**关于阿里云轻量型应用服务器部署java web项目**

实际上本来是想购买云服务器ECS的，但是由于重度手残买成了轻量型应用服务器，郁闷了很久。相比于传统ECS，服务器的配置简化了，带宽也明显提升，当然轻量二字不是随便说说的，有利也有弊。它的配置方式和ECS基本是一样的，但这里也把配置方式讲一下。

**一.购买轻量型应用服务器**

对于学生党们，阿里云有针对学生的云翼计划，每个月10块。

所以首先自己要先注册阿里云，然后经过个人验证以及学生认证，最后通过后就可以开始购买我们的学生服务器。

关于云翼计划，直接在百度上搜云翼计划进入后就可以了。

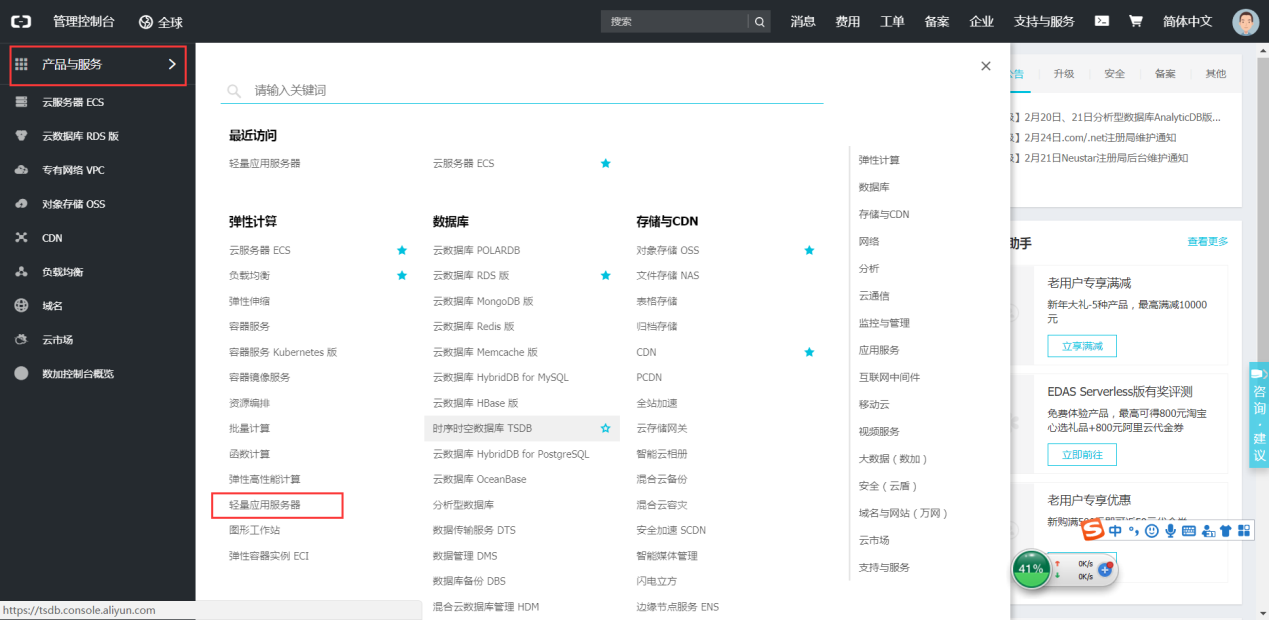
购买流程我简单讲一下，服务器的地址一般有华南区，华东区和华北区。实际上购买哪个地址的服务器都是可以的，你选择了哪个地区的服务器，该地区的用户人群访问你的服务器就会比其他地区的人快。我这里选择的是华东区的服务器。然后系统镜像我这里暂时选择的是windows版本的，这里后面在服务器设置是可以进行系统更改的，不用太在意。



