


# Test report - Deployment of Arktos Cluster with Mizar CNI on GCE

This document captures the steps to deploy an Arktos cluster lab with mizar cni. The machine in this lab used are GCE e2-standard-8 (8 vCPUs, 32 GB memory) and the storage size is 128GB), Ubuntu 18.04 LTS.

Date-27.09.2021

## Create an instance on GCE

Created instance on GCE

 **Filter** Enter property name or value

| <input type="checkbox"/> | Status | Name ↑     | Zone          | Creation time                       | Machine type  | Internal IP          | External IP   | Connect |
|--------------------------|--------|------------|---------------|-------------------------------------|---------------|----------------------|---------------|---------|
| <input type="checkbox"/> | ✓      | instance-1 | us-central1-a | Sep 27, 2021, 11:04:34 AM UTC+05:30 | e2-standard-8 | 10.128.15.197 (nic0) | 34.70.131.140 | SSH ▾ ⋮ |

SSH instance with credentials.

## Step-1: Update kernel (If required)

To check kernel, run following command

```
uname -a
```

output:

```
ubuntu@instance-1:~$ uname -a
Linux instance-1 5.4.0-1051-gcp #55~18.04.1-Ubuntu SMP Sun Aug 1 20:38:04 UTC 2021 x86_64 x86_64 x86_64 GNU/Linux
```

Here kernel version is 5.4.0-1051-gcp which is less than the required kernel version, so to update the kernel version to 5.6.0-rc2, we used the following steps :

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

output:

```

linux-image-5.6.0-rc2-dbg_5.6.0-rc2-1_amd64.deb 100%[=====] 818.09M 66.5MB/s in 14s
2021-09-27 05:39:50 (60.2 MB/s) - './linux-5.6-rc2/linux-image-5.6.0-rc2-dbg_5.6.0-rc2-1_amd64.deb' saved [857827912/857827912]

--2021-09-27 05:39:50-- https://mizar.s3.amazonaws.com/linux-5.6-rc2/linux-image-5.6.0-rc2-1_amd64.deb
Resolving mizar.s3.amazonaws.com (mizar.s3.amazonaws.com)... 52.216.80.64
Connecting to mizar.s3.amazonaws.com (mizar.s3.amazonaws.com)[52.216.80.64]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 56427036 (54M) [application/x-www-form-urlencoded]
Saving to: './linux-5.6-rc2/linux-image-5.6.0-rc2-1_amd64.deb'

linux-image-5.6.0-rc2-1_amd64.deb 100%[=====] 53.81M 63.4MB/s in 0.8s
2021-09-27 05:39:51 (63.4 MB/s) - './linux-5.6-rc2/linux-image-5.6.0-rc2-1_amd64.deb' saved [56427036/56427036]

--2021-09-27 05:39:51-- https://mizar.s3.amazonaws.com/linux-5.6-rc2/linux-libc-dev_5.6.0-rc2-1_amd64.deb
Resolving mizar.s3.amazonaws.com (mizar.s3.amazonaws.com)... 52.216.80.64
Connecting to mizar.s3.amazonaws.com (mizar.s3.amazonaws.com)[52.216.80.64]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1082248 (1.0M) [text/x-gzip]
Saving to: './linux-5.6-rc2/linux-libc-dev_5.6.0-rc2-1_amd64.deb'

linux-libc-dev_5.6.0-rc2-1_amd64.deb 100%[=====] 1.03M 6.40MB/s in 0.2s
2021-09-27 05:39:52 (6.40 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 ... 65616 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-5.4.0-1051-gcp
Found initrd image: /boot/initrd.img-5.4.0-1051-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)?n

```

## Step-2: Install dependencies

Relogin and Run the following steps to install dependencies required for arktos deployment:

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

output:

```

ubuntu@instance-1:~$ 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: 104406, done.
remote: Counting objects: 100% (1069/1069), done.
remote: Compressing objects: 100% (633/633), done.
remote: Total 104406 (delta 529), reused 617 (delta 421), pack-reused 103337
Receiving objects: 100% (104406/104406), 333.08 MiB | 30.71 MiB/s, done.
Resolving deltas: 100% (63116/63116), done.
Checking out files: 100% (20762/20762), done.

