

大型架构及配置技术

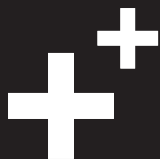
NSD

ARCHITECTURE

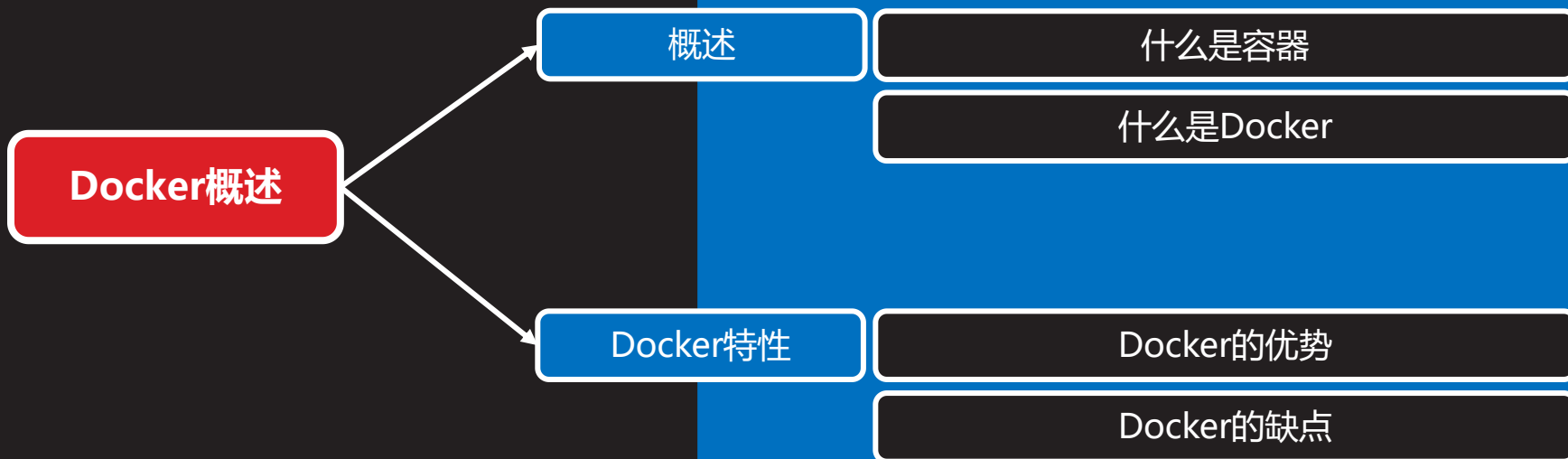
DAY01

内容

上午	09:00 ~ 09:30	Docker概述
	09:30 ~ 10:20	
	10:30 ~ 11:20	部署Docker
	11:30 ~ 12:00	Docker镜像
下午	14:00 ~ 14:50	
	15:00 ~ 15:50	Docker基本命令
	16:10 ~ 17:00	
	17:10 ~ 18:00	总结和答疑



Docker概述



概述

什么是容器

- 容器技术已经成为应用程序封装和交付的核心技术
- 容器技术的核心有以下几个内核技术组成：
 - CGroups(Control Groups)-资源管理
 - NameSpace-进程隔离
 - SELinux安全
- 由于是在物理机上实施隔离，启动一个容器，可以像启动一个进程一样快速



