PfSense 2.8.1 // OpenVPN

LA STRUCTURE PKI (Public Key Infrastructure)

L'Architecte de la confiance. Sans ça, c'est juste deux ordinateurs qui se crient des chiffres aléatoires.

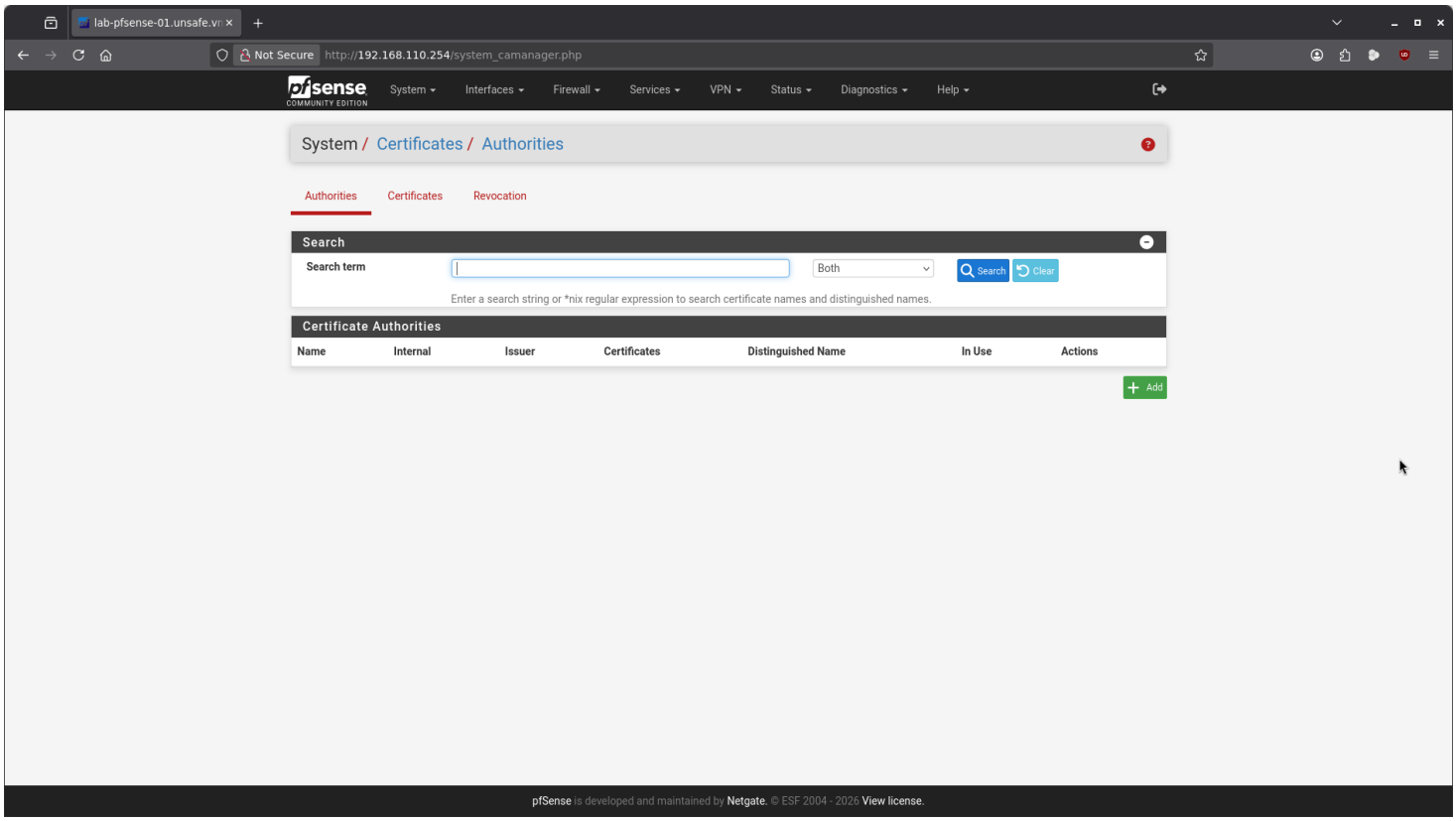
Avant de tunneliser quoi que ce soit, nous devons établir une chaîne de confiance.

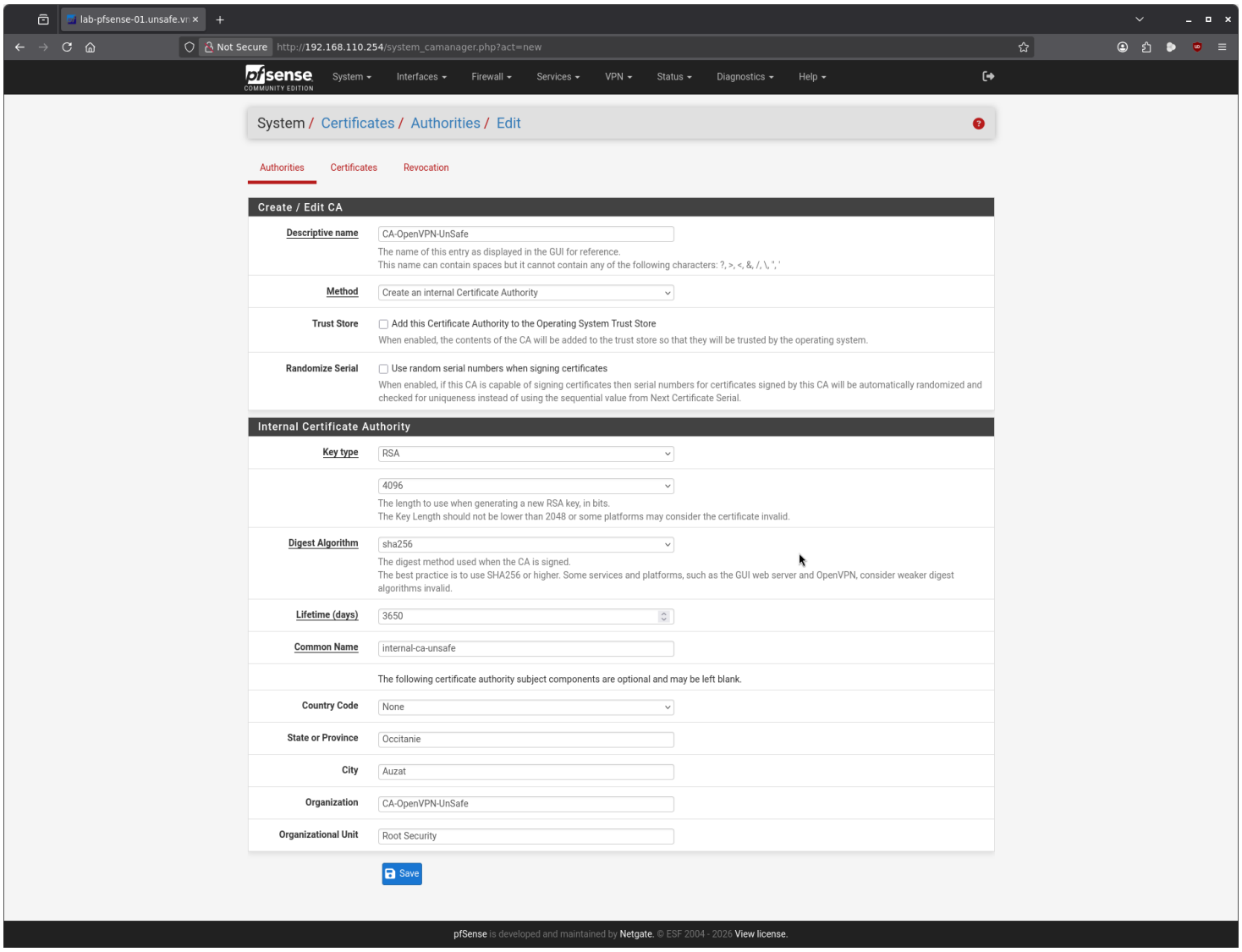
L'Autorité de Certification (CA)

C'est le Dieu de votre VPN. Il signe tout.

Menu : System > Cert. Manager > **Authorities**.

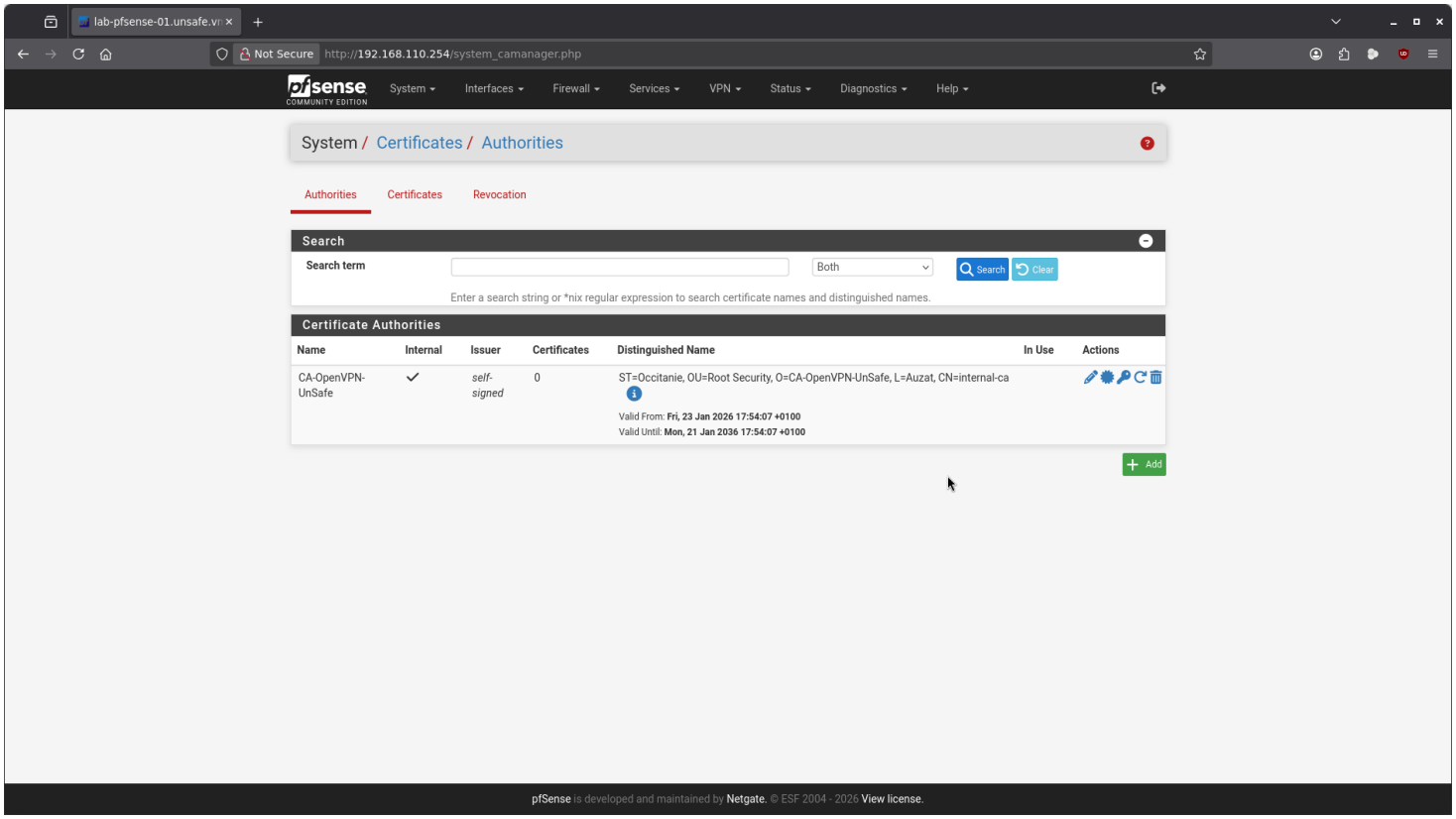
Action : + Add.





Descriptive Name	CA-OpenVPN-UnSafe	
Method	Create an internal CA	
Key Type	RSA / 4096	**Niveau de paranoïa : Élevé.** 2048 suffisait amplement, mais avec 4096, même un ordinateur quantique prendra une RTT avant de casser ça. C'est lourd, c'est lent, c'est solide. C'est validé.
Digest Algo	SHA256	
Lifetime	3650 (Jours)	10 ans. L'optimisme pur de croire que ce serveur (ou nous) existera encore dans une décennie.
Common Name	internal-ca-unsafe	
Location	Occitanie / Auzat	La Silicon Valley de l'Ariège.
Organizational Unit	Root Security	

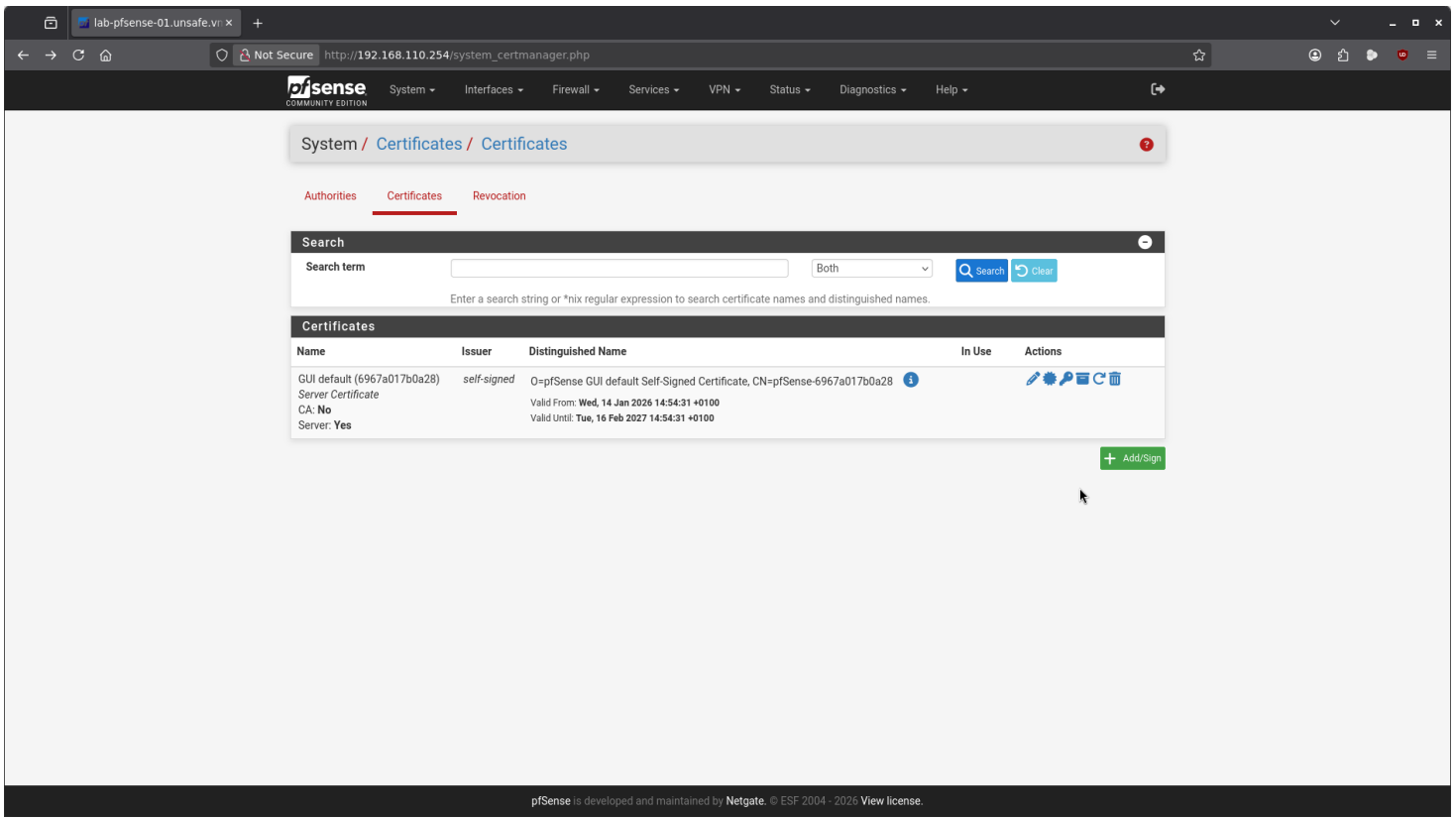
Note : Une fois sauvegardée, cette CA devient la racine de confiance. Si vous perdez la clé privée de cette CA (stockée dans pfSense), tous les futurs certificats clients/serveurs deviennent inutiles. Ne supprimez jamais cette entrée.

