Test report - Arktos deployment with Mizar CNI

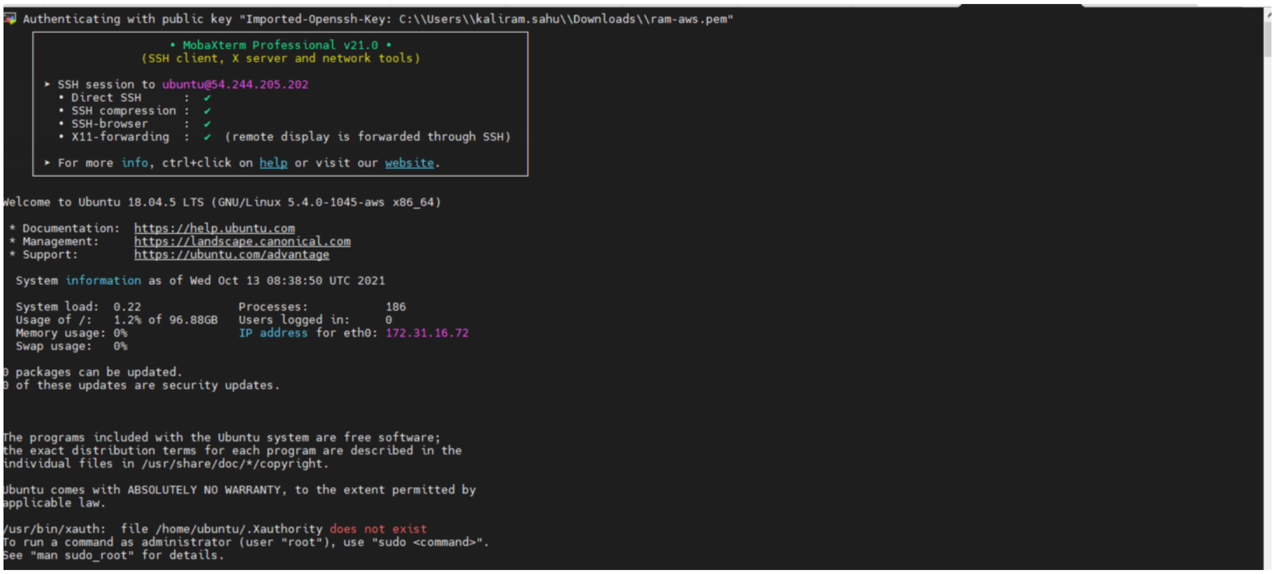
Date:13-Oct-21

Create an instance on AWS

preferred OS is **Ubuntu 18.04**. If you are using AWS, the recommended instance size is `t2.2xlarge` and the storage size is `128GB` or more.

SSH instance with credentials:

`output:`



```
Authenticating with public key "Imported-Openssh-Key: C:\\Users\\kaliram.sahu\\Downloads\\ram-aws.pem"
    ┌─────────────────────────────────────────────────────────────────────┐
    │                     • MobaXterm Professional v21.0 •                │
    │                 (SSH client, X server and network tools)            │
    │                                                                     │
    │ ➤ SSH session to ubuntu@54.244.205.202                              │
    │   • Direct SSH      :  ✓                                            │
    │   • SSH compression :  ✓                                            │
    │   • SSH-browser     :  ✓                                            │
    │   • X11-forwarding  :  ✓  (remote display is forwarded through SSH) │
    │                                                                     │
    │ ➤ For more info, ctrl+click on help or visit our website.           │
    └─────────────────────────────────────────────────────────────────────┘

Welcome to Ubuntu 18.04.5 LTS (GNU/Linux 5.4.0-1045-aws x86_64)

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

  System information as of Wed Oct 13 08:38:50 UTC 2021

  System load:  0.22               Processes:           186
  Usage of /:   1.2% of 96.88GB   Users logged in:     0
  Memory usage: 0%                 IP address for eth0: 172.31.16.72
  Swap usage:   0%

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



The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

/usr/bin/xauth:  file /home/ubuntu/.Xauthority does not exist
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```

Step 1:- Check the kernel version and update if version below 5.6:

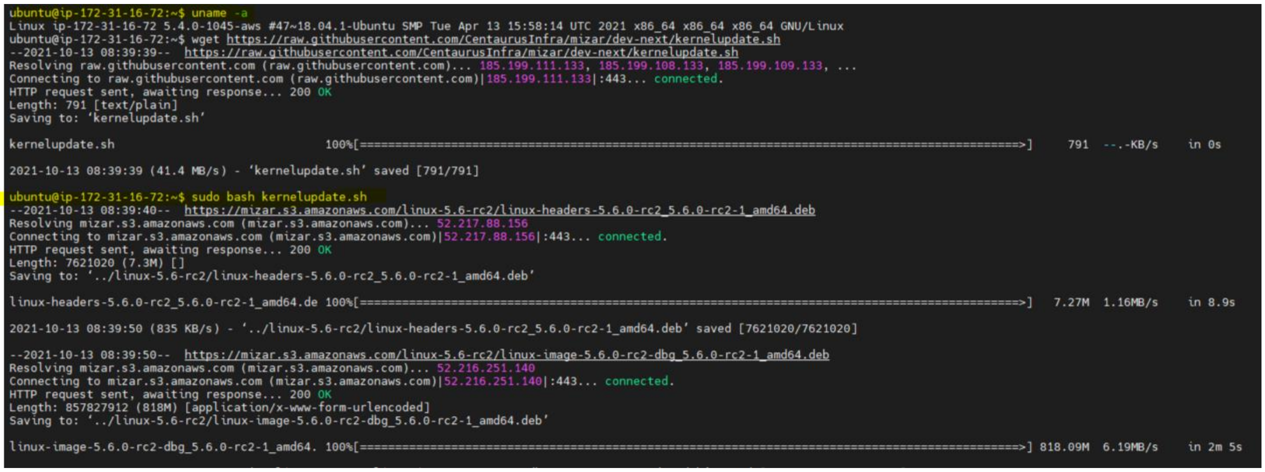
```
uname -a
```

Update the kernel if the kernel version is below `5.6.0-rc2`

```
wget https://raw.githubusercontent.com/CentaurusInfra/mizar/dev-next/kernelupdate.sh
```

```
sudo bash kernelupdate.sh
```

output:



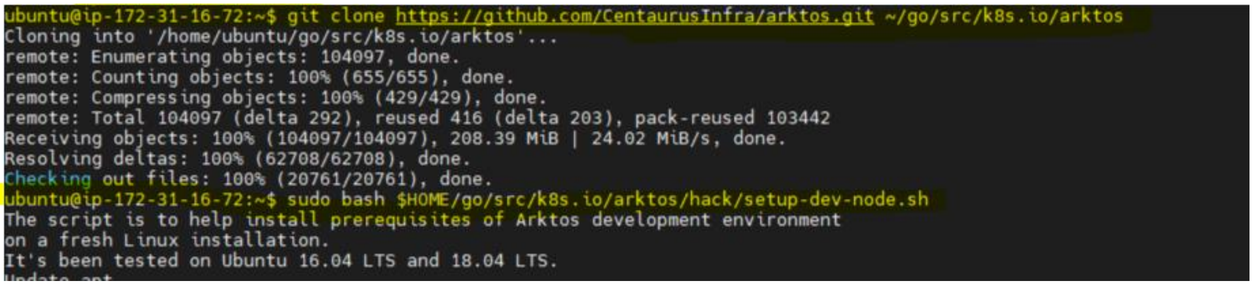
```
ubuntu@ip-172-31-16-72:~$ uname -a
Linux ip-172-31-16-72 5.4.0-1045-aws #47~18.04.1-Ubuntu SMP Tue Apr 13 15:58:14 UTC 2021 x86_64 x86_64 x86_64 GNU/Linux
ubuntu@ip-172-31-16-72:~$ wget https://raw.githubusercontent.com/CentaurusInfra/mizar/dev-next/kernelupdate.sh
--2021-10-13 08:39:39--  https://raw.githubusercontent.com/CentaurusInfra/mizar/dev-next/kernelupdate.sh
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.111.133, 185.199.108.133, 185.199.109.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.111.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 791 [text/plain]
Saving to: ‘kernelupdate.sh’

kernelupdate.sh                         100%[===========================================================================================>]     791  --.-KB/s    in 0s

