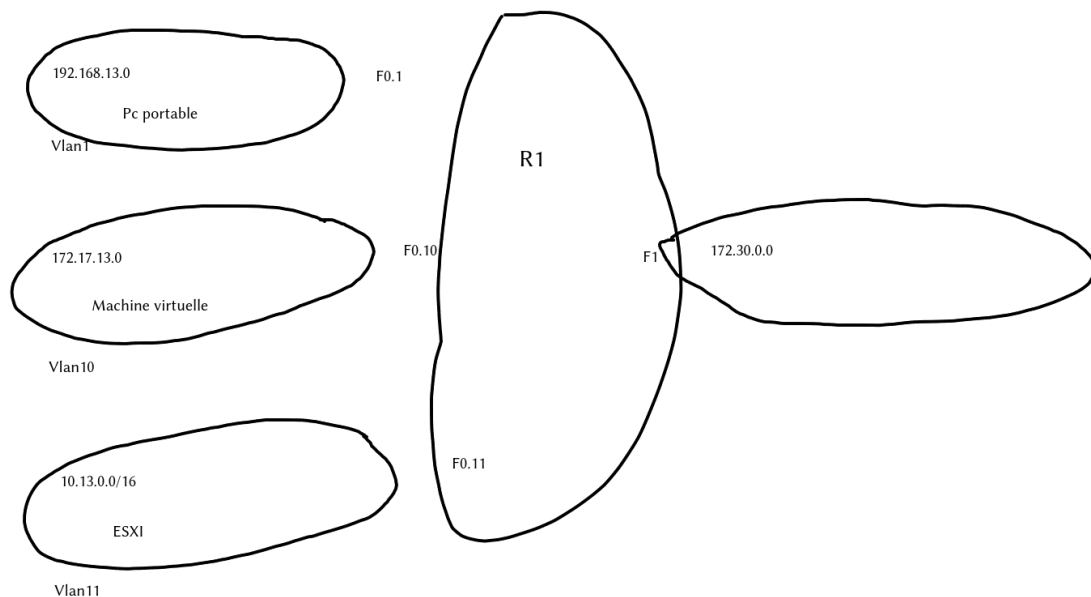


TP: Gestion routage LAN sur vlans + VPN

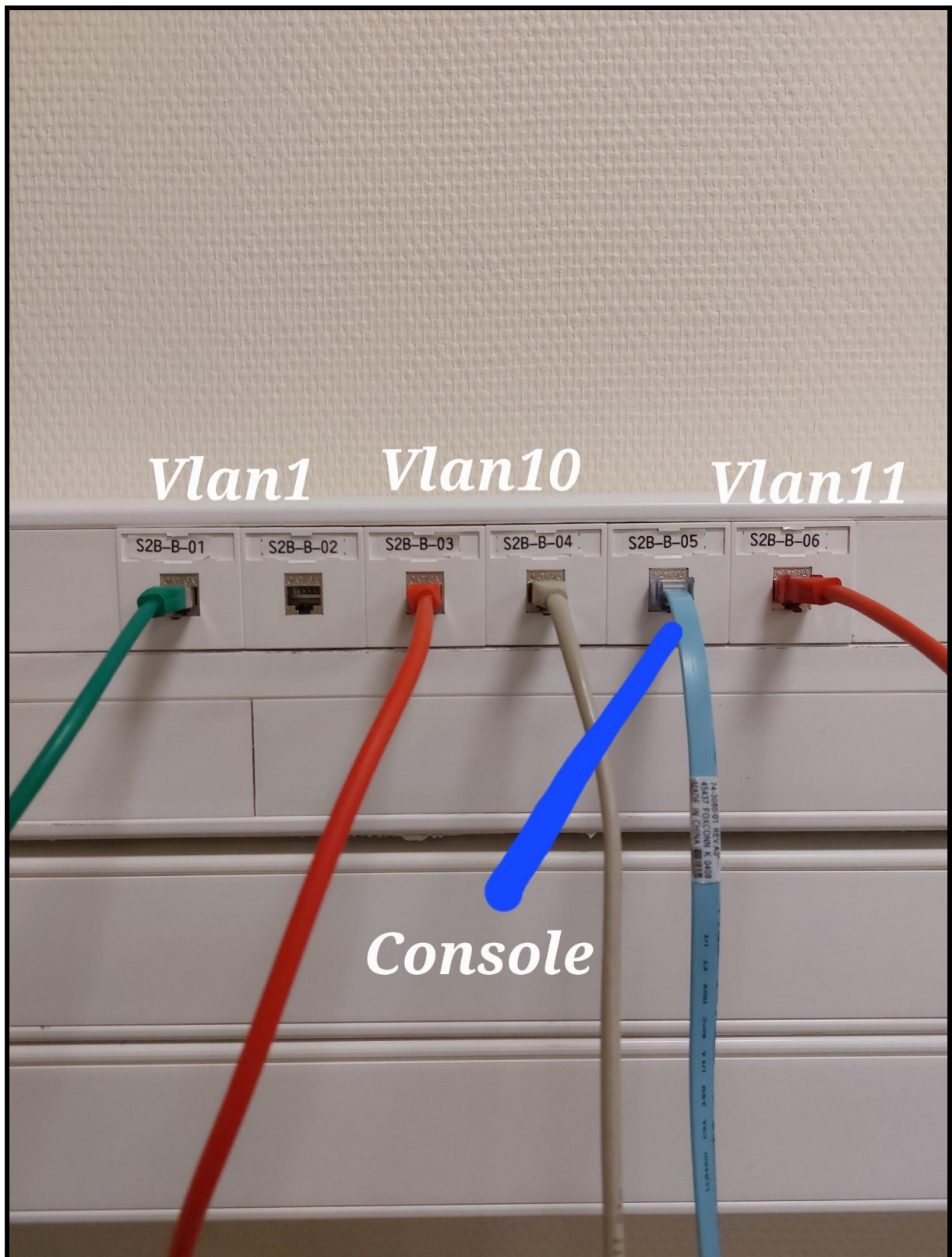
Allag
Mehdi
SIO1

Introduction: Il sera dans ce type question d'effectuer la connexion entre différentes LAN d'une ville et de les relier au MAN avec un accès à internet. Les deux LAN admin, et scol seront ici représentés par des vlan que nous pourrons accéder plus tard via une machine virtuelle.

Aperçu schéma de niveau 3:



Aperçu branchement des câbles:



-Cable bleu —> cable console pour se connecter aux différentes interfaces Cisco (routeur, switch)

-Câbles 01, 03 et 06 -> connectées depuis nos prises murales vers la baie de brassage, puis reliés à notre switch pour la configuration de nos trois vlans différents.



Aperçu du câblage côté routeur:

- Cable blanc —> relie le routeur au reseau depuis l'interface GE 0/1
- Cable jaune —> pour acceder aux l'interfaces phisique du routeur.
- Cable bleu —> relié au switch via interface GE 0/1

Reconfiguration de l'équipement (au cas ou...):

Les switches et routeurs utilisés pour réaliser ce tp ont probablement déjà été utilisés pour d'autres projets et ont donc probablement des configurations pré-existantes qui pourraient interférer avec notre projet.

Pour configurer le routeur Cisco, nous utiliserons les commandes suivantes après s'être connecté avec un port console:

-enable

-write erase (On nous demande de confirmer la suppression de la mémoire nvram)

-erase

-reload

Pour reconfigurer notre switch:

-clear config all

write memory to keep conf

Configuration des vlans:

Nous commencerons par faire nos trois vlans.

