Upgrade

NOTE

Product names changed when ForgeRock became part of Ping Identity. Learn more about the name changes from New names for ForgeRock products.

Java Agent supports the following types of upgrade:

- Drop-in software update:

  Usually, an update from a version of Java Agent to a newer minor version, as defined in Ping Identity Product Support Lifecycle Policy | PingGateway and Agents. For example, update from 2024.9 to 2024.11 can be a drop-in software update.

  Drop-in software updates can introduce additional functionality and fix bugs or security issues. Consider the following restrictions for drop-in software updates:

  - Don’t require any update to the configuration
  - Can’t cause feature regression
  - Can change default or previously configured behavior **only** for bug fixes and security issues
  - Can deprecate **but not remove** existing functionality

- Major upgrade:

  Usually, an upgrade from a version of Java Agent to a newer major version, as defined in Ping Identity Product Support Lifecycle Policy | PingGateway and Agents. For example, upgrade from 2023.3 to 2024.3 is a major upgrade.

  Major upgrades can introduce additional functionality and fix bugs or security issues. Major upgrades do not have the restrictions of drop-in software update. Consider the following features of major upgrades:

  - Can require code or configuration changes
  - Can cause feature regression
  - Can change default or previously configured behavior
  - Can deprecate **and** remove existing functionality

This guide describes how to upgrade a single Java Agent instance. To upgrade sites with multiple Java Agent instances, one by one, stop, upgrade, and then restart each server individually, leaving the service running during the upgrade.

For information about upgrade between supported versions of Java Agent, refer to Ping Identity Product Support Lifecycle Policy | PingGateway and Agents.

Example installation for this guide

Unless otherwise stated, the examples in this guide assume the following installation:

- Java Agent installed on `https://agent.example.com:443/app`.
- AM installed on `https://am.example.com:8443/am`.
- Work in the top-level realm `/`.

If you use a different configuration, substitute in the procedures accordingly.

Drop-in software update

The examples in this section assume that the agent is installed in `/path/to/java_agents/`*`agent_type`*, and the update is from the minor version 2024.9 to the minor version 2024.11.

Tomcat Java Agent software update

1. Read the release notes for information about changes in Java Agent.
2. Download the agent binaries from the BackStage download site and extract them to a temporary directory.

   The example in this section is extracted to `/tmp`, and the .jar files are in `/tmp/tomcat_agent/lib`.
3. Back up the directories for the agent installation and the web application container configuration:
   - In local configuration mode:

     ```
     $ cp -r /path/to/java_agents/tomcat_agent /path/to/backup
     $ cp -r /path/to/tomcat/webapps/agentapp /path/to/backup
     ```
   - In remote configuration mode, back up as described in AM's Maintenance guide.
4. Redirect client traffic away from protected web applications.
5. Stop the web applications where the agent is installed.
6. Locate the following files in the container:

- `agent.jar`
- `jee-agents-sdk-`*`version`*`.jar`

The following example finds `./lib/jee-agents-sdk-2024.9.jar`:

Unix Windows

```
$ cd /opt/container
$ find . -type f -name 'jee-agents-*.jar' -print
./lib/jee-agents-sdk-2024.9.jar
```

7. If `agent.jar` is present in the container, delete it.
8. Replace `jee-agents-sdk-`*`version`*`.jar` with the newer downloaded version. The following example replaces `jee-agents-sdk-2024.9.jar`:

Unix Windows

```
$ cd /opt/container
$ rm -f lib/jee-agents-sdk-2024.9.jar
$ cp /tmp/tomcat_agent/lib/jee-agents-sdk-2024.11.jar lib
```

9. (Optional) Update the .jar files outside the container.

   a. Using the `.amAgentLocator` file, find the directory in which the agent was originally installed:

   Unix Windows

   ```
   $ cd /opt/container
   $ cat .amAgentLocator; echo

   /path/to/java_agents/tomcat_agent
   ```

   b. View the content of the `lib` subdirectory:

   Unix Windows

   ```
   $ cd /path/to/java_agents/tomcat_agent/lib
   $ ls -F

   agent.jar
   ```

```
jee-agents-installtools-2024.9.jar
jee-agents-sdk-2024.9.jar
```

   c. Replace the files with the newer downloaded version:

      Unix Windows

      ```
      $ rm -f *
      $ cp /tmp/java_agents/tomcat_agent/lib/*.jar .
      $ ls -F

      agent.jar
      jee-agents-installtools-2024.11.jar
      jee-agents-sdk-2024.11.jar
      ```

10. Replace the current `agentadmin` file with the newer downloaded version:

    Unix Windows

    ```
    $ cd /path/to/java_agents/tomcat_agent/bin
    $ rm agentadmin
    $ cp /tmp/tomcat_agent/bin/agentadmin .
    ```

11. Start the web applications where the agent is installed.
12. Check that the agent is performing as expected:
    a. Check the correct version of the agent is running:
       - Set the log level to `trace`, as described in Manage logs.
       - In `/path/to/java_agents/`*`agent_type`*`/Agent_`*`n`*`/logs/debug`, search for lines containing the string `X-ForgeRock-Edge-Metadata`. The version number is given in the header.

