

# 1 Docker 安装 Redis

```
docker search redis
```

```
docker pull redis:4.0
```

```
docker run -p 6379:6379 -v $PWD/data:/data -d redis:4.0 redis-server --appendonly yes
```

命令说明：

-p 6379:6379：将容器的 6379 端口映射到主机的 6379 端口

-v \$PWD/data:/data：将主机中当前目录下的 data 挂载到容器的/data

redis-server --appendonly yes：在容器执行 redis-server 启动命令，并打开 redis 持久化配置

进入 shell 测试

```
docker exec -it 6f6af6591948 redis-cli
```

```
127.0.0.1:6379> keys *
(empty list or set)
127.0.0.1:6379> set key1 value1
OK
127.0.0.1:6379> keys *
1) "key1"
127.0.0.1:6379> exit
```

# 2 Docker 安装 nginx

<http://www.runoob.com/docker/docker-install-nginx.html>

```
docker pull nginx
```

```
docker run -p 80:80 --name mynginx -v $PWD/www:/www -v $PWD/conf/nginx.conf:/etc/nginx/nginx.conf -v $PWD/logs:/wwwlogs -d nginx
```

```
docker run -p 80:80 --name mynginx -v $PWD/www:/www -v $PWD/conf:/etc/nginx -v $PWD/logs:/wwwlogs -d nginx
```

命令说明：

-p 80:80：将容器的 80 端口映射到主机的 80 端口

--name mynginx：将容器命名为 mynginx

-v \$PWD/www:/www：将主机中当前目录下的 www 挂载到容器的/www

-v \$PWD/conf/nginx.conf:/etc/nginx/nginx.conf：将主机中当前目录下的 nginx.conf 挂载到容器

的/etc/nginx/nginx.conf

-v \$PWD/logs:/wwwlogs: 将主机中当前目录下的 logs 挂载到容器的/wwwlogs

```
docker stop `docker ps -a | grep nginx | awk '{print $1}'`
```

```
docker rm `docker ps -a | grep nginx | awk '{print $1}'`
```

## 3 安装 nginx

- 下载:

<http://nginx.org/en/download.html>



- 解压

```
tar -zxf nginx-1.14.2.tar.gz -C /opt
```

- 装插件

```
yum install gcc-c++
```

```
yum install -y pcre pcre-devel
```

```
yum install -y zlib zlib-devel
```

```
yum install -y openssl openssl-devel
```

- Make

```
cd /opt/nginx-1.14.2/
```

```
./configure --prefix=/home/nginx
```

```
make install
```

- 启动和测试

```
Cd /home/nginx/sbin/
```

```
./nginx
```

```
ps -aux|grep nginx
```

```
curl -i localhost
```

在浏览器中输入: ip 即可

## Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to [nginx.org](http://nginx.org).  
Commercial support is available at [nginx.com](http://nginx.com).

*Thank you for using nginx.*

- 停止

```
./nginx -s quit
```

- 配置文件生效

```
./nginx -s reload
```

- 开机自启

```
vi /etc/rc.local
```

添加一行：

```
/home/nginx/sbin/nginx
```

## 4 Docker 安装 mysql

### 4.1 安装

```
cd /opt/
```

```
mkdir mysql_docker
```

```
cd /opt/mysql_docker/
```

```
docker pull mysql
```

```
docker run --name mysqlserver -v $PWD/conf:/etc/mysql/conf.d -v $PWD/logs:/logs -v $PWD/data:/var/lib/mysql -e MYSQL_ROOT_PASSWORD=123456 -d -i -p 3306:3306 mysql:latest
```

### 4.2 采用 docker 安装的目录问题

Mysql 的关键目录：

Basedir：程序安装目录，如 D:/program/mysql5.5/

Datadir：数据存放路径，如 C:\ProgramData\MySQL\MySQL Server 5.5\Data\

配置文件目录：D:\program\mysql5.5\my.ini

运行 mysql 镜像

参数说明: -P:暴露所有端口(映身出来的端口是随机的), -p:容器端口和机器端口映射(端口是确定的), 下方的-p 3306(宿主机端口):3306(容器端口);

```
docker run --name mysqlserver -v $PWD/conf:/etc/mysql/conf.d -v $PWD/logs:/logs -v $PWD/data:/var/lib/mysql -e MYSQL_ROOT_PASSWORD=123456 -d -i -p 3306:3306 mysql:5.7
```

