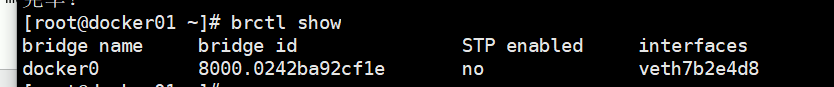
Docker 网络

Docker容器之间默认使用桥接网络



yum install bridge-utils



交换机上的网卡，就是nginx在docker0上的网卡

查看nginx的网络

docker inspect b20c602574de

因为docker0也是物理机上的一张网卡，所以物理机也可以通过docker0访问nginx容器ip地址

curl 172.17.0.2

由于docker容器之间默认使用桥接网络，所以两台物理机之间的docker容器是不能直接通信的，想要通信可以通过1、端口转发，将容器端口映射到物理机端口。

2、还可以使用overlay网络实现通信

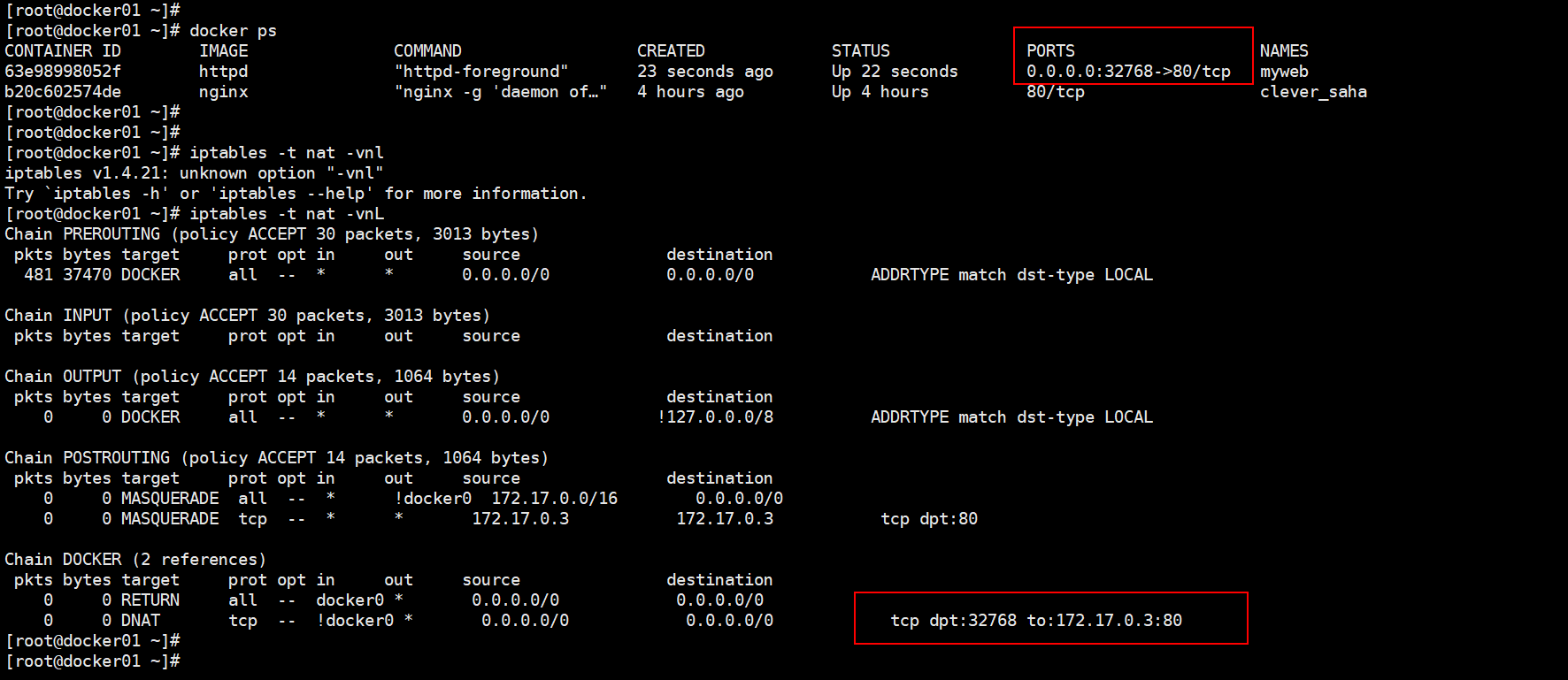
Host网络表示容器使用物理机的网络命名空间

None网络表示没有网络，用于运行一些不需要网络的任务，比如说批处理数据



通过iptables转发建立本地端口和容器的映射关系

docker run --name myweb --rm -p 80 httpd



同一物理机有多个docker network（交换机），network之间的的容器可以通过开启物理机的网络转发、修改iptables规则实现通信。当然，既然已经将容器放到不同的交换机下，那么就意味着这容器就不需要通信了

存储卷 volume

docker run --name b2 -it -v /data busybox

容器中有/data目录

docker container inspect b2

"Source": "/var/lib/docker/volumes/70836ea4485b4e302f6876d43d99783a362ac27f3e11e170772e535cfcd4bba7/\_data",

"Destination": "/data",

宿主机的Source目录和容器的Destination目录建立了映射关系

修改Source、Destination会相互影响

指定本机/data/volumes/b2目录映射到容器/data busybox

docker run --name b2 -it --rm -v /data/volumes/b2:/data busybox

docker run --name infracon -it -v /data/infracon/volume/:/data/web/html busybox

nginx（busybox）使用container:infracon的网络，存储卷跟infracon相同

docker run --name nginx --network container:infracon --volumes-from infracon -it busybox

Dockerfile

Vim Dockerfile

FROM nginx:1.14-alpine

LABEL maintainer="MageEdu"

ENV NGX\_DOC\_ROOT='/data/web/html/'

ADD index.html ${NGX\_DOC\_ROOT}

ADD entrypoint.sh /bin/

CMD ["/usr/sbin/nginx","-g","daemon off;"]

ENTRYPOINT ["/bin/entrypoint.sh"]

Vim entrypoint.sh

#!/bin/sh

#

cat > /etc/nginx/conf.d/www.conf << EOF

server {

server\_name ${HOSTNAME};

listen ${IP:-0.0.0.0}:${PORT:-80};

root ${NGX\_DOC\_ROOT:-/usr/share/nginx/html};

}

EOF

exec "$@"

chmod +x entrypoint.sh

vim index.html

<p>hello dockerfile</p>

构建

docker build -t test01 .

运行

docker run --name test01 --rm -P test01

传入PORT环境变量修改端口

docker run --name test01 --rm -P -e "PORT=8080" test01

端口改成了nginx的另一个端口改成了8080