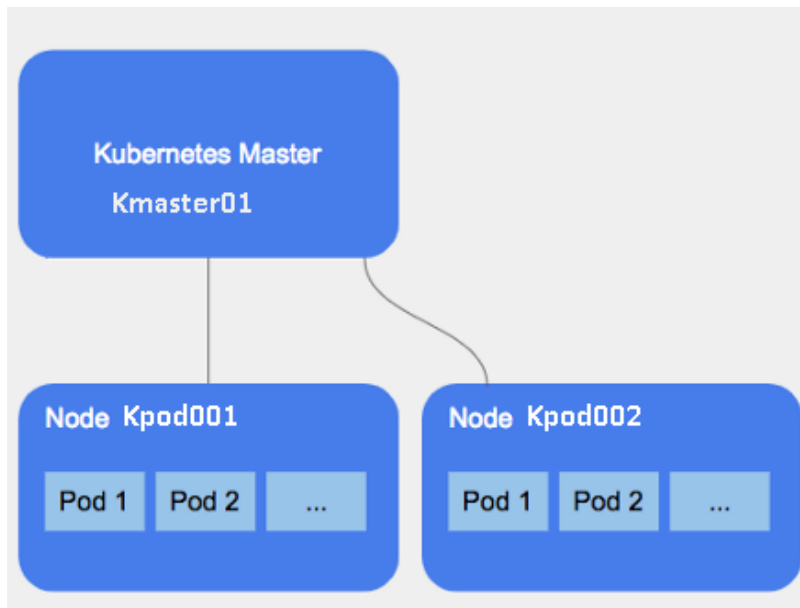


Cluster Kubernetes

Arquitectura actual

- **Kmaster01 (master)**
 - **Kpod001 (nodo o worker)**
 - **Kpod002 (nodo o worker)**



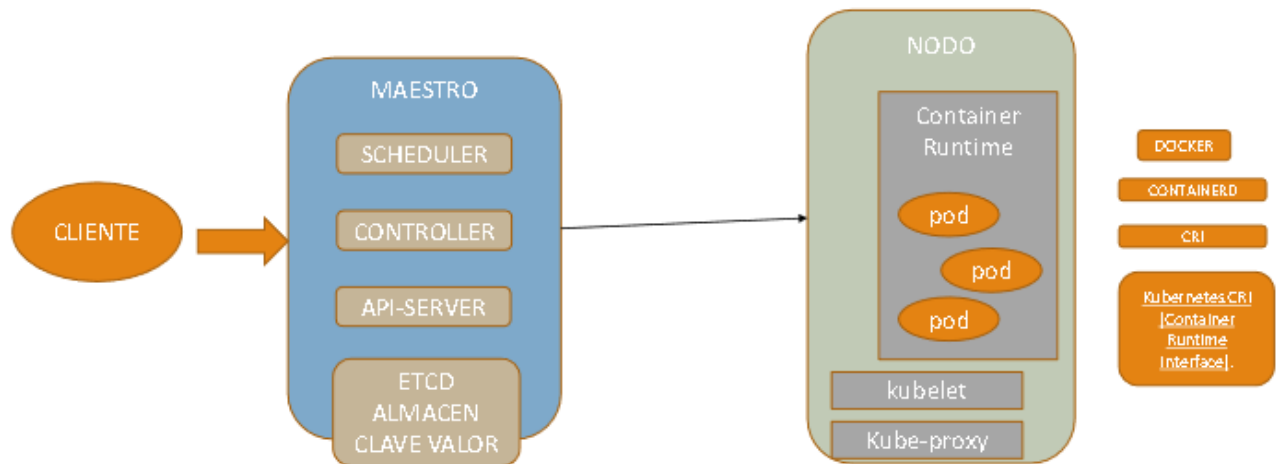
Objetos de K8S

Para entender lo que está corriendo dentro de un cluster de k8s hay que entender los objetos que contiene el mismo:

Entre ellos tenemos, en orden Jerarquico:

- **Proyectos**: objeto de rancher, que esta por encima de los namespace. Basicamente es una forma de segmentar namespaces.
- **Namespaces**: contiene diferentes aplicaciones. La idea de los namespaces es de segmentar aplicaciones o ambientes. Contiene los deplolements, statefulSet, ReplicaSet y DaemonSet
- **Deployments**: objetos mas comunes para tratar microservicios. Aca se encuentran los replicaSet y los pods. Generalmente se usa UNO por micro
- **DaemonSet**: Son como los deployments, pero se aseguran de tener solo un pod en cada nodo
- **ReplicaSet**: los replicaSet son los encargados de lograr el estado deseado de cantidad de pods que impone el deployments
- **Pods**: unidad mínima en k8s donde corren los contenedores.

En resumen:



Algunos conceptos que salen de aca:

Maestro:

- **Scheduler:** Es el encargado de la planificación de cuando corra todo o de encolar si fuera necesario
- **Controller:** es el que controla el estado general del master y la comunicación con los nodos
- **Api-Server:** api de comunicación con el resto de la arquitectura
- **ETCD:** base de datos de K8s

Nodo:

- **Pod**
 - **Kubelet:** servicio de K8s que corre en un nodo (esto pone en Ready al nodo)
 - **Kube-proxy:** es el encargado de permitir la conexión con el master (si hay algún problema de comunicación entre el nodo master y worker)

Todo el cluster es esto:

```
[root@kmaster01 ~]# kubectl get pods --namespace kube-system
NAME                                READY STATUS RESTARTS AGE
coredns-558bd4d5db-29ns6            1/1   Running 1    284d
coredns-558bd4d5db-757sw            1/1   Running 1    284d
etcd-kmaster01                      1/1   Running 3    284d
kube-apiserver-kmaster01            1/1   Running 0    33d
kube-controller-manager-kmaster01  1/1   Running 1    33d
kube-proxy-5d8v2                    1/1   Running 5    284d
kube-proxy-hk695                    1/1   Running 10   284d
kube-proxy-mrjvf                    1/1   Running 3    284d
kube-scheduler-kmaster01            1/1   Running 1    33d
weave-net-j2vqj                     2/2   Running 3    284d
weave-net-nslwj                     2/2   Running 10   284d
```



weave-net-qkbbkq

2/2

Running 19 284d

Estados de los nodos

- **Ready**—able to run pods.
- **NotReady**—not operating due to a problem, and cannot run pods.
- **SchedulingDisabled**—the node is healthy but has been marked by the cluster as not schedulable.
- **Unknown**—if the node controller cannot communicate with the node, it waits a default of 40 seconds, and then sets the node status to unknown.

Instalacion OnPremise de Kubernetes

OTRA OPCION: <https://www.keitaro.com/2021/09/03/setting-up-a-kubernetes-on-premise-cluster-with-kubeadm/>

PRUEBAS POR PLAY WITH K8S

```
kubeadm init --apiserver-advertise-address $(hostname -i) --pod-network-cidr 10.5.0.0/16
```

```
kubectl apply -f https://raw.githubusercontent.com/cloudnativelabs/kube-router/master/daemonset/kubeadm-kuberouter.yaml
```

```
kubectl apply -f https://raw.githubusercontent.com/kubernetes/website/master/content/en/examples/application/nginx-app.yaml
```

```
kubeadm join 192.168.0.33:6443 --token c5yyk2.1v9lf7jj2k7gsyx4 \
--discovery-token-ca-cert-hash
sha256:5d0644e36c1f8bcd78fc6c13f6419474896bfd800998b47435cf16da9082440c
```

Primero bajamos la version prebuildead a segun la documentacion:

<https://kubernetes.io/es/docs/setup/release/building-from-source/>

En el repositorio, nos vamos a la version mas reciente

<https://github.com/kubernetes/kubernetes/blob/master/CHANGELOG/CHANGELOG-1.22.md>



Bajamos el binario del server

Server Binaries

filename	sha512 hash
kubernetes-server-linux-amd64.tar.gz	9372ea845dafc41949bbfa4c45c197f282b61950eefa19bb7486c58991ce76c7f7eeb1e97a3

Node Binaries

filename	sha512 hash
kubernetes-node-linux-amd64.tar.gz	4ae063266207869ba4a6f9290b75785a4e555f3c8ebc7856fcb57b14a3b29993afc106e68b

y tambien tenemos los nodos

Ahora debemos instalar la herramienta de contenedores, en este caso instalamos docker

VER NOTA "Instalacion de Docker dependiendo la distribucion"

Instalacion a traves de KUBEADM

```
cat <<EOF | sudo tee /etc/yum.repos.d/kubernetes.repo
[kubernetes]
name=Kubernetes
baseurl=https://packages.cloud.google.com/yum/repos/kubernetes-el7-`$basearch`
enabled=1
gpgcheck=1
repo_gpgcheck=1
gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg
https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
exclude=kubelet kubeadm kubectl
EOF
```

Set SELinux in permissive mode (effectively disabling it)

```
sudo setenforce 0
```

```
sudo sed -i 's/^SELINUX=enforcing$/SELINUX=permissive/' /etc/selinux/config
```

```
sudo yum install -y kubelet kubeadm kubectl --disableexcludes=kubernetes
```



```
sudo systemctl enable --now kubelet
```

Configuring a cgroup driver

El primer error cuando le das la inicializacion kubeadm init, nos da

```
[root@kmaster01 ~]# kubeadm init --config kubeadm-config.yaml
W0727 07:50:19.599779 123635 kubelet.go:215] detected "cgroupfs" as the Docker cgroup
driver, the provided value "systemd" in "KubeletConfiguration" will be overridden
[init] Using Kubernetes version: v1.21.0
[preflight] Running pre-flight checks
[WARNING HTTPProxy]: Connection to "https://172.16.8.200" uses proxy
"http://128.2.0.7:3128". If that is not intended, adjust your proxy settings
[WARNING HTTPProxyCIDR]: connection to "10.96.0.0/12" uses proxy
"http://128.2.0.7:3128". This may lead to malfunctioning cluster setup. Make sure that Pod
and Services IP ranges specified correctly as exceptions in proxy configuration
[WARNING IsDockerSystemdCheck]: detected "cgroupfs" as the Docker cgroup driver.
The recommended driver is "systemd". Please follow the guide at
https://kubernetes.io/docs/setup/cri/
[WARNING FileExisting-tc]: tc not found in system path
[WARNING Service-Kubelet]: kubelet service is not enabled, please run 'systemctl
enable kubelet.service'
error execution phase preflight: [preflight] Some fatal errors occurred:
[ERROR Swap]: running with swap on is not supported. Please disable swap
[preflight] If you know what you are doing, you can make a check non-fatal with `--ignore-
preflight-errors=...`
```

To see the stack trace of this error execute with --v=5 or higher

El archivo:

```
[root@kmaster01 ~]# cat kubeadm-config.yaml
# kubeadm-config.yaml
kind: ClusterConfiguration
apiVersion: kubeadm.k8s.io/v1beta2
kubernetesVersion: v1.21.0
---
kind: KubeletConfiguration
apiVersion: kubelet.config.k8s.io/v1beta1
cgroupDriver: systemd
```

Solucion

```
kubeadm reset
```

```
echo 'Environment="KUBELET_EXTRA_ARGS=--fail-swap-on=false"' >>
/etc/systemd/system/kubelet.service.d/10-kubeadm.conf
```

```
systemctl daemon-reload
```

```
systemctl restart kubelet
```



Cambiar el Cgroups en docker

```
cat > /etc/docker/daemon.json <<EOF
{
  "exec-opts": ["native.cgroupdriver=systemd"],
  "log-driver": "json-file",
  "log-opts": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
EOF
```

```
systemctl restart docker
```

```
[root@kpod001 ~]# docker info | grep Cgroup
Cgroup Driver: systemd
Cgroup Version: 1
```

