**安装文档**

1. **软件环境**

Centos7.5+XShell

**二、JDK安装**

将事先准备好的jdk-8u191-linux-x64.tar.gz通过XShell工具上传至服务器，具体步骤如下：

1. 进入/usr/local目录，创建java文件夹。
2. 将jdk-8u191-linux-x64.tar.gz拖入shell界面，提示-bash: rz: command not found,说明没有安装 lrzsz。
3. 使用命令yum -y install lrzsz，待安装好重新拖入jdk。
4. 使用命令tar -zxvf jdk-8u191-linux-x64.tar.gz进行解压。



图1 解压文件

1. 使用命令vi /etc/profile进行环境的配置。
2. 在文末添加： export JAVA\_HOME=/usr/local/java/jdk

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

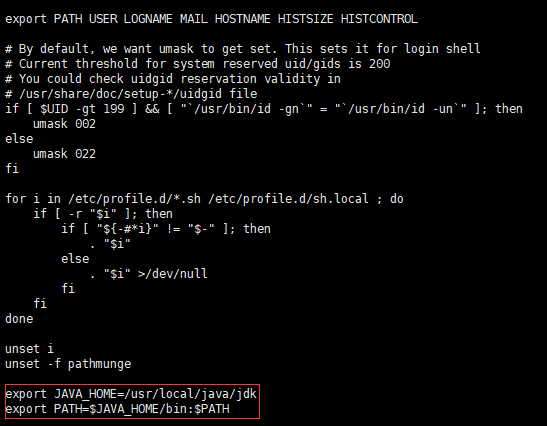


图2 环境配置

1. 保存退出，使用命令source /etc/profile，让环境生效。

8）使用命令java -version检验环境是否生效。结果如图3所示，说明安装成功。



图3 java版本

**三、Tomcat安装**

去官网找到tomcat8的下载界面，下载安装包，如图4所示。

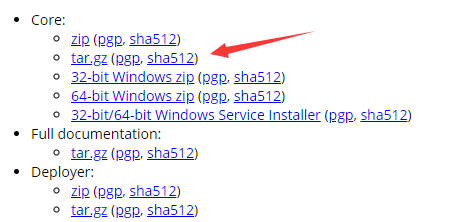


图4 tomcat安装包

下载完成后的具体安装步骤如下：

1. 进入/usr/local目录，创建tomcat文件夹，将下载好的tar包放至当前目录。
2. 使用命令tar -zxvf apache-tomcat-8.5.42.tar.gz进行解压。解压完成如图5所示。



