[Edit](http://maxiang.info/#/?provider=evernote&guid=cfcc3023-24a6-4ef5-af12-5c1f2b1359f4&notebook=%E4%BA%91%E8%AE%A1%E7%AE%97)

# docker 初探

云计算 docker

* [docker 初探](evernotecid://C2FCC64E-8AFB-4E5E-8FCB-E231245AC2D2/appyinxiangcom/4048852/ENNote/p333)
  + [docker 应用场景](evernotecid://C2FCC64E-8AFB-4E5E-8FCB-E231245AC2D2/appyinxiangcom/4048852/ENNote/p333)
  + [镜像](evernotecid://C2FCC64E-8AFB-4E5E-8FCB-E231245AC2D2/appyinxiangcom/4048852/ENNote/p333)
    - [创建容器及启动容器](evernotecid://C2FCC64E-8AFB-4E5E-8FCB-E231245AC2D2/appyinxiangcom/4048852/ENNote/p333)
    - [docker 容器网络设置](evernotecid://C2FCC64E-8AFB-4E5E-8FCB-E231245AC2D2/appyinxiangcom/4048852/ENNote/p333)
    - [启动容器的方式](evernotecid://C2FCC64E-8AFB-4E5E-8FCB-E231245AC2D2/appyinxiangcom/4048852/ENNote/p333)
    - [docker 端口映射](evernotecid://C2FCC64E-8AFB-4E5E-8FCB-E231245AC2D2/appyinxiangcom/4048852/ENNote/p333)
    - [docker 挂载文件](evernotecid://C2FCC64E-8AFB-4E5E-8FCB-E231245AC2D2/appyinxiangcom/4048852/ENNote/p333)
    - [查看nfs容器配置](evernotecid://C2FCC64E-8AFB-4E5E-8FCB-E231245AC2D2/appyinxiangcom/4048852/ENNote/p333)
  + [定制镜像](evernotecid://C2FCC64E-8AFB-4E5E-8FCB-E231245AC2D2/appyinxiangcom/4048852/ENNote/p333)
  + [使用dockerfile定制容器](evernotecid://C2FCC64E-8AFB-4E5E-8FCB-E231245AC2D2/appyinxiangcom/4048852/ENNote/p333)
    - [docker run](evernotecid://C2FCC64E-8AFB-4E5E-8FCB-E231245AC2D2/appyinxiangcom/4048852/ENNote/p333)
  + [docker 工具](evernotecid://C2FCC64E-8AFB-4E5E-8FCB-E231245AC2D2/appyinxiangcom/4048852/ENNote/p333)

## docker 应用场景

目前搞Docker 的原因排序： 技术储备（别的公司都在搞，我们不能拉下） 装逼（我靠，别的公司都有了。我们不能没有啊。） 为自己（都在学docker。不搞以后没法混了） 为公司（Docker能解决我们问题）

Docker改变了什么？

面向产品：产品交付

面向开发：简化环境配置

面向测试：多版本测试

面向运维：环境一致性

面向架构：自动化扩容（微服务）

## 镜像

在介绍docker 镜像之前需要介绍一个AUFS

AUFS是一种Union File System，所谓UnionFS就是把不同物理位置的目录合并mount到同一个目录中。UnionFS的一个最主要的应用是，把一张CD/DVD和一个硬盘目录给联合 mount在一起，然后，你就可以对这个只读的CD/DVD上的文件进行修改（当然，修改的文件存于硬盘上的目录里）。

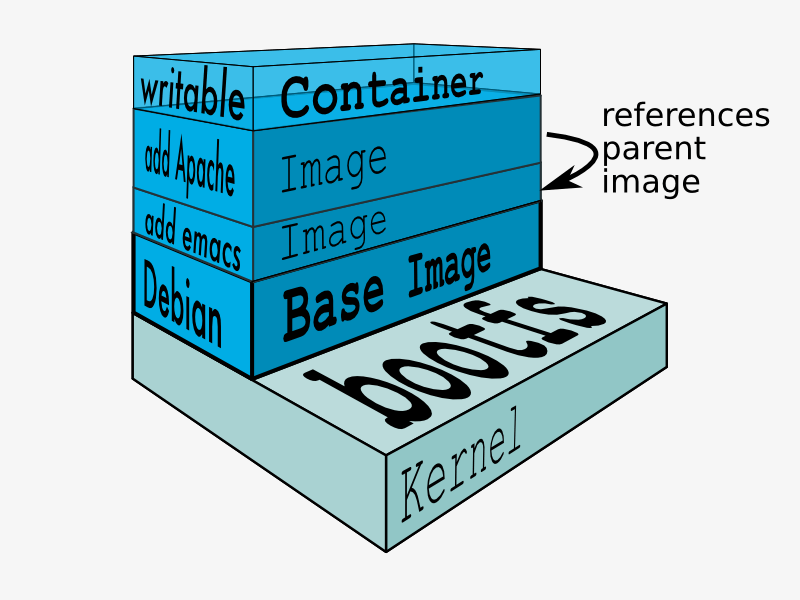
AUFS有所有Union FS的特性，把多个目录，合并成同一个目录，并可以为每个需要合并的目录指定相应的权限，实时的添加、删除、修改已经被mount好的目录。而且，他还能在多个可写的branch/dir间进行负载均衡。

1.**mount** -t aufs -o dirs=./fruits:./vegetables none ./mnt

2.mount -t aufs -o dirs=./fruits=rw:./vegetables=rw none ./mnt

mount aufs过后默认情况下，命令行上第一个目录是可读可写的，后面的都是只读的，我们也可以指定特殊权限。

docker 便把这种特性用于容器的镜像,使用aufs 技术来做分层镜像。



Docker的镜像就是一个只读模板，里面仅安装了用户需要的其它应用程序，镜像用来创建docker容器，可以使用 docker pull来获取镜像

**docker pull [OPTIONS] NAME[:TAG|@DIGEST]**

举例

1.$ docker pull centos

2.$ docker pull dl.dockerpool.com:5000/centos

3.$ docker save centos > /opt/centos.tar.gz

4.$ docker load < /opt/centos.tar.gz

5.$ docker rmi id # 删除镜像

6.$ docker images # 查看现有镜像

查看到的镜像每列的意思为该镜像来自那个仓库，然后是镜像的标记， ID号（唯一） 创建时间与创建的大小

### 创建容器及启动容器

创建容器的方式

1.$ docker run centos /bin/**echo** "hehe" #执行命令后退出

2.hehe

3.$ docker ps -a #查看容器，-a包括停止运行的容器

4.CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

5.bf92bf50c5c4 centos "/bin/echo hehe" About a minute ago Exited (0) About a minute ago desperate\_fermat

