ESICAD

Sécurisation des accès distants

SOMMAIRE

1. Contexte et cahier des charges

Notre client souhaite mettre en place un site web sécurisé (serveur web et base de données). Un seul utilisateur pourrait accéder, déposer et modifier des fichiers sur le serveur web.

Il veut également une zone sécurisée pour éviter les attaques extérieures malveillantes en passant par le site web.

Puis, une solution doit être mise en place pour que des fichiers de différents utilisateurs soient partagés et accessibles facilement.

Enfin, il souhaiterait mettre à la disposition de ses employés une solution d'accès à distance sécurisée à son réseau interne.

Les employés doivent pouvoir accéder au serveur de fichier et avoir les mêmes autorisations qu'en étant connecté dans l'entreprise.

Le serveur web doit également être accessible en connexion VPN.

Une des seules contraintes matérielles imposée est l'utilisation d'un routeur virtuel.

Pour cela, nous allons créer une zone démilitarisée (DMZ) dans laquelle nous mettrons le serveur WEB, FTP, DNS. Une DMZ, littéralement une « zone démilitarisée » en anglais est une zone spéciale exposée. C'est vers elle que circule tous les informations qui vont vers internet. Cela sert à protéger le réseau contre les menaces extérieures.

Nous aurons un deuxième réseau, LAN, dans lequel se trouvera le serveur de base de données, le serveur de fichiers et un poste administrateur.

Nous allons utiliser comme routeur virtuel PFsense.

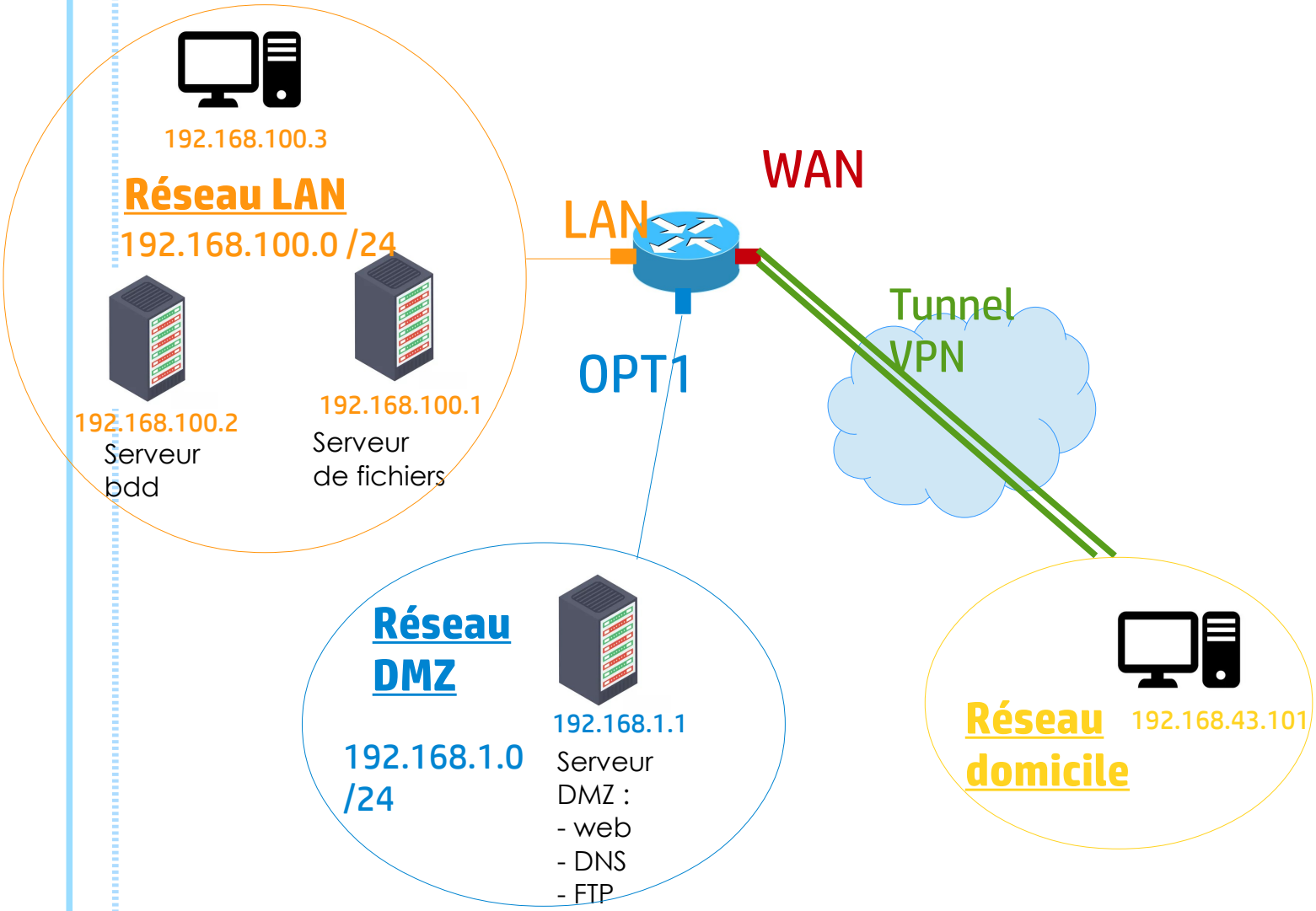
Nous utiliserons OpenVPN comme solution pour se connecter de manière sécurisée au réseau interne depuis l'extérieur.

1.1. Objectifs

- Mettre en place et configurer un routeur virtuel
- Tout le monde peut accéder au serveur web

- Seul le poste serveur où se trouve le serveur web a accès à la bdd
- Les réseaux LAN et DMZ ont accès à internet
- Accès au FTP seulement par le poste administrateur du réseau LAN
- Permettre aux clients nomades ou en télétravail d'accéder aux services hébergés dans un réseau interne à l'entreprise dans une LAN et/ou une DMZ
- Mettre en place des tests de validation répondant aux demandes

1.2. Architecture



2. Mise en œuvre

2.1. Poste serveur "DMZ"

2.1.1. Configuration IP

Au début du projet, ce poste comporte 2 cartes réseaux, une pour aller sur internet et donc pouvoir télécharger les paquets nécessaire et l'autre qui est dans le même réseau que l'interface OPT1 du routeur virtuel.
Lorsque le routeur virtuel est configuré, l'utilisation de la première carte réseau n'est plus nécessaire, donc nous l'enlevons.

- Fichier de configuration des cartes réseaux

```
  GNU nano 3.2                    /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

auto eth0
iface eth0 inet static
        address 192.168.43.3
        netmask 255.255.255.0
        gateway 192.168.43.1

auto eth1
iface eth1 inet static
        address 192.168.1.1
        netmask 255.255.255.0
```

- Vérification des adresses

```
root@dmz:~# 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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:15:5d:38:01:2c brd ff:ff:ff:ff:ff:ff
    inet 192.168.43.30/24 brd 192.168.43.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe38:12c/64 scope link
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 00:15:5d:38:01:2d brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.1/24 brd 192.168.1.255 scope global eth1
       valid_lft forever preferred_lft forever
    inet6 fe80::215:5dff:fe38:12d/64 scope link
       valid_lft forever preferred_lft forever
```

→ ok

2.1.2. Paramétrage Apache 2

Nous avons au préalable téléchargé le paquet Apache2 et vérifié son bon fonctionnement.

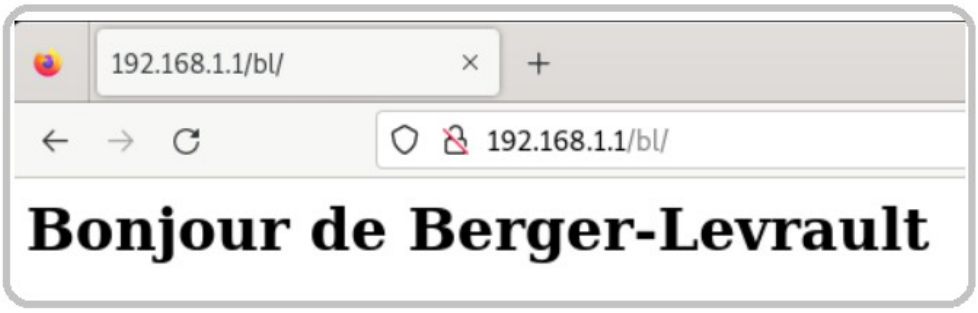
Nous allons donc à présent mettre en place le serveur web composé d'un site web.

- Dans le dossier html, le fichier *index.html* est renommé et un dossier du nom du site est créé, ici bl.

```
root@dmz:/var/www/html# ls -l
total 16
drwxr-xr-x 2 root root  4096 oct.  30 09:59 bl
-rw-r--r-- 1 root root 10701 oct.  30 09:57 index.html.old
```

- Dans ce dossier, un fichier *index.html* est rajouté. C'est dans ce dossier que tous les éléments du site internet vont être mis.

– Vérification de l'accès au site avec un navigateur



→ ok

2.1.3. Configuration du serveur DNS

Le DNS (Domain Name System) a pour fonction principale la traduction des noms de domaines en adresses IP. Il fait correspondre un nom de domaine avec une adresse IP que ce soit pour un site internet mais aussi pour des ordinateurs internes au réseau local.

Pour notre projet, Bind9 et dnsutils ont été installés au préalable sur le poste.

– Le nom FQDN du serveur est renseigné

– Dans le fichier *hosts*, nous entrons une ligne "localhost", une ligne avec le nom FQDN et l'ip localhost et enfin une ligne avec le nom FQDN et l'ip de notre serveur. Cela permet d'associer l'adresse IPV4 du serveur au nom FQDN.

```
 GNU nano 3.2                                  /etc/hosts

127.0.0.1       localhost
127.0.1.1       dmz.bl.lan
192.168.1.1     dmz.bl.lan
```

- Nous indiquons le domaine et la zone de recherche dans le fichier *résolv.conf*

```
 GNU nano 3.2                                /etc/resolv.conf

search bl.lan
domain bl.lan
nameserver 192.168.1.1
```

- Déclaration de la zone DNS "bl.lan" et sa zone inverse pour que les adresses IP puissent être traduites en noms de domaines

```
 GNU nano 3.2                                /etc/bind/named.conf.local

//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";

zone "bl.lan" {
        type master;
        file "etc/bind/db.bl.lan";
        notify yes;
};

zone "1.168.192.in-addr.arpa" {
        type master;
        file "/etc/bind/db.1.168.192.in-addr.arpa";
};
```

- Dans le dossier bind, création des fichiers de la zone directe et la zone inversée. Modification des deux fichiers

- zone directe :

```
  GNU nano 3.2                                                  db.bl.lan

$TTL    10800
$ORIGIN bl.lan.
@       IN SOA dmz.bl.lan. root.dmz.bl.lan (
        20160505;
        3h;
        1h;
        1w;
        1h)
;
@       IN NS   dmz.bl.lan.
dmz IN A 192.168.1.1
www IN A 192.168.1.1
localhost IN A 127.0.0.1
```

- zone inversée :

```
  GNU nano 3.2                                        db.1.168.192.in-addr.arpa

$TTL 10800
$ORIGIN 1.168.192.in-addr.arpa.
@       IN SOA dmz.bl.lan. root.bl.lan. (
        20160505;
        3h;
        1h;
        1w;
        1h);
@       IN NS   dmz.bl.lan.
1       IN PTR  dmz.bl.lan.
```

- Vérification de la syntaxe des fichiers de configuration *named.conf*. L'article "-z" permet de trouver et vérifier les zones DNS.

```
root@dmz:/etc/bind# named-checkconf -z
zone bl.lan/IN: loaded serial 20160505
zone 1.168.192.in-addr.arpa/IN: loaded serial 20160505
zone localhost/IN: loaded serial 2
zone 127.in-addr.arpa/IN: loaded serial 1
zone 0.in-addr.arpa/IN: loaded serial 1
zone 255.in-addr.arpa/IN: loaded serial 1
```

→ ok

- Vérification du bon fonctionnement du serveur DNS

  - zone directe

```
root@dmz:/# dig dmz.bl.lan

; <<>> DiG 9.11.5-P4-5.1+deb10u9-Debian <<>> dmz.bl.lan
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 56639
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 5be15976f34916d6dd703f0d653f8de4de95d376bce1eb0d (good)
;; QUESTION SECTION:
;dmz.bl.lan.                    IN      A

;; ANSWER SECTION:
dmz.bl.lan.             10800   IN      A       192.168.1.1

;; AUTHORITY SECTION:
bl.lan.                 10800   IN      NS      dmz.bl.lan.

;; Query time: 0 msec
;; SERVER: 192.168.1.1#53(192.168.1.1)
;; WHEN: lun. oct. 30 12:05:08 CET 2023
```

→ ok

  - zone inversée

```
root@dmz:/# dig -x 192.168.1.1

; <<>> DiG 9.11.5-P4-5.1+deb10u9-Debian <<>> -x 192.168.1.1
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 64987
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: c191d0123f723be518028f71653f8ea2a2118245315819a9 (good)
;; QUESTION SECTION:
;1.1.168.192.in-addr.arpa.      IN      PTR

;; ANSWER SECTION:
1.1.168.192.in-addr.arpa. 10800 IN      PTR     dmz.bl.lan.

;; AUTHORITY SECTION:
1.168.192.in-addr.arpa. 10800   IN      NS      dmz.bl.lan.

;; ADDITIONAL SECTION:
dmz.bl.lan.             10800   IN      A       192.168.1.1
```

- Vérification du bon fonctionnement de notre serveur DNS : ping vers le nom de notre site internet (DNS)

```
root@dmz:/# ping www.bl.lan
PING www.bl.lan (192.168.1.1) 56(84) bytes of data.
64 bytes from dmz.bl.lan (192.168.1.1): icmp_seq=1 ttl=64 time=0.019 ms
64 bytes from dmz.bl.lan (192.168.1.1): icmp_seq=2 ttl=64 time=0.042 ms
64 bytes from dmz.bl.lan (192.168.1.1): icmp_seq=3 ttl=64 time=0.044 ms
^C
--- www.bl.lan ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 31ms
rtt min/avg/max/mdev = 0.019/0.035/0.044/0.011 ms
```

→ ok

2.1.4. Mise en place du protocole HTTPS

Le protocole HTTPS, « Hypertext Transfer Protocol Secure », permet une communication entre le client web et le serveur web chiffrée, donc plus sécurisée qu'avec le protocole HTTP. Ce chiffrement se fait par une extension du protocole TCP, le TLS (anciennement SSL).
Le serveur web, en HTTPS, a un certificat d'authentification qui est envoyé au client web lorsque ce dernier se connecte au site, pour qu'il se connecte au site et que la crédibilité du domaine soit attestée.

Sur notre poste, openssl est installé.

- Activation du ssl au niveau d'Apache2

```
root@dmz:/etc/apache2/sites-available# a2enmod ssl
Considering dependency setenvif for ssl:
Module setenvif already enabled
Considering dependency mime for ssl:
Module mime already enabled
Considering dependency socache_shmcb for ssl:
Enabling module socache_shmcb.
Enabling module ssl.
See /usr/share/doc/apache2/README.Debian.gz on how to configure SSL and create self-signed certificates.
To activate the new configuration, you need to run:
  systemctl restart apache2
```

- génération clé privée et certificat auto-signé (clé publique) avec une validité de vie d'1 an pour les besoins du projet dans un sous-dossier ssl créé dans le dossier apache2

```
root@dmz:/etc/apache2/ssl# openssl req -new -x509 -keyout /etc/apache2/ssl/apache.key -days 365 -nodes -out
 /etc/apache2/ssl/apache.crt
Generating a RSA private key
.....+++++
.....+++++
writing new private key to '/etc/apache2/ssl/apache.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
```

Dans une situation d'entreprise, nous n'aurions pas utilisé des certificats auto-signés mais serions passé par un organisme tel que "Let's ENCRYPT" pour rendre les certificats viables (service payant).

Configuration des Virtual Host sur les ports 80 et 443

- Nous copions le fichier " " et le renommons avec le nom de notre domaine. Nous le configurons ensuite pour que les personnes allant sur le site soit automatiquement redirigé vers le site en https.
  Nous notons également les paramétrages pour le port 443.

