

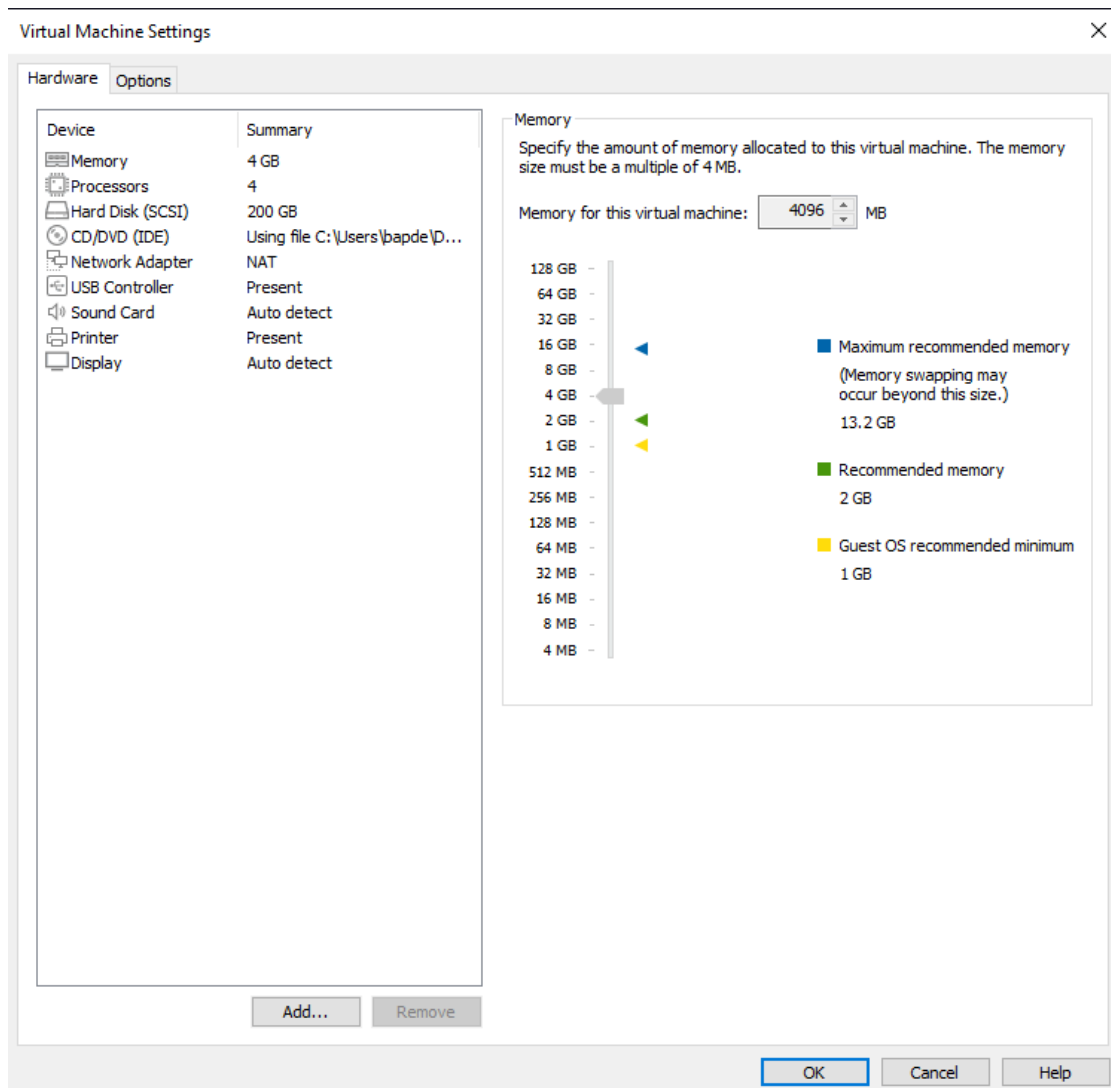
Rendu Final

22/01/22

Dans ce rendu final nous retracerons toutes les étapes qui nous ont permis d'aboutir au lancement d'un serveur FiveM, connecté à un autre serveur qui comporte la base de données du serveur principal ainsi que la création d'un Pfsense.

