HOME>/clearCache.sh` command.

# Standard Guidelines

This section briefly describes the standard guidelines that must be followed when you set up OpenResty Nginx as a proxy server. The guidelines for the OpenResty Nginx server software is derived from the Center for Internet Security.

1. Use the latest stable versions of OpenResty and OpenSSL version.
2. Install OpenResty Nginx in a separate disk mount.
3. The OpenResty Nginx process id must be owned by the root user (or as applicable for the chosen OS) and must have permission **644 (rw-------)** or stricter.
4. OpenResty Nginx must block requests for unknown hosts. Ensure that each server block contains the server_name directive explicitly defined. To verify, search all server blocks in the `nginx.conf` and `nginx/conf.d` files and verify that all server blocks contain the server_name.
5. OpenResty Nginx must listen only on the authorized ports. Search all server blocks in the `nginx.conf` and `nginx/conf.d` files and check for the `listen` directives to verify that only the authorized ports are open for requests.
6. Block the proxy server HTTP port, because Cisco Finesse does not support HTTP.
7. The OpenResty Nginx SSL protocol must be TLS 1.2. Remove support for legacy SSL protocols. Disable weak SSL ciphers.
8. Send the OpenResty Nginx error and access logs to the remote syslog server.
9. Install the **mod_security** module that works as a web application firewall. See the ModSecurity manual for more information. Note that OpenResty Nginx load has not been verified within the **mod_security** module in place.

# Configure the Mapping File

Refer to Host Mapping File for Network Translation.

**Related Topics**

Host Mapping File for Network Translation

## Use Reverse-Proxy as the Mapping File Server

**Note** This appendix has the configuration details, for more information about the pre-requisites, refer to Use Reverse-Proxy as the Mapping File Server, on page 212.

These steps are required only if the reverse-proxy is also used as the proxy mapping file host.

1. Configure the reverse-proxy hostname in the domain controller used by the Finesse, Cisco Unified Intelligence Center and IdS hosts such that its IP address can be resolved.
2. Upload the generated OpenResty® Nginx signed certificates on both the nodes under tomcat-trust of cmplatform and restart the server.
3. Update the **Must-change** values in `<NGINX_HOME>/html/proxymap.txt`.
4. Reload OpenResty® Nginx configurations with the `nginx -s reload` command.
5. Use the `curl` command to validate if the configuration file is accessible from another network host.

## CentOS 8 Kernel Hardening

If the operating system is Cent OS 8 and the installations use a dedicated server for hosting the proxy, harden the kernel by using these `sysctl` configurations:

```
## Configurations for kernel hardening - CentOS8. The file path is /etc/sysctl.conf
## Note that the commented configurations denote that CentOS 8's default value matches
## the recommended/tested value, and are not security related configurations.

# Avoid a smurf attack
net.ipv4.icmp_echo_ignore_broadcasts = 1
# Turn on protection for bad icmp error messages
net.ipv4.icmp_ignore_bogus_error_responses = 1

# Turn on syncookies for SYN flood attack protection
net.ipv4.tcp_syncookies = 1

# Turn on and log spoofed, source routed, and redirect packets
net.ipv4.conf.all.log_martians = 1
net.ipv4.conf.default.log_martians = 1

# Turn off routing
net.ipv4.ip_forward = 0
net.ipv4.conf.all.forwarding = 0
net.ipv6.conf.all.forwarding = 0

net.ipv4.conf.all.mc_forwarding = 0
net.ipv6.conf.all.mc_forwarding = 0

# Block routed packets
net.ipv4.conf.all.accept_source_route = 0
```

```
net.ipv4.conf.default.accept_source_route = 0
net.ipv6.conf.all.accept_source_route = 0
net.ipv6.conf.default.accept_source_route = 0

# Block ICMP redirects
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.default.accept_redirects = 0
net.ipv6.conf.all.accept_redirects = 0
net.ipv6.conf.default.accept_redirects = 0
net.ipv4.conf.all.secure_redirects = 0
net.ipv4.conf.default.secure_redirects = 0
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.send_redirects = 0

# Filter routing packets with inward-outward path mismatch(reverse path filtering)
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.default.rp_filter = 1

# Router solicitations & advertisements related.
net.ipv6.conf.default.router_solicitations = 0
net.ipv6.conf.default.accept_ra_rtr_pref = 0
net.ipv6.conf.default.accept_ra_pinfo = 0
net.ipv6.conf.default.accept_ra_defrtr = 0
net.ipv6.conf.default.autoconf = 0
net.ipv6.conf.default.dad_transmits = 0
net.ipv6.conf.default.max_addresses = 1
net.ipv6.conf.all.accept_ra = 0
net.ipv6.conf.default.accept_ra = 0

# Backlog - increased from default 1000 to 5000.
net.core.netdev_max_backlog = 5000

# Setting syn/syn-ack retries to zero, so that they don't stay in the queue.
net.ipv4.tcp_syn_retries = 0
net.ipv4.tcp_synack_retries = 0

# Max tcp listen backlog. Setting it to 511 to match nginx config
net.core.somaxconn = 511

# Reduce the duration of connections held in TIME_WAIT(seconds)
net.ipv4.tcp_fin_timeout = 6

# Maximum resources allotted
# fs.file-max = 2019273
# kernel.pid_max = 4194304
# net.ipv4.ip_local_port_range = 32768 60999

# TCP window size tuning
# net.ipv4.tcp_window_scaling = 1
# net.core.rmem_default = 212992
# net.core.rmem_max = 212992
# net.ipv4.tcp_rmem = 4096 87380 6291456
# net.ipv4.udp_rmem_min = 4096
# net.core.wmem_default = 212992
# net.core.wmem_max = 212992
# net.ipv4.tcp_wmem = 4096 16384 4194304
# net.ipv4.udp_wmem_min = 4096
# vm.lowmem_reserve_ratio = 256 256 32 0 0
# net.ipv4.tcp_mem = 236373 315167 472746

# Randomize virtual address space
kernel.randomize_va_space = 2

# Congestion control
```

```
# net.core.default_qdisc = fq_codel
# net.ipv4.tcp_congestion_control = cubic

# Disable SysReq
kernel.sysrq = 0

# Controls the maximum size of a message, in bytes
kernel.msgmnb = 65536

# Controls the default maximum size of a message queue
kernel.msgmax = 65536

# Controls the eagerness of the kernel to swap.
vm.swappiness = 1
```