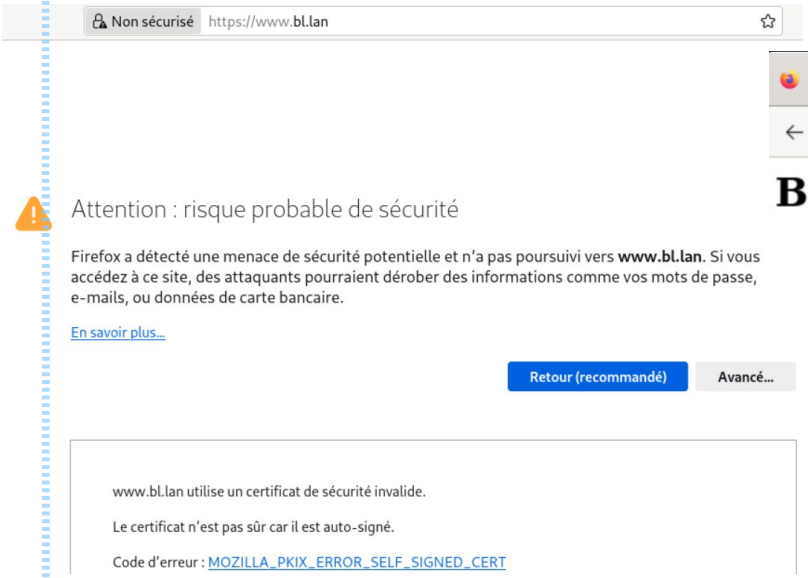
```
  GNU nano 3.2                                    bl.conf

<VirtualHost *:80>
        # The ServerName directive sets the request scheme, hostname and port that
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        ServerName www.bl.lan

        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html

        Redirect Permanent / https://www.bl.lan/

        # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
        # error, crit, alert, emerg.
        # It is also possible to configure the loglevel for particular
        # modules, e.g.

         # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
         # error, crit, alert, emerg.
         # It is also possible to configure the loglevel for particular
         # modules, e.g.
         #LogLevel info ssl:warn

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

         # For most configuration files from conf-available/, which are
         # enabled or disabled at a global level, it is possible to
         # include a line for only one particular virtual host. For example the
         # following line enables the CGI configuration for this host only
         # after it has been globally disabled with "a2disconf".
```

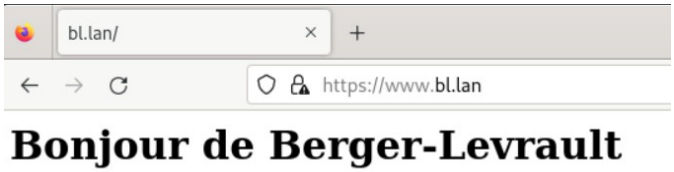
```
<VirtualHost *:443>

        ServerName www.bl.lan:443
        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html/bl

        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
        SSLEngine on
        SSLCertificateFile      /etc/apache2/ssl/apache.crt
        SSLCertificateKeyFile   /etc/apache2/ssl/apache.key

</VirtualHost>
```

- Activation du site "bl"

```
root@dmz:/etc/apache2/sites-available# a2ensite bl.conf
Enabling site bl.
To activate the new configuration, you need to run:
  systemctl reload apache2
root@dmz:/etc/apache2/sites-available# systemctl reload apache2
root@dmz:/etc/apache2/sites-available#
```

- Vérification : accès au site "bl.lan" en https sur un navigateur web





→ le « risque probable de sécurité » qui apparaît lors de l'entrée dans le site est normal puisque nous avons un certificat auto-signé. Il n'est donc pas reconnu par les navigateurs comme étant sécurisé.
→ ok

2.1.5. Serveur FTP

Le protocole FTP est prévu pour pouvoir réaliser des uploads et des downloads, c'est a dire pouvoir transférer des fichiers vers un serveur ou pour les télécharger par exemple.

- Installation de proFTPD

- Vérification du statut

```
root@dmz:/# systemctl status proftpd.service
● proftpd.service - LSB: Starts ProFTPD daemon
   Loaded: loaded (/etc/init.d/proftpd; generated)
   Active: active (running) since Mon 2023-10-30 11:37:49 CET; 40min ago
     Docs: man:systemd-sysv-generator(8)
  Process: 592 ExecStart=/etc/init.d/proftpd start (code=exited, status=0/SUCCESS)
    Tasks: 1 (limit: 4689)
   Memory: 12.3M
   CGroup: /system.slice/proftpd.service
           └─691 proftpd: (accepting connections)
```

→ ok

- Configuration FTP dans /etc/proftpd/conf.d/ sur le fichier *ftp-perso.conf*. Nous avons créé ce fichier personnalise au lieu de modifier directement le fichier *proftpd.conf* pour éviter que la configuration ne soit écrasée par une mise a jour proFTPD.

```
 GNU nano 3.2                                                ftp-perso.conf

#Nom du serveur
ServerName "dmz.bl.lan"

#Message de connexion

DisplayLogin "La connexion au serveur FTP s'est bien effectuéee"

#Désactiver IPV6
UseIPV6 off

#Spécifie le répertoire FTP auquel l'utilisateur est autorisé à accéder
DefaultRoot état

#Autoriser la connexion seulement aux membres du groupe "ftpadmin"
<Limit LOGIN>
DenyGroup !ftpadmin
</Limit>
```

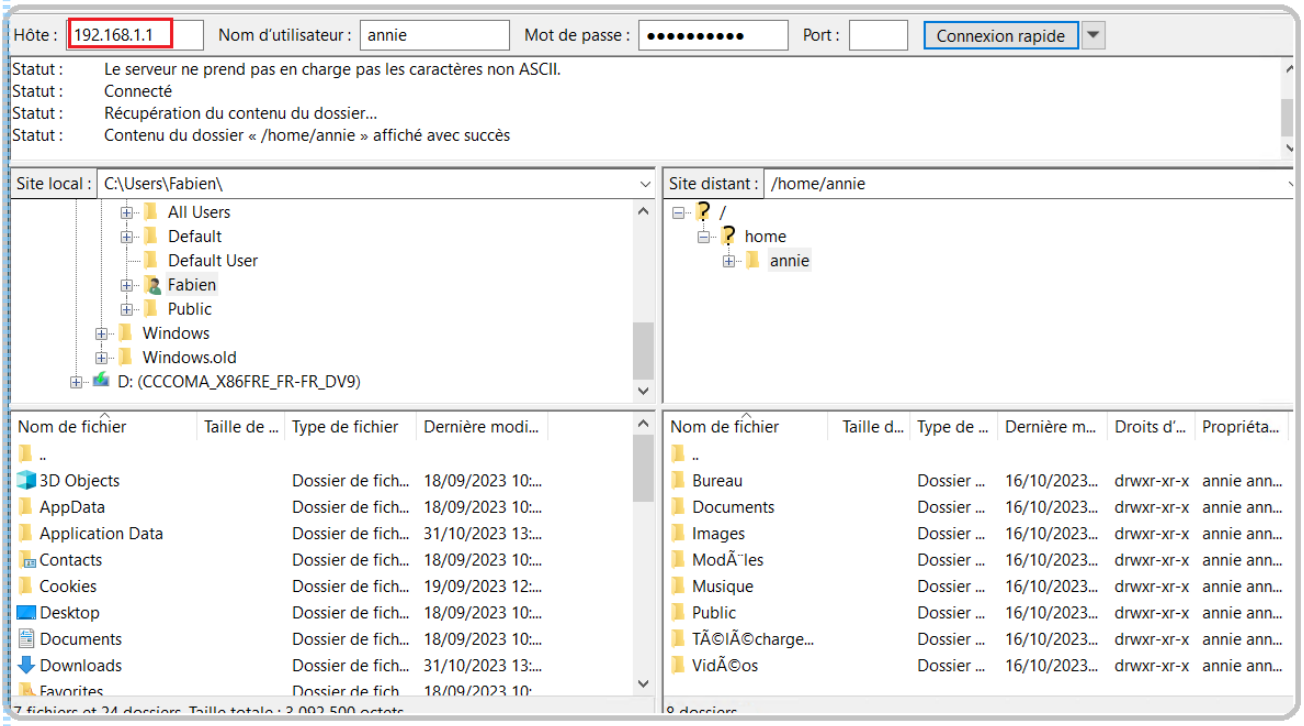
- Création du groupe 'ftpadmin'

```
root@dmz:/etc/proftpd/conf.d# addgroup ftpadmin
Ajout du groupe « ftpadmin » (GID 1002)...
Fait.
root@dmz:/etc/proftpd/conf.d#
```

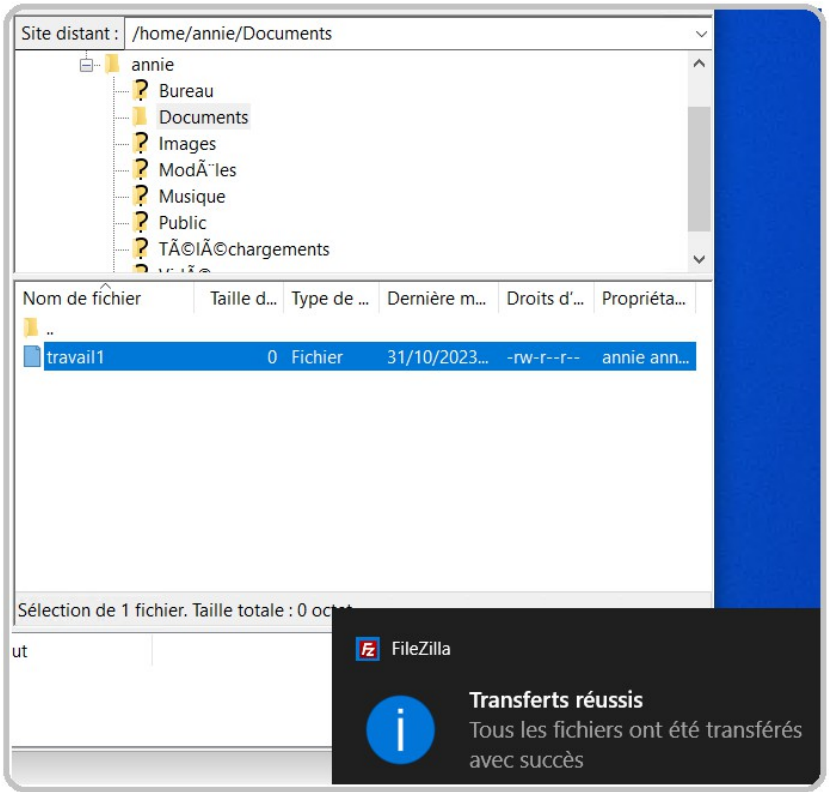
- Ajout de l'utilisateur au groupe 'ftpadmin'

```
root@dmz:~# adduser annie ftpadmin
Ajout de l'utilisateur « annie » au groupe « ftpadmin »...
Adding user annie to group ftpadmin
Fait.
```

- Test de connexion à la session administrateur depuis un client FTP, ici Filezilla



- Vérification de transfert d'un fichier



2.1.6. Php pour Apache2

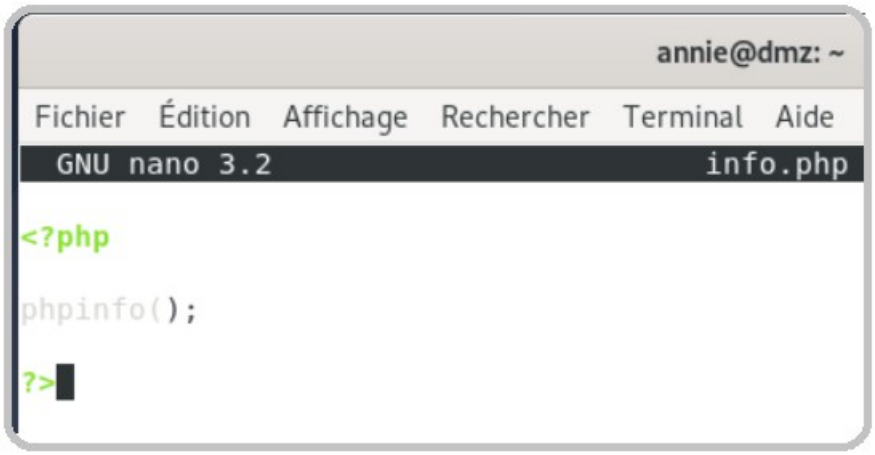
Nous installons php car le site va utiliser une bdd. Pour que le site puisse utiliser les éléments de la bdd, il faut utiliser le langage php.

- Installation de php pour Apache2

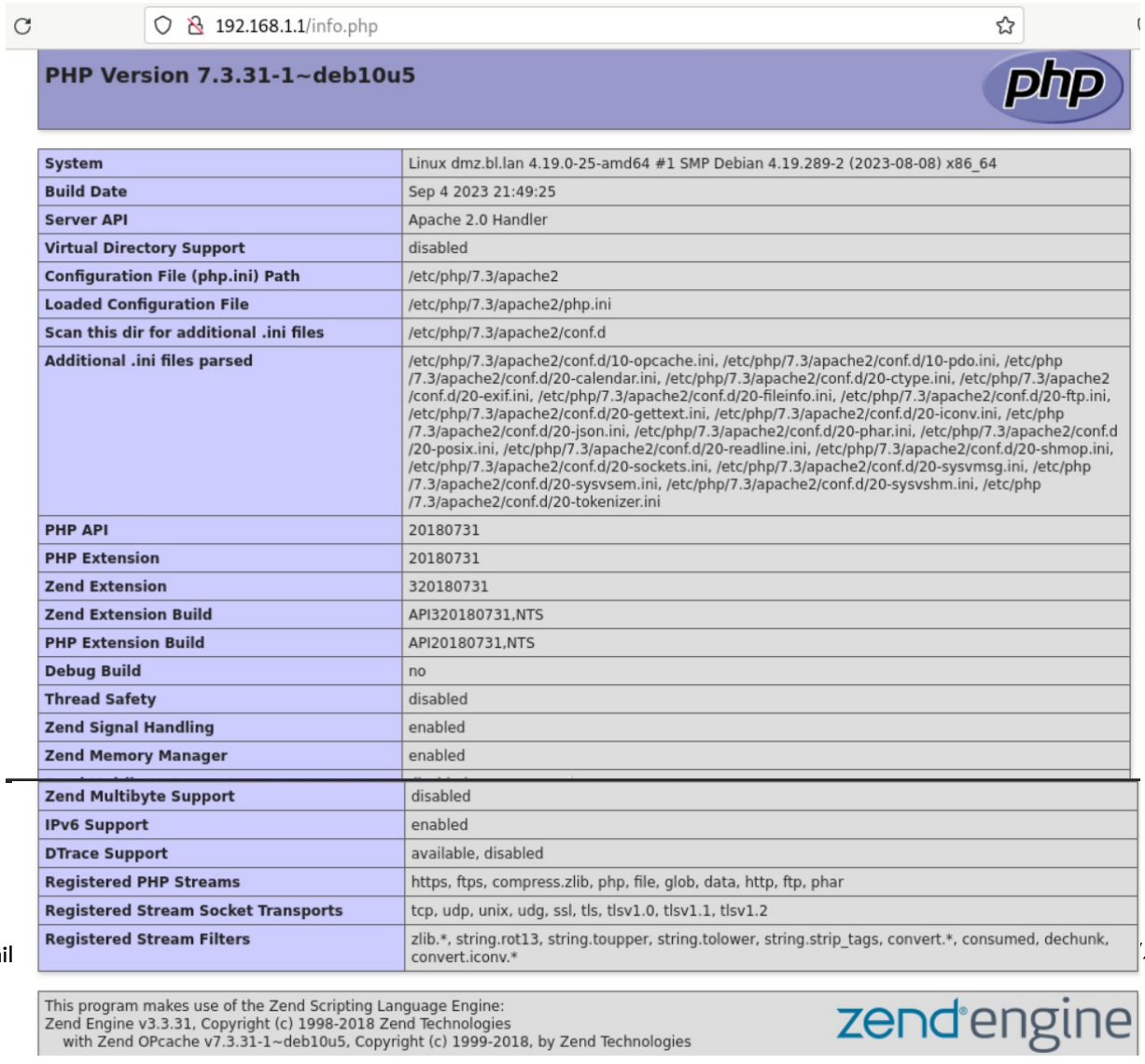
```
root@dmz:~# apt install php libapache2-mod-php
```

```
root@dmz:/# apt install php-mysql
```

- Vérification de la bonne installation de php
  - création d'un fichier info.php dans /var/www/html/ avec lignes :



- Ouverture dans le navigateur

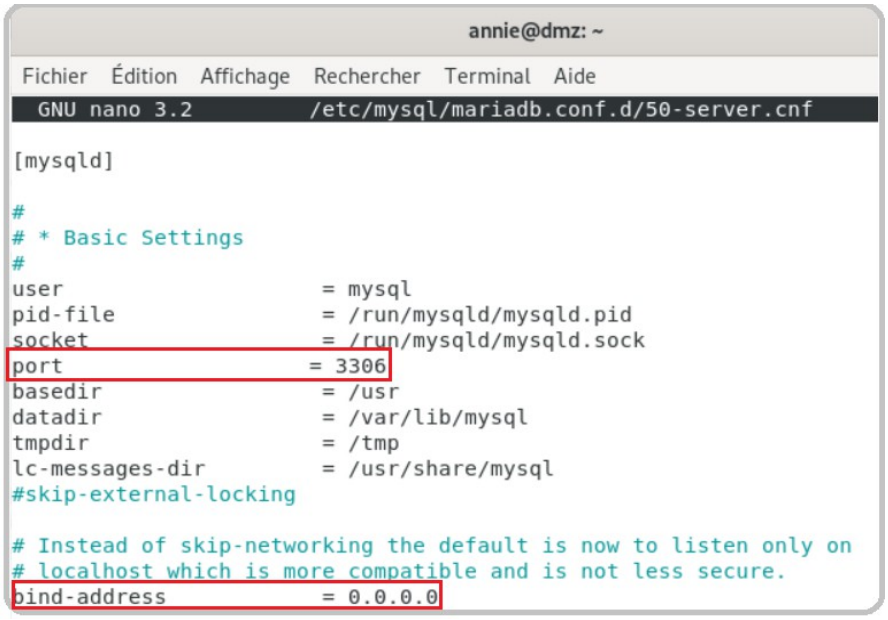


→ ok (nous supprimons par la suite ce fichier pour éviter d'éventuels problèmes)

2.1.7. Installation de MariaDB

Pour pouvoir accéder au serveur Mariadb, il faut que notre poste puisse être capable de communiquer avec. C'est pour cela que nous allons installer un client. Après quelques recherches, nous ne pouvons pas installer un client MYSQL, nous allons donc installer MariaDB et nous en servir seulement pour nous connecter au serveur Mariadb.

