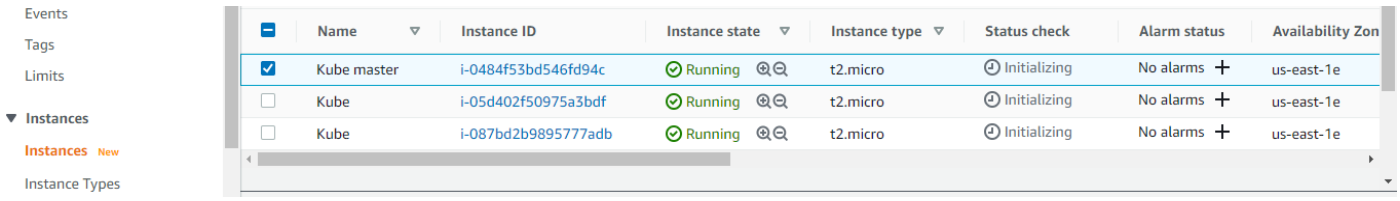
comnnads:

1. sudo su –
2. yum install docker -y
3. docker info

➢ instances are setup



➔ setting up the master

■ installation of docker

➢ connect master to putty.

sudo su –

yum install docker -y

```
You need to be root to perform this command.
[ec2-user@ip-172-31-52-186 ~]$ sudo su -
[root@ip-172-31-52-186 ~]# yum install docker -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                | 3.7 kB  00:00:00
Resolving Dependencies
--> Running transaction check
---> Package docker.x86_64 0:19.03.13ce-1.amzn2 will be installed
--> Processing Dependency: runc >= 1.0.0 for package: docker-19.03.13ce-1.amzn2.x86_64
--> Processing Dependency: containerd >= 1.3.2 for package: docker-19.03.13ce-1.amzn2.x86_64
--> Processing Dependency: pigz for package: docker-19.03.13ce-1.amzn2.x86_64
--> Processing Dependency: libcgroup for package: docker-19.03.13ce-1.amzn2.x86_64
--> Running transaction check
---> Package containerd.x86_64 0:1.4.1-2.amzn2 will be installed
---> Package libcgroup.x86_64 0:0.41-21.amzn2 will be installed
---> Package pigz.x86_64 0:2.3.4-1.amzn2.0.1 will be installed
---> Package runc.x86_64 0:1.0.0-0.1.20200826.gitff819c7.amzn2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

================================================================================
 Package          Arch         Version                          Repository     Size
================================================================================
Installing:
 docker           x86_64       19.03.13ce-1.amzn2               amzn2extra-docker  37 M
Installing for dependencies:
 containerd       x86_64       1.4.1-2.amzn2                    amzn2extra-docker  24 M
```

➢ enable the docker services
systemctl enable docker –now
docker info

```
[root@ip-172-31-52-186 ~]# systemctl enable docker --now
Created symlink from /etc/systemd/system/multi-user.target.wants/docker.service to /usr/lib/systemd/syste
m/docker.service.
```

```
m/docker.service.
[root@ip-172-31-52-186 ~]# docker info
Client:
 Debug Mode: false

Server:
 Containers: 0
  Running: 0
  Paused: 0
  Stopped: 0
 Images: 0
 Server Version: 19.03.13-ce
 Storage Driver: overlay2
  Backing Filesystem: xfs
  Supports d_type: true
  Native Overlay Diff: true
 Logging Driver: json-file
 Cgroup Driver: cgroupfs
 Plugins:
  Volume: local
```

■ installation of kubeadm

➢ yum install kubeadm

```
[root@ip-172-31-52-186 ~]# yum install kubeadm
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                                | 3.7 kB  00:00:00
No package kubeadm available.
Error: Nothing to do
```

fails, have to configure yum again which includes the package for kubeadm.

