DOCKER实验报告

姓名：洪燊

学号：17343042

实验环境

操作系统：Ubuntu 19.02

实验流程

docker安装

1. 更新apt包索引

```
sudo apt-get update
```

2. 安装apt依赖包

```
sudo apt-get install \
    apt-transport-https \
    ca-certificates \
    curl \
    gnupg-agent \
    software-properties-common
```

3. 添加docker官方的GPG密钥：

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
```

4. 通过搜索指纹的后八个字符，验证指纹密钥

```
sudo apt-key fingerprint 0EBFCD88
```

5. 设置稳定版仓库

```
sudo add-apt-repository \
   "deb [arch=amd64] https://download.docker.com/linux/ubuntu \
  $(lsb_release -cs) \
  stable"
```

6. 安装docker engine-community

```
sudo apt-get install docker-ce docker-ce-cli containerd.io
```

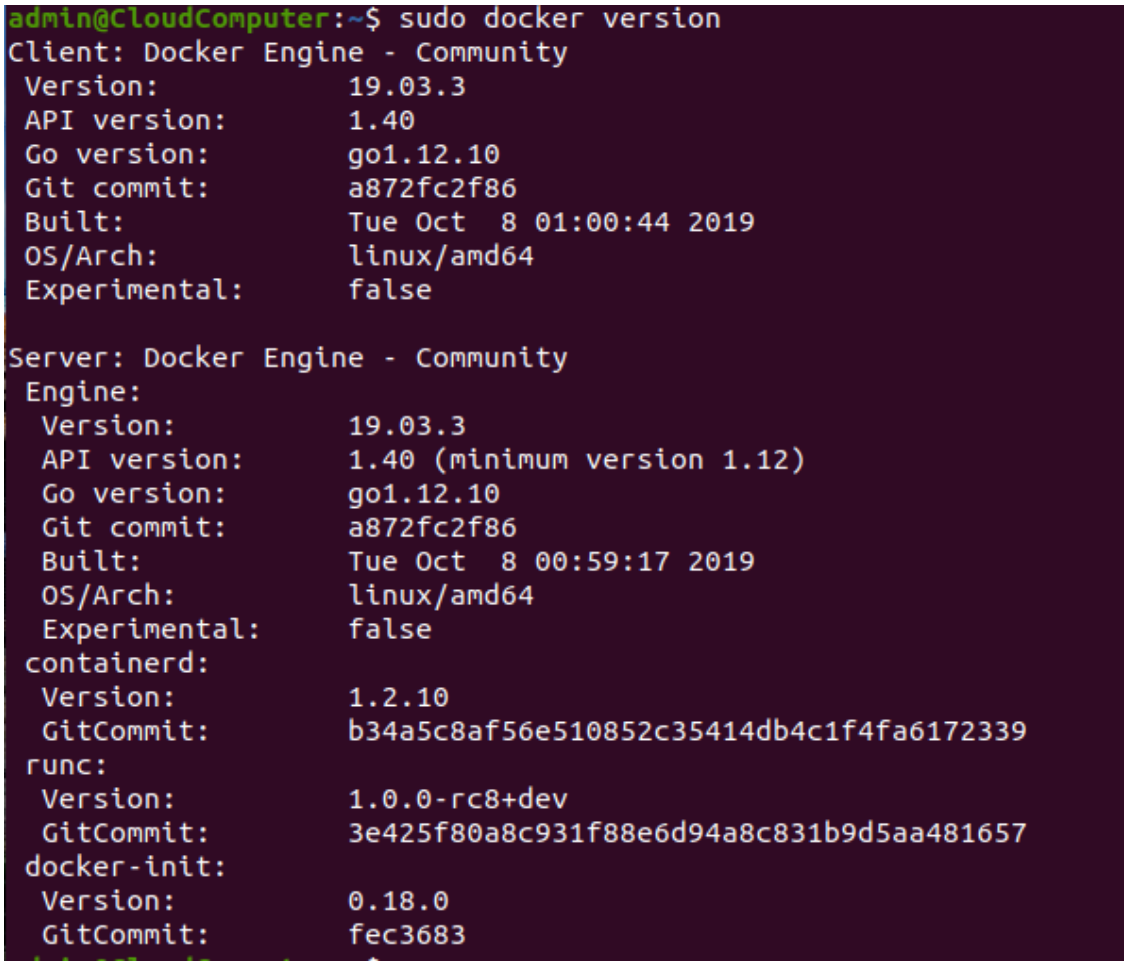
7. 将当前用户添加进docker用户组