2021-10-13 08:39:39 (41.4 MB/s) - ‘kernelupdate.sh’ saved [791/791]

ubuntu@ip-172-31-16-72:~$ sudo bash kernelupdate.sh
--2021-10-13 08:39:40--  https://mizar.s3.amazonaws.com/linux-5.6-rc2/linux-headers-5.6.0-rc2_5.6.0-rc2-1_amd64.deb
Resolving mizar.s3.amazonaws.com (mizar.s3.amazonaws.com)... 52.217.88.156
Connecting to mizar.s3.amazonaws.com (mizar.s3.amazonaws.com)|52.217.88.156|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7621020 (7.3M) []
Saving to: ‘../linux-5.6-rc2/linux-headers-5.6.0-rc2_5.6.0-rc2-1_amd64.deb’

linux-headers-5.6.0-rc2_5.6.0-rc2-1_amd64.de 100%[===========================================================================================>]   7.27M  1.16MB/s    in 8.9s

2021-10-13 08:39:50 (835 KB/s) - ‘../linux-5.6-rc2/linux-headers-5.6.0-rc2_5.6.0-rc2-1_amd64.deb’ saved [7621020/7621020]

--2021-10-13 08:39:50--  https://mizar.s3.amazonaws.com/linux-5.6-rc2/linux-image-5.6.0-rc2-dbg_5.6.0-rc2-1_amd64.deb
Resolving mizar.s3.amazonaws.com (mizar.s3.amazonaws.com)... 52.216.251.140
Connecting to mizar.s3.amazonaws.com (mizar.s3.amazonaws.com)|52.216.251.140|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 857827912 (818M) [application/x-www-form-urlencoded]
Saving to: ‘../linux-5.6-rc2/linux-image-5.6.0-rc2-dbg_5.6.0-rc2-1_amd64.deb’

linux-image-5.6.0-rc2-dbg_5.6.0-rc2-1_amd64. 100%[===========================================================================================>] 818.09M  6.19MB/s    in 2m 5s
```

Step 2 :- Clone the Arktos repository and install the required dependencies:

```
git clone https://github.com/CentaurusInfra/arktos.git ~/go/src/k8s.io/arktos.git
```

```
sudo bash $HOME/go/src/k8s.io/arktos/hack/setup-dev-node.sh
```

output:



```
ubuntu@ip-172-31-16-72:~$ git clone https://github.com/CentaurusInfra/arktos.git ~/go/src/k8s.io/arktos
Cloning into '/home/ubuntu/go/src/k8s.io/arktos'...
remote: Enumerating objects: 104097, done.
remote: Counting objects: 100% (655/655), done.
remote: Compressing objects: 100% (429/429), done.
remote: Total 104097 (delta 292), reused 416 (delta 203), pack-reused 103442
Receiving objects: 100% (104097/104097), 208.39 MiB | 24.02 MiB/s, done.
Resolving deltas: 100% (62708/62708), done.
Checking out files: 100% (20761/20761), done.
ubuntu@ip-172-31-16-72:~$ sudo bash $HOME/go/src/k8s.io/arktos/hack/setup-dev-node.sh
The script is to help install prerequisites of Arktos development environment
on a fresh Linux installation.
It's been tested on Ubuntu 16.04 LTS and 18.04 LTS.
```

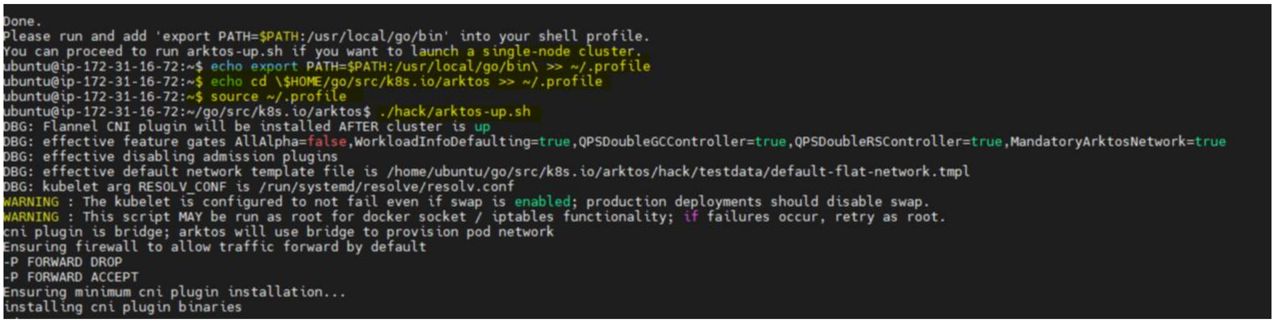
echo export PATH=$PATH:/usr/local/go/bin\ >> ~/.profile

echo cd \$HOME/go/src/k8s.io/arktos >> ~/.profile

source ~/.profile

Step 3 :- Start Arktos cluster

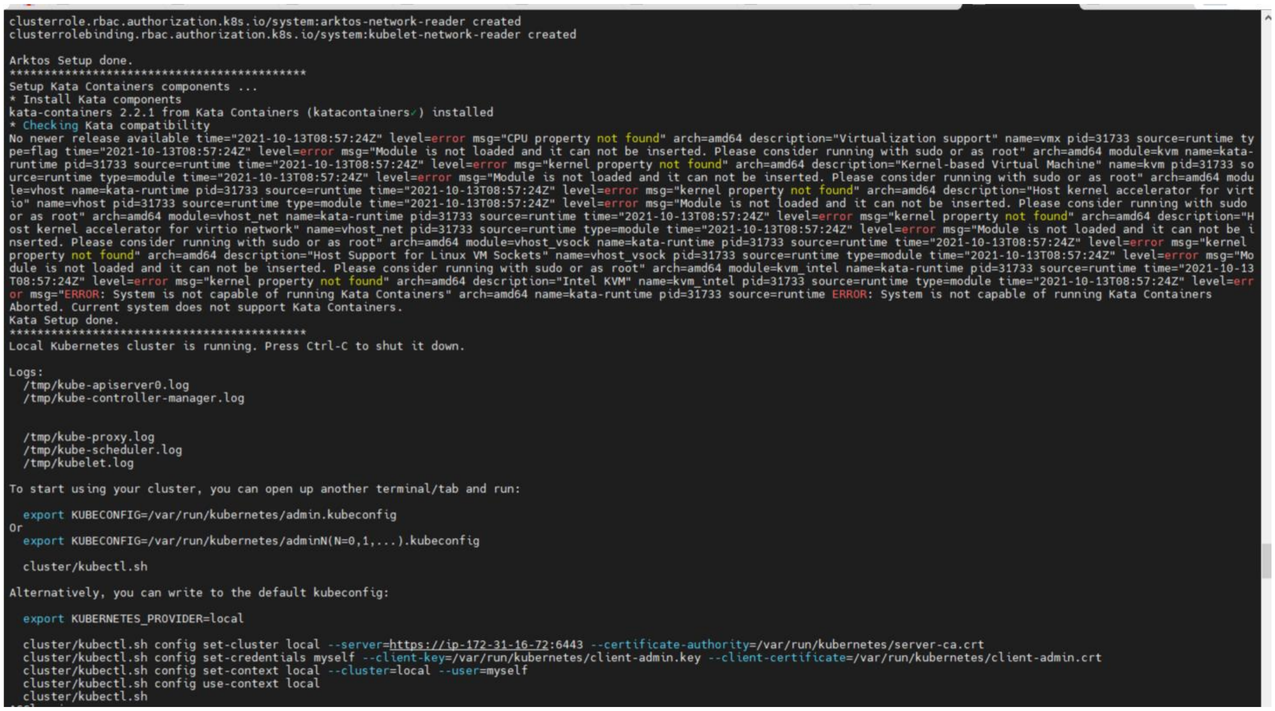
```
CNIPLUGIN=mizar ./hack/arktos-up.sh
```



