Keepalived热备、HAProxy服务器

Keepalived

概述

问：调度器出现单点故障，如何解决？

* + - 1. Keepalived实现了高可用集群
      2. Keepalived最初是为了LVS设计的，专门监控各服务器节点的状态
      3. Keepalived后来加入了VRRP功能，防止单点故障

注：VRRP是一种选择协议，它可以把一个虚拟路由器的责任动态分配到局域网上的 VRRP 路由器中的一台。

VRRP是一种路由容错协议，也可以叫做备份路由协议

VRRP（Virtual Router Redundancy Protocol，虚拟路由冗余协

议）是一种容错协议

VRRP 将局域网的一组路由器（包括一个Master 即活动路由器和若干个Backup 即备份路由器）组织成一个虚拟路由器，称之为一个备份组。

VRRP的工作过程如下：

1. 路由器开启VRRP功能后，会根据优先级确定自己在备份组中的角色。优先级高的路由器成为主用路由器，优先级低的成为备用路由器。主用路由器定期发送VRRP通告报文，通知备份组内的其他路由器自己工作正常；备用路由器则启动定时器等待通告报文的到来。
2. VRRP在不同的主用抢占方式下，主用角色的替换方式不同：

l在抢占方式下，当主用路由器收到VRRP通告报文后，会将自己的优先级与通告报文中的优先级进行比较。如果大于通告报文中的优先级，则成为主用路由器；否则将保持备用状态。

l在非抢占方式下，只要主用路由器没有出现故障，备份组中的路由器始终保持主用或备用状态，备份组中的路由器即使随后被配置了更高的优先级也不会成为主用路由器。

1. 如果备用路由器的定时器超时后仍未收到主用路由器发送来的VRRP通告报文，则认为主用路由器已经无法正常工作，此时备用路由器会认为自己是主用路由器，并对外发送VRRP通告报文。备份组内的路由器根据优先级选举出主用路由器，承担报文的转发功能。
2. keepalived运行原理
3. Keepalived检测每个服务器节点状态
4. 服务器节点异常或工作出现故障，Keepalived将故障节点从集群系统红剔除
5. 故障节点恢复后，Keepalived再将其加入到集群系统中
6. 所有工作自动完成，无需人工干预

Keepalived+LVS

使用Keepalived高可用解决调度器单点失败问题

主、备调度器上配置LVS

主调度器异常时，Keepalived启用备用调度器

HAProxy服务器

概述：

* 1. 它是免费、快速并且可靠的一种解决方案
  2. 使用于那些负载特大的web站点，这些站点通常有需要会话保持或七层处理
  3. 提供高可用性、负载均衡以及基于TCP和HTTP应用的代理

衡量负载均衡器性能的因素

Seesion rate 会话率

* + - * + 每秒钟产生的会话数

Seesion concurrency并发会话数

* + - * + 服务器处理会话的时间越长，并发会话数越多

Data rate数据速率

* + - * + 以MB/s或Mbps衡量
        + 大的对象导致并发会话数增加
        + 高会话数、高数据速率要求更多的内存

HAProxy工作模式

1. mode http
   * + - * 客户端请求被深度分析后再发往服务器
   1. mode tcp

— 客户端与服务器之间建立会话，不检查第七层信息

* 1. mode health
     + - * 仅做健康状态检查，已经不建议使用

HTTP协议解析

HTTP解析

1. 当HAProxy运行在HTTP模式下，HTTP请求（Request）和响应（Response）均被完全分析和索引，这样便于创建恰当的匹配规则
2. 理解HTTP请求和响应，对于更好的创建匹配规则至关重要

HTTP事务模型

1. HTTP协议是事务驱动的
2. 每个请求仅能对应一个响应
3. 常见模型

* HTTP close
  + - * + 客户端向服务器建立一个TCP连接
        + 客户端发型请求给服务器
        + 服务器响应客户端请求后即断开连接
        + 如果客户端到服务器的请求不止一个，那么就要不断地去建立连接
        + TCP三次握手消耗相对较大的形同资源，同时延迟较大
* Keep-alive
  + - * + 一次连接可与传输多个请求
        + 客户端需要知道传输内容的长度，与避免无限期的等待传输结果
        + 降低两个HTTP事务间的延迟
        + 需要相对较少的服务器资源
* Pipelining
  + - * + 仍然使用Keep-alive
        + 在发送后续请求前，不用等前面的请求已经得到回应
        + 适用于有大量图片的页面
        + 降低了多次请求之间的网络延迟

