k8s kubeadm部署高可用集群

`kubeadm` 是官方推出的部署工具,旨在降低kubernetes使用门槛与提高集群部署的便捷性. 同时越来越多的官方文档,围绕kubernetes容器化部署为环境, 所以容器化部署kubernetes已成为趋势.
本文主要内容: 基于 `kubeadm` 部署方式,实现kubernetes的高可用.

master部署

1. 三台master节点上建立etcd集群
2. 使用 `vip` 进行kubeadm初始化master

1. 环境准备

节点	地址
master1,etcd1	10.8.104.16
master2,etcd2	10.8.37.18
master3,etcd3	10.8.125.29
node1	10.8.113.73

操作系统： centos 7..2

`vip:` 10.8.78.31/16

2. 部署etcd集群

三台master节点上部署etcd分布式集群, 部署细节请自行百度.
etcd集群信息 `http://10.8.125.29:2379,http://10.8.104.16:2379,http://10.8.37.18:2379`

3. 编译rpm包

```
yum install docker git -y
systemctl start docker
cd /data
git clone https://github.com/kubernetes/release.git
cd /data/release/rpm
./docker-build.sh
```

4. 安装kubeadm

```
cd /data/release/rpm/output/x86_64
yum localinstall *.rpm -y
systemctl enable docker && systemctl start docker
systemctl enable kubelet && systemctl start kubelet
```

5. 初始化master1

```
#添加vip
ip addr add 10.8.78.31/16 dev eth0
kubeadm init --api-advertise-addresses=10.8.78.31 --external-etcd-endpoints=http://10.8.1
25.29:2379,http://10.8.104.16:2379,http://10.8.37.18:2379
```

`--api-advertise-addresses` 支持多个ip,但是会导致 `kubeadm join` 无法正常加入, 所以对外服务只配置为一个 `vip`

6. 部署其他master

1. 参照master1 安装kubeadm
2. 拷贝master1 的/etc/kubernetes/并启动kubelet

```
scp -r 10.8.104.16:/etc/kubernetes/* /etc/kubernetes/
yum install docker -y
systemctl enable docker && systemctl start docker
systemctl enable kubelet && systemctl start kubelet
```

`kube-controller-manager ``kube-scheduler` 通过 `--leader-elect` 实现了分布式锁. 所以三个master节点可以正常运行.

组件优化

采用 `daemonsets` 方式,实现核心组件实现高可用,

1. dns组件

方案一

```
#1. 在所有master部署dns
kubectl scale deploy/kube-dns  --replicas=3 -n kube-system
```

方案二

```
#1.删除自带dns组件
kubectl delete deploy/kube-dns svc/kube-dns -n kube-system
#2.下载最新的dns组件
cd /data
wget https://raw.githubusercontent.com/kubernetes/kubernetes/master/cluster/addons/dns/kubedns-controller.yaml.base
wget https://raw.githubusercontent.com/kubernetes/kubernetes/master/cluster/addons/dns/kubedns-svc.yaml.base
#3.修改配置
mv kubedns-controller.yaml.base kubedns-daemonsets.yaml
mv kubedns-svc.yaml.base kubedns-svc.yaml
sed -i 's/__PILLAR__DNS__SERVER__/10.96.0.10/g' kubedns-svc.yaml
sed -i 's/__PILLAR__DNS__DOMAIN__/cluster.local/g' kubedns-daemonsets.yaml
```

把Deployment类型改为DaemonSet,并加上master nodeSelector

```
    nodeSelector:
      kubeadm.alpha.kubernetes.io/role: master
```

```
kubectl apply -f kubedns-svc.yaml -f kubedns-daemonsets.yaml
```