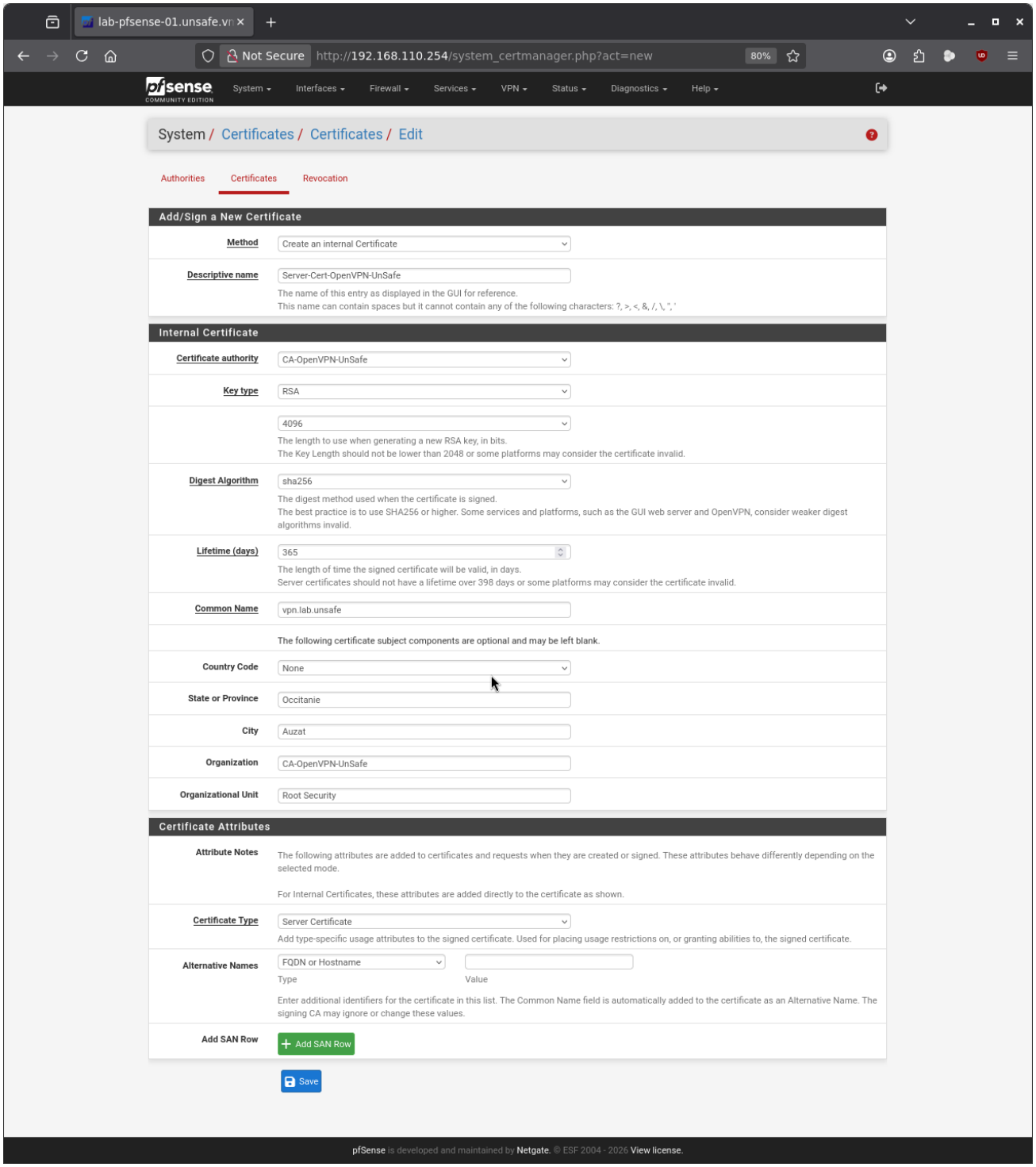


CRÉATION DU CERTIFICAT SERVEUR (L'Oracle)

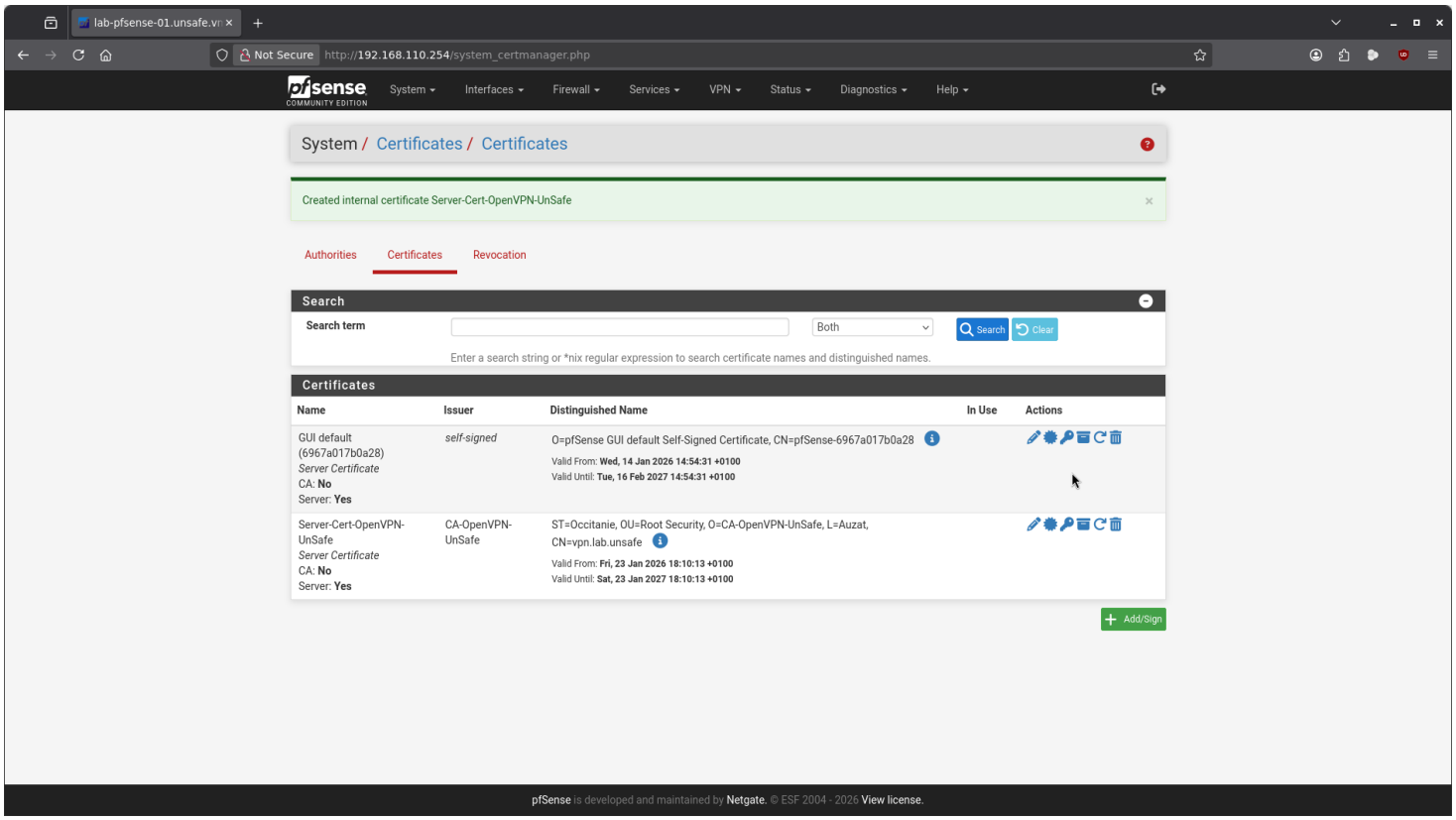
Menu : System > Cert. Manager > **Certificates**.

Action : + Add.





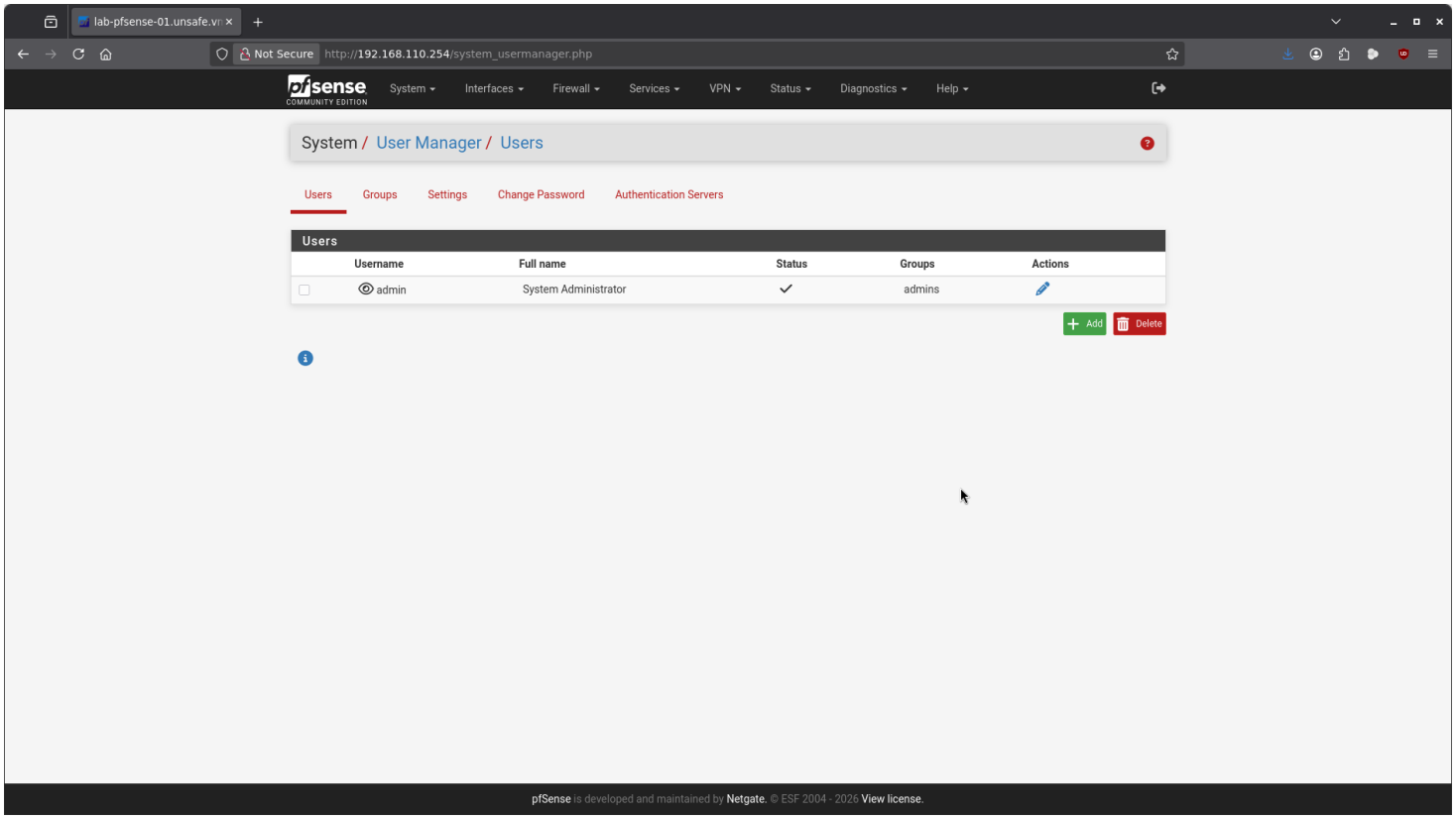
Method	Create an internal Certificate	
Descriptive Name	Server-Cert-OpenVPN-UnSafe	
Certificate Authority	CA-OpenVPN-UnSafe	**CRITIQUE.** On sélectionne notre CA créée juste avant. C'est elle qui valide.
Key Type	RSA / 4096	On reste cohérent avec la CA. Si le patron est parano (4096), l'employé doit l'être aussi.
Digest Algorithm	SHA256	
Common Name (CN)	vpn.lab.unsafe	C'est le nom de domaine complet (FQDN) du serveur. En lab, on invente. En prod, ça doit correspondre au DNS public.
Certificate Type	Server Certificate	**ALERT ROUGE.** Si vous laissez "User", le serveur aura une crise d'identité et le service ne démarrera pas.
Alternative Names	(Laisser vide)	Pas nécessaire pour ce lab, sauf si vous voulez accéder au VPN par plusieurs noms (IP et DNS).

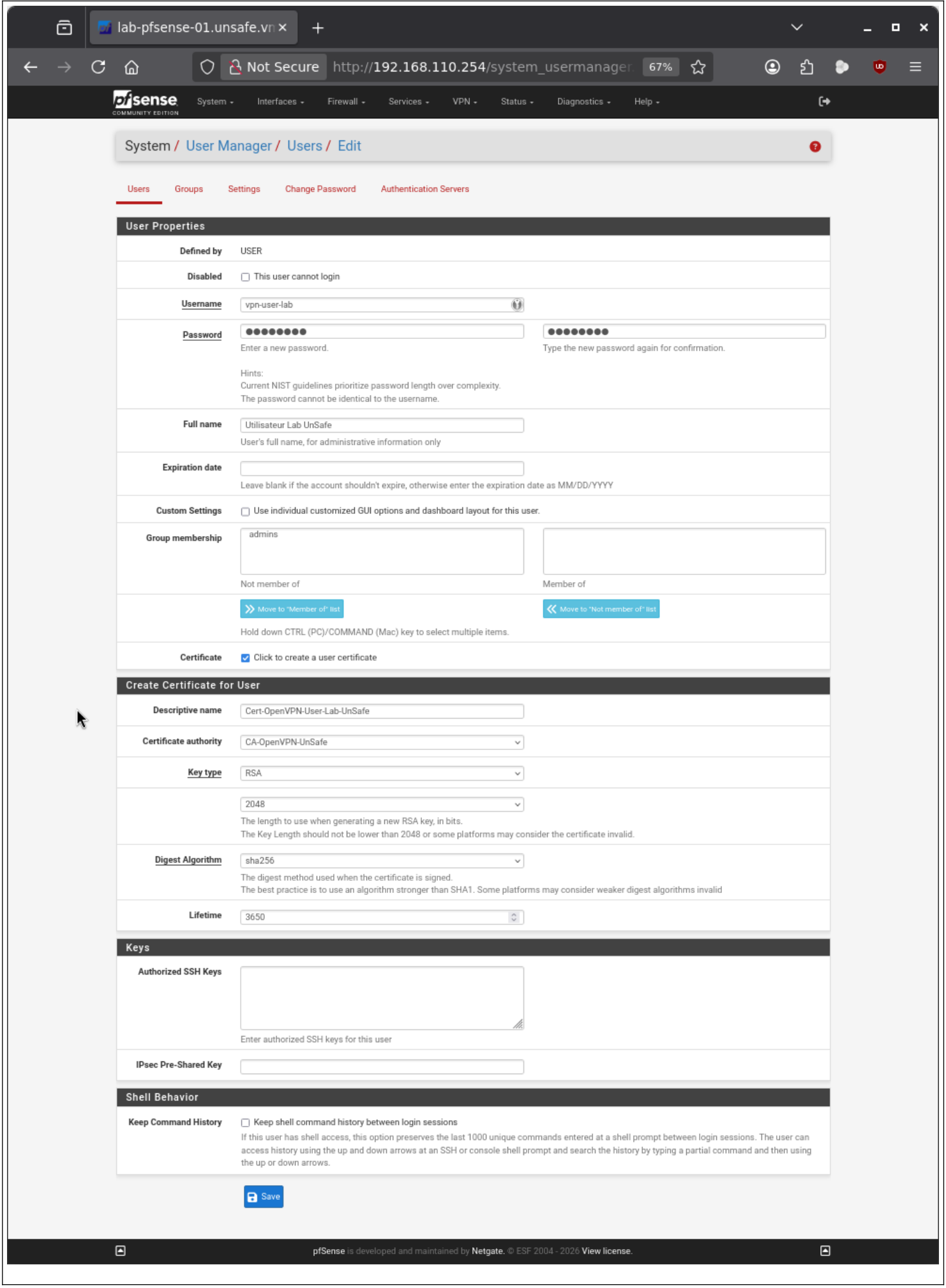


GESTION UTILISATEUR (L'Accès Client)