Reboot after you make the recommended changes.

## IPtables Hardening

IPtables is an application that allows a system administrator to configure the IPv4 and IPv6 tables, chains, and rules provided by the Linux kernel firewall.

The IPtables rules are configured to secure the proxy application from brute force attacks by restricting the access in the Linux kernel firewall.

The comments in the configuration indicate which service is being rate-limited by using the rules.

**Note** If administrators use a different port or expand access to multiple servers using the same ports, they must do appropriate sizing for these ports accordingly.

A sample IPtable is as follows:

```
## Configuration for iptables service
## The file path is /etc/sysconfig/iptables
## Make a note for must-change values to be replaced.
## Restart of the iptable service is required after applying following rules

*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]

# Ensure loopback traffic is configured
-A INPUT -i lo -j ACCEPT
-A OUTPUT -o lo -j ACCEPT
-A INPUT -s 127.0.0.0/8 -j DROP

# Ensure ping openeded only for the particular source and blocked for rest
# Must-Change: Replace the x.x.x.x with valid ip address
-A INPUT -p ICMP --icmp-type 8 -s x.x.x.x -j ACCEPT

# Ensure outbound and established connections are configured
-A INPUT -p tcp -m state --state RELATED,ESTABLISHED -j ACCEPT
-A OUTPUT -p tcp -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT

# Block ssh for external interface
# Must-Change: Replace the ens224 with valid ethernet interface
-A INPUT -p tcp -i ens224 --dport 22 -j DROP
# Open inbound ssh(tcp port 22) connections
-A INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT
```

```
# Configuration for ccp 8445 port
-A INPUT -p tcp -m tcp --dport 8445 --tcp-flags SYN SYN -m connlimit --connlimit-above 10
--connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " Connections to 8445 exceeded connlimit "
-A INPUT -p tcp -m tcp --dport 8445 --tcp-flags SYN SYN -m connlimit --connlimit-above 10
--connlimit-mask 32 --connlimit-saddr -j DROP
 -A INPUT -p tcp -m tcp --dport 8445 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 6/sec
 --hashlimit-burst 8 --hashlimit-mode srcip,dstport --hashlimit-name TCP_8445_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8445 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 8445 hashlimit "
-A INPUT -p tcp -m tcp --dport 8445 --tcp-flags SYN SYN -j DROP

# Configuration for IdS 8553 port
-A INPUT -p tcp -m tcp --dport 8553 --tcp-flags SYN SYN -m connlimit --connlimit-above 6
--connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " IdS connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 8553 --tcp-flags SYN SYN -m connlimit --connlimit-above 6
--connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 8553 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 2/sec
 --hashlimit-burst 4 --hashlimit-mode srcip,dstport --hashlimit-name TCP_8553_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8553 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 8553 hashlimit "
-A INPUT -p tcp -m tcp --dport 8553 --tcp-flags SYN SYN -j DROP

# Configuration for IdP 443 port
-A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -m connlimit --connlimit-above 8
--connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " IdP connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -m connlimit --connlimit-above 8
--connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 4/sec
 --hashlimit-burst 6 --hashlimit-mode srcip,dstport --hashlimit-name TCP_443_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 443 hashlimit "
-A INPUT -p tcp -m tcp --dport 443 --tcp-flags SYN SYN -j DROP


# Must-Change: A2A file transfer has not been considered for below IMNP configuration.
# For A2A for support, these configuration must be recalculated to cater different file
transfer scenarios.

# Configuration for IMNP 5280 port
-A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30
--connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " IMNP connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30
--connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 20/sec
 --hashlimit-burst 25 --hashlimit-mode srcip,dstport --hashlimit-name TCP_5280_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 5280 hashlimit "
-A INPUT -p tcp -m tcp --dport 5280 --tcp-flags SYN SYN -j DROP

# Configuration for IMNP 15280 port
-A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30
 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " IMNP connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30
 --connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto
20/sec --hashlimit-burst 25 --hashlimit-mode srcip,dstport --hashlimit-name TCP_15280_DOS
-j ACCEPT
```

```
-A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 15280 hashlimit "
-A INPUT -p tcp -m tcp --dport 15280 --tcp-flags SYN SYN -j DROP


# Configuration for IMNP 25280 port
-A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30