6.$ docker run --name mydocker -t -i centos /bin/bash

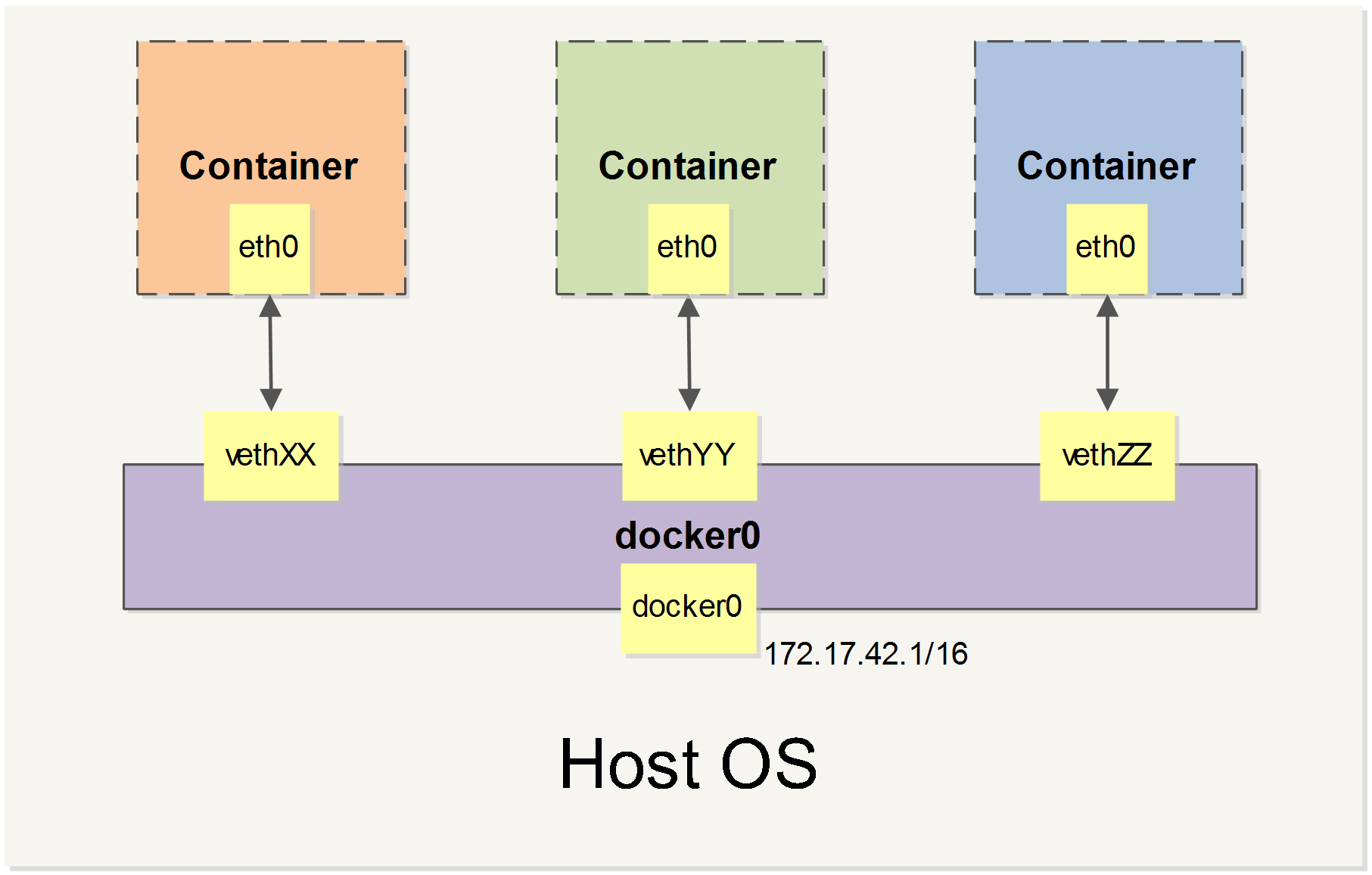
7.#启动一个容器并且进入伪终端, 退出伪终端后容器便退出

8.$ docker stop containerID # 停止容器

9.$ docker rm # 删除容器

### docker 容器网络设置

当docker启动时，会自动创建一个docker0虚拟网桥，实际上是linux的一个bridge，同时docker随机分配一个本地未占用的私有网段中的一个地址给docker0接口，当创建一个容器时，同时会创建一对veth pair接口，当数据包发送到一个接口时，另外一个接口也可以收到相同的数据包，这个接口一端在容器里(eht0)，另一端在本地并挂载到docker0网桥，其名以veth开头



Docker支持的网络模式有：

* none。关闭容器内的网络连接 指定方式--net="none"
* bridge。通过veth接口来连接容器，默认配置。指定方式--net="bridge"
* host。允许容器使用host的网络堆栈信息。 注意：这种方式将允许容器访问host中类似D-BUS之类的系统服务，所以认为是不安全的。指定方式--net="host"
* container。使用另外一个容器的网络堆栈信息。 指定方法：--net="container:name or id"

**None模式**

将网络模式设置为none时，这个容器将不允许访问任何外部router。这个容器内部只会有一个loopback接口，而且不存在任何可以访问外部网络的router。实际上nova-docker用的就是这种方式，这种方式将网络创建的责任完全交给用户，可以实现更加灵活复杂的网络。

**Bridge模式**

Docker默认会将容器设置为bridge模式。此时在主机上面将会存在一个docker0的网络接口，同时会针对容器创建一对veth接口。其中一个veth接口是在主机充当网卡桥接作用，另外一个veth接口存在于容器的命名空间中，并且指向容器的loopback。Docker会自动给这个容器分配一个IP，并且将容器内的数据通过桥接转发到外部。

**Host模式**

当网络模式设置为host时，这个容器将完全共享host的网络堆栈。host所有的网络接口将完全对容器开放。容器的主机名也会存在于主机的hostname中。这时，容器所有对外暴露的端口和对其它容器的连接，将完全失效。

**Container模式**

当网络模式设置为Container时，这个容器将完全复用另外一个容器的网络堆栈。同时使用时这个容器的名称必须要符合下面的格式：–net container:<name|id>.

### 启动容器的方式

使用 -it

1.$ docker run --name mydocker -t -i centos /bin/bash

这种方式是创建容器后直接进入容器，不过退出后容器也跟着停止了，还需要使用**start**再启动

1.$ docker attach 5a8f942ee148

进入一个支行着的窗口，这里需要两次回车便可以进入容器，不过退出后，容器也跟着停止了

1.[root@docker /]# yum install util-linux

2.#如果没有nsenter就需要安装

3.#获取容器的PID

4.$ docker inspect --format '{{.State.Pid}}' 5a8f942ee148

5.39482

6.$ nsenter -t 5a8f942ee148 -u -i -n -p

7.#连接到容器

这样进行容器即使退出，容器也不会停止，我们可以把这个写成一个脚本

1.$ cat ns.sh

2.#!/bin/bash

3.PID=$(docker inspect --format '{{.State.Pid}}' $1)

4.nsenter -t $PID -u -i -n -p

5.$ docker ps -q

6.5a8f942ee148

7.$ sh ns.sh 5a8f942ee148

把容器停止

1.[**root@docker** /]# docker kill $(**docker** ps -q)

2.5a8f942ee148

### docker 端口映射

* -P, –publish-all=false Publish all exposed ports to random ports
* -p, –publish=[] Publish a container’s port(s) to the host

启动nginx，端口随机映射

1.[root@docker ~]# docker run -d -P nginx

2.d4b6ae51ee198f39f24447730d39ad486100533c5f12e699dbaeb751b69ffa32

3.[root@docker ~]# docker ps

4.CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

5.d4b6ae51ee19 nginx "nginx -g 'daemon off" 23 seconds ago Up 22 seconds 0.0.0.0:32769->80/tcp, 0.0.0.0:32768->443/tcp agitated\_torvalds

6.

