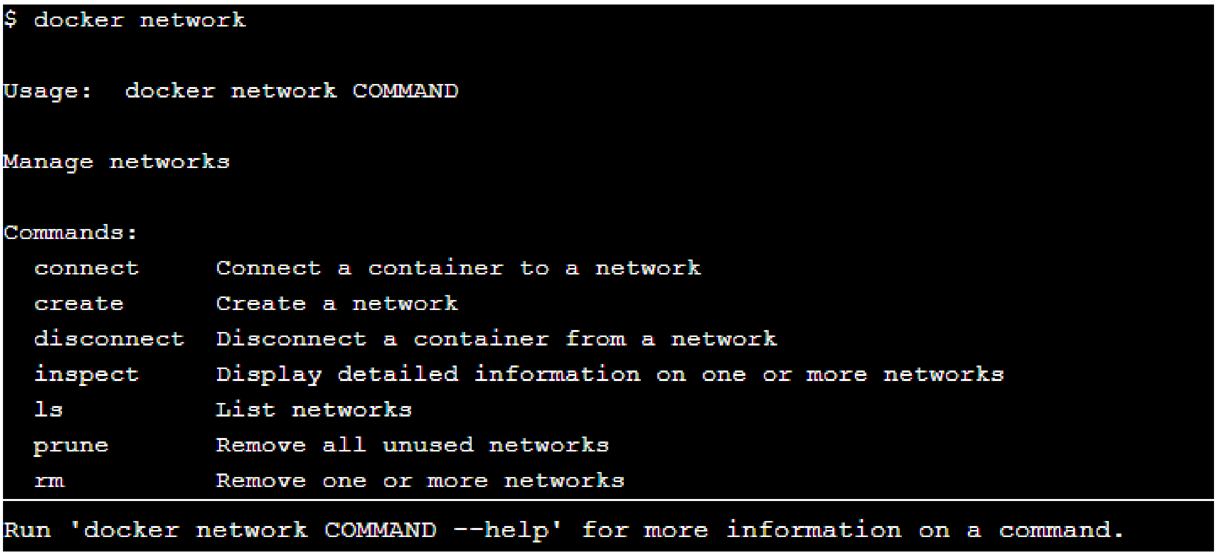
Docker networking Hands-on Lab

Dominika Kawucha

172322

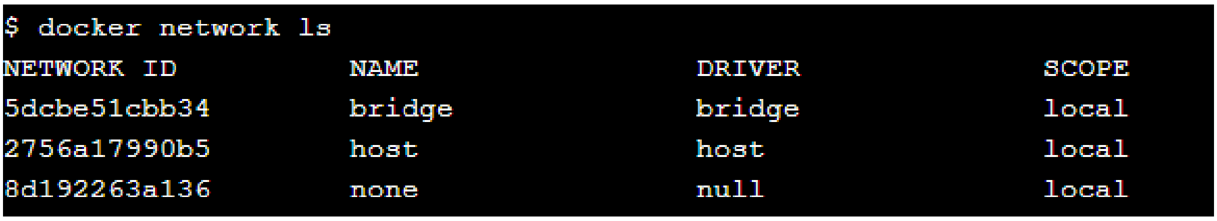
Section #1 - Networking Basics

Step 1: The Docker Network Command



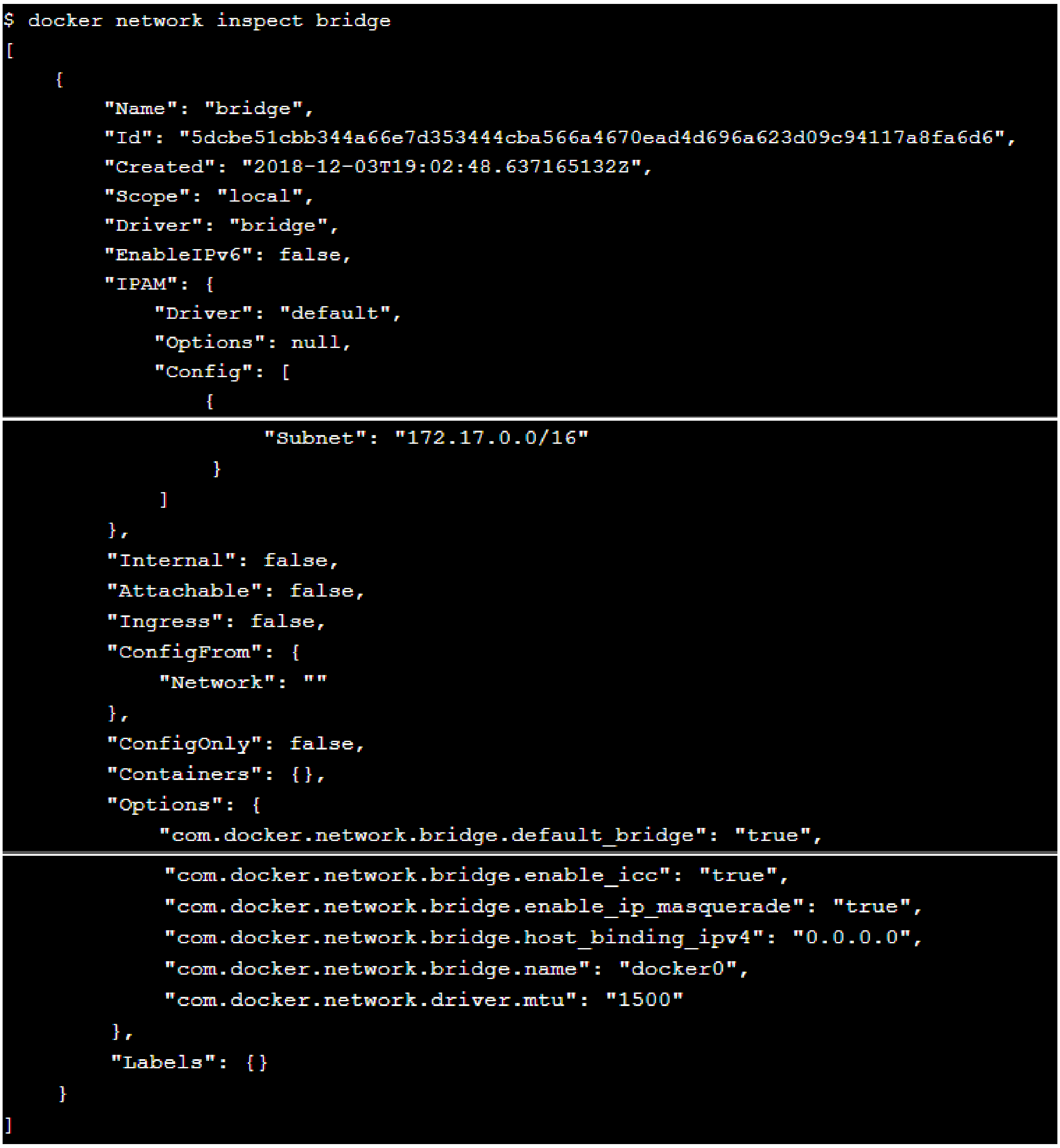
```
$ docker network

Usage:  docker network COMMAND

Manage networks

Commands:
  connect     Connect a container to a network
  create      Create a network
  disconnect  Disconnect a container from a network
  inspect     Display detailed information on one or more networks
  ls          List networks
  prune       Remove all unused networks
  rm          Remove one or more networks

Run 'docker network COMMAND --help' for more information on a command.
```

Step 2: List networks



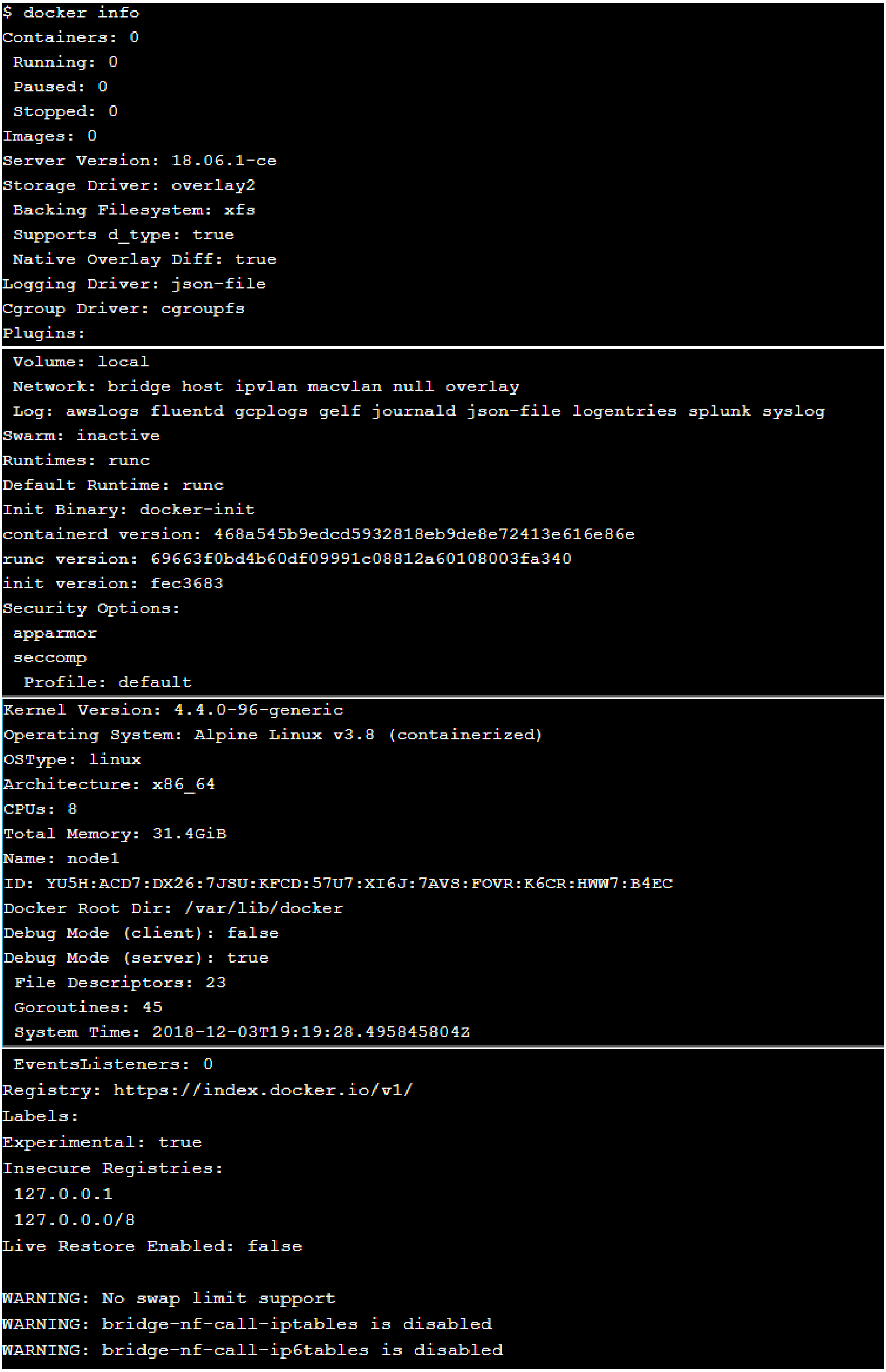
```
$ docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
5dcbe51cbb34        bridge              bridge              local
2756a17990b5        host                host                local
8d192263a136        none                null                local
```

Step 3: Inspect a network



```
$ docker network inspect bridge
[
    {
        "Name": "bridge",
        "Id": "5dcbe51cbb344a66e7d353444cba566a4670ead4d696a623d09c94117a8fa6d6",
        "Created": "2018-12-03T19:02:48.637165132Z",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "172.17.0.0/16"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {},
        "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": {}
    }
]
```

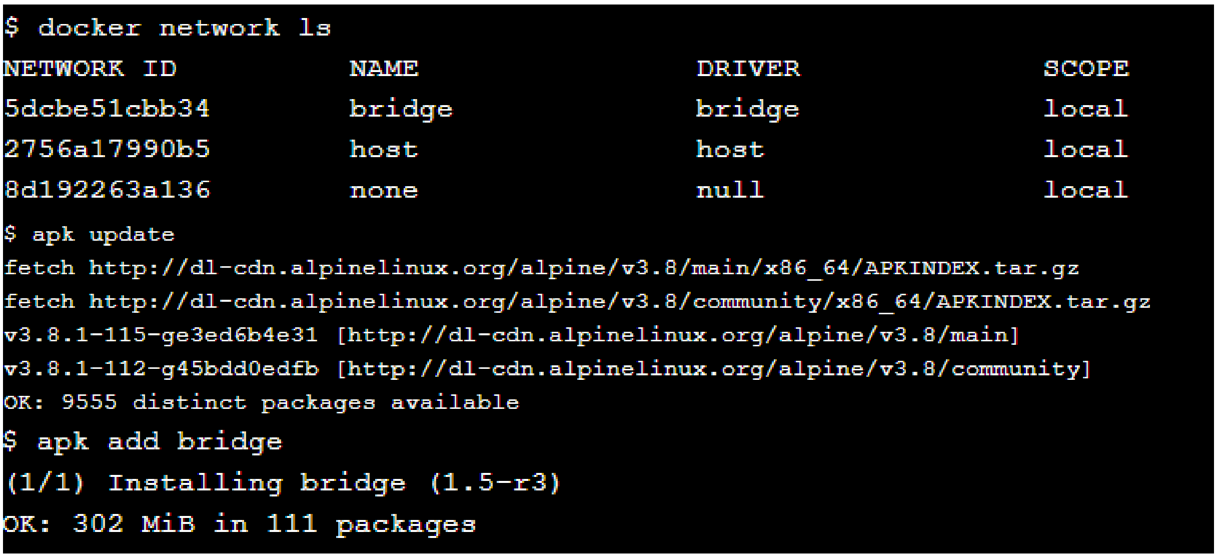
Step 4: List network driver plugins



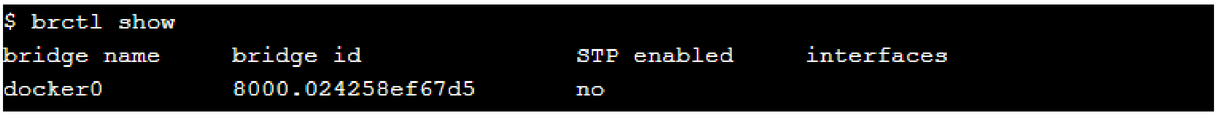
```
$ docker info
Containers: 0
 Running: 0
 Paused: 0
 Stopped: 0
Images: 0
Server Version: 18.06.1-ce
Storage Driver: overlay2
 Backing Filesystem: xfs
 Supports d_type: true
 Native Overlay Diff: true
Logging Driver: json-file
Cgroup Driver: cgroupfs
Plugins:
 Volume: local
 Network: bridge host ipvlan macvlan null overlay
 Log: awslogs fluentd gcplogs gelf journald json-file logentries splunk syslog
Swarm: inactive
Runtimes: runc
Default Runtime: runc
Init Binary: docker-init
containerd version: 468a545b9edcd5932818eb9de8e72413e616e86e
runc version: 69663f0bd4b60df09991c08812a60108003fa340
init version: fec3683
Security Options:
 apparmor
 seccomp
  Profile: default
Kernel Version: 4.4.0-96-generic
Operating System: Alpine Linux v3.8 (containerized)
OSType: linux
Architecture: x86_64
CPUs: 8
Total Memory: 31.4GiB
Name: node1
ID: YU5H:ACD7:DX26:7JSU:KFCD:57U7:XI6J:7AVS:FOVR:K6CR:HWW7:B4EC
Docker Root Dir: /var/lib/docker
Debug Mode (client): false
Debug Mode (server): true
 File Descriptors: 23
 Goroutines: 45
 System Time: 2018-12-03T19:19:28.495845804Z
 EventsListeners: 0
Registry: https://index.docker.io/v1/
Labels:
Experimental: true
Insecure Registries:
 127.0.0.1
 127.0.0.0/8
Live Restore Enabled: false

WARNING: No swap limit support
WARNING: bridge-nf-call-iptables is disabled
WARNING: bridge-nf-call-ip6tables is disabled
```

