

Dispositifs de sécurité

Société menuimetal

Présenté par :

BAYERE Abdoul Fatahou
GUIHARD Mathieu

Sommaire

I. Introduction.....	1
II. Mise en place du Gantt.....	1
III. Mise à jour du schéma réseau.....	2
IV. Contexte.....	2
V. Connexion distante par SSH avec authentification par clef publique.....	3
A. Fonctionnement du service ssh sur le port 2222.....	4
1. Vérification de la prise en compte de l'option depuis le serveur.....	5
2. Scan du réseau avec l'outil nmap sous windows.....	5
B. Mise en place de l' authentification par clef publique pour se connecter en SSH depuis le client Windows.....	7
C. Test de connexion via les clés d'authentification avec le logiciel MobaXterm.....	16
D. Rediriger les logs d'authentification vers Rsyslog et les classer par machine source.....	21
VI. Connexion distante par VPN.....	23
A. Référencement dans le DNS.....	23
B. Référencement dans le GLPI.....	23
C. Référencement dans le Nagios.....	24
D. Installation et configuration de openVPN.....	24
E. Génération du certificat et de la clé d'autorité de certification.....	26
Création de l'autorité de certification :.....	26
F. Génération du certificat et de la clé pour le serveur VPN.....	27
G. Génération du certificat du serveur.....	28
H. Génération des certificats et clés pour les clients VPN.....	28
I. Génération du certificat du client.....	29
J. Génération des paramètres de Diffie-Hellman.....	30
K. Répartition des clés entre client et serveur.....	30
L. Sécurisation du serveur VPN.....	31
M. Le « forward » de paquets sur le serveur VPN.....	34
VII. Fail2BAN.....	34


Liens vers les ressources partagées :

Gestion VMs et VLANS :

 (GUIHARD_BAYERE) Gestion VLAN et VMs du contexte "Menui..."

Gantt :

 Diagramme de Gantt - AP11

 *Mathieu ayant été absent mardi lors des 4h de TP et jeudi lors des 4h de TP du matin, il ne s'est occupé que de la dernière partie (Fail2Ban).*

 *Abdoul a été absent mardi lors des 2h de TP.*

I. Introduction

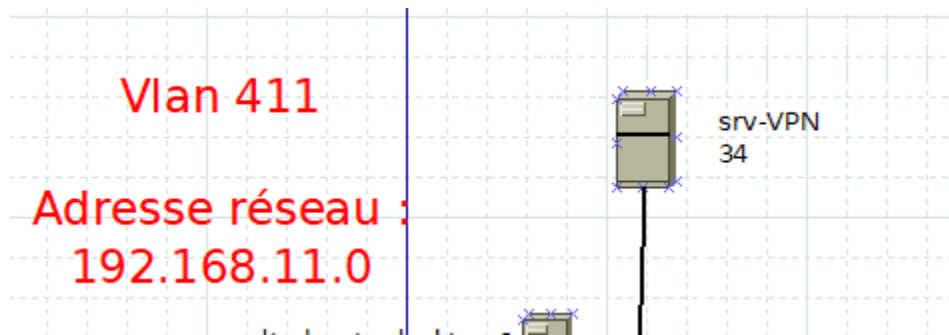
En 1980, Jean Morin crée Menuimetal.SA à Lens, une entreprise spécialisée dans la conception et la fabrication de structures en métal et en verre. Tout en se concentrant sur la production de huisseries et d'éléments de façade, Menuimetal délègue la pose à des partenaires externes. Avec un bureau d'étude capable de répondre aux besoins spécifiques de ses clients, l'entreprise propose des solutions sur mesure et poursuit sa croissance en cherchant à optimiser ses services informatiques.

II. Mise en place du Gantt

Tâches ou WBS					
Lettre	Titre	Jour et heure de début	Antécédent(s)	Durée en heure	Affectée à
A	Clonage et configuration de 3 serveurs de web Linux	19/11/2024 14:00:00		1	
B	Clonage et configuration de 1 cliente pour les tests	19/11/2024 14:15:00	A	1	
C	Vérification de l'accès SSH pour les VMs clonées.	19/11/2024 14:30:00	B	0,5	
D	Référencement des VMs clonées (serveurs et cliente) sur le DNS	19/11/2024 15:00:00	C	0,5	
E	Remonter les VMs sur GLPI et Nagios	19/11/2024 15:30:00	D	1	
F	Configuration de la VM Haproxy	21/11/2024 08:30:00	E	1,5	
G	Sauvegarde et Configuration de la VM MariaDB (maitre)	21/11/2024 10:00:00	F	2	
H	Sauvegarde et Configuration de la VM MariaDB (slave)	21/11/2024 12:00:00	G	2	
I	Tester la réplication et intégrer les logs dans rsyslog.	21/11/2024 13:30:00	H	0,5	
J	Configurer les sections frontend/backend dans haproxy.cfg.	21/11/2024 14:00:00	I	1	
K	Tester la continuité en arrêtant un serveur web.	21/11/2024 14:30:00	J	0,5	
L	Activer les statistiques et commenter les résultats.	21/11/2024 15:00:00	K	0,5	
M	Cloner et configurer le second serveur HaProxy	22/11/2024 13:30:00	L	1	
N	Installer et configurer Heartbeat	22/11/2024 14:30:00	M	1	
O	Tester la tolérance de panne en arrêtant un serveur HaProxy.	22/11/2024 15:30:00	N	1	
P	Documenter les configurations effectuées et les erreurs rencontrées	22/11/2024 16:00:00	O	0,5	
Q	Valider tous les tests	22/11/2024 16:30:00	P	0,5	

Visualisation du Gantt

III. Mise à jour du schéma réseau



Visualisation du schéma réseau mis à jour

IV. Contexte

Questions :

Quelle(s) est/sont la/les différence(s) entre SSH et VPN ?

Le **SSH** permet de se connecter de manière sécurisée à un serveur distant, en protégeant uniquement la connexion entre l'utilisateur et ce serveur. En revanche, le **VPN** crée un tunnel sécurisé pour l'ensemble du trafic internet de l'utilisateur. Il est utilisé pour protéger la navigation sur des réseaux publics, comme le Wi-Fi, et pour accéder à des ressources d'un réseau privé à distance, comme celles d'une entreprise. Alors que le SSH se limite à une connexion spécifique à un serveur, le VPN sécurise toutes les communications de l'appareil.

A partir de quelle source de données fonctionne l'application Fail2ban ?

L'application **Fail2ban** fonctionne en analysant les **fichiers de logs** des différents services et applications sur un serveur, comme SSH, FTP ou HTTP. Ces fichiers contiennent des informations sur les connexions effectuées, les erreurs d'authentification, ou les tentatives d'accès non

autorisées. Par exemple, il peut analyser le fichier **/var/log/auth.log** pour détecter des tentatives de connexion échouées via SSH. Fail2ban utilise des **filtres** qui recherchent des motifs spécifiques dans ces logs pour repérer des activités suspectes. Si un comportement anormal est détecté, comme plusieurs tentatives échouées d'affilée, Fail2ban bloque l'adresse IP concernée pour éviter toute attaque.

Quelle(s) est/sont la/les différence(s) entre IDS et IPS ?

Les systèmes **IDS** et **IPS** sont utilisés pour protéger les réseaux, mais ils fonctionnent différemment. Un **IDS** détecte les intrusions et surveille les activités suspectes, mais il ne fait rien pour empêcher l'attaque. Il se contente de **générer des alertes** pour informer les administrateurs. En revanche, un **IPS** ne se contente pas de détecter les attaques, il les **bloque** aussi en temps réel. Il est donc plus actif et réagit immédiatement pour empêcher que l'attaque ne cause des dégâts. Ainsi, un IDS est passif et sert principalement à alerter, tandis qu'un IPS est actif et protège le réseau en stoppant les intrusions.

V. Connexion distante par SSH avec authentification par clef publique

Question :

Quelle clef ne doit jamais être transférée ?

La clé qui ne doit jamais être transférée est la **clé privée**.


Argumentez sur le fait qu'une authentification par clef publique peut être un moyen de se prémunir des attaques dites de « Man In The Middle » mais n'est pas suffisante ?

L'authentification par clé publique est un moyen efficace pour limiter les attaques de type Man-In-The-Middle, car elle permet de vérifier l'identité des parties en utilisant des paires de **clés asymétriques**. Cependant, elle n'est pas suffisante à elle seule. Si la clé publique n'est pas échangée ou vérifiée dans un canal sécurisé, un attaquant peut fournir une fausse clé publique et usurper l'identité d'une des parties. De plus, l'utilisation de certificats auto-signés ou non validés par une autorité de confiance peut également rendre le système vulnérable. Par ailleurs, en cas de compromission de la clé privée, l'authentification est complètement compromise.

A. Fonctionnement du service ssh sur le port 2222

On ouvre le fichier de configuration et on recherche la ligne **Port 22**, que l'on remplacera par **Port 2222** :

```
root@srv-nagios:~# vim /etc/ssh/sshd_config
```



```
root@srv-nagios:~# service ssh restart
root@srv-nagios:~# service sshd restart
```

Vérification de la modification avec la commande **netstat** :

```
root@srv-nagios:~# netstat -tuln | grep 2222
tcp        0      0 0.0.0.0:2222        0.0.0.0:*          LISTEN
tcp6       0      0 :::2222            :::*                LISTEN
root@srv-nagios:~#
```

1. Vérification de la prise en compte de l'option depuis le serveur

Depuis le serveur GLPI on va se connecter sur le serveur nagios:

```
root@srv-glpi:~# ssh root@192.168.13.2
ssh: connect to host 192.168.13.2 port 22: Connection refused
root@srv-glpi:~#
```

Ici, on constate qu'il nous refuse bien la connexion, car il utilise par défaut le port 22.

Après cette modification lors de la connexion ssh on devra spécifier une option en plus :

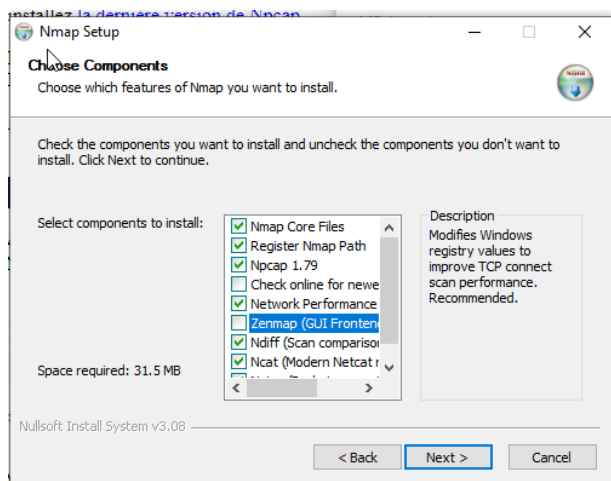
```
root@srv-glpi:~# ssh root@192.168.13.2 -p 2222
root@192.168.13.2's password:
Linux srv-nagios 6.1.0-25-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.106-3 (2024-08-26) x86_64

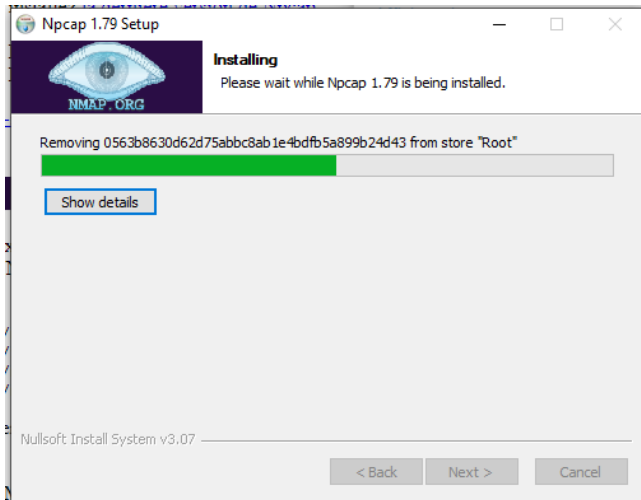
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Nov 26 14:14:24 2024 from 192.168.13.1
root@srv-nagios:~#
```

2. Scan du réseau avec l'outil nmap sous windows

Installation de nmap :





Vérification de la présence des serveur ssh avec nmap :

Pour le serveur nagios :

```
PS C:\Windows\system32> nmap -p 2222 192.168.13.2
Starting Nmap 7.95 ( https://nmap.org ) at 2024-11-27 11:59 Paris, Madrid
Nmap scan report for 192.168.13.2
Host is up (0.0019s latency).

PORT      STATE SERVICE
2222/tcp  open  EtherNetIP-1

Nmap done: 1 IP address (1 host up) scanned in 0.18 seconds
```

