Cisco Packaged Contact Center Enterprise Features Guide, Release 12.6(2)

First Published: 2023-04-28

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

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

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

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

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

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

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

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

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

Preface

Change History

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

Change	See	Date
Added the URL of the Webex Connect datacenter in Canada.	Integrate Cloud Connect with Webex Connect	07 August, 2023

Change	See	Date
Initial Release of Document for Release 12.6(2)		**April, 2023**
A new chapter has been added that has information about how to provision and work with digital channels.	Digital Channels Integration using Webex Connect	
The VPN-less Access to Finesse Desktop section that was in the Mobile Agent chapter has been made into a separate chapter with additional details added.	VPN-less Access to Finesse Desktop	
A new chapter has been added that provides information about how to set up and use the Virtual Agent-Voice Call Transcription feature.	Virtual Agent–Voice Call Transcription	
The **Virtual Agent-Voice** and the **Virtual Agent–Voice for Dialogflow CX** chapters have been combined into one chapter. This chapter also includes information about cloud-based connectors.	Virtual Agent-Voice	
Cisco IdS now supports an asymmetric key encryption for signing the tokens generated for authentication.	Single Sign-On	
A new appendix has been added. This appendix replaces the **Reverse-Proxy Configuration** appendix.	Reverse Proxy Automated Installer	
A new appendix has been added that has guidelines for custom reverse proxy deployment.	Guidelines for Custom Reverse Proxy Deployment	

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. For example: • Choose **Edit** > **Find**. • Click **Finish**.
italic font	Italic font is used to indicate the following: • To introduce a new term. Example: A *skill group* is a collection of agents who share similar skills. • A syntax value that the user must replace. Example: IF (*condition, true-value, false-value*) • 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: • Text as it appears in code or that the window displays. Example: `<html><title>Cisco Systems, Inc. </title></html>`
`< >`	Angle brackets are used to indicate the following: • For arguments where the context does not allow italic, such as ASCII output. • A character string that the user enters but that does not appear on the window such as a password.

CONTENTS

PREFACE

CHAPTER 1

CHAPTER 2

CHAPTER 1

Optional Features in Packaged CCE

Feature Descriptions

You can choose to enable these features at any time after your Packaged CCE system is installed, configured, and operational.

Agent Greeting

Agent Greeting manages the recording and playing of greeting messages from agents. An agent's recorded greeting plays automatically to callers when they connect to that agent. Agents can set up different greetings for different types of callers, if the call center supports that option.

Agent Request

Agent Request allows a customer to make request on the web to receive a return call from an agent. The request is initiated by a Customer Collaboration Platform callback feed.

Courtesy Callback

Courtesy Callback offers customers the option to hang up and then receive a callback when an agent is close to being available, rather than having to wait for an extended time on hold. Customers do not lose their place in the queue. The system collects callback information from the caller, monitors agent availability, and calls the customer when the agent is close to available.

Enterprise Chat and Email

Enterprise Chat and Email (ECE) is an optional feature that provides chat and email functionality to the contact center. The ECE server routes chat and email contacts to agents on their Cisco Finesse desktops. The ECE server can be installed on the Packaged CCE Side B host or on an external server.

ECE includes the following features:

- Email—Email is supported by ECE to create a communication channel between a customer and an agent. There are various steps involved in efficiently responding to emails from customers. Emails are first retrieved into the system and routed to appropriate users or queues. After a response is created, it is processed through the system and sent to the customer.

- Chat—A chat is a real-time interaction between an agent and a customer during which they exchange text messages. As part of a chat, agents can also push web pages to customers. Based on how chat activities are routed to agents, they can be categorized as standalone chats or integrated chats. An integrated chat is routed to an integrated queue and a message is sent to Packaged CCE. The system processes the activity and assigns the chat to an available agent.

- Web Callback—The Web Callback feature allows the user to request a callback by submitting a form on a website. ECE processes the submitted information and connects the user with an agent. ECE then sends a request to Packaged CCE to route the callback request to the agent.
- Delayed Callback—The Delayed Callback feature is similar to Web Callback, but when ECE receives the delayed callback request, it adds the request in the Delayed Callback table. ECE sends the HTML page to the customer, indicating that the customer will receive a callback within a specified time. When the specified time arrives, ECE moves the request to the Packaged CCE queue for routing to Unified CCE.

For more information about this feature, see the Enterprise Chat and Email documentation at https://www.cisco.com/c/en/us/support/customer-collaboration/cisco-enterprise-chat-email/tsd-products-support-series-home.html.

Extension Mobility and Extension Mobility Cross Cluster

Extension Mobility and Extension Mobility Cross Cluster are Cisco Unified Communications Manager features that allow agents to temporarily access their Cisco Unified IP Phone configuration, such as line appearances, services, and speed dials, from other Unified IP Phones.

Extension Mobility works on phones that are located within the same Unified Communications Manager cluster. Extension Mobility Cross Cluster works on phones that are located in different Unified Communications Manager clusters.

As part of the configuration in **Unified Communications Manager Administration**, you create a device profile for each agent that will use Extension Mobility, and associate each device profile with the appropriate agent. You can add either all of the device profiles to the pguser, or all of the phones that the agents use to the pguser. You do not need to add both the profiles and phones to the pguser.

For more information, see the Extension Mobility section of the *Feature Configuration Guide for Cisco Unified Communications Manager* at https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/products-installation-and-configuration-guides-list.html.

Mobile Agent

Mobile Agent supports call center agents who are using phones that are not directly controlled by Packaged CCE. A mobile agent can be located outside of the contact center, using an analog phone or a mobile phone. Mobile agents may also be located in the contact center using an IP phone connection that is not controlled by Packaged CCE. All mobile agents use broadband connection to access the same Agent desktop and agent state controls as non-mobile agents.

Packaged CCE supports both Call by Call and Nailed Connection mode.

Outbound Option

Outbound Option manages and performs outbound dialing campaigns. You configure the system to automatically dial numbers using imported contact lists and filtering rules. When a call connects to a live person, the system transfers the call to an agent skilled in handling that calling campaign.

Post-Call Survey

Post-Call Survey sends a caller to an automated survey after the agent disconnects. A Post-Call Survey is typically used to determine whether customers are satisfied with their call center experiences.

Single Sign-On

Single sign-on (SSO) is an authentication and authorization process. (Authentication proves you are the user you say that you are, and authorization verifies that you are allowed to do what you are trying to do.) SSO allows users 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 to 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.

Task Routing for Third-Party Multichannel Applications

Task Routing application programming interfaces (APIs) provide a standard way to request, queue, route, and handle third-party multichannel tasks in CCE.

Contact Center customers or partners must develop Customer Collaboration Platform and Finesse applications using these APIs to use the Task Routing feature. The Customer Collaboration Platform application submits nonvoice task requests to CCE. The Finesse application enables agents to sign in to different types of media and handle the tasks. Agents log in to and manage their state in each media independently.

Whisper Announcement

Whisper Announcement plays a brief message to an agent before connecting a caller to the agent. The message may include information about the caller or choices the caller made from menu options.

Video Contact Center

Video Contact Center provides high-quality video collaboration between customers and agents. Depending on how Video Contact Center is deployed, customers may connect with agents either from within the enterprise network or from devices outside the enterprise.

Avaya Support

Support for Avaya integration has been provided in Packaged CCE 4000 and 12000 Agent deployments. You can maintain an Avaya Peripheral Gateway (PG) in a Packaged CCE environment and use its intelligent contact center routing capability to route calls to geographically distributed contact center sites.

ICM-to-ICM Gateway Support

Support for ICM-to-ICM Gateway has been provided in Packaged CCE 4000 and 12000 Agent deployments. ICM-to-ICM Gateway extends the ICM software capability by allowing agents to simultaneously pre-route/post-route calls, and supply additional call-related information to a second agent on a different ICM. This enables the initial agent to pass on gathered information without the customer’s needing to repeat it to the second agent.

Integrations with Other Cisco Products

You can extend Packaged CCE functionality by integrating it with other Cisco products.

Cisco Silent Monitoring

Silent monitoring allows a supervisor to listen in on agent calls for quality control and performance evaluation. Packaged CCE supports Unified CM-based silent monitoring.

Supervisors can start Unified CM-based silent monitor sessions by selecting an agent on the Team Performance page on their Cisco Finesse desktops and clicking **Start Monitoring**. They can then click **End** to end the session.

Cisco Customer Collaboration Platform

Cisco Customer Collaboration Platform is a customer-care system that provides capture, filtering, workflow, queuing, and reporting for social media engagement teams. Internet postings captured by Customer Collaboration Platform are referred to as Social Contacts. Customer Collaboration Platform stores the social contacts and groups them into user-defined Campaigns. Each Campaign obtains social contacts from one or more Feeds. Customer Collaboration Platform presents the social contacts to social media customer care personnel who can search, review, categorize, and respond to the postings. Customer Collaboration Platform also produces reporting metrics on the handling of the social contacts.

Customer Collaboration Platform is also used for the following contact center features:

- Agent Request
- Task Routing

For information about Customer Collaboration Platform, see https://www.cisco.com/en/US/products/ps11349/index.html.

Assumptions for Proceeding with Optional Features

This document makes the following assumptions about the state of your Packaged CCE system and the system administrator's knowledge of Packaged CCE:

- Your Packaged CCE system must be installed, configured, and operational.
- System administrators must have access to the following interfaces:
  - Cisco Packaged Contact Center Enterprise (CCE) Administration
  - Script Editor
  - Cisco Customer Collaboration Platform
  - Cisco Finesse
  - Cisco Unified Communications Manager (CUCM) reporting interface
  - Enterprise Chat and Email
- System administrators must be familiar with the following procedures or have access to the Cisco documentation that describes them:
  - Expanded call variables—Know how to use Unified CCE Administration to set variable values and add new variables.
  - Scripting—Know how to use the Script Editor to create new Packaged CCE call routing scripts and modify existing scripts. Understand the scripting technology.
  - CVP scripting—Know how to use the CVP Script Editor to create new or modify existing voice scripts.

The *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* describes all of the above procedures. This guide is available at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

CHAPTER 2

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
- Ensure that vCUBE has access to the following websites:
    - *.cisco.com
    - *.ciscospark.com
    - *.rtmsprod.net
    - *.wbx2.com
- Ensure that Packaged CCE AW, Cloud Connect, CUBE, and Agent Desktop components have access to Webex services to use the Agent Answers.

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.

SUMMARY STEPS

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.
2. Ensure that the Cloud Connect publisher and subscriber are installed.
3. Configure Cloud Connect in Unified CCE Administration. For details, see *Configure Cloud Connect* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.
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.
5. Import the Cloud Connect certificate to the CVP Server.
6. In the Unified CCE Administration console, do the following with the CCAI configuration (created in step 1):
7. To add the Agent Answers gadget to the Cisco Finesse desktop layout:
8. Perform the following steps to configure WebSocket-based forking in CUBE.

DETAILED STEPS

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 *Create VM for Cloud Connect Publisher* and *Create VM for Cloud Connect Subscriber* sections in *Cisco Packaged Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-installation-guides-list.html.

Step 3 Configure Cloud Connect in Unified CCE Administration. For details, see *Configure Cloud Connect* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-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 *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-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 11.

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 11.

Step 7 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 12.

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 8 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 15.

b) Import the WebSocket Connector certificate to CUBE. For details, see Import or Verify WebSocket Connector Certificate to CUBE, on page 15.

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 Packaged 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.

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.

Associate Contact Center AI Configuration

Before you begin

This procedure explains how to associate the Contact Center AI configuration with a call type.

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

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

Step 3 Click **New** to open the **New Call Type** window.

Step 4 Complete the mandatory fields in the **General** tab. For more information, see the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Step 5 Click the **Contact Center AI** tab.

Step 6 In the **Contact Center AI Configuration** search box, click the search icon. A pop-up window displays a list of CCAI configurations.

Step 7 Select the required configuration and click **Save**.

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.

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 updated.

Step 3 Click the **Call Type** tab and select the call type for which Contact Center AI configuration has to be associated.

Step 4 Click the **Contact Center AI** tab.

Step 5 In the **Contact Center AI Configuration** search box, click the search icon. A pop-up window displays a list of CCAI configurations.

Step 6 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.

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.

Configure Contact Center AI Services for an Agent

Administrators can configure Contact Center AI Services for an agent while adding the agent. Supervisors can only enable or disable the services for an agent.

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

Step 2 Click **New** to open the **New Agent** page.
This page has: **General**, **Attributes**, **Skill Groups**, **Supervised Teams**, **Enable Email & Chat**, and **Contact Center AI** tabs.You cannot save the agent until you have entered all required fields on the **General** tab. You can complete other tabs as needed and in any order. For more information, see *Add and Maintain Agents* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Step 3 Click the **Contact Center AI** tab.
Displays a list of services 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 an Agent

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

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.

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

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 14.

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. 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. 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, VAV Transcript, and Transcript. To associate more than one services, seperate the values using semicolon (;). 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**.

SUMMARY STEPS

1. Run the command to verify if the certificate is included.
2. If **IdenTrust** certificates are not present, add the certificates to CUBE.

DETAILED STEPS

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 Packaged CCE 12.6

Before you begin

In Packaged 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 Packaged 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 Packaged CCE 12.6. No changes are required to the existing Google CCAI, CUBE, or Cloud Connect configurations.

While configuring the CCAI services in Packaged 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 Packaged CCE 12.6 and leverage the enhanced capabilities listed above:

SUMMARY STEPS

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.
2. In the Unified CCE Administration console, do one of the following:
3. In the Unified CCE Administration console, enable or disable CCAI Services for an agent or multiple agents.
4. Undo these CCAI settings configured when Packaged CCE was in 12.5:

DETAILED STEPS

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 11.
- 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 11.

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 12.

Step 4 Undo these CCAI settings configured when Packaged 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 Packaged Contact Center Enterprise Features Guide, Release 12.5* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-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 3

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/packaged-contact-center-enterprise/products-device-support-tables-list.html for the list of supported Packaged CCE 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 311

Initial Setup

This section is intended for system administrators responsible for installing and configuring Packaged 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 Defaul Administration, select **Device** > **Phone** > **Built in Bridge**.

Where	What
Unified CCE	Agent Greeting is supported with Type 10 Network VRUs only. (Type 10 is required to al is not configured for a Type 10 VRU, you must modify it accordingly. Agent Greeting requires at minimum three expanded call variables. • user.microapp.ToExtVXML: This is used twice in an Agent Greeting record script: application; the second time is to tell the recording application where to save greetin Use the Unified CCE Administration tool to ensure this variable includes these setti • user.microapp.app_media_lib:This is required in Agent Greeting record and play scr greeting audio files are stored. Maximum Length - 100 and Enabled. • user.microapp.input_type: This is required in Agent Greeting record scripts to limit th No other ECC (Expanded Call Variable) are needed if you serve your files from the Unif default locale directory ("<*web_server_root*>\en-us\app"). However, if you store your fil 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 ECC variables for CVP, see the *Administration Guide for Cisco Unified Customer Voice Po* unified-customer-voice-portal/tsd-products-support-series-home.html. • user.microapp.media_server: Use to identify the Unified CVP media server if it is o • user.microapp.locale: Use to specify the name of the locale directory on the media s • user.microapp.UseVXMLParams: Required in your record script if you include the 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 Package* https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-

Deploy Agent Greeting

Agent Greeting Deployment Tasks

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.

For more information, see *Setup Unified CVP Media Server IIS* section in the *Cisco Packaged Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-installation-guides-list.html.

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

For more information, see *Set Up IVR Service* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Step 4 Configure a Unified CVP media server, if you have not already done so. See Media Server, on page 117.

Step 5 Set the cache size on the VXML Gateway. See Set Cache Size on VXML Gateway, on page 23.

Step 6 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 23.

Step 7 Configure call types to record and play agent greetings. See Configure Call Types, on page 24.

Step 8 Configure dialed numbers to record and play agent greetings. See Configure Dialed Numbers, on page 24.

Step 9 Schedule the Script, on page 25.

Step 10 Define Network VRU Scripts for Agent Greeting, on page 25.

Step 11 In Script Editor:

- To use the installed scripts to record and play agent greetings, see Import Example Agent Greeting Scripts, on page 26.
- To create your own scripts, see Agent Greeting Scripts, on page 29.

Step 12 Modify the Unified CCE call routing scripts to use Play Agent Greeting script, on page 28.

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 28.
  - 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.

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:

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

The built-in prompts are installed on each server at <*CCE_root*>`\wav` and are referenced in the example recording script that is included with Packaged CCE. To deploy the example script, copy the audio prompts to the <*web_server_root*>`\en-us\app` directory on your media server.

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.

Configure Call Types

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

Step 1 In Unified CCE Administration, choose **Overview** > **Call Settings** > **Route Settings**.

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

Step 3 Click **New** to open the **New Call Type** window.

Step 4 Complete the fields to create a call type to record agent greetings. For **Name**, enter **`RecordAgentGreeting`**.

Step 5 Click **Save**.

Step 6 Repeat this procedure to create a call type to play agent greetings. For **Name**, enter **`PlayAgentGreeting`**

Configure Dialed Numbers

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

Step 1 In Unified CCE Administration, choose **Overview** > **Call Settings** > **Route Settings**.

Step 2 Click the **Dialed Number** tab.

Step 3 Click **New** to open the **New Dialed Number** window.

Step 4 Complete the fields to create a dialed number to record agent greetings, as follows:

- **Dialed Number String:** RecordAgentGreeting

  The name must match exactly and is case-sensitive.

- **Routing Type:** Internal Voice
- **Call Type:** RecordAgentGreeting (the call type that you created for recording agent greetings)

Step 5 Click **Save**.

Step 6 Repeat this procedure to create a dialed number to play agent greetings. Complete the **New Dialed Number** fields as follows:

- **Dialed Number String:** PlayAgentGreeting

  The name must match exactly and is case-sensitive.

- **Routing Type:** Internal Voice
- **Call Type:** PlayAgentGreeting (the call type that you created for playing agent greetings)

Schedule the Script

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, log into Packaged CCE Administration and select **Overview** > **Call Settings** > **IVR Settings** > **Network VRU Scripts**.

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.

Table 2: Agent Greeting Network VRU Scripts

Name / VRU Script Name	Configuration Parameter	Interruptible (Y/N)	What it does

`AgentGreeting`
`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.

Name / VRU Script Name	Configuration Parameter	Interruptible (Y/N)	What it does

`GreetingMenu_1_to_9`
`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`
`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`
`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`
`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.
			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.

`GreetingReview`
`PM,-a,A`
	`Y`	Y	This script allows the agent to review the recorded greeting audio file.

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:

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 29.

- 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 25) 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 25.
- **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 25.
- **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 25) 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.

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.

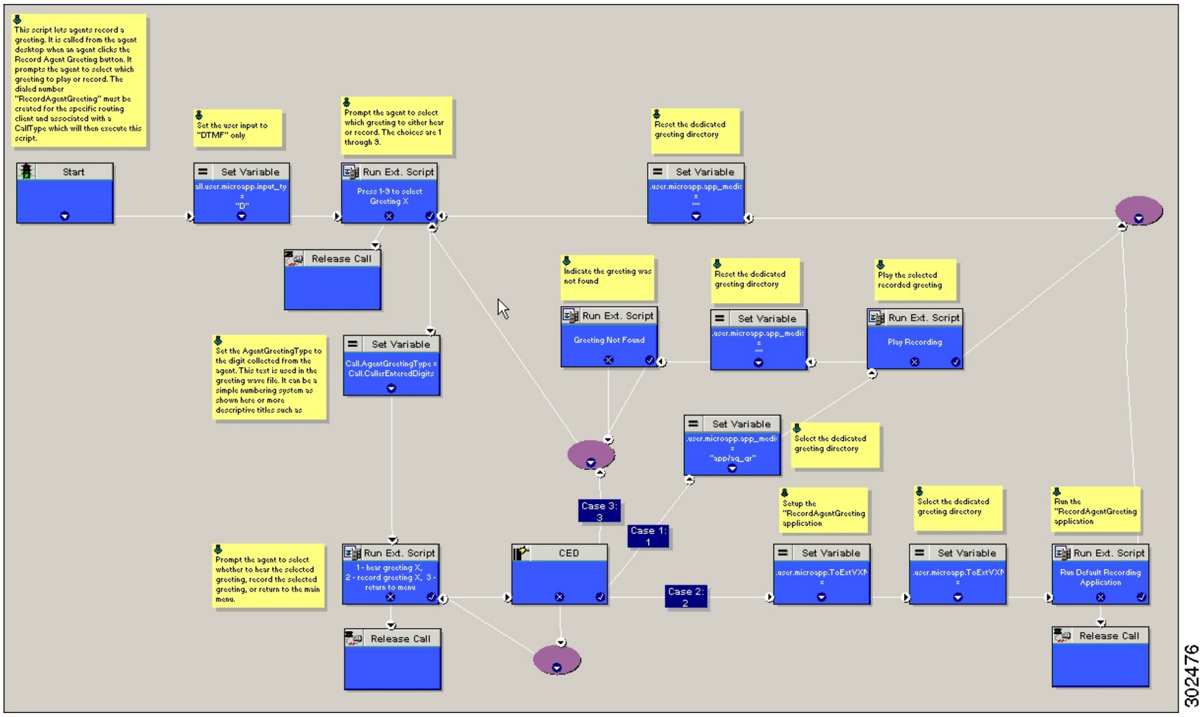
Agent Greeting Recording Script

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



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. The dialed number "RecordAgentGreeting" must be created for the specific routing client and associated with a CallType which will then execute this script.

Set the user input to "DTMF" only

Prompt the agent to select which greeting to either hear or record. The choices are 1 through 9.

Reset the dedicated greeting directory

Indicate the greeting was not found

Reset the dedicated greeting directory

Play the selected recorded greeting

Set the AgentGreetingType to the digit collected from the agent. This text is used in the greeting wave file. It can be a simple numbering system as shown here or more descriptive titles such as

Select the dedicated greeting directory

Setup the "RecordAgentGreeting application

Select the dedicated greeting directory

Run the "RecordAgentGreeting application

Prompt the agent to select whether to hear the selected greeting, record the selected greeting, or return to the main menu.

Start
Set Variable
all.user.microapp.input_ty = "D"
Run Ext. Script
Press 1-9 to select Greeting X
Set Variable
.user.microapp.app_medi: = ""
Release Call
Run Ext. Script
Greeting Not Found
Set Variable
.user.microapp.app_medi: = ""
Run Ext. Script
Play Recording
Set Variable
Call.AgentGreetingType = Call.CallerEnteredDigits
Set Variable
.user.microapp.app_medi: = "app/aq_qr"
Case 3: 3
Case 1: 1
Run Ext. Script
1 - hear greeting X, 2 - record greeting X, 3 - return to menu
CED
Case 2: 2
Set Variable
.user.microapp.ToExtVXM =
Set Variable
.user.microapp.ToExtVXM =
Run Ext. Script
Run Default Recording Application
Release Call
Release Call
302476

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. microapp.input_type	`D`	Sets the allowable input type to DTMF (touch tone).
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: 1 - hear greeting X, 2 - record greeting X, 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: - Once to `"app/ag_gr"` - 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: Call:user.microapp. ToExtVXML[]	Array Index: 2 Value: "ftpPath=*<path_to_dedicated/ directory>*" For example: "ftpPath=en-us/app/ag_gr"	Specifies the FTP information that the CVP Server uses to write greeting files to the media server. The value for array index must be 2. The value consists of: • ftpPath= to set the path to the dedicated directory for agent greeting files. • The path must begin with the locale directory.
Variable: Call:user.microapp. ToExtVXML[]	Array Index: 0 Value: "application=RecordAgentGreeting"	Identifies the external Unified CVP micro-application (RecordAgentGreeting) that is used to record the greeting. 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 CVP Server.

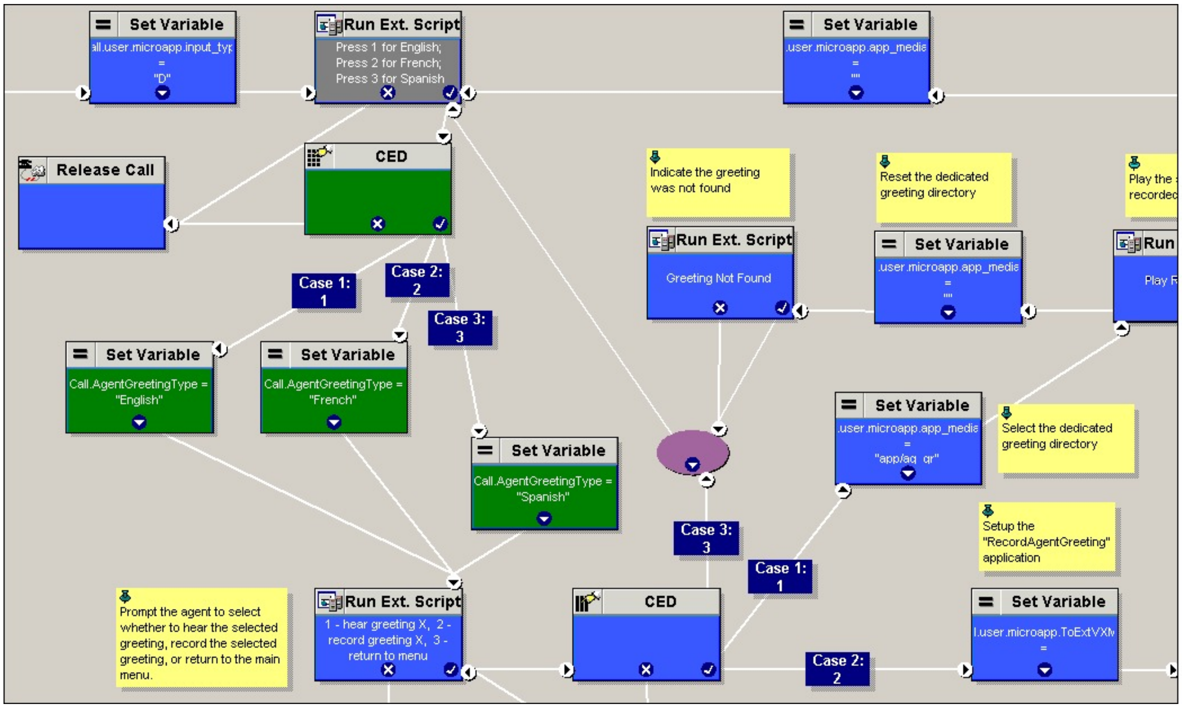
Descriptive Agent Greeting Type Strings

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 2: 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:

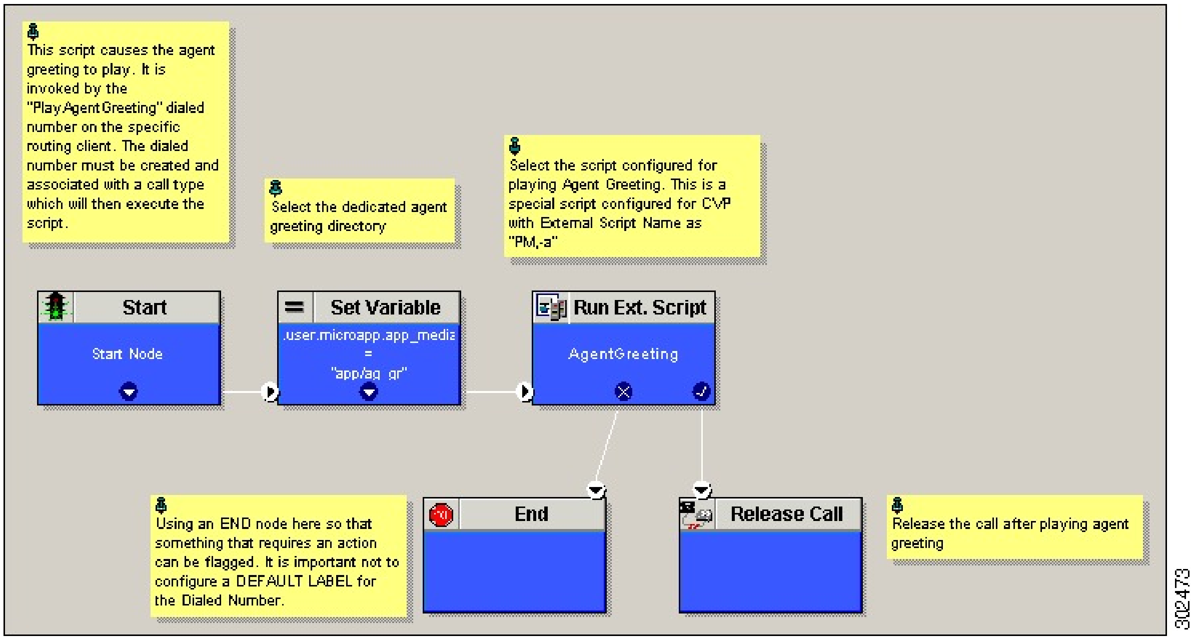
- A Run External Script node that calls the VRU script that plays the greeting.
- A Set Variable node that sets the directory path to your greeting files.

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 3: Agent Greeting Play Script Example



This script causes the agent greeting to play. It is invoked by the "PlayAgentGreeting" dialed number on the specific routing client. The dialed number must be created and associated with a call type which will then execute the script.

Select the dedicated agent greeting directory

Select the script configured for playing Agent Greeting. This is a special script configured for CVP with External Script Name as "PM,-a"

Using an END node here so that something that requires an action can be flagged. It is important not to configure a DEFAULT LABEL for the Dialed Number.

Release the call after playing agent greeting

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.

Administration and Usage

Use Agent Greeting with Your Finesse Desktop

Configure Custom Dialed Number for Finesse Agent Greeting Record

To record Agent Greetings with Finesse, create your own custom dialed number for recording. You may want to create different dialed numbers for different customers.

To record the greeting, your agents can enter the record dialed number using the dial pad on their desktops.

Use the following steps to create a custom dialed number for Finesse Agent Greeting Record:

1. Create a CTI Route Point in Unified CM and associate it with an Application User (PG User).
2. Create a Dialed Number in Unified ICM for the CTI Route Point created in Unified CM.

3. Create a Call Type for the Custom Dialed Number.
4. Associate the Call Type and Dialed Number with the Record Agent Greeting Script.
5. Create a Phonebook Entry in Finesse for the agent to dial the Custom Dialed Number.

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.

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 4

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.

Cisco Customer Collaboration Platform works in a Contact Center Enterprise (CCE) solution to process the request from its inception through the delivery of the callback.

Note Enterprise Chat and Email also offers callback and delayed callback. You can use Agent Request , Enterprise Chat and Email, or both.

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 CCE using a notification mechanism (the Connection to 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.

Note Customer Collaboration Platform supports scalar ECC variables only.

CCE and Agent Request

CCE services in the Agent Request solution:

- 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 Request Prerequisites

Install and configure Customer Collaboration Platform before implementing Agent Request. Customer Collaboration Platform must be geographically colocated with the Unified CCE PG on one side.

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 44.

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 39.

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 Unified CCE PG.
3. The Unified CCE PG sends the request to the agent.
4. 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 4: 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 Packaged CCE for Agent Request

Set up the Media Routing PG and PIM

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*.

Note In Packaged CCE 4000 and 12000 Agents deployment, fetch the Peripheral ID from the PG Explorer tool using Configuration Manager.

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. 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 for the unused Multichannel peripheral that you identified in Step 1.
d) 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.

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) Click **OK**.

Step 7 Accept defaults and click **Next** until the Setup Complete screen opens.

Step 8 At the Setup Complete screen, check **Yes** to start the service. Click **Finish**.

Step 9 Click **Exit Setup**.

Step 10 Repeat from Step 1 for Side B.

Step 11 Navigate to **Unified CCE Administration** > **Infrastructure Settings** > **Inventory**.

Step 12 Add Customer Collaboration Platform as an external machine.

a) Click **Add Machine**.
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.

Unified CCE Administration Tools

This topic explains the Unified CCE Administration tools you use to configure Agent Request.

Before you begin

For details on the procedures for steps 2 to 5, refer to the Unified CCE Administration online help or to the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/en/US/products/ps12586/prod_maintenance_guides_list.html.

Step 1 Sign in to Unified CCE Administration.

Step 2 **Call Type**: Create a call type for Agent Request.

Step 3 **Dialed Number**: Create a dialed number for Agent Request. You use this number when you configure the notification in Customer Collaboration Platform.

a) For **Routing Type**, select Customer Collaboration Platform.
b) For **Media Routing Domain**, select **Cisco_Voice**.
c) For **Call Type**, select the call type that you created in Step 2.

Step 4 **Expanded Call Variable**: You can use an existing Expanded Call Variable, or you can create an expanded call variable for Agent 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 Unified CVP, Finesse, and Customer Collaboration Platform.

Step 5 Create a **Network VRU Script**.

This network VRU script does not refer to a script that you create on a peripheral. This script satisfies a configuration requirement and provides a messaging vehicle to get the value of the estimated wait time to Customer Collaboration Platform using the MR PIM to fulfill the API call.

Use the Manage Network VRU Scripts gadget to create the new network VRU script. Choose a name (for example, VoiceCallback) and enter that name in both Name fields.

No configuration parameters are required for the network VRU script. Optionally, enter a description. In the remaining fields, leave the default values. You reference this configuration object when you configure the Run External Script node in the routing script.

Related Topics

Create Script for Agent Request, on page 42

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

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

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

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**.

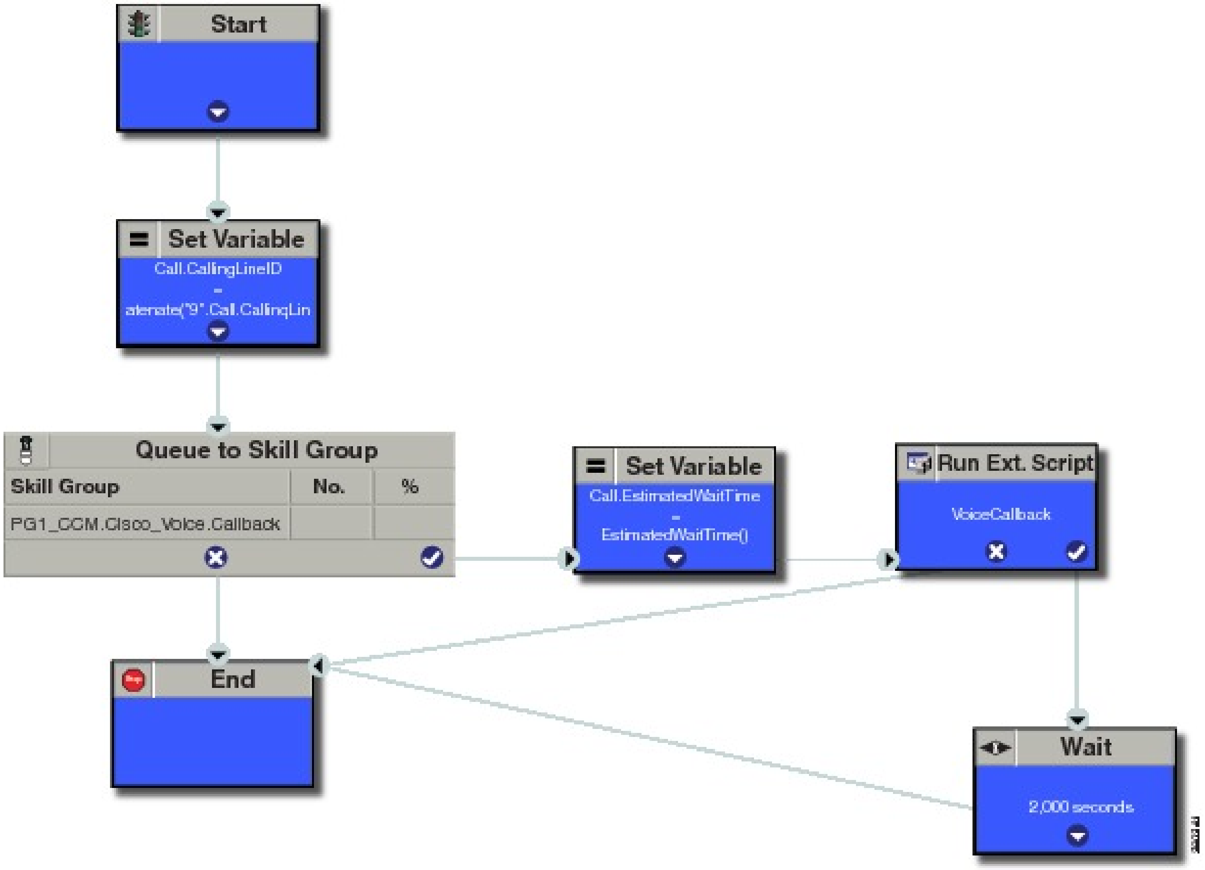
Step 9 In the **Dialed Number/Script Selector** field, enter the dialed number string that you have configured.

See Unified CCE Administration Tools, on page 40.

Step 10 Click **Save**.

Create Script for Agent Request

This illustration shows a sample script. The key below explains the nodes.



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

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.

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.

Set Variable (Call.Estimated Wait Time) node: 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 previous 5 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 recision 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 that the built-in function also applies to inbound calls that queue.

Set the Call Wait time as follows:

1. From the Set Variable node, select **Call** from the Object type drop-down menu.
2. 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.
3. Click **Formula Editor**, and do either of the following:
   - To use the default estimated wait value, click the Built-In Functions tab and choose EstimatedWaitTime()
   - To create a formula and use your own value, click the Variables tab and choose an entry in the Object type list and an entry in the Object list. Then double-click a variable in the Variable list.

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

1. Click the Queue tab.
2. Click **Run External Script**.
3. Click inside the script. A Run External Script node appears.
4. Double-click the node and choose the Network VRU script from the list; 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.

Wait node: The wait period before an agent becomes available.

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

Related Topics

Unified CCE Administration Tools, on page 40

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.

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.

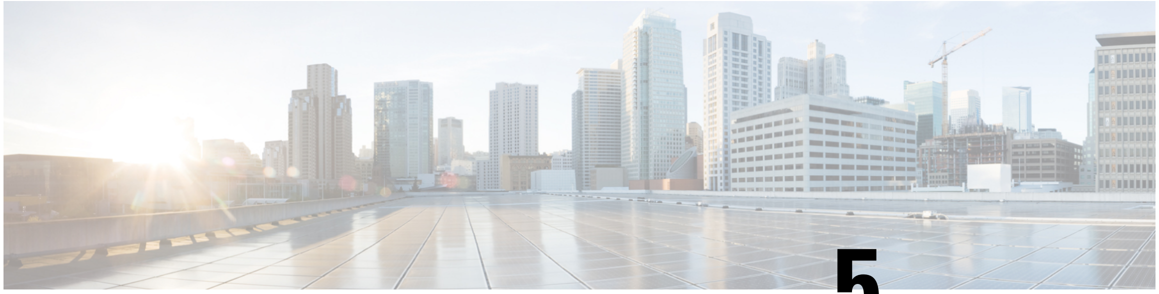
Agent Real Time

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 5

Application Gateway

About Application Gateways

An application gateway is an optional Packaged CCE feature that allows you to invoke an external application from within a script (using a Gateway node). You can pass data to the application and receive data in return, which you can then examine and use for routing decisions.

Before you can use these nodes in a script, you must first configure the gateways.

The application gateway requires connection information to communicate with the external application. You perform this task using the Unified CCE Administration interface.

Configuring Application Gateways

Configure a application gateway for an application you want to access, from within the scripts.

Configuration information includes data such as:

- Type of application the gateway interacts with-a non-Packaged CCE application or an application on another Packaged CCE system
- Form of connection the gateway uses-duplex or simplex
- Fault tolerance strategy for the gateway-described in the following table.

Table 4: Application Gateway Fault Tolerance Strategies

Fault Tolerance Strategy	Description
Duplicate Request	Packaged CCE, both side A and B, connects to separate application gateway hosts. They send simultaneous requests. Each request is sent to both the sides of the gateway. The response that comes back first, is used by both the sides of A and B of ICM.

Fault Tolerance Strategy	Description
Alternate Request	Packaged CCE, Side A and Side B connects to separate application gateway hosts. All requests are sent alternatively to A and B.
Hot Standby	Each router manages a connection to a different host. All requests are directed to the designated primary host. If either host (or connection) fails then all requests are directed to the backup host. This results in the loss of some requests on failures.
None	The application gateway is not duplexed.

Once you specify the configuration information, you can define the connection information for the gateway. For example, the network address of the port, through which the system software communicates with the application.

If your Central Controller is duplexed, you can define separate connection information for each side of the Central Controller. This allows each side to communicate with a local copy of the external application.

Add and Maintain Application Gateways

You can create custom application gateways in Packaged CCE deployment.

Step 1 In Unified CCE Administration, choose **Overview** > **Infrastructure Settings** > **Application Gateways**.

Step 2 To add a new gateway, click **New**.
The **New Application Gateway** page displays.

Step 3 Enter a name for the new application gateway. Maximum length is 32 characters. Valid characters are alphanumeric, period (.), and underscore (_). The first character must be alphanumeric.

Step 4 Enter a description.

Step 5 Select one of the options from the **Encryption** drop-down list.

- **None**: Selected by default. Indicates that the requests are not encrypted.
- **Private Key**: Indicates that the requests are encrypted using a private key.
- **TLS**: Indicates that the requests are encrypted using the TLS protocol.

Step 6 Select one of the options from the **Connection Type** drop-down list.

a) If you select **Duplex**, you can enter data in all the subsequent fields.
b) If you select **Simplex A**, the **Preferred Side** is set to **Side A**, **Fault Tolerance** is set to **None**, and the **Side B** box is disabled.
c) If you select **Simplex B**, the **Preferred Side** is set to **Side B**, **Fault Tolerance** is set to **None**, and the **Side A** box is disabled.

Step 7 Select a side from the **Preferred Side** drop-down list.

Step 8 Select one of the options from the **Fault Tolerance** drop-down list.

- **Alternate Request**: Each router manages connections with different hosts. The routers take turns to send half the request to the host connected to Side A and the other half to the host connected to Side B. If one host fails, the entire load is directed to the surviving host.

- **Duplicate Request**: Each router manages connections with different hosts. Each time a script initiates a request, both the routers communicate with their corresponding hosts. The routers process the response from the host that responds first.
- **Hot Standby**: Each router manages connections with different hosts. All the requests are directed to the designated primary host. If the host or the connection fails, all the requests are directed to the backup host.

Step 9 Enter the following server details in the **Side A** and **Side B** boxes as applicable:

- **In Service**: This option is enabled by default. If you uncheck the **In Service** check box, the connection is no more in service and the router does not send application gateway requests to that connection.
- **Hostname/IP Address**: Enter the IP address, hostname of the server, or fully qualified domain name (FQDN).
- **Port**: Enter the port number.
- **Initialization Data**: This information passes to the Application Gateway host at the time of initialization.

Step 10 Click **Advanced Settings** on Side A or Side B to open the respective **Advanced Settings** dialog box. The following parameters with default values appear:

- **Max Errors**: Indicates the number of consecutive errors that cause the software to declare the host unavailable.
- **Timeouts**
  - **Request**: Indicates the number of milliseconds the Router waits before timing out a request.
  - **Abandon**: Indicates the number of milliseconds the Router waits for a response before considering it as late.
  - **Late**: An internal timeout in milliseconds to communicate between the Router and the Application Gateway interface process.
- **Heartbeats**
  - **Request Timeout**: Indicates the number of milliseconds the Router waits for a response to a heartbeat before considering it as a failure.
  - **Retry Timeout**: Indicates the number of milliseconds the Router waits before retrying a missed heartbeat.
  - **Retry Limit**: Indicates the number of consecutive unanswered heartbeats after which the Router ends the connection.
  - **Interval**: Indicates the number of milliseconds the Router waits between successful heartbeats.
- **Sessions**
  - **Retry Timeout**: Indicates the number of milliseconds the Router waits before trying to reconnect after a connection terminates or a connection attempt fails.
  - **Retry Limit**: Indicates the number of times the Router tries to establish the connection before it quits.
  - **Open Timeout**: Indicates the number of milliseconds the Router waits for a response to an open or close request. If it receives no response within this time, the Router assumes that the request failed.

Step 11 Edit the advanced settings parameters as applicable and click **OK**.

Note Click the **Restore Defaults** button to restore the default values.

Step 12 Click **Save**.

CHAPTER 6

Business Hours

Business Hours Overview

Business hours are the working hours during which you conduct business. You can create and modify business hours and set weekly and daily schedules for each business hour. You can create different business hour schedules for regular working days and holidays. You can also open or close the business hours if there is an emergency.

You can 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

Packaged CCE supports Business Hours for these reference designs:

- 2000 Agents
- 4000 Agents
- 12000 Agents

Business Hours Use Cases

Use the Business Hours feature to manage the incoming customer calls or digital channel communications, 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.

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.

The credential must be of a domain user who is a member of the local administrator group if `ADSecurityGroupUpdate` registry key in AW is zero. If `ADSecurityGroupUpdate` registry key is set to 1, then the user must be available in the Config security group under the instance OU. These credentials must be valid on all CCE components in your deployment (routers, PGs, AWs, and so on).

Note Every time the Active Directory credentials are updated, the credentials configured here must be updated as well.

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 5: 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	Cisco Packaged Contact Center Enterprise Administration and Configuration Guide at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-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.

- **Department**: Associate the business hours to a department so that the change to business hours is restricted to only the user of that department.

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.

Note In case, any new Timezone definition is added or updated at the Windows Operating System (OS) through patch, apply that OS patch at both Side A and Side B at the same time.

Business Hours Status Evaluation

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

1. Force Open or Close
2. Special Hours and Holiday schedule
3. Open/Close as per Business Calendar

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 7

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.

In addition, you can view the transcript of the voice conversation between the customer and virtual agent that happened before the call was transferred to you. You can also view a **Highlights** panel that displays the intents and intent parameters based on customer's query. The **Highlights** panel also displays a confidence score for each intent, and a customer sentiment indicator for each intent and for the entire call.

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.

Note Ensure that the Packaged CCE AW, Cloud Connect, CUBE, and Agent Desktop components have access to Webex services to use the Call Transcription.

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.

SUMMARY STEPS

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.
2. Ensure that the Cloud Connect publisher and subscriber are installed.
3. Configure Cloud Connect in Unified CCE Administration. For details, see *Configure Cloud Connect* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.
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.
5. Import the Cloud Connect certificate to the CVP Server.
6. In the Unified CCE Administration console, do the following with the CCAI configuration (created in step 1):
7. To add the Call Transcript gadget to the Cisco Finesse desktop layout:
8. Perform the following steps to configure WebSocket-based forking in CUBE.

DETAILED STEPS

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 *Create VM for Cloud Connect Publisher* and *Create VM for Celoud Connect Subscriber* sections in *Cisco Packaged Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-installation-guides-list.html.

Step 3 Configure Cloud Connect in Unified CCE Administration. For details, see *Configure Cloud Connect* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-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 *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-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 11.
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 11.

Step 7 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 59.

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 8 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 15.
b) Import the WebSocket Connector certificate to CUBE. For details, see Import or Verify WebSocket Connector Certificate to CUBE, on page 15.
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 Packaged 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.

Configure Contact Center AI Services for an Agent

Administrators can configure Contact Center AI Services for an agent while adding the agent. Supervisors can only enable or disable the services for an agent.

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

Step 2 Click **New** to open the **New Agent** page.
This page has: **General**, **Attributes**, **Skill Groups**, **Supervised Teams**, **Enable Email & Chat**, and **Contact Center AI** tabs.You cannot save the agent until you have entered all required fields on the **General** tab. You can complete other tabs as needed and in any order. For more information, see *Add and Maintain Agents* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Step 3 Click the **Contact Center AI** tab.
Displays a list of services 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 an Agent

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

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.

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

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 14.

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. 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. 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, VAV Transcript, and Transcript. To associate more than one services, seperate the values using semicolon (;). 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 8

Digital Channels Integration Using Webex Connect

Overview

Today's customers want to connect with businesses through any communication channel of their choice. Webex Connect allows the Contact Center business and its customers to interact using digital channels such as email, chat, and SMS.

The Contact Center Enterprise (CCE) solution integrates with Webex Connect to create a seamless omnichannel experience for your customers. This integration helps your customers to interact across voice and digital channels of communication.

Webex Connect offers a rich self-service and bot integration to empower your customers to get answers to some common questions. It provides a unified solution for integrated routing, Agent Desktop, and reporting service. Webex Connect provides a simplified framework that helps partners and customers interact through digital channels.

Prerequisites

The following are the prerequisites for provisioning digital channels for Contact Center Enterprise:

- The following components must be on release 12.6(2) or higher: CCE components (Router, Logger, AW, and PG), Cisco Finesse, Cisco IdS, and Cloud Connect.
- Place an order for Digital service for your customer using Cisco Commerce Workspace (CCW). For more information, see the Cisco Collaboration Flex Contact Center Ordering Guide available at https://www.cisco.com/c/en/us/partners/tools/collaboration-ordering-guides.html.
- If you want your customers to reach you through SMS and if you are provisioning a US phone number for SMS communication, you must first procure a 10 Digit Long Code (10DLC), which is the mandated standard for Application-to-Person (A2P) text messaging. Work with your Account Manager to verify your business and set up your SMS Business account number. After you obtain the number, use it to configure the SMS contact handling. For instructions, see SMS.
- Make sure that you enable the Single Sign-On (SSO) mode for agents who are required to handle digital channel tasks. To enable agents for SSO one at time, use the configuration tools. To enable multiple agents at once for SSO, use the SSO migration tool. For instructions, see the *Migrate Agents and Supervisors to Single Sign-On Accounts* section in the Cisco Packaged Contact Center Enterprise Features Guide.
- Tokens created with Cisco IdS are used to authenticate CCE agents with Webex Engage. This necessitates synchronization of the time fields in the tokens between Cisco IdS and Webex Engage cloud services. As a result, the Cisco IdS NTP server configuration must be synchronized, either directly or indirectly, with a public NTP server that supports leap smearing in order for the premise and cloud NTP configurations to be in sync. Furthermore, when used with digital channels, the lifetime of a Cisco IdS token cannot be set to less than 2 hours.

Redaction of sensitive data

This feature redacts or masks all the sensitive data that you send over Webex Connect to ensure PCI compliance. The PCI service scans all your requests for sensitive data, masks the data, and sends it over to the Agent Desktop. The PCI service also scans the agent responses and sends the redacted data, including attachments that do not contain sensitive data, to the end customers.

The following are the supported file types on which the PCI service applies redaction:

Table 6: Supported File Types

Category	File Extension
Document	html, mhtml, mht, odt, pdf, pdfxml, rtf, shtml, xps, xml, xhtml, txt
E-mail	eml, msg
Microsoft	ods, doc, dochtml, docm, docx, docxml, dot, dothtml, dotx, dotm, pot, pothtml, ppthtml, pptmhtml, pptxml, potm, potx, pps, ppam, ppsm, ppsx, pptx, pptm, ppt, pub, pubhtml, pubmhtml, xls, xlshtml, xlthtml, xlt, xlsm, xltx, xltm, xlam, xlsb

Note Any file types that are not listed in the Supported File Types table are automatically dropped in the transmission. If there are any images in the attachments that are sent in the supported file types, the attachments are automatically dropped. For example, if your PDF file includes images, the file is automatically dropped in the transmission, even though PDF is a supported file type.

Workflow for enabling and managing digital channel interactions

As a partner, complete the following tasks to onboard your customer for Hybrid Services and provision the digital channel capabilities:

Note In some rare cases, the customer administrators can also perform the following tasks to provision the digital channel capabilities for their organization.

Table 7: Initial configurations to enable digital channel capabilities

Step #	Task	Reference
Onboard hybrid services for customers using Control Hub **Note** If your customer is already using a Hybrid Service such as CCAI, skip steps 1 and 2 in this section and directly start with provisioning the digital channel capabilities (step 3).		
1	Place an order for Digital service for your customer using Cisco Commerce Workspace (CCW), and your Contact Center subscription starts. Set up hybrid services for your customer.	See the Set up hybrid services for your organization.
2	(Optional) Create a user with Full Admin role to authorize task requests that are sent from Webex Connect to Contact Center Enterprise (CCE) through Cloud Connect.	• Add users manually in Control Hub • Assign organization account roles in Control Hub
3	Provision digital channels for your customer on Control Hub.	See the Provision Webex Connect digital services for your organization.
Integrate CCE and Webex Connect		
Part 1—Cloud Connect Configurations		

Step #	Task	Reference
4	Install and add the Cloud Connect publisher and subscriber nodes to the inventory for your customer.	See the *Install Publishers/Primary Nodes of VOS-Based Contact Center Applications* and *Install Subscribers/Secondary Nodes of VOS-Based Contact Center Applications* sections in the Cisco Packaged Contact Center Enterprise Installation and Upgrade Guide.
5	Register Cloud Connect in the Unified CCE Administration portal.	See the *Cloud Connect Integration* section in the Cisco Packaged Contact Center Enterprise Administration and Configuration Guide.
6	Configure the Media Routing Peripheral Interface Manager (MR PIM) for the Digital Routing service.	See the *Add PIMs to the Media Routing Peripheral Gateway* section in the Cisco Packaged Contact Center Enterprise Administration and Configuration Guide
7	Establish secure connections between Cloud Connect and Media Routing Peripheral Gateway (MR PG), and set up the Nginx reverse proxy server certificate for the digital channels interaction.	See the *Certificate Management for Digital Channels Integration* section in the Security Guide for Cisco Unified ICM/Contact Center Enterprise.
Part 2—Reverse Proxy Configurations		
8	Set up and configure reverse proxy server for the digital channel interaction.	See Workflow to configure reverse proxy using automated installer, on page 71.
Part 3—CCE Configurations		
9	Create a unique media routing domain (MRD) for each media channel and map it to your media channel if you want granular channel-specific reporting and agent state control. Use the Media Routing Domain List tool to create the MRDs. For Emails and asynchronous social chat channels like SMS, consider a longer **Start Timeout** and **Max Duration** for the Media Routing Domain. You can also map existing MRDs to the Digital Routing media channels. This will avoid having to create new skill groups, precision queues, and scripting logic changes to target the new media channels.	• For instructions about how to create a new MRD, see the online help that is integrated with the Media Routing Domain List tool. • For instructions about how to map the existing MRDs to the Digital Routing media channels, see Set up media channels.

Step #	Task	Reference
10	Associate the MRDs that you have created for the digital channel integration using Webex Connect to the system-defined application paths. For every agent peripheral gateway, there is a system-defined application path that gets created with a suffix "UQ.Desktop". There is also an associated system-defined application called UQ.Desktop that automatically gets created in the sytem and identifies the Cisco Finesse server as a client to the Agent PG, to control Agent states in MRDs created for digital channels. The first number in the application path identifies the Logical Controller ID of the PG. An example of the system-defined application path is `5000.UQ.Desktop`. **Note** If you create the MRDs through the Unified CCE Administration portal, the MRDs get automatically associated to the system-defined application paths. If you have created the MRDs using the Media Routing Domain List tool under Configuration Manager, you must explicitly associate the MRD with the application path.	See the online help that is integrated with the Application Path List tool
11	Create mandatory ECC variables in the Unified CCE Administration portal (**Overview** > **Call Settings** > **Route Settings** > **Expanded Call Variables**). The ECC variables are required to identify the incoming tasks as Digital Routing tasks and the tasks to carry the Webex Engage conversation ID.	• For instructions about how to create ECC variables, see the *Add and Maintain Expanded Call Variables* section in the online help that is integrated with the tool. • For the list of mandatory ECC variables, see the ECC Variables for Digital Routing Tasks, on page 74.
12	Configure digital channel capabilities for your customer in the Unified CCE Administration portal.	See Manage digital channels.
13	Create routing scripts for digital channel interaction.	See the *Example Scripts for Digital Channel Interactions Using Webex Connect* section in the Scripting and Media Routing Guide for Cisco Unified ICM/Contact Center Enterprise.

Step #	Task	Reference
14	(Optional) You can configure an Agent Request or Web Callback feature to allow your customers to place a web callback request to the contact center.	See Agent Request or Web Callback using Webex Connect, on page 90.
Part 4—Cisco Finesse Configurations **Note** Ensure that you complete all the prerequisites listed in the *Prerequisites to configure the Manage Digital Channels gadget* section in the Cisco Finesse Administration Guide before you proceed with Finesse configurations to set up the Manage Digital Channels gadget.		
15	Provision Cloud Connect on Cisco Finesse.	See the *Cloud Connect Server Settings* section in the Cisco Finesse Administration Guide.
16	Add the Manage Digital Channels gadget to the Cisco Finesse desktop layout.	See the *Add Manage Digital Channels Gadget* section in the Cisco Finesse Administration Guide.
Part 5—Webex Connect Configurations		
17	Configure node authorizations in Webex Connect to inject new tasks, retrieve a task's details, and close tasks directly on CCE.	See CCE Integration Nodes and Node Authorizations.
18	Configure channel assets for the required media channels such as SMS, Live Chat, and email.	See the following sections: • SMS • Live Chat • Email
19	Create flows using the Webex Connect Flow Builder feature for the various digital channels interactions such as SMS, Email, and Live Chat. Use one of the following options to create flows: • Import the Webex Connect CCE Flow Templates - 12.6(2) into the Webex Connect portal and tailor them to your specific needs. • Use a pre-built flow template that is available in the Webex Connect portal and customize it to suit your requirements.	See Flow Configurations.
Part 6—Webex Engage Configurations		

Step #	Task	Reference
20	Set up the customer chat widget.	See Administration and Setup Guide for Webex Engage with Cisco Contact Center Enterprise.
21	Verify that the agents who are enabled for digital channel interaction are synchronized to Webex Engage.	
Work with Manage Digital Channels gadget		
22	After you complete your initial configurations to enable digital channels interactions, the agents and supervisors can start interacting with customers using the Manage Digital Channels gadget.	See the Cisco Contact Center Enterprise Manage Digital Channels Gadget User Guide.

Generate public key certificate using Cisco IdS

The Manage Digital Channel gadget authenticates with Webex Engage in the Single Sign-On (SSO) mode, through tokens generated using public key cryptography. Use a secret private key to sign the tokens after which you can verify the tokens using a freely distributed public key certificate. Cisco Identity Service (IdS) generates the public and private keys that you can use to sign and verify the token. Cisco IdS exposes the CLI or REST interfaces to fetch the public key certificate for verifying the token. A public certificate authority (CA) must sign this public key certificate. You must then upload the CA signed public key certificate in Control Hub to authenticate and enable communication between Webex Engage and the Manage Digital Channel gadget.

Note You cannot use a self-signed certificate to provision digital channels in Webex Control Hub.

To generate the public key certificate:

Step 1 Use SSH to log in to the Cisco IdS server's command line interface using the administrator credentials.

Step 2 Run the following CLI command to generate the Certificate Signing Request (CSR) that can be used to obtain a CA signed certificate:

show ids token csr—Displays the CSR corresponding to the public key that is used to validate the tokens.

Step 3 Copy the entire CSR including the header and footer (-----BEGIN NEW CERTIFICATE REQUEST----- and -----END NEW CERTIFICATE REQUEST-----) from the Cisco IdS console and upload the same to the CA provided interface for generating the signed certificate.

Step 4 Save the contents of the CA provided certificate generated using the CSR in step 3, into a file with extension .pem or .der.

You must upload the certificate in Control Hub when you provision the Digital Channels for an Organiztion. For instructions, see Provision Webex Connect digital services for your organization.

Note You need not upload the CA-signed certificate in the Cisco IdS server.

Regenerate public key certificate using Cisco IdS

To regenerate a public-private key pair:

Before you begin

You need to regenerate the public key certificate when the certificate is compromised, or when the CA-signed certificate or the Cisco IdS certificate expire. Regenerating the certificates for token signing requires all agents to logout and relogin to the Cisco Finesse desktop. Therefore, ensure that you plan for downtime to your Contact Center before you regenerate the public-private key pair that Cisco IdS uses to authenticate agents.

Step 1 Sign in to **Cisco Identity Service Management** using the following URL: https://*<hostname of Cisco IdS server>:8553*/cmplatform.

Step 2 From the left navigation pane, choose **Settings** > **Security** > **Keys and Certificates**.

Step 3 In the **Encryption/Signature Key** area, click **Regenerate**.

After you regenerate the new key pair, you must reboot Cisco IdS so that the agents can relogin to their applications. Ensure that a certificate authority signs the regenerated public-private key pair. You must reupload the CA signed certificate to the Control Hub for your agents to resume their tasks in the Manage Digital Channel gadget. For instructions, see Provision Webex Connect digital services for your organization.

Reverse proxy configuration for digital channel interaction

A reverse proxy server is an intermediary server that you must deploy in the Demilitarized Zone (DMZ) in your network. The reverse proxy server forwards task requests and user configuration notifications from Webex Connect and Webex Engage to the Digital Routing and DataConn services respectively, that are running on the Cloud Connect platform. The task requests such as transfer and close that originate from Cisco Finesse are directly invoked on the Digital Routing service.The reverse proxy configuration enables automatic failover to the stand-by Cloud Connect in case of failures.

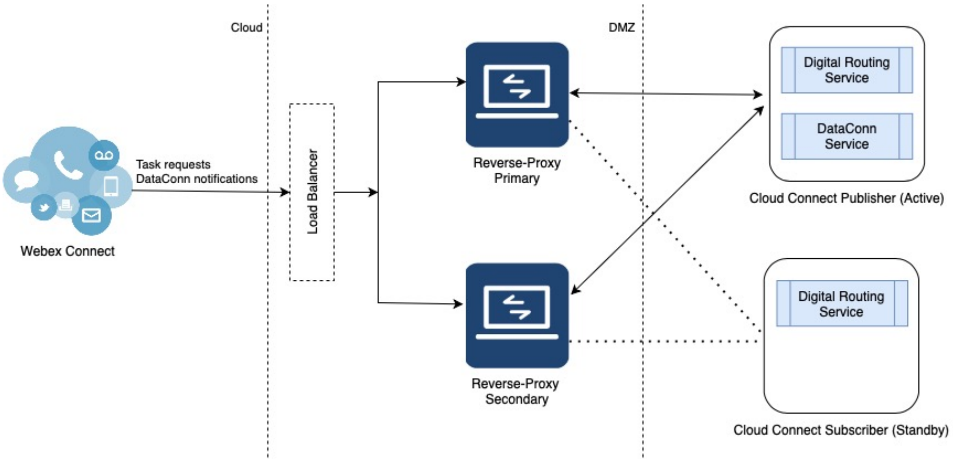
Reverse proxy deployment model

CCE supports the following two deployment models for reverse proxy:

- A pair of reverse proxy servers with physical load balancer.
- A pair of reverse proxy servers with DNS-based load balancer.

Note When using this deployment model with VPN-less Finesse, be sure to set up separate FQDNs for the reverse proxy servers so that Finesse desktop can access the specific reverse proxy IP addresses.

The following diagram depicts the task flow and the reverse proxy deployment model.



The following describes tasks requests and Dataconn flow:

- The Cloud Connect receives task requests from Webex Connect through either a single reverse proxy or a pair of reverse proxy servers deployed in the DMZ network, that is front-ended with a Load Balancer.
- The tasks are sent through reverse proxy only to the active side of Cloud Connect in both the deployment scenarios. That is, when a pair of reverse proxy servers are deployed, both the primary and the secondary reverse proxy servers are connected only to the active side of Cloud Connect.
- The active side of the Digital Routing service can either be on the publisher node or on the subscriber node of Cloud Connect. The following is the behaviour of the Digital Routing service and the DataConn service during failover:
  - The Digital Routing service supports automatic failover and all the task requests are routed through the Digital Routing service on the new active node.
  - The DataConn service runs only on the publisher node and does not failover to the subscriber node. As a result, when there are new users created in CCE, the user configurations are not synchronized in Webex Engage until the DataConn service on the publisher node is up and running.

Note The DataConn service is active only on the Cloud Connect publisher node.

Workflow to configure reverse proxy using automated installer

Complete the following tasks to configure reverse proxy using the automated installer for digital channel interaction:

Note For customers who've opted to set Country of Operation as Canada in Control Hub, you must install the reverse proxy shipped as part of 12.6(2) ES1. This is required to ensure that the traffic originating from the tenants hosted in that data center pass through to Cloud Connect. For a list of mapping between Country of Operation and Data Center, refer to https://help.webex.com/en-us/article/n0p6xa1/Data-Locality-in-Webex-Contact-Center.

Step 1 Install and start the reverse proxy. For instructions, see Reverse Proxy Automated Installer, on page 325.

Note

- Authentication is not supported at the edge of reverse proxy for all the requests and protocols that are related to digital channel interactions.
- Configuration of mapping file is not required for digital channel interactions.
- Configuration of Finesse is mandatory even if you want to install only Cloud Connect in your setup.

Step 2 Configure the digital channel client hosts.

a) Add the list of trusted digital channel client IP addresses and the corresponding hostnames to the reverse proxy Cloud Connect environment configuration file (cloudconnect.env). Use the variable, **NGX_CLOUDCONNECT_CLIENT_IPS** to add the IP addresses of Webex Connect and Webex Engage to the cloudconnect.env file.

Note

- The reverse proxy considers any requests as valid only if it receives requests from the configured hostnames or IP addresses. For more information about the environment configuration files, see Configure deployment environment configurations, on page 337.
- All the currently available IP addresses of Webex Connect and Webex Engage are already added in the 'cloudconnect.env' file. Depending on your deployed datacenter, remove any IP addresses that you do not need.

b) Add load balancer or proxy IP addresses. For more information, see Load balancer, WAF, and proxy support for reverse-proxy deployments, on page 271.

Step 3 Configure the Mutual Transport Layer Security (mTLS) authentication between reverse proxy and Cloud Connect.

a) Add the list of trusted reverse proxy IP addresses and the corresponding hostnames on the publisher and subscriber nodes of Cloud Connect. For details, see Add Proxy IP, on page 72.

b) Configure SSL certificate verification to establish communication between the reverse proxy host and the Digital Routing service. For details, see Configure reverse proxy host verification, on page 74.

Add Proxy IP

You must add the list of trusted reverse proxy IP addresses and the corresponding hostnames on the publisher and subscriber nodes of Cloud Connect. The Cloud Connect nodes consider any requests as valid only if they receive the requests from the configured hostnames or IP addresses. Ensure that the allowed hosts contain only the internal and external FQDN and IP address of the reverse proxy.

Note Do not add the hostname or IP address of the load balancer.

The following is an example of the CLI to add the hosts and IP addresses:

```
admin:utils system reverse-proxy allowed-hosts add 10.78.95.178
Source 10.78.95.178 successfully added
admin:utils system reverse-proxy allowed-hosts add proxy.xyz.com
Source proxy.xyz.com successfully added

Restart Cisco Web Proxy Service for the changes to take effect: utils service restart Cisco
 Web Proxy Service
```

If the added hostname is not resolvable from a component, the following error is displayed:

```
admin:utils system reverse-proxy allowed-hosts add group.facebook

Either IPv4 address or hostname is invalid or is not resolvable. Now validating IPv6 address
 for source group.facebook

Operation failed, please enter valid source(s). Source group.facebook is invalid
```

After adding proxy hosts as trusted hosts through CLI on individual nodes, you must upload proxy server certificates to the Tomcat trust store of the respective components. This is required for proxy authentication to work. Otherwise, the traffic from proxy will be rejected by the components. For information about generating proxy certificates and uploading to the Tomcat trust store, see the *Set up Nginx reverse proxy certificate* and *Generate and Copy CA Certificates of VOS Components* sections in the Security Guide for Cisco Unified ICM/Contact Center Enterprise.

