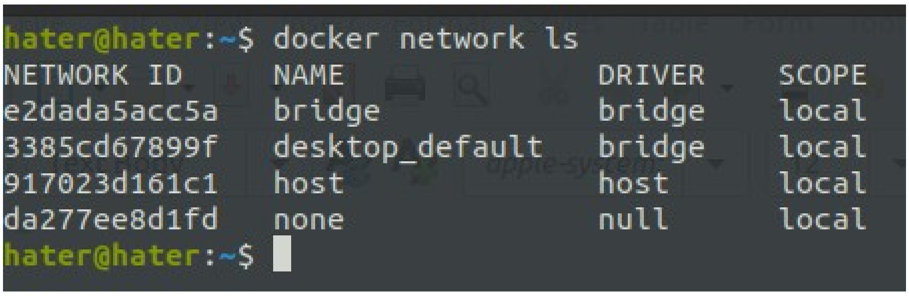
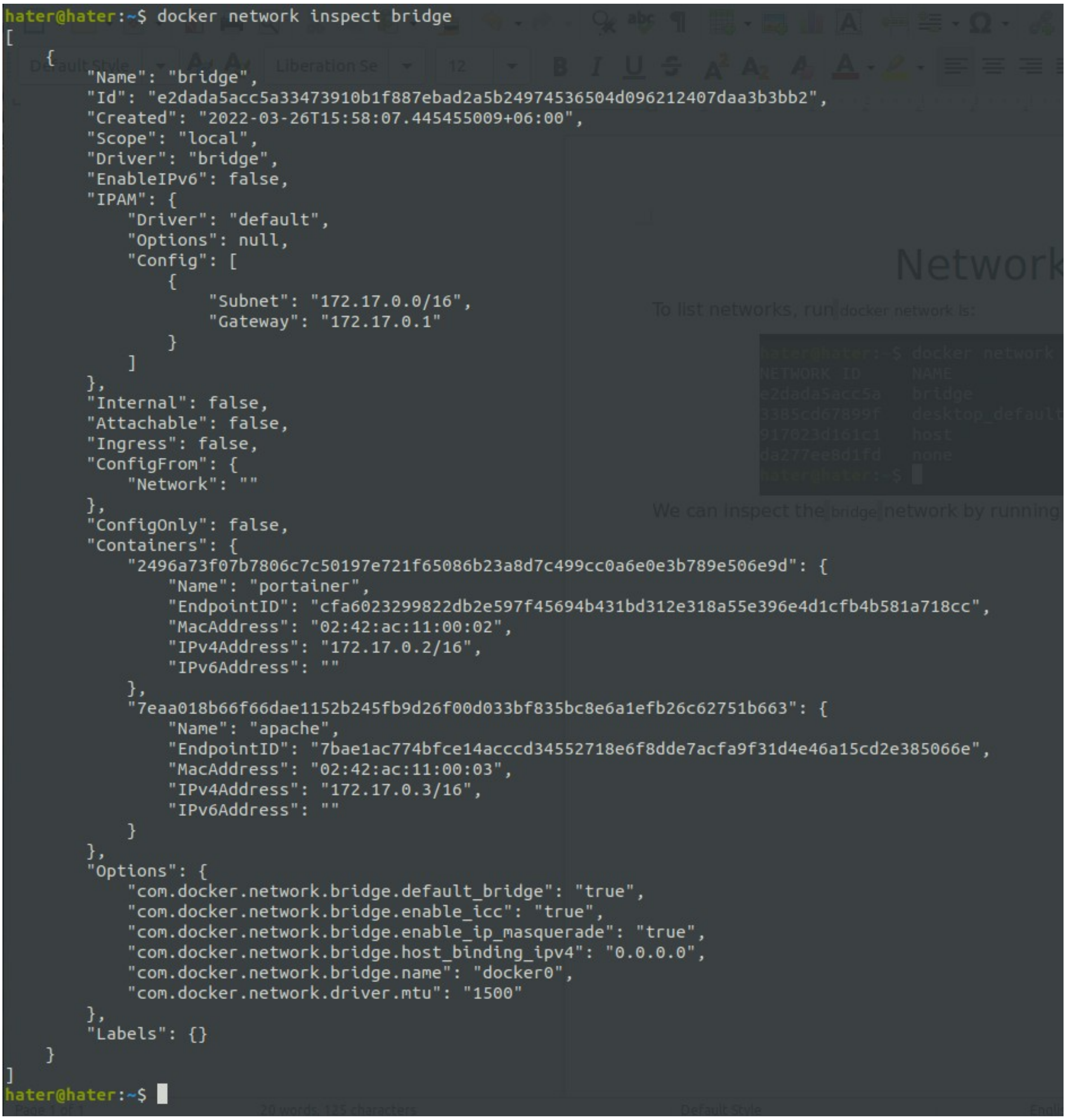
Networking

To list networks, run docker network ls:



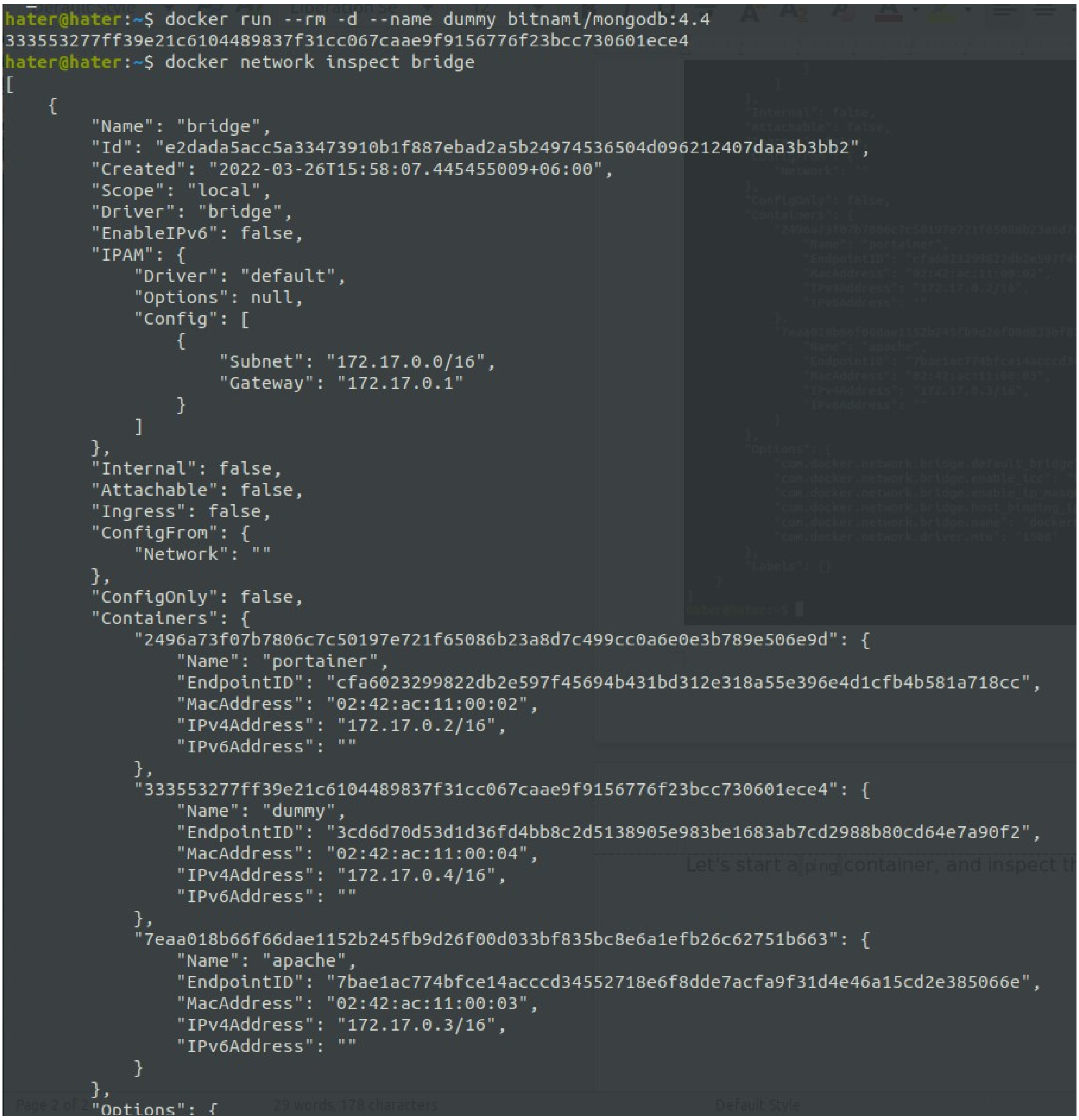
```
hater@hater:~$ docker network ls
NETWORK ID     NAME              DRIVER    SCOPE
e2dada5acc5a   bridge            bridge    local
3385cd67899f   desktop_default   bridge    local
917023d161c1   host              host      local
da277ee8d1fd   none              null      local
hater@hater:~$
```

We can inspect the bridge network by running docker network inspect bridge:



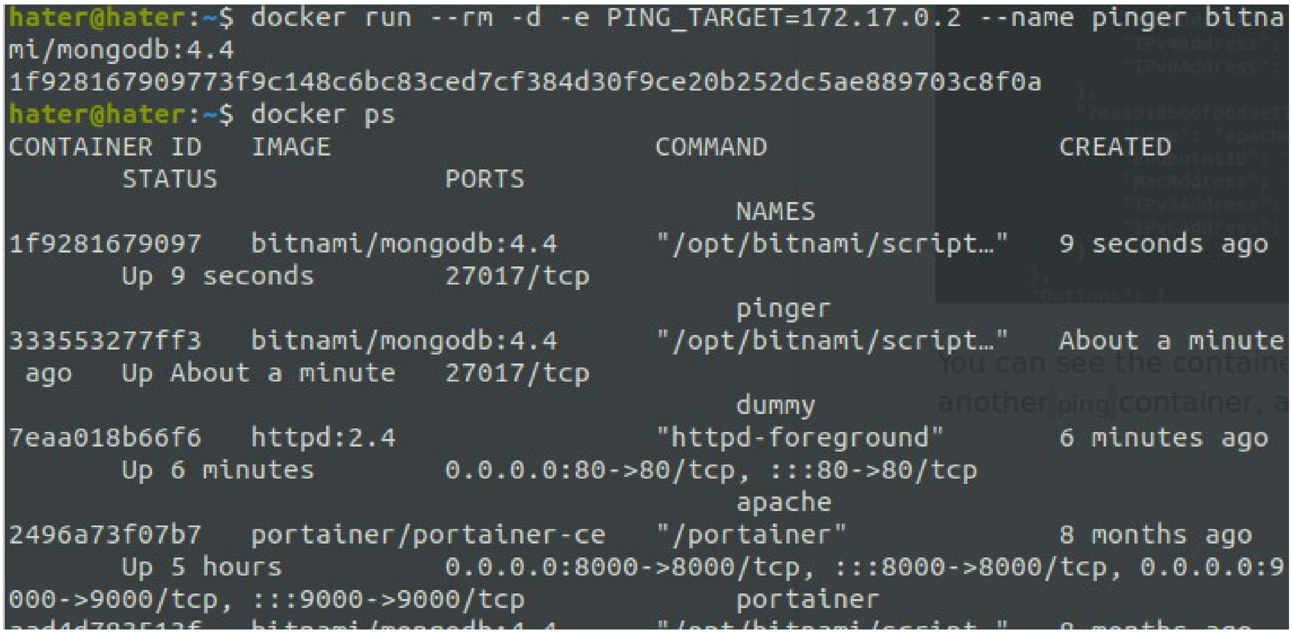
```
hater@hater:~$ docker network inspect bridge
[
    {
        "Name": "bridge",
        "Id": "e2dada5acc5a33473910b1f887ebad2a5b24974536504d096212407daa3b3bb2",
        "Created": "2022-03-26T15:58:07.445455009+06:00",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "172.17.0.0/16",
                    "Gateway": "172.17.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "2496a73f07b7806c7c50197e721f65086b23a8d7c499cc0a6e0e3b789e506e9d": {
                "Name": "portainer",
                "EndpointID": "cfa6023299822db2e597f45694b431bd312e318a55e396e4d1cfb4b581a718cc",
                "MacAddress": "02:42:ac:11:00:02",
