|  |
| --- |
| 文章标题 |
| **作者：**  **归档：学习笔记** |
| **快捷键：**  Ctrl + 1 标题1  Ctrl + 2 标题2  Ctrl + 3 标题3  Ctrl + 4 实例  Ctrl + 5 程序代码  Ctrl + 6 正文 |
| **格式说明：**  蓝色字体：注释  黄色背景：重要  绿色背景：注意 |

****

**老男孩教育教学核心思想6重：重目标、重思路、重方法、重实践、重习惯、重总结**

**学无止境，老男孩教育成就你人生的起点！**

# keepalived

## keepalived软件能干什么?

Keepalived软件起初是专为LVS负载均衡软件设计的，

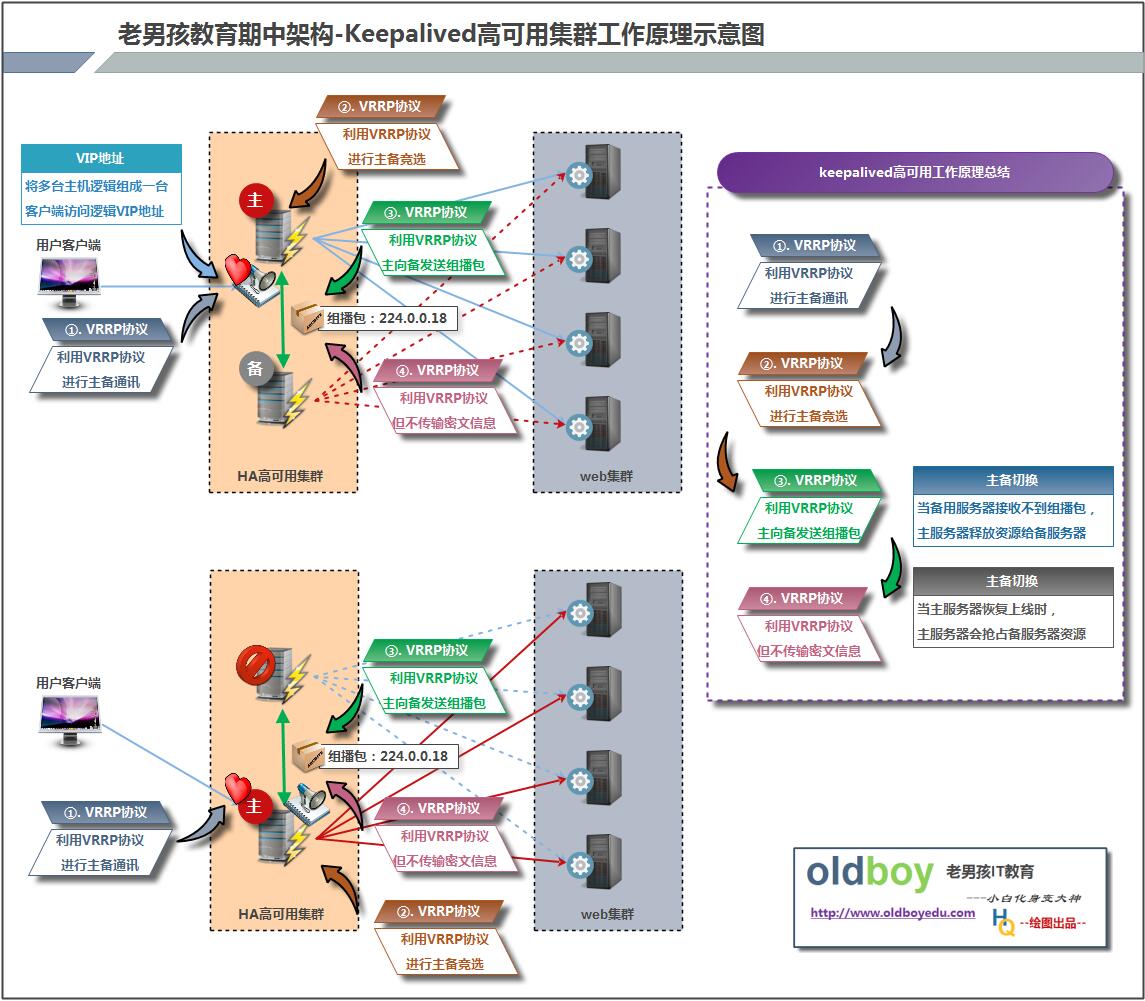
用来管理并监控LVS集群系统中各个服务节点的状态，后来又加入了可以实现高可用的VRRP功能

