



# **Document d'exploitation**

*Buon Jérémy  
BTS SIO SISR*

# Table des matières

- **Définition**
- **Installation**
- **Fonctionnement**
- **Fonctionnalités**

# ○ Définition

HAProxy (High Availability Proxy) est un logiciel open-source de répartition de charge et de proxy inversé. Cela signifie qu'il reçoit les requêtes des clients et les transmet aux serveurs appropriés en fonction de règles de routage configurées. Il peut prendre en charge différents protocoles tels que HTTP, HTTPS, TCP et SSL.

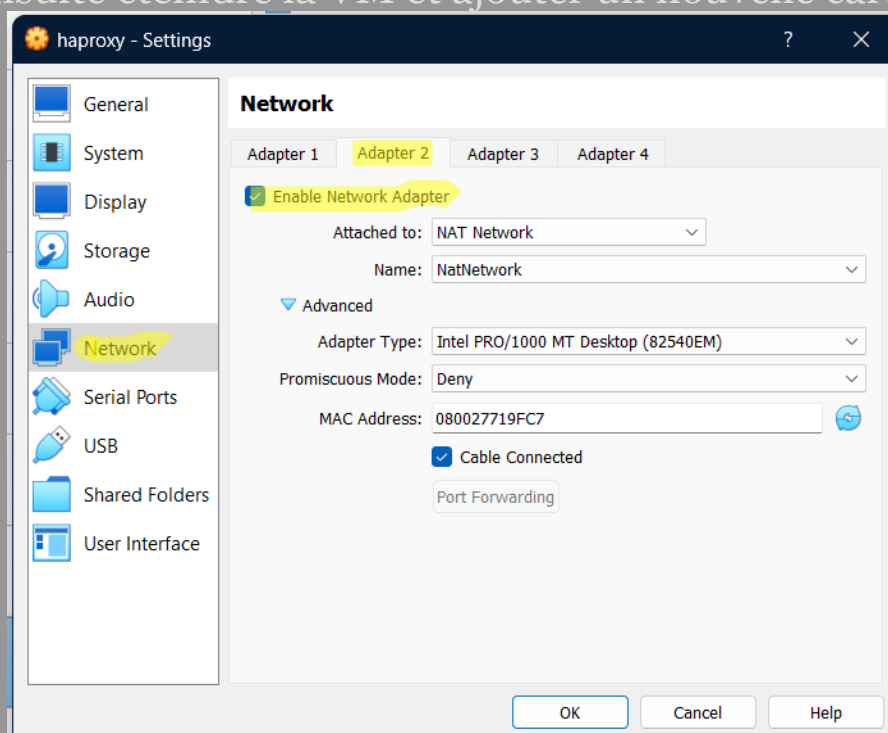
Les principales fonctionnalités de HAProxy incluent la répartition de charge des requêtes entre plusieurs serveurs, afin de garantir un fonctionnement fluide des applications. La détection de disponibilité des serveurs, la mise en cache, la compression, l'équilibrage de charge basé sur le contenu, la limitation des taux, la gestion des sessions et la haute disponibilité, permettent d'améliorer les performances, la fiabilité et la disponibilité des applications web.

