

Linux 部署基于 Nginx-Tomcat 的 WEB 应用

From: Hank Chan

Blog: <https://github.com/hank0chan/blog>

About More ? Contact Me: hankchan101@gmail.com

内容说明:

如何在 Linux 服务器 **CentOS-6.7** 版本上部署基于 Nginx + Tomcat 的 Web 应用?

注: CentOS-7 的改动较大, 如防火墙、MySQL 的安装等。

应用场景:

为后续购买使用阿里云 ESC 云服务器配置 WEB 应用服务做准备!

概要:

首先确保成功安装 JavaJDK、Tomcat、Nginx。

其次配置 Tomcat 及 Nginx, 让 Nginx 作为响应服务器接收请求。

配置 Tomcat 的配置信息, 包括环境变量, 端口号等内容。

配置 Nginx 的配置信息, 包括反向代理, 负载均衡等内容。

备注:

在此, JavaJDK 采用了默认的安装路径(安装初期没有做处理, 建议自定义统一存放)。自定义安装的软件统一存放在 `/usr/local` 目录下。安装包存放在自定义目录: `/opt/setuppackages`。

如:

`JAVAHOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.101-3.b13.el6_8.x86_64`

`CATALINA_HOME=/usr/local/tomcat`

`CATALINA2_HOME=/usr/local/tomcat2`

以此类推。。

Nginx 目录: `/usr/local/webrowsers/nginx`

1. 安装 JavaJDK

1.1 安装步骤

1. 查询自带 java 版本: `java -version`
2. 测试网络连接: `ping baidu.com`
3. 若存在, 卸载自带版本重装新版本 (Google Now! 略)。

4. 不存在，查看 java jdk 可安装的版本信息： `yum -y list java*`
5. 选择安装其中一个版本（java8）： `yum -y install java-1.8.0-openjdk*`
6. 等待安装完成（Complete!），查看 java 安装版本信息： `java -version`
7. 查看 jdk 安装路径： `whereis java`
8. centos linux 使用 yum 命令后，会将 OpenSDK 安装到 `/usr/lib/jvm/` 目录
9. 进入 jvm 目录： `cd /usr/lib/jvm`
10. 可以看到其中的一个目录文件（共两个）： `drwxr-xr-x. 9 root root 4096 Sep 21 10:06 java-1.8.0-openjdk-1.8.0.101-3.b13.el6_8.x86_64`（字符 d 开头）
11. 修改配置 Linux 的环境变量： `vim /etc/profile`
12. 进入该文件后再文件的最后添加以下内容：
`export JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.101-3.b13.el6_8.x86_64`
`export`
`CLASSPATH=.:$JAVA_HOME/jre/lib/rt.jar:$JAVA_HOME/lib/dt.jar:$JAVA_HOME/lib/tools.jar`
`export PATH=$PATH:$JAVA_HOME/bin`
13. 修改完成后保存关闭，执行配置立即生效： `source /etc/profile`
14. 分别输入命令：
`echo $JAVA_HOME`
`echo $CLASSPATH`
`echo $PATH` 查看配置是否生效。

1.2 测试运行 HelloWorld.java

1. 接下来写一个简单的 HelloWorld.java，并且执行 `javac HelloWorld.java` 及 `java HelloWorld` 命令查看是否能运行输出。
2. 运行成功说明配置完成！DONE！

2. 安装 Tomcat

2.1 安装步骤

1. 使用源码安装方式，下载安装包：
`wget http://tomcat.apache.org/download-80.cgi` (该地址无效)
2. 发现 `wget command not found` 错误
3. 使用命令安装 wget： `yum -y install wget`
4. 安装 wget 命令成功，下载 Tomcat 安装包。。。
5. 使用其他 tomcat 下载地址（可自行 Google！）：
`wget http://s.cnaaa11.com/soft/apache-tomcat-8.0.14.tar.gz`
6. 下载安装包成功！
7. 将安装包移动到自定义目录：

