Maxime LE BRAS
SIO2
Bloc 2 - SISR

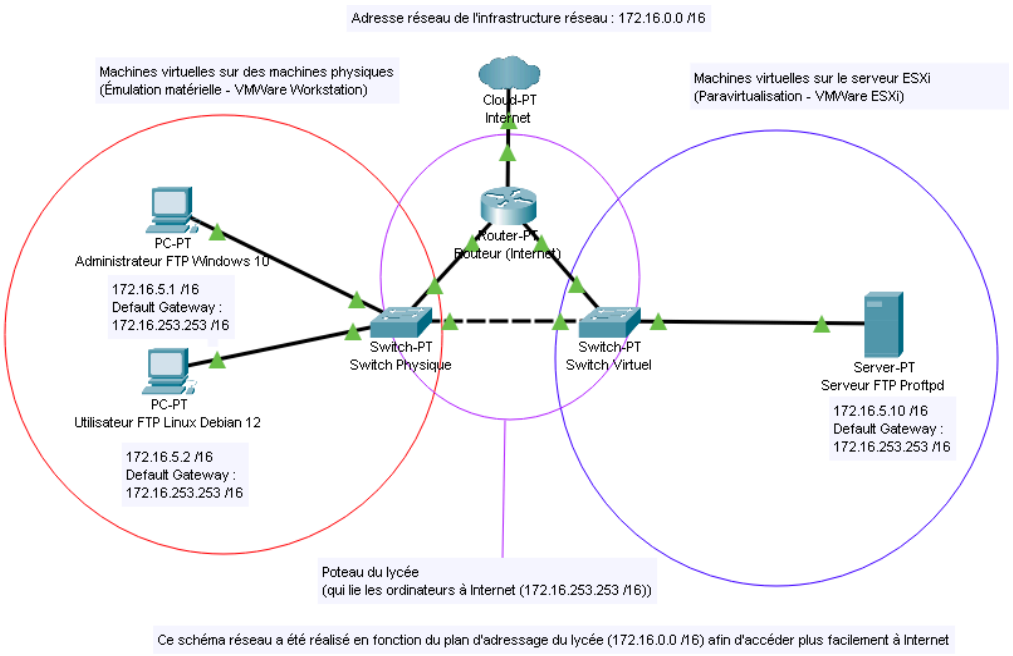
Le 16/10/2024

TP4 - Société NARFAS

Sommaire

Lot 1 - Serveur FTP avec Accès multiples

Schéma réseau :



Ce schéma réseau a été réalisé en fonction du plan d'adressage du lycée (172.16.0.0 /16) afin d'accéder plus facilement à Internet.

Mise en place de l'infrastructure réseau :

Côté émulation matérielle :

```
Carte Ethernet Ethernet0 :

   Suffixe DNS propre à la connexion. . . :
   Description. . . . . . . . . . . . . . : Intel(R) 82574L Gigabit Network Connection
   Adresse physique . . . . . . . . . . . : 00-0C-29-D4-EE-CD
   DHCP activé. . . . . . . . . . . . . . : Non
   Configuration automatique activée. . . : Oui
   Adresse IPv6 de liaison locale. . . . .: fe80::2823:2c67:406b:6fa1%11(préféré)
   Adresse IPv4. . . . . . . . . . . . . .: 172.16.5.1(préféré)
   Masque de sous-réseau. . . . . . . . . : 255.255.0.0
   Passerelle par défaut. . . . . . . . . : 172.16.253.253
   IAID DHCPv6 . . . . . . . . . . . : 100666409
   DUID de client DHCPv6. . . . . . . . : 00-01-00-01-2E-A1-2A-22-00-0C-29-D4-EE-CD
   Serveurs DNS. . .  . . . . . . . . . : 80.10.246.2
                                 80.10.246.129
   NetBIOS sur Tcpip. . . . . . . . . . . : Activé

C:\Users\admin>
```

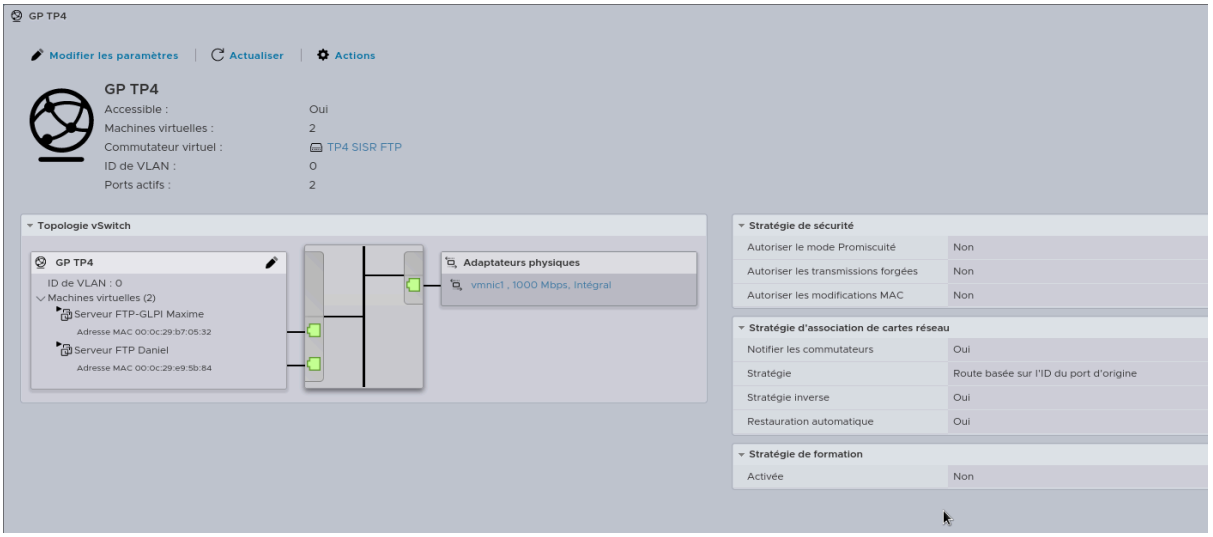
Voici la configuration IP de l'administrateur FTP Windows 10.

```
source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

auto ens33
iface ens33 inet static

address 172.16.5.2
netmask 255.255.0.0
gateway 172.16.253.253
```

Voici la configuration IP de l'utilisateur FTP Debian 12. Il ne faut pas oublier de vérifier si les serveurs DNS du lycée sont mis en place sur la machine en allant sur le fichier /etc/resolv.conf.

On a branché les deux machines physiques sur le switch Cisco de niveau 2 qu'on a relié au poteau du lycée qui permet d'avoir accès à Internet.

<u>Côté paravirtualisation :</u>

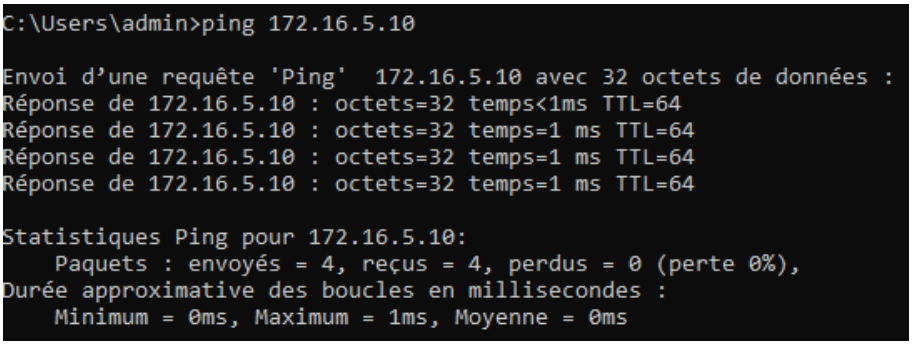


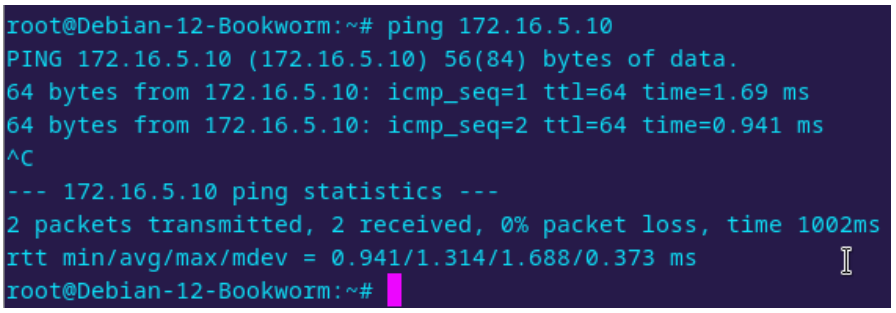
J'ai mis en place un commutateur virtuel sur le serveur ESXi qui va communiquer avec Internet et avec les machines virtuelles VMWare Workstation Pro "Administrateur FTP Windows 10" et "Utilisateur FTP Linux Debian 12" grâce à la mise en place de l'adaptateur physique vmnic1 (<u>Passerelle pour aller vers Internet :</u> 172.16.253.253).
Le plan d'adressage (je l'ai réalisé en fonction de l'adressage IP du lycée pour me faciliter au niveau des tâches à faire) mis en place va permettre à toutes les machines de communiquer ensemble et d'accéder à Internet.

```
auto lo
iface lo inet loopback

iface ens192 inet static
address 172.16.5.10
netmask 255.255.0.0
gateway 172.16.253.253
```

Voici la configuration IP du serveur FTP qui est connecté à Internet via la passerelle par défaut 172.16.253.253 et au switch virtuel qui sera connecté à l'interface vmnic1 qui est redirigée vers Internet.

<u>Tests de communication entre les différentes machines de l'infrastructure réseau :</u>





Les machines "Administrateur FTP Windows 10" et "Utilisateur FTP Linux Debian 12" communiquent bien avec le serveur FTP, donc cela veut dire qu'ils pourront s'authentifier au serveur.

```
root@Debian-12-Bookworm:~# ping www.google.fr
PING www.google.fr (142.250.75.227) 56(84) bytes of data.
64 bytes from par10s41-in-f3.1e100.net (142.250.75.227): icmp_seq=1 ttl=114 time=4.61 ms
64 bytes from par10s41-in-f3.1e100.net (142.250.75.227): icmp_seq=2 ttl=114 time=4.91 ms
^C
--- www.google.fr ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 4.608/4.758/4.908/0.150 ms
```

```
C:\Users\admin>ping www.google.fr

Envoi d'une requête 'ping' sur www.google.fr [142.250.201.163] avec 32 octets de données :
Réponse de 142.250.201.163 : octets=32 temps=7 ms TTL=113

Statistiques Ping pour 142.250.201.163:
    Paquets : envoyés = 1, reçus = 1, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 7ms, Maximum = 7ms, Moyenne = 7ms
Ctrl+C
^C
```

Les machines “Administrateur FTP Windows 10” et “Utilisateur FTP Linux Debian 12” communiquent bien avec Google, donc la liaison entre ces machines et Internet est fonctionnelle.

Pour conclure, mon infrastructure réseau est fonctionnelle !

Mise en place du serveur FTP Proftpd avec la configuration d’utilisateurs virtuels :

```
root@ServeurFTP:~# mkdir /var/www/backupNARFAS
```

```
root@ServeurFTP:/var/www/backupNARFAS# touch factures.txt
root@ServeurFTP:/var/www/backupNARFAS# ls -l
total 0
-rw-r--r-- 1 root root 0 17 nov.  21:46 factures.txt
root@ServeurFTP:/var/www/backupNARFAS#
```

Ici, j’ai créé le répertoire /var/www/backupNARFAS et le fichier “factures.txt” pour réaliser les tests d’accès au serveur FTP selon les utilisateurs (administrateur, utilisateur).

```
root@ServeurFTP:~# apt-get install proftpd
```

Avant d’installer le package proftpd, nous avons fait les mises à jour des paquets de la machine (apt-get update). Ce package va permettre de mettre en place le serveur FTP Proftpd.

```
DefaultRoot ~

# Users require a valid shell listed in /etc/shells to login.
# Use this directive to release that constrain.
# RequireValidShell off

# Port 21 is the standard FTP port.
Port 21

# In some cases you have to specify passive ports range to by-pass
# firewall limitations. Ephemeral ports can be used for that, but
# feel free to use a more narrow range.
PassivePorts 49152 65534

AuthUserFile /etc/ftpd.passwd
```

Pour modifier le fichier de configuration du serveur, il faut taper “nano /etc/proftpd/proftpd.conf”. Ces paramètres sont essentiels pour mettre en place des utilisateurs virtuels sur le serveur FTP et configurer celui-ci, car le “DefaultRoot” permet de restreindre l’utilisateur à rester sur le répertoire qu’on lui a attribué, le port par défaut d’un serveur FTP correspond au nombre 21, puis, les ports passifs permettent de contourner les restrictions du pare-feu dans le cas où notre infrastructure réseau contiendrait un pare-feu et le fichier “/etc/ftpd.passwd” correspond aux informations d’authentification des utilisateurs virtuels (nom d’utilisateur et mot de passe). Lors de la connexion au FTP avec un utilisateur virtuel, le serveur fera la correspondance avec ce fichier.

```
 GNU nano 7.2                                /etc/shells *
# /etc/shells: valid login shells
/bin/sh
/bin/bash
/usr/bin/bash
/bin/rbash
/usr/bin/rbash
/bin/dash
/usr/bin/dash
/usr/bin/sh
/bin/false
```

Ici, j’ai ajouté “/bin/false” dans le fichier “/etc/shells” afin que les utilisateurs virtuels puissent avoir un shell actif pour exécuter des commandes sur le serveur FTP.

```
root@ServeurFTP:~# ftpasswd --passwd --name=admin --uid=33 --gid=33 --home=/var/
www --shell=/bin/false --file=/etc/ftpd.passwd
ftpasswd: using alternate file: /etc/ftpd.passwd
ftpasswd: creating passwd entry for user admin

Password:
Re-type password:

ftpasswd: entry created
root@ServeurFTP:~# ftpasswd --passwd --name=user --uid=33 --gid=33 --home=/var/w
ww/backupNARFAS --shell=/bin/false --file=/etc/ftpd.passwd
ftpasswd: using alternate file: /etc/ftpd.passwd
ftpasswd: creating passwd entry for user user

Password:
Re-type password:

ftpasswd: entry created
root@ServeurFTP:~#
```

Ici, j'ai créé les différents utilisateurs virtuels qui sont "admin" et "user". L'utilisateur "admin" aura accès à tous les répertoires enfants de son répertoire qui est "/var/www", donc il aura accès à "backupNARFAS" alors que "user" a accès qu'au répertoire "/var/www/backupNARFAS" et il ne pourra pas se diriger vers d'autres répertoires.

```
root@ServeurFTP:~# chown 33:33 -R /var/www/
root@ServeurFTP:~# chown 33:33 -R /var/www/backupNARFAS/
root@ServeurFTP:~# ls -ld /var/www/
drwxr-xr-x 3 www-data www-data 4096 17 nov.  21:26 /var/www/
root@ServeurFTP:~# ls -ld /var/www/backupNARFAS/
drwxr-xr-x 2 www-data www-data 4096 17 nov.  21:26 /var/www/backupNARFAS/
root@ServeurFTP:~#
```

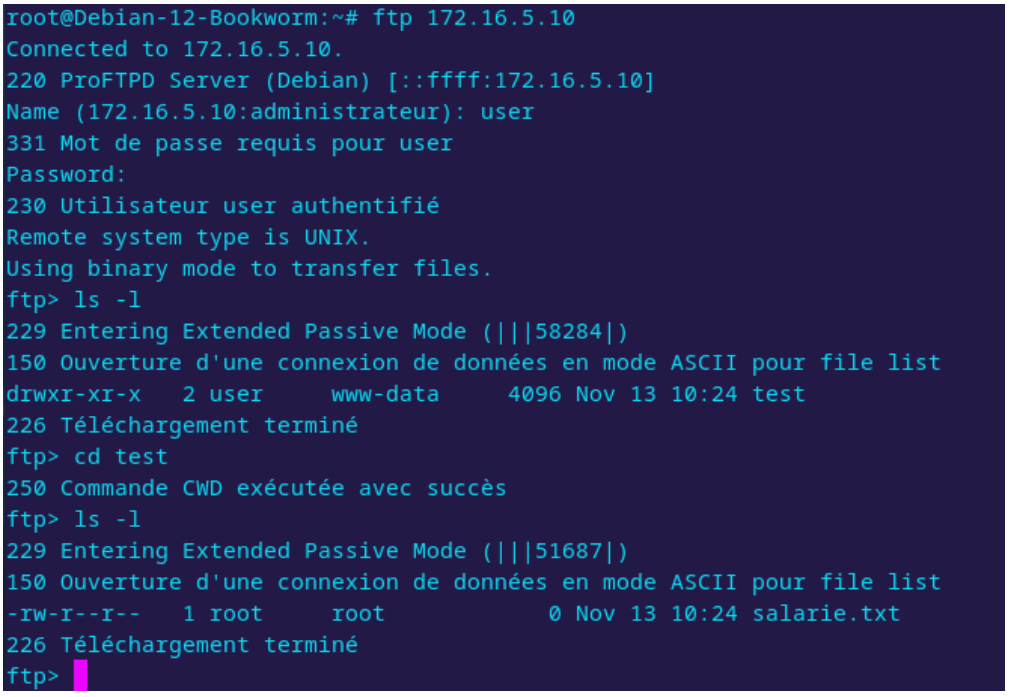
Ces commandes sur les répertoires "/var/www/backupNARFAS" et "/var/www" vont permettre d'introduire ces répertoires dans le serveur FTP.

```
root@ServeurFTP:~# cat /etc/ftpd.passwd
admin:$5$rkGcCaKg8nNPTU1c$q0K1ZiYFNYMzA6Z5M7AzHfk96Y59vjKuig4rBmBVe82:33:33::/va
r/www:/bin/false
user:$5$RZyz16bbkzo6as3E$kE9SrZImJ7Q6k5O95oTYBqKiezKWRh2uyVhF47W1tz6:33:33::/var
/www/backupNARFAS:/bin/false
root@ServeurFTP:~#
```

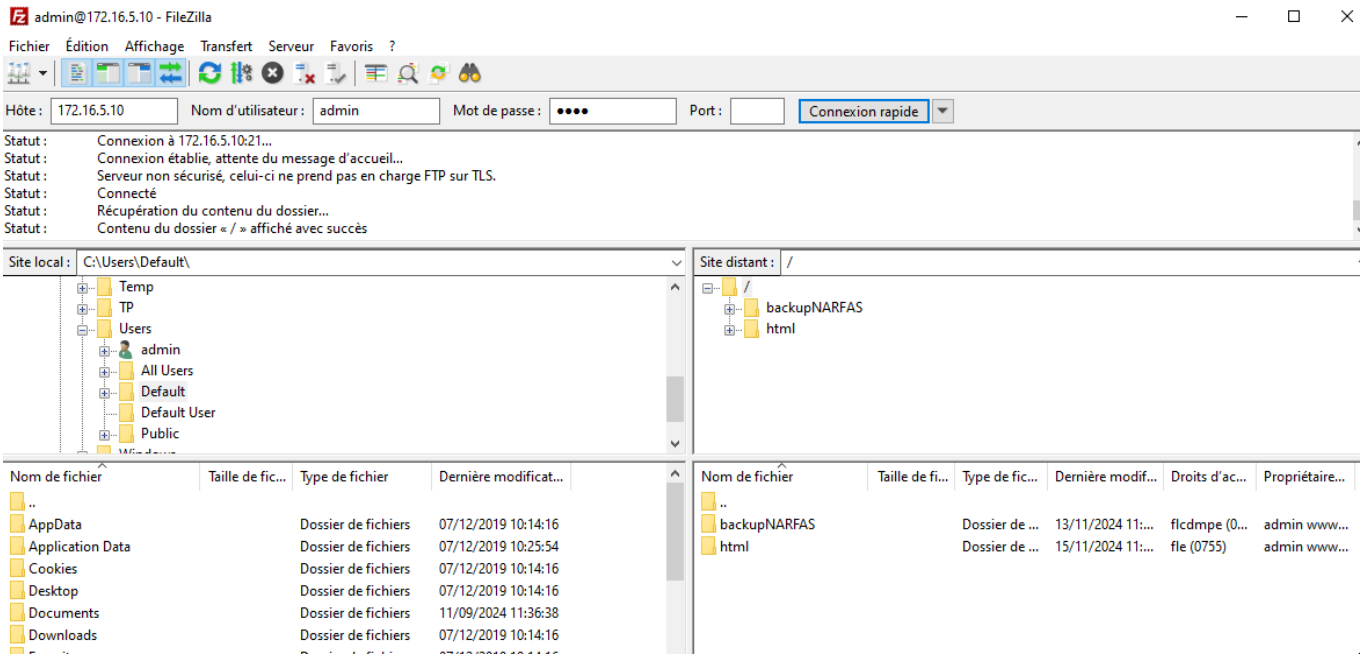
Cette commande permet d'afficher le contenu du fichier "/etc/ftpd.passwd". Ce fichier contient les informations au niveau des utilisateurs virtuels en termes d'authentification.