Section #2 - Bridge Networking

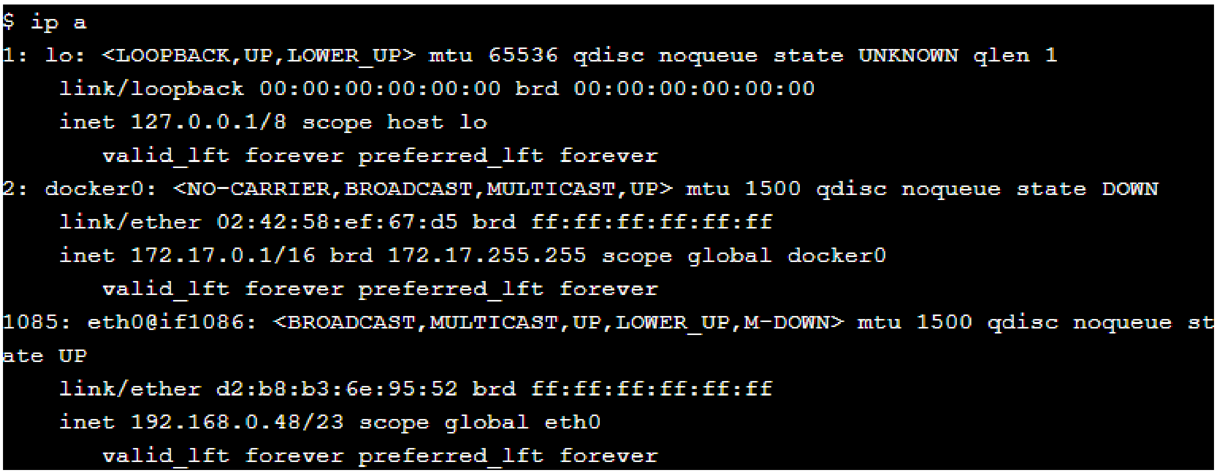
Step 1: The Basics



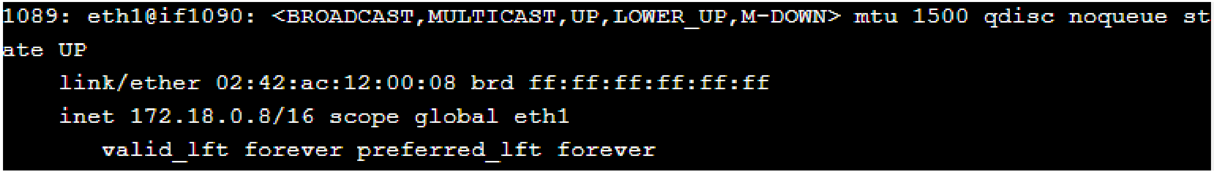
```
$ docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
5dcbe51cbb34        bridge              bridge              local
2756a17990b5        host                host                local
8d192263a136        none                null                local
$ apk update
fetch http://dl-cdn.alpinelinux.org/alpine/v3.8/main/x86_64/APKINDEX.tar.gz
fetch http://dl-cdn.alpinelinux.org/alpine/v3.8/community/x86_64/APKINDEX.tar.gz
v3.8.1-115-ge3ed6b4e31 [http://dl-cdn.alpinelinux.org/alpine/v3.8/main]
v3.8.1-112-g45bdd0edfb [http://dl-cdn.alpinelinux.org/alpine/v3.8/community]
OK: 9555 distinct packages available
$ apk add bridge
(1/1) Installing bridge (1.5-r3)
OK: 302 MiB in 111 packages
```



```
$ brctl show
bridge name     bridge id               STP enabled     interfaces
docker0         8000.024258ef67d5       no
```

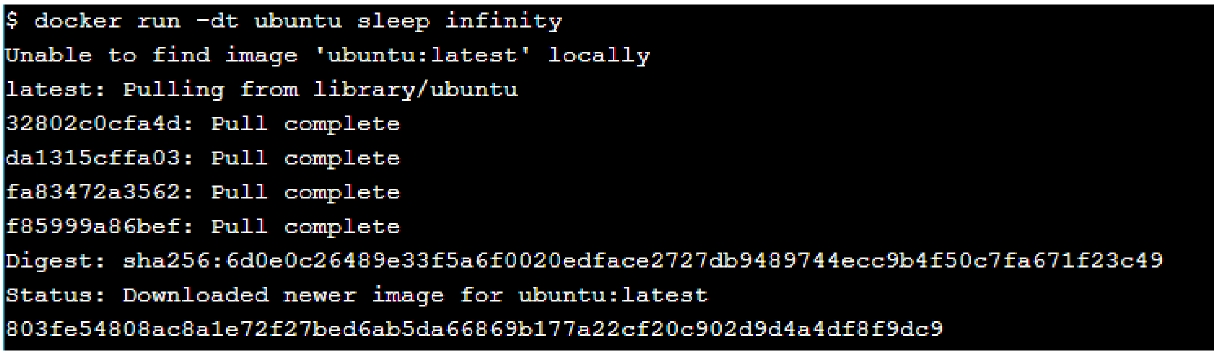


```
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
2: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN
    link/ether 02:42:58:ef:67:d5 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
       valid_lft forever preferred_lft forever
1085: eth0@if1086: <BROADCAST,MULTICAST,UP,LOWER_UP,M-DOWN> mtu 1500 qdisc noqueue st
ate UP
    link/ether d2:b8:b3:6e:95:52 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.48/23 scope global eth0
       valid_lft forever preferred_lft forever
```

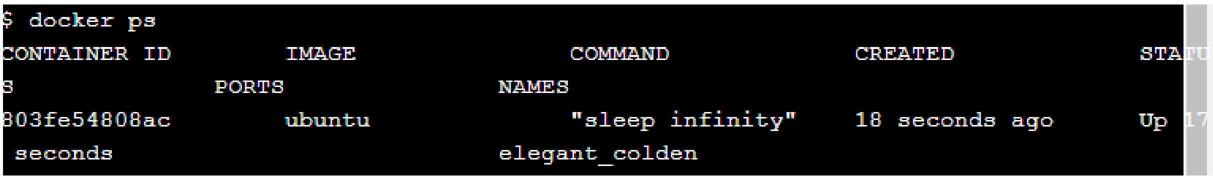


```
1089: eth1@if1090: <BROADCAST,MULTICAST,UP,LOWER_UP,M-DOWN> mtu 1500 qdisc noqueue st
ate UP
    link/ether 02:42:ac:12:00:08 brd ff:ff:ff:ff:ff:ff
    inet 172.18.0.8/16 scope global eth1
       valid_lft forever preferred_lft forever
```

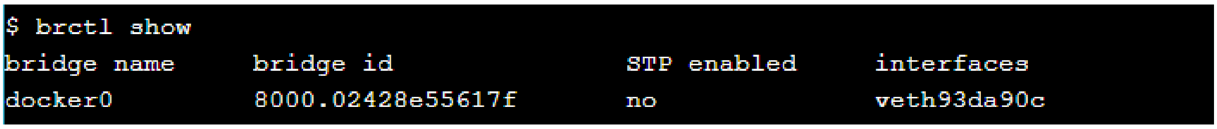
Step 2: Connect a container



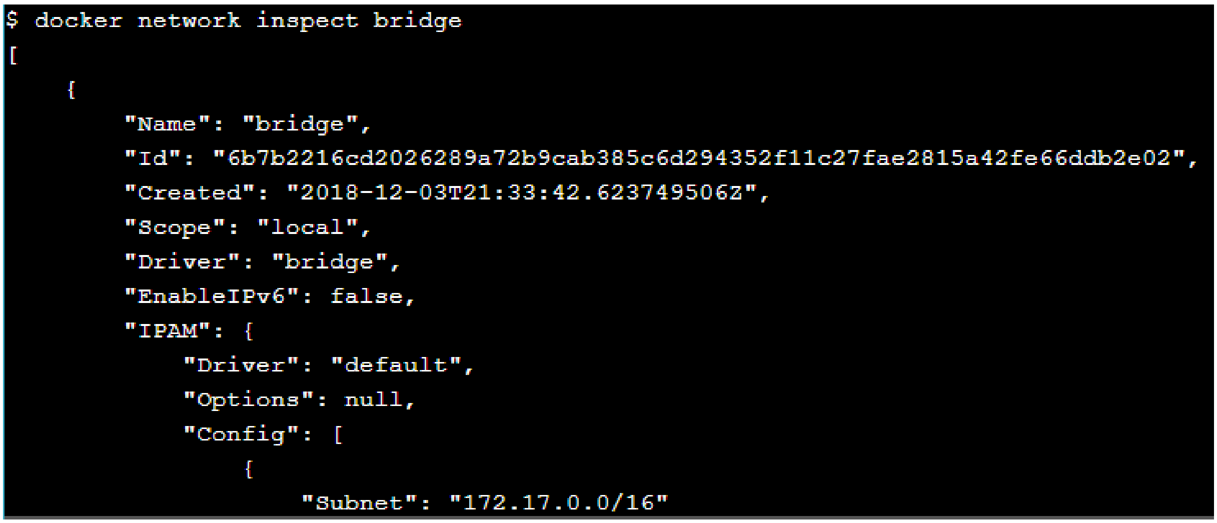
```
$ docker run -dt ubuntu sleep infinity
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
32802c0cfa4d: Pull complete
da1315cffa03: Pull complete
fa83472a3562: Pull complete
f85999a86bef: Pull complete
Digest: sha256:6d0e0c26489e33f5a6f0020edface2727db9489744ecc9b4f50c7fa671f23c49
Status: Downloaded newer image for ubuntu:latest
803fe54808ac8a1e72f27bed6ab5da66869b177a22cf20c902d9d4a4df8f9dc9
```



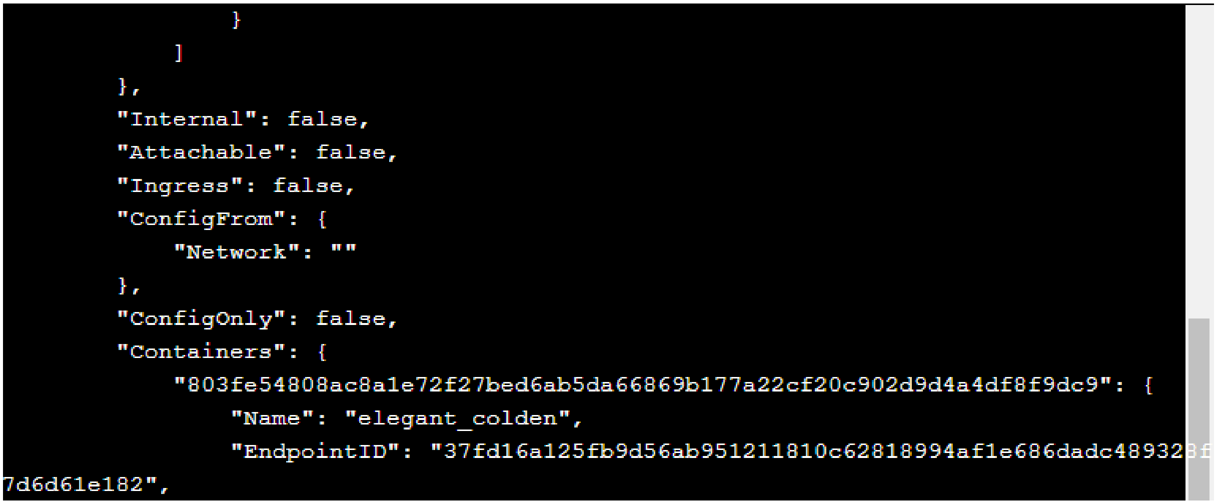
```
$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATU
S                PORTS               NAMES
803fe54808ac        ubuntu              "sleep infinity"    18 seconds ago      Up 17
 seconds                             elegant_colden
```



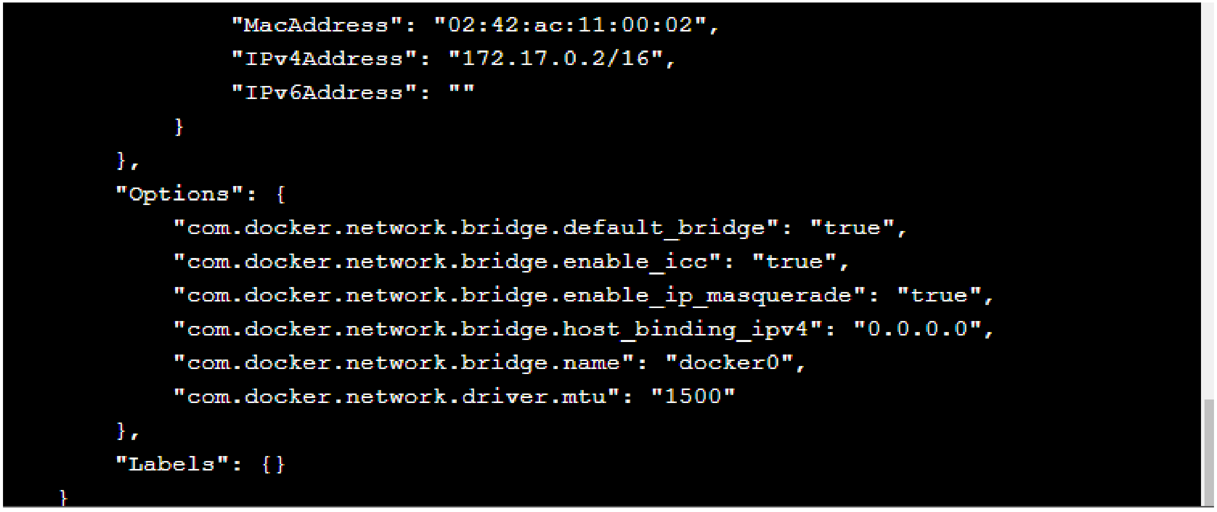
```
$ brctl show
bridge name     bridge id               STP enabled     interfaces
docker0         8000.02428e55617f       no              veth93da90c
```



```
$ docker network inspect bridge
[
    {
        "Name": "bridge",
        "Id": "6b7b2216cd2026289a72b9cab385c6d294352f11c27fae2815a42fe66ddb2e02",
        "Created": "2018-12-03T21:33:42.623749506Z",
        "Scope": "local",
        "Driver": "bridge",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "172.17.0.0/16"
```

