

SDN-Vortex CNI Controller

- *Uses Openvswitch*
- *Init-container provisions secondary IP*

Reference:

<https://github.com/linkernetworks/network-controller>

Lab_1

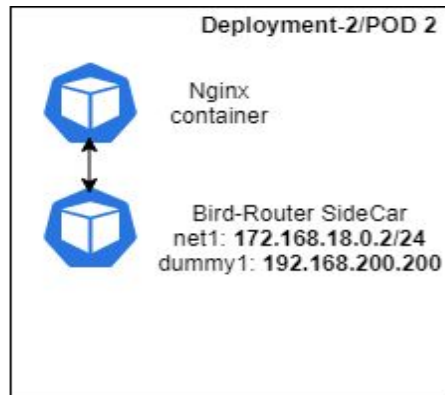
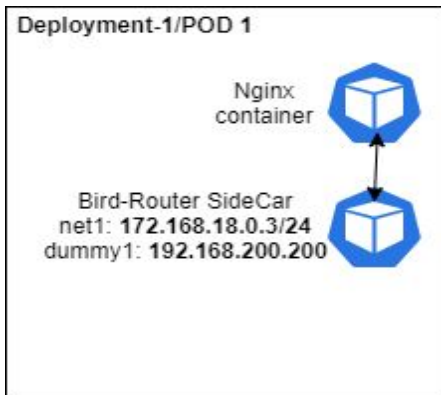
Setup openvswitch:

```
sudo apt-get install -y openvswitch-switch  
sudo ovs-vsctl add-br br0  
sudo ovs-vsctl add-port br0 enp0s9  
sudo ip link set dev br0 up
```

Install the daemonset:

```
kubectl create -f  
https://github.com/infinitydon/kubernetes-on-baremetal/raw/main/sdn-vortex/controller-daemonset.yaml
```

Install sample deployments to test, this will be a combination of an Nginx POD and Bird router sidecar. The bird router sidecar will form an OSPF neighborhood with the Vyos and advertise a dummy interface that will be used to access the Nginx from the Alpine container.



OSPF

OSPF



Vios Router
172.18.0.1/24
10.10.200.1/24

curl 192.168.200.200



docker
Alpine
10.10.200.2/24
GW: 10.10.200.1

Lab_1

Create the bird configmap:

```
kubect1 create -f  
https://github.com/infinitydon/kubernetes-on-baremetal/raw/main/sdn-vortex/bird-ospf-configmap.yaml
```

Create the 2 deployment to test:

```
kubect1 create -f  
https://github.com/infinitydon/kubernetes-on-baremetal/raw/main/sdn-vortex/nginx-bird-deployment-1.yaml
```

```
kubect1 create -f  
https://github.com/infinitydon/kubernetes-on-baremetal/raw/main/sdn-vortex/nginx-bird-deployment-2.yaml
```

Vyos OSPF config:

```
set protocols ospf area 0 network '172.18.0.0/24'
```

References

<https://github.com/linkernetworks/network-controller>

<https://github.com/infinitydon/kubernetes-on-baremetal/tree/main/sdn-vortex>