

# Mise en place d'HAPROXY pour contrer une attaque DoS

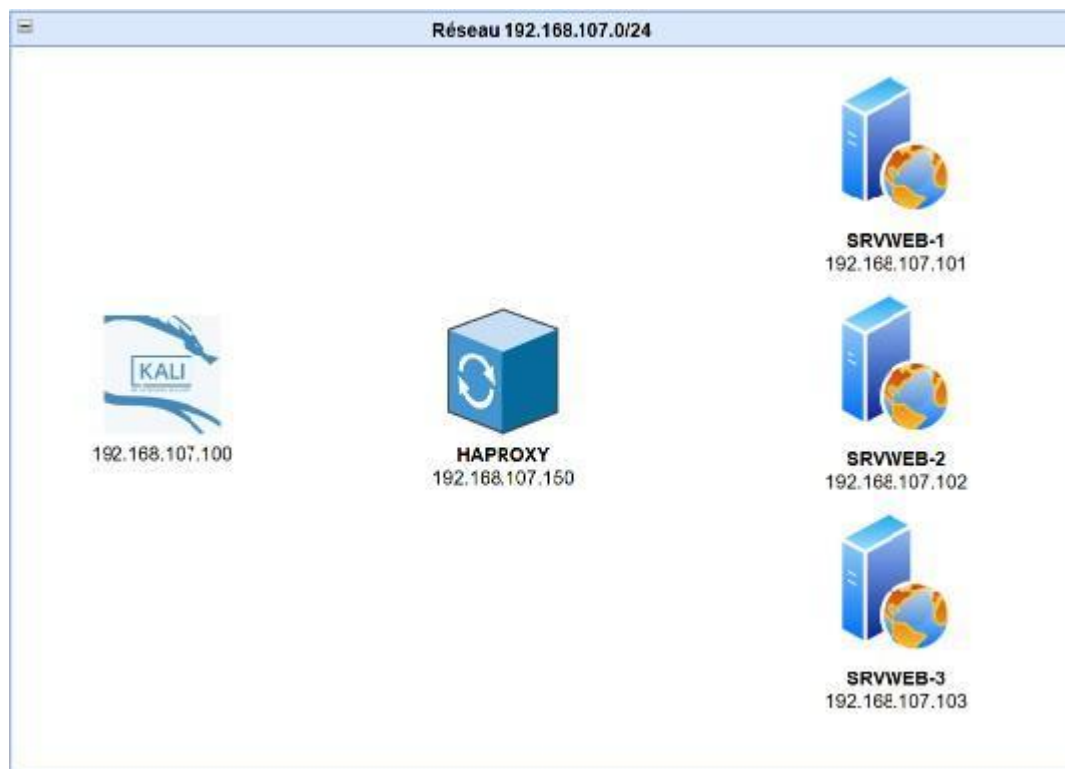
<b>1. Objectif de la Procédure</b>	<b>3</b>
<b>2. Contexte</b>	<b>3</b>
<b>3. Prérequis</b>	<b>4</b>
<b>3. Mise en place des machines</b>	<b>4</b>
Configuration des serveurs web (SRVWEB-X)	5
Configuration de la machine Kali	5
<b>4. Exécution d'une attaque DoS HTTP</b>	<b>6</b>
<b>5. Mise en place de HAProxy en tant que contre mesure</b>	<b>7</b>
Installation de HAProxy sur un serveur Debian	7
<b>6. Vérification de l'efficacité de la solution</b>	<b>8</b>
<b>7. Conclusion</b>	<b>9</b>

## 1. Objectif de la Procédure

L'objectif de cette procédure est de simuler une attaque par déni de service (DoS) HTTP en utilisant **slowhttptest**, puis de mettre en place HAProxy comme contre mesure afin de répartir la charge et assurer la disponibilité du service.

## 2. Contexte

### a) Schéma du contexte



Un reverse proxy est souvent configuré pour écouter les requêtes provenant de l'extérieur (réseau public) et les rediriger vers des machines en interne.

Dans notre cas, afin de faciliter la mise en place nous intégrerons l'ensemble des machines dans un

réseau unique.