```

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

output:

```

Setting up libgcc-7-dev:amd64 (7.5.0-3ubuntu1-18.04) ...
Setting up cpp-7 (7.5.0-3ubuntu1-18.04) ...
Setting up binutils-x86-64-linux-gnu (2.30-21ubuntu1-18.04.5) ...
Setting up cpp (4:7.4.0-1ubuntu2.3) ...
Setting up binutils (2.30-21ubuntu1-18.04.5) ...
Setting up gcc-7 (7.5.0-3ubuntu1-18.04) ...
Setting up gcc (4:7.4.0-1ubuntu2.3) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for libc-bin (2.27-3ubuntu1.4) ...
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  libnuma1
Use 'sudo apt autoremove' to remove it.
The following additional packages will be installed:
  libjq1 libjq1:amd64
The following NEW packages will be installed:
  jq libjq1 libjq1:amd64
0 upgraded, 2 newly installed, 0 to remove and 37 not upgraded.
Need to get 276 kB of archives.
After this operation, 930 kB of additional disk space will be used.
Get:1 http://us-central1-ocw.archive.ubuntu.com/ubuntu bionic/universe amd64 libjq1:amd64 6.7.0-1 [119 kB]
Get:2 http://us-central1-ocw.archive.ubuntu.com/ubuntu bionic/universe amd64 libjq1:amd64 1.5+dfsg-2 [111 kB]
Get:3 http://us-central1-ocw.archive.ubuntu.com/ubuntu bionic/universe amd64 jq amd64 1.5+dfsg-2 [45.6 kB]
Fetched 276 kB in 0s (4280 kB/s)
Selecting previously unselected package libjq1:amd64.
(Reading database ... 100801 files and directories currently installed.)
Preparing to unpack .../libjq1:amd64 6.7.0-1_1_amd64.deb ...
Unpacking libjq1:amd64 (6.7.0-1) ...
Selecting previously unselected package libjq1:amd64.
Preparing to unpack .../libjq1:amd64 1.5+dfsg-2_amd64.deb ...
Unpacking libjq1:amd64 (1.5+dfsg-2) ...
Selecting previously unselected package jq.
Preparing to unpack .../jq_1.5+dfsg-2_amd64.deb ...
Unpacking jq (1.5+dfsg-2) ...
Setting up libjq1:amd64 (6.7.0-1) ...
Setting up libjq1:amd64 (1.5+dfsg-2) ...
Setting up jq (1.5+dfsg-2) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for libc-bin (2.27-3ubuntu1.4) ...
Install golang.
--2021-09-27 05:50:21-- https://dl.google.com/go/go1.13.9.linux-amd64.tar.gz
Resolving dl.google.com (dl.google.com)... 142.250.1.190, 142.250.1.136, 142.250.1.91, ...
Connecting to dl.google.com (dl.google.com)|142.250.1.190|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 120139686 (115M) [application/octet-stream]
Saving to: '/tmp/go1.13.9.linux-amd64.tar.gz'

go1.13.9.linux-amd64.tar.gz      100%[=====] 114.57M  34.8MB/s   in 3.3s

2021-09-27 05:50:25 (34.8 MB/s) - '/tmp/go1.13.9.linux-amd64.tar.gz' saved [120139686/120139686]

Done.
Please run and add 'export PATH=$PATH:/usr/local/go/bin' into your shell profile.
You can proceed to run arkto-up.sh if you want to launch a single-node cluster.
ubuntu@instance-1:~$

```

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

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

```
source ~/.profile
```

output:

```

ubuntu@instance-1:~$ echo export PATH=$PATH:/usr/local/go/bin\ >> ~/.profile
ubuntu@instance-1:~$ echo cd \${HOME}/go/src/k8s.io/arktos >> ~/.profile
ubuntu@instance-1:~$ source ~/.profile
ubuntu@instance-1:~/go/src/k8s.io/arktos$

```

## Step-3: Start Arktos cluster

Run following step to deploy arktos cluster with Mizar as CNI:

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

Finally we got following output, which indicates that arktos cluster created successfully with Mizar as CNI

output:

```

clusterrolebinding.rbac.authorization.k8s.io/virtlet-crd created
serviceaccount/virtlet created
NAME      DESIRED  CURRENT  READY  UP-TO-DATE  AVAILABLE  NODE SELECTOR  AGE
virtlet   1        1        0      1           0          <none>         0s
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="2021-09-27T06:05:59Z" level=error msg="CPU property not found" arch=amd64 description="Virtualization support" name=vmx pid=30026 source=runtime type=flag time="2021-09-27T06:05:59Z" 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=30026 source=runtime time="2021-09-27T06:05:59Z" level=error msg="kernel property not found" arch=amd64 description="Host kernel accelerator for virtio" name=vhost pid=30026 source=runtime type=module time="2021-09-27T06:05:59Z" 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=30026 source=runtime type=module time="2021-09-27T06:05:59Z" level=error msg="Host kernel accelerator for virtio network" name=vhost_net pid=30026 source=runtime type=module time="2021-09-27T06:05:59Z" level=error msg="Module is not loaded and it can not be inserted. Please consider running with sudo or as root" arch=amd64 module=vsock name=kata-runtime pid=30026 source=runtime type=module time="2021-09-27T06:05:59Z" level=error msg="kernel property not found" arch=amd64 description="Host Support for Linux VM Sockets" name=vhost_vsock pid=30026 source=runtime type=module time="2021-09-27T06:05:59Z" 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_intel name=kata-runtime pid=30026 source=runtime type=module time="2021-09-27T06:05:59Z" level=error msg="kernel property not found" arch=amd64 description="Intel KVM" name=kvm_intel pid=30026 source=runtime type=module time="2021-09-27T06:05:59Z" level=error msg="ERROR: System is not capable of running Kata Containers" arch=amd64 name=kata-runtime pid=30026 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/admin(N=0,1,...).kubeconfig
cluster/kubect1.sh

Alternatively, you can write to the default kubeconfig:
export KUBERNETES_PROVIDER=local
cluster/kubect1.sh config set-cluster local --server=https://instance-1:6443 --certificate-authority=/var/run/kubernetes/server-ca.crt
cluster/kubect1.sh config set-credentials myself --client-key=/var/run/kubernetes/client-admin.key --client-certificate=/var/run/kubernetes/client-admin.crt
cluster/kubect1.sh config set-context local --cluster=local --user=myself
cluster/kubect1.sh config use-context local
cluster/kubect1.sh

```

Leave this terminal here as it is (do not close the terminal) and open new terminal of same instance

## Step-4 Check Cluster health

Open new terminal for same instance and run following commands:

### 1) Check node status

```
./cluster/kubect1.sh get nodes -Ao wide
```

#### Output

```

ubuntu@instance-1:~/go/src/k8s.io/arktos$ ./cluster/kubect1.sh get nodes -Ao wide
NAME      STATUS  ROLES  AGE   VERSION  INTERNAL-IP  EXTERNAL-IP  OS-IMAGE             KERNEL-VERSION  CONTAINER-RUNTIME
instance-1 Ready   <none>  6m32s v0.8.0   10.128.15.197 <none>       Ubuntu 18.04.5 LTS   5.6.0-rc2        containerd://1.4.0-beta.1-29-g70b0d3cf
ubuntu@instance-1:~/go/src/k8s.io/arktos$

```

### 2) Check pods status

```
./cluster/kubect1.sh get pods -Ao wide
```

#### Output

```

ubuntu@instance-1:~/go/src/k8s.io/arktos$ ./cluster/kubect1.sh get pods -Ao wide
NAME                NAMESPACE   NAME              ROLES   AGE   STATUS   RESTARTS   AGE   IP              NODE                NOMINATED NODE   READINESS GATES
default            mizar-daemon-0bsxq   7735477782612895179  1/1    Running  0          8m3s   10.128.15.197   instance-1          <none>            <none>
default            mizar-operator-6985d77546-82nqg  1650833051907194888  1/1    Running  0          8m3s   10.128.15.197   instance-1          <none>            <none>
default            pod-767686bb54-5bg6s  5774664824995397753  0/1    ContainerCreating  0          6m45s   <none>          instance-1          <none>            <none>
kube-system        coredns-default-54895bsbfc-qhpkm  6167013444726112653  0/1    ContainerCreating  0          8m3s   <none>          instance-1          <none>            <none>
kube-system        kube-dns-2f4bf79dc-zcjdj  436089655765864667  0/3    ContainerCreating  0          8m3s   <none>          instance-1          <none>            <none>
kube-system        virtlet-06gb4        3236308487394719128  3/3    Running   0          8m2s   10.128.15.197   instance-1          <none>            <none>
ubuntu@instance-1:~/go/src/k8s.io/arktos$

