链滴

jenkins 安装及配置部署操作 （jenkins+svn+tomcat and jenkins+git+maven+tomcat）

作者：mason-roy
原文链接：https://ld246.com/article/1570168906097
来源网站：链滴
许可协议：署名-相同方式共享 4.0 国际 (CC BY-SA 4.0)

<img src="https://ld246.com/images/img-loading.svg" alt="" data-src="https://b3logfile
com/bing/20190413.jpg?imageView2/1/w/960/h/540/interlace/1/q/100">
jenkins rpm 软件包下载地址： <a href="https://ld246.com/forward?goto=http
%3A%2F%2Fpkg.jenkins.io%2Fredhat%2F" target="_blank" rel="nofollow ugc">https://pkg.je
kins.io/redhat/
jenkins 软件包下载地址： <a href="https://ld246.com/forward?goto=https%3A%2F%2Fj
nkins.io%2Fdownload%2F" target="_blank" rel="nofollow ugc">https://jenkins.io/download/
/a>
jenkins 插件下载地址： <a href="https://ld246.com/forward?goto=http%3A%2F%2Fftp
icm.edu.pl%2Fpackages%2Fjenkins%2Fplugins%2F" target="_blank" rel="nofollow ugc">http:
/ftp.icm.edu.pl/packages/jenkins/plugins/
文档相关软件下载链接
链接：<a href="https://ld246.com/forward?goto=https%3A%2F%2Fpan.baidu.com%2Fs%2F1
zs_rjghH1d6Y4VN1KgYIA" target="_blank" rel="nofollow ugc">https://pan.baidu.com/s/1_zs_
jghH1d6Y4VN1KgYIA
提取码：06lb
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
</blockquote>
</blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
<h2 id="jenkins-安装配置-----">jenkins 安装配置 >>>></h2>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
</blockquote>
</blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
因为这的 jenkins 的配置都一样，所以只写一次
1、安装 jdk ，配置好环境变量 JAVA_HOME，后面会有用到 。
<pre><code class="highlight-chroma">[root@jenkins ~]# tar zxf jdk1.8.0_191.tar.gz
[root@jenkins ~]# mv jdk1.8.0_191 /usr/local/jdk1.8.0
[root@jenkins ~]# echo "export JAVA_HOME=/usr/local/jdk1.8.0" >> /etc/profile
[root@jenkins ~]# echo "export PATH=$JAVA_HOME/bin:$PATH" >> /etc/profile
[root@jenkins ~]# source /etc/profile
[root@jenkins ~]# echo $JAVA_HOME
/usr/local/jdk1.8.0
</code></pre>
** 2、安装 jenkins 软件，可以使用 rpm 也可以使用 war 包去运行 。 jenkins 安装依赖 openjk 软件。**
<pre><code class="highlight-chroma">
[root@jenkins ~]# yum localinstall jenkins-2.195-1.1.noarch.rpm -y
[root@jenkins ~]# yum install java-1.8.0-openjdk -y
[root@jenkins ~]# systemctl start jenkins
[root@jenkins ~]# systemctl enable jenkins
[root@jenkins ~]# /sbin/chkconfig jenkins on
</code></pre>
3、打开浏览器 http://ip:port 登陆 jenkins ，密码在下面文件内 。
<pre><code class="highlight-chroma">

[root@jenkins ~]# cat /var/lib/jenkins/secrets/initialAdminPassword
f50bc74962b6449ab7f72cee8ab898f
</code></pre>
<img src="https://ld246.com/images/img-loading.svg" alt="1.png" data-src="https://b3ogfile.com/file/2019/10/1-86a5ad35.png?imageView2/2/interlace/1/format/jpg">
4、网上很多人喜欢“安装推荐的插件”，其实这种方式对于网络不好或者是不能访问网的人来说是不行的，会报错。网络差的会很慢很慢，所以我选 “选择插件来安装”，把所有插件取掉。
<img src="https://ld246.com/images/img-loading.svg" alt="2.png" data-src="https://b3

