

CRI-O기반의 k8s 설치

사전사항

1. OS환경설정

```
$> swapoff -a

$> cat <<EOF | sudo tee /etc/modules-load.d/k8s.conf
br_netfilter
EOF

$> cat <<EOF | sudo tee /etc/sysctl.d/k8s.conf
net.bridge.bridge-nf-call-ip6tables = 1
net.bridge.bridge-nf-call-iptables = 1
EOF

$> sudo sysctl --system
```

2. crio / kubernetes 패키지 리포지터리 구성

```
$> cat /etc/yum.repos.d/libcontainers.repo
[devel_kubic_libcontainers_stable]
name=Stable Releases of Upstream github.com/containers packages (CentOS_8)
type=rpm-md
baseurl=https://download.opensuse.org/repositories/devel:/kubic:/libcontainers:/stable/CentOS_8/
gpgcheck=1
gpgkey=https://download.opensuse.org/repositories/devel:/kubic:/libcontainers:/stable/CentOS_8/repodata/repomd.xml.key
enabled=1
```

```
$> cat /etc/yum.repos.d/cri-o.repo
[cri-o]
name=CRI-O
baseurl=https://pkgs.k8s.io/addons:/cri-o:/stable:/v1.28/rpm/
enabled=1
gpgcheck=1
gpgkey=https://pkgs.k8s.io/addons:/cri-o:/stable:/v1.28/rpm/repodata/repomd.xml.key
```

 k8s 1.24이후 버전에서는 repository url이 변경되었습니다.

```
$> cat /etc/yum.repos.d/k8s.repo
[kubernetes]
name=Kubernetes
baseurl=https://pkgs.k8s.io/core:/stable:/v1.28/rpm/
enabled=1
gpgcheck=1
gpgkey=https://pkgs.k8s.io/core:/stable:/v1.28/rpm/repodata/repomd.xml.key
exclude=kubelet kubeadm kubectl cri-tools kubernetes-cni
```

3. 패키지 설치

```
$> yum install -y cri-o
$> yum install -y kubelet kubeadm kubectl --disableexcludes=kubernetes
$> systemctl enable crio --now
$> systemctl enable kubelet
```

클러스터 생성 (control plain 1번에서만 수행)

1. kubeadm 클러스터 생성

```
$> kubeadm init --control-plane-endpoint 172.21.107.238:6443 --pod-network-cidr 10.250.0.0/16 --ignore-preflight-errors=all --upload-certs
```

결과값중에 control / worker 노드별 join 명령이 다르기 때문에 별도로 복사해놓어야 함

##Control node용

```
$> kubeadm join 172.21.107.238:6443 --token abcd \
--discovery-token-ca-cert-hash sha256:yyy \
--control-plane --certificate-key zzz
```

Worker Node용

```
$> kubeadm join 172.21.107.238:6443 --token abcd \
--discovery-token-ca-cert-hash sha256:yyyy \
--cri-socket \
--ignore-preflight-errors=all
```

2. 인증서 정보 복사

```
$> mkdir -p $HOME/.kube
$> /bin/cp /etc/kubernetes/admin.conf $HOME/.kube/config
$> chown $(id -u):$(id -g) $HOME/.kube/config
$> export KUBECONFIG=/etc/kubernetes/admin.conf
```

3. CNI 설치(Calico)

```
$> curl https://calico-v3-25.netlify.app/archive/v3.25/manifests/calico.yaml -O
$> kubectl apply -f calico.yaml
```

클러스터 연동

1. 타 Control plain 연동 (Control Plain 한대씩 순차 작업 수행)

```
$> kubeadm join 172.21.107.238:6443 --token abcd \
--discovery-token-ca-cert-hash sha256:yyy \
--control-plane --certificate-key zzz
```

2. 노드 연동 확인 (Control plain에서 수행)

```
$> kubectl get no
NAME                STATUS  ROLES                AGE  VERSION
k8stesttx-k8s-master-dev01  Ready  control-plane,master  3h6m v1.23.5
k8stesttx-k8s-master-dev02  Ready  control-plane,master  3h6m v1.23.5
k8stesttx-k8s-master-dev03  Ready  control-plane,master  3h6m v1.23.5
```

3. Worker Node 연동

```
$> kubeadm join 172.21.107.238:6443 --token abcd \
--discovery-token-ca-cert-hash sha256:yyyy \
--ignore-preflight-errors=all
```