- installation MariaDB
- modification du fichier de configuration '50-serv.cnf' : activation du port d'écoute (3306) et bind-address : 0.0.0.0 pour qu'il ne reste pas en localhost.



Le reste est fait lors de l'installation de MariaDB sur le poste serveur Mariadb.

2.2.Routeur

Le routeur va permettre de faire le lien entre les différents réseaux.
Comme indique dans la partie "**Contexte**", nous avons fait le choix d'utiliser un routeur virtuel.
Pour le mettre en place, il faut l'installer sur une machine virtuelle.

Pour les besoins du projet, notre VM possède 3 cartes réseaux. Nous y avons également préinstallé PFsense. Il nous manque plus qu'a le configurer.

Voici l'interface qui apparaît lorsque nous sommes sur la VM PFsense

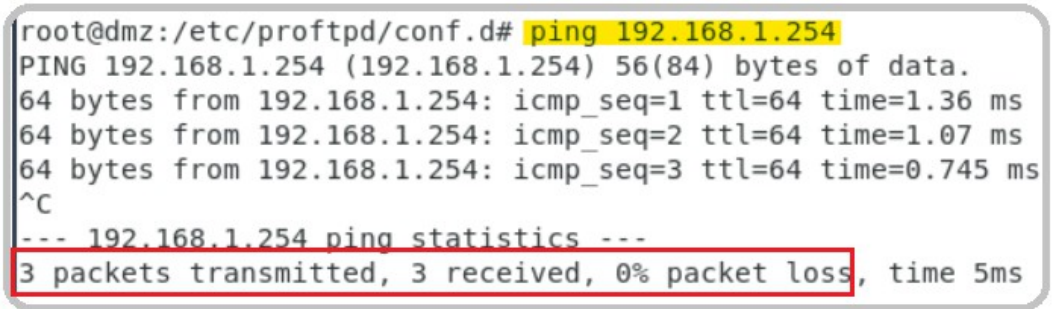
```
0) Logout (SSH only)                  9) pfTop
1) Assign Interfaces                 10) Filter Logs
2) Set interface(s) IP address       11) Restart webConfigurator
3) Reset webConfigurator password    12) PHP shell + pfSense tools
4) Reset to factory defaults         13) Update from console
5) Reboot system                     14) Enable Secure Shell (sshd)
6) Halt system                       15) Restore recent configuration
7) Ping host                         16) Restart PHP-FPM
8) Shell
```

- Il faut assigner les interfaces aux cartes réseaux (option 1 : Assign Interfaces) :
  - **WAN** : hn0
  - **LAN** : hn1
  - **OPT1** : hn2

```
Enter an option: 1


Valid interfaces are:

hn0     00:15:5d:38:01:17   (up) Hyper-V Network Interface
hn1     00:15:5d:38:01:18   (up) Hyper-V Network Interface
hn2     00:15:5d:38:01:1a (down) Hyper-V Network Interface

Do VLANs need to be set up first?
If VLANs will not be used, or only for optional interfaces, it is typical to
say no here and use the webConfigurator to configure VLANs later, if required.

Should VLANs be set up now [y|n]? n

If the names of the interfaces are not known, auto-detection can
be used instead. To use auto-detection, please disconnect all
interfaces before pressing 'a' to begin the process.

Enter the WAN interface name or 'a' for auto-detection
(hn0 hn1 hn2 or a): hn0

Enter the LAN interface name or 'a' for auto-detection
NOTE: this enables full Firewalling/NAT mode.
(hn1 hn2 a or nothing if finished): hn1

Enter the Optional 1 interface name or 'a' for auto-detection
(hn2 a or nothing if finished): hn2
```

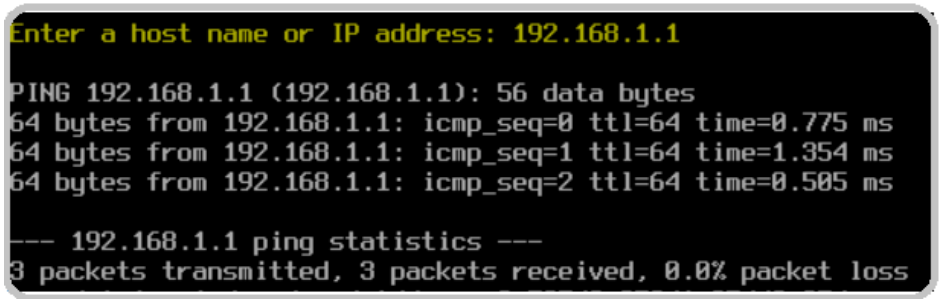
– Attribution d'adresses IP pour chaque interface (option 2 : Set Interface(s) IP address)
  – **WAN** : en DHCP pour aller sur internet
  – **LAN** : ip static : 192.168.100.254
  – **OPT1** : ip static (serveur dmz) : 192.168.1.254

– Test de ping entre poste serveur DMZ et interface OPT1



→ ok

– Ping entre Pfsense (OPT1) et poste serveur DMZ

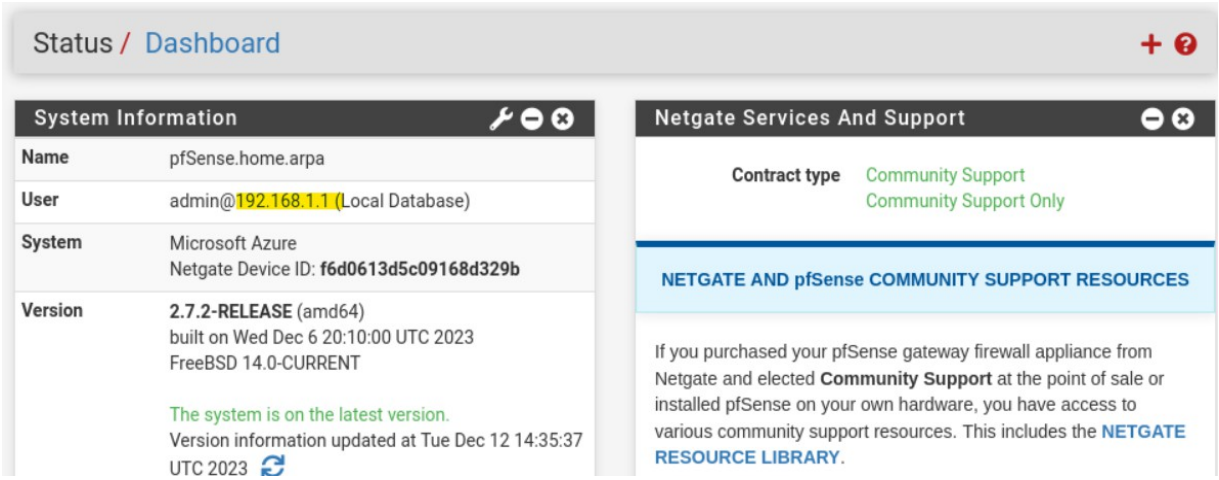


→ ok

Mettre screen ping avec LAN

Depuis poste DMZ : accès pfsense sur le navigateur

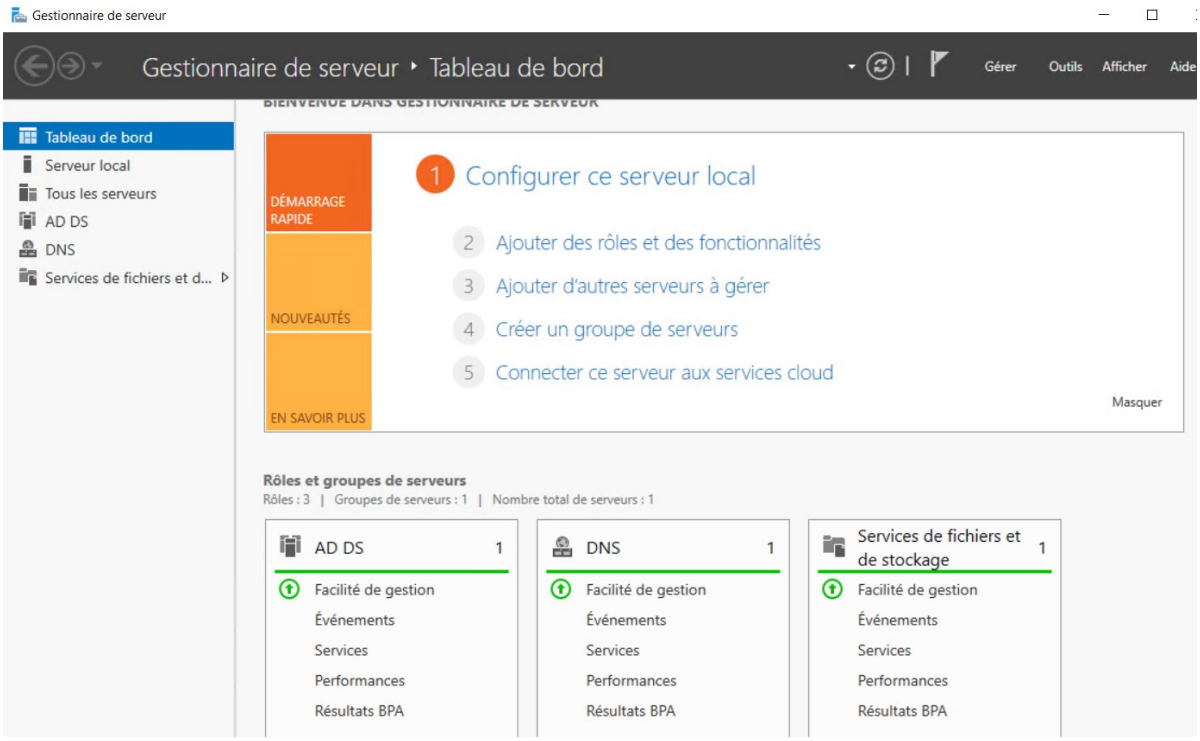
utilisateur : admin
mdp : pfsense

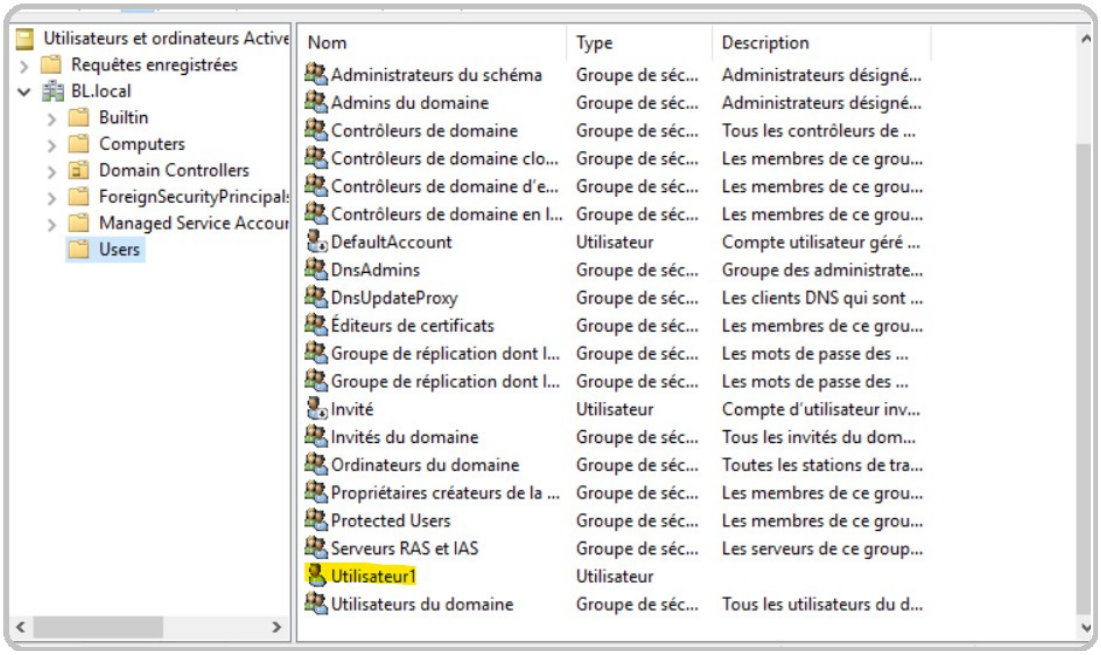


2.3. Serveur de fichiers Windows

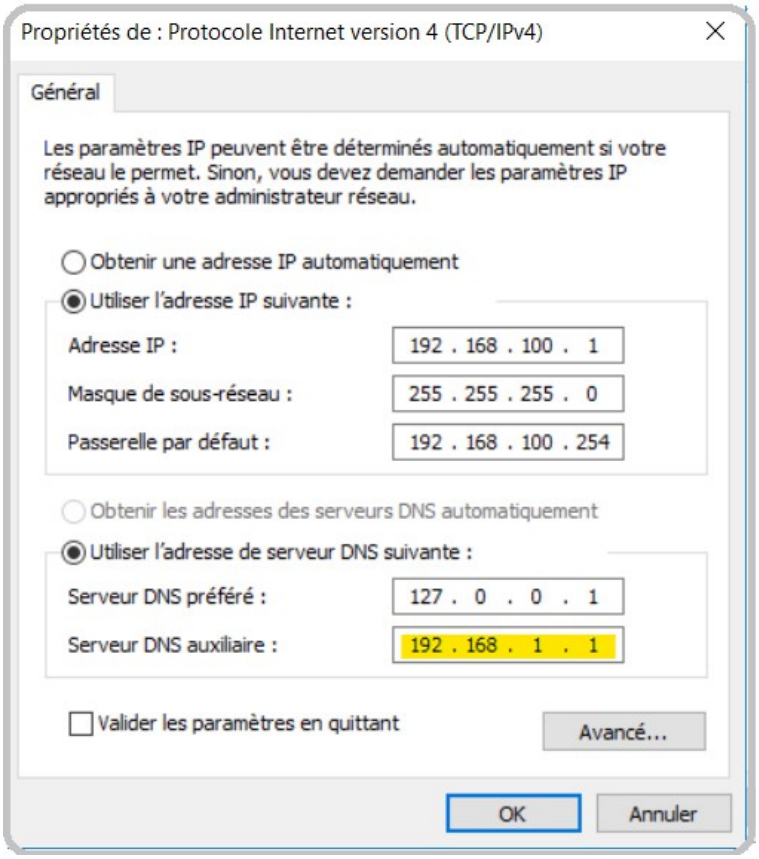
Nous devons mettre en place un serveur de fichier. Dans notre cahier des charges, il est spécifié qu'il serait préférable qu'il soit sous Windows.

Nous avons donc installé sur un poste Windows server 2016. Nous avons configuré Active Directory et créé une session utilisateur.

