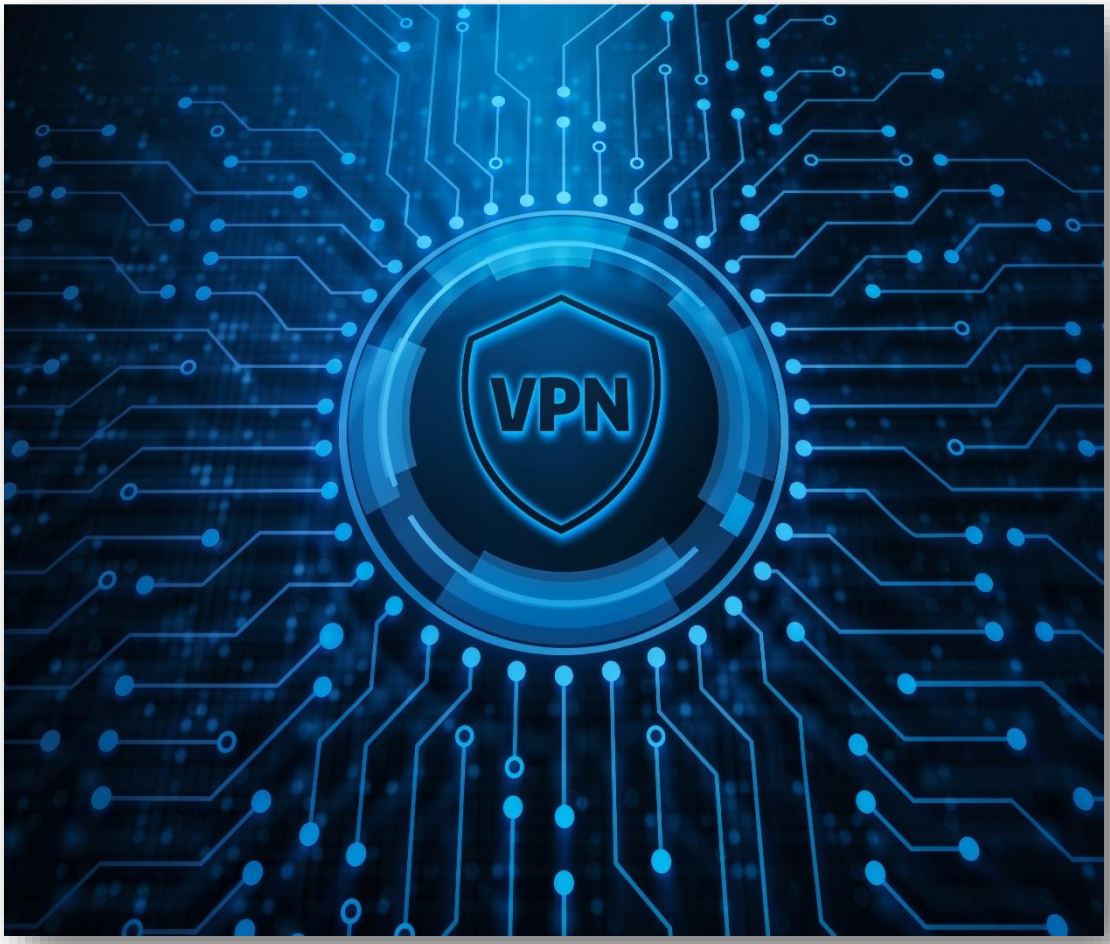
MISE EN PLACE D'UN SERVEUR VPN



Samet ARI

La Plateforme B1-Prepa

Mai-2025

TABLE DES MATIÈRES

Page

1. INTRODUCTION

Dans le cadre de mon projet en réseau et sécurité informatique, j'ai réalisé la mise en place complète d'un serveur **VPN (Virtual Private Network)** utilisant **OpenVPN.** Ce projet m'a permis d'approfondir mes connaissances en sécurité réseau, cryptographie et administration système Linux.

Contexte du Projet

Avec l'augmentation des cyberattaques et la nécessité de sécuriser les communications dans un monde de plus en plus connecté, les **VPN** sont devenus essentiels pour protéger les données en transit. J'ai choisi **OpenVPN** pour sa flexibilité, sa sécurité robuste basée sur les protocoles **TLS/SSL** et sa compatibilité multiplateforme.