Pour le serveur GLPI :

```
PS C:\Windows\system32> nmap -p 2222 192.168.13.1
Starting Nmap 7.95 ( https://nmap.org ) at 2024-11-27 12:02 Paris, Madrid
Nmap scan report for 192.168.13.1
Host is up (0.0019s latency).

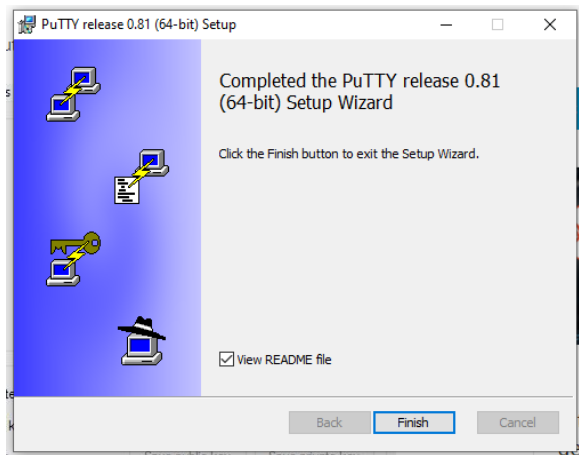
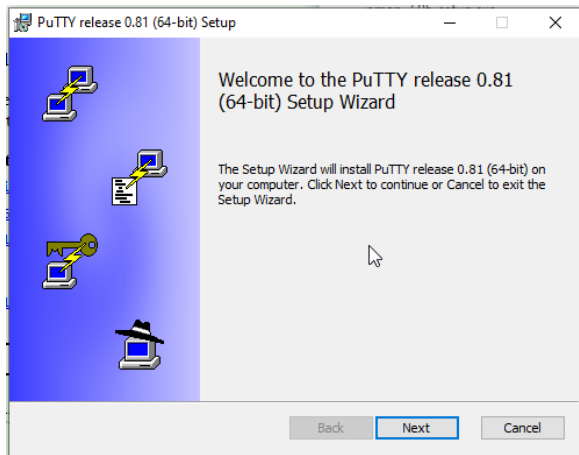
PORT      STATE SERVICE
2222/tcp  open  EtherNetIP-1

Nmap done: 1 IP address (1 host up) scanned in 0.17 seconds
```

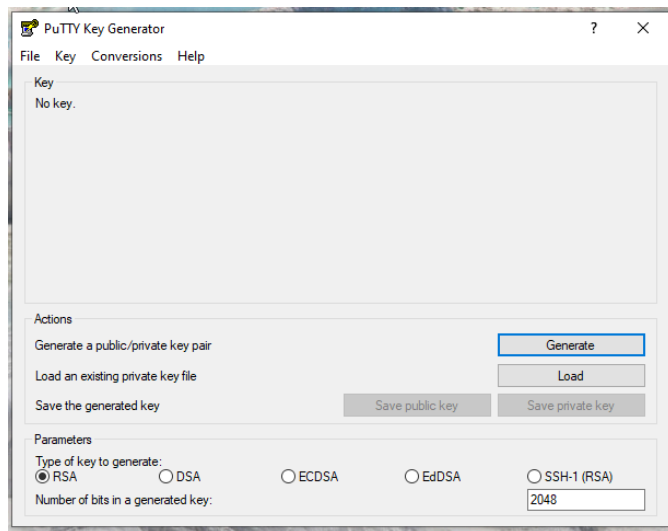
B. Mise en place de l' authentification par clef publique pour se connecter en SSH depuis le client Windows

Installation de PuTTY (PuTTYgen est installé en même temps que PuTTY)

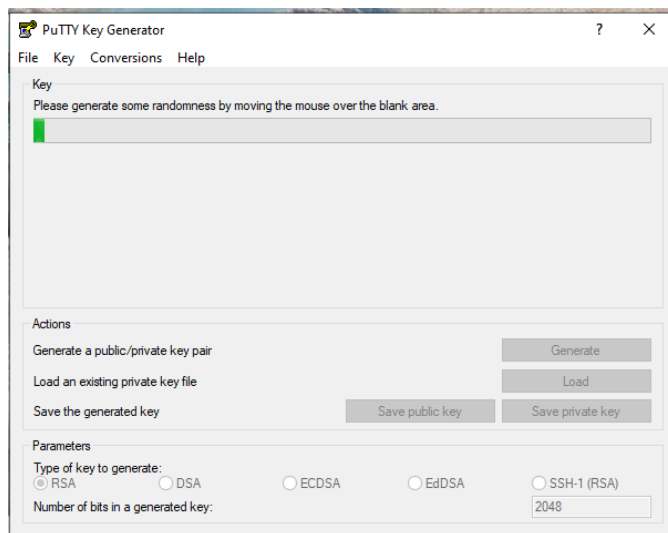
:

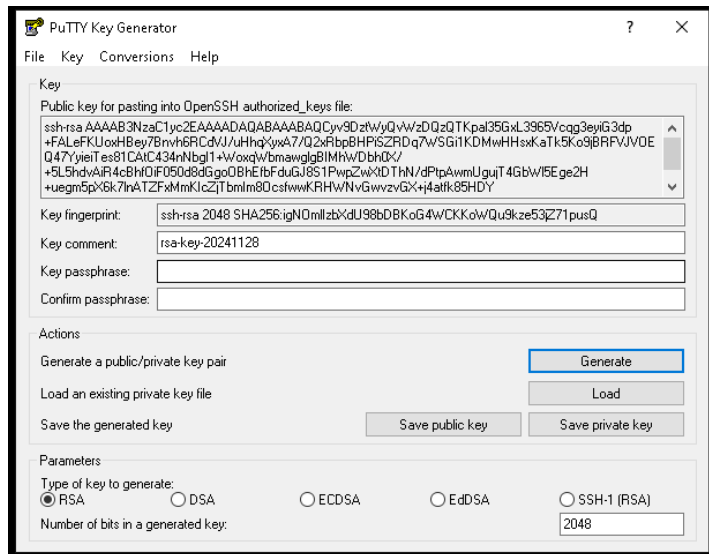


Ouverture de PuTTYgen :



Création du couple clé publique / privé :

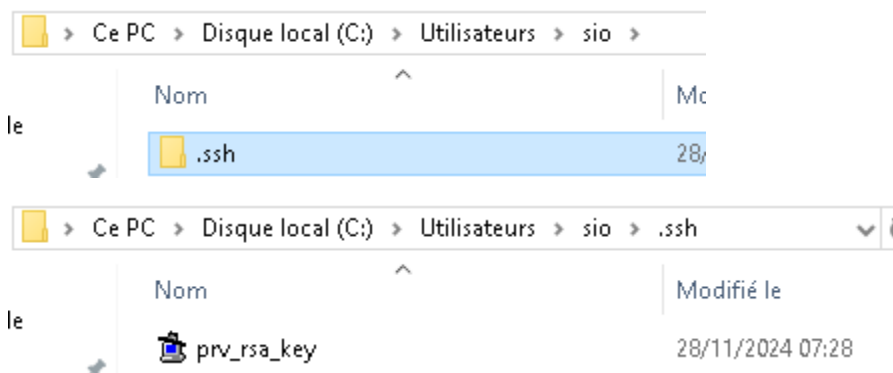




Une fois les clés générées, nous allons sauvegarder la clé privée sur le client Windows et sauvegarder les clés publiques sur les serveurs auxquels nous voulons nous connecter.

Sauvegarde de la clé privé :

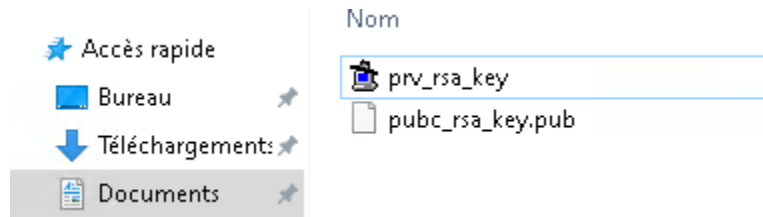
On va créer le répertoire qui va stocker la clé privé :



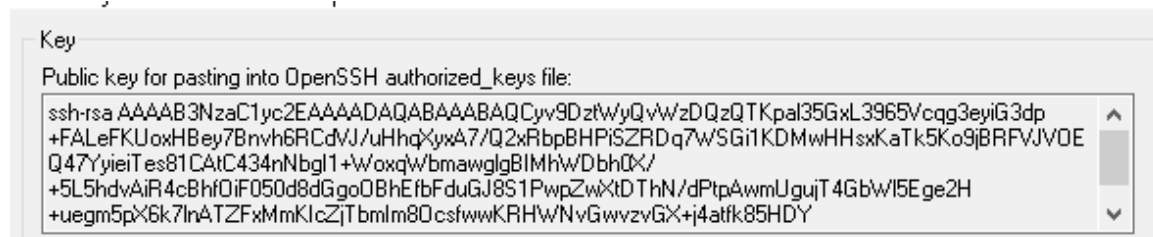
Pour des raisons de sécurité on aurait dû sauvegarder la clé dans un dossier crypté, le stocker en lieu sûr.

Transfert de la clé publique vers les serveur :

Sauvegarde de la clé publique nommé pubc_rsa_key.pub :



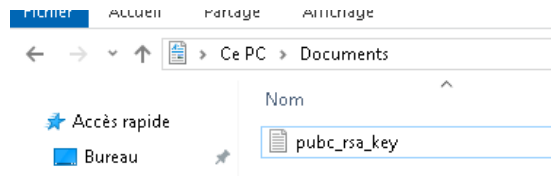
La clé pubc_rsa_key.pub est une clé publique qui ne sera pas compatible avec OpenSSH la clé compatible se trouvera ci dessus :



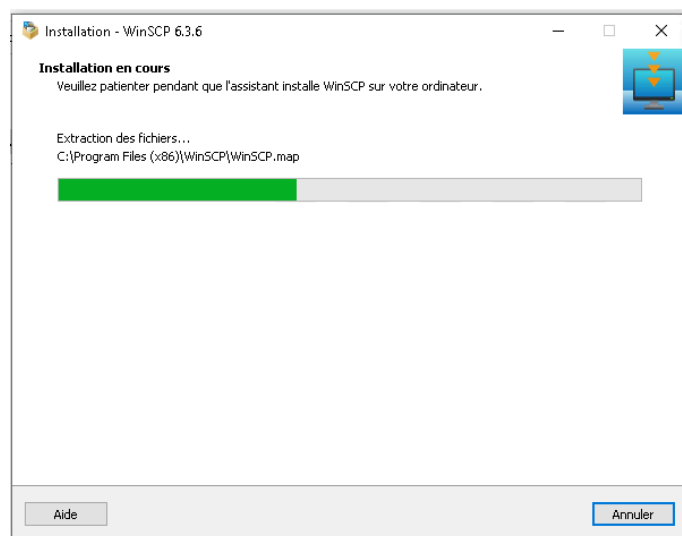
ssh-rsa

```
AAAAB3NzaC1yc2EAAAADAQABAAQBAQCyv9DztWyQvWzDQzQTKpal35GxL3965Vcqg3eyiG3dp+
FALeFKUoxHBey7BnvH6RCdVJ/uHhqXyx7/Q2xRbpBHPiSZRDq7WSGi1KDMwHHsxKaTk5Ko9jBRFVJVOE
Q47YyieiTes81CAtC434nNbgl1+WoxqWbmawglgBIMhWDbh0X/+5L5hdvAiR4cBhfOiF050d8dGgoOBhEfbFduGJ8S1PwpZwXtDThN/dPtpAwmUgujT4GbWI5Eg
e2H+uegm5pX6k7lnATZFxmMklcZjTbmIm8OcsfwwKRHWNVGwvzvGX+j4atfk85HDY+Sc7ThJWsxj9Qpt/vdOBED34hcBmv rsa-key-20241128
```

Ce que nous allons faire, c'est stocker la clé dans un fichier texte afin de la transférer à l'aide de WinSCP vers le serveur cible :



Installation de winscp :



Installation de openssh-sftp-server sur les serveur nagios et glpi:

```
root@srv-nagios:~# apt install openssh-sftp-server  
Lecture des listes de paquets... Fait
```

SFTP est un **protocole de transfert de fichiers sécurisé** basé sur SSH. Il vous permet de transférer des fichiers de manière sécurisée entre votre machine locale et un serveur distant