➢ yum repolist

```
[root@ip-172-31-52-186 ~]# yum repolist
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
repo id                                                   repo name                                   status
!amzn2-core/2/x86_64                                      Amazon Linux 2 core repository              23,094
amzn2extra-docker/2/x86_64                                Amazon Extras repo for docker                  36
repolist: 23,130
[root@ip-172-31-52-186 ~]#
```

currently it has 23 k packages , lets configure for kubeadm

search in k8 docs

vi /etc/yum.repos.d/kubernetes.repo
[root@ip-172-31-89-40 ~]# cat /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

- vi /etc/yum.repos.d/kubernetes.repo
- cat /etc/yum.repos.d/kubernetes.repo

```
[root@ip-172-31-52-186 yum.repos.d]# vi /etc/yum.repos.d/kubernetes.repo
[root@ip-172-31-52-186 yum.repos.d]# cat /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
```

- yum repolist

```
[root@ip-172-31-52-186 yum.repos.d]# yum repolist
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                                | 3.7 kB  00:00:00
kubernetes/x86_64/signature                                               |  844 B  00:00:00
Retrieving key from https://packages.cloud.google.com/yum/doc/yum-key.gpg
Importing GPG key 0xA7317B0F:
 Userid     : "Google Cloud Packages Automatic Signing Key <gc-team@google.com>"
 Fingerprint: d0bc 747f d8ca f711 7500 d6fa 3746 c208 a731 7b0f
 From       : https://packages.cloud.google.com/yum/doc/yum-key.gpg
Is this ok [y/N]: y
Retrieving key from https://packages.cloud.google.com/yum/doc/rpm-package-key.gpg
kubernetes/x86_64/signature                                               | 1.4 kB  00:00:02 !!!
kubernetes/x86_64/primary                                                 |  85 kB  00:00:00
kubernetes                                                                             624/624
repo id                                          repo name                                 status
amzn2-core/2/x86_64                              Amazon Linux 2 core repository            23,094
amzn2extra-docker/2/x86_64                       Amazon Extras repo for docker                 36
kubernetes/x86_64                                Kubernetes                                15+609
repolist: 23,145
[root@ip-172-31-52-186 yum.repos.d]#
```

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

```
[root@ip-172-31-52-186 yum.repos.d]# yum install -y kubelet kubeadm kubectl --disableexcludes=kubernetes
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Resolving Dependencies
--> Running transaction check
---> Package kubeadm.x86_64 0:1.20.2-0 will be installed
--> Processing Dependency: kubernetes-cni >= 0.8.6 for package: kubeadm-1.20.2-0.x86_64

Installed:
  kubeadm.x86_64 0:1.20.2-0                kubectl.x86_64 0:1.20.2-0                kubelet.x86_64 0:1.20.2-0

Dependency Installed:
  conntrack-tools.x86_64 0:1.4.4-5.amzn2.2                  cri-tools.x86_64 0:1.13.0-0
  ebtables.x86_64 0:2.0.10-16.amzn2.0.1                     kubernetes-cni.x86_64 0:0.8.7-0
  libnetfilter_cthelper.x86_64 0:1.0.0-10.amzn2.1           libnetfilter_cttimeout.x86_64 0:1.0.0-6.amzn2.1
  libnetfilter_queue.x86_64 0:1.0.2-2.amzn2.0.2             socat.x86_64 0:1.7.3.2-2.amzn2.0.1

Complete!
```

- checking the status of kubelet and enabling it

  systemctl status kubelet

  systemctl enable kubelet –now

```
[root@ip-172-31-52-186 yum.repos.d]# systemctl status kubelet
● kubelet.service - kubelet: The Kubernetes Node Agent
   Loaded: loaded (/usr/lib/systemd/system/kubelet.service; disabled; vendor preset: disabled)
  Drop-In: /usr/lib/systemd/system/kubelet.service.d
           └─10-kubeadm.conf
   Active: inactive (dead)
     Docs: https://kubernetes.io/docs/

[root@ip-172-31-52-186 yum.repos.d]# systemctl enable kubelet --now
Created symlink from /etc/systemd/system/multi-user.target.wants/kubelet.service to /usr/lib/systemd/system/kubelet.service.
[root@ip-172-31-52-186 yum.repos.d]# systemctl status kubelet

[root@ip-172-31-52-186 yum.repos.d]# systemctl status kubelet
● kubelet.service - kubelet: The Kubernetes Node Agent
   Loaded: loaded (/usr/lib/systemd/system/kubelet.service; enabled; vendor preset: disabled)
  Drop-In: /usr/lib/systemd/system/kubelet.service.d
           └─10-kubeadm.conf
   Active: activating (auto-restart) (Result: exit-code) since Wed 2021-02-03 09:53:09 UTC; 5s ago
     Docs: https://kubernetes.io/docs/
  Process: 1956 ExecStart=/usr/bin/kubelet $KUBELET_KUBECONFIG_ARGS $KUBELET_CONFIG_ARGS $KUBELET_KUBEADM_ARGS $KUBELET_EXTRA_ARGS (cod
e=exited, status=255)
 Main PID: 1956 (code=exited, status=255)
```

