

keepalived高可用部署

原创

GeorgeKai

2018-02-12 23:13:15

评论(0)

1016人阅读

keepalived介绍

Keepalived软件期初是专为LVS负载均衡软件设计的，用来管理并监控LVS几区系统汇总各个服务节点的状态，后来加入了可以实现高可用的VRRP功能。因此Keepalived除了能管理LVS软件外，还可以作为其他服务（例如：Nginx、Haproxy、Mysql等）的高可用解决方案软件。

Keepalived软件主要是通过VRRP协议实现高可用功能的。VRRP是Virtual Router Redundancy Protocol(虚拟路由冗余协议)的缩写，VRRP协议出现的目的就是为了解决静态路由单点故障问题的。它能够保证当个别节点宕机时，整个网络可以不间断的运行。所以，Keepalived一方面具有配置管理LVS的功能，同时还具有对LVS下面节点进行健康检查的功能，另一方面也可实现系统网络服务的高可用功能。

Keepalived软件的官方站点为<http://www.keepalived.org>

1.1 通过keepalived软件可实现的功能

- 1) 管理LVS负载均衡软件（keep和lvs配合）
- 2) 实现对LVS集群节点健康检查（lvs没有健康检查，需要keep）
- 3) 作为系统网络服务的高可用功能
- 4) 注意：通过man keepalived.conf ——获取keepalived软件功能说明信息

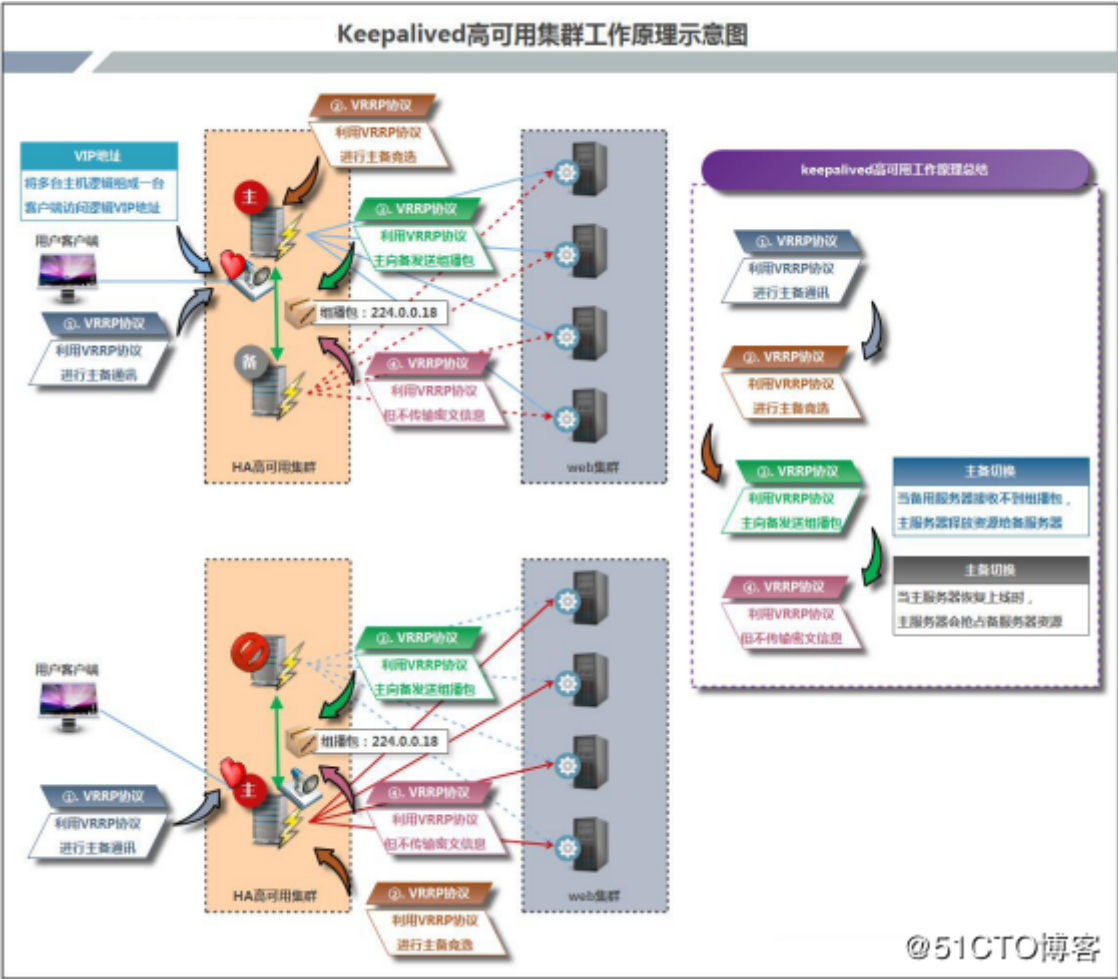
1.2 keepalived软件工作原理（如何实现一台负载坏了，keepalived自动切换）