         For example, the log file can contain the following header: `--header "X-ForgeRock-Edge-Metadata: JPA 2024.11`.

    b. Navigate to a protected page on the website and confirm whether you can access it according to your configuration.
    c. Check logs files for warnings and errors.
13. Allow client traffic to flow to the protected web applications.

JBoss and WildFly Java Agent software update

1. Read the release notes for information about changes in Java Agent.
2. Download the agent binaries from the BackStage download site and extract them to a temporary directory.

   The example in this section is extracted to `/tmp`, and the .jar files are in `/tmp/jboss_agent/lib`.
3. Back up the directories for the agent installation and the web application container configuration:
   - In local configuration mode:

     ```
     $ cp -r /path/to/java_agents/jboss_agent /path/to/backup
     $ cp -r /path/to/jboss/webapps/agentapp /path/to/backup
     ```
   - In remote configuration mode, back up as described in AM's Maintenance guide.
4. Redirect client traffic away from protected web applications.
5. Stop the web applications where the agent is installed.
6. Update the `module.xml` file.

   a. Locate the path to the installation, for example, at `/path/to/jboss/modules/org/forgerock/openam/agent/main/modules/org/forgerock/openam/agent/main`.

   b. If any of the following files are listed, remove the resource for the file:
      - `tyrus-standalone-client-2.1.3.jar`
      - `jee-agents-jboss-common-2024.9.jar`
      - `agent.jar`

   c. Update the resource for `jee-agents-sdk-`*`version`*`.jar` to use the absolute path and the newer downloaded version agent version. For example, change

      ```
      <resource-root path="jee-agents-sdk-2024.9.jar"/>
      ```

      to

      ```
      <resource-root
      path="/path/to/java_agents/jboss_agent/lib/jee-agents-sdk-
      2024.11.jar"/>
      ```
7. Update the .jar files outside the container.

   a. Using the `.amAgentLocator` file, find the directory in which the agent was originally installed:

```
$ cd /opt/container
$ cat .amAgentLocator; echo

/path/to/java_agents/jboss_agent
```

b. View the content of the `lib` subdirectory:

```
$ cd /path/to/java_agents/jboss_agent/lib
$ ls -F

agent.jar
jee-agents-jboss-common-version.jar
jee-agents-sdk-version.jar
tyrus-standalone-client-version.jar
```

c. Replace the files with the newer downloaded version:

```
$ rm -f *
$ cp /tmp/java_agents/jboss_agent/lib/*.jar .
$ ls -F

agent.jar
jee-agents-jboss-common-version.jar
jee-agents-sdk-version.jar
tyrus-standalone-client-version.jar
```

8. Replace the current `agentadmin` file with the newer downloaded version:

```
$ cd /path/to/java_agents/jboss_agent/bin
$ rm agentadmin
$ cp /tmp/jboss_agent/bin/agentadmin .
```

9. Start the web applications where the agent is installed.

10. Check that the agent is performing as expected:

    a. Check the correct version of the agent is running:

    - Set the log level to `trace`, as described in Manage logs.
    - In `/path/to/java_agents/`*`agent_type`*`/Agent_`*`n`*`/logs/debug`, search for lines containing the string `X-ForgeRock-Edge-Metadata`. The version number is given in the header.

      For example, the log file can contain the following header: `--header "X-ForgeRock-Edge-Metadata: JPA 2024.11`.

   b. Navigate to a protected page on the website and confirm whether you can access it according to your configuration.

   c. Check logs files for warnings and errors.

11. Allow client traffic to flow to the protected web applications.

Jetty Java Agent software update

1. Read the release notes for information about changes in Java Agent.
2. Download the agent binaries from the BackStage download site and extract them to a temporary directory.

