jamf | PRO

Jamf Pro Installation and Configuration Guide for Linux

Version 10.18.0

jamf

Jamf has made all efforts to ensure that this guide is accurate.

Jamf
100 Washington Ave S Suite 1100
Minneapolis, MN 55401-2155
(612) 605-6625

Contents

jamf | PRO

Preface

About This Guide

The Jamf Pro server is a web app that functions as the administrative core of Jamf Pro. The Jamf Pro server allows you to perform inventory and remote management and configuration tasks on managed computers and mobile devices. All other administrative applications in Jamf Pro communicate with the Jamf Pro server.

This guide provides step-by-step instructions for installing and configuring Jamf Pro on the Linux platform using the Jamf Pro Installer for Linux. It also includes instructions for performing a manual installation on Red Hat Enterprise Linux or Ubuntu and explains how to perform advanced configuration tasks.

Additional Resources

Jamf Nation

https://www.jamf.com/jamf-nation/
The Jamf Nation website allows you to communicate with other Jamf Pro administrators via discussions, submit feature requests, and access several different types of resources related to Jamf Pro.

Knowledge Base

https://www.jamf.com/jamf-nation/articles
The Knowledge Base contains hundreds of articles that address frequently asked questions and common issues.

Product Documentation

To access the following product documentation for a specific Jamf Pro version, log in to Jamf Nation and go to:
https://www.jamf.com/jamf-nation/my/products

- *Jamf Pro Release Notes*
  The release notes include information on new features and enhancements, system requirements, functionality changes, and bug fixes.
- *Jamf Pro Administrator's Guide*
  This guide contains overviews of features and instructions for performing administrative tasks using Jamf Pro.

In addition, you can search Jamf Nation to find best practice workflows, technical papers, and documentation for other Jamf Pro apps.

Other Resources

For access to other Jamf Pro-related resources, visit the following webpages:

- Resources on jamf.com
  The Resources area on the Jamf website gives you access to product documentation, best practice workflows, technical papers, and more.
- Jamf 100 Course
  The Jamf 100 Course offers a self-paced introduction to Jamf Pro and an enterprise-focused foundation of the macOS, iOS, and tvOS platforms.
- Jamf Online Training Catalog
  The Jamf Online Training catalog provides self-paced modules to help you master Apple device management with Jamf Pro. This resource is available for free to all Jamf customers.

- Jamf Knowledge Base Videos
  The Jamf YouTube channel features Knowledge Base videos and troubleshooting tips on managing computers and mobile devices with Jamf Pro.
- Jamf Marketplace
  The Jamf Marketplace is a central location for you to find, learn about, and utilize valuable tools to integrate with and extend the Jamf platform.

jamf | PRO

Overview of Technologies

Applications and Utilities

This section provides an overview of the applications and utilities that you need to install and maintain Jamf Pro.

Jamf Pro Server

The Jamf Pro server is a web app that functions as the administrative core of Jamf Pro. The Jamf Pro server allows you to perform inventory and remote management and configuration tasks on managed computers and mobile devices. All other administrative applications in Jamf Pro communicate with the Jamf Pro server.

Jamf Pro Installer for Linux

The Jamf Pro Installer for Linux is a .run file that allows you to install and upgrade Jamf Pro on supported Linux operating systems.

To obtain the Jamf Pro Installer for Linux, log in to Jamf Nation, click **Show alternative downloads**, and then click the **Download** button under "Jamf Pro Installer for Linux" on the following page:

https://www.jamf.com/jamf-nation/my/products

Jamf Pro Server Tools

Jamf Pro Server Tools allows you to back up and restore the Jamf Pro database. It also allows you to restart Apache Tomcat and MySQL and modify their settings.

Jamf Pro Server Tools is installed automatically when you run the Jamf Pro Installer. It is located at:

```
/usr/local/jss/bin/server-tools-gui.jar
```

Jamf Pro Server Tools also has a command-line interface (CLI). It is located at:

```
/usr/local/jss/bin/jamf-pro
```

Note: The Jamf Pro installer for Linux includes only the 64-bit version of the CLI. If you are using 32-bit Linux, you must manually download and use the 32-bit version of the CLI. For instructions, see the following Knowledge Base article:
Using the Jamf Pro Server Tools Command-Line Interface

Jamf Pro System Requirements

For system requirements information, see "Jamf Pro System Requirements" in the Jamf Pro Release Notes for your version of Jamf Pro.

jamf | PRO

Installation

Installing Jamf Pro Using the Installer

Installing Jamf Pro using the installer involves the following steps:

1. Install the required software (if you haven't already).
2. Run the Jamf Pro Installer.
3. Create the Jamf Pro database.
4. Connect to the Jamf Pro Server.

Note: For instructions on how to manually install Jamf Pro on Linux, see one of the following:

- Manually Installing Jamf Pro—Red Hat Enterprise Linux
- Manually Installing Jamf Pro—Ubuntu

Requirements

The server used to host Jamf Pro should meet the minimum requirements for operating system, Tomcat version, database configuration, and Java installation. For additional information on these Jamf Pro Server Environment requirements, see "Jamf Pro System Requirements" in the Jamf Pro Release Notes for your version of Jamf Pro.

In addition, the following resources are recommended as the minimum allocation for a typical installation of Jamf Pro:

Linux

- A 64-bit capable Intel processor
- 8 GB of RAM
- 150 GB of disk space available
- The "wget" utility installed
- Ports 8443 and 8080 available

Note: Each installation of Jamf Pro and its required services is unique, and requirements may vary depending on your implementation. Questions regarding scaling an environment's resources beyond the typical recommendations can be submitted to Jamf Support.

Step 1: Install the Prerequisite Software

Java must be installed on the server where you will install Jamf Pro. MySQL must be installed on a server before you can create the Jamf Pro database.

For instructions, see the Installing Java and MySQL for Jamf Pro 10.14.0 or Later Knowledge Base article.

Note: MySQL is not required to be installed on the same server as the Jamf Pro web application. For more information, see Clustering.

Step 2: Run the Jamf Pro Installer

The Jamf Pro Installer for Linux installs Apache Tomcat and the Jamf Pro web app.

1. Copy the Jamf Pro Installer for Linux (`jamfproinstaller.run`) to the server.

   Note: To obtain the Jamf Pro Installer for Linux, log in to Jamf Nation, click **Show alternative downloads**, and then click the **Download** button under "Jamf Pro Installer for Linux" on the following page:
   https://www.jamf.com/jamf-nation/my/products

2. Log in to the server as a user with superuser privileges.
3. Initiate the installer by executing a command similar to the following:

   ```
   sudo sh /path/to/jamfproinstaller.run
   ```

4. When the requirement check is complete, type "y" to proceed.
5. (Red Hat Enterprise Linux only) When the installation is complete, edit the firewall configuration to allow access to port 8443 by executing:

   ```
   sudo system-config-firewall-tui
   ```

6. (Red Hat Enterprise Linux only) Choose **Other** or **Customize**, and manually add port 8443 with TCP protocol. The option you choose depends on whether you have a GUI or shell-only interface.
7. Configure Jamf Pro to start automatically when the server is rebooted:
   a. Check the state of the Tomcat service's “enabled on boot” setting by executing the following command:

      ```
      sudo systemctl is-enabled jamf.tomcat8.service
      ```

b. If the result indicates the Tomcat service is “disabled”, enable the service permanently by executing the following command:

```
sudo systemctl enable jamf.tomcat8
```

c. Confirm the “enabled on boot” setting is "enabled" by executing the following command:

```
sudo systemctl is-enabled jamf.tomcat8.service
```

d. If the Tomcat service is not already running, you can start the Tomcat service manually by rebooting the server or by executing the following command:

```
sudo systemctl start jamf.tomcat8
```

Step 3: Create the Jamf Pro Database

You must create a MySQL database before you can use Jamf Pro. For instructions, see the Creating the Jamf Pro Database Using the Jamf Pro Server Tools Command-Line Interface Knowledge Base article.

Step 4: Connect to the Jamf Pro Server

1. Configure the database connection settings using Jamf Pro Server Tools GUI or CLI. For instructions, see the Editing the Database Connection Using Jamf Pro Server Tools Knowledge Base article.
2. Access Jamf Pro by opening a web browser and typing the protocol, IP address or hostname of the server, and port. For example: https://jamf.mycompany.com:8443/

Installed Files and Folders

The following files and folders are installed when you run the Jamf Pro Installer:

Jamf Pro web app
The files that make up the Jamf Pro web app (formerly the JSS web app) are stored in the following location:
`/usr/local/jss/tomcat/webapps/ROOT/`

Apache Tomcat
Tomcat is the web application server that runs the Jamf Pro web app. A directory named `tomcat` is installed in the following location:
`/usr/local/jss/`

For more information about the version of Tomcat installed by the Jamf Pro Installer, see the Apache Tomcat Versions Installed by the Jamf Pro Installer Knowledge Base article.

jamf.tomcat8

This is the service file for Tomcat. It is installed in the following location:

`/etc/init.d/jamf.tomcat8`

server.xml

The Jamf Pro Installer installs a modified copy of Tomcat's `server.xml` file. This file enables SSL, ensures that Jamf Pro appears in the `root` context, and enables database connection pooling. It is installed in the following location:

`/usr/local/jss/tomcat/conf/`

keystore

Tomcat requires a keystore file to provide connections over SSL. The Jamf Pro Installer creates a default `.keystore` file and stores it in the following location:

`/usr/local/jss/tomcat/`

Jamf Pro Server Tools

Jamf Pro Server Tools, filename `server-tools-gui.jar`, is installed in the following location:

`/usr/local/jss/bin/`

Jamf Pro Server Tools also has a command-line interface (CLI), filename `jamf-pro`, that is installed in the same location.

> **Note:** The Jamf Pro installer for Linux includes only the 64-bit version of the CLI. If you are using 32-bit Linux, you must manually download and use the 32-bit version of the CLI. For instructions, see the following Knowledge Base article:
> Using the Jamf Pro Server Tools Command-Line Interface

Database backup location

By default, Jamf Pro Server Tools stores database backups in the following location:

`/usr/local/jss/backups/database/`

Logs

Logs for the installation and for the Jamf Pro server (formerly the Jamf Software Server) are stored in the following location:

`/usr/local/jss/logs/`

Upgrading Jamf Pro Using the Installer

1. Review the following Knowledge Base articles:
   - Preparing to Upgrade Jamf Pro
   - Incremental Upgrade Scenarios for Jamf Pro 10.0.0 or Later
2. Ensure that you have backed up the current database.
   For more information, see Backing Up the Database.
3. If you are upgrading from Jamf Pro 10.13.0 or earlier, follow the instructions in the Migrating to Java 11 Knowledge Base article to migrate from Java 8 to Java 11.
4. Copy the latest version of the Jamf Pro Installer for Linux (`jamfproinstaller.run`) to the server.