vip地址：将多台主机逻辑组成一台，客户端访问逻辑vip地址

VRRP协议，全称是（Virtual router redundancy protocol）中文名为虚拟路由冗余协议，VRRP的出现是为了解决静态路由单点故障

VRRP是用ip组播的方式（默认组播地址224.0.0.18）实现高可用对之间通信的

工作时主节点发包，备节点接包，当备节点接收不到主节点发的数据包的时候，就启动接管程序接管主节点的资源。备节点可以有多个，通过优先级竞选，但一般keepalived系统运维工作中都是一对的。



@51CTO博客

keepalived软件配置过程

1.3 环境准备说明

主机名	ip地址
lb01	10.0.0.5
lb02	10.0.0.6
web01	10.0.0.82
web02	10.0.0.83
web03	10.0.0.9

1.4 web集群服务器配置文件环境统一（web01,web02,web03配置均一致）

1.4.1 web集群配置如下

```
1.web01 web02 web03 配置相同
[root@ks_web01 conf]# vim nginx.conf
worker_processes 1;
events {
    worker_connections 1024;
}
http {
    '$status $body_bytes_sent "$http_referer" '
    '$http_user_agent' "$http_x_forwarded_for";

    server {
        listen 80;
        server_name bbs.etiantian.org;
        root html/bbs;
        index index.html index.htm;
        access_log logs/www_access.log main;
    }
}
```

```
server {
    listen      80;
    server_name www.etiantian.org;
    root        html/www;
    index       index.html index.htm;
    access_log  logs/www_access.log main;
}
}
```

同步web02上的配置文件

同步web03上的配置文件

```
scp -rp /application/nginx/conf/nginx.conf 172.16.1.8:/application/nginx/conf/nginx.conf
scp -rp /application/nginx/conf/nginx.conf 172.16.1.9:/application/nginx/conf/nginx.conf
```

测试结果:

```
[root@b101 conf]# curl -H host:www.etiantian.org 10.0.0.83/bingbing.html
www web02
[root@b101 conf]# curl -H host:www.etiantian.org 10.0.0.82/bingbing.html
www web01
```

1.4.2 nginx反向代理负载均衡集群服务器配置文件环境统一

```
worker_processes 1;
events {
    worker_connections 1024;
}
http {
    include mime.types;
    default_type application/octet-stream;
    sendfile on;
    keepalive_timeout 65;
    upstream server_pools {
        server 10.0.0.7:80;
        server 10.0.0.8:80;
        server 10.0.0.9:80;
    }
    server {
        listen 80;
        server_name www.etiantian.org;
        location / {
            proxy_pass http://server_pools;
            proxy_set_header Host $host;
            proxy_set_header X-Forwarded-For $remote_addr;
        }
    }
    server {
        listen 80;
        server_name bbs.etiantian.org;
        location / {
            proxy_pass http://server_pools;
            proxy_set_header Host $host;
            proxy_set_header X-Forwarded-For $remote_addr;
        }
    }
}
```

