Panblack

Linux, http://weibo.com/panblack

haproxy/nginx+keepalived负载均衡 双机热备 邮件报警 实战及常见问题

Haproxy 做http和tcp反向代理和负载均衡
keepalived 为两台 Haproxy 服务器做高可用/主备切换。
nginx 为内网服务器做正向代理，如果业务需求有变化，也可以部分替代 haproxy 做 http 反向代理。
如果发生主备切换，向指定邮箱发送报警邮件。

本文比较裹脚布，没耐心的就别看了。

一、两台服务器，系统 CentOS6

主机名	外网IP	内网IP
lbserver_01	202.1.1.101	10.1.1.11/24
lbserver_02	202.1.1.102	10.1.1.12/24
虚拟IP	202.1.1.97	10.1.1.10/24
虚拟IP	202.1.1.98	
虚拟IP	202.1.1.99	

以下示例仅提供lbserver_01的配置，lbserver_02上大部分相同，不同之处会单独注明。

二、安装软件包

1、配置 nginx 官方软件源

```
[root@lbserver_01 ~]# vi /etc/yum.repos.d/nginx.repo
[nginx]
name=nginx repo
baseurl=http://nginx.org/packages/centos/6/x86_64/
gpgcheck=0
enabled=1
```

2、常用工具

```
[root@lbserver_01 ~]# yum -y install telnet man vim wget
zip unzip ntpdate tree gcc iptraf tcpdump  bind-utils
```

3、服务器软件

```
[root@lbserver_01 ~]# yum -y install haproxy keepalived
nginx sendmail mailx
```

Linux(23)
Oracle(2)
python(1)
虚拟化(2)
英语(1)

随笔档案(26)
2019年1月 (1)
2016年3月 (1)
2015年10月 (2)
2015年8月 (1)
2015年7月 (2)
2015年4月 (1)
2014年12月 (2)
2014年10月 (3)
2014年9月 (1)
2014年7月 (1)
2014年3月 (1)
2014年2月 (1)
2013年11月 (1)
2013年5月 (1)
2013年4月 (1)
2013年3月 (1)
2013年2月 (1)
2012年12月 (1)
2012年9月 (1)
2012年8月 (2)

最新评论
1. Re:ssh访问控制，多次失败登录即封掉IP，防止暴力破解
@张玉亭多谢，刚刚改了 ;)...
--Panblack
2. Re:Centos7添加静态路由
你好 我现在 加了route-eth0 还是没有用呢！[root@CDS-YOUZU-SH-002 network-scripts]# cat route-eth0 47.100.65.1/32 .....
--yshuluwa
3. Re:ssh访问控制，多次失败登录即封掉IP，防止暴力破解
第四个测试的时候应该是打开black.list
cat /usr/local/bin/black.txt
--张玉亭
4. Re:五款app原型设计工具对比

4、配置服务

```
[root@lbserver_01 ~]# chkconfig dnsmaq on
[root@lbserver_01 ~]# chkconfig sendmail on
[root@lbserver_01 ~]# chkconfig keepalived on
[root@lbserver_01 ~]# chkconfig haproxy on
```

sendmail 负责发送邮件，dnsmasq为内网提供 DNS 服务。

内网服务器如果需要上网（主要是为了安装软件包），需配置 DNS 和 http_proxy 环境变量，也可同时为 yum 配置proxy参数。

```
[root@lbserver_01 ~]# vim /etc/resolv.conf
```

```
nameserver 10.1.1.10
```

```
[root@lbserver_01 ~]# vim /etc/profile
```

```
export http_proxy=http://10.1.1.10:8000
```

```
[root@lbserver_01 ~]# vim /etc/yum.conf
```

```
proxy=http://10.1.1.10:8000
```

10.1.1.10:8000 这个 IP 和端口在后面的 nginx 配置文件中定义。

三、系统配置文件

1、配置命令行提示符，显示当前完整路径

```
[root@lbserver_01 ~]# vim /root/.bash_profile
PS1='[\u@\h:$PWD]# '

[root@lbserver_01 ~]# source /root/.bash_profile
[root@lbserver_01:/root]#
```

为新建用户自动配置提示符

```
[root@lbserver_01:/root]# vim /etc/skel/.bash_profile
PS1='[\u@\h:$PWD]$ '
```

2、配置报警邮件发送帐号

```
[root@lbserver_01:/root]# vim /etc/mail.rc
set from=SendAlert@youdomain.com smtp=smtp.youdomain.com
set smtp-auth-user=SendAlert@youdomain.com smtp-auth-password=PassForSendAlert smtp-auth=login
```

3、配置 dnsmasq 仅为内网服务，很简单的替换，用 sed 吧

同事给介绍的Mockplus，非常赞的国产工具，个人觉得不比国外那些差，而且在很多方面做的更好!