   Note: To obtain the Jamf Pro Installer for Linux, log in to Jamf Nation and click **Show alternative downloads** below the Jamf Pro DMG on the following page:
   https://www.jamf.com/jamf-nation/my/products

   Note: The Jamf Pro Installer for Linux cannot be used to upgrade Jamf Pro 8.1 or earlier.
5. Log in to the server as a user with superuser privileges.
6. Initiate the installer by executing:

   ```
   sudo sh /path/to/jamfproinstaller.run
   ```
7. Follow the onscreen instructions to complete the upgrade.
8. Verify that the `Connector` settings for port 8443 in the `server.xml` file match the settings listed in the Configuring Supported Ciphers for Tomcat HTTPS Connections Knowledge Base article, and modify them if needed.
9. If you modified the `server.xml` file, restart Tomcat.
10. Configure Jamf Pro to start automatically when the server is rebooted:
    a. Check the state of the Tomcat service's “enabled on boot” setting by executing the following command:

    ```
    sudo systemctl is-enabled jamf.tomcat8.service
    ```

    b. If the result indicates the Tomcat service is “disabled”, enable the service permanently by executing the following command:

    ```
    sudo systemctl enable jamf.tomcat8
    ```

    c. Confirm the “enabled on boot” setting is "enabled" by executing the following command:

    ```
    sudo systemctl is-enabled jamf.tomcat8.service
    ```

d. If the Tomcat service is not already running, you can start the Tomcat service manually by rebooting the server or by executing the following command:

```
sudo systemctl start jamf.tomcat8
```

11. Log in to Jamf Pro and verify devices are checking in as expected.

Important: If you are upgrading to Jamf Pro 10.6.0 or later, you must make a one-time change to the MySQL configuration to avoid performance issues. See "Step 2. Configure MySQL" in the Creating the Jamf Pro Database Using the Jamf Pro Server Tools Command-Line Interface Knowledge Base article for instructions.

Manually Installing Jamf Pro: Red Hat Enterprise Linux

This section provides a basic set of steps for manually installing and configuring Jamf Pro on a Red Hat Enterprise Linux server.

Note: The supporting scripts and configuration used for manual installation differ from an installation using the Jamf Pro Installer for Linux. If you do not want to manually install Jamf Pro, you can use the Jamf Pro Installer for Linux. For information about obtaining the installer and installation instructions, see Installing Jamf Pro Using the Installer.

Note: If you are upgrading to Tomcat 8.5, you will need to manually modify the `server.xml` file to make it compatible with Tomcat 8.5. For more information, see the Server.xml Changes for Tomcat 8.5 Knowledge Base article.

Requirements

The server used to host Jamf Pro should meet the minimum requirements for operating system, Tomcat version, database configuration, and Java installation. For additional information on these Jamf Pro Server Environment requirements, see "Jamf Pro System Requirements" in the Jamf Pro Release Notes for your version of Jamf Pro.

In addition, the following resources are recommended as the minimum allocation for a typical installation of Jamf Pro:

Linux

- A 64-bit capable Intel processor
- 8 GB of RAM
- 150 GB of disk space available
- The "wget" utility installed
- Ports 8443 and 8080 available

Note: Each installation of Jamf Pro and its required services is unique, and requirements may vary depending on your implementation. Questions regarding scaling an environment's resources beyond the typical recommendations can be submitted to Jamf Support.

You must also obtain the following to manually install Jamf Pro:

- Jamf Pro web app (`ROOT.war`)—To obtain this item in the Jamf Pro manual installation archive, log in to Jamf Nation, click **Show alternative downloads**, then click the **Download** button below "Jamf Pro Manual Installation" on the following page: https://www.jamf.com/jamf-nation/my/products

- Jamf Pro Server Tools Command-Line Interface (CLI)—Instructions for obtaining this item are provided below.

Installation and Configuration

Follow the step-by-step instructions in this section to install and configure Jamf Pro on Red Hat Enterprise Linux.

Step 1: Install Java and MySQL

Java and MySQL must be installed on the server before you can create the Jamf Pro database and install Jamf Pro. For instructions, see the Installing Java and MySQL for Jamf Pro 10.14.0 or Later Knowledge Base article.

Step 2: Install Tomcat

Apache Tomcat is the web application server that runs Jamf Pro.

You will need URLs to download and verify the Tomcat binary distribution that you intend to install. If you have access to a web browser on the Red Hat Enterprise Linux system or if you are remotely connected to a Red Hat Enterprise Linux shell session from your computer, you may want to copy these URLs just before executing the download commands. If not, you may want to copy the URLs into a text document for reference as you type them.

Copy the Necessary URLs

1. In a web browser, open the Tomcat 8 download page: https://tomcat.apache.org/download-80.cgi
2. On the Tomcat 8 download page, navigate to **Tomcat 8.5.x** > **Binary Distributions** > **Core**.
3. Right-click the "tar.gz" link, and choose **Copy Link** to copy the URL.
4. Paste the copied URL into a document for reference, or paste it directly into the download command (see below).
5. On the Tomcat 8 download page, right-click the "sha512" link, and choose **Copy Link** to copy the URL.
6. Paste the copied URL into a document for reference, or paste it directly into the download command (see below).

Prepare the System for Tomcat

Note: Tomcat 8.5.42 is used in the commands in this section. When you execute the commands, substitute "8.5.42" with the specific version of Tomcat 8.5.x that you want to install. The most recent version of Tomcat 8.5.x can be downloaded from the following page: https://tomcat.apache.org/download-80.cgi

You may need to install the "wget" utility to execute the commands found in this guide. Install "wget" by executing:

```
sudo yum install wget
```

1. Create a "tomcat" group by executing:

```
sudo groupadd tomcat
```

2. Create a "tomcat" user by executing:

```
sudo useradd -r -g tomcat -d /opt/apache-tomcat-8.5.42 -s /bin/nologin
tomcat
```

3. Create a temporary directory for the downloads and change to the directory by executing:

```
mkdir /tmp/tomcat && cd /tmp/tomcat
```

4. Download the Tomcat binary distribution using the `tar.gz` URL (see above) with a command similar to:

```
wget https://archive.apache.org/dist/tomcat/tomcat-8/v8.5.42/bin/apache-
tomcat-8.5.42.tar.gz
```

5. Download the Tomcat `sha512` reference file using the URL (see above) with a command similar to:

```
wget https://archive.apache.org/dist/tomcat/tomcat-8/v8.5.42/bin/apache-
tomcat-8.5.42.tar.gz.sha512
```

6. Using the filename of the `sha512` reference file, verify the binary distribution has not been modified with a command similar to:

```
sha512sum -c apache-tomcat-8.5.42.tar.gz.sha512
```

   The result should be something like:

```
apache-tomcat-8.5.42.tar.gz: OK
```

7. If the result of the check indicates that the downloaded binary is "OK", extract the contents of the file with a command similar to:

```
tar -zxvf apache-tomcat-8.5.42.tar.gz
```

8. Move the extracted contents to the desired location with a command similar to:

```
sudo mv apache-tomcat-8.5.42 /opt/
```

9. Ensure the "tomcat" user and group have ownership of the directory and contents with a command similar to:

```
sudo chown -R tomcat:tomcat /opt/apache-tomcat-8.5.42
```

10. Create a symlink to the directory with a command similar to:

```
sudo ln -s /opt/apache-tomcat-8.5.42 /opt/tomcat
```

Note: Installing Tomcat in a directory named with the version number and then symlinking to this directory allows for an easy Tomcat upgrade path later—simply install a newer version of Tomcat in a new directory with its version number in the name and change the symlink to point to the new version.

Step 3: Create a Tomcat Management Service

You will need to know the path to the base Java folder (the path without `/bin/java` at the end) to provide it as the JAVA_HOME environment variable in the management service. The specific path to Java will vary depending on the platform, OS version, Open JDK vs. Oracle JDK, etc.

Following are different methods that you can use to find the path to Java on your system:

- Reference the JAVA_HOME environment variable:

```
$ echo $JAVA_HOME
/usr/lib/jvm/java-11-openjdk
```

- Use `alternatives`:

```
$ alternatives --config java

There is 1 program that provides 'java'.

  Selection    Command
-----------------------------------------------
*+ 1           java-11-openjdk.x86_64 (/usr/lib/jvm/java-11-openjdk-
               11.0.3.7-0.0.1.el7_6.x86_64/bin/java)
Enter to keep the current selection[+], or type selection number:
```

- Use `whereis` to find "java" and follow the breadcrumbs:

```
$ whereis java
java: /usr/bin/java
      /usr/lib/java
      /etc/java
      /usr/share/java
      /usr/share/man/man1/java.1.gz
$ ls -la /usr/bin | grep java$
lrwxrwxrwx. 1 root root 22 Jul 16 10:05 java -> /etc/alternatives/java
$ ls -la /etc/alternatives | grep java$
lrwxrwxrwx. 1 root root 65 Jul 16 10:05 java ->
/usr/lib/jvm/java-11-openjdk-11.0.3.7-0.0.1.el7_6.x86_64/bin/java
```

Complete the following steps to create a Tomcat service:

1. Using your preferred text editor, create a tomcat systemd service file with a command similar to:

```
sudo vi /etc/systemd/system/tomcat.service
```

2. Paste the following into the `tomcat.service` file:

```
[Unit]
Description=Jamf Pro Web Application Container
Wants=network.target
After=syslog.target network.target
[Service]
Type=forking
Environment=JAVA_HOME=/usr/lib/jvm/java-11-openjdk-11.0.4.11-0.el7_6.x86_64
Environment=CATALINA_PID=/opt/tomcat/temp/tomcat.pid
Environment=CATALINA_HOME=/opt/tomcat
Environment=CATALINA_BASE=/opt/tomcat
Environment='CATALINA_OPTS=-server -XX:+UseParallelGC'
Environment='JAVA_OPTS=-Djava.awt.headless=true -Djava.net.
preferIPv4Stack=true'
ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/opt/tomcat/bin/shutdown.sh
User=tomcat
Group=tomcat
UMask=0007
RestartSec=10
Restart=always
[Install]
WantedBy=multi-user.target
```

3. Ensure the JAVA_HOME value in the tomcat.service file matches the path to the JDK installed on the system, not including /bin/java at the end (see above for guidance on finding this path).
4. Save the `tomcat.service` file.

5. Restart the systemd daemon by executing:

```
sudo systemctl daemon-reload
```

6. Start the Tomcat service by executing:

```
sudo systemctl start tomcat
```

7. To check the status of Tomcat, execute:

```
systemctl status tomcat
```

8. Enable auto startup of the Tomcat service at boot by executing:

```
sudo systemctl enable tomcat
```

Step 4: Install the Jamf Pro Server Tools CLI

1. Download the Jamf Pro Server Tools CLI. For instructions, see the following Knowledge Base article: Using the Jamf Pro Server Tools Command-Line Interface

> **Note:** On Red Hat Enterprise Linux, the `sudo` command may not use the expected home directory. You may need to use the `su` command or the sudo `su` command to reach the root user account to set configuration options for `jamf-pro`. If you use the `su` command, you must provide the root user password. If you use the `sudo su` command, you can access the root user account by providing the password of the current account. Note that the current account must be configured in the `sudoers` file to use `sudo su`. After executing any `jamf-pro config` command to set configuration values, it is highly recommended that you execute the `jamf-pro config list` command to ensure the values you intended were saved.

2. Configure the Tomcat directory by executing a command similar to the following:

```
jamf-pro config set --tomcat-dir /opt/tomcat
```

3. Configure the Tomcat service by executing a command similar to the following:

```
jamf-pro config set --tomcat-service tomcat.service
```

It is highly recommended that you regularly create backups as you work toward a fully configured and operational Jamf Pro. For instructions, see Backing Up the Database.

> **Note:** You can also download the Jamf Pro Server Tools GUI by clicking the following link: https://archive.services.jamfcloud.com/jamf-pro-server-tools/release/latest/gui/server-tools.jar
>
> For more information, see the Jamf Pro Server Tools Overview Knowledge Base article.

Step 5: Create the Jamf Pro Database

You can create the Jamf Pro database using one of the following methods:

- **Method 1:** Creating the Jamf Pro Database Using the Jamf Pro Server Tools CLI
  Follow the instructions in the Creating the Jamf Pro Database Using the Jamf Pro Server Tools Command-Line Interface Knowledge Base article.
- **Method 2:** Manually Creating the Jamf Pro Database
  Follow the instructions in the Manually Creating the Jamf Pro Database Knowledge Base article.

