

Fiche Pratique N°20a : Configurer WireGuard sur YunoHost pour un accès VPN sécurisé V1.0

Objectif : Installer et configurer WireGuard sur votre serveur YunoHost pour créer un tunnel VPN sécurisé, permettant un accès chiffré à vos services locaux depuis l'extérieur.

Public visé : Intermédiaire (quelques bases en ligne de commande sont utiles, mais la configuration est guidée).

Temps estimé : 30 à 60 minutes

Niveau de difficulté : ★★★☆☆ (Moyen)

1. Pourquoi installer WireGuard en premier ?

WireGuard est la fondation de votre infrastructure sécurisée. C'est le tunnel VPN qui permet d'accéder à vos services locaux (AdGuard Home, SearXNG, Nextcloud, etc.) depuis l'extérieur sans les exposer directement sur Internet.

Les bénéfices concrets :

- **Accès sécurisé à distance :** Accédez à vos services depuis l'extérieur, en 4G ou sur un Wi-Fi public, comme si vous étiez chez vous.
- **Chiffrement intégral :** Tout le trafic est chiffré. Vos identifiants et données sont protégés contre les interceptions.
- **Architecture sécurisée :** Seul le port UDP de WireGuard est exposé sur Internet, protégeant vos autres services.

Fiche Pratique N°20a : Configurer WireGuard sur YunoHost pour un accès VPN sécurisé V1.0

- **Simplicité et rapidité** : WireGuard est très léger, plus rapide qu'OpenVPN, et intégré au noyau Linux.

Cas d'usage typique :

- Vous êtes en déplacement et voulez accéder à vos services locaux en toute sécurité.
- Vous voulez administrer votre serveur à distance sans ouvrir de ports supplémentaires.
- Vous cherchez à protéger vos communications sur un réseau public (café, hôtel).
- Vous souhaitez créer un tunnel privé pour rendre accessibles AdGuard Home, SearXNG et d'autres services depuis l'extérieur (les fiches suivantes s'appuieront sur ce tunnel).

2. Architecture cible et prérequis

Architecture cible :

Élément	IP / Domaine	Rôle
Serveur	192.168.1.200 (IP locale fixe)	Serveur WireGuard, héberge les services locaux
Réseau VPN	10.10.10.0/24	Réseau privé pour les clients VPN
Client Linux	10.10.10.2/32	Poste de travail Linux (Mint, Ubuntu, etc.)
Client Android	10.10.10.3/32	Téléphone mobile

Prérequis :

- Un serveur avec YunoHost installé et une IP fixe sur votre réseau local (ex: 192.168.1.200).
- Un accès administrateur (via SSH ou interface web) sur ce serveur.

Fiche Pratique N°20a : Configurer WireGuard sur YunoHost pour un accès VPN sécurisé V1.0

- Un ordinateur client (Linux) pour la configuration et les tests.
 - Un téléphone Android pour tester l'accès distant (optionnel).
 - **Adresse IPv4 Full Stack** : Votre box Internet doit disposer d'une adresse IPv4 publique et dédiée (voir Fiche N°20, section 4.5). Sans cela, votre client WireGuard ne pourra pas établir de connexion avec votre serveur depuis l'extérieur.
 - Votre box Internet doit permettre la redirection de ports (fonction "Port Forwarding").
-

3. Installation et configuration du serveur

3.1. Installer WireGuard sur le serveur

Depuis un terminal sur votre serveur YunoHost, exécutez :

```
bash  
sudo apt update  
sudo apt install wireguard wireguard-tools -y
```

3.2. Générer les clés du serveur

```
bash  
cd /etc/wireguard  
sudo umask 077  
sudo wg genkey | sudo tee privatekey | sudo wg pubkey | sudo tee publickey
```

Ces deux fichiers (`privatekey` et `publickey`) sont essentiels. Conservez-les précieusement.

