

Kubernetes na vpsAdminOS

Tento návod funguje iba pre Kubernetes 1.18 a staršie

Prerekvizity

- VPS musí bežať na vpsAdminOS na kerneli min. 5.9.10. V čase písania návodu je dostupný tento kernel iba na staging node
- Postup je odskúšaný na fresh minimal Ubuntu 20.04
- Používame latest vanilla kubernetes
- Ako networking používame flannel s host-gw s internou sieťou 10.244.0.0/16
- Kubernetes si chce šahať do /sys a /proc. Hodnoty máme vo vpsAdminOS správne pre vpsAdminOS, preto mu iba nafakeujeme dané súbory aby si tam sám mohol zapísať. Vytvoríme aj fake service v systemd, aby sa pri štarte systému dané súbory správne namapovali.
- **Inštalčný skript je rovnaký pre master aj worker nody**

Spoločný postup pre master aj worker nodes

Vytvoríme si install.sh v /root.

```
#!/bin/bash -x
apt-get install -y apt-transport-https ca-certificates curl software-properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | apt-key add -
add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"
apt-get update
apt-get install -y docker-ce iptables arptables ebttables

wget -q https://storage.googleapis.com/golang/getgo/installer_linux
chmod +x installer_linux
./installer_linux
source /root/.bash_profile

update-alternatives --set iptables /usr/sbin/iptables-legacy
update-alternatives --set ip6tables /usr/sbin/ip6tables-legacy
update-alternatives --set arptables /usr/sbin/arptables-legacy
update-alternatives --set ebttables /usr/sbin/ebttables-legacy
apt-get update && apt-get install -y apt-transport-https curl
curl -s https://packages.cloud.google.com/apt/doc/apt-key.gpg | apt-key add -
cat <<EOF | tee /etc/apt/sources.list.d/kubernetes.list
deb https://apt.kubernetes.io/ kubernetes-xenial main
EOF
```

```
apt-get update
apt-get install -y kubelet kubeadm kubectl

mkdir -p /opt/k8s/fake

cat > /opt/k8s/fake.sh <<EOF
#!/bin/bash
cd /opt/k8s/fake
echo 0 > panic
mount --bind panic /proc/sys/kernel/panic
echo 0 > panic_on_oops
mount --bind panic_on_oops /proc/sys/kernel/panic_on_oops
echo 0 > overcommit_memory
mount --bind overcommit_memory /proc/sys/vm/overcommit_memory

mkdir block
mount -o bind block/ /sys/block/
mount --make-rshared /
EOF

chmod +x /opt/k8s/fake.sh

cat > /etc/systemd/system/fake.service <<EOF
[Unit]
Before=kubelet.service

[Service]
ExecStart=/opt/k8s/fake.sh

[Install]
WantedBy=default.target
EOF

chmod 644 /etc/systemd/system/fake.service

systemctl daemon-reload
systemctl enable fake.service
systemctl start fake.service

cat > /etc/docker/daemon.json <<EOF
{
  "exec-opts": ["native.cgroupdriver=systemd"],
  "log-driver": "json-file",
  "log-opts": {
    "max-size": "100m"
  },
  "storage-driver": "overlay2"
}
EOF

systemctl daemon-reload
```

```
systemctl restart docker  
  
kubeadm config images pull
```

Spustíme inštaláciu a počkáme na úspešné dokončenie základnej inštalácie k8s:

```
chmod +x /root/install.sh  
/root/install.sh
```

Inštalácia je spravená tak, že systém funguje aj po reštarte, avšak trvá 3-5 minút kým znovu nabehnú všetky služby.

Master

Master ma niekoľko špeciálnych krokov. Najprv inicializujeme kubernetes, následne pridáme network.

```
kubeadm init --pod-network-cidr=10.244.0.0/16  
  
mkdir -p $HOME/.kube  
cp -i /etc/kubernetes/admin.conf $HOME/.kube/config  
chown $(id -u):$(id -g) $HOME/.kube/config  
  
wget  
https://raw.githubusercontent.com/coreos/flannel/master/Documentation/kube-flannel.yml  
sed -i 's/vxlan/host-gw/g' kube-flannel.yml  
kubectl apply -f kube-flannel.yml
```

Odporúčam sledovať priebeh deploymentu a počkať, kým budú všetko v stave running s plným počtom:

```
kubectl get pods --all-namespaces
```

Vysledok by mal vyzerat cca:

```
# kubectl --namespace=kube-system get pods  
NAME                                READY   STATUS    RESTARTS   AGE  
coredns-f9fd979d6-f9v99             1/1     Running   1           12m  
coredns-f9fd979d6-v7w2x             1/1     Running   1           12m  
etcd-vps3                            1/1     Running   1           12m  
kube-apiserver-vps3                 1/1     Running   1           12m  
kube-controller-manager-vps3        1/1     Running   1           12m  
kube-flannel-ds-zbc47               1/1     Running   1           12m  
kube-proxy-7zvc5                    1/1     Running   1           12m  
kube-scheduler-vps3                 1/1     Running   1           12m
```

Master ako worker

Ak chceme aby aj master node slúžil ako worker, môžeme ho pridať:

```
kubectl taint nodes --all node-role.kubernetes.io/master-
```

Worker

Získame si najprv na master node token pre pridanie ďalšieho nodu do clustru. Získame tým príkaz ktorý iba copy-paste na worker node:

```
kubeadm token create --print-join-command
```

```
# Ukazka vystupu prikazu kubeadm token create --print-join-command
kubeadm join 37.205.14.241:6443 --token 53r0e7.21pznuukg755rpz3 --
discovery-token-ca-cert-hash
sha256:6be4cb960d16fae2dd7ce96c7a16fc585ce174973c04ded0f91df6cf86681e3a
```

Na master node môžeme sledovať stav nodov:

```
# kubectl get nodes
NAME      STATUS    ROLES    AGE   VERSION
vps3     Ready    master   32m   v1.19.4
vps4     Ready    <none>   21m   v1.19.4
```

From:

<https://kb.vpsfree.cz/> - **Znalostní Báze**

Permanent link:

<https://kb.vpsfree.cz/navody/vps/vpsadminos/kubernetes>

Last update: **2020/12/19 11:03**