VPN-less Access to Finesse Desktop

This feature provides the flexibility for agents and supervisors to access the Finesse desktop from anywhere through the Internet without requiring VPN connectivity to the data center. To enable this feature, a reverse-proxy pair must be deployed in the DMZ.

Cisco Unified CCX supports Historical and Real Time report gadgets in agent and supervisor desktops in VPN-less deployments. The reverse-proxy configuration enables authentication of all requests at the proxy, along with other security enhancements as detailed in the Reverse-Proxy selection and configurations, on page 10 section.

Media access remains unchanged in reverse-proxy deployments. To connect to the media, agents and supervisors can use Cisco Jabber or Webex over Mobile and Remote Access solution (MRA). They can also enable the Extend and Connect feature in this deployment.

When deployed with VPN-less reverse-proxy, Customer Collaboration Platform can be deployed within the DMZ or can be moved within the enterprise.

If you have already deployed a reverse-proxy and want to access the Finesse desktop without connecting to VPN, refer to the VPN-less Finesse configurations section. Otherwise, refer to the Reverse-Proxy selection and configurations section.

Note For deployments at multiple sites, configure the reverse-proxy close to the Unified CCX cluster to reduce latencies and WAN bandwidth consumption.

For OpenResty Nginx-based reverse-proxy rules, installation, configuration, and security hardening instructions, refer to Reverse-Proxy Configuration. Any reverse-proxy supporting the required criteria (as mentioned in Reverse proxy selection criteria, on page 10) can be used in place of OpenResty Nginx for supporting this feature.

For the list of caveats, see the Caveats, on page 17section.

Prerequisites

To configure a VPN-less access to the Finesse desktop:

- Unified CCX must be 12.5(1) SU2 and above.
- Customer Collaboration Platform must be 12.5(1) SU2 and above.
- DMZ with internet connectivity must be available to host the reverse-proxy

VPN-less Finesse configurations

To configure VPN-less access to Finesse desktop, the Contact Center administrators and the network administrators must work in tandem.

Note Don't allow access to the reverse-proxy in your external firewall until all security configurations are in place. To test your changes, use a host that isn’t publicly accessible.

The configuration steps are as follows:

1. Populate Network Translation Data
2. Host the Mapping File
3. Add Proxy IP by Using CLI
4. Configure Reverse-Proxy Host Verification
5. Configure Proxy Mapping by Using CLI
6. Configure CORS and Frame-Ancestors
7. Configure SSO

Populate Network Translation Data

The proxy-config map file is similar to a plain property file in which the values are separated by the equal sign. Left Hand Side (LHS) contains the host and port of Unified CCX and Customer Collaboration Platform. Right Hand Side (RHS) contains the values of the host and port that are exposed via reverse-proxy to access the Finesse desktop.

The network administrator and Unified CCX administrator should create a proxy-config map file that has the mapping for all the default ports of the Cisco components, to which external traffic from the Internet clients have to be redirected. For example, 443 port of Customer Collaboration Platform.

The proxy-config map file must be hosted on a web server that is accessible by the Unified CCX and Customer Collaboration Platform servers. The following list is an example of systems and hosts that are required for a two-node Unified CCX cluster with one Customer Collaboration Platform node using SSO mode:

- Proxy Node1 = ccxproxy1.xyz.com

- Proxy Node2 = ccxproxy2.xyz.com
- Publisher = ccxnode1.internal.com
- Subscriber = ccxnode2.internal.com
- CCP = ccp.internal.com

The following is an example of a mapping file that contains the entries required for a two-node Unified CCX cluster with one Customer Collaboration Platform node using non-SSO mode.

```
ccxnode1.internal.com:443 = ccxproxy1.xyz.com:443
ccxnode2.internal.com:443 = ccxproxy2.xyz.com:443
ccp.internal.com:443 =  ccxproxy1.xyz.com:8442
ccp.internal.com:7443 = ccxproxy1.xyz.com:7442
```

