



Cisco Finesse Administration 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 NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

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

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

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

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

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

© 2010–2023 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface	xiii
Change History	xiii
About This Guide	xiii
Audience	xiv
Related Documents	xiv
Communications, Services, and Additional Information	xiv
Field Notice	xv
Documentation Feedback	xv
Conventions	xv

CHAPTER 1

Getting Started	1
User Accounts	1
Administration Tools	1
Cisco Finesse Administration Console	1
Sign In to Cisco Finesse Administration Console	2
CLI	4
Cisco Unified Operating System Administration	4
Sign In to Cisco Unified Operating System Administration	4
Configure Custom Logon Message Using OS Administration	4
Configure Custom Logon Messages Using CLIs	5
Certificate Management	5
Server-Side Certificate Management	5
Obtain and Upload CA Certificate	6
Produce Certificate Internally	7
Client-Side Certificate Acceptance	8
Client Requirements	8

- Deploy Root Certificate for Internet Explorer 8
- Set Up CA Certificate for Internet Explorer 9
- Set Up CA Certificate for Firefox Browser 9
- Set Up CA Certificate for Chrome and Edge Chromium (Microsoft Edge) Browsers 10
- Manage Expired CA Certificates 10
- Trust Self-Signed Certificate 10
- Add Certificate for HTTPS Gadget 11
- QoS Settings 12
- Localization 12

CHAPTER 2

Manage System Settings 15

- Contact Center Enterprise Administration and Data Server Settings 15
 - Configure Contact Center Enterprise Administration and Data Server Settings 17
- Contact Center Enterprise CTI Server Settings 17
 - Configure Contact Center Enterprise CTI Server Settings 20
- Cluster Settings 21
 - Configure Cluster Settings 21
- Desktop Chat Server Settings 22
 - Configure Desktop Chat Server Settings 23
- Cloud Connect Server Settings 23
 - Configure Cloud Connect Server Settings 24
- Keyboard Shortcuts 24
- Device Selection for Shared ACD Line 25

CHAPTER 3

Manage IP Address and Hostname 27

- IP Address or Hostname or Domain Name Change 27
- Change IP Address or Hostname Task List 27
 - Pre-Change Tasks 28
 - Change IP Address or Hostname using Unified Operating System GUI 29
 - Change IP Address or Hostname Using CLI 30
- Change IP Address Only 32
- Change Domain Name 33
- Post-Change Tasks and Verification 34

CHAPTER 4**Manage Call Variables Layouts 37**

- Call Variables Layouts 37
- Call Variables 37
 - Edit Call Variables 38
- Configure Call Variables Layouts 39
 - Call Variables Popover 40
- Add ECC Variables to Call Variables Layout 40
- Assign Call Variables Layouts 41
- Manipulate Call Variables Layouts with a Workflow 41

CHAPTER 5**Manage Desktop Layout 43**

- Gadgets and Components 43
- Finesse Desktop Layout XML 45
- Default Layout XML 45
- Update Default Desktop Layout 48
- Contact Center AI Gadgets 57
 - Add Agent Answers Gadget 57
 - Add Transcript Gadget 58
 - Methods to Collect Serviceability Logs 59
- Manage Digital Channels gadget 59
 - Prerequisites to configure the Manage Digital Channels gadget 60
 - Add Manage Digital Channels gadget 62
 - Manage Digital Channels gadget properties 62
 - Popover information 66
 - Methods to collect Serviceability logs 66
- Configure Multi-Tab Gadget Layout 66
- Drag-and-Drop and Resize Gadget or Component 69
- Drop Participants from Conference 70
- Customize Desktop Properties 72
- Customize Gadget Properties 74
- Horizontal Header 74
- Customize Title and Logo in the Header 75
- alternateHosts Configuration 76

- Headless Gadget Configuration 76
- Customize Icons in Left Navigation Bar 76
 - Customize Icons for Gadgets 77
 - List of Icons 79
- XML Schema Definition 91
- Live Data Reports 95
 - Prerequisites for Live Data 95
 - Add Live Data Reports to Finesse 95
 - Add Live Data Reports to Default Desktop Layout 96
 - Add Live Data Reports to Custom Desktop Layout 97
 - Add Live Data Reports to Team Layout 97
 - Modify Live Data Stock Reports for Finesse 98
 - Configure Live Data Reports with Multiple Views 98

CHAPTER 6

- Manage Phone Books 101**
 - Phone Books and Contacts 101
 - Add Phone Book 102
 - Edit Phone Book 102
 - Delete Phone Book 103
 - Import Contacts 103
 - Export Contacts 104
 - Add Contact 104
 - Edit Contact 104
 - Delete Contact 105

CHAPTER 7

- Manage Reasons 107**
 - Wrap-Up Reasons 107
 - Add Wrap-Up Reason 109
 - Edit Wrap-Up Reason 109
 - Delete Wrap-Up Reason 109
 - Force Wrap-Up Reason 109
 - Manage Reason Code Conflicts During Upgrade 110
 - Predefined System Reason Codes 110
 - Sign Out Reason Codes 112

Add Sign Out Reason Code	114
Edit Sign Out Reason Code	114
Delete Sign Out Reason Code	114
Not Ready Reason Codes	115
Add Not Ready Reason Code	116
Edit Not Ready Reason Code	117
Delete Not Ready Reason Code	117

CHAPTER 8**Manage Team Resources 119**

Team Resources	119
Assign Phone Books and Reasons to Team	120
Unassign Phone Books and Reasons from Team	120
Assign Custom Desktop Layout to Team	121
Customize Desktop Properties at Team Level	121
Assign Workflows to Team	123
Unassign Workflows from Team	123

CHAPTER 9**Manage Workflows 125**

Workflows and Workflow Actions	125
Workflow Triggers and Outbound Calls	129
Add Browser Pop Workflow Action	130
Add HTTP Request Workflow Action	131
Edit Workflow Action	132
Delete Workflow Action	132
Add Workflow	132
Edit Workflow	133
Delete Workflow	133

CHAPTER 10**Manage Connected Agents 135**

Connected Agents	135
------------------	-----

CHAPTER 11**Manage Security 137**

HTTPS Support	137
HSTS	137

Reset Security or Admin Password 138
 Cross-Origin Resource Sharing (CORS) 139
 Gadget Source Allowed List 139
 Security Enhancements 139

CHAPTER 12

Manage Finesse IP Phone Agent 141
 Finesse IP Phone Agent 141
 One Button Sign In 142
 Finesse IP Phone Service Subscription Options 143
 Set Up Application User, Web Access, and HTTPS Server Parameters 144
 Configure Finesse IP Phone Service in Unified CM 145
 Finesse IP Phone Agent Certificate Management 146
 CA-Signed Certificate 146
 Self-Signed Certificate 147
 Export CUCM Certificate 147
 Import CUCM Certificate 147
 Export Cisco Finesse Certificate 148
 Import Certificate into CUCM Trust Store 148
 Add Service Parameters for One Button Sign In 149
 Subscribe Agent Phones to Manual Subscription Service 150
 Set Up Agent Access to the Self Care Portal 150
 Finesse IP Phone Agent Login During Maintenance Mode 151

CHAPTER 13

Manage Third-Party Gadgets 153
 3rdpartygadget Account 153
 Upload Third-Party Gadgets 153
 Cisco Webex Experience Management 155
 Configure Experience Management Gadgets for Finesse Desktop 155

CHAPTER 14

Perform Routine Maintenance 157
 Cisco Finesse Services 157
 View, Start, or Stop Services 158
 Log Collection 158
 Collect Logs using Cisco Unified Real-Time Monitoring Tool 161

	Syslog Support for Critical Log Messages	161
	Finesse Performance Counters	165
	JMX Counter Thresholds	165
	Remote Account Management	168
<hr/>		
CHAPTER 15	Cisco Finesse Failover Mechanisms	169
	CTI Failover	169
	AWDB Failover	171
	Finesse Desktop Failover	171
	Desktop Behavior	172
	Finesse IP Phone Agent Failover	177
	Guidelines for Optimal Desktop Failover	178
	Failover Planning	181
	Maintenance Mode	183
<hr/>		
CHAPTER 16	Backup and Restore	185
	Backup and Restore	185
	Important Considerations	186
	SFTP Requirements	186
	Primary and Local Agents	187
	Primary Agent Duties	187
	Local Agent Duties	187
	Backup Tasks	188
	Manage Backup Devices	188
	Manage Backup Schedules	188
	Perform Manual Backup	189
	Check Backup Status	189
	Restore the Nodes in HA Setup with Rebuild	189
<hr/>		
CHAPTER 17	Supported Cisco Unified Communications OS Services	191
	Supported Cisco Unified Communications OS Services	191
<hr/>		
APPENDIX A	Cisco Finesse CLI	195
	Cisco Finesse Services	195

Finesse Log Configuration	196
Toaster Notifications	200
Finesse IPPA Inactivity Timeout	200
Configuring Queue Statistics	201
Cross-Origin Resource Sharing (CORS)	201
Gadget Source Allowed List	204
Supported Content Security Policy Directives	206
Finesse System Commands	207
Desktop Properties	208
WebProxy Service	217
utils webproxy cache clear	217
set webproxy access-log-level	218
set webproxy log-severity	219
show webproxy access-log-level	220
show webproxy log-severity	220
Service Properties	220
Log Collection Schedule	226
Upgrade	227
Shutdown	228
Replication Status	228
View Property	229
Update Property	229
Signout from Media Channels	230
Finesse Maintenance Mode Services	230
ConnectedUsersInfo	231
set webapp session maxlimit	231
set webapp session timeout	232
Update Cloud Connect Connection Time	234
AI Services Configuration	234
Digital Channels Configuration	236
Certificate Configuration	237
show tls server cert_type	237
set tls server cert_type	237

APPENDIX B**Certificates for Live Data 239**

- Certificates and Secure Communications 239
- Export Self-Signed Live Data Certificates 239
- Import Self-Signed Live Data Certificates 240
- Obtain and Upload Third-party CA Certificate 241

APPENDIX C**Certificates for Cisco Identity Service 243**

- Export Cisco Identity Service Certificates 243
- Import Cisco IdS Certificates 243



Preface

- [Change History](#), on page xiii
- [About This Guide](#), on page xiii
- [Audience](#), on page xiv
- [Related Documents](#), on page xiv
- [Communications, Services, and Additional Information](#), on page xiv
- [Field Notice](#), on page xv
- [Documentation Feedback](#), on page xv
- [Conventions](#), on page xv

Change History

This table lists the changes made to this guide. The latest changes appear at the top.

Change	See	Date
Updated Release 12.6(2)		April 2023
Added a new section Manage Digital Channel Gadgets with five subtopics.	Manage Desktop Layout > Manage Digital Channel Gadgets	
Added the Customize Gadget Properties topic.	Manage Desktop Layout > Customize Gadget Properties	
Added the Digital Channels Configuration topic.	Cisco Finesse CLI > Digital Channels Configuration	

About This Guide

The *Cisco Finesse Administration Guide* describes how to administer and maintain Cisco Finesse.

Audience

This guide is prepared for Unified Contact Center Enterprise system administrators who configure, administer, and monitor Cisco Finesse.

For information about administering Finesse within a Unified Contact Center Express environment, see *Cisco Unified Contact Center Express Administration Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-express/products-maintenance-guides-list.html>.

Related Documents

Document or resource	Link
<i>Cisco Finesse Documentation Guide</i>	https://www.cisco.com/en/US/partner/products/ps11324/products_documentation_roadmaps_list.html
<i>Configure SNMP Trap in Cisco Finesse</i>	https://www.cisco.com/c/en/us/support/docs/contact-center/finesse/214387-configure-snmp-trap-in-cisco-finesse.html
Cisco.com site for Finesse documentation	https://www.cisco.com/en/US/partner/products/ps11324/tsd_products_support_series_home.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	<p>Boldface font is used to indicate commands, such as user entries, keys, buttons, folder names, and submenu names.</p> <p>For example:</p> <ul style="list-style-type: none">• Choose Edit > Find.• Click Finish.

Convention	Description
<i>italic font</i>	Italic font is used to indicate the following: <ul style="list-style-type: none">• To introduce a new term. Example: A <i>skill group</i> is a collection of agents who share similar skills.• A syntax value that the user must replace. Example: IF (<i>condition, true-value, false-value</i>)• A book title. Example: See the <i>Cisco Unified Contact Center Enterprise Installation and Upgrade Guide</i>.
window font	Window font, such as Courier, is used for the following: <ul style="list-style-type: none">• Text as it appears in code or that the window displays. Example: <code><html><title>Cisco Systems, Inc. </title></html></code>
< >	Angle brackets are used to indicate the following: <ul style="list-style-type: none">• 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.



CHAPTER 1

Getting Started

- [User Accounts](#), on page 1
- [Administration Tools](#), on page 1
- [Certificate Management](#), on page 5
- [QoS Settings](#), on page 12
- [Localization](#), on page 12

User Accounts

Credentials for the following user accounts are defined during Cisco Finesse installation:

- **Administrator User account:** Use this account to access the CLI and Cisco Unified Communications Operating System Administration.
- **Application User account:** Use this account to access the Cisco Finesse administration console.

Administration Tools

Cisco Finesse Administration Console

The Cisco Finesse administration console is a web-based interface used to configure system settings in Cisco Finesse. The administration console contains tabs to click and access the various administration features. The tab names and the associated tasks are:

- **Settings:** Administration & Data server, Configure CTI server, Cluster Settings, IP Phone Agent Settings, and Desktop Chat server.
- **Call Variables Layout:** Manage the call and ECC variables that appear on the agent desktop call control gadget, team performance gadget, and call popover.
- **Desktop Layout:** Make changes to the default desktop layout for agents and supervisors.
- **Phone Books:** Add, edit, or delete phone books or phone book contacts.
- **Reasons:** Add, edit, or delete Not Ready reason codes, Sign Out reason codes, or Wrap-Up reasons (Reason Codes are disabled for Packaged CCE deployments).

- **Team Resources:** Assign desktop layouts, phone books, reason codes, and wrap-up reasons to specific teams.
- **Workflows:** Create and manage workflows and workflow actions.
- **Connected Agents:** View, search, sort and refresh the list of agents signed in to Cisco Finesse.

The features you configure in the administration console are case-sensitive. For example, you can create two workflows named WORKFLOW and workflow; or two phone books named BOOK and book.



Note Finesse administration tasks are performed only on the primary Finesse server.

Response Caching

To reflect the changes made to system settings in Finesse desktop, the administrator must clear server cache using the CLI **utils webproxy cache clear rest**. Ensure that the agent browser is refreshed for the system settings changes to take effect.

For more information of REST API Response Caching, see *Cisco Finesse Web Services Developer Guide* at <https://developer.cisco.com/docs/finesse/#!rest-api-dev-guide>.

Sign In to Cisco Finesse Administration Console

You can access the Cisco Finesse administration console only through HTTPS.

When you sign in to Cisco Finesse, always use the fully qualified domain name (FQDN) of the Cisco Finesse server in the URL.

Step 1 Direct your browser to `https://<FQDN>:8445/cfadmin`, where *FQDN* is the fully qualified domain name of your primary Finesse server.

Note Ensure that the self-signed certificate that is provided with Cisco Finesse uses the hostname of the server as the Common Name for the certificate by default. The hostname in the URL must match the Common Name on the certificate to avoid an address mismatch error.

Step 2 The first time when you access the administration console using HTTPS, you are prompted to trust the self-signed certificate provided with Finesse. The following table describes the steps for each supported browser.

Note If you are using HTTPS but have installed a CA Certificate, you can skip this step. For more information about installing a CA Certificate, see *Cisco Finesse Installation and Upgrade Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-installation-guides-list.html>.

Option	Description
Internet Explorer:	<ol style="list-style-type: none"> A page appears that states this site is untrusted. Click More information > Go on to the webpage.
Firefox:	<ol style="list-style-type: none"> A page appears that states this connection is untrusted. Click I Understand the Risks, and then click Add Exception.

Option	Description
	<p>c. In the Add Security Exception dialog box, ensure that the check box is Permanently store this exception checked.</p> <p>d. Click Confirm Security Exception.</p>
Chrome and Edge Chromium (Microsoft Edge):	<p>a. A page appears that states this connection is not private.</p> <p>b. In Chrome, click Advanced > Proceed to <Hostname> (unsafe)</p> <p>c. In Microsoft Edge, click Advanced > Continue to <Hostname> (unsafe)</p>

Step 3 On the Sign In page, in the ID field, enter the Application User ID that was used during the installation.

Step 4 In the Password field, enter the Application User password that was used during the installation.

Step 5 Click **Sign In**.

A successful sign-in launches an interface with defined administration gadgets and a Sign Out link.



Note After 30 minutes of inactivity, Cisco Finesse automatically signs you out of the administration console and you must sign in again.

Sign In Using IPv6

If you sign in to the Finesse Administration Console using an IPv6-only client, include HTTPS port in the sign in URL in Step 1 of the preceding procedure.

- For HTTPS access, enter:
`https://<FQDN>:8445/cfadmin`

The remaining steps of the sign in procedure remain the same for IPv6.

Account Locked after Five Failed Sign in Attempts

If an administrator tries to sign in to the Finesse administrator console (or diagnostic portal) with the wrong password five times consecutively, Finesse blocks access to that user account for 30 minutes. For security reasons, Finesse does not alert the user that their account is locked. They must wait 30 minutes and try again.

Similarly, if agents or supervisors sign in to the desktop five times consecutively with the wrong password, Finesse blocks access to that user account. However, in this case, the lockout period is 5 minutes. This restriction also applies when agents and supervisors sign in using the mobile agent or Finesse IP Phone Agent (IPPA).



Note When an agent or supervisor account is locked, subsequent attempts to sign in, even with correct credentials, reset the lockout period to 5 minutes again. For example, if a locked user tries to sign in again after only 4 minutes, the lockout period is reset and the user must wait another 5 minutes.

To view the list of locked users, use the **utils finesse locked_out_users list** CLI command.

CLI

The CLI provides a set of commands applicable to the Operating System and to Cisco Finesse. These commands allow basic maintenance and failure recovery, and enable system administration.

You can access the CLI on the primary Finesse server with a monitor and keyboard at the server console or by Secure Shell (SSH). Use the credentials for the Administrator User account to access the CLI.

Cisco Unified Operating System Administration

This interface is web-based and is used to perform the following system administration functions:

- **Show:** View information on cluster nodes, hardware status, network configuration, installed software, system status, and IP preferences.
- **Settings:** Display and change IP settings, network time protocol (NTP) settings, SMTP settings, time, and version.



Important You cannot change the IP address of a Finesse server after it is installed.

- **Security:** Manage certificates and set up and manage IPSec policies.
- **Software Upgrades:** Perform and upgrade or revert to a previous version.
- **Services:** Use the Ping and Remote Support features.

Sign In to Cisco Unified Operating System Administration

Step 1 Direct your browser to `http://host or IP address/cmplatform`, where *host or IP address* is the hostname or IP address of your server.

Step 2 Sign in with the username and password for the Administrator User account.

Note After you sign in, you can access other Unified Communications Solutions tools from the Navigation drop-down list.

Configure Custom Logon Message Using OS Administration

Cisco Finesse administrator can configure a custom logon messages, which is displayed displayed when the users **Sign In** to the desktop.

To configure the custom logon message using Cisco Unified OS Administration console, run the following steps:



Note The custom message can be stored in a text file and then the file can be uploaded to the platform. By default, there is no custom message configured. However, it is not mandatory to configure the message.

-
- Step 1** Log in to **Cisco Unified OS Administration**.
 - Step 2** Navigate to **Software Upgrades > Custom Logon Message**.
 - Step 3** Click **Choose File**, to upload the customized logon message file
 - Step 4** Select the file and click **Open**.
 - Step 5** Click **Upload File**.
-

Configure Custom Logon Messages Using CLIs

The custom logon message can be configured using CLI commands. For more information about using CLI to configure custom logon message for desktop users see the *Security Banner Message for Desktop Users* section in [Desktop Properties, on page 208](#).

Certificate Management

Finesse provides a self-signed certificate that use or provide a CA certificate. You can obtain a CA certificate from a third-party vendor or produce one internal to your organization.

Finesse does not support wildcard certificates. After you upload a root certificate signed by a certificate authority (CA), the self-signed certificates are overwritten.

If you use the Finesse self-signed certificate, agents must accept the security certificates the first time they sign in to the desktop. If you use a CA certificate, you can accept it for the browser on each client or deploy a root certificate using group policies.



Note If there is a mismatch between the server hostname and the certificate hostname, a certificate address mismatch warning message is displayed in IE. The certificate must be regenerated so that the hostname matches the server hostname before importing to Finesse. If there is a valid reason for the mismatch, uncheck the **Warn about certificate address mismatch** checkbox from **Tools > Internet Options > Advanced > Security** to allow the certificate to be accepted.

Server-Side Certificate Management

By default, Finesse comes with self-signed certificates. If you use these certificates, agents must complete a procedure to accept the certificates the first time they sign in. To simplify the agent experience, obtain and upload a CA certificate or produce your certificate internally.

Obtain and Upload CA Certificate



Note This procedure only applies if you are using HTTPS and is optional. If you are using HTTPS, you can choose to either obtain and upload a CA certificate or use the self-signed certificate provided with Finesse.

To eliminate browser security warnings each time you sign in, obtain an application and root certificate signed by a CA. Use the Certificate Management utility from Cisco Unified Operating System Administration.

To open Cisco Unified Operating System Administration in your browser, enter:

`https://FQDN of primary Finesse server:8443/cmplatform`

Sign in using the username and password for the Application User account created during Finesse installation.



Note You can find detailed explanations in the Security topics of the *Cisco Unified Operating System Administration Online Help*.

Step 1 Generate a CSR.

- a) Click **Security > Certificate Management > Generate CSR**.
- b) From the Certificate Name drop-down list, choose **tomcat** and click **Generate CSR**.

Note To avoid certificate exception warnings, you must access the servers using the FQDN name.

Step 2 Download the CSR.

- a) Select **Security > Certificate Management > Download CSR**.
- b) From the Certificate Name drop-down list, choose **tomcat** and click **Generate CSR**.

Step 3 Generate and download a CSR for the secondary Unified CCX server.

To open Cisco Unified Operating System Administration for the secondary server in your browser, enter:

`https://FQDN of secondary Finesse server:8443/cmplatform`

Step 4 Use the CSRs to obtain the CA root certificate, intermediate certificate, and signed application certificate from the Certificate Authority.

Note To set up the certificate chain, you must upload the certificates in the order described in the following steps.

Step 5 When you receive the certificates, click **Security > Certificate Management > Upload Certificate**.

Step 6 Upload the root certificate.

- a) From the **Certificate Purpose** drop-down list, select **tomcat-trust**.
- b) In the **Upload File** field, click **Browse** and browse to the root certificate file.
- c) Click **Upload File**.

Step 7 Upload the intermediate certificate.

- a) From the **Certificate Purpose** drop-down list, choose **tomcat-trust**.
- b) In the **Upload File** field, click **Browse** and browse to the intermediate certificate file.

c) Click **Upload File**.

Step 8 Upload the application certificate.

- a) From the **Certificate Purpose** drop-down list, choose **tomcat**.
- b) In the **Upload File** field, click **Browse** and browse to the application certificate file.
- c) Click **Upload File**.

Step 9 After the upload is complete, sign out from the Platform Admin page of Finesse.

Step 10 Access the CLI on the primary Finesse server.

Step 11 Enter the command **utils service restart Cisco Finesse Notification Service** to restart the Cisco Finesse Notification service.

Step 12 Enter the command **utils service restart Cisco Finesse Tomcat** to restart the Cisco Finesse Tomcat service.

Step 13 Upload the application certificate to the secondary Finesse server.

The root and the intermediate certificates uploaded to the primary server are replicated to the secondary server.

Step 14 Access the CLI on the secondary Finesse server and restart the Cisco Finesse Notification Service and the Cisco Finesse Tomcat Service.

Produce Certificate Internally

Set up Microsoft Certificate Server for Windows Server 2012 R2

A prerequisite of this procedure is that your deployment includes a Windows Server 2012 R2 (Standard) Active Directory server. Perform the following steps to add the Active Directory Certificate Services role on the Windows Server 2012 R2 (Standard) domain controller.

Before you begin

Microsoft .Net Framework 3.5 Service Pack 1 must be installed. See the Windows Server 2012 documentation for instructions.

Step 1 In Windows, open the **Server Manager**.

Step 2 In the **Quick Start** window, click **Add Roles and Features**.

Step 3 In the **Set Installation Type** tab, choose **Role-based or feature-based installation** and click **Next**.

Step 4 In the **Server Selection** tab, choose the destination server and click **Next**.

Step 5 In the **Server Roles** tab, check the **Active Directory Certificate Services** box and click **Add Features** in the pop-up window.

Step 6 In the **Features** and **AD CS** tabs, click **Next** to accept default values.

Step 7 In the **Role Services** tab, verify that the **Certification Authority** box is checked and click **Next**.

Step 8 In the **Confirmation** tab, click **Install**.

Step 9 After the installation is complete, click the **Configure Active Directory Certificate Service on the destination server** link.

Step 10 Verify the credentials (for the domain Administrator user) and click **Next**.

Step 11 In the **Role Services** tab, check the **Certification Authority** box and click **Next**.

Step 12 In the **Setup Type** tab, choose **Enterprise CA** and click **Next**.

- Step 13** In the **CA Type** tab, choose **Root CA** and click **Next**.
- Step 14** In the **Private Key**, **Cryptography**, **CA Name**, **Validity Period**, and **Certificate Database** tabs, click **Next** to accept default values.
- Step 15** Review the information in the **Confirmation** tab and click **Configure**.

Download CA certificate

A prerequisite of this procedure is that you are using the Windows Certificate Services. Perform the following steps to retrieve the root CA certificate from the certificate authority. After you retrieve the root certificate, each user must install it in the browser used to access Finesse.

- Step 1** On the Windows domain controller, run the CLI command `certutil -ca.cert ca_name.cer`, in which `ca_name` is the name of your certificate.
- Step 2** Save the file. Note where you saved the file so you can retrieve it later.

Client-Side Certificate Acceptance

There are procedures that agents must perform to accept certificates the first time they sign in. The procedure type depends on the method you choose to manage certificates and the browser used by the agents.

Client Requirements

For more information on client requirements, see *Compatibility Information* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-device-support-tables-list.html>.



Note Finesse Desktop client machines should be time synchronized with a reliable NTP server for the correct updates to the Duration fields within Live data reports.

Deploy Root Certificate for Internet Explorer

In environments where group policies are enforced via the Active Directory domain, the root certificate can be added automatically to each user's Internet Explorer. Adding the certificate automatically simplifies user configuration requirements.



Note To avoid certificate warnings, each user must use the FQDN of the Finesse server to access the desktop.

- Step 1** On the Windows domain controller, navigate to **Administrative Tools > Group Policy Management**.

Note Users who have strict Group Policy defined on the Finesse Agent Desktop have to disable **Cross Document Messaging** from **Group Policy Management** to ensure proper functioning of Finesse on Internet Explorer 11.

- Step 2** Right-click Default Domain Policy and select **Edit**.
- Step 3** In the Group Policy Management Console, click **Computer Configuration > Policies > Window Settings > Security Settings > Public Key Policies**.
- Step 4** Right-click Trusted Root Certification Authorities and select **Import**.
- Step 5** Import the *ca_name.cer* file.
- Step 6** Go to **Computer Configuration > Policies > Windows Settings > Security Settings > Public Key Policies > Certificate Services Client - Auto-Enrollment**.
- Step 7** From the Configuration Model list, select **Enabled**.
- Step 8** Sign in as a user on a computer that is part of the domain and open Internet Explorer.
- Step 9** If the user does not have the certificate, run the command **gpupdate.exe /target:computer /force** on the user's computer.
-

Set Up CA Certificate for Internet Explorer

After obtaining and uploading the CA certificates, the certificate must be automatically installed via group policy or all the users must accept the certificate.

In environments where users do not log in directly to a domain or where group policies are not utilized, every Internet Explorer user in the system must perform the following steps one time to accept the certificate:

- Step 1** In Windows Explorer, double-click the *ca_name.cer* file and then click **Open**.
- Note** Here the *ca_name* is the name of your certificate.
- Step 2** In the **Certificate Import Wizard**, select **Current User**.
- Step 3** Click **Install Certificate > Next > Place all certificates in the following store**.
- Step 4** Click **Browse** and choose **Trusted Root Certification Authorities**.
- Step 5** Click **OK > Next > Finish**.
- Step 6** Click **Yes** on the install a certificate from a CA prompt.
- Step 7** To verify that the certificate was installed, from the browser menu on IE, choose **Tools > Internet Options**.
- Step 8** In the **Content** tab, click **Certificates**.
- Step 9** In the **Trusted Root Certification Authorities** tab, ensure that the new certificate appears in the list.
- Step 10** Restart the browser for the certificate installation to take effect.
- Note** If you are using Internet Explorer 11, you may receive a prompt to accept the certificate even if it is signed by a private CA.
-

Set Up CA Certificate for Firefox Browser

Every Firefox user in the system must perform the following steps once to accept the certificate:



Note To avoid certificate warnings, each user must use the FQDN of the Unified CCX server to access the desktop.

-
- Step 1** From the Firefox browser menu, choose **Options**.
 - Step 2** Go to **Privacy and Security** tab.
 - Step 3** Under Certificates section, click **View Certificates**.
 - Step 4** Select **Authorities**.
 - Step 5** Click **Import** and browse to the *ca_name.cer* file.
 - Note** Here the *ca_name* is the name of your certificate.
 - Step 6** Check the **Validate Identical Certificates** check box.
 - Step 7** Restart the browser for the certificate to install.
-

Set Up CA Certificate for Chrome and Edge Chromium (Microsoft Edge) Browsers

- Step 1** In the browser, go to **Settings**.
 - Step 2** In the Chrome browser, select **Advanced Settings > Privacy and Security**, click **Manage Certificates**.
 - Step 3** In the Microsoft Edge browser, select **Privacy, search, and services**. Under **Security**, click **Manage Certificates**.
 - Step 4** Click **Trusted Root Certification Authorities** tab.
 - Step 5** Click **Import** and browse to the *ca_name.cer* file.
In the **Trusted Root Certification Authorities** tab, ensure that the new certificate appears in the list.
 - Step 6** Restart the browser for the certificate to install.
-

Manage Expired CA Certificates

If you receive a certificate expiry alert, it means that the validity of your CA certificate is about to expire. You can delete the certificate after expiry. If you use any CA to sign your certificates, you must upload the new certificates to ensure your system remains operational. Some CA certificates that are shipped with the platform do not require to be uploaded and can be deleted after expiry. For the complete list of CAs that can be safely deleted after expiry, refer to the *Manage Expired CA Certificates* section in the [Cisco Unified Contact Center Express Administration and Operations Guide](#).

Trust Self-Signed Certificate

Trust the self-signed certificate provided by Finesse to eliminate browser warnings each time you sign in to the administration console or agent desktop.

If you have uploaded a CA certificate, you can skip this procedure.

In your browser, enter the URL for the administration console (<https://FQDN of the primary Finesse server/cfadmin>) or the agent desktop (<https://FQDN of the primary Finesse server/desktop>).

Add Certificate for HTTPS Gadget

Add a certificate for a secure HTTP (HTTPS) gadget to load the gadget on the Finesse desktop and successfully perform HTTPS requests to the Finesse server.

This process allows HTTPS communication between the Finesse gadget container and the third-party gadget site for loading the gadget and performing any API calls the gadget makes to the third-party server.



Note A gadget that loads using HTTPS may still use HTTP communication between that gadget and the application server where it resides. If all traffic must be secure, the gadget developer must ensure that HTTPS is used to make API calls to the application server.

The certificate must be signed with a common name. The gadget URL in the desktop layout must use the same name (whether it uses an IP address or an FQDN) as the name with which the certificate is signed. If the certificate name and the name in the gadget URL don't match, the connection isn't trusted, and the gadget doesn't load.

To find the certificate name, enter the gadget URL in your browser. Click the lock icon in the address bar and then click View Details. Look for the common name field.

The Finesse host must be able to resolve this name using the DNS host entered during the installation. To verify that Finesse can resolve the name, run the CLI `utils network ping <hostname>` command.

Step 1

Download the certificate from the third-party host running a Cisco-provided solution.

- a) Sign in to Cisco Unified Operating System Administration on the third-party gadget host (`https://FQDN:8443/cmplatform`, where *FQDN* is the fully qualified domain name of the third-party gadget host).
- b) Click **Security > Certificate Management**.
The **Certificate List** page appears.
- c) In the **Find Certificate List where** drop-down list, select **Common Name/Common Name SerialNumber** and in the next drop-down list, select **contains**.
- d) In the **Select item or enter search text** field, enter hostname or the domain of the host and click **Find**.
All the certificates that have the hostname or the domain of the host that was entered as part of the **Common Name/Common Name SerialNumber** are listed in a tabular format.

Note You can also click **Find** without any search criteria to list all the available certificates. From the list of certificates, identify the required certificates based on the following:

- The **Certificate** column indicates the certificate purpose. The certificates listed as the **tomcat-trust** are used for establishing the webserver(tomcat) identity.
 - The **Key Type** column indicates the algorithm used to generate the digital signature that is included in the certificate. For example **RSA**, **EC** (represents ECDSA).
 - The **Usage** column indicates the certificate type and if the certificate is used to establish trust or is the host certificate. The term **Identity** indicates that the certificate is used for establishing the webserver(tomcat) identity.
- e) Click the hyperlinked **Common Name/Common Name SerialNumber** that you want to download.
The **Certificate Details** pop-up window appears.
 - f) Click **Download .PEM File** or **Download .DER File** and save the file in the required location.

- Step 2** Upload the certificate to the primary Finesse server.
- Sign in to Cisco Unified Operating System Administration on the primary Finesse server (<https://FQDN:8443/cmplatform>, where *FQDN* is the fully qualified domain name of the Finesse server).
 - Click **Security > Certificate Management**.
 - Click **Upload Certificate**.
 - From the Certificate Name drop-down list, select **tomcat-trust**.
 - Click **Browse** and navigate to the tomcat.pem file that you downloaded in the previous step.
 - Click **Upload File**.
- Step 3** Restart Cisco Finesse Tomcat on the primary Finesse server.
- Step 4** After synchronization is complete, restart the Cisco Finesse Tomcat on the secondary Finesse server.

QoS Settings

The Cisco Finesse application currently does not support configuration of QoS settings in network traffic. The QoS classification and marking of traffic should be done at the Switch or Router level for signaling traffic to be prioritized, especially if agents are across WAN.

Localization

Cisco Finesse supports localization for the Finesse agent desktop when Finesse is deployed with Unified CCE. Use the Cisco Option Package (COP) file installation to install the languages you require for your agents and supervisors.

Finesse is installed with US English. If you do not require other languages for your agents and supervisors, you do not need to install the COP files.



Note An appropriate language needs to be selected before login on the desktop. If not, English will be the default language. You cannot uninstall a language pack after it is installed.

Table 2: Supported Languages for Desktop User Interface

Language	Locale File	Language	Locale File
Bulgarian	Bg_BG	Portuguese	pt_BR
Catalan	Ca_ES	Romanian	Ro_RO
Czech	Cs_CZ	Spanish	es_ES
Croatian	Hr_HR	Swedish	sv_SE
Danish	da_DK	Slovak	Sk_SK
Dutch	nl_NL	Slovenian	Sl_SI

Language	Locale File	Language	Locale File
English	en_US	Serbian	Sr_RS
Finnish	fi_FI	Japanese	ja_JP
French	fr_FR	Chinese (simplified)	zh_CN
German	de_DE	Chinese (traditional)	zh_TW
Hungarian	Hu_HU	Korean	ko_KR
Italian	it_IT	Polish	pl_PL
Norwegian	nb_NO	Russian	ru_RU
Turkish	tr_TR		

After you install the COP files, agents and supervisors can set the language on their desktops in the following ways:

- Choose a language from the language selector drop-down list on the sign-in page.
- Change their browser preferred language.
- Pass the locale as part of the agent desktop URL (for example, an agent who wants to use French can enter the following URL: https://hostname_of_Finesse_server/desktop?locale=fr_FR)

The following items are localized on the desktop:

- labels for field names, buttons, and drop-down lists
- prompts
- messages
- tool tips
- page titles
- gadget tab names (Finesse gadgets only)

Configuration data defined using the Finesse administration console (such as Not Ready and Sign Out reason code labels, Wrap-Up reason labels, and phonebook entries) do not depend on the locale chosen for the desktop. For example, if you have defined a Not Ready reason code with a Chinese label, the label appears on the desktop in Chinese, regardless of the language the agent chooses when signing in.



Note If you do not install the language COP files (you use English only for the desktop), you can still use Unicode characters for Finesse data such as reason codes, wrap-up reasons, and phonebook entries. For example, if you define a reason code using Chinese characters, it appears in Chinese on an English-only desktop.

Call variables and ECC variables are Unicode enabled and independent of the desktop locale. However, these Unicode characters may not display correctly in the Unified CCE Script Editor, which can cause problems

with Unified CCE scripts (especially if Unified CCE is installed on a Original Language Windows Operating System (such as Chinese, Japanese, or Russian).

Agent first and last names and team names that use Latin-1 characters appear on the desktop as they are defined in the Unified CCE database. However, if Unified CCE is installed on a Original Language Windows Operating System and the names contain non-Latin-1 characters, the names displayed on the desktop will appear corrupted.

Finesse does not support the following for localization:

- Finesse administration console
- Tab labels for third-party gadgets deployed in the Finesse gadget container



Note You can define the tab labels for third-party gadgets in the Finesse layout XML file. These labels are hard-coded and are independent of the locale chosen on the desktop. You can only define one label for a tab. You cannot define multiple labels for a tab using different languages.

- Agent usernames and team names that consist of characters other than Latin-1



Note Locale-based searching and sorting may not work as expected.



CHAPTER 2

Manage System Settings

- [Contact Center Enterprise Administration and Data Server Settings](#), on page 15
- [Contact Center Enterprise CTI Server Settings](#), on page 17
- [Cluster Settings](#), on page 21
- [Desktop Chat Server Settings](#), on page 22
- [Cloud Connect Server Settings](#), on page 23
- [Keyboard Shortcuts](#), on page 24
- [Device Selection for Shared ACD Line](#), on page 25

Contact Center Enterprise Administration and Data Server Settings

Use the Unified CCE Administration & Data Server Settings gadget to configure the database settings. These settings are required to enable authentication for Finesse agents and supervisors.



Note Primary Administration & Data Server is configured on Side A and Secondary Administration & Data Server is configured on Side B. Make sure Cisco Finesse server on both sides connect to Primary Administration & Data Server on side A and fall back to Secondary Administration & Data Server on side B only when Primary Administration & Data Server goes down.

After you change and save any value on the Contact Center Enterprise Administration & Data Server Settings gadget, restart the Cisco Finesse Tomcat Service on the primary and secondary Finesse server. If you restart the Cisco Finesse Tomcat Service, agents must sign out and sign in again. To avoid this, you can make Contact Center Enterprise Administration & Data Server settings changes and restart the Cisco Finesse Tomcat service during hours when agents are not signed in to the Cisco Finesse desktop.

The following table describes the fields on the Unified CCE Administration & Data Server Settings gadget:

Table 3: Field Descriptions

Field	Description
Primary Host/IP Address	The hostname or IP address of the Unified CCE Administration & Data Server.

Field	Description
Backup Host/IP Address	(Optional) The hostname or IP address of the backup Unified CCE Administration & Data Server.
Database Port	The port of the Unified CCE Administration & Data Server. The default value is 1433. Note Cisco Finesse expects the primary and backup Administration & Data Server ports to be the same, hence the Finesse administration console exposes one port field. You must ensure that the port is the same for the primary and backup Administration & Data Servers.
AW Database Name	The name of the AW Database (AWDB). For example, <i>ucceinstance_awdb</i> .
Domain	(Optional) The domain name of the AWDB. For example, <i>cisco.com</i> .
Username	The username required to sign in to the AWDB. Note If you specify a domain, this user refers to the Administrator Domain user that the AWDB uses to synchronize with the logger. In which case, the AWDB server must use Windows authentication and the configured username must be a domain user. If you do not specify a domain, this user must be an SQL user.
Password	The password required to sign in to the AWDB.

For more information about these settings, see the [Administration Guide for Cisco Unified Contact Center Enterprise](#) and the [Staging Guide for Cisco Unified ICM/Contact Center Enterprise](#).

Actions on the Unified CCE Administration & Data Server Settings gadget:

- **Save:** Saves your configuration changes
- **Revert:** Retrieves the most recently saved enterprise database settings

When you update any of the following fields and click Save, Cisco Finesse attempts to connect to the AWDB:

- Primary Host/IP Address
- Backup Host/IP Address
- Database Port

- AW Database Name

If Cisco Finesse cannot connect to the AWDB, an error message appears and you are asked if you still want to save. If you click **Yes**, the settings are saved. If you click **No**, the settings are not saved. You can change the settings and try again or click **Revert** to retrieve the previously saved settings.

When you update the Username or Password fields and click **Save**, Cisco Finesse attempts to authenticate against the AWDB. If authentication fails, an error message appears and you are asked if you still want to save. Click **Yes** to save the settings or click **No** to change the settings. Click **Revert** to retrieve the previously saved settings.



Note Finesse will not come into service in case of AWDB errors when connecting Cisco Finesse 11.5(1) and higher versions to Unified CCE 11.5(1) and higher versions.

Configure Contact Center Enterprise Administration and Data Server Settings

Configure the Unified CCE Administration & Data Server settings to enable authentication for Finesse agents and supervisors.

-
- Step 1** If you are not already signed in, sign in to the administration console.
- Step 2** In the Unified CCE Administration & Data Server Settings area, enter the Administration & Data Server settings as described in the preceding table. For more information, see [Table 3: Field Descriptions, on page 15](#). Refer to your configuration worksheet if necessary.
- Step 3** Click **Save**.
-

What to do next

The CTI test functionality documented in the *Configure Unified CCE CTI Server Settings* topic depends on AWDB connectivity to determine the CTI version. Or else, the test will not go through.

Contact Center Enterprise CTI Server Settings

Use the Contact Center Enterprise CTI Server Settings gadget to configure the A and B Side CTI servers.

All fields on this tab are populated with default system values or with values an administrator has previously entered. Change values to reflect your environment and preferences.

For configuring secure connection select the Enable SSL encryption check box.

Test the CTI connection for given configuration using the **Test Connection** button.



Note After you make any changes to the values on the Contact Center Enterprise CTI Server Settings gadget, you must restart all the nodes of Cisco Finesse Tomcat. To make changes to other settings (such as Contact Center Enterprise Administration & Data Server settings), you can make those changes and then restart Cisco Finesse Tomcat.

If you restart Cisco Finesse Tomcat, agents must sign out and sign in again. As a best practice, make changes to CTI server settings and restart the Cisco Finesse Tomcat Service during hours when agents are not signed in to the Finesse desktop.

The secure encryption and Test Connection functionality is supported only from Unified CCE 12.5.



Note Although the B Side Host/IP Address and B Side Port fields are not shown as required, A and B Side CTI servers are mandatory for a production deployment of Unified CCE and Cisco Finesse.

The following table describes the fields on the Contact Center Enterprise CTI Server Settings gadget:

Field	Explanation
A Side Host/IP Address	The hostname or IP address of the A Side CTI server. This field is required. This value is typically the IP address of the Peripheral Gateway (PG). The CTI server runs on the PG.
A Side Port	The value of this field must match the port configured during the setup of the A Side CTI server. This field is required and accepts values between 1 and 65535. You can find this value using the Unified CCE Diagnostic Framework Portico tool on the PG box. For more information about Diagnostic Framework Portico, see the <i>Serviceability Guide for Cisco Unified ICM/Contact Center Enterprise</i> . The default value is 42027.
Peripheral ID	The ID of the Agent PG Routing Client (PIM). The Agent PG Peripheral ID should be configured to the same value for the A and B Side CTI server. This field is required and accepts values between 1 and 32767. The default value is 5000.
B Side Host/IP Address	The hostname or IP address of the B Side CTI server.

Field	Explanation
B Side Port	The value of this field must match the port configured during the setup of the B Side CTI server. This field accepts values between 1 and 65535.
Enable SSL encryption	Check this box to enable secure encryption.

Actions on the Contact Center Enterprise CTI Server Settings gadget:

- **Save:** Saves your configuration changes.
- **Revert:** Retrieves the most recently saved server settings.
- **Test Connection:** Tests the CTI connection.

CTI Test Connection

When you click **Test Connection**:

1. Validation is done to check if the CTI server is in maintenance mode.



Note If the CTI server is in maintenance mode, Finesse does not allow test connection. For more information, see section *Unified CCE Agent PG Graceful Shutdown* in *Administration Guide for Cisco Unified Contact Center Enterprise, Release 12.6(1)* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-maintenance-guides-list.html>.

2. Input validation is done on the request attributes.
Host/IP Address must not be empty. Port and Peripheral IDs must be within the valid range.
3. Validation is done to check if the provided Host/IP is resolved by Finesse box.
4. Validation is done to check if AW Database is reachable and if a valid path ID is configured for the provided Peripheral ID.
5. Socket connection is established to the provided Host/IP and port. The connection might fail if there is no route to the provided IP. If SSL encryption box is checked, this step also checks for successful TLS handshake. For TLS handshake to be successful, mutual trust has to be established between Finesse and CTI server.

For information on how to establish trust between Finesse and CTI server, see *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>
6. After successful socket connection, a CTI initialization request is sent to check if the provided host is a CTI host.

If the CTI response is a success for the CTI initialization request and peripheral provided is configured with Unified CCE, it is confirmed to be a CTI host.
7. CTI connection is closed by sending a CTI session close request.



Note If **Test Connection** is successful for Side A or B of the CTI cluster and the other side fails, it is a valid configuration as CTI server works in active-passive mode and connects to the active node. Inactive CTI node will refuse connection on the CTI port. However, Administrator has to ensure that the failed side also has a valid entry for CTI host and port field. System cannot verify this due to server restrictions.

If **Test Connection** is successful on Side A and B of the CTI cluster, then there is an error in the system configuration. Verify that the Side A and B of the CTI node have valid entries for port and host.

Test connection API success result does not guarantee peripheral to be online. It only validates if the peripheral provided is configured with Unified CCE.

Test connection API with insecure connection parameter will function as intended for earlier versions of Unified CCE deployments.

Configure Contact Center Enterprise CTI Server Settings

Access the administration console on the primary Finesse server to configure the A and B Side CTI servers.



Note After you restart Finesse, it can take approximately 6 minutes for all server-related services to restart. Therefore, wait for 6 minutes before you attempt to access the Finesse administration console.



Note If you are using HTTPS, the first time you access the administration console, you see a browser security warning. To eliminate browser security warnings each time you sign in, trust the self-signed certificate provided with Finesse or obtain and upload a CA certificate.

Step 1 Sign in to the administration console on the primary Finesse server:

`https://FQDN of Finesse server/cfadmin`

Step 2 Sign in with the Application User credentials defined during installation.

Step 3 In the Contact Center Enterprise CTI Server Settings area, enter the CTI server settings as described in the following table. Refer to your configuration worksheet if necessary.

Field	Description
A Side Host/IP Address	Enter the hostname or IP address of the A Side CTI server. This value is typically the IP address of the Peripheral Gateway (PG). The CTI server runs on the PG.
A Side Port	Enter the port number of the A Side CTI server. The value of this field must match the port configured during the setup of the A Side CTI server.

Field	Description
Peripheral ID	Enter the ID of the Agent PG Routing Client (PIM). The Agent PG Peripheral ID should be configured to the same value for the A and B Side CTI servers.
B Side Host/IP Address	Enter the hostname or IP address of the B Side CTI server.
B Side Port	Enter the port of the B Side CTI server. The value of this field must match the port configured during the setup of the B Side CTI server.

Step 4 Click **Save**.

Cluster Settings

Use the Cluster Settings gadget to configure a secondary Finesse server. The purpose of a secondary Finesse server is to handle all agent requests if the primary server goes down.

You must complete this configuration *before* you install the secondary Finesse server. For more information about installing a secondary Finesse server, see the *Cisco Finesse Installation and Upgrade Guide*.

The following table describes the fields on the Cluster Settings gadget:

Field	Explanation
Hostname	The hostname of the secondary Finesse server.

Actions on the Cluster Settings gadget:

- **Save:** Saves your configuration changes
- **Revert:** Retrieves the most recently saved cluster settings

Configure Cluster Settings

Configure the cluster settings for the secondary Finesse node. The secondary Finesse node handles agent requests if the primary server goes down.

Step 1 Sign in to the administration console with the Application User credentials.

Step 2 In the Cluster Settings area, in the Hostname field, enter the hostname of the secondary Finesse server.

Step 3 Click **Save**.

Desktop Chat Server Settings

Desktop Chat is an XMPP browser based chat, which is powered by Cisco Instant Messaging and Presence (IM&P) service. It provides presence and chat capabilities within the Unified CM platform. For more details, see *Configuration and Administration of the IM and Presence Service* at <https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/products-installation-and-configuration-guides-list.html>.

Desktop Chat connects to Cisco IM&P servers over port 5280 from the browser hosting the agent desktop. IM&P server visibility and port accessibility needs to be ensured if clients intend to use this feature. The Desktop Chat gadget configures the IM&P host BOSH URL's used by the desktop to communicate with the IM&P server over BOSH HTTP.

IM&P has a clustered design, where users are distributed across multiple nodes in the cluster. The Desktop Chat initially discovers the IM&P nodes that a user has configured, caches this information and communicates with the actual server for subsequent login, until the browser cache is cleared. To spread the initial discovery load, it is advisable to configure the nodes in a round robin fashion if the deployment has more than one Finesse cluster. For example, if there are 5 IM&P nodes configure Finesse cluster A with node 1 & 2, Finesse cluster B with nodes 3 & 4, and so on.

Node availability should be considered while configuring the IM&P URL. The secondary node will be available for discovery in scenarios where the first node is not reachable. The secondary node will be connected for discovery only if the primary node is unreachable.

For the URL to be configured, refer Cisco Unified Presence Administration service, in *System, Service Parameters*. Choose the required IM&P server, select Cisco XCP Web Connection Manager. The URL binding path is listed against the field *HTTP Binding Path*. The full URL to be configured in Finesse is `https://<hostname>:5280/URL-binding-path`.