- inside master, a lot of programs are running and each have their own image, so one way is to pull one by one manually or use kubeadm to config all automatically

  kubeadm config images pull

```
[root@ip-172-31-52-186 ~]# kubeadm config images pull
[config/images] Pulled k8s.gcr.io/kube-apiserver:v1.20.2
[config/images] Pulled k8s.gcr.io/kube-controller-manager:v1.20.2
[config/images] Pulled k8s.gcr.io/kube-scheduler:v1.20.2
[config/images] Pulled k8s.gcr.io/kube-proxy:v1.20.2
[config/images] Pulled k8s.gcr.io/pause:3.2
[config/images] Pulled k8s.gcr.io/etcd:3.4.13-0
[config/images] Pulled k8s.gcr.io/coredns:1.7.0
[root@ip-172-31-52-186 ~]#
```

- have to launch the container for each of the images since 7 images are pulled but none are working. lets verify. docker images – docker ps

```
[root@ip-172-31-52-186 ~]# docker images
REPOSITORY                           TAG                 IMAGE ID            CREATED             SIZE
k8s.gcr.io/kube-proxy                v1.20.2             43154ddb57a8        2 weeks ago         118MB
k8s.gcr.io/kube-apiserver            v1.20.2             a8c2fdb8bf76        2 weeks ago         122MB
k8s.gcr.io/kube-controller-manager   v1.20.2             a27166429d98        2 weeks ago         116MB
k8s.gcr.io/kube-scheduler            v1.20.2             ed2c44fbdd78        2 weeks ago         46.4MB
k8s.gcr.io/etcd                      3.4.13-0            0369cf4303ff        5 months ago        253MB
k8s.gcr.io/coredns                   1.7.0               bfe3a36ebd25        7 months ago        45.2MB
k8s.gcr.io/pause                     3.2                 80d28bedfe5d        11 months ago       683kB
[root@ip-172-31-52-186 ~]# docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED             STATUS              PORTS               NAMES
[root@ip-172-31-52-186 ~]#
```

lets run all of them at once.

kubeadm init : run the container for each images

```
[root@ip-172-31-52-186 ~]# kubeadm init
[init] Using Kubernetes version: v1.20.2
[preflight] Running pre-flight checks
        [WARNING IsDockerSystemdCheck]: detected "cgroupfs" as the Docker cgroup driver. The recommended driver is "syst
llow the guide at https://kubernetes.io/docs/setup/cri/
        [WARNING FileExisting-tc]: tc not found in system path
error execution phase preflight: [preflight] Some fatal errors occurred:
        [ERROR NumCPU]: the number of available CPUs 1 is less than the required 2
        [ERROR Mem]: the system RAM (983 MB) is less than the minimum 1700 MB
        [ERROR FileContent--proc-sys-net-bridge-bridge-nf-call-iptables]: /proc/sys/net/bridge/bridge-nf-call-iptables c
 set to 1
[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
```

it fails. it performs pre-flight checks. lets see the errors and solve them one by one.

kubectl run the pods => where? inside the minikube => container will have the IP => who provide the IP name and network name(CIDR) ? Master