J'ai redémarré le service "proftpd" ce qui revient à relancer le serveur FTP afin de mettre à jour les modifications effectuées en termes de configuration.

Tests d'accès au serveur FTP via les utilisateurs virtuels configurés :



J'ai installé le package "ftp" pour me connecter au serveur FTP avec son adresse IP sur l'utilisateur Linux. Ici, on a réussi à s'authentifier avec le compte "user" ce qui veut dire que la configuration de cet utilisateur virtuel a été faite correctement. Ici, on est bien dans le répertoire "/var/www/backupNARFAS" avec le répertoire enfant "test" qui contient le fichier "salarie.txt".



J'ai installé le logiciel Filezilla Client pour pouvoir me connecter au serveur FTP avec son adresse IP sur l'administrateur Windows. Ici, on a réussi à s'authentifier avec le compte "admin" ce qui veut dire que la configuration de cet utilisateur virtuel a été faite correctement. On voit qu'il peut avoir accès à tous les répertoires enfants du répertoire "/var/www".

Difficultés rencontrées :

Au départ, j'ai utilisé le service vsftpd pour mettre un serveur FTP fonctionnel sauf que la création des utilisateurs virtuels n'était pas fonctionnelle alors que les procédures de création étaient correctes. Je suppose que ce service ne doit pas être compatible avec la version du système d'exploitation de la machine serveur, donc j'ai pris le service proftpd pour réaliser les tâches demandées et j'ai réussi à les faire.

Lot 2 - Un gestionnaire d'inventaire (GLPI)

Les prérequis pour une machine qui héberge un serveur GLPI, sont :

- Debian 12 - 64 bits
- 20 Go de disque dur (Stockage)
- 4 Go de RAM (tout dépend du nombre d'utilisateurs et d'ordinateurs inventoriés dans le serveur)

Installation de GLPI en ligne de commande :

```
root@serveurFTPMaxime:~# apt-get install apache2 php libapache2-mod-php
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets suivants ont été installés automatiquement et ne sont plus nécessaires :
  libdbus-glib-1-2 libwpe-1.0-1 libwpebackend-fdo-1.0-1
Veuillez utiliser « apt autoremove » pour les supprimer.

root@serveurFTPMaxime:~# apt-get install mariadb-server
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets suivants ont été installés automatiquement et ne sont plus nécessaires :
  libdbus-glib-1-2 libwpe-1.0-1 libwpebackend-fdo-1.0-1

root@serveurFTPMaxime:~# apt-get install php-imap php-ldap php-curl php-xmlrpc php-gd p
hp-mysql php-cas
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets suivants ont été installés automatiquement et ne sont plus nécessaires :
  libdbus-glib-1-2 libwpe-1.0-1 libwpebackend-fdo-1.0-1
Veuillez utiliser « apt autoremove » pour les supprimer.
Les paquets supplémentaires suivants seront installés :
```

```
root@serveurFTPMaxime:/usr/src# apt-get install php-intl
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
```

Tout d'abord, il faut installer les packages "apache2", "mariadb-server", "php" et "php-intl", ils sont primordiaux pour l'installation et la configuration de GLPI.

```
root@serveurFTPMaxime:~# mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
      SERVERS IN PRODUCTION USE!  PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
haven't set the root password yet, you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

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

You already have your root account protected, so you can safely answer 'n'.

Switch to unix_socket authentication [Y/n] Y
Enabled successfully!
Reloading privilege tables..
```

Ici, il faut répondre à toutes les questions en disant “Y” pour avoir une configuration optimale et sécurisée du service “mariadb-server”.

```
root@serveurFTPMaxime:~# apt-get install apcupsd php-apcu
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets suivants ont été installés automatiquement et ne sont plus nécessaires :
  libdbus-glib-1-2 libwpe-1.0-1 libwpebackend-fdo-1.0-1
```

Il faut installer ces modules complémentaires afin que GLPI fonctionne correctement.

```
root@serveurFTPMaxime:~# /etc/init.d/apache2 restart
Restarting apache2 (via systemctl): apache2.service.
root@serveurFTPMaxime:~# /etc/init.d/mariadb restart
Restarting mariadb (via systemctl): mariadb.service.
```

Il faut redémarrer les services “apache2” et “mariadb” afin de prendre en compte les modifications réalisées précédemment.

```
root@serveurFTPMaxime:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 31
Server version: 10.11.6-MariaDB-0+deb12u1 Debian 12

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)]>
```

cd

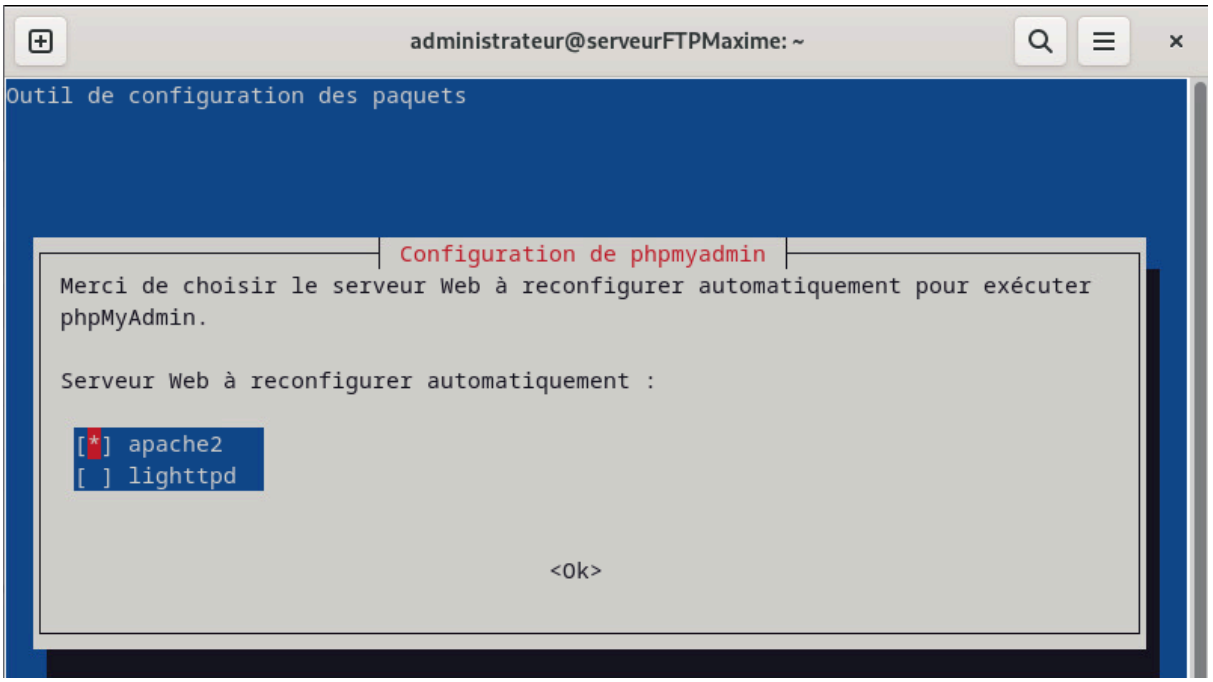
Ensuite, il faut se connecter en “root” au service “mariadb-server” afin de créer la base de données qui servira pour le serveur GLPI.

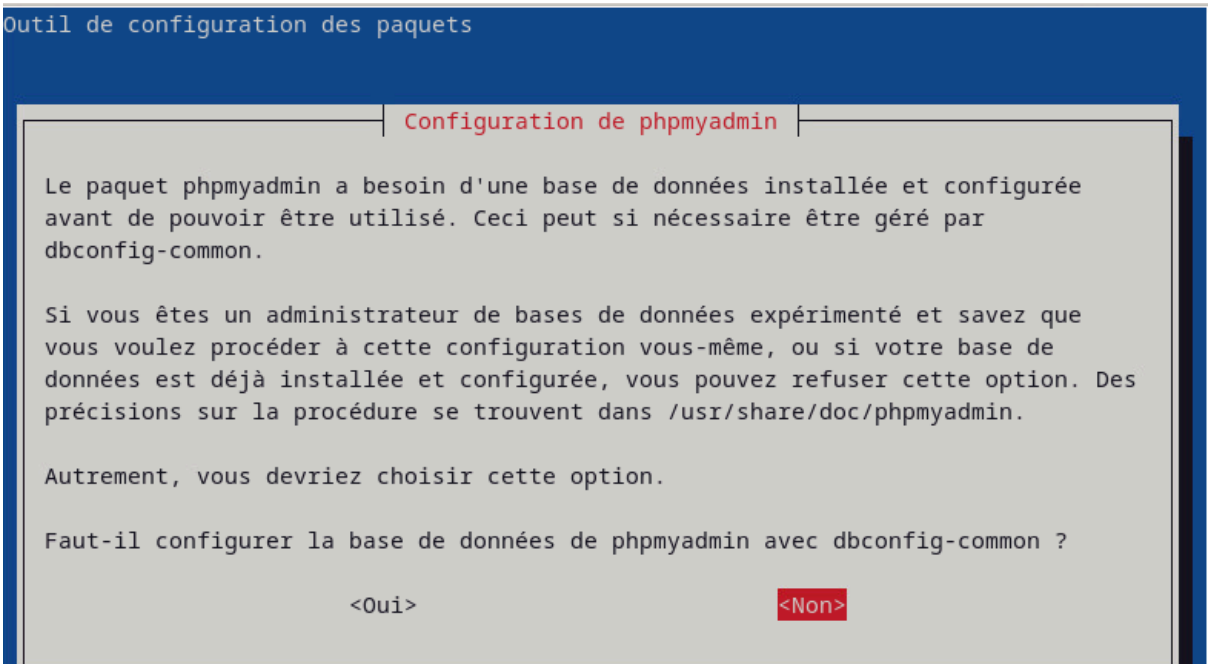
```
MariaDB [(none)]> CREATE DATABASE glpidb;
Query OK, 1 row affected (0,001 sec)

MariaDB [(none)]> GRANT ALL PRIVILEGES ON glpidb.* TO glpiuser@localhost IDENTIFIED BY
"1234";
Query OK, 0 rows affected (0,002 sec)

MariaDB [(none)]> quit
Bye
root@serveurFTPMaxime:~#
```

Ici, j’ai créé la base de données “glpidb”, j’ai attribué des privilèges sur la totalité de “glpidb” à “glpiuser” qui est un utilisateur local à GLPI afin de faire la connexion entre la base de données et le serveur GLPI.

```
root@serveurFTPMaxime:~# apt-get install phpmyadmin
```

L’installation de ce package va permettre de gérer la base de données via une interface graphique (dans notre contexte, c’est inutile mais cela peut être utile en termes de facilité de gestion de la base de données).





Ici, on a configuré “phpmyadmin” afin qu’il soit lié avec le service web “apache2”. Comme la base de données est configurée et installée, on a pas besoin de configurer la base de données de phpmyadmin avec dbconfig-common.

```
root@serveurFTPMaxime:~# cd /usr/src
```

Je me suis redirigé vers ce répertoire pour installer GLPI en ligne de commande, car il contient le code source des programmes et logiciels installés sur la machine serveur.

```
root@serveurFTPMaxime:/usr/src# wget https://github.com/glpi-project/glpi/releas
es/download/10.0.6/glpi-10.0.6.tgz
```

Cette commande permet d’aller chercher le fichier compressé (en .tgz) de la version 10.0.6 de GLPI sur le site de Github.

```
tar -xvzf glpi-10.0.6.tgz -C /var/www/html/
```

Cette commande va permettre de décompresser le fichier d’installation de GLPI 10.0.6 dans le répertoire “/var/www/html” afin que GLPI soit installé dans les répertoires de configuration du service “apache2”.

```
/var/www/html/glpi
```

À la suite de cette commande exécutée précédemment, on aura les fichiers de configuration de GLPI dans un répertoire appelé “/glpi” qui sera créé dans les répertoires de configuration du service “apache2”.

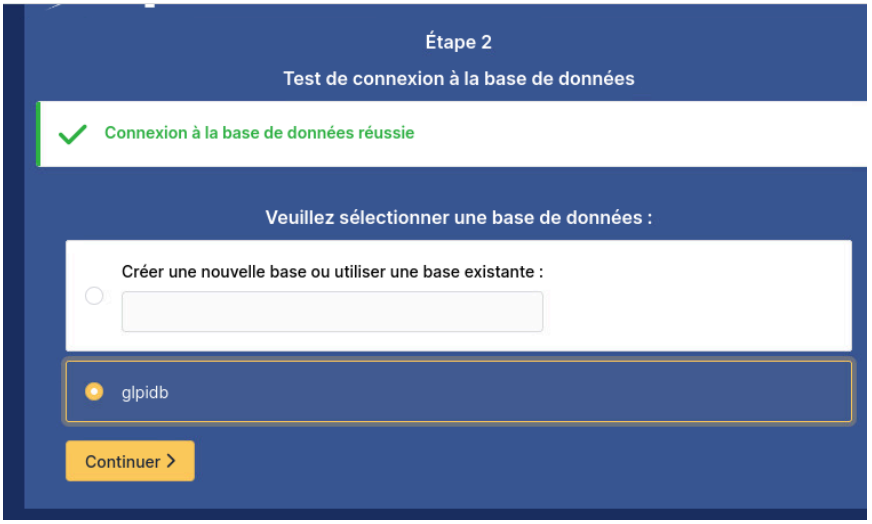
```
root@serveurFTPMaxime:/usr/src# chown -R www-data /var/www/html/glpi/
```

Maintenant, on va attribuer les droits du répertoire “/var/www/html/glpi” au serveur web “apache2” pour qu’il agisse sur les fichiers en interface graphique.

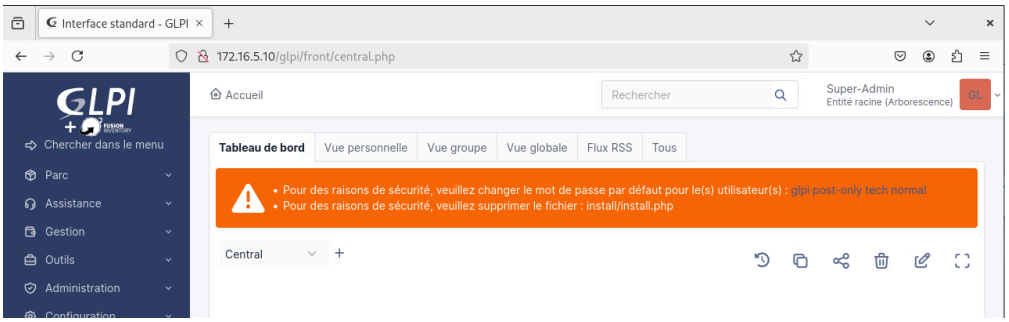
Installation de GLPI en interface graphique :

Ici, on va procéder à l’installation de GLPI en interface graphique, donc il faut appuyer sur le bouton “Installer”.

Ici, on va se connecter en local à la base de données pour lier la base de données et le serveur web en lien avec GLPI via l'authentification mise en place en ligne de commande sur "mariadb-server" (utilisateur : glpiuser, mot de passe : 1234).



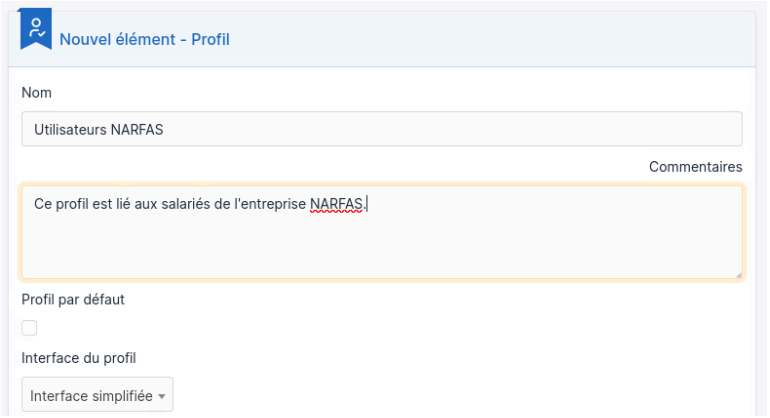
La connexion à la base de données "glpidb" est réussie et on sélectionne "glpidb".



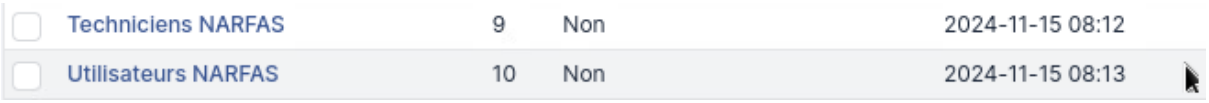
J'ai réussi à se connecter à l'interface graphique de GLPI via le compte administrateur de GLPI (utilisateur : glpi, mot de passe : glpi).