The following is an example of the CLI to view the list of allowed hosts and IP addresses:

```
admin:utils system reverse-proxy allowed-hosts list

Source proxy.xyz.com successfully added list

The following source(s) are configured:

1. 10.78.95.178
2. proxy.xyz.com
3. proxy125.xyz.com
```

The following is an example of the CLI to delete an entry from the list of allowed hosts and IP addresses. This command lists all the configured proxy hosts and IP addresses, and gets user input to delete specific or all proxy hosts and IP addresses.

```
admin:utils system reverse-proxy allowed-hosts delete
Select the reverse-proxy source IP to delete:

 1) 10.78.95.178
 2) proxy.xyz.com
 3) proxy125.xyz.com
 4) all
 5) quit

Please select an option (1 - 5 or "q" ): 1

Delete operation successful
```

Configure reverse proxy host verification

Reverse proxies are deployed in DMZ and are therefore security sensitive. So, the communication between the proxy and the upstream server that it is proxying, is recommended to be secured to a higher degree. This is achieved by having both the TLS certificates and the copy that is uploaded to the relevant servers be mutually verified. Run the following CLI commands on the publisher and subscriber nodes of Cloud Connect to configure SSL certificate verification for establishing communication between the reverse proxy host and the digital routing service:

utils system reverse-proxy client-auth

This command has the following parameters:

- enable
- disable
- status

By default, the host authentication is enabled.

The following is an example of the CLI to view the status of the host authentication:

```
admin:utils system reverse-proxy client-auth status

SSL certificate verification for connections established from reverse proxy hosts is disabled
```

The following is an example of the CLI to enable the host authentication:

```
admin:utils system reverse-proxy client-auth enable
SSL certificate verification enabled for connections established from reverse proxy hosts

Restart Cisco Web Proxy Service for the changes to take effect:
utils service restart Cisco Web Proxy Service
```

The following is an example of the CLI to disable the host authentication:

```
admin:utils system reverse-proxy client-auth disable
SSL certificate verification disabled for connections established from reverse proxy hosts

Restart Cisco Web Proxy Service for the changes to take effect:
utils service restart Cisco Web Proxy Service
```

Custom reverse proxy configuration

If you choose to not use the reverse proxy installer that Cisco has provided and want to deploy a custom reverse proxy, see Guidelines for Custom Reverse Proxy Deployment, on page 397.

ECC Variables for Digital Routing Tasks

This section provides the list of ECC Variables that you must configure for the Digital Channel interaction. All ECC variables are of type scalar and they are case-sensitive.

Table 8: ECC Variables

Variable	Table Size	Length	Mandatory / Optional
`user_DR_Primary`	Used to hold the FQDN of either the publisher node or the subscriber node of Cloud Connect. The FQDN denotes the Digital Routing service node that injected the task into CCE.	[1]1 character + the length of the Cloud Connect FQDN.	Mandatory
`user_DR_Backup`	Used to hold the FQDN of either the publisher node or the subscriber node of Cloud Connect. The FQDN denotes the peer node of the Digital Routing service, which can be used as a backup, in case the Primary node is unavailable.	1 character + the length of the Cloud Connect FQDN.	Mandatory
`user_DR_MediaResourceID`	Used to hold the Conversation ID of Webex Engage, which the Webex Connect injects. The Cisco Finesse uses this conversation ID to load the media in the Manage Digital Channels gadget.	16 characters	Mandatory
`user_DR_CustomerName`	Used to display the customer name in the pop-over notification that an agent views when a task is received in the Manage Digital Channels gadget.	Up to a maximum of 210 characters.	Optional

Variable	Table Size	Length	Mandatory / Optional
`user_DR_MediaChannelName`	Used to populate the media channel name for the task, if you configure and add this ECC variable in the **Unified Contact Center Enterprise Management** portal. See the *Define ECC variables* section in the Cisco Unified Contact Center Enterprise Features Guide. This is typically required if you have mapped more than one Digital Routing media channel to the same MRD but still want unique task icons to appear on the agent desktop. For example, in CCE, the system-defined media channels, chat and SMS are both mapped to the same MRD. If there is a 1:1 mapping between the media channel and MRD, the system automatically identifies the icon to be displayed on the desktop based on the media channel with which the task was associated.	8 character	Optional

[1] If there are lengthy FQDNs, the Digital Routing service compresses the FQDN such that it is below 100 characters, and thereby allow ECC variable some space out of the 2000 bytes limit per call / task to be saved by not having to send the long name. In such cases, the variable can be defined as 100 characters long in CCE. The Digital Routing service automatically fills in the first character to indicate whether the following data is compressed or uncompressed. A prefix of 1 denotes compressed data, while a prefix of 0 denotes uncompressed data.

Manage digital channels

Note To access this feature, ensure that you add Cloud Connect to the inventory in the Unified CCE Administration portal and register it.

As an administrator, perform the following configurations in the Unified CCE Administration portal to manage digital channel capabilities for your customers to reach business:

- Set up media channels such as chat, email, social, voice callback, facebook, and whatsapp. See Set up media channels, on page 77.
- Synchronize agents in AW database to Webex Engage database. See Configure User Sync, on page 80.
- Define Expanded Call Context (ECC) variables to assist agents with relevant information. See Define ECC variables.
- Integrate Cloud Connect with Webex Connect using the Open Authorization v2.0 (OAuth v2.0) standard. See Integrate Cloud Connect with Webex Connect.
- Customize the connection parameters between Cloud Connect and MR PG. See Manage connection between Cloud Connect and MR PG, on page 84.

Set up media channels

Contact Center Enterprise (CCE) supports media channels, such as Live Chat, Email, and SMS, with enhanced capabilities. These media channels are mapped to media type and Media Routing Domains (MRDs) for providing granular channel-specific reporting and agent state control. Based on this configuration, Cloud Connect associates the received task request to the appropriate MRD.

MRDs are used to map agents' skill group with the media channels, based on which the agents are assigned a task. You can map multiple media channels to the same MRD. For example, you can map SMS and Chat channels to the same MRD. For more information on MRDs, see the *Media Routing Domains* section in the Configuration Guide for Cisco Unified ICM Enterprise.

Media type is a broad category of media channels and it is different from the media class or MRD in CCE. The Digital Routing service has internal sub-limits and different maximum queue time parameters that it applies for tasks of a certain Media Type as defined by the Queue Settings. The media types are **Chat**, **Email**, **Telephony** (Voice Callback), **Social**, **Facebook**, and **WhatsApp**.

Webex Connect offers support for the pre-provisioned system defined media channels. You can add a custom media channel if one of the many media channels that Webex Connect supports in the future has to be allowed in the CCE system. If not, you can skip adding a Media Channel.

Note System-defined media channels cannot be deleted.

Queue Settings for media channels is used to configure the maximum queue limit and maximum queue time for each media type, individually. For more information on configuring queue settings for media channels, see Configure queue settings.

To set up media channels:

Step 1 In the Unified CCE Administration portal, navigate to **Overview** > **Digital Channels** > **Digital Channel Settings** > **Media Channel**. The list of existing media channels is displayed in a grid. This is the default page.

Step 2 Click **New**. On the **New Media Channel** page, do the following:

a) In the **Name** field, type the name of the media channel.

b) From the **Media Type** drop-down list, select the required media type.

c) In the **Media Routing Domain** field, search and select the MRD that you want to map to the media channel.

Step 3 Click **Save**. The channel is added to the list on the **Digital Channel Settings** page.

When you have created a media channel, you can modify only the **Media Routing Domain** field. If you want to modify any other fields, you must delete the media channel and create afresh. To delete a media channel, hover over the media channel and click the **X** icon.

Configure queue settings

The Digital Routing service maintains a queue for all the incoming tasks. The service accommodates up to a maximum of 100,000 tasks in the queue at any given point in time. Out of these 100,000 tasks, you can configure the maximum number of tasks that can be queued for each media type in the Digital Routing service. The following are the default values in percentage for each media channel:

Media type	Queue setting in %	Description
Email	50%	50,000 tasks can be queued for Email.
Social	40%	40,000 tasks can be queued for SMS.
Voice callback	5%	5,000 tasks can be queued for Voice callbacks.
Chat	5%	5,000 tasks can be queued for live chat.

You can also define the maximum duration for which a task remains in the queue for each media type.

The following are the scenarios in which the Digital Routing service rejects the tasks and sends them back to Webex Connect:

- The number of tasks exceeds the maximum value that is configured for the media channel. For example, consider that the Digital Routing service queue has 50,000 email tasks, which is the maximum queue setting defined for the Email media channel. If Webex Connect further injects email tasks, the Digital Routing service rejects the incoming tasks and sends them back to Webex Connect with an error code of 20286 (MEDIA_TYPE_QUEUE_LIMIT_EXCEEDED). For example, a flow developer can decide to provide appropriate messaging to the end customer and invoke other backend systems to denote queue capacity issues.
- A task exceeds the maximum duration that is defined for the media channel. The Create Task node is successful. There is a Closed webhook event triggered when the task gets auto-closed after exceeding the maximum time in queue, with a specific disposition code set to CD_MAX_DIALOG_LIFETIME_EXCEEDED.

The flow debug logs show the error code returned by the Create Task node API call. For more information about the Create Task node, see Create Task.

To configure the queue settings for the media channels:

Step 1 In the Unified CCE Administration portal, navigate to **Overview > Digital Channels> Digital Channel Settings > Media Channel**.

Step 2 Click **Queue Settings**.

Step 3 Enter the required values in percentage for each media channel.

Note Make sure that the sum of the percentage values configured for all the media channels is 100.

Step 4 Define the duration in days, hours, and minutes for each media channel. This is the maximum duration for which a task that belongs to the specific media channel remains in the queue after which it gets timed out. The default durations are 3 days for Email and Social media channels, and 2 hours for Voice callbacks and Chat media channels.

Step 5 Click **Save**.

Synchronize CCE agents to Webex Engage

For CCE agents to be able to take up the digital channel tasks that are initiated from Webex Connect, the agent configuration details must be synchronized between the AW server and Webex Engage. The DataConn service that is running on the Cloud Connect platform is responsible for synchronizing agent configurations. The default periodic interval for synchronization is 30 minutes. Only the agent records created or updated during the synchronization period will be synchronized with Webex Engage.

The Finesse Desktop SSO login depends on the agent information that is synchronized between AW server and Webex Engage. The agent identity that is used to authenticate the login request, is possible only after this synchronization is complete.

Important considerations for agent synchronization

Consider the following when you synchronize agents between CCE and Webex Engage for the digital channel integration:

- After synchronizing the agents details between CCE and Webex Engage, if you delete an agent in CCE, the agent status in Webex Engage changes to Inactive. However, the agent record is not deleted in Webex Engage.
- The synchronization of agents details is possible only on the publisher node of Cloud Connect. Also, any update to the agent configuration takes effect only when the publisher node is up and running.
- Ensure that you create agents and update the agent configurations in CCE only. Do not create or update agent configurations in Webex Engage.
- Ensure that you always have the agent details synchronized between CCE and Webex Engage for the agents to be able to login to the Manage Digital Channels gadget.
- When you create a new agent in CCE, you must not associate the agent with an existing person record that is already associated with another agent. If you do not select any person record in the **Select Person** drop-down list in the **ICM Configuration Manager** > **Tools** > **Explorer Tools** > **Agent Explorer** tool, a new person record is automatically created for the agent. Also, you must not associate one person record with multiple agents.
- You must not modify the email address of the agent after the agent record and person record are created. If you need to update the email address, contact Cisco Support.

Enable agents for digital channels

To enable agents for digital channel interaction:

Step 1 In Unified Contact Center Enterprise Management, navigate to**Users** > **Agents**.

Step 2 Select the agent for whom you want to enable digital channels, and then go to the **Enable Digital Channels** tab.

Step 3 Check the **Support Digital Channel** check box to enable the agent for digital channel interaction

Step 4 Click **Save**.

Configure SQL user account for digital channels

To configure SQL user account for digital channels:

Step 1 Launch Microsoft SQL Server Management Studio using System Administrator login credentials on the Administration and Data Server.

Step 2 Navigate to **Security** > **Logins**, right-click **Logins** and select **New Logins**.

Step 3 On the General screen:

a) Enter the Login Name.
b) Select **SQL Server authentication**.
c) Enter and confirm the password.
d) Uncheck **Enforce password policy**.

Step 4 In the **Server Roles** page, check the **Public** check box.

Step 5 On the **User Mapping** page, do the following:

a) Check the **Real-time database** check box.
b) In the **Database role membership for** area, check the **db_datareader** check box.

Step 6 Click **OK**.

What to do next

Ensure that you configure SQL User Account on both the primary and secondary AW databases.

Configure User Sync

To synchronize the CCE agents who are configured for digital channel interactions with Webex Connect:

Step 1 In the Unified CCE Administration portal, navigate to **Overview > Digital Channels> Digital Channel Settings > User Sync**.

Note The DataConn service is active only on the publisher node of Cloud Connect. Ensure that the publisher node is accessible for you to view the **User Sync** page.

Step 2 In the **Network Entry Point** field, enter the hostname or FQDN of the load balancer or the reverse proxy server based on your deployment. It is through this network entry point that the Webex Engage sends the response request to the DataConn service for agent synchronization.

For more information about network entry point, see the *Synchronize CCE agents to Webex Engage* section in the Solution Design Guide for Cisco Packaged Contact Center Enterprise.

Step 3 In the **AW Database Details** area, complete the following fields to configure the DataConn service for the Primary and the Secondary Server:

a) In the **AW Datasource Host** field, enter the IP address or the host name of the AW server that has the agent configurations.
b) In the **Port** field, enter the SQL port number of the AW database server.
c) In the **Database Name field**, enter the name of the AW database server.
d) In the **Database User ID** and **Password** fields, enter the user ID and password of the SQL user account that you created for digital channels. For more information see the *Configure SQL user account for digital channels* section in the Cisco Packaged Contact Center Enterprise Features Guide.

Step 4 Click **Test Connection** to make sure that the DataConn service can read the data from the AW Primary server.

Step 5 Switch on the **Enable Failover** toggle button to enable the failover to the Secondary AW server.

Step 6 To synchronize agents, choose one of the following options:

a. Enable the **Enable Sync** toggle button to automatically synchronize agents in an interval of 30 minutes.

b. Click **Sync Now** to manually synchronize the agents. This option is available only when the **Current Sync Status** field appears as **Scheduled**.

Note If you recreate your AW Database to fix any errors, ensure that you disable User Sync before starting with the recreation process. Enable the User Sync only after the AW Database is created and synchronized with the central controller database.

Step 7 In the **Last sync** field, view the date, time and status of the previous synchronization.

Step 8 In the **Agents Sync Details** field, view the number of agents that are synchronized successfully. The number of agents appears as a clickable link. Click the link to view the list of agent records that have failed to synchronize and the list of agent records that are pending for synchronization. You can choose to refresh the agent records at any given point in time.

Step 9 In the **Current Sync Status** field, you can view one of the following statuses:

a. **No sync is in progress or scheduled**—This status appears when the **Enable Sync** toggle button is off.

b. **Scheduled**—This status appears when the **Enable Sync** toggle button is on.

c. **In Progress**—This status appears when the scheduled sync or manual sync is in progress.

d. **Manual Sync Failed** —This status appears when the manual sync has failed.

e. **Unknown**—When sync status is not available.

Step 10 Click **Save**.

Configure network entry point for agent synchronization using CLI

You must configure the network entry point for the DataConn service that is running on the Cloud Connect platform. It is through this network entry point that the Webex Engage sends the return request in response to the user configuration API request that the DataConn service sent to Webex Engage. For more information, see the *Synchronize CCE agents to Webex Engage* section in the Solution Design Guide for Cisco Packaged Contact Center Enterprise .

To configure the host for the network entry point, run the following command on the Cloud Connect console:

set cloudconnect dataconn settings

The following is the example of the CLI configuration:

```
admin:set cloudconnect dataconn settings
Fetching existing configuration...
Enter the Config details to be saved:
Network Entry Point Host: proxyhost.domain.com
The config details updated successfully.
```

To display the host that is configured for the network entry point, run the following command on the Cloud Connect console:

show cloudconnect dataconn settings

Field mapping between Webex Engage and CCE

Following table lists the payloads and attributes mapping between Webex Engage and the Contact Center Enterprise (CCE):

Webex Engage fields	CCE fields	Remarks
firstName	Person.FirstName	User's first name will contain only alphanumerics. All special characters from the given name will be removed. If this field is empty, the first name will be the username, which is obtained from the user's email without domain name.
lastName	Person.LastName	User's last name will contain only alphanumerics. All special characters from the family name will be removed. If this field is empty, the last name will be the first character of the username, which is obtained from the user's email without domain name.
aliasId	Person.LoginName	Agent's login name. Single Sign-On (SSO) authentication matches the Person.LoginName from AW database against the UPN or SamAccountName data from Cisco IdP and is used as the user_id in the SSO token. The Person.LoginName information is uploaded as aliasID in Webex Engage database for the purposes of token verification.
emailId	Person.Email	
loginId	Agent.SkillTargetID	Webex Engage uses this mapping to route the agent to the correct interaction.

Webex Engage fields	CCE fields	Remarks
concurrency	Not mapped	The Concurrency value is optional and is set to “99” by default on Webex Engage.
roleType	Fixed to customer_care	This is the fixed user role that is assigned to an agent.
status	Agent.Deleted	If a deleted flag is set in the CCE, the value is inactive.
loginUsingEmail	false	By default, every user on Webex Engage is created with the "loginUsingEmail" value as false.
role	team_agent	This is the fixed role that is assigned to team_agent.

Define ECC variables

The Expanded Call Context (ECC) variables are data that are embedded within the call and are visible to the agent on the Agent Desktop. ECC variables are passed back and forth in ECC payloads. ECC variables assist the agent with relevant information without the customer having to repeat the same information.

To define the ECC variables:

Step 1 In the Unified CCE Administration portal, navigate to **Overview** > **Digital Channels** > **Digital Channel Settings** > **ECC Variable**.

Step 2 On the **ECC Variable** page, click the plus icon (+). The **Add ECC Variable** page appears with existing variables **Name** and **Enabled** details. It excludes the built-in variables.

Step 3 Select the required variable. The **ECC Variable** page displays the newly added ECC variable.

Step 4 Click **Save**.

Integrate Cloud Connect with Webex Connect

You can integrate Cloud Connect with Webex Connect using the Open Authorization v2.0 (OAuth v2.0) standard. You must configure Webex Connect client ID and client secret in the Digital Routing service for the service to gain access to Webex Connect using the access token. This set of client credentials uniquely identifies the Cloud Connect and its permissions to access Webex Connect.

Note The Cloud Connect Management service is active only on the publisher node of Cloud Connect.

To configure client credentials for OAuth v2.0 access token:

Step 1 In the Unified CCE Administration portal, navigate to **Overview** > **Digital Channels** > **Digital Channel Settings** > **Integration**.

Step 2 On the **OAuth2 Authentication Details** page, perform the following steps.

a) In the **Name** field, enter a name for the Cloud Connect integration. This field is editable.
b) In the **Description** field, enter a description of what the client application does.
c) In the **Client Id** and **Client Secret** fields, enter the client id and secret key that you can retrieve from the Webex Connect portal (navigate to **Assets** > **Integrations** > **CCE pre-built integrations** > **Actions** > **Manage**).
d) In the field, enter the password that is provided to you at the time of registration. This is the secret key that was used for generating the token.
e) In the **Token Request URL** field, provide one of the following URLs based on the location of Webex Connect datacenter:

- Ireland—https://keycloak-authservice.imiconnect.io/auth/realms/imiconnect_uk_prod/token
- London—https://keycloak-authservice.imiconnect.eu/auth/realms/imiconnect_ln_prod/token
- Sydney—https://keycloak-authservice.imiconnect.com.au/auth/realms/imiconnect_syd_prod/token
- The United States of America—https://keycloak-authservice-us.imiconnect.io/auth/realms/imiconnect_us_prod/token
- Canada—https://keycloak-authservice.imiconnect.ca/auth/realms/imiconnect_cn_prod/token

f) From the **Method** drop-down list, select the **POST** method for Webex Connect integration.
g) From the **Content Type** drop-down list, select a media content type. This determines the response format. The available options are **application/json**, **application/xml**, and **application/x-www-form-urlencoded**. For Webex Connect integration, select **application/x-www-form-urlencoded**.
h) In the **Access Token JSON path**, enter the path in the JSON response to fetch the value of the access token. For the Webex Connect integration, enter *access_token*.
i) (Optional) In the **Header List** section, click **+** icon to add a header name and value and click **Add**. The header name and value are used to pass any additional information that Webex Connect may require for Cloud Connect integration.

Step 3 Click **Save** to save the OAuth2 authentication details.

Step 4 Go to the **Webhook** tab to register the Webhook URL in the Digital Routing service. To fetch the Webhook URL:

a) In the Webex Connect portal, navigate to **Assets** > **Integrations**.
b) In the CCE pre-built integrations row, from the **Actions** column, select **Manage**. The **Manage Integration - Prebuilt Integration** page appears.
c) In the **Inbound Events** section, copy the Webhook URL.

Step 5 Paste the URL in the **Webhook URL** field in the Unified CCE Administration portal (**Overview** > **Digital Channels** > **Digital Channel Settings** > **Integration** > **Webhook**).

Manage connection between Cloud Connect and MR PG

To customize the connection parameters between Cloud Connect and Media Routing Peripheral Gateway (MR PG):

Step 1 In the Unified CCE Administration portal, navigate to **Overview** > **Digital Channels** > **Digital Channel Settings** > **Advanced Settings**.

Step 2 The **Port** is a display-only field and the default value is **38001**. It is through this port that Cloud Connect and MR PG communicates with each other.

Step 3 The **Secured** toggle switch is turned on by default. It is to establish a secured connection between Cloud Connect and MR PG. You can turn-off this toggle switch to disable the secure connection.

Step 4 Click **Save**.

Disposition codes for digital channel interaction

The Webhook notification that the Digital Routing service invokes for a closed task contains a disposition code that denotes the reason for the task closure. Using the disposition code in the Close Task, a flow designer can perform different actions as needed in the Webex Connect flow. For example, if a task fails after it has been queued, an error message can be displayed to customers with the option to contact the business through other support channels or schedule a callback, as opposed to a message indicating that the task has been closed due to a system error.

The following are the list of disposition codes that are available for digital channel interaction:

Table 9: Disposition Codes

Task Disposition	Disposition code value	Description
Normal End	CD_NORMAL_END_TASK	The task ended normally.
Transfer	CD_TASK_TRANSFER	This indicates the disposition when an agent initiates a task transfer using the Finesse desktop. The transfer operation triggers a Webhook notification to Webex Connect which would trigger another NEW task with the same TaskID using the Transferred workflow.

Task Disposition	Disposition code value	Description
Transfer	CD_TASK_TRANSFERRED_ON_AGENT_LOGOUT	This occurs when Finesse server detects an Agent logout, or when the Agent closes the desktop while still having digital channel tasks that aren’t completed or Closed yet. The Finesse server initiates a transfer of the task to the same script selector that routed the original task to the Agent.
Transfer	CD_RING_NO_ANSWER	This indicates that the task timed out while waiting to be accepted by an Agent. Finesse invokes the TaskAction API with the Operation Code "Routed-Transfer" to redirect the task to another agent.
Transfer	CD_TASK_TRANSFER_TIMEOUT	The Digital Routing service sends a TRANSFERRED Webhook notification and waits for a maximum of 15 seconds for Webex Connect to complete its side of the processing, and for a create task request to be resubmitted from Webex Connect with the same TaskID. If Webex Connect fails to reinject the task into the Digital Routing service, then the Digital Routing service marks the task as Closed with this disposition code.

Task Disposition	Disposition code value	Description
Task Lifetime Exceeded	CD_MAX_DIALOG_LIFETIME_EXCEEDED	The task ended because it exceeded the maximum task duration defined for the Media Routing Domain (MRD).
Customer Abandoned	CD_TASK_CUSTOMER_ABANDON	The disposition code is sent from Webex Connect when the task was abandoned by the customer before an agent was assigned to the task.
Customer Abandoned	CD_TASK_ABANDONED_WHILE_OFFERED	The customer cancelled the task before the agent began working on the task. In this task disposition, the agent has viewed the offered task, but the dialog was deleted before the agent accepted the task.
Other	UNKNOWN	The reason for the task termination is unknown.
Other	CD_TASK_INVALID_MEDIA_RESOURCE_ID	The task gets closed by Finesse when the media can’t be loaded in the Webex Engage widget. This depends on a mandatory ECC variable that is used by CCE to relay the Webex Engage ConversationID to Finesse desktop. This is to ensure that the media can be loaded in the widget, once the Agent selects the task.

Task Disposition	Disposition code value	Description
Other	CD_TASK_ENDED_DURING_APP_INIT	This indicates that the task was in progress when the connection between the CTI server and Finesse went down, and the task ended before the connection was reinitialized. When the connection was reinitialized, the Agent PG ended the task.
Failed Task Submission	CD_TASK_FAILED_SENDING_MR_REQUEST	This indicates that the task could not be sent to the MR PG owing to an error. The task is automatically marked as Closed with this disposition code.
Failed Task Submission	CD_FAILED_CREATING_NEW_TASK_EXECUTOR	Internal error in the Digital Routing service while processing a new task request. Such tasks get automatically closed with this disposition code.
Failure	CD_FAILED_INVALID_NEW_TASK_MESSAGE	The task failed as the new task message was invalid.
Failure	CD_FAILED_MEDIA_ROUTING_DISABLED	The task failed as the ICM media routing was disabled.
Failure	CD_FAILED_NO_SCRIPT	The task failed as there was no script to run.
Failure	CD_FAILED_INVALID_MRD_ID	The task failed as a result of invalid Media Routing Domain ID.
Failure	CD_FAILED_ICM_TIMEOUT	The task failed as a result of ICM timeout.

Task Disposition	Disposition code value	Description
Failure	CD_FAILED_INVALID_SCRIPT_SELECTOR	The task failed as a result of invalid dialed number or script selector.
Failure	CD_FAILED_NO_TARGET	The task failed because there was no agent available to be assigned for the task.
Failure	CD_FAILED_ROUTER_RELEASED_TASK	This disposition code is as a result of a Release Node being used in the CCE routing script to terminate the task purposely.
Failure	CD_FAILED_UNKNOWN_ROUTING_PROBLEM	The task failed as a result of unknown routing problem.
Failure	CD_FAILED_DUPLICATE_NEW_TASK_REQUEST	The task failed as it was a duplicate new task request.
Failure	CD_FAILED_UNSUPPORTED_SERVICE_REQUESTED	The task failed as the service requested was unsupported.
Failure	CD_FAILED_AGENT_NOT_MEMBER_OF_QUEUE	The task failed as agent not a member of the specified skill group of the precision queue.
Failure	CD_FAILED_INVALID_QUEUE_TYPE_OR_ID	The task failed as the specified Queue Type or ID. was invalid.
Failure	CD_FAILED_INVALID_DIALOG_ID	The task failed as the dialog ID was invalid.
Failure	CD_FAILED_INVALID_AGENT_ID	The task failed as a result of invalid agent ID.
Failure	CD_FAILED_AGENT_OVER_TASK_LIMIT	The task failed as the agent exceeded the task limit.

Task Disposition	Disposition code value	Description
Failure	CD_FAILED_UNSUPPORTED_AGENT_PERIPHERAL_GATEWAY	The task failed as it was unsupported by the agent peripheral gateway for the service request.
Failure	CD_FAILED_INVALID_AGENT_STATE	The task failed as the agent state was invalid.
Failure	CD_FAILED_INVALID_QUEUE_FOR_MRD	The task failed as a result of invalid queue for Media Routing Domain.
Failure	CD_FAILED_NO_PICK_PULL_NODE	The task failed as no pick pull node available.
Failure	CD_FAILED_AGENT_NOT_READY	The task failed as a result of agent not ready.
Failure	CD_FAILED_UNKNOWN	The session failed for unknown reason.
Failure	CD_FAILED_SESSION_ABANDON	The session failed as it was abandoned.
Failure	CD_FAILED_COMM_FAILURE_ROUTER_AND_PG	The communication failed as a result of router and PG failure.

Agent Request or Web Callback using Webex Connect

The Agent Request or Web Callback feature allows a customer to initiate a request on the web that results in a call from an agent. Use the Webex Connect platform to allow your customers to place a Web Callback request to the contact center. The customer needs to fill out a form with the preferred phone number to receive a callback as soon as an agent is available. Use this feature to switch between media channels when the wait time is more on a channel. For example, if the Live Chat media channel is experiencing an extended wait time, you can offer your customers an option to receive a voice callback from the contact center instead of the customer waiting in the Live Chat channel. The Webex Connect platform provides you the ability to route the web callback requests towards Contact Center Enterprise along with call variables and ECC variables that can carry customer-specific task context.

When CCE receives an agent request or a web callback request, it performs the following tasks:

- Processes the callback request.
- Routes the callback request to an agent and places a call from the agent's phone to the customer.

- Notifies the Webex Connect platform through a Closed Webhook notification that the agent has been selected.

The callback request is automatically closed from the Digital Routing service.

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 Web Callback

To configure a Web Callback request:

Table 10: Web Callback Configurations

Step #	Configure	Where	Configuration details	Reference
1	Call Type	In the Unified CCE Administration portal, navigate to **Overview** > **Call Settings** > **Route Settings** > **Call Type**.	Create a Call Type for the Web Callback request.	See the *Add and Maintain Call Types* section in the Cisco Packaged Contact Center Enterprise Administration and Configuration Guide.

Step #	Configure	Where	Configuration details	Reference
2	Dialed Number	In the Unified CCE Administration portal, navigate to **Overview** > **Call Settings** > **Route Settings** > **Dialed Number**.	**1.** From the **Routing Type** drop-down list, select **Digital Routing**. **2.** From the **Media Routing Domain** drop-down list, select **Cisco_Voice**. **3.** From the **Call Type** drop-down list, select the Call Type that you created in step 1.	See the *Add and Maintain Dialed Numbers* section in the Cisco Packaged Contact Center Enterprise Administration and Configuration Guide.
3	Expanded Call Variable	In the Unified CCE Administration portal, navigate to **Overview** > **Call Settings** > **Route Settings** > **Expanded Call Variables**.	Either use an existing ECC variable or create a new ECC variable for the callback request. **Note** Arrays are not supported with the Web Callback feature. The CCE solutions support the Latin 1 character set only for Expanded Call Context variables and Call variables.	See the *Add and Maintain Expanded Call Variables* section in the Cisco Packaged Contact Center Enterprise Administration and Configuration Guide.

Step #	Configure	Where	Configuration details	Reference
4	Network VRU script	In the Unified CCE Administration portal, navigate to **Overview** > **Call Settings** > **IVR Settings** > **Network VRU Scripts**.	This network VRU script does not refer to a script that you create on a peripheral. This script satisfies a configuration requirement and provides a messaging vehicle to get the value of the estimated wait time to the Webex Connect platform using the MR PIM to fulfill the API call. Use the Manage Network VRU Scripts gadget to create the new network VRU script. Choose a name (for example, VoiceCallback) and enter that name in both Name fields. No configuration parameters are required for the Network VRU Script. Optionally, enter a description. In the remaining fields, leave the default values. You reference this configuration object when you configure the Run External Script node in the routing script.	See the *Add and Maintain Network VRU Scripts* section in the Cisco Packaged Contact Center Enterprise Administration and Configuration Guide.

Step #	Configure	Where	Configuration details	Reference
5	Routing script	**Script Editor**	Create a routing script and schedule it for the Call Type that you created in step 1.	See the following sections: • For instructions, see the *Create Routing Script* and *Schedule Routing Script* sections in the Cisco Packaged Contact Center Enterprise Administration and Configuration Guide. • For a script example, see the *Example Digital Channel Interactions Using Webex Connect* in the Scripting and Media Routing Guide for Cisco Unified ICM/Contact Center Enterprise.
6	A representative flow using CCE Create Task node	**Webex Connect** > **CCE Create Task**	In the **Media Type** and **Media Channel** fields, select **Web Callback**.	For more information, see Create Task.

In the Unified CCE Administration portal, navigate to **Overview** > **Digital Channels** > **Digital Channel Settings** > **Media Channel**. You will see that the "Voice" media channel that has the **Media Type** field set as **Telephony** is associated with the "Cisco_Voice" MRD. For more information, see Set up media channels, on page 77.

In the **Queue Settings** page, ensure that you have the right number of Web Callback requests (which by default is 5000 tasks) that you want to retain in the Digital Routing queue. For more information, see Configure queue settings, on page 78.

Agent Targeting Rule for Web Callback

In addition to the above configurations, ensure that the Agent Targeting rule for the Agent Peripheral also applies to the MR PG routing client, which is required for the Digital Routing service, without which voice calls cannot be routed.

Reporting

Webex Connect reporting

You can view reports related to digital channel information from within the Webex Connect tenant. For more information, see Reports.

View and download the following reports from Webex Connect:

- Count of inbound messages at a channel level
- Count of outbound messages to customers at a channel level
- Flow of tasks counts
- Other standard dashboard reports

Digital Routing reporting

The Cisco Unified Intelligence Center reports include data for voice calls and Digital Routing tasks.

For more information about multichannel reporting data, see the Cisco Packaged Contact Center Enterprise Reporting User Guide.

The Digital Routing media channels are mapped to MRDs and the Unified Intelligence Center reports can be filtered based on these mapped MRDs. Use the following All Fields and Live Data report templates to view the summary reports for each digital channel:

- Agent Real Time
- Agent 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

Digital Routing task-level reporting

The task-level reporting is not available in Cloud Connect. Tasks in the Digital Routing service remain in the service memory only for 15 minutes after the tasks are closed. There is no persistence of closed tasks in the Digital Routing service. This 15-minute interval is a system setting and you cannot modify it.

The individual tasks that are escalated to CCE can be queried using the Route_Call_Detail (RCD) and Termination_Call_Detail (TCD). For more information see the *All Tables* chapter in the Database Schema Handbook for Cisco Unified ICM/Contact Center Enterprise.

CHAPTER 9

Courtesy Callback

Capabilities

Courtesy Callback reduces the time callers have to physically wait on hold or in a queue. The feature enables your system to offer callers (who meet your criteria) the option to receive a courtesy callback by the system instead of waiting on the phone for an agent. The caller who has been queued by Unified CVP can hang up and subsequently be called back when an agent is close to becoming available (preemptive callback).

Preemptive callback does not change the time a customer must wait to be connected to an agent, but rather enables the caller to hang up and not be required to remain in queue listening to music. Callers who have remained in queue or have undergone the callback treatment appears the same to agents answering the call.

If the caller decides to be called back by the system, they leave their name and phone number. Their request remains in the system and when the system determines that an agent will be available soon (or is available), then the system places a call back to the caller. The caller answers the call and confirms that they are the original caller and the system connects the caller to the agent after a brief wait.

In the event that the caller cannot be reached after a configurable max number and frequency of retries, the callback is aborted and the database status is updated appropriately. You can run reports to determine if any manual callbacks are necessary based on your business rules.

Note that you cannot schedule a callback for a specific time.

Note There are a number of prerequisites and design considerations for using this feature. See the Cisco Unified Customer Voice Portal Release Solution Reference Network Design (SRND) guide.

Note The Cisco Unified Customer Voice Portal Release Solution Reference Network Design (SRND) guide also describes how the system determines customer wait time and when to call the customer for the callback.

Callback Criteria

In your callback script, you can establish criteria for offering a caller a courtesy callback. Examples of callback criteria include:

- Number of minutes a customer is expected to be waiting *in queue* that exceeds a maximum number of minutes (based on your average call handling time per customer)

Note The included example scripts use this method for determining callback eligibility.

- Assigned status of a customer (*gold* customers may be offered the opportunity to be called back instead of remaining on the line)
- The service a customer has requested (sales calls, or system upgrades, for example, may be established as callback criteria)

Sample Scripts and Audio Files for Courtesy Callback

The courtesy callback feature is implemented using Unified CCE scripts. The installation provides a set of modifiable example CCE scripts, call studio scripts, and audio files to get you started. You can use these scripts in your implementation after making a few required changes.

Related Topics

CCE Script for Courtesy Callback, on page 109

Typical Use Scenario

If the caller decides to be called back by the system, they leave their name and phone number. Their request remains in the system and the EWT fires when the system places a callback to the caller. The caller answers the call and confirms that they are the original caller, and the system connects the caller to the agent after a short wait.

Note Courtesy Callback is supported for IP originated calls as well.

A typical use of the Courtesy Callback feature follows this pattern:

1. The caller arrives at Unified CVP and the call is treated in the IVR environment.
2. The Call Studio and Packaged CCE Courtesy Callback scripts determine if the caller is eligible for a callback based on the rules of your organization (such as in the prior list of conditions).
3. If a courtesy callback can be offered, the system tells the caller the approximate wait time and offers to call the customer back when an agent is available.
4. If the caller chooses not to use the callback feature, queuing continues as usual.

   Otherwise, the call continues as indicated in the remaining steps.

5. If the caller chooses to receive a callback, the system prompts the caller to record their name and to key in their phone number.

6. The system writes a database record to log the callback information.

Note If the database is not accessible, then the caller is not offered a callback and they are placed in queue.

7. The caller is disconnected from the TDM side of the call. However, the IP side of the call in Unified CVP and Packaged CCEis still active. This keeps the call in the same queue position. No queue music is played, so Voice XML gateway resources used during this time are less than if the caller had actually been in queue.

8. When an agent in the service/skill category the caller is waiting for is close to being available (as determined by your callback scripts), then the system calls the person back. The recorded name is announced when the callback is made to insure the correct person accepts the call.

9. The system asks the caller, through an IVR session, to confirm that they are the person who was waiting for the call and that they are ready for the callback.

   If the system cannot reach the callback number provided by the caller (for example, the line is busy, RNA, network problems, etc.) or if the caller do not confirm they are the caller, then the call is not sent to an agent. The agent is always guaranteed that someone is there waiting when they take the call. The system assumes that the caller is already on the line by the time the agent gets the call.

   This feature is called preemptive callback as the system assumes that the caller is already on the line by the time the agent gets the call and that the caller has to wait minimal time in queue before speaking to an agent.

10. The system presents the call context on the agent screen-pop, as usual.

In the event that the caller cannot be reached after a configurable maximum number and frequency of retries, the callback is stopped and the database status is updated appropriately. You can run reports to determine if any manual callbacks are necessary based on your business rules.

See the *Configuration and Administration Guide for Cisco Unified Customer Voice Portal* http://www.cisco.com/en/US/products/sw/custcosw/ps1006/products_installation_and_configuration_guides_list.html guide which provides a call flow description of the function of the scripts that provide the Courtesy Callback feature.

Initial Setup

The Courtesy Callback feature must be configured on the following servers/gateways:

- Ingress Gateway (IOS configuration)
- VXML Gateway (IOS configuration)
- Virtualized Voice Browser (no special configuration is required if you use VVB instead of VXML Gateway)
- Reporting Server (through the PCCE Administration Tool)
- Media Server (upload of Courtesy Callback media files)

- Unified CVP VXML Server (upload of Call Studio Scripts)
- Packaged CCE

Courtesy Callback Design Considerations

The following design considerations apply for Courtesy Callback feature:

- During Courtesy Callback, callback is made using the same Ingress Gateway through which the call arrived.

Note In Courtesy Callback, outbound calls cannot be made using any other Egress Gateway.

- Calls that allow Callback must be queued using a Unified CVP VXML Server.
- The Unified CVP Reporting Server is a prerequisite for Courtesy Callback.
- Answering machine detection is not available for this feature. During the callback, the best that can be done is to prompt the caller with a brief IVR session and acknowledge with DTMF that they are ready to take the call.
- Calls that are transferred to agents using DTMF *8, TBCT, or hookflash cannot use the Courtesy Callback feature.
- Callbacks are a best-effort function. After a limited number of attempts to reach a caller during a callback, the callback is terminated and marked as failed.
- Customers must configure the allowed or blocked numbers that Callback is allowed to place calls through the Unified CVP Operations Console.
- Media inactivity detection feature on the VXML Gateway can impact waiting callback calls. For more information, see the *Configuration Guide for Cisco Unified Customer Voice Portal (CVP)*.
- Courtesy Callback requires an accurate EWT calculation for its optimal behavior.

  Consider the following recommendations to optimize the EWT, when using Precision Queues for Courtesy Callback :

  - Queue the calls to a single Precision Queue
  - Do not include a `Consider If` expression when you configure a step.
  - Do not include a wait time between steps or use only one step in the Precision Queue.

Configure the Ingress Gateway for Courtesy Callback

The ingress gateway where the call arrives is the gateway that processes the preemptive callback for the call, if the caller elects to receive a callback.

Note A sip-profile configuration is needed on ISR for the courtesy callback feature, only when deploying an IOS-XE version affected by CSCts00930. For more information on the defect, access the Bug Search Tool at https://sso.cisco.com/autho/forms/CDClogin.html.

For more information about sip-profile configuration, see *Design Guide for Cisco Unified Customer Voice Portal*, at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-implementation-design-guides-list.html.

Step 1 Download the **survivability.tcl** file from DevNet: https://developer.cisco.com/site/customer-voice-portal/downloads/courtesy-callback-scripts/

Step 2 Copy the survivability.tcl file to the flash memory of the Ingress gateway.

Step 3 Log in to the ingress gateway.

Step 4 If survivability is not already configured, configure it as described in the "Call Survivability" section of 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 5 To add services to the gateway, you must be in enabled-config application mode. Type these commands at the gateway console:

```
GW81#en
GW81#config
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line.  End with CNTL/Z.
GW81(config)#application
GW81(config-app)#
```

Step 6 Add the following to the survivability service:

```
param ccb id:<host name or ip of this gateway>;loc:<location name>;trunks:<number of callback trunks>
```

Where the definitions of the preceding fields are:

- *id*: A unique identifier for this gateway and is logged to the database to show which gateway processed the original callback request.
- *loc*: An arbitrary location name specifying the location of this gateway.
- *trunks*: The number of DS0's reserved for callbacks on this gateway. Limit the number of T1/E1 trunks to enable the system to limit the resources allowed for callbacks.

The following example shows a basic configuration:

```
service cvp-survivability flash:survivability.tcl
param ccb id:10.86.132.177;loc:doclab;trunks:1
!
```

If you are updating the survivability service, or if this is the first time you created the survivability service, remember to load the application using the command:

```
call application voice load cvp-survivability
```

Step 7 Create the incoming dial peer, or verify that the survivability service is being used on your incoming dial peer. For example:

```
dial-peer voice 978555 pots
service cvp-survivability
incoming called-number 9785551234
direct-inward-dial
!
```

Note: We support both POTS and VoIP dial peers that point to a service provider.

Step 8 Create outgoing dial peers for the callbacks. These dial peers place the actual callback out to the PSTN. For example:

```
dial-peer voice 978554 pots
destination-pattern 978554....
no digit-strip
port 0/0/1:23
!
```

Step 9 Use the following configuration to ensure that SIP is set up to forward SIP INFO messaging:

```
voice service voip
signaling forward unconditional
```

Step 10 Save your changes.

Configure the VXML Gateway for Courtesy Callback

To configure the VXML gateway for Courtesy Callback:

Step 1 Download the **cvp_ccb_vxml.tcl** file from DevNet: https://developer.cisco.com/site/customer-voice-portal/downloads/courtesy-callback-scripts/

Step 2 Copy the **cvp_ccb_vxml.tcl** to the flash memory of the VXML gateway.

Step 3 To add services to the gateway, you must be in enabled-config application mode. Type these commands at the gateway console:

```
GW81#en
GW81#config
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line.  End with CNTL/Z.
GW81(config)#application
GW81(config-app)#
```

Step 4 Add the cvp_cc service to the configuration:

```
service cvp_cc flash:cvp_ccb_vxml.tcl
```

The service does not require any parameters.

Load the application with the command:

```
call application voice load cvp_cc
```

Note The media-inactivity detection feature must be turned off in the VXML Gateway to successfully call back the caller. With media-inactivity enabled on the VXML Gateway, the cvp_cc service will disconnect the waiting callback calls after 'ip rtcp report interval' * 1000-milliseconds interval. This configuration becomes important in a colocated Ingress/VXML setup where media inactivity timers are always enabled. In such scenarios, the 'ip rtcp report interval' must be increased to support the maximum allowable waiting for a callback call as defined by the solution requirements.

Step 5 On the VoIP dial-peer that defines the VRU leg from Packaged CCE, verify that the codec can be used for recording. The following example shows that g711ulaw can be used for recording in Courtesy Callback:

```
dial-peer voice 123 voip
service bootstrap
incoming called-number 123T
dtmf-relay rtp-nte
codec g711ulaw
no vad
!
```

In other words, this example shows the g711ulaw codec set on the 123 voip dial-peer. The codec must be specified explicitly. A codec class cannot be used because recording will not work.

Step 6 Use the following configuration to ensure that SIP is set up to forward SIP INFO messaging:

```
voice service voip
signaling forward unconditional
```

Step 7 VXML 2.0 is required to play the beep to prompt the caller to record their name in the BillingQueue example script. Add the following text to the configuration so the VXML Server uses VXML 2.0:

```
vxml version 2.0
```

Note Whenever vxml version 2.0 is enabled on the gateway,vxml audioerror is off by default. When an audio file cannot be played, error.badfetch will *not* generate an audio error event. To have the gateway generate an error.badfetch event when a file cannot be played, enable vxml audioerror in your gateway configuration. The following example uses config terminal mode to add both commands:

```
config t
vxml version 2.0
vxml audioerror
exit
```

Step 8 Save your changes.

Configure the Reporting Server for Courtesy Callback

Before you begin

A Reporting Server is required for the Courtesy Callback feature. The Reporting Server must be installed prior to completing the following task. For instructions, see the *Cisco Packaged Contact Center Enterprise*

Installation and Upgrade Guide at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-installation-guides-list.html.

Step 1 Log in to PCCE Administration Tool.

Step 2 In **Unified CCE Administration**, choose **Overview** > **Features** > **Courtesy Callback**.

Step 3 From the **Site** drop-down list, choose a site for which you want configure the Courtesy Callback feature. By default, it is 'Main'.

Step 4 From the **CVP Reporting Server** drop-down list, choose a Reporting Server to use for storing Courtesy Callback data.

Note The list includes all the Reporting Servers configured for the site.

If you leave the selection blank, no Reporting Server is associated with the Courtesy Callback deployment.

Step 5 In the **Dialed Number Configuration** section, complete the following:

Fields	Required?	Description
Maximum Callbacks per Dialed Number	Yes	By default, the **Unlimited** option is selected, which is equivalent to an unlimited number of callbacks offered per calling number. The maximum value is 1000. To limit the number of calls, from the same calling number that are eligible to receive a callback: **a.** Select the **Limited** option. **b.** Enter a postive number in the text field to allow Courtesy Callback to validate and allow the specified number of callbacks per calling number.
Allow unmatched Dialed Numbers	Yes	Check the **Allow unmatched Dialed Numbers** check box to allow callbacks to the dialed numbers that are not available in the **Allowed Dialed Number Patterns** list. **Note** If no dialed numbers are present in the **Allowed Dialed Number Patterns** list, then Courtesy Callback does not allow any callbacks.
Allowed Dialed Number Patterns	No	The list of allowed dialed numbers to which callbacks can be sent. By default, the list includes preconfigured allowed dialed number patterns. To add a dialed number pattern: **a.** Click the '+' icon and enter a dialed number pattern. **b.** Click **Add**. To remove a dialed number pattern, click the 'x' icon associated with the number in the list.

Fields	Required?	Description
Denied Dialed Number Patterns	No	The list of denied dialed numbers to which callbacks are never sent. By default, the list includes preconfigured denied dialed number patterns. To add a dialed number pattern: **a.** Click the '+' icon and enter a dialed number pattern. **b.** Click **Add**. To remove a dialed number pattern, click the 'x' icon associated with the number in the list. Denied numbers takes precedence over allowed numbers. • Wildcarded DN patterns can contain "." and "X" in any position to match a single wildcard character • Any of the wildcard characters in the set ">*!T" will match multiple characters but can only be used trailing values because they will always match all remaining characters in the string • The highest precedence of pattern matching is an exact match, followed by the most specific wildcard match. • When the number of characters are matched equally by wildcarded patterns in both the Allowed Dialed Numbers and Denied Dialed Numbers lists, precedence is given to the one in the Denied Dialed Numbers list.

Step 6 Click **Save**.

Configure the Media Server for Courtesy Callback

Several Courtesy-Callback-specific media files are included with the sample scripts for Courtesy Callback.

You can download the Courtesy Callback specific sample media files and scripts (CCBAudioFiles.zip) from DevNet **Customer Voice Portal (CVP)** > **Downloads** > **Courtesy Callback Sample Scripts** at https://developer.cisco.com/site/customer-voice-portal/downloads/courtesy-callback-scripts/

The special audio files should be unzipped and copied to your media server.

CCBAudioFiles.zip has callback-specific application media files under `C:\inetpub\wwwroot\en-us\app` and media files for Say It Smart under `C:\inetpub\wwwroot\en-us\sys`.

Note If you selected the Media File installation option, during the Unified CVP install, the audio files were unzipped and copied to `C:\inetpub\wwwroot\en-us\app` on the installation server.

Note CCBAudioFiles.zip also contains media files for Say It Smart. During installation, these files are copied to `C:\inetpub\wwwroot\en-us\sys`. Copy these files to your media server, if you do not have them there already.

Note The sample scripts are set up to use the default location of `http://<server>:<port>/en-us/app` for the audio files. Later in this configuration process you will change the <server> and <port> parameters in the default location of the audio files in the example scripts to be your media server IP address and port number.

Configure Call Studio Scripts for Courtesy Callback

The Courtesy Callback feature is controlled by a combination of Call Studio scripts and ICM scripts. Refer to the *Solution Design Guide for Cisco Unified Contact Center Enterprise* (formerly the *Cisco Unified Customer Voice Portal Solution Reference Network Design*) at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-implementation-design-guides-list.html for a discussion of the script logic.

To configure the Call Studio scripts, perform the following procedure:

Note This example follows the BillingQueue example application.

Step 1 Extract the example Call Studio Courtesy Callback scripts contained in CourtesyCallbackStudioScripts.zip to a folder of your choice on the computer running Call Studio.

You can access the .zip file from the following two locations:

- From the Unified CVP install media in \CVP\Downloads and Samples\Studio Samples\CourtesyCallbackStudioScripts
- From DevNet (**Customer Voice Portal (CVP)** > **Downloads** > **Courtesy Callback Sample Scripts**) at https://developer.cisco.com/site/customer-voice-portal/downloads/courtesy-callback-scripts/

Step 2 Each folder contains a Call Studio project having the same name as the folder. The five individual projects comprise the Courtesy Callback feature.

Do *not* modify the following scripts:

- CallbackEngine: Keeps the VoIP leg of the call alive when the caller elects to receive the callback (and ends the call) and when the caller actually receives the callback. Do **not** modify this script.
- CallbackQueue: Handles the keepalive mechanism for the call when callers are in queue and listening to the music played by BillingQueue.

Modify the following scripts to suit your business needs:

- BillingQueue: Determines the queue music played to callers.

- CallbackEntry: Modify the initial IVR treatment a caller receives when entering the system and is presented with an opportunity for a callback.
- CallbackWait: Modify the IVR treatment a caller receives when they respond to the callback.

Note Do not change the CCB application names.

Step 3 Start Call Studio by selecting **Start** > **Programs** > **Cisco** > **Cisco Unified Call Studio**.

Step 4 In Call Studio, select **File** > **Import**.

Step 5 In the Import dialog box, expand the Call Studio folder and select **Existing Call Studio Project Into Workspace** .

Step 6 Click **Next**.

Step 7 In the Import Call Studio Project From File System dialog, browse to the location where you extracted the call studio projects. For each of the folders that were unzipped, select the folder (for example BillingQueue) and click **Finish**.

The project is imported into Call Studio. Repeat this action for each of the five folders.

When you are finished importing the five folders, you should see five projects in the *Navigator* window in the upper left.

Step 8 Update the Default Audio Path URI field in Call Studio to contain the IP address and port value for your media server.

For each of the Call Studio projects previously unzipped, complete the following steps:

a) Select the project in the Navigator window of Call Studio.
b) Click **Project** > **Properties** > **Call Studio** > **Audio Settings**.
c) On the Audio Settings window, modify the Default Audio Path URI field by supplying your server IP address and port number for the <*Server*> and <*Port*> placeholders.
d) Click **Apply**, and then click **OK**.

Step 9 Billing Queue Project: If desired, change the music played to the caller while on hold.

You can also create multiple instances of this project if you want to have different hold music for different clients, for example, BillingQueue with music for people waiting for billing, and SalesQueue with music for people waiting for sales. You also need to point to the proper version (BillingQueue or SalesQueue) in the ICM script. In the ICM script, the parameter queueapp=BillingQueue would also have a counterpart, queueapp=SalesQueue.

The CallbackEntry Project (in the following step) contains a node called SetQueueDefaults. This node contains the value Keepalive Interval which must be *greater* than the length of the queue music you use. Refer to the Keepalive Interval in the next step for details.

Step 10 Callback Entry Project: If desired, in the CallbackEntry project, modify the caller interaction settings in the SetQueueDefaults node.

This step defines values for the default queue. You can insert multiple SetQueueDefaults elements here for each queue name, if it is necessary to customize configuration values for a particular queue. If you do not have a SetQueueDefaults element for a given queue, the configuration values in the default queue are used.

Note You can define a Callback_Set_Queue_Defaults node with **Queue Name** parameter set to default. Configuration defined in this default node will be picked whenever a queue type is encountered for which there are no explicitly defined values.

a) In the Call Studio Navigator panel, open the CallBackEntry project and double click **app.callflow** to display the application elements in the script window.
b) Open the Start of Call page of the script using the tab at the bottom of the script display window.
c) Select the SetQueueDefaults node.

d) In the **Element Configuration panel**, select the Setting tab and modify the following default settings as desired:

For the SetQueueDefaults element, the caller interaction values in the Start of Call and the Wants Callback elements, may be edited.

Step 11 Perform the following steps.

a. Set the path for the storage of recorded caller names.

b. Select app.callflow.

c. In the CallbackEntry project, on the Wants Callback page, highlight the Record Name node and click the **Settings** tab in the Element Configuration window of Call Studio.

d. In the Path setting, change the path to the location where you want to store the recorded names of the callers.

By default, Call Studio saves the path string in your VXML Server audio folder. If you are using the default path, you can create a new folder called recordings in the `%CVP_HOME%\VXMLServer\Tomcat\webapps\CVP\audio\` folder on the VXML Server. If you are using IIS as your media server, create a new folder called recordings under `C:\Inetpub\wwwroot\en-us\app` and set that as the path for recordings.

Step 12 Set the name of the Record name file.

From the CallbackEntry project on the Wants Callback page, highlight the **Add Callback to DB** node and select the **Settings** tab in the Element Configuration window of Call Studio.

Change the **Recorded name file** setting to match the location of the recording folder you created.

This setting references the URL of the recordings folder, whereas the Path setting references the file system path.

The AddCallback element setting in the CallbackEntry project is configured to do automatic recorded file deletions. If automatic recorded file deletion is not desired, then remove the value of the Recorded name path setting in the AddCallback element. This removal action assumes that you will be doing the deletion or management of the recorded file yourself.

Step 13 In the CallbackEntry project on the Callback_Set_Queue_Defaults node, be sure the keepalive value (in seconds) is greater than the length of the queue music being played. The default is 120 seconds.

Step 14 Save the **CallbackEntry** project.

Step 15 CallbackWait Project: Modifying values in the CallbackWait application.

In this application, you can change the IVR interaction that the caller receives at the time of the actual callback. The caller interaction elements in **CallbackWait** > **AskIfCallerReady (page)** may be modified. Save the project after you modify it. The WaitLoop retry count can also be modified from the default of six retries in the Check Retry element. This will allow a larger window of time to pass before the call is dropped from the application. It is used in a failure scenario when the CallbackServlet on the reporting server cannot be reached. For instance, in a reboot or a service restart, this allows more time for the reporting server to reload the entry from the database when it is initializing. If the reporting server is not online within the retry window, then the entry will not be called back.

Step 16 Validate each of the five projects associated with the Courtesy Callback feature by right-clicking each Courtesy Callback project in the Navigator window and selecting **Validate**.

Deploy VXML Application to VXML Server

You can deploy a VXML application to the VXML Server using the File Transfer feature in Packaged CCE Administration web application.

Step 1 After validating and saving your applications, in the navigator panel of Call Studio (top left), right-click and select all the applications you want to deploy.

Step 2 Click **Deploy**.

Step 3 In the Deploy Destination area, select **Archive File** and click **Browse**.

Step 4 Navigate to the archive folder that you have set up; for example, `C:\Users\Administrator\Desktop\Sample`.

Step 5 Enter the name of the file; for example, **Samplefile.zip**.

Step 6 Click **Save**.

Step 7 In the Deploy Destination area, click **Finish**.

Step 8 Log in to the Packaged CCE Administration web application from the principal AW machine.

Step 9 Choose **Overview** > **Call Settings** > **IVR Settings** > **File Transfers**.

Step 10 Click **New** to open the **New File Transfer** page.

Step 11 Click **Add to Server** to open the **Upload File** pop-up.

Note You can upload one file at a time.

Step 12 Click **Click to select** and select a zip file to upload.

Step 13 Click **Upload**.

The file is uploaded to AW and listed under **Available Files in the Server**.

Note You can hover over a row and click the **x** icon to delete a file from the server.

Step 14 Select one or more sites for the file transfer.

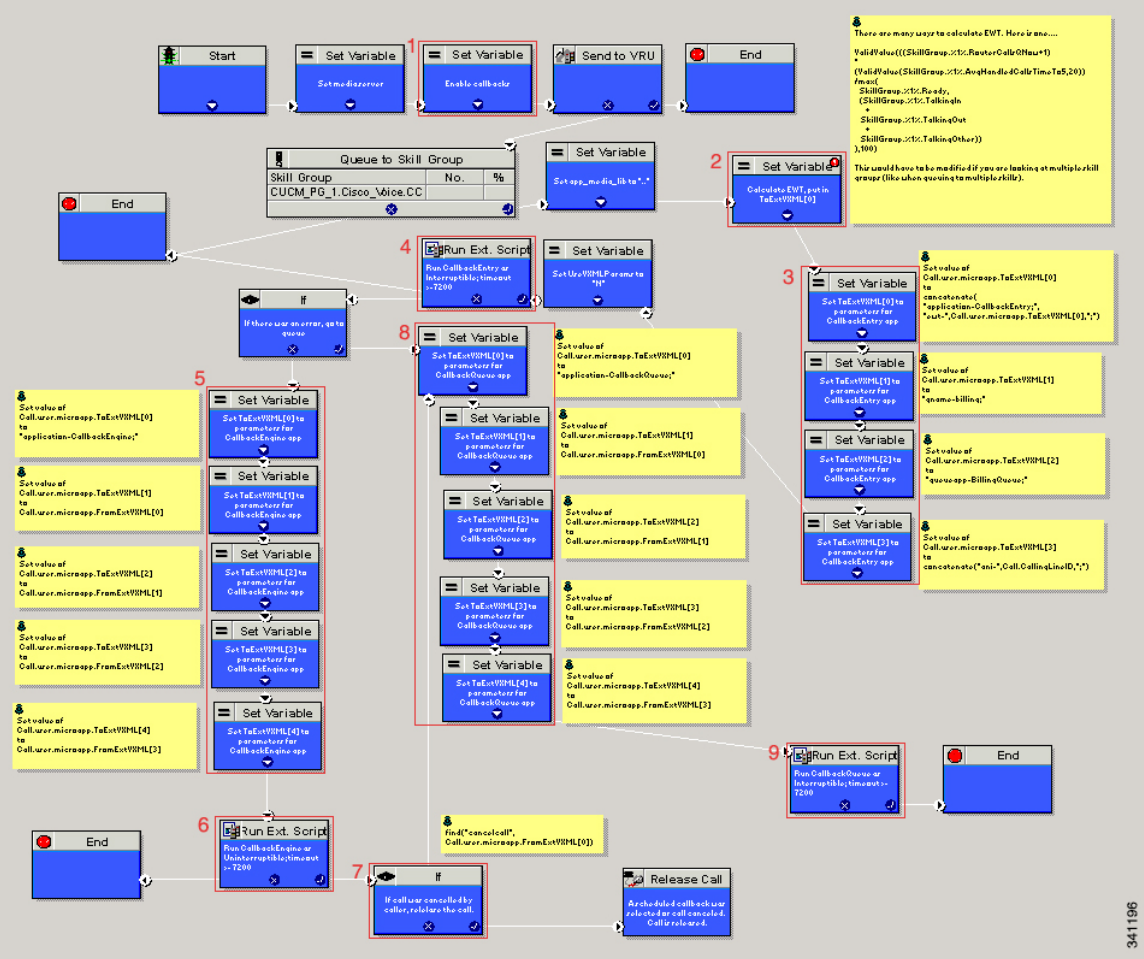
Step 15 On the **Available Files in the Server** list, select the files that you want to transfer and click **Save**.
This initiates the transfer of the selected files to VXML Server of the selected sites.

CCE Script for Courtesy Callback

The following discussion provides an overview of the scripts used for the courtesy callback feature. There are nine numbered blocks, or sets of blocks, identified in the following figure.

Note In the following example, the yellow comment blocks describe first the value being set and then the place where the value is being sent.

Figure 5: Setting Value for Courtesy Callback


Start
Set Variable
Set mediaserver
1
Set Variable
Enable callbacks
Send to VRU
End
There are many ways to calculate EWT. Here is one....
ValidValue(((SkillGroup.%1%.RouterCallsQNow+1)
*
(ValidValue(SkillGroup.%1%.AvgHandledCallsTimeTo5,20))
/max(
SkillGroup.%1%.Ready,
(SkillGroup.%1%.TalkingIn
+
SkillGroup.%1%.TalkingOut
+
SkillGroup.%1%.TalkingOther))
),100)
This would have to be modified if you are looking at multiple skill groups (like when queuing to multiple skills).
Queue to Skill Group
Skill Group
No.
%
CUCM_PG_1.Cisco_Voice.CC
Set Variable
Set app_media_lib to ".."
2
Set Variable
Calculate EWT, put in ToExtVXML[0]
End
4
Run Ext. Script
Run CallbackEntry as Interruptible; timeout >-7200
Set Variable
Set UseVXMLParams to "N"
3
Set Variable
Set ToExtVXML[0] to parameters for CallbackEntry app
Set value of Call.user.microapp.ToExtVXML[0] to concatenate("application-CallbackEntry;", "out-",Call.user.microapp.ToExtVXML[0],";")
If
If there was an error, go to queue
8
Set Variable
Set ToExtVXML[0] to parameters for CallbackQueue app
Set value of Call.user.microapp.ToExtVXML[0] to "application-CallbackQueue;"
Set Variable
Set ToExtVXML[1] to parameters for CallbackEntry app
Set value of Call.user.microapp.ToExtVXML[1] to "qname-billing;"
5
Set value of Call.user.microapp.ToExtVXML[0] to "application-CallbackEngine;"
Set Variable
Set ToExtVXML[0] to parameters for CallbackEngine app
Set Variable
Set ToExtVXML[1] to parameters for CallbackQueue app
Set value of Call.user.microapp.ToExtVXML[1] to Call.user.microapp.FromExtVXML[0]
Set Variable
Set ToExtVXML[2] to parameters for CallbackEntry app
Set value of Call.user.microapp.ToExtVXML[2] to "queueapp-BillingQueue;"
Set value of Call.user.microapp.ToExtVXML[1] to Call.user.microapp.FromExtVXML[0]
Set Variable
Set ToExtVXML[1] to parameters for CallbackEngine app
Set Variable
Set ToExtVXML[2] to parameters for CallbackQueue app
Set value of Call.user.microapp.ToExtVXML[2] to Call.user.microapp.FromExtVXML[1]
Set Variable
Set ToExtVXML[3] to parameters for CallbackEntry app
Set value of Call.user.microapp.ToExtVXML[3] to concatenate("ani-",Call.CallingLineID,";")
Set value of Call.user.microapp.ToExtVXML[2] to Call.user.microapp.FromExtVXML[1]
Set Variable
Set ToExtVXML[2] to parameters for CallbackEngine app
Set Variable
Set ToExtVXML[3] to parameters for CallbackQueue app
Set value of Call.user.microapp.ToExtVXML[3] to Call.user.microapp.FromExtVXML[2]
Set value of Call.user.microapp.ToExtVXML[3] to Call.user.microapp.FromExtVXML[2]
Set Variable
Set ToExtVXML[3] to parameters for CallbackEngine app
Set Variable
Set ToExtVXML[4] to parameters for CallbackQueue app
Set value of Call.user.microapp.ToExtVXML[4] to Call.user.microapp.FromExtVXML[3]
Set value of Call.user.microapp.ToExtVXML[4] to Call.user.microapp.FromExtVXML[3]
Set Variable
Set ToExtVXML[4] to parameters for CallbackEngine app
9
Run Ext. Script
Run CallbackQueue as Interruptible; timeout >-7200
End
6
End
Run Ext. Script
Run CallbackEngine as Uninterruptible;timeout >-7200
find("cancelcall", Call.user.microapp.FromExtVXML[0])
7
If
If call was cancelled by caller, release the call.
Release Call
A scheduled callback was selected or call canceled. Call is released.
341196

The following bullets provide descriptions for the numbered blocks in the preceding graphic:

- Block 1: Enable callback or shut it off.
- Block 2: Compute average wait time. Once the caller is *in queue*, calculate the Estimated Wait Time (EWT) for that queue and place the value in ToExtVXML[0].

  If there is poor statistical sampling because of sparse queues and the wait time cannot be calculated in the VXML Server, use the ICM-calculated estimated wait time.

  One method of calculating EWT (the method used in this example) is:

```
ValidValue(((SkillGroup.%1%.RouterCallsQNow+1)
*
(ValidValue(SkillGroup.%1%.AvgHandledCallsTimeTo5,20))
/max(
SkillGroup.%1%.Ready,
(SkillGroup.%1%.TalkingIn
+
SkillGroup.%1%.TalkingOut
+
SkillGroup.%1%.TalkingOther))
),100)
```

Modify this method if you are looking at multiple skill groups (when queuing to multiple skills).

- Block 3: Set up parameters to be passed.
- Block 4: Run this block and prompt the caller. If the caller does not accept the offer for a callback, keep the caller in the queue and provide queue music.
- Block 5: Set up variables. Call flow returns to this block if the caller elects to receive a callback. Otherwise, the call remains queuing in the queuing application (BillingQueue in this example) on the VXML Server.
- Block 6: Run external to Callback engine to keep the call alive. If the agent becomes available and there is no caller, then agent can't interrupt (do not want an agent to pick up and have no one there).
- Block 7: Has the caller rejected the callback call? If no, then go to block 8.
- Block 8: Compute average wait time, as in block 2.
- Block 9: Set up variables.
- Block 10: Put caller briefly into queue (after caller accepts the actual callback call).

Modifiable Example Scripts and Sample Audio Files

The courtesy callback feature is implemented using Unified CCE scripts. Modifiable example scripts are provided. These scripts determine whether or not to offer the caller a callback, depending on the callback criteria (previously described). Sample audio files are also provided.

The example scripts and audio files are located on the CVP installation media in the `\CVP\Downloads and Samples\` folder.

The files provided are:

- `CourtesyCallback.ICMS`, the ICM script, in the `ICMDownloads` subfolder
- `CourtesyCallbackStudioScripts.zip`, a collection of Call Studio scripts, in the `helloStudio` Samples subfolder.