Figure 1: Hostname Mapping Example

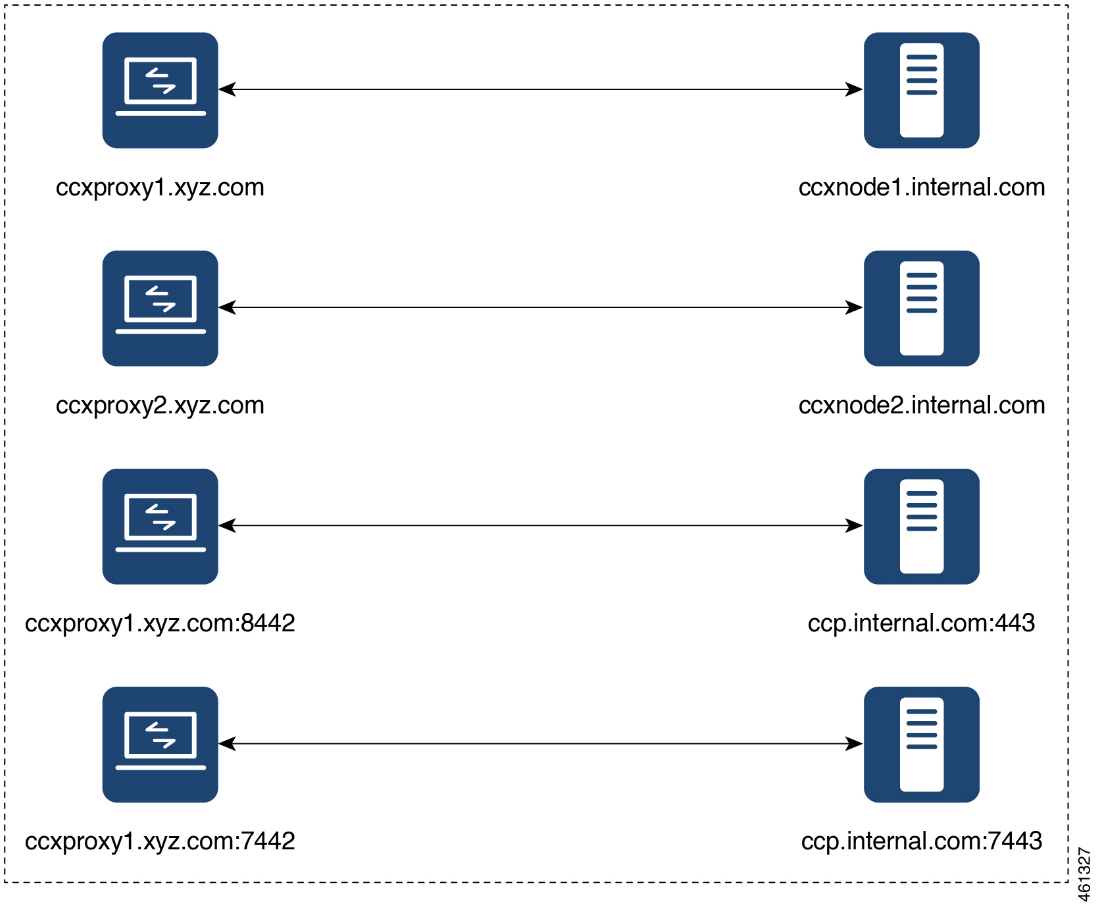
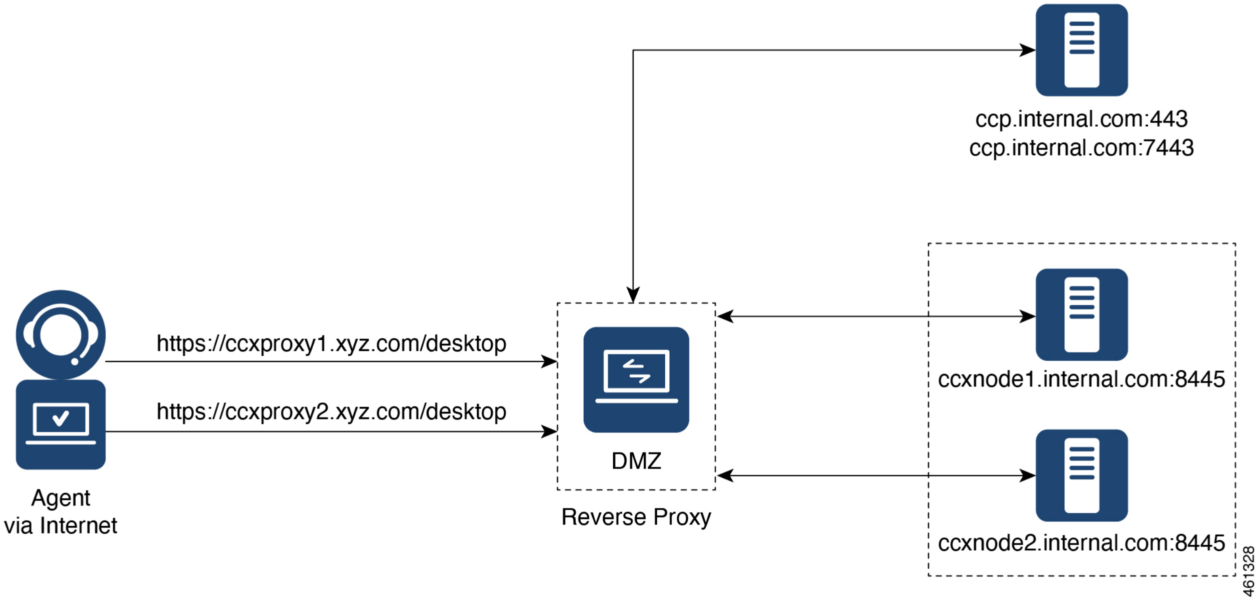


