

TEST DE SOLUTIONS VPN

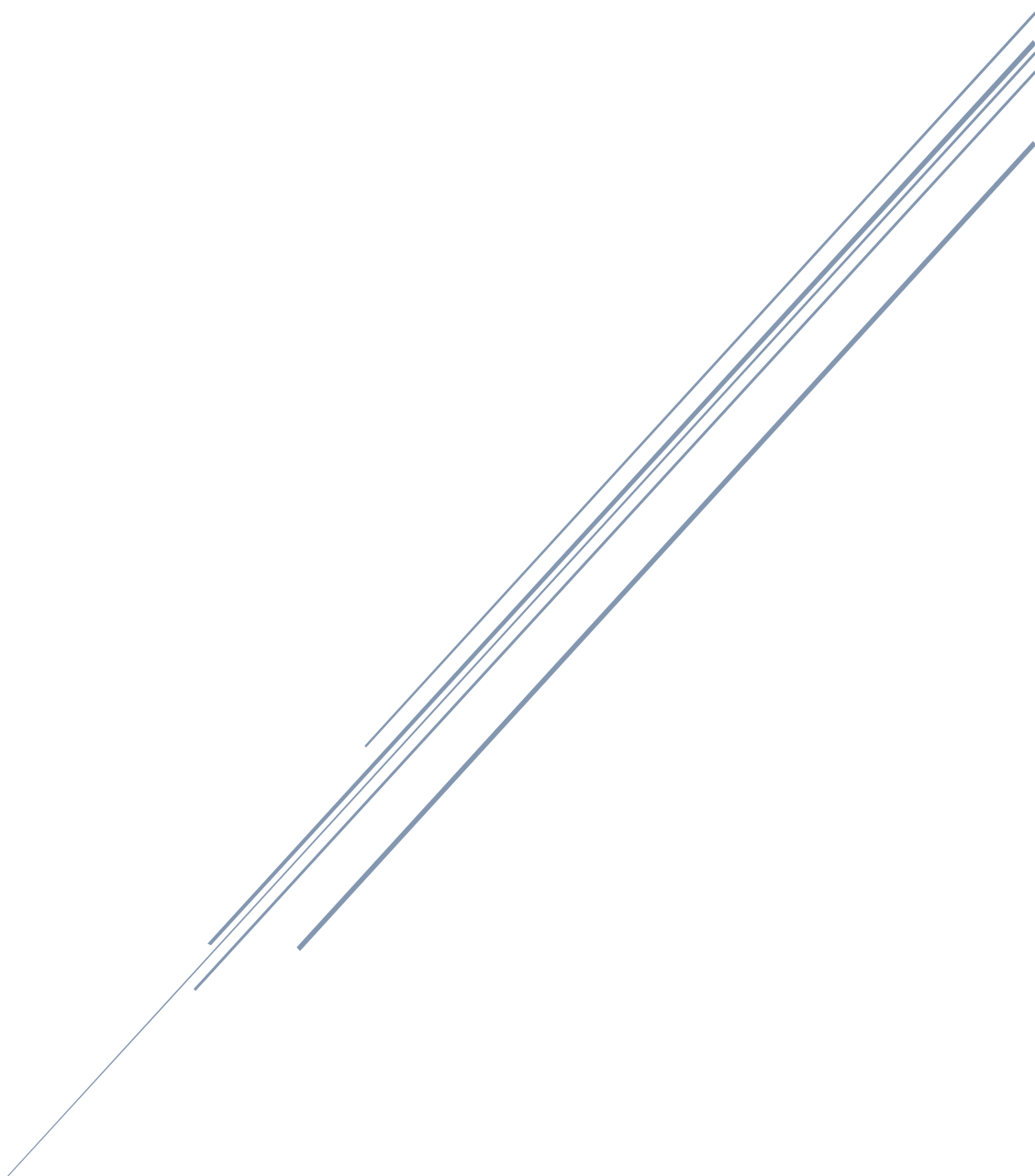


Table des matières

Algo VPN

1. Installation.....	2
Algo VPN	2
SSH	5
Ansible.....	6
Curl.....	6
WireGuard	6
2. Conclusion.....	9

Open VPN

1. Installation.....	11
2. Configuration	11
3. Conclusion.....	13

Final

1. Comparatif Final.....	15
---------------------------------	-----------

ALGO VPN

1. Installation

Algo VPN

a) Passer en root & upgrade

```
sudo su  
apt update && apt upgrade -y
```

b) Installer les dépendances nécessaires

```
apt install python3 python3-pip git -y
```

c) Installer git si besoin

```
apt install git
```

d) Cloner le dépôt

```
git clone https://github.com/trailofbits/algo
```

e) Accéder au répertoire Algo

```
cd algo
```

f) Installé et configurer l'environnement virtuel

```
"python3 -m pip install --upgrade pip virtualenv"  
"python3 -m virtualenv --python="$(command -v python3)"  
.env"source .env/bin/activate"
```

g) Installé les dépendances

```
python3 -m pip install -r requirements.txt
```

h) Lancé le script d'installation Algo

```
"./algo"
```

a) Installer sur un serveur Ubuntu existant

```
[Cloud prompt]
What provider would you like to use?
1. DigitalOcean
2. Amazon Lightsail
3. Amazon EC2
4. Microsoft Azure
5. Google Compute Engine
6. Hetzner Cloud
7. Vultr
8. Scaleway
9. OpenStack (DreamCompute optimised)
10. CloudStack (Exoscale optimised)
11. Linode
12. Install to existing Ubuntu latest LTS server (for more advanced users)

Enter the number of your desired provider
:
12
```

b) Sélectionnez-‘Y’ pour les 2 prochaines questions

```
Do you want macOS/iOS clients to enable "Connect On Demand" when connected to cellular networks?
[y/N]
:

TASK [Cellular On Demand prompt] *****
ok: [localhost]
[Wi-Fi On Demand prompt]
Do you want macOS/iOS clients to enable "Connect On Demand" when connected to Wi-Fi?