7.[root@docker ~]# docker run -d -p 81:80 nginx

8.32dfe215c6ceb27ef262c332e90b33529fc2c11e68a3ee1f681bbdbf6e7ab245

9.[root@docker ~]# docker ps

10.CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

11.32dfe215c6ce nginx "nginx -g 'daemon off" 3 seconds ago Up 2 seconds 443/tcp, 0.0.0.0:81->80/tcp

### docker 挂载文件

docker可以允许挂载主机的文件或者目录到容器内，具体方法有

1.$ docker run -it -v /opt:/opt:ro centos #可以指定挂载方式,只读

2. rw 读写

3. -v ~/.bash\_hitstory:/.bash\_**history** #挂载文件

4.

5.$ docker run -d --name nfs -v /data centos

6.b230150ff218c335964a266bf469674948204b983cb9b52f2caa6e79c79853ad

7.[root@docker ~]# docker run -it --name test1 --volumes-from nfs centos

### 查看nfs容器配置

docker inspect nfs

里面会告诉你所有容器的相关设置

## 定制镜像

制作镜像类似git，

[root@docker ~]# docker commit -m “mynginx” f0b75c3b5fef oldboyedu/mynignx:v1

0e600fe53d0109f5ae785c937ea2b378fa1ae620ee7f99b3733c763308019232

[root@docker ~]# docker images

REPOSITORY TAG IMAGE ID CREATED VIRTUAL SIZE

oldboyedu/mynignx v1 0e600fe53d01 38 seconds ago 198.3 MB

docker.io/centos latest 60e65a8e4030 29 hours ago 196.6 MB

docker.io/nginx latest 813e3731b203 9 days ago 133.8 MB

docker.io/registry latest a8706c2bfd21 2 weeks ago 422.8 MB

[root@docker ~]#

手动构建容器

1.docker run --name mynginx -it centos 容器

2. yum install -y nginx

3.docker commit -m "my nginx" 9f2a8b5e4977 oldboyedu/mynginx:v1 镜像

4.docker run -it --name nginxv1 oldboyedu/mynginx:v1 容器

5. vim /etc/nginx/nginx.conf

6. daemon off;

7.docker commit -m "my nginx" cf5cfafa5a7c oldboyedu/mynginx:v2 镜像

8.docker run -d -p 82:80 oldboyedu/mynginx:v2 nginx 容器

## 使用dockerfile定制容器

1.# This docker file

2.# VERSION 1

3.# Author: Jason Zhao

4.# Base image

5.**FROM** centos

6.

7.#Maintainer

8.**MAINTAINER** jason.zhao jason.zhao@oldboyedu.com

9.

10.#Commands

11.**RUN** rpm -ivh http://mirrors.aliyun.com/epel/epel-release-latest-7.noarch.rpm

12.**RUN** yum install -y nginx

13.**ADD** index.html /usr/share/nginx/html/index.html

14.**RUN** **echo** "daemon off;" >> /etc/nginx/nginx.conf

15.**EXPOSE** 80

16.CMD ["nginx"]

[root@docker nginx]# docker build -t oldboyedu/mynginx:v3 /opt/dockerfile/nginx/

[root@docker nginx]# docker run -d -p 83:80 oldboyedu/mynginx:v3

### docker run

dockerHub ->> github

docker registry ->> gitlab

[root@docker nginx]# docker run -d -p 5000:5000 registry

3701696e71022a83ecb32580eb8463b297667521795d1e9426d21e4589119970

[root@docker nginx]# docker tag oldboyedu/mynginx:v3 10.0.0.151:5000/oldboyedu/mynginx:latest

[root@docker nginx]# curl 10.0.0.152:5000/v1/search

{“num\_results”: 0, “query”: “”, “results”: []}

docker tag oldboyedu/mynginx:v3 10.0.0.151:5000/oldboyedu/mynginx:latest

root@linux-node1 ~]# cd /etc/pki/CA

[root@linux-node1 CA]# touch ./{serial,index.txt}

[root@linux-node1 CA]# echo “00” > serial

[root@linux-node1 CA]# openssl genrsa -out private/cakey.pem 2048

openssl req -new -x509 -key private/cakey.pem -days 3650 -out cacert.pem

Country Name (2 letter code) [XX]:CN

State or Province Name (full name) []:Beijing

Locality Name (eg, city) [Default City]:Beijing

Organization Name (eg, company) [Default Company Ltd]:oldboyedu

Organizational Unit Name (eg, section) []:docker

Common Name (eg, your name or your server’s hostname) []:registry.oldboyedu.com

Email Address []:admin@oldboyedu.com

[root@linux-node1 CA]# cd /etc/ssl

[root@linux-node1 ssl]# openssl genrsa -out nginx.key 2048

openssl req -new -key nginx.key -out nginx.csr

Country Name (2 letter code) [XX]:CN

State or Province Name (full name) []:Beijing

Locality Name (eg, city) [Default City]:Beijing

Organization Name (eg, company) [Default Company Ltd]:oldboyedu

Organizational Unit Name (eg, section) []:docker

Common Name (eg, your name or your server’s hostname) []:registry.oldboyedu.com

Email Address []:admin@oldboyedu.com

Please enter the following ‘extra’ attributes

to be sent with your certificate request

A challenge password []:

An optional company name []:

openssl ca -in nginx.csr -days 3650 -out nginx.crt

cat /etc/pki/CA/cacert.pem >> /etc/pki/tls/certs/ca-bundle.crt

htpasswd -c /etc/nginx/conf.d/docker-registry.htpasswd oldboy

docker login -u oldboy -p 123123 -e admin@oldboyedu.com registry.oldboyedu.com

docker push registry.oldboyedu.com/oldboyedu/mynginx:latest

docker tag oldboyedu/mynginx:v2 registry.oldboyedu.com/oldboyedu/mynginx:latest

docker push registry.oldboyedu.com/oldboyedu/mynginx:latest

<http://echarts.baidu.com/doc/example.html>

<http://segmentfault.com/a/1190000002531072>

shipyard 界面管理容器

systemctl restart docker

vim /etc/sysconfig/docker

OPTIONS=’–selinux-enabled –insecure-registry 192.168.56.11:5000 -H tcp://0.0.0.0:235 -H unix://var/run/docker.sock’