2. 网络组件

基于稳定性与兼容性考虑, 采用 `Canal` 作为网络组件

```
wget https://raw.githubusercontent.com/tigera/canal/master/k8s-install/kubeadm/canal.yaml
#1.删掉canal.yaml中关于etcd的部署代码
#2.修改`etcd_endpoints`为已部署的etcd集群`
kubectl apply -f canal.yaml
```

```
etcd_endpoints: "http://10.8.125.29:2379,http://10.8.104.16:2379,http://10.8.37.18:2379"
```

canal启动完毕后，dns组件会处于正常状态

3. kube-discovery

kube-discovery 主要负责集群密钥的分发,如果这个组件不正常, 将无法正常新增节点 `kubeadm join`

方案一

```
kubectl scale deploy/kube-discovery --replicas=3 -n kube-system
```

方案二

```
#1. 导出kube-discovery配置
kubectl get deploy/kube-discovery -n kube-system -o yaml > /data/kube-discovery.yaml
#2. 把Deployment类型改为DaemonSet,并加上master nodeSelector
#3. 删掉自带kube-discovery
kubectl delete deploy/kube-discovery svc/kube-dns -n kube-system
#4. 部署kube-discovery
kubectl apply -f kube-discovery.yaml
```

Deployment转为DaemonSet, 如果报错,请根据报错内容删减配置. 主要是去掉状态配置与 `replicas` 和 `strategy`

4. label node

给所有master节点打上 `role=master` 标签, 以使DaemonSet类型的组件自动部署到所有master节点

```
kubectl label node 10-8-125-29 kubeadm.alpha.kubernetes.io/role=master
kubectl label node 10-8-37-18 kubeadm.alpha.kubernetes.io/role=master
```

vip 漂移

到目前为止,三个master节点 相互独立运行,互补干扰. `kube-apiserver` 作为核心入口, 可以使用 keepalived 实现高可用, `kubeadm join暂时不支持负载均衡的方式`

1. keepalived

```
yum install -y keepalived
```

`/etc/keepalived/keepalived.conf`

```
global_defs {
    router_id LVS_k8s
}

vrrp_script CheckK8sMaster {
    script "curl -k https://10.8.104.16:6443"
    interval 3
    timeout 9
    fall 2
    rise 2
}

vrrp_instance VI_1 {
    state MASTER
    interface eth0
    virtual_router_id 61
    priority 115
    advert_int 1
    mcast_src_ip 10.8.104.16
    nopreempt
    authentication {
        auth_type PASS
        auth_pass sqP05dQgMSlzrxHj
    }
    unicast_peer {
        #10.8.104.16
        10.8.37.18
        10.8.125.29
    }
    virtual_ipaddress {
        10.8.78.31/16
    }
    track_script {
        CheckK8sMaster
    }

}
```

```
systemctl enable keepalived
systemctl restart keepalived
```

keepalived模式为 主-从-从 ,拷贝配置到其他master节点,并做修改:

1. `curl -k https://10.8.104.16:6443` 检查本机kube-apiserver是否正常运行
2. `state MASTER` 另外两个节点为 `state BACKUP`
3. `priority 115` 逐次降低优先级,
4. 修改相应的 ip
5. `systemctl enable keepalived;systemctl restart keepalived`

验证

1. 加入节点

```
cd /data/release/rpm/output/x86_64
yum localinstall *.rpm -y
systemctl enable docker && systemctl start docker
systemctl enable kubelet && systemctl start kubelet
kubeadm join --token=eb6a6d.d3e65ed6e64a5bc6 10.8.78.31
```

```
kubectl get node
NAME            STATUS          AGE
10-8-104-16     Ready,master    h
10-8-113-73     Ready           h
10-8-125-29     Ready,master    h
10-8-37-18      Ready,master    h
```

2. 验证master宕机影响

```
#查看当前vip所在的节点
ip a
```

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1454 qdisc pfifo_fast state UP qlen 1000
    link/ether 52:54:00:bf:a6:d4 brd ff:ff:ff:ff:ff:ff
    inet 10.8.37.18/16 brd 10.8.255.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet 10.8.78.31/16 scope global secondary eth0
       valid_lft forever preferred_lft forever
```

修改节点dns服务器

`/etc/resolv.conf`

```
search default.svc.cluster.local svc.cluster.local cluster.local
options timeout:1 attempts:1 ndots:5
nameserver 10.96.0.10
nameserver 10.8.255.1
nameserver 10.8.255.2
nameserver 114.114.114.114
```

开三个node节点的命令窗口,分别执行以下命令.

