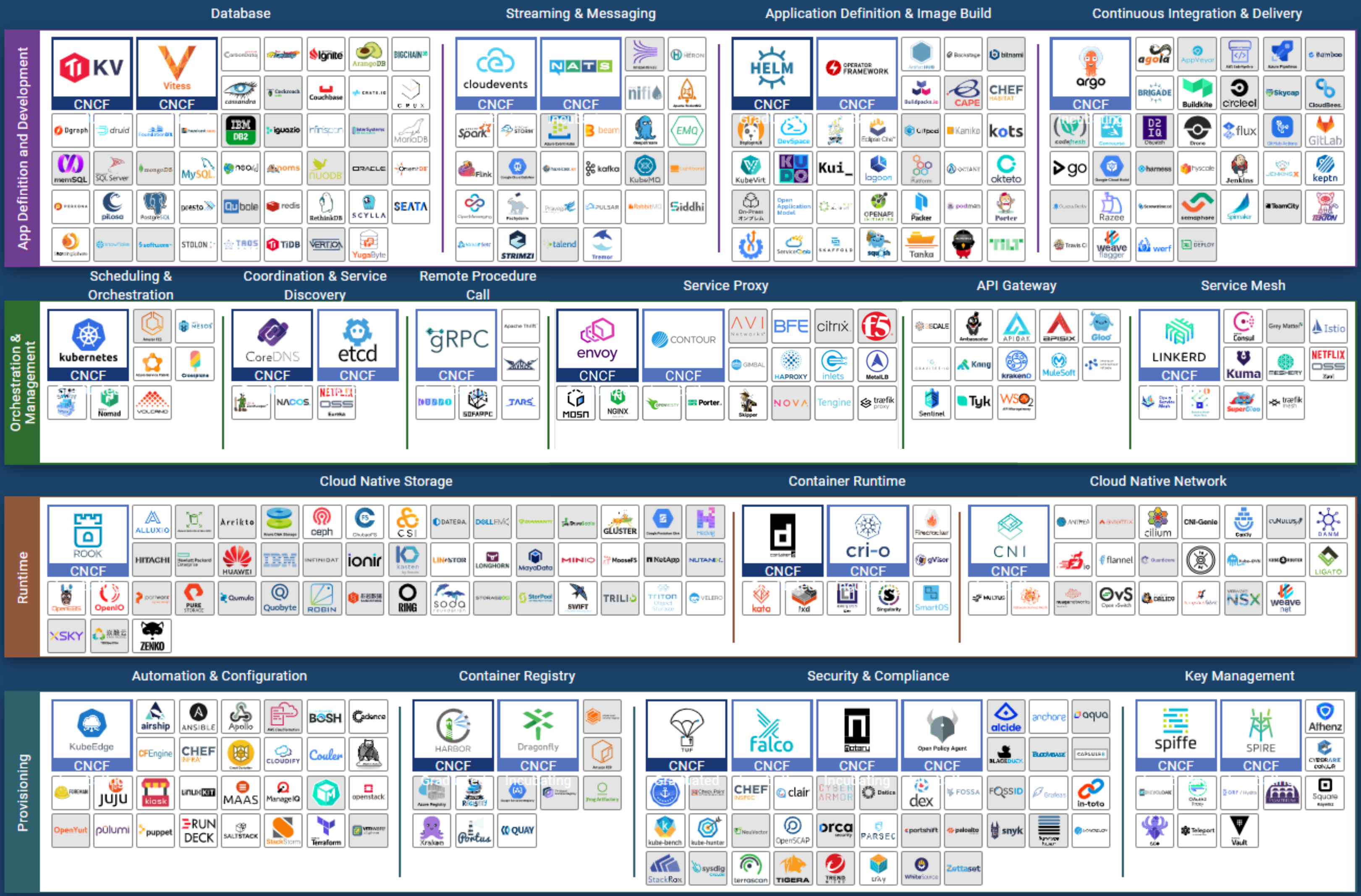
CNCF 온프렘 로드밸런서
Porter와 MetalLB
지극히 개인적인 비교

조훈 (Hoon Jo)

- CCIE DC, CKA&D, VCIX-NV6, RHCE, GCP-PCA&DE

https://github.com/SysNet4Admin

https://app.vagrantup.com/SysNet4Admin



App Definition and Development

Database

Streaming & Messaging

Application Definition & Image Build

Continuous Integration & Delivery

Orchestration & Management

Scheduling & Orchestration

Coordination & Service Discovery

Remote Procedure Call

Service Proxy

API Gateway

Service Mesh

Runtime

Cloud Native Storage

Container Runtime

Cloud Native Network

Provisioning

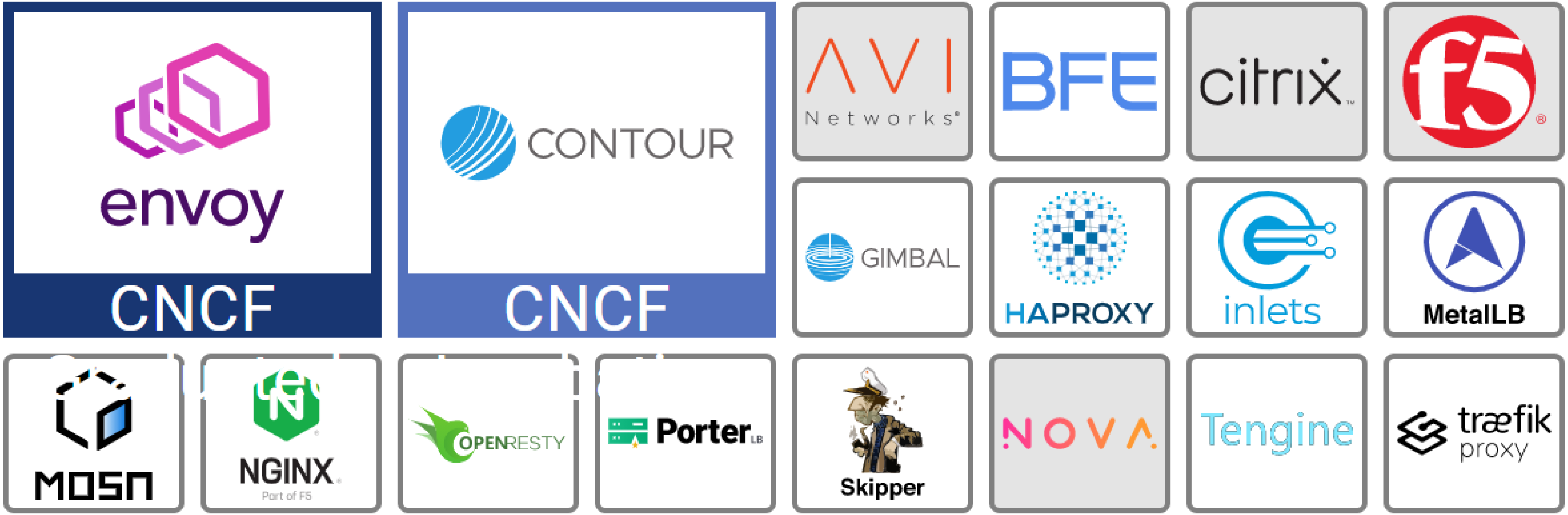
Automation & Configuration

Container Registry

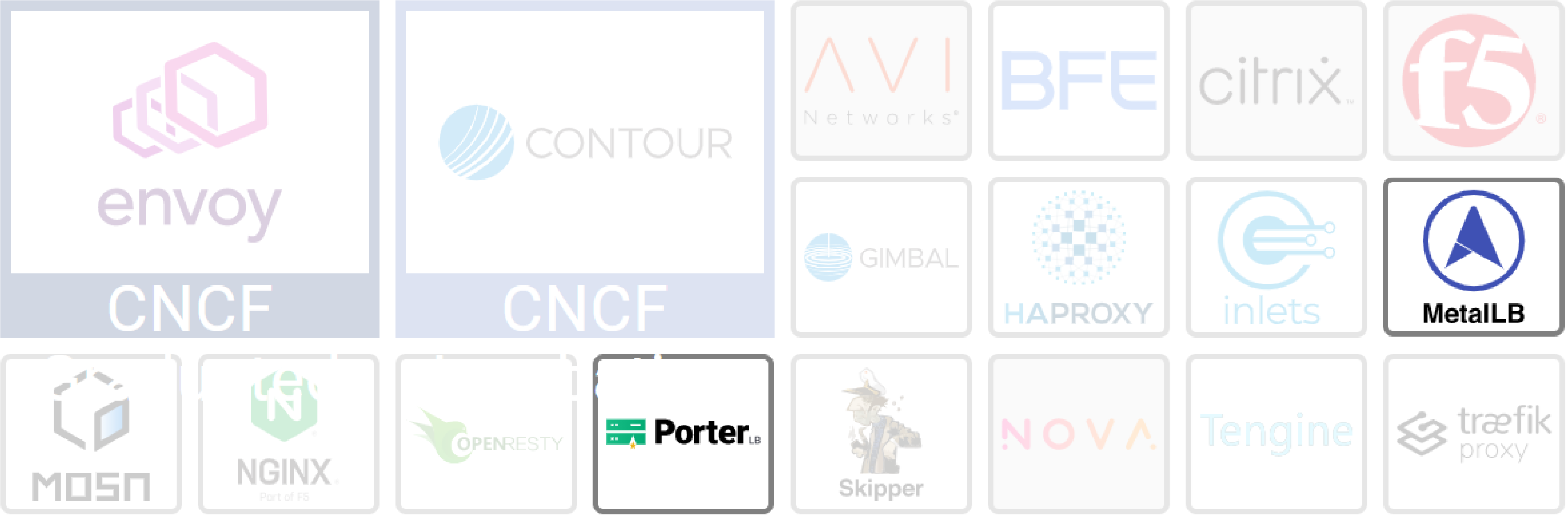
Security & Compliance

Key Management

Service Proxy



Service Proxy



Hello!

[클라우드☁솔루션_아키택처]@Magazone

CCIE DC, CKA&D, VCIX-NV6, RHCE, GCP-PCA&DE

Linkedin(🔗): https://www.linkedin.com/in/hoonjo/

시스템/네트워크 IT 벤더의 경험 이후 Megazone GCP 클라우드 팀으로 자리를 옮겨서 클라우드 기술 스택을 쌓고 있다. 근 시일 내에 쿠버네티스가 모든 인프라의 기반 기술이 될 것이라고 믿고 있으며 이에 발맞추어 쿠버네티스관련 책을 집필하고 있다.

인프런/유데미에서 인프라 자동화를 위한 앤서블에 대한 강의를 진행하고 있으며, 가지고 있는 지식을 나누는 것을 즐겨 인프런에

'CKA(공인 쿠버네티스 관리자) 잘 준비하는 법'에 대한 강의를 기재하였다. 또한 페이스북에 있는 'IT 인프라 엔지니어 그룹'과 '앤서블 유저그룹'의 운영진을 맡고 있다.