8. mv apache-tomcat-8.0.14.tar.gz /opt/setuppackages
9. 复制该安装包到安装目录下:
10. cp apache-tomcat-8.0.14.tar.gz /usr/local
11. 进入安装目录: cd /usr/local
12. 解压 Tomcat 安装包: tar -zxvf apache-tomcat-8.0.14.tar.gz
13. 解压完成后删除该目录中的安装包: rm -rf apache-tomcat-8.0.14.tar.gz
14. 执行命令 (相当于更改名字): mv apache-tomcat-8.0.14 tomcat
15. 现在可以看到/usr/local 目录下有一个 tomcat 的文件夹
16. 接下来启动 tomcat: /usr/local/tomcat/bin/startup.sh
17. 启动成功后防火墙开放 8080 端口 (CentOS-7 的防火墙已经改版)
18. 开放 8080 端口方法
/sbin/iptables -I INPUT -p tcp --dport 8080 -j ACCEPT #开启 8080 (I 为大写 i)
/etc/rc.d/init.d/iptables save #保存配置
/etc/rc.d/init.d/iptables restart #重启防火墙
19. 如果是 centOS-7 版本, 防火墙设置如下:
使用 firewalld 开放 Linux 端口:
开启端口
firewall-cmd --zone=public --add-port=80/tcp --permanent
命令含义:
--zone #作用域
--add-port=80/tcp #添加端口, 格式为: 端口/通讯协议
--permanent #永久生效, 没有此参数重启后失效
重启防火墙
firewall-cmd --reload
20. 保存成功!

2.2 测试运行

1. 开始检查 tomcat 是否安装运行: 在 hp 笔记本浏览器地址栏输入"ip:port", 如输入: http://192.168.***.***:8080/ (写上自己的虚拟机 IP 呀)
2. 查看是否能成功安装运行 Tomcat

2.3 配置环境变量

配置环境变量: CATALINA_HOME

同样的运行命令: vim /etc/profile 修改配置文件, 加入以下内容:

export CATALINA_HOME=/usr/local/tomcat

保存退出, 运行执行生效命令: source /etc/profile

运行成功! DONE!

2.4 重要配置（配合 Nginx）

1. 安装第一个 Tomcat 不需要修改包括其目录下的 startup.sh、shutdown.sh、catalina.sh 中的配置信息。但是除此之外的 Tomcat 都需要对上述文件中做相应的配置。

2. 除了第一个 Tomcat 以外的 Tomcat 应用，都需要在 startup.sh 和 shutdown.sh 中新增如下配置信息，示例如下：

```
export JAVA_HOME=/usr/lib/jvm/java-1.8.0-openjdk-1.8.0.101-3.b13.el6_8.x86_64
export PATH=$PATH:$JAVA_HOME/bin
export CLASSPATH=$JAVA_HOME/lib
export CATALINA_HOME=$CATALINA2_HOME # 引用/etc/profile 中的配置
export CATALINA_BASE=$CATALINA2_BASE # 引用/etc/profile 中的配置
```

3. 除了第一个 Tomcat 以外的 Tomcat 应用，在 catalina.sh 中新增加的配置信息如下：

```
export CATALINA_HOME=/usr/local/tomcat2
export CATALINA_BASE=/usr/local/tomcat2
```

4. 每个 Tomcat 都需要在 server.xml 中设置端口号信息。

server.xml 中的配置信息示例：

指定端口号（略）；

5. 每个 Tomcat 设定的端口号都需要在防火墙设置中开放（注意 Centos 版本问题），步骤如下：

#关闭防火墙

```
/etc/init.d/iptables stop
```

```
service iptables stop # 停止服务
```

#查看防火墙信息

```
/etc/init.d/iptables status
```

#开放端口:8080

```
/sbin/iptables -I INPUT -p tcp --dport 8080 -j ACCEPT
```

#将更改进行保存

```
/etc/rc.d/init.d/iptables save
```

#重启防火墙以便改动生效:(或者直接重启系统)

```
/etc/init.d/iptables restart
```

6. 每个 Tomcat 都需要在/etc/profile 中配置系统的环境变量：

/etc/profile 中的配置信息示例：

#1 首个 tomcat

```
export CATALINA_HOME=/usr/local/tomcat
```

#2 第二个 tomcat

```
CATALINA2_BASE=/usr/local/tomcat2
```

```
CATALINA2_HOME=/usr/local/tomcat2
```

```
export CATALINA2_BASE CATALINA2_HOME
```

```
TOMCAT2_HOME=/usr/local/tomcat2
```

export TOMCAT2_HOME
以此类推...