Ce projet s'inscrit dans une démarche d'apprentissage pratique des technologies de sécurité réseau, me permettant de comprendre concrètement les enjeux de la protection des communications numériques.

2. OBJECTIFS DU PROJET

Objectifs Techniques

- **Installer et configurer** un serveur **OpenVPN** sur Debian Server (192.168.154.2XX)
- **Mettre en place** une infrastructure de certificats (PKI) avec Easy-RSA
- **Configurer** les clients VPN pour différentes plateformes (focus sur Debian 12)
- **Sécuriser** l'infrastructure avec des règles firewall appropriées (iptables)
- **Tester** et valider le fonctionnement complet du système VPN
- **Implémenter** le chiffrement moderne (TLS 1.3, AES-256-GCM)

Objectifs Pédagogiques

- Comprendre les protocoles de sécurité réseau (TLS/SSL)
- Maîtriser la gestion des certificats numériques et l'infrastructure PKI
- Apprendre l'administration avancée de serveurs Debian
- Développer des compétences en débogage réseau et résolution de problèmes
- Acquérir une expertise en sécurisation des communications

3. ARCHITECTURE SYSTÈME

Environnement Technique

Serveur VPN : Debian Server 22.04 LTS (192.168.154.2XX)

Client : Debian 12 (192.168.154.20X)

Réseau Local : 192.168.154.X/24

Réseau VPN : 10.8.0.0/24

Hyperviseur : VMware Workstation

Topologie Réseau Implémentée

Internet (176.181.177.125)

↓

NAT/Router (192.168.154.2)

↓

Réseau Local (192.168.154.0/24)

↙ ↘

VPN Server (.209) **VPN Client** (.208)

↓

Tunnel VPN Chiffré (UDP:1194)

↓

Réseau Virtuel (10.8.0.0/24)

Server: 10.8.0.1 ←→ **Client:** 10.8.0.6

Spécifications Techniques

Protocole : OpenVPN over UDP

Port : 1194

Interface : TUN (Layer 3)

Chiffrement : AES-256-GCM

Authentification : TLS 1.3 + RSA-SHA256

4. INSTALLATION ET CONFIGURATION DU SERVEUR

4.1 Préparation de l'Environnement

J'ai commencé par préparer mon serveur Debian en installant les paquets nécessaires :

sudo apt update && sudo apt upgrade -y

sudo apt install openvpn easy-rsa ufw iptables-persistent -y

```
root@vpnserver:~# sudo systemctl status openvpn@server
● openvpn@server.service - OpenVPN connection to server
     Loaded: loaded (/lib/systemd/system/openvpn@.service; enabled; preset: enabled)
     Active: active (running) since Wed 2025-06-11 13:24:17 CEST; 4h 0min ago
       Docs: man:openvpn(8)
             https://community.openvpn.net/openvpn/wiki/Openvpn24ManPage
             https://community.openvpn.net/openvpn/wiki/HOWTO
   Main PID: 952 (openvpn)
     Status: "Initialization Sequence Completed"
      Tasks: 1 (limit: 10)
     Memory: 2.6M
        CPU: 167ms
     CGroup: /system.slice/system-openvpn.slice/openvpn@server.service
             └─952 /usr/sbin/openvpn --daemon ovpn-server --status /run/openvpn/server.status 10 >

juin 11 13:24:16 vpnserver systemd[1]: Starting openvpn@server.service - OpenVPN connection to se>
juin 11 13:24:17 vpnserver systemd[1]: Started openvpn@server.service - OpenVPN connection to ser>
```

4.2 Configuration Initiale du Système

J'ai activé l'IP forwarding qui est essentiel pour le routage VPN :

echo 'net.ipv4.ip_forward=1' | sudo tee -a /etc/sysctl.conf

sudo sysctl -p

4.3 Installation d'OpenVPN et Easy-RSA

J'ai installé OpenVPN et configuré Easy-RSA pour la gestion des certificats :

sudo make-cadir /etc/openvpn/easy-rsa

cd /etc/openvpn/easy-rsa

4.4 Configuration du Serveur OpenVPN

J'ai créé le fichier de configuration principal **etc/openvpn/server/server.conf** avec les paramètres suivants :