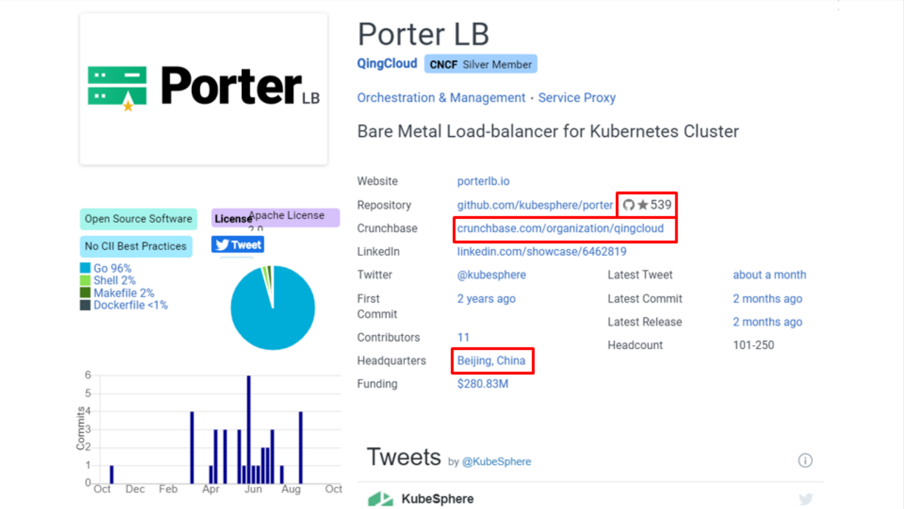
집필한 책으로는『시스템/네트워크 관리자를 위한 파이썬 실무 프로그래밍』(위키북스)』과『우아하게 앤서블』(비제이퍼블릭) 이 있고 IT잡지에 기고문을 쓰는 것을 즐긴다.

Porter LB

VS.

MetalLB

1.Porter와 MetalLB 비기술영역 비교



Porter LB

QingCloud **CNCF** Silver Member

Orchestration & Management · Service Proxy

Bare Metal Load-balancer for Kubernetes Cluster

Website	porterlb.io		
Repository	github.com/kubesphere/porter ★539		
Crunchbase	crunchbase.com/organization/qingcloud		
LinkedIn	linkedin.com/showcase/6462819		
Twitter	@kubesphere	Latest Tweet	about a month
First Commit	2 years ago	Latest Commit	2 months ago
		Latest Release	2 months ago
Contributors	11	Headcount	101-250
Headquarters	Beijing, China		
Funding	$280.83M		

Open Source Software

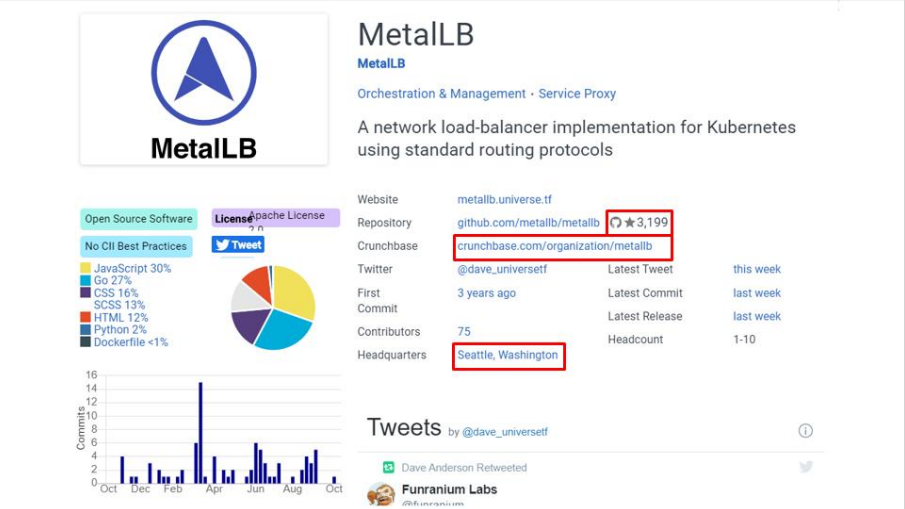
License Apache License 2.0

No CII Best Practices

Tweet

Tweets by @KubeSphere

KubeSphere



MetalLB

MetalLB

Orchestration & Management · Service Proxy

A network load-balancer implementation for Kubernetes using standard routing protocols

Website	metallb.universe.tf		
Repository	github.com/metallb/metallb ★3,199		
Crunchbase	crunchbase.com/organization/metallb		
Twitter	@dave_universetf	Latest Tweet	this week
First Commit	3 years ago	Latest Commit	last week
		Latest Release	last week
Contributors	75	Headcount	1-10
Headquarters	Seattle, Washington		

Open Source Software

License Apache License 2.0

No CII Best Practices

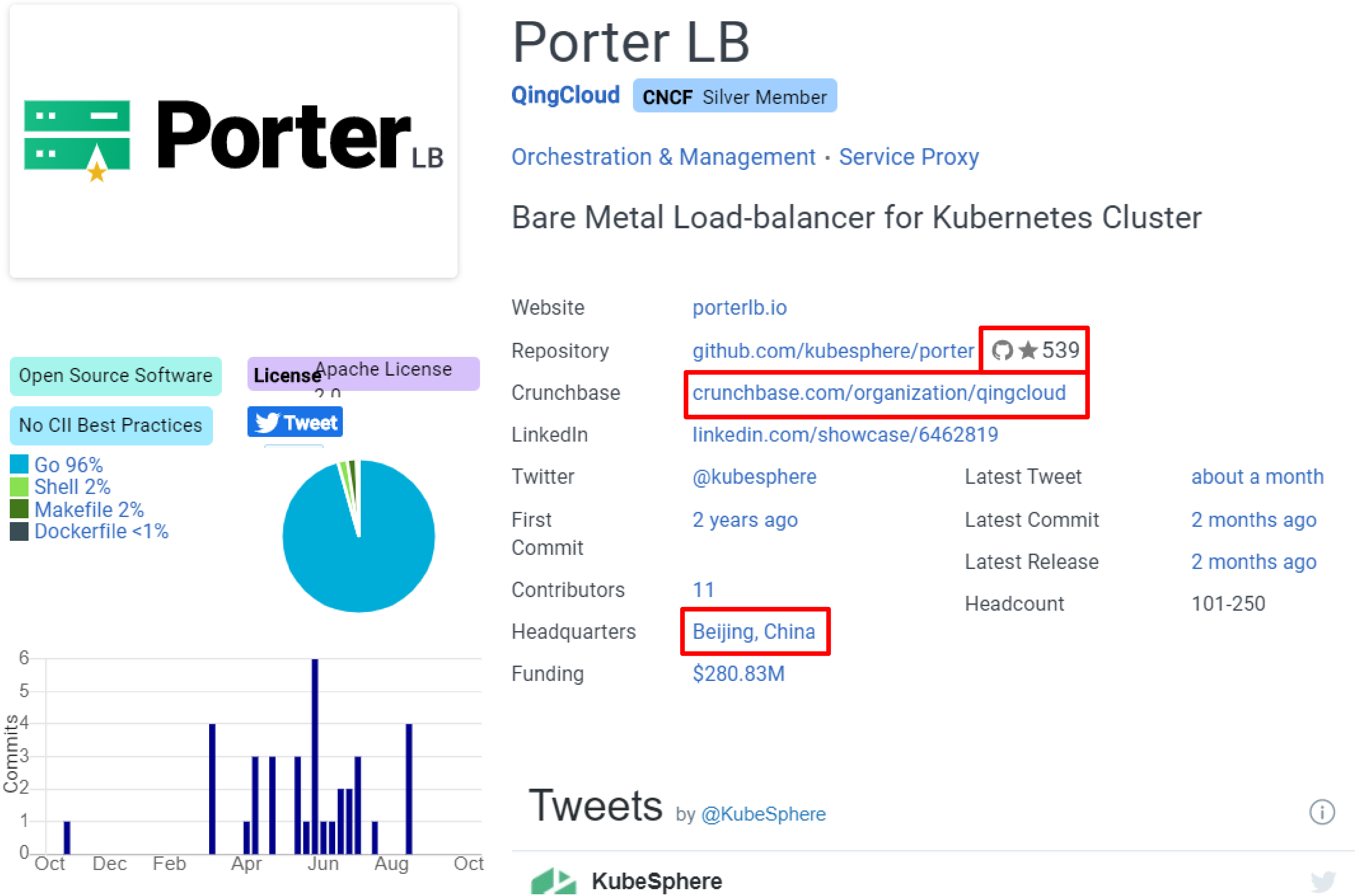
Tweet

Tweets by @dave_universetf

Dave Anderson Retweeted

Funranium Labs

@funranium



Porter LB

Porter LB

QingCloud **CNCF** Silver Member

Orchestration & Management · Service Proxy

Bare Metal Load-balancer for Kubernetes Cluster

Website	porterlb.io		
Repository	github.com/kubesphere/porter ★ 539		
Crunchbase	crunchbase.com/organization/qingcloud		
LinkedIn	linkedin.com/showcase/6462819		
Twitter	@kubesphere	Latest Tweet	about a month
First Commit	2 years ago	Latest Commit	2 months ago
		Latest Release	2 months ago
Contributors	11	Headcount	101-250
Headquarters	Beijing, China		
Funding	$280.83M		

