

Docker Network

- [Setting up docker network for Vlans](#)

Setting up docker network for Vlan

Setting Up a VLAN and Docker IPvlan Network

This guide demonstrates how to create a VLAN on a host interface and then use it to run a Docker container on a dedicated subnet using the IPvlan driver.

1. Create a VLAN Interface

```
sudo ip link add link eth0 name eth0.107 type vlan id 107
sudo ip link set eth0.107 up
```

Explanation:

- `ip link add link eth0 name eth0.107 type vlan id 107`
This creates a new VLAN interface (`eth0.107`) attached to the physical interface `eth0` with VLAN ID `107`.
- `ip link set eth0.107 up`
Activates the VLAN interface so it can start sending and receiving traffic.

2. Create a Docker IPvlan Network

```
docker network create -d ipvlan \  
--subnet 10.14.107.0/24 \  
--gateway 10.14.107.1 \  
-o parent=eth0.107 \  
test2
```

Explanation:

- `docker network create -d ipvlan`
Creates a Docker network using the **IPvlan** driver, which allows containers to appear as if they are directly connected to the physical network.
- `--subnet 10.14.107.0/24`
Defines the subnet for the Docker network.

- `--gateway 10.14.107.1`
Sets the gateway for the subnet.
 - `-o parent=eth0.107`
Binds the IPvlan network to the host VLAN interface (`eth0.107`). Replace `eth0.107` with your host interface if different.
 - `test2`
Name of the Docker network.
-

3. Run a Docker Container on the IPvlan Network

```
sudo docker run -itd --rm --network test2 --ip 10.14.107.50 --name test2 ubuntu
```

Explanation:

- `--network test2`
Connects the container to the previously created IPvlan network.
 - `--ip 10.14.107.50`
Assigns a static IP address to the container within the subnet.
 - `--name test2`
Gives the container a recognizable name.
 - `-itd --rm ubuntu`
Runs an Ubuntu container in detached mode, interactive terminal enabled, and removes it automatically when stopped.
-

Result:

You now have a Docker container (`test2`) running on its own VLAN (`eth0.107`) with a dedicated subnet. This setup is useful for network isolation, lab testing, or managing multiple services with their own IP addresses.