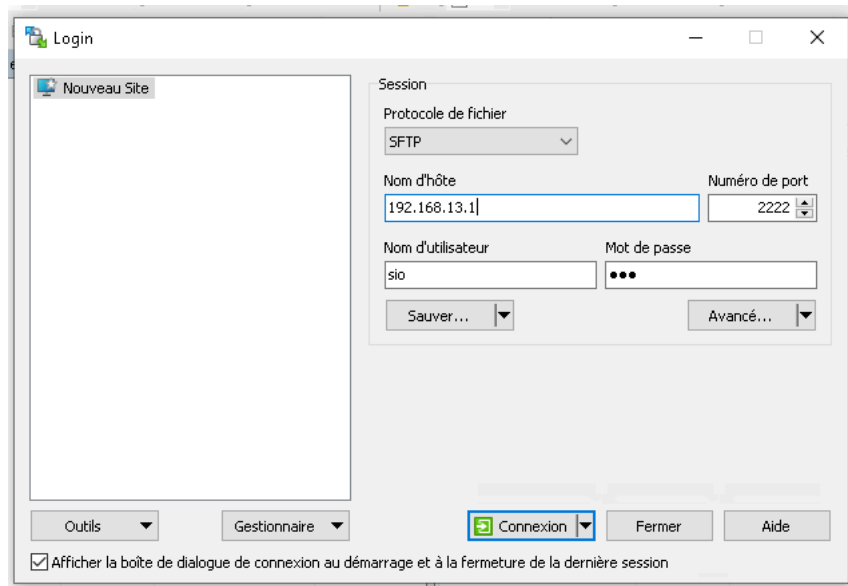
Vérification de la présence d'openssh-sftp-server :

```
root@srv-nagios:~# apt list --installed | grep openssh-sftp  
  
WARNING: apt does not have a stable CLI interface. Use with caution in scripts.  
  
openssh-sftp-server/stable,stable-security,now 1:9.2p1-2+deb12u3 amd64 [installé]  
root@srv-nagios:~#
```

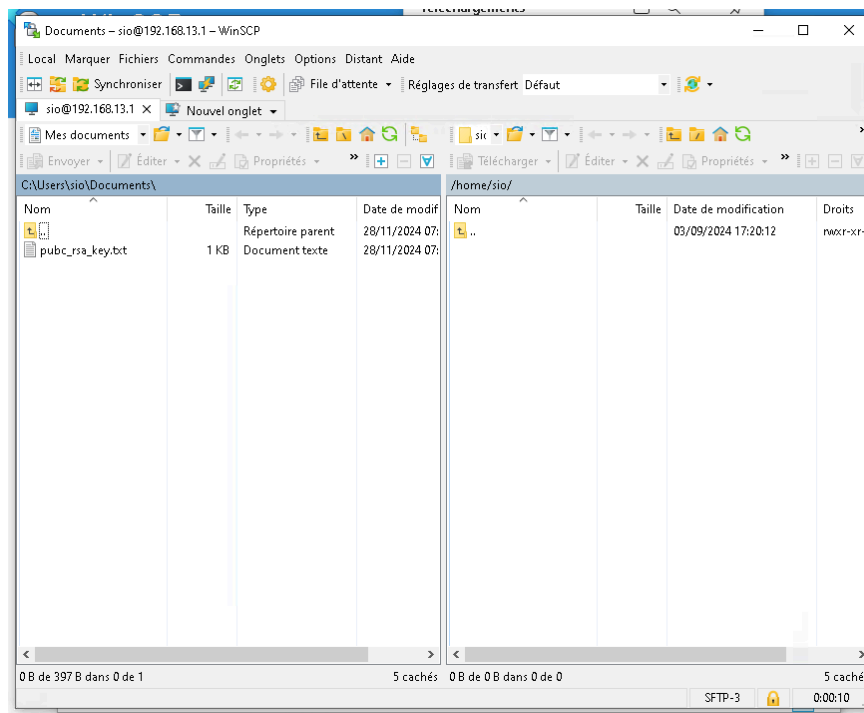
```
root@srv-glpi:~# apt list --installed | grep openssh-sftp  
  
WARNING: apt does not have a stable CLI interface. Use with caution in scripts.  
  
openssh-sftp-server/stable,stable-security,now 1:9.2p1-2+deb12u3 amd64 [installé]  
root@srv-glpi:~#
```

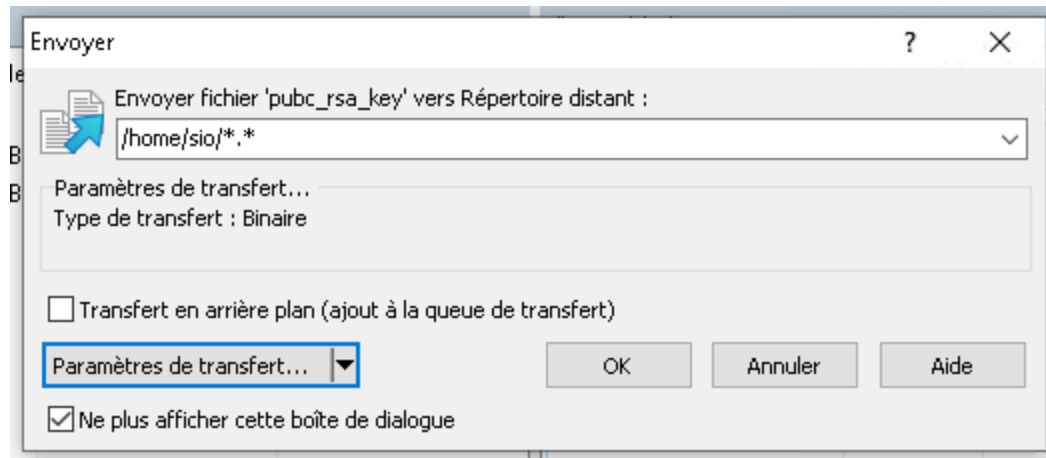
On a installé le serveur SFTP pour permettre le transfert de fichiers sécurisés entre mon ordinateur et le serveur distant, en utilisant le protocole SFTP qui repose sur SSH pour garantir la confidentialité et l'intégrité des données échangées.

Envoi de la clé publique avec winSCP :

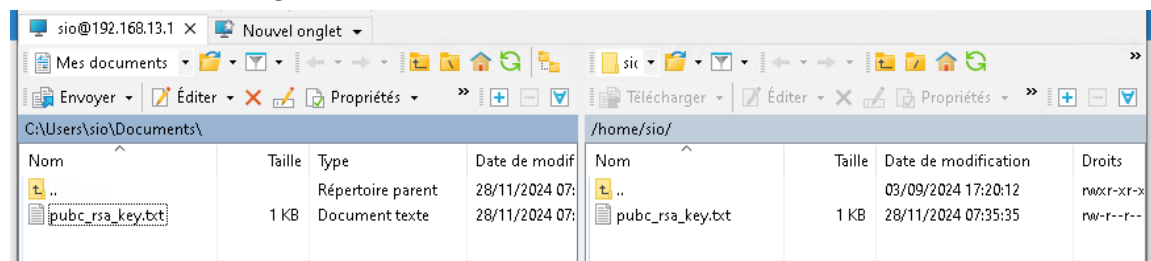


Une fois connecté on transfère la clé publique :

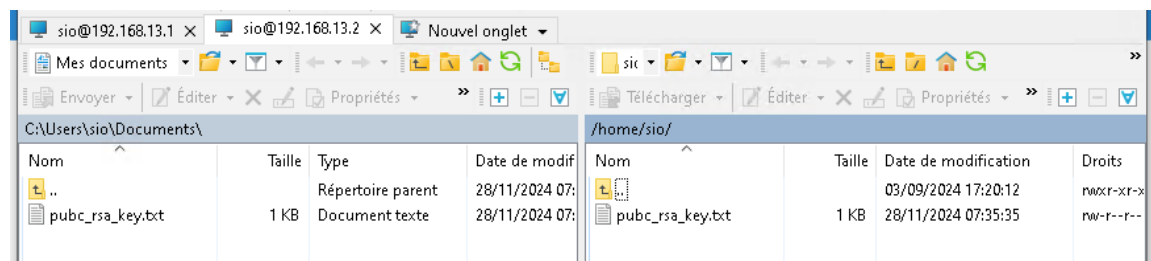




sur le serveur Nagios :



sur le serveur glpi :



Vérification que le transfert a bien été effectué :

```
sio@srv-glpi:~$ ls
pubc_rsa_key.txt
```

```
sio@srv-nagios:~$ ls
pubc_rsa_key.txt
```

Une fois le transfert terminé on va passé du côté serveur pour effectuer les modification nécessaire pour l'authentification en SSH :

Configuration de la clé sur le serveur ssh :

Attention la clé publique doit se trouver dans l'utilisateur et non dans root

On va effectuer les manipulation suivante en étant connecté en sio :

```
$ mkdir -p ~/.ssh
:~$ cat ~/pubc_rsa_key.txt >> ~/.ssh/authorized_keys

~$ chmod 700 ~/.ssh
~$ chmod 600 ~/.ssh/authorized_keys
~$
```

Pour des raison de sécurité une fois l'ajout de la clé on va la supprimer :

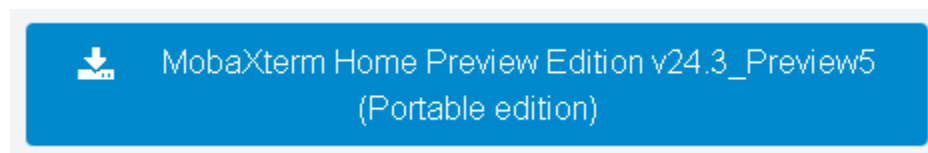
```
~$ rm ~/pubc_rsa_key.txt
```

Vérification de la présence de la clé :

```
sio@srv-glpi:~$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQCyv9DztWyQvWzDQzQTKpal35GxL3965Vcgg3eyiG3dp+FALeFKUoxHBey7Bnvh
6RCdVJ/uHhqYxyA7/Q2xRbpBHPiSZRDq7WSG1lKDMwHHsxKaTk5Ko9jBRFVJV0EQ47YyieiTes81CatC434nNbgI1+WoxqWbmawg
lgBIMhWDbh0X/+5L5hdvA1R4cBhf0iF050d8dGgo0BhEfbFduGJ8S1PwpZwXtDThN/dPtpAwuUgujT4GbWl5Ege2H+uegm5pX6k7
lnATZFxmMkIcZjTbmIm80csfwKRHWNVGwvzvGX+j4atfk85HDY+Sc7ThJwsxj9Qpt/vd0BED34hcBmv rsa-key-20241128sio
@srv-glpi:~$
```

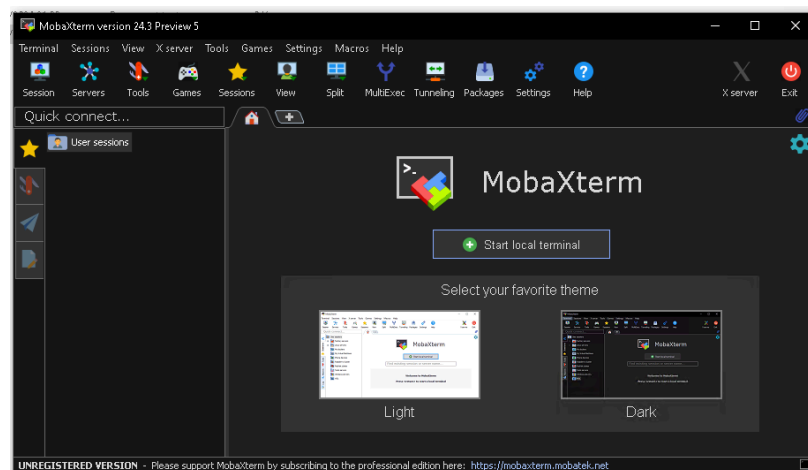
C. Test de connexion via les clés d'authentification avec le logiciel MobaXterm

Installation de MobaXterm :

