

Packaging Tomcat for Linux Distributions

...

Coty Sutherland



Coty Sutherland

Software Engineer, Red Hat
JBoss Web Server Project

Supported Tomcat, httpd, and
JBoss EAP/JBossWeb for ~3
years

ASF Tomcat committer since
late 2016

ASF Member

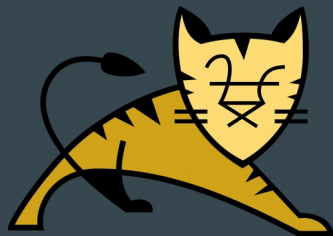
Fedora tomcat and tomcat-
native package co-maintainer
since 2015



Red Hat

Agenda

- Distribution Overview
- Why use Distributions?
- The Apache Tomcat Distribution
- The Fedora Distribution
- The Ubuntu Distribution
- The Docker Container Distribution
- Discussion/Questions



Distribution Overview

Apache Tomcat, Fedora, Ubuntu, Dockerized Tomcat

Most Tomcat distributions follow the same basic recipe to create their packages:

1. Download the source package provided by the ASF distribution (or the binary, Docker and other cases)
2. Build Tomcat, and any extras needed
3. Add OS specific things such as service scripts and wrappers Tomcat
4. Distribute the resulting package via the OS' package manager (e.g. yum)



Why Use Distributions?

PROS

- Easy and reliable way to install, configure, and update/remove packages
- Potential for automation using system tools
- Standardized installation
- Standardized method to start and stop daemons
- Easily utilize other libraries on the system, such as log4j
- System stability provided by targeted bug and security fixes



Why Use Distributions?

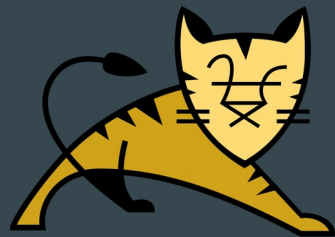
CONS

- Bugs in wrapper components or service scripts that are distribution specific
- Difficult for developers to understand/use
- Added complexity to configure (compared to the ASF distribution)
- Requires some administration skills specific to the distribution



The Apache Tomcat Distribution

- The Apache Tomcat team distributes an archive containing the pre-built Tomcat jars and for Windows we also include the executables you need to install Tomcat as a service.
- We provide extra things in the archive distributions:
 - The API documentation (docs webapp)
 - Tomcat Client Deployer
 - Embedded Tomcat JARs (used for running Tomcat in-process)



The Fedora Distribution

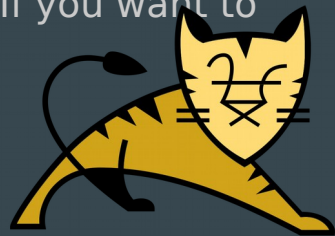
Fedora 30

- What is an RPM package?
- Packaging breakdown, what's with all the sub-packages?
- Directory Layout
- update-alternatives for API JARs
- Installing and Configuring Tomcat
- Starting and Stopping Tomcat
- Updating and Removing Tomcat
- Contributing



What is an RPM Package?

- The RPM format is the software packaging format used by Red Hat flavors of Linux. Everything you need to know about an RPM is in its spec file
- A spec file is a plain text file that contains information about the package, instructions for the package builder (rpmbuild) to build and structure the package, and instructions for the package manager to carry out upon install/uninstall/update (if needed)
- The tomcat spec for Fedora Rawhide is [here](#). Let's take a look...
 - Note that the tomcat spec is very complex compared to others. If you want to see a simpler spec file, then take a look at Fedora's guide to [Creating RPM Packages](#).



Packaging Breakdown

What's with all the Tomcat sub-packages?