Use the Desktop Chat Server Settings to configure chat settings for the Finesse desktop. The following table describes the fields on the Desktop Chat Server Settings gadget.

Field	Explanation
Primary Chat Server	Enter the IM&P primary server URL of Desktop Chat.
Secondary Chat Server	Enter the IM&P secondary server URL of Desktop Chat.

Actions on the Desktop Chat Server gadget:

- **Save:** Saves your configuration changes
- **Revert:** Retrieves the most recently saved server settings



Important For Desktop Chat to work without any issues, ensure the following services are running on IM&P:

- Cisco Presence Engine
- Cisco XCP Text Conference Manager
- Cisco XCP Web Connection Manager
- Cisco XCP Connection Manager
- Cisco XCP Directory Service
- Cisco XCP Authentication Service
- Cisco XCP File Transfer Manager



Note Desktop Chat requires the Cisco IM and Presence certificates to be trusted. To start the Desktop Chat without experiencing an exception, you must add the certificate to the browser trust store, or configure IM and Presence with CA-signed certificate, or push self-signed certificate through group policies in supported browsers. For more information on accepting certificates, see the *Accept Security Certificates* section, in the *Common Tasks* chapter of *Cisco Finesse Agent and Supervisor Desktop User Guide for Cisco Unified Contact Center Express* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-express/products-user-guide-list.html>.

For more information on adding certificates to the browser trust store, see Certificate Management.



Note Desktop Chat is not supported with the unrestricted versions of IM&P.

Configure Desktop Chat Server Settings

- Step 1** Sign in to the administration console with the Application User credentials.
- Step 2** In the **Desktop Chat Server Settings** area, enter the IM&P primary and secondary server URL of the Desktop Chat.
- Step 3** Click **Save**.

Note Desktop Chat requires Cisco Unified Presence 12.5 and higher versions.

Cloud Connect Server Settings

Cloud Connect is a component that hosts services that allow customers to use cloud capabilities such as Cisco Webex Experience Management. The administrator can configure the Cloud Connect server settings in the Finesse administration console to contact the Cisco cloud services.

For more information, see the Cisco Webex Experience Management Integration section in *Cisco Unified Contact Center Enterprise Features Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html>.

The following table describes the fields on the Cloud Connect server settings gadget:

Field	Explanation
Username	(mandatory) The Cloud Connect administrator username required to sign in to Cloud Connect.
Password	(mandatory) The Cloud Connect administrator password required to sign in to Cloud Connect.
Publisher Address	(mandatory) The FQDN of the Cloud Connect publisher.
Subscriber Address	(optional) The FQDN of the Cloud Connect subscriber.

Actions on the Cloud Connect Server Settings gadget:

- **Save:** Saves your configuration changes.
- **Revert:** Retrieves the most recently saved Cloud Connect server settings.

For more information on importing Cloud Connect certificates, see the *Cloud Connect Certificates* section in *Cisco Finesse Installation and Upgrade Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-installation-guides-list.html>.

Configure Cloud Connect Server Settings

-
- Step 1** Sign in to the administration console on the primary Finesse server using the URL: `https://FQDN of Finesse server:8445/cfadmin`.
- Step 2** Select **Settings > Cloud Connect Server Settings**.
- Step 3** Enter the **Username**, **Password**, and **Publisher Address** of the Cloud Connect server.
(optional) Enter the **Subscriber Address** for multinode deployment.
- Step 4** Click **Save**.
-

Keyboard Shortcuts

Keyboard shortcuts provide an alternate way to perform a specific action on the Finesse agent and supervisor desktop. For more information, see *Access Keyboard Shortcuts* section in the *Cisco Finesse Agent and Supervisor Desktop User Guide*.

Keyboard Shortcut Conflicts

Keyboard shortcut conflicts occur if multiple gadgets use the same keyboard shortcut. This causes a particular key combination to be disabled until the conflict is resolved.

Keyboard shortcut conflicts at the page level can be resolved only by modifying the keyboard shortcuts at the gadget level. To modify the keyboard shortcuts at the gadget level, contact developer support services.

Keyboard shortcut conflict can occur in the following scenarios:

Conflict Scenario	Resolution
Conflicts can occur between keyboard shortcuts at the page level and gadget level.	This conflict cannot be resolved by the Finesse administrator.
Conflicts can occur when two gadgets have the same keyboard shortcut, and both are in the same tab.	Move one of the gadgets to another tab.
Conflicts can occur when there are multiple instances of the same gadget and focus is on the active tab*.	Move one of the gadgets to another tab.

* - Active tab refers to the tab that is currently being used.

The administrator can use the CLI command to disable the keyboard shortcuts for the Finesse agent and supervisor desktop. For more information on CLI commands, see *Desktop Properties*.



Note

- After deploying the third-party gadgets, the administrator must sign in as an agent and a supervisor to check if there are any keyboard shortcut conflicts and resolve them.
- The third-party gadget providers can use the keyboard shortcuts JavaScript library as a guideline to provide a consistent desktop user experience.
- The ECE (Enterprise Chat and Email) keyboard shortcuts are available only if ECE gadget is configured in the Unified CCE deployment. If it is not configured, the third-party gadget developers can use the ECE shortcut keys that are listed in **Keyboard Shortcuts List**. For more information, see the *Cisco Finesse Agent and Supervisor Desktop User Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-user-guide-list.html>.

Device Selection for Shared ACD Line

An administrator can configure agents or teams to use extension associated with multiple devices through agent desk settings. Enabling **ACD shared line usage** for an agent or supervisor allows them to select one of the end points (Desk Phone with EM, Desk Phone without EM, or Jabber, and so on) on the shared Automatic Call Distribution (ACD) lines as their active CTI device while signing in to Cisco Finesse.

For more information on agent desk settings, see the *Agent Administration Tasks* section in *Administration Guide for Cisco Unified Contact Center Enterprise* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-maintenance-guides-list.html>.

If you have enabled the device selection feature, when an agent or a supervisor sign in with the desired extension, the device selection screen displays a list of associated devices. For more information on device selection, see the *Select Active Device* section in *Cisco Finesse Agent and Supervisor Desktop User Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-user-guide-list.html>.



CHAPTER 3

Manage IP Address and Hostname

- [IP Address or Hostname or Domain Name Change](#), on page 27
- [Change IP Address or Hostname Task List](#), on page 27
- [Change IP Address Only](#), on page 32
- [Change Domain Name](#), on page 33
- [Post-Change Tasks and Verification](#), on page 34

IP Address or Hostname or Domain Name Change



Note Before you change the IP address or hostname or domain name, update the DNS server with a new host entry.

You can change the IP address or hostname or domain name of the Cisco Finesse cluster nodes in your deployment. These changes can be done for a variety of reasons such as:

- to resolve a duplicate IP address or hostname.
- to move the server from one segment to another.



Note Hostname supports 24 alphanumeric characters.

Change IP Address or Hostname Task List

The following table lists the tasks to perform before you change the IP address or hostname for Cisco Finesse cluster nodes.

Procedure

	Command or Action	Purpose
Step 1	System health checks before the IP address or hostname change	Perform system health checks before the IP address or hostname change.

	Command or Action	Purpose
Step 2	Change IP Address or Hostname using Unified Operating System GUI or Change IP Address or Hostname Using CLI .	Change IP address or hostname for the Cisco Finesse cluster node using either the Unified OS GUI or Command Line Interface (CLI).
Step 3	Post-Change Tasks and Verification .	Verify system health checks after the IP address or hostname change.

Pre-Change Tasks

Perform the following system health checks for the Cisco Finesse cluster nodes before you change the IP address or hostname or domain name.



Note If you do not receive the results that you expect when you perform these tasks, do not continue until you have resolved the issue.

Step 1 Check the active ServerDown alerts to ensure that all servers are active and available on the Cisco Finesse cluster nodes.

- Check by using RTMT—Select **Alert Central** from the tree hierarchy or choose **System > Tools > Alert > Alert Central** and verify the alerts.
- Check by using CLI—Enter **file search activelog syslog/CiscoSyslog ServerDown** command and verify the application event log.

Sample Output:

```
admin:file search activelog syslog/CiscoSyslog ServerDown
Searching path: /var/log/active/syslog/CiscoSyslog
Searching file: /var/log/active/syslog/CiscoSyslog
Search completed
```

Step 2 Check the DB replication status on all the Cisco Finesse cluster nodes to ensure that all servers are replicating database changes successfully.

- Check by using RTMT—Choose **CallManager > Service > Database Summary** and verify the replication status.
- Check by using CLI—Enter **show perf query class "Number of Replicates Created and State of Replication"** command and verify the application event log.

Sample Output:

```
admin: show perf query class "Number of Replicates Created and
State of Replication"
==>query class :
- Perf class (Number of Replicates Created and State of Replication)
has instances and values:
ReplicateCount -> Number of Replicates Created = 344
ReplicateCount -> Replicate_State = 2
```

Note All nodes must show the Replicate_State as 2. The value 2 refers that the replication state is good.

Step 3 Check network connectivity and DNS server configuration by running the **utils diagnose module validate_network** command on all Cisco Finesse cluster nodes.

Sample Output:

```
admin:utils diagnose module validate_network
Log file: platform/log/diag1.log
Starting diagnostic test(s)
=====
test - validate_network      : Passed
Diagnostics Completed
The final output will be in Log file: platform/log/diag1.log
Please use 'file view activelog platform/log/diag1.log' command to see the output
```

- Step 4** Run a manual Disaster Recovery System (DRS) backup to ensure that all Cisco Finesse cluster nodes and active services are backed up successfully. For more information, see *Disaster Recovery* chapter in the *Administration Guide for Cisco Unified Communications Manager* at <https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/products-maintenance-guides-list.html>.

Disaster Recovery

Important If you have DRS backup, then you can restore your system from the backup files in case of any failure.

Note The DRS backup files are invalid after successful change of hostname, IP address, or domain name.

Change IP Address or Hostname using Unified Operating System GUI

You can use Cisco Unified Operating System Administration to change the IP address or hostname of the Cisco Finesse cluster nodes in your deployment.

Before you begin

- Perform the system health checks on your deployment. For more information, see
- Ensure that Single Sign-On is disabled.

Step 1 In Cisco Unified OS Administration, choose **Settings > IP > Ethernet**.

Step 2 Change the **Hostname** and **IP Address**. If required, change the **Default Gateway**.

Step 3 Click **Save**.

Node services automatically restart with the new changes. Restarting services ensures the proper update and service-restart sequence for the changes to take effect.

Changing the hostname triggers an automatic self-signed certificate regeneration.

Note Do not proceed if the new hostname does not resolve to the correct IP address.

Step 4 Restart the Cisco Finesse cluster nodes using CLI **utils system restart**.

a) Enter **y** and press **Enter** to restart the system.

Sample Output:

```
admin:utils system restart
*** WARNING ***
Please make sure database replication setup is complete. Use the command 'utils dbreplication
runtimestate' to get the current status.
```

```

If database replication setup is in progress, restarting the server may leave database replication
unable to complete.
Do you really want to restart ?
Enter (yes/no)? yes
Appliance is being Restarted ...
Warning: Restart could take up to 5 minutes.
Stopping Service Manager...
- Service Manager shutting down services... Please Wait

```

What to do next

[Post-Change Tasks and Verification, on page 34.](#)

Change IP Address or Hostname Using CLI

You can use the CLI to change the IP address or hostname of the Cisco Finesse cluster nodes in your deployment.



Note If the certificates are invalid or have expired, you must renew the certificates before using the CLI set network hostname.

Before you begin

- Perform the system checks on your deployment. For more information, see
- Ensure that Single Sign-On is disabled.

Step 1 Sign in to the CLI of the Cisco Finesse cluster node that you want to change.

Step 2 Enter **set network hostname**.

Step 3 Follow the prompts to change the hostname, IP address, and default gateway.

- Enter the new hostname and press **Enter**.
- Enter **yes**, if you also want to change the IP address. Otherwise, press **Enter** and move to Step 4.
- Enter the new IP address.
- Enter the subnet mask.
- Enter the address of the gateway.

Step 4 Verify that all your input is correct and enter **yes** to start the process.

Note Do not proceed if the new hostname does not resolve to the correct IP address.

Sample Output:

```

admin:set network hostname
***  W A R N I N G  ***
Do not close this window without first canceling the command.
This command will automatically restart system services.
The command should not be issued during normal operating
hours.
=====

```

Note: Please verify that the new hostname is a unique name across the cluster and, if DNS services are utilized, any DNS configuration is completed before proceeding.

```

=====
Security Warning : This operation will regenerate
                  all CUCM Certificates including any third party
                  signed Certificates that have been uploaded.
Enter the hostname:: samplehostname
Would you like to change the network ip address at this time [yes]:: yes
Warning: Do not close this window until command finishes.
ctrl-c: To quit the input.
          ***  W A R N I N G  ***
=====
Note: Please verify that the new ip address is unique
      across the cluster.
=====
Enter the ip address:: 10.10.10.9
Enter the ip subnet mask:: 255.255.255.224
Enter the ip address of the gateway:: 10.10.10.1
Hostname:      samplehostname
IP Address:    10.10.10.9
IP Subnet Mask: 255.255.255.224
Gateway:      10.10.10.1

Do you want to continue [yes/no]? yes

calling 1 of 6 component notification script: ahostname_callback.sh
Info(0): Processnode query returned using finesse25:
name
=====
finesse25
updating server table from:'finesse25', to: 'samplehostname'
Rows: 1
updating database, please wait 90 seconds
updating database, please wait 60 seconds
updating database, please wait 30 seconds
calling 2 of 6 component notification script: clm_notify_hostname.sh
calling 3 of 6 component notification script: drf_notify_hostname_change.py
calling 4 of 6 component notification script: idsLocalPrefsUpdateFile.sh
Going to trigger /usr/bin/python /usr/local/cm/lib/dblupdatefiles-plugin.py -f=samplehostname,finesse25
calling 5 of 6 component notification script: regenerate_all_certs.sh
calling 6 of 6 component notification script: update_idsenv.sh
calling 1 of 3 component notification script: afupdateip.sh
calling 2 of 3 component notification script: ahostname_callback.sh
Info(0): Processnode query returned using 192.168.1.25:
name
=====
calling 3 of 3 component notification script: clm_notify_hostname.sh

```

Step 5 Restart the Cisco Finesse cluster node using CLI `utils system restart`.

a) Enter **y** and press **Enter** to restart the system.

Sample Output:

```

admin:utils system restart
          ***  W A R N I N G  ***
Please make sure database replication setup is complete. Use the command 'utils dbreplication
runtimestate' to get the current status.
If database replication setup is in progress, restarting the server may leave database replication
unable to complete.
Do you really want to restart ?
Enter (yes/no)? yes
Appliance is being Restarted ...

```

```
Warning: Restart could take up to 5 minutes.
Stopping Service Manager...
- Service Manager shutting down services... Please Wait
```

What to do next

[Post-Change Tasks and Verification, on page 34.](#)

Change IP Address Only

You can use the CLI to change the IP address of the Cisco Finesse cluster nodes in your deployment.

Before you begin

- Perform the system health checks on your deployment. For more information, see [System Health Checks](#).
- Ensure that Single Sign-On is disabled.

Step 1 Sign in to the CLI of the Cisco Finesse cluster node that you want to change.

Step 2 Enter **set network ip eth0 addr mask gw**.

Table 4: Syntax Description

Parameters	Description
eth0	Specifies Ethernet interface 0.
<i>addr</i>	Specifies the server IP address that you want to assign.
<i>mask</i>	Specifies the server network mask that you want to assign.
<i>gw</i>	Specifies the default gateway of the server that you want to assign.

a) Enter **y** and press **Enter** to start the process.

Sample Output:

```
admin:set network ip eth0 10.53.57.101 255.255.255.224 10.53.56.1
*** W A R N I N G ***
This command will restart system services
=====
Note: Please verify that the new ip address is unique
      across the cluster and, if DNS services are
      utilized, any DNS configuration is completed
      before proceeding.
=====
Continue (y/n)?
```

Step 3 Restart the Cisco Finesse cluster nodes using CLI **utils system restart**.

a) Enter **y** and press **Enter** to restart the system.

Sample Output:

```

admin:utils system restart
      ***  W A R N I N G  ***
Please make sure database replication setup is complete. Use the command 'utils dbreplication
runtimestate' to get the current status.
If database replication setup is in progress, restarting the server may leave database replication
unable to complete.
Do you really want to restart ?
Enter (yes/no)? yes
Appliance is being Restarted ...
Warning: Restart could take up to 5 minutes.
Stopping Service Manager...
- Service Manager shutting down services... Please Wait

```

What to do next

[Post-Change Tasks and Verification, on page 34.](#)

Change Domain Name

You can use the CLI to change the domain name of the Cisco Finesse cluster nodes in your deployment.

Before you begin

- Perform the system health checks on your deployment. For more information, see
- Ensure that Single Sign-On is disabled.

Step 1 Sign in to the CLI of the Cisco Finesse cluster node that you want to change.

Step 2 Enter **set network domain** *name*.

Table 5: Syntax Description

Parameters	Description
<i>name</i>	Specifies the system domain name that you want to assign..

a) Enter **y** and press **Enter** to start the process.

Sample Output:

```

admin:set network domain sampledomainname
      ***  W A R N I N G  ***
Adding/deleting or changing domain name on this server will break
database replication. Once you have completed domain modification
on all systems that you intend to modify, please reboot all the
servers in the cluster. This will ensure that replication keeps
working correctly. After the servers have rebooted, please
confirm that there are no issues reported on the Cisco Unified
Reporting report for Database Replication.

```

The server will now be rebooted. Do you wish to continue.

```
Security Warning : This operation will regenerate host certificates.
Continue (y/n)? y
```

Step 3 Restart the Cisco Finesse cluster nodes using CLI **utils system restart**.

a) Enter **y** and press **Enter** to restart the system.

Sample Output:

```
admin:utils system restart
***  W A R N I N G  ***
Please make sure database replication setup is complete. Use the command 'utils dbreplication
runtimestate' to get the current status.
If database replication setup is in progress, restarting the server may leave database replication
unable to complete.
Do you really want to restart ?
Enter (yes/no)? yes
Appliance is being Restarted ...
Warning: Restart could take up to 5 minutes.
Stopping Service Manager...
- Service Manager shutting down services... Please Wait
```

What to do next

- Enter **utils service list** to verify list of all services and their states.
- Enter **show network eth0 detail** to verify the new domain name.
- [Post-Change Tasks and Verification, on page 34](#).

Post-Change Tasks and Verification

Verify if the IP address, hostname, or domain name changes that were made to your deployment are implemented successfully.



Note If you do not receive the results that you expect when you perform these tasks, do not continue with this procedure. Resolve any problems that you find, and then continue.

-
- Step 1** Check the active ServerDown alerts to ensure that all servers in the Cisco Finesse cluster nodes are active and available.
- Step 2** If you are on a subscriber node, and the `show network cluster output` displays incorrect publisher information, use the `set network cluster publisher hostname/IP_address` CLI command to change the publisher hostname or IP address.
- Step 3** Restart the Cisco Finesse cluster nodes using CLI `utils system restart`. Make sure that the cluster output displays the correct publisher before proceeding.
- Step 4** Check the db replication status on all the Cisco Finesse cluster nodes to ensure that all servers are replicating database changes successfully.

- Step 5** Check network connectivity and DNS server configuration by running the **utils diagnose module validate_network** command on all the Cisco Finesse cluster nodes.
- Step 6** Start a manual DRS backup to ensure that all the Cisco Finesse cluster nodes and active services are backed up successfully.
- Step 7** Enable SSO and perform the following tasks.
- Regenerate the SAML certificate.
 - Reestablish trust relationship between Identity Provider (IdP) and Cisco Identity Service (IdS).
 - If the components are registered earlier, then
 - Reregister all the SSO components.
 - Perform the SSO Test to check if all the SSO components are registered. Verify that the test is successful for each component.
- For more information, see the *Single Sign-On* chapter in *Cisco Unified Contact Center Enterprise Features Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html>.
- Step 8** Regenerate and upload the following certificates that contain new hostname or domain name.
- CA-signed certificate to Cisco Finesse.
 - Cisco Finesse tomcat certificate to aCTI server and third-party server (if necessary).
 - Cisco Finesse tomcat-trust certificate to Cisco Unified Intelligence Center and Live Data.
 - Cisco Finesse tomcat certificate to Customer Collaboration Platform as tomcat-trust.
- Step 9** Update Cross-Origin Resource Sharing (CORS) requests for Cisco Finesse, Cisco Unified intelligence center, and Live Data.
- Step 10** Enable Shindig allowed list to add Cisco Finesse new FQDN for Cisco Finesse, Unified Intelligence Center, and Live Data.
- Step 11** Verify and update Finesse desktop layout with new FQDN for the resource loading.
- Step 12** Update Unified CCE inventory with the Cisco Finesse IP address.
- From Step 6 to Step 10, after you complete each step, you must restart the services to reflect new changes.
-



CHAPTER 4

Manage Call Variables Layouts

- [Call Variables Layouts, on page 37](#)
- [Call Variables, on page 37](#)
- [Configure Call Variables Layouts, on page 39](#)
- [Add ECC Variables to Call Variables Layout, on page 40](#)
- [Assign Call Variables Layouts, on page 41](#)
- [Manipulate Call Variables Layouts with a Workflow, on page 41](#)

Call Variables Layouts

You can use the Call Variables Layouts gadget to define how call variables appear on the Finesse agent desktop. You can configure up to 200 unique Call Variables Layouts (one default and 199 custom layouts). As part of this functionality:

- Each layout has a name (required) and description (optional).
- You can change the name and description of the default Call Variables Layout.
- You cannot delete the default Call Variables Layout.
- Finesse appends (*Default*) to the name of the default Call Variables Layout.
- To display a custom Call Variables Layout, in the Unified CCE routing script set the `user.Layout ECC` variable to the name of a configured Call Variables Layout. In this case, if no custom layouts match the `user.Layout` value (or no custom layouts are configured), Finesse displays the default layout.
- Finesse retains the custom layout as specified by the `user.Layout ECC` variable on CTI server failover. During PG failover, Finesse changes the active call layout to the default layout while retaining the call variables and time indicators.

Call Variables

Each Call Variables Layout supports one variable in the header of the call control gadget and up to a total of 20 variables in two columns below the header (up to 10 in each column). You can use call variables, Extended Call Context (ECC) variables, or the following Outbound Option ECC variables:

- BACampaign

- BAAccountNumber
- BAResponse
- BAStatus
- BADialedListID
- BATimeZone
- BABuddyName

Columns can be empty.

The administrator can include the following additional fields in the Call Variables Layout. These variables appear as a drop-down list in the call variable gadget which the admin can assign to a layout.

- queueNumber
- queueName
- callKeyCallId
- callKeyPrefix
- callKeySequenceNum
- wrapUpReason



Note

The callKeyPrefix indicates the day when the call was routed.

The callKeyCallId indicates the unique number for the call routed on that day.

To uniquely locate the call in Unified CCE database records, concatenate the two variables callKeyPrefix and callKeyCallId.

To enable Outbound Option data to appear in Cisco Finesse, the administrator must edit the Default Layout to include some or all Outbound Option variables.

Edit Call Variables

Administrator can set call variables (callVariable1 to callVariable10) values and ECC variable values as editable. Amongst BA (campaign-based outbound calls) variables, only BAResponse can be edited. The agent and the supervisor can edit the call variable values during an active call or in the wrap-up state.

**Note**

- Cisco Finesse refers to the ECC variable length from the AWDB and this length is validated while you edit the ECC variable. Cisco Finesse server takes about 15 minutes to update these changes from AWDB. Agents must sign in again for the ECC variable configuration changes to reflect in the Cisco Finesse desktop.
- Call variables edit operation updates the values of the variables within the particular call. All entities listening to dialog events receive the updated call variables through the Cisco Finesse notifications. If any CTI clients are connected to the same Agent PG, they also receive notifications of the changed call data though CTI call events. However, application scripts or databases that are used to populate the call variables are not directly affected by this edit.

**Note**

Call variables edit operation updates the values of the variables within the particular call. All entities listening to dialog events receive the updated call variables through the Cisco Finesse notifications. If any CTI clients are connected to the same Agent PG, they also receive notifications of the changed call data though CTI call events. However, application scripts or databases that are used to populate the call variables are not directly affected by this edit.

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

Configure Call Variables Layouts

Step 1

From the Manage Call Variables Layouts gadget:

- Click **New** to create a new Call Variables Layout.
- Choose a layout from the list and click **Edit** to modify an existing Call Variables Layout (or click **Delete** to remove it).

Step 2

Under **Create New Layout** (or under Edit <layout name> when editing an existing layout):

- Enter a name for the Call Variables Layout (maximum 40 characters).
- Enter a description of the Call Variables Layout (maximum 128 characters).

Step 3

Under Call Header Layout:

- Enter the display name that you want to appear in the header of the Call Control gadget on the Finesse desktop. For example, Customer Name (maximum 50 characters).
- From the drop-down list, choose the call variable or Outbound Option ECC variable that you want to appear in the header. For example, callVariable3 (maximum 32 characters).

Step 4

In the Call Body Left-Hand Layout and Call Body Right-Hand Layout areas:

- a) Click **Add Row** to add a new row (or click the “X” to delete a row).

b) For each row:

- Enter the display name that you want to appear on the desktop. For example, Customer Name (maximum 50 characters).
- Enter the corresponding call variable or Outbound Option ECC variable from the drop-down list (maximum 32 characters).

Step 5 Select up to five call variables using the check box. The selected call variables are displayed in agent call popover and supervisor active call details.

Note If you do not select any call variables, the first two call variables from the Call Body Left-Hand layout area will be displayed in the agent call popover and supervisor active call details. If there are no call variables in the Left-hand layout area, then the call variables in the Right-Hand Layout will be selected.

Step 6 Turn on the toggle switch to enable the edit option for a specific call variable. By default, this option is turned off.

Note

- Call variable (callVariable1 to callVariable10) values are editable.
- ECC variable values are editable.
- Amongst BA variables (campaign-based outbound calls), only BAResponse value is editable.

Step 7 Click **Save** to save the changes, or **Cancel** to discard the changes.

Note When you modify the Call Variables Layout of the agent desktop, the changes you make take effect after three seconds. However, agents or supervisors who are signed in when the changes are made must sign out and sign back in to see those changes reflected on their desktops.

Step 8 To view the latest configured Call Variables Layout, click **Refresh** from the Manage Call Variables Layouts gadget.

Call Variables Popover

In the call layout popover configuration, you can configure the call header and up to five call variables in the Call Variables Layout. These variables are displayed in the agent's call popover and active call details in the Team Performance gadget for a supervisor.

If you do not select any call variables, the first two call variables from the Call Body Left-Hand layout area will be displayed in the agent call popover and supervisor active call details. If there are no call variables in the Left-hand layout area, then the call variables in the Right-Hand Layout will be selected.

In upgrade scenarios, by default, the first two call variables will be displayed in the agent call popover and supervisor active call details. You can modify the configuration of the popover variables to improve the agent and supervisor experience.

Add ECC Variables to Call Variables Layout

Step 1 In the header or the row where you want the ECC variable to appear, from the Variable drop-down list, choose **Custom**.

- Step 2** In the Custom/ECC Variable Name field, enter the name of the ECC variable you want to appear on the agent desktop.
- Step 3** Click **Set**.
- The ECC variable now appears in the Variable drop-down list for selection.
-

Assign Call Variables Layouts

- Step 1** In CCE Configuration Manager, create an ECC variable called **user.Layout** in the Expanded Call Variable list.
- Note** If a user.layout and a user.Layout are specified, Finesse will prioritize user.layout over user.Layout. If the layout specified in the user.Layout or user.layout is not found, Finesse uses the Default layout.
- Step 2** Add **user.Layout** to the CCE routing script. Use a Set Variable node in an appropriate place in the script to set the value of user.Layout to the name of the call variables layout to display. The layout name should match the name of a call variables layout that was created on the Call Variables Layout tab in Finesse Administration.
-

Manipulate Call Variables Layouts with a Workflow

You can manipulate the call variables layout that an agent sees when a call is answered by using a workflow. To do so, configure an HTTP Request workflow action and set the value of the ECC variable user.Layout to the name of the custom layout to display.

For information about how and when workflows are run, see **Workflows and Workflow Actions**.

For more details, see the section, "Adding an HTTP Request Workflow Action" in the technical paper *Cisco Finesse: How to Create a Screen-Pop Workflow*.



CHAPTER 5

Manage Desktop Layout

- [Gadgets and Components](#), on page 43
- [Finesse Desktop Layout XML](#), on page 45
- [Default Layout XML](#), on page 45
- [Update Default Desktop Layout](#), on page 48
- [Contact Center AI Gadgets](#), on page 57
- [Manage Digital Channels gadget](#), on page 59
- [Configure Multi-Tab Gadget Layout](#), on page 66
- [Drag-and-Drop and Resize Gadget or Component](#), on page 69
- [Drop Participants from Conference](#), on page 70
- [Customize Desktop Properties](#), on page 72
- [Customize Gadget Properties](#), on page 74
- [Horizontal Header](#), on page 74
- [Customize Title and Logo in the Header](#), on page 75
- [alternateHosts Configuration](#), on page 76
- [Headless Gadget Configuration](#), on page 76
- [Customize Icons in Left Navigation Bar](#), on page 76
- [XML Schema Definition](#), on page 91
- [Live Data Reports](#), on page 95

Gadgets and Components

Gadgets

Cisco Finesse is an OpenSocial gadget, which is an XML document that defines metadata for an OpenSocial Gadget container. The gadgets are applications that are placed within the Cisco Finesse desktop. This helps administrator to provide access to the contact center agents for all the applications that is required to service calls inside a single application.

Cisco Finesse comes with default gadgets such as, the team performance gadget, call control gadget, and call popover. JavaScript library is available for any customers with specific requirements that are not available out of the box.

Gadgets are listed in the desktop layout using the `<gadget>` tag.



Note Finesse Desktop is tested to perform well with an average of 20 gadgets per Desktop (across all tabs), over a sign in period of 8 minutes for 2000 users (agents and supervisors). When you increase the total number of gadgets that are configured on the Desktop, the CPU consumption marginally increases during users sign in. When all the configured gadgets are enabled for all the users, it impacts the Finesse server. Higher number of gadgets will also need more browser memory and network bandwidth.

If considerably larger number of gadgets are configured or if more users sign in (more than the tested number of users) in a short time frame, you must monitor the CPU consumption and network bandwidth during users sign in and ensure that the end-point devices have enough memory.

Failover uses optimization to sign in the users quickly and is not considered the same as a new browser sign in.

Third-party gadgets are hosted on the Cisco Finesse server using the `3rdpartygadget` web application or on an external web server. Gadgets can make REST requests to services hosted on external servers using the Cisco Finesse JavaScript Library API. To avoid browser cross-origin issues, REST requests are proxied through the backend Shindig web application. Third-party gadgets must implement their own authentication mechanisms for third-party REST services.

For more information about gadgets, see <https://developer.cisco.com/docs/finesse/>.

Cisco Finesse Desktop Interface APIs used for desktop specific configurations and settings can be found in the *Cisco Finesse Desktop Interface API Guide* on [DevNet](#).

Components

Components are simple scripts that are loaded into the desktop directly at predefined positions as directed by the layout, without an enclosing frame and its document.

Components are introduced in the desktop to overcome a few rendering limitations and performance considerations inherent to gadgets.

The `<component>` tag lists the components in the desktop layout. Currently, the layout validations prevent creating custom components. Hence, default components are allowed in the desktop layouts. The default desktop functionalities are currently registered as components to provide flexibility and to reduce the load on the server.

Multi-Tab Gadgets

Finesse desktop supports accessing multiple gadgets through tabs within a single gadget called Multi-Tab gadget. The Multi-Tab gadget allows rendering of more gadgets in a single desktop view to present more information to agents and supervisors in a concise and readily accessible manner. With Multi-Tab gadgets, agents and supervisors do not have to scroll the page or switch the desktop container tab to see additional information.

The Multi-Tab gadget can host any gadget supported by the desktop. Multiple instances of Multi-Tab gadgets are supported, which allows agents and supervisors to stack groups of gadgets to customize their desktop.

For more information on the list of features that are supported by Multi-Tab gadgets, see the *Multi-Tab Gadgets* section in *Cisco Finesse Agent and Supervisor Desktop User Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-user-guide-list.html>.

For more information on configuring the layout for Multi-Tab gadget, see [Configure Multi-Tab Gadget Layout, on page 66](#).

Call Control Gadget in Multi-Tab

Finesse desktop Call Control gadget can be hosted as a tab within the Multi-Tab gadget.

When the Call Control gadget is deployed in a Multi-Tab gadget, the Call Control gadget automatically hides or shows depending on whether a call is active. It also shows notifications for changes in the call context. The notifications can be customized through desktop properties. For more information see *Notifications of Call Control in Multi-Tab Gadgets* at [Desktop Properties, on page 208](#).

For more information on the Multi-Tab gadgets see the *Multi-Tab Gadgets* section in *Cisco Finesse Agent and Supervisor Desktop User Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-user-guide-list.html>.

Finesse Desktop Layout XML

The Finesse Layout XML defines the layout of the Finesse desktop, and the gadgets and components displayed on the desktop.

Use the Manage Desktop Layout gadget to upload an XML layout file to define the layout of the Finesse desktop for agents and supervisors.

Actions on the **Manage Desktop Layout** gadget are as follows.

- Edit the code using any of the following editors:
 - **Text Editor**
 - **XML Editor**
- **View Default Layout** - Displays the Cisco Finesse default layout.
- **Restore Default Layout** - Restores the Cisco Finesse desktop to the default layout.
- **Save** - Saves your configuration changes.
- **Revert** - Retrieves and applies the most recently saved desktop layout.

Default Layout XML

The **Manage Desktop Layout** supports the following types of editors:

- **Text Editor**—A plain text editor. It is the default editor. You can use the **Expand All** option to see all the code details and Search text box to refine your search results.
- **XML Editor**—An XML editor.



Note

- You cannot add or edit comments (`<!--` and `-->`) in the **XML Editor**.
 - In this document, all the examples that are related to desktop layout are applicable for text editor.
-

Both the editors support the following features:

- Expand and collapse option.
- Syntax highlights and color code for the visual indication.
- Auto-complete suggestions and hints for valid elements in the tags.

The Cisco Finesse default desktop layout XML for Unified CCE and Packaged CCE contains optional gadgets and notes. The notes describe how to modify the layout for your deployment type.

Optional Live Data gadgets in the layout XML are commented out. After you install and configure Live Data, remove the comment tags from the reports that you want to appear on the desktop.

Following are the updates available in the default layout XML for Cisco Finesse desktop:

- Sample configurations for customizing desktop properties are added to the default layout (**Desktop Layout**) and team-specific layout (**Team Resources > Desktop Layout**).

For upgraded layouts, sample configurations for customizing desktop properties do not appear by default. The administrator must copy the XML from the **View Default Layout** and add to the respective custom layouts.

- Horizontal Header is available in the layout configuration and the Header can be customized.
- Title and Logo of Cisco Finesse desktop can be customized.
- Desktop Chat, TeamMessage, Dialer, Agent Identity, and Non-Voice State Control are added as part of the header component.

For upgraded layouts, TeamMessage and Desktop Chat will not appear by default. The XML must be copied from the default layout and added to the respective custom layouts. See *Cisco Cisco Finesse Installation & Upgrade Guide*.

- Vertical tabs in Cisco Finesse desktop are moved to collapsible left navigation bar for which the icons can be customized.
- Support for inbuilt java script components has been added.
- The **ID** attribute (optional) is the ID of the HTML DOM element used to display the gadget or component. The ID should start with an alphabet and can contain alphanumeric characters along with hyphen(-) and underscore(_). It can be set through the Cisco Finesse Administrative portal and has to be unique across components and gadgets.
- The **managedBy** attribute (optional) for Live Data gadgets defines the gadgets which manage these Live Data gadgets. The value of **managedBy** attribute for Live Data gadgets is **team-performance**. This means that the rendering of the gadget is managed by the Team Performance gadget. These gadgets are not rendered by default, but will be rendered when the options Show State History and Show Call History are selected in the Team Performance gadget.

For upgraded layouts, the **managedBy** attribute will be introduced, and will have the value of the **ID** of the Team Performance gadget in the same tab. If there are multiple instances of Team Performance gadgets and Live Data gadget pairs, they will be associated in that order. If the **ID** of the Team Performance gadget is changed, the value of the **managedBy** attribute should also be updated to reflect the same **ID** for the Live Data gadgets. Otherwise, the Team Performance gadget instance will not show its respective Live Data gadgets.

- Gadgets that are hosted within a Multi-Tab gadget need to contain the **ID** of the Multi-Tab gadget as the value of its **managedBy** attribute.

- The **Hidden** attribute (optional) is used to support headless gadgets. When an attribute is set to `hidden="true"`, then the gadget is loaded by the container, but will not be displayed. The default value set for the attribute is `"false"`.
- The **Hidden** attribute can be used within Multi-Tab gadgets to make specific gadget tabs to appear or hide depending on the context.
- The **default** attribute (`default="true"`) can be used to set a particular gadget as the default focus so that its contents are visible when hosted within a Multi-Tab gadget. The default gadget is always listed as the left most gadget.



Note After the user customizes the order of the gadgets by dragging them, this attribute will no longer be supported. For more information on the Multi-Tab gadget order, see the *Ordering of Multi-Tab Gadget Tabs* section in *Cisco Finesse Agent and Supervisor Desktop User Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-user-guide-list.html>.

- **maxRows** is changed from being a query parameter to an attribute.

Example of **maxRows** being a query parameter:

```
<gadget id="team-performance">/desktop/scripts/js/teamPerformance.js?maxRows=5</gadget>
```

Example of **maxRows** being an attribute:

```
<gadget id="team-performance" maxRows="5">/desktop/scripts/js/teamPerformance.js</gadget>
```

During an upgrade it will be removed from the URL of the team performance gadget and added as an attribute. The **maxRows** attribute (optional) is used to adjust the height of the Team Performance gadget. If there are multiple instances of the Team Performance gadget, each instance height can be set by using this attribute. During an upgrade the height of the team performance gadget will be retained. By default the **maxRows** attribute value is set to 10 rows.

If any changes are made to the component IDs or URLs in the default XML layout, the following features may not work as expected.

Note that the components can be rearranged in any order to show on the Cisco Finesse desktop.

Feature	Component ID	URL
Title and Logo	cd-logo	<url>/desktop/scripts/js/logo.js</url>
Voice State Control	agent-voice-state	<url>/desktop/scripts/js/agentvoicestate.component.js</url>
Non-voice state control	nonvoice-state-menu	<url>/desktop/scripts/js/nonvoice-state-menu.component.js</url>
TeamMessage	broadcastmessagepopover	<url>/desktop/scripts/js/teammessage.component.js</url>
Desktop Chat	chat	<url>/desktop/scripts/js/chat.component.js</url>
Dialer	make-new-call-component	<url>/desktop/scripts/js/makenewcall.component.js</url>
Agent identity	identity-component	<url>/desktop/scripts/js/identity-component.js</url>

Update Default Desktop Layout

When you modify the layout of the Finesse desktop, the changes you make take effect on the desktop after 3 seconds. However, agents who are signed in when the changes are made must sign out and sign in again to see those changes reflect on the desktop.



Note The call control gadget is supported at both page level and tab level. However, the call control does not support being added to multiple pages and cannot detect this error correctly. Therefore, take care to ensure that the call control gadget is not configured to be added to multiple pages when editing the desktop layout.

The version tag of Desktop Layout XML can't be edited.

For the changes to take effect, refresh the page, or sign out and sign in again into Cisco Finesse.

Step 1 Click **Desktop Layout**.

Step 2 Select from the following editors:

- **Text Editor**
- **XML Editor**

Step 3 Make changes to the XML as required.

Example:

If you want to add a new tab called Reports, add the following XML within the tabs tags under the `<role>Agent</role>` tag:

```
<tab>
  <id>reports</id>
  <icon>Reports</icon>
  <label>Reports</label>
</tab>
```

If you want to add this tab to the supervisor desktop, add the XML within the tabs tags under the `<role>Supervisor</role>` tag.

To add a gadget to a tab, add the XML for the gadget within the gadgets tag for that tab.

```
<gadgets>
<gadget>https://<ipAddress>/gadgets/<gadgetname>.xml</gadget>
</gadgets>
```

Replace `<ipAddress>` with the IP address of the server where the gadget resides.

If you want to add multiple columns to a tab on the Finesse desktop, add the gadgets for each column within the columns tags for that tab. You can have up to four columns on a tab.

```
<tabs>
  <tab>
    <id>home</id>
    <icon>home</icon>
    <label>finesse.container.tabs.agent.homeLabel</label>
    <columns>
      <column>
```



```

        <gadgets>
          <gadget>/desktop/scripts/js/queueStatistics.js</gadget>
        </gadgets>
      </column>
    </columns>
  </tab>
<tab>
  <id>myHistory</id>
  <icon>history</icon>
  <label>finesse.container.tabs.agent.myHistoryLabel</label>
  <columns>
    <column>
      <!-- The following gadgets are used for viewing the call history
and state history of an agent. -->
    </column>
  </columns>
</tab>

```

Step 4 Click **Save**.

Finesse validates the XML file to ensure that it's valid XML syntax and conforms to the Finesse schema.

Step 5 After you save your changes, if you want to revert to the last saved desktop layout, click **Revert**. If you want to revert to the default desktop layout, click **Restore Default Layout**.

Note During upgrade, any changes made to the Cisco Finesse Default Layout won't be updated. Click on **Restore Default Layout** to get the latest changes.

The Finesse default XML layout is as follows:

```

<finesseLayout xmlns="http://www.cisco.com/vtg/finesse">
  <!-- DO NOT EDIT. The version number for the layout XML. -->
  <version>1252.02</version>
  <configs>
    <!-- The Title for the application which can be customized. -->
    <config key="title" value="Cisco Finesse"/>
    <!-- The following entries are examples of changing defaults for desktop properties.

    To change any property, uncomment the respective line and set the appropriate value.

    For more details on the properties that can be customized, refer to the Cisco Finesse
Administration Guide.
    Note: The customized properties can only be set in the configs section and are not
role-specific. -->
    <!-- <config key="enableDragDropAndResizeGadget" value="false"/> -->
    <!-- <config key="wrapUpCountDown" value="true"/> -->
    <!-- <config key="desktopChatAttachmentEnabled" value="true"/> -->
    <!-- <config key="forceWrapUp" value="true"/> -->
    <!-- Possible Values: supervisor_only, conference_controller_and_supervisor, all
-->
    <!-- <config key="enableDropParticipantFor" value="supervisor_only"/> -->
    <!-- Possible Values: agents, all -->
    <!-- <config key="dropParticipant" value="agents"/> -->
    <!-- <config key="displayNavLevelGadgetsFirstInMultitab" value="false"/> -->
    <!-- The logo file for the application -->
    <!-- For detailed instructions on using custom icons for logos and tabs,
please refer to the section "Customize Title and Logo in the Header"
in the Finesse Administration Guide. -->
    <!-- <config key="logo" value="/3rdpartygadget/files/cisco_finext_logo.png"/> -->

```

```

    </configs>
  <header>
    <!-- Please ensure that at least one gadget/component is present within every
headercolumn tag -->
  <leftAlignedColumns>
    <headercolumn width="300px">
      <component id="cd-logo">
        <url>/desktop/scripts/js/logo.js</url>
      </component>
    </headercolumn>
    <headercolumn width="230px">
      <component id="agent-voice-state">
        <url>/desktop/scripts/js/agentvoicestate.component.js</url>
      </component>
    </headercolumn>
    <headercolumn width="251px">
      <component id="nonvoice-state-menu">
        <url>/desktop/scripts/js/nonvoice-state-menu.component.js</url>
      </component>
    </headercolumn>
  </leftAlignedColumns>
  <rightAlignedColumns>
    <headercolumn width="50px">
      <component id="broadcastmessagepopover">
        <url>/desktop/scripts/js/teammessage.component.js</url>
      </component>
    </headercolumn>
    <headercolumn width="50px">
      <component id="chat">
        <url>/desktop/scripts/js/chat.component.js</url>
      </component>
    </headercolumn>
    <headercolumn width="50px">
      <component id="make-new-call-component">
        <url>/desktop/scripts/js/makenewcall.component.js</url>
      </component>
    </headercolumn>
    <headercolumn width="72px">
      <component id="identity-component">
        <url>/desktop/scripts/js/identity-component.js</url>
      </component>
    </headercolumn>
  </rightAlignedColumns>
</header>
  <layout>
    <role>Agent</role>
    <page>
      <gadget>/desktop/scripts/js/callcontrol.js</gadget>

      <!-- The page level Multi-Tab gadget. -->
      <gadget
id="agentMultiTabGadgetContainer">/desktop/scripts/js/tabbedGadgets.js</gadget>

      <!-- The following gadget is for WXM Customer Experience Journey.
      If WXM is onboarded successfully with all configurations, then replace the url
      with the actual url obtained by exporting the Cisco Finesse gadget from WXM -->
      <!-- <gadget
managedBy="agentMultiTabGadgetContainer">/3rdpartygadget/files/CXService/CiscoCXJourneyGadget.xml</gadget>
      -->

      <!-- The following gadget is for displaying the Answers based on the ongoing

```

```

conversation between
    the agent and the customer. If CCAI service is configured and enabled for an
agent, the gadget will be rendered on
    uncommenting the below code-->
    <!-- <gadget managedBy="agentMultiTabGadgetContainer"
hidden="true">/3rdpartygadget/files/ccaiGadgets/answersGadget.xml</gadget> -->

    <!-- The following gadget is for displaying the Transcript based on the ongoing
conversation between
    the agent and the customer. If CCAI service is configured and enabled for an
agent, the gadget will be rendered on
    uncommenting the below code-->
    <!-- <gadget managedBy="agentMultiTabGadgetContainer"
hidden="true">/3rdpartygadget/files/ccaiGadgets/transcriptGadget.xml</gadget> -->

</page>
<tabs>
  <tab>
    <id>home</id>
    <icon>home</icon>
    <label>finesse.container.tabs.agent.homeLabel</label>
    <columns>
      <column>
        <gadgets>
          <gadget default="true"
managedBy="agentMultiTabGadgetContainer">/desktop/scripts/js/queueStatistics.js</gadget>

          <!-- The Multi-Tab gadget to show live data reports. Uncomment
this for displaying the gadgets managed by it. -->
          <!-- <gadget
id="agentReportsMultiTabContainer">/desktop/scripts/js/tabbedGadgets.js</gadget> -->
          <!--
The following Gadgets are for LiveData.
If you wish to show LiveData Reports, then do the following:
            1) Uncomment each Gadget you wish to show.
            2) Replace all instances of "my-cuic-server.com" with the
Fully Qualified Domain Name of your Intelligence Center Server.
            3) [OPTIONAL] Adjust the height of the gadget by changing
the "gadgetHeight" parameter.
IMPORTANT NOTES:
            - In order for these Gadgets to work, you must have performed
all documented pre-requisite steps.
            - Do *NOT* change the viewId (unless you have built a custom
report and know what you are doing).
            - The "teamName" will be automatically replaced with the
Team Name of the User logged into Finesse (for Team-specific layouts).
-->
          <!-- HTTPS Version of LiveData Gadgets -->
          <!-- TEAM STATUS REPORTS: -->
          <!-- 1. Agent Default view (default) -->
          <!-- <gadget
managedBy="agentReportsMultiTabContainer">https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?

gadgetHeight=310&amp;viewId_1=99E6C8E210000141000000D80A0006C4&amp;filterId_1=agent.id=CL%20teamName</gadget>
-->

          <!-- 2. Agent Skill Group Default view -->
          <!-- <gadget
managedBy="agentReportsMultiTabContainer">https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?

gadgetHeight=310&amp;viewId_1=9AB7848B10000141000001C50A0006C4&amp;filterId_1=agent.id=CL%20teamName</gadget>
-->

```

```

        </gadgets>
      </column>
    </columns>
  </tab>
  <!-- The following gadget is for displaying the Digital Channel Gadget. The
gadget will be rendered on
  uncommenting the below code. Multiple instances of manage digital channels
gadget in desktop layout are not supported.
  Only single instance of manage digital channel gadget can be configured in desktop
  layout.-->
  <!--tab>
    <id>manageDigitalChannel</id>
    <icon>mui-chat-active_16</icon>
    <label>finesse.container.tabs.agent.manageDigitalChannelLabel</label>
    <columns>
      <column>
        <gadgets>

<gadget>/3rdpartygadget/files/ManageDigitalChannel/ManageDigitalChannel.xml</gadget>
        </gadgets>
      </column>
    </columns>
  </tab-->
  <tab>
    <id>myStatistics</id>
    <icon>column-chart</icon>
    <label>finesse.container.tabs.agent.myStatisticsLabel</label>
    <columns>
      <column>
        <gadgets>
          <gadget default="true"
managedBy="agentMultiTabGadgetContainer">https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?

gadgetHeight=150&amp;viewId=0B8D11317ED54A80B64F3AE28C5139E5&amp;filterId=agentStats.id=CL%20teamName</gadget>

Analytics.
          <!-- The following gadget is for WXM Customer Experience
          If WXM is onboarded successfully with all configurations, then
          replace the url
          with the actual url obtained by exporting the Cisco Finesse
          gadget from WXM -->
          <!--
<gadget>/3rdpartygadget/files/CXService/CiscoCXAnalyticsGadget.xml</gadget> -->
          </gadgets>
        </column>
      </columns>
    </tab>
    <tab>
      <id>myHistory</id>
      <icon>history</icon>
      <label>finesse.container.tabs.agent.myHistoryLabel</label>
      <columns>
        <column>
          <!-- The following gadgets are used for viewing the call history
and state history of an agent. -->
          <gadgets>
            <gadget default="true"
managedBy="agentMultiTabGadgetContainer">https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?

gadgetHeight=280&amp;viewId=5FA44C6F930C4A64A6775B21A17EED6A&amp;filterId=agentTaskLog.id=CL%20teamName</gadget>
          </gadgets>
        </column>
      </columns>
    </tab>
  </gadgets>
</column>
</columns>
</tab>
</tab>