图5 解压 tomcat文件

1. 进入bin目录，使用命令./startup.sh启动tomcat。
2. 在浏览器中输入ip:8080，结果如图6所示。

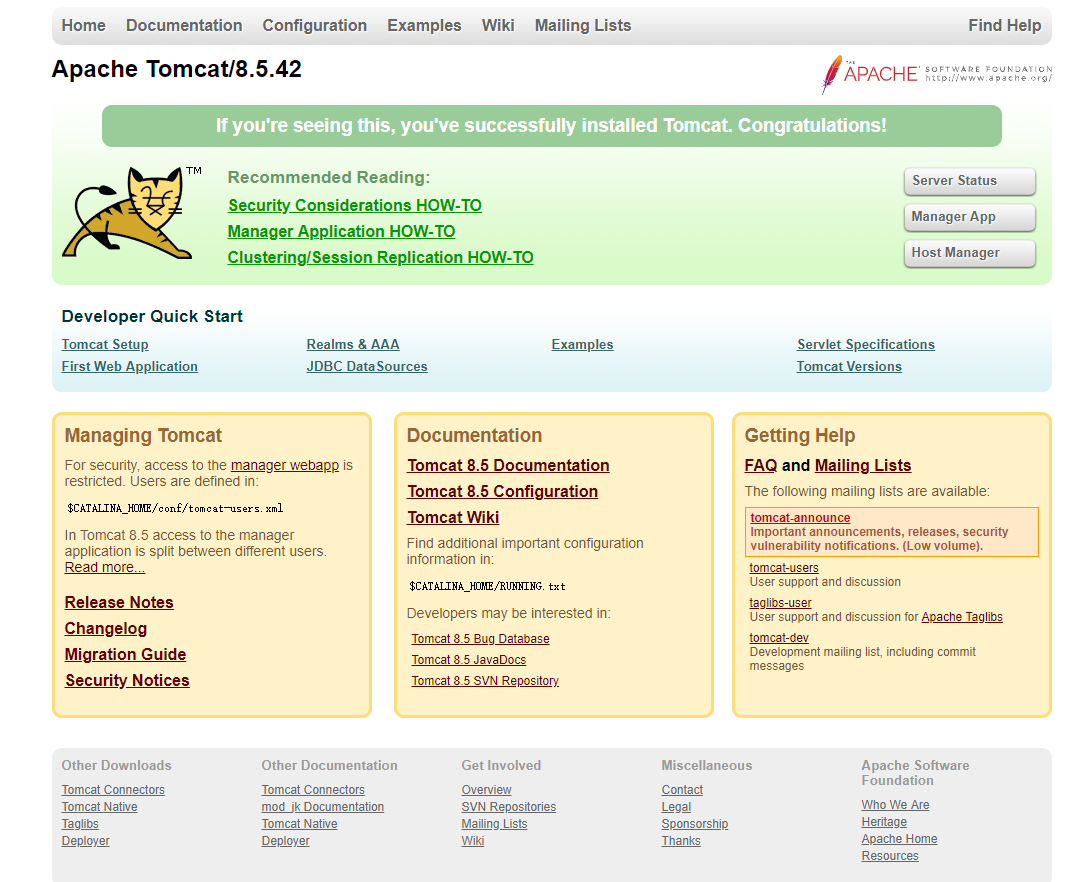


图6 tomcat运行图

Tomcat主要命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| ./startup.sh | 启动tomcat 服务 |
| ./shutdown.sh | 关闭tomcat服务 |

1. **Nginx安装**

Nginx需要配置较多前置条件，具体步骤如下：

1. 下载前置环境，使用命令yum -y install gcc automake autoconf libtool make安装make；使用命令yum install gcc gcc-c++安装g++。
2. 进入/usr/local，创建nginx文件夹。
3. 安装PCRE库。进入nginx文件夹，使用命令wgetftp://ftp.csx.cam.ac.

uk/pub/software/programming/pcre/pcre-8.39.tar.gz 安装PCRE库。解压文件，进入文件中，使用命令./configure；make；make install。



图6 解压pcre

1. 安装openssl。使用命令wget https://www.openssl.org/source/opens

sl-1.0.1t.tar.gz安装openssl，进行解压。



图7 解压openssl

1. 安装zlib-devel。使用命令yum install -y zlib-devel。
2. 安装nginx。使用命令wget http://nginx.org/download/nginx-1.1.10.

tar.gz。使用命令tar -zxvf nginx-1.1.10.tar.gz进行解压，进入文件中，使用命令./configure；make；make install。



图8 解压nginx

1. 使用命令/usr/local/nginx/sbin/nginx -c /usr/local/nginx/conf/nginx.conf进行启动。
2. 在浏览器中输入ip，检查nginx是否启动成功。页面如图9所示。

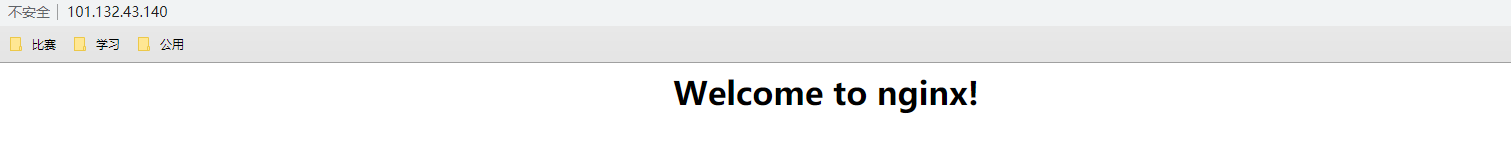


图9 欢迎页面

Nginx主要命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| /usr/local/nginx/sbin/nginx -c /usr/local/nginx/conf/nginx.conf | 启动nginx 服务 |
| /usr/local/nginx/sbin/nginx -t /usr/local/nginx/conf/nginx.conf | 验证配置是否正确 |
| /usr/local/nginx/sbin/nginx -s reload | 重启nginx服务 |
| /usr/localinx/sbininx -s stop | 停止nginx服务 |

1. **Tomcat集群搭建**

Tomcat集群搭建主要修改相关结点，具体步骤如下：

1. 使用命令cp -r tomcat tomcat2，cp -r tomcat tomcat3，拷贝tomcat。
2. 进入conf中的server.xml文件将tomcat2和tomcat3的端口进行修改。需要修改的端口有：

<Server port="8006" shutdown="SHUTDOWN">

<Connector port="8081" protocol="HTTP/1.1"

     connectionTimeout="20000"

redirectPort="8443" />

<Connector port="8010" protocol="AJP/1.3" redirectPort="8443" />

<Server port="8007" shutdown="SHUTDOWN">

<Connector port="8082" protocol="HTTP/1.1"

     connectionTimeout="20000"

redirectPort="8443" />

<Connector port="8011" protocol="AJP/1.3" redirectPort="8443" />

按照顺序，依次启动tomcat,tomcat2,tomcat3。经过测试均可运行。

1. **Nginx实现负载均衡**

1）修改Nginx的server中的配置项，首先要添加upstream，可以修改端口，由于没有占用端口，便没有修改。最后将tomcatCluster加入location /中，添加proxy\_pass一行，保存退出。