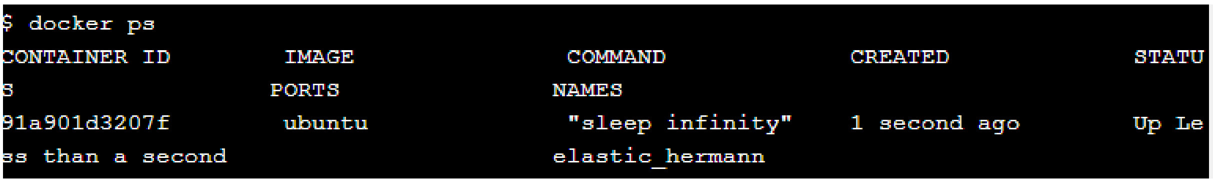


```
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "803fe54808ac8a1e72f27bed6ab5da66869b177a22cf20c902d9d4a4df8f9dc9": {
                "Name": "elegant_colden",
                "EndpointID": "37fd16a125fb9d56ab951211810c62818994af1e686dadc489323f
7d6d61e182",
```

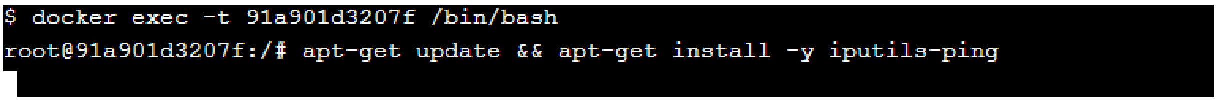


```
                "MacAddress": "02:42:ac:11:00:02",
                "IPv4Address": "172.17.0.2/16",
                "IPv6Address": ""
            }
        },
        "Options": {
            "com.docker.network.bridge.default_bridge": "true",
            "com.docker.network.bridge.enable_icc": "true",
            "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": {}
    }
```

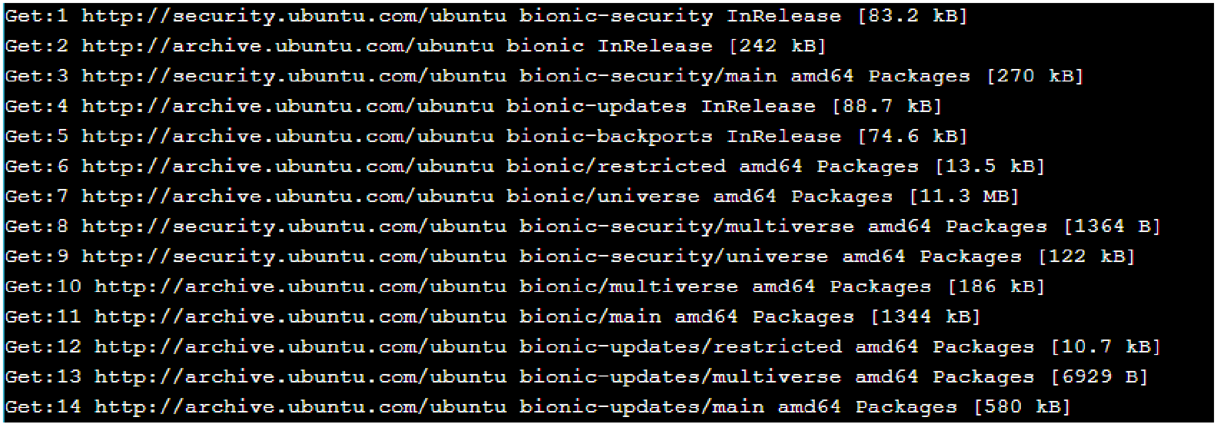
Step 3: Test network connectivity



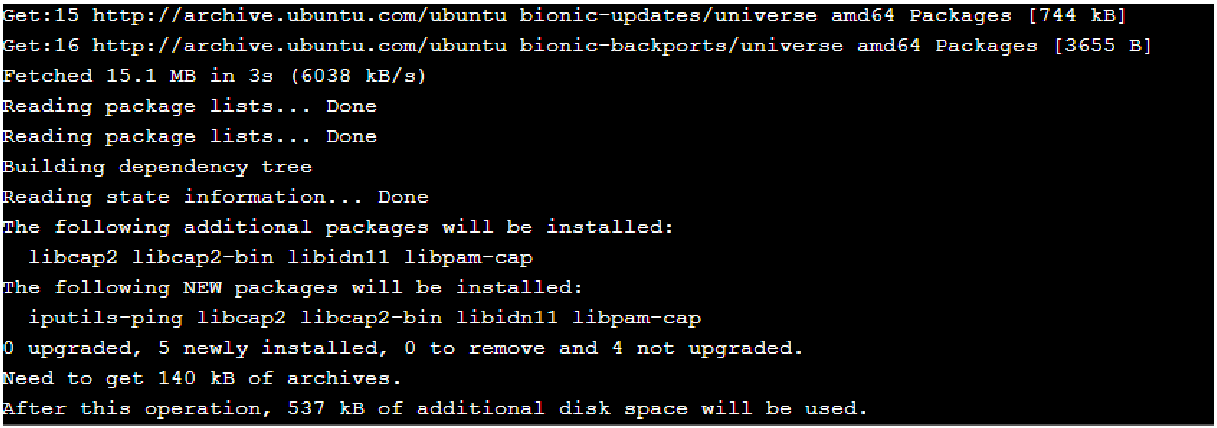
```
$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATU
S                  PORTS               NAMES
91a901d3207f        ubuntu              "sleep infinity"    1 second ago        Up Le
ss than a second                       elastic_hermann
```



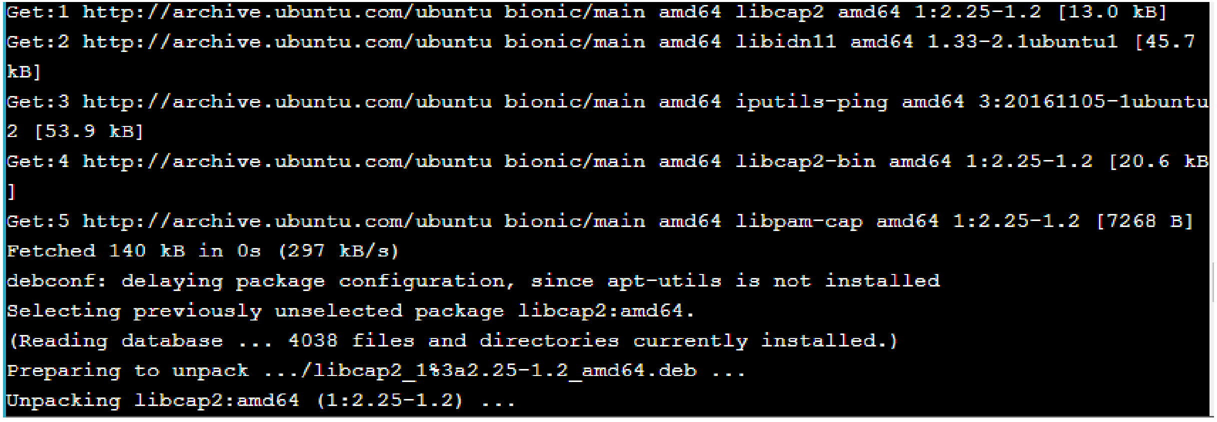
```
$ docker exec -t 91a901d3207f /bin/bash
root@91a901d3207f:/# apt-get update && apt-get install -y iputils-ping
```



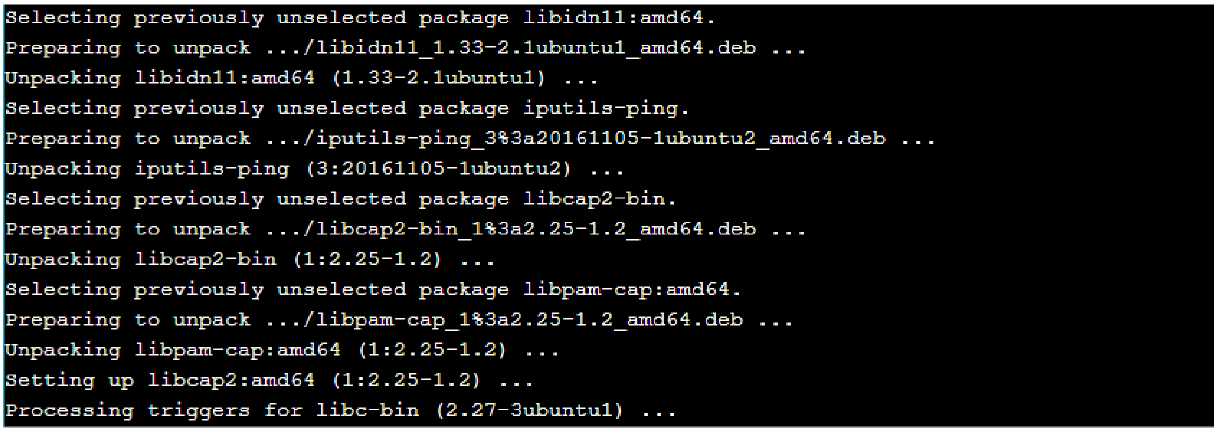
```
Get:1 http://security.ubuntu.com/ubuntu bionic-security InRelease [83.2 kB]
Get:2 http://archive.ubuntu.com/ubuntu bionic InRelease [242 kB]
Get:3 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [270 kB]
Get:4 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:5 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:6 http://archive.ubuntu.com/ubuntu bionic/restricted amd64 Packages [13.5 kB]
Get:7 http://archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [11.3 MB]
Get:8 http://security.ubuntu.com/ubuntu bionic-security/multiverse amd64 Packages [1364 B]
Get:9 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 Packages [122 kB]
Get:10 http://archive.ubuntu.com/ubuntu bionic/multiverse amd64 Packages [186 kB]
Get:11 http://archive.ubuntu.com/ubuntu bionic/main amd64 Packages [1344 kB]
Get:12 http://archive.ubuntu.com/ubuntu bionic-updates/restricted amd64 Packages [10.7 kB]
Get:13 http://archive.ubuntu.com/ubuntu bionic-updates/multiverse amd64 Packages [6929 B]
Get:14 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [580 kB]
```



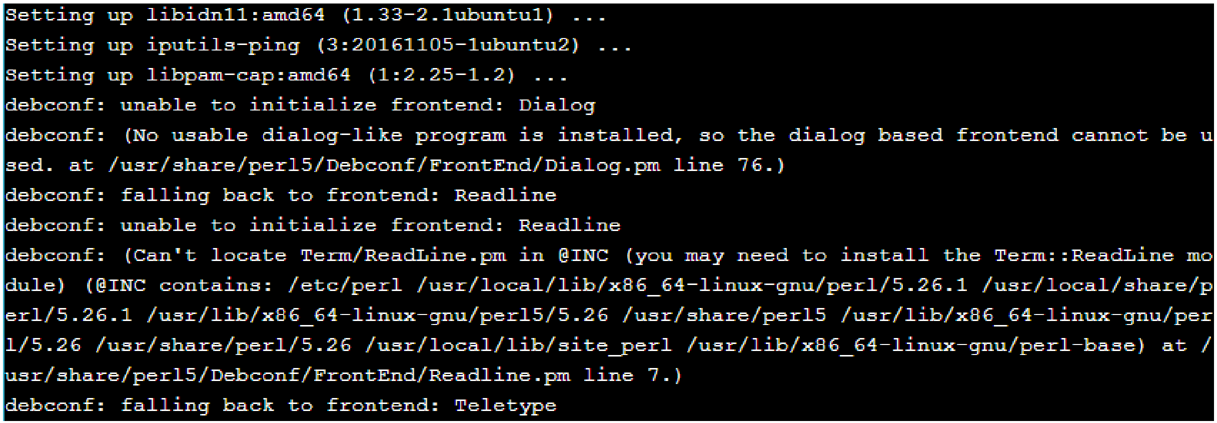
```
Get:15 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [744 kB]
Get:16 http://archive.ubuntu.com/ubuntu bionic-backports/universe amd64 Packages [3655 B]
Fetched 15.1 MB in 3s (6038 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libcap2 libcap2-bin libidn11 libpam-cap
The following NEW packages will be installed:
  iputils-ping libcap2 libcap2-bin libidn11 libpam-cap
0 upgraded, 5 newly installed, 0 to remove and 4 not upgraded.
Need to get 140 kB of archives.
After this operation, 537 kB of additional disk space will be used.
```



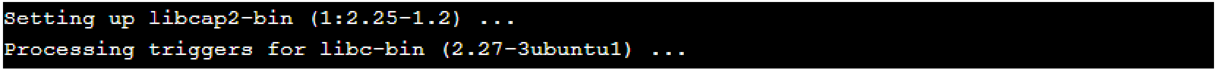
```
Get:1 http://archive.ubuntu.com/ubuntu bionic/main amd64 libcap2 amd64 1:2.25-1.2 [13.0 kB]
Get:2 http://archive.ubuntu.com/ubuntu bionic/main amd64 libidn11 amd64 1.33-2.1ubuntu1 [45.7
kB]
Get:3 http://archive.ubuntu.com/ubuntu bionic/main amd64 iputils-ping amd64 3:20161105-1ubuntu
2 [53.9 kB]
Get:4 http://archive.ubuntu.com/ubuntu bionic/main amd64 libcap2-bin amd64 1:2.25-1.2 [20.6 kB
]
Get:5 http://archive.ubuntu.com/ubuntu bionic/main amd64 libpam-cap amd64 1:2.25-1.2 [7268 B]
Fetched 140 kB in 0s (297 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package libcap2:amd64.
(Reading database ... 4038 files and directories currently installed.)
Preparing to unpack .../libcap2_1%3a2.25-1.2_amd64.deb ...
Unpacking libcap2:amd64 (1:2.25-1.2) ...
```