 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " IMNP connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -m connlimit --connlimit-above 30
 --connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto
20/sec --hashlimit-burst 25 --hashlimit-mode srcip,dstport --hashlimit-name TCP_25280_DOS
-j ACCEPT
-A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 25280 hashlimit "
-A INPUT -p tcp -m tcp --dport 25280 --tcp-flags SYN SYN -j DROP


# Configuration for CUIC 8444 port
-A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -m connlimit --connlimit-above 6
--connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " CUIC connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -m connlimit --connlimit-above 6
--connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 2/sec
 --hashlimit-burst 4 --hashlimit-mode srcip,dstport --hashlimit-name TCP_8444_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 8444 hashlimit "
-A INPUT -p tcp -m tcp --dport 8444 --tcp-flags SYN SYN -j DROP

# Configuration for CUIC 8447 port
-A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -m connlimit --connlimit-above 6
--connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " CUIC connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -m connlimit --connlimit-above 6
--connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 2/sec
 --hashlimit-burst 4 --hashlimit-mode srcip,dstport --hashlimit-name  TCP_8447_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 8447 hashlimit "
-A INPUT -p tcp -m tcp --dport 8447 --tcp-flags SYN SYN -j DROP

# Configuration for LiveData 12005  port
-A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -m connlimit --connlimit-above 10
 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " LD connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -m connlimit --connlimit-above 10
 --connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 6/sec
 --hashlimit-burst 8 --hashlimit-mode srcip,dstport --hashlimit-name TCP_12005_DOS -j ACCEPT
-A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 12005 hashlimit "
-A INPUT -p tcp -m tcp --dport 12005 --tcp-flags SYN SYN -j DROP

# Configuration for LiveData 12008  port
-A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -m connlimit --connlimit-above 10
 --connlimit-mask 32 --connlimit-saddr -m limit --limit 1/min --limit-burst 1 -j LOG
--log-prefix " LD connection limit exceeded"
-A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -m connlimit --connlimit-above 10
 --connlimit-mask 32 --connlimit-saddr -j DROP
-A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -m hashlimit --hashlimit-upto 6/sec
 --hashlimit-burst 8 --hashlimit-mode srcip,dstport --hashlimit-name  TCP_12008_DOS -j
ACCEPT
```

```
-A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -m limit --limit 1/min --limit-burst
 1 -j LOG --log-prefix " Exceeded 12008 hashlimit "
-A INPUT -p tcp -m tcp --dport 12008 --tcp-flags SYN SYN -j DROP

# Block all other ports
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited

COMMIT
```

**Note** The rules that are provided block the DNS resolution at the proxy. So, all the hostnames of the components that are configured in the proxy must be explicitly added to the host resolution file `/etc/hosts`.

Interface level rules must be added to restrict access to only users accessing via LAN and to block public access to port 10000, which is used for accessing the proxy map file. For example,

```
-A INPUT -p tcp -m tcp -i <PRIVATE_INTERFACE> --dport 10000 --tcp-flags SYN SYN -m hashlimit
--hashlimit-upto 35/sec --hashlimit-burst 2000 --hashlimit-mode srcip,dstport --hashlimit-name
TCP_10000_DOS -j ACCEPT -A INPUT -p tcp -m tcp -i <PRIVATE_INTERFACE> --dport 10000
--tcp-flags SYN SYN -m limit --limit 1/min --limit-burst 1 -j LOG --log-prefix " Exceeded
hashlimit " -A INPUT -p tcp -m tcp -i <PRIVATE_INTERFACE> --dport 10000 --tcp-flags SYN SYN
-j DROP
```

These rules could be applied directly by editing the `/etc/sysconfig/iptables` file manually. Alternatively, save the configuration into a file such as `iptables.conf` and run cat `iptables.conf >>/etc/sysconfig/iptables` to apply the rules.

Restart the IPtables service after you apply the rules. To restart the IPtables service, enter `systemctl restart iptables`.

# Restrict Client Connections

In addition to the previous IPtables configuration, installations that know the address range for clients who use the proxy must use this knowledge to secure the proxy access rules. This helps to secure the proxy from malicious botnets which are often created in the IP address range of countries that have more lax rules with regards to online security. Restrict the IP address ranges to country-based, state-based, or ISP-based IP ranges if you are sure of the access patterns.

# Block Client Connections

Block the specific range of addresses when an attack is identified to be made from an IP address or a range of IP addresses. In such cases, the requests from those IP addresses can be blocked with **iptable** rules.