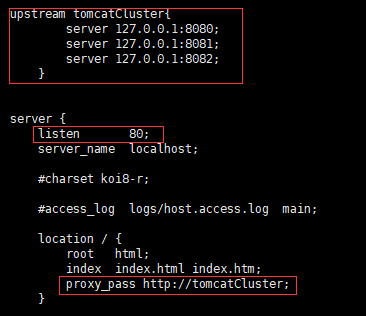


图10 配置项

2）使用命令/usr/local/nginx/sbin/nginx -t -c /usr/local/nginx/conf/nginx.conf

检查配置文件是否正确。



图11 检查结果

3）可观察到配置正确，重新启动nginx。进入sbin目录，使用命令./nginx -s reload。

4）在浏览器中输入ip:80，可以发现已经跳转到tomcat界面。

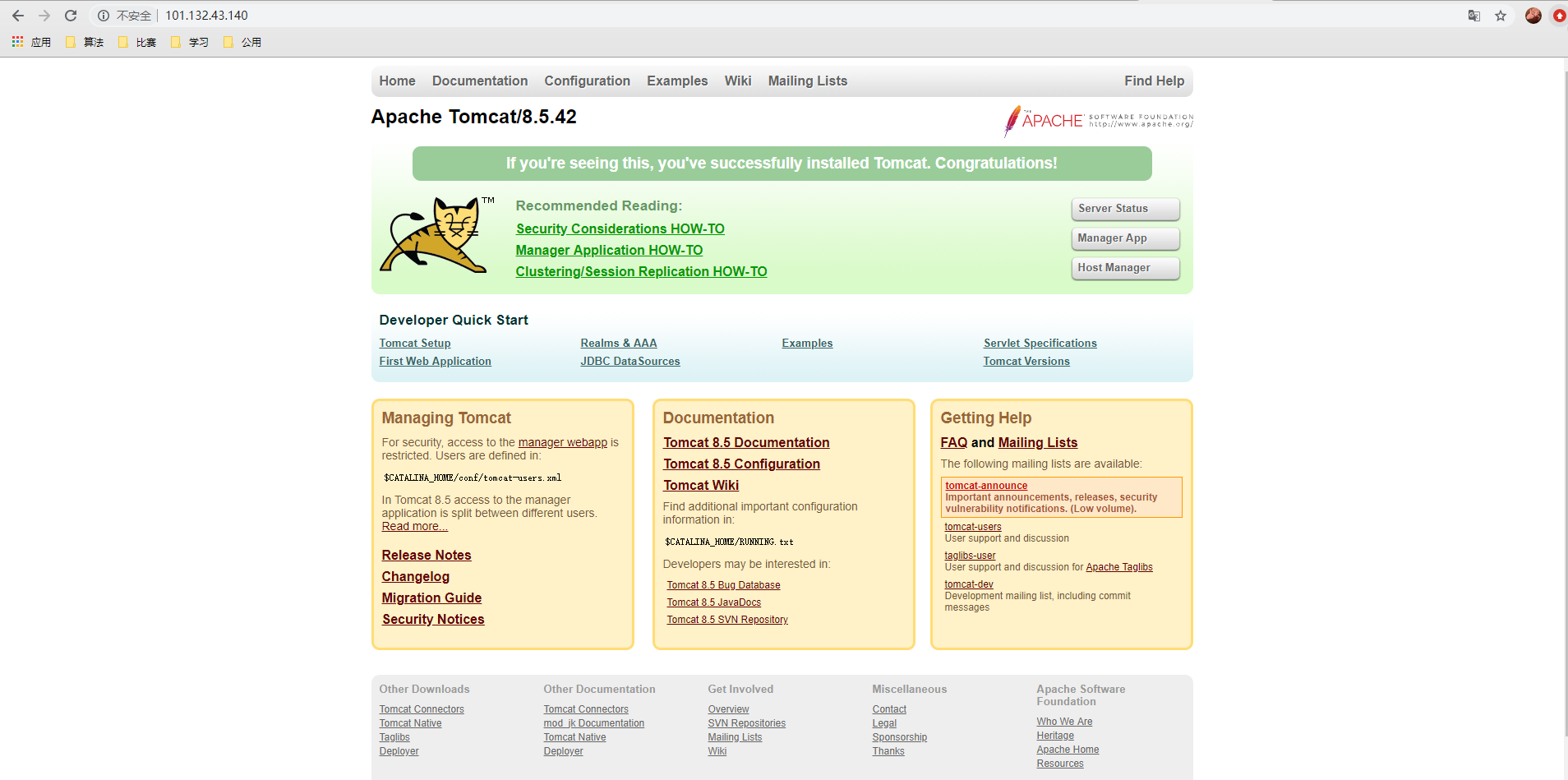


图11 最终结果图

1. **ZooKeeper安装**
2. 将事先准备好的zookeeper上传至服务器的/usr/local目录。
3. 配置zookeeper的环境变量，使用命令vi /etc/profile，添加下面的语句：

export ZK\_HOME=/usr/local/zk/zookeeper

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



图12 zk环境配置图

1. 使用命令source profile使其生效。
2. 进入conf目录，使用命令cp -r zoo\_sample.cfg zoo.cfg拷贝zoo\_sample.cfg为zoo.cfg文件。
3. 修改zoo.cfg文件，找到dataDir，修改数据存储目录，如图13所示，保存退出。



