Calico network in high scale

From 5 nodes to 5000

Richa’rd Kova’cs

1. Agenda

➔ **What is Calico ?**
A few words about the context.

➔ **Default installation**
What is under the hood.

➔ **Unstress Kube API server**
Save resources by decreasing calls.

➔ **Problem statement**
Limitations of the default config.

➔ **Solutions**
Scale BGP messages and connections.

1. Agenda

Me, myself and I

Kubernetes Network Engineer

@ IBM Cloud

Many years DevOps background

OSS activist

Knowns as mhmxs

Scale BGP messages and connections.

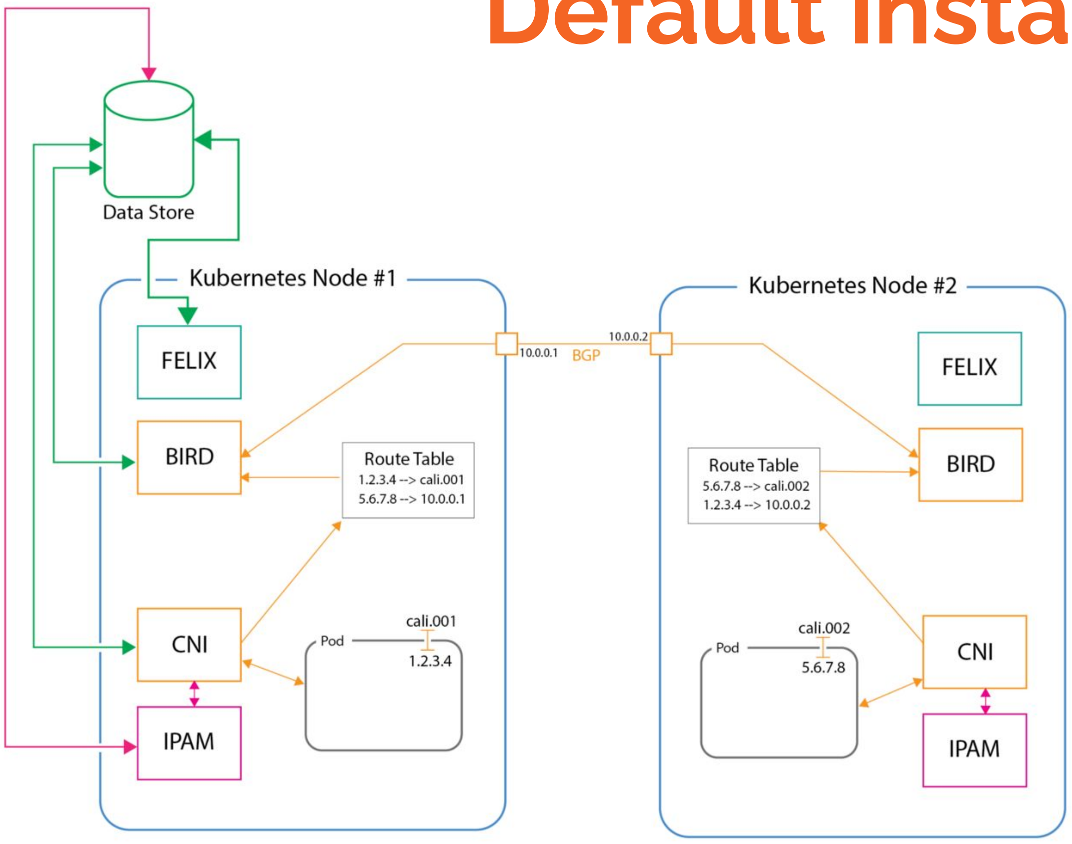
What is Calico ?

is an open source networking & network security solution for containers, VMs, and bare-metal workloads.

is a CNI plugin, providing compatibility layer with Kubernetes to implement overlay network

is used by the big players in public cloud space and has big community under CNCF’s umbrella

