## SinaHaproxy-1.4.19-v5.3版本发布文档

#### 背景信息

此次版本是基于haproxy官方的1.4.19源码进行开发，除了支持官方所有的功能外，合入SINA线上1.4.15的补丁，并添加了支持服务器自动上下线、忽略配置文件错误、优化rewrite功能等。旨在实现以下目标:

1. 服务器的自动上下线功能，也是自动化运维工作的一部分。将后端服务器上下线通过API接口开放，并实现此类变更不影响线上服务。
2. 容忍配置文件局部错误，不影响其它业务。线上Haproxy服务器往往承载了多个业务，并且这些业务方是跨部门的，为了在实施日常变更中，由于配置文件改错，导致其它业务受到影响，开发此功能。
3. 后端服务器响应时延统计

对于业务运维人员提出的流量经过七层负载均衡Haproxy后，延时增加的问题，在SinaHaproxy-1.4.19版本上增加了四项延时统计

* 连接时间：Haproxy与后端服务器的TCP连接建立时间
* 首包时间：从转发请求后收到后端响应首包时间
* 响应时长：从转发请求后，到收到完整的后端响应报文时长
* 七层处理延时：接收到完整的用户请求到连接后服务器这段时间

1. 七层规则的热加载

支持在 haproxy 不停服务的状态下,更新指定服务的七层规则。同时实现每个服务七层规则的隔离，即可以使每个服务的七层服务单独热加载，不影响其他服务。 同时提供七层规则校验工具，方便在七层规则的提交前，校验七层规则是否存在错误。同时将七层规则的加载情况反馈给七层规则管理系统。

1. 每次请求的时间消耗分析

方便线上事故的分析，需要统计客户端的连接时间，以及负载均衡连接后端服务器的时间，七层规则处理时间，后端服务器响应时间。 将这些信息放入http header 中，方便将来的流量分析业务。

1. 方便线上系统事故跟踪

在haproxy帮助信息中增加svn版本号和编译时间。

#### 新增特性

1. 支持后端服务器自动上下线功能，将backend的server列表从haproxy.cfg中移出，每个backend单独一个文件存放server列表，并实现自动加载，以及变更不丢连接。
2. 新增对侦听地址冲突检查的功能，对于在同一配置文件中出现同一IP地址+端口在配置文件中出现俩次及以上，使用Haproxy对配置文件正确性检查的时候，将会提示配置错误。
3. 添加了Haproxy在启动阶段忽略部分服务的配置错误，实现对单个服务变更不影响同一台负载均衡上的其它服务。如果配置错误发生在frontend中，那么该frontend的所有配置将不会生效；如果配置错误发生在backend中，那么所有用到该backend的frontend配置均不会生效。
4. 优化的reqrep的功能。在使用reqrep或者reqirep进行Rewrite的时候，在匹配请求头里的request line(使GET /index.html HTTP/1.0)时，只匹配request line里的路径部分（/index.html），方便线上运维配置。
5. 新增一条Unix socket command：show delay，用来输出时延统计信息，字段内容格式如下：

Frontend: Haproxy的协议分析、七层规则的处理以及Haproxy本身的处理逻辑导致的时延(七层延时)  
七层的处理延时（单位:ms）：

[0,3],[3,5],[5,10],[10,+∞]

Backend：连接后端服务器的延时，首包时间，响应时长。

以下是各个延时划分的区间段(单位：ms),Haproxy会统计各个区间段的数。

响应时长(单位：ms)：

[0,1000],[1000,2000],[2000,3000],[3000,5000],[5000,+∞]

连接时间(单位：ms)：

[0,1000],[1000,2000],[2000,+∞]

首包时间(单位：ms)：

[0,1000],[1000-3000],[3000,5000],[5000,+∞]

1. 在1.4.19版本里，如果请求头里有X-Forwarded-For字段，那么默认将不再插入X-Forwarded-For字段，即使在Backend里已经打开了option forwardfor开关，但如果在option forwardfor后面配置“if-always”的话（option forwardfor if-always），那么将不管是否包含X-Forwarded-For,都将插入X-Forwarded-For。
2. 每请求得时间消耗统计

在http 请求返回的header 中增加key为SINA-TS的行，其value数据采用base64编码。解码以后的数据格式为： “queue\_time connect\_time 4layer\_dely 7layer\_delay srv\_delay。 其中各个字段的含义为：

