

K8S кластер

Задание в процессе выполнения

Создаём одну VM, которую определим под master-ноду.

Прежде всего, устанавливаем на неё CRI в виде containerd. Можно было бы поставить docker, который подтянет containerd, с помощью пакетного менеджера. Но в k8s нам не нужны средства сборки образов и проч. функционал Docker, кроме как container runtime.

Перед установкой сервисов kubernetes временно отключим swap по рекомендации разработчиков k8s.

```
swapoff -a
```

Поставим сначала kubeadm, kubelet, kubectl [по официальной документации](#).

Установка утилит для кубера

```
#
sudo apt-get update

# ,
sudo apt-get install -y apt-transport-https ca-certificates curl gpg

# Kubernetes
curl -fsSL https://pkgs.k8s.io/core:/stable:/v1.31/deb/Release.key | sudo gpg --dearmor -o /etc/apt/keyrings/
/kubernetes-apt-keyring.gpg

# Kubernetes
echo 'deb [signed-by=/etc/apt/keyrings/kubernetes-apt-keyring.gpg] https://pkgs.k8s.io/core:/stable:/v1.31/deb/
/' | sudo tee /etc/apt/sources.list.d/kubernetes.list

# apt
sudo apt-get update

# ""
sudo apt-mark hold kubelet kubeadm kubectl

# () kubelet
sudo systemctl enable --now kubelet
```

Затем поставим CRI

```
# containerd
wget https://github.com/containerd/containerd/releases/download/v1.7.0/containerd-1.7.0-linux-amd64.tar.gz
tar Cxzvf /usr/local containerd-1.7.0-linux-amd64.tar.gz
rm containerd-1.7.0-linux-amd64.tar.gz

# containerd
mkdir /etc/containerd/
containerd config default > /etc/containerd/config.toml

# cgroup
sed -i 's/SystemdCgroup \= false/SystemdCgroup \= true/g' /etc/containerd/config.toml

# systemd containerd
wget https://raw.githubusercontent.com/containerd/containerd/main/containerd.service
mv containerd.service /etc/systemd/system/

# runc
wget https://github.com/opencontainers/runc/releases/download/v1.1.4/runc.amd64
install -m 755 runc.amd64 /usr/local/sbin/runc
rm runc.amd64

# :
wget https://github.com/containerd/containerd/releases/download/v1.2.0/cni-plugins-linux-amd64-v1.2.0.tgz
mkdir -p /opt/cni/bin
tar Cxzvf /opt/cni/bin cni-plugins-linux-amd64-v1.2.0.tgz
rm cni-plugins-linux-amd64-v1.2.0.tgz

# containerd
systemctl daemon-reload
systemctl enable --now containerd
```

Инициализируем нашу VM как master-node:

```
kubeadm init --pod-network-cidr=192.168.0.0/16
```

```
vsevolod@k8s-pc:~$ sudo -i
[sudo] password for vsevolod:
root@k8s-pc:~# kubeadm init --pod-network-cidr=192.168.0.0/16
[init] Using Kubernetes version: v1.31.0
[preflight] Running pre-flight checks
error execution phase preflight: [preflight] Some fatal errors occurred:
[ERROR FileContent--proc-sys-net-ipv4-ip_forward]: /proc/sys/net/ipv4/ip_forward contents are not set to 1
[preflight] If you know what you are doing, you can make a check non-fatal with `--ignore-preflight-errors=...`
To see the stack trace of this error execute with --v=5 or higher
```

Включим IPv4 packet forwarding:

```
# sysctl params required by setup, params persist across reboots
cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf
net.ipv4.ip_forward = 1
EOF

# Apply sysctl params without reboot
sudo sysctl --system
```

Укажем переменную окружения KUBECONFIG, которая указывает на путь к файлу /etc/kubernetes/admin.conf. Это временно указывает kubelet, что нужно использовать этот файл конфигурации для подключения к кластеру.

```
export KUBECONFIG=/etc/kubernetes/admin.conf
```

Проверим состояние нашего кластера

```
root@k8s-pc:~# kubectl get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
k8s-pc	NotReady	control-plane	47m	v1.31.0

Master-нода имеет статус NotReady. Посмотрим логи сервиса kubelet

```
root@k8s-pc:~# kubectl get nodes
NAME          STATUS    ROLES    AGE   VERSION
k8s-pc        NotReady  control-plane  46m   v1.31.0
root@k8s-pc:~# journalctl -u kubelet -f
Sep 04 10:53:08 k8s-pc kubelet[2190]: E0904 10:53:08.477794 2190 kubelet.go:2901] "Container runtime network not ready" networkReady="NetworkReady=false reason:NetworkP
loginNotReady message:Network plugin returns error: cni plugin not initialized"
Sep 04 10:53:09 k8s-pc kubelet[2190]: E0904 10:53:09.346881 2190 dns.go:153] "Nameserver limits exceeded" err="Nameserver limits were exceeded, some nameservers have be
en omitted, the applied nameserver line is: 1.1.1.1 10.20.33.1 8.8.8.8"
Sep 04 10:53:13 k8s-pc kubelet[2190]: E0904 10:53:13.487801 2190 kubelet.go:2901] "Container runtime network not ready" networkReady="NetworkReady=false reason:NetworkP
loginNotReady message:Network plugin returns error: cni plugin not initialized"
Sep 04 10:53:18 k8s-pc kubelet[2190]: E0904 10:53:18.491442 2190 kubelet.go:2901] "Container runtime network not ready" networkReady="NetworkReady=false reason:NetworkP
loginNotReady message:Network plugin returns error: cni plugin not initialized"
```

Проблема связана с тем, что CNI ещё не настроен.

Теперь установим Calico CNI. Просто применяем манифесты, с описанием кастомных ресурсов и их определением

```
kubectl create -f https://raw.githubusercontent.com/projectcalico/calico/v3.28.1/manifests/tigera-operator.yaml
kubectl create -f https://raw.githubusercontent.com/projectcalico/calico/v3.28.1/manifests/custom-resources.yaml
kubectl taint nodes --all node-role.kubernetes.io/control-plane-
```

```
root@k8s-pc:~# kubectl get pod -A
```

NAMESPACE	NAME	READY	STATUS	RESTARTS	AGE
calico-apiserver	calico-apiserver-6485596bb8-82zvz	0/1	ContainerCreating	0	4s
calico-apiserver	calico-apiserver-6485596bb8-sfpr2	0/1	ContainerCreating	0	4s
calico-system	calico-kube-controllers-7c87479d7-qhbmh	1/1	Running	0	115s
calico-system	calico-node-cj4hg	1/1	Running	0	100s
calico-system	calico-typha-644d864f77-svg9v	1/1	Running	0	115s
calico-system	csi-node-driver-285t6	2/2	Running	0	117s
kube-system	coredns-6f6b679f8f-drgsp	1/1	Running	0	78m
kube-system	coredns-6f6b679f8f-wlj77	1/1	Running	0	78m
kube-system	etcd-k8s-pc	1/1	Running	0	79m
kube-system	kube-apiserver-k8s-pc	1/1	Running	0	79m
kube-system	kube-controller-manager-k8s-pc	1/1	Running	0	79m
kube-system	kube-proxy-rm82v	1/1	Running	0	78m
kube-system	kube-scheduler-k8s-pc	1/1	Running	0	79m
tigera-operator	tigera-operator-55748b469f-llt6p	1/1	Running	0	3m56s

CoreDns после появления CNI настроены.