Introduction to Minikube

TrendMicro Consumer WSE

Eric C Huang

2017/7/14

What is Minikube? v0.20.0

Minikube is a tool that makes it easy to run Kubernetes locally.

https://github.com/kubernetes/minikube

Platforms

Virtualization

- VirtualBox
- Hyper-V

- VirtualBox
- xhyve
- VMWare Fusion

- VirtualBox
- KVM

Support Features

DNS	NodePorts	Ingress	~~LoadBalancer~~
Container Runtime	Persistent Volumes	ConfigMaps Secrets	Dashboards

https://goo.gl/Xuz8sL

Minikube Architecture

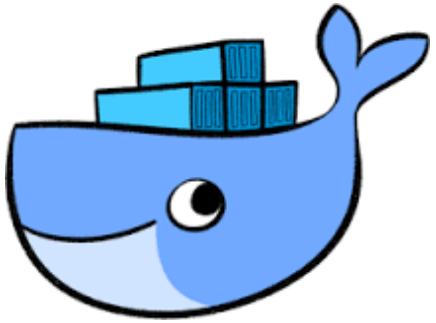



VM

Master

Node

localkube



Container Runtime

Minikube Installation - macOS

$ brew install kubectl

```
Updating Homebrew...
```

```
==> Downloading https://homebrew.bintray.com/bottles/kubernetes-cli-1.7.0.sierra.bottle.tar.gz
######################################################################## 100.0%
==> Pouring kubernetes-cli-1.7.0.sierra.bottle.tar.gz
==> Using the sandbox
==> Caveats
Bash completion has been installed to:
  /usr/local/etc/bash_completion.d

zsh functions have been installed to:
  /usr/local/share/zsh/site-functions
==> Summary
🍺  /usr/local/Cellar/kubernetes-cli/1.7.0: 5 files, 69.2MB
```

$ kubectl version --client

```
Client Version: version.Info{Major:"1", Minor:"7", GitVersion:"v1.7.0", GitCommit:"d3ada
0119e776222f11ec7945e6d860061339aad", GitTreeState:"clean", BuildDate:"2017-06-30T09:51:
01Z", GoVersion:"go1.8.3", Compiler:"gc", Platform:"darwin/amd64"}
```

$ brew cask install minikube

```
==> Satisfying dependencies
All Formula dependencies satisfied.
==> Downloading https://storage.googleapis.com/minikube/releases/v0.20.0/minikube-darwin
######################################################################## 100.0%
==> Verifying checksum for Cask minikube
==> Installing Cask minikube
==> Linking Binary 'minikube-darwin-amd64' to '/usr/local/bin/minikube'.
🍺 minikube was successfully installed!
```

$ minikube version

```
minikube version: v0.20.0
```

$ brew install docker-machine-driver-xhyve

```
==> Installing docker-machine-driver-xhyve
==> Downloading https://homebrew.bintray.com/bottles/docker-machine-driver-xhyve-0.3.3.s
######################################################################## 100.0%
==> Pouring docker-machine-driver-xhyve-0.3.3.sierra.bottle.tar.gz
==> Caveats
This driver requires superuser privileges to access the hypervisor. To
enable, execute
    sudo chown root:wheel /usr/local/opt/docker-machine-driver-xhyve/bin/docker-machine-driver-xhyve
    sudo chmod u+s /usr/local/opt/docker-machine-driver-xhyve/bin/docker-machine-driver-xhyve
==> Summary
🍺  /usr/local/Cellar/docker-machine-driver-xhyve/0.3.3: 3 files, 10.3MB
```

$ sudo chown root:wheel /usr/local/opt/docker-machine-driver-xhyve/bin/docker-machine-driver-xhyve

$ sudo chmod u+s /usr/local/opt/docker-machine-driver-xhyve/bin/docker-machine-driver-xhyve

Kubernetes Versions

$ minikube get-k8s-version

```
The following Kubernetes versions are available:
        - v1.7.0
        - v1.7.0-rc.1
        - v1.7.0-alpha.2
        - v1.6.4
        - v1.6.3
        - v1.6.0
        - v1.6.0-rc.1
        - v1.6.0-beta.4
        - v1.6.0-beta.3
        - v1.6.0-beta.2
        - v1.6.0-alpha.1
        - v1.6.0-alpha.0
```

```
        - v1.5.3
        - v1.5.2
        - v1.5.1
        - v1.4.5
        - v1.4.3
        - v1.4.2
        - v1.4.1
        - v1.4.0
        - v1.3.7
        - v1.3.6
        - v1.3.5
        - v1.3.4
        - v1.3.3
        - v1.3.0
```