什么是Docker

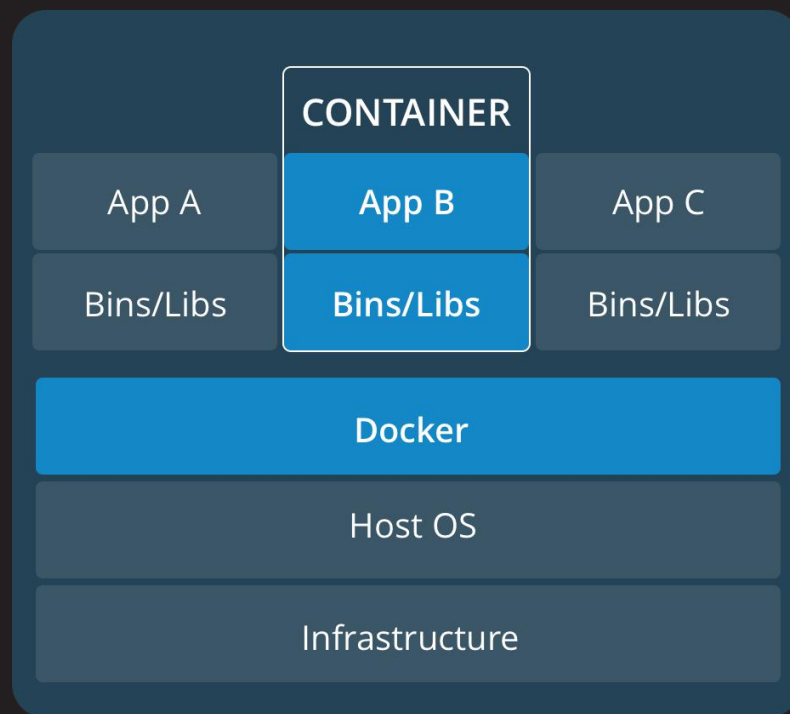
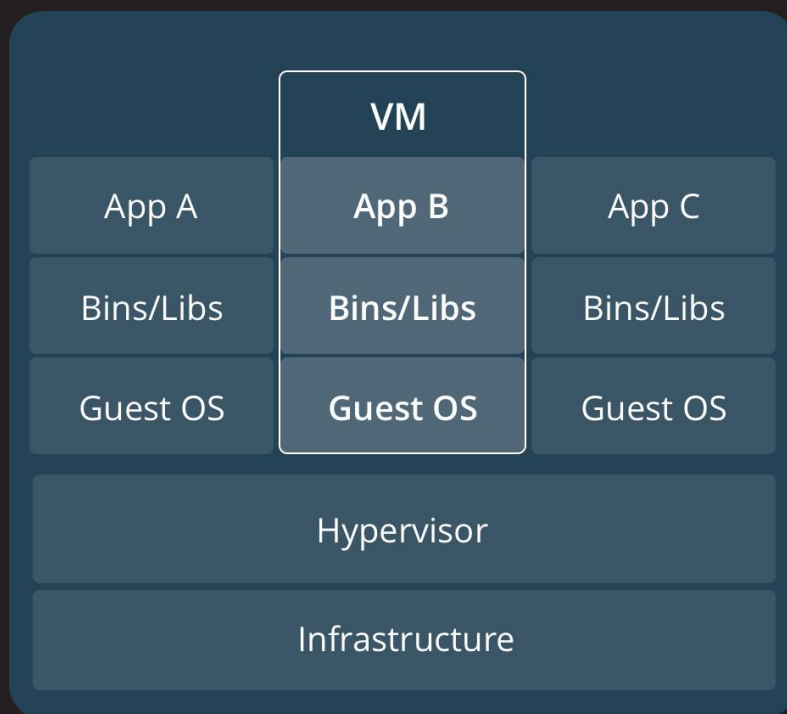
- Docker是完整的一套容器管理系统
- Docker提供了一组命令，让用户更加方便直接地使用容器技术，而不需要过多关心底层内核技术



Docker特性

Docker优点

- 相比于传统的虚拟化技术，容器更加简洁高效
- 传统虚拟机需要给每个VM安装操作系统
- 容器使用的共享公共库和程序



Docker的缺点

- 容器的隔离性没有虚拟化强
- 共用Linux内核，安全性有先天缺陷
- SELinux难以驾驭
- 监控容器和容器排错是挑战



部署Docker

部署Docker

安装软件

安装前准备

安装Docker

安装软件



安装前准备

- 需要64位操作系统
- 至少RHEL6.5以上的版本，强烈推荐RHEL7
- 关闭防火墙（不是必须）



安装Docker

- 软件包列表：
 - docker-engine
 - docker-engine-selinux

```
[root@jacob ~]# rpm -ivh docker-engine-1.12.1-1.el7.centos.x86_64.rpm  
[root@jacob ~]# rpm -ivh docker-engine-selinux-1.12.1-  
1.el7.centos.noarch.rpm
```

```
[root@jacob ~]# systemctl start docker  
[root@jacob ~]# systemctl enable docker
```