systemctl restart docker

curl -s <http://10.0.0.152:235/info> | python -mjson.tool

<http://shipyard-project.com/>

讲师-赵班长 2015/12/26 17:29:11

<http://shipyard-project.com/docs/deploy/>

## docker 工具

<http://shipyard-project.com/>

<http://shipyard-project.com/docs/deploy/>

<http://www.open-open.com/news/view/17df723>

<http://www.07net01.com/2015/09/934439.html>

swarm nesos k8s

shipyard dashboard 管理两台物理机

编写一个dockerfile sshd redis lnmp memcached ,wordpress

熟悉 swarm mesos k8s

elk logstash elasticsearch yum

%23%20docker%20%u521D%u63A2%0A@%28%u4E91%u8BA1%u7B97%29%5Bdocker%5D%0A%0A%5Btoc%5D%0A%0A%23%23%20docker%20%u5E94%u7528%u573A%u666F%0A%0A%u76EE%u524D%u641EDocker%20%u7684%u539F%u56E0%u6392%u5E8F%uFF1A%20%20%u6280%u672F%u50A8%u5907%uFF08%u522B%u7684%u516C%u53F8%u90FD%u5728%u641E%uFF0C%u6211%u4EEC%u4E0D%u80FD%u62C9%u4E0B%uFF09%20%20%20%u88C5%u903C%uFF08%u6211%u9760%uFF0C%u522B%u7684%u516C%u53F8%u90FD%u6709%u4E86%u3002%u6211%u4EEC%u4E0D%u80FD%u6CA1%u6709%u554A%u3002%uFF09%20%20%20%u4E3A%u81EA%u5DF1%uFF08%u90FD%u5728%u5B66docker%u3002%u4E0D%u641E%u4EE5%u540E%u6CA1%u6CD5%u6DF7%u4E86%uFF09%20%20%20%u4E3A%u516C%u53F8%uFF08Docker%u80FD%u89E3%u51B3%u6211%u4EEC%u95EE%u9898%uFF09%0A%0ADocker%u6539%u53D8%u4E86%u4EC0%u4E48%uFF1F%0A%u9762%u5411%u4EA7%u54C1%uFF1A%u4EA7%u54C1%u4EA4%u4ED8%0A%u9762%u5411%u5F00%u53D1%uFF1A%u7B80%u5316%u73AF%u5883%u914D%u7F6E%0A%u9762%u5411%u6D4B%u8BD5%uFF1A%u591A%u7248%u672C%u6D4B%u8BD5%0A%u9762%u5411%u8FD0%u7EF4%uFF1A%u73AF%u5883%u4E00%u81F4%u6027%0A%u9762%u5411%u67B6%u6784%uFF1A%u81EA%u52A8%u5316%u6269%u5BB9%uFF08%u5FAE%u670D%u52A1%uFF09%0A%0A%23%23%20%u955C%u50CF%0A%u5728%u4ECB%u7ECDdocker%20%u955C%u50CF%u4E4B%u524D%u9700%u8981%u4ECB%u7ECD%u4E00%u4E2AAUFS%0A%0AAUFS%u662F%u4E00%u79CDUnion%20File%20System%uFF0C%u6240%u8C13UnionFS%u5C31%u662F%u628A%u4E0D%u540C%u7269%u7406%u4F4D%u7F6E%u7684%u76EE%u5F55%u5408%u5E76mount%u5230%u540C%u4E00%u4E2A%u76EE%u5F55%u4E2D%u3002UnionFS%u7684%u4E00%u4E2A%u6700%u4E3B%u8981%u7684%u5E94%u7528%u662F%uFF0C%u628A%u4E00%u5F20CD/DVD%u548C%u4E00%u4E2A%u786C%u76D8%u76EE%u5F55%u7ED9%u8054%u5408%20mount%u5728%u4E00%u8D77%uFF0C%u7136%u540E%uFF0C%u4F60%u5C31%u53EF%u4EE5%u5BF9%u8FD9%u4E2A%u53EA%u8BFB%u7684CD/DVD%u4E0A%u7684%u6587%u4EF6%u8FDB%u884C%u4FEE%u6539%uFF08%u5F53%u7136%uFF0C%u4FEE%u6539%u7684%u6587%u4EF6%u5B58%u4E8E%u786C%u76D8%u4E0A%u7684%u76EE%u5F55%u91CC%uFF09%u3002%0AAUFS%u6709%u6240%u6709Union%20FS%u7684%u7279%u6027%uFF0C%u628A%u591A%u4E2A%u76EE%u5F55%uFF0C%u5408%u5E76%u6210%u540C%u4E00%u4E2A%u76EE%u5F55%uFF0C%u5E76%u53EF%u4EE5%u4E3A%u6BCF%u4E2A%u9700%u8981%u5408%u5E76%u7684%u76EE%u5F55%u6307%u5B9A%u76F8%u5E94%u7684%u6743%u9650%uFF0C%u5B9E%u65F6%u7684%u6DFB%u52A0%u3001%u5220%u9664%u3001%u4FEE%u6539%u5DF2%u7ECF%u88ABmount%u597D%u7684%u76EE%u5F55%u3002%u800C%u4E14%uFF0C%u4ED6%u8FD8%u80FD%u5728%u591A%u4E2A%u53EF%u5199%u7684branch/dir%u95F4%u8FDB%u884C%u8D1F%u8F7D%u5747%u8861%u3002%0A%60%60%60%0Amount%20-t%20aufs%20-o%20dirs%3D./fruits%3A./vegetables%20none%20./mnt%0Amount%20-t%20aufs%20-o%20dirs%3D./fruits%3Drw%3A./vegetables%3Drw%20none%20./mnt%0A%60%60%60%0Amount%20aufs%u8FC7%u540E%u9ED8%u8BA4%u60C5%u51B5%u4E0B%uFF0C%u547D%u4EE4%u884C%u4E0A%u7B2C%u4E00%u4E2A%u76EE%u5F55%u662F%u53EF%u8BFB%u53EF%u5199%u7684%uFF0C%u540E%u9762%u7684%u90FD%u662F%u53EA%u8BFB%u7684%uFF0C%u6211%u4EEC%u4E5F%u53EF%u4EE5%u6307%u5B9A%u7279%u6B8A%u6743%u9650%u3002%0A%0Adocker%20%u4FBF%u628A%u8FD9%u79CD%u7279%u6027%u7528%u4E8E%u5BB9%u5668%u7684%u955C%u50CF%2C%u4F7F%u7528aufs%20%u6280%u672F%u6765%u505A%u5206%u5C42%u955C%u50CF%u3002%0A%0A%21%5BAlt%20text%5D%28./1451130163996.png%29%0A%0ADocker%u7684%u955C%u50CF%u5C31%u662F%u4E00%u4E2A%u53EA%u8BFB%u6A21%u677F%uFF0C%u91CC%u9762%u4EC5%u5B89%u88C5%u4E86%u7528%u6237%u9700%u8981%u7684%u5176%u5B83%u5E94%u7528%u7A0B%u5E8F%uFF0C%u955C%u50CF%u7528%u6765%u521B%u5EFAdocker%u5BB9%u5668%uFF0C%u53EF%u4EE5%u4F7F%u7528%20%60docker%20pull%20%20%60%u6765%u83B7%u53D6%u955C%u50CF%0A%0A\*\*docker%20pull%20%5BOPTIONS%5D%20NAME%5B%3ATAG%7C@DIGEST%5D\*\*%0A%0A%u4E3E%u4F8B%0A%60%60%60bash%0A%24%20docker%20pull%20centos%0A%24%20docker%20pull%20dl.dockerpool.com%3A5000/centos%0A%24%20docker%20save%20centos%20%3E%20/opt/centos.tar.gz%0A%24%20docker%20load%20%3C%20/opt/centos.tar.gz%0A%24%20docker%20rmi%20id%20%23%20%u5220%u9664%u955C%u50CF%0A%24%20docker%20images%20%23%20%u67E5%u770B%u73B0%u6709%u955