Problemas al levantar servicios que quedaron mal

```
[preflight] Some fatal errors occurred:
  [ERROR FileAvailable--etc-kubernetes-kubelet.conf]: /etc/kubernetes/kubelet.conf
already exists
  [ERROR Port-10250]: Port 10250 is in use
  [ERROR FileAvailable--etc-kubernetes-pki-ca.crt]: /etc/kubernetes/pki/ca.crt already
exists
```

Solucion

```
[root@kpod001 ~]# kubeadm reset
```

Instalacion de un NODO y agregarlo al cluster

```
[root@kmaster01 ~]# cat /etc/hosts
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
```

```
[root@kmaster01 ~]# ping kpod002
PING kpod002.redcoto.com.ar (172.16.8.202) 56(84) bytes of data.
64 bytes from 172.16.8.202 (172.16.8.202): icmp_seq=1 ttl=64 time=0.404 ms
^C
--- kpod002.redcoto.com.ar ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.404/0.404/0.404/0.000 ms
[root@kmaster01 ~]# ssh kpod002
```



kubernetes

```
[root@kpod002 ~]# vi
```

```
/etc/selinux/config
[root@kpod002 ~]# cat <<EOF | sudo tee /etc/yum.repos.d/kubernetes.repo
> [kubernetes]
> name=Kubernetes
> baseurl=https://packages.cloud.google.com/yum/repos/kubernetes-el7-\$basearch
> enabled=1
> gpgcheck=1
> repo_gpgcheck=1
> gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg
https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
> exclude=kubelet kubeadm kubectl
> EOF
[kubernetes]
name=Kubernetes
baseurl=https://packages.cloud.google.com/yum/repos/kubernetes-el7-\$basearch
enabled=1
gpgcheck=1
repo_gpgcheck=1
gpgkey=https://packages.cloud.google.com/yum/doc/yum-key.gpg
https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
exclude=kubelet kubeadm kubectl
```

```
[root@kpod002 ~]# yum install -y kubelet kubeadm kubectl --disableexcludes=kubernetes
```

```
Kubernetes                                354 B/s |
844 B   00:02
Kubernetes                                5.7 kB/s | 3.4
kB   00:00
Importing GPG key 0x307EA071:
Userid   : "Rapture Automatic Signing Key (cloud-rapture-signing-key-2021-03-01-08_01_09.pub)"
Fingerprint: 7F92 E05B 3109 3BEF 5A3C 2D38 FEEA 9169 307E A071
From      : https://packages.cloud.google.com/yum/doc/yum-key.gpg
Importing GPG key 0x836F4BEB:
Userid   : "gLinux Rapture Automatic Signing Key (//depot/google3/production/borg/cloud-rapture/keys/cloud-rapture-pubkeys/cloud-rapture-signing-key-2020-12-03-16_08_05.pub)
<glinux-team@google.com>"
Fingerprint: 59FE 0256 8272 69DC 8157 8F92 8B57 C5C2 836F 4BEB
From      : https://packages.cloud.google.com/yum/doc/yum-key.gpg
Kubernetes                                1.5 kB/s | 975 B   00:00
Importing GPG key 0x3E1BA8D5:
Userid   : "Google Cloud Packages RPM Signing Key <gc-team@google.com>"
Fingerprint: 3749 E1BA 95A8 6CE0 5454 6ED2 F09C 394C 3E1B A8D5
From      : https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
Kubernetes                                28 kB/s | 126 kB   00:04
Oracle Linux 8 BaseOS Latest (x86_64)      6.6 MB/s | 39
MB   00:05
```



Oracle Linux 8 Application Stream

(x86_64) 7.4 MB/s | 27 MB 00:03
Latest Unbreakable Enterprise Kernel Release 6 for Oracle Linux 8 (x86_64) 7.3 MB/s
| 25 MB 00:03

Dependencies resolved.

Package	Architecture	Version	Repository
---------	--------------	---------	------------

Installing:

kubeadm	x86_64	1.22.0-	
0	kubernetes	9.3 M	
kubect	x86_64	1.22.0-	
0	kubernetes	9.6 M	
kubelet	x86_64	1.22.0-	
0	kubernetes	23 M	

Installing dependencies:

conntrack-tools	x86_64	1.4.4-	
10.el8	ol8_baseos_latest	205 k	
cri-tools	x86_64	1.13.0-	
0	kubernetes	5.1 M	
kubernetes-cni	x86_64	0.8.7-	
0	kubernetes	19 M	
libnetfilter_cthelper	x86_64	1.0.0-	
15.el8	ol8_baseos_latest	24 k	
libnetfilter_cttimeout	x86_64	1.0.0-	
11.el8	ol8_baseos_latest	24 k	
libnetfilter_queue	x86_64	1.0.4-	
3.el8	ol8_baseos_latest	31 k	
socat	x86_64	1.7.3.3-	
2.el8	ol8_appstream	303 k	

Transaction Summary

Install 10 Packages

Total download size: 67 M

Installed size: 313 M

Downloading Packages:



(1/10):

14bfe6e75a9efc8eca3f638eb22c7e2ce759c67f95b43b16fae4ebabde1549f3-cri-tools-1.13.0-0.x86_64.rpm 2.0 MB/s | 5.1 MB 00:02

(2/10): fd7e970fe8ce2cee319e2731660cf814a9be4d839a365c427665f02918ae2926-kubectl-1.22.0-0.x86_64.rpm 2.1 MB/s | 9.6 MB 00:04

(3/10): f617750909f36af30362e3347c40c1e4aaa8c1786af970820e31683b2cbd402f-kubeadm-1.22.0-0.x86_64.rpm 2.0 MB/s | 9.3 MB 00:04

(4/10): contrack-tools-1.4.4-10.el8.x86_64.rpm 968 kB/s | 205 kB 00:00

(5/10): libnetfilter_cthelper-1.0.0-15.el8.x86_64.rpm 511 kB/s | 24 kB 00:00

(6/10): libnetfilter_cttimeout-1.0.0-11.el8.x86_64.rpm 737 kB/s | 24 kB 00:00

(7/10): libnetfilter_queue-1.0.4-3.el8.x86_64.rpm 914 kB/s | 31 kB 00:00

(8/10): socat-1.7.3.3-2.el8.x86_64.rpm 2.9 MB/s | 303 kB 00:00

(9/10): f0fa64eb57c216036ea5c6d33ae38511424c4966a4054d82528252996fac9499-kubelet-1.22.0-0.x86_64.rpm 3.8 MB/s | 23 MB 00:06

(10/10): db7cb5cb0b3f6875f54d10f02e625573988e3e91fd4fc5eef0b1876bb18604ad-kubernetes-cni-0.8.7-0.x86_64.rpm 3.1 MB/s | 19 MB 00:06

Total 6.2 MB/s
| 67 MB 00:10

warning: /var/cache/dnf/kubernetes-33343725abd9cbdc/packages/14bfe6e75a9efc8eca3f638eb22c7e2ce759c67f95b43b16fae4ebabde1549f3-cri-tools-1.13.0-0.x86_64.rpm: Header V4 RSA/SHA512 Signature, key ID 3e1ba8d5: NOKEY

Kubernetes 5.0 kB/s |
3.4 kB 00:00

Importing GPG key 0x307EA071:

Userid : "Rapture Automatic Signing Key (cloud-rapture-signing-key-2021-03-01-08_01_09.pub)"

Fingerprint: 7F92 E05B 3109 3BEF 5A3C 2D38 FEEA 9169 307E A071

From : <https://packages.cloud.google.com/yum/doc/yum-key.gpg>

Key imported successfully

Importing GPG key 0x836F4BEB:

Userid : "gLinux Rapture Automatic Signing Key (/depot/google3/production/borg/cloud-rapture/keys/cloud-rapture-pubkeys/cloud-rapture-signing-key-2020-12-03-16_08_05.pub) <glinux-team@google.com>"

Fingerprint: 59FE 0256 8272 69DC 8157 8F92 8B57 C5C2 836F 4BEB

From : <https://packages.cloud.google.com/yum/doc/yum-key.gpg>

Key imported successfully



```
Kubernetes 1.7 kB/s |
975 B 00:00
Importing GPG key 0x3E1BA8D5:
Userid : "Google Cloud Packages RPM Signing Key <gc-team@google.com>"
Fingerprint: 3749 E1BA 95A8 6CE0 5454 6ED2 F09C 394C 3E1B A8D5
From : https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
Key imported successfully
warning: /var/cache/dnf/ol8_baseos_latest-e4c6155830ad002c/packages/conntrack-tools-
1.4.4-10.el8.x86_64.rpm: Header V3 RSA/SHA256 Signature, key ID ad986da3: NOKEY
Oracle Linux 8 BaseOS Latest
(x86_64) 62 kB/s | 3.1 kB 00:00
Importing GPG key 0xAD986DA3:
Userid : "Oracle OSS group (Open Source Software group) <build@oss.oracle.com>"
Fingerprint: 76FD 3DB1 3AB6 7410 B89D B10E 8256 2EA9 AD98 6DA3
From : /etc/pki/rpm-gpg/RPM-GPG-KEY-oracle
Key imported successfully
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
Preparing :
1/1
Installing : socat-1.7.3.3-
2.el8.x86_64 1/10
Installing : libnetfilter_queue-1.0.4-
3.el8.x86_64 2/10
Running scriptlet: libnetfilter_queue-1.0.4-
3.el8.x86_64 2/10
Installing : libnetfilter_cttimeout-1.0.0-
11.el8.x86_64 3/10
Running scriptlet: libnetfilter_cttimeout-1.0.0-
11.el8.x86_64 3/10
Installing : libnetfilter_cthelper-1.0.0-
15.el8.x86_64 4/10
Running scriptlet: libnetfilter_cthelper-1.0.0-
15.el8.x86_64 4/10
Installing : conntrack-tools-1.4.4-
10.el8.x86_64 5/10
Running scriptlet: conntrack-tools-1.4.4-
10.el8.x86_64 5/10
Installing : kubernetes-cni-0.8.7-
0.x86_64 6/10
Installing : kubelet-1.22.0-
0.x86_64 7/10
```



Installing : kubectl-1.22.0-

0.x86_64	8/10
Installing : cri-tools-1.13.0-	
0.x86_64	9/10
Installing : kubeadm-1.22.0-	
0.x86_64	10/10
Running scriptlet: kubeadm-1.22.0-	
0.x86_64	10/10
Verifying : cri-tools-1.13.0-	
0.x86_64	1/10
Verifying : kubeadm-1.22.0-	
0.x86_64	2/10
Verifying : kubectl-1.22.0-	
0.x86_64	3/10
Verifying : kubelet-1.22.0-	
0.x86_64	4/10
Verifying : kubernetes-cni-0.8.7-	
0.x86_64	5/10
Verifying : conntrack-tools-1.4.4-	
10.el8.x86_64	6/10
Verifying : libnetfilter_cthelper-1.0.0-	
15.el8.x86_64	7/10
Verifying : libnetfilter_cttimeout-1.0.0-	
11.el8.x86_64	8/10
Verifying : libnetfilter_queue-1.0.4-	
3.el8.x86_64	9/10
Verifying : socat-1.7.3.3-	
2.el8.x86_64	10/10

Installed:

conntrack-tools-1.4.4-10.el8.x86_64 cri-tools-1.13.0-0.x86_64 kubeadm-1.22.0-0.x86_64 kubectl-1.22.0-0.x86_64 kubelet-1.22.0-0.x86_64 kubernetes-cni-0.8.7-0.x86_64 libnetfilter_cthelper-1.0.0-15.el8.x86_64 libnetfilter_cttimeout-1.0.0-11.el8.x86_64 libnetfilter_queue-1.0.4-3.el8.x86_64 socat-1.7.3.3-2.el8.x86_64

Complete!

[root@kpod002 ~]# swapoff -a

[root@kpod002 ~]# service iptables stop

Redirecting to /bin/systemctl stop iptables.service

Failed to stop iptables.service: Unit iptables.service not loaded.

