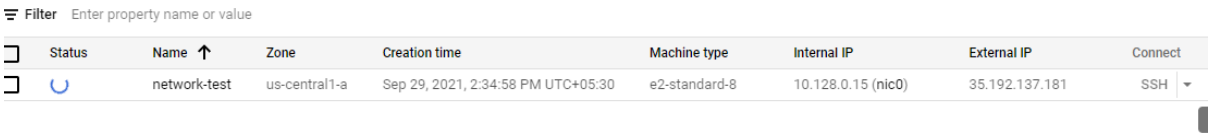
Arktos deployment with Mizar CNI on GCE

Instance size: - e2-standard-8
OS: Ubuntu 18.04 LTS 64-bit



Status	Name	Zone	Creation time	Machine type	Internal IP	External IP	Connect
	network-test	us-central1-a	Sep 29, 2021, 2:34:58 PM UTC+05:30	e2-standard-8	10.128.0.15 (nic0)	35.192.137.181	SSH

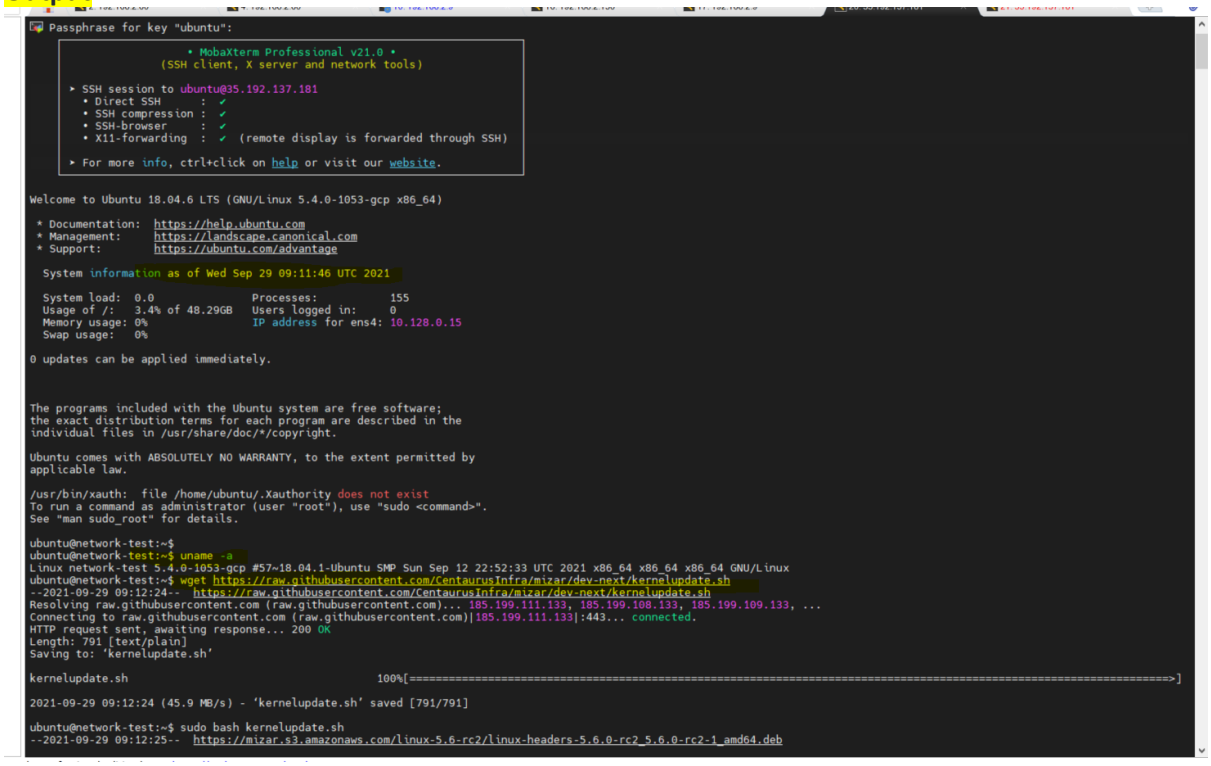
Step-1 Check the kernel version & update the kernel