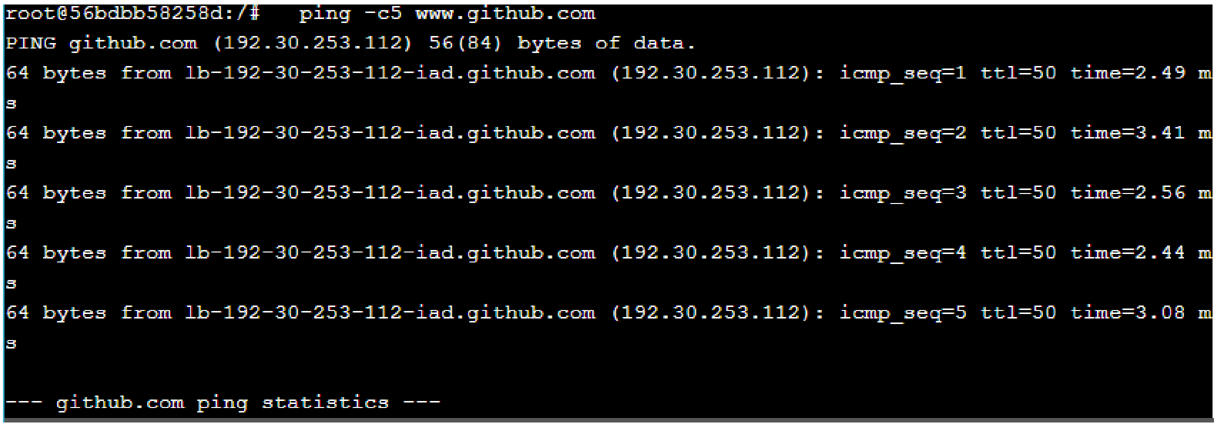
```
Selecting previously unselected package libidn11:amd64.
Preparing to unpack .../libidn11_1.33-2.1ubuntu1_amd64.deb ...
Unpacking libidn11:amd64 (1.33-2.1ubuntu1) ...
Selecting previously unselected package iputils-ping.
Preparing to unpack .../iputils-ping_3%3a20161105-1ubuntu2_amd64.deb ...
Unpacking iputils-ping (3:20161105-1ubuntu2) ...
Selecting previously unselected package libcap2-bin.
Preparing to unpack .../libcap2-bin_1%3a2.25-1.2_amd64.deb ...
Unpacking libcap2-bin (1:2.25-1.2) ...
Selecting previously unselected package libpam-cap:amd64.
Preparing to unpack .../libpam-cap_1%3a2.25-1.2_amd64.deb ...
Unpacking libpam-cap:amd64 (1:2.25-1.2) ...
Setting up libcap2:amd64 (1:2.25-1.2) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
```



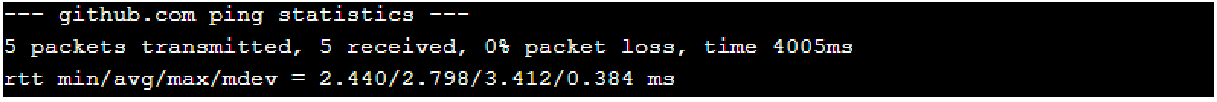
```
Setting up libidn11:amd64 (1.33-2.1ubuntu1) ...
Setting up iputils-ping (3:20161105-1ubuntu2) ...
Setting up libpam-cap:amd64 (1:2.25-1.2) ...
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend cannot be u
sed. at /usr/share/perl5/Debconf/FrontEnd/Dialog.pm line 76.)
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (Can't locate Term/ReadLine.pm in @INC (you may need to install the Term::ReadLine mo
dule) (@INC contains: /etc/perl /usr/local/lib/x86_64-linux-gnu/perl/5.26.1 /usr/local/share/p
erl/5.26.1 /usr/lib/x86_64-linux-gnu/perl5/5.26 /usr/share/perl5 /usr/lib/x86_64-linux-gnu/per
l/5.26 /usr/share/perl/5.26 /usr/local/lib/site_perl /usr/lib/x86_64-linux-gnu/perl-base) at /
usr/share/perl5/Debconf/FrontEnd/Readline.pm line 7.)
debconf: falling back to frontend: Teletype
```



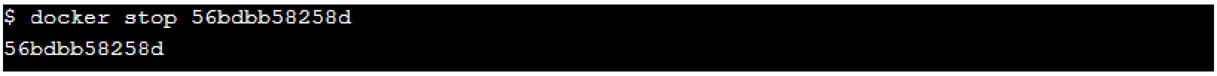
```
Setting up libcap2-bin (1:2.25-1.2) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
```



```
root@56bdbb58258d:/#   ping -c5 www.github.com
PING github.com (192.30.253.112) 56(84) bytes of data.
64 bytes from lb-192-30-253-112-iad.github.com (192.30.253.112): icmp_seq=1 ttl=50 time=2.49 m
s
64 bytes from lb-192-30-253-112-iad.github.com (192.30.253.112): icmp_seq=2 ttl=50 time=3.41 m
s
64 bytes from lb-192-30-253-112-iad.github.com (192.30.253.112): icmp_seq=3 ttl=50 time=2.56 m
s
64 bytes from lb-192-30-253-112-iad.github.com (192.30.253.112): icmp_seq=4 ttl=50 time=2.44 m
s
64 bytes from lb-192-30-253-112-iad.github.com (192.30.253.112): icmp_seq=5 ttl=50 time=3.08 m
s

--- github.com ping statistics ---
```

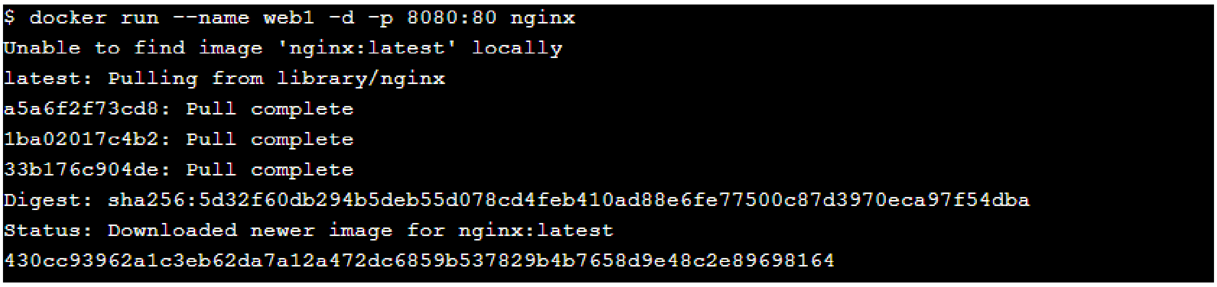


```
--- github.com ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 2.440/2.798/3.412/0.384 ms
```

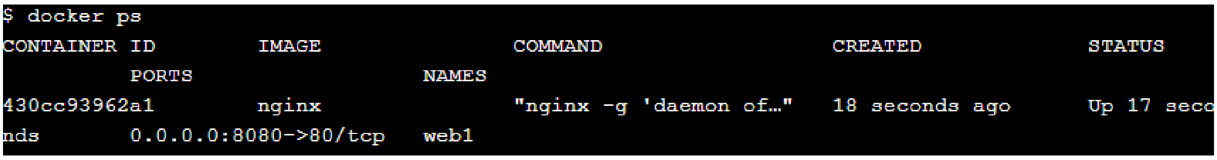


```
$ docker stop 56bdbb58258d
56bdbb58258d
```

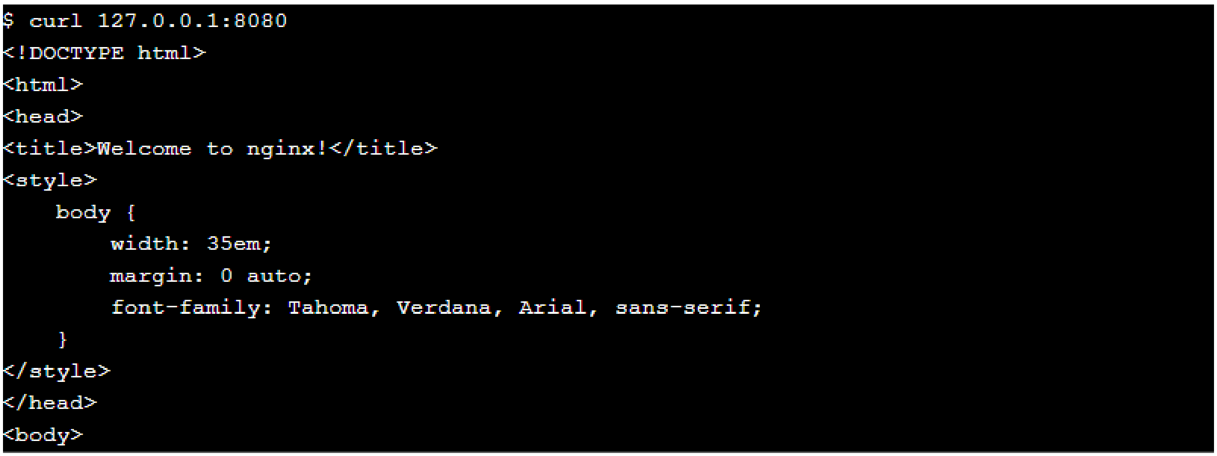
Step 4: Configure NAT for external connectivity



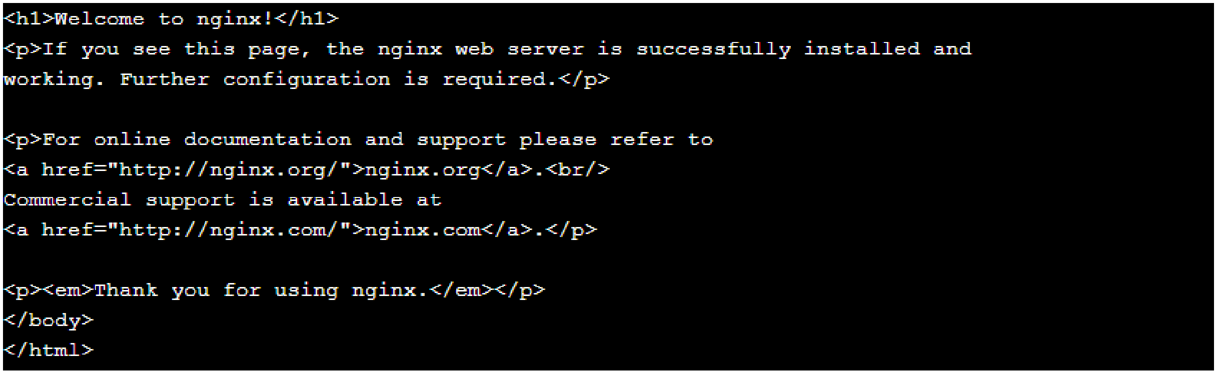
```
$ docker run --name web1 -d -p 8080:80 nginx
Unable to find image 'nginx:latest' locally
latest: Pulling from library/nginx
a5a6f2f73cd8: Pull complete
1ba02017c4b2: Pull complete
33b176c904de: Pull complete
Digest: sha256:5d32f60db294b5deb55d078cd4feb410ad88e6fe77500c87d3970eca97f54dba
Status: Downloaded newer image for nginx:latest
430cc93962a1c3eb62da7a12a472dc6859b537829b4b7658d9e48c2e89698164
```



```
$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS
          PORTS                  NAMES
430cc93962a1        nginx               "nginx -g 'daemon of…"   18 seconds ago      Up 17 seco
nds        0.0.0.0:8080->80/tcp   web1
```



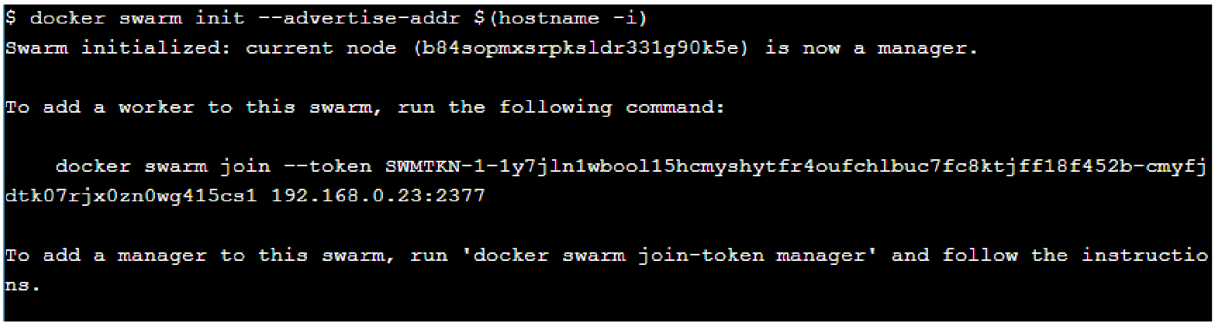
```
$ curl 127.0.0.1:8080
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
    body {
        width: 35em;
        margin: 0 auto;
        font-family: Tahoma, Verdana, Arial, sans-serif;
    }
</style>
</head>
<body>
```



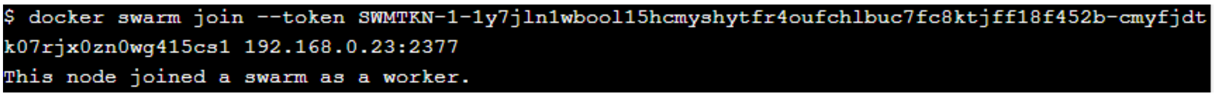
```
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

Section #3 - Overlay Networking

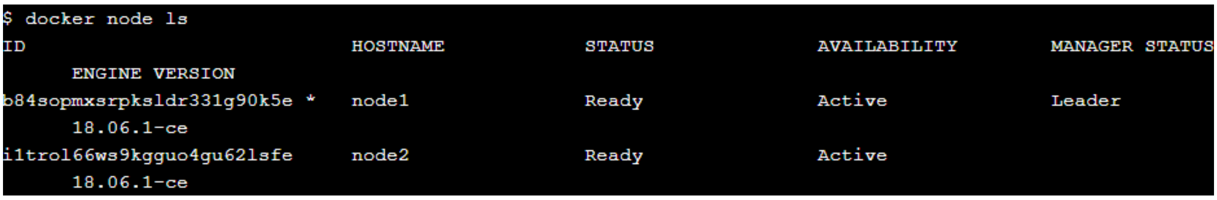
Step 1: The Basics



```
$ docker swarm init --advertise-addr $(hostname -i)
Swarm initialized: current node (b84sopmxsrpksldr331g90k5e) is now a manager.

