Navigating the Sea of Kubernetes Local Clusters

Ara Pulido, Developer Relations at Datadog
Datadog is a monitoring and analytics platform that helps companies improve observability of their infrastructure and applications.
Developer Relations at Datadog

Working on Kubernetes projects for 3 years

CKA; part of the team that created CKAD

@arapulido
ara.pulido@datadoghq.com
19 major releases since 2015

+90k commits

+2k contributors
Kubernetes popularity since 2016
Companies are choosing K8s

- Kubernetes is extensible and flexible (API driven)
- It has a large community
- It helps with multi-cloud strategy
Developer experience
No :(
Yes!
Why local clusters are useful?

- Great Learning Tool
- Quick Feedback Loop
- Good for CI/CD Workflows
The sea of local Kubernetes clusters
Containers
Linux Containers
The sea of local Kubernetes clusters

- MINIKUBE
- KIND
- MICROK8S
- K3S / K3D
- FIREKUBE
**minikube**

- Started in 2016, a year after the first Kubernetes release
- Cross platform (Linux, MacOS, Windows)
- Follows Kubernetes stable releases
- LoadBalancer, NodePort, Ingress, Container Runtimes
- Single Node Cluster
- Add-ons based
minikube

Linux VM as K8s node

Hypervisor

Host OS

Hardware
$ minikube status
There is no local cluster named "minikube"
To fix this, run: "minikube start"

$ minikube start
minikube v1.9.2 on Darwin 10.14.6
Automatically selected the hyperkit driver. Other choices: docker, virtualbox
Starting control plane node m01 in cluster minikube
Creating hyperkit VM (CPUs=2, Memory=4000MB, Disk=20000MB) ...
Preparing Kubernetes v1.18.0 on Docker 19.03.8 ...
Enabling addons: default-storageclass, storage-provisioner
Done! kubectl is now configured to use "minikube"
There is no local cluster named "minikube"
To fix this, run: "minikube start"

minikube v1.9.2 on Darwin 10.14.6

Automatically selected the hyperkit driver. Other choices: docker, virtualbox
Starting control plane node m01 in cluster minikube
Creating hyperkit VM (CPUs=2, Memory=4000MB, Disk=20000MB) ...
Preparing Kubernetes v1.18.0 on Docker 19.03.8 ...
Enabling addons: default-storageclass, storage-provisioner

Done! kubectl is now configured to use "minikube"

NAME STATUS   ROLES AGE VERSION
minikube Ready master 81m v1.18.0
Black Friday!

Enter coupon code 'BFRIDAY' at checkout for a $5.10 discount

Shop by Categories

- Bags
- Mugs
- Clothing

**COOL HATS**

<table>
<thead>
<tr>
<th>Item</th>
<th>Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datadog Tote</td>
<td>$15.99</td>
</tr>
<tr>
<td>Datadog Bag</td>
<td>$22.99</td>
</tr>
<tr>
<td>Datadog Baseball Jersey</td>
<td>$19.99</td>
</tr>
<tr>
<td>Datadog Jr. Spaghetti</td>
<td>$19.99</td>
</tr>
<tr>
<td>Datadog Ringer T-Shirt</td>
<td></td>
</tr>
<tr>
<td>Apache Baseball Jersey</td>
<td></td>
</tr>
<tr>
<td>Spree Baseball Jersey</td>
<td></td>
</tr>
<tr>
<td>Spree Jr. Spaghetti</td>
<td></td>
</tr>
</tbody>
</table>
kind - Kubernetes In Docker

- Started in 2019
- Works anywhere Docker works
- Was designed for automated testing of Kubernetes itself (CI)
- Uses “container nodes” with systemd, a CRI, Kubernetes
- Multi-node cluster
kind