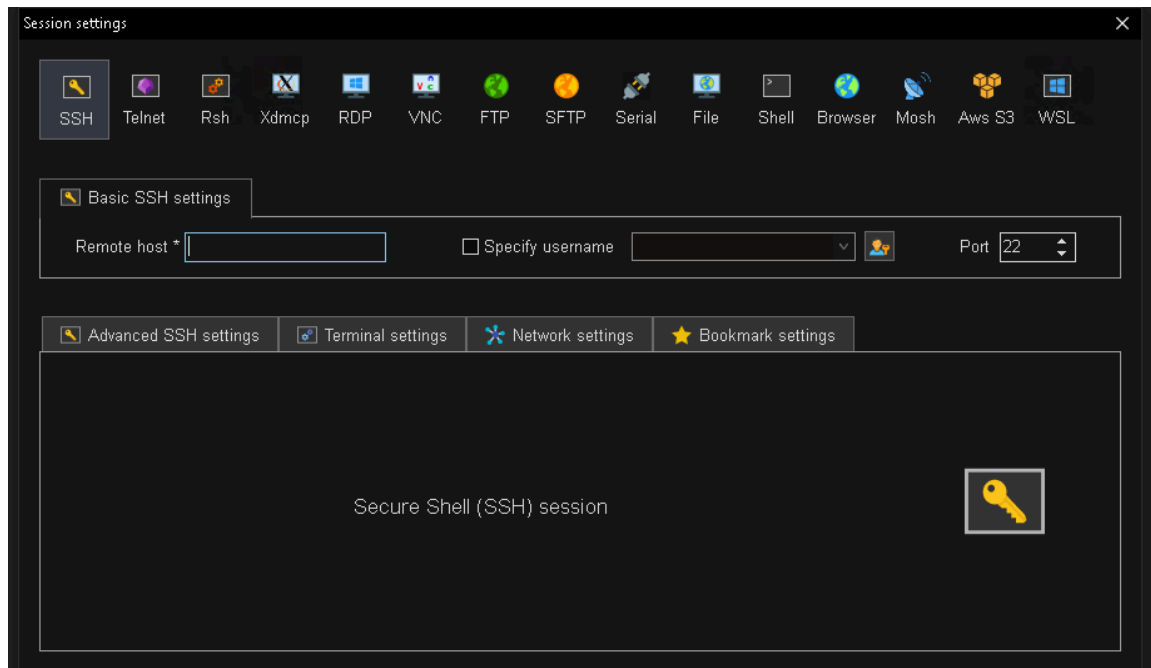


Ici, nous allons installer l'édition portable car, contrairement à l'édition classique, nous n'avons pas besoin de l'installer sur le système et nous pouvons la lancer directement.

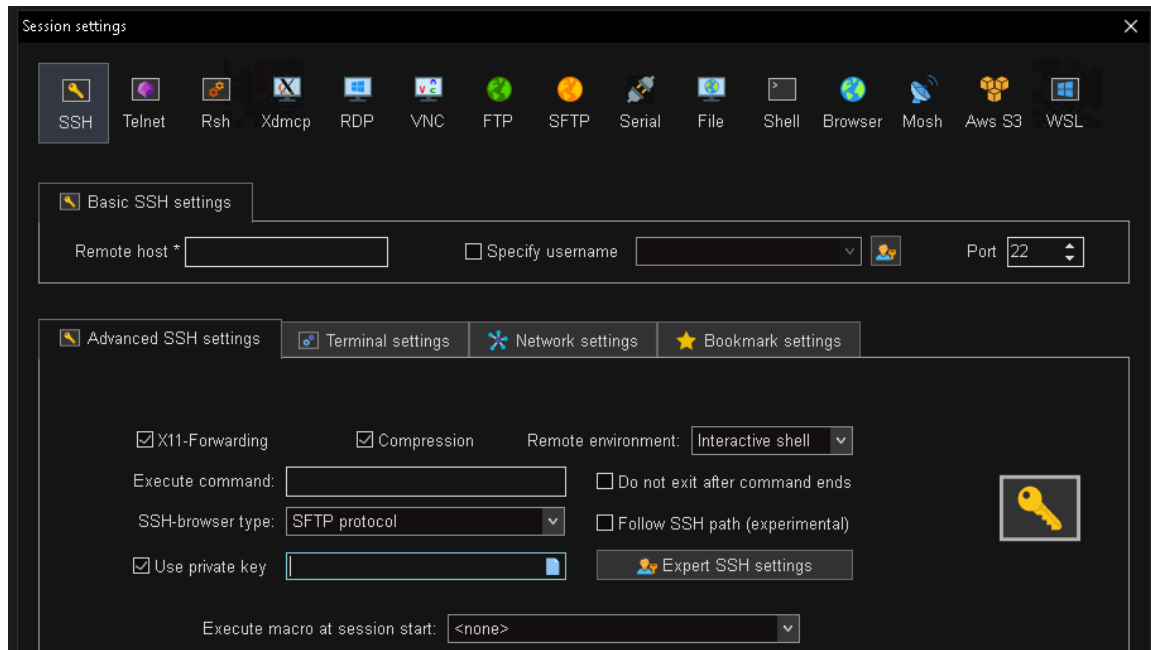
Ce PC > Téléchargements > MobaXterm_Portable_v24.3_Preview5				
	Nom	Modifié le	Type	Taille
de gements nts	CygUtils.plugin	27/11/2024 21:32	Fichier PLUGIN	17 748 Ko
	CygUtils64.plugin	27/11/2024 21:32	Fichier PLUGIN	11 723 Ko
	MobaXterm_Personal_24.3_Preview5	27/11/2024 21:32	Application	16 639 Ko

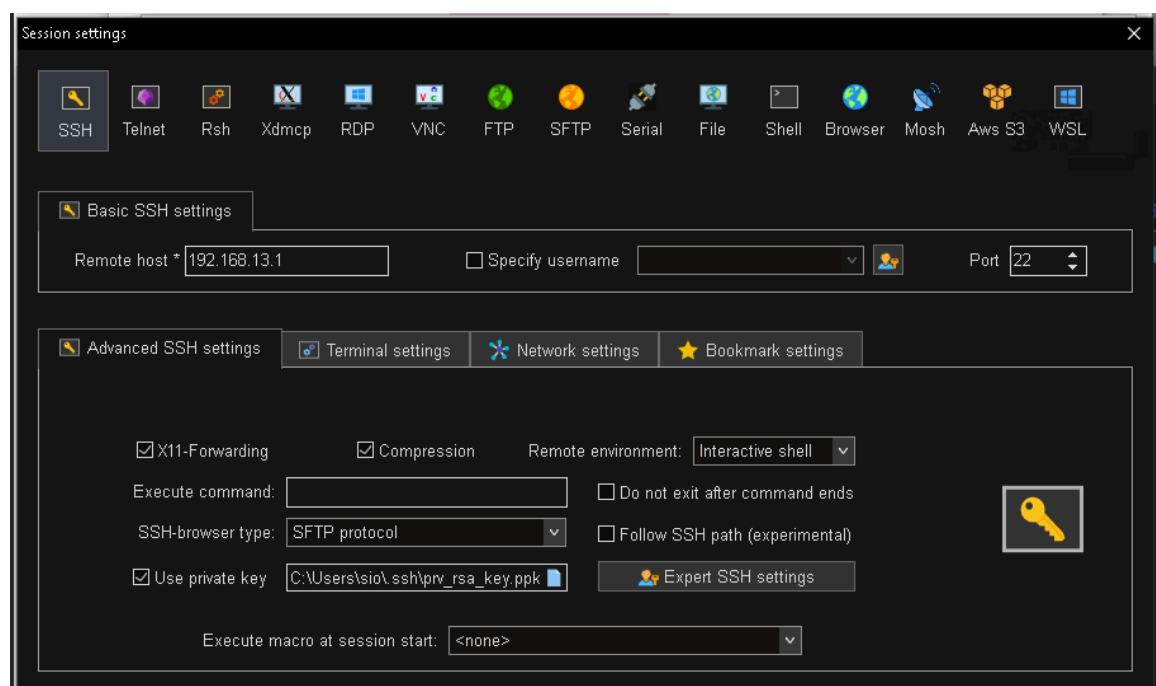
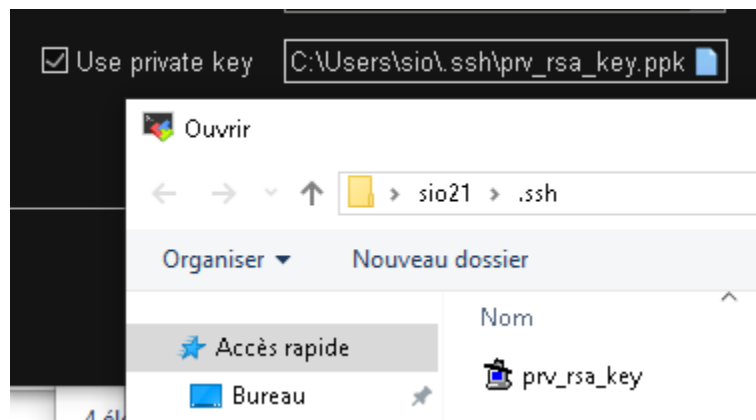


Configuration de la connexion SSH avec authentification par clé :



Ajout de la clé privée :