--cindy2

5. Re:增加ssh无密码信任连接的安全性

学习了

--安迪老大

阅读排行榜

1. Centos7添加静态路由(41060)
2. Centos7系统配置上的变化（二）网络管理基础(31185)
3. WebVirtMgr - A pretty good kvm web-based management tool(14102)
4. Centos7系统配置上的变化（一）(14042)
5. 五款app原型设计工具对比(11350)

评论排行榜

1. Ubuntu配置adsl + squid + iptables代理服务器(8)
2. Centos6安装oracle10g(7)
3. haproxy/nginx+keepalived负载均衡 双机热备 邮件报警 实战及常见问题(5)
4. Centos7系统配置上的变化（一）(4)
5. WebVirtMgr - A pretty good kvm web-based management tool(4)

推荐排行榜

1. 增加ssh无密码信任连接的安全性(4)
2. Centos7系统配置上的变化（二）网络管理基础(3)
3. 正则表达式备忘录-Regular Expressions Cheatsheet中文版(2)
4. Centos7添加静态路由(2)
5. Centos7系统配置上的变化（三）为网络接口添加多IP(1)

```
[root@lbserver_01:/root]# sed 's/#interface=/interface=eth1/g' /etc/dnsmasq.conf -i
```

4、limits

```
[root@lbserver_01:/root]# vim /etc/security/limits.conf
```

```
*          soft    nofile          1024
*          hard    nofile          65536
*          soft    nproc           2048
*          hard    nproc           16384
*          soft    stack           10240
*          hard    stack           32768
```

5、配置haproxy keepalived 日志

以下两个文件需要修改参数：

```
[root@lbserver_01:/root]# vim /etc/sysconfig/keepalived
```

```
KEEPALIVED_OPTIONS="-d -D -S 0"
```

```
[root@lbserver_01:/root]# vim /etc/sysconfig/rsyslog
```

```
SYSLOGD_OPTIONS="-r -c 2"
```

以下文件需要添加两行：

```
[root@lbserver_01:/root]# vim /etc/rsyslog.conf
```

```
local0.*    /var/log/keepalived.log
local2.*    /var/log/haproxy.log
```

keepalived日志正常，而haproxy日志目前还没有生成，原因慢慢找吧。

6、防火墙

```
[root@lbserver_01:/root]# vim /etc/sysconfig/iptables
```

```
# Firewall configuration written by system-config-firewall
# Manual customization of this file is not recommended.
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
```

```
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 2888 -j
ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j
ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 8080 -j
ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 443 -j
ACCEPT
-A INPUT -m state --state NEW -s 10.1.1.0/24 -m tcp -p tcp --
dport 8000 -j ACCEPT
-A INPUT -m state --state NEW -s 10.1.1.0/24 -m tcp -p tcp --
dport 53 -j ACCEPT
-A INPUT -m state --state NEW -s 10.1.1.0/24 -m udp -p udp --
dport 53 -j ACCEPT
-A INPUT -d 224.0.0.18 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

- tcp 2888 是修改后的 sshd 端口，替代默认的 22 ，可以很大程度上防止无聊的小白来暴力破解。
- tcp 80 8080 443 负载均衡对外开放的端口
- tcp 8000 仅为内网服务，是 nginx 对内正向代理
- tcp/udp 53 仅为内网服务，提供 DNS 解析转发
- **-A INPUT -d 224.0.0.18 -j ACCEPT** 这一条非常关键。224.0.0.18 是 keepalived 的组播地址，确保当前主机组播自己的状态。缺少这条指令会产生很诡异的问题，比如双机网卡上都有虚拟IP地址，但主机是有效的；而备机接管虚拟 IP 后说啥也不还给主机了。

7、nginx正向代理

[root@lbserver_01:/root]# vim /etc/nginx/conf.d/default.conf

