comment-installer-tomcat-avec-nginx-en-tant-que-proxy-inversé-sur-ubuntu-22-04

Apache Tomcat is an open-source Java web server and Servlet container used to host web applications written in Java. It is the first choice of web developers to build and maintain dynamic websites and applications based on the Java software platform. It is an open-source project developed by the Apache Software Foundation. Tomcat enables a webserver to handle dynamic Java-based web content.

This tutorial will show you how to install Apache Tomcat on Ubuntu 22.04.

Prerequisites

- A server running Ubuntu 22.04.
- A valid domain name pointed with your server IP.
- A root password is configured on the server.

Install Java JDK

Apache Tomcat is a Java-based application so Java must be installed on your server. If Java is not installed you can install it using the following command:

```
apt install default-jdk -y
```

Once Java is installed, you can verify the Java version using the following command:

```
java -version
```

You will get the following output:

```
openjdk version "11.0.15" 2022-04-19
OpenJDK Runtime Environment (build 11.0.15+10-Ubuntu-0ubuntu0.22.04.1)
OpenJDK 64-Bit Server VM (build 11.0.15+10-Ubuntu-0ubuntu0.22.04.1, mixed mode, sharing)
```

Install Apache Tomcat on Ubuntu 22.04

First, it is a good idea to run Tomcat as a separate user. You can create a Tomcat user with the following command:

```
useradd -m -d /opt/tomcat -U -s /bin/false tomcat
```

Next, download the latest version of Apache Tomcat with the following command:

```
wget https://dlcdn.apache.org/tomcat/tomcat-10/v10.0.20/bin/apache-tomcat-10.0.20.tar.gz
```

Once the Apache Tomcat is downloaded, extract the downloaded file inside the /opt directory:

```
tar xzvf apache-tomcat-10*tar.gz -C /opt/tomcat --strip-components=1
```

Next, set proper ownership and permission to the Tomcat directory:

```
chown -R tomcat:tomcat /opt/tomcat/
chmod -R u+x /opt/tomcat/bin
```

Once you are finished, you can proceed to the next step.

Create Tomcat Administrative User

By default, Tomcat can be accessed without any authentication. So it is recommended to enable the authentication and create an administrative user for Tomcat. You can add it by editing the Tomcat user configuration file:

```
nano /opt/tomcat/conf/tomcat-users.xml
```

Add the following lines above the line </tomcat-users>:

```
<role rolename="admin-gui" />
<user username="admin" password="yourpassword" roles="manager-gui,admin-gui" />
```

Save and close the file when you are finished.

Enable Tomcat Remote Access

By default, Tomcat is configured to access only from the local host. So it is recommended to enable the Tomcat remote access for managing Tomcat from the remote host.

To enable the manager app from a remote host, edit the following file:

```
nano /opt/tomcat/webapps/manager/META-INF/context.xml
```

Remove the following line:

```
  <Valve className="org.apache.catalina.valves.RemoteAddrValve"
         allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" />
```

To enable the host manager app from a remote host, edit the following file:

```
nano /opt/tomcat/webapps/host-manager/META-INF/context.xml
```

Remove the following line:

```
  <Valve className="org.apache.catalina.valves.RemoteAddrValve"
         allow="127\.\d+\.\d+\.\d+|::1|0:0:0:0:0:0:0:1" />
```

Save and close the file when you are finished.

Create a Service File for Apache Tomcat

Next, you will need to create a service file to manage the Tomcat service via systemd. You can create it using the following command:

```
nano /etc/systemd/system/tomcat.service
```

Add the following lines:

```
[Unit]
Description=Tomcat
After=network.target

[Service]
Type=forking

User=tomcat
Group=tomcat

Environment="JAVA_HOME=/usr/lib/jvm/java-1.11.0-openjdk-amd64"
Environment="JAVA_OPTS=-Djava.security.egd=file:///dev/urandom"
Environment="CATALINA_BASE=/opt/tomcat"
Environment="CATALINA_HOME=/opt/tomcat"
```

```
Environment="CATALINA_PID=/opt/tomcat/temp/tomcat.pid"
Environment="CATALINA_OPTS=-Xms512M -Xmx1024M -server -XX:+UseParallelGC"

ExecStart=/opt/tomcat/bin/startup.sh
ExecStop=/opt/tomcat/bin/shutdown.sh

[Install]
WantedBy=multi-user.target
```

