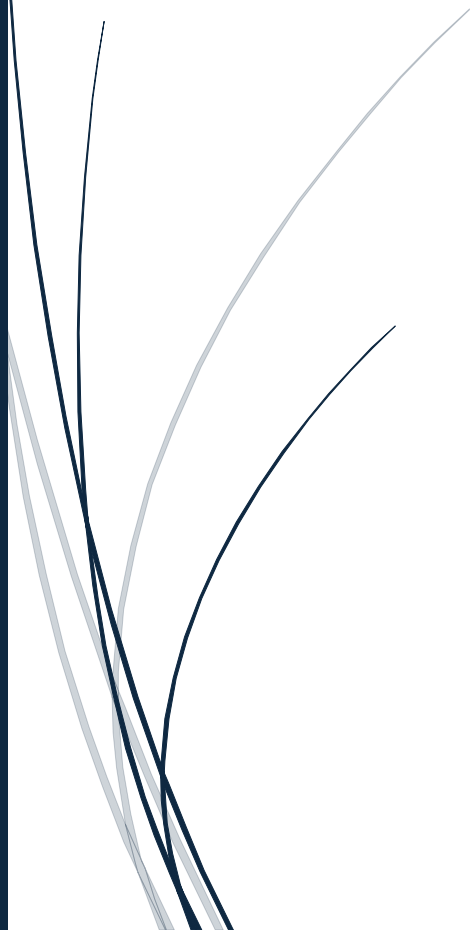


08/04/2025

# Configuration du Pare-feu iptables et Mise en place du proxy pour une DMZ



Anthony FALLA

## Table des matières

**Aucune entrée de table des matières n'a été trouvée.**

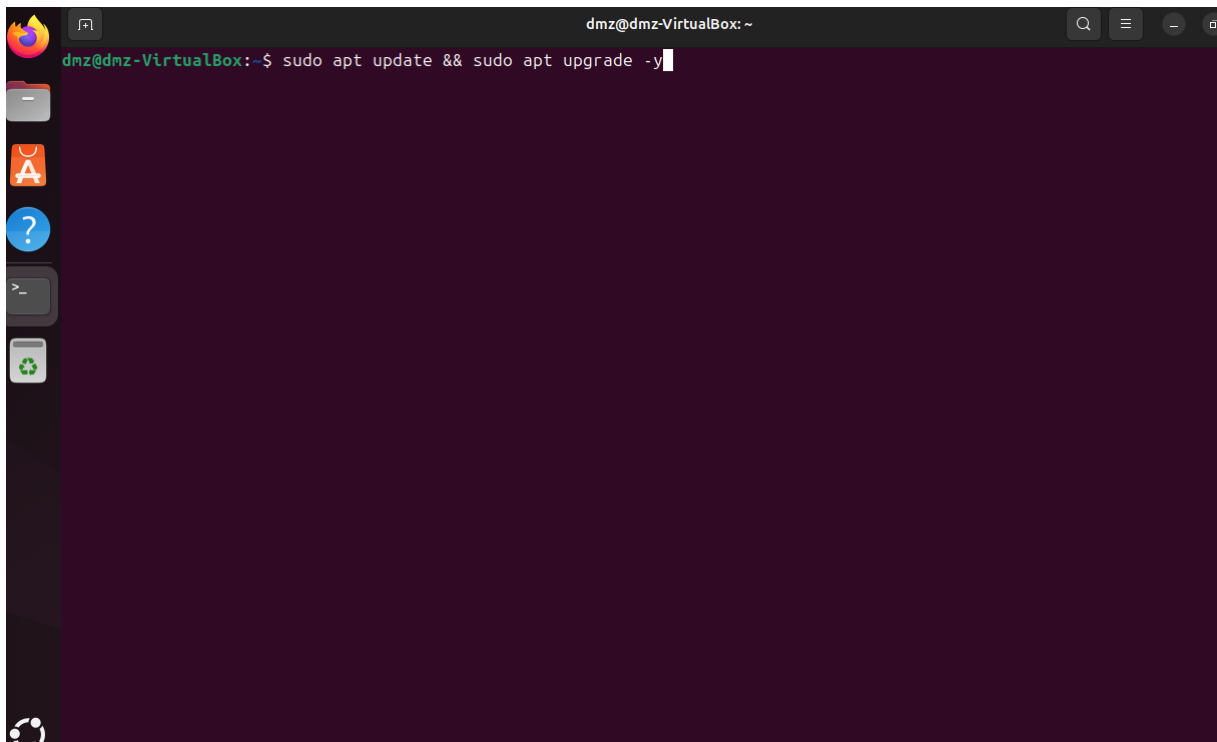
## Préconditions :

- Le logiciel VirtualBox doit être installé.
- 3 machines virtuelles doivent être créées (ici sous Ubuntu)
- Utiliser 3 cartes réseau : NAT, un réseau interne pour la DMZ et un autre pour la machine client.