To add a worker to this swarm, run the following command:

    docker swarm join --token SWMTKN-1-1y7jln1wbool15hcmyshytfr4oufchlbuc7fc8ktjff18f452b-cmyfj
dtk07rjx0zn0wg415cs1 192.168.0.23:2377

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructio
ns.
```

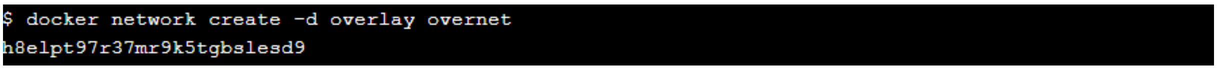


```
$ docker swarm join --token SWMTKN-1-1y7jln1wbool15hcmyshytfr4oufchlbuc7fc8ktjff18f452b-cmyfjdt
k07rjx0zn0wg415cs1 192.168.0.23:2377
This node joined a swarm as a worker.
```

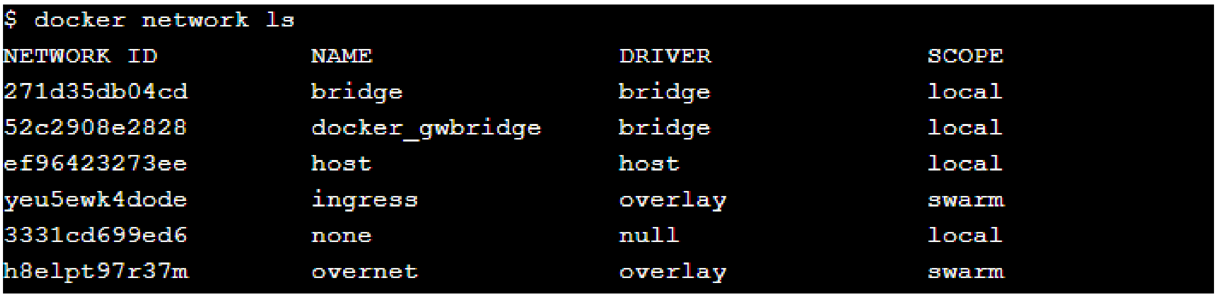


```
$ docker node ls
ID                            HOSTNAME            STATUS              AVAILABILITY        MANAGER STATUS
      ENGINE VERSION
b84sopmxsrpksldr331g90k5e *   node1               Ready               Active              Leader
      18.06.1-ce
i1trol66ws9kgguo4gu62lsfe     node2               Ready               Active
      18.06.1-ce
```

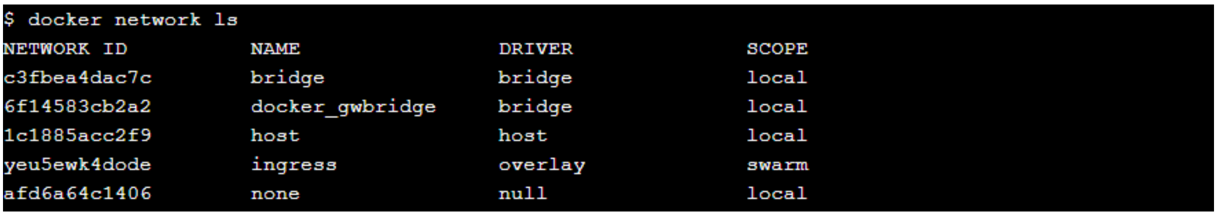
Step 2: Create an overlay network



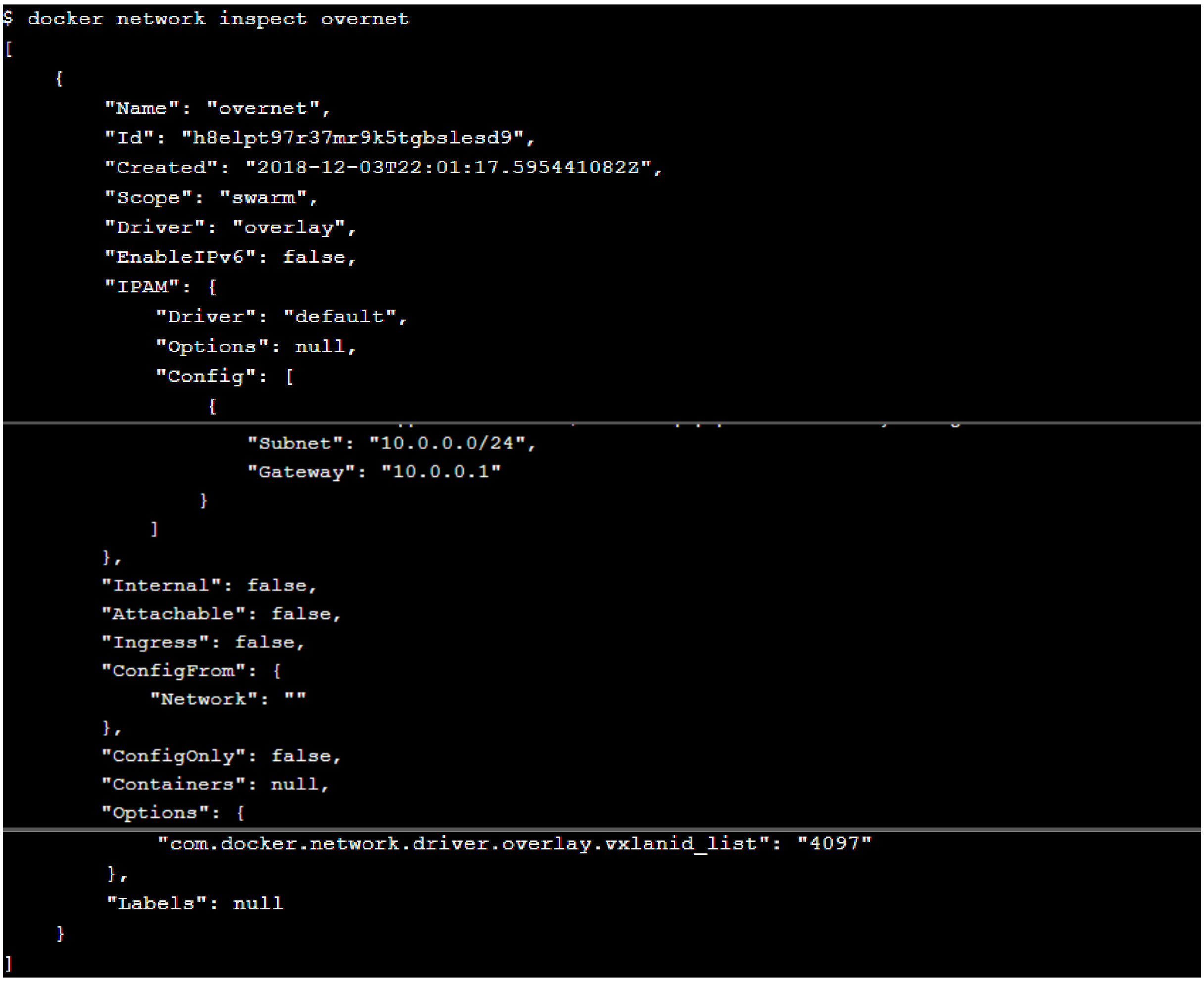
```
$ docker network create -d overlay overnet
h8elpt97r37mr9k5tgbslesd9
```



```
$ docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
271d35db04cd        bridge              bridge              local
52c2908e2828        docker_gwbridge     bridge              local
ef96423273ee        host                host                local
yeu5ewk4dode        ingress             overlay             swarm
3331cd699ed6        none                null                local
h8elpt97r37m        overnet             overlay             swarm
```

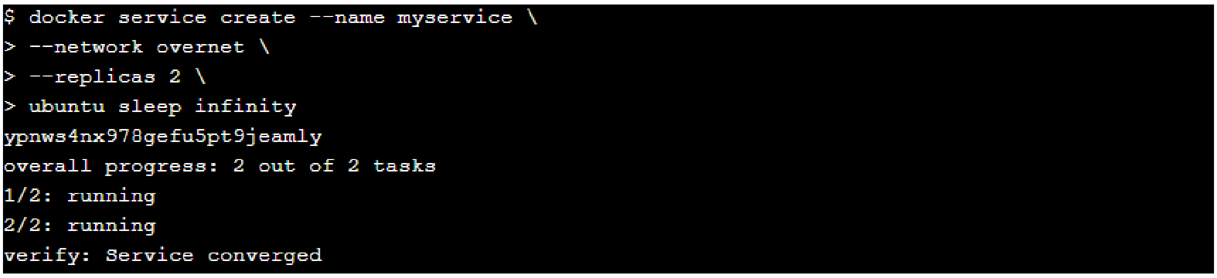


```
$ docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
c3fbea4dac7c        bridge              bridge              local
6f14583cb2a2        docker_gwbridge     bridge              local
1c1885acc2f9        host                host                local
yeu5ewk4dode        ingress             overlay             swarm
afd6a64c1406        none                null                local
```

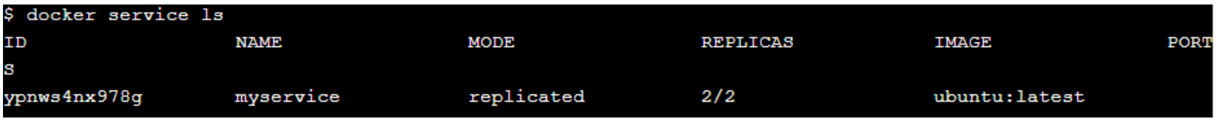


```
$ docker network inspect overnet
[
    {
        "Name": "overnet",
        "Id": "h8elpt97r37mr9k5tgbslesd9",
        "Created": "2018-12-03T22:01:17.595441082Z",
        "Scope": "swarm",
        "Driver": "overlay",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "10.0.0.0/24",
                    "Gateway": "10.0.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": null,
        "Options": {
            "com.docker.network.driver.overlay.vxlanid_list": "4097"
        },
        "Labels": null
    }
]
```

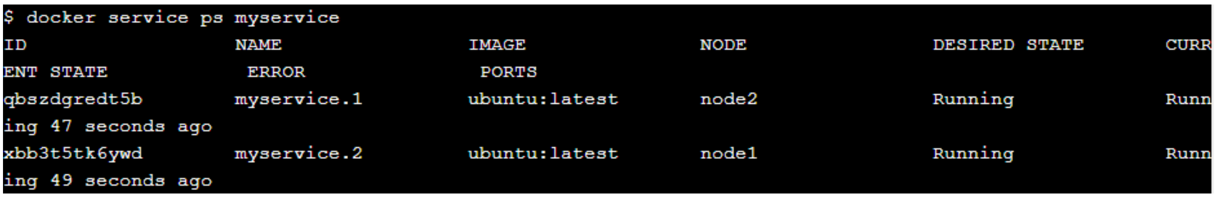
Step 3: Create a service



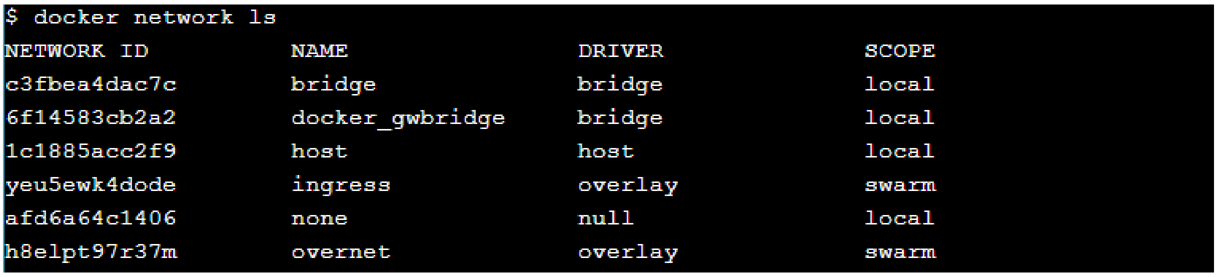
```
$ docker service create --name myservice \
> --network overnet \
> --replicas 2 \
> ubuntu sleep infinity
ypnws4nx978gefu5pt9jeamly
overall progress: 2 out of 2 tasks
1/2: running
2/2: running
verify: Service converged
```



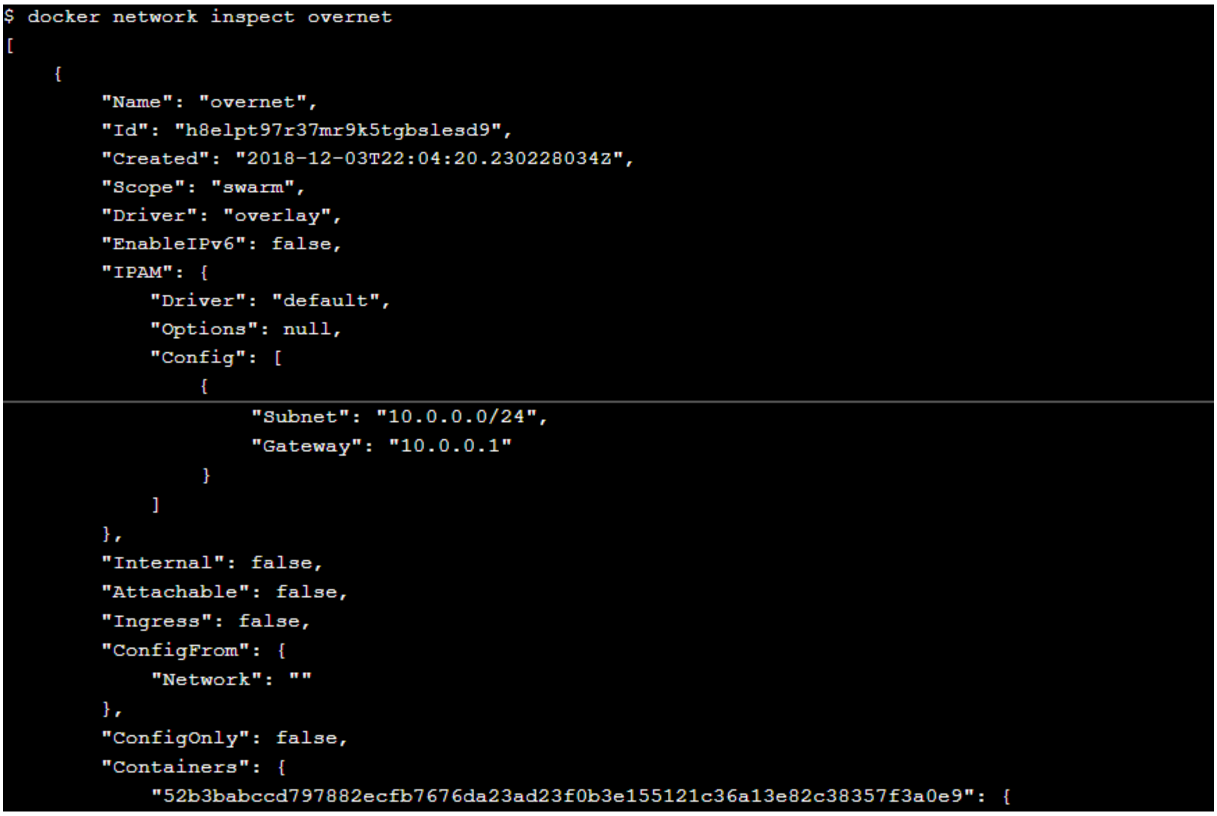
```
$ docker service ls
ID                  NAME                MODE                REPLICAS            IMAGE               PORT
S
ypnws4nx978g        myservice           replicated          2/2                 ubuntu:latest
```



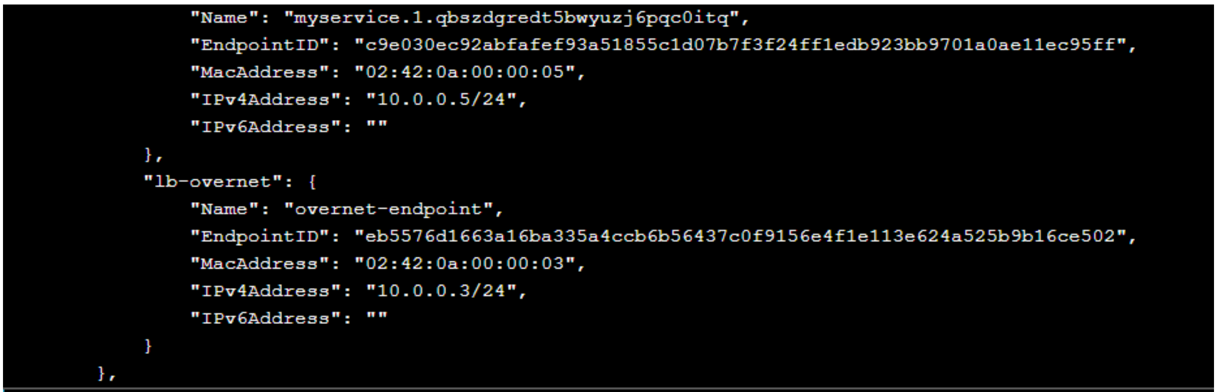
```
$ docker service ps myservice
ID                  NAME                IMAGE               NODE                DESIRED STATE       CURR
ENT STATE            ERROR               PORTS
qbszdgredt5b        myservice.1         ubuntu:latest       node2               Running             Runn
ing 47 seconds ago
xbb3t5tk6ywd        myservice.2         ubuntu:latest       node1               Running             Runn
ing 49 seconds ago
```



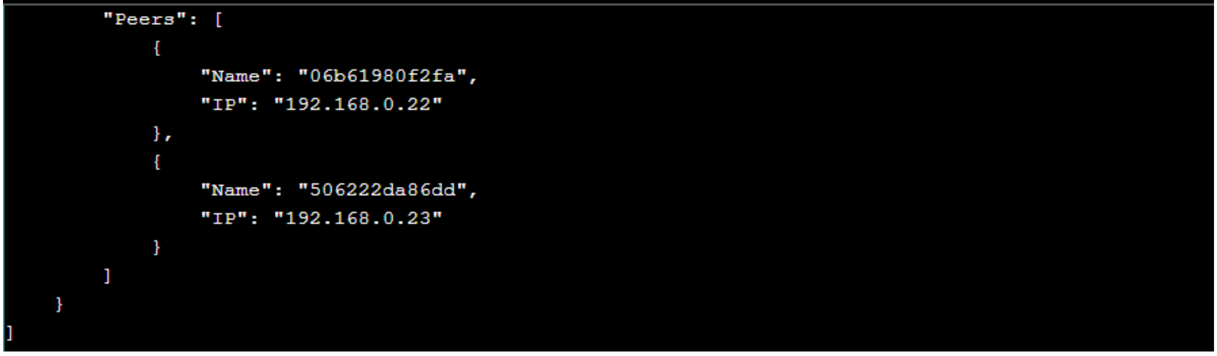
```
$ docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
c3fbea4dac7c        bridge              bridge              local
6f14583cb2a2        docker_gwbridge     bridge              local
1c1885acc2f9        host                host                local
yeu5ewk4dode        ingress             overlay             swarm
afd6a64c1406        none                null                local
h8elpt97r37m        overnet             overlay             swarm
```