HTTP头部信息

请求头部包含许多有关的客户端环境和请求正文的有用信息

URL ：统一资源定位符http：//www.baidu.com/music/mp3/xxx.mp3

URI ：除去协议剩下的部分/music/mp3/xxx.mp3

集群调度软件对比

Nginx分析

优点

* + - * + 工作在7层，可以针对http做分流策略
        + 正则表达式比HAProxy强大
        + 安装、配置、测试简单，通过日志可以解决多数问题
        + 并发量可以达到几万次
        + Nginx还可以作为Web服务器使用

缺点

* + - * + 仅支持http、https、mail协议，应用面小
        + 监控检查仅通过端口，无法使用URL检查

LVS分析

优点

* + - * + 负载能力强，工作在4层，对内存、CPU消耗低
        + 配置性低，没有太多可配置性，减少人为错误
        + 应用面广，几乎可以为所有应用提供负载均衡

缺点

* + - * + 不支持正则表达式，不能实现动静分离
        + 如果网站架构庞大，LVS-DR配置比较繁琐

HAProxy分析

优点

* + - * + 支持session、cokkie功能
        + 可以通过URL进行健康检查
        + 效率、负载均衡速度，高于Nginx，低于LVS
        + HAProxy支持TCP，可以对MYSQL进行负载均衡
        + 调度算法丰富

缺点

* + - * + 正则弱于Nginx
        + 日志依赖于syslogd，不支持apache日志

1. 使用Keepalived软件配置HA（高可用）集群

把主机55和主机57配置 提供网站的服务HA集群

配置步骤

1. 分别在主机55和57运行网站服务并编写网页文件test.html
2. 分别在主机55和57上安装Keepalived软件
3. 分别在主机55和57上修改配置文件

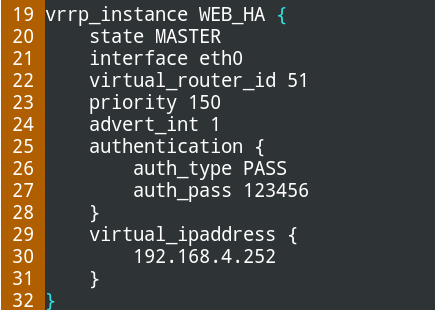
http://192.168.4.252/test.html

* 1. 配置主服务器（55）
     + 1. 备份主配置文件

# cp /etc/keepalived/keepalived.conf{,.bak}

* + - 1. 修改配置文件（必须注释掉 vrrp\_strict 行，其他内容修改如下图）

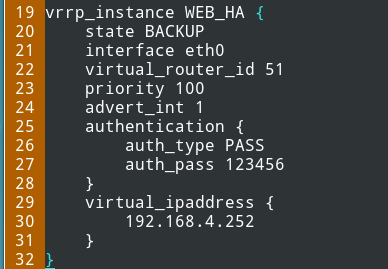
# vim /etc/keepalived/keepalived.conf



* 1. 配置备用服务器（57）

修改配置文件

# vim /etc/keepalived/keepalived.conf



1. 分别主机55和57上启动keepalived服务（先启动优先级高的）

# systemctl start keepalived.service

1. 查看分配的192.168.4.252IP地址

55主机 # ip add show | grep 192.168.4



57主机 # ip add show | grep 192.168.4



1. 测试是否能够通信（正常通信即可）

# ping 192.168.4.252

1. 测试高可用功能（保证55，57上面的HTTP服务是运行的）

# curl <http://192.168.4.252/test.html> （显示55web页面）

down掉55主机的keepalived服务，再次进行测试，则显示57web页面

1. 综合应用：keepalived+LVS 配置LVS调度器HA集群

需求：把主机56配置为备用的LVS调度器，当分发器54主机宕机后，客户端主机仍然可以连接VIP地址192.168.4.254 访问网站集群

配置步骤;