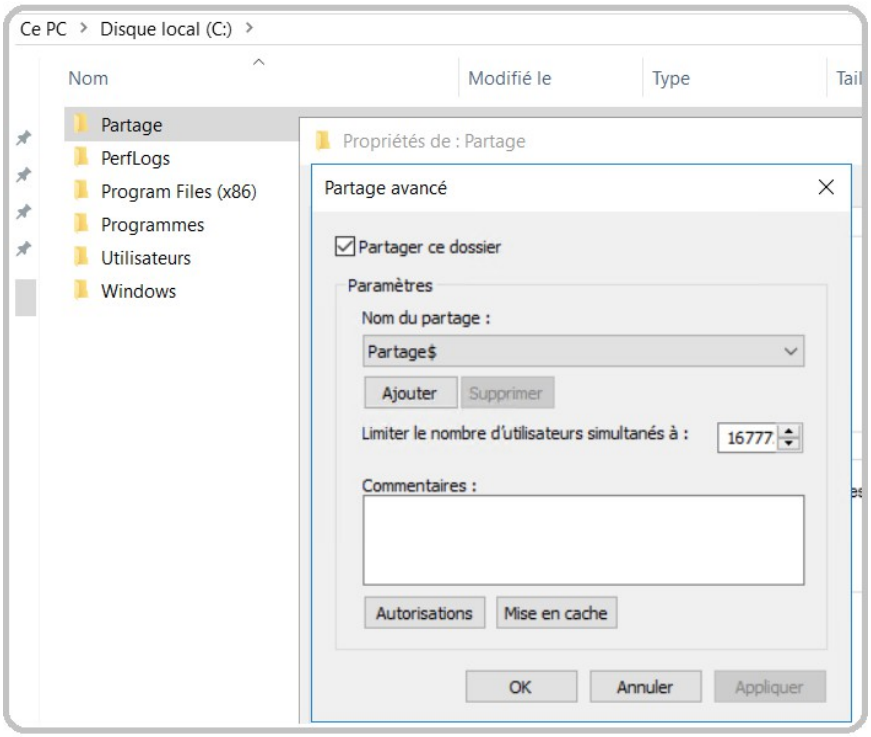




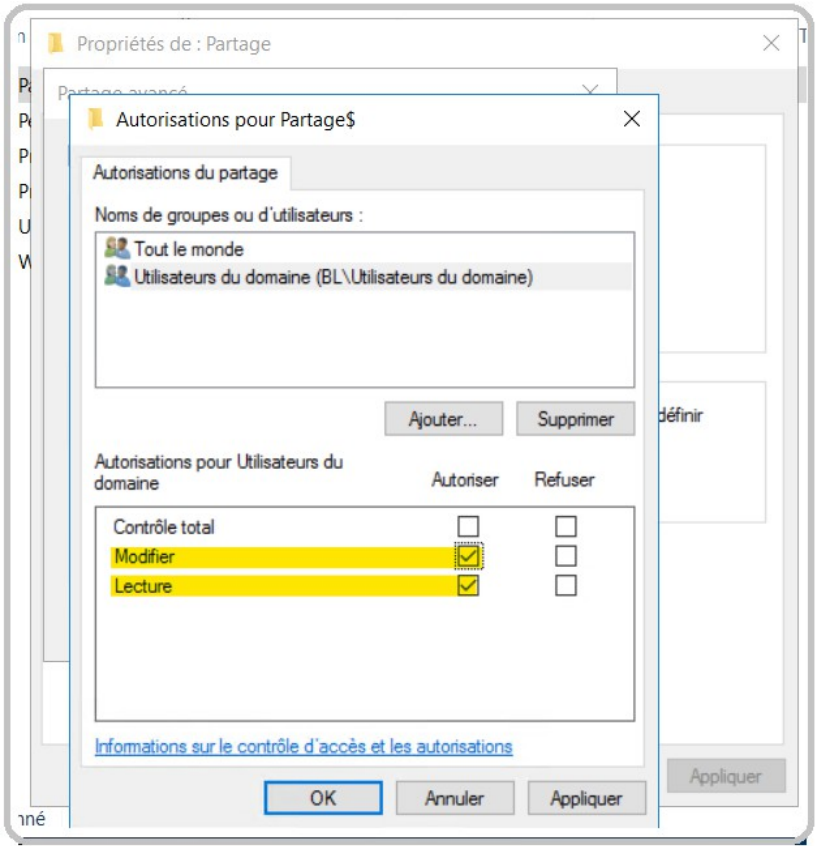
Nous avons rattaché ce serveur au DNS de notre poste serveur DMZ.



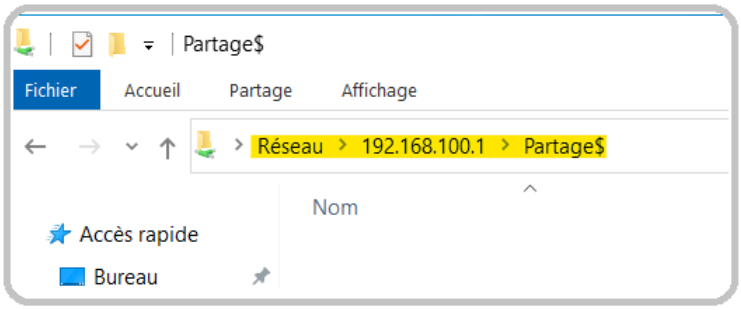
- Création du dossier partagé et partage de ce dernier



- Autorisations pour les utilisateurs



- Vérification : utilisateur d'un autre ordinateur a accès à ce dossier



→ ok

2.4. Serveur Mariadb

Nous sommes à présent sur le poste Mariadb. Nous allons y installer le gestionnaire de bdd, le paramétrer et faire en sorte que le poste serveur dmz puisse se connecter à la bdd.

- Configuration IP

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
        address 192.168.100.2
        netmask 255.255.255.0
        gateway 192.168.100.254
```

- Installation du paquet "mariadb-server"

- Configuration et sécurisation

```
root@dbmaria:~# mysql_secure_installation
```

```
Setting the root password ensures that nobody can log into the MariaDB
root user without the proper authorisation.

Set root password? [Y/n] Y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
 ... Success!


By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them.  This is intended only for testing, and to make the installation
go a bit smoother.  You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] Y
 ... Success!

Normally, root should only be allowed to connect from 'localhost'.  This
ensures that someone cannot guess at the root password from the network.
```

```
Normally, root should only be allowed to connect from 'localhost'.  This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] Y
 ... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access.  This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] Y
 - Dropping test database...
 ... Success!
 - Removing privileges on test database...
 ... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n] Y
 ... Success!
```

– Création d'un compte administrateur et attribution de privilèges

```
MariaDB [(none)]> GRANT ALL PRIVILEGES ON *.*  TO 'administrateur'@'localhost';
Query OK, 0 rows affected (0,000 sec)
```

- Tests de l'installation MariaDB en local

```
root@dbmaria:~# systemctl status mariadb.service
● mariadb.service - MariaDB 10.3.39 database server
   Loaded: loaded (/lib/systemd/system/mariadb.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2023-11-05 15:03:12 CET; 16min ago
     Docs: man:mysqld(8)
           https://mariadb.com/kb/en/library/systemd/
 Main PID: 2702 (mysqld)
   Status: "Taking your SQL requests now..."
    Tasks: 31 (limit: 2324)
   Memory: 75.1M
   CGroup: /system.slice/mariadb.service
           └─2702 /usr/sbin/mysqld

nov. 05 15:03:12 dbmaria systemd[1]: Starting MariaDB 10.3.39 database server...
nov. 05 15:03:12 dbmaria systemd[1]: Started MariaDB 10.3.39 database server.
nov. 05 15:03:12 dbmaria /etc/mysql/debian-start[2737]: Upgrading MySQL tables if necessary.
root@dbmaria:~#
```

→ ok

- Connexion avec session administrateur

```
root@dbmaria:~# mysql -u administrateur -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 41
Server version: 10.3.39-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

→ MariaDB est opérationnel et utilisateur est capable de s'authentifier avec succès

Par défaut, nous ne pouvons nous connecter sur MariaDB qu'en local.
Notre but est de se connecter à un serveur distant (depuis poste serveur dmz dans notre cas).

- Dans/etc/mysql/mariadb.conf.d/50-server.cnf, nous allons modifier la valeur de "bind-address" qui est par défaut en localhost. Nous attribuons la valeur 0.0.0.0 pour que l'accès soit possible de tout poste qui a les identifiants pour s'identifier.

```
 GNU nano 3.2          /etc/mysql/mariadb.conf.d/50-server.cnf

[mysqld]

#
# * Basic Settings
#
user                    = mysql
pid-file                = /run/mysqld/mysqld.pid
socket                  = /run/mysqld/mysqld.sock
port                    = 3306
basedir                 = /usr
datadir                 = /var/lib/mysql
tmpdir                  = /tmp
lc-messages-dir         = /usr/share/mysql
#skip-external-locking

# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address            = 0.0.0.0
```

– Création d'un utilisateur et attribution de droits pour qu'il puisse se connecter depuis le poste dmz sur la bdd "sitebl"

```
MariaDB [(none)]> GRANT ALL PRIVILEGES ON sitebl.* TO 'admin'@'192.168.1.1' IDENTIFIED BY 'motdepasse'
WITH GRANT OPTION;
Query OK, 0 rows affected (0,000 sec)
```

Pour que le serveur dmz puisse se connecter sur la bdd du poste Mariadb, nous devons entrer une règle dans le pare-feu pour accepter la communication sur le port d'écoute 3306.

☐	✔	0/0 B	IPv4 TCP	192.168.100.2	*	192.168.1.1	3306	*	none	port mysql, communication