3.3. Créer le fichier de configuration `/etc/wireguard/wg0.conf`

```
bash
```

Fiche Pratique N°20a : Configurer WireGuard sur YunoHost pour un accès VPN sécurisé V1.0

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

Copiez le contenu suivant en remplaçant :

- [CONTENU_DU_FICHER_PRIVATEKEY] par le contenu de votre fichier privatekey.
- enp3s0 par le nom de votre interface réseau principale (trouvez-le avec la commande `ip a`).

```
[Interface]
```

```
Address = 10.10.10.1/24
```

```
ListenPort = 51820
```

```
PrivateKey = [CONTENU_DU_FICHER_PRIVATEKEY]
```

```
PostUp = iptables -A FORWARD -i wg0 -j ACCEPT; iptables -t nat -A POSTROUTING -o enp3s0 -j MASQUERADE
```

```
PostDown = iptables -D FORWARD -i wg0 -j ACCEPT; iptables -t nat -D POSTROUTING -o enp3s0 -j MASQUERADE
```

3.4. Activer le routage IP

```
bash
```

```
sudo sysctl -w net.ipv4.ip_forward=1
```

```
sudo sysctl -w net.ipv4.conf.wg0.forwarding=1
```

Rendez la modification permanente en ajoutant la ligne suivante à la fin du fichier `/etc/sysctl.conf` :

```
bash
```

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

3.5. Configurer le pare-feu (nftables)

YunoHost utilise `nftables`. Ouvrez le port UDP `51820` et autorisez le trafic de l'interface WireGuard :

```
bash
```

```
sudo nft add rule inet filter input udp dport 51820 accept
```

Fiche Pratique N°20a : Configurer WireGuard sur YunoHost pour un accès VPN sécurisé V1.0

```
sudo nft add rule inet filter input iifname wg0 accept
```

Sauvegardez les règles pour qu'elles persistent après un redémarrage :

```
bash
```

```
sudo nft list ruleset | sudo tee /etc/nftables.conf > /dev/null
```

3.6. Ajouter les clients (peers)

Pour chaque client (Linux, Android), vous devez ajouter une section [Peer] dans le fichier `wg0.conf`. Nous le ferons lors de la configuration des clients (partie 4).

3.7. Démarrer et activer le tunnel

```
bash
```

```
sudo wg-quick up wg0
```

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

Le tunnel est maintenant actif sur le serveur.

4. Configuration du client Linux (poste de travail)

4.1. Installer WireGuard sur le client

```
bash
```

```
sudo apt update
```

```
sudo apt install wireguard wireguard-tools -y
```

4.2. Générer les clés du client

```
bash
```

Fiche Pratique N°20a : Configurer WireGuard sur YunoHost pour un accès VPN sécurisé V1.0

```
cd /etc/wireguard
sudo umask 077
sudo wg genkey | sudo tee client-privatekey | sudo wg pubkey | sudo tee client-publickey
```

4.3. Ajouter le client au serveur

Sur le serveur, ouvrez le fichier `/etc/wireguard/wg0.conf` et ajoutez une nouvelle section `[Peer]` :

```
ini

[Peer]
PublicKey = [CONTENU_DU_FICHIER_client-publickey_DU_CLIENT]
AllowedIPs = 10.10.10.2/32
```

Remplacez `[CONTENU_DU_FICHIER_client-publickey_DU_CLIENT]` par le contenu de votre fichier `client-publickey`. Rechargez la configuration du serveur :

```
bash

sudo wg-quick down wg0
sudo wg-quick up wg0
```

4.4. Créer le fichier de configuration du client (`/etc/wireguard/wg0.conf`)

Sur votre machine cliente, créez le fichier :

```
bash

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

Copiez le contenu suivant en remplaçant :

- `[CLE_PRIVÉE_DU_CLIENT]` par le contenu de votre fichier `client-privatekey`.
- `[CLE_PUBLIQUE_DU_SERVEUR]` par le contenu du fichier `publickey` du serveur.
- `votre-domaine-ou-ip-publique.fr` par votre nom de domaine ou l'IP publique de votre box.

