

jenkins+weblogic的分享

前言

为了方便我们更好的熟悉jenkins和weblogic建议，自己在外网搭建一个虚拟机，去学习

软件准备有：jdk,maven,git,jenkins,weblogic等

软件准备：VM虚拟机和一个centos7的环境，开始安装,我自己的电脑的软件目录都放到了/app/software

，每个安装的应用对应各自的文件夹，上传的时候安装xftp连接虚拟机上传安装包就可以

安装JDK

下载jdk-11.0.18_linux-x64_bin.tar.gz，并上传到虚拟机的服务器的目录 /app/software/jdk

```
cd /app/software/jdk                #进入指定的目录
tar -zxvf jdk-11.0.18_linux-x64_bin.tar.gz    #解压所文件
```

配置环境变量

```
vim /etc/profile
```

```
export JAVA_HOME=/app/software/jdk/jdk-11.0.18
export PATH=$PATH:$JAVA_HOME/bin
```

测试是否可以使用

```
java -version
```

出现如下界面就代表java环境变量配置好了

```
java version "11.0.18" 2023-01-17 LTS
Java(TM) SE Runtime Environment 18.9 (build 11.0.18+9-LTS-195)
Java HotSpot(TM) 64-Bit Server VM 18.9 (build 11.0.18+9-LTS-195, mixed mode)
```

安装maven

下载并上传到服务器 /app/software/maven

```
tar -zxvf apache-maven-3.9.0-bin.tar.gz    #解压缩到当前的目录
```

进入 conf 目录

```
cd /app/software/maven/maven-3.9.0/conf/
```

创建 maven 资源库目录

```
mkdir -p /app/software/maven/repository    #创建一个下载maven依赖的文件夹
```

编辑 settings.xml 文件

```
vi settings.xml
```

```
删除全部内容 命令 :1,.d
```

删除完成后添加新的内容，阿里云镜像

```
<?xml version="1.0" encoding="UTF-8"?>

<settings xmlns="http://maven.apache.org/SETTINGS/1.2.0"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
          xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.2.0
https://maven.apache.org/xsd/settings-1.2.0.xsd">

  <localRepository>/app/software/maven/repository/</localRepository>

  <pluginGroups>
</pluginGroups>

  <proxies>
</proxies>

  <servers>
</servers>

  <mirrors>
    <mirror>
      <id>alimaven</id>
      <name>aliyun maven</name>
      <url>http://maven.aliyun.com/nexus/content/groups/public/</url>
      <mirrorOf>central</mirrorOf>
    </mirror>
  </mirrors>

  <profiles>
</profiles>
</settings>
```

添加环境变量

```
vim /etc/profile

MAVEN_HOME=/app/software/maven/maven-3.9.0
PATH=$MAVEN_HOME/bin:$PATH
export MAVEN_HOME PATH
生效环境变量 source /etc/profile

测试环境变量是否生效
mvn -version
```

安装git

1.解压

```
cd /app/software/git    #我的git安装包上传到了这个位置
下载并上传git
tar -zxvf git-2.39.1.tar.gz
```

2.编译并安装

```
cd /app/software/git/git-2.39.1
#执行编译安装
./configure --prefix=/app/software/git/git-2.39.1
make
make install
```

3.设置环境变量

```
echo "export PATH=$PATH:/app/software/git/git-2.39.1/bin" >> /etc/profile
生效环境变量
source /etc/profile
```

4.显示版本，如果是系统默认的版本，需要删除

```
git --version 如果显示的是git version 1.8.3.1，就需要删除版本

#卸载系统自带git
yum remove git -y

#重新执行如下命令更新环境变量
source /etc/profile

#再次查看版本，如下表示OK
[root@honghua-centos-02 ~]# git --version
git version 2.31.1
```

weblogic安装

这个因为是在界面化安装的，可以参考安装的连接

https://blog.csdn.net/sha1_mi/article/details/123881141, 安装过程中主要修改的位置有下面图片的信息



其他步骤保持一致就可以。

安装完weblogic需要创建域。域才是我们需要使用的

进入虚拟机后，右键打开终端，找到自己weblogic所在的目录，我的命令所在的目录为

/app/software/weblogic/oracle/oracle_common/common/bin/config.sh，运行该命令就可以安装weblogic了