### Block Distinct IP Addresses

To block multiple distinct IP addresses, add a line to the **IPTables** configuration file for each IP address.

For example, to block the addresses 192.0.2.3 and 192.0.2.4, enter:

```
iptables -A INPUT -s 192.0.2.3 -j DROP iptables -A INPUT -s 192.0.2.4 - j DROP.
```

### Block a Range of IP Addresses

Block multiple IP addresses in a range and add a single line to the **IPTables** configuration file with the IP address range.

For example, to block the addresses from 192.0.2.3 to 192.0.2.35, enter:

```
iptables -A INPUT -m iprange --src-range 192.0.2.3-192.0.2.35 -j DROP.
```

### Block All IP Addresses in a Subnet

Block all IP addresses in an entire subnet by adding a single line to the **IPTables** configuration file by using the classless inter-domain routing notation for the IP address range. For example, to block all class **C addresses**, enter:

```
iptables -A INPUT -s 192.0.0.0/16 -j DROP.
```

# SELinux

Security-Enhanced Linux (SELinux) is a platform security framework integrated as an enhancement into the Linux OS. The procedure to install and add SELinux policies to run OpenResty as the reverse-proxy is provided next.

1. Stop the process with the `openresty -s stop` command.
2. Configure and start or stop OpenResty Nginx server with the `systemctl` command so that during boot up the OpenResty process will start automatically. Enter these commands as root user.
   a. Go to `/usr/lib/systemd/system`.
   b. Open the file called `openresty.service`.
   c. Update the content of the file as per `PIDFile` location.

```
[Unit]
Description=The OpenResty Application Platform
After=syslog.target network-online.target remote-fs.target nss-lookup.target
Wants=network-online.target

[Service]
Type=forking
PIDFile=/usr/local/openresty/nginx/logs/nginx.pid
ExecStartPre=/usr/local/openresty/nginx/sbin/nginx -t
ExecStart=/usr/local/openresty/nginx/sbin/nginx
ExecReload=/bin/kill -s HUP $MAINPID
ExecStop=/bin/kill -s QUIT $MAINPID
PrivateTmp=true

[Install]
WantedBy=multi-user.target
```

   d. As root user, enter `sudo systemctl enable openresty`.
   e. Start or stop the OpenResty service with the `systemctl start openresty / systemctl stop openresty` command and ensure that the process starts or stops as root user.

1. **Install SElinux**
   - By default, only some SELinux packages will be installed in CentOS.

- The **policycoreutils-devel** package and its dependencies must be installed in order to generate the SELinux policy.
- Enter the following command to install **policycoreutils-devel**

```
yum install policycoreutils-devel
```

- Ensure that after you install the package, the `sepolicy` command works.

```
usage: sepolicy [-h] [-P POLICY]


{booleans,communicate,generate,gui,interface,manpage,network,transition}
                ...

SELinux Policy Inspection Tool
```

2. **Create a New Linux User and Map with SElinux User**

   a. Enter `semanage login -l` to view the mapping between Linux users and SELinux users.

```
[root@loadproxy-cisco-com ~]# semanage login -l

Login Name           SELinux User         MLS/MCS Range        Service

__default__          unconfined_u         s0-s0:c0.c1023       *              *
root                 unconfined_u         s0-s0:c0.c1023       *
```

   b. As root, create a new Linux user (**nginx** user) that is mapped to the SELinux **user_u** user.

```
useradd -Z user_u nginxuser
[root@loadproxy-cisco-com ~]# passwd nginxuser
Changing password for user nginxuser.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

   c. In order to view the mapping between **nginxuser** and **user_u**, enter this command as root:

```
[root@loadproxy-cisco-com ~]# semanage login -l

Login Name           SELinux User         MLS/MCS Range        Service

__default__          unconfined_u         s0-s0:c0.c1023       *
nginxuser            user_u               s0                   *
root                 unconfined_u         s0-s0:c0.c1023       *
```

   d. SELinux **__default__** login is by default mapped to the SELinux **unconfined_u** user. By default, it is required to confine **user_u** by using the following command:

```
semanage login -m -s user_u -r s0 __default__
```

   In order to check if the command worked properly, enter `semanage login -l`. It should produce this output:

```
Login Name           SELinux User         MLS/MCS Range        Service

__default__          user_u               s0                   *
nginxuser            user_u               s0                   *
root                 unconfined_u         s0-s0:c0.c1023       *
```

464783

   e. Modify nginx.conf and perform change ownership for nginxuser.

1. Enter `chown -R nginxuser:nginxuser *` in the `<Openresty-install-directory>` directory.
2. Modify the **nginx.conf** file to include nginxuser as the user for running worker processes.

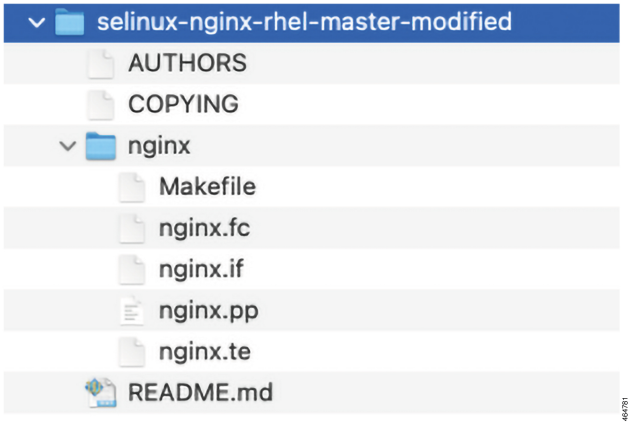
```
........