同步b102 的nginx.conf配置文件

```
[root@b101 conf]# scp -rp nginx.conf 172.16.1.6:/application/nginx/conf/nginx.conf
172.16.1.6:/application/nginx/conf/nginx.conf
```

测试访问b101和b102实现负载均衡:

```
[root@b101 conf]# curl -H host:www.etiantian.org 10.0.0.5/bingbing.html
[root@b101 conf]# curl -H host:www.etiantian.org 10.0.0.6/bingbing.html
```

部署Keepalived软件

1.5 keepalived软件安装部署

1. keepalived软件安装部署

```
### 1b01 1b02负载服务器上均安装
yum install -y keepalived
rpm -qa keepalived
```

keepalived包含的主要配置:

```
[root@1b01 conf]# rpm -ql keepalived
/etc/keepalived          --- keepalived主目录
/etc/keepalived/keepalived.conf    --- keepalived服务主配置文件
/etc/rc.d/init.d/keepalived    --- keepalived服务启动脚本文件
```

2. 进行默认配置测试

```
### 启动1b01 1b02的keepalived服务
/etc/init.d/keepalived start
```

```
[root@1b01 nginx]# ip a
```

说明: 存在默认配置虚IP地址信息

通过抓包可以看到vrrp数据包信息

1.5.1 配置文件说明

前提需要了解配置文件内容信息 (man keepalived.conf)

配置文件的组成部分

- GLOBAL CONFIGURATION ###全局定义 (默认配置文件的01-13行)
- VRRPD CONFIGURATION ###虚拟ip的配置 (默认配置文件15-30行)
- LVS CONFIGURATION ###配置与管理lvs

! Configuration File for keepalived ---注释信息

```
=====
global_defs {                                --- 全局配置标题
    notification_email {                    --- 定义管理员邮箱信息,
        976060283@qq.com
        976060283@qq.com
    }
    notification_email_from oldboy@163.com --- 定义利用什么邮箱发送邮件(如: 163)
    smtp_server smtp.163.com                --- 定义邮件服务器信息
    smtp_connect_timeout 30                 --- 定义邮件发送超时时间
    router_id oldboy01                     --- (重点参数) 局域网keepalived主机身份标识信息
                                           #每一个keepalived主机身份标识信息唯一
}

=====
vrrp_instance VI_1 {                        --- vrrp协议相关配置(vip地址设置)
    state MASTER                            --- keepalived角色描述(状态)信息, 可以配置参数(MASTER BACKUP)
    interface eth0                          --- 表示将生成虚IP地址, 设置在指定的网卡上(一般为外网卡)
    virtual_router_id 51                    --- 表示keepalived家族标识信息
    priority 100                            --- keepalived服务竞选主备服务器优先级设置(越大越优先)
    advert_int 1                            --- 主服务组播包发送间隔时间
    authentication {                       --- 主备主机之间通讯认证机制,
        auth_type PASS                     --- 采用明文认证机制
        auth_pass 1111                    --- 编写明文密码(用于主备认证)
    }
    virtual_ipaddress {                    --- 设置虚拟IP地址信息
        10.0.0.3
    }
}
```

1.5.2 搭建基础的keepalived配置文件

#主1b01 的配置:

```
global_defs {
    router_id 1b01
}

vrrp_instance oldboy_43 {
```

```

state MASTER
interface eth0
virtual_router_id 43
priority 150
advert_int 1          ——必须与备服务器保持一致
authentication {
    auth_type PASS
    auth_pass 6666
}
virtual_ipaddress {
    10.0.0.3
}
}

```

虚拟IP地址显示信息:

