**Morning：**

1、Docker概述

**-What’s容器？**

容器：应用程序封装和交付的核心技术；实质为使用了隔离技术的一个进程。

容器技术的核心：Cgroups-资源管理、NameSpace-进程管理、SELinux；

Docker为一个容器管理系统。

**-容器与虚拟化的比较**

虚拟化需要装操作系统，容器使用的是共享公共库和程序，更加简洁高效；

容器隔离性没有虚拟化强，共用Linux内核，一个容器进程崩溃，所有崩溃； 容器SELinux难以驾驭，难以完成监控以及排错。

1. Docker的部署

**-Docker的部署**

1）需要64位OS，至少RHEL7以上(内核3.0)，关闭防火墙(非必须)

2）准备2台虚拟机(2核4G内存20G硬盘扩容)

3）IP地址：192.168.1.10(docker01)、192.168.1.11(docker02)

4） 配置/etc/hosts、自定义yum源(docker-engine、docker-en..-selinx)

5）安装yum -y install docker-engine

6）启动服务systemctl start docker

7）版本验证docker version

**-What’s镜像**

1）启动容器的核心(相当于后端盘)；

2）采用分层设计(每次将前端&后端盘进行打包，作为新的后端盘)；

3）使用快照的COW技术(读取时，从上往下层读取)。

**-关于Docker的小知识**

1）Docker六大名字空间：

主机名、网络、用户、进程、根目录、信号向量(进程PID与真机隔离)

2）如何证明容器不是虚拟机?

-没有上帝进程systemd(1)，systemctl -a没有进程

-容器启动进程，可以在宿主机ps查到，宿主机也可对容器进程进行干预

**Afternoon：**

1、Docker常用命令

**-关于Docker的问题**

1）ubuntu镜像没有yum怎么办？

RHEL与Debian流派的区别：

yum --> apt-get

rpm --> dpkg

2）为什么centos镜像指定网络yum源之后可以直接访问？

**-镜像常用操作**

|  |  |
| --- | --- |
| 命令 | 备注 |
| docker images | 查看本地镜像 |
| docker history 镜像:标签 | 查看镜像制作过程(多层) |
| docker inspect 镜像:标签 | 查看镜像详细信息(PATH、CMD) |
| docker search 镜像名  docker pull 镜像名  docker push 镜像:标签 | 搜索服务相关镜像  下载镜像  上传镜像(需要官网帐号) |
| docker rmi 镜像:标签 | 删除本地镜像(必须先把容器删除) |
| docker save 镜:标 > xx.tar | 镜像的备份/导出 |
| docker load < xx.tar | 镜像的恢复/导入 |
| docker tag 镜:标 (新)镜:标 | 给镜像添加别名 |

相关知识点：

1、镜像的唯一标识是IMAGE ID。文件的唯一ID是绝对路径或inode节点

2、[root@docker02 ~]# file busybox.tar #查看导出文件的类型

**-容器常用操作**

|  |  |
| --- | --- |
| 命令 | 备注 |
| docker run -itd 镜像:标签 容器命令 | 基于镜像创建一个新容器  #-i交互 -t分配伪终端 -d后台  #不写命令使用默认容器命令 |
| docker ps [-a] | 查看容器运行列表[-a 列出所有] |
| docker stop 容器ID  docker start 容器ID  docker restart 容器ID | 关闭容器  启动容器  重启容器 |
| docker inspect 容器ID | 查看容器详细信息 |
| docker top 容器ID | 查看容器进程列表 |
| docker rm 容器ID | 删除容器(容器先停止) |
| docker attach -it 容器ID /bin/bash  docker exec -it 容器ID /bin/bash | 进入容器，exit会关闭容器  进入容器，exit不会关闭容器 |

注意事项：

1）容器ID只要输入前几位即可(保证唯一)

2）docker inspect -f '{{.NetworkSettings.IPAddress}}' 1fd

-f对容器详细信息进行过滤，常用于脚本

3）删除所有容器：docker rm $(docker stop $(docker ps -aq))

4）关于attach与exec：

#attach相当于使用console进入容器(挂在上帝进程下)

#exec相当于打开新的bash子进程进入容器

#使用attach如何退出终端(Ctrl p + Ctrl q)

作业：

修改Apache及Nginx的主页内容为hello nsd1808。

1. 启动一个centos容器，安装apache
2. [root@8958 ~]# cat /usr/lib/systemd/system/httpd.service

[root@8958 ~]# /usr/sbin/httpd $OPTIONS -DFOREGROUND

Ctrl p + Ctrl q

#查看httpd的启动命令，退出终端

3）Debian系统如何查看nginx的网页文件？

方法1： dpkg -l #列出所有包

dpkg -L nginx #查看nginx包的文件目录信息

方法2： nginx -T #查看nginx配置

Tips：/usr/lib/systemd/system/xx.service #xx为服务名

[Unit] #描述信息

Description=The Apache HTTP Server

After=network.target remote-fs.target nss-lookup.target

Documentation=man:httpd(8)

Documentation=man:apachectl(8)

[Service]

Type=notify

EnvironmentFile=/etc/sysconfig/httpd #记录相关参数

ExecStart=/usr/sbin/httpd $OPTIONS -DFOREGROUND #启动命令

ExecReload=/usr/sbin/httpd $OPTIONS -k graceful #重启命令

ExecStop=/bin/kill -WINCH ${MAINPID} #停止命令

KillSignal=SIGCONT

PrivateTmp=true

[Install] #运行等级

WantedBy=multi-user.target