user nginxuser nginxuser;

..........
```

3. **Write the SELinux Policy for OpenResty Nginx**
   a. Instead of generating a new default custom policy for OpenResty Nginx with the `sepolicy generate --init /usr/bin/nginx` command, start with an existing policy.

   The **nginx.fc** file (File Contexts file) and **nginx.te** (Type Enforcement file) files, that are downloaded from the following location, are modified for reverse-proxy usage:




   This modified version can be used as a reference because it is updated for a particular use case.

   b. Download the **selinux-nginx-rhel-master-modified.tar** file from the Software Download .

   c. Extract the **.tar** file and navigate to the **nginx** directory within it.

   d. Open the **.fc** file and verify the required file paths of **Nginx installer**, **cache**, and **pid** files.

   e. Compile the configuration with the `make` command.

   f. The **nginx.pp** file is generated.

   g. Load the policy with the `semodule` command.

```
semodule -i nginx.pp
```

   h. Go to **/root** and create an empty file called `touch /.autorelabel`.

   i. Reboot the system.

   j. Enter the following command to verify that the policy is loaded successfully:

```
semodule --list-modules=full
```

```
[root@loadproxy-cisco-com ~]# semodule --list-modules=full
400 nginx               pp
200 container           pp
200 flatpak             pp
100 abrt                pp
100 accountsd           pp
100 acct                pp
100 afs                 pp
100 aiccu               pp
100 aide                pp
100 ajaxterm            pp
100 alsa                pp
```

464782

**k.** OpenResty Nginx should run without any violation. (Violation logs will be available in **/var/log/messages** and **/var/log/audit/audit.log**).

**l.** Enter the following command to check the status of OpenResty Nginx:

