Docker tutorial

Ivan Murashko

Contents

Introduction

There are several examples of docker usage. They are collected in one place mainly for future references.

The source code for examples can be found on the article git repository [1] in the folder **dockertutorial/src**.

1 Base commands

1.1 Simple program run

You can run a command (**uname -a**) with

```
$ docker run ubuntu uname -a
```

The container **ubuntu:latest** will be used in the case.

The command execution status can be viewed with

```
$ docker ps -a
CONTAINER ID   IMAGE   COMMAND     CREATED         ...
b5e8d82cce29   ubuntu  "uname -a"  5 seconds ago  ...
```

If you run the docker with –**rm** flag then the status info will not be stored.

1.2 Interactive session

You can run an interactive shell with

```
$ docker run -it ubuntu /bin/bash
```

where **-it** means –**interactive** –**tty**, **ubuntu** the latest ubuntu image and **/bin/bash** - the command to be start

1.3 Docker as a daemon

First of all run the docker in interactive mode and as daemon

```
$ docker run -itd ubuntu
```

possible output

```
649dae02de59ea3eb065a40b1248b2d322986e563ab12af3126fa4bb4710008a
```

Check the docker run in daemon mode with

```
$ docker ps
```

output:

```
$ docker ps
CONTAINER ID        IMAGE               COMMAND      ...
649dae02de59        ubuntu              "/bin/bash" ...
```

Execute **ls /var** command in the run docker

```
$ docker exec -it 649dae02de59 ls /var
backups  cache  lib  local  lock  log  mail  opt  run  spool  tmp
```

Stop it with

```
$ docker stop 649dae02de59
649dae02de59
```

Check the result

```
$ docker ps
CONTAINER ID        IMAGE               COMMAND    ...
```

1.4 Network daemon

You can run a nginx web server with the following command

```
$ docker run -p 8080:80 -d nginx
```

This will make the nginx server available on the host machine via port 8080:

```
$ telnet localhost 8080
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Server: nginx/1.17.4
Date: Sat, 28 Sep 2019 18:44:45 GMT
Content-Type: text/html
Content-Length: 612
Last-Modified: Tue, 24 Sep 2019 14:49:10 GMT
Connection: close
ETag: "5d8a2ce6-264"
Accept-Ranges: bytes

Connection closed by foreign host.
```

1.5 Stop all

You can stop all containers with

```
$ docker container stop $(docker container ls -aq)
```

1.6 Cleanup

The following command will remove everything

```
$ docker system prune -a
WARNING! This will remove:
        - all stopped containers
        - all networks not used by at least one container
        - all images without at least one container associated to them
        - all build cache
Are you sure you want to continue? [y/N] y
Deleted Containers:
b5e8d82cce2942a24c709b630ff4e0dd705b89d78f2777065446ce97cf152cab
...
Total reclaimed space: 6.113GB
```

References

[1] Murashko, I. Articles git repository. — 2019. — https://github.com/ivanmurashko/articles.