```

```

<gadget>https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?gadgetHeight=280&
viewId=56BC5CCE8C37467EA4D4EFA8371258BC&filterId=agentStateLog.id=CL%20teamName</gadget>

    </gadgets>
  </column>
</columns>
</tab>
<!--
The following Gadgets are for LiveData.
If you wish to show More LiveData Reports, then do the following:
  1) Uncomment each Gadget you wish to show.
  2) Replace all instances of "my-cuic-server.com" with the Fully Qualified
Domain Name of your Intelligence Center Server.
  3) [OPTIONAL] Adjust the height of the gadget by changing the "gadgetHeight"
parameter.
IMPORTANT NOTES:
  - In order for these Gadgets to work, you must have performed all documented
pre-requisite steps.
  - Do *NOT* change the viewId (unless you have built a custom report and
know what you are doing).
  - The "teamName" will be automatically replaced with the Team Name of the
User logged into Finesse (for Team-specific layouts).
-->
<!-- If you are showing the "More Live Data Reports" tab, then also uncomment
this section.
<tab>
  <id>moreLiveDataReports</id>
  <icon>reports-more</icon>
  <label>finesse.container.tabs.agent.moreLiveDataReportsLabel</label>
  <gadgets>
-->
<!-- Multi-Tab gadget for extended live data reports. Uncomment this for
displaying the gadgets managed by it.-->
<!-- <gadget
id="moreLiveDataMultiTabContainer"/>desktop/scripts/js/tabbedGadgets.js</gadget> -->
<!-- HTTPS Version of LiveData Gadgets -->
<!-- AGENT REPORTS: 1. Agent Default view (default) -->
<!-- <gadget default="true"
managedBy="agentMultiTabGadgetContainer">https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?

gadgetHeight=310&viewId_1=99E6C8E210000141000000D80A0006C4&filterId_1=agent.id=CL%20teamName</gadget>-->

  <!-- AGENT SKILL GROUP REPORTS: 1. Agent Skill Group Default view (default) -->

  <!-- <gadget
managedBy="moreLiveDataMultiTabContainer">https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?

gadgetHeight=310&viewId_1=9AB7848B10000141000001C50A0006C4&filterId_1=agent.id=CL%20teamName</gadget>-->

  <!-- QUEUE STATUS SKILL GROUP REPORTS: 1. Skill Group Default view (default),
2. Skill Group Utilization view -->
  <!-- <gadget
managedBy="moreLiveDataMultiTabContainer">https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?

gadgetHeight=310&viewId_1=B7371BE210000144000002870A0007C5&filterId_1=skillGroup.id=CL%20teamName&

```

```

viewId_2=9E760C8B1000014B0000005A0A0006C4&amp;filterId_2=skillGroup.id=CL%20teamName</gadget>-->

    <!-- QUEUE STATUS PRECISION QUEUE REPORTS: 1. Precision Queue Default view
    (default), 2. Precision Queue Utilization view -->
    <!-- <gadget
    managedBy="moreLiveDataMultiTabContainer">https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?

gadgetHeight=310&amp;viewId_1=B71A630C10000144000002480A0007C5&amp;filterId_1=precisionQueue.id=CL%20teamName&amp;

viewId_2=286B86F01000014C000005330A0006C4&amp;filterId_2=precisionQueue.id=CL%20teamName</gadget>-->

    <!-- If you are showing the "more reports" tab, then uncomment this section
    too.
        </gadgets>
    </tab>
    -->
    </tabs>
</layout>
<layout>
    <role>Supervisor</role>
    <page>
        <gadget>/desktop/scripts/js/callcontrol.js</gadget>

        <!-- The page level Multi-Tab gadget. -->
        <gadget
    id="supervisorMultiTabGadgetContainer">/desktop/scripts/js/tabbedGadgets.js</gadget>

        <!-- The following gadget is for WXM Customer Experience Journey.
        If WXM is onboarded successfully with all configurations, then replace the url
        with the actual url obtained by exporting the Cisco Finesse gadget from WXM -->

        <!-- <gadget
    managedBy="supervisorMultiTabGadgetContainer">/3rdpartygadget/files/CXService/CiscoCXJourneyGadget.xml</gadget>
        -->

        <!-- The following gadget is for displaying the Answers based on the ongoing
        conversation between
        the agent and the customer. If service is enabled for an agent, the gadget will
        be rendered on
        uncommenting the below code-->
        <!-- <gadget managedBy="supervisorMultiTabGadgetContainer"
    hidden="true">/3rdpartygadget/files/ccaiGadgets/answersGadget.xml</gadget> -->

        <!-- The following gadget is for displaying the Transcript based on the ongoing
        conversation between
        the agent and the customer. If service is enabled for an agent, the gadget will
        be rendered on
        uncommenting the below code-->
        <!-- <gadget managedBy="supervisorMultiTabGadgetContainer"
    hidden="true">/3rdpartygadget/files/ccaiGadgets/transcriptGadget.xml</gadget> -->

    </page>
    <tabs>
        <tab>
            <id>home</id>
            <icon>home</icon>
            <label>finesse.container.tabs.supervisor.homeLabel</label>
            <columns>
                <column>
                    <gadgets>

```

```

        <gadget default="true"
managedBy="supervisorMultiTabGadgetContainer"
id="team-performance">/desktop/scripts/js/teamPerformance.js</gadget>
    <!-- The following gadgets are used for viewing the call history
and state history of an agent selected in the Team Performance Gadget. -->
    <!-- The following gadgets are managed (loaded and displayed)
by the team performance gadget (associated with id "team-performance").
This association is done using the mapping of managedBy
attribute of the managed gadgets, to the id of managing gadget.
If the id for team performance gadget is changed, the
values for the associated managedBy attribute
for the managed gadgets, also need to be updated with the
new id.
These managed gadgets are not displayed by default, but
would be displayed when the option
"view history" is selected, for an agent, in the team
performance gadget.
Note: As managed gadgets are not displayed by default,
placing managed gadgets alone on
separate columns of their own, would display blank space
in that area.
For more details on managed gadgets and managedBy attribute,
please refer to Finesse Administration Guide.
-->
    <gadget
managedBy="team-performance">https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?

gadgetHeight=275&viewId=630CB4C96B0045D9BFF295A49A0BA45E&filterId=agentTaskLog.id=AgentEvent:Id&

type=dynamic&maxRows=20</gadget>
    <gadget
managedBy="team-performance">https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?

gadgetHeight=275&viewId=56BC5CCE8C37467EA4D4EFA8371258BC&filterId=agentStateLog.id=AgentEvent:Id&

type=dynamic&maxRows=20</gadget>
    <!-- The following gadget is for WXM Customer Experience
Analytics.
If WXM is onboarded successfully with all configurations, then
replace the url
with the actual url obtained by exporting the Cisco Finesse
gadget from WXM -->
    <!--
<gadget>/3rdpartygadget/files/CXService/CiscoCXAnalyticsGadget.xml</gadget> -->
    </gadgets>
    </column>
</columns>
</tab>
<!-- The following gadget is for displaying the Digital Channel Gadget. The
gadget will be rendered on
uncommenting the below code-->
<!--tab>
    <id>manageDigitalChannel</id>
    <icon>mui-chat-active_16</icon>
    <label>finesse.container.tabs.agent.manageDigitalChannelLabel</label>
    <columns>
        <column>
            <gadgets>

```

```

<gadget>/3rdpartygadget/files/ManageDigitalChannel/ManageDigitalChannel.xml?
maxDialogLimit=5&interruptAction=IGNORE&dialogLogoutAction=CLOSE</gadget>
  </gadgets>
</column>
</columns>
</tab-->
<tab>
  <id>myHistory</id>
  <icon>history</icon>
  <label>finesse.container.tabs.agent.myHistoryLabel</label>
  <columns>
    <column>
      <!-- The following gadgets are used for viewing the call history
and state history of a logged in supervisor. -->
      <gadgets>
        <gadget default="true"
managedBy="supervisorMultiTabGadgetContainer">
https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?
gadgetHeight=280&viewId=5FA44C6F930C4A64A6775B21A17EED6A&filterId=agentTaskLog.id=CL%20teamName</gadget>
<gadget>https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?gadgetHeight=280&
viewId=56BC5CCE8C37467EA4D4EFA8371258BC&filterId=agentStateLog.id=CL%20teamName</gadget>
      </gadgets>
    </column>
  </columns>
</tab>
<tab>
  <id>teamData</id>
  <icon>team-data</icon>
  <label>finesse.container.tabs.supervisor.teamDataLabel</label>
  <columns>
    <column>
      <!-- The following gadget is used by the supervisor to view an
agent's queue interval details. -->
      <gadgets>
        <gadget default="true" managedBy="supervisorMultiTabGadgetContainer">
https://my-cuic-server.com:8444/cuicui/gadget/LiveData/LiveDataGadget.xml?
gadgetHeight=310&viewId=0B8D11317ED54A80B64F3AE28C5139E5&filterId=agentStats.id=CL%20teamName</gadget>
<gadget>https://my-cuic-server.com:8444/cuicui/gadget/Historical/HistoricalGadget.xml?
viewId=BD9A8B7DBE714E7EE758A9D472F0E7DC&
linkType=htmlType&viewType=Grid&refreshRate=900&@start_date=RELDATE%20THISWEEK&
@end_date=RELDATE%20THISWEEK&@agent_list=CL%20~teams~&gadgetHeight=360</gadget>
      </gadgets>
    </column>
  </columns>
</tab>
<tab>
  <id>queueData</id>
  <icon>storage</icon>

```



```

<label>finesse.container.tabs.supervisor.queueDataLabel</label>
<columns>
  <column>
    <gadgets>
      <gadget default="true"
managedBy="supervisorMultiTabGadgetContainer">/desktop/scripts/js/queueStatistics.js</gadget>
    </gadgets>
  </column>
</columns>
</tab>

</tabs>
</layout>
</finesseLayout>

```

Contact Center AI Gadgets

Cisco solutions such as Unified CCE leverages Artificial Intelligence (AI) and Natural Language Understanding (NLU) to provide services that assist agents. These Contact Center AI services are available for the agents through the Agent Answers gadget and the Call Transcript gadget on the Cisco Finesse desktop.



Note If you are connected to Unified CCE 12.5(1), by default **Agent Answers** and **Call Transcript** gadgets are enabled. You can use the **utils finesse set_property webservices 12.5.CCE.supportedAgentServices CLI [values]** to disable and enable the gadgets.

Add Agent Answers Gadget

This procedure explains how to add an Agent Answers gadget in **Cisco Finesse Administration**.



Note Agent Answers can be added within the Multi-Tab gadget. For more information about Multi-Tab gadget, see the [Configure Multi-Tab Gadget Layout](#) section.

-
- Step 1** Sign in to the administration console on the primary Cisco Finesse server using the URL: *https://FQDN of Finesse server:8445/cfadmin*.
- Step 2** Click **Desktop Layout**.
- Step 3** Select from the following editors:
- Text Editor
 - XML Editor
- Step 4** Uncomment the following code snippet within the gadgets tag for the required role.
For example, if you want the **Agent Answers** gadget to appear for an agent and supervisor on the home page, uncomment (remove the comment (`<!--` and `-->`)) the following code snippet in the gadgets tag within the `<id>home</id>` tab under roles `<role>Agent</role>` and `<role>Supervisor</role>`.

```
<gadgets>
  <!-- <gadget managedBy="agentMultiTabGadgetContainer"
hidden="true"/>/3rdpartygadget/files/ccaiGadgets/answersGadget.xml</gadget> -->
</gadgets>
```

- Note**
- For customized Cisco Finesse desktop layouts or during Cisco Finesse upgrade to Release 12.6(1), the administrator must manually add the gadget.
 - The gadget has been optimized to work when deployed within the multi-tab gadget. It is highly recommended to configure the gadget using a similar layout.
 - The gadget will be hidden if it is disabled in CCE Admin. This would take effect in Finesse after re-login.

Step 5 Click Save.

For more information about enabling a service, see the Agent Answers chapter in the following documents:

- Cisco Unified Contact Center Enterprise Features Guide, Release 12.6(1) at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html>
- Cisco Packaged Contact Center Enterprise Features Guide, Release 12.6(1) at <https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html>

For information about how to use this gadget, see the [Contact Center AI Gadgets Help](#).

Add Transcript Gadget

This procedure explains how to add the Transcript gadget in **Cisco Finesse Administration**.

Step 1 Sign in to the administration console on the primary Cisco Finesse server using the URL: *https://FQDN of Finesse server:8445/cfadmin*.

Step 2 Click **Desktop Layout**.

Step 3 Select from the following editors:

- Text Editor
- XML Editor

Step 4 Uncomment the following code snippet within the gadgets tag for the required role.

For example, if you want the **Transcript** gadget to appear on the home page, uncomment (remove the comment (<!-- and -->)) the following code snippet in the gadgets tag within the <id>home</id> tab, for the required role.

- <role>Agent</role>:

```
<gadgets>
  <!-- <gadget managedBy="agentMultiTabGadgetContainer"
hidden="true"/>/3rdpartygadget/files/ccaiGadgets/transcriptGadget.xml</gadget> -->
</gadgets>
```

- <role>Supervisor</role>:

```
<gadgets>
  <!-- <gadget managedBy="supervisorMultiTabGadgetContainer"
  hidden="true">/3rdpartygadget/files/ccaiGadgets/transcriptGadget.xml</gadget> -->
</gadgets>
```

- Note**
- For customized Cisco Finesse desktop layouts or during Cisco Finesse upgrade to Release 12.6(1), the administrator must manually add the gadget.
 - The gadget has been optimized to work when deployed within the multi-tab gadget. It is highly recommended to configure the gadget using a similar layout.
 - The gadget will be hidden if it is disabled in CCE Admin. This would take effect in Finesse after re-login.

Step 5 Click **Save**.

For more information about enabling a service, see the Call Transcription chapter in the following documents:

- Cisco Unified Contact Center Enterprise Features Guide, Release 12.6(1) at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html>
- Cisco Packaged Contact Center Enterprise Features Guide, Release 12.6(1) at <https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html>

For information about how to use this gadget, see the [Contact Center AI Gadgets Help](#).

Methods to Collect Serviceability Logs

If an agent experiences problems with the Agent or Transcript gadgets, the agent can send a set of desktop logs to the Cisco Finesse server. These desktop logs are also called as serviceability logs. The serviceability logs can be sent in the following two different ways:

- **Send Error Report**

For more information on using the **Send Error Report**, See: *Common Tasks* chapter in the *Cisco Finesse Agent and Supervisor Desktop User Guide*.

- **Log Collection Schedule**

For more information on using the **Log Collection Schedule**, see the Log Collection Schedule section.

Manage Digital Channels gadget

The Contact Center Enterprise (CCE) solution integrates with Webex Connect to offer digital channel capabilities for customers to communicate with the customer contact center. The agents and supervisors can interact with customers who are reaching out through the digital channels supported by Webex Engage, using the Manage Digital Channels gadget. The Manage Digital Channels gadget is a cloud-hosted Cisco Finesse gadget, where agents can sign into the digital channels and manage tasks. The tasks routed to the agents are displayed using Webex Engage, who can then start, transfer, or end it.

The Manage Digital Channel gadget needs to access https 443 ports on the internet. The URI used will vary depending on the region.

- https://*.imiengage.io/
- https://*.imiengage.com/
- https://*.imiengage.com.au/
- https://*.imiengage.net/
- https://*.imichat.us/
- https://*.imichat.io/
- https://*.imi.chat/

For more information about the digital channels integration, see the *Digital Channels Integration Using Webex Connect* chapter in the Cisco Unified Contact Center Enterprise Features Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html>.

Prerequisites to configure the Manage Digital Channels gadget

Before you configure the Manage Digital Channels gadget for your agents and supervisors, make sure that you meet the following prerequisites:

- You have installed or upgraded to Cisco Finesse version 12.6(2). For more information, see Cisco Finesse Installation and Upgrade Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-installation-guides-list.html>.
- You have configured the Cloud Connect server 12.6(2) in Finesse. For more information, see the *Cloud Connect Server Settings* section in the Cisco Finesse Administration Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html>.
- You have integrated Cloud Connect with Webex Connect. For more information, see the *Integrate Cloud Connect with Webex Connect* section in the following documents:
 - Cisco Unified Contact Center Enterprise Features Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html>
 - Cisco Packaged Contact Center Enterprise Features Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html>
- The Manage Digital Channels gadget must be available in the configured layout for the agents and supervisors when they log in.
- The Finesse desktop needs internet connectivity with a maximum latency of 400 milliseconds and minimum bandwidth of five mbps (5 mbps) or better. This is required to load the Manage Digital Channels gadget and associated functionalities that are cloud-hosted services.
- The following components must be on release 12.6(2):
 - CCE components (Router, Logger, AW, and PG), Cisco Finesse, and Cisco IdS server.
- Ensure that agents are members of the skill group that is associated to the non-voice MRD, which is linked to the system-defined application path (suffixed with "UQ.Desktop") so that the Manage Digital Channel gadget recognizes that the agents are configured to interact over non-voice channels that Finesse

manages. For more information, see the *Media Routing Domains* section in the *Scripting and Media Routing Guide for Cisco Unified ICM/Contact Center Enterprise*.

- You have configured the Finesse Digital Channel State Control on the Finesse desktop. For more information on configuring digital channel-specific information, see *Channel Service* section in the *REST API Developer Guide*.
- Ensure that the Contact Center Enterprise is running in Single Sign-On (SSO) or Hybrid mode because the Manage Digital Channels gadget is supported only for SSO mode login. For more information about SSO, see the *Single Sign-On* chapter in the following documents:
 - Cisco Unified Contact Center Enterprise Features Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html>
 - Cisco Packaged Contact Center Enterprise Features Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html>
- Ensure that you always have the agent details synchronized between CCE and Webex Engage for the agents to be able to log in to the Manage Digital Channels gadget. For instructions about how to synchronize the agents between CCE and Webex Engage, see the *Synchronize CCE agents to Webex Engage* section in the following documents:
 - Cisco Unified Contact Center Enterprise Features Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html>
 - Cisco Packaged Contact Center Enterprise Features Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html>
- You have defined the specific set of Expanded Call Context (ECC) variables to assist the agents with relevant information about the calls that they are handling. For more information, see the *Define ECC variables* section in the following documents:
 - Cisco Unified Contact Center Enterprise Features Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html>
 - Cisco Packaged Contact Center Enterprise Features Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html>
- You have generated the public key certificate using Cisco Identity Service (IdS) 12.6(2) or higher, to establish a secure connection between the Manage Digital Channels gadget that is available in Cisco Finesse and the Webex Engage that is hosted on the cloud data center. For more information, see the *Generate public key certificate using Cisco IdS* section in the following documents:
 - Cisco Unified Contact Center Enterprise Features Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html>
 - Cisco Packaged Contact Center Enterprise Features Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html>
- Ensure that all agents have valid phone numbers (extension numbers) to log in to the Finesse Desktop, even if they have signed up for digital channels interaction only.
- Ensure that the **utils finesse user_signout_channel type** CLI is used to sign out the users from either all media channels or voice and task routing channels. For information on how to use this CLI, refer to the [Signout from Media Channels](#) topic.

Add Manage Digital Channels gadget

This procedure explains how to add **Manage Digital Channels** gadget in **Cisco Finesse Administration**.

- Step 1** Sign in to the administration console on the primary Cisco Finesse server using the URL: *https://FQDN of Finesse server:8445/cfadmin*.
- Step 2** Click **Desktop Layout**.
- Step 3** Select from the following editors:
- Text Editor
 - XML Editor
- Step 4** Sample configurations for customizing manage digital channel properties are added to the default layout (Desktop Layout) and team-specific layout (Team Resources > Desktop Layout). For upgraded layouts, sample configurations for customizing manage digital channel properties do not appear by default. For the gadget to appear in the Agent Desktop, the administrator must copy the XML code below from the View Default Layout, add it to the respective custom layouts, and uncomment the tab.

```
<!--tab>
    <id>manageDigitalChannel</id>
    <icon>mui-chat-active_16</icon>
    <label>finesse.container.tabs.agent.manageDigitalChannelLabel</label>
    <columns>
        <column>
            <gadgets>
                <gadget maxDialogLimit="5" interruptAction="IGNORE"
dialogLogoutAction="TRANSFER">/3rdpartygadget/files/ManageDigitalChannel/ManageDigitalChannel.xml</gadget>
            </gadgets>
        </column>
    </columns>
</tab-->
```

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

For more information about integrating digital channels using Webex Connect, see the *Digital Channels Integration using Webex Connect* chapter in the following documents:

- Cisco Unified Contact Center Enterprise Features Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html>
- Cisco Packaged Contact Center Enterprise Features Guide at <https://www.cisco.com/c/en/us/support/customer-collaboration/packaged-contact-center-enterprise/products-maintenance-guides-list.html>

Manage Digital Channels gadget properties

Through the desktop layout in the Finesse administrator console, you can configure the gadget properties using the login parameters, by passing the properties described below, as explained in Step 3 of the [Customize Gadget Properties](#) section, within the gadget tag.

The following table lists the properties and action parameters required by the gadget:

Property	Description
maxDialogLimit	<p>Represents the default value for the maximum number of concurrent tasks that are routed to the Agent in any of the multiple Unified CCE Media Routing Domains (MRD) which is used to represent the digital channels on the desktop.</p> <p>The default value for this is 5. Any value from 1 through 10 is allowed. The default value can be overridden by the media-specific property for the same as explained in the maxDialogLimit-mrd_id property.</p>
interruptAction	<p>This value sets the default behavior for all underlying MRDs that represent the digital channels unless overridden by MRD specific behavior. The allowed values are ACCEPT and IGNORE.</p> <p>The default is IGNORE. This default can be overridden by the media specific property for the same as explained below.</p>
dialogLogoutAction	<p>The action to be performed if agents terminate the desktop session when there are active tasks being handled by that agent. The possible values are CLOSE and TRANSFER (default). CLOSE will terminate the tasks and TRANSFER will cause the task to be re-routed to the same script selector or dialled number used to inject the task.</p>
maxDialogLimit-mrd_id	<p>Modifies the maxDialogLimit for the Media Routing Domain (MRD) represented by mrd_id. The possible values and limits are the same as specified above for maxDialogLimit parameter.</p> <p>For example, maxDialogLimit-5000=10. In this case, the property modifies the maxDialogLimit for the MRD with ID 5000 to 10.</p>
interruptAction-mrd_id	<p>Modifies the interruptAction for the Media Routing Domain (MRD) represented by mrd_id. The possible values and limits are the same as specified above for interruptAction parameter.</p> <p>For example, interruptAction-5000=IGNORE. In this case, the property modifies the interruptAction for the MRD with ID 5000 to IGNORE.</p>

Property	Description
dialogLogoutAction-mrd_id	<p>Modifies the dialogLogoutAction for the Media Routing Domain (MRD) represented by mrd_id. The possible values and limits are the same as specified above for dialogLogoutAction parameter.</p> <p>For example, dialogLogoutAction-5000=CLOSE. In this case, the property modifies the dialogLogoutAction for the MRD with ID 5000 to CLOSE.</p>
digitalChannelSignInFailureMaxRetries	<p>Defines the maximum number of retries to be attempted after a digital channel login fails.</p> <p>The default value is 1.</p>
digitalChannelSignInFailureRetryDelay	<p>Defines the time delay between two failed login retry attempts.</p> <p>The default value is 60000 milliseconds (1 minute).</p>
loadingConversationFailureMaxRetries	<p>Due to initialization delays, digital tasks loaded from the internet hosted service could at times fail. The gadget is expected to retry to load the conversations in such scenarios.</p> <p>This defines the maximum number of retries attempted by the gadget to load the conversations of the selected tasks (email, chat, sms, or any other media).</p> <p>The default value is 4.</p>
loadingConversationFailureRetryDelay	<p>Defines the time delay between retrying to load the tasks.</p> <p>The default value is 3000 milliseconds.</p>
gadget-version	<p>Allows you to select the version of the Manage Digital Channels gadget that has to be rendered on the Finesse desktop.</p> <p>The default value of this property is LATEST. This value ensures that the desktop always gets the latest version of the gadget deployed on cloud.</p> <p>You can access the list of published versions using the https://wxcce-digital.cisocccservice.com/LATEST/ChangeLog.txt URI.</p>



Note If any of the above property is either invalid or unavailable, then default values are considered as parameters for media login. For non-interruptable MRDs, the interruptAction passed can only be IGNORE. If the value is anything other than this, it is not considered and IGNORE is used by default.

Once the agents are signed into Finesse, the CCE assigns tasks to these agents in the Media Routing Domain (MRD). The agents can then perform actions, including transferring non-voice dialogs and accepting or ignoring interrupts. Each dialog represents a task and an agent can handle a predefined maximum number of concurrent dialogs.

If the agent is interrupted by a task or call when handling a dialog, then Finesse can either:

- **ACCEPT** the interrupt—in which case the agent is put into **INTERRUPTED** state and cannot work further on dialogs in the interrupted MRD.
- **IGNORE** the interrupt—in which case the agent's state does not change and the agent can continue to work on the dialogs in the MRD.

If the agent decides to log out of any channel (media or voice), then the active dialogs associated with the agent are either transferred or closed. To transfer tasks, Finesse resubmits the tasks into the system as new tasks. The agent can log out from:

- only voice dialogs
- all Finesse media (recommended)

For voice calls, data is fetched from the phone book. For media other than voice, a script selector or route point is used, wherein the value of the script selector variable maps to the value of the script selector that is created through Unified ICM Configuration Manager to fetch the data. These script selectors or route points are configured in CCE for the particular MRD against which the task was sent. For more information on script selectors, refer to *Script Administration* chapter of *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/series.html#%7Etab-documents>.

The following is a sample gadget configuration done on Finesse Admin console:

```
<gadget maxDialogLimit="3"
  maxDialogLimit-5000="2"
  maxDialogLimit-5003="1"
  interruptAction="IGNORE"
  interruptAction-5000="IGNORE"
  interruptAction-5000="ACCEPT"
  dialogLogoutAction="CLOSE"
>/3rdpartygadget/files/ManageDigitalChannel/ManageDigitalChannel.xml</gadget>
```

From the above configuration, you can observe that all the properties of the gadget are added as key-value pairs in the form of (propertyname-mrd), instead of the url parameters. In this configuration, **maxDialogLimit=3** and the **maxDialogLimit-5000=2** is considered for chat MRD. If the per-MRD property is not specified, the global **maxDialogLimit** is considered. If none of these properties are specified for the Manage Digital Channels gadget, **maxDialogLimit** is considered as **5**, **interruptAction** as **IGNORE**, and **dialogLogoutAction** as **CLOSE**.



Note If agents don't accept the incoming interactions within the predefined time, the interactions get routed to the next available agent and the agents' state changes to **Not Ready** automatically.

Popover information

The agent dashboard is displayed with the ECC variables corresponding to the layout transmitted in `user.layout`. If `user.layout` ECC variable is not present in the task presented to the agent, then the default ECC variable layout configured for voice will be used to display the popover information. The configuration steps for creating a call variable layout and specifying the variables that are used for popover and callheader for a task is the same as that followed for voice calls. For more information, refer to [Manage Call Variables Layouts](#).



Note To display a custom Call Variables Layout, in the Unified CCE routing script set the `user.Layout` ECC variable to the name of a configured Call Variables Layout. If the digital channel tasks forwarded to the Finesse Desktop do not contain the `user.Layout` variable with the custom layout name, or if the specified custom layout is not found, then the digital channel gadget displays the task using the default layout configured for voice. The variable configured as the call header is also used in the Task List and Task.

Methods to collect Serviceability logs

If an agent experiences problems with the Manage Digital Channels gadget, the agent can send a set of desktop logs to the Cisco Finesse server. These desktop logs are also called as serviceability logs. You can send the serviceability logs in the following ways:

- **Send Error Report**

For more information on using the **Send Error Report**, see the *Common Tasks* chapter in the [Cisco Finesse Agent and Supervisor Desktop User Guide](#).

- **Log Collection Schedule**

For more information on using the log collection schedule, see the [Log Collection Schedule](#) section.

- **SysLogs Support**

Cisco Finesse generates syslogs for various failure scenarios.

For more information about the list of syslogs that Finesse generates for the Manage Digital Channels gadget, see the [Syslog Support for Critical Log Messages](#) section.

Configure Multi-Tab Gadget Layout

You can configure Multi-Tab gadgets in the Cisco Finesse desktop layout with the following steps:



Note Multi-Tab gadgets can be configured to include Agent Answers gadget. For more information about Agent Answers, see the *Contact Center AI Gadgets* section.

Step 1 Click **Desktop Layout**.

Step 2 Select from the following editors:

- Text Editor

- XML Editor

Step 3 Add the `<gadget id="uniqueId">/desktop/scripts/tabbedgadgets.js</gadget>` code snippet in the existing layout, to configure a Multi-Tab gadget. The **gadget id** must be a unique variable within the desktop layout.

The following figures explain the configuration of Multi-Tab gadgets in Cisco Finesse Agent and Supervisor desktops.

Figure 1: Default layout

```
<page>
  <gadget>/desktop/scripts/js/callcontrol.js</gadget>
  <gadget>/3rdpartygadget/files/CXService/CiscoCXJourneyGadget.xml</gadget>
  <gadget>/3rdpartygadget/files/ccaiGadgets/answersGadget.xml</gadget>
</page>
```

Figure 2: Multi-Tab Gadget layout

Queue Name	# Calls	Max Time	Ready	Not Ready	Active			Wrap Up	
					In	Out	Other	Ready (Pending)	Not Ready (Pending)
0001.SG	0	00:00:00	2	0	0	0	0	0	0
0002.SG	2	00:00:00	2	2	0	0	0	0	0
0003.SG	5	00:00:00	4	1	0	0	0	0	0
0004.SG	3	00:00:00	5	1	0	0	0	0	0
0006.SG	0	00:00:00	6	3	0	0	0	0	0
0007.SG	2	00:00:00	8	2	0	0	0	0	0
0008.SG	5	00:00:00	1	2	0	0	0	0	0
0009.SG	3	00:00:00	2	2	0	0	0	0	0
0010.SG	1	00:00:00	4	2	0	0	0	0	0
0011.SG	2	00:00:00	6	1	0	0	0	0	0
0012.SG	2	00:00:00	2	1	0	0	0	0	0
0013.SG	2	00:00:00	1	1	0	0	0	0	0

```

<page>
  <gadget>/desktop/scripts/js/callcontrol.js</gadget>
  <gadget id="MultiTabGadget1">/desktop/scripts/js/tabbedGadgets.js</gadget>
  <gadget managedBy="gadgetmanager1">/desktop/scripts/js/queueStatistics.js</gadget>
  <gadget
managedBy="MultiTabGadget1">/3rdpartygadget/files/CXService/CiscoCXJourneyGadget.xml</gadget>
  <gadget managedBy="MultiTabGadget1">/3rdpartygadget/files/ccaiGadgets/answersGadget.xml</gadget>
</page>

```

There are no restrictions on the number of gadgets that can be configured within a Multi-Tab gadget. If more than seven gadgets are configured, a drop-down icon appears next to the seventh gadget tab. When clicked, it displays a drop-down list of the remaining gadgets (starting from the eighth gadget). The maximum number of visible gadget tabs displayed in the Multi-Tab gadget header is configured through the desktop property `maxTabsOnmulti-tabGadgetHeader` or by using the CLI command: `utils finesse set_property desktop maxTabsOnTabbedGadgetHeader [value]`. For more information, see *Maximum Number of Visible Multi-Tab Gadget Tabs* at [Desktop Properties](#), on page 208.

Note More than seven gadget tabs can be visible if you use a resolution higher than the default desktop resolution of 1366x768.

Adding Page-Level Gadgets in a Single Multi-Tab Gadget

Multi-Tab gadgets can contain both page-level and other gadgets. The page-level gadgets contained in a Multi-Tab gadget are always visible irrespective of the selected desktop container tab. However, the gadgets that are not configured as page-level gadgets are hidden or displayed depending on the selected desktop container tab.

Note

- If a Multi-Tab gadget is not configured as a page-level gadget, all the gadgets that are managed by the Multi-Tab gadget should also be present in the same desktop container tab.
- If a Multi-Tab gadget is configured as a page-level gadget but has no gadgets assigned, the Multi-Tab Gadget is not visible on the desktop.
- If a Multi-Tab gadget is configured as a non page-level gadget but has no gadgets assigned, the Multi-Tab Gadget displays a warning message for desktop users. You must avoid this configuration for an optimal Finesse desktop user experience.

Drag-and-Drop and Resize Gadget or Component

The administrator can configure the drag-and-drop and resize gadget or component features for agents and supervisors to customize their Finesse desktop.

- The drag-and-drop feature allows agents and supervisors to drag (and drop) the gadget or the component to the required position on the desktop layout.
- The resize feature allows the agents and supervisors to shrink or expand the gadget or the component to a custom size on the desktop layout.
- Each desktop tab can be customized to have a unique layout without affecting other tabs.
- When you resize the browser, based on the width of the browser, gadgets in the desktop layout are automatically rearranged one below the other to fit the available screen area.

**Note**

By default, the drag-and-drop and resize features are disabled. The administrator must set the `enableDragDropAndResizeGadget` desktop property value as `true` to enable these features.

The administrator can customize the desktop property value of these features through the desktop layout:

- **Default layout (Desktop Layout)**—In the **Text Editor**, remove the comment (`<!--and -->`) from the `enableDragDropAndResizeGadget` code snippet and enter the value as `true` to add these features to the desktop layout. For more information, see [Customize Desktop Properties, on page 72](#).

The following is the sample code snippet, as displayed in the default **Desktop Layout**.

```
<!-- <config key="enableDragDropAndResizeGadget" value="false"/> -->
```

- **Team-specific layouts (Manage Team Resources > Desktop Layout)**—Select a specific team and then in the **Text Editor**, remove the comment (`<!--and -->`) from the `enableDragDropAndResizeGadget` code snippet and enter the value as `true` to add these features to the team desktop layout. For more information, see [Customize Desktop Properties at Team Level, on page 121](#).

The following is the sample code snippet, as displayed in the team **Desktop Layout**.

```
<!-- <config key="enableDragDropAndResizeGadget" value="false"/> -->
```

**Note**

- For upgraded layouts, the sample configuration for customizing desktop property (`enableDragDropAndResizeGadget`) doesn't appear by default in the **Desktop Layout**. Administrators must copy the XML from the **View Default Layout** and add to the respective custom layouts.
- For new layouts, the sample configuration for customizing desktop property (`enableDragDropAndResizeGadget`) appears by default in the **Desktop Layout**.
- The administrator can also use the CLI and set the **utils finesse set_propertydesktop enableDragDropAndResizeGadget** to *true* to enable these features. For more information see *Desktop Properties*.
- If the drag-and-drop and resize gadget feature is enabled, the maximize or collapse functionality isn't available in the Multi-Tab gadgets. Drag and reorder of gadgets inside a Multi-Tab gadget is independent of the `enableDragDropAndResizeGadget` property.
- If the property value is defined in the team-specific desktop layout (**Manage Team Resources > Desktop Layout**), the team-specific desktop layout takes precedence over the property value defined in the **Desktop Layout** and CLI.
- These features aren't applicable for gadgets that don't have a defined title. For more information, see the *Gadget Limitations* section in the *Cisco Finesse Web Services Developer Guide* at <https://developer.cisco.com/docs/finesse/#!/rest-api-dev-guide>.

For more information, see the *Cisco Finesse Agent and Supervisor Desktop User Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-user-guide-list.html>.

Drop Participants from Conference

**Note**

Cisco Finesse now permits an agent or supervisor, who is the participant in a conference call, to drop another participant. The extended capability must be enabled at the `Dialog-Drop Participant from Conference` API and desktop level. By default, this feature is available only for supervisors.

The administrator can set permission to allow an agent or a supervisor, who is the participant in a conference call using the Finesse desktop, to drop other agents, supervisors, or non-agents from the conference call.

The desktop property to drop participants from a conference call are:

- `enableDropParticipantFor`—This property is used to set permissions to allow the agents, supervisors, or both to drop other participants from the conference call.

The possible values for the `enableDropParticipantFor` property are:

- `supervisor_only`—(default value) Only the supervisor, who is a participant of the conference call, can drop other agents in the conference call.
- `conference_controller_and_supervisor`—The supervisor who is a participant of the conference call or an agent who initiated the conference call (conference controller) can drop other participants.
- `all`—Any agent or supervisor who is a participant of the conference call can drop other participants.

For example, if the permission is set to be *supervisor_only*, when the supervisor barges in to a call between an agent and a customer, the supervisor is the only one who can make a request to drop the agent from the call. This leaves the supervisor on the call with the customer.

- **dropParticipant**—This property controls the listing of agents and non-agents in the **Drop** options list of participants in the Finesse desktop.

The possible values for the **dropParticipant** property are:

- **agents**—(default value) The agents or supervisors who are participants of the conference call are displayed in the **Drop** options list of participants in the Finesse desktop.
- **all**—The agents, supervisors, CTI Route Point, IVR port, a device to which no agent is signed in, or a caller device in the conference call are displayed in the **Drop** options list of participants in the Finesse desktop.

For more information, see the *Intercept a Call* section in the *Common Tasks* chapter of *Cisco Finesse Agent and Supervisor Desktop User Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-user-guide-list.html>.



Note To enable the supervisor or call controller to drop an unmonitored extension in Cisco Unified CCE, in Release 12.0(1) or higher, set the **DropAnyPartyEnabled** registry key to *1* in the Dynamic Registry of the CTI server. The supervisor cannot drop a CTI Route Point, IVR port, a device to which no agent is signed in, a caller device, or other agents for whom SILENT_MONITOR is not initiated by the supervisor.

For more information, see the *Enable Dropping Call Participants from a Conference Call* section in *Cisco Contact Center Gateway Deployment Guide for Cisco Unified ICM/CCE* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-programming-reference-guides-list.html>.

The administrator can customize the desktop property value for this feature through the desktop layout:

- **Default layout (Desktop Layout)**—In the **Text Editor**, remove the comment (`<!--and -->`) from the `enableDropParticipantFor` and `dropParticipant` code snippets. Enter the appropriate value against the code snippet to control the feature using desktop layout. For more information, see [Customize Desktop Properties, on page 72](#).

The following is the sample code snippet, as displayed in the default **Desktop Layout**.

```
<!-- <config key="enableDropParticipantFor" value="supervisor_only"/> -->
<!-- <config key="dropParticipant" value="agents"/> -->
```

- **Team-specific layouts (Manage Team Resources > Desktop Layout)**—Select a specific team layout, and then in the **Text Editor**, remove the comment (`<!--and -->`) from the `enableDropParticipantFor` and `dropParticipant` code snippets. Enter the appropriate value against the code snippet to control the feature using team desktop layout. For more information, see [Customize Desktop Properties at Team Level, on page 121](#).

The following is the sample code snippet, as displayed in the team **Desktop Layout**.

```
<!-- <config key="enableDropParticipantFor" value="supervisor_only"/> -->
<!-- <config key="dropParticipant" value="agents"/> -->
```

**Note**

- The administrator can also use the CLI **utils finesse set_property webservice enableDropParticipantFor** and set permission to allow an agent or a supervisor, who is the participant in a conference call to drop other agents, supervisors, or non-agents from the conference call. Enabling this capability at a an API level is a prerequisite for the desktop property changes. For more information, see [Service Properties, on page 220](#).
- The `Dialog-Drop Participant` from Conference API allows an agent or supervisor to make a request to drop other agents, supervisors, or non-agents from a conference based on the permission set by the administrator. By default, this API is only available for supervisors (based on the `supervisor_only` property); but the administrator can use the Finesse CLI to expand access. For more information, see <https://developer.cisco.com/docs/finesse/#!/rest-api-dev-guide>.
- The `Dialog-Drop Participant` from Conference API and the team-specific layout must have the same value to have the expected behavior, as defined by the Finesse desktop property. If the values differ, then the most restrictive value takes precedence. The order is *all*, *conference_controller_and_supervisor*, and *supervisor_only*.

For example, at the API level, it is restricted to be *supervisor_only*, and the Finesse desktop level restriction at team-specific layout is *all*, then the restriction at API level applies, as that is more restrictive than the desktop level.
- For upgraded layouts, sample configuration for customizing desktop property (`enableDropParticipantFor` and `dropParticipant`) does not appear by default in the **Desktop Layout**. The administrator must copy the respective code snippets of the XML from the **View Default Layout** and add to the respective custom layouts.
- For new installations, sample configuration for customizing desktop property (`enableDropParticipantFor` and `dropParticipant`) appears by default in the **Desktop Layout**.

Limitation

Finesse desktop cannot distinguish agents from another peripheral gateway (PG) as agents. So the agents from another PG are displayed using the same icon used for non-agents (👤) in the Finesse desktop.

Customize Desktop Properties

You can customize the Finesse desktop properties.

-
- Step 1** Click **Desktop Layout**.
 - Step 2** Select from the following editors:
 - **Text Editor**
 - **XML Editor**
 - Step 3** Enter the desktop property name in the config key tag.
 - Step 4** Enter the possible value of the desktop property in the value tag.

The following are the sample desktop property entries, as displayed in the default **Desktop Layout**. To change these desktop property entries in **Text Editor**, remove the comment (`<!--` and `-->`) and set appropriate values.

```
<!-- <config key="enableDragDropAndResizeGadget" value="false"/> -->
<!-- <config key="wrapUpCountDown" value="true"/> -->
<!-- <config key="desktopChatAttachmentEnabled" value="true"/> -->
<!-- <config key="forceWrapUp" value="true"/> -->
```

Note If the property value is defined in the **Desktop Layout**, then the **Desktop Layout** value takes precedence over the property value defined using the CLI. For more information on Finesse CLIs, see *Desktop Properties*.

The following table lists the supported desktop properties:

Config Key	Value	Default Value
enableDragDropAndResizeGadget	true false	false
enableShortCutKeys	true false	true
forceWrapUp	true false	true
wrapUpCountDown	true false	true
showWrapUpTimer	true false	true
desktopChatAttachmentEnabled	true false	true
desktopChatMaxAttachmentSize	Range: 1—10 (MB)	5
desktopChatUnsupportedFileTypes	Unsupported file formats include comma-separated valid file extensions. For example: .exe, .sh	.exe, .msi, .sh, .bat
showAgentHistoryGadgets	true false	true
showActiveCallDetails (for Supervisor Only)	true false	true
pendingDTMFThresholdCount	Range: 1—20	20
dtmfRequestTimeoutInMs	Range: 1000—200000 (1 to 200 seconds)	5000 (5 seconds)
enableDropParticipantFor	supervisor_only conference_controller_and_supervisor all	supervisor_only
dropParticipant	agents all	agents

For more information on Finesse desktop properties, see *Desktop Properties*.

Step 5 Click **Save**.

The change takes effect when the agent or supervisor refreshes the Finesse desktop or sign out and sign in again.

Note If you clear the **Override System Default** check box and click **Save**. The changes are overwritten, and the editing pane reverts to the default desktop layout XML.

Customize Gadget Properties

You can customize the gadget properties to control the appearance and behavior of the gadget. The gadget behavior can be customized on a per team basis by customizing their behavior in team specific layout. This can be done by customizing the properties that a gadget supports, which are typically configured statically in the UserPrefs section of the gadget XML, into the gadget tag of the respective gadget, as shown in Step 3 below. By modifying the desktop layout for each team, the gadget properties can be customized for the required team.

To customize the gadget properties:

Step 1 Click **Desktop Layout**.

Step 2 Select from the following editors:

- **Text Editor**
- **XML Editor**

Step 3 Find the gadget for which the property has to be modified. For example,

```
<gadget>/3rdpartygadget/files/ManageDigitalChannel/ManageDigitalChannel.xml</gadget>
```

Step 4 Add the required property modifications within the gadget tag as shown below:

```
<gadget dialogLogoutAction="TRANSFER"
maxDialogLimit = "10">
/3rdpartygadget/files/ManageDigitalChannel/ManageDigitalChannel.xml</gadget>
```

Note For more information on the list of supported gadget properties, refer to the respective gadgets. To customize the Manage Digital Channels gadget properties, see [Manage Digital Channels gadget properties](#).

Step 5 Click **Save**.

The change takes effect when the agent or supervisor refreshes the Finesse desktop or sign out and sign in again.

Note If you clear the **Override System Default** check box and click **Save**, the changes are overwritten and the editing pane reverts to the default gadget layout XML.

Horizontal Header

The Horizontal Header on the Finesse desktop has the following components from left to right. All these components can be removed and replaced with custom gadgets as required.

- **Logo:** Default is Cisco logo. Can be customized.

- **Product Name:** Default is Cisco Finesse. Can be customized.
- **Agent State for Voice:** Displays agent state for voice call.
- **Agent State for Digital Channels:** Displays agent state for digital channels.
- **Dialer Component:** Agent can make a new call.
- **Identity Component:** Displays agent name and signout functionality with reason codes.



Note The sum of widths set for all gadgets and components in the header (inside right aligned columns and left aligned columns) should not exceed the total header width. If it exceeds the header width, some of the gadgets/components will not be visible.

Customize Title and Logo in the Header

You can customize the title and logo displayed on the Finesse desktop:

Step 1 Click **Desktop Layout**.

Step 2 Select from the following editors:

- **Text Editor**
- **XML Editor**

Step 3 Enter the product name in the config value tag with title key.

Step 4 Upload the logo file just like any third-party gadget.

For more information, see section *Upload Third-Party Gadgets* in *Cisco Finesse Admin Guide*.

Step 5 Enter the URL of the logo file in the config value tag with logo key.

Example:

```
<configs>
  <!-- The Title for the application which can be customised.-->
  <config value="product.full-name" key="title"/>
  <!-- The logo file for the application-->
  <!--<config key="logo" value="/3rdpartygadgets/<some_sample_image>"-->
</configs>
```

The customized logo and product name is displayed on the Finesse desktop.



Note The file size that can be uploaded for the logo must be kept within 40 pixels. The file types supported are .svg, .png, .gif, and .jpeg/jpg.

alternateHosts Configuration

The `<gadget>` element in the Finesse Layout XML provides an attribute to specify alternate hosts from which the gadget can be loaded. This allows the Cisco Finesse desktop to load the gadget using a different host if the primary server is unavailable.

The **alternateHosts** attribute contains a comma-separated list of FQDNs that will be used if the primary-host-FQDN is unavailable.

```
<gadget alternateHosts="host1,host2,host3,...">
  https://<primary-host-FQDN>/<gadget-URL>
</gadget>
```

The **alternateHosts** attribute is only applicable for gadgets with an absolute URL. That is URLs containing the FQDN of a host, an optional port, and the complete URL path to the gadget. For example: `<gadget alternateHosts="host1,host2">https://primary host/relative_path</gadget>`

If loading the gadget from the primary-host fails, the Cisco Finesse container attempts to load the gadget from the alternate hosts in the order specified in the **alternateHosts** attribute.

The Cisco Finesse desktop may fail to load the gadget even if some of the hosts are reachable. In such cases, refresh the Cisco Finesse desktop.

When the gadget is specified with a relative URL, for example: `<gadget >/3rdpartygadgets/relative_path</gadget>`, the **alternateHosts** attribute does not apply and is ignored by the Cisco Finesse desktop.



Note If the host serving the gadget fails after the Cisco Finesse desktop was successfully loaded, the desktop must be refreshed in order to load the gadget from an alternate host. The gadget does not implement its own failover mechanism.

Headless Gadget Configuration

Headless gadgets are gadgets which do not need a display space, but can be loaded and run like a background task in the browser. The **Hidden** attribute (optional) is used to support headless gadgets in the layout XML. When an attribute is set to "hidden=true", then the gadget is loaded by the container, but will not be displayed. The default value set for the attribute is "false".

Customize Icons in Left Navigation Bar

You can add icons (both custom and inbuilt) to the collapsible left navigation bar of the Finesse desktop:

Step 1 Click **Desktop Layout**.

Step 2 Select from the following editors:

- **Text Editor**
- **XML Editor**

Step 3 Enter name of the gadget or component in the id tag.

Step 4 Enter the value of the icon in the icon tag.

Step 5 Upload the icon file just like any third-party gadget.

For more information, see section *Upload Third-Party Gadgets* in *Cisco Finesse Admin Guide*.

Note When adding a custom icon, provide the path in the icon tag and if you are adding an inbuilt icon, provide the icon value in the icon tag

Example:

```
<tab>
  <id>myHistory</id>
  <icon>/3rdpartygadgets/<some_sample_image>
  <label>finesse.container.tabs.agent.myHistoryLabel</label>
  <columns>
    <column>
      <!-- The following gadgets are used for viewing the call history and state history
of an agent. -->
      <gadgets>
        <gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget
.xml?gadgetHeight=280&viewId=5FA44C6F930C4A64A6775B21A17EED6A&
filterId=agentTaskLog.id=CL%20teamName</gadget>
        <gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget
.xml?gadgetHeight=280&viewId=56BC5CCE8C37467EA4D4EFA8371258BC&
filterId=agentStateLog.id=CL%20teamName</gadget>
      </gadgets>
    </column>
  </columns>
</tab>
```

Note The file size that can be uploaded in the left navigation bar as custom icons is 25 pixels by 25 pixels. The maximum width of the tab title in the left navigation bar must be 80 pixels or less. The file types supported are .svg, .png, .gif, and .jpeg/jpg.

Customize Icons for Gadgets





















As part of the Cisco Finesse container, various standard icons are available. Use the following procedure to customize the icons for the gadgets hosted in Finesse desktop.

Step 1 Click **Desktop Layout**.

Step 2 Select from the following editors:

- **Text Editor**
- **XML Editor**

Step 3 Enter the value of the icon in the icon tag. Get the icon name from the [List of Icons, on page 79](#). The icon name is located on the right of the icon image. For example, search.

	search ← Icon Name		remove-contain
	dial		remove-outline
	keyboard		close
	close-keyboard		exit-contain
	delete		exit-outline
	trash		refresh
	add		more
	add-contain		sign-in
	add-outline		forced-sign-in
	Remove / Delete		sign-out

Note Icon name is case sensitive. Enter the icon name exactly as displayed in the [List of Icons, on page 79](#).