## Mise à jour de la machine

Avant d'effectuer les opérations, il est primordial de mettre à jour la machine. Pour ce faire, tapons dans le terminal la commande suivante :

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



## Configuration d'une adresse IP statique

Nous allons maintenant mettre une adresse ip publique. Nous allons donc utiliser la commande :

```
ls /etc/netplan/
```

Cette commande permet de voir les fichiers présents dans netplan.

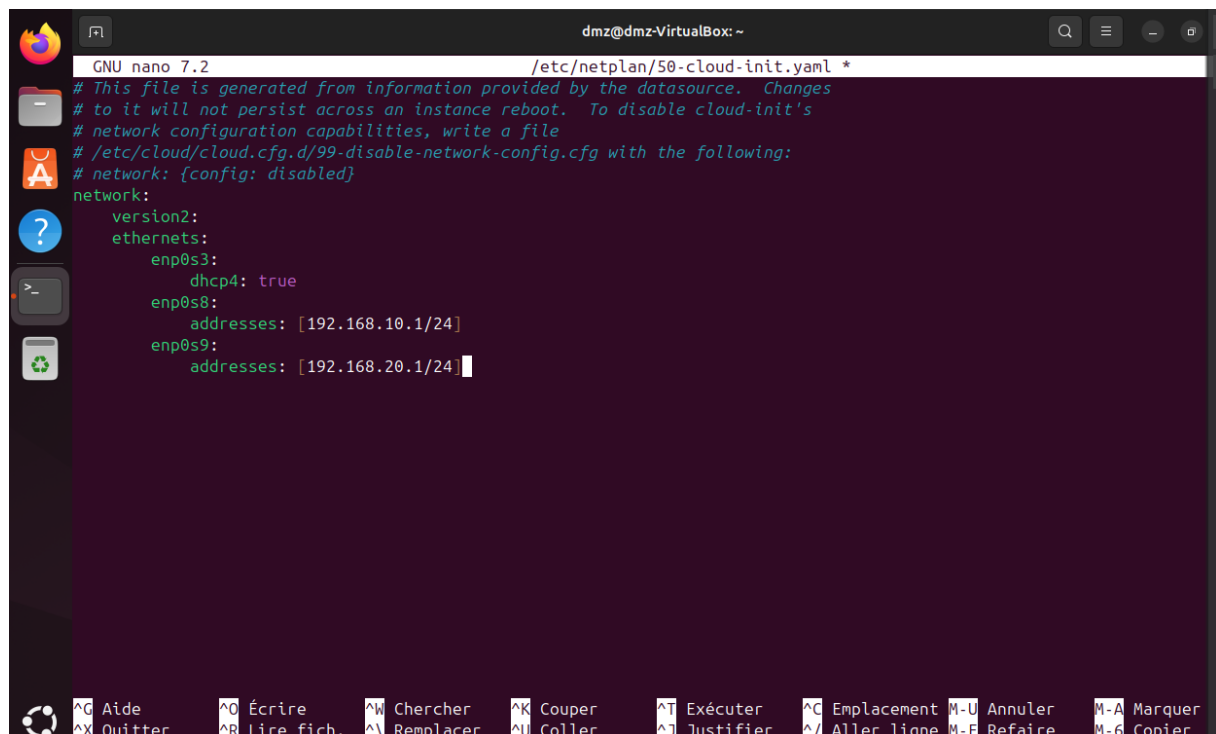
```
dmz@dmz-VirtualBox:~$ ls /etc/netplan/
50-cloud-init.yaml
dmz@dmz-VirtualBox:~$
```

Accédons à ce fichier avec :

```
sudo nano /etc/netplan/50-cloud-init.yaml
```

```
dmz@dmz-VirtualBox:~$ sudo nano /etc/netplan/50-cloud-init.yaml
```