Figure 2: Network Architecture Example


ccp.internal.com:443
ccp.internal.com:7443
https://ccxproxy1.xyz.com/desktop
https://ccxproxy2.xyz.com/desktop
DMZ
Reverse Proxy
ccxnode1.internal.com:8445
ccxnode2.internal.com:8445
Agent
via Internet
461328

Host the Mapping File

The mapping file that is created in the *Populate Network Translation Data* section, is used by the solution components (Unified CCX and Customer Collaboration Platform) servers to modify their responses, to enable clients to access the solution via the reverse-proxy. This requires the file to be hosted on any web server accessible by the component servers. The reverse-proxy server, Unified CCX server, or any web server configured by the administrator can be used for this purpose.

To access the mapping file, the host server's SSL certificate must be uploaded (using Cisco Unified OS Administration in Unified CCX server) to the individual nodes of the services. After uploading the file, verify if the URL is accessible from Unified CCX and Customer Collaboration Platform servers. For example, *https://proxyserver.xyz.com:10000/proxymap.txt*. HTTP-based URLs are allowed for hosting the mapping file through HTTPS, which is the recommended access scheme.

Add Proxy IP by Using CLI

The administrator must use CLI to add the list of trusted reverse-proxy IP addresses and their corresponding hostnames. This must be done on all the nodes of Unified CCX and Customer Collaboration Platform. These components consider only requests from the configured hosts or IP addresses as valid.

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

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

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

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

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

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

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

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

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

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

Source proxy.xyz.com successfully added list

The following source(s) are configured:

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

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

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

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

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

Delete operation successful
```

Configure Reverse-Proxy Host Verification

You can configure SSL certificate verification for communication between reverse-proxy host and the Cisco Web Proxy Service by running the following CLI command on both publisher and subscriber nodes of Finesse:

utils system reverse-proxy client-auth

This command has the following parameters:

- enable
- disable
- status

By default, the host authentication is enabled.

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

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

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

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

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

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

Note After enabling the reverse-proxy host authentication, browser-based clients that connect to Finesse Desktop via LAN hostname must select a client certificate. A pop-up is displayed on systems where client certificates are installed. Clients can choose any of the certificates listed in the pop-up, and continue to connect to Finesse.

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

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

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

Configure Proxy Mapping by Using CLI

The proxy-config map file can be configured in the Unified CCX and Customer Collaboration Platform servers using the `utils system reverse-proxy config-uri` command. If the URL is configured to use HTTPS protocol, Unified CCX and Customer Collaboration Platform must have the certificate (certificate of the web server hosting the URL) uploaded in */cmplatform*. The administrator can configure a maximum of two URLs. The URL that is added first takes precedence and that URL is polled to detect changes in the mapping file. When the URL is not accessible, the alternate URL is used. The following is an example of the CLI to list the configured proxy-config map URLs:

```
admin:utils system reverse-proxy config-uri list

Currently no source is configured
```

The following is an example of the CLI to configure the proxy-config map URL on the Unified CCX and Customer Collaboration Platform servers:

```
admin:utils system reverse-proxy config-uri add https://saproxy.xyz.com:10000/proxymap.txt

Source https://saproxy.xyz.com:10000/proxymap.txt successfully added

admin:utils system reverse-proxy config-uri list

The following source(s) are configured:

1. https://saproxy.cisco.com:10000/proxymap.txt
```

The following is an example of the CLI to delete existing proxy-config map URLs. This command lists all the configured proxy-config URLs and gets user input to delete specific or all proxy-config URLs:

```
admin:utils system reverse-proxy config-uri delete
Select the reverse-proxy source URI to delete:

 1) https://saproxy.xyz.com:10000/proxymap.txt
 2) all
 q) quit

Please select an option (1 - 2 or "q" ): 1

Delete operation successful
```

The following is an example of the CLI to set the proxy-config update frequency (in minutes). Based on the set frequency, the local file system of Unified CCX and Customer Collaboration Platform are updated with the content from the proxy-config map file. Before configuring the URL, this command does not return any value. After configuring the proxy-config map URL, by default it returns one minute as the value.