Step 6: Allocate Additional Memory to Tomcat

> **Note:** To accommodate a large number of computers in Jamf Pro, it may be necessary to allocate additional Java Virtual Machine (JVM) memory to Tomcat. If there are other services running on your server, make sure to leave enough memory to accommodate them when configuring the `--max-memory` setting with Jamf Pro Server Tools.

1. Open Terminal.
2. Set the minimum Tomcat memory by executing the following command:

```
jamf-pro server config set --min-memory 256M
```

3. Set the maximum Tomcat memory by executing the following command:

```
jamf-pro server config set --max-memory 512M
```

Step 7: Configure the Firewall

On systems where the firewall is enabled, the following instructions will assist you in configuring the firewall to allow inbound access on port 8443, which provides access to the Jamf Pro web application.

> **Note:** If the firewall is not enabled, the following commands are not necessary. However, it is highly recommended that you follow industry best practices for production web-accessible systems, including enabling the firewall. For more information about securing your Jamf Pro server, contact your Jamf account representative.

1. Check to see if the firewall is running by executing the following command:

```
systemctl status firewalld
```

2. If the firewall is "active":
   a. List the ports that are open by executing the following command:

```
sudo firewall-cmd --list-ports
```

b. Edit the firewall configuration to allow access to port 8443 by executing the following commands:

```
sudo firewall-cmd --zone=public --add-port=8443/tcp --permanent
sudo firewall-cmd --reload
```

Step 8: Install Jamf Pro

1. (Optional) Create a `/tmp/jamf` directory in which to temporarily store downloads, tools, and scripts for Jamf Pro by executing the following command:

```
mkdir /tmp/jamf
```

2. (Optional) Copy the ROOT.war file from the Jamf Pro manual download archive to the `/tmp/jamf` directory that you just created.

3. Stop the Tomcat service by executing:

```
sudo systemctl stop tomcat
```

or

```
sudo service tomcat stop
```

4. Move the default Tomcat web app out of the way by executing:

```
sudo mv /opt/tomcat/webapps/ROOT /opt/tomcat/webapps/TOMCAT
```

5. Copy the Jamf Pro web app archive to the Tomcat `webapps` directory by executing a command similar to the following:

```
sudo cp /tmp/jamf/ROOT.war /opt/tomcat/webapps/
```

6. Assign appropriate access to the ROOT.war by executing:

```
sudo chown tomcat:tomcat /opt/tomcat/webapps/ROOT.war
sudo chmod 750 /opt/tomcat/webapps/ROOT.war
```

7. Generate a keystore to enable SSL for Tomcat by executing a command similar to the following with the `keypass`, `storepass`, `dname,` and `validity` values customized for your environment:

```
sudo keytool -genkey -alias tomcat -keyalg RSA -keypass
  "changeit" -storepass "changeit" -dname "CN=jamf.mycompany.com,
  OU=Jamf IT, O=Jamf, L=Minneapolis, ST=MN, C=US"
  -keystore /opt/tomcat/keystore -validity <numdays>
```

Make sure to enter the following attributes as appropriate to your site:

Attribute	Value	Example
CN=	Fully qualified domain name of the server	jamf.mycompany.com
OU=	Organizational unit	Jamf IT
O=	Organization	Jamf
L=	Location (city or office)	Minneapolis
ST=	State, province, or county	MN
C=	Country or region	US

8. Back up the Tomcat `server.xml` configuration file by executing:

```
sudo cp /opt/tomcat/conf/server.xml /opt/tomcat/conf/server.xml.bak
```

9. Open the `server.xml` configuration file in a text editor with a command similar to:

```
sudo vi /opt/tomcat/conf/server.xml
```

10. Locate the following comment section in the file (note that this is the `Connector on port 8443` not the `Connector on port 8443 with HTTP/2` section):

```
<!-- Define a SSL/TLS HTTP/1.1 Connector on port 8443
        This connector uses the NIO implementation.
        ...
-->
```

11. Replace the commented out (initial `<!--` and trailing `-->`) Connector tag immediately following the comment (shown in "a" below) with the Connector tag text shown in "b" below:
a.

```
<!-- Define a SSL/TLS HTTP/1.1 Connector on port 8443
     ...
-->
<!--
<Connector port="8443" ...
  ...
/>
-->
```

b.

```
<!-- Define a SSL/TLS HTTP/1.1 Connector on port 8443
           This connector uses the NIO implementation that requires the
           JSSE style configuration. When using the APR/native
           implementation, the OpenSSL style configuration is required
           as described in the APR/native documentation -->
<Connector URIEncoding="UTF-8"
           server="Apache Tomcat"
           port="8443"
           executor="tomcatThreadPool"
           SSLEnabled="true"
           maxPostSize="-1"
           scheme="https"
           protocol="org.apache.coyote.http11.Http11Nio2Protocol"
           sslImplementationName="org.apache.tomcat.util.net.jsse.
JSSEImplementation"
           secure="true">
  <SSLHostConfig sslProtocol="TLS"
                 protocols="TLSv1.2"
                 honorCipherOrder="true"
                 certificateVerification="none"
                 ciphers="TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384,
                          TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,
                          TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384,
                          TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384,
                          TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,
                          TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,
                          TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256,
                          TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256,
                          TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384,
                          TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384,
                          TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA,
                          TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA,
                          TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384,
                          TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384,
                          TLS_ECDH_RSA_WITH_AES_256_CBC_SHA,
                          TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA,
                          TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,
                          TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256,
                          TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,
                          TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA,
                          TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256,
                          TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256,
                          TLS_ECDH_RSA_WITH_AES_128_CBC_SHA,
                          TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA" >
    <Certificate type="RSA"
                 certificateKeystoreFile="/opt/tomcat/keystore"
                 certificateKeystorePassword="changeit" />
  </SSLHostConfig>
</Connector>
```

12. Before saving the file, ensure the `keystorePass` value is set to the value you used for `storepass` when creating the keystore.

13. Save the `server.xml` file.
14. Start the tomcat service by executing:

    ```
    sudo systemctl start tomcat
    ```

    or

    ```
    sudo service tomcat start
    ```

    The Tomcat service will automatically unpack the `ROOT.war` file into a `ROOT` directory in the `webapps` directory.

Step 9: Connect to the Jamf Pro Server

1. Configure the database connection settings using Jamf Pro Server Tools GUI or CLI. For instructions, see the Editing the Database Connection Using Jamf Pro Server Tools Knowledge Base article.
2. Access Jamf Pro by opening a web browser and typing the protocol, IP address or hostname of the server, and port. For example: https://jamf.mycompany.com:8443/

Manually Installing Jamf Pro: Ubuntu

This section provides a basic set of steps for manually installing and configuring Jamf Pro on Ubuntu LTS Server.

Note: The supporting scripts and configuration used for manual installation differ from an installation using the Jamf Pro Installer for Linux. If you do not want to manually install Jamf Pro, you can use the Jamf Pro Installer for Linux. For information about obtaining the installer and installation instructions, see Installing Jamf Pro Using the Installer.

Note: If you are upgrading to Tomcat 8.5, you will need to manually modify the server.xml file to make it compatible with Tomcat 8.5. For more information, see the Server.xml Changes for Tomcat 8.5 Knowledge Base article.

Requirements

The server used to host Jamf Pro should meet the minimum requirements for operating system, Tomcat version, database configuration, and Java installation. For additional information on these Jamf Pro Server Environment requirements, see "Jamf Pro System Requirements" in the Jamf Pro Release Notes for your version of Jamf Pro.

In addition, the following resources are recommended as the minimum allocation for a typical installation of Jamf Pro:

Linux

- A 64-bit capable Intel processor
- 8 GB of RAM
- 150 GB of disk space available
- The "wget" utility installed
- Ports 8443 and 8080 available

Note: Each installation of Jamf Pro and its required services is unique, and requirements may vary depending on your implementation. Questions regarding scaling an environment's resources beyond the typical recommendations can be submitted to Jamf Support.

You must also obtain the following to manually install Jamf Pro:

- Jamf Pro web app (`ROOT.war`)—To obtain this item in the Jamf Pro manual installation archive, log in to Jamf Nation, click **Show alternative downloads**, then click the **Download** button below "Jamf Pro Manual Installation" on the following page: https://www.jamf.com/jamf-nation/my/products
- Jamf Pro Server Tools Command-Line Interface (CLI)—Instructions for obtaining this item are provided below.

Installation and Configuration

Follow the step-by-step instructions in this section to install and configure Jamf Pro on Ubuntu LTS Server.

Step 1: Install Java and MySQL

Java and MySQL must be installed on the server before you can create the Jamf Pro database and install Jamf Pro. For instructions, see the Installing Java and MySQL for Jamf Pro 10.14.0 or Later Knowledge Base article.

Step 2: Install Tomcat

Apache Tomcat is the web application server that runs Jamf Pro.

You will need URLs to download and verify the Tomcat binary distribution that you intend to install. If you have access to a web browser on the Ubuntu system or if you are remotely connected to an Ubuntu shell session from your computer, you may want to copy these URLs just before executing the download commands. If not, you may want to copy the URLs into a text document for reference as you type them.

Copy the Necessary URLs

1. In a web browser, open the Tomcat 8 download page: https://tomcat.apache.org/download-80.cgi
2. On the Tomcat 8 download page, navigate to **Tomcat 8.5.x** > **Binary Distributions** > **Core**.
3. Right-click the "tar.gz" link, and choose **Copy Link** to copy the URL.
4. Paste the copied URL into a document for reference, or paste it directly into the download command (see below).
5. On the Tomcat 8 download page, right-click the "sha512" link, and choose **Copy Link** to copy the URL.
6. Paste the copied URL into a document for reference, or paste it directly into the download command (see below).

Prepare the System for Tomcat

Note: Tomcat 8.5.42 is used in the commands in this section. When you execute the commands, substitute "8.5.42" with the specific version of Tomcat 8.5.x that you want to install. The most recent version of Tomcat 8.5.x can be downloaded from the following page: https://tomcat.apache.org/download-80.cgi

1. Create a "tomcat" group by executing:

```
sudo groupadd tomcat
```

2. Create a "tomcat" user by executing:

```
sudo useradd -r -g tomcat -d /opt/apache-tomcat-8.5.42 -s /bin/nologin
tomcat
```

3. Create a temporary directory for the downloads and change to the directory by executing:

```
mkdir /tmp/tomcat && cd /tmp/tomcat
```

4. Download the Tomcat binary distribution using the `tar.gz` URL (see above) with a command similar to:

```
wget https://archive.apache.org/dist/tomcat/tomcat-8/v8.5.42/bin/apache-
tomcat-8.5.42.tar.gz
```

5. Download the Tomcat `sha512` reference file using the URL (see above) with a command similar to:

```
wget https://archive.apache.org/dist/tomcat/tomcat-8/v8.5.42/bin/apache-
tomcat-8.5.42.tar.gz.sha512
```

6. Using the filename of the `sha512` reference file, verify the binary distribution has not been modified with a command similar to:

```
sha512sum -c apache-tomcat-8.5.42.tar.gz.sha512
```

   The result should be something like:

```
apache-tomcat-8.5.42.tar.gz: OK
```

7. If the result of the check indicates that the downloaded binary is "OK", extract the contents of the file with a command similar to:

```
tar -zxvf apache-tomcat-8.5.42.tar.gz
```

8. Move the extracted contents to the desired location with a command similar to:

```
sudo mv apache-tomcat-8.5.42 /opt/
```

9. Ensure the "tomcat" user and group have ownership of the directory and contents with a command similar to:

```
sudo chown -R tomcat:tomcat /opt/apache-tomcat-8.5.42
```

10. Create a symlink to the directory with a command similar to:

```
sudo ln -s /opt/apache-tomcat-8.5.42 /opt/tomcat
```

Note: Installing Tomcat in a directory named with the version number and then symlinking to this directory allows for an easy Tomcat upgrade path later—simply install a newer version of Tomcat in a new directory with its version number in the name and change the symlink to point to the new version.