[y/N]
:
```

c) Sélectionnez

“HomeNet”

```
TASK [Wi-Fi On Demand prompt] *****
ok: [localhost]
[Trusted Wi-Fi networks prompt]
List the names of any trusted Wi-Fi networks where macOS/iOS clients should not use "Connect On Demand"
(e.g., your home network. Comma-separated value, e.g., HomeNet,OfficeWifi,AlgoWifi)
:
```

d) Sélectionnez 'Y' pour les 3 prochaines questions

```
ok: [localhost]
[Retain the PKI prompt]
Do you want to retain the keys (PKI)? (required to add users in the future, but less secure)
[y/N]
:

TASK [Retain the PKI prompt] *****
ok: [localhost]
[DNS adblocking prompt]
Do you want to enable DNS ad blocking on this VPN server?
[y/N]
:

TASK [DNS adblocking prompt] *****
ok: [localhost]
[SSH tunneling prompt]
Do you want each user to have their own account for SSH tunneling?
[y/N]
:
```

e) Entrez l'IP récupérer avec Curl

```
Enter the public IP address or domain name of your server: (IMPORTANT! This is used to verify the certificate)
[192.168.1.20]
:
```

f) Message final Algo

```
ok: [192.168.1.20] => {
  "msg": [
    [
      "\n#                                     Congratulations!           #\n",
      "\n#                                     Your Algo server is running.      #\n",
      "\n#      Config files and certificates are in the ./configs/ directory.    #\n",
      "\n#      Go to https://whoer.net/ after connecting                        #\n",
      "\n#      and ensure that all your traffic passes through the VPN.          #\n",
      "\n#      Local DNS resolver 172.22.212.26                                   #\n",
      ""
    ],
    "\n#      The p12 and SSH keys password for new users is 3t5BIQsbF          #\n",
    "\n#      The CA key password is JT8KUE4le@xNoHt@                          #\n",
    ""
  ]
}
```

