1. DOCKER安装

1、开启网络转发：

# echo 1 >/proc/sys/net/ipv4/ip\_forward

2、安装docker-ce：

wget -O /etc/yum.repos.d/docker-ce.repo https://download.docker.com/linux/centos/docker-ce.repo

或：

echo '''[docker-ce-stable]

name=Docker CE Stable - $basearch

baseurl=https://download.docker.com/linux/centos/7/$basearch/stable

enabled=1

gpgcheck=1

gpgkey=https://download.docker.com/linux/centos/gpg''' > /etc/yum.repos.d/docker-ce.repo

cat docker-ce.repo

yum makecache

yum list | grep docker

yum install -y yum-utils device-mapper-persistent-data lvm2

yum install docker-ce -y

或使用官方安装脚本安装Docker：

$ curl -fsSL https://get.docker.com/ | sh

3、启动docker引擎：

# chkconfig docker on && service docker start

4、基本信息查看:

[root@localhost /]# docker version

#查看docker的版本号，包括客户端、服务端、依赖的Go等

[root@localhost /]# docker info

#查看系统(docker)层面信息，包括管理的images, containers数等。

5、修改容器运行目录:

mkdir /etc/systemd/system/docker.service.d

cat > /etc/systemd/system/docker.service.d/docker.conf <<EOF

[Service]

ExecStart=

ExecStart=/usr/bin/dockerd -H unix:///var/run/docker.sock -H tcp://0.0.0.0:2375 --graph=/data/docker --storage-driver=overlay2

EOF

6、配置国内镜像

cat > /etc/docker/daemon.json <<EOF

{

"registry-mirrors": ["https://62cvzbat.mirror.aliyuncs.com"],

"registry-mirrors":["https://registry.docker-cn.com"],

"insecure-registries":["registry:5000"],

"log-driver":"json-file",

"log-opts":{"max-size":"1024m","max-file":"2"}

}

EOF

7、重新加载配置：

systemctl daemon-reload && systemctl restart docker

8、防火墙放行端口：

firewall-cmd --add-port=2375/tcp --permanent;firewall-cmd --add-port=2375/udp --permanent;

firewall-cmd --add-port=2376/tcp --permanent;firewall-cmd --add-port=2376/udp --permanent;

firewall-cmd --reload;iptables -S

1. docker镜像

搭建私有仓库：

1、下载registry镜像：

# docker pull registry #默认从中国镜像仓库中拉取 <https://registry.docker-cn.com>

2、查看下载到本机的镜像：

# docker images

3、运行registry：

启动本地镜像源：

# docker run -d -p 5000:5000 --name=registry --restart=always --privileged=true --log-driver=none -v /data/docker/registrydata:/var/lib/registry registry

-d后台运行；-p指定端口；-v把registry的镜像路径/var/lib/registry映射到本机的/data/registry

3、防火墙放行5000端口：

firewall-cmd --add-port=5000/tcp --permanent;firewall-cmd --add-port=5000/udp --permanent;firewall-cmd --reload;iptables -S

添加hosts名称解析：

echo "172.0.0.1 registry" >> /etc/hosts

检查5000端口：

# netstat -an | grep 5000

tcp6 0 0 :::5000 :::\* LISTEN

# telnet 127.0.0.1 5000

Trying 127.0.0.1...

Connected to 127.0.0.1.

Escape character is '^]'.