Création des profils et des utilisateurs:

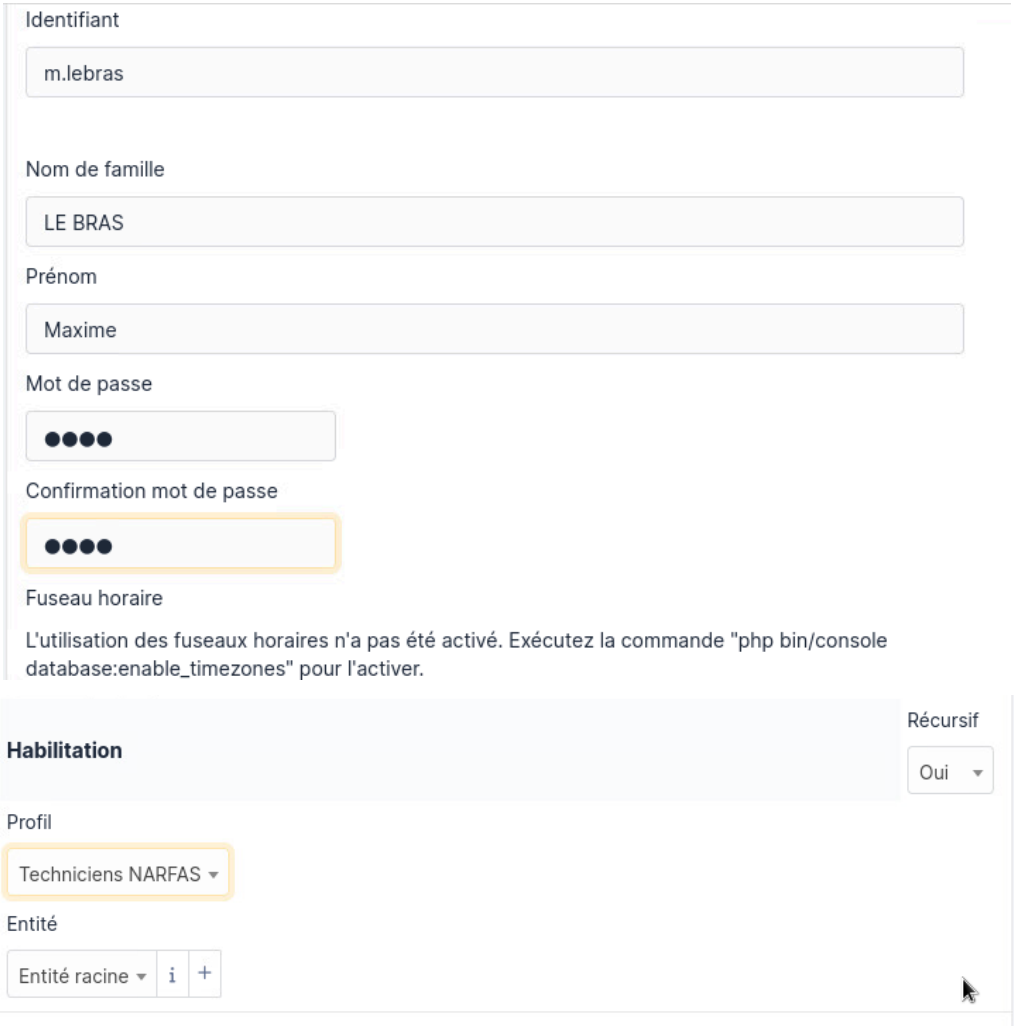
Ici, j'ai créé le profil "Techniciens NARFAS" (en configurant les droits) afin d'intégrer les utilisateurs correspondants à cette fonction de technicien avec une interface standard puisqu'ils connaissent bien GLPI.



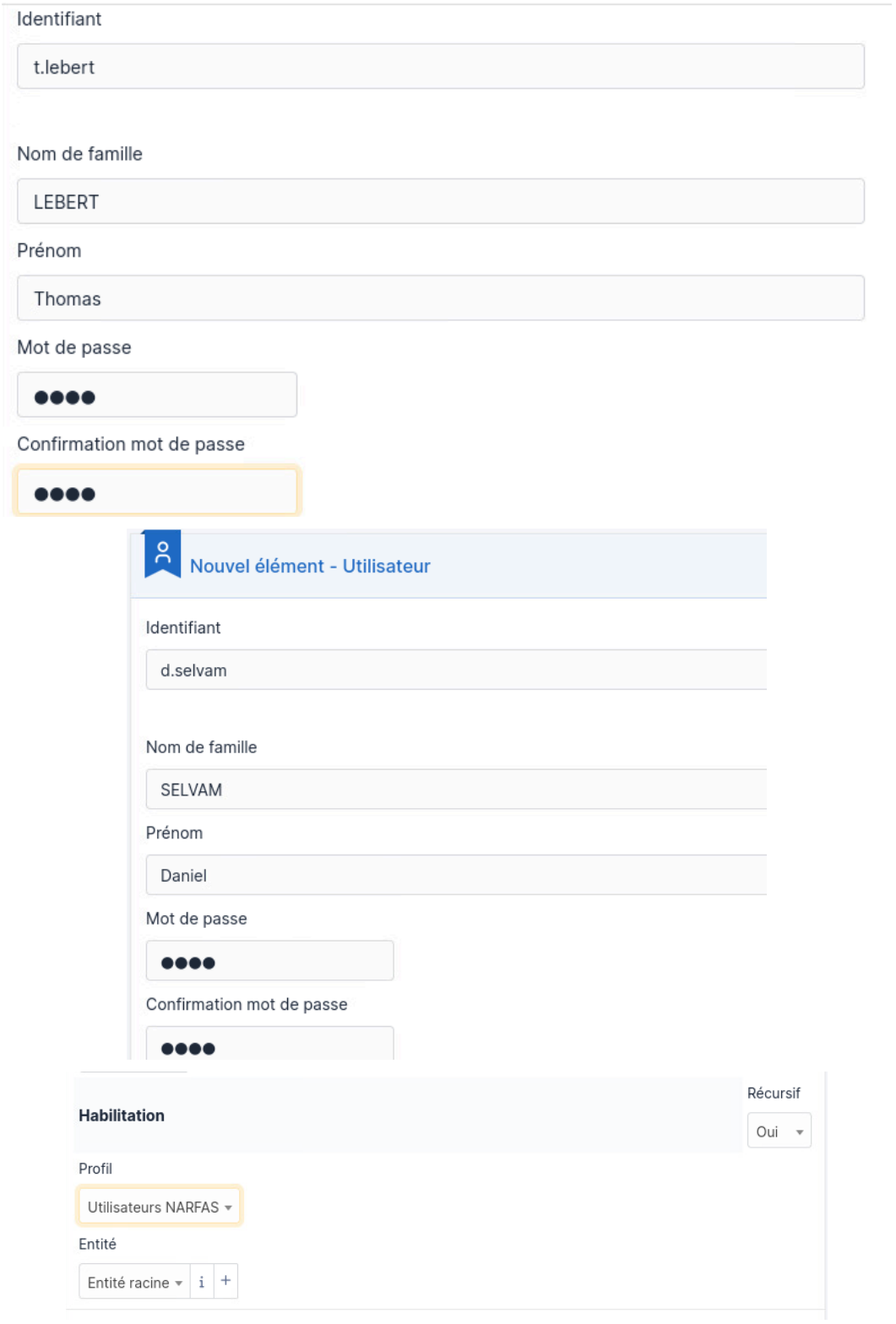
Ici, j'ai créé le profil "Utilisateurs NARFAS" (en configurant les droits pour qu'ils gèrent les tickets des utilisateurs de l'entreprise NARFAS (clôture du ticket, attribution des tickets aux techniciens, etc…)) afin d'intégrer les salariés de la société NARFAS dans ce profil avec une interface simplifiée afin qu'ils ne soient pas perdus lorsqu'ils iront sur cette interface.



Les profils ont bien été créés.

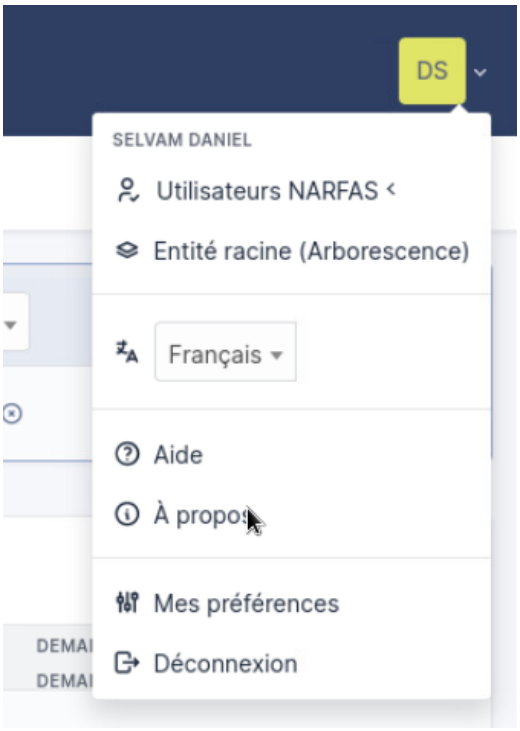


Ici, j’ai créé un utilisateur qui a les droits de technicien informatique sur GLPI en intégrant celui-ci dans le profil “Techniciens NARFAS”.

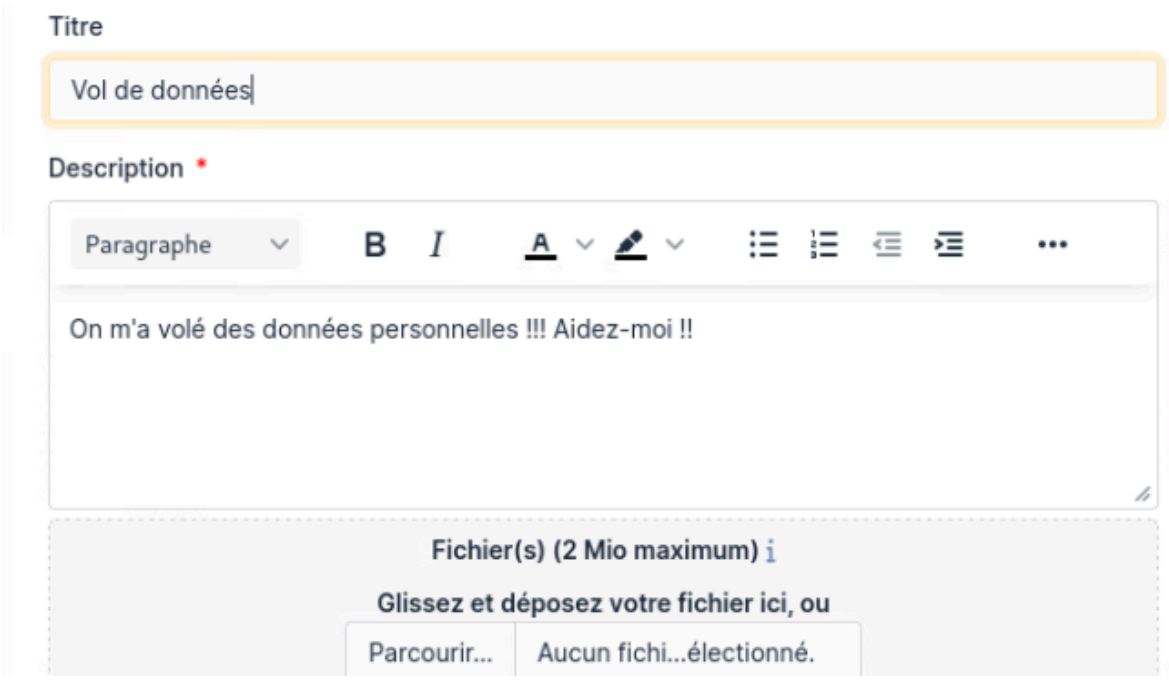


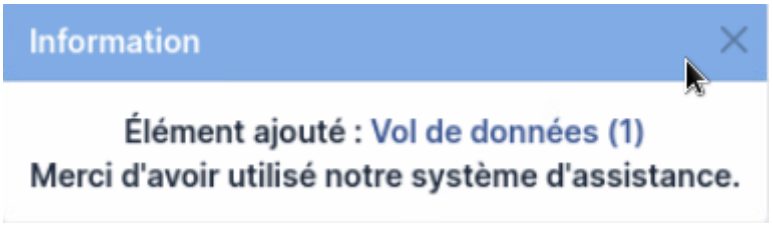
Ici, j'ai créé deux utilisateurs pour les deux salariés de l'entreprise NARFAS (SELVAM Daniel et LEBERT Thomas) dans le cas où ils ont des soucis informatiques et ils auront des droits restreints par rapport aux techniciens (ils pourront créer un ticket par exemple) sur GLPI en intégrant celui-ci dans le profil "Utilisateurs NARFAS".

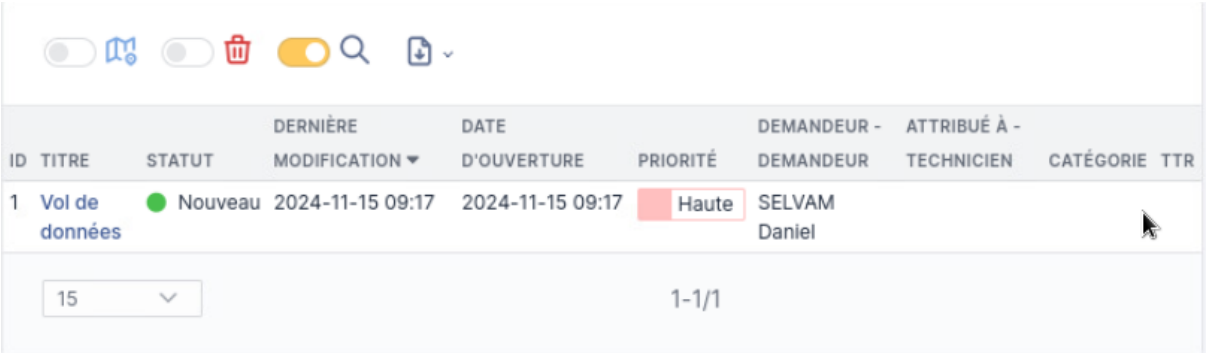
Création du ticket d'intervention par l'utilisateur, prise en main de ce ticket par le technicien avec clôture du ticket :



J'ai réussi à me connecter sur l'utilisateur de SELVAM Daniel et on voit bien qu'il est bien relié au profil "Utilisateurs NARFAS".

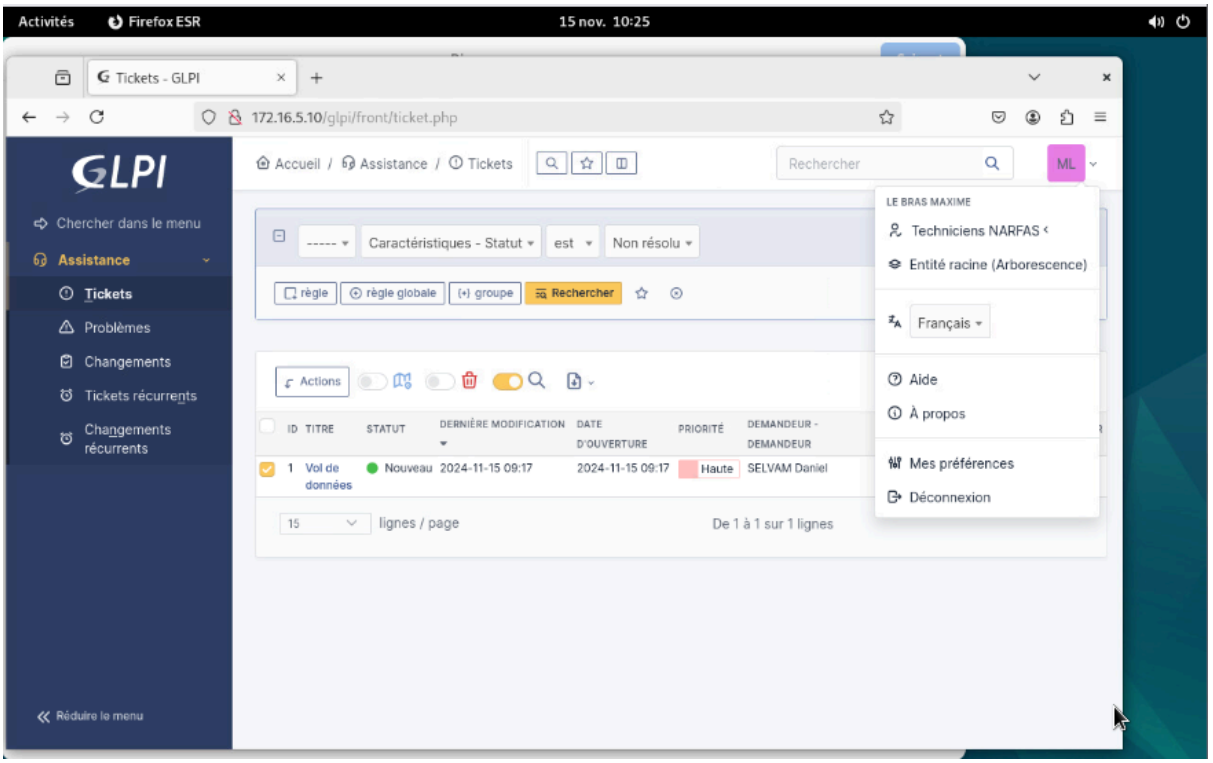


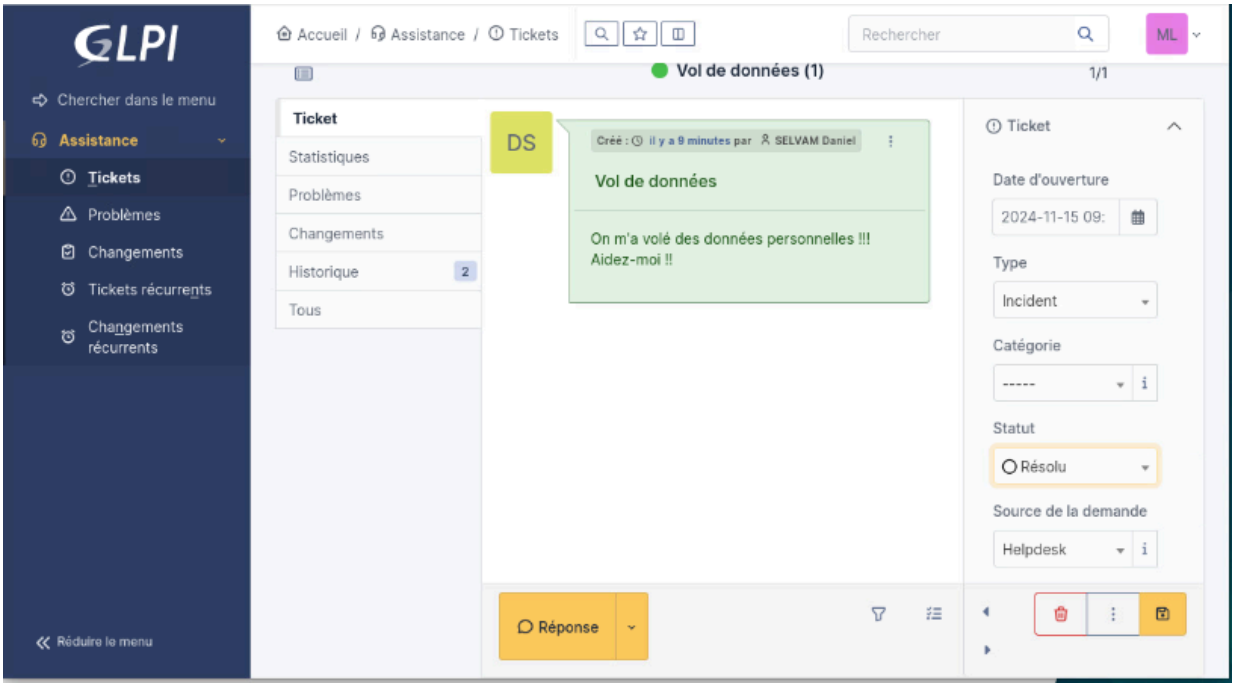




Ici, j'ai créé un ticket d'intervention par rapport à un problème de vol de données concernant l'utilisateur "SELVAM Daniel" et j'ai bien indiqué que la priorité est haute puisque l'incident est très grave.

