

搭建Docker（安装中间件）

一.更换腾讯YUM源

- 1. 虚拟机网络设置为NAT模式
- 2. 备份当前的YUM源配置文件（一开始是用阿里云的但是不成功所以换了腾讯云（不要用阿里云的巨坑浪费了我半天）

```
mv /etc/yum.repos.d/CentOS-Base.repo /etc/yum.repos.d/CentOS-Base.repo_bak
```

- 3. 下载腾讯云的yum配置文件

```
wget -O /etc/yum.repos.d/CentOS-Base.repo
http://mirrors.cloud.tencent.com/repo/centos7_base.repo
```

```
[root@localhost yum.repos.d]# wget -O /etc/yum.repos.d/CentOS-Base.repo http://mirrors.cloud.tencent.com/repo/centos7_base.repo
--2024-09-12 19:57:15-- http://mirrors.cloud.tencent.com/repo/centos7_base.repo
Resolving mirrors.cloud.tencent.com (mirrors.cloud.tencent.com)... 198.18.0.92
Connecting to mirrors.cloud.tencent.com (mirrors.cloud.tencent.com)|198.18.0.92|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1694 (1.7K) [application/octet-stream]
Saving to: '/etc/yum.repos.d/CentOS-Base.repo'

100%[=====>] 1,694 --.-K/s in 0s

2024-09-12 19:57:16 (437 MB/s) - '/etc/yum.repos.d/CentOS-Base.repo' saved [1694/1694]

[root@localhost yum.repos.d]# ls
CentOS-Base.repo      CentOS-CR.repo        CentOS-fasttrack.repo  CentOS-Sources.repo  CentOS-x86_64-kernel.repo
CentOS-Base.repo_bak  CentOS-Debuginfo.repo CentOS-Media.repo      CentOS-Vault.repo
[root@localhost yum.repos.d]# yum clean all
Loaded plugins: fastestmirror, langpacks
Repository cr is listed more than once in the configuration
Repository fasttrack is listed more than once in the configuration
```

- 4. 运行YUM源更新命令

```
yum clean all
```

```
yum makecache
```

```
[root@localhost yum.repos.d]# yum clean all
Loaded plugins: fastestmirror, langpacks
Repository cr is listed more than once in the configuration
Repository fasttrack is listed more than once in the configuration
Cleaning repos: extras os updates
Cleaning up list of fastest mirrors
[root@localhost yum.repos.d]# yum makecache
Loaded plugins: fastestmirror, langpacks
Repository cr is listed more than once in the configuration
Repository fasttrack is listed more than once in the configuration
Determining fastest mirrors
extras | 2.9 kB 00:00:00
os | 3.6 kB 00:00:00
updates | 2.9 kB 00:00:01
(1/10): extras/7/x86_64/primary_db | 253 kB 00:00:01
(2/10): extras/7/x86_64/filelists_db | 305 kB 00:00:02
(3/10): os/7/x86_64/group_gz | 153 kB 00:00:01
(4/10): extras/7/x86_64/other_db | 154 kB 00:00:01
(5/10): os/7/x86_64/primary_db | 6.1 MB 00:00:27
(6/10): os/7/x86_64/other_db | 2.6 MB 00:00:17
(7/10): os/7/x86_64/filelists_db | 7.2 MB 00:00:51
(8/10): updates/7/x86_64/filelists_db | 15 MB 00:00:41
(9/10): updates/7/x86_64/other_db | 1.6 MB 00:00:04
(10/10): updates/7/x86_64/primary_db | 27 MB 00:00:56
Metadata Cache Created
```