Etape 1 : Création d'une VM Debian 11

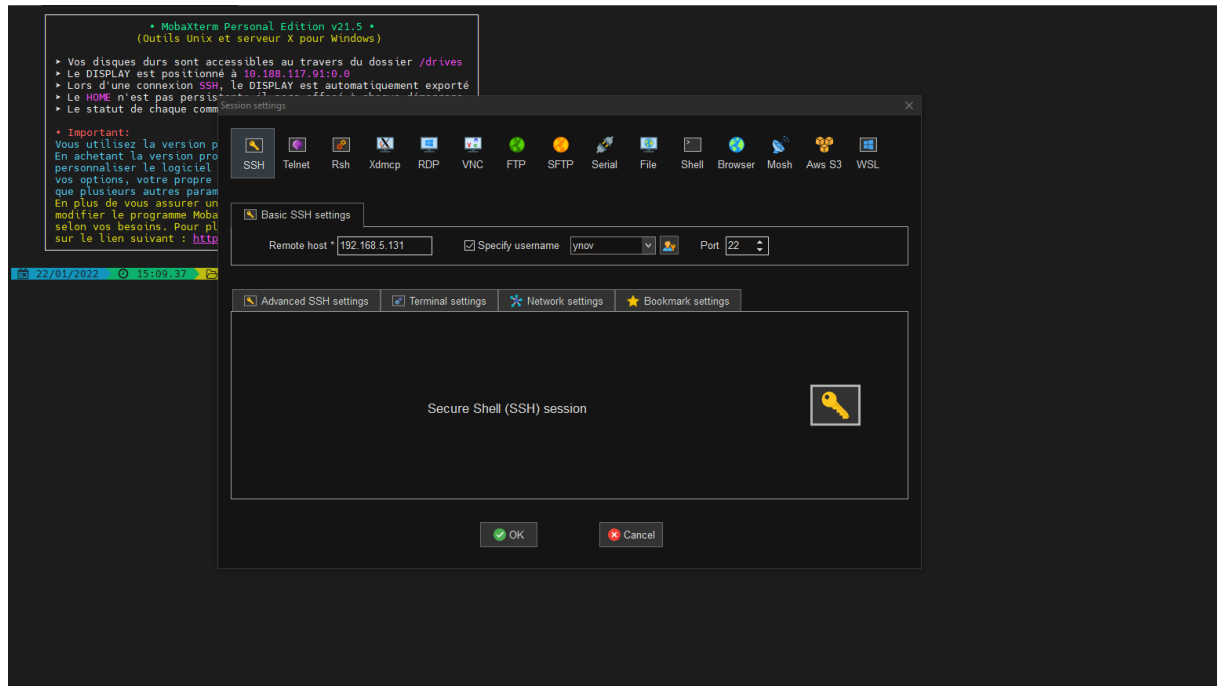
La première étape du projet a été de créer une machine virtuelle sur Vmware Workstation en GUI (Graphical User Interface) pour installer les fichiers de bases nécessaires au démarrage du serveur. Voici la configuration de notre VM :



Nous avons allouée tant d'espace disque pour pouvoir faire tourner le serveur correctement et avoir assez de place pour pouvoir développer dessus si nous venons à réellement utiliser le serveur. 4gb de ram suffisent largement pour faire tournée le serveur. Nous avons aussi mis un processeur de 4 cœurs qui fais aussi très bien tourner le serveur.

Etape 2 : Installation de la base sur notre VM

Pour la deuxième étape, nous avons installé la base via Moba Xterm, en prenant l'adresse IP de notre VM et l'utilisateur en question.



On a donc maintenant un dossier appelé « nubozombi » contenant un fichier ressource qui contient lui-même toutes les ressources nécessaires comme des scripts de Métier etc... On a aussi un dossier s'appelant Alpine et qui contient tous les artefacts FiveM pour Linux. Le dossier cache servira à relancer les ressources beaucoup plus rapidement. Il reste deux fichiers : run.sh et server.cfg mais nous y reviendrons dessus plus tard.

Etape 3 : Mise en place des artifacts Fivem

Maintenant que notre base se situe sur notre VM, le dossier Alpine étant vide, nous avons besoins de le remplir. Les artifacts FiveM permettent de garder la même version de gta5 et donc d'avoir le moins de bugs possible sur notre serveur. Nous allons donc nous rendre sur le site officiel de FiveM : cfx.re.

Dans la catégorie « build » pour linux nous retrouverons la dernière version des artifacts FiveM proposée. Il nous suffit de les glisser dans le dossier alpine situé à la racine de notre base.