ogfile.com/file/2019/10/2-af1120cc.png?imageView2/2/interlace/1/format/jpg">
5、取消所有插件，一个都不安装
<img src="https://ld246.com/images/img-loading.svg" alt="3.png" data-src="https://b3ogfile.com/file/2019/10/3-a10f5915.png?imageView2/2/interlace/1/format/jpg">
6、配置管理员用户，一定要配置，不然你后面进去就非常麻烦的。
<img src="https://ld246.com/images/img-loading.svg" alt="4.png" data-src="https://b3ogfile.com/file/2019/10/4-178242c0.png?imageView2/2/interlace/1/format/jpg">
7、输入 jenkins 实例 url ，一般使用默认即可 。
<img src="https://ld246.com/images/img-loading.svg" alt="5.png" data-src="https://b3ogfile.com/file/2019/10/5-07b2140e.png?imageView2/2/interlace/1/format/jpg">
8、安装完毕，开始使用 jenkins
<img src="https://ld246.com/images/img-loading.svg" alt="6.png" data-src="https://b3ogfile.com/file/2019/10/6-4bfe0cd4.png?imageView2/2/interlace/1/format/jpg">
9、登陆后的界面 。
<img src="https://ld246.com/images/img-loading.svg" alt="7.png" data-src="https://b3ogfile.com/file/2019/10/7-69789f5a.png?imageView2/2/interlace/1/format/jpg">
10、选择管理 jenkins ，插件管理。
<img src="https://ld246.com/images/img-loading.svg" alt="8.png" data-src="https://b3ogfile.com/file/2019/10/8-90d1715f.png?imageView2/2/interlace/1/format/jpg">
11、发现已安装插件内，没有任何插件 。
<img src="https://ld246.com/images/img-loading.svg" alt="9.png" data-src="https://b3ogfile.com/file/2019/10/9-55d7be81.png?imageView2/2/interlace/1/format/jpg">
==========
安装插件的三种方式：
插件安装一：
找一台已经安装好插件的机器，把/var/lib/jenkins/plugins 目录里面的文件跟目录全部拷贝出来。
插件对应的文件导入到 /var/lib/jenkins/plugins 目录下面，然后修改权限 chown -R jenkins:jenkins *
重启 systemctl restart jenkins
插件安装二：
到 http://ftp.icm.edu.pl/packages/jnkins/plugins/ 下载对应的插件，选择插件管理里面的高级，上传插件--> 选择文件-->传 （jenkins 2.175 用的是 hpi 文件），上传完之后就会自动开始安装。这种方法比较慢，需要一个件一个文件的选择。
插件安装三：
到可选插件内勾选自己想要的插件，点击安装即可 。
以下是以完成插件安装的截图 ：
<img src="https://ld246.com/images/img-loading.svg" alt="10.png" data-src="https://b3ogfile.com/file/2019/10/10-8a9d9e82.png?imageView2/2/interlace/1/format/jpg">
12、配置后面要用到管理的参数，publish over ssh key ，管理 jenkins --> 系统置 ，找到这里将 jenkins 的 key 放入这里，下面配置后期需要推送文件的服务器 ip，远程账号以及程上去的目录位置。 （jenkins 与需要推送文件的服务器之间需要免密登陆，远程账号不一定用 root 也可以用其他普通用户）

jenkins 的安装配置到此结束；
<h2 id="一-jenkins-svn-tomcat">一、jenkins+svn+tomcat</h2>
项目说明：
1、svn 服务器用于存放页面代码以及更新版本库 ，所有的页面从 svn 上面进行控制 ；
2、tomcat 用来部署页面，所有的访问都通过 tomcat 进行 ；
3、jenkins 用来做更新版本控制，需要带有回滚版本的功能 ；
搭建思路：
1、开发人员统一将文件上传到 svn 对应的目录下面。例如 svn://xxx/work1/dist.zip

