Projet E5 – Mise en place d’un intranet avec certificat auto-signé

I. Préparation de l’environnement

1) Création de la VM

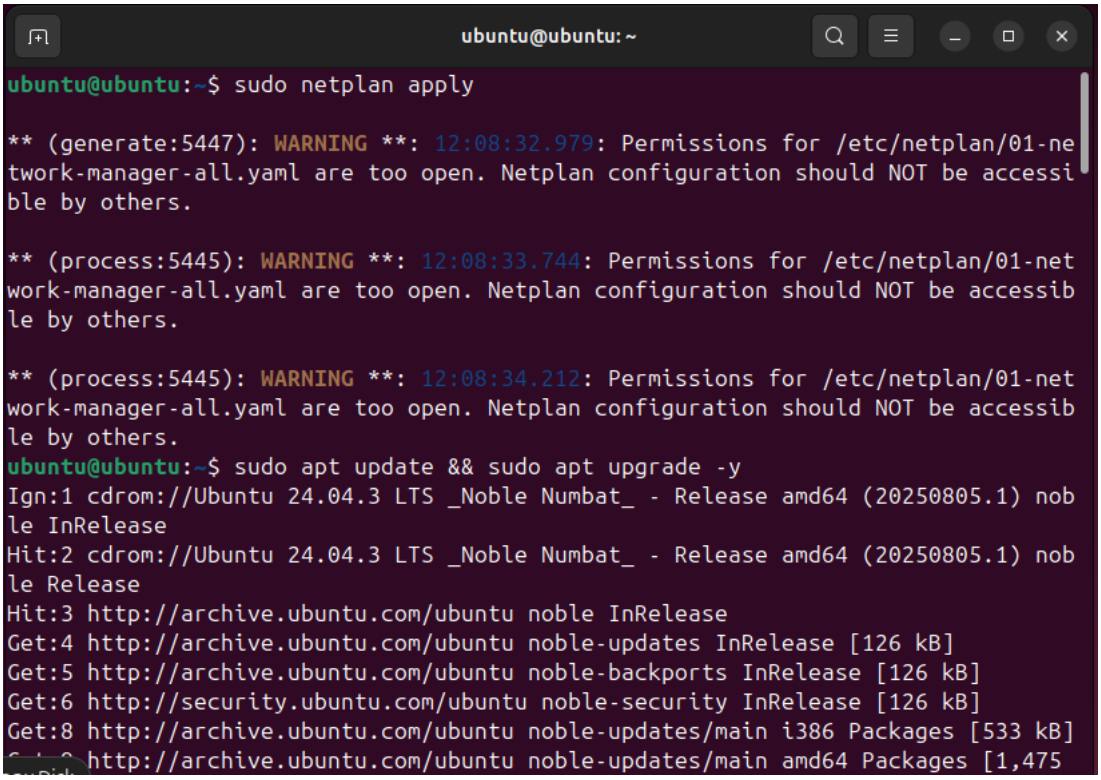
Voici la configuration qu’on va utiliser :

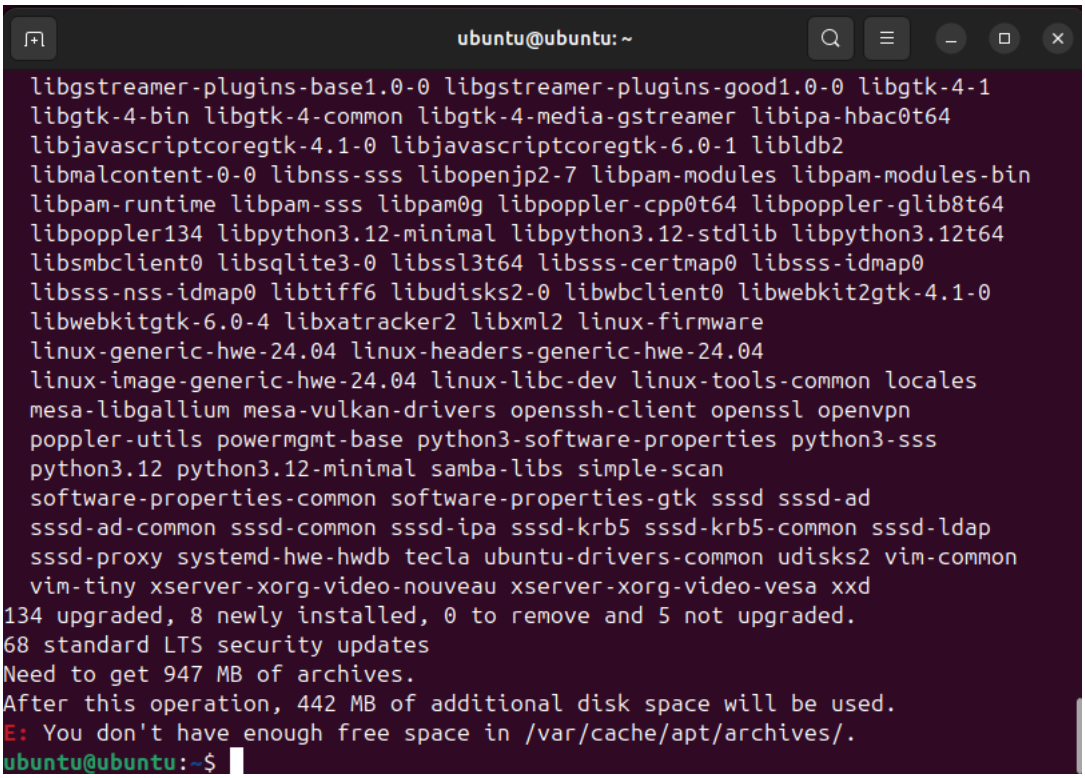
- 2 vCPU, 4 Go RAM, 20 Go disque dur
- OS : Ubuntu Server 22.04