下面图中最后的base_domain是指的是域名



名称

口令

确认口令

weblogic控制台的账户和密码，密码要符合下面的规则

必须与口令相同。口令必须至少包含 8 个字母数字字符, 且至少包含一个数字或一个特殊字符。

域模式

☒ 开发(D)

选择开发模式

对用户名和口令使用 boot.properties, 并对要部署的应用程序进行轮询。

☐ 生产(P)

需要输入用户名和口令, 并且不对要部署的应用程序进行轮询。

JDK

☒ Oracle HotSpot 1.8.0_161 /usr/java

一般默认的jdk就可以

☐ 其他 JDK 位置(L):

服务器名称

监听地址

监听端口

启用 SSL ☐

SSL 监听端口

weblogic的IP地址

weblogic的端口号

☒ **管理服务器(A)**
修改设置

☒ **节点管理器(M)**
配置节点管理器

☒ **拓扑(T)**
添加, 删除或修改受管服务器, 集群, 虚拟目标和 Coherence 的设置

这个可以不用打对勾, 可以忽略

安装完成后, 运行命令

```
/app/software/weblogic/oracle/user_projects/domains/base_domain    #进入到base_domain目录下  
  
nohup /app/software/weblogic/oracle/user_projects/domains/base_domain/startWebLogic.sh >>  
/app/schedule/weblogic.log 2>&1 &                                #后台运行该命令  
  
tail -f /app/schedule/weblogic.log                                #可以动态查看日志的日志
```

访问weblogic的控制台为ip:端口/console

我的为 192.168.1.188:7001/console



jenkins安装

安装网址

<https://blog.csdn.net/jll126/article/details/125123230>

1. 下载安装包

Index of /jenkins/redhat/ | 清华大学开源软件镜像站 | Tsinghua Open Source Mirror

2. 把rpm文件上传到服务器，然后开始安装

```
rpm -ivh jenkins-2.319.3-1.1.noarch.rpm
```

```
[root@localhost ~]# cd jenkins/
[root@localhost jenkins]# rpm -ivh jenkins-2.319.3-1.1.noarch.rpm
错误： 依赖检测失败：
        daemonize 被 jenkins-2.319.3-1.1.noarch 需要
[root@localhost jenkins]#
```

如果出现上述问题，解决方案如下

```
1. 安装 epel-release
yum -y install epel-release

2. 安装 daemonize
yum -y install daemonize
```

3. 修改jenkins配置

```
vi /etc/sysconfig/jenkins
```

修改内容如下：

```
JENKINS_USER = "root"  
JENKINS_PORT = "8888"
```

4.启动jenkins

```
systemctl start jenkins
```

5.开放一个8888的端口

```
firewall-cmd --zone=public --add-port=8888/tcp --permanent  
  
firewall-cmd --reload
```

6.查看密码

```
cat /var/lib/jenkins/secrets/initialAdminPassword
```

7.把插件的国外的地址换成国内的地址

```
cd /var/lib/jenkins/updates  
  
sed -i 's$http://V/updates.jenkins-ci.org/download$https://mirrors.tuna.tsinghua.edu.cn/jenkins$g' default.json  
sed -i 's$http://www.google.com$https://www.baidu.com$g' default.json
```

8.把advance中的地址，换成清华大学的地址

```
https://mirrors.tuna.tsinghua.edu.cn/jenkins/updates/updatecenter.json
```

