This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Performance Management Advisors Deployment Guide

Deploy and Configure Apache

5/1/2025

Contents

Deploy and Configure Apache

Use the information on this page to configure an Apache Web Server instance to direct HTTP requests to the appropriate server within your Advisors deployment.

Genesys strongly recommends that you configure Apache to accept HTTP over SSL (HTTPS) connections. The streaming protocols used by Advisors are not required to be encrypted, but this is a more secure form of communication, and helps prevent possible interference from legacy virus scanners, firewalls, proxies, and so on, which don't properly support streaming protocols and might attempt to buffer unencrypted traffic. Using HTTPS connections helps to ensure both the security and reliability of the connection to the Advisors server. Requests between Apache and the Tomcat server running Advisors can also use HTTPS connections if needed.

Configure Apache Modules

The recommended Advisors Apache configuration requires the following modules:

- ssl
- headers
- proxy
- proxy_ajp
- proxy_http
- proxy_wstunnel

To enable Apache modules, edit the relevant file or use the relevant configuration tools for your environment. In many installations, this will involve editing your `httpd.conf` file. For more information on the files used to configure Apache, see the Apache documentation describing the files used to configure Apache.

For example, to use the SSL module, uncomment that line:

Uncomment this line:

```
#LoadModule ssl_module modules/mod_ssl.so
```

It now looks like this:

```
LoadModule ssl_module modules/mod_ssl.so
```

Configure HTTPS

To configure Apache to support HTTPS:

- Obtain or generate the SSL security certificate and private key.
- Configure Apache to use your certificate.

Obtaining a Certificate

An SSL certificate signing request (CSR) can be generated and submitted to a certificate authority using OpenSSL or a similar tool. You can then issue a certificate if you are your own certificate authority, or a certificate can be issued by a third-party certificate authority.

The OpenSSL `req` command can be used to generate the request, or to generate a self-signed certificate in a single step. For more information, see the OpenSSL documentation.

Configure Apache to use your Certificate

In general, to configure Apache to use your certificate, add the following configuration to the Apache virtual host that is used for Advisors, and for the port on which HTTPS connections are accepted (the default HTTPS port is 443):

```
SSLEngine on
SSLCertificateFile      /path/to/your/certificate.pem
SSLCertificateKeyFile /path/to/your/certificate.key
```

For example, to configure the certificate globally in Apache, use the following configuration:

```
<VirtualHost *:443>
SSLEngine on
SSLCertificateFile      /path/to/your/certificate.pem
SSLCertificateKeyFile /path/to/your/certificate.key
</VirtualHost>
```

For more information about virtual hosts, see the Apache virtual host documentation.

Configure Routing for Advisors Components

Advisors components can be distributed across many servers. The Apache configuration enables proper routing of requests to these components. In some cases, there might be multiple installations of the same component. In these cases, requests can be load-balanced to different Apache servers, each one directing these requests to different servers.

Each Advisors component routing entry in the Apache configuration directs its request to *<hostname>* (see the configuration example below). In your configuration, change *<hostname>* to the host name of the server on which the component is installed. This will vary depending on your particular installation. Note that requests are matched to the first ProxyPass entry in the order in which they are listed within the Apache configuration, so Genesys recommends that you add the routing information in the same order that is outlined in the following configuration example.

Template Configuration Example

In the following template configuration example, the text might wrap to multiple lines, but each ProxyPass directive must be on a single line in the Apache configuration.

Also take care to use the appropriate port for the URL and its protocol being proxied. In this example, requests to Tomcat over AJP use port 8009 while websocket communication uses HTTP port 8080. The specific ports used might vary depending on your Tomcat configuration, if modified from the default.

If you have Pulse Advisors release 9.0.003.09 or higher installed, also see Configuring Tomcat AJP connector.

```
#Route to resource management console
ProxyPass /rmc/ ajp://<hostname>:8009/rmc/

#Route to CCAdv accessibility web services
ProxyPass /ca-xml/ ajp://<hostname>:8009/ca-xml/

#Route to Workforce accessibiltiy web services
ProxyPass /wu/ ajp://<hostname>:8009/wu/

#Route to Advisors metric graphing
ProxyPass /ea-ws/ ajp://<hostname>:8009/ea-ws/
ProxyPass /dashboard/ ajp://<hostname>:8009/dashboard/

#Route to Advisors administration module
ProxyPass /admin ajp://<hostname>:8009/admin

#Route to FA server
ProxyPass /fa/com.informiam.fa.admin.gwt.AdminConsole/ ajp://<hostname>:8009/fa/
com.informiam.fa.admin.gwt.AdminConsole/ timeout=86400
ProxyPass /fa/ ajp://<hostname>:8009/fa/

#Route to Advisors web services
ProxyPass /adv/websocket/wsconnection/info ajp://<hostname>:8009/adv/websocket/wsconnection/
info
ProxyPassMatch /adv/websocket/wsconnection/(.*)/(.*)/websocket ws://<hostname>:8080/adv/
websocket/wsconnection/$1/$2/websocket
ProxyPass /adv/ ajp://<hostname>:8009/adv/

#Route to Advisors platform installation
ProxyPass /base-ws/ ajp://<hostname>:8009/base-ws/
ProxyPass /nav-service/ ajp://<hostname>:8009/nav-service/
ProxyPass /prefs-service/ ajp://<hostname>:8009/prefs-service/
ProxyPass / ajp://<hostname>:8009/
```

