Kubernetes networking with Calico

Hemanth Nakkina, Solution Architect, Ericsson

Abhijeet Singh, Director, AT&T

Uday T Kumar, Solution Architect, Ericsson

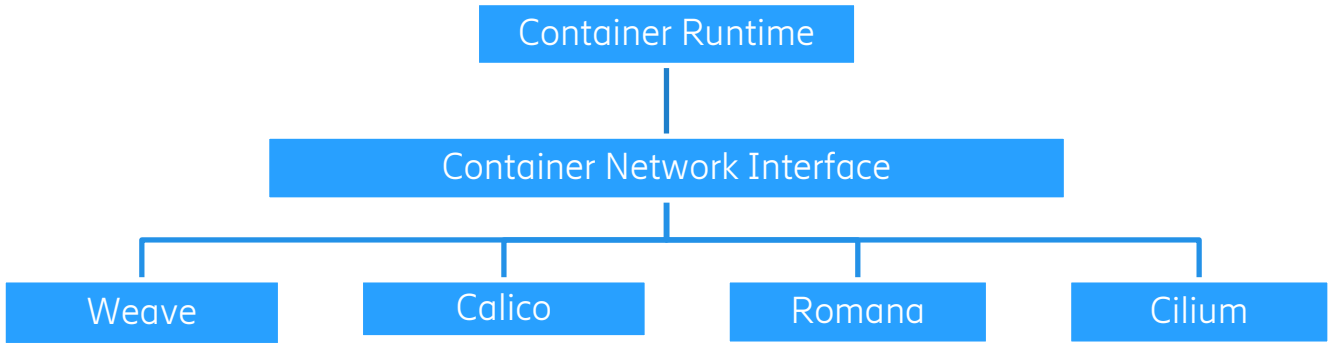
“ There is no such thing as Container Networking ”

— Kelsey Hightower, Google Dev Evangelist.

Title of his talk. Source: devopsnetworkingforum2016.sched.com

Networking for Containers

— CNI (Container Network Interface): Specification that act as interface between Container runtime and networking model implementations



Basic Network requirements

— IPAM and lifecycle management of network devices
— Connectivity in Container network
— Route advertisement

Sample CNI configuration

```
{
 "name": "k8s-pod-network",
 "cniVersion": "0.3.0",
 "plugins": [
  {
   "type": "calico",
   "etcd_endpoints": "http://10.96.232.136:6666",
   "log_level": "info",
   "mtu": 1500,
   "ipam": {
       "type": "calico-ipam"
   },
   "policy": {
       "type": "k8s",
       "k8s_api_root": "https://10.96.0.1:443",
       "k8s_auth_token": "<auth token>"
   },
   "kubernetes": {
       "kubeconfig": "/etc/cni/net.d/calico-kubeconfig"
   }
  },
  {
   "type": "portmap",
   "snat": true,
   "capabilities": {"portMappings": true}
  }
 ]
}
```