①. 管理LVS负载均衡软件

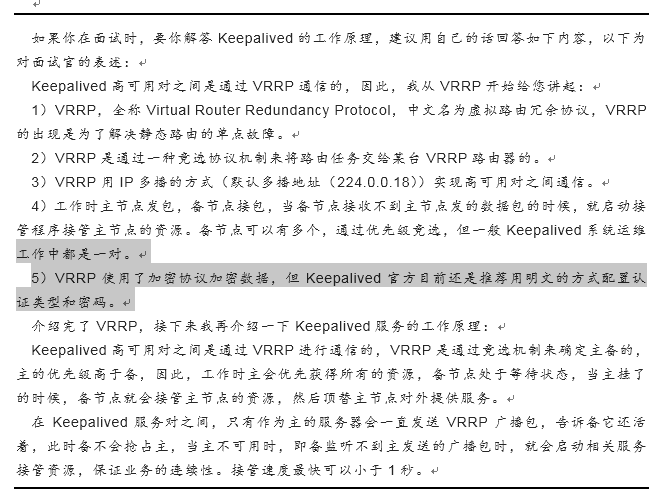
②. 实现对LVS集群节点健康检查功能

③. 作为系统网络服务的高可用功能

## keepalived软件工作原理？(重点)



1. VRRP协议，全称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的单点故障。



2）VRRP是用过IP多播的方式（默认多播地址（224.0.0.18））实现高可用对之间通信的。

3）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。

## 统一环境配置

### 环境准备说明：

lb01 10.0.0.5

lb02 10.0.0.6

web01 10.0.0.8

web02 10.0.0.7

web03 10.0.0.9

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

cat www.conf

server {

listen 80;

server\_name www.etiantian.org;

location / {

root html/www;

index index.html index.htm;

}

}

cat bbs.conf

server {

listen 80;

server\_name bbs.etiantian.org;

location / {

root html/bbs;

index index.html index.htm;

}

}

### 同步三台web服务器配置

scp -rp {www.conf,bbs.conf} 172.16.1.7:/application/nginx/conf/extra/

scp -rp {www.conf,bbs.conf} 172.16.1.9:/application/nginx/conf/extra/

### web服务主配置文件环境统一

[root@web01 extra]# cat ../nginx.conf

worker\_processes 1;

events {

worker\_connections 1024;

}

http {

log\_format main '$remote\_addr - $remote\_user [$time\_local] "$request" '

'$status $body\_bytes\_sent "$http\_referer" '

'"$http\_user\_agent" "$http\_x\_forwarded\_for"';

access\_log logs/access.log main;

include mime.types;

default\_type application/octet-stream;

sendfile on;

keepalive\_timeout 65;

include extra/www.conf;

include extra/bbs.conf;

}

#### 发送到其他web服务器

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

scp -rp ../nginx.conf 172.16.1.7:/application/nginx/conf/

### 测试

#### web01测试环境准备

[root@web01 www]# for name in www bbs;do echo $name `hostname` >/application/nginx/html/$name/bingbing.html;done

[root@web01 www]# for name in www bbs;do cat /application/nginx/html/$name/bingbing.html;done

www web01

bbs web01

#### web02测试环境准备

[root@web02 conf]# for name in www bbs;do echo $name `hostname` >/application/nginx/html/$name/bingbing.html;done

[root@web02 conf]# for name in www bbs;do cat /application/nginx/html/$name/bingbing.html;done

www web02

bbs web02

#### web03测试环境准备

[root@web03 conf]# for name in www bbs;do echo $name `hostname` >/application/nginx/html/$name/bingbing.html;done