C%u50CF%0A%60%60%60%0A%u67E5%u770B%u5230%u7684%u955C%u50CF%u6BCF%u5217%u7684%u610F%u601D%u4E3A%u8BE5%u955C%u50CF%u6765%u81EA%u90A3%u4E2A%u4ED3%u5E93%uFF0C%u7136%u540E%u662F%u955C%u50CF%u7684%u6807%u8BB0%uFF0C%20ID%u53F7%uFF08%u552F%u4E00%uFF09%20%u521B%u5EFA%u65F6%u95F4%u4E0E%u521B%u5EFA%u7684%u5927%u5C0F%0A%0A%23%23%23%20%u521B%u5EFA%u5BB9%u5668%u53CA%u542F%u52A8%u5BB9%u5668%0A%u521B%u5EFA%u5BB9%u5668%u7684%u65B9%u5F0F%0A%60%60%60bash%0A%24%20docker%20run%20centos%20/bin/echo%20%22hehe%22%20%23%u6267%u884C%u547D%u4EE4%u540E%u9000%u51FA%0Ahehe%0A%24%20docker%20ps%20-a%20%23%u67E5%u770B%u5BB9%u5668%uFF0C-a%u5305%u62EC%u505C%u6B62%u8FD0%u884C%u7684%u5BB9%u5668%0ACONTAINER%20ID%20%20IMAGE%20%20%20%20COMMAND%20%20%20%20%20%20%20%20%20%20%20%20%20CREATED%20%20%20%20%20%20%20%20%20%20%20%20%20%20STATUS%20%20%20%20PORTS%20%20%20%20%20%20%20%20%20%20NAMES%0Abf92bf50c5c4%20%20centos%20%20%20%22/bin/echo%20hehe%22%20%20%20%20About%20a%20minute%20ago%20%20%20Exited%20%280%29%20About%20a%20minute%20ago%20%20%20%20%20desperate\_fermat%0A%24%20docker%20run%20--name%20mydocker%20-t%20-i%20centos%20/bin/bash%20%0A%23%u542F%u52A8%u4E00%u4E2A%u5BB9%u5668%u5E76%u4E14%u8FDB%u5165%u4F2A%u7EC8%u7AEF%2C%20%u9000%u51FA%u4F2A%u7EC8%u7AEF%u540E%u5BB9%u5668%u4FBF%u9000%u51FA%0A%24%20docker%20stop%20containerID%20%23%20%u505C%u6B62%u5BB9%u5668%0A%24%20docker%20rm%20%23%20%u5220%u9664%u5BB9%u5668%0A%60%60%60%0A%0A%23%23%23%20docker%20%u5BB9%u5668%u7F51%u7EDC%u8BBE%u7F6E%0A%u5F53docker%u542F%u52A8%u65F6%uFF0C%u4F1A%u81EA%u52A8%u521B%u5EFA%u4E00%u4E2Adocker0%u865A%u62DF%u7F51%u6865%uFF0C%u5B9E%u9645%u4E0A%u662Flinux%u7684%u4E00%u4E2Abridge%uFF0C%u540C%u65F6docker%u968F%u673A%u5206%u914D%u4E00%u4E2A%u672C%u5730%u672A%u5360%u7528%u7684%u79C1%u6709%u7F51%u6BB5%u4E2D%u7684%u4E00%u4E2A%u5730%u5740%u7ED9docker0%u63A5%u53E3%uFF0C%u5F53%u521B%u5EFA%u4E00%u4E2A%u5BB9%u5668%u65F6%uFF0C%u540C%u65F6%u4F1A%u521B%u5EFA%u4E00%u5BF9veth%20pair%u63A5%u53E3%uFF0C%u5F53%u6570%u636E%u5305%u53D1%u9001%u5230%u4E00%u4E2A%u63A5%u53E3%u65F6%uFF0C%u53E6%u5916%u4E00%u4E2A%u63A5%u53E3%u4E5F%u53EF%u4EE5%u6536%u5230%u76F8%u540C%u7684%u6570%u636E%u5305%uFF0C%u8FD9%u4E2A%u63A5%u53E3%u4E00%u7AEF%u5728%u5BB9%u5668%u91CC%28eht0%29%uFF0C%u53E6%u4E00%u7AEF%u5728%u672C%u5730%u5E76%u6302%u8F7D%u5230docker0%u7F51%u6865%uFF0C%u5176%u540D%u4EE5veth%u5F00%u5934%0A%21%5BAlt%20text%5D%28./1451915298931.png%29%0A%0ADocker%u652F%u6301%u7684%u7F51%u7EDC%u6A21%u5F0F%u6709%uFF1A%0A-%20none%u3002%u5173%u95ED%u5BB9%u5668%u5185%u7684%u7F51%u7EDC%u8FDE%u63A5%20%u6307%u5B9A%u65B9%u5F0F%60--net%3D%22none%22%60%0A-%20bridge%u3002%u901A%u8FC7veth%u63A5%u53E3%u6765%u8FDE%u63A5%u5BB9%u5668%uFF0C%u9ED8%u8BA4%u914D%u7F6E%u3002%u6307%u5B9A%u65B9%u5F0F%60--net%3D%22bridge%22%60%0A-%20host%u3002%u5141%u8BB8%u5BB9%u5668%u4F7F%u7528host%u7684%u7F51%u7EDC%u5806%u6808%u4FE1%u606F%u3002%20%u6CE8%u610F%uFF1A%u8FD9%u79CD%u65B9%u5F0F%u5C06%u5141%u8BB8%u5BB9%u5668%u8BBF%u95EEhost%u4E2D%u7C7B%u4F3CD-BUS%u4E4B%u7C7B%u7684%u7CFB%u7EDF%u670D%u52A1%uFF0C%u6240%u4EE5%u8BA4%u4E3A%u662F%u4E0D%u5B89%u5168%u7684%u3002%u6307%u5B9A%u65B9%u5F0F%60--net%3D%22host%22%60%0A-%20container%u3002%u4F7F%u7528%u53E6%u5916%u4E00%u4E2A%u5BB9%u5668%u7684%u7F51%u7EDC%u5806%u6808%u4FE1%u606F%u3002%20%u6307%u5B9A%u65B9%u6CD5%uFF1A%60--net%3D%22container%3Aname%20or%20id%22%60%0A%0A\*\*None%u6A21%u5F0F\*\*%0A%u5C06%u7F51%u7EDC%u6A21%u5F0F%u8BBE%u7F6E%u4E3Anone%u65F6%uFF0C%u8FD9%u4E2A%u5BB9%u5668%u5C06%u4E0D%u5141%u8BB8%u8BBF%u95EE%u4EFB%u4F55%u5916%u90E8router%u3002%u8FD9%u4E2A%u5BB9%u5668%u5185%u90E8%u53EA%u4F1A%u6709%u4E00%u4E2Aloopback%u63A5%u53E3%uFF0C%u800C%u4E14%u4E0D%u5B58%u5728%u4EFB%u4F55%u53EF%u4EE5%u8BBF%u95EE%u5916%u90E8%u7F51%u7EDC%u7684router%u3002%u5B9E%u9645%u4E0Anova-docker%u7528%u7684%u5C31%u662F%u8FD9%u79CD%u65B9%u5F0F%uFF0C%u8FD9%u79CD%u65B9%u5F0F%u5C06%u7F51%u7EDC%u521B%u5EFA%u7684%u8D23%u4EFB%u5B8C%u5168%u4EA4%u7ED9%u7528%u6237%uFF0C%u53EF%u4EE5%u5B9E%u73B0%u66F4%u52A0%u7075%u6D3B%u590D%u6742%u7684%u7F51%u7EDC%u3002%0A%0A\*\*Bridge%u6A21%u5F0F\*\*%0ADocker%u9ED8%u8BA4%u4F1A%u5C06%u5BB9%u5668%u8BBE%u7F6E%u4E3Abridge%u6A21%u5F0F%u3002%u6B64%u65F6%u5728%u4E3B%u673A%u4E0A%u9762%u5C06%u4F1A%u5B58%u5728%u4E00%u4E2Adocker0%u7684%u7F51%u7EDC%u63A5%u53E3%uFF0C%u540C%u65F6%u4F1A%u9488%u5BF9%u5BB9%u5668%u521B%u5EFA%u4E00%u5BF9veth%u63A5%u53E3%u3002%u5176%u4E2D%u4