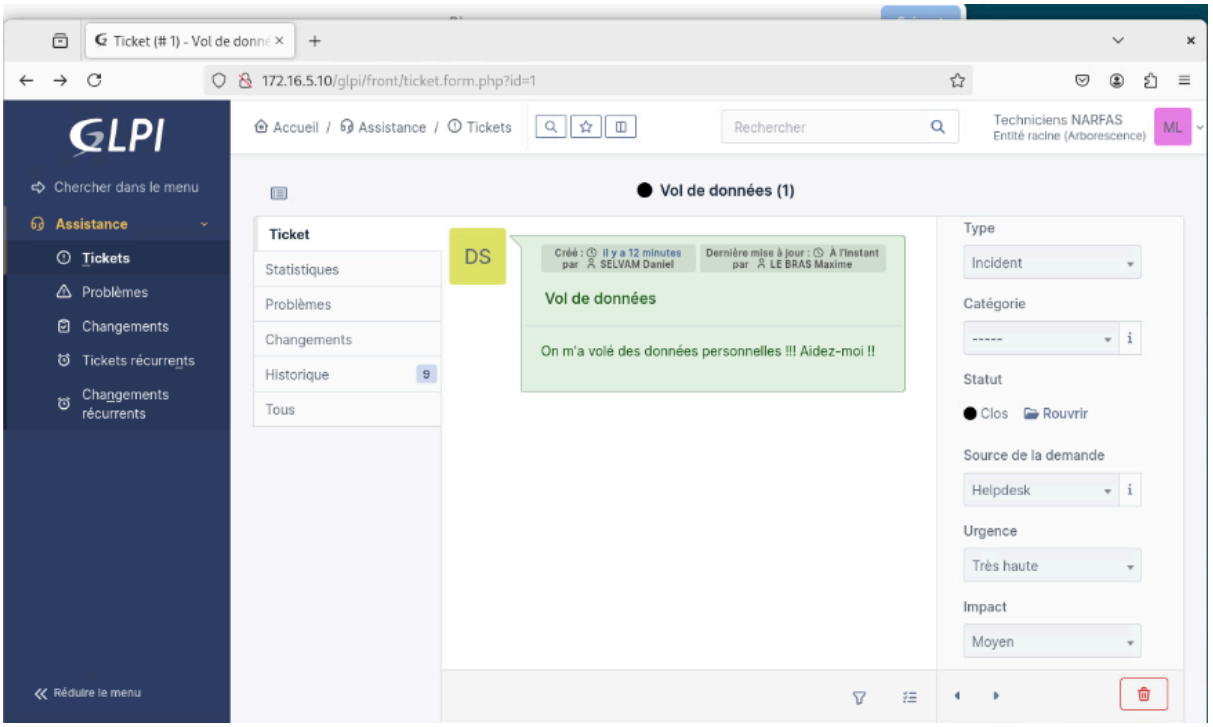
Ici, on s'authentifie au compte technicien "LE BRAS Maxime" pour voir s'il visualise le ticket d'intervention réalisé par SELVAM Daniel précédemment.

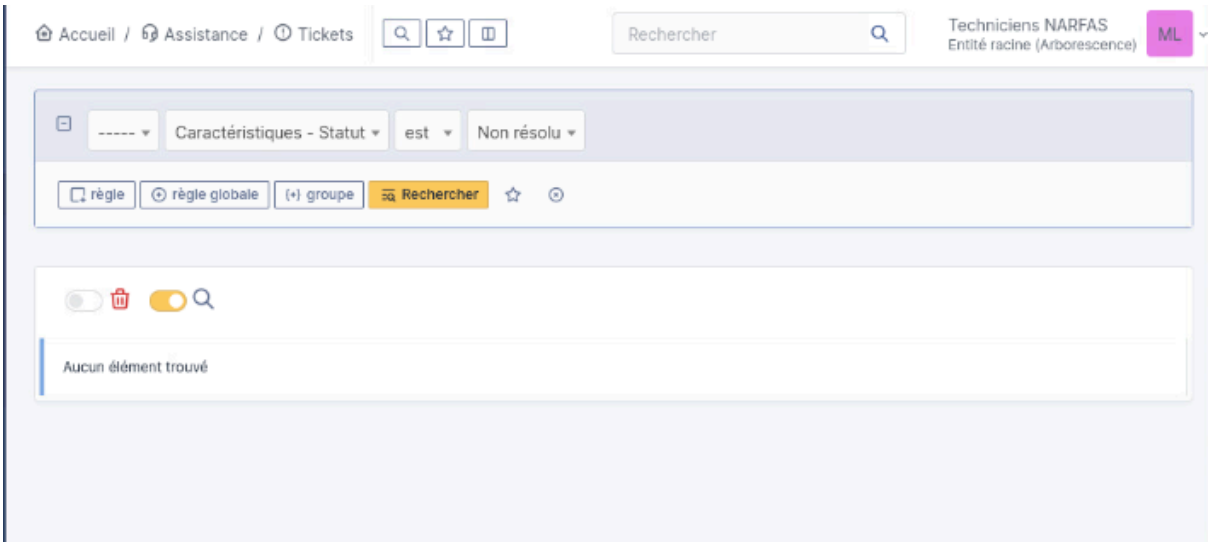




Le technicien LE BRAS Maxime voit bien le ticket d'intervention réalisé par SELVAM Daniel.

On peut attribuer des rôles aux utilisateurs par rapport au ticket d'intervention : le demandeur est celui qui a subi l'incident et qu'il demande de l'aide aux techniciens, l'observateur est celui qui analyse le ticket. Donc la personne à qui on attribue le ticket est l'observateur normalement.

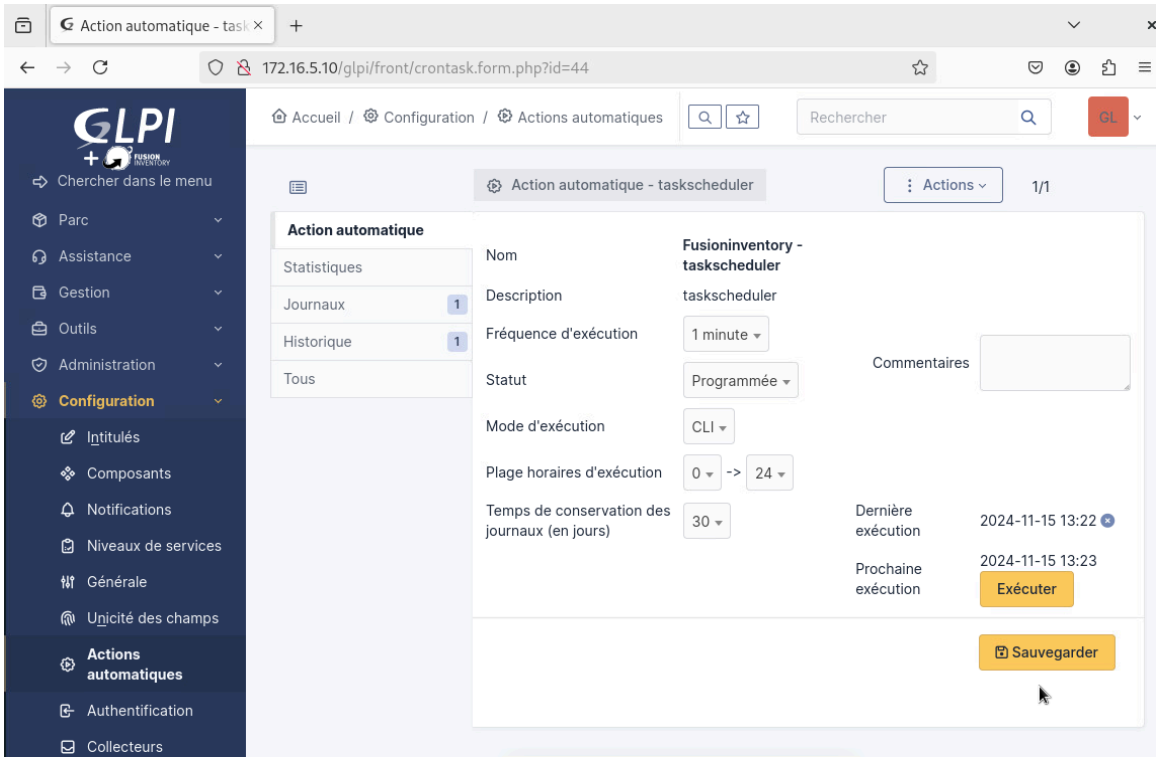




Pour clôturer un ticket, il faut changer le statut du ticket de “Non résolu” à “Clos”. On voit ici que le ticket n’apparaît plus pour le technicien (aussi pour l’utilisateur).

Lot 3 - FusionInventory

Mise en place du fonctionnement du crontab :



Le crontab regroupe les actions automatiques du plugin FusionInventory dans le serveur GLPI et il assure le bon fonctionnement de celles-ci. Pour qu'il fonctionne correctement, il faut aller dans Configuration > Actions automatiques et rechercher dans la barre de recherche "taskscheduler", puis, on doit l'exécuter pour permettre la mise en route du crontab. Si on l'exécute pas, on aura un message d'erreur sur l'interface web en disant qu'il n'est pas fonctionnel.

Installation et paramétrage du plugin FusionInventory dans le serveur GLPI :

```
root@serveurFTPMaxime:/usr/src# wget https://github.com/fusioninventory/fusionin
ventory-for-glpi/releases/download/glpi10.0.6%2B1.1/fusioninventory-10.0.6+1.1.t
ar.bz2
--2024-11-15 11:29:56--  https://github.com/fusioninventory/fusioninventory-for-
glpi/releases/download/glpi10.0.6%2B1.1/fusioninventory-10.0.6+1.1.tar.bz2
Résolution de github.com (github.com)… 140.82.121.4
Connexion à github.com (github.com)|140.82.121.4|:443… connecté.
requête HTTP transmise, en attente de la réponse… 302 Found
Emplacement : https://objects.githubusercontent.com/github-production-release-as
set-2e65be/618511/61d5ef0c-903a-4510-b120-17c66493b029?X-Amz-Algorithm=AWS4-HMAC
```

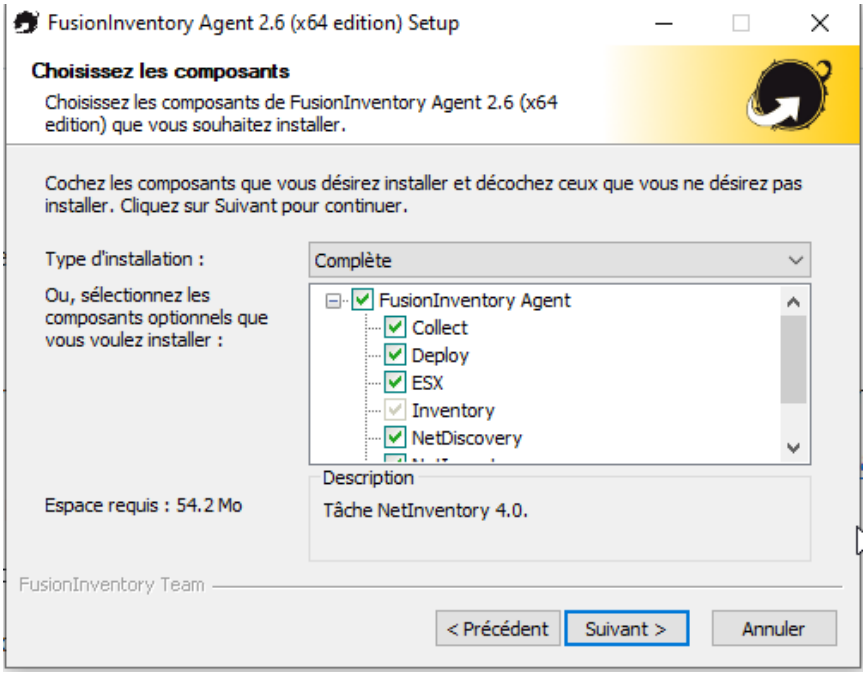
Ici, j'ai été cherché le fichier d'installation du plugin FusionInventory (version 1.1) sur le site de Github et elle est compatible avec la version de mon serveur GLPI (10.0.6).

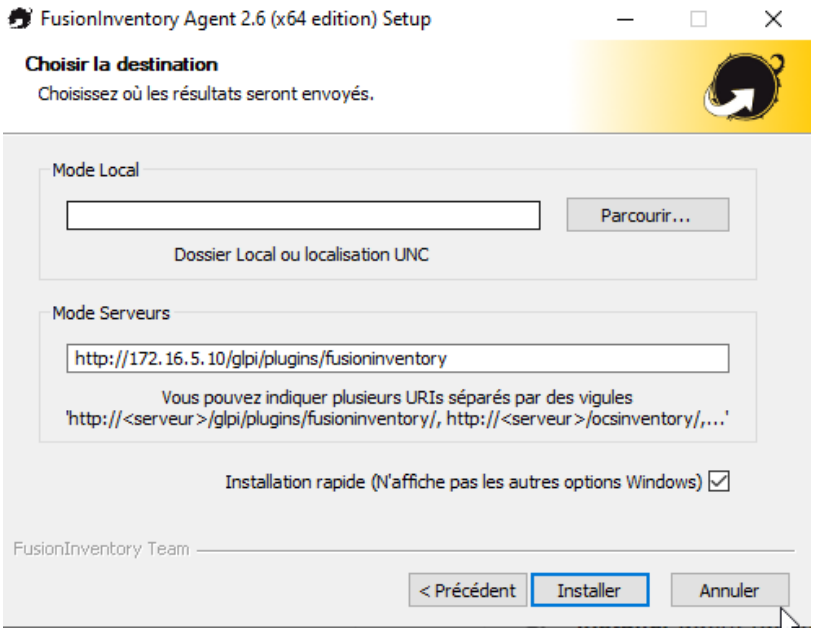
```
root@serveurFTPMaxime:/usr/src# tar -xjf fusioninventory-10.0.6+1.1.tar.bz2 -C /
var/www/html/glpi/plugins
```

Je l'ai décompressé dans le répertoire "/var/www/html/glpi/plugins" afin qu'on repère le plugin FusionInventory dans l'interface web de configuration du serveur GLPI.

On voit que le plugin FusionInventory est bien présent dans GLPI et on l'a bien activé pour qu'il puisse fonctionner avec le serveur GLPI.

Installation d’un agent FusionInventory sur Windows:





Ici, on coche tous les composants pour pouvoir réaliser toutes les tâches possibles sur cet agent FusionInventory et on relie le serveur GLPI et le client FusionInventory.

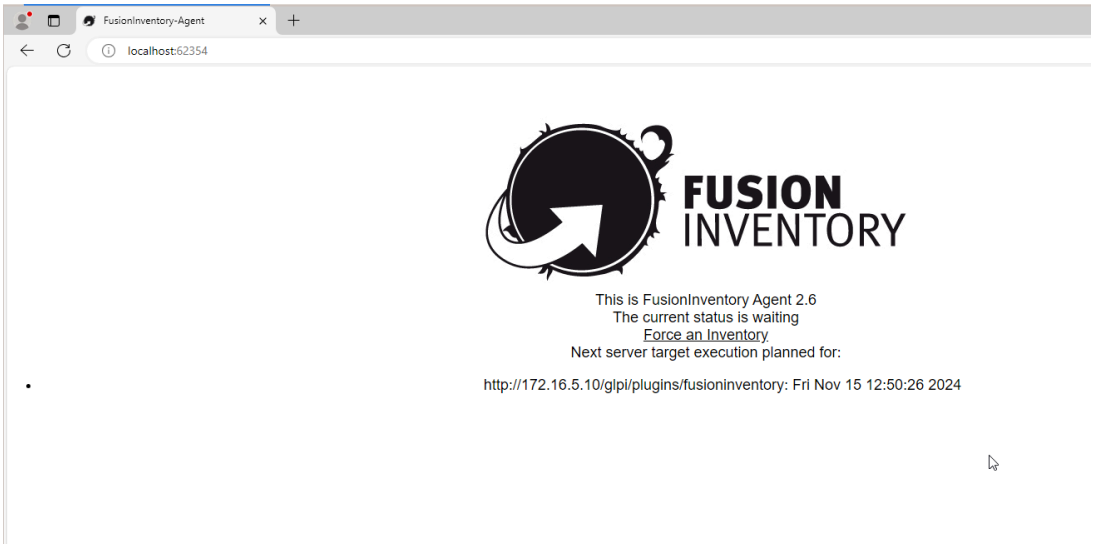
Installation d'un agent FusionInventory sur Linux :

```
root@Debian-12-Bookworm:~# apt-get install fusioninventory-agent
Lecture des listes de paquets... Fait
Construction de l'arbre des dépendances... Fait
Lecture des informations d'état... Fait
Les paquets supplémentaires suivants seront installés :
  hdparm libfile-which-perl libnet-cups-perl libnet-ip-perl libparse-edid-perl
  libproc-daemon-perl libproc-processtable-perl libsocket-getaddrinfo-perl
  libtext-template-perl libuniversal-require-perl libxml-treepp-perl
  libxml-xpath-perl libyaml-libyaml-perl libyaml-perl libyaml-tiny-perl
  net-tools powermgmt-base
Paquets suggérés :
  smartmontools read-edid libyaml-shell-perl
Les NOUVEAUX paquets suivants seront installés :
  fusioninventory-agent hdparm libfile-which-perl libnet-cups-perl
  libnet-ip-perl libparse-edid-perl libproc-daemon-perl
```

```
  GNU nano 7.2                  /etc/fusioninventory/agent.cfg *
# fusioninventory agent configuration

# all defined values match default
# all commented values are examples



#
# Target definition options
#

# send tasks results to an OCS server
#server = http://server.domain.com/ocsinventory
# send tasks results to a FusionInventory for GLPI server
server = http://172.16.5.10/glpi/plugins/fusioninventory/
# write tasks results in a directory
#local = /tmp
```

Ici, j'ai installé le package "fusioninventory-agent" et j'ai configuré l'agent FusionInventory afin qu'il puisse faire la liaison avec le serveur GLPI.

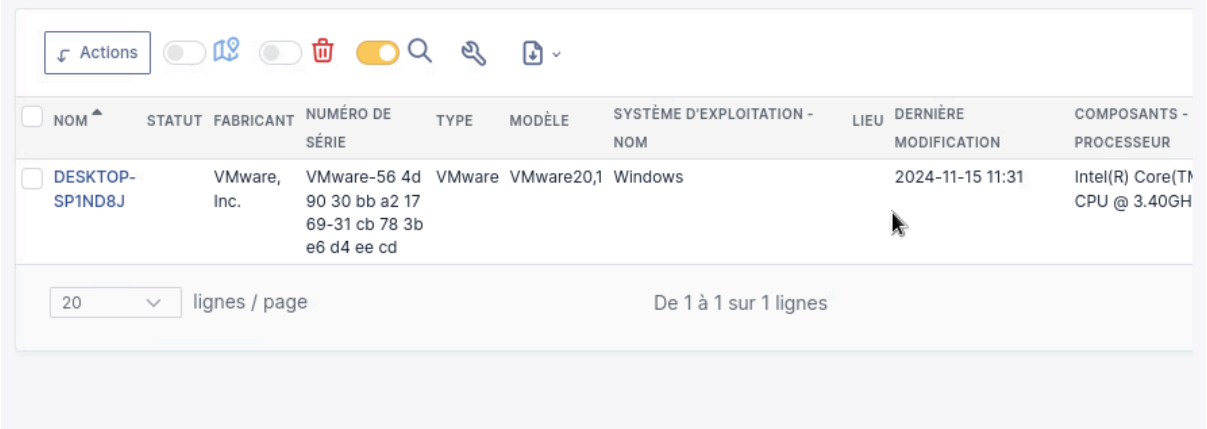
Remontée automatique des informations des agents Windows et Linux sur le serveur GLPI :

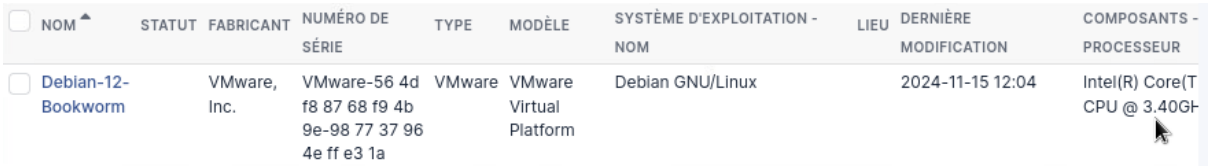



```
root@Debian-12-Bookworm:~# fusioninventory-agent
[info] target server0: server http://172.16.5.10/glpi/plugins/fusioninventory/
[info] sending prolog request to server0
[info] running task Inventory
[info] New inventory from Debian-12-Bookworm-2024-11-15-15-02-39 for server0
root@Debian-12-Bookworm:~#
```

Pour remonter les informations de ces clients, il faut forcer l'inventaire. Sous Linux Debian 12, il faut exécuter la commande "fusioninventory-agent" et sous Windows 10, il faut aller se connecter en localhost avec le port 62354 et sélectionner "Force an Inventory".