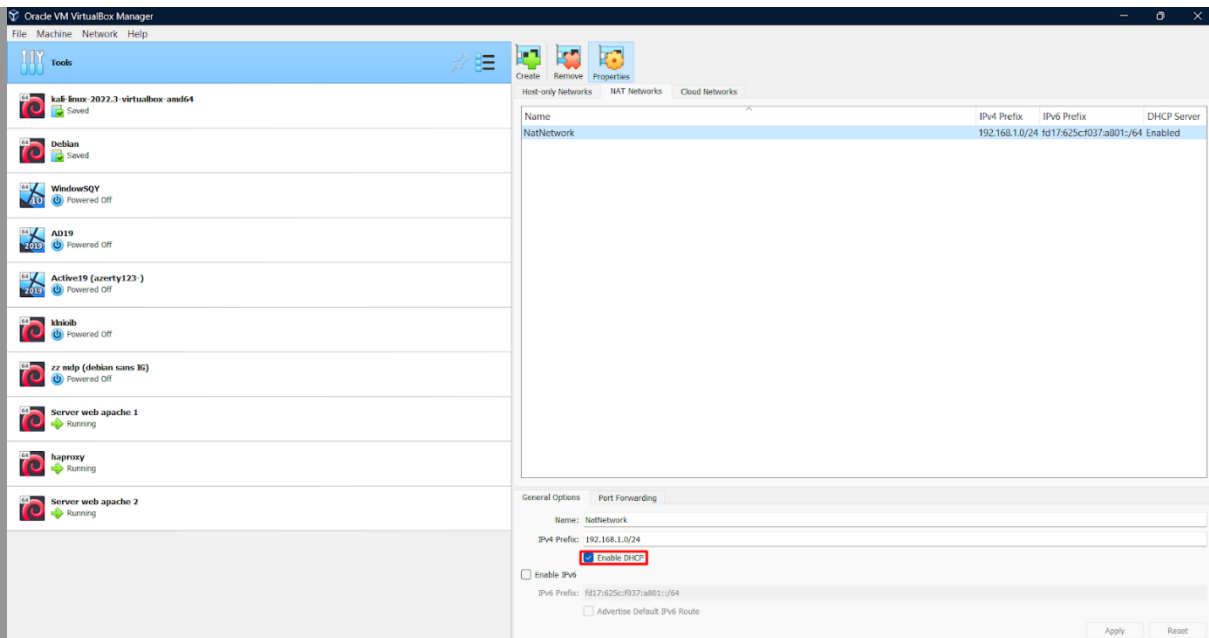
# ○ Installation

Sur une VM Debian, taper la commande *apt install haproxy* pour installer HAproxy.

```
root@debian:/home/zz# apt install haproxy
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  liblua5.3-0
Paquets suggérés :
  vim-haproxy haproxy-doc
Les NOUVEAUX paquets suivants seront installés :
  haproxy liblua5.3-0
0 mis à jour, 2 nouvellement installés, 0 à enlever et 0 non mis à jour.
Il est nécessaire de prendre 2 020 ko dans les archives.
Après cette opération, 4 316 ko d'espace disque supplémentaires seront utilisés.
Souhaitez-vous continuer ? [O/n] o
Réception de :1 http://deb.debian.org/debian bullseye/main amd64 liblua5.3-0 amd64 5.3.3-1.1+b1 [120
kB]
Réception de :2 http://security.debian.org/debian-security bullseye-security/main amd64 haproxy amd6
4 2.2.9-2+deb11u4 [1 900 kB]
2 020 ko réceptionnés en 1s (3 844 ko/s)
Sélection du paquet liblua5.3-0:amd64 précédemment désélectionné.
(Lecture de la base de données... 34008 fichiers et répertoires déjà installés.)
Préparation du dépaquetage de .../liblua5.3-0_5.3.3-1.1+b1_amd64.deb ...
Dépaquetage de liblua5.3-0:amd64 (5.3.3-1.1+b1) ...
Sélection du paquet haproxy précédemment désélectionné.
Préparation du dépaquetage de .../haproxy_2.2.9-2+deb11u4_amd64.deb ...
Dépaquetage de haproxy (2.2.9-2+deb11u4) ...
Paramétrage de liblua5.3-0:amd64 (5.3.3-1.1+b1) ...
Paramétrage de haproxy (2.2.9-2+deb11u4) ...
Created symlink /etc/systemd/system/multi-user.target.wants/haproxy.service → /lib/systemd/system/ha
prox.service.
Traitement des actions différées (« triggers ») pour libc-bin (2.31-13+deb11u5) ...
Traitement des actions différées (« triggers ») pour rsyslog (8.2102.0-2+deb11u1) ...
Traitement des actions différées (« triggers ») pour man-db (2.9.4-2) ...
root@debian:/home/zz#
```

Il faut ensuite éteindre la VM et ajouter une nouvelle carte réseau.





Il faut aussi activer le DHCP dans les paramètres du NAT Network.  
Il faut ensuite vérifier que la carte réseau est bien présente avec `ip a`.

```
root@debian:/home/zz# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:c2:9e:ed brd ff:ff:ff:ff:ff:ff
    inet6 fe80::a00:27ff:fec2:9eed/64 scope link dadfailed tentative
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:71:9f:c7 brd ff:ff:ff:ff:ff:ff
root@debian:/home/zz#
```

Puis, la configurer.

```
GNU nano 5.4 /etc/network/interfaces
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug enp0s3
iface enp0s3 inet dhcp

allow-hotplug enp0s8
iface enp0s8 inet static
```

Il faut la redémarrer pour que la configuration soit prise en compte avec `systemctl restart networking`, et revérifier avec `ip a`.

```

root@debian:/home/zz# systemctl restart networking
root@debian:/home/zz# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST> mtu 1500 qdisc pfifo_fast state DOWN group default qlen 1000
    link/ether 08:00:27:c2:9e:ed brd ff:ff:ff:ff:ff:ff
3: enp0s8: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 08:00:27:71:9f:c7 brd ff:ff:ff:ff:ff:ff
root@debian:/home/zz# ifup enp0s3
bash: ifup : commande introuvable
root@debian:/home/zz# /sbin/ifup enp0s3

