# keepalived-2.0.4 安装配置手册

## 安装包准备

安装包下载

进入到http://www.keepalived.org/download.html 下载2.0.4版本

将下载好的安装包上传到

/opt/aspire/product/dicmp/keepalived目录

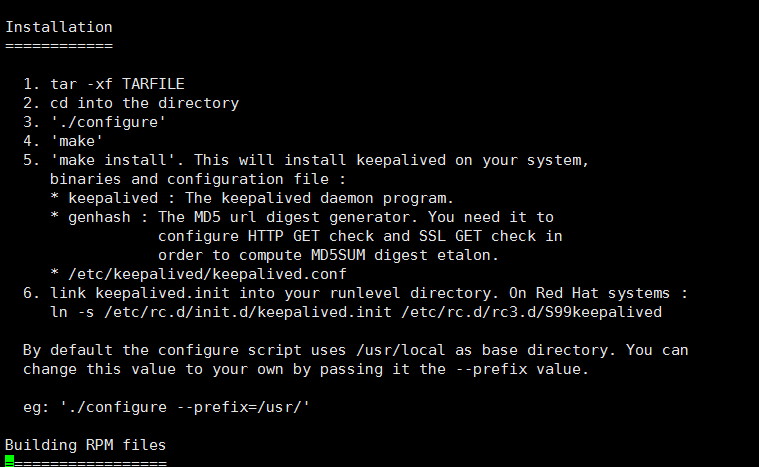
执行解压命令

tar -zxvf keepalived-2.0.4.tar.gz

进入到/opt/aspire/product/dicmp/keepalived/keepalived-2.0.4目录

## 安装

查看INSTALL文件，我们可以看到keepalived的安装方式如下：



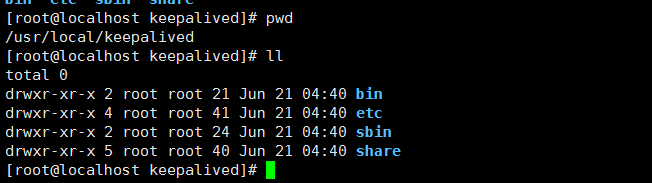
我们前面已经把文件解压并且进入到了keepalived的解压包目录中，所以我们只要依次执行3.4.5三个步骤即可:

./configure --prefix=/usr/local/keepalived

make && make install

在INSTALL文件中说明了需要初始化的依赖包，请按照要求提前安装。

安装完成后，我们进入到/urs/local/keepalived目录可以看到



## 服务制作

mkdir /etc/keepalived

cp /usr/local/keepalived/etc/keepalived/keepalived.conf /etc/keepalived/

cp /usr/local/keepalived/etc/sysconfig/keepalived /etc/sysconfig/

ln -s /usr/local/keepalived/sbin/keepalived /usr/sbin/

chkconfig keepalived on

## 配置

在前面我已经执行了

cp /usr/local/keepalived/etc/keepalived/keepalived.conf /etc/keepalived/

这里，我们把/etc/keepalived/keepalived.conf 做一个备份，再安装如下配置进行

**注意：**

1. **两个节点都要设置为BACKUP**
2. **interface ens33 这个需要安装机器实际的网卡来设置**
3. **router\_id node134 这个是各个节点的名称**
4. **script "/etc/keepalived/haproxy\_check.sh"需要创建检测脚本，后续有脚本**
5. **virtual\_router\_id 35 这个值两个机器都必须一直的**
6. **virtual\_ipaddress可以随便写一个没有被占用的ip即可**
7. **priority 值大的机器上，必须要配置nopreempt，开启抢占模式**