4. 노드 연동 확인 (Control plain에서 수행)

```
$> kubectl get no
NAME                STATUS  ROLES    AGE  VERSION
...
k8stesttx-k8s-worker-dev01  Ready  <none>    40m  v1.23.5
k8stesttx-k8s-worker-dev02  Ready  <none>    40m  v1.23.5
k8stesttx-k8s-worker-dev03  Ready  <none>    40m  v1.23.5
```

k8s 인증서 10년으로 연장

1. 인증서 연장 스크립트

```
$> git clone https://github.com/yuyicai/update-kube-cert.git
$> cd update-kube-cert
$> chmod 755 update-kubeadm-cert.sh
$> ./update-kubeadm-cert.sh all
```

2. 업데이트 전 인증서 정보 확인

```
$> kubeadm certs check-expiration
[check-expiration] Reading configuration from the cluster...
[check-expiration] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config -o yaml'
```

CERTIFICATE	EXPIRES	RESIDUAL TIME	CERTIFICATE AUTHORITY	EXTERNALLY MANAGED
admin.conf	Apr 17, 2023 06:09 UTC	364d	ca	no
apiserver	Apr 17, 2023 06:09 UTC	364d	ca	no
apiserver-etcd-client	Apr 17, 2023 06:09 UTC	364d	etcd-ca	no
apiserver-kubelet-client	Apr 17, 2023 06:09 UTC	364d	ca	no
controller-manager.conf	Apr 17, 2023 06:09 UTC	364d	ca	no
etcd-healthcheck-client	Apr 17, 2023 06:09 UTC	364d	etcd-ca	no
etcd-peer	Apr 17, 2023 06:09 UTC	364d	etcd-ca	no
etcd-server	Apr 17, 2023 06:09 UTC	364d	etcd-ca	no
front-proxy-client	Apr 17, 2023 06:09 UTC	364d	front-proxy-ca	no
scheduler.conf	Apr 17, 2023 06:09 UTC	364d	ca	no

CERTIFICATE AUTHORITY	EXPIRES	RESIDUAL TIME	EXTERNALLY MANAGED
ca	Apr 17, 2032 04:17 UTC	9y	no
etcd-ca	Apr 17, 2032 04:17 UTC	9y	no
front-proxy-ca	Apr 17, 2032 04:17 UTC	9y	no

3. 인증서 업데이트 (Control Plain 1대씩 순차 작업 수행, 서버단위로 30초 가량 대기 필요)

```
$> chmod +x cert_update.sh
$> ./cert_update.sh
...
```

4. 인증서 갱신정보 확인

```
$> kubeadm certs check-expiration
[check-expiration] Reading configuration from the cluster...
[check-expiration] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config -o yaml'
```

CERTIFICATE	EXPIRES	RESIDUAL TIME	CERTIFICATE AUTHORITY	EXTERNALLY MANAGED
admin.conf	Apr 17, 2032 06:09 UTC	9y	ca	no
apiserver	Apr 17, 2032 06:09 UTC	9y	ca	no
apiserver-etcd-client	Apr 17, 2032 06:09 UTC	9y	etcd-ca	no
apiserver-kubelet-client	Apr 17, 2032 06:09 UTC	9y	ca	no
controller-manager.conf	Apr 17, 2032 06:09 UTC	9y	ca	no
etcd-healthcheck-client	Apr 17, 2032 06:09 UTC	9y	etcd-ca	no
etcd-peer	Apr 17, 2032 06:09 UTC	9y	etcd-ca	no
etcd-server	Apr 17, 2032 06:09 UTC	9y	etcd-ca	no
front-proxy-client	Apr 17, 2032 06:09 UTC	9y	front-proxy-ca	no
scheduler.conf	Apr 17, 2032 06:09 UTC	9y	ca	no

CERTIFICATE AUTHORITY	EXPIRES	RESIDUAL TIME	EXTERNALLY MANAGED
ca	Apr 17, 2032 04:17 UTC	9y	no
etcd-ca	Apr 17, 2032 04:17 UTC	9y	no
front-proxy-ca	Apr 17, 2032 04:17 UTC	9y	no

Reference

- <https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/install-kubeadm/>
- <https://github.com/yuyicai/update-kube-cert.git>

☹Revision #11

★Created 8 June 2022 03:08:07 by artop0420

✍Updated 1 September 2024 02:20:36 by artop0420