Therefore, need to give the IP range in the command. ask help

kubeadm init -h

```
e-1")
      --node-name string                     Specify the node name.
      --pod-network-cidr string              Specify range of IP addresses for the pod network. If set, the control plan
e will automatically allocate CIDRs for every node.
      --service-cidr string                  Use alternative range of IP address for service VIPs. (default "10.96.0.0/1
2")
```

kubeadm init --pod-network-cidr=10.240.0.0/16

```
[root@ip-172-31-52-186 ~]# kubeadm init --pod-network-cidr=10.240.0.0/16
[init] Using Kubernetes version: v1.20.2
[preflight] Running pre-flight checks
        [WARNING IsDockerSystemdCheck]: detected "cgroupfs" as the Docker cgroup driver. The recommended driver is "systemd". Please fo
llow the guide at https://kubernetes.io/docs/setup/cri/
        [WARNING FileExisting-tc]: tc not found in system path
error execution phase preflight: [preflight] Some fatal errors occurred:
        [ERROR NumCPU]: the number of available CPUs 1 is less than the required 2
        [ERROR Mem]: the system RAM (983 MB) is less than the minimum 1700 MB
        [ERROR FileContent--proc-sys-net-bridge-bridge-nf-call-iptables]: /proc/sys/net/bridge/bridge-nf-call-iptables contents are not
 set to 1
[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
```

- 1st: [WARNING IsDockerSystemdCheck]: we have to use system instead of cgroups.

  Trouble Shooting:

  vi /etc/docker/daemon.json

  {
  "exec-opts": ["native.cgroupdriver=systemd"]
  }

  cat /etc/docker/daemon.json

```
[root@ip-172-31-52-186 docker]# vi daemon.json
[root@ip-172-31-52-186 docker]# cat /etc/docker/daemon.json
{
  "exec-opts": ["native.cgroupdriver=systemd"]
}

[root@ip-172-31-52-186 docker]#
```

to make the changes : restart the docker

systemctl restart docker

docker info | grep Driver - check the

```
[root@ip-172-31-52-186 docker]# systemctl restart docker
[root@ip-172-31-52-186 docker]# docker info | grep Driver
 Storage Driver: overlay2
 Logging Driver: json-file
 Cgroup Driver: systemd
WARNING: bridge-nf-call-iptables is disabled
WARNING: bridge-nf-call-ip6tables is disabled
```

- again checking

```
[root@ip-172-31-52-186 ~]#  kubeadm init --pod-network-cidr=10.240.0.0/16
[init] Using Kubernetes version: v1.20.2
[preflight] Running pre-flight checks
        [WARNING FileExisting-tc]: tc not found in system path
error execution phase preflight: [preflight] Some fatal errors occurred:
        [ERROR NumCPU]: the number of available CPUs 1 is less than the required 2
        [ERROR Mem]: the system RAM (983 MB) is less than the minimum 1700 MB
        [ERROR FileContent--proc-sys-net-bridge-bridge-nf-call-iptables]: /proc/sys/net/bridge/bridge-nf-call-iptables contents are not set to 1
[preflight] If you know what you are doing, you can make a check non-fatal with `--ignore-preflight-errors=...`
```

2nd error: [WARNING FileExisting-tc]: tc not found in system path

Trouble Shooting:

- need to yum install iproute-tc

```
To see the stack trace of this error execute with --v=5 or higher
[root@ip-172-31-52-186 ~]# yum install iproute-tc
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core                                                               | 3.7 kB  00:00:00
Resolving Dependencies
--> Running transaction check
---> Package iproute-tc.x86_64 0:4.15.0-1.amzn2.0.4 will be installed
```

- again checking

```
[root@ip-172-31-52-186 ~]#  kubeadm init --pod-network-cidr=10.240.0.0/16
[init] Using Kubernetes version: v1.20.2
[preflight] Running pre-flight checks
error execution phase preflight: [preflight] Some fatal errors occurred:
        [ERROR NumCPU]: the number of available CPUs 1 is less than the required 2
        [ERROR Mem]: the system RAM (983 MB) is less than the minimum 1700 MB
        [ERROR FileContent--proc-sys-net-bridge-bridge-nf-call-iptables]: /proc/sys/net/bridge/bridge-nf-call-iptables contents are not set to 1
[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
```

- ⇨ 3rd error: FileContent—proc-sys-net-bridge-bridge-nf-call-iptables.