Vérification de la remontée automatique des informations des agents FusionInventory :

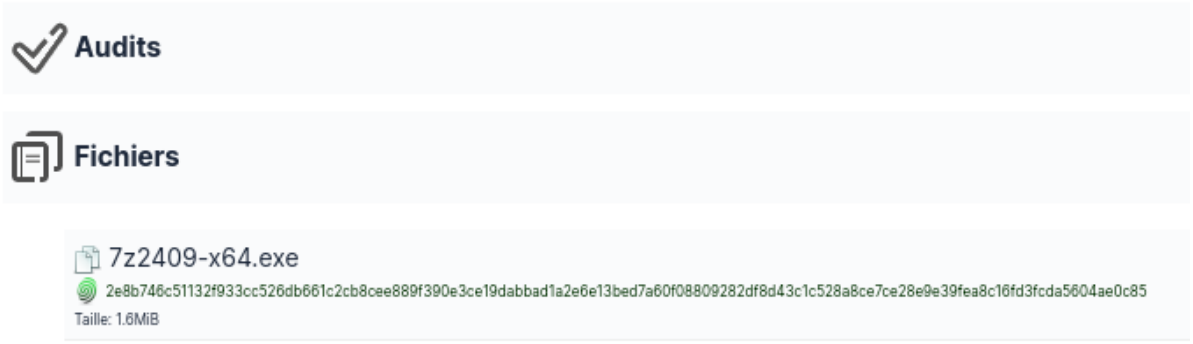




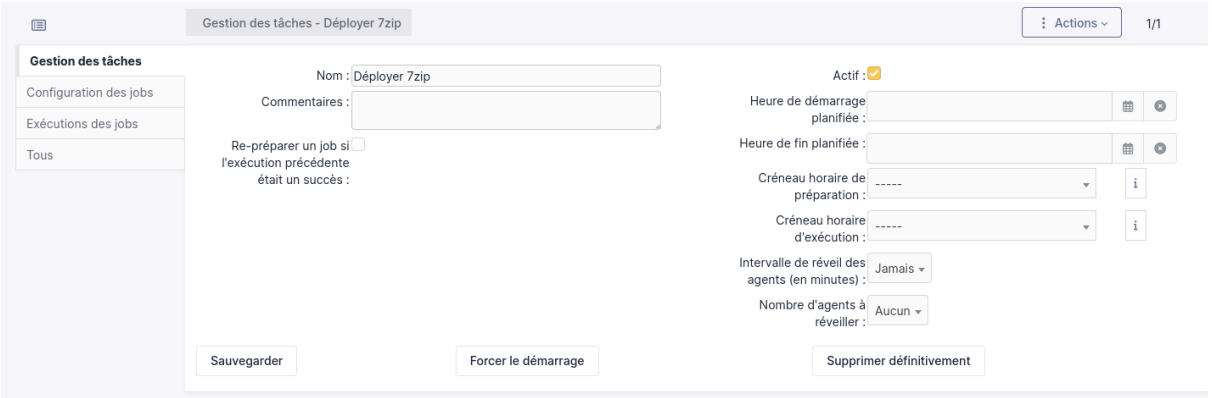
NOM	STATUT	FABRICANT	NUMÉRO DE SÉRIE	TYPE	MODÈLE	SYSTÈME D'EXPLOITATION - NOM	LIEU	DERNIÈRE MODIFICATION	COMPOSANTS - PROCESSEUR
Debian-12-Bookworm		VMware, Inc.	VMware-56 4d f8 87 68 f9 4b 9e-98 77 37 96 4e ff e3 1a	VMware	VMware Virtual Platform	Debian GNU/Linux		2024-11-15 12:04	Intel(R) Core(T CPU @ 3.40GH

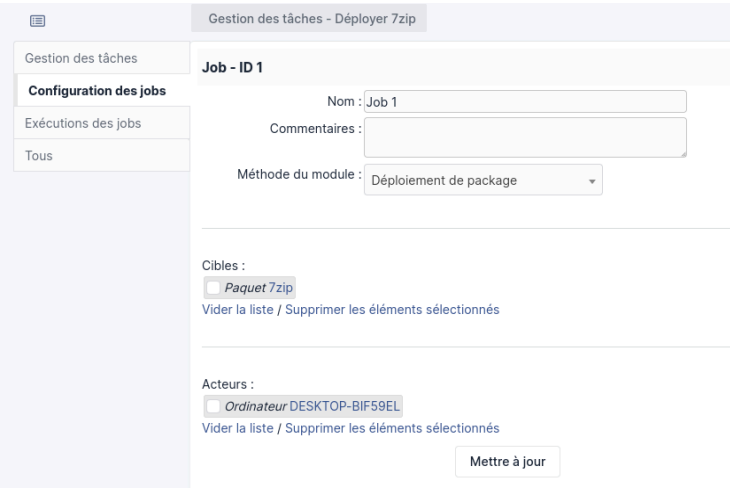
J'ai réussi à faire remonter les informations des différents agents FusionInventory dans le serveur GLPI. On peut avoir les informations sur les composants de ces agents avec les logiciels installés, le système d'exploitation utilisé, le processeur, les interfaces réseaux, etc…

<u>Déploiement automatique d'un package (contenant le logiciel 7zip) sur l'agent FusionInventory Windows :</u>

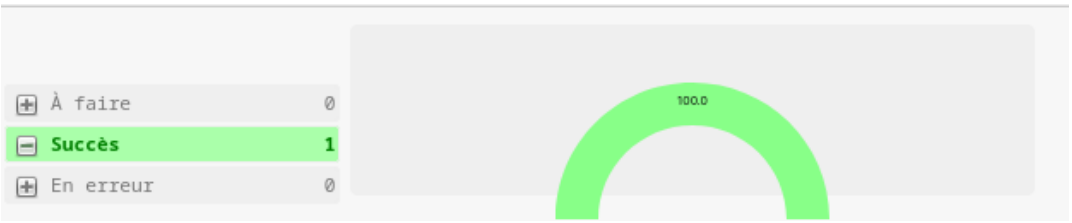


Tout d'abord, il faut créer le package où il contiendra le fichier exécutable de 7zip ainsi que des lignes de commande permettant l'installation silencieuse de 7zip sur l'agent FusionInventory Windows. Pour créer et configurer un package, il faut aller dans Administration > FusionInventory > Déployer > Gestion des paquets > Ajouter.





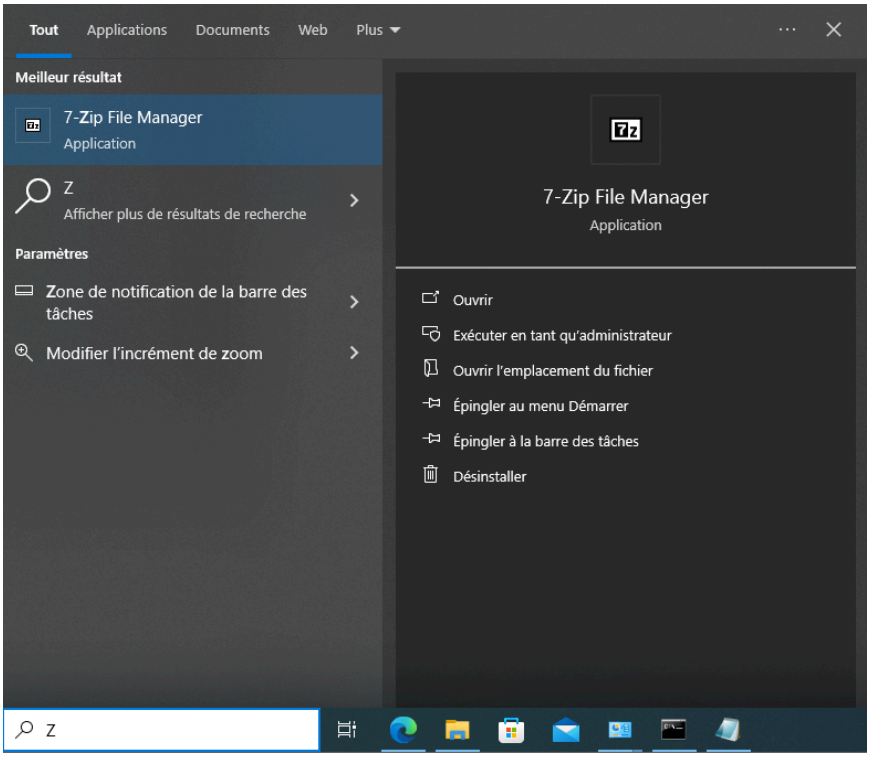
Puis, il faut mettre en place la tâche de déploiement en introduisant le package (7zip) précédemment en créant la tâche. Pour créer et configurer une tâche, il faut aller dans Administration > FusionInventory > Tâches > Gestion des tâches > Ajouter. Ensuite, il faut configurer le “job” qui va faire le déploiement du serveur GLPI/FusionInventory à l’agent FusionInventory Windows. Après avoir réalisé toutes ces étapes, il faut planifier la tâche ou forcer le démarrage pour pouvoir réaliser la tâche de déploiement demandée.




```
[Sat Nov 30 15:22:21 2024][debug2] Finished job 674b1bd6da659...
[Sat Nov 30 15:22:21 2024][debug2] http://172.16.5.1/glpi/plugins/fusioninventory/b/deploy/?action=setStatus&uuid=674b1bd6da659&part=job&status=ok&machineid
[Sat Nov 30 15:22:21 2024][debug2] All deploy jobs processed
```

On remarque ici que la tâche a bien été exécutée et réussie à priori !

Vérification du déploiement de 7zip sur l'agent FusionInventory :



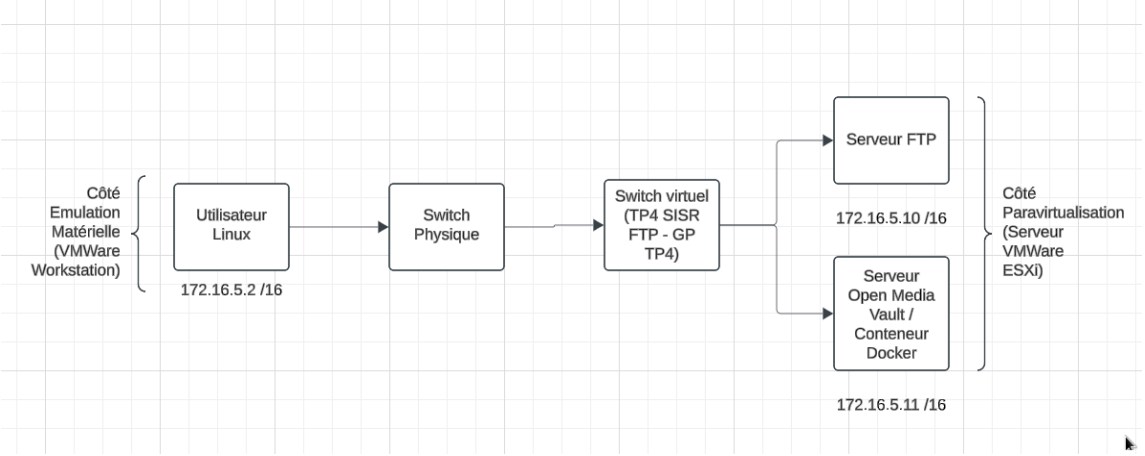
Le logiciel 7zip a bien été installé sur l'agent FusionInventory Windows, donc la tâche demandée a été réussie.

Difficultés rencontrées :

La difficulté principale que j'ai réussi à résoudre au bout d'un certain temps, est de démarrer le déploiement, alors que les procédures et les mises en place sont fonctionnelles. La solution que j'ai trouvé, est d'appuyer plusieurs fois sur l'option "Forcer le démarrage" jusqu'à temps que la tâche se déploie.

Lot 4 - NAS Open Source “Open Media Vault”

Schéma réseau :



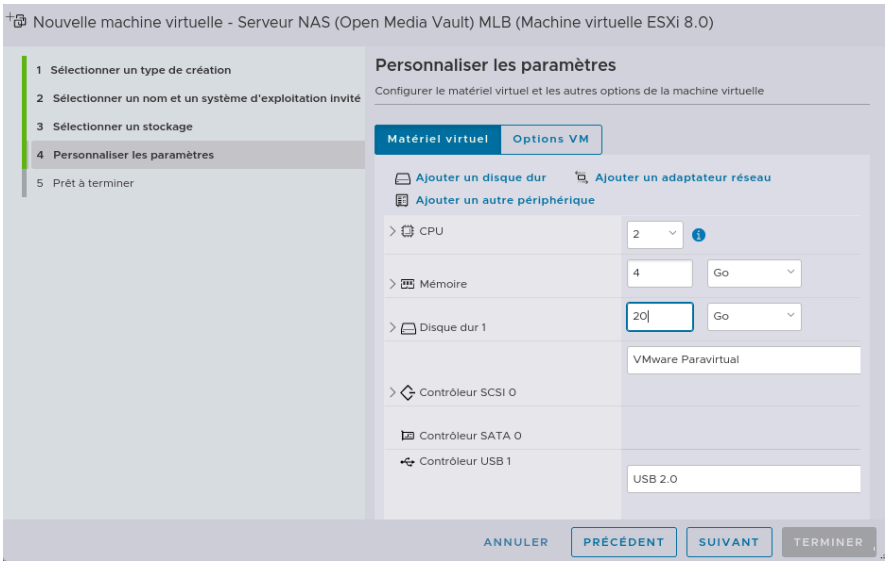
Prérequis au niveau de la configuration du serveur Debian 12 pour mettre en place Open Media Vault :

Prerequisites

Before installing openmediavault make sure your hardware is supported.

- **CPU**: Any x86-64 or ARM compatible processor
- **RAM**: 1 GiB capacity
- **HDD**:
  - **System Drive**: min. 4 GiB capacity (but more than the capacity of the RAM)
  - **Data Drive**: capacity according to your needs

Configuration du serveur Debian 12 pour héberger Open Media Vault :



Tests de fonctionnement de la communication entre les machines du réseau :

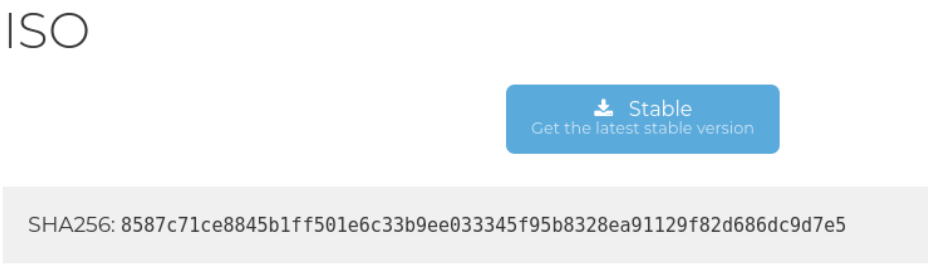
```
root@UtilisateurLinuxNAS:~# ping 172.16.5.10
PING 172.16.5.10 (172.16.5.10) 56(84) bytes of data.
64 bytes from 172.16.5.10: icmp_seq=1 ttl=64 time=0.854 ms
^C
--- 172.16.5.10 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.854/0.854/0.854/0.000 ms
root@UtilisateurLinuxNAS:~# ping 172.16.5.11
PING 172.16.5.11 (172.16.5.11) 56(84) bytes of data.
64 bytes from 172.16.5.11: icmp_seq=1 ttl=64 time=0.816 ms
64 bytes from 172.16.5.11: icmp_seq=2 ttl=64 time=0.567 ms
^C
--- 172.16.5.11 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1022ms
rtt min/avg/max/mdev = 0.567/0.691/0.816/0.124 ms
root@UtilisateurLinuxNAS:~#
```

```
root@serveurFTPMaxime:~# ping 172.16.5.11
PING 172.16.5.11 (172.16.5.11) 56(84) bytes of data.
64 bytes from 172.16.5.11: icmp_seq=1 ttl=64 time=0.551 ms
64 bytes from 172.16.5.11: icmp_seq=2 ttl=64 time=0.388 ms
^C
--- 172.16.5.11 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1016ms
rtt min/avg/max/mdev = 0.388/0.469/0.551/0.081 ms
root@serveurFTPMaxime:~# ping 172.16.5.2
PING 172.16.5.2 (172.16.5.2) 56(84) bytes of data.
64 bytes from 172.16.5.2: icmp_seq=1 ttl=64 time=0.986 ms
^C
--- 172.16.5.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.986/0.986/0.986/0.000 ms
root@serveurFTPMaxime:~#
```

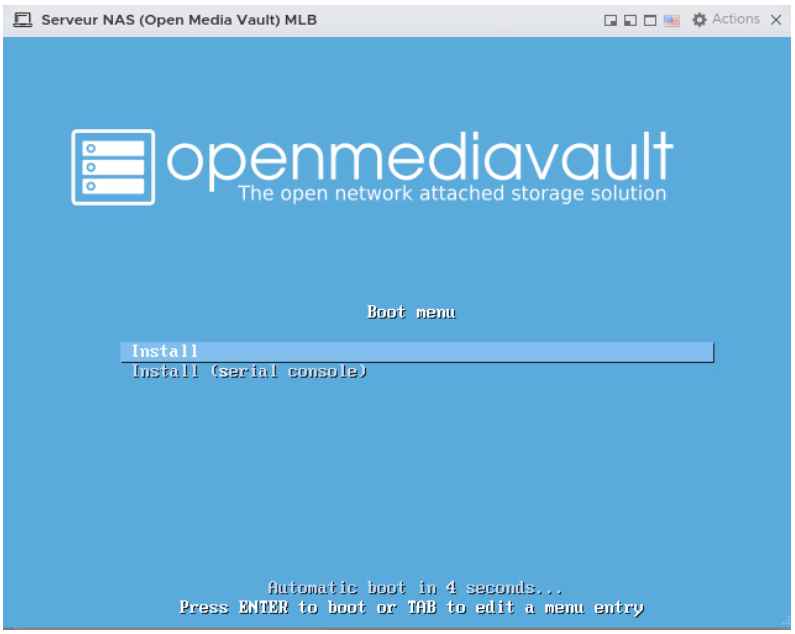
```
root@ServeurNAS-OMV-Maxime:~# ping 172.16.5.2
PING 172.16.5.2 (172.16.5.2) 56(84) bytes of data.
64 bytes from 172.16.5.2: icmp_seq=1 ttl=64 time=1.52 ms
64 bytes from 172.16.5.2: icmp_seq=2 ttl=64 time=1.11 ms
^C
--- 172.16.5.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.112/1.316/1.520/0.204 ms
root@ServeurNAS-OMV-Maxime:~# ping 172.16.5.10
PING 172.16.5.10 (172.16.5.10) 56(84) bytes of data.
64 bytes from 172.16.5.10: icmp_seq=1 ttl=64 time=0.568 ms
64 bytes from 172.16.5.10: icmp_seq=2 ttl=64 time=0.422 ms
^C
--- 172.16.5.10 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.422/0.495/0.568/0.073 ms
root@ServeurNAS-OMV-Maxime:~#
```

Comme les commandes ping fonctionnent entre toutes les machines (Utilisateur et serveurs) alors le réseau est opérationnel.

