Cisco Finesse Administration Guide, Release 12.0(1)

First Published: 2019-01-11

Last Modified: 2020-07-02

CONTENTS

Preface

This guide describes how to administer Cisco Finesse.

Change History

The following table lists the changes made to this guide for Cisco Finesse 12.0(1) release version:

Change	Date
Added Edge Chromium browser details	2020
Set Up CA Certificate for Chrome and Edge Chromium (Microsoft Edge) Browsers	
Added desktop chat search.	
Added Content Security Policy directives.	
Added desktop chat organization unit (OU) search.	
Added new service property configuration CLI for port 5223.	
Added customize icon details.	

Change	Date
Initial Release of Document for Release 12.0(1)	2019
Look and feel of the Admin console is enhanced as part of user interface refresh.	
Cisco Finesse admin console is supported on Microsoft Edge.	
Configure the call header and up to five call variables in the Call Variable Layout. These variables are displayed in the agent's call popover and active call details in the Team Performance gadget.	
Set the CTI Encryption in the CTI Server Settings gadget from the Administration Console. You can test the CTI connection for the given configuration.	
When you modify the layout of the Finesse desktop, it can take up to 120 seconds to reflect the changes.	
Reason code tables support search across reason codes and reason code labels.	
The following changes are made in the Default Layout XML for 12.0(1): • managedBy Attribute is added in the Live Data gadget. • hidden attribute added to support headless gadgets. • MaxRow is changed from being a query parameter to an attribute. Queue Statistics and Query Parameter gadget URLs are different and are replaced automatically during an upgrade.	
The following attributes are added in the Default Layout XML and can be customized: • Horizontal Header • Title and Logo • Icons in the left navigation bar	
Desktop Chat Server settings can be configured from the admin console. Desktop Chat failover scenarios are added.	
The support for in-built java script components is added.	

Change	Date
The following CLIs are added: • To view the property values of any property file. • To set the property values of any property file. • To enable CORS for both Cisco Finesse and OpenFire and to configure the allowed origin list. • To configure media channels from which the users are signed out. • To enable or disable Gadget Source Allowed List functionality and to configure source(s) in the allowed list.	
New configuration parameters are added for the desktop via CLIs: • To enable or disable active call details in the team performance gadget. • To enable or disable view history in the team performance gadget. • To specify unsupported file types in Desktop Chat. • To set the maximum attachment size in Desktop Chat. • To configure the Wrap-Up timer via CLI. The showWrapUpTimer property can be used to show or hide timer in wrap-up state. • To configure Force Wrap-Up Reason via CLI for agents by the administrator. • To set the desktop notification connection type. By default it is WebSockets.	
Workflows and Workflow actions can be created for voice and digital channels.	

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
Cisco Finesse Documentation Guide	https://www.cisco.com/en/US/partner/products/ps11324/products_documentation_roadmaps_list.html
Configure SNMP Trap in Cisco Finesse	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	Boldface font is used to indicate commands, such as user entries, keys, buttons, folder names, and submenu names. For example: • Choose **Edit** > **Find**. • Click **Finish**.
italic font	Italic font is used to indicate the following: • To introduce a new term. Example: A *skill group* is a collection of agents who share similar skills. • A syntax value that the user must replace. Example: IF (*condition, true-value, false-value*) • A book title. Example: See the *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide*.
`window font`	Window font, such as Courier, is used for the following: • Text as it appears in code or that the window displays. Example: `<html><title>Cisco Systems, Inc. </title></html>`
`< >`	Angle brackets are used to indicate the following: • For arguments where the context does not allow italic, such as ASCII output. • A character string that the user enters but that does not appear on the window such as a password.

CHAPTER **1**

Getting Started

This chapter describes the interfaces used to configure, administer, and maintain Cisco Finesse and how to access them.

- User Accounts, on page 1
- Administration Tools, on page 1
- Certificate Management, on page 5
- QoS Settings, on page 12
- Localization, on page 13

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, Context Service Management, 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.

Cisco Finesse Administration Guide, Release 12.0(1)

1

- **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.

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.

Sign In to Cisco Finesse Administration Console

The Cisco Finesse administration console supports both HTTP and secure HTTP (HTTPS). Whether the administration console uses HTTP or HTTPS depends on whether HTTPS Redirect is enabled (by default, HTTPS Redirect is enabled). The URLs in this procedure use HTTP.

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

Procedure

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

Note   Ensure that the self-signed certificate provided with 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 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 HTTP to access the administration console, this step is not required.

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 the *Cisco Finesse Installation and Upgrade Guide*

Option	Description
Internet Explorer:	a. A page appears that states this site is untrusted. b. Click **More information** > **Go on to the webpage**.
Firefox:	a. A page appears that states this connection is untrusted. b. Click **I Understand the Risks**, and then click **Add Exception**.

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

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, 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 HTTP or HTTPS port in the sign in URL in Step 1 of the preceding procedure.

• For HTTPS access, enter:

https://<*FQDN*>:8445/cfadmin

• For HTTP access, enter:

http://<*FQDN*>:8082/cfadmin

The remaining steps of the sign in procedure remain the same for 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. This reset does not apply to the administrator account.

To view if a user account is locked, enter the **file get activelog desktop recurs compress** CLI command.

Extract the zipped output and search the catalina.out logs (/opt/cisco/desktop/finesse/logs/catalina.out) for the following message referring to the locked username:

```
An attempt was made to authenticate the locked user "<username>"
```

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

Procedure

Step 1	Direct your browser to https://*FQDN:8443/cmplatform*, where *FQDN* is the fully-qualified domain name 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.

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*.

Note Updating Cisco Finesse tomcat-trust certificate causes a temporary outage due to the impact on Cisco Notification Service. It is recommended to plan the certificate update during a maintenance window.

Procedure

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. Do not select "Multi-server (SAN)" as Multi-Server Subject Alternate Name (SAN) Certificates are not supported with Cisco Finesse.

   For information on updating Subject Alternate Names (SANs), refer to *Configuration Examples and TechNotes > Unified CCE Solution: Procedure to Obtain and Upload Third-Party CA certificates*.

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.

Procedure

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.

Procedure

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 icon]

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

Procedure

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:

Procedure

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.

Procedure

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

Procedure

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.

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.