```
$ docker network inspect overnet
[
    {
        "Name": "overnet",
        "Id": "h8elpt97r37mr9k5tgbslesd9",
        "Created": "2018-12-03T22:04:20.230228034Z",
        "Scope": "swarm",
        "Driver": "overlay",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "10.0.0.0/24",
                    "Gateway": "10.0.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "52b3babccd797882ecfb7676da23ad23f0b3e155121c36a13e82c38357f3a0e9": {
```

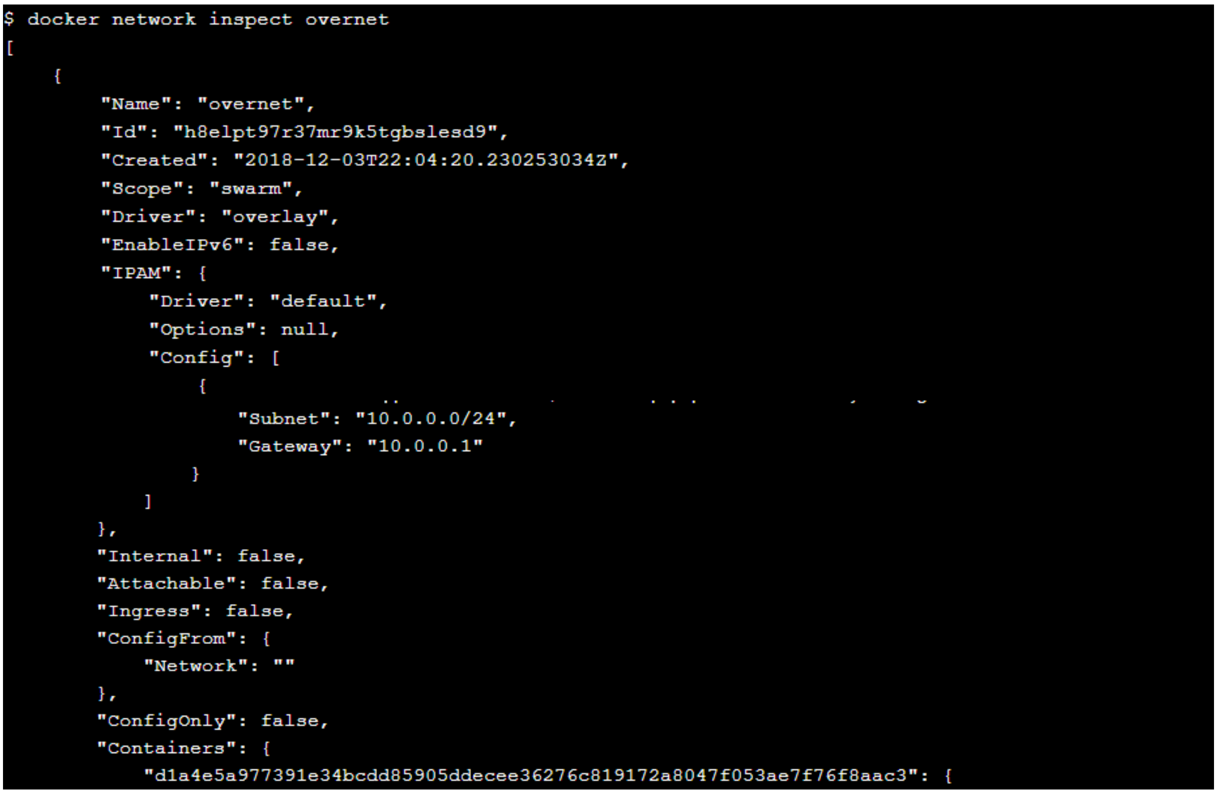


```
                "Name": "myservice.1.qbszdgredt5bwyuzj6pqc0itq",
                "EndpointID": "c9e030ec92abfafef93a51855c1d07b7f3f24ff1edb923bb9701a0ae11ec95ff",
                "MacAddress": "02:42:0a:00:00:05",
                "IPv4Address": "10.0.0.5/24",
                "IPv6Address": ""
            },
            "lb-overnet": {
                "Name": "overnet-endpoint",
                "EndpointID": "eb5576d1663a16ba335a4ccb6b56437c0f9156e4f1e113e624a525b9b16ce502",
                "MacAddress": "02:42:0a:00:00:03",
                "IPv4Address": "10.0.0.3/24",
                "IPv6Address": ""
            }
        },
```

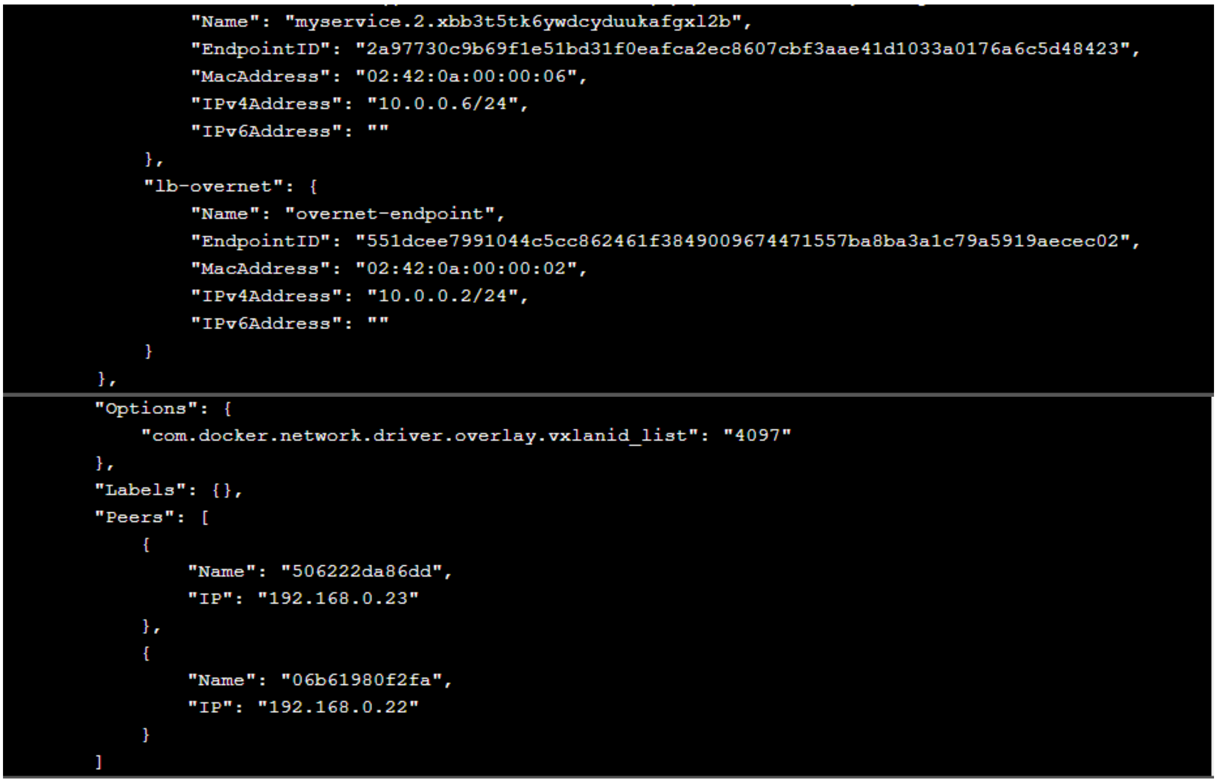


