



链滴

kubeadm 安装 k8s v1.13.1 HA 详细教程之二：keepalived+haproxy 安装

作者：[18582596683](#)

原文链接：<https://ld246.com/article/1547737157238>

来源网站：[链滴](#)

许可协议：[署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

该操作在所有master进行

1.部署keepalived

1.1 yum安装keepalived

```
yum install -y keepalived
```

1.2 配置keepalived

```
###第1个master
```

```
[root@k8s01 ~]# cat /etc/keepalived/keepalived.conf  
! Configuration File for keepalived
```

```
global_defs {  
    router_id LVS_DEVEL  
}
```

```
vrpp_script check_haproxy {  
    script "killall -0 haproxy"  
    interval 3  
    weight -2  
    fall 10  
    rise 2  
}
```

```
vrpp_instance VI_1 {  
    state MASTER  
    interface eth0  
    virtual_router_id 51  
    priority 100  
    advert_int 1  
    authentication {  
        auth_type PASS  
        auth_pass 1111  
    }  
    virtual_ipaddress {  
        192.168.158.138  
    }  
    track_script {  
        check_haproxy  
    }  
}
```

```
###第2个master
```

```
[root@k8s02 ~]# cat /etc/keepalived/keepalived.conf  
! Configuration File for keepalived
```

```
global_defs {
```

```
router_id LVS_DEVEL
}

vrrp_script check_haproxy {
    script "killall -0 haproxy"
    interval 3
    weight -2
    fall 10
    rise 2
}

vrrp_instance VI_1 {
    state BACKUP
    interface eth0
    virtual_router_id 51
    priority 99
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        192.168.158.138
    }
    track_script {
        check_haproxy
    }
}
```

###第3个master

```
[root@k8s03 ~]# cat /etc/keepalived/keepalived.conf
! Configuration File for keepalived
```

```
global_defs {
    router_id LVS_DEVEL
}

vrrp_script check_haproxy {
    script "killall -0 haproxy"
    interval 3
    weight -2
    fall 10
    rise 2
}

vrrp_instance VI_1 {
    state BACKUP
    interface eth0
    virtual_router_id 51
    priority 98
    advert_int 1
    authentication {
```

```

    auth_type PASS
    auth_pass 1111
}
virtual_ipaddress {
    192.168.158.138
}
track_script {
    check_haproxy
}
}

```

#####注意:

>1.killall -0 根据进程名称检测进程是否存活, 如果服务器没有该命令, 请使用yum install psmisc -安装

>2.第一个master节点的state为MASTER, 其他master节点的state为BACKUP

>3.priority表示各个节点的优先级, 范围: 0~250 (非强制要求)

1.3 启动并加入开机启动项

```
[root@k8s01 ~]# systemctl enable keepalived.service
```

```
[root@k8s01 ~]# systemctl start keepalived.service
```

```
[root@k8s01 ~]# systemctl status keepalived.service
```

- keepalived.service - LVS and VRRP High Availability Monitor

Loaded: loaded (/usr/lib/systemd/system/keepalived.service; enabled; vendor preset: disabled)

Active: active (running) since — 2019-01-14 21:20:51 CST; 7s ago

Process: 4692 ExecStart=/usr/sbin/keepalived \$KEEPALIVED_OPTIONS (code=exited, status=SUCCESS)

Main PID: 4693 (keepalived)

Tasks: 3

Memory: 2.6M

CGroup: /system.slice/keepalived.service

├─4693 /usr/sbin/keepalived -D

├─4694 /usr/sbin/keepalived -D

└─4695 /usr/sbin/keepalived -D

```
1月 14 21:20:55 k8s01 Keepalived_vrrp[4695]: VRRP_Instance(VI_1) Dropping received VRRP packet...
```

```
1月 14 21:20:56 k8s01 Keepalived_vrrp[4695]: (VI_1): ip address associated with VRID 51 not present in MASTER advert : 192.168.158.138
```

```
1月 14 21:20:56 k8s01 Keepalived_vrrp[4695]: bogus VRRP packet received on eth0 !!!
```

```
1月 14 21:20:56 k8s01 Keepalived_vrrp[4695]: VRRP_Instance(VI_1) Dropping received VRRP packet...
```

```
1月 14 21:20:57 k8s01 Keepalived_vrrp[4695]: Sending gratuitous ARP on eth0 for 192.168.158.138
```

```
1月 14 21:20:57 k8s01 Keepalived_vrrp[4695]: VRRP_Instance(VI_1) Sending/queueing gratuitous ARPs on eth0 for 192.168.158.138
```

```
1月 14 21:20:57 k8s01 Keepalived_vrrp[4695]: Sending gratuitous ARP on eth0 for 192.168.158.138
```

```
1月 14 21:20:57 k8s01 Keepalived_vrrp[4695]: Sending gratuitous ARP on eth0 for 192.168.158.138
```

```
1月 14 21:20:57 k8s01 Keepalived_vrrp[4695]: Sending gratuitous ARP on eth0 for 192.168.158.138
```

```
1月 14 21:20:57 k8s01 Keepalived_vrrp[4695]: Sending gratuitous ARP on eth0 for 192.168.158.138
```