kind: Cluster
apiVersion: kind.x-k8s.io/v1alpha4
nodes:
- role: control-plane
  image: kindest/node:v1.18.0
- role: worker
  image: kindest/node:v1.18.0
<table>
<thead>
<tr>
<th>NAME</th>
<th>STATUS</th>
<th>ROLES</th>
<th>AGE</th>
<th>VERSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>kind-control-plane</td>
<td>Ready</td>
<td>master</td>
<td>81m</td>
<td>v1.18.0</td>
</tr>
<tr>
<td>kind-worker</td>
<td>Ready</td>
<td>&lt;none&gt;</td>
<td>80m</td>
<td>v1.18.0</td>
</tr>
<tr>
<td>NAME</td>
<td>STATUS</td>
<td>ROLES</td>
<td>AGE</td>
<td>VERSION</td>
</tr>
<tr>
<td>-------------------</td>
<td>--------</td>
<td>---------</td>
<td>-----</td>
<td>---------</td>
</tr>
<tr>
<td>kind-control-plane</td>
<td>Ready</td>
<td>master</td>
<td>81m</td>
<td>v1.18.0</td>
</tr>
<tr>
<td>kind-worker</td>
<td>Ready</td>
<td>&lt;none&gt;</td>
<td>80m</td>
<td>v1.18.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CONTAINER ID</th>
<th>IMAGE</th>
<th>COMMAND</th>
<th>CREATED</th>
<th>STATUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>49c35555d26a</td>
<td>kindest/node:v1.18.0</td>
<td>&quot;/usr/local/bin/entr...&quot;</td>
<td>About an hour ago</td>
<td>Up About an hour</td>
</tr>
<tr>
<td>127.0.0.1:32768-&gt;6443/tcp</td>
<td>kind-control-plane</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8a2ad04669f8</td>
<td>kindest/node:v1.18.0</td>
<td>&quot;/usr/local/bin/entr...&quot;</td>
<td>About an hour ago</td>
<td>Up About an hour</td>
</tr>
</tbody>
</table>
Normal  NodeHasSufficientMemory  12m (x5 over 12m)  kubelet, kind-control-plane  Node kind-control-plane status
Normal  NodeHasNoDiskPressure  12m (x5 over 12m)  kubelet, kind-control-plane  Node kind-control-plane status
Normal  NodeHasNoDiskPressure  12m (x5 over 12m)  kubelet, kind-control-plane  Node kind-control-plane status
Normal  NodeHasSufficientPID  12m (x5 over 12m)  kubelet, kind-control-plane  Node kind-control-plane status
Normal  Starting  12m  kubelet, kind-control-plane  Starting kubelet.
Normal  NodeHasSufficientMemory  12m (x5 over 12m)  kubelet, kind-control-plane  Node kind-control-plane status
Normal  NodeHasNoDiskPressure  12m (x5 over 12m)  kubelet, kind-control-plane  Node kind-control-plane status
Normal  NodeHasSufficientPID  12m (x5 over 12m)  kubelet, kind-control-plane  Node kind-control-plane status
Normal  NodeAllocatableEnforced  12m  kubelet, kind-control-plane  Updated Node Allocatable limit across pods
Warning  readOnlySysFS  11m  kube-proxy, kind-control-plane  CRI error: /sys is read-only: cannot modify cgroup limits, problems may arise later (If running Docker, see docker issue #24000)
Normal  Starting  11m  kube-proxy, kind-control-plane  Starting kube-proxy.
Normal  NodeReady  11m  kubelet, kind-control-plane  Node kind-control-plane status
Normal  NodeReady  11m  kubelet, kind-control-plane  Node kind-control-plane status
ara.pulido@COMP-C02YX0HHLVDP:~/sea_local_clusters/kind$ kubectl get pods
NAME                      READY STATUS    RESTARTS AGE
advertisements-655d5d4cdd-cpgw9  1/1 Running   0       10m
db-6ddf48c95-c6g2r          1/1 Running   0       10m
discounts-dfd9bbdbb-hyknmm    1/1 Running   0       10m
frontend-8695cb4c77-zfvs4     1/1 Running   0       10m
ara.pulido@COMP-C02YX0HHLVDP:~/sea_local_clusters/kind$
SWEET SUNDAY!