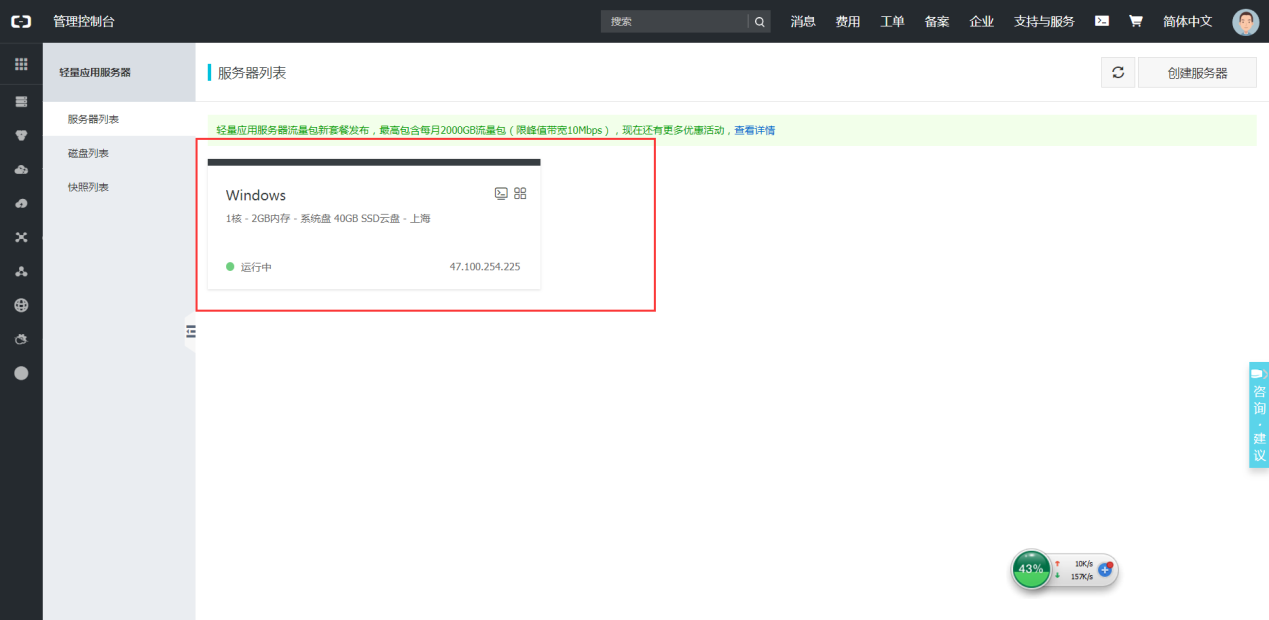
**二.将远程主机配置成服务器**

进入管理控制台

产品与服务->轻量应用服务器

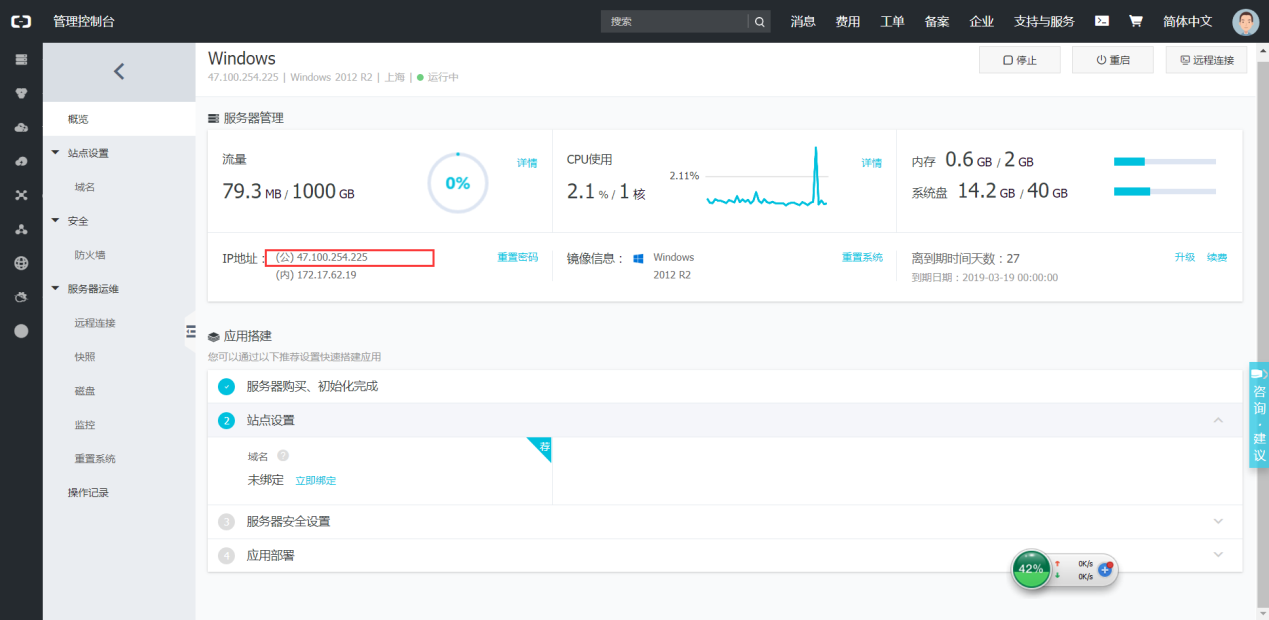


之后就可以看到我们的服务器实例，这里只有一个，是我刚刚购买的服务器

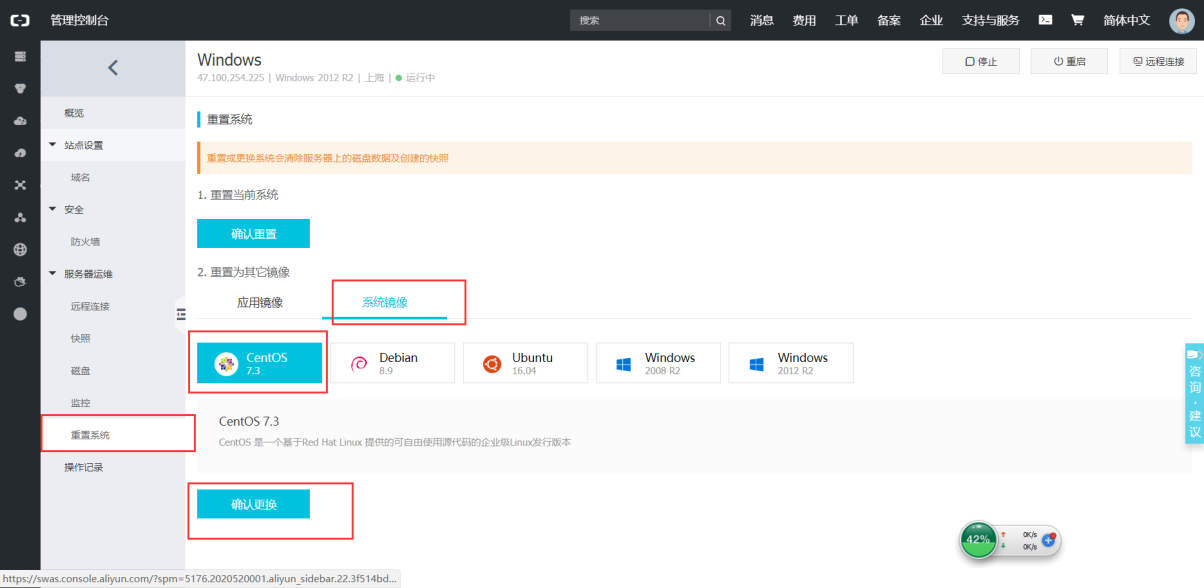


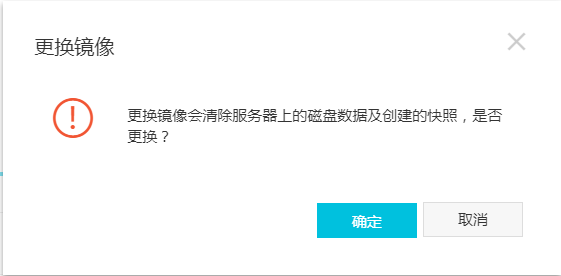
点击进入，可以看到我们当前服务器的各项参数

画红圈的部分是我们的服务器的公网ip地址，简单来说，就是别的用户要访问我的服务器，就必须通过这个ip地址来访问。这里记住它一下。

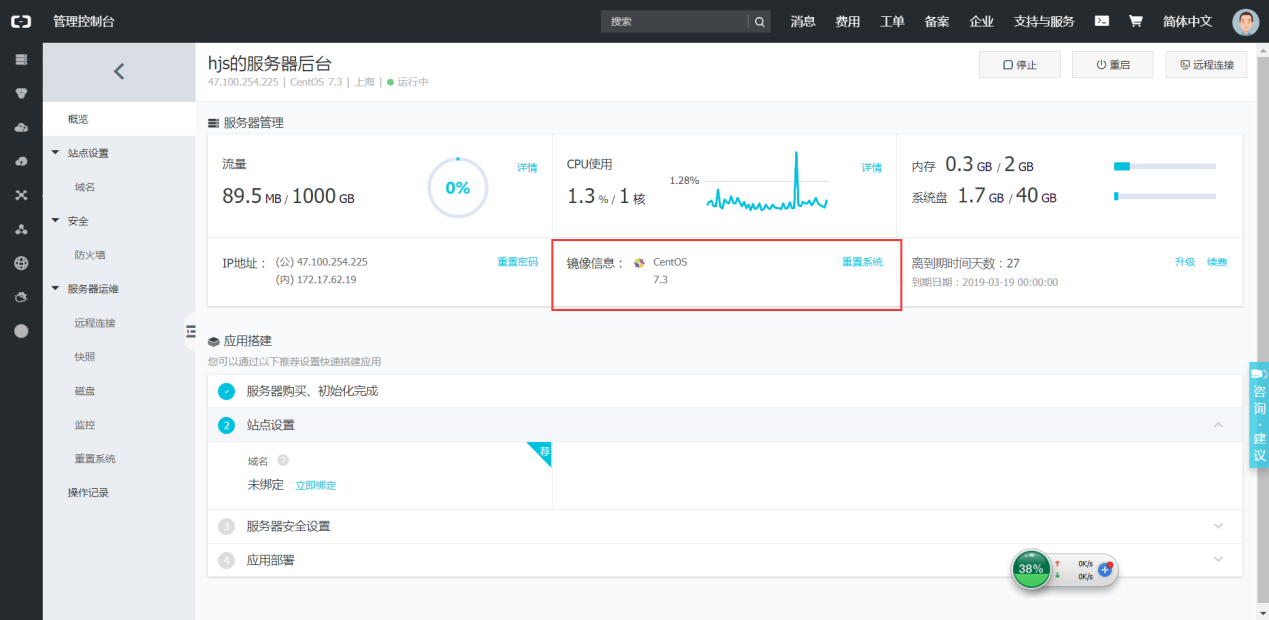


为了方便，我准备重置下系统镜像，换成CentOS。但如果是想重置系统，也可以在如下的设置里进行重置，但注意如果原系统上已经有重要文件，重置系统和重置镜像，都会清空丢失！





重置后效果如下：



为什么要选择linux的系统镜像，而不是用windows的系统呢？其实普通用户来说，使用windows系统的话，操作和配置上会更加简单和直观，而linux多以命令行来执行操作，相对来说在配置和后期的操作上会显得更为麻烦一些。但是linux系统服务器的安全性会比windows更高，因为windows会更容易受到黑客的攻击而导致服务器崩溃。总而言之，如果想从事服务器端后台工作的开发者来说，还是选择linux系统更为合适。

接下来就是一系列的配置流程了：

接下来安装Xshell和Xftp

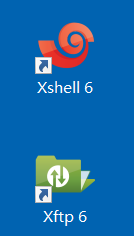
Xshell：控制远端云服务器主机的软件

即Xshell能在我们的windows界面下访问远端不同搭载在不同操作系统上的服务器。简单地说就是我们可以通过这款软件来控制我们的云服务器ECS或轻量型应用云服务器。