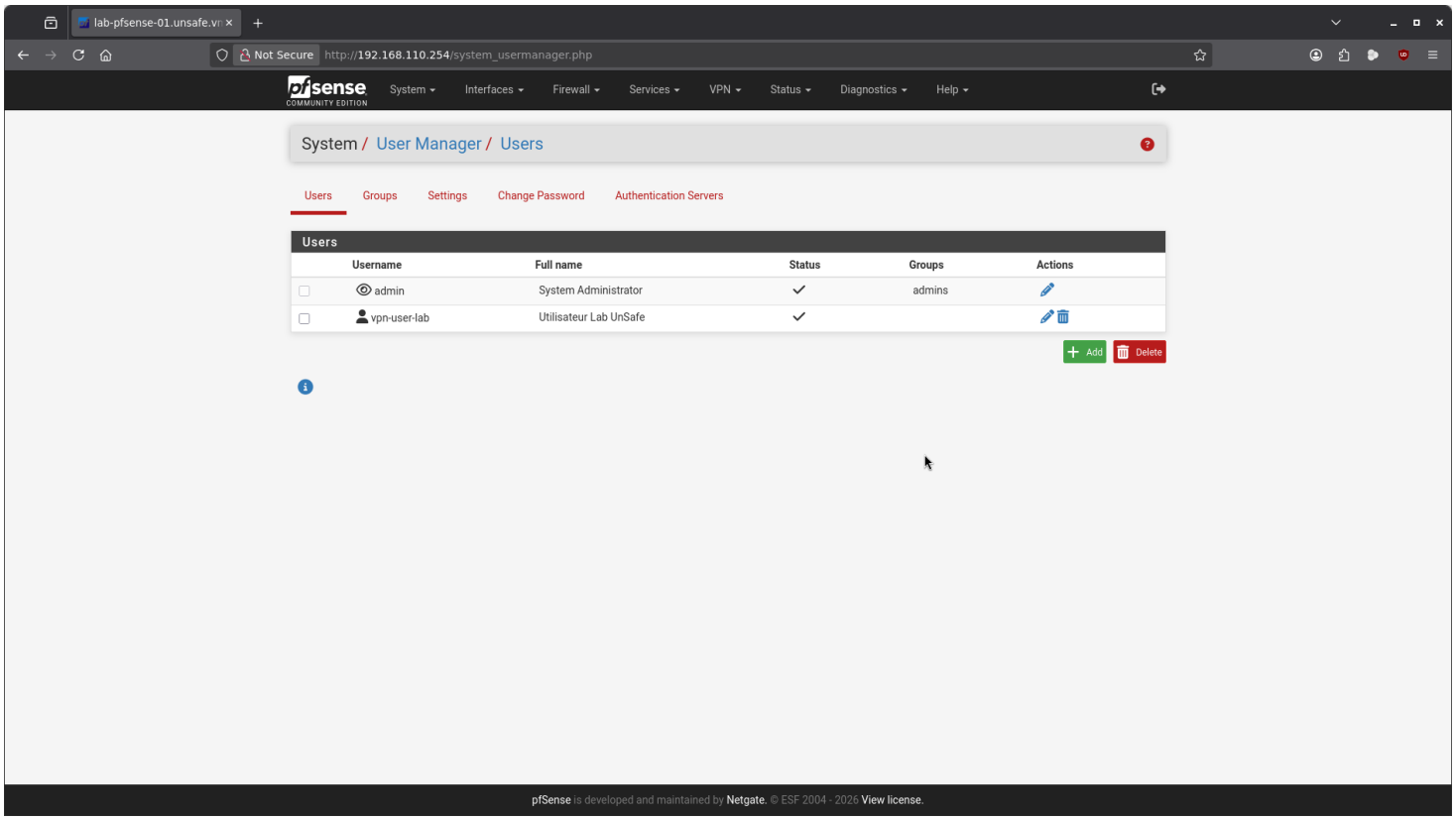
Menu : System > User Manager > **Users**.

Action : + Add.





Username	vpn-user-lab	
Password		Pas de "123456". Si vous mettez ça, Skynet gagnera.
Full Name	Utilisateur Lab UnSafe	
Certificate	[COCHER LA CASE]	**CRITIQUE.** Cochez *"Click to create a user certificate"*. C'est le bouton magique qui évite de faire 15 clics ailleurs.
Descriptive Name	Cert-OpenVPN-User-Lab-UnSafe	Le nom du fichier de clé.
Certificate Authority	CA-OpenVPN-UnSafe	Toujours le Patron. C'est lui qui signe le laissez-passer.
Key Length	2048	Ici, 2048 suffit pour un client. Mais si vous êtes en mode "Forteresse", mettez 4096 pour être raccord avec le serveur.

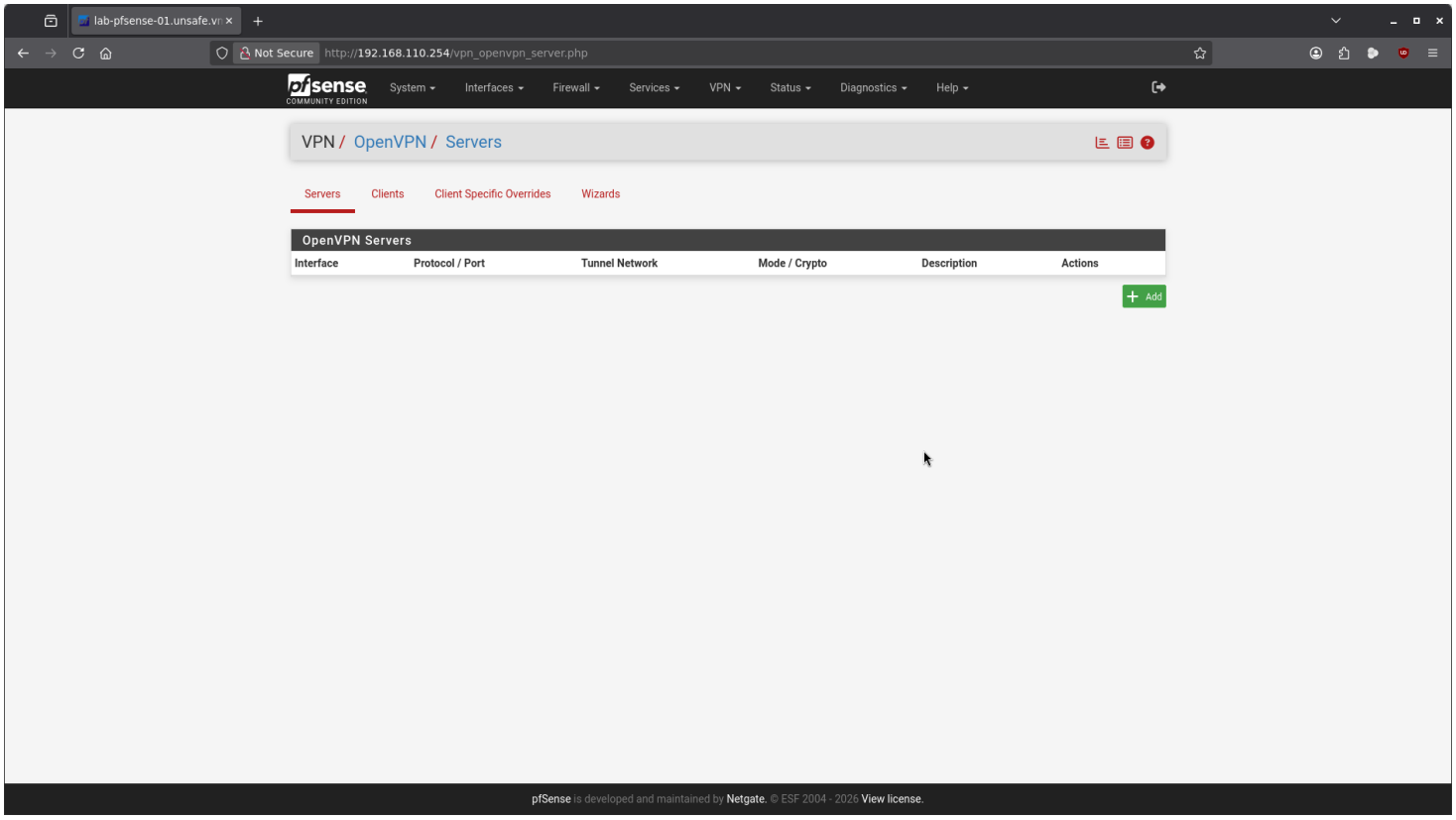


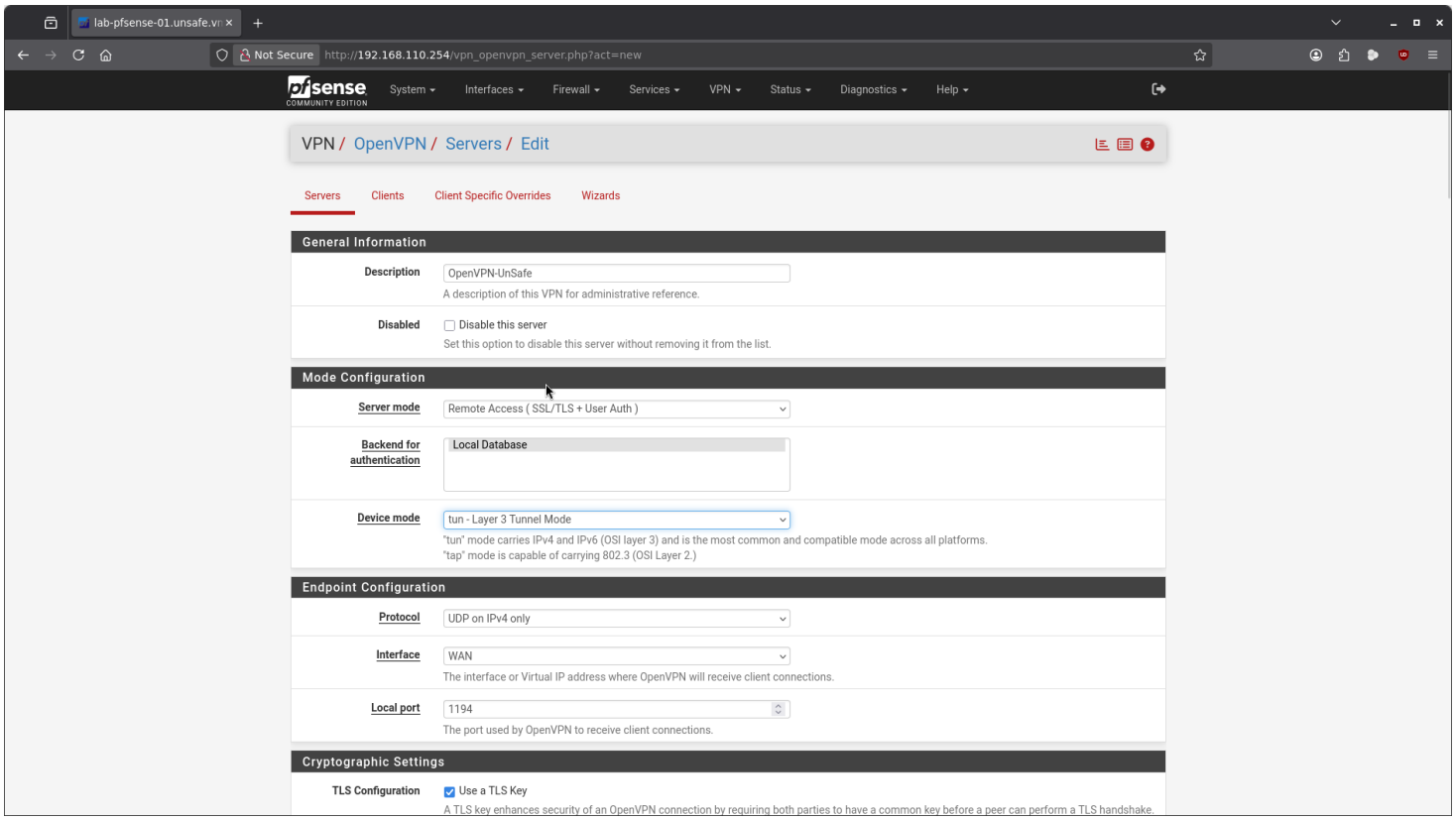
CONFIGURATION DU SERVEUR OPENVPN (Le Tunnel)

Menu : VPN > OpenVPN > **Servers**.

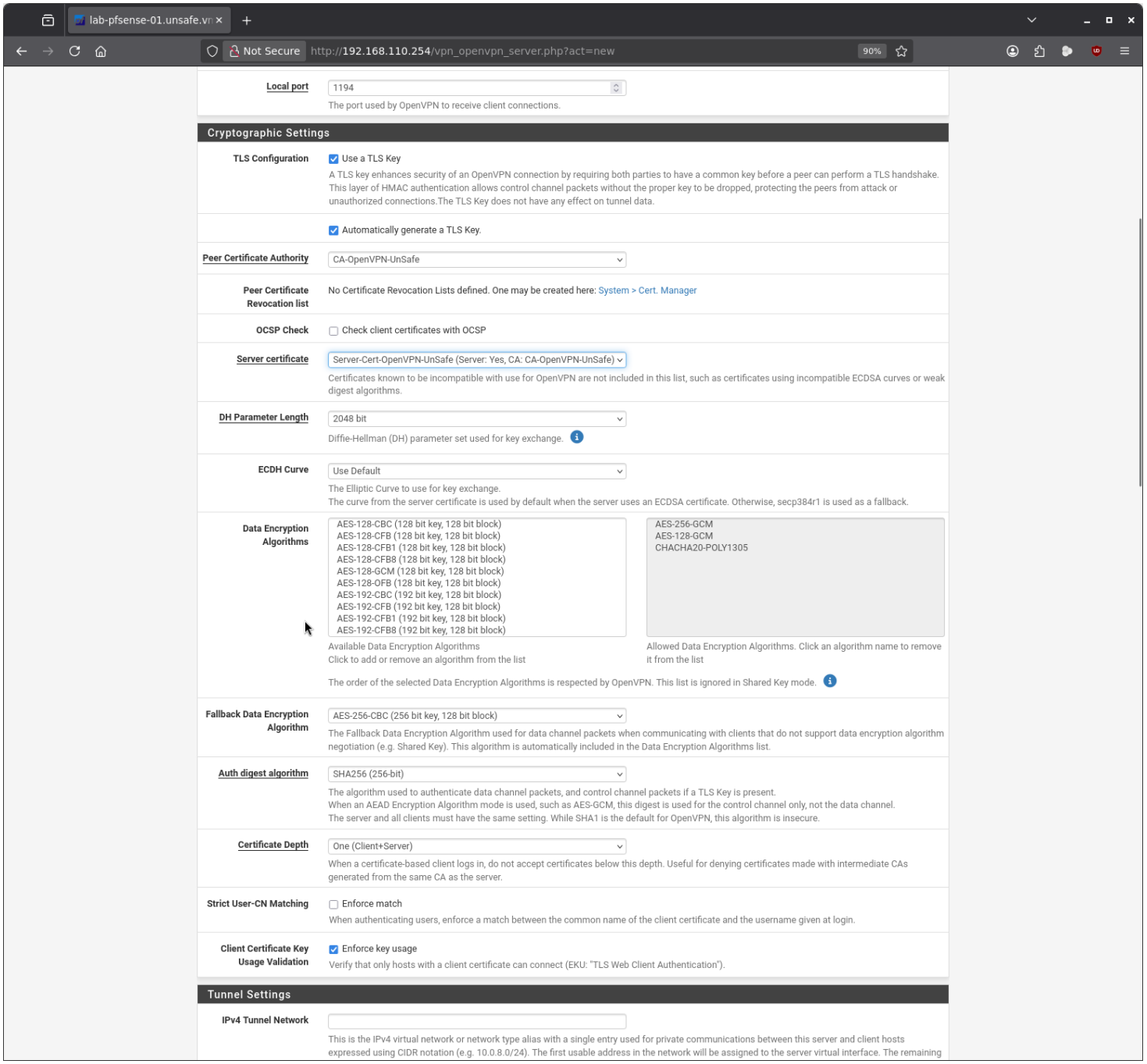
Action : + Add.