Configurons le fichier comme ceci :



```
dmz@dmz-VirtualBox: ~
GNU nano 7.2 /etc/netplan/50-cloud-init.yaml *
# This file is generated from information provided by the datasource. Changes
# to it will not persist across an instance reboot. To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
  version2:
  ethernet:
    enp0s3:
      dhcp4: true
    enp0s8:
      addresses: [192.168.10.1/24]
    enp0s9:
      addresses: [192.168.20.1/24]
```

L'interface enp0s3 correspond au NAT. Enp0s8 l'interface liée à la DMZ et enp0s9 à l'interface client. Sauvegardons le fichier avec Ctrl + X et appliquons la configuration avec :

```
sudo netplan apply
```

```
dmz@dmz-VirtualBox:~$ sudo netplan apply
```

## Configuration du pare-feu iptables

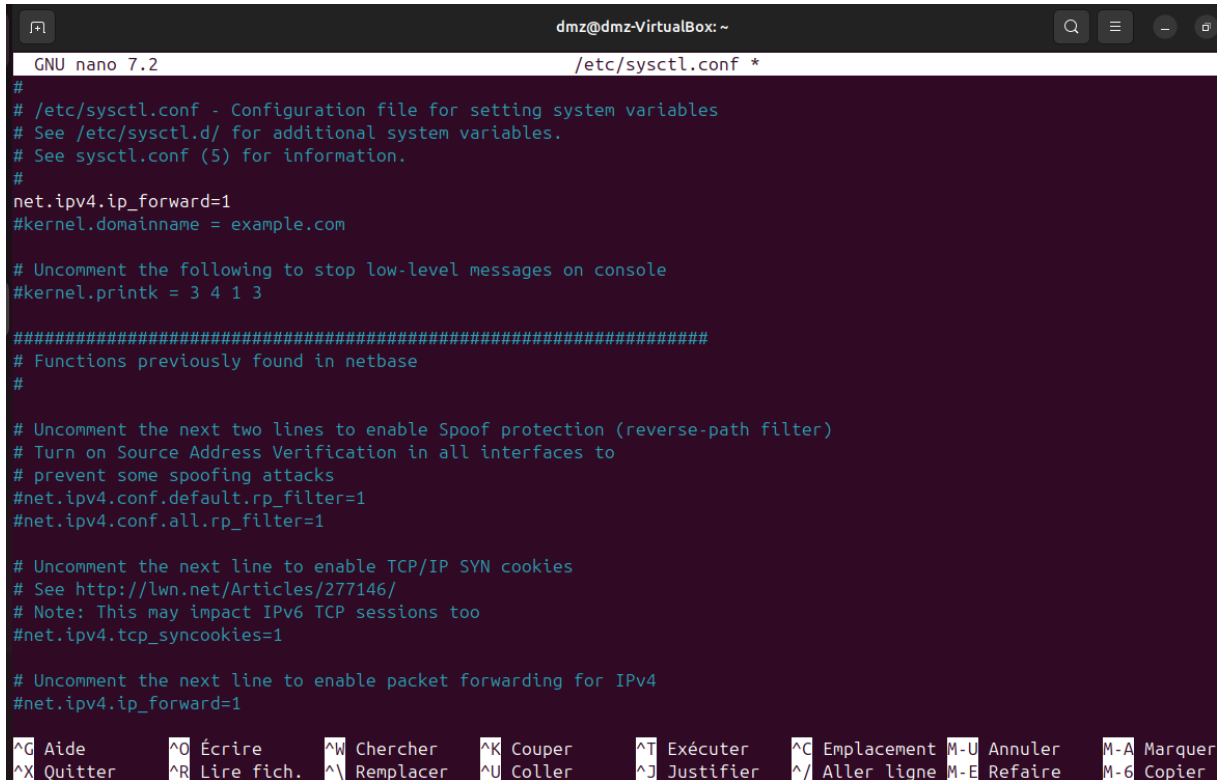
Activons désormais le forwarding IP avec la commande :

```
sudo nano /etc/sysctl.conf
```

```
dmz@dmz-VirtualBox:~$ sudo nano /etc/sysctl.conf
```