Etape 4 : Création d'une deuxième VM et installation de MariaDB

Maintenant que notre base est à jour, il nous faut encore déposée le fichier SQL du serveur sur une base de données. On a donc créé une deuxième VM basée sur la même configuration que la première. Une fois que l'installation de la VM est terminée, nous installons MariaDB sur cette VM en utilisant la commande dans le terminal : `apt-get install mariadb-server`.

Etape 5 : Configuration de MariaDB

Maintenant que MariaDB est installé sur notre deuxième VM, nous pouvons vérifier que seulement trois bases de données sont présente en utilisant la commande :

show databases ;

Pour créer notre propre base de données on utilisera la commande : create databases essentialmode ;

On a maintenant dans notre interface MariaDB :

```
ynov@debian3: ~  
ynov@debian3:~$ su root  
Mot de passe :  
root@debian3:/home/ynov# mysql  
Welcome to the MariaDB monitor.  Commands end with ; or  
Your MariaDB connection id is 32  
Server version: 10.5.12-MariaDB-0+deb11u1 Debian 11  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab  
Type 'help;' or '\h' for help. Type '\c' to clear the current  
statement.  
  
MariaDB [(none)]> show databases;  
+-----+  
| Database |  
+-----+  
| essentialmode |  
| information_schema |  
| mysql |  
| performance_schema |  
+-----+  
4 rows in set (0.009 sec)  
  
MariaDB [(none)]> █
```

Par la suite on copie colle le fichier SQL fournis dans la base et nous précisons en haut du fichier SQL que nous allons utilisée la base essentialmode avec la commande : use database essentialmode ;

Maintenant on peut copier-coller notre script SQL dans notre terminal MariaDB et nous avons donc maintenant toutes les tables dans notre base essentialmode en utilisant deux commandes :

```
use database essentialmode ;
```

```
show tables ;
```

Notre base de données est maintenant configurée, il nous restera plus qu'à la lier au serveur dans le server.cfg.

Etape 6 : Maintenant nous sommes très proche d'avoir terminée la configuration du serveur. Le fichier server.cfg étant vide, il faut le remplir. Dans un premier temps, nous allons dire au server.cfg de démarrer toutes les ressources. On va prendre chaque nom de ressource et mettre dans le server.cfg :

```
ensure Le_nom_de_la_ressource
```

(En sachant que « ensure » peut aussi être remplacé par « start »)

On fait cela pour chaque ressource du serveur en revenant à la ligne.

On obtient à la fin :

```
ensure mapmanager
ensure chat
ensure spawnmanager
ensure sessionmanager
ensure fivem
ensure hardcap
ensure scoreboard
ensure playernames
ensure killfeed
ensure bob74_ipl

ensure essentialmode
ensure es_ui
ensure es_admin2
ensure mysql-async
ensure esplugin_mysql

ensure async
ensure es_extended
ensure esx_inventoryhud
ensure esx_menu_default
ensure esx_menu_list
ensure esx_menu_dialog
ensure esx_voice
ensure esx_addons_gcphone
ensure gcphone
ensure cron
ensure skinchanger
ensure esx_skin
ensure esx_addonaccount
ensure esx_addoninventory
ensure esx_datastore
ensure esx_accessories
ensure esx_clotheshop
ensure esx_weaponshop
ensure esx_weaponshop_sell
ensure esx_lscustom
```

Etape 7 : Liaison de la base de données au serveur

Pour lier la base de données au serveur, nous allons mettre plusieurs commandes dans le serveur.cfg.

Tout d'abord nous allons dire au serveur.cfg d'utiliser l'adresse IP de la base de données et de préciser le port. Dans notre cas nous mettrons :

192.168.5.131 : 3306

Le port 3306 est utilisé car c'est le port par défaut pour MySQL.

Maintenant que les ports sont définis il reste à mettre les clés de licence FiveM et Steam (celles-ci sont fournies par cfx.re et steam)

Il nous reste maintenant juste à préciser au serveur que nous allons utiliser une base de données sur l'adresse

IP 192.168.5.131, que la base de données se nomme essentialmode et que nous utilisons le user « root » qui a pour mot de passe « toor ».