SSH

i) Installé Open-SSH

`dpkg -s openssh-server`

`apt install openssh-server`

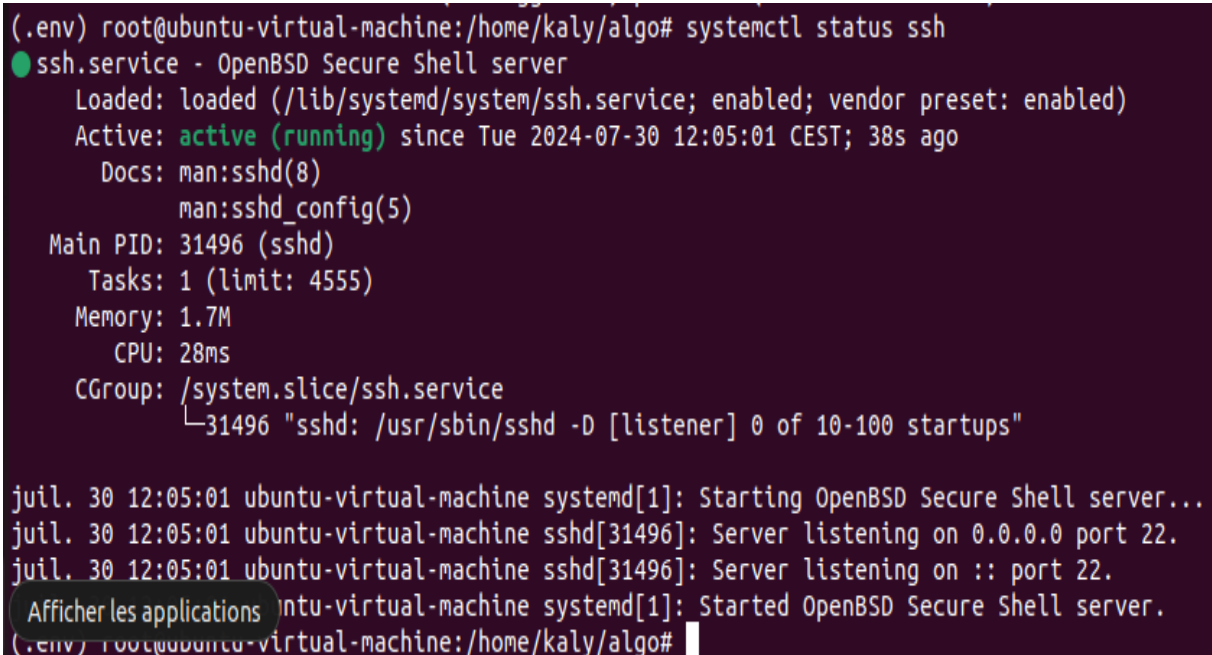
j) Configurer les ports & les autorisation

`"sudo nano /etc/ssh/sshd_config"`

- Port 22
- PermitRootLogin yes
- PubkeyAuthentication yes
- PasswordAuthentication no

k) Assurez-vous que le service est en cours

`"systemctl status ssh"`



```
(.env) root@ubuntu-virtual-machine:/home/kaly/algo# systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2024-07-30 12:05:01 CEST; 38s ago
     Docs: man:sshd(8)
           man:sshd_config(5)
  Main PID: 31496 (sshd)
    Tasks: 1 (limit: 4555)
   Memory: 1.7M
      CPU: 28ms
   CGroup: /system.slice/ssh.service
           └─31496 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

juil. 30 12:05:01 ubuntu-virtual-machine systemd[1]: Starting OpenBSD Secure Shell server...
juil. 30 12:05:01 ubuntu-virtual-machine sshd[31496]: Server listening on 0.0.0.0 port 22.
juil. 30 12:05:01 ubuntu-virtual-machine sshd[31496]: Server listening on :: port 22.
Afficher les applications ubuntu-virtual-machine systemd[1]: Started OpenBSD Secure Shell server.
(.env) root@ubuntu-virtual-machine:/home/kaly/algo#
```