- Datadir:

```
root@8855431faec6:/etc/mysql/conf.d# cd /var/lib/mysql
root@8855431faec6:/var/lib/mysql# ls
#innodb temp binlog.000002 .pem fastwave ib_logfile1 mysql private_key.pem server-key.pem
auto.cnf binlog.index client-cert.pem ib_buffer_pool ibdata1 mysql.ibd public_key.pem sys
binlog.000001 ca-key.pem client-key.pem ib_logfile0 ibtmp1 performance_schema server-cert.pem undo_001
root@8855431faec6:/var/lib/mysql#
```

- 配置文件目录:

如果想要自定义配置, 建议向 /etc/mysql/conf.d 目录中创建 .cnf 文件。新建的文件可以任意起名, 只要保证后缀名是 cnf 即可。新建的文件中的配置项可以覆盖 /etc/mysql/my.cnf 中的配置项。

- Docker 教程安装 mysql,对 dockerfile 也有介绍

<http://www.runoob.com/docker/docker-install-mysql.html>

## 4.3 采用 docker 安装如何进入到本地 mysql shell

进入到 mysql 容器内部

```
docker exec -it mysqlserver bash
```

```
mysql -uroot -p //输入密码
```

```
show databases; //然后就可以输入执行的命令了
```

## 4.4 问题和解决方法

Navicat 链接 mysql 显示 Clinet dose not support authentication protocol request by server ;consider upgrading MySQL client

解决方法:

```
docker exec -it mysqlserver bash
```

```
mysql -uroot -p //输入密码
```

```
show databases; //然后就可以输入执行的命令了
```

```
select host,user from user;
```

```
ALTER USER 'root'@'%' IDENTIFIED WITH mysql_native_password BY '123456';
```

```
alter user 'root'@'%' IDENTIFIED by 'password' PASSWORD EXPIRE NEVER;
```

```
FLUSH PRIVILEGES;
```

//8.0 以上版本可用:

```
alter user 'root'@'%' IDENTIFIED BY '123456';
```

## 5 使用 Docker 安装 Hbase

### 1. 拉取 Hbase 镜像

```
docker pull harisekhon/hbase
```

### 2. 安装镜像，将实例化为一个运行容器，即一组 Hbase 数据库进程

```
docker run -d -h myhbase -p 2181:2181 -p 8080:8080 -p 8085:8085 -p 9090:9090 -p 9095:9095 -p 16000:16000 -p 16010:16010 -p 16201:16201 -p 16301:16301 --name hbasedockerharisekhon/hbase
```

-d 表示后台

-h 定义容器 host

-p 表示端口映射

- name 表示容器别名（我这里命名 hbase1.3）

harisekhon/hbase 是 image 镜像

### 3. 查看效果

```
dockerps -a
```

如出现以下回显信息，则说明容器安装成功且能正常提供服务：

```
[root@master ~]# docker ps -a
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS
7034ddb38970       harisekhon/hbase   "/bin/sh -c \"entryp... 16 hours ago       Up 16 hours
0.0.0.0:2181->2181/tcp, 0.0.0.0:8080->8080/tcp, 0.0.0.0:8085->8085/tcp, 0.0.0.0:9090->9090/tcp, 0.0.0.0:9095->9095/tcp, 0.0.0.0:16000->16000/tcp, 0.0.0.0:16010->16010/tcp, 0.0.0.0:16201->16201/tcp, 0.0.0.0:16301->16301/tcp
hbase1.3
[root@master ~]#
```

还可以通过浏览器查看，输入地址：<http://{虚拟机或物理机 IP}:16010/master-status>  
效果如下：

←

→

🔄

🔒 不安全 | 10.101.43.196:16010/master-status

APACHE

HBASE

Home

Table Details

Procedures & Locks

Process Metrics

Local Logs

Log Level

Debug Dump

Metrics Dump

HBase Configuration

Master

myhbase

Region Servers

Base Stats

Memory

Requests

Storefiles

Compactions

Replications

ServerName	Start time	Last contact	Version
myhbase,16020,1542221500821	Wed Nov 14 18:51:40 GMT 2018	0 s	2.1.0
Total: 1			

Backup Masters

ServerName	Port	Start Time
Total: 0		

Tables

User Tables

System Tables

Snapshots

已测试 hbase shell OK

```
docker pull dajobe/hbase
```

```
docker run -d -h testhbase -p 2181:2181 -p 8080:8080 -p 8085:8085 -p 9090:9090 -p 9095:9095 -p 16000:16000 -p 16010:16010 -p 16201:16201 -p 16301:16301 --name hbasedocker dajobe/hbase
```

```
docker exec -it b7d285b80ed6 /bin/bash
```

```
hbase shell
```