nesux

nexus介绍

私服是指私有服务器,是假设在局域网的一种特殊的远程仓库,目的是代理远程仓库及部署第三方构建.有了私服之后,当maven需要下载构件时,直接请求私服,私服上存在则下载到本地仓库;否则,私服请求外部的远程仓库,将构件下载到私服,在提供给本地仓库下载.

nexus是一个强大的maven仓库管理器,它极大的简化了本地内部仓库的维护和外部仓库的访问.
nexus是一套开箱即用的系统不需要数据库,它使用文件系统加Lucene来组织数据
nexus使用ExtJS来开发界面,利用Restlet来提供完整的REST APIs,通过IDEA和Eclipse集成使用
nexus支持webDAV与LDAP安全身份认证.
nexus提供了强大的仓库管理功能,构件搜索功能,它基于REST,友好的UI是一个extjs的REST客户端,占用较少的内存,基于简单文件系统而非数据库.
好处

1）加速构建;
2）节省带宽;
3）节省中央maven仓库的带宽;
4）稳定（应付一旦中央服务器出问题的情况）;
5）控制和审计;
6）能够部署第三方构件;
7）可以建立本地内部仓库;
8）可以建立公共仓库
这些优点使得Nexus日趋成为最流行的Maven仓库管理器

CentOS7.x上用nexus搭建yum仓库

一、实验背景

在生产环境中，我们不可能所有的服务器都能连接外网更新rpm包，比较理想的环境是：有一台Linux服务器可以连接外网，剩余的服务器通过这台yum服务器更新。

传统的做法是先把包下载到内网中的yum服务器上，然后通过createrepo命令生成本地仓库，其余服务器通过HTTP/FTP访问这个链接，这种做法比较费时费事。有没有一种比较好的方式，让我们直接通过这台服务器代理连接到公网的163、阿里云yum仓库呢?

这就是本次介绍的Nexus代理，无论你的客户机是CentOS6还是CentOS7又或者是Ubuntu，不论你是想用yum还是pip又或者是npm包管理器，Nexus都能满足你的需求。

你只需要将nexus放到能连外网的服务器上，通过nexus暴露服务就可以了。

二、实验环境

操作系统：CentOS Linux release 7.6.1810 (Core)

nexusServer： 192.168.100.188

yumClient： 192.168.100.241

三、安装nexus

1、基于docker方式部署

部署nexus服务器

```
setenforce 0

sed -i 's/SELINUX=.*/SELINUX=permissive/g' /etc/selinux/config
```

安装docker服务

```
[root@nexusServer ~]# yum -y install  yum-utils device-mapper-
persistent-data lvm2

[root@nexusServer ~]# ll /etc/yum.repos.d/CentOS-*
-rw-r--r--. 1 root root 1664 11月 23 2018 /etc/yum.repos.d/CentOS-
Base.repo
-rw-r--r--. 1 root root 1309 11月 23 2018 /etc/yum.repos.d/CentOS-
CR.repo
-rw-r--r--. 1 root root  649 11月 23 2018 /etc/yum.repos.d/CentOS-
Debuginfo.repo
-rw-r--r--. 1 root root  314 11月 23 2018 /etc/yum.repos.d/CentOS-
fasttrack.repo
-rw-r--r--. 1 root root  630 11月 23 2018 /etc/yum.repos.d/CentOS-
Media.repo
-rw-r--r--. 1 root root 1331 11月 23 2018 /etc/yum.repos.d/CentOS-
Sources.repo
-rw-r--r--. 1 root root 5701 11月 23 2018 /etc/yum.repos.d/CentOS-
Vault.repo
```

```
[root@nexusServer ~]# yum-config-manager  --add-repo
https://download.docker.com/linux/centos/docker-ce.repo

[root@nexusServer ~]# ll --time-style=long-iso
/etc/yum.repos.d/*.repo
-rw-r--r--. 1 root root 1664 2018-11-23 21:16
/etc/yum.repos.d/CentOS-Base.repo
-rw-r--r--. 1 root root 1309 2018-11-23 21:16
/etc/yum.repos.d/CentOS-CR.repo
-rw-r--r--. 1 root root  649 2018-11-23 21:16
/etc/yum.repos.d/CentOS-Debuginfo.repo
-rw-r--r--. 1 root root  314 2018-11-23 21:16
/etc/yum.repos.d/CentOS-fasttrack.repo
-rw-r--r--. 1 root root  630 2018-11-23 21:16
/etc/yum.repos.d/CentOS-Media.repo
-rw-r--r--. 1 root root 1331 2018-11-23 21:16
/etc/yum.repos.d/CentOS-Sources.repo
-rw-r--r--. 1 root root 5701 2018-11-23 21:16
/etc/yum.repos.d/CentOS-Vault.repo
-rw-r--r--. 1 root root 1919 2021-06-11 17:14
/etc/yum.repos.d/docker-ce.repo


#--showduplicates命令的使用方法,这个参数对于我们来说十分有用。尤其是当你遇
到版本不匹配兼容、软件依赖问题，就可以使用这个命令，找到你要 的软件版本**


[root@nexusServer ~]# yum list docker-ce  --showduplicates | sort
-r

已加载插件：fastestmirror
可安装的软件包

 * updates: mirrors.huaweicloud.com
   Loading mirror speeds from cached hostfile
 * extras: mirrors.huaweicloud.com
   docker-ce.x86_64            3:20.10.7-3.el7
docker-ce-stable
   docker-ce.x86_64            3:20.10.6-3.el7
docker-ce-stable
   docker-ce.x86_64            3:20.10.5-3.el7
docker-ce-stable
   docker-ce.x86_64            3:20.10.4-3.el7
docker-ce-stable
```

```
   docker-ce.x86_64            3:20.10.3-3.el7
docker-ce-stable
   docker-ce.x86_64            3:20.10.2-3.el7
docker-ce-stable
   docker-ce.x86_64            3:20.10.1-3.el7
docker-ce-stable
   docker-ce.x86_64            3:20.10.0-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.9-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.8-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.7-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.6-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.5-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.4-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.3-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.2-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.15-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.14-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.1-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.13-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.12-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.11-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.10-3.el7
docker-ce-stable
   docker-ce.x86_64            3:19.03.0-3.el7
docker-ce-stable
   docker-ce.x86_64            3:18.09.9-3.el7
docker-ce-stable
   docker-ce.x86_64            3:18.09.8-3.el7
docker-ce-stable
```

```
   docker-ce.x86_64            3:18.09.7-3.el7
docker-ce-stable
   docker-ce.x86_64            3:18.09.6-3.el7
docker-ce-stable
   docker-ce.x86_64            3:18.09.5-3.el7
docker-ce-stable
   docker-ce.x86_64            3:18.09.4-3.el7
docker-ce-stable
   docker-ce.x86_64            3:18.09.3-3.el7
docker-ce-stable
   docker-ce.x86_64            3:18.09.2-3.el7
docker-ce-stable
   docker-ce.x86_64            3:18.09.1-3.el7
docker-ce-stable
   docker-ce.x86_64            3:18.09.0-3.el7
docker-ce-stable
   docker-ce.x86_64            18.06.3.ce-3.el7
docker-ce-stable
   docker-ce.x86_64            18.06.2.ce-3.el7
docker-ce-stable
   docker-ce.x86_64            18.06.1.ce-3.el7
docker-ce-stable
   docker-ce.x86_64            18.06.0.ce-3.el7
docker-ce-stable
   docker-ce.x86_64            18.03.1.ce-1.el7.centos
docker-ce-stable
   docker-ce.x86_64            18.03.0.ce-1.el7.centos
docker-ce-stable
   docker-ce.x86_64            17.12.1.ce-1.el7.centos
docker-ce-stable
   docker-ce.x86_64            17.12.0.ce-1.el7.centos
docker-ce-stable
   docker-ce.x86_64            17.09.1.ce-1.el7.centos
docker-ce-stable
   docker-ce.x86_64            17.09.0.ce-1.el7.centos
docker-ce-stable
   docker-ce.x86_64            17.06.2.ce-1.el7.centos
docker-ce-stable
   docker-ce.x86_64            17.06.1.ce-1.el7.centos
docker-ce-stable
   docker-ce.x86_64            17.06.0.ce-1.el7.centos
docker-ce-stable
   docker-ce.x86_64            17.03.3.ce-1.el7
docker-ce-stable
```

```
   docker-ce.x86_64            17.03.2.ce-1.el7.centos
docker-ce-stable
   docker-ce.x86_64            17.03.1.ce-1.el7.centos
docker-ce-stable
   docker-ce.x86_64            17.03.0.ce-1.el7.centos
docker-ce-stable
 * base: mirrors.huaweicloud.com

yum -y install docker-ce

systemctl  start docker

systemctl  status docker

systemctl  enable  docker

docker version
```

拉取nexus镜像

```
docker pull sonatype/nexus3:3.16.0

docker images

mkdir /opt/nexus-data

chown -R  200 /opt/nexus-data
```

注：容器中nexus的默认运行用户是nexus,uid和gid为200

运行nexus容器

```
docker run -d  --name nexus  --ulimit nofile=65536:65536  -p
192.168.100.188:8081:8081 -v /opt/nexus-data:/nexus-data
sonatype/nexus3:3.16.0

docker logs  -f  nexus

docker  ps  -a

ss  -tan
```

停止和删除命令行启动的nexus服务

```
# docker stop nexus

# docker rm nexus
```

基于linux系统部署

1.环境说明

安装环境:

操作系统：CentOS Linux release 7.6.1810 (Core)
JDK：jdk1.8 64位
nexus：nexus3.0.0

2 安装jdk
nexus3.x需要JDK1.8支持，所以我们首先在Linux下面安装JDK1.8.