```
Done.
Please run and add 'export PATH=$PATH:/usr/local/go/bin' into your shell profile.
You can proceed to run arktos-up.sh if you want to launch a single-node cluster.
ubuntu@ip-172-31-16-72:~$ echo export PATH=$PATH:/usr/local/go/bin\ >> ~/.profile
ubuntu@ip-172-31-16-72:~$ echo cd \$HOME/go/src/k8s.io/arktos >> ~/.profile
ubuntu@ip-172-31-16-72:~$ source ~/.profile
ubuntu@ip-172-31-16-72:~/go/src/k8s.io/arktos$ ./hack/arktos-up.sh
DBG: Flannel CNI plugin will be installed AFTER cluster is up
DBG: effective feature gates AllAlpha=false,WorkloadInfoDefaulting=true,QPSDoubleGCController=true,QPSDoubleRSController=true,MandatoryArktosNetwork=true
DBG: effective disabling admission plugins
DBG: effective default network template file is /home/ubuntu/go/src/k8s.io/arktos/hack/testdata/default-flat-network.tmpl
DBG: kubelet arg RESOLV_CONF is /run/systemd/resolve/resolv.conf
WARNING : The kubelet is configured to not fail even if swap is enabled; production deployments should disable swap.
WARNING : This script MAY be run as root for docker socket / iptables functionality; if failures occur, retry as root.
cni plugin is bridge; arktos will use bridge to provision pod network
Ensuring firewall to allow traffic forward by default
-P FORWARD DROP
-P FORWARD ACCEPT
Ensuring minimum cni plugin installation...
installing cni plugin binaries
```

Deployment successfully done.

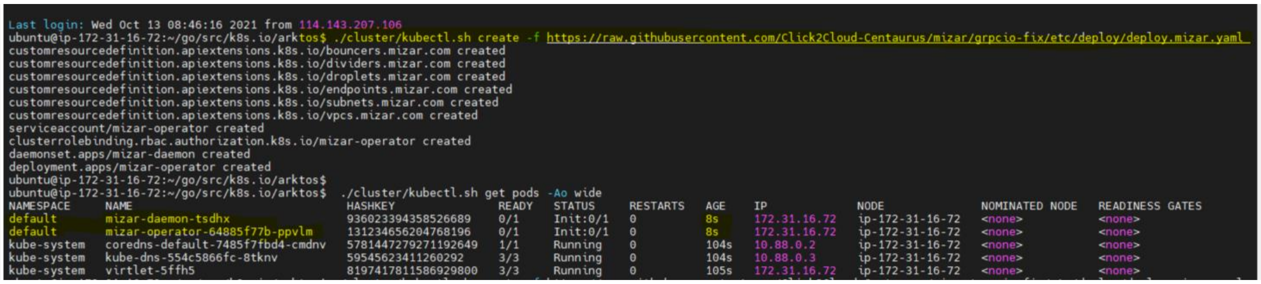
output:



```
clusterrole.rbac.authorization.k8s.io/system:arktos-network-reader created
clusterrolebinding.rbac.authorization.k8s.io/system:kubelet-network-reader created