  The following example scripts are provided:

  - BillingQueue: Plays queue music to callers. Can be customized.
  - Callback Engine: Keeps the VoIP leg of the call alive when the caller elects to receive the callback (and ends the call) and when the caller actually receives the callback. *Do not* modify this script.
  - CallbackEntry: Initial IVR when caller enters the system and is presented with opportunity for a callback. Can be customized.
  - CallbackQueue: Handles the keepalive mechanism for the call when callers are in queue. Do *not* modify this script.
  - CallbackWait: Handles IVR portion of call when caller is called back. Can be customized.

- `CCBAudioFiles.zip` contains sample audio files that accompany the sample studio scripts.

  If you use `CCBAudioFiles.zip`, unzip the contents onto the media server. `CCBAudioFiles.zip` has Courtesy Callback-specific application media files under `en-us\app` and media files for **Say It**

Smart under `en-us\sys`. If you already have media files for **Say It Smart** on your media server, then you only require the media files under `en-us\app`.

Note The Courtesy Callback sample files and scripts are also available on DevNet (**Customer Voice Portal (CVP)** > **Downloads** > **Courtesy Callback Sample Scripts**) at https://developer.cisco.com/site/customer-voice-portal/downloads/courtesy-callback-scripts/.

Overview of CCE Script Configuration for Courtesy Callback

The provided CCE script for Courtesy Callback contains the necessary sample elements for the Courtesy Callback feature. However, you must merge this script into your existing CCE scripts.

As a starting point and to run a simple test, import the script into the CCE script editor, validate it with the CCE script editor validation tool to locate nodes that need extra configuration (such as for Network VRU scripts and expanded call variables), and then modify the script according to your existing CCE environment.

The general process is as follows:

1. Locate each queue point in every CCE script. For example: Queue To Skill Group, Queue to Enterprise Skill Group, Queue to Scheduled Target or Queue to Agent.
2. Categorize each queue point according to the pool of resources that it is queuing for. Each unique pool of resources will ultimately require a queue in VXML Server if Courtesy Callback is going to be offered for that resource pool. For example, using the following example, QueueToSkill X and QueueToSkill Z are queuing for the exact same resource pool (despite the different queuing order). Queue to Skill Y, however, is queuing to a different pool because it includes Skill Group D.
   - QueueToSkillGroup X is queuing for Skill Group A, B, C in that order.
   - QueueToSkillGroup Y is queuing for Skill Group A, C and D in that order.
   - QueueToSkillGroup Z is queuing for Skill Group C, B, A in that order.
3. Assign a unique name to each unique resource pool. In the above example, we can use names ABC and ACD as example names.
4. For each resource pool, decide whether callbacks will be allowed in that resource pool. If yes, then every occurrence of that resource pool in all ICM scripts must be set up to use VXML Server for queuing. This is to ensure that the Courtesy Callback mechanism in the VXML Server gets a full, accurate picture of each resource pool's queue.
5. For any queue point where Courtesy Callback will be offered, modify all CCE scripts that contain this queue point according to the guidelines in the following CCE script examples.

Configure the CCE Script for Courtesy Callback

Many of the following configuration items relate to the numbered blocks in the diagram and provide understanding for CCE Script for Courtesy Callback (for more information, see CCE Script for Courtesy Callback, on page 109). Steps that refer to specific blocks are noted at the beginning of each step.

To configure CCE to use the sample Courtesy Callback CCE script, perform the following steps:

Step 1 Copy the CCE example script, **CourtesyCallback.ICMS** to the CCE Admin Workstation.

The example CCE script is available in the following locations:

- On the CVP install media in `\CVP\Downloads and Samples\`.
- On DevNet (**Customer Voice Portal (CVP)** > **Downloads** > **Courtesy Callback Sample Scripts**) at https://developer.cisco.com/site/customer-voice-portal/downloads/courtesy-callback-scripts/

Step 2 Perform these steps:

a) Enable the user.CourtesyCallbackEnabled ECC variable.
b) Enable the user.microapp.ToExtVXML ECC variable and verify that it is set up as an array with a maximum array size of 5 elements.
c) Enable the user.microapp.FromExtVXML ECC variable and verify that it is set up as an array with a maximum array size of 4 elements.

Step 3 Make sure the VXML_Server_Interruptible and the VXML_Server_Noninterruptible Network VRU scripts exist.

Step 4 Once the script is open in Script Editor, open the **Set media server** node and specify the URL for your VXML Server.

For example: **http://10.86.132.139:7000/CVP**

With the current implementation of CVP, you do not have to specify the VXML Server URL. You do, however, have to enter *some* numeric value; for example "1" (with quotes).

Step 5 Map the route and skill group to the route and skill group available for courtesy callback.

a) In Script Editor, select **File** > **Import Script...**.
b) In the script location dialog, select the **CourtesyCallback.ICMS** script and click **Open**.
c) In the Import Script - Manual Object Mapping window, map the route and skill group to the route and skill group available for courtesy callback (identified previously).

In Packaged CCE deployments, the route tool does not exist. Routes have one-on-one mappings with skill groups, so when you create a skill group, a route is created with the same name.

Step 6 **Block #2:** If you wish to use a different estimated wait time (EWT), modify the calculation in block #2. You must do this if you use a different method for calculating EWT or if you are queuing to multiple skill groups.

Step 7 **Block #3:** Set up the parameters to be passed to CallbackEntry (VXML application).

Note This step assumes that you have already configured the CCE and expanded call variables not related to Courtesy Callback.

Variable values specific to Courtesy callback include:

ToExtVXML[0] = concatenate("application=CallbackEntry",";ewt=",Call.user.microapp.ToExtVXML[0])

ToExtVXML[1] = "qname=billing";

ToExtVXML[2] = "queueapp=BillingQueue;"

ToExtVXML[3] = concatenate("ani=",Call.CallingLineID,";");

Definitions related to these variables are:

- CallbackEntry is the name of the VXML Server application that is run.
- ewt is calculated in **Block #2**.

- qname is the name of the VXML Server queue into which the call is placed. There must be a unique qname for each unique resource pool queue.
- queueapp is the name of the VXML Server queuing application that is run for this queue.
- ani is the caller's calling Line Identifier.

Step 8 Verify that you have at least one available skill group to map to the skill group in the example script.

Step 9 Save the script, then associate the call type and schedule the script.

For information about scheduling scripts, see the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Administration and Usage

Element Specifications for Courtesy Callback

The example IVR scripts provided for Courtesy Callback work as installed. To change how Courtesy Callback works, you can change the configuration of Courtesy Callback elements. This section lists the elements associated with Courtesy Callback and briefly describes the purpose of each one.

Callback_Add

The `Callback_Add` element is used to add a callback object to the database after all the callback information has been collected from the caller. In addition, it can be optionally configured to automatically delete old recorded files at specified intervals. These recorded files are the files produced by the Record element when the user records their name if they want a call back in the CallbackEntry application.

Callback_Disconnect_Caller

The `Callback_Disconnect_Caller` element is responsible for disconnecting the caller’s leg of the call. The IP leg of the call for Unified CVP is preserved to hold the caller’s *place in line* until the callback is made back to the caller.

Callback_Enter_Queue

The `Callback_Enter_Queue` element is responsible for adding a new caller to the queue. This element must be run for all callers even if the caller may not be offered a callback.

Callback_Get_Status

The `Callback_Get_Status` element is responsible for retrieving all information about the callback related to the current call (if a callback exists).

Callback_Reconnect

The `Callback_Reconnect` element is responsible for reconnecting the caller’s leg of the call.

Callback_Set_Queue_Defaults

The `Callback_Set_Queue_Defaults` element is responsible for updating the DBServlet with the values that should be used for each queue. There is always a *default* queue type. The values are used whenever a queue type is encountered for which there are no explicitly defined values. For example, if an administrator has defined values for a *billing* and *default* queues, but the caller is queued for *mortgages*. In that case, the application uses the values from `Callback_Set_Queue_Defaults`.

Note When the DBServlet is not reachable to check the callback status for the duration of keepalive interval, the callback entry in the Reporting Server gets marked as a stale cached entry and subsequently gets cleared. As a result, a callback is not initiated.

Callback_Update_Status

The `Callback_Update_Status` element is responsible for updating the database after a callback disconnect or reconnect.

Callback_Validate

The `Callback_Validate` element is responsible for verifying whether or not a callback can be offered to the caller during this call. Depending on the outcome of the validation, the Validate element exits with one of four states.

Callback_Wait

The `Callback_Wait` element is responsible for *sleeping* the application for X seconds. The application hands control back to cvp_ccb_vxml.tcl with the parameter wait=X.

CHAPTER 10

Media Server

About Media Server

Many of the optional features in Packaged CCE require a Cisco Unified Customer Voice Portal (CVP) media server to store and serve supporting `.wav` files. This chapter describes how to set up a CVP media server. It also describes expanded call variable settings that are related to the media server and requirements for accessing a media server in call routing scripts.

The features that require a CVP media server include Agent Greeting, Courtesy Callback, Post-Call Survey, and Whisper Announcement.

Prepare a Media Server

A media server is installed by default on each of the CVP servers in a Packaged CCE deployment.

Step 1 Ensure that IIS is properly configured and running on the server. It must be listening on port 80. To validate proper configuration of the media server, launch a browser from a remote machine that is able to ping the CVP server and attempt to access and play one of the default media files installed during the CVP installation such as **http://<cvp_ip>/en-us/app/en_1.wav**. If the file is accessible, the media server is installed correctly.

Note Use Microsoft IIS with Unified CVP. This component is automatically installed as part of the CVP server package installation.

Step 2 Ensure the server is accessible to CVP, Unified CCE, and your agent desktops.

Step 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, and specify to start immediately.
k) On the FTP SSL Settings, select **Allow SSL Connections**.
l) On the **Authentication and Authorization** section select the type of authentication required. If using basic, note the name and password of the account.
m) Select the authorization; for anonymous select **Anonymous users**.
n) Set the read and write permissions.

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

Step 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.

Step 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 expanded call variable **call.user.microapp.ToExtVXML** to send the ftpPath parameter to the recording application. Make sure the expanded call variable length is long enough, or it may get truncated and fail.

Step 6 To allow 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, configure a cache expiration value in IIS Manager. 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.

To configure a cache expiration value in IIS Manager:

a) Find the site you are using, go to the agent greeting folder you created (ag_gr), and then select **HTTP Response Headers**.
b) Click **Set Common Headers** on Actions panel.
c) Select **Expire Web Content** and set the desired value.

What to do next

- 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. Reset the gateway to clear the cache.

  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 VXML Gateway is 30 seconds.

- Add Media Servers in the **Unified CCE Administration** > **Infrastructure** > **Inventory** page. For more information, see the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Reference a Media Server in CCE Scripts

Specify Media Server in Routing Scripts

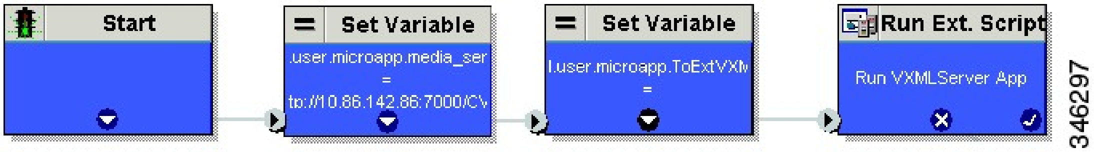
When you configure media servers in CVP, you can specify a default media server. The benefit to specifying a default media server is that your scripts do not need a Set Variable node to access the default media server. For this to work, you must make sure that the files a script requests are stored on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

See the following example.



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`. The physical location of the default storage directory is `c:\inetpub\wwwroot\en-us\app`. To take advantage of this, Packaged 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 Unified CCE Administration, navigate to **Overview** > **Call Settings** > **Route Settings** > **Expanded Call Variables**.

CHAPTER 11

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, on page 325 .

Related Topics

Reverse Proxy Automated Installer, on page 325

Capabilities

Cisco Unified Mobile Agent Description

Mobile Agent enables an agent to use any PSTN phone and in case you want to use the VPN connection (for agent desktop communications) see VPN-less Access to Finesse Desktop, on page 269

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 Packaged 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.

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

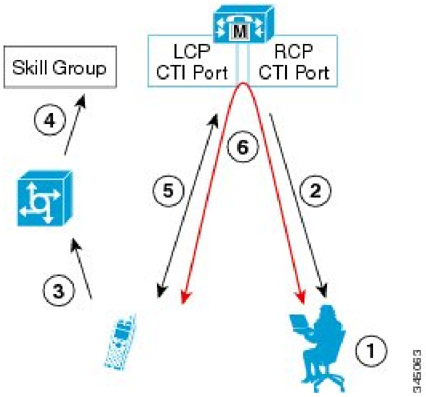
To configure Mobile Agent in call by call delivery mode, you must set the wrap-up timer to at least one second using the Agent Desktop Settings List tool in the Configuration Manager.

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.

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 6: Nailed Connection Call Flow



The nailed connection call flow works as follows:

1. At login, the agent enters the directory number of the local CTI port (LCP) and the remote phone number in the Desktop.

   The remote phone number can be any phone number reachable by Unified CM.

   When the agent clicks the Login button, a call is initiated to the agent's remote CTI port (RCP) and the agent's remote phone rings.
2. When the agent answers the call, the call is then *nailed up*. This means that the agent will remain on this call until the agent logs out or ends the call.

3. A customer's call arrives in the system and, through Packaged CCE configuration and scripting, is queued for a skill group/precision queue. (This is no different than existing processing for local agents.)
4. When the agent clicks the Answer button, the voice path between the agent and the customer phone is established, and the two parties can talk.
5. When the system assigns an agent to the call, the call is routed to the agent's LCP port. The agent then hears the connect tone on the headset.
6. When the call ends, the customer connection is terminated and the agent state returns 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.

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 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 136

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.

- 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.

For more information about Agent Greeting, see Capabilities, on page 19.

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

Phone Requirements

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

Note When Unified Mobile Agent phones are located on a cluster and a SIP Trunk is used to connect the cluster to another cluster under Packaged CCE control, you must either use SIP phones as Mobile Agent phones or select **mtp required** on the Packaged CCE cluster to allow Mobile Agent calls to work.

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 132.

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.
- Packaged CCE intercepts the call arriving on the CTI Port and directs Unified CM to connect the call to the Mobile Agent.

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 Packaged CCE application. The ports are recognized by Packaged CCE 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.

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

- For the total number of supported Unified Mobile Agents and more information about Unified Mobile Agent capability, see *Solution Design Guide for Cisco Packaged Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-technical-reference-list.html.
- 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.

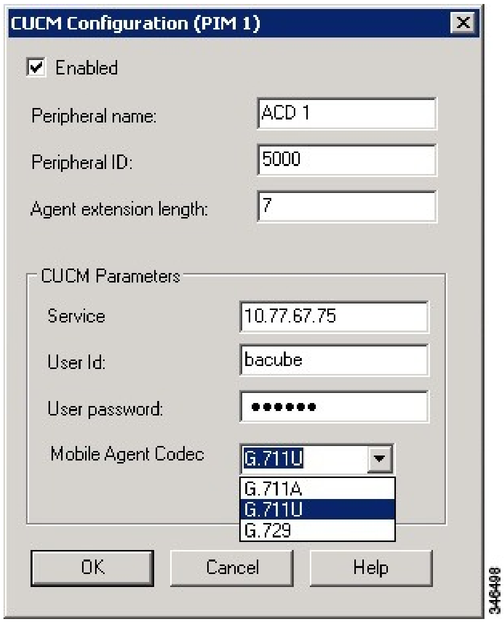
The caller hears a repeated ringtone while Unified CCE makes these connections.

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 7: Mobile Agent Codec Selection



CUCM Configuration (PIM 1)

☑ Enabled

Peripheral name: ACD 1

Peripheral ID: 5000

Agent extension length: 7

CUCM Parameters

Service 10.77.67.75

User Id: bacube

User password: ••••••

Mobile Agent Codec G.711U

Unsupported Features

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

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

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.

Note The service level for Unified Mobile Agent calls might be different than the service level for 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 Packaged 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.

Initial Setup

Summary of Unified Mobile Agent System Configuration Tasks

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

Table 11: 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 128
Configure Unified CM Call Duration Timer	Maximum Call Duration Timer configuration, on page 131
Configure Agent Desk Settings	Agent Desk Setting Configuration for Unified Mobile Agent, on page 131
Configure Media Termination Points	Media Termination Points Configuration, on page 132

Unified CM CTI Port Configuration and Mapping for Unified Mobile Agent

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.

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. 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. You must disable the hold tone.

Perform the following steps to disable the hold tone:

1. Log in to Unified CM Administration and navigate to **System** > **Service Parameters**.
2. Scroll down to the **Tone on Hold Time** field and set the value to **0**.
3. Click **Save**.

Note Because Tone on Hold Time is a cluster-wide setting, it will be applied to all nodes, not just the currently selected node.

Configure Unified CM CTI Ports for Unified Mobile Agent

Perform the following steps to configure CTI Ports.

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 name; click **OK** when finished.

Using the naming convention format LCP*yyyy*:

- LCP identifies the CTI Port as a local device.
- *yyyy* is the local CTI Port.

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 where you want to assign the network CTI port. Do not select Default. (The device pool defines sets of common characteristics for devices.)

Step 8 For Device Security Profile, select **Cisco CTI Port - Standard SCCP Non-Secure Profile**.

Step 9 Click **Save**.

Step 10 Click **Apply config**.

Step 11 In the Association section, select **Add a New DN**.

Step 12 Add a unique directory number for the CTI port you just created.

Step 13 In Maximum Number of Calls, enter **2**.

Step 14 In Busy Trigger, enter **1**.

Step 15 When finished, click **Save**, and click **Apply config**.

Step 16 Repeat the preceding steps to configure the network CTI port.

In Device Name, using the naming convention format RCP*yyyy*, 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.
- *yyyy* is the network CTI port.

The name RCP0000 represents the local port.

Note The port number for both LCP and RCP must be the same even though the directory numbers are different.

Step 17 In Description, enter text that identifies the network CTI port.

Step 18 Use the **Device Pool** drop-down list to choose where you want to assign the network CTI port. Do not select Default.. (The device pool defines sets of common characteristics for devices.)

Step 19 Click **Save**.

Step 20 In the Association Information section, select **Add a New DN**.

Step 21 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 22 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.

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**.

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 Unified Mobile Agent will be logged in *longer than* 12 hours, use the following instructions to either one of the following:

- Increase the Maximum Call Duration Timer setting.
- Disable the timer entirely.

If your Mobile Agent deployment uses intercluster trunks between your CTI ports and your mobile agent's phone, you must set these service parameters on both the local and remote Unified CM clusters.

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 **Cisco CallManager Service**.

The **Service Parameters Configuration** page appears.

Step 4 In the Cluster-wide Parameters (Feature - General) section, specify a **Maximum Call Duration Timer** setting.

Step 5 Click **Save**.

Agent Desk Setting Configuration for Unified Mobile Agent

You can configure Agent Desk Settings through the PCCE Administration tool.

Configure Desk Settings with Unified CCE Administration

This section describes how to configure Desk Settings in Unified CCE Administration to accommodate Unified Mobile Agent features.

The following procedure describes how to configure one desk setting. Repeat this process for each different desk setting in your deployment.

Step 1 In Unified CCE Administration, choose **Overview** > **Desktop Settings** > **Desk Settings**.

Step 2 Click **New** to create a new desk setting or click the name of an existing desk setting to edit it.

Step 3 Complete the required fields.

Step 4 From the Mobile Agent drop-down list, select one of the following options:

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

Step 5 Click **Save**.

Associate Desk Setting with a Mobile Agent

After you have configured agent desk settings, you need to associate the desk setting with a mobile agent.

Step 1 Access Unified CCE Administration.

Step 2 Select **Overview** > **User Setup** > **Agents**.

The List of Agents window appears.

Step 3 Select the agent that you want to associate.

The Edit <agent> window appears.

Step 4 In the Desk Settings box, select the desk setting that has the Mobile Agent enabled.

Step 5 Click **Save**.

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.

The following table lists the steps in configuring MTPs in Unified CM. Make sure you have completed the tasks in the checklist.

Table 12: Checklist for Unified CM SIP Trunk Configuration

Check when done	Task
	Add MTP resources to Unified CM, on page 134
	Configure MTP resources in Unified CM , on page 134
	Associate a Media Resource Group List with Device Pools, on page 134
	Quarantine Unified CM software-based resources, on page 135
	Configure MTPs with SIP Trunks, on page 135
	Enable Call Progress Tones for Agent-Initiated Calls, on page 135
	Verify MTP Resource Utilization, on page 136

Add MTP resources to Unified CM

Perform these steps to add MTPs to Unified CM.

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 configuration 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 that 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 Cisco Call Manager server group you configured on each of your gateways.

Configure MTP resources in Unified CM

The following section explains how to create media resource groups and media resource group lists.

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 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 a Media Resource Group List with Device Pools

The following procedure shows how to associate a media resource group list (MRGL) with device pools.

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:

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.

Configure MTPs with SIP Trunks

If you use SIP trunks, you must configure MTPs. 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.

Step 1 Log in to Unified CM Administration and select **Device** > **Trunk**.

Step 2 Select the trunk on which you want to configure MTPs.

At a minimum all trunks whose destination is unified CVP need to have this configuration. This requirement also applies to all TDM trunks that are used to connect to Mobile Agent phones through service providers.

Step 3 Depending on the scenario listed below, perform the corresponding step. Note that if you configure trunk groups to dynamically insert MTPs, only the calls that require MTPs use them.

- Insert MTPs for inbound and outbound calls through a given trunk: In the Trunk Configuration settings, check the **Media Termination Point Required** check box.
- Dynamically allocate MTPs when Cisco Unified Intelligent Contact Management detects media or signaling incompatibility between the caller and called endpoints: In the Trunk Group Configuration settings, for the DTMF Signaling Method, select **RFC2833**.

Enable Call Progress Tones for Agent-Initiated Calls

For an agent to hear call progress tones for agent-initiated calls, additional configuration is required if **MTP Required** is not enabled. If instead you have dynamic MTP allocation by forcing mismatched DTMF settings, then you should configure the Unified CM to enable Early Offer.

For information on configuring the Unified CM, see the Unified CM product documentation at https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/tsd-products-support-series-home.html. Ringback and other call progress tones are not generated by the Cisco Annunciator, as is the case for regular phones and softphones. Instead, Mobile Agent relies on these tones being generated by the called party (and the Early Offer setting triggers these tones to be sent to the agent).

Note This selection does not affect MTP sizing for IP phones and other endpoints that support RFC 2833 signaling, as is the case for many Cisco phones (including the 6900 series and the 794x and 796x phones).

Verify MTP Resource Utilization

Because Unified CM comes preconfigured with software MTP resources, these resources may sometimes be used to provide MTP for Unified Mobile Agent calls without proper configuration. Cisco does not support the use of Unified CM-based software MTPs. You can quarantine the Unified CM-based MTPs. (See Quarantine Unified CM software-based resources, on page 135.) To ensure that the IOS-based MTPs are being used for Unified Mobile Agents, perform the following verification steps:

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 nodes 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 only the IOS-based resources are active during the Mobile Agent phone call. Also, ensure that all Unified UC-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.

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 Packaged CCE agents. Mobile supervisors can perform all call control functions except for silent monitoring.

Sign in to Cisco Finesse Desktop

Step 1 Enter the following URL in your browser: 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*>:8445/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.

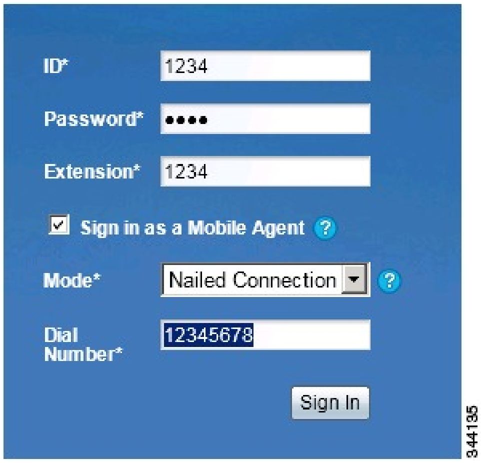
The Mode and Dial Number fields appear.

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.



ID* 1234

Password* ••••

Extension* 1234

☑ Sign in as a Mobile Agent

Mode* Nailed Connection

Dial Number* 12345678

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.
Mode	Call by Call or Nailed Connection
Dial Number	The number of the phone being used.

Step 8 Click **Sign In**.

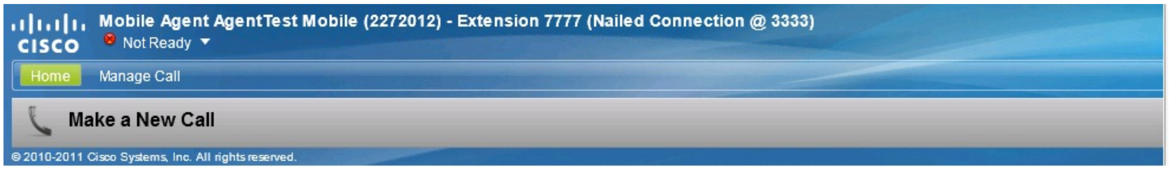
Note In Nailed Connection mode, the desktop must receive and answer a setup call before sign-in is complete.

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

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


cisco
Mobile Agent AgentTest Mobile (2272012) - Extension 7777 (Nailed Connection @ 3333)
Not Ready
Home
Manage Call
Make a New Call
© 2010-2011 Cisco Systems, Inc. All rights reserved.
344134

Enable Ready State

You must be in Ready state to process incoming calls.

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

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 33.

CHAPTER 12

Outbound Option

Capabilities

Features

Outbound Option enables call centers to manage outbound calls. With Outbound Option, you can configure a contact center to automatically dial customer contacts from imported lists and direct a call to an available agent. This application transfers a call to an agent only if a live contact is reached.

A summary of major features in Outbound Option follows:

Automated dialing

The dialer automatically dials contact numbers, screens for busy signals, no answers, and answering machines, and transfers calls to agents. The dialer transfers a call to an available agent only when it reaches a live contact.

Campaigns

Users create calling campaigns using a set of tabs in a graphical user interface (GUI). A campaign is a set of numbers that will be automatically dialed and a set of agents who will talk to contacted customers.

More specifically, a campaign consists of imported contact lists and agent skill groups.

Imported contact lists and Do Not Call lists

You can import lists of customers you want to call and lists of customers who you do not want to call. You can configure Outbound Option to import both types of lists either by continuously polling or at scheduled intervals. You can also specify whether imported lists will replace existing lists or be appended to them.

Agent skill groups

You assign agents to campaigns by using skill groups. A skill group defines a set of agents with specific capabilities, such as language skills, product knowledge, or training that is associated with a campaign. Agents might belong to multiple skill groups and thus be part of multiple campaigns.

Campaign management

Outbound Option uses a dialing list that is associated with a campaign and directs dialers to place calls to customers. The dialer then directs contacted customers to agents.

Choice of dialing modes

Outbound Option supports the following dialing modes:

- Preview mode lets the agent preview the contact information on the desktop and decide whether the SIP dialer should dial a contact.
- Direct Preview mode is similar to Preview mode; however, the dialer places the calls from the agent's phone. This mode prevents abandoned calls and false positive detection of answering machines.
- Progressive mode dials a configured number of calls per available agent.
- Predictive mode adjusts the number of calls dialed per agent based on the current abandon rate.

Callbacks

If a customer requests a callback for a later date and time, the agent can enter the request in the system, and the dialer schedules the call appropriately. The following callback types are supported:

- Personal callbacks specify that the customer receive a callback from the same agent who made the initial contact.
- Regular callbacks are handled by any available agent.

Call analysis

The Call Progress Analysis (CPA) feature uses a combination of call signaling and media stream analysis to identify different types of calls, such as faxes and modems, answering machines, and operator intercepts.

Sequential dialing

The sequential dialing feature allows up to ten phone numbers per customer record.

Abandoned and retry calls

You can configure campaigns to retry abandoned calls.

Campaign prefix digits for dialed numbers

You can configure a prefix for customer number, which can be used to identify specific campaigns.

Activity reports

Outbound Option reporting features include agent, campaign, dialer, and skill groups report templates.

Two-Way Replication

If you choose to enable Outbound Option, you can also enable Outbound Option High Availability. Outbound Option High Availability supports two-way replication between the Outbound Option database on Logger Side A and the Outbound Option database on Logger Side B.

Outbound API

Outbound API allows you to use REST APIs to create, modify, and delete Outbound Option campaigns.

Outbound API provides a streamlined mechanism for creating campaigns with a single preconfigured query rule and import rule.

Administrative scripts are not required for Outbound Option campaigns created with the Outbound API. If an administrative script is provided, the information in the script overrides the information defined in the API.

Outbound API consists of the following APIs:

- Outbound Campaign API: Use this API to define new Outbound Option campaigns, and to view, edit, or delete existing campaigns. You can also use this API to disable all campaigns at once (emergency stop).
- Do Not Call API: Use this API to set the Do Not Call (DNC) import rule configuration for Outbound Option. This prevents the Dialer from dialing the numbers on the DNC list.
- Import API: Use this API to import customer contact information for an Outbound Option campaign.
- Time Zone API: Use this API to list all available time zones and to get information for a specified time zone. You also use this API with the Outbound Campaign API to set the default time zone for an Outbound Option campaign.
- Campaign Status API: Use this API to get the real-time status of running Outbound Option campaigns.
- Personal Callback (PCB) API: Use this API to configure your Outbound Option campaign to handle personal callbacks. You can create personal callback records individually or in bulk. You can also use this API to update or delete personal callback records.

For more information about Outbound API, see the *Cisco Packaged Contact Center Enterprise Developer Reference Guide* at https://developer.cisco.com/site/packaged-contact-center/documentation/.

Dialing Modes

Outbound Option supports various dialing modes, described in the following sections.

Note All dialing modes reserve an agent at the beginning of every Outbound Option call cycle by sending a reservation call to the agent.

Predictive Dialing

In predictive dialing, the dialer determines the number of customers to dial per agent based on the number of lines available per agent and the configured maximum abandon rate. The agent must take the call if that agent is logged in to a campaign skill group.

A Predictive Dialer is designed to increase the resource utilization in a call center. It is designed to dial several customers per agent. After reaching a live contact, the Predictive Dialer transfers the customer to a live agent along with a screen pop to the agent’s desktop. The Predictive Dialer determines the number of lines to dial per available agent based on the target abandoned percentage.

Outbound Option predictive dialing works by keeping outbound dialing at a level where the abandon rate is below the maximum allowed abandon rate. Each campaign is configured with a maximum allowed abandon rate. In Predictive mode, the dialer continuously increments the number of lines it dials per agent until the abandon rate approaches the configured maximum abandon rate. The dialer lowers the lines per agent until the abandon rate goes below the configured maximum. In this way, the dialer stays just below the configured maximum abandon rate. Under ideal circumstances, the dialer internally targets an abandon rate of 85% of the configured maximum abandon rate. Due to the random nature of outbound dialing, the actual attainable abandon rate at any point in time may vary for your dialer.

Preview Dialing

Preview dialing reserves an agent before initiating an outbound call and presents the agent with a popup window. The agent can then Accept or Reject the call with the following results:

- **Accept**: The customer is dialed and transferred to the agent.
- **Reject**: The agent is released. The system then delivers another call to the agent, either another Preview outbound call, or a new inbound call.
- **Rejects-Close**: The agent is released and the record is closed so it is not called again. The system then delivers another call to the agent, either another Preview outbound call or a new inbound call.

Direct Preview Dialing

The Direct Preview mode is similar to the Preview mode, except that the dialer automatically places the call from the agent's phone after the agent accepts. Because the call is initiated from the agent's phone, the agent hears the ringing, and there is no delay when the customer answers. However, in this mode, the agent must deal with answering machines and other results that the Dialer Call Progress Analysis (CPA) handles for other campaign dialing modes.

Note

- A *zip tone* is a tone that announces incoming calls. There is no zip tone in Direct Preview mode.
- If you select *personal callback* as the callback option for a Direct Preview mode campaign, a *personal callback* is dialed in Preview mode. The *personal callback* is processed like a call in the Preview mode.

Progressive Dialing

Progressive Dialing is similar to predictive dialing (see Predictive Dialing, on page 143). The only difference is that in Progressive Dialing mode, Outbound Option does not calculate the number of lines to dial per agent, but allows users to configure a fixed number of lines that will always be dialed per available agent.

Note In the Outbound dialer log, the Progressive dialing mode is also logged as Predictive.

Initial Setup and Maintenance

This section is intended for system administrators who install and configure Packaged CCE. It describes the one-time tasks required to set up Outbound Option. It also discusses occasional upgrade and maintenance tasks. It contains the following topics:

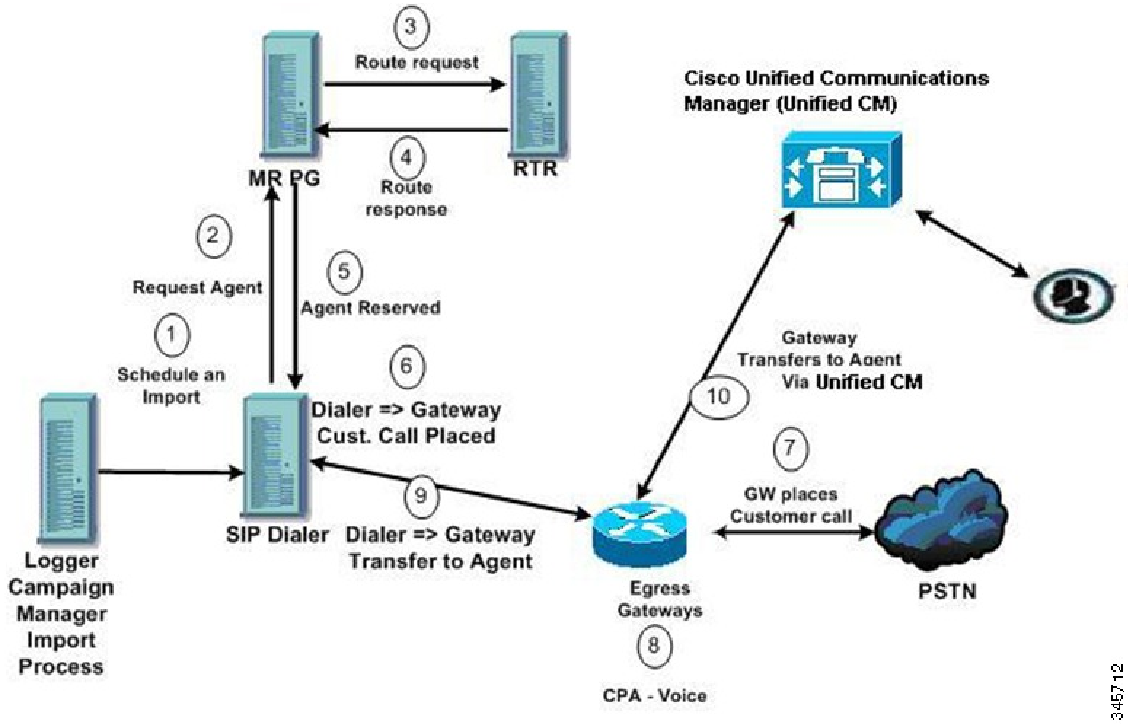
- Unified Communications Manager and Gateway Configuration, on page 148
- Outbound Option Software Installation Steps, on page 155
- Maintenance Considerations, on page 163

Outbound SIP Dialer Call Flows

Call Flow Diagram for Packaged CCE

This figure illustrates a transfer to agent call flow for a SIP dialer Outbound Option campaign.

Figure 8: SIP Dialer Agent Campaign Call Flow


3
Route request
MR PG
RTR
4
Route response
Cisco Unified Communications Manager (Unified CM)
2
Request Agent
5
Agent Reserved
1
Schedule an Import
6
Dialer => Gateway Cust. Call Placed
Gateway Transfers to Agent Via Unified CM
10
7
GW places Customer call
9
SIP Dialer
Dialer => Gateway Transfer to Agent
Logger Campaign Manager Import Process
Egress Gateways
PSTN
8
CPA - Voice
345712

The call flow works as follows:

1. You schedule the import and the campaign starts. Records are delivered to the dialer.
2. The dialer looks for an available agent through the media routing interface.
3. The media routing peripheral gateway (MR PG) forwards the request to the router.
4. The routing script identifies an agent and responds to the MR PG.
5. The MR peripheral interface manager (PIM) notifies the dialer that the agent is reserved.
6. The dialer signals the gateway to call the customer.
7. The gateway calls the customer and notifies the dialer of the attempted call.

8. Call Progress Analysis (CPA) is done at the gateway. When voice is detected, the gateway notifies the dialer.

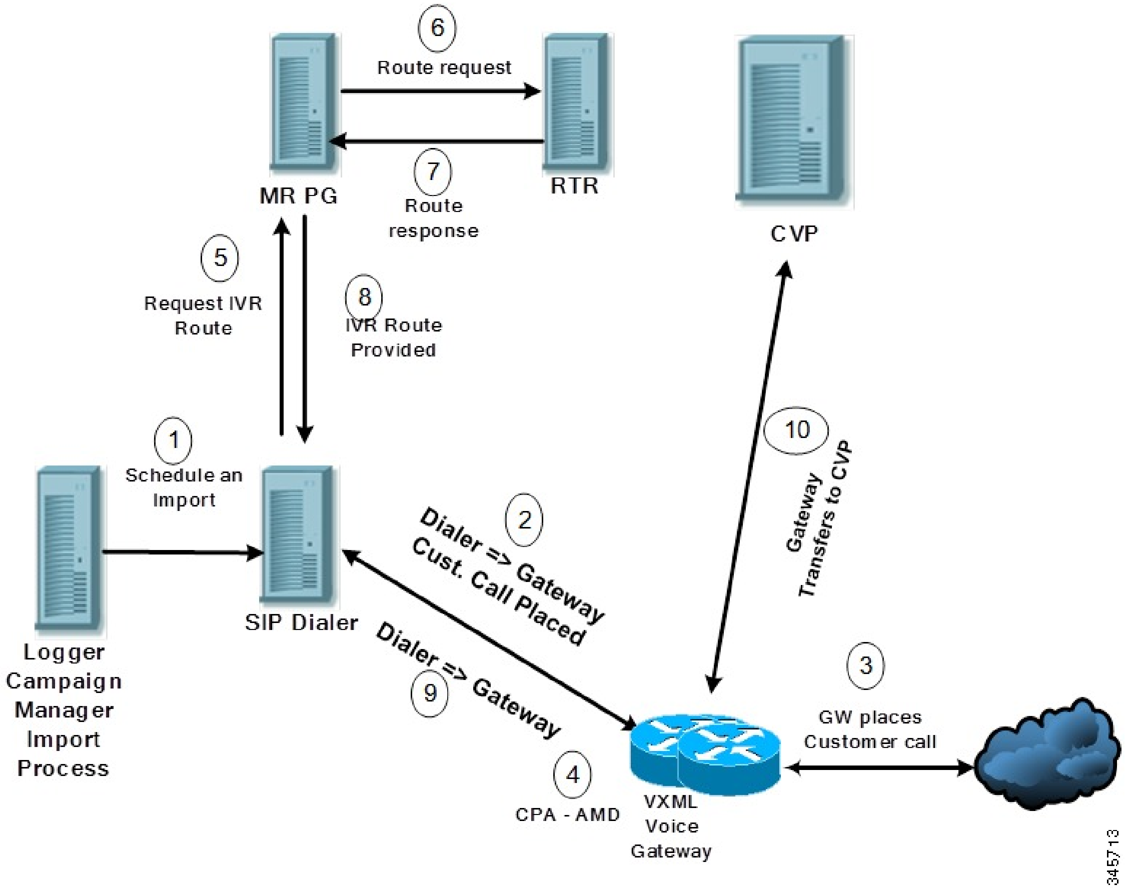
9. The dialer directs the voice gateway to transfer the call to the reserved agent by the agent extension.

10. The gateway directs the call to the agent through Unified Communications Manager (using dial peer configuration to locate the Unified Communications Manager). Media are set up between the gateway and the agent's phone.

Unattended VRU Call Flow Diagram for Packaged CCE

This figure illustrates a transfer-to-VRU call flow for a SIP dialer Outbound Option campaign.

Figure 9: SIP Dialer Unattended IVR Campaign Call Flow



The call flow works as follows:

1. An unattended VRU campaign starts and schedules an import. Records are delivered to the dialer.

2. The dialer sends a SIP INVITE to the voice gateway to call a customer.

3. The gateway calls the customer.

4. Call Progress Analysis (CPA) detects an answering machine (AMD) and notifies the dialer.

5. The dialer sends a VRU route request to the MR PG.

6. The MR PG forwards the route request to the router which invokes the routing script.
7. The router sends the route response with the network VRU label to the MR PG.
8. The MR PG forwards the route response to the dialer.
9. The dialer sends a SIP REFER request for the label to the voice gateway.
10. The voice gateway transfers the call to Unified CVP.

Unified CVP then takes control of the call.

Unified CCE Configuration for Outbound Option

This section provides procedures for configuring Unified CCE for Outbound Option.

Configure the Dialer Component

You deploy the Dialer as a single redundant pair for each Agent PG with agents who handle Outbound Option calls.

Step 1 Make sure that all Packaged CCE services are running.

Step 2 In the **Unified CCE Configuration Manager**, expand **Outbound Option Option** and double-click **Dialer** to display the **Outbound Option Option Dialer configuration** window.

Step 3 Click **Retrieve**.

Step 4 Click **Add** to add a new dialer.

Step 5 Enter the required information on the **Dialer General Tab**. See the *Configuration Manager Online Help* for details of these fields.

Step 6 Click **Save**.

Step 7 Select the **Port Map Selection** tab to display the port map configuration. See the *Configuration Manager Online Help* for details of configuring these mappings.

Step 8 Click **Add**

Step 9 Configure a set of ports and their associated extensions.

A Dialer can support 3000 ports. The allowed **Telephony Port** range is from 0 to 2999.

Step 10 Click **OK**. The port mappings appear on the **Port Map Selection** tab.

Step 11 Click **Save** to save all the configuration information.

Configure System Options

Step 1 In **Unified CCE Administration**, choose **Organization** > **Campaigns** to open the Campaigns page.

Step 2 Define the dialing time range to use for all your Outbound Option campaigns.

For more information on campaign configuration, refer the **Manage Campaigns** section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Enable Expanded Call Context Variables

Perform the following steps to enable the expanded call context variables.

Step 1 In **Unified CCE Administration**, click **Overview** > **Call Settings** > **Route Settings** > **Expanded Call Variables**.

Step 2 Enable all BAxxxx variables (BAAccountNumber, BABuddyName, BACampaign, BADialedListID, BAResponse, BAStatus, and BATimezone).

What to do next

By default, the solution includes the predefined BAxxxx ECC variables in the "Default" ECC payload. You can also create a custom ECC payload for your Outbound Option call flows. Always remember that you cannot use an ECC variable unless it exists in one of the ECC payloads that you use for a call flow.

Packet Capture for Troubleshooting

For the SIP Dialer to capture data, ensure that the dialer on the Unified CCE PG machine uses the active interface from the Ethernet Interface list. You can determine the active interface with a network protocol analyzer tool such as Wireshark, which you can download from www.wireshark.org. The interface with network packets is the active interface.

You can change the SIP Dialer packet capture parameters to use the active interface from the Windows Registry Editor. Change the interface name option (-i) in the `CaptureOptions` key to the number of the active interface. For example, to use the third interface, set the value for -i to `-i 3`.

Capture files are in the `HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\ICM\`<*customer instance*>`\Dialer` registry key location.

Unified Communications Manager and Gateway Configuration

In the next phase of Outbound Option installation, you set up Unified Communications Manager and its related gateway.

The following table lists the steps that comprise Unified CM setup.

Table 13: Unified CM Configuration Steps for Deployments with SIP Dialer

Step Number	Procedure
1	Disable Ringback During Transfer to Agent for SIP, on page 149
2	Configuration of Voice Gateways, on page 151
3	Configure SIP Trunks, on page 154

Disable Ringback During Transfer to Agent for SIP

The voice gateway generates a ringback tone to the customer. To prevent the gateway from generating a ringback, apply a SIP normalization script to the Unified Communications Manager SIP trunk.

Apply this SIP normalization script only to the SIP trunk that handles the inbound call from the voice gateway for agent transfer.

- If your deployment uses the same gateway for both PSTN calls and the dialer, complete all steps, 1 to 13, to create a dedicated SIP trunk and apply the normalization script.

Note The trunk for PSTN calls still needs a 180 RINGING SIP message for inbound calls to trigger the gateway to play ringback to the PSTN.

For more information, see the TechNote *Disable Ringback During Transfer to Agent for SIP* at https://www.cisco.com/c/en/us/support/docs/customer-collaboration/packaged-contact-center-enterprise/200323-Cisco-Packaged-Contact-Center-Enterprise.html.

- If your deployment has a dedicated SIP trunk to handle the agent transfer dialer, complete steps 1 to 2 and 8 to 13 to apply the normalization script to your SIP trunk.

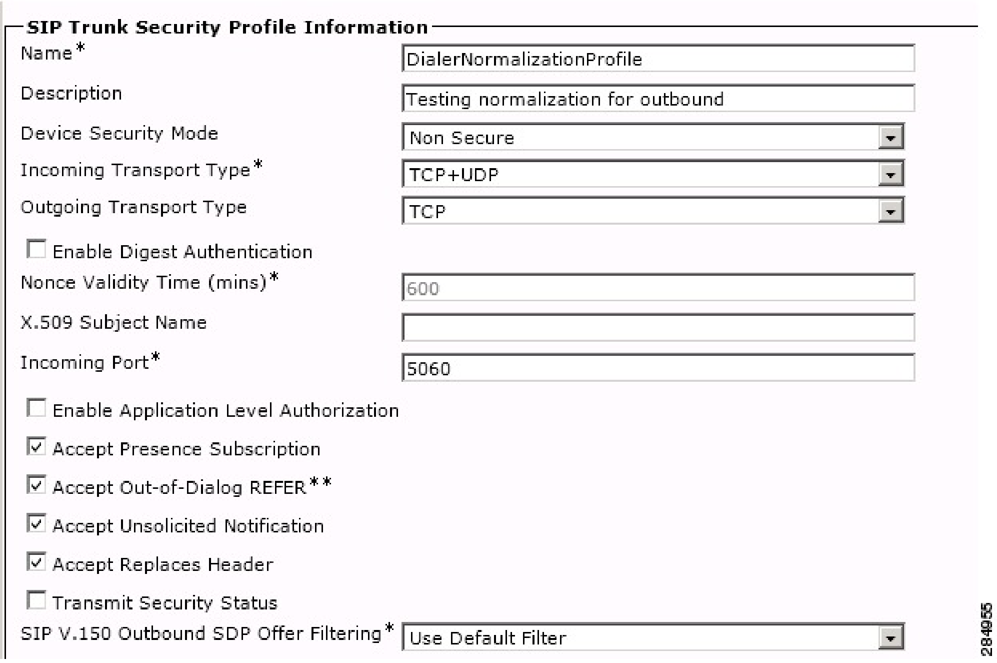
Step 1 Navigate to `https://<IP_address>:8443` where `<IP_address>` identifies the Unified Communications Manager server.

Step 2 Sign in to Unified Communications Manager.

Step 3 To create a SIP trunk security profile in Unified Communications Manager, select **Communications Manager GUI** > **System** > **Security** > **SIP Trunk Security Profile** > **[Add New]**.

The default port is 5060.

Figure 10: SIP Security Profile



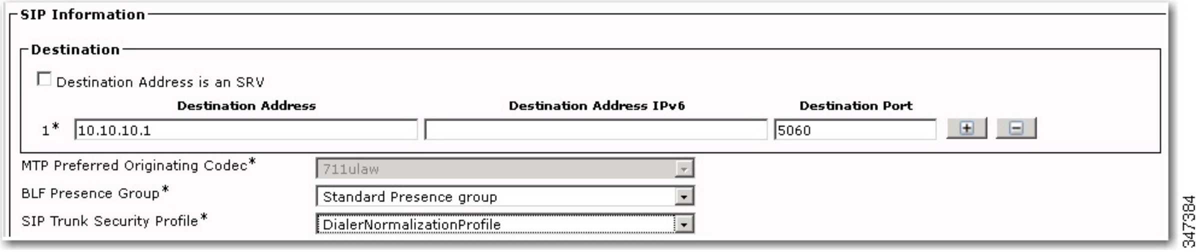
SIP Trunk Security Profile Information

Name*	DialerNormalizationProfile
Description	Testing normalization for outbound
Device Security Mode	Non Secure
Incoming Transport Type*	TCP+UDP
Outgoing Transport Type	TCP
□ Enable Digest Authentication	
Nonce Validity Time (mins)*	600
X.509 Subject Name	
Incoming Port*	5060
□ Enable Application Level Authorization	
☑ Accept Presence Subscription	
☑ Accept Out-of-Dialog REFER**	
☑ Accept Unsolicited Notification	
☑ Accept Replaces Header	
□ Transmit Security Status	
SIP V.150 Outbound SDP Offer Filtering*	Use Default Filter

Step 4 Click **Save**.

Step 5 Create a new SIP trunk and add the new SIP Trunk Security Profile.

Figure 11: Create a New SIP Trunk



SIP Information

Destination

□ Destination Address is an SRV

	Destination Address	Destination Address IPv6	Destination Port
1*	10.10.10.1		5060

MTP Preferred Originating Codec*	711ulaw
BLF Presence Group*	Standard Presence group
SIP Trunk Security Profile*	DialerNormalizationProfile

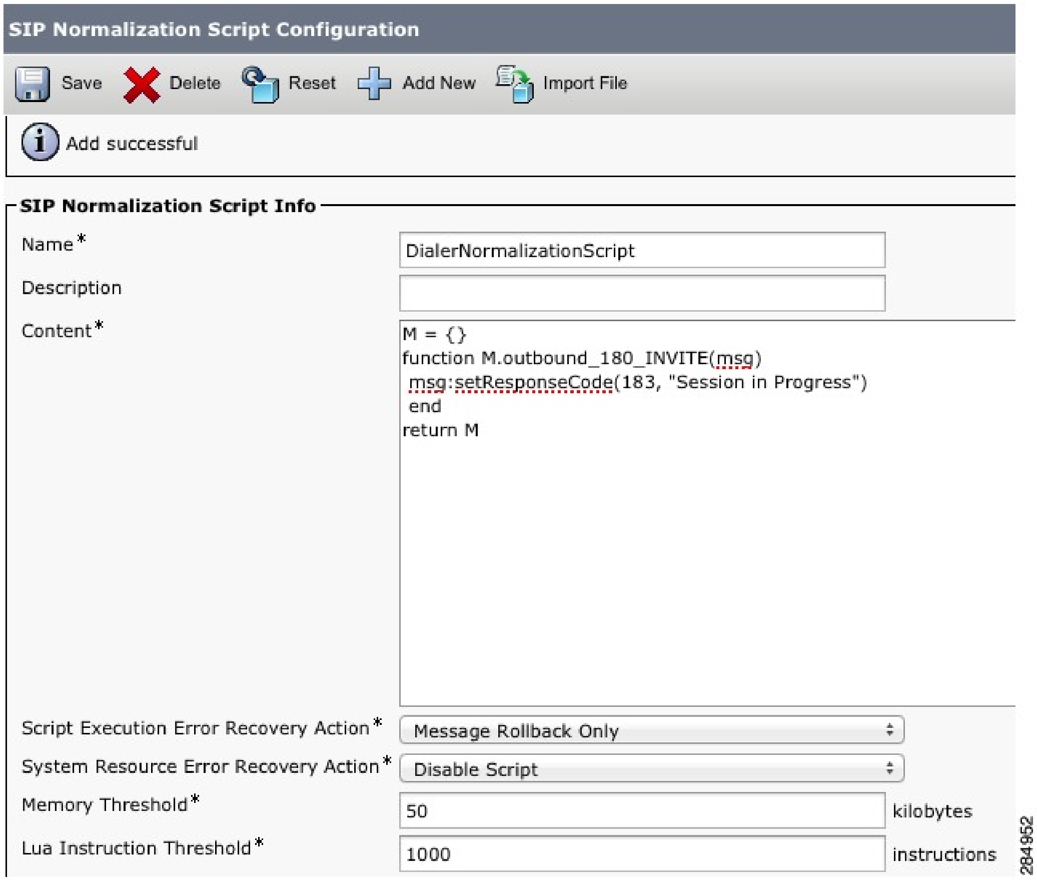
Step 6 Click **Save**.

Step 7 Click **Reset**.

Step 8 In **Communications Manager GUI** > **Devices** > **Device Settings** > **SIP Normalization Scripts** > **[Create New]**, enter the following SIP normalization script into the content field. All other values remain set to default.

```
M = {}
function M.outbound_180_INVITE(msg)
msg:setResponseCode(183, "Session in Progress")
end
return M
```

Figure 12: Add Normalization Script



SIP Normalization Script Configuration

Save | Delete | Reset | Add New | Import File

Add successful

SIP Normalization Script Info

Name*: DialerNormalizationScript

Description:

Content*:

```
M = {}
function M.outbound_180_INVITE(msg)
 msg:setResponseCode(183, "Session in Progress")
 end
return M
```

Script Execution Error Recovery Action*: Message Rollback Only

System Resource Error Recovery Action*: Disable Script

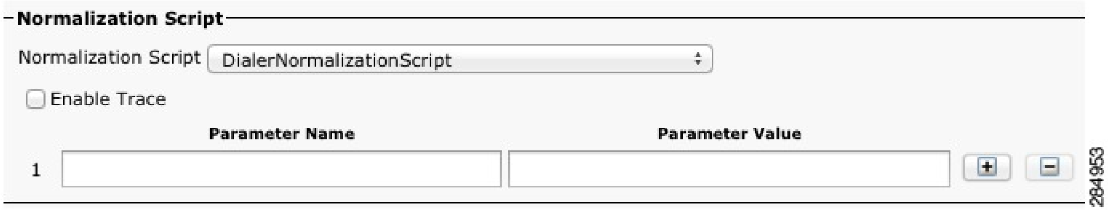
Memory Threshold*: 50 kilobytes

Lua Instruction Threshold*: 1000 instructions

Step 9 Click **Save**.

Step 10 Associate the new normalization script with the SIP trunk.

Figure 13: Associate Script with Trunk



Normalization Script

Normalization Script: DialerNormalizationScript

□ Enable Trace

	Parameter Name	Parameter Value
1		

Step 11 Click **Save**.

Step 12 Click **Reset**.

Configuration of Voice Gateways

Telecom carriers sometimes send an ISDN alerting message without a progress indicator. This situation causes the voice gateway to send a SIP 180 Ringing message, instead of a SIP 183 Session In Progress message, to the SIP dialer. The SIP dialer can process provisional messages such as 180, 181, 182, and 183 with or without Session Description Protocol (SDP). When the SIP dialer receives these provisional messages without SDP, the dialer does not perform Call Progress Analysis (CPA) and the Record CPA feature is disabled.

To enable the SIP dialer to perform CPA, add the following configuration to the POTS dial-peer of the voice gateway: `"progress_ind alert enable 8"`. This code sends a SIP 183 message to the SIP dialer.

Telecom carriers sometimes send an ISDN alerting message without a progress indicator. This situation causes the voice gateway to send a SIP 180 Ringing message, instead of a SIP 183 Session In Progress message, to the SIP dialer. The SIP dialer can process provisional messages such as 180, 181, 182, and 183 with or without Session Description Protocol (SDP). The SIP Dialer processes the CPA information along with the SDP information because the SDP information is part of these provisional messages. But if the dialer receives these provisional messages without SDP, the dialer does not perform Call Progress Analysis (CPA) and the Record CPA feature is disabled. If the next provisional message changes the SDP information, the dialer processes the SDP information.

Enable 100rel for Outbound Option. Otherwise, Outbound calls from the SIP Dialer fail. The following two sections provide examples of voice gateway configuration from the command line.

Configure Rel1xx Supported for Dial-Peer for the SIP Dialer

The following example shows how to enable rel1xx on a voice dial-peer for the SIP dialer. It uses 8989 for the tag of the voice dial-peer.

```
GW(config)#config t
GW(config-dial-peer)#dial-peer voice 8989 voip
GW(config-dial-peer)#voice-class sip rel1xx supported 100rel
GW(config-dial-peer)#exit
GW(config)#exit
GW#wr
```

This short procedure results in the following dial-peer configuration. (Only the bolded line is relevant to this discussion.)

```
dial-peer voice 8989 voip
incoming called-number 978T
voice-class sip rel1xx supported "100rel"
dtmf-relay rtp-nte h245-signal h245-alphanumeric
codec g711ulaw
```

Configure Outgoing Dial-Peer for a Dialing Customer

The following example shows how to configure an outgoing dial-peer for a dialing customer.

```
GW(config)#config t
GW(config-dial-peer)#dial-peer voice 97810 voip
GW(config-dial-peer)#destination-pattern 97810[1-9]
GW(config-dial-peer)#port 1/0:23
GW(config-dial-peer)#forward-digits all
GW(config-dial-peer)#exit
GW(config)#exit
GW#wr
```

This short procedure results in the following dial-peer configuration for a dialing customer.

```
dial-peer voice 97810 pots
destination-pattern 97810[1-9]
port 1/0:23
forward-digits all
```

Configure Rel1xx Disable for Unified CVP Voice Dial-Peer

The following example shows how to disable rel1xx for a Unified CVP voice dial-peer. It uses 8989 for the tag of the voice dial-peer.

```
GW(config)#config t
GW(config-dial-peer)#dial-peer voice 8989 voip
GW(config-dial-peer)#voice-class sip rel1xx disable
GW(config-dial-peer)#exit
GW(config)#exit
GW#wr
```

This short procedure results in the following dial-peer configuration. (Only the bolded line is relevant to this discussion.)

```
dial-peer voice 8989 voip
description CVP SIP ringtone dial-peer
service ringtone
incoming called-number 9191T
voice-class sip rel1xx disable
dtmf-relay rtp-nte h245-signal h245-alphanumeric
codec g711ulaw
no vad
```

Configure an Outgoing Dial-Peer for Transferring Call to Agent

The following example shows the outgoing dial-peer configuration for transferring calls to agents.

```
dial-peer voice 11000 voip
 destination-pattern 11T
 session protocol sipv2
 session target ipv4:10.10.10.31(this is Call Manager's IP address)
 voice-class codec 1
 voice-class sip rel1xx supported "100rel"
 dtmf-relay rtp-nte h245-signal h245-alphanumeric
 no vad
```

Note In a SIP Dialer with Unified CVP VRU deployment, dialer-related call flows do not invoke call-survivability scripts that are enabled on an incoming POTS dial-peer in the Ingress gateway. However, enabling a call-survivability script on an Inbound POTS dial-peer does not negatively affect dialer-related call flows.

Configure Transcoding Profile for Cisco Unified Border Element

The following example shows the transcoding profile for Cisco UBE.

Note Transcoding impacts port density.

```
dspfarm profile 4 transcode universal
     codec g729r8
     codec g711ulaw
     codec g711alaw
     codec g729ar8
     codec g729abr8
     maximum sessions 250
     associate application CUBE
     !
```

Configure Cisco Unified Border Element

While configuring Cisco UBE, ensure that you:

- Configure the three dial-peers in the Cisco UBE.

  The dial-peers are used for:

  - Incoming calls from the dialer.
  - Outgoing calls to the terminating network from the Cisco UBE.
  - Calls to be routed to the Cisco Unified Communications Manager.

- Issue the following commands globally to configure the Cisco UBE:

  - **no supplementary-service sip refer**
  - **supplementary-service media-renegotiate**

Note Virtual CUBE does not support CPA. Use a dedicated physical gateway if your solution needs CPA.

Configure SIP Trunks

Unified CM is connected to the voice gateway by SIP Trunks, which you configure on Unified CM. Set up route patterns for the Dialer which are appropriate for your dial pattern.

See the *System Configuration Guide for Cisco Unified Communications Manager* at https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/products-installation-and-configuration-guides-list.html for instructions on how to configure SIP trunks. For information about logging in to Ingress or VXML gateways, refer to the sections on configuring gateways for Courtesy Callback 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.

Configure a SIP trunk on Unified CM from Unified CM to the voice gateway. Specify the IP address of the voice gateway in the **Destination** field.

Configure E1 R2 Signaling

The Outbound Option Dialer may be configured with systems using the E1 R2 signaling protocol. E1 R2 signaling is a channel associated signaling (CAS) international standard that is used with E1 networks in Europe, Latin America, Australia, and Asia. For more information, see E1 R2 Signaling Theory

The high-level procedure for configuring an E1 R2 controller for use with the Outbound dialer is summarized below. For full configuration details, see E1 R2 Signaling Configuration and Troubleshooting.

Step 1 Set up an E1 controller connected to the private automatic branch exchange (PBX) or switch. Ensure that the framing and linecoding of the E1 are properly set for your environment.

Step 2 For E1 framing, choose either **CRC** or **non-CRC**.

Step 3 For E1 linecoding, choose either **HDB3** or **AMI**.

Step 4 For the E1 clock source, choose either **internal** or **line**. Keep in mind that different PBX's may have different requirements for their clock source.

Step 5 Configure line signaling.

Step 6 Configure interregister signaling.

Step 7 Customize the configuration with the **cas-custom** command.

Example E1 R2 Settings

```
controller E1 0/0/0
  framing NO-CRC4
  ds0-group 1 timeslots 1-15,17-31 type r2-digital r2-compelled ani
  cas-custom 1
  country telmex
  category 2
  answer-signal group-b 1
  caller-digits 4
  dnis-digits min 4 max 13
  dnis-complete
  timer interdigit incoming 1000
  groupa-callerid-end
```

Outbound Option Software Installation Steps

This section discusses the tasks that are associated with installing Outbound Option and related components. Before proceeding, navigate to the Side A Unified CCE AW-HDS-DDS and stop all ICM services there. Then perform the steps in the following sections.

Software and Database Creation

In the next phase, you install the Outbound Option component software and create its database. The following table lists the steps that comprise software installation and database creation.

Table 14: Software Installation and Database Creation Steps

Step	Procedure
1	Configure the Logger for Outbound Option, on page 158
2	Create Outbound Option Database, on page 157
3	Add MR PIM for Outbound, on page 160
4	Install Dialer Component on the PG Virtual Machine, on page 160

Outbound Option Database

Outbound Option uses a dedicated SQL database on the Logger. The installation includes creating this database. The installer collects some business-related data to properly create the database.

If you enable Outbound Option High Availability, ensure that the Logger virtual machine datastore is large enough to accommodate both the Logger database and the Outbound Option database on Logger Side A and Logger Side B.

Consider these guidelines when determining minimum drive size:

- For a 4000 (or smaller) Agent Reference Design solution, add an extra 500 GB of disk space for the Outbound Option database.
- For a 12000 Agent Reference Design solution, add an extra 1 TB of disk space for the Outbound Option database.

Note When using Outbound Option High Availability on a Packaged CCE system, the maximum number of records that can be imported without adding any extra disk space to the Rogger VM is one million.

Outbound Option for High Availability: Preliminary Two-Way Replication Requirements

If you plan to set up Outbound Option for High Availability two-way replication, there are several preliminary requirements.

Create an Outbound Option Database on Logger Side A and Side B

If you have enabled Outbound Option on Logger Side A in a previous release, you must:

- Stop all Logger services on Logger Side A.
- Perform a full database backup for the Outbound Option database on Logger Side A and restore it to Logger Side B. Use SQL Server Management Studio (SSMS) to complete this task.

If you have not enabled Outbound Option in a previous release, you must create an Outbound Option database on Logger Side A and Logger Side B. Use the ICMDBA utility to complete this task.

Note If the database replication fails and it is resolved, the Outbound Option HA must be enabled again. In such a case, you must again synchronize the databases on the Active and Standby sides. Perform a full database backup for the Outbound Option database on Active side and restore it to the Standby side.

Define Logger Public Interface Hostname on Logger Side A and Logger Side B

As you configure Outbound Option for High Availability, you must define the Logger Public Interface Hostname on both sides of the Logger. IP addresses are not allowed.

Make Campaign Manager and Dialer Registry Setting Customizations on Both Side A and Side B

If you customize any Campaign Manager and Dialer registry settings on one side, you must make the same updates for the registry settings on the other side.

Stop the Logger Service Before Enabling or Disabling Outbound Option High Availability

Before you enable or disable Outbound Option High Availability, stop the Logger service on the applicable side or sides.

Create Outbound Option Database

Before you use Outbound Option, estimate the size for the Outbound Option database.

Step 1 Collect the following information:

- The size, in bytes, of each customer record in the import file. If the size is less than 128 bytes, use 128. (RecordSize)
- The number of records that are imported. (RecordCount)
- Do the records from new imports replace or append to records that are already in the database?

Step 2 Estimate the contact table size as follows:

- If imports overwrite existing records: Do not change record count.
- If imports append to existing records: RecordCount = total number of rows kept in a customer table at a time.
- contact-table-size = RecordSize * RecordCount * 1.18

Step 3 Estimate the dialing list table size as follows:

- If imports overwrite existing records: RecordCount = number of rows imported * 1.5. (50% more rows are inserted into the dialing list than are imported.)
- If imports append existing records: RecordCount = total number of rows kept in customer table at a time * 1.5
- dialing-list-table-size = rows in dialing list * 128 bytes * 4.63

Step 4 Calculate the database size using this formula:

```
(Number of rows in all DL tables * (size of one row + size of index) ) +
(Number of rows in personal call back table * (size of one row + size of index) ) +
(Number of rows in Contact List table * (size of one row + size of index))
```

Step 5 Start ICMDBA by entering **ICMDBA** in the Microsoft Windows **Run** dialog box or command window.

Step 6 Select the **Logger**. Then, select **Database** > **Create**.

Step 7 In the **Create Database** window, specify the Outbound Option database type.

Step 8 Click **Add**. The **Add Device** window appears.

Use this window to create a new data device and log device for the Outbound Option database. Specify the disk drive letter and size in megabytes for each new device.

Step 9 Click **OK** to create the device.

Step 10 Click **Create**, and then click **Start**.

Step 11 Click **Close**.

If necessary, you can later edit the device to change storage size, or remove a device, using the **Database** > **Expand** option.

⚠

Caution You cannot make manual changes to the contents of the Outbound Option database. Do not use triggers in the Outbound Option database. Do not add or modify triggers for the dialing lists or personal callback list. The Dialer_Detail table in the logger or HDS contains the information that custom applications require. Extract that information from the historical database server (HDS) to a separate server where the custom application can process the data without impacting the HDS.

Note If you have used the ICMDBA tool to create an Outbound Option database on Side B of Unified CCE Rogger and you later uninstall Packaged CCE, you can manually delete the database after the uninstallation by using SQL Server Management Studio (SSMS).

Note When you use the **Append** option to import records to the **Outbound Contact Table**, the size of the Blended Agent (BA) database keeps increasing and it occupies all the available space in the disk. Hence, you must manually purge the **Outbound Contact Table** to create more space on the disk.

What to do next

You must enable autogrowth on the Outbound Option database. For details, refer to the section about verifying database configuration in the *Outbound Option Guide for Unified Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-user-guide-list.html.

Configure the Logger for Outbound Option

Use this procedure to configure the Logger for Outbound Option.

You can (optionally) configure the Logger to enable Outbound Option and Outbound Option High Availability. Outbound Option High Availability facilitates two-way replication between the Outbound Option database on Logger Side A and the Outbound Option database on Logger Side B. Use the ICMDBA tool to create an outbound database on Side A and Side B; then set up the replication by using Web Setup.

Perform the following procedure on both the Side A and Side B Loggers to configure Outbound Option or Outbound Option High Availability. Both Logger machines must be up and operational.

Important Before you configure the Logger for Outbound Option High Availability:

- Confirm that an Outbound Option database exists on Logger Side A and Logger Side B.

Step 1 Open the Web Setup tool.