$ minikube start --vm-driver=xhyve --kubernetes-version="v1.6.4" --cpus=2 --memory=1024 --disk-size=8g

```
Starting local Kubernetes v1.6.4 cluster...
Starting VM...
Moving files into cluster...
Setting up certs...
Starting cluster components...
Connecting to cluster...
Setting up kubeconfig...
Kubectl is now configured to use the cluster.
```

$ minikube status

```
minikube: Running
localkube: Running
kubectl: Correctly Configured: pointing to minikube-vm at 192.168.64.5
```

$ kubectl config current-context

```
minikube
```

$ kubectl get node

```
NAME       STATUS    AGE       VERSION
minikube   Ready     1m        v1.6.4
```

$ kubectl version

```
Client Version: version.Info{Major:"1", Minor:"7", GitVersion:"v1.7.0", GitCommit:"d3ada
0119e776222f11ec7945e6d860061339aad", GitTreeState:"clean", BuildDate:"2017-06-30T09:51:
01Z", GoVersion:"go1.8.3", Compiler:"gc", Platform:"darwin/amd64"}
Server Version: version.Info{Major:"1", Minor:"6", GitVersion:"v1.6.4", GitCommit:"d6f43
3224538d4f9ca2f7ae19b252e6fcb66a3ae", GitTreeState:"dirty", BuildDate:"2017-06-22T04:31:
09Z", GoVersion:"go1.7.5", Compiler:"gc", Platform:"linux/amd64"}
```

$ minikube stop

```
Stopping local Kubernetes cluster...
Machine stopped.
```

$ minikube delete

```
Deleting local Kubernetes cluster...
Machine deleted.
```

Minikube Installation - Windows

1. chocolatey package - Kubernetes Command Line Interface (CLI)
   $ choco install kubernetes-cli

2. chocolatey package - minikube
   $ choco install minikube

https://chocolatey.org/

Create a Simple Service on Minikube

- $ kubectl run nginx-node --image=nginx --port=80

```
deployment "nginx-node" created
```

- $ kubectl get pod

```
NAME                          READY     STATUS    RESTARTS   AGE
nginx-node-3344567341-hv78j   1/1       Running   0          3m
```

- $ kubectl get deployment

```
NAME         DESIRED   CURRENT   UP-TO-DATE   AVAILABLE   AGE
nginx-node   1         1         1            1           4m
```

- $ kubectl expose deployment nginx-node --type=NodePort

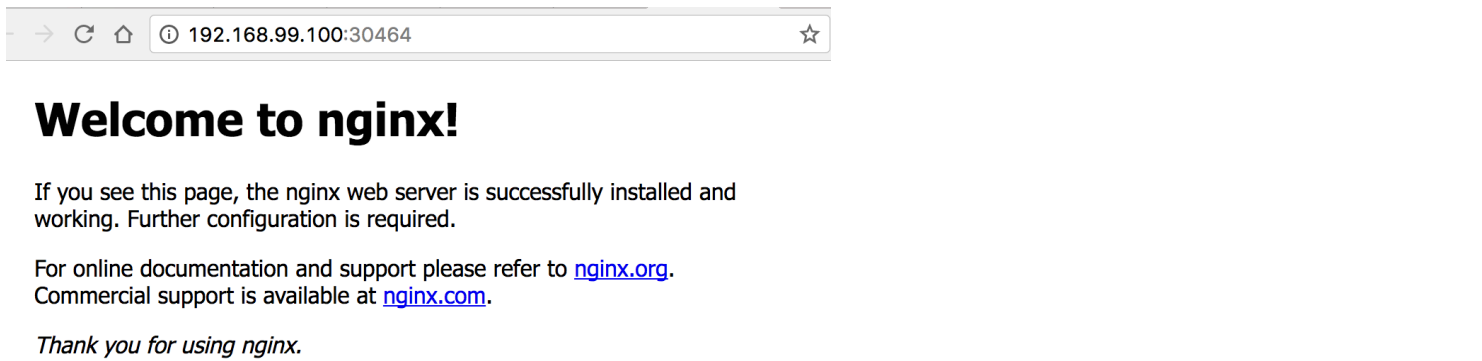
```
service "nginx-node" exposed
```

$ kubectl get service

```
NAME         CLUSTER-IP   EXTERNAL-IP   PORT(S)        AGE
kubernetes   10.0.0.1     <none>        443/TCP        2h
nginx-node   10.0.0.247   <nodes>       80:30464/TCP   3m
```

