**KeepAlive安装及配置手册**

# 一、简介

        负载平衡是一种在真实服务器集群中分配IP流量的方法，可提供一个或多个高度可用的虚拟服务。在设计负载均衡拓扑时，重要的是要考虑负载均衡器本身的可用性以及它背后的真实服务器。用C编写的类似于layer3, 4 & 7交换机制软件，具备我们平时说的第3层、第4层和第7层交换机的功能。

[Keepalived](https://www.baidu.com/s?wd=Keepalived&tn=24004469_oem_dg&rsv_dl=gh_pl_sl_csd" \t "_blank)为负载平衡和高可用性提供了框架。负载平衡框架依赖于[众所周知](https://www.baidu.com/s?wd=%E4%BC%97%E6%89%80%E5%91%A8%E7%9F%A5&tn=24004469_oem_dg&rsv_dl=gh_pl_sl_csd)且广泛使用的Linux虚拟服务器（IPVS）内核模块，该模块提供第4层负载平衡。Keepalived实现了一组健康检查器，以根据其健康动态地自适应地维护和管理负载平衡的服务器池。虚拟冗余路由协议（VRRP）实现高可用性。VRRP是路由器故障转移的基础砖块。此外，keepalived实现了一组挂钩到VRRP有限状态机，提供低级别和高速协议交互。每个Keepalived框架可以独立使用或一起使用，以提供弹性基础设施。

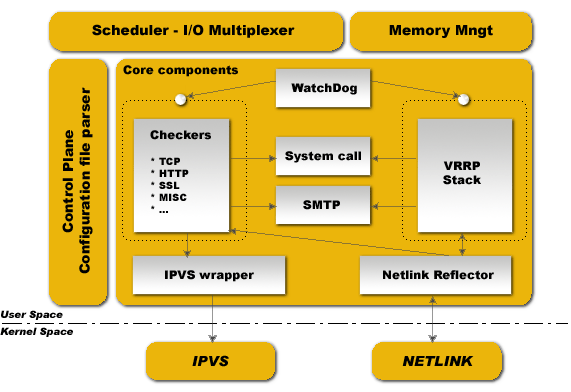
在这种情况下，负载平衡器也可以被称为导向器或LVS路由器。

简而言之，Keepalived提供了两个主要功能：

* 健康检查LVS系统
* 实施VRRPv2堆栈以处理负载均衡器故障转移

**Keepalived使用四个**[**Linux内核**](https://www.baidu.com/s?wd=Linux%E5%86%85%E6%A0%B8&tn=24004469_oem_dg&rsv_dl=gh_pl_sl_csd)**组件：**

1. LVS框架：使用getsockopt和setsockopt调用来获取和设置套接字上的选项。
2. Netfilter框架：支持NAT和伪装的IPVS代码。
3. Netlink接口：设置和删除网络接口上的VRRP虚拟IP。
4. 组播：将VRRP通告发送到保留的VRRP MULTICAST组（224.0.0.18）。

  
上图是Keepalived的功能体系结构，大致分两层：用户空间（user space）和内核空间（kernel space）。  
**内核空间**：主要包括IPVS（IP虚拟服务器，用于实现网络服务的负载均衡）和NETLINK（提供高级路由及其他相关的网络功能）两个部份。  
**用户空间**：

* WatchDog：负载监控checkers和VRRP进程的状况
* VRRP Stack：负载负载均衡器之间的失败切换FailOver，如果只用一个负载均稀器，则VRRP不是必须的。
* Checkers：负责真实服务器的健康检查healthchecking，是keepalived最主要的功能。换言之，可以没有VRRP Stack，但健康检查healthchecking是一定要有的。
* IPVS wrapper：用户发送设定的规则到内核ipvs代码
* Netlink Reflector：用来设定vrrp的vip地址等。

Keepalived的所有功能是配置keepalived.conf文件来实现的。

# 二、安装keepalived

## 2.1 方法一：yum方式

1、yum安装

安装依赖包

[root@localhost ~]# yum install -y curl gcc openssl-devel libnl3-devel net-snmp-devel

[root@localhost ~]# yum install -y keepalived

2、初始化及启动

[root@localhost ~]# systemctl start keepalived   //启动keepalived

[root@localhost ~]# systemctl enable keepalived  //加入开机启动keepalived

[root@localhost ~]# systemctl restart keepalived  //重新启动keepalived

[root@localhost ~]# systemctl status keepalived   //查看keepalived状态

## 2.2方法二、源码包

1、官网[下载keepalived](http://www.keepalived.org/download.html" \t "_blank)的最新版本，解压并安装