Step 2 Choose **Component Management** > **Loggers**.

Step 3 Choose the Logger that you want to configure, and click **Edit**.

Step 4 Click **Next** twice.

Step 5 On the Additional Options page, click the **Enable Outbound Option** check box.

Step 6 Click the **Enable High Availability** check box to enable Outbound Option High Availability on the Logger. Checking this check box enables High Availability two-way replication between the Outbound Option database on Logger Side A and the Outbound Option database on Logger Side B. Two-way replication requires that you check this check box on the Additional Options page for both Logger Side A and Side B. If you disable two-way replication on one side, you must also disable it on the other side.

You must enable Outbound Option in order to enable Outbound Option High Availability. Similarly, if you have enabled High Availability, you must disable High Availability (uncheck the **Enable High Availability** check box) before you can disable Outbound Option (uncheck the **Enable Outbound Option** check box).

Step 7 If you enable High Availability, enter a valid public server hostname address for **Logger Side A** and **Logger Side B**. Entering a server IP address instead of a server name is not allowed.

Step 8 If you enable High Availability, enter the **Active Directory Account Name** that the opposite side Logger runs under or a security group that includes that account.

Note While using Outbound Option High Availability, if you want to change the **Logger Public Interface** or **Active Directory Account Name**, you must disable Outbound Option High Availability using logger setup. Only after disabling Outbound Option HA, change the **Logger Public Interfaces** or **Active Directory Account Name**, then re-enable Outbound Option High Availability to update the new **Logger Public Interface** or **Active Directory Account Name**.

Step 9 Select the **Syslog** box to enable the Syslog event feed process (cw2kfeed.exe).

Note The event feed is processed and sent to the Syslog collector only if the Syslog collector is configured. For more information about the Syslog event feed process, see the *Serviceability 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 10 Click **Next**.

Step 11 Review the Summary page, and click **Finish**.

Additional Two-Way Outbound Option Database Replication Consideration

Keep in mind the following consideration when setting up two-way replication.

Import to Active Side

Importing a local file succeeds only if you import it to the active side. To avoid having to identify which side is active, you can use any of the following methods:

- Create a Microsoft Windows file share that is accessible to both sides with the same mapping; for example, `//<machine_name>/drive/file`, viewable from both sides.
- Use Microsoft Windows Distributed File System (DFS). With DFS, you can set up a local drive that DFS updates for you. DFS also makes sure that operations are replicated. For more information, see your Microsoft documentation.
- For campaigns created by using the Outbound API, you can use the Import API to import contacts without identifying the active side. For more information, see the *Cisco Packaged Contact Center Enterprise Developer Reference* at https://developer.cisco.com/site/packaged-contact-center/documentation/.

Add MR PIM for Outbound

Step 1 Access the Unified CCE PG on Side A.

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

Step 3 In the Instance Components panel of the **Components Setup** screen, select the **PG2A Instance** component for Side A. (Select **PG2B** for Side B.) Then click **Edit**.

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

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

Step 6 From the **Peripheral Gateway Component Properties** screen, click **Add** and select **PIM2**.

Note Select **PIM2** and **peripheralID 5003** even if you are not using PIM1 for another machine.

Step 7 Configure with the client type of Media Routing as follows:

a) Check **Enabled**.
b) Enter `MR2` or a name of your choice for the **Peripheral name**.
c) Enter `5003` for the **Peripheral ID**.
d) Enter `Unified CCE PG Side A IP` (Side B IP for **PG2B**) for the **Application Hostname(1)**.
e) Retain the default value for the **Application Connection port (1)**.
f) Enter `Unified CCE PG Side B IP` (Side A IP for **PG2B**) for the **Application Hostname(2)**.
g) Retain the default value for the **Application Connection port (2)**.
h) Enter `5` for the **Heartbeat interval (sec)**.
i) Enter `10` for the **Reconnect interval (sec)**.
j) 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.

k) Click **OK**.

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. Then click **Finish**.

Step 10 Click **Exit Setup**.

Step 11 Repeat from Step 1 for the Unified CCE PG on Side B.

Install Dialer Component on the PG Virtual Machine

Step 1 Stop all Packaged CCE Services.

Step 2 On both the Side A and Side B PGs, run Peripheral Gateway Setup. Select **Start** > **All Programs** > **Cisco Unified CCE Tools** > **Peripheral Gateway Setup**.

Step 3 In the **Cisco Unified ICM/Contact Center Enterprise & Hosted Components Setup** dialog, select an instance from the left column under **Instances**.

Step 4 Click **Add** in the **Instance Components** section.

The **ICM Component Selection** dialog opens.

Step 5 Click **Outbound Option Dialer**.

The **Outbound Option Dialer Properties** dialog opens.

Step 6 Check **Production mode** and **Auto start at system startup**. These options set the Dialer Service startup type to Automatic, so the dialer starts automatically when the machine starts up.

The **SIP (Session Initiation Protocol)** Dialer Type is automatically selected.

Step 7 Click **Next**.

Step 8 Supply the following information on this page:

- In the **SIP Dialer Name** field, enter the name of the SIP dialer. For example, `Dialer_for_Premium_Calling_List`. There's a 32-character limit. The name entered here must match the name that is configured in Configuration Manager.
- For **SIP Server Type**, select Cisco voice gateway .
- In the **SIP Server** field, enter the hostname or IP address of the Cisco voice gateway.

  Note The **SIP Server** hostnames are restricted to a maximum of 16 characters.
- In the **SIP Server Port** field, enter the port number of the SIP Server port. Default is 5060.

Click **Next**.

Step 9 On the **Outbound Option Dialer Properties** dialog, specify the following information:

- **Campaign Manager server**—The hostname or IP address of the Outbound Option server (the hostname or IP address of Unified CCE Rogger Side A) in Packaged CCE.
- **Campaign Manager server A**—If the Campaign Manager is set up as duplex, enter the hostname or IP address of the machine where the Side A Campaign Manager is located. If the Campaign Manager is set up as simplex, enter the same hostname or IP address in this field and the **Campaign Manager server B** field. You must supply a value in this field.
- **Campaign Manager server B**—If the Campaign Manager is set up as duplex, enter the hostname or IP address of the machine where the Side B Campaign Manager is located. If the Campaign Manager is set up as simplex, enter the same hostname or IP address in this field and the **Campaign Manager server A** field. You must supply a value in this field.

  In simplex mode, make sure not to provide same port number in server Side B and Side A. For more information about port ranges, see the *Port Utilization Guide*.
- **Enable Secured Connection**— Allows you to establish secured connection between the following:
  - CTI server and dialer
  - MR PIM and dialer

  Check the **Enable Secured Connection** check box to enable secured connection.

  Note Before you enable secured connection between the components, ensure to complete the security certificate management process.

For more information, see the *Security 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.

- **CTI server A**—The hostname or IP address of Unified CCE PG Side A.
- **CTI server port A**—The port number that the dialer uses to create an interface with CTI server-Side A. The default is 42027.
- **CTI server B**—The hostname or IP address of Unified CCE PG Side B.
- **CTI server port B**—The port number that the dialer uses to create an interface with CTI server-Side B. The default is 43027.
- **Heart beat**—The interval between dialer checks for the connection to the CTI server, in milliseconds. The default value is 500.
- **Media routing port**—The port number that the dialer uses to create an interface with the Media Routing PIM on the Media Routing PG. The default is 38001. Make sure the Media routing port matches that of the MR PG configuration.

Step 10 Click **Next**. A **Summary** screen appears.

Step 11 Click **Next** to begin the dialer installation.

Optional - Edit Dialer Registry Value for Auto-Answer

If you enable auto answer in the CallManager with a zip tone, you must disable auto answer in the Dialer or Dialers, if there are more than one. A zip tone is a tone sent to the agent's phone to signal that a customer is about to be connected.

To disable auto answer in the Dialer, after the Dialer process runs for the first time, change the value of the following registry key to 0:

HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\ICM\<*instance_name*>
\Dialer\AutoAnswerCall

Auto Answer Configuration on Agent Phones

The dialer component is preconfigured during installation to auto answer Outbound Option related calls to the Outbound Option agent. However, this default configuration does not provide a zip tone to the agent (which notifies of incoming calls), so agents must monitor the agent application for incoming customer calls.

To enable zip tone, enable auto-answer on the agent’s phone configuration in Unified CM. This solution adds about a second onto the transfer time. This solution is identical to the solution that is used for Unified CCE.

For Mobile Agents using the nailed connection, the Unified CM auto answer setting does not provide a zip tone, but contact center enterprise does provide an option for playing a notification tone to the agent using the agent desk settings.

Enabling auto answer in the agent desk settings or in the dialer component in conjunction with the Unified CM can be problematic. Therefore, disable the auto answer option in the dialer component, and enable it either in the agent desk settings or in Unified CM.

Verify connections

The Diagnostic Framework Portico provides details about the health of the installation even before any campaign configuration is initiated or before any call is placed. The interface contains the following details about the dialer status.

Step 1 Navigate to the Outbound Option Dialer component in the Diagnostic Framework Portico.

The Node Name is Dialer. The Process name is BADialer.

Step 2 Verify that the Campaign Manager (CM) has a status of Active (A).

Step 3 Verify that the CTI Server (CTI) has a status of Active (A).

Step 4 Verify that the number of Configured Ports equals the number of Ready Ports.

Step 5 Verify that the MR has a status of Active (A).

Maintenance Considerations

This section contains information about maintaining the Outbound Option application.

SIP Dialer Voice Gateway Over-capacity Errors

If your network monitoring tool receives an alarm in the SIP dialer about being over capacity, you can ignore the alarm unless it becomes an ongoing issue. This section describes the source of the alarm and remedial actions associated.

If the Voice Gateway in a SIP dialer implementation is over capacity, the SIP Dialer receives the following message: `SIP 503 messages if the SIP Dialer is deployed with Voice Gateway only`

If the percentage of SIP 503 messages reaches 1% of all messages, the SIP dialer raises an alarm.

Use one of the following measures to attempt to remedy the problem if Voice Gateway capacity becomes an ongoing issue:

- Check the Voice Gateway configuration. If there are errors, fix them and reset Port Throttle to its original value. Port Throttle (the calls-per-second rate at which the dialer dials outbound calls) is set on the Dialer General tab in the Configuration Manager.
- Check the sizing information. Adjust the value of Port Throttle according to the documented guidelines.
- Enable the auto-throttle mechanism by setting the Dialer registry setting **EnableThrottleDown** to 1.

  To set **EnableThrottleDown**, open the Registry Editor (regedit.exe) on the PG machine and navigate to `HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\ICM\<customerinstance>\Dialer`.

The SIP dialer performs an automatic throttle down when the percentage of SIP 503 messages reaches 2% of all messages, if the auto-throttle mechanism is enabled. This throttle down means that the SIP dialer decreases the configured value of Port Throttle by approximately 10%.

If one throttle down does not correct the problem, the SIP dialer performs more throttle downs until either the problem is corrected or the value of Port Throttle is throttled down to 50% of the originally configured value.

For each automatic throttle down, alarm and trace messages clearly provide detailed information about the adjusted port throttle value, configured port throttle value, and time duration.

Even after the problem is corrected, the dialer does not automatically throttle back to the configured value. To increase the throttle back to the configured value, run the **updateportthrottle /portthrottle <configured value>** command using the process monitoring tool Procmon.

Update the North American Numbering Plan Data

The Regional Prefix Update Tool (RPUT) is used to update the Packaged CCE database to the latest North American Local Exchange NPA NXX Database (NALENND).

You can use this tool only if Packaged CCE is using the North American Numbering Plan.

The RPUT is composed of the following two files (installed in the `ICM\bin` directory on the Unified CCE AW-HDS-DDS server):

- region_prefix_data.txt (or the <DatafileName>)

  Contains the data this tool uses to update the region prefix table in the Packaged CCE database. Note that you should change paths to the `ICM\bin` directory.

- regionfix.exe

  This executable reads the region_prefix_data.txt data file and updates the region prefix table.

The RPUT is run from the command line as described in the following procedure.

Step 1 Open a command prompt (Select **Start** > **Run**, and enter **cmd**, then click **OK**).

Step 2 Change the path to `ICM\bin`.

Step 3 Enter the following at the prompt: `regionfix.exe <DatafileName>` (where *<DatafileName>* is the name of the data file).

The Regional Prefix Update Tool then shows the version of the input data file and asks if you want to proceed. If you proceed, the tool connects to the Packaged CCE database. The number of records that are to be updated, deleted, and inserted appear. These records are put into three different files:

- region_prefix_update.txt (which includes preserved Custom Region Prefixes)
- region_prefix_new.txt
- region_prefix_delete.txt

Step 4 You can either delete or retain the entries present in the region_prefix_delete.txt file while performing the insertions and updates. To retain the entries, type **No** when the tool prompts you to delete the entries. Type **Yes** to delete the entries.

Step 5 Check the contents of the files before proceeding.

Step 6 Answer **Yes** to proceed with the update.

When the update is complete, the tool displays the following message:

```
Your region prefix table has been successfully updated.
```

Administration and Usage

Campaign configuration

Campaign Task List

The following table lists the steps that are required to create both an agent and IVR campaign, and the location of the instructions for the task.

Table 15: Steps for Creating a Campaign

Step Number	Task	Where Discussed
1	Create one or more skill groups for the campaign.	Configure Skill Group, on page 165
2	Configure the call type using the Packaged CCE Call Type gadget.	Create a Call Type, on page 166
3	Create a dialed number on the MR client using the Packaged CCE Dialed Number gadget. This dialed number is for agent reservation.	Configure Dialed Numbers, on page 166
4	Create DN for Abandon to IVR on the MR PG for the SIP dialer.	Configure Dialed Numbers, on page 166
5	Create DN for AMD to IVR on the MR PG for the SIP dialer.	Configure Dialed Numbers, on page 166
6	Configure a campaign using the Outbound Option Campaign tool.	Create a Campaign, on page 166
7	Configure a routing script using the Script Editor.	Set Up Routing Scripts, on page 173
8	Configure an administrative script using the Script Editor.	Set Up Administrative Scripts, on page 169
9	Voice gateway configuration.	Voice Gateway and Unified CVP Configuration for a VRU Campaign, on page 167

Configure Skill Group

Add at least one skill group. For an agent campaign, add at least one agent to the skill group. Log the agent in to the skill group, and make the agent ready for the agent campaign. You do not need to add an agent for a VRU campaign skill group. For information about configuring skill groups, see the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Create a Call Type

The dialed numbers and routing scripts that you will create will reference *call types*, so you should create them as needed. For information about creating call types, see the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html. For example, you can create one call type for an agent campaign and another for a VRU campaign. You need to associate the call types with the dialed numbers you created earlier.

Configure Dialed Numbers

Configure at least two dialed numbers on the outbound routing client: one for the agent campaign and one for the VRU campaign. See the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html for information about configuring dialed numbers.

Create a Campaign

The Campaign feature in the Packaged CCE webadmin is used to add, delete, and modify Outbound Option campaigns.

Before you create a campaign, first configure the following information:

- At least one skill group
- The following dialed numbers with Routing Type set to Outbound Voice:
  - One for accessing the agent reservation script (not required for transfer to VRU campaigns).
  - One for transferring the call to the VRU for abandon treatment when no agents are available. This number must be different from the previous number.
  - One for transferring the call to the VRU for answering machine detection (AMD) or transfer to VRU campaign treatment. This number can be the same as the previous number, but different from the first number.

For more information, refer the **Manage Campaign** section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Notes on Editing a Campaign in Progress

You can edit most campaign configuration settings while a campaign is running. The changes take effect with new calls after the setting has been changed. However, do not edit the following settings while a campaign is in progress:

- Do not modify the **Maximum Attempts**value. Modifying this value while a campaign is in progress can cause a long delay in record retrieval and longer agent idle times.
- Do not delete a skill group while a campaign is in progress.

(Optional) Configure Personal Callbacks

PersonalizedCallback is an optional feature in Outbound Option. Personal Callback enables an agent to schedule a callback to a customer for a specific date and time. A personal callback connects the agent who originally spoke to the customer back to the customer at the customer-requested time.

When you create campaigns, you enable the callback feature individually for each campaign.

For more information, refer the **Manage Campaign** section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Voice Gateway and Unified CVP Configuration for a VRU Campaign

For a VRU campaign, you need to configure a dial-peer in the voice gateway. This dial peer is used to instruct the voice gateway to transfer the call to Unified CVP. It must match the Network VRU label that is configured on the MR routing client with type 10 Unified CVP network VRU.

In base configuration, this label is preconfigured with default value 66611110000. Follow the steps in this example.

Step 1 Add a dial-peer to match the network VRU label in the outbound routing client.

Example:

```
dial-peer voice 6661111 voip
description ******To CVP1*****
destination-pattern 6661111T
session protocol sipv2
session target ipv4:10.10.10.10
voice-class codec 1
voice-class sip rel1xx supported "100rel"
dtmf-relay rtp-nte h245-signal h245-alphanumeric
no vad
```

Note The call can be transferred to only one Unified CVP; in the above example, the call is transferred to CVP1.

Step 2 Configure a dial-peer for the VRU leg. This is the same dial-peer as the inbound call flow whose call is transferred to Unified CVP.

Example:

```
dial-peer voice 777111 voip
description Used for VRU leg
service bootstrap
incoming called-number 7771111T
voice-class sip rel1xx disable
dtmf-relay rtp-nte h245-signal h245-alphanumeric
codec g711ulaw
no vad
```

Step 3 A Routingpattern needs to be configured on Unified CCE Administration so that Unified CVP can route the call to the VXML gateway after it receives the run script request from the router. This dialed pattern is the same one as the inbound call flow that transfers a call to VRU. If the base configuration has not been changed, the pattern is 777111*.

Note It is possible that the procedures in Steps 2 and 3 may have been done already during installation. For more information, see the Cisco Packaged Contact Center Enterprise Administration and Configuration Guide at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html

Outbound Option Scripting

Outbound Option uses Packaged CCE scripting configured on the Administrative Workstation to manage campaigns.

There are two types of scripts:

- Administrative Scripts
- Agent Reservation Routing Scripts

Administrative Scripts for Outbound Option

Outbound Option administrative scripts enable, disable, or throttle campaign skill groups for outbound campaigns. The scripts can also automatically close out a skill group for a specific campaign. The administrative scripts can use time or any other conditional factor that the script can access to close a skill group. You can perform this scripting at the skill group level to provide more flexibility for managing larger campaigns with multiple skill groups.

Enable a campaign skill group by setting the campaign mode to one of the available modes: Preview, Direct Preview, Progressive, or Predictive. Schedule an administrative script to run at regular intervals. Disable the campaign skill group in the administrative script by creating a script node to change the campaign mode to inbound for that skill group.

An administrative script controls a campaign skill group. You can only map a campaign skill group to one campaign at a time. Multiple administrative scripts controlling the same skill group can result in conflicting campaign mode requests.

Note Both the Outbound API and administrative scripts can set the dialing mode for a campaign. The value set by the administrative script takes precedence over the value set by the API.

You can also use administrative scripts to control the percentage of agents that a campaign skill group can use. A script can also set whether to use a skill group for other campaigns or inbound calls.

Note To allow the outbound control and percent configured values from Campaign Skill group (set by either the Configuration Manager Campaign Skill Group tab or the Campaign API) to apply without restarting the router. If you use an administrative script to set the outbound control and percent variables in operation and if you want to employ these configured value on the Campaign Skillgroup, set the outbound control and percent variable to -1 in the administrative script accordingly.

Set Up Administrative Scripts

Use the Script Editor application to create an administrative script for each skill group to set the OutboundControl variable and the skill group reservation percentage. The Outbound Option Dialer uses the value of this variable to determine which mode each skill group uses.

Note

- If the OutboundControl variable is not set, the skill group defaults to inbound. See chapter 1, "Outbound Business Concepts" for detailed information about Outbound Option outbound dialing modes.
- Make sure that the routing client for the translation route labels is Unified CM, which makes the outgoing call.

Perform the following steps to create the administrative script:

Step 1 Open the Script Editor application.

Step 2 Select **File** > **New** > **Administrative Script**.

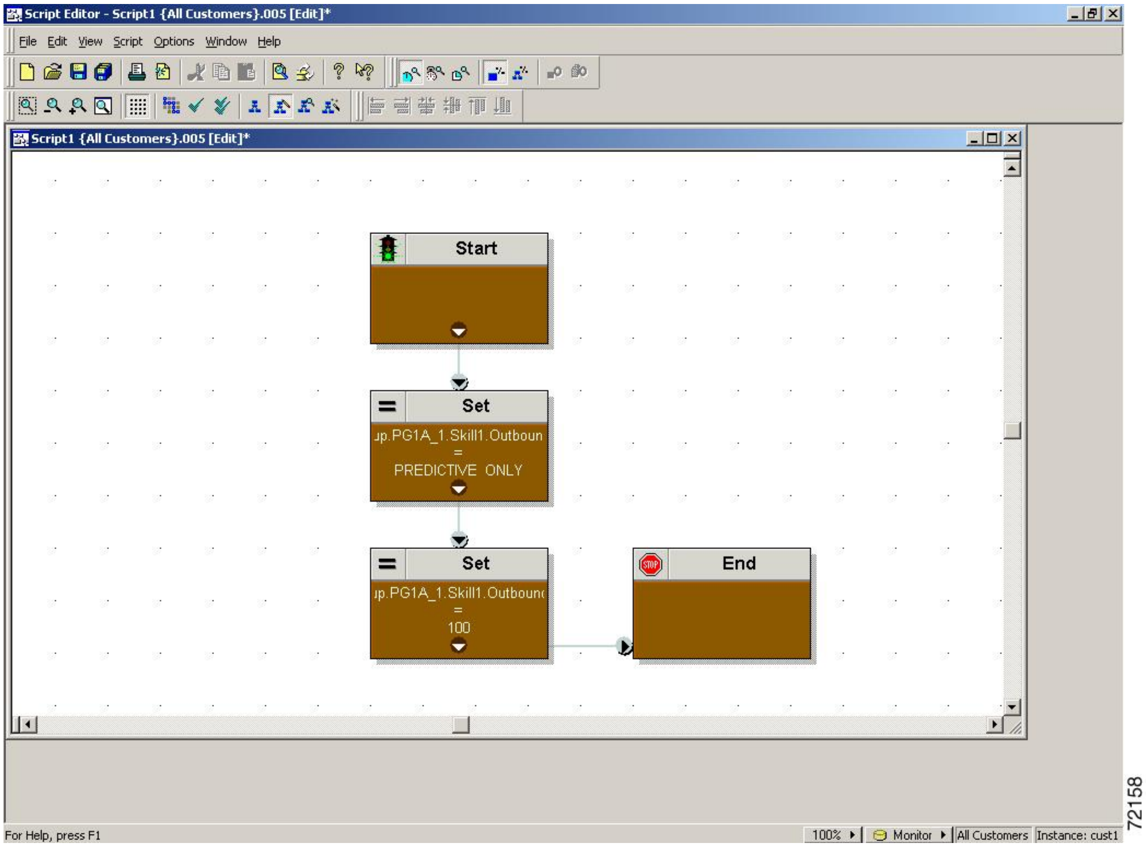
Step 3 Create an administrative script.

One script can be used to control all Outbound Option skill groups or multiple scripts can control multiple Outbound Option skill groups. For example, if you want to control skill groups at different times of the day, you need multiple administrative scripts; however, if you are going to initialize the groups all in the same way, you need only one script (with additional Set nodes).

Step 4 Set up the script with the following nodes (required): Start, Set Variable, and End.

The following diagram displays a simple administrative script where both the OutboundControl variable and the outbound percentage are set for a skill group. A script in a production call center is typically more complex, perhaps changing these variables according to time of day or service level.

Figure 14: Sample Administrative Script


Script Editor - Script1 {All Customers}.005 [Edit]*
File Edit View Script Options Window Help
Script1 {All Customers}.005 [Edit]*
Start
Set
up.PG1A_1.Skill1.Outboun
=
PREDICTIVE ONLY
Set
up.PG1A_1.Skill1.Outbounc
=
100
End
For Help, press F1
100%
Monitor
All Customers
Instance: cust1
72158

Note The Transfer to VRU feature requires an IF node in the administrative script to disable it if the VRU is not available. Also, to ensure timely responses to VRU outages, set the administrative script to run every minute.

Step 5 Set the OutboundControl variable. Setting this variable enables contact center managers to control the agent mode.

Right-click on the work space and select **NEW** > **Object** > **Set Variable** to open the Set Properties window.

- For Object Type, select a skill group.
- For variable, select **OutboundControl**.

Set this variable to one of the values listed in the following table.

Table 16: OutboundControl Variable Values

Value String	Description
INBOUND	Agents take inbound calls only. Outbound dialing is disabled for the skill group.
PREDICTIVE_ONLY	Agents in the skill group are dedicated for outbound Predictive calls only.
PREVIEW_ONLY	Agents in the skill group are dedicated for outbound Preview calls only.
PROGRESSIVE_ONLY	Agents in the skill group are dedicated for outbound Progressive calls only.

Value String	Description
PREVIEW_DIRECT_ONLY	Agents only place outbound calls and hear ringtones, such as phone ringing or busy signal.

Note If the administrative script is changed and the SET variable is removed, the value of the OutboundControl variable is the same as it was the last time the script was run. However, if the Central Controller is restarted, the value resets to INBOUND.

Step 6 Right-click on the work space and select **NEW** > **Object** > **Set Variable** to open the Set Properties window.

- For Object Type, select a skill group.
- For variable, select **OutboundControl**.

Step 7 Set the OutboundPercent variable in the same administrative script; for example, select the OutboundPercent variable in the Set Properties window and enter the agent percentage in the Value field. This variable controls the percentage of agents, which are logged in to a particular skill group, used for outbound dialing. For example, if 100 agents are logged in to a skill group, and the OutboundPercent variable is set to 50%, 50 agents are allocated for outbound dialing for this campaign skill group. This setup allows the rest of the agents to be used for inbound or other active campaigns. The default is 100%.

Note This variable does not allocate specific agents for outbound dialing, just a total percentage. The default is 100%.

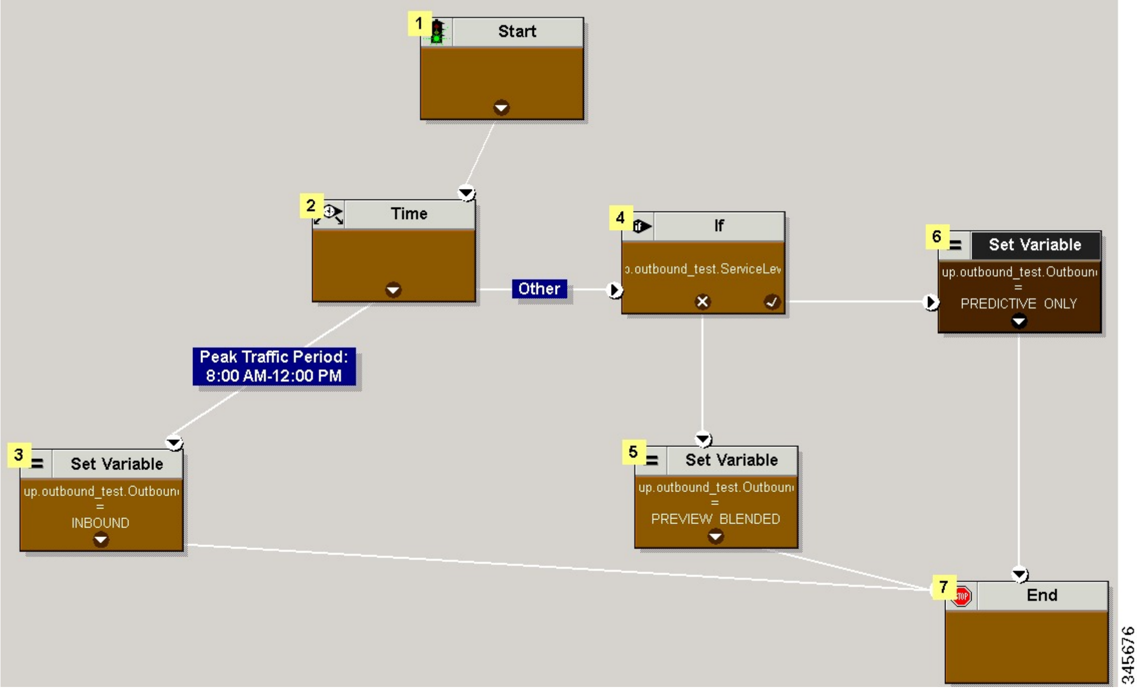
Step 8 Schedule the script.

a) Select **Script** > **Administrative Manager**. An Administrative Manager dialog box appears.
b) Click **Add**.
c) On the Script tab, select the administrative script.
d) On the Period tab, specify the run frequency of the script. (For example, every minute of every day.)
e) (Optional) Enter a description on the Description tab.
f) Click **OK** to close the Add Administrative Schedule dialog box.
g) Click **OK** to close the Administrative Manager.

Sample Administrative Script: ServiceLevelControl

The following figure demonstrates how to control skill group modes based on “Service Level,” which maximizes the resource utilization in a call center and maintains an acceptable service level at the same time.

Figure 15: ServiceLevelControl Script



This script divides the day into two parts:

- **Peak Traffic Period (8:00 a.m. to 12:00 p.m.)**: During this period, the OutboundControl variable is set to INBOUND only, because more agents are required to handle inbound calls.
- **Other Periods**: During all other time periods, the OutboundControl variable is set according to the service level in the past half hour. If the skill group service level in the past half-hour period is over 85%, the OutboundControl variable gets set to PREDICTIVE_ONLY, which maximizes the efficiency of outbound campaigns. If during any half-hour period the skill group service level drops below 85%, the OutboundControl variable is switched to PREVIEW_BLENDED, so that the agents in the skill group can accept inbound calls to improve the service level. When the agents are not in an inbound call, Outbound Option presents the agents with a Preview outbound call, maximizing the resource utilization for the call center at the same time.

Add the IF Node

To add the IF node, follow these steps:

Step 1 Select ObjectType as Skillgroup.

Step 2 Select the skill group that was created for outbound as the Object.

Step 3 Select ServiceLevelHalf as the variable.

Routing Scripts for Outbound Option

Two types of routing scripts are described later in this document. One is for Agent Campaign and one is for VRU Campaign.

Set Up Routing Scripts

Use the Script Editor application to create a reservation script that uses the dialed number for the Outbound Routing Type and routes through one of the following methods:

- Using a Select node to the previously configured skill group.
- Using Dynamic Route Target by ID in the Skill Group node.

Before beginning this procedure, you must create and configure a skill group. For information about creating skill groups, see the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/en/US/products/ps12586/tsd_products_support_series_home.html.

The following sections contain diagrams displaying sample routing scripts.

Script for Agent Campaign Without Personal Callback

The following steps and accompanying diagrams provide an example of how to create a script for an agent campaign without personal callback.

Step 1 Using the **Dialed Number** tool, associate the Outbound Voice dialed numbers with the configured call type.

Step 2 Using the **Call Type Manager** in **Script Editor**, associate the MR dialed numbers with the configured call type and newly created reservation script.

Figure 16: Sample Script for Agent Campaign Without Personal Callback (Using Select Node)

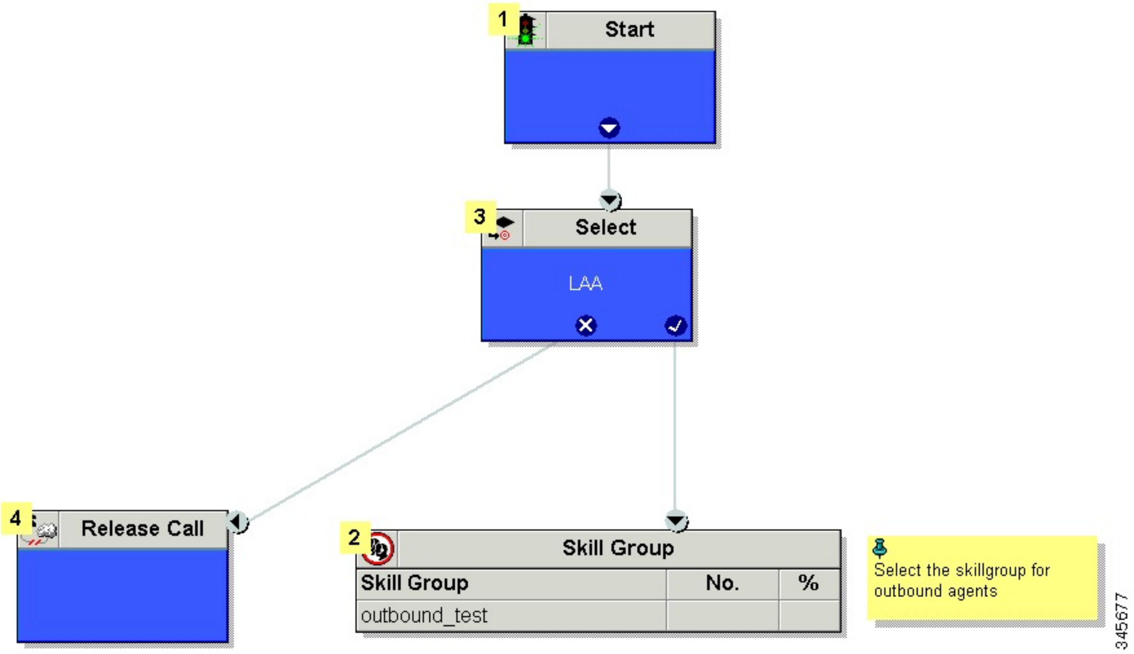
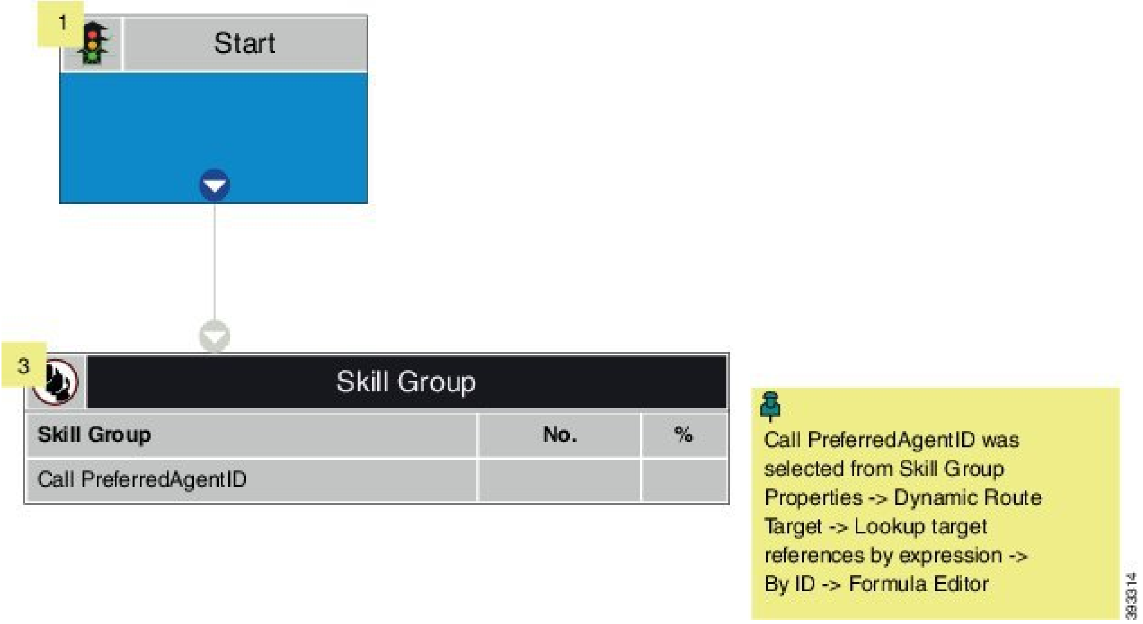


Figure 17: Sample Script for Agent Campaign Without Personal Callback (Using Dynamic Route Target by ID)


1
Start
3
Skill Group
Skill Group
No.
%
Call PreferredAgentID
Call PreferredAgentID was selected from Skill Group Properties -> Dynamic Route Target -> Lookup target references by expression -> By ID -> Formula Editor
393314

Script for Agent Campaign with Personal Callback

The following steps and accompanying diagram provide an example of how to create a script for an agent campaign with personal callback.

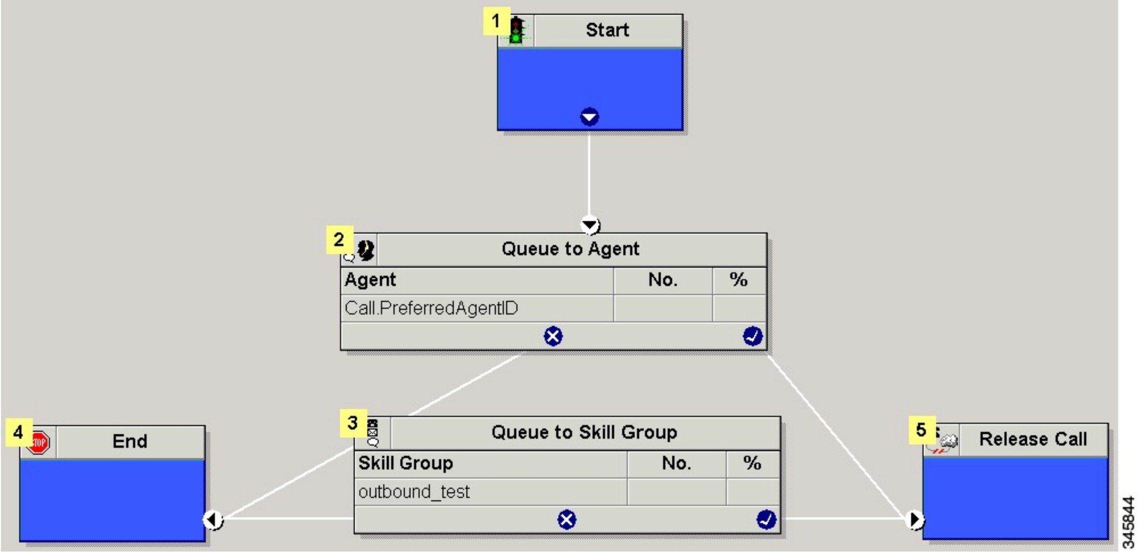
Include the following nodes:

- Add a queue-to-agent node.
- Add a Queue to Skill Group Node after the Queue to Agent Node. Use a skill group that handles outbound calls.
- End the script in a release call node for a successful case; otherwise end the script with the END node.

Step 1 Using the Dialed Number tool, associate the Outbound Voice dialed numbers and Personal Callback dialed numbers with the configured call type.

Step 2 Using the Script Editor Call Type Manager, associate the call type with the newly created reservation script.

Figure 18: Sample Script for Agent Campaign with Personal Callback



Configure Queue to Agent Node

Step 1 In **Script Editor**, double-click the **Queue to Agent** node.

Step 2 In the **Agent Expression** column, enter `Call.PreferredAgentID`.

Step 3 Confirm that the **Peripheral** column is left blank.

Step 4 Click **OK** to save the **Queue to Agent** node.

Step 5 Save and then schedule the script. When scheduling the script, use the call type that is configured for personal callback.

For more information about script scheduling, see the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/en/US/products/ps12586/tsd_products_support_series_home.html.

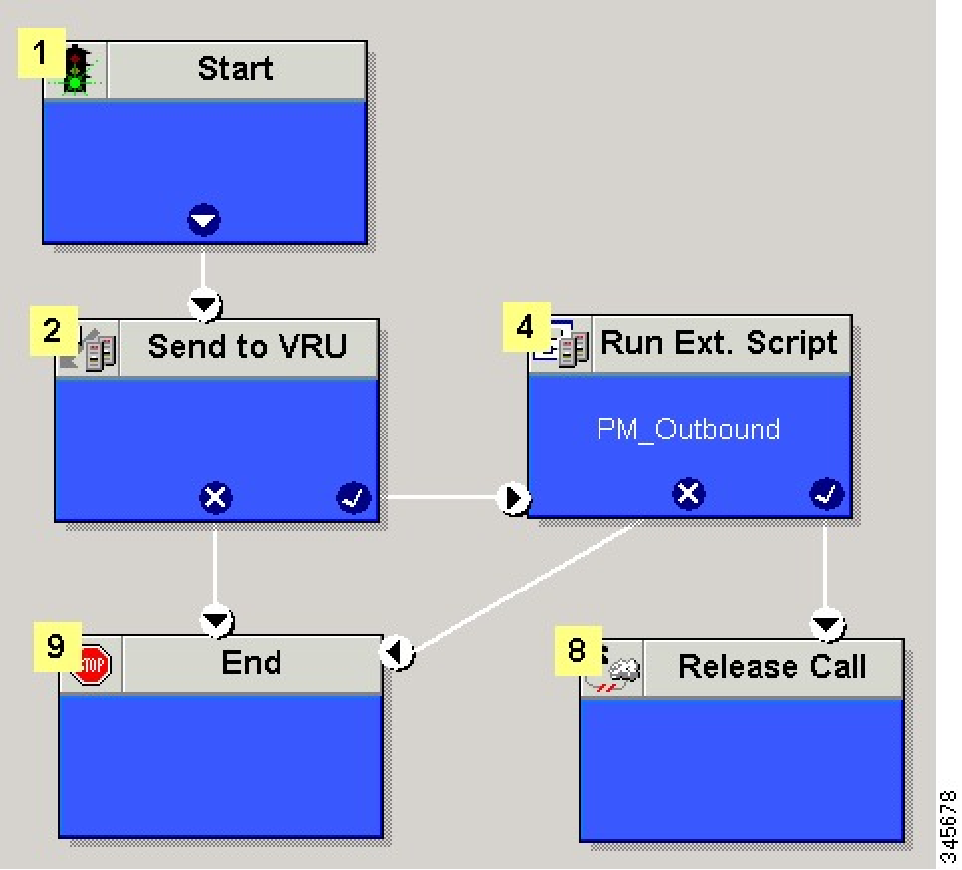
Script for VRU Campaign

The following steps and accompanying diagram provide an example of how to create a script for a VRU campaign.

Step 1 Using the Dialed Number tool, associate the Outbound Voice dialed numbers with the configured call type.

Step 2 Using the Script Editor Call Type Manager, associate the call type with the newly created reservation script.

Figure 19: Sample Script for VRU Campaign



SIP Dialer Recording Parameters Configuration

When recording is enabled in a campaign, the number of recording files that result can be large. The following table lists registry settings that you can adjust to regulate the number of recording sessions and the maximum recording file size.

Registry Setting	Default Setting	Description
MaxAllRecordFiles	500,000,000	The maximum recording file size (in bytes) of all recording files.
MaxMediaTerminationSessions	200	The maximum number of media termination sessions if recording is enabled in the Campaign configuration.
MaxPurgeRecordFiles	100,000,000	The maximum recording file size (in bytes) when the total recording file size, MaxAllRecordFiles, is reached.
MaxRecordingSessions	100	The maximum number of recording sessions if recording is enabled in the Campaign configuration.

Recording files are in the `HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\ICM\<customer instance>\Dialer directory`.

Note Only the G.711 codec is supported for recording. To record outbound calls, configure the G.711 on the voice gateway.

Verification of Dialed Number

Outbound Option places agents in the Reserved state before using them for an outbound call. The dialer uses the dialed number to route to an agent. The following procedure describes how to verify that this mechanism works properly.

Verify DN Configuration

When an Outbound Option Dialer is installed in a Unified CCE environment, the dialer uses the dialed number to make routing requests through the Media Routing (MR) Peripheral Gateway. The following verification steps assume that you have completed all the applicable configuration and reservation script generation.

Step 1 Log in an agent to a skill group participating in an outbound campaign, and make the agent available. (Note the dialed number, which was configured in the Skill Group Selection tab in the Campaign component.) If a different dialed number is used for predictive and preview calls, make sure to verify both dialed numbers.

Step 2 Run the Script Editor application and select the **Call Tracer** utility from the **Script** > **Call Tracer** menu. Select the routing client that is associated with the MR PG and select the Dialed Number.

Step 3 Press **Send Call** to simulate a route request and note the results. If a label was returned for the agent who was logged in above, the reservation script is working properly and the dialer can reserve agents through this script.

Verify Campaign Configuration

As a final step to verify that you configured your Outbound Option campaign correctly, create a small campaign of one or two entries that dial work phones or your mobile phone.

Campaign Management

Single Campaign Versus Multiple Campaigns

You might choose to run multiple campaigns because of different calling policies (for example, time rules) or to run different outbound modes simultaneously.

From the perspective of dialer port allocation, running fewer campaigns with a larger agent pool is more efficient. Dialer ports are allocated based on the number of agents assigned and the current number of lines per agent to dial. The more campaigns you have that are active, the more the ports are distributed across the campaigns, which affects overall efficiency.

Results from Individual Customers

After running a campaign, you can generate a list of customers who were reached, not reached, or have invalid phone numbers.

Interpret Information from Dialer_Detail Table

The Dialer_Detail table is a single table that contains the customer call results for all campaigns. When you view the Dialer_Detail table, note that each attempted Outbound Option call is recorded as an entry in the table. Each entry lists the number called and which numbers are invalid.

For more information, see the appendix on the Dialer Detail Table.

Management of Campaign Manager Database Tables

The Campaign Manager tables, Dialing_List and Personal_Callback_List can grow to be large. If the database size grows too large, Campaign Manager performance can significantly slow down. To limit the size of the Outbound Option database, a stored procedure is run daily at midnight to purge records that are no longer needed.

By default, records are removed from the Personal_Callback_List table when the record's **CallStatus** is either C, M, or D, and the **CallbackDateTime** for the record is more than five days old. In the Dialing_List table, records are removed by default when **CallStatusZone1** has a value of either C, M, or D, and **ImportRuleDate** is more than five days old.

You can change the status and age of the records to be removed by modifying the Campaign Manager registry values on the Logger machine. The registry settings are located in HKEY_LOCAL_MACHINE\SOFTWARE\Cisco Systems, Inc.\ICM\<instance name>\LoggerA\BlendedAgent\CurrentVersion in the Outbound Option registry.

- To specify the records to remove from the Personal_Callback_List table, set **PersonalCallbackCallStatusToPurge** and **PersonalCallbackDaysToPurgeOldRecords**.

Note **PersonalCallbackCallStatusToPurge** is not added by default. To change the call status of the records to remove, create this registry setting manually.

- To specify the records to remove from the Dialing_List table, set **DialingListCallStatusToPurge** and **DialingListDaysToPurgeOldRecords**.

Note **DialingListCallStatusToPurge** is not added by default. To change the call status of the records to remove, create this registry setting manually.

To specify the age of the records to be removed, set **PersonalCallbackDaysToPurgeOldRecords** or **DialingListDaysToPurgeOldRecords** to specify the number of days to keep the record before it is removed. For the Personal Callback list, this value is the number of days after the personal callback is scheduled (CallbackDateTime). For the Dialing List, this value is the number of days after the record is imported (ImportRuleDate). The default is 5. The valid range is 1 to 30. If the value is not set or set to 0, the automated purge is disabled.

To set the call status of the records to be removed, set **PersonalCallbackCallStatusToPurge** or **DialingListCallStatusToPurge** to a string containing the call status types to apply when purging personal callback or dialing list records. For example, if the string contains “C,M,F,L,I,” all records with these call statuses, that are also older than the number of days specified by **PersonalCallbackDaysToPurgeOldRecords** or **DialingListDaysToPurgeOldRecords**, are removed from the database.

You can specify the following call status values:

Value	Description
U	Unknown
F	Fax
I	Invalid Number
O	Operator
L	Not Allocated
X	Agent Not Available
C	Closed
M	Max Calls
D	Dialed

Management of Predictive Campaigns

The following sections provide guidelines to follow when working with predictive campaigns.

Initial Values for Lines per Agent

Determining the initial value for the number of lines per agent is not as simple as inverting the hit rate. If a campaign has a 20% hit rate, you cannot assume that five lines per agent is the applicable initial value for the campaign if you are targeting a 3% abandon rate. The opportunity for abandoned calls increases geometrically as the lines per agent increases; therefore, set the initial value conservatively in the campaign configuration.

If the reports show that the abandon rate is below target and does not come back in line very quickly, modify the initial value in the campaign configuration to immediately correct the lines per agent being dialed.

End-of-Day Calculation for Abandon Rate

It is not unusual for a campaign to be over the abandon rate target for any given 30 minute period. The dialer examines the end-of-day rate when managing the abandon rate. If the overall abandon rate is over target for the day, the system targets a lower abandon rate for remaining calls until the average abandon rate falls into line. This end-of-day calculation cannot work until after the campaign has been running for one hour. Small sample sizes due to short campaigns or campaigns with fewer agents might not give the dialer enough time to recover from an initial value that is too high.

Similarly, if the campaign is significantly under the target abandon rate, it might begin dialing more frequently with an abandon rate over target for a while to compensate in the abandon rate.

Transfer of Answering Machine Detection Calls to Agents

When enabling the Transfer AMD (Answering Machine Detection) to agent option for an agent campaign or enabling the Transfer AMD to IVR option for an IVR campaign, consider the increase in calls to the target resources (agents or IVR) when determining the initial value. If the expectation is that the AMD rate and the live voice rate are over 50%, perhaps start out with an initial value of 1.1 or even one line per agent to stay under a 3% abandon rate.

Management of Agent Idle Time

One of the key reporting metrics for administrators managing campaigns is the amount of time agents spend idle between calls.

There are many possible reasons for longer idle times, such as a combination of one or more of the following:

- A dialing list with a low hit rate. The solution is to create an improved list.
- A small agent pool results in fewer calls, resulting in slower adjustments. One solution is to add more agents to the pool.
- Shorter average handle times means agents become available more frequently. A shorter handle time means that the agent idle time percentage will climb.
- Not enough dialer ports deployed or too many agents. Deploy more ports or use fewer agents.
- A large number of retry attempts at the beginning of a day when running with append imports resulting in lower hit rates. Prioritize pending over retries.
- Modifying the maximum number of attempts up or down in an active campaign. This activity can interrupt the Campaign Manager's processing of dialer requests for records, as mentioned earlier in this chapter. One solution is to perform the activity during off hours.
- Running out of records to dial. Import new records.

Sources of Higher Idle Times in Reports

The following Outbound Option reports provide information regarding sources of higher idle times:

- Campaign Consolidated Reports: These reports provide a very useful overview of a campaign by combining campaign and agent skill group statistics into a single report. They provide average idle time, campaign hit rate, the number of agents working on the campaign, as well as their Average Handle Time per call. Low hit rates and low average handle times result in more work for the dialer to keep those agents busy.
- Dialer Capacity Reports: These reports show how busy the dialers are and how much time was spent at full capacity when the dialer was out of ports. They also provide the average reservation call time as well as the average time each dialer port spent contacting customers.

Dialer Saturation

If both Dialers have relatively low idle times and high all ports busy times, then it is likely the Dialers have been oversubscribed. The combination of number of agents, Dialing List hit rate, and average handle time are likely more than the deployed number of ports the Dialer can handle.

To solve this problem, perform one of the following actions:

- Reduce the number of agents working on the campaign.

- Add more Dialer ports to the solution.

Few Available Records

Call Summary Count reports show how many records in the aggregate campaign dialing lists have been closed and how many are still available to dial.

Reports

This section provides an overview of the Outbound Option reports available in the Cisco Unified Intelligence Center.

For detailed report template descriptions, see the *Cisco Packaged Contact Center Enterprise Reporting User Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

For directions on importing report templates into Cisco Unified Intelligence Center and configuring Cisco Unified Intelligence Center data sources for Packaged CCE, see the *Cisco Packaged Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-installation-guides-list.html.

Outbound Option Reports

This section describes the Outbound Option reports, created using the Unified Intelligence Center.

- Outbound Historical Reports Bundle, on page 181
- Outbound Realtime Reports Bundle, on page 182
- Agent Reports, on page 183
- Campaign and Dialer Reports, on page 184
- Skill Group Reports, on page 185

Additionally, sample custom report templates are available from the Cisco Developer Network (https://developer.cisco.com/web/ccr/documentation.)

Note Call Type reporting can be used on Outbound Option reservation calls and transfer to VRU calls. Call Type reporting is not applicable for outbound customer calls because a routing script is not used.

Outbound Historical Reports Bundle

The Outbound Options Historical reports receive data from the historical data source. Reports are populated with interval data that has a default refresh rate of 15 minutes.

Half-hour/Daily: Provides statistics for each half-hour period. Many of the half-hour reports are also available in a daily report format.

The Outbound Historical bundle contains the following reports:

Report	Description
Attempts per Campaign Daily	Shows the status (summary and percentage) of each campaign for the selected time period and the breakdown of attempts (in percentage) of each campaign for the selected time period.
Campaign Consolidated Daily	Shows the daily activity and performance of the selected campaigns and their skill groups for the selected time period and provides analysis of the actual customer calls (outbound calls which reached live voice, inbound calls, or calls transferred to the campaign's skill group) for the selected campaigns and their skill groups for the selected time period.
Campaign Consolidated Half Hour	Shows the list of Consolidated Calls and Agent Statistics per Campaign by Half Hour and Breakdown of completed calls.
Campaign Half Hour Summary	Shows the status for all campaigns for the selected time period, the status (summary and percentage) of each campaign for the selected time period and the breakdown of attempts (in percentage) of each campaign for the selected time period.
Dialer Call Result Summary Half Hour Dialer Capacity Daily Dialer Capacity Half Hour	Shows the status of each dialer for the selected time period.
Import Rule	Shows the status of imported records for the selected time period.
Query Rule Within Campaign Daily	Shows the breakdown of attempts (in percentage) of each campaign for the selected time period and the status (summary and percentage) of each campaign for the selected time period.

Outbound Realtime Reports Bundle

The Outbound Option Real Time reports display current information about a system entity; for example, the number of tasks an agent is currently working on or the number of agents currently logged in to a skill group. By default, the reports automatically query the Admin Workstation database on the distributor every 15 seconds. The data is written to the database by the Router almost every 10 seconds.

The Outbound Real Time Reports Bundle contains the following reports:

Report	Description
Call Summary Count per Campaign Real Time	Shows status of all campaign records, and the currently valid campaign dialing times.
Dialer Real Time	Shows the status of each dialer, including the number of contacts dialed today and the result of each attempt.

Report	Description
Import Status Real Time	Shows the status of Outbound Option import records.

Agent Reports

In addition to the reports contained in the Outbound Reports bundles, other Agent reports also provide information about Outbound activities:

Report	Outbound Option Fields
Agent Queue Real-Time Agent Real-Time Agent Skill Group Real-Time Agent Team Real-Time	The **Direction** field indicates the direction of the call that the agent is currently working on including Other Out/Outbound Direct Preview, Outbound Reserve, Outbound Preview, or Outbound Predictive/Progressive. The **Destination** field indicates the type of outbound task on which the agent is currently working.
Agent Team State Counts	The **Active Out** field shows the number of agents currently working on outbound tasks.
Agent State Real Time Graph	For agents handling Outbound Option calls, the **Hold** state indicates that the agent has been reserved for a call. The Outbound Dialer puts the agent on hold while connecting the call.

Interpreting agent data for Outbound Option tasks, requires understanding how Outbound Option reserves agents, reports calls that are connected to agents, and handles calls that are dropped by customers before the calls are connected.

The Outbound Option Dialer assigns and connects calls differently than regular contact center enterprise routing. Report data for agents handling Outbound Option calls therefore differs from data for agents handling typical voice calls and multichannel tasks.

When the Outbound Dialer calls a customer, it reserves the agent to handle the call. The Dialer places a reservation call to the agent and changes the agent's state to Hold. This reservation call is reported as a Direct In call to the agent.

For typical calls, the agent is placed into Reserved state when the contact center reserves the agent to handle a call. For Outbound Option calls, reports show the agent in Hold state when reserved for a call and the time that agent spends reserved is reported as Hold Time.

When the customer answers the call, the Outbound Option Dialer transfers the call to an agent. The call is now reported as a Transfer In call to the agent. When the customer call is transferred to the agent, the Dialer drops the reservation call and classifies it as Abandon on Hold.

The abandoned call wait time, set in the Campaign Configuration screen, determines how calls are reported if the caller ends the call. Calls are counted in the Customer Abandon field in both Real Time and Historical campaign query templates only if the customer ends the call before the abandoned call wait time is reached.

For agent reporting per campaign, Outbound Option provides reports that accurately represent the Outbound Option agent activity for a contact center, including information grouped by skill group.

The following list describes the data that are presented in the agent reports.

- A real-time table that shows Outbound Option agent activity that is related to Outbound Option calls.
- A historical table that shows agent daily performance for Outbound Option predictive calls, by skill group.
- A historical table that shows agent daily performance for Outbound Option preview calls, by skill group.
- A historical table that shows agent daily performance for Outbound Option reservation calls, by skill group.

Campaign and Dialer Reports

Outbound Option provides a campaign report template that describes the effectiveness of a campaign and the dialer. This list can be used for Agent and VRU campaigns.

Observe the following guidelines when using the campaign reports:

- Campaign Real Time reports describe how many records are left in the campaign dialing list.
- Both Campaign and Dialer Half Hour reports provide the call result counts.

Note Campaign Real Time reports capture call results since the last Campaign Manager restart only. If the Campaign Manager restarts, data collected before the restart is lost.

Note When the active Campaign Manager fails over, partial campaign interval reports are generated for the relevant interval based on the data that was available after failover. Some of the campaign statistics collected prior to failover will be missing.

The campaign interval tables used in Reporting are impacted due to this scenario.

The following list describes the data that is presented in the campaign reports.

- A summary of call results for query rules within a campaign since the beginning of the day.
- A summary of call results for a campaign since the beginning of the day. It includes a summary of all query rules within the campaign.
- A view of what is configured for valid campaign calling times for zone1 and zone2 for selected campaigns. The times are relative to the customer’s time zone.
- A view of what is configured for valid campaign calling times for zone1 and zone2 for selected campaign query rules. The zone times are relative to the customer’s time zone. The query rule start and stop times are relative to the Central Controller time.
- How many records for selected query rules have been dialed to completion, and how many records remain.
- How many records for selected campaigns have been dialed to completion, and how many records remain.
- A summary of call results for selected campaign query rules for selected half-hour intervals.
- A summary of call results for all query rules for selected campaigns for selected half-hour intervals.

- A historical table by half-hour/daily report that shows the status (summary and percentage) of each campaign for the selected time period.
- A historical table by breakdown of attempts (in percentage) of each campaign for the selected time period.
- A historical table by half-hour/daily report that shows the status (summary and percentage) per query rule of each campaign for the selected time period.
- A historical table by breakdown of attempts (in percentage) per query rule of each campaign for the selected time period.
- A summary half-hour/daily report that shows activity and performance of the selected campaigns and their skill group for the selected time period, including abandon rate, hit rate, and agent idle times.
- A historical table by breakdown of actual customer calls (outbound calls which reached live voice, inbound calls, or calls transferred to the campaign skill group) for the selected campaigns and their skill groups for the selected time period.

Dialer Reports

The Outbound Option Dialer reports provide information about the dialer. These reports include information about performance andresource usage. The templates also enable you to determine whether you need more dialer ports to support more outbound calls.

The following list describes the data presented in the Outbound Option Dialer reports:

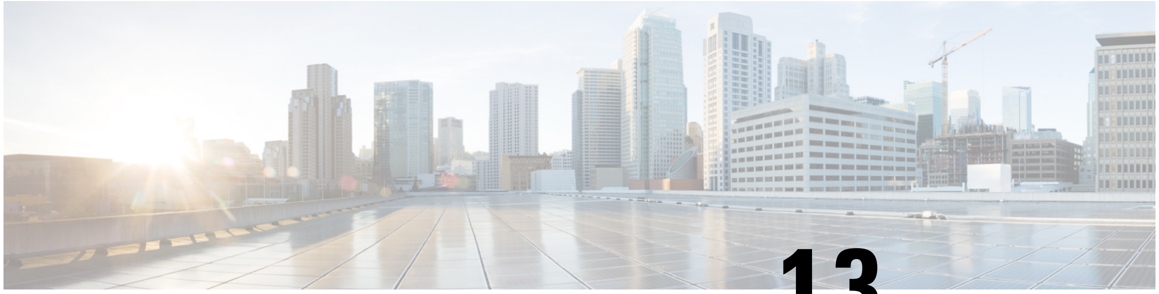
- A real-time table that shows contact, busy, voice, answering machine, and special information tone (SIT) detection for each dialer. A SIT consists of three rising tones indicating a call has failed.
- An historical table that records contact, busy, voice, answering machine, and SIT Tone detection for each dialer by half-hour intervals.
- Displays information about the amount of time the dialer was idle or had all ports busy.
- Displays Dialer status on a port-by-port basis used for troubleshooting. If this report does not display any records, then the data feed is disabled by default. It is only enabled for troubleshooting purposes.

Skill Group Reports

For skill group reporting per campaign, Outbound Option provides reports that represent the skill group activity for a contact center.

The following list describes the data presented in the skill group reports:

- A real-time table that shows all skill groups and their associated Outbound Option status.
- A historical table that records Outbound Option counts for the agent states *signed on*, *handle*, *talk*, and *hold* by half-hour intervals.



CHAPTER 13

Post Call Survey

Capabilities

A Post Call Survey takes place after usual call treatment. It is typically used to determine whether customers are satisfied with their call center experiences. This feature lets you configure a call flow that, after the agent disconnects from the caller, optionally sends the call to a Dialed Number configured for a Post Call Survey.

The Unified CCE script can enable and disable Post Call Survey on a per-call basis by testing for conditions and setting an expanded call variable that controls post call survey. 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 can set the expanded call variable that controls whether the call gets transferred to the Post Call Survey dialed number.

The Post Call Survey call works just like a regular call from the Unified CCE point of view. Scripts can be invoked and the customer can use the keypad on a touch tone phone and/or voice with ASR/TTS to respond to questions asked during the survey. During Post Call Survey, the call context information is retrieved from the original customer call.

Note The call context for the post call survey includes all context up to the point where the call is transferred to the agent. Context that the agent creates after the transfer is not included in the post call survey context.

Design Considerations

Observe the following conditions when designing the Post Call Survey feature:

- A Post Call Survey is triggered by the hang-up event from the last agent. When the 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 is transferred to the Post Call Survey number.
    - If **user.microapp.isPostCallSurvey** is set to **y** (the implied default), the call is transferred 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.
- REFER call flows are not supported with Post Call Survey. The two features conflict: REFER call flows remove Unified CVP from the call and Post Call Survey needs Unified CVP because the agent has already disconnected.
- For Unified CCE reporting purposes, when a survey is initiated, the call context of the customer call that was just transferred to the agent is replicated into the call context of the Post Call Survey call.

Initial Setup

To set up the Post Call Survey feature:

Step 1 Create one or more survey scripts and add the files to the CVP media servers. See Create a Survey Script, on page 188.

Step 2 Configure Unified CCE for Post Call Survey. This step adds a required expanded call context variable, adds a new call type for Post Call Survey, maps incoming dialed number to a survey dialed number pattern, and associates your survey dialed number patterns to the survey call type. See Configure Packaged CCE for Post Call Survey, on page 188.

Step 3 Modify your Unified CCE call routing scripts to launch the survey scripts. See Modify CCE Scripts for Post Call Survey, on page 191.

The scripts can optionally contain nodes that test for conditions and dynamically control whether a call is transferred to the survey.

Create a Survey Script

To create a survey script or application that queries the caller for information, use the CVP Call Studio tool. For more information on Unified CVP Call Studio, see User Guide for Cisco Unified CVP VXML Server and Unified Call Studio.

What to do next

Map CVP dialed number patterns to the survey script numbers.

Configure Packaged CCE for Post Call Survey

You can enable and disable Post Call Survey within a CCE routing script by using the ECC variable **variableuser.microapp.isPostCallSurvey**. A value of *n* or *y* disables and enables the feature. (The value is case-insensitive.)

Configure the ECC variable to a value of n or y before either the label node or the Queue to Skillgroup node. This configuration 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 once Post Call Survey is configured in the Unified CCE Administration. Dialed Number is mapped to the Post Call Survey Dialed Number patter to automatically transfer the call.

Note

- The Post Call Survey DN is called if the Unified CVP has received at least one CONNECT message from CCE (either from the VRU leg or from the Agent leg). Use the END node in your CCE routing script if the Post Call Survey is not required for the calls disconnected from the IVR.
- If Router Requery is configured incorrectly and the Ring-No-Answer timeout expires, the caller is still transferred to the Post Call Survey DN. This can occur if a Queue node is used and Enable target requery is not checked.

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

Step 2 Click **New** to open the **New Expanded Call Variable** window.

Step 3 Create a new ECC variable with **Name:**`user.microapp.isPostCallSurvey`.

Step 4 Set **Max Length:** to 1.

Step 5 Check the **Enabled** checkbox. Then click **Save**.

In your CCE routing scripts, remember that, at script start, the default behavior of Post Call Survey equals **enabled**, even if **user.microapp.isPostCallSurvey** has not yet been set in the script. You can turn **off** Post Call Survey 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*.

Step 6 Navigate to **Overview** > **Call Settings** > **Route Settings** > **Call Types**.

Step 7 Add the call type for Post Call Survey, and click **Save**.

Step 8 Navigate to **Overview** > **Call Settings** > **Route Settings** > **Dialed Numbers**.

Step 9 Click **New** and complete the following fields:

Field	Required?	Description
Dialed Number String	yes	The value used to route the call, which is the Post Call Survey Dialed Number. Enter a string value that is unique for the routing type, maximum of 25 characters. **Note** The **External Voice** and **Post Call Survey** routing types must not have the same dialed number strings for the same site.
Description	no	Enter a maximum of 255 characters to describe the dialed number string.

Field	Required?	Description
Department	yes (for departmental administrators)	A departmental administrator must select one department from the popup list to associate with this dialed number. The list shows all this administrator's departments. When a departmental administrator selects a department for the dialed number, the popup list for call type includes global call types and call types in the same department as the dialed number. A global administrator can leave this field as Global (the default), which sets the dialed number as global (belonging to no departments). A global administrator can also select a department for this Dialed Number. When an administrator changes the department, selections for call type are cleared if the selections do not belong to the new department or the global department.
Routing Type	yes	From the drop-down menu, select **Post Call Survey:** . **Post Call Survey:** Select this option for Post Call Survey dialed number strings that apply to voice calls coming from Cisco Unified Customer Voice Portal (CVP). This option is similar to External Voice where the calls comes from outside of the enterprise through a gateway. However, Unified CVP directs the calls internally to Post Call Survey after agent ends the call. This option allows you to enter the Post Call Survey Dialed Number and associate the Dialed Number Patterns to the Post Call Survey Dialed Number. For remote sites, the **Post Call Survey** option is available if the site is configured to VRU PG.
Media Routing Domain	no	The Media Routing Domain associated with the dialed number. Media Routing Domains (MRDs) organize how requests for media are routed. The system routes calls to agents who are associated with a particular communication medium; for example, voice or email. The selection of Routing Type determines what appears in this field. • If the Routing Type is External Voice, Internal Voice, or Outbound Voice, the Media Routing Domain is Cisco_Voice and you cannot change it. • If the Routing Type is Multichannel, click the **magnifying glass** icon to display the **Select Media Routing Domain** popup window.
Call Type	no	Use the drop-down menu to select the call type that you created for Post Call Survey.

Field	Required?	Description
PCS Enabled Dialed Number Patterns	no	**Note** The **PCS Enabled Dialed Number Patterns** field appears if the **Routing Type** is **Post Call Survey**. Enter one or more dialed number patterns that allow calls to transfer to the Post Call Survey dialed number entered in the **Dialed Number String** field. The field allows maximum of 512 characters that can have the comma separated list without any spaces. Both alphanumeric and special characters are supported.
Ringtone Media File	no	**Note** The **Ringtone Media File** field appears if the **Routing Type** is **External Voice**. Enter filename of the custom ringtone - maximum of 256 characters without any spaces.