Enter coupon code ‘OFF’ at checkout for a $300.10 discount

COOL HATS

Shop by Categories

- Bags
- Mugs
- Clothing

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>spree</td>
<td>spree</td>
<td>spree</td>
<td>spree</td>
</tr>
<tr>
<td>NO IMAGE AVAILABLE</td>
<td>NO IMAGE AVAILABLE</td>
<td>NO IMAGE AVAILABLE</td>
<td>NO IMAGE AVAILABLE</td>
</tr>
<tr>
<td>Datadog Tote</td>
<td>Datadog Bag</td>
<td>Datadog Baseball Jersey</td>
<td>Datadog Jr. Spaghetti</td>
</tr>
</tbody>
</table>
microk8s

- K8s distribution aimed for developers & IoT devices
- Made by Canonical and packaged as a snap
- Better tested in Ubuntu than other distributions
- flanneld CNI by default
- Multi-node clusters
- Add-ons based
microk8s
microK8s

Ubuntu OS as K8s node

Hardware

snapd

Ubuntu OS as K8s node

HyperKit

MacOS

Hardware

Ubuntu OS as K8s node

Hyper-V

Windows

Hardware
Instant Ubuntu VMs

A mini-cloud on your Mac or Windows workstation.
multipass launch -n control-plane --mem 4G --disk 40G

Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-96-generic x86_64)

* Documentation:  https://help.ubuntu.com
* Management:  https://landscape.canonical.com
* Support:  https://ubuntu.com/advantage

System information as of Mon Apr 20 13:09:05 CEST 2020

System load:  0.48       Processes:      114
Usage of /:  2.5% of 38.60GB    Users logged in:   0
Memory usage: 3%     IP address for enp0s2: 192.168.64.19
Swap usage:  0%

0 packages can be updated.
0 updates are security updates.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@control-plane:~$
Launched: control-plane

Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-96-generic x86_64)

* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage

System information as of Mon Apr 20 13:09:05 CEST 2020

System load: 0.48  Processes: 114
Usage of /: 2.5% of 38.60GB  Users logged in: 0
Memory usage: 3%  IP address for enp0s2: 192.168.64.19
Swap usage: 0%

0 packages can be updated.
0 updates are security updates.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@control-plane:~$ sudo snap install microk8s --classic
2020-04-20T13:09:33+02:00 INFO Waiting for restart...
microk8s v1.18.1 from Canonical installed
ubuntu@control-plane:~$
microk8s v1.18.1 from Canonical installed

ubuntu@control-plane:~$ sudo microk8s status --wait-ready
microk8s is running

addons:
  cilium: disabled
dashboard: disabled
dns: disabled
fluentd: disabled
gpu: disabled
helm: disabled
helm3: disabled
ingress: disabled
istio: disabled
jaeger: disabled
knative: disabled
kubeflow: disabled
linkerd: disabled
metallb: disabled
metrics-server: disabled
prometheus: disabled
rbac: disabled
registry: disabled
storage: disabled

ubuntu@control-plane:~$ sudo microk8s.kubectl get nodes
NAME    STATUS   ROLES      AGE     VERSION
control-plane Ready <none> 11s  v1.18.1

ubuntu@control-plane:~$
ubuntu@control-plane:~$ exit
logout
ara.pulido@COMP-C02YX0HHLVD$:~/sea_local_clusters/microk8s$ multipass transfer --help
Usage: multipass transfer [options] <source> [<source> ...] <destination>
Copy files between the host and instances.

Options:
- h, --help  Display this help
- v, --verbose Increase logging verbosity, repeat up to three times for more detail

Arguments:
source One or more paths to transfer, prefixed with <name:> for paths inside the instance, or '-' for stdin
destination The destination path, prefixed with <name:> for a path inside the instance, or '-' for stdout

ara.pulido@COMP-C02YX0HHLVD$:~/sea_local_clusters/microk8s$ multipass transfer control-plane:/tmp/config
ara.pulido@COMP-C02YX0HHLVD$:~/sea_local_clusters/microk8s$ ls -la
total 4
drwxr-xr-x 3 ara.pulido staff 96 Apr 20 16:31 .
drwxr-xr-x 5 ara.pulido staff 160 Apr 17 16:53 ..
-rw-r--r-- 1 ara.pulido staff 1869 Apr 20 16:31 config
ara.pulido@COMP-C02YX0HHLVD$:~/sea_local_clusters/microk8s$ kubectl get nodes
NAME         STATUS    ROLES       AGE     VERSION
control-plane Ready <none> 3h20m v1.18.1
ara.pulido@COMP-C02YX0HHLVD$:~/sea_local_clusters/microk8s$
ubuntu@control-plane:~$ exit
logout
ara.pulido@COMP-C02Y0X0HHLVDP:~/sea_local_clusters/microk8s$ multipass transfer --help
Usage: multipass transfer [options] <source> [<source> ...] <destination>
Copy files between the host and instances.