Arktos Setup done.
*******************************************
Setup Kata Containers components ...
* Install Kata components
kata-containers 2.2.1 from Kata Containers (katacontainers✓) installed
* Checking Kata compatibility
No newer release available time="2021-10-13T08:57:24Z" level=error msg="CPU property not found" arch=amd64 description="Virtualization support" name=vmx pid=31733 source=runtime ty
pe=flag time="2021-10-13T08:57:24Z" level=error msg="Module is not loaded and it can not be inserted. Please consider running with sudo or as root" arch=amd64 module=kvm name=kata-
runtime pid=31733 source=runtime time="2021-10-13T08:57:24Z" level=error msg="kernel property not found" arch=amd64 description="Kernel-based Virtual Machine" name=kvm pid=31733 so
urce=runtime type=module time="2021-10-13T08:57:24Z" level=error msg="Module is not loaded and it can not be inserted. Please consider running with sudo or as root" arch=amd64 modu
le=vhost name=kata-runtime pid=31733 source=runtime time="2021-10-13T08:57:24Z" level=error msg="kernel property not found" arch=amd64 description="Host kernel accelerator for virt
io" name=vhost pid=31733 source=runtime type=module time="2021-10-13T08:57:24Z" level=error msg="Module is not loaded and it can not be inserted. Please consider running with sudo
or as root" arch=amd64 module=vhost_net name=kata-runtime pid=31733 source=runtime time="2021-10-13T08:57:24Z" level=error msg="kernel property not found" arch=amd64 description="H
ost kernel accelerator for virtio network" name=vhost_net pid=31733 source=runtime type=module time="2021-10-13T08:57:24Z" level=error msg="Module is not loaded and it can not be i
nserted. Please consider running with sudo or as root" arch=amd64 module=vhost_vsock name=kata-runtime pid=31733 source=runtime time="2021-10-13T08:57:24Z" level=error msg="kernel
property not found" arch=amd64 description="Host Support for Linux VM Sockets" name=vhost_vsock pid=31733 source=runtime type=module time="2021-10-13T08:57:24Z" level=error msg="Mo
dule is not loaded and it can not be inserted. Please consider running with sudo or as root" arch=amd64 module=kvm_intel name=kata-runtime pid=31733 source=runtime time="2021-10-13
T08:57:24Z" level=error msg="kernel property not found" arch=amd64 description="Intel KVM" name=kvm_intel pid=31733 source=runtime type=module time="2021-10-13T08:57:24Z" level=err
or msg="ERROR: System is not capable of running Kata Containers" arch=amd64 name=kata-runtime pid=31733 source=runtime ERROR: System is not capable of running Kata Containers
Aborted. Current system does not support Kata Containers.
Kata Setup done.
*******************************************
Local Kubernetes cluster is running. Press Ctrl-C to shut it down.

Logs:
  /tmp/kube-apiserver0.log
  /tmp/kube-controller-manager.log


  /tmp/kube-proxy.log
  /tmp/kube-scheduler.log
  /tmp/kubelet.log

To start using your cluster, you can open up another terminal/tab and run:

  export KUBECONFIG=/var/run/kubernetes/admin.kubeconfig
Or
  export KUBECONFIG=/var/run/kubernetes/adminN(N=0,1,...).kubeconfig

  cluster/kubectl.sh

Alternatively, you can write to the default kubeconfig:

  export KUBERNETES_PROVIDER=local

  cluster/kubectl.sh config set-cluster local --server=https://ip-172-31-16-72:6443 --certificate-authority=/var/run/kubernetes/server-ca.crt
  cluster/kubectl.sh config set-credentials myself --client-key=/var/run/kubernetes/client-admin.key --client-certificate=/var/run/kubernetes/client-admin.crt
  cluster/kubectl.sh config set-context local --cluster=local --user=myself
  cluster/kubectl.sh config use-context local
  cluster/kubectl.sh
```

4. Leave the "arktos-up.sh" terminal and open another terminal to the master node. Verify mizar pods i.e. mizar-operator and mizar-daemon pods are in running state,

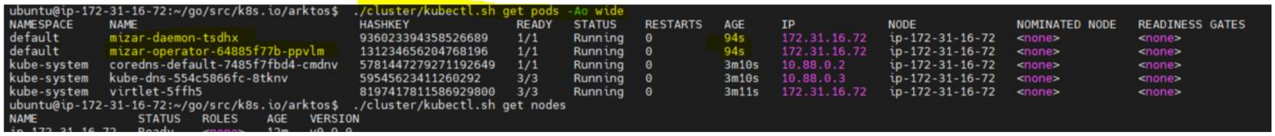
./cluster/kubectl.sh create -f https://raw.githubusercontent.com/CentaurusInfra/mizar/dev-next/etc/deploy/deploy.mizar.yaml



