

Sprawozdanie z *PaaS*

Docker Networking Hands-on Lab

Joanna Szewczyk
Nr albumu: 195220

W niektórych miejscach pominęłam część wyników poleceń, gdyż na tak małym ekranie, jaki udostępnia Docker niektóre wyniki musiałyby być umieszczane na wielu screenshot'ach. W miejscach, w których coś jest pominięte, jest umieszczona odpowiednia informacja.

```
If the commandline doesn't appear in the terminal, make sure popups are enabled or try resizing the browser window.

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

#
# The PWD team.
#####
[node1] (local) root@192.168.0.23 ~
$ docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
6ef0491dc0f9        bridge              bridge              local
6d938c77db3a        host                host                local
29d06590990b        none                null                local
[node1] (local) root@192.168.0.23 ~
$

$ docker network inspect bridge
[
  {
    "Name": "bridge",
    "Id": "6ef0491dc0f9a91451e9340224f3c7bdf38b2c7b4d4bbb94f79d89becb2667ea",
    "Created": "2018-12-01T18:58:59.926979981Z",
    "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": {}
}
}

$ 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
[node1] (local) root@192.168.0.23 ~
$ apk add bridge
(1/1) Installing bridge (1.5-r3)
OK: 302 MiB in 111 packages
[node1] (local) root@192.168.0.23 ~
$ 
$ brctl show
bridge name      bridge id        STP enabled      interfaces
docker0          8000.024202bf3bdd  no
[node1] (local) root@192.168.0.23 ~
$ 
$ 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
74c9036a9caef48747ec7d1e4a13cbf4706a0d267a9360cd32e4c4057e29f300
[node1] (local) root@192.168.0.23 ~
$ 

```

```

$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED
STATUS             PORTS              NAMES
74c9036a9cae       ubuntu             "sleep infinity"   41 seconds ago
Up 37 seconds              pensive_nobel
[node1] (local) root@192.168.0.23 ~

$
$ brctl show
bridge name      bridge id          STP enabled        interfaces
docker0          8000.024202bf3bdd  no                 veth1f2bd61
[node1] (local) root@192.168.0.23 ~

$
$ docker network inspect bridge
[
  {
    "Name": "bridge",
    "Id": "6ef0491dc0f9a91451e9340224f3c7bdf38b2c7b4d4bbb94f79d89becb2667ea"
  },
  {
    "Created": "2018-12-01T18:58:59.926979981Z",
    "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": {
      "74c9036a9cae48747ec7d1e4a13cbf4706a0d267a9360cd32e4c4057e29f300":
      {
        "Name": "pensive_nobel",
        "EndpointID": "7c0126b568fb6f19b1816e191e091cae2d73488c01cfa3754401622e956fe966",
        "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": {}
}
]
$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
74c9036a9cae        ubuntu             "sleep infinity"   8 minutes ago      Up 8 minutes        0.0.0.0:80->80/tcp   pensive_nobel
[node1] (local) root@192.168.0.23 ~
$
$ docker exec -it 74c9036a9cae /bin/bash
root@74c9036a9cae:/#
root@74c9036a9cae:/# apt-get update && apt-get install -y iputils-ping
Get:1 http://archive.ubuntu.com/ubuntu bionic InRelease [242 kB]
Get:2 http://security.ubuntu.com/ubuntu bionic-security InRelease [83.2 kB]
Get:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:4 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:5 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 Packages [122 kB]
Get:6 http://archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [11.3 MB]
Get:7 http://security.ubuntu.com/ubuntu bionic-security/multiverse amd64 Packages [1364 B]
Get:8 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [266 k

```

Tutaj dość dużo linii instalacji, końcówka:

```

cannot be used. 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 module) (@INC contains: /etc/perl /usr/local/lib/x86_64-linux-gnu/perl/5
.26.1 /usr/local/share/perl/5.26.1 /usr/lib/x86_64-linux-gnu/perl5/5.26 /usr/share
/perl5 /usr/lib/x86_64-linux-gnu/perl/5.26 /usr/share/perl/5.26 /usr/local/lib/sit
e_perl /usr/lib/x86_64-linux-gnu/perl-base) at /usr/share/perl5/Debconf/FrontEnd/R
eadline.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) ...
$ ping -c5 www.github.com
PING www.github.com (192.30.253.112): 56 data bytes
64 bytes from 192.30.253.112: seq=0 ttl=51 time=1.867 ms
64 bytes from 192.30.253.112: seq=1 ttl=51 time=1.820 ms
64 bytes from 192.30.253.112: seq=2 ttl=51 time=1.965 ms
64 bytes from 192.30.253.112: seq=3 ttl=51 time=1.911 ms
64 bytes from 192.30.253.112: seq=4 ttl=51 time=1.970 ms

--- www.github.com ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 1.820/1.906/1.970 ms
root@74c9036a9cae:/# exit
exit
[node1] (local) root@192.168.0.23 ~
$ docker stop 74c9036a9cae
74c9036a9cae
$ 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
d269e0cle9fd4a0a4b97fb39304a1c6453297985767cb1e39927db0d6e495c62
[node1] (local) root@192.168.0.23 ~
$ docker ps
CONTAINER ID          IMAGE          COMMAND                  CREATED
STATUS              PORTS         NAMES
d269e0cle9fd         nginx         "nginx -g 'daemon of..." About a minute ago
Up About a minute    0.0.0.0:8080->80/tcp    web1
[node1] (local) root@192.168.0.23 ~
$ 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>