[root@kpod002 ~]# systemctl stop firewalld

[root@kpod002 ~]# systemctl disable firewalld

Removed /etc/systemd/system/multi-user.target.wants/firewalld.service.

Removed /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.

[root@kpod002 ~]# echo br_netfilter > /etc/modules-load.d/br_netfilter.conf



kubernetes

```
[root@kpod002 ~]# systemctl restart
```

```
systemd-modules-load.service
```

```
[root@kpod002 ~]# echo 1 > /proc/sys/net/bridge/bridge-nf-call-iptables
```

```
[root@kpod002 ~]# echo 1 > /proc/sys/net/bridge/bridge-nf-call-ip6tables
```

```
[root@kpod002 ~]# echo 1 > /proc/sys/net/ipv4/ip_forward
```

```
[root@kpod002 ~]# yum install -y yum-utils
```

Last metadata expiration check: 0:01:26 ago on Fri 13 Aug 2021 01:24:11 PM -03.

Dependencies resolved.

```
=====
=====
=====
```

Package	Size	Architecture	Version	Repository
---------	------	--------------	---------	------------

```
=====
=====
=====
```

Installing:

yum-utils		noarch	4.0.18-	
4.el8	ol8_baseos_latest		71 k	

Upgrading:

dnf-plugins-core		noarch	4.0.18-	
4.el8	ol8_baseos_latest		69 k	
python3-dnf-plugins-core		noarch	4.0.18-	
4.el8	ol8_baseos_latest		234 k	

Transaction Summary

```
=====
=====
=====
```

Install 1 Package

Upgrade 2 Packages

Total download size: 375 k

Downloading Packages:

(1/3): dnf-plugins-core-4.0.18-

4.el8.noarch.rpm

224 kB/s | 69

kB 00:00

(2/3): python3-dnf-plugins-core-4.0.18-

4.el8.noarch.rpm

685 kB/s | 234 kB 00:00

(3/3): yum-utils-4.0.18-

4.el8.noarch.rpm

20 kB/s | 71

kB 00:03

Total

104 kB/s |

375 kB 00:03

Running transaction check



Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

Preparing :

1/1

Upgrading : python3-dnf-plugins-core-4.0.18-4.el8.noarch

1/5

Upgrading : dnf-plugins-core-4.0.18-4.el8.noarch

2/5

Installing : yum-utils-4.0.18-4.el8.noarch

3/5

Cleanup : dnf-plugins-core-4.0.17-5.el8.noarch

4/5

Cleanup : python3-dnf-plugins-core-4.0.17-5.el8.noarch

5/5

Running scriptlet: python3-dnf-plugins-core-4.0.17-5.el8.noarch

5/5

Verifying : yum-utils-4.0.18-4.el8.noarch

1/5

Verifying : dnf-plugins-core-4.0.18-4.el8.noarch

2/5

Verifying : dnf-plugins-core-4.0.17-5.el8.noarch

3/5

Verifying : python3-dnf-plugins-core-4.0.18-4.el8.noarch

4/5

Verifying : python3-dnf-plugins-core-4.0.17-5.el8.noarch

5/5

Upgraded:

dnf-plugins-core-4.0.18-4.el8.noarch
4.0.18-4.el8.noarch

python3-dnf-plugins-core-

Installed:

yum-utils-4.0.18-4.el8.noarch

Complete!

[root@kpod002 ~]# yum-config-manager --add-

repo <https://download.docker.com/linux/centos/docker-ce.repo>

Adding repo from: <https://download.docker.com/linux/centos/docker-ce.repo>

[root@kpod002 ~]# yum remove podman

Dependencies resolved.

=====

Package	Architecture	Version	Repo
sitory	Size		



```
=====
=====
=====
```

Removing:

podman	x86_64	2.0.5-	
5.0.1.module+el8.3.0+7866+f387f528		@AppStream	51 M

Removing dependent packages:

cockpit-podman	noarch	18.1-	
2.module+el8.3.0+7866+f387f528		@AppStream	4.9 M

Removing unused dependencies:

common	x86_64	2:2.0.20-	
2.module+el8.3.0+7866+f387f528		@AppStream	118 k

libvarlink	x86_64	18-	
3.el8		@anaconda	117 k

podman-catatonit	x86_64	2.0.5-	
5.0.1.module+el8.3.0+7866+f387f528		@AppStream	752 k

Transaction Summary

```
=====
=====
=====
```

Remove 5 Packages

Freed space: 57 M

Is this ok [y/N]: y

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

Preparing :

1/1

Running scriptlet: cockpit-podman-18.1-

2.module+el8.3.0+7866+f387f528.noarch	1
/1	

Erasing : cockpit-podman-18.1-

2.module+el8.3.0+7866+f387f528.noarch	1
/5	

Erasing : podman-2.0.5-

5.0.1.module+el8.3.0+7866+f387f528.x86_64	
2/5	

Running scriptlet: podman-2.0.5-

5.0.1.module+el8.3.0+7866+f387f528.x86_64	
2/5	



```
Erasing      : podman-catatonit-
2.0.5-
5.0.1.module+el8.3.0+7866+f387f528.x86_64      3
/5
Erasing      : conmon-2:2.0.20-
2.module+el8.3.0+7866+f387f528.x86_64
4/5
Erasing      : libvarlink-18-
3.el8.x86_64      5/5
Running scriptlet: libvarlink-18-
3.el8.x86_64      5/5
Verifying    : cockpit-podman-18.1-
2.module+el8.3.0+7866+f387f528.noarch      1
/5
Verifying    : conmon-2:2.0.20-
2.module+el8.3.0+7866+f387f528.x86_64
2/5
Verifying    : libvarlink-18-
3.el8.x86_64      3/5
Verifying    : podman-2.0.5-
5.0.1.module+el8.3.0+7866+f387f528.x86_64
4/5
Verifying    : podman-catatonit-2.0.5-
5.0.1.module+el8.3.0+7866+f387f528.x86_64      5
/5
```

```
Removed:
cockpit-podman-18.1-2.module+el8.3.0+7866+f387f528.noarch    conmon-2:2.0.20-
2.module+el8.3.0+7866+f387f528.x86_64    libvarlink-18-3.el8.x86_64
podman-2.0.5-5.0.1.module+el8.3.0+7866+f387f528.x86_64    podman-catatonit-2.0.5-
5.0.1.module+el8.3.0+7866+f387f528.x86_64
```

```
Complete!
[root@kpod002 ~]# yum remove containerd runc buildah
No match for argument: containerd
Dependencies resolved.
```

```
=====
=====
=====
```

Package	Architecture	Version	Repo
sitory	Size		

```
=====
=====
=====
```

```
Removing:
buildah      x86_64      1.15.1-
2.0.1.module+el8.3.0+7866+f387f528      @AppStream      28 M
```



runc	x86_64	1.0.0-68.rc92.module+el8.3.0+7866+f387f528	@AppStream	13 M
Removing unused dependencies:				
container-selinux	noarch	2:2.144.0-1.module+el8.3.0+7866+f387f528	@AppStream	46 k
containers-common	x86_64	1:1.1.1-3.0.1.module+el8.3.0+7866+f387f528	@AppStream	69 k
criu	x86_64	3.14-2.module+el8.3.0+7866+f387f528	@AppStream	1.3 M
fuse-overlayfs	x86_64	1.1.2-3.module+el8.3.0+7866+f387f528	@AppStream	134 k
fuse3-libs	x86_64	3.2.1-12.0.2.el8	@anaconda	279 k
libnet	x86_64	1.1.6-15.el8	@AppStream	170 k
libslirp	x86_64	4.3.1-1.module+el8.3.0+7866+f387f528	@AppStream	129 k
protobuf-c	x86_64	1.3.0-4.el8	@AppStream	56 k
slirp4netns	x86_64	1.1.4-2.module+el8.3.0+7866+f387f528	@AppStream	95 k

Transaction Summary

=====

=====

=====

Remove 11 Packages

Freed space: 43 M

Is this ok [y/N]: y

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

Preparing :

1/1

Erasing : buildah-1.15.1-

2.0.1.module+el8.3.0+7866+f387f528.x86_64

1/11

Erasing : containers-common-1:1.1.1-

3.0.1.module+el8.3.0+7866+f387f528.x86_64

2/1

1

Erasing : container-selinux-2:2.144.0-

1.module+el8.3.0+7866+f387f528.noarch

3/11



Running scriptlet: container-selinux-

```
2:2.144.0-
1.module+el8.3.0+7866+f387f528.noarch 3/11
Erasing      : fuse-overlayfs-1.1.2-
3.module+el8.3.0+7866+f387f528.x86_64 4/
11
Erasing      : slirp4netns-1.1.4-
2.module+el8.3.0+7866+f387f528.x86_64
5/11
Erasing      : runc-1.0.0-
68.rc92.module+el8.3.0+7866+f387f528.x86_64
6/11
Erasing      : criu-3.14-
2.module+el8.3.0+7866+f387f528.x86_64
7/11
Erasing      : libnet-1.1.6-
15.el8.x86_64 8/11
Running scriptlet: libnet-1.1.6-
15.el8.x86_64 8/11
Erasing      : protobuf-c-1.3.0-
4.el8.x86_64 9/11
Erasing      : libslirp-4.3.1-
1.module+el8.3.0+7866+f387f528.x86_64
10/11
Erasing      : fuse3-libs-3.2.1-
12.0.2.el8.x86_64 11/11
Running scriptlet: fuse3-libs-3.2.1-
12.0.2.el8.x86_64 11/11
Verifying    : buildah-1.15.1-
2.0.1.module+el8.3.0+7866+f387f528.x86_64
1/11
Verifying    : container-selinux-2:2.144.0-
1.module+el8.3.0+7866+f387f528.noarch 2/11
Verifying    : containers-common-1:1.1.1-
3.0.1.module+el8.3.0+7866+f387f528.x86_64 3/1
1
Verifying    : criu-3.14-
2.module+el8.3.0+7866+f387f528.x86_64
4/11
Verifying    : fuse-overlayfs-1.1.2-
3.module+el8.3.0+7866+f387f528.x86_64 5/
11
Verifying    : fuse3-libs-3.2.1-
12.0.2.el8.x86_64 6/11
Verifying    : libnet-1.1.6-
15.el8.x86_64 7/11
```



Verifying : libslirp-4.3.1-

1.module+el8.3.0+7866+f387f528.x86_64
8/11

Verifying : protobuf-c-1.3.0-

4.el8.x86_64

9/11

Verifying : runc-1.0.0-

68.rc92.module+el8.3.0+7866+f387f528.x86_64
10/11

Verifying : slirp4netns-1.1.4-

2.module+el8.3.0+7866+f387f528.x86_64
1/11

1

Removed:

buildah-1.15.1-2.0.1.module+el8.3.0+7866+f387f528.x86_64 container-

selinux-2:2.144.0-1.module+el8.3.0+7866+f387f528.noarch

containers-common-1:1.1.1-3.0.1.module+el8.3.0+7866+f387f528.x86_64 criu-

3.14-2.module+el8.3.0+7866+f387f528.x86_64

fuse-overlayfs-1.1.2-3.module+el8.3.0+7866+f387f528.x86_64 fuse3-libs-

3.2.1-12.0.2.el8.x86_64

libnet-1.1.6-15.el8.x86_64 libslirp-4.3.1-

1.module+el8.3.0+7866+f387f528.x86_64

protobuf-c-1.3.0-4.el8.x86_64 runc-1.0.0-

68.rc92.module+el8.3.0+7866+f387f528.x86_64

slirp4netns-1.1.4-2.module+el8.3.0+7866+f387f528.x86_64

Complete!

[root@kpod002 ~]# yum install docker-ce docker-ce-cli containerd.io

Docker CE Stable - x86_64 70

kB/s | 15 kB 00:00

Dependencies resolved.