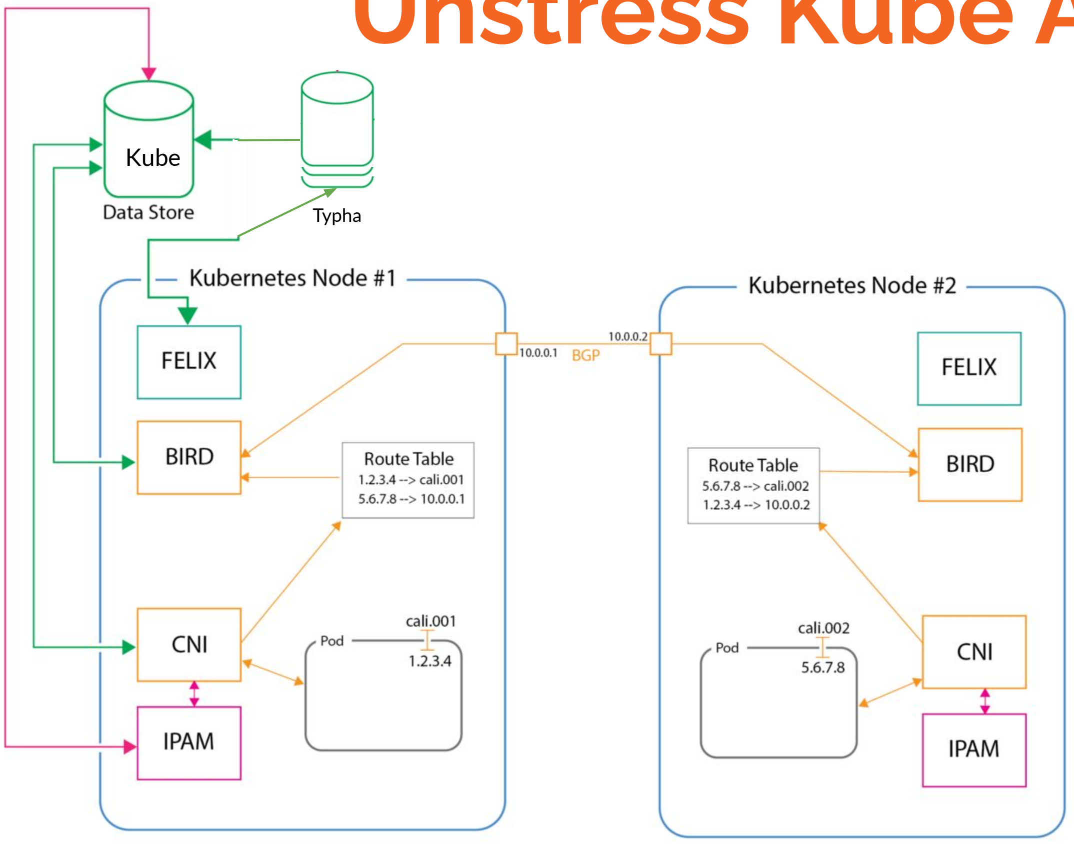
Default installation



- Datastore [ETCD || Kubernetes]
- Felix [network policy enforcement]
- Bird [BGP implementation]
- CNI [CNI implementation]
- IPAM [IP address mgmt]

src: https://medium.com/@k.grundy/project-calico-kubernetes-integration-overview-a3a860cd974e