```

Tutaj dużo linijek kodu HTML

```
docker swarm init --advertise-addr $(hostname -i)
Swarm initialized: current node (flb3iwpbpu4bpf9p0dhgwje0) is now a manager.

$ docker node ls
ID                                     HOSTNAME          STATUS          AVAILABILITY
MANAGER STATUS                       ENGINE VERSION
flb3iwpbpu4bpf9p0dhgwje0 *          node1             Ready           Active
Leader                               18.06.1-ce

$ docker network create -d overlay overnet
j065b1c1wm3o39hvvcohmmloru

$ docker network ls
NETWORK ID          NAME                DRIVER            SCOPE
889c9c0aa99c        bridge              bridge            local
7860a6e0d29b        docker_gwbridge     bridge            local
20e78ce11e18        host                host              local
v5kq0pjgpbphc       ingress             overlay           swarm
e983aa7300de         none                null              local
j065b1c1wm3o         overnet             overlay           swarm

$ docker network inspect overnet
[
  {
    "Name": "overnet",
    "Id": "j065b1c1wm3o39hvvcohmmloru",
    "Created": "2018-12-01T20:58:52.829973709Z",
    "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
  }
]
```

```

$ docker service create --name myservice \
> --network overnet \
> --replicas 2 \
> ubuntu sleep infinity
wefgamudgx0s835ltbf67hrw1
overall progress: 2 out of 2 tasks
1/2: running
2/2: running
verify: Service converged
$ docker service ls

```

ID	NAME	MODE	REPLICAS
wefgamudgx0s	myservice	replicated	2/2

```

buntu:latest
$ docker service ps myservice

```

ID	NAME	IMAGE	NODE
z5brau5bvykz	myservice.1	ubuntu:latest	node1
vhrrf0glre58j	myservice.2	ubuntu:latest	node1

```

$ docker network ls

```

NETWORK ID	NAME	DRIVER	SCOPE
dc8b7e909e80	bridge	bridge	local
71191c3e46a9	host	host	local
72055818da31	none	null	local

```

$ docker network inspect overnet
[
  {
    "Name": "overnet",
    "Id": "ul60pdf7a3y67r51j5zspge0d",
    "Created": "2018-12-01T21:25:35.095716476Z",
    "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": {

```



```

    },
    "ConfigOnly": false,
    "Containers": {
      "0bc8c6f8c4c8ecddb5ba90f3a63ee785a7cb3d85ca4e5b8f31f180f5c39b22d": {
        "Name": "myservice.1.z5brau5bvykzrm89qwiki35u03",
        "EndpointID": "92ed79f2bfa038961b775b6a13e05b398b62e85ba88b5b7bb2
9d71e1970c477d",
        "MacAddress": "02:42:0a:00:00:04",
        "IPv4Address": "10.0.0.4/24",
        "IPv6Address": ""
      },
      "c974daf71406b8c3cc35da21a2d8e2268bf066ed18dcd3560787f61cce3c7851": {
        "Name": "myservice.2.vhrf0glre58jhbssqzlr3uo7",
        "EndpointID": "ca263fb16e9c3a413e79c16ded51ac3300e48bfe5e8d6b2798
8db8fe22c03e8f",
        "MacAddress": "02:42:0a:00:00:05",
        "IPv4Address": "10.0.0.5/24",
        "IPv6Address": ""
      },
      "lb-overnet": {
        "Name": "overnet-endpoint",
        "EndpointID": "29809b7f5d2786c586edcb45aa2308c69557d805042bab9ee5
4411d1a6427e0e",
        "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": "44370e67992b",
        "IP": "192.168.0.23"
      }
    ]
  }
}

```

```

$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED
STATUS             PORTS              NAMES
0bc8c6f8c4c8       ubuntu:latest      "sleep infinity"        6 minutes ago
Up 6 minutes                myservice.1.z5brau5bvykzrm89qwiki35u03
c974daf71406       ubuntu:latest      "sleep infinity"        6 minutes ago
Up 6 minutes                myservice.2.vhrf0glre58jhbssqzlr3uo7
4f9cf81d8696       nginx              "nginx -g 'daemon of..." 10 minutes ago
Up 10 minutes            0.0.0.0:8080->80/tcp    web1
a4e52d75393e       ubuntu             "sleep infinity"        20 minutes ago
Up 19 minutes                hardcore_shaw

```

```

$ docker exec -it a4e52d75393e /bin/bash
root@a4e52d75393e:/# apt-get update && apt-get install -y iputils-ping
Hit:1 http://archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://security.ubuntu.com/ubuntu bionic-security InRelease
Hit:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:4 http://archive.ubuntu.com/ubuntu bionic-backports InRelease
Reading package lists... Done
Reading package lists... Done
Building dependency tree
Reading state information... Done
iputils-ping is already the newest version (3:20161105-1ubuntu2).
0 upgraded, 0 newly installed, 0 to remove and 3 not upgraded.
root@a4e52d75393e:/# 
[node1] (local) root@192.168.0.23 ~
$ docker service inspect myservice
[
  {
    "ID": "wefgamudgx0s835ltbf67hrw1",
    "Version": {
      "Index": 15
    },
    "CreatedAt": "2018-12-01T21:25:34.930281651Z",
    "UpdatedAt": "2018-12-01T21:25:34.931137455Z",
    "Spec": {
      "Name": "myservice",
      "Labels": {},

```

Tutaj dość dużo linii tekstu

```

[node1] (local) root@192.168.0.23 ~
$ docker service rm myservice
myservice
[node1] (local) root@192.168.0.23 ~
$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED
STATUS            PORTS              NAMES
4f9cf81d8696       nginx              "nginx -g 'daemon of..." 20 minutes ago
Up 20 minutes      0.0.0.0:8080->80/tcp web1
a4e52d75393e       ubuntu            "sleep infinity"    29 minutes ago
Up 29 minutes      _hardcore_shaw
[node1] (local) root@192.168.0.23 ~
$ docker swarm leave --force
Node left the swarm.

```