Step 10 Click **Save**.

Step 11 Restart the active generic PG (side A or B) to register the new ECC variable.

If the ECC variable already existed, 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. Therefore, if you do not want the survey to run, without first reaching an agent (such as 'after hours of treatment'), you must set the isPostCallSurvey to *n* before the initial 'Run script request'.

Modify CCE Scripts for Post Call Survey

In Script Editor, modify your CCE call routing scripts for incoming calls as follows:

- Add nodes to invoke the call studio survey script, if needed. The following notes explain when you might need to explicitly add nodes to call the survey script.

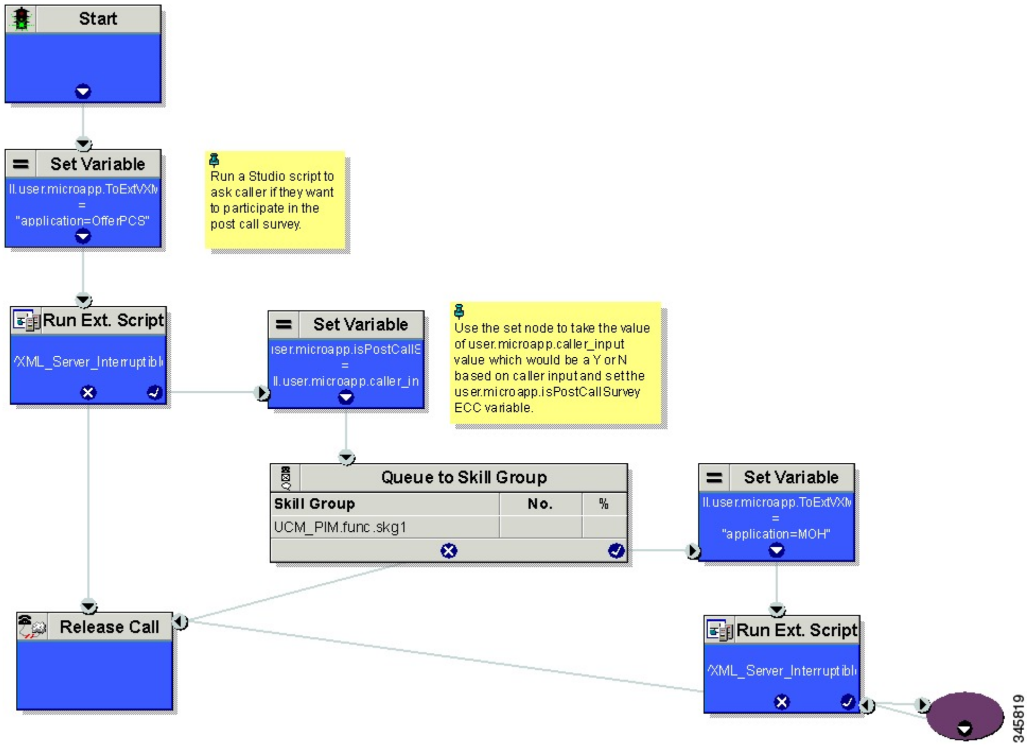
  If a DN is mapped for Post Call Survey, the call is automatically transferred to the configured Post Call Survey dialed number.

  Note The Post Call Survey dialed number is only called if the script ends with a call to an agent. If the script completes without going to an agent then the call is not directed to the Post Call Survey dialed number . In these cases, you can, for example, use a *Send to Script* node in your Unified CCE script to direct the call to the Post Call Survey script.

- Optionally, you can add nodes in the script to test for conditions for which you want to turn the survey off.

- To dynamically control whether the survey is offered to callers, you must explicitly set the **user.microapp.isPostCallSurvey** expanded call context variable to **y** and **n**.
- To offer the survey to all callers, you do not need to set the variable in the script. It is set to **y** by default.
- Configure the expanded call context variable to a value of *n* or *y* before the Queue to Skillgroup node. This sends the correct value to Unified CVP before the agent transfer.

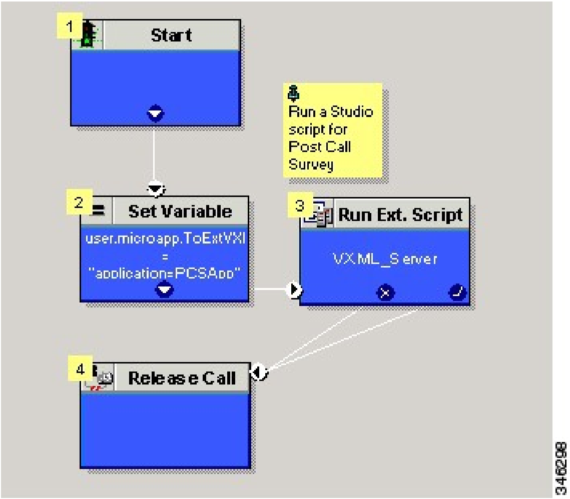
The following example calls a script that asks callers if they want to participate in a survey. The script then sets the **user.microapp.isPostCallSurvey** variable according to the caller's response.



Run a Studio script to ask caller if they want to participate in the post call survey.

Use the set node to take the value of user.microapp.caller_input value which would be a Y or N based on caller input and set the user.microapp.isPostCallSurvey ECC variable.

Create a routing script for the Post Call Survey Call Type to play your survey script or application to the caller. The following script is an example:


1
Start
Run a Studio script for Post Call Survey
2
Set Variable
user.microapp.ToExtVXI = "application=PCSApp"
3
Run Ext. Script
VXML_Server
4
Release Call
346298

Administration and Usage

Get Survey Results

For reporting purposes, in both the CVP and the CCE databases, a post call survey call has the same RouterCallKey, Call GUID, and call context as the original inbound call.

To obtain survey results, you query or create a report that gathers survey data from the CVP database.

For more information on how to configure a Data Source, see the *Cisco Packaged Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-installation-guides-list.html.

Step 1 In Cisco Unified Intelligence Center Reporting tool, connect to the CVP database.

Step 2 Create a query that identifies survey calls, gathers call information from those calls, and extracts data related to specific survey dialed numbers:

a) In the Call_Type table, test for Event_Type = Post_Call_Survey.

b) If true, use that entry's call information to query the VXML_Element table and get the VXML data for the call.

c) In the VXML data, you can identify the exact survey that a caller participated in from the dialed number used to place the Post Call Survey.

Step 3 To report on the results of a particular survey, collate the VXML data for all calls with that survey's dialed number.

Step 4 To identify answers to survey questions, in the CauseRef table, the CauseID is 20, and the Cause is Post Call Answer.

CHAPTER 14

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 Packaged 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.

This is only applicable for Unified CCE deployment.

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.

Cisco IdS has now shifted to asymmetric key encryption for signing the tokens generated for authentication. This allows any client with the matching public key to easily and conveniently authorize any other client that has a token that is signed using the matching private key. You can access the public key using the Cisco IdS CLIs that are described in Access Public Key Signing Certificate. The public key can be freely distributed without any security concerns. All solution components monitor the public encryption key exposed by Cisco IdS using the REST APIs for their token authentication purposes. For more information about the SSO SDK, see Cisco Finesse REST API with SSO Guide.

Note Due to the change in Cisco IdS token mechanism starting 12.6.2, agents must log out and log in to Finesse Desktop after Cisco IdS is upgraded to 12.6(2). To avoid this requirement, install 12.6(2) ES02. For more information, see the Cisco Unified Contact Center Enterprise Installation and Upgrade Guide or Cisco Packaged Contact Center Enterprise Installation and Upgrade Guide.

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.

Related Topics

Access Public Key Signing Certificate

Contact Center Enterprise Reference Design Support for Single Sign-On

Packaged CCE supports single sign-on for these reference designs:

- 2000 Agents
- 4000 Agents
- 12000 Agents

Coresidency of Cisco Identity Service by Reference Design

Reference Design	Packaged CCE Solution
2000 Agent	Cisco IdS is coresident with Unified Intelligence Center and Live Data on a single VM.
4000 Agent	Standalone Cisco IdS VM
12000 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.

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 17: 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

Operation	Allowed on Publisher	Allowed on Subscriber
Update AuthCode/Token Expiry	Always	Only when publisher is connected
Download Token Public Key	Always	Always
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

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.

Single Sign-On Flow

Single sign-on (SSO) configuration by an administrator follows this flow:

Step 1 Install the appropriate release of Packaged CCE. For more information, see Cisco Packaged Contact Center Enterprise Installation and Upgrade Guide at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-installation-guides-list.html

Step 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

For Packaged CCE deployments, the Cisco IdS is installed as a service on the Unified Intelligence Center VMs.

Step 3 Install and configure the Identity Provider (IdP).

Step 4 Configure System Inventory.

Step 5 Configure the Cisco IdS.

Step 6 Register and test SSO-compatible components with the Cisco IdS.

Step 7 Choose the SSO mode.

Step 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.

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 199
2	Set Authentication Type. See Authentication Types, on page 200.
3	Configure an Identity Provider (IdP), on page 199
4	Enable Signed SAML Assertions, on page 204
5	Optionally Customize the AD FS Sign-In Page in Windows Server to Allow User ID, on page 204

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 2.0, see *AD FS Content Map* at http://aka.ms/adfscontentmap.

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.

Note The Secure Hash Algorithm (SHA) used for signature verification between:

- IdP and Cisco IdS: SHA-1, SHA-256
- Cisco IdS and the application browsers: SHA-256

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.

After the expiry of SHA-1, the administrator must configure SHA-256..

Perform this procedure after the upgrade has completed successfully.

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>
```

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 2.0 see https://social.technet.microsoft.com/wiki/contents/articles/1600.ad-fs-2-0-how-to-change-the-local-authentication-type.aspx
- 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.

SUMMARY STEPS

1. In the Server Manager, open **AD FS Management**.
2. For form-based authentication, set the **Authentication Methods** for **Intranet** to **Forms Authentication**.
3. Download LAN SP metadata and reverse-proxy cluster SP metadata from the Cisco IdS publisher.
4. Download the IdP metadata file, `federationmetadata.xml`, from the following location:
5. Do one of the following to upload the IdP metadata file, you downloaded at step 4, to the Cisco Ids server:
6. Follow these steps to create a Relying Party Trust.
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:
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.
9. In the **Edit Claim Rules**/**Edit Claim Issuance Policy** window that opens, click **Add Rule** and then click **OK**.
10. In the **Add Transform Claim Rule Wizard** that opens, follow these steps to create the first claim:
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:
12. Click **OK**.

DETAILED STEPS

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**.

Field	Description
Mapping of LDAP Attributes to Outgoing Claims	If the identifier is a Security Account Name (SAM), do the following: • Select **SAM-Account-Name** as one of the LDAP attributes and set the **Outgoing Claim Type** to "uid." • Select **User-Principal-Name** as one of the LDAP attributes and set the **Outgoing Claim Type** to "user_principal". If the identifier is a User Principal Name (UPN), do the following: • Select **User-Principal-Name** as one of the LDAP attributes and set the **Outgoing Claim Type** to "uid." • Select **User-Principal-Name** again as the LDAP attribute and set the **Outgoing Claim Type** to "user_principal." The "uid" identifies the authenticated user in the claim sent to the applications. 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).

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.

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.

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.

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.

The credential must be of a domain user who is a member of the local administrator group if the `ADSecurityGroupUpdate` registry key in AW is zero. If the `ADSecurityGroupUpdate` registry key is set to 1, then the user must be available in the Config security group under the instance OU. These credentials must be valid on all CCE components in your deployment (routers, PGs, AWs, and so on).

Note Every time the Active Directory credentials are updated, the credentials configured here must be updated as well.

d) Click **Save**.

Set Up the External HDS for Single Sign-On

If you have an external HDS in 2000 Agent deployments, manually associate it with a default Cisco IdS by performing the following instructions.

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**.

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 In Packaged CCE 4000 or 12000 Agent deployments:

- 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.

In Packaged CCE 2000 Agent deployments, you must manually associate an external HDS with a default Cisco Identity Service (Cisco IdS). For more information, see Set Up the External HDS for Single Sign-On, on page 206.

Step 1 In the Unified CCE Administration, choose **Overview** > **Infrastructure Settings** > **Device Configuration** > **Identity Service**.

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 Click **Save**.

Step 7 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. Regenerating the certificates for token signing requires all agents to logout and relogin to the Cisco Finesse desktop. Therefore ensure that you plan for downtime to your Contact Center before you regenerate the public-private key pair that Cisco IdS uses to authenticate agents. After you regenerate the new key pair, you must reboot Cisco IdS so that the agents can relogin to their applications. Ensure that you CA sign the regenerated public-private key pair, if required and then reupload it to the clients that are dependent on the public-private key pair. For example, if you are using the digital channels service, you must reupload the private-public key pair on to Control Hub for your agents to resume with the Manage Digital Channel gadget. For instructions, see *Provision Webex Connect digital services for your organization*.
- 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 8 Click **Save**.

Step 9 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 10 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 11 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 12 Click **Identity Service Settings**.

Step 13 Click **Troubleshooting** to perform some optional troubleshooting.

Step 14 From the **Log Level** drop-down list, set the local log level by choosing **Error**, **Warning**, **Info** (the default), **Debug**, or **Trace**.

Step 15 To receive errors in Syslog format, enter the name of the Remote Syslog Server in the **Host** (Optional) field.

Step 16 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.

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.

Step 1 In the Unified CCE Administration, navigate to **Features** > **Single Sign-OnOverview** > **Infrastructure Settings** > **Device Configuration** > **Identity Service**.

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.

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, Edge Chromium (Microsoft Edge), 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. 2. Click the **Customize and control Google Chrome** icon. 3. Click **Settings**. 4. In the **On startup** section of the **Settings** page, verify that the **Open the New Tab page** option is selected.
Edge Chromium (Microsoft Edge)	1. Open Microsoft Edge. 2. Click the **Settings and more (Alt+F)** (...) icon. 3. Click **Settings**. 4. On the **Settings** page, click **On startup**, and verify that the **Open a new tab** radio button is selected.
Firefox	1. Open Firefox. 2. Click the **Open menu** icon. 3. Click **Options**. 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.

Step 1 In Unified CCE Administration, navigate to **Overview** > **Bulk Import**.

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. 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.

When the bulk job completes, the agents and supervisors are enabled for SSO and their usernames are updated. You can open an individual user's record in the Agent tool in Unified CCE Administration to see the changes.

What to do next

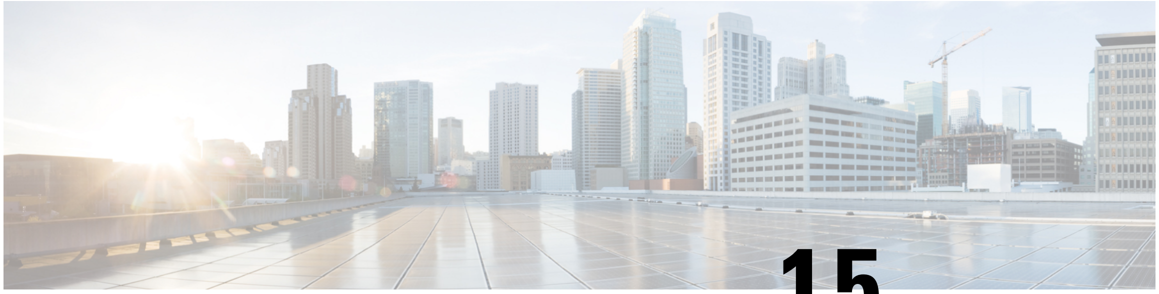
After all of the agents and supervisors in your deployment are migrated to SSO accounts, you can enable SSO globally in your deployment.

Related Documentation

Refer to the following documents and other resources for more details about single sign-on.

See this information	Located here	For these details
Solution Design Guide for Cisco Packaged Contact Center Enterprise	https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/tsd-products-support-series-home.html	Design considerations and guidelines for deploying the Cisco Packaged CCE system.
Virtualization for Cisco Packaged CCE	https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/uc_system/virtualization/pcce_virt_index.html	Information about deploying Packaged CCE (including single sign-on) on VMware.
Release Notes for Cisco Packaged Contact Center Enterprise Solution	https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-release-notes-list.html	New features and changes for this release of the Packaged CCE solution.
Contact Center Enterprise Compatibility Matrix	https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-device-support-tables-list.html	Packaged CCE requirements.
Unified CCE Administration Single Sign-On Tool	Online help	Changes to support single sign-on.

See this information	Located here	For these details
System Inventory Tool	This guide.	Information related to adding SSO-compatible components to the inventory.



CHAPTER 15

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.

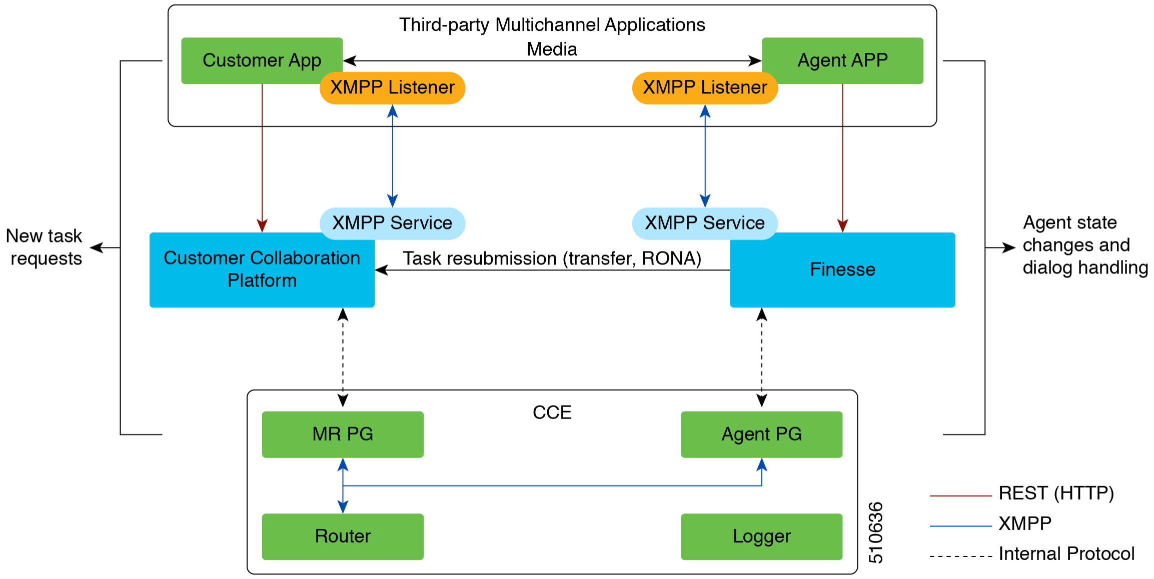
Enterprise Chat and Email provides universal queue out of the box. Third-party multichannel applications can use the universal queue by integrating with CCE through the Task Routing APIs.

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 20: 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 the Unified CCE PG on one side.
- 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.

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 Tools, on page 237

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 Whether Customer Collaboration Platform re-queues or discards the task when Customer Collaboration Platform recovers from a failure. 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.
dialogLogoutAction Whether active tasks are closed or transferred when an agent signs out or loses presence. 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.

API Parameter or Setting	API/Tool	Possible Values	
		Interactive Task/MRD	Non-interactive Task/MRD
Start Timeout 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. 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 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 Packaged Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-technical-reference-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.

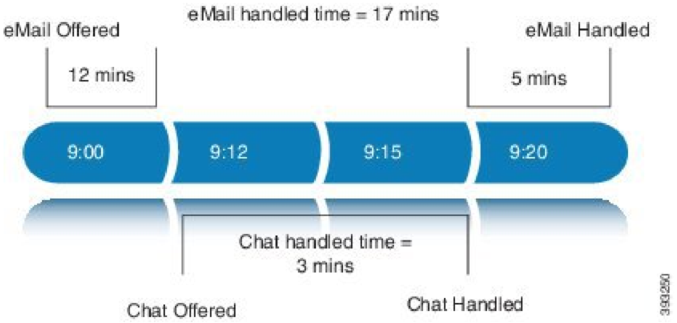
  The agent, CCE task, and Finesse dialog state in the interrupted MRD change to INTERRUPTED.

  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.

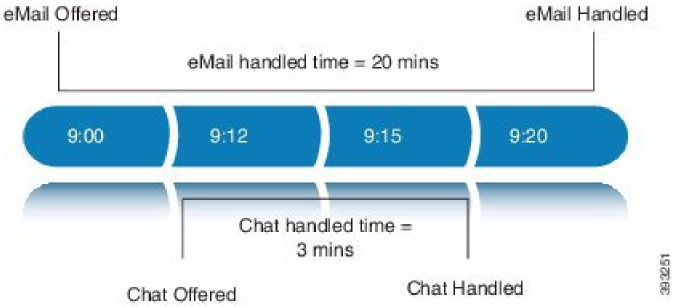


- **Ignore:** When an agent is interrupted by another task while working on a task in an interruptible MRD, Finesse ignores the interrupt event.

  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.

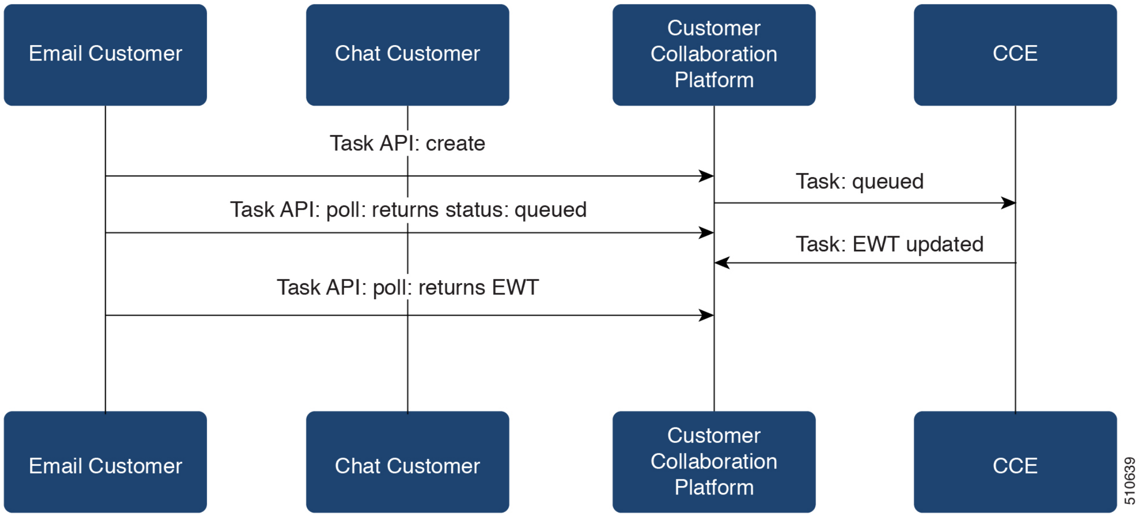
Task Routing API Request Flows

Task Routing API Basic Task Flow

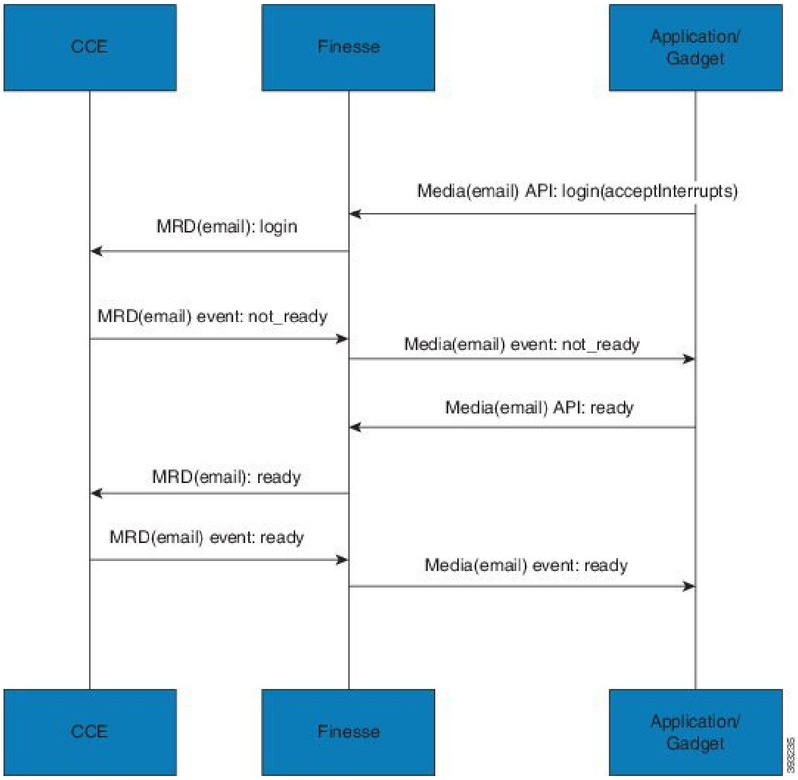
This topic provides the Customer Collaboration Platform and Finesse API calls and events when an active email task is interrupted by a chat request.

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.

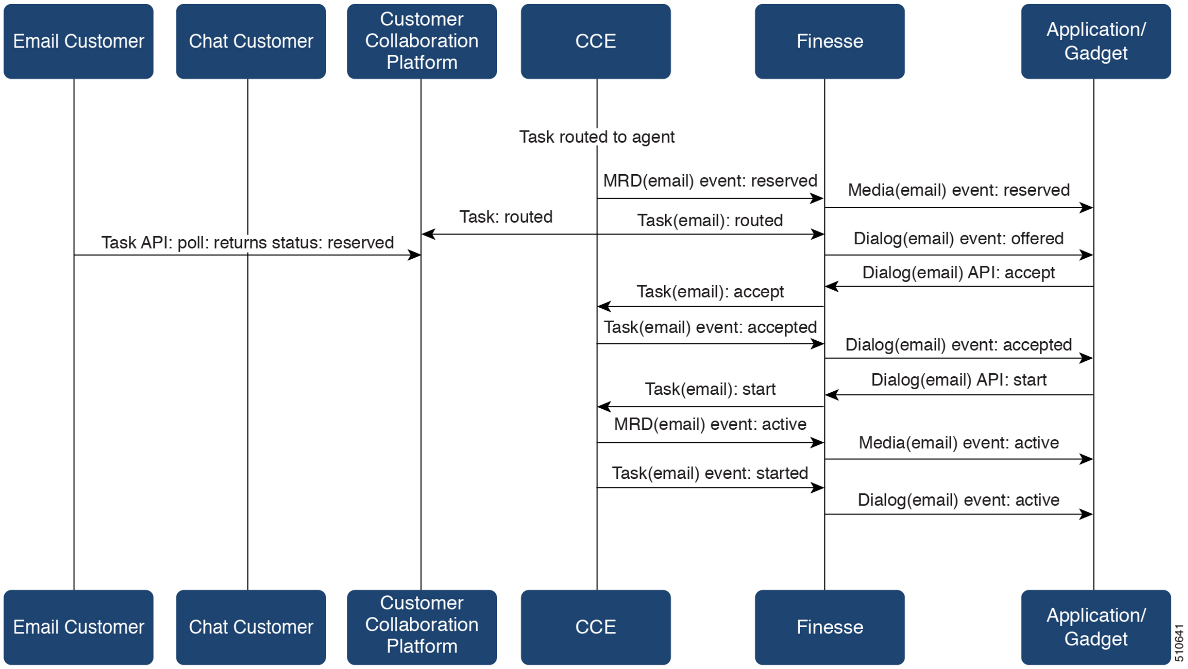
1. The email application submits a new email task request to CCE, and polls for status and Estimated Wait Time (EWT).


Email Customer
Chat Customer
Customer Collaboration Platform
CCE
Task API: create
Task: queued
Task API: poll: returns status: queued
Task: EWT updated
Task API: poll: returns EWT
Email Customer
Chat Customer
Customer Collaboration Platform
CCE
510639

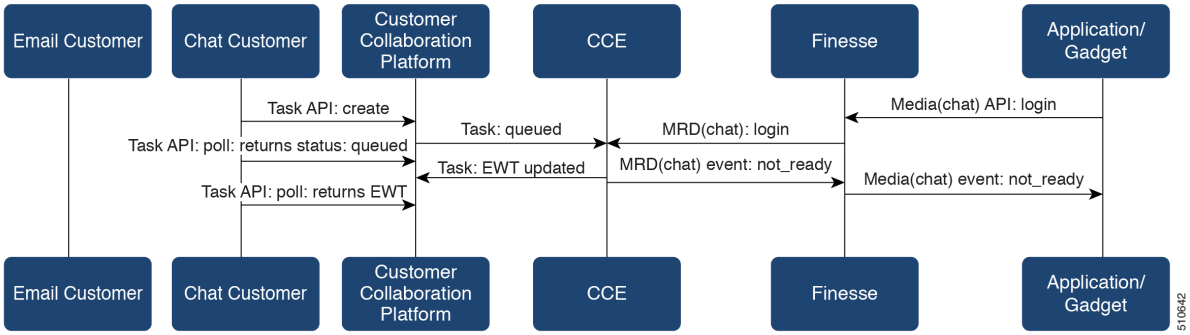
2. An agent signs in to the email MRD and changes state to Ready.



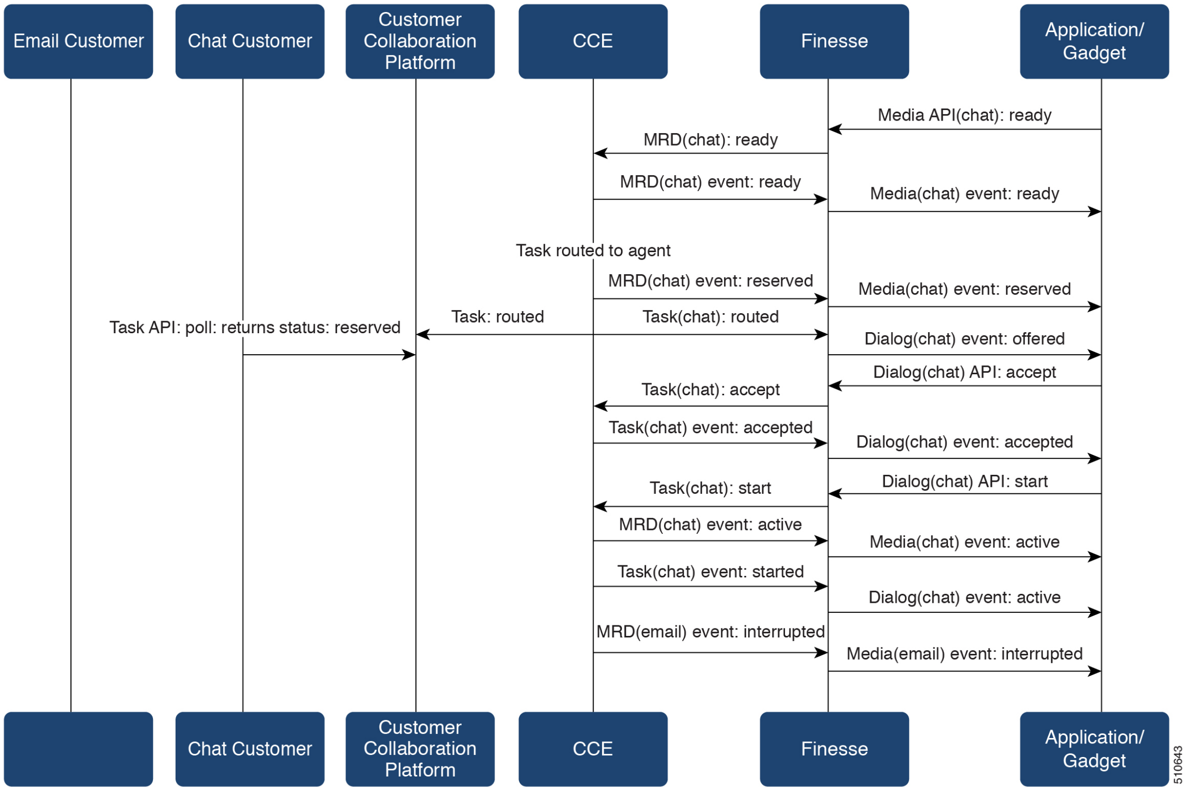
3. CCE assigns the agent the email task. The Call and ECC variables used to create the task are included in the dialog's media properties, and contain information such as the handle to the email. The variables can be used to reply to the email. The agent starts work on the email dialog in Finesse.


Email Customer
Chat Customer
Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
Task routed to agent
MRD(email) event: reserved
Media(email) event: reserved
Task: routed
Task(email): routed
Task API: poll: returns status: reserved
Dialog(email) event: offered
Dialog(email) API: accept
Task(email): accept
Task(email) event: accepted
Dialog(email) event: accepted
Dialog(email) API: start
Task(email): start
MRD(email) event: active
Media(email) event: active
Task(email) event: started
Dialog(email) event: active
Email Customer
Chat Customer
Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
510641

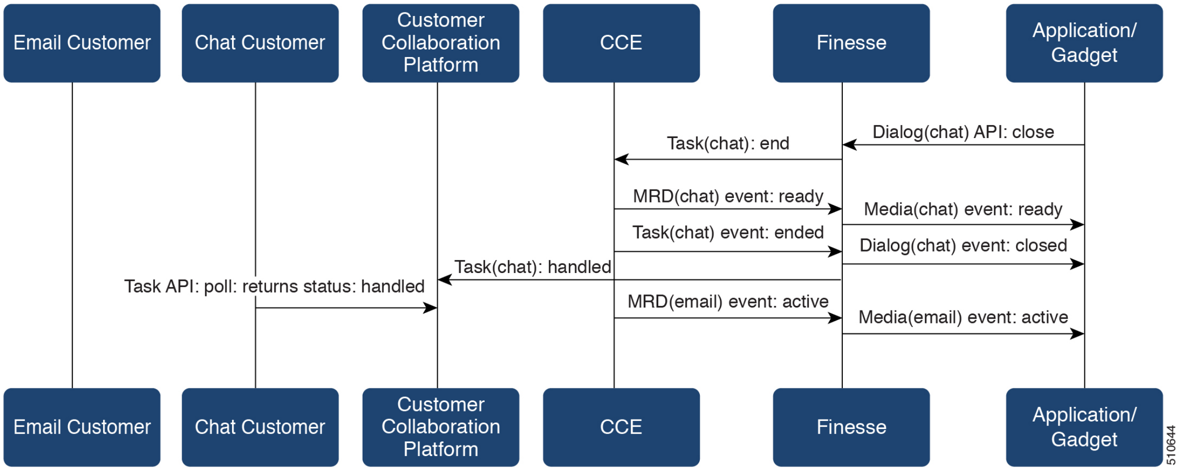
4. The chat application submits a new chat request, and polls for status and EWT. The same agent logs into the chat MRD.



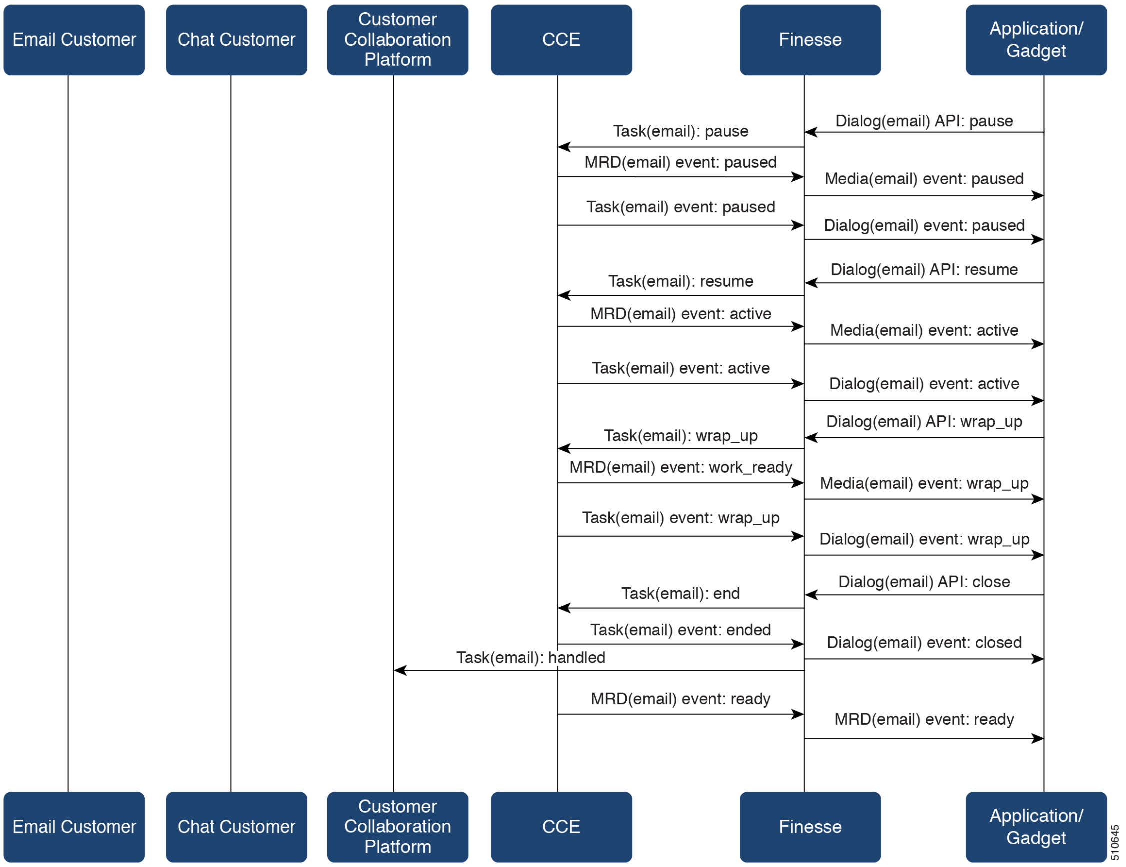
5. The agent changes state to Ready in the chat MRD. CCE assigns the chat task to the agent. The Call and ECC variables used to create the task are included in the dialog's media properties, and contain information such as the chat room URL. The variables can be used to join the chat room with the customer. The agent starts the chat dialog in Finesse. The Email dialog is interrupted.


Email Customer
Chat Customer
Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
Media API(chat): ready
MRD(chat): ready
MRD(chat) event: ready
Media(chat) event: ready
Task routed to agent
MRD(chat) event: reserved
Media(chat) event: reserved
Task: routed
Task(chat): routed
Task API: poll: returns status: reserved
Dialog(chat) event: offered
Dialog(chat) API: accept
Task(chat): accept
Task(chat) event: accepted
Dialog(chat) event: accepted
Dialog(chat) API: start
Task(chat): start
MRD(chat) event: active
Media(chat) event: active
Task(chat) event: started
Dialog(chat) event: active
MRD(email) event: interrupted
Media(email) event: interrupted
Chat Customer
Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
510643

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.

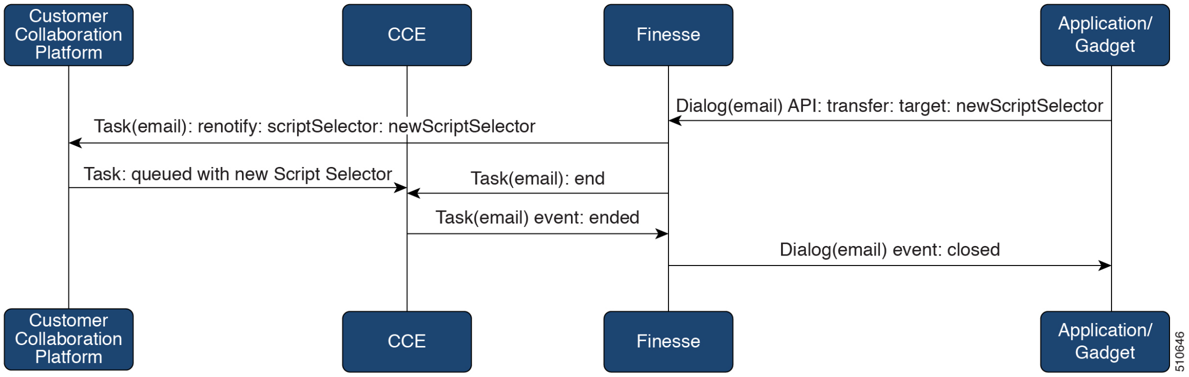


7. The agent continues working on the email dialog, including pausing, resuming, and wrapping up the dialog. The agent closes the dialog. Finesse sends a handle event to Customer Collaboration Platform for the email task. The application is responsible for sending the email reply to the customer.


Email Customer
Chat Customer
Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
Dialog(email) API: pause
Task(email): pause
MRD(email) event: paused
Media(email) event: paused
Task(email) event: paused
Dialog(email) event: paused
Dialog(email) API: resume
Task(email): resume
MRD(email) event: active
Media(email) event: active
Task(email) event: active
Dialog(email) event: active
Dialog(email) API: wrap_up
Task(email): wrap_up
MRD(email) event: work_ready
Media(email) event: wrap_up
Task(email) event: wrap_up
Dialog(email) event: wrap_up
Dialog(email) API: close
Task(email): end
Task(email) event: ended
Dialog(email) event: closed
Task(email): handled
MRD(email) event: ready
MRD(email) event: ready
Email Customer
Chat Customer
Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
510645

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.

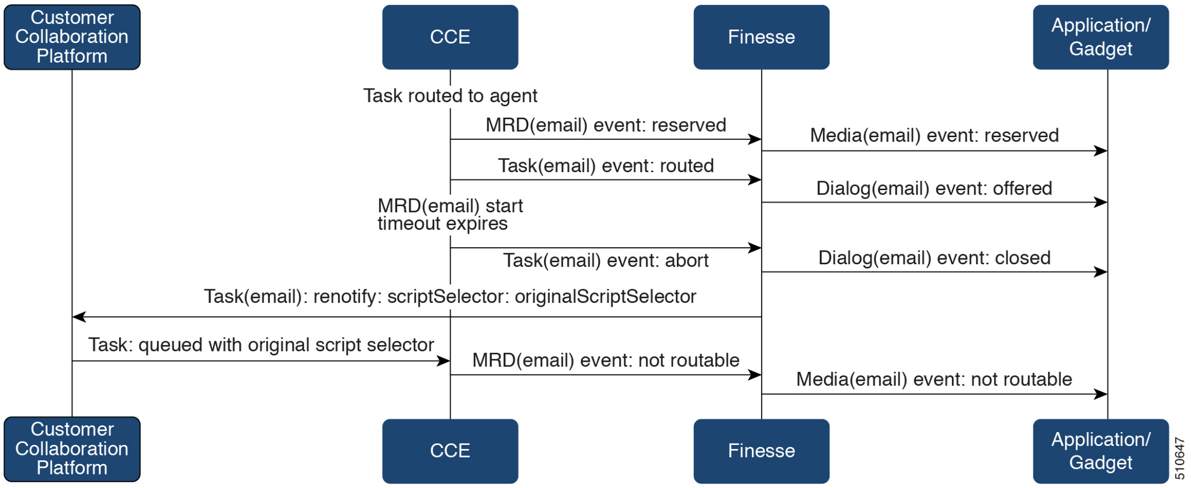

Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
Dialog(email) API: transfer: target: newScriptSelector
Task(email): renotify: scriptSelector: newScriptSelector
Task: queued with new Script Selector
Task(email): end
Task(email) event: ended
Dialog(email) event: closed
Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
510646

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.


Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
Task routed to agent
MRD(email) event: reserved
Media(email) event: reserved
Task(email) event: routed
Dialog(email) event: offered
MRD(email) start timeout expires
Task(email) event: abort
Dialog(email) event: closed
Task(email): renotify: scriptSelector: originalScriptSelector
Task: queued with original script selector
MRD(email) event: not routable
Media(email) event: not routable
Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
510647

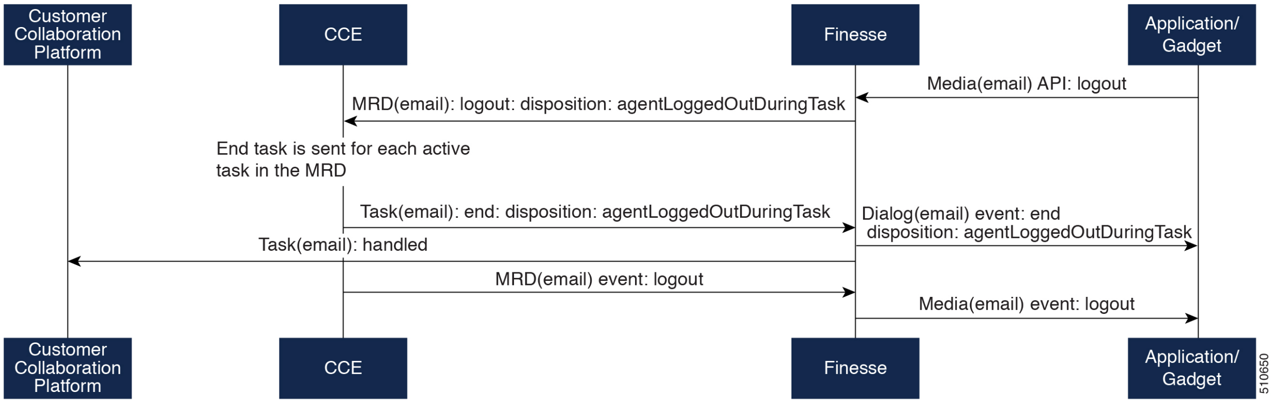
1. The task is routed to an agent, and the dialog is offered to the agent.
2. The Media Routing Domain's Start Timeout threshold expires.
3. CCE instructs Finesse to end the dialog. Finesse puts the dialog in the CLOSED state, with the disposition code CD_RING_NO_ANSWER. Finesse does not send a handled 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

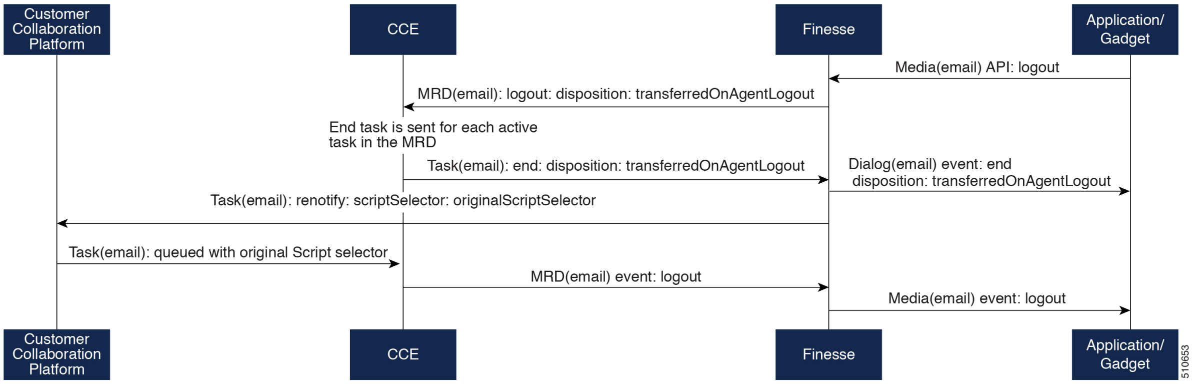
This illustration provides the Customer Collaboration Platform and Finesse API calls and events when agents are set to have assigned tasks closed on sign out.


Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
Media(email) API: logout
MRD(email): logout: disposition: agentLoggedOutDuringTask
End task is sent for each active task in the MRD
Task(email): end: disposition: agentLoggedOutDuringTask
Dialog(email) event: end disposition: agentLoggedOutDuringTask
Task(email): handled
MRD(email) event: logout
Media(email) event: logout
Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
510650

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.


Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
Media(email) API: logout
MRD(email): logout: disposition: transferredOnAgentLogout
End task is sent for each active task in the MRD
Task(email): end: disposition: transferredOnAgentLogout
Dialog(email) event: end disposition: transferredOnAgentLogout
Task(email): renotify: scriptSelector: originalScriptSelector
Task(email): queued with original Script selector
MRD(email) event: logout
Media(email) event: logout
Customer Collaboration Platform
CCE
Finesse
Application/ Gadget
510653

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. 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. **Automatic transfer request from Finesse** (for transfer on sign out or RONA): Results in a lost transfer request. **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. 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. 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: **Customer Collaboration Platform failure:** NOTIFICATION_CCE_ CUSTOMERCOLLABORATIONPLATFORM_SYSTEM_FAILURE **MR connection failure:** NOTIFICATION_CCE_CONNECTION_LOST **Offered and active tasks:** No impact.
Customer Collaboration Platform	Customer Collaboration Platform overruns the new task queue limit. 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. **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 **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. **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. **Queued tasks:** No impact. **Offered tasks:** These tasks RONA because the agent cannot accept them. **Active tasks:** These tasks fail over to the other Finesse server and are recovered on that server. **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 **Automatic or agent transfer requests:** No impact	**Queued tasks:** No impact. **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. **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. If the tasks are transferred, the disposition code is CD_TASK_TRANSFERRED_AGENT_LOGOUT. 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 **Automatic or agent transfer requests:** No impact	**Queued tasks:** No impact. **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. **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 **Automatic transfer requests from Finesse** (for transfer on logout or RONA): Results in lost transfer requests. **Agent transfer request:** The request fails, and Finesse retains the task.	**Queued tasks:** No impact. **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. **Active tasks:** These tasks fail over to the other Finesse server and are recovered on that server. **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 **Automatic or agent transfer requests:** Results in lost transfers.	**Queued tasks:** No impact **Offered and active tasks:** These tasks are lost

Task Routing Setup

Initial Setup

Step	Task	Notes
Set up CCE		
1	Set up the MR PG and PIM for Customer Collaboration Platform. See Set up the Media Routing PG and PIM, on page 236.	

Step	Task	Notes
2	Add Customer Collaboration Platform as an External Machine in the System Inventory. See Add Customer Collaboration Platform as an External Machine, on page 237.	The system configures the following settings automatically in Customer Collaboration Platform Administration: • Enables and configures the **CCE Multichannel Routing settings**. • Configures the Task feed and the associated campaign and Connection to CCE notification needed for the Task Routing feature.
3	Configure the following in Unified CCE Administration: • Media Routing Domains • Call types • Dialed numbers • Skill groups or precision queues • ECC variables • Agent desk settings See Unified CCE Administration Tools, on page 237.	
4	Increase the TCDTimeout registry key value, if you are using precision queues and will be submitting potentially long tasks, like email. See Increase TCDTimeout Value, on page 239.	
5	Create routing scripts See Create Routing Scripts for Task Routing, on page 239.	
Create Custom Customer Collaboration Platform and Finesse Applications		
6	Create the Customer Collaboration Platform multichannel application to begin task requests. See Sample Customer Collaboration Platform HTML Task Application, on page 240.	
7	Create the Finesse applications to manage nonvoice agent and dialog states. See Sample Finesse Code for Task Routing, on page 240.	
Set up Finesse		

Step	Task	Notes
8	Upload the Finesse applications to the desktop layout (optional). See the *Cisco Finesse Administration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-user-guide-list.html.	

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

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. 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 for the unused Multichannel peripheral that you identified in Step 1.
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 Accept defaults and click **Next** until the Setup Complete screen opens.

Step 8 At the Setup Complete screen, check **Yes** to start the service. Click **Finish**.

Step 9 Click **Exit Setup**.

Step 10 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 Unified CCE 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.

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 Machine**.

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 Click **Save**.

Unified CCE Administration Tools

This topic explains the Unified CCE Administration tools you need to configure Task Routing.

For details on the procedures for these steps, refer to the Unified CCE Administration online help.

Step 1 Sign in to Unified CCE Administration.

Step 2 Configure the following:

Item to Configure	**Details**
Media Routing Domains	Create an MRD for each type of task that the custom application submits to CCE (email, chat, and so on).
Call Types	Create call types for Task Routing.
Dialed Numbers	Create dialed numbers for Task Routing. Add the numbers or strings that the third-party multichannel application will use when submitting task requests. • For **Routing Type**, select Customer Collaboration Platform. • For **Media Routing Domain**, select one of the Task Routing MRDs you created. • For **Call Type**, select a call type that you created for Task Routing. **Important** Each dialed number must be associated with a call type. Default call type is not supported for tasks submitted with Task Routing APIs.
Skill Groups	Configure either skill groups or precision queues. If you configure skill groups: • For **Media Routing Domain**, select one of the Task Routing MRDs you created. • Assign agents to the skill group.
Precision Queues	Configure either skill groups or precision queues. If you configure precision queues: • For **Media Routing Domain**, select one of the Task Routing MRDs you created. • Associate agents with attributes that are part of the precision queue steps.
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. **Note** Arrays are not supported with the Task Routing feature. 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.

Item to Configure	Details
Network VRU Script	Create a Network VRU Script that references the Network VRU (MR_Network_VRU). The Network VRU Script is used to return estimated wait time to customers. You can accept the default values. 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. For more information on writing scripts to return estimated wait time, see the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.
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 delete the existing value. 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 to 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 Unified CCE Rogger.

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 *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-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.

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/.

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
- 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 *Cisco Packaged Contact Center Enterprise Reporting User Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html for information about multichannel reporting data.

CHAPTER 16

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 toward agents.

Note The Dialogflow ES GCP project trial version should not be used in a production environment.

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 that is 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.

For accessing VAV functionality, Cisco Virtualized Voice Browser (VVB) uses one of the following connectors to leverage AI services:

Cloud-based connector: VVB uses a cloud-based connector to connect to the Cisco CCAI service. This service is enabled through the *VirtualAgentVoice* element of Cisco Unified Call Studio. VAV currently supports Google's Dialogflow CX service via cloud-based connector.

Premise-based connector: VVB uses an original connector to connect to the Google Dialogflow service. This service is enabled through the *Dialogflow* or *DialogflowCX* elements of Cisco Unified Call Studio. VAV currently supports Google's Dialogflow ES and CX services via premise-based connector.

Note You can configure the Virtual Agent–Voice (VAV) feature of VVB 12.5(1) keeping the Cisco Unified Contact Center Enterprise Controller in 12.0 version (as in multistage upgrade). However, in this case, the configuration user interface for the VAV service account will not be available in the Cisco Unified Contact Center Enterprise Administration. So, System Administrators can use the *Command Execution Pane* for such configurations.

For more information, see the *Command Execution Pane* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Onboarding Experience

From 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 the Unified CCE solution and CCE Administration Portal in the 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) and above	Cisco-billed (OEM)	Webex Control Hub	AnalyzeContent (V2)
12.5(1) and above	Vendor-billed (non-OEM)	NOAMP/CCEAdmin	DetectIntent (V2)

The following table lists the support for VAV Dialogflow CX and VAV Dialogflow ES via cloud-based or premise-based connectors for OEM and non-OEM customers:

	VAV Dialogflow CX (OEM)	VAV Dialogflow ES (OEM)	VAV Dialogflow CX (Non-OEM)	VAV Dialogflow ES (Non-OEM)
Via cloud-based connector	Yes (12.6(2) and above)	No	No	No
Via premise-based connector	Yes (12.6(1) and above)	Yes (12.5(1) and above)	No	Yes (12.5(1) and above)

VAV via Cloud-Based Connector

Overview

Virtual Agent–Voice via cloud-based connector leverages Cisco's cloud-based Artificial Intelligence (AI) and Natural Language Understanding (NLU) services for designing virtual voice agents and creating complex IVR call flows.

The Webex Contact Center AI (Webex CCAI) services platform enables integration with speech-based services from different vendors. On the premise side, VVB interfaces with the Orchestrator service and connects to the CCAI service via cloud-based connector. This service is enabled through the *VirtualAgentVoice* element of Cisco Unified Call Studio.

For details of the *VirtualAgentVoice* element, refer to the *Element Specifications for Cisco Unified CVP VXML Server and Call Studio* guide at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-programming-reference-guides-list.html.

The partial response feature is supported via cloud-based connector. It notifies the caller that their request is being processed. By default, the partial prompt will stop playing when the final response is received from the webhook. For more information, see the Configure Partial Response in Dialogflow CX article.

Important Considerations

Consider the following before configuring VAV via a cloud-based connector:

- Supported codec is u-law.
- This feature is available with Cisco subscription services only.
- Google performs the voice activity detection.
- If the call gets transferred to a real agent, then the transcript of the voice conversation between the customer and the virtual voice agent is displayed in the *Transcript* widget in the Agent Desktop (provided that the *Transcript* widget is configured on the Agent Desktop).
- Non barge-in prompt is supported only until the control gets transferred to the cloud.

Prerequisites

To configure VAV via cloud-based connector, you should have completed the following:

- The allowed list in your customer's network must include the following URLs:
  - U2C (port 443): US region - https://u2c-a.wbx2.com/u2c/api/v1/user/catalog
  - U2C (port 443): Asia region - https://u2c-r.wbx2.com/u2c/api/v1/user/catalog
  - U2C (port 443): Europe region - https://u2c-k.wbx2.com/u2c/api/v1/user/catalog
  - Orchestrator (port 443): https://insight-orchestrator.wxcc-us1.cisco.com and https://insight-orchestrator.prod-us1.rtmsprod.net
  - ID broker (port 443): https://idbroker.webex.com
  - WS Connector (port 443): https://wsconnector.wxcc-us1.cisco.com and https://wsconnector.prod-us1.rtmsprod.net
- The allowed list of your customer's network must include the following group of URLs:
  - accounts.google.com/o/oauth2
  - *.cisco.com
  - *.ciscoservice.com

    - *.ciscoccservice.com
    - *.ciscospark.com
    - *.cloud.google.com/dialogflow
    - *.google.com
    - *.googleapis.com
    - *.gcr.io
    - *.rtmsprod.net
    - *.webex.com
    - *.wbx2.com

- The customer's CX Agent ID and GCP Project ID are 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 is mapped with Cisco's GCP partner projects.
    - Control Hub credentials and Hybrid Org are generated.
- Access the associated location settings for regions at Google at https://cloud.google.com/dialogflow/cx/docs/concept/region
- Add non-proxy hosts using VVB CLI commands. For more information, refer to the *Command Line Interface* > *Set Commands* section in the *Operations Guide for Cisco Virtualized Voice Browser* at https://www.cisco.com/c/en/us/support/customer-collaboration/virtualized-voice-browser/products-maintenance-guides-list.html.

CCE (on premise):

- Cisco VVB is upgraded to 12.6(2) using a generic COP.
- 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.
- Average bandwidth consumption per call is 106 kbps.
- The new Cisco Unified Call Studio application with the *VirtualAgentVoice* element is deployed. You can download and install the latest patch from https://software.cisco.com/download/specialrelease/c359e375005563ceec2081c9151b482e.

For details of the *VirtualAgentVoice* element, refer to the *Element Specifications for Cisco Unified CVP VXML Server and Call Studio* guide at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-customer-voice-portal/products-programming-reference-guides-list.html.

- Ensure that the Packaged CCE AW, Cloud Connect, and VVB components have access to Webex services to use the Virtual Agent–Voice via cloud-based connector.

CCAI (in Cisco Webex Control Hub):

- Service provider-specific **Integration Connector** is configured. For more details, see the article https://help.webex.com/en-us/article/9zvadfb/Configure-Google-CCAI-connector-for-Webex-Contact-Center.
- Design Google Dialogflow CX Agents. For more information, refer to Google documentation at https://cloud.google.com/dialogflow/cx/docs/.
- CX conversational profile ID is created using credentials provided by Cisco via email. For detailed steps, see the section *Create a Conversation Profile using Google Cloud SDK*.

  Note En-US global customers can create the conversational profile ID through the UI at https://agentassist.cloud.google.com/projects. While creating the profile, enable the option **Enable virtual agent** under **Choose to use Dialogflow**. This profile ID can be used both for the Agent Answers and VAV features.

  This conversation profile URL must be used to create the CCAI configuration.

- CCAI configuration is created at https://admin.webex.com/. For detailed steps, refer to the Create a Contact Center AI configuration article.

For further assistance, contact Cisco Support.

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.

Step 1 Ensure that the Cloud Connect publisher and subscriber nodes are installed.

For more information, see the *Create VM for Cloud Connect Publisher* and *Create VM for Cloud Connect Subscriber* sections in *Cisco Packaged Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-installation-guides-list.html.

Step 2 Register and add Cloud Connect in Control Hub. For more details, see https://help.webex.com/en-us/article/n24wo0fb/Register-Cloud-Connect.

Step 3 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 Administration* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html..

Step 4 Configure Cloud Connect with CVP and VVB devices in Unified CCE Administration. For details, see *Configure Cloud Connect* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Note After an upgrade, Cloud Connect has to be reconfigured with VVB devices.

Step 5 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 6 View the default CCAI configuration (created as part of CCAI prerequisites). If required, synchronize the configuration (using *Sync* option) in Unified CCE Administration.

For details, see the *Manage Features* > *Contact Center AI Configuration* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Step 7 Design Google Dialogflow CX application.

For more information, refer to Google documentation at https://dialogflow.cloud.google.com/cx/projects.

For further assistance, contact the Cisco TAC team.

Hybrid IVR with VAV via Cloud-Based Connector

Dialogflow CX enables contact centers to use cloud-based AI features via Cisco Cloud. With Cisco's Hybrid IVR functionality, customers who have on-premises applications can leverage their traditional ASR/TTS/CRM integrations, along with cloud-based Dialogflow CX AI capabilities. They can select a few nodes or sections of their application to be processed in the cloud and few nodes to be processed on-premises. For example, in an application, OTP generation can be performed on-premises, while other tasks can be processed in the cloud.

Hybrid IVR with Dialogflow CX provides the flexibility to switch from the cloud services to the on-premises application, to use the on-premises subroutine calls, and perform the required operations. After these operations are done, the flow can go back to the cloud services and resume from where it left.

The switching between the cloud services and the on-premises application is achieved by defining a custom payload with an event and associated parameters. This custom payload is passed to the on-premises application. When the flow resumes in the cloud, Dialogflow CX uses the event and parameters that are received from the on-premises application.

The *VirtualAgentVoice* element of Cisco Unified Call Studio is used to engage the Google Dialogflow CX services with Hybrid IVR functionality via Cisco Cloud.

To configure the Hybrid IVR functionality with Dialogflow CX, see Configure Hybrid IVR, on page 249.

Note

- Hybrid IVR and IVR transcription functionalities are available *only* with Dialogflow CX from 12.6(2) and higher.
- The *VirtualAgentVoice* element supports both Cisco DTMF and Nuance adapters.

Configure Hybrid IVR

Task flow to configure Hybrid IVR with Google Dialogflow CX agent via Cisco Cloud:

Step 1 In the Google Dialogflow CX console, select the project and the agent.

Step 2 In the Google Dialogflow CX agent screen, under the **Build** tab, select the required flow and the required page (**Start**/**End Flow**/**End Session**) in this flow, in which a fulfillment is needed from the on-premises application. The details of the selected page appears.

Step 3 Under **Routes** section, define a route and conditions that satisfy the custom exit criteria, for triggering the transition.

Step 4 In this route, under **Fulfillment** section, click **Add dialogue option** and select **Custom payload**.

Note Only **Custom payload** has to be defined here. Do not add any other dialogue options.

Step 5 Add the custom payload of type *Execute_Request* that defines the action and elements to be filled in the Unified Call Studio *VirtualAgentVoice* element.

The format for defining the custom payload is as follows:

```
{"Execute_Request":{
                   "Event_Name": "<Name of the event>",
                   "Data" : {
                            "Params":{
                                     "<param1 name>": "<param1 value>",
                                     "<param2 name>": "<param2 value>"
                                     }
                            }
                   }
}
```

Note Ensure that you map this event name with the same event name in Unified Call Studio *VirtualAgentVoice* element for decision mapping.

Step 6 Select **Page** under the **Transition** section, to set the transition to the same page when the flow resumes.

Step 7 For re-entry from the on-premises application to cloud, create an event handler. This event name is to be provided to the *VirtualAgentVoice* element during re-entry. For more information, refer to https://cloud.google.com/dialogflow/cx/docs/reference/rest/v3/EventHandler.

It’s the responsibility of the Dialogflow CX developer to create this event handler. Otherwise, the application fails.

VAV via Premise-Based Connector

Virtual Agent–Voice via premise-based connector leverages Google's cloud-based Artificial Intelligence (AI) and Natural Language Understanding (NLU) services for designing virtual voice agents and creating complex IVR call flows.

VAV currently supports two flavors of Google Dialogflow services via premise-based connector:

VAV 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. These agents 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.

VVB uses a premise-based connector to connect to the Google Dialogflow service. This service is enabled through the *DialogflowCX* element of Cisco Unified Call Studio.

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.

Note

- Only u-law codec is supported.
- This feature is available with Cisco subscription services only.

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)*.

Note Ensure that the Packaged CCE AW and Cloud Connect Components have access to Webex services and the VVB components have access to both Webex and Google services to use the Virtual Agent-Voice for Dialogflow CX via premise-based connector.

To configure Google Dialogflow CX, you should have completed the following procedures:

- The customer's CX Agent ID and GCP Project ID are 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 is mapped with Cisco's GCP partner projects.
  - 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 252.

  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 and *VirtualAgentVoice* 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 and *VirtualAgentVoice* 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 the Google Dialogflow CX Agent with Cisco Packaged CCE solution.

Step 1 Ensure that the Cloud Connect publisher and subscriber nodes are installed.

For more information, see the *Create VM for Cloud Connect Publisher* and *Create VM for Cloud Connect Subscriber* sections in *Cisco Packaged Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-installation-guides-list.html.

Step 2 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 Administration* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html..

Step 3 Configure Cloud Connect with CVP and VVB devices in Unified CCE Administration. For details, see *Configure Cloud Connect* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Step 4 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 5 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 https://help.webex.com/en-us/article/9zvadfb/Configure-Google-CCAI-connector-for-Webex-Contact-Center 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 6 View the default CCAI configuration. If required, synchronize the configuration (using *Sync* option) in Unified CCE Administration.

For details, see the *Manage Features > Contact Center AI Configuration* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Step 7 Design Google Dialogflow CX Agents. For more information, refer to Google documentation at https://dialogflow.cloud.google.com/cx/projects.

Step 8 Configure a welcome event for all Google Dialogflow CX Agents. For details, see Create a Welcome Event, on page 255.

Note

- You can configure custom events to override the default welcome event. Provide the element data *event_name* in the Dialogflow CX element and ensure that you configure the same name in the CX Agent to start the flow. For instructions, see *Configure custom events in Dialogflow CX*.
- For further assistance, contact Cisco Support.

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.

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/Ql036mwSUa3cjg",
"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 Create the conversation profile using REST API for Dialogflow by using Postman:

a) Add headers: Content-type Application/json.

b) Authorization: Bearer - add the token generated with the Google command.

c) Add the method as POST.

d) In the URL section, add the address after replacing the regionId and the projectId appropriately. The address is in the following format: https://<regionId>-dialogflow.googleapis.com/v2beta1/projects/<projectId>/locations/<regionId>/conversationProfiles

e) In the body section, choose RAW and JSON.

f) In the **Agent** tag, use the appropriate regionId, projectId, and agentId.