二·更换腾讯YUM源

1. 添加腾讯云docker-ce仓库地址

```
yum install -y yum-utils
```

```
yum-config-manager --add-repo https://mirrors.cloud.tencent.com/docker-ce/linux/centos/docker-ce.repo
```

```
[root@localhost yum.repos.d]# yum install -y yum-utils
Loaded plugins: fastestmirror, langpacks
Repository cr is listed more than once in the configuration
Repository fasttrack is listed more than once in the configuration
Loading mirror speeds from cached hostfile
Package yum-utils-1.1.31-54.el7_8.noarch already installed and latest version
Nothing to do
[root@localhost yum.repos.d]# yum-config-manager --add-repo https://mirrors.cloud.tencent.com/docker-ce/linux/centos/docker-ce.repo
Loaded plugins: fastestmirror, langpacks
Repository cr is listed more than once in the configuration
Repository fasttrack is listed more than once in the configuration
adding repo from: https://mirrors.cloud.tencent.com/docker-ce/linux/centos/docker-ce.repo
grabbing file https://mirrors.cloud.tencent.com/docker-ce/linux/centos/docker-ce.repo to /etc/yum.repos.d/docker-ce.repo
repo saved to /etc/yum.repos.d/docker-ce.repo
[root@localhost yum.repos.d]# yum install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
Loaded plugins: fastestmirror, langpacks
Repository cr is listed more than once in the configuration
Repository fasttrack is listed more than once in the configuration
Loading mirror speeds from cached hostfile
docker-ce-stable
(1/2): docker-ce-stable/7/x86_64/updateinfo | 3.5 kB 00:00:00
(2/2): docker-ce-stable/7/x86_64/primary_db | 55 B 00:00:01
Resolving Dependencies
--> Running transaction check
--> Package containerd.io.x86_64 0:1.6.33-3.1.el7 will be installed
--> Processing Dependency: container-selinux >= 2:2.74 for package: containerd.io-1.6.33-3.1.el7.x86_64
--> Package docker-buildx-plugin.x86_64 0:0.14.1-1.el7 will be installed
--> Package docker-ce.x86_64 3:26.1.4-1.el7 will be installed
--> Processing Dependency: docker-ce-rootless-extras for package: 3:docker-ce-26.1.4-1.el7.x86_64
--> Package docker-ce-cli.x86_64 1:26.1.4-1.el7 will be installed
--> Package docker-compose-plugin.x86_64 0:2.27.1-1.el7 will be installed
--> Running transaction check
--> Package container-selinux.noarch 2:2.119.2-1.911c772.el7_8 will be installed
--> Package docker-ce-rootless-extras.x86_64 0:26.1.4-1.el7 will be installed
--> Processing Dependency: fuse-overlayfs >= 0.7 for package: docker-ce-rootless-extras-26.1.4-1.el7.x86_64
--> Processing Dependency: slirp4netns >= 0.4 for package: docker-ce-rootless-extras-26.1.4-1.el7.x86_64
--> Running transaction check
--> Package fuse-overlayfs.x86_64 0:0.7.2-6.el7_8 will be installed
--> Processing Dependency: libfuse3.so.3(FUSE_3.2)(64bit) for package: fuse-overlayfs-0.7.2-6.el7_8.x86_64
--> Processing Dependency: libfuse3.so.3(FUSE_3.2)(64bit) for package: fuse-overlayfs-0.7.2-6.el7_8.x86_64
```