                "IPv4Address": "172.17.0.2/16",
                "IPv6Address": ""
            },
            "7eaa018b66f66dae1152b245fb9d26f00d033bf835bc8e6a1efb26c62751b663": {
                "Name": "apache",
                "EndpointID": "7bae1ac774bfce14acccd34552718e6f8dde7acfa9f31d4e46a15cd2e385066e",
                "MacAddress": "02:42:ac:11:00:03",
                "IPv4Address": "172.17.0.3/16",
                "IPv6Address": ""
            }
        },
        "Options": {
            "com.docker.network.bridge.default_bridge": "true",
            "com.docker.network.bridge.enable_icc": "true",
            "com.docker.network.bridge.enable_ip_masquerade": "true",
            "com.docker.network.bridge.host_binding_ipv4": "0.0.0.0",
            "com.docker.network.bridge.name": "docker0",
            "com.docker.network.driver.mtu": "1500"
        },
        "Labels": {}
    }
]
hater@hater:~$
```

Let's start a ping container, and inspect this again:



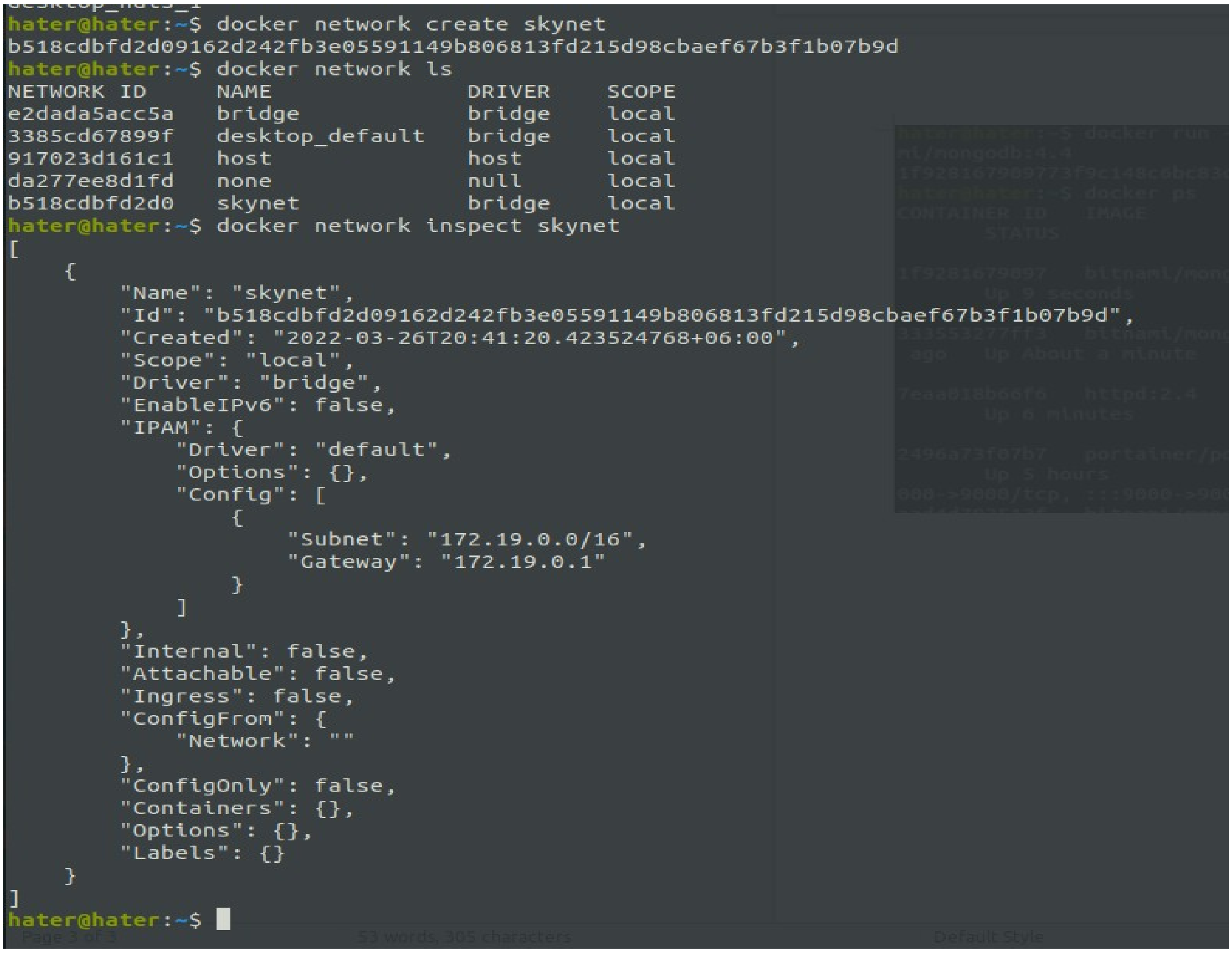
```
hater@hater:~$ docker run --rm -d --name dummy bitnami/mongodb:4.4
333553277ff39e21c6104489837f31cc067caae9f9156776f23bcc730601ece4
hater@hater:~$ docker network inspect bridge
[
    {
        "Name": "bridge",