[root@web03 conf]# for name in www bbs;do cat /application/nginx/html/$name/bingbing.html;done

www web03

bbs web03

#### 测试环境搭建好重启服务

/application/nginx/sbin/nginx -t

/application/nginx/sbin/nginx -s reload

#### web环境测试结果

[root@web01 www]# curl -H host:www.etiantian.org 10.0.0.8/bingbing.html

www web01

[root@web01 www]# curl -H host:bbs.etiantian.org 10.0.0.8/bingbing.html

bbs web01

[root@web01 www]# curl -H host:www.etiantian.org 10.0.0.7/bingbing.html

www web02

[root@web01 www]# curl -H host:bbs.etiantian.org 10.0.0.7/bingbing.html

bbs web02

[root@web01 www]# curl -H host:www.etiantian.org 10.0.0.9/bingbing.html

www web03

[root@web01 www]# curl -H host:bbs.etiantian.org 10.0.0.9/bingbing.html

bbs web03

## nginx反向代理负载均衡集群服务器配置文件环境统一

### lb01 配置

[root@lb01 conf]# cat nginx.conf

####lb01 nginx.conf

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;

}

}

}

### lb02 配置发送

scp -rp /application/nginx/conf/nginx.conf 172.16.1.6:/application/nginx/conf/

## keepalived 配置部署

### 第一个里程碑：keepalived软件安装部署

lb01 lb02负载服务器上均安装

#### 下载安装

yum install -y keepalived

rpm -qa keepalived

rpm -ql keepalived

#### 查看安装的主要文件

[root@lb01 conf]# rpm -ql keepalived

/etc/keepalived

/etc/keepalived/keepalived.conf 配置文件

/etc/rc.d/init.d/keepalived 启动文件

### 第二个里程碑：进行默认配置测试

#### 启动lb01 lb02的keepalived服务

/etc/init.d/keepalived start

#### ip addr

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

### 第三个里程碑：进行服务配置文件编写

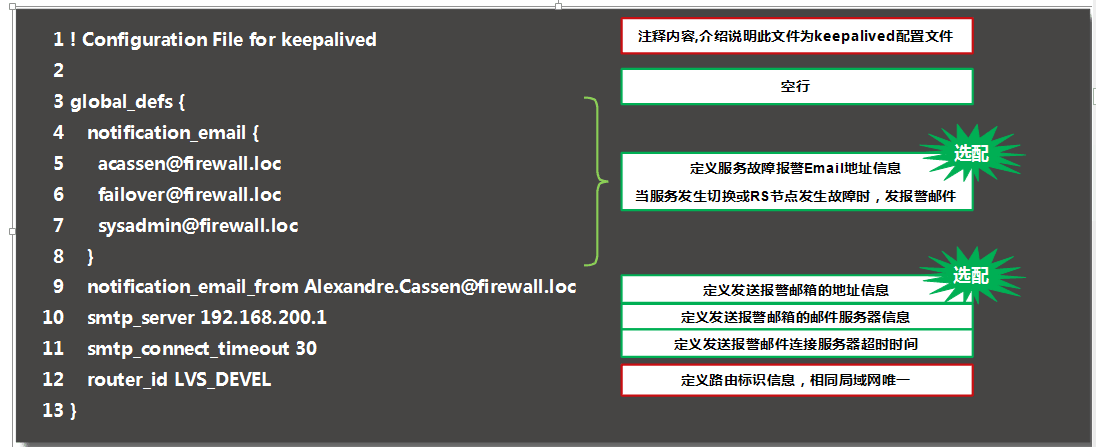
#### 配置文件的组成部分

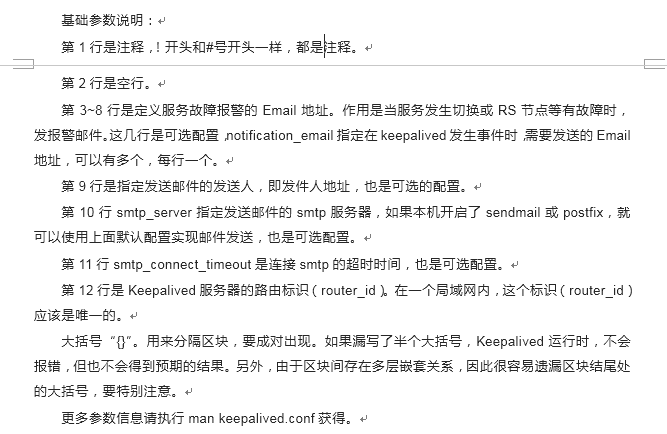
· GLOBAL CONFIGURATION ###全局定义（默认配置文件的01-13行）