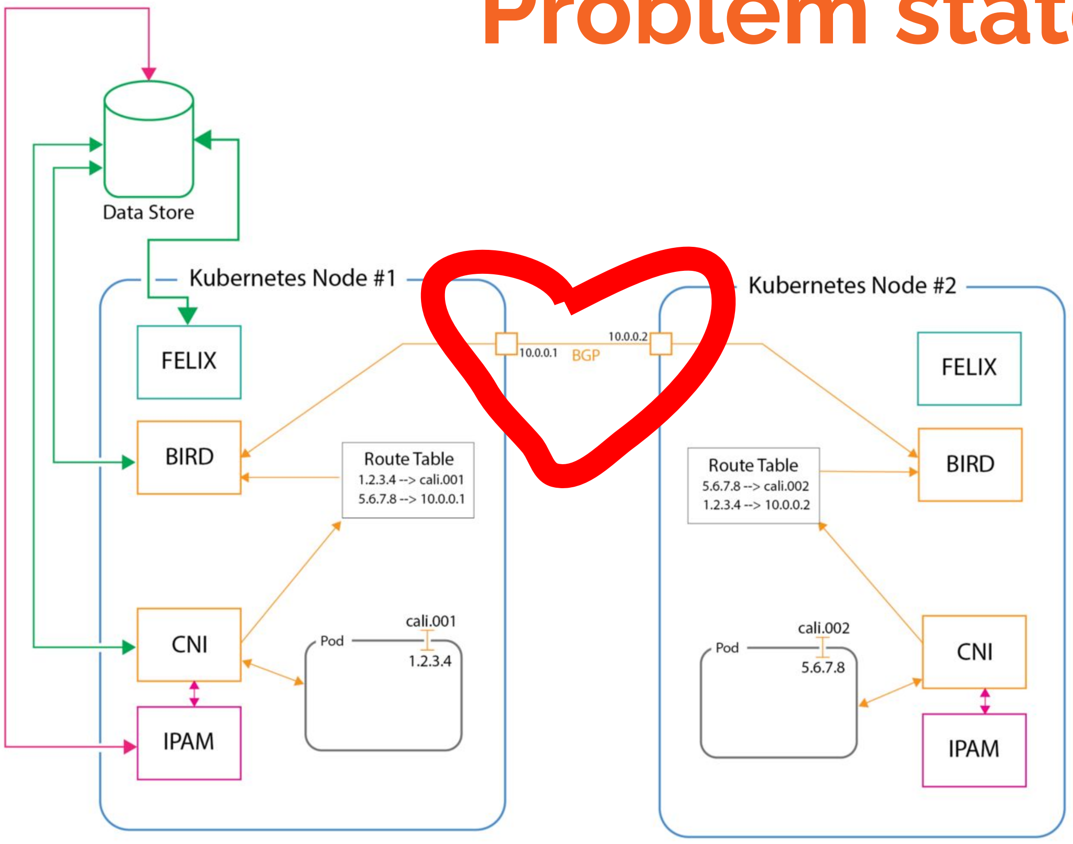
Unstress Kube API server



- The Typha daemon sits between the datastore and Felix

src: https://medium.com/@k.grundy/project-calico-kubernetes-integration-overview-a3a860cd974e

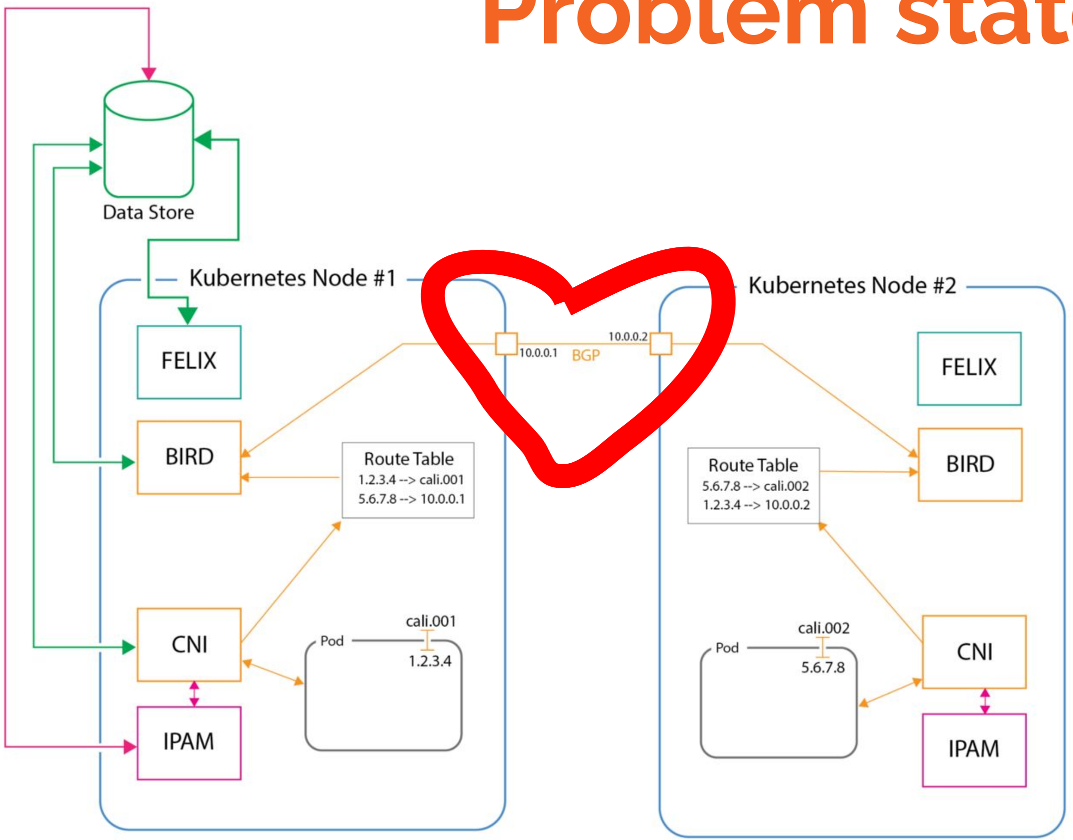
Problem statement



- BGP would be the bottleneck
- Default IPIP encapsulation
- All traffic encapsulated by default
- NAT ongoing
- Node and peer selectors are all()

src: https://medium.com/@k.grundy/project-calico-kubernetes-integration-overview-a3a860cd974e