Finalement on obtient en haut du server.cfg :

```
# Only change the IP if you're using a server with multiple network interfaces, otherwise change the port only.  
endpoint_add_tcp "192.168.5.130:3306"  
endpoint_add_udp "192.168.5.130:3306"  
sv_licenseKey 3zrzdqqr6f705wea7kpa6ect4h1pp2r1  
set steam_webApiKey "A5B20B5B9593320F9D848C7F315BA46D"  
# DB connection string  
set mysql_connection_string "server=192.168.1.100;database=essentialmode;userid=root;password=toor"  
set es_enableCustomData 1
```

Etape 8 : Réglages dans la config de MySQL

Le problème lors des connexions à une base de données qui se situe sur un autre serveur c'est que par défaut la bind adresse est mise sur : 127.0.0.1. Pour que la connexion se fasse, il faut modifier la bind adresse par 0.0.0.0 ou par l'adresse IP de notre VM.

Etape 9 : Création d'un script qui va lancer le serveur.

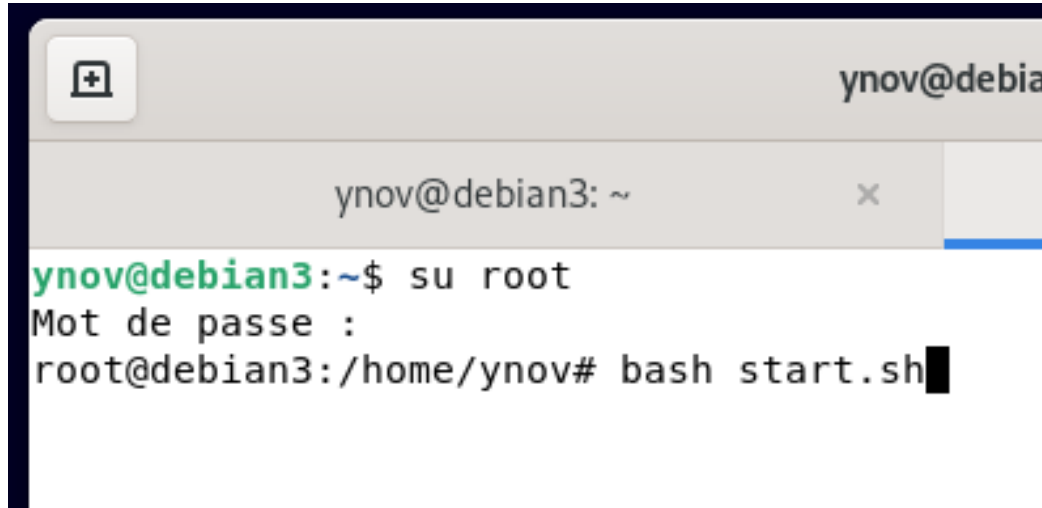
Comme nous l'avons appris lors des TP Linux, nous allons créer un script en bash pour lancer notre serveur :

```
1 #!/bash/bin  
2 cd nubozombi  
3 bash run.sh +exec server.cfg
```