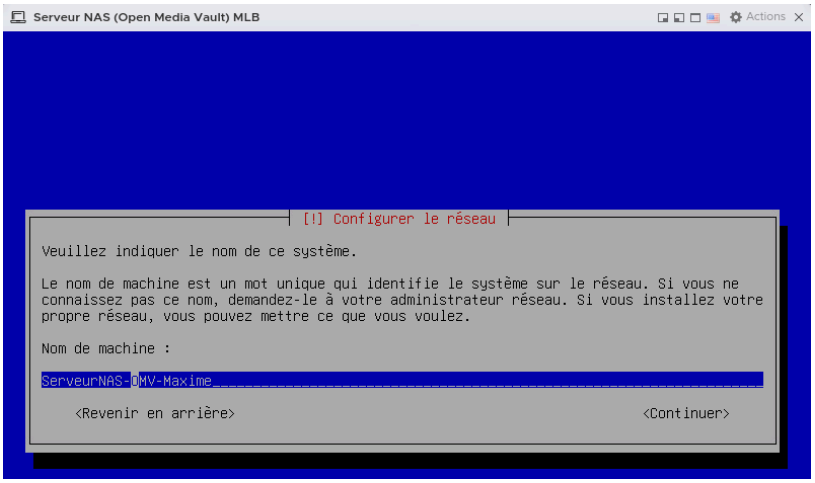
Installation et paramétrage d'Open Media Vault :

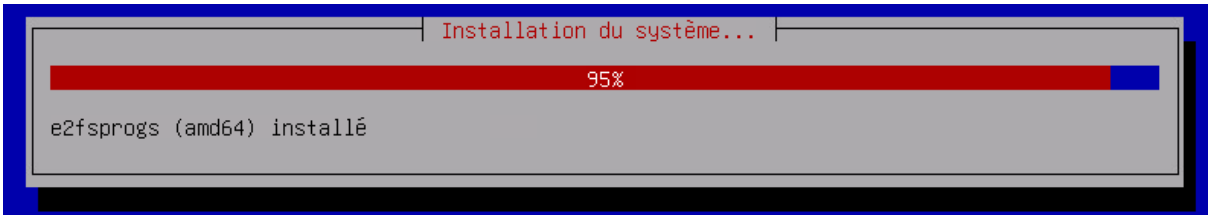


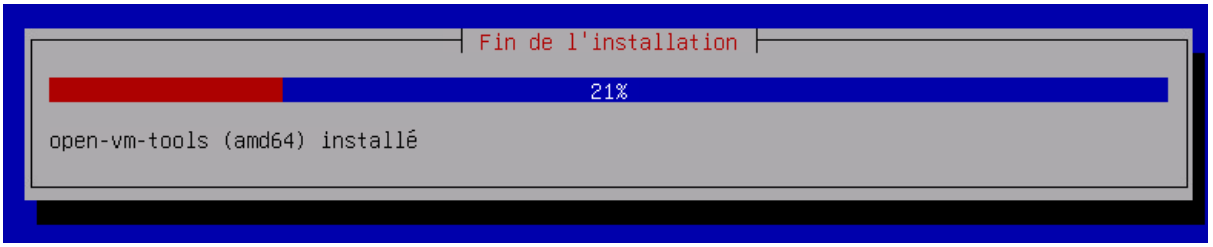
Tout d'abord, il faut télécharger l'image ISO d'Open Media Vault, puisque OMV est une distribution Linux. Open Media Vault est un serveur NAS (Network Attached Storage) qui est open source et il offre des avantages conséquents avec la centralisation des données, la sauvegarde des données, l'accessibilité au réseau local et le coût moindre (gratuit) de cette solution. Il peut être amélioré avec l'introduction de plugins comme Docker (omv-extras). Aussi, il peut contenir des services comme SSH (Secure Shell), SFTP (Secure File Transfer Protocol), SMB/CIFS (Secure Message Block/Common Internet File System), etc…



Ici, on accède à la phase d'installation et de configuration du serveur NAS OpenMediaVault.







J’ai configuré et installé le serveur NAS avec succès. De plus, j’ai mis en place un nom de domaine, l’authentification root avec un mot de passe pour pouvoir modifier certains paramètres et effectuer des installations sur le serveur, le gestionnaire de démarrage GRUB sur /dev/sda1(l’installation de OMV est similaire à une installation classique d’une distribution Debian 12)

```
openmediavault 7.0-32 (Sandworm) ServeurNAS-OMV-Maxime tty1
Copyright (C) 2009-2024 by Volker Theile. All rights reserved.

To manage the system visit the openmediavault workbench:

ens33: 172.16.4.18

By default the workbench administrator account has the
username 'admin' and password 'openmediavault'.
It is recommended that you change the password for this account
within the workbench or using the 'omv-firstaid' CLI command.

For more information regarding this appliance, please visit the
web site: https://www.openmediavault.org

ServeurNAS-OMV-Maxime login: _
```

On remarque que le paramétrage réseau est fonctionnel puisqu'on voit qu'on a adressé une adresse IP dynamique provenant du serveur DHCP (Dynamic Host Configuration Protocol) du lycée (switch virtuel avec vmnic1 qui est connecté à Internet).

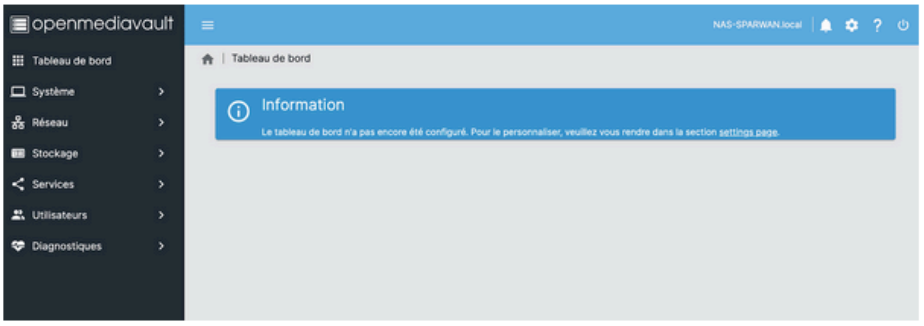
Se connecter à l'interface

Une fois sur l'écran de connexion, il vous suffit de saisir les identifiants suivants :

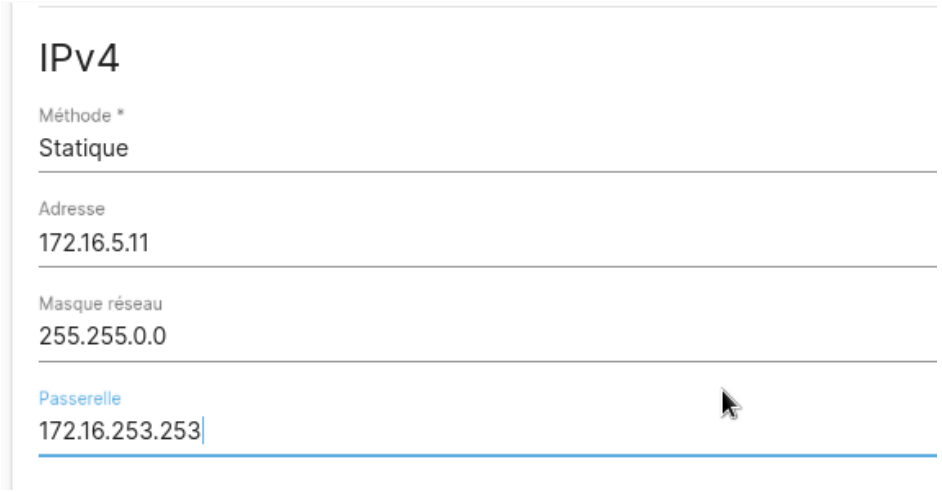
Les identifiants par défaut sont :

Nom d'utilisateur : admin

Mot de passe : openmediavault



D'après les informations que j'ai trouvé sur Internet, l'authentification par défaut à l'interface web de configuration d'Open Media Vault est la suivante : admin (nom d'utilisateur) et openmediavault (mot de passe).



Ici, j'accède à l'interface web d'OMV et on va mettre en place une configuration IP statique puisque c'est un serveur. J'ai configuré ceci en fonction de la plage d'adresses et du réseau que j'ai mis en place. Faire cette configuration me permettra d'avoir accès à Internet et de réaliser des commandes d'installation de plugins via Internet, ce qui va me procurer un gain de temps incroyable.

```
root@ServeurNAS-OMV-Maxime:~# 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 noprefixroute
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:a5:f9:88 brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 172.16.5.11/16 brd 172.16.255.255 scope global ens33
       valid_lft forever preferred_lft forever
root@ServeurNAS-OMV-Maxime:~#
```

```
root@ServeurNAS-OMV-Maxime:~# ping www.google.fr
PING www.google.fr (142.250.201.163) 56(84) bytes of data.
64 bytes from par21s23-in-f3.1e100.net (142.250.201.163): icmp_seq=1 ttl=116 time=4.44 ms
64 bytes from par21s23-in-f3.1e100.net (142.250.201.163): icmp_seq=2 ttl=116 time=4.69 ms
64 bytes from par21s23-in-f3.1e100.net (142.250.201.163): icmp_seq=3 ttl=116 time=4.74 ms
64 bytes from par21s23-in-f3.1e100.net (142.250.201.163): icmp_seq=4 ttl=116 time=4.66 ms
^C
--- www.google.fr ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 4.438/4.633/4.743/0.116 ms
```

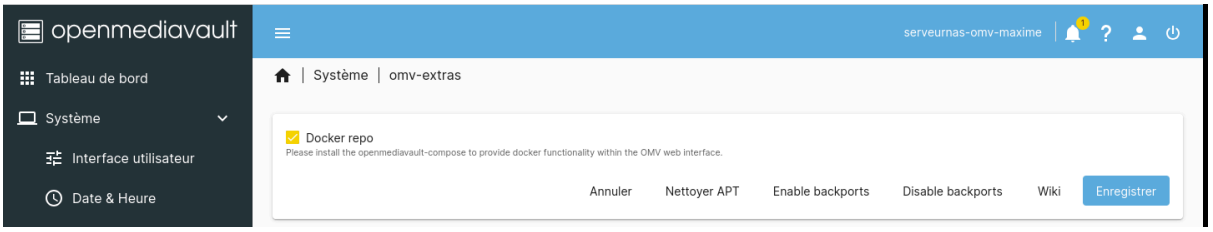
On peut conclure que la mise en place de la configuration réseau a bien été réalisée.

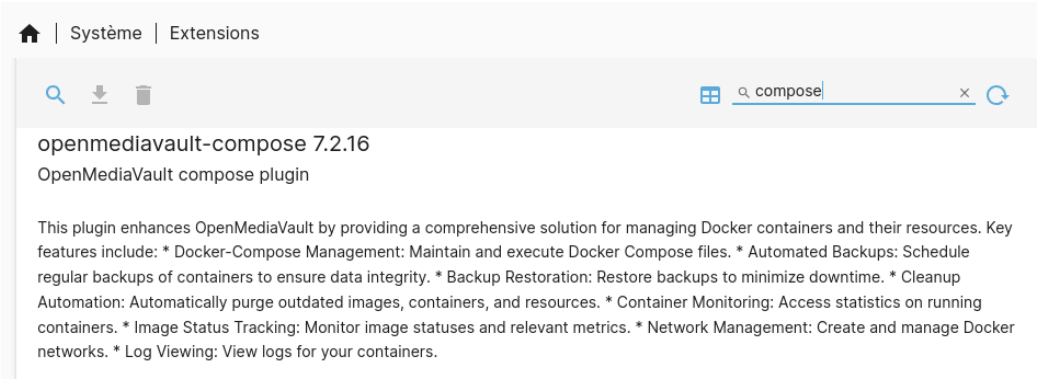
Installation de l’application ZoneMinder via un conteneur Docker (Portainer) :

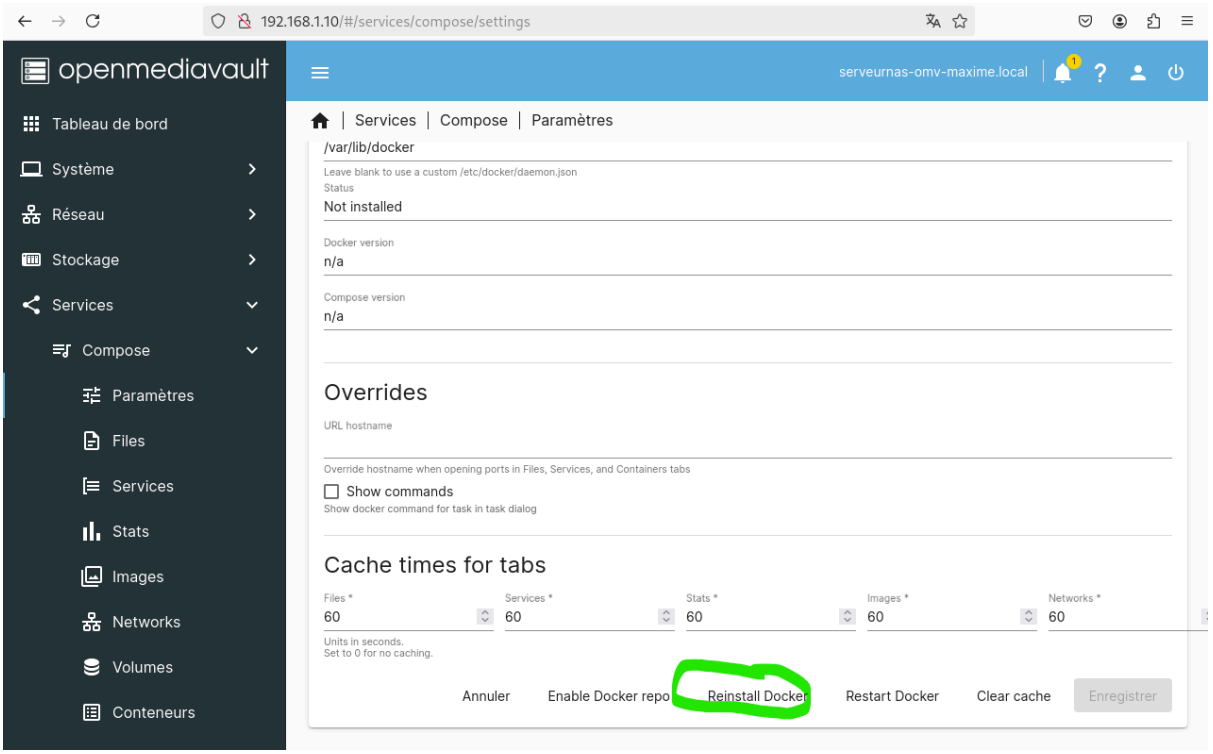
Tout d’abord, ZoneMinder est une solution open source efficace de vidéosurveillance, acceptant des sources vidéo analogiques (connectées via une carte d'acquisition), des webcams ou encore des caméras réseau.

```
root@ServeurNAS-OMV-Maxime:~# wget -O - https://github.com/OpenMediaVault-Plugin-Developers/packages
/raw/master/install | bash
```

Cette commande permet d’installer omv-extras via un site web ce qui permettra d’installer les plugins OMV selon les besoins attendus.



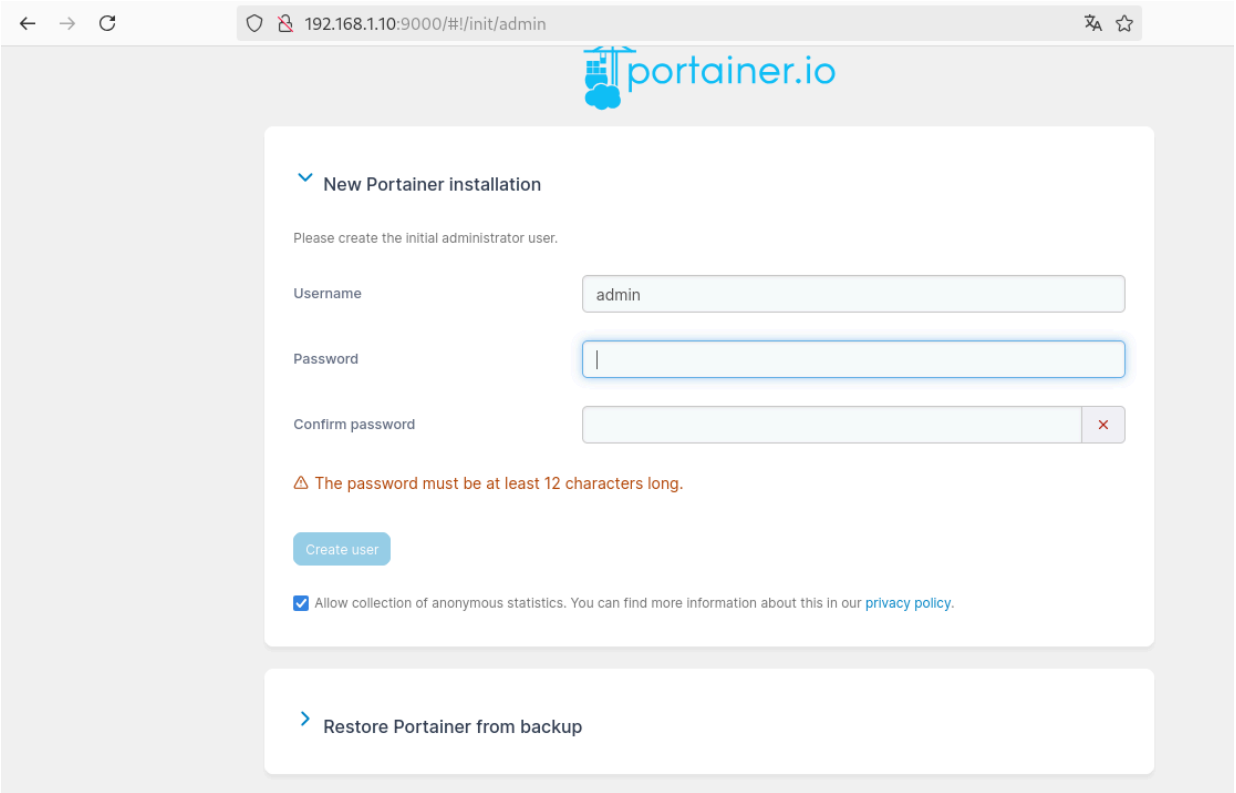




Toutes ces configurations permettent l'installation de Docker dans le serveur NAS OMV. Pour installer Docker correctement, il faut appuyer sur l'option "Reinstall Docker".

```
root@ServeurNAS-OMV-Maxime:~# docker pull portainer/portainer-ce
Using default tag: latest
latest: Pulling from portainer/portainer-ce
2a8c27161aa3: Pull complete
679061c2c821: Pull complete
d40df14c1d7a: Pull complete
8215717c7c10: Pull complete
542669febe7c: Pull complete
6c27c7f45b54: Pull complete
070d3bf2528e: Pull complete
846480e9f8b0: Pull complete
c7053d7d4c2a: Pull complete
a2ed6de7fb5f: Pull complete
4f4fb700ef54: Pull complete
Digest: sha256:f2a7f5abd4735f9cd91563c6134e014b15168c4018beea87f1eec9d9618b2ad4
Status: Downloaded newer image for portainer/portainer-ce:latest
docker.io/portainer/portainer-ce:latest
root@ServeurNAS-OMV-Maxime:~# docker run -d -p 9000:9000 --name portainer --restart always -v /var/r
un/docker.sock:/var/run/docker.sock -v portainer_data:/data portainer/portainer-ce
64fab4858863206c3f1d29d4d6e51e8918ee7cd4a4fc10d1c41b126a80367003
root@ServeurNAS-OMV-Maxime:~#
```

Cela représente la mise en place de Portainer, il a les mêmes fonctionnalités que Docker sauf que c'est plus facile d'utilisation puisqu'on utilise uniquement l'interface graphique pour gérer les conteneurs alors qu'avec Docker, on doit faire cette gestion via une ligne de commande.