2、jenkins 收集到所有的 dist.zip 之后 先打包一个压缩文件到 workspace/dist/xxx-x.tar.gz ， 用于面还原用
3、jenkins 推送.tar.gz 文件到 tomcat 对应的目录下并解压，解压之前先清空里面的文件 。
192.168.255.130 jenkins - 2.195
192.168.255.100 svn - 1.7.14
192.168.255.101 tomcat - 8
<img src="https://ld246.com/images/img-loading.svg" alt="12.png" data-src="https://b3ogfile.com/file/2019/10/12-b77724a8.png?imageView2/2/interlace/1/format/jpg">
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
</blockquote>
</blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
svn 搭建及配置 >>>>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
</blockquote>
</blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
**1、安装 svn **
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@svn ~]# yum install subversion -y
</code></pre>
2、创建 svn 库对应的目录
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@svn ~]# mkdir -p /server/svn
[root@svn ~]# sv
admin create /server/svn
[root@svn ~]# ls /
erver/svn/
conf db format
ooks locks README.txt
</code></pre>
3、对 svn 版本库进行配置
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@svn ~]# cd /server/svn/conf/
[root@svn conf]#
s
authz passwd sv
serve.conf
</code></pre>
4、添加登陆用户名跟密码，均为 appuser
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@svn conf]# tail -2 passwd
    [users]
    appuser = appu
er
</code></pre>
5、设置用户访问权限，appuser 可以对 svn 的 / 有读写权限，其他用户没有权限</st
ong>
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@svn conf]# tail -3 authz
    [/]
    appuser = rw
    * =
</code></pre>
6、配置 svn 密码库文件、权限库文件以及 svn 根的对应目录
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@svn conf]# vim svnserve.conf
    anon-access =
ead
    auth-access =
rite
    password-db =
passwd
    authz-db = aut
z
    realm = /server

svn
</code></pre>
7、启动 svn 服务
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@svn conf]# svnserve -d -r /server/svn/
[root@svn conf]#
ps -ef| grep svnserve
root   2391   1
 0 16:22 ?   00:00:00 svnserve -d -r /server/svn/
root   2393 21
7 0 16:22 pts/0   00:00:00 grep --color=auto svnserve
</code></pre>
8、创建 svn 客户端目录并检出库内容
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@svn conf]# mkdir -p /data/tomcat
[root@svn conf]#
vn checkout svn://192.168.255.100/   /data/tomcat
[root@svn conf]#
d /data/tomcat/
</code></pre>
9、添加文件
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@svn tomcat]# echo "test 16:40" > 1.txt
</code></pre>
10、将文件添加到 svn 库
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@svn tomcat]# svn add 1.txt
A   1.txt
</code></pre>
11、提交版本更新并备注提交信息
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@svn tomcat]# svn commit -m 'test create 1.txt' ./
Adding   1.txt
Transmitting file d
ta .
Committed revisi
n 1.
</code></pre>
12、查看 svn 状态
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@svn tomcat]# svn status -v
                 0   0
?   .
                 1   1
appuser   1.txt
</code></pre>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
tomcat 搭建及配置 >>>>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
安装 openjdk 跟 tomcat 并启动，使用默认配置即可
<pre><code class="highlight-chroma">[root@tomcat ~]# yum install java-1.8.0-openjdk -y
[root@tomcat ~]# tar zxvf apache-tomcat-8.0.30.tar.gz
[root@tomcat ~]# mv apache-tomcat-8.0.30 /usr/local/tomcat

[root@tomcat ~]# cd /usr/local/tomcat/
[root@tomcat to

cat]# bin/startup.sh
</code></pre>
确认服务是否起来
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@tomcat tomcat]# netstat -anputl| grep LISTEN
tcp   0   0 0.0
0.0:22   0.0.0.0:*   LISTEN   923/sshd
tcp   0   0 127
0.0.1:25   0.0.0.0:*   LISTEN   1169/master
tcp6   0   0 :::
080   :::*   LISTEN   2345/java
tcp6   0   0 :::
2   :::*   LISTEN   923/sshd
tcp6   0   0 ::1
25   :::*   LISTEN   1169/master
tcp6   0   0 12
.0.0.1:8005   :::*   LISTEN   2345/java
tcp6   0   0 :::
009   :::*   LISTEN   2345/java
</code></pre>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
</blockquote>
</blockquote>