Save and close the file then reload the systemd daemon to apply the changes:

```
systemctl daemon-reload
```

Next, start the Tomcat service and enable it to start at system reboot with the following command:

```
systemctl start tomcat
systemctl enable tomcat
```

You can check the status of the Apache Tomcat using the following command:

```
systemctl status tomcat
```

You will get the following output:

```
? tomcat.service - Tomcat
     Loaded: loaded (/etc/systemd/system/tomcat.service; disabled; vendor preset: enabled)
     Active: active (running) since Fri 2022-04-29 08:11:54 UTC; 6s ago
    Process: 18959 ExecStart=/opt/tomcat/bin/startup.sh (code=exited, status=0/SUCCESS)
   Main PID: 18966 (java)
      Tasks: 29 (limit: 4630)
     Memory: 116.4M
        CPU: 5.312s
     CGroup: /system.slice/tomcat.service
             ??18966 /usr/lib/jvm/java-1.11.0-openjdk-amd64/bin/java -Djava.util.logging.config.file=/opt/tomcat/conf/logging.properties -Dja>

Apr 29 08:11:54 ubuntu systemd[1]: Starting Tomcat...
Apr 29 08:11:54 ubuntu startup.sh[18959]: Tomcat started.
Apr 29 08:11:54 ubuntu systemd[1]: Started Tomcat.
```

At this point, Tomcat is started and listens on port 8080. You can check it with the following command:

```
ss -antpl | grep java
```

You will get the following output:

```
LISTEN 0      1      [::ffff:127.0.0.1]:8005            *:*    users:(("java",pid=18966,fd=53))
LISTEN 0      100                     *:8080            *:*    users:(("java",pid=18966,fd=43))
```

Once you are finished, you can proceed to the next step.

Configure Nginx as a Reverse Proxy for Tomcat

Next, you will need to create an Nginx as a reverse proxy for Apache Tomcat. First, install the Nginx web server with the following command:

```
apt-get install nginx -y
```

Once the Nginx web server is installed, create an Nginx virtual host configuration file using the following command:

```
nano /etc/nginx/conf.d/tomcat.conf
```

Add the following lines:

```
server {
  listen 80;

  server_name    tomcat.example.com;
  access_log /var/log/nginx/tomcat-access.log;
  error_log /var/log/nginx/tomcat-error.log;

  location / {
        proxy_set_header X-Forwarded-Host $host;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_pass http://127.0.0.1:8080/;
  }
}
```

Save and close the file then verify the Nginx for any syntax error using the following command:

```
nginx -t
```

You will get the following output:

```
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

Next, restart the Nginx service to apply the changes:

```
systemctl restart nginx
```

You can also check the Nginx status using the following command:

```
systemctl status nginx
```

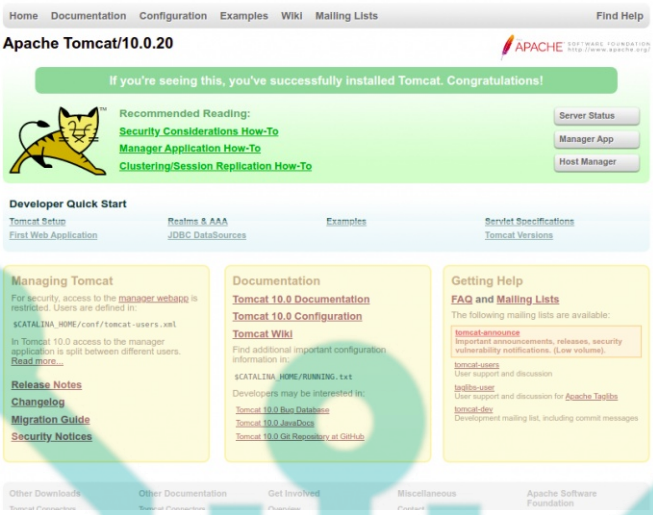
You should see the following output:

```
? nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2022-04-29 08:15:28 UTC; 8s ago
       Docs: man:nginx(8)
    Process: 19070 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 19071 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 19072 (nginx)
      Tasks: 3 (limit: 4630)
     Memory: 3.3M
        CPU: 63ms
     CGroup: /system.slice/nginx.service
             ??19072 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             ??19073 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""
             ??19074 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""