```
server {
    listen       8000;
    server_name  localhost;
    resolver     10.1.1.10;
    #charset koi8-r;
    access_log  /var/log/nginx/default.access.log  main;

    location / {
     proxy_pass http://$http_host$request_uri;
    }
```

配置文件其他部分保持默认就可以了。

8、keepalived配置

- keepalived是双机热备的首选。平时主机获得虚拟IP，提供服务，备机在旁边看着。
- 如果主机DOWN、网络断、服务出现故障，则备机立刻接管虚拟IP。
- 当主机恢复时，又会自动抢回主机地位，备机接着看。

[root@lbserver_01:/root]# vim /etc/keepalived/keepalived.conf

```
! Configuration File for keepalived

global_defs {

   router_id HAP_KAD1

! 默认配置文件里本段还有notification_email设置，我不用，所以删掉了。
}


! vrrp_script 设置检查服务健康状况的脚本，脚本内容见下文。
! interval 1 每秒运行一次
! weight -10 如果脚本运行结果不为0（即失败或无结果），即说明主机haproxy服务故障，priority减10，此时备机priority比主机大，将自动选举为主机，原主机降为备机。
! priority 范围是1到254。为了避免priority不断降低最后两机都变为1，理论上应该再配置vrrp_script使恢复正常的服务器提升priority，提升脚本需要配 weight <正数>，但实际实验中发现基本没有这个必要。如果生产中两机不断的竞相自降 priority，说明有大麻烦发生了。

! 检查haproxy的脚本
vrrp_script check-haproxy {
    script "/usr/local/bin/chk-haproxy.sh"
    interval 1
    weight -10
}

! 检查nginx的脚本
vrrp_script check-nginx {
    script "/usr/local/bin/chk-nginx.sh"
    interval 1
    weight -10
}
```

```
vrrp_instance haproxy {

    ! 不设 MASTER，双机都是 BACKUP，只根据 priority选举主机地位。
    ! 备机 lbserver_02 的 priority 为 220

    state BACKUP
    interface eth0
    virtual_router_id 43
    priority 225

    advert_int 1
    authentication {
        auth_type PASS
        auth_pass PasswordForAuth
    }
    virtual_ipaddress {
    202.1.1.97/24
    202.1.1.98/24
    202.1.1.99/24
    }
    track_script {
        check-haproxy
    }

    ! 如果本机变为主机，则运行指定的脚本，脚本内容是使用 /etc/mail.rc 中配置的发信帐号向指定的接收邮箱发邮件。
    notify_master "/usr/local/bin/haproxy-master-change.sh"
}

! 为内部正向代理设置主备切换。其实没有必要，但是如果 nginx 要配置 http 反向代理，这部分还是必须的。
! 注意这里的 interface, virtual_router_id 跟上面vrrp_instance haproxy 的是不同的。
! virtual_ipaddress 10.1.1.10/24 ，还记得这个IP么?

vrrp_instance internalproxy {
    state BACKUP
    interface eth1
    virtual_router_id 80
    priority 225
    advert_int 1
    authentication {
        auth_type PASS
        auth_pass PasswordForAuth
    }

    virtual_ipaddress {
        10.1.1.10/24
    }
    track_script {
```

```
        check-nginx
    }
    notify_master "/usr/local/bin/nginx-master-change.sh"
}
```

9、haproxy 配置文件

[root@lbserver_01:/root]# vim /etc/haproxy/haproxy.cfg

```
#---------------------------------------------------------------------
# Example configuration for a possible web application.  See the
# full configuration options online.
#
#   http://haproxy.1wt.eu/download/1.4/doc/configuration.txt
#
#---------------------------------------------------------------------

#---------------------------------------------------------------------
# Global settings
#---------------------------------------------------------------------
global
    # to have these messages end up in /var/log/haproxy.log you will
    # need to:
    #
    # 1) configure syslog to accept network log events.  This is done
    #    by adding the '-r' option to the SYSLOGD_OPTIONS in
    #    /etc/sysconfig/syslog
    #
    # 2) configure local2 events to go to the /var/log/haproxy.log
    #   file. A line like the following can be added to
    #   /etc/sysconfig/syslog
    #
    #    local2.*                       /var/log/haproxy.log