**Node1配置**

! Configuration File for keepalived

bal\_defs {

router\_id node134

}

vrrp\_script chk\_haproxy {

script "/etc/keepalived/haproxy\_check.sh"

interval 2

weight 2

}

vrrp\_instance VI\_1 {

state BACKUP

interface ens33

virtual\_router\_id 35

priority 98

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1111

}

track\_script {

chk\_haproxy

}

virtual\_ipaddress {

192.168.239.130

}

}

**Node2配置**

! Configuration File for keepalived

! Configuration File for keepalived

global\_defs {

router\_id node131

}

vrrp\_script chk\_haproxy {

script "/etc/keepalived/haproxy\_check.sh"

interval 2

weight 2

}

vrrp\_instance VI\_1 {

state BACKUP

interface ens33

virtual\_router\_id 35

priority 100

nopreempt

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1111

}

track\_script {

chk\_haproxy

}

virtual\_ipaddress {

192.168.239.130

}

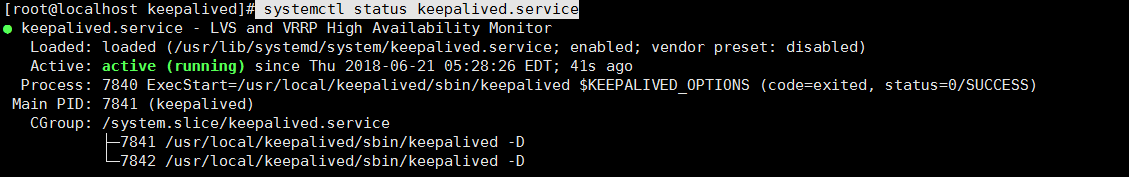
}

## 启动

systemctl start keepalived.service

## 查看状态

systemctl status keepalived.service



## 重启

systemctl restart keepalived.service

## 停止

systemctl stop keepalived.service

## haproxy\_check脚本

#!/bin/bash

START\_HAPROXY="/usr/local/haproxy/sbin/haproxy -D -f /usr/local/haproxy/haproxy.cfg"

LOG\_FILE="/usr/local/keepalived/log/haproxy-check.log"