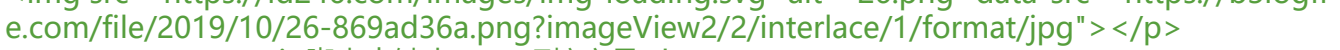
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
jenkins 项目部署配置 >>>>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
</blockquote>
</blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
1、创建项目 -->> 构建一个自由风格的软件项目；
<img src="https://ld246.com/images/img-loading.svg" alt="13.png" data-src="https://b3ogfile.com/file/2019/10/13-1c3ab2fd.png?imageView2/2/interlace/1/format/jpg">
2、选择参数化构建过程 （这里的参数化将会在后面项目构建处进行体现）
<img src="https://ld246.com/images/img-loading.svg" alt="14.png" data-src="https://b3ogfile.com/file/2019/10/14-25425204.png?imageView2/2/interlace/1/format/jpg">
**
3、配置 svn 库地址跟 svn 账号密码**
<img src="https://ld246.com/images/img-loading.svg" alt="15.png" data-src="https://b3ogfile.com/file/2019/10/15-2e954440.png?imageView2/2/interlace/1/format/jpg">
4、选择用户名跟密码，配置 appuser （svn 的账号密码）
<img src="https://ld246.com/images/img-loading.svg" alt="16.png" data-src="https://b3ogfile.com/file/2019/10/16-ec7bef5b.png?imageView2/2/interlace/1/format/jpg">
5、选择刚才配置的账号密码作为凭证 （凭证也可以在 jenkins 设置里面进行添加）
<img src="https://ld246.com/images/img-loading.svg" alt="17.png" data-src="https://b3ogfile.com/file/2019/10/17-f19179ca.png?imageView2/2/interlace/1/format/jpg">
6、添加构建步骤，选择执行 shell 脚本
<img src="https://ld246.com/images/img-loading.svg" alt="18.png" data-src="https://b3ogfile.com/file/2019/10/18-283865ae.png?imageView2/2/interlace/1/format/jpg">
脚本内容即为判断前面 参数化构建过程 内的参数，首先匹配名称，如果名称匹配成功开始匹配下面的 Deploy 或者 RollBACK ，执行对应的语句。 目前对于打包时会做一个判断，保留个打包备份，能够很好的节约 jenkins 的空间。
这里只是做了打包备份，并没有做发布
<img src="https://ld246.com/images/img-loading.svg" alt="19.png" data-src="https://b3ogfile.com/file/2019/10/19-13728904.png?imageView2/2/interlace/1/format/jpg">
脚本的文字
<pre><code class="highlight-chroma">case $Status in
    Deploy)
        echo "Status: $Status"
        path="${WORKSPACE}/dist"
        if [-d $path]; then

```
</span></span><span class="highlight-line"><span class="highlight-cl">            echo "The f
le is already exists!!!"
</span></span><span class="highlight-line"><span class="highlight-cl">        else
</span></span><span class="highlight-line"><span class="highlight-cl">            mkdir -p $
ath
</span></span><span class="highlight-line"><span class="highlight-cl">        fi
</span></span><span class="highlight-line"><span class="highlight-cl">        cd ${WORKS
ACE}
</span></span><span class="highlight-line"><span class="highlight-cl">        tar czf dist/${
OB_NAME}-${BUILD_NUMBER}.tar.gz * --exclude=dist
</span></span><span class="highlight-line"><span class="highlight-cl">        echo "Compl
tion!!!"
</span></span><span class="highlight-line"><span class="highlight-cl">        cd  dist
</span></span><span class="highlight-line"><span class="highlight-cl">        total_file_nu
=`ls -l | grep -v 'total' | wc -l`
</span></span><span class="highlight-line"><span class="highlight-cl">            if [ $total_fi
e_num -gt 10 ];
</span></span><span class="highlight-line"><span class="highlight-cl">            then
</span></span><span class="highlight-line"><span class="highlight-cl">                num=`e
pr $total_file_num - 10`
</span></span><span class="highlight-line"><span class="highlight-cl">                files=`ls
ltr | grep -v 'total' | awk '{print $9}' | head -n $num`
</span></span><span class="highlight-line"><span class="highlight-cl">                echo $fi
es | xargs rm -f
</span></span><span class="highlight-line"><span class="highlight-cl">                echo "
留10个备份，删除早期备份: " $files
</span></span><span class="highlight-line"><span class="highlight-cl">            else
</span></span><span class="highlight-line"><span class="highlight-cl">                echo "
有更多的版本"
</span></span><span class="highlight-line"><span class="highlight-cl">            fi
</span></span><span class="highlight-line"><span class="highlight-cl">        ;;
</span></span><span class="highlight-line"><span class="highlight-cl">    RollBACK)
</span></span><span class="highlight-line"><span class="highlight-cl">        echo "Status:
$Status"
</span></span><span class="highlight-line"><span class="highlight-cl">        echo "version
 $Version"
</span></span><span class="highlight-line"><span class="highlight-cl">        file_old=`ls $
WORKSPACE}/dist | grep $Version`
</span></span><span class="highlight-line"><span class="highlight-cl">        cd ${WORKS
ACE}/dist
</span></span><span class="highlight-line"><span class="highlight-cl">        cp -R $file_ol
 ${JOB_NAME}-${BUILD_NUMBER}.tar.gz
</span></span><span class="highlight-line"><span class="highlight-cl">        ;;
</span></span><span class="highlight-line"><span class="highlight-cl">    *)
</span></span><span class="highlight-line"><span class="highlight-cl">    exit
</span></span><span class="highlight-line"><span class="highlight-cl">        ;;
</span></span><span class="highlight-line"><span class="highlight-cl">esac
</span></span></code></pre>
<p><strong>7、添加第二个构建步骤，此处先关闭 tomcat，然后删除需要上传文件的目录里面的
容。 （也可以把脚本写到 tomcat 的服务器里面，写这里是为了能够让管理者更加直观的看到做了
么）<br>
jenkins 的构建步骤都是严格按照先后顺序来执行的，所以要做什么操作请按照顺序来； </strong><
p>
<p><img src="https://ld246.com/images/img-loading.svg" alt="20.png" data-src="https://b3
```