Pour cela:

```
en
conf t
vlan (name) #assigner un nom à notre vlan
exit
int range fa0/(port-port) #assignation du vlan à un/des ports
switchport mode access
switchport mode access vlan (nom du vlan)
no shut
end
```

Mise en place du mode trunk:

- Modetrunk (switch série 2960)
- configure terminal
- interface gigabitEthernet 0/1
- switchport mode trunk
- no shut

Configuration final des vlans:

```
Switch#show vlan
```

VLAN	Name	Status	Ports
1	default	active	Fa0/1, Fa0/2, Fa0/7, Fa0/8 Fa0/9, Fa0/10, Fa0/11, Fa0/12 Fa0/13, Fa0/14, Fa0/15, Fa0/16 Fa0/17, Fa0/18, Fa0/19, Fa0/20 Fa0/21, Fa0/22, Fa0/23, Fa0/24 Gi0/2
3	VLAN0003	active	
10	Windows	active	Fa0/3, Fa0/4
11	esix	active	Fa0/5, Fa0/6
22	VLAN0022	active	
1002	fddi-default	act/unsup	
1003	token-ring-default	act/unsup	
1004	fddinet-default	act/unsup	
1005	trnet-default	act/unsup	

VLAN	Type	SAID	MTU	Parent	RingNo	BridgeNo	Stp	BrdgMode	Trans1	Trans2
1	enet	100001	1500	-	-	-	-	-	0	0
3	enet	100003	1500	-	-	-	-	-	0	0
10	enet	100010	1500	-	-	-	-	-	0	0

VLAN	Type	SAID	MTU	Parent	RingNo	BridgeNo	Stp	BrdgMode	Trans1	Trans2
11	enet	100011	1500	-	-	-	-	-	0	0
22	enet	100022	1500	-	-	-	-	-	0	0
1002	fddi	101002	1500	-	-	-	-	-	0	0
1003	tr	101003	1500	-	-	-	-	-	0	0
1004	fdnet	101004	1500	-	-	-	ieee	-	0	0
1005	trnet	101005	1500	-	-	-	ibm	-	0	0

Les vlans default, windows et esix sont bien respectivement assignées au ports 1-2, 3-4 et 5-6 de notre interface FastEthernet/0, comme prévu.

Configuration du routeur:

```
-en
-conf t
-interface gigabitethernet 0/1
-encapsulation dot1q
- ip add 172.30.13.1
-no ip proxy-arp
-no sh
```

Nous mettrons en place de la même manière toutes les ip de nos vlans aux différentes sous-interfaces du routeur.

Route final ver le routeur de la salle depuis l'interface Gigabitethernet0/0:

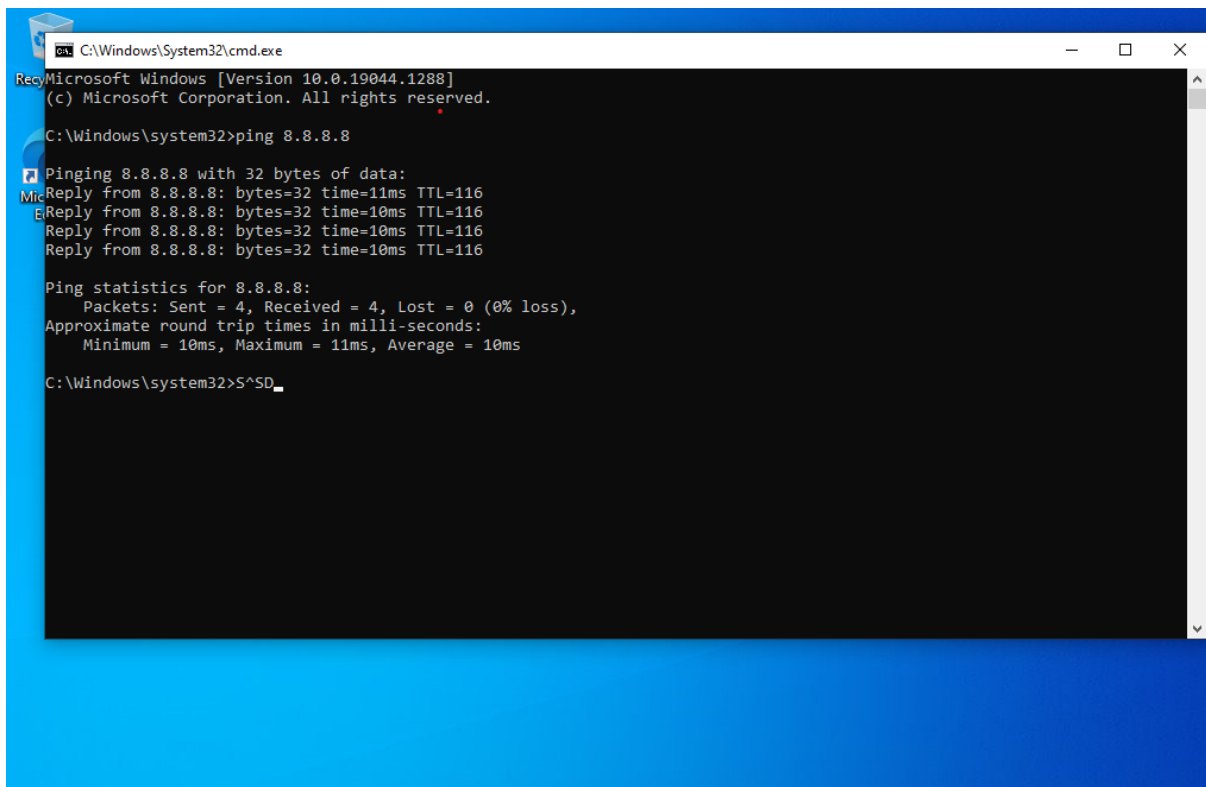
```
en
interface gigabitethernet0/0
```

```
ip route 0.0.0.0 0.0.0.0 172.30.0.40  
no sh
```