En amont sur le serveur nagios et glpi on va autoriser que les connexion par clé en éditant le fichier /etc/ssh/sshd_config :

```
PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2

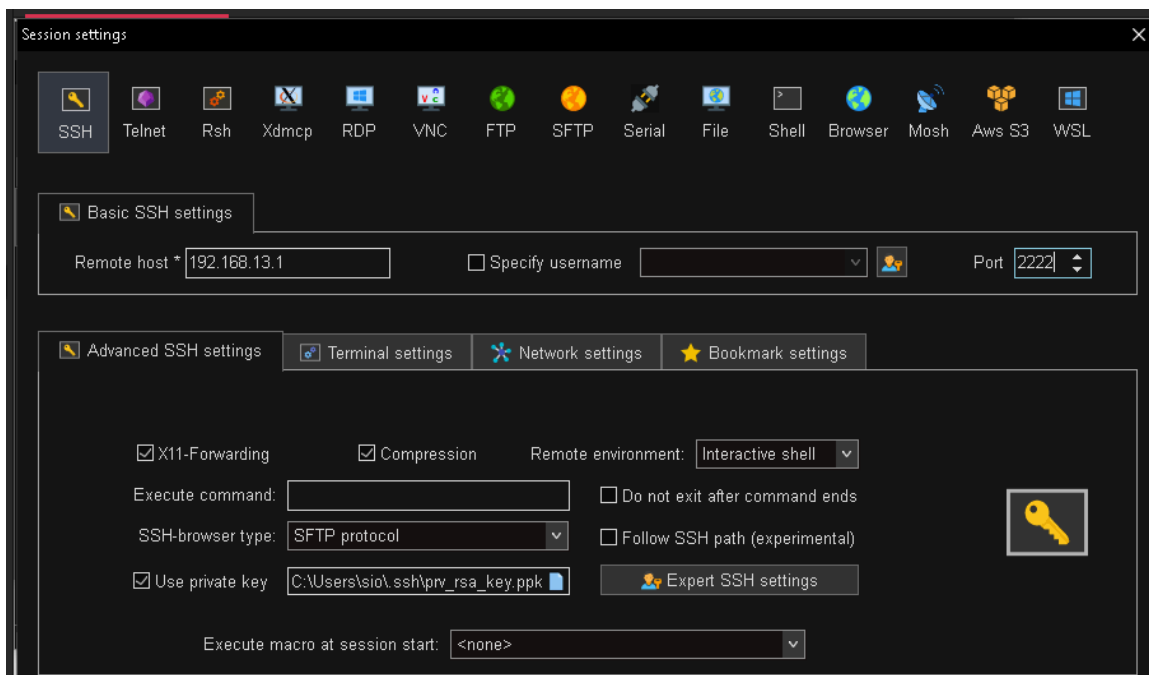
#AuthorizedPrincipalsFile none

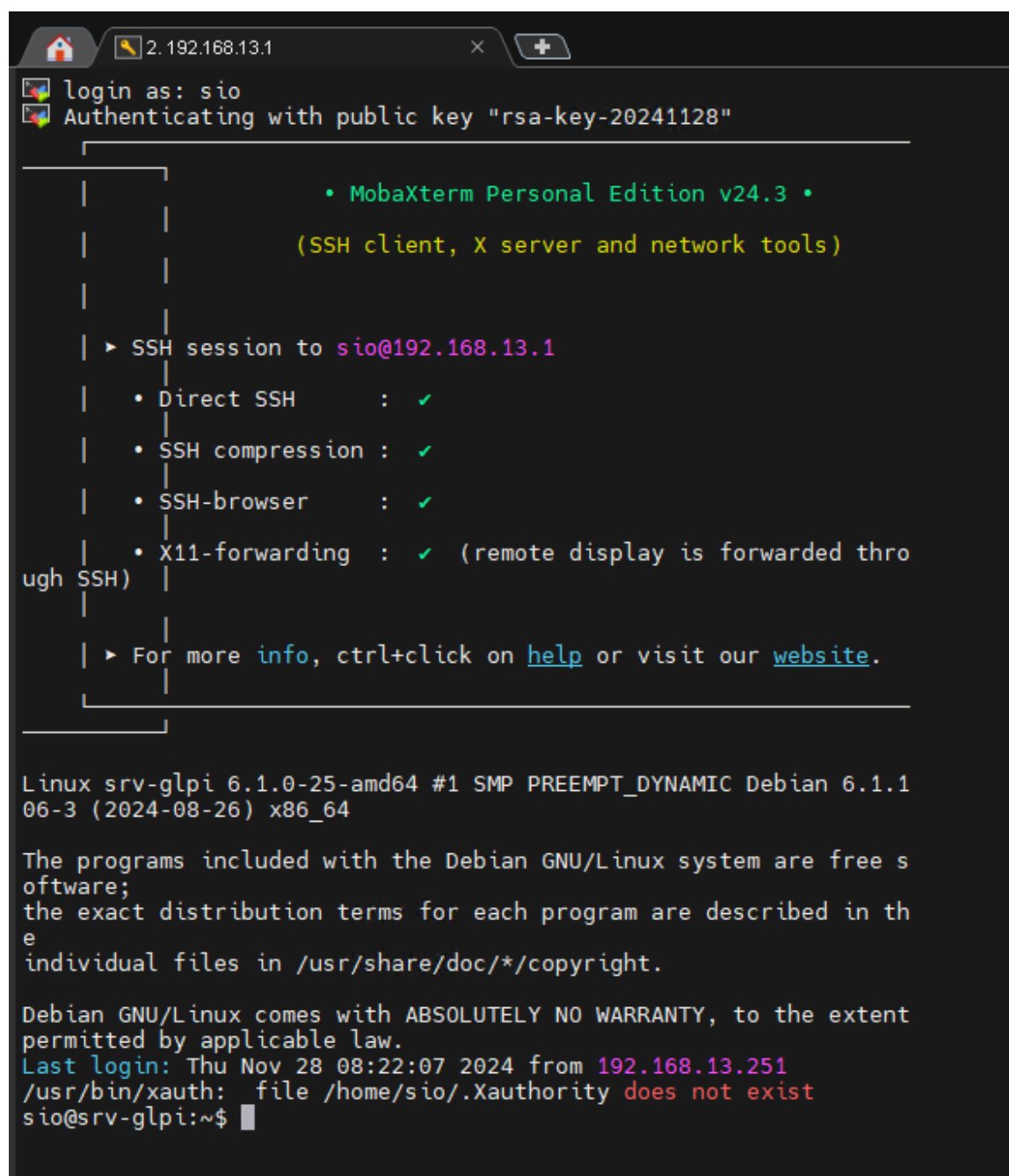
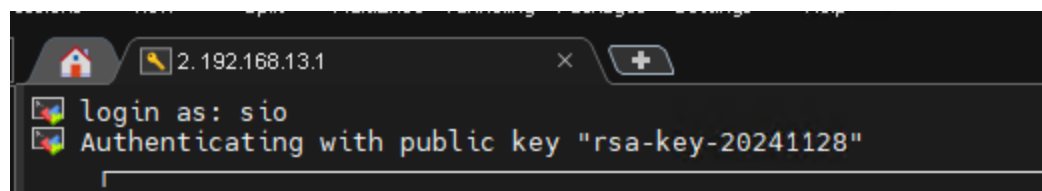
#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no
# Change to yes if you don't trust ~/.ssh/known_hosts for
# HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts and ~/.shosts files
#IgnoreRhosts yes

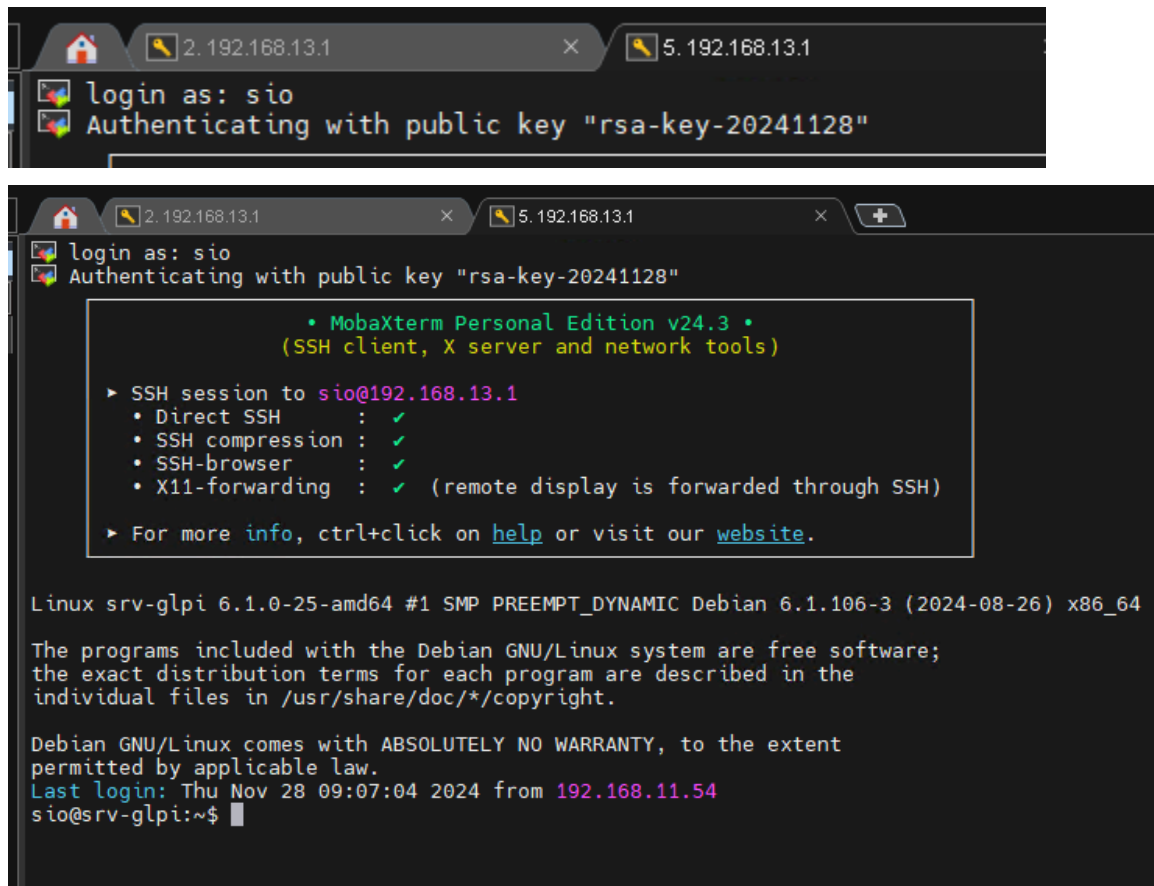
# To disable tunneled clear text passwords, change to no here!
PasswordAuthentication no
#PermitEmptyPasswords no
```

Connexion vers le serveur nagios :





Pour le serveur nagios :



The image shows two screenshots of a MobaXterm terminal window. The top screenshot shows the login process for user 'sio' on host 2.192.168.13.1, authenticating with a public key 'rsa-key-20241128'. The bottom screenshot shows the terminal after login, displaying the MobaXterm version (v24.3) and session details. A list of session options is shown: Direct SSH, SSH compression, SSH-browser, and X11-forwarding, all marked as successful. Below this, the terminal shows the Linux version (Debian 6.1.106-3) and the last login time (Thu Nov 28 09:07:04 2024).

```
login as: sio
Authenticating with public key "rsa-key-20241128"

• MobaXterm Personal Edition v24.3 •
  (SSH client, X server and network tools)

▶ SSH session to sio@192.168.13.1
  • Direct SSH : ✓
  • SSH compression : ✓
  • SSH-browser : ✓
  • X11-forwarding : ✓ (remote display is forwarded through SSH)

▶ For more info, ctrl+click on help or visit our website.

Linux srv-glpi 6.1.0-25-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.106-3 (2024-08-26) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Nov 28 09:07:04 2024 from 192.168.11.54
sio@srv-glpi:~$
```

D. Rediriger les logs d'authentification vers Rsyslog et les classer par machine source

Nous allons devoir configurer le serveur rsyslog dans le fichier **rsyslog.conf** et y placer ces lignes :

```
$template syslog, "/var/log/clients/%fromhost%/syslog.log"
*. * ?syslog
```


Du côté du serveur Nagios et GLPI, nous allons installer rsyslog et ajouter la ligne suivante :

```
auth.* @192.168.11.23:514
#*. * @192.168.11.23:514
```

Les logs seront triés dans des dossiers en fonction de leur IP, contenant les fichiers syslog :

```
root@srv-rsyslog-debian:~# ls /var/log/clients/ | grep 192.168.13
192.168.13.1
192.168.13.2
```

```
root@srv-rsyslog-debian:~# vim /var/log/clients/192.168.13.1/syslog.log
```

```
2024-11-28T09:54:25+01:00 srv-glpi systemd-logind[438]: New session 1717 of user sio.
2024-11-28T09:54:29+01:00 srv-glpi su[76666]: (to root) sio on pts/0
2024-11-28T10:22:31+01:00 srv-glpi sshd[76662]: Received disconnect from 192.168.13.251 port 50608:11: Bye Bye
2024-11-28T10:22:31+01:00 srv-glpi sshd[76662]: Disconnected from user sio 192.168.13.251 port 50608
2024-11-28T10:22:31+01:00 srv-glpi systemd-logind[438]: Session 1717 logged out. Waiting for processes to exit.
2024-11-28T10:22:31+01:00 srv-glpi systemd-logind[438]: Removed session 1717.
2024-11-28T10:22:37+01:00 srv-glpi sshd[76745]: Received disconnect from 192.168.13.251 port 43200:11: Bye Bye [preauth]
2024-11-28T10:22:37+01:00 srv-glpi sshd[76745]: Disconnected from authenticating user sio 192.168.13.251 port 43200 [preauth]
2024-11-28T10:22:37+01:00 srv-glpi sshd[76747]: Accepted password for sio from 192.168.13.251 port 34120 ssh2
2024-11-28T10:22:37+01:00 srv-glpi systemd-logind[438]: New session 1720 of user sio.
2024-11-28T10:22:40+01:00 srv-glpi su[76757]: (to root) sio on pts/0
```

Ici, on voit que les logs auth ont été redirigés vers rsyslog et sont stockés en fonction de leur IP source.

VI. Connexion distante par VPN

Clonage de notre srv-VPN :

Configuration du nom et de l'ip :

```
sio@srv-VPN:~$ ip a | grep 192.168
    inet 192.168.11.34/24 brd 192.168.11.255 scope global ens18
sio@srv-VPN:~$
```

A. Référencement dans le DNS

```
mariaDB-slave IN A 192.168.11.30
srv-VPN IN      A 192.168.11.34
~
```

test de connexion avec la commande nslookup :

```
root@dns:~# nslookup
> srv-VPN
Server:      192.168.12.1
Address:     192.168.12.1#53

Name:   srv-VPN.menuimetal.fr
Address: 192.168.11.34
> _
```

B. Référencement dans le GLPI



Placement de la machine dans la salle serveur :



C. Référencement dans le Nagios

Ajout de la machine :

srv-OMV		UP	11-28-2024 11:50:10	43d 20h 8m 9s	PING OK - Paquets perdus = 0%, RTA = 1.16 ms
---------	--	----	---------------------	---------------	--

ajout de quelque services :

srv-VPN	Check SSH	OK	11-28-2024 11:59:41
	Check Users	OK	11-28-2024 12:00:31
	Local Disk	OK	11-28-2024 12:00:58
	PING	OK	11-28-2024 12:01:16
	Total Process	OK	11-28-2024 12:00:50

D. Installation et configuration de openVPN

Installation du paquet :

apt install openvpn

Vérification de la présence d'openssl :

```
root@srv-VPN:~# openssl version
OpenSSL 3.0.14 4 Jun 2024 (Library: OpenSSL 3.0.14 4 Jun 2024)
root@srv-VPN:~#
```

Copiez le dossier `/usr/share/easy-rsa/` dans `/etc/openvpn/` :

```
cp -r /usr/share/easy-rsa/ /etc/openvpn/
```