Petit erreur * → # !/bin/bash

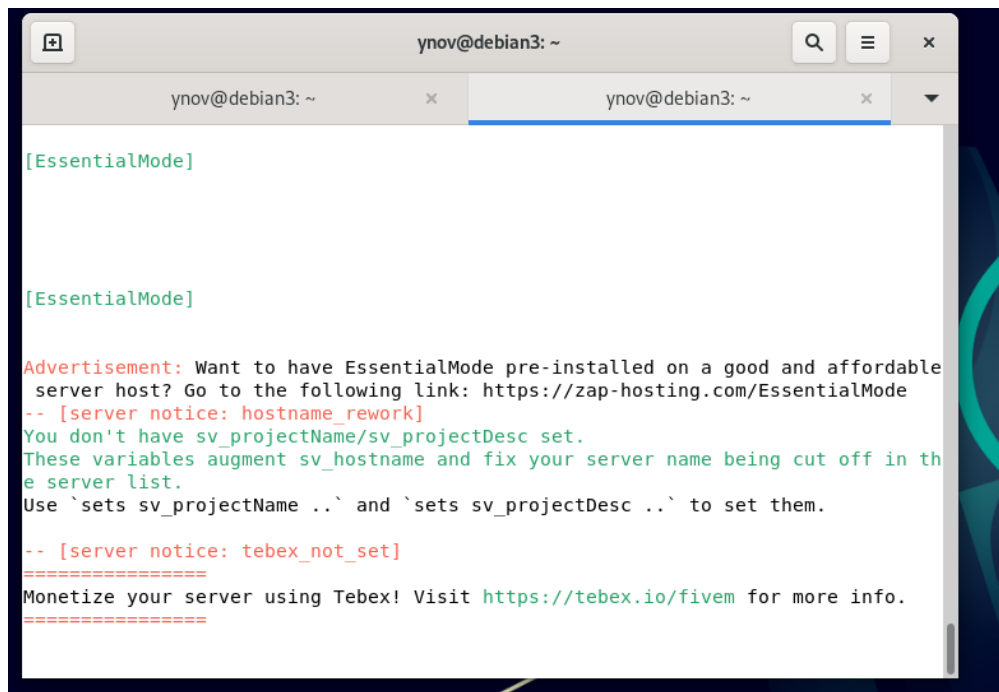
Etape 10 : Lancement du serveur

Maintenant que tout est configuré, il nous suffit de lancer notre script bash :



```
ynov@debian3: ~  
ynov@debian3:~$ su root  
Mot de passe :  
root@debian3:/home/ynov# bash start.sh
```

Et on obtient normalement :

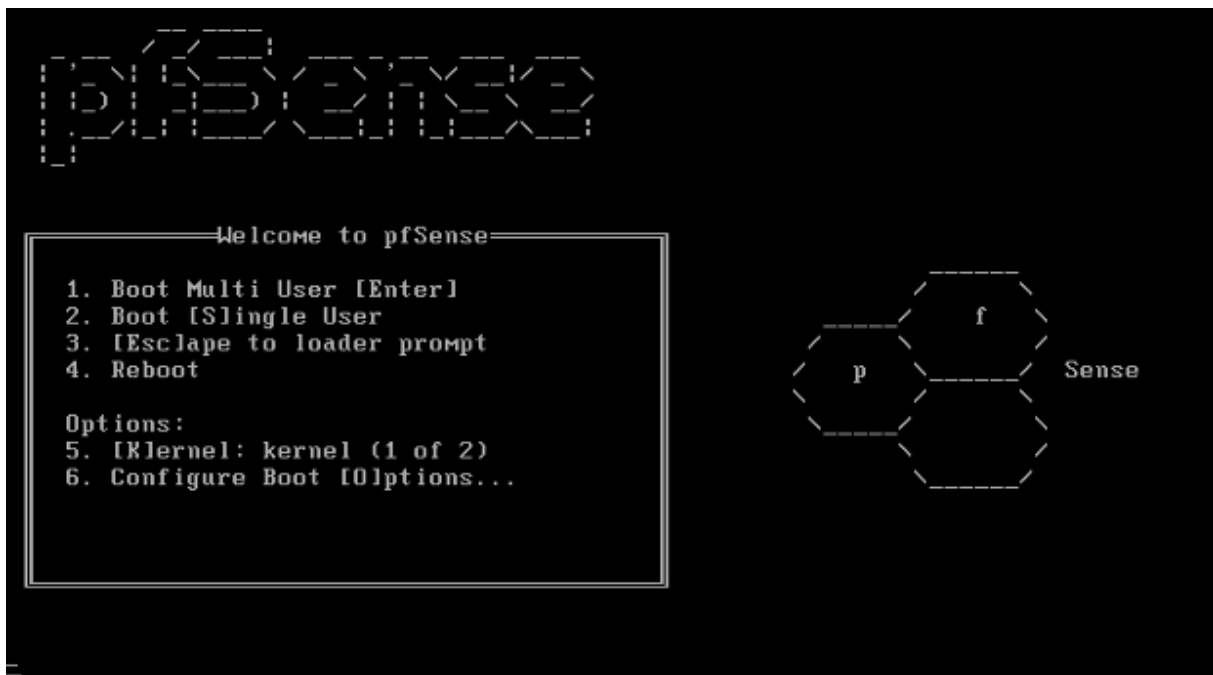


```
[EssentialMode]  
  
[EssentialMode]  
  
Advertisement: Want to have EssentialMode pre-installed on a good and affordable  
server host? Go to the following link: https://zap-hosting.com/EssentialMode  
-- [server notice: hostname_rework]  
You don't have sv_projectName/sv_projectDesc set.  
These variables augment sv_hostname and fix your server name being cut off in th  
e server list.  
Use `sets sv_projectName ..` and `sets sv_projectDesc ..` to set them.  
  
-- [server notice: tebex_not_set]  
=====
```