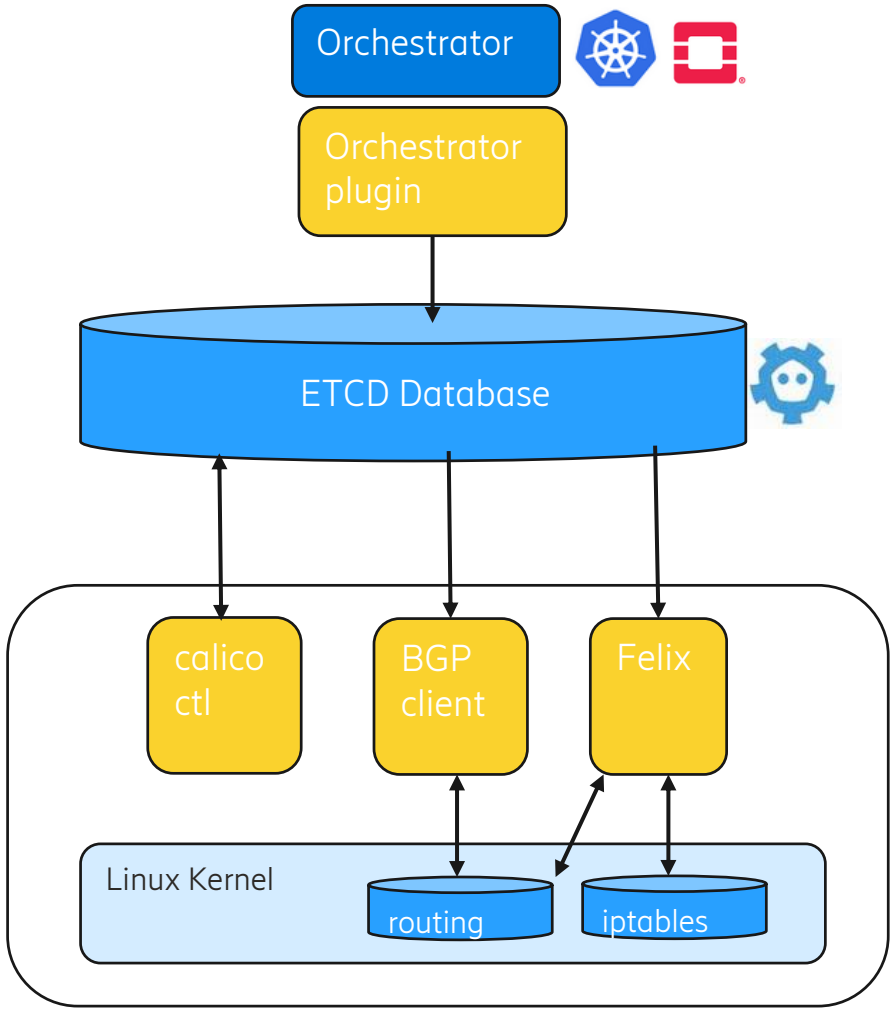
Calico Architecture

Designed to simplify, scale and secure cloud networks by

- Layer 3 based routing approach
- BGP for Routes distribution
- Policy driven network security implemented by iptable rules

Components

- Felix
- Orchestrator plugin
- Etcd
- BGP Client
- BGP Route reflector



Calico – Deployment on k8s

Helm chart - https://github.com/openstack/openstack-helm-infra/tree/master/calico

```
openstack@k8sm1:~/logs$ sudo kubectl get pods --all-namespaces -o wide | grep calico
kube-system   calico-etcd-k4bxk                          1/1       Running   4          9d        10.0.2.6         k8sm1
kube-system   calico-kube-controllers-5d74847676-hjcg2   1/1       Running   7          9d        10.0.2.6         k8sm1
kube-system   calico-node-kp6t6                          2/2       Running   1          2h        10.0.2.7         k8sn1
kube-system   calico-node-r2dfv                          2/2       Running   12         9d        10.0.2.6         k8sm1
```

