



## Keepalived 配置文件解释

Keepalived的所有配置都在一个配置文件里面，主要分为三类：

- 全局配置
- VRRPD配置
- LVS 配置

配置文件是以配置块的形式存在，每个配置块都在一个闭合的{}范围内，所以编辑的时候需要注意大括号的闭合问题。**#和！ 开头都是注释。**

### 全局配置

全局配置是对整个 Keepalived 生效的配置，一个典型的配置如下：

```
global_defs {
    notification_email {          #设置 keepalived 在发生事件（比如切换）的时候，需要发送到的email地址，可以设置多个，每行一个。
        acassen@firewall.loc
        failover@firewall.loc
        sysadmin@firewall.loc
    }
    notification_email_from Alexandre.Cassen@firewall.loc
    #设置通知邮件发送来自于哪里，如果本地开启了sendmail的话，可以使用上面的默认值。
    smtp_server 192.168.200.1     #指定发送邮件的smtp服务器。
    smtp_connect_timeout 30      #设置smtp连接超时时间，单位为秒。
    router_id LVS_DEVEL         #是运行keepalived的一个表示，多个集群设置不同。
}
```

---

### VRRPD配置



VRRPD 的配置是 Keepalived 比较重要的配置，主要分为两个部分 VRRP 同步组和 VRRP实例，也就是想要使用 VRRP 进行高可用选举，那么就一定需要配置一个VRRP实例，在实例中来定义 VIP、服务器角色等。

## VRRP Sync Groups

不使用Sync Group的话，如果机器（或者说router）有两个网段，一个内网一个外网，每个网段开启一个VRRP实例，假设VRRP配置为检查内网，那么当外网出现问题时，VRRPD认为自己仍然健康，那么不会发生Master和Backup的切换，从而导致了问题。Sync group就是为了解决这个问题，可以把两个实例都放进一个Sync Group，这样的话，group里面任何一个实例出现问题都会发生切换。

```
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
}
```

## VRRP实例 (instance) 配置

VRRP实例就表示在上面开启了VRRP协议，这个实例说明了VRRP的一些特征，比如主从，VRID等，可以在每个interface上开启一个实例。

```
vrrp_instance VI_1 {
    state MASTER      #指定实例初始状态，实际的MASTER和BACKUP是选举决定的。
    interface eth0     #指定实例绑定的网卡
    virtual_router_id 51 #设置VRID标记，多个集群不能重复(0..255)
    priority 100       #设置优先级，优先级高的会被竞选为Master，Master要高于BACKUP至少50
```



```
advert_int 1          #检查的时间间隔，默认1s
nopreempt             #设置为不抢占，说明：这个配置只能在BACKUP主机上面设置
preempt_delay         #抢占延迟，默认5分钟
debug                #debug级别
authentication {      #设置认证
    auth_type PASS     #认证方式，支持PASS和AH，官方建议使用PASS
    auth_pass 1111     #认证的密码
}
virtual_ipaddress {
    #设置VIP，可以设置多个，用于切换时的地址绑定。
    格式：#<IPADDR>/<MASK> brd <IPADDR> dev <STRING> scope <SCOPE> label <LABEL>
    192.168.200.16/24 dev eth0 label eth0:1
    192.168.200.17/24 dev eth1 label eth1:1
    192.168.200.18
}
}
```

## LVS 配置

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keepalived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virtual\_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|lbc|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

```
}
}
```

---