l) Restart SHH

`'sudo systemctl restart ssh`

m) Surveillance de port

`sudo tcpdump -i ens33 port 22`

n) Forcer le mode promiscuous

```
sudo ip link set ens33 promisc on
```

Ansible

o) Installer Ansible

```
sudo apt install ansible
```

p) Configurer le fichier Ansible config.cfg

```
sudo nano config.cfg
```

```
« ansible_ssh_port: 22 »
```

Curl

a) Installer Curl










```
sudo apt install curl
```

b) Noté L'IP

```
Curl ifconfig.me
```

c) Configurer Algo VPN & PfSense

Derrière un routeur PfSense, vous devez vous assurer que les ports 500 et 4500 sont ouverts et redirigés vers votre serveur VPN.

<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	WAN	UDP	*	*	WAN address	4500 (IPsec NAT-T)	192.168.1.20	4500 (IPsec NAT-T)	NAT for IPsec IKEv2	  
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	WAN	UDP	*	*	WAN address	500 (ISAKMP)	192.168.1.20	500 (ISAKMP)	NAT for IPsec IKEv2	  
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	WAN	TCP	*	*	WAN address	2222	192.168.1.20	2222		  

WireGuard

a) Installez WireGuard

```
sudo apt-get install wireguard
```

b) Générez la clé privée du serveur

```
wg genkey | sudo tee /etc/wireguard/server_private.key
```

c) Générez la clé publique du serveur

```
sudo cat /etc/wireguard/server_private.key | wg pubkey | sudo tee  
/etc/wireguard/server_public.key
```

d) Générez la clé privée du client :

```
wg genkey | sudo tee /etc/wireguard/client_private.key
```

e) Générez la clé publique du client :

```
sudo cat /etc/wireguard/client_private.key | wg pubkey | sudo tee  
/etc/wireguard/client_public.key
```

f) Créez le fichier de configuration du serveur :

```
sudo nano /etc/wireguard/wg0.conf
```

g) Ajoutez le contenu suivant au fichier wg0.conf

```
[Interface]
```

```
PrivateKey = <contenu de /etc/wireguard/server_private.key>
```

```
Address = 10.49.0.1/24
```

```
ListenPort = 51820
```

```
[Peer]
```

```
PublicKey = <contenu de /etc/wireguard/client_public.key>
```

```
AllowedIPs = 10.49.0.2/32
```

h) Créez le fichier de configuration du client

```
sudo nano /home/kaly/algo/configs/client_phone.conf
```

i) Ajoutez le contenu suivant au fichier client_phone.conf :

```
[Interface]
```

```
PrivateKey = <contenu de /etc/wireguard/client_private.key>
```

```
Address = 10.49.0.2/24
```


DNS = 8.8.8.8

[Peer]

PublicKey = <contenu de /etc/wireguard/server_public.key>

Endpoint = <adresse IP publique du serveur>:51820

AllowedIPs = 0.0.0.0/0

PersistentKeepalive = 25

j) Démarrez le service WireGuard :

```
sudo systemctl start wg-quick@wg0
```

k) Activez le service WireGuard au démarrage

```
sudo systemctl enable wg-quick@wg0
```

l) Vérifiez l'état de WireGuard

```
sudo wg show
```

m) Activez le forwarding IP

```
echo 1 | sudo tee /proc/sys/net/ipv4/ip_forward
```

Service WireGuard

```
sudo systemctl stop wg-quick@wg0
```

```
sudo systemctl status wg-quick@wg0
```

```
sudo systemctl start wg-quick@wg0
```

2. Conclusion

Algo VPN, se présente comme une solution de déploiement **VPN** moderne et sécurisée. Cependant, l'expérience utilisateur peut varier considérablement, en particulier lors de l'installation et de la configuration initiale.