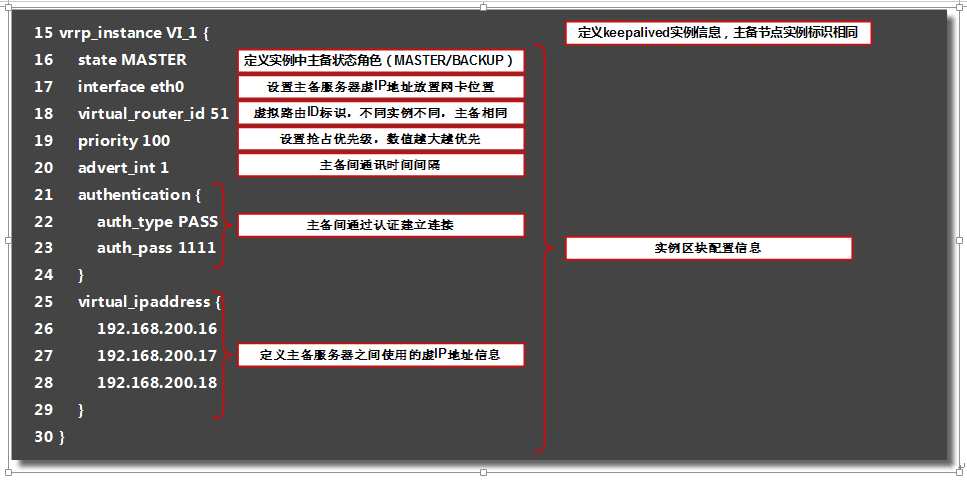
· VRRPD CONFIGURATION ###虚拟ip的配置（默认配置文件15-30行）