```
Last login: Wed Oct 13 08:46:16 2021 from 114.143.207.106
ubuntu@ip-172-31-16-72:~/go/src/k8s.io/arktos$ ./cluster/kubectl.sh create -f https://raw.githubusercontent.com/Click2Cloud-Centaurus/mizar/grpcio-fix/etc/deploy/deploy.mizar.yaml
customresourcedefinition.apiextensions.k8s.io/bouncers.mizar.com created
customresourcedefinition.apiextensions.k8s.io/dividers.mizar.com created
customresourcedefinition.apiextensions.k8s.io/droplets.mizar.com created
customresourcedefinition.apiextensions.k8s.io/endpoints.mizar.com created
customresourcedefinition.apiextensions.k8s.io/subnets.mizar.com created
customresourcedefinition.apiextensions.k8s.io/vpcs.mizar.com created
serviceaccount/mizar-operator created
clusterrolebinding.rbac.authorization.k8s.io/mizar-operator created
daemonset.apps/mizar-daemon created
deployment.apps/mizar-operator created
ubuntu@ip-172-31-16-72:~/go/src/k8s.io/arktos$
ubuntu@ip-172-31-16-72:~/go/src/k8s.io/arktos$  ./cluster/kubectl.sh get pods -Ao wide
NAMESPACE     NAME                                HASHKEY               READY   STATUS      RESTARTS   AGE    IP             NODE               NOMINATED NODE   READINESS GATES
default       mizar-daemon-tsdhx                  936023394358526689    0/1     Init:0/1    0          8s     172.31.16.72   ip-172-31-16-72    <none>           <none>
default       mizar-operator-64885f77b-ppvlm      131234656204768196    0/1     Init:0/1    0          8s     172.31.16.72   ip-172-31-16-72    <none>           <none>
kube-system   coredns-default-7485f7fbd4-cmdnv    5781447279271192649   1/1     Running     0          104s   10.88.0.2      ip-172-31-16-72    <none>           <none>
kube-system   kube-dns-554c5866fc-8tknv           59545623411260292     3/3     Running     0          104s   10.88.0.3      ip-172-31-16-72    <none>           <none>
kube-system   virtlet-5ffh5                       8197417811586929800   3/3     Running     0          105s   172.31.16.72   ip-172-31-16-72    <none>           <none>
```

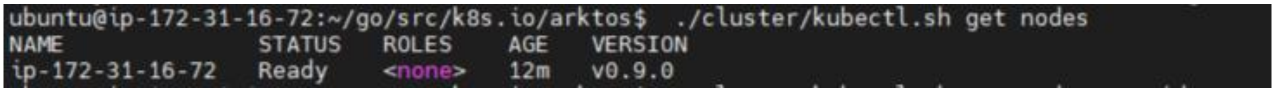
Step-4:- Check Cluster health

./cluster/kubectl.sh get pods -Ao wide