图13 zk地址配置图

1. 在相应的目录创建对应的文件。
2. 进入bin目录，使用命令./zkServer.sh start启动成功。



图14 zk启动成功图

ZooKeeper主要命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| ./zkServer.sh start | 启动zookeeper服务 |
| ./zkServer.sh status | 查看zookeeper状态 |
| ./zkServer.sh stop | 停止zookeeper服务 |
| ./zkServer.sh restart | 重启zookeeper服务 |

1. **MemCache安装**

MemCache首先需要安装libevent库，然后再进行MemCache的安装。具体步骤如下：

1. 创建memcache目录，进入使用wget https://github.com/downloads/libeve

nt/libevent/libevent-1.4.14b-stable.tar.gz命令进行下载。

1. 使用命令tar -zxvf libevent-1.4.14b-stable.tar.gz进行解压。进入文件中，使

用命令./configure；make；make install。

1. 使用命令wget http://www.memcached.org/files/memcached-1.4.20.tar.gz

进行MemCache的下载，解压文件。



图15 memcache解压图

1. 进入目录使用命令./configure --with-libevent=/usr/localbevent/ ；

make；make install进行安装。

1. 使用命令/usr/local/bin/memcached -d -m 64 -I 20m -u root -l 127.0.0.1 -p

11211 -c 1024 -P /usr/local/memcache/memcache/memcached.pid进行启动。

启动参数表

|  |  |
| --- | --- |
| **选项** | **含义说明** |
| -d | 指定memcached进程作为一个守护进程启动 |
| -m <num> | 指定分配给memcached使用的内存，单位是MB |
| -u <username> | 运行memcached的用户 |
| -l <ip\_addr> | 监听的服务器IP地址，如果有多个地址的话，使用逗号分隔，格式可以为“IP地址:端口号”，例如：-l 指定192.168.0.184:19830,192.168.0.195:13542；端口号也可以通过-p选项指定 |
| -p <num> | Memcached监听的端口，要保证该端口号未被占用 |
| -c <num> | 设置最大运行的并发连接数，默认是1024 |
| -R <num> | 为避免客户端饿死（starvation），对连续达到的客户端请求数设置一个限额，如果超过该设置，会选择另一个连接来处理请求，默认为20 |
| -k | 设置锁定所有分页的内存，对于大缓存应用场景，谨慎使用该选项 |
| -P | 保存memcached进程的pid文件 |
| -s <file> | 指定Memcached用于监听的UNIX socket文件 |
| -a <perms> | 设置-s选项指定的UNIX socket文件的权限 |
| -U <num> | 指定监听UDP的端口，默认11211，0表示关闭 |
| -M | 当内存使用超出配置值时，禁止自动清除缓存中的数据项，此时Memcached不可以，直到内存被释放 |
| -r | 设置产生core文件大小 |
| -f <factor> | 用于计算缓存数据项的内存块大小的乘数因子，默认是1.25 |
| -n | 为缓存数据项的key、value、flag设置最小分配字节数，默认是48 |
| -C | 禁用CAS |
| -h | 显示Memcached版本和摘要信息 |
| -v | 输出警告和错误信息 |
| -vv | 打印信息比-v更详细：不仅输出警告和错误信息，也输出客户端请求和响应信息 |
| -i | 打印libevent和Memcached的licenses信息 |
| -t <threads> | 指定用来处理请求的线程数，默认为4 |
| -D <char> | 用于统计报告中Key前缀和ID之间的分隔符，默认是冒号“:” |
| -L | 尝试使用大内存分页（pages） |
| -B <proto> | 指定使用的协议，默认行为是自动协商（autonegotiate），可能使用的选项有auto、ascii、binary。 |
| -I <size> | 覆盖默认的STAB页大小，默认是1M |
| -F | 禁用flush\_all命令 |
| -o <options> | 指定逗号分隔的选项，一般用于用于扩展或实验性质的选项 |

MemCache主要命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| telnet 127.0.0.1 11211 | 连接memcache |
| quit | 退出 |
| set | 修改 |
| add | 添加 |
| replace | 替代 |
| get | 获取 |
| delete | 删除 |

1. **MongoDB安装**

Linux mongodb具体安装过程可以参考官网，官网地址为https://docs.mongod

b.com/manual/tutorial/install-mongodb-on-red-hat/。具体步骤如下：

1. 使用命令cd /etc/yum.repos.d/，进入yum仓库，接着使用命令touch mongodb-org-4.0.repo，创建文件。
2. 在创建的文件中添加如下内容：