Complexité d'Installation

L'installation d'**Algo VPN** peut s'avérer complexe et difficile pour de nombreux utilisateurs. Les étapes d'installation nécessitent souvent une compréhension approfondie des systèmes **Linux**, des lignes de commande, et des concepts de réseaux.

Sécurité et Performance

Malgré les difficultés d'installation, **Algo VPN** offre des avantages en termes de sécurité. Il utilise des protocoles modernes comme **WireGuard** et **IPsec** avec **IKEv2**, qui sont reconnus pour leur robustesse et leurs performances. La configuration par défaut est orientée vers la sécurité, ce qui est un point positif pour ceux qui parviennent à le mettre en place.

Flexibilité et Personnalisation

Algo VPN permet une grande flexibilité dans le choix du fournisseur d'hébergement et offre des options de personnalisation. Cependant, cette flexibilité peut aussi contribuer à la complexité de l'installation, car elle nécessite une compréhension des différentes plateformes cloud et de leurs spécificités.

Support et Documentation

L'expérience utilisateur pourrait être grandement améliorée par une documentation plus détaillée et accessible, ainsi que par un meilleur support pour les utilisateurs rencontrant des difficultés lors de l'installation.

Conclusion Générale

Algo VPN est une solution qui promet une sécurité élevée et des performances optimales, mais au prix d'un processus d'installation qui peut être décourageant pour de nombreux utilisateurs. Bien qu'il offre des avantages en termes de sécurité et de flexibilité, il n'est peut-être pas la

solution la plus adaptée pour ceux qui recherchent une expérience "plug-and-play" ou qui n'ont pas de solides connaissances techniques.

Pour les utilisateurs prêts à investir du temps dans l'apprentissage et la configuration, Algo VPN peut offrir une solution VPN robuste et personnalisable.

OPEN VPN

1. Installation

a) Installer OpenVPN & OpenSSH

```
"sudo apt install openvpn openssh-server"
```

2. Configuration

Open VPN

a) Télécharger le script d'installation

```
"wget https://git.io/vpn -O openvpn-install.sh"
```

b) Rendre le script exécutable

```
"chmod +x openvpn-install.sh"
```

c) Exécuter le script d'installation

```
"sudo ./openvpn-install.sh"
```

Suivre les instructions à l'écran pour configurer OpenVPN

Easy-RSA

d) Installer Easy-RSA

```
"sudo apt install easy-rsa"
```

e) Configuration de Easy-RSA

```
"mkdir ~/easy-rsa  
ln -s /usr/share/easy-rsa/* ~/easy-rsa/  
cd ~/easy-rsa"
```

f) Initialisation de l'infrastructure PKI

```
"./easyrsa init-pki"
```

g) Création de l'autorité de certification (CA)

```
"./easyrsa build-ca nopass"
```

h) Génération de la demande de certificat pour le serveur

```
"/easyrsa gen-req server nopass"
```

i) Signature de la demande de certificat du serveur

```
"/easyrsa sign-req server server"
```

j) Copie des fichiers générés dans le répertoire OpenVPN

```
"sudo cp ~/easy-rsa/pki/ca.crt /etc/openvpn/server/  
sudo cp ~/easy-rsa/pki/issued/server.crt /etc/openvpn/server/  
sudo cp ~/easy-rsa/pki/private/server.key /etc/openvpn/server/  
sudo cp ~/easy-rsa/pki/dh.pem /etc/openvpn/server/  
sudo cp ~/easy-rsa/ta.key /etc/openvpn/server/"
```

k) Création du fichier de configuration du serveur

```
"sudo nano /etc/openvpn/server.conf"
```