ogfile.com/file/2019/10/20-a0f9f8ea.png?imageView2/2/interlace/1/format/jpg">
<img src="https://ld246.com/images/img-loading.svg" alt="21.png" data-src="https://b3ogfile.com/file/2019/10/21-df2dd752.png?imageView2/2/interlace/1/format/jpg">
8、将之前打包的压缩包发送到 tomcat 上面并执行解压，解压完毕之后执行启动 tomat 命令。 （这个可以放在构建步骤里面也可以放在构建后操作步骤里面，效果是一样的，只要顺序错）
<img src="https://ld246.com/images/img-loading.svg" alt="22.png" data-src="https://b3ogfile.com/file/2019/10/22-f6652a23.png?imageView2/2/interlace/1/format/jpg">
<img src="https://ld246.com/images/img-loading.svg" alt="23.png" data-src="https://b3ogfile.com/file/2019/10/23-f49a5ee9.png?imageView2/2/interlace/1/format/jpg">
因为项目需求不同，所以这里做的都是全量版本更新，如果想要做 svn 的增量版本更的话可以使用下面的操作方式
内容来源： http://blog.csdn.net/q13554515812/article/details/86651851
进入【系统管理】-【Jenkins 命令行接口】，进入【Jenkins 命令行】页面，下载 jenkins-cli.jar，到 Svn 所在服务器的 root 目录下，如下图：
（发布项目的用户必须要有可执行权限，默认的项目执行用户家目录配置在 系统配置- -> publish over ssh key -> ssh servers -> Remote Directory ）
<img src="https://ld246.com/images/img-loading.svg" alt="24.png" data-src="https://b3logfile.com/file/2019/10/24-91e6c5c3.png?imageView2/2/interlace/1/format/jpg">
进入 Svn 的 hooks 目录下，创建文件 post-commit，并赋予执行权限，如下图：
<img src="https://ld246.com/images/img-loading.svg" alt="25.png" data-src="https://b3logfile.com/file/2019/10/25-4f95213f.png?imageView2/2/interlace/1/format/jpg">
编辑文件 post-commit，内容如下：
-auth 为添加认证， test:123qwe 为账号跟密码， build 为打包， test 为项目名称。
<pre><code class="language-bash highlight-chroma">#!/bin/bash
source /etc/profile
java -jar /root/jenkins-cli.jar -s http://47.104.77.127:8080/jenkins/ -auth test:123qwe build test
</code></pre>
svn 提交代码后，查看 Jenkins 任务是否触发。
FAQ
Q：svn 钩子 post-commit 出现 255 错误
R：post-commit 脚本文件的权限不对
S：post-commit 脚本必须有 +x 权限，给 post-commit 添加可执行权限
参见：https://blog.csdn.et/webnoties/article/details/40539431
Q：svn 提交代码，触发 post-commit 钩子，出现如下报错：