```

Activer les interfaces réseau spécifiée en leur attribuant des adresses IP et en configurant les paramètres réseau associés.

```

root@debian:/home/zz# /sbin/ifup enp0s8
Internet Systems Consortium DHCP Client 4.4.1
Copyright 2004-2018 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/enp0s8/08:00:27:71:9f:c7
Sending on   LPF/enp0s8/08:00:27:71:9f:c7
Sending on   Socket/fallback
Created duid "\000\001\000\001+\247G\221\010\000'q\237\307".
DHCPDISCOVER on enp0s8 to 255.255.255.255 port 67 interval 6
DHCPDISCOVER on enp0s8 to 255.255.255.255 port 67 interval 9
DHCPOFFER of 192.168.1.5 from 192.168.1.3
DHCPREQUEST for 192.168.1.5 on enp0s8 to 255.255.255.255 port 67
DHCPACK of 192.168.1.5 from 192.168.1.3
bound to 192.168.1.5 -- renewal in 252 seconds.
root@debian:/home/zz# _

```

Pour vérifier l'intégrité du réseau, il faut ping les machines entre elles.

Il faut ping HAproxy depuis un serveur apache

```

zz@debian:~$ ping 192.168.1.5
PING 192.168.1.5 (192.168.1.5) 56(84) bytes of data.
64 bytes from 192.168.1.5: icmp_seq=1 ttl=64 time=0.789 ms
64 bytes from 192.168.1.5: icmp_seq=2 ttl=64 time=0.399 ms
64 bytes from 192.168.1.5: icmp_seq=3 ttl=64 time=0.824 ms
64 bytes from 192.168.1.5: icmp_seq=4 ttl=64 time=0.665 ms
^C
--- 192.168.1.5 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3054ms
rtt min/avg/max/mdev = 0.399/0.669/0.824/0.166 ms

```

# ○ Fonctionnement

Il faut ensuite éditer le fichier de configuration pour ajouter un frontend et un backend.

*Un frontend représente la configuration du serveur proxy inversé pour accepter les connexions des clients, tandis qu'un backend représente la configuration pour gérer les connexions avec les serveurs backend. Le frontend reçoit les requêtes des clients et le backend achemine ces requêtes vers les serveurs backend appropriés pour obtenir la réponse attendue.*

```
GNU nano 5.4 /etc/haproxy/haproxy.cfg
# See: https://ssl-config.mozilla.org/#server=haproxy&server-version=2.0.
ssl-default-bind-ciphers ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-G
ssl-default-bind-ciphersuites TLS_AES_128_GCM_SHA256:TLS_AES_256_GCM_SHA3
ssl-default-bind-options ssl-min-ver TLSv1.2 no-tls-tickets

defaults
    log          global
    mode         http
    option        httplog
    option        dontlognull
    timeout connect 5000
    timeout client  50000
    timeout server  50000
    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

frontend web
    bind *:80
    default_backend web_servers

backend web_servers
    balance roundrobin
    mode http
    server server1 192.168.1.6:80 check
    server server2 192.168.1.4:80 check
```

frontend

bind \*:80 signifie que l'IP disponible sera choisie.

backend

server "nomduserver" "IP" check

server "nomduserver" "IP" check

Et une fois les changements faits, il faut restart.

```
root@debian:/home/zz# systemctl restart haproxy
root@debian:/home/zz# systemctl status haproxy
```

Il faut ensuite vérifier le statut par précaution, avec systemctl status haproxy.

```

root@debian:/home/zz# systemctl status haproxy
● haproxy.service - HAProxy Load Balancer
   Loaded: loaded (/lib/systemd/system/haproxy.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2023-03-17 17:52:11 CET; 3min 13s ago
     Docs: man:haproxy(1)
           file:/usr/share/doc/haproxy/configuration.txt.gz
  Process: 786 ExecStartPre=/usr/sbin/haproxy -f $CONFIG -c -q $EXTRA_OPTS (code=exited, status=0/0)
 Main PID: 788 (haproxy)
    Tasks: 2 (limit: 2337)
   Memory: 40.2M
      CPU: 109ms
   CGroup: /system.slice/haproxy.service
           └─788 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy.pid
             └─790 /usr/sbin/haproxy -Ws -f /etc/haproxy/haproxy.cfg -p /run/haproxy.pid -S /run/haproxy.pid

mars 17 17:52:11 debian haproxy[788]: Proxy web started.
mars 17 17:52:11 debian haproxy[788]: Proxy web_servers started.
mars 17 17:52:11 debian haproxy[788]: Proxy web_servers started.
mars 17 17:52:11 debian haproxy[788]: [NOTICE] 075/175211 (788) : New worker #1 (790) forked
mars 17 17:52:11 debian systemd[1]: Started HAProxy Load Balancer.
mars 17 17:52:11 debian haproxy[790]: [WARNING] 075/175211 (790) : Server web_servers/server1 is DOWN
mars 17 17:52:11 debian haproxy[790]: [WARNING] 075/175212 (790) : Server web_servers/server2 is DOWN
mars 17 17:52:12 debian haproxy[790]: [NOTICE] 075/175212 (790) : haproxy version is 2.2.9-2+deb11u4
mars 17 17:52:12 debian haproxy[790]: [NOTICE] 075/175212 (790) : path to executable is /usr/sbin/haproxy
mars 17 17:52:12 debian haproxy[790]: [ALERT] 075/175212 (790) : backend 'web_servers' has no servers
lines 1-24/24 (END)