[root@localhost ~]# yum install -y curl gcc openssl-devel libnl3-devel net-snmp-devel（也要安装依赖包）

[root@master src]# pwd

/usr/local/src

[root@master src]# wget http://www.keepalived.org/software/keepalived-2.0.7.tar.gz

[root@master src]# tar xvf keepalived-2.0.7.tar.gz

[root@master src]# cd keepalived-2.0.7

[root@master keepalived-2.0.7]# ./configure --prefix=/usr/local/keepalived

[root@master keepalived-2.0.7]# make && make install

完成后会在以下路径生成：

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

/usr/local/etc/sysconfig/keepalived

/usr/local/sbin/keepalived

2、初始化及启动

# keepalived启动脚本变量引用文件，默认文件路径是/etc/sysconfig/，也可以不做软链接，直接修改启动脚本中文件路径即可（安装目录下）

[root@localhost /]# cp /usr/local/keepalived/etc/sysconfig/keepalived /etc/sysconfig/keepalived

# 将keepalived主程序加入到环境变量（安装目录下）

[root@localhost /]# cp /usr/local/keepalived/sbin/keepalived /usr/sbin/keepalived

# keepalived启动脚本（源码目录下），放到/etc/init.d/目录下就可以使用service命令便捷调用

[root@localhost /]# cp /usr/local/src/keepalived-2.0.7/keepalived/etc/init.d/keepalived /etc/init.d/keepalived

# 将配置文件放到默认路径下

[root@localhost /]# mkdir /etc/keepalived

[root@localhost /]# cp /usr/local/keepalived/etc/keepalived/keepalived.conf /etc/keepalived/keepalived.conf

* 加为系统服务：chkconfig –add keepalived
* 开机启动：chkconfig keepalived on
* 查看开机启动的服务：chkconfig –list
* 启动、关闭、重启service keepalived start|stop|restart

# 三、keepalived.conf配置文件说明

keepalived服务安装完成之后，后面的主要工作就是在keepalived.conf文件中配置HA和负载均衡。一个功能比较完整的常用的keepalived配置文件，主要包含三块：全局定义块、VRRP实例定义块和虚拟服务器定义块。全局定义块是必须的，如果keepalived只用来做ha，虚拟服务器是可选的。下面是一个功能比较完整的配置文件模板：

#全局定义块

global\_defs {

# 邮件通知配置

notification\_email {

email1

email2

}

notification\_email\_from email

smtp\_server host

smtp\_connect\_timeout num

lvs\_id string

router\_id string ## 标识本节点的字条串,通常为hostname

}

#VRRP 实例定义块

vrrp\_sync\_group string {

group {

string

string

}

}

vrrp\_instance string {

state MASTER|BACKUP

virtual\_router\_id num

interface string

mcast\_src\_ip @IP

priority num

advert\_int num

nopreempt

smtp\_alert

lvs\_sync\_daemon\_interface string

authentication {

auth\_type PASS|AH

auth\_pass string

}

virtual\_ipaddress { # Block limited to 20 IP addresses @IP

@IP

@IP

}

}

#虚拟服务器定义块

virtual\_server (@IP PORT)|(fwmark num) {

delay\_loop num

lb\_algo rr|wrr|lc|wlc|sh|dh|lblc

lb\_kind NAT|DR|TUN

persistence\_timeout num

protocol TCP|UDP

real\_server @IP PORT {

weight num

notify\_down /path/script.sh

TCP\_CHECK {

connect\_port num

connect\_timeout num

}

}

real\_server @IP PORT {

weight num

MISC\_CHECK {

misc\_path /path\_to\_script/script.sh(or misc\_path “/path\_to\_script/script.sh <arg\_list>”)

}

}

real\_server @IP PORT {

weight num

HTTP\_GET|SSL\_GET {

url {

# You can add multiple url block path alphanum

digest alphanum

}

connect\_port num

connect\_timeout num

nb\_get\_retry num

delay\_before\_retry num

}

}

}

### 全局定义块

1、email通知（notification\_email、smtp\_server、smtp\_connect\_timeout）：用于服务有故障时发送邮件报警，可选项，不建议用。需要系统开启sendmail服务，建议用第三独立监控服务，如用[nagios](http://baike.baidu.com/link?url=eiaO-UZufjBG-j-e-nMS2RZrjxE_Xd2PpecyOrAq3sv0umvlhfpMlkR7pO-wEnpV4Vb2e7DWnZ9kKDfeh9YiDa)全面监控代替。  
2、lvs\_id：lvs负载均衡器标识，在一个网络内，它的值应该是唯一的。  
3、router\_id：用户标识本节点的名称，通常为hostname  
4、花括号｛｝：用来分隔定义块，必须成对出现。如果写漏了，keepalived运行时不会得到预期的结果。由于定义块存在嵌套关系，因此很容易遗漏结尾处的花括号，这点需要特别注意。