Vérification que le dossier soit copié :

```
root@srv-VPN:~# cp -r /usr/share/easy-rsa/ /etc/openvpn/
root@srv-VPN:~# ls -l /etc/openvpn/
total 16
drwxr-xr-x 2 root root 4096 11 nov. 2023 client
drwxr-xr-x 3 root root 4096 28 nov. 13:38 easy-rsa
drwxr-xr-x 2 root root 4096 11 nov. 2023 server
-rwxr-xr-x 1 root root 1468 11 nov. 2023 update-resolv-conf
root@srv-VPN:~#
```

Initialisation de la PKI :

```
root@srv-VPN:/etc/openvpn/easy-rsa# ./easyrsa init-pki
```

```
root@srv-VPN:/etc/openvpn/easy-rsa# ./easyrsa init-pki
* Notice:

init-pki complete; you may now create a CA or requests.

Your newly created PKI dir is:
* /etc/openvpn/easy-rsa/pki

* Notice:
IMPORTANT: Easy-RSA 'vars' file has now been moved to your PKI above.
root@srv-VPN:/etc/openvpn/easy-rsa#
```

Modifiez le fichier `/etc/openvpn/easy-rsa/pki/vars` qui servira au script `easyrsa` :

```
set_var EASYRSA "${0%/*}"
```

```
set_var EASYRSA_PKI "$PWD/pki"
```

```
set_var EASYRSA_DN "org"
```

```
set_var EASYRSA_REQ_COUNTRY    "FR"  
set_var EASYRSA_REQ_PROVINCE   "IDF"  
set_var EASYRSA_REQ_CITY       "MELUN"  
set_var EASYRSA_REQ_ORG        "menuimetal"  
set_var EASYRSA_REQ_EMAIL      "toto@menuimetal.fr"  
set_var EASYRSA_REQ_OU         "SIO"
```

```
set_var EASYRSA_ALGO           rsa
```

```
set_var EASYRSA_CA_EXPIRE      3650
```

```
set_var EASYRSA_CERT_EXPIRE    825
```

```
#  
set_var EASYRSA_EXT_DIR "$EASYRSA/x509-types"
```

```
set_var EASYRSA_REQ_CN         "srv-VPN"
```

```
set_var EASYRSA_DIGEST         "sha256"
```

E. Génération du certificat et de la clé d'autorité de certification

Création de l'autorité de certification :

```
root@srv-VPN:/etc/openvpn/easy-rsa# ./easyrsa build-ca nopass
```

```
* Notice:
```

```
CA creation complete and you may now import and sign cert requests.  
Your new CA certificate file for publishing is at:  
/etc/openvpn/easy-rsa/pki/ca.crt
```

F. Génération du certificat et de la clé pour le serveur VPN

```
pki/issued/SrvVPN.cer: OK  
root@srv-VPN:/etc/openssl/easy-rsa# ./easyrsa gen-req SrvVPN
```

```
-----  
Country Name (2 letter code) [FR]:  
State or Province Name (full name) [IDF]:  
Locality Name (eg, city) [MELUN]:  
Organization Name (eg, company) [menuimetal]:  
Organizational Unit Name (eg, section) [SI0]:  
Common Name (eg: your user, host, or server name) [SrvVPN]:srv-VPN  
Email Address [toto@menuimetal.fr]:  
* Notice:  
  
Keypair and certificate request completed. Your files are:  
req: /etc/openssl/easy-rsa/pki/reqs/SrvVPN.req  
key: /etc/openssl/easy-rsa/pki/private/SrvVPN.key  
root@srv-VPN:/etc/openssl/easy-rsa#
```

G. Génération du certificat du serveur

```
Using configuration from /etc/openvpn/easy-rsa/pki/ae3d83f0/temp.e7ca0adb
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName             :PRINTABLE:'FR'
stateOrProvinceName     :ASN.1 12:'IDF'
localityName            :ASN.1 12:'MELUN'
organizationName        :ASN.1 12:'menuimetal'
organizationalUnitName  :ASN.1 12:'SIO'
commonName              :ASN.1 12:'SrvVPN'
emailAddress            :IA5STRING:'toto@menuimetal.fr'
Certificate is to be certified until Mar  3 14:20:51 2027 GMT (825 days)

Write out database with 1 new entries
Database updated

* Notice:
Certificate created at: /etc/openvpn/easy-rsa/pki/issued/SrvVPN.crt
```

Validité du certificat généré :

`openssl verify -CAfile pki/ca.crt pki/issued/SrvVPN.crt`

```
root@srv-VPN:/etc/openvpn/easy-rsa# openssl verify -CAfile pki/ca.crt pki/issued/SrvVPN.crt
pki/issued/SrvVPN.crt: OK
root@srv-VPN:/etc/openvpn/easy-rsa#
```

H. Génération des certificats et clés pour les clients VPN

```
root@srv-VPN:/etc/openvpn/easy-rsa# ./easyrsa gen-req CltVPN
* WARNING:

Unsupported characters are present in the vars file.
These characters are not supported: (') (&) (`) ($) (#)
Sourcing the vars file and building certificates will probably fail ..

* Notice:
Using Easy-RSA configuration from: /etc/openvpn/easy-rsa/pki/vars

* Notice:
Using SSL: openssl OpenSSL 3.0.14 4 Jun 2024 (Library: OpenSSL 3.0.14 4 Jun 2024)
```

```
-----  
Country Name (2 letter code) [FR]:  
State or Province Name (full name) [IDF]:  
Locality Name (eg, city) [MELUN]:  
Organization Name (eg, company) [menuimetal]:  
Organizational Unit Name (eg, section) [SIO]:  
Common Name (eg: your user, host, or server name) [CltVPN]:srv-VPN  
Email Address [toto@menuimetal.fr]:  
* Notice:  
  
Keypair and certificate request completed. Your files are:  
req: /etc/openvpn/easy-rsa/pki/reqs/CltVPN.req  
key: /etc/openvpn/easy-rsa/pki/private/CltVPN.key  
  
root@srv-VPN:/etc/openvpn/easy-rsa#
```

I. Génération du certificat du client

```
* Notice:  
Certificate created at: /etc/openvpn/easy-rsa/pki/issued/CltVPN.crt
```

A partir de la commande openssl, vérifiez la validité du certificat client :

```
root@srv-VPN:/etc/openvpn/easy-rsa# openssl verify -CAfile pki/ca.crt pki/issued/CltVPN.crt  
pki/issued/CltVPN.crt: OK  
root@srv-VPN:/etc/openvpn/easy-rsa#
```


J. Génération des paramètres de Diffie-Hellman

```
.....+.
+++++* Notice:
DH parameters of size 2048 created at /etc/openvpn/easy-rsa/pki/dh.pem
root@srv-VPN:/etc/openvpn/easy-rsa# 
root@srv-VPN:/etc/openvpn/easy-rsa# cat /etc/openvpn/easy-rsa/pki/dh.pem
-----BEGIN DH PARAMETERS-----
MIIBCACQAQEASvr6B93g8iV4u12cgKf1PSatBQM06oyDqdjaen2/Bql/0pTzVkQv
00jLs20YRLV9u91eLzlsPeUmlEDwWRefli5q/zN9hSyE+bI6znBPfeEKH492AyGb
a9vhTBqV+tyb9ut2e9dMU0155b0ESxoToVQFWvAuodwOu0ew4ma0s+/WTIs0D6c
irielqUXD4nMHhk7lExm0BQcNTRifTuvp/5L3bzyUGBdq8sRiwIpor09XQ220kSG
Azj9vSx6mu4X4mJKJYZvKA/FwlQJHbcgknlgA2IhIN191ST56sTJgQ/WvfRN0HC
lmW7RgDnpWfv0flR6Yw3fBtsvGZ3aoAYFwIBAg==
-----END DH PARAMETERS-----
root@srv-VPN:/etc/openvpn/easy-rsa#
```

K. Répartition des clés entre client et serveur

C:\Users\sio\Documents*.*				/home/sio/				
Nom	Taille	Type	Date de modification	Nom	Taille	Date de modification	Droits	Propriété
..		Répertoire parent	29/11/2024 09:11:01	..		03/09/2024 17:20:12	rxwx-r-x	root
ca.crt	2 KB	Certificat de sécur...	28/11/2024 17:22:10	ca.crt	2 KB	28/11/2024 17:22:10	rw-----	sio
CitVPN.crt	6 KB	Certificat de sécur...	29/11/2024 10:03:35	CitVPN.crt	6 KB	29/11/2024 10:03:35	rw-----	sio
CitVPN.key	2 KB	Fichier KEY	29/11/2024 10:05:06	CitVPN.key	2 KB	29/11/2024 10:05:06	rw-----	sio
pubc_rsa_key.txt	1 KB	Document texte	28/11/2024 07:35:35					

Positionnez les différents fichiers nécessaires au serveur dans le répertoire adéquat d'openvpn :

```
root@srv-VPN:/etc/openvpn/easy-rsa/pki# cp ca.crt /etc/openvpn/server/
```

```
root@srv-VPN:/etc/openvpn/easy-rsa/pki# mv private/SrvVPN.key /etc/openvpn/server/
```

```
root@srv-VPN:/etc/openvpn/easy-rsa/pki# mv issued/SrvVPN.crt /etc/openvpn/server/
```

```
root@srv-VPN:/etc/openvpn/easy-rsa/pki# mv dh.pem /etc/openvpn/server/
```

L. Sécurisation du serveur VPN

Création d'un groupe nommé openvpn :

```
root@srv-VPN:~# addgroup openvpn
```

```
openvpn:x:1001:  
root@srv-VPN:~# cat /etc/group | grep "openvpn"  
openvpn:x:1001:
```

GID : 1001

```
root@srv-VPN:~# useradd -M openvpn -s /bin/false -g 1001  
root@srv-VPN:~#
```

Mise en place d'un chroot :

```
root@srv-VPN:~# mkdir -p /etc/openvpn/jail/tmp  
root@srv-VPN:~# _
```

Sécurisation du Handshake SSL/TLS :

```
root@srv-VPN:~# openvpn --genkey secret ta.key  
root@srv-VPN:~# ls  
glpi-agent-1.7.1-linux-installer.pl glpi-agent-1.7.1-linux-installer.pl.1 ta.key  
root@srv-VPN:~#
```

```
root@srv-VPN:~# mv ta.key /etc/openvpn/server/  
root@srv-VPN:~# ls /etc/openvpn/server/ | grep ta.key  
ta.key  
root@srv-VPN:~#
```

Configuration dans **/etc/openvpn/server.conf** et démarrage du serveur VPN :

```
mode server
tls-server

local 192.168.11.34

port 1194

proto udp

dev tun

ca /etc/openvpn/server/ca.crt

cert /etc/openvpn/server/SrvVPN.crt

key /etc/openvpn/server/SrvVPN.key

dh /etc/openvpn/server/dh.pem

server 10.8.0.0 255.255.255.0

push "route 192.168.11.0"

push "route 10.8.0.1 255.255.255.255"

push "dhcp-option DNS 192.168.12.1"

push "redirect-gateway def1"

keepalive 10 120

user openvpn
group openvpn

persist-key

persist-tun

client-to-client

max-clients 10

tls-auth /etc/openvpn/server/ta.key

key-direction 0

chroot /etc/openvpn/jail

log /var/log/openvpn.log

status /var/log/openvpn-status.log
```

Vérification du bon fonctionnement du VPN :

```
root@srv-VPN:~# systemctl status openvpn
● openvpn.service - OpenVPN service
   Loaded: loaded (/lib/systemd/system/openvpn.service; enabled; preset: enabled)
   Active: active (exited) since Fri 2024-11-29 16:09:24 CET; 4min 48s ago
   Process: 3333 ExecStart=/bin/true (code=exited, status=0/SUCCESS)
   Main PID: 3333 (code=exited, status=0/SUCCESS)
      CPU: 1ms

nov. 29 16:09:24 srv-VPN systemd[1]: Starting openvpn.service - OpenVPN service...
nov. 29 16:09:24 srv-VPN systemd[1]: Finished openvpn.service - OpenVPN service.
root@srv-VPN:~#
```