Routing With HTTPS Connections From Apache

In addition to configuring HTTPS connections for incoming requests to Apache, as described in the Configure HTTPS section, you can also configure an HTTPS connection between Apache and Tomcat. To do this, you must update the following information in the ProxyPass entries that route requests:

- Use the HTTPS and WSS protocols, which replace AJP and WS.
- Specify the port. By default, the Tomcat HTTPS connector is configured to use port 8443.

The Tomcat HTTPS connector can be used on port 8443 without any additional configuration. If you will not be using the default configuration, see the documentation that describes how to customize the configuration of the Tomcat HTTPS connector.

[+] Example HTTPS Routing Configuration

```
#Route to resource management console
ProxyPass /rmc/ https://<hostname>:8443/rmc/

#Route to CCAdv accessibility web services
ProxyPass /ca-xml/ https://<hostname>:8443/ca-xml/

#Route to Workforce accessibiltiy web services
ProxyPass /wu/ https://<hostname>:8443/wu/

#Route to Advisors metric graphing
ProxyPass /ea-ws/ https://<hostname>:8443/ea-ws/
ProxyPass /dashboard/ https://<hostname>:8443/dashboard/

#Route to Advisors administration module
ProxyPass /admin https://<hostname>:8443/admin

#Route to FA server
ProxyPass /fa/com.informiam.fa.admin.gwt.AdminConsole/ https://<hostname>:8443/fa/com.informiam.fa.admin.gwt.AdminConsole/ timeout=86400
ProxyPass /fa/ https://<hostname>:8443/fa/

#Route to Advisors web services
ProxyPass /adv/websocket/wsconnection/info https://<hostname>:8443/adv/websocket/wsconnection/info
ProxyPassMatch /adv/websocket/wsconnection/(.*)/(.*)/websocket wss://<hostname>:8443/adv/websocket/wsconnection/$1/$2/websocket
ProxyPass /adv/ https://<hostname>:8443/adv/

#Route to Advisors platform installation
ProxyPass /base-ws/ https://<hostname>:8443/base-ws/
ProxyPass /nav-service/ https://<hostname>:8443/nav-service/
ProxyPass /prefs-service/ https://<hostname>:8443/prefs-service/
ProxyPass / https://<hostname>:8443/
```

Configuring Tomcat AJP connector

Applicable only to release 9.0.003.09 and higher.

Starting with release 9.0.003.09, Advisors Platform includes Tomcat server version 9.0.33 to address a potential security vulnerability with AJP requests. For details, see Injection and potential Remote Code Execution (CVE-2020-1938). As a result of this upgrade, you need to do the following Pulse Advisors configuration:

- On the Platform installation of the Advisors web services, edit the `apache-tomcat-9.0.12/conf/server.xml` file to add the following attribute in the AJP connector section: `secret="`*`your_secret`*`"`. The value of `secret` is a validation key between Apache and Tomcat. For example:

```
<Connector port="${AJPPort}" protocol="AJP/1.3"

      connectionTimeout="${WebConnectorConnTimeout}"

      maxThreads="${MaxThreadPoolSize}" maxConnections="${MaxThreadPoolSize}"
minSpareThreads="${MinThreadPoolSize}"

      redirectPort="${HTTPSPort}" relaxedQueryChars="[,]" secret="your_secret"
address="TOMCAT_HOST_IP"/>
```

- If the Apache HTTP Server and Tomcat are on different hosts, then you must add an attribute of `address="`*`TOMCAT_HOST_IP`*`"` to the AJP connector section as well. See the preceding example. For detailed information, see Tomcat configuration documentation.
- The Apache HTTP Server Proxy configuration needs to be updated to pass the configured secret in each of the ProxyPass directives. For example, here is an updated Proxypass directive (do not enclose the `secret` string in quotation marks):

```
ProxyPass / ajp://<hostname>:8009/ secret=your_secret
```

  The addition of the secret to the ProxyPass directive is supported starting with Apache Web Server 2.4.42.

  For details, see Apache mod_proxy_ajp configuration.