[mongodb-org-4.0]

name=MongoDB Repository

baseurl=https://repo.mongodb.org/yum/redhat/$releasever/mongodb-org/4.0/x86\_64/

gpgcheck=1

enabled=1

gpgkey=https://www.mongodb.org/static/pgp/server-4.0.asc保存内容并退出。

1. 使用命令yum install -y mongodb-org进行安装。
2. 使用命令service mongod start 进行启动。



图16 mongodb启动图

mongodb主要命令

|  |  |
| --- | --- |
| 命令 | 作用 |
| service mongod start | 启动mongodb 服务 |
| mongo | 进入mongodb客户端 |
| show dbs | 查看mongodb中的库 |
| chkconfig mongod on | 开机启动 |
| service mongod stop | 停止服务 |
| service mongod restart | 重启服务 |
| service mongod status | 查看服务状态 |
| yum erase $(rpm -qa | grep mongodb-org) | 卸载mongodb |
| rm -r /var/log/mongodb  rm -r /var/lib/mongodb | 删除数据目录 |

1. **Source Tree安装**
2. 打开官网，点击如图所示位置进行下载。

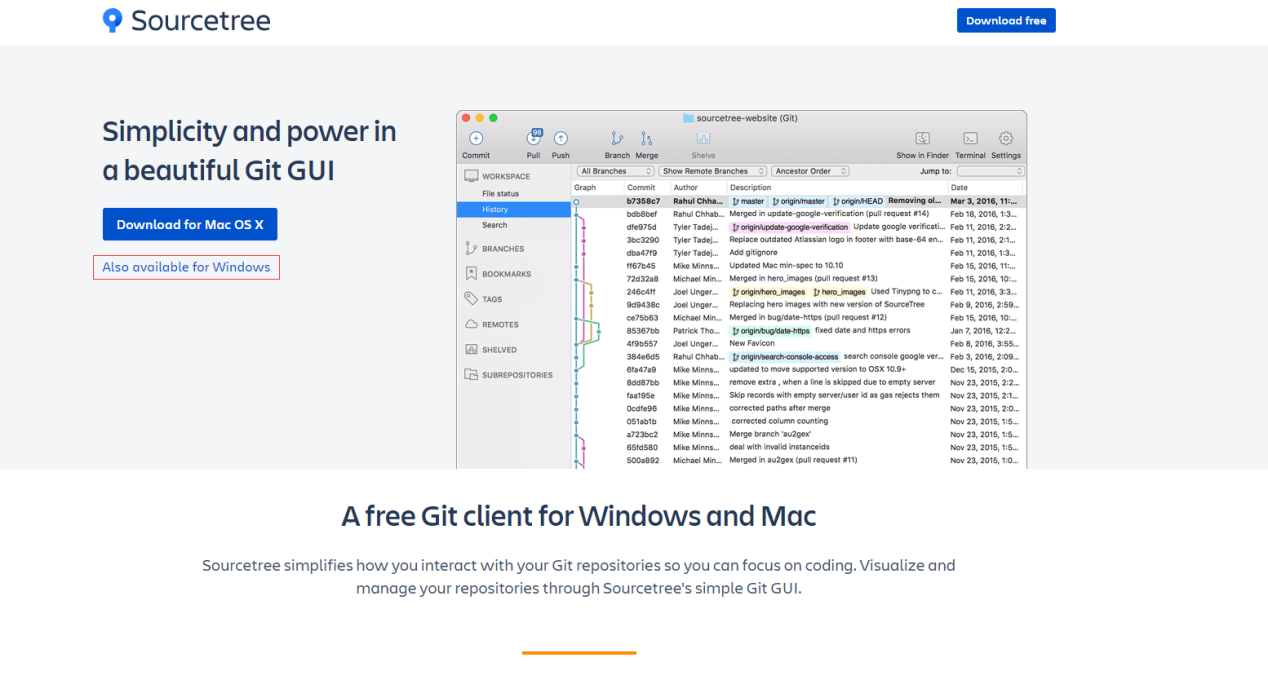


图17 SourceTree下载图

1. 点击下载好的.exe文件，点击登录Bitbucket账号,如图18所示。