Example





















An example of the desktop layout using the *Search* and *Close-Keyboard* icons.

```
<tab>
  <id>home</id>
  <icon>search</icon>
  <label>finesse.container.tabs.agent.homeLabel</label>
  <columns>
    <column>
      <gadgets>
        <gadget>/desktop/scripts/js/queueStatistics.js</gadget>
      </gadgets>
    </column>
  </columns>
</tab>
<tab>
  <id>sample</id>
  <icon>close-keyboard</icon>
  <label>finesse.container.tabs.agent.homeLabel2</label>
  <columns>
    <column>
      <gadgets>
        <gadget>/desktop/scripts/js/samplequeue.js</gadget>
      </gadgets>
    </column>
  </columns>
</tab>
```





















```
</columns>
</tab>
```

List of Icons













The following are the icons for Actions.

	search		remove-contain
	dial		remove-outline
	keyboard		close
	close-keyboard		exit-contain
	delete		exit-outline
	trash		refresh
	add		more
	add-contain		sign-in
	add-outline		forced-sign-in
	Remove / Delete		sign-out











The following are the icons for Audio.

	microphone		line-out-right
	mute		audio-settings
	mic-in		headset
	speaker		headset-cross
	speaker-cross		active-speaker
	volume-cross		locked-speaker
	audio-min		active-speaker-cross
	audio		bluetooth-contain-cross
	speaker-out-left		handset-cross
	line-out-left		headset-outline
























The following are the icons for Camera.

	video		zoom-in
	video-cross		zoom-out
	aux-camera		
	self-view		
	self-view-crossed		
	self-view-alt		
	web-camera		
	camera		
	swap-camera		
	swap-video-camera		

The following are the icons for Chat.

	chat
	chats
	persistent-chat
	comment
	waiting-silence
	broadcast-message
	invite
	send
	emoticons
	bot-outline







The following are the icons for Collaboration.

	schedule-add		leave-meeting		micro-blog
	day		community		timeline
	week		web-sharing		bookmark
	calendar-icon-date		mobile-presenter		chapters
	external-calendar		presentation		feedback
	instant-meeting		slides		like
	webex		point		
	meeting-room		extension-mobility		
	conference		participant-list		
	meet-me		browser		



















The following are the icons for Contacts.

510682


























510683

	contact
	add-contact
	remove-contact
	directory
	contact-card
	star

The following are the icons for Content.

	attachment		watchlist
	link		playlist
	document		prevent-download
	create-page		prevent-download-container
	move-page		download
	notes		download-contain
	image		upload
	folder		upload-contain
	export		share
	import		share-contain














The following are the icons for Editor.

	edit		screen-capture-square		view-feed-multi
	draw		popout		video-preview-telePresence
	transcript		filter		panel-slides-left
	annotation		picture-in-picture		panel-slides-right
	list-view		video-layout		print
	thumbnail-view		layout		
	text-format		view-side-by-side		
	text-color		view-stacked		
	text-size		view-feed-single		
	fullscreen		view-feed-dual		








The following are the icons for Email.

	email		send-email
	read-email		
	spam		
	inbox		
	outbox		
	sent		
	universal-inbox		
	arrow-right-tail		
	arrow-left-tail		
	reply-all		






The following are the icons for Hardware.

	display		power
	multi-display		dc-power
	soft-phone		ac-power
	video-input		power-contain
	computer		charging
	notebook-in		battery
	devices		
	idefix		
	mobile-phone		
	tablet		















The following are the icons for Media.

	image
	sound
	music
	graph
	text
	tables
	zip

The following are the icons for Navigation.






























	home		hamburger-menu
	android-home		way-nav
	right-arrow		right-arrow-contained
	right-arrow-contain		right-arrow-closed-contained
	right-arrow-outline		right-arrow-closed-outline
	touch		
	touch-point		
	touch-gesture		
	back		
	recent-apps		

The following are the icons for Network.

	wifi		signal-3
	proximity		signal-4
	proximity-not-connected		public-network
	bluetooth		private-network
	bluetooth-contained		
	bluetooth-outline		
	ethernet		
	no-signal		
	signal-1		
	signal-2		

































The following are the icons for Notifications and Alerts.

List of Icons

	warning		quality		location
	alert-badge		broken-image		compass
	error		blocked		flagged
	info		check		keywords
	help		certified		dms
	lock		bell		popup-dialog
	unlock		bell-cross		applications
	private		alarm		application
	privacy		running-application		default-app
	report		pin		













510892

The following are the icons for Phone.






























	calls		incoming-call		call-forward-divert		key-expansion-module
	other-phone		outgoing-call		merge-call		desk-phone
	call-log		missed-call		group-call		
	work		rtprx		hunt-group		
	desk-phone		rtptx		edit-call		
	voicemail		rtprx-rtptx-duplex		intercom		
	callback		speed-dial		intercom-whisper		
	redial		off-hook		intercom-duplex-connected		
	DND		alerting		forward-to-mobility		
	swap-calls		parked		transfer-to-mobile		

510893





The following are the icons for Sources.

	pc		sd
	disc		custom-desktop
	document-camera		
	whiteboard		
	general-source		
	disc-not-connected		
	document-camera-cross		
	whiteboard-cross		
	general-source-cross		
	usb		































































The following are the icons for Settings.

	settings		animation		reset
	sliders		accessibility		backup-data
	user		setup-assistant		bug
	admin		tools		lock-contain
	activities		hue		ground
	profile-settings		brightness		storage
	ringer-settings		volume		data-usage
	language		call-rate		numbered-inputs
	wallpaper		vibrate		numbered-outputs
	manage-cables		time		

The following are the icons for Video Controls.

	play
	play-container
	stop
	pause
	skip-fw
	skip-bw
	ffw
	fbw
	circle

































The following are the icons for Miscellaneous Icons.

	circle-bar chart		circle-pie chart		line-chart		D
	circle-column chart		unknown-customer		inbound-call		R
	dashboard		circle-note		outbound-call		RD
	circle-gauge		circle-custom-widget		call-back		SC
	circle-line chart		grid		phone-outline		SE
	event		bar-chart		chat-outline		VL
	social		bars		circle-grid		organization-setup
	web		text-and-font		drag-row		campaign-outbound
	node		report-view		edit-properties		desktop-agent
	formula		resize		key		
	maximize		manage-team		thumbs-down-outline		
	save		manage-call		thumbs-up-filled		
	history		analysis		thumbs-down-filled		
	minimize		analysis-active				
	tabs		manage-chat				
	vd-silent-monitoring		manage-email				
	time-arrow		reports-more				
	device-outSync		fb-chat				
	team-data		fb-group-chat				
	phone-cross		thumbs-up-outline				








































510898

510899







































List of Icons

	applause		folder		recurring		webhook
	at		highlighter		rotate-object-ccw		paired-audio
	at-contain		highlighter-check		rotate-object-cw		
	bot-one		highlighter-line		Spark		
	bot-two		integration		team-collapsed-view		
	bot-three		media-viewer		team-expanded-view		
	bot-four		paired-call		too-fast		
	cisco-logo		pencil		too-slow		
	feedback-clear		Q and A		video-group		
	feedback-result		raise-hand		video-tips		

510900

	arrow-back		asterisk		circle-analysis		content-share
	arrow-down		audio-broadcast		circle-care		data
	arrow-next		bottom		circle-location		device-inProgress
	arrow-up		chevron-down		circle-supervisor		device-inSync
	call-forwarding		chevron-left		circle-webex		diagnostics-active
	call-handling		chevron-right		clipboard		diagnostics
	care-filled		chevron-up		clock		edit-time
	chat-active		checkbox		cloud-active		end-call
	check-gear		circle-agent		cloud		endpoint-active
	check-refresh		eraser		company-active		

510901

	Euro		info-outline		panel-close		screen-capture
	help-outline		laser-pointer		pass-mouse		settings-active
	filter		left-arrow		plan-review		sort
	glyphicon-calendar		lightbulb		people-active		tools-active
	glyphicon-time		location-active		plugin		top
	grid-large		manage-recordings-tab		poll		user-chat
	grid-list		manage-recordings		priority		video-settings
	home-active		minus		plus		yen
	image-contain		new-call		question-circle		
	eraser		paired-call-outline		report-definition		

510902

For more information on customizing the visual experience, see *Visual Design Kit* at <https://developer.cisco.com/docs/finesse/#!visual-design-guide>.

XML Schema Definition

You must ensure that the XML uploaded conforms to the XML schema definition for Finesse. The XML schema definition for Finesse is as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="http://www.cisco.com/vtg/finesse" targetNamespace="http://www.cisco.com/vtg/finesse"
elementFormDefault="qualified">
  <!-- definition of version element -->
  <xs:element name="version">
    <xs:simpleType>
      <xs:restriction base="xs:double">
        <xs:pattern value="[0-9\.]+" />
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <!-- The below elements are for common desktop header and configs -->
  <!-- Copied from:
https://github5.cisco.com/cdu-shared/common-desktop/blob/master/java/layout-manager/src/main/resources/layoutSchema.xsd
-->
  <!-- If the common-desktop XSD changes, this too needs to be updated -->
  <!-- Only difference is that, column has been renamed to headercolumn, since column is
already there in finesse desktop layout -->
  <xs:complexType name="configs">
    <xs:sequence>
      <xs:element name="config" type="config" minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
```

```

<xs:complexType name="config">
  <xs:attribute name="key">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:pattern value="[a-zA-Z]*" />
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="value" type="xs:string" />
</xs:complexType>
<xs:complexType name="header">
  <xs:choice>
    <xs:sequence>
      <xs:element name="leftAlignedColumns" type="listOfColumns" minOccurs="1"
maxOccurs="1" />
      <xs:element name="rightAlignedColumns" type="listOfColumns" minOccurs="0"
maxOccurs="1" />
    </xs:sequence>
    <xs:sequence>
      <xs:element name="rightAlignedColumns" type="listOfColumns" minOccurs="1"
maxOccurs="1" />
    </xs:sequence>
  </xs:choice>
</xs:complexType>
<xs:complexType name="component">
  <xs:sequence>
    <xs:element name="url" type="xs:string" minOccurs="1" maxOccurs="1" />
    <xs:element name="stylesheet" type="xs:string" minOccurs="0" maxOccurs="1" />
  </xs:sequence>
  <xs:attribute name="id" use="required">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:pattern value="." />
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="order">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:pattern value="[0-9]{0,10}" />
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>
<xs:complexType name="listOfColumns">
  <xs:sequence>
    <xs:element name="headercolumn" type="headercolumn" minOccurs="1"
maxOccurs="unbounded" />
  </xs:sequence>
</xs:complexType>
<xs:complexType name="headercolumn">
  <xs:choice minOccurs="0" maxOccurs="1">
    <xs:element ref="gadget" />
    <xs:element name="component" type="component" />
  </xs:choice>
  <xs:attribute name="width">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:pattern value="[0-9]+(px|%)" />
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>
<!-- The above elements are for common desktop header and configs -->

```

```

<!-- definition of role type -->
<xs:simpleType name="role">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Agent" />
    <xs:enumeration value="Supervisor" />
    <xs:enumeration value="Admin" />
  </xs:restriction>
</xs:simpleType>
<!-- definition of simple elements -->
<xs:element name="id">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:pattern value="[a-zA-Z]([-_\.a-zA-Z0-9])*" />
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="label">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:minLength value="1" />
      <xs:pattern value="^[^\r\n]+" />
      <!-- This regex restricts the label string from carriage returns or newline
characters -->
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="icon" type="xs:anyURI" />
<xs:element name="gadget">
  <xs:complexType>
    <xs:simpleContent>
      <xs:extension base="restrictWhiteSpaces">
        <!-- <xs:attribute name="staticMessage" type="xs:string"/> -->
        <xs:attribute name="id">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:pattern value="[a-zA-Z]([-_a-zA-Z0-9])*" />
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="alternateHosts" type="xs:string" />
        <xs:attribute name="managedBy" type="xs:string" />
        <xs:attribute name="hidden" type="xs:boolean" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:element>
<xs:element name="role" type="role" />
<xs:element name="gadgets">
  <!-- Grouping of a set of gadgets -->
  <xs:complexType>
    <xs:sequence minOccurs="0" maxOccurs="unbounded">
      <!-- No limit to number of gadget URIs for now -->
      <xs:element ref="gadget" />
      <!-- URI of the gadget xml -->
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:simpleType name="restrictWhiteSpaces">
  <xs:restriction base="xs:anyURI">
    <xs:minLength value="1" />
    <xs:pattern value="\S+" />
    <!-- This regex restricts anyURI from containing whitespace within -->
  </xs:restriction>
</xs:simpleType>

```

```

<xs:element name="column">
  <!-- Grouping of a set of gadgets within a column -->
  <xs:complexType>
    <xs:sequence minOccurs="0" maxOccurs="unbounded">
      <!-- No limit to number of gadget URIs for now -->
      <xs:element ref="gadgets" />
      <!-- URI of the gadget xml -->
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="columns">
  <!-- Grouping of a set of columns -->
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="column" minOccurs="0" maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="page">
  <!-- Grouping of a set of persistent gadgets -->
  <xs:complexType>
    <xs:sequence minOccurs="0" maxOccurs="unbounded">
      <!-- No limit to number of gadget URIs for now -->
      <xs:element ref="gadget" />
      <!-- URI of the gadget xml -->
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="tab">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="id" />
      <!-- Id of the tab selector in the desktop -->
      <xs:element ref="icon" minOccurs="0" maxOccurs="1" />
      <xs:element ref="label" />
      <!-- Label of the tab selector -->
      <xs:choice>
        <xs:element ref="gadgets" minOccurs="0" maxOccurs="1" />
        <xs:element ref="columns" minOccurs="0" maxOccurs="1" />
      </xs:choice>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="tabs">
  <!-- Grouping of tabs -->
  <xs:complexType>
    <xs:sequence maxOccurs="unbounded">
      <!-- No limit to number of tabs for now -->
      <xs:element ref="tab" />
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="layout">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="role" />
      <!-- Type of the role -->
      <xs:element ref="page" />
      <!-- List of page gadgets -->
      <xs:element ref="tabs" />
      <!-- Grouping of tabs for this particular role -->
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

<xs:element name="finesseLayout">
  <!-- Layout of the desktop -->
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="version" />
      <xs:element name="configs" type="configs" minOccurs="0" maxOccurs="1" />
      <xs:element name="header" type="header" minOccurs="1" maxOccurs="1" />
      <xs:sequence maxOccurs="3">
        <!-- only support 3 roles for now -->
        <xs:element ref="layout" />
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:schema>

```

Live Data Reports

Prerequisites for Live Data

Before you add Live Data reports to the desktop, you must meet the following prerequisites:

- Download the Live Data reports from Cisco.com and import them into Cisco Unified Intelligence Center. Verify that the reports are working in Unified Intelligence Center.
- You must use HTTPS for both Cisco Unified Intelligence Center and Finesse. The default setting for both after a fresh installation is HTTPS.

For more information, see *Administration Console User Guide for Cisco Unified Intelligence Center* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-intelligence-center/products-maintenance-guides-list.html>.

- Ensure that user integration synchronization is enabled for Cisco Unified Intelligence Center.

For more information, see *Administration Console User Guide for Cisco Unified Intelligence Center* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-intelligence-center/products-maintenance-guides-list.html>.

- For HTTPS, you must upload security certificates to the Finesse, Cisco Unified Intelligence Center and Live Data servers. Finesse, Cisco Unified Intelligence Center, and Live Data are installed with self-signed certificates. However, if you use the self-signed certificates, agents and supervisors must accept certificates in the Finesse desktop when they sign in before they can use the Live Data gadget. To avoid this requirement, you can provide a CA certificate instead. You can obtain a CA certificate from a third-party certificate vendor or produce one internal to your organization.

Add Live Data Reports to Finesse

To add Live Data reports to the Finesse desktop. The procedure that you follow depends on several factors, described in the following table.

Procedure	When to use
Add Live Data reports to default desktop layout	After a fresh installation or after an upgrade if you have not customized the default desktop layout.

Procedure	When to use
Add Live Data reports to custom desktop layout	If you have customized the Finesse desktop layout.
Add Live Data reports to team layout	To the desktop layout for specific teams only.



Note The line breaks and spaces that appear in the example text of the procedures are provided only for readability and must not be included in the actual code.



Note After you add a gadget, sign in to the Finesse desktop and make sure it appears the way you want. If you use a report with a large number of rows, you may want to adjust the gadget height or the screen resolution on the computer used to access the desktop to make the report easier to read or make more rows appear on the screen without needing to scroll down.

Agents who are signed in when you change the desktop layout must sign out and sign back in to see the change on their desktops.

Add Live Data Reports to Default Desktop Layout

The Finesse default layout XML contains commented XML code for the Live Data report gadgets available for the Finesse desktop.

This procedure explains how to add the Live Data report gadgets to the default desktop layout. Use this procedure after a fresh installation of Finesse. If you upgraded Finesse and do not have a custom desktop layout, click **Restore Default Layout** on the Manage Desktop Layout gadget and then follow the steps in this procedure.

Step 1 Click **Desktop Layout**.

Step 2 Select from the following editors:

- **Text Editor**
- **XML Editor**

Step 3 Add the reports to the desktop layout:

- **Text Editor** - Remove the comment (`<!--` and `-->`) from each report that you want to add to the desktop layout and then replace “my-cuic-server” with the FQDN of the Cisco Unified Intelligence Center Server.
- **XML Editor** - To add a gadget, select `<layout > <tabs > <gadgets>` Add `<gadget>` and enter the required parameters such as `Id`, `alternate hosts`, `managed by`, `maximum rows`, and the hidden details.

Note You must select the reports that match the protocol (HTTPS) your agents use to access the Cisco Finesse desktop.

Step 4 Click **Save**.

Note In a dynamic type gadget, multiple `viewId` parameters is not supported. Check the URL in the error message before proceeding to save the default XML layout. The name value `"type=dynamic` must be part of the gadget URL.

Note If you select a TDM agent in the Team Performance Gadget, the recent state history data of the selected agent is not populated.

Add Live Data Reports to Custom Desktop Layout

The Finesse default layout XML contains commented XML code for the Live Data report gadgets available for the Finesse desktop.

To add the Live Data report gadgets to a custom desktop layout.

Step 1 Click the **Desktop Layout** tab.

Step 2 Select from the following editors:

- **Text Editor**
- **XML Editor**

Step 3 Copy the XML code for the HTTPS report you want to add from the Finesse default layout XML.

Example:

To add the Agent Report for HTTPS, copy the following:

```
<gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget.xml?
  gadgetHeight=310&viewId_1=99E6C8E210000141000000D80A0006C4&
  filterId_1=agent.id=CL%20teamName&
  viewId_2=9AB7848B10000141000001C50A0006C4&
  filterId_2=agent.id=CL%20teamName
</gadget>
```

Step 4 Replace my-cuic-server with the FQDN of your Cisco Unified Intelligence Center Server.

Step 5 Click **Save**.

Add Live Data Reports to Team Layout

The Finesse default layout XML contains commented XML code for the Live Data report gadgets available for the Finesse desktop.

To add the Live Data report gadgets to the desktop layout of a specific team:

Step 1 Click the **Desktop Layout** tab.

Step 2 Select from the following editors:

- **Text Editor**
- **XML Editor**

Step 3 Copy the XML code for the HTTPS report you want to add from the Finesse default layout XML.

Example:

To add the Agent Report for HTTPS, copy the following:

```
<gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget.xml?
  gadgetHeight=310&viewId_1=99E6C8E210000141000000D80A0006C4&
  filterId_1=agent.id=CL%20teamName&viewId_2=9AB7848B10000141000001C50A0006C4&
  filterId_2=agent.id=CL%20teamName
</gadget>
```

- Step 4** Click **Team Resources**.
- Step 5** Select the team from the list of teams for which you want to add the report.
- Step 6** In the Resources for <team name> area, click the **Desktop Layout** tab.
- Step 7** Check the **Override System Default** check box.
- Step 8** Select from the following editors:
- **Text Editor**
 - **XML Editor**
- Step 9** Replace “my-cuic-server” with the FQDN of your Cisco Unified Intelligence Center Server.
- Step 10** Click **Save**.

Modify Live Data Stock Reports for Finesse

To modify the Live Data stock reports in Cisco Unified Intelligence Center and add the modified report to the Finesse desktop layout:



Note To make sure the modified gadget renders in the Finesse desktop, you must give the appropriate permission for that report in Cisco Unified Intelligence Center.

- Step 1** Click the **Desktop Layout**.
- Step 2** Select from the following editors:
- **Text Editor**
 - **XML Editor**
- Step 3** In Cisco Unified Intelligence Center, in Edit view of the report, select the view for which you want to create a gadget URL and then click **Links**.
- The HTML Link field displays the permalink of the customized report.
- Step 4** Replace my-cuic-server with the FQDN of the Cisco Unified Intelligence Center Server.
- Step 5** Add the customized gadget URL to the desktop layout XML in the Manage Desktop Layout gadget and click **Save**.

Configure Live Data Reports with Multiple Views

Cisco Unified Intelligence Center allows you to display multiple Live Data reports or views on a single gadget. Agents can select the desired view to display from a drop-down list on the gadget toolbar, which lists up to five report views in *Report Name - View Name* format.

This procedure describes how to add multiple Live Data views to the Finesse desktop layout using the `viewId_n` and `filterId_n` keys. You can specify up to five report views to appear in your gadget. The first view among the five is the default view. There is no defined order for how the remaining views are displayed.

Finesse still supports the display of a single gadget using a single `viewId`. However, if you specify the single `viewId` along with multiple `viewId_n` keys, the multiple views are used and the single `viewId` is ignored.



Note To make sure the modified gadget renders in the Finesse desktop, you must give the appropriate permission for that report in Unified Intelligence Center.

Step 1 For each report or view that you want to include in the gadget, obtain the associated `viewId` from the permalink for the view:

- a) In Unified Intelligence Center, in Edit view of the report, select the desired view then click **Links**.

The HTML Link field displays the permalink of the customized report.

- b) Copy the permalink of the customized report from the **HTML Link** field, and paste it in a text editor, and then copy the `viewID` value from the permalink and save it.

Example:

Copy the `viewId`, which is underlined in this example, from the permalink for the report.

```
https://<Server Name>:8444/cuic/permalink/PermalinkViewer.htmx?
viewId=5C90012F10000140000000830A4E5B33&linkType=htmlType&viewType=Grid
```

Step 2 From the Finesse default layout XML, copy the gadget URL for one of the Live Data reports and paste it into a text editor.

Example:

Copy the URL for the Agent Skill Group for HTTPS from the default layout XML and paste it into a text editor:

```
<gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget.xml?gadgetHeight=310&
viewId_1=9AB7848B10000141000001C50A0006C4&filterId_1=agent.id=CL%20teamName</gadget>
```

Step 3 To update the URL to refer to a different report view, populate the `viewId_1` value (after the equal sign) with the desired `viewId` obtained in step 1.

Example:

The following shows the URL updated with the example `viewId` copied from step 1.

```
<gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget.xml?gadgetHeight=310&
viewId_1=5C90012F10000140000000830A4E5B33&filterId_1=agent.id=CL%20teamName</gadget>
```

Step 4 For each additional view you want to include:

- a) At the end of the URL, copy and paste the `viewId_1` and `agentId_1` strings with a leading ampersand.

Example:

```
<gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget.xml?gadgetHeight=310&
```

```
viewId_1=5C90012F10000140000000830A4E5B33&filterId_1=agent.id=CL%20teamName&
viewId_1=5C90012F10000140000000830A4E5B33&filterId_1=agent.id=CL%20teamName</gadget>
```

- b) Update the copied viewId_1 and filterId_1 in the URL to the next available integer (in this example, viewId_2 and filterId_2).

Example:

```
<gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget.xml?gadgetHeight=310&
viewId_1=5C90012F10000140000000830A4E5B33&filterId_1=agent.id=CL%20teamName&
viewId_2=5C90012F10000140000000830A4E5B33&filterId_2=agent.id=CL%20teamName</gadget>
```

- c) Populate the copied viewId value (after the equal sign) with the value defined in the permalink for the desired report (in this example, 99E6C8E210000141000000D80A0006C4).

Example:

```
<gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget.xml?gadgetHeight=310&
viewId_1=5C90012F10000140000000830A4E5B33&filterId_1=agent.id=CL%20teamName&
viewId_2=99E6C8E210000141000000D80A0006C4&filterId_2=agent.id=CL%20teamName</gadget>
```

- d) Make sure that the filterId value matches the type required by the report type, as follows:

- Agent Reports: filterId_N=agent.id=CL%20teamName
- Agent Skill Group Reports: filterId_N=agent.id=CL%20teamName
- Skill Group Reports: filterId_N=skillGroup.id=CL%20teamName
- Precision Queue Reports: filterId_N=precisionQueue.id=CL%20teamName

Step 5 Replace my-cuic-server with the FQDN of your Cisco Unified Intelligence Center Server.

Step 6 Add the customized gadget URL to the desktop layout XML in the Manage Desktop Layout gadget and click **Save**.



CHAPTER 6

Manage Phone Books

- [Phone Books and Contacts, on page 101](#)
- [Add Phone Book, on page 102](#)
- [Edit Phone Book, on page 102](#)
- [Delete Phone Book, on page 103](#)
- [Import Contacts, on page 103](#)
- [Export Contacts, on page 104](#)
- [Add Contact, on page 104](#)
- [Edit Contact, on page 104](#)
- [Delete Contact, on page 105](#)

Phone Books and Contacts

Finesse supports the following number of phone books:

- 10 global phone books
- 300 team phone books

The system supports a total of 50,000 contacts. The total number of contacts per agent across all phone books is limited to 6000.

Use the Manage Phone Books gadget to view, add, edit, or delete phone books and phone book contacts. Click the Name or Assign To headers to sort the phone books in ascending or descending order. Click the last Name, First Name, Number, or Note headers to sort the contacts in ascending or descending order.

The following table describes the fields on the Manage Phone Books gadget:

Field	Explanation
Name	The name of the phone book. It must be unique, and can be a maximum of 64 alphanumeric characters.
Assign To	Indicates if the phone book is global (All Users) or team (Teams).
Last Name	The last name of a contact. The last name can be a maximum of 128 characters. This field is optional.

Field	Explanation
First Name	The first name of a contact. The first name can be a maximum of 128 characters. This field is optional.
Number	The phone number for the contact. The phone number can be 1-32 characters long and cannot be blank.
Note	Optional text that describes the contact. The note can be a maximum of 128 characters.

Actions on the Manage Phone Books gadget:

- **New:** Add a new phone book or contact
- **Edit:** Edit an existing phone book or contact
- **Delete:** Delete a phone book or contact
- **Refresh:** Reload the list of phone books or contacts from the server
- **Import:** Import a list of contacts to the phone book
- **Export:** Export a list of contacts from the phone book

Add Phone Book

- Step 1** In the Manage Phone Books gadget, click **New**.
- Step 2** In the **Name** field, enter a name for the phone book.
- Note** Phone book names can be a maximum of 64 characters.
- Step 3** From the **Assign To** drop-down, select **All Users** if the phone book is global or **Teams** if the phone book is available to specified teams.
- Step 4** Click **Save**.
-

Edit Phone Book

- Step 1** In the Manage Phone Books gadget, select the phone book you want to edit.
- Step 2** Click **Edit**.
- Step 3** In the **Name** field, enter the new name for the phone book. If you want to change who can access the phone book, in the **Assign To** drop-down, choose **All Users** or **Teams**.
- Step 4** Click **Save**.

If you change the Assign To field from Teams to All Users, click **Yes** to confirm the change.

Delete Phone Book

- Step 1** In the Manage Phone Books gadget, select the phone book that you want to delete.
- Step 2** Click **Delete**.
- Step 3** Click **Yes** to confirm the deletion of the selected phone book.

Import Contacts

The Import function allows you to replace all the contacts in a phone book with a new list of contacts, or to populate a new phone book with contacts.

The import list must be in the specified comma separated values (CSV) format, and can contain a maximum of 6000 contacts. Import lists that contain more than 6000 contacts are rejected with an error message.

The CSV file contains the fields described in the following table:

Field	Max Length	Can Be Blank?	Permitted Characters
First Name	128	Yes	Note The CSV file that contains the contacts to import must use Latin encoding.
Last Name	128	Yes	
Phone Number	32	No	
Notes	128	Yes	

The following is an example of a phone book CSV file:

```
"First Name","Last Name","Phone Number","Notes"
"Amanda","Cohen","6511234",""
"Nicholas","Knight","612-555-1228","Sales"
"Natalie","Lambert","952-555-9876","Benefits"
"Joseph","Stonetree","651-555-7612","Manager"
```

A phone book CSV file must conform to this format and include the headers in the first line. During import, the file is scanned for illegal characters. If any are found, they are replaced with question marks.



Note Exported CSV files always show each field enclosed in double quotes to ensure that any commas or double quotes that are part of the actual filed data are not mistaken for field delimiters. If your data does not include these characters, you can omit the double quotes in files you prepare for importing.

-
- Step 1** In the Manage Phone Books gadget, select the phone book into which you want to import a list of contacts.
- Step 2** Click **Import**.
- Step 3** Click **Browse** and navigate to the location of the CSV file containing the contacts you want to import.
- Note** The CSV file must use Latin encoding.
- Step 4** Click **OK**.
-

Export Contacts

The Export function allows you to extract a list of contacts from an existing phone book. The exported list is saved in CSV format.

- Step 1** In the Manage Phone Books gadget, select the phone book that contains the contacts you want to export.
- Step 2** Click **Export**.
- Step 3** Click **Open** to open the CSV file in Excel, or click the **Save** drop-down list and choose **Save**, **Save as**, or **Save and open**.
- Step 4** A message appears that gives you the option to view the downloaded file, open the folder into which the download was saved, view the Internet Explorer View Downloads window, or dismiss the message without viewing the file.
- Step 5** A message appears that gives you the option to view the downloaded file, open the folder into which the download was saved, view the Internet Explorer View Downloads window, or dismiss the message without viewing the file.
-

Add Contact

-
- Step 1** In the Manage Phone Books gadget, select the phone book to which you want to add a contact.
- The List of Contacts for <phone book name> area appears.
- Step 2** Click **New**.
- Step 3** Complete the fields. The First Name, Last Name, and Note fields are optional and have a maximum length of 128 characters. The Number field is required and has a maximum length of 32 characters.
- Step 4** Click **Save**.
-

Edit Contact

-
- Step 1** In the Manage Phone Books gadget, select the phone book that contains the contact you want to edit.

The List of Contacts for <phone book name> area appears.

- Step 2** Select the contact you want to edit.
 - Step 3** Click **Edit**.
 - Step 4** Edit the fields that you want to change. The First Name, Last Name, and Note fields are optional and have a maximum of 128 characters. The Number field is required and has a maximum of 32 characters.
 - Step 5** Click **Save**.
-

Delete Contact

- Step 1** In the Manage Phone Books gadget, select the phone book that contains the contact you want to delete.
The List of Contacts for <phone book name> area appears.
 - Step 2** Select the contact that you want to delete.
 - Step 3** Click **Delete**.
 - Step 4** Click **Yes** to confirm the deletion of the selected contact.
-



CHAPTER 7

Manage Reasons

- [Wrap-Up Reasons, on page 107](#)
- [Manage Reason Code Conflicts During Upgrade, on page 110](#)
- [Predefined System Reason Codes, on page 110](#)
- [Sign Out Reason Codes, on page 112](#)
- [Not Ready Reason Codes, on page 115](#)

Wrap-Up Reasons

Wrap-Up reasons represent the reasons that agents can apply to calls and digital interactions. A Wrap-Up reason indicates why a customer contacted the contact center. For example, you may have one Wrap-Up reason for sales interactions and another for support interactions.

You can configure Wrap-Up reasons to be available globally to all agents or only to specific teams. The wrap-up reasons are applicable to both voice and digital channels.

Use the Manage Wrap-Up Reasons gadget to view, add, edit, or delete Wrap-Up reasons. Click the Reason Label header to sort the Wrap-Up reasons in ascending or descending order. Click the Global header to sort the Wrap-Up reasons by whether they are global (Yes) or not (No).



Note Cisco Finesse supports a maximum of 100 global and 1500 team Wrap-Up reasons. No more than 100 Wrap-Up reasons can be assigned to any one team.

Cisco Finesse supports the wrap-up functionality for all types of inbound and outbound calls.

Cisco Finesse supports the wrap-up functionality for all the digital channel interactions.

To enable wrap-up, you must configure both of the following attributes in the Unified CCE Agent Desktop Settings:

For more information about configuring Agent Desktop Settings, see the *Configuration Manager Online Help* for Unified CCE.



Note The showWrapUpTimer property can be used to show or hide timer in wrap-up state.
If showWrapUpTimer is set to true then timer is displayed.
If showWrapUpTimer is set to false then timer is hidden.



Note Wrap-Up timer is configurable. By default wrapUpCountDown property is set to true. The timer counts down by default when the agent is in wrap-up state. For more information, see *Desktop Properties*.
For Example, if you set the timer to 30 seconds, by default the timer starts from 30 and ends at zero.
The default behavior can be changed by setting the wrapUpCountDown property to false.
If an agent is configured for wrap-up and selects a pending state during a call, when the call finishes that agent goes into wrap-up and not the pending state selected during the call. The agent can end wrap-up by either selecting a new state (Ready or Not Ready) or letting the wrap-up timer expire. If the agent selects a new state, the new state overrides the pending state selected during the call. If the wrap-up timer expires, the agent transitions to the pending state.
Pending states are not applicable for digital channels.

The following table describes the fields on the Manage Wrap-Up Reasons gadget:

Field	Explanation
Reason Label	The label for the Wrap-Up reason. This label must be unique for each Wrap-Up reason and has a maximum length of 39 bytes (which equals 39 US English characters). Both alphanumeric and special characters are supported.
Global?	Yes/No. Indicates if the Wrap-Up reason is available globally to all agents (Yes) or to specific teams of agents (No).

Actions on the Manage Wrap-Up Reasons gadget:

- **New:** Add a new Wrap-Up reason
- **Edit:** Edit an existing Wrap-Up reason
- **Delete:** Delete a Wrap-Up reason
- **Refresh:** Reload the list of Wrap-Up reasons from the server



Note When you add, edit, or delete a Wrap-Up reason, the changes you make take effect on the agent or supervisor desktop after three seconds. However, agents who are signed in when the changes are made must sign out and sign in again to see those changes reflected on their desktops.

Add Wrap-Up Reason

- Step 1** In the Manage Wrap-Up Reasons gadget, click **New**.
- Step 2** In the Reason Label field, add a label for the Wrap-Up reason.
- Note** Wrap-Up reason labels are limited to 39 bytes.
- Step 3** If the Wrap-Up reason is global, select the Global? check box. If the Wrap-Up reason is specific to a team, clear the Global? check box.
- Note** By default, the Global? check box is selected.
- Step 4** Click **Save**.
-

Edit Wrap-Up Reason

- Step 1** In the Manage Wrap-Up Reasons gadget, select the Wrap-Up reason that you want to edit.
- Step 2** Click **Edit**.
- The Edit Wrap-Up Reason area appears.
- Step 3** In the Wrap-Up Reason Label field, enter the new label for the Wrap-Up reason. If you want to change who has access to the Wrap-Up reason, select or clear the Global? check box.
- Step 4** Click **Save**.
-

Delete Wrap-Up Reason

- Step 1** In the Manage Wrap-Up Reasons gadget, select the Wrap-Up reason that you want to delete.
- Step 2** Click **Delete**.
- A question appears asking you to confirm that you want to delete the selected Wrap-Up reason.
- Step 3** Click **Yes** to confirm the deletion of the selected Wrap-Up reason.
-

Force Wrap-Up Reason

For voice channel-If the Force Wrap-Up reason is configured, agents must select a Wrap-Up reason before changing the state after the call ends. The agent cannot change the state until the Wrap-up reason is applied. The Wrap-Up reason can be selected during the call or after the call ends.

For digital channels-If the Force Wrap-Up reason is configured, agents must select a Wrap-Up reason before transferring or ending an interaction.



Note The Force Wrap-Up reason is enabled by default. Use the CLI commands to disable and enable this feature. For more information, see *Desktop Properties*.

Manage Reason Code Conflicts During Upgrade

System Reason Codes are auto-generated reason codes that may conflict with custom reason codes when upgrading from an older version to Cisco Finesse 11.6(1). If there is a reason code conflict then the following message appears when you sign in to the administration console:

Custom reason codes conflict with system reason codes. Resolve to avoid reporting inconsistency.



Note Clear your browser cache to ensure that you are allowed to view and resolve system reason code conflicts.



Note When performing an upgrade from an earlier version in a Unified CCE deployment, modify the following custom reason codes: 999, 255, 20001, 20002, 20003, and 50041. This is done to avoid conflict with the system reason codes.

All conflicting reason codes are highlighted. To edit, select each conflicting reason code and click **Edit**. The **Edit Reason Code** area appears. Select the reason code from the available options listed or enter any other code you wish. The code must be unique to the particular category (Not Ready or Sign Out).

Once resolved, the reason code gets sorted based on the reason code number and placed in the table accordingly.

Predefined System Reason Codes

For Not Ready system reason codes and Sign Out system reason codes, only the reason code label can be edited and saved. The Global attribute and system code cannot be modified. In case the system reason code label is modified and you wish to revert to the default label, refer to the following list of predefined system reason codes:

System Reason Code	Reason Label	Reason Label Description
32767	Not Ready - Call Not Answered	Agent state changed because the agent did not answer the call.
32762	Not Ready - Offhook	The system issues this reason code in the following scenarios: <ul style="list-style-type: none"> When the agent goes off-hook to place a call, the Finesse desktop changes the agent status to Not Ready with this reason code.

		<ul style="list-style-type: none"> When the agent is in Ready state and a call is placed from the ACD (Automatic Call Distribution) line, the system issues this reason code. <p>Note Reason Code—50006 is used in place of 32762 in Cisco Finesse Release 12.5(1) when used along with Unified CCE Release 12.5(1) or higher.</p>
50001	Logged Out - System Disconnect	The CTI OS client disconnected, logging the agent out.
50002	Logged Out - System Failure	A CTI OS component disconnected, causing the agent to be logged out or set to the Not Ready state. This could be due to closing the agent desktop application, heart beat time out, or a CTI OS Server failure.
50002	Not Ready - Connection Failure	The system issues this reason code when the agent is forcibly logged out in certain cases.
50003	Logged Out - Device Error	Agent was logged out because the Unified CM reported the device out of service.
50004	Logged Out - Inactivity Timeout	Agent was logged out due to agent inactivity as configured in agent desk settings.
50005	Not Ready - Non ACD Busy	For a Unified CCE agent deployment, where the Agent Phone Line Control is enabled in the peripheral and the Non ACD Line Impact is configured to impact agent state, the agent is set to Not Ready while talking on a call on the Non ACD line with this reason code.
50006	Not Ready - Offhook	<p>The system issues this reason code in the following scenarios:</p> <ul style="list-style-type: none"> When the agent goes off-hook to place a call, the Unified CCE changes the agent status to Not Ready with this reason code. When the agent is in Ready state and a call is placed from the ACD (Automatic Call Distribution) line, the system issues this reason code. <p>Note This reason code is used from Cisco Finesse Release 12.5(1) along with Unified CCE Release 12.5(1) or higher.</p>
50010	Not Ready - Call Overlap	Agent was set to Not Ready state because the agent was routed two consecutive calls that did not arrive.
50020	Logged Out - Queue Change	Agent was logged out when the agent's skill group dynamically changed on the Administration & Data Server.

50030	Logged Out - Device Conflict	If an agent is logged in to a dynamic device target that is using the same Dialed Number (DN) as the PG static device target, the agent is logged out.
50040	Logged Out - Mobile Agent Call Fail	Mobile agent was logged out because the call failed.
50041	Not Ready - Mobile Call Not Answered	Mobile agent state changed to Not Ready because the call fails when the mobile agent's phone line rings busy.
50042	Logged Out - Mobile Agent Disconnect	Mobile agent was logged out because the phone line disconnected while using nailed connection mode.
65535	Not Ready - System Reinitialized	Agent reinitialized (used if peripheral restarts).
65534	Not Ready - System Reset	PG reset the agent, usually due to a PG failure.
65533	Not Ready - Extension Modified	An administrator modified the agent's extension while the agent was logged in.
20001	Not Ready - Starting Force Logout	Places the agent in the Not Ready state first before forcefully logging them off.
20002	Logged Out - Force Logout	Forces the logout request; for example, when Agent A attempts to log in to Cisco Agent Desktop and Agent B is already logged in under that agent ID, Agent A is asked whether or not to force the login. If Agent A answers yes, Agent B is logged out and Agent A is logged in. Reports then show that Agent B logged out at a certain time with a reason code of 20002 (Agent B was forcibly logged out).
20003	Not Ready - Agent Logout Request	If not already in the Logout state, request is made to place agent in the Not Ready state. Then logout request is made to log agent out.
999	Not Ready - Supervisor Initiated	The system issues this reason code when the agent's state is forcibly changed to Not Ready by the Supervisor.
999	Logged Out - Supervisor Initiated	The system issues this reason code when the agent's state is forcibly changed to Logout by the Supervisor.
255	Logged Out - Connection Failure	The system issues this reason code when the agent is forcibly logged out when there is a connection failure between the Cisco Finesse Desktop and the Cisco Finesse Server.

Sign Out Reason Codes

Sign Out reason codes represent reasons that agents can select when they sign out of the Finesse desktop.

Use the Manage Reason Codes (Sign Out) gadget to view, add, edit, or delete Sign Out reason codes. Click the Reason Label or Reason Code headers to sort the Sign Out reason codes by label or by reason code, in

ascending or descending order. Click the Type header to sort and display system or custom reason codes. Click the Global header to sort the reason codes by whether they are global (Yes) or not (No).

Sign Out reason codes can be global (visible to all agents) or team (visible only to agents on specified teams).



Note Finesse supports 200 Sign Out reason codes. These include 100 global Sign Out reason codes, and 100 Sign Out team reason codes. The team reason codes can be mapped to any team, and the same reason code can be mapped to multiple teams.

The following table describes the fields on the Manage Reason Codes (Sign Out) gadget:

Field	Explanation
Reason Label	The label for the Sign Out reason code. The label has a maximum length of 40 characters and should be unique for each Sign Out reason code. Alphanumeric and special characters are supported.
Type	The type of reason code (System or Custom). The column is default and can be sorted to display both System reason codes and Custom reason codes.
Reason Code	A code for the Sign Out reason. The code can be any value between 1 and 65535 and must be unique.
Global?	Yes/No. Indicates if the reason code is available globally to all agents (Yes) or to specific teams of agents (No).

Actions on the Manage Reason Codes (Sign Out) gadget:

- **New:** Add a new Sign Out reason code
- **Edit:** Edit an existing Sign Out reason code
- **Delete:** Delete a Sign Out reason code
- **Refresh:** Reload the list of Sign Out reason codes from the server



Note When you add, edit, or delete a Sign Out reason code, the changes you make take effect on the Finesse desktop after three seconds. However, agents who are signed in when the changes are made must sign out and sign in again to see those changes reflected on their desktops.

When an agent clicks Sign Out on the desktop, any configured Sign Out codes appear in a drop-down list. The agent can select the code that represents why that agent is signing out.

Add Sign Out Reason Code

Step 1 In the Manage Reason Codes (Sign Out) gadget, click **New**.

Step 2 In the Reason Label box, enter a label for the reason code.

Note Sign Out reason code labels are limited to 40 characters.

Step 3 In the Reason Code box, an auto populated reason code is displayed. If you choose not to save the prepopulated reason, you can enter your own reason code.

Note The code must be between 1 and 65535 and must be unique.
Ensure there are no leading or trailing spaces.

Step 4 If the reason code is global, select the Global? check box. If the reason code is specific to a team, clear the Global? check box.

Note By default, the Global? check box is selected.

Step 5 Click **Save**.

Edit Sign Out Reason Code

Step 1 In the Manage Reason Codes (Sign Out) gadget, select the reason code that you want to edit.

Step 2 Click **Edit**.

Step 3 If you want to change the label of the Sign Out reason code, in the Reason Label field, enter a new label for the reason code. If you want to change the code, in the Reason Code field, enter the new code. If you want to change who has access to the code, select or clear the Global? check box.

Step 4 Click **Save**.

Delete Sign Out Reason Code



Note An error may occur if an agent selects a Sign Out reason code after it has been deleted. Agents who are signed in when you make changes to Sign Out reason codes must sign out and sign back in to see those changes reflected on their desktops.

Step 1 In the Manage Reason Codes (Sign Out) gadget, select the Sign Out reason code that you want to delete.

Step 2 Click **Delete**.

Step 3 Click **Yes** to confirm the deletion of the selected Sign Out reason code.

Not Ready Reason Codes

Not Ready reason codes represent reasons that agents can select when they change their state to Not Ready.

Use the Manage Reason Codes (Not Ready) gadget to view, add, edit, or delete Not Ready reason codes.

1. Click the Reason Label or Reason Code headers to sort the Not Ready reason codes by label or reason code in ascending or descending order.
2. Click the Type header to sort and display system or custom reason codes.
3. Click the Global header to sort reason codes by whether they are global (Yes) or not (No).

Not Ready reason codes can be global (visible to all agents) or team (visible only to agents on specified teams).



Note Finesse supports a total of 200 Not Ready reason codes. This includes a maximum of 100 global Not Ready reason codes, and 100 team Not Ready reason codes. The team reason codes can be mapped to any team, and the same reason code can be mapped to multiple teams.

The following table describes the fields on the Manage Reason Codes (Not Ready) gadget:

Field	Explanation
Reason Label	The label for the Not Ready reason code. The label has a maximum length of 40 characters and should be unique for each Not Ready reason code. Alphanumeric and special characters are supported.
Type	The type of reason code (System or Custom). The column is default and can be sorted to display both System reason codes and Custom reason codes.
Reason Code	A code for the Not Ready reason. The code can be any value between 1 and 65535 and must be unique.
Global?	Yes/No. Indicates if the reason code is available globally to all agents (Yes) or to specific teams of agents (No).

Actions on the Manage Reason Codes (Not Ready) gadget:

- **New:** Add a new Not Ready reason code
- **Edit:** Edit an existing Not Ready reason code
- **Delete:** Delete a Not Ready reason code
- **Refresh:** Reload the list of Not Ready reason codes from the server



Note When you add, edit, or delete a Not Ready reason code, the changes you make take effect on the Finesse desktop after three seconds. However, agents who are signed in when the changes are made must sign out and sign back in to see those changes reflected on their desktops.

When an agent signs in to the Finesse desktop, the agent state is set to Not Ready. The agent can then choose to go to Ready status or choose from one of the configured Not Ready reason codes from the agent state drop-down list.

If an agent wants to change from Ready to Not Ready status, that agent can choose the appropriate Not Ready reason code from the list of configured codes.

An agent who is on a call can select a state to be applied when the call is complete. For example, if an agent wants to be in Not Ready state when the call ends, that agent can choose Not Ready from the drop-down list while still on the call. The Finesse desktop shows the agent in Talking state and a pending state of Not Ready.

If the agent also applies a Not Ready reason code, the desktop shows the pending state with the reason code (in this case, Not Ready - Lunch).

Pending state changes appear on the desktop while the agent's state is Talking (for example, on hold, in a consult call, conference, or silent monitor call).



Note During a PG or CTI server failover, the pending state of an agent is not retained.

Add Not Ready Reason Code

Step 1 In the Manage Reason Codes (Not Ready) gadget, click **New**.

Step 2 In the Reason Label box, enter a label for the reason code.

Note Not Ready reason code labels are limited to 40 characters.

Step 3 In the Reason Code box, an auto populated reason code is displayed. If you choose not to save the prepopulated reason code, you can enter your own reason code.

Note The code must be between 1 and 65532 and must be unique.

Ensure there are no leading or trailing spaces.

Step 4 If the reason code is global, select the Global? check box. If the reason code is specific to a team, clear the Global? check box.

Note By default, the Global? check box is selected.

Step 5 Click **Save**.

Note The Finesse server removes leading or trailing spaces before saving the Reason Label in the database.

Edit Not Ready Reason Code

- Step 1** In the Manage Reason Codes (Not Ready) gadget, select the reason code that you want to edit.
- Step 2** Click **Edit**.
- Step 3** If you want to change the label for the Not Ready reason code, in the Reason Label field, enter a new label for the reason code. If you want to change the code, in the Reason Code field, enter the new code. If you want to change who has access to the code, select or clear the Global? check box.
- Step 4** Click **Save**.
-

Delete Not Ready Reason Code



Note An error may occur if an agent selects a Not Ready reason code after it has been deleted. Agents who are signed in when you make changes to Not Ready reason codes must sign out and sign back in to see those changes reflected on their desktops.

- Step 1** In the Manage Reason Codes (Not Ready) gadget, select the Not Ready reason code that you want to delete.
- Step 2** Click **Delete**.
- Step 3** Click **Yes** to confirm the deletion of the selected reason code.
-



CHAPTER 8

Manage Team Resources

- [Team Resources](#), on page 119
- [Assign Phone Books and Reasons to Team](#), on page 120
- [Unassign Phone Books and Reasons from Team](#), on page 120
- [Assign Custom Desktop Layout to Team](#), on page 121
- [Assign Workflows to Team](#), on page 123
- [Unassign Workflows from Team](#), on page 123

Team Resources

Use the Manage Team Resources gadget on the Team Resources tab to assign and unassign phone books, reasons, custom desktop layouts, and workflows to teams. Click the Name or ID header to sort the teams in ascending or descending order.

The Manage Team Resources gadget contains six tabs, each enabling you to assign or unassign resources to a team. The tabs are defined in the following table:

Tab Name	Description
Desktop Layout	Use this tab to customize the desktop layout for the team. The default layout is defined in the Manage Desktop Layout gadget. You can define one custom layout for the team.
Phone Books	Use this tab to assign and unassign phone books to the team. Only phone books that are defined in the Manage Phone Books gadget as available to teams are available for assignment.
Reason Codes (Not Ready)	Use this tab to assign and unassign Not Ready reason codes to the team. Only Not Ready reason codes that are defined in the Manage Reason Codes (Not Ready) gadget as available to teams (not global) are available for assignment.
Reason Codes (Sign Out)	Use this tab to assign and unassign Sign Out reason codes to the team. Only Sign Out reason codes that are defined in the Manage Reason Codes (Sign Out) gadget as available to teams (not global) are available for assignment.
Wrap-Up Reasons	Use this tab to assign and unassign Wrap-Up reasons to the team. Only Wrap-Up reasons that are defined in the Manage Wrap-Up Reasons gadget as available to teams (not global) are available for assignment.

Tab Name	Description
Workflows	Use this tab to assign and unassign workflows to the team. Only workflows that are defined in the Manage Workflows gadget are available for assignment.

Actions on the Manage Team Resources Gadget

- **Add:** Assign a phone book, reason, or workflow to the team
- **Save:** Save the phone book, reason, desktop layout assignment, or workflow to the team
- **Revert:** Cancel any changes made before they are saved
- **Refresh:** Refresh the list of teams



Note If you select a team and then click Refresh, the team is deselected and the Resources area for that team disappears. The list of teams is refreshed and you must select a team again.

Add or Delete a Team When Database is Not Accessible

If you add or delete a team when Finesse cannot access the Finesse database, those changes do not appear in the Finesse administration console unless you restart Cisco Finesse Tomcat or the CTI server.

Assign Phone Books and Reasons to Team

-
- Step 1** In the Manage Team Resources gadget, select a team.
- Step 2** Click the tab for the resource you want to assign for the selected team.
- Step 3** Click **Add**.
- Step 4** Select one or more resources from the list to assign them to the team.
Resources you assign are highlighted in blue in the Add <resources> popup and added to the List of <resources> area.
- Step 5** When you finish assigning resources, click **Save**.

Note You can make changes on all resource tabs and then save them at the same time. If there is an error on one resource tab but not others, the changes on the tabs with no errors are saved while the changes on the tab with errors are not saved.

Unassign Phone Books and Reasons from Team

-
- Step 1** In the Manage Team Resources gadget, select a team.

Step 2 Click the tab for the resource you want to unassign from the selected team.

Step 3 Click the red X next to the resource you want to unassign.

Step 4 Click **Save**.

Assign Custom Desktop Layout to Team

Step 1 In the **Manage Team Resources** gadget, select a team.

Step 2 Click **Desktop Layout**.

The Desktop Layout XML area appears. The area contains the default desktop layout XML.

Step 3 Select the **Override System Default** check box.

The XML becomes editable.

Step 4 Select from the following editors:

- **Text Editor**
- **XML Editor**

For more information, see *Default Layout XML*.

Step 5 Edit the XML.

Step 6 Click **Save**.

The custom desktop layout replaces the default desktop layout for the team after 10 seconds. If a supervisor or agent is signed in when the change is saved, the change does not take effect on their desktop until the supervisor or agent signs out and signs in again.

Note If you clear the **Override System Default** check box, any changes you made to the XML are lost and the XML in the editing pane reverts to the default desktop layout XML.



Note If the Supervisor is managing single/multiple teams, the custom layout of the team for which the supervisor is a resource/agent is displayed. However, if the supervisor is not the resource/agent of a team, the default layout is displayed.

Customize Desktop Properties at Team Level

You can customize the Finesse desktop properties for a specific team.

Step 1 In the **Manage Team Resources** gadget, select a team.

Step 2 Click **Desktop Layout**.

Step 3 Select the **Override System Default** check box.

Step 4 Select from the following editors:

- **Text Editor**
- **XML Editor**

Step 5 Enter the desktop property name in the config key tag.

Step 6 Enter the possible value of the desktop property in the value tag.

The following are the sample desktop property entries, as displayed in the default **Desktop Layout**. To change these desktop property entries in **Text Editor**, remove the comment (`<!--` and `-->`) and set appropriate values.