C'est le cœur du réacteur. Suivez ces instructions à la lettre. Une erreur ici et votre VPN sera aussi utile qu'un pare-feu en papier mâché.

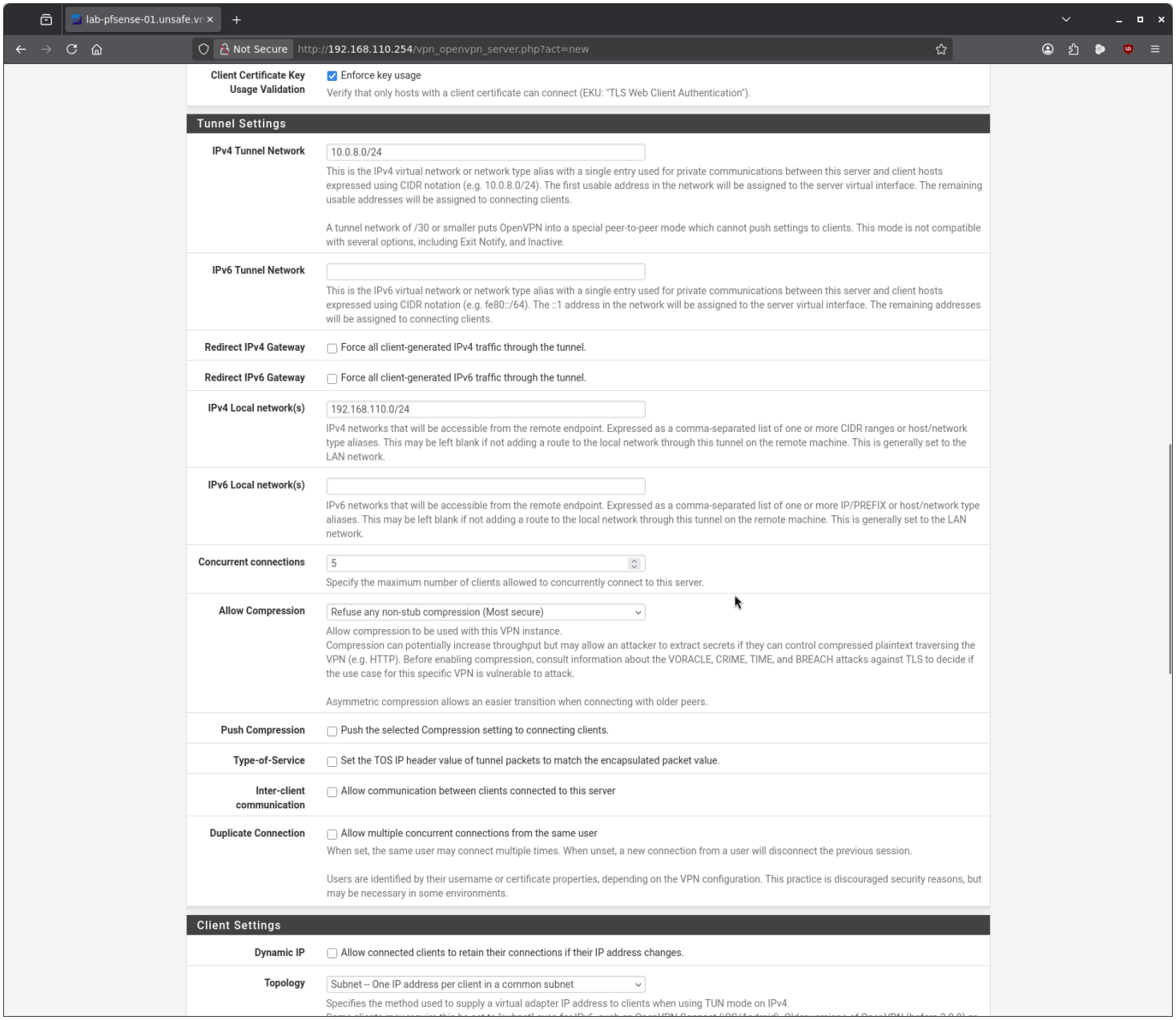




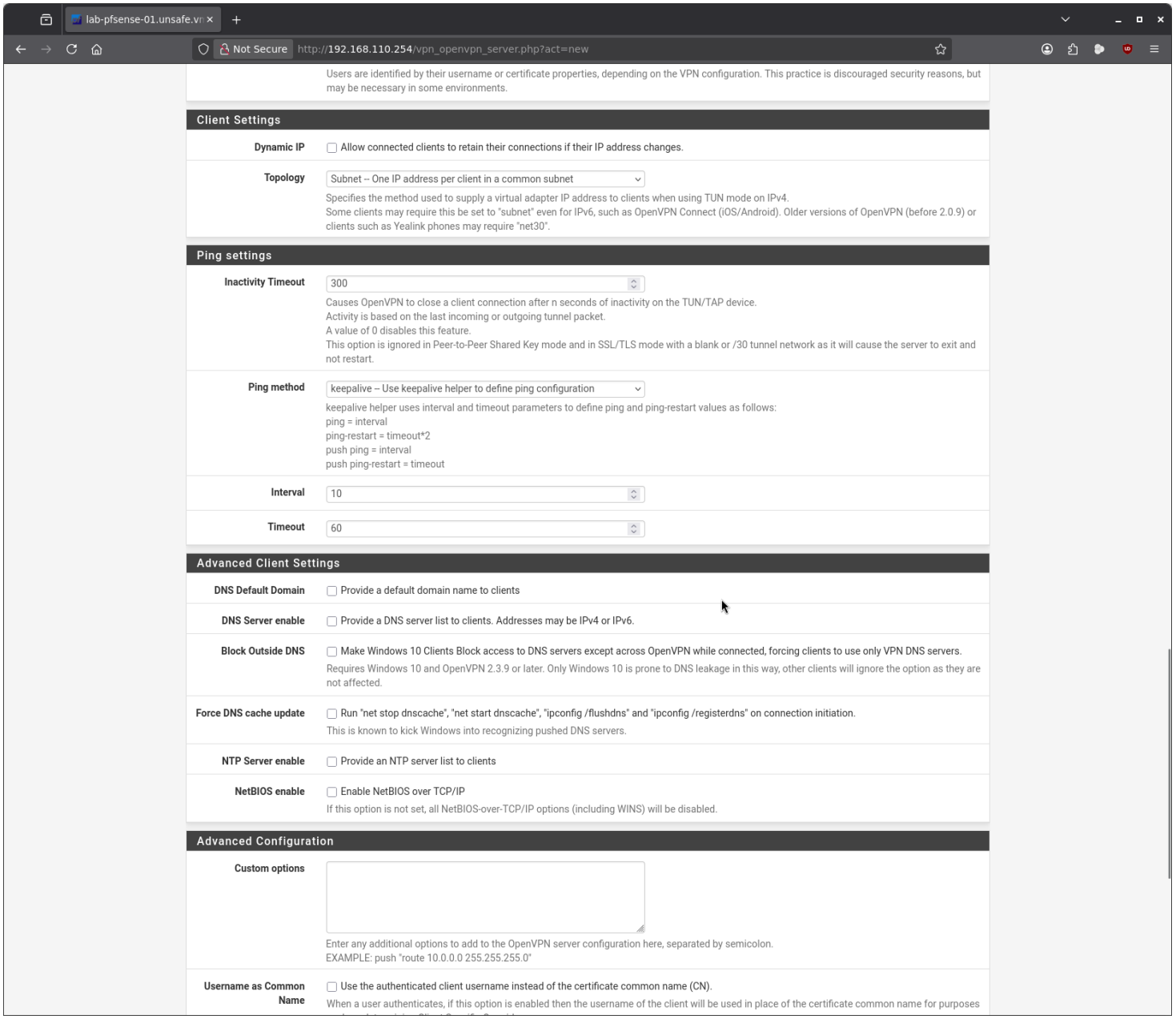
Description	OpenVPN-UnSafe	
Server Mode	Remote Access (SSL/TLS + User Auth)	Ceinture et bretelles. On veut le certificat de la machine **ET** le mot de passe de l'humain. Sécurité maximale.
Backend for Authentication	Local Database	On utilise la base locale (là où vit notre ami vpn-user-lab).
Device Mode	tun - Layer 3 Tunnel Mode	
Protocol	UDP on IPv4 only	**UDP** est vital pour la vitesse. TCP sur TCP, c'est l'enfer de la latence (le fameux "TCP Meltdown").
Interface	WAN	C'est la porte d'entrée publique.
Local Port	1194	Le classique.

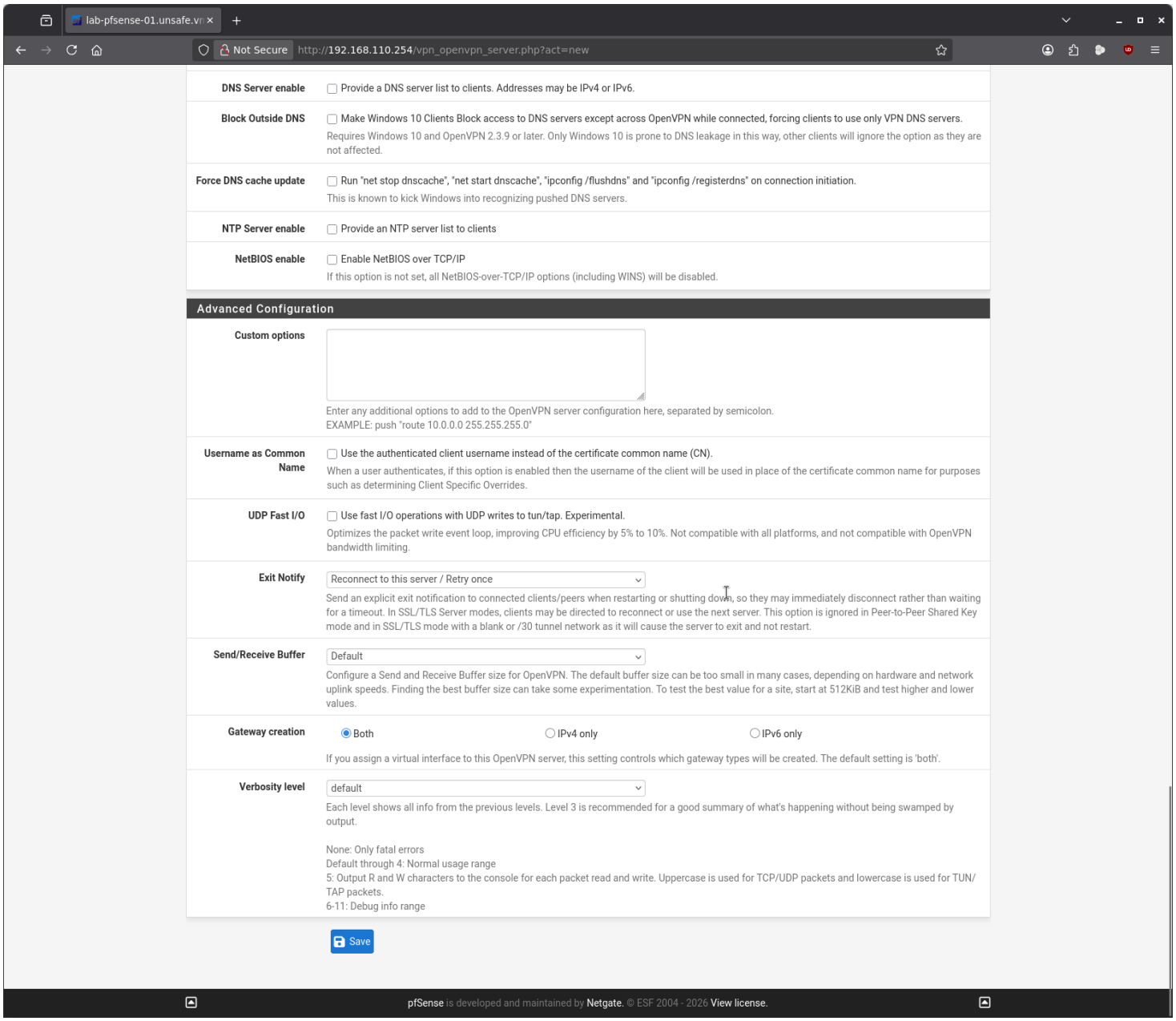


Peer Certificate Authority	CA-OpenVPN-UnSafe	On dit au serveur : "Fais confiance uniquement aux gens qui ont un badge signé par le Patron".
Server Certificate	Server-Cert-OpenVPN-UnSafe	**ATTENTION** : ne selectionner pas le certificats de l'utilisateur !
DH Parameter Length	2048	
Auth Digest Algorithm	SHA256	

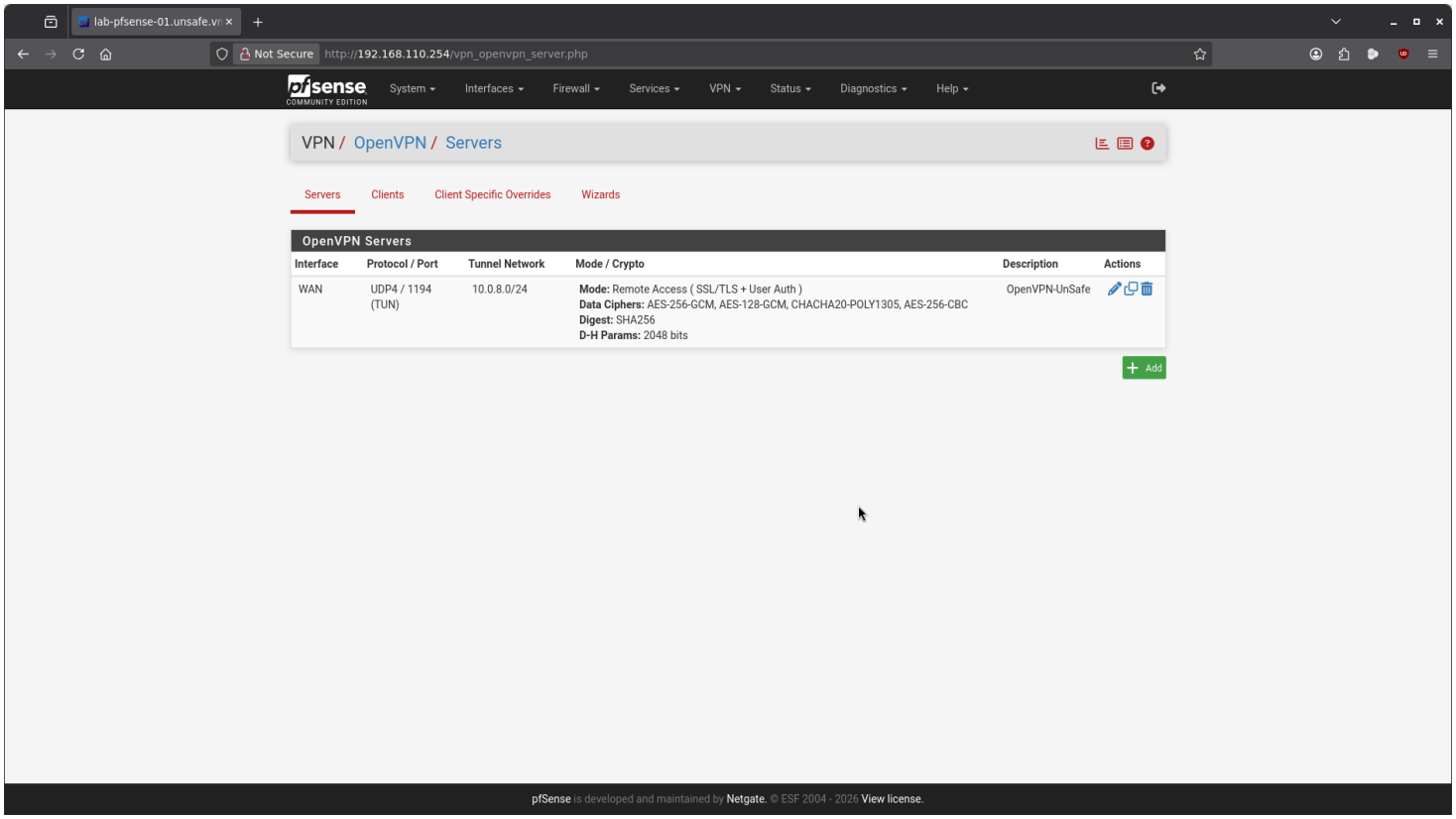


IPv4 Tunnel Network	10.0.8.0/24	C'est le réseau virtuel *à l'intérieur* du VPN. Il **DOIT** être différent de votre LAN. Si vous mettez 192.168.110.0/24 ici, tout va planter (conflit de routage).
IPv4 Local Network(s)	192.168.110.0/24	C'est le réseau LAN auquel vous voulez accéder. C'est la "récompense" une fois connecté.
Concurrent Connections	5	
Allow Compression	Refuse any non-stub compression (Most secure)	





Pour le reste gardez exactement ce que vous avez à l'écran.
Descendez tout en bas.
Cliquez sur **Save**.