- Tomcat is broken up into sub-packages to allow users to install the minimum number of packages required to run the daemon.
- Here is a list including a short description of all of the Tomcat RPMs that are available in Fedora 30 (ignoring ones that are not sub-

```
[csutherl@localhost ~]$ dnf search tomcat | grep ^tomcat | egrep -v '(taglibs|tomcatjss|tomcat-native)' | uniq
Last metadata expiration check: 0:00:39 ago on Sun 08 Sep 2019 12:36:16 PM EDT.
tomcat.noarch : Apache Servlet/JSP Engine, RI for Servlet 4.0/JSP 2.3 API
tomcat-lib.noarch : Libraries needed to run the Tomcat Web container
tomcat-docs-webapp.noarch : The docs web application for Apache Tomcat
tomcat-javadoc.noarch : Javadoc generated documentation for Apache Tomcat
tomcat-jsvc.noarch : Apache jsvc wrapper for Apache Tomcat as separate service
tomcat-webapps.noarch : The ROOT and examples web applications for Apache Tomcat
tomcat-jsp-2.3-api.noarch : Apache Tomcat JavaServer Pages v2.3 API
tomcat-servlet-4.0-api.noarch : Apache Tomcat Java Servlet v4.0 API
tomcat-jsp-2.3-api.noarch : Apache Tomcat JavaServer Pages v2.3 API
tomcat-servlet-4.0-api.noarch : Apache Tomcat Java Servlet v4.0 API
tomcat-el-3.0-api.noarch : Apache Tomcat Expression Language v3.0 API
tomcat-admin-webapps.noarch : The host-manager and manager web applications for
```



Directory Layout

Where do all the things go?

- The Tomcat RPMs follow the FHS standard which determines where things go. Using FHS makes all packages installed via RPM easy for admins to manage by putting everything in the same structure
- For Tomcat, /usr/share/tomcat is \$CATALINA_HOME and includes symlinks to the FHS standard locations for other things

```
[csutherl@localhost ~]$ sudo ls -l /usr/share/tomcat
total 4
drwxr-xr-x. 2 root root 4096 Sep  3 10:01 bin
lrwxrwxrwx. 1 root tomcat 11 Aug  6 13:31 conf -> /etc/tomcat
lrwxrwxrwx. 1 root tomcat 22 Aug  6 13:31 lib -> /usr/share/java/tomcat
lrwxrwxrwx. 1 root tomcat 15 Aug  6 13:31 logs -> /var/log/tomcat
lrwxrwxrwx. 1 root tomcat 22 Aug  6 13:31 temp -> /var/cache/tomcat/temp
lrwxrwxrwx. 1 root tomcat 23 Aug  6 13:31 webapps -> /var/lib/tomcat/webapps
lrwxrwxrwx. 1 root tomcat 22 Aug  6 13:31 work -> /var/cache/tomcat/work
```



Update-alternatives for API jars

* This is a relic and will be removed at some point.

- Back to the API sub-packages...
- The concept of alternatives on Red Hat Linux systems is pretty easy to grasp. Basically it creates a link that's easily configurable to the reference binary/library installed on the system
- Example: Tomcat provides the reference implementation for the Servlet API on Red Hat flavors of Linux. Each time it is installed or updated the alternatives link is updated to reflect that
- Here is what the servlet link looks like. It points references for `/etc/alternatives/servlet` to the correct JAR location:

```
[csutherl@localhost ~]$ ll /etc/alternatives/servlet
lrwxrwxrwx. 1 root root 42 Aug 16 10:14 /etc/alternatives/servlet -> /usr/share/java/tomcat-servlet-4.0-api.jar
```

Installing and Configuring Tomcat

- To install Tomcat on a Red Hat Linux machine use the Yellowdog Updater, Modified (or simply 'yum') package manager. All you need is the name of the package to be installed (e.g. tomcat), and you can install it such as:
 - `$ yum install tomcat`
- The 'tomcat' parent package needs a few other packages (dependencies), which are automatically installed: 'tomcat-lib', 'tomcat-el-2.2-api', 'tomcat-jsp-2.2-api', and 'tomcat-servlet-3.0-api'
- This is a minimal installation, which means no webapps are included. If you need the admin webapps (for example), execute the following to install them:
 - `$ yum install tomcat-admin-webapps`



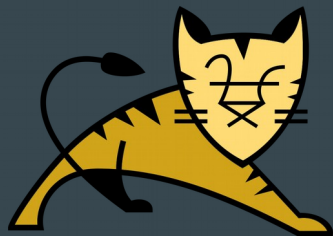
Installing and Configuring tomcat, Cont'd.

