

---

## 持续集成篇

### Maven 私有库和本地库的安装与配置

#### Sonatype Nexus + Maven

环境: CentOS 6.6 Final、JDK7、Sonatype Nexus、Maven

IP: 192.168.4.221

root 用户操作

前提: 已安装 JDK7 并配置好了环境变量

1、下载最新版 Nexus (本教程使用的是: nexus-2.11.2-03-bundle.tar.gz), 下载地址:

<http://www.sonatype.org/nexus/go/>

```
# wget https://sonatype-download.global.ssl.fastly.net/nexus/oss/nexus-2.11.2-03-bundle.tar.gz
```

2、解压

```
# mkdir nexus
# tar -zxvf nexus-2.11.2-03-bundle.tar.gz -C nexus
# cd nexus
# ls
```

nexus-2.11.2-03 sonatype-work

(一个 nexus 服务, 一个私有库目录)

3、编辑 Nexus 的 nexus.properties 文件, 配置端口和 work 目录信息 (保留默认)

```
# cd nexus-2.11.2-03
# ls
bin conf lib LICENSE.txt logs nexus NOTICE.txt tmp
```

查看目录结构, jetty 运行

```
# cd conf
# vi nexus.properties
# Jetty section
application-port=8081
application-host=0.0.0.0
nexus-webapp=${bundleBasedir}/nexus
nexus-webapp-context-path=/nexus

# Nexus section
nexus-work=${bundleBasedir}/../sonatype-work/nexus
runtime=${bundleBasedir}/nexus/WEB-INF
```

4、编辑 nexus 脚本, 配置 RUN\_AS\_USER 参数

```
# vi /root/nexus/nexus-2.11.2-03/bin/nexus
```



龙果学院微信公众号: ron-coo

```
#RUN_AS_USER=
```

改为:

```
RUN_AS_USER=root
```

5、防火墙中打开 8081 端口

```
# vi /etc/sysconfig/iptables
```

添加:

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 8081 -j ACCEPT
```

保存后重启防火墙

```
# service iptables restart
```

6、启动 nexus

```
# /root/nexus/nexus-2.11.2-03/bin/nexus start
```

```
*****
```

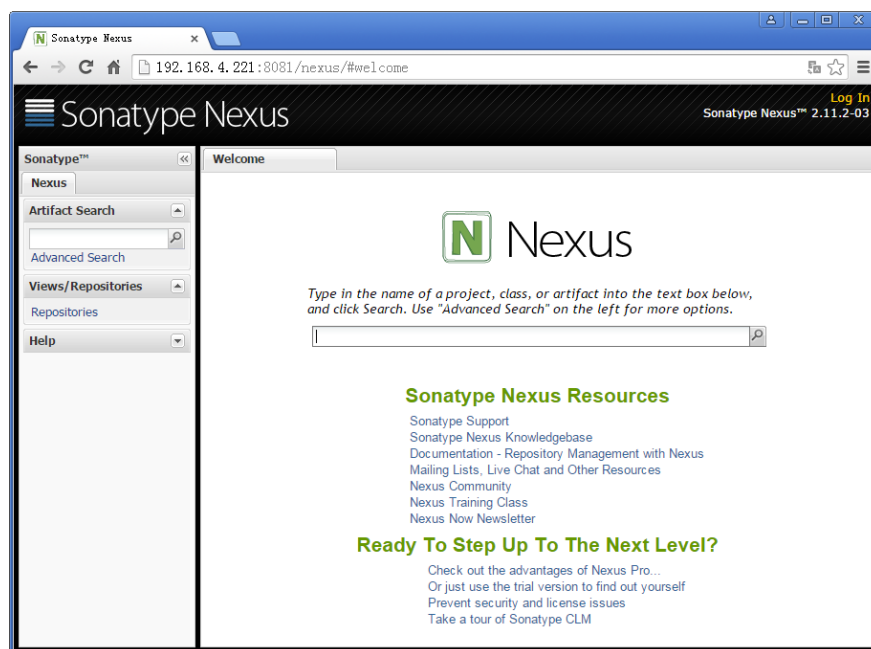
```
WARNING - NOT RECOMMENDED TO RUN AS ROOT
```