$ minikube service nginx-node

```
Opening kubernetes service default/nginx-node in default browser...
```



Create a Simple Service on Minikube

- $ kubectl run nginx-node --image=nginx --port=80
- $ kubectl get pod
- $ kubectl get deployment
- $ kubectl expose deployment nginx-node --type=NodePort
- $ kubectl get service
- $ kubectl describe service nginx-node
- $ minikube service nginx-node

Local
Minikube

- $ kubectl run nginx-node --image=nginx --port=80
- $ kubectl get pod
- $ kubectl get deployment
- $ kubectl expose deployment nginx-node --type=**LoadBalancer** **(AWS ELB)**
- $ kubectl get service
- $ kubectl describe service nginx-node

AWS
Kubernetes

Minikube Addons

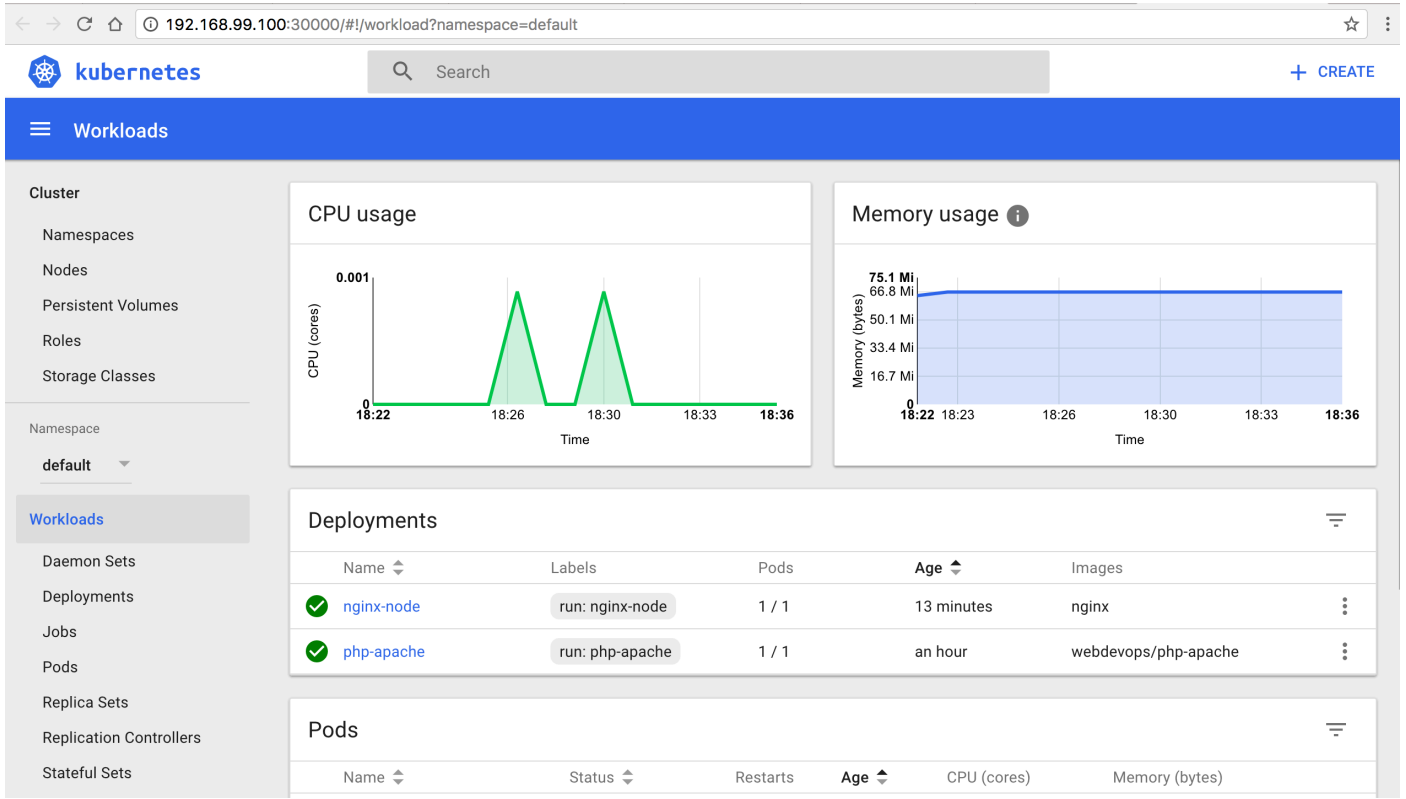
$ minikube addons list

```
- registry: disabled
- registry-creds: disabled
- addon-manager: enabled
- dashboard: enabled
- default-storageclass: enabled
- kube-dns: enabled
- heapster: disabled
- ingress: disabled
```

addon-manager | default-storageclass | kube-dns | dashboard | registry | registry-creds | heapster | ingress