JDK下载地址:

http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

（需要登录才能下载）

私人网盘：链接: https://pan.baidu.com/s/1IcCvfMCTHcQGWwvIAbUkAA 密码: 46lo

第一步：将下载的包解压到自己的安装目录

```
[root@vm1 java]# pwd
/java
[root@vm1 java]# ll
total 252932
```

```
-rw-rw-r-- 1 chenyantao chenyantao 114063112 Jun 21 07:46 jdk-
8u291-linux-x64.rpm
-r-------- 1 chenyantao chenyantao 144935989 Jun 21 07:51 jdk-
8u291-linux-x64.tar.gz

[root@vm1 java]# tar xvf jdk-8u291-linux-x64.tar.gz -C /java/
jdk1.8.0_291/
jdk1.8.0_291/COPYRIGHT
jdk1.8.0_291/LICENSE
jdk1.8.0_291/README.html
jdk1.8.0_291/THIRDPARTYLICENSEREADME.txt
jdk1.8.0_291/bin/
jdk1.8.0_291/bin/java-rmi.cgi
.....
[root@vm1 jdk1.8.0_291]# ll /java/jdk1.8.0_291/
total 25796
drwxr-xr-x 2 10143 10143     4096 Apr  7 15:24 bin
-r--r--r-- 1 10143 10143     3244 Apr  7 15:23 COPYRIGHT
drwxr-xr-x 3 10143 10143      132 Apr  7 15:23 include
-rw-r--r-- 1 10143 10143  5228315 Mar 19 02:57 javafx-src.zip
-rw-r--r-- 1 10143 10143      195 Apr  7 15:24 jmc.txt
drwxr-xr-x 6 10143 10143      198 Apr  7 15:24 jre
drwxr-xr-x 4 10143 10143       31 Apr  7 15:24 legal
drwxr-xr-x 4 10143 10143      223 Apr  7 15:24 lib
-r--r--r-- 1 10143 10143       44 Apr  7 15:23 LICENSE
drwxr-xr-x 4 10143 10143       47 Apr  7 15:23 man
-r--r--r-- 1 10143 10143      159 Apr  7 15:23 README.html
-rw-r--r-- 1 10143 10143      486 Apr  7 15:23 release
-rw-r--r-- 1 10143 10143 21151243 Apr  7 15:23 src.zip
-rw-r--r-- 1 10143 10143      190 Mar 19 02:57
THIRDPARTYLICENSEREADME-JAVAFX.txt
-r--r--r-- 1 10143 10143      190 Apr  7 15:23
THIRDPARTYLICENSEREADME.txt
[root@vm1 jdk1.8.0_291]#
```

第二步：配置系统环境变量

使用vi编辑/etc/profile文件：

```
cat >> /etc/profile <<EOF
export JAVA_HOME=/java/jdk1.8.0_291/
export
CLASSPATH=.:$JAVA_HOME/jre/lib/rt.jar:$JAVA_HOME/lib/dt.jar:$JAVA_
HOME/lib/tools.jar
export PATH=$PATH:$JAVA_HOME/bin
EOF

source /etc/profile
```

第三步：验证
输入java -version命令，如果得到如下信息表示安装成功：

```
[root@vm1 ~]# java -version
java version "1.8.0_291"
Java(TM) SE Runtime Environment (build 1.8.0_291-b10)
Java HotSpot(TM) 64-Bit Server VM (build 25.291-b10, mixed mode)
```

3.安装nexus
nexus下载地址：

官网：需要注册才能下载，http://www.sonatype.com/download-oss-sonatype

私人网盘：

链接: https://pan.baidu.com/s/1496zFST-cOHqN2ogtu57BA 密码: 5k3g

推荐下载3.20.1-01版本的，我使用3.25.0-03版本的部署完毕验证失败客户端无法使用

```
[root@vm1 nexus3]# ll /nexus/nexus-3.2*.tar.gz
-rw-rw-r-- 1 nexus nexus 136225275 Jun 21 09:50 /nexus/nexus-
3.20.1-01-unix.tar.gz
-rw-rw-r-- 1 nexus nexus 160022971 Jun 21 09:53 /nexus/nexus-
3.25.0-03-unix.tar.gz
```

第一步：将下载的文件放到安装目录下，解压

```
[root@vm1 etc]# find /nexus/ -iname *.tar.gz -exec ls -l {} \;
```

```
-rw-rw-r-- 1 chenyantao chenyantao 136225275 Jun 21 09:50
/nexus/nexus-3.20.1-01-unix.tar.gz
-rw-rw-r-- 1 chenyantao chenyantao 160022971 Jun 21 09:53
/nexus/nexus-3.25.0-03-unix.tar.gz
[root@vm1 etc]# find /nexus/ -iname *.tar.gz | xargs ls -l
-rw-rw-r-- 1 chenyantao chenyantao 136225275 Jun 21 09:50
/nexus/nexus-3.20.1-01-unix.tar.gz
-rw-rw-r-- 1 chenyantao chenyantao 160022971 Jun 21 09:53
/nexus/nexus-3.25.0-03-unix.tar.gz
```

我们使用最新版本

```
[root@vm1 etc]# find /nexus/ -iname nexus-3.25.0-03-unix.tar.gz |
xargs -I ff tar xvf ff  -C /nexus/
nexus-3.25.0-03/.install4j/9d17dc87.lprop
nexus-3.25.0-03/.install4j/MessagesDefault
nexus-3.25.0-03/.install4j/build.uuid
nexus-3.25.0-03/.install4j/i4j_extf_0_17is1ik.utf8
nexus-3.25.0-03/.install4j/i4j_extf_10_17is1ik_10358jn.png
nexus-3.25.0-03/.install4j/i4j_extf_11_17is1ik_1gne9sv.png
nexus-3.25.0-03/.install4j/i4j_extf_12_17is1ik_sc8j43.png
nexus-3.25.0-03/.install4j/i4j_extf_13_17is1ik_10nxrsm.png
nexus-3.25.0-03/.install4j/i4j_extf_14_17is1ik_yd7am4.png
nexus-3.25.0-03/.install4j/i4j_extf_15_17is1ik_vu6hgs.png
nexus-3.25.0-03/.install4j/i4j_extf_16_17is1ik_1g1wykh.png
nexus-3.25.0-03/.install4j/i4j_extf_17_17is1ik_18gg8kx.png
nexus-3.25.0-03/.install4j/i4j_extf_17_17is1ik_18gg8kx@2x.png
nexus-3.25.0-03/.install4j/i4j_extf_18_17is1ik_11g5ail.png
nexus-3.25.0-03/.install4j/i4j_extf_19_17is1ik_fyoktp.png
nexus-3.25.0-03/.install4j/i4j_extf_1_17is1ik.properties
nexus-3.25.0-03/.install4j/i4j_extf_20_17is1ik_11g5ail.png
nexus-3.25.0-03/.install4j/i4j_extf_21_17is1ik_wtm4no.png
nexus-3.25.0-03/.install4j/i4j_extf_2_17is1ik_dfqahl.png
nexus-3.25.0-03/.install4j/i4j_extf_3_17is1ik_ijvpzt.png
nexus-3.25.0-03/.install4j/i4j_extf_4_17is1ik_2foqqs.png
nexus-3.25.0-03/.install4j/i4j_extf_5_17is1ik_iqj32.png
nexus-3.25.0-03/.install4j/i4j_extf_6_17is1ik_1piu8ry.png
nexus-3.25.0-03/.install4j/i4j_extf_7_17is1ik.html
nexus-3.25.0-03/.install4j/i4j_extf_8_17is1ik_1niwaxy.ico
nexus-3.25.0-03/.install4j/i4j_extf_9_17is1ik_1m92816.png
nexus-3.25.0-03/.install4j/i4jparams.conf
nexus-3.25.0-03/.install4j/i4jruntime.jar
nexus-3.25.0-03/NOTICE.txt
....

[root@vm1 etc]# ll /nexus/nexus-3.25.0-03/
```

```
total 76
drwxr-xr-x  3 root root    73 Jun 21 10:02 bin
drwxr-xr-x  2 root root    26 Jun 21 10:02 deploy
drwxr-xr-x  7 root root   104 Jun 21 10:02 etc
drwxr-xr-x  5 root root   206 Jun 21 10:02 lib
-rw-r--r--  1 root root   395 Jul  8  2020 NOTICE.txt
-rw-r--r--  1 root root 17321 Jul  8  2020 OSS-LICENSE.txt
-rw-r--r--  1 root root 41954 Jul  8  2020 PRO-LICENSE.txt
drwxr-xr-x  3 root root  4096 Jun 21 10:02 public
drwxr-xr-x 21 root root  4096 Jun 21 10:02 system
[root@vm1 etc]#
```