LA SÉCURITÉ (Firewall Rules)

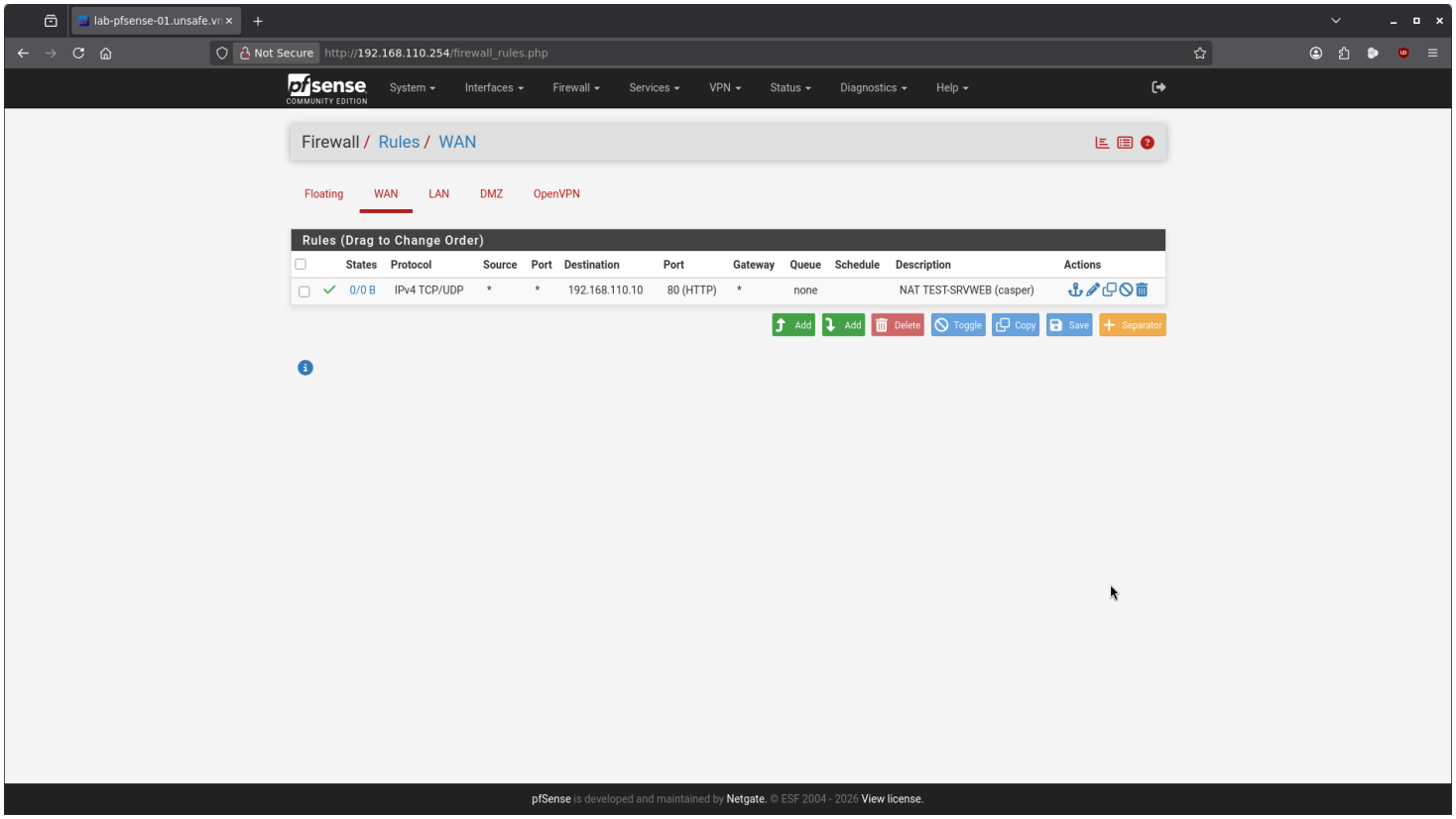
L'ouverture des vannes.

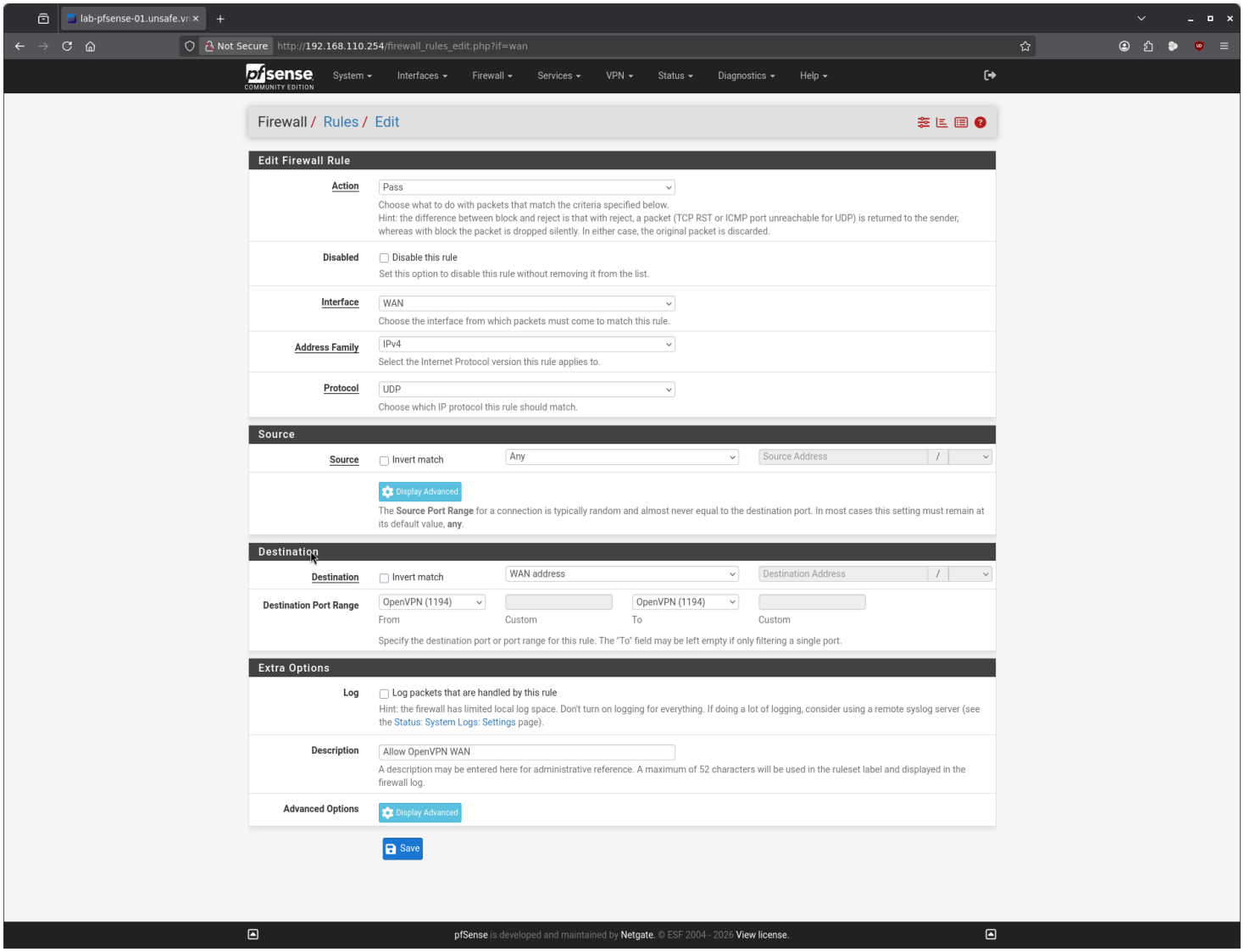
La Porte d'Entrée (WAN)

Il faut autoriser les paquets chiffrés à toucher le service OpenVPN.

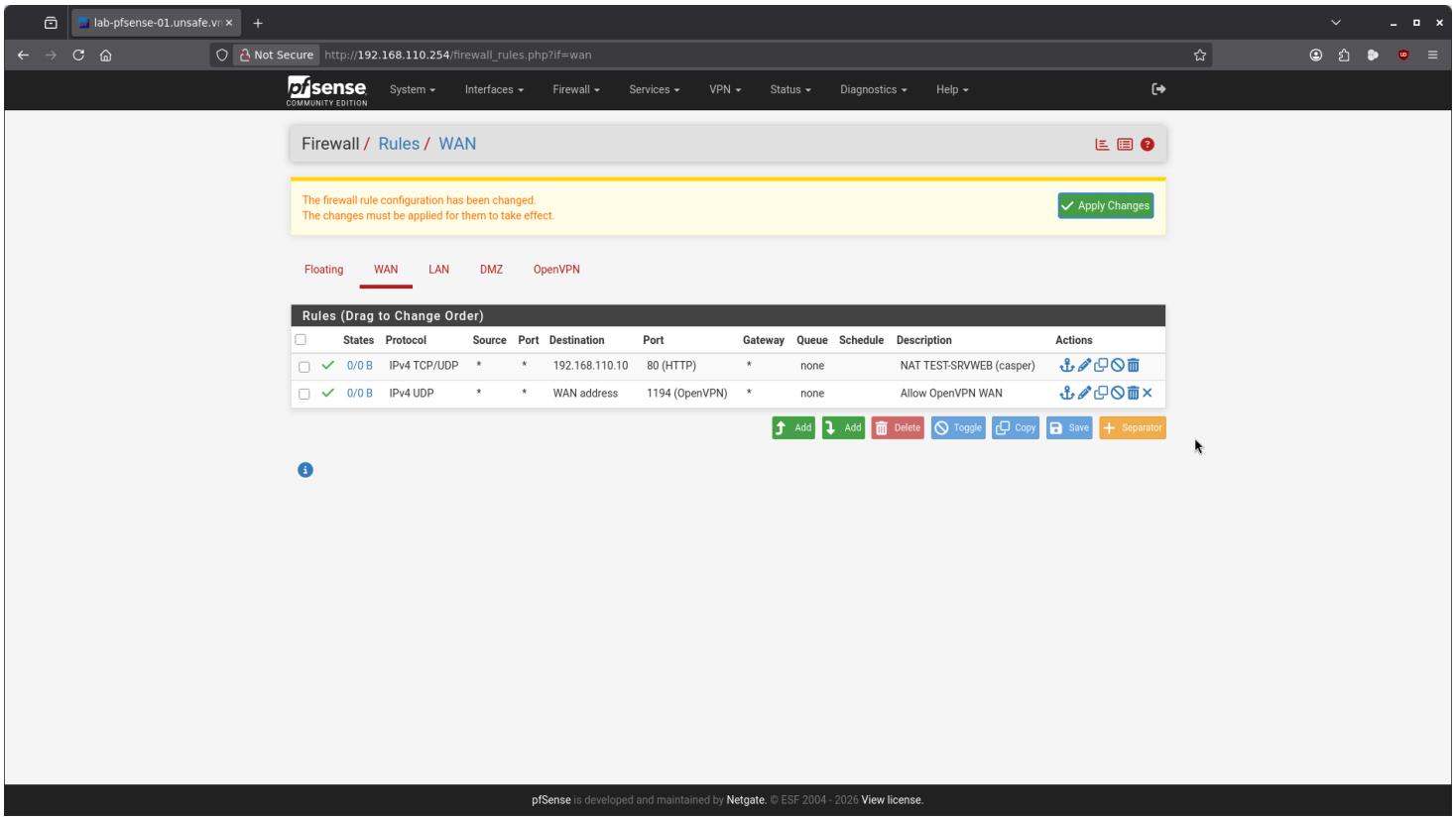
Menu : Firewall > Rules > **WAN**.

Action : + Add (Flèche vers le bas pour mettre à la fin, ou haut, peu importe en Lab).

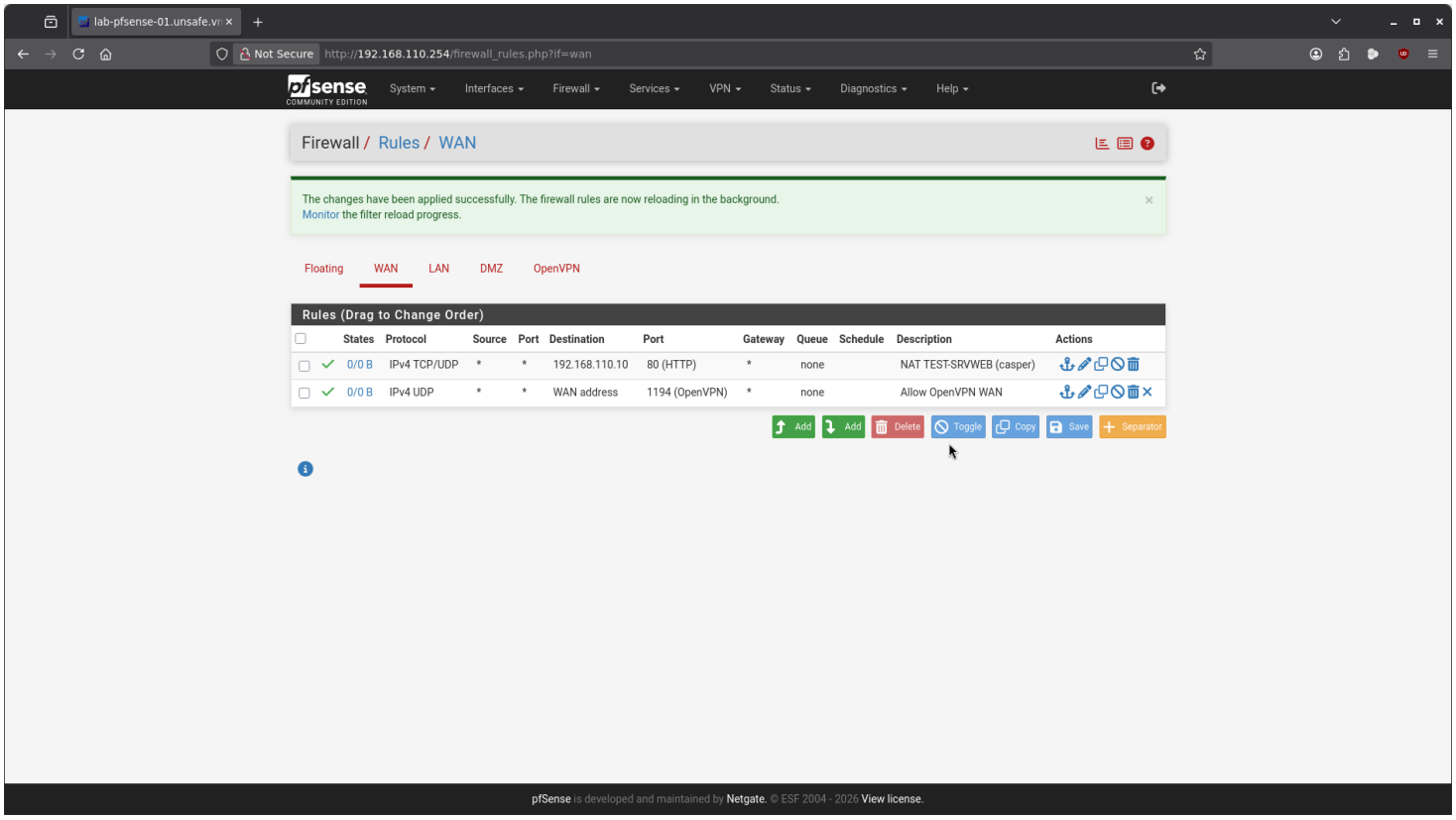




Action	Pass	On laisse entrer.
Interface	WAN	Logique.
Address Family	IPv4	On reste simple.
Protocol	UDP	**Vital.** Si vous mettez TCP ici alors que le serveur est en UDP, ça ne marchera jamais.
Source	Any	N'importe qui sur internet (ou votre réseau amont) peut tenter sa chance.
Destination	WAN address	
Destination Port Range	1194 (From) / 1194 (To)	Tapez "1194" ou cherchez "OpenVPN" dans la liste.
Log		
Description	Allow OpenVPN WAN	Pour se souvenir pourquoi on a ouvert ce port.



