Arktos deployment with Mizar CNI

This document is intended for new users to install the Arktos platform with Mizar as the underlying network technology.

Prepare lab machine, the 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.

1. Check the kernel version:

Command:

```
uname -a
```

Output:

```
Linux ip-172-31-27-46 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
```

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

Commands:

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

```
sudo bash kernelupdate.sh
```

```
uname -a
```

Output:

```
ubuntu@ip-172-31-27-46:~$ wget https://raw.githubusercontent.com/CentaurusInfra/mizar/dev-next/kernelupdate.sh
--2021-10-23 14:40:14--  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

2021-10-23 14:40:14 (43.0 MB/s) - 'kernelupdate.sh' saved [791/791]
```

```
ubuntu@ip-172-31-27-46:~$ sudo bash kernelupdate.sh
--2021-10-23 14:41:02--  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)... 52.217.203.169
Connecting to mizar.s3.amazonaws.com (mizar.s3.amazonaws.com)|52.217.203.169|: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 100%[===================================================================>]   7.27M  6.10MB/s    in 1.2s

2021-10-23 14:41:04 (6.10 MB/s) - '../linux-5.6-rc2/linux-headers-5.6.0-rc2_5.6.0-rc2-1_amd64.deb' saved [7621020/7621020]

--2021-10-23 14:41:04--  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)... 52.217.203.169
Connecting to mizar.s3.amazonaws.com (mizar.s3.amazonaws.com)|52.217.203.169|:443... connected.
```

```
Found initrd image: /boot/initrd.img-5.6.0-rc2
Found linux image: /boot/vmlinuz-5.4.0-1045-aws
Found initrd image: /boot/initrd.img-5.4.0-1045-aws
done
Setting up linux-libc-dev:amd64 (5.6.0-rc2-1) ...
Reboot host (y/n)?y
Rebooting
```

```
ubuntu@ip-172-31-27-46:~$ uname -a
Linux ip-172-31-27-46 5.6.0-rc2 #1 SMP Tue Feb 25 18:54:05 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
```

2. Clone the Arktos repository and install the required dependencies:

git clone https://github.com/Click2Cloud-Centaurus/arktos.git ~/go/src/k8s.io/arktos

cd ~/go/src/k8s.io/arktos

git checkout cni-mizar

sudo bash ./hack/setup-dev-node.sh

Output:

```
ubuntu@ip-172-31-27-46:~$ git clone https://github.com/Click2Cloud-Centaurus/arktos.git ~/go/src/k8s.io/arktos
Cloning into '/home/ubuntu/go/src/k8s.io/arktos'...
remote: Enumerating objects: 61570, done.
remote: Counting objects: 100% (974/974), done.
remote: Compressing objects: 100% (569/569), done.
remote: Total 61570 (delta 554), reused 612 (delta 388), pack-reused 60596
Receiving objects: 100% (61570/61570), 221.49 MiB | 25.97 MiB/s, done.
Resolving deltas: 100% (37714/37714), done.
Checking out files: 100% (20761/20761), done.
ubuntu@ip-172-31-27-46:~$ cd ~/go/src/k8s.io/arktos
ubuntu@ip-172-31-27-46:~/go/src/k8s.io/arktos$ git checkout cni-mizar
error: pathspec 'cni-mizar' did not match any file(s) known to git.
ubuntu@ip-172-31-27-46:~/go/src/k8s.io/arktos$ sudo bash ./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-west-2.ec2.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://us-west-2.ec2.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:3 http://us-west-2.ec2.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:4 http://us-west-2.ec2.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-west-2.ec2.archive.ubuntu.com/ubuntu bionic/universe Translation-en [4941 kB]
Get:7 http://us-west-2.ec2.archive.ubuntu.com/ubuntu bionic/multiverse amd64 Packages [151 kB]
```

Command:

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

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

source ~/.profile

Output:

```
ubuntu@ip-172-31-27-46:~/go/src/k8s.io/arktos$ echo export PATH=$PATH:/usr/local/go/bin\ >> ~/.profile
ubuntu@ip-172-31-27-46:~/go/src/k8s.io/arktos$ echo cd \$HOME/go/src/k8s.io/arktos >> ~/.profile
ubuntu@ip-172-31-27-46:~/go/src/k8s.io/arktos$ source ~/.profile
ubuntu@ip-172-31-27-46:~/go/src/k8s.io/arktos$ ▮
```

3. Start Arktos cluster Error

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

```
ubuntu@ip-172-31-27-46:~/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,Mandatory
ArktosNetwork=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 mizar; arktos onebox cannot use it out of box.
suggest to use bridge cni plugin.
if you really want to use this plugin, you need to config cni plugin by yourself.
ubuntu@ip-172-31-27-46:~/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,Mandatory
ArktosNetwork=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 mizar; arktos onebox cannot use it out of box.
suggest to use bridge cni plugin.
if you really want to use this plugin, you need to config cni plugin by yourself.
ubuntu@ip-172-31-27-46:~/go/src/k8s.io/arktos$ sudo 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,Mandatory
ArktosNetwork=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.
cni plugin is mizar; arktos onebox cannot use it out of box.
suggest to use bridge cni plugin.
if you really want to use this plugin, you need to config cni plugin by yourself.
ubuntu@ip-172-31-27-46:~/go/src/k8s.io/arktos$ pwd
/home/ubuntu/go/src/k8s.io/arktos
```

EC2 instance is rebooted 3 times and run the command "CNIPLUGIN=mizar ./hack/arktos-up.sh**" on every reboot, but still got the same error.**