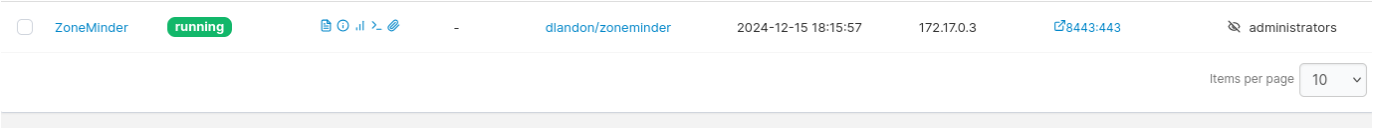


Pour pouvoir prendre en main Portainer, il faut se connecter à l'adresse IP du serveur NAS (192.168.1.10) via le port 9000 et il faut configurer l'authentification du compte admin.

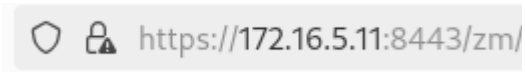
```
root@ServeurNAS-OMV-Maxime:~# docker pull dlandon/zoneminder
```

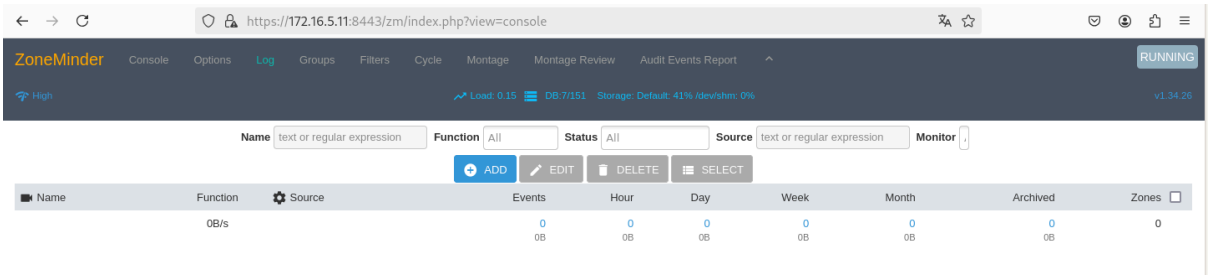
```
root@ServeurNAS-OMV-Maxime:~# docker run -d --name="ZoneMinder" --net="bridge" --privileged="false"
--shm-size="5G" -p 8443:443 -e TZ="Europe/Paris" -e PUID="99" -e PGID="100" -e INSTALL_HOOK="0" -e I
NSTALL_FACE="0" -e INSTALL_TINY_YOLOV3="0" -e INSTALL_YOLOV3="0" -e INSTALL_TINY_YOLOV4="0" -e INSTA
LL_YOLOV4="0" -e MULTI_PORT_START="0" -e MULTI_PORT_END="0" -v "/mnt/Zoneminder/data":"/var/cache/zo
neminder":rw dlandon/zoneminder_
```

Ici, j'ai mis en place le conteneur Docker ZoneMinder en ligne de commande.



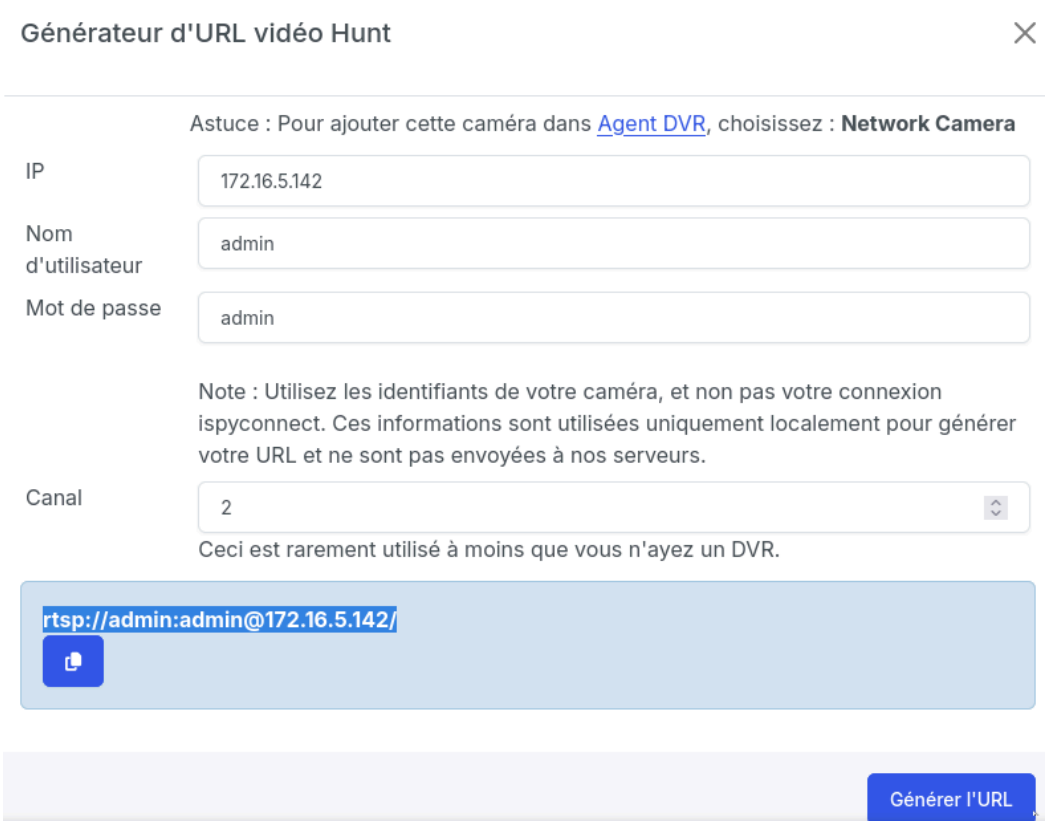
On voit bien que ce conteneur a bien été configuré et qu'il est en état de fonctionnement.

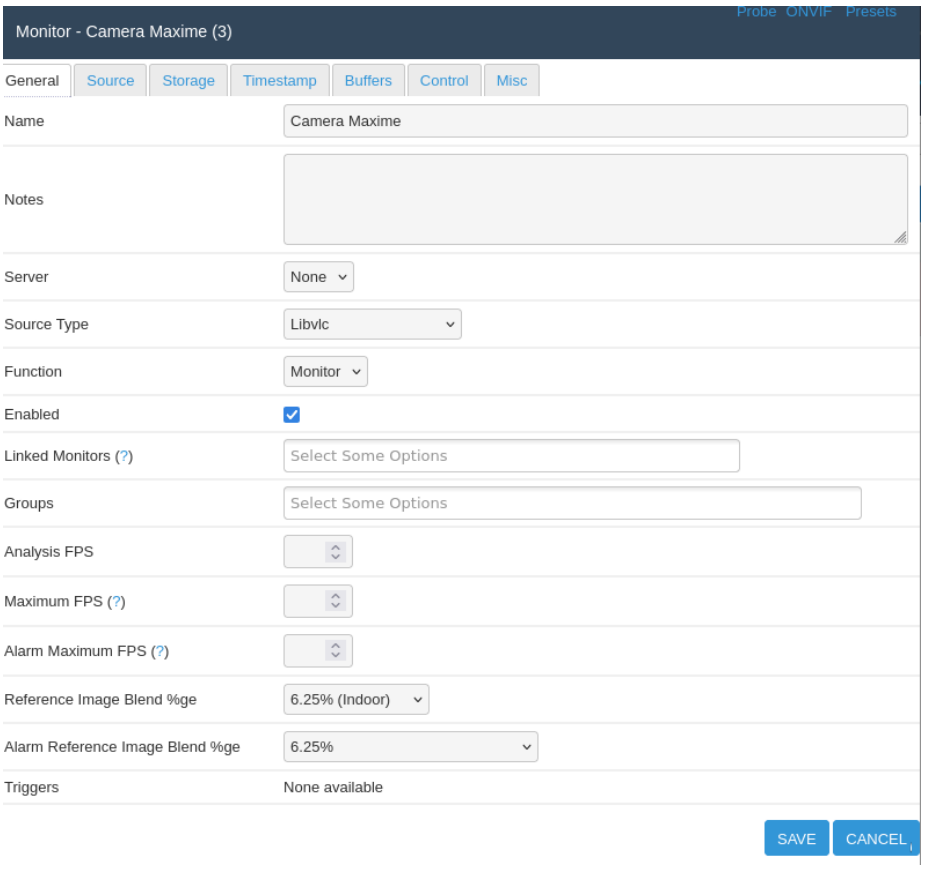


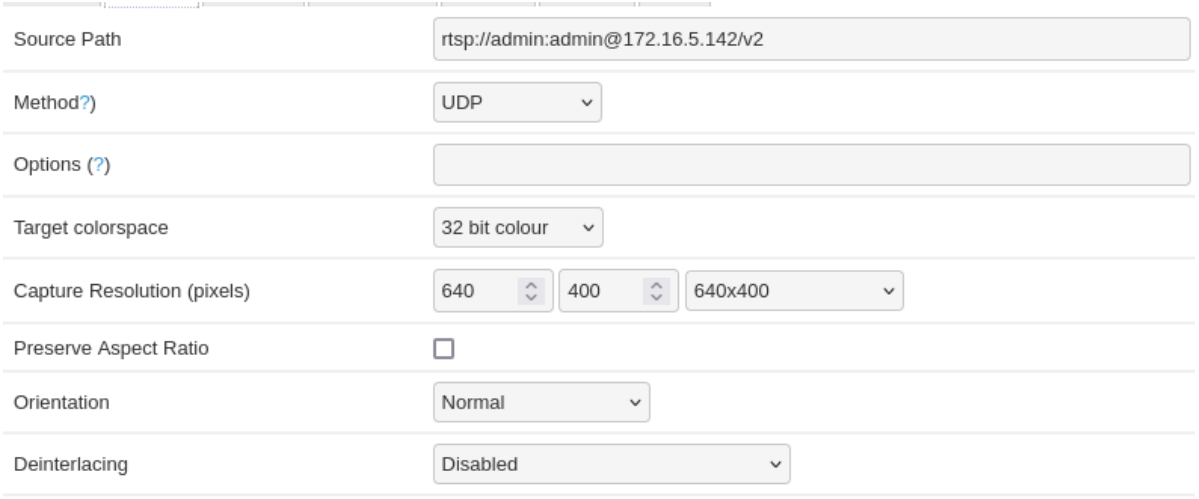


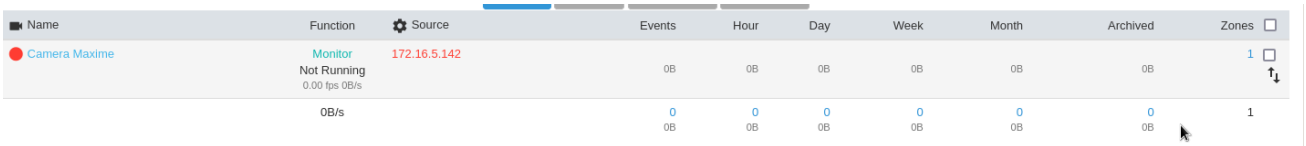
Pour pouvoir se connecter à l'interface web du conteneur ZoneMinder et procéder à son installation, il faut taper l'url suivante : **https://172.16.5.11:8443/zm/** La connexion au conteneur et l'installation de l'application ont bien été effectuées.

Création des partages nécessaires à la sauvegarde des enregistrements de ZoneMinder :







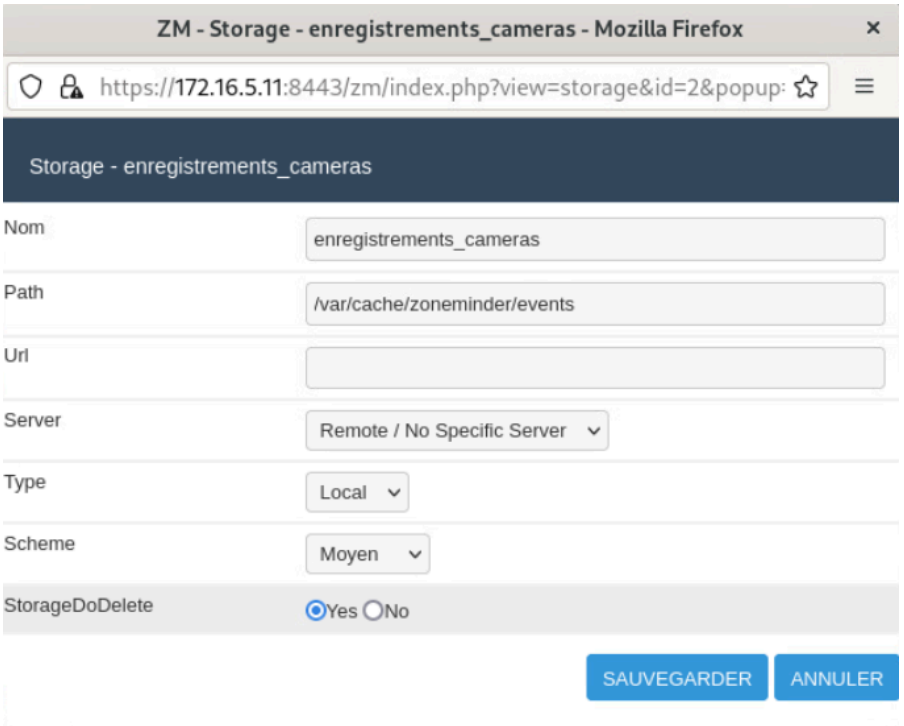


Tout d'abord, on fait la mise en place de la caméra de surveillance IP qui est reliée à notre réseau.

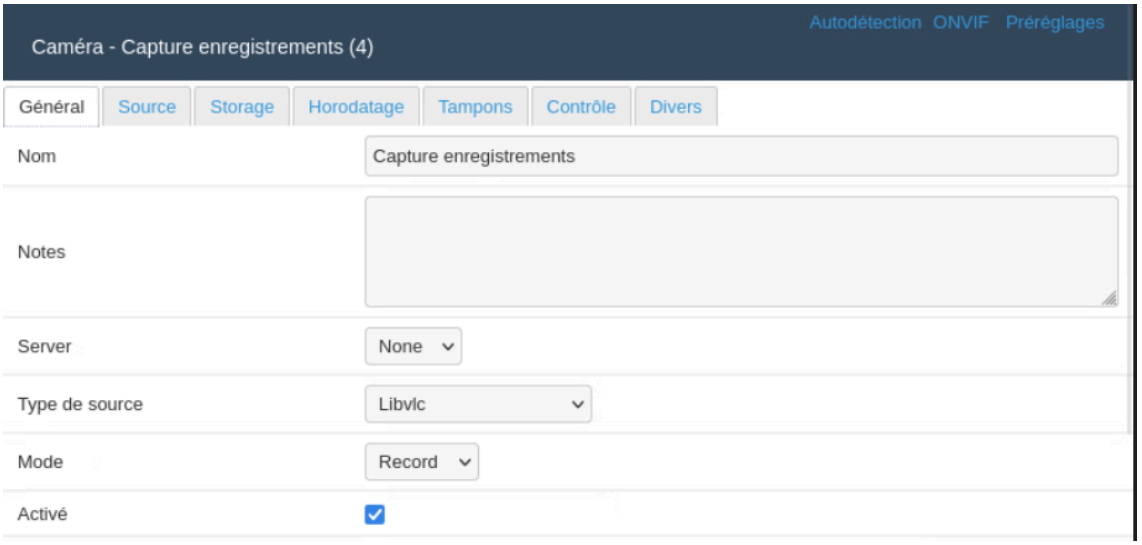
```
root@ServeurNAS-OMV-Maxime:~# ping 172.16.5.142
PING 172.16.5.142 (172.16.5.142) 56(84) bytes of data.
64 bytes from 172.16.5.142: icmp_seq=1 ttl=64 time=1.10 ms
64 bytes from 172.16.5.142: icmp_seq=2 ttl=64 time=1.03 ms
64 bytes from 172.16.5.142: icmp_seq=3 ttl=64 time=1.01 ms
64 bytes from 172.16.5.142: icmp_seq=4 ttl=64 time=0.451 ms
64 bytes from 172.16.5.142: icmp_seq=5 ttl=64 time=1.17 ms
^C
--- 172.16.5.142 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 0.451/0.950/1.167/0.255 ms
root@ServeurNAS-OMV-Maxime:~#
```

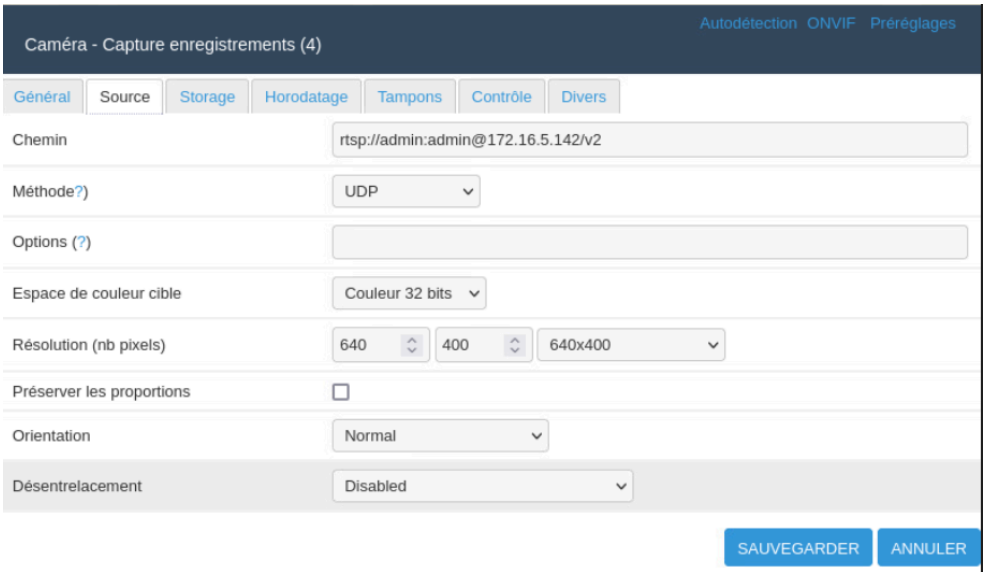
```
root@UtilisateurLinuxNAS:~# ping 172.16.5.142
PING 172.16.5.142 (172.16.5.142) 56(84) bytes of data.
64 bytes from 172.16.5.142: icmp_seq=1 ttl=64 time=2.47 ms
64 bytes from 172.16.5.142: icmp_seq=2 ttl=64 time=0.732 ms
^C
--- 172.16.5.142 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1003ms
rtt min/avg/max/mdev = 0.732/1.602/2.473/0.870 ms
root@UtilisateurLinuxNAS:~#
```

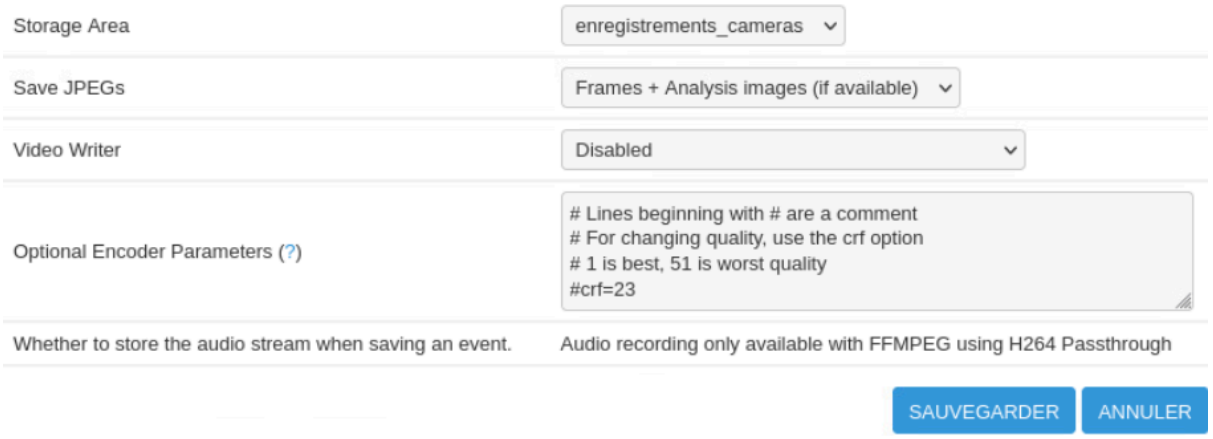
On voit ici que la caméra communique bien avec toute l'infrastructure réseau (machines : client, serveur, etc…).

