

Test report - Deployment of Arktos Cluster without Mizar CNI on GCP

This document captures the steps to deploy an Arktos cluster lab without 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.

Install go lang 1.13.9

Date-13 Dec. 2021

Create an instance on GCP

Created instance on GCP

| | | | | | | | | | |
|--------------------------|-------------------------------------|-------------|---------------|-------------------------------------|---------------|--------------------|----------------------------------|-------|---|
| <input type="checkbox"/> | <input checked="" type="checkbox"/> | prajwal-ark | us-central1-a | Dec 13, 2021, 10:36:30 AM UTC+05:30 | e2-standard-8 | 10.128.0.12 (nic0) | 104.198.130.82 ↗ | SSH ▾ | ⋮ |
|--------------------------|-------------------------------------|-------------|---------------|-------------------------------------|---------------|--------------------|----------------------------------|-------|---|

SSH instance with credentials.

Step-1: Update kernel (If required)

To check kernel, run following command

```
uname -a
```

```
root@prajwal-ark:~# uname -a
Linux prajwal-ark 5.4.0-1058-gcp #62~18.04.1-Ubuntu SMP Mon Nov 15 07:49:04 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
```

```

Continue kernel update (y/n)?y
Updating kernel
Selecting previously unselected package linux-headers-5.6.0-rc2.
(Reading database ... 71135 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) ...
Preparing to unpack .../linux-libc-dev_5.6.0-rc2-1_amd64.deb ...
Unpacking linux-libc-dev:amd64 (5.6.0-rc2-1) over (4.15.0-163.171) ...
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-5.4.0-1058-gcp
Found initrd image: /boot/initrd.img-5.4.0-1058-gcp
Adding boot menu entry for EFI firmware configuration
done

```

Step-2: Install dependencies

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

```
mkdir -p $GOPATH/src/github.com
```

```
cd $GOPATH/src/github.com
```

```
git clone https://github.com/CentaurusInfra/arktos
```

```
cd arktos
```

```
make
```

```

root@prajwal-ark:~# mkdir -p $GOPATH/src/github.com
root@prajwal-ark:~# cd $GOPATH/src/github.com
root@prajwal-ark:/src/github.com# git clone https://github.com/CentaurusInfra/arktos
Cloning into 'arktos'...
remote: Enumerating objects: 104329, done.
remote: Counting objects: 100% (887/887), done.
remote: Compressing objects: 100% (535/535), done.
remote: Total 104329 (delta 447), reused 603 (delta 327), pack-reused 103442
Receiving objects: 100% (104329/104329), 208.63 MiB | 29.88 MiB/s, done.
Resolving deltas: 100% (62941/62941), done.
Checking out files: 100% (20766/20766), done.
root@prajwal-ark:/src/github.com# cd arktos
root@prajwal-ark:/src/github.com/arktos# make
+++ [1213 05:42:52] Building go targets for linux/amd64:
./vendor/k8s.io/code-generator/cmd/deepcopy-gen
+++ [1213 05:43:00] Building go targets for linux/amd64:
./vendor/k8s.io/code-generator/cmd/defaulter-gen
+++ [1213 05:43:07] Building go targets for linux/amd64:
./vendor/k8s.io/code-generator/cmd/conversion-gen
+++ [1213 05:43:18] Building go targets for linux/amd64:
./vendor/k8s.io/kube-openapi/cmd/openapi-gen
+++ [1213 05:43:29] Building go targets for linux/amd64:
./vendor/github.com/go-bindata/go-bindata/go-bindata

```

Install Docker & containerd:

```
sudo apt-get remove docker docker-engine docker.io containerd runc
```

```
sudo apt-get update
```

```
sudo apt-get install \
```

```
ca-certificates \
```

```
curl \
```

```
gnupg \
```

```
lsb-release
```

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o  
/usr/share/keyrings/docker-archive-keyring.gpg
```

```
echo \
```

```
"deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-  
keyring.gpg] https://download.docker.com/linux/ubuntu \
```

```
$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

```
sudo apt-get update
```

```
sudo apt-get install docker-ce docker-ce-cli containerd.io
```

```
systemctl status docker
```

```
apt-get install containerd
```

```
root@prajwal-ark:/src/github.com/arktos# sudo apt-get remove docker docker-engine docker.io containerd runc
Reading package lists... Done
Building dependency tree
Reading state information... Done
Package 'docker-engine' is not installed, so not removed
Package 'docker' is not installed, so not removed
Package 'docker.io' is not installed, so not removed
The following package was automatically installed and is no longer required:
  libnuma1
Use 'sudo apt autoremove' to remove it.
The following packages will be REMOVED:
  containerd runc
0 upgraded, 0 newly installed, 2 to remove and 7 not upgraded.
After this operation, 167 MB disk space will be freed.
Do you want to continue? [Y/n] y
(Reading database ... 101855 files and directories currently installed.)
Removing containerd (1.5.5-0ubuntu3~18.04.1) ...
Removing runc (1.0.1-0ubuntu2~18.04.1) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
root@prajwal-ark:/src/github.com/arktos# sudo apt-get update
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://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Fetched 252 kB in 1s (501 kB/s)
Reading package lists... Done
root@prajwal-ark:/src/github.com/arktos# sudo apt-get install \
> ca-certificates \
```

Run Arktos

The easiest way to run Arktos is to bring up a single-node cluster in your local development box:

```
cd $GOPATH/src/github.com/arktos
```

hack/arktos-up.sh

```
root@prajwal-ark:/src/github.com/artkos# hack/artkos-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,MandatoryArtkosNetwork=true
DBG: effective disabling admission plugins
DBG: effective default network template file is /src/github.com/artkos/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.
cni plugin is bridge; artkos 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
./
./flannel
./ptp
./host-local
./firewall
./portmap
./tuning
./vlan
./host-device
./bandwidth
./sbr
./static
./dhcp
./ipvlan
./macvlan
./loopback
./bridge
2021-12-13 05:59:19 URL=https://objects.githubusercontent.com/github-production-release-asset-2e65be/84575398/53318dd0-bf61-11ea-9a9f-40364582ae9d?X-Amz-Alc
r=AKIAIWNJXAXCNSUHC5H3KZP2Q2T12133ZP2EUS&host=us-east-1%2Faws-4-request&X-Amz-Date=20211213T055919Z&X-Amz-Expires=300&X-Amz-Signa
t=F2i17236ec924af1f245376d24562a3da0675537927181a8465e2a5edea38065X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=84575398&response-content-disposition=a
ct;38320filenametgz;cni-plugins-linux-amd64-v0.8.2.tgz&response-content-type=application%2Fortet-stream [36662740/36662740] -> "" [1]
setting up cni conf file
```

Deployment failed:

[illegible]

Terminal was stuck in above mentioned state and nodes too were not getting ready

```
NAME          STATUS    ROLES    AGE    VERSION
prajwal-ark   NotReady  <none>   4h13m  v0.9.0
root@prajwal-ark:/src/github.com/arktos#
```