```
{
"name": "TACCXTest",
"automatedAgentConfig": {
"agent": "projects/<projectId>/locations/<regionId>/agents/<agentId>”
},
"displayName": "TACCXTest",
"humanAgentAssistantConfig": {
"messageAnalysisConfig": {
"enableEntityExtraction": true,
"enableSentimentAnalysis": true
}
}
}
```

Example response:

```
{
"name": "projects/tacprojectrtp2020/locations/us-central1/conversationProfiles/dCv4lC1uQlO36mwSUa3cjg",
"displayName": "TACCXTest",
"automatedAgentConfig": {
"agent":
"projects/tacprojectrtp2020/locations/us-central1/agents/5cca975a-bbb3-40d0-aa2a-1bf453d9bf5c/environments/draft"

    },
"humanAgentAssistantConfig": {
"notificationConfig": {},
"messageAnalysisConfig": {
"enableEntityExtraction": true,
"enableSentimentAnalysis": true
        }
    },
"languageCode": "en-US"
}
```

This is the conversation profile that you obtain in the example. *projects/tacprojectrtp2020/locations/us-central1/conversationProfiles/dCv4lC1uQlO36mwSUa3cjg.* You can use this profile while creating the Control Hub configuration.

For more information, see https://cloud.google.com/sdk/gcloud/reference.

Step 8 In a new browser tab, open https://agentassist.cloud.google.com/ and select the appropriate project. The list of profiles is displayed.

Step 9 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.

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.

Note To override the default welcome event, provide the element data *event_name* in the DialogflowCX element. The same name must be configured in the CX Agent to start the flow.

VAV for Dialogflow ES

Note The Diaglogflow ES GCP project trial version should not be used in a production environment.

Dialogflow ES via VVB (OEM Users)

Overview

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
12.5 and higher	12.5 and higher	NA	11.6/12.0 with A2Q approval	11.6/12.0 with A2Q approval

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.

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 *Create VM for Cloud Connect Publisher* and *Create VM for Cloud Connect Subscriber* sections in *Cisco Packaged Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-installation-guides-list.html.

Step 3 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 Administration* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html..

Step 4 Configure Cloud Connect with CVP and VVB devices in Unified CCE Administration. For details, see *Configure Cloud Connect* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Step 5 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 6 View the default CCAI configuration (created in step 1). If required, synchronize the configuration (using *Sync* option) in Unified CCE Administration.

For details, see the *Manage Features > Contact Center AI Configuration* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-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.

Dialogflow ES via VVB (Non-OEM Users)

Overview

From 12.5(1) release, non-OEM users use NOAMP in Unified CCE solution, and CCE Administration Portal in Packaged CCE solution for VAV onboarding.

The following table lists the onboarding channel and Google APIs used for non-OEM customers:

Release	Services Billing	Onboarding	Google APIs
12.6(1) and 12.6(2)	Vendor-billed (non-OEM)	NOAMP/CCEAdmin	DetectIntent (V2)

Prerequisites

- - 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.

- Access key for Google Text-to-Speech. For more information, see https://cloud.google.com/text-to-speech/docs/quickstart-client-libraries.

- Access key for Google Speech-to-Text. For more information, see https://cloud.google.com/speech-to-text/docs/quickstart-client-libraries.

- Configure Dialogflow Agent:

  1. Get the Dialogflow Agent Key. For more information, see https://dialogflow.com/docs/reference/v2-auth-setup.
  2. Migrate your Dialogflow Agent to Enterprise Essential (**Console Left Bar** > **Migrate from Standard to Enterprise Essential**).
  3. Enable the enhanced Speech Model in Dialogflow console (**Settings** > **Speech** > **Enable Enhanced Speech Model and Data Logging)**.

Note 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.

- 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 259. To know more about generating JSON key, see Generate JSON Key (for Non-OEM Users), on page 260.
- 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	NA	11.6/12.0 with A2Q approval	11.6/12.0 with A2Q approval

Note Trial version of Dialogflow ES GCP project should not be used in production environment.

VAV Onboarding for Non-OEM Users Task Flow

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 259.

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 260.

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**).

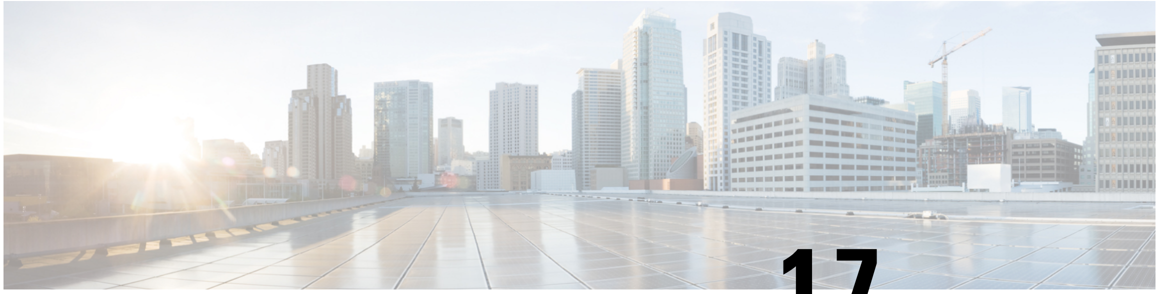
If this option is enabled, speech recognition data is shared with Google. For more information see https://cloud.google.com/speech-to-text/docs/enhanced-models.

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 PCCE Deployment	*Virtual Agent–Voice* section in *PCCE Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-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.

Information	Resource
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/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: • *Dialogflow Element* • *DialogflowIntent Element* • *DialogflowParam Element* • *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 17

Virtual Agent–Voice Call Transcription

VAV Transcript Overview

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

If a customer has interacted with a virtual agent at the beginning of the call and then the call gets routed to an agent, the **Transcript** gadget displays the transcript of the voice conversation between the customer and the virtual agent along with the live transcript. It helps in gathering context from the earlier interaction with the virtual agent and capturing high level summary points for wrapping up the call. In addition, the gadget displays the **Highlights** panel where you can view the following information:

- Intents and intent parameters (appear only if offered by the CCAI cloud provider). The intents and intent parameters are based on the customers' queries. For example, a customer's query is to book a flight ticket. The intent for this query is Flight Booking and the parameters for this intent are Source, Destination, and Date of departure depending on the customer’s interaction with the virtual agent.
- A confidence score of high, medium, or low for each intent.
- A customer sentiment indicator – happy, neutral, or sad, for each intent (appears only if offered by the CCAI cloud provider).

You can also view the overall sentiment indicator of the customer for the entire call.

The **Transcript** gadget shows the transcript of the voice conversation along with their timestamp in the local time zone of the agent desktop.

On the gadget interface, you can do the following:

- Filter the transcripts based on Customer, Virtual Agent, and Agent.
- Search the transcripts using keywords.

Note

- The Search box is disabled when the call is not active.
- If redaction or advance security settings are not enabled, PII and PCI information is also reflected in the gadget.

Prerequisites

To configure VAV Call Transcription:

Note Ensure that the Packaged CCE AW, VVB, Cloud Connect, and Agent Desktop components have access to Webex services to use VAV Call Transcription.

Procedure

	Command or Action	Purpose
Step 1	Complete the prerequisites for configuring VAV via Cloud-based connector. For more details, see Prerequisites, on page 264.	**Note** For this feature to work, you must deploy VAV via Cloud-based connector. The following components must be on release 12.6(1) or higher: • Cisco Unified CCE components (Router, Logger, AW, and PG) • Cisco Finesse • Cisco Unified CVP • Cloud Connect For further assistance, you can contact the Cisco TAC team. For more details, see https://www.cisco.com/c/en/us/support/web/tsd-cisco-worldwide-contacts.html.

Configuration Task Flow

Follow this procedure to enable the Cisco Contact Center AI (CCAI) Services that equip your Cisco Contact Center for VAV Call Transcription service.

Step 1 Configure VAV via Cloud-based connector. See VAV via Cloud-Based Connector, on page 244.

Step 2 Add the **Transcript** gadget and the VAV Transcription service to the Cisco Finesse desktop layout:

a. Enable the **Transcript** gadget in Cisco Finesse Administration.

   For details, see the *Manage Desktop Layout* section in the *Cisco Finesse Administration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html.

b. Enable the VAV Transcript 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 59.

Once enabled, the **Transcript** gadget appears on the **Home** tab. For details on how to use the gadget, see the Cisco Contact Center AI Gadgets guide at https://ccaigadgets.produs1.ciscoccservice.com/doc/en_us/index.html.

Note Gadget auto-hide/un-hide and notifications capability is available only if the gadget is configured as a multitab gadget in Cisco Finesse. For more details, see *Call Transcript Gadget* in the *Cisco Finesse Administration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html.

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.

Configure Contact Center AI Services for an Agent

Administrators can configure Contact Center AI Services for an agent while adding the agent. Supervisors can only enable or disable the services for an agent.

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

Step 2 Click **New** to open the **New Agent** page.
This page has: **General**, **Attributes**, **Skill Groups**, **Supervised Teams**, **Enable Email & Chat**, and **Contact Center AI** tabs.You cannot save the agent until you have entered all required fields on the **General** tab. You can complete other tabs as needed and in any order. For more information, see *Add and Maintain Agents* section in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Step 3 Click the **Contact Center AI** tab.
Displays a list of services 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 an Agent

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

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.

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

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 14.

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. 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. If no agent is found with the given username, the Contact Center AI Services association fails.

Field	Required?	Description
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, VAV Transcript, and Transcript. To associate more than one services, seperate the values using semicolon (;). 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 18

VPN-less Access to Finesse Desktop

VPN-less access to Finesse desktop

To enable this feature, use the Cisco provided Reverse Proxy Automated Installer that provides an in-built reverse-proxy, which is based on the OpenResty® Nginx proxy. Alternatively, any custom reverse-proxy pair must be deployed in the DMZ and configured to enable this feature.

VPN-less access to the desktop supports all standard functionality on the desktop including Real-Time and Historical Reports. The SSO authentication is supported along with mechanisms to tunnel ADFS access through the same proxy.

Media access remains unchanged in reverse-proxy deployments. To connect to the media, agents and supervisors can use the Cisco Jabber over Mobile and Remote Access solution (MRA). Otherwise, use the Mobile Agent capability of Contact Center Enterprise with a PSTN or mobile endpoint.

After deploying a reverse-proxy, if you want to access the Cisco Finesse desktop without connection to the VPN, read this chapter. Also, refer to the Nginx rules provided as part of the Reverse-Proxy installer artifact to replicate the same in your custom reverse-proxy. Refer to Reverse-Proxy selection and configurations section to determine the capabilities required for a custom reverse-proxy.

Note For Nginx-based reverse-proxy rules, installation, configuration, and security hardening instructions, refer to the Reverse Proxy Installer, on page 327 section. Any reverse-proxy that meets the requirements as mentioned in the Minimum and additional requirements section can be used in place of Nginx.

For the list of issues, see the Caveats section.

Attention The term "upstream servers" is used in this section to refer to all the solution components such as Finesse, CUIC, Cisco IdS, and IM&P servers that are configured to be accessed through reverse-proxy.

Prerequisites

To configure VPN-less access to the Finesse desktop:

- Finesse, IdS, and Cisco Unified Intelligence Center must be 12.6(1) ES02 or above.
- In coresident deployments, LiveData and Cisco Unified Intelligence Center should be 12.6 ES02 or above.
-
- Unified CCE and LiveData standalone must be 11.6(1) or above with the latest ES for the respective versions.
- DMZ with internet connectivity must be available to host the reverse-proxy.

Supported Reverse-Proxy Deployment Models

Reverse-Proxy deployment allows agents and supervisors to concurrently access the Finesse desktop from both LAN and via reverse-proxy. Cisco Contact Center supports the following two deployment models for VPN-less access to Finesse desktop using reverse-proxy:

- One Finesse cluster connects to one high availability (HA) pair of reverse-proxy.
- Multiple Finesse clusters connect to one HA pair of reverse-proxy.

Note This is applicable to Finesse, IdS, Cisco Unified Intelligence Center, and Live Data clusters.

Figure 21: Single Finesse Cluster per HA Reverse-Proxy

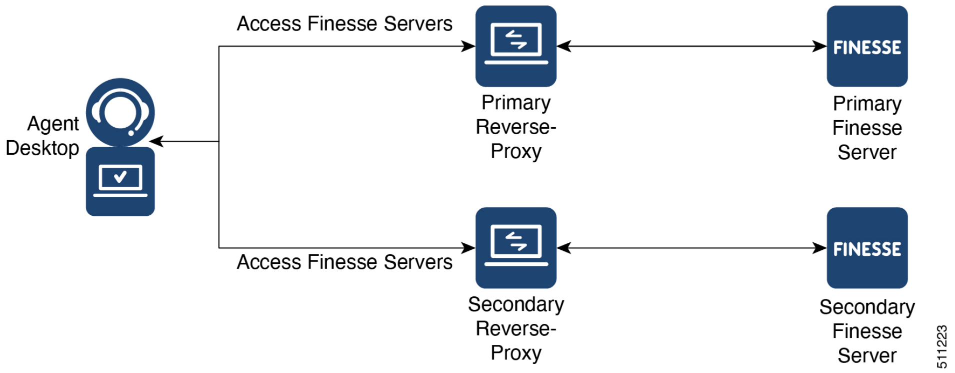
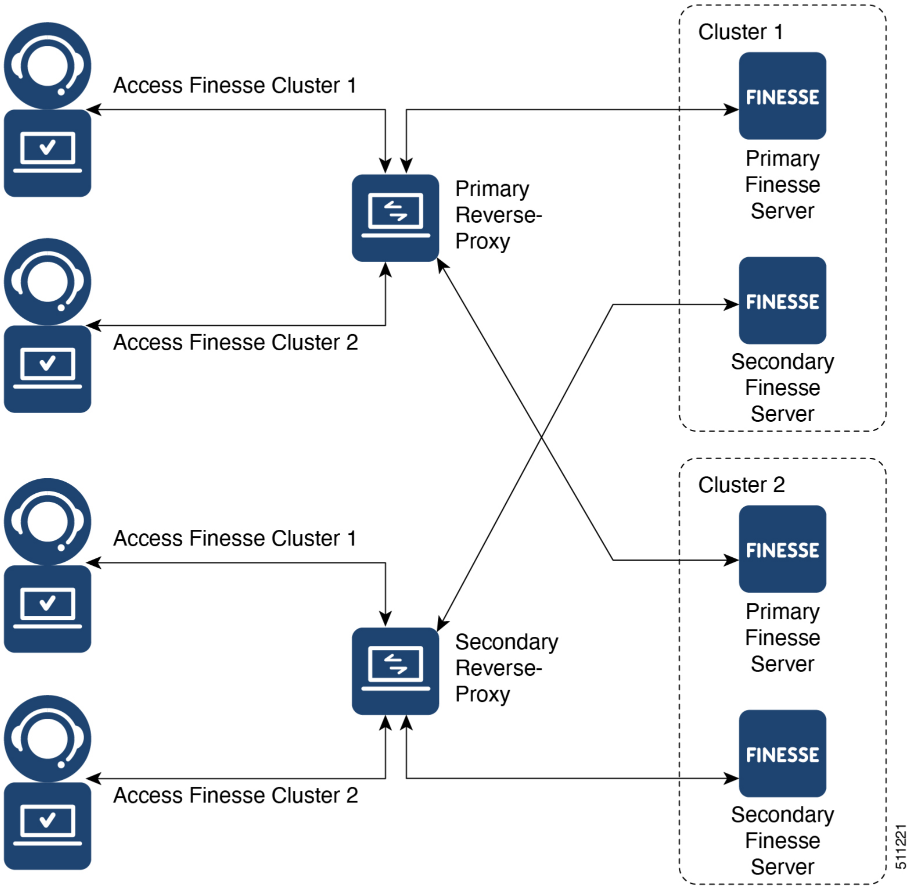


Figure 22: Multiple Finesse Clusters per HA Reverse-Proxy

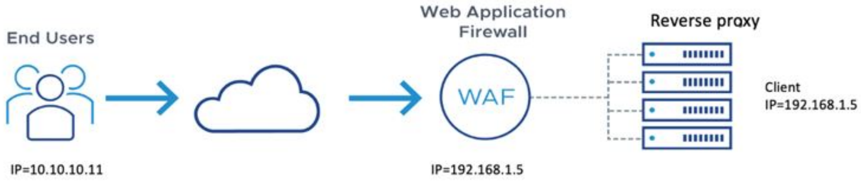


Load balancer, WAF, and proxy support for reverse-proxy deployments

The reverse-proxy configurations have security features that are dependent on the information about the actual client IP which is making the request. This information is required for enforcing security features such as enforcing rate limits, logging of client activity and blocking brute force attempts and for logging access to the system.

Deployments which directly terminate the internet Agent connections on the reverse-proxy don't need anything special to be done here, as the reverse-proxy recieves the client IP due to the client connections terminating on 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 itself, the client IPs are no longer visible to the reverse-proxy.



This happens when there are Load Balancers, Web Application Firewall (WAF), and so on, or when the client access itself is made from behind a forward proxy. CDN deployments also employ multiple reverse-proxies and fall into the same deployment category.

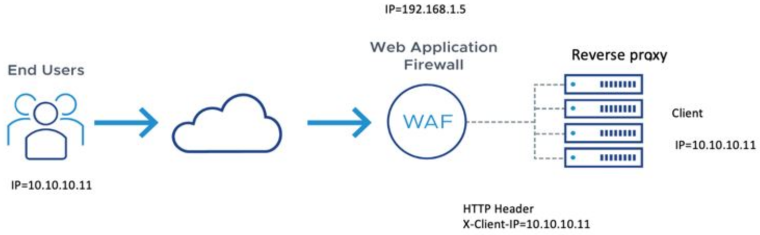
Such deployments **MUST** add certain configurations to enable the reverse-proxy to identify the actual client IP. The configurations that are required for such deployments are as follows:

1. The public IPs or the hostnames of these devices which forward the requests to the proxy in the reverse-proxy configurations, must be added in the **core.env** file using the variable *NGX_LOAD_BALANCER_IPS*.

2. The new requests originating from the intermediary devices, **MUST** populate the HTTP request header fields with the end-client IP to communicate the same to the reverse-proxy.

   The name of this field isn't predetermined and can be configured in the `core.env` file, in the variable *NGX_LOAD_BALANCER_REAL_IP_HEADER*.

Note All CDN deployments provide a mechanism to extract the client IP as an 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 doesn't 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 to the upstream 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 aren't blocked or rate limited. Ensure that the configurations are working as expected before going live.

   Refer to the Frequently Asked Questions, on page 351 section for instructions on how to send the requests to the proxy, and also on how to check whether a client is blocked or rate-limited.

6. Pretest the deployments with all WAF/IPS rules enabled to verify that the desktop API 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 are deployed.

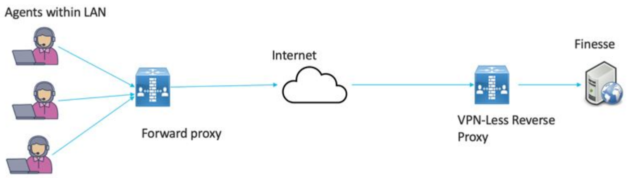
Note The reverse-proxy configurations provided have no protection against layer-3 attacks such as IP address spoofing or flooding. The proxy provides only rate-limiting, brute force attack detection, and restricting of requests to the allowed destinations. The operating system IP configurations are hardened to a certain level but there are no further protections that are available. It's assumed that the relevant operating system hardening and traffic protection devices are employed to secure the deployment of 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 that don't have the HTTP header support can skip the second and third points that are mentioned above. However, this causes a suboptimal deployment which will be functional but loses some critical security features listed previously and isn't a recommended configuration for certain features such as client IP logging for debugging purposes and blocking users attempting to brute force guess passwords.

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 IPs 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 have agents accessing the reverse-proxy deployments both from the internet and 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 IPs 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 IPs in a predefined header. For example, `X-Client-IP` as shown above.

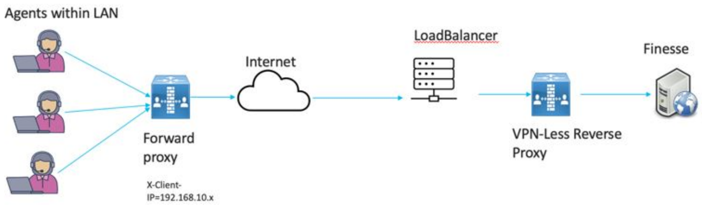
3. If there are other intermediary devices such as a Load Balancer or WAF at the network where Finesse is deployed, before the requests reach the reverse-proxy, these devices must be able to allow the Forward proxy by its IP address and then transmit the HTTP header without any changes.

Note It's crucial that the Forward proxy IP address is identified and only requests from this IP should be allowed to contain the predefined header from step 3. All other requests unless they are an identified proxy IP should strip this header `X-Client-IP`.

4. In the `core.env` file, the variable *NGX_LOAD_BALANCER_REAL_IP_HEADER* should identify the header used to send the client IP. For example, `X-Client-IP`.
5. The public IPs or the hostnames of the forward proxy needs to be added in the installer `core.env` using the variable *NGX_LOAD_BALANCER_IPS*, if the deployment doesn't have any other intermediary such as WAFS before the request reaches the reverse-proxy

Note Deployments that don't have the HTTP header support can skip the steps 2–4. However, this causes a suboptimal deployment which is functional but loses certain security features listed above. These are dependent on client IP knowledge and these deployments are therefore not suggested.

Reverse-proxy deployment can't support multiple HTTP header names to transmit the client IP corresponding to different Forward proxies that the network is interacting with.



Deployments that need to support forward proxies AND intermediary devices.

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. It must be configured as described in the section above. The Forward proxy information isn't 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 isn't set up 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 IP must be configured in the *NGX_LOAD_BALANCER_IPS* variable so that the proxy isn't blocked or rate-limited. This would be a suboptimal deployment and such deployments aren't suggested due to security constraints.

Authentication

For all requests and protocols that are accepted at the proxy, authentication is enforced before they are forwarded to the respective component servers (Finesse, LD, CUIC, and IM&P). The component servers also enforce authentication locally. Authentications made at the proxy use the Finesse login credentials, irrespective of the component server to which the requests are made.

Persistent connections such as WebSockets that rely on post connection application protocols (such as XMPP) for authentication, are authenticated at the proxy by verifying the peer IP address of the connection. The peer IP address must correspond to a system that has successfully authenticated an API request prior to establishing the socket connection.

Requests that do not require authentication, such as static files and images, are configured to be served by the reverse-proxy from its cache.

Non-SSO

Non-SSO authentication doesn't require any extra component configurations and works along with the Nginx authentication scripts provided with the corresponding ES release. Authentication relies on the Finesse sign-in credentials. Access to all the end points are validated using the Finesse authentication services.

The list of valid users is cached at the proxy locally (updated every 15 minutes), which is used to validate the user in a request. User credentials are validated by forwarding the request to the configured Finesse URI and thereafter the credential hash is cached locally (every 15 minutes) to authenticate new requests locally. Any change in the username or password will take effect only after 15 minutes.

SSO

SSO authentication requires the administrator to configure the Cisco IdS token encryption key at the Nginx server within the configuration file. Obtain the Cisco IdS token encryption key from the Cisco IdS server using the **show ids secret** CLI command. For the SSO authentication to work, configure the key as part of one of the mandatory replacements that the administrator must perform.

Cisco Identity Service 12.6(2)

After the reverse-proxy backend onboarding is complete using the **utils system reverse-proxy allowed-hosts add** CLI, the new reverse-proxy installer components will be able to successfully get the public key automatically from the Cisco IdS server and use that for authenticating the JWT tokens.

Note ADFS 3.0 can be configured to be accessed through the reverse-proxy with the Introduction, on page 325. Other IdP proxy configuration (proxy configuration, internet visibility, and High Availability for IdP) must be done separately by referring to the relevant IdP documentation. However, VPN-less configuration allows you to configure a different IdP hostname corresponding to the IdP-proxy to access the Finesse desktop.

Cisco IdS SAML configuration has to be performed for the SSO authentication to work at the proxy. For more information on Cisco IdS SAML configuration, see the Single Sign-On chapter.

After SSO authentication is configured, a pair of valid tokens can be used to access any of the endpoints in the system. The proxy configuration validates the credentials by intercepting the token retrieval requests made to Cisco IdS or by decrypting valid tokens and thereafter caching them locally for further validations.

Authenticate WebSocket connections

WebSocket connections don't have a standard authentication mechanism. Therefore, applications rely on post-connection application level protocol payloads for validating the established connection. However, this mechanism is used to establish unauthenticated connections at scale, mounting DoS or DDoS attacks on the servers.

To mitigate this possibility, the provided OpenResty® Nginx reverse-proxy configuration performs specific checks before allowing WebSocket connections. The WebSocket connections are accepted only from those IP addresses that have successfully made an authenticated REST request. The REST request must be authenticated before establishing the WebSocket connection.

Reverse-proxy deployments that use L7 intermediaries, such as Content Delivery Network (CDN), often redirect traffic through interim servers before the traffic reaches the reverse-proxy. In such deployments, ensure that the **X-Forwarded-For** headers are correctly relayed to identify the client IP address. The **X-Forwarded-For** headers are used to authenticate the WebSocket connection by matching it with the previously authenticated REST request.

Note The clients that attempt to create WebSocket connections before issuing any REST requests, an **Authorization Failed** error message is displayed.

Host-Mapping file for network translation

Reverse-proxy deployment requires an invalid mapping file to configure the list of externally visible hostname/port combinations. It's also used to map to the actual server names and ports that are used by the Finesse, IdS, and CUIC servers. This mapping file which is configured on internal servers is the key configuration that allows the clients connected over the internet to be redirected to the required hosts and ports that are used on the internet.

The mapping file has to be deployed on a web server accessible to the component servers. Its URI must be configured using a dedicated web server available within the LAN. If such a server isn't available, the reverse-proxy can be used instead. This requires that the proxy is accessible from within the LAN. Using the reverse-proxy presents a risk of exposing the information to external systems which can make an unauthorized connection to the DMZ.

Note For details on how to configure a Proxy Map to be served by the reverse-proxy itself, see the Reverse Proxy Automated Installer, on page 325 section.

For all the requests that come through the reverse-proxy, the Finesse, IdS, and CUIC servers check the host-mapping file, to translate the internal hostnames and ports that are used on the LAN. They are translated to the publicly resolvable hostnames and ports that have to be used on the internet. This mapping file referred to as the Proxy-config map file, is the key configuration that allows the clients connected over the reverse-proxy to be redirected to the required hosts and ports that are used on the internet.

Configure the proxy-config map file by using the CLI available on Finesse, Cisco IdS, and CUIC servers. For details on the mapping file format and the data configured, refer to the *Populate Network Translation Data* section. For details on the CLI used to configure the file, refer to the *Configure Proxy Mapping by Using CLI* section.

Configure the proxy-config map file by using the CLI available on Finesse, Cisco IdS, and CUIC servers. For details on the mapping file format and the data configured, refer to the *Populate Network Translation Data* section. For details on the CLI used to configure the file, refer to the **utils system reverse-proxy config-uri** CLI in the Configure Proxy Mapping by Using CLI section.

Port management

One of the main design aspects in deploying a reverse-proxy are the domain and the ports used to access the application. These aspects are interdependent and influence each other when designing the deployment.

The reverse-proxy must be able to determine to which upstream server an incoming request can be forwarded to where an incoming request should be forwarded to. This can be accomplished by changing either the port or the hostname used to access the application. Primarily, the combination of host and port must be unique for the proxy to differentiate and route traffic to the correct upstream component. It's a requirement for the proxy to even start correctly. The following are the options available to design the domain and port access:

- Use a common domain and differentiate application access using multiple ports.
- Use a common port and differentiate application access using multiple domains

After determining the domain and the port distribution, perform the following steps:

1. Change the proxy map configuration to match the required port and domain. See the Configure Proxy Mapping by Using CLI, on page 288 section.
2. The respective upstream component environment configuration in the reverse-proxy installer must be configured with the required hostname and port. See the Configure deployment environment configurations, on page 337 section.

Using a common domain with multiple ports

The following example illustrates how multiple application servers can be configured using this access pattern:

- FinesseA = ReverseProxyDomain.com:8445
- FinesseB = ReverseProxyDomain.com:8446
- Finesse1A = ReverseProxyDomain.com:8447
- Finesse2B = ReverseProxyDomain.com:8448

The following are the benefits of using multiple ports:

- More granular packet level rate-limits applicable to each application can be applied at the ingress point to control rate-limits. Domain-level access means that the rate-limits can’t be granular.
- A single-domain requires only a single SSL certificate to access the application. It could be a factor in reducing costs, unlike a multiple-domain application which requires a wildcard certificate.

The following are the disadvantages in using multiple ports:

- Certain network deployments like CDNs don't support custom ports.
- Security devices that automatically apply security rules might require custom configurations with non-standard ports.
- Multiple ports must be opened in the DMZ firewall (10–15 ports are required for a standard 2k deployment). This isn’t recommended by the network security teams.
- There’s an increased overhead regarding the port manageability.
- Deploying new instances of the application requires firewall/network changes.

Note Ports other than the ones mentioned in the ProxyMap must be blocked and shouldn’t be available for access on the reverse-proxy host. This must be blocked at the ingress point as the proxy doesn’t currently have rules to block this access at network level.

The Cisco provided installer supports running multiple instances which cater to different sets of upstream servers, to aid in ease of maintenance. Multiple instances of the installer don't allow to use the same ports across different instances of the proxy. Only one process can bind to the same TCP port.

Consider the above two points when deciding the port management strategy against proxy installer configuration.

Using a common port and with multiple domains

The following example illustrates how multiple application servers can be configured using this access pattern.:

- FinesseA = FinesseA-ReverseProxyDomain.com:443
- FinesseB = FinesseB-ReverseProxyDomain.com:443
- Finesse1A = Finesse1A-ReverseProxyDomain.com:443
- Finesse2B = Finesse2B-ReverseProxyDomain.com:443

The single port configuration reverses the pros and cons listed above with the multiple port configuration.

Note Supporting a single port of access requires Unified Intelligence Center and LiveData components to be on 12.6(2)versions.

DNS Configuration for Finesse, IdS, and CUIC servers

Each Finesse, IdS, CUIC, IM&P, and third-party component servers corresponding to a host that needs internet access must be addressable from the internet. This calls for a hostname and associated port which is resolvable from the internet to be mapped to the public port and matching IP of the reverse-proxy. It’s required so that the traffic is directed to the respective component servers.

DNS registration of the publicly resolvable hostnames and the corresponding IP addresses is mandatory before the requests reach the reverse-proxy.

SSL Certificates

For the hostnames that are configured, corresponding to each unique hostname that is used by the internet client, the respective certificates must be acquired and configured on the reverse-proxy. Even though self-signed certificates are supported, they are risky because the users access directly from the internet. The clients can be more secure by using CA-signed certificates. The best practice is to get CA certificates for proxy servers and third-party-gadget servers.

Reverse-Proxy selection and configurations

This section provides information about the recommendations for hardware, performance, scale, and gadget compatibility for reverse-proxy.

If you choose to not use the reverse-proxy installer that Cisco has provided and want to deploy a custom reverse proxy, see the Guidelines for Custom Reverse Proxy Deployment, on page 397 section.

Performance and Hardware Recommendations

Packaged CCE deployments can be supported by Open Source Nginx 1.20 running on a CentOS 8.0 (4.18.0-305 64 bit) distribution, with the configurations and settings (mentioned in the Installing Nginx site) on a dual core 4 CPU (8 logical CPU) Intel Xeon CPU E5-2690 v2 (3.00GHz, 25MB cache) at an average of 10% CPU usage and peak of 15% CPU usage during logins.

Note The load has been tested on 2K deployments.

It is expected that the same configuration can support three Finesse clusters with the required CUIC LD reports, and access to IdS.

A minimum of 8 GB memory is recommended for the proxy server when all other nonessential services and graphical subsystems are disabled.

Note It is recommended that deployments gradually onboard new solution components to the proxy until 50-55% of the proxy CPU is free. With this it can cope with unexpected spikes in traffic from the internet.

Additional memory must be configured based on the in-memory cache configuration added to Nginx.

The data analysis of top equivalent performance capture, made with the nmon tool, is available from the Installer download page https://software.cisco.com/download/home/283613135/type/284259728/release/12.6(2). The data represents the state of the proxy for desktop and supervisor operations, on a sample 2000 and 4000 UCCE deployment using SSO logins and CUIC LD reports as configured in the default layout for all agents for a period of eight hours. It can be used to derive the compute, disk, and network requirements for an installation using Nginx on comparable hardware.

Determine Scale and Hardware for Proxy

Contact Center administrators should analyze the hardware required for the reverse-proxy, based on the number of agents and supervisors who may access the Finesse desktop without connecting to VPN. You can use the reference request rates provided for Finesse, IdS, and CUIC.

The type of proxy selected guides the hardware to be used, depending on whether the proxy is shipped as an installable software or is a hardware-based application.

Sizing configurations are pre-tested for OpenResty® Nginx proxy. Custom proxy deployments should consult their product documentation or run basic scaling tests to determine the rates that can be supported by the respective proxy and scale their hardware accordingly.

Determine Gadget Compatibility

Determining the gadget compatibility is an important activity for planning a VPN-less Finesse deployment.

After deploying the reverse-proxy, all Cisco-provided gadgets (Cisco Finesse and Cisco Unified Intelligence Center) work seamlessly with their respective servers of Release 12.6(1) or later. The Webex Experience Management and CCAI gadgets also work seamlessly with VPN-less Finesse deployments.

In some scenarios, depending on the gadget design, custom third-party gadgets require workarounds to enable them to work with the reverse-proxy deployment. Refer to the following sections to determine if any of your gadgets require workarounds.

Note

- Gadgets that are loaded from servers other than Finesse server should use **exclude-url** feature in the gadget XML specification to load the Finesse resources such as Finesse.js. For more information, refer to the **Use Gadget URI Exclude Feature to Refer to Finesse Resources** section.
- If you use two different URLs, one internal and one external, in Enterprise Chat and Email (ECE), you must update the Finesse desktop layout to use only the external URL. If you use an internal-only ECE (for integrations that support only ECE email routing), you must change the ECE web server to ensure that the ECE services are accessible externally.

Gadget Types and VPN-less Compatibility

Finesse gadgets are classified into the following types based on how they are designed operationally:

- Gadgets that are self-contained within the desktop. These gadgets do not have to make any additional network requests, or are restricted to invoking Finesse APIs and APIs on the internet.
- Gadgets that provide their functionality by communicating with an accompanying server that is deployed in the DMZ and is reachable directly from the internet and LAN.

  Note To enable the same desktop layout to be used by both LAN-based and internet-based clients, the server installed in a DMZ should also be reachable from servers such as Finesse in LAN, and from clients that are running within the LAN.

- Gadgets that need to communicate with an accompanying server deployed in LAN, but uses desktop-provided **makeRequest** API to communicate to the server. The **makeRequest** API routes all the requests through the Finesse server and does not directly reach the server that is deployed in the LAN.

  Note These requests succeed in a reverse-proxy deployment only if the requests are made using the hostname and port. The hostname and the port must be reachable from LAN because the requests are run by Finesse server which runs on LAN.

- Gadgets that have to communicate directly with any one of the following types of accompanying server:
  - Server deployed within the LAN and is not reachable directly from the internet.
  - Server that communicates with an additional port apart from the HTTP port used to load the gadget.

The last two types of gadgets have to be modified to be used in a reverse-proxy deployment. The steps required to enable these gadgets to be accessed from internet clients are as follows:

- Enable VPN-less access for custom gadgets
- Send hostname and port information to gadgets
- Use gadget's **URI Exclude** feature to refer to Finesse resources

Enable VPN-less Access for Custom Gadgets

Gadgets that communicate directly with accompanying servers that are deployed in LAN must handle the following aspects to work correctly in a reverse-proxy deployment:

- Use the right hostname and port for communicating with its accompanying server.

  A gadget can find the correct hostname and port corresponding to the server from which the gadget was loaded, by using the **gadgets.util.getUrlParameters().up_urlPrefs** API provided by the Finesse Javascript API.

  To find additional ports or hostnames that are required, data can be passed in as gadget preference such that the additional host and port information can be sent to the gadget. For more information, refer to the **Send Hostname and Port Information to Gadgets** section.

- Ensure that the communications are forwarded correctly by the reverse-proxy.

  After the gadget starts communicating with the correct host and port information, the hostname and port number have to be forwarded to the server deployed in the LAN. This can be done by opening the appropriate ports in the DMZ firewall. Also, ensure that the appropriate ports and rules are added to the reverse-proxy rules to forward the traffic to the correct server in the LAN.

- **Best Practice:** If requests to external servers are made using Finesse authentication headers, a common validation is enabled to authenticate the requests at the proxy. Gadgets that do not use Finesse authentication should plan to implement their own custom authentication schemes to ensure that the requests are validated at the proxy before sending to the Finesse server.

Send Hostname and Port Information to Gadgets

Gadgets that send host and port information corresponding to a server deployed within the LAN can use the **UserPreferences** feature supported by Finessse gadgets. This feature allows a configurable, named information to be passed to the gadget. The information can be referenced within the gadget XML or programmatically by using a Javascript.

For more information on how to use **UserPreferences** method, refer to https://developer.cisco.com/docs/finesse/#!gadget-preferences.

The **UserPreferences** that are created for this purpose should start with the keyword *externalServerHostAndPort* in its name. This enables Finesse to substitute the host and port that are provided with the corresponding entry from the **proxyMap** file. For example:

```
<UserPref name="externalServerHostAndPort_chat” display_name="Chat_externalServerHostAndPort"
default_value="SMHostName:7443" datatype ="hidden"/>
```

Note The `default_value` parameter is not case sensitive.

When accessed from the LAN, the **UserPreferences** continues to have the default value that is configured in the XML. However, when accessed through the reverse-proxy, the **UserPreferences** receives the value from the **proxyMap** file. For example:

```
SMHostName:7443=external-proxy-host:4043
```

When accessed through the reverse-proxy, the gadget receives the port **4043** and host name as **external-proxy-host**.

Use Gadget URI Exclude Feature to Refer to Finesse Resources

Add the following content within the `ModulePrefs` tag of the gadget XML to ensure that the resources that are loaded from Finesse server are excluded from concatenation. This step is mandatory for gadgets that load their XML from custom servers.

```
<Optional feature="content-rewrite">
<!-- these files will be directly served by Finesse, not through shindig -->
<Param name="exclude-url">finesse.min.js</Param></Optional>
```

Finesse URL

Agents and supervisors should bookmark two different pairs of URLs (publisher and subscriber) for accessing the Finesse desktop through both the Contact Center network and the reverse-proxy.

VPN-less Finesse configurations

To configure VPN-less access to Finesse desktop, the Contact Center administrators and the network administrators must work in tandem.

Note Don't allow access to the reverse-proxy in your external firewall until all security configurations are in place. To test your changes, use a host that isn’t publicly accessible.

The configuration steps are as follows:

1. Populate Network Translation Data
2. Host the Mapping File
3. Add Proxy IP by Using CLI
4. Configure Reverse-Proxy Host Verification
5. Configure Proxy Mapping by Using CLI
6. Configure CORS and Frame-Ancestors
7. Configure SSO

Populate Network Translation Data

The Proxy-config map file is similar to a plain property file in which the values are separated by the equal sign. Left Hand Side (LHS) contains the host and port of Finesse, IdS, and Cisco Unified Intelligence Center. Right Hand Side (RHS) contains the values of the host and port that are exposed through reverse-proxy to access the Finesse desktop.

Network administrator and Finesse administrator must create a Proxy-config map file that has the mapping for all the default ports of the Cisco components. The external traffic from the Internet will be redirected to the default ports. For example, 8445 port of Finesse, 8553 port of IdS, and 8444 port of Cisco Unified Intelligence Center.

The Proxy-config map file must be hosted on a web server that is accessible by the Finesse, IdS, and Cisco Unified Intelligence Center servers. The following list is an example of systems and hosts that are required for a two-node Finesse cluster with two Cisco Unified Intelligence Center nodes using SSO mode:

- Publisher = finesse1.internal.com
- Subscriber = finesse2.internal.com
- IdS Publisher = idspub.internal.com
- IdS Subscriber = idssub.internal.com
- IdP = idp.internal.com (optional)
- CUIC Publisher = cuicpub.internal.com
- CUIC Subscriber = cuicsub.internal.com
- Proxy Node1 = proxy1.xyz.com
- Proxy Node2 = proxy2.xyz.com

If the selected proxy supports port-based forwarding, the following is an example of a mapping file that contains the entries that are required for a two-node Finesse cluster with two Cisco Unified Intelligence Center nodes using non-SSO mode.

```
finesse1.internal.com:8445=finesse1.xyz.com:443
finesse2.internal.com:8445=finesse2.xyz.com:443
idspub.internal.com:8553=idspub.xyz.com:443
idssub.internal.com:8553=idssub.xyz.com:443
idp.internal.com:443=idp.xyz.com:443
cuicpub.internal.com:8444=cuicpub.xyz:8444
cuicsub.internal.com:8444=cuicsub.xyz:8444
```

Note

- The IdP entry `idp.internal.com:443=idp.xyz.com:443` is optional. You must add this entry when the IdP hostname configured with IdS is different for reverse-proxy and LAN.
- The LHS entries of the hostname in the proxy-map file are not case sensitive. The RHS entries are case sensitive and must match exactly with the hostnames that are configured.
- If the proxy map file entries do not contain colon (:), it is assumed that only the hostname is entered. If you have not provided any port value, the port number 443 is considered as the default port.

Figure 23: Hostname Mapping Example

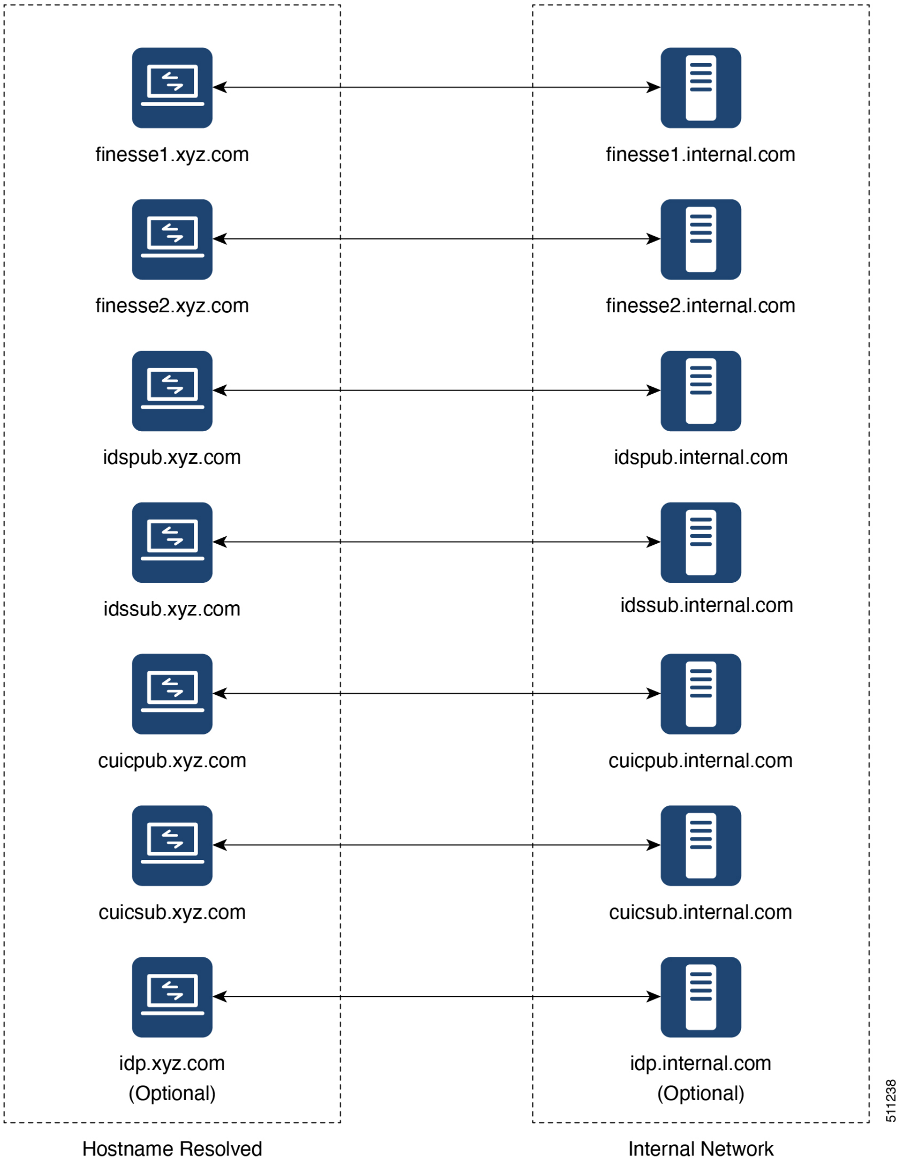
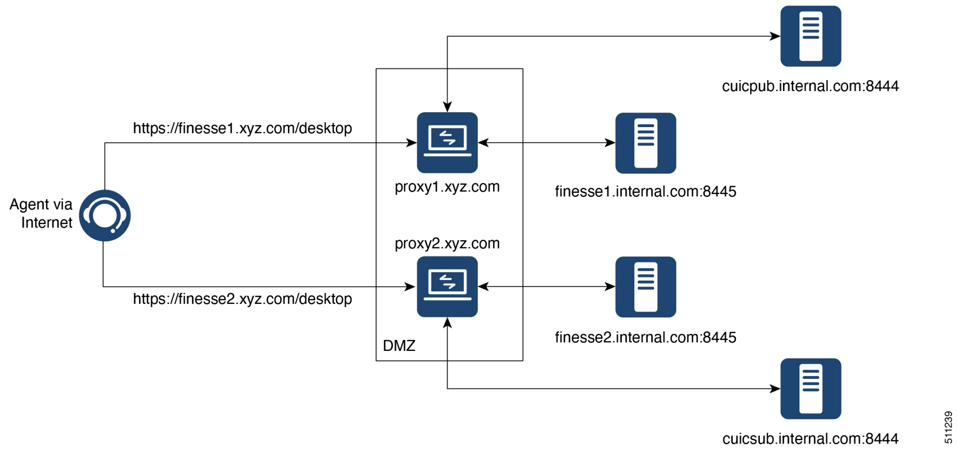


Figure 24: Network Architecture Example


cuicpub.internal.com:8444
https://finesse1.xyz.com/desktop
proxy1.xyz.com
finesse1.internal.com:8445
Agent via Internet
proxy2.xyz.com
https://finesse2.xyz.com/desktop
finesse2.internal.com:8445
DMZ
cuicsub.internal.com:8444
511239

Host the Mapping File

The mapping file that is created in the *Populate Network Translation Data* section, is used by the solution components (Finesse, IdS, and CUIC servers) servers to modify their responses, to enable clients to access the solution via the reverse-proxy. This requires the file to be hosted on any web server accessible by the component servers. The reverse-proxy server, Finesse server, or any web server configured by the administrator can be used for this purpose.

To access the mapping file, the host server's SSL certificate must be uploaded (using the cmplatform admin application) to the individual nodes of the services. After uploading the file, verify if the URL is accessible from Finesse, IdS, and CUIC servers. For example, *https://proxyserver.xyz.com:10000/proxymap.txt*. HTTP-based URLs are allowed for hosting the mapping file through HTTPS, which is the recommended access scheme.

The following is an example of the CLI to view the content of the proxy map file.

```
admin:utils system reverse-proxy show-proxy-config-map

finesseXX.autobot.cvp:8445=astproxy.cisco.com:8445
finesseXYZ.autobot.cvp:8445=astproxy125.cisco.com:8445
cuic-YY.autobot.cvp:8444=astproxy.cisco.com:8444
cuic-YY.autobot.cvp:8447=astproxy.cisco.com:8447
livedata-LL.autobot.cvp:12005=astproxy.cisco.com:12005
livedata-LL.autobot.cvp:12008=astproxy.cisco.com:12008
cuic-PQR.autobot.cvp:8444=astproxy125.cisco.com:8444
cuic-PQR.autobot.cvp:8447=astproxy125.cisco.com:8447
livedata-ABC.autobot.cvp:12005=astproxy125.cisco.com:12005
livedata-ABC.autobot.cvp:12008=astproxy125.cisco.com:12008
ids.autobot.cvp:8553=astproxy.cisco.com:8553
ids2.autobot.cvp:8553=astproxy125.cisco.com:8553
finadfs-WXY.finesse.com:443=astproxy-idp.cisco.com:443
fincup1-WX.cisco.com:5280=astproxy.cisco.com:5280
fincup2-PQ.cisco.com:5280=astproxy125.cisco.com:15280
fincup3-RST.cisco.com:5280=astproxy.cisco.com:25280
```

Add Proxy IP by Using CLI

The administrator must use CLI to add the list of trusted reverse-proxy IP addresses and their corresponding hostnames. This must be done on all the nodes of Finesse, IdS, CUIC, and LiveData (12.6(1) ES01 and above). These components consider only requests from the configured hosts or IP addresses as valid.

Note

- Ensure to add the hostnames that are resolvable from the respective components from where the CLI is run.
- Ensure to add both public and private IP addresses of the reverse-proxy.
- The allowed hosts must not contain the hostname or IP address of the load balancer. It should contain only the internal and external hostname and IP address of the reverse-proxy.

The following is an example of the CLI to add the hosts and IP addresses:

```
admin:utils system reverse-proxy allowed-hosts add 10.78.95.178
Source 10.78.95.178 successfully added
admin:utils system reverse-proxy allowed-hosts add proxy.xyz.com
Source proxy.xyz.com successfully added

Restart Cisco Web Proxy Service for the changes to take effect: utils service restart Cisco
 Web Proxy Service
```

If the added hostname is not resolvable from a component, the following error is displayed:

```
admin:utils system reverse-proxy allowed-hosts add group.facebook

Either IPv4 address or hostname is invalid or is not resolvable. Now validating IPv6 address
 for source group.facebook

Operation failed, please enter valid source(s). Source group.facebook is invalid
```

After adding proxy hosts as trusted hosts through CLI on individual nodes, you must upload proxy server certificates to the Tomcat trust store of the respective components. This is required for proxy authentication to work. Otherwise, the traffic from proxy will be rejected by the components. For information about generating proxy certificates and uploading to the Tomcat trust store, see the *Set up Nginx reverse proxy certificate* and *Generate and Copy CA Certificates of VOS Components* sections in the Security Guide for Cisco Unified ICM/Contact Center Enterprise.

Note If you are upgrading from 12.6(1) ES01, you must copy and upload proxy server certificates to the Tomcat trust store of the respective components. The certificates are required at the component server to verify and allow TLS connections from the proxy.

The following is an example of the CLI to view the list of allowed hosts and IP addresses:

```
admin:utils system reverse-proxy allowed-hosts list

Source proxy.xyz.com successfully added list

The following source(s) are configured:

1. 10.78.95.178
```

```
2. proxy.xyz.com
3. proxy125.xyz.com
```

The following is an example of the CLI to delete an entry from the list of allowed hosts and IP addresses. This command lists all the configured proxy hosts and IP addresses, and gets user input to delete specific or all proxy hosts and IP addresses.

```
admin:utils system reverse-proxy allowed-hosts delete
Select the reverse-proxy source IP to delete:

 1) 10.78.95.178
 2) proxy.xyz.com
 3) proxy125.xyz.com
 4) all
 5) quit

Please select an option (1 - 5 or "q" ): 1

Delete operation successful
```

Configure Reverse-Proxy Host Verification

You can configure SSL certificate verification for communication between reverse-proxy host and the Cisco Web Proxy Service by running the following CLI command on both publisher and subscriber nodes of Finesse:

utils system reverse-proxy client-auth

This command has the following parameters:

- enable
- disable
- status

By default, the host authentication is enabled.

The following is an example of the CLI to view the status of the host authentication:

```
admin:utils system reverse-proxy client-auth status

SSL certificate verification for connections established from reverse proxy hosts is disabled
```

The following is an example of the CLI to enable the host authentication:

```
admin:utils system reverse-proxy client-auth enable
SSL certificate verification enabled for connections established from reverse proxy hosts

Restart Cisco Web Proxy Service for the changes to take effect:
utils service restart Cisco Web Proxy Service
```

Note After enabling the reverse-proxy host authentication, browser-based clients that connect to Finesse Desktop via LAN hostname must select a client certificate. A pop-up is displayed on systems where client certificates are installed. Clients can choose any of the certificates listed in the pop-up, and continue to connect to Finesse.

The following is an example of the CLI to disable the host authentication:

```
admin:utils system reverse-proxy client-auth disable
SSL certificate verification disabled for connections established from reverse proxy hosts
```

```
Restart Cisco Web Proxy Service for the changes to take effect:
utils service restart Cisco Web Proxy Service
```

Configure Proxy Mapping by Using CLI

The Proxy-config map file can be configured in the Finesse, IdS, and CUIC servers using the `utils system reverse-proxy config-uri` command. If the URL is configured to use HTTPS protocol, Finesse, IdS, and CUIC must have the certificate (certificate of the web server hosting the URL) uploaded in */cmplatform*. The administrator can configure a maximum of two URLs. The URL that is added first takes precedence and that URL is polled to detect changes in the mapping file. When the URL is not accessible, the alternate URL is used. The following is an example of the CLI to list the configured Proxy-config map URLs:

```
admin:utils system reverse-proxy config-uri list

Currently no source is configured
```

The following is an example of the CLI to configure the Proxy-config map URL on the Finesse, IdS, and CUIC servers:

```
admin:utils system reverse-proxy config-uri add https://saproxy.xyz.com:10000/proxyconfig.txt

Operation failed, please enter valid source(s). Source
https://saproxy.xyz.com:10000/proxyconfig.txt is invalid

admin:utils system reverse-proxy config-uri add https://saproxy.xyz.com:10000/proxymap.txt

Source https://saproxy.xyz.com:10000/proxymap.txt successfully added

admin:utils system reverse-proxy config-uri list

The following source(s) are configured:

1. https://saproxy.cisco.com:10000/proxymap.txt
```

The following is an example of the CLI to delete existing Proxy-config map URLs. This command lists all the configured Proxy-config URLs and gets user input to delete specific or all Proxy-config URLs:

```
admin:utils system reverse-proxy config-uri delete
Select the reverse-proxy source URI to delete:

 1) https://saproxy.xyz.com:10000/proxymap.txt
 2) all
 q) quit

Please select an option (1 - 2 or "q" ): 1

Delete operation successful
```

The following is an example of the CLI to set the Proxy-config update frequency (in minutes). Based on the set frequency, the local file system of Finesse, IdS, and CUIC are updated with the content from the Proxy-config map file. Before configuring the URL, this command does not return any value. After configuring the Proxy-config map URL, by default it returns one minute as the value.

```
admin:utils system reverse-proxy show-config-update-frequency
No config-uri configured

admin:utils system reverse-proxy config-uri add https://saproxy.xyz.com:10000/proxymap.txt
```

```
Source https://saproxy.xyz.com:10000/proxymap.txt successfully added

admin:utils system reverse-proxy show-config-update-frequency
1 minute

admin:utils system reverse-proxy set-config-update-frequency 5

admin:utils system reverse-proxy show-config-update-frequency
5 minutes
```

Configure CORS and Frame-Ancestors

Add both the primary and secondary reverse-proxy origins on publisher and subscriber nodes of Finesse and CUIC. If you change Cross-Origin Resource Sharing (CORS) allowed list and frame-ancestors, you must restart Finesse Notification and Tomcat services. For information about restarting Finesse notification service, see the *Cisco Finesse Services* section in *Cisco Finesse Administration Guide*.

- Administrators must add the list of proxy server origins on the allowed list of CORS origins, if the CORS setting is enabled on Finesse, CUIC, and Live Data .
- Frame-ancestors are added automatically while adding the reverse-proxy trusted hosts in Finesse servers.
- Administrators must add frame-ancestors while adding reverse-proxy trusted hosts in CUIC servers.
- Administrators must delete the corresponding allowed list of CORS and frame-ancestors entries while deleting the trusted hosts of a reverse-proxy.

Caution If you do not delete the corresponding CORS and frame-ancestors entries, it becomes a security vulnerability.

Note CORS and frame-ancestors are not applicable to IdS.

For information about deleting CORS see the *Cross-Origin Resource Sharing (CORS)* section in the *Cisco Finesse Administration Guide*.

For more information about configuring CORS, see the Live Data CORS Configuration section in Cisco Unified Contact Center Enterprise Installation and Upgrade Guide.

For information about deleting frame-ancestors see the *Supported Content Security Policy Directives* section in the *Cisco Finesse Administration Guide*.

Configure SSO

If SSO is enabled Packaged CCE, SSO must be configured for VPN-less access. Otherwise, agents and supervisors can't login to the Cisco Finesse desktop.

The steps to configure SSO are as follows:

1. Administrator must download proxy specific SAML SP metadata from IdS administration interface.
2. Add proxy relying party trust with IdP.
3. Add proxy redirect URIs to Finesse clients manually via IdS admin interface.

4. Validate SSO configuration for reverse-proxy from IdS admin

For more information, see the Single Sign-On.

Note

- Proxy configuration does not reflect in IdS in any one of the following scenarios:
    - IdP metadata is not uploaded
    - IdS is in maintenance mode
    - Maintenance mode is completed.
- If proxy configuration is changed for IdS hosts, administrator must reestablish trust on IdP for new IdS proxy hosts after downloading new metadata file from IdS admin. Administrator must reestablish **Relying Party Trusts** with IdP. For more information, refer to the Integrate Cisco IdS with AD FS
- If proxy configuration is changed for Cisco Finesse hosts, administrator must manually update the allowed Finesse client redirect URIs list on IdS admin interface. For more information, refer to theConfigure the Cisco Identity Service. Client name is "Finesse" and the URLs that are to be added are as follows:
    - `https://<finesseReverseProxySideAHost:finesseReverseProxySideAPort>/desktop/sso/authcode`
    - `https://<finesseReverseProxySideBHost:finesseReverseProxySideBPort>/desktop/sso/authcode`
-

Serviceability

Monitor Connected Agents and Supervisors

The reverse-proxy has to be monitored by using the proxy-specific features. For more information, refer to the specific reverse-proxy documentation.

Cisco Finesse allows administrators to view the list of currently connected agents and supervisorsCCEAdmin. The administrator can filter and see the agents and supervisors who are connected to the Finesse desktop based on the connection type. For example, agents and supervisors connected through the Contact Center network and those connected through reverse-proxy can be seen. For more information, see the *Connected Agents* section in *Cisco Finesse Administration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html. Administrators can also view the summary of connected users by using the following CLI command:

```
admin:utils finesse show_connected_users summary

Total Connected Users: 6

Desktop Users: 1
FIPPA Users: 2
Third-party Users: 3

Users connected to Finesse via LAN/WAN: 5
Users connected to Finesse via Proxy: 1

To view the complete list of signed-in users, log in to the Cisco Finesse
Administration Console, and navigate to the Connected Agents tab.
```

To view the real-time list of connected users by using an API, see the *ConnectedUsersInfo* section in *Cisco Finesse Web Services Developer Guide* at https://developer.cisco.com/docs/finesse/#!rest-api-dev-guide.

API Modifications to Support Reverse-Proxy Deployments

Finesse SystemInfo API

SystemInfo API is now secured when it is accessed through a reverse-proxy. The API is accessible with agent and supervisor credentials. The following field has been added to support this feature:

- **httpsPort:** HTTPS port has to be used for all Finesse API and desktop notifications.

For more information, see the *SystemInfo* and *ConnectedUsersInfo* sections in *Cisco Finesse Web Services Developer Guide* at https://developer.cisco.com/docs/finesse/#!rest-api-dev-guide.

HTTP Return codes returned by the reverse-proxy

These are the HTTP return codes returned by the proxy in exceptional scenarios. The HTTP error code and the situations in which these custom error codes are sent are as follows:

Error Code	Error	Reason
444	No Response	Accessing https ports with http requests.
401	Unauthorized	Authentication Failed - Note, even SystemInfo requires to be authenticated when accessed through reverse-proxy.
403	Forbidden	When restricted pages are accessed or the user is unauthorized. `cfadmin/FIPPA/`, `test sso`, and some of the pages are restricted through reverse-proxy.
405	Method not allowed	Accessing URLs with invalid HTTP methods. For example, most of the Cisco IdS URLs support only GET, POST, and OPTIONS methods. Anything other than these methods results in this error.
412	Precondition Failed	Check the **User-Agent** field of the request header. Reverse-proxy validates some of the pre-configured **User-Agent** headers that could be triggered from Bots.

Error Code	Error	Reason
417	Expectation Failed	The server can’t meet the requirements of the **Expect request-header** field. Please check the referrer header field is populated as one of the valid configured referrer headers. Please check the **NGX_VALID_REFERRERS** in the **core.env** file has all the valid referrer headers configured for the failed request.
421	Misdirected Request	Accessing URL that isn’t supported through reverse-proxy. Reverse-proxy configured to support only a set of upstream URLs. Any URL other than that is requested results in this error.
429	Too Many Requests	The response status code indicates that the user has sent too many requests in a given amount of time ("rate limiting"). Need to check the configured LB and rate limiting configurations on the **core.env** file and component's envs.

Historical and Real Time Gadgets

The Cisco Unified Intelligence Center release 12.6.1 ES02 and above, supports Historical and Real Time report gadgets in agent and supervisor desktops in VPN-less deployments. To configure the Historical and Real Time report gadgets, refer to the *Configure Historical Report Gadgets in Cisco Finesse* section in *Cisco Unified Intelligence Center User Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-intelligence-center/products-user-guide-list.html.

Note

- Stock reports and custom reports can be viewed in VPN-less supervisor desktop. However, before viewing the custom reports as gadgets in VPN-less supervisor desktop, run the command, **set cuic properties allow-proxy-custom-report on**.
- To configure the data set size for Historical report, run the command, **set cuic properties vpnless-response-size-ht**. By default, the data set size for HT is set to 8MB.
- To configure the data set size for Real Time report, run the command, **set cuic properties vpnless-response-size-rt**. By default, the data set size for RT is set to 300KB.

  If the data set size is more than the configured value, the gadget will display the following error message:

  ```
  Failed to load the gadget. Response size is more than allowed limit. Please contact
  your Administrator.
  ```

  This limitation is applicable on VPN-less deployments only. For more information about configuring the data set size, see *set Cisco Unified Intelligence Center properties* section in *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.

Security Guidelines

For information about security guidelines, see the *Security Guidelines for Reverse-Proxy* in *Security 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.

Caveats

Reverse-proxy deployment allows agents and supervisors to concurrently access the Cisco Finesse desktop from both LAN and through reverse-proxy. After configuring the reverse-proxy, when the agents and supervisors access the Finesse desktop through LAN, all the features work seamlessly. However, when the Finesse desktop is accessed through the reverse-proxy, the caveats are as follows:

- Finesse IP Phone Agent (FIPPA) isn’t supported.
- Administrative applications and the corresponding APIs of Finesse, IdS, and Cisco Unified Intelligence Center aren’t supported.
- Multiple devices accessing the Finesse desktop through Network Address Translation (NAT) isn’t supported.
- Multiple users accessing the VPN-less desktop from behind a common proxy isn’t supported when multiple sites are involved.
- If threshold images are used in Live Data, Real Time, and Historical gadgets, add the reverse-proxy rules to allow images to be accessed through reverse-proxy. For more information on threshold images rules, refer to the Nginx TechNote article section.
- After upgrading Finesse to 12.6(1), Cisco Unified Intelligence Center must be upgraded to 12.6(1) for the Live Data (LD) gadgets to work. Refer to the Unified CCE Compatibility Matrix for general

compatibility between CUIC and Finesse when accessed through the Contact Center network or the reverse-proxy.

- Third-party gadgets on the Finesse desktop could be incompatible with the reverse-proxy deployment. For more information on gadget compatibility, see the Determine Gadget Compatibility, on page 279 section.
- **Finesse API Compatibility:**
  - Finesse Desktop supports only the WebSocket notification mechanism over reverse-proxy. For third-party servers, BOSH or XMPP over TCP communication through reverse-proxy isn’t supported.
  - When the SystemInfo API is accessed through a reverse-proxy, the authorization headers are required.

CHAPTER 19

Webex Experience Management Integration

Experience Management Overview

Note To enable this feature in Packaged CCE, install the following patches:

- ICM 12.5(1)_ES7
- CVP 12.5(1)_ES6
- Cloud Connect 12.5(1)ES1
- Finesse 12.5(1)ES2

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, Packaged CCE supports:

- Customer experience surveys - Set up and send surveys to customers, after an interaction, to collect feedback about their interaction.
- Experience Management Post Call Survey
- 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 Voice 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 Packaged CCE for Experience Management Voice, SMS and Email Survey, on page 298 .

The Experience Management Post Call Survey call works just like a regular call from the Unified CCE point of view. Scripts are invoked, CVP refers the call to Experience Management, and the customer uses the keypad on a phone to respond to questions asked during the survey. During Experience Management Post Call Survey, the call context information is retrieved from the original customer call.

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 Packaged CCE, follow this task flow:

Sequence	Task
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.
2	Experience Management Activation Team creates: **1.** Accounts and provisions the same. **2.** Default spaces and metric groups for your accounts. To know more about creating spaces, see Space Creation. **3.** Standard questionnaires for Experience Management Post Call Survey and publishes the same. To know more about creating questionnaires, see Questionnaires.

Sequence	Task
3	After creating the account and provisioning, 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.
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.
5	Provision Experience Management service using CLI on Cloud Connect. For more information, see Provision Experience Management Service on Cloud Connect, on page 298.
6	Ensure that the Cloud Connect publisher and subscriber are installed. For more information, see the *Create VM for Cloud Connect Publisher* and *Create VM for Cloud Connect Subscriber* sections in *Cisco Packaged Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-installation-guides-list.html
7	Configure Cloud Connect in Unified CCE Admininstration. For details on how to do this, see *Configure Cloud Connect* section in *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.
8	Import the following certificates to the CVP Call Sever: • 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.
9	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.
10	Configure Packaged CCE Experience Management. For more information, see the topic Configure Expanded Call Variables, on page 299
11	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 301

Sequence	Task
12	Modify CCE scripts. For more information, see *Experience Management Scripting* in *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html. Associate the CCE script with the Call Type created in the previous step.
13	Add Experience Management gadgets into Finesse desktop layout. For more information, see Cisco Webex Experience Management Gadgets.

Provision Experience Management Service on Cloud Connect

Provision Experience Management service using the following CLI on Cloud Connect.