telnet 127.0.0.1 5000成功。

echo "39.98.199.122 registry" >> /etc/hosts 临时修改hosts 从远程仓库下载镜像

4、下载镜像：

# docker pull 仓库地址:端口/名称空间/镜像名称:版本

docker pull registry:5000/g1/portainer:v2

docker pull registry:5000/g1/php:v2

docker pull registry:5000/g1/es:v2

docker pull registry:5000/g1/redis:v2

docker pull registry:5000/g1/nginx:v2

docker pull registry:5000/g1/tomcat:v3

docker pull registry:5000/g1/mysql:v2

docker pull registry:5000/g1/jdk:v2

docker pull registry:5000/g1/samba:v2

下载完成把hosts改回127.0.0.1

5、上传镜像：（把下载完成的镜像上传到本地仓库）

# docker push 仓库地址:端口/名称空间/镜像名称:版本

docker push registry:5000/g1/portainer:v2

docker push registry:5000/g1/php:v2

docker push registry:5000/g1/es:v2

docker push registry:5000/g1/redis:v2

docker push registry:5000/g1/nginx:v2

docker push registry:5000/g1/tomcat:v3

docker push registry:5000/g1/mysql:v2

docker push registry:5000/g1/jdk:v2

docker push registry:5000/g1/samba:v2

远程仓库下载镜像的速度太慢，部署安装时可以用硬盘把镜像传输到客户机器。此时步骤4和步骤5调换。即先把镜像上传到本地仓库，再从本地仓库下载。

通过restful接口查看仓库中的镜像：

# curl http://registry:5000/v2/\_catalog

{"repositories":["g1/es","g1/mysql","g1/nginx","g1/php","g1/redis","g1/solr","g1/tomcat"]}

获取镜像标签信息：

# curl http://registry:5000/v2/g1/mysql/tags/list

{"name":"g1/mysql","tags":["v2"]}

1. 镜像实例化：
2. 创建程序和数据目录：/data目录下存放配置文件和数据，/app目录下存放配置文件和程序：

（1）创建空目录

```

mkdir /data/zhxg/{app/{nginx,php,tomcat},data/{redis,solr,elasticsearch/{data,config,logs},mysql3306/{data,log},mysql3307/{data,log},mysql3308/{data,log}}} -p

touch /data/zhxg/data/mysql3306/log/error.log

```

（2）远程拷

scp -rp root@192.168.20.1:/data/zhxg /data/zhxg

（3）传输太慢，用磁盘拷

/data/zhxg

1. 实例化

\* 配置库

docker stop mysql3306 && docker rm mysql3306

docker run -idt --name mysql3306 -m 32g --cpus 8 -p 3306:3306 -v /data/zhxg/data/mysql3306/my.cnf:/etc/my.cnf -v /data/zhxg/data/mysql3306:/data/mysql registry:5000/g1/mysql:v2

~~\* 信息库~~

~~docker stop mysql3307 && docker rm mysql3307~~

~~docker run -idt --name mysql3307 -m 4g --cpus 2 -p 3307:3306 -v /data/zhxg/data/mysql3307/my.cnf:/etc/my.cnf -v /data/zhxg/data/mysql3307:/data/mysql registry:5000/g1/mysql:v2~~

~~\* 个人库~~

~~docker stop mysql3308 && docker rm mysql3308~~

~~docker run -idt --name mysql3308 -m 4g --cpus 2 -p 3308:3306 -v /data/zhxg/data/mysql3308/my.cnf:/etc/my.cnf -v /data/zhxg/data/mysql3308:/data/mysql registry:5000/g1/mysql:v2~~

\* redis

```

docker stop redis && docker rm redis

docker run -idt --name redis --privileged -m 8g --cpus 2 -p 6379:6379 -v /data/zhxg/data/redis:/data/redis registry:5000/g1/redis:v2

```

\* es

```

docker stop es && docker rm es

docker run -idt --name es --privileged -m 16g --cpus 8 -p 9200:9200 -p 9300:9300 -v /data/zhxg/data/elasticsearch:/data/elasticsearch -v /data/zhxg/data/elasticsearch/config:/usr/local/elasticsearch/config registry:5000/g1/es:v2

```

~~\* solr~~

~~```~~

~~docker stop solr && docker rm solr~~

~~docker run -idt --name solr -m 6g --cpus 2 -p 8983:8983 -v /data/zhxg/data/solr:/data/solr registry:5000/g1/solr:v2~~

~~```~~

\* tomcat：

```

docker stop tomcat && docker rm tomcat

docker run -idt --name tomcat --add-host api1.zhxg.local:172.16.104.68 \

--add-host api2.zhxg.local:172.16.104.68 \

--add-host web1.zhxg.local:172.16.104.68 \

--add-host base.mysql.zhxg.local:172.16.104.68 \

--add-host info.mysql.zhxg.local:172.16.104.68 \

--add-host user1.mysql.zhxg.local:172.16.104.68 \

--add-host solr.zhxg.local:172.16.104.68 \

--add-host image.zhxg.local:172.16.104.68 \

--add-host push.zhxg.local:172.16.104.68 \

--add-host words.sync.istarshine.com:172.16.104.68 \

--add-host userinfo-syncl.istarshine.com:172.16.104.68 \

--add-host redis.zhxg.local:172.16.104.68 \

--add-host files.zhxg.local:172.16.104.68 \

--add-host samba.zhxg.local:172.16.104.68 \

--add-host es.zhxg.local:172.16.104.68 -m 4g --cpus 2 -p 8080:8080 -p 8899:8899 \

-v /data/zhxg/app/tomcat:/usr/local/tomcat/webapps -v /data/zhxg/app/scheduledTask:/home/scheduledTask \

registry:5000/g1/tomcat:v3