```
*****
```

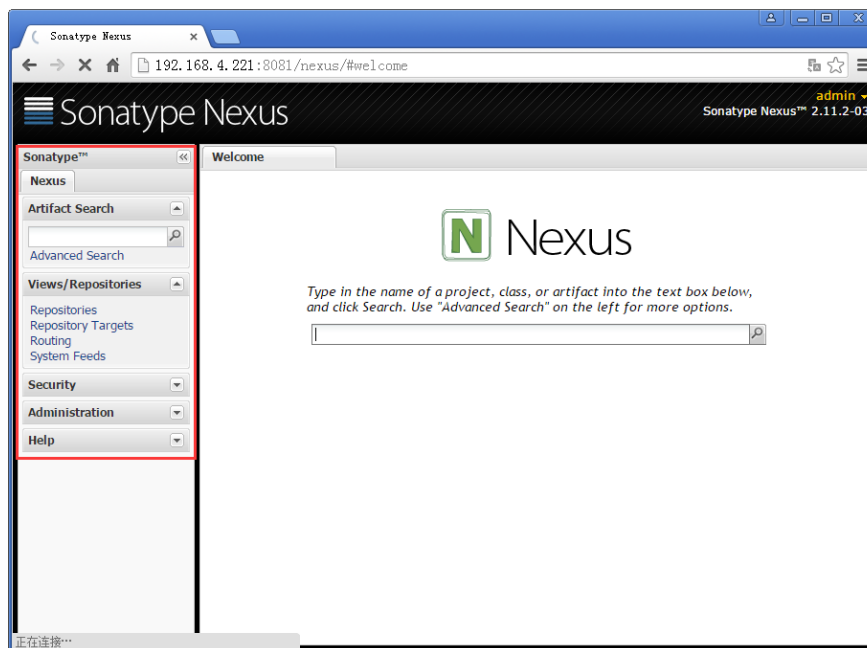
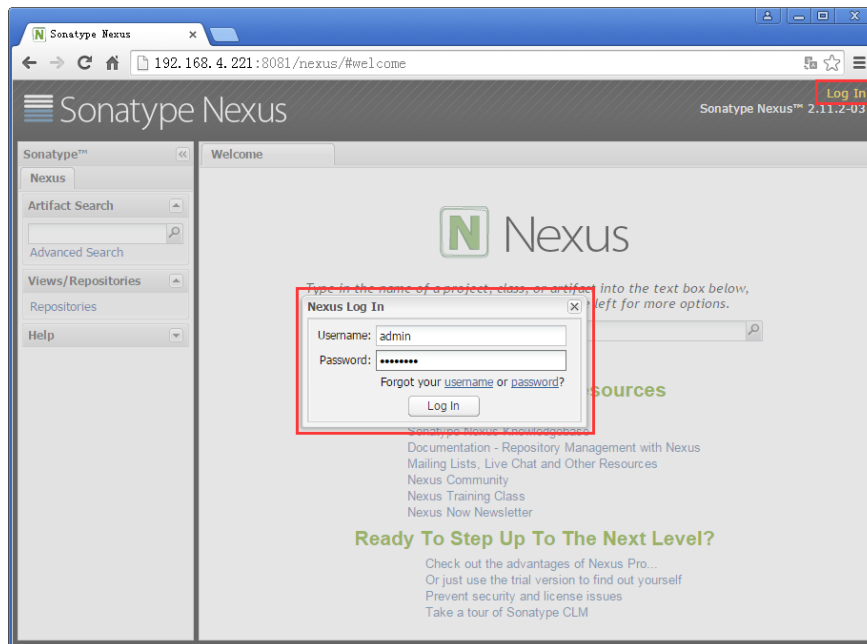
```
Starting Nexus OSS...
```