Procedure

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.

a) Sign in to Cisco Unified Operating System Administration on the primary Finesse server (http://*FQDN*:8443/cmplatform, where *FQDN* is the fully qualified domain name of the Finesse server).

b) Click **Security** > **Certificate Management**.

c) Click **Upload Certificate**.

d) From the Certificate Name drop-down list, select **tomcat-trust**.

e) Click **Browse** and navigate to the tomcat.pem file that you downloaded in the previous step.

f) 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
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: http://*FQDN*/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 Context data (WrapUp Reasons, call variables, and ECC variables) is Unicode enabled and independent of the desktop locale.

The following restrictions apply to Call Context data with localized characters:

Variable	Limit
Wrap-Up Reasons	Limited to 40 bytes of UTF-8 data.
Call Variables 1-10	Limited to 40 bytes of UTF-8 data. **Note** If Finesse sends a set call data request that exceeds 40 bytes of data, the request fails.
ECC Variables	UTF-8 data is limited to the maximum size in bytes for ECC variables specified in Unified CCE.

If any limits in this table are exceeded, the variable data is truncated. This is more likely with localized characters that occupy more than one byte in size. For example, characters with an accent require two bytes to store one character and Asian characters require three or four bytes.

Agent first and last names appear on the desktop as they are defined in the Unified CCE database. If the names contain Japanese, Chinese, or Korean characters, they appear correctly on the desktop. However, the maximum supported size for the agent first and last names in these languages is 10 bytes. If the names exceed 10 bytes, they are truncated.

For details on setting the correct Windows locale and SQL collation settings for Unified CCE, See the *Cisco Unified Contact Center Enterprise Installation and Upgrade Guide*.

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

You can configure CTI server, Administration & Data server, cluster settings, Finesse IP Phone Agent (IPPA), and Cisco Context Service settings on the Settings tab of the Cisco Finesse administration console.

For information about Finesse IPPA settings, see *Manage Finesse IP Phone Agent*.

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.
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, *ucceinstance_*awdb).
Domain	(Optional) The domain name of the AWDB.
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.

Procedure

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 18. 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 12.0.

> **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 *Serviceability Guide for Cisco Unified ICM/Contact Center Enterprise*. 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. 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.

2. Validation is done to check if the provided Host/IP is resolved by Finesse box.

3. Validation is done to check if AW Database is reachable and if a valid path ID is configured for the provided Peripheral ID.

4. 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

5. 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.

6. 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.

Procedure

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

http://*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.

Procedure

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**.

Context Service Settings

Cisco Context Service is a cloud-based omnichannel solution for Unified CCE. It captures your customer's interaction history by providing flexible storage of customer-interaction data across any channel.

Context Service works out-of-the-box with Cisco Customer Collaboration products. Context Service also provides an SDK interface for integration with your own applications or third-party applications to capture end-to-end customer-interaction data.

For more information about Context Service and to check service availability, see https://help.webex.com/ community/context-service.

Context Service Network Connectivity Requirements

Context Service requires the call center components using Context Service to be able to connect to the public Internet.

Context Service uses port 443 (HTTPS).

The following URLs must be allowed list in your firewall so that your contact center components can connect to, and receive data from Context Service.

- `*.webex.com`

- `*.wbx2.com`

- `*.ciscoccservice.com`

Note Use wildcard URLs in your allowed list as Context Service is accessed through multiple subdomains. Context Service subdomain names can dynamically change.

If you register Context Service by enabling the proxy setting option, configure the browser proxy with the URL specified in the Context Service Management Gadget. Refer to the following links to configure the proxy settings for the related browsers:

Chrome	https://support.google.com/chrome/answer/96815?hl=en
Firefox	https://support.mozilla.org/en-US/kb/ advanced-panel-settings-in-firefox
Internet Explorer	https://windows.microsoft.com/en-in/windows/ change-internet-explorer-proxy-server-settings#1TC=windows-7

Configure Context Service Settings

Use the Context Service Management gadget to register Cisco Finesse with the Context Service.

Procedure

Step 1 Sign in to the Cisco Finesse administration console.

Step 2 To register Cisco Finesse with the Context Service, in the Context Service Management gadget, click **Register**.

Note Before initiating Context Service registration you must make sure pop-ups are enabled.

If the Finesse FQDN is not added as an exception in the block popup windows settings of the browser, the registration and deregistration popup windows do not close automatically. You have to manually close the pop-up windows.

If you are not able to see the **Register** button and a message appears asking you to refresh the page, clear your browser cache and try again.

If you wish to configure a Proxy Server for Context Service, check the **Enable Proxy Setting** option, enter the following Client Setting parameters and click **Save**.

Field	Description
Proxy Server URL	Proxy Server address
Timeout	The number of milliseconds (ms) the system waits before rejecting the Context Service cloud connectivity. Default: 1000 milliseconds Range: 200 to 15,000 milliseconds.
Lab Mode	Radio button indicates if the Context Service is in production or lab mode. • Enable—Context Service switches to lab mode. • Disable (default)—Context Service is in production mode.

Click **Register** to configure Cisco Finesse with Context Service.

Note If changes are made to the Context Service Parameters, do not reregister unless the Context Service connectivity takes more than 30 seconds.

Step 3 You are prompted to sign in and enter your Cisco Cloud Collaboration Management admin credentials to complete the registration.

Step 4 After a successful registration, if you want to deregister Cisco Finesse from the Context Service, click **Deregister**.

Note If you wish to cancel the registration, click **Cancel**.

If registration fails or context service cannot be reached, click **Register** to register again.

Note	If you use Firefox, enable the **dom.allow_scripts_to_close_windows** config to ensure that any additional tabs opened for context service registration close as expected. To perform this:

    a. Enter `about:config` in the Firefox browser.

    b. Click **I accept the risk**.

    c. Search for `dom.allow_scripts_to_close_windows` config.

    d. Double click to change the value field to `True`.

    e. Restart your browser.

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 supported with the unrestricted versions of IM&P only if Finesse is accessed via HTTP. To access Finesse using HTTP, use the **utils finesse application_https_redirect disable** CLI.

Configure Desktop Chat Server Settings

Procedure

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.

CHAPTER 3

Manage Call Variables Layouts

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).

