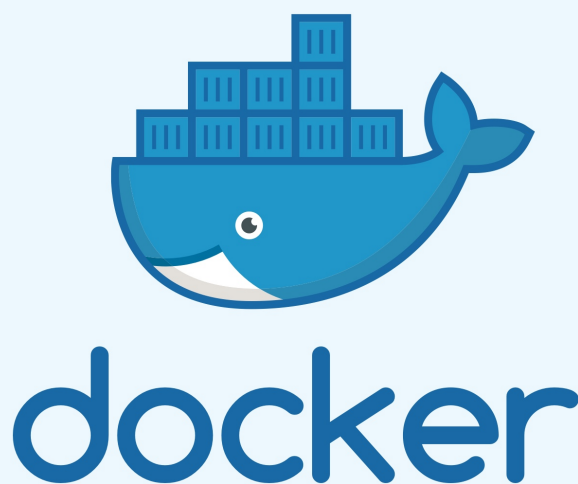


八个基本的 Docker 容器管理命令

在本文中，我将介绍八个基本的 Docker 容器命令，这些命令对于在 Docker 容器上执行基本操作很有用，比如运行，列表，停止，查看日志，删除等等。如果你对 Docker 的概念不熟悉，推荐你推荐你到网上查看相关的入门介绍，这篇文章就不详细介绍了。现在我们赶快进入要了解的命令中：



如果想及时了解Spark、Hadoop或者HBase相关的文章，欢迎关注微信公众号：iteblog_hadoop

如何运行 Docker 容器？

众所周知，Docker 容器只是一个运行在宿主操作系统上的应用程序，所以对于一个 Docker 来说，你需要一个镜像（image）来运行它。当 Docker 镜像使用进程的方式运行时，我们称之为 Docker 容器（Docker container）。你可以在本地获取 Docker 镜像，也可以从 Docker Hub 上下载。Docker Hub 是一个存储公有和私有镜像的集中仓库。官方的 Docker Hub 位于 hub.docker.com。当你使用 Docker 引擎运行容器时，它首先会搜索本地镜像，如果本地没有找到，则会从 Docker Hub 上拉取相关的镜像。

让我们运行一个 Apache web 服务器的 Docker 镜像（比如 httpd 进程）。你需要使用 docker container run 命令。旧的命令是 docker run，但因为 Docker 后来添加了子命令部分，所以新版本支持下面的命令：

```
root@kerneltalks # docker container run -d -p 80:80 httpd
Unable to find image 'httpd:latest' locally
latest: Pulling from library/httpd
3d77ce4481b1: Pull complete
73674f4d9403: Pull complete
d266646f40bd: Pull complete
ce7b0dda0c9f: Pull complete
01729050d692: Pull complete
014246127c67: Pull complete
7cd2e04cf570: Pull complete
Digest: sha256:f4610c3a1a7da35072870625733fd0384515f7e912c6223d4a48c6eb749a8617
Status: Downloaded newer image for httpd:latest
c46f2e9e4690f5c28ee7ad508559ceee0160ac3e2b1688a61561ce9f7d99d682
```

Docker 的 run 命令中的镜像名是必须需要输入的，同事还支持很多可选参数。常用的参数有：

- -d：从当前 shell 分离容器
- -p X:Y：将容器的端口 Y 与主机的端口 X 进行绑定
- --name：给你的容器取名。如果未设定，它将设置成一个随机生成的名字
- -e：在启动容器时传递环境变量及其值

从上面的输出可以看到，我们使用 httpd 作为映像名称来运行容器。因为在本地没有找到这个镜像，Docker 引擎从 Docker Hub 将它拉取下来。请注意，Docker 引擎下载了镜像 httpd:latest，其中：后面跟着的是版本号，这是 Docker 容器映像的命名约定。如果你想要运行特定版本的容器，你可以在镜像名后面标明相关的版本号。如果没有提供版本号，Docker 引擎将始终拉取最新的版本。

最后一行输出显示了新运行 httpd 容器的唯一容器标识。

如何列出所有运行的 Docker 容器？

现在，你的容器正在运行，你可能需要检查它，或者你想要列出机器上所有运行的容器。你可以使用 docker container ls 命令来达到这个目的。在旧的 Docker 版本中，你需要使用 docker ps 命令来实现。

```
root@kerneltalks # docker container ls
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAME
--------------	-------	---------	---------	--------	-------	------

```
S
c46f2e9e4690 httpd "httpd-
foreground" 11 minutes ago Up 11 minutes 0.0.0.0:80->80/tcp cranky_cori
```

结果是按列进行显示的，每一列的值的含义分别为：

- Container ID ：容器唯一 ID 的前 12 个字母
- Image ：运行容器的镜像名
- Command ：容器启动后运行的命令
- Created ：创建时间
- Status ：容器的当前状态
- Ports ：与主机端口绑定的详细信息
- Names ：容器的名称（因为我们启动容器的时候并没有指定名称，所以这里显示的是随机生成的名字）

如何查看 Docker 容器的运行日志？

由于在第一步中我们使用了 -d 参数来将容器从当前 shell 分离出来，这样它将会在后台运行。在这种情况下，我们不清楚容器里面发生了什么，所以需要查看容器的运行日志。Docker 为我们提供了 logs 命令，它将容器名称或ID作为参数。

