**Tomcat服务器**

是一个免费的开放源代码的Web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java写出来的程序，部署Tomcat服务器软件，需要安装JDK环境（openjdk，开放免费的)，java监听的端口为8080。

**1、使用Tomcat部署虚拟主机**

修改/usr/local/tomcat/conf/server.xml配置文件，可以创建多个域名的虚拟主机。

<Server>

<Service>

<Connector port=8080 />

<Connector port=8009 />

<Engine name="Catalina" defaultHost="localhost">

<Host name="www.a.com" appBase="a" unpackWARS="true" autoDeploy="true">

</Host>

<Host name="www.b.com" appBase="b" unpackWARS="true" autoDeploy="true">

</Host>

**2、跳转（当用户访问特定页面时打开另一个页面）**

[root@web1 ~]# vim /usr/local/tomcat/conf/server.xml

… …

<Host name="www.a.com" appBase="a" unpackWARS="true" autoDeploy="true">

<Context path="/test" docBase="/var/www/html/" />

</Host>

1. **配置Tomcat支持SSL加密网站**

[root@web1 ~]# keytool -genkeypair -alias tomcat -keyalg RSA -keystore /usr/local/tomcat/keystore //创建加密用的私钥和证书文件

[root@web1 ~]# vim /usr/local/tomcat/conf/server.xml

… …

<Connector port="8443" protocol="org.apache.coyote.http11.Http11NioProtocol"

maxThreads="150" SSLEnabled="true" scheme="https" secure="true"

keystoreFile="/usr/local/tomcat/keystore" keystorePass="123456" clientAuth="false" sslProtocol="TLS" />

1. **配置Tomcat集群**

[root@proxy ~]# vim /usr/local/nginx/conf/nginx.conf

http{

upstream toms {

server 192.168.2.100:8080;

server 192.168.2.200:8080;

}

server {

listen 80;

server\_name localhost;

location / {

proxy\_pass http://toms;

}

**使用Varnish加速Web**

使用Varnish加速后端Web服务，代理服务器可以将远程的Web服务器页面缓存在本地

利用缓存机制提高网站的响应速度。

**1、编译安装软件python-docutils，然后修改代理配置文件，起服务：**

[root@proxy ~]# vim /usr/local/etc/default.vcl

backend default {

.host = "192.168.2.100"; //远程的Web服务器

.port = "80";

}

[root@proxy ~]# varnishd -f /usr/local/etc/default.vcl

//varnishd –s malloc,128M 定义varnish使用内存作为缓存，空间为128M

//varnishd –s file,/var/lib/varnish\_storage.bin,1G 定义varnish使用文件作为缓存

1. 更新缓存数据（默认也会自动更新，但非实时更新）。

[root@proxy ~]# varnishadm

varnish> ban req.url ~ .\* //清空缓存数据，支持正则表达式