Configuration final du routeur:

```
interface Embedded-Service-Engine0/0
  no ip address
  shutdown
!
interface GigabitEthernet0/0
  no ip address
  no ip proxy-arp
  duplex auto
  speed auto
!
interface GigabitEthernet0/0.1
  encapsulation dot1Q 1 native
  ip address 192.168.12.1 255.255.255.0
  no ip proxy-arp
!
interface GigabitEthernet0/0.10
  encapsulation dot1Q 10
  ip address 172.17.12.1 255.255.255.0
  no ip proxy-arp
!
interface GigabitEthernet0/0.11
  encapsulation dot1Q 11
  ip address 10.12.0.1 255.255.0.0
  no ip proxy-arp
!
interface GigabitEthernet0/1
  ip address 172.30.0.12 255.255.0.0
  no ip proxy-arp
  duplex auto
  speed auto
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
```

Test final de connectivité depuis notre machine virtuelle:



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19044.1288]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:
Reply from 8.8.8.8: bytes=32 time=11ms TTL=116
Reply from 8.8.8.8: bytes=32 time=10ms TTL=116
Reply from 8.8.8.8: bytes=32 time=10ms TTL=116
Reply from 8.8.8.8: bytes=32 time=10ms TTL=116

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 10ms, Maximum = 11ms, Average = 10ms