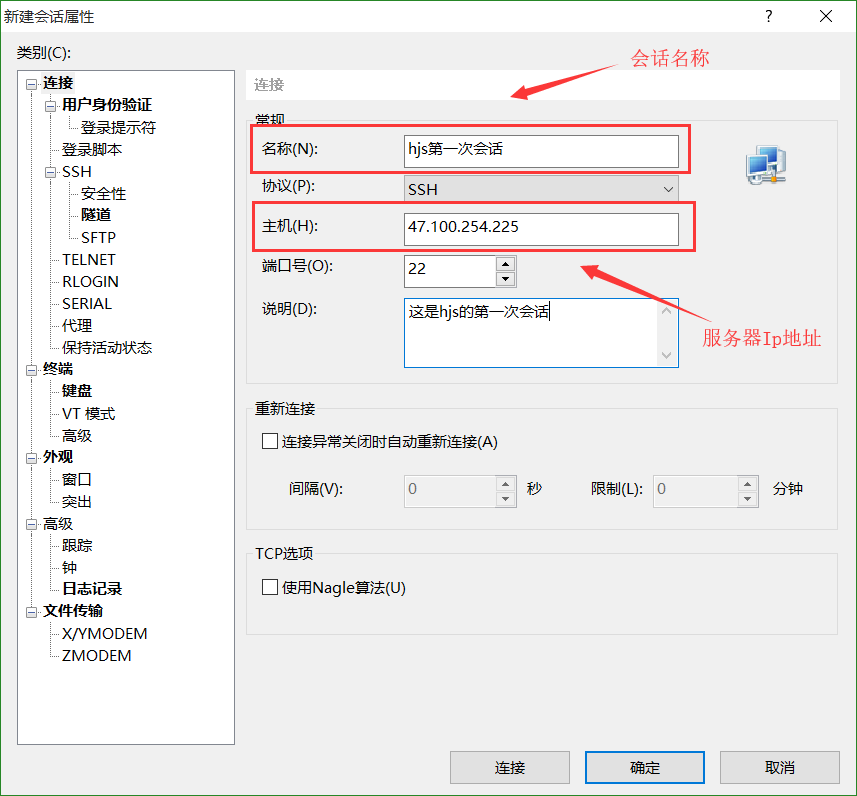
Xftp：能在当前电脑和远端云服务器之间传输文件的软件

一款功能强大的基于windows平台的SFTP、FTP文件传输软件。简单地说通过这款软件我们能在当前主机和远端云服务器主机之间传输文件。

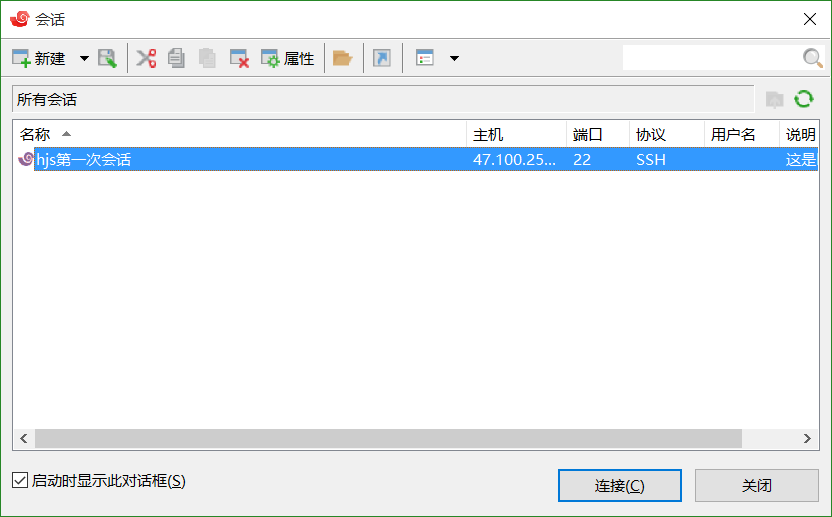
这两款软件如果用作学校或家庭用途，为免费。如果用作商业用途，就要收费。具体差别我没有仔细去研究。可以直接从官网下载，也可以从我的网盘链接里下载。