l) Ajoutez le contenu suivant dans le fichier de configuration

```
port 1194  
proto udp  
dev tun  
ca /etc/openvpn/server/ca.crt  
cert /etc/openvpn/server/server.crt  
key /etc/openvpn/server/server.key  
dh /etc/openvpn/server/dh.pem  
server 10.8.0.0 255.255.255.0  
ifconfig-pool-persist /etc/openvpn/server/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  
cipher AES-256-CBC  
user nobody  
group nogroup  
persist-key  
persist-tun  
status /etc/openvpn/server/openvpn-status.log  
verb 3
```

m) Vérification des permissions des fichiers

```
sudo chown -R root:root /etc/openvpn/server/  
sudo chmod 600 /etc/openvpn/server/server.key
```

Pare-Feu

n) Autoriser le trafic OpenVPN port UDP 1194 & Open

```
sudo ufw allow 1194/udp  
sudo ufw allow 22/tcp
```

Vérification des services

o) Vérifier le statut d'OpenVPN et OpenSSH

```
sudo systemctl status openvpn@server  
sudo systemctl status ssh
```

Exécuter à nouveau le script pour ajouter un nouveau client

“sudo ./openvpn-install.sh”

```
● openvpn@server.service - OpenVPN connection to server  
Loaded: loaded (/lib/systemd/system/openvpn@.service; enabled; vendor preset: enabled)  
Active: active (running) since Wed 2024-07-31 11:05:12 CEST; 11min ago  
Docs: man:openvpn(8)  
       https://community.openvpn.net/openvpn/wiki/Openvpn24ManPage  
       https://community.openvpn.net/openvpn/wiki/HOWTO  
Main PID: 955 (openvpn)  
Status: "Initialization Sequence Completed"  
Tasks: 1 (limit: 4554)  
Memory: 3.3M  
CPU: 54ms  
CGroup: /system.slice/system-openvpn.slice/openvpn@server.service  
        └─955 /usr/sbin/openvpn --daemon ovpn-server --status /run/openvpn/server.status 10 --cd /etc/openvpn --script-security
```

```
juil. 31 11:05:12 ubuntu22-virtual-machine ovpn-server[955]: Could not determine IPv4/IPv6 protocol. Using AF_INET  
juil. 31 11:05:12 ubuntu22-virtual-machine ovpn-server[955]: Socket Buffers: R=[212992->212992] S=[212992->212992]  
juil. 31 11:05:12 ubuntu22-virtual-machine ovpn-server[955]: UDPv4 link local (bound): [AF_INET][undef]:1194  
juil. 31 11:05:12 ubuntu22-virtual-machine ovpn-server[955]: UDPv4 link remote: [AF_UNSPEC]  
juil. 31 11:05:12 ubuntu22-virtual-machine ovpn-server[955]: GID set to nogroup  
juil. 31 11:05:12 ubuntu22-virtual-machine ovpn-server[955]: UID set to nobody  
juil. 31 11:05:12 ubuntu22-virtual-machine ovpn-server[955]: MULTI: multi_init called, r=256 v=256  
juil. 31 11:05:12 ubuntu22-virtual-machine ovpn-server[955]: IFCONFIG POOL IPv4: base=10.8.0.4 size=62  
juil. 31 11:05:12 ubuntu22-virtual-machine ovpn-server[955]: IFCONFIG POOL LIST  
juil. 31 11:05:12 ubuntu22-virtual-machine ovpn-server[955]: Initialization Sequence Completed
```

3. Conclusion

Open VPN est l'un des protocoles VPN les plus populaires et les plus utilisés dans le monde. Il a été développé pour offrir une solution VPN flexible, sécurisée et compatible avec une grande variété de plateformes.

Sécurité

OpenVPN est reconnu pour sa robustesse en matière de sécurité. Il utilise des protocoles de chiffrement de pointe tels que **AES-256**, ainsi que des certificats pour l'authentification. Cette combinaison offre une protection solide contre les interceptions et les attaques.