uname -a

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

sudo bash kernelupdate.sh

Output



```
linux-libc-dev_5.6.0-rc2-1_amd64.deb                100%[=============================================================

2021-09-29 09:12:40 (4.65 MB/s) - '../linux-5.6-rc2/linux-libc-dev_5.6.0-rc2-1_amd64.deb' saved [1082248/1082248]

Continue kernel update (y/n)?y
Updating kernel
Selecting previously unselected package linux-headers-5.6.0-rc2.
(Reading database ... 65612 files and directories currently installed.)
Preparing to unpack .../linux-headers-5.6.0-rc2_5.6.0-rc2-1_amd64.deb ...
Unpacking linux-headers-5.6.0-rc2 (5.6.0-rc2-1) ...
Selecting previously unselected package linux-image-5.6.0-rc2-dbg.
Preparing to unpack .../linux-image-5.6.0-rc2-dbg_5.6.0-rc2-1_amd64.deb ...
Unpacking linux-image-5.6.0-rc2-dbg (5.6.0-rc2-1) ...
Selecting previously unselected package linux-image-5.6.0-rc2.
Preparing to unpack .../linux-image-5.6.0-rc2_5.6.0-rc2-1_amd64.deb ...
Unpacking linux-image-5.6.0-rc2 (5.6.0-rc2-1) ...
Selecting previously unselected package linux-libc-dev:amd64.
Preparing to unpack .../linux-libc-dev_5.6.0-rc2-1_amd64.deb ...
Unpacking linux-libc-dev:amd64 (5.6.0-rc2-1) ...
Setting up linux-headers-5.6.0-rc2 (5.6.0-rc2-1) ...
Setting up linux-image-5.6.0-rc2-dbg (5.6.0-rc2-1) ...
Setting up linux-image-5.6.0-rc2 (5.6.0-rc2-1) ...
update-initramfs: Generating /boot/initrd.img-5.6.0-rc2
Sourcing file `/etc/default/grub'
Sourcing file `/etc/default/grub.d/50-cloudimg-settings.cfg'
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-5.6.0-rc2
Found initrd image: /boot/initrd.img-5.6.0-rc2
Found linux image: /boot/vmlinuz-5.4.0-1053-gcp
Found initrd image: /boot/initrd.img-5.4.0-1053-gcp
Adding boot menu entry for EFI firmware configuration
done
Setting up linux-libc-dev:amd64 (5.6.0-rc2-1) ...
Reboot host (y/n)?y
Rebooting
```

Kernel updated successfully.

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

```
git clone https://github.com/Click2Cloud-Centaurus/arktos.git
~/go/src/k8s.io/arktos -b default-cni-mizar

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

Output

```
Last login: Wed Sep 29 09:11:47 2021 from 114.143.207.106
ubuntu@network-test:~$ git clone https://github.com/Click2Cloud-Centaurus/arktos.git ~/go/src/k8s.io/arktos -b default-cni-mizar
Cloning into '/home/ubuntu/go/src/k8s.io/arktos'...
remote: Enumerating objects: 104467, done.
remote: Counting objects: 100% (1130/1130), done.
remote: Compressing objects: 100% (683/683), done.
remote: Total 104467 (delta 564), reused 635 (delta 431), pack-reused 103337
Receiving objects: 100% (104467/104467), 333.11 MiB | 32.58 MiB/s, done.
Resolving deltas: 100% (63151/63151), done.
Checking out files: 100% (20762/20762), done.
ubuntu@network-test:~$ 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.
Update apt.
Hit:1 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:3 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:4 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [8570 kB]
Get:5 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:6 http://us-central1.gce.archive.ubuntu.com/ubuntu bionic/universe Translation-en [4941 kB]
```

```
echo export PATH=$PATH:/usr/local/go/bin\ >> ~/.profile
echo cd \$HOME/go/src/k8s.io/arktos >> ~/.profile
source ~/.profile
```