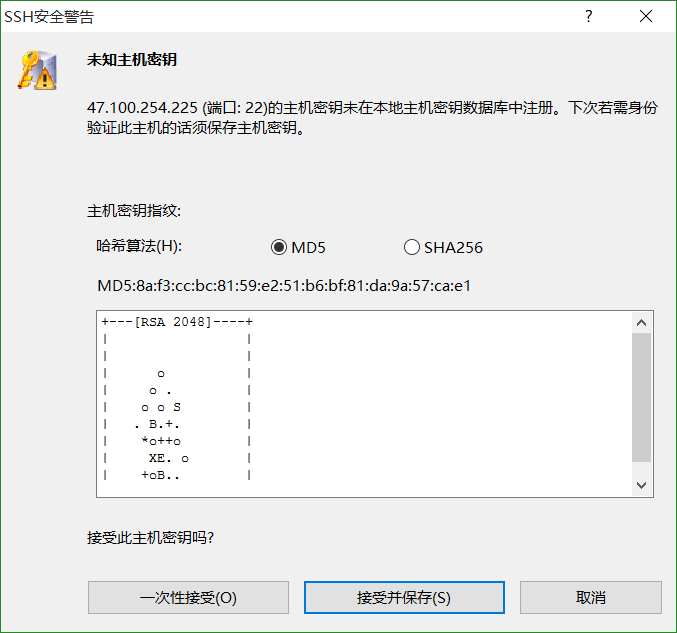
（1）运行Xshell，新建会话



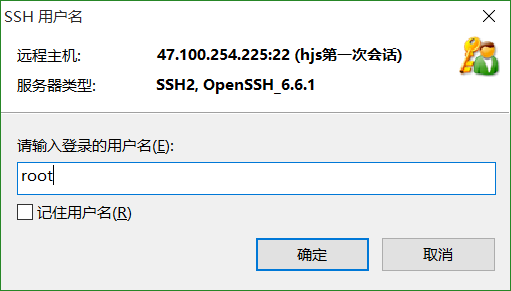
点击确定建立会话，选中建立的会话，点击连接



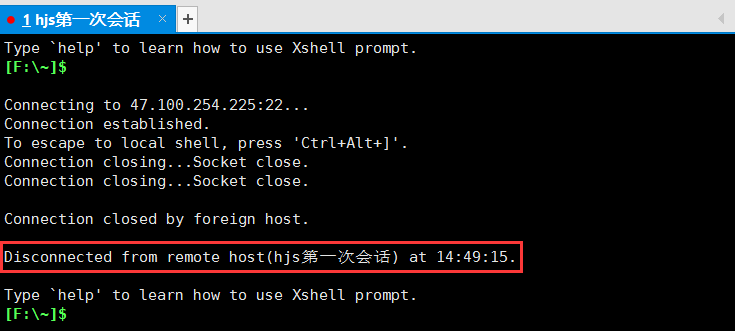
弹出如下对话框，点击接受并保存：



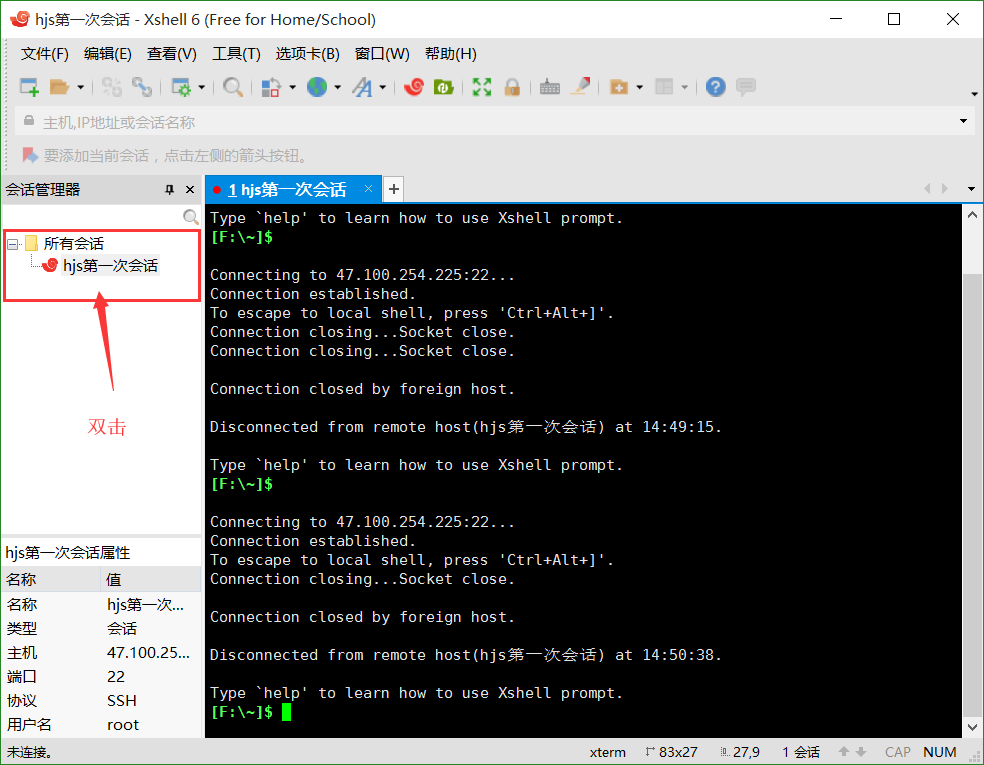
弹出如下对话框，输入登录用户名root，记住用户名，再点击确定



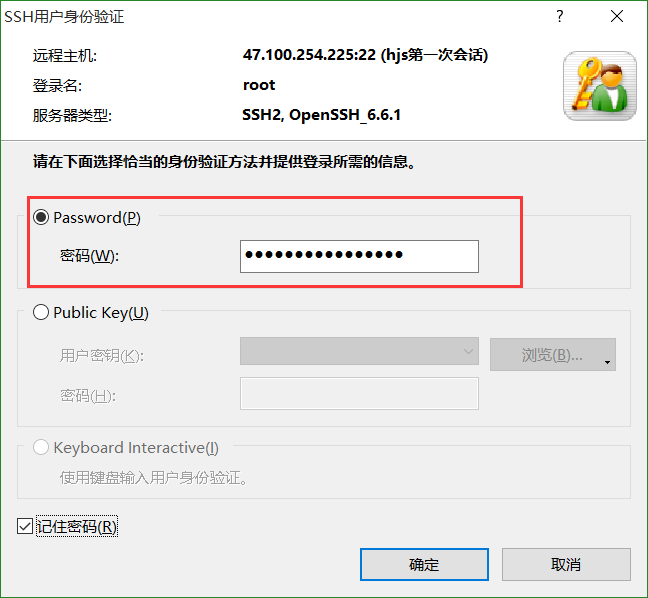
此时命令框显示连接断开？？



这是因为我们还没有输入密码。。。

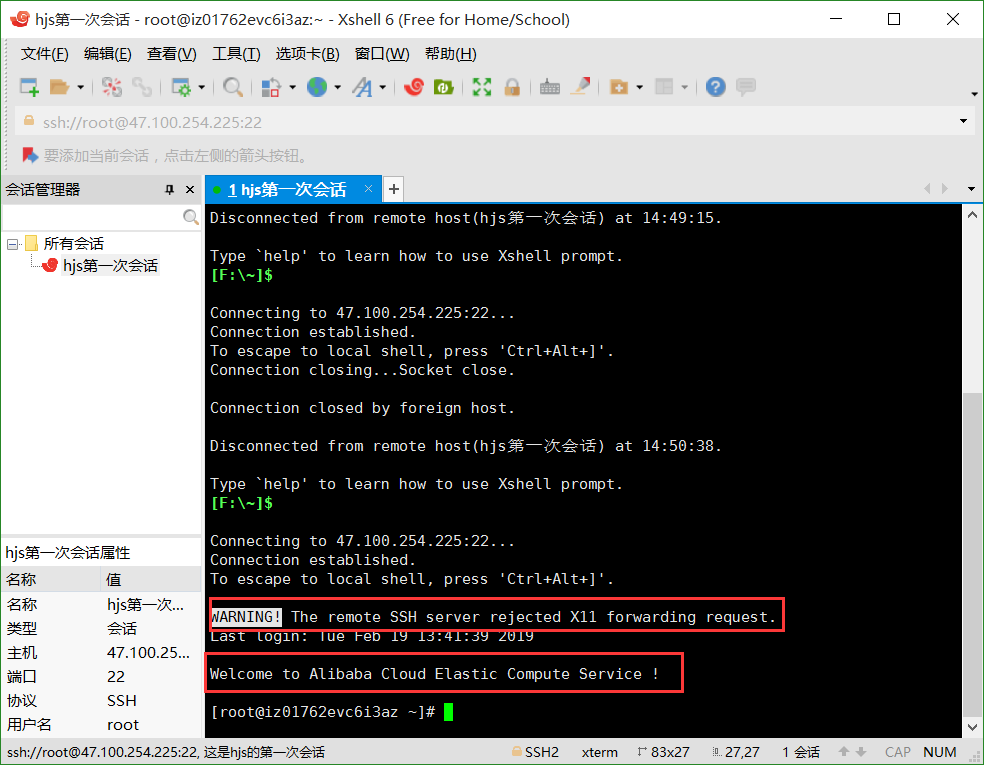


输入密码，再点击确定



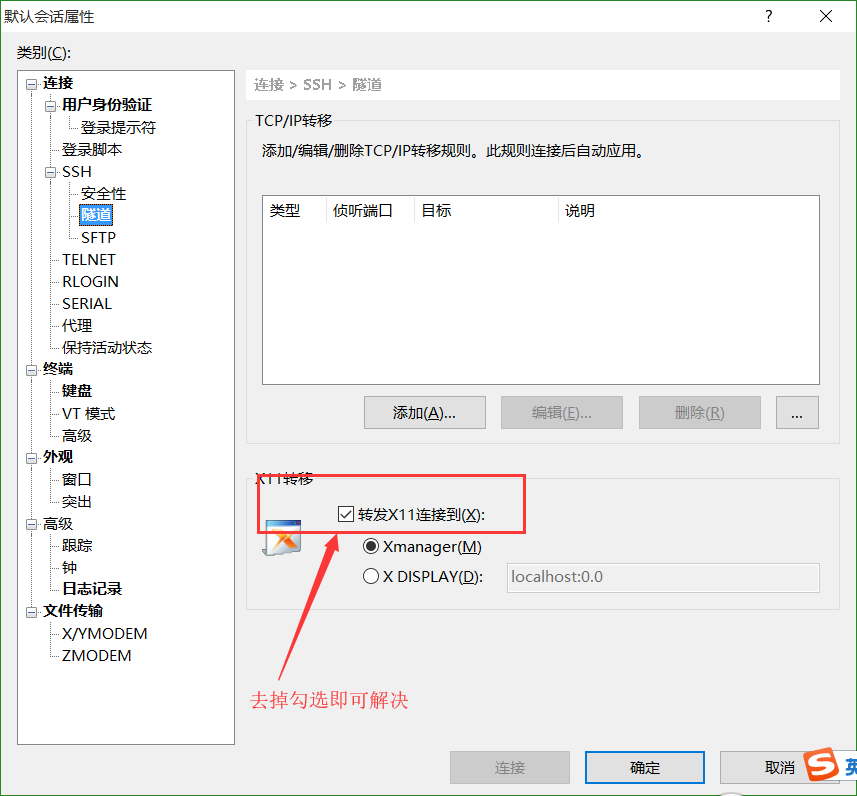
此时对话框弹出连接建立成功信息。此时我们当前主机与远端的阿里云服务器建立成功连接。

但是也弹出了警告：WARNING! The remote SSH server rejected X11 forwarding request.