- Configuring Tomcat is done the same as the ASF distribution with the addition of two files, `/etc/sysconfig/tomcat` and `/etc/tomcat/tomcat.conf`. These files are optional, but they give you similar capabilities as `setenv`. For example, you can enable the `SECURITY_MANAGER` in either of these
- Generally `tomcat.conf` is used as the main configuration with `/etc/sysconfig/tomcat` allowing you to override things set in `tomcat.conf` on a per-service level (for multi-instance installs). Instructions for how to setup a multi-instance install are in the configuration file's comments



Starting and Stopping the Tomcat Service

- Using the system provided scripts make it very easy to start, stop, and check the status of the Tomcat daemon. They make it very easy for users to enable Tomcat to start on boot too!
- Examples on next slide
- A note about Fedora tomcat's logging (excluding EPEL 6)...
 - After moving from sysV to the systemd service unit method stdout/stderr no longer log to catalina.out. The events that recently logged to catalina.out now log to the system's journal. If users need to see stdout/stderr logging, check the journal (`journalctl -u tomcat`)



Start/Stop/Status Example

```
[csutherl@localhost ~]$ sudo systemctl start tomcat
```

```
[csutherl@localhost ~]$ sudo systemctl stop tomcat
```

```
[csutherl@localhost ~]$ sudo systemctl status tomcat
```

```
● tomcat.service - Apache Tomcat Web Application Container
```

```
   Loaded: loaded (/usr/lib/systemd/system/tomcat.service; disabled; vendor preset: disabled)
```

```
   Active: inactive (dead)
```

```
Sep 08 12:48:30 localhost.localdomain systemd[1]: Stopping Apache Tomcat Web Application Container...
```

```
Sep 08 12:48:30 localhost.localdomain server[17195]: 08-Sep-2019 12:48:30.420 INFO [Thread-4] org.apache.coyote.Abst>
```

```
Sep 08 12:48:30 localhost.localdomain server[17195]: 08-Sep-2019 12:48:30.428 INFO [Thread-4] org.apache.coyote.Abst>
```

```
Sep 08 12:48:30 localhost.localdomain server[17195]: 08-Sep-2019 12:48:30.434 INFO [Thread-4] org.apache.catalina.co>
```

```
Sep 08 12:48:30 localhost.localdomain server[17195]: 08-Sep-2019 12:48:30.437 INFO [Thread-4] org.apache.coyote.Abst>
```

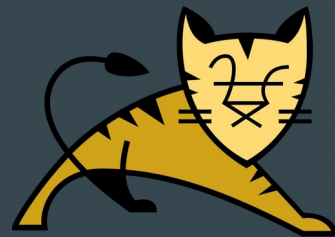
```
Sep 08 12:48:30 localhost.localdomain server[17195]: 08-Sep-2019 12:48:30.439 INFO [Thread-4] org.apache.coyote.Abst>
```

```
Sep 08 12:48:30 localhost.localdomain server[17195]: 08-Sep-2019 12:48:30.440 INFO [Thread-4] org.apache.coyote.Abst>
```

```
Sep 08 12:48:30 localhost.localdomain server[17195]: 08-Sep-2019 12:48:30.441 INFO [Thread-4] org.apache.coyote.Abst>
```

```
Sep 08 12:48:30 localhost.localdomain systemd[1]: tomcat.service: Succeeded.
```

```
Sep 08 12:48:30 localhost.localdomain systemd[1]: Stopped Apache Tomcat Web Application Container.
```



Updating and Removing Tomcat

- Both updating and removing Tomcat are very simple. The syntax for updating a package on Fedora is shown below along with how to remove a package
- Note that when updating the tomcat parent package all sub-packages are updated, but because of how the dependencies are configured removing it requires the '*' to get all the sub-packages
 - `$ sudo yum update tomcat`
 - `$ sudo yum remove tomcat-*`
- Updating with the package manager is one of the big benefits of using an OS' package over the ASF distribution. When you run update command it basically installs everything again leaving configuration files in place so you do not need to configure the service again



Contributing

A Quick Note

- Generally pretty quiet as far as new bugs
- I update around once a month, depending on ASF tomcat and tomcat-native releases. The links below are for the package pages of tomcat and tomcat-native. If you're interested in contributing or learning more about the packages, please take a look:
 - <https://apps.fedoraproject.org/packages/tomcat/overview/>
 - <https://apps.fedoraproject.org/packages/tomcat-native/overview/>



The Ubuntu Distribution

Tomcat Deb Packaging

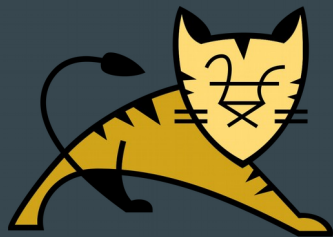
Ubuntu 19.04 (Disco)

- What is a DEB package?
- Packaging breakdown, what's with all the sub-packages?
- Directory layout
- Compared Fedora
- Installing and configuring Tomcat
- Starting and stopping Tomcat
- Updating and removing Tomcat
- Contributing



What is a DEB Package?

