

Mini-projet - Mise en place de Reverse Proxy Nginx

Encadrement	Professeur My. El Mehdi BOUHAMIDI
Réalisation	Yassine CHERRADI

UNIVERSITE CADI AYYAD
MARRAKECH



I. Définition :

serveur mandataire (n.m.)

1. dispositif informatique associé à un serveur et réalisant, pour des applications autorisées, des fonctions de médiation, telle que le stockage des documents les plus fréquemment demandés ou l'établissement de passerelles.

II. Types :

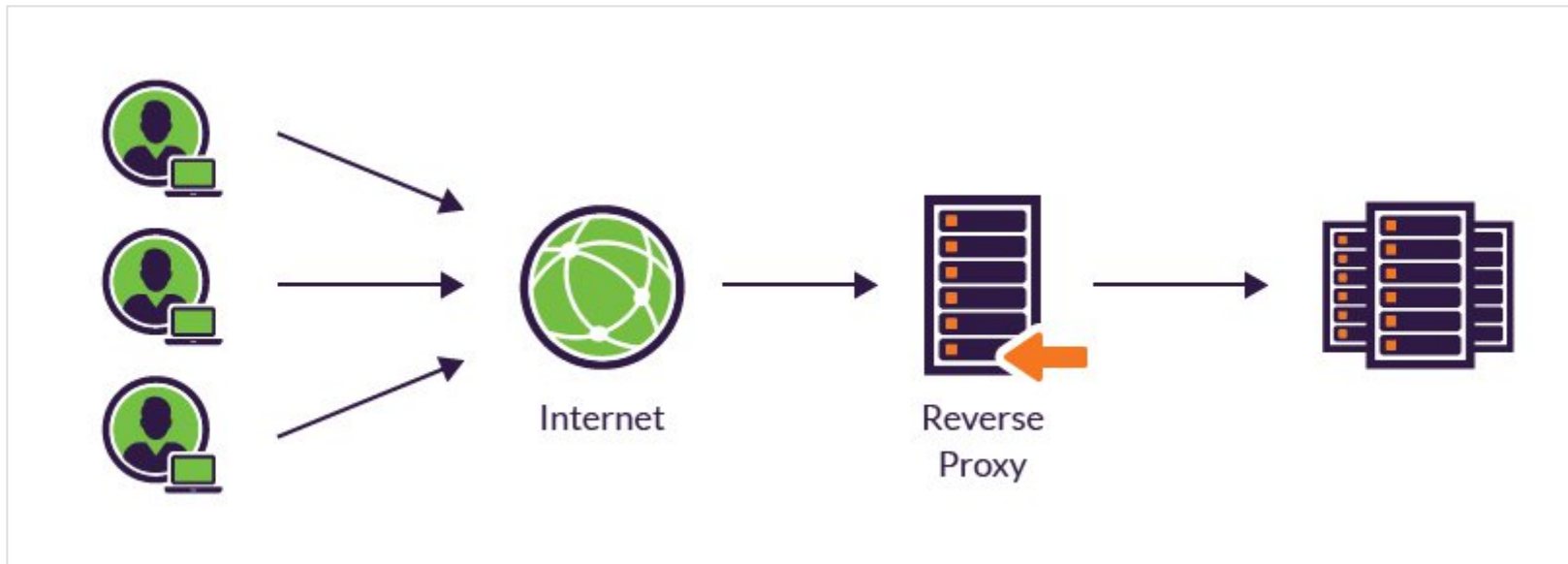
1. Reverse proxy :

Un Reverse proxy est un point de connexion intermédiaire situé à la périphérie d'un réseau. Il reçoit les demandes de connexion HTTP initiales, agissant comme le point d'extrémité réel.

Essentiellement le flic de trafic de votre réseau, le Reverse proxy sert de passerelle entre les utilisateurs et le serveur d'origine de votre application. Ce faisant, il assure la gestion de toutes les politiques et l'acheminement du trafic.