```
ps -aefZ | grep nginx
```

```
[root@loadproxy-cisco-com ~]# ps -aefZ |grep nginx
system_u:system_r:nginx_t:s0    root        1686       1  0 16:14 ?        00:00:00 nginx: master process /usr/bin/nginx
system_u:system_r:nginx_t:s0    nginxus+    1687    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1688    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1689    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1690    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1691    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1692    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1693    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1694    1686  0 16:14 ?        00:00:00 nginx: worker process
system_u:system_r:nginx_t:s0    nginxus+    1695    1686  0 16:14 ?        00:00:00 nginx: cache manager process
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 root 2543 2252  0 16:17 pts/0 00:00:00 grep --color=auto nginx
```

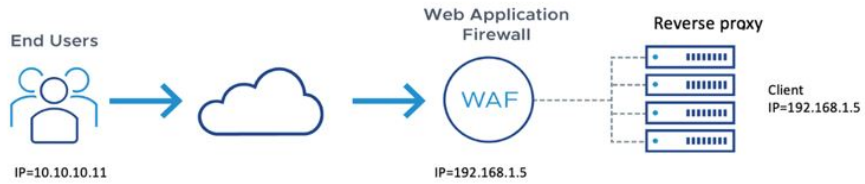
464784

**m.** Now the Finesse agent desktop or supervisor desktop should be accessible.

# Load Balancer, WAF, and Proxy support for reverse-proxy deployments

The reverse-proxy configurations have security features that are dependent on the information of the actual client IP. Requesting rate limits, logging of client activity, blocking the users due to multiple wrong credentials require the configuration to track the client IP to appropriately rate-limit, block, or log the actual users' access.

No specific configurations are required for deployments which directly terminate the agent connections on the reverse-proxy. The reverse-proxy has the information of the client IP due to the connections directly reaching the reverse-proxy. However, when other network devices are used to terminate the client connections, before forwarding them as fresh requests to the reverse-proxy, the client IPs are no longer visible to the reverse-proxy.



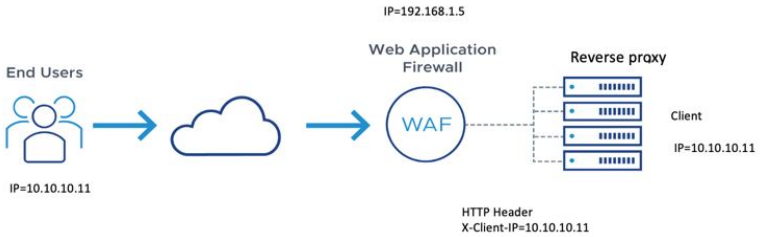
This happens when there are Load Balancers, Web Application Firewall (WAF), or other proxies deployed in front of a reverse-proxy. The CDN deployments work as an intermediary reverse-proxy/WAF and fall into the same deployment category.

Such deployments **MUST** add certain reverse-proxy configurations to enable the reverse-proxy to identify the actual client IP. The configurations that are required for such deployments are as follows:

1. The public IPs and private IPs of the devices which will forward the requests to the reverse-proxy must be to be added in **nginx.conf** http block and **maps.conf** geo block as mentioned in the **##Must-change** comment.
2. The new requests originating from the intermediary devices, **MUST** populate HTTP request header fields with the end-client IP to communicate the same to the reverse-proxy.

   The name of the request header field is configured in **nginx.conf** file using `real_ip_header` directive. For example: `real_ip_header X-REAL-IP;`

**Note** All CDN deployments provide a mechanism to extract the client IP as a HTTP header containing a single-client IP as part of the request payload. A custom header is often recommended to avoid conflict with the standard `X-FORWARDED-FOR` header. The VPN-less reverse-proxy deployments are also recommended to provide the client IP using a custom header for similar reasons.




3. For security purpose, the devices which are front ending the reverse-proxy **MUST** replace `X-FORWARDED-FOR` and `X-REAL-IP` headers provided by the client with the actual client IP or drop them if the deployment does not need these headers.
4. If the deployment is using a custom HTTP header for communicating to the client IP, the particular field **MUST** be replaced with the client IP before forwarding them upstream to the reverse-proxy.
5. Verify the configuration by transmitting a high rate of requests to a Finesse API such as `SystemInfo/DesktopConfig` from an external client. Verify through the Load Balancer or WAF to ensure that the client is blocked while the Load Balancer or intermediate devices are not blocked or rate limited. Ensure that the configurations are working as expected before going live.

   Refer to the Logging section for instructions on how to check whether a client is blocked or rate limited.
6. Deployments that employ WAF or other security devices must ensure that the desktop API traffic patterns are compatible with them before going live with the deployment. Certain WAF rules can be too restrictive and may need some modifications before they can be deployed.

**Note** The reverse-proxy configurations provided have no protection against layer-3 attacks such as IP address spoofing or flooding. The proxy provides only rate limiting, brute force attack detection, and restricting of requests to the allowed destinations. The operating system IP configurations are hardened to a certain level but there are no further protections that are available. It is assumed that the relevant operating system hardening and traffic protection devices are employed to secure the deployment Cisco Contact Center.

For more details refer to the *Security Guidelines for Reverse-Proxy Deployment* section in the *Security Guide for Cisco Unified ICM/Contact Center Enterprise, Release 12.6(1)* guide.

Load Balancers and other devices which does not have the HTTP header support can skip second and third points that are mentioned above. However, this causes a sub-optimal deployment which will be functional but loses certain features such as client IP logging for debugging purposes and blocking users attempting to brute force guess passwords.

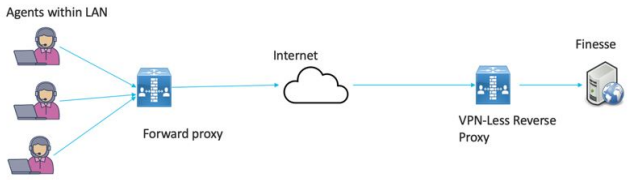
The websocket authentications will also be not effective at the reverse-proxy, which will not cause any loss in functionality but will allow all websocket request to reach the upstream component before authentication can be enforced.

### Related Topics

Logging, on page 226
Troubleshooting