- The deb format is the software packaging format that the Debian distribution and its derivatives (including Ubuntu) use. These packages are archives which contain three files, debian-binary, control.tar.gz, and data.tar.gz. Two of them are effectively meta-data, while the third contains the installable artifacts
- If you want to inspect any of the .deb packages, you can download them with apt-get and use dpkg to list its contents:



```

root@coty-Standard-PC-Q35-ICH9-2009:/tmp# apt-get download tomcat9
Get:1 http://us.archive.ubuntu.com/ubuntu disco/universe amd64 tomcat9 all 9.0.16-3 [35.8 kB]
Fetched 35.8 kB in 0s (256 kB/s)
root@coty-Standard-PC-Q35-ICH9-2009:/tmp# dpkg -c tomcat9_9.0.16-3_all.deb
drwxr-xr-x root/root          0 2019-02-26 03:31 ./
drwxr-xr-x root/root          0 2019-02-26 03:31 ./etc/
drwxr-xr-x root/root          0 2019-02-26 03:31 ./etc/cron.daily/
-rwxr-xr-x root/root        538 2019-02-05 04:11 ./etc/cron.daily/tomcat9
drwxr-xr-x root/root          0 2019-02-26 03:31 ./etc/logrotate.d/
drwxr-xr-x root/root          0 2019-02-26 03:31 ./etc/rsyslog.d/
-rw-r--r-- root/root        381 2019-02-05 04:11 ./etc/rsyslog.d/tomcat9.conf
drwxr-xr-x root/root          0 2019-02-26 03:31 ./etc/tomcat9/
drwxr-xr-x root/root          0 2019-02-26 03:31 ./etc/tomcat9/Catalina/
drwxr-xr-x root/root          0 2019-02-26 03:31 ./etc/tomcat9/policy.d/
-rw-r--r-- root/root       2192 2019-02-05 04:11 ./etc/tomcat9/policy.d/01system.policy
-rw-r--r-- root/root        330 2019-02-05 04:11 ./etc/tomcat9/policy.d/02debian.policy
-rw-r--r-- root/root       3237 2019-02-05 04:11 ./etc/tomcat9/policy.d/03catalina.policy
-rw-r--r-- root/root       5435 2019-02-05 04:11 ./etc/tomcat9/policy.d/04webapps.policy
-rw-r--r-- root/root       2040 2019-02-05 04:11 ./etc/tomcat9/policy.d/50local.policy
drwxr-xr-x root/root          0 2019-02-26 03:31 ./lib/
drwxr-xr-x root/root          0 2019-02-26 03:31 ./lib/systemd/
drwxr-xr-x root/root          0 2019-02-26 03:31 ./lib/systemd/system/
-rw-r--r-- root/root        986 2019-02-26 03:30 ./lib/systemd/system/tomcat9.service
drwxr-xr-x root/root          0 2019-02-26 03:31 ./usr/
drwxr-xr-x root/root          0 2019-02-26 03:31 ./usr/lib/
drwxr-xr-x root/root          0 2019-02-26 03:31 ./usr/lib/sysusers.d/
-rw-r--r-- root/root       277 2019-02-05 04:11 ./usr/lib/sysusers.d/tomcat9.conf
drwxr-xr-x root/root          0 2019-02-26 03:31 ./usr/libexec/
drwxr-xr-x root/root          0 2019-02-26 03:31 ./usr/libexec/tomcat9/
-rwxr-xr-x root/root        523 2019-02-05 04:11 ./usr/libexec/tomcat9/tomcat-start.sh
-rwxr-xr-x root/root        658 2019-02-05 04:11 ./usr/libexec/tomcat9/tomcat-update-policy.sh
drwxr-xr-x root/root          0 2019-02-26 03:31 ./usr/share/
drwxr-xr-x root/root          0 2019-02-26 03:31 ./usr/share/doc/
drwxr-xr-x root/root          0 2019-02-26 03:31 ./usr/share/doc/tomcat9/

```

Packaging Breakdown

What's with all the Tomcat sub-packages?