· LVS CONFIGURATION ###配置与管理lvs

#### 配置参数说明







### 搭建基础的keepalived配置文件

#### lb01配置

global\_defs {

router\_id LVS\_01

}

vrrp\_instance VI\_1 {

state MASTER

interface eth0

virtual\_router\_id 51

priority 150

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1111

}

virtual\_ipaddress {

10.0.0.3/24 dev eth0 label eth0:1

}

}

#### lb02 配置

global\_defs {

router\_id LVS\_02

}

vrrp\_instance VI\_1 {

state BACKUP

interface eth0

virtual\_router\_id 51

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1111

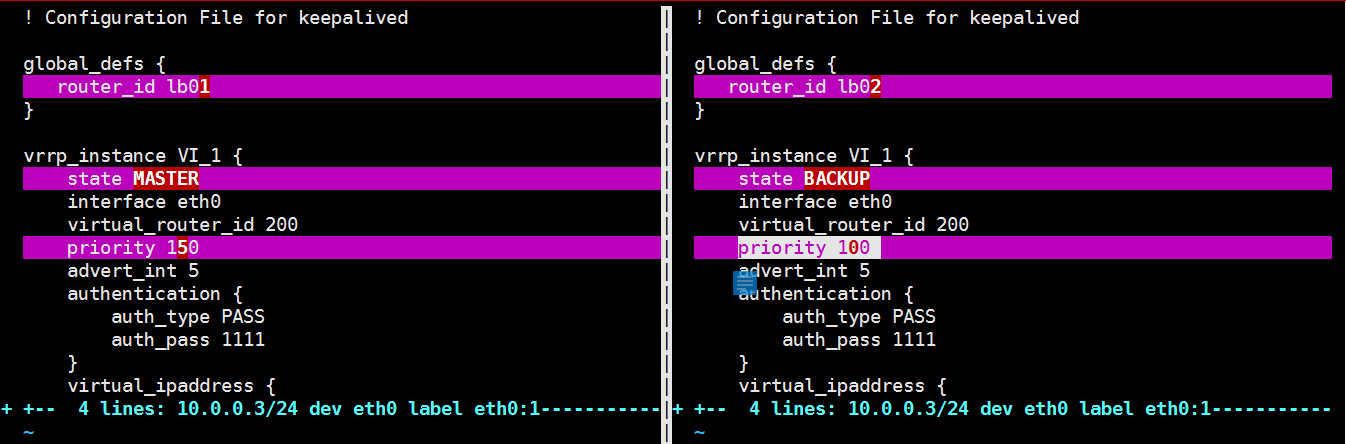
}

virtual\_ipaddress {

10.0.0.3/24 dev eth0 label eth0:1

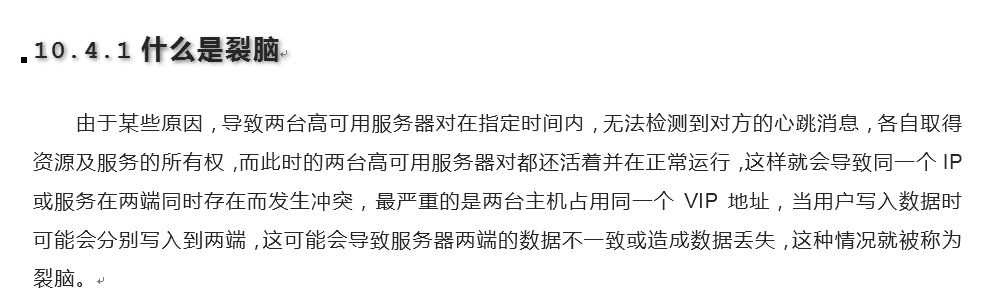
}

}

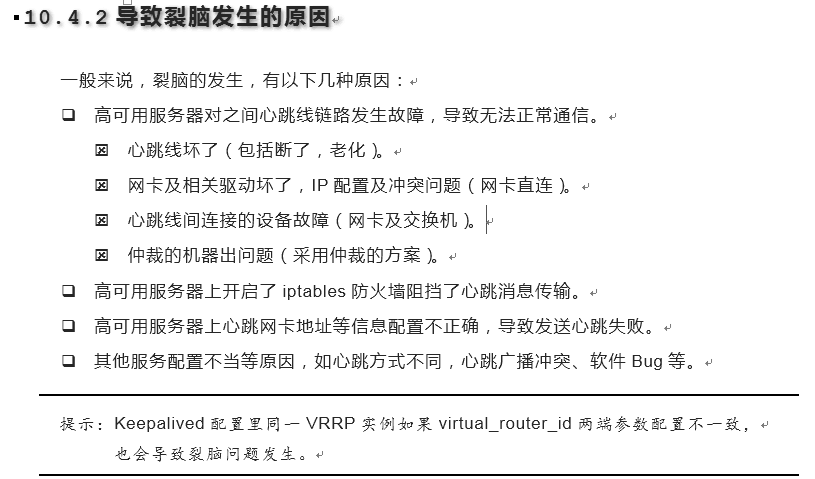


说明：进行抓包观察配置效果；并且对比两个负载均衡服务器的配置文件

## keepaliver软件脑裂概念说明



### 脑裂情况出现的原因



#### 防火墙开启状态

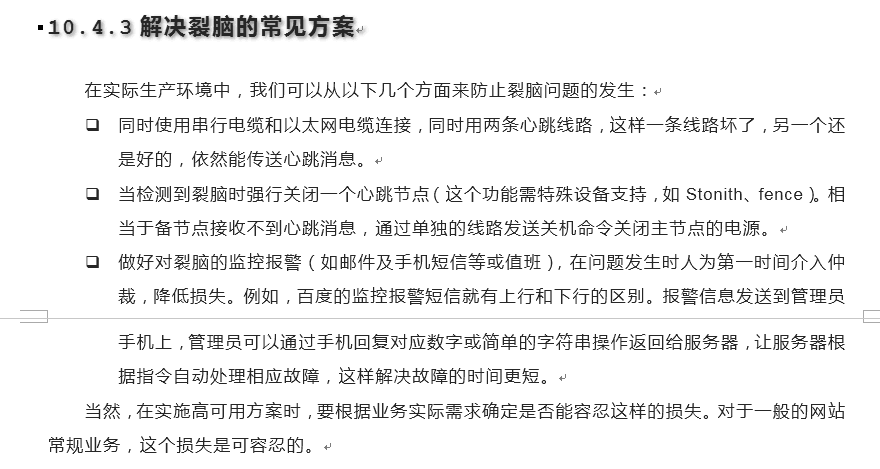
开启防火墙即可模拟出脑裂的情况

/etc/init.d/iptables start

#### 心跳线断掉

#### nginx 关闭

### 脑裂情况解决的方法



#### 制作监控脚本---lb02

报警的条件：只要lb02 上面有vip

1. lb01 挂了
2. 心碎

如果lb02 出现10.0.0.3 虚拟IP，则说明lb02 成为主负载机，lb01 挂掉

#!/bin/bash

#desc: jiankong lb02 vip

if [ `ip a s eth0 |grep -c "10.0.0.3"` == 1 ];then

echo "baojing"

fi

### nginx反向代理-负载均衡 ---做高可用

#### 统一lb01 lb02 反向代理 配置文件

####lb01 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 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;

