CN LAB TASK 04
Fall 2020

Momina Atif Dar
P18-0030

Host (Admin) Connection:

```
(base) momina@death-eater:~$ sudo docker run --name admin1 --network host -d nginx
881fcb8e062629b4cd33e387a25196b0bf3012e7667067fd411b71e7dc27daf8
(base) momina@death-eater:~$ sudo docker inspect host
[
    {
        "Name": "host",
        "Id": "fc7d23b1f50475f9142985e8da55aadff9dc7065c89548b6e2a10aa470ded7f2",
        "Created": "2020-04-28T01:53:42.577279804+05:00",
        "Scope": "local",
        "Driver": "host",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": []
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "881fcb8e062629b4cd33e387a25196b0bf3012e7667067fd411b71e7dc27daf8": {
                "Name": "admin1",
                "EndpointID": "8850ddea05927e4ed57554f3f49cf0283e4eb4f430c5bd501df8ddba0ef7b710",
                "MacAddress": "",
                "IPv4Address": "",
                "IPv6Address": ""
            }
        },
        "Options": {},
        "Labels": {}
    }
]
```

Customer Network Creation:

```
(base) momina@death-eater:~$ sudo docker network create customer
d5c0cac6f590d614e9c373ee5b1d132bad25f0fbe4a28d03f0f5477ce91839ee
(base) momina@death-eater:~$ sudo docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
dc976fee5137        bridge              bridge              local
d5c0cac6f590        customer            bridge              local
fc7d23b1f504        host                host                local
471fbb1d1733        mynet               bridge              local
135d15930b43        none                null                local
```

Production Network Creation:

```
(base) momina@death-eater:~$ sudo docker network create production
36932dce42598f726499f32222272a5049fd94b9473c27836f63bbb4a62cd6fc
(base) momina@death-eater:~$ sudo docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
dc976fee5137        bridge              bridge              local
d5c0cac6f590        customer            bridge              local
fc7d23b1f504        host                host                local
471fbb1d1733        mynet               bridge              local
135d15930b43        none                null                local
36932dce4259        production          bridge              local
```

3 containers in Customer:

```
(base) momina@death-eater:~$ sudo docker run --name c1 --network customer -d nginx
[sudo] password for momina:
04980ccf93e93a7507e79182d64c64548157dc22d5494018455aab0d6fe46b30
```

```
(base) momina@death-eater:~$ sudo docker run --name c2 --network customer -d nginx
182436a82e06e7e6fd94b3ebe1b7cdd5b5a7292dfed2b3c28ea7d08db854d05a
(base) momina@death-eater:~$ sudo docker run --name c3 --network customer -d nginx
9c68b3e63e4c354f9ba58d149d365c362898c04c804faf45b0ee0a4a62e2abf2
(base) momina@death-eater:~$ sudo docker inspect customer
[
    {
        "Name": "customer",
        "Id": "d5c0cac6f590d614e9c373ee5b1d132bad25f0fbe4a28d03f0f5477ce91839ee",
        "Created": "2020-09-28T13:14:59.709260051+05:00",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": {},
            "Config": [
                {
                    "Subnet": "172.21.0.0/16",
                    "Gateway": "172.21.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "04980ccf93e93a7507e79182d64c64548157dc22d5494018455aab0d6fe46b30": {
                "Name": "c1",
                "EndpointID": "3b2954372ec494860091206108b8b2dbecd039a2c2214d7958cb4b6095e00794",
                "MacAddress": "02:42:ac:15:00:02",
                "IPv4Address": "172.21.0.2/16",
                "IPv6Address": ""
            },
            "182436a82e06e7e6fd94b3ebe1b7cdd5b5a7292dfed2b3c28ea7d08db854d05a": {
                "Name": "c2",
                "EndpointID": "9d8f7bf08edc9d81ca479a80add576b41aa2850b22ecbcb07cab1579f3e37011",
                "MacAddress": "02:42:ac:15:00:03",
                "IPv4Address": "172.21.0.3/16",
                "IPv6Address": ""
            },
            "9c68b3e63e4c354f9ba58d149d365c362898c04c804faf45b0ee0a4a62e2abf2": {
                "Name": "c3",
                "EndpointID": "c1f84d6418100beb20034e38378d76b98da0a305a118f92415e4b5f5c9e54682",
                "MacAddress": "02:42:ac:15:00:04",
                "IPv4Address": "172.21.0.4/16",
                "IPv6Address": ""
            }
        },
        "Options": {},
```