```
root@kerneltalks # docker container logs cranky_cori
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using
172.17.0.2. Set the 'ServerName' directive globally to suppress this message
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using
172.17.0.2. Set the 'ServerName' directive globally to suppress this message
[Thu May 31 18:35:07.301158 2018] [mpm_event:notice] [pid 1:tid 139734285989760] AH0048
9: Apache/2.4.33 (Unix) configured -- resuming normal operations
[Thu May 31 18:35:07.305153 2018] [core:notice] [pid 1:tid 139734285989760] AH00094: Com
mand line: 'httpd -D FOREGROUND'
```

这里我在命令里面使用了容器名称来作为参数。你可以 httpd 容器中查看到 Apache 相关的日志。

如何识别 Docker 容器的进程？

容器是一个使用宿主资源来运行的进程。如果这是真的，那么你可以在宿主上的进程表中找到容器的进程。让我们看看如何检查宿主上的容器进程。

Docker 使用著名的 top 命令作为其子命令的名称来查看容器产生的进程。它使用容器名称或 ID 作为参数。在旧版本的 Docker 中，只能使用 docker top 命令。然而在新版本中，我们可以使用 docker top 和 docker container top 命令。

```
root@kerneltalks # docker container top cranky_cori
UID      PID      PPID     C STIME   TTY      TIME    CMD
root     15702    15690    0 18:35   ?        00:00:00 httpd -DFOREGROUND
bin      15729    15702    0 18:35   ?        00:00:00 httpd -DFOREGROUND
bin      15730    15702    0 18:35   ?        00:00:00 httpd -DFOREGROUND
bin      15731    15702    0 18:35   ?        00:00:00 httpd -DFOREGROUND
root@kerneltalks # ps -ef |grep -i 15702
root     15702 15690 0 18:35 ?      00:00:00 httpd -DFOREGROUND
bin      15729 15702 0 18:35 ?      00:00:00 httpd -DFOREGROUND
bin      15730 15702 0 18:35 ?      00:00:00 httpd -DFOREGROUND
bin      15731 15702 0 18:35 ?      00:00:00 httpd -DFOREGROUND
root     15993 15957 0 18:59 pts/0  00:00:00 grep --color=auto -i 15702
```

第一个输出里面列出该容器产生的进程列表。它包含了所有信息，包括 uid、pid，ppid、开始时间、启动命令等等。这里显示的所有进程号你都可以在宿主的进程表里搜索到。这就是我们在第二个命令中所做的。所以，这证明容器实际上只是宿主主机上的进程。

如何停止 Docker 容器？

请使用 stop 命令！同样，它采用容器名称或 ID 作为参数。

```
root@kerneltalks # docker container stop cranky_cori
cranky_cori
```

如何列出已停止或未运行的 Docker 容器？

现在我们停止了我们的容器，如果我们尝试使用 ls 命令列出容器，我们将无法在里面看到它。

```
root@kerneltalks # docker container ls
CONTAINER ID  IMAGE  COMMAND  CREATED  STATUS  PORTS  NAMES
```

因此在这种情况下，如果想查看停止或未运行的容器，你需要在 ls 命令里同时使用 -a

参数，如下：

```
root@kerneltalks # docker container ls -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
c46f2e9e4690	httpd	"httpd-foreground"	33 minutes ago	Exited (0) 2 minutes ago		cranky_cori

使用了 -a 参数之后，我们就可以查看已停止的容器。请注意这个容器的状态被标注为“已退出”（Exited）。由于容器只是一个进程，使用已退出比使用停止比较合适。

如何启动 Docker 容器？

现在让我们来启动这个已停止的容器，这和运行容器还是有所区别的。当你运行 (run) 容器时，你将在一个全新的容器里面运行命令。当你启动 (start) 一个容器时，你正在启动一个已经保存旧状态的旧的已停止的容器。它将以停止时的状态重新开始运行。

```
root@kerneltalks # docker container start c46f2e9e4690
c46f2e9e4690
```

```
root@kerneltalks # docker container ls -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
c46f2e9e4690	httpd	"httpd-foreground"	35 minutes ago	Up 8 seconds	0.0.0.0:80->80/tcp	cranky_cori

如何移除 Docker 容器？

如果我们需要从 Docker 引擎里面移除容器，我们可以使用 rm 命令。但是你不可以移除正在运行中的容器。移除之前你需要先停止容器。不过你也可以使用 rm 命令的同时带上 -f 参数来强制移除容器，不过并不推荐这么做。

```
root@kerneltalks # docker container rm cranky_cori
cranky_cori
```

```
root@kerneltalks # docker container ls -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

容器被移除之后，我们将无法再使用 ls -a 命令查看容器了。

本文翻译自：[8 basic Docker container management commands](#)

本博客文章除特别声明，全部都是原创！
原创文章版权归过往记忆大数据（[过往记忆](#)）所有，未经许可不得转载。
本文链接：[【】（）](#)