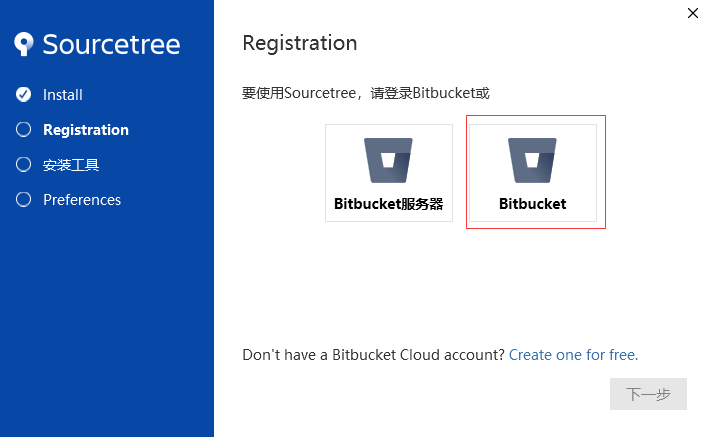


图18 SourceTree安装图

1. 注册Bitbucket账号。

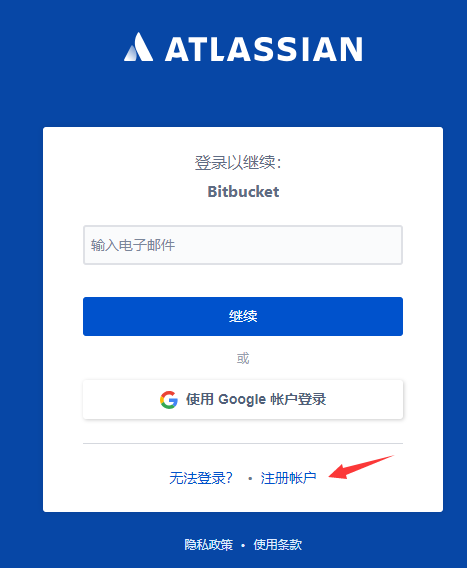


图19 SourceTree安装图

1. 登录账号完毕如图20所示。



图20 SourceTree安装图

1. 点击下一步。



图21 SourceTree安装图

1. 安装完成，如图22所示。

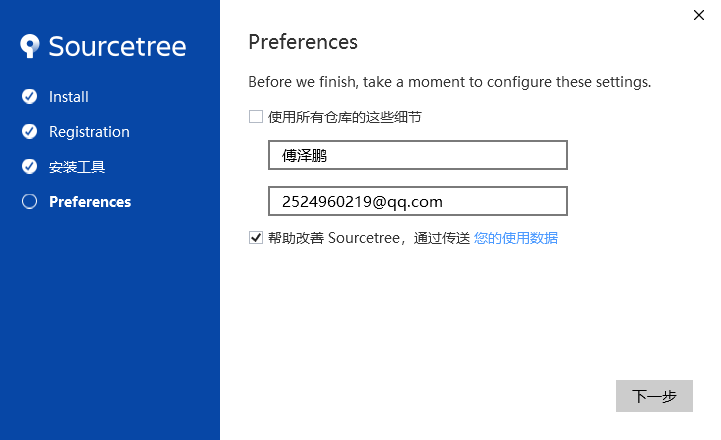


图22 SourceTree安装图

1. 点击编辑账户，设置open ssh，source tree会自动识别本机上已有的ssh。

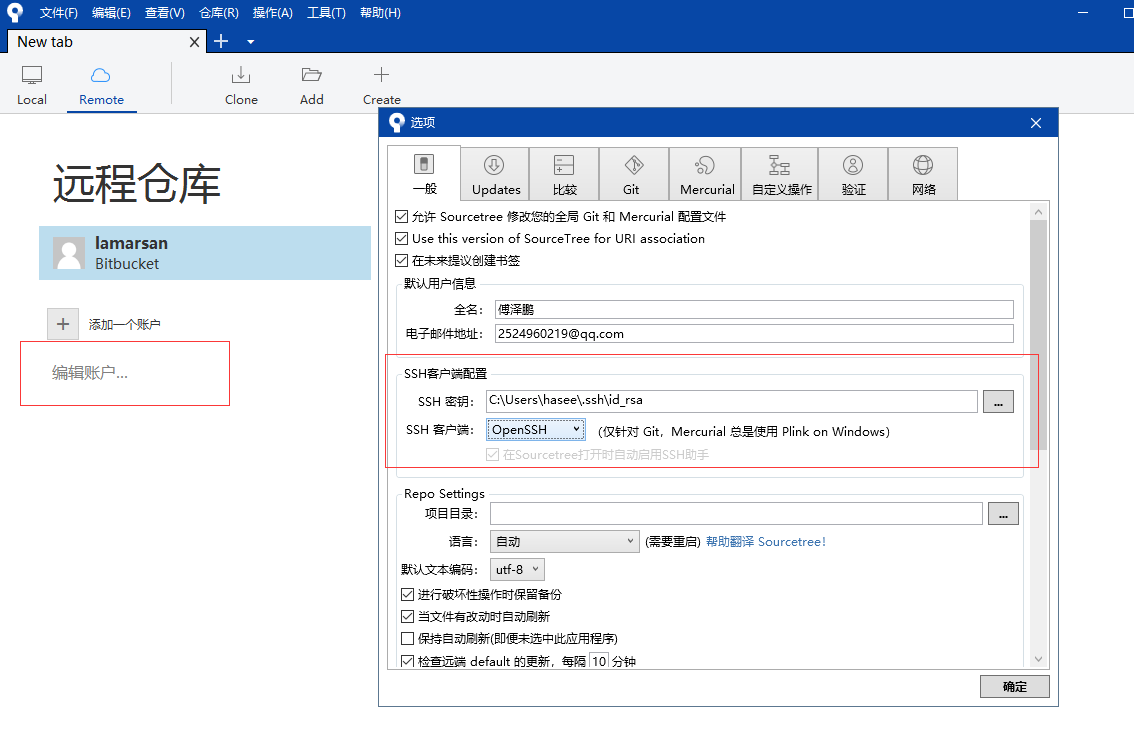


图23 ssh添加图

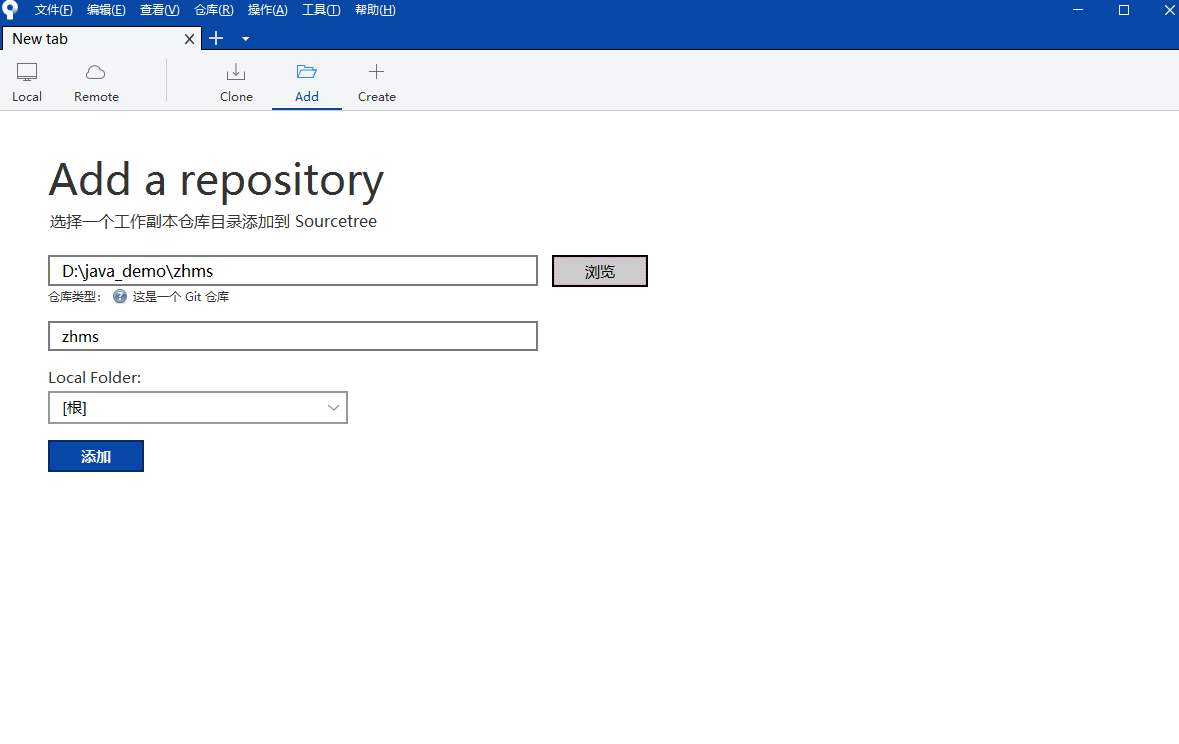
1. 添加已有的仓库，点击add，选择本地仓库目录。  
   

图22 添加仓库图

1. 拉取远程仓库文件，点击项目上的拉取即可，如图23所示。

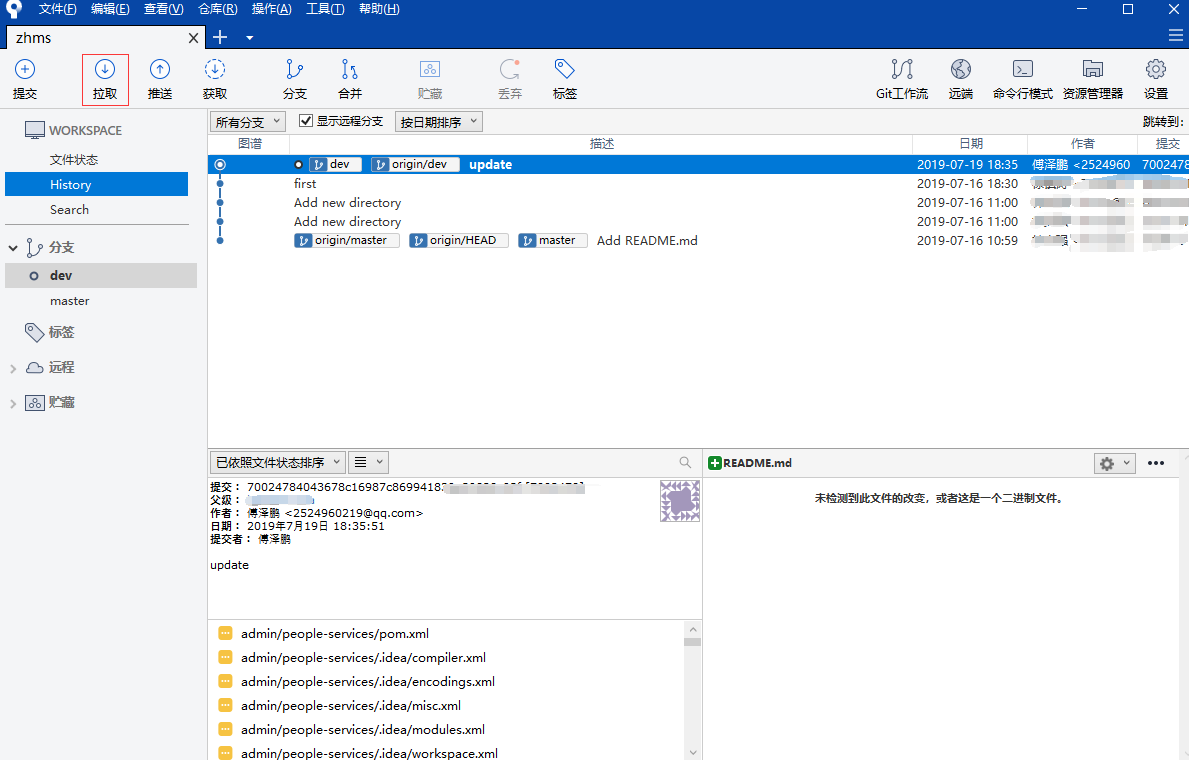


图23 添加远程仓库图

1. **kafka-manager安装**
2. 阿里镜像服务：

<https://help.aliyun.com/document_detail/60750.html?spm=a2c4g.11186623.6.549.24b97418DLTV3i>。



1. 配置完毕后，使用docker拉取kafka-manager。

docker pull sheepkiller/kafka-manager

1. 启动kafka-manager：

docker run -d -p 9000:9000 -eZK\_HOSTS=192.168.9.91 docker.io/sheepkiller/kafka-manager