Consultation des log :

```
root@srv-VPN:~# tail -f /var/log/openvpn.log
2024-11-29 16:09:56 net_addr_ptp_v4 add: 10.8.0.1 peer 10.8.0.2 dev tun1
2024-11-29 16:09:56 net_route_v4_add: 10.8.0.0/24 via 10.8.0.2 dev [NULL] table 0 metric -1
2024-11-29 16:09:56 sitnl_send: rtnl: generic error (-17): File exists
2024-11-29 16:09:56 NOTE: Linux route add command failed because route exists
2024-11-29 16:09:56 Could not determine IPv4/IPv6 protocol. Using AF_INET
2024-11-29 16:09:56 Socket Buffers: R=[212992->212992] S=[212992->212992]
2024-11-29 16:09:56 TCP/UDP: Socket bind failed on local address [AF_INET]192.168.11.34:1194: Address already in use (errno=98)
2024-11-29 16:09:56 Exiting due to fatal error
2024-11-29 16:09:56 Closing TUN/TAP interface
2024-11-29 16:09:56 net_addr_ptp_v4_del: 10.8.0.1 dev tun1
```

```
root@srv-VPN:~# journalctl -f -u openvpn
nov. 29 15:17:56 srv-VPN systemd[1]: Starting openvpn.service - OpenVPN service...
nov. 29 15:17:56 srv-VPN systemd[1]: Finished openvpn.service - OpenVPN service.
nov. 29 16:09:24 srv-VPN systemd[1]: openvpn.service: Deactivated successfully.
nov. 29 16:09:24 srv-VPN systemd[1]: Stopped openvpn.service - OpenVPN service.
nov. 29 16:09:24 srv-VPN systemd[1]: Stopping openvpn.service - OpenVPN service...
nov. 29 16:09:24 srv-VPN systemd[1]: Starting openvpn.service - OpenVPN service...
nov. 29 16:09:24 srv-VPN systemd[1]: Finished openvpn.service - OpenVPN service.
```

Vérification de l'obtention de la nouvelle interface IP :

```
root@srv-VPN:~# 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: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether bc:24:11:0b:0f:e1 brd ff:ff:ff:ff:ff:ff
    altname enp0s18
    inet 192.168.11.34/24 brd 192.168.11.255 scope global ens18
        valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:fe0b:fe1/64 scope link
        valid_lft forever preferred_lft forever
58: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 500
    link/none
    inet 10.8.0.1 peer 10.8.0.2/32 scope global tun0
        valid_lft forever preferred_lft forever
    inet6 fe80::34cc:e6fb:9cb:8481/64 scope link stable-privacy
        valid_lft forever preferred_lft forever
root@srv-VPN:~#
```

M.Le « forward » de paquets sur le serveur VPN

```
# Uncomment the next line  
net.ipv4.ip_forward=1
```

VII. Fail2BAN

Installez l'application fail2ban sur les machines où se trouvent les services SSH et VPN :

Installation sur GLPI :

```
root@srv-glpi:~# apt install fail2ban
```

Installation sur OpenVPN :

```
root@srv-VPN:~# apt install fail2ban
```

Augmentez le niveau de log à « Debug » pour Fail2ban :

On ouvre le fichier **/etc/fail2ban/fail2ban.conf** :

```
root@srv-glpi:~# vim /etc/fail2ban/fail2ban.conf
```

```
root@srv-VPN:~# vim /etc/fail2ban/fail2ban.conf
```

On change **INFO** par **DEBUG** :

```
[DEFAULT]

# Option: loglevel
# Notes.: Set the log level output.
#         CRITICAL
#         ERROR
#         WARNING
#         NOTICE
#         INFO
#         DEBUG
# Values: [ LEVEL ] Default: INFO
#
loglevel = DEBUG
```

La configuration de Fail2ban par défaut est définie dans le fichier jail.conf, ce fichier est automatiquement modifié lors des mises à jour du service, il est donc recommandé d'effectuer la configuration du service fail2ban dans un fichier de paramètres local jail.local :

cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local

Faire en sorte que le client Windows 10 autorisé à se connecter aux services SSH et VPN ne puisse jamais être banni :

Adresse IP du client Windows : **192.168.11.54**

Fichier **/etc/fail2ban/jail.local** :

```
# "ignoreself" specifies whether the local resp. own IP addresses should be ignored
# (default is true). Fail2ban will not ban a host which matches such addresses.
ignoreself = true

# "ignoreip" can be a list of IP addresses, CIDR masks or DNS hosts. Fail2ban
# will not ban a host which matches an address in this list. Several addresses
# can be defined using space (and/or comma) separator.
ignoreip = 127.0.0.1/8 192.168.11.54
```

On limite cette exception à un service spécifique (SSH) :

```
[sshd]
enabled = true
ignoreip = 192.168.11.54
#
```

Test de connexion :

La connexion échoue 3 fois :

```
28/11/2024 15:07.39 /home/mobaxterm ssh sio@192.168.13.1
Warning: Permanently added '192.168.13.1' (ED25519) to the list of known hosts.
sio@192.168.13.1's password:
sio@192.168.13.1's password:
sio@192.168.13.1's password:
sio@192.168.13.1: Permission denied (publickey,password).
```

L'IP du client Windows n'est pas bannie.

```
root@srv-glpi:~# fail2ban-client status sshd
Status for the jail: sshd
|- Filter
| |- Currently failed: 0
| |- Total failed: 0
| '- File list: /var/log/auth.log
'- Actions
  |- Currently banned: 0
  |- Total banned: 0
  '- Banned IP list:
```

Modifiez la prison (Jail in English) pour le service SSH (ligne 274 du fichier **jail.local**) :

- o Activez la surveillance.
- o Modifiez le port réseau comme voulu précédemment.
- o Ajouter le filtre pour SSH

o Bannissez les IP au bout de trois essais infructueux pendant 60 secondes.

```
[sshd]
enabled = true
ignoreip = 192.168.11.54
port = 2222
filter = sshd
logpath = /var/log/auth.log
maxretry = 3
bantime = 60
findtime = 600
#
```

Redémarrez le service Fail2ban puis vérifiez l'état de la prison pour SSH (commande **fail2ban-client status sshd**) :

```
root@srv-glpi:~# fail2ban-client status sshd
Status for the jail: sshd
|- Filter
| |- Currently failed: 0
| |- Total failed: 0
| `-- File list: /var/log/auth.log
`-- Actions
    |- Currently banned: 0
    |- Total banned: 0
    `-- Banned IP list:
```

Depuis une autre VM équipée d'un client SSH, tentez une connexion avec un mauvais mot de passe puis vérifiez que son adresse IP est bannie :

Test depuis notre VM cliente Ubuntu :

```
admin@clt-ubuntu-desktop-1:~$ ssh sio@192.168.13.1
The authenticity of host '192.168.13.1 (192.168.13.1)' can't be established.
ED25519 key fingerprint is SHA256:PpuM17hi1M6thT218VvcSwRGqf8v3JzLOjcJSEFl+XI.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.13.1' (ED25519) to the list of known hosts.
sio@192.168.13.1's password:
Permission denied, please try again.
sio@192.168.13.1's password:
Permission denied, please try again.
sio@192.168.13.1's password:
sio@192.168.13.1: Permission denied (publickey,password).
```


L'IP est bien bannie après 3 tentatives échouées :

```
root@srv-glpi:~# fail2ban-client status sshd
Status for the jail: sshd
|- Filter
| |- Currently failed: 0
| |- Total failed:      3
| `-- File list:        /var/log/auth.log
`-- Actions
   |- Currently banned: 1
   |- Total banned:     1
   `-- Banned IP list:  192.168.11.31
```

Vérifiez les logs du service Fail2ban avec journalctl :

journalctl -u fail2ban.service -f

```
nov. 28 15:57:16 srv-glpi fail2ban-server[77871]: Server ready
nov. 28 16:02:22 srv-glpi systemd[1]: Stopping fail2ban.service - Fail2Ban Service...
nov. 28 16:02:23 srv-glpi fail2ban-client[77911]: Shutdown successful
nov. 28 16:02:23 srv-glpi systemd[1]: fail2ban.service: Deactivated successfully.
nov. 28 16:02:23 srv-glpi systemd[1]: Stopped fail2ban.service - Fail2Ban Service.
nov. 28 16:02:23 srv-glpi systemd[1]: Started fail2ban.service - Fail2Ban Service.
nov. 28 16:02:23 srv-glpi fail2ban-server[77912]: 2024-11-28 16:02:23.696 fail2ban.configreader [77912]: WARNING 'allowip' not defined in 'Definition'. Using default on
as: 'auto'
nov. 28 16:02:23 srv-glpi fail2ban-server[77912]: Server ready
```

Faire en sorte que les logs de fail2ban soit redirigé vers votre serveur Rsyslog :

Configuration de Fail2ban pour utiliser Rsyslog :

Fichier **/etc/fail2ban/fail2ban.conf** :

On modifie la destination des logs :

```
# Option: logtarget
# Notes.: Set the log target. This could be a file, SYSTEMD-JOURNAL, SYSLOG, STDERR or STDOUT.
#         Only one log target can be specified.
#         If you change logtarget from the default value and you are
#         using logrotate -- also adjust or disable rotation in the
#         corresponding configuration file
#         (e.g. /etc/logrotate.d/fail2ban on Debian systems)
# Values: [ STDOUT | STDERR | SYSLOG | SYSOUT | SYSTEMD-JOURNAL | FILE ] Default: STDERR
#
logtarget = RSYSLOG
```

Rsyslog avait déjà été installé pour GLPI.

On va lui demander de rediriger tous les logs en + de ceux d'authentification déjà envoyés pour récupérer ceux de Fail2Ban :

Fichier **/etc/rsyslog.conf** :

On crée un filtre sur le Rsyslog du serveur Fail2Ban pour lui demander de rediriger les logs de Fail2Ban vers le fichier **/var/log/fail2ban.log** :

```
if $programname == 'fail2ban' then @192.168.11.23:514
```

if \$programname == 'fail2ban' then @192.168.11.23:514

On vérifie sur le serveur Rsyslog (**tail -f /var/log/syslog**) :

```
root@srv-rsyslog-debian:~# tail -f /var/log/syslog
2024-11-28T16:36:22+01:00 srv-glo1 fail2ban-server[78254]: 2024-11-28 16:36:22,963 fail2ban.configreader [78254]: WARNING 'allowip_v6' not defined in 'Definitio
on'. Using default one: 'auto'
2024-11-28T16:36:23+01:00 srv-glo1 fail2ban-server[78254]: Server ready
2024-11-28T16:37:41+01:00 srv-glo1 systemd[1]: Started session-1742.scope - Session 1742 of User sio.
2024-11-28T16:37:41+01:00 srv-glo1 systemd[1]: Started session-1743.scope - Session 1743 of User sio.
2024-11-28T16:37:46+01:00 srv-glo1 systemd[1]: session-1742.scope: Deactivated successfully.
2024-11-28T16:37:46+01:00 srv-glo1 systemd[1]: session-1743.scope: Deactivated successfully.
2024-11-28T16:39:01+01:00 srv-glo1 CRON[76290]: (root) CMD ( [ -x /usr/lib/php/sessionclean ] && if [ ! -d /run/systemd/system ]; then /usr/lib/php/sessionclea
n; fi)
2024-11-28T16:39:01+01:00 srv-glo1 systemd[1]: Starting phpsessionclean.service - Clean php session files...
2024-11-28T16:39:01+01:00 srv-glo1 systemd[1]: phpsessionclean.service: Deactivated successfully.
2024-11-28T16:39:01+01:00 srv-glo1 systemd[1]: Finished phpsessionclean.service - Clean php session files.
:q
^C
root@srv-rsyslog-debian:~# tail -f /var/log/syslog
2024-11-28T16:36:22+01:00 srv-glo1 fail2ban-server[78254]: 2024-11-28 16:36:22,963 fail2ban.configreader [78254]: WARNING 'allowip_v6' not defined in 'Definitio
on'. Using default one: 'auto'
2024-11-28T16:36:23+01:00 srv-glo1 fail2ban-server[78254]: Server ready
2024-11-28T16:37:41+01:00 srv-glo1 systemd[1]: Started session-1742.scope - Session 1742 of User sio.
2024-11-28T16:37:41+01:00 srv-glo1 systemd[1]: Started session-1743.scope - Session 1743 of User sio.
2024-11-28T16:37:46+01:00 srv-glo1 systemd[1]: session-1742.scope: Deactivated successfully.
2024-11-28T16:37:46+01:00 srv-glo1 systemd[1]: session-1743.scope: Deactivated successfully.
2024-11-28T16:39:01+01:00 srv-glo1 CRON[76290]: (root) CMD ( [ -x /usr/lib/php/sessionclean ] && if [ ! -d /run/systemd/system ]; then /usr/lib/php/sessionclea
n; fi)
2024-11-28T16:39:01+01:00 srv-glo1 systemd[1]: Starting phpsessionclean.service - Clean php session files...
2024-11-28T16:39:01+01:00 srv-glo1 systemd[1]: phpsessionclean.service: Deactivated successfully.
2024-11-28T16:39:01+01:00 srv-glo1 systemd[1]: Finished phpsessionclean.service - Clean php session files.
```