On n'oublie pas d'appliquer les changements !

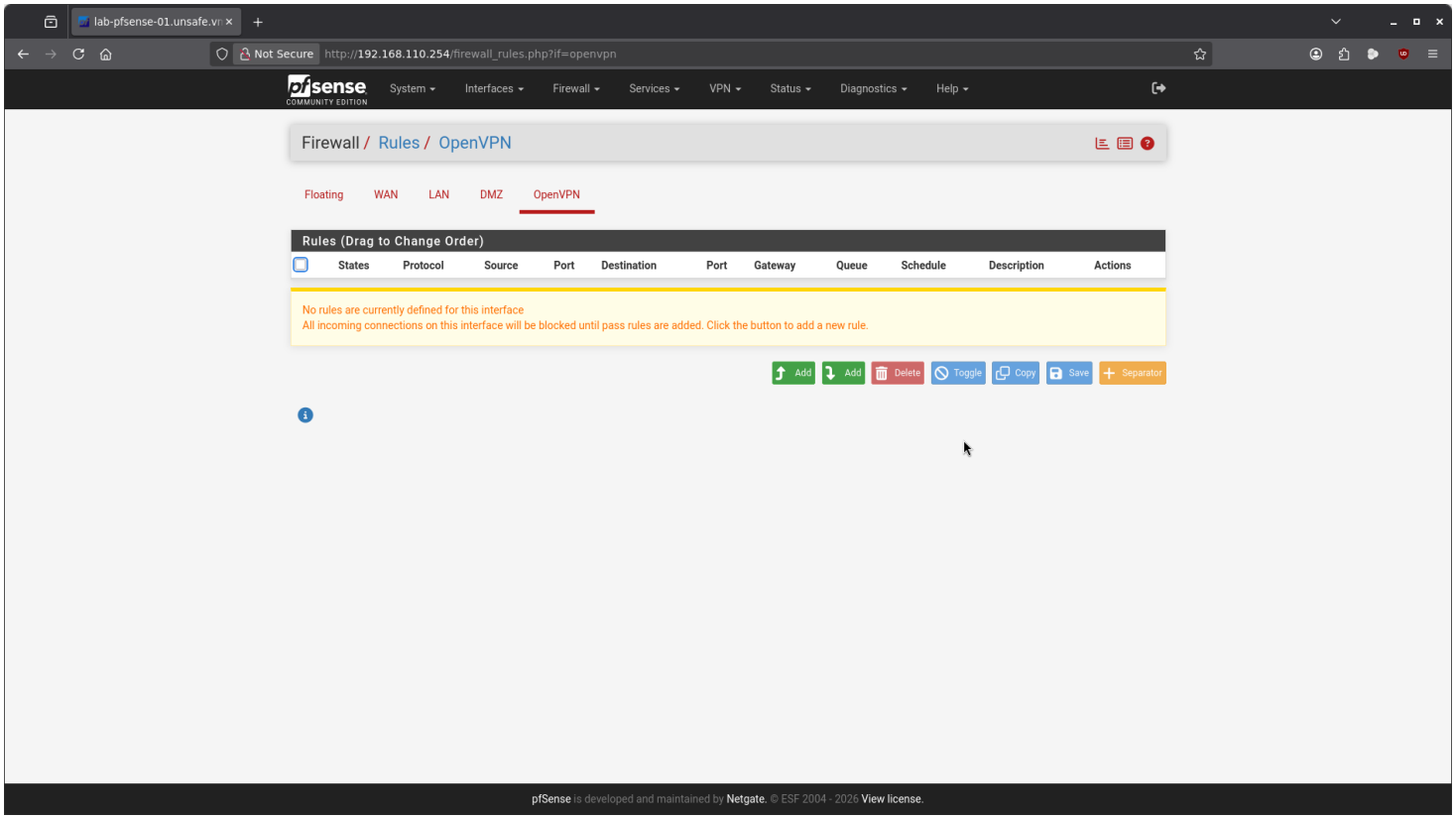


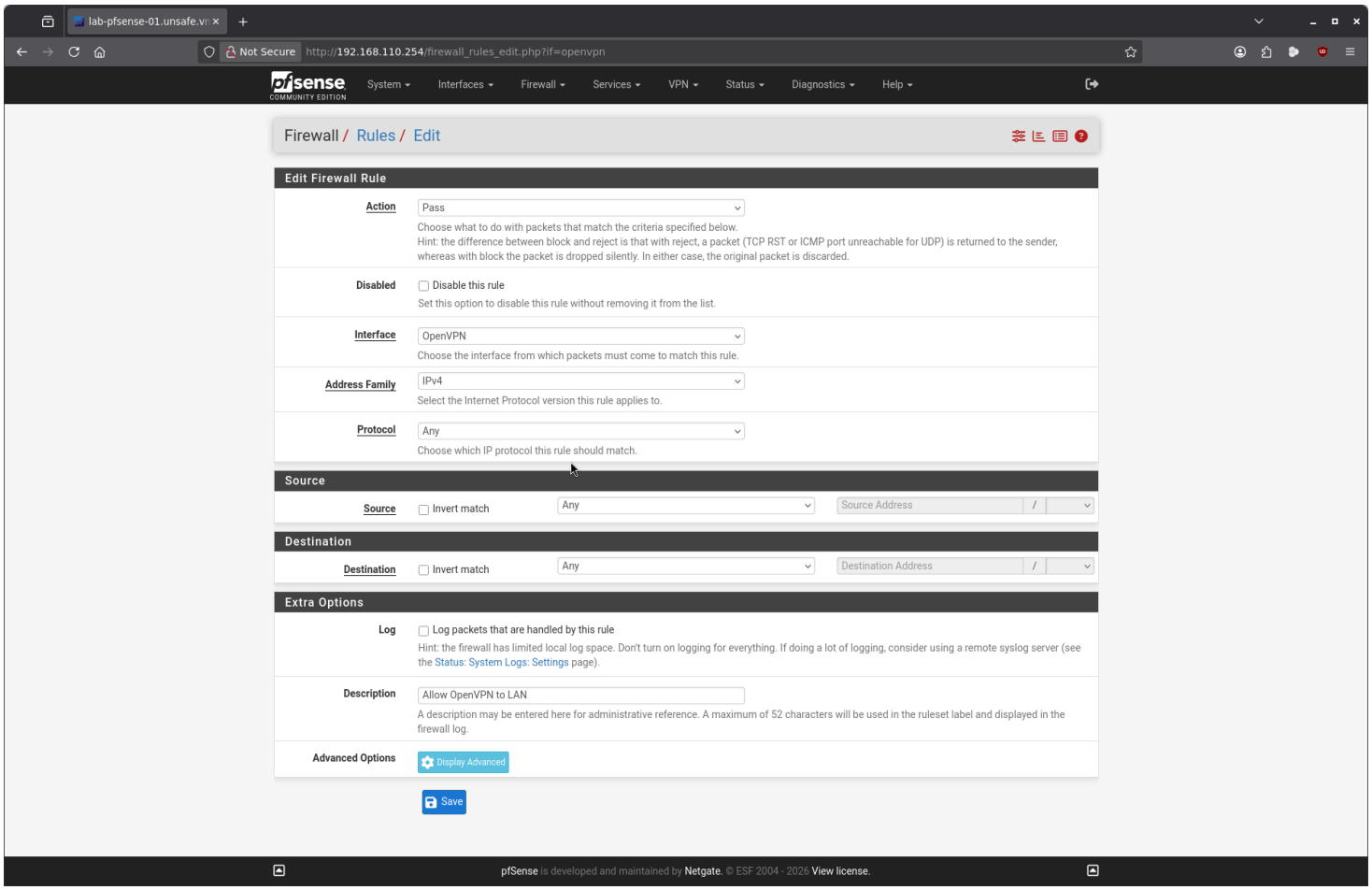
La Liberté de Mouvement (Interface OpenVPN)

Une fois le client connecté et authentifié, il est "dans" le tunnel. Mais pfSense bloque tout par défaut *à l'intérieur* aussi. Il faut lui donner le droit de parler au LAN.

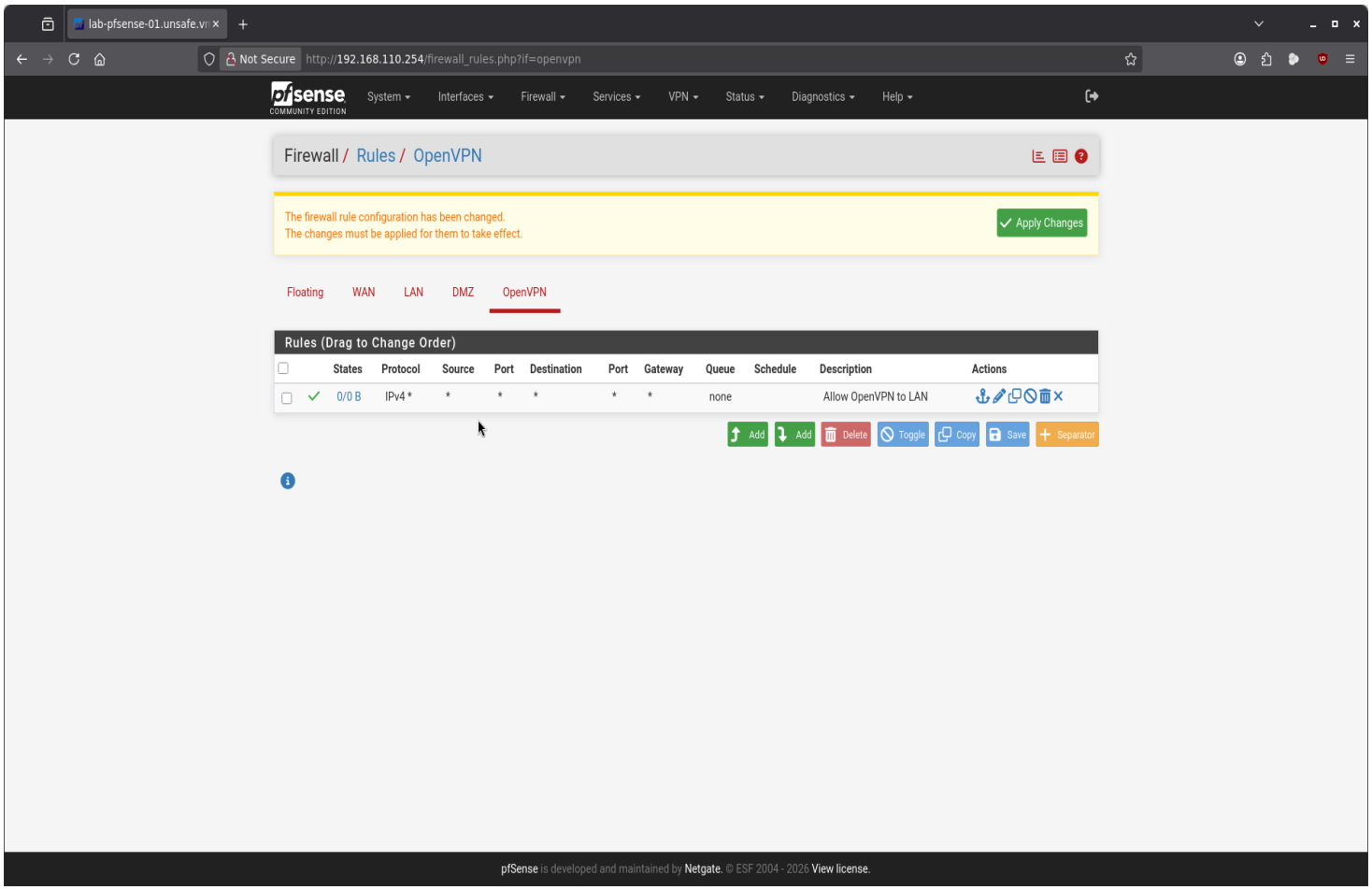
Menu : Firewall > Rules > **OpenVPN**.

Action : + Add.

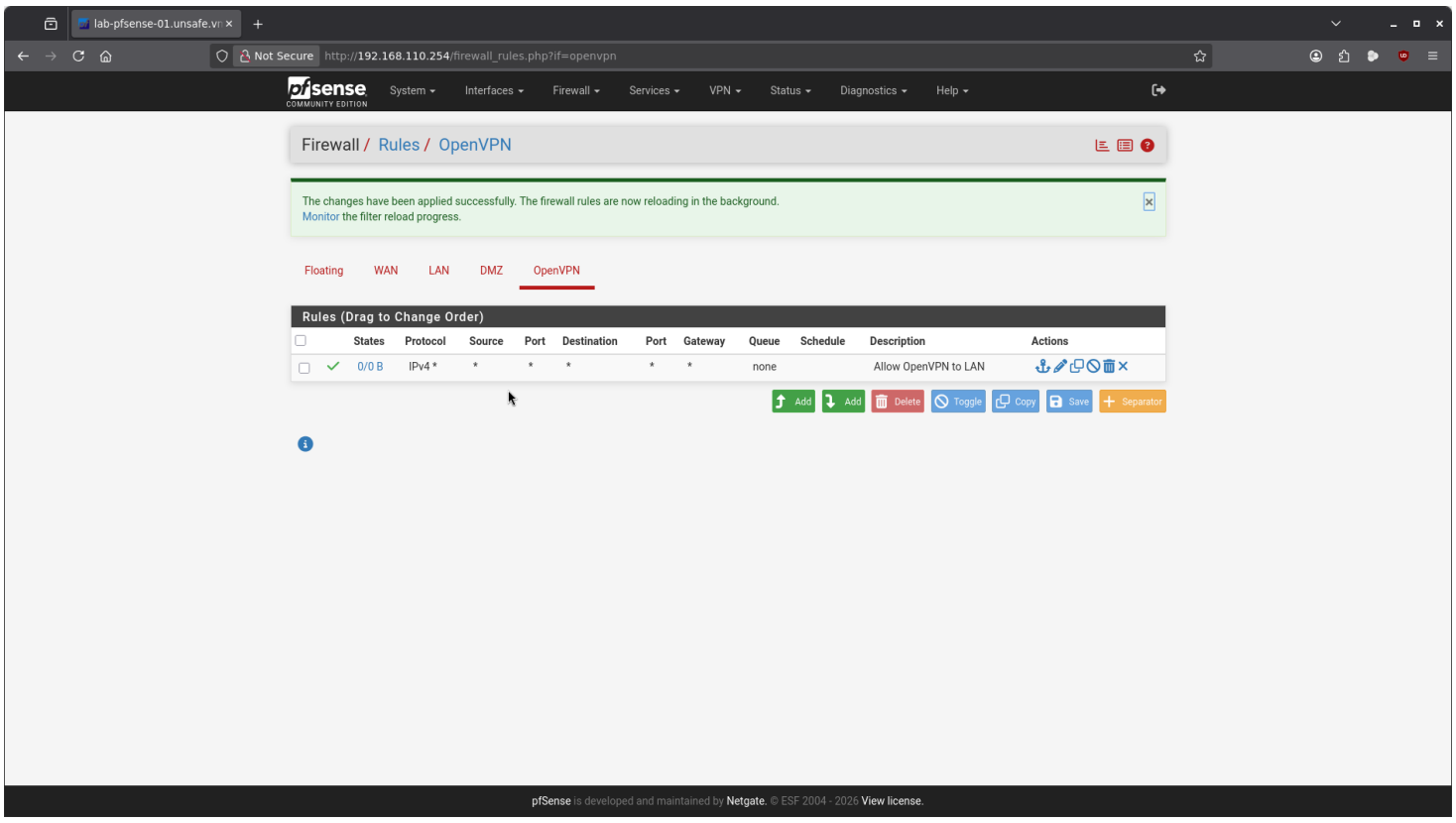




Action	Pass	
Interface	OpenVPN	C'est l'onglet magique qui apparaît quand on crée un serveur.
Protocol	Any	On laisse tout passer (TCP, UDP, ICMP...).
Source	Any	
Destination	Any	
Description	Allow OpenVPN to LAN	**Mode Lab.** En production, on restreindrait l'accès à des IP précises. Ici, on est des cow-boys, on ouvre tout ("Any / Any").