Dans ce fichier, ajoutons la commande :

```
net.ipv4.ip_forward=1
```



```
GNU nano 7.2 /etc/sysctl.conf *
#
# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.
#
net.ipv4.ip_forward=1
#kernel.domainname = example.com

# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3

#####
# Functions previously found in netbase
#

# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
#net.ipv4.ip_forward=1

^G Aide      ^O Écrire   ^W Chercher ^K Couper   ^T Exécuter ^C Emplacement M-U Annuler  M-A Marquer
^X Quitter   ^R Lire fich. ^\ Remplacer ^U Coller   ^J Justifier ^/ Aller ligne M-E Refaire  M-6 Copier
```

Activons ce changement avec la commande :

```
sudo sysctl -p
```

```
dmz@dmz-VirtualBox:~$ sudo sysctl -p
net.ipv4.ip_forward = 1
dmz@dmz-VirtualBox:~$
```

Ajoutons désormais les règles au pare-feu.

Par sécurité, on commence toujours par vider les règles avec ces 2 commandes :

```
sudo iptables -F
```

```
sudo iptables -t nat -F
```

```
dmz@dmz-VirtualBox:~$ sudo iptables -F
dmz@dmz-VirtualBox:~$ sudo iptables -t nat -F
```

Commençons par autoriser le réseau NAT à accéder à internet avec la commande :

```
sudo iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE
```

```
dmz@dmz-VirtualBox:~$ sudo iptables -t nat -A POSTROUTING -o enp0s3 -j MASQUERADE
```

## Installation du Proxy Squid

A présent, installons le proxy squid avant de continuer la configuration de pare-feu avec la commande :

```
sudo apt install squid -y
```

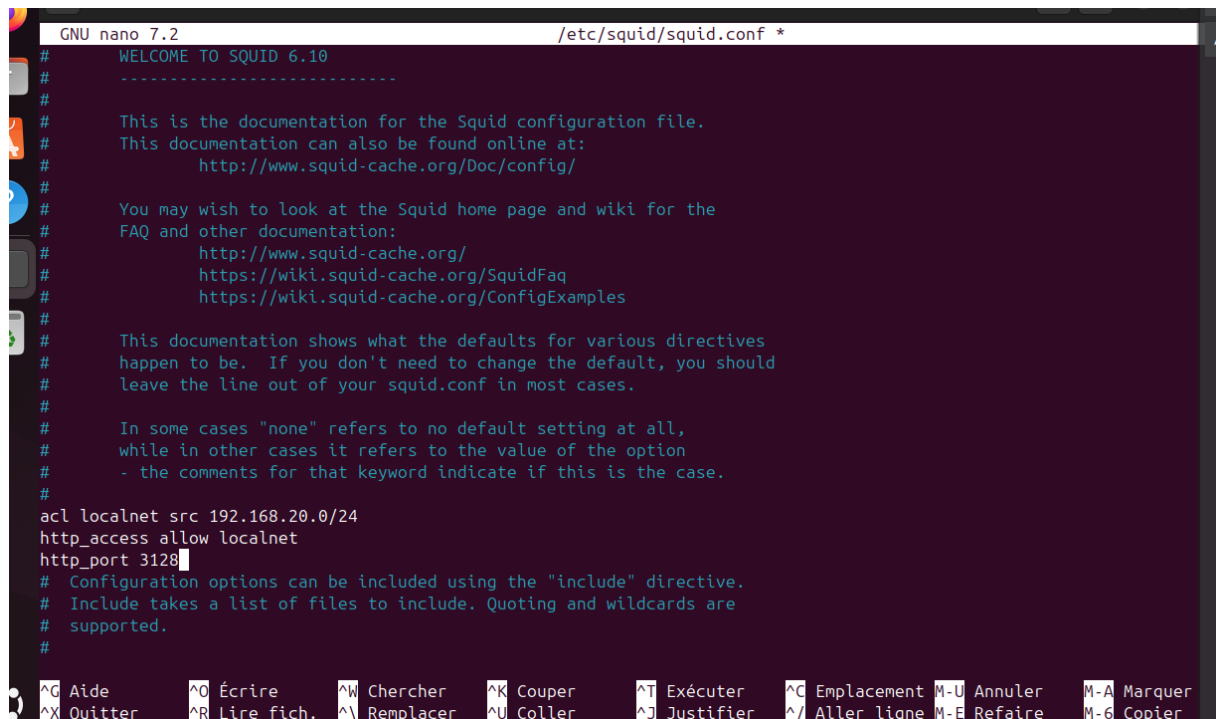
```
dmz@dmz-VirtualBox:~$ sudo apt install squid -y
```