```
<!-- <config key="enableDragDropAndResizeGadget" value="false"/> -->
<!-- <config key="wrapUpCountDown" value="true"/> -->
<!-- <config key="desktopChatAttachmentEnabled" value="true"/> -->
<!-- <config key="forceWrapUp" value="true"/> -->
```

Note If the property value is defined in the team-specific desktop layout (**Manage Team Resources** > **Desktop Layout**), then the team-specific desktop layout takes precedence over the property value defined in the **Desktop Layout** and CLI.

For more information on customizing desktop properties at **Desktop Layout**, see *Customize Desktop Properties*.

For more information on Finesse CLIs, see *Desktop Properties*.

The following table lists the desktop properties that support team-level updates:

Config Key	Value	Default Value
enableDragDropAndResizeGadget	true false	false
enableShortCutKeys	true false	true
forceWrapUp	true false	true
wrapUpCountDown	true false	true
showWrapUpTimer	true false	true
desktopChatAttachmentEnabled	true false	true
desktopChatMaxAttachmentSize	Range: 1—10 (MB)	5
desktopChatUnsupportedFileTypes	Unsupported file formats include comma-separated valid file extensions. For example: .exe, .sh	.exe, .msi, .sh, .bat
showAgentHistoryGadgets	true false	true
showActiveCallDetails (for Supervisor Only)	true false	true
pendingDTMFThresholdCount	Range: 1—20	20
dtmfRequestTimeoutInMs	Range: 1000—200000 (1 to 200 seconds)	5000 (5 seconds)

Config Key	Value	Default Value
enableDropParticipantFor	supervisor_only conference_controller_ and_supervisor all	supervisor_only
dropParticipant	agents all	agents

For more information on Finesse desktop properties, see *Desktop Properties*.

Step 7 Click **Save**.

The change takes effect when the agent or supervisor refreshes the Finesse desktop or sign out and sign in again.

Note If you clear the **Override System Default** check box and click **Save**. The changes are overwritten, and the editing pane reverts to the default desktop layout XML.

Assign Workflows to Team

Step 1 In the Manage Team Resources gadget, select a team.

Step 2 Click the Workflows tab.

Step 3 Click **Add**.

Step 4 Select one or more workflows from the list to assign them to the team.

Workflows you assign are highlighted in blue in the Add Workflows popup and added to the List of Workflows area.

Step 5 Workflows are run in the order they are listed. Use the up and down arrows to move a selected workflow to the desired position in the list.

Step 6 When you has finished assigning workflows, click **Save**.

Note You can make changes on all resource tabs and then save them at the same time. If there is an error on one resource tab but not on others, the changes on the tabs with no errors are saved while the changes on the tab with errors are not saved.

Unassign Workflows from Team

Step 1 In the Manage Team Resources gadget, select a team.

Step 2 Click the Workflows tab.

Step 3 Click the red X next to the workflow to unassign.

Step 4 Click **Save**.



CHAPTER 9

Manage Workflows

- [Workflows and Workflow Actions, on page 125](#)
- [Add Browser Pop Workflow Action, on page 130](#)
- [Add HTTP Request Workflow Action, on page 131](#)
- [Edit Workflow Action, on page 132](#)
- [Delete Workflow Action, on page 132](#)
- [Add Workflow, on page 132](#)
- [Edit Workflow, on page 133](#)
- [Delete Workflow, on page 133](#)

Workflows and Workflow Actions

You can use workflows to automate common repetitive agent tasks. A workflow has a unique name and a helpful description. Use the Manage Workflows and Manage Workflow Actions gadgets to view, add, edit, or delete workflows and workflow actions.

All workflows are team-level workflows. You cannot create a global workflow. If you need a global workflow, create a team workflow and assign it to all teams.

Cisco Finesse supports the following number of workflows and workflow actions:

- 100 workflows per Cisco Finesse system
- 100 actions per Cisco Finesse system
- 20 workflows per team
- Five conditions per workflow
- Five actions per workflow
- Five variables per action

The following fields can be used to configure workflows:

- queueNumber
- queueName
- callKeyCallId

- callKeyPrefix
- callKeySequenceNum
- wrapUpReason
- For Voice - Call variables, Outbound Option variables, queue details, wrap-up reasons, agent details, or team details.
- For Email - Queue name and email attributes like From, To, Cc, Bcc, or Subject.
- For Chat - Queue name, chat type, or system defined customer details as available from the web chat form.

Click the column headers to sort workflows and workflow actions in ascending or descending order.

The following table describes the fields on the Manage Workflows gadget:

Field	Explanation
Name	The name of the workflow must be unique and can have a maximum length of 40 characters.
Description	The description of the workflow can have a maximum length of 128 characters.
Media	The media of the workflow. You can configure the media to Voice and any preferred Digital Channel.

The following table describes the fields on the Manage Workflow Actions gadget:

Field	Explanation
Name	The name of the workflow action must be unique and can have a maximum length of 64 characters.
Type	The type of workflow. Possible values are Browser Pop and HTTP Request.

Actions on the Manage Workflows and Manage Workflow Actions gadgets:

- **New:** Add a new workflow or workflow action
- **Edit:** Edit a workflow or workflow action
- **Delete:** Delete a workflow or workflow action
- **Refresh:** Reload the list of workflows or workflow actions from the server.

You can configure workflow actions to be handled by the Cisco Finesse desktop or in a third-party gadget. A third-party gadget can be designed to handle the action differently than Cisco Finesse does.

Each workflow must contain only one trigger. Triggers are based on Cisco Finesse dialog events.



Note You can configure the trigger only after you select the media.

- Voice dialog events include the following:

- When a Call arrives
- When a Call is answered
- When a Call ends
- When making a Call



Note The call variable workflow responds as expected only when you add both the conditions **Is not equal** and **Is not empty**.

- While previewing an Outbound Option call.
- Digital Channels dialog events include the following:
 - When a task is offered
 - When a task is accepted



Note Some solutions such as ECE don't provide a separate accept task functionality. Therefore, the tasks that are offered are auto accepted, which simultaneously generate the **task is accepted** event along with the **task is offered** event. In such scenarios, use only one event (**task is accepted** or **task is offered**) for configuring workflows because there is no difference between these two events.

- When a task is active
- When a task is paused
- When a task is interrupted
- When a task is closed

The workflow engine uses the following simple logic to determine whether to run a workflow:



Note The workflow logic and examples are similar for all media.

- Its trigger set and conditions are evaluated against each dialog event received.
- The workflow engine processes workflow events for the first call that matches any configured workflow's trigger set and conditions. No other workflows run until this call has ended. If the agent accepts a second call while still on the first call, workflows do not run on the second call even after the first call has ended.
- After a workflow for a particular trigger type (for example, Call Arrives) runs, it never triggers again for the same dialog ID.

The workflow engine caches workflows for an agent when the agent signs in. Workflows do not change for the agent until the agent signs out and signs in again or refreshes the browser.



Note Whenever the browser is refreshed, the workflows that trigger the following events run:

- when a call arrives
- when a call is answered
- when making a call

When an agent refreshes the browser, the workflow engine considers the call as newly arrived or newly made. If an HTTP request action is part of the workflow, the HTTP request is sent when the agent refreshes the browser. Applications that receive the HTTP requests must account for this scenario.

An example of a workflow is a Call Arrival event that triggers an action that collects information from the dialog event (for example, the ANI or customer information) and displays a web page containing customer information.

You can filter trigger events by the value of the data that comes in the event. You can configure a workflow to run if any of the conditions are met or if all the conditions are met.

Individual conditions comprise of the following:

- A piece of event data to be examined. For example, **DNIS** or call variables.
- A comparison between the event data and the values entered (for example **contains, is equal to, is not equal to, begins with, ends with, is empty, is not empty, and is in list**).

When the trigger and its conditions are satisfied, a list of actions assigned to the workflow are run. The actions are run in the listed order.

Workflows run only for agents and supervisors who are Cisco Finesse users. The Workflow Engine is a JavaScript library that runs client-side on a per-user basis within the Cisco Finesse desktop application. The desktop retrieves the workflows that are to be run for a user from the server when the user signs in or when the browser is refreshed.



Note Changes made to a workflow or its actions while a user is signed in are not automatically pushed to that user.

It is possible to set workflows, conditions, and actions that are contradictory so that a workflow or action cannot function. Workflows are not validated.

If multiple workflows are configured for a team, the Workflow Engine evaluates them in the configured order. The Workflow Engine ignores workflows with no actions. When the Workflow Engine finds a workflow with a matching trigger for an event and the workflow conditions evaluate to true, that workflow is used, and the subsequent workflows in the list are not evaluated. Workflows with no conditions evaluate to true if the event matches the workflow trigger. All workflows are enabled by default. Only one workflow for a specific user can run at a time.

The Workflow Engine retrieves dialog-based variables that are used in workflow conditions from the dialog that triggered the workflow. If a variable is not found in the dialog, its value is considered to be empty.

The Workflow Engine runs the actions that are associated with the matched workflow in the order in which they are listed. The Workflow Engine runs actions in a workflow even if the previously run action fails. Failed actions are logged.

The Cisco Finesse server controls the calls that are displayed to the Cisco Finesse user. If the user has multiple calls, the workflow applies only to the first call that matches a trigger. If the first call displayed does not match any triggers but the second call does match a trigger, the Workflow Engine evaluates and processes the triggers for the second call.

A call is considered to be the first displayed call if it is the only call on the Cisco Finesse desktop when it appears. If two calls on a phone are merged (as they are in a conference call), then the first displayed call flag value of the surviving call is used.

If a user has a call and the user refreshes the browser, the Workflow Engine evaluates the call as it is. If the dialog data (call variable values) change, the data may not match the trigger and conditions of the original workflow. The data may match a different workflow or no workflows at all.

If a user has multiple calls and the user refreshes the browser, the Workflow Engine treats the first dialog received from the Cisco Finesse server as the first displayed call. This call may not be the same call that was first displayed before the refreshing the browser. Dialogs received for any other call are ignored because they are not considered as first displayed calls. After refreshing the browser, if dialogs for more than one call are received before the Workflow Engine is loaded, none of the dialogs are evaluated because they are not considered as first displayed calls.

Workflows that are run for both Cisco Finesse agents and supervisors. The team to which the supervisor belongs (as distinguished from the team that the supervisor manages) determines which workflows run for the supervisor. Put the supervisors in their own team to keep agent workflows from being run for them.

Workflow Triggers and Outbound Calls



Note When you create a workflow specifically for Outbound Option calls, add a condition of BASTatus is not empty (except for the Workflow Trigger 'When a call arrives' as BASTatus will be empty at that point of time). This condition ensures that the workflow can distinguish Outbound Option calls from agent-initiated outbound calls.

The following table illustrates when workflows trigger in outbound call scenarios:

Workflow Trigger	Direct Preview Outbound Call	Preview Outbound Call	Progressive/Predictive Outbound Call
While previewing a call	When the agent previews the call (before accepting or rejecting it)	When the agent previews the call (before accepting or rejecting it)	Does not trigger
When a call arrives	Does not trigger	When the agent accepts the call	When the call arrives on the agent desktop
When a call is answered	When the customer answers the call and during failover	When the customer answers the call and during failover	When the customer answers the call
When a call is made	When the customer call is initiated	When the customer call is initiated	When the customer call is initiated, and during failover

Workflow Trigger	Direct Preview Outbound Call	Preview Outbound Call	Progressive/Predictive Outbound Call
When a call ends	When the customer call ends	When the customer call ends	When the customer call ends

Add Browser Pop Workflow Action

The Browser Pop workflow action opens a browser window or tab on the user's desktop when workflow conditions are met.



Note Whether the action opens a new window or tab on the desktop depends on the target user's browser settings.

Step 1 In the Manage Workflow Actions gadget, click **New**.

Step 2 In the Name box, enter a name for the action.

Note Workflow action names are limited to 64 characters.

Step 3 From the Type drop-down list, choose **Browser Pop**.

Step 4 From the Handled By drop-down list, choose what will run the action, either the Finesse Desktop or Other (a third-party gadget).

Step 5 In the Window Name box, enter the ID name of the window that is opened. Any action that uses this window name reuses that specific window.

Note Window names are limited to 40 characters, and can be blank. If you leave the window name blank, a new window opens every time the action runs.

Step 6 Enter the URL of the browser window and click the tag icon at the right of the box and select one or more variables from the drop-down list to add tags.

Example:

http://www.google.com/search?q= &

For every variable you select, you can enter test data in the Sample Data box. A sample URL is automatically built in the Browser URL box below the Sample Data area. To test the URL, click Open to open the URL in your browser.

Note Finesse does not validate the URL you enter.

Step 7 Click **Save**.

Add HTTP Request Workflow Action

The HTTP Request workflow action makes an HTTP request to an API on behalf of the desktop user.

-
- Step 1** In the Manage Workflow Actions area, click **New**.
- Step 2** In the Name box, enter a name for the action.
- A workflow action name can contain a maximum of 64 characters.
- Step 3** From the Type drop-down list, select **HTTP Request**.
- Step 4** From the Handled By drop-down list, select what will run the action, the Finesse desktop or Other (a third-party gadget).
- Step 5** From the Method drop-down list, select the method to use.
- You can select either PUT or POST.
- Step 6** From the Location drop-down list, select the location.
- If you are making the HTTP request to a Finesse API, select **Finesse**. If you are making a request to any other API, select **Other**.
- Step 7** In the Content Type box, enter the content type.
- The default content type is application/xml, which is the content type for Finesse APIs. If you are using a different API, enter the content types for that API (for example, application/JSON).
- Step 8** In the URL box, enter the URL to which to make the request. To add variables to the URL, click the tag icon at the right of the box and select one or more variables from the drop-down list.
- Note** The drop-down list contains variables from all the configured media channels.
- Example:**
- The following is the URL example for a Finesse API:
- ```
/finesse/api/Dialog/
```
- Note** When the location is FINESSE, do not specify the protocol, host, and port information. Finesse automatically fetches these details when the REST request is run.
- If you want to make a request to another API, you must enter the entire URL (for example, http://googleapis.com).
- You can click the tag icon at the right of the box and select one or more variables from the drop-down list to add tags to the URL. In the preceding example, to add the dialogId, click the tag icon and select dialogId from the list.
- Step 9** In the Body box, enter the text for the request. The body must match the content type (for example, if the content types is application/xml, the body must contain XML). To add variables to the body, click the tag icon at the right of the box and select one or more variables from the drop-down list.

**Example:**

To make an HTTP request to the Dialog - Start a recording API, enter the following into the Body box:

```

<Dialog>
<targetMediaAddress> extension ✕ </targetMediaAddress>
<requestedAction>SEND_DTMF</requestedAction>
<actionParams><ActionParam><name>dtmfString</name><value>8</value></ActionParam></actionParam>
</Dialog>

```

To add the extension, click the tag icon and select extension.

For every variable you add, you can enter test data in the Sample Data box.

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

---

## Edit Workflow Action

---

- Step 1** In the Manage Workflow Actions gadget, select the action that you want to edit.
  - Step 2** Click **Edit**.
  - Step 3** Edit the fields that you want to change.
  - Step 4** Click **Save**.
- 

## Delete Workflow Action

---

- Step 1** In the Workflow Actions gadget, select the action that you want to delete.
  - Step 2** Click **Delete**.
  - Step 3** Click **Yes** to confirm the deletion of the selected action.
- 

## Add Workflow

---

- Step 1** In the Manage Workflows gadget, click **New**.
- Step 2** From the **Choose Media** drop-down, select the media.
  - Note** In case of a voice only configuration, the **Choose Media** drop-down will display only Voice.
- Step 3** In the **Name** box, enter the name of the workflow.
  - Note** The name is limited to 40 characters.
- Step 4** In the **Description** box, enter a description of the workflow.

**Note** The description is limited to 128 characters.

**Step 5** In the **When to perform Actions** drop-down list, select the event that triggers the workflow.

**Note** The drop-down actions change depending on the selected media.

**Step 6** In the **How to apply Conditions** box, select if all conditions are met, or if any conditions are met, and then click **Add Condition** to add up to five conditions.

**Note** Variables in the drop-down for conditions are grouped depending on the selected media.

**Example:**

For example, you can specify that the action is taken when CallVariable 1 equals 123 and CallVariable 2 begins with 2.

**Step 7** In the Ordered List of Actions area, click **Add** to open the Add Actions area. Click an action in this area to add it to the Ordered List of Actions.

**Step 8** Use the up and down arrows next to the Ordered List of Actions to move actions into the performance order.

**Step 9** Click **Save**.

**Step 10** Assign the workflow to one or more teams.

**Note** A workflow does not run until it is assigned to a team.

---

## Edit Workflow

---

**Step 1** In the Manage Workflows gadget, select the workflow you want to edit.

**Step 2** Click **Edit**.

**Note** The media for an existing workflow can be changed by editing the workflow.

**Step 3** Edit the fields that you want to change.

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

---

## Delete Workflow

---

**Step 1** In the Manage Workflows gadget, select the workflow that you want to delete.

**Step 2** Click **Delete**.

**Step 3** Click **Yes** to confirm the deletion of the selected workflow.

---





# CHAPTER 10

## Manage Connected Agents

- [Connected Agents, on page 135](#)

### Connected Agents

Use the Connected Agents gadget on the Connected Agents tab to view the list of agents currently signed in to Cisco Finesse.

You can use this gadget to determine which agents are signed in to the Publisher Side or the Subscriber Side. You can use this gadget also to filter the client types and identify the client type through which an agent has logged in. The client types can be Finesse Desktop, Finesse IP Phone, and Custom Desktop.

The list of signed-in agents is displayed in the form of a table, the Connected Agents table.

You can search the Connected Agents table for certain entries, sort the table, or refresh the table to view the latest data. The number of agents signed in to the Publisher Side and the Subscriber Side are displayed in the gadget (above the table).

The columns of the Connected Agents table are displayed below:

| Column         | Explanation                                                              |
|----------------|--------------------------------------------------------------------------|
| Agent Name     | The first and last name of an agent.                                     |
| Username       | The Agent ID or username required to sign in to Cisco Finesse.           |
| Extension      | The extension number of the agent.                                       |
| Team           | The team the agent belongs to.                                           |
| Connected Time | The total duration (in hh:mm:ss) for which the agent has been logged in. |
| Connected Side | Publisher/Subscriber/Both Sides.                                         |
| Finesse Host   | The Finesse host through which the agent is connected.                   |

#### Actions on the Connected Agents Gadget

- **Search:** Searches for the entered text across all columns of the Connected Agents table.

- **Sort:** Sorts the column values of the Connected Agents table in ascending or descending order.
- **Filter:** Filters agents connected to Both Sides, the Publisher side, or the Subscriber side.

The default selection for this drop-down box is Both Sides.

- **Refresh:** Refreshes the Connected Agents table. When the Refresh button is clicked, a new REST API call is made to both the publisher and subscriber servers to get the latest information about the signed-in agents.

The time at which the agent information was last fetched from the server is displayed beside the Refresh button (For example, Updated 45 minutes ago).





# CHAPTER 11

## Manage Security

---

- [HTTPS Support](#), on page 137
- [Reset Security or Admin Password](#), on page 138
- [Cross-Origin Resource Sharing \(CORS\)](#), on page 139
- [Gadget Source Allowed List](#), on page 139
- [Security Enhancements](#), on page 139

## HTTPS Support

Cisco Finesse does not support plain HTTP but supports only secure HTTP (HTTPS). In response to clients accessing Finesse using plain HTTP, the 301 HTTP redirect is issued to the secured port 8445.



---

**Note** Cisco Finesse supports HTTP/2 protocol by default.

---

To access the administration console using HTTPS, enter the following URL in your browser:

```
https://FQDN:8445/cfadmin
```

Where FQDN is the name of your primary Finesse server and 8445 is the port number.

Similarly, agents and supervisors can access their desktops using HTTPS as follows:

```
https://FQDN:8445/desktop
```

To eliminate browser security warnings each time you access the administration console or agent desktop through HTTPS, you can obtain and upload a CA certificate or you can use the self-signed certificate that is provided with Finesse.

If you add custom gadgets that perform HTTPS requests to Finesse, you must add a certificate to the Finesse server for that gadget.

## HSTS

Finesse supports HTTP Strict Transport Security (HSTS) for increased security. HSTS is automatically enabled, in which case the Finesse server sends HTTPS responses indicating to browsers that Finesse can only be accessed using HTTPS. If users then try to access Finesse using HTTP instead of HTTPS, the browser changes

the connection to HTTPS before generating any network traffic. This functionality prevents browsers from sending requests to Finesse using unencrypted HTTP before the server can redirect them.

## Reset Security or Admin Password

If you need to reset the security or admin password, you must perform the following steps on the console of the system using VSphere. You cannot ssh to the system to run the command.

- 
- Step 1** Sign in to the platform window with the following username and password:  
pwrecovery/pwreset
- The following messages appear:  
Welcome to Platform password reset.  
Admin and Security password reset are possible.  
Press any key when ready.
- Step 2** Press any key to continue.
- The following messages appear:  
If you have a CD or DVD in the disk drive, remove it now.  
Press any key to continue.
- Step 3** If there is a disk in the disk drive, remove it. When you are ready, press any key to continue.
- The system checks to ensure that you have removed the disk from the drive.  
The following message appears:  
Insert a valid CD or DVD into the disk drive.
- Step 4** Connect the CD/DVD drive and point it to the ISO image.
- The system checks to ensure you have inserted the disk.  
After the system verifies that you have inserted a disk, you are prompted to choose one of the following options:  
Enter 'a' for admin password reset.  
Enter 's' for security password reset.  
Enter 'q' for quit.
- Step 5** Select the appropriate option and provide the new password.
- The system resets the password.
-

# Cross-Origin Resource Sharing (CORS)

Finesse supports CORS requests and allows the customization of the domains which are allowed to make CORS requests to the Finesse desktop. Once CORS is enabled via the CLI **utils finesse cors enable**, the CORS origin request from external domains is blocked by the browser. To enable specific domains to access Finesse desktop via CORS, the domains need to be added to the CORS origin allowed list using the CLI **utils finesse cors allowed\_origin add**. For more information on the CORS CLIs, see *Cisco Finesse CLI*.

## Gadget Source Allowed List

Shindig proxies requests from the Finesse desktop to external servers and this introduces the possibility of server side request forgery (SSRF). To prevent SSRF, you can choose to restrict outgoing connections requested by the gadgets to specific URIs by enabling Shindig allowed listing CLIs and adding the required URIs to the allowed list. For more information on Gadget Source Allowed List CLIs, see *Cisco Finesse CLI*.

## Security Enhancements

The security enhancements in Cisco Finesse are as follows:

- After the fresh install, by default the **utils system reverse-proxy client-auth** is enabled. If this is enabled and there are multiple certificates in the client system, when agents login to Finesse through LAN, it forces the agents to select one of the certificates to communicate with the Finesse server. If the deployments aren't configured for VPN-less access to Finesse, disable it by running the **utils system reverse-proxy client-auth disable** command on both the Finesse nodes.
- By default, Cisco Finesse Notification Service unsecure XMPP port 5222 and BOSH/WebSocket (HTTP) port 7071 are disabled.

Use the CLI command **utils finesse set\_property webservices enableInsecureOpenfirePort true** to enable these ports.

- Validation of the X.509 certificate is enforced. It is mandatory to have the following valid non-expired X.509 CA or self-signed certificates, which must be imported into the Cisco Finesse trust store.
  - Cisco Finesse node certificates are available by default. The administrator must verify the validity of the certificates, as non-expired certificates are invalid.
    - Valid non-expired Cisco Finesse primary certificate must be present on the secondary Cisco Finesse.
    - Valid non-expired Cisco Finesse secondary certificate must be present on the primary Cisco Finesse.
  - Import the CUCM certificate to both the primary and secondary Finesse nodes.
  - Import the IdS certificate to both the primary and secondary Finesse nodes.
  - Import the SocialMiner server certificates to both the primary and secondary Finesse nodes in the Unified CCE.

- Import the LiveData server certificates to both the primary and secondary Finesse nodes in the Unified CCE.
- Import the Cloud Connect server certificates to both the primary and secondary Finesse nodes in the Unified CCE.

You can override the trust certificate enforcement by using the CLI command **utils finesse set\_property webservices trustAllCertificates true**.

For more information on CLI commands, see *Service Properties*.



## CHAPTER 12

# Manage Finesse IP Phone Agent

- [Finesse IP Phone Agent, on page 141](#)
- [One Button Sign In, on page 142](#)
- [Finesse IP Phone Service Subscription Options, on page 143](#)
- [Set Up Application User, Web Access, and HTTPS Server Parameters, on page 144](#)
- [Configure Finesse IP Phone Service in Unified CM, on page 145](#)
- [Finesse IP Phone Agent Certificate Management, on page 146](#)
- [Add Service Parameters for One Button Sign In, on page 149](#)
- [Subscribe Agent Phones to Manual Subscription Service, on page 150](#)
- [Set Up Agent Access to the Self Care Portal, on page 150](#)
- [Finesse IP Phone Agent Login During Maintenance Mode, on page 151](#)

## Finesse IP Phone Agent

With Finesse IP Phone Agent (IPPA), agents and supervisors can access Finesse features on their Cisco IP Phones as an alternative to accessing Finesse through the browser. Finesse IPPA supports fewer features than the Finesse desktop in the browser, but it does allow agents and supervisors to receive and manage Finesse calls if they lose or do not have access to a computer.

### Supervisor Tasks

Finesse IPPA does not support supervisor tasks such as monitor, barge, and intercept, but supervisors can sign in and perform all agent tasks on their IP Phones.

### Administration Tasks

After you configure Finesse IPPA, the administration tasks that you perform for the Finesse desktop also apply for the supported Finesse IPPA features. For example, the Call Variables Layouts that you configure for the desktop also apply for Finesse IPPA, although the column layout is modified to fit the IP Phone screen.

### Reason Code Limitations

- On the IP Phone, Finesse can display a maximum of 100 Not Ready, Wrap Up, or Sign Out reason codes. If more than 100 codes are configured, the phone lists the first 100 applicable codes (global or applicable team codes).

- When Finesse IPPA displays reason codes, some IP Phone models truncate the codes due to character length limitations on the phone. To ensure they meet your requirements, verify the display of the reason codes on all phone models in your environment.

### Finesse IP Phone Agent Service Access Protocol

Finesse IPPA phone clients communicate with the Finesse server using Secure HTTP (HTTPS) protocol.

### Failure Behavior

Unlike the Finesse desktop, the Finesse IP Phone Agent does not automatically failover to the alternate Finesse server. To resume usual operations in a failure scenario, the Finesse IPPA agents must exit from the current Finesse IP Phone service and manually sign in to another configured Finesse service that connects to an alternate Finesse server.

To ensure continued operations in a failure situation, you must configure at least two Finesse IP Phone services in Unified CM, each pointing to different Finesse servers.

## One Button Sign In

With One Button Sign In, you can set up the Finesse IPPA phones with prepopulated agent ID, extension, and password. In this case, agents can sign in to Finesse on the IP Phone without credentials just by selecting Cisco Finesse from the Services menu.

Alternatively, you can set up One Button Sign In and prepopulate only a subset of agent credentials. For example:

- You can prepopulate only the agent ID and extension, forcing the agents to manually enter their password at sign-in for increased security.
- You can prepopulate only the extension, forcing agents to manually enter their ID and password at sign-in (useful for agents who share the same phone).

You can use Unified CM Administration to prepopulate the agent credentials, or you can set up the agents with access to the Unified CM Self Care Portal to prepopulate their own credentials.

The following table shows examples of how you can assign the responsibility of defining agent credentials to the administrator or the agent, or share that responsibility between them:

| Example Set Up                                    | Prepopulated in Unified CM Administration<br>(by Administrator) | Prepopulated in Self Care Portal<br>(by Agent) | Entered at Sign-In<br>(by Agent) |
|---------------------------------------------------|-----------------------------------------------------------------|------------------------------------------------|----------------------------------|
| Administrator populates the extension only        | extension                                                       | -                                              | id<br>password                   |
| Administrator populates the ID and extension      | id<br>extension                                                 | -                                              | password                         |
| Agents enter password only using Self Care Portal | id<br>extension                                                 | password                                       | -                                |

| Example Set Up                                            | Prepopulated in Unified CM Administration (by Administrator) | Prepopulated in Self Care Portal (by Agent) | Entered at Sign-In (by Agent) |
|-----------------------------------------------------------|--------------------------------------------------------------|---------------------------------------------|-------------------------------|
| Agents enter all credentials using Self Care Portal       | -                                                            | id<br>extension<br>password                 | -                             |
| Agents enter ID and extension only using Self Care Portal | -                                                            | id<br>extension                             | password                      |

## Finesse IP Phone Service Subscription Options

To set up access to Finesse on agent IP phones in Cisco Unified Communications Manager, you must create the Finesse IP Phone service to which the phones can subscribe. To set up the Finesse service, you can choose one of the following options:

- Set up an enterprise subscription to automatically subscribe all IP phones in the cluster to the Finesse service. (Not supported with One Button Sign In.)
- Set up a manual subscription, and manually subscribe each IP phone to the Finesse service.
- Set up a manual subscription, and set up the agents with access to the Unified CM Self Care Portal to subscribe to the Finesse service.

The following table lists the Finesse IPPA configuration procedures and indicates which procedures are required depending on the subscription option you choose:

| Finesse IPPA Configuration Procedures                                   | Enterprise Subscription | Manual Subscription                          |                                                                                       |
|-------------------------------------------------------------------------|-------------------------|----------------------------------------------|---------------------------------------------------------------------------------------|
|                                                                         |                         | Administrator Manually Subscribes the Phones | Agents Manually Subscribe Their Phones Using the Self Care Portal                     |
| <i>Set Up Application User, Web Access, and HTTPS Server Parameters</i> | Required                | Required                                     | Required                                                                              |
| <i>Configure Finesse IP Phone Service in Unified CM</i>                 | Required                | Required                                     | Required                                                                              |
| <i>Add Service Parameters for One Button Sign In</i>                    | Not applicable          | Required only with One Button Sign In        | Required only with One Button Sign In                                                 |
| <i>Subscribe Agent Phones to Manual Subscription Service</i>            | Not applicable          | Required                                     | Optional. Allows the administrator to enter agent credentials for One Button Sign In. |

| Finesse IPPA Configuration Procedures              | Enterprise Subscription | Manual Subscription                                                            |                                                                   |
|----------------------------------------------------|-------------------------|--------------------------------------------------------------------------------|-------------------------------------------------------------------|
|                                                    |                         | Administrator Manually Subscribes the Phones                                   | Agents Manually Subscribe Their Phones Using the Self Care Portal |
| <i>Set Up Agent Access to the Self Care Portal</i> | Not applicable          | Optional. Allows agents to enter their own credentials for One Button Sign In. | Required                                                          |

## Set Up Application User, Web Access, and HTTPS Server Parameters

To support Finesse IPPA functionality, you must configure an application user in Unified Communications Manager that is associated with all Finesse IPPA phones. For proper Finesse IPPA operation, you must also set the Web Access and HTTPS Server parameters in Unified CM.

The following steps are required for both manual and enterprise subscriptions:

### Before you begin

Set up call capabilities for the agent phones in Cisco Unified Communications Manager.

---

**Step 1** Set the following parameters in Unified CM:

- Set the **Web Access** parameter to **Enabled**.
- Set the **HTTPS Server** parameter to **HTTPS Only**.

To set these parameters in Cisco Unified CM Administration, use either of the following pages:

- Phone Configuration page (Product Specific Configuration portion of page): choose **Device > Phone**.
- Enterprise Phone Configuration page: choose **System > Enterprise Phone Configuration**.

**Step 2** Configure an application user in Unified Communications Manager.

- a) In Cisco Unified Communications Manager Administration, select **User Management > Application User**.
- b) Click **Add New**.
- c) Under User Information, enter a user ID and password for the new user.

The password must be 95 characters or less and must contain ASCII characters only.

- d) Under Device Information, in the Available Devices pane, select all phones that Finesse IP Phone Agents will use and move them to the Controlled Devices pane using the arrows.
- e) Under Permissions Information, click **Add to Access Control Group**.
- f) From the list of search results, select **Standard CTI Enabled** and **Standard CTI Allow Control Of All Devices** and then click **Add Selected**.

The application user is added to the Standard CTI Enabled and Standard CTI Allow Control Of All Devices groups.



g) Click **Save** at the bottom of the page.

**Note** In UCCX deployments, usage of an existing RMCM User for Finesse IPPA is known to cause problems in functionality, however, the physical phones must be associated with the RMCM User.

**Step 3** Enter the application user's credentials in the Finesse IP Phone Agent Settings gadget.

- a) Sign in to the Cisco Finesse Administration Console.
- b) Choose **Settings > IP Phone Agent Settings**.
- c) Under Phone URL Authentication Settings, enter the same username and password that you entered in Unified CM for the application user.

The password must be 95 characters or less and must contain ASCII characters only.

- d) Click **Save**.
- e) Restart Cisco Finesse Tomcat on the primary Finesse server.
- f) After replication is complete, restart Cisco Finesse Tomcat on the secondary Finesse server.

**Note** For Finesse IP Phone Agent (IPPA) from 11.0 (1) onwards, the User Device Profile (UDP) must be associated with the Finesse IP Phone Agent Application User along with the physical phones for agents using Extension Mobility. The Finesse Service URL must use the complete FQDN of the Unified CCX server.

---

## Configure Finesse IP Phone Service in Unified CM

The following procedure describes the steps required for manual and enterprise subscription.

---

**Step 1** Log in to the Unified CM Administration using administrator credentials.

**Step 2** Select **Device > Device Settings > Phone Services**.

**Step 3** Click **Add New** to create a new IP phone service.

**Step 4** In the **Service Name** field, enter **Cisco Finesse** (or another service name that is appropriate for your environment).

**Step 5** In the **Service URL** field, enter: `http://Finesse FQDN:8082/fippa/#DEVICENAME#`

**Note** The **Service URL** entry is mandatory for Unified CM.

**Step 6** In the **Secure-Service URL** field, enter: `https://Finesse FQDN:8445/fippa/#DEVICENAME#` to configure Finesse IP Phone Agent.

- Note**
- Support to HTTP is disabled from Cisco Finesse, Release 12.5(1) onwards. Step 5 and Step 6 are mandatory to save the Finesse IP Phone Agent settings.
  - Import certificates for Finesse IP Phone Agent to communicate with the Finesse server using Secure HTTP (HTTPS) mode. For more information, see *Finesse IP Phone Agent Certificate Management*.

**Step 7** Ensure that the **Service Category** is set to **XML Service**, and the **Service Type** is set to **Standard IP Phone Service**.

**Step 8** Check the **Enable** check box.

**Step 9** Perform one of the following:

- To automatically subscribe all phones in the cluster to the Finesse service, check the **Enterprise Subscription** check box, and click **Save**. Agents and supervisors can now access Cisco Finesse by selecting it from the **Services** menu on subscribed IP phones.

**Note** One Button Sign In is not supported with enterprise subscriptions.

- To subscribe only the desired phones to the Finesse service manually, leave the **Enterprise Subscription** check box unchecked and click **Save**.

**Step 10** With a two-node Finesse setup (primary and secondary Finesse servers), perform the preceding steps again to create a secondary Finesse service that points to the secondary Finesse server. When you create the secondary service, note the following procedural differences:

- At Step 4, in the **Service Name** field, enter a name that distinguishes the secondary service from the primary service, such as **Cisco Finesse Secondary**.
- At Step 5 and Step 6, replace *Finesse FQDN* with the FQDN of the secondary server.

**Note** The language used in Finesse IPPA is selected based on the User Locale field in Unified CM. The language selected based on the User Locale must be available in the Unified CCX language pack for Unified CCX deployments and Unified CCE pack for Unified CCE deployments.

If the language selected based on the User Locale in Unified CM is not available in the respective deployments, Finesse IPPA displays all content in the default language (U.S. English).

---

## Finesse IP Phone Agent Certificate Management

The administrator must perform the following operations to enable Finesse IP phones to communicate with the Finesse server using HTTPS.

- For a CA-signed certificate, see [CA-Signed Certificate, on page 146](#).
- For a self-signed certificate, see [Self-Signed Certificate, on page 147](#).

### CA-Signed Certificate

- 
- Step 1** Obtain the CA-signed certificate from the signed authority for both Cisco Finesse and CUCM server.
- Step 2** Import the CA-signed certificate of CUCM to the Cisco Finesse server trust store as **tomcat-trust**. For more information, see [Import CUCM Certificate, on page 147](#).
- Step 3** Import the CA-signed certificate of Cisco Finesse certificate to the CUCM trust store as **Phone-trust**. For more information, see [Import Certificate into CUCM Trust Store, on page 148](#).
-

## Self-Signed Certificate

---

- Step 1** Export the self-signed CUCM certificate from the Cisco Unified Operating System Administration. For more information, see [Export CUCM Certificate, on page 147](#).
  - Step 2** Import the downloaded self-signed CUCM certificate to the Cisco Finesse trust store as **tomcat-trust**. For more information, see [Import CUCM Certificate, on page 147](#).
  - Step 3** Export the self-signed Cisco Finesse certificate from the Cisco Unified Operating System Administration. For more information, see [Export Cisco Finesse Certificate, on page 148](#).
  - Step 4** Import the downloaded self-signed Cisco Finesse certificate to the CUCM trust store as **Phone-trust**. For more information, see [Import Certificate into CUCM Trust Store, on page 148](#).
- 

## Export CUCM Certificate

---

- Step 1** Sign in to Cisco Unified OS Administration on the CUCM server using the following URL: `https://FQDN of CUCM server:8443/cmplatform`.
  - Step 2** Select **Security > Certificate Management**.
  - Step 3** Enter the search criteria as **tomcat** and then click **Find** to filter the certificate.  
  
The tomcat certificates list is displayed. If you do not find the tomcat certificate for your server in the **Certificate List**, then click **Generate Self-signed**. When the certificate generation is complete, reboot your server. Then restart this procedure. For more information on generating the self-signed certificate, see *Cisco Unified Operating System Administration Online Help*.
  - Step 4** Click the tomcat certificate hyperlink in the **Common Name** column.  
The tomcat **Certificate Details** dialog box is displayed.
  - Step 5** Click **Download .PEM File**.
  - Step 6** Save the .PEM file in your local machine.
- 

### What to do next

Follow the same steps for both publisher and subscriber nodes. It is also to be followed for all the CUCM node certificates.

## Import CUCM Certificate

---

- Step 1** Sign in to Cisco Unified OS Administration on the Cisco Finesse server using the following URL: `https://FQDN of Finesse server:8443/cmplatform`.
- Step 2** Select **Security > Certificate Management > Upload Certificate/Certificate chain**.
- Step 3** From the **Certificate Purpose** drop-down list, select **tomcat-trust**.
- Step 4** In the **Upload File** field, click **Choose File** and browse to the tomcat.pem or CA-signed certificate file that you saved on your system.
- Step 5** Click **Upload**.

**Step 6** Restart Cisco Finesse tomcat and Cisco Finesse Notification Service.

**Note** Follow the same steps for both publisher and subscriber nodes. It is also to be followed for all the CUCM node certificates.

## Export Cisco Finesse Certificate

**Step 1** Sign in to Cisco Unified OS Administration on the Cisco Finesse server using the following URL: `https://FQDN of Finesse server:8443/cmplatform`.

**Step 2** Select **Security > Certificate Management**.

**Step 3** Enter the search criteria as **tomcat-trust** and then click **Find** to filter the certificate.

The tomcat-trust certificates list is displayed. If you do not find the tomcat certificate for your server in the **Certificate List**, then click **Generate Self-signed**. When the certificate generation is complete, reboot your server. Then restart this procedure. For more information on generating the self-signed certificate, see *Cisco Unified Operating System Administration Online Help*.

**Step 4** Click the tomcat-trust certificate hyperlink in the **Common Name** column. The tomcat **Certificate Details** dialog box is displayed.

**Step 5** Click **Download .PEM File**.

**Step 6** Save the .PEM file in your local machine.

### What to do next

Follow the same steps for both publisher and subscriber nodes. It is also to be followed for all the CUCM node certificates.

## Import Certificate into CUCM Trust Store

**Step 1** Sign in to Cisco Unified OS Administration on the CUCM server using the following URL: `https://FQDN of CUCM server:8443/cmplatform`.

**Step 2** Select **Security > Certificate Management > Upload Certificate/Certificate chain**.

**Step 3** From the **Certificate Purpose** drop-down list, select **Phone-trust**.

**Step 4** In the **Upload File** field, click **Browse** and browse to the tomcat.pem or CA-signed certificate file that you saved on your system.

**Step 5** Click **Upload**.

**Step 6** Reboot the Cisco Unified Communications Manager (CUCM) server.

**Note** Follow the same steps for both publisher and subscriber nodes. It is also to be followed for all the CUCM node certificates.

# Add Service Parameters for One Button Sign In

With One Button Sign In, for any agent credentials that you want prepopulated, you must set up corresponding service parameters in Unified CM.

Only perform this procedure if you are setting up One Button Sign In. Otherwise, skip this.

**Step 1** From Cisco Unified Communications Manager Administration, select the Finesse phone service (under **Device** > **Device Settings** > **Phone Services**).

**Step 2** Click **New** to the right of the Parameters box.

**Step 3** Set up service parameters for the agent id, extension, and password credentials as per the following table. Enter only the parameters that you want prepopulated for the agents. For each parameter, enter the required field values and click **Save**. To add parameters, click **Add New** and enter the required values.

| Field                                          | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Parameter Name</b>                          | Enter a parameter name in lower case exactly similar to — id, extension, and password.<br>The values entered are the exact query string parameters used for the subscription URL.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>Parameter Display Name</b>                  | Enter a descriptive parameter name; for example, id, extension, and password.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>Default Value</b>                           | Leave the default value blank for all parameters.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>Parameter Description</b>                   | Enter a description of the parameter. The user can access this text when they subscribe to the service.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <b>Parameter is Required</b>                   | If the administrator prepopulates the parameter in Unified CM Administration, check the <b>Parameter is Required</b> box.<br><br>However, if the agent prepopulates the parameter in the Self Care Portal, two options are available: <ul style="list-style-type: none"> <li>• If the agents prepopulates all defined parameters, check the <b>Parameter is Required</b> box for each parameter.</li> <li>• If the agent and administrator share the responsibility of prepopulating the parameters, set only the administrator-defined parameters as required. This configuration ensures that the administrator can save the subscription without prepopulating all parameters. In this case, the administrator first prepopulates the required parameters, and then the agents prepopulate the nonrequired parameters.</li> </ul> |
| <b>Parameter is a Password (mask contents)</b> | Check this box for the password only.<br><br>This check box masks the password entries in the Self Care Portal, to display asterisks rather than the user entry.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |

When you save the last parameter, click **Save and Close**.

### What to do next

You can prepopulate the agent credentials when you subscribe the agent phones, or the agents can prepopulate their own credentials using the Unified CM Self Care Portal.

## Subscribe Agent Phones to Manual Subscription Service

If you set up the Finesse service as a manual subscription, you can subscribe the agent phones to the Finesse service in Unified CM and optionally define agent credentials for One Button Sign In.

If you prefer to allow the agents to subscribe to the Finesse service using the Self Care Portal and prefer not to specify One Button Sign In credentials for the agents, you can skip this procedure.

- 
- Step 1** From the menu bar, select **Device > Phone**.
- Step 2** Select the phone that you want to subscribe to the Finesse service.
- Step 3** From the **Related Links** drop-down list on the upper right side of the window, select **Subscribe/Unsubscribe Services** and click **Go**.
- The **Subscribed IP phone services** window displays for this phone.
- Step 4** From the **Select a Service** drop-down list, select **Cisco Finesse**.
- Step 5** Click **Next**.
- Step 6** *(Applicable for One Button Sign In only)* Enter values for any of the defined service parameters (id, password, and extension) that you do not want the agents to enter using the Self Service Portal or at sign-in.
- Step 7** Click the **Subscribe** button to subscribe this phone to the Cisco Finesse service.
- The Cisco Finesse service displays in the **Subscribed Services** list.
- Step 8** Click **Save**.
- The subscribed agents or supervisors can now access Cisco Finesse by selecting it from the **Services** menu on their IP phones.
- Step 9** With a two-node Finesse setup (primary and secondary Finesse servers), perform this procedure again to also subscribe the phones to the secondary Finesse service that points to the secondary Finesse server.
- 

## Set Up Agent Access to the Self Care Portal

You can optionally set up the agents with access to the Unified CM Self Care Portal to prepopulate their own credentials and to subscribe to the Finesse service.

If you are not setting up One Button Sign In, or not enabling the agents with access to the Self Care Portal, skip this procedure.

- 
- Step 1** From the Unified CM Administration page, select **System > Enterprise Parameters**.
- Step 2** Under the Self Care Portal Parameters, in the **Self Care Portal Default Server** field, select the IP address of the Unified CM Publisher server from the drop-down list and click **Save**.

- Step 3** Select **User Management > End User**.
- Step 4** Select the user that you want to set up with access to the User Care Portal.
- Step 5** Under Permissions Information, click **Add to Access Control Group**.
- Step 6** From the list of Access Control groups displayed, check **Standard CCM End Users** and click **Add Selected**.
- Step 7** Click **Save**.

---

With access enabled to the Self Care Portal, agents can sign in to the portal at `http://<UCM address>/ucmuser` to subscribe to the Finesse service and enter their own credentials under **Phones > Phone Settings > Services**.



---

**Note** In a two-node Finesse setup with two services configured, the agents must enter their credentials on the primary and secondary Finesse services.

---

## Finesse IP Phone Agent Login During Maintenance Mode

Maintenance Mode for Cisco Finesse servers is supported starting from release 12.6(1) in order to support unscheduled down times. When the Cisco Finesse node is in maintenance mode, Finesse IPPA users are not allowed to login. Finesse IPPA users receive the following message if they attempt to login to a Cisco Finesse node which is currently under maintenance mode:

```
This Finesse Service is unavailable due to maintenance in progress. Please sign in using the alternate Finesse service.
```







## CHAPTER 13

# Manage Third-Party Gadgets

---

- [3rdpartygadget Account](#), on page 153
- [Upload Third-Party Gadgets](#), on page 153
- [Cisco Webex Experience Management](#), on page 155

## 3rdpartygadget Account

The 3rdpartygadget account is used to upload third-party gadgets to the Finesse server. Before you can use this account, you must set the password.

Run the following command to set or reset the password of the 3rdpartygadget account (where *password* is the new password for the account):

```
utils reset_3rdpartygadget_password password
```



---

**Note** Passwords cannot contain spaces or double quotes.

---



---

**Note** If you plan to upload third-party gadgets to the Finesse server, you must have a developer support services contract or work with a Cisco partner who has a developer support services contract. For more information about uploading third-party gadgets, see the *Cisco Finesse Web Services Developer Guide*.

---

## Upload Third-Party Gadgets

After you set the password for the 3rdpartygadget account, you can use SFTP to upload third-party gadgets to the Finesse server, as illustrated in the following example. Note that third-party gadget files must be .xml files. It does not support .jsp files.



---

**Note** Finesse allows you to upload third-party gadgets to your own web server, however, you must ensure that the Finesse server has access to your web server.

---

```
my_workstation:gadgets user$ sftp 3rdpartygadget@<finesse>
3rdpartygadget@<finesse>'s password:
Connected to <finesse>.
sftp> cd /files
sftp> put HelloWorld.xml
Uploading HelloWorld.xml to /files/HelloWorld.xml
HelloWorld.xml
sftp> exit
```

After you upload a gadget, it is available under the following URL:

`https://<finesse>/3rdpartygadget/files/`

To access the gadget uploaded in the previous example, use the following URL:

`https://<finesse>/3rdpartygadget/files/HelloWorld.xml`

When you add a gadget to the desktop layout, that gadget can be referenced using a relative path. For more information on adding third party gadgets to the Finesse desktop layout, see the section *Manage Desktop Layout* in the *Cisco Finesse Administration Guide*.

To include the gadget that was uploaded in the previous example in the desktop layout, add the following XML (highlighted) to the layout:

```
<finesseLayout xmlns="http://www.cisco.com/vtg/finesse">
 <layout>
 <role>Agent</role>
 <page>
 <gadget>/desktop/gadgets/CallControl.jsp</gadget>
 <gadget>/3rdpartygadget/files/HelloWorld.xml</gadget>
 </page>
 ...
 </layout>
 <layout>
 <role>Supervisor</role>
 <page>
 <gadget>/desktop/gadgets/CallControl.jsp</gadget>
 <gadget>/3rdpartygadget/files/HelloWorld.xml</gadget>
 </page>
 ...
 </layout>
</finesseLayout>
```



**Note** You cannot delete, rename or change permissions of a folder while using SFTP in 3rd party gadget accounts for Unified CCX deployments. To perform these actions, SELinux has to be in permissive mode. This can be accomplished by running the following CLI command:

**utils os secure permissive**



---

**Note** Because of browser caching and caching in the Finesse web server, you may need to clear the browser cache or restart the Cisco Finesse Tomcat service before gadget changes take effect. If you make a change to a gadget and the change is not reflected on the Finesse desktop, clear your browser cache.

If you do not see the changes after you clear the browser cache, use the following CLI command to restart the Cisco Finesse Tomcat service:

```
admin:utils service restart Cisco Finesse Tomcat
```

---

### Third-Party Gadget Limitations

Third-party gadgets must be .xml files. You cannot use .jsp files.

## Cisco Webex Experience Management

Cisco Webex Experience Management (Experience Management) is the platform for Customer Experience Management (CEM), integrated with powerful tools that allow you to see your business from your customers' perspective. Experience Management has all the sophisticated features and functionality including customer journey mapping, text analytics, and predictive modeling in a single point-n-click platform.

For more information, see the Cisco Webex Experience Management Integration section in *Cisco Unified Contact Center Enterprise Features Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html>.

Cloud Connect is a component that hosts services that allow customers to use cloud capabilities such as Cisco Webex Experience Management. The administrator can configure the Cloud Connect server settings in the Finesse administration console to contact the Cisco cloud services. For more information, see [Cloud Connect Server Settings, on page 23](#).

## Configure Experience Management Gadgets for Finesse Desktop

Experience Management Activation Team provides the details to log in to Experience Management. For more information, see [Cisco Webex Experience Management Activation](#).

Gadgets are displayed by default in the **Spaces** tab of the Experience Management. To know more about default gadgets and derive meaningful insights from them, see [Cisco Webex Experience Management Gadgets](#).

To export the Cisco Finesse gadget code from Experience Management, see [Export Cisco Finesse Gadget Code](#).

To add Experience Management gadgets in Finesse desktop layout, see [Add Experience Management Gadgets](#).

### Experience Management Gadgets—Task Activity Notification



---

**Note** This is supported from Cisco Finesse, Release 12.5(1) ES3 onwards.

---

`TaskActivityNotification` API is a mechanism by which the Finesse desktop can now provide a way for gadgets to sense the digital channel task worked on by an agent, across different media. The notifications

inform the desktop and other subscribers about which non-voice media dialog is currently selected or de-selected by the agent.

This can be used for showing or processing the relevant information associated with the task activity automatically, in supporting gadgets, which subscribe for these activity notifications, when the agent switches between different tasks.



---

**Note** The feature requires participating gadgets to publish and subscribe for the activity notifications. The Finesse desktop by itself cannot provide these notifications or provide task activity processing based on these notifications.

---

The implementation of task activity notification is provided by digital channel gadgets such as Cisco Enterprise Chat and Email (ECE). The ECE gadget provides task notifications, and Cisco Webex Experience Management gadget subscribes and displays customer journey information corresponding to the task which is currently active.

For more information on the `TaskActivityNotification` API, see *Developer Guide* at <https://developer.cisco.com/docs/finesse/#!rest-api-dev-guide>.

For more information on digital survey, see the Webex Experience Management Digital Channel chapter in *Cisco Unified Contact Center Enterprise Features Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-feature-guides-list.html>.

For more information on Cisco Enterprise Chat and Email, see <https://www.cisco.com/c/en/us/support/customer-collaboration/cisco-enterprise-chat-email/series.html>.



## CHAPTER 14

# Perform Routine Maintenance

---

- [Cisco Finesse Services](#), on page 157
- [Log Collection](#), on page 158
- [Collect Logs using Cisco Unified Real-Time Monitoring Tool](#), on page 161
- [Finesse Performance Counters](#), on page 165
- [JMX Counter Thresholds](#), on page 165
- [Remote Account Management](#), on page 168

## Cisco Finesse Services

You can access the following Finesse services from the CLI:

- **Cisco Finesse Notification Service:** This service is used for messaging and events. If this service is not started, you cannot view call events, agent state changes, or statistics, and the Finesse Desktop will not load after sign-in.
- **Cisco Finesse Tomcat:** This service contains all deployed Finesse applications. A restart of the Cisco Finesse Tomcat service requires that all agents sign out and sign back in.

The deployed applications in the Cisco Finesse Tomcat service include:

- **Finesse Desktop application:** Provides the user interface for agents and supervisors.
- **Finesse Rest API application:** Provides integration with the Cisco CTI Server for the Finesse desktop and Finesse administration application. The APIs available to a user depends on the role associated with that user's credentials. This application also provides a programming interface that can be used by third-party applications that are written to use the Finesse REST API.
- **Finesse Administration application:** Provides the administrative operations for Finesse.

If a Cisco Finesse service-related problem exists, restart a Finesse service as a last resort. Most service-related problems cannot be corrected by restarting a service. Restart A Cisco DB only if the service is down.



**Note** To restart the Cisco Finesse Notification Service, you must stop and start services in the following order:

1. Stop the Cisco Finesse Tomcat service.
2. Stop the Cisco Finesse Notification Service.
3. Start the Cisco Finesse Notification Service.
4. Start the Cisco Finesse Tomcat service.

## View, Start, or Stop Services

**Step 1** Sign in to the CLI using the credentials for the Administrator User account.

**Step 2** To view a list of all services and their states, enter the following command: **utils service list**.

Services are shown in one of the following states: STOPPED, STARTING, or STARTED.

STOPPED means the service is not running. STARTING means the service is starting operation and performing any initialization. STARTED means the service has successfully initialized and is operational.

**Step 3** To start a service, enter the following command: **utils service start** *service name*.

**Example:**

For example, to start Cisco Finesse Tomcat, enter the command **utils service start Cisco Finesse Tomcat**.

**Step 4** To stop a service, enter the following command: **utils service stop** *service name*.

**Example:**

For example, to stop Cisco Finesse Tomcat, enter the command **utils service stop Cisco Finesse Tomcat**.

## Log Collection

These commands prompt you to specify a secure FTP (SFTP) server location to which the files will be uploaded.

To obtain logs:

- Install log: **file get install desktop-install.log**

Use this command to see the installation log after the system is installed.

This log is written to the SFTP server and stored as a text file written to this path: *<IP Address>\<date time stamp>\install\desktop-install.log*

- Desktop logs: **file get activelog desktop recurs compress**

Use this command to obtain logs for the Finesse web applications. This command uploads a zip file that contains the following directories:

- **webservices:** Contains the logs for the Finesse backend that serves the Finesse REST APIs. The maximum size of an uncompressed desktop log file is 100 MB. The maximum size of this directory

is approximately 4.5 GB. After a log file reaches 100 MB, that file is compressed and a new log file is generated. Output to the last compressed desktop log file wraps to the log file created next. The log file wrap-up duration can vary, based on the number of users on the system. Timestamps are placed in the file name of each desktop log.

- **desktop:** Contains logs from the Finesse agent desktop gadget container that holds the Finesse desktop gadgets. Any container-level errors with Finesse agent desktop will appear in these log files.
- **admin:** Contains logs from the Finesse administration gadget container that holds the administration gadgets. Any container-level errors with the Finesse administration console appear in these log files.
  - **audit-log:** Audit logs contain all admin operations (including Finesse admin UI and REST client operations) and supervisor operations for Team Message. The maximum size of an uncompressed audit log file is 100 MB. The maximum size of total audit log files (including compressed log files) is approximately 1 GB. After a log file reaches 100 MB, that file is compressed and a new log file is generated. The log file wrap-up duration can vary, based on the number of users on the system. The log contains the following parameters:
    - Timestamp
    - User Id of the administrator
    - Method of operation (PUT, POST, DELETE ). GET operations will not be logged
    - URL
    - Payload
- **clientlogs:** Contains the client-side logs that are submitted from the Cisco Finesse agent desktop to the Finesse server. Each log file is no larger than 1.5 MB and contains a timestamp and the agent ID of the agent who submitted the file. A new log file is created each time that an agent submits client-side logs (the data is not appended to an existing log file). The maximum size of this directory is 100 MB. The directory holds a maximum number of 25000 clientlog files. When the directory exceeds the size limit or the file count, the oldest files are deleted.
- **openfireservice:** Contains startup and shutdown-related information logs for the Cisco Finesse Notification Service.
- **openfire:** Contains limited error and information logs for the Cisco Finesse Notification Service.
- **realm:** Contains the logs for authentication requests from clients that are handled by the Finesse backend.
- **db:** Contains the Finesse database logs.
- **/finesse/logs:** Contains the logs for the Cisco Finesse Tomcat service.
- **3rdpartygadget:** Contains information, error, startup, and shutdown-related logs for the Cisco Finesse 3rdpartygadget server.
- **jmx:** Contains the JMX counters data that is generated by the JMX logger process. It contains important jmx counters that are exposed by Finesse and openfire.
- **finesse\_maintenance\_mode.log:** Contains the logs of Cisco Finesse hook script implementation of orchestration manager.

These logs are stored in the following path on the SFTP server: *<IP address><date time stamp>\active\_nnn.tgz*, where *nnn* is timestamp in long format.

- WebProxy Service logs: **file get activelog webproxy recurs compress**

Use this command to obtain logs for the WebProxy Service. The maximum size of an uncompressed webproxy log file is 10 MB. The maximum size of this directory is approximately 500 MB. After a log file reaches 10 MB, that file is compressed and wraps to the new log file which is generated. The log file wrap-up duration can vary, based on the number of users on the system. Timestamps are placed in the file name of each webproxy log.

These logs are stored in the following path on the SFTP server: *<IP address><date time stamp>\active\_nnn.tgz*, where *nnn* is timestamp in long format.

This command uploads a zip file that contains the following log files:

- **access.log**: Contains the webproxy access logs after you configure the access log-level using the **set webproxy access-log-level** CLI. For more information on CLI commands, see *WebProxy Service*.
- **error.log**: Contains the webproxy error logs.
- **webproxy\_cli.log**: Contains the webproxy CLI logs. For more information on CLI commands, see *WebProxy Service*.
- **webproxy\_launcher.log**: Contains the logs after the WebProxy Service is launched.




---

**Note** To access the individual log file, use the command **file get activelog webproxy/<log filename>**.

For example, **file get activelog webproxy/error.log**

---

- Servm log: **file get activelog platform/log/servm\*.\* compress**

Use this command to obtain logs that are generated by the platform service manager that manages the starting and stopping of the Finesse services.

The desktop and servm logs are compressed to one set of files.

These logs are stored to the following path on the SFTP server: *<IP address><date time stamp>\active\_nnn.tgz*, where *nnn* is the timestamp in long format.

- Platform Tomcat logs: **file get activelog tomcat/logs recurs compress**

These logs are stored to the following path on the SFTP server: *<IP address><date time stamp>\active\_nnn.tgz*, where *nnn* is the timestamp in long format.

- Install log: **file get install install.log**

These logs are stored to the following path on the SFTP server: *<IP address><date time stamp>\active\_nnn.tgz*, where *nnn* is timestamp in long format.




---

**Note** Log collection may fail when you use the compress flag if there are a lot of log files. If collection fails, run the command again without the compress flag.

---



### Call Variables Logging

From Cisco Finesse Release 12.5(1) onwards, the call variables logging in Cisco Finesse logs are disabled by default. The callVariables contain sensitive user information and this property allows the administrator to decide whether the information must be captured in the logs. You can enable the call variables logging by using the CLI commands.

## Collect Logs using Cisco Unified Real-Time Monitoring Tool

Cisco Finesse supports the Cisco Unified Real-Time Monitoring Tool (RTMT) for log collection. Use the following procedure to collect logs using Unified RTMT.



---

**Note** Finesse supports RTMT only for log collection. Other RTMT features are not supported.

---

### Before you begin

Download and install RTMT on a client computer from the following URL:

<https://FQDN:8443/plugins/CcmServRtmtPlugin.exe>

where *FQDN* is the Fully Qualified Domain Name of the Finesse server.

- 
- Step 1** Log in to Unified RTMT using Finesse administrator credentials.
  - Step 2** In the tree hierarchy, select **Trace & Log Central**.
  - Step 3** Double-click **Collect Files**.  
The Trace Collection wizard appears.
  - Step 4** Select the services and Finesse nodes for which you want to collect logs, and complete the wizard.
- 

### What to do next

For detailed instructions, see the *Cisco Unified Real-Time Monitoring Tool Administration Guide*, which is listed here:

<https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/products-maintenance-guides-list.html>

## Syslog Support for Critical Log Messages

Cisco Finesse generates syslogs for critical log messages. Use the following procedure to view the logs using Unified RTMT.

**Before you begin**

Download and install RTMT on a client computer from the following URL:

<https://FQDN:8443/plugins/CcmServRtmtPlugin.exe>, where FQDN is the Fully Qualified Domain Name of the Finesse server.

- 
- Step 1** Log in to Unified RTMT using Finesse administrator credentials.
- Step 2** In the tree hierarchy, select **SysLog Viewer** or choose **System > Tools > SysLog Viewer > Open SysLog Viewer**.
- Step 3** From the **Select a Node** drop-down list, choose the server where the logs that you want to view are stored.
- Step 4** Under the **Logs** tab, select **Application Logs > CiscoSyslog** to view and save the syslog file.

**Tip** When you double-click the CiscoSyslog message, the **Show Detail** dialog displays the syslog definition and recommended actions in an adjacent pane.

For more information, see the [Cisco Unified Real-Time Monitoring Tool Administration Guide](#).

**Note** System log messages generated by Cisco Finesse are also available under **SysLog Viewer > System Logs > messages**.

The following are the different types of messages and corresponding descriptions that are captured in the **SysLog Viewer > System Logs > messages**.

- CTI\_SOCKET\_ERROR  
System has encountered an error connecting to the CTI server.
- CTI\_CONNECTION\_LOST  
System has lost contact with the CTI server.
- CTI\_OPEN\_FAILURE  
CTI Server rejected open request.
- CTI\_CONNECTION\_RETRIES\_EXCEEDED  
System has failed to connect to the CTI server in the allowed number of retries.
- CTI\_CONNECTION\_ESTABLISHED  
System has successfully connected to the CTI server.
- SUBSYS\_INIT\_ERROR  
Error initializing subsystem.
- UNABLE\_TO\_CONNECT\_TO\_XMPP\_SERVER  
Unable to connect xmpp server.
- DB\_SS\_CONNECTION\_CHECK  
There was an error connecting to the database.
- cfservice\_CORE\_ERROR\_DB\_CONNECTION  
Unable to connect to the Database.
- AWDB\_NOT\_ACCESSIBLE

Unable to connect to AWDB server.

- VOS\_DB\_ADAPTER\_ERROR  
There was an error on the VOS DB Adapter operation.
- FINESSE\_APP\_STARTUP\_ERROR  
Error during Finesse Application Startup.
- OF\_STATE\_CHANGED  
OF subsystem state successfully changed.
- CONNECTED\_TO\_XMPP\_SERVER  
Successfully connected to xmpp server.
- SSO\_API\_ERROR  
Error processing REST API Request for SSO.
- API\_ERROR\_DETAIL  
Error processing REST API request.
- AWDB\_CONNECTION\_ERROR  
Error while connecting the AWDB server.
- AWDB\_CONNECTION\_SWITCH\_SUCCESS  
AWDB server connection successfully switched.
- CTI\_DISCONNECT  
Disconnecting primary CTI connection.
- STANDBY\_CTI\_DISCONNECT  
Disconnecting standby CTI connection.
- RESET\_STANDBY\_CONNECTION  
Resetting standby CTI connection.
- RESET\_PRIMARY\_CTI\_CONNECTION  
Resetting primary CTI connection.
- SWAP\_CTI\_CONNECTIONS  
Swapping primary and standby CTI Connections.
- STANDBY\_ACMI\_CONNECTION\_ESTABLISHED  
Established connection with standby CTI.
- STANDBY\_ACTIVE\_EVENT  
Received standby active event from standby CTI.
- MAINTENANCE\_MODE\_REQUEST\_TIMEOUT  
CTI STANDBY\_ACTIVE\_EVENT isn't received before the threshold time.

- MAINTANCE\_MODE\_CONTINUE\_CTI  
CTI confirms to continue maintenance mode.
- MAINTANCE\_MODE\_DUPLICATE\_REQUEST  
Duplicate maintenance mode request received from CTI.
- FINESSE\_MAINTENANCE\_MODE\_FAILED  
Finesse current node went out of service.
- FINESSE\_MAINTENANCE\_MODE\_FAILED  
Finesse alternate node is in maintenance mode.
- FINESSE\_MAINTENANCE\_MODE\_FAILED  
Internal error occurred while performing the maintenance mode operation.
- FINESSE\_MAINTENANCE\_MODE\_FAILED  
Finesse couldn't migrate these agents {count}
- FINESSE\_MAINTENANCE\_MODE\_FAILED  
Finesse Maintenance Mode operation failed with reason {reason}
- FINESSE\_MAINTENANCE\_MODE\_REQ\_REJECTED Finesse OUT\_OF\_SERVICE  
Rejecting the maintenance mode request.
- FINESSE\_MAINTENANCE\_MODE\_REQ\_REJECTED  
CTI maintenance is in progress.
- FINESSE\_MAINTENANCE\_MODE\_REQ\_REJECTED  
Finesse maintenance is already completed.
- FINESSE\_MAINTENANCE\_MODE\_REQ\_REJECTED  
This is a single node setup.
- FINESSE\_MAINTENANCE\_MODE\_REQ\_REJECTED  
The alternate node of Finesse is OUT\_OF\_SERVICE.
- FINESSE\_MAINTENANCE\_MODE\_REQ\_REJECTED  
The alternate node of Finesse is in maintenance mode.
- FINESSE\_MAINTENANCE\_MODE\_REQ\_ACCEPTED  
Changing Finesse Maintenance Mode to IN\_PROGRESS.
- FINESSE\_MAINTENANCE\_MODE\_COMPLETED  
Finesse successfully migrated all the agents to the alternate node.
- DRAPI\_REST\_OPERATION\_JOB\_DISCARD  
Clearing the Digital Routing API rest job when operation has failed multiple times and the maximum number of retries are over.

- `DRAPI_HOST_ALERT`  
Failover of Digital Routing API host-pair.  
Failover isn't supported when the Digital Routing API host backup isn't configured.
- `DRAPIAsyncRestClient`  
Failed to create SSL connection to Digital Routing API.

## Finesse Performance Counters

You can get the JMX counters from Finesse using the following methods of Performance API:

- `list`
- `read`

For more information about these methods, refer to the *Cisco Finesse Web Services Developer and JavaScript Guide* on the [DevNet](#) site.

For the list of JMX counters that are available for digital channels, see the *JMX Counters for Digital Channels* table in the *JMX Counter Thresholds* topic.

## JMX Counter Thresholds

Cisco Finesse provides important JMX counters with associated threshold values that can be used to monitor the health of Finesse. The following tables list the JMX counters with corresponding threshold values at the login phase and steady phase (the time after the agents have logged in).

**Table 6: JMX Counters on Tomcat Processes (Port 12399) - Login Phase Thresholds**

JMX Counter	Description	JMX Object Name
<code>ThreadCount</code>	The number of threads running at the current moment.	<code>java.lang:type=...</code>
<code>PeakThreadCount</code>	The maximum number of threads run at the same time since the JVM was started or the peak was reset.	<code>java.lang:type=...</code>
<code>currentThreadCount</code>	The number of threads the thread pool currently has (both busy and free).	<code>Catalina:type=...</code>
<code>currentThreadsBusy</code>	The number of threads currently processing requests.	<code>Catalina:type=...</code>
<code>RequestLongestTime</code>	The maximum amount of time taken to complete an API request, in milliseconds.	<code>com.cisco.cel...</code>
<code>processCPULoad</code>	The CPU load in this process.	<code>java.lang:type=...</code>

JMX Counter	Description	JMX Object Name
NumOfActiveAgentsLoggedIn	The number of agents logged in with XMPP Presence as available in the current side.	com.cisco.ccbu:c
NumOfAgentsLoggedIn	The number of agents and supervisors logged in currently.	com.cisco.ccbu:c

Table 7: JMX Counters for Digital Channels

Counter Name	Description	JMX Object Name
CloseTasksRequestsInterval	Close task API requests during interval. <b>Note</b> This does not include retry request.	com.cisco.ccbu:category=DRAPITasksStatsBean,component0=DRAPITasksStats
TotalCloseTasksRequests	Total number of Close task API requests to Digital Routing API from starting of the Finesse service. <b>Note</b> This does not include retry request.	com.cisco.ccbu:category=DRAPITasksStatsBean,component0=DRAPITasksStats
TransferTasksRequestsInterval	Transfer task API requests during interval. <b>Note</b> This does not include retry request.	com.cisco.ccbu:category=DRAPITasksStatsBean,component0=DRAPITasksStats
TotalTransferTasksRequests	Total number of Transfer task API requests to Digital Routing API from starting of the Finesse service. <b>Note</b> This does not include retry request.	com.cisco.ccbu:category=DRAPITasksStatsBean,component0=DRAPITasksStats
DiscardedTasksInterval	Discarded Digital Routing API tasks during interval.	com.cisco.ccbu:category=DRAPITasksStatsBean,component0=DRAPITasksStats
TotalDiscardedTasks	Total number of discarded Digital Routing API tasks from starting of the Finesse service.	com.cisco.ccbu:category=DRAPITasksStatsBean,component0=DRAPITasksStats

Counter Name	Description	JMX Object Name
FailoversInInterval	Failovers during intervals.	com.cisco.ccbu:category= DRAPITasksStatsBean,component0= DRAPITasksStats
TotalFailovers	Total number of Digital Routing API failovers from starting of the Finesse service.	com.cisco.ccbu:category= DRAPITasksStatsBean,component0= DRAPITasksStats
PendingRequestsInQueue	Total number of Digital Routing API rest requests that are pending in queue.	com.cisco.ccbu:category= DRAPITasksStatsBean,component0= DRAPITasksStats

**Table 8. JMX Counters on Tomcat Processes (Port 12399) - Steady Phase Thresholds**

JMX Counter	Description	JMX Object Name
ThreadCount	The number of threads running at the current moment.	java.lang:type=
PeakThreadCount	The maximum number of threads run at the same time since the JVM was started or the peak was reset.	java.lang:type=
TotalCallsInSystem	The total number of active calls in the system.	com.cisco.ccbu:
AverageProcessingTime	The average time taken for processing CTI messages, in milliseconds.	com.cisco.ccbu:
currentThreadCount	The number of threads the thread pool currently has (both busy and free).	Catalina:type=
currentThreadsBusy	The number of threads currently processing requests.	Catalina:type=
RunnablesQueued	Runnables (CTI Messages) still queued.	com.cisco.ccbu:
TasksQueued	The tasks (such as client requests and CTI messages) queued.	com.cisco.ccbu:
RequestLongestTime	The maximum amount of time taken to complete an API request, in milliseconds.	com.cisco.ccbu:
processCPULoad	The CPU load in this process.	java.lang:type=

The following table lists the thresholds for counters related to Openfire processes.

Table 9: Counters Related to Openfire (JMX Port 12348)

JMX Counter	Description	JMX Object Name
ExecutingTaskCount	The number of tasks (messages published to node) that are run currently.	com.cisco.ccbu.f
QueuedTaskCount	The number of tasks in the queue. Messages that are getting published to a node are placed in the queue.	com.cisco.ccbu.f
PeakThreadCount	The maximum number of threads run at the same time since the JVM was started or the peak was reset.	java.lang:type =
ThreadCount	The number of threads running at the current moment.	java.lang:type =
processCPULoad	The recent CPU usage for the Java Virtual Machine process.	java.lang:type =

## Remote Account Management

Run the following command to enable, disable, create, and check the status of a remote access account:

### **utils remote\_account**

A remote account generates a passphrase that allows Cisco support personnel to get access to the system for the specified life of the account.

- **utils remote\_account create** *account life*  
*account* is the account name. *life* indicates the life of the account in days.
- **utils remote\_account disable**
- **utils remote\_account enable**
- **utils remote\_account status**





## CHAPTER 15

# Cisco Finesse Failover Mechanisms

---

- [CTI Failover, on page 169](#)
- [AWDB Failover, on page 171](#)
- [Finesse Desktop Failover, on page 171](#)
- [Desktop Behavior, on page 172](#)
- [Finesse IP Phone Agent Failover, on page 177](#)
- [Guidelines for Optimal Desktop Failover, on page 178](#)
- [Failover Planning, on page 181](#)
- [Maintenance Mode, on page 183](#)

## CTI Failover

CTI failover is when the Finesse server disconnects from one CTI server and reconnects to the same or another CTI server.

The prerequisites for successful CTI failover are as follows:

- Unified Contact Center Enterprise (Unified CCE) must be configured in duplex mode.
- The B Side CTI host and port must be configured through the Finesse administration console.

In the duplex mode, if Finesse loses connection to CTI server, it attempts to connect to the server which is running. Finesse alternates between the configured servers until it makes a successful connection.

While failover is in progress, Finesse transitions to `OUT_OF_SERVICE` state. During this period, Finesse does not entertain client requests or send out events. Any request made during this time receives a 503 Service Unavailable error message.

After reconnecting to a CTI server and transitioning to `IN_SERVICE` state, Finesse responds to client requests and publishes events.

Connection to the CTI server can be lost due to the following reasons:

- Finesse misses three consecutive heartbeats from the connected CTI server (heartbeat interval is five seconds).
- Finesse socket that is opened to the CTI server fails.

After the failover is complete, the last state of call control, call data, or agent state are published as events to all clients. This allows Finesse clients to reflect an accurate view of the call control, call data, and agent state.

If an agent either makes or answers a call, and then ends that call during failover (that is, the entire call takes place during failover), the corresponding events are not published.



**Note** An agent or supervisor who signs in after being on an active conference with other devices (which are not associated with another agent or supervisor) may experience unpredictable behavior with the Finesse desktop due to incorrect call notifications from Unified CCE. These limitations also encompass failover scenarios where a failover occurs while the agent or supervisor is participating in a conference call. For example, an agent is in a conference call when the Finesse server fails. When the agent is redirected to the other Finesse server, that agent may see unpredictable behavior on the Finesse desktop. Examples of unpredictable behavior include, but are not limited to, the following:

- The Finesse desktop does not show all participants in a conference call.
- The Finesse desktop does not show that the signed-in agent or supervisor is in an active call.
- The Finesse receives inconsistent call notifications from Unified CCE.

Despite these limitations, the agent and supervisor can continue to perform general operations on the phone. Desktop behavior returns to usual after the agent or supervisor drops off the conference call.

### Prevent Non-Voice Task RONAs during CTI Reconnect

When CTI disconnection happens, the agent state is changed to WORK, on the respective non-voice Media Routing Domain (MRD), to prevent tasks getting routed to the disconnected agents. Previous releases of Unified CCE used to change the agent states back to an available state when the CTI connection is re-established, even though the media handling gadgets and the media channels are not initialized by then.

The media handling gadgets, and the media channels are initialized only after the Finesse desktop failover completes.

Due to the significant delay in desktop failing over after the Finesse server reconnects to the CTI server, chances of occurrence of RONA (Redirection on No Answer) are high when dealing with non-voice tasks.

Unified CCE, Release 12.5(1) or later allows the agent state to remain in WORK mode after CTI reconnection. This allows the agents to change to an available state in non-voice MRD explicitly after the Finesse desktop and media channels are initialized. This avoids the task being routed to the user before the agent is ready to handle non-voice media tasks.

By default, Cisco Finesse Release 12.5(1) retains the earlier behavior, which can be modified using the **enableAutoWorkModeStateChange** property. By default, this property is set to *true*, and the administrator can set to *false* to change to the new behavior.



**Note** This behavior is supported from Unified CCE Release 12.5(1) onwards, and only after the relevant non-voice gadgets or custom desktop or clients support this behavior.

The agents remain in the WORK mode until they are explicitly set to active on the respective MRD using the REST API. This informs the CTI that the media channel is available (and connected) and the tasks can be routed to the respective user on that MRD.

The `Media-Change Agent from Work State to Active API` allows a user to change the agent state from `WORK` state to active (`READY` or `NOT_READY`), which is automatically computed by Unified CCE. Users can only use this API when an agent state is `WORK`.

## AWDB Failover

The prerequisites for AWDB failover are as follows:

- The secondary Administrative Workstation Database (AWDB) is configured.
- The secondary AWDB host is configured through the Cisco Finesse administration console.

Agents and supervisors are authenticated against the AWDB database. When an agent or supervisor makes a successful API request (such as a sign in or call control request), the credentials are cached in Cisco Finesse for 30 minutes from the time of the request. After a user is authenticated, that user continues to be authenticated until 30 minutes pass, even if both AWDBs are down. Cisco Finesse attempts to reauthenticate the user against the AWDB only after the cache expires.

If Cisco Finesse loses connection to the primary Administration & Data server, and the preceding prerequisites have been implemented, AWDB failover occurs. After Cisco Finesse loses connection to the primary Administration & Data server, it tries to reconnect to the secondary server.

Cisco Finesse repeats this process for every API request until it can connect to one of the Administration & Data servers. During failover, Cisco Finesse does not process any requests, but clients can still receive events.

## Finesse Desktop Failover

Desktop failover can occur for the following reasons:

- When the Finesse desktop loses network connectivity to the Finesse Notification Service.
- When the Finesse Tomcat Service becomes *Unavailable*
- When the Finesse REST API Service becomes *Unavailable*
- When the Finesse Notification Service becomes *Unavailable*
- When the Finesse loses connection to CTI servers



---

**Note**

- After the failover, the pending state of an agent will not be displayed once the agent fails over to the subscriber. The pending state change is lost during the failover, as the agent will be logged out, and logged in again.
  - Finesse is `IN_SERVICE`, coordinates the distribution of desktop reloads, such that failover and consequent desktop reloads are evenly distributed to prevent overwhelming of the Finesse server. Configuration data such as reason codes, workflows and so on are not reloaded during failover to improve the performance.
-

If the server that an agent is connected transitions to `OUT_OF_SERVICE`, the agent receives a notification that the connection with the server is lost. The Finesse desktop:

- Checks whether the subscriber is available and `IN_SERVICE`.
- Continues to check whether the publisher recovers its state.

If the subscriber is available, then the desktop automatically signs the agent into the subscriber. If the publisher recovers its state, the desktop notifies the agent that it has reconnected.

The failover logic has three triggers to detect desktop failure:

- The Finesse desktop receives a `SystemInfo` event that the publisher is `OUT_OF_SERVICE`.
- The Finesse Notification Service is disconnected.
- The XMPP presence of “Finesse” user changes to *Unavailable*.

No matter which trigger is detected, the desktop reconnection logic is as follows:

1. Poll `SystemInfo` for publisher every 20 seconds.
2. If `SystemInfo` API reports Finesse is `IN_SERVICE`, check the Finesse Notification Service.
3. If `SystemInfo` is `IN_SERVICE`, check whether the last CTI heartbeat status of the side being connected is a success.




---

**Note** The last CTI heartbeat status is checked to ensure that the subscriber is healthy before failover, and thus does not immediately transition to `OUT_OF_SERVICE` after the client has failed over. This may occur in CTI failure, since both Finesse servers connect to the same PG and CTI server, and a CTI failure can cause both Finesse servers to disconnect and connect to the alternate PG. Depending on the network topology the subscriber might be slower to sense a network disconnect.

---

4. If XMPP is disconnected, make the Finesse Notification Service request.
5. If the Finesse Notification Service is successful and Finesse service is `IN_SERVICE`, refresh the data.

The failover logic prefers to stay with the publisher. If the failover logic detects that the subscriber is available, it checks the publisher one more time. If the publisher has recovered, the desktop reconnects to the publisher. If the publisher is still down, the desktop connects the agent to the subscriber. In this case, the agent does not automatically reconnect to the failed server after it recovers, but instead remains connected to the subscriber.

If the Finesse Notification Service is the source of failure, the desktop makes three attempts to reconnect before changing the state of the desktop to disconnected. These attempts occur before the failover logic begins.

## Desktop Behavior

Cisco Finesse sends a code of 255 to the CTI server and you may see a different code on the CTI server side. The actual behavior of the desktop under these conditions depends on the setting for Logout on Agent Disconnect (LOAD) in Unified CCE. By default, the CTI server places the agent in Not Ready state.



**Note** Finesse takes up to 120 seconds to detect when an agent closes the browser. If the browser crashes, Finesse waits 60 seconds before sending a forced logout request to the CTI server. Under these conditions, Finesse can take up to 180 seconds to sign out the agent.

The following table lists the conditions under which Finesse sends this code to the CTI server.

Scenario	Desktop Behavior	Server Action	Results
The agent closes the browser, the browser crashes, or the agent clicks the Back button on the browser.	Finesse desktop makes a best-effort attempt to notify the server.	Finesse receives a presence notification of <i>Unavailable</i> from the client. Finesse waits 60 seconds, and then sends a forced logout request to the CTI server.	<p><b>Race Conditions</b></p> <ol style="list-style-type: none"> <li>1. The agent closes the browser window. Finesse receives a presence notification of <i>Unavailable</i> for the user. Finesse tries to sign the agent out; however, that agent is already signed out.</li> <li>2. If the browser crashes, it can take the Finesse server up to 120 seconds to detect that the client is gone and send a presence notification to Finesse. A situation can occur where the client signs into the subscriber before the publisher receives the presence notification caused by the browser crash. In this case, the agent may be signed out or put into Not Ready state on the subscriber.</li> <li>3. If the Finesse desktop is running over a slower</li> </ol>

			<p>network connection, Finesse may not always receive an <i>Unavailable</i> presence notification from the client browser. In this situation, the behavior mimics a browser crash, as described in the preceding condition.</p> <p>4. If agent is in Not Ready State before Failover, agent moves to Not Ready — Connection Failure after CTI Disconnect or Reconnect.</p> <p>If agent is in Ready State before Failover, agent moves to Not Ready — Connection Failure upon his next state change to Not Ready.</p> <p><b>Non-voice media channel handling</b></p> <p>If agent is in Active or Not Ready state before Failover, agent moves to previous (Active or Not Ready) state after Failover.</p>
The client refreshes the browser	—	Finesse receives a presence notification of <i>Unavailable</i> from the client. Finesse waits 60 seconds before sending a forced logout request to the CTI	—

		server to allow the browser to reconnect after the refresh.	
The client encounters a network glitch (Finesse is IN_SERVICE)	Connection to the Finesse server temporarily goes down, consequently the client fails over to the subscriber.	The publisher receives a presence notification of <i>Unavailable</i> from the client. Finesse is IN_SERVICE, so it sends a forced logout request to the CTI server for the agent.	<p><b>Race Conditions</b></p> <p>A situation can occur where the forced logout does not happen before the client signs in to the subscriber. If the agent is on a call, the publisher sends the forced logout request after the call ends. The agent will be signed out or put into Not Ready state when the call ends, even though the client is already signed in to the subscriber.</p> <p>If agent is in Not Ready State before Failover, agent moves to Not Ready — Connection Failure after CTI Disconnect or Reconnect.</p> <p>If agent is in Ready State before Failover, agent moves to Not Ready — Connection Failure upon the next state change to Not Ready.</p> <p><b>Non-voice media channel handling</b></p> <p>If agent is in Active or Not Ready state before Failover, agent moves to previous (Active or Not Ready) state after Failover.</p>
The Refresh Token has expired. For more information on tokens, see <a href="https://developer.cisco.com/docs/finesse/#single-sign-on-apis">https://developer.cisco.com/docs/finesse/#single-sign-on-apis</a> .	Finesse desktop sends a forced logout request to the CTI server.	The Finesse server forwards the forced logout request to the CTI server.	The session expiry warning appears 10 minutes and 5 minutes before the Refresh Token expires. In the last minute, a

countdown timer appears till the Refresh Token expires. The agent is forcefully logged out when the timer reaches zero and must log in again.

For Unified CCE, the state of the agent changes to Log Out or Not Ready based on the Load parameter set as below.

**Load parameter = 0**

- When the agent's current state is Not Ready, Ready, or Wrap-Up, the agent's state after force logout is changed to Not Ready – Connection Failure.
- When the agent's current state is Talking, the Agent goes into Not Ready – Connection Failure state after the call ends.

**Load parameter = 1**

- When the agent's current state is Not Ready, Ready, or Wrap-Up, the agent goes to Logged Out – System Failure.
- When the agent's current state is Talking, the Agent goes to Logged Out –



			<p>System Failure immediately even though the call is still active.</p> <p><b>Non-voice media channel handling</b></p> <p>If agent is in Active or Not Ready state before Failover, agent moves to previous (Active or Not Ready) state after Failover.</p>
--	--	--	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



**Note** For more information on non-voice states, refer to **Agent States for Nonvoice Media** section in Task Routing Developer Guide at <https://developer.cisco.com/docs/>.

### Desktop Chat Failover

The following table lists the desktop chat failover scenarios:

Failover Type	Desktop Chat Behavior
Cisco IM&P server failover	The desktop chat status is retained, and all active chat sessions are lost.
Finesse server failover	The desktop chat status is retained, and all active chat sessions are lost.
CTI server failover	The desktop chat status and all chat sessions are retained.

## Finesse IP Phone Agent Failover

Finesse IPPA failover can occur for the following reasons:

- The Finesse REST API Service transitions to OUT\_OF\_SERVICE.
- The Finesse Notification Service transitions to OUT\_OF\_SERVICE.
- If Finesse IPPA detects a server failure before Finesse fails over to the alternate CTI server, then Finesse IPPA declares the Finesse server OUT\_OF\_SERVICE.

The server that an agent is connected transitions to OUT\_OF\_SERVICE, the Finesse IP Phone Agent (IPPA) displays a notification that the server is unavailable. Finesse IPPA continues to check whether the current Finesse server recovers its state and notifies the agent if it reconnects.

Finesse IPPA attempts to reconnect to the server every 5 seconds and declares it `OUT_OF_SERVICE` after three failed attempts. The total time required for the transition to `OUT_OF_SERVICE` is approximately 15 seconds.

Unlike the Finesse desktop, Finesse IPPA does not check whether the subscriber is available. To connect to subscriber, the agent must exit the publisher, and manually sign into the subscriber.

Finesse IPPA failover logic has the following two triggers to detect failure:

- Finesse IPPA receives a `SystemInfo` event that the publisher is `OUT_OF_SERVICE`.  
Finesse IPPA polls `SystemInfo` every 5 seconds to check whether the Finesse server is `IN_SERVICE`. After three attempts, if the Finesse server is not `IN_SERVICE`, Finesse IPPA displays a server unavailable message to the agent.
- Finesse IPPA receives notification that the Finesse Notification Service is disconnected.  
Finesse IPPA tries every 5 seconds to reconnect with the XMPP server. After three attempts, if the Finesse Notification Service cannot be reestablished, Finesse IPPA displays a server unavailable message to the agent.

While the agent is still signed into the current service, Finesse IPPA continues attempting to reestablish the connections with the Finesse and XMPP servers. If they both resume service, Finesse IPPA displays the **Sign In** screen and the agent can sign in again and continue as usual.

Alternately, the agent must exit the current Finesse service and try to connect using an alternate Finesse service.

## Guidelines for Optimal Desktop Failover

The following are the guidelines to optimize failover scenarios, to avoid server overload and unnecessary delays.

- [Browser Configuration, on page 178](#)
- [Agent Configuration, on page 179](#)
- [Finesse Configuration, on page 180](#)
- [Agent PG Configuration, on page 180](#)
- [CUIC Configuration, on page 180](#)
- [Common Configuration Safeguards, on page 181](#)



---

**Note** The guidelines for optimal failover ensure that desktop initialization time and general system performance is optimized.

---

### Browser Configuration

Finesse browser failover performance depends on the number of requests made to the Finesse server. Fewer the requests, lesser the system load on the Finesse server. The following browser-specific configurations

ensure that the browser does not fetch static resources unnecessarily from the server, and it is a key requirement for faster failover.



**Note** Clear the browser cookies before logging in to the Finesse desktop. This avoids unexpected expiry of the Refresh Token in the Single Sign-On mode for Unified CCE.

- Avoid loading the Finesse desktop with `bypassServerCache=true&nocache` as a query parameter in the desktop URL. The `bypassServerCache` is to bypass Webproxy cache, and `nocache` is to bypass Shindig cache.
- Host systems must have at least 200 MB of free disk space more than the free space required by the operating system (OS).
- Adequate network bandwidth must be available between the Finesse desktop and the Finesse server. Lower latency results in faster failover.

For more information on bandwidth measurements, see *Finesse Bandwidth Calculator for Unified Contact Center Enterprise* and *Cisco Unified Contact Center Express Bandwidth Calculator* at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-technical-reference-list.html>.

- Host systems must have adequate memory and CPU without being overloaded at any point in time. A slow host slows the browser enough to cause it to fail and reload resources randomly during failover.
- External gadget hosting servers must prefer CA-signed certificates for easy integration with the browser. If they are self-signed, then import those certificates into the agent browser.

For more information, see *Accept Security Certificates* section in *Cisco Finesse Agent and Supervisor Desktop User Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-user-guide-list.html>.

### Firefox Configurations

Disable the Race Cache With Network (RCWN) in all the agent desktops to avoid any unwanted requests to the Finesse server. If RCWN is enabled, the Firefox browser by-passes the cached data and fetches the static requests again from the server. Set the `network.http.rcwn.enabled` configuration as `false`.

For more information, see <https://support.mozilla.org/en-US/questions/1267945>.

### Google Chrome, Internet Explorer, and Edge Chromium (Microsoft Edge) Configurations

Import the Finesse self-signed certificates on Google Chrome, Internet Explorer, and Microsoft Edge browsers trust store.

For more information, see the *Accept Security Certificates* section in *Cisco Finesse Agent and Supervisor Desktop User Guide for Cisco Unified Contact Center Express* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-express/products-user-guide-list.html>.

### Agent Configuration

Agents configured must be evenly distributed between the publisher and subscriber. This prevents all agents from failing over when there is an outage that affects only one of the deployed Finesse server.

The number of agents failing over impacts system load and has a linear relationship with the maximum time taken for the operation to complete.

## Finesse Configuration

The number of signed-in users, the gadget types, and the average number of gadgets configured per user, significantly impacts failover load.

The following are the best practices for ensuring a trouble-free failover.

- **Number of Gadgets per Agent**—Gadget-initiated requests constitute the bulk of the requests made during Finesse desktop failover or startup. Configuring fewer gadgets in the desktop layout results in faster desktop failover and startup. The administrator must configure the team or desktop layouts such that only the required gadgets for each team are available in the desktop layout.
- **Type of Gadget**—XML-based gadgets load much faster than gadgets served using a dynamic JSP-based URL. The gadget content is also cached at the WebProxy Service, which allows the Finesse server to scale further. The JSP-based gadgets take thrice the time to load than the XML-based gadgets.

The Unified CCE deployments must ensure that Cisco Unified Intelligence Center (CUIC) 12.5(1) servers are deployed, and CUIC JSP-based URLs are replaced using the CLI **utils finesse layout updateCuicGadgetUrl** to reduce latency and improve performance.

- **Finesse Server Capacity**—Deployments with 2000 active users and configuring more than eight XML-gadgets per user on average, or more than six JSP-based gadgets per user on average, are recommended to deploy Cisco Finesse OVA with 8 vCPUs.

When OVA with 8 vCPUs is configured for Finesse, the time taken for CTI server/Agent PG failover and desktop failover improves by 20 percent. This configuration is supported on all deployment types including the 24000 Agent deployment type. For more information on OVA with 8 vCPUs, see *Virtualization for Cisco Finesse* at [https://www.cisco.com/c/dam/en/us/td/docs/voice\\_ip\\_comm/uc\\_system/virtualization/virtualization-cisco-finesse.html](https://www.cisco.com/c/dam/en/us/td/docs/voice_ip_comm/uc_system/virtualization/virtualization-cisco-finesse.html).

- **Gadget Configuration**—Gadgets developers must follow certain best practices to ensure that gadgets load faster.

For more information, see *Best Practices for Gadget Development* section of *Cisco Finesse Web Services Developer Guide* at <https://developer.cisco.com/docs/finesse/#best-practices-for-gadget-development>.

- **Secondary Resources**—The preloading of server resources reduces latency and improves performance during desktop failover. By default, the **preLoadSecondaryResources** property is enabled. Disabling this property impacts failover time. For more information, see *Desktop Properties*.

## Agent PG Configuration

Upgrading to the latest Agent PG version allows faster detection of failure, which results in faster failover. When Cisco Finesse connects to Agent PG 11.6(x) and 12.0(x) versions, it incurs a delay of up to 40 seconds to detect CTI failover compared to Agent PG 12.5(1) or later.

## CUIC Configuration

CUIC 12.5 (1) or later supports only XML-based gadgets corresponding to the JSP-based gadget URLs that supported the real-time reports.

Cisco Finesse 12.5 (1) provides CLI **utils finesse layout updateCuicGadgetUrl**, which automatically changes the JSP references to XML with no functional changes.

Switching to XML-based gadgets reduces latency and improves performance. After the CUIC or Coresident deployment installation, run the command to optimize the faster failover.

### Common Configuration Safeguards

- Import the self-signed certificates into the browser.
- Do not disable browser caching for Finesse desktop.
- Do not clear the cache every time the browser is launched.
- Distribute the agents between the publisher and subscriber.

## Failover Planning

### CTI Failover

CTI failover happens when the Finesse server disconnects from the CTI server/Agent PG due to network failure or server error. In these scenarios, the Finesse server is unavailable to its clients. If the desktop is connected, it displays that the server is unavailable and tries to reconnect to the available Finesse server.

The duration required for Finesse CTI failover depends on the following factors.

- Agent PG 12.5(1) or later versions results in faster failover.
- Available bandwidth from the CTI server to the Finesse server.
- Round Trip Time (RTT) between the CTI server and the Finesse server in case of WAN deployments.
- The number of signed-in users.
- The number of gadgets configured.
- The number of active non-voice tasks.

When deployed with Agent PG 12.5(1) or later, CTI server/Agent PG failover varies approximately from 35 seconds to 75 seconds. When deployed with Agent PG 12.0(1) or 11.6(1), CTI server/Agent PG failover varies approximately from 75 seconds to 120 seconds. The numbers indicated varies depending on the customer configuration and the above-stated factors. It must be used as a guideline to determine the approximate range for failover time.



---

**Note** The time indicated does not include CTI server/Agent PG server recovery time. It only indicates the time taken for Finesse to reconnect and be IN\_SERVICE, once an active Agent PG is detected.

---

### Desktop Failover

The Finesse desktop failover happens in all failure scenarios. The Finesse desktop tries to find an active server and fails over to it, once it has located a reachable server which is IN\_SERVICE.

The duration required for Finesse desktop failover depends on the following factors.

- Bandwidth available to the client to reach the Finesse server.
- RTT between the client and the Finesse server.
- The number of signed-in users.

- The number of gadgets configured in the desktop per user.
- Type of gadgets (XML) and the resources it loads. For more information on Finesse gadgets, see <https://developer.cisco.com/docs/finesse/#finesse-gadgets>.
- The number of vCPU configured on the Finesse server.
- Agent PG version.
- The time required for the upstream CTI and the Finesse servers to become reachable and be IN\_SERVICE.

The desktop failover performance improvements in Cisco Finesse 12.5(1) are available in all releases irrespective of the Agent PG version.

When deployed with Agent PG 12.5(1) or later, desktop failover with the default desktop layout varies approximately from 50 seconds to 110 seconds. When deployed with Agent PG 12.0(1) or 11.6(1), desktop failover is approximately 40 seconds more when compared to Agent PG 12.5(1) or later.

The numbers indicated varies depending on the customer configuration and the above-stated factors. It must be used as a guideline to determine the approximate range for failover time.

The following table displays the time taken for failover for different configurations (illustration purpose only) using the out-of-the-box gadgets with Agent PG 12.5(1).

Affected Users	Number of Gadgets (XML)	LAN or WAN	Time Taken (seconds)
1000	6	LAN	50-55
2000	6	LAN	70-80
2000	6	WAN (RTT of 400 milliseconds)	80-110


**Note**

- The time indicated does not include the Finesse or CTI server recovery time. It only indicates the time taken for the desktop to reconnect all agents post detecting a reachable Finesse server, which is IN\_SERVICE.
- Deployments with 2000 active users and configuring more than eight XML-gadgets per user on average, or more than six JSP-based gadgets per user on average, are recommended to deploy Cisco Finesse OVA with 8 vCPUs.
- During failover, agents are redirected to the subscriber and are signed in automatically, and desktop is reloaded. Expected bandwidth utilization reaches up to approximately 250 Mbps for 90 seconds (peak), to ensure all 2000 agents failover successfully from one side to another. The bandwidth requirements increase depending on the type and number of gadgets configured for teams.

# Maintenance Mode

## Agent PG Maintenance Mode

Cisco Finesse Release 12.6(1) in conjunction with Cisco Unified CCE 12.6(1) supports PG maintenance mode, which allows the Cisco Finesse server to reconnect to an alternate PG without interrupting the current operations. When the PG maintenance mode is initiated, Finesse desktop users and the Finesse IPPA users do not see any interruption during the sign in, state operations, or call operations.

The PG maintenance mode operation attempts to move the Cisco Finesse connections to an alternate CTI server or PG and is designed to complete without any disruptions to the Cisco Finesse clients. However in the event of an error, Cisco Finesse disconnects all its active connections and rediscovers an active CTI server. For more information, see section *Unified CCE Agent PG Graceful Shutdown* in *Administration Guide for Cisco Unified Contact Center Enterprise, Release 12.6(1)* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-maintenance-guides-list.html>.

## Finesse Maintenance Mode

Cisco Finesse Release 12.6(1) in conjunction with Cisco Unified CCE 12.6(1) supports maintenance mode in Finesse. Administrators can transition active Cisco Finesse nodes into maintenance mode for performing administrative tasks without causing any disruption to contact center activities. This feature is implemented as an automatic phased migration of the Cisco Finesse desktops from the primary node to the secondary node with minimal disruption to the agent activities. Finesse Maintenance Mode can be initiated using the `finesse maintenance initiate` CLI. For more information about the CLI, see the [Finesse Maintenance Mode Services](#) section.



---

**Note** Cisco Finesse maintenance mode is different from the PG maintenance mode, which is used to switch the active CTI server to support PG maintenance mode activities.

---

### Desktop Reload Behavior During Finesse Maintenance Mode

- Finesse Desktops - Agents logged into the Finesse desktop connected to the node scheduled for maintenance are alerted about pending reloads in advance. The agents are alerted again before the actual reload so that any pending work can be saved, just before the desktop reloads and connects to the alternate Finesse node. The actual reload of the desktop ranges between 5 to 15 seconds for individual agents.
- Finesse IP Phones - Agents logged into Cisco Finesse on Finesse IPPA phones are logged out after repeated scheduled warnings and are expected to manually re-login to the alternate node.



---

**Note** **Deployments using Self-Signed Certificates for the Cisco Finesse Desktop**

The desktop agents must accept both primary and secondary certificates available in the Cisco Finesse server during login. If the secondary Cisco Finesse server certificates are not accepted during login, these individuals are not migrated and this can cause the migration to be reported as abandoned. However, in this situation, administrators can review the remaining individual users connected to the desktop from the **Connected Agents** tab in Cisco Finesse Administrator and request them to reconnect to the alternate server.

---



---

**Note** Automatic migration of third-party desktops during maintenance mode is not supported by Cisco Finesse. If Cisco Finesse is configured to support third-party desktop migration during the maintenance mode, Cisco Finesse expects all the third-party agents to be migrated in 60 minutes. The maintenance mode is marked as failed if any of the agents are not logged out or migrated within the specified time.

---

### Third-party Gadgets

Third-party gadget developers can register with client services to receive notifications (*FINESSE\_MAINTENANCE\_MODE\_EVENT*) when maintenance mode is scheduled. These notifications are triggered 15 seconds before the desktop reload. The DesktopCache API can be used to save and restore the gadget state after reloading to the alternate node of the Cisco Finesse server.

### Finesse Maintenance Mode Failure

Finesse maintenance mode can fail due to the following reasons:

- Desktop fails to reconnect to the alternate node in the scheduled interval and after attempted retries. This can happen if the secondary Cisco Finesse server is not available or if the agent has not accepted the secondary certificates.
- Delay in completing the maintenance mode can also cause the Finesse maintenance mode to be abandoned. By default, this is set to 1 hour and can be configured using the CLI command **utils finesse set\_property webservices maxMMTimeSec**.
- Alternate Cisco Finesse node's unavailability or disconnection can cause the Finesse maintenance mode operation to be abandoned.





## CHAPTER 16

# Backup and Restore

---

- [Backup and Restore](#), on page 185
- [Important Considerations](#), on page 186
- [SFTP Requirements](#), on page 186
- [Primary and Local Agents](#), on page 187
- [Backup Tasks](#), on page 188
- [Restore the Nodes in HA Setup with Rebuild](#), on page 189

## Backup and Restore

Cisco Finesse uses the backup and restore tools that are provided by the common Cisco Unified Communications platform services for complete data backup-and-restore capabilities. Cisco DRS allows you to perform regularly scheduled automatic or user-invoked data backups and to restore data if the system fails.

To access the Disaster Recovery System (DRS) application, direct your browser to the following URL: `https://Finesse Server IP:8443/drf`, where *Finesse Server IP* is the IP address of your Finesse server.



---

**Note** Cisco Finesse does not support One-Step Restore with the DRS application.

---

In the case of high availability (HA), Cisco DRS performs a cluster-level backup, which means that it collects backups for all servers to Cisco Finesse and archives the backup data to a remote SFTP server.

DRS backs up and restores its own settings, that is, backup device settings (saved in file `drfDevice.xml`) and schedule settings (saved in file `drfSchedule.xml`) as part of the platform component. Once a server is restored with these files, you do not need to reconfigure DRS backup device and schedule settings.



---

**Note** Cisco DRS uses the SSL-based communication between the Primary Agent and the Local Agent for authentication and encryption of data between the publisher and subscriber nodes. Cisco DRS uses the IPsec certificates for its Public/Private Key encryption. If you delete the IPsec truststore (`hostname.pem`) file from the Certificate Management pages, then Cisco DRS will not work as expected. If you delete the IPsec-trust file manually, then you must ensure that you upload the IPsec certificate to the IPsec-trust. For more information about the certificate management, see, *Cisco Unified Communications Manager Security Guide* at <https://www.cisco.com/c/en/us/support/unified-communications/unified-communications-manager-callmanager/products-maintenance-guides-list.html>.

---

## Important Considerations

Following are the important considerations when you perform the backup and restore procedures:

- Before you run a backup or a restore, make sure that both nodes in a cluster are running the same version of Cisco Finesse. If different nodes are running different versions, you will have a certificate mismatch and your backup or restore fails.
- Before you restore Cisco Finesse, make sure that the hostname, IP address, DNS configuration, version, and deployment type matches the hostname, IP address, DNS configuration, version, and deployment type of the backup file that you want to restore.
- Before you restore Cisco Finesse, ensure that the version that is installed on the server matches the version of the backup file that you want to restore. Cisco DRS supports restore only for matching versions of Cisco Finesse. For example, Cisco DRS does not allow you to restore from Version 8.5(1).1000-1 to Version 9.0(1).1000-1, or from Version 8.5(2).1000-1 to Version 9.0(1).1000-2.
- Schedule backups during off-peak hours to avoid call-processing interruptions and impact to service.
- After you use the recovery disk to bring a server with a corrupted file system into a bootable and semi-functional state, rebuild the server.



---

**Note** If you do not rebuild the server, you may notice missing directories, lost permissions, or corrupted soft links.

---

## SFTP Requirements

To back up data to a remote device on the network, you must have an SFTP server that is configured and accessible from the Cisco Finesse node to run the backup. Cisco allows you to use any SFTP server products that have been certified with Cisco through the Interoperability Verification Testing (IVT) process. Cisco Developer Network (CDN) partners, such as GlobalSCAPE, certify their products with a specified version.

Cisco uses the following servers for internal testing. You may use one of the servers, but you must contact the vendor for support:

- Open SSH (see <http://sshwindows.sourceforge.net/>)
- Cygwin (see <http://www.cygwin.com/>)
- Titan (see <http://www.titanftp.com/>)

Cisco does not support use of the SFTP product freeFTPD, because it has a 1-GB file-size limit.

**Note**

- For issues with third-party products that have not been certified through the IVT process, contact the third-party vendor for support.
- While a backup or restore is running, you cannot perform any Operating System (OS) Administration tasks because Cisco DRS blocks all OS Administration requests. However, you can use CLI commands to back up or restore the system.

## Primary and Local Agents

The system automatically starts the Primary Agent service on each node of the cluster, but it is functional only on the first node. Both servers in the Cisco Finesse cluster must have Local Agent running to perform the backup and restore functions.

**Note**

By default, a Local Agent automatically gets activated on each node of the cluster.

## Primary Agent Duties

The Primary Agent (PA) performs the following duties:

- Stores system-wide component registration information.
- Maintains a complete set of scheduled tasks in an XML file. The PA updates this file when it receives updates of schedules from the user interface. The PA sends tasks (that can be run) to the applicable Local Agents, as scheduled. Local Agents run immediate backup tasks without delay.
- Lets you perform activities such as configuring backup devices, scheduling backups by adding new backup schedules, viewing or updating an existing schedule, displaying status of the schedules that are run, and performing system restoration.
- Stores backup data on a remote network location.

## Local Agent Duties

In the Cisco Finesse cluster, the Local Agent runs backup and restore scripts on each node in the cluster.

**Note**

Cisco DRS uses an SSL-based communication between the Primary Agent and the Local Agent for authentication and encryption of data between the publisher and subscriber nodes. Cisco DRS uses IPsec certificates for its Public/Private Key encryption. This certificate exchange is handled internally; you do not need to make any configuration changes to accommodate this exchange.

# Backup Tasks

You can perform the following backup tasks using Cisco DRS:

- Create and manage backup devices
- Create and manage backup schedules
- Perform manual backup and check backup status
- Estimate size of backup tar file
- View history of last 20 backups

## Manage Backup Devices

Before using Cisco DRS, you must configure the locations where the backup files will be stored. You can configure up to ten backup devices. Perform the following steps to configure backup devices.

- 
- Step 1** Access the DRS application (<https://Finesse server IP:8443/drf>).
- Step 2** Select **Backup > Backup Device**.
- Step 3** Click **Add New** to add a new device or click the device name to edit settings of an existing backup device.
- Step 4** Enter the backup device name and select destination. For more details on the field description, see the detailed online help provided with the DRS application.
- Step 5** Click **Save**.

**Note** You cannot delete a backup device that is configured as the backup device in a backup schedule.

---

## Manage Backup Schedules

You can create up to ten backup schedules. Each backup schedule has its own set of properties, including a schedule for automatic backups, and a storage location.



---

**Caution** Schedule backups during off-peak hours to avoid call-processing interruptions and impact to service.

---

- 
- Step 1** Access the DRS application (<https://Finesse server IP:8443/drf>).
- Step 2** Select **Backup > Scheduler**.
- Step 3** Click **Add New** to add a new schedule or click the schedule name to edit settings of an existing backup schedule.
- Step 4** Enter the backup schedule name, select the backup device, and select feature as **Finesse**.
- Step 5** Enter the backup date and frequency details as required. For more details on the field description, see the detailed online help provided with the DRS application.

**Step 6** Click **Save**.

**Step 7** Select a schedule from the **Schedule List** and then click **Enable Selected Schedules**.

- Note**
- If you plan to schedule a backup on a two-node deployment, ensure that both the servers in the cluster are running the same version of Cisco Finesse and are communicating in the network. Servers that are not communicating at the time of the scheduled backup will not be backed up.
  - Do not schedule a backup to run while the **Update Database Statistics** task is running. By default, this task is set to run every Saturday at 3:00 a.m. and Shrink-repack on Sunday at 3:00 a.m..

---

## Perform Manual Backup

**Step 1** Access the DRS application ([https://Finesse\\_server\\_IP:8443/drf](https://Finesse_server_IP:8443/drf)).

**Step 2** Select **Backup > Manual Backup**.

**Step 3** Select a backup device and feature as **Finesse**.

**Step 4** Click **Start Backup** to start the manual backup.

**Note** Click **Estimate Size** to get the approximate size of the disk space that the backup file consumes on the SFTP server.

To perform backup tasks on virtual machines, see *Unified Communications VMware Requirements*, at: [https://www.cisco.com/c/dam/en/us/td/docs/voice\\_ip\\_comm/uc\\_system/virtualization/virtualization-software-requirements.html](https://www.cisco.com/c/dam/en/us/td/docs/voice_ip_comm/uc_system/virtualization/virtualization-software-requirements.html).

---

## Check Backup Status

**Step 1** Access the DRS application ([https://Finesse\\_server\\_IP:8443/drf](https://Finesse_server_IP:8443/drf)).

**Step 2** Select **Backup > Current Status** to check the backup status.

**Caution** The backup to the remote server to be completed within 20 hours otherwise the backup session times out and you will have to start the fresh backup.

---

## Restore the Nodes in HA Setup with Rebuild

In a high availability (HA) setup, if a hard-drive failure or other critical hardware or software failure occurs, you may need to rebuild the primary and the secondary Finesse nodes (publisher and subscriber node). Perform the following steps to restore the Finesse nodes to its last backed up state.



---

**Caution** Cisco Finesse data can only be retrieved from the backup file. The recent Finesse configuration data, which is not backed up, must be manually configured in the Cisco Finesse administration console after the restore.

---

- 
- Step 1** Perform a fresh installation of Finesse. Make sure to install the same version of Finesse, using the same administrator credentials, network configuration, and security password that you used for the initial installation.
- Step 2** Access the DRS application (<https://Finesse server IP:8443/dfs>).
- Step 3** From the Restore menu, select **Restore Wizard**.
- Step 4** Select a backup device. Choose the location where your backup is stored.
- Step 5** Select the backup file and feature as **Finesse**.
- Step 6** When prompted to choose the nodes, either choose both nodes or choose each node to individually restore them.
- Step 7** After the restore process is complete, restart the node.
- Step 8** Run the following command on the primary Finesse server:  
**utils dbreplication stop all**
- Step 9** Run the following CLI command on the primary Finesse server to set up replication:  
**utils dbreplication reset all**

**Note** The dbreplication reset command can take some time to complete.

Run the CLI command **utils dbreplication runtimestate** on the primary Finesse node. If the RTMT counter value for replication status is 2 on all nodes, replication is functioning properly.



---

**Note** After the installation is complete, check that the dbreplication is functioning and allowing the data to propagate from the primary to the secondary node. However, if you need to restore third-party gadgets to the secondary node, you must either upload them again or run the restore process on the secondary node.

Always check the dbreplication status after any restore, using the CLI command **utils dbreplication runtimestate**.

---



## CHAPTER 17

# Supported Cisco Unified Communications OS Services

---

- [Supported Cisco Unified Communications OS Services, on page 191](#)

## Supported Cisco Unified Communications OS Services

The following sections list the Cisco Unified Communications OS services that Cisco Finesse supports. For more information about CLI commands, see [Command Line Interface Guide for Cisco Unified Communications Solutions](#).



---

**Note** Other commands listed in the *Command Line Interface Guide for Cisco Unified Communications Solutions* are not tested or qualified for Finesse. Some of those commands may return only platform-specific information. Others may not work for Finesse. Finesse supports only the commands from the guide that are listed here.

Some of these commands may warn about invalidating a software license. As Finesse is not a licensed server, you can disregard these warnings.

---

### File Commands

- file check
- file delete
- file get
- file list
- file search
- file tail
- file view

### Show Commands

- show account

- show date
- show disk usage
- show hardware
- show logins
- show myself
- show network
- show network ipprefs
- show open
- show packages
- show perf
- show status
- show tech all
- show tech dberrcode
- show tech gateway
- show tech locales
- show tech params
- show tech prefs
- show tech repltimeout
- show tech runtime
- show tech systables
- show tech systems
- show tech version
- show timezone
- show trace
- show version
- show network ipv6 settings
- show tls server min-version
- show tls client min-version

### Utils Commands

- utils core active list
- utils core inactive list



- `utils csa enable`
- `utils csa disable`
- `utils csa status`
- `utils dbreplication clusterreset`
- `utils dbreplication dropadmindb`
- `utils dbreplication forcedatasyncsub`
- `utils dbreplication reset`
- `utils dbreplication runtimestate`
- `utils dbreplication setrepltimeout`
- `utils dbreplication stop`
- `utils diagnose test`
- `utils firewall ipv4`
- `utils iostat`
- `utils network arp`
- `utils network capture eth0`
- `utils network connectivity`
- `utils network host`
- `utils network ping`
- `utils network traceroute`
- `utils ntp`
- `utils ntp config`
- `utils ntp restart`
- `utils ntp server add`
- `utils ntp server delete`
- `utils ntp server list`
- `utils ntp status`
- `utils ntp start`
- `utils remote_account`
- `utils reset_application_ui_administrator_name`
- `utils reset_application_ui_administrator_password`
- `utils service`
- `utils system`

- `utils system boot`
- `utils system restart`
- `utils system upgrade`
- `utils vmtools status`

### Set Commands

- `set network ipv6 gateway`
- `set network ipv6 service disable`
- `set network ipv6 service enable`
- `set network ipv6 static_address`
- `set tls server min-version <version>`
- `set tls client min-version <version>`



---

**Note** Cisco SNMP integration with Finesse is restricted to platform MIBs. Finesse does not have any application-specific MIBs.

---



## APPENDIX **A**

### Cisco Finesse CLI

---

- [Cisco Finesse Services](#), on page 195
- [Finesse Log Configuration](#), on page 196
- [Toaster Notifications](#), on page 200
- [Finesse IPPA Inactivity Timeout](#), on page 200
- [Configuring Queue Statistics](#), on page 201
- [Cross-Origin Resource Sharing \(CORS\)](#) , on page 201
- [Gadget Source Allowed List](#), on page 204
- [Supported Content Security Policy Directives](#), on page 206
- [Finesse System Commands](#) , on page 207
- [Desktop Properties](#), on page 208
- [Service Properties](#), on page 220
- [Log Collection Schedule](#), on page 226
- [Upgrade](#), on page 227
- [Shutdown](#), on page 228
- [Replication Status](#), on page 228
- [View Property](#) , on page 229
- [Update Property](#) , on page 229
- [Signout from Media Channels](#), on page 230
- [Finesse Maintenance Mode Services](#), on page 230
- [ConnectedUsersInfo](#), on page 231
- [set webapp session maxlimit](#), on page 231
- [set webapp session timeout](#), on page 232
- [Update Cloud Connect Connection Time](#), on page 234
- [AI Services Configuration](#), on page 234
- [Digital Channels Configuration](#), on page 236
- [Certificate Configuration](#), on page 237

### Cisco Finesse Services

To view, start, or stop services:

- **show network all detail** : View the platform TCP/IP services, UDP services, and Unix domain sockets used by Cisco Finesse:

- **utils service list**: This command retrieves a list of all services and their status.

Services are shown in one of the following states:

STOPPED means the service is not running. STARTING means the service is starting operation and performing any necessary initialization. STARTED means the service has successfully initialized and is operational.

- **utils service start *service name***: This command starts the named service.
- **utils service stop *service name***: This command stops the named service.
- **utils service start Cisco Finesse Tomcat**: This command starts Cisco Finesse Tomcat.
- **utils service stop Cisco Finesse Tomcat**: This command stops Cisco Finesse Tomcat.
- **utils service restart Cisco Finesse Tomcat**: This command restarts Cisco Finesse Tomcat.




---

**Note** If a Cisco Finesse service-related problem exists, restart the Finesse service. Note that most service-related problems cannot be corrected by restarting a service.

---

## Finesse Log Configuration

Use the following CLI commands to add, delete, update, or view the logger configuration in the system for Finesse.

### utils finesse log configuration add

Creates a custom log configuration in the Finesse system. The logs record information about the encountered issues of different severity levels for a specific Finesse module.

#### Command Syntax

**utils finesse log configuration add** [*module*] [*name*] [*level*]

#### Options



- 
- Note**
- Adding multiple module names, log configuration names, and log configuration level values are not supported.
  - Log configuration with name ROOT is not allowed.
- 
- *module*—Unique name of Finesse module for which log configuration has to be added. The module name is case sensitive. The following are the valid Finesse modules.
    - *admin*—Finesse administration module.
    - *audit*—Finesse audit module for all administration (including Finesse admin UI and REST client) and supervisor operations.

- *desktop*—Finesse desktop module.
  - *diagnostics*—Finesse diagnostics module.
  - *FIPPA*—Finesse IP Phone Agent (IPPA) application module.
  - *realm*—Finesse realm module.
  - *shindig*—Shindig web application module.
  - *valve*—Finesse valve module.
  - *webservices*—Finesse webservices module.
- *name*—Package name or fully qualified class name of the Finesse application. The name is case sensitive.
  - *level*—Defines the different severity level associated with the log configuration. The following are the valid log configuration levels.
    - *OFF*—Turns off the severity level.
    - *ERROR*—Sets the severity level to error.
    - *WARN*—Sets the severity level to warning.
    - *INFO*—Sets the severity level to information.
    - *DEBUG*—Sets the severity level to debug.
    - *TRACE*—Sets the severity level to trace.
    - *ALL*—Sets the severity level to all.




---

**Note** Setting the log configuration level to *DEBUG* or *TRACE* impacts system performance. This must be done in consultation with Cisco support to ensure that the modules with high log output are not be enabled with *TRACE* levels in production servers.

---

### Example

The following is the sample output for creating the log configuration named *com.cisco.cc.common.subsystem* under the Finesse *webservices* module with log configuration level as *DEBUG*.