```

实例化tomcat镜像后

进入tomcat镜像

docker exec -it tomcat /bin/bash

运行日报程序

nohup java -server -Xms1024m -Xmx2048m -XX:+UseParallelGC -XX:+AggressiveOpts -XX:+UseFastAccessorMethods -Dfile.encoding=utf-8 -jar -Dlog.path=/home/logs/ /home/scheduledTask/scheduledTask.jar --spring.profiles.active=ytj start 5 >/dev/null 2>&1 &

\* php 秘书

```

docker stop php && docker rm php

docker run -idt --name php --add-host api1.zhxg.local:172.16.104.68 \

--add-host api2.zhxg.local:172.16.104.68 \

--add-host web1.zhxg.local:172.16.104.68 \

--add-host base.mysql.zhxg.local:172.16.104.68 \

--add-host info.mysql.zhxg.local:172.16.104.68 \

--add-host user1.mysql.zhxg.local:172.16.104.68 \

--add-host solr.zhxg.local:172.16.104.68 \

--add-host image.zhxg.local:172.16.104.68 \

--add-host push.zhxg.local:172.16.104.68 \

--add-host words.sync.istarshine.com:172.16.104.68 \

--add-host userinfo-syncl.istarshine.com:172.16.104.68 \

--add-host redis.zhxg.local:172.16.104.68 \

--add-host files.zhxg.local:172.16.104.68 \

--add-host samba.zhxg.local:172.16.104.68 \

--add-host es.zhxg.local:172.16.104.68 -m 2g --cpus 2 -p 80:80 -v /data/zhxg/app/php/html:/var/www/html registry:5000/g1/php:v2

```

\* php 后台

```

docker stop php1 && docker rm php1

docker run -idt --name php1 --add-host api1.zhxg.local:172.16.104.68 \

--add-host api2.zhxg.local:172.16.104.68 \

--add-host web1.zhxg.local:172.16.104.68 \

--add-host base.mysql.zhxg.local:172.16.104.68 \

--add-host info.mysql.zhxg.local:172.16.104.68 \

--add-host user1.mysql.zhxg.local:172.16.104.68 \

--add-host solr.zhxg.local:172.16.104.68 \

--add-host image.zhxg.local:172.16.104.68 \

--add-host push.zhxg.local:172.16.104.68 \

--add-host words.sync.istarshine.com:172.16.104.68 \

--add-host userinfo-syncl.istarshine.com:172.16.104.68 \

--add-host redis.zhxg.local:172.16.104.68 \

--add-host files.zhxg.local:172.16.104.68 \

--add-host samba.zhxg.local:172.16.104.68 \

--add-host es.zhxg.local:172.16.104.68 -m 2g --cpus 2 -p 82:80 -v /data/zhxg/app/php1/html:/var/www/html registry:5000/g1/php:v2

```

\* nginx

```

docker stop nginx && docker rm nginx

docker run -idt --name nginx -m 4g --cpus 2 -p 8081:80 -v /data/zhxg/app/nginx:/etc/nginx/conf.d -v /data/zhxg/data/smbfiles:/data/smbfiles registry:5000/g1/nginx:v2

```

Samba镜像

docker stop samba && docker rm samba

docker run -idt --name samba --cpus 4 -m 8g -p 139:139 -p 445:445 -v /data/zhxg/data/smbfiles:/data/smbfiles registry:5000/g1/samba:v2

