参考：<http://www.cnblogs.com/liwei0526vip/p/6391833.html>

# [使用HeartBeat实现高可用HA的配置过程详解](http://www.cnblogs.com/liwei0526vip/p/6391833.html)

**一、写在前面**

    HA即(high available)高可用，又被叫做双机热备，用于关键性业务。简单理解就是，有2台机器 A 和 B，正常是 A 提供服务，B 待命闲置，当 A 宕机或服务宕掉，会切换至B机器继续提供服务。常见的实现高可用的开源软件有 heartbeat 和 keepalived。

    这样，一台 web 服务器一天24小时提供web服务，难免会存在 web 服务挂掉或服务器宕机的情况，那么用户就访问不了服务了，这当然不是我们期望的。如果这样，有2台服务器，A对外提供 web 服务，B作为备用，如果A挂掉，那么B立刻替代A的位置去提供 web 服务，这样对用户来说是透明的。但是有个问题，服务器A的 ip 是 10.0.0.100，服务器B的 ip 是 10.0.0.101，显然向用户提供A或B的ip地址是不可行的，因为用户总不能去切换ip来访问的吧。这时heartbeat或keepalived可以提供一个虚拟IP：10.0.0.102，用户只需要访问 10.0.0.102，当A提供服务时，VIP（虚拟IP） 会设置在A服务器上，当B提供服务时，VIP会设置在B服务器上，这样就可以让用户通过访问 10.0.0.102 来获取web服务，即使A或B服务器切换也不影响用户的正常访问。

下面我们使用 heartbeat 来做 HA 集群，并且把 nginx 服务作为 HA 对应的服务。

**二、准备实验环境**

**服务器A：**  
主机名：master  
操作系统：CentOS6.8 64位  
eth0网卡地址：192.168.0.18  
eth1网卡地址：172.16.254.18

**服务器B：**  
主机名：slave  
操作系统：CentOS6.8 64位  
eth0网卡地址：192.168.0.28  
eth1网卡地址：172.16.254.28

**虚拟VIP：**  
VIP：192.168.0.38

**三、设置主机名**

**master节点设置hostname**

hostname master

vim /etc/sysconfig/network

编辑配置文件:

HOSTNAME=master

**slave节点设置hostname**

# hostname slave

# vim /etc/sysconfig/network

编辑配置文件:

HOSTNAME=slave

**四、关闭防火墙和selinux(2台节点都要操作)**

**关闭iptables**

# iptables -F

# service iptables save

# service iptables stop

**关闭selinux：**

# setenforce 0

# sed -i 's/SELINUX=enforcing/SELINUX=disabled/' /etc/selinux/config

**五、配置hosts文件(2台节点都操作)**

# vim /etc/hosts

增加内容如下：

192.168.0.18 master

192.168.0.28 slave

**六、安装epel扩展源 (2台都操作)**

# yum install -y epel-release

**七、安装heartbeat (2台都操作)**

# yum install -y heartbeat\* libnet nginx

**八、主master节点配置**

**1、拷贝配置文件：**

# cd /usr/share/doc/heartbeat-3.0.4/

# cp authkeys ha.cf haresources /etc/ha.d/

# cd /etc/ha.d

**2、修改authkeys**

# vim authkeys

更改或增加如下内容：

auth 3

3 md5 Hello!

然后修改其权限

# chmod 600 authkeys

**3、编辑haresources文件**

# vim haresources

加入下面一行：

master 192.168.0.38/24/eth0:0 nginx

说明：master为主节点hostname，192**.**168**.**0**.**38为vip，**/**24为掩码为24的网段，eth0**:**0为vip的设备名，nginx为heartbeat监控的服务，也是两台机器对外提供的核心服务。

**4、编辑ha.cf**

[复制代码](javascript:void(0);)

# vim ha.cf

修改为如下内容：

debugfile /var/log/ha-debug

logfile /var/log/ha-log

logfacility local0

keepalive 2

deadtime 30

warntime 10

initdead 60

udpport 694

ucast eth1 172.16.254.28

auto\_failback on

node master

node slave

ping 172.16.254.1

respawn hacluster /usr/lib64/heartbeat/ipfail

[复制代码](javascript:void(0);)

**5、配置说明：**

**debugfile /var/log/ha-debug：**该文件保存heartbeat的调试信息。  
**logfile /var/log/ha-log：**heartbeat的日志文件。  
**keepalive 2**：心跳的时间间隔，默认时间单位为秒s。  
**deadtime 30**：超出该时间间隔未收到对方节点的心跳，则认为对方已经死亡。  
**warntime 10**：超出该时间间隔未收到对方节点的心跳，则发出警告并记录到日志中。  
**initdead 60**：在某系统上，系统启动或重启之后需要经过一段时间网络才能正常工作，该选项用于解决这种情况产生的时间间隔，取值至少为deadtime的2倍。  
**udpport 694**：设置广播通信使用的端口，694为默认使用的端口号。  
**ucast eth1 172.16.254.28**：设置对方机器心跳检测的网卡和IP。  
**auto\_failback on**：heartbeat的两台主机分别为主节点和从节点。主节点在正常情况下占用资源并运行所有的服务，遇到故障时把资源交给从节点由从节点运行服务。在该选项设为on的情况下，一旦主节点恢复运行，则自动获取资源并取代从节点，否则不取代从节点。  
**respawn heartbeat /usr/lib/heartbeat/ipfail**：指定与heartbeat一同启动和关闭的进程，该进程被自动监视，遇到故障则重新启动。最常用的进程是ipfail，该进程用于检测和处理网络故障，需要配合ping语句指定的ping node来检测网络连接。如果你的系统是64bit，请注意该文件的路径。

**九、把主节点上的三个配置文件拷贝到从节点**

# cd /etc/ha.d

# scp authkeys ha.cf haresources slave:/etc/ha.d

**十、从节点slave编辑ha.cf**

# vim /etc/ha.d/ha.cf

只需要更改一个地方如下:

ucast eth1 172.16.254.28改为ucast eth1 172.16.254.18

**十一、启动heartbeat服务**

配置完毕后，先master启动，后slave启动。

# service heartbeat start

**十二、检查测试**

# ifconfig

看是否有接口 eth0:0

# ps aux | grep nginx

看是否有nginx进程

**十三、测试方式1**

主节点上故意禁ping

# iptables -I INPUT -p icmp -j DROP

**十四、测试方式2**

主节点停止heartbeat服务

# service heartbeat stop

**十五、测试脑裂**

主节点master和从节点slave都down掉eth1网卡

# ifdown eth1