HAPS=`ps -C haproxy --no-header |wc -l`

date "+%Y-%m-%d %H:%M:%S" >> $LOG\_FILE

echo "check haproxy status" >> $LOG\_FILE

if [ $HAPS -eq 0 ];then

echo "try to start haproxy" >> $LOG\_FILE

$START\_HAPROXY >> $LOG\_FILE 2>&1

sleep 3

if [ `ps -C haproxy --no-header |wc -l` -eq 0 ];then

echo " systemctl stop keepalived.service " >> $LOG\_FILE

systemctl stop keepalived.service

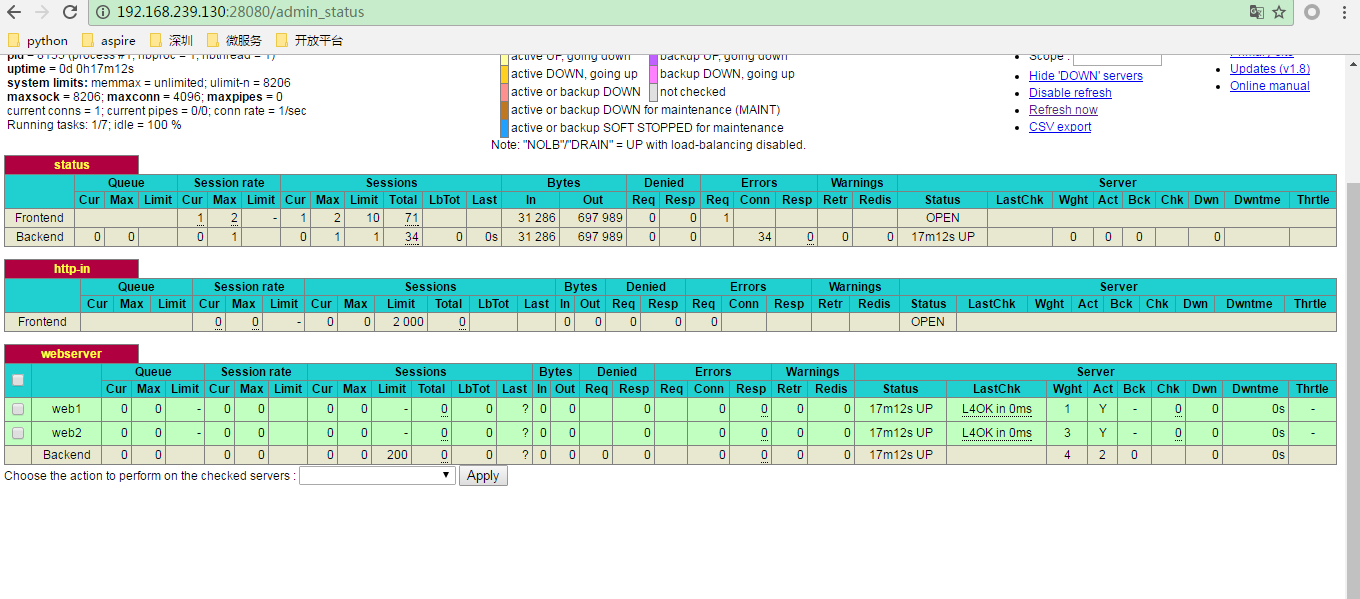
fi

fi

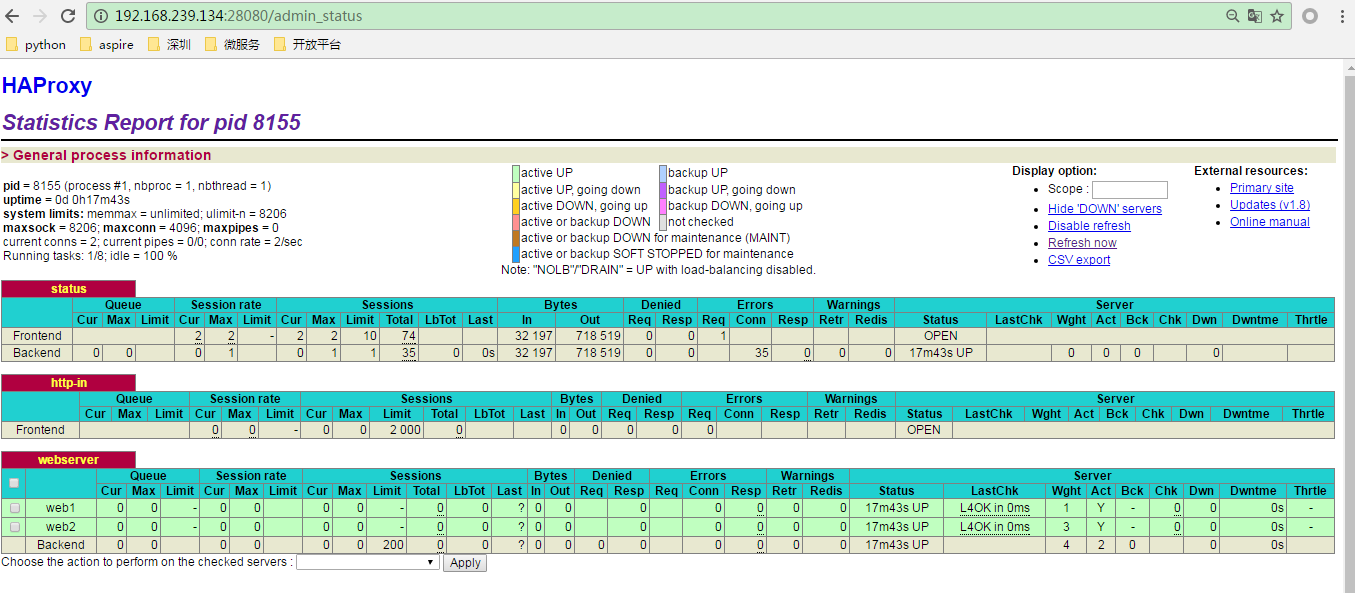
## 效果检查

启动机器上面keepalived后，我们可以得到如下的效果

1. kill掉haproxy后，会立刻重启
2. 我们可以通过192.168.239.130:haproxy\_port访问到haproxy
3. 我们关闭掉134上的keepalived，192.168.239.130地址会自动的被131机器抢占了



而实际上，我们的haproxy启动在134，131机器上面



我们可以通过status命令查看到每个机器上面的地址变化