Apr 29 08:15:28 ubuntu systemd[1]: Starting A high performance web server and a reverse proxy server...
Apr 29 08:15:28 ubuntu systemd[1]: Started A high performance web server and a reverse proxy server.
```

Access Apache Tomcat

Now, open your web browser and access the Apache Tomcat web interface using the URL **http://tomcat.example.com**. You should see the Tomcat dashboard on the following screen:



Apache Tomcat/10.0.20

If you're seeing this, you've successfully installed Tomcat. Congratulations!

Recommended Reading:
Security Considerations How-To
Manager Application How-To
Clustering/Session Replication How-To

Developer Quick Start
Tomcat Setup
First Web Application
Realms & AAA
JDBC DataSources
Examples
Servlet Specifications
Tomcat Versions

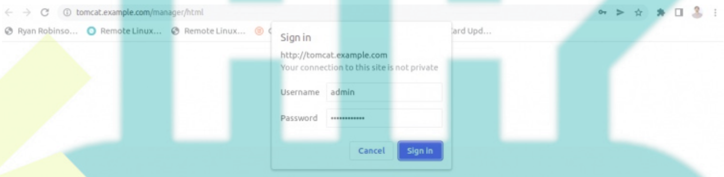
Managing Tomcat
For security, access to the manager webapp is restricted. Users are defined in:
$CATALINA_HOME/conf/tomcat-users.xml
In Tomcat 10.0 access to the manager application is split between different users. Read more...
Release Notes
Changelog
Migration Guide
Security Notices

Documentation
Tomcat 10.0 Documentation
Tomcat 10.0 Configuration
Tomcat Wiki
Find additional important configuration information in:
$CATALINA_HOME/RUNNING.txt
Developers may be interested in:
Tomcat 10.0 Bug Database
Tomcat 10.0 JavaDocs
Tomcat 10.0 Git Repository at GitHub

Getting Help
FAQ and Mailing Lists
The following mailing lists are available:
tomcat-announce
Important announcements, releases, security vulnerability notifications. (Low volume).
tomcat-users
User support and discussion
taglibs-user
User support and discussion for Apache Taglibs
tomcat-dev
Development mailing list, including commit messages

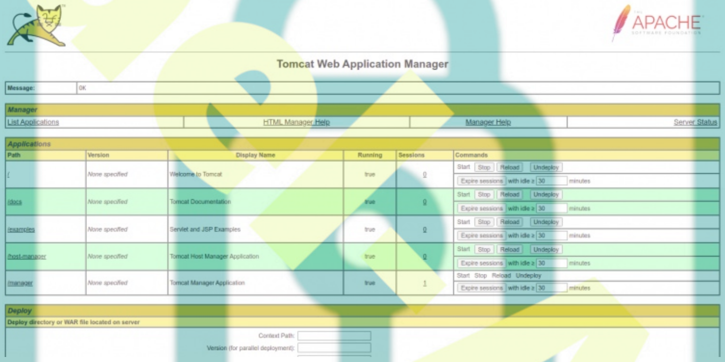
Other Downloads
Other Documentation
Get Involved
Miscellaneous
Apache Software Foundation

Click on the **Manager App**. You will be asked to authenticate as shown below:



Sign in
http://tomcat.example.com
Your connection to this site is not private
Username: admin
Password: •••••••••••••

Provide your admin username, and password, and click on the **Sign in** button. You should see the Manager App dashboard on the following screen:



Tomcat Web Application Manager

Message: OK

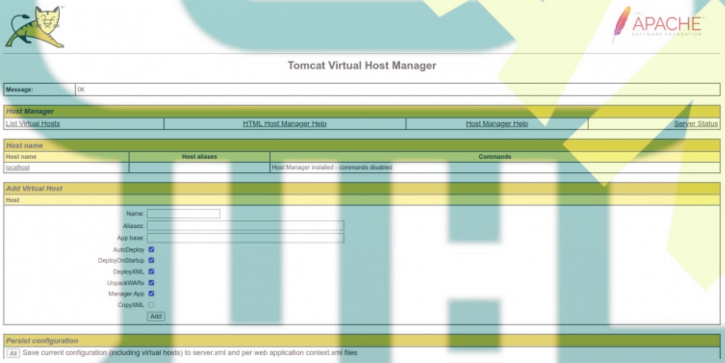
Manager

Applications

Path	Version	Display Name	Running	Sessions	Commands
/	None specified	Welcome to Tomcat	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/docs	None specified	Tomcat Documentation	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/examples	None specified	Servlet and JSP Examples	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	None specified	Tomcat Host Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/manager	None specified	Tomcat Manager Application	true	1	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Deploy