\* jdk (jar 运行环境)

```

docker stop jdk && docker rm jdk

docker run -idt --name jdk --add-host api1.zhxg.local:172.16.104.68 --add-host api2.zhxg.local:172.16.104.68 --add-host web1.zhxg.local:172.16.104.68 --add-host base.mysql.zhxg.local:172.16.104.68 --add-host info.mysql.zhxg.local:172.16.104.68 --add-host user1.mysql.zhxg.local:172.16.104.68 --add-host solr.zhxg.local:172.16.104.68 --add-host image.zhxg.local:172.16.104.68 --add-host push.zhxg.local:172.16.104.68 --add-host words.sync.istarshine.com:172.16.104.68 --add-host userinfo-syncl.istarshine.com:172.16.104.68 --add-host redis.zhxg.local:172.16.104.68 --add-host files.zhxg.local:172.16.104.68 --add-host samba.zhxg.local:172.16.104.68 --add-host es.zhxg.local:172.16.104.68 -m 32g --cpus 4 -p 10000:10000 -v /data/zhxg/app/jars:/home/jars registry:5000/g1/jdk:v2

```

## 容器操作：

```

启动已存在的容器：docker start mysql

停止已存在的容器：docker stop mysql

删除已存在的容器：docker rm mysql

进入容器：# docker exec -it 容器名 /bin/bash

退出容器：组合键 CTRL +pq

```

## 防火墙放行端口：

```

firewall-cmd --add-port=9000/tcp --permanent;firewall-cmd --add-port=9000/udp --permanent; \

firewall-cmd --add-port=6379/tcp --permanent;firewall-cmd --add-port=6379/udp --permanent; \

firewall-cmd --add-port=9200/tcp --permanent;firewall-cmd --add-port=9200/udp --permanent; \

firewall-cmd --add-port=3307/tcp --permanent;firewall-cmd --add-port=3307/udp --permanent; \

firewall-cmd --add-port=3308/tcp --permanent;firewall-cmd --add-port=3308/udp --permanent; \

firewall-cmd --add-port=3306/tcp --permanent;firewall-cmd --add-port=3306/udp --permanent; \

firewall-cmd --add-port=8080/tcp --permanent;firewall-cmd --add-port=8080/udp --permanent; \

firewall-cmd --add-port=80/tcp --permanent;firewall-cmd --add-port=80/udp --permanent;\

firewall-cmd --add-port=139/tcp --permanent;firewall-cmd --add-port=139/udp --permanent; \

firewall-cmd --add-port=445/tcp --permanent;firewall-cmd --add-port=445/udp --permanent;\

firewall-cmd --add-port=8899/tcp --permanent;firewall-cmd --add-port=8899/udp --permanent; \

firewall-cmd --add-port=10000/tcp --permanent;firewall-cmd --add-port=10000/udp --permanent;\

firewall-cmd --reload;iptables -S

JAVA服务部署。

1、进入tomcat docker exec

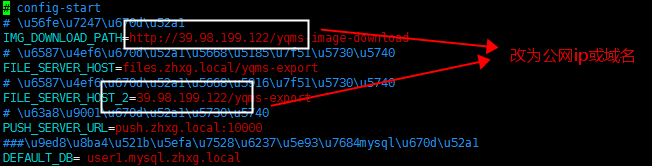
2、杀掉tomcat进程

3、删除ROOT.war和yqms.war

4、cd /usr/local/tomcat/webapps/ROOT/WEB-INF/classes

vim config.properties

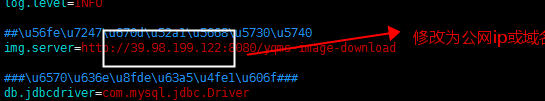
修改配置如图

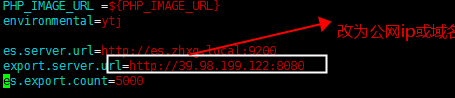


修改另一个工程的配置

cd /usr/local/tomcat/webapps/yqms/WEB-INF/classes

vim application.properties

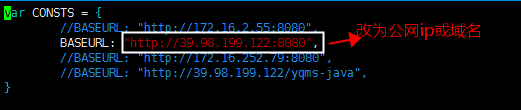




修改完成之后启动tomcat

PHP及前端配置修改

cd /var/www/html/Public/yqcn/js



日报服务配置修改

cd /data/zhxg/app/scheduledTask/

vim application.properties

将file.server=http://172.16.104.68/download/

修改为公网域名