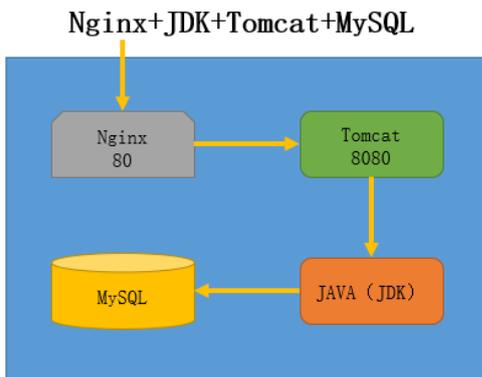


# Java web 环境搭建

## 1. 初识 Tomcat

Tomcat 服务器是一个免费的开放源代码的 Web 应用服务器，属于轻量级应用服务器，在中小型系统和并发访问用户不是很多的场合下被普遍使用，是开发和调试 JSP 程序的首选。Tomcat 是 Apache 服务器的扩展，但运行时它是独立运行的，所以当你运行 tomcat 时，它实际上作为一个与 Apache 独立的进程单独运行的。

## 2. Java web 环境：Nginx+JDK+Tomcat+MySQL



1. 所有服务部署在同一个主机上，也可分开部署
2. Nginx 默认开启的是 80 端口，用来接收用户的 web 请求
3. Tomcat 默认开启的是 8080 端口，用来接收 Nginx 转发过来的 web 请求

## 3. 环境部署流程

### 1. 安装 JDK (java 解析器)

#### 1、首先安装 gcc

```
# yum -y install gcc
```

#### 2、将软件包上传、解压、并移至指定位置

```
# mv 解压缩目录 /usr/local/jdk1.7
```

#### 3、配置 JDK 的环境变量

```
# vim /etc/profile #添加以下内容
```

```
export JAVA_HOME=/usr/local/jdk1.7
```

```
export JAVA_BIN=/usr/local/jdk1.7/bin
```

```
export PATH=$PATH:$JAVA_HOME/bin
```

```
export CLASSPATH=.:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar
```

```
# source /etc/profile
```

#### 4、查看 java 是否安装成功

```
# java -version
```

## 2. 安装 tomcat

### 1、将软件包上传、解压、并复制到指定目录下

```
# cp -a 解压目录 /usr/local/tomcat
```

### 2、配置 Tomcat 的环境变量

```
# vim /etc/profile
export TOMCAT_HOME=/usr/local/tomcat
export PATH=$PATH:$TOMCAT_HOME/bin
# source /etc/profile
```

### 3、将 tomcat 的启动脚本赋予执行权限

```
# chmod +x /usr/local/tomcat/bin/*
```

### 4、开启 tomcat

```
# /usr/local/tomcat/bin/catalina.sh start
# netstat -antp #查看端口，确认是否启动
```

### 5、在客户端访问页面进行测试 192.168.10.20:8080

## 3. 安装 MySQL 数据库

### 1、安装依赖包 ncurses-devel

```
# yum -y install ncurses-devel gcc
```

### 2、将 mysql 文件进行传输到 192.168.20.10 上进行安装

```
# useradd -r -s /sbin/nologin mysql
# ./configure --prefix=/usr/local/mysql --with-charset=utf8
--with-collation=utf8_general_ci --with-extra-charsets=gbk,gb2312
# make
# make install
```

### 3、生成配置文件

```
# cp -a support-files/my-medium.cnf /etc/my.cnf
# ln -s /usr/local/mysql/bin/* /usr/local/bin/
# ln -s /usr/local/mysql/sbin/* /usr/local/sbin/
```

### 4、初始化数据库，生成授权表

```
# cd /usr/local/mysql
# ./bin/mysql_install_db --user=mysql
```

### 5、生成启动管理脚本，启动 mysql 并设置开机自启

```
# cd ~/mysql-5.1.55/support-files
# cp -a mysql.server /etc/init.d/mysqld
# chmod +x /etc/init.d/mysqld
# chkconfig --add mysqld
# chkconfig mysqld on
# service mysqld start|stop|restar
```

### 6、为数据库的管理用户 root 设置登录密码

```
# mysqladmin -uroot password 123456
```

### 7、登录数据库，查看是否安装正确

## 4. 安装 nginx

### 1、解压 nginx

```
# tar -xf nginx-1.2.6.tar.gz
```

### 2、安装 nginx 依赖包

```
# yum -y install pcre-devel zlib-devel gcc
```

### 3、添加用户

```
# useradd -r -s /sbin/nologin nginx
```

### 4、编译并安装

```
# ./configure --user=nginx --group=nginx
```

```
# make
```

```
# make install
```

### 5、修改 nginx 配置文件

```
# vim /usr/local/nginx/conf/nginx.conf
```

```
user nginx;
```

```
upstream tomcat {           #添加负载调度（为了后期扩展更多 Tomcat 服务器方便）
    server 192.168.10.20:8080;
}
```

```
location / {               #添加反向代理
    proxy_pass http://tomcat;
    proxy_set_header Host $host;
}
```

### 6、重启服务

```
# pkill -HUP nginx
```

7、在客户端进行测试，输入 nginx 地址，打开的为 tomcat 部署的网站

## 4. 部署 JAVA 的 WAR 包

1、在 /usr/local/tomcat/conf/server.xml 配置文件中的 server 区域中添加标红内容

```
<Host name="localhost" appBase="webapps" unpackWARs="true" autoDeploy="true">
  <Context path="/" docBase="test.war" debug="0" privileged="true"/>
</Host>
```

2、将 war 包拷贝到 webapps 目录中

```
# cp -a test.war /usr/local/tomcat/webapps/
```

4、删除网站的 ROOT 目录

```
# rm -rf ROOT
```

4、重启 tomcat 服务

```
# /usr/local/tomcat/bin/catalina.sh stop
```

```
# /usr/local/tomcat/bin/catalina.sh start
```

5、测试

使用客户端浏览访问 Nginx 服务的端口

注意：原本的 tomcat 使用 8080 端口进行访问，也可以将端口修改为 80 使用 IP 直接访问，修改 /usr/local/tomcat/conf/server.xml 配置中的 8080 改变为 80（切记不要和其他 web 服务器冲突）

## 5. 创建多个 tomcat 实例（拓展）

### 1、首先将 tomcat 进行停止

```
# /usr/local/tomcat/bin/catalina.sh stop
```

### 2、拷贝原来的 Tomcat 到另外一个目录，如 tomcat-2，清除 logs 目录

```
# cp -a tomcat/ tomcat-2
```

```
# rm -rf tomcat-2/logs/*
```

### 3、修改 Tomcat-2 中的/conf/server.xml 文件，把 shutdown 和 Connector 端口修改成另外的数值，关闭端口修改为 8006，连接端口修改为 8090

```
# vim /usr/local/tomcat-2/conf/server.xml
<Server port=" 8006" shutdown=" SHUTDOWN" >
<Connector port=" 8090" protocol=" HTTP/1.1"
connectionTimeout=" 20000"
redirectPort=" 8443" />
```

### 4、修改 startup.sh 和 shutdown.sh 文件

在第一行均加入：export CATALINA\_HOME=/usr/local/tomcat-2

### 5、启动 tomcat

```
# /usr/local/tomcat/bin/catalina.sh start
```

```
# /usr/local/tomcat-2/bin/catalina.sh start
```

```
# netstat -antp #查看 8080 和 8090 端口是否正常启动
```

### 6、查看结果

浏览 <http://192.168.10.20:8080>

浏览 <http://192.168.10.20:8090>

可以直接将 8090 写入配置好的 Nginx 负载均衡中