}

access\_log logs/access\_www.log main;

}

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;

}

access\_log logs/access\_blog.log main;

}

}

#### 进行测试

## 01 测试10.0.0.5 lb01服务器

curl -H Host:www.etiantian.org 10.0.0.5/oldboy.html

curl -H Host:bbs.etiantian.org 10.0.0.5/oldboy.html

## 01 测试10.0.0.6 lb01服务器

curl -H Host:www.etiantian.org 10.0.0.6/oldboy.html

curl -H Host:bbs.etiantian.org 10.0.0.6/oldboy.html

说明：通过以上测试，确认两台lb服务器，均可实现负载调度功能

#### 把域名解析到 vip上面

10.0.0.3 www.etiantian.org blog.etiantian.org bbs.etiantian.org

#### 问题小结：

1.是否解析 ping

2.浏览器缓存

3.服务没重启

### 排错过程：

01：利用负载服务器，在服务器上curl所有节点信息（web服务器配置有问题）

02；curl 负载均衡服务器地址，可以实现负载均衡

03：windows绑定虚拟IP，浏览器上进行测试

## 更改nginx反向代理只监听vip地址

10.0.0.3/bingbing.html 可以使用

10.0.0.5/bingbing.html 不可以使用

10.0.0.6/bingbing.html 不可以使用

### 第一个里程碑：修改反向代理服务配置文件，只监听vip地址

#### 修改 lb01 lb02 nginx.conf 配置文件

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：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：lb02上不存在vip地址，无法监听，需要修改内核文件

[root@lb01 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

[root@lb01 conf]# ip a s eth0

2: eth0: <BROADCAST,MULTICAST,UP,LOWER\_UP> mtu 1500 qdisc pfifo\_fast state UP qlen 1000