Trouble Shooting:

vim /etc/sysctl.d/k8s.conf

net.bridge.bridge-nf-call-ip6tables = 1
net.bridge.bridge-nf-call-iptables = 1

cat /etc/sysctl.d/k8s.conf

sysctl --system

```
[root@ip-172-31-52-186 ~]# vim /etc/sysctl.d/k8s.conf
[root@ip-172-31-52-186 ~]# cat  /etc/sysctl.d/k8s.conf
net.bridge.bridge-nf-call-ip6tables = 1
net.bridge.bridge-nf-call-iptables = 1

[root@ip-172-31-52-186 ~]# sysctl --system
* Applying /etc/sysctl.d/00-defaults.conf ...
kernel.printk = 8 4 1 7
kernel.panic = 30
net.ipv4.neigh.default.gc_thresh1 = 0
net.ipv6.neigh.default.gc_thresh1 = 0
net.ipv4.neigh.default.gc_thresh2 = 15360
net.ipv6.neigh.default.gc_thresh2 = 15360
net.ipv4.neigh.default.gc_thresh3 = 16384
net.ipv6.neigh.default.gc_thresh3 = 16384
* Applying /usr/lib/sysctl.d/00-system.conf ...
```

sysctl -a | grep bridge-bridge-nf-call

```
[root@ip-172-31-52-186 ~]# sysctl -a | grep bridge-bridge-nf-call
sysctl: reading key "net.ipv6.conf.all.stable_secret"
sysctl: reading key "net.ipv6.conf.default.stable_secret"
sysctl: reading key "net.ipv6.conf.docker0.stable_secret"
sysctl: reading key "net.ipv6.conf.eth0.stable_secret"
sysctl: reading key "net.ipv6.conf.lo.stable_secret"
```

- ⇨ 4th error: [ERROR NumCPU]: the number of available CPUs 1 is less than the required 2 : resources requirement

Trouble Shooting:

skip these errors for now

now checking

```
[root@ip-172-31-52-186 ~]# kubeadm init --pod-network-cidr=10.240.0.0/16     --ignore-preflight-errors=NumCPU    --ignore-preflight-erro
rs=Mem
```

```
  sudo chown $(id -u):$(id -g) $HOME/.kube/config

Alternatively, if you are the root user, you can run:

  export KUBECONFIG=/etc/kubernetes/admin.conf

You should now deploy a pod network to the cluster.
Run "kubectl apply -f [podnetwork].yaml" with one of the options listed at:
  https://kubernetes.io/docs/concepts/cluster-administration/addons/

Then you can join any number of worker nodes by running the following on each as root:

kubeadm join 172.31.52.186:6443 --token y3934u.hvqcghz22oucu6rs \
    --discovery-token-ca-cert-hash sha256:c140d470d1bd13e48adf450bae48aae0d88a94f42472a5aa06842302d95f295e
```

Yes, it works.

and the last two line is the token which is required if any worker node wants to connect to this master.

