

# HaProxy

This is not a tutorial of how haproxy works, this is just some notes on a config i did, and some of the options i used that made it stable for what i needed.

In the example bellow you will find a acceptable cipher, how to add a cookie sessions on HA, SSL offloading, xforward's, ha stats, good timeout vaules, and a httpchk.

```
global
    log 127.0.0.1 local0 warning
    maxconn 10000
    user haproxy
    group haproxy
    daemon
    spread-checks 5
    tune.ssl.default-dh-param 2048
    ssl-default-bind-ciphers ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-
SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-SHA384:DHE-RSA-AES128-GCM-
SHA256:DHE-DSS-AES128-GCM-SHA256:kEDH+AESGCM:ECDHE-RSA-AES128-SHA256:ECDHE-ECDSA-AES128-
SHA256:ECDHE-RSA-AES128-SHA:ECDHE-ECDSA-AES128-SHA:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES256-
SHA384:ECDHE-RSA-AES256-SHA:ECDHE-ECDSA-AES256-SHA:DHE-RSA-AES128-SHA256:DHE-RSA-AES128-
SHA:DHE-DSS-AES128-SHA256:DHE-RSA-AES256-SHA256:DHE-DSS-AES256-SHA:DHE-RSA-AES256-SHA:AES128-
GCM-SHA256:AES256-GCM-SHA384:AES128-SHA256:AES256-SHA256:AES128-SHA:AES256-
SHA:AES:CAMELLIA:DES-CBC3-SHA:!aNULL:!eNULL:!EXPORT:!DES:!RC4:!MD5:!PSK:!aECDH:!EDH-DSS-DES-
CBC3-SHA:!EDH-RSA-DES-CBC3-SHA:!KRB5-DES-CBC3-SHA

defaults
    log global
    option dontlognull
    retries 3
    option redispatch
    maxconn 10000
    mode http
    option dontlognull
    option httpclose
    option httpchk
    timeout connect 5000ms
```

```

timeout client 150000ms
timeout server 30000ms
timeout check 1000

listen lb_stats
  bind      {PUBLIC IP}:80
  balance roundrobin
  server lb1 127.0.0.1:80
  stats uri /
  stats realm "HAProxy Stats"
  stats auth admin:FsoqyNpJAYuD

frontend frontend_{PUBLIC IP}_https
  mode tcp
  bind      {PUBLIC IP}:443 ssl crt /etc/haproxy/ssl/domain.com.pem no-ssl3
  reqadd X-Forwarded-Proto:\ https
  http-request add-header X-CLIENT-IP %[src]
  option forwardfor
  default_backend backend_cluster_http_web1_web2

frontend frontend_{PUBLIC IP}_http
  bind      {PUBLIC IP}:80
  reqadd X-Forwarded-Proto:\ https
  http-request add-header X-CLIENT-IP %[src]
  option forwardfor
  default_backend backend_cluster_http_web1_web2

frontend frontend_www_custom
  bind      {PUBLIC IP}:666
  option forwardfor
  default_backend backend_cluster_http_web1_web2

backend backend_cluster_http_web1_web2
  option httpchk HEAD /
  server web1 10.1.2.100:80 weight 1 check cookie web1 inter 1000 rise 5 fall 1
  server web2 10.1.2.101:80 weight 1 check cookie web2 inter 1000 rise 5 fall 1

```

Enable xforward on httpd.conf on the web servers

```
LogFormat "%{X-Forwarded-For}i %h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\" "  
combine  
LogFormat "%{X-Forwarded-For}i %h %l %u %t \"%r\" %s %b \"%{Referer}i\" \"%{User-agent}i\""  
combined-forwarded
```

## Cookie

It is also possible to use the session cookie provided by the backend server.

```
backend www  
    balance roundrobin  
    mode http  
    cookie PHPSESSID prefix indirect nocache  
    server web1 10.1.2.100:80 check cookie web1  
    server web2 10.1.2.101:80 check cookie web2
```

In this example we will intercept the PHP session cookie and add / remove the reference of the backend server.

The prefix keyword allows you to reuse an application cookie and prefix the server identifier, then delete it in the following queries.

Default name of cookies by type of feeder backend:

Java : JSESSIONID

ASP.Net : ASP.NET\_SessionId

ASP : ASPSESSIONID

PHP : PHPSESSID

## Active/Passive config

```
backend backend_web1_primary  
    option httpchk HEAD /  
    server web1 10.1.2.100:80 check  
    server web2 10.1.2.101:80 check backup  
  
backend backend_web2_primary  
    option httpchk HEAD /  
    server web2 10.1.2.100:80 check  
    server web1 10.1.2.101:80 check backup
```

Test config file:

```
haproxy -c -V -f /etc/haproxy/haproxy.cfg
```

# Hapee Check syntax

```
/opt/hapee-1.7/sbin/hapee-lb -c
```

## Hapee VRRP

```
# /etc/hapee-1.7/hapee-vrrp.cfg

vrrp_script chk_hapee {
    script "pidof hapee-lb"
    interval 2
}

vrrp_instance vrrp_1 {
    interface eth0
    virtual_router_id 51
    priority 101
    virtual_ipaddress_excluded {
        eth0
        eth1
    }
    track_interface {
        eth0 weight -2
        eth1 weight -2
    }
    track_script {
        chk_hapee
    }
}

vrrp_instance vrrp_2 {
    interface eth1
    virtual_router_id 51
    priority 101
    virtual_ipaddress_excluded {
```

```
        X.X.X.X
    }
    track_interface {
        eth0 weight -2
        eth1 weight -2
    }
    track_script {
        chk_hapee
    }
}
```

# Doc

<https://cbonte.github.io/haproxy-dconv/>

---

Revision #22

Created 2017-07-21 03:46:18 UTC by Dave

Updated 2019-07-15 17:45:33 UTC by Dave