Step 3: Create a Tomcat Management Service

You will need to know the path to the base Java folder (the path without `/bin/java` at the end) to provide it as the `JAVA_HOME` environment variable in the management service. The specific path to Java will vary depending on the platform, OS version, OpenJDK vs. Oracle JDK, etc.

Following are different methods that you can use to find the path to Java on your system:

- Reference the `JAVA_HOME` environment variable:

```
$ echo $JAVA_HOME
/usr/lib/jvm/java-11-openjdk-amd64
```

- Use `update-alternatives`:

```
$ sudo update-alternatives --config java
There is only one alternative in link group java (providing /usr/bin
/java): /usr/lib/jvm/java-11-openjdk-amd64/bin/java
Nothing to configure.
```

- `Use whereis to find "java" and follow the breadcrumbs:`

```
$ whereis java
java: /usr/bin/java /usr/share/java /usr/share/man/man1/java.1.gz
$ ls -la /usr/bin | grep java$
lrwxrwxrwx 1 root root 22 Jun 26 11:29 java -> /etc/alternatives/java
$ ls -la /etc/alternatives | grep java$
lrwxrwxrwx 1 root root 43 Jun 26 11:32 java -> /usr/lib/jvm/java-11-
openjdk-amd64/bin/java
```

Complete the following steps to create a Tomcat service:

1. Using your preferred text editor, create a tomcat systemd service file with a command similar to:

```
sudo vi /etc/systemd/system/tomcat.service
```

2. Paste the following into the `tomcat.service` file:

```
[Unit]
Description=Jamf Pro Web Application Container
Wants=network.target
After=syslog.target network.target
[Service]
Type=forking
Environment=JAVA_HOME=/usr/lib/jvm/java-11-openjdk-amd64
Environment=CATALINA_PID=/opt/tomcat/temp/tomcat.pid
Environment=CATALINA_HOME=/opt/tomcat
Environment=CATALINA_BASE=/opt/tomcat
Environment='CATALINA_OPTS=-server -XX:+UseParallelGC'
Environment='JAVA_OPTS=-Djava.awt.headless=true -Djava.net.
preferIPv4Stack=true'
ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/opt/tomcat/bin/shutdown.sh
User=tomcat
Group=tomcat
UMask=0007
RestartSec=10
Restart=always
[Install]
WantedBy=multi-user.target
```

3. Ensure the JAVA_HOME value in the `tomcat.service` file matches the path to the JDK installed on the system, not including `/bin/java` at the end (see above for guidance on finding this path).
4. Save the `tomcat.service` file.
5. Restart the systemd daemon by executing:

```
sudo systemctl daemon-reload
```

6. Start the Tomcat service by executing:

```
sudo systemctl start tomcat
```

7. To check the status of Tomcat, execute:

```
systemctl status tomcat
```

Note: You need to press the Q key to exit from the status reporting.

8. Enable auto startup of the Tomcat service at boot by executing:

```
sudo systemctl enable tomcat
```

Step 4: Install the Jamf Pro Server Tools CLI

1. Download the Jamf Pro Server Tools CLI. For instructions, see the following Knowledge Base article: Using the Jamf Pro Server Tools Command-Line Interface
2. Configure the Tomcat directory by executing a command similar to the following:

```
jamf-pro config set --tomcat-dir /opt/tomcat
```

3. Configure the Tomcat service by executing a command similar to the following:

```
jamf-pro config set --tomcat-service tomcat.service
```

It is highly recommended that you regularly create backups as you work toward a fully configured and operational Jamf Pro. For instructions, see Backing Up the Database.

Note: You can also download the Jamf Pro Server Tools GUI by clicking the following link: https://archive.services.jamfcloud.com/jamf-pro-server-tools/release/latest/gui/server-tools.jar

For more information, see the Jamf Pro Server Tools Overview Knowledge Base article.

Step 5: Create the Jamf Pro Database

You can create the Jamf Pro database using one of the following methods:

- **Method 1:** Creating the Jamf Pro Database Using the Jamf Pro Server Tools CLI
  Follow the instructions in the Creating the Jamf Pro Database Using the Jamf Pro Server Tools Command-Line Interface Knowledge Base article.
- **Method 2:** Manually Creating the Jamf Pro Database
  Follow the instructions in the Manually Creating the Jamf Pro Database Knowledge Base article.

Step 6: Allocate Additional Memory to Tomcat

Note: To accommodate a large number of computers in Jamf Pro, it may be necessary to allocate additional Java Virtual Machine (JVM) memory to Tomcat. If there are other services running on your server, make sure to leave enough memory to accommodate them when configuring the `--max-memory` setting with Jamf Pro Server Tools.

1. Open Terminal.
2. Set the minimum Tomcat memory by executing the following command:

```
jamf-pro server config set --min-memory 256M
```

3. Set the maximum Tomcat memory by executing the following command:

```
jamf-pro server config set --max-memory 512M
```

Step 7: Configure the Firewall

On systems where the firewall is enabled, the following instructions will assist you in configuring the firewall to allow inbound access on port 8443, which provides access to the Jamf Pro web application.

> **Note:** If the firewall is not enabled, the following commands are not necessary. However, it is highly recommended that you follow industry best practices for production web-accessible systems, including enabling the firewall. For more information about securing your Jamf Pro server, contact your Jamf account representative.

1. Check to see if the firewall is running by executing:

```
sudo ufw status
```

2. If the firewall is running, edit the firewall configuration to allow access to port 8443 by executing the following commands:

```
sudo ufw allow 8443/tcp
sudo ufw reload
```

Step 8: Install Jamf Pro

1. (Optional) Create a `/tmp/jamf` directory in which to temporarily store downloads, tools, and scripts for Jamf Pro by executing:

```
mkdir /tmp/jamf
```

2. (Optional) Copy the `ROOT.war` file from the Jamf Pro manual download archive to the `/tmp/jamf` directory that you just created.
3. Stop the Tomcat service by executing:

```
sudo systemctl stop tomcat
```

or

```
sudo service tomcat stop
```

4. Move the default Tomcat web app out of the way by executing:

```
sudo mv /opt/tomcat/webapps/ROOT /opt/tomcat/webapps/TOMCAT
```

5. Copy the Jamf Pro web app archive to the Tomcat `webapps` directory by executing a command similar to the following:

```
sudo cp /tmp/jamf/ROOT.war /opt/tomcat/webapps/
```

6. Assign appropriate access to the `ROOT.war` by executing:

```
sudo chown tomcat:tomcat /opt/tomcat/webapps/ROOT.war
sudo chmod 750 /opt/tomcat/webapps/ROOT.war
```

7. Generate a keystore to enable SSL for Tomcat by executing a command similar to the following with the `keypass`, `storepass`, `dname`, and `validity` values customized for your environment:

```
sudo keytool -genkey -alias tomcat -keyalg RSA -keypass
"changeit" -storepass "changeit" -dname "CN=jamf.mycompany.com,
OU=Jamf IT, O=Jamf, L=Minneapolis, ST=MN, C=US"
-keystore /opt/tomcat/keystore -validity <numdays>
```

Make sure to enter the following attributes as appropriate to your site:

Attribute	Value	Example
CN=	Fully qualified domain name of the server	jamf.mycompany.com
OU=	Organizational unit	Jamf IT
O=	Organization	Jamf
L=	Location (city or office)	Minneapolis
ST=	State, province, or county	MN
C=	Country or region	US

8. Back up the Tomcat `server.xml` configuration file by executing:

```
sudo cp /opt/tomcat/conf/server.xml /opt/tomcat/conf/server.xml.bak
```

9. Open the `server.xml` configuration file in a text editor with a command similar to:

```
sudo vi /opt/tomcat/conf/server.xml
```

10. Locate the following comment section in the file (note that this is the `Connector on port 8443` not the `Connector on port 8443 with HTTP/2` section):

```
<!-- Define a SSL/TLS HTTP/1.1 Connector on port 8443
        This connector uses the NIO implementation.
        ...
-->
```

11. Replace the commented out (initial <!-- and trailing -->) Connector tag immediately following the comment (shown in "a" below) with the Connector tag text shown in "b" below:
a.

```
<!-- Define a SSL/TLS HTTP/1.1 Connector on port 8443
     ...
-->
<!--
<Connector port="8443" ...
  ...
</Connector>
-->
```

b.

```
<Connector URIEncoding="UTF-8"
           server="Apache Tomcat"
           port="8443"
           executor="tomcatThreadPool"
           SSLEnabled="true"
           maxPostSize="-1"
           scheme="https"
           protocol="org.apache.coyote.http11.Http11Nio2Protocol"
           sslImplementationName="org.apache.tomcat.util.net.jsse.
JSSEImplementation"
           secure="true">
  <SSLHostConfig sslProtocol="TLS"
                 protocols="TLSv1.2"
                 honorCipherOrder="true"
                 certificateVerification="none"
                 ciphers="TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384,
                          TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384,
                          TLS_ECDH_RSA_WITH_AES_256_GCM_SHA384,
                          TLS_ECDH_ECDSA_WITH_AES_256_GCM_SHA384,
                          TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256,
                          TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256,
                          TLS_ECDH_RSA_WITH_AES_128_GCM_SHA256,
                          TLS_ECDH_ECDSA_WITH_AES_128_GCM_SHA256,
                          TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384,
                          TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384,
                          TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA,
                          TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA,
                          TLS_ECDH_RSA_WITH_AES_256_CBC_SHA384,
                          TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA384,
                          TLS_ECDH_RSA_WITH_AES_256_CBC_SHA,
                          TLS_ECDH_ECDSA_WITH_AES_256_CBC_SHA,
                          TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256,
                          TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256,
                          TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA,
                          TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA,
                          TLS_ECDH_RSA_WITH_AES_128_CBC_SHA256,
                          TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA256,
                          TLS_ECDH_RSA_WITH_AES_128_CBC_SHA,
                          TLS_ECDH_ECDSA_WITH_AES_128_CBC_SHA" >
    <Certificate type="RSA"
                 certificateKeystoreFile="/opt/tomcat/keystore"
                 certificateKeystorePassword="changeit" />
  </SSLHostConfig>
</Connector>
```

12. Before saving the file, ensure the `keystorePass` value is set to the value you used for `storepass` when creating the keystore.
13. Save the `server.xml` file.

14. Start the Tomcat service by executing:

```
sudo systemctl start tomcat
```

or

```
sudo service tomcat start
```

The Tomcat service will automatically unpack the ROOT.war file into a ROOT directory in the webapps directory.

Step 9: Connect to the Jamf Pro Server

1. Configure the database connection settings using Jamf Pro Server Tools GUI or CLI. For instructions, see the Editing the Database Connection Using Jamf Pro Server Tools Knowledge Base article.
2. Access Jamf Pro by opening a web browser and typing the protocol, IP address or hostname of the server, and port. For example: https://jamf.mycompany.com:8443/

jamf | PRO

Setup

Setting Up Jamf Pro

The first time you connect to the Jamf Pro server, the Jamf Pro Setup Assistant guides you through the following setup tasks:

- Accept the license agreement.
- Enter your activation code.
- Create your first Jamf Pro user account.
- Enter your Jamf Pro URL.
  The Jamf Pro URL is the URL that client applications, computers, and mobile devices will connect to when communicating with the Jamf Pro server.

After you complete the Jamf Pro Setup Assistant, you can click the setup tips that are displayed onscreen to start configuring commonly used settings.