link/ether 00:0c:29:27:4e:e9 brd ff:ff:ff:ff:ff:ff

inet 10.0.0.5/24 brd 10.0.0.255 scope global eth0

inet6 fe80::20c:29ff:fe27:4ee9/64 scope link

valid\_lft forever preferred\_lft forever

无法监听，不存在虚拟IP10.0.0.3

#### 解决办法

echo 'net.ipv4.ip\_nonlocal\_bind = 1' >>/etc/sysctl.conf ---实现监听本地不存在的ip地址

##/etc/sysctl.conf 加上

sysctl -p 加上后生效

echo "1" >/proc/sys/net/ipv4/ip\_nonlocal\_bind

## 让keepalived监控 nginx

### 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

如何知道nginx挂了

1）端口

2）进程

ps -ef |grep nginx |grep -v grep |wc -l

#### 第二个里程碑-根据条件-书写脚

Nginx服务占用一个进程，脚本占用一个进程，如果统计数值小于2 则说明Nginx未启动，则关闭keepalived

#!/bin/bash

#name: check\_web.sh

#desc: check nginx and kill keepalived

if [ `ps -ef |grep nginx |grep -v grep |wc -l` -lt 2 ];then

/etc/init.d/keepalived stop

fi

#### 第三个里程碑-添加权限

[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

#### 第五个里程碑-把脚本放入到keepalived.conf

下面是lb02的配置文件

global\_defs {

router\_id LVS\_02

}

vrrp\_script check\_web {

script "/server/scripts/check\_web.sh" 执行的脚本

interval 2

weight 2

}

vrrp\_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

}

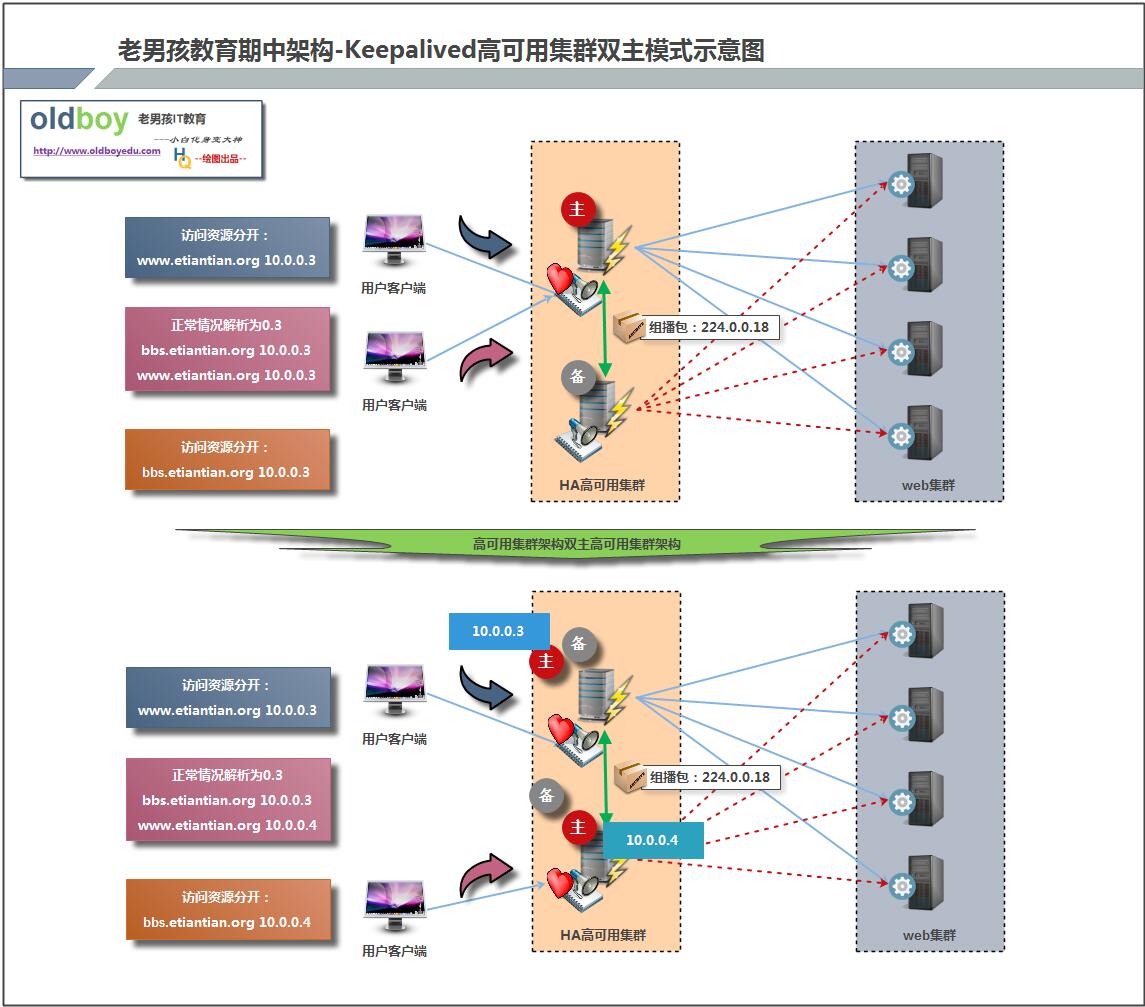
track\_script {

check\_web 执行

}

}

## keepalived多实例配置



### 第一个里程碑-配置keepalived-配置双主

#### lb01 配置

global\_defs {

router\_id LVS\_01

}

vrrp\_instance VI\_1 {

state MASTER

interface eth0

virtual\_router\_id 51

priority 150

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1111

}

virtual\_ipaddress {

10.0.0.3/24

}

}

vrrp\_instance VI\_2 {

state BACKUP

interface eth0

virtual\_router\_id 52

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 2222

}

virtual\_ipaddress {

10.0.0.4/24

}

}

virtual\_server 10.0.0.3 80 {

delay\_loop 6

lb\_algo wrr

lb\_kind DR

nat\_mask 255.255.255.0

persistence\_timeout 50

protocol TCP

real\_server 10.0.0.7 80 {

weight 1

TCP\_CHECK {

connect\_timeout 8

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

real\_server 10.0.0.8 80 {

weight 1

TCP\_CHECK {

connect\_timeout 8

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

}

virtual\_server 10.0.0.4 80 {

delay\_loop 6

lb\_algo wrr

lb\_kind DR

nat\_mask 255.255.255.0

persistence\_timeout 50

protocol TCP

real\_server 10.0.0.7 80 {

weight 1

TCP\_CHECK {

connect\_timeout 8