– Test de connexion du poste dmz (192.168.1.1) vers la bdd hébergée sur le poste Mariadb

```
root@dmz:~# mysql -u admin -p -h 192.168.100.2 -P 3306
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 36
Server version: 10.3.39-MariaDB-0+deb10u1 Debian 10

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

2.6. VPN

Tout d'abord, VPN signifie « Virtual Private Network ». C'est un réseau privé virtuel.

Il va créer un lien virtuel, ici entre l'entreprise et l'ordinateur d'un employé chez lui. Dans ce lien, les données sont isolées et sécurisées de bout en bout. Un employé va donc pouvoir de chez lui accéder à des fichiers importants de l'entreprise en se connectant à distance avec un VPN.

Un VPN masque également l'adresse IP de l'ordinateur de l'utilisateur et utilise une adresse du serveur VPN qui devient donc lui-même l'élément visible sur le net. L'utilisateur est donc moins facilement identifiable avec son adresse IP.

Dans le cadre de ce projet, nous avons décidé d'utiliser OpenVPN. C'est un VPN open source mûr (crée en 2001) et ayant une bonne réputation. Il prend en charge la technologie Open SSL et des clés de cryptages 256bits.

Il est compatible avec beaucoup d'environnements (Linux, Mac, Windows...). Il est plus simple d'utilisation que le VPN Ipsec et est compatible avec davantage de services VPN que WireGuard même s'il est un peu plus lent que ce dernier. Un autre avantage est que le port d'écoute est personnalisable.

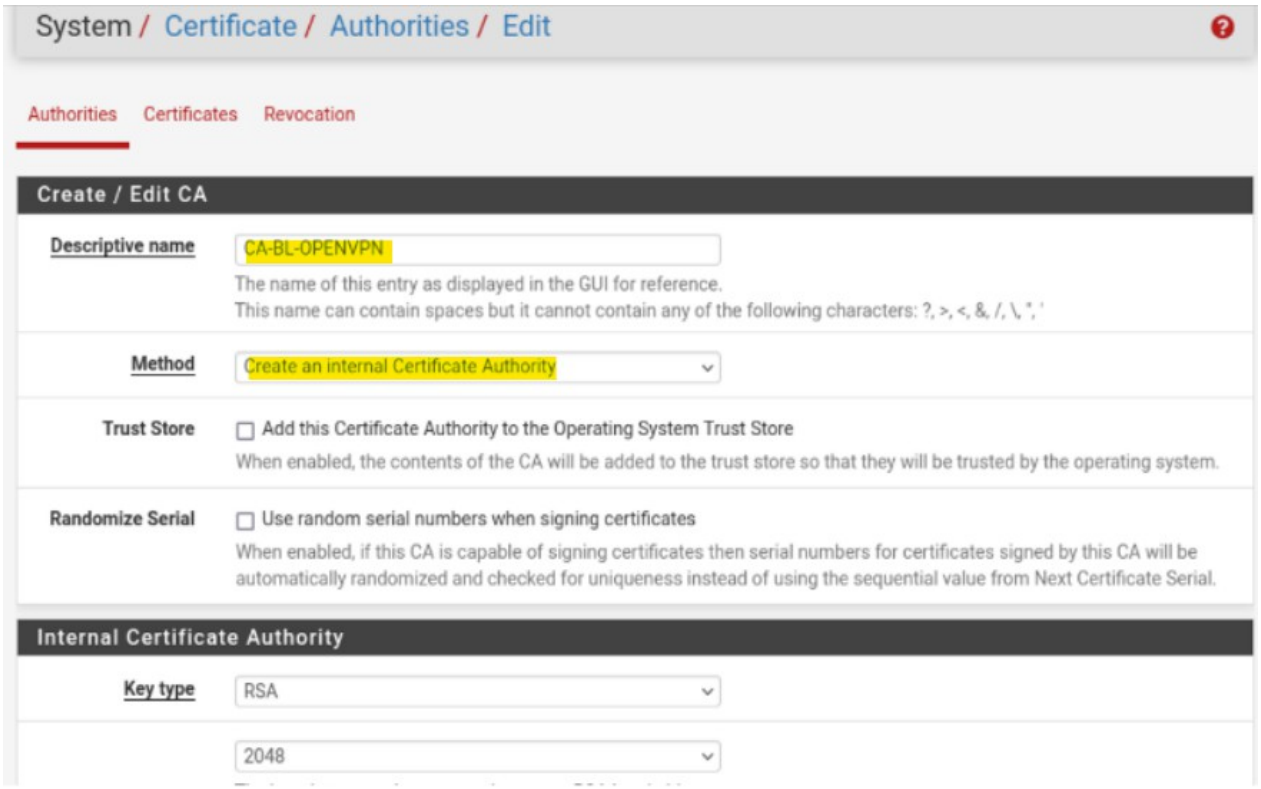
Nous avons également choisi ce VPN car il est facilement paramétrable avec le Pfsense.

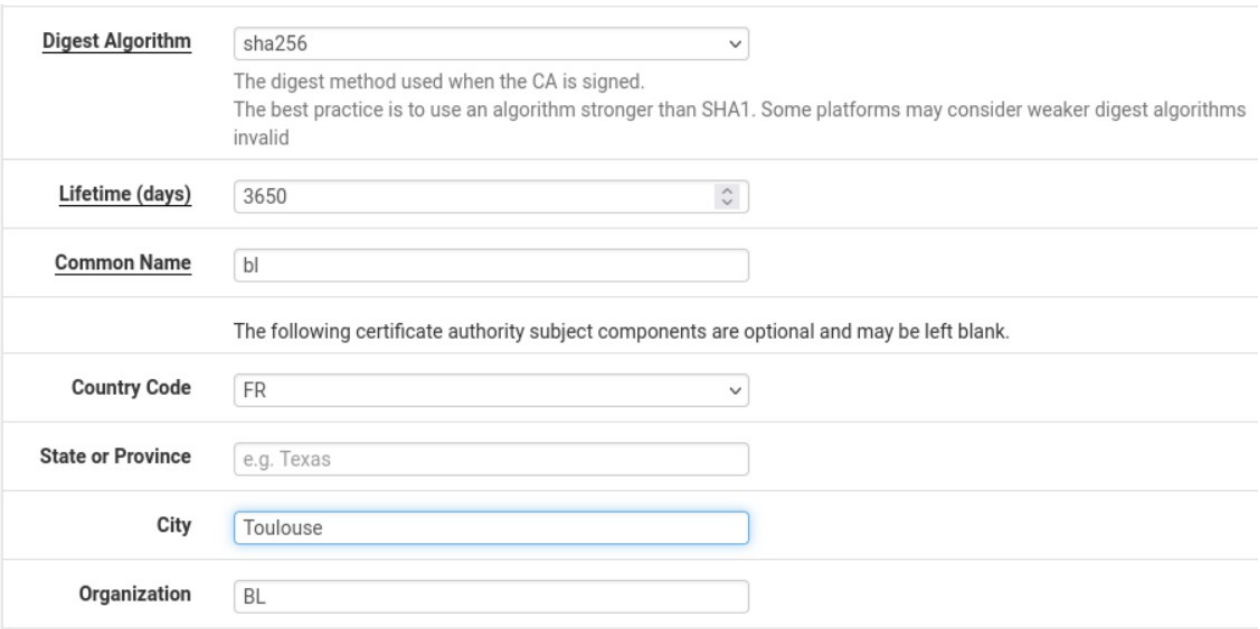
Dans notre cas, le PC client va établir la connexion à l'aide d'un client OpenVPN auprès du pare-feu Pfsense sur lequel est activé et configuré OpenVPN.

2.5.1. Création de l'autorité certificative et d'un certificat Server

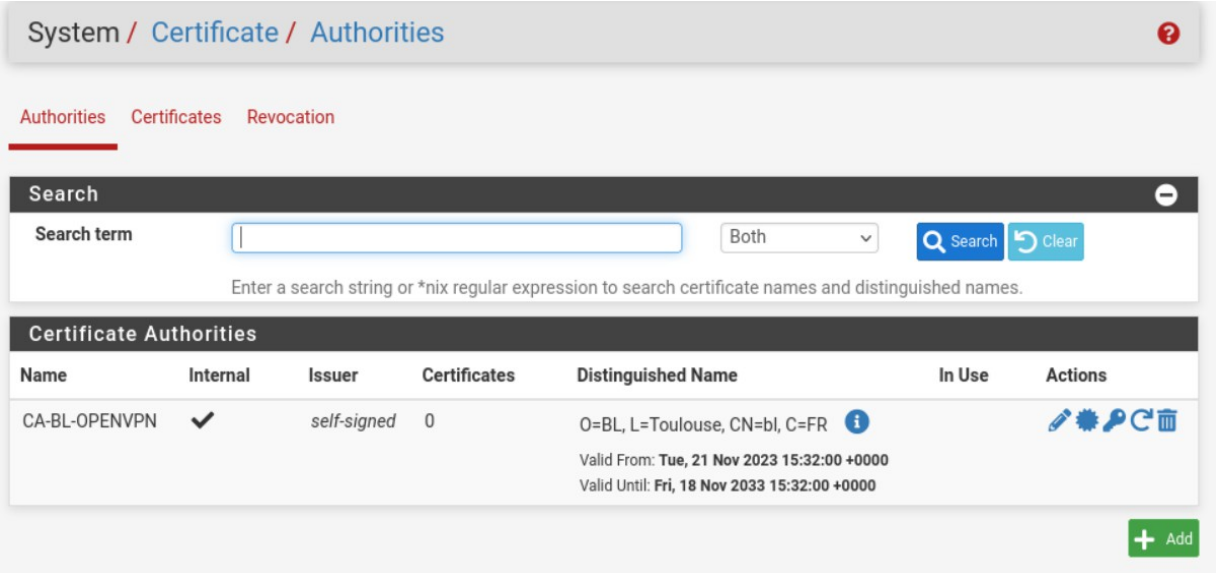
Nous allons, sur PFsense, créer une autorité de certification interne. Puis, nous mettrons en place un certificat "Server".

– Création de l'autorité de certification interne dans System/Certificate / Autorities. Nous l'appelons "CA-BL-OPENVPN"

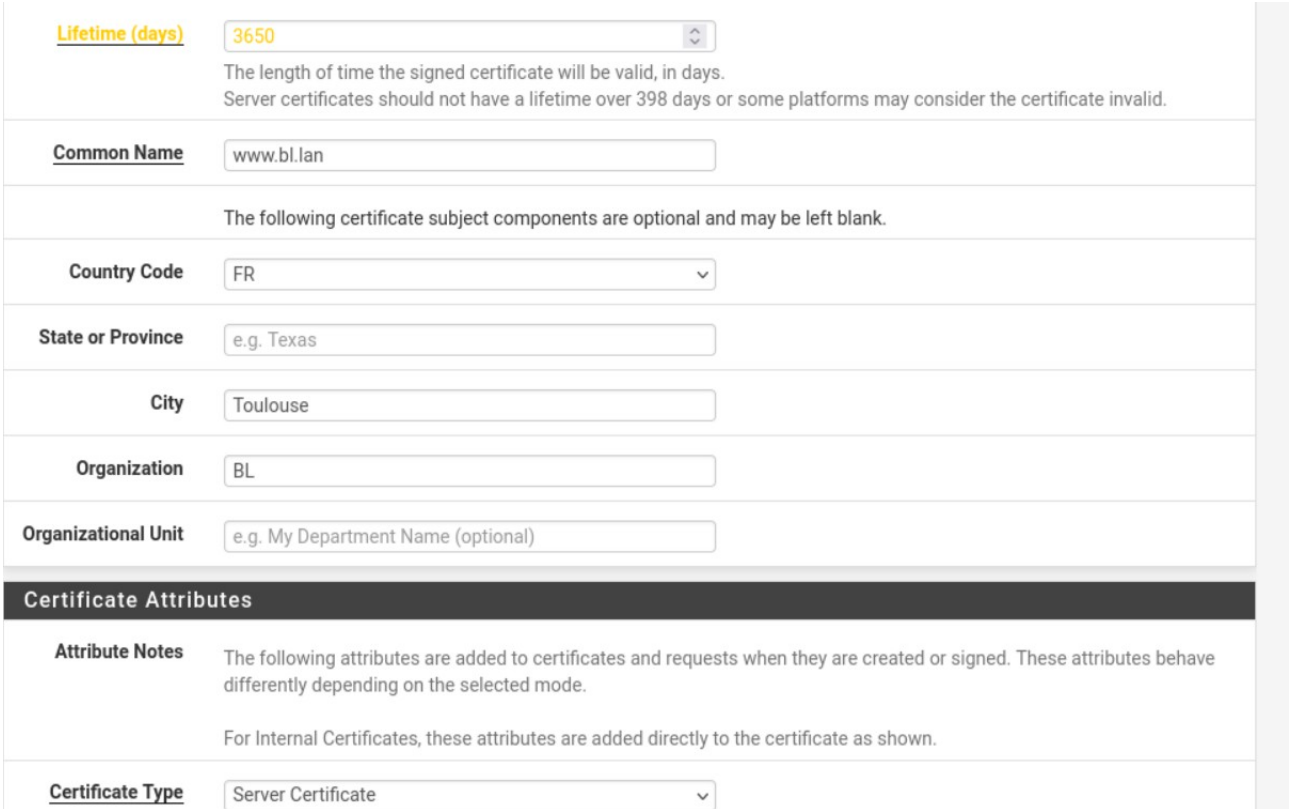