默认显示信息: inet 10.0.0.3/32 scope global eth0

修改显示信息为: inet 10.0.0.3/24 scope global secondary eth0:1

#lb02 的配置:

```

global_defs {
    router_id lb02
}

vrrp_instance oldboy_43 {
    state BACKUP
    interface eth0
    virtual_router_id 43
    priority 100
    advert_int 1          ——必须与主服务器保持一致
    authentication {
        auth_type PASS
        auth_pass 6666
    }
    virtual_ipaddress {
        10.0.0.3
    }
}

```

说明: 主备服务器配置文件区别

01. router_id 不同
02. state BACKUP 不同
03. priority 不同

说明: 进行抓包观察配置效果; 并且对比两个负载均衡服务器的配置文件

1.5.3 主备服务器都启动keepalived服务

```

[root@b101 conf]# /etc/init.d/keepalived start
Starting keepalived: [ OK ]
[root@b102 conf]# /etc/init.d/keepalived start
Starting keepalived: [ OK ]
/etc/init.d/keepalived restart ——重启
/etc/init.d/keepalived stop ——停止
/etc/init.d/keepalived reload ——重新加载 (如果服务stop后, reload是起不来的)

```

1.5.4 裂脑概念, 原因, 解决方法

10.4.1 什么是裂脑

由于某些原因,导致两台高可用服务器对在指定时间内,无法检测到对方的心跳消息,各自取得资源及服务的所有权,而此时的两台高可用服务器对都还活着并在正常运行,这样就会导致同一个 IP 或服务在两端同时存在而发生冲突,最严重的是两台主机占用同一个 VIP 地址,当用户写入数据时可能会分别写入到两端,这可能会导致服务器两端的数据不一致或造成数据丢失,这种情况就被称为裂脑。

@51CTO博客

10.4.2 导致裂脑发生的原因

一般来说,裂脑的发生,有以下几种原因:

- 高可用服务器对之间心跳线链路发生故障,导致无法正常通信。
 - ▣ 心跳线坏了(包括断了,老化)。
 - ▣ 网卡及相关驱动坏了,IP配置及冲突问题(网卡直连)。
 - ▣ 心跳线间连接的设备故障(网卡及交换机)。
 - ▣ 仲裁的机器出问题(采用仲裁的方案)。
- 高可用服务器上开启了 iptables 防火墙阻挡了心跳消息传输。
- 高可用服务器上心跳网卡地址等信息配置不正确,导致发送心跳失败。
- 其他服务配置不当等原因,如心跳方式不同,心跳广播冲突、软件 Bug 等。

提示:Keepalived 配置里同一 VRRP 实例如果 virtual_router_id 两端参数配置不一致,也会导致裂脑问题发生。

@51CTO博客

10.4.3 解决裂脑的常见方案

在实际生产环境中,我们可以从以下几个方面来防止裂脑问题的发生:

- 同时使用串行电缆和以太网电缆连接,同时用两条心跳线路,这样一条线路坏了,另一个还是好的,依然能传送心跳消息。
- 当检测到裂脑时强行关闭一个心跳节点(这个功能需特殊设备支持,如 Stonith、fence)。相当于备节点接收不到心跳消息,通过单独的线路发送关机命令关闭主节点的电源。
- 做好对裂脑的监控报警(如邮件及手机短信等或值班),在问题发生时人为第一时间介入仲裁,降低损失。例如,百度的监控报警短信就有上行和下行的区别。报警信息发送到运维

编写脚本解决脑裂:

编写脚本的思路:

如何进行监控脑裂情况?

01. 在什么服务器上监控?

在备服务器上监控

02. 监控什么信息?