Vous pouvez évidemment adapter le troisième octet du réseau (192.168.X.0/24) en fonction de votre

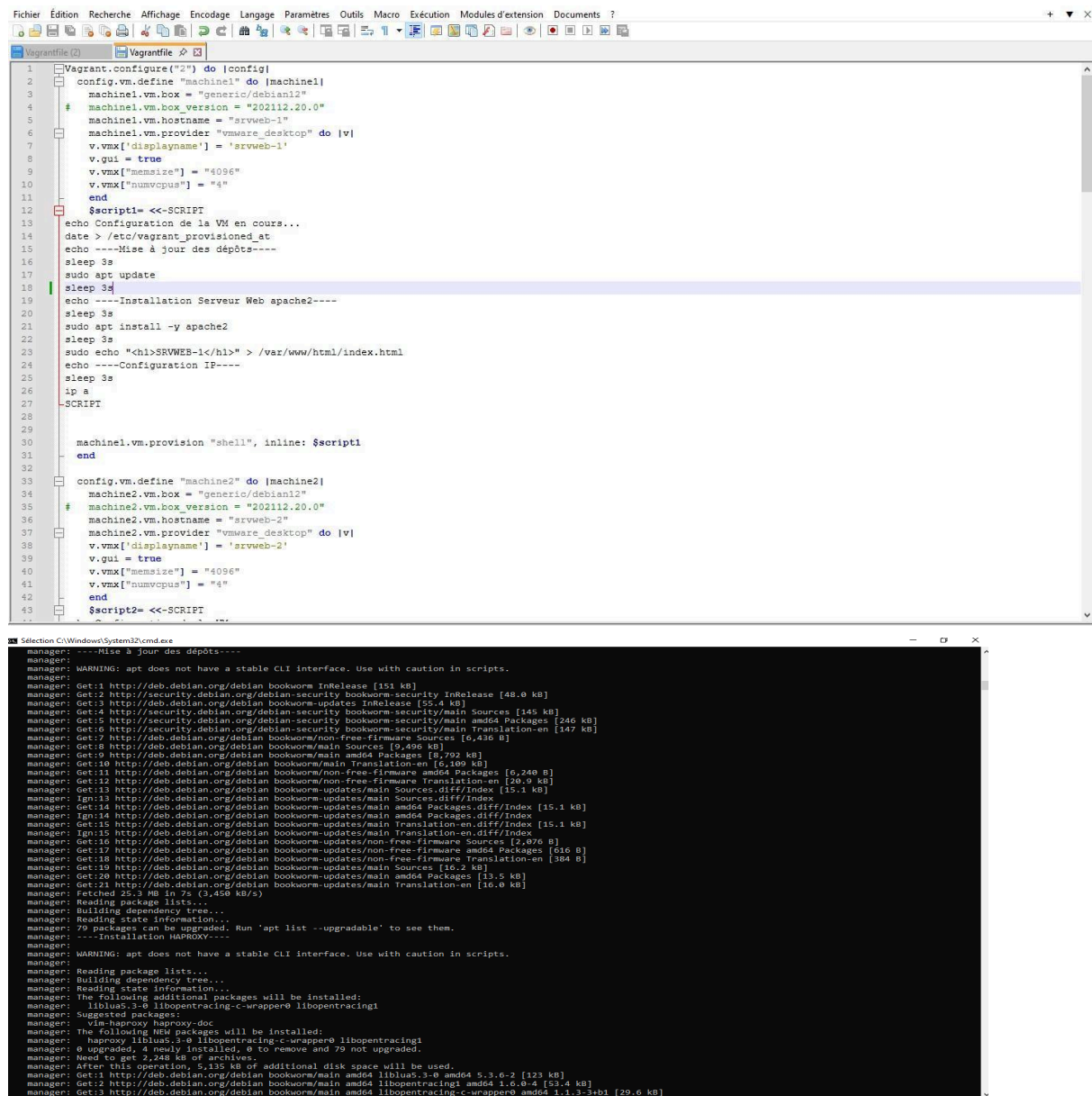
environnement de travail (réseau NAT de votre hyperviseur de niveau 2).

### 3. Prérequis

- 4 machines virtuelles Debian 12 (2-4 CPU, 4 Go RAM, 20 Go stockage, réseau NAT)
- 1 machine virtuelle Kali Linux
- Apache2 installé sur les serveurs Debian
- HAProxy installé sur un serveur Debian

### 4. Mise en place des machines

Ici nous utiliserons Vagrant



The image shows a Vagrantfile configuration for two Debian 12 virtual machines and a terminal window showing the execution of the Vagrantfile.