### VRRP实例定义块

1. vrrp\_sync\_group：同步vrrp级，用于确定失败切换（FailOver）包含的路由实例个数。即在有2个负载均衡器的场景，一旦某个负载均衡器失效，需要自动切换到另外一个负载均衡器的实例是哪
2. group：至少要包含一个vrrp实例，vrrp实例名称必须和vrrp\_instance定义的一致
3. vrrp\_instance：vrrp实例名  
   **1> state**：实例状态，只有MASTER 和 BACKUP两种状态，并且需要全部大写。抢占模式下，其中MASTER为工作状态，BACKUP为备用状态。当MASTER所在的服务器失效时，BACKUP所在的服务会自动把它的状态由BACKUP切换到MASTER状态。当失效的MASTER所在的服务恢复时，BACKUP从MASTER恢复到BACKUP状态。  
   **2> interface**：对外提供服务的网卡接口，即VIP绑定的网卡接口。如：eth0，eth1。当前主流的服务器都有2个或2个以上的接口（分别对应外网和内网），在选择网卡接口时，一定要核实清楚。  
   **3> mcast\_src\_ip**：本机IP地址  
   **4> virtual\_router\_id**：虚拟路由的ID号，每个节点设置必须一样，可选择IP最后一段使用，相同的 VRID 为一个组，他将决定多播的 MAC 地址。  
   **5> priority**：节点优先级，取值范围0～254，MASTER要比BACKUP高  
   **6> advert\_int**：MASTER与BACKUP节点间同步检查的时间间隔，单位为秒  
   **7> lvs\_sync\_daemon\_inteface**：负载均衡器之间的监控接口,类似于 HA HeartBeat 的心跳线。但它的机制优于 Heartbeat，因为它没有“裂脑”这个问题,它是以优先级这个机制来规避这个麻烦的。在 DR 模式中，lvs\_sync\_daemon\_inteface与服务接口interface使用同一个网络接口  
   **8> authentication**：验证类型和验证密码。类型主要有 PASS、AH 两种，通常使用PASS类型，据说AH使用时有问题。验证密码为明文，同一vrrp 实例MASTER与BACKUP使用相同的密码才能正常通信。  
   **9> smtp\_alert**：有故障时是否激活邮件通知  
   **10> nopreempt**：禁止抢占服务。默认情况，当MASTER服务挂掉之后，BACKUP自动升级为MASTER并接替它的任务，当MASTER服务恢复后，升级为MASTER的BACKUP服务又自动降为BACKUP，把工作权交给原MASTER。当配置了nopreempt，MASTER从挂掉到恢复，不再将服务抢占过来。  
   **11> virtual\_ipaddress**：虚拟IP地址池，可以有多个IP，每个IP占一行，不需要指定子网掩码。注意：这个IP必须与我们的设定的vip保持一致。

### 虚拟服务器virtual\_server定义块

1. virtual\_server：定义一个虚拟服务器，这个ip是virtual\_ipaddress中定义的其中一个，后面一个空格，然后加上虚拟服务的端口号。  
   1> delay\_loop：健康检查时间间隔，单位：秒  
   2> lb\_algo：负载均衡调度算法，互联网应用常用方式为wlc或rr  
   3> lb\_kind：负载均衡转发规则。包括DR、NAT、TUN 3种，一般使用路由（DR）转发规则。  
   4> persistence\_timeout：http服务会话保持时间，单位：秒  
   5> protocol：转发协议，分为TCP和UDP两种
2. real\_server：真实服务器IP和端口，可以定义多个  
   1> weight：负载权重，值越大，转发的优先级越高  
   2> notify\_down：服务停止后执行的脚本  
   3> TCP\_CHECK：服务有效性检测  
   \* connect\_port：服务连接端口  
   \* connect\_timeout：服务连接超时时长，单位：秒  
   \* nb\_get\_retry：服务连接失败重试次数  
   \* delay\_before\_retry：重试连接间隔，单位：秒