## Access VPN-Less proxy through Forward proxy and NAT

The VPN-less configuration assumes that the proxy is accessed by clients/agents from the internet, who have separate individual IP's which can be used for enforcing security features. However, not all deployments dedicatedly use agents from the internet with their own unique IP addresses. Most deployments will have agents accessing the reverse-proxy deployments both from the internet as well as from LAN using the same reverse-proxy access URI.




So, if you have a deployment which uses agents behind a proxy or a NAT that looks like what is shown above, certain configuration changes have to be made to ensure that the end-user IP's are correctly communicated to the reverse-proxy. The steps to configure are as follows:

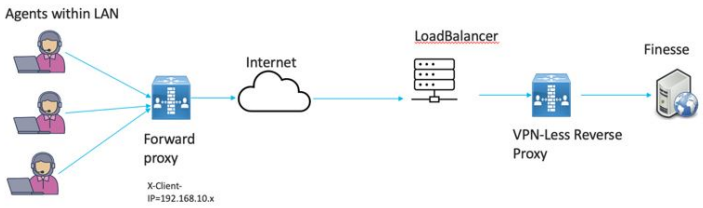
1. The Forward proxy (device A in the diagram above) has to be well-known in advance.
2. The Forward proxy device has to transmit the agent IP’s in a predefined header. For example, `X-REAL-IP` as shown above.
3. If there are other intermediary devices such as a Load Balancer or WAF at the network where Finesse is deployed, before the requests reach the reverse-proxy, these devices must be able to allow the Forward proxy by its IP address and then transmit the HTTP header without any changes.

**Note** It is crucial that the Forward proxy IP address is identified and only requests from this IP should be allowed to have the `X-REAL-IP` transmitted.

4. The **nginx.conf** http block and **maps.conf** geo block files should be updated with the list of Forward proxies' private and public IPs as mentioned in the **##Must-change** comments.

**Note** Deployments that do not have the HTTP header support can skip the steps 2 to 4. However, this causes a sub-optimal deployment which will be functional but loses certain security features listed above, which are dependent on client IP knowledge and these deployments are therefore not suggested.

Ensure that the deployment cannot support multiple HTTP header names to transmit the client IP corresponding to different Forward proxies that the network is interacting with.




Deployments such as these, should transmit or detect the final client IP of the users who are connecting from behind the **Forward proxy A** and this would be an agreement between Load Balancer and the Forward Proxy.

The Load Balancer or the final intermediary devices that forward requests to the VPN-less reverse-proxy should transmit the required headers and will need configuration as described in the section above. The Forward proxy information is not required to be added to the VPN-less configuration, if the intermediary device is able to identify the correct client IPs and transmit them to the reverse-proxy using the steps mentioned above.

However, if the actual client IP resolution is not setup between the Forward proxy and the Load Balancer, the reverse-proxy considers the IP of the Forward proxy as the actual client IP.

In this case, to avoid rate limiting to block the Forward proxy, its private and public IP addresses must be configured in **nginx.conf** http block and **maps.conf** geo block files. Both the files must be updated with the list of Forward-proxies' IP as mentioned in the **##Must-change** comments, so that the proxy is not blocked or rate limited. This would be a sub-optimal deployment and transmitting the actual client IP is recommended for a more effective deployment.

# Verifying Reverse-Proxy Configuration

## Finesse

**Procedure**

**Step 1** From the DMZ, open *https://<reverseproxy:port>/finesse/api/SystemInfo* and check if it's reachable.

**Step 2** Check if the **<host>** values in both **<primaryNode>** and **<secondaryNode>** are valid in reverse-proxy hostnames. It shouldn't be Finesse hostnames.

**Note**
- If CORS status is "enabled", you must explicitly add the reverse-proxy domain name to the list of CORS trusted domain names.
- Reverse-proxy supports a maximum of 8000 folders (including subdirectories) in the `finesse/3rdpartygadget` folder.

## Cisco Unified Intelligence Center and LiveData

**Procedure**

**Step 1** If the Finesse hostnames are seen in the response instead of reverse-proxy hostnames, validate the proxy-mapping configurations and check if the allowed hosts are properly added in Finesse servers as described in the section *Populate Network Translation Data*.

**Step 2** If LiveData gadgets load properly in Finesse Desktop, then CUIC and LiveData proxy configurations are proper.

**Step 3** To validate the Cisco Unified Intelligence Center and LiveData configurations, make the HTTP requests from the DMZ to the following URLs and check if they are reachable:

- https://<reverseproxy:cuic_port>/cuic/rest/about
- https://<reverseproxy:ldweb_port>/livedata/security
- https://<reverseproxy:ldsocketio_port>/security

## Cisco Identity Service

To validate Cisco IdS configuration, perform the following steps:

**Procedure**

**Step 1** Log in to the IdSAdmin interface at **https://<ids_LAN_host:ids_port>:8553/idsadmin** from the LAN because the admin interface is not exposed over reverse-proxy.

**Step 2** Choose **Settings** > **IdS Trust**.

**Step 3** Validate that the proxy cluster publisher node is listed on Download SP metadata page, and click **Next**.

**Step 4** Validate that the IDP proxy is correctly displayed if configured on Upload IDP metadata page, and click **Next**.

**Step 5** Initiate test SSO via all proxy cluster nodes from the Test SSO page and validate that all are successful. This requires client machine connectivity to reverse-proxy nodes.

# Brute Force Attack Prevention Configuration

Finesse 12.6 ES02 and above authentication scripts actively prevent brute force attacks that can be used to guess the user password. The scripts do this by blocking the IP address used to access the service, after a certain number of failed attempts in a short time. These requests will be rejected by **418 client error**. The number of failed requests, time interval, and blocking duration are configurable.

## Attack Detection Parameters

Configurations are present in the **<nginx-install-directory>/conf/conf.d/maps.conf** file.