```
Started Nexus OSS.
```

7、浏览器中打开: <http://192.168.4.221:8081/nexus/>



8、登录，默认用户名 **admin**, 默认密码 **admin123**:

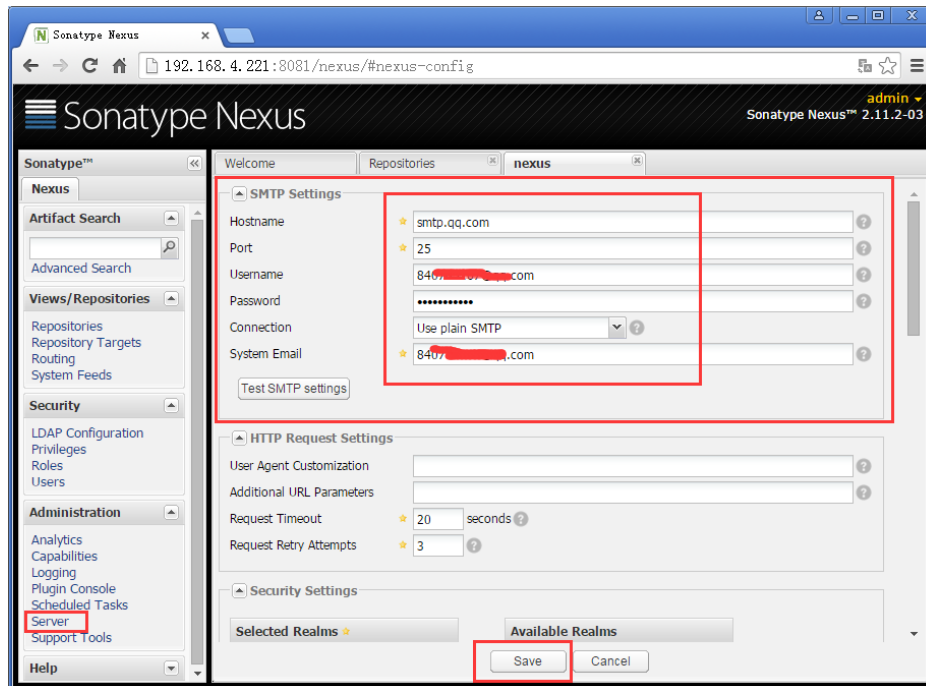


到此，Nexus 已安装完成，接下来是 Nexus 的配置

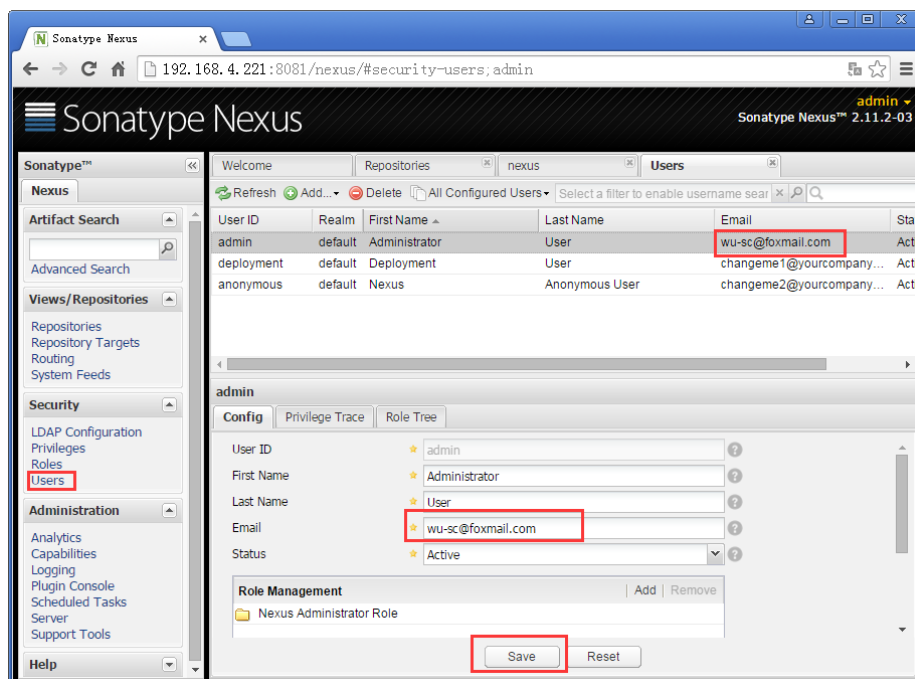


## Nexus 配置 (登录后)

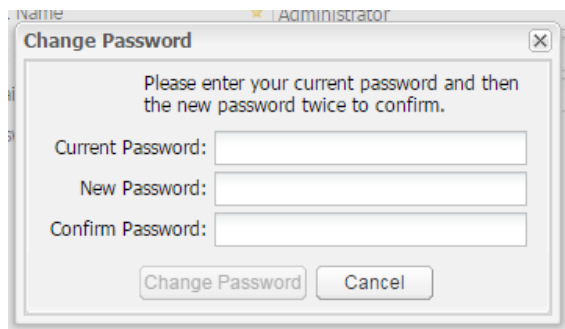
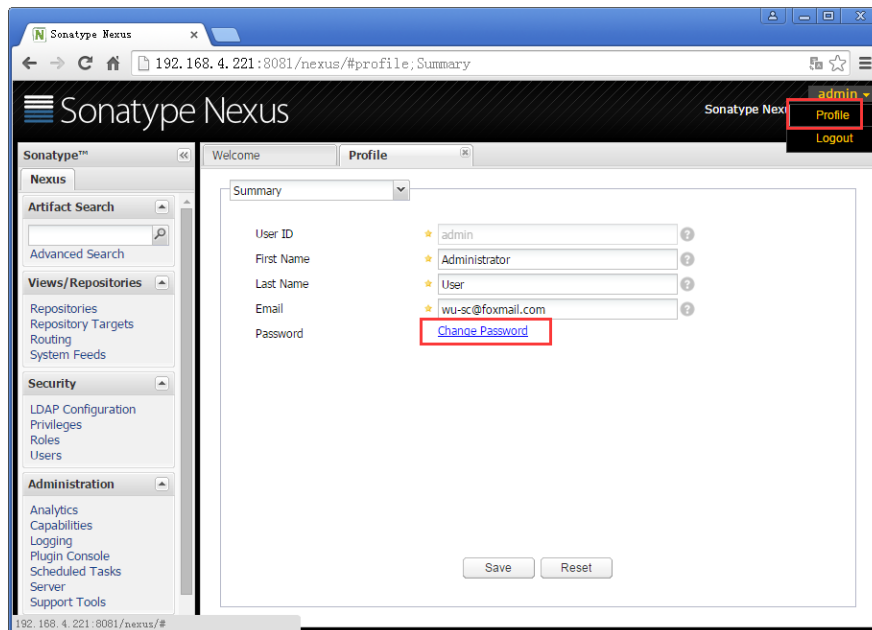
1、菜单 Administration/Server 配置邮箱服务地址 (如果忘记密码, 可以通过该邮箱找回密码)



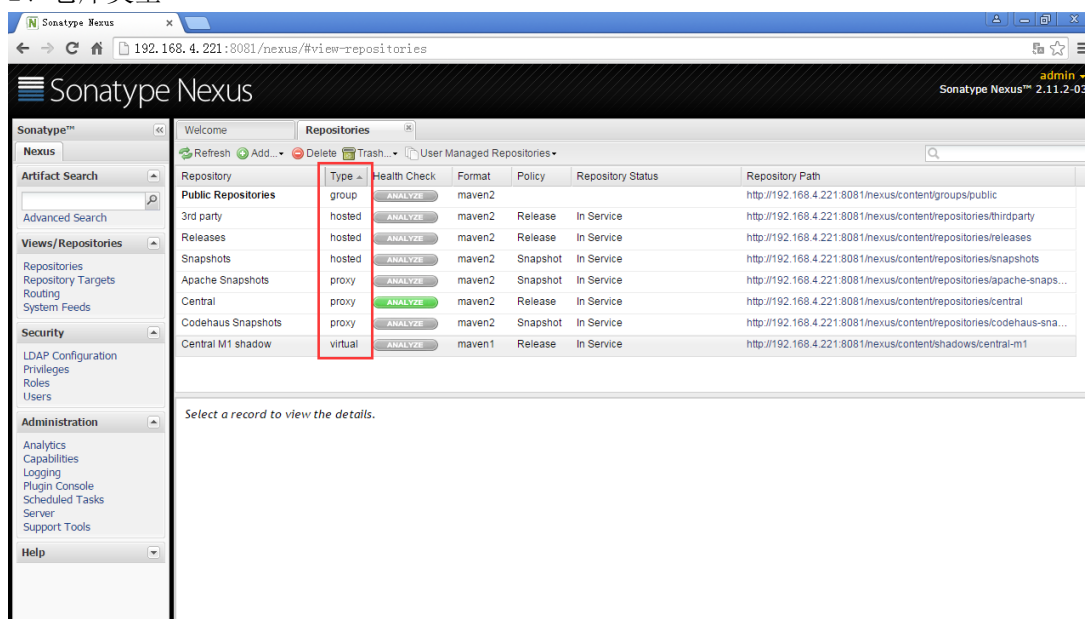