Options:
  -h, --help      Display this help
  -v, --verbose   Increase logging verbosity, repeat up to three times for more detail

Arguments:
  source          One or more paths to transfer, prefixed with <name:> for paths inside the instance, or '-' for stdin
  destination     The destination path, prefixed with <name:> for a path inside the instance, or '-' for stdout

ara.pulido@COMP-C02Y0X0HHLVDP:~/sea_local_clusters/microk8s$ multipass transfer control-plane:/tmp/config
ara.pulido@COMP-C02Y0X0HHLVDP:~/sea_local_clusters/microk8s$ ls -la

```
total 4
drwxr-xr-x 3 ara.pulido staff 96 Apr 20 16:31 .
drwxr-xr-x 5 ara.pulido staff 160 Apr 17 16:53 ..
-rw-r--r-- 1 ara.pulido staff 1869 Apr 20 16:31 config
```
ara.pulido@COMP-C02Y0X0HHLVDP:~/sea_local_clusters/microk8s$ KUBECONFIG=./config kubectl get nodes

```
NAME    STATUS    ROLES    AGE   VERSION
control-plane Ready <none> 3h20m v1.18.1
```
ara.pulido@COMP-C02Y0X0HHLVDP:~/sea_local_clusters/microk8s$
Enabling default storage class

deployment.apps/hostpath-provisioner created
storageclass.storage.k8s.io/microk8s-hostpath created
serviceaccount/microk8s-hostpath created
clusterrole.rbac.authorization.k8s.io/microk8s-hostpath created
clusterrolebinding.rbac.authorization.k8s.io/microk8s-hostpath created

Storage will be available soon

Enabling metrics-server

clusterrolebinding.rbac.authorization.k8s.io/metrics-server:system:auth-delegator created
rolebinding.rbac.authorization.k8s.io/metrics-server:auth-reader created
apiservice.apiregistration.k8s.io/v1beta1.metrics.k8s.io created
serviceaccount/metrics-server created
configmap/metrics-server-config created
deployment.apps/metrics-server-v0.2.1 created
service/metrics-server created
clusterrole.rbac.authorization.k8s.io/system:metrics-server created
clusterrolebinding.rbac.authorization.k8s.io/system:metrics-server created
clusterrole.rbac.authorization.k8s.io/system:aggregated-metrics-reader created
clusterrolebinding.rbac.authorization.k8s.io/system:aggregate-to-all-authorities created

metrics-server enabled

ubuntu@control-plane:~$ sudo microk8s.kubectl get pods -n kube-system

NAME                               READY STATUS    RESTARTS AGE
---                                 ------ -------- -------------
coredns-588fd544bf-swmgf           1/1   Running 0 2d2h
depath-hostpath-provisioner-75f8c8f6cd-f7fsq 1/1   Running 0 23s
metrics-server-v0.2.1-95c9bd697-prlh  2/2   Running 0 20s

ubuntu@control-plane:~$
k3s / k3d
k3s

- Lightweight K8s distro aimed for developers & IoT devices
- Made by Rancher and packaged as a single binary
- flanneld CNI by default
- Multi-node clusters
- Experimental k3d for container nodes
k3s

k3s server
- kube-api-server
- kube-scheduler
- kube-proxy
- kubelet
- containerd

k3s agent
- kube-proxy
- flannel
- kubelet
- containerd

Hardware

DATADOG
Connecting to github-production-release-asset-2e65be.s3.amazonaws.com (github-production-release-asset-2e65be.s3.amazonaws.com)|52.216.132.67|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 52359168 (50M) [application/octet-stream]
Saving to: ‘k3s’