备上面出现vip（虚ip）情况

- 1) 脑裂情况会出现
- 2) 正常主备切换也会出现

方法一：

```
#!/bin/bash
IP_info=$(ip a|grep -c eth0)
```

```
while true
do
if [ $IP_info -gt 2 ]
then
echo "keepalived error!!!"
fi
sleep 3;
done
```

方法二：

```
#!/bin/bash
#desc: jiankong lb02 vip
if [ `ip a s eth0 |grep -c "10.0.0.3"` == 1 ];then
echo "baojing"
fi
```

1.6 拍错过程思路（最小化拍错）

拍错过程：

1. 利用负载均衡服务器，在服务器上curl所有节点信息
2. curl负载均衡服务器地址，可以实现负载均衡
3. windows绑定虚拟ip, 浏览器上进行测试

查看keepalived日志文件：

```
tail -f /var/log/messages
```

PS：所有服务通用拍错方法：

1. 对服务非常了解
2. 没有办法的解决办法
 01. 重启服务
 02. 重新部署
 03. 重装系统

1.7 更改nginx反向代理只监听vip地址

#提高安全性：

通过nginx反向代理只监听vip地址，不监听物理地址

通过修改配置文件监听窗口，lb02k无法监听（没有vip），需要修改内核文件

更改nginx反向代理只监听vip地址：

10.0.0.3

第一个里程碑：修改反向代理服务配置文件，只监听vip地址

####lb01 lb02 nginx.conf （1 和2 都配置监听地址）

```
vim nginx.conf
worker_processes 1;
events {
    worker_connections 1024;
}
http {
    include mime.types;
    default_type application/octet-stream;
    sendfile on;
    keepalive_timeout 65;
    log_format main '$remote_addr - $remote_user [$time_local] "$request" '
        '$status $body_bytes_sent "$http_referer" '
        '"$http_user_agent" "$http_x_forwarded_for"';
```

```

upstream server_pools {
    server 10.0.0.7;
    server 10.0.0.8;
    server 10.0.0.9;
}

server {
    listen 10.0.0.3:80;
    server_name www.etiantian.org;
    location / {
        proxy_pass http://server_pools;
    }
}
proxy_set_header Host $host;
proxy_set_header X-Forwarded-For $remote_addr;
access_log logs/access_www.log main;

server {
    listen 10.0.0.3:80;
    server_name blog.etiantian.org;
    location / {
        proxy_pass http://server_pools;
    }
}
proxy_set_header Host $host;
proxy_set_header X-Forwarded-For $remote_addr;
access_log logs/access_blog.log main;
}
}

```

说明（安全性问题）：在修改反向代理服务器配置文件监听地址时，多个server都需要配置监听地址，否则仍旧使用默认监听所有

第二个里程碑：1b02上不存在vip地址，无法监听，解决方法：需要修改内核文件（见下面）

```

[root@1b02 conf]# /application/nginx/sbin/nginx -t
nginx: the configuration file /application/nginx-1.10.2/conf/nginx.conf syntax is ok
nginx: [emerg] bind() to 10.0.0.3:80 failed (99: )
nginx: configuration file /application/nginx-1.10.2/conf/nginx.conf test failed

```

说明:nginx 没有办法监听本地不存在的vip地址

解决方法：

方法一：echo 'net.ipv4.ip_nonlocal_bind = 1' >>/etc/sysctl.conf ---实现监听本地不存在的vip地址

##/etc/sysctl.conf 加上

sysctl -p

方法二： echo "1" >/proc/sys/net/ipv4/ip_nonlocal_bind

第三个里程碑：进行测试

```
curl -H host:www.etiantian.org 10.0.0.3/bingbing.html
```

web403报错原因：

01 服务端阻止客户端访问

02 服务端首页文件不存在

1.8 让keepalived监控nginx反向代理服务

###vip什么时候 什么条件 才会飘走？

1. 当服务器宕机
2. 防火墙

nginx挂了

如何让keepalived监控nginx nginx挂了，keepalived跟着殉情

####第一个里程碑-keepalived监控nginx条件

1. 如何nginx挂了---我如何知道nginx挂了?

