

Docker macvlan setup

The presentation configures Docker macvlan using the setup in "router01_build.pdf" and "router01_nonroot_docker-start_set-addr.pdf" documents.

Preuss
3/7/2020

```
albatross01@linux-s2020:~> /sbin/ip a sh
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    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
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:2c:05:ca brd ff:ff:ff:ff:ff:ff
    inet 192.168.117.168/24 brd 192.168.117.255 scope global noprefixroute dynamic eth0
        valid_lft 1490sec preferred_lft 1490sec
    inet6 fe80::7bd7:3627:c0be:957c/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:2c:05:d4 brd ff:ff:ff:ff:ff:ff
    inet 192.168.100.25/24 scope global eth1
        valid_lft forever preferred_lft forever
    inet6 2001:db8:abba:100::25/64 scope global
        valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe2c:5d4/64 scope link
        valid_lft forever preferred_lft forever
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:2c:05:de brd ff:ff:ff:ff:ff:ff
    inet 172.16.200.30/24 scope global eth2
        valid_lft forever preferred_lft forever
    inet6 2001:db8:abba:200::30/64 scope global
        valid_lft forever preferred_lft forever
    inet6 fe80::3de7:fe22:1bde:2f2f/64 scope link
        valid_lft forever preferred_lft forever
5: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:4d:ef:50:fd 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
albatross01@linux-s2020:~>
```

The presentation verifies the IP addresses are set correctly.

```
albatross01@linux-s2020:~> docker version
```

```
Client:
```

```
Version:           19.03.5
API version:       1.40
Go version:        go1.12.12
Git commit:        633a0ea838f1
Built:             Thu Dec 12 12:00:00 2019
OS/Arch:           linux/amd64
Experimental:     false
```

```
Server:
```

```
Engine:
```

```
Version:           19.03.5
API version:       1.40 (minimum version 1.12)
Go version:        go1.12.12
Git commit:        633a0ea838f1
Built:             Thu Dec 12 12:00:00 2019
OS/Arch:           linux/amd64
Experimental:     false
```

```
containerd:
```

```
Version:           v1.2.10
GitCommit:         b34a5c8af56e510852c35414db4c1f4fa6172339
```

```
runc:
```

```
Version:           1.0.0-rc8+dev
GitCommit:         3e425f80a8c931f88e6d94a8c831b9d5aa481657
```

```
docker-init:
```

```
Version:           0.1.3_catatonit
GitCommit:
```

```
albatross01@linux-s2020:~> █
```

The presentation verifies Docker is running.

```
albatross01@linux-s2020:~> 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.
```

```
albatross01@linux-s2020:~> █
```

The presentation shows the "docker network" options.

```
albatross01@linux-s2020:~> docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
a9ff53448eaa        bridge             bridge              local
e90590eac5bf        host               host                local
99e3796fd85f        none               null                local
albatross01@linux-s2020:~> █
```

The presentation shows the current docker networks.

```
albatross01@linux-s2020:~> docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
a9ff53448eaa        bridge             bridge              local
e90590eac5bf        host               host                local
99e3796fd85f        none              null                local
albatross01@linux-s2020:~> docker network create -d macvlan --subnet 192.168.100.0/24 --gateway=192.168.100.50 --subnet=2001:db8:abba:100::25/64 --gateway=2001:db8:abba:100::50 -o parent=eth1 internal01
783bb9fc00caa6dc8d53e9c3348ce5f634c9cbc10ee2aa17a20dedc70fccf0d1
albatross01@linux-s2020:~> █
```

The presentation creates the "internal01" macvlan.

Note: Docker requires additional configuration to support IPv6. These configurations are not shown in this presentation.

```
albatross01@linux-s2020:~> docker network create -d macvlan --subnet 172.16.200.0/24 --gateway=172.16.200.60 --subnet=2001:db8:abba:200::/64 --gateway=2001:db8:abba:200:
:60 -o parent=eth2 internal02
4bf624d8506ad542eba8933b8a5e642f8bace085a29bd26c4030aafbbba29cf4
albatross01@linux-s2020:~> █
```

The presentation creates the "internal02" macvlan.

Note: Docker requires additional configuration to support IPv6. These configurations are not shown in this presentation.

```
albatross01@linux-s2020:~> docker network create -d macvlan --subnet 172.16.200.0/24 --gateway=172.16.200.60 --subnet=2001:db8:abba:200::/64 --gateway=2001:db8:abba:200:
:60 -o parent=eth2 internal02
4bf624d8506ad542eba8933b8a5e642f8bace085a29bd26c4030aafbbba29cf4
albatross01@linux-s2020:~> docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
a9ff53448eaa        bridge              bridge              local
e90590eac5bf        host                host                local
783bb9fc00ca        internal01          macvlan             local
4bf624d8506a        internal02          macvlan             local
99e3796fd85f        none                null                local
albatross01@linux-s2020:~> █
```

The presentation shows the current docker networks.


```
albatross01@linux-s2020:~> docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
a9ff53448eaa        bridge             bridge              local
e90590eac5bf        host               host                local
783bb9fc00ca        internal01         macvlan             local
4bf624d8506a        internal02         macvlan             local
99e3796fd85f        none               null                local
albatross01@linux-s2020:~> docker network inspect internal01
[
  {
    "Name": "internal01",
    "Id": "783bb9fc00caa6dc8d53e9c3348ce5f634c9cbc10ee2aa17a20dedc70fccf0d1",
    "Created": "2020-03-07T11:41:26.567386207-06:00",
    "Scope": "local",
    "Driver": "macvlan",
    "EnableIPv6": false,
    "IPAM": {
      "Driver": "default",
      "Options": {},
      "Config": [
        {
          "Subnet": "192.168.100.0/24",
          "Gateway": "192.168.100.50"
        },
        {
          "Subnet": "2001:db8:abba:100::25/64",
          "Gateway": "2001:db8:abba:100::50"
        }
      ]
    },
    "Internal": false,
    "Attachable": false,
    "Ingress": false,
    "ConfigFrom": {
      "Network": ""
    },
    "ConfigOnly": false,
    "Containers": {},
    "Options": {
      "parent": "eth1"
    },
    "Labels": {}
  }
]
albatross01@linux-s2020:~>
```

The presentation shows the configuration of "internal01".