=====

=====

=====

Package	Architecture	Version	Reposit
ory	Size		

=====

=====

=====

Installing:

containerd.io	x86_64	1.4.9-	
3.1.el8		docker-ce-stable	30 M
docker-ce	x86_64	3:20.10.8-	
3.el8		docker-ce-stable	22 M
docker-ce-cli	x86_64	1:20.10.8-	
3.el8		docker-ce-stable	29 M

Installing dependencies:



```
container-
selinux                noarch                2:2.164.1-
1.module+el8.4.0+20289+730b73cc                ol8_appstream                52 k
docker-ce-rootless-extras                x86_64                20.10.8-
3.el8                docker-ce-stable                4.6 M
docker-scan-plugin                x86_64                0.8.0-
3.el8                docker-ce-stable                4.2 M
fuse-common                x86_64                3.2.1-
12.0.3.el8                ol8_baseos_latest                22 k
fuse-overlayfs                x86_64                1.6-
1.module+el8.4.0+20289+730b73cc                ol8_appstream                73 k
fuse3                x86_64                3.2.1-
12.0.3.el8                ol8_baseos_latest                51 k
fuse3-libs                x86_64                3.2.1-
12.0.3.el8                ol8_baseos_latest                95 k
libcgroup                x86_64                0.41-
19.el8                ol8_baseos_latest                70 k
libslirp                x86_64                4.3.1-
1.module+el8.4.0+20289+730b73cc                ol8_appstream                69 k
slirp4netns                x86_64                1.1.8-
1.module+el8.4.0+20289+730b73cc                ol8_appstream                51 k
```

Transaction Summary

```
=====
=====
=====
```

Install 13 Packages

Total download size: 90 M

Installed size: 377 M

Is this ok [y/N]: y

Downloading Packages:

(1/13): docker-ce-20.10.8-

3.el8.x86_64.rpm

3.9 MB/s | 22

MB 00:05

(2/13): docker-ce-rootless-extras-20.10.8-

3.el8.x86_64.rpm

2.9 MB/s | 4.6 MB 00:01

(3/13): docker-scan-plugin-0.8.0-

3.el8.x86_64.rpm

3.2 MB/s | 4.2

MB 00:01

(4/13): fuse-common-3.2.1-

12.0.3.el8.x86_64.rpm

276 kB/s | 22

kB 00:00

(5/13): fuse3-3.2.1-

12.0.3.el8.x86_64.rpm

986 kB/s | 51

kB 00:00



(6/13): fuse3-libs-3.2.1-

12.0.3.el8.x86_64.rpm 1.5 MB/s | 95
kB 00:00

(7/13): libcgroup-0.41-
19.el8.x86_64.rpm 1.7 MB/s | 70
kB 00:00

(8/13): container-selinux-2.164.1-
1.module+el8.4.0+20289+730b73cc.noarch.rpm 785 kB/s
| 52 kB 00:00

(9/13): fuse-overlayfs-1.6-
1.module+el8.4.0+20289+730b73cc.x86_64.rpm 1.6
MB/s | 73 kB 00:00

(10/13): libslirp-4.3.1-
1.module+el8.4.0+20289+730b73cc.x86_64.rpm 870
kB/s | 69 kB 00:00

(11/13): slirp4netns-1.1.8-
1.module+el8.4.0+20289+730b73cc.x86_64.rpm 930
kB/s | 51 kB 00:00

(12/13): docker-ce-cli-20.10.8-
3.el8.x86_64.rpm 2.6 MB/s | 29
MB 00:11

(13/13): containerd.io-1.4.9-
3.1.el8.x86_64.rpm 2.5 MB/s | 30
MB 00:11

Total 7.7 MB/s
| 90 MB 00:11

warning: /var/cache/dnf/docker-ce-stable-fa9dc42ab4cec2f4/packages/containerd.io-1.4.9-
3.1.el8.x86_64.rpm: Header V4 RSA/SHA512 Signature, key ID 621e9f35: NOKEY

Docker CE Stable - x86_64 14
kB/s | 1.6 kB 00:00

Importing GPG key 0x621E9F35:

Userid : "Docker Release (CE rpm) <docker@docker.com>"

Fingerprint: 060A 61C5 1B55 8A7F 742B 77AA C52F EB6B 621E 9F35

From : <https://download.docker.com/linux/centos/gpg>

Is this ok [y/N]: y

Key imported successfully

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

Preparing :

1/1



Installing : docker-scan-plugin-

0.8.0-
3.el8.x86_64 1/13
Running scriptlet: docker-scan-plugin-0.8.0-
3.el8.x86_64 1/13
Installing : docker-ce-cli-1:20.10.8-
3.el8.x86_64 2/13
Running scriptlet: docker-ce-cli-1:20.10.8-
3.el8.x86_64 2/13
Running scriptlet: container-selinux-2:2.164.1-
1.module+el8.4.0+20289+730b73cc.noarch 3/13
Installing : container-selinux-2:2.164.1-
1.module+el8.4.0+20289+730b73cc.noarch 3/13
Running scriptlet: container-selinux-2:2.164.1-
1.module+el8.4.0+20289+730b73cc.noarch 3/13
Installing : containerd.io-1.4.9-
3.1.el8.x86_64 4/13
Running scriptlet: containerd.io-1.4.9-
3.1.el8.x86_64 4/13
Installing : libslirp-4.3.1-
1.module+el8.4.0+20289+730b73cc.x86_64 5/13
Installing : slirp4netns-1.1.8-
1.module+el8.4.0+20289+730b73cc.x86_64 6/13
Running scriptlet: libcgrou-0.41-
19.el8.x86_64 7/13
Installing : libcgrou-0.41-
19.el8.x86_64 7/13
Running scriptlet: libcgrou-0.41-
19.el8.x86_64 7/13
Installing : fuse3-libs-3.2.1-
12.0.3.el8.x86_64 8/13
Running scriptlet: fuse3-libs-3.2.1-
12.0.3.el8.x86_64 8/13
Installing : fuse-common-3.2.1-
12.0.3.el8.x86_64 9/13
Installing : fuse3-3.2.1-
12.0.3.el8.x86_64 10/13
Installing : fuse-overlayfs-1.6-
1.module+el8.4.0+20289+730b73cc.x86_64 11/13
Running scriptlet: fuse-overlayfs-1.6-
1.module+el8.4.0+20289+730b73cc.x86_64 11/13
Installing : docker-ce-rootless-extras-20.10.8-
3.el8.x86_64 12/13



Running scriptlet: docker-ce-rootless-

extras-20.10.8-3.el8.x86_64	12/13
Installing : docker-ce-3:20.10.8-3.el8.x86_64	13/13
Running scriptlet: docker-ce-3:20.10.8-3.el8.x86_64	13/13
Running scriptlet: container-selinux-2:2.164.1-1.module+el8.4.0+20289+730b73cc.noarch	13/13
Running scriptlet: docker-ce-3:20.10.8-3.el8.x86_64	13/13
Verifying : containerd.io-1.4.9-3.1.el8.x86_64	1/13
Verifying : docker-ce-3:20.10.8-3.el8.x86_64	2/13
Verifying : docker-ce-cli-1:20.10.8-3.el8.x86_64	3/13
Verifying : docker-ce-rootless-extras-20.10.8-3.el8.x86_64	4/13
Verifying : docker-scan-plugin-0.8.0-3.el8.x86_64	5/13
Verifying : fuse-common-3.2.1-12.0.3.el8.x86_64	6/13
Verifying : fuse3-3.2.1-12.0.3.el8.x86_64	7/13
Verifying : fuse3-libs-3.2.1-12.0.3.el8.x86_64	8/13
Verifying : libcgroupp-0.41-19.el8.x86_64	9/13
Verifying : container-selinux-2:2.164.1-1.module+el8.4.0+20289+730b73cc.noarch	10/13
Verifying : fuse-overlayfs-1.6-1.module+el8.4.0+20289+730b73cc.x86_64	11/13
Verifying : libslirp-4.3.1-1.module+el8.4.0+20289+730b73cc.x86_64	12/13
Verifying : slirp4netns-1.1.8-1.module+el8.4.0+20289+730b73cc.x86_64	13/13

Installed:

container-selinux-2:2.164.1-1.module+el8.4.0+20289+730b73cc.noarch	containerd.io-1.4.9-3.1.el8.x86_64
docker-ce-3:20.10.8-3.el8.x86_64	docker-ce-cli-1:20.10.8-3.el8.x86_64



```
docker-ce-rootless-extras-20.10.8-
3.el8.x86_64
docker-scan-plugin-0.8.0-
3.el8.x86_64
fuse-common-3.2.1-12.0.3.el8.x86_64
fuse-overlays-1.6-
1.module+el8.4.0+20289+730b73cc.x86_64
fuse3-3.2.1-12.0.3.el8.x86_64
fuse3-libs-3.2.1-
12.0.3.el8.x86_64
libcgroup-0.41-19.el8.x86_64
libslirp-4.3.1-
1.module+el8.4.0+20289+730b73cc.x86_64
slirp4netns-1.1.8-1.module+el8.4.0+20289+730b73cc.x86_64
```

Complete!

```
[root@kpod002 ~]# systemctl start docker
[root@kpod002 ~]# systemctl enable docker
Created symlink /etc/systemd/system/multi-user.target.wants/docker.service →
/usr/lib/systemd/system/docker.service.
[root@kpod002 ~]# vi /etc/resolv.conf
[root@kpod002 ~]# cat > /etc/docker/daemon.json <<EOF
> {
> "exec-opts": ["native.cgroupdriver=systemd"],
> "log-driver": "json-file",
> "log-opts": {
> "max-size": "100m"
> },
> "storage-driver": "overlay2"
> }
> EOF
[root@kpod002 ~]#
[root@kpod002 ~]# systemctl restart docker
[root@kpod002 ~]# docker info | grep Cgroup
Cgroup Driver: systemd
Cgroup Version: 1
```

AÑADIR NODOS AL CLUSTER DE k8S

Genero de nuevo el token

```
[root@kmaster01 ~]# kubeadm token create
kgbzyp.gp5e0177m7u7sz3k
```

```
[root@kpod002 ~]# kubeadm join 172.16.8.200:6443 --token
kgbzyp.gp5e0177m7u7sz3k --discovery-token-ca-cert-hash
sha256:dfe37a02c9d21d53f7963b6c5725b9ddd6056df12ba8a9739b784b5ed39d7980 --v=5
I0813 13:28:14.762628 71341 join.go:405] [preflight] found NodeName empty; using OS
hostname as NodeName
```



```
I0813 13:28:14.762701 71341
initconfiguration.go:116] detected and using CRI socket:
/var/run/dockershim.sock
[preflight] Running pre-flight checks
I0813 13:28:14.762754 71341 preflight.go:92] [preflight] Running general checks
I0813 13:28:14.762786 71341 checks.go:245] validating the existence and emptiness of
directory /etc/kubernetes/manifests
I0813 13:28:14.762818 71341 checks.go:282] validating the existence of file
/etc/kubernetes/kubelet.conf
I0813 13:28:14.762825 71341 checks.go:282] validating the existence of file
/etc/kubernetes/bootstrap-kubelet.conf
I0813 13:28:14.762845 71341 checks.go:106] validating the container runtime
I0813 13:28:14.891665 71341 checks.go:132] validating if the "docker" service is enabled
and active
I0813 13:28:14.926761 71341 checks.go:331] validating the contents of file
/proc/sys/net/bridge/bridge-nf-call-iptables
I0813 13:28:14.926832 71341 checks.go:331] validating the contents of file
/proc/sys/net/ipv4/ip_forward
I0813 13:28:14.926869 71341 checks.go:649] validating whether swap is enabled or not
I0813 13:28:14.926909 71341 checks.go:372] validating the presence of executable
conntrack
I0813 13:28:14.926950 71341 checks.go:372] validating the presence of executable ip
I0813 13:28:14.926971 71341 checks.go:372] validating the presence of executable
iptables
I0813 13:28:14.926994 71341 checks.go:372] validating the presence of executable mount
I0813 13:28:14.927039 71341 checks.go:372] validating the presence of executable
nsenter
I0813 13:28:14.927071 71341 checks.go:372] validating the presence of executable
ebtables
I0813 13:28:14.927098 71341 checks.go:372] validating the presence of executable ethtool
I0813 13:28:14.927245 71341 checks.go:372] validating the presence of executable socat
I0813 13:28:14.927864 71341 checks.go:372] validating the presence of executable tc
[WARNING FileExisting-tc]: tc not found in system path
I0813 13:28:14.928157 71341 checks.go:372] validating the presence of executable touch
I0813 13:28:14.928263 71341 checks.go:520] running all checks
I0813 13:28:15.114091 71341 checks.go:403] checking whether the given node name is
valid and reachable using net.LookupHost
I0813 13:28:15.114975 71341 checks.go:618] validating kubelet version
I0813 13:28:15.241712 71341 checks.go:132] validating if the "kubelet" service is enabled
and active
[WARNING Service-Kubelet]: kubelet service is not enabled, please run 'systemctl
enable kubelet.service'
I0813 13:28:15.268839 71341 checks.go:205] validating availability of port 10250
I0813 13:28:15.269044 71341 checks.go:282] validating the existence of file
/etc/kubernetes/pki/ca.crt
I0813 13:28:15.269073 71341 checks.go:432] validating if the connectivity type is via proxy
or direct
```