- docker ps : u will now see every program of kubeadm is running in the containers.

```
[root@ip-172-31-52-186 ~]# docker ps
CONTAINER ID        IMAGE                  COMMAND                  CREATED             STATUS              PORTS               NAMES
8e5df0ebcedf        43154ddb57a8           "/usr/local/bin/kube…"   26 minutes ago      Up 26 minutes                           k8s_kub
e-proxy_kube-proxy-gbwqc_kube-system_8000bce8-5f04-4207-a897-18844d343b3d_0
b9f662122cd5        k8s.gcr.io/pause:3.2   "/pause"                 26 minutes ago      Up 26 minutes                           k8s_POD
_kube-proxy-gbwqc_kube-system_8000bce8-5f04-4207-a897-18844d343b3d_0
a21c3a7c0643        0369cf4303ff           "etcd --advertise-cl…"   27 minutes ago      Up 27 minutes                           k8s_etc
d_etcd-ip-172-31-52-186.ec2.internal_kube-system_2aa0c0bcbe890eecb10d62001d803afe_0
01aaa3c97605        ed2c44fbdd78           "kube-scheduler --au…"   27 minutes ago      Up 27 minutes                           k8s_kub
e-scheduler_kube-scheduler-ip-172-31-52-186.ec2.internal_kube-system_69cd289b4ed80ced4f95a59ff60fa102_0
a2b23553382f        a27166429d98           "kube-controller-man…"   27 minutes ago      Up 27 minutes                           k8s_kub
e-controller-manager_kube-controller-manager-ip-172-31-52-186.ec2.internal_kube-system_38662a5a33a1b9424a79ea41bae7875b_0
0716431aa7aa        a8c2fdb8bf76           "kube-apiserver --ad…"   27 minutes ago      Up 27 minutes                           k8s_kub
e-apiserver_kube-apiserver-ip-172-31-52-186.ec2.internal_kube-system_696c7fb5e3e41af1c15fc17a78319566_0
3198cf4aecec        k8s.gcr.io/pause:3.2   "/pause"                 27 minutes ago      Up 27 minutes                           k8s_POD
_kube-controller-manager-ip-172-31-52-186.ec2.internal_kube-system_38662a5a33a1b9424a79ea41bae7875b_0
56337285be1b        k8s.gcr.io/pause:3.2   "/pause"                 27 minutes ago      Up 27 minutes                           k8s_POD
_kube-apiserver-ip-172-31-52-186.ec2.internal_kube-system_696c7fb5e3e41af1c15fc17a78319566_0
96a4e6cb1bb8        k8s.gcr.io/pause:3.2   "/pause"                 27 minutes ago      Up 27 minutes                           k8s_POD
_etcd-ip-172-31-52-186.ec2.internal_kube-system_2aa0c0bcbe890eecb10d62001d803afe_0
4b8b9eb7ca8e        k8s.gcr.io/pause:3.2   "/pause"                 27 minutes ago      Up 27 minutes                           k8s_POD
_kube-scheduler-ip-172-31-52-186.ec2.internal_kube-system_69cd289b4ed80ced4f95a59ff60fa102_0
```

kubeadm successfully setup……

- let's see the status of pods

kubectl get pods

```
[root@ip-172-31-52-186 ~]# kubectl get pods
The connection to the server localhost:8080 was refused - did you specify the right host or port?
[root@ip-172-31-52-186 ~]#
```

fails. WHY ?

Ans. kubectl contact to API : therefore should know the IP, user and port of it.

Trouble Shooting:

- asking for help: kubectl get pods -h

```
Your Kubernetes control-plane has initialized successfully!

To start using your cluster, you need to run the following as a regular user:

  mkdir -p $HOME/.kube
  sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
  sudo chown $(id -u):$(id -g) $HOME/.kube/config

Alternatively, if you are the root user, you can run:

  export KUBECONFIG=/etc/kubernetes/admin.conf
```

To start using your cluster, you need to run the following as a regular user:

mkdir -p $HOME/.kube

sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config

sudo chown $(id -u):$(id -g) $HOME/.kube/config

have to run above three commands

```
[root@ip-172-31-52-186 ~]# mkdir -p $HOME/.kube
[root@ip-172-31-52-186 ~]# sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
[root@ip-172-31-52-186 ~]# sudo chown $(id -u):$(id -g) $HOME/.kube/config
[root@ip-172-31-52-186 ~]#
```

```
[root@ip-172-31-52-186 ~]#
[root@ip-172-31-52-186 ~]# kubectl get pods
No resources found in default namespace.
[root@ip-172-31-52-186 ~]#
```

working now……..

- systemctl status kubectl

```
[root@ip-172-31-52-186 ~]# systemctl status kubelet
● kubelet.service - kubelet: The Kubernetes Node Agent
   Loaded: loaded (/usr/lib/systemd/system/kubelet.service; enabled; vendor preset: disabled)
  Drop-In: /usr/lib/systemd/system/kubelet.service.d
           └─10-kubeadm.conf
   Active: active (running) since Wed 2021-02-03 12:16:15 UTC; 24min ago
     Docs: https://kubernetes.io/docs/
 Main PID: 8923 (kubelet)
```

- kubectl get nodes: currently it has only one i.e. master

```
[root@ip-172-31-52-186 ~]# kubectl get nodes
NAME                          STATUS     ROLES                  AGE   VERSION
ip-172-31-52-186.ec2.internal   NotReady   control-plane,master   29m   v1.20.2
[root@ip-172-31-52-186 ~]#
```

- the above name is same as the hostname

```
[root@ip-172-31-52-186 ~]# hostname
ip-172-31-52-186.ec2.internal
[root@ip-172-31-52-186 ~]#
```

master setup is done……