Open Source Software

License Apache License 2.0

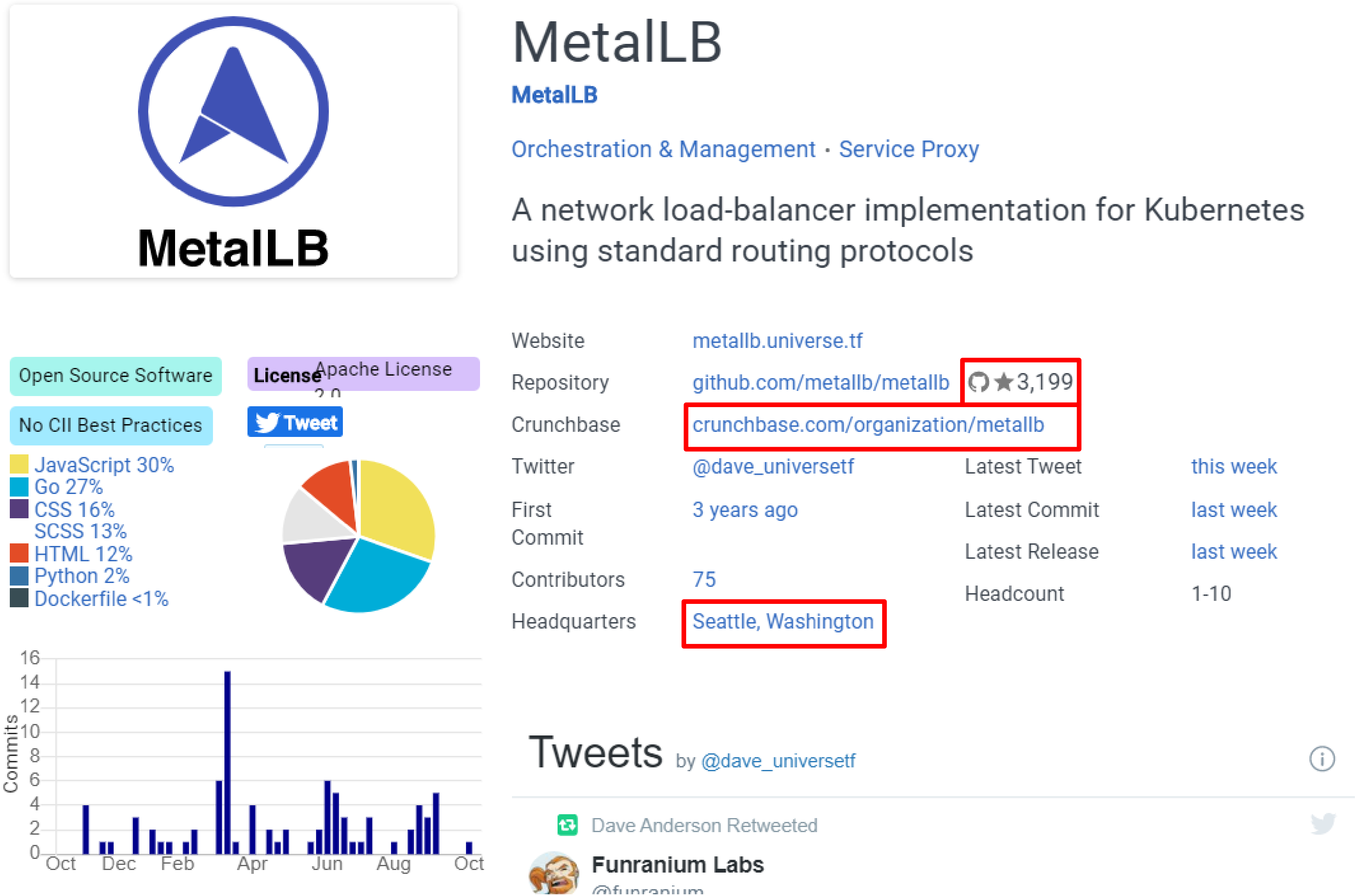
No CII Best Practices

Tweet

🟦	Go 96%
🟩	Shell 2%
🟩	Makefile 2%
■	Dockerfile <1%

Tweets by @KubeSphere

KubeSphere



MetalLB

MetalLB

Orchestration & Management · Service Proxy

A network load-balancer implementation for Kubernetes using standard routing protocols

Open Source Software

License Apache License 2.0

No CII Best Practices

Tweet

🟨	JavaScript 30%
🟦	Go 27%
🟪	CSS 16%
	SCSS 13%
🟥	HTML 12%
🟦	Python 2%
⬛	Dockerfile <1%

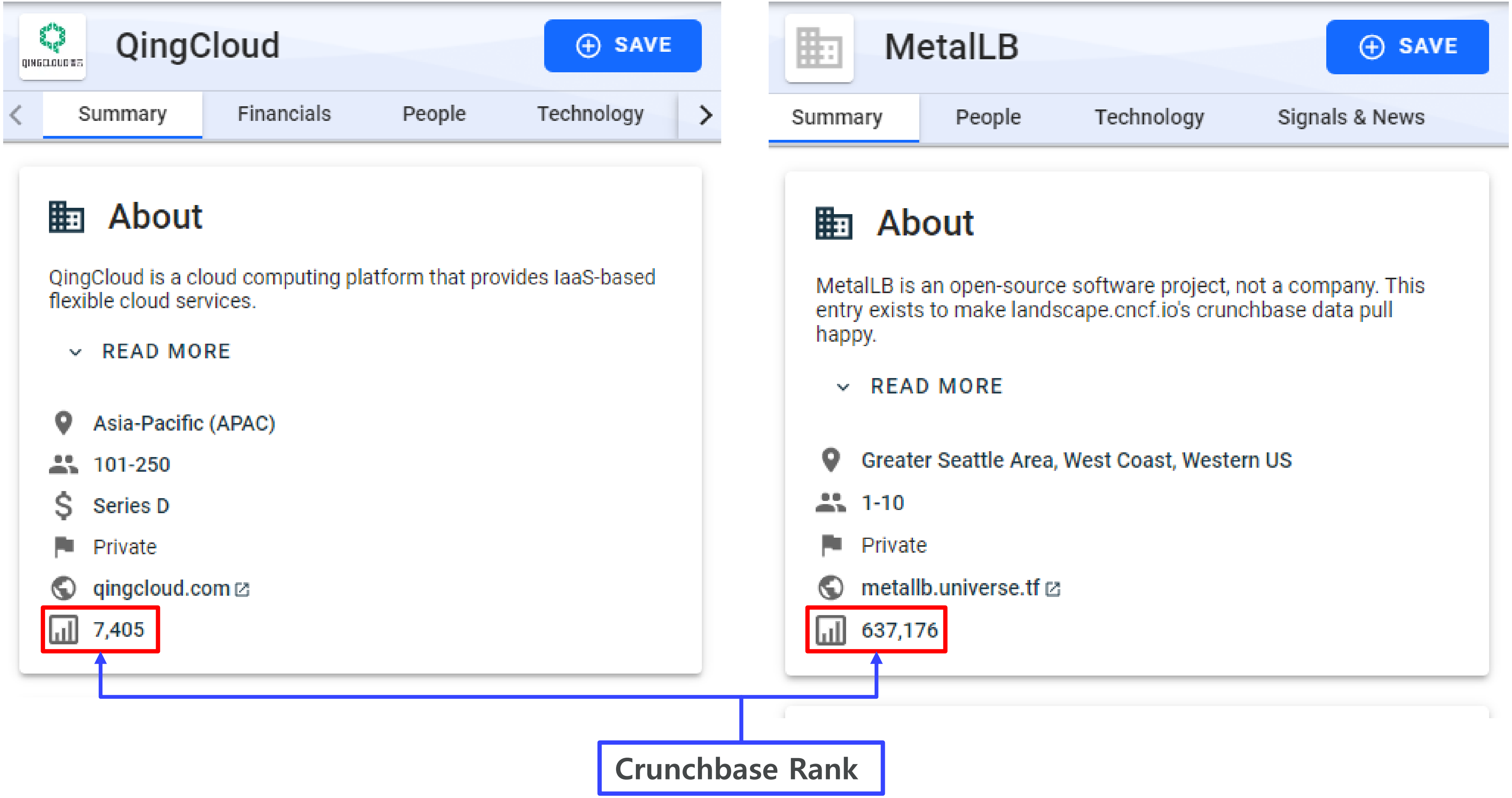
Website	metallb.universe.tf		
Repository	github.com/metallb/metallb	★3,199	
Crunchbase	crunchbase.com/organization/metallb		
Twitter	@dave_universetf	Latest Tweet	this week
First Commit	3 years ago	Latest Commit	last week
		Latest Release	last week
Contributors	75	Headcount	1-10
Headquarters	Seattle, Washington		

Tweets by @dave_universetf

Dave Anderson Retweeted

Funranium Labs

@funranium



About

QingCloud is a cloud computing platform that provides IaaS-based flexible cloud services.

Asia-Pacific (APAC)
101-250
Series D
Private
qingcloud.com
7,405

About

MetalLB is an open-source software project, not a company. This entry exists to make landscape.cncf.io's crunchbase data pull happy.

Greater Seattle Area, West Coast, Western US
1-10
Private
metallb.universe.tf
637,176

Crunchbase Rank

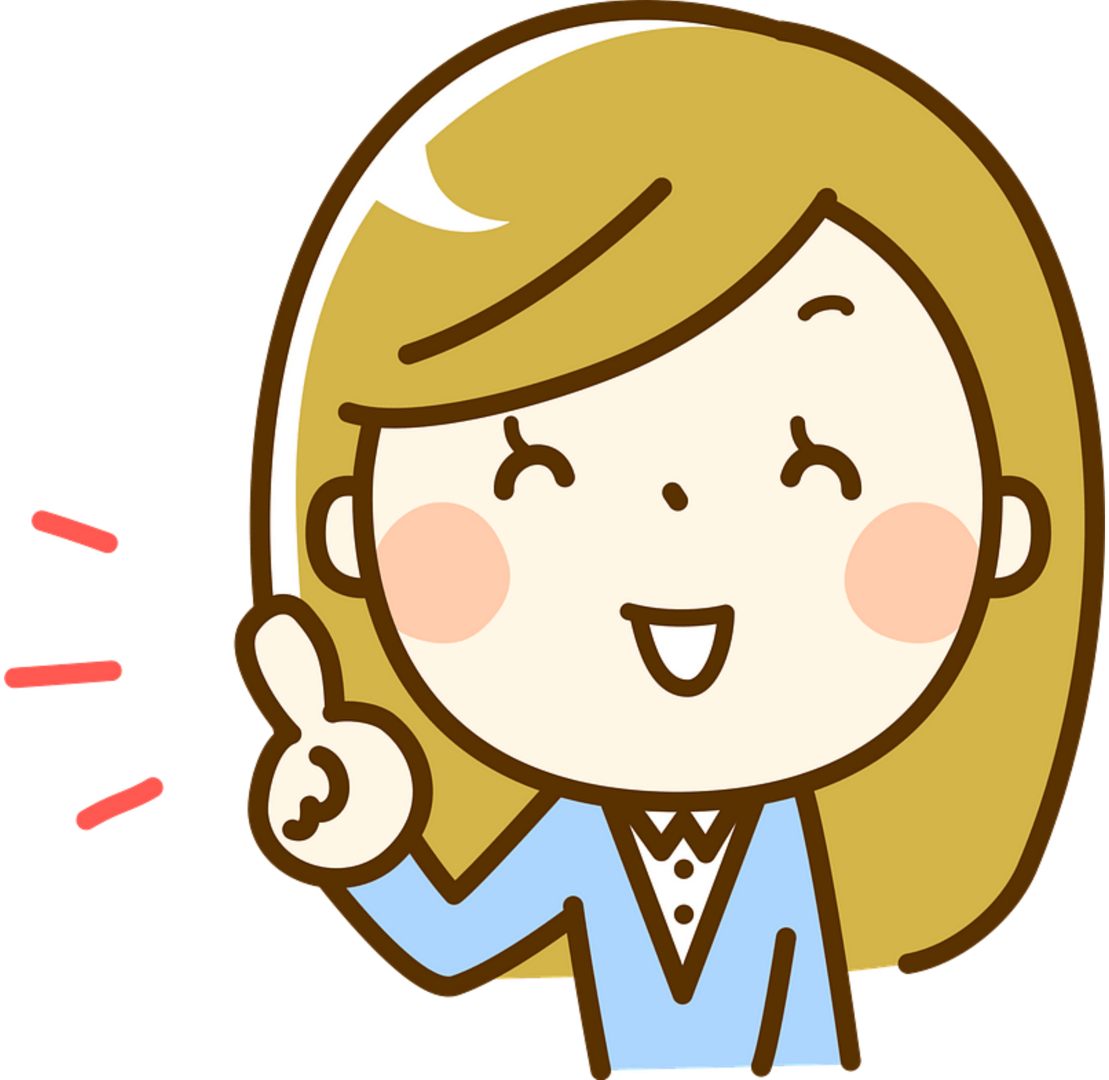
What is Crunchbase Rank?

Crunchbase Rank is a dynam rganizations, and Schools)
in the Crunchbase dataset. It

The Crunchbase Rank algorit ng the number of connections
a profile has, the level of com articles, and acquisitions.
A company’s Rank is fluid and ith time-sensitive events.
Events such as product launc nd news affect a company’s
Crunchbase Rank.

https://about.crunchbase.com

쉽게 말해서 투자 가치가 있다
조오오다 !