- After an upgrade from a release earlier than Cisco Finesse Release 11.0, Finesse migrates the previously configured default layout and assigns it the default name (Default Layout) and description (Layout used when no other layout matches the user layout Custom/ECC Variable).

- 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.

Configure Call Variables Layouts

Procedure

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 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 7   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

Procedure

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

Procedure

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 4

Manage Desktop Layout

You can define the layout of the Finesse desktop on the Desktop Layout tab.

☞

Important Requirements, such as processor speed and RAM, for clients that access the Finesse desktop can vary. Desktops that receive events for more than one agent (such as agent and supervisor desktops running Live Data reports that contain information about other agents and skill groups) require more processing power than desktops that receive events for a single agent.

Factors that determine how much power is required for the client include, but are not limited to, the following:

- Contact center traffic

- Additional integrated gadgets in the desktop (such as Live Data reports or third-party gadgets)

- Other applications that run on the client and share resources with the Finesse desktop

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/.

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.

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.

- **Finesse Default Layout XML** - Expands to show the layout XML for the default Finesse desktop.

- **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 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:

- 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.

- 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".

- **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, it can take up to 120 seconds to reflect the changes. However, agents who are signed in when the changes are made must sign out and sign in again to see those changes reflecting on the desktop.

Note The call control gadget is only supported at the page level. You must ensure that the call control gadget (<gadget>/desktop/scripts/js/callcontrol.js</gadget>) is placed within the <page></page> tag for it to work correctly. Don't place this gadget within a <tab></tab> tag.

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.

Procedure

Step 1	Click **Desktop Layout**.
Step 2	In the Finesse Layout XML area, 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>http://<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>
    <tab>
        <id>manageCustomer</id>
        <icon>profile-settings</icon>
        <label>finesse.container.tabs.agent.manageCustomerLabel</label>
        <gadgets>
            <gadget>/3rdpartygadget/files/FinextGadget.xml</gadget>
```

```
            </gadgets>
        </tab>
```

Step 3   Click **Save**.

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

Step 4   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.

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 colums) 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:

Procedure

Step 1   Click **Desktop Layout**.

Step 2   Enter the product name in the config value tag with title key.

Step 3   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 4   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:

Procedure

Step 1   Click **Desktop Layout**.

Step 2   Enter name of the gadget or component in the id tag.

Step 3   Enter the value of the icon in the icon tag.

Step 4   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

.jsp?gadgetHeight=280&amp;viewId=5FA44C6F930C4A64A6775B21A17EED6A&amp;
                        filterId=agentTaskLog.id=CL%20teamName</gadget>
                    <gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget

.jsp?gadgetHeight=280&amp;viewId=56BC5CCE8C37467EA4D4EFA8371258BC&amp;
                        filterId=agentStateLog.id=CL%20teamName</gadget>
                    </gadgets>
              </column>
            </columns>
        </tab>
        <tab>
            <id>manageCustomer</id>
            <icon>/3rdpartygadgets/<some_sample_image>
            <label>finesse.container.tabs.agent.manageCustomerLabel</label>
            <gadgets>
                <gadget>/desktop/gadgets/CustomerContext.xml</gadget>
```

```
            </gadgets>
        </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.

Procedure

Step 1   Click **Desktop Layout**.

Step 2   Enter the value of the icon in the icon tag. Get the icon name from the . The icon name is located on the right of the icon image. For example, search.

Icon	Name		Icon	Name
🔍	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 .

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>
```

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

510879

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

510880

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		

510881

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.

👤	contact
👤+	add-contact
👤-	remove-contact
📇	directory
📇	contact-card
⭐	star

510884

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

510885

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		

510886

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		

510887

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		

510888

The following are the icons for Media.

⬛	image
⬛	sound
♫	music
◕	graph
≡	text
⁞	tables
⁞	zip

510889

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		

510890

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		

510891

The following are the icons for Notifications and Alerts.

⚠	warning	☑	quality	📍	location
❗	alert-badge	☒	broken-image	🧭	compass
✖	error	⊘	blocked	🚩	flagged
ℹ	info	✓	check	A★	keywords
?	help	✔	certified	DMS	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			

510894

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	

510895

The following are the icons for Video Controls.

▶ play

⏺ play-container

⬛ stop

⏸ pause

⏭ skip-fw

⏮ skip-bw

⏩ ffw

⏪ fbw

⬤ circle

510896

The following are the icons for Miscellaneous Icons.

circle-bar chart	circle-pie chart	line-chart	D
circle-column chart	unknown-cstomer	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	

510898

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	

510899

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	

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	

510900

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		

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/ccbu-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
 alredy 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 -->
```

```xml
<!-- 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>
```

```xml
<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 either HTTP or HTTPS for both Cisco Unified Intelligence Center and Finesse. You cannot use HTTP for one and HTTPS for the other. The default setting for both after a fresh installation is HTTPS. If you want to use HTTP, you must enable it on both Cisco Unified Intelligence Center and Finesse.

  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.
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. The gadgets are divided into two categories: HTTPS version of Live Data gadgets and HTTP version of Live Data gadgets.

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.

Procedure

Step 1 Click **Desktop Layout**.

Step 2 Remove the comment characters (<!-- and -->) from each report that you want to add to the desktop layout. Make sure that you choose the reports that match the method that your agents use to access the Finesse desktop (HTTP or HTTPS).

Step 3 Replace "my-cuic-server" with the FQDN of your Cisco Unified Intelligence Center Server.

Step 4 Optionally, change the gadget height.

Example:

The height that is specified in the Live Data gadget URLs is 310 pixels. If you want to change the height, change the gadgetHeight parameter in the URL to the desired value. For example, if you want the gadget height to be 400 pixels, change the code as follows, replacing 310 with 400:

```
<gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget.jsp?
        gadgetHeight=400&viewId_1=99E6C8E210000141000000D80A0006C4&
```

```
                    filterId_1=agent.id=CL%20teamName&viewId_2=9AB7848B10000141000001C50A0006C4&
                    filterId_2=agent.id=CL%20teamName
                    </gadget>
```

To maintain the optimal display of the gadget with scroll bars, set the value for the gadget height to a minimum of 200 pixels. If the report does not require scroll bars, for example a one-row report, you can set a smaller gadget height (for example, 100 pixels). If you do not specify anything for the gadget height (if you remove the 310 from the URL), it defaults to 170 pixels.