- Ubuntu follows the same idea that minimalism is good, so it also includes some tomcat sub-packages that can be installed as needed
- Here is a list of the Ubuntu sub-packages for tomcat:

```
root@coty-Standard-PC-Q35-ICH9-2009:/tmp# apt list | grep tomcat9

WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

libtomcat9-embed-java/disco,disco 9.0.16-3 all
libtomcat9-java/disco,disco,now 9.0.16-3 all [installed,automatic]
tomcat9-admin/disco,disco 9.0.16-3 all
tomcat9-common/disco,disco,now 9.0.16-3 all [installed,automatic]
tomcat9-docs/disco,disco 9.0.16-3 all
tomcat9-examples/disco,disco 9.0.16-3 all
tomcat9-user/disco,disco 9.0.16-3 all
tomcat9/disco,disco,now 9.0.16-3 all [installed]
```



Directory Layout

Where do all the things go?

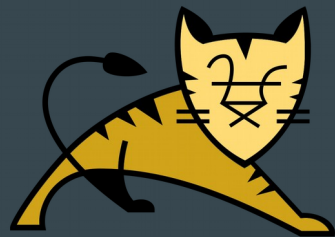
- Debian follows the FHS also,
<https://wiki.debian.org/FilesystemHierarchyStandard>
- \$CATALINA_HOME in Debian's tomcat9 package is /var/lib/tomcat9 and you can see the links to other FHS-compliant directories:

```
root@coty-Standard-PC-Q35-ICH9-2009:~# ls -l /var/lib/tomcat9
total 12
lrwxrwxrwx 1 root  root   12 Feb 26  2019 conf -> /etc/tomcat9
drwxr-xr-x 2 tomcat tomcat 4096 Feb 26  2019 lib
lrwxrwxrwx 1 root  root   17 Feb 26  2019 logs -> ../../log/tomcat9
drwxr-xr-x 2 root  root  4096 Sep  8 12:56 policy
drwxrwxr-x 3 tomcat tomcat 4096 Sep  8 12:56 webapps
lrwxrwxrwx 1 root  root   19 Feb 26  2019 work -> ../../cache/tomcat9
```



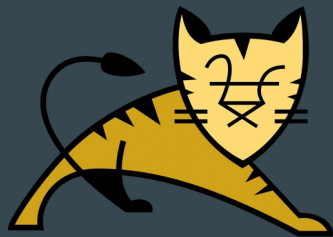
Compared to Fedora

- Some similarities between Ubuntu and Fedora:
 - Both are Linux systems and follow the FHS
 - Both system's distributions are easily manageable by using the system's package manager (yum for Fedora and apt for Debian/Ubuntu)
- One notable difference is despite the shared method of starting and stopping the service, Ubuntu still uses the sysV implementation instead of a systemd service unit for tomcat, so it hasn't yet migrated completely to systemd (at least not in Disco)



Installing and Configuring Tomcat

- Installing the tomcat package on Ubuntu is just as easy as Fedora's installation process:
 - `$ apt install tomcat9`
- And installing sub-packages works the same way:
 - `$ apt install tomcat9-admin`
- Configuring Tomcat in Ubuntu is similar to how you would do it with the ASF distribution. All of the scripts are in the bin directory, `/usr/share/tomcat9/bin/`



Starting and Stopping the Tomcat Service

This syntax is the same as Fedora's:

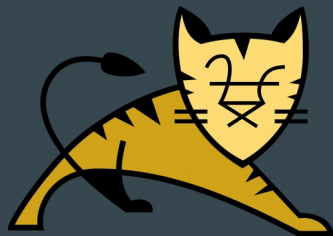
```
root@coty-Standard-PC-Q35-ICH9-2009:~# systemctl start tomcat9
root@coty-Standard-PC-Q35-ICH9-2009:~# systemctl stop tomcat9
root@coty-Standard-PC-Q35-ICH9-2009:~# systemctl status tomcat9
● tomcat9.service - Apache Tomcat 9 Web Application Server
   Loaded: loaded (/lib/systemd/system/tomcat9.service; enabled; vendor preset: enabled)
   Active: inactive (dead) since Sun 2019-09-08 13:05:38 EDT; 2s ago
     Docs: https://tomcat.apache.org/tomcat-9.0-doc/index.html
   Process: 6385 ExecStart=/bin/sh /usr/libexec/tomcat9/tomcat-start.sh (code=exited, status=143)
   Main PID: 6385 (code=exited, status=143)

Sep 08 12:56:39 coty-Standard-PC-Q35-ICH9-2009 tomcat9[6385]: Deployment of web application directory [/
Sep 08 12:56:39 coty-Standard-PC-Q35-ICH9-2009 tomcat9[6385]: Starting ProtocolHandler ["http-nio-8080"]
Sep 08 12:56:39 coty-Standard-PC-Q35-ICH9-2009 tomcat9[6385]: Server startup in [2,076] milliseconds
Sep 08 13:05:38 coty-Standard-PC-Q35-ICH9-2009 systemd[1]: Stopping Apache Tomcat 9 Web Application Serv
Sep 08 13:05:38 coty-Standard-PC-Q35-ICH9-2009 tomcat9[6385]: Pausing ProtocolHandler ["http-nio-8080"]
Sep 08 13:05:38 coty-Standard-PC-Q35-ICH9-2009 tomcat9[6385]: Stopping service [Catalina]
Sep 08 13:05:38 coty-Standard-PC-Q35-ICH9-2009 tomcat9[6385]: Stopping ProtocolHandler ["http-nio-80
Sep 08 13:05:38 coty-Standard-PC-Q35-ICH9-2009 tomcat9[6385]: Destroying ProtocolHandler ["http-nio-
Sep 08 13:05:38 coty-Standard-PC-Q35-ICH9-2009 systemd[1]: tomcat9.service: Succeeded.
Sep 08 13:05:38 coty-Standard-PC-Q35-ICH9-2009 systemd[1]: Stopped Apache Tomcat 9 Web Application
```



Updating and Removing Tomcat

- This is the same as Fedora also;
 - `$ sudo apt update tomcat9`
 - `$ sudo apt remove tomcat9-*`



Contributing

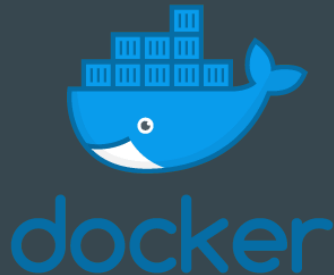
A very quick note...

- I'm not a maintainer of the Ubuntu tomcat package; if more information is required it is recommended to reach out directly to the Ubuntu maintainers, located on the package information page for tomcat9:
 - <https://packages.ubuntu.com/disco/tomcat9>



The Docker Container Distribution

- What is Docker?
- Dockerized Tomcat, a brief overview
- Installing and Configuring
- Starting and Stopping
- Updating
- Contributing



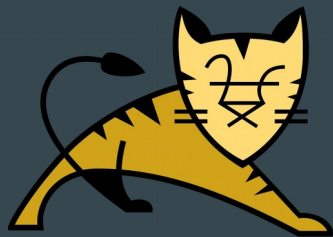
What is Docker?

From opensource.com: “Docker is a tool designed to make it easier to create, deploy, and run applications by using containers. Containers allow a developer to package up an application with all of the parts it needs, such as libraries and other dependencies, and ship it all out as one package. By doing so, thanks to the container, the developer can rest assured that the application will run on any other Linux machine regardless of any customized settings that machine might have that could differ from the machine used for writing and testing the code.”



Dockerized Tomcat, a [very] brief overview

- Information about the Tomcat Docker Container can be found on [DockerHub](#)
- Sources for the Dockerfile are on the [docker-library GitHub](#)
- Example [Dockerfile for Tomcat 9 with Java 8](#) quick look



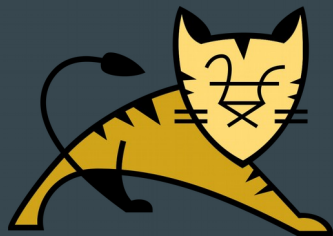
Installing and Configuring

- Installing can be done at the same as startup, but if you want to install the container image, you would issue a pull, such as:
 - `$ docker pull tomcat:latest`
- Configuration is nonexistent. Everything is configured in the Dockerfile ahead of time.



