mpoumpouka checked_az

463c9df   on May 12

2 contributors

Raw   Blame   History
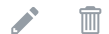
123 lines (98 sloc)   3.5 KB

class: center, middle

Section 6 - Docker Networking Basics

3 - Docker Network DNS - LAB

Overview

Test network connectivity from container shells

Objectives

- Create a custom bridge network "net-test".
- Create two containers (centos and ubuntu), one based on centos:7 image and one based on ubuntu:18.10.
- Both containers should be connected to the "net-test" virtual network.

- Use two different terminal windows to start a bash in both *centos* and *ubuntu*, using `-it` options.
- Learn about the docker container `--rm` option, so you can save the cleanup.
- Ensure `ping` and `nslookup` tools are installed on both containers:
  - ubuntu:
    - `apt-get update && apt-get install -y iputils-ping && apt-get install -y dnsutils`
  - centos:
    - `yum install -y iputils && yum install -y bind-utils`
- Use the `ping` command to verify the network connectivity and the DNS resolving.

Solution

step 1

- Create a custom bridge network "net-test":

```
# docker network create net-test
```

step 2 (from terminal window #1)

- Start ubuntu container:

```
# docker container run -it --rm --net net-test --name ubuntu ubuntu:18.10
```

> Since the default CMD is `bash` we are now on the ubuntu bash prompt

step 3 (from terminal window #2)

- Start centos container:

```
docker container run -it --rm --net net-test --name centos centos:7
### step 3 (from terminal #2 window - ubuntu)
```

> Since the default CMD is `bash` we are now on the centos bash prompt.

step 4 (from terminal window #1 - ubuntu bash)

- Install `ping` and `nsloolup` .
- Execute `ping` and `nsloolup` to verify network connectivity.

```
root@5da55426589f:/# apt-get update && apt-get install -y iputils-ping && apt-get
...

root@5da55426589f:/# ping -c3 centos
PING centos (172.19.0.3) 56(84) bytes of data.
64 bytes from centos.net-test (172.19.0.3): icmp_seq=1 ttl=64 time=0.092 ms
64 bytes from centos.net-test (172.19.0.3): icmp_seq=2 ttl=64 time=0.104 ms
64 bytes from centos.net-test (172.19.0.3): icmp_seq=3 ttl=64 time=0.106 ms


root@5da55426589f:/# nslookup centos
Server:         127.0.0.11
Address:        127.0.0.11#53

Non-authoritative answer:
Name:   centos
Address: 172.19.0.3
```

step 5 (from terminal window #2 - centos bash)

- Install `ping` and `nsloolup` .
- Execute `ping` and `nsloolup` to verify network connectivity.

```
[root@cf6601a0562e /]# yum install -y iputils && yum install -y bind-utils
...

[root@cf6601a0562e /]# ping -c3 ubuntu
PING ubuntu (172.19.0.2) 56(84) bytes of data.
64 bytes from ubuntu.net-test (172.19.0.2): icmp_seq=1 ttl=64 time=0.038 ms
64 bytes from ubuntu.net-test (172.19.0.2): icmp_seq=2 ttl=64 time=0.094 ms
64 bytes from ubuntu.net-test (172.19.0.2): icmp_seq=3 ttl=64 time=0.179 ms
...

[root@cf6601a0562e /]# nslookup ubuntu
Server:         127.0.0.11
Address:        127.0.0.11#53

Non-authoritative answer:
Name:   ubuntu
Address: 172.19.0.2
```

step 6 (from terminal window #1 - ubuntu bash)

- exit the ubuntu container shell:

```
root@5da55426589f:/# exit
```

step 7 (from terminal window #2 - centos bash)

- exit the centos container shell:

```
[root@cf6601a0562e /]# exit
```

step 8 (from terminal window #1) .

- Verify that the container have been automatically stopped and deleted, since we have used the `--rm` option

```
$ docker container ls -a
CONTAINER ID    IMAGE    COMMAND    CREATED    STATUS    PORTS    NAMES
```