2) Installation du système

Avec VMware Station, on a installé l’ISO Ubuntu avec comme nom de machine « intranet »

Avec la commande **« sudo netplan apply** » et **sudo apt update && sudo apt upgrade -y** », on va se mettre en mode administrateur puis mettre à jour le système.



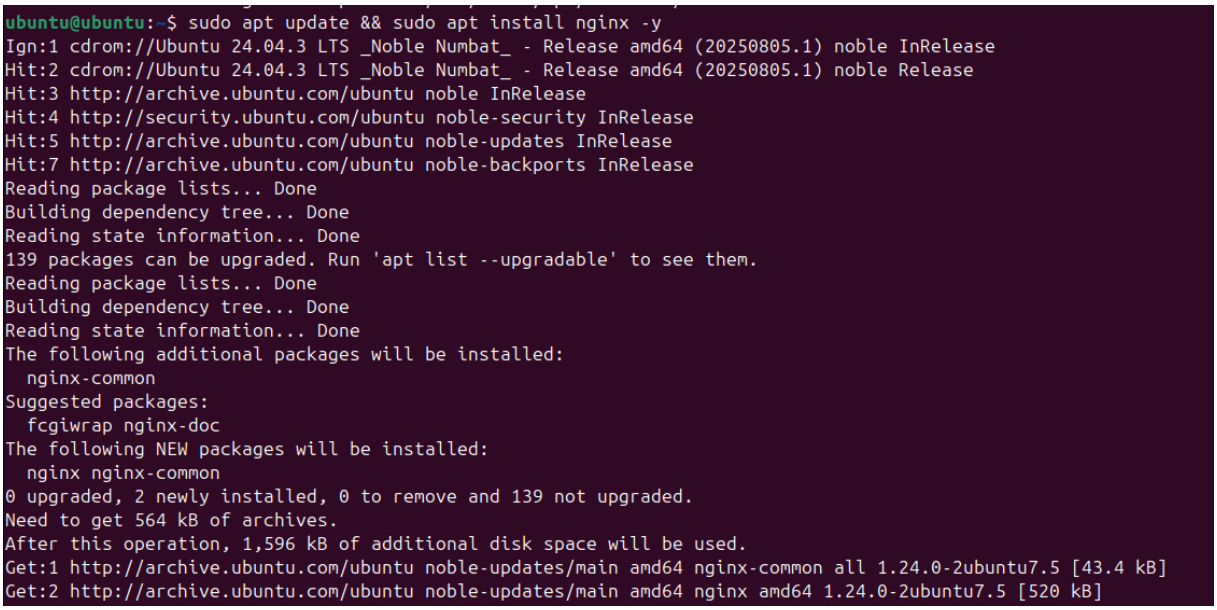


II. Installation du serveur web

Installer Nginx

Avec la commande « **sudo apt update && sudo apt install nginx -y** », nous allons installer Nginx pour installer le serveur web.