# 四 keepalive配置文件详解

## 模板实例

第一部分：全局定义块

　　1、email通知。作用：有故障，发邮件报警。

　　2、Lvs负载均衡器标识（lvs\_id）。在一个网络内，它应该是唯一的。

　　3、花括号“{}”。用来分隔定义块，因此必须成对出现。如果写漏了，keepalived运行时，不会得到预期的结果。由于定义块内存在嵌套关系，因此很容易遗漏结尾处的花括号，这点要特别注意。

global\_defs {

notification\_email { #指定keepalived在发生切换时需要发送email到的对象，一行一个

sysadmin@fire.loc

}

notification\_email\_from Alexandre.Cassen@firewall.loc #指定发件人

smtp\_server localhost #指定smtp服务器地址

smtp\_connect\_timeout 30 #指定smtp连接超时时间

router\_id LVS\_DEVEL #运行keepalived机器的一个标识

}

第二部分：vrrp\_sync\_group作用：确定失败切换（FailOver）包含的路由实例个数。即在有2个负载均衡器的场景，一旦某个负载均衡器失效，需要自动切换到另外一个负载均衡器的实例是哪些？ 实例组group{}至少包含一个vrrp实例

vrrp\_sync\_group VG\_1{ #监控多个网段的实例

group {

　　　　VI\_1 #实例名

　　　　VI\_2

　　　　......

}

notify\_master /path/xx.sh #指定当切换到master时，执行的脚本

netify\_backup /path/xx.sh #指定当切换到backup时，执行的脚本

notify\_fault "path/xx.sh VG\_1" #故障时执行的脚本

notify /path/xx.sh

smtp\_alert #使用global\_defs中提供的邮件地址和smtp服务器发送邮件通知

}

第三部分：vrrp\_instance，实例名出自实例组group所包含的那些名字。

vrrp\_instance VI\_1 {

state BACKUP #指定那个为master，那个为backup，如果设置了nopreempt这个值不起作用，主备考priority决

定

interface eth0 #设置实例绑定的网卡

dont\_track\_primary #忽略vrrp的interface错误（默认不设置）

track\_interface{ #设置额外的监控，里面那个网卡出现问题都会切换

eth0

eth1

}

mcast\_src\_ip #发送多播包的地址，如果不设置默认使用绑定网卡的primary ip

garp\_master\_delay #在切换到master状态后，延迟进行gratuitous ARP请求

virtual\_router\_id 50 #VPID标记

priority 99 #优先级，高优先级竞选为master

advert\_int 1 #检查间隔，默认1秒

nopreempt #设置为不抢占 注：这个配置只能设置在backup主机上，而且这个主机优先级要比另外一台高

preempt\_delay #抢占延时，默认5分钟

debug #debug级别

authentication { #设置认证

auth\_type PASS #认证方式

auth\_pass 111111 #认证密码

}

virtual\_ipaddress { #设置vip

192.168.202.200

}

}

第四部分：

虚拟服务器virtual\_server定义块 ，虚拟服务器定义是keepalived框架最重要的项目了，是keepalived.conf必不可少的部分。 该部分是用来管理LVS的，是实现keepalive和LVS相结合的模块。ipvsadm命令可以实现的管理在这里都可以通过参数配置实现，注意：real\_server是被包含在viyual\_server模块中的，是子模块。

virtual\_server 192.168.202.200 23 { //VIP地址，要和vrrp\_instance模块中的virtual\_ipaddress地址一致