    #这一条配合/etc/rsyslog.conf
    log         127.0.0.1 local2

    chroot      /var/lib/haproxy
    pidfile     /var/run/haproxy.pid
```

```
    #自定义最大连接数
    maxconn     10240

    user        haproxy
    group       haproxy
    daemon

    # turn on stats unix socket
    stats socket /var/lib/haproxy/stats

    #默认值是1024，haproxy启动时会报错，但并不影响服务。改成2048后据说会有一点儿性能问题。
    tune.ssl.default-dh-param 2048

#---------------------------------------------------------------------
# common defaults that all the 'listen' and 'backend' sections will
# use if not designated in their block
#---------------------------------------------------------------------
defaults
    mode                    http
    log                     global
    option                  httplog
    option                  dontlognull
    option http-server-close
    option                  redispatch
    retries                 3
    timeout http-request    10s
    timeout queue           1m
    timeout connect         10s
    timeout client          1m
    timeout server          1m
    timeout http-keep-alive 10s
    timeout check           10s
    maxconn                 8192

#---------------------------------------------------------------------
# main frontend which proxys to the backends
#---------------------------------------------------------------------

# https网站发布，server.pem 是crt文件和key文件合并而成：cat server.key server.crt| tee server.pem
# http网站发布除了 bind *:80 和无需指定服务器证书之外，其他配置相同
# 另外，haproxy 跟 keepalived 配合时，bind 后面只能用 * 。如果指定了 IP，备机haproxy将无法启动，因为平时状态备机是不具备虚拟IP地址的。
```

```
frontend ssl
    mode http
    bind *:443  ssl crt /etc/haproxy/server.pem
    option httplog
    option httpclose

    # 这条可以使后端web服务器获得用户真实 IP，在httpd，tomcat 等日志格式配置里，用 %{X-Forwarded-For}i 记录用户真实 IP 。
    option forwardfor   except 127.0.0.0/8

    stats  hide-version

    #根据域名指定后端服务器
    #    acl名称                        条件               域名（-i 忽略大小写）
    acl domain_starts_with_www      hdr_beg(host)    -i www.example.com
    acl domain_starts_with_auth     hdr_beg(host)    -i auth.example.com
    acl domain_starts_with_shops    hdr_beg(host)    -i shops.example.com

    #这里千万不能单独指定 www 的backend，那样会使所有的url规则失效。www 只能作为default后端
    use_backend auth    if domain_starts_with_auth
    use_backend shops   if domain_starts_with_shops

    #根据域名后的路径指定后端服务器
    acl url_users       path_beg    -i /users
    acl url_topsales    path_beg    -i /topsales

    #if 后面多个 acl 时，and用空格代替， or 必须要写
    use_backend users    if domain_starts_with_www  url_users
    use_backend topsales if domain_starts_with_www url_topsales

    #其他不符和acl条件的统统指定给后端 www
    default_backend        www

backend www
    #需要保持会话session的必须用source算法，即同一个外网IP的用户始终访问同一台后端。
    balance    source

    # haproxy 设置了服务器证书，后端真实web服务器用普通http就可以了
    server    httpd_tomcat1 10.1.1.20:80 check
    server    httpd_tomcat2 10.1.1.21:80 check
    server    httpd_tomcat1 10.1.1.22:80 check
    server    httpd_tomcat2 10.1.1.23:80 check
```

```
backend auth
    #auth 无需保持会话，所以用随机负载均衡算法 roundrobin 即可。
    balance    roundrobin
    server    httpd_tomcat_auth 10.1.1.30:80 check
    server    httpd_tomcat_auth 10.1.1.31:80 check

backend shops
    balance    source
    server    httpd_tomcat_auth 10.1.1.40:80 check
    server    httpd_tomcat_auth 10.1.1.41:80 check
    server    httpd_tomcat_auth 10.1.1.42:80 check
    server    httpd_tomcat_auth 10.1.1.43:80 check

backend users
    balance    source
    server    httpd_tomcat_auth 10.1.1.50:80 check
    server    httpd_tomcat_auth 10.1.1.51:80 check

backend topsales
    balance    source
    server    httpd_tomcat_auth 10.1.1.60:80 check
    server    httpd_tomcat_auth 10.1.1.61:80 check

#haproxy 还支持tcp反向代理
frontend ssh *:3306
    mode        tcp
    maxconn     128
    option      tcplog
    default_backend mysql

backend mysql
    mode        tcp
    balance     source
    server      mysql1    10.1.1.200:3306 check