Pour savoir si le service tourne, on va utiliser la commande « **systemctl status nginx** »



```
ubuntu@ubuntu:~$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: en>
     Active: active (running) since Wed 2025-10-01 12:13:38 UTC; 16s ago
       Docs: man:nginx(8)
    Process: 7008 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_proce>
    Process: 7009 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (c>
   Main PID: 7039 (nginx)
      Tasks: 3 (limit: 2197)
     Memory: 4.0M (peak: 6.7M)
        CPU: 180ms
     CGroup: /system.slice/nginx.service
             ├─7039 "nginx: master process /usr/sbin/nginx -g daemon on; master>
             ├─7041 "nginx: worker process"
             └─7042 "nginx: worker process"

Oct 01 12:13:38 ubuntu systemd[1]: Starting nginx.service - A high performance >
Oct 01 12:13:38 ubuntu systemd[1]: Started nginx.service - A high performance w>
lines 1-17/17 (END)
```

III. Création du site intranet

1) Créer le dossier web

Après avoir installer Nginx, on va devoir créer le dossier web avec la commande :

sudo mkdir -p /var/www/intranet

sudo chown -R www-data:www-data /var/www/intranet

```
ubuntu@ubuntu:~$ sudo mkdir -p /var/www/intranet
ubuntu@ubuntu:~$ sudo chown -R www-data:www-data /var/www/intranet
ubuntu@ubuntu:~$
```

2) Création de la page d'accueil

On va également créer la page d'accueil avec cette commande :

echo "<h1>Bienvenue sur l'intranet (Nginx)</h1>" | sudo tee /var/www/intranet/index.html

```
ubuntu@ubuntu:~$ echo "<h1> Bienvenue sur l'intranet Nginx</h1>" | sudo tee /var/www/intranet/index.html
<h1> Bienvenue sur l'intranet Nginx</h1>
ubuntu@ubuntu:~$
```

3) Créer un bloc serveur http

Pour créer un bloc serveur http, on va devoir utiliser cette commande : sudo nano /etc/nginx/sites-available/intranet

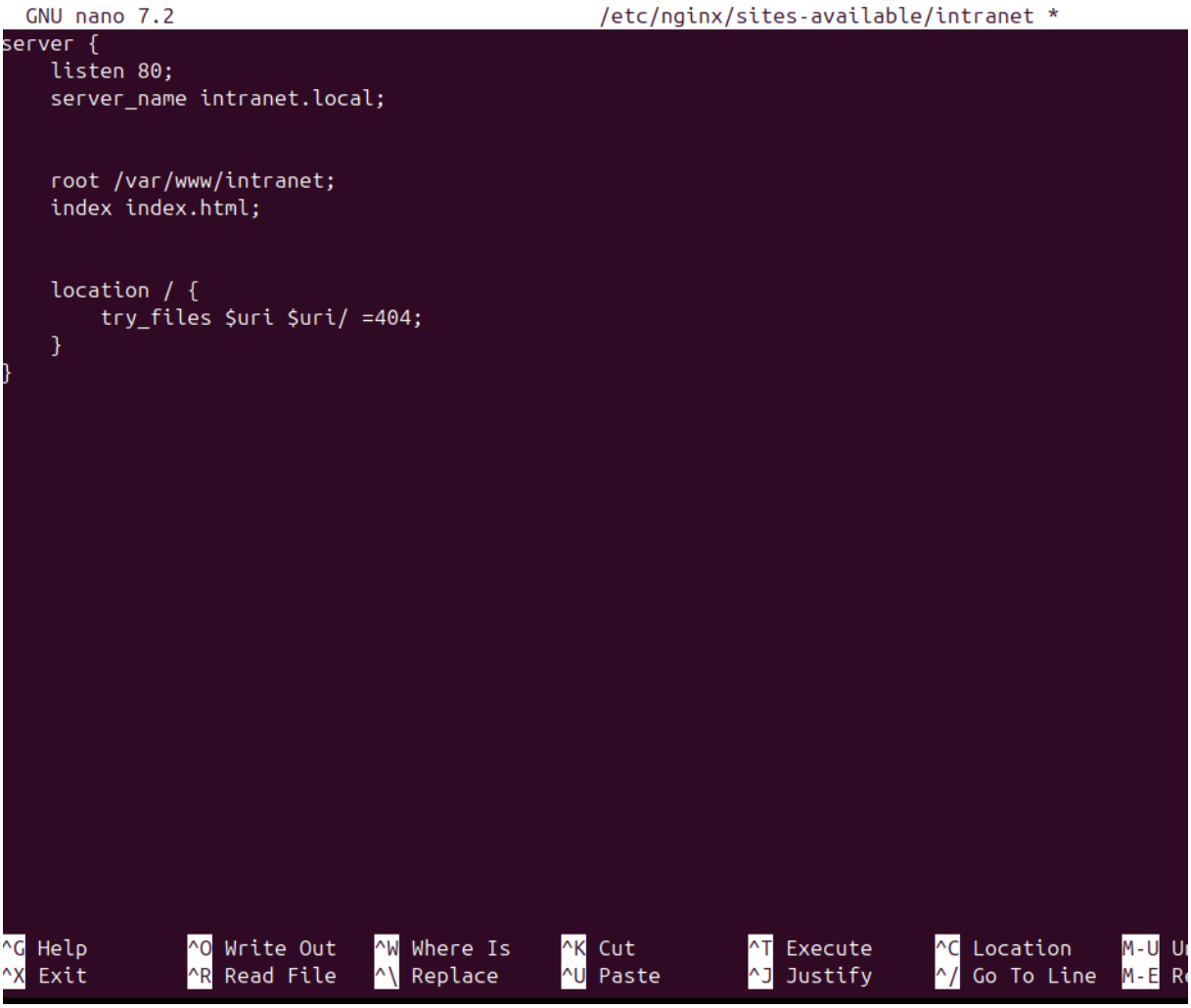
```
ubuntu@ubuntu:~$ sudo nano /etc/nginx/sites-available/intranet
```