**Vagrantfile (2):**

```
1 Vagrant.configure("2") do |config|
2   config.vm.define "machine1" do |machine1|
3     machine1.vm.box = "generic/debian12"
4     machine1.vm.box_version = "202112.20.0"
5     machine1.vm.hostname = "srvweb-1"
6     machine1.vm.provider "vmware_desktop" do |v|
7       v.vmx["displayname"] = "srvweb-1"
8       v.gui = true
9       v.vmx["memsize"] = "4096"
10      v.vmx["numvcpus"] = "4"
11    end
12    $script1 = <<-SCRIPT
13    echo Configuration de la VM en cours...
14    date > /etc/vagrant.provisioned_at
15    echo ----Mise à jour des dépôts----
16    sleep 3s
17    sudo apt update
18    sleep 3s
19    echo ----Installation Serveur Web apache2----
20    sleep 3s
21    sudo apt install -y apache2
22    sleep 3s
23    sudo echo "chi>SRWEB-1</chi" > /var/www/html/index.html
24    echo ----Configuration IP----
25    sleep 3s
26    ip a
27  -SCRIPT
28
29  machine1.vm.provision "shell", inline: $script1
30 end
31
32
33 config.vm.define "machine2" do |machine2|
34   machine2.vm.box = "generic/debian12"
35   machine2.vm.box_version = "202112.20.0"
36   machine2.vm.hostname = "srvweb-2"
37   machine2.vm.provider "vmware_desktop" do |v|
38     v.vmx["displayname"] = "srvweb-2"
39     v.gui = true
40     v.vmx["memsize"] = "4096"
41     v.vmx["numvcpus"] = "4"
42   end
43   $script2 = <<-SCRIPT
```

**Terminal Output (C:\Windows\System32\cmd.exe):**

```
manager: ----Mise à jour des dépôts----
manager: WARNING: apt does not have a stable CLI interface. Use with caution in scripts.
manager: Get:1 http://deb.debian.org/debian bookworm InRelease [191 kB]
manager: Get:2 http://security.debian.org/debian-security bookworm-security InRelease [48.0 kB]
manager: Get:3 http://deb.debian.org/debian bookworm-updates InRelease [55.4 kB]
manager: Get:4 http://security.debian.org/debian-security bookworm-security/main Sources [145 kB]
manager: Get:5 http://security.debian.org/debian-security bookworm-security/main amd64 Packages [246 kB]
manager: Get:6 http://security.debian.org/debian-security bookworm-security/main Translation-en [147 kB]
manager: Get:7 http://deb.debian.org/debian bookworm/non-free-firmware Sources [6,436 B]
manager: Get:8 http://deb.debian.org/debian bookworm/main Sources [9,496 kB]
manager: Get:9 http://deb.debian.org/debian bookworm/main amd64 Packages [8,792 kB]
manager: Get:10 http://deb.debian.org/debian bookworm/main Translation-en [6,109 kB]
manager: Get:11 http://deb.debian.org/debian bookworm/non-free-firmware amd64 Packages [6,240 B]
manager: Get:12 http://deb.debian.org/debian bookworm/non-free-firmware Translation-en [20.9 kB]
manager: Get:13 http://deb.debian.org/debian bookworm-updates/main Sources.diff/Index [15.1 kB]
manager: Ign:13 http://deb.debian.org/debian bookworm-updates/main Sources.diff/Index [15.1 kB]
manager: Get:14 http://deb.debian.org/debian bookworm-updates/main amd64 Packages.diff/Index [15.1 kB]
manager: Ign:14 http://deb.debian.org/debian bookworm-updates/main amd64 Packages.diff/Index [15.1 kB]
manager: Ign:15 http://deb.debian.org/debian bookworm-updates/main Translation-en.diff/Index [15.1 kB]
manager: Ign:15 http://deb.debian.org/debian bookworm-updates/main Translation-en.diff/Index [15.1 kB]
manager: Get:16 http://deb.debian.org/debian bookworm-updates/non-free-firmware Sources [2,076 B]
manager: Get:17 http://deb.debian.org/debian bookworm-updates/non-free-firmware amd64 Packages [616 B]
manager: Get:18 http://deb.debian.org/debian bookworm-updates/non-free-firmware Translation-en [384 B]
manager: Get:19 http://deb.debian.org/debian bookworm-updates/main Sources [16.2 kB]
manager: Get:20 http://deb.debian.org/debian bookworm-updates/main amd64 Packages [13.5 kB]
manager: Get:21 http://deb.debian.org/debian bookworm-updates/main Translation-en [10.0 kB]
manager: Fetched 25.3 MB in 7s (3,450 kB/s)
manager: Reading package lists...
manager: Building dependency tree...
manager: Reading state information...
manager: The following additional packages will be installed:
manager:   liblua5.3-0 libpcre2-trace-c-wrapper libpcre2-trace-c
manager: Suggested packages:
manager:   vim-haproxy haproxy-doc
manager: The following NEW packages will be installed:
manager:   haproxy liblua5.3-0 libpcre2-trace-c-wrapper libpcre2-trace-c
manager: 0 upgraded, 4 newly installed, 0 to remove and 79 not upgraded.
manager: Need to get 2,248 kB of archives.
manager: After this operation, 5,135 kB of additional disk space will be used.
manager: Get:1 http://deb.debian.org/debian bookworm/main amd64 liblua5.3-0 amd64 5.3.6-2 [123 kB]
manager: Get:2 http://deb.debian.org/debian bookworm/main amd64 libpcre2-trace-c amd64 1.0.0-4 [53.4 kB]
manager: Get:3 http://deb.debian.org/debian bookworm/main amd64 libpcre2-trace-c-wrapper amd64 1.1.3-3+b1 [29.6 kB]
```