138

```
[root@k8s01 ~]# ip addr show eth0
```

```
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group  
efault qlen 1000
```

```
link/ether 52:54:00:83:7d:49 brd ff:ff:ff:ff:ff:ff
```

```
inet 192.168.158.131/24 brd 192.168.158.255 scope global noprefixroute eth0
```

```
valid_lft forever preferred_lft forever
```

```
inet 192.168.158.138/32 scope global eth0
```

```
valid_lft forever preferred_lft forever
```

```
inet6 fe80::3d6b:3fb8:fc5a:163e/64 scope link tentative noprefixroute dadfailed
```

```
valid_lft forever preferred_lft forever
```

```
inet6 fe80::138d:843c:9ef4:edfe/64 scope link tentative noprefixroute dadfailed
```

```
valid_lft forever preferred_lft forever
```

```
inet6 fe80::1ee9:e6e8:75dc:9865/64 scope link tentative noprefixroute dadfailed
```

```
valid_lft forever preferred_lft forever
```

2.部署haproxy

2.1 yum安装haproxy

```
yum install -y haproxy
```

1.2 配置haproxy(所有master一样的配置)

```
[root@k8s01 ~]# cat /etc/haproxy/haproxy.cfg
```

```
#-----  
# Example configuration for a possible web application. See the  
# full configuration options online.
```

```
#
```

```
# http://haproxy.1wt.eu/download/1.4/doc/configuration.txt
```

```
#
```

```
#-----
```

```
#-----
```

```
# Global settings
```

```
#-----
```

```
global
```

```
# to have these messages end up in /var/log/haproxy.log you will
```

```
# need to:
```

```
#
```

```
# 1) configure syslog to accept network log events. This is done
```

```
# by adding the '-r' option to the SYSLOGD_OPTIONS in
```

```
# /etc/sysconfig/syslog
```

```
#
```

```
# 2) configure local2 events to go to the /var/log/haproxy.log
```

```
# file. A line like the following can be added to
```

```
# /etc/sysconfig/syslog
```

```
#
```

```
# local2.* /var/log/haproxy.log
```

```
#
```

```
log 127.0.0.1 local2
```

```

chroot    /var/lib/haproxy
pidfile   /var/run/haproxy.pid
maxconn   4000
user      haproxy
group     haproxy
daemon

# turn on stats unix socket
stats socket /var/lib/haproxy/stats

#-----
# common defaults that all the 'listen' and 'backend' sections will
# use if not designated in their block
#-----
defaults
  mode                http
  log                 global
  option              httplog
  option              dontlognull
  option http-server-close
  option forwardfor   except 127.0.0.0/8
  option              redispatch
  retries             3
  timeout http-request 10s
  timeout queue       1m
  timeout connect     10s
  timeout client      1m
  timeout server      1m
  timeout http-keep-alive 10s
  timeout check       10s
  maxconn             3000

#-----
# kubernetes apiserver frontend which proxys to the backends
#-----
frontend kubernetes-apiserver
  mode                tcp
  bind                *:16443
  option              tcplog
  default_backend     kubernetes-apiserver

#-----
# round robin balancing between the various backends
#-----
backend kubernetes-apiserver
  mode                tcp
  balance              roundrobin
  server k8s01 192.168.158.131:6443 check
  server k8s02 192.168.158.132:6443 check
  server k8s03 192.168.158.133:6443 check

#-----
# collection haproxy statistics message

```

```
#-----  
listen stats  
  bind          *:1080  
  stats auth    admin:awesomePassword  
  stats refresh 5s  
  stats realm   HAProxy\ Statistics  
  stats uri     /admin?stats
```

2.3 启动并加入开机启动项

```
[root@k8s01 ~]# systemctl enable haproxy.service  
[root@k8s01 ~]# systemctl start haproxy.service  
[root@k8s01 ~]# systemctl status haproxy.service  
● haproxy.service - HAProxy Load Balancer  
  Loaded: loaded (/usr/lib/systemd/system/haproxy.service; enabled; vendor preset: disabled)  
  Active: active (running) since — 2019-01-14 21:26:15 CST; 9s ago  
  Main PID: 4735 (haproxy-systemd)  
    Tasks: 3  
   Memory: 2.5M  
   CGroup: /system.slice/haproxy.service  
           └─4735 /usr/sbin/haproxy-systemd-wrapper -f /etc/haproxy/haproxy.cfg -p /run/hapr  
xy.pid  
           └─4736 /usr/sbin/haproxy -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -Ds  
           └─4737 /usr/sbin/haproxy -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -Ds
```

```
1月 14 21:26:15 k8s01 systemd[1]: Started HAProxy Load Balancer.  
1月 14 21:26:15 k8s01 haproxy-systemd-wrapper[4735]: haproxy-systemd-wrapper: executing  
/usr/sbin/haproxy -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -Ds  
1月 14 21:26:15 k8s01 haproxy-systemd-wrapper[4735]: [WARNING] 013/212615 (4736) : conf  
g : 'option forwardfor' ignored for frontend 'kubernetes-apiserver' as it requires HTTP mode.  
1月 14 21:26:15 k8s01 haproxy-systemd-wrapper[4735]: [WARNING] 013/212615 (4736) : conf  
g : 'option forwardfor' ignored for backend 'kubernetes-apiserver' as it requires HTTP mode.  
[root@k8s01 ~]# ss -lnt | grep -E "16443|1080"  
LISTEN 0      128          *:1080          *.*  
LISTEN 0      128          *:16443         *.*
```