```
ubuntu@ip-172-31-16-72:~/go/src/k8s.io/arktos$  ./cluster/kubectl.sh get pods -Ao wide
NAMESPACE     NAME                                HASHKEY               READY   STATUS    RESTARTS   AGE     IP             NODE               NOMINATED NODE   READINESS GATES
default       mizar-daemon-tsdhx                  936023394358526689    1/1     Running   0          94s     172.31.16.72   ip-172-31-16-72    <none>           <none>
default       mizar-operator-64885f77b-ppvlm      131234656204768196    1/1     Running   0          94s     172.31.16.72   ip-172-31-16-72    <none>           <none>
kube-system   coredns-default-7485f7fbd4-cmdnv    5781447279271192649   1/1     Running   0          3m10s   10.88.0.2      ip-172-31-16-72    <none>           <none>
kube-system   kube-dns-554c5866fc-8tknv           59545623411260292     3/3     Running   0          3m10s   10.88.0.3      ip-172-31-16-72    <none>           <none>
kube-system   virtlet-5ffh5                       8197417811586929800   3/3     Running   0          3m11s   172.31.16.72   ip-172-31-16-72    <none>           <none>
ubuntu@ip-172-31-16-72:~/go/src/k8s.io/arktos$  ./cluster/kubectl.sh get nodes
NAME              STATUS   ROLES    AGE   VERSION
```

./cluster/kubectl.sh get nodes



```
ubuntu@ip-172-31-16-72:~/go/src/k8s.io/arktos$  ./cluster/kubectl.sh get nodes
NAME              STATUS   ROLES    AGE   VERSION
ip-172-31-16-72   Ready    <none>   12m   v0.9.0
```

./cluster/kubectl.sh get vpc -Ao wide

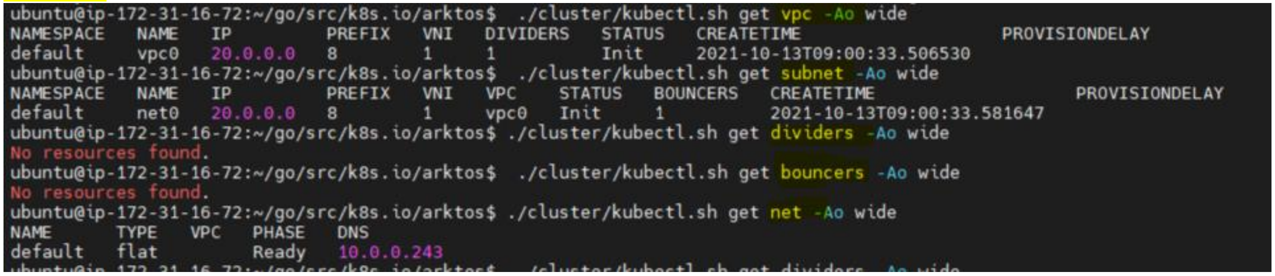
./cluster/kubectl.sh get subnet -Ao wide

./cluster/kubectl.sh get dividers -Ao wide

./cluster/kubectl.sh get bouncers -Ao wide

./cluster/kubectl.sh get net -Ao wide

output:



```
ubuntu@ip-172-31-16-72:~/go/src/k8s.io/arktos$  ./cluster/kubectl.sh get vpc -Ao wide
NAMESPACE   NAME   IP         PREFIX   VNI   DIVIDERS   STATUS   CREATETIME                   PROVISIONDELAY
default     vpc0   20.0.0.0   8        1     1          Init     2021-10-13T09:00:33.506530
ubuntu@ip-172-31-16-72:~/go/src/k8s.io/arktos$  ./cluster/kubectl.sh get subnet -Ao wide
NAMESPACE   NAME   IP         PREFIX   VNI   VPC    STATUS   BOUNCERS   CREATETIME                   PROVISIONDELAY
default     net0   20.0.0.0   8        1     vpc0   Init     1          2021-10-13T09:00:33.581647
ubuntu@ip-172-31-16-72:~/go/src/k8s.io/arktos$ ./cluster/kubectl.sh get dividers -Ao wide
No resources found.
ubuntu@ip-172-31-16-72:~/go/src/k8s.io/arktos$  ./cluster/kubectl.sh get bouncers -Ao wide
No resources found.
ubuntu@ip-172-31-16-72:~/go/src/k8s.io/arktos$ ./cluster/kubectl.sh get net -Ao wide
NAME      TYPE   VPC   PHASE   DNS
default   flat         Ready   10.0.0.243
```