```

### 3) Check vpc status

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

#### Output

```
ubuntu@instance-1:~/go/src/k8s.io/arktos$ ./cluster/kubectl.sh get vpcs -Ao wide
NAMESPACE  NAME  IP      PREFIX  VNI  DIVIDERS  STATUS  CREATETIME  PROVISIONDELAY
default    vpc0  20.0.0.0  8       1    1         Provisioned  2021-09-27T06:47:02.104597  42.064971
```

### 4) Check subnets

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

#### Output

```
ubuntu@instance-1:~/go/src/k8s.io/arktos$ ./cluster/kubectl.sh get subnets -Ao wide
NAMESPACE  NAME  IP      PREFIX  VNI  VPC  STATUS  BOUNCERS  CREATETIME  PROVISIONDELAY
default    net0  20.0.0.0  8       1    vpc0  Provisioned  1         2021-09-27T06:47:02.208350  62.121736
ubuntu@instance-1:~/go/src/k8s.io/arktos$
```

### 5) Check net

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

#### Output

```
ubuntu@instance-1:~/go/src/k8s.io/arktos$ ./cluster/kubectl.sh get net -Ao wide
NAME      TYPE    VPC              PHASE  DNS
default   mizar   system-default-network  Ready  10.0.0.164
ubuntu@instance-1:~/go/src/k8s.io/arktos$
```

### 6) Check dividers

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

#### Output

```
ubuntu@instance-1:~/go/src/k8s.io/arktos$ ./cluster/kubectl.sh get dividers -Ao wide
NAMESPACE  NAME  VPC  IP  MAC  DROPLET  STATUS  CREATETIME  PROVISIONDELAY
default    vpc0-d-adb42b77-2823-402d-bbbf-904ff3ad551f  vpc0  instance-1  Provisioned  2021-09-27T06:47:44.154017  0.272695
ubuntu@instance-1:~/go/src/k8s.io/arktos$

ubuntu@instance-1:~/go/src/k8s.io/arktos$ ./cluster/kubectl.sh get dividers -Ao wide
No resources found.
ubuntu@instance-1:~/go/src/k8s.io/arktos$
```

## 7) Check bouncers

```
./cluster/kubectrl.sh get bouncers -Ao wide
```

### Output

```
ubuntu@instance-1:~/go/src/k8s.io/arktos$ ./cluster/kubectrl.sh get bouncers -Ao wide
NAMESPACE NAME VPC NET IP MAC DROPLET STATUS CREATETIME PROVISIONDELAY
default net0-b-48f8d246-7aac-486e-b367-ffed1ec49a5e vpc0 net0 instance-1 Provisioned 2021-09-27T06:48:04.321609 0.95841
ubuntu@instance-1:~/go/src/k8s.io/arktos$
```

## 8) Pod deployment:

### Output

```
ubuntu@instance-1:~/go/src/k8s.io/arktos$ ./cluster/kubectrl.sh get pods -Ao wide
NAMESPACE NAME HASHKEY READY STATUS RESTARTS AGE IP NODE NOMINATED NODE READINESS GATES
default mizar-daemon-8bsxq 7735477782612895179 1/1 Running 0 8m3s 10.128.15.197 instance-1 <none> <none>
default mizar-operator-6985d77546-82nqg 1650833051907194888 1/1 Running 0 8m3s 10.128.15.197 instance-1 <none> <none>
default pod-767686bb54-5bg6s 5774664824995397753 0/1 ContainerCreating 0 6m45s <none> instance-1 <none> <none>
kube-system coredns-default-54895b5bfc-qhpkw 6167013444726112653 0/1 ContainerCreating 0 8m3s <none> instance-1 <none> <none>
kube-system kube-dns-7f4bf79dc-zcjdj 4360096557658664667 0/3 ContainerCreating 0 8m3s <none> instance-1 <none> <none>
kube-system virtlet-d8gb4 3236308487394719128 3/3 Running 0 8m2s 10.128.15.197 instance-1 <none> <none>
```

Pod getting stuck in **ContainerCreating** state.