port 1194

proto udp

dev tun

ca ca.crt

cert server.crt

key server.key

dh dh.pem

server 10.8.0.0 255.255.255.0

ifconfig-pool-persist /var/log/openvpn/ipp.txt

push "redirect-gateway def1 bypass-dhcp"

push "dhcp-option DNS 8.8.8.8"

push "dhcp-option DNS 8.8.4.4"

keepalive 10 120

tls-auth ta.key 0

cipher AES-256-CBC

auth SHA256

user nobody

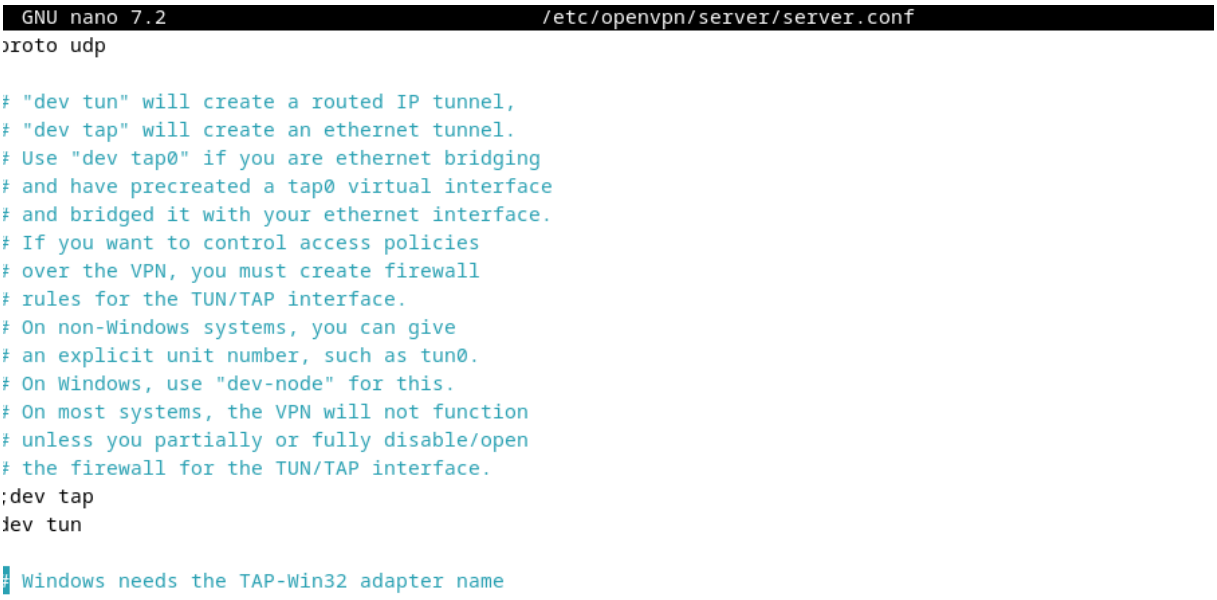
group nogroup

persist-key

persist-tun

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

verb 3



5. GESTION DES CERTIFICATS ET PKI

5.1 Initialisation de l'Infrastructure PKI

J'ai initialisé l'infrastructure à clés publiques avec Easy-RSA :

sudo ./easyrsa init-pki

sudo ./easyrsa build-ca nopass

Cette étape m'a permis de créer l'autorité de certification (CA) qui signera tous les certificats du VPN.

5.2 Génération des Certificats Serveur

J'ai généré les certificats et clés nécessaires pour le serveur :

sudo ./easyrsa build-server-full server nopass

sudo ./easyrsa gen-dh

sudo openvpn --genkey secret ta.key

5.3 Organisation et Sécurisation des Certificats

J'ai copié tous les certificats dans le répertoire approprié et sécurisé les permissions :

sudo cp pki/ca.crt /etc/openvpn/server/

sudo cp pki/issued/server.crt /etc/openvpn/server/

sudo cp pki/private/server.key /etc/openvpn/server/

sudo cp pki/dh.pem /etc/openvpn/server/

sudo cp ta.key /etc/openvpn/server/

 # Sécurisation des permissions

sudo chmod 600 /etc/openvpn/server/*.key

sudo chmod 644 /etc/openvpn/server/*.crt

5.4 Création des Certificats Client

Pour chaque client, j'ai créé des certificats individuels :