E00%u4E2Aveth%u63A5%u53E3%u662F%u5728%u4E3B%u673A%u5145%u5F53%u7F51%u5361%u6865%u63A5%u4F5C%u7528%uFF0C%u53E6%u5916%u4E00%u4E2Aveth%u63A5%u53E3%u5B58%u5728%u4E8E%u5BB9%u5668%u7684%u547D%u540D%u7A7A%u95F4%u4E2D%uFF0C%u5E76%u4E14%u6307%u5411%u5BB9%u5668%u7684loopback%u3002Docker%u4F1A%u81EA%u52A8%u7ED9%u8FD9%u4E2A%u5BB9%u5668%u5206%u914D%u4E00%u4E2AIP%uFF0C%u5E76%u4E14%u5C06%u5BB9%u5668%u5185%u7684%u6570%u636E%u901A%u8FC7%u6865%u63A5%u8F6C%u53D1%u5230%u5916%u90E8%u3002%0A%0A\*\*Host%u6A21%u5F0F\*\*%0A%u5F53%u7F51%u7EDC%u6A21%u5F0F%u8BBE%u7F6E%u4E3Ahost%u65F6%uFF0C%u8FD9%u4E2A%u5BB9%u5668%u5C06%u5B8C%u5168%u5171%u4EABhost%u7684%u7F51%u7EDC%u5806%u6808%u3002host%u6240%u6709%u7684%u7F51%u7EDC%u63A5%u53E3%u5C06%u5B8C%u5168%u5BF9%u5BB9%u5668%u5F00%u653E%u3002%u5BB9%u5668%u7684%u4E3B%u673A%u540D%u4E5F%u4F1A%u5B58%u5728%u4E8E%u4E3B%u673A%u7684hostname%u4E2D%u3002%u8FD9%u65F6%uFF0C%u5BB9%u5668%u6240%u6709%u5BF9%u5916%u66B4%u9732%u7684%u7AEF%u53E3%u548C%u5BF9%u5176%u5B83%u5BB9%u5668%u7684%u8FDE%u63A5%uFF0C%u5C06%u5B8C%u5168%u5931%u6548%u3002%0A%0A\*\*Container%u6A21%u5F0F\*\*%0A%u5F53%u7F51%u7EDC%u6A21%u5F0F%u8BBE%u7F6E%u4E3AContainer%u65F6%uFF0C%u8FD9%u4E2A%u5BB9%u5668%u5C06%u5B8C%u5168%u590D%u7528%u53E6%u5916%u4E00%u4E2A%u5BB9%u5668%u7684%u7F51%u7EDC%u5806%u6808%u3002%u540C%u65F6%u4F7F%u7528%u65F6%u8FD9%u4E2A%u5BB9%u5668%u7684%u540D%u79F0%u5FC5%u987B%u8981%u7B26%u5408%u4E0B%u9762%u7684%u683C%u5F0F%uFF1A--net%20container%3A%3Cname%7Cid%3E.%0A%0A%23%23%23%20%u542F%u52A8%u5BB9%u5668%u7684%u65B9%u5F0F%0A%u4F7F%u7528%20-it%20%0A%60%60%60bash%0A%24%20docker%20run%20--name%20mydocker%20-t%20-i%20centos%20/bin/bash%20%0A%60%60%60%0A%u8FD9%u79CD%u65B9%u5F0F%u662F%u521B%u5EFA%u5BB9%u5668%u540E%u76F4%u63A5%u8FDB%u5165%u5BB9%u5668%uFF0C%u4E0D%u8FC7%u9000%u51FA%u540E%u5BB9%u5668%u4E5F%u8DDF%u7740%u505C%u6B62%u4E86%uFF0C%u8FD8%u9700%u8981%u4F7F%u7528\*\*start\*\*%u518D%u542F%u52A8%0A%60%60%60bash%0A%24%20docker%20attach%205a8f942ee148%0A%60%60%60%0A%u8FDB%u5165%u4E00%u4E2A%u652F%u884C%u7740%u7684%u7A97%u53E3%uFF0C%u8FD9%u91CC%u9700%u8981%u4E24%u6B21%u56DE%u8F66%u4FBF%u53EF%u4EE5%u8FDB%u5165%u5BB9%u5668%uFF0C%u4E0D%u8FC7%u9000%u51FA%u540E%uFF0C%u5BB9%u5668%u4E5F%u8DDF%u7740%u505C%u6B62%u4E86%0A%0A%60%60%60bash%0A%5Broot@docker%20/%5D%23%20yum%20install%20util-linux%0A%23%u5982%u679C%u6CA1%u6709nsenter%u5C31%u9700%u8981%u5B89%u88C5%0A%23%u83B7%u53D6%u5BB9%u5668%u7684PID%0A%24%20docker%20inspect%20--format%20%27%7B%7B.State.Pid%7D%7D%27%205a8f942ee148%0A39482%0A%24%20nsenter%20-t%205a8f942ee148%20-u%20-i%20-n%20-p%20%0A%23%u8FDE%u63A5%u5230%u5BB9%u5668%0A%60%60%60%0A%u8FD9%u6837%u8FDB%u884C%u5BB9%u5668%u5373%u4F7F%u9000%u51FA%uFF0C%u5BB9%u5668%u4E5F%u4E0D%u4F1A%u505C%u6B62%uFF0C%u6211%u4EEC%u53EF%u4EE5%u628A%u8FD9%u4E2A%u5199%u6210%u4E00%u4E2A%u811A%u672C%0A%60%60%60bash%0A%24%20cat%20ns.sh%20%0A%23%21/bin/bash%0APID%3D%24%28docker%20inspect%20--format%20%27%7B%7B.State.Pid%7D%7D%27%20%241%29%0Ansenter%20-t%20%24PID%20-u%20-i%20-n%20-p%0A%24%20docker%20ps%20%20-q%0A5a8f942ee148%0A%24%20sh%20ns.sh%205a8f942ee148%0A%60%60%60%0A%0A%u628A%u5BB9%u5668%u505C%u6B62%0A%60%60%60%0A%5Broot@docker%20/%5D%23%20docker%20kill%20%24%28docker%20ps%20%20-q%29%0A5a8f942ee148%0A%60%60%60%0A%0A%23%23%23%20docker%20%u7AEF%u53E3%u6620%u5C04%0A-%20%20-P%2C%20--publish-all%3Dfalse%20%20%20%20%20%20%20%20%20Publish%20all%20exposed%20ports%20to%20random%20ports%0A-%20%20-p%2C%20--publish%3D%5B%5D%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20Publish%20a%20container%27s%20port%28s%29%20to%20the%20host%0A%0A%u542F%u52A8nginx%uFF0C%u7AEF%u53E3%u968F%u673A%u6620%u5C04%0A%60%60%60%0A%5Broot@docker%20%7E%5D%23%20docker%20run%20-d%20-P%20nginx%0Ad4b6ae51ee198f39f24447730d39ad486100533c5f12e699dbaeb751b69ffa32%0A%5Broot@docker%20%7E%5D%23%20docker%20ps%20%0ACONTAINER%20ID%20%20%20%20%20%20%20%20IMAGE%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20COMMAND%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20CREATED%20%20%20%20%20%20%20%20%20%20%20%20%20STATUS%20%20%20%20%20%20%20%20%20%20%20%20%20%20PORTS%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20NAMES%0Ad4b6ae51ee19%20%20%20%20%20%20%20%20nginx%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%22nginx%20-g%20%27daemon%20off%22%20%20%2023%20seconds%20ago%20%20%20%20%20%20Up%2022%20seconds%20%20%20%20%20%20%200.0.0.0%3