   The example in this section is extracted to `/tmp`, and the .jar files are in `/tmp/jetty_agent/lib`.
3. Back up the directories for the agent installation and the web application container configuration:
   - In local configuration mode:

     ```
     $ cp -r /path/to/java_agents/jetty_agent /path/to/backup
     $ cp -r /path/to/jetty/webapps/agentapp /path/to/backup
     ```
   - In remote configuration mode, back up as described in AM's Maintenance guide.
4. Redirect client traffic away from protected web applications.
5. Stop the web applications where the agent is installed.
6. Replace the following files with the newer downloaded versions.
   - `agent.jar`
   - `jee-agents-installtools-`*`version`*`.jar`
   - `jee-agents-sdk-`*`version`*`.jar`

   The following example replaces `jee-agents-sdk-2024.9.jar`:

   ```
   $ cd /path/to/java_agents/jetty_agent/lib
   $ rm -f jee-agents-sdk-2024.9.jar
   $ cp /tmp/jetty_agent/lib/jee-agents-sdk-2024.11.jar .
   ```
7. Update the Jetty configuration:

   a. Go to the Jetty base directory.

   ```
   $ cd /path/to/jetty-base/modules
   ```

b. In `amlogin.mod`, delete the line for `/path/to/java_agents/jetty_agent/lib/agent.jar` if it is present. This file isn't required from Java Agent 2023.9.

c. In `amlogin.mod`, update the version number for `jee-agents-sdk-version.jar`. The following example includes `jee-agents-sdk-2024.11.jar`:

```
# Jetty AM module
#
[depend]
server
security
jndi
webapp
plus
[xml]
etc/amlogin.xml
[lib]
/path/to/java_agents/jetty_agent/Agent_001/config
/path/to/java_agents/jetty_agent/locale
/path/to/java_agents/jetty_agent/lib/jee-agents-sdk-
2024.9.jar
```

8. Replace the current `agentadmin` file with the newer downloaded version:

```
$ cd /path/to/java_agents/jetty_agent/bin
$ rm agentadmin
$ cp /tmp/jetty_agent/bin/agentadmin .
```

9. Start the web applications where the agent is installed.

10. Check that the agent is performing as expected:

a. Check the correct version of the agent is running:

- Set the log level to `trace`, as described in Manage logs.
- In `/path/to/java_agents/agent_type/Agent_n/logs/debug`, search for lines containing the string `X-ForgeRock-Edge-Metadata`. The version number is given in the header.

  For example, the log file can contain the following header: `--header "X-ForgeRock-Edge-Metadata: JPA 2024.11`.

b. Navigate to a protected page on the website and confirm whether you can access it according to your configuration.

c. Check logs files for warnings and errors.

11. Allow client traffic to flow to the protected web applications.

WebLogic Java Agent software update

1. Read the release notes for information about changes in Java Agent.
2. Download the agent binaries from the BackStage download site and extract them to a temporary directory.

   The example in this section is extracted to `/tmp`, and the .jar files are in `/tmp/weblogic_agent/lib`.
3. Back up the directories for the agent installation and the web application container configuration:
   - In local configuration mode:

     ```
     $ cp -r /path/to/java_agents/weblogic_agent
     /path/to/backup
     ```
   - In remote configuration mode, back up as described in AM's Maintenance guide.
4. Add the following file to the backup:
   - `/Oracle/Middleware/Oracle_Home/user_projects/domains/base_domain/setAgentEnv_AdminServer.sh`
5. Redirect client traffic away from protected web applications.
6. Stop the web applications where the agent is installed.
7. Update the .jar files in the installation directory.

   a. Using the `.amAgentLocator` file, find the directory in which the agent was originally installed:

   ```
   $ cd /opt/container
   $ cat .amAgentLocator; echo

   /path/to/java_agents/weblogic_agent
   ```

   b. View the content of the `lib` subdirectory:

   ```
   $ cd /path/to/java_agents/weblogic_agent/lib
   $ ls -F

   agent.jar
   jee-agents-installtools-2024.9.jar
   jee-agents-sdk-2024.9.jar
   ```

c. Replace the files with the newer downloaded version:

```
$ rm -f *
$ cp /tmp/java_agents/weblogic_agent/lib/*.jar .
$ ls -F

agent.jar
jee-agents-installtools-2024.11.jar
jee-agents-sdk-2024.11.jar
```

8. Update the environment settings:

   a. Locate the `setAgentEnv_AdminServer.sh` file. The file can be in a directory such as `/Oracle/Middleware/Oracle_Home/user_projects/domains/base_domain/`.

   b. If any of the following files are listed, remove the information for the file:

      - `/path/to/java_agents/weblogic_agent/lib/agent.jar`.
      - `/path/to/java_agents/weblogic_agent/lib/jee-agents-installtools-launcher-`*`version`*. The installation launcher was removed in Java Agent 2023.6.
      - `/path/to/java_agents/weblogic_agent/lib/jee-agents-installtools-`*`version`*`.jar`.