```
admin:utils finesse log configuration add webservices com.cisco.cc.common.subsystem DEBUG

Warning: Creating the custom log configurations may affect the performance of the Finesse
system.

Press ENTER to continue. Press any other key to exit :

Creating the log configuration, please wait...

Successfully added the log configuration. Changes might take approximately 30 seconds to
take effect..
```

**utils finesse log configuration update**

Updates an existing custom log configuration in the Finesse system.



- 
- Note**
- Updating multiple module names, log configuration names, and log configuration level values are not supported.
  - Audit log configuration cannot be updated.
- 

**Command Syntax**

**utils finesse log configuration update** *[module] [name] [level]*

**Options**

- *module*—Unique name of Finesse module for which log configuration has to be updated. The module name is case sensitive. For more information on the Finesse modules, see [utils finesse log configuration add](#).
- *name*—Package name or fully qualified class name of the Finesse application. The name is case sensitive.
- *level*—Defines the different severity level associated with the log configuration. For more information on the severity levels, see [utils finesse log configuration add](#).



- 
- Note** Setting the log configuration level to DEBUG or TRACE impacts system performance.
- 

**Example**

The following is the sample output for updating the log configuration named *com.cisco.cc.common.subsystem* under the Finesse *webservices* module with log configuration level as *TRACE*.

```
admin:utils finesse log configuration update webservices com.cisco.cc.common.subsystem TRACE
Warning: Updating the log configuration level to DEBUG or TRACE may affect the performance
of the Finesse system.
Press ENTER to continue. Press any other key to exit :
Updating the log configuration, please wait...
Successfully updated the log configuration. Changes might take approximately 30 seconds to
take effect.
```

**utils finesse log configuration delete**

Deletes an existing custom log configuration in the Finesse system.



- 
- Note**
- ROOT log configurations cannot be deleted.
  - Deleting multiple log configuration names are not supported.
-

## Command Syntax

**utils finesse log configuration delete** [*module*] [*name*]

### Options

- *module*—Unique name of the Finesse module. The module name is case sensitive.
- *name*—Package name or fully qualified class name of the Finesse application. The name is case sensitive.

### Example

The following is the sample output for deleting the log configuration named *com.cisco.cc.common.subsystem* under the Finesse *webservices* module.

```
admin:utils finesse log configuration delete webservices com.cisco.cc.common.subsystem
Deleting log configuration, please wait...
```

```
Successfully deleted the log configuration. Changes might take approximately 30 seconds to
take effect.
```

## utils finesse log configuration list

Lists all log configurations in the Finesse system.

### Command Syntax

**utils finesse log configuration list**

### Example

The following is the sample output for all the log configuration in the Finesse system.

```
admin:utils finesse log configuration list
Requesting log configurations, please wait...
Below is the list of log configurations in Finesse.
```

No.	Module	Level	Name
1.	admin	ROOT	
2.	audit	ROOT	INFO
3.	desktop	ROOT	INFO
4.	diagnostics	ROOT	INFO
5.	FIPPA	ROOT	INFO
6.	realm	ROOT	INFO
7.	shindig	ROOT	INFO
8.	valve	ROOT	INFO
9.	webservices	ROOT	INFO
10.	FIPPA	org.jivesoftware	WARN
11.	webservices	org.hibernate	INFO
12.	webservices	com.cisco.cc.common.subsystem	

## Toaster Notifications

Toaster notifications are enabled by default after a fresh installation of Cisco Finesse. Use the following CLI commands to disable, enable, and check the status of the toaster notifications:

- **utils finesse toaster enable [closeTimeout]**: This command enables the Cisco Finesse toaster notification.

While enabling toaster notification, use the **closeTimeout** parameter (timeout in seconds) to set the time interval after which toaster automatically closes. If no parameter is specified, timeout is set to 8 seconds by default. The valid range for timeout activity is between 5-15 seconds. The browser must be refreshed for timeout changes to take effect.



**Note** The configured timeout for browser notifications depends on the operating system and browser settings. The timeout value is honored in Chrome browser in Windows OS. However, the other supported browsers do not honor the configured notification timeout value consistently.

- **utils finesse toaster disable**: This command disables the Cisco Finesse toaster notification.
- **utils finesse toaster status**: This command displays the status (enable or disable) of the Cisco Finesse toaster notification.



**Note** Cisco Finesse Toaster Notification does not work with Internet Explorer browser.

## Finesse IPPA Inactivity Timeout

Use the following CLI commands to enable or disable the Inactivity Timeout feature in Finesse IPPA. You must either disable the Finesse Inactivity Timeout feature or increase the timeout in the range of 120 seconds to one day (in seconds), so that the Finesse IPPA agent is not logged out if on any other screen:

- **utils finesse ippa\_inactivity\_timeout enable**: This command enables Finesse IPPA Inactivity Timeout.



**Note** The default time set for inactivity timeout is 120 seconds.

- **utils finesse ippa\_inactivity\_timeout disable**: This command disables Finesse IPPA Inactivity Timeout.



**Note** When inactivity timeout is disabled, you will not be logged out of Finesse IPPA, if the agent is on any other screen.



- **utils finesse ippa\_inactivity\_timeout enable inactivity\_timeout**: This command enables the Finesse IPPA Inactivity Timeout with timeout set to n seconds.



---

**Note** Minimum value of n must be 120 seconds and maximum value can be up to one day (86400 seconds).

---

- **utils finesse ippa\_inactivity\_timeout status**: This command checks the status of Finesse IPPA Inactivity Timeout.



---

**Note** The Finesse IPPA Inactivity Timeout CLIs should be run on primary and secondary Finesse servers. Enabling or disabling this feature requires a restart of Cisco Finesse Tomcat, and restart must be done in the maintenance window. During upgrade, the inactivity timeout configuration is not retained and should be re-configured post upgrade.

To know how this feature works on specific IP phone models, see <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-device-support-tables-list.html>

---

## Configuring Queue Statistics

The Queue Statistics gadget is enabled by default as part of Cisco Finesse new installation (Unified CCE only). When performing a system upgrade from Cisco Finesse 11.5(1), the desktop custom layout needs to be modified by the administrator for the Queue Statistics gadget to be displayed on the Agent and Supervisor desktop.

Use the following CLI commands to enable and disable the queue statistics polling or check the status of the queue statistics polling:

- **utils finesse queue\_statistics enable**
- **utils finesse queue\_statistics disable**
- **utils finesse queue\_statistics status**

After performing a system upgrade, during switch-version the queue statistics polling will be enabled by default. The procedure to disable the queue statistics polling remains the same.



---

**Note** When enabled, Queue Statistics supports a maximum of 2000 users (Agents and Supervisors).

---

## Cross-Origin Resource Sharing (CORS)

In a fresh install of Cisco Finesse, CORS mode is in a permissive state (**enable\_all**) by default, which permits CORS preflight requests from browser-based applications from any domain. You can configure the CORS

mode to be more restrictive by changing the mode to **enable** and by adding the required browser origins to be allowed using the following CORS CLIs.




---

**Important** After you make changes to the CORS status or to the allowed origin list, restart Cisco Finesse Tomcat and Notification Services for the changes to take effect.

---

- **utils finesse cors enable:** This command allows CORS for Cisco Finesse APIs and OpenFire requests for allowed origin list. It responds to browser CORS preflight requests and allows valid domains to make Finesse API/OpenFire requests.



- 
- Note**
- Use the **utils finesse cors allowed\_origin** CLI to customize the allowed origin list.
  - Any custom headers used in the CORS requests must be added using **utils finesse cors allowed\_headers** CLI.
- 

- **utils finesse cors enable\_all:** This command allows all origins to make cross domain requests. It responds and allows CORS preflight requests from any domain to make Finesse API/OpenFire requests.



- 
- Note** This isn't a secure configuration and is included only to support backward compatibility.
- 

- **utils finesse cors disable:** This command restricts CORS for Cisco Finesse APIs and OpenFire requests. It disallows or prevents CORS preflight requests from any external domain to make Finesse API and OpenFire requests.



- 
- Note** If the allowed origin list is already present, the list is preserved and used when CORS is enabled. The CLI changes are reflected only after you clear your cache and close and reopen the browser.
- 

- **utils finesse cors status:** This command displays the CORS status (enable\_all, enabled, or disabled) on the console.

For allowing any other header, the following set of CLI commands are added to enable CORS for both Cisco Finesse and OpenFire and to configure the allowed origin list:

- **utils finesse cors allowed\_origin list:** This command lists all the origins in the allowed origin list.
- **utils finesse cors allowed\_origin add:** This command adds origins to the allowed origin list. Origins can be added by using a comma-separated string. For example:

```
utils finesse cors allowed_origin add https://origin1.com:[port]
```

```
utils finesse cors allowed_origin add https://origin1.com: [port], https://origin2.com:[port]
```



- 
- Note**
- The wildcard character star (\*) isn't a valid origin in the allowed origin list.
  - The maximum number of characters (cumulative) that are permissible in allowed origin is 4000.
- 

- **utils finesse cors allowed\_origin delete:** This command deletes origins from the allowed origin list.




---

**Note** Delete lists all the origins in the allowed origin list. The origins can be deleted by selecting the appropriate ones from the list. For example:

**utils finesse cors allowed\_origin delete**

1: http://google.com

2: https://www.cisco.com

3: https://def.com

4: https://abc.com:7777

a: all

q: quit

Select the index of origin(s) to be deleted [1-4 or a,q]

---

By default the following headers are allowed and exposed:

- **allowed\_headers:** Content-Type, X-Requested-With, accept, Origin, Authorization, Access-Control-Request-Method, Access-Control-Request-Headers, requestId, Range.
- **exposed\_headers:** Access-Control-Allow-Origin, Access-Control-Allow-Credentials, Access-Control-Allow-Methods, Access-Control-Allow-Headers, Access-Control-Max-Age.




---

**Note** These headers can't be modified. Custom headers can be added or removed using the following CLIs:

---

- **utils finesse cors allowed\_headers list:** This command lists all the allowed headers for CORS. The list is used to validate incoming requests to Finesse.
- **utils finesse cors allowed\_headers add:** This command adds one or more allowed headers for CORS. Multiple headers can be added as a comma-separated string. For example:
  - `utils finesse cors allowed_headers add header1`
  - `utils finesse cors allowed_headers add header1,header2,header3`




---

**Note** The wildcard character star (\*) isn't supported.

---

- **utils finesse cors allowed\_headers delete:** This command lists the choices for deleting the allowed headers. The choice should be an index as displayed in the list of allowed headers. The list provides the option to delete a single header or all configured custom headers. For example:

**utils finesse cors allowed\_headers delete**

1: header1

2: header2

a: all

q: quit

Select the index of the allowed header to be deleted [1-2 or a,q]: 1

- **utils finesse cors exposed\_headers list:** This command lists all the exposed headers for CORS. The list will be used by the browser to validate the accessible headers in the response.

- **utils finesse cors exposed\_headers add:** This command adds one or more exposed headers for CORS. Multiple headers can be added by a comma-separated string. For example:

utils finesse cors exposed\_headers add header1

utils finesse cors exposed\_headers add header1,header2,header3




---

**Note** The wildcard character star (\*) isn't supported

---

- **utils finesse cors exposed\_headers delete:** This command lists the choices for deleting the exposed headers. The choice should be an index as displayed in the list of allowed headers. The list provides option to delete a single header or all configured custom headers. For example:

**utils finesse cors exposed\_headers delete**

1: header1

2: header2

a: all

q: quit

Select the index of the exposed header to be deleted [1-2 or a,q]: 1

All CLIs are node specific and must be run on all nodes in the cluster.

## Gadget Source Allowed List

Shindig proxies requests from the Finesse desktop to external servers and this introduces the possibility of server side request forgery (SSRF). To prevent SSRF, you can choose to allow outgoing connections for specified sources to be used in the gadgets by adding URLs to the allowed list. Note that this functionality is disabled by default for Cisco Finesse.

Use the following CLIs to enable or disable Gadget Source allowed list functionality and to configure source(s) in the allowed list:

- **utils finesse gadget\_source\_check enable:** This command enables allowed list for Cisco Finesse.

- **utils finesse gadget\_source\_check disable**: This command disables allowed list for Cisco Finesse.
- **utils finesse gadget\_source\_check status**: This command prints the allowed list status (enabled or disabled) on Cisco Finesse console.
- **utils finesse gadget\_source\_check allowed\_list list**: This command lists all the source(s) in the allowed list.
- **utils finesse gadget\_source\_check allowed\_list add**: This command adds source(s) to the allowed list. For example,
  - **utils finesse gadget\_source\_check allowed\_list add** <https://www.abc.com:8445>.
  - **utils finesse gadget\_source\_check allowed\_list add** <https://www.abc.com:8445>,  
<http://www.abc.com>.



**Note** Wildcard character \* is not supported.

The allowed list feature does not perform hostname resolutions. The format of the allowed list entry should match the format in which the gadget requests for a resource.

If **utils finesse gadget\_source\_check** is enabled, you must add the CUIC URLs to **utils finesse gadget\_source\_check allowed\_list** for the stock gadgets to load. For example,

- `utils finesse gadget_source_check enable`
- `utils finesse gadget_source_check allowed_list add https://<CUIC_Pub_FQDN>`
- `utils finesse gadget_source_check allowed_list add https://<CUIC_Pub_FQDN>:8444`
- `utils finesse gadget_source_check allowed_list add https://<CUIC_Sub_FQDN>`
- `utils finesse gadget_source_check allowed_list add https://<CUIC_Sub_FQDN>:8444`

If you do not add the CUIC URLs, Finesse Desktop fails to load and an appropriate error message is displayed.

- **utils finesse gadget\_source\_check allowed\_list delete**: This command deletes source(s) from the allowed list. For example:
  - **utils finesse gadget\_source\_check allowed\_list delete**
    - 1: <http://origin1:8080>
    - 2: <https://origin2:7777>
    - a: all
    - q: quit

Select the index of origin to be deleted [1-2 or a,q]: 1




---

**Note** All CLIs are node-specific and must be run on all nodes in the cluster.

After any changes are done to gadget source status or to the allowed list, restart Cisco Finesse Tomcat for changes to take effect.

---

## Supported Content Security Policy Directives

Content Security Policy (CSP) is a standardized set of security directives that can inform the browser of the policies to be used to help mitigate various forms of attacks. CSP frame-ancestor policy defines the allowable locations from where the Finesse desktop can be accessed as an embedded HTML content, which can help prevent click-jacking attacks.

Use the following CLI commands to view, add, or delete the frame-access sources in the response header of Cisco Finesse. This ensures that only the configured sources can embed the Cisco Finesse in an iFrame within their HTML pages.




---

**Note** Internet Explorer does not support frame-ancestors, and therefore will not block any websites from loading Cisco Finesse within it.

---

- **utils finesse frame\_access\_allowed\_list add** [*source1,source2*]—This command adds one or more frame sources, thereby allowing the configured sources to embed the Cisco Finesse in their iFrames. Multiple sources can be provided as a comma-separated list. The source should be of the following format:

- https://<fqdn>:[port]
- https://IP:[port]
- https://<fqdn1>:port, https://<fqdn2>:port



- 
- Note**
- Wildcard character \* is also supported for the FQDN and port entries, which indicates that all the legal FQDN or ports are valid.
  - The maximum number of characters (cumulative) that are permissible in allowed list is 2000.
- 

```
admin:utils finesse frame_access_allowed_list add
https://www.abc.com:8445,https://*.abc.com,https://*.abc.com:*,https://10.21.255.25
```

Source(s) successfully added.

Ensure Source(s) is added to the frame access list in all Finesse nodes in the cluster.

Restart Cisco Finesse Tomcat and Cisco Finesse Notification Service for the changes to take effect:

```
utils service restart Cisco Finesse Tomcat
utils service restart Cisco Finesse Notification Service
```

- **utils finesse frame\_access\_allowed\_list delete**—This command displays an indexed list of all the configured frame sources that have been allowed to access Cisco Finesse. Enter the corresponding index number to delete a single source or all the configured sources.

```
admin:utils finesse frame_access_allowed_list delete
```

```
1: https://www.abc.com:8445
2: https://*.abc.com
3: https://*.abc.com:*
4: https://10.21.255.25
a: all
q: quit
```

```
Select the index of source to be deleted [1-4 or a,q]: 1
Sources deleted successfully.
```

```
Restart Cisco Finesse Tomcat and Cisco Finesse Notification Service for the changes to
take effect:
```

```
utils service restart Cisco Finesse Tomcat
utils service restart Cisco Finesse Notification Service
```

- **utils finesse frame\_access\_allowed\_list list**—This command lists all the frame sources that are allowed to access Cisco Finesse.

```
admin:utils finesse frame_access_allowed_list list
```

```
The following source(s) are configured in the frame access list:
```

```
1: https://www.abc.com:8445
2: https://*.abc.com
3: https://*.abc.com:*
4: https://10.21.255.25
```

## Finesse System Commands

Configure the following Cisco Finesse system CLIs:

### Notifications

Use the following CLI commands to enable or disable the Cisco Finesse notifications. By default, this feature is disabled.

- To enable: **utils finesse notification logging enable**
- To disable: **utils finesse notification logging disable**

### Node Statistics

Use the following CLI command to view the run-time statistics for the current node.

- To view: **utils finesse node\_statistics list**

```
admin:utils finesse node_statistics list
```

```
Warning: Running this command frequently will affect system performance.
Press ENTER to continue. Press any other key to exit :
```

```
Wait while the statistics (updated every 5 secs) are being fetched...
```

```
The following are the runtime statistics for the current node.
```

```

Active Dialogs Count: 0
Active Tasks Count: 0
Average Configured Media per Agent Count: 0
Average Logged in Media per Agent Count: 0
Average Skill Groups per Agent Count: 0
Max Skill Groups per Agent Count: 0
Total Time for Finesse to Start (in seconds): 32
Logged in Agents on current node: 0
Unique Configured Skill Groups per Agent Count: 0

```

For more information, see *RuntimeConfigInfo API Parameters* section in the *Cisco Finesse Web Services Developer Guide* at <https://developer.cisco.com/docs/finesse/>.

### Locked Out Users

Use the following command to list the locked out users:

- To list: **utils finesse locked\_out\_users list**

```

admin: utils finesse locked_out_users list

Total locked out users : 2

1) agent1
2) agent3

```

If there are no locked out users:

```

admin: utils finesse locked_out_users list

Total locked out users : 0

```

## Desktop Properties

Configure the desktop properties using the following CLIs for the features.




---

**Note** Refresh the browser for the changes to take effect.

---

### Active Call Details in the Team Performance Gadget

Use the following CLI commands to enable or disable the active call details:

- To enable: **utils finesse set\_property desktop showActiveCallDetails true**
- To disable: **utils finesse set\_property desktop showActiveCallDetails false**

### View History in the Team Performance Gadget



Use the following CLI commands to enable or disable the agent history:

- To enable: **utils finesse set\_property desktop showAgentHistoryGadgets true**
- To disable: **utils finesse set\_property desktop showAgentHistoryGadgets false**

### Force Wrap-Up Reason

Use the following CLI commands to enable or disable the force wrap-up reason:



---

**Note** This is applicable to both voice and non-voice channels.

---

- To enable: **utils finesse set\_property desktop forceWrapUp true**
- To disable: **utils finesse set\_property desktop forceWrapUp false**

### Show Wrap-Up Timer

Use the following CLI commands to show or hide the timer in wrap-up state:



---

**Note** This is applicable to both voice and non-voice channels.

---

- To hide the timer in wrap-up state: **utils finesse set\_property desktop showWrapUpTimer false**
- To display the timer in wrap-up state: **utils finesse set\_property desktop showWrapUpTimer true**

By default, the value of this property is set to true.

### Wrap-Up Timer Count Down

Use the following CLI commands to set the wrap-up timer to count down or count up the time:



---

**Note** This is applicable to both voice and non-voice channels.

---

- To count up the time: **utils finesse set\_property desktop wrapUpCountDown false**
- To count down the time: **utils finesse set\_property desktop wrapUpCountDown true**

By default, the value of this property is set to true.

### Wrap-Up Button for All Call Types

Use the following CLI command to enable the Wrap-Up button for all call types:

**utils finesse set\_property desktop enableWrapupButtonForAllCallTypes true**

By default, the value of this property is false.

During outbound calls, certain scenarios such as agent-to-agent calls can cause wrap-up operation to fail. However, if this exception scenario does not affect your deployment and you have specific requirements, use this property to enable the **Wrap-Up** button for all call types.

When you use the CLI command **utils finesse set\_property desktop enableWrapupButtonForAllCallTypes false** to disable the Wrap-Up button, the button will still be available for the following call types:

- Outbound
- Outbound Callback
- Out

### Notification Connection Type

Use the following CLI commands to update the desktop notification connection type as WebSockets or BOSH:

- For WebSockets: **utils finesse set\_property desktop notificationConnectionType websocket**
- For BOSH: **utils finesse set\_property desktop notificationConnectionType bosh**

By default, the connection type is WebSockets.

### Desktop Chat Attachment

Use the following CLI commands to enable or disable the attachment support in Desktop Chat:

- To enable: **utils finesse set\_property desktop desktopChatAttachmentEnabled true**
- To disable: **utils finesse set\_property desktop desktopChatAttachmentEnabled false**

By default, attachments are enabled in the Desktop Chat.

### Desktop Chat Maximum Attachment Size

Use the following CLI commands to configure the attachment size in Desktop Chat:

- **utils finesse set\_property desktop desktopChatMaxAttachmentSize *Attachment Size***
- For example, to set the maximum attachment size to 2 MB, use:
- utils finesse set\_property desktop desktopChatMaxAttachmentSize 2097152**




---

**Note** The maximum attachment size configurable is up to 10 MB.

---

If you don't configure the maximum attachment size, by default, the maximum attachment size is set to 5 MB.

### Desktop Chat Unsupported File Types

The .exe, .msi, .sh, and .bat file types are not supported by default. Use the following CLI commands to override the default list and customize the file types that won't be supported in the Desktop Chat:

- **utils finesse set\_property desktop desktopChatUnsupportedFileTypes *File Types***
- For example, to set the .jar and .bin as unsupported file types, use:
- utils finesse set\_property desktop desktopChatUnsupportedFileTypes jar,bin**

Multiple file types can be added using a comma-separated string.

### Automatic Desktop Login Retries

Cisco Finesse supports automatic desktop login retries when the desktop login fails due to device-related errors. The following properties allow the administrator to control how this feature behaves:

- To enable: **utils finesse set\_property desktop enableRetryLoginFeature true**




---

**Attention** The **utils finesse set\_property desktop enableRetryLoginFeature true** command is not enabling automatic desktop login retries. So, to enable automatic desktop login retries, use the following command:

**utils finesse set\_property desktop retryLoginAfterLogoutPhoneFailure true**

To view the status of automatic desktop login retries, use the following command:

**utils finesse show\_property desktop retryLoginAfterLogoutPhoneFailure**

To disable the automatic desktop login retries, use the following command:

**utils finesse set\_property desktop enableRetryLoginFeature false**

---

- If this feature is enabled, you can define the retry attempts and intervals.
  - To set the number of retry attempts: **utils finesse set\_property desktop loginFailureRetryAttempts <value>**  
The maximum retry attempts are 10. Default value is 3.
  - To set intervals: **utils finesse set\_property desktop loginFailureRetryInterval <value>**  
The login retry has a configurable amount of delay between each retry to allow the device to recover. The minimum and maximum interval between retries is 15-180 seconds. Default value is 60 seconds.




---

**Note** Reducing the retry interval increases the load on the system when there is a system-wide outage of devices.

---

By default, the value of this property is set to true.

### Sign in as a Mobile Agent

Use the following CLI commands to enable or disable the Sign in as a Mobile Agent feature on the Cisco Finesse sign in page:

- To enable: **utils finesse set\_property desktop enableMobileAgentLogin true**
- To disable: **utils finesse set\_property desktop enableMobileAgentLogin false**

By default, the value of this property is set to true.

### Enable or Disable Keyboard Shortcuts

Use the following CLI commands to enable or disable the keyboard shortcuts for the Cisco Finesse agent and supervisor desktop:

- To enable: **utils finesse set\_property desktop enableShortCutKeys true**
- To disable: **utils finesse set\_property desktop enableShortCutKeys false**

By default, the value of this property is set to true.

### Enable or Disable Drag-and-Drop and Resize for a Gadget or Component

Use the following CLI commands to enable or disable the drag-and-drop and resize features for a gadget or component in the Cisco Finesse desktop:

- To enable: **utils finesse set\_property desktop enableDragDropAndResizeGadget true**
- To disable: **utils finesse set\_property desktop enableDragDropAndResizeGadget false**

By default, the value of this property is set to false. For more information on using the drag-and-drop and resize features, see the *Cisco Finesse Agent and Supervisor Desktop User Guide* at <https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-user-guide-list.html>.

### Enable or Disable Preloading of the Secondary Resources

Use the following CLI commands to enable or disable the preloading of the secondary server resources from the alternate side during desktop sign in:

- To enable: **utils finesse set\_property desktop preLoadSecondaryResources true**
- To disable: **utils finesse set\_property desktop preLoadSecondaryResources false**

The preloaded resources are **images**, **CSS**, **JS**, and **HTML**. The preloading reduces latency and improves performance during desktop failover. By default, the value of this property is set to true.

### Security Banner Message for Desktop Users

Cisco Finesse supports custom banner messages in the desktop Sign In page. The administrator defines the banner message for Cisco Finesse desktop users so that they are aware of the security policy while using Cisco Finesse. The banner message can have a maximum of 220 characters. It supports both alphanumeric and special characters. By default, the banner message is not displayed.

- To add the security banner message to the desktop Sign In page: **utils finesse set\_property desktop desktopSecurityBannerMessage <value>**

The following example displays the sample security banner that is defined for desktop Sign In page.

```
admin:utils finesse set_property desktop desktopSecurityBannerMessage "IMPORTANT: Finesse
may only be accessed by authorized users!"
```

```
Property successfully updated.
Ensure property is updated in all Finesse nodes in the cluster.
```

```
No service restart required. Ensure browser is refreshed for the changes to take effect.
```

- To remove the security banner message in the desktop Sign In page: **utils finesse set\_property desktop desktopSecurityBannerMessage ""**



---

**Note** Cisco Finesse Administration Console and Cisco Finesse desktop now support messages configured in Cisco Unified OS Administration by using the custom logon message feature. From Unified CCX Release 12.5(1)SU1, it's recommended that you use the custom logon message feature as an alternative to the security banner message feature to convey important information to Cisco Finesse desktop users and administrators. For more information about setting up custom logon message, see the *Set Up Customized Logon Message* section in the *Cisco Unified Operating System Administration Guide for Cisco Unified CCX and Cisco Unified IP IVR* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-express/products-maintenance-guides-list.html>.

---

### Enable High Contrast Look and Feel

By default, Cisco Finesse Desktop uses high contrast colors in icons, buttons, text elements, and so on to improve the visibility of desktop elements.

Use the following CLI commands to enable or disable the high contrast colors in Desktop:

- To enable: **utils finesse set\_property desktop enhanceContrast true**
- To disable: **utils finesse set\_property desktop enhanceContrast false**

### WORK Mode Retention for Non-Voice

Use the following CLI commands to enable or disable the user to remain in WORK mode after CTI reconnection:

- To enable agent to retain WORK mode after CTI reconnection (non-voice): **utils finesse set\_property desktop enableAutoWorkModeStateChange false**
- To make the agent available automatically after CTI reconnection (non-voice): **utils finesse set\_property desktop enableAutoWorkModeStateChange true**

By default, the value of this property is set to *true* (disabled).

The administrator can enable this CLI, to allow the agents to change to an available state in non-voice MRD explicitly after the Cisco Finesse desktop and media channels are initialized (contrary to the previous behavior where, agents moving automatically to available state, and causing RONA, because of the delay in re-initialization of the gadgets which handle non-voice media).

### Dual-Tone Multi-Frequency (DTMF) Desktop Behavior



---

**Note** To enable this CLI in Cisco Finesse, install Cisco Finesse 12.5(1)ES2 COP or higher.

---

The **Wrap-Up** button and the call control buttons, **Hold**, **Transfer**, **Consult**, and **End** are disabled across all calls when DTMF **Keypad** is opened, and until the responses to all DTMF requests are completed or have timed out.

### DTMF Pending Requests Threshold Count

When the network or the server is slow to respond, then the response to DTMF requests are delayed. DTMF keypad prevents new operations when more than a configured number of outstanding responses are pending. The default value is 20.

- To configure the DTMF threshold count for pending requests: **utils finesse set\_property desktop pendingDTMFThresholdCount <value>**

The following example displays the sample DTMF threshold count.

```
admin:utils finesse set_property desktop pendingDTMFThresholdCount 15
```

```
Property successfully updated.
Ensure property is updated in all Finesse nodes in the cluster.
```

```
No service restart required. Ensure the desktop browser is refreshed for the changes
to take effect.
```

### DTMF Request Timeout

Cisco Finesse waits for a configured time for each DTMF request. The default timeout is 5 seconds.

- To configure the DTMF timeout for pending requests: **utils finesse set\_property desktop dtmfRequestTimeoutInMs <value>**




---

**Note** The timeout value must be entered in milliseconds.

---

The following example displays the sample DTMF timeout count.