9.启动jenkins

```
systemctl start jenkins
```

访问jenkins的地址为：ip+8888

Jenkins 安装后配置和访问

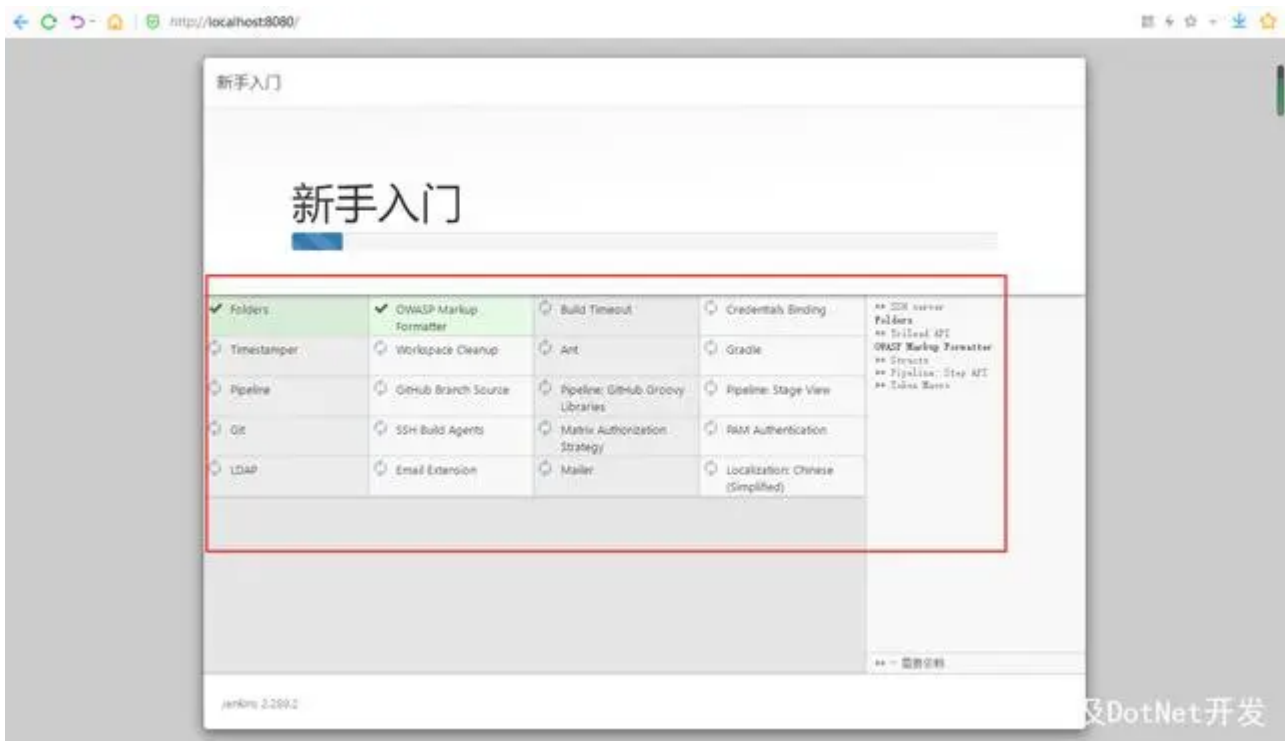
1.Jenkins安装完毕后会自动在浏览器中打开默认端口，第一次登陆Jenkins时需要先进行解锁，按照路径提示找到密码，可以使用记事本打开该文件，然后输入管理员密码进行解锁Jenkins，然后点击“继续”，如下图所示：



2. 在打开的安装插件界面，选择“安装推荐的插件”进行安装，如下图所示：



3. 插件安装需要一段时间，安装过程如下图所示：



4. 插件安装完毕后，进入创建管理员用户界面，这里需要我们创建一个管理员，然后点击“保存并完成”如下图所示：



5. 在实例配置界面选择“保存并完成”，如下图所示：



6. 提示Jenkins已经就绪，点击“开始使用Jenkins”，如下图所示：



7. Jenkins主界面如下图所示：这里的话，**我们自己设置端口为8888，访问的时候记得改一下端口号**