1) 端口

2) 进程

```
netstat -lnptu|grep nginx|wc -l
```

如果统计结果不等于1, 说明nginx没起来

那么就 /etc/init.d/keepalived stop

```
##> -gt      greater than
```

```
##>= -ge     greater equal
```

```
##< -lt      less than
```

```
##<= -le     less equal
```

```
##== -eq     equal
```

```
##!= -ne     no equal
```

####第二个里程碑-根据条件-书写脚本

```
vim check_web.sh
```

```
#!/bin/bash
```

```
#name: check_web.sh
```

```
#desc: check nginx and kill keepalived
```

```
if [ `netstat -lnptu |grep nginx |wc -l` -ne 1 ];then
    /etc/init.d/keepalived stop
fi
```

####第三个里程碑-添加执行权限

```
[root@lb02 conf]# chmod +x /server/scripts/check_web.sh
```

```
[root@lb02 conf]# ll /server/scripts/check_web.sh
```

```
-rwxr-xr-x 1 root root 174 Mar 30 17:47 /server/scripts/check_web.sh
```

####第四个里程碑-测试

执行脚本

停止nginx, 看keepalived是否停止

并查看vip是否飘到lb02

####第五个里程碑-放入到keepalived.conf

####下面是lb02的配置文件 lb01上面自己修改下。

```
global_defs {
    router_id LVS_02
}
```

```
vrrip_script check_web {
    script "/server/scripts/check_web.sh"  --- 表示将一个脚本信息赋值给变量check_web
    interval 2                             --- 执行监控脚本的间隔时间
    weight 2                               --- 利用权重值和优先级进行运算, 从而降低主服务器优先级
                                           使之变为备服务器
}
```

```
vrrip_instance VI_1 {
    state BACKUP
    interface eth0
    virtual_router_id 51
    priority 100
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 1111
    }
    virtual_ipaddress {
        10.0.0.3/24 dev eth0 label eth0:1
    }
    track_script {
        check_web
    }
}
```