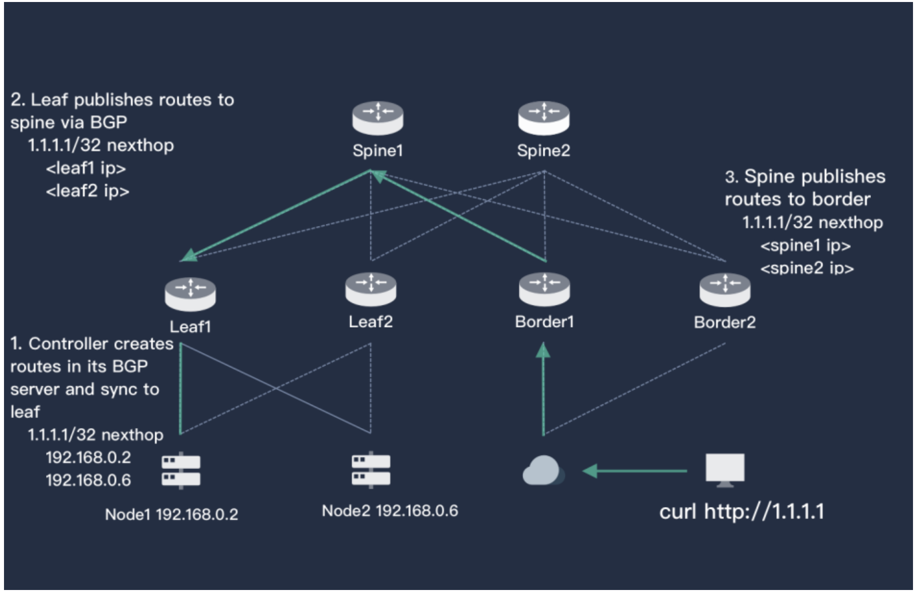
2.Porter와 MetalLB 기술영역 비교

Porter LB

Core Features

- ECMP routing load balancing
- BGP dynamic routing configuration
- VIP management
- LoadBalancerIP assignment in Kubernetes services
- Installation with Helm Chart
- Dynamic BGP server configuration through CRD
- Dynamic BGP peer configuration through CRD

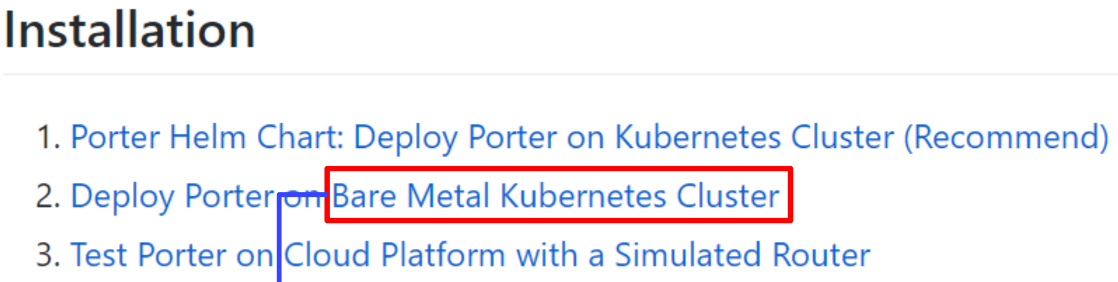
https://github.com/kubesphere/porter



2. Leaf publishes routes to spine via BGP
1.1.1.1/32 nexthop
<leaf1 ip>
<leaf2 ip>

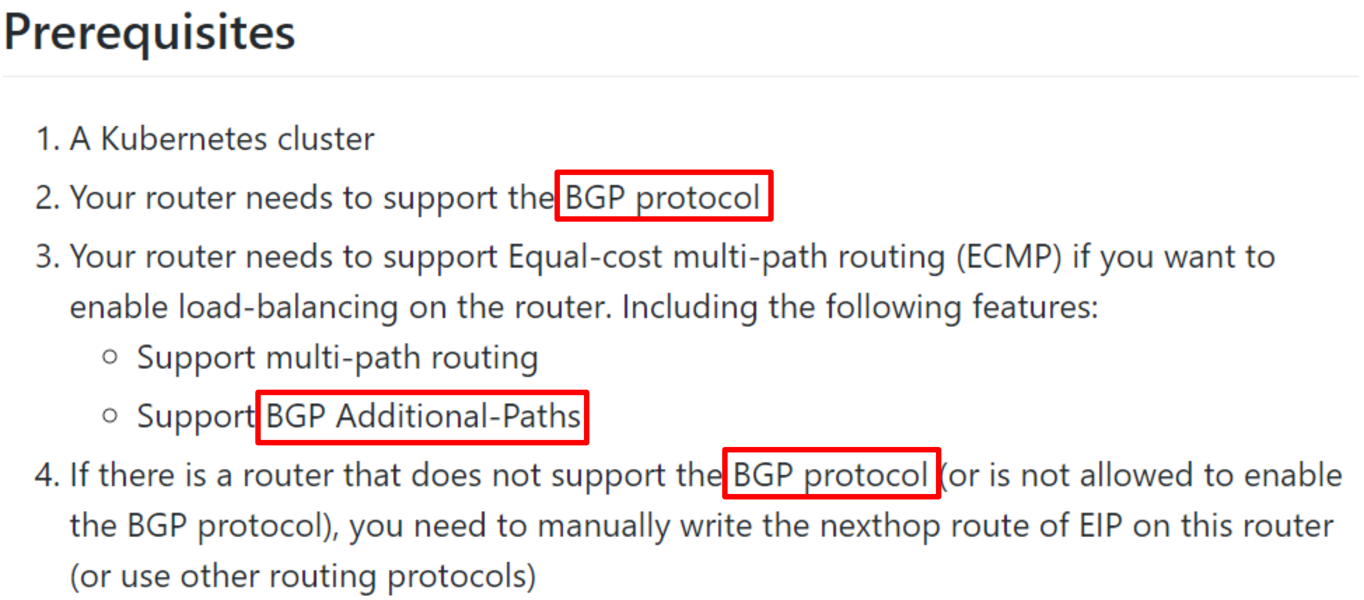
3. Spine publishes routes to border
1.1.1.1/32 nexthop
<spine1 ip>
<spine2 ip>

1. Controller creates routes in its BGP server and sync to leaf
1.1.1.1/32 nexthop
192.168.0.2
192.168.0.6



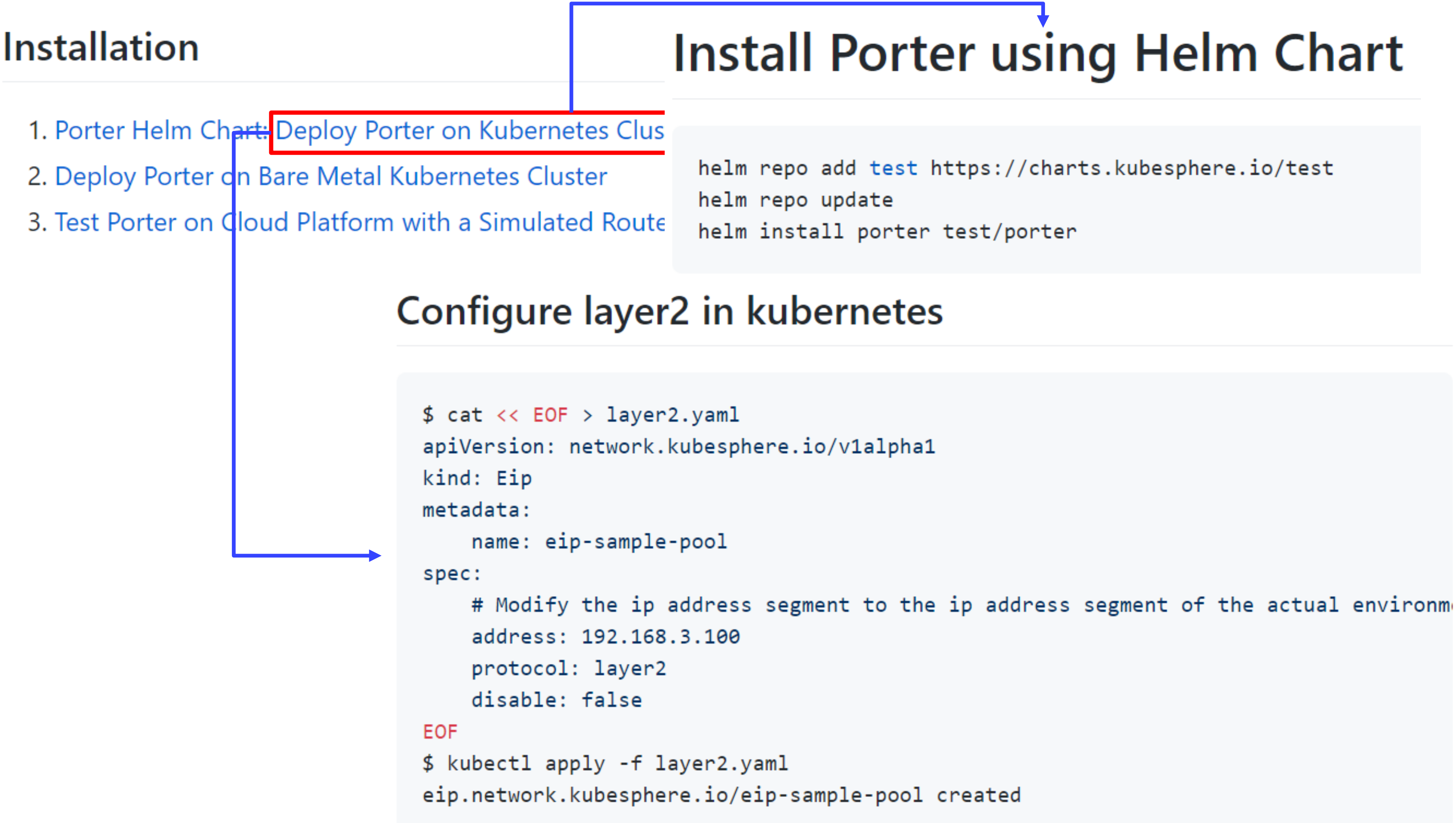
Installation

1. Porter Helm Chart: Deploy Porter on Kubernetes Cluster (Recommend)
2. Deploy Porter on Bare Metal Kubernetes Cluster
3. Test Porter on Cloud Platform with a Simulated Router



Prerequisites

1. A Kubernetes cluster
2. Your router needs to support the BGP protocol
3. Your router needs to support Equal-cost multi-path routing (ECMP) if you want to enable load-balancing on the router. Including the following features:
   - Support multi-path routing
   - Support BGP Additional-Paths
4. If there is a router that does not support the BGP protocol (or is not allowed to enable the BGP protocol), you need to manually write the nexthop route of EIP on this router (or use other routing protocols)



Installation

1. Porter Helm Chart: Deploy Porter on Kubernetes Clus
2. Deploy Porter on Bare Metal Kubernetes Cluster
3. Test Porter on Cloud Platform with a Simulated Route