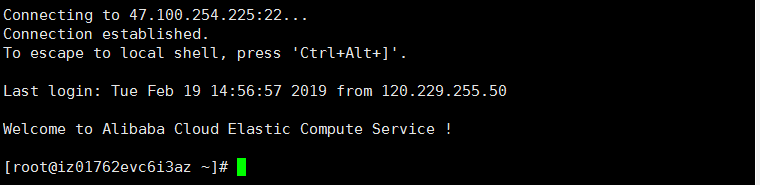


解决方法：

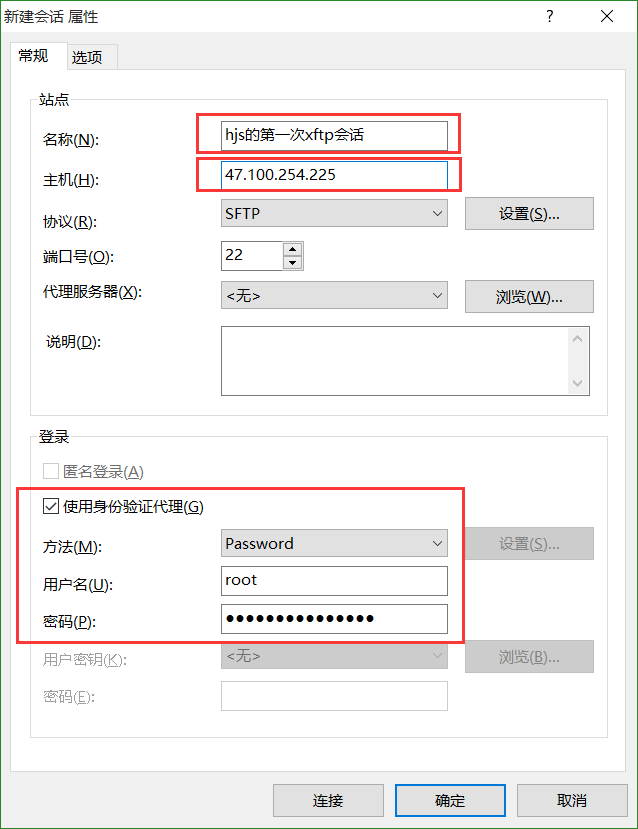
左上角：文件-属性-隧道



再双击会话重新建立连接，解决。



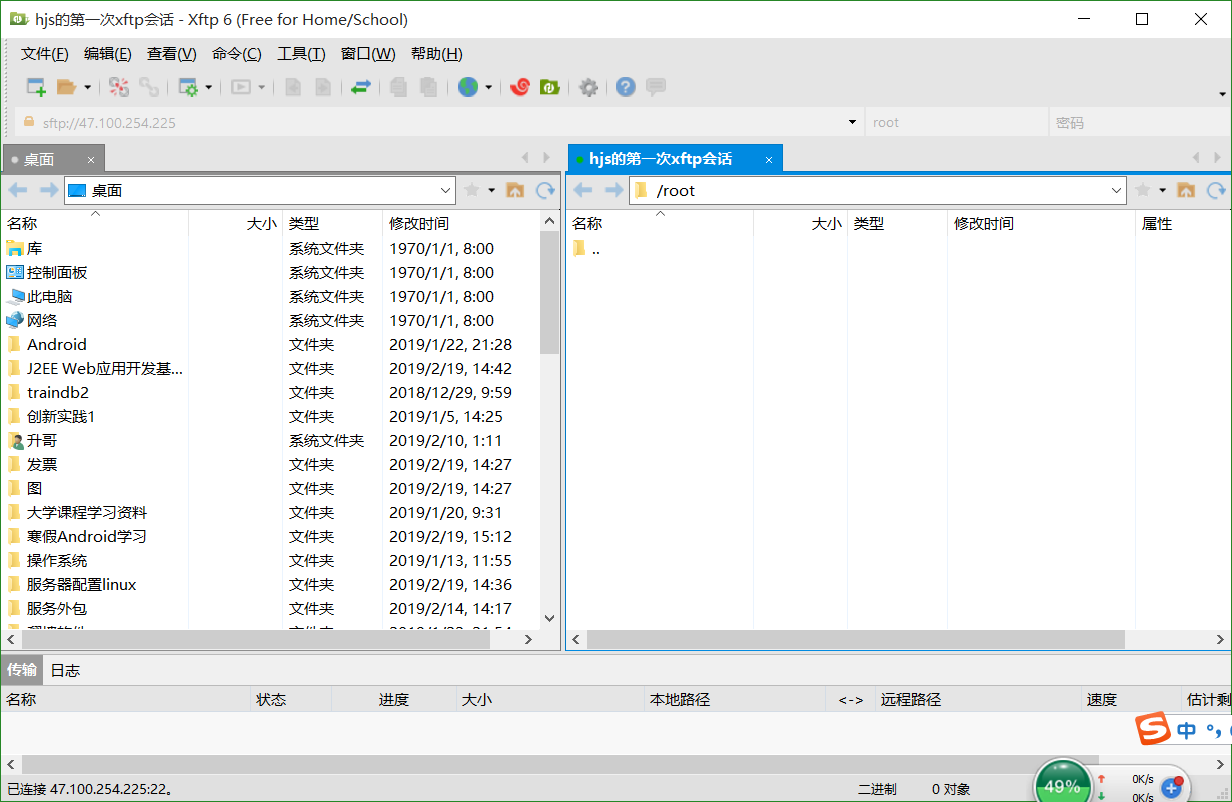
（2）运行Xftp，按照类似的方式建立会话。



点击确定，Xftp运行界面如下：

左边栏是我们当前主机的文件系统，这个显而易见

右边栏是我们远程服务器主机的linux文件系统，在右边点击返回键，你可以看到该系统下多个文件目录，我们后面会这里创建文件夹，来存放我们传输过去的jdk等安装文件。我们可以在这里手动之间创键，也可以在阿里云的服务器后台里通过建立连接后，以linux命令方式来创建。后面只介绍linux命令的创建的创建、安装和配置方式



还有一种方法建立Xftp会话：

直接在Xshell左上栏：窗口->传输新建文件 来免密打开并建立Xftp会话。

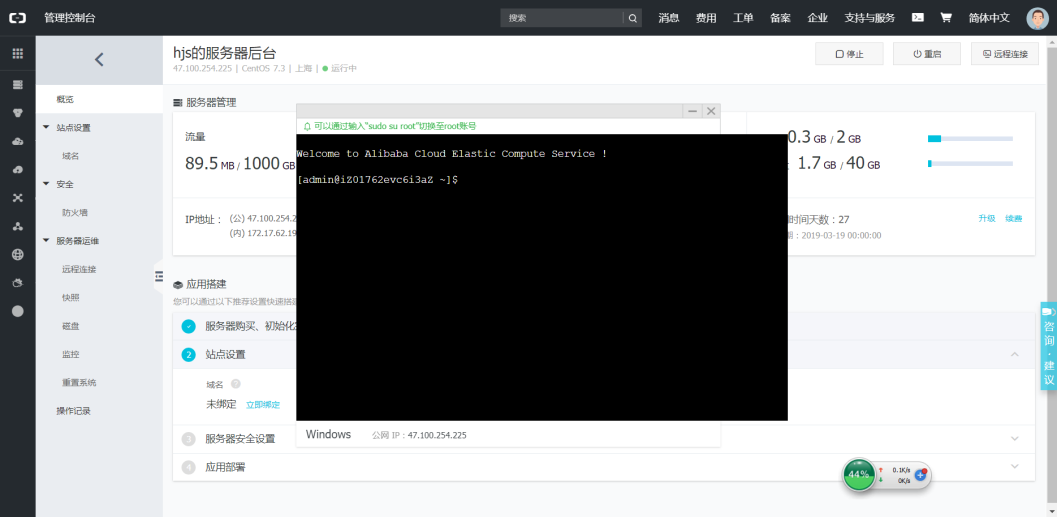
接下来就是部署我们java项目的准备工作了

在这里说一下，在云服务器上部署Java 项目，一共有三种方式，分别为Java镜像部署、一键安装包部署，以及手动部署。前两种方式的特点是非常方便和快捷，但是要需要花比较高的价钱来购买服务。手动部署就是我们自己来进行人工操作了，就是很费事。

接下来还是介绍下手动部署：

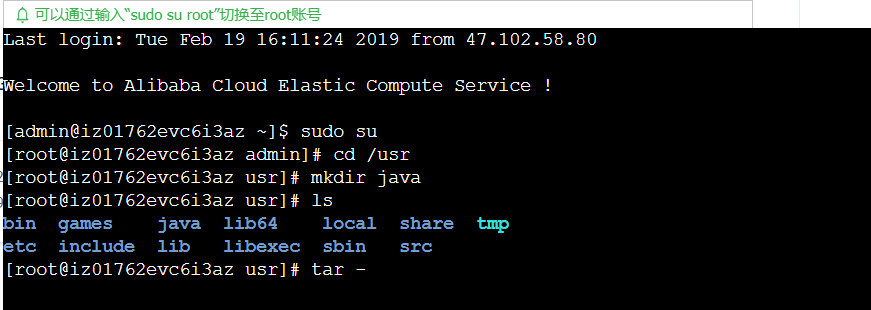
（1）主机下载linux版本的jdk和tomcat

接下来点击远程连接，进入linux命令行界面，开始我们的配置工作：



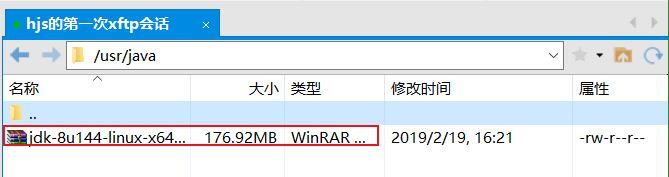
首先，用sudo su命令获得管理员权限，因为接下来会涉及到系统环境配置等管理员权限的操作。

