

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

This document captures the steps to deploy an Arktos cluster lab without Mizar CNI. The machine in this lab used are t2.2xlarge (8 vCPUs, 32 GB memory) and the storage size is 128GB), Ubuntu 18.04 LTS.

Install golang 1.13.9

Date-13 Dec. 2021

Create an instance on AWS

Created instance on AWS

<input type="checkbox"/>	prajwal-arktos	i-0a95f2aa1ced16754	Running	🔍	t2.2xlarge	2/2 checks passed	No alarms	+	us-east-2b	ec2-3-134-255-184.us-...
--------------------------	----------------	---------------------	---------	---	------------	-------------------	-----------	---	------------	--------------------------

SSH instance with credentials.

Step-1: Update kernel (If required)

To check kernel, run following command

```
uname -a
```

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

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@ip-172-31-22-171:/src/github.com# cd arktos
root@ip-172-31-22-171:/src/github.com/arktos# make
Running copyright check for repo: /src/github.com/arktos, logging to _output/ArktosCopyrightTool.log
/src/github.com/arktos /src/github.com/arktos
warning: inexact rename detection was skipped due to too many files.
warning: you may want to set your diff.renameLimit variable to at least 3067 and retry the command.
/src/github.com/arktos
/src/github.com/arktos /src/github.com/arktos
warning: inexact rename detection was skipped due to too many files.
warning: you may want to set your diff.renameLimit variable to at least 3067 and retry the command.
/src/github.com/arktos
Inspecting copyright files, writing logs to _output/ArktosCopyrightTool.log
Done.
+++ [1213 12:08:58] Building go targets for linux/amd64:
cmd/kube-proxy
cmd/kube-apiserver
cmd/kube-controller-manager
cmd/cloud-controller-manager
cmd/workload-controller-manager
cmd/kubelet
cmd/kubeadm
cmd/hyperkube
cmd/kube-scheduler
vendor/k8s.io/apiextensions-apiserver
cluster/gce/gci/mounter
cmd/kubectl
cmd/gendocs
cmd/genkubedocs
cmd/genman
cmd/genyaml
cmd/genswaggertypedocs
cmd/linkcheck
vendor/github.com/onsi/ginkgo/ginkgo
test/e2e/e2e.test
cmd/kubemark
vendor/github.com/onsi/ginkgo/ginkgo
test/e2e_node/e2e_node.test

```

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@ip-172-31-22-171:~# 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 'containerd' is not installed, so not removed
Package 'docker.io' is not installed, so not removed
Package 'runc' is not installed, so not removed
0 upgraded, 0 newly installed, 0 to remove and 32 not upgraded.
root@ip-172-31-22-171:~# sudo apt-get update
Hit:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:3 http://us-east-2.ec2.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Hit:4 https://download.docker.com/linux/ubuntu bionic InRelease
Get:5 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Fetched 252 kB in 0s (542 kB/s)
Reading package lists... Done
root@ip-172-31-22-171:~# sudo apt-get install \
> ca-certificates \
> curl \
> gnupg \
> lsb-release
Reading package lists... Done
Building dependency tree
Reading state information... Done
lsb-release is already the newest version (9.20170808ubuntu1).
ca-certificates is already the newest version (20210119~18.04.2).
curl is already the newest version (7.58.0-2ubuntu3.16).
gnupg is already the newest version (2.2.4-1ubuntu1.4).
0 upgraded, 0 newly installed, 0 to remove and 32 not upgraded.
root@ip-172-31-22-171:~# curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
echo \
```

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@ip-172-31-22-171:/src/github.com/arktos# cd $GOPATH/src/github.com/arktos
root@ip-172-31-22-171:/src/github.com/arktos# hack/arktos-up.sh
DBG: Flannel CNI plugin will be installed AFTER cluster is up
DBG: effective feature gates AllAlpha=false,WorkloadInfoDefaulting=true,QPSDoubleGCCController=true,QPSDoubleRSCController=true,MandatoryArktosNetwork=true
DBG: effective disabling admission plugins
DBG: effective default network template file is /src/github.com/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.
cni plugin is bridge; arktos will use bridge to provision pod network
Ensuring firewall to allow traffic forward by default
-P FORWARD ACCEPT
Ensuring minimum cni plugin installation...
found bridge, host-local, loopback
setting up cni conf file...
/etc/cni/net.d/bridge.conf already exists; keep it.
done with bridge cni plugin installation
WARNING : The kubelet is configured to not fail even if swap is enabled; production deployments should disable swap.
Getting runtime deployment file libvirt-qemu
--2021-12-13 12:10:48-- https://raw.githubusercontent.com/futurewei-cloud/arktos-vm-runtime/release-0.6/deploy/apparmor/libvirt-qemu
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.109.133, 185.199.110.133, 185.199.111.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.109.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 6578 (6.4K) [text/plain]
Saving to: '/tmp/libvirt-qemu'

/tmp/libvirt-qemu      100%[=====] 6.42K  --.-KB/s  in 0s
```

Deployment failed:

```
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
Waiting for node ready at api server
```

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

```
root@ip-172-31-22-171:/src/github.com/arktos# ./cluster/kubectll.sh get nodes
NAME                STATUS    ROLES    AGE   VERSION
ip-172-31-22-171    NotReady <none>    6m5s  v0.9.0
root@ip-172-31-22-171:/src/github.com/arktos# ./cluster/kubectll.sh get nodes
NAME                STATUS    ROLES    AGE   VERSION
ip-172-31-22-171    NotReady <none>    6m6s  v0.9.0
root@ip-172-31-22-171:/src/github.com/arktos#
```