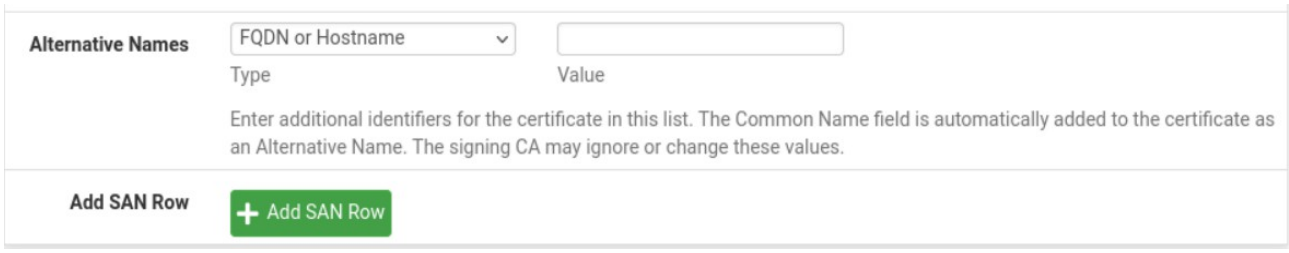


- Vérification de l'existence de l'autorité de certification



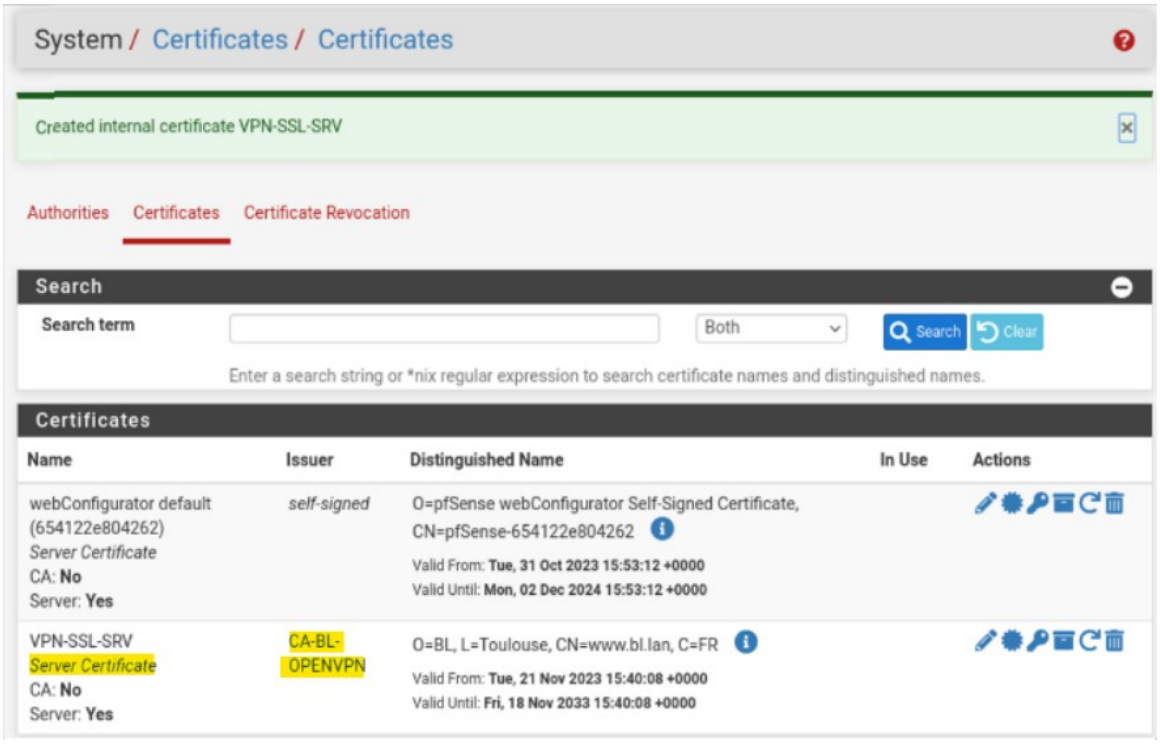
- Création du certificat "Server" en s'appuyant sur notre autorité certificative dans System / Certificate / Certificates





manque un screen création certificat

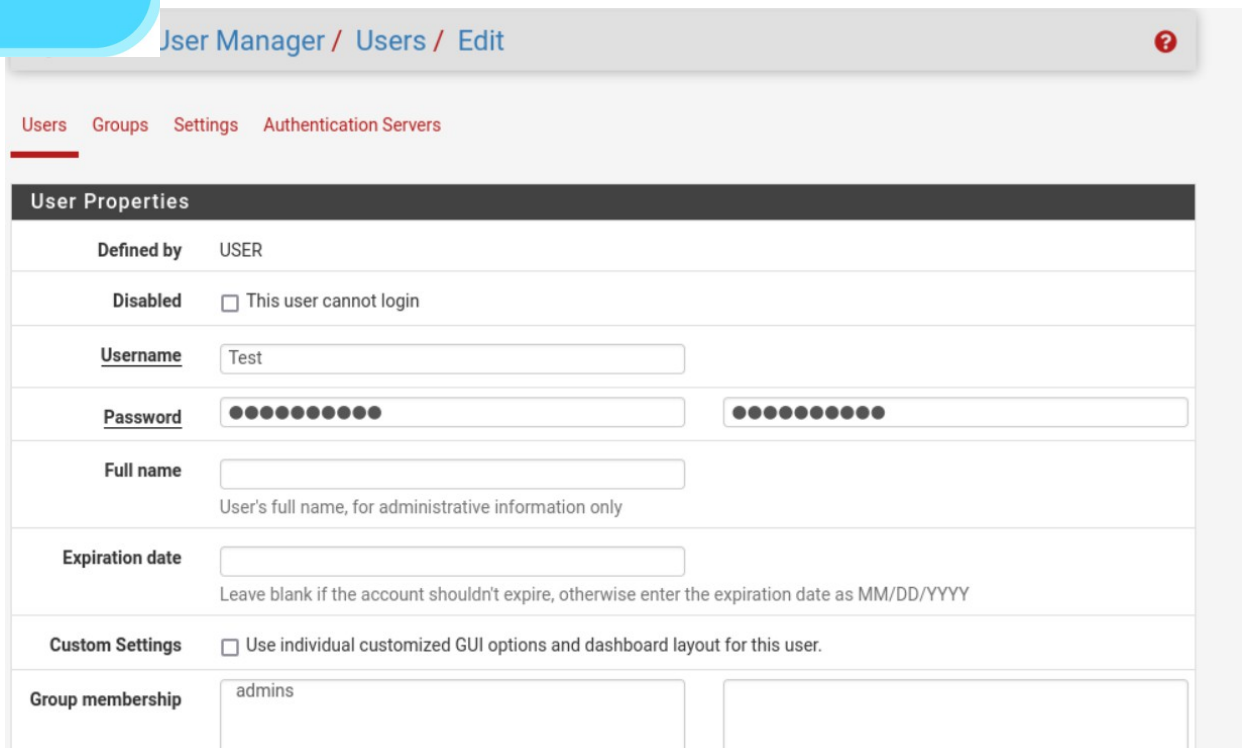
- Vérification existence du certificat "Server"



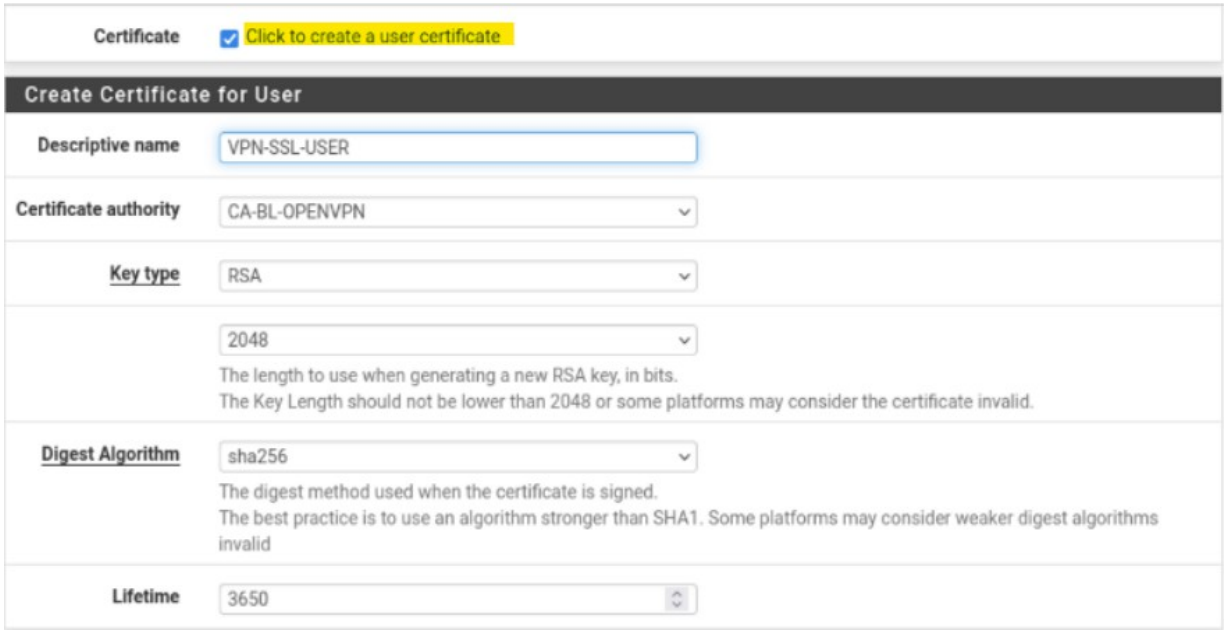
2.5.2. Création d'un utilisateur et d'un certificat utilisateur

Nous allons maintenant créer un utilisateur et générer en même temps un certificat "User" pour qu'il ait l'autorité de se connecter au serveur VPN.

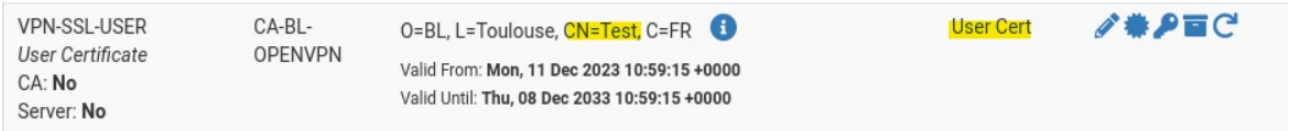
- Création de l'utilisateur dans System / User Managers / Users



– Au même endroit, nous cochons sur "Click to create a user certificate". Ce certificat va également s'appuyer sur notre autorité certificative.

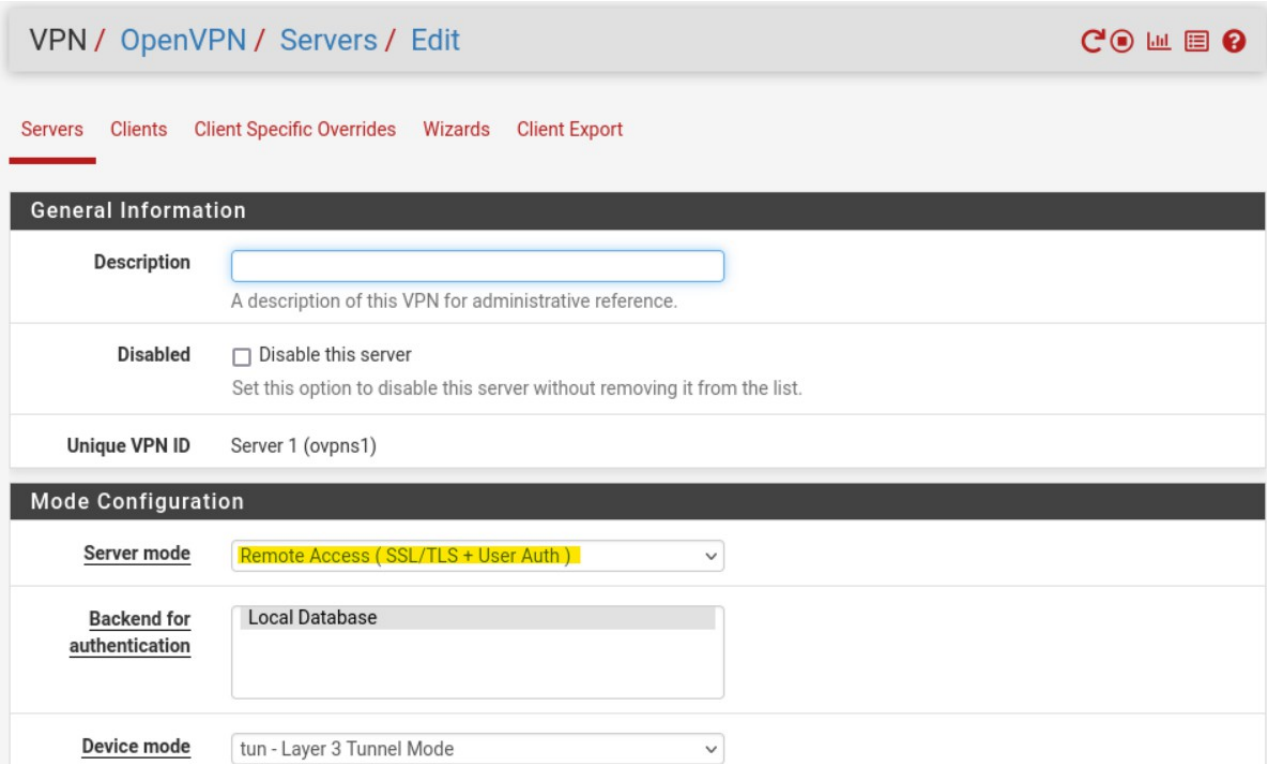


- Vérification : existence de l'utilisateur + son certificat

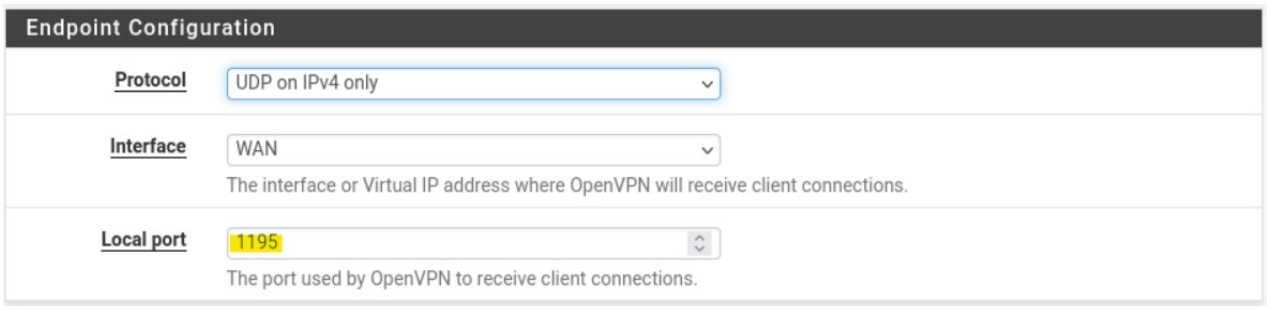


2.5.3. Configuration du serveur VPN

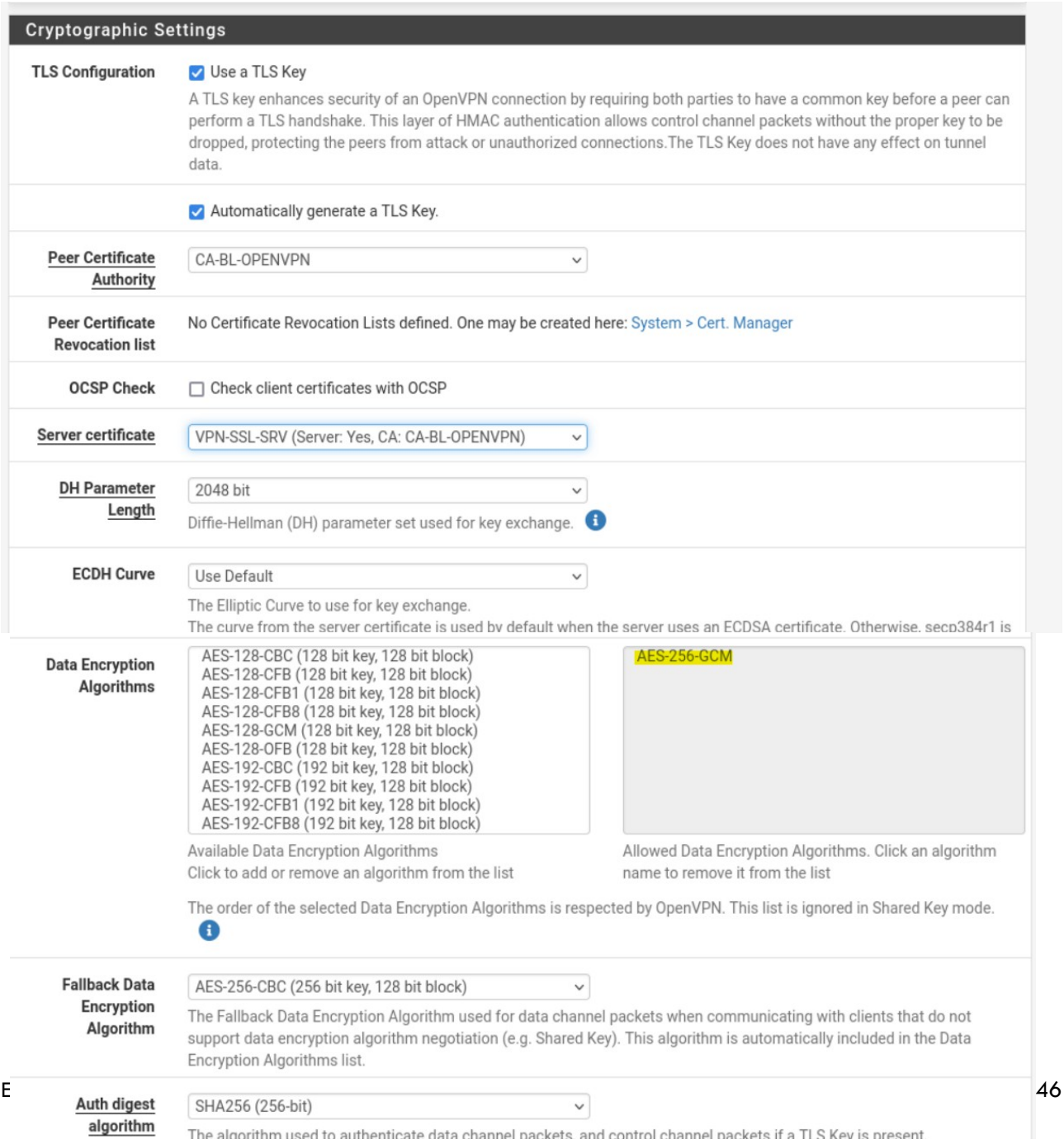
- Paramétrage dans VPN / OpenVPN / Servers .

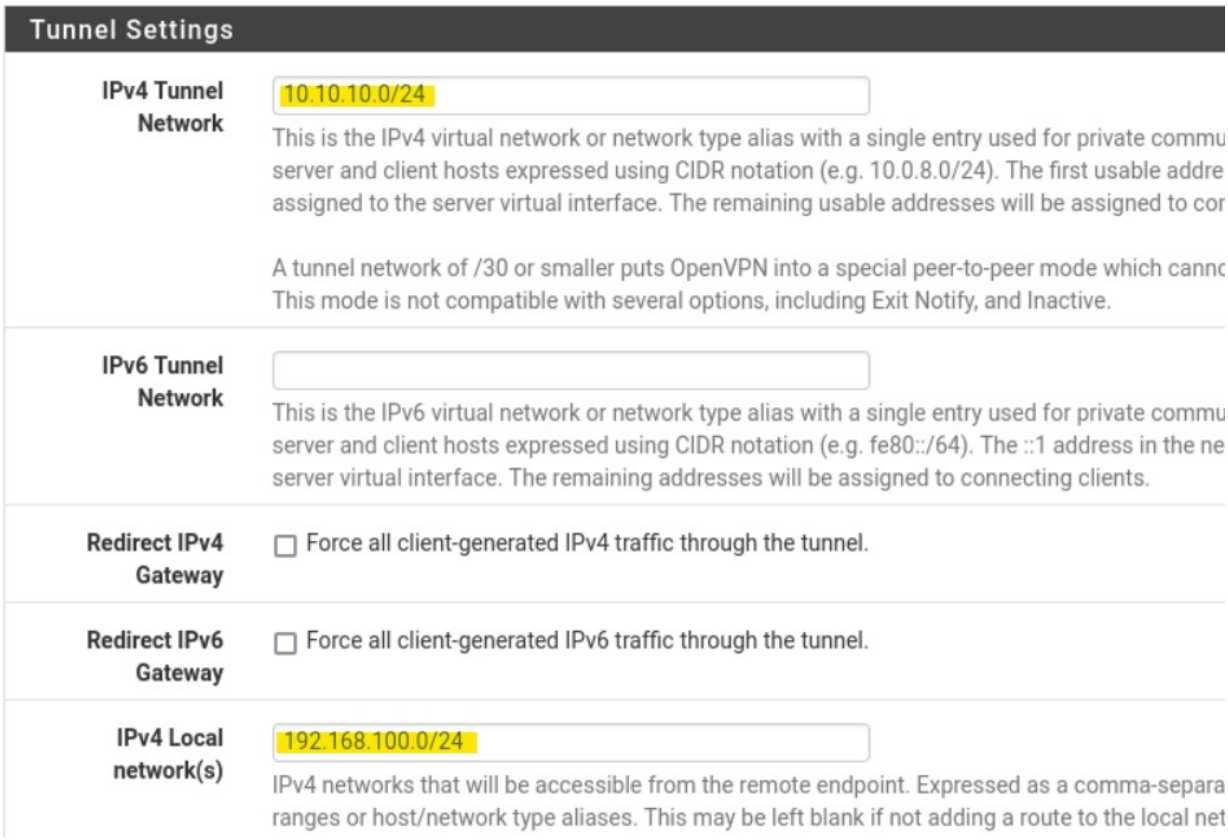


Dans notre cas, le "Server mode" et en Remot Access car nous
Jtiliser l'authentification utilisateur et utiliser SSL/TLS.



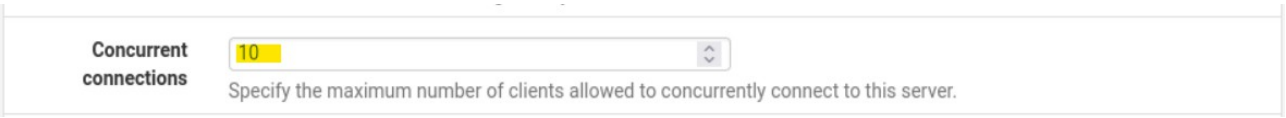
Nous changeons le port par défaut pour que cela soit plus sécurisé.



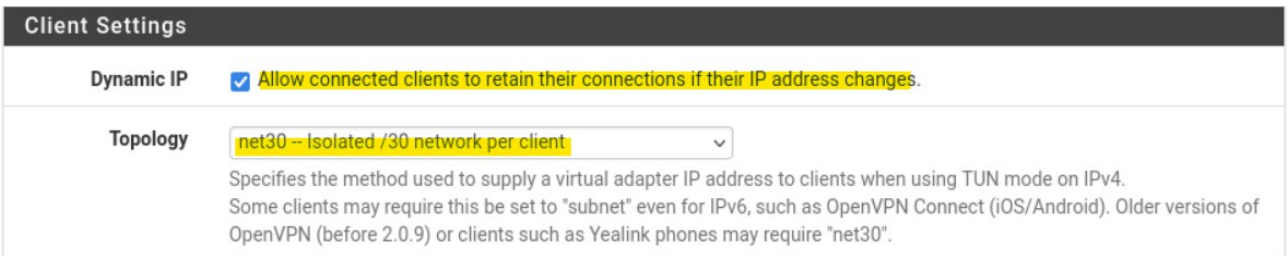


Le réseau sur lequel l'utilisateur sera connecté lorsqu'il se connectera au VPN est le 10.10.10.0/24.

Nous rendons ensuite accessible à travers ce tunnel le réseau LAN (192.168.100.0/24).

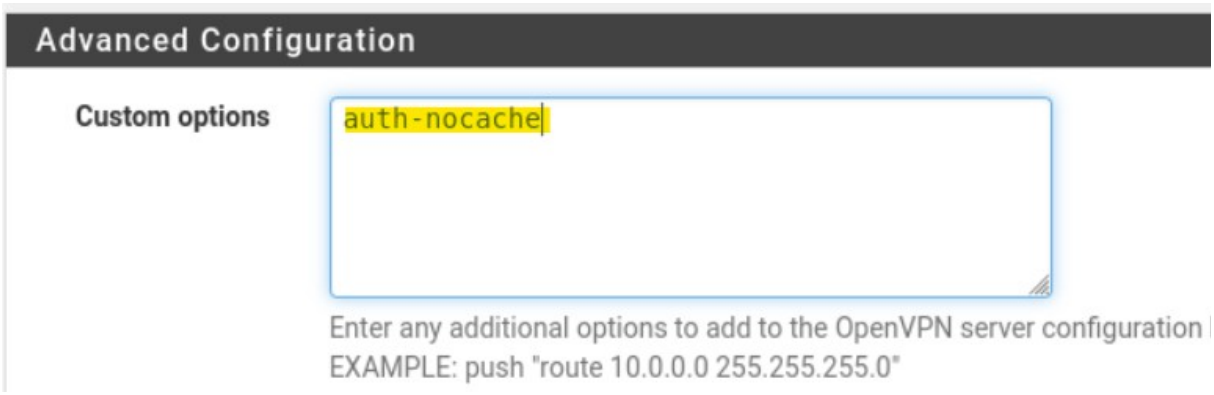


Pour les phases de tests, nous laissons le nombre de connexions simultanées à 10. Nous adapterons par la suite selon les besoins du client.



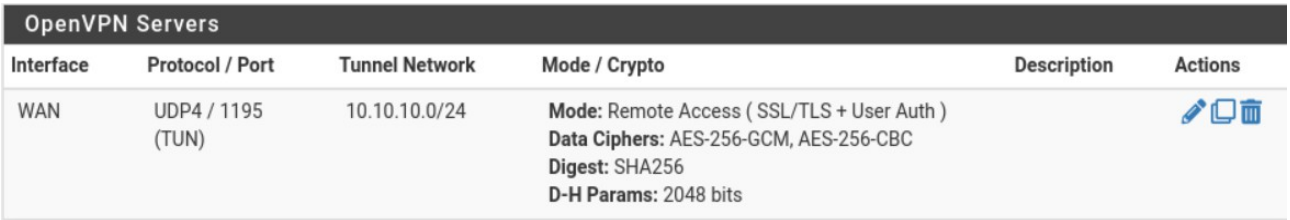
Nous activons l'option "Dynamic IP" pour que l'utilisateur, si son adresse IP change lorsqu'il est connecté en VPN, ne perde pas sa connexion au serveur.