1. **RocketMQ安装**
2. 下载并解压：

<https://www.apache.org/dyn/closer.cgi?path=rocketmq/4.5.2/rocketmq-all-4.5.2-bin-release.zip>。



1. 修改Java内存大小：runserver.sh与brokerserver.sh。
2. 配置conf/broker.conf，添加：

namesrvAddr=101.132.43.140:9876

brokerIP1=101.132.43.140

1. 启动namesrv命令：nohup bin/mqnamesrv &。



1. 启动broker命令：nohup sh bin/mqbroker -c conf/broker.conf &。

Java HotSpot(TM) 64-Bit Server VM warning: Using the DefNew young collector with the CMS collector is deprecated and will likely be removed in a future release

Java HotSpot(TM) 64-Bit Server VM warning: UseCMSCompactAtFullCollection is deprecated and will likely be removed in a future release.

Java HotSpot(TM) 64-Bit Server VM warning: MaxNewSize (2097152k) is equal to or greater than the entire heap (2097152k). A new max generation size of 2097088k will be used.

The Name Server boot success. serializeType=JSON

The broker[c.com, 192.168.9.93:10911] boot success. serializeType=JSON and name server is 192.168.9.93:9876

1. 关闭命令：mqshutdown namesrv/mqshutdown broker
2. 若要外网访问，开放9876与10911端口。
3. **遇到的问题**
4. 搭建Tomcat集群的时候，执行./shutdown.sh命令抛出异常，提示SEVERE: Could not contact [localhost:[8005]]. Tomcat may not be running.说明Tomcat可能没有启动成功。通过查找相关资料，进行问题排查，发现是jdk的一个漏洞，进入jdk/jre/lib/security，修改java.security，将“securerandom.source = file：/ dev / random”修改为“securerandom.source = file：/ dev /./ urandom”。

2）启动memcache的时候报了如下的错误：/usr/local/bin/memcached: error while loading shared libraries: libevent-1.4.so.2: cannot open shared object file: No such file or directory。没有找到libraries库，使用命令find / -name libevent-2.1.so.6，发现在/usr/local/lib/libevent-1.4.so.2中，使用软连接命令 ln -s /usr/local/lib/libe

vent-1.4.so.2 /usr/lib64/libevent-1.4.so.2，再次启动，启动成功。