1. 在56主机上安装提供LVS服务的软件ipvsadm
2. 删除54主机上策略和VIP地址

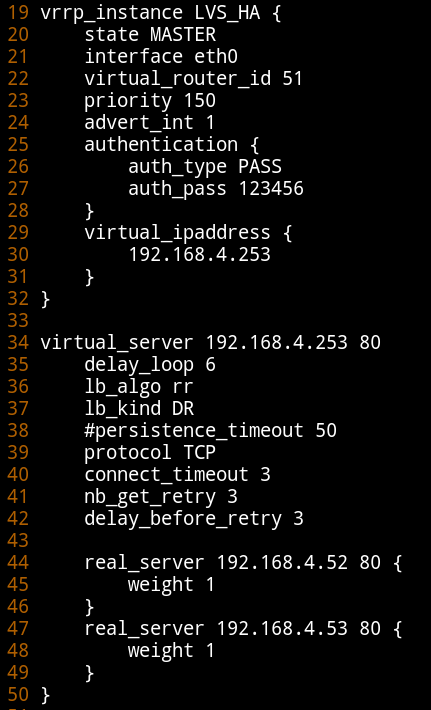
# ipvsadm-C

# ifdown eth0-------->ifup eth0

删除/etc/rc.local文件里的ifconfig eth0:1 192.168.4.253

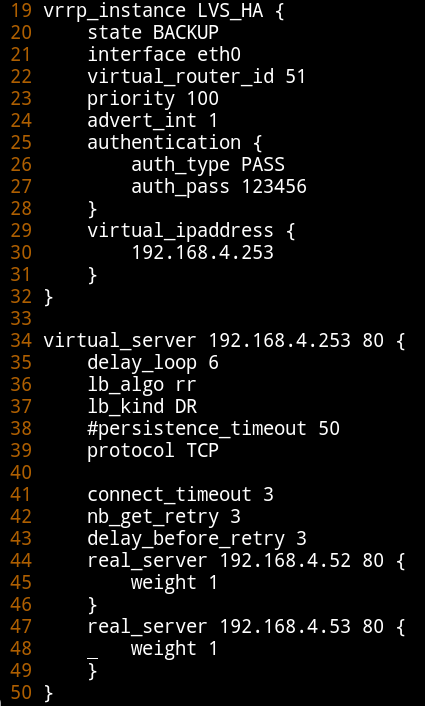
1. 分别在54和56主机上安装keepalived软件
2. 修改配置文件/etc/keepalived/keepalived.conf
   1. 主分发器host54（注释14行 vrrp\_strict）