Install Porter using Helm Chart

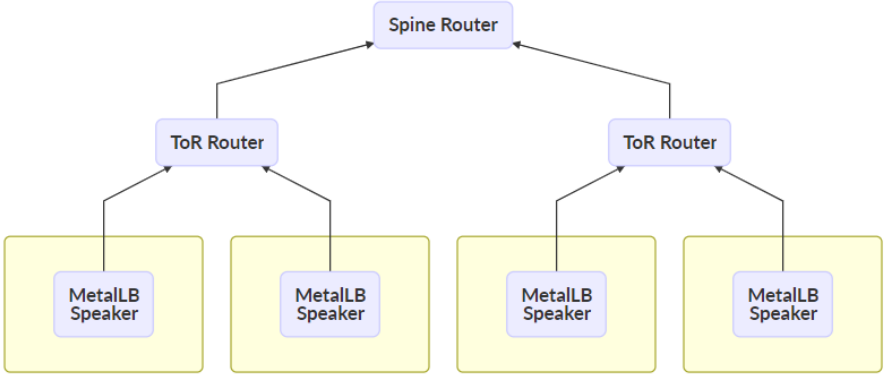
```
helm repo add test https://charts.kubesphere.io/test
helm repo update
helm install porter test/porter
```

Configure layer2 in kubernetes

```
$ cat << EOF > layer2.yaml
apiVersion: network.kubesphere.io/v1alpha1
kind: Eip
metadata:
    name: eip-sample-pool
spec:
    # Modify the ip address segment to the ip address segment of the actual environm
    address: 192.168.3.100
    protocol: layer2
    disable: false
EOF
$ kubectl apply -f layer2.yaml
eip.network.kubesphere.io/eip-sample-pool created
```

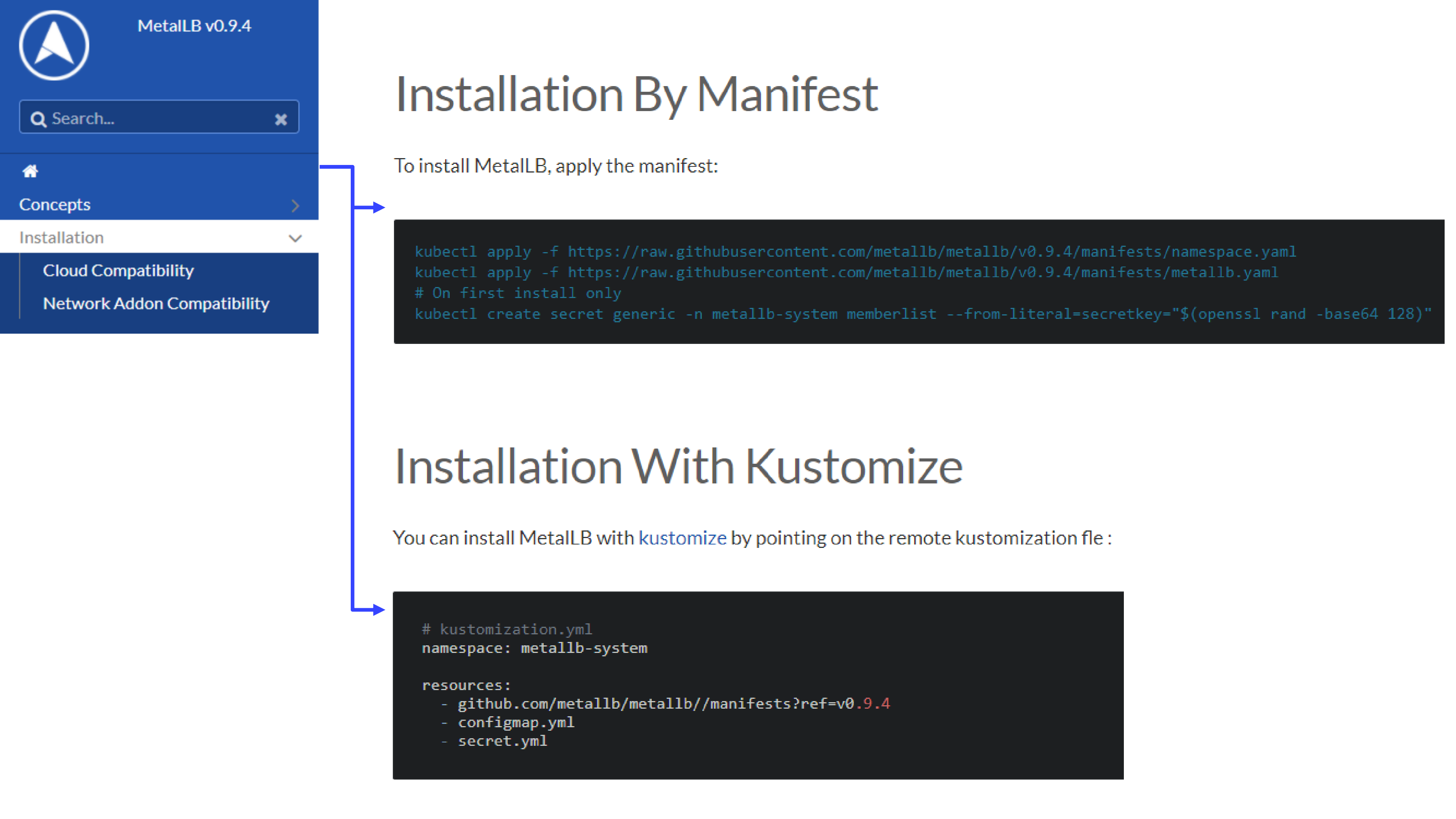
MetalLB

Layer 2 mode (ARP/NDP)
In layer 2 mode, one machine in the cluster takes ownership of the service, and uses standard address discovery protocols (for IPv4, for IPv6) to make those IPs reachable on the local network. From the LAN’s point of view, the announcing machine simply has multiple IP addresses. The sub-page has more details on the behavior and limitations of layer 2 mode.



BGP
In BGP mode, all machines in the cluster establish peering sessions with nearby routers that you control, and tell those routers how to forward traffic to the service IPs. Using BGP allows for true load balancing across multiple nodes, and fine-grained traffic control thanks to BGP’s policy mechanisms. The sub-page has more details on BGP mode’s operation and limitations.

https://metallb.universe.tf/



Installation By Manifest

To install MetalLB, apply the manifest:

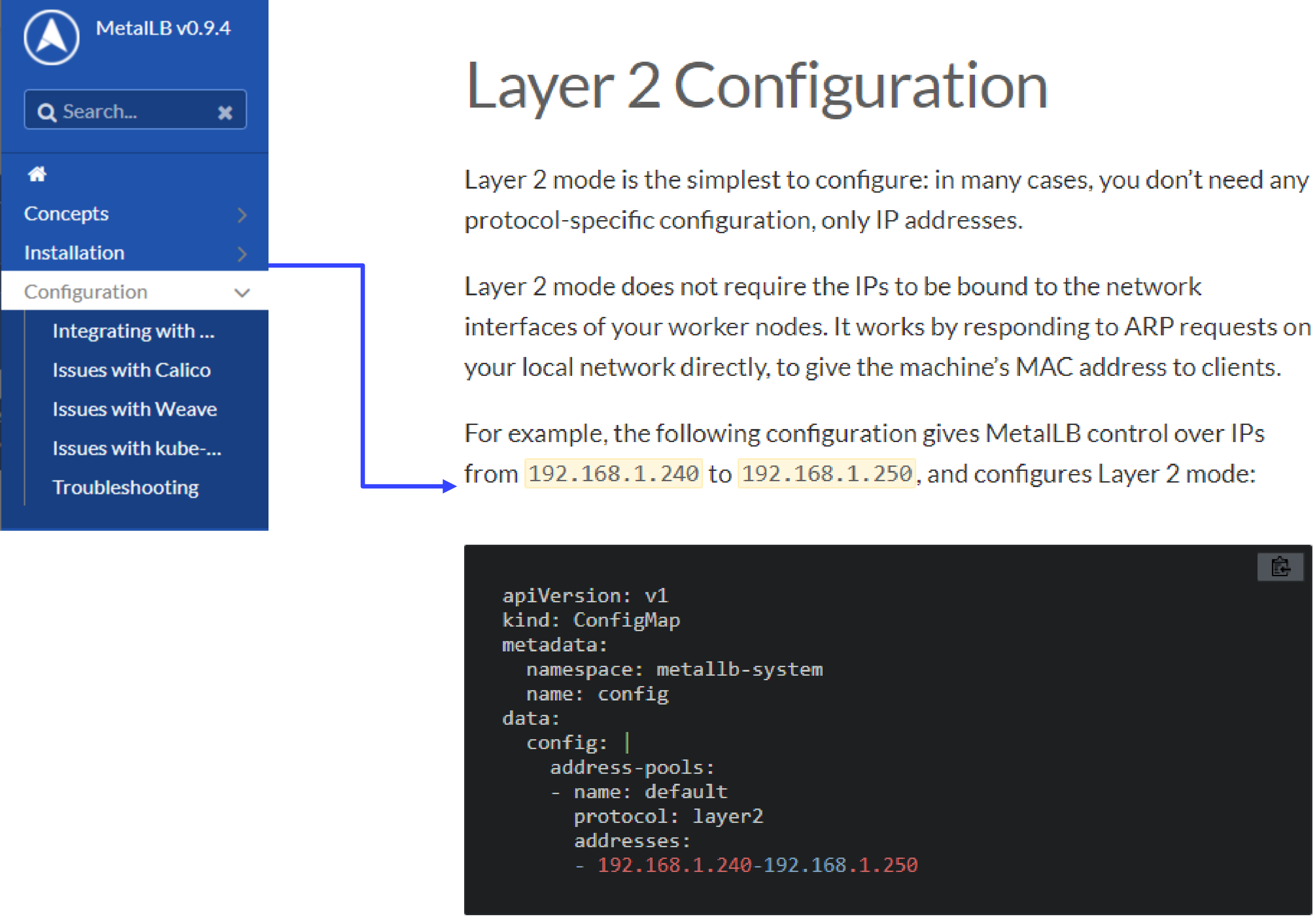
```
kubectl apply -f https://raw.githubusercontent.com/metallb/metallb/v0.9.4/manifests/namespace.yaml
kubectl apply -f https://raw.githubusercontent.com/metallb/metallb/v0.9.4/manifests/metallb.yaml
# On first install only
kubectl create secret generic -n metallb-system memberlist --from-literal=secretkey="$(openssl rand -base64 128)"
```

Installation With Kustomize

You can install MetalLB with kustomize by pointing on the remote kustomization file :

```
# kustomization.yml
namespace: metallb-system

resources:
  - github.com/metallb/metallb//manifests?ref=v0.9.4
  - configmap.yml
  - secret.yml
```



Layer 2 Configuration

Layer 2 mode is the simplest to configure: in many cases, you don't need any protocol-specific configuration, only IP addresses.

Layer 2 mode does not require the IPs to be bound to the network interfaces of your worker nodes. It works by responding to ARP requests on your local network directly, to give the machine's MAC address to clients.

For example, the following configuration gives MetalLB control over IPs from `192.168.1.240` to `192.168.1.250`, and configures Layer 2 mode:

```yaml
apiVersion: v1
kind: ConfigMap
metadata:
  namespace: metallb-system
  name: config
data:
  config: |
    address-pools:
    - name: default
      protocol: layer2
      addresses:
      - 192.168.1.240-192.168.1.250
```

Porter LB

Compared with MetalLB
Apparently, Porter is similar to MetalLB, both are service proxy, and they are designed for bare metal Kubernetes clusters as well.

