

Environment Setup for Releases

Introduction

This page describes what is necessary to configure your environment for an Apache Tomcat release build. Many of these steps only need to be taken a single time, but you may need to review them when something changes such as versions of toolchain components, or updating signing key(s), etc.

Prerequisites

- A git client installed and on your path
- A Subversion client installed and on your path
- Apache Ant installed and on your path (see BUILDING.txt in the root of the code repository for version requirements)
- The latest release of the minimum Java version that the Tomcat version runs on installed and on your path
 - Java 17 is currently the minimum version of Java that can be used for Apache Tomcat 8.5.x and 9.0.x builds, though the minimum version of Java to *run* the released binaries is still Java 7 and Java 8, respectively.
- Operating system
 - Windows just works
 - Linux also requires Wine (standard package manager version should be fine)
 - MacOS also requires Wine (see below)
- GnuPG installed
- A public key that is part of the Apache web of trust
- A configuration file for DigiCert to properly-sign the release artifacts
- A reasonable internet connection (you will need to upload ~100MB)

Platform-Specific Notes

Windows

Installation of the necessary support packages can be performed on a Windows 10 Virtual Machine from [Microsoft's Edge Development resources](#). Download + unpack the archive for your VM software and launch the VM. Login, open a PowerShell.exe window as Administrator, and installed [Chocolatey](#):

```
PS C:\Users\IEUser> Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))
```

Once Chocolatey is installed, you can install all the above prerequisites at once:

```
PS C:\Users\ISUser> choco install git svn adoptopenjdk11 ant gnupg sed xsltproc
```

This command will run for a while, and ask you repeatedly if it's okay to run "chocolateyInstall.ps1", which you will have to do to proceed. Once the above command has completed, you have all the software prerequisites installed and on your PATH. Well, once you quit PowerShell and launch a new CMD, EXE or PowerShell window, of course 😊

MacOS

12.x (Monterey), 13.x (Ventura), and 14.x (Sonoma)

- Install [homebrew](#) if you haven't already.
- Install Wine 9.0 using `brew install --no-quarantine wine-stable` ([ref](#))
- If you are using an Apple Silicon-based Mac, you will need to `/usr/sbin/softwareupdate --install-rosetta --agree-to-license` ([ref](#)) to be able to run the NSIS installer-generator.

11.x.x (Big Sur) - old instructions

- Install [homebrew](#) if you haven't already.
- Install [wine-crossover](https://github.com/Gcenx/homebrew-wine) from <https://github.com/Gcenx/homebrew-wine>
- Configure a 32-bit wine environment using:
 - `WINEARCH=win32 WINEPREFIX=~/.wine32 winecfg`
- Then before you start the release ensure the following environment variables are set:
 - `export WINEARCH=win32`
 - `export WINEPREFIX=~/.wine32`

Linux

- Install the Wine package from your package manager. On Debian Bookworm, this required:
 - `dpkg --add-architecture i386`
 - `apt-get install wine wine32`
- Configure a Wine32 environment using:
 - `WINEARCH=win32 WINEPREFIX=~/.wine32 winecfg`
- Then before you start the release ensure the following environment variables are set:
 - `export WINEARCH=win32`

- `export WINEPREFIX=~/.wine32`