```
admin:utils finesse set_property desktop dtmfRequestTimeoutInMs 4000
```

```
Property successfully updated.
Ensure property is updated in all Finesse nodes in the cluster.
```

```
No service restart required. Ensure the desktop browser is refreshed for the changes
to take effect.
```

### Enable Automatic Device Selection for Single Device

Use the following CLI commands to enable or disable display of **Select Your Preferred Device** window, during agent sign-in, when device selection is enabled for the agent, and the extension is associated with only one active device at the time of sign-in:

- To enable: **utils finesse set\_property desktop enableDeviceSelectionForSingleDevice true**
- To disable: **utils finesse set\_property desktop enableDeviceSelectionForSingleDevice false**




---

**Note** This property is applicable only when there's a single active device available for the extension at the time of sign-in. However, the device selection window comes up when there's no active device, and when there are multiple active devices for the extension.

---

Cisco Finesse implicitly selects one device as agent's active device when the property is set to **False**. If the value is set to **True**, then the device selection window appears. By default, the value is set to **True** for Unified CCE deployments.

### Maximum Number of Visible Multi-Tab Gadget Tabs

The **utils finesse set\_property desktop maxTabsOnTabbedGadgetHeader** CLI command is used to configure the maximum number of gadgets that appear on the Multi-Tab gadget header. Administrators can customize the **maxTabsOnTabbedGadgetHeader** property to display more gadget tabs in the Multi-Tab gadget header before they are moved to the gadget drop-down. There are no restrictions on the maximum number of gadget tabs that appear on the Multi-Tab gadget header. The default value is set to seven.

Use the following commands to view and set the maximum number of tabs that appear in the Multi-Tab gadget header:

- **utils finesse show\_property desktop maxTabsOnTabbedGadgetHeader**: This command displays the maximum number of tabs that are configured to be displayed in the header of Multi-Tab gadgets. The default value is seven.

#### Example

```
admin:utils finesse show_property desktop maxTabsOnTabbedGadgetHeader
The value of property 'maxTabsOnTabbedGadgetHeader' is '7'
```

- **utils finesse set\_property desktop maxTabsOnTabbedGadgetHeader <value>**: This command sets the maximum number of tabs that are displayed in the header of the Multi-Tab gadgets.

#### Example

```
admin:utils finesse set_property desktop maxTabsOnTabbedGadgetHeader 5
Property successfully updated.
Ensure property is updated in all Finesse nodes in the cluster.
No service restart required.
Ensure the desktop browser is refreshed for the changes to take effect.
```




---

**Note** This command is used to set the maximum number of gadgets that can be displayed in the header. If you have configured more gadgets for a Multi-tab gadget, the remaining gadgets are listed in the drop-down.

---

### Non-Page Level Gadgets Preceding Page-Level Gadgets in a Multi-Tab Gadget

A Multi-Tab Gadget can consist of both page-level and non page-level gadgets. In this case, page-level gadgets appear first in sequence in the Multi-Tab header by default. However, you can use the following CLI commands to allow non page-level gadgets to appear before the page-level gadgets in a Multi-tab gadget header:

- **utils finesse show\_property desktop displayNavLevelGadgetsFirstInMultitab**

The **utils finesse show\_property desktop displayNavLevelGadgetsFirstInMultitab** command allows you to check if non page-level gadgets are displayed first in the Multi-Tab gadget header. If the result of the command is **true**, the non page-level gadgets are displayed first in sequence. If the result of the command is **false**, the page-level gadgets are displayed first in sequence.

#### Example

```
admin:utils finesse show_property desktop displayNavLevelGadgetsFirstInMultitab
The value of property 'displayNavLevelGadgetsFirstInMultitab' is 'true'
```

- **utils finesse set\_property desktop displayNavLevelGadgetsFirstInMultitab <value>**

The **utils finesse set\_property desktop displayNavLevelGadgetsFirstInMultitab** command configures the order of non page-level gadgets and page-level gadgets in a Multi-Tab gadget header. If the command

is set to **true**, the non page-level gadgets are displayed first in sequence. If the command is set to **false**, the page-level gadgets are displayed first in sequence.

### Example

```
admin:utils finesse set_property desktop displayNavLevelGadgetsFirstInMultitab true
Property successfully updated.
Ensure property is updated in all Finesse nodes in the cluster.
No service restart required. Ensure the desktop browser is refreshed for the changes
to take effect.
```

## Notifications from Call Control in Multi-Tab Gadgets

Use the following CLI commands to configure the notification settings for Call Control gadget tab when it is configured as a tab within a Multi-Tab gadget:

- **utils finesse set\_property desktop showNotificationOnCallVariablesChange true**

The **utils finesse set\_property desktop showNotificationOnCallVariablesChange** command changes the notification settings when the call variables are updated.

### Example

```
admin:utils finesse set_property desktop showNotificationOnCallVariablesChange true
Property successfully updated.
Ensure property is updated in all Finesse nodes in the cluster.
No service restart required. Ensure the desktop browser is refreshed for the changes
to take effect.
```

- **utils finesse set\_property desktop showNotificationOnParticipantChange true**

The **utils finesse set\_property desktop showNotificationOnParticipantChange** command changes the notification settings when participants in the call change.

### Example

```
admin:utils finesse set_property desktop showNotificationOnParticipantChange true
Property successfully updated.
Ensure property is updated in all Finesse nodes in the cluster.
No service restart required. Ensure the desktop browser is refreshed for the changes
to take effect.
```

- **utils finesse set\_property desktop showNotificationOnWrapUpReasonsChange true**

The **utils finesse set\_property desktop showNotificationOnWrapUpReasonsChange** command changes the notification settings whenever the wrap-up reasons for the call change. This command is applicable to a call with multiple participants, and the notification appears for all agents excluding the agent who changed the wrapUpReason. So, the notification does not appear for a call with a single agent.

### Example

```
admin:utils finesse set_property desktop showNotificationOnWrapUpReasonsChange true
Property successfully updated.
Ensure property is updated in all Finesse nodes in the cluster.
No service restart required. Ensure the desktop browser is refreshed for the changes
to take effect.
```

- **utils finesse set\_property desktop showNotificationOnNumCallsChange true**

The **utils finesse set\_property desktop showNotificationOnNumCallsChange** command changes the notification settings when the number of calls change for an agent.

### Example



```
admin:utils finesse set_property desktop showNotificationOnNumCallsChange true
Property successfully updated.
Ensure property is updated in all Finesse nodes in the cluster.
No service restart required. Ensure the desktop browser is refreshed for the changes
to take effect.
```

## WebProxy Service

WebProxy Service acts as a transparent reverse proxy between external clients and the Finesse service. It provides SSL termination and caching services to the Finesse server to reduce latency and improve performance.

Configuration changes done on the Finesse server may not be immediately available to the clients due to the intermediary webproxy cache. The administrator can clear the intermediary webproxy cache using **utils webproxy cache clear**.

WebProxy cache is automatically cleared when you restart the Finesse Tomcat service. Static resources (images and scripts), Shindig gadget XML, and resources are cached until the Finesse Tomcat service is restarted or explicitly cleared by the administrator.

For more information on REST API Response Caching, see *REST API Developer Guide* at <https://developer.cisco.com/docs/finesse/>.

The logging level of the WebProxy Service is managed using the web proxy log-levels CLI.




---

**Note** WebProxy Service CLIs are node-specific and must be run on all nodes in the cluster.

---

Proxy cache bypassing reduces performance and must be used for debugging purposes during the gadget development or troubleshooting.

Server cache for the Finesse API can be bypassed by including `bypassServerCache=true` as a query parameter in the request or clear server cache using **utils webproxy cache clear**.

Server cache for the Finesse desktop can be bypassed by including `bypassServerCache=true&nocache` as a query parameter in the desktop URL.

### utils webproxy cache clear

This command clears the cache from the WebProxy Service.

#### Command Syntax

```
utils webproxy cache clear {all | webproxy | desktop | rest | shindig}
```

#### Options

- *all*—Clears all the configured caches.
- *webproxy*—Clears the default webproxy cache.
- *desktop*—Clears the desktop cache. The desktop cache contains static HTML, CSS, scripts, and icons used in the Finesse desktop.
- *rest*—Clears the REST APIs cache. The REST API responses cached are:
  - ECCVariableConfig

- MediaDomain
- TeamResource APIs include ReasonCodes, WrapUpReasons, MediaPropertiesLayouts, PhoneBooks, and WorkFlows. The responses of the TeamResource API are cached at the team-level.
- *shindig*—Clears the Shindig cache. The Shindig cache contains XML gadget definition (if request-response) and gadget resources (concat request-response).
- *authmode*—Clears the UserAuthMode API cache.

### Command Modes

Administrator (admin)

### Requirements

Command privilege level: 1

Allowed during upgrade: Yes

Applies to: Unified CCE, Unified CCX, and Packaged CCE

### Example

```
admin:utils webproxy cache clear desktop
Successfully cleared desktop cache
```

## set webproxy access-log-level

This command sets the log-level for the access logs generated by the WebProxy Service. The access logs record information about all external requests that reach the proxy. The requests are logged in the access log after the request is processed.

### Command Syntax

```
set webproxy access-log-level {off|info|debug}
```

### Options

- *off*—Turns off the logging into the access logs of the WebProxy Service.
- *info*—Sets the log-level for access logs of the WebProxy Service to information. This captures the data of each request such as time, client, host, user, and so on.
- *debug*—Sets the log-level for access logs of the WebProxy Service to debug. This captures the detailed data of each request for debugging.




---

**Note** Setting the access logs to debug impacts performance. Hence, avoid using in the production deployment.

---

### Command Default

The default value is *off*.

### Command Modes

Administrator (admin)

### Requirements

Command privilege level: 1

Allowed during upgrade: Yes

Applies to: Unified CCE, Unified CCX, and Packaged CCE

### Example

```
admin:set webproxy access-log-level off
Webproxy access log-level is turned off

admin:set webproxy access-log-level info
Successfully set webproxy access log-level to info
Service restarted
```

## set webproxy log-severity

This command sets the severity of the error logs that are generated by the WebProxy Service. The error logs record information about encountered issues of different severity levels.

### Command Syntax

**set webproxy log-severity** {*debug* | *warn* | *error* | *crit* | *alert* | *emerg*}

### Options

- *debug*—Sets the severity level to debug.



---

**Note** Setting the error logs to debug impacts performance. Hence, avoid using in the production deployment.

---

- *warn*—Sets the severity level to warning.
- *error*—Sets the severity level to error.
- *crit*—Sets the severity level to critical.
- *alert*—Sets the severity level to alert.
- *emerg*—Sets the severity level to emergency.

### Command Default

The default value is *warn*.

### Command Modes

Administrator (admin)

### Requirements

Command privilege level: 1

Allowed during upgrade: Yes

Applies to: Unified CCE, Unified CCX, and Packaged CCE

### Example

**show webproxy access-log-level**

```
admin:set webproxy log-severity warn
Successfully set webproxy log severity to warn
Service restarted
```

**show webproxy access-log-level**

This command displays the configured log-level for the access logs of the WebProxy Service.

**Command Syntax**

```
show webproxy access-log-level
```

**Command Modes**

Administrator (admin)

**Requirements**

Command privilege level: 1

Allowed during upgrade: Yes

Applies to: Unified CCE, Unified CCX, and Packaged CCE

**Example**

```
admin:show webproxy access-log-level
Current webproxy access log-level is: info
```

**show webproxy log-severity**

This command displays the configured severity level for the error logs of the WebProxy Service.

```
show webproxy log-severity
```

**Command Modes**

Administrator (admin)

**Requirements:**

Command privilege level: 1

Allowed during upgrade: Yes

Applies to: Unified CCE, Unified CCX, and Packaged CCE

**Example**

```
admin:show webproxy log-severity
Current webproxy log-severity is: warn
```

# Service Properties

Configure the service properties using the following CLIs for the features.




---

**Note** The CLIs require Cisco Finesse Tomcat restart except for desktop related properties.

---

### Security Banner Message for Administrators

Cisco Finesse supports custom banner messages in the administration Sign In page. The administrator defines the banner message for the users so that they are aware of the security policy while using Cisco Finesse. The banner message can have a maximum of 220 characters. It supports both alphanumeric and special characters. By default, the banner message is not displayed.

- To add the security banner message to the administrator Sign In page: **utils finesse set\_property admin adminSecurityBannerMessage <value>**

The following example displays the sample security banner that is defined for the administrator Sign In page.

```
admin:utils finesse set_property admin adminSecurityBannerMessage "IMPORTANT: Finesse may
only be accessed by authorized users!"
```

```
Property successfully updated.
Ensure property is updated in all Finesse nodes in the cluster.
```

```
Restart Cisco Finesse Tomcat Service for the changes to take effect:
utils service restart Cisco Finesse Tomcat
```

- To remove the security banner message in the administrator Sign In page: **utils finesse set\_property admin adminSecurityBannerMessage ""**




---

**Note** Cisco Finesse Administration Console and Finesse desktop now support messages configured in Cisco Unified OS Administration by using the custom logon message feature. For more information about setting up a custom logon message, see the *Set Up Customized Logon Message* section in the *Cisco Unified Operating System Administration Guide for Cisco Unified CCX and Cisco Unified IP IVR* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-express/products-maintenance-guides-list.html>.

---

### Enable or Disable User Authentication Discovery API

Use the following CLI commands to enable or disable the `UserAuthMode` API. This API allows a client to discover the authentication mode of a user in Unified CCE deployments when the system is in hybrid mode (SSO or non-SSO). By default, this API is enabled.




---

**Note** This API does not require HTTP BASIC authentication. It is provided for third-party integration to decide if the user authentication must proceed with SSO or non-SSO authentication modes.

---

- To enable: **utils finesse set\_property webservices enableUserAuthMode true**
- To disable: **utils finesse set\_property webservices enableUserAuthMode false**

### Enable or Disable Plain XMPP Socket—Port 5222

Use the following CLI commands to enable or disable the Cisco Finesse Notification Service plain XMPP port (5222). This port can be enabled only if you have third-party solutions that connect directly to the Cisco Finesse Notification Service over plain Transmission Control Protocol (TCP) connection. This port is not required for the Finesse desktop or BOSH/WebSocket based integrations. By default, the port is disabled.

- To enable: **utils finesse set\_property webservices enableInsecureOpenfirePort true**
- To disable: **utils finesse set\_property webservices enableInsecureOpenfirePort false**

### Enable or Disable Secure XMPP Socket—Port 5223

Use the following CLI commands to enable or disable the external access to the Cisco Finesse Notification Service TCP-based XMPP port (5223). The port must be enabled for external client connectivity only if you have third-party solutions that connect directly to the Cisco Finesse Notification Service over this port. By default, the port is enabled (value is set to *true*).

When the port is enabled, it can be accessed by the Cisco Finesse nodes (primary and secondary) and by external clients. When the port is disabled, it cannot be accessed by external clients.

- To enable: **utils finesse set\_property webservices enableExternalNotificationPortAccess true**
- To disable: **utils finesse set\_property webservices enableExternalNotificationPortAccess false**




---

**Note** Restart Cisco Finesse Tomcat and Cisco Finesse Notification Services for the changes to take effect.

---

### Restricting Access to the External XMPP Notification Port 5223

Use the following CLI commands to restrict the IP addresses from accessing the TCP-based XMPP notification port (5223) available for external client connectivity. You can add, delete, or view the configured IP addresses only when the **enableExternalNotificationPortAccess** property is enabled on all the Finesse nodes in the cluster.




---

**Note** These restrictions do not affect the desktop XMPP notification port 7443.

---

To enable access to port 5223 use the CLI command **utils finesse set\_property webservices enableExternalNotificationPortAccess true**.

- **utils finesse notification external\_port\_access add [ip1,ip2,ip3]**—This command adds one or more IP addresses to the list of hosts that are configured to access Cisco Finesse XMPP notification port 5223. Multiple IP addresses can be provided as a comma-separated list. Wildcard character \* is not supported.

#### Example

```
admin:utils finesse notification external_port_access add 10.10.10.21,10.10.255.25
```

```
Successfully added 2 IP address(es). Ensure that the IP address(es) are added,
and verify that external notification port access is enabled in all the Finesse nodes
in the cluster.
```

```
Please refer to 'utils finesse show_property webservices
enableExternalNotificationPortAccess'.
```

```
Restart Cisco Finesse Notification Service for the changes to take effect:
utils service restart Cisco Finesse Notification Service
```

- **utils finesse notification external\_port\_access delete**—This command deletes one or more IP addresses from the list of hosts that are configured to access Cisco Finesse XMPP notification port 5223. Multiple IP addresses can be provided as a comma-separated list.

#### Example

```
admin:utils finesse notification external_port_access delete 10.10.10.21,10.10.255.25
```

```
Successfully deleted 2 IP address(es). Verify that the IP address(es) are deleted in
all the Finesse nodes in the cluster.
```

```
Restart Cisco Finesse Notification Service for the changes to take effect:
utils service restart Cisco Finesse Notification Service
```

- **utils finesse notification external\_port\_access delete\_all**—This command deletes all the configured IP addresses allowed to access the Cisco Finesse XMPP notification port 5223.

#### Example

```
admin:utils finesse notification external_port_access delete_all
```

```
Do you want to delete all IP address(es) (y/n): y
```

```
Successfully deleted all IP address(es). Verify that the IP address(es) are deleted in
all the Finesse nodes in the cluster.
```

```
Restart Cisco Finesse Notification Service for the changes to take effect:
utils service restart Cisco Finesse Notification Service
```

- **utils finesse notification external\_port\_access list**—This command lists all the configured IP addresses allowed to access the Cisco Finesse XMPP notification port 5223.

#### Example

```
admin:utils finesse notification external_port_access list
```

```
The following IP address(es) are configured to access the notification port:
10.10.10.21
10.10.255.25
```

```
External notification port access is disabled in the present node. Verify that is enabled
in all the Finesse nodes in the cluster.
```

```
Please refer to 'utils finesse show_property webservices
enableExternalNotificationPortAccess'.
```

### Enable or Disable Enforcement of X.509 Certificate Trust Validation

Use the following CLI commands to enable or disable the validation of the X.509 CA or the selfsigned certificate. From Release 12.5(1) onwards, Cisco Finesse validates SSL certificates of all the servers (CUCM and SocialMiner) it communicates. This requires the custom CA providers or the selfsigned certificates of the server it communicates to be present in the Cisco Finesse Tomcat trust store. If the certificates are not added into the Cisco Finesse trust store, then certain interactions can fail. It is advised to add the certificates into the Cisco Finesse trust store. If any user chooses to ignore the validation, enforcement can be turned off. This CLI allows users to disable or enable validation. By default, the validation is turned on.

- To enable: **utils finesse set\_property webservices trustAllCertificates true**
- To disable: **utils finesse set\_property webservices trustAllCertificates false**

### Enable or Disable Call Variables Logging

Use the following CLI commands to enable or disable the call variables logging. The callVariables contain sensitive user information and this property allows the administrator to decide whether the information must be captured in the logs. By default the property is disabled.

- To enable:

```
utils finesse set_property webservices logCallVariables true
```

```
utils finesse set_property fippa logCallVariables true
```

- To disable:

```
utils finesse set_property webservices logCallVariables false
```

```
utils finesse set_property fippa logCallVariables false
```

### Permissions to Drop Participants from Conference




---

**Note** To enable this CLI in Cisco Finesse, install Finesse 12.5(1) ES3 COP or higher.

---

Use the following commands to allow an agent or a supervisor, who is the participant in a conference call, to drop another agent, supervisor, or caller (participants) from the conference call.




---

**Note** Only agents and supervisors can drop participants in the conference call.

---

- **utils finesse set\_property webservices enableDropParticipantFor supervisor\_only**—This command allows only the supervisor, who is a participant of the conference call, to drop other agents in the conference call. The supervisor cannot drop a CTI Route Point, IVR port, a device to which no agent is signed in, or a caller device. By default, this property is set to **supervisor\_only**.
- **utils finesse set\_property webservices enableDropParticipantFor conference\_controller\_and\_supervisor**—This command allows,
  - the supervisor to drop any agents, CTI Route Point, IVR port, a device to which no agent is signed in, or a caller device in the conference call.
  - the conference controller (an agent who initiated the conference call) to drop another agent, supervisor, CTI Route Point, IVR port, a device to which no agent is signed in, or a caller device in the conference call.





**Note** To enable the supervisor or call controller to drop an unmonitored extension in Cisco Unified CCE, in Release 12.0(1) or higher, set the **DropAnyPartyEnabled** registry key to *1* in the Dynamic Registry of the CTI server. The supervisor cannot drop a CTI Route Point, IVR port, a device to which no agent is signed in, a caller device, or other agents for whom SILENT\_MONITOR is not initiated by the supervisor.

For more information, see the *Enable Dropping Call Participants from a Conference Call* section in *Cisco Contact Center Gateway Deployment Guide for Cisco Unified ICM/CCE* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-programming-reference-guides-list.html>.

- **utils finesse set\_property webservices enableDropParticipantFor all**—This command allows any agent or supervisor in the conference call to drop another agent, supervisor or the caller. To ensure that this feature works properly on Finesse desktop, you must update the **enableDropParticipantFor** value for desktop also. For more information on enabling the desktop property, refer to [Drop Participants from Conference, on page 70](#).

### Enable Team API Access for All Users

A new configuration property `enableTeamAPIAccessForAllusers` is added for enabling the Team API access for all agents and supervisors. When you enable this property, all agents and supervisors will be able to access information of all the teams without any restriction.

If this property is disabled, only administrator and supervisors can access the Team API. Supervisors can access the information of the teams that they are assigned to and Administrators can access all the teams. By default this property is disabled.

Use the following command to enable this configuration property:

**utils finesse set\_property webservices enableTeamAPIAccessForAllusers true.**

#### Example:

```
admin:utils finesse set_property webservices enableTeamAPIAccessForAllusers true
```

```
Property successfully updated.
Ensure property is updated in all Finesse nodes in the cluster.
```

```
Restart Cisco Finesse Tomcat Service for the changes to take effect:
utils service restart Cisco Finesse Tomcat
```

Use the following command to disable this configuration property:

**utils finesse set\_property webservices enableTeamAPIAccessForAllusers false**

#### Example:

```
admin:utils finesse set_property webservices enableTeamAPIAccessForAllusers false
```

```
Property successfully updated.
Ensure property is updated in all Finesse nodes in the cluster.
```

```
Restart Cisco Finesse Tomcat Service for the changes to take effect:
utils service restart Cisco Finesse Tomcat
```

# Log Collection Schedule

Use the following CLIs to create, list, and delete automatic desktop log collection schedules for agents and supervisors. This can also be used for debugging purposes.

**utils finesse desktop\_auto\_log\_collection create:** This command creates a schedule that collects the agent's browser logs. You can create up to five log collection schedules for up to 15 agents.

While creating the log schedule, specify the agent IDs, log collection interval, and duration up to when the logs are to be collected.

The log collection interval and the duration have to be between 30 to 900 seconds. The logs that are collected during the schedule are received in a .zip file format. The logs are collected at:  
/opt/cisco/desktop/logs/clientlogs.

## Example:

```
admin:utils finesse desktop_auto_log_collection create

Initializing command line interface...
Checking Cisco Finesse Tomcat status...

Enter agent IDs to continue. (Maximum 15 agents) [Example : 1001001,1001002] : 1001002
Agent IDs entered: 1001002
Enter duration in seconds.(value between 30 and 900) : 240
Duration entered: 240
Enter interval in seconds.(value between 30 and 240) : 60
Interval entered: 60

Successfully scheduled client log collection for the specified agent(s).

Ensure the same is enabled in all the Finesse nodes in the cluster..
```

**utils finesse desktop\_auto\_log\_collection list:** This command lists all active log collection schedules.

## Example:

```
admin:utils finesse desktop_auto_log_collection list

Initializing command line interface...
Checking Cisco Finesse Tomcat status...
These are the live log collection schedules:

Schedule ID:1 Created At: Thu Jun 6 23:23:53 PDT 2019
Duration: 240 seconds
Frequency: 60 seconds
Agent Ids: 1001002
```

**utils finesse desktop\_auto\_log\_collection delete:** This command deletes the active log collection schedules. When this command is run, all the active log collection schedules are displayed and you are prompted to enter the Schedule ID that you want to delete.

## Example:

```
admin:utils finesse desktop_auto_log_collection delete

Initializing command line interface...
Checking Cisco Finesse Tomcat status...
These are the live log collection schedules:

Schedule ID:1 Created At: Thu Jun 6 23:23:53 PDT 2019
Duration: 240 seconds
```

```

Frequency: 60 seconds
Agent Ids: 1001002
Enter schedule ID to delete (enter 'all' to delete all): 1
Schedule ID entered: 1

Successfully deleted the log collection with schedule id : 1

```

## Upgrade

Upgrade-related commands are grouped under **utils system upgrade**.

**utils system upgrade initiate:** This command allows you to initiate and install upgrades and Cisco Option Package (COP) files from both local and remote directories.

**utils system upgrade cancel:** This command allows you to cancel an upgrade.

**utils finesse layout updateCuicGadgetUrl:** The command has the following options:

- **utils finesse layout updateCuicGadgetUrl 12.5.1**—This command allows you to change the .jsp references of Cisco Unified Intelligence Center (CUIC) gadgets to .xml with no functional changes in the Finesse desktop layout. This command updates the CUIC gadgets URL path to work with CUIC, Release 12.5(1). The changes are applicable for both Finesse default desktop layout and team desktop layouts.




---

### Note

- The administrator must run the CLI on the primary Finesse server.
  - The changes are not reversible. The changes are not compatible with CUIC releases prior to 12.5(1).
- 

```

admin:utils finesse layout updateCuicGadgetUrl 12.5.1
This command will update the CUIC URI configured in the desktop layouts to work with
CUIC version 12.5.1.The changes are applied to both default and team layouts.
WARNING - The changes are not reversible & are not compatible with older CUIC
versions.Please ensure that you have the matching CUIC version.
Do you wish to continue [y / n] ?yes

All CUIC URIs have been modified in layouts.

Restart Cisco Finesse Tomcat for the changes to take effect:
utils service restart Cisco Finesse Tomcat

```

- **utils finesse layout updateCuicGadgetUrl 12.6.1+**—This command allows you to change the .jsp references of CUIC to .xml with no functional changes in the Finesse desktop layout. This command updates the CUIC gadgets URL path to work with CUIC, Release 12.6(1) and later versions. The changes are applicable for both Finesse default desktop layout and team desktop layouts.




---

### Note

- The administrator must run the CLI on the primary Finesse server.
  - The changes are not reversible. The changes are not compatible with CUIC releases prior to 12.5(1).
-

```

admin:utils finesse layout updateCuicGadgetUrl 12.6.1+
This command will update the CUIC URI configured in the desktop layouts to work with
CUIC version 12.6 & later versions.The changes are applied to both default and team
layouts.
WARNING - The changes are not reversible & are not compatible with older CUIC
versions.Please ensure that you have the matching CUIC version.

Do you wish to continue [y / n] ?y

All CUIC URIs have been modified in layouts.

Restart Cisco Finesse Tomcat for the changes to take effect:
utils service restart Cisco Finesse Tomcat

```

Cisco Finesse Release 12.5(1) onwards, CUIC supports only XML gadgets. Switching to XML-based gadgets reduces latency and improves performance. After the installation of CUIC or Co-resident deployment, run this command to optimize the loading of CUIC gadgets.

For CUIC Release 12.5(1) gadgets (Live Data and Historical) to load in Cisco Finesse, the administrator must enable CORS on CUIC server using the **utils cuic cors enable** command.

For more information, see *Administration Console User Guide for Cisco Unified Intelligence Center* at <https://www.cisco.com/c/en/us/support/customer-collaboration/unified-intelligence-center/products-maintenance-guides-list.html>.

## Shutdown

Use the following command to shut down Finesse:

**utils system shutdown**

If the virtual hosts running the Finesse servers are also shut down during a maintenance event, to power up Finesse after the maintenance event is complete, you must sign in to the ESXi host or its vCenter with vSphere Client and power up the virtual machines for primary and secondary Finesse servers.

## Replication Status

To check replication status, run the following command on the *primary* Finesse server:

- **utils dbreplication runtimestate**

This command returns the replication status on primary and secondary Finesse servers.

- Check the RTMT counter value for replication. If all nodes in the cluster show a replication status of 2, replication is functioning correctly.
- If the RTMT counter value for replication status is 3 or 4 for all nodes in the cluster, replication is set up but an error occurred and replication is not functioning properly.
- If the majority of the nodes show a value of 0 or 1, run the command **utils dbreplication reset all** from the primary Finesse server.
- If any node shows any replication value other than 1 or 2, replication is not set up correctly.
- To fix replication, contact Cisco Technical Support.



---

**Note** The DB replication setup must be completed to reflect the following primary node changes to the secondary node.

- Reason Codes
  - Wrap-Up Reasons
  - Media Properties Layouts
  - Phone Books
  - Workflows
  - Team Message
- 

## View Property

Use the following CLIs to view the property values across all property files.

- **utils finesse show\_property fippa property\_name**: To view the specified Finesse IPPA property's value.
- **utils finesse show\_property desktop property\_name**: To view the specified desktop property's value.
- **utils finesse show\_property webservices property\_name**: To view the specified web service property's value.
- **utils finesse show\_property admin securityBannerMessage**: To view the specified banner message for the administrator Sign In page.



---

**Note** The View property CLIs do not support multiple values.

---

## Update Property

Use the following CLIs to update the property values across all property files.

- **utils finesse set\_property desktop property\_name property\_value**: To update an existing property value used by the Finesse desktop service.
- **utils finesse set\_property fippa property\_name property\_value**: To update an existing property value used by the Finesse IPPA service.
- **utils finesse set\_property webservices property\_name property\_value**: To update an existing property value used by the Finesse web service.
- **utils finesse set\_property admin adminSecurityBannerMessage**: To update an existing property value used by the Finesse administrator for the security banner message.

## Signout from Media Channels

The CLI **utils finesse user\_signout\_channel** is used by the Administrator to configure the media channels from which the users are signed out.

When signing out from Cisco Finesse, the CLI **utils finesse user\_signout\_channel type** lists all the choices of media channels from which the user is signed out. For example:

### **utils finesse user\_signout\_channel type**

1: signout user from voice channel.

2: signout user from voice and non-voice media channels configured for Cisco Finesse.

a: signout from all media channels configured for the user.




---

**Note** This is default behavior. It is suitable if the non-voice media is running as a gadget within Finesse Desktop and hence, it is valid to assume that the desktop user cannot handle tasks when signing out of Finesse.

---

q: quit.

Select the choice of media [1-2 or a,q]: 2

User signout channel type is now changed to "signout user from voice and non-voice media channels configured for Cisco Finesse."




---

**Note** **user\_signout\_channel type** must be updated for all Cisco Finesse nodes in the cluster.

For any changes done to media channels, it will take fifteen minutes for the new media channels signout to take effect.

---

The CLI **utils finesse user\_signout\_channel status** displays the type of media channels from which the user is signed out.

## Finesse Maintenance Mode Services

Use the following CLI commands to start and check the status of the Finesse Maintenance Mode on the Cisco Finesse server.

### **utils finesse maintenance initiate**

Initiates the maintenance mode on the Cisco Finesse server from the command line.

### **utils finesse maintenance status**

Checks the status of the maintenance mode of the Cisco Finesse server from the command line.

## ConnectedUsersInfo

Use the following CLI command to view the list of users connected to the Cisco Finesse server where the CLI is run.

### **utils finesse show\_connected\_users summary**

Provides the summary information about the connected users in the Cisco Finesse server where the CLI is run.

If the above command is run, it lists the total number of users connected to the Cisco Finesse server where the CLI is run along with the number of users connected through Cisco Finesse Desktop, Finesse IP Phone, and third-party desktops.

Example is as follows:

```
admin: utils finesse show_connected_users summary
Total Connected Users: 2
Desktop Users: 2
FIPPA Users: 0
Third-party Users: 0
Users connected to Finesse via LAN/WAN: 1
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.
```

## set webapp session maxlimit

This command sets the maximum limit for the number of concurrent Unified CCX web application sessions per user.

For the new setting to become effective, you must restart the node. Until you restart the node, the system continues to use the old values. In a HA setup, you must run this command on both the nodes. This command prompts you to restart the node.



**Note** Restart the nodes during off-peak traffic hours to avoid impact on the system performance.

This setting is preserved during software upgrades on both the nodes.

If the number of sessions is limited to 1 on both nodes, a user is allowed to have one session each on both the nodes.

### Command syntax

**set webapp session maxlimit *number***

### Syntax Description

Parameters	Description
<i>number</i>	Specifies the number to limit the concurrent web application sessions.  The value ranges from 1 to 10.

Parameters	Description
	<p>Default value is 10.</p> <p><b>Note</b> When you exceed the defined limit for maximum number of signed in sessions, the interface sign-in page displays the Logon Status message as: The Session limit has already been reached for &lt;username&gt;. Please logout from those sessions or wait &lt;Value&gt; minutes for inactive sessions to be automatically closed.</p>

### Command Modes

Administrator

### Requirements

Command privilege level: 1

Allowed during upgrade: No

Applies to: Cisco Identity Service Management, Disaster Recovery System, Cisco Unified CCX Administration, Cisco Finesse Administration, Cisco Unified Serviceability, Cisco Unified CCX Serviceability, Cisco Unified OS Administration, and Cisco Unified Intelligence Center.

### Example

```
admin:set webapp session maxlimit 4

*****W A R N I N G*****
C*****
The node needs to be restarted for the changes to take effect.This will
disconnect active web sessions and all web applications on this node will be
unavailable
until the node restarts.This node restart will take several minutes to complete.
Do you want to continue (yes/no) ? yes

*****Restarting node*****

The system is going down for reboot in 1 Minute
The webapp session limit has been successfully set to 4.
```

## set webapp session timeout

This command sets the time in minutes to invalidate any inactive Unified CCX web application sessions. After the set time elapses, the users are logged off from any of the inactive Unified CCX web sessions. The default session timeout value is 30 minutes.



For the new setting to become effective, you must restart the node. Until you restart the node, the system continues to use the old values. In a HA setup, you must run this command on both the nodes. This command prompts you to restart the node.



**Note** Restart the nodes during off-peak traffic hours to avoid impact on the system performance. This setting is preserved during software upgrades on both the nodes.

### Command syntax

**set webapp session timeout** *minutes*

### Syntax Description

Parameters	Description
<i>minutes</i>	Specifies the time, in minutes, that must elapse before a web application times out and logs off the user. <ul style="list-style-type: none"> <li>• Value range: 5-35000 minutes</li> <li>• Default value: 30 minutes</li> </ul>

### Command Modes

Administrator

### Requirements

Command privilege level: 1

Allowed during upgrade: No

Applies to: Cisco Identity Service Management, Disaster Recovery System, Cisco Unified CCX Administration, Cisco Finesse Administration, Cisco Unified Serviceability, Cisco Unified CCX Serviceability, and Cisco Unified OS Administration.

### Example

```
admin:set webapp session timeout 20
Continuing with this operation will set the session-timeout for web sessions to
20 minutes
after the node has been rebooted.
Continue (y/n)?y
web session-timeout updated to 20 minutes.

The node has to be rebooted for the changes to take effect immediately.
This will disconnect active web sessions.
Continue (y/n)?n
The updated web session time-out would take effect on next reboot

The current session-timeout used for web sessions and applications is 30 minutes.
The updated session-timeout value of 20 minutes will take effect on restart of
the node.
```

## Update Cloud Connect Connection Time

When there is a low bandwidth, the default value that is used for an HTTP client for obtaining the Cloud Connect token may not be sufficient and result in timeout. The following CLI commands can be used to view and update the connection timeout values. The default value for `cloudconnectHttpConnectionTimeout` is 5000 milliseconds and the default value for `cloudconnectHttpReadTimeout` is 10000 milliseconds.

To view the current values, use the following commands:

```
utils finesse show_property webservices cloudconnectHttpConnectionTimeout
```

```
utils finesse show_property webservices cloudconnectHttpReadTimeout
```

To update the values, use the following commands:

```
utils finesse set_property webservices cloudconnectHttpConnectionTimeout
```

```
<time_value_in_milliseconds>
```

```
utils finesse set_property webservices cloudconnectHttpReadTimeout
```

```
<time_value_in_milliseconds>
```

For example, the `utils finesse set_property webservices cloudconnectHttpConnectionTimeout 8000` command updates the HTTP connection timeout value to 8000 milliseconds.

The `utils finesse set_property webservices cloudconnectHttpReadTimeout 12000` command updates the HTTP connection read timeout value to 12000 milliseconds.

## AI Services Configuration

Finesse supports configuring AI services that are available through Agent Answers, Call Transcript, and Recording gadgets. By default, the AI services are disabled in Finesse. Use the following CLI commands to enable, disable, or view the status of the AI services. To enable or disable the AI services, the commands must be run in all the Finesse clusters.




---

**Note** Enable the AI services only if you are using Unified CCE 12.5(1). For Unified CCE 12.6(1) and above, this service is available without enabling.

---

Use the following CLI command to enable AI services:

```
utils finesse set_property webservices enableCustomAgentServices true
```

**Example:**

```
admin:utils finesse set_property webservices enableCustomAgentServices true
```

```
Property successfully updated.
```

```
Ensure property is updated in all Finesse nodes in the cluster.
```

```
Restart Cisco Finesse Tomcat Service for the changes to take effect:
```

```
utils service restart Cisco Finesse Tomcat
```

Use the following CLI command to disable AI services:

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

**Example:**

```
admin:utils finesse set_property webservices enableCustomAgentServices false
```

Property successfully updated.  
Ensure property is updated in all Finesse nodes in the cluster.

Restart Cisco Finesse Tomcat Service for the changes to take effect:  
utils service restart Cisco Finesse Tomcat

Use the following CLI command to view the status of the AI services:

```
utils finesse show_property webservices enableCustomAgentServices
```

**Example:**

```
admin:utils finesse show_property webservices enableCustomAgentServices
```

```
The value of property 'enableCustomAgentServices' is 'false'
```




---

**Note** If **enableCustomAgentServices** is enabled in Finesse, it will take the precedence over the configuration done in Cisco Unified CCE 12.6(1).

---

The configured AI services can also be managed through Finesse CLI commands. By default, AI services available through Agent Answers and Call Transcript gadgets are configured.

Use the following CLI command to modify configured AI services:

```
utils finesse set_property webservices customAgentService <values>
```

The parameter **values** allows comma separated numeric values ranging from 1 to 3. One or multiple values can be entered.

- 1—Agent Answers
- 2—Call Transcript
- 3—Recording

To enable only the Agent Answers gadget, use the following CLI:

```
utils finesse set_property webservices customAgentServices 1
```

**Example 1:**

```
admin:utils finesse set_property webservices customAgentServices 1
```

Property successfully updated.  
Ensure property is updated in all Finesse nodes in the cluster.

Restart Cisco Finesse Tomcat Service for the changes to take effect:  
utils service restart Cisco Finesse Tomcat

To enable all the AI gadgets, use the following CLI:

```
utils finesse set_property webservices customAgentServices 1,2,3
```

**Example 2:**

```
admin:utils finesse set_property webservices customAgentServices 1,2,3
```

Property successfully updated.

Ensure property is updated in all Finesse nodes in the cluster.

```
Restart Cisco Finesse Tomcat Service for the changes to take effect:
utils service restart Cisco Finesse Tomcat
```

## Digital Channels Configuration

With the integration of Digital Channels in Finesse, each nonvoice task is routed to Finesse through the Digital Routing API (DR-API) CloudConnect service. All the nonvoice tasks are associated with the DR-API CloudConnect host pair. Finesse uses this information to invoke the REST API request to close and transfer tasks. When such requests to DR-API hosts fail, Finesse uses the retry policy to reinvoke those requests. The policies used by Finesse can be managed through CLIs. To update the values, run the following commands on all the Finesse clusters:

- **utils finesse set\_property webservices mrdScriptSelectorPollingInterval <Time in seconds>**—To set the time interval in seconds to poll the updates for script selectors from Unified CCE. The default value is 900 seconds. To view the current value, use the **utils finesse show\_property webservices mrdScriptSelectorPollingInterval** command. For more information on script selector, see the [Manage Digital Channels gadget properties, on page 62](#) section.
- **utils finesse set\_property webservices drapiStatusPollingInterval <Time in seconds>**—To set the time interval in seconds to check the DR-API service status. When a REST API request to DR-API hosts fails, Finesse initiates a background polling mechanism to obtain the service status of DR-API hosts associated with nonvoice tasks. The background polling mechanism is terminated when DR-API hosts information is active. The time interval to obtain the service status can be managed by this CLI. The default value is 30 seconds. To view the current value, use the **utils finesse show\_property webservices drapiStatusPollingInterval** command.
- **utils finesse set\_property webservices drapiRequestRetryInterval <Time in seconds>**—To set the time interval in seconds to retry the DR-API REST request for nonvoice nonchat tasks. The default value is 60 seconds. To view the current value, use the **utils finesse show\_property webservices drapiRequestRetryInterval** command.
- **utils finesse set\_property webservices drapiMaxTimeToWaitBeforeRequestDiscard <Time in seconds>**—Time in seconds to discard the DR-API REST request if there's a failure of nonvoice nonchat tasks. It's the difference between the current time and the first DR-API REST request made for a particular task. The default value is 7200 seconds. To view the current value, use the **utils finesse show\_property webservices drapiMaxTimeToWaitBeforeRequestDiscard** command.
- **utils finesse set\_property webservices drapiRequestRetryIntervalForChat <Time in seconds>**—Time interval in seconds to retry the DR-API REST request for chat nonvoice conversations. The default value is 30 seconds. To view the current value, use the **utils finesse show\_property webservices drapiRequestRetryIntervalForChat** command.
- **utils finesse set\_property webservices drapiMaxTimeToWaitBeforeRequestDiscardForChat <Time in seconds>**—Time in seconds to discard the DR-API REST request if there's a failure for chat nonvoice conversations. It's the difference between the current time and the first DR-API REST request made for a particular task. The default value is 1800 seconds. To view the current value, use the **utils finesse show\_property webservices drapiMaxTimeToWaitBeforeRequestDiscardForChat** command.
- **utils finesse set\_property webservices drapiMinRandomDelayForRequestsInMs <Time in milliseconds>**— Minimum random delay in milliseconds before invoking the DR-API REST request.

The default value is 10 milliseconds. To view the current value, use the **utils finesse show\_property webservices drapiMinRandomDelayForRequestsInMs** command.

- **utils finesse set\_property webservices drapiMaxRandomDelayForRequestsInMs <Time in milliseconds>**— Maximum random delay in milliseconds before invoking the DR-API REST request. The default value is 1000 milliseconds. To view the current value, use the **utils finesse show\_property webservices drapiMaxRandomDelayForRequestsInMs** command.
- **utils finesse set\_property webservices drapiCongestionBackOffTimeInSec <Time in seconds>**— Backoff time in seconds to submit DR-API REST request if congestion or HTTP error due to too many requests has been detected in DR-API servers. The default value is 5 seconds. To view the current value, use the **utils finesse show\_property drapiCongestionBackOffTimeInSec** command.

An example of the `set_property` output is as follows:

```
utils finesse set_property webservices mrdScriptSelectorPollingInterval 900
Property successfully updated.
Ensure property is updated in all Finesse nodes in the cluster.
```

```
Restart Cisco Finesse Tomcat Service for the changes to take effect:
utils service restart Cisco Finesse Tomcat
```

The output is the same for all the Digital Channels Configuration commands.

## Certificate Configuration

### show tls server cert\_type

This command displays the configured certificate type used by the server for TLS connections.

#### Command syntax

```
show tls server cert_type
```

#### Requirements

Level privilege: 1

Command privilege level: 1

Allowed during upgrade: No

```
admin:show tls server cert_type
The server certificate type is set to ECDSA

Command successful
```

### set tls server cert\_type

Use this command to set the server certificate type to either RSA or ECDSA ciphers for TLS connections. The certificate set to the specified type is then presented for the cipher negotiations on all incoming TLS communications.

#### Command syntax

**set tls server cert\_type [option]****Option**

ecdsa—Sets the certificate type to ECDSA.

rsa—Sets the certificate type to RSA.

**Requirements**

Level privilege: 1

Command privilege level: 1

Allowed during upgrade: No

**Example**

```
admin:set tls server cert_type rsa

Configuring the server to use RSA certificates for all inbound connections.

Do you want to continue (y/n) ? y
Yes entered
Configuring the server to use RSA ciphers for inbound connections.

Successfully configured the server to use RSA certificate for all inbound
connections.

A system reboot will occur for the changes to take effect.
It is highly recommended that you perform a system backup
after the system reboot.
Ensure all the nodes in the cluster are running on the same
certificate type by running the 'set' command

Broadcast message from root@uccxfirstnode (Mon Jul 5 10:31:04 2021):

The system is going down for reboot in 1 Minute

Broadcast message from root@uccxfirstnode (Mon 2021-07-05 10:31:05 IST):

The system is going down for reboot at Mon 2021-07-05 10:32:04 IST!
```




---

**Note** After the system reboots, the self-signed and CA certificates of the servers, whose certificate type has changed, must be regenerated and re-uploaded into the client servers.

---



## APPENDIX **B**

# Certificates for Live Data

---

- [Certificates and Secure Communications](#), on page 239
- [Export Self-Signed Live Data Certificates](#), on page 239
- [Import Self-Signed Live Data Certificates](#), on page 240
- [Obtain and Upload Third-party CA Certificate](#), on page 241

## Certificates and Secure Communications

For secure Cisco Finesse, Cisco Unified Intelligence Center, AWDB, and Live Data server-to-server communication, perform any of the following:

- Use the self-signed certificates provided with Live Data.



---

**Note** When using self-signed certificates, agents must accept the Live Data certificates in the Finesse desktop when they sign in before they can use the Live Data gadget.

---

- Obtain and install a Certification Authority (CA) certificate from a third-party vendor.
- Produce a Certification Authority (CA) certificate internally.



---

**Note** After the successful upgrade, the CAs that are unapproved by Cisco are removed from the platform trust store. You can add them back, if necessary.

- For information about the list of CAs that Cisco supports, see the Cisco Trusted External Root Bundle at <https://www.cisco.com/security/pki>.
  - For information about adding a certificate, see [Insert a new tomcat-trust certificate](#).
- 

## Export Self-Signed Live Data Certificates

Live Data installation includes the generation of self-signed certificates. If you choose to work with these self-signed certificates (rather than producing your own CA certificate or obtaining a CA certificate from a

third-party certificate vendor), first export the certificates from Live Data and Cisco Unified Intelligence Center. You must export from both Side A and Side B of the Live Data and Cisco Unified Intelligence Center servers. Once done, import the certificates into Finesse, importing both Side A and Side B certificates into each side of the Finesse servers.

When using other self-signed certificates, agents must accept the Live Data certificates in the Finesse desktop when they sign in to use the Live Data gadget.

- 
- Step 1** Sign in to Cisco Unified Operating System Administration on the Live Data server using the following URL: <https://hostname of Live Data server/cmplatform>.
- Step 2** From the **Security** menu, choose **Certificate Management**.
- Step 3** Click **Find**.
- Step 4** Perform one of the following:
- If the tomcat-trust certificate for your server is not on the list, click **Generate New**. When the certificate generation is complete, reboot your server. Then restart this procedure.
  - If the tomcat-trust certificate for your server is on the list, click the certificate to select it. (Ensure that the certificate you select includes the hostname for the server.)
- Step 5** Click **Download .PEM File** and save the file to your desktop.  
Perform these steps for both Side A and Side B.
- Step 6** After you have downloaded the Live Data certificates, sign in to Cisco Unified Operating System Administration on the Cisco Unified Intelligence Center server using the following URL: <https://hostname of CUIC server/cmplatform>, and repeat steps 2 to 5.
- 

### What to do next

Import the Live Data and Cisco Unified Intelligence Center certificates into the Finesse servers.

## Import Self-Signed Live Data Certificates

To import the certificates into the Finesse servers, use the following procedure:

- 
- Step 1** Sign in to Cisco Unified Operating System Administration on the Finesse server using the following URL: <https://FQDN of Finesse server:8443/cmplatform>
- Step 2** From the **Security** menu, choose **Certificate Management**.
- Step 3** Click **Upload Certificate**.
- Step 4** From the **Certificate Name** drop-down list, choose **tomcat-trust**.
- Step 5** Click **Browse** and browse to the location of the Live Data or Cisco Unified Intelligence Center certificate (with the **.pem** file extension).
- Step 6** Select the file, and click **Upload File**.
- Step 7** Repeat steps 3 to 6 for the remaining unloaded certificate.



**Step 8** After you upload both certificates, restart Cisco Finesse Tomcat on the Finesse server.

---

**What to do next**

Perform these steps for both Side A and Side B.

## Obtain and Upload Third-party CA Certificate

You can use a Certification Authority (CA) certificate provided by a third-party vendor to establish an HTTPS connection between the Live Data, Finesse, and Cisco Unified Intelligence Center servers.

To use third-party CA certificates:

- From the Live Data servers, generate and download Certificate Signing Requests (CSR) for root and application certificates.
- Obtain root and application certificates from the third-party vendor.
- Upload the appropriate certificates to the Live Data, Unified Intelligence Center, and Finesse servers.

Follow the instructions provided in the *Unified CCE Solution: Procedure to Obtain and Upload Third-Party CA certificates (Version 11.x)* technical note at : <https://www.cisco.com/c/en/us/support/docs/customer-collaboration/unified-contact-center-enterprise-1101/200286-Unified-CCE-Solution-Procedure-to-Obtai.html> .





## APPENDIX **C**

# Certificates for Cisco Identity Service

---

- [Export Cisco Identity Service Certificates, on page 243](#)
- [Import Cisco IdS Certificates, on page 243](#)

## Export Cisco Identity Service Certificates

---

- Step 1** Sign in to Cisco Unified Operating System Administration on the Cisco IdS server using the following URL:  
`https://hostname of Cisco IdS server: 8443/cmplatform.`
- Step 2** Select **Security > Certificate Management**.
- Step 3** Enter the search criteria as **tomcat-trust** and then click **Find** to filter the certificate.
- The tomcat-trust certificates list is displayed. If you do not find the tomcat certificate for your server in the **Certificate List**, then click **Generate Self-signed**. When the certificate generation is complete, reboot your server. Then restart this procedure. For more information on generating the self-signed certificate, see *Cisco Unified Operating System Administration Online Help*.
- Step 4** Click the tomcat-trust certificate hyperlink in the **Common Name** column.  
The tomcat **Certificate Details** dialog box is displayed.
- Step 5** Click **Download .PEM File**.
- Step 6** Save the .PEM file in your local machine.
- 

### What to do next

- Perform the same steps for both the primary and secondary Finesse nodes.
- Import the Cisco IdS certificates into the Finesse trust store as tomcat-trust.

## Import Cisco IdS Certificates

---

- Step 1** Sign in to Cisco Unified Operating System Administration on the Finesse server using the following URL: `https://FQDN of Finesse server: 8443/cmplatform.`

- Step 2** Select **Security > Certificate Management > Upload Certificate/Certificate chain**.
- Step 3** From the **Certificate Purpose** drop-down list, select **tomcat-trust**.
- Step 4** In the **Upload File** field, click **Choose File** and browse to the **tomcat.pem** file that you saved on your system.
- Step 5** Click **Upload**.
- Step 6** Reboot the Cisco Finesse node.

**Note** Perform the same steps for both the primary and secondary Finesse nodes.

---