## Configuration des serveurs web (SRVWEB-X)

Installer Apache2 :

```
sudo apt update && sudo apt install -y apache2
```

Modifier la page d'accueil de chaque serveur :

```
echo "SRVWEB-1" | sudo tee /var/www/html/index.html
```

Configurer une IP fixe pour chaque serveur.

Tester l'accessibilité des serveurs via un navigateur.



## Configuration de la machine Kali

Mettre à jour les paquets :

```
sudo apt update && sudo apt upgrade -y
```

Installer slowhttptest :

```
sudo apt install -y slowhttptest
```

## 5. Exécution d'une attaque DoS HTTP

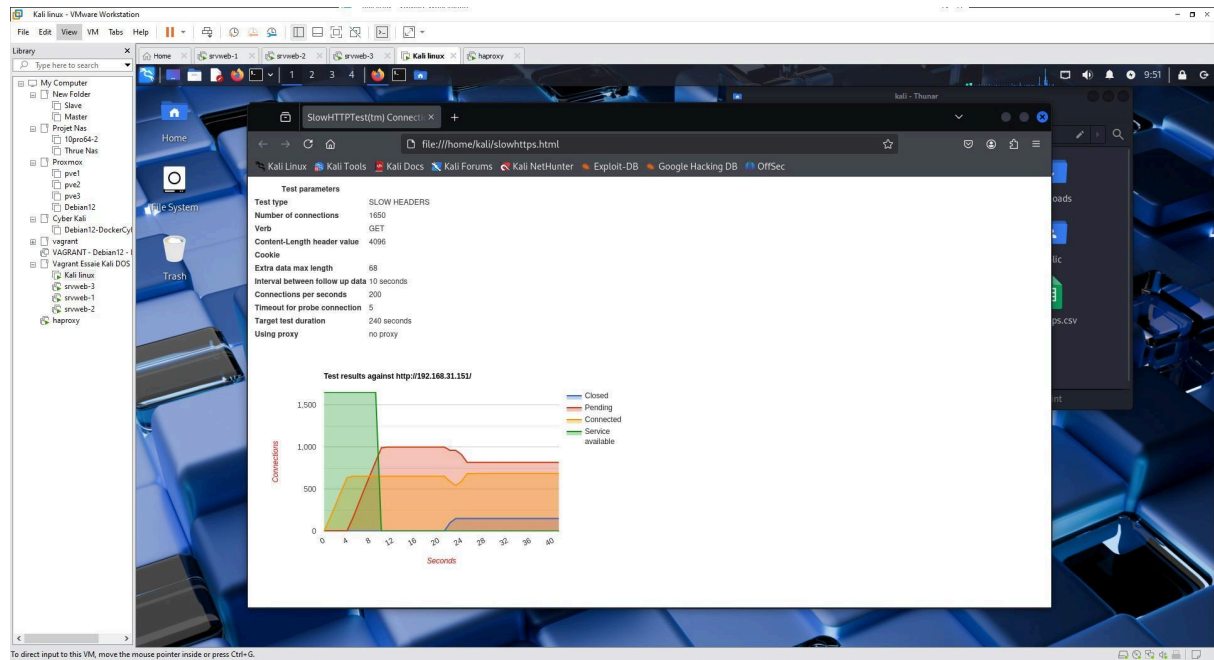
```
kali@kali: ~  
File Actions Edit View Help  
verb: GET  
cookie:  
Content-Length header value: 4096  
follow up data max size: 68  
interval between follow up data: 10 seconds  
connections per seconds: 200  
probe connection timeout: 5 seconds  
test duration: 240 seconds  
using proxy: no proxy  
  
Fri Mar 7 09:48:56 2025:  
slow HTTP test status on 40th second:  
  
initializing: 0  
pending: 815  
connected: 685  
error: 0  
closed: 150  
service available: NO  
^CFri Mar 7 09:48:58 2025:  
Test ended on 41th second  
Exit status: Cancelled by user  
CSV report saved to slowhttps.csv  
HTML report saved to slowhttps.html  
  
(kali@kali)-[~]  
$ slowhttptest -c 1650 -H -g -o slowhttps -i 10 -r 200 -t GET -u http://192  
.168.31.151
```