2. 安装docker

```
yum install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
```

```
[root@localhost yum.repos.d]# yum install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
Loaded plugins: fastestmirror, langpacks
Repository cr is listed more than once in the configuration
Repository fasttrack is listed more than once in the configuration
Loading mirror speeds from cached hostfile
docker-ce-stable
(1/2): docker-ce-stable/7/x86_64/updateinfo | 3.5 kB 00:00:00
(2/2): docker-ce-stable/7/x86_64/primary_db | 55 B 00:00:01
Resolving Dependencies
--> Running transaction check
--> Package containerd.io.x86_64 0:1.6.33-3.1.el7 will be installed
--> Processing Dependency: container-selinux >= 2:2.74 for package: containerd.io-1.6.33-3.1.el7.x86_64
--> Package docker-buildx-plugin.x86_64 0:0.14.1-1.el7 will be installed
--> Package docker-ce.x86_64 3:26.1.4-1.el7 will be installed
--> Processing Dependency: docker-ce-rootless-extras for package: 3:docker-ce-26.1.4-1.el7.x86_64
--> Package docker-ce-cli.x86_64 1:26.1.4-1.el7 will be installed
--> Package docker-compose-plugin.x86_64 0:2.27.1-1.el7 will be installed
--> Running transaction check
--> Package container-selinux.noarch 2:2.119.2-1.911c772.el7_8 will be installed
--> Package docker-ce-rootless-extras.x86_64 0:26.1.4-1.el7 will be installed
--> Processing Dependency: fuse-overlayfs >= 0.7 for package: docker-ce-rootless-extras-26.1.4-1.el7.x86_64
--> Processing Dependency: slirp4netns >= 0.4 for package: docker-ce-rootless-extras-26.1.4-1.el7.x86_64
--> Running transaction check
--> Package fuse-overlayfs.x86_64 0:0.7.2-6.el7_8 will be installed
--> Processing Dependency: libfuse3.so.3(FUSE_3.2)(64bit) for package: fuse-overlayfs-0.7.2-6.el7_8.x86_64
--> Processing Dependency: libfuse3.so.3(FUSE_3.2)(64bit) for package: fuse-overlayfs-0.7.2-6.el7_8.x86_64
```

3. 配置docker的国内源

```
vi /etc/docker/daemon.json
```

```
4. {
    "registry-mirrors": [
```

```

        "https://dockerpull.com",
        "https://docker.anyhub.us.kg",
        "https://dockerhub.jobcher.com",
        "https://dockerhub.icu",
        "https://docker.aws19527.cn"
    ]
}

```

5. 使docker的配置生效

```
systemctl daemon-reload
```

```
systemctl restart docker
```

![屏幕截图 2024-09-13 191256.png](attachment:7dfccd8320505d49aa5367d0d66d32eb)

