

# HAPROXY

## Introducción

En este documento detallaremos el proceso de instalación y configuración del servicio de balanceo de peticiones HAProxy. Hay que tener en cuenta que no siempre es recomendable su uso, ya que existen otros servicios de balanceo que pueden ser más efectivos según en que entornos (IPVSADM, nginx....)

## Enlaces interesantes

Configuración para mucho tráfico:

<https://medium.freecodecamp.org/how-we-fine-tuned-haproxy-to-achieve-2-000-000-concurrent-ssl-connections-d>

HAProxy Transparent mode:

<https://armaganyaman.blogspot.com/2015/04/haproxy-transparent-mode-on-centos-7.html>

## Instalación de HAProxy

Para la instalación de HAProxy, necesitamos realizar la descarga de los ficheros de source, para descomprimirlos, compilarlos y, posteriormente, instalarlos.

Nos descargamos el tar.gz de ficheros de source del HAProxy del siguiente enlace:

<http://www.haproxy.org/download/<version>/src/haproxy-<version>.tar.gz>

## Compilación

```
cd /install
```

```
wget http://www.haproxy.org/download/1.9/src/haproxy-1.9.6.tar.gz
```

A continuación descomprimos el archivo en /usr/local/src (o en otra ruta si fuera más pertinente):

```
cd /usr/local/src
```

```
tar -zxf /install/haproxy-1.9.6.tar.gz
```

Una vez descomprimido entramos en la carpeta y compilamos el paquete. Con tal de compilar, será necesario instalar los siguientes paquetes:

- gcc
- pcre-static
- pcre-devel
- systemd-devel
- openssl-devel

El comando a usar será el siguiente:

```
yum install gcc pcre-static pcre-devel systemd-devel openssl-devel
```

Una vez instalados los paquetes necesarios, empezaremos con la compilación. La línea que debemos usar es la siguiente:

```
cd /usr/local/src/haproxy-1.9.6  
make -j 4 TARGET=linux2628 USE_NS=1 USE_TFO=1 USE_OPENSSL=1 USE_ZLIB=1 USE_PCRE=1 USE_SYSTEMD=  
make install
```

Una vez instalado el servicio, añadiremos los scripts de arranque del servicio. En este caso la versión de Systemd.

/etc/systemd/system/haproxy.service

```
[Unit]  
Description=HAProxy Load Balancer  
After=network.target
```

```
[Service]  
Environment="CONFIG=/etc/haproxy/haproxy.cfg" "PIDFILE=/run/haproxy.pid"  
ExecStartPre=/usr/local/sbin/haproxy -f $CONFIG -c -q  
ExecStart=/usr/local/sbin/haproxy -Ws -f $CONFIG -p $PIDFILE  
ExecReload=/usr/local/sbin/haproxy -f $CONFIG -c -q  
ExecReload=/bin/kill -USR2 $MAINPID  
KillMode=mixed  
Restart=always  
SuccessExitStatus=143  
Type=notify
```

```
[Install]  
WantedBy=multi-user.target
```

systemctl daemon-reload

Finalmente, ya podemos empezar con la configuración inicial del servicio de HAProxy.

# Configuración de HAProxy

Para configurar correctamente el servicio de Balanceo de peticiones HAProxy, simplemente hay que crear un fichero con la configuración deseada en `/etc/haproxy/haproxy.cfg` (puede ser en otra ruta, pero esta es la más típica):

`/etc/haproxy/haproxy.cfg`

```
global
  log /dev/log    local0
  log /dev/log    local1 notice
  maxconn 2000000
  chroot /var/lib/haproxy
  stats socket /run/haproxy/admin.sock mode 660 level admin
  stats timeout 30s
  user haproxy
  group haproxy
  daemon
  nbproc 4
  cpu-map 1 0
  cpu-map 2 1
  cpu-map 3 2
  cpu-map 4 3
```

```
# Configuraciones generales del servicio
defaults
  log global
  mode tcp
  option tcplog
  option dontlognull
  maxconn 2000000
  timeout connect 3000000
  timeout client 6600000
  timeout server 6600000
# errorfile 400 /etc/haproxy/errors/400.http
# errorfile 403 /etc/haproxy/errors/403.http
# errorfile 408 /etc/haproxy/errors/408.http
# errorfile 500 /etc/haproxy/errors/500.http
# errorfile 502 /etc/haproxy/errors/502.http
# errorfile 503 /etc/haproxy/errors/503.http
# errorfile 504 /etc/haproxy/errors/504.http
```

```
# Configuración de la parte de frontal del servidor (en este caso, en el puerto 80. Es imprescindible configurar la p
frontend http-in
  bind *:80
  default_backend http-servers
```

```
# Configuración de los servidores a los que va a hacer el balanceo
backend http-servers
    mode tcp
    balance roundrobin
    server <SERVERNAME> <SERVERIP>:<PORT> send-proxy
#p.e: server COOL-AD22 173.230.133.22:9080 send-proxy
```

```
# Configuración de la parte de frontal del servidor (puerto 443)
frontend https-in
    bind *:443
    default_backend https-servers
```

```
# Configuración de los servidores a los que va a hacer el balanceo
backend https-servers
    mode tcp
    balance roundrobin
    server <SERVERNAME> <SERVERIP>:<PORT> send-proxy
#p.e: server COOL-AD22 173.230.133.22:9443 send-proxy
```

Una vez creado el fichero ya podremos arrancar el servicio mediante los scripts anteriormente configurados.

# Configuración de Frontales

Con la configuración que hemos realizado de HAProxy, es necesario que en los frontales también hagamos algunos cambios. En este caso, al estar configurado con Nginx los cambios serían los siguientes:

Ejemplo de fichero de configuración del virtualhost con el dominio que queramos pasar por el balanceador:

```
# block for proxy traffic
server {
```

```
    # port elb is forwarding ssl traffic to
    listen 9443 ssl proxy_protocol;
```

```
    # sets the proper client ip
    real_ip_header proxy_protocol;
```

```
    # load balancer subnet ip
    set_real_ip_from 10.0.0.0/16;
```

```
    server_name acme.com www.acme.com;
```

```
    ssl on;
    ssl_certificate /etc/ssl/acme/acme.com.crt;
    ssl_certificate_key /etc/ssl/acme/acme.com.key;
```

```
}
```

Como veis lo que hacemos es configurar para que escuche por un puerto diferente para el tráfico que viene del balanceador, ya que ese tráfico viene encriptado con el proxy\_protocol:

If you have multiple server blocks running on the same port (virtual hosts), any port that includes proxy\_protocol in

---

Revision #1

Created 28 November 2023 19:29:55 by adminROM

Updated 28 November 2023 19:30:09 by adminROM