配置 nexus

创建nexus用户，并设置该用户File Handle Limits

```
[root@vm1 etc]# useradd nexus
[root@vm1 etc]# echo "nexus - nofile 65536" >>
/etc/security/limits.conf
[root@vm1 etc]# # mv /nexus/nexus-3.25.0-03/ /nexus/nexus-3
[root@vm1 etc]# chown -R nexus:nexus /nexus/
# 设置服务启动用户
[root@vm1 etc]# echo 'run_as_user="nexus"' > /nexus/nexus-3.25.0-
03/bin/nexus.rc

启动nexus
[root@vm1 log]# /nexus/nexus-3.25.0-03/bin/nexus start

# 最后，查看log了解服务运行状态
[root@vm1 log]# tail -1000f /nexus/sonatype-
work/nexus3/log/nexus.log
....
```

第三步：访问私服
nexus启动成功之后，我们就可以访问咱们的私服了。默认的端口是8081， RUL为：http://serveripaddress:port ， e.g. http://localhost:8081/。

如果忘记密码按照如下步骤修改密码:

此处我们将admin用户密码重置为admin123，具体执行如下：

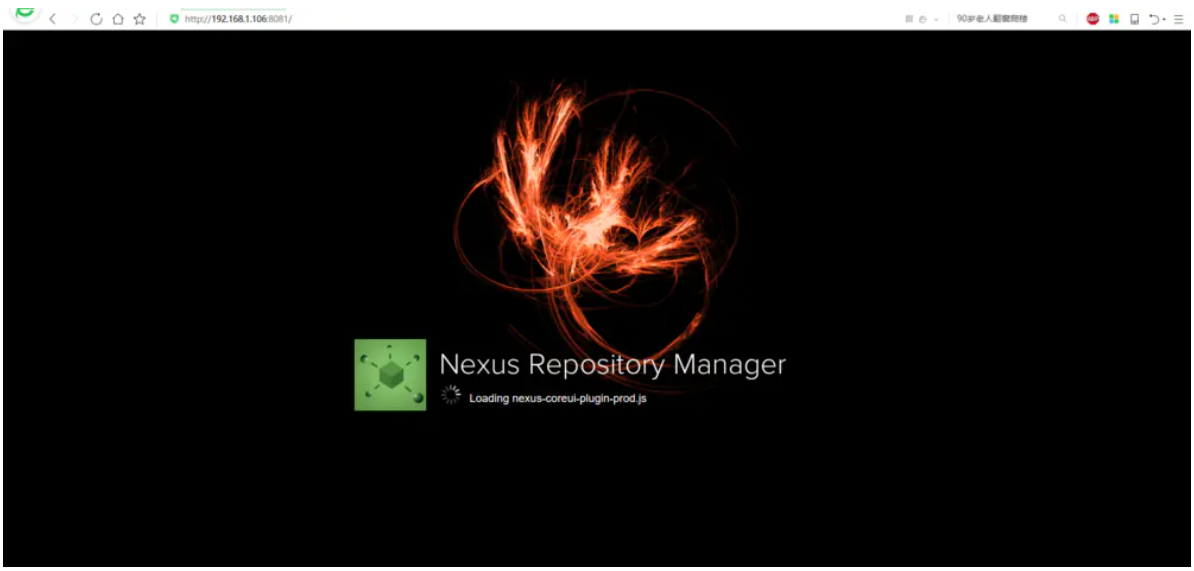
```
1、停服务
[root@vm1 ~]# /nexus/nexus-3.25.0-03/bin/nexus stop
2、切换数据库
```

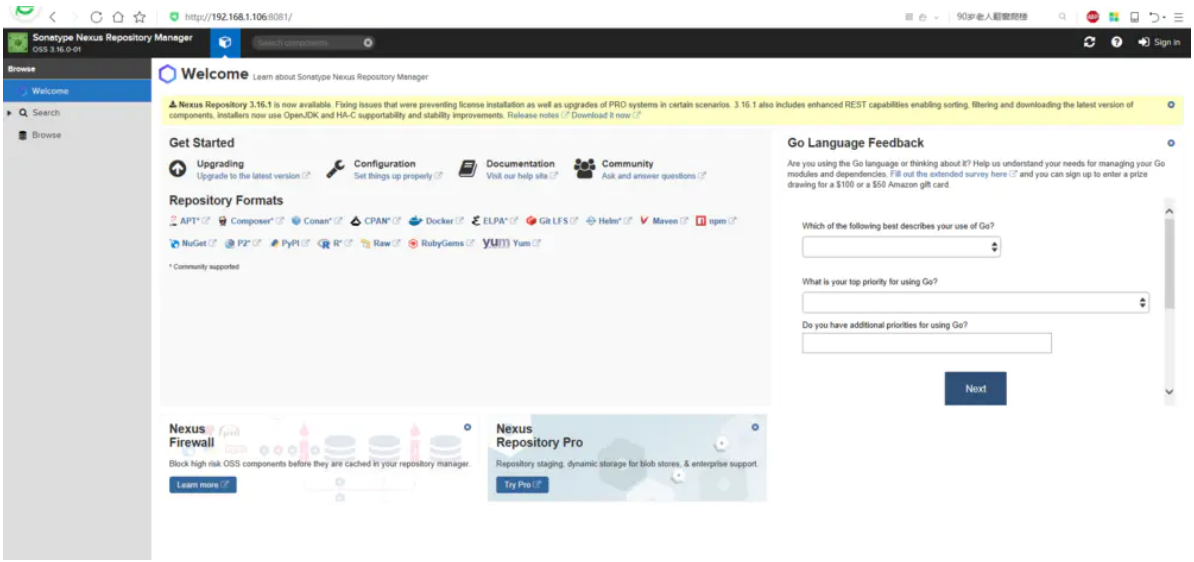
```
[root@vm1 ~]# java -jar /nexus/nexus-3.25.0-03/lib/support/nexus-
orient-console.jar

OrientDB console v.2.2.36 (build
d3beb772c02098ceaea89779a7afd4b7305d3788, branch 2.2.x)
https://www.orientdb.com
Type 'help' to display all the supported commands.
```
3、连接数据库
```
orientdb> connect plocal:/nexus/sonatype-work/nexus3/db/security/
admin admin

Connecting to database [plocal:/nexus/sonatype-
work/nexus3/db/security/] with user 'admin'...
2021-06-22 08:06:28:798 WARNI {db=security} Storage 'security' was
not closed properly. Will try to recover from write ahead log...
[OLocalPaginatedStorage]
2021-06-22 08:06:28:802 WARNI {db=security} Record
com.orientechnologies.orient.core.storage.impl.local.paginated.wal
.OCheckpointEndRecord{lsn=LSN{segment=12, position=52}} will be
skipped during data restore [OLocalPaginatedStorage]OK
```
4、查看系统用户
```
orientdb {db=security}> select * from user where id = "admin"

+----+-----+------+-----+------+--------+-----------+-------------
--+-----------------------------------------------------------------
-----------------------------------------------+
|#   |@RID |@CLASS|id   |status|lastName|firstName  |email
|password
                                                 |
+----+-----+------+-----+------+--------+-----------+-------------
--+-----------------------------------------------------------------
-----------------------------------------------+
|0   |#12:0|user  |admin|active|User
|Administ...|admin@exampl...|$shiro1$SHA-
512$1024$ke/xytEvnWV90kG+E84Ukw==$pGKtjMXdNedIVGMul7vs2mpGGDEA46ED
5ckQjtOFkw5ODgQQXX6oe6Ssoofb...|
+----+-----+------+-----+------+--------+-----------+-------------
--+-----------------------------------------------------------------
-----------------------------------------------+

1 item(s) found. Query executed in 0.005 sec(s).
```
5、修改密码

```
orientdb {db=security}> update user SET password="$shiro1$SHA-
512$1024$NE+wqQq/TmjZMvfI7ENh/g==$V4yPw8T64UQ6GfJfxYq2hLsVrBY8D1v+
bktfOxGdt4b/9BthpWPNUy/CBk6V9iA0nHpzYzJFWO8v/tZFtES8CA==" UPSERT
WHERE id="admin"

Updated record(s) '1' in 0.034000 sec(s).

orientdb {db=security}> select * from user where id = "admin"



+----+-----+------+-----+------+--------+----------+-------------
--+-----------------------------------------------------------------
-----------------------------------------------+
|#   |@RID |@CLASS|id   |status|lastName|firstName  |email
|password
                                                 |
+----+-----+------+-----+------+--------+----------+-------------
--+-----------------------------------------------------------------
-----------------------------------------------+
|0   |#12:0|user  |admin|active|User
|Administ...|admin@exampl...|$shiro1$SHA-
512$1024$NE+wqQq/TmjZMvfI7ENh/g==$V4yPw8T64UQ6GfJfxYq2hLsVrBY8D1v+
bktfOxGdt4b/9BthpWPNUy/CBk6V...|
+----+-----+------+-----+------+--------+----------+-------------
--+-----------------------------------------------------------------
-----------------------------------------------+

1 item(s) found. Query executed in 0.002 sec(s).
orientdb {db=security}> quit

7、启动服务
[root@vm1 ~]# /nexus/nexus-3.25.0-03/bin/nexus start
```