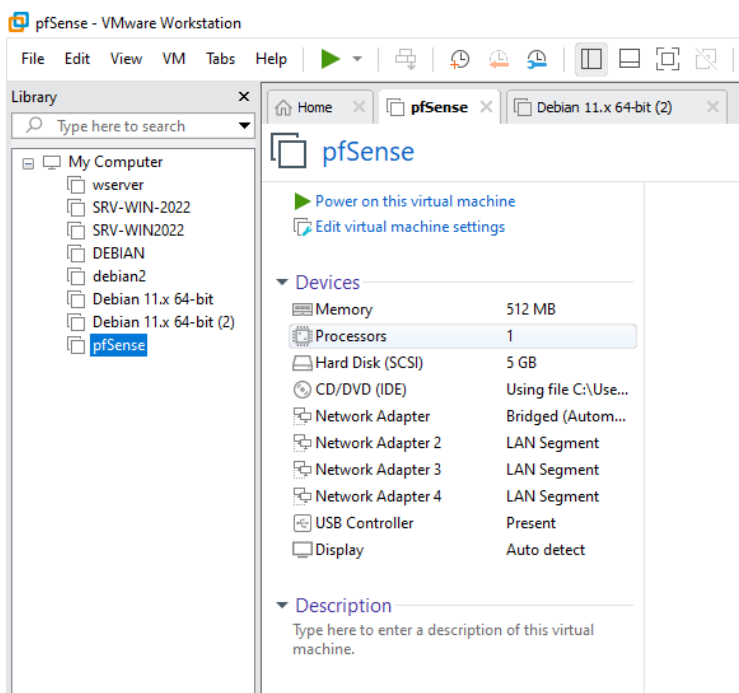
Si l'on obtient ça c'est que le serveur s'est bien lancé et que tout fonctionne.

Etape 11 : Installation d'une VM et de Pfsense

Il suffit de créer une VM et d'y mettre l'iso de Pfsense :



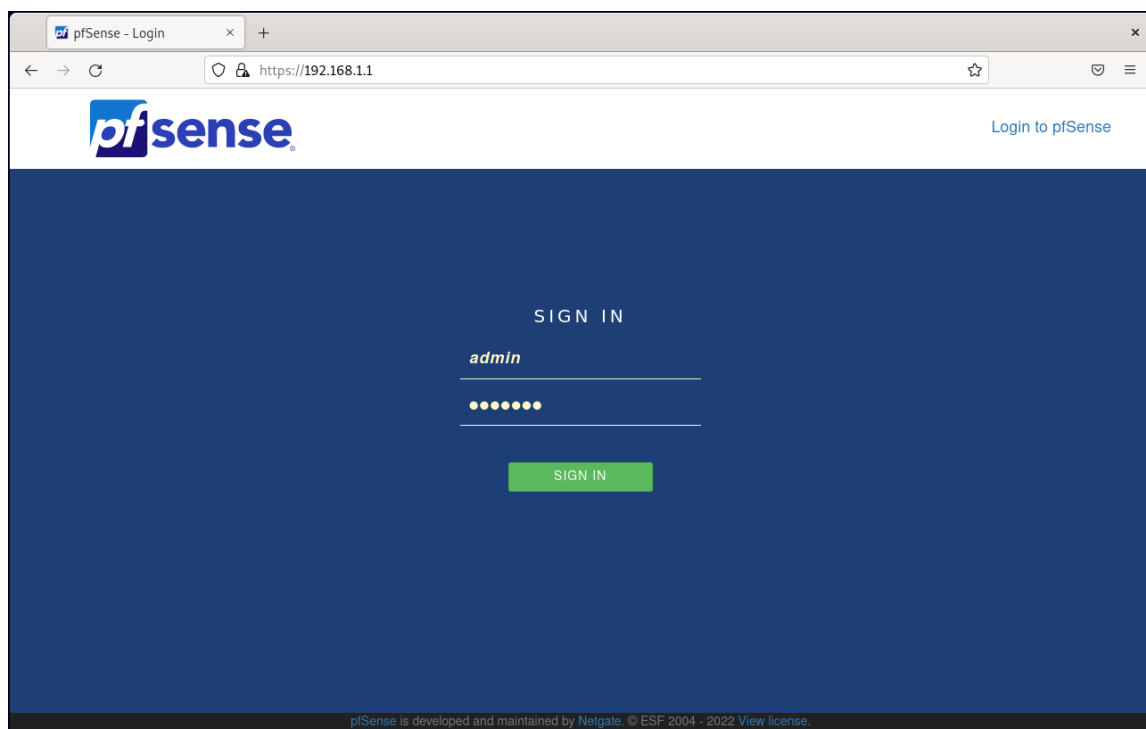
Une fois l'installation de Pfsense terminée, il faut maintenant créer 3 LAN segment dans l'interface VMware Workstation de la VM Pfsense :