Problem statement



- BGP would be the bottleneck
  - VXLAN only doesn't need
- Default IPIP encapsulation
  - VXLAN poor throughput
- All traffic encapsulated by default
  - Cross subnet
- NAT ongoing
  - Special cases, advanced
- Node and peer selectors are all()
  - Route Reflectors

src: https://medium.com/@k.grundy/project-calico-kubernetes-integration-overview-a3a860cd974e

—

Single cluster

```
     _______________
    /               \
   RR1-------RR2-------RR3
   |/ \ / \ /  \ / \ / \|
   |\ / \ / \  / \ / \ /|
Client1    Client2    Client3
```

Numbers

# of nodes	500
# of RRs	3
Redundancy	3
# of clients / RR	497
# of RRs / RR	2
Connections / RR	499

- The simplest route reflector topology contains only one cluster ID
- There are only one group of route reflectors and one group for clients
- This topology doesn't scale well and useful only for single zone or single region clusters

Single cluster

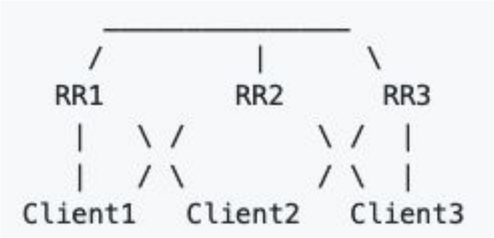
```
      ____________________
     /                    \
    RR1--------RR2--------RR3
    |/ \ / \ /  \ / \ / \|
    |\ / \ / \  / \ / \ /|
Client1     Client2     Client3
```

- The number of client connections per route reflector could be a bottleneck very easy

Numbers

# of nodes	500
# of RRs	3
Redundancy	3
# of clients / RR	497
# of RRs / RR	2
Connections / RR	499

Multi cluster



- Each Route Reflector has its own cluster ID
- Route Reflectors are constituting one mesh
- Clients are connecting to 3 different clusters
- Full table advertising

Numbers

# of nodes	2000
# of RRs	11
Redundancy	3
# of clients / RR	542
# of RRs / RR	10
Connections / RR	~552

Multi cluster

```
    _____________________
   /          |          \
RR1          RR2          RR3
 |   \ /           \ /    |
 |   / \           / \    |
Client1    Client2    Client3
```

- BGP update message size and number could be a bottleneck

Numbers

# of nodes	2000
# of RRs	11
Redundancy	3
# of clients / RR	542
# of RRs / RR	10
Connections / RR	~552

```
     _________________
    /            _______\
 RR1_______  RR2       RR3
  |   \ /           \ /  |
  |   / \           / \  |
Client1   Client2   Client3
```

Quorum cluster

- Route Reflectors can be divided into clusters based on cluster ID
- Route Reflectors do not share routes that they learned from remote clients
- Clients are connecting to at least 2 different quorum

Numbers

# of nodes	2000
# of RRs	13
Redundancy	3
# of Quorums	286
# of clients per quorum	7
# of quorum / RR	63
# of clients / RR	456
# of RRs / RR	12
Connections / RR	~468

―

Quorum cluster

```
       _________________
      /                 ___\
   RR1_______  RR2        RR3
    |   \ /          \ /   |
    |   / \          / \   |
 Client1   Client2    Client3
```

- ~3000 (depends on the flavor) nodes the BGP connection number became bottleneck
- Increasing the number of Route Reflectors can decrease the number of connections but increases the size of the BGP update messages

Numbers

# of nodes	2000
# of RRs	13
Redundancy	3
# of Quorums	286
# of clients per quorum	7
# of quorum / RR	63
# of clients / RR	456
# of RRs / RR	12
Connections / RR	~468

```
    ________
   /        \
  S1---S2---S3
  X          X
R1  R2  R3    R4  R5  R6
|  X  X  |    |  X  X  |
C1  C2  Cn   C3  C4  Cm
```

Hierarchy cluster

- Mimic the structure of a datacenter network
- Dividing the cluster into "racks"
- There's no need for a direct session between R1<>R2 and R3<>R4 as they'll receive each other's routes via a spine RR

Numbers

# of nodes	5000
# of racks	10
# of RRs	33
Redundancy	3
# of clients / RR	447
# of RRs / RR	3
Connections / RR	~500

Thank you!

Any question?

Links!

Auto scaler operator is on the way

Proposal doc:
https://github.com/mhmxs/calico-route-reflector-operator-proposal
POC:
https://github.com/mhmxs/calico-route-reflector-operator

https://www.linkedin.com/in/mhmxs/
https://twitter.com/mhmxs
https://github.com/mhmxs