If that is not intended, adjust your

proxy settings

```
I0813 13:28:15.269187 71341 join.go:475] [preflight] Discovering cluster-info
I0813 13:28:15.269250 71341 token.go:80] [discovery] Created cluster-info discovery
client, requesting info from "172.16.8.200:6443"
I0813 13:28:15.273107 71341 token.go:217] [discovery] Failed to request cluster-info, will
try again: Get "https://172.16.8.200:6443/api/v1/namespaces/kube-
public/configmaps/cluster-info?timeout=10s": Forbidden
I0813 13:28:21.184891 71341 token.go:217] [discovery] Failed to request cluster-info, will
try again: Get "https://172.16.8.200:6443/api/v1/namespaces/kube-
public/configmaps/cluster-info?timeout=10s": Forbidden
I0813 13:28:27.601457 71341 token.go:217] [discovery] Failed to request cluster-info, will
try again: Get "https://172.16.8.200:6443/api/v1/namespaces/kube-
public/configmaps/cluster-info?timeout=10s": Forbidden
I0813 13:28:33.605114 71341 token.go:217] [discovery] Failed to request cluster-info, will
try again: Get "https://172.16.8.200:6443/api/v1/namespaces/kube-
public/configmaps/cluster-info?timeout=10s": Forbidden
```

Lo revisas asi:

```
[root@kmaster01 .kube]# kubectl get nodes -o wide
NAME      STATUS ROLES      AGE  VERSION  INTERNAL-IP  EXTERNAL-IP  OS-
IMAGE      KERNEL-VERSION  CONTAINER-RUNTIME
kmaster01 Ready  control-plane,master  284d  v1.21.3  172.16.8.200  <none>      Oracle
Linux Server 8.4  5.4.17-2102.203.6.el8uek.x86_64  docker://20.10.7
kpod001   Ready  <none>          284d  v1.22.0  172.16.8.201  <none>      Oracle Linux
Server 8.3  5.4.17-2011.7.4.el8uek.x86_64  docker://20.10.8
kpod002   Ready  <none>          284d  v1.22.0  172.16.8.202  <none>      Oracle Linux
Server 8.3  5.4.17-2011.7.4.el8uek.x86_64  docker://20.10.8
```

```
[root@kpod002 ~]# logout
```

Connection to kpod002 closed.

```
[root@kmaster01 ~]# ssh kpod002
```

root@kpod002's password:

Activate the web console with: `systemctl enable --now cockpit.socket`

Last login: Fri Aug 13 13:22:18 2021 from 172.16.8.200

```
[root@kpod002 ~]# kubeadm join 172.16.8.200:6443 --token
```

```
kgbzyg.gp5e0177m7u7sz3k --discovery-token-ca-cert-hash
```

```
sha256:dfc37a02c9d21d53f7963b6c5725b9ddd6056df12ba8a9739b784b5ed39d7980 --v=5
```

```
I0813 13:28:44.365047 71521 join.go:405] [preflight] found NodeName empty; using OS
hostname as NodeName
```

```
I0813 13:28:44.365117 71521 initconfiguration.go:116] detected and using CRI socket:
/var/run/dockershim.sock
```

```
[preflight] Running pre-flight checks
```



I0813 13:28:44.365171 71521

preflight.go:92] [preflight] Running general checks

I0813 13:28:44.365205 71521 checks.go:245] validating the existence and emptiness of directory /etc/kubernetes/manifests

I0813 13:28:44.365231 71521 checks.go:282] validating the existence of file /etc/kubernetes/kubelet.conf

I0813 13:28:44.365237 71521 checks.go:282] validating the existence of file /etc/kubernetes/bootstrap-kubelet.conf

I0813 13:28:44.365247 71521 checks.go:106] validating the container runtime

I0813 13:28:44.489969 71521 checks.go:132] validating if the "docker" service is enabled and active

I0813 13:28:44.529486 71521 checks.go:331] validating the contents of file /proc/sys/net/bridge/bridge-nf-call-iptables

I0813 13:28:44.529563 71521 checks.go:331] validating the contents of file /proc/sys/net/ipv4/ip_forward

I0813 13:28:44.529601 71521 checks.go:649] validating whether swap is enabled or not

I0813 13:28:44.529661 71521 checks.go:372] validating the presence of executable conntrack

I0813 13:28:44.529693 71521 checks.go:372] validating the presence of executable ip

I0813 13:28:44.529715 71521 checks.go:372] validating the presence of executable iptables

I0813 13:28:44.529738 71521 checks.go:372] validating the presence of executable mount

I0813 13:28:44.529770 71521 checks.go:372] validating the presence of executable nsenter

I0813 13:28:44.529794 71521 checks.go:372] validating the presence of executable ebtables

I0813 13:28:44.529815 71521 checks.go:372] validating the presence of executable ethtool

I0813 13:28:44.529835 71521 checks.go:372] validating the presence of executable socat

I0813 13:28:44.529857 71521 checks.go:372] validating the presence of executable tc

[WARNING FileExisting-tc]: tc not found in system path

I0813 13:28:44.529941 71521 checks.go:372] validating the presence of executable touch

I0813 13:28:44.529965 71521 checks.go:520] running all checks

I0813 13:28:44.661502 71521 checks.go:403] checking whether the given node name is valid and reachable using net.LookupHost

I0813 13:28:44.662310 71521 checks.go:618] validating kubelet version

I0813 13:28:44.739628 71521 checks.go:132] validating if the "kubelet" service is enabled and active

[WARNING Service-Kubelet]: kubelet service is not enabled, please run 'systemctl enable kubelet.service'

I0813 13:28:44.769442 71521 checks.go:205] validating availability of port 10250

I0813 13:28:44.769640 71521 checks.go:282] validating the existence of file /etc/kubernetes/pki/ca.crt

I0813 13:28:44.769674 71521 checks.go:432] validating if the connectivity type is via proxy or direct

I0813 13:28:44.769772 71521 join.go:475] [preflight] Discovering cluster-info

I0813 13:28:44.769816 71521 token.go:80] [discovery] Created cluster-info discovery client, requesting info from "172.16.8.200:6443"



I0813 13:28:44.787935 71521
token.go:118] [discovery] Requesting info from
"172.16.8.200:6443" again to validate TLS against the pinned public key
I0813 13:28:44.798758 71521 token.go:135] [discovery] Cluster info signature and
contents are valid and TLS certificate validates against pinned roots, will use API Server
"172.16.8.200:6443"
I0813 13:28:44.798784 71521 discovery.go:52] [discovery] Using provided
TLSBootstrapToken as authentication credentials for the join process
I0813 13:28:44.798798 71521 join.go:489] [preflight] Fetching init configuration
I0813 13:28:44.798808 71521 join.go:534] [preflight] Retrieving KubeConfig objects
[preflight] Reading configuration from the cluster...
[preflight] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-
config -o yaml'
I0813 13:28:44.813156 71521 interface.go:431] Looking for default routes with IPv4
addresses
I0813 13:28:44.813174 71521 interface.go:436] Default route transits interface "bond0.8"
I0813 13:28:44.814516 71521 interface.go:208] Interface bond0.8 is up
I0813 13:28:44.814667 71521 interface.go:256] Interface "bond0.8" has 2 addresses
:[172.16.8.202/24 fe80::3a68:ddff:fe36:8b50/64].
I0813 13:28:44.814714 71521 interface.go:223] Checking addr 172.16.8.202/24.
I0813 13:28:44.814764 71521 interface.go:230] IP found 172.16.8.202
I0813 13:28:44.814780 71521 interface.go:262] Found valid IPv4 address 172.16.8.202 for
interface "bond0.8".
I0813 13:28:44.814813 71521 interface.go:442] Found active IP 172.16.8.202
I0813 13:28:44.932321 71521 preflight.go:103] [preflight] Running configuration
dependant checks
I0813 13:28:44.932362 71521 controlplaneprepare.go:219] [download-certs] Skipping
certs download
I0813 13:28:44.932394 71521 kubelet.go:112] [kubelet-start] writing bootstrap kubelet
config file at /etc/kubernetes/bootstrap-kubelet.conf
I0813 13:28:44.933975 71521 kubelet.go:120] [kubelet-start] writing CA certificate at
/etc/kubernetes/pki/ca.crt
I0813 13:28:44.935249 71521 kubelet.go:141] [kubelet-start] Checking for an existing Node
in the cluster with name "kpod002" and status "Ready"
I0813 13:28:44.939131 71521 kubelet.go:155] [kubelet-start] Stopping the kubelet
[kubelet-start] Writing kubelet configuration to file "/var/lib/kubelet/config.yaml"
[kubelet-start] Writing kubelet environment file with flags to file
"/var/lib/kubelet/kubeadm-flags.env"
[kubelet-start] Starting the kubelet
[kubelet-start] Waiting for the kubelet to perform the TLS Bootstrap...
I0813 13:28:50.119783 71521 cert_rotation.go:137] Starting client certificate rotation
controller
I0813 13:28:50.121395 71521 kubelet.go:190] [kubelet-start] preserving the crisocket
information for the node
I0813 13:28:50.121418 71521 patchnode.go:31] [patchnode] Uploading the CRI Socket
information "/var/run/dockershim.sock" to the Node API object "kpod002" as an annotation



This node has joined the cluster:

- * Certificate signing request was sent to apiserver and a response was received.
- * The Kubelet was informed of the new secure connection details.

Run 'kubectl get nodes' on the control-plane to see this node join the cluster.

Finalmente

```
[root@kmaster01 ~]# kubectl get no
NAME      STATUS    ROLES                  AGE    VERSION
kmaster01 NotReady  control-plane,master  3d2h   v1.21.3
kpod001   NotReady  <none>                 9m54s  v1.22.0
kpod002   NotReady  <none>                 10s    v1.22.0
```

Luego viene la instalación de los pods de red.