给用户配置邮箱地址, 方便忘记密码时找回:



## 用户修改密码



## 2、仓库类型



**group 仓库组:** Nexus 通过仓库组的概念统一管理多个仓库, 这样我们在项目中直接请



龙果学院微信公众号: ron-coo

求仓库组即可请求到仓库组管理的多个仓库;

**hosted 宿主仓库:** 主要用于发布内部项目构件或第三方的项目构件(如购买商业的构件)以及无法从公共仓库获取的构件(如 oracle 的 JDBC 驱动)

**proxy 代理仓库:** 代理公共的远程仓库;

**virtual 虚拟仓库:** 用于适配 Maven 1;

一般用到的仓库种类是 hosted、proxy

### Hosted 仓库常用类型说明:

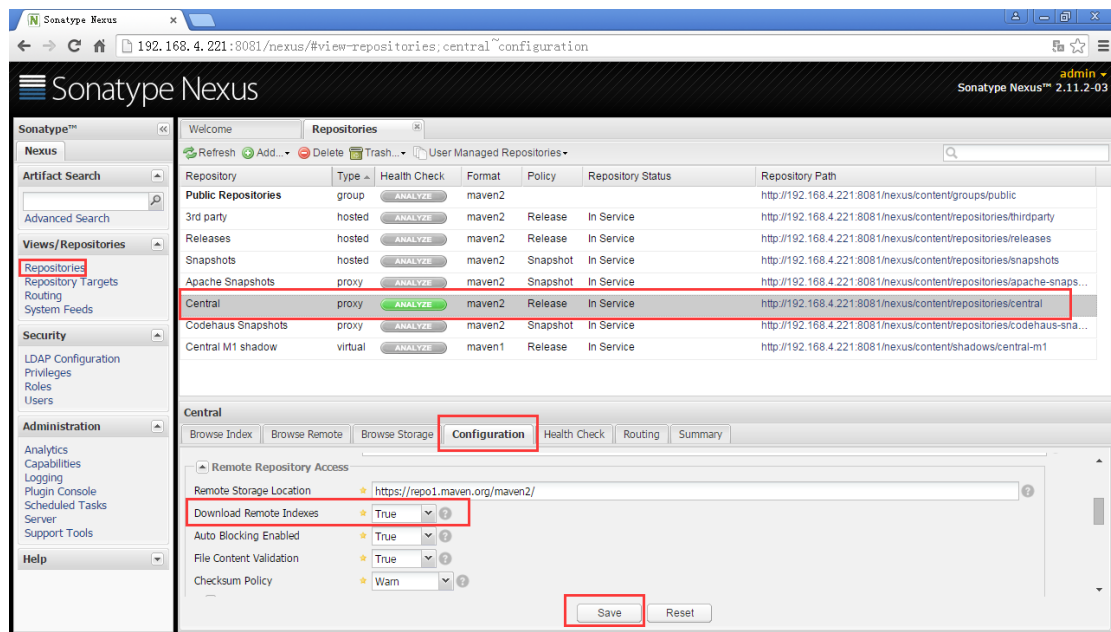
**releases** 内部的模块中 release 模块的发布仓库

**snapshots** 发布内部的 SNAPSHOT 模块的仓库

**3rd party** 第三方依赖的仓库, 这个数据通常是由内部人员自行下载之后发布上去

如果构建的 Maven 项目本地仓库没有对应的依赖包, 那么就会去 Nexus 私服去下载, 如果 Nexus 私服也没有此依赖包, 就回去远程中央仓库下载依赖, 这些中央仓库就是 proxy。Nexus 私服下载成功后再下载至本地 Maven 库供项目引用。

3、设置 proxy 代理仓库 (Apache Snapshots/Central/Codehaus Snapshots) 准许远程下载, 如:



4、Maven 本地库的安装与配置 (请看参考视频的操作)

环境变量、setting.xml

5、MyEclipse 中的 Maven 配置 (请看参考视频的操作)

6、项目的构建与发布演示 (请看参考视频的操作)

修改 common-parent 中的 pom.xml 中的私有库地址

7、上传第三方包操作演示



龙果学院微信公众号: ron-coo