我们先在linux系统下的usr中新建文件夹java来存放我们的jdk和tomcat

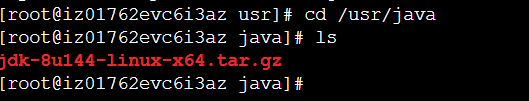


创建成功。

在xttp中将下载好的jdk传到该目录中



回到阿里云后台命令行界面，检查到传输成功了。

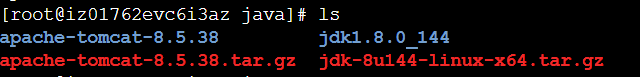


同理把apache压缩包也传输进这个文件夹里。

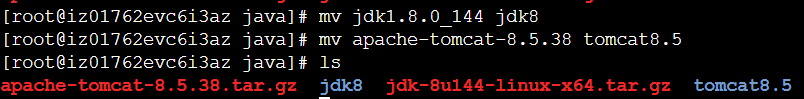


解压：



我们在这里修改一下文件名，方便后面配置时可以少写点命令，毕竟linux就是傻瓜式的用命令行来操作的。最后显示下当前文件项目，可看到更改后的显示如下所示。

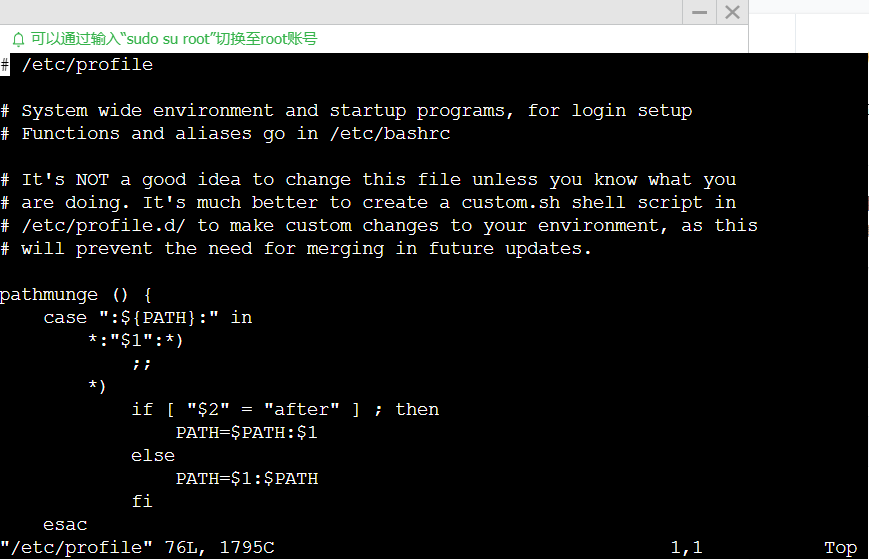


接下来配置环境变量

输入：



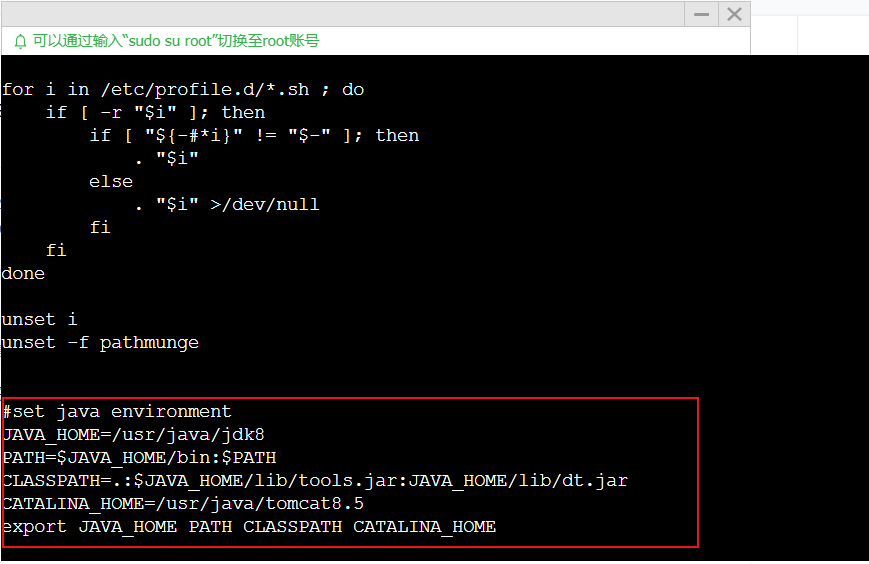
显示效果如下：



按i键进入编辑状态

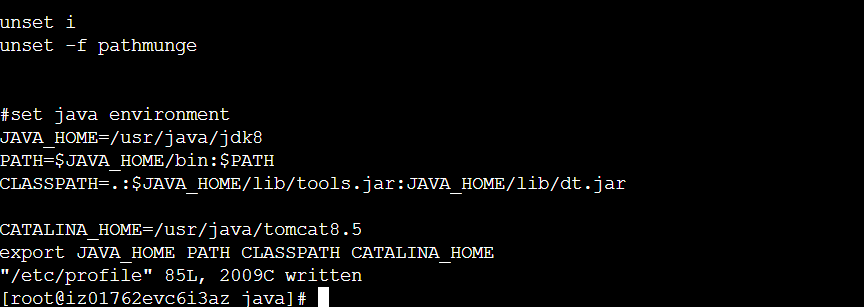


注意现在开始不要出现误删或误加的操作，这里是系统环境变量的配置，会影响系统其它位置的运行。一直跳到文本的最末尾，配置JDK和tomcat环境如下：



按esc键退出编辑状态，:wq保存并退出

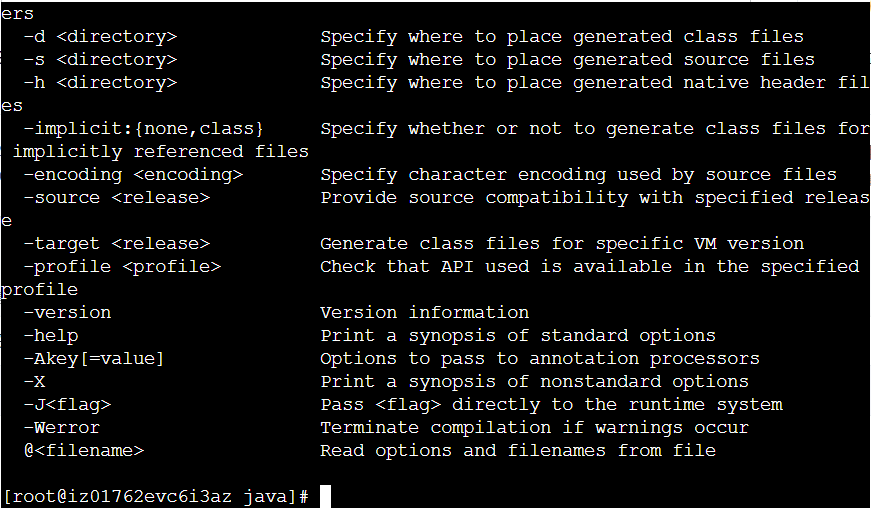
显示效果如下：



启用配置：



输入javac检验：

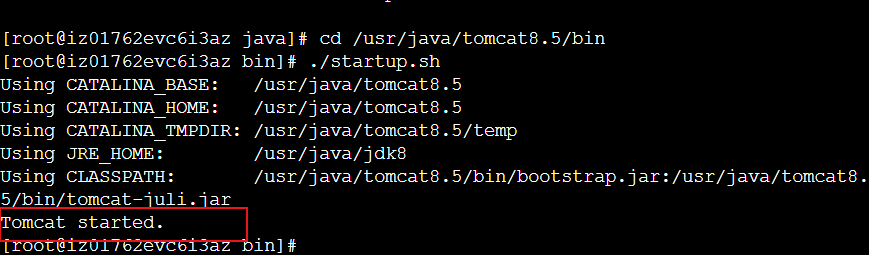


进入tomcat下的bin目录执行启动脚本，启动tomcat



注： ./shutdown.sh 为关闭脚本命令。可关闭tomcat

效果如下：



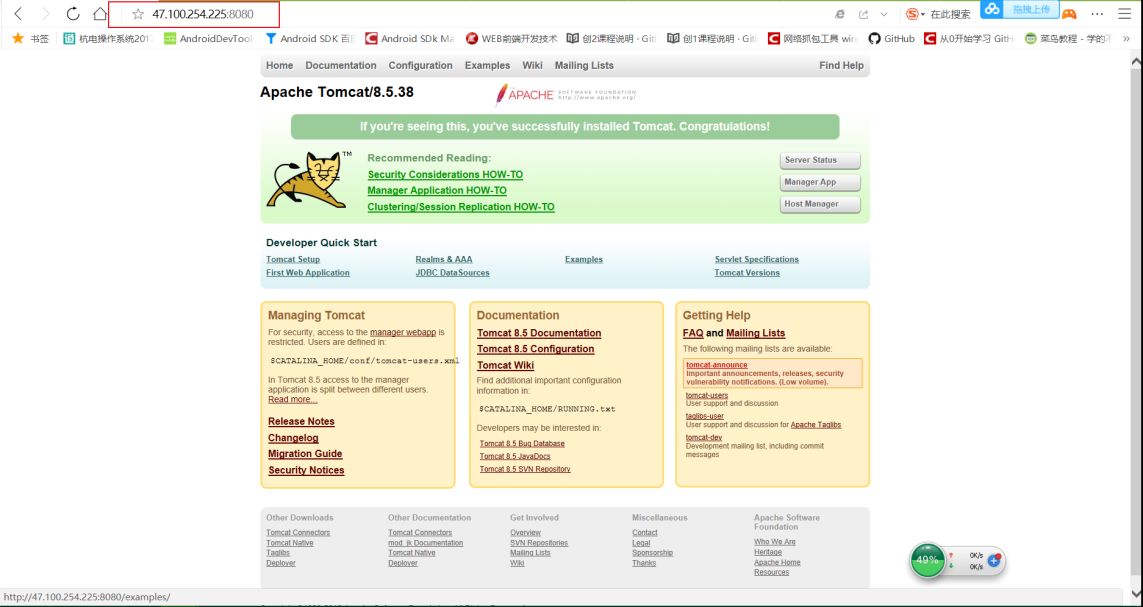
本来以为大功告成，访问http:// 47.100.254.225:8080，发现还是访问不了…

研究了好一会，发现还需要阿里云添加安全组规则

1. 进入控制台   
2. 选择 轻量应用服务器   
3. 在服务器列表中选择需要配置的服务器   