Los pods de proxy quedan tirados porque no encuentran la imagen en los kpod

Dando este error:

Events:

Type	Reason	Age	From	Message
Normal	Scheduled	100s	default-scheduler	Successfully assigned kube-system/kube-proxy-fhs87 to kpod001
Warning	FailedCreatePodSandBox	<invalid>	kubelet	Failed to create pod sandbox: rpc error: code = Unknown desc = failed pulling image "k8s.gcr.io/pause:3.5": Error response from daemon: Get "https://k8s.gcr.io/v2/": net/http: request canceled while waiting for connection (Client.Timeout exceeded while awaiting headers)

esto se soluciona bajando la imagen y cargandola en el pod:

```
docker load -i pause.tar
```

Luego armamos los pods de red con lo siguiente

```
export kubever=$(kubectl version | base64 | tr -d '\n')
```

```
# kubectl apply -f "https://cloud.weave.works/k8s/net?k8s-version=$kubever"
```

Para esto encuentre el siguiente manifiesto



```
[root@master-node home]# export kubever=$(kubectl version | base64 | tr -d '\n')
[root@master-node home]# kubectl apply -f "https://cloud.weave.works/k8s/net?k8s-version=$kubever"
serviceaccount/weave-net created
clusterrole.rbac.authorization.k8s.io/weave-net created
clusterrolebinding.rbac.authorization.k8s.io/weave-net created
role.rbac.authorization.k8s.io/weave-net created
rolebinding.rbac.authorization.k8s.io/weave-net created
daemonset.apps/weave-net created
[root@master-node home]#
```

- Saber info del cluster

```
[root@kmaster01 k8s_home]# kubectl cluster-info
Kubernetes control plane is running at https://172.16.8.200:6443
CoreDNS is running at https://172.16.8.200:6443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy
```

<https://www.tecmint.com/install-a-kubernetes-cluster-on-centos-8/>

Now if you check the status of your master-node, it should be 'Ready'.

```
# kubectl get nodes
```

```
[root@master-node home]# kubectl get nodes
NAME           STATUS    ROLES    AGE   VERSION
master-node    Ready     master   23m   v1.17.0
[root@master-node home]#
```

Comandos de administración



1. Validar cluster activo

```
[root@kmaster01 ~]# kubectl get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
kmaster01	Ready	control-plane,master	281d	v1.21.3
kpod001	Ready	<none>	281d	v1.22.0
kpod002	Ready	<none>	281d	v1.22.0

Importante siempre verlos en READY

- Para más detalle

```
[root@kmaster01 ~]# kubectl get nodes -o wide
```

NAME	STATUS	ROLES	AGE	VERSION	INTERNAL-IP	EXTERNAL-IP	OS-IMAGE	KERNEL-VERSION	CONTAINER-RUNTIME
kmaster01	Ready	control-plane,master	281d	v1.21.3	172.16.8.200	<none>	Oracle Linux Server 8.4	5.4.17-2102.203.6.el8uek.x86_64	docker://20.10.7
kpod001	Ready	<none>	281d	v1.22.0	172.16.8.201	<none>	Oracle Linux Server 8.3	5.4.17-2011.7.4.el8uek.x86_64	docker://20.10.8
kpod002	Ready	<none>	281d	v1.22.0	172.16.8.202	<none>	Oracle Linux Server 8.3	5.4.17-2011.7.4.el8uek.x86_64	docker://20.10.8

2. Obtener servicios

```
[root@kmaster01 ~]# kubectl get svc
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
app-v1	ClusterIP	10.107.181.3	<none>	8080/TCP	183d
fluentd-aggregator	ClusterIP	10.97.246.165	<none>	9880/TCP,24224/TCP	22d
fluentd-forwarder	ClusterIP	10.104.75.157	<none>	9880/TCP	22d
fluentd-headless	ClusterIP	None	<none>	9880/TCP,24224/TCP	22d
kafka	ClusterIP	10.109.97.151	<none>	9092/TCP	18d
kafka-headless	ClusterIP	None	<none>	9092/TCP,9093/TCP	18d
kafka-zookeeper	ClusterIP	10.106.222.7	<none>	2181/TCP,2888/TCP,3888/TCP	18d
kafka-zookeeper-headless	ClusterIP	None	<none>	2181/TCP,2888/TCP,3888/TCP	18d
kubernetes	ClusterIP	10.96.0.1	<none>	443/TCP	281d
mongo	ClusterIP	None	<none>	27017/TCP	31d
mysql-db	NodePort	10.106.214.68	<none>	3306:30719/TCP	35d
nginx-kvm	ClusterIP	10.97.211.65	<none>	8080/TCP	31d
node-mongo	NodePort	10.102.48.1	<none>	8080:30000/TCP	31d

Lo más importante de este get, es ver el tipo de servicio si es ClusterIP (ingreso solamente interno) o NodePort (ingreso externo a través de algún puerto)



Si no se le especifica el namespace siempre toma el default, por lo que, aclarándolo, nos mostrara esos servicios que están contenidos allí

```
[root@kmaster01 ~]# kubectl get svc -n kube-system
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
kube-dns	ClusterIP	10.96.0.10	<none>	53/UDP,53/TCP,9153/TCP	281d

Para mas info del servicio

```
[root@kmaster01 ~]# kubectl get svc --namespace apigw-ns
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
kong-db-srv-cip	ClusterIP	10.108.96.11	<none>	5432/TCP	24d
kong-srv-cip	ClusterIP	10.96.167.213	<none>	8000/TCP,8001/TCP	24d
kong-srv-np	NodePort	10.100.187.134	<none>	8000:30600/TCP,8001:30601/TCP	24d

kongaui-srv-np	NodePort	10.109.185.225	<none>	1337:30621/TCP	24d
----------------	----------	----------------	--------	----------------	-----

```
[root@kmaster01 ~]# kubectl describe svc kong-srv-np --namespace apigw-ns
```

Name: kong-srv-np
Namespace: apigw-ns
Labels: <none>
Annotations: field.cattle.io/publicEndpoints: [{"port":30600,"protocol":"TCP","serviceName":"apigw-ns:kong-srv-np","allNodes":true},{ "port":30601,"protocol":"TCP","serviceName":"apigw-ns:kong-srv-np","allNodes":true}]
Selector: app=kong
Type: NodePort
IP Family Policy: SingleStack
IP Families: IPv4
IP: 10.100.187.134
IPs: 10.100.187.134
Port: kong-np 8000/TCP
TargetPort: 8000/TCP
NodePort: kong-np 30600/TCP
Endpoints: 10.46.0.20:8000
Port: kong-np-admin 8001/TCP
TargetPort: 8001/TCP
NodePort: kong-np-admin 30601/TCP
Endpoints: 10.46.0.20:8001
Session Affinity: None
External Traffic Policy: Cluster
Events: <none>

3. Ver Deployments

```
[root@kmaster01 ~]# kubectl get deploy --namespace default
```

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
------	-------	------------	-----------	-----



			mysql	0/0	0	0
35d						
nginx-test	2/2	2	2		31d	
node-mongo	0/0	0	0		31d	
tomcat-deployment	20/20	20	20		31d	

En los deployments voy a tener todo el detalle de mi micro, por ejemplo

```
[root@kmaster01 ~]# kubectl describe deployments kong-db --namespace apigw-ns
```

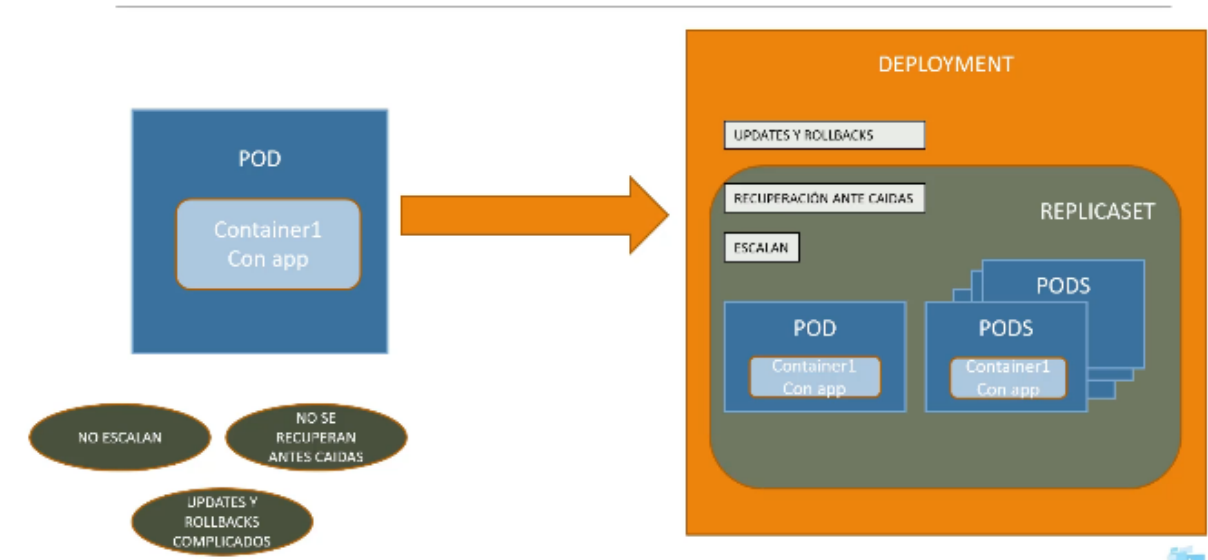
```
Name:          kong-db
Namespace:      apigw-ns
CreationTimestamp:  Tue, 03 May 2022 12:23:20 -0300
Labels:         app=kong-db
Annotations:     deployment.kubernetes.io/revision: 3
Selector:       workload.user.cattle.io/workloadselector=apps.deployment-api-gw-kong-db
Replicas:       1 desired | 1 updated | 1 total | 1 available | 0 unavailable
StrategyType:    RollingUpdate
MinReadySeconds: 0
RollingUpdateStrategy: 25% max unavailable, 25% max surge
Pod Template:
  Labels: app=kong-db
           workload.user.cattle.io/workloadselector=apps.deployment-api-gw-kong-db
  Containers:
    kong-db:
      Image:  postgres:9.6
      Port:   5432/TCP
      Host Port: 0/TCP
      Environment:
        POSTGRES_USER: kong
        POSTGRES_DB: kong
        POSTGRES_PASSWORD: kong
      Mounts:      <none>
  Volumes:
    kong-pvc:
      Type: PersistentVolumeClaim (a reference to a PersistentVolumeClaim in the same namespace)
      ClaimName: kong-pvc
      ReadOnly: false
  Conditions:
    Type      Status Reason
    ----      -
    Available  True  MinimumReplicasAvailable
    Progressing True  NewReplicaSetAvailable
  OldReplicaSets: <none>
  NewReplicaSet: kong-db-7ff4f5454d (1/1 replicas created)
```


Events: <none>

4. Pods

Siempre deben estar contenidos en los Deployments

Desventajas de PODS frente a Deployments:



5. Volumen

```
[root@kmaster01 ~]# kubectl get pv
```

NAME	CAPACITY	ACCESS MODES	RECLAIM POLICY	STATUS	CLAIM
STORAGECLASS	REASON	AGE			
fc-pv-volume	10Gi	RWO	Retain	Terminating	default/fc-pv-volume
fc-vol	182d				
fc-vol	10Gi	RWO	Retain	Bound	default/fc-vol-claim
171d					
kafka-pv	8Gi	RWX	Retain	Bound	default/data-kafka-0
longhorn-static	21d				
kafka-zookeeper-pv	8Gi	RWX	Retain	Bound	default/data-kafka-
zookeeper-0	longhorn-static	21d			
kong-pv	5Gi	RWX	Retain	Bound	apigw-ns/kong-pvc
longhorn-static	27d				
konga-pv	5Gi	RWX	Retain	Bound	apigw-ns/konga-pvc
longhorn-static	27d				
pv-nginx	1Gi	RWO	Retain	Bound	monitoring-ns/pvc-nginx
longhorn-static	39d				



		pv-testing	1Gi	RWO	
Retain	Bound	default/pv-testing			
47d					
pv-testing-vol	2Gi	RWO	Retain	Bound	monitoring-ns/pvc-testing-vol
longhorn-static		39d			
pv-vol-mongo	1Gi	RWX	Retain	Bound	default/pvc-vol-mongo
longhorn-static		35d			
pv-vol-mysql	5Gi	RWO	Retain	Bound	default/pvc-vol-mysql
longhorn-static		38d			
pv-vol-tomcat	1Gi	RWX	Retain	Bound	default/pvc-vol-tomcat
longhorn-static		34d			
pvfc	5Gi	RWO	Retain	Bound	default/fc-lun-claim
167d					