配置nesux创建仓库

浏览器访问: http:192.168.100.181:8081





默认登录用户名密码：admin/admin123

yum私服有三种类型：

- `hosted`：本地存储，即同 yum 官方仓库一样提供本地私服功能
- `proxy`：提供代理其他仓库的类型，如我们常用的163仓库
- `group`：组类型，实质作用是组合多个仓库为一个地址，相当于一个透明代理**

那么就来一个一个创建。

1，创建blob存储。

为其创建一个单独的存储空间，命名为`yum-hub`

1）Type
选择"File"。

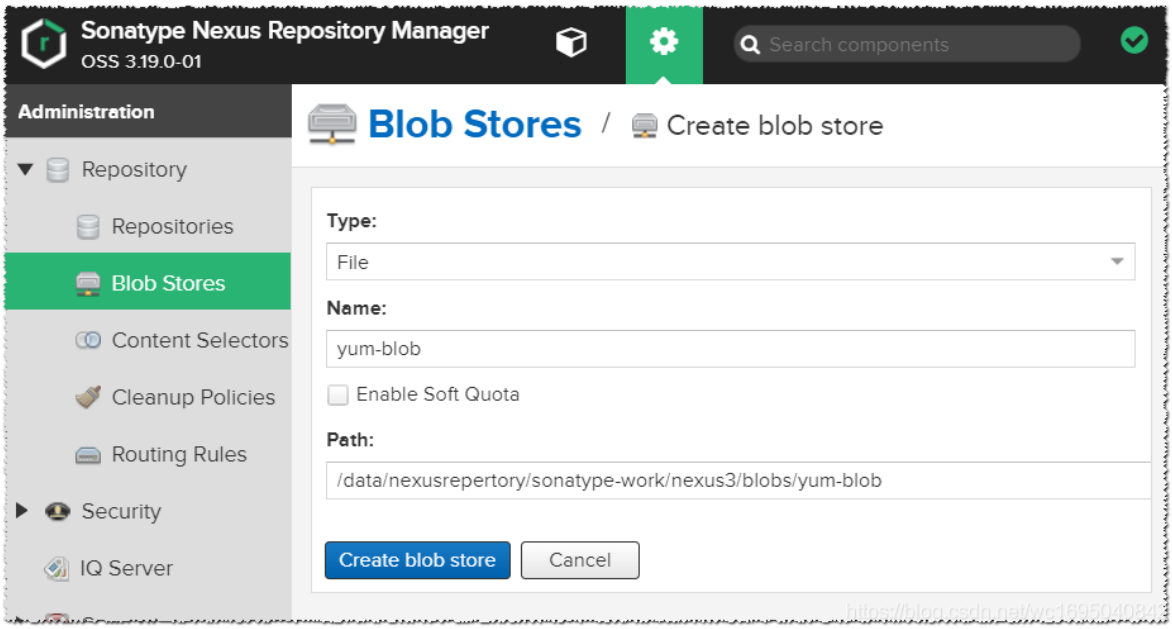
2）Name
就叫yum-blob吧。

3）Enable Soft Quota

限制目录的大小。我这边就不限制了。如果要限制的话，就勾选上，并填上限制的条件和限制的值就OK了。

4）Path

在填入Name之后，path会自动生成。



2、创建一个hosted类型的仓库

点击"Repository"–>“Repositories”–>“Create repository”，选择yum(hosted)。

1）Name

就叫yum-hosted-my吧。

2）Online

勾选，可以设置这个仓库是在线还是离线。

3）Yum

Repodata Depth：指定创建repodata文件夹的存储库深度，这里选择"2"。

Deploy Policy：布局策略

Strict：严格

Permissive：宽松

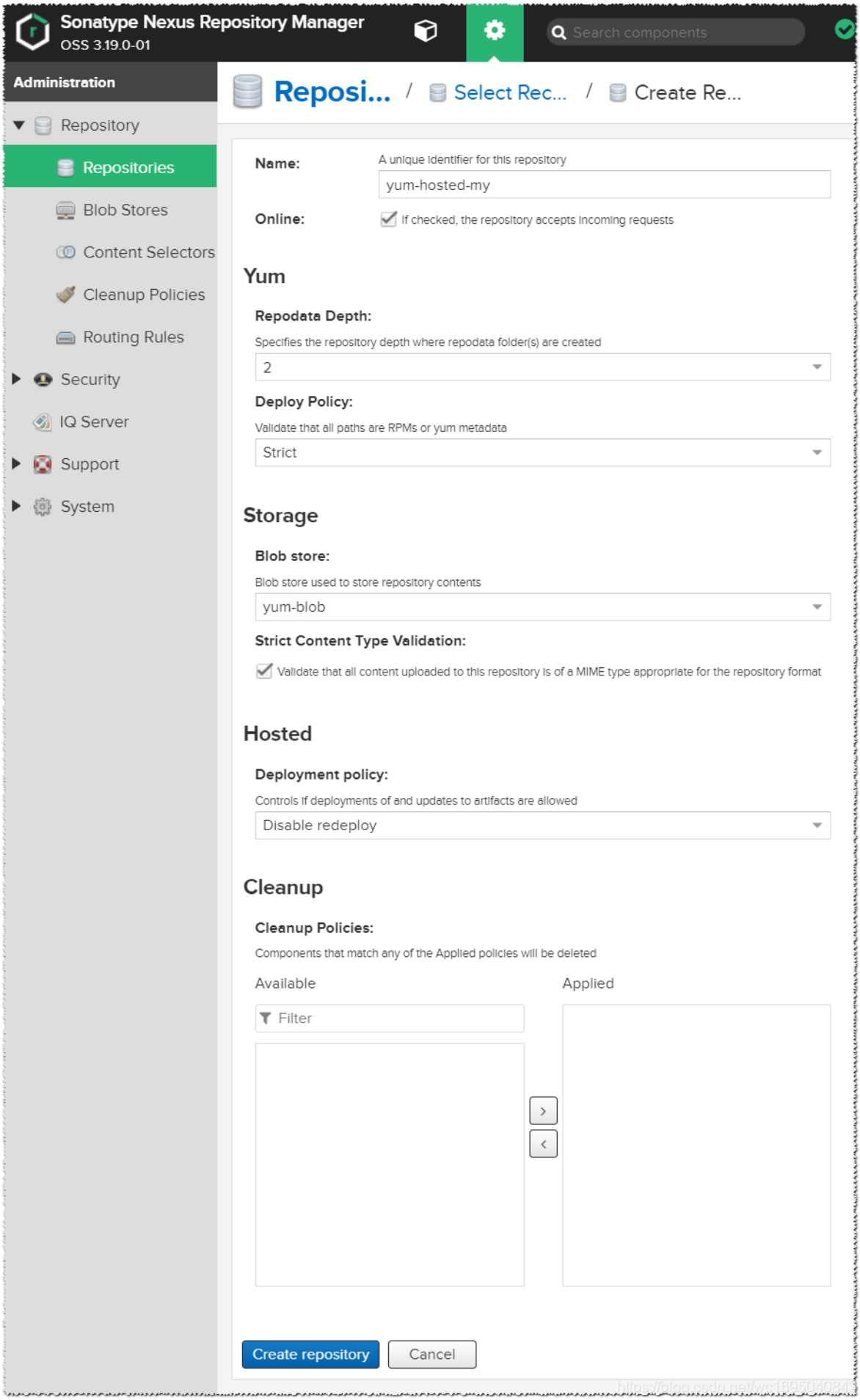
这里选择默认的Strict。

4）Storang
Blob store：选择此仓库使用的Blob存储，这里选择之前创建的yum-blob。
Strict Content Type Validation：验证上传内容格式，这里就用默认的勾选。

5）Hosted
Deployment Policy：部署策略，有三个选项，分别是：
Allow Redeploy：允许重新部署
Disable Redeploy：禁止重新部署
Read-Only：只读

我这里使用默认的"Disable Redeploy"，如果是开发环境，可以选择"Allow Redeploy"。

6）Cleanup
Cleanup Policies：清除策略，这个是新增的功能，这里先不进行设置


Sonatype Nexus Repository Manager
OSS 3.19.0-01
Search components
Administration
Repository
Repositories
Blob Stores
Content Selectors
Cleanup Policies
Routing Rules
Security
IQ Server
Support
System
Reposi...
Select Rec...
Create Re...
Name:
A unique identifier for this repository
yum-hosted-my
Online:
If checked, the repository accepts incoming requests
Yum
Repodata Depth:
Specifies the repository depth where repodata folder(s) are created
2
Deploy Policy:
Validate that all paths are RPMs or yum metadata
Strict
Storage
Blob store:
Blob store used to store repository contents
yum-blob
Strict Content Type Validation:
Validate that all content uploaded to this repository is of a MIME type appropriate for the repository format
Hosted
Deployment policy:
Controls if deployments of and updates to artifacts are allowed
Disable redeploy
Cleanup
Cleanup Policies:
Components that match any of the Applied policies will be deleted
Available
Applied
Filter
Create repository
Cancel