```
admin:utils system reverse-proxy show-config-update-frequency
No config-uri configured

admin:utils system reverse-proxy config-uri add https://saproxy.xyz.com:10000/proxymap.txt

Source https://saproxy.xyz.com:10000/proxymap.txt successfully added

admin:utils system reverse-proxy show-config-update-frequency
1 minute

admin:utils system reverse-proxy set-config-update-frequency 5

admin:utils system reverse-proxy show-config-update-frequency
5 minutes
```

Configure CORS and Frame-Ancestors

Add both the primary and secondary reverse-proxy origins on publisher and subscriber nodes of Unified CCX. If you change Cross-Origin Resource Sharing (CORS) allowed list and frame-ancestors, you must restart Finesse Notification and Tomcat services. For information about restarting Finesse notification service, see the *Cisco Finesse Services* section in *Cisco Finesse Administration Guide*.

- Administrators must add the list of proxy server origins on the allowed list of CORS origins, if the CORS setting is enabled on Unified CCX and Customer Collaboration Platform.

  Set the reverse proxy URL in the `utils cuic cors allowed_orgins add <URL>` command as the allowed list of CORS origins. For example:

  ```
  utils cuic cors allowed_orgins add https://saproxy.xyz.com:8445
  ```

Note Run this command on both the Publisher and Subscriber Unified CCX nodes.

- Frame-ancestors are added automatically while adding the reverse-proxy trusted hosts in Unified CCX servers.

- Administrators must delete the corresponding allowed list of CORS and frame-ancestors entries while deleting the trusted hosts of a reverse-proxy.

Caution If you do not delete the corresponding CORS and frame-ancestors entries, it becomes a security vulnerability.

Note CORS and frame-ancestors are not applicable to IdS.

For information about deleting CORS see the *Cross-Origin Resource Sharing (CORS)* section in the *Cisco Finesse Administration Guide*.

For more information about configuring CORS, see the Live Data CORS Configuration section in Cisco Unified Contact Center Enterprise Installation and Upgrade Guide.

For information about deleting frame-ancestors see the *Supported Content Security Policy Directives* section in the *Cisco Finesse Administration Guide*.

Import of Reverse-Proxy Certificates

Ensure the certificates from both the OpenResty Nginx reverse-proxies are imported from the Publisher nodes.

To import the certificates, do the following:

1. In the **Cisco Unified OS Administration** interface, select **Security** > **Certificate Management** > **Upload Certificate/Certificate chain**.
2. Upload the certificate.
   a. From the **Certificate Purpose** drop-down list, select **tomcat-trust**.
   b. In the **Upload File** field, click **Browse** and select the certificate file.
   c. Click **Upload File**.
3. Run the `utils system restart` command to restart both the Unified CCX nodes in the cluster.

Configure SSO

If SSO is enabled , SSO must be configured for VPN-less access. Otherwise, agents and supervisors can't login to the Cisco Finesse desktop.

The steps to configure SSO are as follows:

1. Administrator must download proxy specific SAML SP metadata from IdS administration interface.
2. Add proxy relying party trust with IdP.
3. Add proxy redirect URIs to Finesse clients manually via IdS admin interface.
4. Validate SSO configuration for reverse-proxy from IdS admin

For more information, see the *Single-Sign On* chapter in *Cisco Unified Contact Center Express Features Guide*.

Note

- Proxy configuration does not reflect in IdS in any one of the following scenarios:
  - IdP metadata is not uploaded
  - IdS is in maintenance mode
  - Maintenance mode is completed.
- If proxy configuration is changed for IdS hosts, administrator must reestablish trust on IdP for new IdS proxy hosts after downloading new metadata file from IdS admin. Administrator must reestablish **Relying Party Trusts** with IdP. For more information, refer to the Integrate Cisco IdS with AD FS *Configure the Cisco Identity Service* section in the *Cisco Unified Contact Center Express Features Guide*
- If proxy configuration is changed for Cisco Finesse hosts, administrator must manually update the allowed Finesse client redirect URIs list on IdS admin interface. For more information, refer to theConfigure the Cisco Identity Service*Configure the Cisco Identity Service* section in the *Cisco Unified Contact Center Express Features Guide*.. Client name is "Finesse" and the URLs that are to be added are as follows:
  - `https://<finesseReverseProxySideAHost:finesseReverseProxySideAPort>/desktop/sso/authcode`
  - `https://<finesseReverseProxySideBHost:finesseReverseProxySideBPort>/desktop/sso/authcode`
- If SAML certificate is regenerated, the SAML certificate must be updated for corresponding **Relying Party Trusts** in IdP. Configure the same port used to access IdP via LAN to access IdP via proxy. For more information, refer to the *Hostname or IP Address Change* section in the *Cisco Unified Contact Center Express Features Guide*.

Serviceability

Monitor Connected Agents and Supervisors

The reverse-proxy has to be monitored by using the proxy-specific features. For more information, refer to the specific reverse-proxy documentation.

Cisco Finesse allows administrators to view the list of currently connected agents and supervisors. The administrator can filter and see the agents and supervisors who are connected to the Finesse desktop based on the connection type. For example, agents and supervisors connected through the Contact Center network and those connected through reverse-proxy can be seen. For more information, see the *Connected Agents* section in *Cisco Finesse Administration Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/finesse/products-maintenance-guides-list.html. Administrators can also view the summary of connected users by using the following CLI command:

```
admin:utils finesse show_connected_users summary

Total Connected Users: 6

Desktop Users: 1
```

```
FIPPA Users: 2
Third-party Users: 3

Users connected to Finesse via LAN/WAN: 5
Users connected to Finesse via Proxy: 1

To view the complete list of signed-in users, log in to the Cisco Finesse
Administration Console, and navigate to the Connected Agents tab.
```

To view the real-time list of connected users by using an API, see the *ConnectedUsersInfo* section in *Cisco Finesse Web Services Developer Guide* at https://developer.cisco.com/docs/finesse/#!rest-api-dev-guide.

API Modifications to Support Reverse-Proxy Deployments

Finesse SystemInfo API

SystemInfo API is now secured when it is accessed through a reverse-proxy. The API is accessible with agent and supervisor credentials. The following field has been added to support this feature:

- **httpsPort:** HTTPS port has to be used for all Finesse API and desktop notifications.

For more information, see the *SystemInfo* and *ConnectedUsersInfo* sections in *Cisco Finesse Web Services Developer Guide* at https://developer.cisco.com/docs/finesse/#!rest-api-dev-guide.

Reverse-Proxy selection and configurations

Reverse proxy selection criteria

Contact Center administrators must select an appropriate reverse-proxy. Any reverse-proxy that meets the following minimum requirements can be used:

- Supports HTTP2/TLS 1.2 and secure Websockets.
- Has proper logging mechanism for easy debugging of issues
- Supports multiple Unified CCX and Customer Collaboration Platform servers from a single reverse-proxy.
- Supports periodic revalidation of cached content. This is required because any updates or installations on the internal hosts don't require a manual intervention to clear the cached content of the proxy.
- Supports custom authentications or provides alternative mechanisms such as an enterprise login to prevent unauthenticated access of solution components.

Note When you use Cisco-provided reverse-proxy configuration, the requests are authenticated at the proxy before they are forwarded to the upstream servers. When you are configuring a custom reverse-proxy, you must create this authentication layer if they have to be as secure as the Cisco provided configuration. You should consider this configuration step while planning to implement VPN-less access to Finesse using a custom reverse-proxy.

- Enables caching of static resources with support for cache-control header to reduce DoS/DDoS attack vectors and to scale the proxy. Any proxy that needs to support more than a few hundred users and does not provide response caching features should be deployed with a Content Delivery Network (CDN) with support for cache-control headers so that load and security guidelines are met.

Note CDN deployment is also recommended with caching proxies such as OpenResty® Nginx to eliminate the impact of DDoS attacks.

- Supports X-Forwarded headers. These headers are used by the solution to decide how to handle a request.

Additional Requirements

Some desirable requirements in a reverse-proxy are as follows:

- Consider deploying proxies that are built on non-blocking IO-based technology instead of the traditional thread-per-request architecture, to scale better.
- Consider proxies that provide response substitution capabilities which allow workarounds for custom gadgets as custom gadgets may not work with reverse-proxy directly.

Note Finesse Desktop Chat over reverse-proxy requires response substitution capability.

- Support for port-based forwarding can be used to reduce the cost of deployment by avoiding the need for multiple externally resolvable hostnames, public DNS records, and corresponding certificates for each internal server that has to be accessed.
- Support for custom plugin/modules, which can be used to enhance the authentication model and provide a more robust security posture.

Performance and hardware recommendation

For details, see Performance and Hardware Recommendations.

Configure Reverse-Proxy

Install the host OS and reverse-proxy of your choice. Consider the following points while configuring the reverse-proxy:

- Configure SSL certificates as required.
- Refer to the specific proxy documentation and configure the proxy rules for each service with the same host and port that is configured in the mapping file.
- IdS and IdP trust should be configured before proxy mapping configuration is done. Otherwise, proxy configuration changes will not be processed by IdS.
- For IdS hosts, if proxy configuration is changed, the administrator must re-establish trust on IdP for new IdS proxy hosts after downloading new metadata file from IdS admin.

- For Finesse hosts, if proxy configuration is changed, the administrator must manually add or update the allowed Finesse client redirect URIs from IdS administration interface.
- Whenever SAML certificate is regenerated or IdP metadata is uploaded, proxy configurations are generated afresh.

To secure the reverse-proxy, refer to the *Security Guidelines* section in the *Solution Design Guide for Cisco Unified Contact Center Express* available at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-express/products-implementation-design-guides-list.html.

Determine Scale and Hardware for Proxy

Contact Center administrators should analyze the hardware required for the reverse-proxy, based on the number of agents and supervisors who may access the Finesse desktop without connecting to VPN. You can use the reference request rates provided for Unified CCX and Customer Collaboration Platform in Reverse-Proxy Configuration.

The type of proxy selected guides the hardware to be used, depending on whether the proxy is shipped as an installable software or is a hardware-based application.

Sizing configurations are pre-tested for OpenResty® Nginx proxy. Custom proxy deployments should consult their product documentation or run basic scaling tests to determine the rates that can be supported by the respective proxy and scale their hardware accordingly.

Hardware Recommendations

A standard Unified CCX can be supported by OpenResty Nginx 1.19 running on a CentOS 7.4.1708 distribution, with the configurations and settings (mentioned in the Installing Nginx site) on a dual core 2 CPU (4 logical CPU) Intel Xeon CPU E5-2690 v2 (3.00GHz, 25MB cache) at an average of 10% CPU usage and peak of 15% CPU usage during logins.