[root@kmaster01 ~]# **kubectl get pvc**

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS	AGE
data-kafka-0	Bound	kafka-pv	8Gi	RWX	longhorn-static	21d
data-kafka-zookeeper-0	Bound	kafka-zookeeper-pv	8Gi	RWX	longhorn-static	21d
fc-lun-claim	Bound	pvfc	5Gi	RWO		167d
fc-pv-volume	Bound	fc-pv-volume	10Gi	RWO	fc-vol	182d
fc-vol-claim	Bound	fc-vol	10Gi	RWO		171d
pv-testing	Bound	pv-testing	1Gi	RWO		47d
pvc-vol-mongo	Bound	pv-vol-mongo	1Gi	RWX	longhorn-static	35d
pvc-vol-mysql	Bound	pv-vol-mysql	5Gi	RWO	longhorn-static	38d
pvc-vol-tomcat	Bound	pv-vol-tomcat	1Gi	RWX	longhorn-static	34d

NOTA: cada PV debe estar en BOUND con su PVC. SINO NO ANDA

Directorios Importantes:

Kmaster:

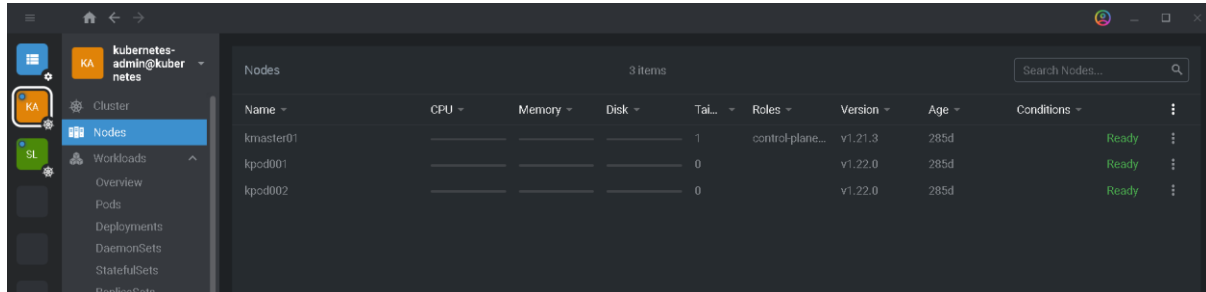
- /traefik --- Proxypass para ingresar a los micro (esto se movio al slnxdock02)
- /rancher ---- directorio para rancher y binario de Helm
- /k8s_home ---- directorio home para manifiestos de K8s
- /var/lib/docker/ --- directorio raiz de Docker
- /etc/kubernetes/ --- configuración y manifiestos de k8s
- /var/lib/kubelet/ --- configuración de kubelet
- /var/lib/etcd --- base de k8s

Kpod001 y Kpod002



- /var/lib/longhorn --- directorio de replica de LongHorn

Acceso por lens:



Para esto te pide las credenciales del cluster. Las mismas las encuentras en:

```
[root@kmaster01 ~]# cd .kube
[root@kmaster01 .kube]# ls -arlt
total 20
-rw----- 1 root root 5596 Aug 19 2021 config
drwxr-xr-x 3 root root 4096 Aug 19 2021 .
drwxr-x--- 4 root root 4096 Aug 19 2021 cache
dr-xr-x--- 8 root root 4096 May 30 17:50 ..
[root@kmaster01 .kube]# cat config
apiVersion: v1
clusters:
- cluster:
    certificate-authority-data:
    LS0tLS1CRUdJTiBDRVJUSUZJQ0FURSB0tLS0tCk1JSUM1ekNDQWMrZ0F3SUJBZ0lCQURBTklna3
    Foa2lHOXcwQkFRc0ZBREFWTVJNd0VRWURWUVFERXdwcmRXSmwKY201bGRHVnpNQjRYRF
    RJeE1EZ3hPVEV4TWpBek4xb1hEVE14TURneE56RXhNakF6TjFvd0ZURVRNQkVHQTFVRQpBe
    E1LYTNWVpYSnVaWFJsY3pDQ0FTSXdEUUVKS29aSWh2Y05BUUVCQlFBRGdnRVBIBRENDQVF
    vQ2dnRUJBTDIICkRvQ3JHaFFeajhIUUndCSjU0MEMzMEOzaUEzSmlGQU9zRUUyc3RlclFHQWti
    MU9SbnFkNIBUQ3ILQWwxZnloNTIKV2gxMn
```

Administracion Por Rancher

El acceso de rancher esta por los Nodos

<https://kpod001:31620/dashboard/auth/login>

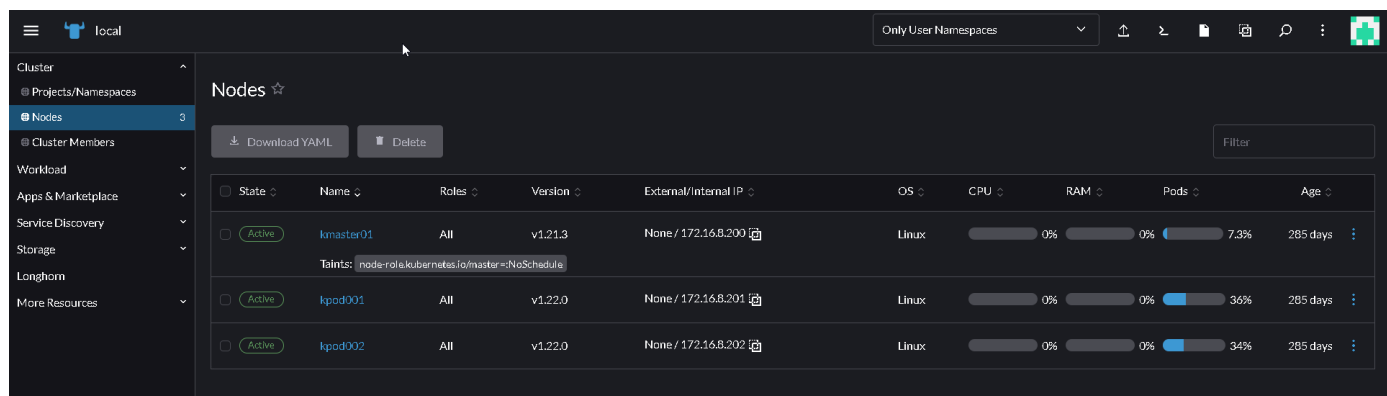
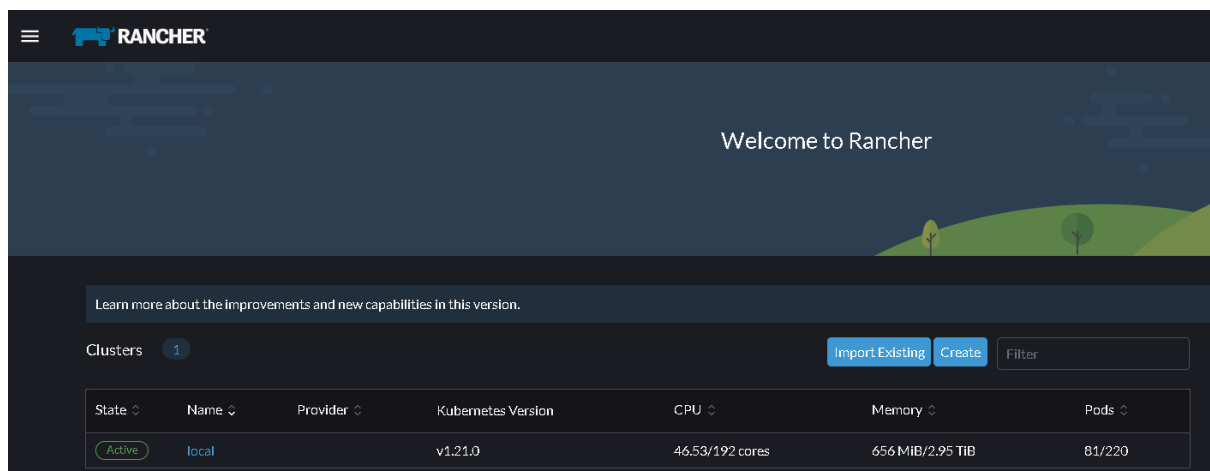
<https://kpod002:31620/dashboard/auth/login>

user: admin
pass: RGKCUyArsDhkodlAtLIY

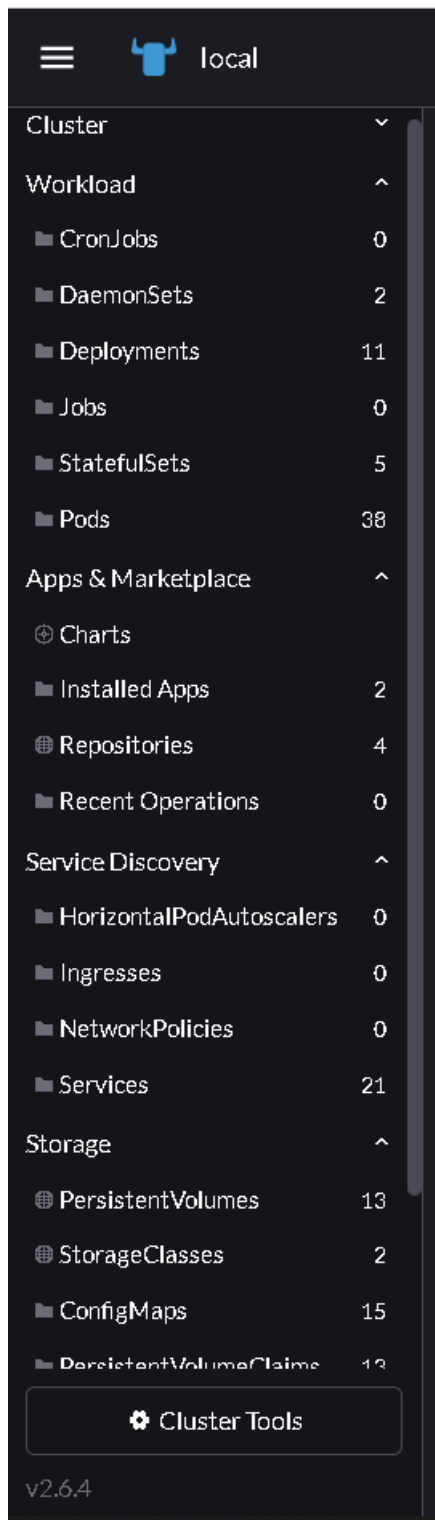
Ni bien te logueas, podés ver el acceso a todos tus clusters