```
#验证vip漂移的网络影响
ping 10.8.78.31
#验证kube-apiserver故障影响
while true; do  sleep 1; curl -k https://10.8.78.31:6443; done
#验证dns解析影响
while true; do  sleep 1; nslookup kubernetes.default.svc.cluster.local; done
```

关闭master 10.8.37.18 机器

```
64 bytes from 10.8.78.31: icmp_seq=61 ttl=64 time=0.192 ms
From 10.8.104.16 icmp_seq=62 Time to live exceeded
64 bytes from 10.8.78.31: icmp_seq=64 ttl=64 time=0.164 ms
64 bytes from 10.8.78.31: icmp_seq=65 ttl=64 time=0.139 ms
```

```
Unauthorized
curl: (7) Failed connect to 10.8.78.31:6443; No route to host
curl: (7) Failed connect to 10.8.78.31:6443; No route to host
Unauthorized
Unauthorized
```

```
** server can't find kubernetes.default.svc.cluster.local: NXDOMAIN

Server:         10.8.255.1
Address:        10.8.255.1#53

** server can't find kubernetes.default.svc.cluster.local: NXDOMAIN

Server:         10.96.0.10
Address:        10.96.0.10#53
```

粗略估算,影响kube-apiserver为5秒,影响dns解析服务为10秒

```
[root@10-8-104-16 data]#kubectl get node
NAME            STATUS           AGE
10-8-104-16     Ready,master     h
10-8-113-73     Ready            h
10-8-125-29     Ready,master     h
10-8-37-18      NotReady,master  h
```

```
[root@10-8-104-16 data]# kubectl get all -n kube-system
NAME                                        READY    STATUS     RESTARTS   AGE
po/calico-policy-controller-fxjzw           1/1      Running    0          h
po/canal-node-jcz7                          3/3      Running    3          h
po/canal-node-gnk3                          3/3      Running    3          h
po/canal-node-s2br                          3/3      Running    0          h
po/canal-node-l1c9w                         3/3      NodeLost   6          h
po/dummy-2088944543-hmh5                    1/1      Running    0          h
po/kube-apiserver-10-8-104-16               1/1      Running    3          h
po/kube-apiserver-10-8-125-29               1/1      Running    2          h
po/kube-apiserver-10-8-37-18                1/1      Unknown    4          h
po/kube-controller-manager-10-8-104-16      1/1      Running    6          h
po/kube-controller-manager-10-8-125-29      1/1      Running    6          h
po/kube-controller-manager-10-8-37-18       1/1      Unknown    5          h
po/kube-discovery-w20c                      1/1      NodeLost   2          h
po/kube-discovery-wcrw                      1/1      Running    1          h
po/kube-discovery-tnfs4                      1/1      Running    1          h
po/kube-dns-pf48                            4/4      Running    4          h
po/kube-dns-cq4m5                           4/4      NodeLost   8          h
po/kube-dns-w8nq1                           4/4      Running    4          h
po/kube-proxy-bpt5                          1/1      Running    1          h
po/kube-proxy-blxhl                         1/1      Running    0          h
po/kube-proxy-dc9dz                         1/1      NodeLost   2          h
po/kube-proxy-z3q0n                         1/1      Running    1          h
po/kube-scheduler-10-8-104-16               1/1      Running    8          h
po/kube-scheduler-10-8-125-29               1/1      Running    7          h
po/kube-scheduler-10-8-37-18                1/1      Unknown    7          h

NAME           CLUSTER-IP     EXTERNAL-IP    PORT(S)          AGE
svc/kube-dns   10.96.0.10     <none>         53/UDP,53/TCP    h

NAME                   DESIRED    SUCCESSFUL    AGE
jobs/configure-canal   1          1             h

NAME                          DESIRED   CURRENT   READY    AGE
rs/calico-policy-controller   1         1         1        h
rs/dummy-2088944543           1         1         1        h
```

以下为参考配置

分类: 开源技术

标签: kubeadm, kubernetes, k8s, k8s高可用
« 上一篇：Vagrant 部署python开发环境
» 下一篇：kubernetes组件