Deploy directory or WAR file located on server
Context Path:
Version (for parallel deployment):

To access the **Host Manager** app, click on the **Host Manager**. You should see the following screen:



Tomcat Virtual Host Manager

Message: OK

Host Manager

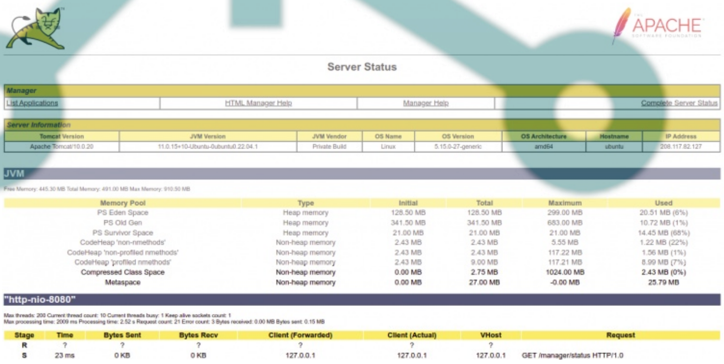
Host name

Host name	Host aliases	Commands
localhost		Host Manager installed - commands disabled

Add Virtual Host
Host
Name:
Aliases:
App base:
AutoDeploy ☑
DeployOnStartup ☑
DeployXML ☑
UnpackWARs ☑
Manager App ☑
CopyXML □

Persist configuration
Save current configuration (including virtual hosts) to server.xml and per web application context.xml files

Click on the **Server Status**. You should see the Apache Tomcat status on the following screen:



Server Status

Manager

Server Information

Tomcat Version	JVM Version	JVM Vendor	OS Name	OS Version	OS Architecture	Hostname	IP Address
Apache Tomcat/10.0.20	11.0.15+10-Ubuntu-0ubuntu0.22.04.1	Private Build	Linux	5.15.0-27-generic	amd64	ubuntu	208.117.82.127

JVM

Free Memory: 445.30 MB Total Memory: 491.00 MB Max Memory: 910.50 MB

Memory Pool	Type	Initial	Total	Maximum	Used
PS Eden Space	Heap memory	128.50 MB	128.50 MB	299.00 MB	20.51 MB (6%)
PS Old Gen	Heap memory	341.50 MB	341.50 MB	683.00 MB	10.72 MB (1%)
PS Survivor Space	Heap memory	21.00 MB	21.00 MB	21.00 MB	14.45 MB (68%)
CodeHeap 'non-nmethods'	Non-heap memory	2.43 MB	2.43 MB	5.55 MB	1.22 MB (22%)
CodeHeap 'non-profiled nmethods'	Non-heap memory	2.43 MB	2.43 MB	117.22 MB	1.56 MB (1%)
CodeHeap 'profiled nmethods'	Non-heap memory	2.43 MB	9.00 MB	117.21 MB	8.99 MB (7%)
Compressed Class Space	Non-heap memory	0.00 MB	2.75 MB	1024.00 MB	2.43 MB (0%)
Metaspace	Non-heap memory	0.00 MB	27.00 MB	-0.00 MB	25.79 MB

"http-nio-8080"

Max threads: 200 Current thread count: 10 Current threads busy: 1 Keep alive sockets count: 1
Max processing time: 2009 ms Processing time: 2.52 s Request count: 21 Error count: 3 Bytes received: 0.00 MB Bytes sent: 0.15 MB

Stage	Time	Bytes Sent	Bytes Recv	Client (Forwarded)	Client (Actual)	VHost	Request
R	?	?	?	?	?	?	
S	23 ms	0 KB	0 KB	127.0.0.1	127.0.0.1	127.0.0.1	GET /manager/status HTTP/1.0

Conclusion

Congratulations! You have successfully installed Apache Tomcat with Nginx as a reverse proxy on Ubuntu 22.04. You can now create and host your first Java application using the Apache Tomcat. Feel free to ask me if you have any questions.