2020-04-24 14:21:41 (5.42 MB/s) - ‘k3s’ saved [52359168/52359168]

```
vagrant@controlplane:~$ ls -la
total 51168
drwxr-xr-x  5 vagrant vagrant  4096 Apr 24 14:21 .
drwxr-xr-x  4 root   root     4096 Apr 24 14:20 ...
-rw-r--r--  1 vagrant vagrant   220 Aug 13 2019 .bash_logout
-rw-r--r--  1 vagrant vagrant  3771 Aug 13 2019 .bashrc
drw-------  2 vagrant vagrant  4096 Apr 24 14:20 .cache
drw-------  3 vagrant vagrant  4096 Apr 24 14:20 .gnupg
-rw-r--r--  1 vagrant vagrant 52359168 Mar 25 16:15 k3s
-rw-r--r--  1 vagrant vagrant   807 Aug 13 2019 .profile
drw-------  2 vagrant vagrant  4096 Apr 24 14:20 .ssh
-rw-r--r--  1 vagrant vagrant  165 Apr 24 14:21 .wget-hsts
```

```
vagrant@controlplane:~$ chmod a+x k3s
vagrant@controlplane:~$ sudo mv k3s /usr/local/bin/
```
TX packets 6765 bytes 505356 (505.3 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 172.28.128.8 netmask 255.255.255.0 broadcast 172.28.128.255
inet6 fe80::a00:27ff:fedc:265a prefixlen 64 scopeid 0x20<link>
ether 08:00:27:dc:26:5a txqueuelen 1000 (Ethernet)
RX packets 12 bytes 3544 (3.5 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 13 bytes 1550 (1.5 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
inet 127.0.0.1 netmask 255.0.0.0
inet6 ::1 prefixlen 128 scopeid 0x10<host>
loop txqueuelen 1000 (Local Loopback)
RX packets 32 bytes 2870 (2.8 KB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 32 bytes 2870 (2.8 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

vagrant@controlplane:~$ sudo k3s server --flannel-iface enp0s8
INFO[0000] Preparing data dir /var/lib/rancher/k3s/data/6a3098e6644f5f0dbfe14e5e4a99bb8fddf6063063ae89ffdefd71b7de11a1f1430
Error while processing event 
"/sys/fs/cgroup/cpu,cpuacct/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": 
0x40000100 == IN_CREATE|IN_ISDIR): inotify_add_watch 
"/sys/fs/cgroup/cpu,cpuacct/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": no such file or directory

Error while processing event 
"/sys/fs/cgroup/blkio/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": 
0x40000100 == IN_CREATE|IN_ISDIR): inotify_add_watch 
"/sys/fs/cgroup/blkio/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": no such file or directory

Error while processing event 
"/sys/fs/cgroup/memory/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": 
0x40000100 == IN_CREATE|IN_ISDIR): inotify_add_watch 
"/sys/fs/cgroup/memory/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": no such file or directory

Error while processing event 
"/sys/fs/cgroup/devices/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": 
0x40000100 == IN_CREATE|IN_ISDIR): inotify_add_watch 
"/sys/fs/cgroup/devices/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": no such file or directory

Error while processing event 
"/sys/fs/cgroup/pids/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": 
0x40000100 == IN_CREATE|IN_ISDIR): inotify_add_watch 
"/sys/fs/cgroup/pids/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": no such file or directory

2139 watcher.go:87] Error while processing event 
"/sys/fs/cgroup/pids/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": 
0x40000100 == IN_CREATE|IN_ISDIR): inotify_add_watch 
"/sys/fs/cgroup/pids/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": no such file or directory

2139 watcher.go:87] Error while processing event 
"/sys/fs/cgroup/pids/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": 
0x40000100 == IN_CREATE|IN_ISDIR): inotify_add_watch 
"/sys/fs/cgroup/pids/system.slice/run-r668747636c394d37a6d3db78b65ecf2b.scope": no such file or directory

2139 garbagecollector.go:639] failed to discover some groups: map[metrics.k8s.io/v1beta1:the server is currently unable to handle the request]