En ce qui concerne la topologie, nous choisissons "net30 – Isolated/30 network per client " pour que chaque utilisateur soit isolé dans un sous-réseau. Cela a pour conséquence que les utilisateurs ne peuvent pas communiquer entre eux. C'est une action qui est plus sécurisée mais qui prend 4 adresses IP pour chaque utilisateur (une pour l'ordinateur, une pour le pare-feu, l'adresse de réseau et celle de broadcast).

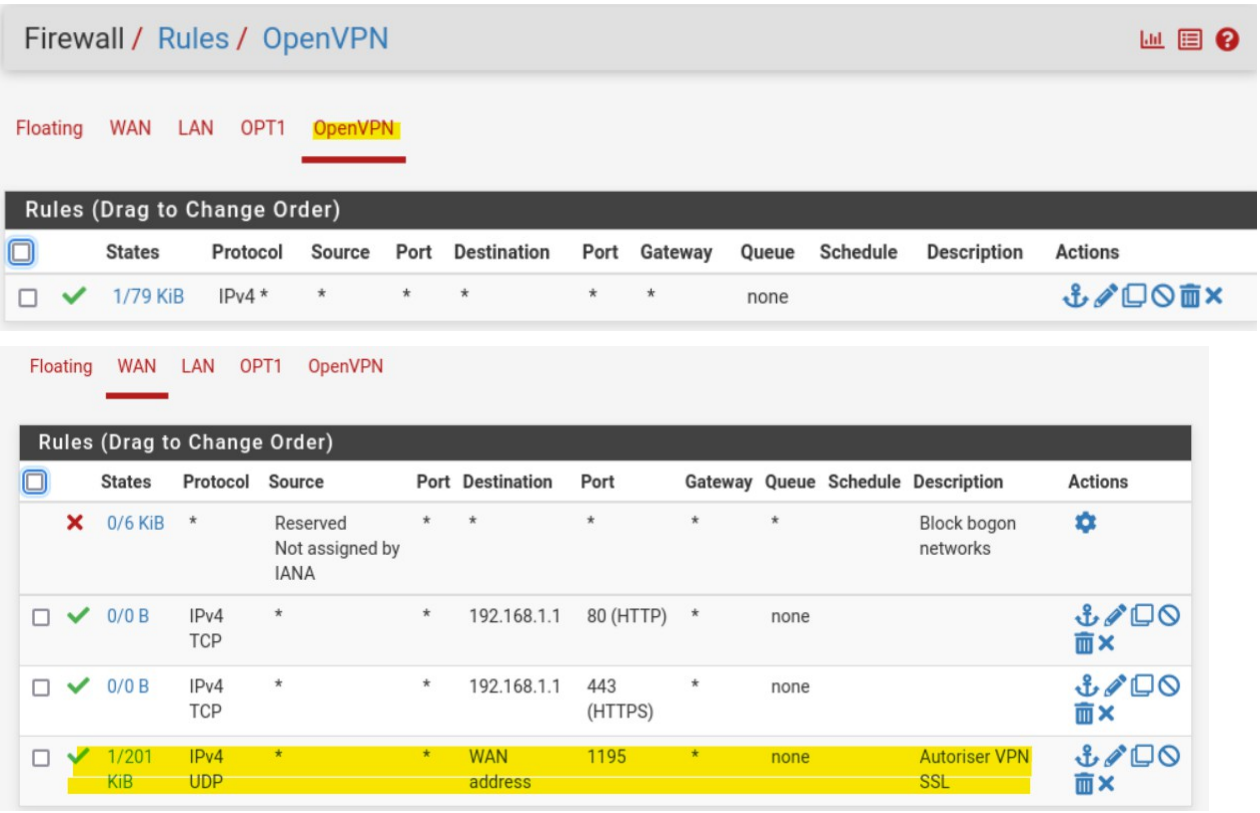


En entrant ce texte, les identifiants ne sont pas mis en cache, ce qui peut être un peu plus sécurisé contre le vol d'identifiants.

– Vérification de l'existence du serveur créé



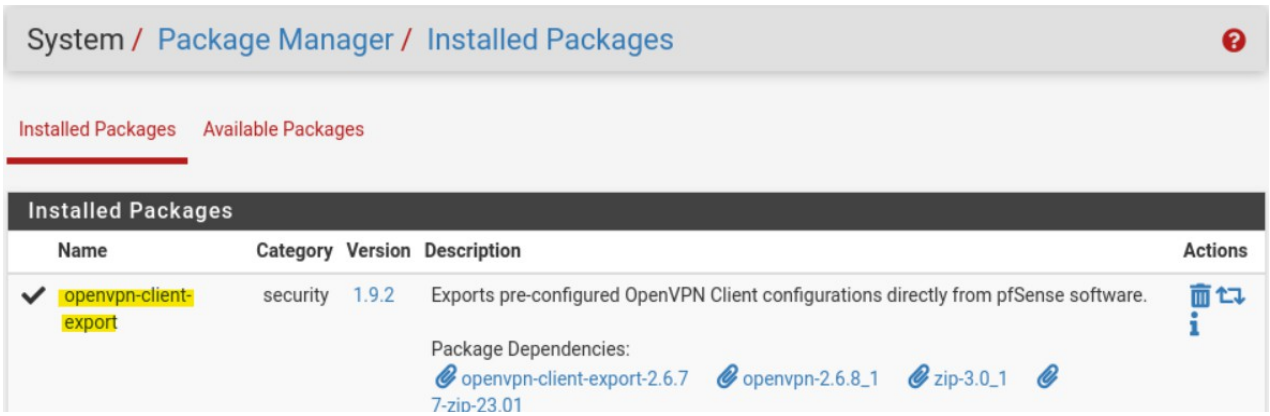
- Entrer les règles sur le pare-feu pour autoriser le flux VPN et autoriser l'accès aux ressources, ici le serveur de fichiers



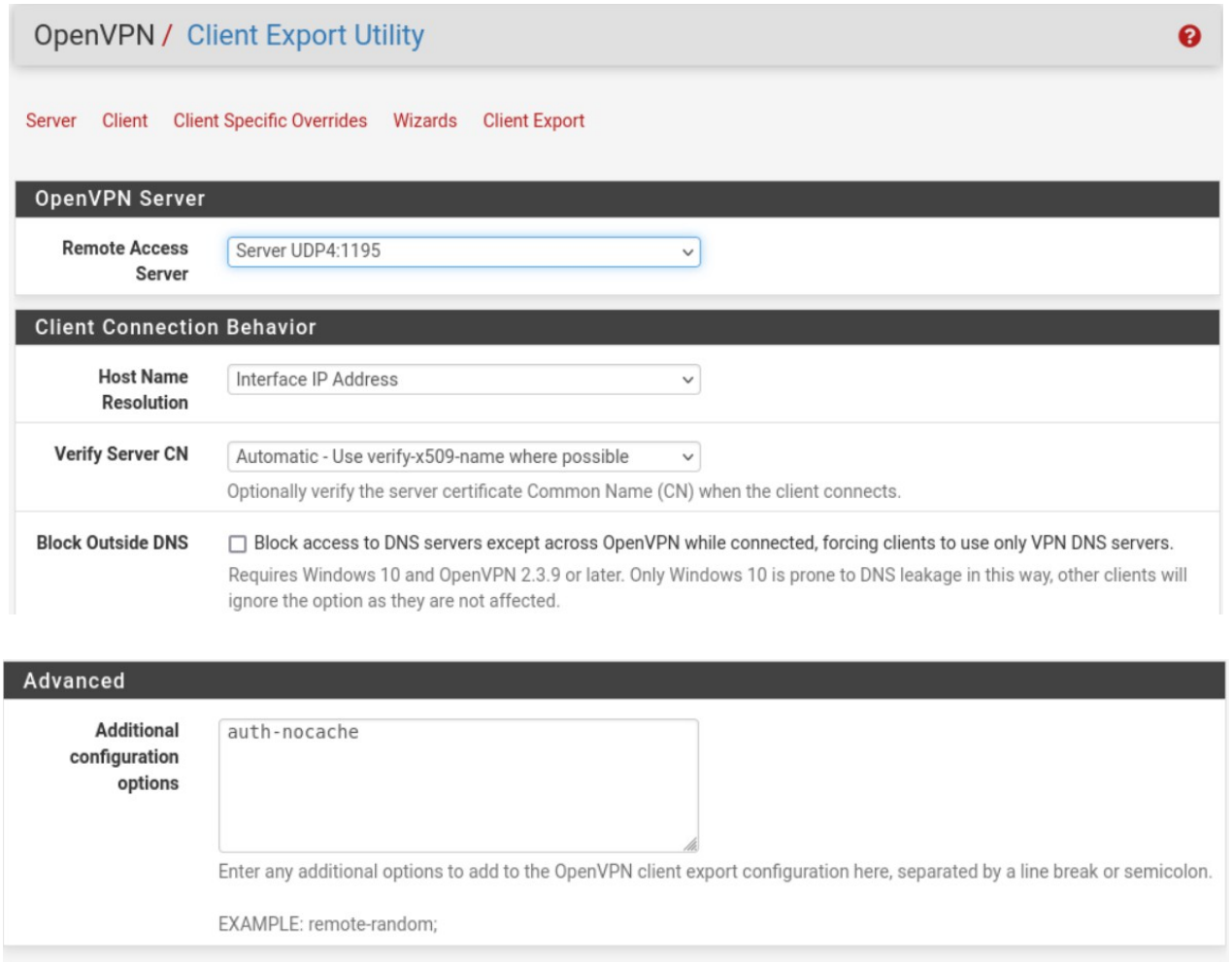
2.5.4. Export de la configuration VPN

Nous allons à présent télécharger la configuration sur les postes utilisateurs concernés. Les ordinateurs sont sous Windows 10 professionnel.

- Il est tout d'abord nécessaire de télécharge et installer un paquet permettant aux utilisateurs d'exporter la configuration VPN sur leur poste.



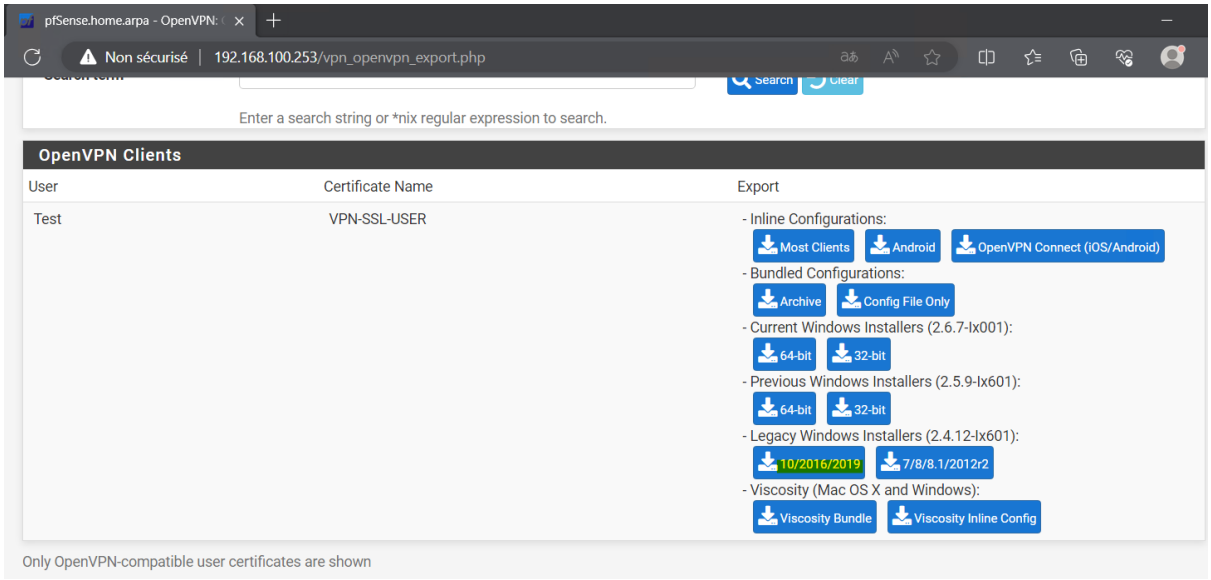
– Dans VPN / OpenVPN / Client Export Utility, nous entrons les données pour l'export.



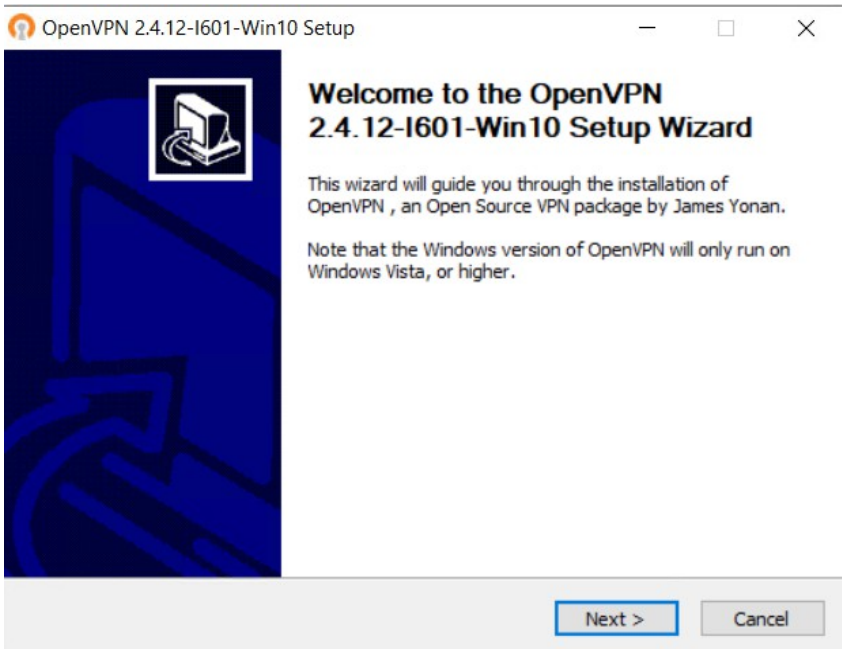
2.5.5. Utilisation du client VPN sur un poste utilisateur

Pour avoir le client, nous devons le télécharger depuis Pfsense. Pour cela, dans un premier temps, le poste est sur le réseau LAN. Une fois le client téléchargé, nous ferons en sorte que l'ordinateur ne soit plus sur un réseau de l'entreprise.

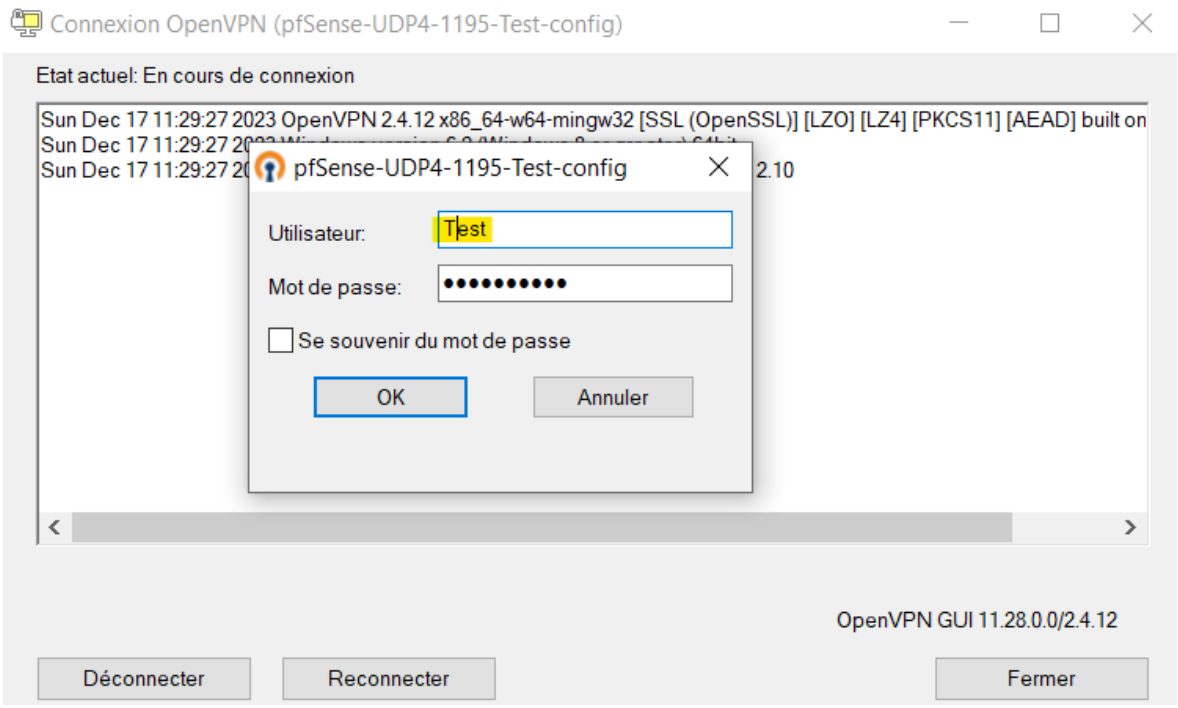
- Téléchargement du client OpenVPN



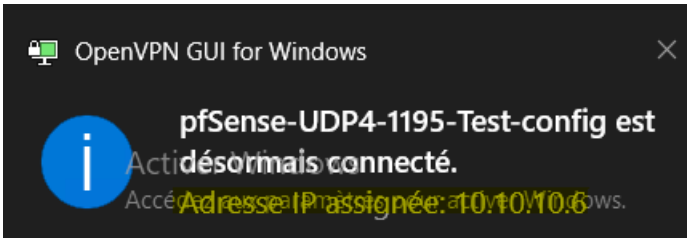
- Installation d'OpenVPN et de la configuration



- Connexion avec l'utilisateur créé



- L'ordinateur s'est bien connecté au serveur VPN et une adresse du réseau VPN a été attribuée

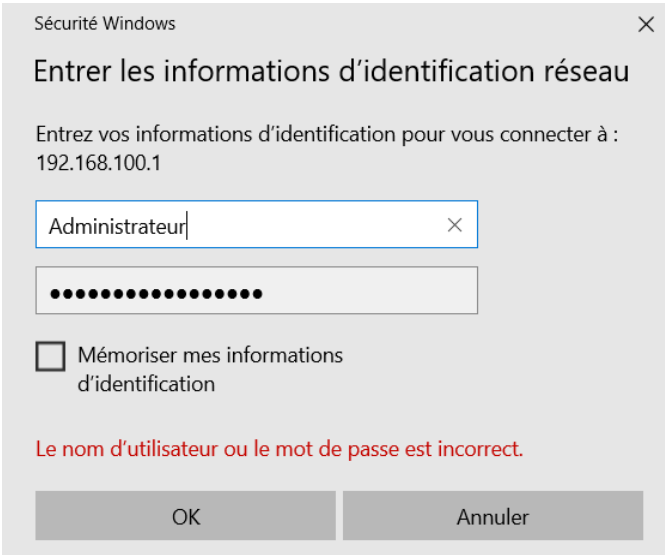


- Vérification : adresse IP et ping vers le poste serveur de fichiers

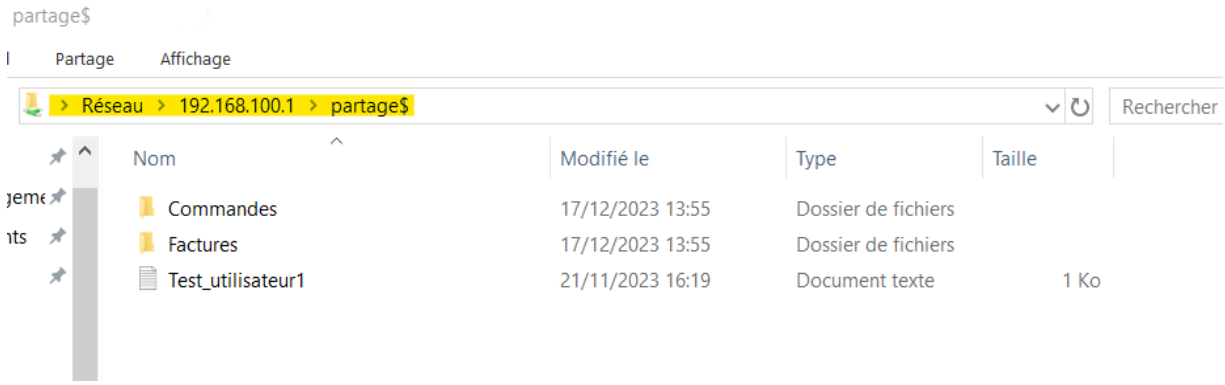
```
   Adresse IPv4. . . . . . . . . . . . . .: 10.10.10.6
   Masque de sous-réseau. . . . . . . . . : 255.255.255.252
   Passerelle par défaut. . . . . . . . . :