#vim /etc/keepalived/keepalived.conf



* 1. 备用分发器host56（注释14行 vrrp\_strict）

#vim /etc/keepalived/keepalived.conf



1. 启动54 和56 主机上keepalived服务并设置为开机运行
2. 客户端连接VIP地址192.168.4.253 访问网站集群

客户端50 #curl curl <http://192.168.4.253/test.html>

轮询显示52和53主机上的web页面

1. 验证分发器，能否实现高可用

主机54宕机后，客户端依然可连接VIP地址192.168.4.253 访问网站集群

扩展：配置keepalived 对 LVS 服务的realserver 做健康性检查

修改主机 54 和 56 配置文件/etc/keepalived/keepalived.conf



1. 使用HAProxy软件 配置LB集群（基于端口、传输层）

环境准备：55 和 57网站服务器,客户端 50，分发器 56

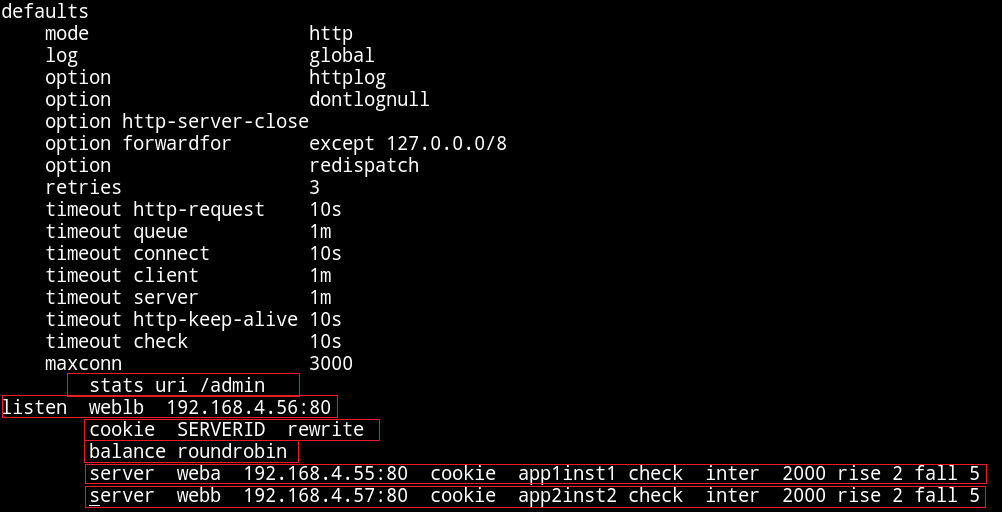
host56：

1. 装包haproxy
2. 修改配置文件/etc/haproxy/haproxy.cfg
   1. 备份

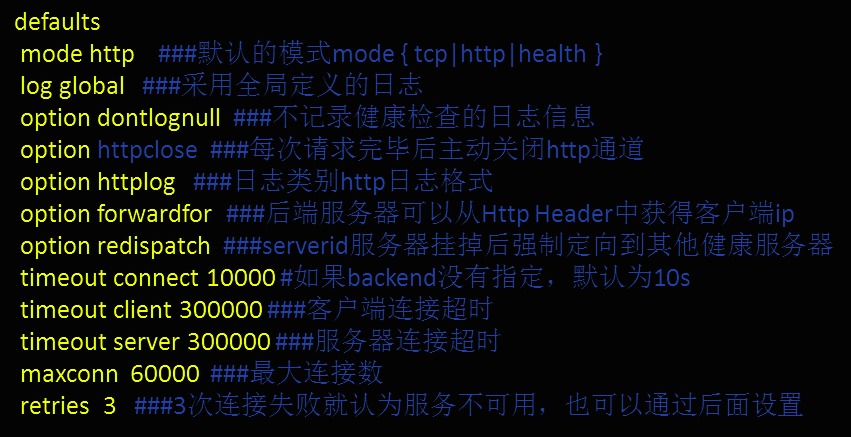
# cp /etc/haproxy/haproxy.cfg{,.bak}

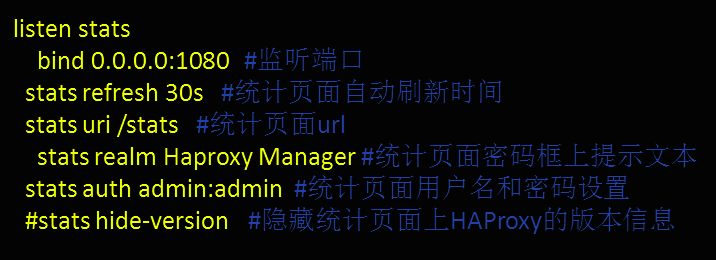
* 1. 修改

# /etc/haproxy/haproxy.cfg



配置文件说明：