Minikube Addons - dashboard

$ minikube service kubernetes-dashboard --namespace=kube-system



Minikube Addon - registry-creds

$ minikube addons configure registry-creds
(Pull image from **ECR/GCR/Docker Registry** on Minikube)

```
Do you want to enable AWS Elastic Container Registry? [y/n]: y
-- Enter AWS Access Key ID: AKI[redacted]5WA
-- Enter AWS Secret Access Key: qsgI+2[redacted]3avnlnqX4BCb
-- Enter AWS Region: us-east-1
-- Enter 12 digit AWS Account ID: 796[redacted]238

Do you want to enable Google Container Registry? [y/n]: n

Do you want to enable Docker Registry? [y/n]: n
registry-creds was successfully configured
```

$ minikube addons enableregistry-creds

- $ kubectl run unicorn-worker --image=79xxxxxxx238.dkr.ecr.us-east-1.amazonaws.com/signin_service/unicorn:0.x.x-1x9

```
NAME                              READY     STATUS    RESTARTS   AGE
nginx-node-3344567341-6dxfr       1/1       Running   0          1h
nginx-node-3344567341-91tcg       1/1       Running   0          1h
nginx-node-3344567341-dfs5x       1/1       Running   0          1h
unicorn-worker-3440052272-41z9c   1/1       Running   0          9m
unicorn-worker-3440052272-m9bzw   1/1       Running   0          9m
unicorn-worker-3440052272-qcvl5   1/1       Running   0          9m
```

- $ kubectl describe pod unicorn-worker-xxxxxxxxx

```
  FirstSeen     LastSeen        Count   From                    SubObjectPath                   Type            Reason          Message
  ---------     --------        -----   ----                    -------------                   --------        ------          -------
  11m           11m             1       default-scheduler                                       Normal          Scheduled       Successfully assigned unicorn-worker-3440052272-41
z9c to minikube
  11m           11m             1       kubelet, minikube       spec.containers{unicorn-worker} Normal          Pulling         pulling image "7[redacted]238.dkr.ecr.us-east-1.amaz
onaws.com/signin_service/unicorn:0.[redacted]"
  1m            1m              1       kubelet, minikube       spec.containers{unicorn-worker} Normal          Pulled          Successfully pulled image "79[redacted]238.dkr.ecr.us
-east-1.amazonaws.com/signin_service/unicorn:0.3.6-109"
  1m            1m              1       kubelet, minikube       spec.containers{unicorn-worker} Normal          Created         Created container with id 4381[redacted]f4a98827
ea0881078707dd819b978f6b3944f14b23fb4e21
  1m            1m              1       kubelet, minikube       spec.containers{unicorn-worker} Normal          Started         Started container with id 438137[redacted]8827
ea0881078707dd819b978f6b3944f14b23fb4e21
```

Pull image from ECR on AWS

- **Using AWS EC2 Container Registry**
- Image: **ACCOUNT.dkr.ecr.REGION.amazonaws.com/imagename:tag**
- IAM Policy:
  - ecr:GetAuthorizationToken
  - ecr:BatchCheckLayerAvailability
  - ecr:GetDownloadUrlForLayer
  - ecr:GetRepositoryPolicy
  - ecr:DescribeRepositories
  - ecr:ListImages
  - ecr:BatchGetImage

https://goo.gl/bwTAM8

Minikube Addon - Heapster

- $ minikube addons enable heapster
- $ minikube service monitoring-grafana --namespace kube-system

HPA (Horizontal Pod Autoscaling)

- $ kubectl run nginx --image=nginx --port=80 --limits='cpu=20m'
- $ kubectl expose deployment nginx --type=NodePort
- $ minikube service nginx
- $ kubectl autoscale deployment nginx --min=1 --max=5 --cpu-percent=20
- $ kubectl get hpa (horizontalpodautoscalers)

```
NAME      REFERENCE                TARGETS           MINPODS   MAXPODS   REPLICAS   AGE
nginx     Deployment/nginx-node    <unknown> / 20%   1         5         1          12m
```

https://goo.gl/WSzxq6

- $ kubectl exec nginx-4105716676-3sb64 -it bash
  - $ apt-get update && apt-get install stress
  - $ stress --cpu 100 --io 100 --vm 10 -i 100
- $ kubectl get hpa

```
NAME      REFERENCE          TARGETS     MINPODS   MAXPODS   REPLICAS   AGE
nginx     Deployment/nginx   23% / 20%   1         5         2          5m
```