Flexibilité

L'un des principaux avantages **d'OpenVPN** est sa flexibilité. Il peut être configuré pour fonctionner sur presque toutes les plateformes, y compris Windows, macOS, Linux, Android et iOS. **OpenVPN** peut être utilisé en mode **TCP** ou **UDP**, ce qui permet de s'adapter à différents types de réseaux et de besoins.

Performance

En termes de performance, OpenVPN est généralement plus lent que les protocoles plus récents comme WireGuard. Cela est dû à sa complexité et à la quantité de traitement nécessaire pour le chiffrement et le déchiffrement des données. Cependant, OpenVPN reste performant et fiable pour la plupart des usages courants.

Support et compatibilité

OpenVPN bénéficie d'un large support et est compatible avec de nombreux dispositifs et systèmes d'exploitation. Il est également supporté par de nombreux fournisseurs de services VPN commerciaux, ce qui en fait un choix populaire pour les entreprises et les particuliers.

Conclusion Générale

OpenVPN est une solution VPN éprouvée et fiable, offrant une grande flexibilité et une sécurité robuste, il reste une option solide pour ceux qui ont besoin d'une solution VPN flexible et compatible avec une large gamme de dispositifs. Pour les utilisateurs recherchant une performance maximale et une installation simplifiée, des alternatives comme WireGuard peuvent être plus appropriées. Cependant, pour ceux qui privilégient la sécurité et la flexibilité, OpenVPN demeure un choix de premier ordre.

1. Comparatif Final

Protocole

- Algo VPN utilise WireGuard comme protocole VPN principal.
- OpenVPN utilise son propre protocole propriétaire.

Performance

- Algo VPN, grâce à l'utilisation de WireGuard, offre généralement de meilleures performances en termes de vitesse et de latence.
- OpenVPN, bien que fiable, est généralement plus lent que WireGuard.

Facilité d'utilisation

- Algo VPN vise à simplifier le déploiement de VPN, mais peut être complexe pour les utilisateurs novices.
- OpenVPN dispose d'une documentation extensive et d'un large support communautaire.

Sécurité

- Les deux offrent un haut niveau de sécurité.
- Algo VPN bénéficie de la sécurité moderne de WireGuard.
- OpenVPN a une longue histoire de sécurité éprouvée.

Flexibilité

- Algo VPN offre une bonne flexibilité pour le déploiement sur différentes plateformes cloud.
- OpenVPN offre plus d'options de configuration et de personnalisation.

Compatibilité

- Algo VPN, utilisant WireGuard, a une bonne compatibilité avec les systèmes modernes.
- OpenVPN a une compatibilité plus large, notamment avec des systèmes plus anciens.

En résumé

Algo VPN, basé sur WireGuard, offre des performances supérieures pour le déploiement sur le cloud, ce qui le rend attrayant pour les utilisateurs recherchant une solution moderne et rapide. OpenVPN, en revanche, reste un choix solide pour ceux qui ont besoin d'une flexibilité maximale, d'une large compatibilité, et qui apprécient sa longue histoire de fiabilité et de sécurité éprouvée.

Le choix entre Algo VPN et OpenVPN dépendra donc des besoins spécifiques de l'utilisateur : pour une solution rapide et moderne orientée cloud, Algo VPN est préférable, tandis que pour une flexibilité maximale et une compatibilité étendue, OpenVPN reste un excellent choix.

Fin

Malgré tous mes efforts pour configurer et installer les solutions VPN, je n'ai pas réussi à établir une connexion fonctionnelle au réseau créé avec Algo VPN. Toutes les étapes de configuration ont été suivies avec soin, mais le résultat final n'a pas permis d'accéder au réseau.

En conséquence, je n'ai pas pu réaliser les tests de vitesse et les comparaisons pratiques initialement prévus.

Les conclusions et les comparaisons dans ce document reposent donc sur des recherches et des données théoriques plutôt que sur des résultats pratiques.