1.1. Fonctionnement :

1. Réception d'une demande de connexion d'un utilisateur
2. Achever une poignée de main TCP à trois voies, mettre fin à la connexion initiale
3. Connexion avec le serveur d'origine et transmission de la demande originale



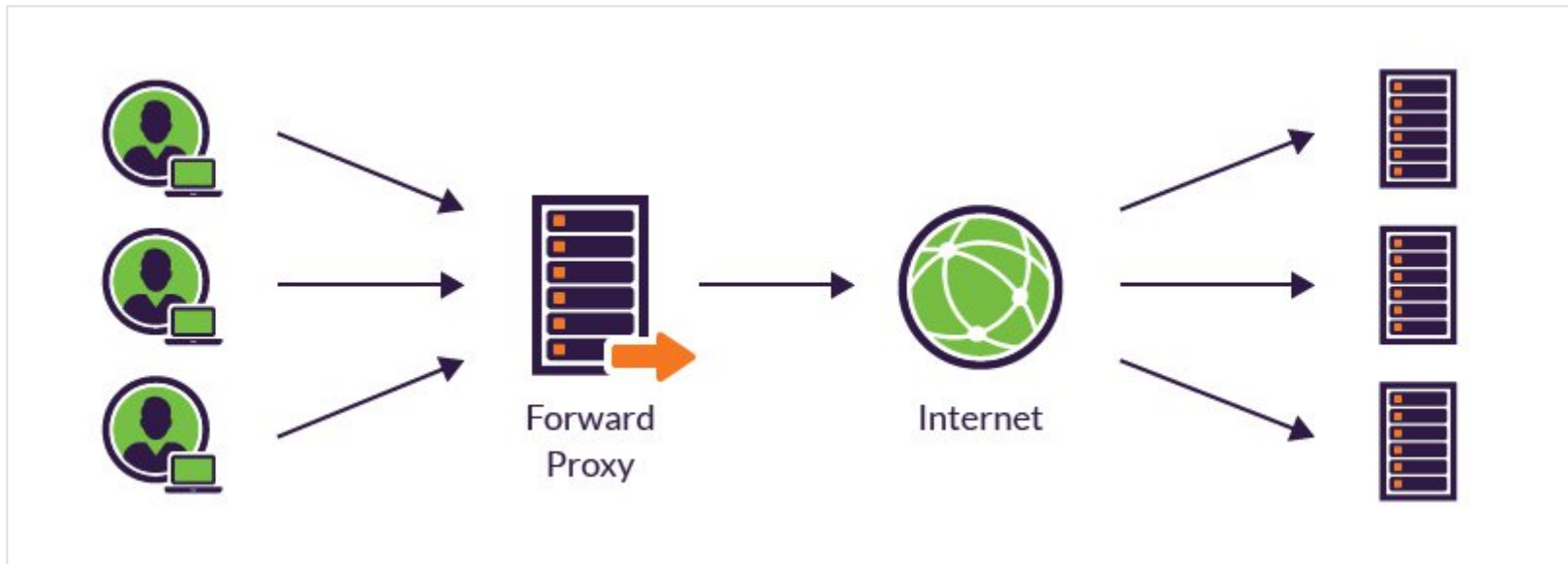
2. Forward proxy :

En revanche, un Forward proxy est également placé à la périphérie de votre réseau, mais il régule le trafic sortant en fonction de politiques prédéfinies dans les réseaux partagés. En outre, il dissimule l'adresse IP d'un client et bloque le trafic entrant malveillant.

Les Forward proxy sont généralement utilisés en interne par les grandes organisations, telles que les universités et les entreprises.

2.1. Usage :

- Empêcher les employés de visiter certains sites web
- Surveiller l'activité en ligne des employés
- Empêcher le trafic malveillant d'atteindre un serveur d'origine
- Améliorer l'expérience de l'utilisateur en mettant en cache le contenu des sites externes



IV. Atelier Pratique :

1. Environnement :

1.1. Caractéristiques :

Environnement de virtualisation	Environnement de déploiement	Adresse du réseau	Type d'adaptateur réseau de la VM	Adresse IP du Serveur
VirtualBox 6.1.10 Ubuntu	Ubuntu 16.04.7 LTS	192.168.58.0/24	Host-only Adapter, 'vboxnet' - Paravirtualized Network (virtio-net)	192.168.58.42

1.2. Dépendances :

```
sudo apt-get install apache2 php7.0 libapache2-mod-php7.0 nginx php7.0-fpm bind9
```

2. Configuration :

2.1. Réglage de l'interface réseau :

```
sudo vim /etc/network/interfaces
auto enp0s3
iface enp0s3 inet static
address 192.168.58.42
netmask 255.255.255.0
```