```
set cloudconnect cherrypoint config
```

Configure Cloud Connect in Packaged CCE Admininstration. For details on how to do this, see *Configure Cloud Connect* topic at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

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/.

Configure Packaged 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 299
- Upload Audio Files for Questions in Experience Management, on page 300
- Configure Dialed Number and Call Type, on page 301
- Associate Survey to Call Type in Unified CCE Admin, on page 302

Configure Expanded Call Variables

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

Step 2 From the list of ECC variables, click on the `user.microapp.isPostCallSurvey` variable to open it.

a) Set **Max Length:** to 1.
b) Check the **Enabled** checkbox.
c) Click **Save**.

In your CCE routing scripts, remember that, at script start, the default behavior of Post Call Survey equals **enabled**, even if **user.microapp.isPostCallSurvey** has not yet been set in the script. You can turn **off** Post Call Survey 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*.

Step 3 Create a new ECC variable with **Name:**`user.CxSurveyInfo`.

a) Set the **Max Length** to 133 for Type 10 VRUs. For all other routing clients, set **Max Length** to 120.
b) Check the **Enabled** check box.

Step 4 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 5 Populate the **POD. ID** variable.

For more information on populating this variable, refer to the topic Configure POD.ID.

Step 6 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 18: 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 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 *Expanded Call Variables* in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-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 *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-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

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

Step 2 Click **New** to open the Call Type window.

Step 3 Enter the Name of the Call Type for Experience Management survey.

Step 4 Click **Save** You will be re-directed to the List window and the confirmation message is displayed.

Step 5 Navigate to **Overview** > **Call Settings** > **Route Settings** > **Dialed Numbers**.

Step 6 Click **New** and complete the following fields:

Field	Required?	Description
Dialed Number String	Yes	This value is used to route the call.
Description	No	Enter a maximum of 255 characters to describe the dialed number string.
Department	No (Yes for departmental administrators)	A departmental administrator must select one department from the pop-up list to associate with this dialed number. The list shows all this administrator's departments. When a departmental administrator selects a department for the dialed number, the pop-up list for call type includes global call types and call types in the same department as the dialed number. A global administrator can leave this field as Global (the default), which sets the dialed number as global (belonging to no departments). A global administrator can also select a department for this Dialed Number. When an administrator changes the department, selections for call type are cleared if the selections do not belong to the new department or the global department.
Site	Yes	The **Site** field displays Main by default for Packaged CCE 2000 Agents deployment. For Packaged CCE 4000 Agents and 12000 Agents deployments, **Site** is a mandatory field and has no default value. To add a site: **a.** Click the **magnifying glass** icon to display the list of sites. **b.** Select the site.

Field	Required?	Description
Peripheral Set	Yes	This field is available only in Packaged CCE 4000 Agents and 12000 Agents deployments. To add a peripheral set: **a.** Click the **magnifying glass** icon to display the list of peripheral sets configured for the selected **Site**. **b.** Select the peripheral set.
Routing Type	Yes	From the drop-down menu, select External Voice. These calls are referred to as external because they typically come from outside of the enterprise through a gateway. External Voice is the selection for calls that come in from customers and must be answered by agents or sent to the VRU.
Media Routing Domain	Yes	The Media Routing Domain associated with the dialed number. The selection of Routing Type determines what appears in this field. Because the **Routing Type** is **External Voice**, the Media Routing Domain is always Cisco_Voice.
Call Type	Yes	Click on the magnifying glass icon. From the **Select Call Type** pop-up window, enter or select the call type you created in step 3. Associating a dialed number with a call type ensures appropriate routing and affects reporting.
Ringtone Media File	No	This field appears when the **Routing Type** is **External Voice**. Enter file name of the custom ringtone for the user-defined Dialed Numbers, a maximum of 256 characters without any spaces.

Step 7 Click **Save**. You will be re-directed to the List window and the confirmation message is displayed.

Step 8 To create the PCS dialed number refer topic, Configure Packaged CCE for Post Call Survey, on page 188.

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.

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

The list of all the **Call Types** are displayed.

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 20

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, Packaged 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 Packaged CCE, perform the following 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 *Create VM for Cloud Connect Publisher* and *Create VM for Cloud Connect Subscriber* sections in *Cisco Packaged Contact Center Enterprise Installation and Upgrade Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-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 307.

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 Packaged CCE Experience Management integration. For more information, see Configure Packaged CCE for Digital Channel Survey , on page 307.

Step 9 Configure Call Type and Dialed Number. For more information, see Configure Call Type, Dialed Number, and Survey Association, on page 309.

Step 10 Modify CCE Scripts. For more information, see *Scripts for Experience Management* in *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Associate the CCE script with the Call Type created in the previous step.

Step 11 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
```

Configure Cloud Connect in Packaged CCE Admininstration. For details on how to do this, see *Configure Cloud Connect* topic at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Configure Packaged 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 307
- Configure Call Type, Dialed Number, and Survey Association, on page 309
- Associate Survey to Call Type in Unified CCE Admin, on page 309

Configure Expanded Call Variables

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

Step 2 From the list of ECC variables, click on the `user.microapp.isPostCallSurvey` variable to open it.

a) Set **Max Length** to 1.
b) Check the **Enabled** check box.
c) 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 3 Create a new ECC variable with **Name:**`user.CxSurveyInfo`.

a) Set **Max Length** to 80.

b) Check the **Enabled** check box.

Step 4 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 5 Populate the **POD. ID** variable.

For more information on populating this variable, refer to the topic Configure POD.ID , on page 308.

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 19: 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 *Expanded Call Variables* in the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Configure Call Type, Dialed Number, and Survey Association

Step 1 In **Unified CCE Administration**, navigate to **Overview** > **Call Settings** > **Route Settings** > **Call Types** and click **New** to create a Call Type.

For more information on how to create Call Type, refer to the section *Call Type* in *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html

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 309.

Step 3 Navigate to **Overview** > **Call Settings** > **Route Settings** > **Dialed Numbers** and click **New** to create Dialed Number.

Note Select **Enterprise Chat and Email** as routing type for dialed number.

For more information on how to create Dialed Number, refer to the section *Dialed Number* in *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html

Step 4 Click **Save**. You will be re-directed to the List window and the confirmation message is displayed.

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.

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

The list of all the **Call Type** are displayed.

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.

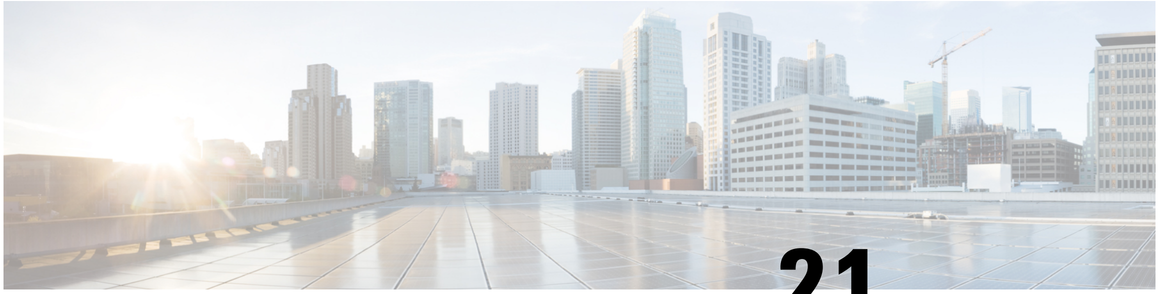
a) The Experience Management tab is enabled with the following options:

- 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 21

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.

Whisper Announcement is supported for blended outbound agents when they receive inbound calls.

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 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.
- 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. Media Server, on page 117.
2. Create Whisper Announcement Audio Files, on page 312.
3. Deploy Whisper Announcement Audio Files to Media Server, on page 313.
4. Configure Whisper Service Dialed Numbers, on page 313.
5. Add Whisper Announcement to Routing Scripts, on page 314.
6. Fail-Safe Timeout for Whisper Announcement in Unified CCE, on page 315.

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 315.

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.

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*.

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 315.

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

Ensure that your call routing scripts can access the Whisper Announcement audio files that you stored on a CVP media server. If you configure a default media server, and you store the audio files on the default server, you may not have to add any additional nodes to the scripts. For more information, see . To test the access, see Test Whisper Announcement File Path, on page 315.

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 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 **value**. This value is 20 seconds and represents the maximum Whisper Announcement play time that Unified CCE uses to calculate its report data.

The value was chosen 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. While the value is configurable in Unified CCE, changing the value is not supported in Unified CCE.

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:

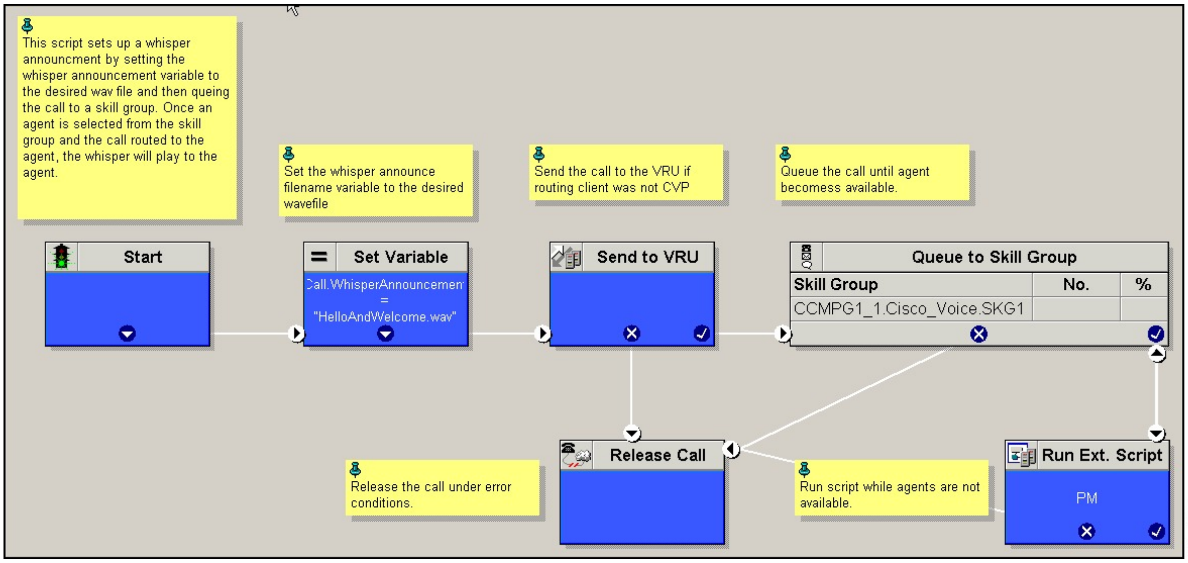
- **WA.ICMS**—This script plays a Whisper Announcement.
- **WA_AG.ICMS—**This script plays both a Whisper Announcement and an Agent Greeting to play on the same call flow.

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.



This script sets up a whisper announcment by setting the whisper announcement variable to the desired wav file and then queing the call to a skill group. Once an agent is selected from the skill group and the call routed to the agent, the whisper will play to the agent.

Set the whisper announce filename variable to the desired wavefile

Send the call to the VRU if routing client was not CVP

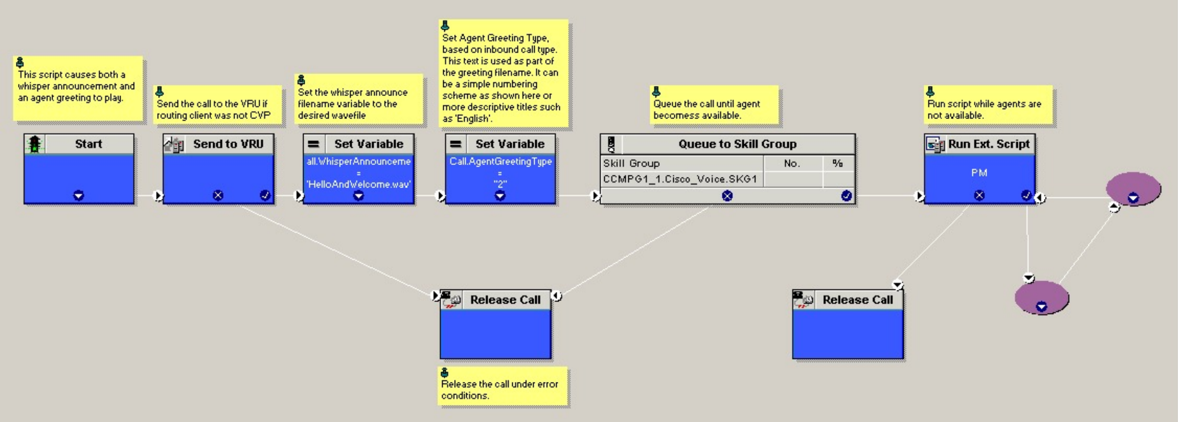
Queue the call until agent becomess available.

Release the call under error conditions.

Run script while agents are not available.

WA_AG.ICMS Script

This script causes both a Whisper Announcement and an Agent Greeting to play.



This script causes both a whisper announcement and an agent greeting to play.

Send the call to the VRU if routing client was not CVP

Set the whisper announce filename variable to the desired wavefile

Set Agent Greeting Type, based on inbound call type. This text is used as part of the greeting filename. It can be a simple numbering scheme as shown here or more descriptive titles such as 'English'.

Queue the call until agent becomess available.

Run script while agents are not available.

Release the call under error conditions.

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:

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.

Administration and Usage

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 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.

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.

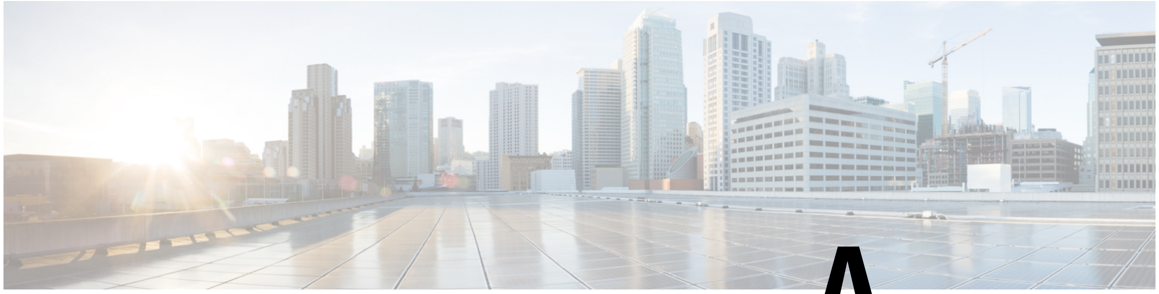
Whisper Announcement in Agent Desktop Software

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

Avaya Support

Avaya Support

Prerequisite

Make sure you have Avaya Automatic Call Distribution (ACD) versions that are compatible with Packaged CCE deployments. For more information, see the *Contact Center Enterprise Solution Compatibility Matrix* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-device-support-tables-list.html.

Avaya Support Overview

Support for Avaya ACD int has been provided in Packaged CCE 4000 and 12000 Agent deployments. You can maintain an Avaya Peripheral Gateway (PG) in a Packaged CCE environment and use its intelligent contact center routing capability to route calls to geographically distributed contact center sites.

For detailed information about the required Avaya configurations, see chapter *Unified ICM Software Configuration* in the *Cisco Unified ICM ACD Supplement for Avaya Communication Manager Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-technical-reference-list.html.

Note Note that Avaya PG must be deployed on a separate VM. Also Avaya agents cannot be associated with a department.

Tools that Support Avaya Configurations

Configuration Manager Tools and nodes in the Script Editor have been enabled to facilitate the support for Avaya ACD in Packaged CCE 4000 and 12000 agent deployments. For the complete list of nodes and tools, see the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

The following restrictions apply to the tools that support Avaya PG configurations.

Table 20: Configuration Manager Tool Restrictions

Configuration Manager Tool	Restriction
Agent Explorer	• Only supports Avaya PG configurations • Does not support selecting persons who are already associated with the CUCM Peripheral agents
Person List	Does not list persons who are already associated with the CUCM peripheral agents
Dialed Number/Script Selector List	Supports addition of Dialed Numbers for Avaya Agents and NIC Routing Clients
Skill Group Explorer	Only supports Avaya PG configurations
Bulk Configuration Tools	The following bulk tools only support Avaya PG configurations. • Agent Bulk Insert • Dialed Number Bulk Insert • Skill Group Bulk Insert • Agent Bulk Edit • Dialed Number Bulk Edit • Skill Group Bulk Edit • Person Bulk Insert • Person Bulk Edit

For design details, scalability constraints and sizing factors, see the *Solution Design Guide for Cisco Packaged Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-technical-reference-list.html.

You can also view historical and real-time stock reports for Avaya ACD. For more information, see the *Cisco Packaged Contact Center Enterprise Reporting User Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

APPENDIX **B**

Do Not Call Table

Do_Not_Call Table

The Do_Not_Call table includes all the phone numbers and extensions that, when matched exactly, are not dialed during an Outbound Option campaign.

The following table lists the Do_Not_Call table column names and provides their descriptions.

Column Name	Type	Description
Phone	VARCHAR(20)	The Do Not Call phone number.
PhoneExt	VARCHAR(8)	The extension for the Do Not Call phone number. **Note** Although the phone number extension is imported into the table, it is currently not used for any dialing operations.

Do Not Call Considerations

Consider the following for the Do Not Call feature:

- When you upgrade to or downgrade from Cisco Unified CCE, Release 11.6(1), the Do Not Call table is not available. Therefore, import the Do Not Call table again after upgrade or downgrade.
- Do not configure multiple Do Not Call import rules.
- A customer number is dialled even if the number is listed in the Do Not Call table. This occurs when:
  - the Campaign Manager restarts.
  - one of the routers is not available during the import of the Do Not Call records.
- Do not perform manual operations on database including database replication.

APPENDIX C

ICM to ICM Gateway Support

ICM to ICM Gateway Support

Prerequisite

Make sure you have ICM-to-ICM Gateway versions that are compatible with Packaged CCE deployments. For more information, see the *Contact Center Enterprise Solution Compatibility Matrix* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-device-support-tables-list.html.

ICM-to-ICM Gateway Overview

Support for ICM-to-ICM Gateway has been provided in Packaged CCE 4000 and 12000 Agent deployments, where Packaged CCE can act as a client or a server. The call requests can be routed from Packaged CCE to remote Unified CCE/Unified ICM deployments and vice versa.

ICM-to-ICM Gateway extends the ICM software capability by allowing agents to simultaneously pre-route/post-route calls, and supply additional call-related information to a second agent on a different ICM. This enables the initial agent to pass on gathered information without the customer's needing to repeat it to the second agent.

Following are some business scenarios where ICM-to-ICM Gateway functionality can be useful.

- A customer calls the institutional department of a financial corporation for customer service assistance with a company-sponsored 401k. The customer then asks to be transferred to the retail department to obtain assistance with a personal account.
- Two corporations (for example, a bank and an insurance company), each of which has a contact center that uses an ICM, merge. It may often be desirable to transfer a call between the two companies; for example, to sell insurance to a bank customer.
- A customer calls a hotel to make a reservation. The hotel agent then asks the customer if he/she also needs to rent a car, and then transfers the customer to a car rental agent.
- A company uses an outsourcer to handle part of its overflow traffic. For example, the company service department handles paid support calls in-house but transfers warranty service requests to the outsourcer.

- A multi-national corporation encompasses several geographic regions; each geographic region has its own ICM.

In all these cases, ICM-to-ICM Gateway enables the call-related data to be transferred along with the call so the customer does not need to supply this information again.

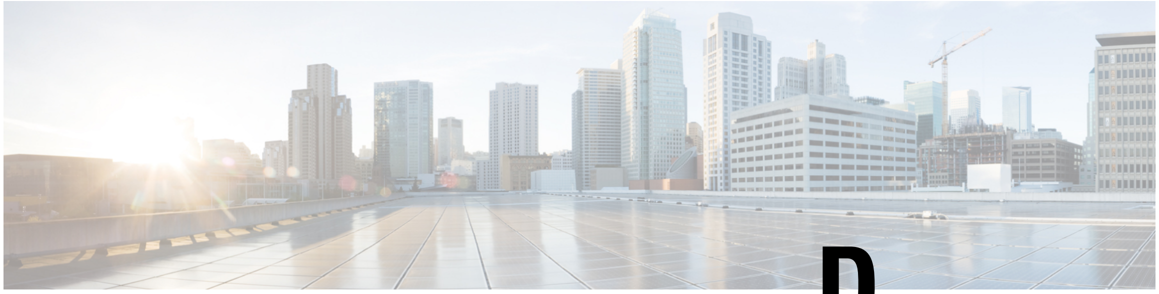
For more information about ICM-to-ICM Gateway call flows and configuration, see the *ICM-to-ICM Gateway User Guide for Cisco Unified ICM Enterprise & Hosted Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-and-configuration-guides-list.html

Tools Supported for ICM-to-ICM Gateway

Configuration Manager Tools and nodes in the Script Editor have been enabled to facilitate the ICM-to-ICM Gateway capability in Packaged CCE 4000 and 12000 Agent deployments. For the complete list of nodes and tools, see the *Cisco Packaged Contact Center Enterprise Administration and Configuration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html.

Note The **Application Gateway List** tool only supports remote ICM configuration.

For design details, scalability constraints and sizing factors, see the *Solution Design Guide for Cisco Packaged Contact Center Enterprise* at https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-technical-reference-list.html.



APPENDIX D

Reverse Proxy Automated Installer

Introduction

The Reverse proxy installer and associated artifacts can be downloaded from https://software.cisco.com/download/home/283613135/type/284259728/release/12.6(2)

The content in this chapter is provided as a guidance for customers to install and configure the Cisco provided reverse proxy artifacts, which ships with an embedded Nginx based OpenResty® reverse proxy.

For information on the deployment details and the pre-requisites required, see VPN-less access to Finesse desktop, on page 269

Cisco does not support install or configuration requests for custom reverse proxy images and network configurations related issues. Queries that are related to this subject can be discussed on Cisco community forums.

For older format of VPN-less Finesse, see Cisco Finesse 12.6 ES07 Readme.

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 ES07 and above
- Cisco Unified Intelligence Center - 12.6 ES03 and above
- Cisco Identity Service - 12.6 ES03 and above
- Cisco Unified CCE and Packaged CCE - 12.0 and above
- Cisco Cloud Connect-12.6(2)
- ADFS 3.0 for being used as IDP in SSO deployments

Note To use VPN-Less access to Finesse desktop feature, you must upgrade Finesse, Cisco IdS, and Cisco Unified Intelligence Center to releases mentioned above.

If you are using LiveData 12.6(1), you must upgrade LiveData to releases mentioned above.

Packaged CCE and Unified CCE 2k deployments must be on 12.6 version of CCE to support the coresident deployment of Livedata (LD) and Cisco Unified Intelligence Center.

Related Topics

Performance

Background Information

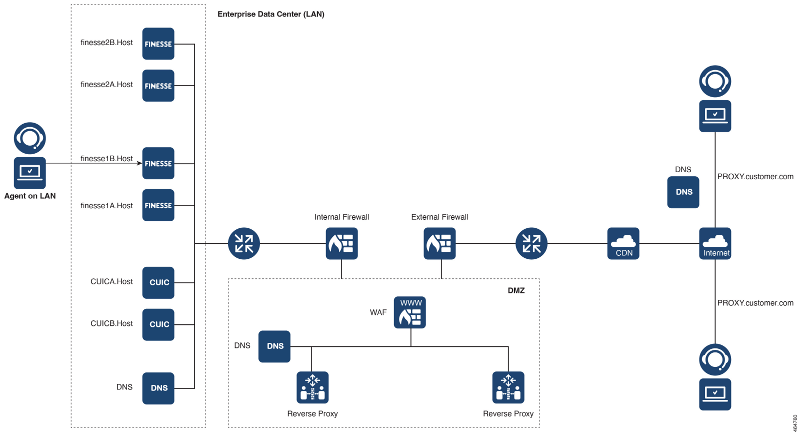
This deployment model is supported for the Unified CCE and Packaged CCE solutions.

Deployment of a reverse-proxy is supported (available from 12.6 ES07) as an option to access the Cisco Finesse desktop without connecting to a VPN. This feature provides the flexibility for agents to access the Finesse desktop from anywhere through the Internet.

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 Minimum and additional requirements 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

VPN-less access to Finesse desktop, on page 269

Reverse Proxy Installer

The Reverse Proxy Installer (referred to as Installer in this document) is an automated tool to make the Reverse Proxy deployment for Cisco Unified Contact Center a simple and error free exercise.

This Installer replaces the older VPN-less Finesse configuration provided as part of the 12.6 ES 01 and ES07 releases, which required manual install of the proxy along with editing of the provided rules for creating a VPN-less deployment.

The following are the Cisco Unified Contact Center solution components which are supported by the Reverse Proxy Installer:

- Cisco Finesse
- Cisco Identity Service
- Cisco Unified Intelligence Center
- Cisco Unified Cloud Connect

Installer Components

Reverse-proxy within container

The Installer deploys the latest load tested and qualified OpenResty® reverse-proxy binary, in a docker container and seeds it within the required configurations automatically. (Follow the process in the sections below).

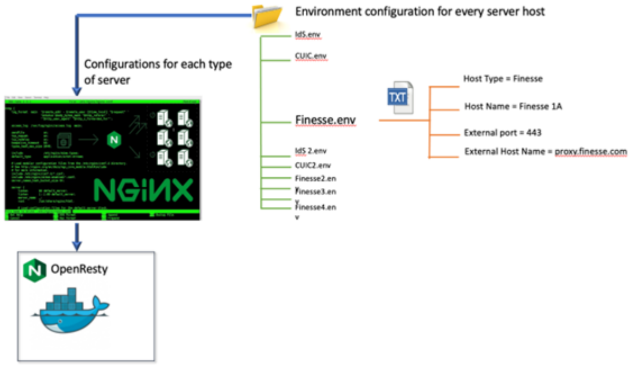
This makes it very easy to run the reverse-proxy configuration required to support the VPN-less infrastructure, thus simplifying the deployment immensely, without requiring compilation or the knowledge of NGINX configuration.You dont need to know how to compile or install the open source NGINX.

Containerized proxy instances are also more secure due to its locked down nature and provides an additional barrier for an intruder to overcome compared to a proxy process running on bare a metal operating system.

Proxy Configuration

Environment Files and Templates

The proxy configuration has been split into environment configuration and proxy rule configurations is also known as templates.



The simple and unique environment values which differentiate each upstream component server is collated within the respective environment files, with one file for each upstream component server. These are automatically combined into the proxy rules for each unique type of upstream component server (For example: Cisco Finesse, Cisco Identity Service etc) by the Installer and are then fed into the included OpenResty® NGINX proxy which then proceeds to deploy these rules.

This allows easy instantiation of any number of supported upstream component server hosts as required by adding new environment files corresponding to new servers.

The proxy configuration rules, known as rule templates, contain the necessary NGINX rules to access the server and you don't have to understand or change them.

This also makes Installer upgrades easy as the environment files containing the configurations are rarely changed and can be persisted through multiple Installer changes without requiring much NGINX expertise.

Sample environment files

The Installer archives come with a sample environment that can be used as a starting point to create a new VPN-less Finesse deployment.

Each file it contains is an environment for a unique type of upstream host natively supported by the configurations /Reverse Proxy rules provided, such as Finesse, Unified Intelligence Center and Cisco IdS.

The administrator should clone this directory and create multiple copies of each environment corresponding to each upstream component host which has to be exposed via the proxy and supply this directory location to the Installer which will then proceed to instantiate each host configuration based on the supplied parameters contained in the environment file.

Reverse Proxy Installer components	Directory/ File Name	Description
Reverse proxy instance	*reverse-proxy-openresty-container/* |	Contains the docker container image that is used to create the container instance. This contains the OpenResty® NGINX proxy and other dependent libraries and modules OpenResty® version packaged: Latest CentOS version 7 based OpenResty® available in the docker hub.
Component Configuration Templates	*reverse-proxy-openresty-configs/*	Contains the OpenResty® configuration templates. These templates are used to generate final OpenResty® configurations from provided deployment configuration data.
Host OS configurations:	*reverse-proxy-os-configs/*	These templates are used to generate the final OpenResty® configurations from the deployment configuration data provided. Contains the OS configurations for hardening the host. CentOS version 7 is the only supported OS. You must manually install the OS configurations using the `install_os_settings.sh` that is available in this directory. **Note** The OS configurations are tested with OpenResty® version 1.19. These configurations are expected to work with other distributions. You may need to make some minor updates as required.
Installer configuration	*installer.env*	Contains the configuration data for the Reverse Proxy Installer.

Reverse Proxy Installer components	Directory/ File Name	Description
Proxy configuration	*sample_envs/*	Contains the sample env configuration data for reference. Use this sample env when you are preparing the configuration data for your deployment.
Launcher scripts	*proxy_launcher.sh*	Launcher script to perform the **start|stop|reload|clear_cache** operations for a given Installer instance.
version.txt:		Contains Installer versions. Reverse proxy Installer creates all the configurations afresh on every restart and no configuration is retained. Any additional changes made to the existing proxy configurations are lost after the restart..

Upgrade notes for 12.6(1) ES01-7 based reverse proxy configurations

The configuration formats have been modified with the new Installer-based configuration and can't be reused as is. The information contained, however, can be easily extracted and plugged into the new Installer configuration, and the format will not be changed further.

The CLI configurations and proxy map data do not need to be altered. However, as previously described, the manner in which the upstream component server hosts and their associated configurations are provided to the reverse proxy instance has now changed, see the section *Environment Files and Templates* in Reverse Proxy Installer, on page 327

The following are some important points to consider when you upgrade your reverse proxy instance using the automated Installer:

- The data required in the component host environment will match the individual values replaced in the template configurations using the ## Must-Change notations from the older configurations. This can be used as a reference to fill the data if required.
- *Tmpfs* is not used in the new Installer, and earlier configurations run with "tmpfs" can be ignored. However, disk subsystem slowness can impact the proxy performance and needs to be performant.

Install and Operations

Setup reverse proxy

To set up the reverse proxy server, refer to the following:

Related Topics

Proxy Hardware requirements, on page 331
Prepare Host, on page 331
Install the reverse proxy Installer package, on page 332
Configure host OS, on page 332
Configure proxy hardware resources and other critical runtime options, on page 333

Proxy Hardware requirements

The following are the hardware requirements to set up a reverse proxy server for a 2K cluster that includes Cisco Finesse, Cisco Identity Server (IdS), Cisco Unified Intelligence Center, Live Data, Enterprise Chat and Email, and Cisco Cloud Connect:

Following are the hardware requirements:

- CPU: 2vCPU for 2000 agents and 4vCPU for 4000 agents Deployment.
- Memory: 8 GB
- Disk: 80 GB
    - Cache disk space requirements for the following components:
        - **Finesse: 3 GB for one upstream**
        - **CUIC: 200 MB for one upstream**
        - **IdS: There is nothing cached.**
- Ethernet interfaces must be gigabit speed and connected to gigabit ethernet switches. 10/100 ethernet is not supported.

Note Disk slowness can cause proxy performance to be hampered. Please monitor the solution to ensure the disk has adequate IO throughput.

Running the linux command **dd if=/dev/zero of=/root/junk bs=2k count=1000 oflag=dsync** should show a minimum of 5MB per second of throughput and completion time of less than 0.3 seconds to write the data out.

Prepare Host

To prepare the host, follow the steps below:

Step 1 Install the latest build of CentOS Linux 7.x (7.9 or later) .

Step 2 To Install the **envsubst utility**, run the command **yum install gettext** .

```
yum install gettext
```

Step 3 Install the Docker. For instructions, see the Docker documentation at https://docs.docker.com/get-docker/

Note Uninstall podman on Cent OS or RHEL if already installed as the podman conflicts with the Docker installation. Run the following command to uninstall podman:yum erase podman buildah

Step 4 Perform the post-installation steps to manage Docker. For instructions, see the Post-installation steps for Linux section at https://docs.docker.com/engine/install/linux-postinstall

Step 5 Run the following command to install logrotate on the host:.

- **yum install logrotate -y command**

Step 6 Run the following commands to uninstall or stop the firewall daemon service on Cent OS:

- **sudo systemctl stop firewalld**
- **sudo systemctl disable firewalld**
- **sudo systemctl mask --now firewalld**

Step 7 Run the following commands to install the iptables service on Cent OS::

- **sudo yum install iptables-services -y**
- **sudo systemctl start iptables**
- **sudo systemctl enable iptables**

Install the reverse proxy Installer package

To install the package:

Step 1 Download or copy the Installer zip on the host.

Step 2 Extract the archive (.zip) to the location where you need the Installer to be running from.

Configure host OS

The following are the OS hardening configurations for the reverse-proxy host that are included in the Installer/reverse-proxy-os-configs/ folder:

- Kerner hardening configurations, that is sysctl configurations
- Logrotate config
- CentOS version 7

The Installer script is provided to install the required configurations automatically. Various options can be provided in the script to control the installation and configuration.

Run the **`install_os_settings.sh`** script with the required options. Check the usage information below:

```
USAGE: ./install_os_settings.sh [OPTIONAL_ARGS]
OPTIONAL_ARGS: -k -l -i -p <source-ip1> -p <source-ip2> ... -s <interface1> -s <interface2>
 ... -r <interface1:source-ip1> -r <interface1:source-ip2> -r <interface2:source-ip1> ...
```

```
-k: configure kernel hardening
-l: configure logrotate for given log directory
-i: configure iptables
-p: allowed source ip for ICMP ping messages. By default ICMP ping is blocked for all hosts.
 This option is ignored if -i or iptables configuration option is not given
-s: network interface name to allow SSH access to. By default ssh access is blocked for all
 hosts. This option is ignored if -i or iptables configuration option is not given
-r: disable rate limits for a source-ip on an interface. Provide value as INTERFACE:SOURCE_IP.
 By default rate limits applies for all. This option is ignored if -i or iptables
configuration option is not given

Example usage: ./install_os_settings.sh -k -l ~/reverse_proxy/proxy25.autobot.cvp/logs -i
-p allowed.host.for.ping.1 -p allowed.host.for.ping.2 -s ssh_interface1 -s ssh_interface2
-s -s ssh_interface3 -r interface1:host1 -r interface1:host2 -r interface2:host1
```

Note If you are installing the reverse-proxy to proxy a specific component, for example, Cloud Connect, you must not install the iptable rules for other components. As an administrator, you must select the component rules and the port values to be installed and configured for the iptables. The port value may change depending on the deployment. The script does not control the customization of iptable rules.

Related Topics

Configure Log Rotation

Configure proxy hardware resources and other critical runtime options

The Installer script, `installer/proxy_launcher.sh` that is used to deploy reverse proxy takes the following input arguments:

- `installer.env`: Path to `installer.env` file containing Installer configuration data.
- `proxy_env_dir/`: Path to `proxy_env_dir/` file containing proxy configuration data.

Note The `installer.env` file contains configuration properties to configure Installer options. The sample file is provided in the Installer package, and it should be used as a reference configuration to prepare actual configuration.

To configure the proxy hardware resources and other runtime options, follow the below steps:

Step 1 Copy the sample file, `installer.env` to any other directory, and rename it. You can use the proxy name or the customer name that maps to a particular proxy instance, if there are multiple proxy instances running on the same host.

Step 2 Modify the installer options as required. Options included in the configuration file with their intended purpose.

Configure SSL certificates

The environment configuration file for each component includes an SSL CONFIG section that has configurations to set up the SSL connector for the component. In addition, the configurations are used to configure the following:

- Either custom certificates that you have generated manually or certificates that the Installer has generated can be used for reverse proxy.
  - If you choose to use the custom certificate that is either, CA signed or self-signed, which you have generated, place the certificate inside the ssl directory mentioned in the option HOST_SSL_VOL(defaults to ${HOST_WORKING_DIR}/ssl).
  - You can also allow the Installer to generate the self-signed certificate. When starting the Installer/proxy_launcher.sh script, set the CREATE_SELF_SIGNED_SSL_CERT option to true. For more information, see Configure proxy hardware resources and other critical runtime options. The Installer generates the certificate and includes it in the ssl directory mentioned in the option HOST_SSL_VOL(defaults to ${HOST_WORKING_DIR}/ssl), only if the required certificate and key names are not present in the ssl directory. The Installer does not overwrite the existing files.

  These certificates are used to configure the SSL connector for individual component configurations.
- Supported TLS protocol versions
- Supported TLS ciphers
- SSL session cache size and timeout
- SSL stapling configurations
- Mutual TLS authentication for upstream connections: By default, this option is disabled. To enable this option, modify the following configurations:
  - Set the NGX_PRXY_SSL_VERIFY option to "on"
  - NGX_PRXY_SSL_TRUST_CRT → Trust file containing certificate of upstream being proxied. Certificate from this file will be verified by NGINX against what is provided by upstream during the TLS handshake.

Note Self-signed certificates to be used only for testing and development purposes and CA signed certificates are mandatory for production deployments. 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.

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.

Following parameters are used to configure stapling **NGX_SSL_STAPLING & NGX_SSL_STAPLING_VERIFY** on the respective component's env files. They are set by default "off".

Configure Mutual TLS Authentication Between Reverse-Proxy and Components

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, by default this feature is 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.

Configure the Mapping File

Refer to Host-Mapping file for network translation, on page 276.

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 335.

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.

Configure deployment environment configurations

The environment configuration data is the main input that the Installer needs to generate the actual proxy configurations from the templates. There is a sample environment data, Installer/sample-envs/ that is provided within the installer.zip. The sample environment data contains up-to-date reference envs for all supported components. The following are the contents of the sample env directory:

```
installer/sample_envs/
 |- core.env
 |- dirs.env
 |- finesse.env
|- ids.env
|- cuic.env
|- livedata.env
|- chat.env
|- cloudconnect.env
```

These property files are divided into 3 categories:

- `core.env` : **Mandatory**: File containing OpenResty® NGINX core configurations data. This is required to configure OpenResty® NGINX core configurations.

  This environment config file contains data for *reverse proxy core config template files*. Core config files include details specific to the running NGINX instance and is applied generally to all the components until or unless it is overridden at the component level.

  The core config template file includes:

  - `cache.conf`: Template file containing cache configurations
  - `common.conf`: Template file containing common configurations
  - `logging.conf`: Template file containing logging configurations
  - `maps.conf`: Template file containing constants and other variable configurations
  - `rate_limit.conf`: Template file containing rate limit configurations
  - `static.conf`: Template file containing static configurations
  - `ssl_config.conf`: Template file containing ssl connector configurations for common server blocks like status endpoint and static files endpoint

  Values provided in the `core.env` file will be used to substitute all the placeholders in the above files.

- `dirs.env` : **Mandatory**: File containing various OpenResty® NGINXdirectory paths as per OpenResty® installation. This is required to configure directory paths in the configuration templates.

  This environment data contains information regarding the OpenResty® installation directory structure. Defaults are included as per the default OpenResty® installation.

  ```
  • # Directory location for various openresty folders required to
  • # configure configurations accordingly.
  •
  • # Home directory for openresty nginx installation
  • NGX_HOME="/usr/local/openresty/nginx"
  • # Openresty directory containing static resources
  • NGX_HTML_DIR="${NGX_HOME}/html"
  • # Openresty directory containing lua resources
  • NGX_LUA_DIR="${NGX_HOME}/lua"
  • # Cache directory where various resources for components will be cached
  • NGX_CACHE_DIR="${NGX_HOME}/cache"
  • # Openresty directory containing SSL resources like certs, keys etc.
  • NGX_SSL_DIR="${NGX_HOME}/ssl"
  • # Openresty directory where openresty logs will be put
  • NGX_LOG_DIR="${NGX_HOME}/logs"
  • # Openresty directory containing NGINX configurations - core configs, components
  configs etc.
  NGX_CONF_DIR="${NGX_HOME}/conf"
  ```

- `component envs` : **Optional**: Files containing configuration data for proxied solution components like Finesse, Cisco IdS, Unified Intelligence Center, Live Data, Cisco IM&P etc. One environment config file has to be created per upstream solution component being proxied.

Some properties are mandatory in component environment config files without which configs will not be generated for those components. These properties are:

- `TEMPLATE_TYPE` : Defines which type of upstream component is being configured, so that the correct templates can be referred to generate the actual configurations. Value can be one of `finesse`, `ids`, `cuic`, `livedata` and `chat`
- `NGX_COMP_DIR_NAME` : Defines the output directory where configuration files for the component will be generated. Final output location for the files will be `./configs_out/conf/components/<NGX_COMP_DIR_NAME>/`. Also, this directory will be used to form the file including the paths in various configuration files of the component.

Note Ensure that each environment config file has a unique output directory name (`NGX_COMP_DIR_NAME`) and hostname (`NGX_COMP_HOSTNAME`).

Other properties are different for different components and defaults for all the components are provided in their respective env files.

Follow the below steps to configure these options:

Step 1 Copy the **installer/sample-envs/** directory and the **installer/installer.env** file to a separate directory and modify it. After the files are copied to a new directory at **~/proxy_config/proxy_instance_name.** Rename the files such that they can be mapped to a running proxy instance.

Step 2 Modify the **core.env** file for OpenResty® configuration.

For most of the options the default values can be used, but for some of the options you will have to change the values as per the deployment.

Step 3 Validate all the property values given in the core.env file.

Note Do Not Rename this file.

Step 4 Provision is available in the dirs.env file if you choose to deploy the configurations on a custom NGINXinstallation. If you choose to use the Installer as is, avoid modifying the dirs.env file.

Note Do Not Rename this file.

Step 5 Retain the env files of components that you require and delete the rest.

For example, for a proxied Finesse cluster running on non-sso mode with Live Data and Unified Intelligence Center reports you will have to retain the **finesse.env, cuic.env and livedata.env** files in the directory and must delete the rest of the files (chat.env, ids.env and cloudconnect.env). The remaining env files present in the directory will be processed by the Installer.

Step 6 Rename the component env files as per their hostnames, as it is easy to identify them. Modify the component env file values as per the requirement of the deployment. Generally, you can modify only the hostname values as per the deployment and retain the defaults for the other options.

Step 7 Property description in all the env files should be self-explanatory and it should provide the information regarding the purpose and the usage of a given property. Also, **Do not modify any property name** or **delete any property from the . env file.**

Note All the properties are essential for the Installer, and incase of any missing property the Installer will not be able to open the proxy instance. Override or change the defaults only for the required properties.

Add or Remove the Unified CCE solution component

Any number of Unified CCE solution components can be proxied through the installer.

To add or remove any of the component proxies, the corresponding component environment configuration file must exist in the env directory. Installer will then generate the proxy configurations for all the required components from the start as per the contents in the env directory.

Configure Auth URL for components

The component configuration file has an option to redirect to finesse nodes auth url to perform the authentication at the proxy. This needs to be configured for the component config files as per the deployment, to redirect them to the same cluster finesse node which contains user data. For more information, see NGX_AUTH_URL= https://reverseproxy.host.domain:8445/finesse/api/UserAuth.

Multi-cluster deployment

The reverse-proxy installer supports Unified CCE or Packaged CCE that are larger than 2k deployments. These deployments must expose multiple Finesse nodes to the Agents over the internet and needs extra CUIC nodes.

These additional nodes are supported by multiple pairs of reverse-proxy or by configuring the extra nodes. The extra nodes work as added upstream servers on the same proxy pair using a single HA pair of the reverse-proxy.

Adding more upstream servers is as simple as creating a new environment (env) file. The **env** file corresponds to the upstream server type and modify specific details such as its hostname.

For example, a deployment containing three Finesse clusters must have three Finesse **env** files in the **env** directory as follows:

- Side A proxy **env** directory:
  - finesse1a.env
  - finesse2a.env
  - finesse3a.env
- Side B proxy **env** directory:
  - finesse1b.env
  - finesse2b.env
  - finesse3b.env

You can extended the same for multiple clusters of other components as required.

Consider multi-cluster deployments for the port and the hostname management. The prerequisite for the installer to communicate through proxy is that the hostname and the port pair are unique for a component across all other components.

To plan the hosts and ports used in the individual component **env** files, see the Port management section.

Starting the reverse proxy

To start the proxy instance from the Installer, we need to open the script with the required `installer.env` file from the `proxy_env_dir/` path as input args. Check the following steps below:

```
USAGE: ./proxy_launcher.sh [options...] (start|stop|reload|clear_cache)
Options: -e <ENV-DIR> -i <INSTALLER-ENV-FILE1> -i <INSTALLER-ENV-FILE2> ...
INSTALLER-ENV-FILE: Mandatory : Installer env files ... Multiple files can be provided to
override base env
ENV-DIR: Mandatory for start, Optional for other actions : Reverseproxy environment config
 data directory
Example usage: ./proxy_launcher.sh -e /path/to/env/dir -i /installer/env/1 -i
/installer/env/override/1 -i /installer/env/override/2 start
```

When the command **start** is initiated, Installer performs the following:

Step 1 Validates if the input arguments are correct, directories exist and if the mandatory files exist.

Step 2 Creates the required working directory, and volume mounts on the host as per the Installer.env file entries.

Step 3 Generates the required OpenResty® configs, it runs the command `run.sh` inside the `reverse-proxy-openresty-configs/` directory.

Step 4 Modifies the generated configs to their respective directories inside the working directory.

Step 5 Creates the self-signed SSL certificate for the reverse proxy to use it if required and configures it in the `installer.env` config.

SSL certificate will be generated only if there is no other file with the same filename in the directory or no other file is overwritten.

Note Load docker image will be provided as part of the Installer. This can be overridden from `installer.env` file, if required you can also choose to load a different image.

Step 6 It runs the container with the required arguments as per the Installer config data.

Serviceability

Bootstrap checks or validations

The Installer validates the configurations that are provided via the *.env* files and will stop the deployment if it senses certain common errors. This is done to prevent lengthy debugging on the configurations provided, which can easily be caught in the validation phase.

These are the errors which are currently sensed and reported during the validation phase.

Scenario	Sample Error Message
An unknown template type mentioned on the *.env*file which is not known to the Installer.	[ERROR]: Unknown TEMPLATE_TYPE cuic_1230 found in file cuic.env. Exiting.

Scenario	**Sample Error Message**
The *.env* file doesn't contain the property TEMPLATE_TYPE which identifies the type of upstream component	[ERROR]: TEMPLATE_TYPE variable missing in file cuic.env. Exiting.
A particular variable is not present in primary *.env* file for the template type, but is found added to a particular *.env* file being processed in the customenv dir.	[ERROR]: Below unused variable found in *./sample_envs/*. Exiting. **NGX_FIN_TEST_HOSTNAME**
NGX_LOAD_BALANCER_IPScontains values which cannot be parsed as a valid IP.	[ERROR]: **NGX_LOAD_BALANCER_IPS** should contain only IP addresses. Exiting.
NGX_LOAD_BALANCER_REAL_IP_HEADER is configured but **NGX_LOAD_BALANCER_IPS** is not configured.	[ERROR]: **NGX_LOAD_BALANCER_REAL_IP_HEADER** should be configured only when **NGX_LOAD_BALANCER_IPS** is configured. Exiting.
NGX_LOAD_BALANCER_REAL_IP_HEADER is empty but **NGX_LOAD_BALANCER_IPS** is configured.	[ERROR]: **NGX_LOAD_BALANCER_REAL_IP_HEADER** is empty. It should contain header details when **NGX_LOAD_BALANCER_IPS** is configured, Exiting.
One of the mandatory variables not configured. (Currently, limited to host and port of the upstream).	[ERROR]: **NGX_PRXY_CHAT_HOSTNAME**’s value is not configured. Exiting.
Same variable is encountered more than once in the *.env* file being processed.	[ERROR]: **NGX_PRXY_CLOUDCONNECT_HOSTNAME**’s value is configured in multiple places. Exiting.
Mandatory variable is configured more than once.	[ERROR]: **NGX_FIN_HOSTNAME**’s configured more than one time. Exiting.
Duplicate environment variable.	[ERROR]: Following variables were found to be duplicate in file `sample_env/finesse.env.` Exiting.
More than one version for Unified Intelligence Center or LiveData configured.	[ERROR]: Multiple versions of env files detected for Unified Intelligence Center, retain one type and retry. Exiting.
More than one Cisco IdS instance configured. (Each side of the proxy should only have a single instance of Ids configured).	[ERROR]: Number of Cisco IdS instance should not be more than 1. Exiting.
Env file is not readable.	[ERROR]: File sample_env/core.env does not exist or does not have appropriate permissions. Exiting.

Scenario	Sample Error Message
Master template is altered. This is just warning, it will not exit the installation.	[!!! WARNING !!!] Master templates have been altered. Note: Some of the pre-install checks that are based on the templates configurations will be skipped.
Master env is altered. This is just warning, it will not exit the installation.	[!!! WARNING !!!] Master master_env have been altered. Note: Some of the pre-install checks that are based on the templates configurations will be skipped.
Custom env dir which is passed as a run time option to the Installer is missing	[ERROR]: Directory sample_env/core does not exist. Exiting.
Certificate-based authentication is enabled for a particular upstream server (using NGX_PRXY_SSL_VERIFY="on"), without defining the certificate path.	[ERROR]: Mutual Transport Layer Security validation is enabled for finesse, but the upstream server certificate path in `NGX_PRXY_SSL_TRUST_CRT` is empty. Exiting.
Certificate-based authentication is enabled for a particular upstream server(using NGX_PRXY_SSL_VERIFY="on"). But the certificate is not present, non-readable, or empty.	[ERROR]: Mutual TLS validation is enabled for Finesse, but the upstream server certificate /root/reverse_proxy/contactcenter-reverseproxy/ssl/upstream_finesse_trust.crt is not present, not readable or invalid. Exiting.

Launcher logs

Proxy instance launcher logs can be located at *${HOST_WORKING_DIR}/logs/openresty_launcher.log*. During the NGINX startup, check the logs to see if there are any error information inside the container instance.

Openresty pid file is also located in the same folder at ***${HOST_WORKING_DIR}/logs/openresty.pid.***

Access and error logs

You can locate the Nginx access and error logs for a given proxy instance at the `logs` directory inside the proxy working directory as `${HOST_WORKING_DIR}/logs/access.log and ${HOST_WORKING_DIR}/logs/error.log`. Check these log files for any debugging information about the OpenResty® startup.

To uniquely identify the Digital Routing task requests, the reverse proxy server generates access logs with the `trackingId` field. The following is the snippet of the `access.log` with the `trackingId` field:

```
[09/Feb/2023:07:24:25 +0000] conn_stats:"7 : 1" client:"35.168.152.254"
host:"pccedrdmzproxy-cc.cisco.com" host_addr:"173.39.15.27"
host_to_upstream:"pccedrdmzproxy-cc.cisco.com->10.10.10.95:8445"
user:"-" server_block:"173.39.15.27:443" request:"POST /drapi/v1/tasks HTTP/1.1" requestid:"-"
 server_cache_bypass:"-" cookie:"-" user_agent:"Apache-HttpClient/4.5.2 (Java/1.8.0_242)"
referer:"-" cache_status:"-" rsp_status:"201(201)" body_bytes_sent:"56"
time_taken:"0.021(0.022)" up_connect_time:"0.002" up_header_time:"0.022" up_bytes_sent:"1297"
 up_bytes_rcvd:"852" trackingId:"WebexConnect_ea54eac0-1d2a-4e09-9fa2-cb212dad13df"
```

If there are failures in the Digital Routing task requests, the reverse proxy server generates error logs with the `trackingId` field only when you set the trace level to debug.

To enable the debug trace level for the reverse proxy server:

1. In the `"<reverse_proxy_installed_dir>/conf"` directory, locate and open the nginx.conf file.
2. In the nginx.conf file, find the statement, `[error_log ${NGX_LOG_DIR}/error.log info;]`.

3. Change the trace level from `info` to `debug` as follows: `[error_log ${NGX_LOG_DIR}/error.log debug;]`.
4. Reload the reverse proxy server for the change to take effect.

The following is the snippet of the `error.log` with the `trackingId` field:

```
2023/02/14 08:01:59 [debug] 206#206: *5 [lua] log_dr_requests.lua:4:  conn_stats:5:1
client:172.16.102.61 host:173.39.15.27 host_addr:173.39.15.27
host_to_upstream:173.39.15.27->10.10.10.95:8445
user:nil server_block:pccedrdmzproxy-cc.cisco.com:443 request:GET /drapi/v1/tasks?from=0
HTTP/1.1 requestid:nil server_cache_bypass:nil cookie:nil user_agent:PostmanRuntime/7.29.2
 referer:nil
cache_status:nil rsp_status:200(200) body_bytes_sent:46 time_taken:0.004(0.005)
up_connect_time:0.002 up_header_time:0.005 up_bytes_sent:3411 up_bytes_rcvd:733
trackingId:WebexConnect_ea54eac0-1d2a-4e09-9fa2-cb212dad13df
```

IP blocking logs

A separate log file is maintained to track the IPs that block the running proxy instance at `${HOST_WORKING_DIR}/logs/blocking.log`. This file can be supplied to the tools such as *fail2ban* to automate the blocking of IP addresses at IP table level.

Client IPs are blocked if a client makes several failed authentication requests in a given time interval.

Syslogs

Syslogs are released by the reverse-proxy. By default, syslogs are pushed to the local endpoint. However, proxies can be configured to push this to the remote endpoint.

Syslogs are released when the client IP is blocked by the reverse proxy.

Reloading configuration and clearing cache

Static file hosting

Reverse-proxy provides provision to host the static files as required at `${HOST_WORKING_DIR}/html`. You can add any of the static files that must be accessed through proxy such as *proxymap.txt*. These files are accessible through a static file access endpoint provided by the proxy. The endpoint hostname and the port are configurable through the *core.env* file.

By default, you can access the static files deployed on the reverse-proxy at the URL *https://[ip-of-proxy-host]:10000/staticfile*.

To configure access from a different port, use the *NGX_PRXY_STATIC_FILES_PORT* option provided in the **core.env** file.

The static file port isn’t opened by default in the IP tables. If necessary, it must be explicitly opened by the administrator. The same must be opened in the DMZ firewall to access from the internet.

Note While enabling access to this port over the internet, you must be cautious as this port isn’t covered under DOS preventive measures.

Reverse-proxy caching

Each and every proxy instance caches the files as specified by different components inside the `${HOST_WORKING_DIR}/cache` directory. Inside the cache directory, every upstream has a separate directory where the cache files for that upstream is present. The sample on how the cache is maintained is as follows:

```
${HOST_WORKING_DIR}/cache
|- client_temp
|- proxy_temp
|- finesse125.autobot.cvp
|- desktop
|- layout
|- openfire
|- rest
|- shindig
|- cuic126.autobot.cvp
|- cuic
|- cuicdoc
```

To get the latest upstream resources, the cache has to be cleared. Administrator can either do this manually by clearing all the files inside each and every directory as required or can run the script provided inside the container to clear the cache automatically.

```
docker exec <PROXY_INSTANCE_NAME>
        /usr/local/openresty/nginx/sbin/openresty_launcher.sh clear_cache
```

Caching behaviors such as cache expiration, cache sizes, and so on, can be configured from the individual component **env** files. The configuration options for different components' **env** files are as follows:

- Finesse
  - NGX_FIN_DESKTOP_CACHE_SIZE
  - NGX_FIN_DESKTOP_CACHE_MAX_SIZE
  - NGX_FIN_DESKTOP_CACHE_INACTIVE_DURATION
  - NGX_FIN_SHINDIG_CACHE_SIZE
  - NGX_FIN_SHINDIG_CACHE_MAX_SIZE
  - NGX_FIN_SHINDIG_CACHE_INACTIVE_DURATION
  - NGX_FIN_OPENFIRE_CACHE_SIZE
  - NGX_FIN_OPENFIRE_CACHE_MAX_SIZE
  - NGX_FIN_OPENFIRE_CACHE_INACTIVE_DURATION
  - NGX_FIN_REST_CACHE_SIZE
  - NGX_FIN_REST_CACHE_MAX_SIZE
  - NGX_FIN_REST_CACHE_INACTIVE_DURATION
  - NGX_FIN_LAYOUT_CACHE_SIZE
  - NGX_FIN_LAYOUT_CACHE_MAX_SIZE
  - NGX_FIN_LAYOUT_CACHE_INACTIVE_DURATION
- CUIC

- NGX_CUIC_CACHE_SIZE
- NGX_CUIC_CACHE_MAX_SIZE
- NGX_CUIC_CACHE_INACTIVE_DURATION
- NGX_CUICDOC_CACHE_SIZE
- NGX_CUICDOC_CACHE_MAX_SIZE
- NGX_CUICDOC_CACHE_INACTIVE_DURATION

Use configurations with custom NGINX installation

The proxy Installer package can be deployed as a standalone. However, you can use the following steps to deploy only the generated configuration with the third-party NGINX installations:

Step 1 Navigate to the directory **`reverse-proxy-openresty-configs/`** inside the proxy Installer.

Step 2 For third-party NGINX installations, ensure to change the **dirs.env** as per the NGINX installation directory structure.

Step 3 Generate the configurations by running the command **./run.sh <ENV-DIR>** where the **ENV-DIR** is the path of the directory containing the environment configuration data files.

Step 4 Copy the **conf, html, lua** folders from the **~/configs-out** directory to the **NGX_HOME** directory.

Note This requires NGINX installation with Lua support.

Upstream component configuration specifics

Verifying Reverse-Proxy Configuration

Finesse

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 the reverse-proxy hostnames. It shouldn’t be the 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 sub-directories) in the `finesse/3rdpartygadget` folder.

Cisco Unified Intelligence Center and LiveData

Step 1 If you find the Finesse hostnames in the response instead of reverse-proxy hostnames, validate the proxy-mapping configurations. Also, check if the allowed hosts are properly added in Finesse servers as described in the Populate Network Translation Data section.

Step 2 If the LiveData gadgets load properly in the Finesse Desktop, the CUIC and LiveData proxy configurations are correct.

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

Related Topics

Populate Network Translation Data, on page 282

Cisco Identity Service

To validate the Cisco IdS configuration, perform the following steps:

Step 1 Log in to the Cisco IdS Admin interface at **https://<ids_LAN_host:ids_port>:8553/idsadmin** from the LAN because the admin interface isn’t exposed over reverse-proxy.

Step 2 Choose **Settings** > **IdS Trust**.

Step 3 Verify that the proxy cluster publisher node is listed on the Download SP metadata page, and click **Next**.

Step 4 Verify that the IDP proxy is correctly displayed (if configured on the Upload IDP metadata page) and click **Next**.

Step 5 Initiate test SSO through all proxy cluster nodes from the Test SSO page and validate that all are successful. This requires client system connectivity to reverse-proxy nodes.

Security

Authentication

Note Authentication isn’t enabled for Digital Channel requests accepted by the proxy.

Proxy supports the authentication at the Edge. Authentication is supported for Single Sign-On (SSO) and Non-SSO deployments. Authentication is enforced for all the requests and the protocols that are accepted by the proxy before they are forwarded to the upstream component servers.

The authentication is enforced by the component servers locally. All authentication uses the common Finesse sign-in credentials to authenticate the requests. Persistent connections, such as websockets which rely on application protocols such as Extensible Messaging and Presence Protocol (XMPP) for authentication, the connections are authenticated at the proxy by validating the IP address. Connections from an IP address are allowed only if there's a successful application authentication made from the IP address, before initiating the websocket connection.

Non-SSO authentication

Non-SSO authentication doesn't require any extra configurations. It works without the NGINX configuration scripts after the required script replacements are made. Authentication relies on the username and password used to sign in to Finesse.

Access to all the endpoints are validated with Finesse authentication services. The list of valid users is cached at the proxy locally (updates the cache every 15 minutes), which is used to validate the user in a request. User credentials are validated by forwarding the request to the configured Finesse URI and thereafter the credential hash is cached locally (cached 15 minutes) to authenticate new requests locally. If there's any change to the username or password, it takes effect only after 15 minutes.

SSO authentication

SSO authentication for Cisco IdS 12.6(1) (latest ES) requires that the administrator configure the Cisco IdS token encryption key at the NGINX server within the configuration file. You can obtain the Cisco IdS token encryption key from the Cisco IdS server with the **show ids secret** CLI command. They key has to be configured as part of the **core.env (NGX_JWT_SECRET option)** file that the administrator has to perform in the scripts before the SSO authentication can work.

For Cisco IdS in 12.6.(2) and above this need not be configured, as the proxy automatically add this information from the backend. For more information on Single Sign-On, see Cisco Unified Contact Center Enterprise Features Guide.

The SSO user guide for the Cisco IdS SAML configurations to be performed for the proxy resolution to work for Cisco IdS. After SSO authentication is configured, a valid pair of tokens can be used to access any of the endpoints in the system. The proxy configuration validates the credentials by intercepting the token retrieval requests made to Cisco IdS or by decrypting valid tokens and thereafter caching them locally for further validations.

Authentication for Websocket connections

Websocket connections can't be authenticated with the standard authorization header, as custom headers aren't supported by original websocket implementations in the browser. Application-level authentication protocols, where the authentication information contained in the payload doesn't prevent websocket connection establishment. So, malicious entities can render DOS or DDOS attacks just by creating innumerable connections to overwhelm the system.

To mitigate this possibility, the NGINX reverse-proxy configurations provided have specific checks to allow the websocket connections to be accepted ONLY from those IP addresses which have successfully made an authenticated REST request before establishing the websocket connection. It implies that the clients which attempt to create websocket connections before a REST request is issued, gets an authorization failed error and isn't the supported usage scenario.

Validating unauthenticated static resources

All valid endpoints that can be accessed without any authentication are actively tracked in the ES04 scripts . If invalid URIs are requested to these unauthenticated paths, they are rejected without sending the requests to the components' servers.

Brute Force attack prevention

The proxy authentication scripts actively prevent brute force attacks which can be used to guess the user password. It does this by blocking the IP address which is used to access the service. After some failed attempts in a short time, these requests are rejected with the HTTP error 418. You can access the details of the blocked IP addresses from the **${HOST_WORKING_DIR}/logs/blocking.log** and **${HOST_WORKING_DIR}/logs/error.log** files.

You can configure the threshold for failed requests, the time interval for the threshold, and the blocking duration. The configurations are present in the **core.env** file. The following are the options:

- **NGX_CLIENT_LOCK_THRESHOLD**: Request authorization failure threshold for a source IP
- **NGX_CLIENT_LOCK_DURAION**: Request authorization failure threshold over a given interval for a source IP
- **NGX_CLIENT_BLOCK_DURAION**: Sets the duration (in seconds) of blocking a client to avoid brute force attack

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

You can find the IP addresses that are blocked.