3，创建一个proxy类型的yum仓库。

代理仓库（Proxy Repository）是远程仓库的代理，当用户向这个代理仓库请求一个依赖包时，这个代理仓库会先在本地查找，如果存在，会直接提供给用户进行下载；如果在代理仓库本地查找不到，就会从配置的远程中央仓库中进行下载，下载到私服上之后再提供给用户下载。所以一般我们把私服架设在内网之中，这样可以节省外网带宽，并且大大提高了用户下载依赖的速度。
点击"Repository"–>“Repositories”–>“Create repository”，选择yum(proxy)。

1）Name
因为我要代理阿里云的yum仓库，所以就叫"yum-proxy-aliyun"。

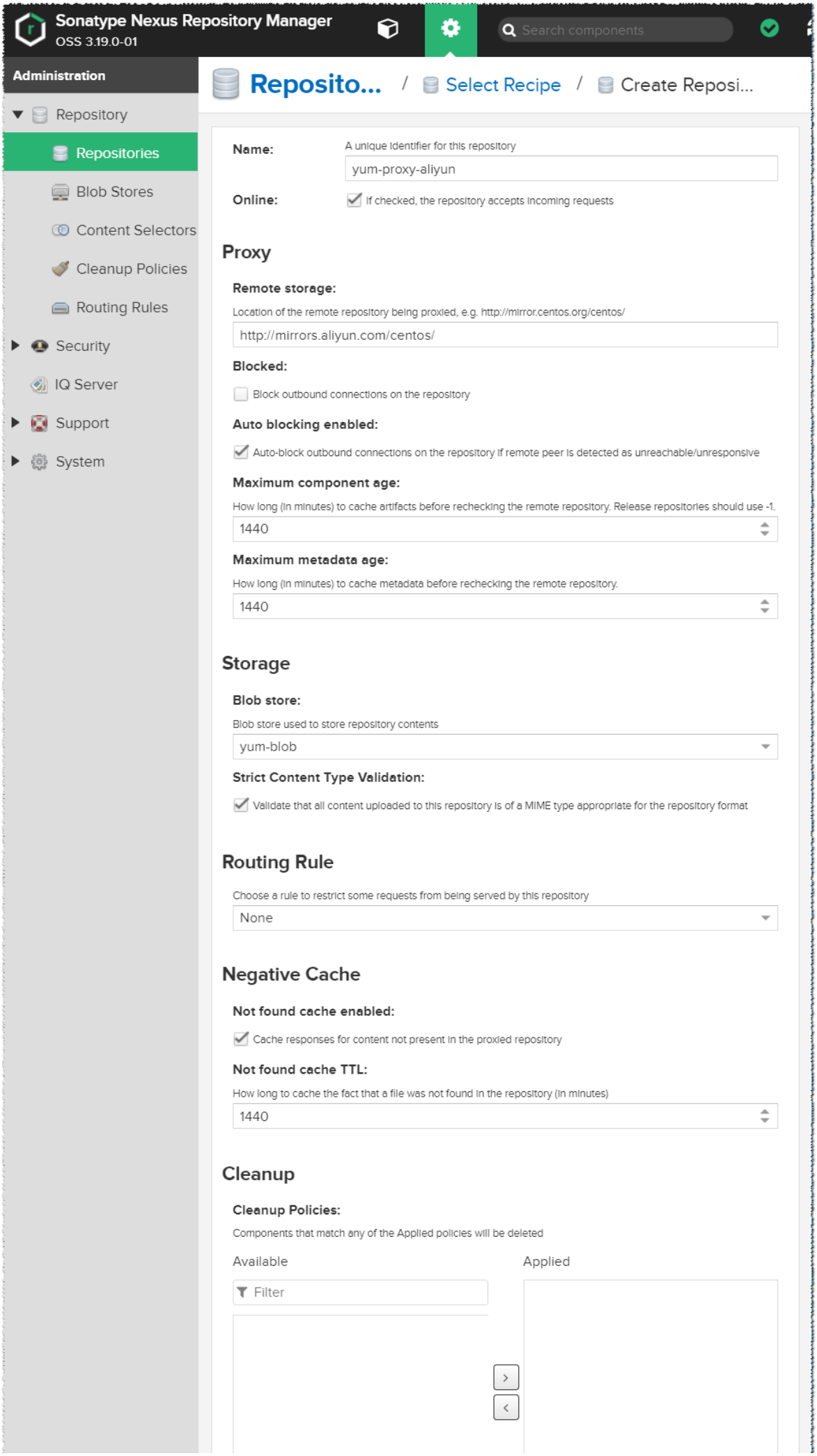
2）Online
勾选，设置成在线。

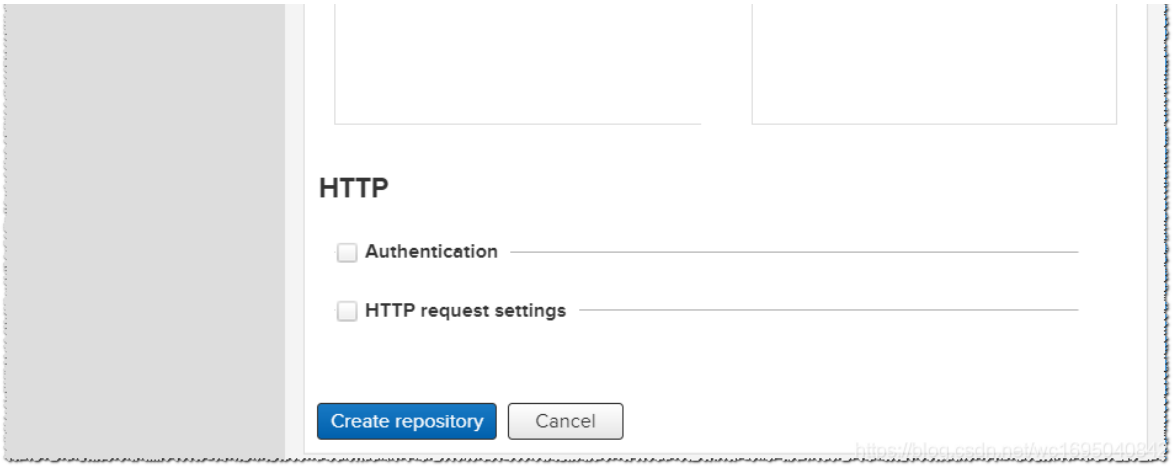
3）Proxy
Remote storage：设置远程中央仓库的地址，我这里设置成阿里云的yum仓库地址—http://mirrors.aliyun.com/centos/

其他的用默认值即可。

4）Storage
Blob store：选择yum-blob
Strict Content Type Validation：验证上传内容格式，这里就用默认的勾选。

5）Routing,Negative Cache,Cleanup,HTTP
都使用默认配置。


Sonatype Nexus Repository Manager
OSS 3.19.0-01
Search components
Administration
Repository
Repositories
Blob Stores
Content Selectors
Cleanup Policies
Routing Rules
Security
IQ Server
Support
System
Reposito...
Select Recipe
Create Reposi...
Name:
A unique identifier for this repository
yum-proxy-aliyun
Online:
If checked, the repository accepts incoming requests
Proxy
Remote storage:
Location of the remote repository being proxied, e.g. http://mirror.centos.org/centos/
http://mirrors.aliyun.com/centos/
Blocked:
Block outbound connections on the repository
Auto blocking enabled:
Auto-block outbound connections on the repository if remote peer is detected as unreachable/unresponsive
Maximum component age:
How long (in minutes) to cache artifacts before rechecking the remote repository. Release repositories should use -1.
1440
Maximum metadata age:
How long (in minutes) to cache metadata before rechecking the remote repository.
1440
Storage
Blob store:
Blob store used to store repository contents
yum-blob
Strict Content Type Validation:
Validate that all content uploaded to this repository is of a MIME type appropriate for the repository format
Routing Rule
Choose a rule to restrict some requests from being served by this repository
None
Negative Cache
Not found cache enabled:
Cache responses for content not present in the proxied repository
Not found cache TTL:
How long to cache the fact that a file was not found in the repository (in minutes)
1440
Cleanup
Cleanup Policies:
Components that match any of the Applied policies will be deleted
Available
Applied
Filter



4，创建一个group类型的yum仓库。

仓库组（Repository Group）的目的是将多个仓库（代理仓库和宿主仓库）聚合，对用户暴露统一的地址。当用户需要获取某一个依赖包时，请求的是仓库组的地址，系统将会根据仓库组配置的仓库顺序依次查找。