三•安装MySQL

创建挂载目录

```
== mkdir -p /home/docker/mysql/{log,data,conf,mysql-files}==
```

执行命令安装MySQL

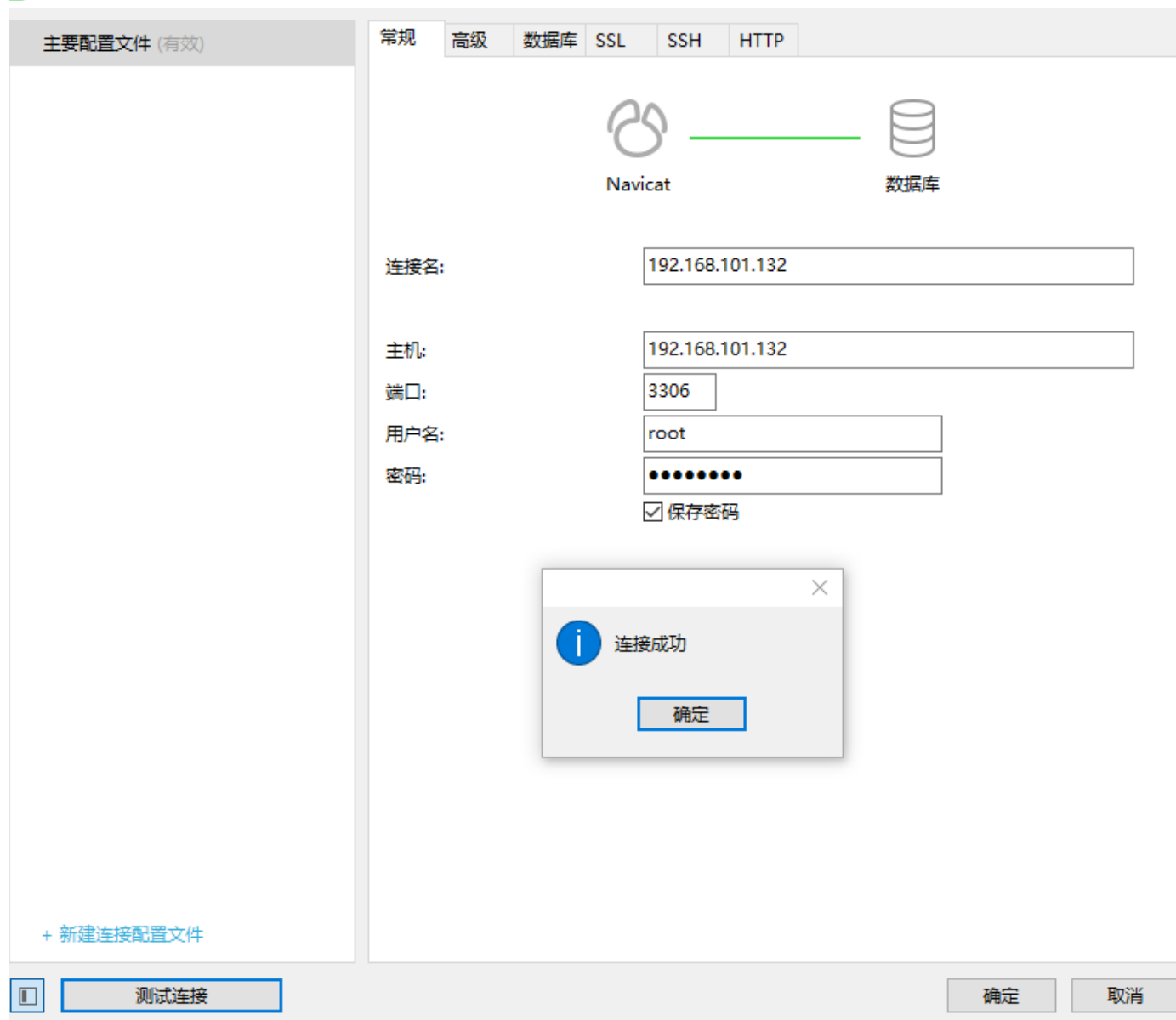
```
docker run -p 3306:3306 --name mysql -v /home/docker/mysql/log:/var/log/mysql -v
/home/docker/mysql/data:/var/lib/mysql -v /home/docker/mysql/conf:/etc/mysql -v
/home/docker/mysql/mysql-files:/var/lib/mysql-files -e MYSQL_ROOT_PASSWORD=root -d
mysql:8.0.24
```

```

[root@localhost yum.repos.d]# mkdir -p /home/docker/mysql/{log,data,conf,mysql-files}
[root@localhost yum.repos.d]# docker run -p 3306:3306 --name mysql -v /home/docker/mysql/log:/var/log/mysql -v /home/docker/m
ysql/data:/var/lib/mysql -v /home/docker/mysql/conf:/etc/mysql -v /home/docker/mysql/mysql-files:/var/lib/mysql-files -e MYSQL
L_ROOT_PASSWORD=root -d mysql:8.0.24
Unable to find image 'mysql:8.0.24' locally
8.0.24: Pulling from library/mysql
f7ec5a41d630: Pull complete
9444bb562699: Pull complete
6a4207b96940: Pull complete
181cefd361ce: Pull complete
8a2090759d8a: Pull complete
15f235e0d7ee: Pull complete
d870539cd9db: Pull complete
493aaa84617a: Pull complete
bfc0e534fc78: Pull complete
fae20d253f9d: Pull complete
9350664305b3: Pull complete
e47da95a5aab: Pull complete
Digest: sha256:04ee7141256e83797ea4a84a4d31b1f1bc10111c8d1bc1879d52729ccd19e20a
Status: Downloaded newer image for mysql:8.0.24
9be7fb931dcfc8cd6bb69bd9b29b7cf763e3460ac3d5d250f6255887a22dffff3
[root@localhost yum.repos.d]# mkdir -p /home/docker/redis/data