10424 14:24:37.282032 2139 shared_informer.go:204] Caches are synced for garbage collector
<table>
<thead>
<tr>
<th>NAME</th>
<th>STATUS</th>
<th>ROLES</th>
<th>AGE</th>
<th>VERSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>controlplane</td>
<td>Ready</td>
<td>master</td>
<td>40s</td>
<td>v1.17.4+k3s1</td>
</tr>
</tbody>
</table>

vagrant@controlplane:~$
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 478 bytes 112765 (112.7 KB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

vetha40ba89c: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1450
     inet6 fe80::ec18:59ff:fe3a:b4fc prefixlen 64 scopeid 0x20<link>
          ether ee:18:59:3a:b4:fc txqueuelen 0 (Ethernet)
     RX packets 12 bytes 908 (908.0 B)
     RX errors 0 dropped 0 overruns 0 frame 0
     TX packets 42 bytes 3284 (3.2 KB)
     TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

vethd1126fb7: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1450
     inet6 fe80::582d:83ff:fe00:536f prefixlen 64 scopeid 0x20<link>
          ether 5a:2d:83:60:53:6f txqueuelen 0 (Ethernet)
     RX packets 570 bytes 447364 (447.3 KB)
     RX errors 0 dropped 0 overruns 0 frame 0
     TX packets 523 bytes 219885 (219.8 KB)
     TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

vagrant@controlplane:~$ sudo k3s kubectl get nodes
NAME      STATUS    ROLES     AGE       VERSION
node      Ready    <none>    38s       v1.17.4+k3s1
controlplane Ready    master 6m6s    v1.17.4+k3s1
vagrant@controlplane:~$
**k3d**

- Easy for devs wrapper to launch k3s clusters in Docker
- Nodes will be run as docker containers
- Installation script detects your OS and installs a k3d binary
curl -s https://raw.githubusercontent.com/rancher/k3d/master/install.sh | bash

Preparing to install k3d into /usr/local/bin
Password:

k3d installed into /usr/local/bin/k3d

Run 'k3d --help' to see what you can do with it.
```bash
kubectl cluster-info

ara.pulido@COMP-C02YX0HHLVDP:~$ k3d list

+----------+-------------+----------+-------+--------+
| NAME     | IMAGE       | STATUS   | WORKERS |
+----------+-------------+----------+-------+--------+
| k3s-default | docker.io/rancher/k3s:v1.17.3-k3s1 | running | 0/0   |        |
+----------+-------------+----------+-------+--------+

ara.pulido@COMP-C02YX0HHLVDP:~$ docker ps

<table>
<thead>
<tr>
<th>CONTAINER ID</th>
<th>IMAGE</th>
<th>COMMAND</th>
<th>CREATED</th>
<th>STATUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>92096295d0db</td>
<td>rancher/k3s:v1.17.3-k3s1</td>
<td>/bin/k3s server --h...</td>
<td>16 seconds ago</td>
<td>Up 15 seconds</td>
</tr>
</tbody>
</table>

ara.pulido@COMP-C02YX0HHLVDP:~$ export KUBECONFIG="$(k3d get-kubeconfig --name=k3s-default)"

ara.pulido@COMP-C02YX0HHLVDP:~$ k get nodes

<table>
<thead>
<tr>
<th>NAME</th>
<th>STATUS</th>
<th>ROLES</th>
<th>AGE</th>
<th>VERSION</th>
</tr>
</thead>
<tbody>
<tr>
<td>k3d-k3s-default-server</td>
<td>Ready</td>
<td>master</td>
<td>28s</td>
<td>v1.17.3+k3s1</td>
</tr>
</tbody>
</table>

ara.pulido@COMP-C02YX0HHLVDP:~$ k get pods -n kube-system

<table>
<thead>
<tr>
<th>NAME</th>
<th>READY</th>
<th>STATUS</th>
<th>RESTARTS</th>
<th>AGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>local-path-provisioner-58fb86bdfd-5mknk5</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>22s</td>
</tr>
<tr>
<td>metrics-server-6d6d4c7b5-rt776</td>
<td>1/1</td>
<td>Running</td>
<td>0</td>
<td>22s</td>
</tr>
<tr>
<td>coredns-d798c9dd-pfgxg</td>
<td>0/1</td>
<td>Running</td>
<td>0</td>
<td>22s</td>
</tr>
<tr>
<td>traefik-6787cdddb4b-w8pq9</td>
<td>0/1</td>
<td>ContainerCreating</td>
<td>0</td>
<td>2s</td>
</tr>
<tr>
<td>svclb-traefik-w6b6z</td>
<td>0/2</td>
<td>ContainerCreating</td>
<td>0</td>
<td>2s</td>
</tr>
<tr>
<td>helm-install-traefik-hk49z</td>
<td>0/1</td>
<td>Completed</td>
<td>0</td>
<td>22s</td>
</tr>
</tbody>
</table>
``
per variable)
--volume source:destination, -v source:destination Mount one or more volumes into every created node
(Docker notation: source:destination)
--k3s https://<host>[:<port>] Add a k3d node to a non-k3d k3s cluster (specify k3s server URL like this https://<host>[:<port>]) [requires k3s-secret or k3s-token]
--k3s-secret value, -s value Specify k3s cluster secret (or use --k3s-token to use a node token)
--k3s-token value, -t value Specify k3s node token (or use --k3s-secret to use a cluster secret)[overrides k3s-secret]

ara.pulido@COMP-C02YX0HHLVDP:~/k3s$ k3d add-node -n k3s-default
INFO[0000] Adding 1 agent-nodes to k3d cluster k3s-default...
INFO[0000] Created agent-node with ID 4fbe10471cde29c0854f812d47de38307cc3ed5985d26f3f68598b8e9394d4e8
ara.pulido@COMP-C02YX0HHLVDP:~/k3s$ k get nodes
NAME STATUS ROLES AGE VERSION
k3d-k3s-default-server Ready master 14m v1.17.3+k3s1
k3d-k3s-default-worker-1 NotReady <none> 6s v1.17.3+k3s1
ara.pulido@COMP-C02YX0HHLVDP:~/k3s$ docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS
4fbe10471cde rancher/k3s:v1.17.3-k3s1 /bin/k3s agent 14 seconds ago Up 14 seconds
k3d-k3s-default-worker-1
92096295d0db rancher/k3s:v1.17.3-k3s1 /bin/k3s server --h... 15 minutes ago Up 15 minutes
ara.pulido@COMP-C02YX0HHLVDP:~/k3s$
firekube
GitOps
firekube

- GitOps ready Kubernetes cluster
- Using ignite and firecraker for lightweight VMs (only Linux)
- Using docker in docker (footloose) if not in Linux
- Based on kubeadm
Firekube

Firekube is a Kubernetes cluster working on top of ignite and firecracker. Firekube clusters are operated with GitOps.

*ignite* and *firecracker* only work on Linux as they need KVM. Fortunately we will also work on macOS using footloose: the Kubernetes nodes are then running inside containers.

Creating a Firekube cluster

**Prerequisites**: docker, git, kubectl 1.14+

1. Fork this repository.
2. Clone your fork and `cd` into it. Use the `SSH` git URL as the script will push an initial commit to your fork:

```bash
export user="" # Your GitHub handle or org

git clone git@github.com:$user/wks-quickstart-firekube.git
cd wks-quickstart-firekube
```
Creating SSH key
Creating virtual machines
INFO[0000] Docker Image: quay.io/footloose/centos7:0.6.0 present locally
INFO[0000] Creating machine: firekube-node0 ...
INFO[0000] Creating machine: firekube-node1 ...
Creating Cluster API manifests
Updating container images and git parameters
INFO[0000] wksctl version 0.8.2-beta.1 is available; please update at https://github.com/weaveworks/wksctl/releases/tag/v0.8.2-beta.1
Pushing initial cluster configuration
[master 66c3cb7] Initial cluster configuration
5 files changed, 69 insertions(+), 3 deletions(-)
create mode 100644 footloose.yaml
create mode 100644 machines.yaml
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 8 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 1.06 KiB | 1.06 MiB/s, done.
Total 7 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
To github.com:arapulido/wks-quickstart-firekube.git
  2628729..66c3cb7 HEAD -> master
Installing Kubernetes cluster
INFO[2020-04-28T14:43:33+02:00] Finishing
INFO[2020-04-28T14:43:33+02:00] Starting
INFO[2020-04-28T14:43:33+02:00] Finishing

INFO[0000] wksctl version 0.8.2-beta.1 is available; please update at https://github.com/weaveworks/wksctl/releases/tag/v0.8.2-beta.1
To use kubectl with the example cluster, enter:

```bash
$ export KUBECONFIG=/Users/ara.pulido/.wks/weavek8sops/example/kubeconfig
```