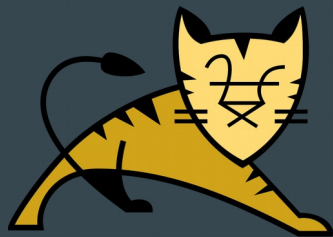
Starting and Stopping

- To start the vanilla (which isn't super useful) tomcat container, you can simply run ``$ docker run tomcat:latest`` and it will be downloaded and execute.
 - By default it runs in the foreground. To run in the background, like ``$ docker run -d -p 8081:8080 tomcat:latest``
- To stop, ctrl-C; the container runs with ``catalina.sh run`` by default.
 - To stop a background, ``$ docker stop $CONTAINER_NAME``. Container names are generated randomly (unless you specify one), so you'd have to find the container you want to stop using ``$ docker ps``



Updating

- Updating occurs in your Dockerfile, you pick the exact version you want.
- Once you chose a new version, just stop the container and start the new one.



Examples

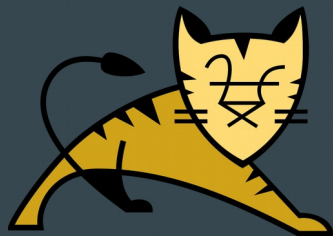
- We provide a Dockerfile in the ASF Tomcat sources, [res/tomcat-maven/Dockerfile](#).
- Small example that packages an application in a container and extends the tomcat Docker image.

```
[csutherl@localhost docker]$ docker image build -t tomcat-badapp ./
Sending build context to Docker daemon 17.41 kB
Step 1/3 : FROM tomcat:9.0
--> 8726a0509f5b
Step 2/3 : MAINTAINER coty
--> Using cache
--> d2e99e82a28b
Step 3/3 : COPY badapp/ /usr/local/tomcat/webapps/badapp/
--> Using cache
--> 8516328b3c2d
Successfully built 8516328b3c2d
[csutherl@localhost docker]$ docker container run -it --publish 8081:8080 tomcat-badapp
Using CATALINA_BASE:   /usr/local/tomcat
Using CATALINA_HOME:   /usr/local/tomcat
Using CATALINA_TMPDIR: /usr/local/tomcat/temp
Using JRE_HOME:        /usr/local/openjdk-11
Using CLASSPATH:        /usr/local/tomcat/bin/bootstrap.jar:/usr/local/tomcat/bin/tomcat-juli.jar
NOTE: Picked up JDK_JAVA_OPTIONS:  --add-opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/
d-opens=java.rmi/sun.rmi.transport=ALL-UNNAMED
```



Contributing

- Check out the project on [GitHub](#)!



Discussion

Questions?

- I'd like to know how distribution maintainers can better provide Tomcat to users
- What questions are left unanswered after this presentation?
- Does anyone use any other distributions (maybe ?) that you would like to see covered in a future talk?
- What are some problems caused by the packaged distribution (not tomcat bugs) that you're facing today?
- Reasons for/against using Tomcat from an OS distribution rather than the ASF distribution?



User Questions from Freenode #tomcat

Q: Looking at the `/usr/lib/systemd/system` directory, what is the difference between `tomcat.service` and `tomcat@.service`?

A: `tomcat@` is for named services and is generally only used if you have multiple instances of tomcat on a machine

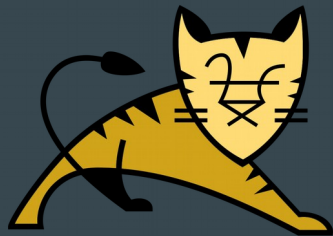
Q: Can someone explain what the file `/etc/logrotate.d/tomcat` does?

A: That file is a configuration file for the logrotate daemon which configures it to rotate `catalina.out`. Unfortunately, it's a bug because `catalina.out` isn't used after moving to systemd, so you can safely remove it (unless you're using `jsvc` which DOES log to `catalina`)



RPM Build/Install Demo

- Building an RPM with rpmbuild
- Installing the newly created RPMs (localinstall)
 - Would anyone like to see how to create a local repository and install from there?
- Enabling the Security Manager via configuration
- Starting, checking status, and stopping the service
- Removing the package(s)





Red Hat

THANK YOU!

Coty Sutherland



github.com/csutherl



linkedin.com/in/cotysutherland



twitter.com/cotysutherland