```

使用Navicat连接MySQL数据库，显示结果如下：



四•安装Redis

1. 创建挂载目录

```
mkdir -p /home/docker/redis/data
```

2. 执行命令安装Redis

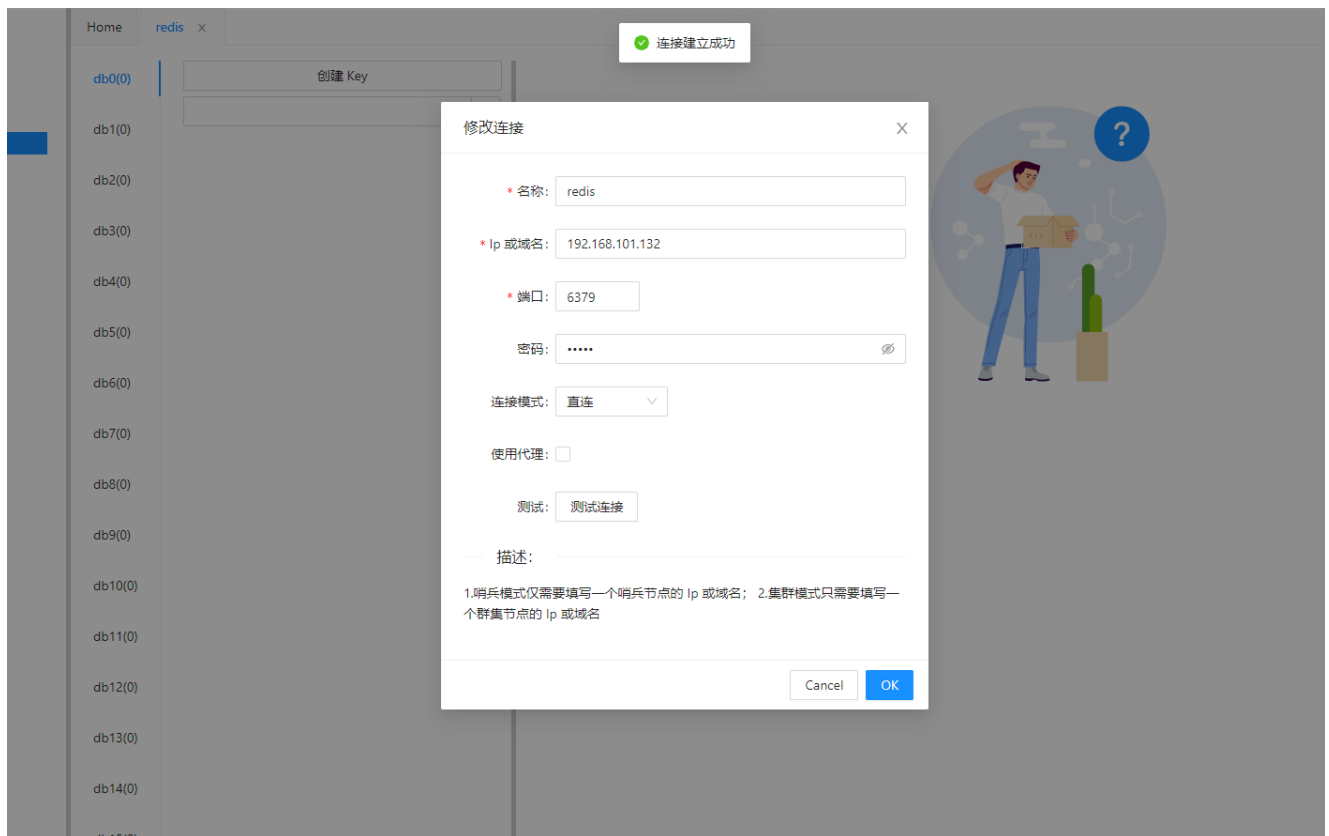
```
docker run -p 6379:6379 --name redis -v /home/docker/redis/data:/data -d redis redis-server --appendonly yes --requirepass redis
```

```

root@localhost yum.repos.d]# mkdir -p /home/docker/redis/data
root@localhost yum.repos.d]# docker run -p 6379:6379 --name redis -v /home/docker/redis/data:/data -d redis redis-server --a
pendonly yes --requirepass redis
Unable to find image 'redis:latest' locally
latest: Pulling from library/redis
2318d6c47ec: Pull complete
d7fd66f27f2: Pull complete
110a3d5b3155: Pull complete
312cf3f6b3e: Pull complete
39877ab23d0: Pull complete
1394ffc7248: Pull complete
f4fb700ef54: Pull complete
a03cb6163ab: Pull complete
Digest: sha256:eadf354977d428e347d93046bb1a5569d701e8deb68f090215534a99dbcb23b9
Status: Downloaded newer image for redis:latest
258cd9aafe14dd11c1e0aa3f102d9a8125157dd796bd11ee356e9097c17a7ece
root@localhost yum.repos.d]# mkdir -p /home/docker/nacos/data
root@localhost yum.repos.d]# mkdir -p /home/docker/nacos/conf
root@localhost yum.repos.d]# cd /home/docker/nacos/conf
root@localhost conf]# vim application.properties
root@localhost conf]# rm -rf application.properties
root@localhost conf]# vim application.properties
root@localhost conf]# docker run -d -p 8848:8848 -p 9848:9848 -p 9849:9849 -e MODE=standalone -e JVM_XMS=256m -e JVM_XMX=256
m -v /home/docker/nacos/conf/application.properties:/home/nacos/conf/application.properties --name nacos nacos/nacos-server
Unable to find image 'nacos/nacos-server:latest' locally
latest: Pulling from nacos/nacos-server
6a83fedfae6: Pull complete