```
ara.pulido@COMP-C02YX0HHLVDP:~/firekube/wks-quickstart-firekube$ export KUBECONFIG=/Users/ara.pulido/.wks/weavek8sops/example/kubeconfig
```

```
ara.pulido@COMP-C02YX0HHLVDP:~/firekube/wks-quickstart-firekube$ kubectl get nodes
NAME     STATUS   ROLES     AGE     VERSION
node0    Ready    master    40s     v1.14.1
```

```
ara.pulido@COMP-C02YX0HHLVDP:~/firekube/wks-quickstart-firekube$ docker ps
CONTAINER ID   IMAGE                  COMMAND                  CREATED            STATUS                      PORTS                NAMES
016676f7865e  quay.io/footloose/centos7:0.6.0  "/sbin/init"   3 minutes ago   Up 3 minutes 0.0.0.0:2223->22/tcp, 0.0.0.0:6444->6444/tcp, 0.0.0.0:30081->30080/tcp, 0.0.0.0:30444->30443/tcp  firekube-node1
d5acfc8b9a1d  quay.io/footloose/centos7:0.6.0  "/sbin/init"   3 minutes ago   Up 3 minutes 0.0.0.0:6443->6443/tcp, 0.0.0.0:30080->30080/tcp, 0.0.0.0:30443->30443/tcp, 0.0.0.0:2222->22/tcp  firekube-node0
```
<table>
<thead>
<tr>
<th>NAME</th>
<th>CPU %</th>
<th>RSS MEMORY</th>
<th>TX</th>
<th>RX</th>
<th>STATUS</th>
<th>START</th>
</tr>
</thead>
<tbody>
<tr>
<td>weave_weave-net-6fqvf</td>
<td>0 %</td>
<td>28 MB</td>
<td>80.8 KB</td>
<td>68.1 KB</td>
<td>up</td>
<td>21 minutes ago</td>
</tr>
<tr>
<td>weave_weave-net-zfm6z</td>
<td>&lt; 1 %</td>
<td>27 MB</td>
<td>22.9 KB</td>
<td>17.0 KB</td>
<td>up</td>
<td>23 minutes ago</td>
</tr>
<tr>
<td>weave-npc_weave-net-6fqvf</td>
<td>0 %</td>
<td>23 MB</td>
<td>80.8 KB</td>
<td>68.1 KB</td>
<td>up</td>
<td>21 minutes ago</td>
</tr>
<tr>
<td>weave-npc_weave-net-zfm6z</td>
<td>0 %</td>
<td>23 MB</td>
<td>22.9 KB</td>
<td>17.0 KB</td>
<td>up</td>
<td>23 minutes ago</td>
</tr>
<tr>
<td>flux_flux-7c659bdf8b-56mls</td>
<td>0 %</td>
<td>22 MB</td>
<td>0 B</td>
<td>0 B</td>
<td>up</td>
<td>23 minutes ago</td>
</tr>
<tr>
<td>controller_wks-controller-5bbcf574659-6hzv</td>
<td>&lt; 1 %</td>
<td>6 MB</td>
<td>0 B</td>
<td>0 B</td>
<td>up</td>
<td>23 minutes ago</td>
</tr>
<tr>
<td>memcached_memcached-6bc6886f9f-hvkp</td>
<td>0 %</td>
<td>4 MB</td>
<td>2.40 KB</td>
<td>494 B</td>
<td>up</td>
<td>23 minutes ago</td>
</tr>
</tbody>
</table>
ts=2020-04-28T12:50:03.268273054Z caller=sync.go:545 method=Sync cmd="kubectl apply -f -" took=721.293998ms err=null output="namespace/fake-traffic created
namespace/weavek8sops unchanged
service/advertisements created
service/db created
service/discounts created
clusterrole.rbac.authorization.k8s.io/flux unchanged
serviceaccount/flux unchanged
service/frontend created
service/memcached unchanged
serviceaccount/postgres created
persistentvolume/task-pv-volume created
secret/db-password created
configmap/docker unchanged
clusterrolebinding.rbac.authorization.k8s.io/flux unchanged
configmap/repo unchanged
persistentvolumeclaim/task-pvc-volume created

Takeaways
- Local clusters are not production
- Know how your local cluster works
- Docker in Docker is a good compromise
Thank you!

Questions?