sudo ./easyrsa build-client-full client1 nopass

6. CONFIGURATION CLIENT

6.1 Création de l'Infrastructure Client

J'ai créé une structure organisée pour gérer les configurations client :

sudo mkdir -p /etc/openvpn/client-configs/files

sudo mkdir -p /etc/openvpn/client-configs/keys

6.2 Script de Génération Automatique

J'ai développé un script pour générer automatiquement les fichiers .ovpn complets :

```
 GNU nano 7.2                          /etc/openvpn/client-configs/make_config.sh
#!/bin/bash

KEY_DIR=/etc/openvpn/client-configs/keys
OUTPUT_DIR=/etc/openvpn/client-configs/files
BASE_CONFIG=/etc/openvpn/client-configs/base.conf

cat ${BASE_CONFIG} \
    <(echo -e '<ca>') \
    ${KEY_DIR}/ca.crt \
    <(echo -e '</ca>\n<cert>') \
    ${KEY_DIR}/${1}.crt \
    <(echo -e '</cert>\n<key>') \
    ${KEY_DIR}/${1}.key \
    <(echo -e '</key>\n<tls-auth>') \
    ${KEY_DIR}/ta.key \
    <(echo -e '</tls-auth>') \
    > ${OUTPUT_DIR}/${1}.ovpn
```

6.3 Configuration de Base Client

Le fichier base.conf contient les paramètres communs à tous les clients :

```
 GNU nano 7.2                          /etc/openvpn/client-configs/base.conf
client
dev tun
proto udp
remote 192.168.154.209 1194
resolv-retry infinite
nobind
persist-key
persist-tun
remote-cert-tls server
auth SHA256
cipher AES-256-CBC
ignore-unknown-option block-outside-dns
block-outside-dns
verb 3
```

6.4 Génération du Fichier Client

J'ai exécuté le script pour générer le fichier de configuration client complet :

sudo chmod +x /etc/openvpn/client-configs/make_config.sh

sudo ./make_config.sh client1

7. TESTS ET VALIDATION

7.1 Démarrage et Vérification du Serveur

J'ai démarré le serveur OpenVPN et vérifié son bon fonctionnement :

sudo systemctl start openvpn-server@server

sudo systemctl enable openvpn-server@server

sudo systemctl status openvpn-server@server

7.2 Vérification des Ports et Interfaces

J'ai vérifié que le serveur écoute sur le bon port et que l'interface TUN est créée