进入 Jenkins【系统管理】-【全局安全配置】-【授权策略】选中【登录用户可以做任何事】后保存<
r>
方案二：
命令中添加-auth 参数，修改命令如下：
java -jar /root/jenkins-cli.jar -s <a href="https://ld246.com/forward?goto=http%3A%2F%2F47
104.77.127%3A8080" target="_blank" rel="nofollow ugc">http://47.104.77.127/jenkins/
auth test:123qwe build test
<h2 id="二-jenkins-git-maven-tomcat">二、jenkins+git+maven+tomcat</h2>
项目说明：
1、git 服务器用于存放页面代码以及更新版本库，所有的页面从 git 上面进行控制；
2、tomcat 用来部署页面，所有的访问都通过 tomcat 进行；
3、jenkins 用来做更新版本控制，需要带有回滚版本的功能；
4、git 服务器上面全部均为源代码，需要 Jenkins 具备自动打包功能，减轻开发工作量；
搭建思路：
1、开发人员统一使用 git 来进行代码的上传下载，git 需要具备安全功能；
2、在 jenkins 上面配置 maven，将代码拉取下来之后进行打包再发送到 tomcat 上面进行部署；<b
>
3、jenkins 推送文件到 tomcat 对应的目录下之前先停止 tomcat，推送完之后启动 tomcat；
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
</blockquote>
</blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
git 部署配置 >>>>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
</blockquote>
</blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
首先到 github 上面注册一个账号，创建一个私有仓库并将源代码上传上去。</stron
>
将服务器公钥放到 git 服务端；
<img src="https://ld246.com/images/img-loading.svg" alt="27.png" data-src="https://b3
ogfile.com/file/2019/10/27-70088bb2.png?imageView2/2/interlace/1/format/jpg">
<img src="https://ld246.com/images/img-loading.svg" alt="28.png" data-src="https://b3
ogfile.com/file/2019/10/28-a7add000.png?imageView2/2/interlace/1/format/jpg">
**jenkins 服务器上面需要安装 git **
<pre><code class="highlight-chroma"><span class="highlight
cl">[root@jenkins ~]# yum install git -y
[root@jenkins ~]#
git --version
  git version 1.8.3
1
</code></pre>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
maven 部署配置 >>>>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
<pre><code class="highlight-chroma">[root@jenkins ~]# tar zxvf apache-maven-3.6.2-bin.tar.gz
[root@jenkins ~]# mv apache-maven-3.6.2 /usr/local/maven-3.6.2

[root@jenkins ~]# cd /usr/local/maven-3.6.2/bin/
[root@jenkins bin# ./mvn -v
    Apache Maven .6.2 (40f52333136460af0dc0d7232c0dc0bcf0d9e117; 2019-08-27T11:06:16-04:00)
    Maven home: /sr/local/maven-3.6.2
    Java version: 1.80_201, vendor: Oracle Corporation, runtime: /usr/local/jdk1.8.0/jre
    Default locale: n_US, platform encoding: UTF-8
    OS name: "linux , version: "3.10.0-514.el7.x86_64", arch: "amd64", family: "unix"
</code></pre>
需要配置好 jdk， 参考第一个项目的 jdk 配置；
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
tomcat 部署配置 >>>>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>

</blockquote>
</blockquote>
</blockquote>

cd /usr/local/tomcat/
[root@tomcat tomcat]# bin/startup.sh
</code></pre>
确认服务是否起来
<pre><code class="highlight-chroma">[root@tomcat tomcat]# netstat -anputl| grep LISTEN
tcp   0   0 0.0.0.0:22   0.0.0.0:*   LISTEN   923/sshd
tcp   0   0 127.0.0.1:25   0.0.0.0:*   LISTEN   1169/master
tcp6   0   0 :::8080   :::*   LISTEN   2345/java
tcp6   0   0 :::22   :::*   LISTEN   923/sshd
tcp6   0   0 ::1:25   :::*   LISTEN   1169/master
tcp6   0   0 127.0.0.1:8005   :::*   LISTEN   2345/java
tcp6   0   0 :::8009   :::*   LISTEN   2345/java
</code></pre>
网络上面有很多都是使用 tomcat 自带的 manager 来进行编译后发布的，但是因为这个不太稳，经常会出现问题，所以这里不做配置。 使用另外一种方式来编译发布 。
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
</blockquote>
</blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
Jenkins 部署配置 >>>>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>