2.2. Apache :

```
mkdir /var/www/yacyncherradi.net
chown user:www-data /var/www/yacyncherradi.net
chmod 750 /var/www/yacyncherradi.net
mkdir /var/www/yacyncherradi.net/public_html
cd /var/www/yacyncherradi.net/public_html
echo "Yacyn CHERRADI's homepage !" > index.html
echo "<?php echo phpinfo(); " > info.php

vim /etc/apache2/ports.conf
<IfModule ssl_module>
    Listen 127.0.0.1:44344
</IfModule>

mkdir /etc/apache2/ssl
cd /etc/apache2/ssl
openssl req -x509 -nodes -days 90 -newkey rsa:2048 -keyout ssl.key -out ssl.crt

vim /etc/apache2/sites-available/yacyncherradi.net.conf
<VirtualHost 127.0.0.1:44344>
    ServerName 192.168.58.42
    ServerAdmin admin@yacyncherradi.net
    DirectoryIndex index.php index.html
    DocumentRoot /var/www/yacyncherradi.net/public_html

    SSLEngine on
    SSLCertificateFile /etc/apache2/ssl/ssl.crt
    SSLCertificateKeyFile /etc/apache2/ssl/ssl.key

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
```

```
</VirtualHost>
a2enmod ssl
a2dissite 000-default.conf
a2ensite yacyncherradi.net.conf
systemctl restart apache2

curl -Ik https://127.0.0.1:44344/ # k (--insecure) : turns off curl's verification of the certificate
wget --no-check-certificate https://127.0.0.1:44344/info.php
```

2.3. Nginx :

```
mkdir /etc/nginx/ssl
cd /etc/nginx/ssl
openssl req -x509 -nodes -days 90 -newkey rsa:2048 -keyout self.key -out self.crt

vim /etc/nginx/nginx.conf
user www-data;
worker_processes auto;

events {
    worker_connections 1024;
}

http {
    include mime.types;

    server {
        listen 80;
        server_name 192.168.58.42;
        return 301 https://$host$request_uri;
    }

    server {
        listen 443 ssl http2;
        ssl_certificate /etc/nginx/ssl/self.crt;
        ssl_certificate_key /etc/nginx/ssl/self.key;

        location / {
            proxy_pass "https://127.0.0.1:44344/";
        }
    }
}

nginx -t
```

```
systemctl restart nginx
```

2.4. DNS :

```
mkdir /etc/bind/yacyncherradi.net
cd /etc/bind/yacyncherradi.net
vim db.yacyncherradi.net
;
; BIND data file for yacyncherradi.net
;
$TTL      604800
@ IN      SOA      yacyncherradi.net. root.yacyncherradi.net. (
                        2          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL
;
@ IN      NS       ns.yacyncherradi.net.
@ IN      A        192.168.58.42
ns IN     A        192.168.58.42
www IN    CNAME    ns
```

```
vim db.192
;
; BIND reverse data file for yacyncherradi.net
;
$TTL      604800
@ IN      SOA      yacyncherradi.net. root.yacyncherradi.net. (
                        1          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL
;
@ IN      NS       ns.
42 IN     PTR      ns.yacyncherradi.net.
```

```
vim /etc/bind/named.conf.local
zone "yacyncherradi.net" {
    type master;
    file "/etc/bind/yacyncherradi.net/db.yacyncherradi.net";
```

```
};  
  
zone "58.168.192.in-addr.arpa" {  
    type master;  
    file "/etc/bind/yacyncherradi.net/db.192";  
};  
  
systemctl restart bind9  
ping yacyncherradi.net
```

2.5. Verification (@Client)

```
vim /etc/resolv.conf  
nameserver 192.168.58.42  
search yacyncherradi.net  
  
ping yacyncherradi.net
```

V. Sources :

1. [Sensagent Dictionnaires - définition, synonyme et encyclopédie : Proxy.](#)
2. [Imperva Learning Center : Reverse Proxy.](#)
3. [NGINX Fundamentals: High Performance Servers from Scratch](#)
4. [Apache HTTP Server Version 2.4 Documentation](#)
5. [nginx documentation](#)