        "Id": "e2dada5acc5a33473910b1f887ebad2a5b24974536504d096212407daa3b3bb2",
        "Created": "2022-03-26T15:58:07.445455009+06:00",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "172.17.0.0/16",
                    "Gateway": "172.17.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "2496a73f07b7806c7c50197e721f65086b23a8d7c499cc0a6e0e3b789e506e9d": {
                "Name": "portainer",
                "EndpointID": "cfa6023299822db2e597f45694b431bd312e318a55e396e4d1cfb4b581a718cc",
                "MacAddress": "02:42:ac:11:00:02",
                "IPv4Address": "172.17.0.2/16",
                "IPv6Address": ""
            },
            "333553277ff39e21c6104489837f31cc067caae9f9156776f23bcc730601ece4": {
                "Name": "dummy",
                "EndpointID": "3cd6d70d53d1d36fd4bb8c2d5138905e983be1683ab7cd2988b80cd64e7a90f2",
                "MacAddress": "02:42:ac:11:00:04",
                "IPv4Address": "172.17.0.4/16",
                "IPv6Address": ""
            },
            "7eaa018b66f66dae1152b245fb9d26f00d033bf835bc8e6a1efb26c62751b663": {
                "Name": "apache",
                "EndpointID": "7bae1ac774bfce14acccd34552718e6f8dde7acfa9f31d4e46a15cd2e385066e",
                "MacAddress": "02:42:ac:11:00:03",