Modifions la configuration de squid grâce à la commande :

```
sudo nano /etc/squid/squid.conf
```

```
dmz@dmz-VirtualBox:~$ sudo nano /etc/squid/squid.conf
```

Ajoutons ceci pour permettre au réseau client d'accéder à internet.



```
GNU nano 7.2 /etc/squid/squid.conf *
# WELCOME TO SQUID 6.10
# -----
#
# This is the documentation for the Squid configuration file.
# This documentation can also be found online at:
# http://www.squid-cache.org/Doc/config/
#
# You may wish to look at the Squid home page and wiki for the
# FAQ and other documentation:
# http://www.squid-cache.org/
# https://wiki.squid-cache.org/SquidFaq
# https://wiki.squid-cache.org/ConfigExamples
#
# This documentation shows what the defaults for various directives
# happen to be. If you don't need to change the default, you should
# leave the line out of your squid.conf in most cases.
#
# In some cases "none" refers to no default setting at all,
# while in other cases it refers to the value of the option
# - the comments for that keyword indicate if this is the case.
#
acl localnet src 192.168.20.0/24
http_access allow localnet
http_port 3128
# Configuration options can be included using the "include" directive.
# Include takes a list of files to include. Quoting and wildcards are
# supported.
#
```

Une fois ceci fait, redémarrons squid avec la commande :

```
sudo systemctl restart squid
```

```
dmz@dmz-VirtualBox:~$ sudo systemctl restart squid
```

Ajoutons maintenant les autres règles du pare-feu.

Autorisons l'IP forwarding avec ces commandes :

```
dmz@dmz-VirtualBox:~$ sudo iptables -A FORWARD -i enp0s9 -o enp0s3 -j ACCEPT
dmz@dmz-VirtualBox:~$ sudo iptables -A FORWARD -i enp0s8 -o enp0s3 -j ACCEPT
dmz@dmz-VirtualBox:~$ sudo iptables -A FORWARD -i enp0s3 -o enp0s9 -m state --state RELATED,ESTABLISHED -j ACCEPT
dmz@dmz-VirtualBox:~$ sudo iptables -A FORWARD -i enp0s3 -o enp0s8 -m state --state RELATED,ESTABLISHED -j ACCEPT
```

Maintenant, nous allons rendre ces règles permanentes. Pour ce faire, on va utiliser les commandes :

```
sudo apt install iptables-persistent
```

```
sudo netfilter-persistent save
```

```
dmz@dmz-VirtualBox:~$ sudo apt install iptables-persistent
```

```
dmz@dmz-VirtualBox:~$ sudo netfilter-persistent save
run-parts: executing /usr/share/netfilter-persistent/plugins.d/15-ip4tables save
run-parts: executing /usr/share/netfilter-persistent/plugins.d/25-ip6tables save
```

Testons de ping les passerelles et le DNS sur chacune des machines :

Machine DMZ :

```
dmz@dmz-VirtualBox:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=254 time=17.6 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=254 time=14.0 ms
^C
--- 8.8.8.8 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 13.988/15.794/17.601/1.806 ms
dmz@dmz-VirtualBox:~$ ping 192.168.10.1
PING 192.168.10.1 (192.168.10.1) 56(84) bytes of data.
64 bytes from 192.168.10.1: icmp_seq=1 ttl=64 time=2.09 ms
64 bytes from 192.168.10.1: icmp_seq=2 ttl=64 time=3.36 ms
^C
--- 192.168.10.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 2.089/2.724/3.359/0.635 ms
```