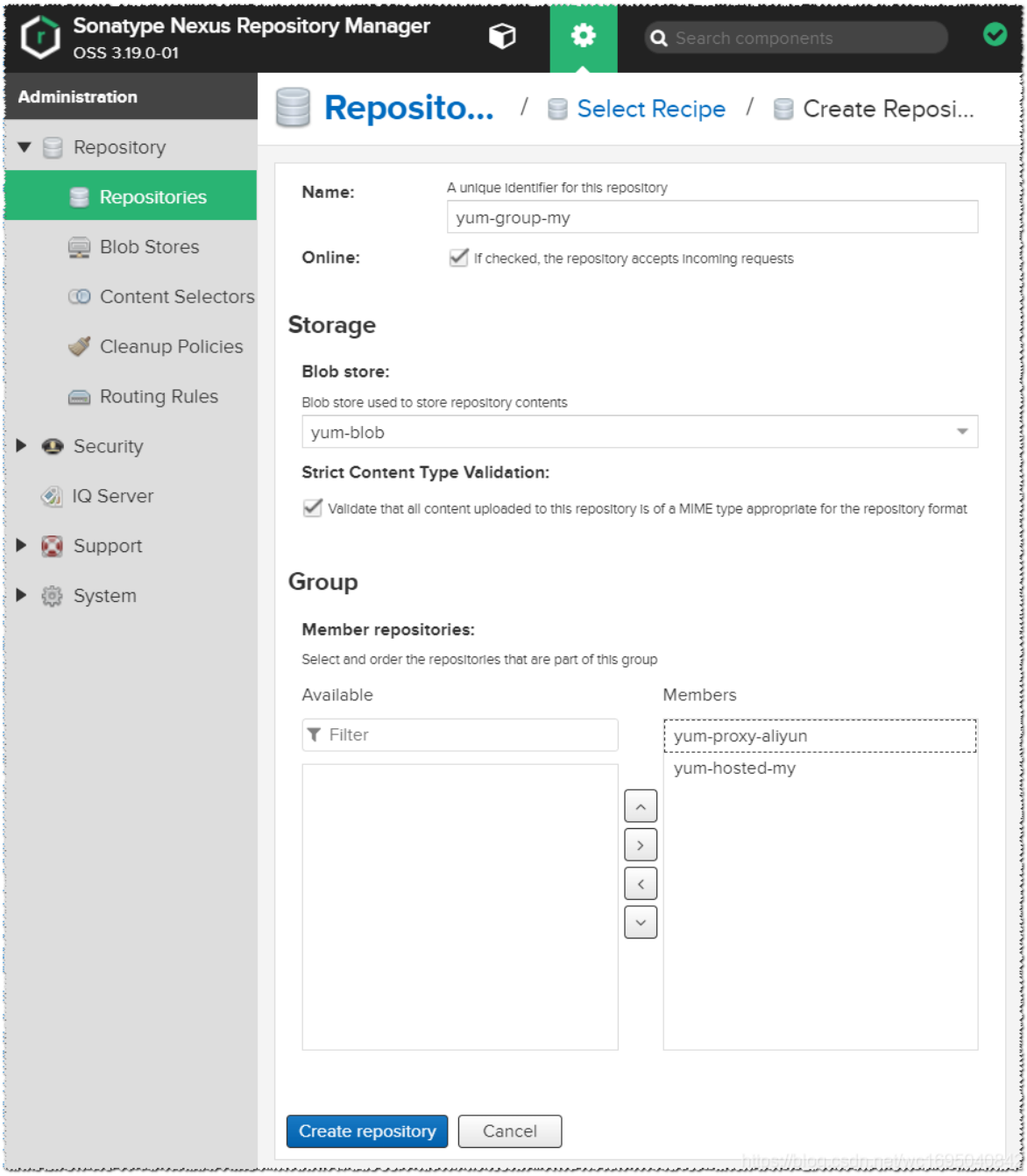
点击"Repository"–>“Repositories”–>“Create repository”，选择yum(gruop)。

1）Name
yum-group-my

2）Online
勾选，设置成在线

3）Storage
Blob store：选择yum-blob
Strict Content Type Validation：使用默认的勾选

4）Group
将左侧的Available中的仓库列表添加到右侧的Members中。



5，构建缓存。

新建一台环境干净的主机，此时需要保证这台主机能够上网，因为私服当中还没有进行初始化。

先简单配置一下，将yum源指向到私服中来。

1，将原有的移走。

```
[root@7-3 ~]$ cd /etc/yum.repos.d/
[root@7-3 yum.repos.d]$ls
CentOS-Base.repo  CentOS-CR.repo  CentOS-Debuginfo.repo  CentOS-
fasttrack.repo  CentOS-Media.repo  CentOS-Sources.repo  CentOS-
Vault.repo
[root@7-3 yum.repos.d]$mkdir bak
[root@7-3 yum.repos.d]$mv * bak
mv: cannot move ‘bak’ to a subdirectory of itself, ‘bak/bak’
[root@7-3 yum.repos.d]$ls
bak
```

2，创建一个新的源。

```
[root@7-3 yum.repos.d]$ vim nexus.repo
```

添加如下内容：

其中的url就是私服当中创建的group的对外地址，后面的`$releasever/os/$basearch/`不要漏掉了。

```
[nexus]
name=Nexus Repository
baseurl=http://192.168.106.65:8081/repository/group-
yum/$releasever/os/$basearch/
enabled=1
gpgcheck=0
```

注意这还不是完整内容，我第一次构建的时候只写了这些内容，以求私服自己能够通过刚刚配置的proxy将远程的包拉下来，最后发现这种方式，死活都是无法成功的。

因此，这里还应该将163的源配置添加进来。

完整内容应该如下：

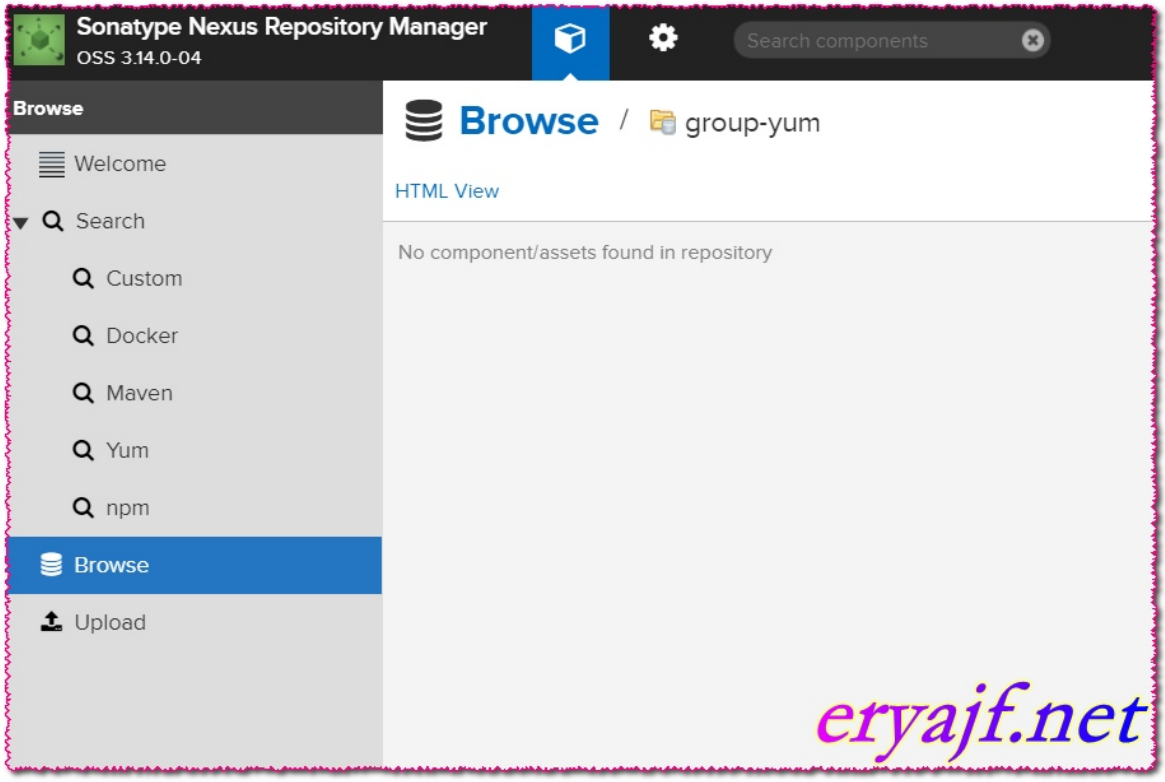
```
[root@7-3 yum.repos.d]$cat nexus.repo
[nexus]
name=Nexus Repository
baseurl=http://192.168.106.65:8081/repository/group-
yum/$releasever/os/$basearch/
enabled=1
gpgcheck=0

#released updates
[updates]
name=CentOS-$releasever-Updates-163.com
```

```
#mirrorlist=http://mirrorlist.centos.org/?
release=$releasever&arch=$basearch&repo=updates
baseurl=http://mirrors.163.com/centos/$releasever/updates/$basearc
h/
gpgcheck=1
gpgkey=http://mirrors.163.com/centos/RPM-GPG-KEY-CentOS-7

#additional packages that may be useful
[extras]
name=CentOS-$releasever-Extras-163.com
#mirrorlist=http://mirrorlist.centos.org/?
release=$releasever&arch=$basearch&repo=extras
baseurl=http://mirrors.163.com/centos/$releasever/extras/$basearch
/
gpgcheck=1
gpgkey=http://mirrors.163.com/centos/RPM-GPG-KEY-CentOS-7

#additional packages that extend functionality of existing
packages
[centosplus]
name=CentOS-$releasever-Plus-163.com
baseurl=http://mirrors.163.com/centos/$releasever/centosplus/$base
arch/
gpgcheck=1
enabled=0
gpgkey=http://mirrors.163.com/centos/RPM-GPG-KEY-CentOS-7
```