```
## These two constants indicate five auth failures from a client can be allowed in thirty
seconds.
## if the threshold is crossed,client ip will be blocked.
map $host $auth_failure_threshold_for_lock {
## Must-change Replace below two parameters as per requirement
default 5 ;
}
map $host $auth_failure_counting_window_secs {
## Must-change Replace below two parameters as per requirement
default 30;
}
## This indicates duration of blocking a client to avoid brute force attack
map $host $ip_blocking_duration {
## Must-change Replace below parameter as per requirement
default 1800;
}
```

## Logging

The details of the blocked IP addresses can be accessed from the files **<nginx-install-directory>/logs/blocking.log** and **<nginx-install-directory>/logs/error.log**. To find the IP addresses that are blocked, run the following commands from the directory **<nginx-install-directory>/logs.**

```
grep "will be blocked for" blocking.log
grep "IP is already blocked." error.log
2021/10/29 17:30:59 [emerg] 1181750#1181750: *19 [lua] block_unauthorized_users.lua:153:
_redirectAndSendError(): 10.68.218.190
will be blocked for 30 minutes for exceeding retry limit., client: 10.68.218.190, server:
saproxy.cisco.com, request: "GET
```

```
/finesse/api/SystemInfo?nocache=1636456574482 HTTP/2.0", host: "saproxy.cisco.com:8445",
referrer:
"https://saproxy.cisco.com:8445/desktop/container/?locale=en_US&"

2021/10/29 19:21:00 [error] 943068#943068: *43 [lua] block_unauthorized_users.lua:53:
10.70.235.30 :: IP is already blocked...,
client: 10.70.235.30, server: saproxy.cisco.com, request: "GET
/finesse/api/SystemInfo?nocache=1635591686497 HTTP/2.0", host:
"saproxy.cisco.com:8445", referrer:
"https://saproxy.cisco.com:8445/desktop/container/?locale=en_US"
```

It is recommended that customers integrate with **Fail2ban** or a similar intrusion prevention system to add the blocked IP addresses to the IPtable or firewall rules.

# Install and Configure Fail2ban

Fail2ban can be configured to monitor the `blocking.log` to identify the IP addresses that are blocked by OpenResty Nginx on detecting brute force attacks, and ban the IP addresses for a configurable duration. Do the following to install and configure Fail2ban on a CentOS reverse-proxy:

### Procedure

**Step 1** Install Fail2ban using yum

```
yum update && yum install epel-release yum install fail2ban
```

**Step 2** Create a local jail

Jail configurations allow the administrator to configure various properties such as the ports that are to be banned from being accessed by any blocked IP address, the duration for which the IP address stays blocked, the filter configuration used for identifying the blocked IP address from the log file monitored, and so on. Steps to add a custom configuration for banning IP addresses that are blocked from accessing the upstream servers are as follows:

**a.** Go to Fail2ban installation directory (in this example /etc/fail2ban)

```
cd /etc/fail2ban
```

**b.** Make a copy of jail.conf into jail.local to keep the local changes isolated.

```
cp jail.conf jail.local
```

**c.** Add the following jail configurations to the end of the file jail.local, and substitute the ports in the template with the actual ones. Update ban time configurations as required.

```
# Jail configurations for HTTP connections.
[finesse-http-auth]
enabled = true
# The ports to be blocked. Add any additional ports.
port = http,https,<finesse-ports>,<cuic-ports>,<any-other-ports-to-be-blocked>
# Path to nginx blocking logs.
logpath = /usr/local/openresty/nginx/logs/blocking.log
# The filter configuration.
filter = finesseban
# Block the IP from accessing the port, once the IP is blocked by lua.
maxretry= 1
# Duration for retry set to 3 mins. Doesn't count as the maxretry is 1
findtime= 180
```

```
# Lock time is set to 3 mins. Change as per requirements.
bantime = 180
```

**Step 3** Configure a filter

A filter tells Fail2ban what to look for in the logs to identify the host to be banned. The steps to create a filter is as follows:

**a.** Create filter.d/finesseban.conf

```
touch filter.d/finesseban.conf
```

**b.** Add the following lines into the file filter.d/finesseban.conf

```
[Definition] # The regex match that would cause blocking of the host. failregex = <HOST>
will be blocked for
```

**Step 4** Start Fail2ban

Run the following command to start fail2ban:

```
fail2ban-client start
```

Open fail2ban log files and verify that there are no errors. By default, logs for fail2ban go into the file `/var/log/fail2ban.log`.

# Troubleshoot

## Troubleshoot SELinux

**Procedure**

**Step 1** If OpenResty Nginx is not started by default or the Finesse Agent Desktop is not accessible, set SELinux to **permissive** mode with this command:

```
setenforce 0
```

**Step 2** Try to restart OpenResty Nginx with the **`systemctl restart nginx`** command.

**Step 3** All the violations will be available in **/var/log/messages** and **/var/log/audit/audit.log**.

**Step 4** You are required to regenerate the **.te** file with allow rules for addressing those violations by executing any one of the following commands:

- `cat /var/log/audit/audit.log | audit2allow -m nginx1 > nginx1.te. # this will create nginx1.te file`
- `ausearch -c 'nginx' --raw | audit2allow -M  my-nginx  # this will create my-nginx.te file`

**Step 5** Update the original **nginx.te** file present in the **selinux-nginx-rhel-master-modified/nginx** directory with the newly generated allow rules.

**Step 6** Compile the **nginx.te** file by using the **make** command.

**Step 7** The **nginx.pp** file is regenerated.

**Step 8** Load the policy by using the **semodule** command.

```
semodule -i nginx.pp
```

**Step 9** Change SELinux to **enforce** mode by using the **setenforce** command.

**Step 10** Reboot the system.

**Step 11** Repeat this procedure until all the violations are fixed.

Reverse-Proxy Configuration