一.环境

```
centos7:
192.168.238.160(xmjmaster)
192.168.238.161(xmjslave1)
```

二.docker安装

1.卸载历史版本

```
#查看安装
yum list installed | grep docker
#卸载
yum -y remove containerd.io.x86_64
yum -y remove docker-ce.x86_64
yum -y remove docker-ce-cli.x86_64
#删库
rm -rf /var/lib/docker
```

2.安装官方yum源

```
yum install -y yum-utils
yum-config-manager --add-repo https://download.docker.com/linux/centos/docker-ce.repo

cd /etc/yum.repos.d
ll
```

3.安装Docker引擎

```
yum install -y docker-ce docker-ce-cli containerd.io
```

4.启动docker

```
#开机启动
systemctl enable docker
#启动
systemctl start docker
#查看Docker状态
docker info
```

三.Swarm集群安装

1.下载镜像

```
# 拉取镜像
docker pull swarm
```

2.创建新集群

```
[root@xmjmaster yum.repos.d]# docker swarm init --advertise-addr 192.168.238.160
```

```
[root@xmjmaster yum.repos.d]# docker swarm init --advertise-addr 192.168.238.160
Swarm initialized: current node (55xg05fr3lwgmlohwbnbyivk3) is now a manager.

To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-2trra1xu8vdr5crmfojr5ya3o6hs7me7nidpidubyyyip9z66t-apjvo86nyilchgmr4x59j99kb 192.168.238.160:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

[root@xmjmaster yum.repos.d]#
```

3.添加工作节点到集群

```
[root@xmjslave1 yum.repos.d]# docker swarm join --token SWMTKN-1-2trra1xu8vdr5crmfojr5ya3o6hs7me7nidpidubyyyip9z66t-apjvo86nyilchgmr4x59j99kb 192.168.238.160:2377
```

```
[root@xmjslave1 yum.repos.d]# docker swarm join --token SWMTKN-1-2trra1xu8vdr5crmfojr5ya3o6hs7me7nidpidubyyyip9z66t-apjvo86nyilchgmr4x59j99kb 192.168.238.160:2377
This node joined a swarm as a worker.
[root@xmjslave1 yum.repos.d]#
```

4.查看集群状态和节点信息

```
#查看集群状态
docker info
#查看节点信息
docker node ls
```

```
[root@xmjmaster yum.repos.d]# docker node ls
ID                            HOSTNAME            STATUS              AVAILABILITY        MANAGER STATUS      ENGINE VERSION
55xg05fr3lwgmlohwbnbyivk3 *   xmjmaster           Ready               Active              Leader              19.03.13
5f04ywzzgh5g1pthh66exp951     xmjslave1           Ready               Active                                  19.03.13
[root@xmjmaster yum.repos.d]#
```

四.Compose的安装

1.下载Docker Compose的当前稳定版本

```
curl -L "https://github.com/docker/compose/releases/download/1.26.0/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
```

2.将可执行权限应用于二进制文件

```
chmod +x /usr/local/bin/docker-compose
```

3.添加到环境中

```
#ln -s :  软链接
ln -s /usr/local/bin/docker-compose /usr/bin/docker-compose
```

4.测试安装

```
docker-compose --version
```

```
[root@xmjmaster compose]# docker-compose --version
docker-compose version 1.26.0, build d4451659
[root@xmjmaster compose]#
```

五.swarm和compose整合

1.编写docker-compose.yml

```yaml
version: "3.0"
services:
  nginx:
    image: nginx:1.18.0
    environment:
      - TZ=Asia/beijing
    ports:
      - 80:80
      - 443:443
    volumes:
      - /docker/nginx/log:/var/log/nginx
      - /docker/nginx/www:/etc/nginx/html
      - /etc/letsencrypt:/etc/letsencrypt
    deploy:
      mode: replicated
      replicas: 2
  mysql:
    image: mysql:5.7.30
    ports:
      - 13306:3306
    command:
      --default-authentication-plugin=mysql_native_password
      --character-set-server=utf8mb4
      --collation-server=utf8mb4_general_ci
      --explicit_defaults_for_timestamp=true
      --lower_case_table_names=1
      --default-time-zone=+8:00
    environment:
      MYSQL_ROOT_PASSWORD: "123456"
    volumes:
       - "/docker/mysql/db:/var/lib/mysql"
    deploy:
      mode: replicated
      replicas: 2
  redis:
    image: redis:5.0.9
    environment:
      - TZ=Asia/beijing
    ports:
      - 6379:6379
    volumes:
      - /docker/redis/data:/data
    deploy:
      mode: replicated
      replicas: 2
```

```bash
mkdir -p /docker/nginx/log
mkdir -p /docker/nginx/www
mkdir -p /etc/letsencrypt
mkdir -p /docker/mysql/db
mkdir -p /docker/redis/data
```

2.运行

```bash
docker stack deploy -c docker-compose.yml web
```

3.查看服务

```
docker stack services web
```

```
[root@xmjmaster compose]# docker stack services web
ID                  NAME                MODE                REPLICAS            IMAGE               PORTS
jvbj991powws        web_redis           replicated          2/2                 redis:5.0.9         *:6379->6379/tcp
rx4kqqr7jgld        web_mysql           replicated          2/2                 mysql:5.7.30        *:13306->3306/tcp
xfbvl3v2isto        web_nginx           replicated          2/2                 nginx:1.18.0        *:80->80/tcp, *:443->443/tcp
[root@xmjmaster compose]#
```

