

2021.3.9 Docker 容器操作

容器是 Docker 的另一个核心概念，简单说，容器是镜像的一个运行实例，是独立运行的一个或一组应用以及它所必须的运行环境，包括文件系统，系统类库，shell 环境等，镜像只是只读模板，而容器会给这个模板一个额外的可写层

下面将具体的介绍围绕容器的具体操作：

· 创建容器与运行容器

Docker 的创建就是将镜像加载到容器的过程，Docker 的容器**十分轻量级**，用户可以随时创建或者删除。新创建的容器默认处于停止状态，不运行任何程序，需要在其中发起一个进程来启动容器，这个进程是该容器的唯一进程，所以当该进程结束的时候，容器也会完全停止。停止的容器可以重新启动并保留原来的修改。可以使用 `docker create` 命令新建一个容器。

命令格式：`docker create[选项]... 镜像运行的程序`

常用选项：

- i ：让容器的输入保持打开
- t ：让 docker 分一个伪终端
- d ：守护进程形式运行

第一点注意：在生产环境中，每开启一个容器都要加上 `--restart=always`，避免 docker 服务重启时，容器自动停止运行

```
[root@server04 ~]# docker create -it nginx /bin/bash #创建一个 nginx 镜像
c9d2123030e1a562196b583da5c023875320915d9953ae056759bf857f62d548
```

注意：如果创建容器命令报错的话，很有可能是 ipv4 的转发没有开启

解决方法：开启 ipv4，修改文件 `/etc/sysctl.conf`

```
[root@server04 ~]# vim /etc/sysctl.conf
net.ipv4.ip_forward = 1      #表示开启转发
[root@server04 ~]# sysctl -p  #查看状态
net.ipv4.ip_forward = 1
```

Docker 容器基本命令

```
Docekr ps -a      #查看所有的容器
Dooker start c4f45c4cf094 #介于 ID 号启动一个容器
Dooker stop c4f45c4cf094 #介于 ID 号关闭一个容器
Docekr ps -l      #列出正在运行的容器
docker run -it nginx /bin/bash #创建并运行一个容器（注意提示符会发生改变）
注意点：root@f0d0e8d1b365:/# exit 退出（如果使用 exit 退出容器，那么容器也会被停止）
快捷键退出：ctrl +pq（这种方式退出时，容器不会停止工作）
[root@server04 ~]# docker logs 345 #查看容器的输出的信息
```

·使用 `docker` 命令进入容器的方式有两种：

第一种：格式 `docker exec -it ID号 /bin/bash`

这种方法需要指定容器的伪终端

```
[root@server04 ~]# docker exec -it 345f9bafb921 /bin/bash #这种方式就是要给容器指定一个伪终端
```

第二种：格式 `docker attach 容器ID`

这种方法需要查看容器有没有伪终端，有的话可以进入，没有则相反

```
[root@server04 ~]# docker attach 345f9bafb921 #这种方法需要查看容器有没有伪终端
```

4. 容器的导入和导出

用户可以将任何一个 Docker 容器从一台机器迁移到另一台机器。在迁移过程中，首先需要将已经创建好的容器导出为文件，可以使用 `docker export` 命令实现，无论这个容器是处于运行状态还是停止状态均可导出。导出之后可将导出文件传输到其他机器，通过相应的导入命令实现容器的迁移。

命令格式：`docker export 容器ID/名称>文件名`

容器的导出：

格式：`docker export 容器ID > 导出的文件名`

```
[root@server04 ~]# docker export 345f9bafb921 > xxx.tar #导出容器，导出文件名为 xxx.tar
```

容器的导入：

格式：`cat 文件名 | docker import - 导入名 (注意空格)`

注意：容器的导入是以镜像的形式导入进去的

```
[root@server04 ~]# cat xxx.tar | docker import - lnmp:v1 #导入容器
```

```
[root@server04 ~]# docker images #查看是否导入成功
```

```
lnmp          v1          e3b22a86f326  51 seconds ago  131MB
```

```
[root@server04 ~]# docker run -it lnmp:v1 /bin/bash #进入到容器种
```

```
root@d0a08812adab:/# ls #查看有没有你想要的内容
```

```
1.txt  boot  docker-entrypoint.d  etc  lib  media  opt  root  sbin  sys  usr
bin    dev   docker-entrypoint.sh  home lib64 mnt    proc  run   srv   tmp  var
```

5) 拷贝文件到容器中

格式：`docker cp 文件名 容器ID 存放位置`

```
[root@server04 ~]# docker cp 2.txt d0a08812adab:tmp #将文件复制到/tmp 下
可以进入容器查看一下是否复制成功
```

6) 删除容器

格式：`docker rm 容器ID/名称`

```
[root@server04 ~]# docker rm c9d2123030e1 #删除一个容器
```

7) 容器的开启，停止，运行，重启，

格式：`docker |start|stop|restart| kill` ： 开启，停止，重启，杀死

```
[root@server04 ~]# docker stop c9d2123030e1 #停止一个容器
```


