# Docker 简明文档

## Docker安装

由于apt官方库里的docker版本可能比较旧，所以先卸载可能存在的旧版本：

$ sudo apt-get remove docker docker-engine docker-ce docker.io

更新apt包索引：

$ sudo apt-get update

安装以下包以使apt可以通过HTTPS使用存储库（repository）：

$ sudo apt-get install -y apt-transport-https ca-certificates curl software-properties-common

添加Docker官方的GPG密钥：

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

使用下面的命令来设置stable存储库：

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

再更新一下apt包索引：

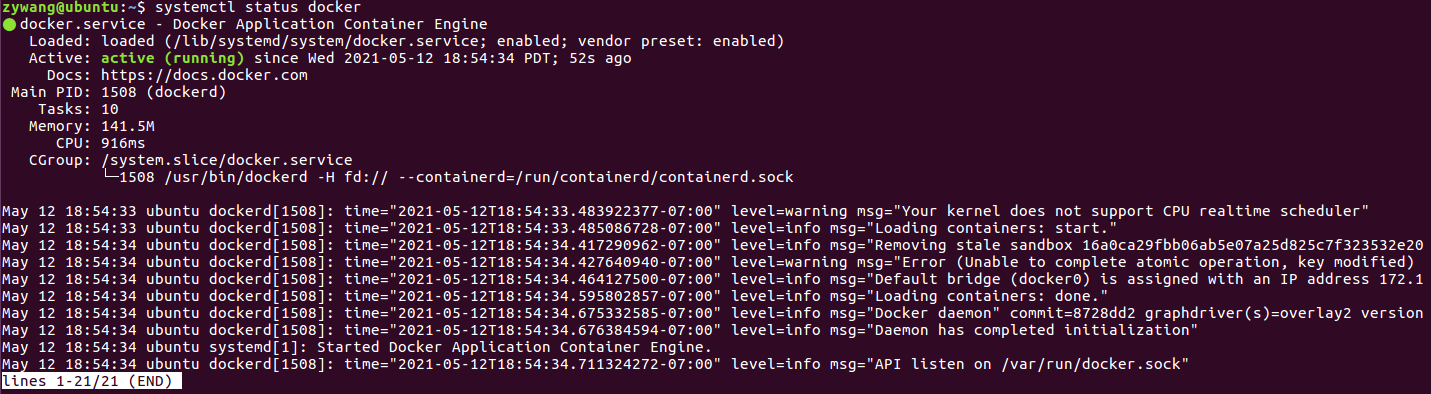
$ sudo apt-get update

安装最新版本的Docker CE：

$ sudo apt-get install -y docker-ce

验证docker查看docker服务是否启动：

$ systemctl status docker



如果出现上图，就表示成功了

若未启动，则启动docker服务：

$ sudo systemctl start docker

启动后：

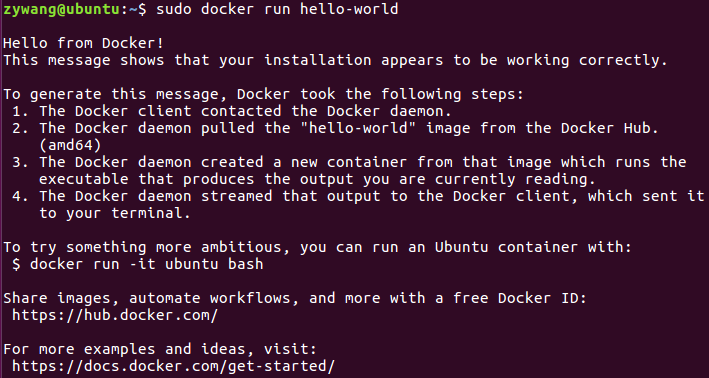
$ sudo docker run hello-world

有以下图片内容输出,表示docker安装成功

如果失败，可能由于系统时钟不同步引起

$ sudo apt install ntpdate

$ sudo docker run hello-world



## 二．docker使用

查找Ubuntu镜像

$ sudo docker search ubuntu

从搜索到的镜像里选择一个，安装Ubuntu镜像

$ sudo docker pull ubuntu

查看docker镜像

$ sudo docker images

运行docker镜像

$ sudo docker run -it -d --name ubuntu\_test -p 8088:80 ubuntu

-name自定义容器名，-p指定端口映射，前者为虚拟机端口，后者为容器端口,成功后返回容器id

查看所有启动的容器(查看所有容器加 $ sudo docker ps -a)

$ sudo docker ps

进入容器(或者把容器id改为容器名，也可以进入)

$ sudo docker exec -it ubuntu\_test /bin/bash

**容器里面也是ubuntu虚拟机，同样适用Linux命令，在这里你可以部署你想要的一切**

退出容器

$ exit

停止容器

$ sudo docker stop id

## 三．镜像制作、删除、打包与导入

制作docker镜像

$ sudo docker commit ubuntu\_test ubuntu\_test1:1.0 （容器名 镜像名:TAG）

1.0为版本号，需要添加，镜像名字随意

此时镜像只能本地使用，在其他机器使用需打包

$ sudo docker save -o ubuntu\_test1.tar ubuntu\_test1:1.0

将制作的镜像包上传至服务器，假设你在新的服务器上已经安装好了docker，那么使用以下命令将我们刚才上传的压缩包导入为镜像：

# 导入方法一

$ docker load --input ubuntu\_test1.tar (包前可以加包所在的目录)

# 导入方法二

$ docker load -i ubuntu\_test1.tar

# 导入方法三

$ docker load < ubuntu\_test1.tar

使用ubuntu\_test1:latest镜像创建并启动一个容器，并以shell的交互模式进入容器内，使用exit退出容器后容器停止运行

docker run -it ubuntu\_test1:latest

使用ubuntu\_test1:latest镜像创建并启动一个容器，但是不进入容器内部，而是在后台运行

docker run -itd ubuntu\_test1:latest

本地镜像删除

docker rmi 镜像id

容器删除

docker rm 容器id

删除失败场景1：

如果删除镜像失败，根据提示信息会告诉你该镜像正在被某个容器所使用，所以必须先删除容器，再删除镜像

删除失败场景2：

在删除时还可能遇到如下情况，多个镜像id相同，此时删除也会不成功，此时我们还可以根据"REPOSITORY"和"TAR"进行删除： sudo docker rmi REPOSITORY:TAG

## 四．镜像上传DockerHub

1.注册自己的Docker hub账号，<https://hub.docker.com/>

2.注册后账号在本地登录

$ sudo docker login

账号：zywang

密码：xxxxxx

3.先sudo docker images 查看本地镜像

如果REPOSITORY的名字不是docker账号/仓库名 zywang/spice3f5,上传会失败

可以通过

$ sudo docker tag 镜像ID 账号/仓库名：新TAG 来更改

4.上传

$ sudo docker push 账号/仓库名:TAG

5.登录到自己的DockersHub去验证