4. 在左侧 安全 中选择 防火墙   
5. 在右侧选择 添加规则   
6. 在 应用类型、协议、端口范围 三项中，分别填 自定义、TCP、8080



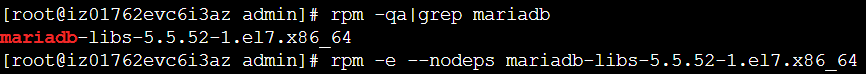
现在可以访问了。



接下来是对数据库的下载和配置

在centos7以上的版本系统中，默认搭载的数据库从MySQL改变为Mariadb。这个数据库没有用过，也不知道怎么样，这里我把它删了，换成我熟悉的MySQL。

卸载Mariadb



在安装MySQL之前我们最好再排除一下系统有没有可能已经存在了MySQL，如果存在了，是肯定安装不成功的，如果安装了，用 # rpm -e mysql 命令删除



如果直接安装配置自己传过去的下载好的MySQL tar包，可能会出现比较多的问题，需要另外自己下载依赖以及修改配置等。这里为了方便直接采用yum安装





开机自动启动：



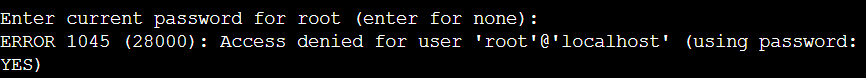
启动MySQL服务进程



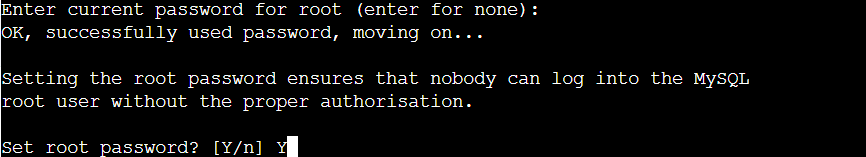
配置root用户密码



针对以下情况连续三次回车，出现ERROR提醒不要管，因为我们是第一次安装，没有设置过密码，因此默认为blank。

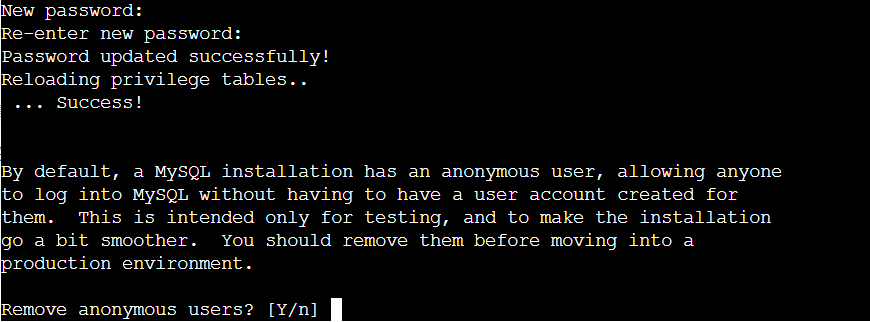


第三次回车后，出现以下信息：



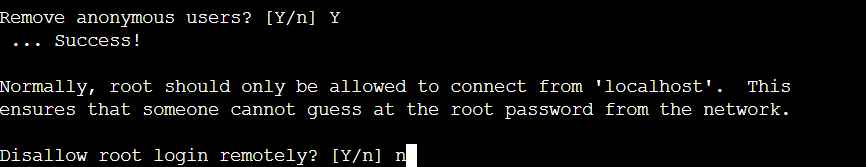
选Y，表示我们给root用户设置密码

要注意，设置密码时它是设置隐藏没让你看到的…不要因为键盘失灵或卡死了。直接输入就是了。

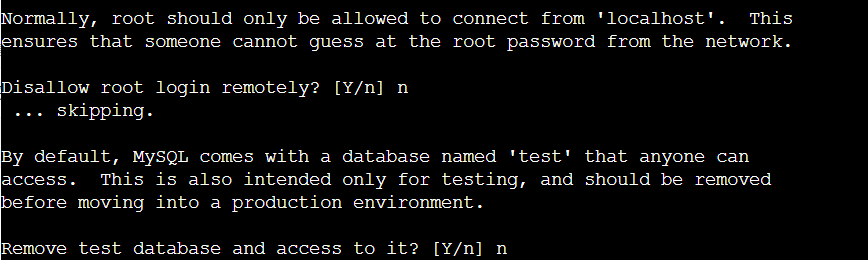


接下来选择是否删除临时用户，选择是。

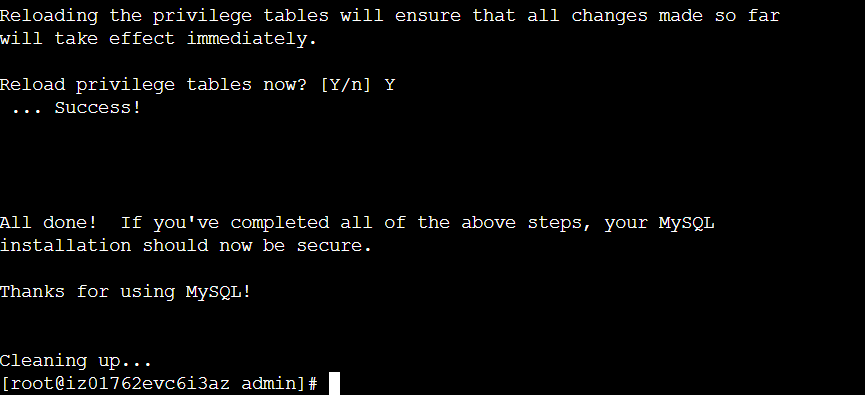
然后选择是否不允许远程登录，选择否



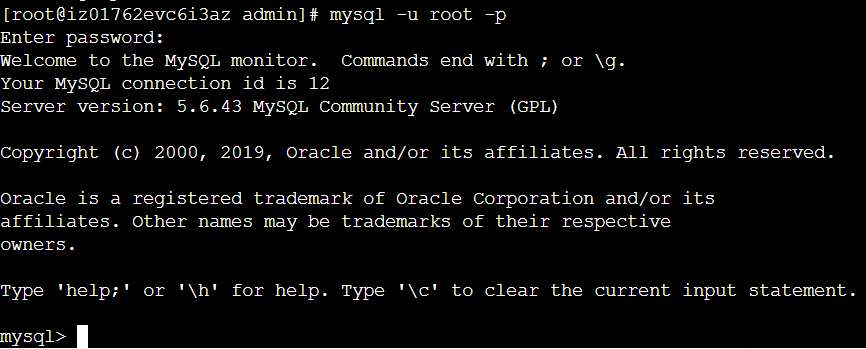
最后它问你是否要删除本来存在在MySQL的一个test数据库，该数据库是用于测试的，删不删取决于你。