Carte Ethernet Ethernet :

   Suffixe DNS propre à la connexion. . . :
   Adresse IPv6. . . . . . . . . . . . . .: 2a04:cec0:122a:7f0d:638c:976a:17fb:739d
   Adresse IPv6 temporaire . . . . . . . .: 2a04:cec0:122a:7f0d:5042:524f:69bc:78bc
   Adresse IPv6 de liaison locale. . . . .: fe80::e7ff:be77:766d:eb88%6
   Adresse IPv4. . . . . . . . . . . . . .: 192.168.43.101
   Masque de sous-réseau. . . . . . . . . : 255.255.255.0
   Passerelle par défaut. . . . . . . . . : fe80::b2e1:7eff:fe51:151d%6
                                       192.168.43.1

C:\Users\Felix>ping 192.168.100.1

Envoi d'une requête 'Ping'  192.168.100.1 avec 32 octets de données :
Réponse de 192.168.100.1 : octets=32 temps=1 ms TTL=127
Réponse de 192.168.100.1 : octets=32 temps=1 ms TTL=127
Réponse de 192.168.100.1 : octets=32 temps=2 ms TTL=127
Réponse de 192.168.100.1 : octets=32 temps=1 ms TTL=127

Statistiques Ping pour 192.168.100.1:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 1ms, Maximum = 2ms, Moyenne = 1ms
```

→ ok

- Accès au serveur de fichier : dans l'explorateur de fichiers, entrer 192.168.100.1



- Vérification accès aux dossiers :



Ensuite, il aurait fallu mettre en place un 2ème tunnel VPN pour que l'employé, à son domicile, ait accès au serveur web. Pour cela, nous aurions créé un autre certificat serveur, un autre utilisateur avec son certificat.

2.6. Règles ACL

Nous allons à présent paramétrer le pare-feu sur le routeur virtuel.

Pour rappel, il faut :

- DMZ (OPT1) : accès au serveur bdd (LAN) et accès internet
- WAN : accès seulement au serveur web
- LAN : accéder au serveur web et à internet en général + seul poste administrateur ait accès au FTP

DMZ (OPT1)

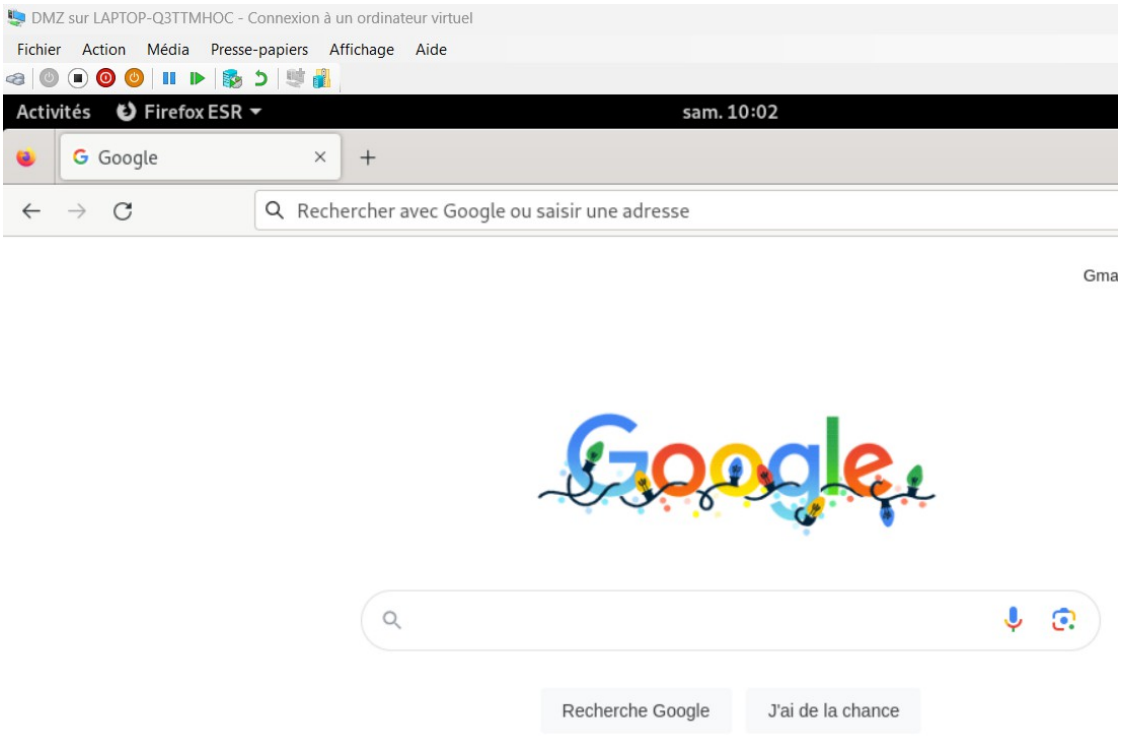
LAN OPT1 OpenVPN

o Change Order)

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
☐	✓ 0/9 KiB	IPv4 ICMP any	*	*	*	*	*	none			
☐	✓ 0/1.04 MiB	IPv4 TCP/UDP	*	*	*	53 (DNS)	*	none		accepter dns	
☐	✓ 0/68 KiB	IPv4 TCP	*	*	*	80 (HTTP)	*	none		accepter accès srv web 80	
☐	✓ 0/1.14 MiB	IPv4 TCP	*	*	*	443 (HTTPS)	*	none		accepter accès srv web 443	

Vérification

- Accès internet :



- Accès serveur bdd : cf connexion au serveur bdd page 27

LAN OPT1 OpenVPN

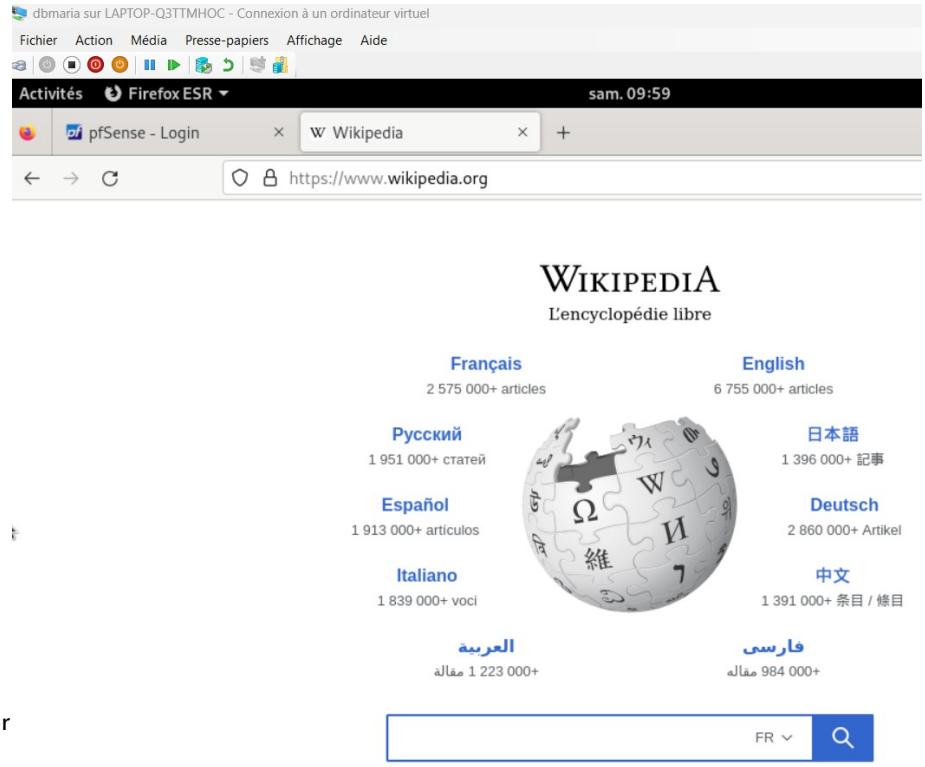
Rules (Drag to Change Order)

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✓	0/0 B	*	*	*	LAN Address	80	*	*		Anti-Lockout Rule	⚙
☐ ✓	0/21 KiB	IPv4 TCP/UDP	*	*	*	53 (DNS)	*	none		Accepter dns	
☐ ✓	0/0 B	IPv4 TCP	*	*	*	443 (HTTPS)	*	none		port 443 https	
☐ ✓	0/0 B	IPv4 ICMP any	*	*	*	*	*	none			
☐ ✓	0/0 B	IPv6 *	LAN net	*	*	*	*	none		Default allow LAN IPv6 to any rule	
☐ ✓	0/0 B	IPv4 TCP	192.168.100.2	*	192.168.1.1	3306	*	none		Port mysql, communication	
☐ ✓	0/0 B	IPv4 TCP	192.168.100.3	*	192.168.1.1	21 (FTP)	*	none		accès ftp port 20 pour 192.168.100.3	
☐ ✓	0/0 B	IPv4 TCP	192.168.100.3	*	192.168.1.1	20	*	none		accès ftp port 21 pour 192.168.100.3	

Add Add Delete Toggle Copy Save Separator

Vérification

- Accès internet :



- Accès ftp seulement depuis 192.168.100.3

```
Carte Ethernet Ethernet :

   Suffixe DNS propre à la connexion. . . :
   Adresse IPv6. . . . . . . . . . . . . .: 2a04:cec0:1222:9979:8858:5ce3:6b38:ba6d
   Adresse IPv6 temporaire . . . . . . . .: 2a04:cec0:1222:9979:e801:e4f:e798:622d
   Adresse IPv6 de liaison locale. . . . .: fe80::45a7:4c05:fd08:f52c%9
   Adresse IPv4. . . . . . . . . . . . . .: 192.168.100.3
   Masque de sous-réseau. . . . . . . . . : 255.255.255.0
   Passerelle par défaut. . . . . . . . . : fe80::b2e1:7eff:fe51:151d%9
                                    192.168.100.253

C:\Users\Administrateur>ftp 192.168.1.1
Connecté à 192.168.1.1.
220 ProFTPD Server (dmz.bl.lan) [192.168.1.1]
200 UTF-8 activÃ©
Utilisateur (192.168.1.1:(none)) : annie
331 Mot de passe requis pour annie
Mot de passe :
230 Utilisateur annie authentifié
ftp>
```

Refus depuis un autre poste

```
Carte Ethernet Ethernet :

   Suffixe DNS propre à la connexion. . . :
   Adresse IPv6. . . . . . . . . . . . . .: 2a04:cec0:1222:9979:1875:8dd8:9d1f:e9bb
   Adresse IPv6 de liaison locale. . . . .: fe80::1875:8dd8:9d1f:e9bb%9
   Adresse IPv4. . . . . . . . . . . . . .: 192.168.100.1
   Masque de sous-réseau. . . . . . . . . : 255.255.255.0
   Passerelle par défaut. . . . . . . . . : fe80::b2e1:7eff:fe51:151d%9
                                    192.168.100.253

Carte Tunnel Connexion au réseau local* 11 :

   Statut du média. . . . . . . . . . . . : Média déconnecté
   Suffixe DNS propre à la connexion. . . :

C:\Users\Administrateur.WIN-5V6S12CAN6S>ftp 192.168.1.1
> ftp: connect :Délai de connexion dépassé
ftp>
```

→ ok

WAN

Floating | WAN | LAN | OPT1 | OpenVPN

Rules (Drag to Change Order)

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✗	0/0 B	*	Reserved Not assigned by IANA	*	*	*	*	*		Block bogon networks	⚙
✓	0/0 B	IPv4 TCP	*	*	192.168.1.1	80 (HTTP)	*	none			⚓ ✎ ⧉ ⊘ 🗑 ✕
✓	0/0 B	IPv4 TCP	*	*	192.168.1.1	443 (HTTPS)	*	none			⚓ ✎ ⧉ ⊘ 🗑 ✕