<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
<blockquote>
</blockquote>
</blockquote>

</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
</blockquote>
1、配置 jenkins 里面的 maven、git、jdk 路径
<img src="https://ld246.com/images/img-loading.svg" alt="29.png" data-src="https://b3ogfile.com/file/2019/10/29-e73f606b.png?imageView2/2/interlace/1/format/jpg">
<img src="https://ld246.com/images/img-loading.svg" alt="30.png" data-src="https://b3ogfile.com/file/2019/10/30-11fa678d.png?imageView2/2/interlace/1/format/jpg">
2、创建一个 maven 项目 。
<img src="https://ld246.com/images/img-loading.svg" alt="31.png" data-src="https://b3ogfile.com/file/2019/10/31-c22dfddc.png?imageView2/2/interlace/1/format/jpg">
3、添加 jenkins 凭证，为 git 添加 key ， key 就是 jenkins 的私钥。
<img src="https://ld246.com/images/img-loading.svg" alt="32.png" data-src="https://b3ogfile.com/file/2019/10/32-37ecbb0b.png?imageView2/2/interlace/1/format/jpg">
此处填写 git 的地址。
<img src="https://ld246.com/images/img-loading.svg" alt="33.png" data-src="https://b3logfie.com/file/2019/10/33-bde15239.png?imageView2/2/interlace/1/format/jpg">
4、Build 处使用 pom.xml ， Goals and options 输入 clean install -D maven.test.kip=true ；
<img src="https://ld246.com/images/img-loading.svg" alt="34.png" data-src="https://b3ogfile.com/file/2019/10/34-9487b115.png?imageView2/2/interlace/1/format/jpg">
5、增加构建后操作，选择 send build artifacts over ssh 。
<img src="https://ld246.com/images/img-loading.svg" alt="35.png" data-src="https://b3ogfile.com/file/2019/10/35-6b935ee6.png?imageView2/2/interlace/1/format/jpg">
首先停止 tomcat， 然后删除掉页面数据 。
第二步将打包好的 war 包 发送到服务器上并运行
<img src="https://ld246.com/images/img-loading.svg" alt="36.png" data-src="https://b3ogfile.com/file/2019/10/36-5d6c210c.png?imageView2/2/interlace/1/format/jpg">
因为在构建当中出现以下报错信息，主要原因是需要页面需要使用到 mysql 以及连接件
<img src="https://ld246.com/images/img-loading.svg" alt="37.png" data-src="https://b3ogfile.com/file/2019/10/37-36126be9.png?imageView2/2/interlace/1/format/jpg">
<img src="https://ld246.com/images/img-loading.svg" alt="38.png" data-src="https://b3ogfile.com/file/2019/10/38-826696c7.png?imageView2/2/interlace/1/format/jpg">
<img src="https://ld246.com/images/img-loading.svg" alt="39.png" data-src="https://b3ogfile.com/file/2019/10/39-2b1977ae.png?imageView2/2/interlace/1/format/jpg">

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl">mysql&gt; create database zrlog_demo ;
</span></span><span class="highlight-line"><span class="highlight-cl">Query OK, 1 row a
fected (0.00 sec)
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">mysql&gt; grant al
 on zrlog_demo.* to zrlog_demo@'%' identified by '123456';
</span></span><span class="highlight-line"><span class="highlight-cl">Query OK, 0 rows
ffected, 1 warning (0.00 sec)
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">mysql&gt;
</span></span><span class="highlight-line"><span class="highlight-cl">mysql&gt; flush pr
vileges;
</span></span><span class="highlight-line"><span class="highlight-cl">Query OK, 0 rows
ffected (0.00 sec)
</span></span></code></pre>
```