C:\Windows\system32>S^SD_
```

Notre réseau est donc bien configuré.

DEUXIÈME PARTIE:

Sommaire:

- I- Solutions choisies
- II- Mise en place des solutions
- III- Conclusion

I- Pourquoi Pritunl + Ubuntu.

Le choix de Pritunl en tant que client VPN est ici assez facile à faire. Le logiciel est simple d'installation avec une interface gui intuitive et directe. Beaucoup de documentations sont disponibles et guident l'utilisateur à travers les différents problèmes possibles à rencontrer. L'installation du client se fera sur une machine virtuelle Ubuntu 20.04. La version de Pritunl que nous utilisons n'est à ce jour pas disponible sur la tout dernière version d'Ubuntu

(22). Il est intéressant de noter que nous aurions pu choisir une machine Debian, CentOS, Debian, ou même Fedora. Cette solution est également gratuite, même s'il existe des versions payantes avec plus de fonctionnalités.

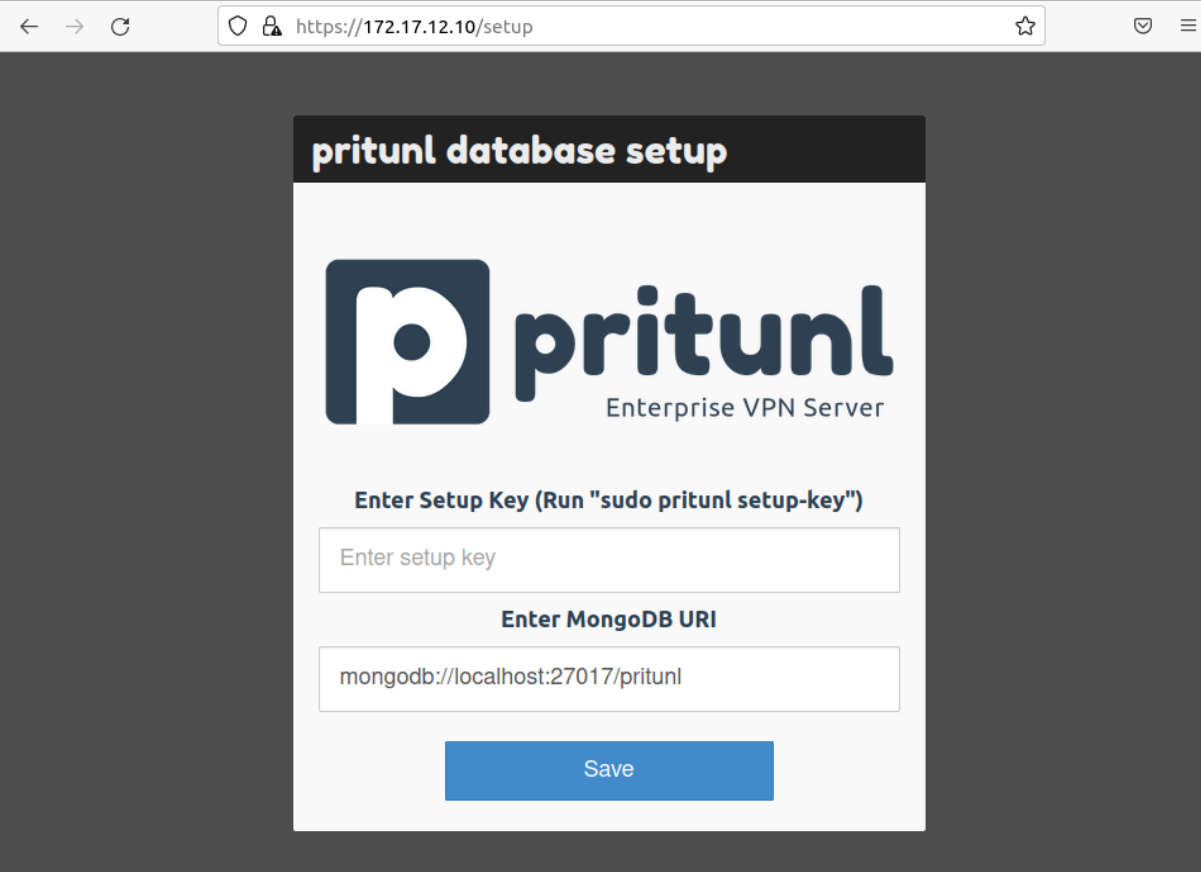
II- Installation et configuration d'un service vpn: Pritunl

Nous verrons sous cette partie comment installer et configurer Pritunl VPN sous Ubuntu 20.04. Pritunl VPN est un serveur VPN open source. Il utilise plutôt simple à mettre en place et possède une interface graphique qui est conviviale et facile à utiliser pour l'utilisateur.

Pour commencer à l'installer, se rendre sur le site officiel de Pritunl ou ils mettent à disposition une commande pour installer le client:


```
1 sudo tee /etc/apt/sources.list.d/pritunl.list << EOF
2 deb https://repo.pritunl.com/stable/apt jammy main
3 EOF
4
5 sudo apt --assume-yes install gnupg
6 gpg --keyserver hkp://keyserver.ubuntu.com --recv-keys
7 7568D9B855FF9E5287D586017AE645C0CF8E292A
8 gpg --armor --export 7568D9B855FF9E5287D586017AE645C0CF8E292A | sudo tee
9 /etc/apt/trusted.gpg.d/pritunl.asc
10 sudo apt update
11 sudo apt install pritunl-client-electron
```

A ce stade, Pritunl VPN est installé et fonctionne. Nous y accéderont depuis le navigateur en utilisant l'IP de notre serveur pour le configurer. Vous devriez obtenir une page comme ci-dessous :



← → ↻ https://172.17.12.10/setup ☆

pritunl database setup



pritunl
Enterprise VPN Server

Enter Setup Key (Run "sudo pritunl setup-key")

Enter MongoDB URI

Save

Il nous faudra générer une clef de configuration:

```
med@med-Virtual-Machine:~$ sudo pritunl setup-key
9e4d1fa2013e432ca828d2a37b297b40
```

Puis ensuite générer un mot de passe et nom d'utilisateur.

```
med@med-Virtual-Machine:~$ sudo pritunl default-password
[undefined][2022-10-19 17:04:46,391][INFO] Getting default administrator password
Administrator default password:
  username: "pritunl"
  password: "p4bAo2lpVsKB"
```

Création de notre serveur:

Il nous faudra par la suite créer un serveur avec les paramètres suivants: (nom, dns, adresse du réseau...) Il compte utilisateur sera également mis en place via les menus suivants:

Add Server

Advanced



Name

monservvpn|

DNS Server

8.8.8.8

Port

13149

Protocol

udp

Virtual Network

192.168.220.0/24

253 Users

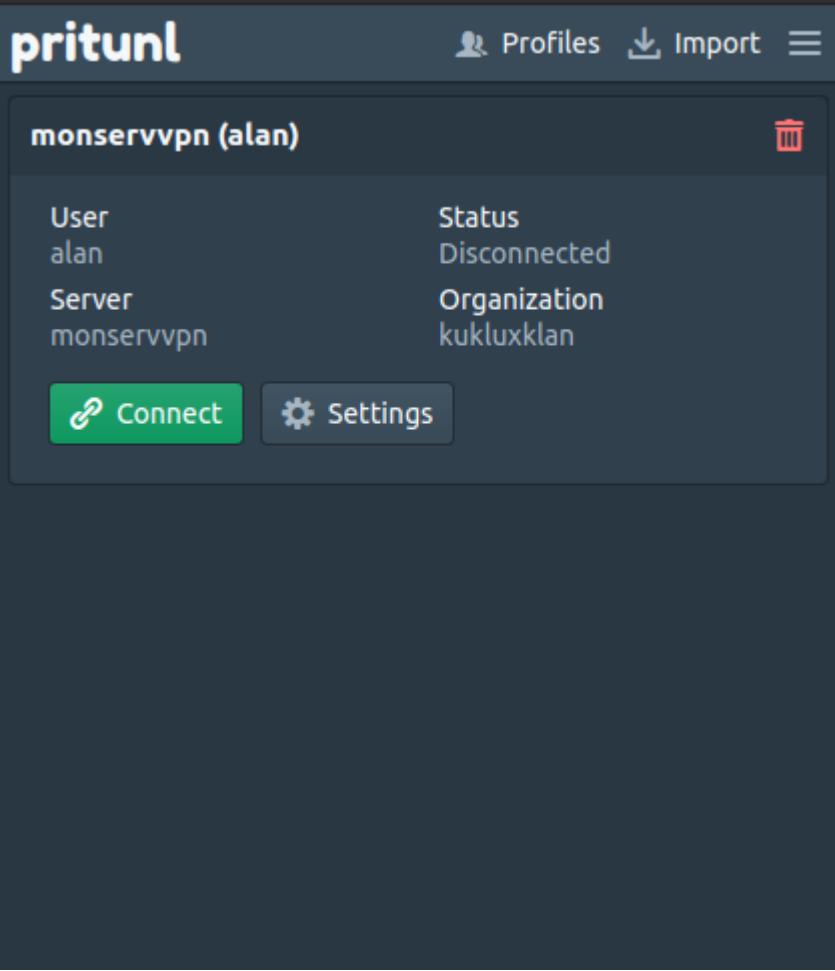
☐ Enable WireGuard

☐ Enable Google Authenticator

☐ Enable IPv6

Cancel

Add



Nous nous contacterons avec l'utilisateur précédemment mis en place.

Nous somme ici connecté:

monservvpn (alan)



User	Online For
alan	19 secs
Server	Organization
monservvpn	kukluxklan
Server Address	Client Address
172.17.12.10	192.168.220.2

Disconnect

Settings

Logs de connections sur l'interface de notre serveur:

Restart Server
Stop Server
Delete Server

Server Output
Bandwidth Graphs