On va maintenant rediriger ces logs dans un fichier spécifique du serveur Rsyslog :

Fichier **/etc/rsyslog.conf** :

```
#template syslog,"/var/log/clients/%fromhost%/syslog.log"
*. * ?syslog
```

Les machines sont bien référencées par IP :

```
root@srv-rsyslog-debian:/var/log/clients# ls
192.168.11.30 192.168.13.1 192.168.13.2 192.168.13.4 dafdpexit mysql srv-rsyslog-debian
```

On vérifie dans le fichier **/var/log/clients/192.168.13.1/syslog.log** qui correspond à l'ip de notre GLPI où Fail2Ban est installé :

```
root@srv-rsyslog-debian:/var/log/clients/192.168.13.1# ls
syslog.log
```

```
root@srv-rsyslog-debian:/var/log/clients/192.168.13.1# tail -f syslog.log
2024-11-28T16:56:21+01:00 srv-gloi sshd[77979]: Received signal 15: terminating.
2024-11-28T16:56:21+01:00 srv-gloi sshd[78375]: Server listening on 0.0.0.0 port 2222.
2024-11-28T16:56:21+01:00 srv-gloi sshd[78375]: Server listening on :: port 2222.
2024-11-28T16:56:22+01:00 srv-gloi sshd[78375]: Received signal 15: terminating.
2024-11-28T16:56:22+01:00 srv-gloi sshd[78379]: Server listening on 0.0.0.0 port 2222.
2024-11-28T16:56:22+01:00 srv-gloi sshd[78379]: Server listening on :: port 2222.
2024-11-28T16:56:39+01:00 srv-gloi sshd[78380]: Accepted publickey for sio from 192.168.11.54 port 10777 ssh2: RSA SHA256:igN0mlIzbXdu98b08KoG4HCKKoNQu9kze53J27
1aUsQ
2024-11-28T16:56:39+01:00 srv-gloi systemd-logind[438]: New session 1745 of user sio.
2024-11-28T16:56:39+01:00 srv-gloi sshd[78382]: Accepted publickey for sio from 192.168.11.54 port 10778 ssh2: RSA SHA256:igN0mlIzbXdu98b08KoG4HCKKoNQu9kze53J27
1aUsQ
2024-11-28T16:56:39+01:00 srv-gloi systemd-logind[438]: New session 1746 of user sio.
```

Paramétrez Fail2ban pour qu'il puisse envoyer un mail à l'administrateur lorsqu'une IP est bannie. Testez la réception du mail depuis un client de messagerie :

Fichier **/etc/fail2ban/jail.local** :

```
# ACTIONS
#

# Some options used for actions

# Destination email address used solely for the interpolations i
n
# jail.{conf,local,d/*} configuration files.
destemail = toto@menuimetal.fr

# Sender email address used solely for some actions
sender = fail2ban

# E-mail action. Since 0.8.1 Fail2Ban uses sendmail MTA for the
# mailing. Change mta configuration parameter to mail if you wan
t to
# revert to conventional 'mail'.
mta = sendmail
```

```
# Choose default action. To change, just override value of 'action' with the
# interpolation to the chosen action shortcut (e.g. action_mw, action_mwl, etc) in jail.local
# globally (section [DEFAULT]) or per specific section
action = %(action_mwl)s
```

Installation de sendmail :

```
root@srv-glpi:~# apt install sendmail
```

Fichier `/etc/mail/sendmail.mc` :

```
dnf # Default Mailer setup
MAILER_DEFINITIONS
FEATURE(`authinfo')dnf
MAILER(`local')dnf
MAILER(`smtp')dnf
define(`SMART_HOST', `[192.168.12.3]')dnf
define(`RELAY_MAILER_ARGS', `TCP $h 25')dnf
define(`confAUTH_OPTIONS', `A p')dnf
define(`confAUTH_MECHANISMS', `PLAIN LOGIN')dnf
```

Fichier `/etc/mail/authinfo` :

```
AuthInfo:192.168.12.3 "U:toto" "P:toto" "M:PLAIN"
```

On vérifie le nom de notre serveur GLPI à l'aide de la commande nslookup sur le serveur DNS :

```
Name:   srv-glpi.menuimetal.fr
Address: 192.168.13.1
```

Fichier /etc/hosts :

```
192.168.13.1   srv-glpi.menuimetal.fr srv-glpi

# The following lines are desirable for IPv6 capable hosts
::1           localhost ip6-localhost ip6-loopback
ff02::1       ip6-allnodes
ff02::2       ip6-allrouters
```

```
root@srv-glpi:~# makemap hash /etc/mail/authinfo < /etc/mail/authinfo
root@srv-glpi:~# m4 /etc/mail/sendmail.mc > /etc/mail/sendmail.cf
root@srv-glpi:~# systemctl restart sendmail.service
```

```
root@srv-glpi:~# systemctl status sendmail
● sendmail.service - LSB: powerful, efficient, and scalable Mail Transport Agent
   Loaded: loaded (/etc/init.d/sendmail; generated)
   Active: active (running) since Fri 2024-11-29 15:02:36 CET; 59s ago
     Docs: man:systemd-sysv-generator(8)
  Process: 84059 ExecStart=/etc/init.d/sendmail start (code=exited, status=0/SUCCESS)
    Tasks: 1 (limit: 2306)
   Memory: 2.3M
      CPU: 91ms
  CGroup: /system.slice/sendmail.service
          └─84114 "sendmail: MTA: accepting connections"

nov. 29 15:02:34 srv-glpi systemd[1]: Stopped sendmail.service - LSB: powerful, efficient, and scalable Mail Transport Agent.
nov. 29 15:02:34 srv-glpi systemd[1]: Starting sendmail.service - LSB: powerful, efficient, and scalable Mail Transport Agent...
nov. 29 15:02:34 srv-glpi su[84091]: (to smmsp) root on none
nov. 29 15:02:34 srv-glpi su[84091]: pam_unix(su:session): session opened for user smmsp(uid=105) by (uid=0)
nov. 29 15:02:34 srv-glpi su[84091]: pam_unix(su:session): session closed for user smmsp
nov. 29 15:02:34 srv-glpi sm-mta[84114]: starting daemon (8.17.1.9): SMTP+queueing@00:10:00
nov. 29 15:02:36 srv-glpi sendmail[84059]: Starting Mail Transport Agent (MTA): sendmail.
nov. 29 15:02:36 srv-glpi systemd[1]: Started sendmail.service - LSB: powerful, efficient, and scalable Mail Transport Agent.
```

```
root@srv-glpi:~# tail -f /var/log/mail.log
2024-11-29T14:11:45.252808+01:00 srv-glpi sendmail[82861]: gethostbyaddr(192.168.13.1) failed: 1
2024-11-29T14:11:45.257301+01:00 srv-glpi sendmail[82861]: alias database /etc/mail/aliases rebuilt by sio
2024-11-29T14:11:45.258034+01:00 srv-glpi sendmail[82861]: /etc/mail/aliases: 0 aliases, longest 0 bytes, 0 bytes total
2024-11-29T14:11:45.737453+01:00 srv-glpi sm-mta[82928]: gethostbyaddr(192.168.13.1) failed: 1
2024-11-29T14:11:45.755764+01:00 srv-glpi sm-mta[82930]: starting daemon (8.17.1.9): SMTP+queueing@00:10:00
2024-11-29T14:44:32.375588+01:00 srv-glpi sm-mta[83419]: NOQUEUE: SYSERR(root): /etc/mail/sendmail.cf: line 1794: unknown configuration line "\n (authinfo)"
2024-11-29T14:45:21.209408+01:00 srv-glpi sm-mta[83548]: NOQUEUE: SYSERR(root): /etc/mail/sendmail.cf: line 1794: unknown configuration line "\n (authinfo)"
2024-11-29T14:59:34.626520+01:00 srv-glpi sm-mta[83810]: NOQUEUE: SYSERR(root): /etc/mail/sendmail.cf: line 1794: unknown configuration line "\n (authinfo)"
2024-11-29T15:02:01.552217+01:00 srv-glpi sm-mta[83993]: starting daemon (8.17.1.9): SMTP+queueing@00:10:00
2024-11-29T15:02:34.675634+01:00 srv-glpi sm-mta[84114]: starting daemon (8.17.1.9): SMTP+queueing@00:10:00
```

```
root@srv-glpi:~# systemctl restart fail2ban.service
```

Test de réception de la notification si un compte est banni :

On reteste avec notre client Ubuntu auquel on a préalablement débanni l'IP :

```
root@srv-glpi:~# fail2ban-client status
Status
|- Number of jail:      1
`- Jail list:  sshd
root@srv-glpi:~# fail2ban-client set sshd unbanip 192.168.11.31
0
```

```
sio@clt-ubuntu-desktop-1:~$ ssh -p 2222 sio@192.168.13.1
The authenticity of host '[192.168.13.1]:2222 ([192.168.13.1]:2222)' can't be es
tablished.
ED25519 key fingerprint is SHA256:PpuM17hi1M6thT218VvcSwRGqf8v3JzLOjcJSEFl+XI.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[192.168.13.1]:2222' (ED25519) to the list of known
hosts.
sio@192.168.13.1's password:
Permission denied, please try again.
sio@192.168.13.1's password:
Permission denied, please try again.
sio@192.168.13.1's password:
sio@192.168.13.1: Permission denied (password).
```

On reçoit bien le mail sur le compte toto@menuimetal.fr :

Fail2Ban
fail2ban@srv-glpi.menuimetal.fr

Pour toto@menuimetal.fr

[Fail2Ban] sshd: banned 192.168.11.31 from srv-glpi.menuimetal.fr

Hi,

The IP 192.168.11.31 has just been banned by Fail2Ban after 3 attempts against sshd.

Here is more information about 192.168.11.31 :
missing whois program

Lines containing failures of 192.168.11.31 (max 1000)

```
2024-11-28T15:59:16.904205+01:00 srv-glpi sshd[77883]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.11.31 user=sio
2024-11-28T15:59:18.620845+01:00 srv-glpi sshd[77883]: Failed password for sio from 192.168.11.31 port 49060 ssh2
2024-11-28T15:59:23.139087+01:00 srv-glpi sshd[77883]: Failed password for sio from 192.168.11.31 port 49060 ssh2
2024-11-28T15:59:26.772421+01:00 srv-glpi sshd[77883]: Failed password for sio from 192.168.11.31 port 49060 ssh2
2024-11-28T15:59:27.339253+01:00 srv-glpi sshd[77883]: Connection closed by authenticating user sio 192.168.11.31 port 49060 [preauth]
2024-11-28T15:59:27.339506+01:00 srv-glpi sshd[77883]: PAM 2 more authentication failures; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.11.31 user=sio
2024-11-29T15:45:53.026542+01:00 srv-glpi sshd[84544]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser= rhost=192.168.11.31 user=sio
2024-11-29T15:45:55.247006+01:00 srv-glpi sshd[84544]: Failed password for sio from 192.168.11.31 port 48116 ssh2
2024-11-29T15:46:00.534232+01:00 srv-glpi sshd[84544]: Failed password for sio from 192.168.11.31 port 48116 ssh2
2024-11-29T15:46:03.763510+01:00 srv-glpi sshd[84544]: Failed password for sio from 192.168.11.31 port 48116 ssh2
```

Regards,

Fail2Ban

Installation du plugin Fail2Ban sur Wordpress :

☐ **WP fail2ban**
[Settings](#) | [Upgrade](#) | [Add-Ons](#) | [Deactivate](#)

Write a myriad of WordPress events to syslog for integration with fail2ban.
Version 5.3.4 | By Charles Lecklider | [Visit plugin site](#)

Installation de Fail2Ban :

```
root@srv-web-wp:~# apt install fail2ban
```

```
root@srv-web-wp:~# cp /var/www/html/wp-content/plugins/wp-fail2ban/wordpress.conf /etc/fail2ban/filter.d/_
```

```
#  
# JAILS  
#  
[wordpress]  
enabled = true  
port = http, https  
filter = wordpress  
maxretry = 3  
banaction = iptables-multiport  
logpath = /var/log/auth.log  
#
```

```
root@srv-web-wp:/var/www/html/wp-content/plugins/wp-fail2ban# systemctl restart fail2ban.service  
root@srv-web-wp:/var/www/html/wp-content/plugins/wp-fail2ban#
```