```

使用QuickRedis来连接redis:



五·安装RabbitMQ

1. 执行命令安装RabbitMQ

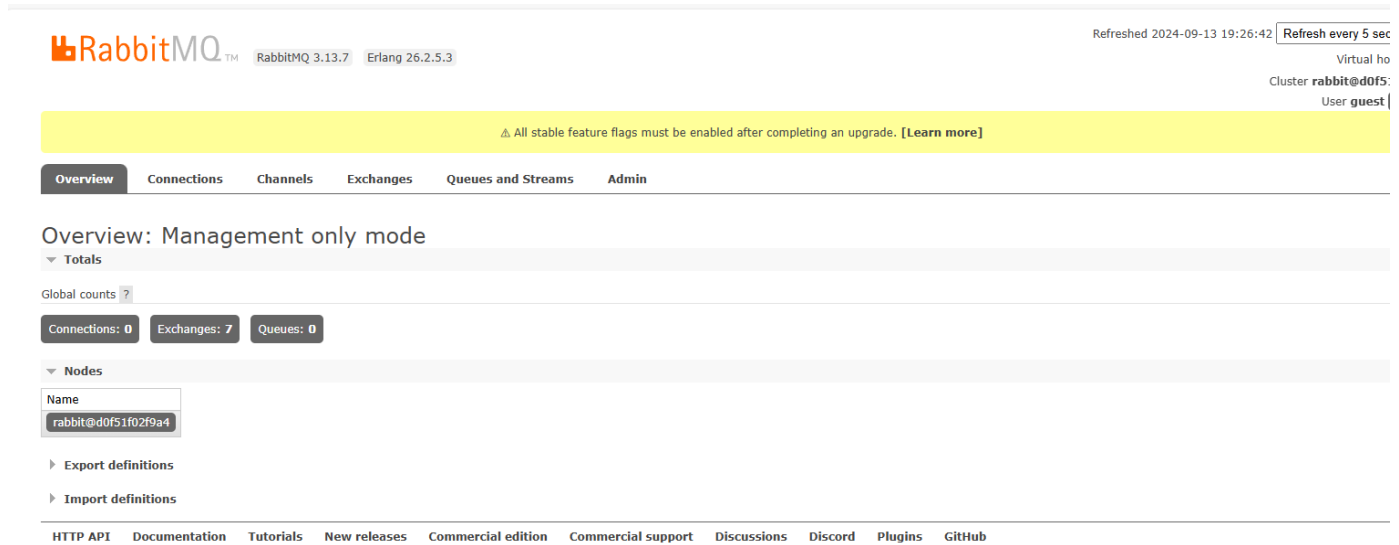
```
docker run -p 5672:5672 -p 15672:15672 --name rabbitmq -d rabbitmq
```