```
Done.
Please run and add 'export PATH=$PATH:/usr/local/go/bin' into your shell profile.
You can proceed to run arktos-up.sh if you want to launch a single-node cluster.
ubuntu@network-test:~$ echo export PATH=$PATH:/usr/local/go/bin\ >> ~/.profile
ubuntu@network-test:~$ echo cd \$HOME/go/src/k8s.io/arktos >> ~/.profile
ubuntu@network-test:~$ source ~/.profile
ubuntu@network-test:~/go/src/k8s.io/arktos$ CNIPLUGIN=mizar ./hack/arktos-up.sh
```

Step-3 Start Arktos cluster

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

Output

Failed - error cannot import package …

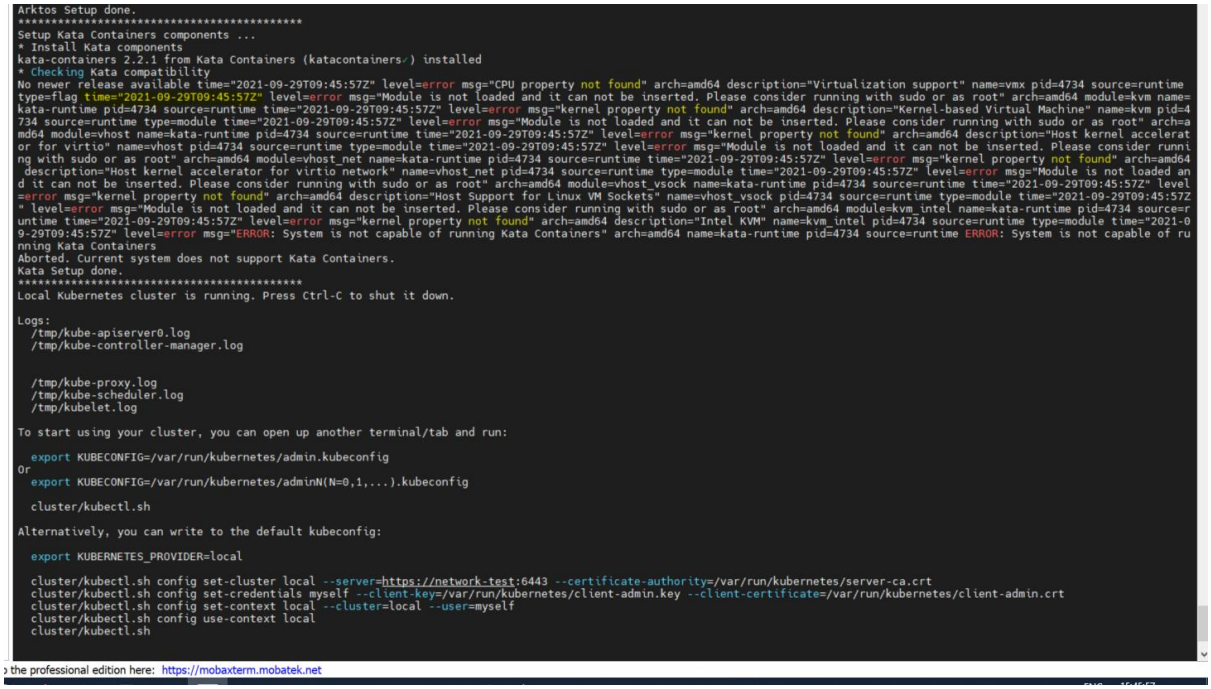
```
Stopping Apparmor service
make: Entering directory '/home/ubuntu/go/src/k8s.io/arktos'
make[1]: Entering directory '/home/ubuntu/go/src/k8s.io/arktos'
+++ [0929 09:27:45] Building go targets for linux/amd64:
    ./vendor/k8s.io/code-generator/cmd/deepcopy-gen
+++ [0929 09:27:45] Building go targets for linux/amd64:
    ./vendor/k8s.io/code-generator/cmd/defaulter-gen
+++ [0929 09:27:45] Building go targets for linux/amd64:
    ./vendor/k8s.io/code-generator/cmd/conversion-gen
+++ [0929 09:27:45] Building go targets for linux/amd64:
    ./vendor/k8s.io/kube-openapi/cmd/openapi-gen
+++ [0929 09:27:49] Building go targets for linux/amd64:
    ./vendor/github.com/go-bindata/go-bindata
F0929 09:27:58.245152   17028 defaulter.go:292] cannot import package ../../../../vendor/k8s.io/api/authorization/v1beta1
Makefile.generated_files:249: recipe for target 'gen_defaulter' failed
make[1]: *** [gen_defaulter] Error 1
make[1]: *** Waiting for unfinished jobs....
make[1]: Leaving directory '/home/ubuntu/go/src/k8s.io/arktos'
Makefile:560: recipe for target 'generated_files' failed
make: *** [generated_files] Error 2
make: Leaving directory '/home/ubuntu/go/src/k8s.io/arktos'
!!! Error in ./hack/arktos-up.sh:138
  Error in ./hack/arktos-up.sh:138. 'make -j4 -C "${KUBE_ROOT}" WHAT="cmd/kubectl cmd/hyperkube cmd/kube-apiserver cmd/kube-controller-manager cmd/workload-controller-manager cm
belet cmd/kube-proxy cmd/kube-scheduler cmd/arktos-network-controller"' exited with status 2
Call stack:
  1: ./hack/arktos-up.sh:138 main(...)
Exiting with status 1
ubuntu@network-test:~/go/src/k8s.io/arktos$ CNIPLUGIN=mizar ./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.
```

Re-run same script without any changes.

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

Deployment Successfully done.

Output



Step-4 Check Cluster health

Open new terminal for same instance and run following commands:

Check node, Pods, Vpc, Subnet, dividers & , bouncers， Net status

