# kubernetes1.13.1单master集群快速安装文档

kubernetes 1.13.1 单节点master 集群部署

参考：
https://blog.csdn.net/qq_42006894/article/details/86214085
https://blog.csdn.net/networken/article/details/84991940

集群信息

10.22.60.26 master
10.22.60.172 node01
10.22.60.173 node02

1、安装基本服务（所有节点：master + node）

```
yum install -y net-tools epel-release
yum install -y vim yum-utils device-mapper-persistent-data lvm2
```

2、配置docker-ce 和 k8s yum 源（所有节点：master + node）

```
yum-config-manager --add-repo
http://mirrors.aliyun.com/docker-ce/linux/centos/docker-ce.repo

cat <<EOF > /etc/yum.repos.d/kubernetes.repo
[kubernetes]
name=Kubernetes
baseurl=https://mirrors.aliyun.com/kubernetes/yum/repos/kubernetes-el7-x86
_64/
enabled=1
gpgcheck=1
repo_gpgcheck=1
gpgkey=https://mirrors.aliyun.com/kubernetes/yum/doc/yum-key.gpg
https://mirrors.aliyun.com/kubernetes/yum/doc/rpm-package-key.gpg
EOF
```

3、防火墙、Selinux和swap分区（所有节点：master + node）

```
sudo systemctl stop firewalld.service #firewall
sudo systemctl disable firewalld.service #firewall
sudo swapoff -a
sudo setenforce 0
sudo vi /etc/selinux/config
#SELINUXdisabled
SELINUX=disabled
```

4、安装docker-ce 和k8s（所有节点：master + node）

```
yum install docker-ce-18.06.0.ce
#  yum install docker-cedocker-ce
systemctl enable docker
systemctl start docker



yum install kubectl kubelet kubernetes-cni kubeadm
systemctl enable kubelet
# swap/etc/sysconfig/kubelet swap
vim /etc/sysconfig/kubelet
KUBELET_EXTRA_ARGS="--fail-swap-on=false"
```

5、修改镜像源（所有节点：master + node）

```
vim /etc/systemd/system/kubelet.service.d/10-kubeadm.conf

# Note: This dropin only works with kubeadm and kubelet v1.11+
[Service]
Environment="KUBELET_KUBECONFIG_ARGS=--bootstrap-kubeconfig=/etc/kubernete
s/bootstrap-kubelet.conf --kubeconfig=/etc/kubernetes/kubelet.conf"
Environment="KUBELET_CONFIG_ARGS=--config=/var/lib/kubelet/config.yaml"
# This is a file that "kubeadm init" and "kubeadm join" generates at
runtime, populating the KUBELET_KUBEADM_ARGS variable dynamically
EnvironmentFile=-/var/lib/kubelet/kubeadm-flags.env
# This is a file that the user can use for overrides of the kubelet args as
a last resort. Preferably, the user should use
# the .NodeRegistration.KubeletExtraArgs object in the configuration files
instead. KUBELET_EXTRA_ARGS should be sourced from this file.
# "KUBE_PAUSE" pause"KUBE_PAUSE"
Environment="KUBE_PAUSE=--pod-infra-container-image=10.22.60.25/kubernetes
/pause:3.1"
EnvironmentFile=-/etc/sysconfig/kubelet
ExecStart=
ExecStart=/usr/bin/kubelet $KUBELET_KUBECONFIG_ARGS $KUBELET_CONFIG_ARGS
$KUBELET_KUBEADM_ARGS $KUBELET_EXTRA_ARGS $KUBE_PAUSE
```

6、桥接网络设置（所有节点：master + node）

```
modprobe br_netfilter
cat <<EOF > /etc/sysctl.d/k8s.conf
net.bridge.bridge-nf-call-ip6tables = 1
net.bridge.bridge-nf-call-iptables = 1
EOF
sysctl -p /etc/sysctl.d/k8s.conf
ls /proc/sys/net/bridge
```

7、配置 /etc/hosts（所有节点：master + node）

```
10.22.60.26 ODCBSCMCP01
10.22.60.172 DCK8SNO103
10.22.60.173 DCK8SNO104
```

8、集群初始化（master节点操作：请记录初始化最后打印出的kubeadm join 信息）

```
# --ignore-preflight-errors=Swap swap
kubeadm init \
--kubernetes-version=v1.13.1 \
--pod-network-cidr=10.244.0.0/16 \
--apiserver-advertise-address=10.22.60.26 \
--image-repository=registry.odc.sunline.cn/kubernetes


mkdir -p $HOME/.kube
sudo cp -i /etc/kubernetes/admin.conf $HOME/.kube/config
sudo chown $(id -u):$(id -g) $HOME/.kube/config
```

9、安装calico网络插件

```
wget
https://raw.githubusercontent.com/Lentil1016/kubeadm-ha/1.13.0/calico/rbac
.yaml
wget
https://raw.githubusercontent.com/Lentil1016/kubeadm-ha/1.13.0/calico/cali
co.yaml
"calico.yaml"
"calico.yaml"IP"10.244.0.0/16"
kubectl apply -f rbac.yaml
kubectl apply -f calico.yaml
```

10、安装dashboard

```
wget
https://raw.githubusercontent.com/cherryleo/k8s-apps/master/k8s-dashboard/
kubernetes-dashboard.yaml
"10.22.60.25/kubernetes/kubernetes-dashboard-amd64:v1.10.1"
kubectl apply -f kubernetes-dashboard.yaml
```

11、配置dashboard的登录权限

```
cat admin-user.yaml
apiVersion: v1
kind: ServiceAccount
metadata:
name: admin-user
namespace: kube-system

---
apiVersion: rbac.authorization.k8s.io/v1beta1
kind: ClusterRoleBinding
metadata:
name: admin-user
roleRef:
apiGroup: rbac.authorization.k8s.io
kind: ClusterRole
name: cluster-admin
subjects:
- kind: ServiceAccount
name: admin-user
namespace: kube-system

kubectl apply -f admin-user.yaml

token
kubectl -n kube-system describe secret $(kubectl -n kube-system get secret
| grep admin-user | awk '{print $1}')
```