우리는 경쟁자로 인식한다.

Cons

•Support Linux only.

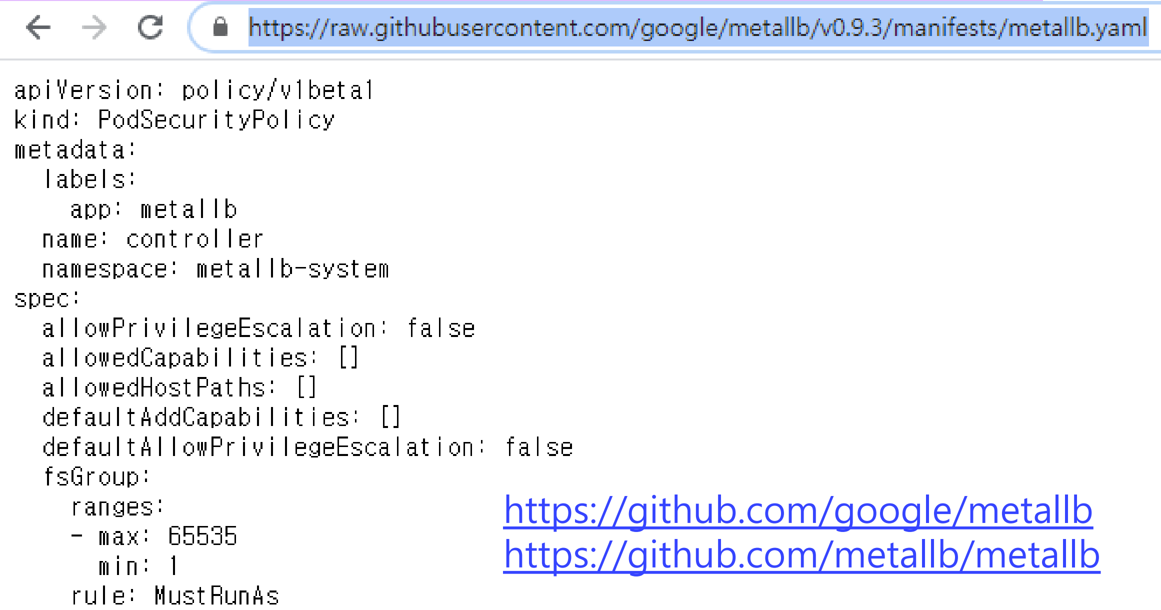
https://github.com/kubesphere/porter

MetalLB

Developers

MetalLB's copyright was owned by Google, until March 2019. However, it was never an official Google project. The project doesn't have any form of corporate sponsorship.
MetalLB was created by , working on MetalLB in their spare time as motivation allows. The original

우리는 구글

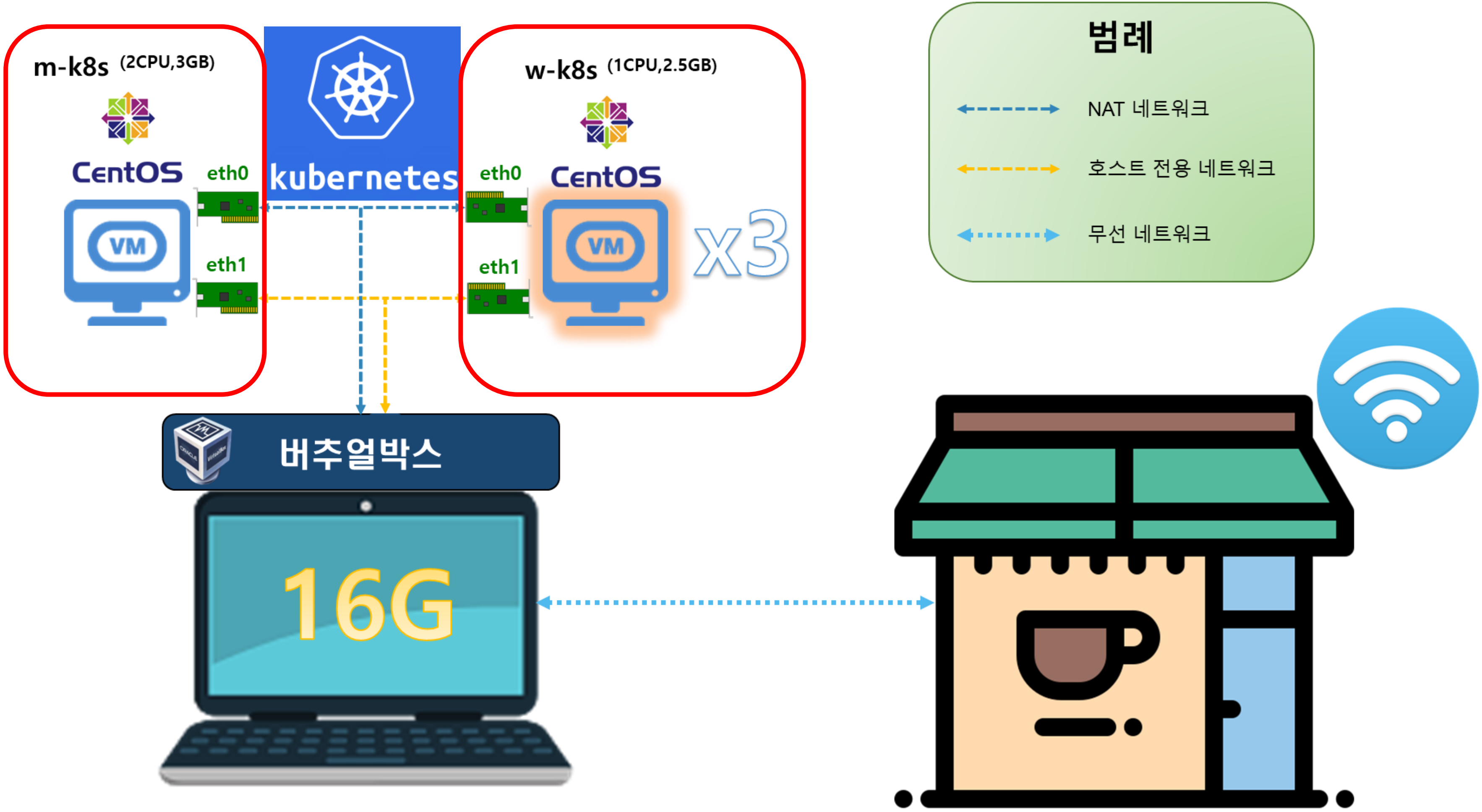


https://raw.githubusercontent.com/google/metallb/v0.9.3/manifests/metallb.yaml

```
apiVersion: policy/v1beta1
kind: PodSecurityPolicy
metadata:
  labels:
    app: metallb
  name: controller
  namespace: metallb-system
spec:
  allowPrivilegeEscalation: false
  allowedCapabilities: []
  allowedHostPaths: []
  defaultAddCapabilities: []
  defaultAllowPrivilegeEscalation: false
  fsGroup:
    ranges:
    - max: 65535
      min: 1
    rule: MustRunAs
```

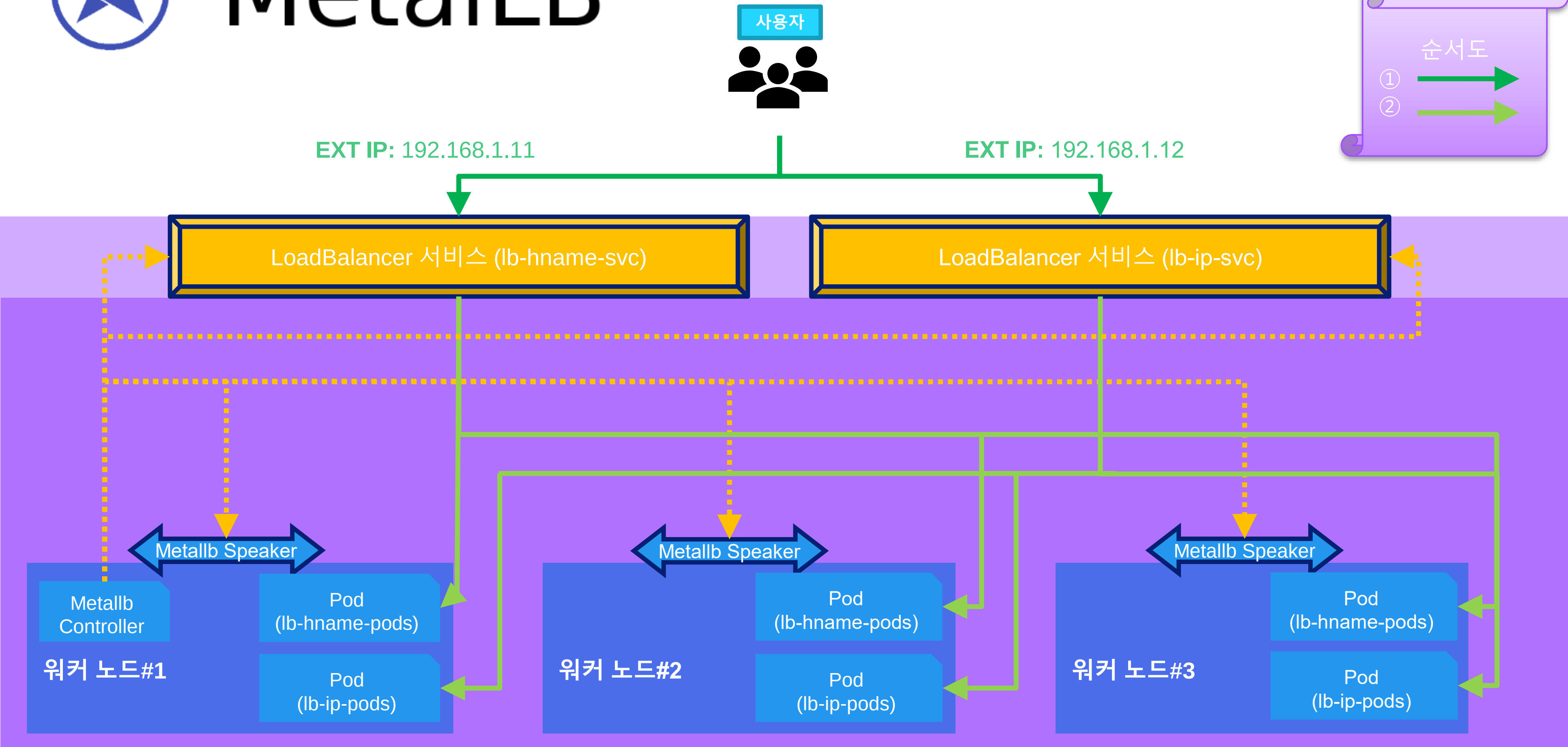
https://github.com/google/metallb
https://github.com/metallb/metallb

3.Porter와 MetalLB 실제 동작 확인



https://github.com/sysnet4admin/IaC/tree/master/k8s-SingleMaster-18.9_9_w_auto-compl