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

real\_server 10.0.0.8 80 {

weight 1

TCP\_CHECK {

connect\_timeout 8

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

}

#### lb02 配置

global\_defs {

router\_id LVS\_02

}

vrrp\_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

}

}

vrrp\_instance VI\_2 {

state MASTER

interface eth0

virtual\_router\_id 52

priority 150

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 2222

}

virtual\_ipaddress {

10.0.0.4/24

}

}

virtual\_server 10.0.0.3 80 {

delay\_loop 6

lb\_algo wrr

lb\_kind DR

nat\_mask 255.255.255.0

persistence\_timeout 50

protocol TCP

real\_server 10.0.0.7 80 {

weight 1

TCP\_CHECK {

connect\_timeout 8

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

real\_server 10.0.0.8 80 {

weight 1

TCP\_CHECK {

connect\_timeout 8

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

}

virtual\_server 10.0.0.4 80 {

delay\_loop 6

lb\_algo wrr

lb\_kind DR

nat\_mask 255.255.255.0

persistence\_timeout 50

protocol TCP

real\_server 10.0.0.7 80 {

weight 1

TCP\_CHECK {

connect\_timeout 8

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

real\_server 10.0.0.8 80 {

weight 1

TCP\_CHECK {

connect\_timeout 8

nb\_get\_retry 3

delay\_before\_retry 3

connect\_port 80

}

}

}

### 第二个里程碑-配置nginx 负载均衡

#### lb01 lb02 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 10.0.0.4:80;

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;

}

}

### 第三个里程碑-windows hosts解析

10.0.0.3 www.etiantian.org

10.0.0.4 bbs.etiantian.org

### 第四个里程碑-浏览器进行测试

## 总结本章重点回顾

Keepalived及VRRP协议原理。(\*\*\*\*\*\*)

Keepalived配置文件重要参数。(\*\*\*\*\*)

1.全局定义

2.模块-vrrp-实例

3.管理lvs

Keepalived单实例VIP漂移的配置及实战。

裂脑的概念及如何规避解决裂脑问题。\*\*\*\*\*

Keepalived单多实例结合Nginx代理配置实战。

服务监听网卡上不存在IP地址问题解决方案。

高可用服务只针对物理服务器问题解决方案。 keepalived监控nginx

多组Keepalived服务器对通信冲突问题解决方案（virtual router id 多播224.0.0.18）。

### 课后题目： 

配置指定文件接收Keepalived服务日志。

如何用系统的logrotate 切割nginx keepalived

开发监测Keepalived裂脑的思路及脚本。

### 思考：

nginx优化时还会讲解？

01：nginx软件已经在编译时指定用户了，如何进行修改

[root@lb02 conf]# /application/nginx/sbin/nginx -V

nginx version: nginx/1.10.3

built by gcc 4.4.7 20120313 (Red Hat 4.4.7-18) (GCC)

configure arguments: --user=nginx --group=nginx --prefix=/application/nginx-1.10.3 --with-http\_stub\_status\_module