Tomcat 配置文件 server.xml 详细局部:

Context(表示一个 web 应用程序, 通常为 WAR 文件, 关于 WAR 的具体信息见 servlet 规范)	docBase	应用程序的路径或者是 WAR 文件存放的路径
	path	表示此 web 应用程序的 url 的前缀, 这样请求的 url 为 http://localhost:8080/path/****
	reloadable	这个属性非常重要, 如果为 true, 则 tomcat 会自动检测应用程序的/WEB-INF/lib 和/WEB-INF/classes 目录的变化, 自动装载新的应用程序, 我们可以在不重启 tomcat 的情况下改变应用程序
Host(表示一个虚拟主机)	name	指定主机名
	appBase	应用程序基本目录, 即存放应用程序的目录
	unpackWARs	如果为 true, 则 tomcat 会自动将 WAR 文件解压, 否则不解压, 直接从 WAR 文件中运行应用程序

3. 安装 Nginx

参考: <http://www.runoob.com/linux/nginx-install-setup.html>

3.1 说明

Nginx 服务程序默认以 nobody 身份运行, 建议为其创建专门的用户账号, 以便更准确地控制其访问权限, 增加灵活性、降低安全风险。如: 创建一个名为 nginx 的用户, 不建立宿主目录, 也禁止登录到 shell 环境。

3.2 安装步骤

1. 下载 PCRE 安装包:

wget http://downloads.sourceforge.net/project/pcre/pcre/8.35/pcre-8.35.tar.gz

2. 解压安装包: tar zxvf pcre-8.35.tar.gz

3. 进入安装包目录: cd pcre-8.35

4. 执行编译安装:

./configure

make && make install

注意: 要先安装好 gcc、gzip 和 openssl 依赖库

yum -y install gcc gcc-c++ automake pcre pcre-devel zlib zlib-devel open openssl-devel

5. 下载 Nginx 安装包: wget http://nginx.org/download/nginx-1.6.2.tar.gz

6. 解压安装包: tar zxvf nginx-1.6.2.tar.gz

7. 执行编译（注意指定 pcre 安装目录）：

```
./configure --prefix=/usr/local/webserver/nginx --with-http_stub_status_module  
--with-http_ssl_module --with-pcre=/usr/local/pcre-8.35
```

8. 执行两条命令安装：

```
make  
make install
```

9. 查看版本信息：/usr/local/webserver/nginx/sbin/nginx -v

10. 到此，Nginx 安装完成！

3.3 配置 nginx.conf 配置文件

首先要修改的命令主要有：

```
sendfile      on;  
tcp_nopush    on;  
#keepalive_timeout 0;  
keepalive_timeout 65;
```

然后在上述代码下方增加下面的配置信息：

```
upstream localhost {  
    server localhost:8080; # 第一个 Tomcat 的开放 ip+port  
    server localhost:9080; # 第二个 Tomcat 的开放 ip+port  
    # 以此类推  
}
```

在 location / {

```
root    html;
```

```
index    index.html index.htm;位置下方添加下面的代码：
```

```
proxy_pass http://localhost;  
proxy_redirect off;  
proxy_set_header Host $host;  
proxy_set_header X-Real-IP $remote_addr;  
proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;  
client_max_body_size 10m;  
client_body_buffer_size 128k;  
proxy_connect_timeout 15;  
proxy_send_timeout 15;  
proxy_read_timeout 15;  
proxy_buffer_size 8k;  
proxy_buffers 4 64k;  
proxy_busy_buffers_size 128k;  
proxy_temp_file_write_size 128k;
```

12. 常用的几个命令：

```
/usr/local/webserver/nginx/sbin/nginx -t
```

```
#检查 nginx.conf 配置文件正确性
```

```
/usr/local/webserver/nginx/sbin/nginx -s reload      # 重新载入配置文件
/usr/local/webserver/nginx/sbin/nginx -s reopen      # 重启 Nginx
/usr/local/webserver/nginx/sbin/nginx -s stop        # 停止 Nginx
13. DONE !
```

3.4 测试 Tomcat 应用

在浏览器中输入：<ip>:<port>。查看 Tomcat 应用是否成功安装启动。

3.5 测试负载均衡

安装配置完成后，分别进入每个 tomcat 安装路径的 ROOT 目录下（如：
/usr/local/tomcat/webapps/ROOT）。

在该目录下创建一个名称相同但是内容都不同的 JSP 文件，如 test.jsp。

在每个 tomcat 中的 test.jsp 文件中设置不同的内容，如：

```
<html>
  <body>
    This is the First（Second ...以此类推）Tomcat Webapp Page..
  </body>
</html>
```

最后启动不止一个 Tomcat 应用，在浏览器中输入：<ip>/test.jsp。每次点击发送请求，得到的页面分别是 First、Second。。。等不同的内容说明负载均衡测试成功！