You may also want to make changes to the following pre-configured settings to ensure they meet the needs of your organization. These settings are important because over time, they can significantly affect the size of your database and your levels of network traffic:

- **"Update Inventory" policy—**Determines how often computers submit inventory to Jamf Pro.
  For more information, see "Computer Inventory Collection" in the *Jamf Pro Administrator's Guide*.
- **Recurring check-in frequency—**Determines the interval at which computers check in with Jamf Pro for available policies.
  For more information, see "Recurring Check-in Frequency" in the *Jamf Pro Administrator's Guide*.
- **Mobile device inventory collection frequency—**Determines how often mobile devices submit inventory to Jamf Pro.
  For more information, see "Mobile Device Inventory Collection Settings" in the *Jamf Pro Administrator's Guide*.

Related Information

For related information, see the following Knowledge Base article:

Network Ports Used by Jamf Pro
Learn about the network ports that you may need to configure when setting up Jamf Pro.

Jamf Pro User Accounts and Groups

Jamf Pro is a multi-user application. Jamf Pro user accounts and groups allow you to grant different privileges and levels of access to each user.

When configuring a Jamf Pro user account or group, you can grant access to the full Jamf Pro or to a specific site. You can grant privileges by choosing one of the following privilege sets:

- **Administrator**—Grants all privileges.
- **Auditor**—Grants all read privileges.
- **Enrollment Only**—Grants all privileges required to enroll computers and mobile devices.
- **Custom**—Requires you to grant privileges manually. For a Custom user account or group to have access to a particular function, privileges may need to be granted for multiple objects. For example, to create a mobile device configuration profile, the user needs privileges for both "Mobile Devices" and "Mobile Device Configuration Profiles".

If there are multiple users that should have the same access level and privileges, you can create a group with the desired access level and privileges and add accounts to it. Members of a group inherit the access level and privileges from the group. Adding an account to multiple groups allows you to grant a user access to multiple sites.

There are two ways to create Jamf Pro user accounts and groups: you can create standard accounts or groups, or you can add them from an LDAP directory service.

Important: It is recommended that you have at least one account that is not from an LDAP directory service in case the connection between the Jamf Pro server and the LDAP server is interrupted.

The Jamf Pro User Accounts and Groups settings also allow you to do the following:

- Configure account preferences for each Jamf Pro user account.
- Configure the password settings in the Password Policy for all standard Jamf Pro user accounts.
- Unlock a Jamf Pro user account that is locked.

Requirements

To add accounts or groups from an LDAP directory service, you need an LDAP server set up in Jamf Pro. For more information, see "Integrating with LDAP Servers" in the *Jamf Pro Administrator's Guide*.

Creating a Jamf Pro User Group

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **System Settings**.
4. Click **Jamf Pro User Accounts & Groups**  .
5. Click **New**  .
6. Do one of the following:
   - To create a standard Jamf Pro user group, select **Create Standard Group** and click **Next**.
   - To add a Jamf Pro user group from an LDAP directory service, select **Add LDAP Group** and click **Next**. Then follow the onscreen instructions to search for and add the group.
7. Use the Group pane to configure basic settings for the group.
8. If you chose "Custom" from the **Privilege Set** pop-up menu, click the **Privileges** tab and select the checkbox for each privilege that you want to grant the group.
9. Click **Save**.

Creating a Jamf Pro User Account

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **System Settings**.
4. Click **Jamf Pro User Accounts & Groups**  .
5. Click **New**  .
6. Do one of the following:
   - To create a standard Jamf Pro user account, select **Create Standard Account** and click **Next**.
   - To add a Jamf Pro user account from an LDAP directory service, select **Add LDAP Account** and click **Next**. Then follow the onscreen instructions to search for and add the account.
7. On the Account pane, enter information about the account as needed.

8. Choose an access level from the **Access Level** pop-up menu:
   - To grant full access to Jamf Pro, choose “Full Access”.
   - To grant access to a site, choose “Site Access”.

     Note: The “Site Access” option is only displayed if there are sites in Jamf Pro.For more information on adding sites to Jamf Pro, see "Sites" in the *Jamf Pro Administrator's Guide*.

   - To add the account to a standard group, choose “Group Access”.

     Note: The “Group Access” option is only displayed if there are standard groups in Jamf Pro. For more information on creating groups, see Creating a Jamf Pro User Group.

9. Do one of the following:
   - If you granted the account full access or site access, choose a privilege set from the **Privilege Set** pop-up menu. Then, if you chose “Custom”, click the **Privileges** tab and select the checkbox for each privilege that you want to grant the account.
   - If you added the account to a group, click the **Group Membership** tab and select the group or groups you want to add the account to.
10. Click **Save**.

Configuring Account Preferences

You can configure Language & Region and Search preferences for each Jamf Pro user account. Language & Region preferences allow you to configure settings such as date format and time zone. Search preferences allow you to configure settings for computer, mobile device, and user searches.

1. Log in to Jamf Pro.
2. At the top of the page, click the account settings  icon and then click **Account Preferences**.
3. Click the **Language & Region** tab and use the pop-up menus to configure language and region preferences.
4. Click the **Search Preferences** tab and use the pop-up menus to configure search preferences.

   Note: The default search preference is “Exact Match”. For most items, the option can be changed to either “Starts with” or “Contains”.

5. Click **Save**.

Configuring the Password Policy

The Password Policy in Jamf Pro allows you to configure the password settings. The Password Policy applies to all standard Jamf Pro user accounts. You can configure the following password settings:

- Number of login attempts allowed before a Jamf Pro user is locked out of the account
- Password length and age
- Password reuse limitations
- Password complexity
- Settings to allow a user to unlock their own account

Note: The settings configured in the Password Policy do not apply to Jamf Pro user accounts added from an LDAP directory service.

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings** .
3. Click **System Settings**.
4. Click **Jamf Pro User Accounts & Groups** .
5. Click **Password Policy**.
6. Click **Edit**.
7. Use the settings on the pane to specify the password settings.
8. Click **Save**.

The settings are applied immediately.

Unlocking a Jamf Pro User Account

A Jamf Pro user could be locked out of their account if they exceed the specified number of allowed login attempts. If the Password Policy is configured to allow the user to unlock their account, the user can reset their password to unlock their account. In this case, an email is immediately sent to the email address associated with the account in Jamf Pro allowing the user to unlock their account by resetting their password. For an email to be sent, an SMTP server must be set up in Jamf Pro. For more information, see Integrating with an SMTP Server.

In addition, a Jamf Pro user account that is locked can be manually unlocked from Jamf Pro by another Jamf Pro user with the Administrator privilege set.

The access status of the account is displayed as "Disabled" in Jamf Pro until the account is unlocked.

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings** .
3. Click **System Settings**.
4. Click **Jamf Pro User Accounts & Groups** .
   A list of Jamf Pro user accounts and groups is displayed.

5. Click the Jamf Pro user account that has an access status of “Disabled”, which means the account is locked.
6. Click **Edit**.
7. Choose “Enabled” from the **Access Status** pop-up menu to unlock the account.
8. Click **Save**.

The Jamf Pro user account is unlocked immediately.

Related Information

For related information, see the following section in the *Jamf Pro Administrator’s Guide*:

"Sites"
Learn about sites and how to add them to Jamf Pro.

Activation Code

The Activation Code settings in Jamf Pro allow you to update the activation code for your license. You can also change the organization name associated with the license and view licensing information.

Updating the Activation Code

Every time you receive a new activation code, it must be updated in Jamf Pro.

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **System Settings**.
4. Click **Activation Code**  .
5. Click **Edit**.
6. Enter the new activation code.
7. Click **Save**.

Integrating with an SMTP Server

Integrating with an SMTP server allows you to do the following:

- Send email notifications to Jamf Pro users when certain events occur. For more information, see "Email Notifications" in the *Jamf Pro Administrator's Guide*.
- Send enrollment invitations via email.
- Send mass emails to end users.

To integrate with an SMTP server, you need to configure the SMTP Server settings in Jamf Pro.

Configuring the SMTP Server Settings

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **System Settings**.
4. Click **SMTP Server**  .
5. Click **Edit**.
6. Configure the settings on the pane.
7. Click **Save**.

Testing the SMTP Server Settings

Once the SMTP Server settings are configured, you can send a test email from Jamf Pro.

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **System Settings**.
4. Click **SMTP Server**  .
5. Click **Test**.
6. Enter a test email address and click **Test** again.

A message displays, reporting whether or not the email was sent successfully.

Related Information

For related information, see the following sections in the *Jamf Pro Administrator's Guide*:

- "Email Notifications"
  Learn about the different email notifications that can be sent to Jamf Pro users.
- "User-Initiated Enrollment for Computers"
  Find out how to send computer enrollment invitations via email.
- "User-Initiated Enrollment for Mobile Devices"
  Find out how to send mobile device enrollment invitations via email.
- "Performing Mass Actions for Computers"
  Find out how to send a mass email to computer users.
- "Performing Mass Actions for Mobile Devices"
  Find out how send a mass email to mobile device users.

Change Management

Change Management allows you to track the changes that happen in Jamf Pro, such as the creation of a Jamf Pro user account. The Change Management settings in Jamf Pro allow you to log those changes to a log file (JAMFChangeManagement.log) on the Jamf Pro host server and log the changes to a syslog server.

The Change Management logs can also be viewed in Jamf Pro. The information displayed includes:

- Date/time the change took place
- Username of the administrator who made the change
- Object type (such as a Jamf Pro user account)
- Object name (such as the username of a Jamf Pro user account)
- Action (such as “Created”)
- Details about the change

In addition, you can view the changes to a specific object in that object’s history. For more information, see "Viewing the History of a Jamf Pro Object" in the *Jamf Pro Administrator's Guide*.

Requirements

To log changes to a log file, the account used to run Tomcat must have write permissions for the directory where the `JAMFChangeManagement.log` file is located.

Configuring the Change Management Settings

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **System Settings**.
4. Click **Change Management**  .
5. Click **Edit**.
6. Configure the settings on the pane.
7. Click **Save**.

Viewing Change Management Logs in Jamf Pro

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **System Settings**.
4. Click **Change Management**  .
5. Click **Logs**.
   The Change Management logs are displayed on the pane.
6. Do one of the following:
   - To view the object associated with a change, click the object in the Object Name column.
   - To view details about the change, click **Details** in the Details column.

GSX Connection

The GSX Connection settings allow you to integrate Jamf Pro with Apple's Global Service Exchange (GSX) to look up and populate the following purchasing information for computers and mobile devices:

- Purchase date
- Warranty expiration date

Note: GSX may not always return complete purchasing information. Only the information found in GSX is returned.

To integrate Jamf Pro with GSX, you must first create a GSX account and obtain a certificate from Apple. Then you can configure the GSX Connection settings in Jamf Pro, which involves entering GSX account information, retrieving an API token from Apple, and uploading the Apple certificate.

You can also use Jamf Pro to test the GSX connection and upload a renewed Apple certificate when needed.

Requirements

To configure the GSX Connection settings, you need:

- A GSX account with the "Manager" role, access to Web Services, and access to coverage/warranty information
- An Apple certificate (.pem or .p12)

For instructions on creating a GSX account and obtaining an Apple certificate, see the Integrating with Apple's Global Service Exchange (GSX) Knowledge Base article.

Configuring the GSX Connection Settings

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **Global Management**.
4. Click **GSX Connection**  .
5. Click **Edit**.
6. Select **Enable Connection to GSX**.

Note: This setting and others on this pane may already be configured if Jamf Pro was used to generate a CSR.

7. Enter the username and account number, including the leading zeros, for the GSX account.
8. Log in to your Apple GSX account, retrieve the API token, and then enter it in the **API Token** field in Jamf Pro.

   Note: The API token is not displayed after you finish configuring the GSX connection or when you edit an existing GSX connection. This is because the API token changes with every request and will always be different.

9. In the Certificate-based Authentication section, click **Upload**.
10. The **URI** field will be populated automatically.
11. Follow the onscreen instructions to upload the Apple certificate (.pem or .p12).