                "IPv4Address": "172.17.0.3/16",
                "IPv6Address": ""
            }
        },
        "Options": {
```

You can see the container was added to the default network. Now let's add another ping container, and set it to ping our first.



```
hater@hater:~$ docker run --rm -d -e PING_TARGET=172.17.0.2 --name pinger bitna
mi/mongodb:4.4
1f928167909773f9c148c6bc83ced7cf384d30f9ce20b252dc5ae889703c8f0a
hater@hater:~$ docker ps
CONTAINER ID   IMAGE                        COMMAND                  CREATED
       STATUS              PORTS
                                                       NAMES
1f9281679097   bitnami/mongodb:4.4          "/opt/bitnami/script…"   9 seconds ago
       Up 9 seconds        27017/tcp
                                                       pinger
333553277ff3   bitnami/mongodb:4.4          "/opt/bitnami/script…"   About a minute
 ago   Up About a minute   27017/tcp
                                                       dummy
7eaa018b66f6   httpd:2.4                    "httpd-foreground"       6 minutes ago
       Up 6 minutes        0.0.0.0:80->80/tcp, :::80->80/tcp
                                                       apache
2496a73f07b7   portainer/portainer-ce       "/portainer"             8 months ago
       Up 5 hours          0.0.0.0:8000->8000/tcp, :::8000->8000/tcp, 0.0.0.0:9
000->9000/tcp, :::9000->9000/tcp                       portainer
```

To create a new network, use the docker network create command and provide it a network name.



```
hater@hater:~$ docker network create skynet
b518cdbfd2d09162d242fb3e05591149b806813fd215d98cbaef67b3f1b07b9d
hater@hater:~$ docker network ls
NETWORK ID     NAME              DRIVER    SCOPE
e2dada5acc5a   bridge            bridge    local
3385cd67899f   desktop_default   bridge    local
917023d161c1   host              host      local
da277ee8d1fd   none              null      local
b518cdbfd2d0   skynet            bridge    local
hater@hater:~$ docker network inspect skynet
[
    {
        "Name": "skynet",
        "Id": "b518cdbfd2d09162d242fb3e05591149b806813fd215d98cbaef67b3f1b07b9d",
        "Created": "2022-03-26T20:41:20.423524768+06:00",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": {},
            "Config": [
                {
                    "Subnet": "172.19.0.0/16",
                    "Gateway": "172.19.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {},
        "Options": {},
        "Labels": {}
    }
]
hater@hater:~$ 
```