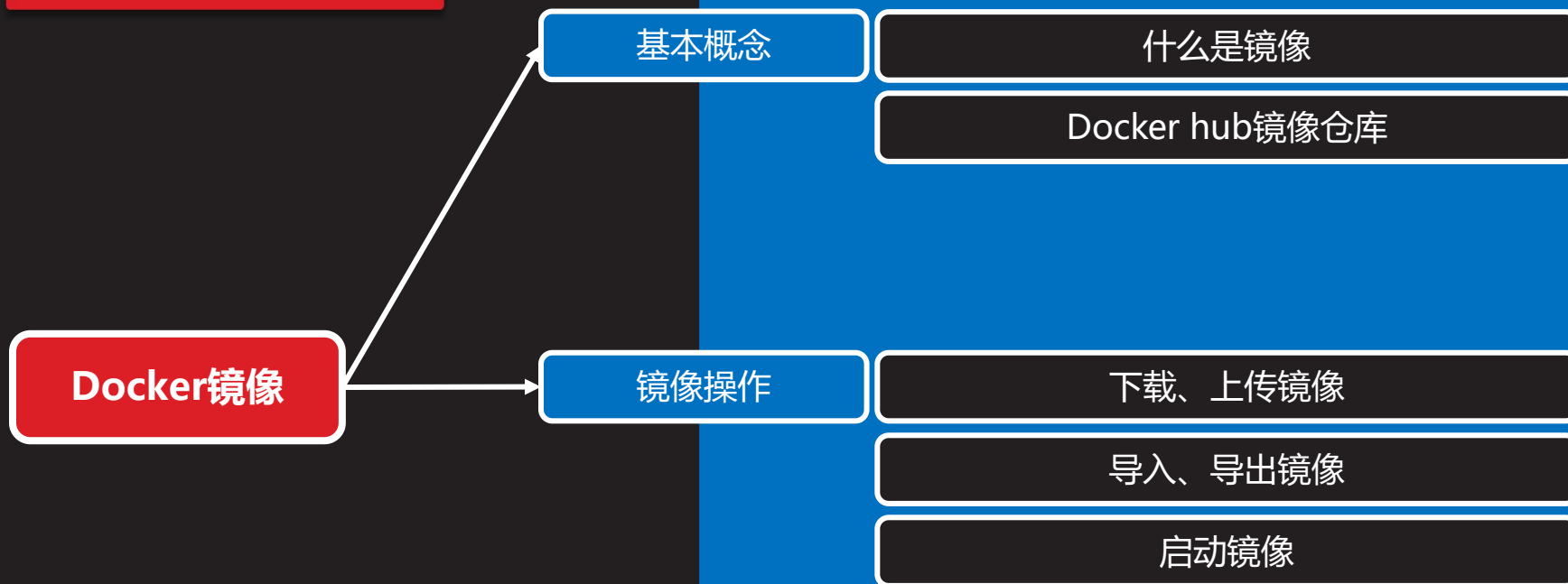
案例1：安装Docker

安装docker平台所需要的软件

- docker-engine
- docker-engine-selinux
- 关闭防火墙



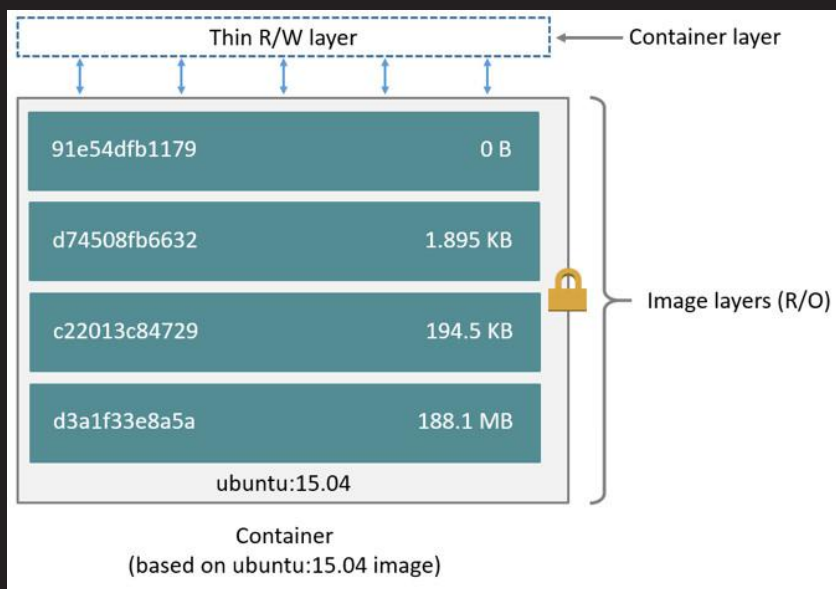
Docker镜像



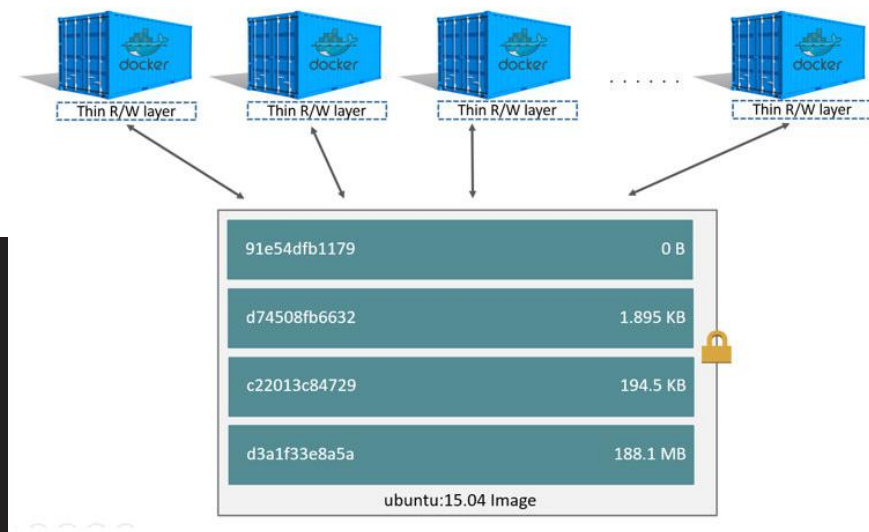
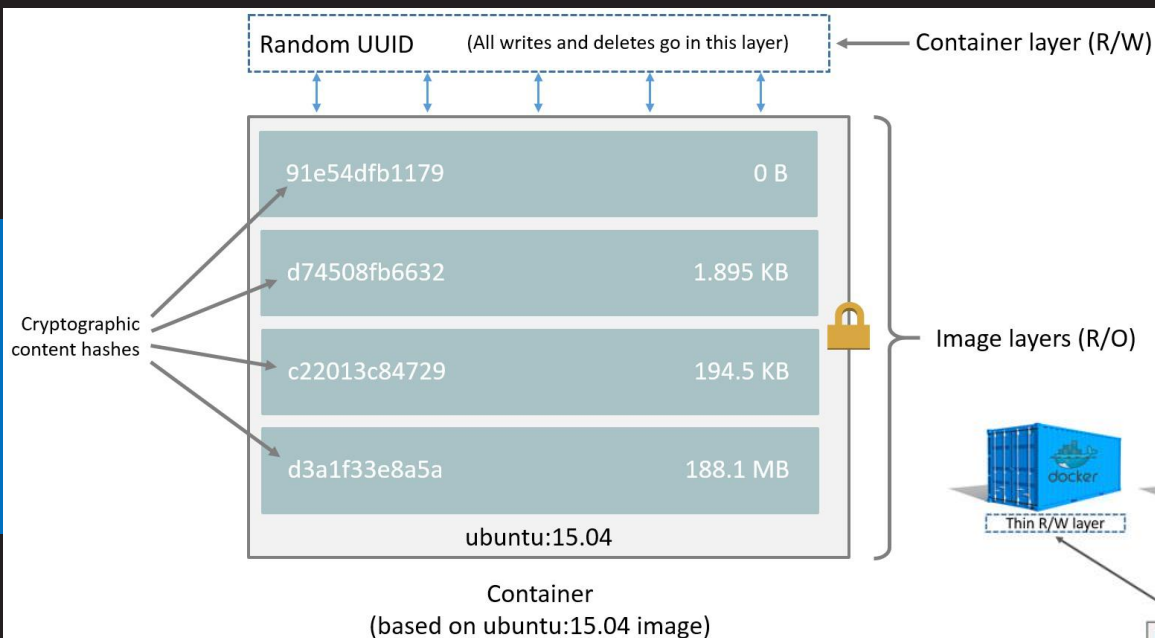
基本概念

什么是镜像

- 在Docker中容器是基于镜像启动的
- 镜像是启动容器的核心
- 镜像采用分层设计
- 使用快照的COW技术，确保底层数据不丢失



什么是镜像 (续1)



```
[root@jacob ~]# docker images
```

Docker hub镜像仓库

- <https://hub.docker.com>
- Docker官方提供公共镜像的仓库(Registry)

```
[root@jacob ~]# docker search rhel7  
[root@jacob ~]# docker search centos  
[root@jacob ~]# docker search nginx  
[root@jacob ~]# docker search mysql
```