　　　　delay\_loop 6 #健康检查时间间隔

　　　　lb\_algo rr #lvs调度算法rr|wrr|lc|wlc|lblc|sh|dh

　　　　lb\_kind DR #负载均衡转发规则NAT|DR|RUN

　　　　persistence\_timeout 5 #会话保持时间

　　　　protocol TCP #使用的协议

　　　　persistence\_granularity <NETMASK> #lvs会话保持粒度

　　　　virtualhost <string> #检查的web服务器的虚拟主机（host：头）

　　　　sorry\_server<IPADDR> <port> #备用机，所有realserver失效后启用

real\_server 192.168.200.5 23 { //RS的真实IP地址

weight 1 #默认为1,0为失效

inhibit\_on\_failure #在服务器健康检查失效时，将其设为0，而不是直接从ipvs中删除

notify\_up <string> | <quoted-string> #在检测到server up后执行脚本

notify\_down <string> | <quoted-string> #在检测到server down后执行脚本

TCP\_CHECK { //常用

connect\_timeout 3 #连接超时时间

nb\_get\_retry 3 #重连次数

delay\_before\_retry 3 #重连间隔时间

connect\_port 23 健康检查的端口的端口

bindto <ip>

}

HTTP\_GET | SSL\_GET{ //不常用

url{ #检查url，可以指定多个

path /

digest <string> #检查后的摘要信息

status\_code 200 #检查的返回状态码

}

connect\_port <port>

bindto <IPADD>

connect\_timeout 5

nb\_get\_retry 3

delay\_before\_retry 2

}

SMTP\_CHECK{ //不常用

host{

connect\_ip <IP ADDRESS>

connect\_port <port> #默认检查25端口

bindto <IP ADDRESS>

}

connect\_timeout 5

retry 3

delay\_before\_retry 2

helo\_name <string> | <quoted-string> #smtp helo请求命令参数，可选

}

MISC\_CHECK{ //不常用

misc\_path <string> | <quoted-string> #外部脚本路径

misc\_timeout #脚本执行超时时间

misc\_dynamic #如设置该项，则退出状态码会用来动态调整服务器的权重，返回0 正常，不修改；返回1，

检查失败，权重改为0；返回2-255，正常，权重设置为：返回状态码-2

}

}

## **生产环境配置文件实例：**

[root@LB2 ~]# cat /etc/keepalived/keepalived.conf

! Configuration File for keepalived

global\_defs {

notification\_email {

49000448@qq.com

}

notification\_email\_from Alexandre.Cassen@firewall.loc

smtp\_server 10.0.0.1

smtp\_connect\_timeout 30

router\_id LVS\_2

}

vrrp\_instance VI\_1 {

state BACKUP

interface eth0

virtual\_router\_id 55

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1111

}

virtual\_ipaddress {

192.168.220.110/24

}

virtual\_server 192.168.220.110 80 {

delay\_loop 6

lb\_algo wrr

lb\_kind DR

nat\_mask 255.255.255.0

persistence\_timeout 300

protocol TCP

}

real\_server 192.168.220.129 80 {

weight 1

TCP\_CHECK {

connect\_timeout 8

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

real\_server 192.168.220.138 80 {

weight 1

TCP\_CHECK {

connect\_timeout 8

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

}

[root@LB2 ~]#

## **为keepalive指定日志文件**

默认keepalive的日志文件是

/var/log/messages

[root@LB2 ~]# tail -5 /var/log/messages

Dec 7 22:40:10 LB2 Keepalived\_vrrp: VRRP\_Instance(VI\_1) Entering BACKUP STATE

Dec 7 22:40:10 LB2 Keepalived\_vrrp: VRRP sockpool: [ifindex(2), proto(112), fd(10,11)]

Dec 7 22:40:10 LB2 Keepalived\_healthcheckers: Opening file '/etc/keepalived/keepalived.conf'.

Dec 7 22:40:10 LB2 Keepalived\_healthcheckers: Configuration is using : 8425 Bytes

Dec 7 22:40:10 LB2 Keepalived\_healthcheckers: Using LinkWatch kernel netlink reflector...

[root@LB2 ~]#

## **修改配置**

[root@LB2 ~]# vim /etc/sysconfig/keepalived

　　......

　　#KEEPALIVED\_OPTIONS="-D" //注释掉该行

　　KEEPALIVED\_OPTIONS="-D -d -S 0" //添加改行

　　......

[root@LB2 ~]# vim /etc/rsyslog.conf

　　......

　　local0.\* /var/log/keepalived.log //添加改行

[root@LB2 ~]# /etc/init.d/rsyslog restart //重启日志服务

Shutting down system logger: [ OK ]

Starting system logger: [ OK ]

[root@LB2 ~]# /etc/init.d/keepalived restart //重启keepalive服务

Stopping keepalived: [ OK ]

Starting keepalived: [ OK ]

[root@LB2 ~]# tail -5 /var/log/keepalived.log

Dec 8 01:02:06 LB2 Keepalived\_vrrp: Virtual IP = 1

Dec 8 01:02:06 LB2 Keepalived\_vrrp: 192.168.220.110/24 brd 192.168.220.110 dev eth0 scope global

Dec 8 01:02:06 LB2 Keepalived\_vrrp: Using LinkWatch kernel netlink reflector...

Dec 8 01:02:06 LB2 Keepalived\_vrrp: VRRP\_Instance(VI\_1) Entering BACKUP STATE

Dec 8 01:02:06 LB2 Keepalived\_vrrp: VRRP sockpool: [ifindex(2), proto(112), fd(10,11)]

[root@LB2 ~]#