Step 5   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. The gadgets are divided into two categories: HTTPS version of Live Data gadgets and HTTP version of Live Data gadgets.

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

Procedure

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

Step 2   Click **Finesse Default Layout XML** to show the default layout XML.

Step 3   Copy the XML code for the report you want to add from the Finesse default layout XML. If your agents use HTTP to access Finesse, copy the XML code for the HTTP report. If they use HTTPS, copy the XML code for the HTTPS report.

Example:

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

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

Step 4   Paste the XML within the tab tags where you want it to appear.

Example:

To add the report to the home tab of the agent desktop:

```
<layout>
  <role>Agent</role>
  <page>
    <gadget>/desktop/gadgets/CallControl.jsp</gadget>
  </page>
```

```
<tabs>
  <tab>
    <id>home</id>
    <label>finesse.container.tabs.agent.homeLabel</label>
    <gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget.jsp?
            gadgetHeight=310&viewId_1=99E6C8E210000141000000D80A0006C4&
            filterId_1=agent.id=CL%20teamName&
            viewId_2=9AB7848B10000141000001C50A0006C4&
            filterId_2=agent.id=CL%20teamName
    </gadget>
  </tab>
  <tab>
    <id>manageCall</id>
    <label>finesse.container.tabs.agent.manageCallLabel</label>
  </tab>
</tabs>
</layout>
```

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

Step 6   Optionally, change the gadget height.

Example:

The height specified in the Live Data gadget URLs is 310 pixels. If you want to change the height, change the gadgetHeight parameter in the URL to the desired value. For example, if you want the gadget height to be 400 pixels, change the code as follows:

```
<gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget.jsp?
        gadgetHeight=400&viewId_1=99E6C8E210000141000000D80A0006C4&
        filterId_1=agent.id=CL%20teamName&viewId_2=9AB7848B10000141000001C50A0006C4&
        filterId_2=agent.id=CL%20teamName
        </gadget>
```

To maintain the optimal display of the gadget with scroll bars, set the value for the gadget height to a minimum of 200 pixels. If the report does not require scroll bars, for example a one-row report, you can set a smaller gadget height (for example, 100 pixels). If you do not specify anything for the gadget height (if you remove the 310 from the URL), it defaults to 170 pixels.

Step 7   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. The gadgets are divided into two categories: HTTPS version of Live Data gadgets and HTTP version of Live Data gadgets.

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

Procedure

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

Step 2   Click **Finesse Default Layout XML** to show the default layout XML.

Step 3   Copy the XML code for the report you want to add from the Finesse default layout XML. If your agents use HTTP to access Finesse, copy the XML code for the HTTP report. If they use HTTPS, copy the XML code for the HTTPS report.

Example:

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

```
<gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget.jsp?
        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   Paste the XML within the tab tags where you want it to appear.

Example:

To add the report to the home tab of the agent desktop:

```
<layout>
  <role>Agent</role>
  <page>
    <gadget>/desktop/gadgets/CallControl.jsp</gadget>
  </page>
  <tabs>
    <tab>
      <id>home</id>
      <label>finesse.container.tabs.agent.homeLabel</label>
      <gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget.jsp?
              gadgetHeight=310&viewId_1=99E6C8E210000141000000D80A0006C4&
              filterId_1=agent.id=CL%20teamName&
              viewId_2=9AB7848B10000141000001C50A0006C4&
              filterId_2=agent.id=CL%20teamName
      </gadget>
    </tab>
    <tab>
      <id>manageCall</id>
      <label>finesse.container.tabs.agent.manageCallLabel</label>
    </tab>
  </tabs>
</layout>
```

Step 9   Replace "my-cuic-server" with the FQDN of your Cisco Unified Intelligence Center Server.

Step 10   Optionally, change the gadget height.

Example:

The height specified in the Live Data gadget URLs is 310 pixels. If you want to change the height, change the gadgetHeight parameter in the URL to the desired value. For example, if you want the gadget height to be 400 pixels, change the code as follows:

```
<gadget>https://my-cuic-server:8444/cuic/gadget/LiveData/LiveDataGadget.jsp?
        gadgetHeight=400&viewId_1=99E6C8E210000141000000D80A0006C4&
        filterId_1=agent.id=CL%20teamName&viewId_2=9AB7848B10000141000001C50A0006C4&
        filterId_2=agent.id=CL%20teamName
        </gadget>
```

To maintain the optimal display of the gadget with scroll bars, set the value for the gadget height to a minimum of 200 pixels. If the report does not require scroll bars, for example a one-row report, you can set a smaller gadget height (for example, 100 pixels). If you do not specify anything for the gadget height (if you remove the 310 from the URL), it defaults to 170 pixels.

Step 11   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.

Procedure

Step 1   Click the **Desktop Layout**.

Step 2   Click **Finesse Default Layout XML** to show the default layout XML.

Step 3   Copy the gadget URL for the report you want to modify from the Finesse default layout XML and paste it into a text editor.

Example:

If you want to modify the Agent Report for HTTPS, copy the following URL and paste it into a text editor:

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

Step 4   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 5   Copy the permalink of the customized report from the **HTML Link** field, and paste it in a text editor. Then copy the viewId value from this link into the desired view.

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 6   Replace the desired viewId value in the gadget URL with the viewId value from the permalink of the customized report.

Note   For more information on modifying reports, see *Cisco Unified Intelligence Center User Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-intelligence-center/products-user-guide-list.html.

Step 7   Replace my-cuic-server with the FQDN of the Cisco Unified Intelligence Center Server.

Step 8   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.

Procedure

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.jsp?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.jsp?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.jsp?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.jsp?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.jsp?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**.

Configure Live Data Reports with Multiple Views

CHAPTER **5**

Manage Phone Books

On the Phone Books tab of the Cisco Finesse administration console, you can create and manage global and team phone books and phone book contacts. Global phone books are available to all agents; team phone books are available to agents in that team.

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 1500.

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).

Field	Explanation
Last Name	The last name of a contact. The last name can be a maximum of 128 characters. This field is optional.
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

Procedure

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

Procedure

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

Procedure

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 1500 contacts. Import lists that contain more than 1500 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	Alphanumeric characters
Last Name	128	Yes	**Note** The CSV file that contains the contacts to import must use Latin encoding.
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.

Procedure

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.

Procedure

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

Procedure

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

Procedure

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

Procedure

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 6

Manage Reasons

The Reasons tab on the Cisco Finesse administration console allows you to view, add, edit, and delete Not Ready reason codes, Sign Out reason codes, and Wrap-Up reasons.

The reason codes you configure in Finesse are not automatically populated in Unified CCE. To populate them across the solution, you must configure the reason codes in both Finesse and Unified CCE.

Note Reason code tables support search across reason codes and reason code labels. You can configure different reason codes with the same reason code label across various teams.

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

Procedure

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

Procedure

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.

Procedure

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.

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

Procedure

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

Procedure

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.

Procedure

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.

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:

		• When the agent goes off-hook to place a call, the Finesse desktop 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.
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.
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.

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.

Wrap-Up Reasons

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

You can configure Wrap-Up reasons to be available globally to all agents or only to specific teams.

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.

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

- Set the Work mode on incoming attribute to either *Optional* or *Required*.

- Set the Work mode on outgoing attribute to either *Optional* or *Not Allowed*.

If the Work mode on incoming attribute is set to Required, agents automatically transition to wrap-up state after an incoming or Outbound Option call ends. If the Work mode on incoming attribute is set to Optional, agents must select Wrap-Up from the agent state drop-down list while on a call to transition to wrap-up state when the call ends. If the agent does not select Wrap-Up during the call, the agent does not transition to wrap-up state when the call ends.

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.

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

Procedure

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

Procedure

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

Procedure

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 transfering 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 Team Resources

You can assign phone books, reason codes, wrap-up reasons, custom desktop layouts, and workflows to teams on the Team Resources tab of the administration console.

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.

Tab Name	Description
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.
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

Procedure

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

Procedure

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

Procedure

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	Edit the XML.
Step 5	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.

Assign Workflows to Team

Procedure

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

Procedure

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 **8**

Manage Workflows

On the Workflows tab of the Cisco Finesse administration console, you can create and manage workflows and workflow actions.

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

  - 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, it's 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.

Procedure

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= [callVariable1 ✖] & [callVariable2 ✖]

   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.

Procedure

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:

Note   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>
<requestedAction>START_RECORDING</requestedAction>
<targetMediaAddress> extension ✖ </targetMediaAddress>
</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

Procedure

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

Procedure

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

Procedure

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

Procedure

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

Procedure

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.

Manage Security

HTTP and HTTPS Support

The Cisco Finesse administration console and agent desktop support both HTTP and secure HTTP (HTTPS). 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.

To access the administration console using HTTP, enter the following URL:

http://*FQDN*/cfadmin

Similarly, agents and supervisors can access their desktops using HTTP or HTTPS as follows:

  • http://*FQDN*/desktop

  • https://*FQDN:8445*/desktop

For HTTPS access, you can eliminate browser security warnings by choosing to trust the self-signed certificate provided with Finesse or uploading a CA certificate.

By default, HTTPS access is enabled. You can run the Cisco Finesse HTTPS Redirect CLI command to disable HTTPS and allow HTTP access for the Finesse administration console and the agent desktop.

If you add custom gadgets that perform HTTPS requests to Finesse, you must add a certificate to the Finesse server for that gadget.

Finesse HTTPS Redirect

Enable Cisco Finesse HTTPS Redirect to enforce HTTPS to access the Finesse desktop and administration console. If Cisco Finesse HTTPS Redirect is enabled, agents and supervisors who attempt to access the desktop

with HTTP are redirected to HTTPS. Administrators who attempt to access the administration console with HTTP are also redirected to HTTPS.

If Cisco Finesse HTTPS Redirect is disabled, the desktop and the administration console can be accessed with HTTP or HTTPS.

Note This command does not impact the Finesse REST APIs.

In a two-node setup, if you enable or disable HTTPS Redirect only on the primary Finesse server, the setting does not replicate to the secondary Finesse server. You must enter the required commands on the primary and secondary Finesse server.

Use the folloing commands to view the status of, enable, or disable Cisco Finesse HTTPS Redirect:

- **utils finesse application_https_redirect status**: This command retrieves the status of Cisco Finesse HTTPS Redirect. It displays whether Cisco Finesse HTTPS Redirect is currently enabled or disabled on the system.

  Note On the secondary server, the HTTPS redirect status appears as enabled for the Finesse Agent Desktop only. For Finesse Admin, the HTTPS redirect status always appears as disabled on the secondary server because Finesse Admin is not available on the secondary server.

- **utils finesse application_https_redirect enable**: This command enables Cisco Finesse HTTPS Redirect.

  You must stop the Cisco Finesse Tomcat Service before you can enable Cisco Finesse HTTPS Redirect. You can use the following command to stop this service: **utils service stop Cisco Finesse Tomcat**.

