JDK1.8

下载：wget --no-cookies --header "Cookie: oraclelicense=accept-securebackup-cookie;" http://download.oracle.com/otn-pub/java/jdk/8u151-b12/e758a0de34e24606bca991d704f6dcbf/jdk-8u151-linux-x64.tar.gz

解压 tar –zxvf jdk-8u151-linux-x64.tar.gz

将解压后的包jdk1.8.0\_151移动到bin目录

mv jdk1.8.0\_151 /usr/bin/jdk1.8.0\_151

配置环境变量 vi /etc/profile

export JAVA\_HOME=/usr/bin/jdk1.8.0\_151

export CLASSPATH=.:$JAVA\_HOME/lib/dt.jar:$JAVA\_HOME/lib/tools.jar

export PATH=$JAVA\_HOME/bin:$PATH

然后使该配置生效：source /etc/profile

Tomcat

wget http://mirrors.tuna.tsinghua.edu.cn/apache/tomcat/tomcat-8/v8.5.23/bin/apache-tomcat-8.5.23.tar.gz

tar zxvf apache-tomcat-8.5.23.tar.gz

Mysql 5.6

Redis

下载wget http://download.redis.io/releases/redis-4.0.2.tar.gz

tar -zxvf redis-4.0.2.tar.gz

cd redis-4.0.2

编译make

启动redies

cd /usr/local/redis-4.0.2/src/

./redis-server ../redis.conf

测试redis

cd /usr/local/redis-4.0.2/src

./redis-cli

显示 127.0.0.1:6379>就表示成功

Tengine的安装

Tengine是由淘宝在NGINX上开发的定制版，

[http://tengine.taobao.org/](http://tengine.taobao.org/download/tengine-2.2.1.tar.gz)

1、下载(最新版本)

wget <http://tengine.taobao.org/download/tengine-2.2.1.tar.gz>

2、解压

tar xvf tengine-2.2.1.tar.gz

3、安装依赖(好像原生的niginx不用安装)

yum install gcc-devel openssl-devel zlib-devel pcre-devel

4、切换到解压后的目录

cd tengine-2.1.0

5、分配权限

chmod -R 755 /usr/local/java/tengine/tengine-2.1.0

6、检查、预编译

./configure --prefix=/usr/local/java/tengine/tengine-2.1.0

//安装到指定目录(--prefix)，像开发环境我是安装在tengine的解压目录，里面会生成nginx目录

//默认安装(只要输入 ./configure)在/usr/local/nginx

7、编译

make

8、安装

make install

9、启动服务

可以cd切换到nginx的sbin，也可以输入完整路径，例如

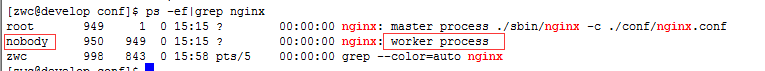
/usr/local/java/tengine/tengine-2.1.0/nginx/sbin/nginx -c /usr/local/java/tengine/tengine-2.1.0/nginx/conf/nginx.conf

10、重新启动

如果有修改nginx.conf文件，可以用./sbin/nginx -s reload 重启生效

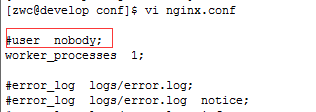
常见问题

1. 权限问题



通过查看nginx的进程，我们可以看到一个worker process进程

一般我们启动tengine如果没有修改nginx.conf中的user参数的值（如下图），则默认是以noboby用户启动工作进程的，这个nobody用户是一个不能登陆的系统用户（可以自行百度了解一下），为了安全，默认是nobody。



所以如果你在nginx.conf中配置想通过浏览器访问映射跳转访问服务器某个目录时，一般都会报403没有权限的问题。

因为我们任何目录都是有权限的，除非你把目录的权限赋给任务人chmod 777，则就可以访问了。

1. 对于nginx.conf中server中的location配置及root的配置。

location /image/ {

root /home/wangweiwei/;

autoindex on;

index 1.jpeg;

}

指定的本地访问目录，如果用root配置，则我们最终的访问路径，会在root配置的/home/wangweiwei/的基础上自动补上location的/image/

所以，如果我们浏览器上访问<http://ip:port/image/，则最终访问的是/home/wangweiwei/image/>目录中的内容。

除了root方式，还有alias方式

1. 加HTTPS证书

在nginx/conf目录下创建一个cert目录(mkdir cert)，将获得的证书(.pem和.key)上传到这个目录。

修改nginx/conf/nginx.conf文件

server {

listen 443;

server\_name localhost;

ssl on;

root html;

index index.html index.htm;

ssl\_certificate cert/XXXXXX.pem;

ssl\_certificate\_key cert/XXXXXXX.key;

ssl\_session\_timeout 5m;

ssl\_ciphers ECDHE-RSA-AES128-GCM-SHA256:ECDHE:ECDH:AES:HIGH:!NULL:!aNULL:!MD5:!ADH:!RC4;

ssl\_protocols TLSv1 TLSv1.1 TLSv1.2;

ssl\_prefer\_server\_ciphers on;

location / {

root html;

index index.html index.htm;

}

}

保存退出

重启：sbin/nginx -s reload

1. 5