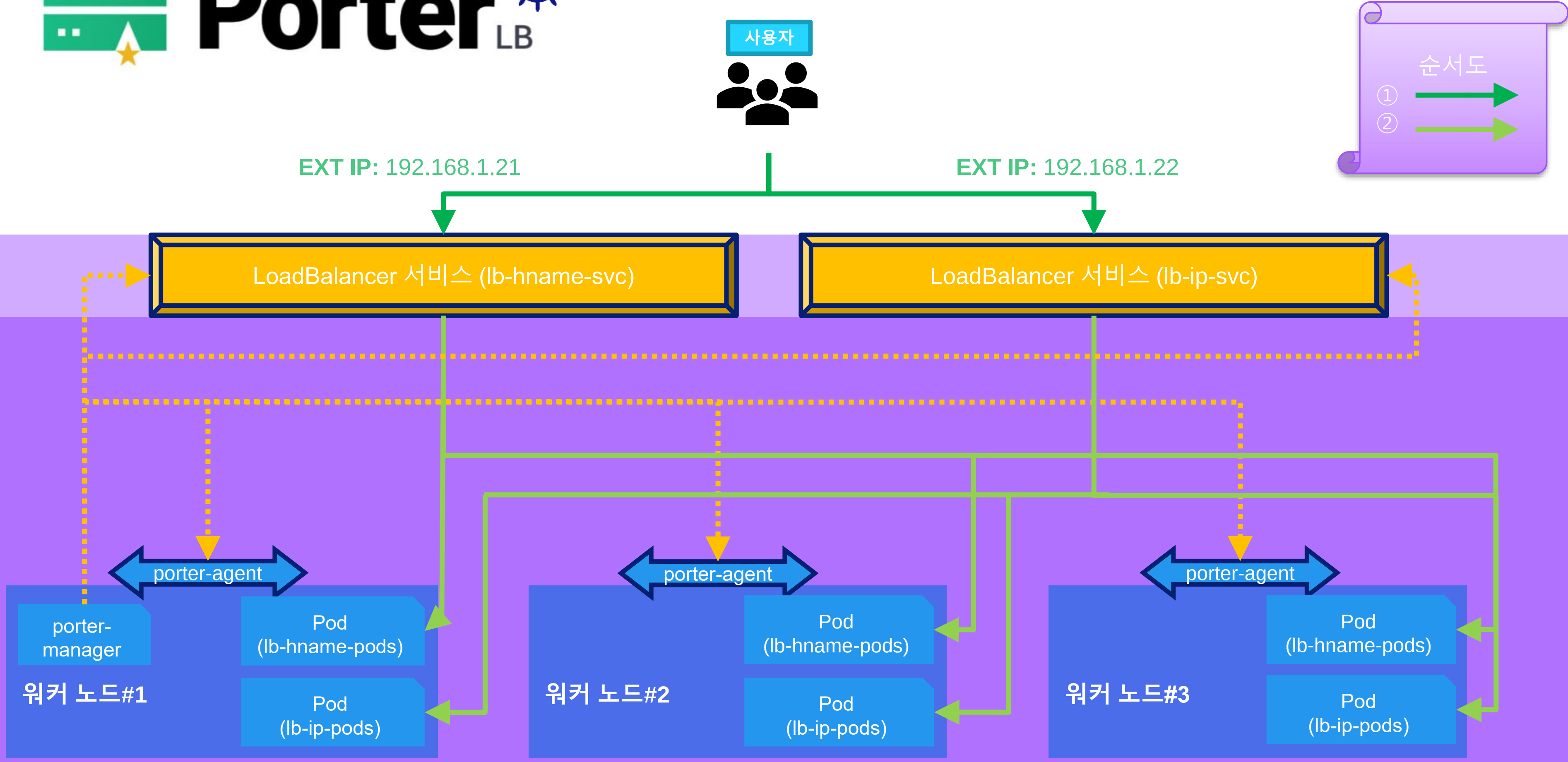
MetalLB


사용자
EXT IP: 192.168.1.11
EXT IP: 192.168.1.12
순서도
①
②
LoadBalancer 서비스 (lb-hname-svc)
LoadBalancer 서비스 (lb-ip-svc)
Metallb Speaker
Metallb Controller
Pod (lb-hname-pods)
Pod (lb-ip-pods)
워커 노드#1
Metallb Speaker
Pod (lb-hname-pods)
Pod (lb-ip-pods)
워커 노드#2
Metallb Speaker
Pod (lb-hname-pods)
Pod (lb-ip-pods)
워커 노드#3

Proof Of Concept

Porter LB

HELM



사용자

EXT IP: 192.168.1.21

EXT IP: 192.168.1.22

순서도

①

②

LoadBalancer 서비스 (lb-hname-svc)

LoadBalancer 서비스 (lb-ip-svc)

porter-agent

porter-agent

porter-agent

porter-manager

Pod (lb-hname-pods)

Pod (lb-ip-pods)

워커 노드#1

Pod (lb-hname-pods)

Pod (lb-ip-pods)

워커 노드#2

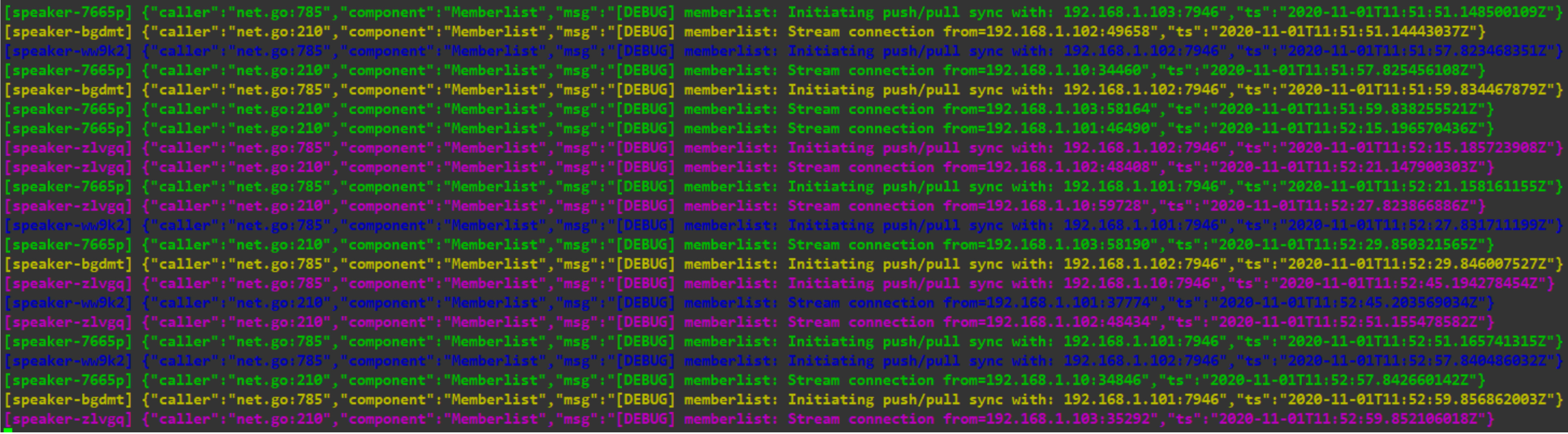
Pod (lb-hname-pods)

Pod (lb-ip-pods)

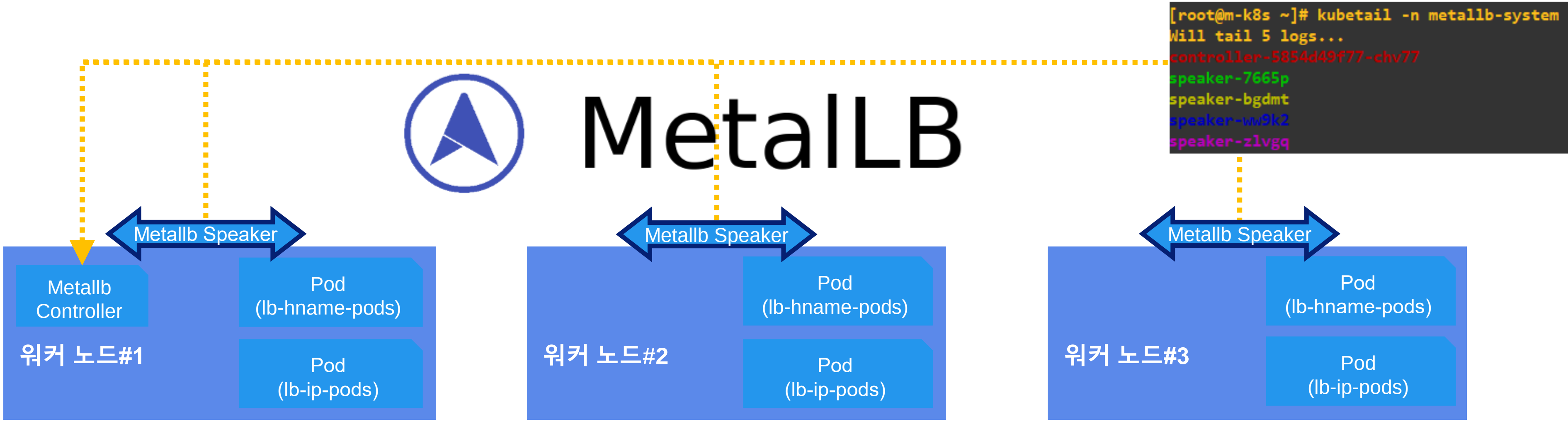
워커 노드#3

Proof Of Concept

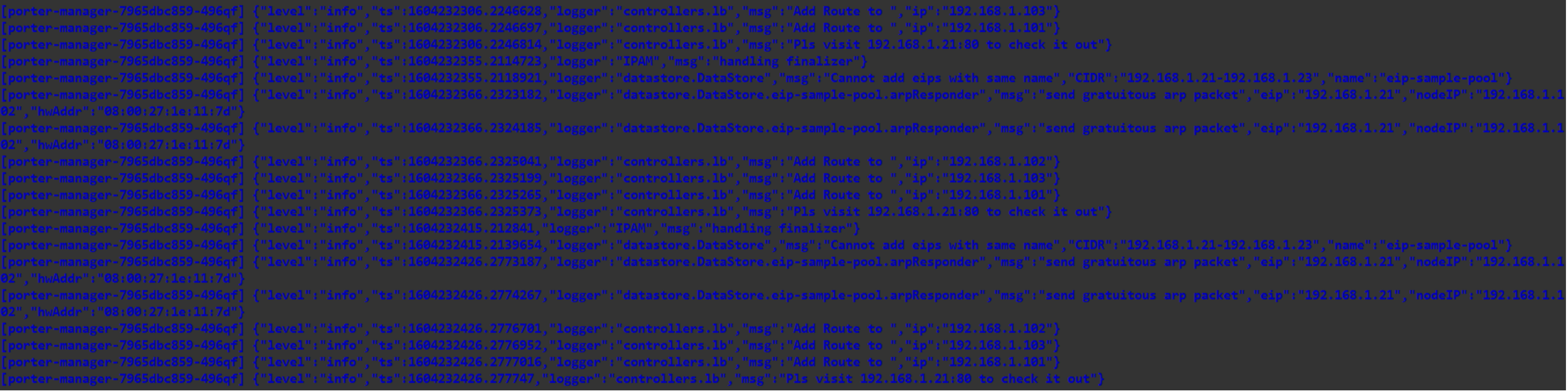
4. 결론 (TL;DR)