On n'oublie pas d’appliquer les changements !



DÉPLOIEMENT CLIENT (Le Package Magique)

Exportation de la charge utile pour l'agent de terrain.

Pour éviter de copier-coller des clés RSA de 4096 bits à la main (et de faire une erreur de syntaxe à la ligne 342), nous utilisons l'outil d'exportation.

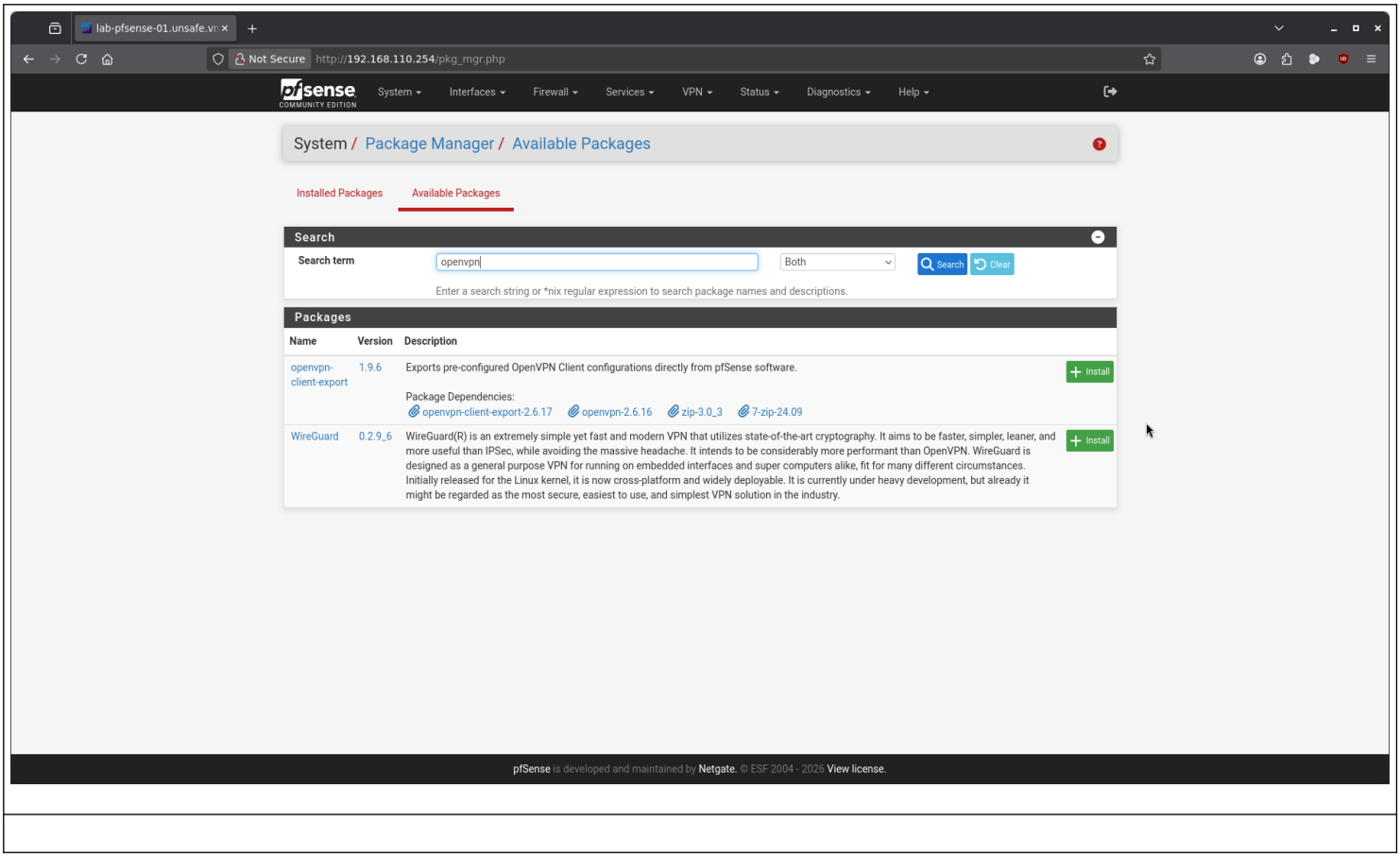
Installation de l'outil

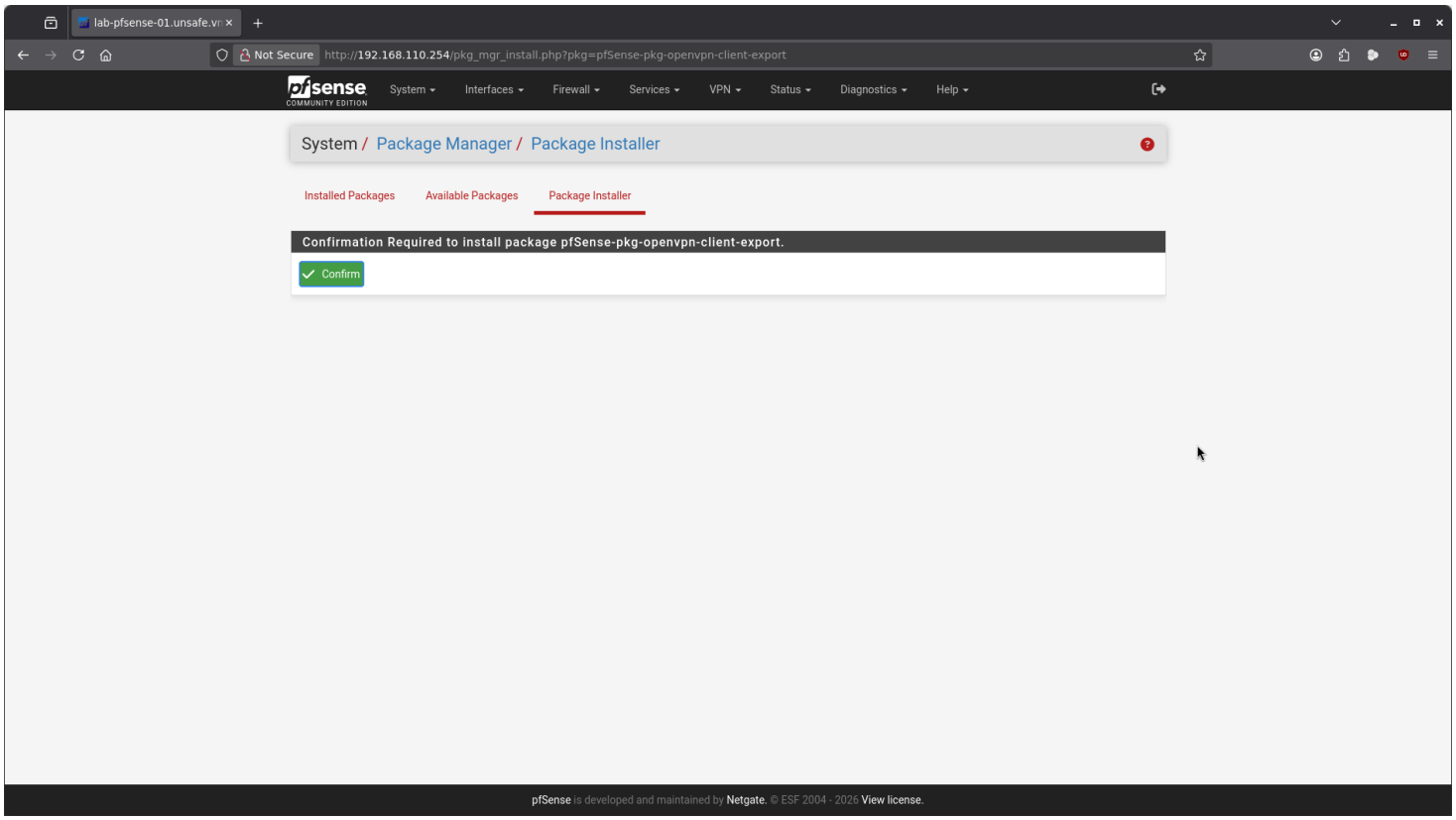
Menu : System > Package Manager > Available Packages.

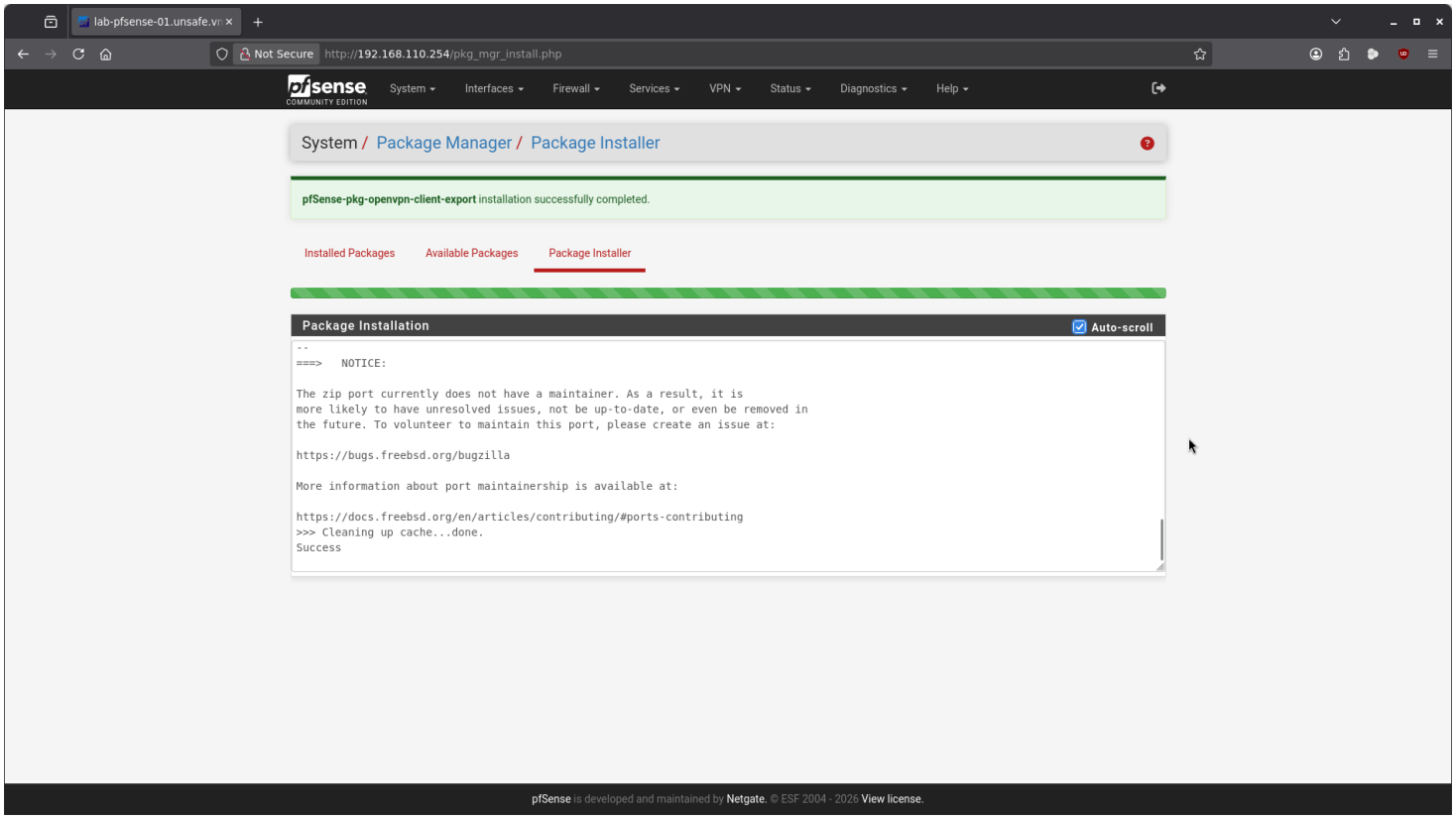
Recherche : openvpn-client-export.

Action : Cliquez sur Install > Confirm.

Attendez que pfSense finisse de mâchouiller les fichiers (Success).