  If the Cisco Finesse Tomcat Service is not stopped, the command to enable Cisco Finesse HTTPS Redirect fails. This command also fails if Cisco Finesse HTTPS Redirect is already enabled.

  After you enable Cisco Finesse HTTPS Redirect, start the Cisco Finesse Tomcat Service using the command **utils service start Cisco Finesse Tomcat** .

- **utils finesse application_https_redirect disable**: This command disables Cisco Finesse HTTPS Redirect.

  You must stop the Cisco Finesse Tomcat Service before you can disable Cisco Finesse HTTPS Redirect. You can use the following command to stop this service: **utils service stop Cisco Finesse Tomcat**.

  If the Cisco Finesse Tomcat Service is not stopped, the command to disable Cisco Finesse HTTPS Redirect fails. This command also fails if Cisco Finesse HTTPS Redirect is already disabled.

  After you disable Cisco Finesse HTTPS Redirect, start the Cisco Finesse Tomcat Service using the command **utils service start Cisco Finesse Tomcat**.

HSTS

Finesse supports HTTP Strict Transport Security (HSTS) for increased security. HSTS is automatically enabled when you enable HTTPS Redirect, 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, 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.

Procedure

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 list CLIs and adding the required URIs to the allowed list. For more information on Gadget Source Allowed List CLIs, see *Cisco Finesse CLIs*.

Manage Finesse IP Phone Agent

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.

HTTP Only

Finesse IPPA phone clients communicate with the Finesse server using HTTP only, whether or not HTTPS access is enabled on Finesse.

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 (by Administrator)	Prepopulated in Self Care Portal (by Agent)	Entered at Sign-In (by Agent)
Administrator populates the extension only	extension	-	id password
Administrator populates the ID and extension	id extension	-	password
Agents enter password only using Self Care Portal	id 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 extension password	-
Agents enter ID and extension only using Self Care Portal	-	id 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**
Set Up Application User, Web Access, and HTTPS Server Parameters	Required	Required	Required
Configure Finesse IP Phone Service in Unified CM	Required	Required	Required
Add Service Parameters for One Button Sign In	Not applicable	Required only with One Button Sign In	Required only with One Button Sign In
Subscribe Agent Phones to Manual Subscription Service	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**
Set Up Agent Access to the Self Care Portal	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.

Procedure

Step 1 Set the following parameters in Unified CM:

- Set the **Web Access** parameter to **Enabled**.

- Set the **HTTPS Server** parameter to **HTTP and HTTPS Enabled**.

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.

Procedure

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 Ensure that the **Service Category** is set to **XML Service**, and the **Service Type** is set to **Standard IP Phone Service**.

Step 7 Check the **Enable** check box.

Step 8 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 9 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, in the **Service URL** field, replace *Finesse FQDN* with the FQDN of the secondary server.

Note Since Finesse IPPA works only over HTTP, avoid using Secured Phone URL Parameters in Unified CM.

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.

Procedure

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
Parameter Name	Enter a parameter name in lower case exactly similar to — id, extension, and password. The values entered are the exact query string parameters used for the subscription URL.
Parameter Display Name	Enter a descriptive parameter name; for example, id, extension, and password.

Field	Description
Default Value	Leave the default value blank for all parameters.
Parameter Description	Enter a description of the parameter. The user can access this text when they subscribe to the service.
Parameter is Required	If the administrator prepopulates the parameter in Unified CM Administration, check the **Parameter is Required** box. However, if the agent prepopulates the parameter in the Self Care Portal, two options are available: • If the agents prepopulates all defined parameters, check the **Parameter is Required** box for each parameter. • 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.
Parameter is a Password (mask contents)	Check this box for the password only. 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.

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.

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.

CHAPTER 11

Manage Third-Party Gadgets

- 3rdpartygadget Account, on page 113
- Upload Third-Party Gadgets, on page 114

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.

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*.

To set (or reset) the 3rdpartygadget account password, access the CLI and run the following command:

utils reset_3rdpartygadget_password

You are prompted to enter a password. After you enter a password, you are prompted to confirm the password.

The password for the 3rdpartygadget account must be between 5 and 32 characters long and cannot contain spaces or double quotes (").

Note   If the third-party gadget hosted in Cisco Finesse is sending a REST request to the web server via Shindig, using the SHA256 certificate, the maximum key length cannot exceed 2048.

Note   Third-party gadgets are migrated across upgrades and included in DRS backup and restore.

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:

http://<finesse>/3rdpartygadget/files/

> **Note** For Unified CCX deployments you must specify port 8082.

To access the gadget uploaded in the previous example, use the following URL:

http://<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.

Perform Routine Maintenance

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.

- **Finesse Diagnostic Portal application:** Provides performance-related information for Finesse.

- **Finesse IP Phone Agent (IPPA) application:** Allows agents and supervisors to perform Finesse operations on their Cisco IP Phone.

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

Procedure

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.

- **finesse-dp:** Contains startup, error, and information logs generated by the Finesse Diagnostic Portal application.

- **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.

- **fippa:** Contains logs for the Finesse IP Phone Agent (IPPA) application.

- **finesse-auth:** Contains the logs for Finesse authentication with the Cisco Context Service.

- **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.

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.

- Context Service registration log: **file get activelog ccbu/logs/fusion-mgmt-connector**

Use this command to obtain the fusion-mgmt-connector logs generated by Finesse during the registration and deregistration with Cisco Context Service.

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.

- 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.

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.

Procedure

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.

Procedure

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.

- 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.

Cisco Finesse Notification Service Logging

Use the following commands to view the status of, enable, or disable Cisco Finesse Notification Service logging:

• **utils finesse notification logging status**: This command displays whether Cisco Finesse Notification Service logging is currently enabled or disabled on the system.

Note Ensure the Cisco Finesse Notification Service is running before you run the command to retrieve the status of Cisco Finesse Notification Service logging. If the service is not running, this command fails.

• **utils finesse notification logging enable**: This command enables Cisco Finesse Notification Service logging.

Note Ensure that the Cisco Finesse Notification Service is running before you run the command to enable Cisco Finesse Notification Service logging. If the service is not running, this command fails. This command also fails if Cisco Finesse Notification Service logging is already enabled.

If you enable logging and then restart the Cisco Finesse Notification Service, logging is automatically disabled.

• **utils finesse notification logging disable**: This command disables Cisco Finesse Notification Service logging.

Note Ensure that the Cisco Finesse Notification Service is running before you run the command to disable Cisco Finesse Notification Service logging. If the service is not running, this command fails. This command also fails if the Cisco Finesse Notification Service logging is already disabled.

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 **13**

Cisco Finesse Failover Mechanisms

CTI Failover

Benchmark Parameters: For a Contact Center with the capacity of 2000 logged in agents and 6000 to 12000 configured agents, it takes up to 120 to 150 seconds (with up to 200 milliseconds WAN delay) for the Finesse server to recover its state during the CTI Failover in a voice only deployment. Failover involving Digital Channels can take longer based on the tasks and MRDs configured. The Finesse client desktop failover will be initiated after the Finesse server is back IN_SERVICE and can take a few more minutes.

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.

- 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, Release12.5 (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.

- Cisco Finesse can connect to the secondary AWDB host.

- The Distributor service is running on the secondary AWDB host.

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.

If Cisco Finesse cannot connect to either of the Administration & Data servers and the cache has expired, the systems returns the following errors:

- Agents and supervisors who attempt to sign in to the Finesse desktop receive an "Invalid user ID or password" error message.

- Administrators cannot update or retrieve settings in the Cisco Finesse administration console.

- Users who are already signed in to Cisco Finesse receive an "Operation timed out" error message.

- Users who make API requests receive an 401 "Unauthorized" HTTP error message.

If Cisco Finesse loses connection to one AWDB and then receives requests, these requests may time out before Cisco Finesse can detect that the connection is down and connect to the alternate AWDB. In this scenario, the user (administrator, agent, or supervisor) may need to retry the operation for it to succeed.

Finesse Desktop Failover

With a two-node Finesse setup (primary and secondary Finesse servers), if the server that an agent is connected to goes out of service, the agent receives a notification that the connection with the server was lost. The Finesse desktop:

- Continues to check whether the current Finesse server recovers its state.

- Checks if the other Finesse server is available and in service.

If the other Finesse server is available, the desktop automatically signs the agent into the other server. If the current Finesse server recovers its state, the desktop notifies the agent that it has reconnected.

The Finesse smarter failover logic has three triggers to detect desktop failure:

- The Finesse desktop receives a SystemInfo event that the current server is OUT_OF_SERVICE.

- The user XMPP connection is disconnected.

- The "finesse" XMPP user presence changes to unavailable.

No matter which trigger is detected, the desktop reconnection logic is as follows:

1. Poll SystemInfo for current server every 20 seconds and the other Finesse server between 45-150 seconds.

2. If SystemInfo is IN_SERVICE, check the user XMPP connection.

3. If SystemInfo is IN_SERVICE, check if the lastCTIHeartbeatStatus is a success.

> **Note** This is to ensure that the second side is healthy before failover, and does not immediately go 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 second server might be slower to sense a network disconnect.

4. If XMPP is disconnected, make an user XMPP connection request.

5. If user XMPP is connected and the server is IN_SERVICE, refresh the data.

While polling SystemInfo every 20 seconds, the desktop also checks the availability of the alternate server every 45-150 seconds. The smarter failover logic prefers to stay with the current server. If the failover logic detects that the alternate server is available, it checks the current server one more time. If the current server has recovered, the desktop reconnects to the current server. If the current server is still down, the desktop connects the agent to the alternate server. In this case, the agent does not automatically reconnect to the failed server after it recovers but instead remains connected to the alternate server.

If the user XMPP connection 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 smarter failover logic begins.

Desktop failover can occur for the following reasons:

- The Cisco Finesse Tomcat Service goes down.

- The Finesse Webapp Service goes down.

- The Cisco Finesse Notification Service goes down.

- Finesse loses connection to both CTI servers.

> **Note** After Finesse failover, the pending state of an agent will not be displayed once the agent fails over to the secondary Finesse node. The pending state change is reflected on the desktop only after the call ends.

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 *Unavailable* from the client. Finesse waits 60 seconds, and then sends a forced logout request to the CTI server.	**Race Conditions** 1. The agent closes the browser window. Finesse receives a presence notification of *Unavailable* for the user. Finesse tries to sign the agent out; however, that agent is already signed out. 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.
			3. If the Finesse desktop is running over a slower network connection, Finesse may not always receive an *Unavailable* presence notification from the client browser. In this situation, the behavior mimics a browser crash, as described in the preceding condition.
			4. If agent is in Not Ready State before Failover, agent moves to Not Ready — Connection Failure after CTI Disconnect or Reconnect.
			If agent is in Ready State before Failover, agent moves to Not Ready — Connection Failure upon his next state change to Not Ready.
The client refreshes the browser	—	Finesse receives a presence notification of *Unavailable* 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 *Unavailable* from the client. Finesse is IN_SERVICE, so it sends a forced logout request to the CTI server for the agent.	**Race Conditions** 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. If agent is in Not Ready State before Failover, agent moves to Not Ready — Connection Failure after CTI Disconnect or Reconnect. If agent is in Ready State before Failover, agent moves to Not Ready — Connection Failure upon the next state change to Not Ready.
The Refresh Token has expired. For more information on tokens, see https://developer.cisco.com/docs/finesse/#single-sign-on-apis.	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 – System Failure immediately even though the call is still active.

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.

Backup and Restore

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://*FQDN*:8443/drf, where *FQDN* is the fully-qualified domain name 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.

Procedure

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.

Procedure

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

Procedure

Step 1   Access the DRS application (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.

Check Backup Status

Procedure

Step 1   Access the DRS application (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.

Procedure

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/drf).

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**.

Supported Cisco Unified Communications OS Services

• Supported Cisco Unified Communications OS Services, on page 143

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.

Cisco Finesse CLI

Commands Supported for Cisco Finesse

Finesse supports the following CLI commands and has qualified their use.

Finesse HTTPS Redirect

Enable Cisco Finesse HTTPS Redirect to enforce HTTPS to access the Finesse desktop and administration console. If Cisco Finesse HTTPS Redirect is enabled, agents and supervisors who attempt to access the desktop with HTTP are redirected to HTTPS. Administrators who attempt to access the administration console with HTTP are also redirected to HTTPS.

If Cisco Finesse HTTPS Redirect is disabled, the desktop and the administration console can be accessed with HTTP or HTTPS.

Note   This command does not impact the Finesse REST APIs.

In a two-node setup, if you enable or disable HTTPS Redirect only on the primary Finesse server, the setting does not replicate to the secondary Finesse server. You must enter the required commands on the primary and secondary Finesse server.

Use the folling commands to view the status of, enable, or disable Cisco Finesse HTTPS Redirect:

- **utils finesse application_https_redirect status**: This command retrieves the status of Cisco Finesse HTTPS Redirect. It displays whether Cisco Finesse HTTPS Redirect is currently enabled or disabled on the system.

    Note   On the secondary server, the HTTPS redirect status appears as enabled for the Finesse Agent Desktop only. For Finesse Admin, the HTTPS redirect status always appears as disabled on the secondary server because Finesse Admin is not available on the secondary server.

- **utils finesse application_https_redirect enable**: This command enables Cisco Finesse HTTPS Redirect.

    You must stop the Cisco Finesse Tomcat Service before you can enable Cisco Finesse HTTPS Redirect. You can use the following command to stop this service: **utils service stop Cisco Finesse Tomcat**.

    If the Cisco Finesse Tomcat Service is not stopped, the command to enable Cisco Finesse HTTPS Redirect fails. This command also fails if Cisco Finesse HTTPS Redirect is already enabled.

    After you enable Cisco Finesse HTTPS Redirect, start the Cisco Finesse Tomcat Service using the command **utils service start Cisco Finesse Tomcat** .

- **utils finesse application_https_redirect disable**: This command disables Cisco Finesse HTTPS Redirect.

    You must stop the Cisco Finesse Tomcat Service before you can disable Cisco Finesse HTTPS Redirect. You can use the following command to stop this service: **utils service stop Cisco Finesse Tomcat**.

    If the Cisco Finesse Tomcat Service is not stopped, the command to disable Cisco Finesse HTTPS Redirect fails. This command also fails if Cisco Finesse HTTPS Redirect is already disabled.

    After you disable Cisco Finesse HTTPS Redirect, start the Cisco Finesse Tomcat Service using the command **utils service start Cisco Finesse Tomcat**.

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 commans starts Cisco Finesse Tomcat.

- **utils service stop Cisco Finesse Tomcat**: This commans stops Cisco Finesse Tomcat.

- **utils service restart Cisco Finesse Tomcat**: This commans 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.

Cisco Finesse Trace Logging

Use the following commands to toggle trace logs for Cisco Finesse, enable trace logs for Finesse IPPA, and enable debug logs for realm.

Note Enabling trace logging may cause an overload in the system and must be used for debugging purposes only.

- **utils finesse trace enable**:

  This command allows you to:

  - Enable trace logs for Cisco Finesse.

  - Turn on command dispatcher logs.

  - Enable trace logs for Finesse IPPA.

  - Enable debug logs for Realm.

- **utils finesse trace disable**

  This command allows you to:

  - Disable trace logs for Cisco Finesse.

  - Turn off command dispatcher logs.

  - Disable trace logs for Finesse IPPA.

  - Disable debug logs for Realm.

| Note | After execution of each command, wait for 60 seconds for the changes to take effect. |

- **utils finesse trace status**

  This command allows you to displays status as:

  - Enabled - When all four actions are enabled.

  - Disabled - When all four actions are disabled.

  If all actions are not enabled or disabled, a warning message is displayed.

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 1500 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 charging 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 http://origin1.com:[port]

  utils finesse cors allowed_origin add http://origin1.com: [port], http://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:8082

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 whitelist list**: This command lists all the source(s) in the allowed list.

- **utils finesse gadget_source_check whitelist add**: This command adds source(s) to the allowed list. For example,

  - **utils finesse gadget_source_check whitelist add** https://www.abc.com:8445.

  - **utils finesse gadget_source_check whitelist 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 whitelist delete**: This command deletes source(s) from the allowed list. For example:

  - **utils finesse gadget_source_check whitelist delete**

    - 1: http://origin1:8080

    - 2: https://origin2:8082

- 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

Note   To enable this feature in Cisco Finesse, install Finesse 12.0(1) ES4 COP or higher.

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_whitelist 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_whitelist 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_whitelist 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_whitelist 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_whitelist list**—This command lists all the frame sources that are allowed to access Cisco Finesse.