3，构建缓存。

现在，就可以通过makecache将远程的包拉到内部私服当中了。

操作之前，就像古代变戏法一般的，依旧先去私服看一眼`group-yum`当中是否有包存在，这是一个固定流程哈。

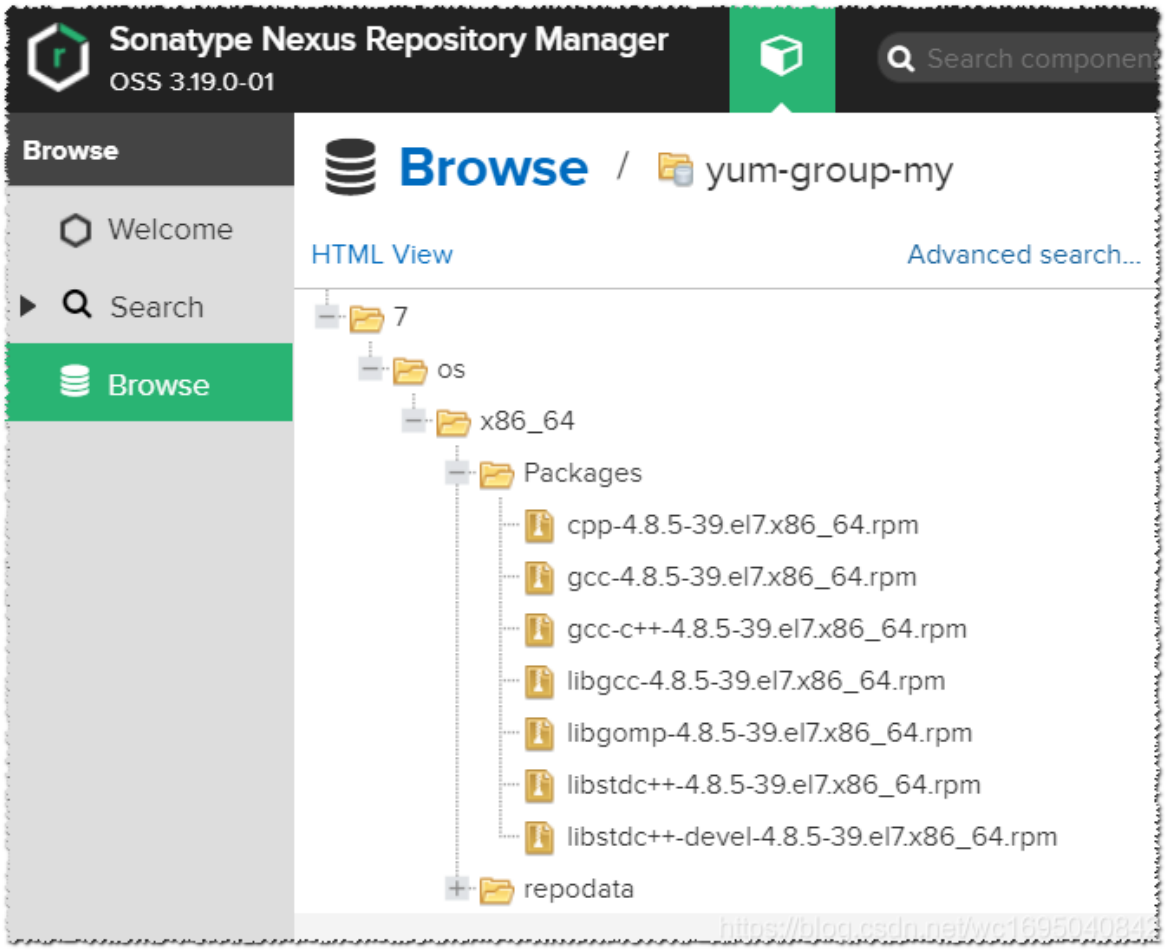


可以看到空空如也，那么通过如下三步操作创建缓存。

```
[root@7-3 yum.repos.d]$yum clean all
Loaded plugins: fastestmirror
Cleaning repos: extras nexus updates
Cleaning up everything
Cleaning up list of fastest mirrors
[root@7-3 yum.repos.d]$yum makecache
Loaded plugins: fastestmirror
extras

                                | 3.4 kB  00:00:00
nexus

                                | 1.8 kB  00:00:00
updates

                                | 3.4 kB  00:00:00
(1/12): extras/7/x86_64/prestodelta

                                | 100 kB  00:00:00
(2/12): extras/7/x86_64/primary_db

                                | 204 kB  00:00:00
(3/12): extras/7/x86_64/other_db

                                | 126 kB  00:00:00
```

```
(4/12): extras/7/x86_64/filelists_db

                                | 604 kB  00:00:00
(5/12): nexus/7/x86_64/group_gz

                                 | 167 kB  00:00:00
(6/12): nexus/7/x86_64/primary

                                | 2.9 MB  00:00:00
(7/12): nexus/7/x86_64/other

                                | 1.6 MB  00:00:00
(8/12): nexus/7/x86_64/filelists

                                | 7.1 MB  00:00:00
(9/12): updates/7/x86_64/prestodelta

                                | 679 kB  00:00:00
(10/12): updates/7/x86_64/filelists_db

                                | 3.4 MB  00:00:00
(11/12): updates/7/x86_64/other_db

                                | 578 kB  00:00:00
(12/12): updates/7/x86_64/primary_db

                                | 6.0 MB  00:00:01
Determining fastest mirrors
nexus

                                                9911/9911
nexus

                                                9911/9911
nexus

                                                9911/9911
Metadata Cache Created

[root@7-3 yum.repos.d]$yum update -y #这个过程比较长，内容比较多，不完全复制了。
```

当上边的第三步执行完成之后，此时我们可以回到刚刚那个空白的页面，看看内容是否上来了。



就是这么神奇。

6，验证一下效果。

验证的方式其实也很简单，找一台不能上网但是可以与刚刚私服通信的主机，将其yum源指向的配置好的私服，看看安装软件什么的是否可以so easy。

或者是将其他的源都切断，然后yum源仅仅指向私服，看看安装是否顺利。

这里采用第二种方式简单试验一下。

1，将原有的移走。

Copy

```
[root@7-2 ~]$cd /etc/yum.repos.d/
[root@7-2 yum.repos.d]$ls
CentOS-Base.repo  CentOS-CR.repo  CentOS-Debuginfo.repo  CentOS-
fasttrack.repo  CentOS-Media.repo  CentOS-Sources.repo  CentOS-
Vault.repo
[root@7-2 yum.repos.d]$mkdir bak
[root@7-2 yum.repos.d]$mv * bakmv: cannot move ‘bak’ to a
subdirectory of itself, ‘bak/bak’[root@7-2 yum.repos.d]$lsbak
```

此时尝试一下安装。

Copy

```
[root@7-2 yum.repos.d]$yum -y install httpd Loaded plugins:
fastestmirror Determining fastest mirrorsThere are no enabled
repos. Run "yum repolist all" to see the repos you have. To enable
Red Hat Subscription Management repositories:     subscription-
manager repos --enable <repo> To enable custom repositories:
yum-config-manager --enable <repo>
```

2，创建一个新的源。

Copy

```
[root@7-2 yum.repos.d]$cat nexus.repo
[nexus]name=Nexus Repository
baseurl=http://192.168.106.65:8081/repository/group-
yum/$releasever/os/$basearch/
enabled=1
gpgcheck=0
```