#End
```

四、各种辅助脚本

1、服务健康状况检查脚本

`/usr/local/bin/chk-haproxy.sh`

```
#!/bin/bash
ps aux|grep "/usr/sbin/haproxy"|grep -v grep
```

/usr/local/bin/chk-nginx.sh

```
#!/bin/bash
ps aux|grep nginx|egrep '(master|worker)'
```

最初的chk-haproxy.sh是这样的：
netstat -lntp|egrep '(.*443.*haproxy|.*3306.*haproxy)'

实验时工作的不错，等配好了后端服务器做压力测试，却突然连续收到master-change邮件。
用top一看，主机的 netstat 进程 CPU占用率都超过 haproxy 了，达到20%+。估计是 chk-haproxy.sh 脚本发生运行失败导致keepalived进行了主备切换，切换后主机chk-haproxy.sh脚本恢复正常，备机chk-haproxy.sh脚本可能过载，所以很快又收到master-change邮件。用 ps aux 替换 netstat 就没有问题了。

2、邮件通知脚本
/usr/local/bin/haproxy-master-change.sh
#!/bin/bash

```
echo "`uptime; ip addr show eth0; echo`" | mail -s "`hostname -s` to HAPROXY master."  -c supervisor@example.com receiver-01@example.com receiver-02@example.com
```

/usr/local/bin/nginx-master-change.sh

```
#!/bin/bash
echo "`uptime; ip addr show eth1; echo`" | mail -s "`hostname -s` to NGINX master." receiver-03@example.com receiver-04@example.com
```

echo "...." 邮件正文，含主机名和 IP 地址，
-s 邮件标题
-c 抄送地址（必须写在收件人前面）
最后可以跟 n 个收件人地址

3、服务器状态查看脚本
捕获 keepalived 组播数据，可以看到当前主机的 priority （prio 225）。
/usr/local/bin/kawatch.sh

```
#!/bin/bash
tcpdump -vvv -n -i eth0 dst 224.0.0.18
```

运行起来是这样的：

```
tcpdump: listening on eth0, link-type EN10MB (Ethernet),
capture size 65535 bytes
23:10:14.066857 IP (tos 0xc0, ttl 255, id 7007, offset 0,
flags [none], proto VRRP (112), length 52)
    202.1.1.101 > 224.0.0.18: VRRPv2, Advertisement, vrid 43,
prio 225, authtype simple, intvl 1s, length 32, addrs(3):
202.1.1.97,202.1.1.98,202.1.1.99 auth "PasswordForAuth"
23:10:15.068072 IP (tos 0xc0, ttl 255, id 7008, offset 0,
flags [none], proto VRRP (112), length 52)
    202.1.1.101 > 224.0.0.18: VRRPv2, Advertisement, vrid 43,
prio 225, authtype simple, intvl 1s, length 32, addrs(3):
202.1.1.97,202.1.1.98,202.1.1.99 auth "PasswordForAuth"
23:10:16.069224 IP (tos 0xc0, ttl 255, id 7009, offset 0,
flags [none], proto VRRP (112), length 52)
    202.1.1.101 > 224.0.0.18: VRRPv2, Advertisement, vrid 43,
prio 225, authtype simple, intvl 1s, length 32, addrs(3):
202.1.1.97,202.1.1.98,202.1.1.99 auth "PasswordForAuth"
......
```

以下是常用的 netstat 指令，总用懒得敲那么多字母，就精简一下吧：

/usr/local/bin/nsl

```
#!/bin/bash
netstat -lntp
```

/usr/local/bin/nsa

```
#!/bin/bash
netstat -antp
```

/usr/local/bin/nse

```
#!/bin/bash
netstat -antp|grep ESTABLISHED
```

主机IP信息

```
[root@lbserver_01:/root]# ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state
UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
```

