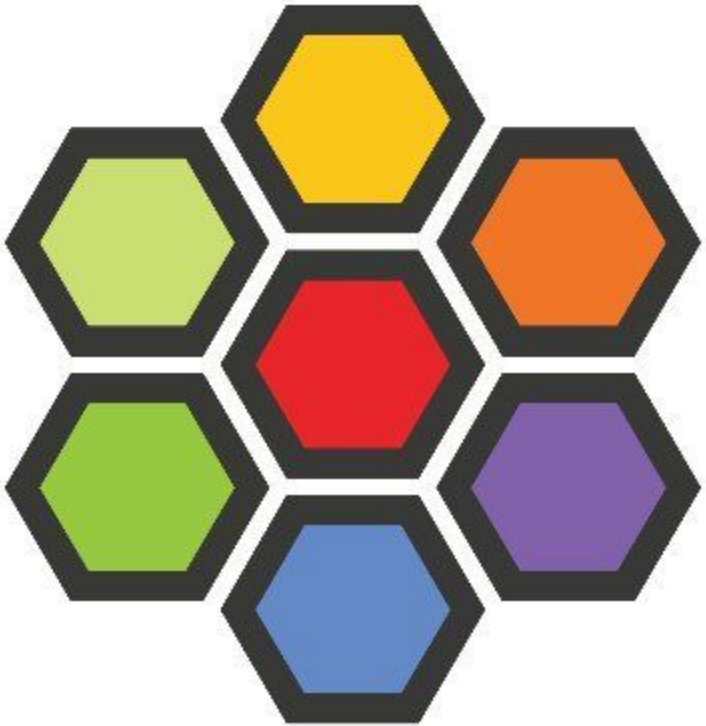
Cilium



Features

- *Uses eBPF instead of iptables*
- *Can replace kube-proxy entirely*
- *Multi-cluster connectivity (clusterMesh)*
- *Prometheus compatible Network metrics*
- *Network visibility via Hubble*
- *Advanced network policy*

Lab_1

Cleanup the kubeadm cluster and re-install again:

Run this on every node in the cluster: `sudo kubeadm reset;sudo rm -rf /etc/cni/net.d;rm -rf .kube/;sudo reboot`

Initialise kubeadm on the control-plane node: `sudo kubeadm init --apiserver-advertise-address=10.10.100.2 --pod-network-cidr=192.168.0.0/16`

Join the the 2 worker node to the control-plane: `sudo kubeadm join …..`

Install helm binary:

```
curl -fsSL -o get_helm.sh https://raw.githubusercontent.com/helm/helm/master/scripts/get-helm-3

chmod 700 get_helm.sh

./get_helm.sh
```

Lab_1

Install cilium CNI plus Hubble (visibility):

```
helm repo add cilium https://helm.cilium.io/

helm install cilium cilium/cilium --version 1.9.1 \
   --namespace kube-system \
   --set hubble.listenAddress=":4244" \
   --set hubble.relay.enabled=true \
   --set hubble.ui.enabled=true \
   --set ipam.operator.clusterPoolIPv4PodCIDR=192.168.0.0/16
```

Install some test resources to confirm everything is working:

```
kubectl create ns cilium-test

kubectl apply -n cilium-test -f
https://raw.githubusercontent.com/cilium/cilium/v1.9/examples/kubernetes/connectivity-check/connectivity-check.yaml
```

Lab_1

Check status of the pods that were created

```
kubectl get pods -n cilium-test
```

Expose hubble via port-forwarding:

```
kubectl port-forward -n kube-system svc/hubble-ui --address 0.0.0.0 12000:80
```

From the GNS3 topology it is better to access the hubble by using VirtualBox NAT port-forwarding also, add a new rule that uses the Linux host IP and port 12000. With this you will be to access hubble over the LAN i.e. in my windows PC (for my own setup).

Lab_2

Advertise POD cidr via BGP

Dockerfile for bird routing software: https://github.com/pierky/dockerfiles/raw/master/bird/2.0.7/Dockerfile

Build and push image:

```
docker build -t registry.gitlab.com/infinitydon/registry/bird:v2.0.7
.;docker push registry.gitlab.com/infinitydon/registry/bird:v2.0.7
```

We will start by resetting the cluster and install the helm chart with following parameters:

```
helm install cilium cilium/cilium --version 1.9.2 \
   --namespace kube-system \
   --set hubble.listenAddress=":4244" \
   --set hubble.relay.enabled=true \
   --set hubble.ui.enabled=true \
   --set ipam.operator.clusterPoolIPv4PodCIDR=192.168.0.0/16 \
   --set tunnel=disabled \
   --set nativeRoutingCIDR=10.10.0.0/16 \
   --set autoDirectNodeRoutes=true
```

`tunnel=disabled -` Disable overlay mode, this is needed if POD IPs will be advertised.

`nativeRoutingCIDR=10.10.0.0/16 -` This is the LAN IP range, so traffic related to this range will not be NATTED, since in our case we want POD IPs to communicate directly in LAN, other traffic like those destined to the internet will be NATTED (masquerade) by cilium.

`autoDirectNodeRoutes=true -` Cilium will share node POD IP ranges among all the kubernetes nodes, this will also enable coreDNS pod to connect to the kubernetes API service IP after Cilium is installed, otherwise the coreDNS pods will not start up properly, if this option is false then the routes must be configured manually.

Lab_2

Install the Bird configmap and daemonset, this will form BGP neighborship with the Vyos router.

```
kubectl create -f
https://github.com/infinitydon/kubernetes-on-baremetal/raw/main/cilium/
bird-configmap.yaml
```

```
kubectl create -f
https://github.com/infinitydon/kubernetes-on-baremetal/raw/main/cilium/
bird-daemonset.yaml
```

Vyos configuration:

```
set protocols bgp 64500 neighbor 10.10.100.3 remote-as '64500'
set protocols bgp 64500 neighbor 10.10.100.4 remote-as '64500'
```

Create sample deployment and check if POD is accessible from the Alpine container.

References

https://docs.cilium.io/en/v1.9/gettingstarted/k8s-install-kubeadm/

https://github.com/cilium/cilium/blob/master/install/kubernetes/cilium/values.yaml

https://docs.cilium.io/en/v1.9/gettingstarted/kubeproxy-free/#kubeproxy-free

https://docs.cilium.io/en/v1.9/concepts/observability/#concepts-observability

https://docs.cilium.io/en/v1.9/gettingstarted/grafana/

https://cilium.io/blog/2019/03/12/clustermesh

https://cilium.io/blog/2020/06/22/cilium-18#hostfw

https://kubernetes.io/docs/reference/setup-tools/kubeadm/kubeadm-reset/