再尝试安装：

Copy

```
[root@7-2 yum.repos.d]$yum -y install httpd
```

```
Loaded plugins: fastestmirrornexus

                                                                | 1.8
kB  00:00:00(1/2): nexus/7/x86_64/group_gz

                                            | 167 kB  00:00:00(2/2):
nexus/7/x86_64/primary

                        | 2.9 MB  00:00:00Loading mirror speeds
from cached hostfilenexus


9911/9911Resolving Dependencies--> Running transaction check--->
Package httpd.x86_64 0:2.4.6-80.el7.centos will be installed-->
Processing Dependency: httpd-tools = 2.4.6-80.el7.centos for
package: httpd-2.4.6-80.el7.centos.x86_64--> Processing
Dependency: /etc/mime.types for package: httpd-2.4.6-
80.el7.centos.x86_64--> Processing Dependency: libaprutil-1.so.0()
(64bit) for package: httpd-2.4.6-80.el7.centos.x86_64-->
Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-
2.4.6-80.el7.centos.x86_64--> Running transaction check--->
Package apr.x86_64 0:1.4.8-3.el7_4.1 will be installed---> Package
apr-util.x86_64 0:1.5.2-6.el7 will be installed---> Package httpd-
tools.x86_64 0:2.4.6-80.el7.centos will be installed---> Package
mailcap.noarch 0:2.1.41-2.el7 will be installed--> Finished
Dependency Resolution Dependencies Resolved
================================================================
================================================================
============================================== Package
                                  Arch
       Version
Repository
Size================================================================
================================================================
============================================================Installi
ng: httpd                                          x86_64
                       2.4.6-80.el7.centos
           nexus                                    2.7 MInstalling
for dependencies: apr
x86_64                                    1.4.8-3.el7_4.1
                              nexus
103 k apr-util                                        x86_64
                         1.5.2-6.el7
             nexus                                      92 k httpd-
tools                                    x86_64
```

```
            2.4.6-80.el7.centos
nexus                                    89 k mailcap
                      noarch
2.1.41-2.el7                                          nexus
                           31 k Transaction
Summary=============================================================
=============================================================
=============================================================Insta
ll  1 Package (+4 Dependent packages) Total download size: 3.0
MInstalled size: 10 MDownloading packages:(1/5): apr-1.4.8-
3.el7_4.1.x86_64.rpm

             | 103 kB  00:00:00(2/5): apr-util-1.5.2-
6.el7.x86_64.rpm

        |  92 kB  00:00:00(3/5): httpd-tools-2.4.6-
80.el7.centos.x86_64.rpm

     |  89 kB  00:00:00(4/5): mailcap-2.1.41-2.el7.noarch.rpm

                                                    |  31 kB
00:00:00(5/5): httpd-2.4.6-80.el7.centos.x86_64.rpm

                                        | 2.7 MB  00:00:03---------
-------------------------------------------------------------------
-------------------------------------------------------------------
---------------------------------------------Total


880 kB/s | 3.0 MB  00:00:03Running transaction checkRunning
transaction testTransaction test succeededRunning transaction
Installing : apr-1.4.8-3.el7_4.1.x86_64

                                                  1/5  Installing :
apr-util-1.5.2-6.el7.x86_64

                                    2/5  Installing : httpd-tools-
2.4.6-80.el7.centos.x86_64

                      3/5  Installing : mailcap-2.1.41-
2.el7.noarch

                   4/5  Installing : httpd-2.4.6-
80.el7.centos.x86_64
```

```
                      5/5  Verifying  : httpd-tools-2.4.6-
80.el7.centos.x86_64

               1/5  Verifying  : apr-1.4.8-3.el7_4.1.x86_64


  2/5  Verifying  : mailcap-2.1.41-2.el7.noarch

                                                          3/5
Verifying  : httpd-2.4.6-80.el7.centos.x86_64

                                                 4/5  Verifying  :
apr-util-1.5.2-6.el7.x86_64

                                    5/5 Installed:  httpd.x86_64
0:2.4.6-80.el7.centos Dependency Installed:  apr.x86_64 0:1.4.8-
3.el7_4.1              apr-util.x86_64 0:1.5.2-6.el7
httpd-tools.x86_64 0:2.4.6-80.el7.centos
mailcap.noarch 0:2.1.41-2.el7 Complete!
```

服务端启动方式改进，将nexus注册成系统服务

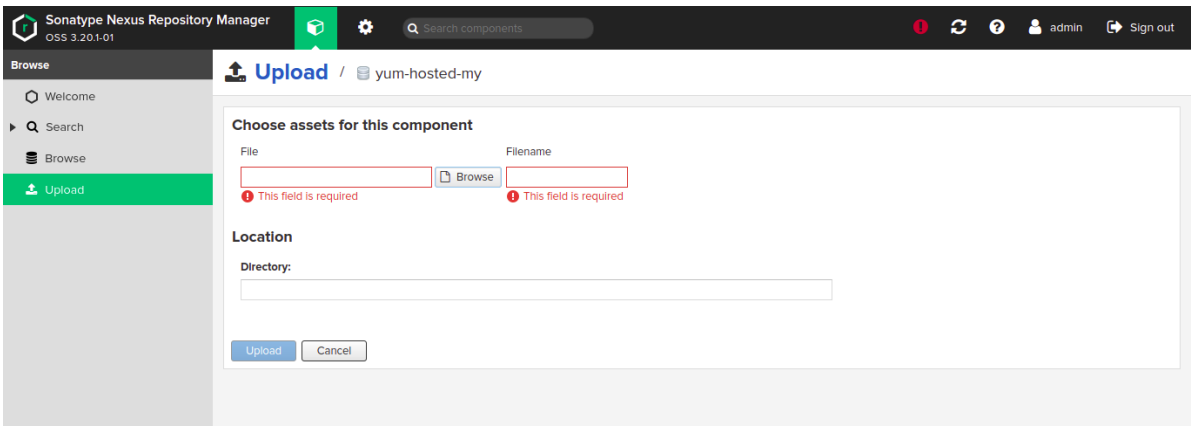
```
vim /etc/systemd/system/nexus.service

########################################################

[Unit]

Description=Nexus

Documentation=https://www.sonatype.com

After=network-online.target firewalld.service docker.service

Requires=docker.service

[Service]

ExecStartPre=-/usr/bin/docker rm -f nexus

ExecStart=/usr/bin/docker run \

--name nexus \

--ulimit nofile=65536:65536 \
```

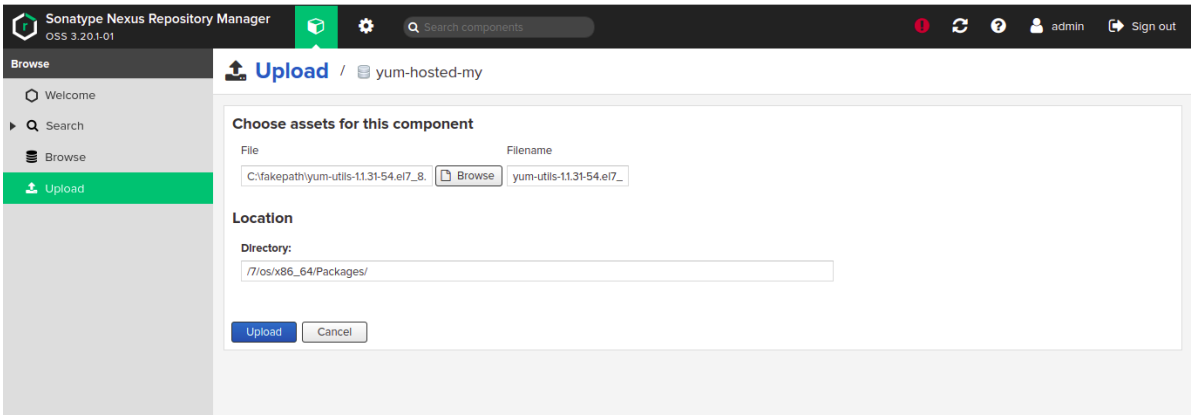
```
-p 192.168.1.107:8081:8081 \

-v /opt/nexus-data:/nexus-data \

sonatype/nexus3:3.16.0

ExecStop=/usr/bin/docker stop nexus

LimitNOFILE=65535

Restart=on-failure

StartLimitBurst=3

StartLimitInterval=60s

[Install]

WantedBy=multi-user.target

##########################################################



用systemd启动服务

systemctl daemon-reload

systemctl start  nexus

systemctl enable nexus

systemctl status nexus
```

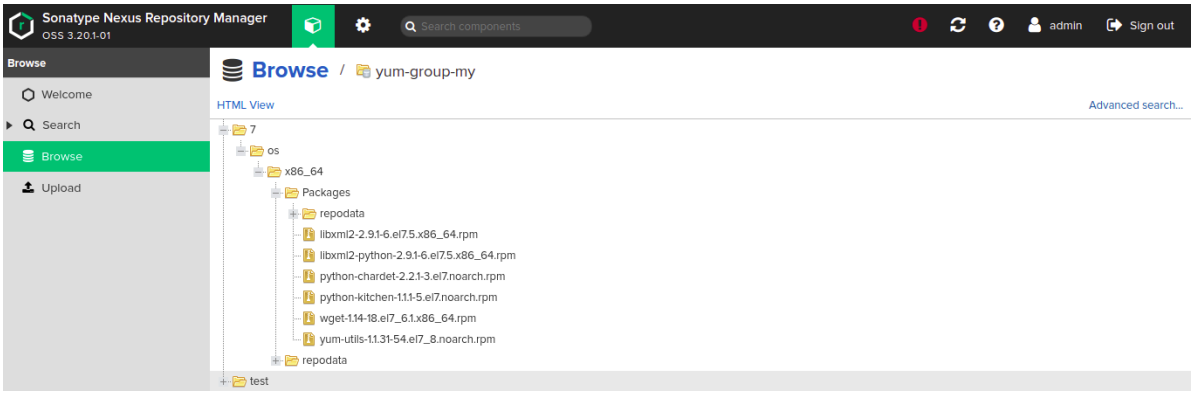
4、验证上传

我这里有yum-utils所需的包及依赖包，现在使用这些包进行上传测试

```
[root@vm1 packages]# find /var/cache/yum/x86_64/7/ -iname *.rpm -
exec basename {} \;
libxml2-2.9.1-6.el7.5.x86_64.rpm
libxml2-python-2.9.1-6.el7.5.x86_64.rpm
python-chardet-2.2.1-3.el7.noarch.rpm
python-kitchen-1.1.1-5.el7.noarch.rpm
yum-utils-1.1.31-54.el7_8.noarch.rpm
```







5、参考

nexus3搭建yum源

https://blog.51cto.com/daibaiyang119/2116205

http://limingming.org/index.php/2018/12/nexus3-yum-repo

企业级开源仓库nexus3实战应用

http://www.eryajf.net/category/%E6%9C%AF%E4%B8%9A%E4%B8%93%E6%94%BB/%E6%9C%8D%E5%8A%A1%E7%B1%BB%E7%9B%B8%E5%85%B3/nexus