A minimum of 8 GB memory is recommended for the proxy server when all other nonessential services and graphical subsystems are disabled.

Note Additional memory has to be configured based on the in-memory cache configuration added to OpenResty Nginx as described in the Cache Configuration section of Reverse-Proxy Configuration.

It is recommended that deployments gradually onboard new solution components to the proxy until the proxy is always left with 50-55% free CPU so that it can cope with unexpected spikes in traffic from the internet.

Determine Gadget Compatibility

Determining the gadget compatibility is an important activity for planning a VPN-less Finesse deployment.

After deploying the reverse-proxy, all Cisco-provided gadgets (Unified CCX and Customer Collaboration Platform) work seamlessly with their respective servers of Release 12.5(1) SU2 or later. The Webex Experience Management and CCAI gadgets also work seamlessly with VPN-less Finesse deployments.

In some scenarios, depending on the gadget design, custom third-party gadgets require workarounds to enable them to work with the reverse-proxy deployment. Refer to the following sections to determine if any of your gadgets require workarounds.

Note

- Gadgets that are loaded from servers other than Finesse server should use **exclude-url** feature in the gadget XML specification to load the Finesse resources such as Finesse.js. For more information, refer to the **Use Gadget URI Exclude Feature to Refer to Finesse Resources** section.
- If you use two different URLs, one internal and one external, in Enterprise Chat and Email (ECE), you must update the Finesse desktop layout to use only the external URL. If you use an internal-only ECE (for integrations that support only ECE email routing), you must change the ECE web server to ensure that the ECE services are accessible externally.

Gadget Types and VPN-less Compatibility

Finesse gadgets are classified into the following types based on how they are designed operationally:

- Gadgets that are self-contained within the desktop. These gadgets do not have to make any additional network requests, or are restricted to invoking Finesse APIs and APIs on the internet.
- Gadgets that provide their functionality by communicating with an accompanying server that is deployed in the DMZ and is reachable directly from the internet and LAN.

Note To enable the same desktop layout to be used by both LAN-based and internet-based clients, the server installed in a DMZ should also be reachable from servers such as Finesse in LAN, and from clients that are running within the LAN.

- Gadgets that need to communicate with an accompanying server deployed in LAN, but uses desktop-provided **makeRequest** API to communicate to the server. The **makeRequest** API routes all the requests through the Finesse server and does not directly reach the server that is deployed in the LAN.

Note These requests succeed in a reverse-proxy deployment only if the requests are made using the hostname and port. The hostname and the port must be reachable from LAN because the requests are run by Finesse server which runs on LAN.

- Gadgets that have to communicate directly with any one of the following types of accompanying server:
  - Server deployed within the LAN and is not reachable directly from the internet.
  - Server that communicates with an additional port apart from the HTTP port used to load the gadget.

The last two types of gadgets have to be modified to be used in a reverse-proxy deployment. The steps required to enable these gadgets to be accessed from internet clients are as follows:

- Enable VPN-less access for custom gadgets

- Send hostname and port information to gadgets
- Use gadget's **URI Exclude** feature to refer to Finesse resources

Enable VPN-less Access for Custom Gadgets

Gadgets that communicate directly with accompanying servers that are deployed in LAN must handle the following aspects to work correctly in a reverse-proxy deployment:

- Use the right hostname and port for communicating with its accompanying server.

  A gadget can find the correct hostname and port corresponding to the server from which the gadget was loaded, by using the **gadgets.util.getUrlParameters().up_urlPrefs** API provided by the Finesse Javascript API.

  To find additional ports or hostnames that are required, data can be passed in as gadget preference such that the additional host and port information can be sent to the gadget. For more information, refer to the **Send Hostname and Port Information to Gadgets** section.

- Ensure that the communications are forwarded correctly by the reverse-proxy.

  After the gadget starts communicating with the correct host and port information, the hostname and port number have to be forwarded to the server deployed in the LAN. This can be done by opening the appropriate ports in the DMZ firewall. Also, ensure that the appropriate ports and rules are added to the reverse-proxy rules to forward the traffic to the correct server in the LAN.

- **Best Practice:** If requests to external servers are made using Finesse authentication headers, a common validation is enabled to authenticate the requests at the proxy. Gadgets that do not use Finesse authentication should plan to implement their own custom authentication schemes to ensure that the requests are validated at the proxy before sending to the Finesse server.

Send Hostname and Port Information to Gadgets

Gadgets that send host and port information corresponding to a server deployed within the LAN can use the **UserPreferences** feature supported by Finessse gadgets. This feature allows a configurable, named information to be passed to the gadget. The information can be referenced within the gadget XML or programmatically by using a Javascript.

For more information on how to use **UserPreferences** method, refer to https://developer.cisco.com/docs/finesse/#!gadget-preferences.

The **UserPreferences** that are created for this purpose should start with the keyword *externalServerHostAndPort* in its name. This enables Finesse to substitute the host and port that are provided with the corresponding entry from the **proxyMap** file. For example:

```
<UserPref name="externalServerHostAndPort_chat” display_name="Chat_externalServerHostAndPort"
default_value="SMHostName:7443" datatype ="hidden"/>
```

Note The `default_value` parameter is not case sensitive.

When accessed from the LAN, the **UserPreferences** continues to have the default value that is configured in the XML. However, when accessed through the reverse-proxy, the **UserPreferences** receives the value from the **proxyMap** file. For example:

```
SMHostName:7443=external-proxy-host:4043
```

When accessed through the reverse-proxy, the gadget receives the port **4043** and host name as **external-proxy-host**.

Use Gadget URI Exclude Feature to Refer to Finesse Resources

Add the following content within the `ModulePrefs` tag of the gadget XML to ensure that the resources that are loaded from Finesse server are excluded from concatenation. This step is mandatory for gadgets that load their XML from custom servers.

```
<Optional feature="content-rewrite">
<!-- these files will be directly served by Finesse, not through shindig -->
<Param name="exclude-url">finesse.min.js</Param></Optional>
```

Host Header Configuration

The following are the mandatory HTTP headers that reverse-proxy has to set along with the actual headers set by the client before forwarding the headers to the Finesse server.

Table 1:

Header	Description
X-Client-IP	The reverse-proxy should populate this custom header as the client's IP address before forwarding it to the Finesse server. This is used to log the client's IP in the Finesse server.
Host	The Host request header specifies the host and port number of the server to which the request is being sent. If no port is included, the default port for the service requested (for example, 443 for an HTTPS URL and 80 for an HTTP URL) is used. An HTTP/1.1 proxy ensures that any request message it forwards contains an appropriate Host header field to identify the service being requested by the proxy. This value is used by Finesse to find if the request is sent via the allowed list of proxies configured in Finesse. The hostname and port value of the reverse-proxy should be set. Otherwise, the Finesse validation fails and returns HTTP 400 Error.

Header	Description
X-Forwarded-For	The `X-Forwarded-For` (XFF) header is used for identifying the originating IP address of a client connecting to a web server through an HTTP proxy or a load balancer. The IP of the reverse-proxy has to be appended or set. Finesse uses this header to find if the request is from the allowed list of reverse-proxies. When the request is forwarded through multiple reverse-proxies, the values of all reverse-proxies are appended to the rightmost value of this header.
X-Forwarded-Port	The reverse-proxy should set the listening port on this header. Finesse server receives all the requests internally via 8445 port. This header value helps Finesse to set the valid configuration.

The following are the standard headers manipulated by the proxy:

Table 2:

Header	Description
Connection	Any Connection value in the HTTP header that is set by the client should be cleared and forwarded to the Finesse server. This has to be done so that the Finesse server decides the connection management and not the Finesse client. This prevents security outages.
Accept-Encoding	The reverse-proxy clears the Accept-Encoding header to have better control over compression aspects of the response.

Finesse URL

