

Understand Docker Networking

Task 1: Create a Docker Network :

1. Create a user-defined bridge network :

```
sudo docker network create mynetwork
```

Task 2: Run Containers on "mynetwork" :

1. Run a container named container1 and connect it to the mynetwork :

```
sudo docker run -d --name container1 --network mynetwork alpine sleep 1d
```

2. Run another container named container2 and connect it to the same network :

```
sudo docker run -d --name container2 --network mynetwork alpine sleep 1d
```

Task 3: Inspect the Bridge Network :

1. Check the network settings for the mynetwork :

```
sudo docker network inspect mynetwork
```

```
[ec2-user@ip-172-31-20-3 ~]$ sudo docker network inspect mynetwork
[
  {
    "Name": "mynetwork",
    "Id": "de37052dacb76594170cd97dab3054df36659fe427f0cd8c57e12ec978807c74",
    "Created": "2023-11-24T08:25:43.200159299Z",
    "Scope": "local",
    "Driver": "bridge",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": {},
      "Config": [
        {
          "Subnet": "172.18.0.0/16",
          "Gateway": "172.18.0.1"
        }
      ]
    }
  }
]
```

2. Copy the Ip Address of the containers and save it somewhere :

```
"Containers": {
  "85fa746e39286d944ad46c5c7183b3ad77b81444c28572dafa342834cc0e01d3": {
    "Name": "container2",
    "EndpointID": "6459eb94b95faf12547244cb610ab4caab9fd799f1a51f7f1846377339241ea6",
    "MacAddress": "02:42:ac:12:00:03",
    "IPv4Address": "172.18.0.3/16",
    "IPv6Address": ""
  },
  "d877c93fela7ff28bc2ad71491c5625111374adf6646f03e266a7ce245894f00": {
    "Name": "container1",
    "EndpointID": "fd359c6965f103b96c00a17ec96219felf7dcc3fadee59404231c16eca98196f",
    "MacAddress": "02:42:ac:12:00:02",
    "IPv4Address": "172.18.0.2/16",
    "IPv6Address": ""
  }
},
}
```

Task 4: Test Communication in the containers :

1. Let's go inside of one container and ping the other one:

```
sudo docker exec -it container1 sh
```

2. Now we are in the Container1 let's try to ping :

```
ping -c 4 <IP_of_container2>
```

Replace the <IP_of_container2> to the actual Ip address that we saved in previous task.

```
/ # ping -c 4 172.18.0.2
PING 172.18.0.2 (172.18.0.2): 56 data bytes
64 bytes from 172.18.0.2: seq=0 ttl=255 time=0.048 ms
64 bytes from 172.18.0.2: seq=1 ttl=255 time=0.059 ms
64 bytes from 172.18.0.2: seq=2 ttl=255 time=0.061 ms
64 bytes from 172.18.0.2: seq=3 ttl=255 time=0.059 ms

--- 172.18.0.2 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.048/0.056/0.061 ms
```

3. You can also ping the container through the name of container :

```
/ # ping -c 4 container2
PING container2 (172.18.0.3): 56 data bytes
64 bytes from 172.18.0.3: seq=0 ttl=255 time=0.088 ms
64 bytes from 172.18.0.3: seq=1 ttl=255 time=0.114 ms
64 bytes from 172.18.0.3: seq=2 ttl=255 time=0.073 ms
64 bytes from 172.18.0.3: seq=3 ttl=255 time=0.075 ms

--- container2 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 0.073/0.087/0.114 ms
```