```



## ○ Fonctionnalités

Pour faire un transfert de fichiers entre serveurs, il faut installer les paquets SSH sur le serveur "client" avec la commande *apt install openssh-server*.

```
root@debian:/home/zz# apt install openssh-server
```

Une fois le paquet installé, il faut générer une clé SSH :

```
root@debian:/home/zz# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Created directory '/root/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa
Your public key has been saved in /root/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:yWUH+c2IkSwRuywlzj7gjfHVZVj1vbcG614rb0UN0Wk root@debian
The key's randomart image is:
+---[RSA 3072]-----+
|      o+.o..oo.|
|      ..+= ..E.|
|      . o.++++o.o|
|      o = *.+o o.o|
|      o + S . ....|
|      . B o . ...o|
|      o = . ...o |
|      . ....o |
|      .o+o |
+---[SHA256]-----+
root@debian:/home/zz# hostname -I
192.168.1.4
```

Une fois la clé générée il faudra l'adresse IP de l'ordinateur pour copier la clé sur le serveur client.

L'envoi de la clé nécessite d'entrer un mot de passe.

Pour cela, il faudra changer le fichier *sshd\_config*.

```
root@debian:/home/zz# nano /etc/ssh/sshd_config_
```

On change la ligne en commentaire passant de *prohibit-password* à *yes*.

```

GNU nano 5.4 /etc/ssh/sshd_config
# sshd_config(5) for more information.

# This sshd was compiled with PATH=/usr/bin:/bin:/usr/sbin:/sbin

# The strategy used for options in the default sshd_config shipped with
# OpenSSH is to specify options with their default value where
# possible, but leave them commented. Uncommented options override the
# default value.

Include /etc/ssh/sshd_config.d/*.conf

#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6

^G Aide      ^O Écrire    ^W Chercher  ^K Couper    ^T Exécuter  ^C EmplacementM-U Annuler
^X Quitter   ^R Lire fich.^_ Remplacer  ^U Coller    ^J Justifier ^_ Aller ligneM-E Refaire

```

Il faut ensuite redémarrer le paquet.

Ensuite, il sera possible de synchroniser les deux serveurs.

```

root@debian:/home/zz# ssh-copy-id -i ~/.ssh/id_rsa root@192.168.1.6
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/root/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install all the new keys
root@192.168.1.6's password:

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'root@192.168.1.6'"
and check to make sure that only the key(s) you wanted were added.

```

Il faut ensuite mettre en place une synchro de la page [\*index.html\*](#) toutes les 5 min grâce à la commande [\*crontab -e\*](#).

```
root@debian:/# crontab -e
```

```

GNU nano 5.4 /tmp/crontab.z9Gf0X/crontab
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# min h dom mon dow  command
*/5 * * * * scp /var/www/html/index.html root@192.168.1.6:/var/www/html/index.html

```

\*/5 pour une synchro toutes les 5 min.  
Il faudra vérifier dans les log si tout a bien fonctionné.

```

root@debian:/# nano /var/log/syslog_
Mar 24 14:02:48 debian dhclient[414]: DHCPREQUEST for 192.168.1.4 on enp0s3 to 255.255.255.255 port 67
Mar 24 14:02:48 debian named[450]: no longer listening on 169.254.13.42#53
Mar 24 14:02:48 debian dhclient[414]: DHCPACK of 192.168.1.4 from 192.168.1.3
Mar 24 14:02:48 debian named[450]: listening on IPv4 interface enp0s3, 192.168.1.4#53
Mar 24 14:02:48 debian dhclient[414]: bound to 192.168.1.4 -- renewal in 232 seconds.
Mar 24 14:05:01 debian CRON[2074]: (root) CMD (scp /var/www/html/index.html root@192.168.1.6:/var/www/html/index.html)
Mar 24 14:06:40 debian dhclient[414]: DHCPREQUEST for 192.168.1.4 on enp0s3 to 192.168.1.3 port 67
Mar 24 14:06:40 debian dhclient[414]: DHCPACK of 192.168.1.4 from 192.168.1.3
Mar 24 14:06:40 debian dhclient[414]: bound to 192.168.1.4 -- renewal in 276 seconds.

```