Scale Out

```
NAME      REFERENCE          TARGETS     MINPODS   MAXPODS   REPLICAS   AGE
nginx     Deployment/nginx   49% / 20%   1         5         4          8m
```

Scale Out

```
NAME      REFERENCE          TARGETS     MINPODS   MAXPODS   REPLICAS   AGE
nginx     Deployment/nginx   0% / 20%    1         5         1          12m
```

Scale IN

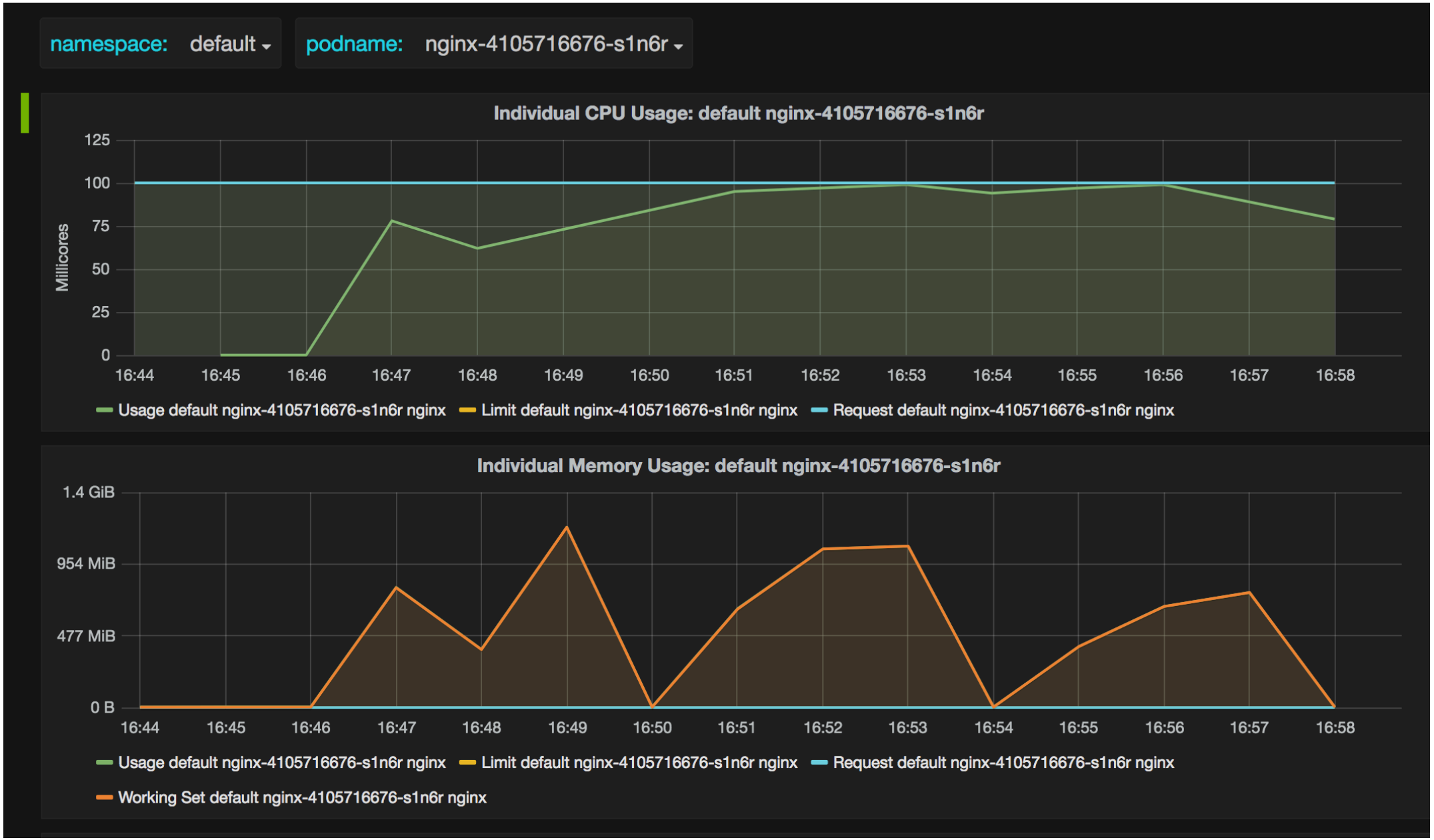
```
NAME      REFERENCE          TARGETS     MINPODS   MAXPODS   REPLICAS   AGE
nginx     Deployment/nginx   95% / 20%   1         5         4          19m
```