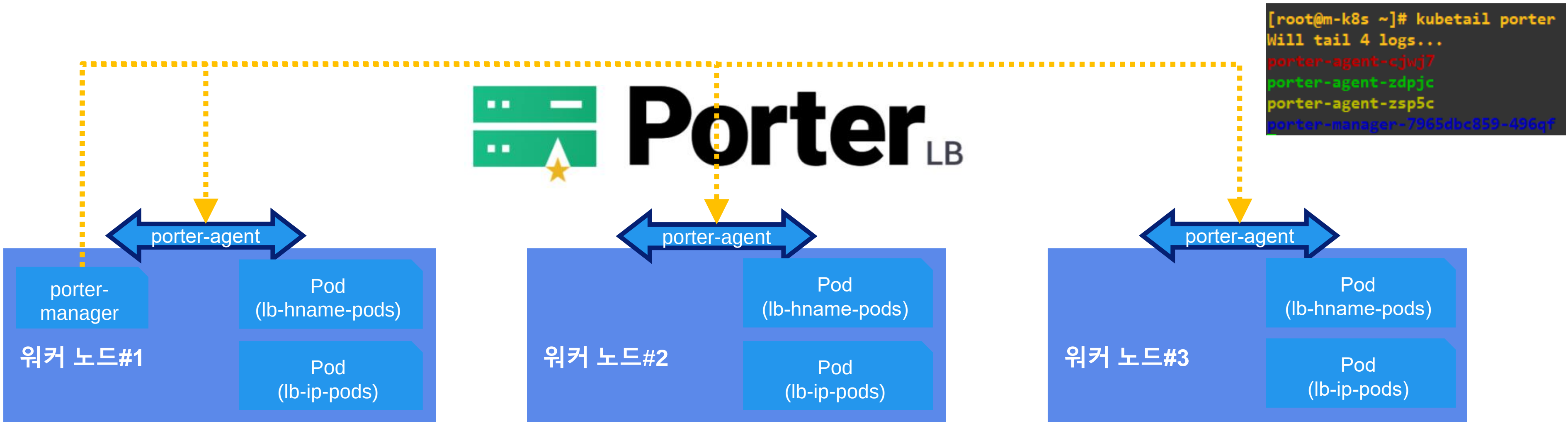
```
[speaker-7665p] {"caller":"net.go:785","component":"Memberlist","msg":"[DEBUG] memberlist: Initiating push/pull sync with: 192.168.1.103:7946","ts":"2020-11-01T11:51:51.148500109Z"}
[speaker-bgdmt] {"caller":"net.go:210","component":"Memberlist","msg":"[DEBUG] memberlist: Stream connection from=192.168.1.102:49658","ts":"2020-11-01T11:51:51.14443037Z"}
[speaker-ww9k2] {"caller":"net.go:785","component":"Memberlist","msg":"[DEBUG] memberlist: Initiating push/pull sync with: 192.168.1.102:7946","ts":"2020-11-01T11:51:57.823468351Z"}
[speaker-7665p] {"caller":"net.go:210","component":"Memberlist","msg":"[DEBUG] memberlist: Stream connection from=192.168.1.10:34460","ts":"2020-11-01T11:51:57.825456108Z"}
[speaker-bgdmt] {"caller":"net.go:785","component":"Memberlist","msg":"[DEBUG] memberlist: Initiating push/pull sync with: 192.168.1.102:7946","ts":"2020-11-01T11:51:59.834467879Z"}
[speaker-7665p] {"caller":"net.go:210","component":"Memberlist","msg":"[DEBUG] memberlist: Stream connection from=192.168.1.103:58164","ts":"2020-11-01T11:51:59.838255521Z"}
[speaker-7665p] {"caller":"net.go:210","component":"Memberlist","msg":"[DEBUG] memberlist: Stream connection from=192.168.1.101:46490","ts":"2020-11-01T11:52:15.196570436Z"}
[speaker-zlvgq] {"caller":"net.go:785","component":"Memberlist","msg":"[DEBUG] memberlist: Initiating push/pull sync with: 192.168.1.102:7946","ts":"2020-11-01T11:52:15.185723908Z"}
[speaker-zlvgq] {"caller":"net.go:210","component":"Memberlist","msg":"[DEBUG] memberlist: Stream connection from=192.168.1.102:48408","ts":"2020-11-01T11:52:21.147900303Z"}
[speaker-7665p] {"caller":"net.go:785","component":"Memberlist","msg":"[DEBUG] memberlist: Initiating push/pull sync with: 192.168.1.101:7946","ts":"2020-11-01T11:52:21.158161155Z"}
[speaker-zlvgq] {"caller":"net.go:210","component":"Memberlist","msg":"[DEBUG] memberlist: Stream connection from=192.168.1.10:59728","ts":"2020-11-01T11:52:27.823866886Z"}
[speaker-ww9k2] {"caller":"net.go:785","component":"Memberlist","msg":"[DEBUG] memberlist: Initiating push/pull sync with: 192.168.1.101:7946","ts":"2020-11-01T11:52:27.831711199Z"}
[speaker-7665p] {"caller":"net.go:210","component":"Memberlist","msg":"[DEBUG] memberlist: Stream connection from=192.168.1.103:58190","ts":"2020-11-01T11:52:29.850321565Z"}
[speaker-bgdmt] {"caller":"net.go:785","component":"Memberlist","msg":"[DEBUG] memberlist: Initiating push/pull sync with: 192.168.1.102:7946","ts":"2020-11-01T11:52:29.846007527Z"}
[speaker-zlvgq] {"caller":"net.go:785","component":"Memberlist","msg":"[DEBUG] memberlist: Initiating push/pull sync with: 192.168.1.10:7946","ts":"2020-11-01T11:52:45.194278454Z"}
[speaker-ww9k2] {"caller":"net.go:210","component":"Memberlist","msg":"[DEBUG] memberlist: Stream connection from=192.168.1.101:37774","ts":"2020-11-01T11:52:45.203569034Z"}
[speaker-zlvgq] {"caller":"net.go:210","component":"Memberlist","msg":"[DEBUG] memberlist: Stream connection from=192.168.1.102:48434","ts":"2020-11-01T11:52:51.154785582Z"}
[speaker-7665p] {"caller":"net.go:785","component":"Memberlist","msg":"[DEBUG] memberlist: Initiating push/pull sync with: 192.168.1.101:7946","ts":"2020-11-01T11:52:51.165741315Z"}
[speaker-ww9k2] {"caller":"net.go:785","component":"Memberlist","msg":"[DEBUG] memberlist: Initiating push/pull sync with: 192.168.1.102:7946","ts":"2020-11-01T11:52:57.840486032Z"}
[speaker-7665p] {"caller":"net.go:210","component":"Memberlist","msg":"[DEBUG] memberlist: Stream connection from=192.168.1.10:34846","ts":"2020-11-01T11:52:57.842660142Z"}
[speaker-bgdmt] {"caller":"net.go:785","component":"Memberlist","msg":"[DEBUG] memberlist: Initiating push/pull sync with: 192.168.1.101:7946","ts":"2020-11-01T11:52:59.856862003Z"}
[speaker-zlvgq] {"caller":"net.go:210","component":"Memberlist","msg":"[DEBUG] memberlist: Stream connection from=192.168.1.103:35292","ts":"2020-11-01T11:52:59.852106018Z"}
```



```
[root@m-k8s ~]# kubetail -n metallb-system
Will tail 5 logs...
controller-5854d49f77-chv77
speaker-7665p
speaker-bgdmt
speaker-ww9k2
speaker-zlvgq
```



```
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232306.2246628,"logger":"controllers.lb","msg":"Add Route to ","ip":"192.168.1.103"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232306.2246697,"logger":"controllers.lb","msg":"Add Route to ","ip":"192.168.1.101"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232306.2246814,"logger":"controllers.lb","msg":"Pls visit 192.168.1.21:80 to check it out"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232355.2114723,"logger":"IPAM","msg":"handling finalizer"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232355.2118921,"logger":"datastore.DataStore","msg":"Cannot add eips with same name","CIDR":"192.168.1.21-192.168.1.23","name":"eip-sample-pool"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232366.2323182,"logger":"datastore.DataStore.eip-sample-pool.arpResponder","msg":"send gratuitous arp packet","eip":"192.168.1.21","nodeIP":"192.168.1.1
02","hwAddr":"08:00:27:1e:11:7d"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232366.2324185,"logger":"datastore.DataStore.eip-sample-pool.arpResponder","msg":"send gratuitous arp packet","eip":"192.168.1.21","nodeIP":"192.168.1.1
02","hwAddr":"08:00:27:1e:11:7d"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232366.2325041,"logger":"controllers.lb","msg":"Add Route to ","ip":"192.168.1.102"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232366.2325199,"logger":"controllers.lb","msg":"Add Route to ","ip":"192.168.1.103"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232366.2325265,"logger":"controllers.lb","msg":"Add Route to ","ip":"192.168.1.101"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232366.2325373,"logger":"controllers.lb","msg":"Pls visit 192.168.1.21:80 to check it out"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232415.212841,"logger":"IPAM","msg":"handling finalizer"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232415.2139654,"logger":"datastore.DataStore","msg":"Cannot add eips with same name","CIDR":"192.168.1.21-192.168.1.23","name":"eip-sample-pool"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232426.2773187,"logger":"datastore.DataStore.eip-sample-pool.arpResponder","msg":"send gratuitous arp packet","eip":"192.168.1.21","nodeIP":"192.168.1.1
02","hwAddr":"08:00:27:1e:11:7d"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232426.2774267,"logger":"datastore.DataStore.eip-sample-pool.arpResponder","msg":"send gratuitous arp packet","eip":"192.168.1.21","nodeIP":"192.168.1.1
02","hwAddr":"08:00:27:1e:11:7d"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232426.2776701,"logger":"controllers.lb","msg":"Add Route to ","ip":"192.168.1.102"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232426.2776952,"logger":"controllers.lb","msg":"Add Route to ","ip":"192.168.1.103"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232426.2777016,"logger":"controllers.lb","msg":"Add Route to ","ip":"192.168.1.101"}
[porter-manager-7965dbc859-496qf] {"level":"info","ts":1604232426.277747,"logger":"controllers.lb","msg":"Pls visit 192.168.1.21:80 to check it out"}
```



```
[root@m-k8s ~]# kubetail porter
Will tail 4 logs...
porter-agent-cjwj7
porter-agent-zdpjc
porter-agent-zsp5c
porter-manager-7965dbc859-496qf
```

MetalLB는 **Porter**와 비교할때 기본적인 동작에는 차이가 있지만 기능적으로 동일하며, 쿠버네티스답게(?) 동작하는 것은 MetallLB로 보여집니다. 그리고 여러가지 측면으로 볼때 현재에는 MetallLB가 안정적일 것으로 보입니다.

따라서 온프레미스에서 쿠버네티스 로드밸런서가 필요하다면 **MetalLB**를 쓰세요. 지금 바로 쓰세요!

다만 기본설정은 네트워크 관점에서는 효율적으로 동작하지 않으니 **네트워크 엔지니어**와 상담해서 구체적으로 구성을 잡아나가세요.

감사……!
압도적
감사…!

질문 시간 입니다!!!