镜像操作

下载、上传镜像

- 下载镜像（从镜像仓库中下载镜像）

```
[root@server0 ~]# docker help pull
docker pull [OPTIONS] NAME[:TAG|@DIGEST]
```

```
[root@server0 ~]# docker pull rhel7
```

- 上传镜像（上传镜像到仓库）

```
[root@server0 ~]# docdocker help push
Usage: docker push [OPTIONS] NAME[:TAG]
```

```
[root@server0 ~]# docker push rhel7
```



导入、导出镜像

- 导入镜像（通过tar包文件导入镜像）

```
[root@server0 ~]# docker load < xx.tar  
[root@server0 ~]# docker images
```

- 导出镜像（将本地镜像导出为tar文件）

```
[root@server0 ~]# docker images  
[root@server0 ~]# docker save image_name > xx.tar
```



启动镜像

- 启动centos镜像生成一个容器

```
[root@server0 ~]# docker images
```

```
[root@server0 ~]# docker run -it centos bash
```

- 开启另一个终端（查看容器信息）

```
[root@server0 ~]# docker ps
```



案例2：镜像基本操作

镜像基本操作：

- 导入镜像 (centos、nginx、mysql)
- 导出镜像为tar包文件 (方便其他主机使用镜像)
- 启动centos、nginx镜像



Docker基本命令

Docker基本命令

镜像常用命令

命令列表

docker images

docker history

docker inspect

docker rmi

docker save|load

docker tag

容器常用命令

命令列表

docker run

docker ps

docker stop|start|restart

docker attach|exec

docker inspect

docker top

docker rm

镜像常用命令

命令列表

- 命令列表

- docker images

//查看镜像列表

- docker history

//查看镜像制作历史

- docker inspect

//查看镜像底层信息

- docker pull

//下载镜像

- docker push

//上传镜像

- docker rmi

//删除本地镜像

- docker save

//镜像另存为tar包

- docker load

//使用tar包导入镜像

- docker search

//搜索镜像

- docker tag

//修改镜像名称和标签



docker images

- 查看镜像列表
 - 镜像仓库名称
 - 镜像标签
 - 镜像ID
 - 创建时间
 - 大小

```
[root@jacob ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
centos	latest	980e0e4c79ec	8 months ago	196.7 MB



docker history

- 查看镜像历史
 - 了解镜像制作过程
 - 详细参考后面的dockerfile内容

```
[root@jacob ~]# docker history centos
```

IMAGE	CREATED	CREATED BY	SIZE	COMMENT
980e0e4c79ec	8 months ago	/bin/sh -c #(nop) CMD ["/bin/bash"]	0 B	
<missing>	8 months ago	/bin/sh -c #(nop) LABEL name=CentOS Base Ima	0 B	
<missing>	8 months ago	/bin/sh -c #(nop) ADD file:e336b45186086f7d9d	196.7 MB	
<missing>	8 months ago	/bin/sh -c #(nop) MAINTAINER https://github.	0 B	



docker inspect

- 查看镜像底层信息
 - 了解镜像环境变量、存储卷、标签等信息

```
[root@jacob ~]# docker inspect centos
```

```
... ..
{
  "Id": "sha256:980e0e4c79ec933406e467a296ce3b86685e6b42eed2f873745e6a91d718e37a",
  "RepoTags": [
    "centos:latest"
  ],
  "RepoDigests": [],
  "Parent": "",
  "Comment": "",
  "Created": "2016-09-06T21:10:20.397787682Z",
  "Container": "37446a1771cbec3e85b76d9159fd6a5a92743655cb92a65661e8a174bad81c7e "
  ... ..
```



docker rmi

- 删除本地镜像
 - 注意：启动容器时删除镜像会提示错误

```
[root@jacob ~]# docker rmi centos
```

```
Error response from daemon: conflict: unable to remove repository reference  
"centos" (must force) - container 3daba799135c is using its referenced image  
980e0e4c79ec
```



docker save|load

- 保存本地镜像另存为tar文件
 - 方便其他人使用tar包导入镜像