To find the IP addresses that are blocked, run the following commands from the directory ***{HOST_WORKING_DIR}/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"
```

Note It’s recommended that the customers integrate with Fail2ban or similar to add the ban to the IP table or firewall rules.

Caching CORS headers

When the first option request is successful, then the following response headers are cached at the proxy for five minutes. These headers are cached for each respective upstream server.

- access-control allow-headers
- access-control-allow-origin
- access-control-allow-methods
- access-control-expose-headers and
- access-control-allow-credentials

Install and configure Fail2ban

Fail2ban scans log files and bans IPs that show the malicious signs such as too many password failures, seeking for exploits, and so on. Generally, Fail2Ban is used to update the firewall rules to reject the IP addresses for a specified amount of time. It can also be configured for any arbitrary actions such as sending an email. For more information, see https://www.fail2ban.org/.

Fail2ban can be configured to monitor the blocking log to identify the IP addresses that are blocked by NGINX on detecting brute force attacks, and ban them for a configurable duration.

The following are the steps to install and configure Fail2ban on a CentOS reverse-proxy:

Step 1 Install the 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.

Use the following steps to add a custom configuration for banning the IP addresses that are blocked from accessing the upstream servers:

a) Navigate to the Fail2ban installation directory (in this example `/etc/fail2ban`) `cd /etc/fail2ban`.
b) Create a copy of **jail.conf** into **jail.local** to keep the local changes isolated in **cp jail.conf jail.local**.
c) Add the following jail configurations to the end of the **jail.local** file. Substitute the ports in the template with the actual ones. Update the 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 = ${HOST_WORKING_DIR}/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 are 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 **fail2ban-client start** command to start Fail2ban.

Open the Fail2ban log files and verify that there are no errors. By default, logs for Fail2ban go into the `/var/log/fail2ban.log` file.

Step 5 Validate static resource URLs. All valid endpoints which can be accessed without authentication are actively tracked in the proxy scripts.

Requests to these unauthenticated paths are actively rejected, if an invalid URI is requested, without sending these requests to the upstream server.

Frequently Asked Questions

Why does the proxy launcher fail to restart the Reverse Proxy?

The environment settings are incorrect. Correct any errors in the environment data and retry.The log file is stored at `${HOST_WORKING_DIR}/logs/openresty_launcher.log`. Using the command **docker ps -a**, see if the container is up and running.

How can I solve the OpenResty® launch error?

Some error during OpenResty® start. Fix any of the errors listed in the error log file available at `${HOST_WORKING_DIR}/logs/error.log` and try to restart.

Why is the content not refreshed to the end user?

Cache is not updated with latest contents. Run the following command to clear the cache:

docker exec <PROXY_HOSTNAME> `/usr/local/openresty/nginx/sbin/openresty_launcher.sh clear_cache`. The error log file available at `${HOST_WORKING_DIR}/logs/access.log`

Why is configuration generation from templates unsuccessful?

Failed to validate while generating the configuration. Correct any problems or failures reported on the console or in the error file. The error file as follows "Configuration generation from templates fails".

How can I fix problems or failures reported on the console or in the error file?

Reverse proxy is not included in the authorized list. Use this list of CLI Reverse Proxy authorized hosts and confirm if the list of Reverse Proxy authorized host names configured on Cisco IdS and Finesse boxes. This must contain the Reverse Proxy hostname and the allowed IP address.

What causes intermittent failures of Finesse REST API?

Because of the NGINX proxy rate limit issue, gadgets are not loading in the Finesse desktop. This results in intermittent Finesse REST API failures.

How do I determine which OpenResty® version is being used in the Installer?

Run the following command in the proxy instance to check the OpenResty® version on the Installer:

docker inspect <proxy_instance_name> | grep resty_rpm_version | cut -d ":" -f2

Why does proxy send HTTP error code 4xx ?

Refer to the HTTP HTTP Return codes returned by the reverse-proxy, on page 291 section.

Environment Files

The reverse-proxy installer behavior is driven using user-editable configuration files called environment files (.env). The environment file contains configuration data in the form of **key=value** pairs, which are referred to as properties. Each upstream component has custom environment files and properties specific to the respective component. The installer also has its own specific environment files, used to customize its behavior. Reverse-proxy installation requires the administrator to modify the properties to match the deployment. The following tables list and describe these properties, with their default values and guidance about changing them:

Note Reverse Proxy Automated Installer, on page 325 is a per-requisite reading for this chapter.

Installer env properties

The installer runs the container (which is in a docker), that contains the proxy. The properties determine the configuration of the container like the resources made available to it and the network configurations and such. By default, the properties are set to 2000 users deployment. Deployments which are bigger or smaller than 2000 users must verify these values and modify them appropriately.

Property Name, Description, and Default	Change Recommended?	When to Change?
CONTAINER_NAME Specifies the name of the reverse-proxy container—generally the reverse-proxy hostname. **Default:** proxy25.autobot.cvp	Yes	When you change the name of the container.
CONTAINER_NETWORK_MODE Specifies the network mode of the container. **Default:** host	Yes, If required.	If you use the host network mode for a container, the network stack for that container isn't isolated from the Docker host. [2] The other value is **bridge**. A bridge network creates a separate network for containers to communicate with each other, even if it is isolated from other networks on the host. This is useful when you want to deploy multiple containers on a single host and communicate with each other, but not with the outside world.
CONTAINER_DNS_RESOLVER Specifies a list of DNS servers separated by the | symbol. **Default:** 1.1.1.1|8.8.8.8	Yes	If an IP address changes, update the list.
CONTAINER_DNS_SEARCH_DOMAIN Specifies a DNS search domain to use when resolving hostnames inside the container. This property takes one or more domain names as arguments, separated by commas. In this example, the DNS search domain is `example.com`. Inside the container, the DNS resolver appends the search domain to the hostname and attempts to resolve it. If you ping the webserver inside the container, the DNS resolver tries to resolve `webserver.example.com`; if that fails, it tries to resolve `webserver`. **Default:** search.domain.1|search.domain.2	Yes	—
CREATE_SELF_SIGNED_SSL_CERT Specifies whether to create a self-singed certificate during the reverse-proxy installation. **Default:** TRUE	Yes, If required.	If the CA-signed certificates are present, you don't need to install self-signed certificates during the installation. In this case, change to FALSE.

Property Name, Description, and Default	Change Recommended?	When to Change?
CERTIFICATE_COMMON_NAME Specifies the common name for the certificate. This value is required to create self-signed certificates. Used on the next property. **Default:** `*.cisco.com`	Yes, If required.	Required only for creating self-singed certificates.
CERTIFICATE_SUBJECT Specifies the subject line to be used on the self-signed certificate. **Default:** /C=IN/ST=KA/L=BLR/O=Cisco/OU=CCBU/CN=${CERTIFICATE_COMMON_NAME}	Yes, If required.	Required only for creating self-singed certificates.
SSL_CERT_NAME Specifies the name of the certificate file to be auto-generated. **Default:** reverseproxy.crt	Yes, If required.	Required only for creating self-singed certificates.
SSL_KEY_NAME Specifies the name of the key file to be auto-generated. **Default:** reverseproxy.key	Yes, If required.	Required only for creating self-singed certificates.
SSL_CERT_KEY_LENGTH Specifies the certificate key length to create the self-signed certificate. **Default:** 2048	Yes, If required.	Required only for creating self-singed certificates.
SSL_CERT_EXPIRY_IN_DAYS Certificate expiry in days, to be specified in the self-signed certificate. **Default:** 1095	Yes, If required.	Required only for creating self-singed certificates.
AUTO_RESTART_CONTAINER Toggles auto-restart of the reverse-proxy container when the host system reboots. **Default:** 0	Yes	Enable this property only when the reverse-proxy is in working condition. [3]
NOFILE_LIMIT Specifies the initial and maximum number of open file descriptors that a container can have. Option in Docker is used to set system resource limits on a container. **Default:** nofile=102400:102400	Yes, If required.	nofile=204800:204800 for a 4000 deployment.
CPU_LIMIT Specifies the number of CPUs that a container can use. **Default:** 2	Yes, If required.	4 for 4000 deployment

Property Name, Description, and Default	Change Recommended?	When to Change?
MEM_LIMIT Specifies the maximum amount of memory that a container can use, in bytes or using a human-readable format. **Default:** 4G	Yes, If required.	8G for 4000 deployment
MEM_SWAP_LIMIT Specifies the maximum amount of memory and swap for a container—in bytes or using a human-readable format such as 1G for 1 gigabyte. **Default:** 8G	Yes, If required.	
MEM_RES Sets a soft limit on the minimum amount of memory to be available for the container. **Default:** 2G	Yes, If required.	4G for 4000 deployment

[2] The container shares the host networking namespace, and the container doesn’t allocate its own IP address. For example, if you run a container which binds to port 80 and you use host networking, the container application is available on port 80 on the host IP address.

[3] If enabled and the container stops because of miss-configuration, setting the value with 1 keeps trying to restart the container. Also, the reverse-proxy container keeps running, until it is explicitly stopped.

Note Ensure that the host has adequate resources to run the container with the modified resource constraints.

Installer env properties that aren’t recommended to be altered

Note These properties are provided for reference and they are available in the configuration to provide flexibility, to adjust the behavior if necessary, and in exceptional situations. It isn't recommended to change casually without extensive testing.

Property Name, Description, and Default	Change Recommended?	When to Change?
CONTAINER_IMAGE A Docker image is a read-only template that contains a set of instructions for creating a container that can run on the Docker platform. **Default:** reverse-proxy-openresty-container:12.6(2)	No	Never
HOST_WORKING_DIR Specifies the working directory of the container **Default:** ~/reverse_proxy/${CONTAINER_NAME}	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_HOME Specifies the home directory of the NGINX server inside the container. **Default:** /usr/local/openresty/NGINX	No	
HOST_CACHE_VOL Specifies the host system directory used to mount on the container.[4] Mapped with the following container directory: NGX_CACHE_DIR. **Default:** ${HOST_WORKING_DIR}/cache	No	
HOST_SSL_VOL Specifies the host system directory used to mount on the container. Mapped to the following container directory: NGX_SSL_DIR **Default:** ${HOST_WORKING_DIR}/ssl	No	
HOST_LOGS_VOL Specifies the host system directory used to mount on the container. Mapped to the following container directory: NGX_LOG_DIR **Default:** ${HOST_WORKING_DIR}/logs	No	
HOST_CONF_VOL Specifies the host system directory used to mount on the container. Mapped with the container directory mentioned here: NGX_CONF_DIR **Default:** ${HOST_WORKING_DIR}/conf	No	
HOST_HTML_VOL Specifies the host system directory used to mount on the container. Mapped to the following container directory: NGX_HTML_DIR **Default:** ${HOST_WORKING_DIR}/html	No	
HOST_LUA_VOL Specifies the host system directory used to mount on the container. Mapped to the following container directory: NGX_LUA_DIR **Default:** ${HOST_WORKING_DIR}/lua	No	
NGX_CACHE_DIR Specifies the container directory location mapped with the corresponding host system directory specified in the HOST_CACHE_VOL property. **Default:** ${NGX_HOME}/cache	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_SSL_DIR Specifies the container directory location mapped with the corresponding host system directory mentioned in the HOST_LOGS_VOL property. **Default:** ${NGX_HOME}/ssl	No	
NGX_LOG_DIR Specifies the container directory location mapped with the corresponding host system directory mentioned in the HOST_LOGS_VOL property. **Default:** ${NGX_HOME}/logs	No	
NGX_CONF_DIR Specifies the container directory location mapped with the corresponding host system's directory mentioned in the HOST_CONF_VOL property. **Default:** ${NGX_HOME}/conf	No	
NGX_HTML_DIR Specifies the container directory location mapped with the corresponding host system directory mentioned in the HOST_HTML_VOL property. **Default:** ${NGX_HOME}/html	No	
NGX_LUA_DIR Specifies the container's directory location mapped with the corresponding host system's directory mentioned on this property HOST_LUA_VOL. **Default:** ${NGX_HOME}/lua	No	
MEM_SWAPPINESS Controls how aggressively the kernel should swap memory pages of the container to disk when the container exceeds its memory limit. **Default:** 1	No	
LOAD_CONTAINER_IMAGE_FROM_TAR This property is commented out by default. The default value (when it's commented) is true. **Default:** This property is commented by default.	No	You can change the value to load the container image from a different location.

Property Name, Description, and Default	Change Recommended?	When to Change?
REVERSE_PROXY_CONTAINER_IMAGE_TAR Specifies the location of the container image tar file. This property is commented out by default. ${SCRIPTPATH} is the location of the proxy_launcher.sh script. **Default:** ${SCRIPTPATH}//reverse-proxy-openresty-container/reverse-proxy-openresty-container.tar.gz	No	
RESTART_COND	No	Not used.

[4] Volume mounting is a concept used in computer systems to make a directory or file from one file system available to another file system. It's a method for sharing data between containers in a Docker environment or between a container and the host system.

Core properties

These are the basic properties that determine the behavior of the included OpenResty® Nginx proxy and control various aspects of its runtime behavior. It also contains request rates and various cache sizes setting for Nginx.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_DNS_RSLVR Specify the name servers used to resolve the names of upstream servers into addresses. Use spaces to separate multiple DNS server IP addresses. **Default:** 192.168.1.3 192.168.1.4 192.168.1.5	Yes	Update the entries with the IP addresses of the DNS servers.
NGX_JWT_SECRET OpenResty® Constants(defined in maps.conf) configuration. JWT secret pulled from IdS host using CLI "show ids secret" This secret is used to verify and validate tokens at proxy for authentication in SSO mode This secret is applicable only for IdS < 12.6(2). **Default:** TWSFbB9J6fBnu/D/hrHiQl2O0WEgrVj69ZiHJCtwahI=	Yes, if IdS is running in < 12.6(2) version	Update it with the output of this command from IdS: "show ids secret"
NGX_SYSLOG_SVR_IP Specifies the syslog server IP to which NGINX pushes some specific notification logs when the access for an IP is blocked. **Default:** 127.0.0.1	Yes, if necessary.	The current syslog server is the current reverse-proxy. This can be changed to the IP for any syslog server, based on the configuration.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_VALID_REFERRERS Specifies the “Referrer” request header field values for which the request is allowed. Request is blocked for all other referrers. The value is case-sensitive. Include all reverse-proxy hostnames, IdS hostnames and ADFS hostname in this list. They are required for reverse-proxy and other functionality. **Default:** proxy_pub.host.domain|proxy_sub.host.domain| ids_pub.host.domain|ids_sub.host.domain|adfs.host.domain	Yes	If not updated, the pages return with 417 HTTP error code. Make sure there are no typos in the hostnames.
NGX_LOCALHOST_IPS Specifies the list of IPs assigned to the reverse-proxy host across all NICs. Include all public and private IPs for reverse-proxy in this list. Include the alternate side reverse-proxy's IP addresses as well. **Default:** 192.168.1.69|192.168.1.169	Yes	Update all the reverse-proxy IPs here.
NGX_RATELIMIT_DISABLE_IPS Specifies a list of IP addresses for which rate limits aren't applied. **Default:** 192.168.1.69|192.168.1.169|127.0.0.1	Yes	All the IP address that should be allowed to exclude on rate-limiting. Update the list with all the public and private IPs of both the primary and secondary reverse-proxy. It can also include any other load balancer or proxy which are forwarding requests to reverse-proxy.
NGX_LOAD_BALANCER_IPS Hostnames aren’t supported as a permissible value in NGX_LOAD_BALANCER_IPS The list of entries should be | separated # Example: "192.162.1.0/24|10.78.95.76" Alternatively, if the internet client connection is stopped at the reverse-proxy directly, these variables MUST be empty.	Yes, If required.	If the load balancer forwards requests to the reverse-proxy, populate with the load balancer IP addresses.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_LOAD_BALANCER_REAL_IP_HEADER Devices must also send the end client IP alone, in a custom header. Add the name of the custom header used for this purpose to the NGX_LOAD_BALANCER_REAL_IP_HEADER variable. For example, "**X-Real-IP**". If you use the **X-Forwarded-For** as the field used to detect the client IP, include all trusted devices that can appear in this list in the NGX_LOAD_BALANCER_IPS variable. The first untrusted IP encountered is used as the client IP. We don't recommend using this field (**X-Forwarded-For**) for detecting the client IP.	Yes, If required.	

Core properties that are not recommended to be altered

Note These properties are provided for reference and they are available in the configuration, to provide flexibility and adjust the behavior if necessary, in exceptional situations, and aren't recommended to be changed casually without extensive testing.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_NUM_WKR_PRC OpenResty® NGINX core configurations. Specifies the number of worker processes. The value "auto" uses the number of available CPU cores. **Default:** auto	No	
NGX_PID_FILE Defines a file that stores the process ID of the main process. **Default:** openresty.pid	No	
NGX_WKR_CPU_AFFINITY Binds the worker processes to the sets of CPUs. The value "auto" binds worker processes automatically to the available CPUs. **Default:** auto	No	
NGX_WKR_PRIORITY Defines the scheduling priority for worker processes like it's done by the nice command. A negative number means higher priority. The allowed range varies from -20 to 20. **Default:** 0	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_NUM_RLIMIT Changes the limit on the maximum number of open files (RLIMIT_NOFILE) for worker processes. Used to increase the limit without restarting the main process. **Default:** 102400	No	
NGX_MULTI_ACCEPT If multi_accept is disabled, a worker process accepts one new connection at a time. Otherwise, a worker process accepts all new connections at a time. **Default:** on	No	
NGX_NUM_WKR_CONN Specifies the maximum number of simultaneous connections that can be opened by a worker process. **Default:** 10240	No	
NGX_SEND_FILE Enables or disables the use of **sendfile**.	No	No
NGX_TCP_NOPUSH Enables or disables the use of the TCP_NOPUSH socket option on FreeBSD or the TCP_CORK socket option on Linux. The options are enabled only when the **sendfile** is used. **Default:** on	No	
NGX_MAP_HASH_BUCKET_SIZE Specifies the bucket size for the map variables hash tables. **Default:** 128	No	
NGX_SERVERNAMES_HASH_BUCKET_SIZE Specifies the bucket size for the server names hash tables. **Default:** 512	No	
NGX_JWT_EXPIRY Specifies the JWT token expiry in seconds as configured in the IdS host. Token cache expiry time in reverse-proxy. Reverse-proxy keeps the cached token for 2 hours for the default configuration of 1-hour access token expiry time configured in IdS. **Default:** 7200	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_IDS_PUBLIC_KEY_POLL_INTERVAL Specifies the IdS public key poll frequency in seconds. The frequency at which reverse-proxy polls the ids to get the public key value. The default is once in 5 minutes. **Default:** 300	No	
NGX_CLIENT_LOCK_THRESHOLD If the threshold to detect DoS attacks is crossed in the specified interval, the client IP is blocked for the specified duration. **Default:** 5	No	
NGX_CLIENT_LOCK_DURATION Specifies the request authorization failure threshold over a given interval for a source IP. **Default:** 30	No	
NGX_CLIENT_BLOCK_DURATION Specifies the duration of blocking (in seconds) for clients to avoid brute force attacks. The block duration for the client IP is 30 minutes. **Default:** 1800	No	
NGX_SYSLOG_SVR_PORT Specifies the port for the syslog server. **Default:** 514	No	Usually the syslog server listens on 514, if the syslog server is configured to listen on some other port then this can be changed.
NGX_LOG_FILE Specifies the OpenResty® logging file. **Default:** access.log	No	
NGX_LOG_FORMAT Specifies the OpenResty® NGINX access log format name as specified in logging.conf. **Default:** info	No	Not recommended to change on a production system. You can change it to the debug format in LAB setup for more detailed logging.
NGX_LOG_BUFFER Specifies the OpenResty® NGINX access log buffer size. When this buffer is full or the flush interval is reached, the system writes the logs to the disk. **Default:** 16k	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_LOG_FLUSH_INTERVAL Specifies the OpenResty® NGINX access log flush interval. Logs are written to the disk after this interval is reached or the log buffer is full. **Default:** 30s	No	Not recommended changing on production servers. For a LAB system, you can reduce this value to 1 to 5s so you can check the access.log file updates immediately.
NGX_PROXY_CACHE_LOCK Only one request at a time can populate a new cache element identified according to the proxy_cache_key directive by passing a request to the server, which is enabled with reverse-proxy. Other requests of the same cache element either wait for a response to appear in the cache or the cache lock for this element to be released, up to the time set by the NGX_PROXY_CACHE_LOCK_TIMEOUT value. **Default:** on	No	
NGX_PROXY_CACHE_LOCK_TIMEOUT Specifies the timeout for NGX_PROXY_CACHE_LOCK. When the time expires, the request is passed to the server, which is enabled with reverse-proxy; however, the response isn't cached. **Default:** 30s	No	
NGX_PROXY_CACHE_LOCK_AGE If the last request passed to the server, which is enabled with reverse-proxy, for populating a new cache element hasn't completed for the specified time, one more request passes to the server, which is enabled with reverse-proxy. **Default:** 5s	No	
NGX_PROXY_CACHE_BACKGROUND_UPDATE Allows starting a background sub request to update an expired cache item, while a stale cached response is returned to the client. **Default:** on	No	
NGX_PROXY_CACHE_REVALIDATE Enables revalidation of expired cache items using conditional requests with the "If-Modified-Since" and "If-None-Match" header fields. **Default:** on	No	
NGX_PROXY_CACHE_VALID Specifies the caching time for 200, 301, and 302 responses. **Default:** 24h	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_VARIABLES_HASH_BUCKET_SIZE Specifies the bucket size for the variables hash table. **Default:** 128	No	
NGX_KEEPALIVE_TIMEOUT Specifies a timeout during which a keep-alive client connection stays open on the server side. The zero value disables keep-alive client connections. **Default:** 20s	No	
NGX_SEND_TIMEOUT Specifies a timeout for transmitting a response to the client. The timeout is set only between two successive write operations, not for the transmission of the whole response. **Default:** 10s	No	
NGX_CLIENT_HEADER_TIMEOUT Specifies the timeout for reading the client request header. **Default:** 10s	No	
NGX_CLIENT_BODY_TIMEOUT Specifies a timeout for the reading the client request body. The timeout is set only for a period between two successive read operations, not for the transmission of the whole request body. **Default:** 10s	No	
NGX_RESET_TIMEDOUT_CONNECTION Enables or disables resetting timed out connections and connections closed with the non-standard code 444. **Default:** on	No	
NGX_CLIENT_HEADER_BUFFER_SIZE Specifies the buffer size for reading the client request header. **Default:** 4K	No	
NGX_CLIENT_BODY_BUFFER_SIZE Specifies the buffer size for reading the client request body. **Default:** 2k	No	
NGX_CLIENT_MAX_BODY_SIZE Specifies the maximum allowed size of the client request body. **Default:** 15m	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_LARGE_CLIENT_HEADER_BUFFER_NUM Specifies the maximum number of buffers used for reading a large client request header. Buffers are allocated only on demand. **Default:** 2	No	
NGX_LARGE_CLIENT_HEADER_BUFFER_SIZE Specifies the maximum size of buffers used for reading a large client request header. A request line can't exceed the size of one buffer. Buffers are allocated only on demand. **Default:** 8K	No	
NGX_UNDERSCORES_IN_HEADERS Enables or disables the use of underscores in client request header fields. **Default:** on	No	
NGX_KEEPALIVE_REQUESTS Specifies the maximum number of requests that are served through one keep-alive connection. After the maximum number of requests are made, the connection is closed. **Default:** 500	No	
NGX_HTTP2_MAX_CONCURRENT_STREAMS Specifies the maximum number of concurrent HTTP/2 streams in a connection. **Default:** 150	No	
NGX_SERVER_TOKENS Enables or disables emitting NGINX version on error pages and in the "Server" response header field. **Default:** off	No	
NGX_LIMIT_CONN_DRY_RUN Enables the dry-run mode for limiting HTTP connections. In this mode, the number of connections isn't limited. However, in the shared memory zone, the number of excessive connections is considered as usual. **Default:** off	No	On a production system, this should be always "off". If the system is running in lab mode, you can toggle this "on" to avoid rate limiting.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_LIMIT_REQ_DRY_RUN Enables the dry-run mode for limiting HTTP requests. In this mode, the number of connections isn’t limited, however, in the shared memory zone, the number of excessive connections is considered as usual. **Default:** off	No	On a production setup, this should be always "off". If the system is running in lab mode, you can toggle this "on" to avoid rate limiting.
NGX_LIMIT_CONN_LOG_LEVEL Specifies the desired logging level for cases when the server limits the number of connections. **Default:** error	No	
NGX_LIMIT_REQ_LOG_LEVEL Specifies the desired logging level for cases when the server refuses to process requests due to rate exceeding, or delays request processing. **Default:** error	No	
NGX_LIMIT_REQ_STATUS Specifies the status code to return in response to rejected requests due to HTTP request rate limits. This is the standard HTTP error code for rate-limiting errors. **Default:** 429	No	
NGX_LIMIT_CONN_STATUS Specifies the status code to return in response to rejected requests due to HTTP connection rate limits. **Default:** 503	No	Error code returned when the connection limits are reached.
NGX_CHAT_REQUEST_RATE_LIMIT Specifies the HTTP request rate limit for chat access. **Default:** 30r/s	No	
NGX_IDS_REQUEST_RATE_LIMIT Specifies the HTTP request rate limit for IdS access. **Default:** 5r/s	No	
NGX_FIN_REQUEST_RATE_LIMIT Specifies the HTTP request rate limit for Finesse access. **Default:** 45r/s	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_FIN_CLIENT_LOG_REQUEST_RATE_LIMIT Specifies the HTTP request rate limit for Finesse client log requests. **Default:** 5r/s	No	
NGX_FIN_SSO_VALVE_REQUEST_RATE_LIMIT Specifies the HTTP request rate limit for Finesse SSO valve requests. **Default:** 5r/s	No	
NGX_CUIC_REQUEST_RATE_LIMIT Specifies the HTTP request rate limit for CUIC access. **Default:** 50r/s	No	
NGX_CUIC_HISTORICAL_REPORT_REQUEST_RATE_LIMIT Specifies the HTTP request rate limit for CUIC historical report requests. **Default:** 16r/s	No	
NGX_CUIC_REALTIME_REPORT_REQUEST_RATE_LIMIT Specifies the HTTP request rate limit for CUIC realtime report requests. **Default:** 48r/s	No	
NGX_CUIC_REPORT_EXECUTION_REQUEST_RATE_LIMIT Specifies the HTTP request rate limit for CUIC report execution requests. **Default:** 12r/s	No	
NGX_LIVEDATA_REQUEST_RATE_LIMIT Specifies the HTTP request rate limit for livedata access. **Default:** 25r/s	No	
NGX_CLOUDCONNECT_DR_TASK_REQUEST_RATE_LIMIT Specifies the HTTP request rate limit for DR API task request access. **Default:** 100r/s	No	
NGX_CLOUDCONNECT_USER_SYNC_CALLBACK_REQUEST_RATE_LIMIT Specifies the HTTP request rate limit for user sync callback request access. **Default:** 5r/m	No	
NGX_PRXY_STATIC_FILES_PORT Specifies the OpenResty® static content configuration. The reverse-proxy port is used to serve static files under the HTML directory. **Default:** 10000	No	This location serves the proxy-map information. You can change the port number if necessary.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_PRXY_STATUS_IP Specifies the reverse-proxy IP used to access OpenResty® NGINX stats over the "/reverseproxy_status" endpoint Internal request is accessible from only the host system. **Default:** 127.0.0.1	No	
NGX_PRXY_STATUS_PORT Specifies the reverse-proxy port used to access OpenResty® NGINX stats over the "/reverseproxy_status" endpoint. **Default:** 10001	No	
NGX_USERTIMERTHREAD_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 100k	No	
NGX_USERLIST_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 50m	No	
NGX_CREDENTIALSSTORE_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 100m	No	
NGX_USERCOUNT_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 100k	No	
NGX_CLIENTSTORAGE_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 100m	No	
NGX_BLOCKINGRESOURCES_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 100m	No	
NGX_TOKENCACHE_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 10m	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_IPSTORE_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 10m	No	
NGX_DESKTOPURLLIST_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 10m	No	
NGX_DESKTOPURLCOUNT_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 100k	No	
NGX_THIRDPARTYGADGETURLLIST_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 100m	No	
NGX_THIRDPARTYGADGETURLCOUNT_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 100k	No	
NGX_CORSHEADERSSTORE_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 100k	No	
NGX_TIMERTHREADSSTORE_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 100k	No	
NGX_ALTERNATE_HOSTS_SHRD_DICT_SIZE Specifies the LUA shared dictionary sizes used by reverse-proxy internally. **Default:** 100k	No	

Directory (DIR) properties

The following table lists the directory properties and the default values for various OpenResty® folders.

Note These properties are provided for reference and they are available in the configuration. They provide flexibility to adjust the behavior if necessary, in exceptional situations, and aren't recommended changing casually without extensive testing.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_CACHE_DIR Specifies the cache directory where various resources for components are cached. **Default:** ${NGX_HOME}/cache	No	
NGX_CONF_DIR Specifies the OpenResty® directory containing NGINX configurations, such as core and component configurations. **Default:** ${NGX_HOME}/conf	No	
NGX_HOME Specifies the home directory for OpenResty® nginx installation. **Default:** /usr/local/openresty/nginx	No	
NGX_HTML_DIR Specifies the OpenResty® directory **Default:** ${NGX_HOME}/html Directory containing static resources.	No	
NGX_LOG_DIR Specifies the OpenResty® directory where OpenResty® logs are stored. **Default:** ${NGX_HOME}/logs	No	
NGX_LUA_DIR Specifies the OpenResty® directory containing lua resources. **Default:** ${NGX_HOME}/lua	No	
NGX_SSL_DIR Specifies the OpenResty® directory containing SSL resources like certs and keys. **Default:** ${NGX_HOME}/ssl	No	

Common Properties

Common SSL-Related Properties

The following table lists SSL-related properties that are common across components.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_PRXY_SSL_TRUST_CRT Specifies a file with trusted CA certificates in the PEM format used to verify the certificate of the Finesse HTTPS server, which is enabled for reverse-proxy. **Default:** ${NGX_SSL_DIR}/upstreams_finesse_trust.crt	Yes, if necessary.	Point to the exact location of upstream Finesse's certificate for mutual trust establishment.
NGX_SSL_CRT SSL connector configuration for the component access. Specifies the location of a file with the certificate, for the given component, in the PEM format. **Default:** ${NGX_SSL_DIR}/reverseproxy.crt	Yes, if necessary.	Update the location of the Finesse reverse-proxy certificate.
NGX_SSL_KEY Specifies the location of a file with the secret key, for the given component, in the PEM format. **Default:** ${NGX_SSL_DIR}/reverseproxy.key	Yes, if necessary.	If the location of the file changes.

Common SSL-Related properties that are not recommended to be altered

Note These properties are provided for reference and they are available in the configuration. They provide flexibility to adjust the behavior if necessary, in exceptional situations, and aren't recommended changing casually without extensive testing.

Property Name, Description, and Default	Change Recommended?	When to Change?
#NGX_SSL_DHPARAM Specifies a file with DH parameters for DHE ciphers. **Default:** ${NGX_SSL_DIR}/dhparam.pem This property is disabled by default. Uncomment the parameter to use it.	No. Can be changed if necessary.	Enable for more security, to prevent affecting by an attack exploiting the Logjam vulnerability. For more information, see *Understanding Logjam and Future-Proofing Your Infrastructure* at: https://cisco.blogs.com
NGX_PRXY_SSL_VERIFY Enables or disables verification of the HTTPS server certificate, which is enabled for reverse-proxy. **Default:** on	No	Changing this to off disables SSL verification of the requests.
NGX_PRXY_SSL_VERIFY_DEPTH Specifies the verification depth in the HTTPS server certificates chain, which is enabled for reverse-proxy. This is for DoS prevention. Building the chain might be an exponential algorithm with backoff, so with the openssl default of 100 a malicious backend might cause denial of service in nginx. **Default:** 10; less than 4 is easy to break.	No	
NGX_SSL_CACHE_SIZE Specifies the SSL session cache size for session parameters storage for client connections. This cache is shared between all worker processes. The cache size is specified in bytes; one megabyte can store about 4000 sessions. Each shared cache should have an arbitrary name. **Default:** 10m	No	
NGX_SSL_CIPHERS Specifies the OpenSSL format for enabled ciphers. **Default:** EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH	No	This list contains all the strong ciphers available. You can update the list if ciphers are found to be vulnerable or to add new supported ciphers.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_SSL_PRFR_SRVR_CIPHERS Prefer server ciphers over client ciphers. **Default:** on	No	
NGX_SSL_PROTO Specifies the TLS versions enabled for the connections. To specify multiple values, use space delimiters. # Example: NGX_SSL_PROTO="TLSv1 TLSv1.1 TLSv1.2" Generally TLS version 1.2 is supported for the webapp requests. TLS v1 and TLS v1.1 aren't recommended. **Default:** TLSv1.2	No	
NGX_SSL_SESSION_TICKETS Enables or disables session resumption through TLS session tickets. **Default:** off	No	Enable to resume sessions and avoid keeping a per-client session state. The TLS server encapsulates the session state into a ticket and forwards it to the client. The client can later resume a session using the obtained ticket.
NGX_SSL_SSN_TIMEOUT Specifies a time during which a client may reuse the session parameters—how long each session lives in reverse-proxy. **Default:** 30m	No	
NGX_SSL_STAPLING Enables or disables stapling of OCSP responses by the server. SSL stapling means that revocation information about the servers certificate (that is, the OCSP response) are included in the TLS handshake together with the server certificate. **Default:** off	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_SSL_STAPLING_VERIFY Enables or disables the verification of OCSP responses by the server. Allows the presenter of a certificate to bear the resource cost involved in providing OCSP responses by appending ("stapling") a time-stamped OCSP response signed by the CA to the initial TLS handshake, eliminating the need for clients to contact the CA". **Default:** off	No	

Cisco Finesse Properties

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_PRXY_FIN_HOSTNAME Reverse-proxy hostname through which this Finesse host is accessed. **Default:** reverseproxy.host.domain	Yes	New installation or if the hostname of the reverse-proxy must be accessed from the internet changes.
NGX_PRXY_FIN_PORT Specifies the reverse-proxy port over which this Finesse host are accessed. **Default:** 8445	Yes, if necessary.	New installation or if the port number of the reverse-proxy must be accessed from the internet changes.
NGX_FIN_HOSTNAME Specifies the upstream Finesse hostname. **Default:** finesse.host.domain	Yes	New installation or if the hostname of the Finesse box changes.
NGX_AUTH_URL Specifies the Finesse URL to fetch the users list to perform authentication at proxy. **Default:** https://reverseproxy.host.domain:8445/finesse/api/UserAuth	Yes	Replace "reverseproxy.host.domain" with the FQDN of the reverse-proxy.

Cisco Finesse properties that are not recommended to be altered

Note These properties are provided for reference and they exist in the configuration to provide flexibility to adjust the behavior if necessary, in exceptional situations and aren't recommended changing casually without extensive testing.

Property Name, Description, and Default	**Change Recommended?**	**When to Change?**
NGX_FIN_UPSTREAM_KEEPALIVE Specifies the proxy backend configurations for Finesse, and activates the cache for connections to an upstream IdS server. The value sets the maximum number of idle keep-alive connections to upstream servers that are preserved in the cache of each worker process. When this number is exceeded, the least recently used connections are closed. **Default:** 128	No	This default is the optimal value for keeping alive the upstream connection per component.
TEMPLATE_TYPE Specifies the component template type—valid values are: chat, cuic, Finesse, ids, livedata, cuic_12.6(1), livedata_12.6(1), and idp-adfs3. **Default:** Finesse	No	Never
NGX_FIN_DESKTOP_CACHE_SIZE Specifies the proxy cache configuration for the Finesse desktop and the initial size of the Finesse desktop endpoints cache. **Default:** 10m	No	The default is the optimal value.
NGX_FIN_DESKTOP_CACHE_MAX_SIZE Specifies the maximum size for the Finesse desktop endpoints cache. When the size exceeds or there isn't enough free space, it removes the least recently used data. **Default:** 50m	No	The default is the optimal value.
NGX_FIN_DESKTOP_CACHE_INACTIVE_DURATION Specifies the content inactive duration for the Finesse desktop endpoints cache. Cached data that is not accessed during the time specified gets removed from the cache regardless of their duration. **Default:** 3y	No	The default is the optimal value. After this time, the cache contents expire.
NGX_FIN_SHINDIG_CACHE_SIZE Specifies the proxy cache configuration for Finesse Shindig and the initial size of the Finesse Shindig endpoints cache. **Default:** 10m	No	
NGX_FIN_SHINDIG_CACHE_MAX_SIZE Max size for Finesse shindig endpoints cache. When the size exceeds or there isn't enough free space, it removes the least recently used data. **Default:** 500m	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_FIN_SHINDIG_CACHE_INACTIVE_DURATION Specifies the content inactive duration for the Finesse desktop endpoints cache. Cached data that aren't accessed during the time specified get removed from the cache regardless of their freshness. **Default:** 3y	No	
NGX_FIN_OPENFIRE_CACHE_SIZE Specifies the initial size of the Finesse openfire endpoints cache. **Default:** 10m	No	
NGX_FIN_OPENFIRE_CACHE_MAX_SIZE Specifies the maximum size for the Finesse openfire endpoints cache. When the size exceeds or there isn't enough free space, it removes the least recently used data. **Default:** 10m	No	
NGX_FIN_OPENFIRE_CACHE_INACTIVE_DURATION Specifies the content inactive duration for the Finesse desktop endpoints cache. Cached data that is not accessed during the time specified is removed from the cache regardless of their duration. **Default:** 3y	No	
NGX_FIN_REST_CACHE_SIZE Specifies the initial size of the Finesse REST endpoints cache **Default:** 10m	No	
NGX_FIN_REST_CACHE_MAX_SIZE Specifies the maximum size for the Finesse REST endpoints cache. When the size exceeds or there isn't enough free space, it removes the least recently used data. **Default:** 1500m	No	
NGX_FIN_REST_CACHE_INACTIVE_DURATION Specifies the content inactive duration for the Finesse desktop endpoints cache. Cached data that is not accessed during the time specified, get removed from the cache regardless of their duration. **Default:** 40m	No	
NGX_FIN_LAYOUT_CACHE_SIZE Specifies the initial size of the Finesse layout endpoints cache. **Default:** 150m	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_FIN_LAYOUT_CACHE_MAX_SIZE Specifies the maximum size for the Finesse layout endpoints cache. When the size exceeds or there isn't enough free space, the system removes the least recently used data. **Default:** 300m	No	
NGX_FIN_LAYOUT_CACHE_INACTIVE_DURATION Specifies the inactive duration for content stored in the Finesse desktop endpoints cache. Cached data that aren't accessed during the time specified get removed from the cache regardless of their duration. **Default:** 40m	No	
NGX_FIN_HTTP1_CONN_LIMIT Specifies the number of concurrent HTTP/1.1 connections allowed per source IP **Default:** 12	No	
NGX_FIN_HTTP2_CONN_LIMIT Specifies the number of concurrent HTTP/2 streams allowed per source IP. **Default:** 150	No	
NGX_FIN_DESKTOP_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit for desktop endpoints **Default:** 50	No	
NGX_FIN_SHINDIG_CORE_RPC_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit for shindig rpc endpoints **Default:** 40	No	
NGX_FIN_SHINDIG_IFR_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit for shindig ifr endpoints **Default:** 30	No	
NGX_FIN_SSOVALVE_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit for SSO valve endpoints **Default:** 5	No	
NGX_FIN_LIMIT_REQ_STATUS Specifies the HTTP response code to return when the HTTP request rate reaches the limit **Default:** 429	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_FIN_PROXY_BUFFER_SIZE Specifies the size of the buffer used for reading the first part of the response received from the server, which is enabled for reverse-proxy. Configurations are overridden from the common config for Finesse. **Default:** 8k	No	
NGX_FIN_MAX_TEMP_FILE_SIZE Applies when buffering of responses from the server (which is enabled for reverse-proxy) is enabled. If the whole response doesn't fit into the buffers set by the NGX_FIN_PROXY_BUFFER_SIZE, the system saves part of the response in a temporary file. **Default:** 100m	No	

Chat properties

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_PRXY_CHAT_PORT Specifies the reverse-proxy port over which this chat server is accessed. **Default:** 5280	Yes, if necessary.	New installation or if not changed, 5280 is the port for the chat server.
NGX_PRXY_CHAT_HOSTNAME Specifies the reverse-proxy hostname over which this chat server is accessed. **Default:** reverseproxy.host.domain	Yes	New installation or if the chat hostname of the reverse-proxy is accessed from the the internet changes.
NGX_CHAT_HOSTNAME Specifies the hostname of the chat server. **Default:** chat1.host.domain	Yes	New installation or if the host name of the upstream chat server changes.
NGX_CHAT_HOST[1-8]_PROXY Required for substituting user home and backup node information in the log-in response. (Similar keys configurations exist for 4 HA clusters of chat servers.) **Default:** reverseproxy.host.domain:5280/reverseproxy-sub.host.domain:15280	Yes	New installation or if the chat server proxy hostname FQDN changes.

Property Name, Description, and Default	**Change Recommended?**	**When to Change?**
NGX_CHAT_HOST[1-8] Specifies the proxy access information for all chat nodes in the deployment. Required for substituting user home and backup node information in the log-in response. (Similar keys configurations exist for 4 HA clusters of chat servers.) **Default:** chat[1-4].host.domain/chat[1-4]-sub.host.domain	Yes	New installation or if the chat server upstream hostname FQDN changes.
NGX_CHAT_BIND_PATH Specifies the binding URL for the chat server. **Default:** httpbinding	Yes	If the binding URL for the chat server changes.
NGX_AUTH_URL Specifies the Finesse URL to fetch the users list to perform authentication at proxy. **Default:** https://proxy.host.domain:8445/finesse/api/UserAuth	Yes	New installation or if proxy.host.domain is the hostname of the the reverse-proxy.

Chat properties that are not recommended to be altered

Note These properties are provided for reference and they are available in the configuration. They provide flexibility to adjust the behavior if necessary, in exceptional situations, and aren’t recommended for changing casually without extensive testing.

Property Name, Description, and Default	**Change Recommended?**	**When to change?**
TEMPLATE_TYPE Specifies the component template type—valid values are: chat, cuic, finesse, ids, livedata. **Default:** chat	No	
NGX_CHAT_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit. **Default:** 10	No	
NGX_CHAT_PORT Specifies the port for the chat server. **Default:** 5280	No	New installation or if the port number for the upstream chat server is different.

Property Name, Description, and Default	Change Recommended?	When to change?
NGX_CHAT_HTTP2_CONN_LIMIT Specifies the number of concurrent HTTP/2 streams allowed per source IP. **Default:** 30	No	
NGX_CHAT_HTTP1_CONN_LIMIT Specifies the rate limit for desktop chat, the number of concurrent HTTP/1.1 connections allowed per source IP. **Default:** 6	No	

Cloud Connect properties

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_PRXY_CLOUDCONNECT_PORT Specifies the reverse-proxy port over which this Cloud Connect server is accessed. **Default:** 443	Yes, if necessary.	If there is a change in the port number.
NGX_PRXY_CLOUDCONNECT_HOSTNAME Specifies the reverse-proxy hostname over which this Cloud Connect server is accessed. **Default:** reverseproxy.host.domain	Yes	If there is a change in the chat hostname of the reverse-proxy that has must accessed from the internet.
NGX_CLOUDCONNECT_HOSTNAME Specifies the hostname for the Cloud Connect server and the hostname of the Publisher or Primary Cloud Connect node. (Conversely on the alternate side) **Default:** cloudconnect.host.domain	Yes	If there is a change in the FQDN for the upstream Cloud Connect server.
NGX_CLOUDCONNECT_FAILOVER_HOSTNAME Specifies the hostname of the Cloud Connect failover node and the hostname of the Subscriber or Secondary Cloud Connect node. (Conversely on the alternate side) **Default:** cloudconnct.failover.hostname	Yes	If there is change in the FQDN of the alternate Cloud Connect server.
NGX_CLOUDCONNECT_CLIENT_IPS Creates a list of known IP addresses for clients connecting to Cloud Connect services like DigitalRouting and UserSync. **Default:** 35.161.238.252|35.166.68.236|34.240.73.178|3.9.155.97|54.206.189.15|52.62.185.51| 13.210.45.137|52.40.46.90|52.214.81.91|3.9.151.19|3.105.22.233|52.17.23.194	Yes	Validate and update the list when there is a change in the IP addresses.

Cloud Connect properties that are not recommended to be altered

Note These properties are provided for reference and they exist in the configuration to provide flexibility to adjust the behavior if necessary, in exceptional situations and aren't recommended changing casually without extensive testing.

Property Name, Description, and Default	Change Recommended?	When to Change?
TEMPLATE_TYPE Specifies the component template type—valid values are: chat, cuic, finesse, ids, livedata, cuic_12.6(1), livedata_12.6(1), and idp-adfs3. **Default:** cloudconnect	No	
NGX_CLOUDCONNECT_USER_SYNC_CALLBACK_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit for user sync bulk requests. **Default:** 10	No	
NGX_CLOUDCONNECT_UPSTREAM_KEEPALIVE Activates the cache for connections to the upstream Cloud Connect server. Specifies the maximum number of idle keepalive connections to upstream servers that are preserved in the cache of each worker process. When the count exceeds this number, the least recently used connections close. **Default:** 150	No	
NGX_CLOUDCONNECT_LIMIT_REQ_STATUS Specifies the HTTP response code to return when the HTTP request rate reaches the limit. **Default:** 429	No	
NGX_CLOUDCONNECT_HTTP2_CONN_LIMIT Specifies the number of concurrent HTTP/2 streams allowed per source IP. **Default:** 250	No	
NGX_CLOUDCONNECT_HTTP1_CONN_LIMIT Specifies the number of concurrent HTTP/1.1 connections allowed per source IP—the rate limit for Digital Routing requests. **Default:** 50	No	
NGX_CLOUDCONNECT_HEALTHCHECK_TIMEOUT Specifies the timeout for the health check API in milliseconds, if there is a failure. **Default:** 2000	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_CLOUDCONNECT_HEALTHCHECK_SSL_VERIFY Specifies whether to verify the SSL certificate for health checks. **Default:** FALSE	No	
NGX_CLOUDCONNECT_HEALTHCHECK_RISE Specifies the number of successive health check successes before turning up a peer. **Default:** 2	No	
NGX_CLOUDCONNECT_HEALTHCHECK_RESPONSE_CODES Generates a comma-separated list of valid HTTP status codes for the health check API. **Default:** {200}	No	
NGX_CLOUDCONNECT_HEALTHCHECK_INTERVAL Specifies the interval between health checks in milliseconds. **Default:** 2000	No	
NGX_CLOUDCONNECT_HEALTHCHECK_FALL Specifies the number of successive health check failures before turning down a peer. **Default:** 2	No	
NGX_CLOUDCONNECT_HEALTHCHECK_CONCURRENCY Specifies the number of concurrent health checks. **Default:** 2	No	
NGX_CLOUDCONNECT_HEALTHCHECK_API Specifies the active health check configurations for the DR API. The system uses this URL to check the health of the Cloud Connect. **Default:** /drapi/v1/ping	No	
NGX_CLOUDCONNECT_DR_TASK_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit for digital routing tasks. **Default:** 200	No	

Cisco IdS properties

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_PRXY_IDS_HOSTNAME Specifies the reverse-proxy hostname over which this IdS host is accessed. **Default:** reverseproxy.host.domain	Yes	New installation or if the reverse-proxy IdS hostname changes.
NGX_PRXY_IDS_PORT Specifies the reverse-proxy port over which this IdS host is accessed. **Default:** 8553	Yes, If required.	New installation or if the reverse-proxy port changes.
NGX_IDS_HOSTNAME IdS server actual hostname **Default:** ids.host.domain	Yes	New installation or if the FQDN host name of the IdS server changes.

Cisco IdS properties that are not recommended to be altered

Note These properties are provided for reference and they are available in the configuration. They provide flexibility to adjust the behavior if necessary, in exceptional situations, and aren’t recommended changing casually without extensive testing.

Property Name, Description, and Default	Change Recommended?	When to Change?
TEMPLATE_TYPE Specifies the component template type—valid values are: chat, cuic, finesse, ids, livedata, cuic_12.6(1), livedata_12.6(1), and idp-adfs3. **Default:** ids	No	
NGX_IDS_UPSTREAM_KEEPALIVE Proxy backend configurations for IDS. Activates the cache for connections to the upstream IDP server. Specifies the maximum number of idle keepalive connections to upstream servers that are preserved in the cache of each worker process. When the count exceeds this number, the least recently used connections close. **Default:** 128	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_IDS_HTTP1_CONN_LIMIT Rate limits configuration for IdS. Specifies the number of concurrent HTTP/1.1 connections allowed per source IP. **Default:** 4	No	
NGX_IDS_HTTP2_CONN_LIMIT Specifies the number of concurrent HTTP/2 streams allowed per source IP. **Default:** 4	No	
NGX_IDS_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit. **Default:** 4	No	

IdP Properties (ADFS 3.0)

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_PRXY_IDP_HOSTNAME Specifies the reverse-proxy hostname over which this IdP host will be accessed. **Default:** adfs-reverseproxy.host.domain	Yes	New installation or if the FQDN of the reverse-proxy IdP host changes.
NGX_PRXY_IDP_PORT Specifies the reverse-proxy port over which this IdP host will be accessed. **Default:** 443	Yes, If required.	New installation or if unchanged, reverse-proxy uses the 443 port for CUIC.
NGX_IDP_HOSTNAME Specifies the CUIC server hostname. **Default:** idp.host.domain	Yes	New installation or if the FQDN of the IdP host changes.

IdP Properties (ADFS 3.0) that are not recommended to be altered

Note These properties are provided for reference and they are available in the configuration. They provide flexibility to adjust the behavior if necessary, in exceptional situations, and are not recommended changing casually without extensive testing.

Property Name, Description, and Default	Change Recommended?	When to Change?
TEMPLATE_TYPE Specifies the component template type—valid values are: chat, cuic, finesse, ids, livedata, cuic_12.6(1), livedata_12.6(1), and idp-adfs3. **Default:** idp-adfs3	No	
NGX_IDP_UPSTREAM_KEEPALIVE Proxy backend configurations for IDP. Activates the cache for connections to upstream IDP server Specifies the maximum number of idle keepalive connections to upstream servers that are preserved in the cache of each worker process. When the count exceeds this number, the least recently used connections close. **Default:** 128	No	

Livedata Properties

LiveData 12.6(1) Properties

Note Use livedata_12.6(1).env when the upstream Livedata is still on Release12.6(1). Otherwise, use livedata.env.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_PRXY_LD_HOSTNAME Specifies the reverse-proxy hostname over which this LD host is accessed. **Default:** reverseproxy.host.domain	Yes	If the reverse-proxy livedata host FQDN changes.
NGX_PRXY_LD_PORT Specifies the reverse-proxy port over which this livedata host is accessed. **Default:** 12005	Yes, If required.	If there is a change in the reverse-proxy port for livedata.
NGX_PRXY_LD_SCKT_IO_PORT Specifies the reverse-proxy port over which the socket IO endpoint of this livedata host will be accessed. **Default:** 12008	Yes, If required.	If there is a change in the reverse-proxy port for socket IO connections.
NGX_LD_HOSTNAME Specifies the Livedata server hostname. **Default:** livedata.host.domain	Yes	If there is a change in the upstream livedata host FQDN.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_LD_SIO_UPSTREAM_KEEPALIVE Activates the cache for connections to the upstream livedata server socketio endpoint. **Default:** 128	No	—
NGX_LD_BACKEND_FAILOVER Livedata failover configuration. This is used to translate the location header during failover. For SideA LiveData, point to SideB LiveData and for SideB Livedata, point to SideA Livedata.	Yes	If there is a change in the FQDN or port.
NGX_LD_BACKEND_PROXY_FAILOVER Livedata proxy failover configuration. Each side points to the other side of the reverse-proxy node.	Yes	If there is a change in the FQDN or port.

LiveData 12.6(1) properties that are not recommended to be altered

Note These properties are provided for reference and they are available in the configuration. They provide flexibility to adjust the behavior if necessary, in exceptional situations, and aren't recommended changing casually without extensive testing.

Property Name, Description, and Default	Change Recommended?	When to Change?
TEMPLATE_TYPE Specifies the component template type—valid values are: chat, cuic, finesse, ids, livedata, cuic_12.6(1), livedata_12.6(1), and idp-adfs3. **Default:** livedata_12.6(1)	No	—
NGX_LD_WEB_UPSTREAM_KEEPALIVE Proxy backend configurations for livedata. Activates the cache for connections to the upstream livedata server. Specifies the maximum number of idle keep alive connections to upstream servers that are preserved in the cache of each worker process. When the count exceeds this number, the least recently used connections close. **Default:** 128	No	—

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_LD_SIO_UPSTREAM_KEEPALIVE Activates the cache for connections to the upstream livedata server socketio endpoint. **Default:** 128	No	—
NGX_LD_HTTP1_CONN_LIMIT Specifies the number of concurrent HTTP/1.1 connections allowed per source IP. **Default:** 12	No	—
NGX_LD_HTTP2_CONN_LIMIT Specifies the number of concurrent HTTP/2 streams allowed per source IP. **Default:** 150	No	—
NGX_LD_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit. **Default:** 25	No	—
NGX_LD_PROXY_BUFFER_SIZE **Default:** 8k	No	—
NGX_LD_MAX_TEMP_FILE_SIZE **Default:** 100m	No	—

LiveData 12.6(2) Properties

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_PRXY_LD_HOSTNAME Specifies the reverse-proxy hostname over which this LD host is accessed. **Default:** reverseproxy.host.domain	Yes	If the reverse-proxy livedata host FQDN changes.
NGX_PRXY_LD_PORT Specifies the reverse-proxy port over which this livedata host is accessed. **Default:** 443	Yes, If required.	If the reverse-proxy port for livedata changes.
NGX_LD_HOSTNAME Specifies the Livedata server hostname. **Default:** livedata.host.domain	Yes	If the upstream livedata host FQDN changes.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_LD_BACKEND_FAILOVER Specifies the livedata failover configurations. This is used to translate the location header during failover. **Default:** hostname:port	Yes	For the SideA upstream LiveData, point to the upstream SideB LiveData. For the SideB upstream, point to the upstream SideA.
NGX_LD_BACKEND_PROXY_FAILOVER Specifies the livedata failover node for this livedata host. This is used to translate the location header during failover. **Default:** hostname:port	Yes	Point to the other side of the reverse-proxy node.

LiveData 12.6(2) properties that are not recommended to be altered

Note These properties are provided for reference and they exist in the configuration to provide flexibility to adjust the behavior if necessary, in exceptional situations and aren’t recommended to be changed casually without extensive testing.

Property Name, Description, and Default	Change Recommended?	When to Change?
TEMPLATE_TYPE Specifies the component template type—valid values are: chat, cuic, finesse, ids, livedata, cuic_12.6(1), livedata_12.6(1), and idp-adfs3. **Default:** livedata	No	
NGX_LD_WEB_UPSTREAM_KEEPALIVE Proxy backend configurations for livedata. Activates the cache for connections to the upstream livedata server. Specifies the maximum number of idle keepalive connections to upstream servers that are preserved in the cache of each worker process. When the count exceeds this number, the least recently used connections are closed. **Default:** 128	No	
NGX_LD_SIO_UPSTREAM_KEEPALIVE Activates the cache for connections to the upstream livedata server socketIO endpoint. **Default:** 128	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_LD_HTTP1_CONN_LIMIT Specifies the number of concurrent HTTP/1.1 connections allowed per source IP. **Default:** 12	No	
NGX_LD_HTTP2_CONN_LIMIT Specifies the number of concurrent HTTP/2 streams allowed per source IP. **Default:** 150	No	
NGX_LD_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit. **Default:** 25	No	
NGX_LD_PROXY_BUFFER_SIZE Sets the size of the buffer used for reading the first part of the response received from the server, which is enabled with reverse-proxy. **Default:** 8k	No	
NGX_LD_MAX_TEMP_FILE_SIZE When buffering of responses from the server (which is enabled with reverse-proxy) is enabled, and the whole response doesn't fit into the NGX_LD_PROXY_BUFFER_SIZE, a part of the response can be saved to a temporary file. **Default:** 100m	No	

CUIC Properties

Unified Intelligence Center 12.6(1) properties

Note Use this env file when the upstream Unified Intelligence Center is still on 12.6(1) release. Otherwise use cuic.env.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_PRXY_CUIC_HOSTNAME Specifies the reverse-proxy hostname over which this CUIC host is accessed. **Default:** reverseproxy.host.domain	Yes	If the reverse-proxy CUIC host FQDN changes.

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_PRXY_CUIC_PORT Specifies the reverse-proxy port over which this cuic host is accessed. **Default:** 8444	Yes, If required.	New installation or if the port number changes.
NGX_PRXY_CUIC_DOC_PORT Specifies the reverse-proxy port over which this CUIC host doc endpoint is accessed. **Default:** 8447	Yes, If required.	New installation or if the port number changes.
NGX_CUIC_HOSTNAME Specifies the CUIC server hostname. **Default:** cuic.host.domain	Yes	If the upstream CUIC host FQDN changes.

Unified Intelligence Center 12.6(1) properties that are not recommended to be altered

Note These properties are provided for reference and they are available in the configuration. They provide flexibility to adjust the behavior if necessary, in exceptional situations, and aren’t recommended changing casually without extensive testing.

Property Name, Description, and Default	Change Recommended?	When to Change?
TEMPLATE_TYPE Specifies the component template type—valid values are: chat, cuic, finesse, ids, livedata, cuic_12.6(1), livedata_12.6(1), idp-adfs3. **Default:** cuic_12.6(1)	No	Never
NGX_CUIC_UPSTREAM_KEEPALIVE Activates the cache for connections to the upstream CUIC server. Specifies the maximum number of idle keep alive connections to upstream servers that are preserved in the cache of each worker process. When this number exceededs, the least recently used connections are closed. **Default:** 128	No	—
NGX_CUICDOC_UPSTREAM_KEEPALIVE Activates the cache for connections to the upstream CUIC server doc endpoint. **Default:** 128	No	—

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_CUIC_CACHE_SIZE Specifies the initial size of the CUIC proxy cache. **Default:** 15m	No	—
NGX_CUIC_CACHE_MAX_SIZE Specifies the maximum size for the CUIC cache. When the size exceeds or there's not enough free space, the system removes the least recently used data. **Default:** 50m	No	—
NGX_CUIC_CACHE_INACTIVE_DURATION Specifies the content inactive duration for the CUIC cache. Cached data that is not accessed during the time specified get removed from the cache regardless of their freshness. **Default:** 3y	No	—
NGX_CUICDOC_CACHE_SIZE Specifies the initial size of the CUIC doc cache. **Default:** 15m	No	—
NGX_CUICDOC_CACHE_MAX_SIZE Specifies the maximum size for the CUIC doc cache. When the size exceeds or there isn’t enough free space, the system removes the least recently used data. **Default:** 50m	No	—
NGX_CUICDOC_CACHE_INACTIVE_DURATION Specifies the content inactive duration for CUIC doc cache. Cached data that aren’t accessed during the time specified get removed from the cache regardless of their freshness. **Default:** 3y	No	—
NGX_CUIC_HTTP1_CONN_LIMIT Specifies the number of concurrent HTTP/1.1 connections allowed per source IP. **Default:** 12	No	—
NGX_CUIC_HTTP2_CONN_LIMIT Specifies the number of concurrent HTTP/2 streams allowed per source IP. **Default:** 150	No	—

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_CUIC_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit. **Default:** 100	No	—
NGX_CUIC_HISTORICAL_REPORT_CONN_LIMIT Specifies the number of concurrent connections allowed, per source IP, for CUIC historical reports endpoints. **Default:** 4	No	—
NGX_CUIC_HISTORICAL_REPORT_NEW_CONN_LIMIT Specifies the number of concurrent connections allowed, per source IP, for CUIC historical reports newRest endpoints. **Default:** 4	No	—
NGX_CUIC_HISTORICAL_REPORT_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit for CUIC historical reports endpoints. **Default:** 4	No	—
NGX_CUIC_REALTIME_REPORT_CONN_LIMIT Specifies the number of concurrent connections allowed, per source IP, for CUIC realtime reports endpoints. **Default:** 4	No	—
NGX_CUIC_REALTIME_REPORT_NEW_CONN_LIMIT Specifies the number of concurrent connections allowed, per source IP, for CUIC realtime reports newRest endpoints. **Default:** 4	No	—
NGX_CUIC_REALTIME_REPORT_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit for CUIC realtime reports endpoints. **Default:** 4	No	—
NGX_CUIC_REPORT_EXECUTION_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit for CUIC reports execution endpoints. **Default:** 4	No	—

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_CUIC_PROXY_BUFFER_SIZE Specifies the size of the buffer used for reading the first part of the response received from the server, which is enabled for reverse-proxy. **Default:** 8k	No	—
NGX_CUIC_MAX_TEMP_FILE_SIZE When buffering of responses from the server (which is enabled for reverse-proxy) is enabled, and the whole response doesn't fit into the buffers set by the NGX_CUIC_PROXY_BUFFER_SIZE, a part of the response is saved to a temporary file. **Default:** 100m	No	—

Unified Intelligence Center 12.6(2) properties

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_PRXY_CUIC_HOSTNAME Specifies the reverse-proxy hostname over which this cuic host is accessed. **Default:** reverseproxy.host.domain	Yes	If the reverse-proxy CUIC host FQDN changes.
NGX_PRXY_CUIC_PORT Specifies the reverse-proxy port over which this cuic host is accessed. **Default:** 443	Yes, If required.	If the reverse-proxy port changes.
NGX_CUIC_HOSTNAME Specifies the CUIC server hostname. **Default:** cuic.host.domain	Yes	If the upstream CUIC host FQDN changes.

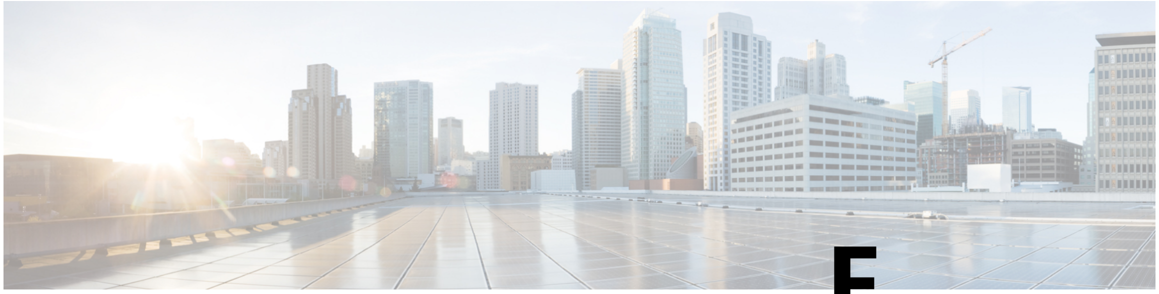
Unified Intelligence Center 12.6(2) properties that are not recommended to be altered

Note These properties are provided for reference and they exist in the configuration to provide flexibility to adjust the behavior if necessary, in exceptional situations and aren’t recommended changing casually without extensive testing.

Property Name, Description, and Default	Change Recommended?	When to Change?
TEMPLATE_TYPE Specifies the component template type—valid values are: chat, cuic, finesse, ids, livedata, cuic_12.6(1), livedata_12.6(1), and idp-adfs3. **Default:** cuic	No	
NGX_CUIC_UPSTREAM_KEEPALIVE Activates the cache for connections to the upstream CUIC server. Specifies the maximum number of idle keepalive connections to upstream servers that are preserved in the cache of each worker process. When the count exceeds this number, the system closes the least recently used connections. **Default:** 128	No	
NGX_CUIC_CACHE_SIZE Specifies the initial size of the CUIC cache. **Default:** 30m	No	
NGX_CUIC_CACHE_MAX_SIZE Specifies the maximum size for the CUIC cache. When the size exceeds the set maximum, or there isn't enough free space, the system removes the least recently used data. **Default:** 100m	No	
NGX_CUIC_CACHE_INACTIVE_DURATION Specifies the content inactive duration for the CUIC cache. Cached data that aren't accessed during the time specified get removed from the cache regardless of their freshness. **Default:** 3y	No	
NGX_CUIC_HTTP1_CONN_LIMIT Specifies the number of concurrent HTTP/1.1 connections allowed per source IP. **Default:** 12	No	
NGX_CUIC_HTTP2_CONN_LIMIT Specifies the number of concurrent HTTP/2 streams allowed per source IP. **Default:** 150	No	
NGX_CUIC_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit. **Default:** 100	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_CUIC_HISTORICAL_REPORT_CONN_LIMIT Specifies the number of concurrent connections allowed per source IP for CUIC historical reports endpoints. **Default:** 4	No	
NGX_CUIC_HISTORICAL_REPORT_NEW_CONN_LIMIT Specifies the number of concurrent connections allowed per source IP for CUIC historical reports newRest endpoints. **Default:** 4	No	
NGX_CUIC_HISTORICAL_REPORT_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit for CUIC historical reports endpoints. **Default:** 4	No	
NGX_CUIC_REALTIME_REPORT_CONN_LIMIT Specifies the number of concurrent connections allowed per source IP for CUIC realtime reports endpoints. **Default:** 4	No	
NGX_CUIC_REALTIME_REPORT_NEW_CONN_LIMIT Specifies the number of concurrent connections allowed per source IP for CUIC realtime reports new REST endpoints. **Default:** 4	No	
NGX_CUIC_REALTIME_REPORT_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit for CUIC realtime reports endpoints. **Default:** 4	No	
NGX_CUIC_REPORT_EXECUTION_REQUEST_BURST_LIMIT Specifies the HTTP request burst limit for CUIC reports execution endpoints. **Default:** 4	No	
NGX_CUIC_PROXY_BUFFER_SIZE Configurations overridden from common config for CUIC. Specifies the size of the buffer used for reading the first part of the response received from the server, which is enabled for reverse-proxy. **Default:** 8k	No	

Property Name, Description, and Default	Change Recommended?	When to Change?
NGX_CUIC_MAX_TEMP_FILE_SIZE When buffering of responses from the server (which is enabled for reverse-proxy) is enabled, and the whole response doesn’t fit into the buffers set by the NGX_CUIC_PROXY_BUFFER_SIZE, a part of the response is saved to a temporary file. **Default:** 100m	No	



APPENDIX E

Guidelines for Custom Reverse Proxy Deployment

This appendix provides you with guidelines for deploying an appropriate reverse proxy.

Note These guidelines are provided as best effort and Cisco does not claim support for any custom reverse proxy deployments.

Reverse proxy selection and configuration for digital channel interactions

Minimum and additional requirements

Minimum requirements

Contact Center administrators must select an appropriate reverse proxy. Any reverse proxy that meets the following minimum requirements can be used:

- Supports HTTP2/TLS 1.2.
- Has proper logging mechanism for easy debugging of issues and includes Tracking ID to easily track the task requests.
- Supports failover between the Cloud Connect nodes with health check.
- Supports X-Forwarded headers. The solution uses these headers to decide how to handle a request when front-ended with load balancer.

Additional Requirements

Some desirable requirements in a reverse-proxy are as follows:

- Consider deploying proxies that are built on non-blocking IO-based technology instead of the traditional thread-per-request architecture, to scale better.

- Apply rate limiting and configure allowed list of Webex Connect or Load balancer IPs.

Performance and hardware recommendation

For details, see Performance and Hardware Recommendations, on page 278.

Configure custom reverse proxy

Install the host OS and reverse-proxy of your choice. Consider the following points while configuring the reverse-proxy:

- Configure SSL certificates as required.
- Configure the Mutual Transport Layer Security (mTLS) authentication between reverse proxy and Cloud Connect.
  - Add the list of trusted reverse proxy IP addresses and the corresponding hostnames on the publisher and subscriber nodes of Cloud Connect. For details, see Add Proxy IP, on page 72.
  - Configure SSL certificate verification to establish communication between the reverse proxy host and the Digital Routing service. For details, see Configure reverse proxy host verification, on page 74.
- Configure both nodes (publisher and subscriber) of Cloud Connect for task requests. Implement HTTP health check and failover to the subscriber node. The health check API that the Digital Routing service supports is */drapi/v1/ping*.
- The DataConn callback requests are routed through the reverse proxy. Configure the DataConn requests to the upstream Cloud Connect publisher node. The DataConn service runs only on the publisher node of CloudConnect.

Host header configuration

The following are the mandatory HTTP headers that reverse-proxy has to set along with the actual headers set by the client before forwarding the headers to the Finesse server.

Table 21: Host header and description

Header	Description
X-Client-IP X-Real-IP	The reverse-proxy must populate this custom header as the client's IP address before forwarding it to Cloud Connect.

Header	Description
Host	The Host request header specifies the host and port number of the server to which the request is being sent. If no port is included, the default port for the service requested (for example, 443 for an HTTPS URL and 80 for an HTTP URL) is used. An HTTP/1.1 proxy ensures that any request message it forwards contains an appropriate Host header field to identify the service being requested by the proxy. This value is used by Cloud Connect to find if the request is sent via the allowed list of proxies configured in Cloud Connect.
X-Forwarded-For	The `X-Forwarded-For` (XFF) header is used for identifying the originating IP address of a client connecting to a web server through an HTTP proxy or a load balancer. The IP of the reverse-proxy has to be appended or set. Cloud Connect uses this header to find if the request is from the allowed list of reverse-proxies. When the request is forwarded through multiple reverse-proxies or load balancer, the values of all reverse-proxies are appended to the rightmost value of this header.
X-Forwarded-Port	The reverse-proxy should set the listening port on this header. Cloud Connect server receives all the requests internally via 8445 port.
Connection	Any Connection value in the HTTP header that is set by the client must be cleared and forwarded to the Cloud Connect server so that the server decides the connection management and not the client. This prevents security outages.

Reverse proxy selection and configuration for VPN-less access to Finesse Desktop

Minimum and additional requirements

Minimum requirements

Contact Center administrators must select an appropriate reverse-proxy. Any reverse-proxy that meets the following minimum requirements can be used:

- Supports HTTP2/TLS 1.2 and secure Websockets.

- Has proper logging mechanism for easy debugging of issues
- Supports multiple Finesse, IdS, and CUIC servers from a single reverse-proxy.
- Supports periodic revalidation of cached content. This is required because any updates or installations on the internal hosts don't require a manual intervention to clear the cached content of the proxy.
- Supports custom authentications or provides alternative mechanisms such as an enterprise login to prevent unauthenticated access of solution components.

Note When you use Cisco-provided reverse-proxy configuration, the requests are authenticated at the proxy before they are forwarded to the upstream servers. When you are configuring a custom reverse-proxy, you must create this authentication layer if they have to be as secure as the Cisco provided configuration. You should consider this configuration step while planning to implement VPN-less access to Finesse using a custom reverse-proxy.

- Enables caching of static resources with support for cache-control header to reduce DoS/DDoS attack vectors and to scale the proxy. Any proxy that needs to support more than a few hundred users and does not provide response caching features should be deployed with a Content Delivery Network (CDN) with support for cache-control headers so that load and security guidelines are met.

Note CDN deployment is also recommended with caching proxies such as OpenResty® Nginx to eliminate the impact of DDoS attacks.

- Supports X-Forwarded headers. These headers are used by the solution to decide how to handle a request.

Additional Requirements

Some desirable requirements in a reverse-proxy are as follows:

- Consider deploying proxies that are built on non-blocking IO-based technology instead of the traditional thread-per-request architecture, to scale better.
- Consider proxies that provide response substitution capabilities which allow workarounds for custom gadgets as custom gadgets may not work with reverse-proxy directly.

Note Finesse Desktop Chat over reverse-proxy requires response substitution capability.

- Support for port-based forwarding can be used to reduce the cost of deployment by avoiding the need for multiple externally resolvable hostnames, public DNS records, and corresponding certificates for each internal server that has to be accessed.
- Support for custom plugin/modules, which can be used to enhance the authentication model and provide a more robust security posture.

Performance and hardware recommendation

For details, see Performance and Hardware Recommendations, on page 278.

Configure Reverse-Proxy

Install the host OS and reverse-proxy of your choice. Consider the following points while configuring the reverse-proxy:

- Configure SSL certificates as required.
- Refer to the specific proxy documentation and configure the proxy rules for each service with the same host and port that is configured in the mapping file.
- IdS and IdP trust should be configured before proxy mapping configuration is done. Otherwise, proxy configuration changes will not be processed by IdS.
- For IdS hosts, if proxy configuration is changed, the administrator must re-establish trust on IdP for new IdS proxy hosts after downloading new metadata file from IdS admin.
- For Finesse hosts, if proxy configuration is changed, the administrator must manually add or update the allowed Finesse client redirect URIs from IdS administration interface.
- Whenever SAML certificate is regenerated or IdP metadata is uploaded, proxy configurations are generated afresh.

To secure the reverse-proxy, refer to the *Security Guidelines* section in the Security Guide for Cisco Unified ICM/Contact Center Enterprise .

Host Header Configuration

The following are the mandatory HTTP headers that reverse-proxy has to set along with the actual headers set by the client before forwarding the headers to the Finesse server.

Table 22:

Header	Description
X-Client-IP	The reverse-proxy should populate this custom header as the client's IP address before forwarding it to the Finesse server. This is used to log the client's IP in the Finesse server.

Header	Description
Host	The Host request header specifies the host and port number of the server to which the request is being sent. If no port is included, the default port for the service requested (for example, 443 for an HTTPS URL and 80 for an HTTP URL) is used. An HTTP/1.1 proxy ensures that any request message it forwards contains an appropriate Host header field to identify the service being requested by the proxy. This value is used by Finesse to find if the request is sent via the allowed list of proxies configured in Finesse. The hostname and port value of the reverse-proxy should be set. Otherwise, the Finesse validation fails and returns HTTP 400 Error.
X-Forwarded-For	The **`X-Forwarded-For`** (XFF) header is used for identifying the originating IP address of a client connecting to a web server through an HTTP proxy or a load balancer. The IP of the reverse-proxy has to be appended or set. Finesse uses this header to find if the request is from the allowed list of reverse-proxies. When the request is forwarded through multiple reverse-proxies, the values of all reverse-proxies are appended to the rightmost value of this header.
X-Forwarded-Port	The reverse-proxy should set the listening port on this header. Finesse server receives all the requests internally via 8445 port. This header value helps Finesse to set the valid configuration.

The following are the standard headers manipulated by the proxy:

Table 23:

Header	Description
Connection	Any Connection value in the HTTP header that is set by the client should be cleared and forwarded to the Finesse server. This has to be done so that the Finesse server decides the connection management and not the Finesse client. This prevents security outages.
Accept-Encoding	The reverse-proxy clears the Accept-Encoding header to have better control over compression aspects of the response.