3 containers in Production:

```
root@28a65764555f:/# (base) momina@death-eater:~$ sudo docker run --name p1 --network production -d nginx
[sudo] password for momina:
67740cd006af3f4a01130aad29725767cfff996bfb0f2cfea5cd28b7a079b63d
(base) momina@death-eater:~$ sudo docker run --name p2 --network production -d nginx
bc8845f3515b0dd6115f89c6e2b2457f2fd5a94546007b0a8f853f424e859517
(base) momina@death-eater:~$ sudo docker run --name p3 --network production -d nginx
8c2c8821eaab3f73d936fa27861a50bb70b54f3a9f08a2cee78976ac9e2bd039
(base) momina@death-eater:~$ sudo docker inspect production
[
    {
        "Name": "production",
        "Id": "36932dce42598f726499f32222272a5049fd94b9473c27836f63bbb4a62cd6fc",
        "Created": "2020-09-28T13:15:32.464346783+05:00",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": {},
            "Config": [
                {
                    "Subnet": "172.22.0.0/16",
                    "Gateway": "172.22.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "67740cd006af3f4a01130aad29725767cfff996bfb0f2cfea5cd28b7a079b63d": {
                "Name": "p1",
                "EndpointID": "66fe62466886b28f83d2456794b4ce8da71654ac36b98fbcd6abbaaa311d8c63",
                "MacAddress": "02:42:ac:16:00:02",
                "IPv4Address": "172.22.0.2/16",
                "IPv6Address": ""
            },
            "8c2c8821eaab3f73d936fa27861a50bb70b54f3a9f08a2cee78976ac9e2bd039": {
                "Name": "p3",
                "EndpointID": "b7e8ac55745db22dd264785434068ea6d8c5cc6b9eb59dfe8813b3acd5d45bb9",
                "MacAddress": "02:42:ac:16:00:04",
                "IPv4Address": "172.22.0.4/16",
                "IPv6Address": ""
            },
            "bc8845f3515b0dd6115f89c6e2b2457f2fd5a94546007b0a8f853f424e859517": {
                "Name": "p2",
                "EndpointID": "8540dc0fc584d7040a5229d5a277dbb43bb8ac3651a7295b0e7660c58eb641f6",
                "MacAddress": "02:42:ac:16:00:03",
                "IPv4Address": "172.22.0.3/16",
                "IPv6Address": ""
```

Installing basic stuff on customer's, production's and admin's container:

```
root@67740cd006af:/# apt-get update
Get:1 http://security.debian.org/debian-security buster/updates InRelease [65.4 kB]
Get:2 http://deb.debian.org/debian buster InRelease [121 kB]
Get:3 http://deb.debian.org/debian buster-updates InRelease [51.9 kB]
Get:4 http://security.debian.org/debian-security buster/updates/main amd64 Packages [234 kB]
Get:5 http://deb.debian.org/debian buster/main amd64 Packages [7906 kB]
64% [5 Packages 4586 kB/7906 kB 58%]
                 243 kB/s 13sa65% [5 Packages 4715 kB/7906 kB 60%]
                                              233 kB/s 13st66% [5 Packages 4766 kB/7906 kB 60%]
                                                                                233 kB/s 13stGet:6 htt
p://deb.debian.org/debian buster-updates/main amd64 Packages [7868 B]

Fetched 8387 kB in 32s (261 kB/s)

Reading package lists... Done
root@67740cd006af:/# apt-get install python
Reading package lists... Done
```

```
root@67740cd006af:/# apt-get install vim
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libgpm2 vim-common vim-runtime xxd
Suggested packages:
  gpm ctags vim-doc vim-scripts
The following NEW packages will be installed:
  libgpm2 vim vim-common vim-runtime xxd
0 upgraded, 5 newly installed, 0 to remove and 3 not upgraded.
Need to get 7425 kB of archives.
After this operation, 33.8 MB of additional disk space will be used.
```

```
root@67740cd006af:/# apt install inetutils-ping net-tools -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libidn11 netbase
The following NEW packages will be installed:
  inetutils-ping libidn11 net-tools netbase
0 upgraded, 4 newly installed, 0 to remove and 3 not upgraded.
Need to get 610 kB of archives.
```

