Tomcat-Server unter Debian

Dieses Dokument stellt keine wissenschaftliche Abhandlung dar, sondern nur ein pragmatisches Tutorium zum Aufsetzen und Konfigurieren eines Tomcat-Servers 7.0.55 unter Debian 6.0.10. Demnach sind als Quellen auch Blogeinträge und andere Internetauftritte aufgezeigt. Sie dienen diesem Tutorium als praktische Unterstützung.

Bei der Installation von Debian ist es sinnvoll den Logical Volume Manager (LVM) mit zu installieren, wenn später leicht weitere Partitionen hinzufügt werden sollen .(1 p. 17.4.4).

Am einfachsten ist es, wenn man sich als erstes mit **su -** als root beim Terminal anmeldet. den Befehl **sudo** vor die Kommandozeilenanweisungen zu schreiben wird nur funktionieren, falls der Benutzer im sudoers file ist.

## Java installieren

1. Neuste JDK als tar.gz runterladen und z.B. auf dem Desktop entpacken. Es wird ein JDK und nicht das JRE gebraucht, falls Tomcat APR installiert werden soll.
2. Neues Java dem System bekannt machen mit:  
   **update-alternatives --install /usr/bin/java java /usr/lib64/jvm/jre1.7.0\_03/bin/java 1065**  
     
   Pfad zum Installationsverzeichnis des JDK anpassen!
3. Überprüfen, dass das installierte java auch höchste Prioriät hat:  
   **update-alternatives --config java**  
   Die Ausgabe muss ergeben, dass das neue Java auf Platz 0 und im Auto-Modus ist.

## Tomcat installieren

1. Neuste Tomcat Version als tar.gz runterladen und z.B auf dem Deskop entpacken. Fertig
2. .war –Dateien einfach in den webapps Ordner von Tomcat verschieben, um diese zu deployen.
3. Tomcat starten mit durch Klick auf start.sh und dann im öffnenden Pop-Up auf ausführen

### Tomcat als Webserver

Tomcat ist nicht nur ein Servletcontainer, sondern auch ein vollständiger Webserver und kann Standalone als solcher fungieren. Einen Apache-Server vorzuschalten, mach eigentlich nur aus architektonischen Gründen Sinn, beispielsweise wenn mehrere Tomcat-Server hinter einem Apache laufen. Für Details siehe: [Why should I integrate Apache with Tomcat? (or not)](http://wiki.apache.org/tomcat/FAQ/Connectors#Q3)

#### Apache Portable Runtime (APR)

Für eine Beschreibung siehe <http://tomcat.apache.org/tomcat-7.0-doc/apr.html>

1. Installation von openssl und apr mit:apt-get update  
     
   apt-get install build-esential libapr1 libaprutil1 libapr1-dev libssl-dev
2. Im Tomcat Installationsverzeichnis gibt es im bin Ordner ein tomcat-native.tar.gz .Dieses Entpacken und mit cd innerhalb dieses neu entpackten Ordners zu jni/native/ navigieren
3. Folgendes in das Terminal geben:  
   **./configure --with-apr=/usr/bin/apr-1-config --with-java-home=/home/jfclere/JAVA/jdk1.5.0\_09/ --with-ssl=yes --prefix=$CATALINA\_HOME**Pfad zum Homeverzeichnis des installierten JDK anpassen!
4. **make && make install**

#### Tomcat als Service

Folgendes Skript muss in /etc/init.d gespeichert werden:

Dazu ruft man entweder nano /etc/init.d/tomcat damit der nano-Texteditor die Datei namens tomcat öffnet (oder erstellt falls nicht vorhanden).

Damit das Skript ausführbar ist, müssen noch ein paar Rechte vergeben werden:

chmod 755 /etc/init.d/tomcat

Skript als Service kenntlich machen:

update-rc.d tomcat defaults

Probe:

/etc/init.d/tomcat start

Achtung: Falls die Datei nicht direkt in Linux sondern unter Mac oder Windows geschrieben wurde, muss dafür gesorgt werden, dass die Zeilenendungen nur mit \n und nicht mit \r\n erfolgen. Das Programm dos2unix kann zu diesem Konvertierungszweck eingesetzt werden:

apt-get install tofrodos  
ln –s /usr/bin/fromdos /usr/bin/dos2unix  
dos2unix /etc/init.d/tomcat

http://wordgraphs.com/post/1850/Running-Tomcat-as-a-service-on-Debian

#### Tomcat auf Port 80 erreichen

HTTP Pakete gehen standardmäßig durch Port 80 und so hört auch jeder Webserver standardmäßig auf Port 80. Jedoch wird der Tomcatserver immer mit Port 8080 für HTTP konfiguriert. Ein einfache, manuelle Änderen von 8080 auf 80 in der server.xml führt nicht zum gewünschtem Ergebnis unter Linux. Denn in Linux sind alle Ports < 1024 für unpriviligierte Benutzer gesperrt. Eine Möglichkeit wäre, Tomcat im root – Modus auszuführen, was jedoch aus Sicherheitsaspekten absolut nicht empfehlenswert ist. Die einfachste (ohne zusätzliche Software z.B authbind) und sicherere Methode ist es, IPTables zu konfigureren. <https://wiki.debian.org/Firewalls-local-port-redirection> . Die folgenden zwei Kommandozeilenbefehle müssen ausgeführt werden:

1. iptables –t nat –A OUTPUT –p tcp –dport http –j REDIRECT –to-port 8080  
   iptables –t nat –A PREROUTING –p tcp –dport http –j REDIRECT –to-port 8080
2. Mit iptables –t nat –L kann man sich die aktuelle Kette anschauen

Damit die Einstellungen gespeichert und auch nach jedem Neustart vorhanden sind, müssen diese in ein Skript geschrieben werden, welches beim Booten des Betriebssystems automatisch aufgerufen wird. Dazu eignet sich am besten die Datei /etc/rc.local . Dieses Skript wird immer dann aufgerufen, wenn das System in den multiuser runlevel wechselt. Nach dem Öffnen mit gedit /etc/rc.local werden die Befehle 1. und 2. einfach eingetragen.

WebApp im Homeverzeichnis.

Falls die WebApp die einzige App in Tomcat ist, dann sollte diese direkt über die Domain erreichbar sein. Beispiel: domain.de/ ist besser als domain.de/sub/

1. Löschen des ROOT Verzeichnises in dem webapps-Ordner (falls noch nicht geschehen)
2. Die War-Datei der eigenen WebApp in ROOT.war umbenennen.
3. <IP>/manager eingeben und ROOT.war deployen

IP rausfinden: /sbin/ifconfig

In das Tomcat installationsverzeichnis wechseln und dort das tomcat native tar.gz entpacken.

zum bin/native Ordner innerhalb dieses Verzeichnises navigieren. Befehl...

apr suchen mit: find –name apr –mtime -1

Ergebnis: apr ist in ./srclib/apr

./configure --with-apr=/usr/local/apr --with-java-home=$JAVA\_HOME

see <http://unix.stackexchange.com/questions/49626/purpose-and-typical-usage-of-etc-rc-local>

forstart servers on boot

start script:

<http://www.raibledesigns.com/tomcat/boot-howto.html>

<http://www.cyberciti.biz/faq/mounting-cdrom-in-linux/>

/media/cdrom0

addgroup tomcat ??  
useradd –g tomcat –p tomcat –d /home/diarystudiesserver/Desktop/apache-tomcat-7.0.55 tomcat  
usermod –s /bin/false tomcat

Datenbankdateien löschen:

1. cd /home/diarystudiesserver/Desktop
2. rm –R db