```
sudo groupadd docker
sudo gpasswd -a ${USER} docker
sudo service docker restart
```

注销当前用户并重新登陆

8. 测试docker是否安装成功

```
sudo docker version
```



运行第一个容器

运行镜像

```
sudo docker run hello-world
```

```
admin@CloudComputer:~$ sudo docker run hello-world
Unable to find image 'hello-world:latest' locally
latest: Pulling from library/hello-world
1b930d010525: Pull complete
Digest: sha256:4fe721ccc2e8dc7362278a29dc660d833570ec2682f4e4194f4ee23e415e1064
Status: Downloaded newer image for hello-world:latest

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/

admin@CloudComputer:~$
```

```
docker run -it ubuntu bash
```

```
admin@CloudComputer:~$ sudo docker run -it ubuntu bash
[sudo] admin 的密码:
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
7ddbc47eeb70: Pull complete
c1bbdc448b72: Pull complete
8c3b70e39044: Pull complete
45d437916d57: Pull complete
Digest: sha256:6e9f67fa63b0323e9a1e587fd71c561ba48a034504fb804fd26fd8800039835d
Status: Downloaded newer image for ubuntu:latest
root@cd5c3833ddc1:/#
root@cd5c3833ddc1:/# ls
bin   dev  home  lib64  mnt  proc  run   srv  tmp  var
boot  etc  lib   media  opt  root  sbin  sys  usr
root@cd5c3833ddc1:/# exit
exit
admin@CloudComputer:~$
```

显示本地镜像库内容

```
docker images
```

```
admin@CloudComputer:~$ sudo docker images
REPOSITORY          TAG                 IMAGE ID            CREATED
SIZE
ubuntu              latest              775349758637        6 weeks ago
64.2MB
hello-world         latest              fce289e99eb9        11 months ago
1.84kB
admin@CloudComputer:~$
```

获得帮助

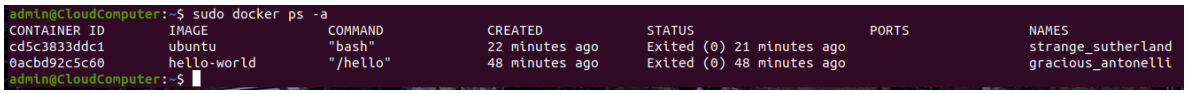
```
docker --help
```

显示运行中容器

```
docker ps
```

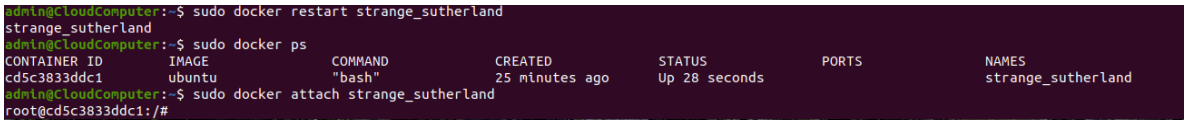
显示所有容器（包含已中止）

```
docker ps -a
```



继续运行原容器并进入

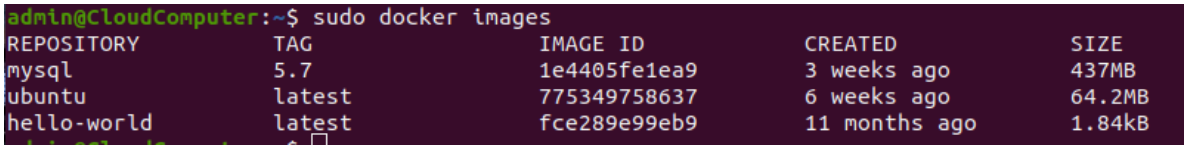
```
docker restart [容器名]
docker ps
docker attach [容器名]
```



MySQL与容器化

拉取MySQL镜像并检查

```
docker pull mysql:5.7
docker images
```



构建docker镜像

在创建有有效dockerfile的文件的文件目录下执行

```
docker build . -t hello
```

```
root@CloudComputer:/home/admin/桌面/mydock# docker build . -t hello
Sending build context to Docker daemon  2.048kB
Step 1/3 : FROM ubuntu
 ---> 775349758637
Step 2/3 : ENTRYPOINT ["top", "-b"]
 ---> Running in 568c06e5a5b8
Removing intermediate container 568c06e5a5b8
 ---> 45a98d0c6762
Step 3/3 : CMD ["-c"]
 ---> Running in 11325c925b1f
Removing intermediate container 11325c925b1f
 ---> 1e17f3f188a2
Successfully built 1e17f3f188a2
Successfully tagged hello:latest
root@CloudComputer:/home/admin/桌面/mydock#
```

使用docker images查看，可以看到创建成功：