Testing the GSX Connection

After the GSX Connection settings are configured, you can test the connection to verify it works.

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **Global Management**.
4. Click **GSX Connection**  .
5. Click **Test**.
6. Click **Test** again.

A message displays, reporting the success or failure of the connection.

A successful connection will display information similar to the following:

```
[Accept: application/json, Content-Type: application/json, X-Apple-
SoldTo: 0000000000, X-Apple-ShipTo: 0000000000] GET https://partner-
connect.apple.com/gsx/api/authenticate/check HTTP/1.1

Response: OK
```

Renewing the Apple Certificate

You can use Jamf Pro to upload a renewed Apple certificate without removing the existing certificate so the connection with GSX is not lost. A notification is displayed 31 days prior to the expiration date of the Apple certificate.

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **Global Management**.
4. Click **GSX Connection**  .
5. Click **Edit.**
6. Click **Renew**.
7. Follow the onscreen instructions to upload a renewed Apple certificate.

Related Information

For related information, see the following sections in the *Jamf Pro Administrator's Guide*:

- "Performing Mass Actions for Computers"
  Find out how to mass look up and populate purchasing information from GSX.
- "Performing Mass Actions for Mobile Devices"
  Find out how to mass look up and populate purchasing information from GSX.
- "Viewing and Editing Inventory Information for a Mobile Device"
  You can look up and populate purchasing information for a single mobile device by editing the device's inventory information in Jamf Pro.
- "Viewing and Editing Inventory Information for a Computer"
  You can look up and populate purchasing information for a single computer by editing the computer's inventory information in Jamf Pro.
- "Local Enrollment Using Recon"
  Find out how to look up and populate purchasing information when enrolling a computer by running Recon locally.
- "Remote Enrollment Using Recon"
  Find out how to look up and populate purchasing information when enrolling a computer by running Recon remotely.

Jamf Pro Summary

The Jamf Pro Summary is a custom report that can be useful for troubleshooting Jamf Pro issues, and for providing information to Jamf for purposes of support or license renewal.

By default, the Jamf Pro Summary includes the following information:

- Number of managed and unmanaged computers
- Number of managed mobile devices
- Operating system on the Jamf Pro host server
- Path to the Jamf Pro web app
- Apache Tomcat version
- Information about the version of Java installed on the Jamf Pro host server
- Information about the MySQL connection and configuration

You can also add information to the Jamf Pro Summary from the following categories as needed:

- Computers
- Mobile Devices
- Users
- System Settings
- Server Infrastructure
- Global Management
- Computer Management
- Computer Management–Management Framework
- Mobile Device Management
- User Management
- Network Organization
- Database

You can view the Jamf Pro Summary in a browser window or send the Jamf Pro Summary to Jamf.

Requirements

To send the Jamf Pro Summary to Jamf, you need a valid Jamf Nation account.

To create a Jamf Nation account, go to:
https://www.jamf.com/jamf-nation/users/new

Viewing the Jamf Pro Summary

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **Jamf Pro Information**.
4. Click **Jamf Pro Summary**  .
5. Select the checkboxes next to the items you want to include.
6. Click **Create**.
   The Jamf Pro Summary displays in a browser window.
7. Click the **Back** button in the web browser to return to the Jamf Pro Summary pane.

Sending the Jamf Pro Summary to Jamf

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **Jamf Pro Information**.
4. Click **Jamf Pro Summary**  .
5. Select the checkboxes next to the items you want to include.
6. Click **Send Summary to Jamf**.
7. Enter your Jamf Nation credentials, and then click **Send**.

   The Jamf Pro Summary is sent to Jamf via Jamf Nation.

Related Information

For related information about Customer Experience Metrics (CEM), see the following Knowledge Base article:

Customer Experience Metrics
Learn about Customer Experience Metrics and how to configure the setting in your Jamf Pro environment.

For related information about Customer Experience Metrics, visit the following webpage:
https://www.jamf.com/products/jamf-pro/customer-experience-metrics/

jamf | PRO

Database Management

Backing Up the Database

You can create database backups as needed or schedule automated database backups using Jamf Pro Server Tools.

Note: The time it takes to create a backup depends on the size of the database.

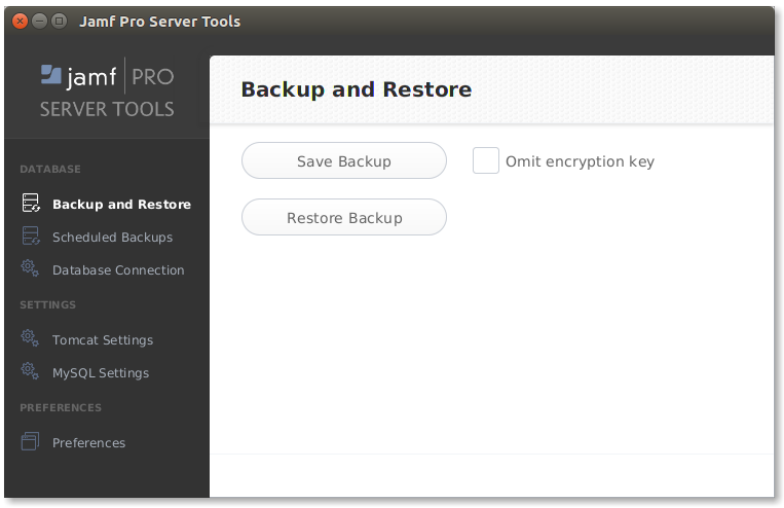
Creating a Database Backup

1. Open the Jamf Pro Server Tools GUI by performing the following steps:
   a. Open a command terminal and enter the following but do not press Enter:

   ```
   java -jar
   ```

   b. Drag the Jamf Pro Server Tools .jar file into the window. This will add the .jar file path to the `java -jar` command.
   c. Press Enter.
2. Click **Database Connection** in the sidebar.
3. Configure the settings to match your database configuration, and then click **Test Connection**.
   - If successful, the message "Successfully Connected" appears. Continue with Step 4 below.
   - If the connection is not successful, an error message will appear.
4. Click **Backup and Restore** in the sidebar.

5. Click **Save Backup**.



6. Choose a location to save the backup and click **Open**.

   Jamf Pro Server Tools saves the backup as a .sql.gz file.

Scheduling Database Backups

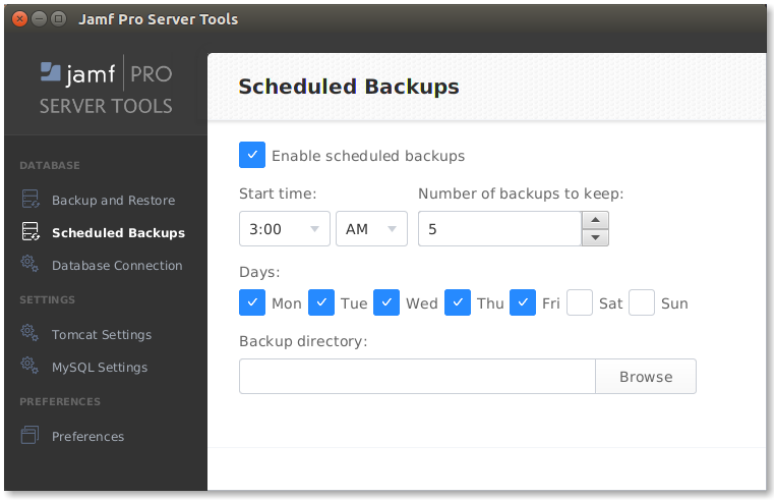
You can schedule database backups to occur on an ongoing basis. You can also automate the deletion of scheduled backup files that are older than a specified number of days.

1. Open the Jamf Pro Server Tools GUI by performing the following steps:
   a. Open a command terminal and enter the following but do not press Enter:

   ```
   java -jar
   ```

   b. Drag the Jamf Pro Server Tools .jar file into the window. This will add the .jar file path to the `java -jar` command.
   c. Press Enter.
2. Click **Scheduled Backups** in the sidebar.
3. Select the **Enable scheduled backups** checkbox.
4. If prompted, enter your Jamf Pro Server Tools configuration password.
5. Choose the hour and the days that you want backups to occur.

6. To automatically delete old backups, enter the number of most-recent backups that you want to keep in the **Backups limit** field. All older backup files will be deleted when the scheduled backups run. To retain all backups, enter "0".
7. To save the backups in a custom location, click **Browse** and select a new location. It is recommended that you store the backups on a separate drive.

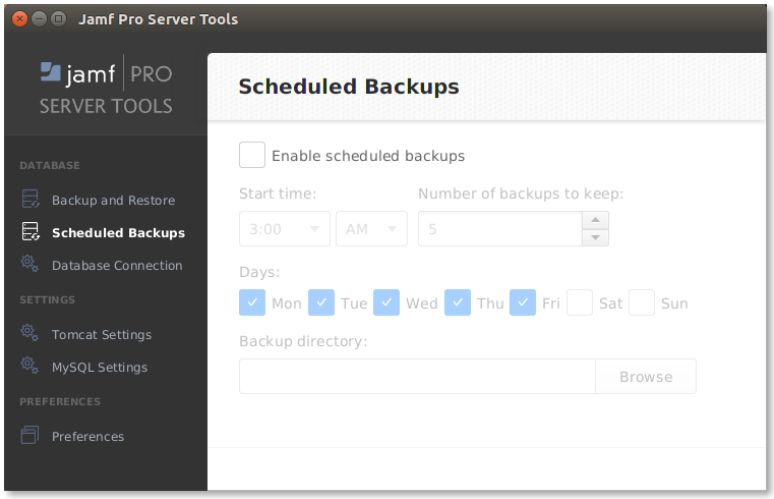


Stopping Scheduled Database Backups

1. Open the Jamf Pro Server Tools GUI by performing the following steps:
   a. Open a command terminal and enter the following but do not press Enter:

   ```
   java -jar
   ```

   b. Drag the Jamf Pro Server Tools .jar file into the window. This will add the .jar file path to the `java -jar` command.
   c. Press Enter.
2. Click **Scheduled Backups** in the sidebar.
3. Deselect the **Enable scheduled backups** checkbox.



Jamf Pro Server Tools immediately stops creating scheduled backups.

Related Information

For related information, see the following Knowledge Base articles:

- Jamf Pro Server Tools Overview
- Backing Up and Restoring the Database Using the Jamf Pro Server Tools Command-Line Interface

Restoring Database Backups

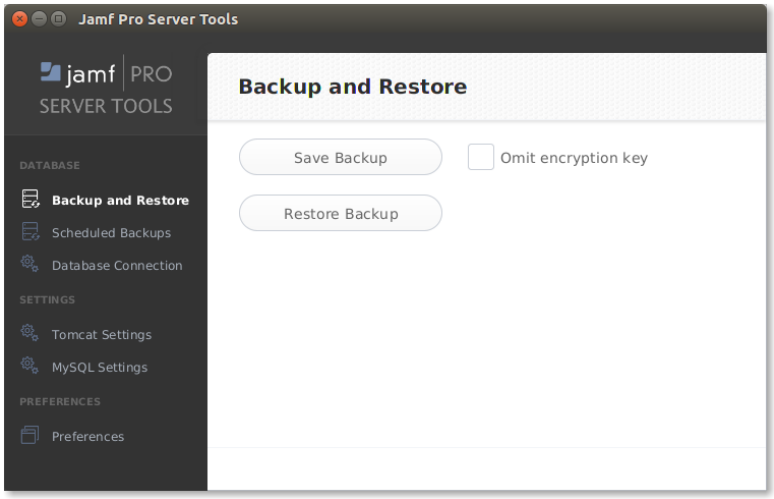
If you need to revert to an earlier version of your database, you can restore a database backup using Jamf Pro Server Tools.