最后同意刷新权限表，就此，数据库和配置结束了。



我们想要进入数据库执行SQL语句，可以输入一下命令，开启一个新的shell窗口，链接数据库。然后就可以进行CURD的命令了。



一切准备就绪，最后就差把我的javaweb项目部署到服务器上了。

首先，在Intellj Idea中把我们的项目打包成war项目：

File—>

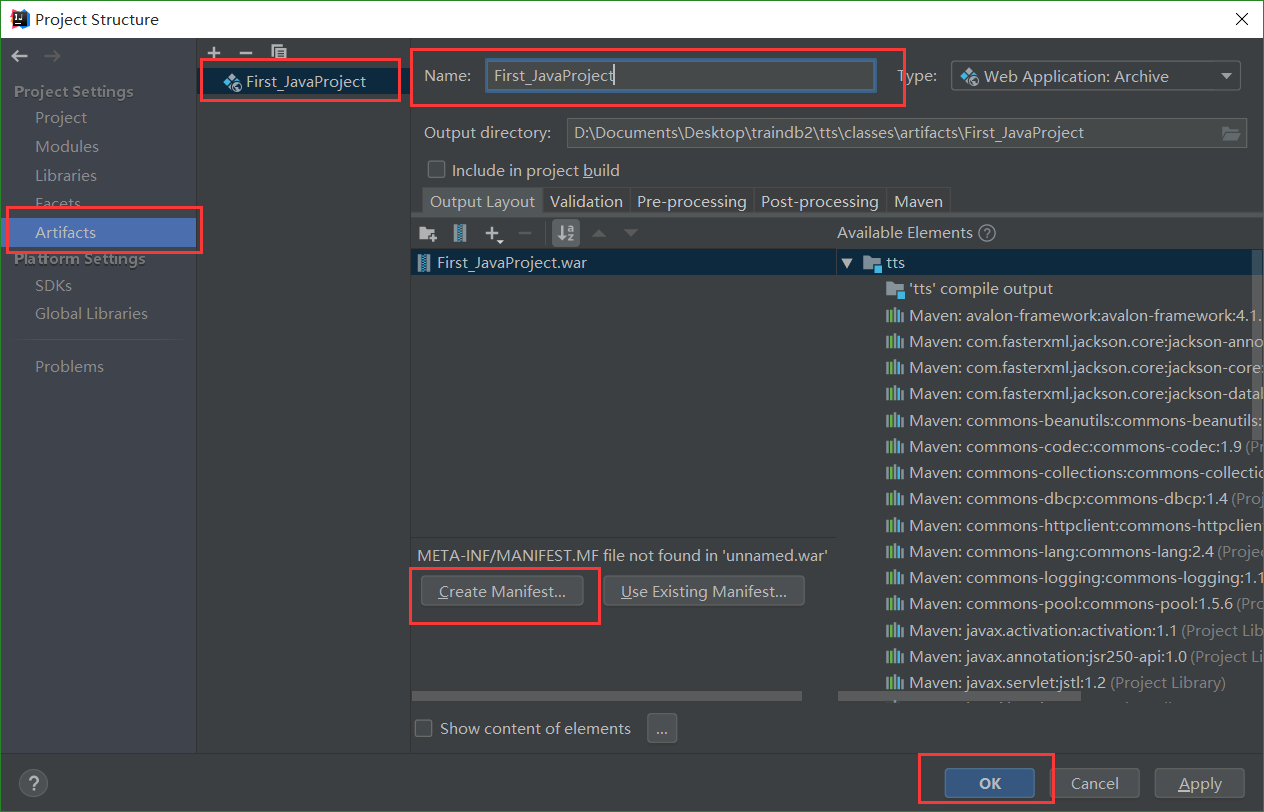
Project Structure—>

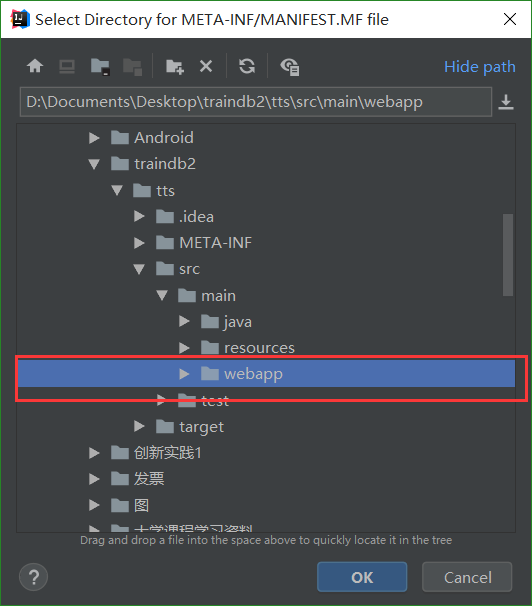
Artifacts—>

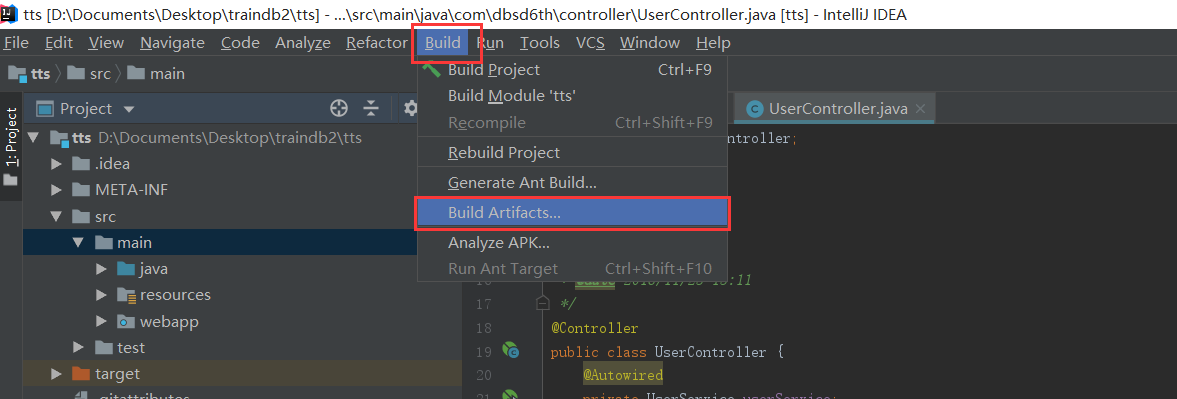
点击+号—>

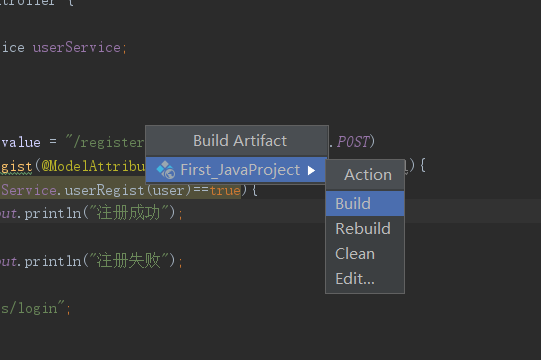
Web Application:Archive

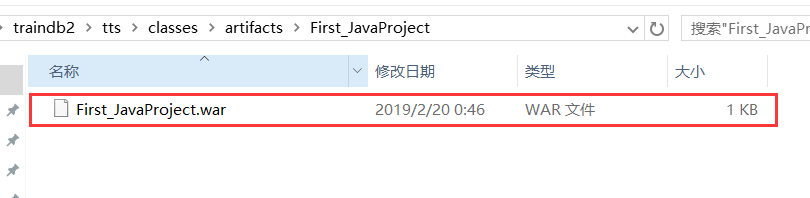
划红线部分为必须填写的部分











将生成war文件发送服务器主机的 Tomcat 目录下的 webapps文件夹下。

重启 Tomcat，就可以访问我们的项目。当然还有我们项目对应的sql文件也要导入到服务器端的mysql来执行。这里就先略了。