五.利用dockerfile将Hot.jar构建成镜像lgedu/hot:1.0

1.编写DockerFile

```
vim DockerFile-hot

FROM java:openjdk-8-alpine
ADD hot-0.0.1-SNAPSHOT.jar hot.jar
EXPOSE 8080
ENTRYPOINT ["java","-jar","/hot.jar"]
```

2.创建镜像/生成镜像的标签信息

```
docker build -f /usr/local/dockerfiles/DockerFile-hot -t lgedu/hot:1.0 .
```

```
[root@xmjmaster dockerfiles]# docker build -f /usr/local/dockerfiles/DockerFile-hot -t lgedu/hot:1.0 .
Sending build context to Docker daemon  16.55MB
Step 1/4 : FROM java:openjdk-8-alpine
 ---> 3fd9dd82815c
Step 2/4 : ADD hot-0.0.1-SNAPSHOT.jar hot.jar
 ---> b9e1e6bf88fe
Step 3/4 : EXPOSE 8080
 ---> Running in bcc850e87fa3
Removing intermediate container bcc850e87fa3
 ---> 8dedb635035d
Step 4/4 : ENTRYPOINT ["java","-jar","/hot.jar"]
 ---> Running in e01f45953804
Removing intermediate container e01f45953804
 ---> 7cc224a5111f
Successfully built 7cc224a5111f
Successfully tagged lgedu/hot:1.0
[root@xmjmaster dockerfiles]# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED             SIZE
lgedu/hot           1.0                 7cc224a5111f        4 seconds ago       162MB
nginx               1.18.0              20ade3ba43bf        2 days ago          133MB
swarm               latest              1a5eb59a410f        3 weeks ago         12.7MB
redis               5.0.9               226c4c322bdb        3 weeks ago         98.3MB
mysql               5.7.30              9cfcce23593a        4 months ago        448MB
java                openjdk-8-alpine    3fd9dd82815c        3 years ago         145MB
[root@xmjmaster dockerfiles]#
```

3.修改docker-compose.yml

```
##追加
  hot:
    image: lgedu/hot:1.0
    ports:
      - 8080:8080
    deploy:
      mode: replicated
      replicas: 2
```

4.将镜像导出上传到xmjslave1,并在xmjslave1导入hot镜像

```
docker save lgedu/hot > /usr/local/dockerfiles/hot.tar
scp hot.tar xmjslave1:/usr/local/dockerfiles/
docker load -i hot.tar
```

5.运行

```
docker stack deploy -c docker-compose.yml web
```

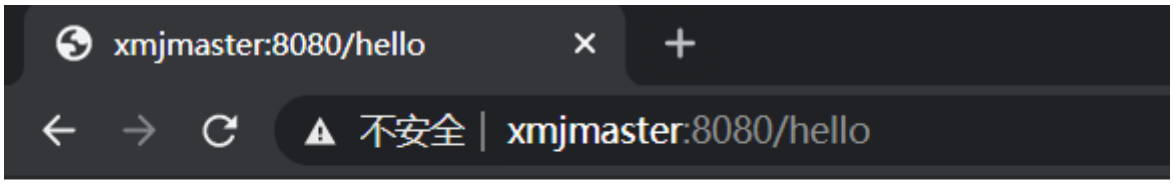
6.查看服务

```
docker stack services web
```

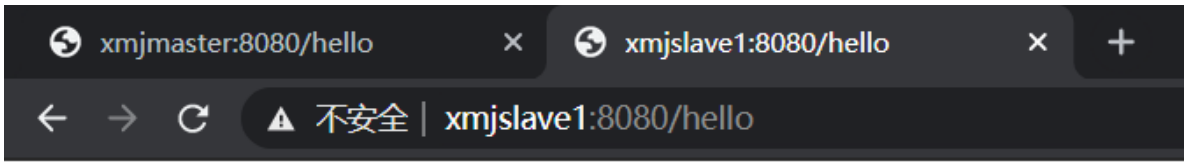
```
[root@xmjmaster compose]# docker stack deploy -c docker-compose.yml web
Creating network web_default
Creating service web_redis
Creating service web_hot
Creating service web_nginx
Creating service web_mysql
[root@xmjmaster compose]# docker stack services web
ID                  NAME                MODE                REPLICAS            IMAGE               PORTS
e40k1fhz6qtz        web_redis           replicated          2/2                 redis:5.0.9         *:6379->6379/tcp
seoje8367c1f        web_hot             replicated          2/2                 lgedu/hot:1.0       *:8080->8080/tcp
sum2742nnre4        web_mysql           replicated          2/2                 mysql:5.7.30        *:13306->3306/tcp
tfrsbh0hulqi        web_nginx           replicated          2/2                 nginx:1.18.0        *:80->80/tcp, *:443->443/tcp
```

7.浏览器访问hot

```
http://xmjmaster:8080/hello
http://xmjslave1:8080/hello
```



hello !!!



hello !!!