```
        "Peers": [
            {
                "Name": "06b61980f2fa",
                "IP": "192.168.0.22"
            },
            {
                "Name": "506222da86dd",
                "IP": "192.168.0.23"
            }
        ]
    }
]
```

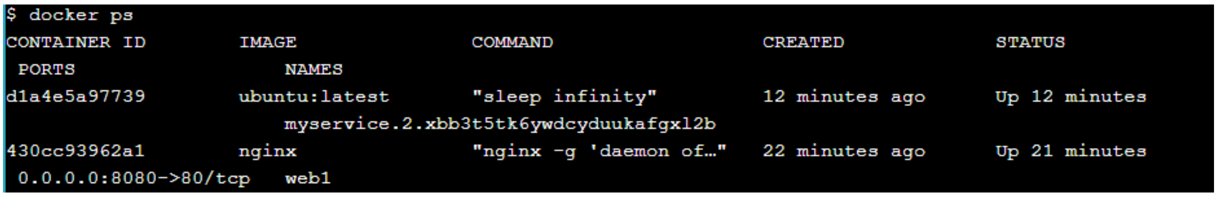
Step 4: Test the network



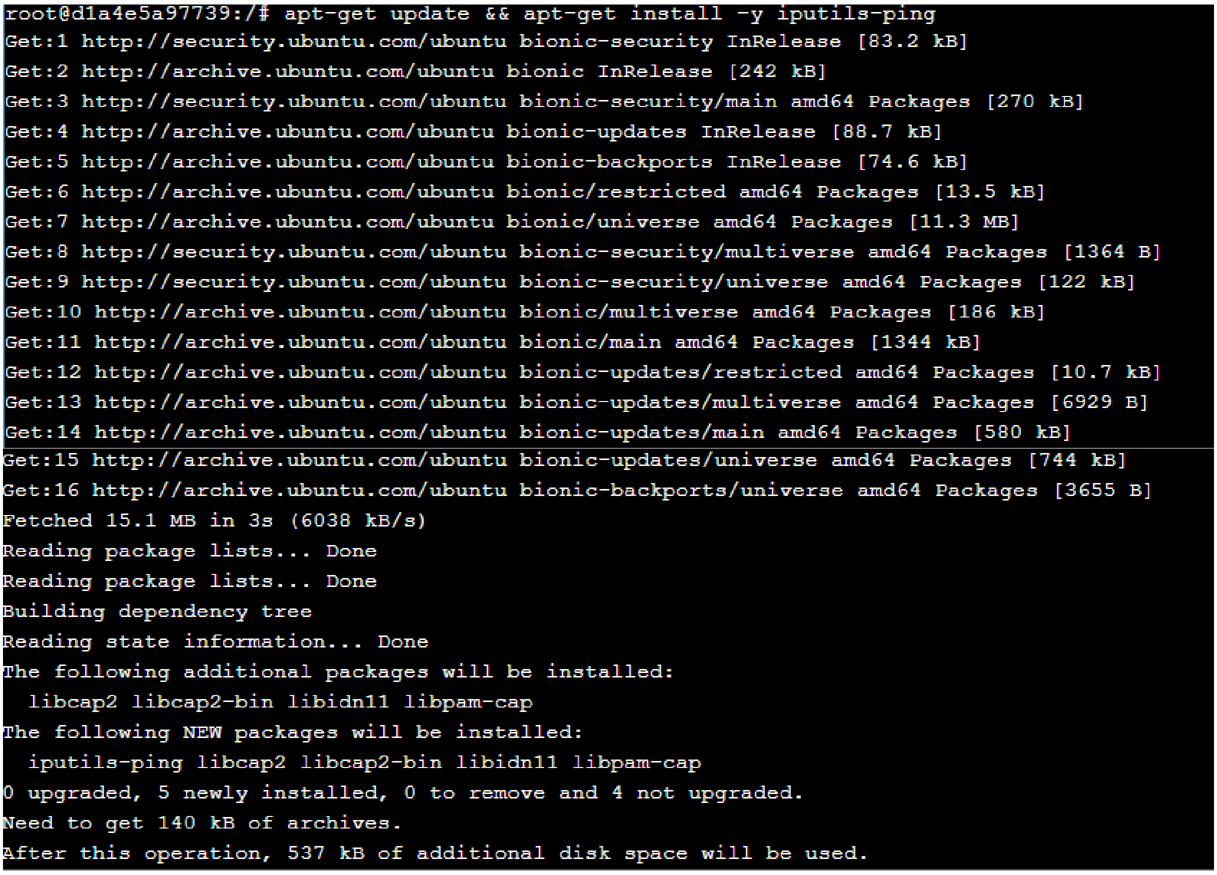
```
$ docker network inspect overnet
[
    {
        "Name": "overnet",
        "Id": "h8elpt97r37mr9k5tgbslesd9",
        "Created": "2018-12-03T22:04:20.230253034Z",
        "Scope": "swarm",
        "Driver": "overlay",
        "EnableIPv6": false,
        "IPAM": {
            "Driver": "default",
            "Options": null,
            "Config": [
                {
                    "Subnet": "10.0.0.0/24",
                    "Gateway": "10.0.0.1"
                }
            ]
        },
        "Internal": false,
        "Attachable": false,
        "Ingress": false,
        "ConfigFrom": {
            "Network": ""
        },
        "ConfigOnly": false,
        "Containers": {
            "d1a4e5a977391e34bcdd85905ddecee36276c819172a8047f053ae7f76f8aac3": {
```



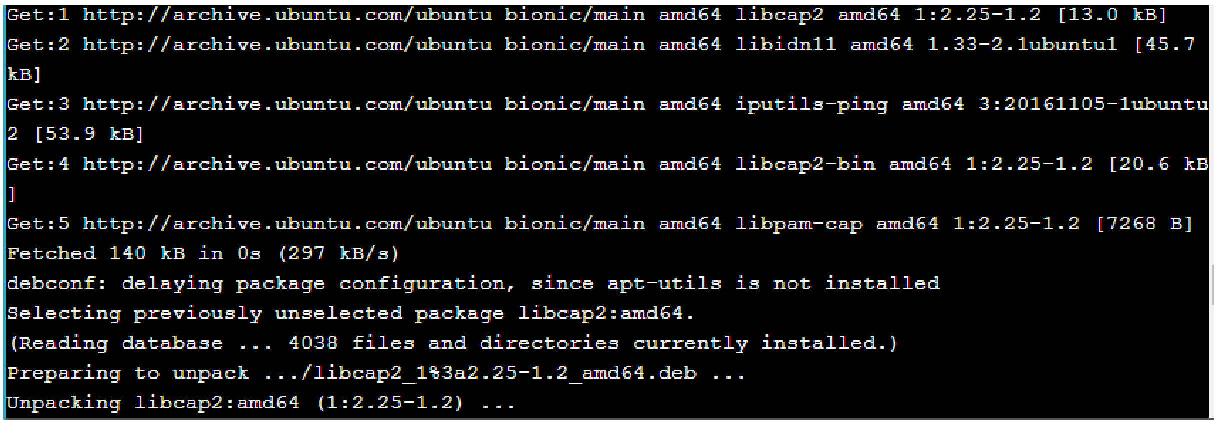
```
                "Name": "myservice.2.xbb3t5tk6ywdcyduukafgxl2b",
                "EndpointID": "2a97730c9b69f1e51bd31f0eafca2ec8607cbf3aae41d1033a0176a6c5d48423",
                "MacAddress": "02:42:0a:00:00:06",
                "IPv4Address": "10.0.0.6/24",
                "IPv6Address": ""
            },
            "lb-overnet": {
                "Name": "overnet-endpoint",
                "EndpointID": "551dcee7991044c5cc862461f3849009674471557ba8ba3a1c79a5919aecec02",
                "MacAddress": "02:42:0a:00:00:02",
                "IPv4Address": "10.0.0.2/24",
                "IPv6Address": ""
            }
        },
        "Options": {
            "com.docker.network.driver.overlay.vxlanid_list": "4097"
        },
        "Labels": {},
        "Peers": [
            {
                "Name": "506222da86dd",
                "IP": "192.168.0.23"
            },
            {
                "Name": "06b61980f2fa",
                "IP": "192.168.0.22"
            }
        ]
```



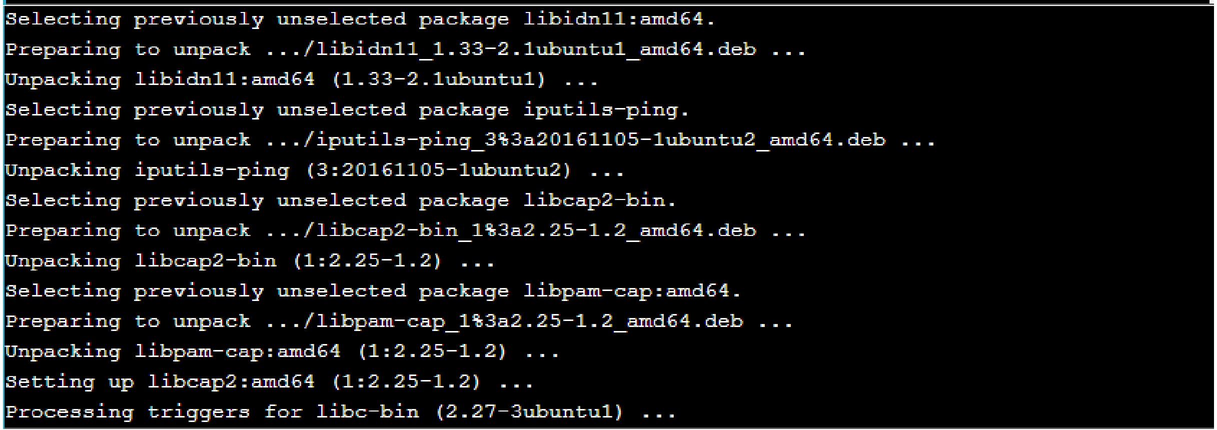
```
$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS
 PORTS                  NAMES
d1a4e5a97739        ubuntu:latest       "sleep infinity"         12 minutes ago      Up 12 minutes
                        myservice.2.xbb3t5tk6ywdcyduukafgxl2b
430cc93962a1        nginx               "nginx -g 'daemon of…"   22 minutes ago      Up 21 minutes
 0.0.0.0:8080->80/tcp   web1
```



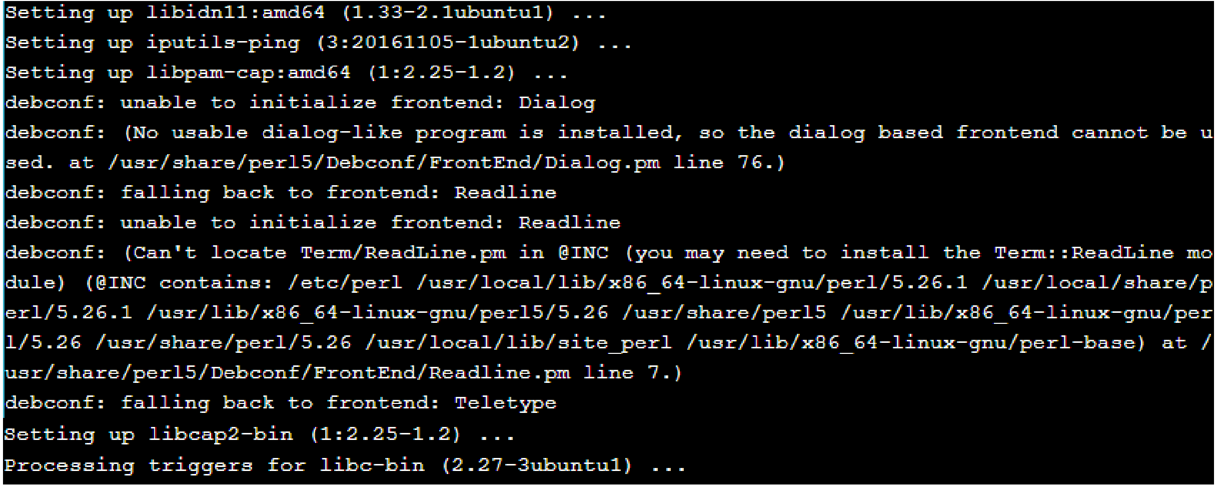
```
root@d1a4e5a97739:/# apt-get update && apt-get install -y iputils-ping
Get:1 http://security.ubuntu.com/ubuntu bionic-security InRelease [83.2 kB]
Get:2 http://archive.ubuntu.com/ubuntu bionic InRelease [242 kB]
Get:3 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [270 kB]
Get:4 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:5 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:6 http://archive.ubuntu.com/ubuntu bionic/restricted amd64 Packages [13.5 kB]
Get:7 http://archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [11.3 MB]
Get:8 http://security.ubuntu.com/ubuntu bionic-security/multiverse amd64 Packages [1364 B]
Get:9 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 Packages [122 kB]
Get:10 http://archive.ubuntu.com/ubuntu bionic/multiverse amd64 Packages [186 kB]
Get:11 http://archive.ubuntu.com/ubuntu bionic/main amd64 Packages [1344 kB]
Get:12 http://archive.ubuntu.com/ubuntu bionic-updates/restricted amd64 Packages [10.7 kB]
Get:13 http://archive.ubuntu.com/ubuntu bionic-updates/multiverse amd64 Packages [6929 B]
Get:14 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [580 kB]
Get:15 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [744 kB]
Get:16 http://archive.ubuntu.com/ubuntu bionic-backports/universe amd64 Packages [3655 B]
Fetched 15.1 MB in 3s (6038 kB/s)
Reading package lists... Done
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libcap2 libcap2-bin libidn11 libpam-cap
The following NEW packages will be installed:
  iputils-ping libcap2 libcap2-bin libidn11 libpam-cap
0 upgraded, 5 newly installed, 0 to remove and 4 not upgraded.
Need to get 140 kB of archives.
After this operation, 537 kB of additional disk space will be used.
```



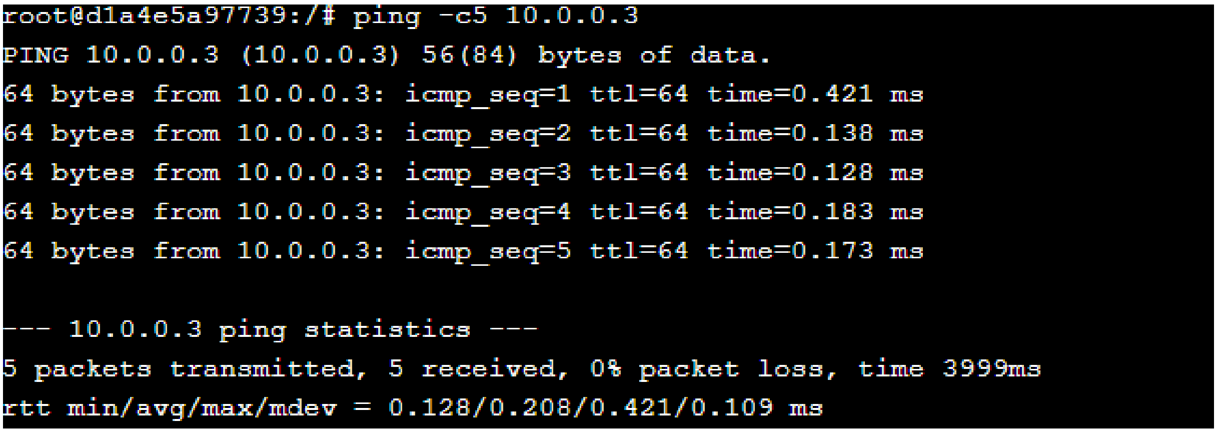
```
Get:1 http://archive.ubuntu.com/ubuntu bionic/main amd64 libcap2 amd64 1:2.25-1.2 [13.0 kB]
Get:2 http://archive.ubuntu.com/ubuntu bionic/main amd64 libidn11 amd64 1.33-2.1ubuntu1 [45.7
kB]
Get:3 http://archive.ubuntu.com/ubuntu bionic/main amd64 iputils-ping amd64 3:20161105-1ubuntu
2 [53.9 kB]
Get:4 http://archive.ubuntu.com/ubuntu bionic/main amd64 libcap2-bin amd64 1:2.25-1.2 [20.6 kB
]
Get:5 http://archive.ubuntu.com/ubuntu bionic/main amd64 libpam-cap amd64 1:2.25-1.2 [7268 B]
Fetched 140 kB in 0s (297 kB/s)
debconf: delaying package configuration, since apt-utils is not installed
Selecting previously unselected package libcap2:amd64.
(Reading database ... 4038 files and directories currently installed.)
Preparing to unpack .../libcap2_1%3a2.25-1.2_amd64.deb ...
Unpacking libcap2:amd64 (1:2.25-1.2) ...
```



```
Selecting previously unselected package libidn11:amd64.
Preparing to unpack .../libidn11_1.33-2.1ubuntu1_amd64.deb ...
Unpacking libidn11:amd64 (1.33-2.1ubuntu1) ...
Selecting previously unselected package iputils-ping.
Preparing to unpack .../iputils-ping_3%3a20161105-1ubuntu2_amd64.deb ...
Unpacking iputils-ping (3:20161105-1ubuntu2) ...
Selecting previously unselected package libcap2-bin.
Preparing to unpack .../libcap2-bin_1%3a2.25-1.2_amd64.deb ...
Unpacking libcap2-bin (1:2.25-1.2) ...
Selecting previously unselected package libpam-cap:amd64.
Preparing to unpack .../libpam-cap_1%3a2.25-1.2_amd64.deb ...
Unpacking libpam-cap:amd64 (1:2.25-1.2) ...
Setting up libcap2:amd64 (1:2.25-1.2) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
```

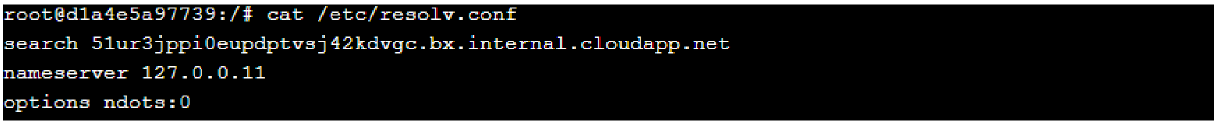


```
Setting up libidn11:amd64 (1.33-2.1ubuntu1) ...
Setting up iputils-ping (3:20161105-1ubuntu2) ...
Setting up libpam-cap:amd64 (1:2.25-1.2) ...
debconf: unable to initialize frontend: Dialog
debconf: (No usable dialog-like program is installed, so the dialog based frontend cannot be u
sed. at /usr/share/perl5/Debconf/FrontEnd/Dialog.pm line 76.)
debconf: falling back to frontend: Readline
debconf: unable to initialize frontend: Readline
debconf: (Can't locate Term/ReadLine.pm in @INC (you may need to install the Term::ReadLine mo
dule) (@INC contains: /etc/perl /usr/local/lib/x86_64-linux-gnu/perl/5.26.1 /usr/local/share/p
erl/5.26.1 /usr/lib/x86_64-linux-gnu/perl5/5.26 /usr/share/perl5 /usr/lib/x86_64-linux-gnu/per
l/5.26 /usr/share/perl/5.26 /usr/local/lib/site_perl /usr/lib/x86_64-linux-gnu/perl-base) at /
usr/share/perl5/Debconf/FrontEnd/Readline.pm line 7.)
debconf: falling back to frontend: Teletype
Setting up libcap2-bin (1:2.25-1.2) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
```

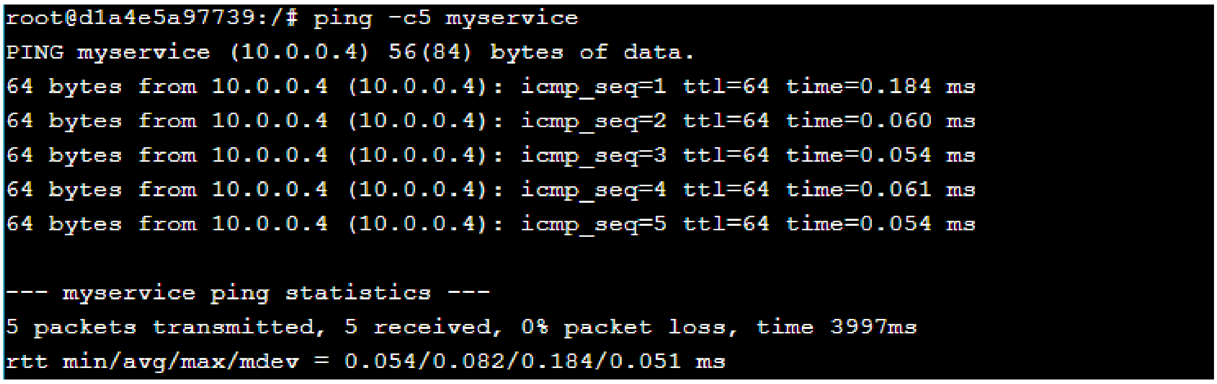