Agents and supervisors should bookmark two different pairs of URLs (publisher and subscriber) for accessing the Finesse desktop through both the Contact Center network and the reverse-proxy.

Historical and Real Time Gadgets

Cisco Unified CCX supports Historical and Real Time report gadgets in agent and supervisor desktops in VPN-less deployments. To configure the Historical and Real Time report gadgets, refer to the *Configure Historical Report Gadgets in Cisco Finesse* section in *Cisco Unified Intelligence Center User Guide* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-intelligence-center/products-user-guide-list.html.

Note

- Stock reports and custom reports can be viewed in VPN-less supervisor desktop. However, before viewing the custom reports as gadgets in VPN-less supervisor desktop, run the command, **set cuic properties allow-proxy-custom-report on**.
- To configure the data set size for Historical report, run the command, **set cuic properties vpnless-response-size-ht**. By default, the data set size for HT is set to 8MB.
- To configure the data set size for Real Time report, run the command, **set cuic properties vpnless-response-size-rt**. By default, the data set size for RT is set to 300KB.

  If the data set size is more than the configured value, the gadget will display the following error message:

  ```
  Failed to load the gadget. Response size is more than allowed limit. Please contact
  your Administrator.
  ```

  This limitation is applicable on VPN-less deployments only. For more information about configuring the data set size, see *set Cisco Unified Intelligence Center properties* section in *Administration Console User Guide for Cisco Unified Intelligence Center* at https://www.cisco.com/c/en/us/support/customer-collaboration/unified-intelligence-center/products-maintenance-guides-list.html.

Security Guidelines

For information about security guidelines, see the *Security Guidelines for Reverse-Proxy* in *Security Guide for Cisco Unified ICM/Contact Center Enterprise* at

https://www.cisco.com/c/en/us/support/customer-collaboration/unified-contact-center-enterprise/products-installation-and-configuration-guides-list.html.

Caveats

Reverse-proxy deployment allows agents and supervisors to concurrently access the Cisco Finesse desktop from both LAN and through reverse-proxy. After configuring the reverse-proxy, when the agents and supervisors access the Finesse desktop through LAN, all the features work seamlessly. However, when the Finesse desktop is accessed through the reverse-proxy, the caveats are as follows:

- Finesse IP Phone Agent (FIPPA) isn’t supported.
- Administrative applications and the corresponding APIs of Finesse, IdS, and Cisco Unified Intelligence Center aren’t supported.
- Multiple devices accessing the Finesse desktop through Network Address Translation (NAT) isn’t supported.
- Multiple users accessing the VPN-less desktop from behind a common proxy isn’t supported when multiple sites are involved.
- If threshold images are used in Live Data, Real Time, and Historical gadgets, add the reverse-proxy rules to allow images to be accessed through reverse-proxy. For more information on threshold images rules, refer to the section.

- **Finesse API Compatibility:**
  - Finesse Desktop supports only the WebSocket notification mechanism over reverse-proxy. For third-party servers, BOSH or XMPP over TCP communication through reverse-proxy isn’t supported.
  - When the SystemInfo API is accessed through a reverse-proxy, the authorization headers are required.