1. 启动服务并设置为开机自启动

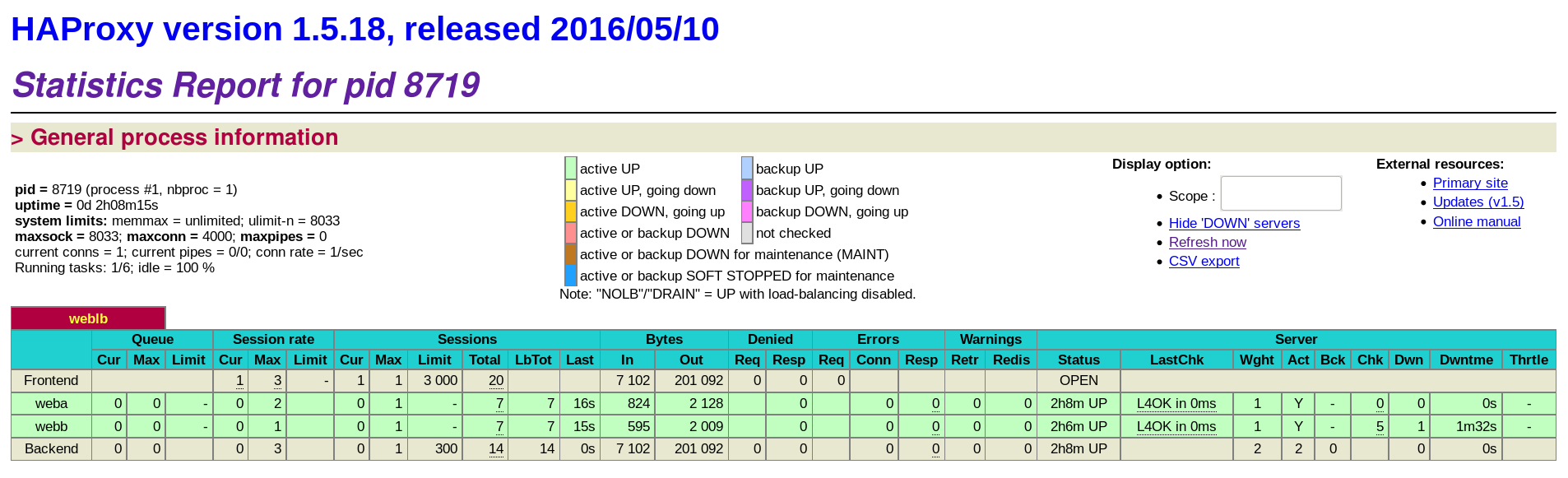
# systemctl start haproxy

# systemctl enable haproxy

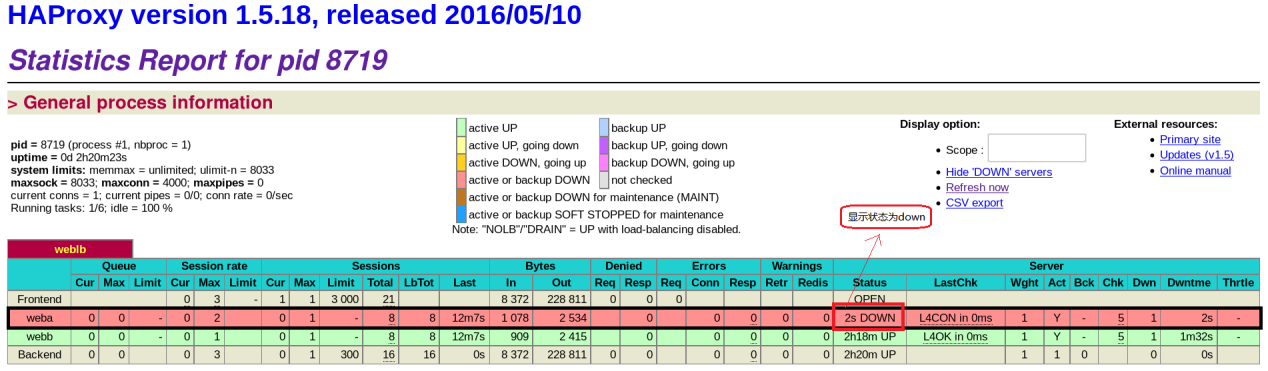
1. 客户端验证50
   1. # curl [http://192.168.4.56/test.html](http://192.168.4.54/test.html) （轮询显示55和57的web页面）

b) #浏览器访问 [http://192.168.4.56/admin](http://192.168.4.54/admin)

页面内容如下：



宕掉其中一台服务器的web服务，则显示如下：（以断开55为例）



再次打开55主机的web服务，则恢复轮询

参数备注：

Queue队列数据的信息（当前队列数量，最大值，队列限制数量）

Session rate 每秒会话率（当前值，最大值，限制数量）

Session总会话量（当前值，最大值，总量，Lbtot: total number of times a server was selected选中一台服务器所用的总时间）

Bytes（入站、出站流量）；

Denied（拒绝请求、拒绝回应）；

Errors（错误请求、错误连接、错误回应）；

Warnings（重新尝试警告retry、重新连接redispatches）；

Server(状态、最后检查的时间（多久前执行的最后一次检查）、权重、备份服务器数量、down机服务器数量、down机时长)