Note: You must stop Tomcat before you restore a database backup.

1. Open the Jamf Pro Server Tools GUI by performing the following steps:
   a. Open a command terminal and enter the following but do not press Enter:

   ```
   java -jar
   ```

   b. Drag the Jamf Pro Server Tools .jar file into the window. This will add the .jar file path to the `java -jar` command.
   c. Press Enter.
2. Click **Tomcat Settings** in the sidebar.
3. Click **Stop Tomcat**.
4. Click **Backup and Restore** in the sidebar.
5. Click **Restore Backup Now**.



6. Select the backup file that you want to restore, and click **Open**.
7. Click **Tomcat Settings** in the sidebar.
8. Click **Start Tomcat**.

The existing database is replaced with the database backup that you selected.

Related Information

For related information, see the following Knowledge Base articles:

- Jamf Pro Server Tools Overview
- Backing Up and Restoring the Database Using the Jamf Pro Server Tools Command-Line Interface

Viewing the Status of Database Tables

MySQL database tables can become corrupt if the database was not shut down properly or if the Jamf Pro host server is too slow to manage the number of computers in your organization. You can view the status of database tables right from Jamf Pro.

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **Jamf Pro Information**.
4. Click **Database Table Summary**  .

jamf | PRO

Server Infrastructure

About Distribution Points

Distribution points are servers used to host files for distribution to computers and mobile devices. The following types of files can be distributed from a distribution point using Jamf Pro:

- Packages
- Scripts
- In-house apps
- In-house books

Jamf Pro supports two types of distribution points:

- File share distribution points
- A cloud distribution point

You can use any combination of these types of distribution points.

By default, the first distribution point you add to Jamf Pro is the master distribution point. The master distribution point is used by all other distribution points as the authoritative source for all files during replication. You can change the master distribution point at any time.

Note: On computers with macOS 10.15 or later that do not have an MDM profile, you must use an HTTP, HTTPS, or cloud distribution point to install packages.

When planning your distribution point infrastructure, it is important to understand the differences between each type of distribution point. The following table explains the key differences:

	File Share Distribution Point	Cloud Distribution Point
Description	Standard server that is configured to be a distribution point	Distribution point that uses one of the following content delivery networks (CDNs) to host files: ▪ Rackspace Cloud Files ▪ Amazon Web Services ▪ Akamai
Maximum Number per Jamf Pro Instance	Unlimited	One

	File Share Distribution Point	Cloud Distribution Point
Server /Platform Requirements	Any server with an Apple Filing Protocol (AFP) or Server Message Block (SMB) share **Note:** File share distribution points cannot be mounted and hosted on the same server.	None
Protocol	AFP, SMB, HTTP, or HTTPS	HTTPS
Ports	▪ AFP: 548 ▪ SMB: 139 ▪ HTTP: 80 ▪ HTTPS: 443	443

Authentication Options	▪ AFP or SMB:
	  ▪ No authentication
	  ▪ Username and password
	▪ HTTP or HTTPS:
	  ▪ No authentication
	  ▪ Username and password
		None

Files that Can Be Hosted	Packages	▪ Packages ▪ In-house apps ▪ In-house books
Parent-Child Capabilities	No	No
File Replication Method	Replication to file share distribution points must be initiated from Jamf Admin.	Replication to a cloud distribution point must be initiated from Jamf Admin.
Selective Replication	Not available when replicating to file share distribution points.	Available when replicating to a cloud distribution point if the master distribution point is a file share distribution point. The files for replication must be specified in Jamf Pro and the replication initiated from Jamf Admin.

Related Information

For related information, see the following sections in this guide:

- File Share Distribution Points
  Find out how to manage file share distribution points in Jamf Pro.
- Cloud Distribution Point
  Find out how to manage the cloud distribution point.

File Share Distribution Points

A server with an AFP or SMB share can be used as a file share distribution point. Before you can use a file share distribution point with Jamf Pro, you must set up the distribution point and add it to Jamf Pro.

Note: A server with an AFP share cannot share files on the Apple File System (APFS), which is the default file system for computers with macOS 10.13 or later. Computers with macOS 10.13 or later that are HFS+ formatted can still support AFP. If you need a file share distribution point for APFS formatted computers, SMB is an option.

For more information on APFS and SMB, see the following Apple macOS Deployment Reference: https://support.apple.com/guide/deployment-reference-macos/welcome/web

For information on setting up a file share distribution point, see the Setting Up a File Share Distribution Point Knowledge Base article.

When you add a file share distribution point to Jamf Pro, you can do the following:

- Make it the master distribution point.
- Choose a failover distribution point.
- Configure HTTP downloads.

Adding a File Share Distribution Point

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **Server Infrastructure**.
4. Click **File Share Distribution Points**  .
5. Click **New**  .
6. Use the General pane to configure basic settings for the distribution point.
7. Click the **File Sharing** tab and enter information about the AFP or SMB share.
8. (Optional) Click the **HTTP** tab and configure HTTP downloads.
9. Click **Save**.

Replicating Files to a File Share Distribution Point

During replication, all files on the master distribution point are replicated to the file share distribution point that you choose.

1. Open Jamf Admin and authenticate to the Jamf Pro server.
2. In the sidebar, select the file share distribution point you want to replicate files to.
3. Click **Replicate**.

Related Information

For related information, see the following section in the *Jamf Pro Administrator's Guide*:

"Network Segments"
You can use network segments to ensure that computers and mobile devices use the closest distribution point by default.

For related information, see the following Knowledge Base articles:

- Setting Up a File Share Distribution Point on Linux Using Samba
  Find out how to use Samba to set up a file share distribution point with an SMB share on a Linux server.
- Using Apache HTTP Server to Enable HTTP Downloads on a Linux File Share Distribution Point
  Find out how to use Apache HTTP Server to enable HTTP downloads on a Linux file share distribution point.
- Using IIS to Enable HTTPS Downloads on a Windows Server 2016 or 2019 File Share Distribution Point
  Find out how to activate Internet Information Services (IIS) and use it to enable HTTPS downloads on a Windows Server 2016 or 2019 file share distribution point.

Cloud Distribution Point

The cloud distribution point uses a content delivery network (CDN) to host packages, in-house apps, and in-house books. Jamf Pro supports the following content delivery services:

- Rackspace Cloud Files
- Amazon S3 or Amazon CloudFront
- Akamai NetStorage
- Jamf Cloud Distribution Service (JCDS)

When you configure the cloud distribution point in Jamf Pro, you can choose to make it the master. You can also choose whether to replicate specific files or the entire contents of the master distribution point if the master is a file share distribution point.

Note: If you plan to use the JCDS for your cloud distribution point, it is recommended that you do not attempt to upload files larger than 20 GB. Due to the file size download limit set by Amazon CloudFront, files larger than 20 GB may not download successfully. For more information, see the following website:
https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/cloudfront-limits.html

Jamf Pro supports the use of signed URLs created with Amazon CloudFront. It also supports Akamai Remote Authentication. For more information about signed URLs created with CloudFront, see the following website:
http://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/private-content-signed-urls.html

For more information about Akamai Remote Authentication, contact your Akamai Account Manager.

Requirements

If you plan to use Akamai for your cloud distribution point, Akamai must be configured to use File Transfer Protocol (FTP).

Note: If you have upgraded from Jamf Pro 8.x, you must migrate the scripts and packages on your master distribution point before configuring the cloud distribution point. For more information, see the Migrating Packages and Scripts Knowledge Base article.

Files that are uploaded to a cloud distribution point cannot have filenames that include the following characters :

/ : ? < > \ * | " [ ] @ ! % ^ #

Configuring the Cloud Distribution Point

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **Server Infrastructure**.
4. Click **Cloud Distribution Point**  .
5. Click **Edit**.
6. Choose a content delivery network from the **Content Delivery Network** pop-up menu.
7. Configure the settings on the pane.
8. Click **Save**.

Testing the Cloud Distribution Point

Once the cloud distribution point is configured, you can test the connection to the content delivery network.

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **Server Infrastructure**.
4. Click **Cloud Distribution Point**  .
5. Click **Test**.
6. Click **Test** again.

A message displays, reporting the success or failure of the connection.

Replicating Files to the Cloud Distribution Point

During replication, files on the master distribution point are replicated to the cloud distribution point via Jamf Admin. The files that are replicated depend on whether the cloud distribution point is configured to replicate specific files or the entire contents of the master.

1. Open Jamf Admin and authenticate to the Jamf Pro server.
2. In the sidebar, select the cloud distribution point you want to replicate files to.
3. Click **Replicate**.

Related Information

For related information, see the following section in the *Jamf Pro Administrator's Guide*:

"Network Segments"
You can use network segments to ensure that computers and mobile devices use the closest distribution point by default. For related information, see the following Knowledge Base article:

Information Required to Configure a Cloud Distribution Point in Jamf Pro
Learn about the information that must be obtained from your cloud services provider to configure the cloud distribution point in Jamf Pro.

For more information about content delivery services, visit the following websites:

- Rackspace Cloud Files
  http://www.rackspace.com/cloud/files/
- Amazon S3
  http://aws.amazon.com/s3/
- Amazon CloudFront
  http://aws.amazon.com/cloudfront/
- Akamai NetStorage
  http://www.akamai.com/html/solutions/netstorage.html
- Jamf Cloud Distribution Service
  http://www.jamfsoftware.com/products/jamf-cloud/

Jamf Infrastructure Manager Instances

A Jamf Infrastructure Manager instance is a service that is managed by Jamf Pro. It can be used to host the following:

- **LDAP Proxy**—This allows traffic to pass securely between Jamf Pro and an LDAP directory service.
  The Infrastructure Manager and the LDAP Proxy typically reside within the DMZ.
  The LDAP Proxy requires integration with an LDAP directory service.
  For more information, see "LDAP Proxy" in the *Jamf Pro Administrator's Guide*.
- **Healthcare Listener**—This allows traffic to pass securely from a healthcare management system to Jamf Pro.
  For more information, see "Healthcare Listener" in the *Jamf Pro Administrator's Guide*.

When you install an instance of the Infrastructure Manager, Jamf Pro allows you to enable the LDAP Proxy or the Healthcare Listener. Infrastructure Manager instances can be installed on Linux and Windows.

For more information, see the *Jamf Infrastructure Manager Installation Guide*.

Viewing Inventory Information for a Jamf Infrastructure Manager Instance

Jamf Pro displays the following inventory information for each Infrastructure Manager instance:

- Last Check-in
- IP Address at Last Check-in
- Operating System
- Operating System Version

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings** .
3. Click **Server Infrastructure.**
4. Click **Infrastructure Managers** .
   A list of Infrastructure Manager instances is displayed along with the services that are installed on each instance.
5. Click the Infrastructure Manager instance you want to view inventory information for.

Further Considerations

- When editing an Infrastructure Manager instance, only the display name and recurring check-in frequency can be changed.

  Note: The default check-in frequency at which the Infrastructure Manager instance checks in with Jamf Pro is 30 seconds.

- An Infrastructure Manager instance cannot be deleted if there are dependencies for the Infrastructure Manager. For example, an Infrastructure Manager cannot be deleted if there is an LDAP Proxy hosted on it. To delete the Infrastructure Manager, you must first disable the LDAP Proxy.
- If a Healthcare Listener is hosted on the Infrastructure Manager, the Healthcare Listener is deleted when the Infrastructure Manager is deleted.

Related Information

For related information, see the following section in the *Jamf Pro Administrator's Guide*:

"Email Notifications"
Learn how to enable an email notification in the event that an Infrastructure Manager instance does not check in with Jamf Pro.

jamf | PRO

Advanced Configuration

SSL Certificate

Jamf Pro requires a valid SSL certificate to ensure that computers and mobile devices communicate with the Jamf Pro server and not an imposter server.