```
admin:utils finesse frame_access_whitelist 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/.

Desktop Properties

Configure the desktop properties using the following CLIs for the features:

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.

- 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. By default the showWrapUpTimer property is set to true.

- 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**

Wrap-Up Timer Count Down

Use the following CLI commands to set the wrap-up timer to count down or count up the time. By default the wrapUpCountDown property is set to true.

- To count up the time:**utils finesse set_property desktop wrapUpCountDown false**

- To count down the time: **utils finesse set_property desktop wrapUpCountDown true**

Notification Connection Type

Use the following CLI commands to update the desktop notification connection type as WebSockets or BOSH. By default the connection type is WebSockets.

- For WebSockets: **utils finesse set_property desktop notificationConnectionType websocket**

- For BOSH: **utils finesse set_property desktop notificationConnectionType bosh**

Desktop Chat Attachment

Use the following CLI commands to enable or disable the attachment support in Desktop Chat. Attachments are enabled by default in Desktop Chat.

- To enable: **utils finesse set_property desktop desktopChatAttachmentEnabled true**

- To disable: **utils finesse set_property desktop desktopChatAttachmentEnabled false**

Desktop Chat Maximum Attachment Size

Use the following CLI commands to configure the attachment size in Desktop Chat. If you do not configure the maximum attachment size, then it is set to 5 MB by default.

- **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.

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 will not be supported in the Desktop Chat. Multiple file types can be added using a comma separated string.

- **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

Configure Desktop Chat Organization Unit (OU) Search

Note To run this CLI in Cisco Finesse, install Finesse 12.0(1) ES4 COP or higher.

Use the following CLI commands to configure the OU based user search for the base LDAP context for desktop chat in HCS for CC. By default, the whole LDAP base context is configured in Cisco Unified Communications Manager IM and Presence Service LDAP search settings. For more details on desktop search see, *Desktop Chat Server Settings*.

To set field key: **utils finesse set_property desktop desktopChatOUSearchFieldKey** *<value>*

To set field value: **utils finesse set_property desktop desktopChatOUSearchFieldValue** *<value>*

The following example displays the search criteria set for chat users who belong to specific OU.

```
admin:utils finesse set_property desktop desktopChatOUSearchFieldKey "OU"

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.

admin:utils finesse set_property desktop desktopChatOUSearchFieldValue "chat"

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.
```

Service Properties

Configure the service properties using the following CLI.

Enable or Disable Secure XMPP Socket—Port 5223

To run this CLI in Cisco Finesse, install Release 11.6(1) ES10 COP or higher.

To run this CLI in Cisco Finesse, install Finesse 12.0(1) ES4 COP or higher.

Use the following CLI commands to enable or disable the external access to the Cisco Finesse Notification Service XMPP port (5223). The port must be enabled for external access 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**

  • To display the current status: **utils finesse show_property webservices enableExternalNotificationPortAccess**

Note   Restart Cisco Finesse Tomcat and Cisco Finesse Notification Services for the changes to take effect.

Upgrade

Upgrade-related commands are grouped under **utils system upgrade**.

utils system upgrade initate: 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.

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.

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.

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.

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.

Certificates for Live Data

Certificates and Secure Communications

For secure Cisco Finesse, Cisco Unified Intelligence Center, 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.

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.

Procedure

Step 1   Sign in to Cisco Unified Operating System Administration on the Live Data server using the following URL: http://*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: http://*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:

Procedure

Step 1   Sign in to Cisco Unified Operating System Administration on the Finesse server using the following URL:

http://*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 .