```
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
4: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast state UP qlen 1000
    link/ether fa:b1:3b:ec:b4:c2 brd ff:ff:ff:ff:ff:ff
    inet 202.1.1.101/24 brd 202.1.1.255 scope global eth0
    inet 202.1.1.97/24 scope global secondary eth0
    inet 202.1.1.98/24 scope global secondary eth0
    inet 202.1.1.99/24 scope global secondary eth0
    inet6 fe80::f0a1:8bfe:feda:c532/64 scope link
       valid_lft forever preferred_lft forever
5: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
pfifo_fast state UP qlen 1000
    link/ether fa:b1:3b:3c:01:0e brd ff:ff:ff:ff:ff:ff
    inet 10.1.1.11/24 brd 10.1.1.255 scope global eth1
    inet 10.1.1.10/24 scope global secondary eth1
    inet6 fe80::f001:aff:fb3c:12e/64 scope link
       valid_lft forever preferred_lft forever
```

分类: Linux

标签: haproxy, nginx, keepalived, mailalert

Panblack
关注 - 3
粉丝 - 22
+加关注

1 0

« 上一篇：ezdpl Linux自动化部署实战
» 下一篇：正则表达式备忘录-Regular Expressions Cheatsheet中文版

posted @ 2015-10-31 23:39 Panblack 阅读(4624) 评论(5) 编辑 收藏

评论列表

#1楼[楼主] 2015-11-03 14:54 Panblack

重要堪误！！！

https网站发布，server.pem 是crt文件和key文件合并而成：

cat server.key server.crt | tee server.pem

最开始顺序写反了，抱歉！应该是key在前，crt 在后。正文已修改为正确的，见红色粗体字。

支持(0) 反对(0)

#2楼[楼主] 2015-11-03 16:52 Panblack

如果证书含End Entity Certificate， Primary Intermediate Certificate，Secondary Intermediate Certificate，按照 KEY, EEC, SID, PID的顺序合并pem证书，比如：

cat Server.key EndEntity.crt SecondIntermediate.crt PrimaryIntermediate.crt |tee server.pem

支持(0) 反对(0)

#3楼[楼主] 2016-06-17 15:19 Panblack

haproxy 日志解决方法：

去掉 /etc/rsyslog.conf 文件里下面两行的注释
$ModLoad imudp
$UDPServerRun 514

支持(0) 反对(0)

#4楼[楼主] 2017-09-08 15:40 Panblack

配置多个SSL证书难住了很久，终于翻了 stackoverflow（好多帖子号称解决了但并不准确） 和 官方文档后找到办法了。

```
frontend https
bind *:443 ssl crt /etc/haproxy/site1.pem crt /etc/haproxy/site2.pem
stats hide-version

acl domain_www_site1_com hdr_beg(host) -i www.site1.com
acl sni_www_site1_com ssl_fc_sni -i www.site1.com
use_backend site1 if domain_www_site1_com sni_www_site1_com

# 适合 wildcard 证书，比如 *.site2.com
acl domain_www_site2_com hdr_beg(host) -i www.site2.com
acl domain_adm_site2_com hdr_beg(host) -i adm.site2.com
acl domain_usr_site2_com hdr_beg(host) -i usr.site2.com
acl sni_site2_com ssl_fc_sni_end -i .site2.com
use_backend www_site2 if domain_www_site2_com sni_site2_com
use_backend adm_site2 if domain_adm_site2_com sni_site2_com
use_backend usr_site2 if domain_usr_site2_com sni_site2_com

# 同样可以:
#use_backend site1 if domain_www_site1_com { ssl_fc_sni www.site1.com }
#use_backend www_site2 if domain_www_site2_com { ssl_fc_sni www.site2.com }
```

```
#use_backend adm_site2 if domain_adm_site2_com { ssl_fc_sni adm.site2.com }
#use_backend usr_site2 if domain_usr_site2_com { ssl_fc_sni usr.site2.com }

backend site1
......

backend www_site2
......

backend adm_site2
......

backend usr_site2
......
```

支持(0) 反对(0)

#5楼[楼主] 2017-09-08 15:44 Panblack

http://cbonte.github.io/haproxy-dconv/1.5/configuration.html#7.3.4-ssl_fc_sni

支持(0) 反对(0)

刷新评论 刷新页面 返回顶部

注册用户登录后才能发表评论，请 登录 或 注册，访问网站首页。

相关博文：

· Nginx+Keepalived双机热备
· Nginx+Keepalived 实现双击热备及负载均衡
· Nginx+keepalived做双机热备加tomcat负载均衡
· keepalived+nginx双机热备+负载均衡
· Nginx+Keepalived(双机热备)搭建高可用负载均衡环境(HA)