The Apache Tomcat settings in Jamf Pro allow you to create an SSL certificate from the certificate authority (CA) that is built into Jamf Pro. You can also upload the certificate keystore for an SSL certificate that was obtained from an internal CA or a trusted third-party vendor.

Note: If your environment is hosted in Jamf Cloud, the Apache Tomcat settings are managed by Jamf Cloud and are not accessible.

Requirements

To create or upload an SSL certificate, Jamf Pro must be installed as the "ROOT" web app, and the user running the Tomcat process must have read/write access to Tomcat's `server.xml` file.

Creating or Uploading an SSL Certificate

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **System Settings**.
4. Click **Apache Tomcat Settings**  .
5. Click **Edit**.
6. Select **Change the SSL certificate used for HTTPS** and click **Next**.
7. Follow the onscreen instructions to upload or create an SSL certificate.
8. Restart Tomcat for the changes to take effect.
   For instructions on how to restart Tomcat, see the following Knowledge Base article:
   Starting and Stopping Tomcat

Related Information

For related information, see the following Knowledge Base article:

Using OpenSSL to Create a Certificate Keystore for Tomcat
Find out how to use OpenSSL to create a certificate keystore that you can upload to Jamf Pro.

Configuring Tomcat to Work with a Load Balancer

When Jamf Pro is behind a load balancer, you must configure the remote IP valve, proxy port, and scheme in Tomcat's `server.xml` file. The Load Balancing settings in Jamf Pro allow you to configure these settings without having to edit the `server.xml` file manually.

Requirements

To configure Load Balancing settings using Jamf Pro, Jamf Pro must be installed as the "ROOT" web app, and the user running the Tomcat process must have read/write access to Tomcat's `server.xml` file.

Configuring Load Balancing Settings

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **System Settings**.
4. Click **Apache Tomcat Settings**  .
5. Click **Edit**.
6. Select **Configure Tomcat for working behind a load balancer** and click **Next**.
7. Follow the onscreen instructions to configure the Load Balancing settings.
8. Restart Tomcat for the changes to take effect.
   For instructions on how to restart Tomcat, see the following Knowledge Base article:
   Starting and Stopping Tomcat

Tomcat Thread Pool Settings

Configuring the Tomcat Thread Pool settings using Jamf Pro allows you to make modifications to Tomcat's `server.xml` file without having to edit it manually.

Requirements

To configure Tomcat Thread Pool settings using Jamf Pro, Jamf Pro must be installed as the "ROOT" web app, and the user running the Tomcat process must have read/write access to Tomcat's `server.xml` file.

Configuring Tomcat Thread Pool Settings

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **System Settings**.
4. Click **Apache Tomcat Settings**  .
5. Click **Edit**.
6. Select **Update the settings for Tomcat's thread pool** and click **Next**.
7. Follow the onscreen instructions to configure the Thread Pool settings.
8. Restart Tomcat for the changes to take effect.
   For instructions on how to restart Tomcat, see the following Knowledge Base article:
   Starting and Stopping Tomcat

Jamf Pro Web App Memory

Jamf Pro allows you to view the amount of memory being used by the web app. If you need to change the amount of memory allocated to the web app, you can use Jamf Pro Server Tools.

Viewing Memory Usage

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings** .
3. Click **Jamf Pro Information**.
4. Click **Memory Usage** .
   Current free and used memory values are displayed.

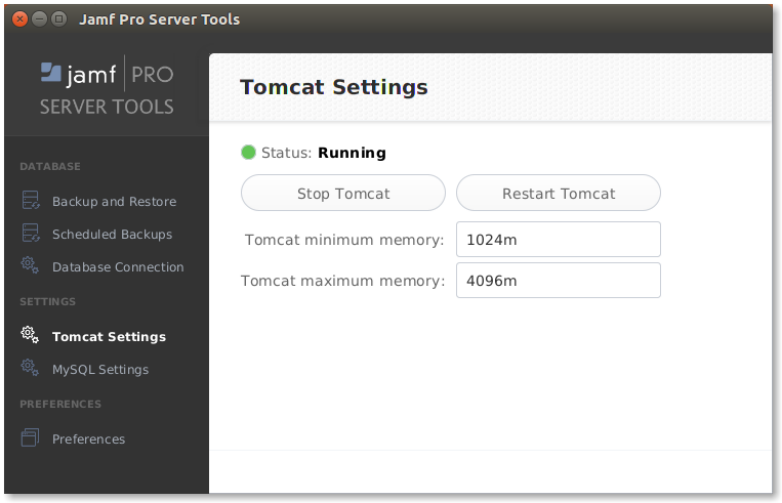
Configuring Web App Memory Using Jamf Pro Server Tools

1. Open Jamf Pro Server Tools on the Jamf Pro host server by executing:

```
sudo java -jar /usr/local/jss/bin/server-tools-gui.jar
```

2. Select **Tomcat Settings**.
3. Enter values in the **Tomcat minimum memory** and **Tomcat maximum memory** fields to configure the amount of memory allocated to the web app.

> **Note:** Type an "m" after the memory value to specify megabytes, e.g., "256m".



4. Restart Tomcat. The changes will take effect after Tomcat restarts.

Configuring Web App Memory Using the System Command Line

1. Open the file that contains the Tomcat memory settings.
   - If you used the Jamf Pro Installer to install Jamf Pro, execute:

     ```
     sudo nano /usr/local/jss/tomcat/bin/setenv.sh
     ```

   - If you did not use the Jamf Pro Installer to install Jamf Pro, create the `setenv.sh` file in the following location:
     `/usr/local/jss/tomcat/bin/`
2. Append a custom JAVA_OPTS environment variable to the bottom of the `setenv.sh` file to configure the amount of memory allocated to the web app. For example, to allocate 1 GB of RAM, the variable should look something like this:

   ```
   JAVA_OPTS=" -Xmx1024M"
   ```

   Note: Custom settings will persist after performing an upgrade.
3. Save and close the file by pressing Control-O, and then Control-X.
4. Restart Tomcat. The changes will take effect after Tomcat restarts.
   For instructions on how to restart Tomcat, see the following Knowledge Base article:
   Starting and Stopping Tomcat

Clustering

A clustered environment is one that has multiple instances of the Jamf Pro web app pointing to the same database. Clustering is useful in large environments that require multiple web apps, or environments with a web app in the DMZ.

When setting up a clustered environment, it is recommended that you configure the Clustering settings in Jamf Pro using the web app that you plan to make the master, and then install other Jamf Pro web apps that point to the same database. However, if you already have multiple Jamf Pro web apps installed and pointed to the same database, you can configure the Clustering settings in Jamf Pro after the fact. For more information on setting up a clustered environment, contact your Jamf account representative.

The Clustering settings in Jamf Pro allow you to configure the frequency at which clustered web apps are synced with the database, and specify which web app should function as the master.

The master web app handles tasks such as upgrading the database schema and flushing logs from the database.

Jamf Pro also allows you to view a list of web apps that are pointed to the same database and information about them.

Requirements

To cluster web apps that are not in the DMZ, you need a load balancer with the address of the Jamf Pro server (formerly the Jamf Software Server). For example:

https://jss.mycompany.com:8443/

The load balancer should route traffic to the servers running the web app.

Configuring Clustering Settings

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings**  .
3. Click **System Settings**.
4. Click **Clustering**  .
5. Click **Edit**.
6. Configure the settings on the pane.
   To specify which web app should function as the master, select the **Master** option for the web app.

7. Click **Save**.
8. If you already have multiple Jamf Pro web apps pointed to the same database, restart Tomcat on any of the web apps for the changes to take effect.
   For instructions on how to restart Tomcat, see the Starting and Stopping Tomcat Knowledge Base article.

Related Information

For related information, see the following Knowledge Base articles:

Caching Configuration
Find out how to configure distributed caching for clustered Jamf Pro environments.

Installing a Jamf Pro Web App in the DMZ
Find out how to install a web app in the DMZ, and learn when in the process you should configure the Clustering settings in Jamf Pro.

Limited Access Settings

If you have a clustered environment, the Limited Access settings in Jamf Pro allow you to disable the Jamf Pro interface and limit the types of devices that can communicate with Jamf Pro. This is most commonly used if you have a web app in the DMZ.

For each Jamf Pro web app, you can choose one of the following Limited Access settings:

- Full Access
- Computer Access Only
- Mobile Device Access Only
- Computer and Mobile Device Access

Choosing anything other than "Full Access" disables the Jamf Pro interface.

Configuring the Limited Access Settings

1. Log in to any of the Jamf Pro web apps.
2. In the top-right corner of the page, click **Settings**  .
3. Click **System Settings**.
4. Click **Limited Access**  .
5. Select a setting for each Jamf Pro web app as needed.
6. Click **Save**.

Flushing Logs

Flushing logs reduces the size of the database and can speed up searches. You can flush the following types of logs:

- Application Usage logs
- Computer Usage logs
- Policy logs
- Jamf Remote logs
- Screen sharing logs
- Jamf Imaging logs
- Computer and mobile device management history
- Computer inventory reports (computer inventory information from past inventory submissions)
- Mobile device inventory reports (mobile device inventory information from past inventory submissions)
- Jamf Pro access logs
- Change Management logs
- Event logs

You can schedule log flushing to take place daily, or you can manually flush logs as needed. You can also choose to flush logs that are older than a certain number of days, weeks, or months.

For information on the types of data flushed with each log and the database tables affected, see the Data and Tables Affected by Log Flushing Knowledge Base article.

Scheduling Log Flushing

1. Log in to Jamf Pro.
2. In the top-right corner of the page, click **Settings** .
3. Click **System Settings**.
4. Click **Log Flushing** .
5. Click **Edit**.
6. Use the pop-up menus to choose the number of days, weeks, or months after which each type of log should be flushed.
7. Choose a time of day from the **Time to Flush Logs Each Day** pop-up menu.
8. Click **Save**.

Manually Flushing Logs

1. Log in to any of the Jamf Pro web apps.
2. In the top-right corner of the page, click **Settings**  .
3. Click **System Settings**.
4. Click **Log Flushing**  .
5. Click **Flush Manually**.
6. Select the checkbox for each type of log you want to flush.
7. From the **Flush Logs Older Than** pop-up menu, choose the number of days, weeks, or months after which logs should be flushed.
8. Click **Flush**.

   A message displays, reporting the success or failure of the flush.

Related Information

For related information, see the following sections in the *Jamf Pro Administrator's Guide*:

- "Viewing and Flushing Policy Logs for a Computer"
  Find out how to view and flush policy logs for a computer.
- "Viewing and Flushing Logs for a Policy"
  Find out how to view and flush logs for a policy.
- "Viewing the History for a Computer"
  Find out how to view the logs and the management history for a computer.
- "Viewing the Management History for a Mobile Device"
  Find out how to view the management history for a mobile device.

Migrating to Another Server

1. Back up the existing Jamf Pro database.
   For more information, see Backing Up the Database.
2. Ensure that the new server meets the requirements for the Jamf Pro Installer, and then follow the instructions in Installing Jamf Pro Using the Installer to install the required software (if needed) and create the Jamf Pro database.
3. Copy the Jamf Pro Installer to the new server.
4. Install Jamf Pro by launching the installer and following the onscreen instructions.
   For more information, see Installing Jamf Pro Using the Installer.
5. Copy the database backup to the new server, and then use Jamf Pro Server Tools to restore the backup.
   For more information, see Restoring Database Backups.
6. Re-upload or create the SSL certificate.
   For more information, see SSL Certificate.
7. Update the DNS entry to point to the new server's IP address.

   Note: If you can't change the DNS entry, you must change the Jamf Pro URL and re-enroll all mobile devices and computers.