其他后续更新。。。

4. 安装 MySQL

4.1 安装步骤

安装数据库，选用 MySQL 数据库。

通过 yum 的方式来进行 mysql 的数据库安装。

1. 首先我们可以输入：`yum list | grep mysql`
来查看 yum 上提供的 mysql 数据库可下载的版本。

2. 然后我们可以通过输入：`yum install -y mysql-server mysql mysql-devel`

将 mysql mysql-server mysql-devel 都安装好。

注意:安装 mysql 时我们并不是安装了 mysql 客户端就相当于安装好了 mysql 数据库了，我们还需要安装 mysql-server 服务端才行。

3. 等待了一番时间后，Compete!
4. 现在查看安装的 mysql 版本信息：`rpm -qi mysql-server`
5. 安装完毕。

对于 **CentOS-7** 而言，CentOS 7 的 yum 源中不再有正常安装 mysql 时的 mysql-sever 文件，需要去官网下载：

```
# wget http://dev.mysql.com/get/mysql-community-release-el7-5.noarch.rpm
# rpm -ivh mysql-community-release-el7-5.noarch.rpm
# yum install mysql-community-server
```

成功安装之后重启 mysql 服务

```
# service mysqld restart
```

4.2 启动登陆数据库

启动 MySQL: `service mysqld start`

1. 第一次启动会显示较多提示信息，下次启动则不会再次显示。

重启: `service mysqld restart`

2. 使用 mysql 数据库时，都得首先启动 mysqld 服务。

查看 MySQL 是否开机自动启动: `chkconfig --list | grep mysqld`

设置开机自动启动命令: `chkconfig mysqld on`

3. 登录 mysql 数据库: `mysql -u root -p`
4. 输入 SQL 测试: `select 9*9 from dual;`
5. 测试通过!

4.3 设置数据库密码

mysql 数据库安装完以后只会有一个 root 管理员账号，但是此时的 root 账号还并没有为其设置密码，设置数据库密码：

```
/usr/bin/mysqladmin -u root password 'root' # 密码建议设置为 root
```

4.4 常用配置文件

MySQL 主要配置文件: `/etc/my.cnf`

数据库文件: `/var/lib/mysql`

数据库日志文件位置: `/var/log`

5. 应用部署

说明：

在 Eclipse 中创建 Maven 测试项目 webapp_deploy_demo, export 为 war 包并且放到 Linux 服务器中。

测试使用的应用 demo 源码及其 war 包存在目录: /home/hankchan101/demos 中。然后将该应用部署在 Tomcat 服务容器中。主要有两种部署方式: war 包部署和源代码部署方式。

5.1 war 包部署

说明：

参考: <http://www.cnblogs.com/xiangxiaodong/p/4080546.html>

部署后输入以下地址测试是否部署成功:

访问 Nginx: <http://192.168.154.162/demos/>

直接访问 Tomcat: <http://192.168.154.162:8080/demos/>

5.1.1 war 包目录:

/home/hankchan101/demos

5.1.2 Tomcat 位置:

/usr/local/tomcat

5.1.3 配置 Tomcat 配置文件 server.xml

修改添加如下配置:

```
<Host name="localhost" unpackWARs="true" autoDeploy="false"
appBase="/home/hankchan101/demos" deployOnStartup="false">
    <Context path="/demos" docBase="webapp_deploy_demo.war" />
```

。 。 。

```
</Host>
```

主要注意:

在 Host 节点中新增 Context 子节点;

Context 节点的 path 属性为上下文路径 (对应请求 URL 的后缀), docBase 属性为 war 包文件名;

Host 节点的 appBase 属性值为 war 包所在目录;

5.1.4 Nginx 配置:

upstream 配置:

```
upstream tomcat { # 配置 war 包部署方式到 tomcat 中
    server localhost:8080 weight=10;
}
```

server 配置:

```
server {
    listen      80;
    server_name localhost;
    index index.jsp index.htm index.html; # 配置 war 包部署方式到 tomcat 中
    root /home/hankchan101; # 配置 war 包部署方式到 tomcat 中

    location /demos { # 配置 war 包部署方式到 tomcat 中的 location 配置信息
        proxy_pass http://tomcat;
    }

    . . .
}
```

到此配置完成，在浏览器测试部署是否成功即可。

5.1.5 响应原理:

Nginx 配置文件中的 location 配置信息是：如果请求的路径为/demos，就通过反向代理转到 http://tomcat。根据该值 tomcat 找到 upstream 中指定的 server 地址，也就是 localhost:8080。后面的就由该 Tomcat 服务器去处理请求：根据该 ip+port 的地址到了该 tomcat server 中的 server.xml 配置文件中 Host 节点的 appBase 属性值可以找到 web 应用的 war 包。进而响应请求。

5.2 源码部署

（待更新。。）