```
[root@jacob ~]# docker save centos > centos.tar  
[root@jacob ~]# ls centos.tar  
centos.tar
```

- 使用tar包文件导入镜像

```
[root@jacob ~]# docker load < centos.tar
```



docker tag

- 重命名镜像名称（复制）

```
[root@jacob ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
centos	latest	980e0e4c79ec	8 months ago	196.7 MB

```
[root@jacob ~]# docker tag centos:latest test:v1
```

```
[root@jacob ~]# docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
centos	latest	980e0e4c79ec	8 months ago	196.7 MB
test	v1	980e0e4c79ec	8 months ago	196.7 MB



容器常用命令

命令列表

- 命令列表
 - docker run //运行容器
 - docker ps //查看容器列表
 - docker stop //关闭容器
 - docker start //启动容器
 - docker restart //重启容器
 - docker attach|exec //进入容器
 - docker inspect //查看容器底层信息
 - docker top //查看容器进程列表
 - docker rm //删除容器



docker run

- 使用镜像启动容器

```
[root@jacob ~]# docker run -it centos bash  
[root@bbb51ac87b08 /]#exit
```

```
[root@jacob ~]# docker run -itd centos bash  
b8f218f2341c12655e6092d7d2e7fd5229824fdefce84075fb3a9569ebf82079  
[root@jacob ~]# docker ps
```



docker ps

- 列出容器列表

- docker ps

查看正在运行的容器

- docker ps -a

查看所有容器列表

- docker ps -aq

仅显示容器id

[root@server0 ~]# docker ps

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
b8f218f2341c	centos	"bash"	9 minutes ago	Up 8 minutes		jovial_pasteur



docker stop|start|restart

- 管理容器

- docker stop 关闭容器
- docker start 开启容器
- docker restart 重启容器

```
[root@server0 ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
b8f218f2341c	centos	"bash"	9 minutes ago	Up 8 minutes		jovial_Pasteur

```
[root@server0 ~]# docker stop b8f
```



docker attach|exec

- 进入容器

- docker attach

进入容器，exit会导致容器关闭

- docker exec

进入容器，退出时不会关闭容器

```
[root@server0 ~]# docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
b8f218f2341c	centos	"bash"	9 minutes ago	Up 8 minutes		jovial_Pasteur
7e4e51117c40	centos	"bash"	5 seconds ago	Up 2 seconds		pedantic_bard

```
[root@server0 ~]# docker attach b8f
```

```
[root@server0 ~]# docker exec -it 7e4e bash
```



docker inspect

- 查看容器底层信息

[root@server0 ~]# docker ps

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
b8f218f2341c	centos	"bash"	9 minutes ago	Up 8 minutes		jovial_Pasteur
7e4e51117c40	centos	"bash"	5 seconds ago	Up 2 seconds		pedantic_bard

[root@server0 ~]# docker inspect 7e4e

```
"Id": "0cc385eb6f63373397e678347890914347dc8efa3cefd809c1ed8c157651a261",
"Gateway": "172.17.0.1",
"IPAddress": "172.17.0.2",
"MacAddress": "02:42:ac:11:00:02"
```



docker top

- 查看容器进程列表

[root@server0 ~]# docker ps

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
b8f218f2341c	centos	"bash"	9 minutes ago	Up 8 minutes		jovial_Pasteur
7e4e51117c40	centos	"bash"	5 seconds ago	Up 2 seconds		pedantic_bard

[root@server0 ~]# docker top 7e4e

UID	PID	PPID	C	STIME	TTY	TIME	CMD
root	3177	3163	0	21:25	pts/8	00:00:00	/bin/bash



docker rm

- 删除容器
 - 注意，删除正在运行的容器时会提示错误

```
[root@server0 ~]# docker rm 7e4e
```

```
[root@server0 ~]# docker rm `docker ps -aq`
```



案例3：镜像与容器常用指令

1. 镜像常用指令练习
2. 容器常用指令练习



总结和答疑

总结和答疑

删除容器

问题现象

故障分析及排除

删除容器

问题现象

- 删除容器，提示错误：

- docker rm 0cc

Error response from daemon: You cannot remove a running container

0cc385eb6f63373397e678347890914347dc8efa3cef
d809c1ed8c157651a261. Stop the container before
attempting removal or use -f



故障分析及排除

- 原因分析
 - 问题1：提示error：You cannot remove a running container
- 解决办法
 - 问题1：先关闭容器，才能删除容器