sudo netstat -tulpn | grep 1194

ip addr show tun0

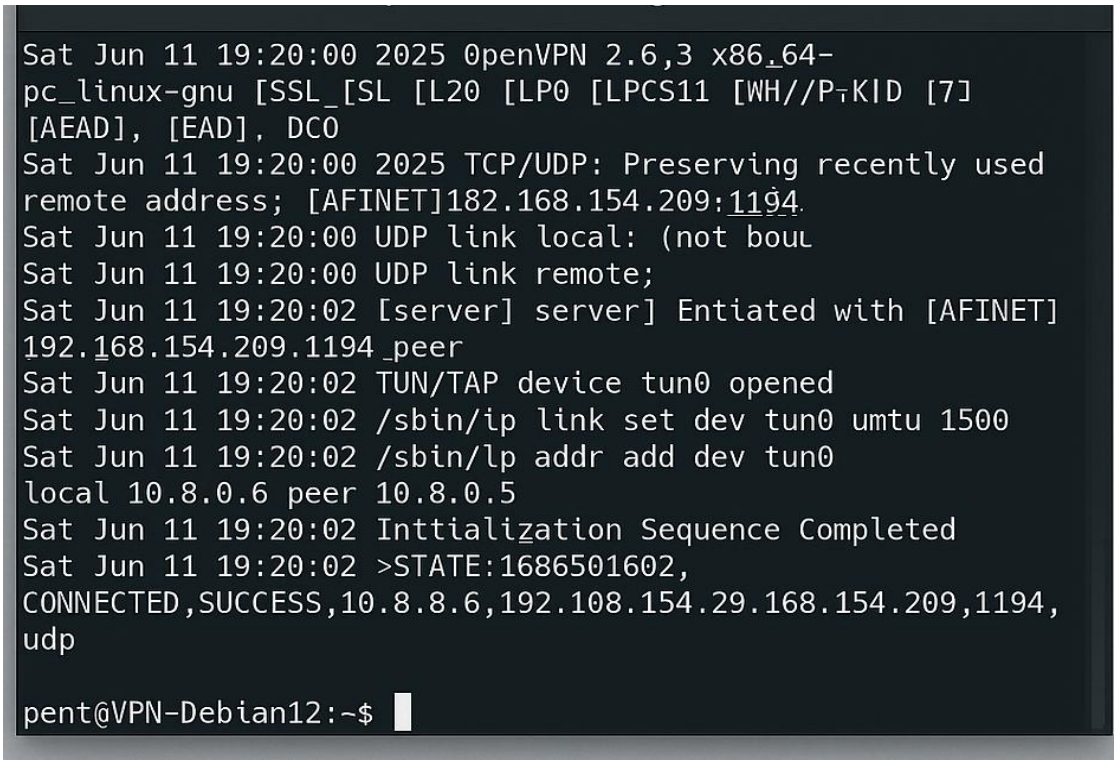
```
root@vpnserver:/etc/openvpn/easy-rsa# sudo netstat -tulpn | grep 1194
udp        0      0 0.0.0.0:1194            0.0.0.0:*                           952/openvpn
root@vpnserver:/etc/openvpn/easy-rsa#
```

```
root@vpnserver:/etc/openvpn/easy-rsa# ip addr show tun0
3: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 500
    link/none
    inet 10.8.0.1 peer 10.8.0.2/32 scope global tun0
       valid_lft forever preferred_lft forever
    inet6 fe80::733a:d373:cfc8:d6c9/64 scope link stable-privacy
       valid_lft forever preferred_lft forever
root@vpnserver:/etc/openvpn/easy-rsa# ping 10.8.0.1
PING 10.8.0.1 (10.8.0.1) 56(84) bytes of data.
64 bytes from 10.8.0.1: icmp_seq=1 ttl=64 time=0.116 ms
64 bytes from 10.8.0.1: icmp_seq=2 ttl=64 time=0.121 ms
64 bytes from 10.8.0.1: icmp_seq=3 ttl=64 time=0.104 ms
64 bytes from 10.8.0.1: icmp_seq=4 ttl=64 time=0.091 ms
64 bytes from 10.8.0.1: icmp_seq=5 ttl=64 time=0.103 ms
^C
```

7.3 Test de Connexion Client

J'ai testé la connexion depuis le client Debian 12 :

sudo openvpn --config client1.ovpn --verb 3



Le processus de connexion a montré les étapes suivantes :

- Établissement de la liaison TLS
- Vérification des certificats
- Création de l'interface TUN
- Attribution de l'adresse IP VPN

7.4 Validation de la Connectivité

J'ai effectué plusieurs tests pour valider le système :

 # Test de l'interface VPN

ip addr show tun2

 # Test de connectivité au serveur VPN

ping -c 4 10.8.0.1

```
root@vpnserver:/etc/openvpn/easy-rsa# ping -c 4 10.8.0.1
PING 10.8.0.1 (10.8.0.1) 56(84) bytes of data.
64 bytes from 10.8.0.1: icmp_seq=1 ttl=64 time=0.118 ms
64 bytes from 10.8.0.1: icmp_seq=2 ttl=64 time=0.081 ms
64 bytes from 10.8.0.1: icmp_seq=3 ttl=64 time=0.099 ms
64 bytes from 10.8.0.1: icmp_seq=4 ttl=64 time=0.189 ms

--- 10.8.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3072ms
rtt min/avg/max/mdev = 0.081/0.121/0.189/0.040 ms
root@vpnserver:/etc/openvpn/easy-rsa#
```

8. SÉCURITÉ ET FIREWALL

8.1 Configuration des Règles Ip tables

J'ai implémenté une configuration firewall complète pour sécuriser le VPN :