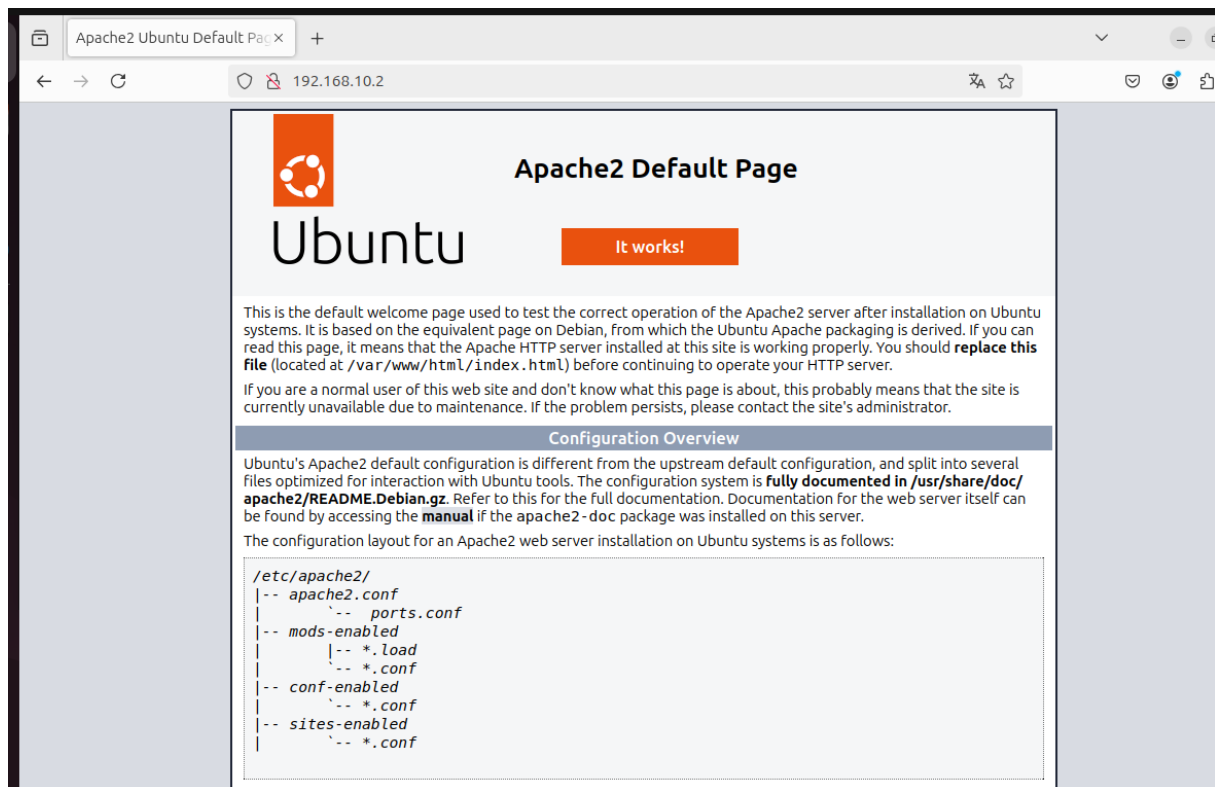
Machine Client :

```
dmz@dmz-VirtualBox:~$ ping 192.168.20.1
PING 192.168.20.1 (192.168.20.1) 56(84) bytes of data.
64 bytes from 192.168.20.1: icmp_seq=1 ttl=64 time=2.55 ms
64 bytes from 192.168.20.1: icmp_seq=2 ttl=64 time=0.879 ms
64 bytes from 192.168.20.1: icmp_seq=3 ttl=64 time=2.83 ms
64 bytes from 192.168.20.1: icmp_seq=4 ttl=64 time=1.94 ms
^C
--- 192.168.20.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3003ms
rtt min/avg/max/mdev = 0.879/2.047/2.829/0.747 ms
dmz@dmz-VirtualBox:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=254 time=20.5 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=254 time=17.9 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=254 time=18.6 ms
^C
--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 17.930/19.038/20.542/1.102 ms
```

Les 2 machines parviennent à envoyer des paquets.

## Accès aux ressources de la DMZ depuis le client

Essayons désormais d'accéder au serveur web dans la DMZ en passant par son adresse IP.



La machine cliente parvient bien à accéder au site de la DMZ. Seulement, la DMZ est actuellement connecté à internet, ce qui peut être problématique.

Retournons donc sur le pare-feu et modifions les règles.