Entrando en el “LOCAL”, que seria nuestro K8S



Adentro encontramos por secciones todos los objetos de nuestro cluster

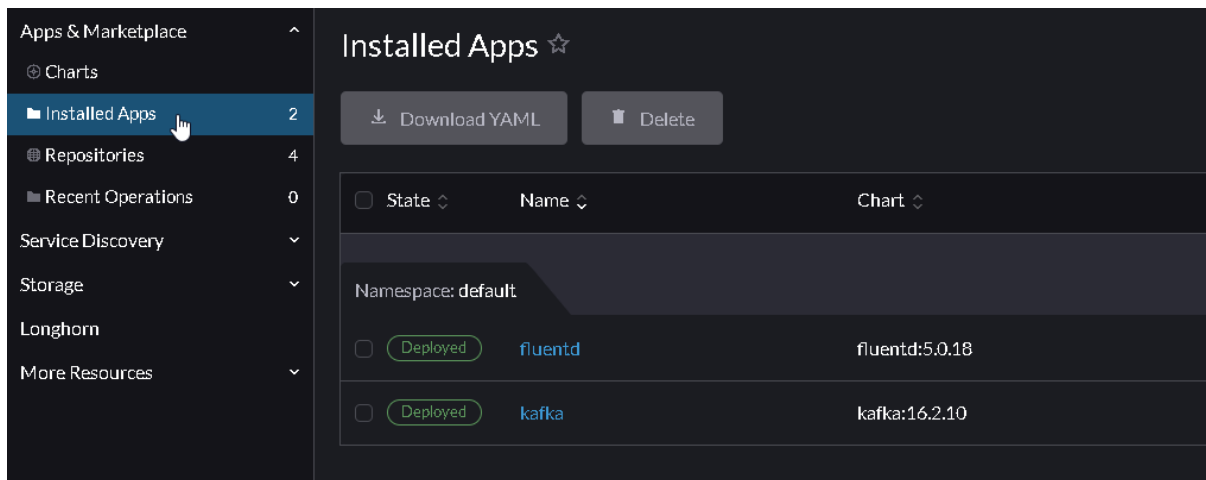


Alli encontramos en cluster:

- Proyectos/Namespace
- Nodos
- Deployments
- Pods
- Toda la parte de la tienda donde bajamos los charts
- Servicios
- Storage (los pv, pvc y configmaps)
- Ingress



Los Charts son micros desplegados en el cluster bajados con Helm desde un repositorio



Estos bajan los manifiestos para aplicar los cambios. Para esto tenemos que configurar los repo. Esto se puede hacer en rancher o por comando

```
[root@kmaster01 ~]# helm repo list
NAME      URL
rancher-stable https://releases.rancher.com/server-charts/stable
rancher-latest https://releases.rancher.com/server-charts/latest
jetstack   https://charts.jetstack.io
longhorn    https://charts.longhorn.io
bitnami     https://charts.bitnami.com/bitnami
```

Instalar un charts

```
helm install stable/prometheus --set server.service.type=NodePort
```

Descripcion del paquete

```
helm inspect prometheus
```

Buscar chart

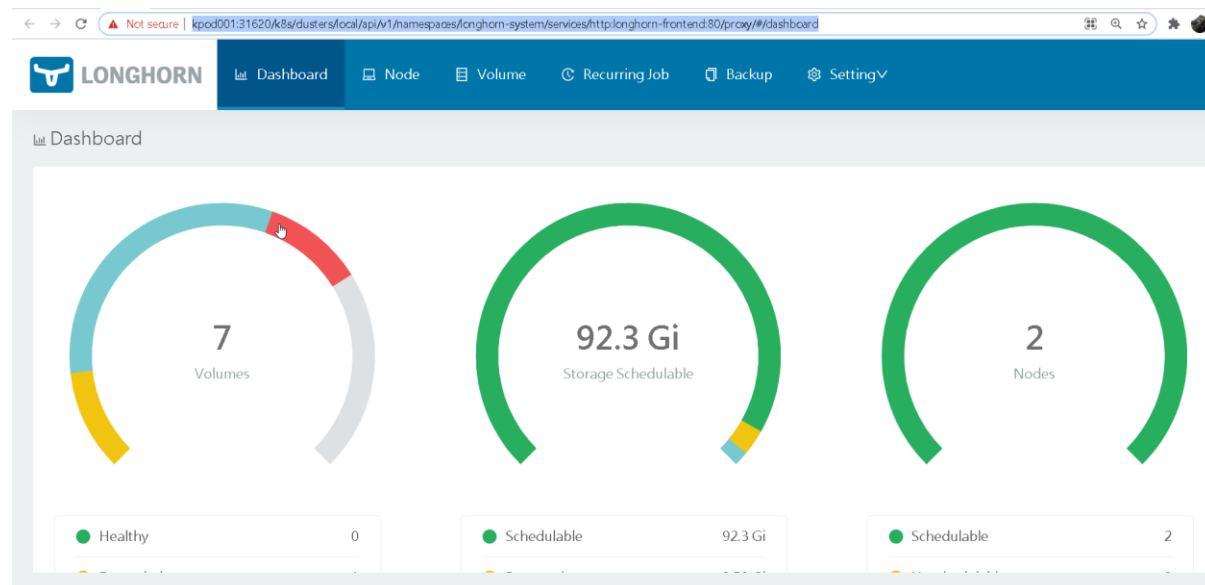
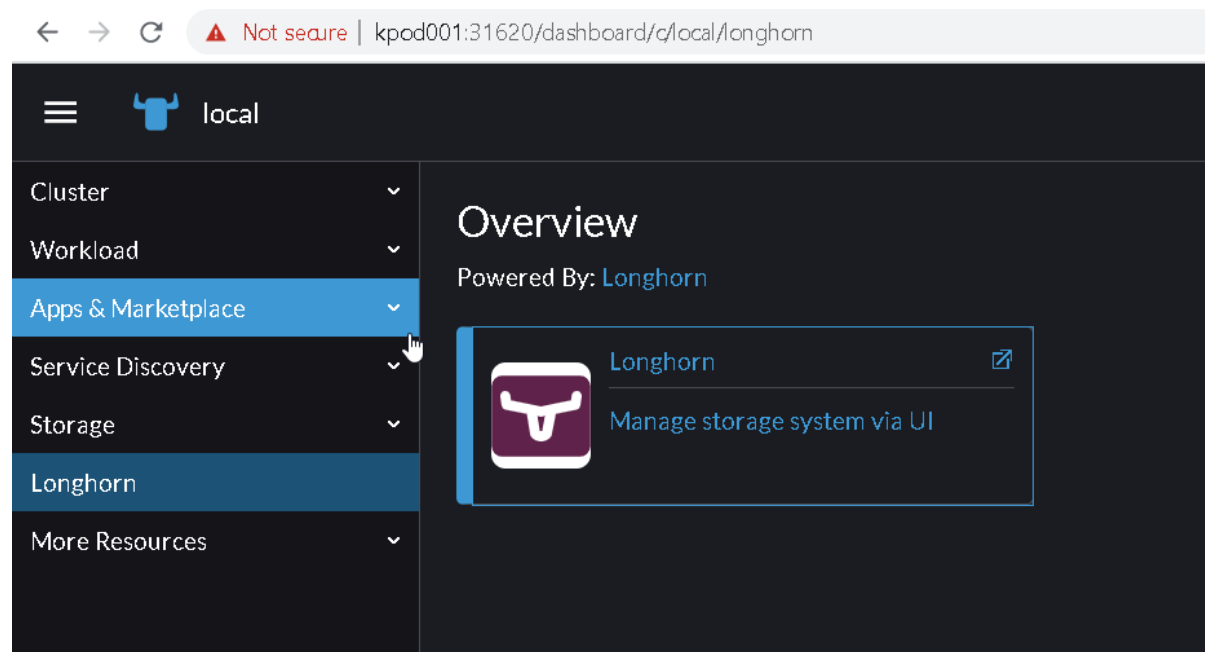
```
[root@kmaster01 deploy_kafka]# helm search repo kafka
```

NAME	CHART VERSION	APP VERSION	DESCRIPTION
bitnami/kafka	16.2.10	3.1.0	Apache Kafka is a distributed streaming platform...
bitnami/dataplatform-bp1	11.0.3	1.0.1	This Helm chart can be used for the automated d...
bitnami/dataplatform-bp2	12.0.3	1.0.1	This Helm chart can be used for the automated d...



LongHorn

Se accede desde Rancher





Accesos:

- Jenkins
 - El nuevo Jenkins productivo esta en:
 - <http://slnxdock02:8888/login?from=%2F>

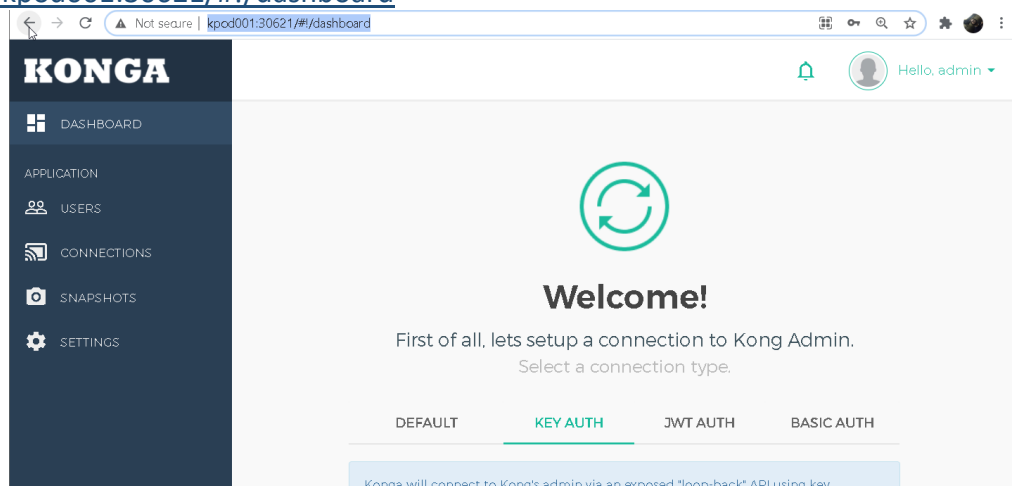


Welcome to Jenkins!

Sign in

☐ Keep me signed in

- Konga
 - <http://kpod001:30621/#!/dashboard>



- Rancher
- <https://kpod001:31620/dashboard/home>



← → ↻ ⚠ Not secure | kpod001:31620/dashboard/home 🔍 ☆ ⚙ 👤 ⋮

☰ **RANCHER** ⋮

Welcome to Rancher

Learn more about the improvements and new capabilities in this version. [What's new in 2.6](#)

Clusters 1 [Import Existing](#) [Create](#)

State	Name	Provider	Kubernetes Version	CPU	Memory	Pods
Active	local		v1.21.0	46.48/192 cores	540 MiB/2.95 TiB	85/220

Community Support ×

- [Docs](#)
- [Forums](#)
- [Cluster](#)

- Traefik
 - <http://slnxdock02:8080/dashboard/#/>

← → × ⚠ Not secure | slnxdock02:8080/dashboard/#/ traefik 2.5.7 Dashboard HTTP TCP UDP Dark theme ⓘ

→ Entrypoints

TRAEFIK
:8080

WEB
:80

⊕ HTTP

Routers

Explore →

Success	100%	10
Warnings	0%	0
Errors	0%	0

Services

Explore →

Success	100%	11
Warnings	0%	0
Errors	0%	0

Middlewares



- Nexus PROD

- <http://slnxdock02:8081/#admin/security/user>

Not secure | slnxdock02:8081/#admin/repository/repositories

Sonatype Nexus Repository Manager
OSS 3.38.0-01

Administration

- Repository
- Repositories**
- Blob Stores
- Cleanup Policies
- Content Selectors
- Proprietary Repositories
- Routing Rules
- Security
 - Privileges
 - Roles
 - Users

Repositories Manage repositories

Create repository Filter

Name	Type	Format	URL	Health check
d...	hosted	docker	O...	copy
d...	proxy	docker	O...	Analyze
m...	proxy	maven2	O...	Analyze
m...	group	maven2	O...	copy
m...	hosted	maven2	O...	copy
m...	hosted	maven2	O...	copy
m...	proxy	docker	O...	Analyze
n...	group	nuget	O...	copy
n...	hosted	nuget	O...	copy
n...	proxy	nuget	O...	Analyze
r...	group	docker	O...	copy

Todas las claves estan en el Keepas