Lancer l'attaque contre **SRVWEB-1** :

```
slowhttptest -c 1000 -H -g -o slowhttp -i 10 -r 200 -t GET -u http://192.168.107.101
```

Observer l'indisponibilité du site depuis un navigateur.

Analyser les logs et le rapport slowhttp.html.



## 6. Mise en place de HAProxy en tant que contre mesure

### Installation de HAProxy sur un serveur Debian

Installer HAProxy :

```
sudo apt update && sudo apt install -y haproxy
```

Configurer HAProxy (/etc/haproxy/haproxy.cfg) :

```
frontend myfrontend
```

```
bind 192.168.107.150:80
```

```
default_backend myservers
```

```
backend myservers
```

```
balance
```

```
roundrobin
```

```
server server1 192.168.107.101:80
```

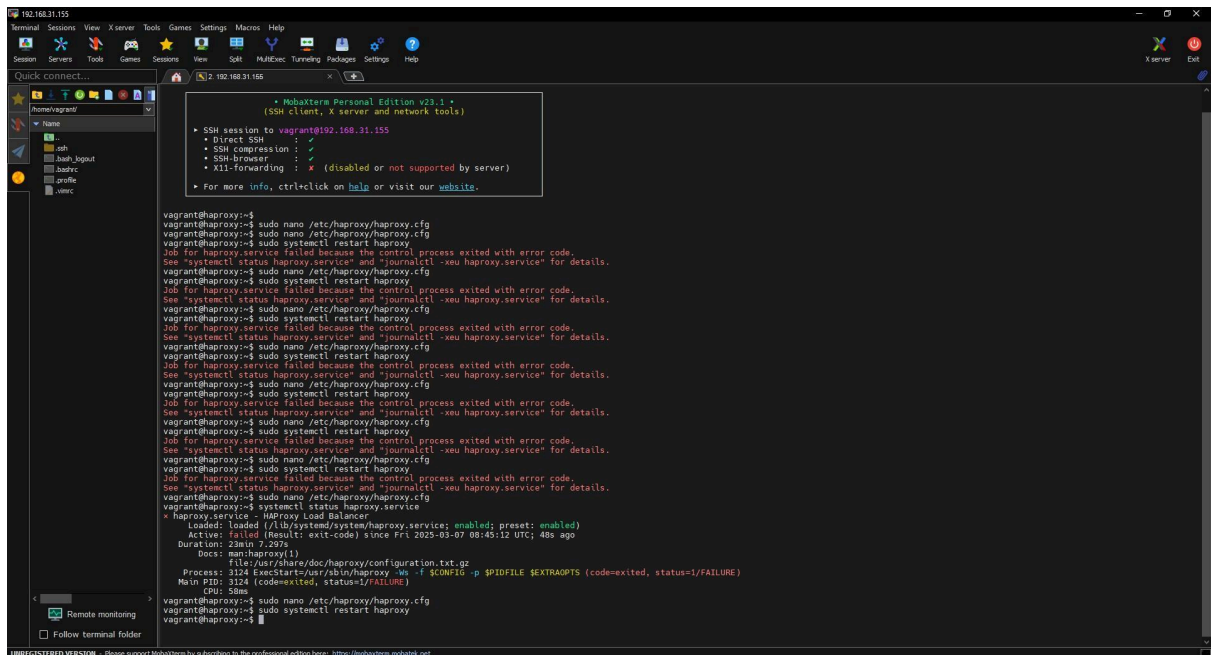
```
check server server2
```

```
192.168.107.102:80 check server
```

```
server3 192.168.107.103:80 check
```