Finding host's (admin) IP:

```
wlp0s20f3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.10.20  netmask 255.255.255.0  broadcast 192.168.10.255
        inet6 fe80::56b4:50cf:996e:87  prefixlen 64  scopeid 0x20<link>
        ether d0:c6:37:f3:07:96  txqueuelen 1000  (Ethernet)
        RX packets 862934  bytes 785275468 (748.8 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 534009  bytes 101317403 (96.6 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@death-eater:/# _
```

Pinging admin from customer's container:

```
root@04980ccf93e9:/# ping 192.168.10.20
PING 192.168.10.20 (192.168.10.20): 56 data bytes
64 bytes from 192.168.10.20: icmp_seq=0 ttl=64 time=0.161 ms
64 bytes from 192.168.10.20: icmp_seq=1 ttl=64 time=0.149 ms
64 bytes from 192.168.10.20: icmp_seq=2 ttl=64 time=0.151 ms
64 bytes from 192.168.10.20: icmp_seq=3 ttl=64 time=0.152 ms
64 bytes from 192.168.10.20: icmp_seq=4 ttl=64 time=0.156 ms
64 bytes from 192.168.10.20: icmp_seq=5 ttl=64 time=0.170 ms
64 bytes from 192.168.10.20: icmp_seq=6 ttl=64 time=0.115 ms
64 bytes from 192.168.10.20: icmp_seq=7 ttl=64 time=0.132 ms
64 bytes from 192.168.10.20: icmp_seq=8 ttl=64 time=0.150 ms
64 bytes from 192.168.10.20: icmp_seq=9 ttl=64 time=0.146 ms
64 bytes from 192.168.10.20: icmp_seq=10 ttl=64 time=0.149 ms
64 bytes from 192.168.10.20: icmp_seq=11 ttl=64 time=0.152 ms
64 bytes from 192.168.10.20: icmp_seq=12 ttl=64 time=0.149 ms
64 bytes from 192.168.10.20: icmp_seq=13 ttl=64 time=0.150 ms
```

Pinging admin from production's container:

```
root@67740cd006af:/# ping 192.168.10.20
PING 192.168.10.20 (192.168.10.20): 56 data bytes
64 bytes from 192.168.10.20: icmp_seq=0 ttl=64 time=0.210 ms
64 bytes from 192.168.10.20: icmp_seq=1 ttl=64 time=0.153 ms
64 bytes from 192.168.10.20: icmp_seq=2 ttl=64 time=0.154 ms
64 bytes from 192.168.10.20: icmp_seq=3 ttl=64 time=0.138 ms
64 bytes from 192.168.10.20: icmp_seq=4 ttl=64 time=0.147 ms
```

Server code:

```
import socket
s= socket.socket()

s.bind(("192.168.10.20",9200))
s.listen(20)
print("socket is bind")
while True:
    clt, adr=s.accept()
    stradr=str(adr)
    print("Connection to "+stradr+" established")
    string="Do you have any order?"
    arr=string.encode("utf-8")
    clt.send(arr)
~
```

Client code:

```
import socket
s=socket.socket()
s.connect(("192.168.10.20",9200))
msg= s.recv(1024)
print(msg.decode("utf-8"))
~
```

Not Working:

```
root@death-eater:/# vi server.py
root@death-eater:/# python server.py
socket is bind
```

```
root@04980ccf93e9:/# vi client.py
root@04980ccf93e9:/# python client.py
```

It's listening on port 9200 but not displaying the message on client side. I have tried various solutions from different sources but to no avail. Because this basic code is not working so I couldn't try two-way communication as well as communication between production and admin.

UPDATE: It works if we disable firewall.