Commençons donc par bloquer les connexions sortantes de la DMZ vers Internet avec la commande :

```
sudo iptables -A FORWARD -i enp0s8 -o enp0s3 -j DROP
```

```
dmz@dmz-VirtualBox:~$ sudo iptables -A FORWARD -i enp0s8 -o enp0s3 -j DROP
```

Autorisons ensuite les requêtes du client vers la DMZ avec les commandes :

```
sudo iptables -A FORWARD -i enp0s9 -o enp0s8 -p tcp --dport 80 -j ACCEPT
```

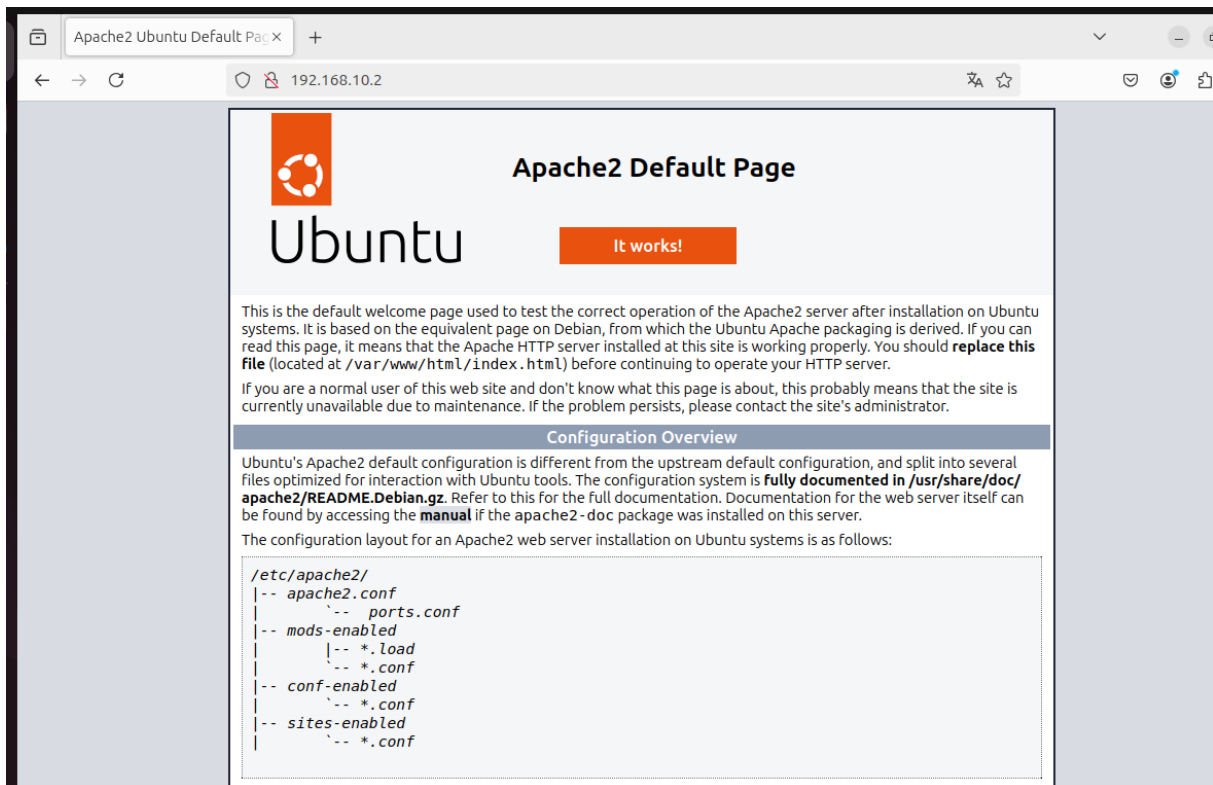
```
sudo iptables -A FORWARD -i enp0s9 -o enp0s8 -p tcp --dport 443 -j ACCEPT
```

```
sudo iptables -A FORWARD -i enp0s8 -o enp0s9 -m state --state ESTABLISHED,RELATED -j ACCEPT
```

```
dmz@dmz-VirtualBox:~$ sudo iptables -A FORWARD -i enp0s9 -o enp0s8 -p tcp --dport 80 -j ACCEPT
dmz@dmz-VirtualBox:~$ sudo iptables -A FORWARD -i enp0s9 -o enp0s8 -p tcp --dport 443 -j ACCEPT
dmz@dmz-VirtualBox:~$ sudo iptables -A FORWARD -i enp0s8 -o enp0s9 -m state --state ESTABLISHED,RELATED -j ACCEPT
```

Sauvegardons ces règles.

Maintenant, vérifions que l'on puisse toujours accéder au site.



Cela conclue donc la configuration du pare-feu.