Cela ramener à un « bloc note » dans lequel on va devoir écrire la commande ci-dessous :

```
server {
    listen 80;
    server_name intranet.local;
```

```
    root /var/www/intranet;
    index index.html;

    location / {
        try_files $uri $uri/ =404;
    }
}
```



Contrôle X pour quitter et sauvegarder avec « Y » puis entrée, on retourne sur bash

On va activer le site avec la commande : sudo ln -s /etc/nginx/sites-available/intranet /etc/nginx/sites-enabled/

sudo nginx -t

sudo systemctl reload nginx

```
ubuntu@ubuntu:~$ sudo ln -s /etc/nginx/sites-available/intranet /etc/nginx/sites-enabled/
ubuntu@ubuntu:~$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
ubuntu@ubuntu:~$ sudo systemctl reload nginx
```

IV. Sécurisation avec HTTPS

Afin qu'il soit auto-certifié, on va crée un dossier pour le certificat pour qu'il soit généré avec la commande :

sudo mkdir /etc/ssl/intranet

puis

sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 \

-keyout /etc/ssl/intranet/intranet.key \

-out /etc/ssl/intranet/intranet.crt

```
ubuntu@ubuntu:~$ sudo mkdir /etc/ssl/intranet
ubuntu@ubuntu:~$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 \
> -keyout /etc/ssl/intranet/intranet.key \
> -out /etc/ssl/intranet/intranet.crt
.....+....+...+..+.+.....+...+..+.....+.....+...+.....+...+...+.....+.....+..
.....+..+.....+.....+...+..+....+..+...+.....+...+.+.....+.++++++++++++++++
++++++++++++++++*.++++++++++++++++++++++++++++++++++++++++++++++++++++++++*.....+..
.....+...+.....+..+....+.....+.+..+...+....+.....++++++++++++++++++++++++++++++++++
....+.+.............+...+...+....+.....+.....+...+.....+.....+.....+.+++++++++
++++++++++++++++++++*.....+...+.....++++++++++++++++++++++++++++++++++++++++++++++
....+..+.....+...+...+.....+....+.....+.+..+....+.....+...+..+.....
..+.....+.....+.....+.....+..+.....+...+.+...+...+.....+.....+
.+.....+.....+.....+..+...+.+.....+.....+...+.....+...+..+...+....+...+.....
.....+..+.+.....+....+...+..+.....+....+.....+...+..+.....+.....+.+..
.....+....+...+.....++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:
```

Common Name (CN) : intranet.local

V. Création d'un bloc serveur HTTPS

Dans bash, nous allons mettre cette commande :

sudo nano /etc/nginx/sites-available/intranet-ssl

```
ubuntu@ubuntu:~$ sudo nano /etc/nginx/sites-available/intranet-ssl
```

Puis, on va écrire ceci :

server {

    listen 443 ssl;

```
    server_name intranet.local;

    root /var/www/intranet;
    index index.html;

    ssl_certificate /etc/ssl/intranet/intranet.crt;
    ssl_certificate_key /etc/ssl/intranet/intranet.key;

    location / {
        try_files $uri $uri/ =404;
    }
}
```

```
 GNU nano 7.2                                /etc/nginx/sites-available/intranet-ssl *
server {
    listen 443 ssl;
    server_name intranet.local;


    root /var/www/intranet;
    index index.html


    ssl_certificate /etc/ssl/intranet/intranet.crt;
    ssl_certificate_key /etc/ssl/intranet/intranet.key;


    location / {
        try_files $uri $uri/ =404;
    }
}
```