####第六个里程碑-测试

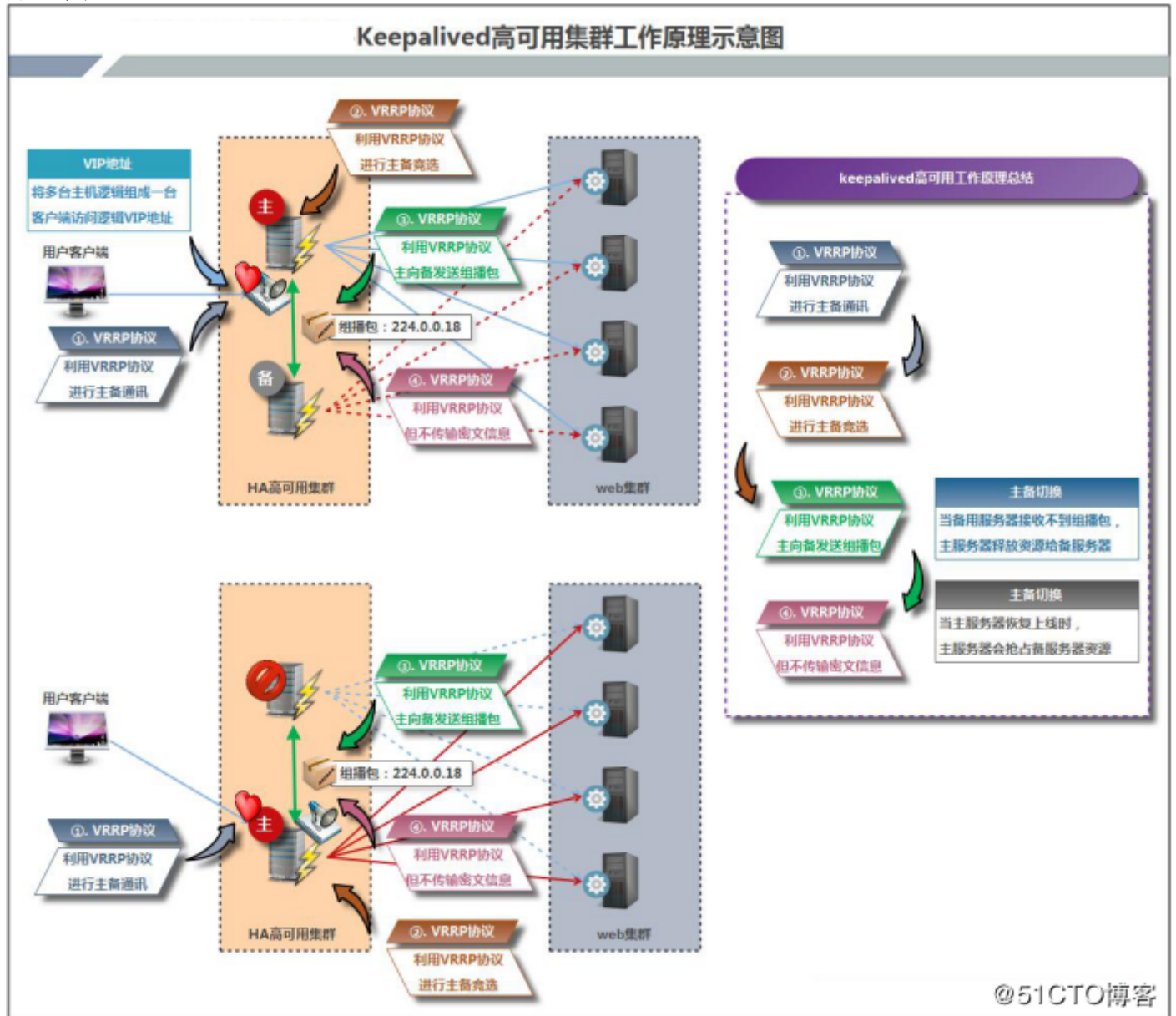
/etc/init.d/keepalived reload

先确定当前nginx服务已开启，并且可以查看到vip信息

然后停止 nginx服务，在查看vip是否飘到lb02

1.9 双实例双主模式的配置（互为主备）

原理图：



1b01 双主配置文件信息

```
[root@lb01 scripts]# cat /etc/keepalived/keepalived.conf
! Configuration File for keepalived
global_defs {
    router_id lb01
}
```

```
vrrp_instance oldboy_43_1 {
    state MASTER
    interface eth0
    virtual_router_id 43
    priority 150
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 6666
    }
    virtual_ipaddress {
```

```

        10.0.0.3
    }
}
vrrp_instance oldboy_43_2 {
    state BACKUP
    interface eth0
    virtual_router_id 44
    priority 100
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 6666
    }
    virtual_ipaddress {
        10.0.0.4
    }
}

```

lb02 双主配置信息

```

[root@lb02 ~]# cat /etc/keepalived/keepalived.conf
! Configuration File for keepalived
global_defs {
    router_id lb02
}

```

```

vrrp_instance oldboy_43_1 {
    state BACKUP
    interface eth0
    virtual_router_id 43
    priority 100
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 6666
    }
    virtual_ipaddress {
        10.0.0.3
    }
}
vrrp_instance oldboy_43_2 {
    state MASTER
    interface eth0
    virtual_router_id 44
    priority 150
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass 6666
    }
    virtual_ipaddress {
        10.0.0.4
    }
}

```

测试抓包进行分析:

www.etiantian.org

可以实现负载均衡，而且主服务器宕掉后，vip会飘到备服务器继续工作

小伙伴们可以关注我的微信公众号：linux运维菜鸟之旅



关注“中国电信天津网厅”公众号，首次绑定可免费领2G流量，为你的学习提供流量！



版权声明：原创作品，如需转载，请注明出处。否则将追究法律责任