* queue\_time （单位：ms）

每个任务在队列中的等待时间

* connect\_time（单位：ms）

haproxy连接后端backendserver 所花费的时间，从accept 客户端接到向后端server连接建立成功花费的时间。

* 4layer\_delay（单位：ms）

4层处理消耗时间， 从向后端server发送connect 请求到 connect 成功所需的时间

* 7layer\_delay（单位:us）

处理七层规则所消耗的时间，因为时间比较短，结果现实的是微秒

* srv\_delay （单位：ms）

后端server处理时间，从向后端server发送数据开始，到收到后端 server返回的结果为止。

#### 合入功能

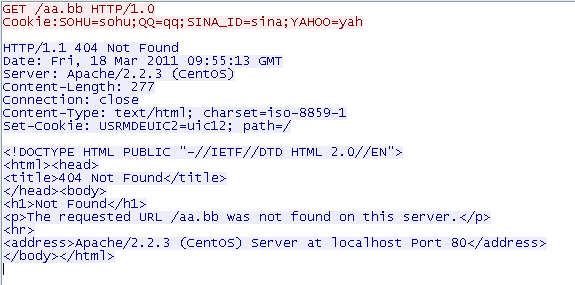
1. clear-cli-cookie cookie\_name [ if acl1 acl 2 or acl3]

这个命令只能在backend中使用。其中clear-cli-cookie用来在将请求提交给后端服务器之前，删除cookie\_name对应的名称和值。

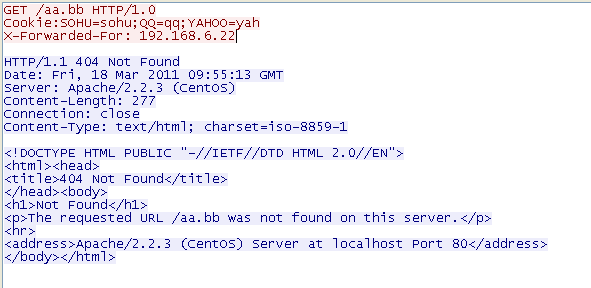
而clear-srv-cookie用来在将响应返回给客户端之前，删除cookie\_name及其value值。

具体效果见图示.在配置文件中，配置clear-cli-cookie SINA\_ID选项，用来删除SINA\_ID的cookie.以下是wireshark抓包截图。

图一、用户提交的请求



图二、提交给后台服务器请求



1. 七层规则关键字cook\_val

在官方版本中，在7层规则方面还不支持跟据cookie来选择后端服务器。只能跟据backend的cookie\_name来选取backend中的后端服务器。应服务部门需求，添加了根据cookie的值来选择backend

1. 导出4xx,5xx响应码统计详细信息，例如响应码500发生了多少…
2. 给每个Haproxy进程指定一个Unix:sock， 用于通过CLI导出每个进程的统计信息，具体路径：/var/run/haproxy\_stat\_ID
3. 在HTTP健康检查中，忽略后端服务器4XX响应码，当后端服务器返回403时，仍然认为后端服务器是正常的
4. 对于balance uri算法，新增一个辅助option hash-filename，来指定根据file name做哈希。
5. 在HTTP响应头里新增一个字段：SINA-LB，导出当前负载均衡的Hostname，方便故障定位排查
6. 新增option hdrname2xforward

option hdrname2xforward hdr-name

将hdr-name转成x-Forwarded-For，但必须通过option change2xforwardfor来打开该项功能。

#### 新增特性说明

##### 服务器自动上下线功能支持

新增了一个option server\_from\_file选项，该选项只能配在Backend或Listen里面。用于支持将池子里的服务器列表单独移出来，放到一个文件中去。服务器列表文件名默认与Backend的文件名相同，但也可以通过server\_file来指定Backend服务器列表对应的文件名。

文件内容格式如下：

#ip,port,weight,maxconn,check\_interval,fall

在后端池服务器文件中，每个服务一行，配置以上六个参数即可。

每个服务器对应的Cookie值，Haproxy会自动生成，由usrmdinst\_和该服务器在文件中的第几行生成。

**可能风险**：如果添加一个服务器，服务器在列表中的顺序发生变化，那么服务器对应的cookie值就会发生变化，导致后续做会话将会转发现不同的服务器上。

##### 地址冲突检查功能

支持通过配置文件的正确性检查工具发现配置文件中是否存在地址重复使用的问题；对于前后配置地址冲突以及Haproxy配置文件中的地址如果被系统其它进程占用的错误，Haproxy在加载配置过程中将忽略此类错误，但发生错误的Proxy配置将不会生效。如果没有服务生效，那么Haproxy将会退出。