Redémarrer HAProxy :

```
sudo systemctl restart haproxy
```



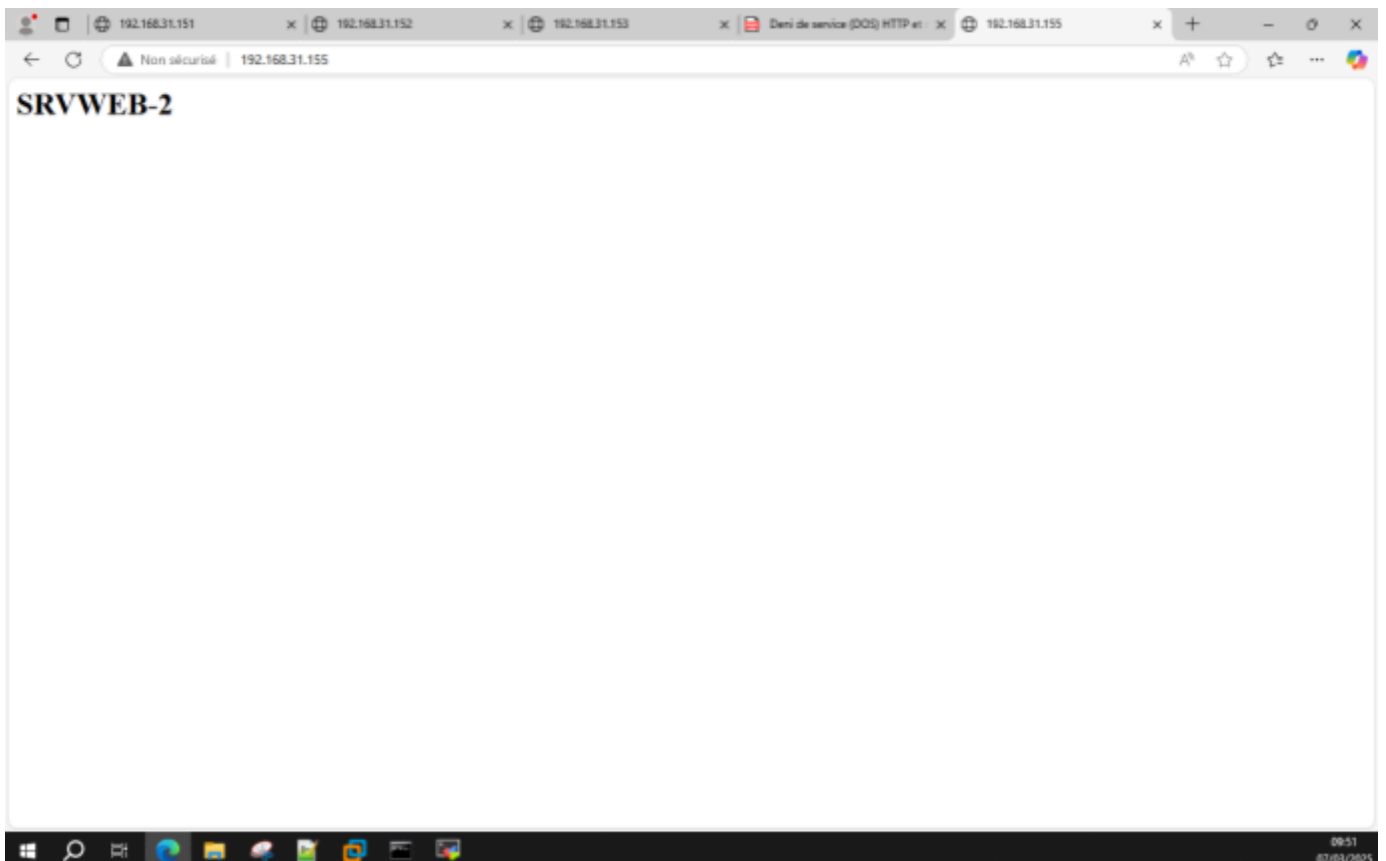
## 7. Vérification de l'efficacité de la solution

Accéder au site via HAProxy (192.168.107.150) et observer la répartition de charge.

Relancer l'attaque DoS contre HAProxy :

```
slowhttptest -c 1000 -H -g -o slowhttp-haproxy -i 10 -r 200 -t GET -u
http://192.168.107.150
```

Observer que le site reste disponible grâce à la répartition de charge.





## **8. Conclusion**

Cette procédure a permis de mettre en œuvre une attaque DoS HTTP et d'analyser son impact. L'utilisation de HAProxy comme reverse proxy a montré son efficacité en maintenant la disponibilité du service grâce à la répartition des requêtes sur plusieurs serveurs.