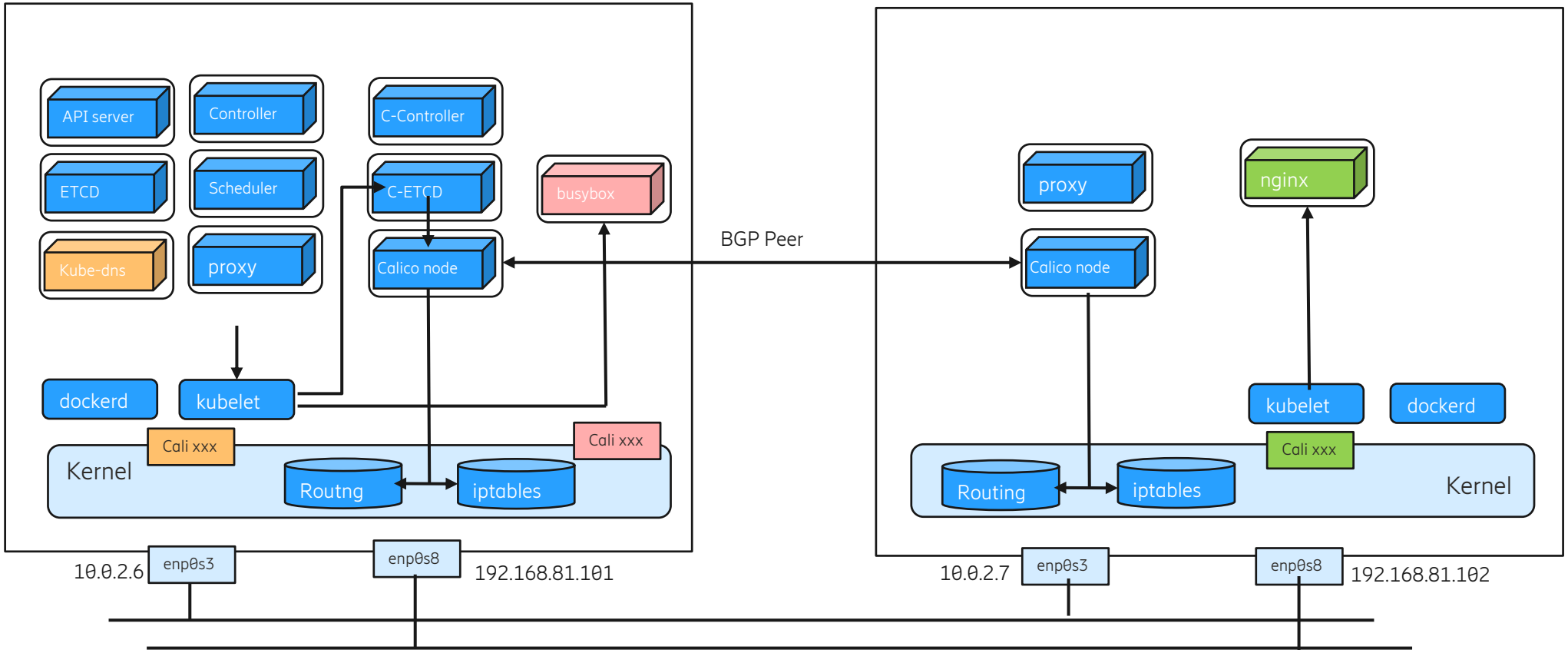
Configuration updates

```
podSubnet: 192.168.0.0/16
# NOTE(portdirect): this should be the physical MTU, the appropriate MTU
# that calico should use will be calculated.
mtu: 1500
settings:
  mesh: "on"
```

```
bgp:
  # our asnumber for bgp peering
  asnumber: 64512
  ipv4:
```

```
    # Detection of source interface for routing
    # options include
    # can-reach=DESTINATION
    # interface=INTERFACE-REGEX
    IP_AUTODETECTION_METHOD: first-found
    IPV6_AUTODETECTION_METHOD: first-found
```

Calico – How it works



default via 10.0.2.1 dev enp0s3
10.0.2.0/24 dev enp0s3 proto kernel scope link src 10.0.2.6
192.168.81.0/24 dev enp0s8 proto kernel scope link src 192.168.81.101

blackhole 192.200.59.192/26 proto bird
192.200.59.193 dev calidf072d3c423 scope link
192.200.59.198 dev cali0aa3720a2c7 scope link
192.200.203.0/26 via 192.168.81.102 dev tunl0 proto bird onlink

default via 10.0.2.1 dev enp0s3
10.0.2.0/24 dev enp0s3 proto kernel scope link src 10.0.2.7
192.168.81.0/24 dev enp0s8 proto kernel scope link src 192.168.81.102

192.200.59.192/26 via 192.168.81.101 dev tunl0 proto bird onlink
blackhole 192.200.203.0/26 proto bird
192.200.203.4 dev cali7bb4560a7c2 scope link