```
root@d1a4e5a97739:/# ping -c5 10.0.0.3
PING 10.0.0.3 (10.0.0.3) 56(84) bytes of data.
64 bytes from 10.0.0.3: icmp_seq=1 ttl=64 time=0.421 ms
64 bytes from 10.0.0.3: icmp_seq=2 ttl=64 time=0.138 ms
64 bytes from 10.0.0.3: icmp_seq=3 ttl=64 time=0.128 ms
64 bytes from 10.0.0.3: icmp_seq=4 ttl=64 time=0.183 ms
64 bytes from 10.0.0.3: icmp_seq=5 ttl=64 time=0.173 ms

--- 10.0.0.3 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3999ms
rtt min/avg/max/mdev = 0.128/0.208/0.421/0.109 ms
```

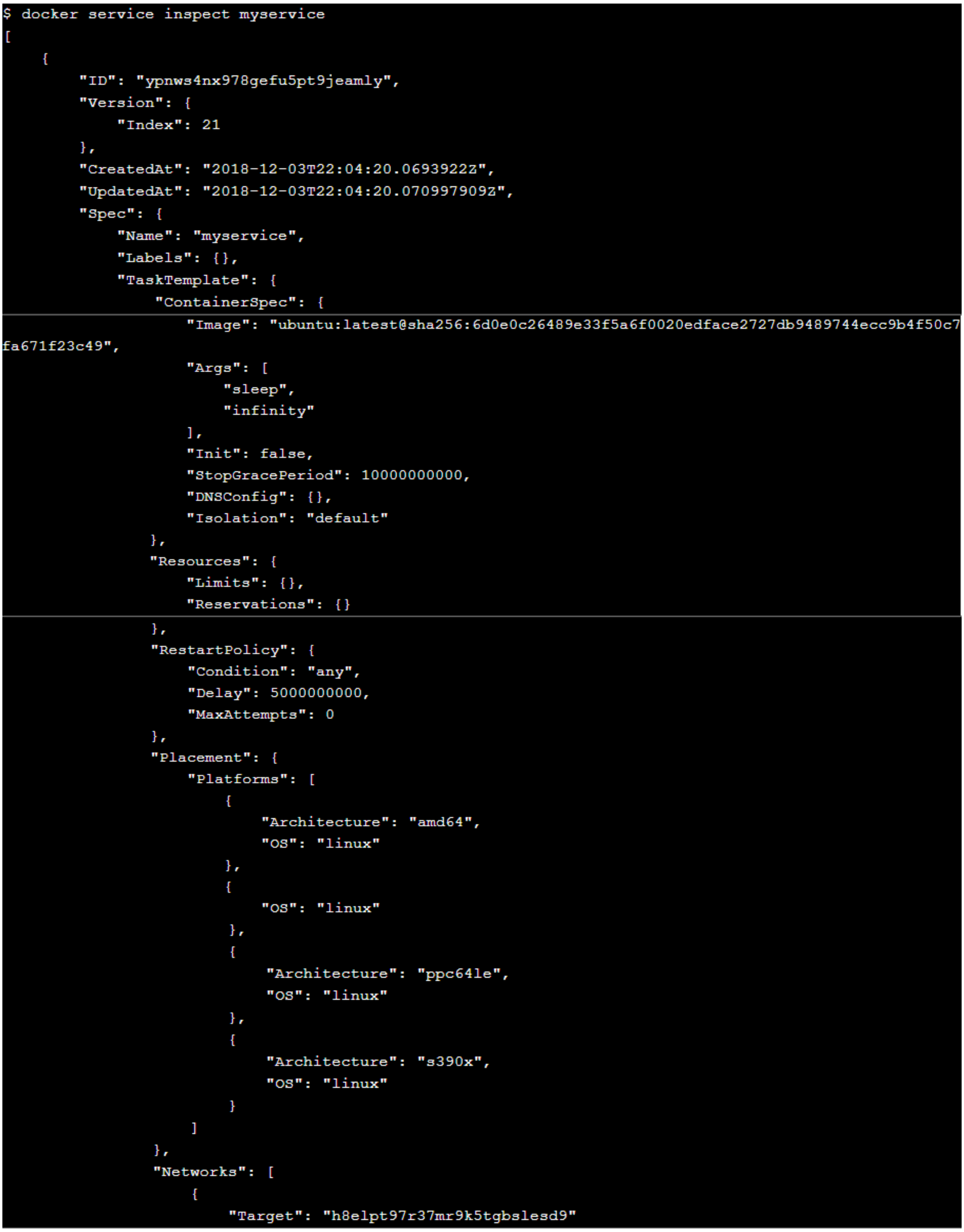
Step 5: Test service discovery



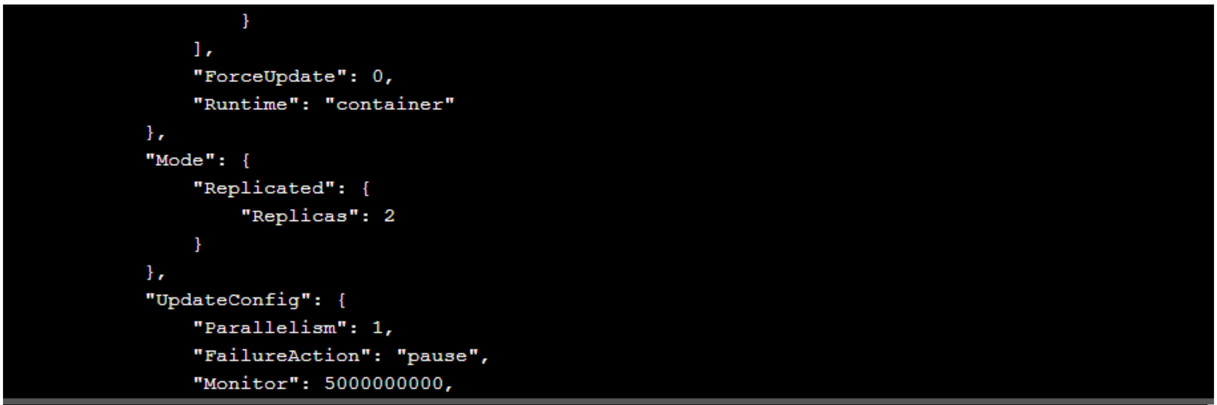
```
root@d1a4e5a97739:/# cat /etc/resolv.conf
search 51ur3jppi0eupdptvsj42kdvgc.bx.internal.cloudapp.net
nameserver 127.0.0.11
options ndots:0
```



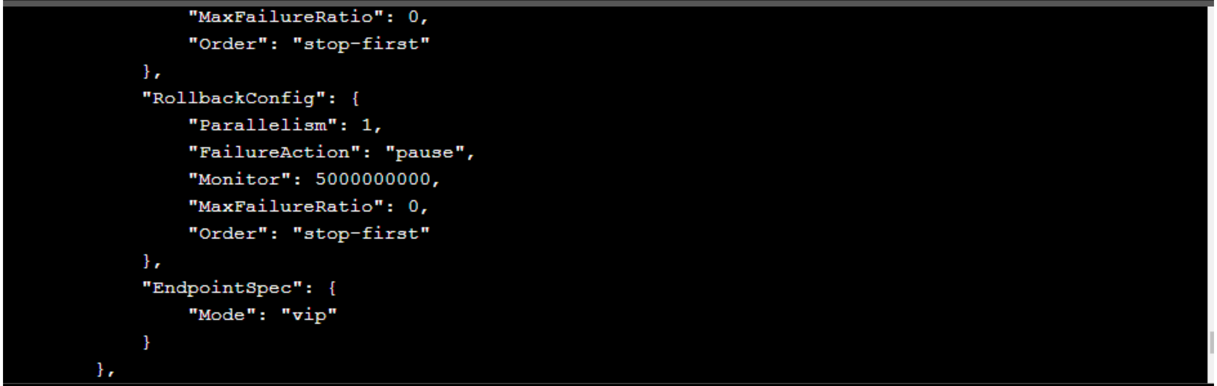
```
root@d1a4e5a97739:/# ping -c5 myservice
PING myservice (10.0.0.4) 56(84) bytes of data.
64 bytes from 10.0.0.4 (10.0.0.4): icmp_seq=1 ttl=64 time=0.184 ms
64 bytes from 10.0.0.4 (10.0.0.4): icmp_seq=2 ttl=64 time=0.060 ms
64 bytes from 10.0.0.4 (10.0.0.4): icmp_seq=3 ttl=64 time=0.054 ms
64 bytes from 10.0.0.4 (10.0.0.4): icmp_seq=4 ttl=64 time=0.061 ms
64 bytes from 10.0.0.4 (10.0.0.4): icmp_seq=5 ttl=64 time=0.054 ms

--- myservice ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3997ms
rtt min/avg/max/mdev = 0.054/0.082/0.184/0.051 ms
```



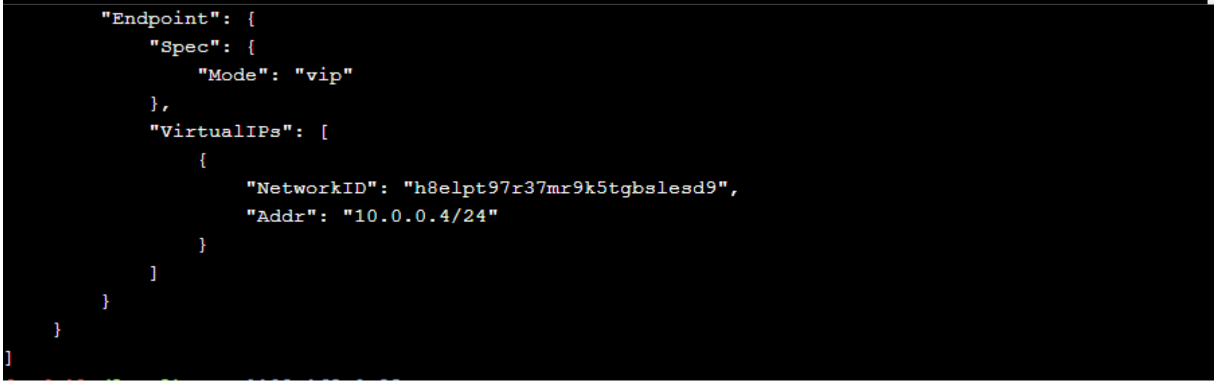
```
$ docker service inspect myservice
[
    {
        "ID": "ypnws4nx978gefu5pt9jeamly",
        "Version": {
            "Index": 21
        },
        "CreatedAt": "2018-12-03T22:04:20.0693922Z",
        "UpdatedAt": "2018-12-03T22:04:20.070997909Z",
        "Spec": {
            "Name": "myservice",
            "Labels": {},
            "TaskTemplate": {
                "ContainerSpec": {
                    "Image": "ubuntu:latest@sha256:6d0e0c26489e33f5a6f0020edface2727db9489744ecc9b4f50c7
fa671f23c49",
                    "Args": [
                        "sleep",
                        "infinity"
                    ],
                    "Init": false,
                    "StopGracePeriod": 10000000000,
                    "DNSConfig": {},
                    "Isolation": "default"
                },
                "Resources": {
                    "Limits": {},
                    "Reservations": {}
                },
                "RestartPolicy": {
                    "Condition": "any",
                    "Delay": 5000000000,
                    "MaxAttempts": 0
                },
                "Placement": {
                    "Platforms": [
                        {
                            "Architecture": "amd64",
                            "OS": "linux"
                        },
                        {
                            "OS": "linux"
                        },
                        {
                            "Architecture": "ppc64le",
                            "OS": "linux"
                        },
                        {
                            "Architecture": "s390x",
                            "OS": "linux"
                        }
                    ]
                },
                "Networks": [
                    {
                        "Target": "h8elpt97r37mr9k5tgbslesd9"
```



```
                    }
                ],
                "ForceUpdate": 0,
                "Runtime": "container"
            },
            "Mode": {
                "Replicated": {
                    "Replicas": 2
                }
            },
            "UpdateConfig": {
                "Parallelism": 1,
                "FailureAction": "pause",
                "Monitor": 5000000000,
```

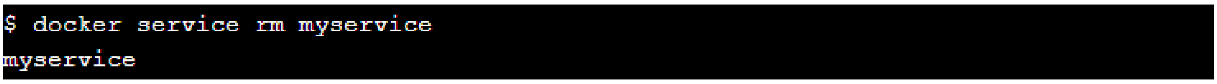


```
                "MaxFailureRatio": 0,
                "Order": "stop-first"
            },
            "RollbackConfig": {
                "Parallelism": 1,
                "FailureAction": "pause",
                "Monitor": 5000000000,
                "MaxFailureRatio": 0,
                "Order": "stop-first"
            },
            "EndpointSpec": {
                "Mode": "vip"
            }
        },
```

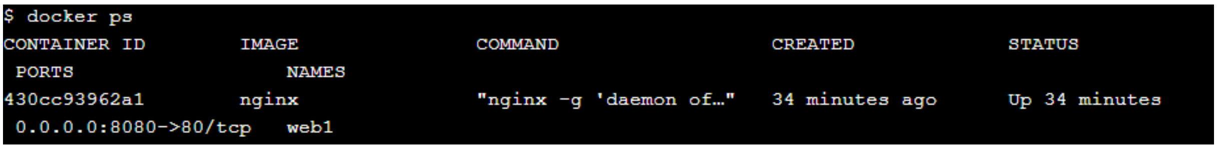


```
        "Endpoint": {
            "Spec": {
                "Mode": "vip"
            },
            "VirtualIPs": [
                {
                    "NetworkID": "h8elpt97r37mr9k5tgbslesd9",
                    "Addr": "10.0.0.4/24"
                }
            ]
        }
    }
]
```

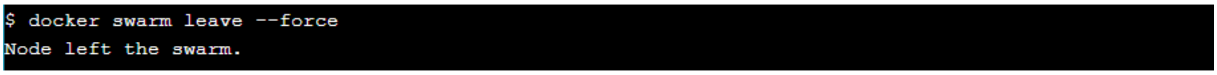
Cleaning Up



```
$ docker service rm myservice
myservice
```



```
$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS
 PORTS                  NAMES
430cc93962a1        nginx               "nginx -g 'daemon of…"   34 minutes ago      Up 34 minutes
 0.0.0.0:8080->80/tcp   web1
```



```
$ docker swarm leave --force
Node left the swarm.
```