Puis, on va l'activer avec cette commande : sudo ln -s /etc/nginx/sites-available/intranet-ssl /etc/nginx/sites-enabled/

sudo nginx -t

sudo systemctl reload nginx

```
ubuntu@ubuntu:~$ sudo ln -s /etc/nginx/sites-available/intranet-ssl /etc/nginx/sites-enabled/
ln: failed to create symbolic link '/etc/nginx/sites-enabled/intranet-ssl': File exists
ubuntu@ubuntu:~$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
ubuntu@ubuntu:~$ sudo systemctl reload nginx
ubuntu@ubuntu:~$
```

On va ensuite rediriger le HTTP en HTTPS dans bash en modifiant le fichier intranet :

server {

    listen 80;

    server_name intranet.local;

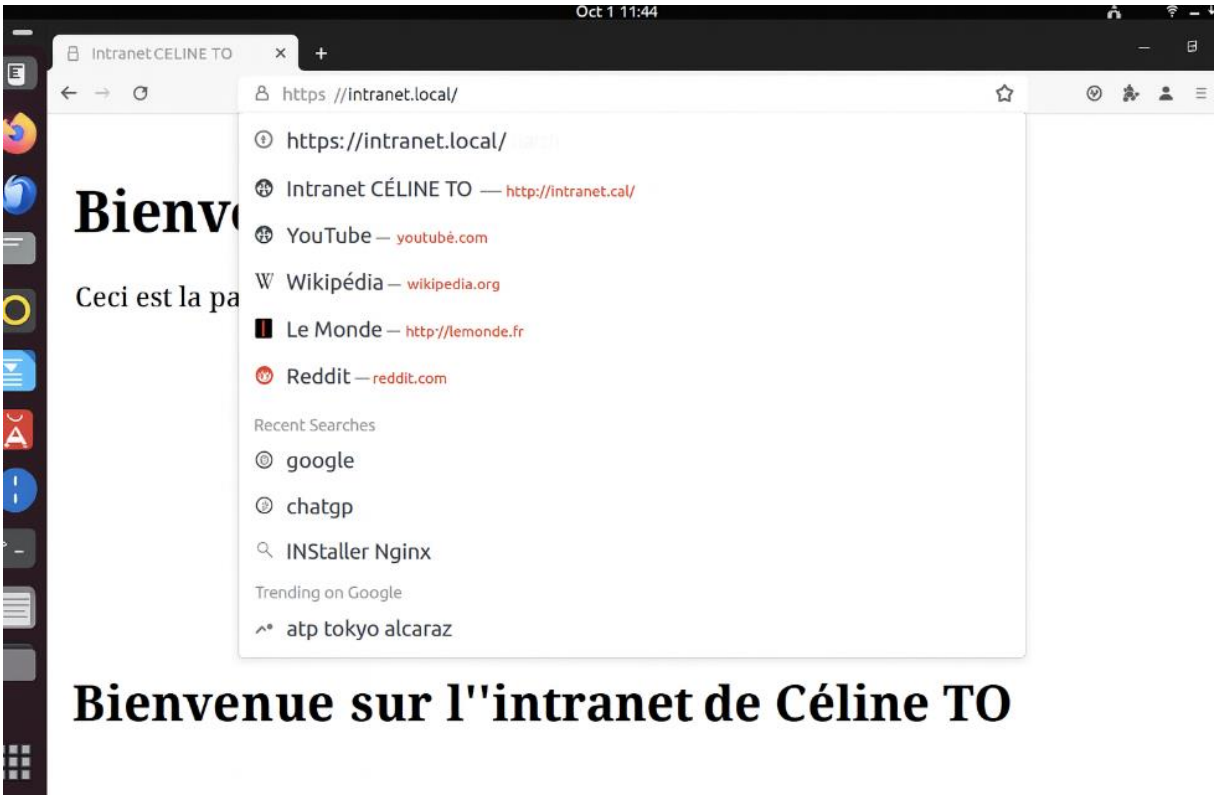
    return 301 https://$server_name$request_uri;

}

On recharge Nginx :

Test du site

On va accéder à : https://intranet.local et une alerte de sécurité apparaît (normal avec auto-signé). Après validation, la page intranet s'affiche en HTTP


Oct 1 11:44
Intranet CELINE TO
https //intranet.local/
https://intranet.local/
Intranet CÉLINE TO — http://intranet.cal/
YouTube — youtubé.com
Wikipédia — wikipedia.org
Le Monde — http://lemonde.fr
Reddit — reddit.com
Recent Searches
google
chatgp
INStaller Nginx
Trending on Google
atp tokyo alcaraz
Ceci est la pa
Bienvenue sur l''intranet de Céline TO