##### 忽略局部配置错误

此功能主要解决在日常变更中，往往只针对单个服务进行变更，但同时该台负载均衡上有其它服务，为了解决对单个服务变更不影响同一台负载均衡上的其它服务，增加此项功能。如果配置错误发生在frontend中，那么该frontend的所有配置将不会生效；如果配置错误发生在backend中，那么所有用到该backend的frontend配置均不会生效。

##### 优化使用reqrsp进行URI Rewrite

官方版本在使用reqrsp对URI进行rewrite的时候，会匹配整个request line（GET /Path HTTP/1.0）,现改为只匹配URI（/Path）

#### 统计时延输出格式

Frontend格式：

Proxy-id，FRONTEND，七层处理时延{0-3ms,3ms-5ms,5ms-10ms,>10ms}

Backend格式：

Proxyid,BACKEND,四层连接时延{0-1,1-2,>3}，首包时间{0-1,1-3,3-5,>5}，响应时延{0-1s,1s-2s,2s-3s,3s-5s,>5s}

SERVER 格式：

Backend-id/ip:port,SERVER,四层连接时延{0-1s,1s-2s,>3s}，首包时间{0s-1s,1s-3s,3s-5s,>5s}，响应时延{0s-1s,1s-2s,2s-3s,3s-5s,>5s}

注：

时延输出顺序按上述顺序，每个时延输出项目按时延段依次输出。

##### 示例

1. 延时统计

[root@2643 haproxy-svn-1.4.19]# echo "show delay" | nc -U /var/run/haproxy.stats\_1

monitor,FRONTEND,0,0,0,0

test1\_com,FRONTEND,5000,0,0,0

test2\_com,FRONTEND,0,0,0,0

pool\_weibo\_com\_zt\_aj,BACKEND,5000,0,0,0,4881,53,66,0,0,4881,53,66

pool\_weibo\_com\_zt\_aj/192.168.5.71:9080,SERVER,0,0,0,0,0,0,0,0,0,0,0,0

pool\_weibo\_com\_zt\_aj/192.168.5.71:9081,SERVER,1667,0,0,0,1624,16,27,0,0,1624,16,27

pool\_weibo\_com\_zt\_aj/192.168.5.71:9082,SERVER,1667,0,0,0,1624,18,25,0,0,1624,18,25

pool\_weibo\_com\_zt\_aj/192.168.5.71:9083,SERVER,1666,0,0,0,1633,19,14,0,0,1633,19,14

pool\_weibo\_com\_zt\_s,BACKEND,0,0,0,0,0,0,0,0,0,0,0,0

pool\_weibo\_com\_zt\_s/192.168.5.71:80,SERVER,0,0,0,0,0,0,0,0,0,0,0,0

cookie\_test,FRONTEND,0,0,0,0

v1,BACKEND,0,0,0,0,0,0,0,0,0,0,0,0

v1/192.168.5.71:9000,SERVER,0,0,0,0,0,0,0,0,0,0,0,0

v2,BACKEND,0,0,0,0,0,0,0,0,0,0,0,0

v2/192.168.5.71:9001,SERVER,0,0,0,0,0,0,0,0,0,0,0,0

1. 每请求时间消耗

向haproxy发送请求，返回如下结果

[root@168 sixing]# curl -I http://192.168.5.53:9000/index.html

HTTP/1.1 200 OK

Server: nginx/1.1.17

Date: Fri, 15 Mar 2013 08:50:02 GMT

Content-Type: text/html

Content-Length: 151

Last-Modified: Fri, 01 Feb 2013 07:46:04 GMT

Connection: close

Accept-Ranges: bytes

SINA-LB:bXU1Mw==

SINA-TS:YzRlNzdiMDQgMCAwIDAgMTkgMAo=

其中最后一行为时间消耗统计数据，反编码以后的数据格式如下： “server\_ip（网络字节序与0xf1e2d3c4异或运算） queue\_time connect\_time 4layer\_dely 7layer\_delay srv\_delay对应的的数据为：c4e77b04 0 0 0 19 0

##### 源代码路径

见svn链接：

<https://svn.intra.sina.com.cn/sysdev/NetworkDevTeam/SHAPEE/Haproxy/tags/haproxy-1.4.19-v5.3>

#### 注意事项

**使用rpm包部署Haproxy时，务必备份Haproxy的配置文件！！！**