```

6 [guarded-meadow-3001] Wed Oct 19 17:16:22 2022 setsockopt(IPV6_V6ONLY=0)
7 [guarded-meadow-3001] Wed Oct 19 17:16:22 2022 UDPv6 link local (bound): [AF_INET6][undef]:13149
8 [guarded-meadow-3001] Wed Oct 19 17:16:22 2022 UDPv6 link remote: [AF_UNSPEC]
9 [guarded-meadow-3001] Wed Oct 19 17:16:22 2022 Initialization Sequence Completed
10 [guarded-meadow-3001] 2022-10-19 17:16:23 COM> SUCCESS: bytecount interval changed
11 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: IV_VER=2.4.7
12 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: IV_PLAT=linux
13 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: IV_PROTO=2
14 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: IV_NCP=2
15 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: IV_LZ4=1
16 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: IV_LZ4v2=1
17 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: IV_LZO=1
18 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: IV_COMP_STUB=1
19 [guarded-meadow-3001] 2022-10-19 17:18:02 COM> SUCCESS: client-auth command succeeded
20 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: IV_COMP_STUBv2=1
21 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: IV_TCPNL=1
22 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: IV_HMADDR=00:15:5d:49:31:35
23 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: IV_SSL=OpenSSL_1.1.1f_31_Mar_2020
24 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: UV_ID=632ad642f0e7456fa571e0a1e13602f7
25 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 peer info: UV_NAME=autumn-plains-3987
26 [guarded-meadow-3001] Wed Oct 19 17:18:01 2022 172.17.12.10:46047 [6350121c5d6acf37da9eacfa] Peer Connection Initiated
27 [guarded-meadow-3001] Wed Oct 19 17:18:02 2022 6350121c5d6acf37da9eacfa/172.17.12.10:46047 MULTI_sva: pool returned IP
28 [guarded-meadow-3001] 2022-10-19 17:18:02 User connected user_id=6350121c5d6acf37da9eacfa
29

```

VPN fonctionnel:

The screenshot displays a VPN status interface on a dark blue background. At the top left, under 'Mon adresse IP:', the IPv4 address is '82.66.54.102' and IPv6 is 'Not detected'. Below this, 'Mes informations IP:' lists: FAI: ProXad/Free SAS, Ville: Claix, Région: Auvergne-Rhône-Alpes, and Pays: France. A central red button reads 'MASQUER MON ADRESSE IP' with a shield icon. Below the button is a link 'Afficher les détails IP complets'. To the right, a map shows the location in the Rhône-Alpes region of France, with a tooltip for '82.66.54.102' and a link 'Cliquez pour plus de détails sur 82.66.54.102'. Below the map, it says 'Emplacement inexact?' with a link 'Mettre à jour mon emplacement d'IP'. A warning message 'Vos informations privées sont exposées!' is positioned between the IP info and the map.

Mon adresse IP:

IPv4: ? 82.66.54.102

IPv6: ? Not detected

Mes informations IP:

FAI: ProXad/Free SAS

Ville: Claix

Région: Auvergne-Rhône-Alpes

Pays: France

Vos informations privées sont exposées!

MASQUER MON ADRESSE IP

[Afficher les détails IP complets](#)

Emplacement inexact?

[Mettre à jour mon emplacement d'IP](#)

-Conclusion:

Après installation et test de notre service, celui-ci fonctionne bien et répond à nos besoins.