Iptable rules related to services

NAT to resolve Service IP to Pod IP

```
Chain KUBE-SERVICES (2 references)
 pkts bytes target     prot opt in     out     source               destination
    0     0 KUBE-MARK-MASQ  tcp  --  *      *      !192.168.0.0/16       10.96.232.136        /* kube-system/calico-etcd: cluster IP */ tcp dpt:6666
    0     0 KUBE-SVC-NTYB37XIWATNM25Y  tcp  --  *      *       0.0.0.0/0            10.96.232.136        /* kube-system/calico-etcd: cluster IP */ tcp dpt:6666
    0     0 KUBE-MARK-MASQ  udp  --  *      *      !192.168.0.0/16       10.96.0.10           /* kube-system/kube-dns:dns cluster IP */ udp dpt:53
    0     0 KUBE-SVC-TCOU7JCQXEZGVUNU  udp  --  *      *       0.0.0.0/0            10.96.0.10           /* kube-system/kube-dns:dns cluster IP */ udp dpt:53
    0     0 KUBE-MARK-MASQ  tcp  --  *      *      !192.168.0.0/16       10.96.0.10           /* kube-system/kube-dns:dns-tcp cluster IP */ tcp dpt:53
    0     0 KUBE-SVC-ERIFXISQEP7F7OF4  tcp  --  *      *       0.0.0.0/0            10.96.0.10           /* kube-system/kube-dns:dns-tcp cluster IP */ tcp dpt:53
    0     0 KUBE-MARK-MASQ  tcp  --  *      *      !192.168.0.0/16       10.96.0.1            /* default/kubernetes:https cluster IP */ tcp dpt:443
    0     0 KUBE-SVC-NPX46M4PTMTKRN6Y  tcp  --  *      *       0.0.0.0/0            10.96.0.1            /* default/kubernetes:https cluster IP */ tcp dpt:443
    0     0 KUBE-MARK-MASQ  tcp  --  *      *      !192.168.0.0/16       10.102.249.96        /* default/nginx: cluster IP */ tcp dpt:80
    0     0 KUBE-SVC-4N57TFCL4MD7ZTDA  tcp  --  *      *       0.0.0.0/0            10.102.249.96        /* default/nginx: cluster IP */ tcp dpt:80
    8   480 KUBE-NODEPORTS  all  --  *      *       0.0.0.0/0            0.0.0.0/0            /* kubernetes service nodeports; NOTE: this must be the last rule
   in this chain */ ADDRTYPE match dst-type LOCAL

Chain KUBE-SVC-4N57TFCL4MD7ZTDA (2 references)
 pkts bytes target     prot opt in     out     source               destination
    0     0 KUBE-SEP-F2W2OWMV4YNNQT44  all  --  *      *       0.0.0.0/0            0.0.0.0/0            /* default/nginx: */ statistic mode random probability
    0.33332999982
    0     0 KUBE-SEP-WEWK3DETZSAOVCUI  all  --  *      *       0.0.0.0/0            0.0.0.0/0            /* default/nginx: */ statistic mode random probability
    0.50000000000
    0     0 KUBE-SEP-F6H4BEWBSORHU2YI  all  --  *      *       0.0.0.0/0            0.0.0.0/0            /* default/nginx: */

Chain KUBE-SEP-F2W2OWMV4YNNQT44 (1 references)
 pkts bytes target     prot opt in     out     source               destination
    0     0 KUBE-MARK-MASQ  all  --  *      *       192.168.104.3        0.0.0.0/0            /* default/nginx: */
    0     0 DNAT       tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           /* default/nginx: */ tcp to:192.168.104.3:80

Chain KUBE-SEP-F6H4BEWBSORHU2YI (1 references)
 pkts bytes target     prot opt in     out     source               destination
    0     0 KUBE-MARK-MASQ  all  --  *      *       192.168.219.68       0.0.0.0/0            /* default/nginx: */
    0     0 DNAT       tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           /* default/nginx: */ tcp to:192.168.219.68:80

Chain KUBE-SEP-WEWK3DETZSAOVCUI (1 references)
 pkts bytes target     prot opt in     out     source               destination
    0     0 KUBE-MARK-MASQ  all  --  *      *       192.168.166.131      0.0.0.0/0            /* default/nginx: */
    0     0 DNAT       tcp  --  *      *       0.0.0.0/0            0.0.0.0/0           /* default/nginx: */ tcp to:192.168.166.131:80
```

Thanks ! Merci !