   c. Change the version of `jee-agents-sdk-`*`version`*`.jar` to the newer downloaded version:

```
...
# Append AGENT_CLASSPATH to the WebLogic server classpath
AGENT_CLASSPATH="/path/to/java_agents/weblogic_agent/lib/jee-agents-sdk-2024.11.jar:/path/to/java_agents/weblogic_agent/locale:/path/to/java_agents/weblogic_agent/Agent_001/config"
CLASSPATH="${CLASSPATH}${CLASSPATHSEP}${AGENT_CLASSPATH}"
export CLASSPATH
...
```

   d. Save the file.

9. Replace the current `agentadmin` file with the newer downloaded version:

```
$ cd /path/to/java_agents/weblogic_agent/bin
$ rm agentadmin
$ cp /tmp/weblogic_agent/bin/agentadmin .
```

10. Start the web applications where the agent is installed.
11. Check that the agent is performing as expected:
    a. Check the correct version of the agent is running:
       - Set the log level to `trace`, as described in Manage logs.
       - In `/path/to/java_agents/agent_type/Agent_n/logs/debug`, search for lines containing the string `X-ForgeRock-Edge-Metadata`. The version number is given in the header.

         For example, the log file can contain the following header: `--header "X-ForgeRock-Edge-Metadata: JPA 2024.11`.

    b. Navigate to a protected page on the website and confirm whether you can access it according to your configuration.

    c. Check logs files for warnings and errors.
12. Allow client traffic to flow to the protected web applications.

Roll back from a drop-in software update

> IMPORTANT
>
> Before you roll back to an earlier version of Java Agent, consider whether any change to the configuration during or since upgrade could be incompatible with the previous version.

Major upgrade

Perform a major upgrade

1. Read the release notes for information about changes in Java Agent.
2. Plan for server downtime.

   Plan to route client applications to another server until the process is complete and you have validated the result. Make sure the owners of client applications are aware of the change, and let them know what to expect.
3. Download the agent binaries from the BackStage download site.
4. Back up the directories for the agent installation and the web application container configuration:
   - In local configuration mode:

     ```
     $ cp -r /path/to/java_agents/agent_type /path/to/backup
     $ cp -r /path/to/agent_type/webapps/agentapp
     ```

```
    /path/to/backup
    ```

  - In remote configuration mode, back up as described in AM's Maintenance guide.

5. Redirect client traffic away from protected web applications.
6. Stop the web applications where the agent is installed.
7. Remove the old Java Agent, as described in Remove Java Agent.
8. Install the new agent.

   The installer creates new versions of the following files, with configuration that is relevant to the new version of the agent:

   - AgentConfiguration.properties
   - AgentBootstrap.properties
   - agent-logback.xml
   - AgentPassword.properties
   - AgentKey.properties

9. Using the agent's release notes and AM's release notes, check for changes and update the configuration.

   > IMPORTANT
   >
   > To prevent errors, do not copy configuration files from the previous installation to the new installation. Use the new version of the files and update then as necessary.

   - In local configuration mode, update `AgentConfiguration.properties` manually to include properties for your environment, using backed-up files for guidance.

     The `AgentBootstrap.properties` file created by the installer contains bootstrap properties relevant to the new version of the agent.

   - In remote configuration mode, change the agent configuration using the AM admin UI.

10. Secure communication between AM and the agent with appropriate keys. For information, refer to Configure AM servers to communicate with Java Agents.
11. Start the web applications where the agent is installed.
12. Check that the agent is performing as expected:

    a. Check the correct version of the agent is running:

       - Set the log level to `trace`, as described in Manage logs.
       - In `/path/to/java_agents/agent_type/Agent_n/logs/debug`, search for lines containing the string `X-ForgeRock-Edge-Metadata`. The version

number is given in the header.

For example, the log file can contain the following header: `--header "X-ForgeRock-Edge-Metadata: JPA 2024.11.`

b. Navigate to a protected page on the website and confirm whether you can access it according to your configuration.

c. Check logs files for warnings and errors.

13. Allow client traffic to flow to the protected web applications.

Roll back from a major upgrade

IMPORTANT

Before you roll back to a previous version of Java Agent, consider whether any change to the configuration during or since upgrade could be incompatible with the previous version.

Post update and upgrade tasks

After upgrade or update, review the what’s new section in the release notes and consider activating new features and functionality.

For information about other post-installation options, refer to Post-installation tasks.

Was this helpful? 👍 👎