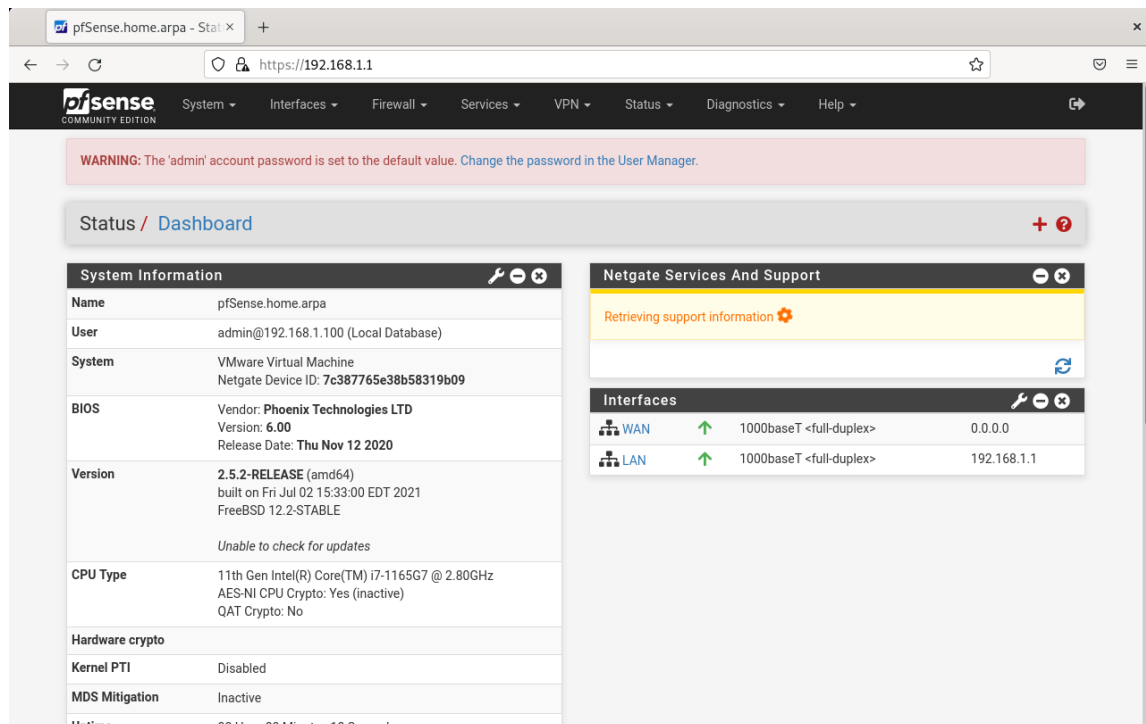
Maintenant que nos 3 LAN segment sont mis, il nous reste juste à les configurer dans Pfsense et à mettre un des LAN segment sur notre VM où l'ont veux se connecter au panel Pfsense.

On change donc la connexion de notre VM pour passer de NAT à LAN Segment.

Maintenant en tapant l'adresse IP de Pfsense sur le moteur de recherche de notre VM, on obtient le panel :



Avec l'identifiant admin et le mot de passe Pfsense on rentre maintenant sur le panel :



Par le fait que nous étions en distanciel et que nous n'avons pas pu toucher à notre box, et donc pas pu finir de configurer notre Pfsense.

Etape 12 : Les problèmes rencontrés

Dans ce projet nous avons rencontré seulement un seul problème majeur : L'utilisateur root n'avais pas les droits pour utiliser la base de données essentialmode.

On lui a donc donné les droits avec la commande :

- `GRANT ALL PRIVILEGES ON essentialmode* TO 'root@ip_de_la_vm' ;`

Avec cette commande l'utilisateur root a tous les droits sur essentialmode.

Merci d'avoir pris le temps de lire notre compte rendu.

Groupe 6 :

DEFRANCE Baptiste

MENOU Lucas

ROSAK Kilian

CAVANIER Jules