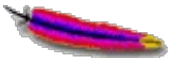
Table Of Contents

nixCraft: Linux Tips, Hacks, Tutorials, And Ideas In Blog Format
http://www.cyberciti.biz/

Linux install and configure pound reverse proxy for Apache http / https web server

Posted by Vivek Gite <vivek@nixcraft.com>

Q. How do I install and configure pound reverse proxy for Apache web sever under Debian Linux?  [1]

A. Pound is a reverse-proxy load balancing server. It accepts requests from HTTP / HTTPS clients and distributes them to one or more Web servers. The HTTPS requests are decrypted and passed to the back-ends as plain HTTP. It will act as:
a) Server load balancer
b) Reverse proxy server
c) Apache reverse proxy etc
d) It can detects when a backend server fails or recovers, and bases its load balancing decisions on this information: if a backend server fails, it will not receive requests until it recovers
e) It can decrypts https requests to http ones
f) Rejects incorrect requests
h) It can be used in a chroot environment (security feature)

If more than one back-end server is defined, Pound chooses one of them randomly, based on defined priorities. By default, Pound keeps track of associations between clients and back-end servers (sessions).

Install Pound Software

Type the following command to install pound:

```
$ sudo apt-get install pound
```

If you are using RHEL / CentOS, grab pound rpm here and type the command:

```
# rpm -ivh pound*
```

If you are using FreeBSD, enter:

```
# cd /usr/ports/www/pound/ && make install clean
```

How it works?

- Let us assume your public IP address 202.54.1.5.
- Pound will run on 202.54.1.5 port 80
- It will forward all incoming http requests to internal host 192.168.1.5 and 192.168.1.10 port 80 or 443
- Pound keeps track of associations between clients and back-end servers

Pound configuration file

- Under Debian / Ubuntu default file located at **/etc/pound/pound.cfg**
- Under FreeBSD it is located at **/usr/local/etc/pound.cfg** (you need to create this file)
- Under RHEL / CentOS you need to create file at **/etc/pound.cfg**

Sample configuration: HTTP Proxy

Forward all incoming request at 202.54.1.5 port 80 request to 192.168.1.5 Apache server running at 8080 port:
Open /etc/pound/pound.cfg file:

```
# vi /etc/pound/pound.cfg
```

To translate HTTP requests to a local internal HTTP server, enter (make sure 192.168.1.5 Apache running listing on port 8080):

```
ListenHTTP
         Address 202.54.1.5
         Port    80
         Service
                  BackEnd
                       Address 192.168.1.5
                       Port    8080
                  End
          End
End
```

Save and close the file. Restart pound:

```
# /etc/init.d/pound restart
```

Following example will distribute the all HTTP/HTTPS requests to two Web servers:

```
ListenHTTP
          Address 202.54.1.5
          Port    80
End

ListenHTTPS
         Address 202.54.1.5
         Port    443
         Cert    "/etc/ssl/local.server.pem"
End
Service
                  BackEnd
                      Address 192.168.1.5
                      Port    80
                  End
                  BackEnd
                      Address 192.168.1.6
                      Port    80
                  End
End
```

For testing purpose you may generate self signed ssl certificate (/etc/ssl/local.server.pem), by entering the following command:

```
# cd /etc/ssl && openssl req -x509 -newkey rsa:1024 -keyout local.server.pem -out
local.server.pem -days 365 -nodes
```

Pound log file

By default pound log message using syslog:

```
# tail -f /var/log/messages
# grep pound /var/log/messages
```

Sample complete configuration file

```
## Minimal sample pound.cfg
######################################################################
## global options:
User  "www-data"
Group  "www-data"
#RootJail "/chroot/pound"
## Logging: (goes to syslog by default)
## 0 no logging
## 1 normal
## 2 extended
## 3 Apache-style (common log format)
LogLevel 1
## check backend every X secs:
Alive  30
```

```
## use hardware-accelleration card supported by openssl(1):
#SSLEngine ""

######################################################################
## listen, redirect and ... to:
# Here is a more complex example: assume your static images (GIF/JPEG) are to be served fro
#       addition,  192.168.0.11  is  to  do  the  hosting for www.myserver.com with URL-bas
#       192.168.0.21 (800Mhz Duron) are for all other requests (cookie-based sessions).  Th
#       The configuration file may look like this:
             # Main listening ports
             ListenHTTP
                 Address 202.54.1.10
                 Port    80
                 Client  10
             End
             ListenHTTPS
                 Address 202.54.1.10
                 Port    443
                 Cert    "/etc/pound/pound.pem"
                 Client  20
             End

             # Image server
             Service
                 URL ".*.(jpg|gif)"
                 BackEnd
                     Address 192.168.1.10
                     Port    80
                 End
             End
            # Virtual host www.myserver.com
             Service
                 URL         ".*sessid=.*"
                 HeadRequire "Host:.*www.nixcraft.com.*"
                 BackEnd
                     Address 192.168.1.11
                     Port    80
                 End
                 Session
                     Type    PARM
                     ID      "sessid"
                     TTL     120
                 End
             End

             # Everybody else
             Service
                 BackEnd
                     Address 192.168.1.20
                     Port    80
                     Priority 5
                 End
                 BackEnd
                     Address 192.168.1.21
                     Port    80
                     Priority 4
                 End
                 Session
                     Type    COOKIE
                     ID      "userid"
                     TTL     180
                 End
             End
```

Suggested readings:

=> Pound project [2]
=> Man pages : pound and poundctl

Article printed from Frequently Asked Questions About Linux / UNIX: **http://www.cyberciti.biz/faq/**

URL to article: **http://www.cyberciti.biz/faq/linux-http-https-reverse-proxy-load-balancer/**

URLs in this post:

[1] Image: **http://www.cyberciti.biz/faq/faq/category/apache/**
[2] Pound project: **http://www.apsis.ch/pound/**