A32769-%3E80/tcp%2C%200.0.0.0%3A32768-%3E443/tcp%20%20%20agitated\_torvalds%0A%0A%5Broot@docker%20%7E%5D%23%20docker%20run%20-d%20-p%2081%3A80%20%20nginx%0A32dfe215c6ceb27ef262c332e90b33529fc2c11e68a3ee1f681bbdbf6e7ab245%0A%5Broot@docker%20%7E%5D%23%20docker%20ps%0ACONTAINER%20ID%20%20%20%20%20%20%20%20IMAGE%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20COMMAND%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20CREATED%20%20%20%20%20%20%20%20%20%20%20%20%20STATUS%20%20%20%20%20%20%20%20%20%20%20%20%20%20PORTS%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20NAMES%0A32dfe215c6ce%20%20%20%20%20%20%20%20nginx%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%22nginx%20-g%20%27daemon%20off%22%20%20%203%20seconds%20ago%20%20%20%20%20%20%20Up%202%20seconds%20%20%20%20%20%20%20%20443/tcp%2C%200.0.0.0%3A81-%3E80/tcp%20%20%0A%60%60%60%0A%0A%23%23%23%20docker%20%u6302%u8F7D%u6587%u4EF6%0Adocker%u53EF%u4EE5%u5141%u8BB8%u6302%u8F7D%u4E3B%u673A%u7684%u6587%u4EF6%u6216%u8005%u76EE%u5F55%u5230%u5BB9%u5668%u5185%uFF0C%u5177%u4F53%u65B9%u6CD5%u6709%0A%60%60%60bash%0A%24%20docker%20run%20-it%20-v%20/opt%3A/opt%3Aro%20centos%20%23%u53EF%u4EE5%u6307%u5B9A%u6302%u8F7D%u65B9%u5F0F%2C%u53EA%u8BFB%0A%09%09%20%20%09%09%09%09%09rw%20%20%20%u8BFB%u5199%0A%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20-v%20%7E/.bash\_hitstory%3A/.bash\_history%20%23%u6302%u8F7D%u6587%u4EF6%0A%0A%24%20docker%20run%20-d%20--name%20nfs%20-v%20/data%20centos%0Ab230150ff218c335964a266bf469674948204b983cb9b52f2caa6e79c79853ad%0A%5Broot@docker%20%7E%5D%23%20docker%20run%20-it%20--name%20test1%20--volumes-from%20nfs%20centos%20%20%0A%60%60%60%0A%0A%23%23%23%u67E5%u770Bnfs%u5BB9%u5668%u914D%u7F6E%0A%0Adocker%20inspect%20nfs%0A%u91CC%u9762%u4F1A%u544A%u8BC9%u4F60%u6240%u6709%u5BB9%u5668%u7684%u76F8%u5173%u8BBE%u7F6E%0A%0A%23%23%20%u5B9A%u5236%u955C%u50CF%0A%u5236%u4F5C%u955C%u50CF%u7C7B%u4F3Cgit%uFF0C%0A%5Broot@docker%20%7E%5D%23%20docker%20commit%20-m%20%22mynginx%22%20f0b75c3b5fef%20oldboyedu/mynignx%3Av1%0A0e600fe53d0109f5ae785c937ea2b378fa1ae620ee7f99b3733c763308019232%0A%5Broot@docker%20%7E%5D%23%20docker%20images%0AREPOSITORY%20%20%20%20%20%20%20%20%20%20%20TAG%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20IMAGE%20ID%20%20%20%20%20%20%20%20%20%20%20%20CREATED%20%20%20%20%20%20%20%20%20%20%20%20%20VIRTUAL%20SIZE%0Aoldboyedu/mynignx%20%20%20%20v1%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%20%200e600fe53d01%20%20%20%20%20%20%20%2038%20seconds%20ago%20%20%20%20%20%20198.3%20MB%0Adocker.io/centos%20%20%20%20%20latest%20%20%20%20%20%20%20%20%20%20%20%20%20%2060e65a8e4030%20%20%20%20%20%20%20%2029%20hours%20ago%20%20%20%20%20%20%20%20196.6%20MB%0Adocker.io/nginx%20%20%20%20%20%20latest%20%20%20%20%20%20%20%20%20%20%20%20%20%20813e3731b203%20%20%20%20%20%20%20%209%20days%20ago%20%20%20%20%20%20%20%20%20%20133.8%20MB%0Adocker.io/registry%20%20%20latest%20%20%20%20%20%20%20%20%20%20%20%20%20%20a8706c2bfd21%20%20%20%20%20%20%20%202%20weeks%20ago%20%20%20%20%20%20%20%20%20422.8%20MB%0A%5Broot@docker%20%7E%5D%23%20%0A%0A%0A%u624B%u52A8%u6784%u5EFA%u5BB9%u5668%0A%60%60%60bash%0Adocker%20run%20--name%20mynginx%20-it%20centos%20%20%u5BB9%u5668%0A%20%20%20yum%20install%20-y%20nginx%0Adocker%20commit%20-m%20%22my%20nginx%22%209f2a8b5e4977%20oldboyedu/mynginx%3Av1%20%20%u955C%u50CF%0Adocker%20run%20-it%20--name%20nginxv1%20oldboyedu/mynginx%3Av1%20%20%u5BB9%u5668%0A%20%20%20vim%20/etc/nginx/nginx.conf%0A%20%20%20%20%20daemon%20off%3B%0Adocker%20commit%20-m%20%22my%20nginx%22%20cf5cfafa5a7c%20oldboyedu/mynginx%3Av2%20%20%u955C%u50CF%0Adocker%20run%20-d%20-p%2082%3A80%20oldboyedu/mynginx%3Av2%20nginx%20%20%u5BB9%u5668%0A%60%60%60%0A%0A%23%23%u4F7F%u7528dockerfile%u5B9A%u5236%u5BB9%u5668%0A%0A%60%60%60%0A%23%20This%20docker%20file%20%0A%23%20VERSION%201%0A%23%20Author%3A%20Jason%20Zhao%0A%23%20Base%20image%0AFROM%20centos%0A%0A%23Maintainer%0AMAINTAINER%20jason.zhao%20jason.zhao@oldboyedu.com%0A%0A%23Commands%0ARUN%20rpm%20-ivh%20http%3A//mirrors.aliyun.com/epel/epel-release-latest-7.noarch.rpm%0ARUN%20yum%20install%20-y%20nginx%0AADD%20index.html%20/usr/share/nginx/html/index.html%0ARUN%20echo%20%22daemon%20off%3B%22%20%3E%3E%20/etc/nginx/nginx.conf%0AEXPOSE%2080%0ACMD%20%5B%22nginx%22%5D%0A%60%60%60%0A%5Broot@docker%20nginx%5D%23%20docker%20build%20-t%20oldboyedu/mynginx%3Av3%20/opt/dockerfile/nginx/%0A%5Broot@docker%20nginx%5D%23%20docker%20run%20-d%20-p%2083%3A80%20oldboyedu/mynginx%3Av3%20%0A%0A%0A%0A%23%23%23%20docker%20run%0A%0AdockerHub%20-%3E%3E%20github%0Adocker%20registry%20-%3E%3E%20gitlab%0A%0A%0A%0A%0A%5Broot@docker%20nginx%5D%23%20docker%20run%20-d%20-p%205000%3A5000%20registry%20%0A3701696e71022a83ecb32580eb8463b297667521795d1e9426d21e4589119970%0A%5Broot@docker%20nginx%5D%23%20docker%20tag%20oldboyedu/mynginx%3Av3%2010.0.0.151%3A5000/oldboyedu/mynginx%3Alatest%0A%0A%5Broot@docker%20nginx%5D%23%20curl%2010.0.0.152%3A5000/v1/search%0A%7B%22num\_results%22%3A%200%2C%20%22query%22%3A%20%22%22%2C%20%22results%22%3A%20%5B%5D%7D%0A%0Adocker%20tag%20oldboyedu/mynginx%3Av3%2010.0.0.151%3A5000/oldboyedu/mynginx%3Alatest%0A%0Aroot@linux-node1%20%7E%5D%23%20cd%20/etc/pki/CA%0A%5Broot@linux-node1%20CA%5D%23%20touch%20./%7Bserial%2Cindex.txt%7D%0A%5Broot@linux-node1%20CA%5D%23%20echo%20%2200%22%20%3E%20serial%20%0A%5Broot@linux-node1%20CA%5D%23%20openssl%20genrsa%20-out%20private/cakey.pem%202048%0A%0Aopenssl%20req%20-new%20-x509%20-key%20private/cakey.pem%20-days%203650%20-out%20cacert.pem%0A%0A%0ACountry%20Name%20%282%20letter%20code%29%20%5BXX%5D%3ACN%0AState%20or%20Province%20Name%20%28full%20name%29%20%5B%5D%3ABeijing%0ALocality%20Name%20%28eg%2C%20city%29%20%5BDefault%20City%5D%3ABeijing%0AOrganization%20Name%20%28eg%2C%20company%29%20%5BDefault%20Company%20Ltd%5D%3Aoldboyedu%0AOrganizational%20Unit%20Name%20%28eg%2C%20section%29%20%5B%5D%3Adocker%0ACommon%20Name%20%28eg%2C%20your%20name%20or%20your%20server%27s%20hostname%29%20%5B%5D%3Aregistry.oldboyedu.com%0AEmail%20Address%20%5B%5D%3Aadmin@oldboyedu.com%0A%0A%5Broot@linux-node1%20CA%5D%23%20cd%20/etc/ssl%0A%5Broot@linux-node1%20ssl%5D%23%20openssl%20genrsa%20-out%20nginx.key%202048%0Aopenssl%20req%20-new%20-key%20nginx.key%20-out%20nginx.csr%0A%0ACountry%20Name%20%282%20letter%20code%29%20%5BXX%5D%3ACN%0AState%20or%20Province%20Name%20%28full%20name%29%20%5B%5D%3ABeijing%0ALocality%20Name%20%28eg%2C%20city%29%20%5BDefault%20City%5D%3ABeijing%0AOrganization%20Name%20%28eg%2C%20company%29%20%5BDefault%20Company%20Ltd%5D%3Aoldboyedu%0AOrganizational%20Unit%20Name%20%28eg%2C%20section%29%20%5B%5D%3Adocker%0ACommon%20Name%20%28eg%2C%20your%20name%20or%20your%20server%27s%20hostname%29%20%5B%5D%3Aregistry.oldboyedu.com%0AEmail%20Address%20%5B%5D%3Aadmin@oldboyedu.com%0A%0APlease%20enter%20the%20following%20%27extra%27%20attributes%0Ato%20be%20sent%20with%20your%20certificate%20request%0AA%20challenge%20password%20%5B%5D%3A%0AAn%20optional%20company%20name%20%5B%5D%3A%0A%0A%0Aopenssl%20ca%20-in%20nginx.csr%20-days%203650%20-out%20nginx.crt%0A%0Acat%20/etc/pki/CA/cacert.pem%20%3E%3E%20/etc/pki/tls/certs/ca-bundle.crt%20%0A%0A%0Ahtpasswd%20-c%20/etc/nginx/conf.d/docker-registry.htpasswd%20oldboy%0A%0Adocker%20login%20-u%20oldboy%20-p%20123123%20-e%20admin@oldboyedu.com%20registry.oldboyedu.com%0A%0A%0Adocker%20push%20registry.oldboyedu.com/oldboyedu/mynginx%3Alatest%0A%0A%0A%0A%0A%0A%0A%0Adocker%20tag%20oldboyedu/mynginx%3Av2%20registry.oldboyedu.com/oldboyedu/mynginx%3Alatest%0Adocker%20push%20registry.oldboyedu.com/oldboyedu/mynginx%3Alatest%0A%0A%0A%0A%0Ahttp%3A//echarts.baidu.com/doc/example.html%0A%0Ahttp%3A//segmentfault.com/a/1190000002531072%0A%0A%0Ashipyard%20%u754C%u9762%u7BA1%u7406%u5BB9%u5668%0A%0Asystemctl%20restart%20docker%0A%0A%20vim%20/etc/sysconfig/docker%0A%0AOPTIONS%3D%27--selinux-enabled%20--insecure-registry%20192.168.56.11%3A5000%20-H%20tcp%3A//0.0.0.0%3A235%20-H%20unix%3A//var/run/docker.sock%27%0A%0Asystemctl%20restart%20docker%0Acurl%20-s%20http%3A//10.0.0.152%3A235/info%20%7C%20python%20-mjson.tool%0A%0Ahttp%3A//shipyard-project.com/%0A%u8BB2%u5E08-%u8D75%u73ED%u957F%202015/12/26%2017%3A29%3A11%0Ahttp%3A//shipyard-project.com/docs/deploy/%0A%0A%0A%23%23%20docker%20%u5DE5%u5177%0A%0Ahttp%3A//shipyard-project.com/%0A%0Ahttp%3A//shipyard-project.com/docs/deploy/%0A%0Ahttp%3A//www.open-open.com/news/view/17df723%0A%0Ahttp%3A//www.07net01.com/2015/09/934439.html%0A%0Aswarm%20nesos%20k8s%0A%0A%0Ashipyard%20dashboard%20%u7BA1%u7406%u4E24%u53F0%u7269%u7406%u673A%0A%u7F16%u5199%u4E00%u4E2Adockerfile%20sshd%20redis%20lnmp%20memcached%20%2Cwordpress%0A%u719F%u6089%20swarm%20mesos%20k8s%0A%0Aelk%20%20logstash%20elasticsearch%20yum%0A