```
./cluster/kubectl.sh get nodes
```

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

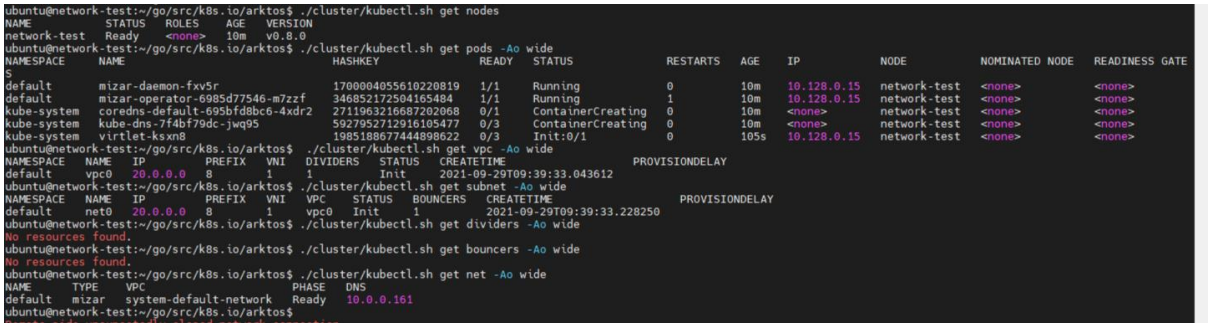
```
 ./cluster/kubectl.sh get vpc -Ao wide
```

```
./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
```



```
ubuntu@network-test:~/go/src/k8s.io/arktos$ ./cluster/kubectl.sh get nodes
NAME           STATUS   ROLES    AGE   VERSION
network-test   Ready    <none>   10m   v0.8.0
ubuntu@network-test:~/go/src/k8s.io/arktos$ ./cluster/kubectl.sh get pods -Ao wide
NAMESPACE     NAME                                HASHKEY                READY   STATUS              RESTARTS   AGE    IP            NODE           NOMINATED NODE   READINESS GATE
S
default       mizar-daemon-fxv5r                  1700004055610220819    1/1     Running             0          10m    10.128.0.15   network-test   <none>           <none>
default       mizar-operator-6985d77546-m7zzf     346852172504165484     1/1     Running             1          10m    10.128.0.15   network-test   <none>           <none>
kube-system   coredns-default-695bfd8bc6-4xdr2    2711963216687202068    0/1     ContainerCreating   0          10m    <none>        network-test   <none>           <none>
kube-system   kube-dns-7f4bf79dc-jwq95            5927952712916105477    0/3     ContainerCreating   0          10m    <none>        network-test   <none>           <none>
kube-system   virtlet-ksxn8                       1985188677444898622    0/3     Init:0/1            0          105s   10.128.0.15   network-test   <none>           <none>
ubuntu@network-test:~/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     2021-09-29T09:39:33.043612
ubuntu@network-test:~/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          2021-09-29T09:39:33.228250
ubuntu@network-test:~/go/src/k8s.io/arktos$ ./cluster/kubectl.sh get dividers -Ao wide
No resources found.
ubuntu@network-test:~/go/src/k8s.io/arktos$ ./cluster/kubectl.sh get bouncers -Ao wide
No resources found.
ubuntu@network-test:~/go/src/k8s.io/arktos$ ./cluster/kubectl.sh get net -Ao wide
NAME      TYPE     VPC                  PHASE   DNS
default   mizar   system-default-network   Ready   10.0.0.161
ubuntu@network-test:~/go/src/k8s.io/arktos$
```