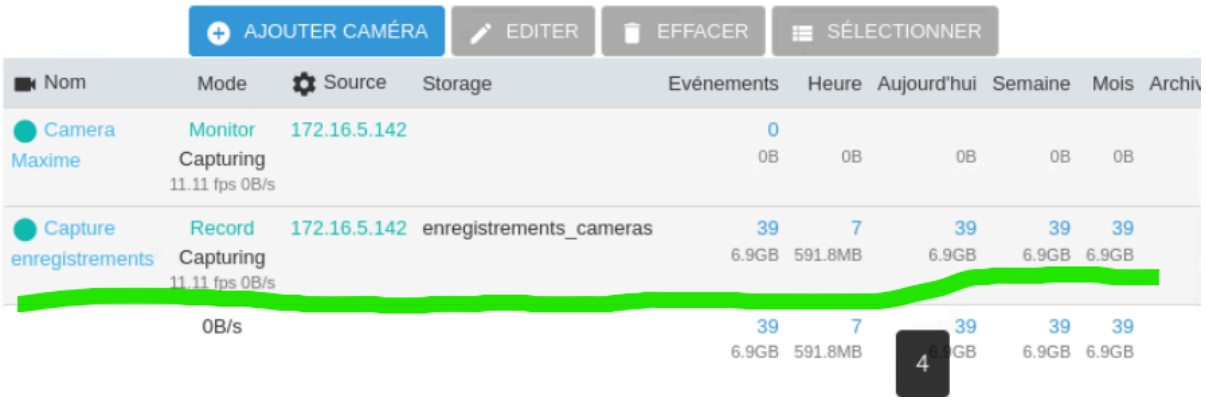


J’ai créé un élément de stockage qui fera comme un dossier partagé au niveau de la sauvegarde des enregistrements de la caméra qui a été mise en place sur ZoneMinder.



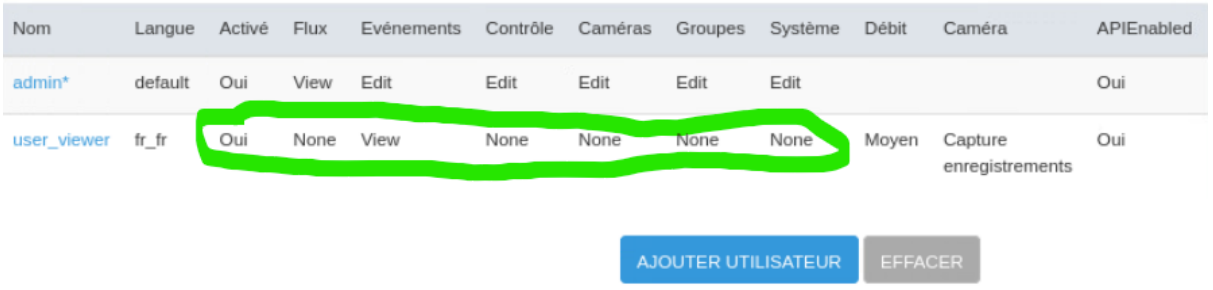




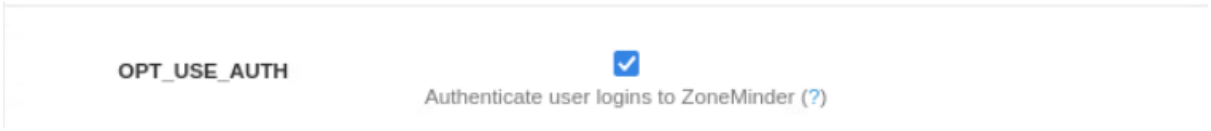


Ici, j'ai mis en place la même caméra (172.16.5.142 /16) avec un autre mode qui est "Record" ce qui va permettre d'enregistrer des séquences/vidéos de la caméra de surveillance et ces vidéos seront stockés sur le dossier partagé "enregistrements_cameras".

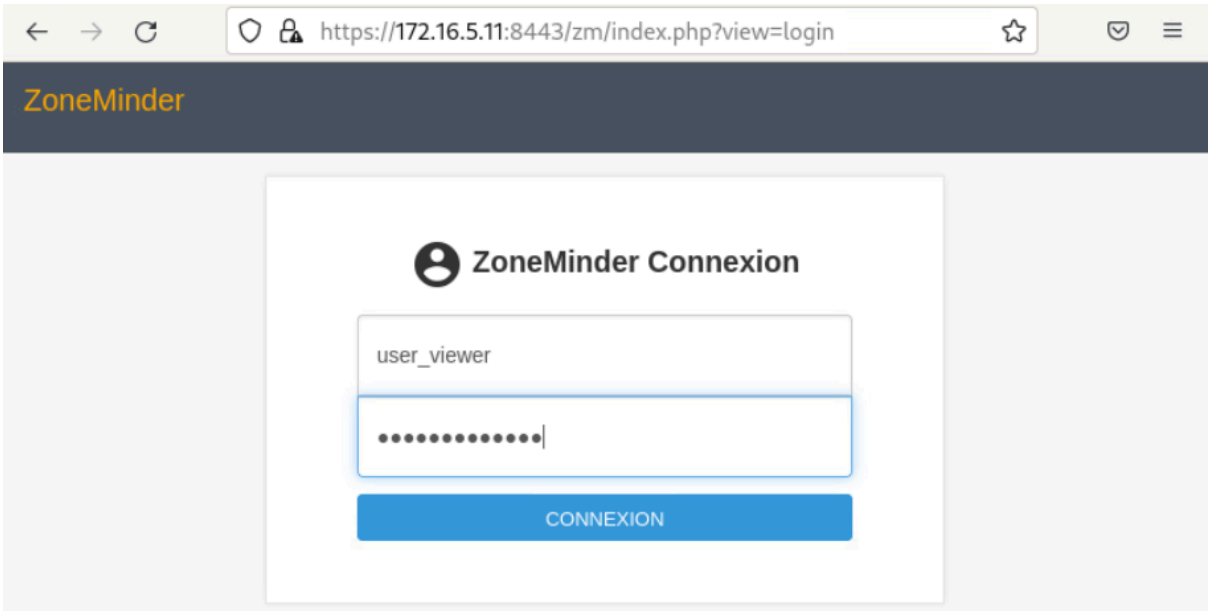
<u>Création de l'utilisateur ZoneMinder capable de visionner les enregistrements sans les modifier :</u>

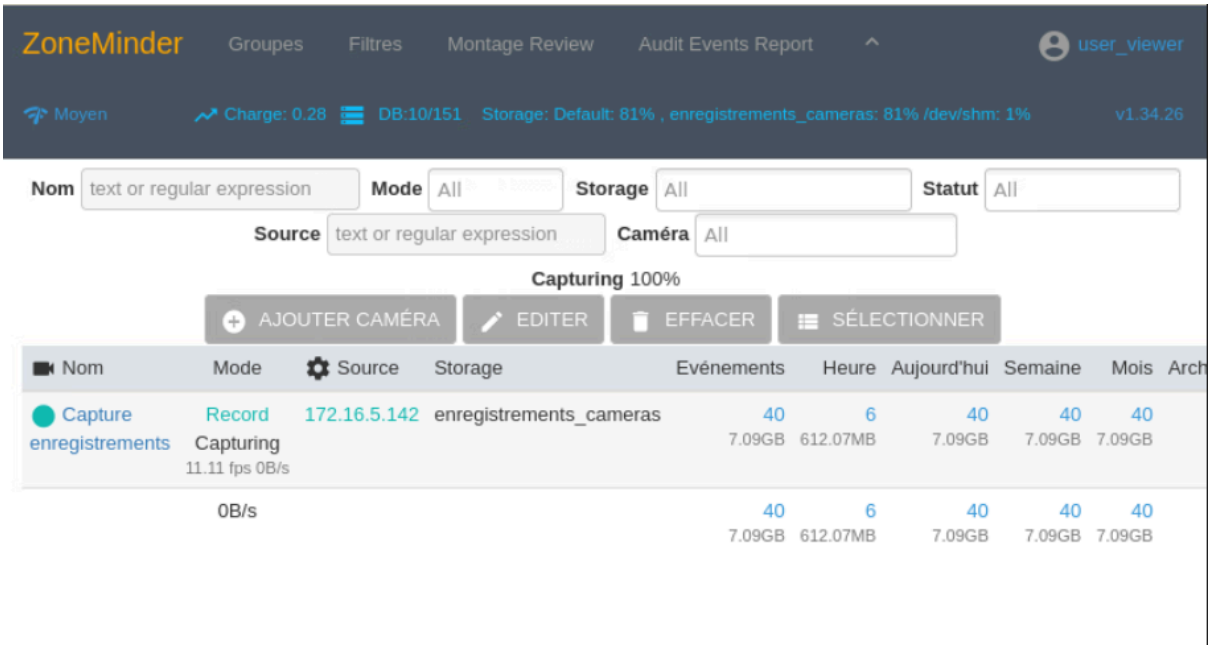


J'ai créé l'utilisateur "user_viewer" (avec un mot de passe : "Maxime91150#@") qui va avoir seulement la possibilité de regarder les différents enregistrements (dans ZoneMinder, ce sont des événements) sans pouvoir les modifier.

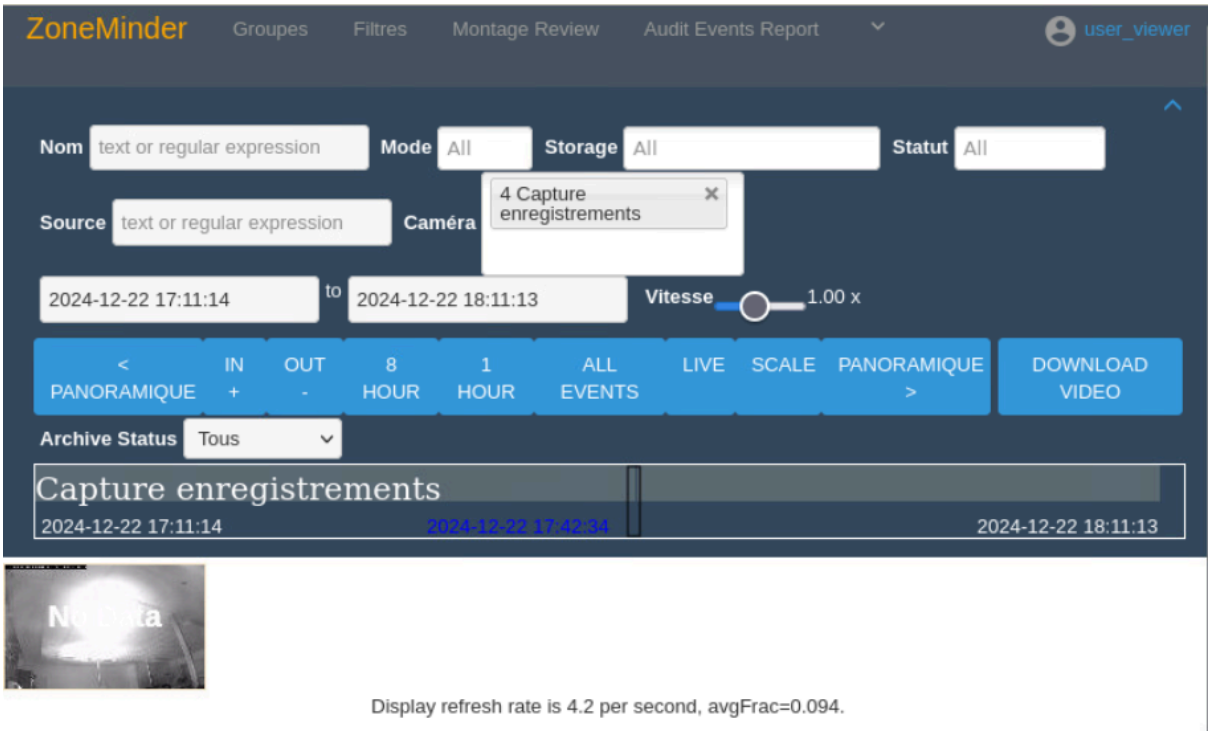


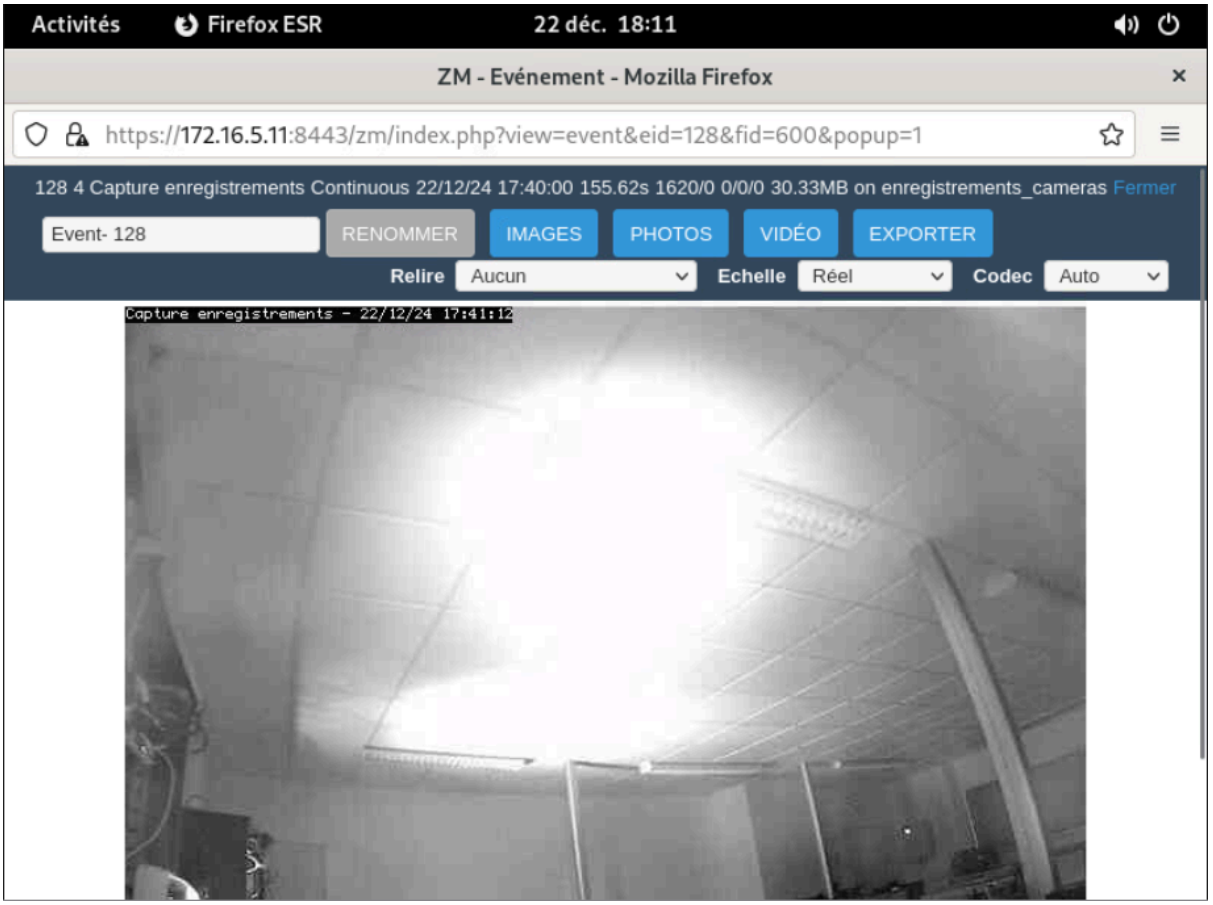
Pour pouvoir tester le compte ZoneMinder que j'ai configuré, il faut activer l'option "OPT_USE_AUTH" ce qui va permettre de pouvoir s'authentifier alors que si on désactive celle-ci alors on va accéder directement au compte administrateur de ZoneMinder.



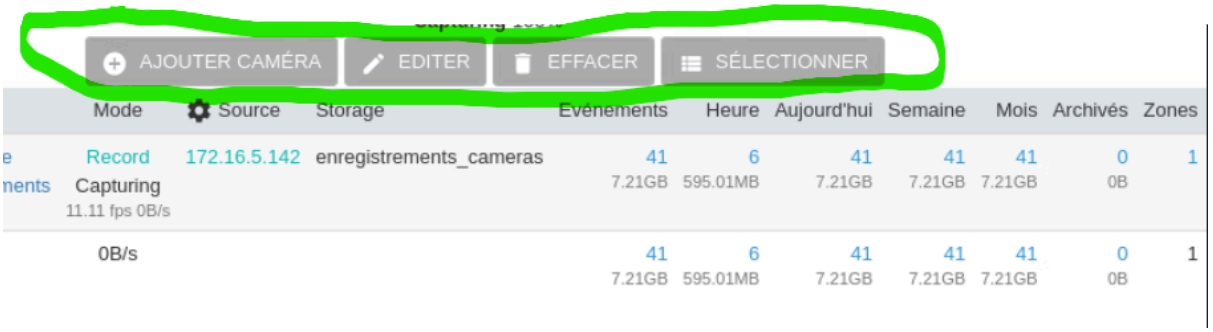


On accède bien au compte "user_viewer" et on voit le dossier partagé "enregistrements_cameras" avec "Capture enregistrements", donc il a bien été configuré.





On voit ici que l'utilisateur "user_viewer" peut regarder les enregistrements de la caméra mais il ne peut pas renommer les enregistrements. Il pourra télécharger ou exporter ces vidéos.



De plus, il ne pourra pas réaliser des modifications au niveau des caméras. En conclusion, l'utilisateur répond bien aux demandes.

Difficultés rencontrées :

Au début, j'avais du mal à mettre en place une adresse IP statique sur le serveur NAS Open Media Vault, j'ai trouvé une solution pour qu'elle ait une seule adresse IP en la configurant sur son interface web de configuration. Ensuite, la mise en place du conteneur Docker ZoneMinder et son accès via Internet, m'ont pris du temps au niveau des recherches. Enfin, la configuration du dossier partagé pour les vidéos de la caméra a été difficile, car je ne savais pas comment le mettre en place mais j'ai réussi à le configurer directement sur l'application ZoneMinder.

Diagramme des tâches

https://docs.google.com/spreadsheets/d/1jpq7P7dFgMUhC7Kj5G8OyVm50AEV9FM35pEdxjijXDU/edit?gid=1115838130#gid=1115838130

Conclusion

Pour conclure, ce TP m’a permis de mettre un serveur FTP avec différents accès pour de multiples utilisateurs (administrateur/utilisateur), un serveur GLPI avec le plugin FusionInventory et un serveur NAS Open Media Vault (gratuit) avec la configuration de conteneurs Docker comme Portainer et ZoneMinder. De plus, cela m’a permis de comprendre l’utilité de ces outils informatiques qui sont fondamentaux pour les entreprises en termes de protection et d’organisation et j’ai pû étoffer mon panel de compétences informatiques.