Scale Out

```
NAME      REFERENCE          TARGETS     MINPODS   MAXPODS   REPLICAS   AGE
nginx     Deployment/nginx   36% / 20%   1         5         5          22m
```

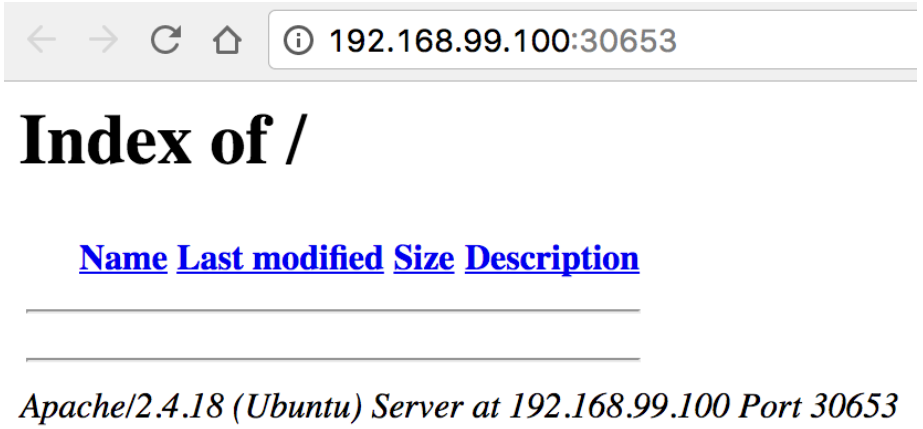
Scale Out

```
NAME                     READY   STATUS    RESTARTS   AGE
nginx-4105716676-05p5j   1/1     Running   0          1m
nginx-4105716676-30vg6   1/1     Running   0          5m
nginx-4105716676-3sb64   1/1     Running   0          5m
nginx-4105716676-8n2h5   1/1     Running   0          5m
nginx-4105716676-s1n6r   1/1     Running   0          24m
```


namespace: default
podname: nginx-4105716676-s1n6r
Individual CPU Usage: default nginx-4105716676-s1n6r
125
100
75
50
25
0
Millicores
16:44
16:45
16:46
16:47
16:48
16:49
16:50
16:51
16:52
16:53
16:54
16:55
16:56
16:57
16:58
Usage default nginx-4105716676-s1n6r nginx
Limit default nginx-4105716676-s1n6r nginx
Request default nginx-4105716676-s1n6r nginx
Individual Memory Usage: default nginx-4105716676-s1n6r
1.4 GiB
954 MiB
477 MiB
0 B
16:44
16:45
16:46
16:47
16:48
16:49
16:50
16:51
16:52
16:53
16:54
16:55
16:56
16:57
16:58
Usage default nginx-4105716676-s1n6r nginx
Limit default nginx-4105716676-s1n6r nginx
Request default nginx-4105716676-s1n6r nginx
Working Set default nginx-4105716676-s1n6r nginx

Minikube Addon - Ingress

- $ kubectl run nginx --image=nginx --port=80
- $ kubectl expose deployment nginx --type=NodePort
- $ kubectl run php-apache --image=webdevops/php-apache --port=80
- $ kubectl expose deployment php-apache --type=NodePort



$ kubectl create -f ingress_example.yaml

```yaml
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  name: test-ingress
spec:
  rules:
  - http:
      paths:
      - path: /
        backend:
          serviceName: nginx
          servicePort: 80
      - path: /my-php
        backend:
          serviceName: php-apache
          servicePort: 80
```

$ kubectl get ingress

```
NAME           HOSTS     ADDRESS          PORTS     AGE
test-ingress   *         192.168.99.100   80        10s
```

- https://192.168.99.100/

- https://192.168.99.100/my-php

(LoadBalancer) -> (Service) -> (Pod) -> (Server) ： L4 Load Balancer (TCP)

(LoadBalancer) -> (Ingress Service) -> (Ingress Pod) -> (Pod) -> (Server) ： L7 Load Balancer (HTTP)

Q & A