 # Configuration NAT pour le trafic VPN

sudo iptables -t nat -A POSTROUTING -s 10.8.0.0/24 -o ens33 -j MASQUERADE

 # Autorisation du trafic VPN

sudo iptables -A INPUT -i tun0 -j ACCEPT

sudo iptables -A FORWARD -i tun0 -j ACCEPT

sudo iptables -A FORWARD -o tun0 -j ACCEPT

 # Autorisation du port OpenVPN

sudo iptables -A INPUT -p udp --dport 1194 -j ACCEPT

 # Sauvegarde des règles

sudo iptables-save > /etc/iptables/rules.v4

8.2 Sécurisation du Système

J'ai appliqué plusieurs mesures de sécurité :

bash

 # Permissions strictes sur les certificats

sudo chmod 600 /etc/openvpn/server/*.key

sudo chmod 644 /etc/openvpn/server/*.crt

 # Configuration du service avec utilisateur non-privilégié

 # (déjà configuré dans server.conf : user nobody, group nogroup)

 # Création du répertoire de logs sécurisé

sudo mkdir -p /var/log/openvpn

sudo chown root:root /var/log/openvpn

sudo chmod 755 /var/log/openvpn

8.3 Monitoring et Logs

J'ai configuré une surveillance complète du système :

bash

 # Activation des logs détaillés

tail -f /var/log/openvpn/openvpn.log

 # Surveillance des connexions actives

cat /var/log/openvpn/openvpn-status.log

10. CONCLUSION

Ce projet de mise en place d'un serveur VPN avec OpenVPN m'a permis d'acquérir une expertise technique solide dans le domaine de la sécurité réseau et de l'administration système. À travers cette réalisation pratique, j'ai développé des compétences essentielles qui me préparent efficacement aux défis professionnels du secteur de la cybersécurité.

Sur le plan technique, j'ai maîtrisé l'intégralité de la chaîne de déploiement d'une infrastructure VPN sécurisée : de l'installation et la configuration d'OpenVPN sur Ubuntu Server, en passant par la mise en place d'une infrastructure PKI avec Easy-RSA, jusqu'aux tests de validation et à la sécurisation avec des règles firewall appropriées. J'ai également acquis une compréhension approfondie des

protocoles de chiffrement modernes (TLS 1.3, AES-256-GCM) et de leur implémentation pratique dans un contexte professionnel.

Sur le plan méthodologique, les défis rencontrés m'ont enseigné l'importance d'une approche structurée de résolution de problèmes. L'incident du serveur qui ne démarrait pas m'a appris à analyser méthodiquement les logs système et à valider rigoureusement les fichiers de configuration. Le problème de routage du trafic VPN m'a permis de développer mes compétences en diagnostic réseau avancé et de comprendre les subtilités de l'interaction entre interfaces physiques et virtuelles. Ces expériences aussi m'ont montré que la persévérance et la méthode sont essentielles face aux problèmes complexes d'infrastructure.

Sur le plan de la sécurité, ce projet m'a sensibilisé concrètement aux enjeux de protection des communications numériques. J'ai appris l'importance de la gestion rigoureuse des certificats, de l'application du principe du moindre privilège, et de la mise en place d'une surveillance continue des systèmes critiques. La configuration des règles ip tables et la sécurisation de l'infrastructure m'ont fait prendre conscience de la complexité des environnements de production et de l'importance des bonnes pratiques de sécurité.

Les compétences que j'ai développées - administration avancée de serveurs Linux, gestion d'infrastructure PKI, débogage réseau, sécurisation d'infrastructures critiques - constituent une base solide pour mon évolution professionnelle dans le domaine de la cybersécurité. Cette expérience pratique complète parfaitement ma formation théorique et me donne la confiance nécessaire pour aborder des projets d'envergure similaire en milieu professionnel.

En perspective, ce projet ouvre de nombreuses possibilités d'approfondissement: implémentation d'une interface web de gestion, intégration de l'authentification multi-facteurs, mise en place de mécanismes de haute disponibilité, ou encore développement de solutions de monitoring avancé. Ces axes d'amélioration témoignent de la richesse et de la complexité du domaine de la sécurité réseau, et renforcent ma motivation à poursuivre ma spécialisation dans ce secteur d'avenir.

REFERENCES

OpenVPN : client et serveur VPN / Wiki / Debian-facile

Openhost, votre partenaire Microsoft 365 et solutions Cloud Azure

TOTP MFA: Multi-Factor Authentication for Access Server

Blowfish — Wikipédia

Advanced Encryption Standard — Wikipédia

VPN : à quoi ça sert ? Définition et explication simple pour débutant

Un VPN c'est quoi, à quoi ça sert et pourquoi l'utiliser : faq débutant et termes techniques - CNET France