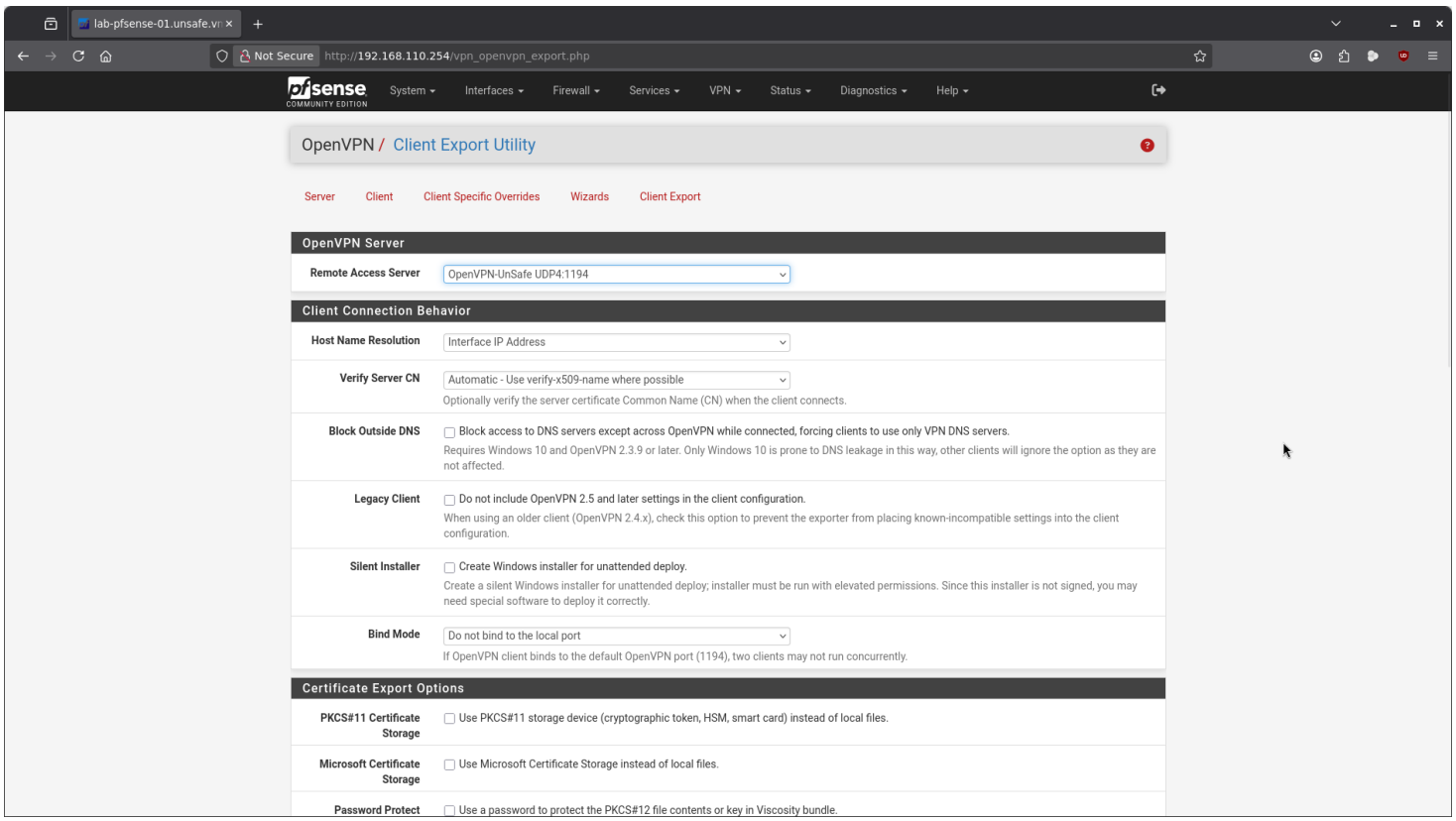


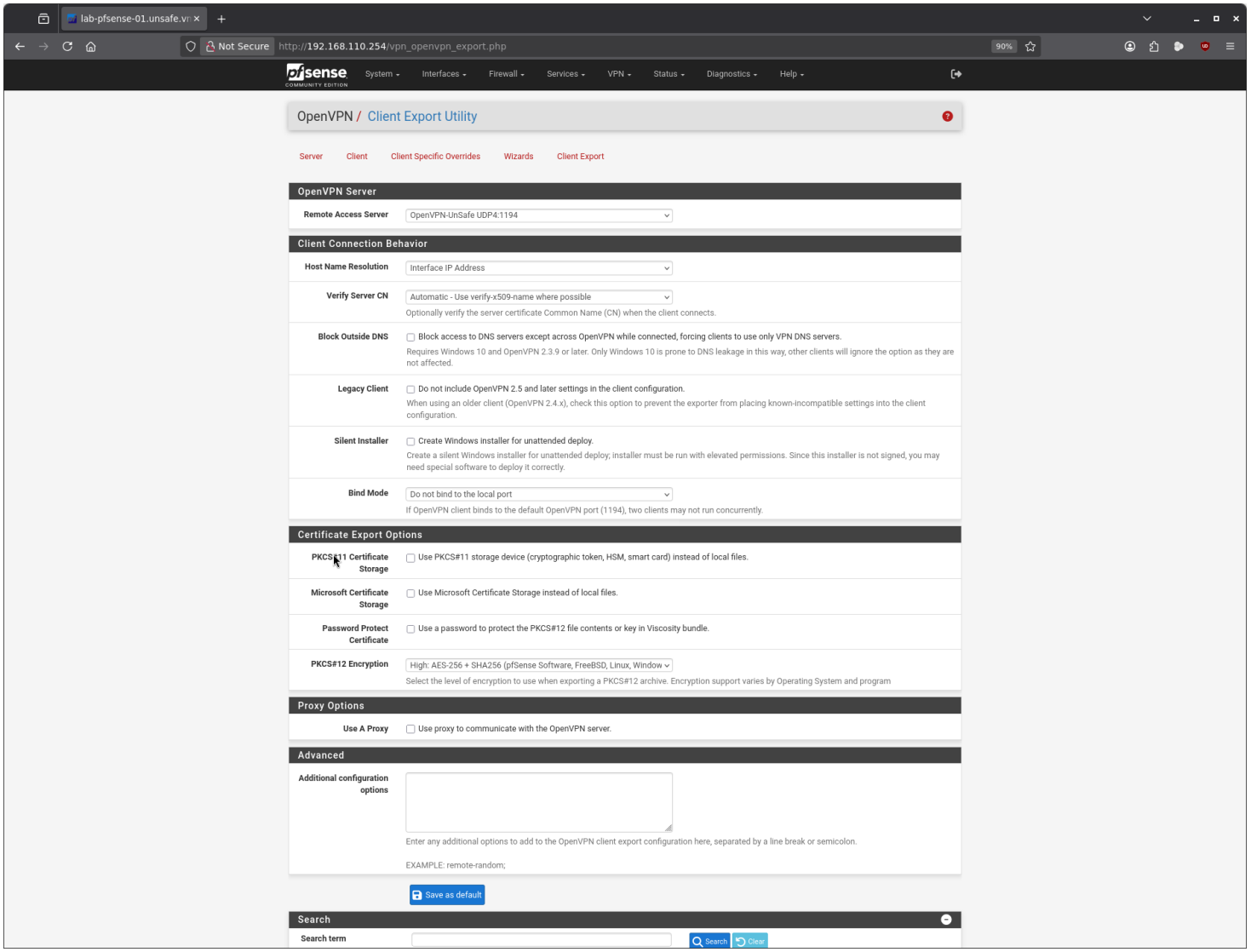
Configuration de l'Export

Une fois installé, un nouvel onglet apparaît.

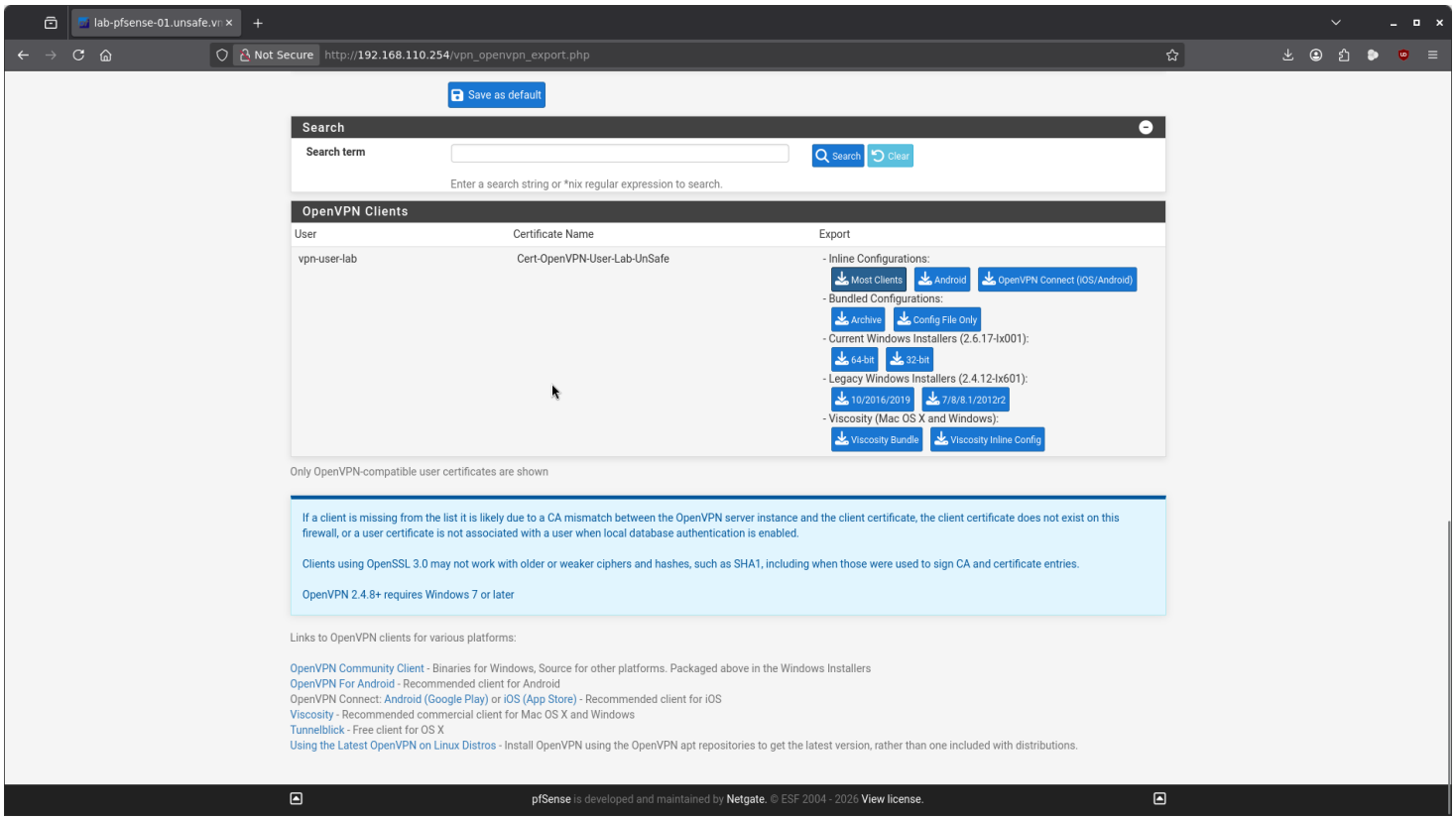
Menu : VPN > OpenVPN > **Client Export**.

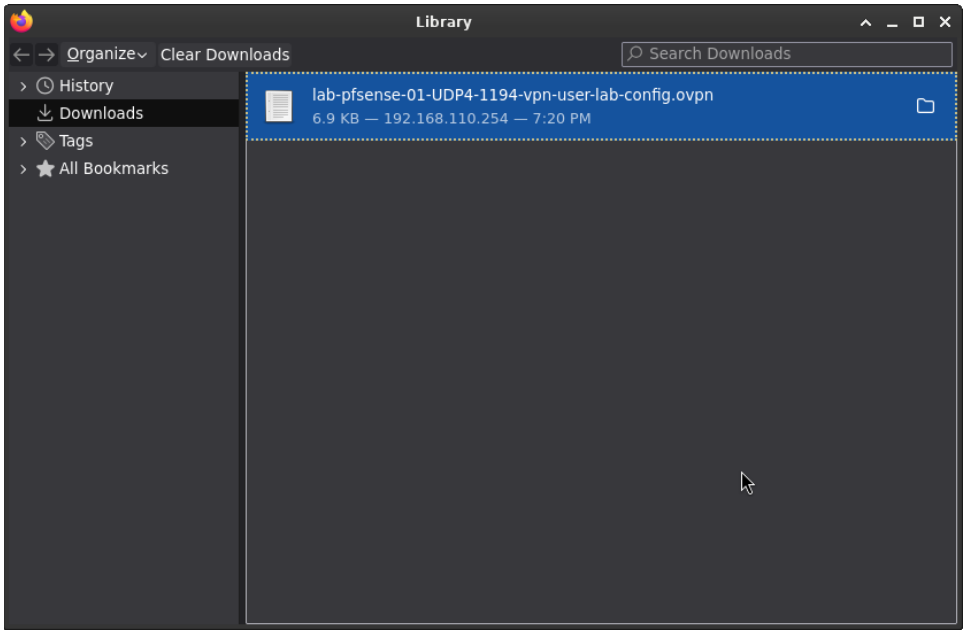
C'est ici qu'on prépare le fichier .ovpn.





Remote Access Server	OpenVPN-UnSafe UDP4:1194	Vérifiez que c'est bien votre serveur qui est sélectionné.
Host Name Resolution	Interface IP Address	
Verify Server CN	Automatic	



Bouton Download	Localisez votre utilisateur **vpn-user-lab** en bas.	
Type de fichier	Most Clients (ou Inline Configurations)	Cela télécharge un fichier unique .ovpn qui contient TOUT (CA, Cert, Clé, Config). C'est le "Happy Meal" du VPN.
		

TEST FINAL (L'Injection Debian)

Le moment de vérité.

Transférez le fichier .ovpn téléchargé vers votre VM Debian (via scp, dossier partagé, ou clé USB virtuelle).

```
jesus@legion:~$ scp Downloads/lab-pfsense-01-UDP4-1194-vpn-user-lab-config.ovpn admin@192.168.100.142:~/
admin@192.168.100.142's password:
lab-pfsense-01-UDP4-1194-vpn-user-lab-config.ovpn
100% 7019    15.2MB/s   00:00
```

```
admin@lab-debian-54:~$ ls -l
total 8
-rw-rw-r-- 1 admin admin 7019 Jan 23 19:35 lab-pfsense-01-UDP4-1194-vpn-user-lab-config.ovpn
```

Installation d’OpenVPN

```
admin@lab-debian-54:~$ sudo apt update && sudo apt install openvpn
[sudo] password for admin:
Hit:1 http://deb.debian.org/debian trixie InRelease
Get:2 http://security.debian.org/debian-security trixie-security InRelease [43.4 kB]
Get:3 http://deb.debian.org/debian trixie-updates InRelease [47.3 kB]
Get:4 http://security.debian.org/debian-security trixie-security/main Sources [119 kB]
Get:5 http://security.debian.org/debian-security trixie-security/main amd64 Packages [97.9 kB]
Get:6 http://security.debian.org/debian-security trixie-security/main Translation-en [62.5 kB]
Fetched 370 kB in 0s (1,760 kB/s)
5 packages can be upgraded. Run 'apt list --upgradable' to see them.
Installing:
  openvpn

Installing dependencies:
  easy-rsa  libduktape207  liblzo2-2    libnl-genl-3-200  libpkcs11-helper1t64  libpolkit-gobject-1-0  opensc-
pkcs11  polkitd    xml-core
  libccid   libeac3        libnl-3-200  libpcsclite1      libpolkit-agent-1-0   opensc                 pcscd
sgml-base

Suggested packages:
  pcmciautils  resolvconf  openvpn-dco-dkms  openvpn-systemd-resolved  sgml-base-doc  debhelper

Summary:
  Upgrading: 0, Installing: 18, Removing: 0, Not Upgrading: 5
  Download size: 2,902 kB
  Space needed: 9,438 kB / 22.3 GB available

Continue? [Y/n]
```

Lancement du tunnel

```
admin@lab-debian-54:~$ sudo openvpn --config lab-pfsense-01-UDP4-1194-vpn-user-lab-config.ovpn
2026-01-23 19:42:00 OpenVPN 2.6.14 x86_64-pc-linux-gnu [SSL (OpenSSL)] [LZO] [LZ4] [EPOLL] [PKCS11]
[MH/PKTINFO] [AEAD] [DCO]
2026-01-23 19:42:00 library versions: OpenSSL 3.5.4 30 Sep 2025, LZO 2.10
2026-01-23 19:42:00 DCO version: N/A
Enter Auth Username: vpn-user-lab      <-- Le Username de notre Utilisateur crée sur PfSense !
Enter Auth Password: ••••••••              <-- Son Password (j'espère que vous l'avez noté)
2026-01-23 19:42:24 TCP/UDP: Preserving recently used remote address: [AF_INET]192.168.100.254:1194
2026-01-23 19:42:24 UDPv4 link local: (not bound)
2026-01-23 19:42:24 UDPv4 link remote: [AF_INET]192.168.100.254:1194
2026-01-23 19:42:24 WARNING: this configuration may cache passwords in memory -- use the auth-nocache option to
prevent this
2026-01-23 19:42:24 [vpn.lab.unsafe] Peer Connection Initiated with [AF_INET]192.168.100.254:1194
2026-01-23 19:42:25 TUN/TAP device tun0 opened
2026-01-23 19:42:25 net_iface_mtu_set: mtu 1500 for tun0
2026-01-23 19:42:25 net_iface_up: set tun0 up
2026-01-23 19:42:25 net_addr_v4_add: 10.0.8.2/24 dev tun0
2026-01-23 19:42:25 Initialization Sequence Completed
```

Une fois que vous avez le message "Completed", **ne fermez pas cette fenêtre**. Ouvrez un **deuxième terminal** sur la Debian et testez la vision à travers le tunnel !

```
admin@lab-debian-54:~$ ip a show tun0
3: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 500
    link/none
    inet 10.0.8.2/24 scope global tun0
       valid_lft forever preferred_lft forever
    inet6 fe80::eb65:e069:489c:246a/64 scope link stable-privacy proto kernel_ll
       valid_lft forever preferred_lft forever
```

Pinger l’adresse IP interne du VPN (à travers le tunnel)

```
admin@lab-debian-54:~$ ping 10.0.8.1 -c 3
PING 10.0.8.1 (10.0.8.1) 56(84) bytes of data.
64 bytes from 10.0.8.1: icmp_seq=1 ttl=64 time=0.505 ms
64 bytes from 10.0.8.1: icmp_seq=2 ttl=64 time=1.07 ms
64 bytes from 10.0.8.1: icmp_seq=3 ttl=64 time=1.18 ms

--- 10.0.8.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2004ms
rtt min/avg/max/mdev = 0.505/0.920/1.183/0.296 ms
```

et des que l’ont coupe la connexion au vps (avec un cltr+C) ...

```
^C2026-01-23 19:54:33 event_wait : Interrupted system call (fd=-1,code=4)
2026-01-23 19:54:33 SIGTERM received, sending exit notification to peer
2026-01-23 19:54:34 net_addr_v4_del: 10.0.8.2 dev tun0
2026-01-23 19:54:34 SIGTERM[soft,exit-with-notification] received, process exiting
admin@lab-debian-54:~$
```

```
admin@lab-debian-54:~$ ping 10.0.8.1 -c 3
PING 10.0.8.1 (10.0.8.1) 56(84) bytes of data.

--- 10.0.8.1 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2055ms
```

Si vous avez ce résultat, **votre VPN fonctionne parfaitement**.