```
root@CloudComputer:/home/admin/桌面/mydock# docker images
REPOSITORY          TAG                 IMAGE ID            CREATED
 SIZE
hello               latest              1e17f3f188a2        About a minute ago
 64.2MB
mysql               5.7                 1e4405fe1ea9        3 weeks ago
 437MB
ubuntu              latest              775349758637        6 weeks ago
 64.2MB
hello-world         latest              fce289e99eb9        11 months ago
 1.84kB
root@CloudComputer:/home/admin/桌面/mydock#
```

运行创建的镜像

```
docker run -it --rm hello -H
```

其中--rm参数让docker退出时还原到默认容器的文件系统内容，运行成功：

```
root@CloudComputer:/home/admin/桌面/mydock# docker run -it --rm hello -H
top - 14:06:13 up  5:00,  0 users,  load average: 0.00, 0.06, 0.02
Threads:   1 total,   1 running,   0 sleeping,   0 stopped,   0 zombie
%Cpu(s):  2.2 us,  0.3 sy,  0.0 ni, 95.4 id,  1.9 wa,  0.0 hi,  0.2 si,  0.0 st
KiB Mem :  8165428 total,  3753012 free,   955568 used,  3456848 buff/cache
KiB Swap:  1459804 total,  1459804 free,        0 used.  6924800 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
    1 root      20   0   36480   3096   2736 R  0.0  0.0   0:00.42 top
```

使用MySQL容器

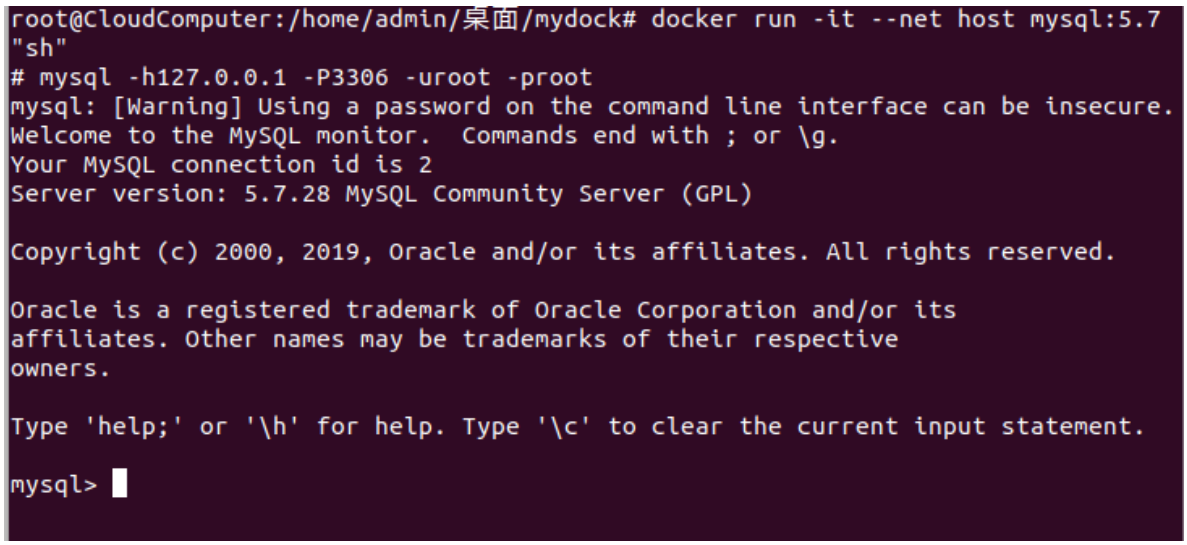
启动服务器

```
sudo docker run -p 3306:3306 --name mysql2 -e MYSQL_ROOT_PASSWORD=root -d
mysql:5.7
```

```
root@CloudComputer:/home/admin/桌面/mydock# sudo docker run -p 3306:3306 --name
mysql2 -e MYSQL_ROOT_PASSWORD=root -d mysql:5.7
0bb87bcab1cfcb6b0f2babc31e14013bde0851ca1e5e66a02a7048cb3b29cfe0
root@CloudComputer:/home/admin/桌面/mydock#
```

启动mysql客户端容器并允许mysql客户端

```
docker run -it --net host mysql:5.7 "sh"
```



同时可以访问容器，查看数据库存放位置等等。

文章分享

[这可能是最为详细的Docker入门吐血总结]