2. 进入容器并开启管理功能

```
docker exec -it rabbitmq /bin/bash
```

```
rabbitmq-plugins enable rabbitmq_management
```

输入: `http://你的虚拟机IP地址:15672/#/`



```
root@d0f51f02f9a4:~# docker exec -it rabbitmq /bin/bash
root@d0f51f02f9a4:~# rabbitmq-plugins enable rabbitmq_management
Enabling plugins on node rabbit@d0f51f02f9a4:
rabbitmq_management
The following plugins have been configured:
  rabbitmq_federation
  rabbitmq_management
  rabbitmq_management_agent
  rabbitmq_prometheus
  rabbitmq_web_dispatch
Applying plugin configuration to rabbit@d0f51f02f9a4...
The following plugins have been enabled:
  rabbitmq_management

started 1 plugins.
root@d0f51f02f9a4:~# ^C
root@d0f51f02f9a4:~# ^C
root@d0f51f02f9a4:~# ls
bin  dev  home  lib32  libx32  mnt  plugins  root  sbin  sys  usr
boot  etc  lib  lib64  media  opt  proc  run  srv  tmp  var
root@d0f51f02f9a4:~# docker ps
bash: docker: command not found
root@d0f51f02f9a4:~# docker ps
```

六•安装Nacos（在这里遇到太多坑花太多时间配置Nacos）

1. 首先创建mysql数据库: nacos（要载入数据库表什么表都可以）
2. 创建挂载目录 `mkdir -p /home/docker/nacos/conf`
3. 在 `/home/docker/nacos/conf` 目录下创建 `application.properties`, 并写入
4.

```
server.servlet.contextPath=/nacos
server.error.include-message=ON_PARAM
```

```
server.port=8848
spring.datasource.platform=mysql
db.num=1
db.url.0=jdbc:mysql://你的虚拟机ip地址:数据库端口/(数据库名)?
characterEncoding=utf8&connectTimeout=1000&socketTimeout=3000&autoReconnect=true&useUn
db.user.0=root
db.password.0=root
db.pool.config.connectionTimeout=30000
db.pool.config.validationTimeout=10000
db.pool.config.maximumPoolSize=20
db.pool.config.minimumIdle=2
```

5. 执行命令安装nacos（注意要先开启数据库Mysql，然后才能开启nacos（还有出错的话经常是无数据源这时候只能在先重启docker和mysql以及nacos））

```
docker run -d -p 8848:8848 -e MODE=standalone -v
/home/docker/nacos/conf/application.properties:/home/nacos/conf/application.properties -
-name nacos nacos/nacos-server
```

```
[root@localhost conf]# docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS
6658a4b8390a   nacos/nacos-server "sh bin/docker-start..." 7 hours ago   Up 5 minutes   0.0.0.0:8848->8848/tcp, :::8848->8848/tcp
d0f51f02f9a4   rabbitmq       "docker-entrypoint.s..." 7 hours ago   Up 7 hours     4369/tcp, 0.0.0.0:5672->5672/tcp, :::5672->5672/tcp, 5671/tcp, 156
91-15692/tcp, 25672/tcp, 0.0.0.0:15672->15672/tcp, :::15672->15672/tcp
258cd9aafef14   redis          "docker-entrypoint.s..." 7 hours ago   Up 7 hours     0.0.0.0:6379->6379/tcp, :::6379->6379/tcp
9be7fb931dcf   mysql:8.0.24   "docker-entrypoint.s..." 7 hours ago   Up 11 minutes   0.0.0.0:3306->3306/tcp, :::3306->3306/tcp, 33060/tcp
mysql
```

暂时完成后面会需要安装Minio和xxl-job等

七•安装XXL-JOB

跟上面的其他操作相似