```
ini

[Interface]
```

Fiche Pratique N°20a : Configurer WireGuard sur YunoHost pour un accès VPN sécurisé V1.0

Address = 10.10.10.2/32

PrivateKey = [CLE_PRIVÉE_DU_CLIENT]

DNS = 192.168.1.200

MTU = 1450

[Peer]

PublicKey = [CLE_PUBLIQUE_DU_SERVEUR]

AllowedIPs = 0.0.0.0/0, ::/0

Endpoint = votre-domaine-ou-ip-publique.fr:51820

PersistentKeepalive = 25

Note sur l'Endpoint : C'est l'adresse publique que le client utilisera pour joindre le serveur depuis Internet. Elle doit être accessible depuis l'extérieur. Assurez-vous que vous avez bien une IPv4 Full Stack pour que cette étape fonctionne.

4.5. Démarrer et tester le tunnel

```
bash
```

```
sudo wg-quick up wg0
```

```
ping 10.10.10.1
```

```
ping 192.168.1.200
```

Si les deux pings fonctionnent, le tunnel est opérationnel.

5. Configuration du client Android (téléphone)

5.1. Installer l'application

Téléchargez et installez **WireGuard** depuis [F-Droid](#) ou le Google Play Store.

5.2. Ajouter le client au serveur

Fiche Pratique N°20a : Configurer WireGuard sur YunoHost pour un accès VPN sécurisé V1.0

Comme pour le client Linux, ajoutez une nouvelle section [Peer] dans le fichier /etc/wireguard/wg0.conf du serveur :

```
ini

[Peer]
PublicKey = [CLE_PUBLIQUE_DU_CLIENT_ANDROID]
AllowedIPs = 10.10.10.3/32
```

Rechargez la configuration :

```
bash

sudo wg-quick down wg0
sudo wg-quick up wg0
```

5.3. Créer la configuration sur le téléphone

1. Ouvrez l'application WireGuard.
2. Appuyez sur le bouton "+".
3. Choisissez **"Créer à partir de zéro"**.

Saisissez les informations suivantes :

Interface :

- **Adresse IP** : 10.10.10.3/32
- **DNS** : 192.168.1.200

Peer :

- **Clé publique** : [CLE_PUBLIQUE_DU_SERVEUR]
- **Endpoint** : votre-domaine-ou-ip-publique.fr:51820
- **Adresses IP autorisées** : 0.0.0.0/0, ::/0
- **Keep Alive** : 25 secondes

5.4. Activer et tester

Fiche Pratique N°20a : Configurer WireGuard sur YunoHost pour un accès VPN sécurisé V1.0

Activez le tunnel. Puis, depuis un navigateur sur votre téléphone, testez l'accès à votre serveur (par exemple, <http://192.168.1.200>). Si la page de YunoHost s'affiche, le VPN est fonctionnel.

6. Rendre le VPN accessible depuis Internet

Prérequis indispensable : Cette étape nécessite une **IPv4 Full Stack** (voir Fiche N°20, section 4.5).

Pour que vos clients puissent se connecter depuis l'extérieur, vous devez configurer votre box Internet.

6.1. Configurer la redirection de port sur votre box

1. Connectez-vous à l'interface de votre box (généralement <http://192.168.1.1> ou <http://192.168.0.1>).
2. Cherchez la section "**Redirection de ports**" (Port Forwarding, NAT).
3. Créez une règle :
 - **Nom** : WireGuard
 - **Port externe** : 51820 (ou un autre port si vous le modifiez)
 - **Protocole** : UDP
 - **IP interne** : 192.168.1.200 (l'IP de votre serveur)
 - **Port interne** : 51820 (doit correspondre au ListenPort défini dans votre wg0.conf)

Note : Si vous modifiez le port, pensez à l'ajuster dans la configuration du serveur et de tous les clients.

6.2. Vérifier l'ouverture du port

Fiche Pratique N°20a : Configurer WireGuard sur YunoHost pour un accès VPN sécurisé V1.0

Sur le serveur, vérifiez que le port est bien en écoute :

```
bash
```

```
sudo ss -ulpn | grep 51820
```

Le résultat doit afficher 0.0.0.0:51820.

7. Dépannage et vérifications

Problèmes fréquents :

Problème	Solution
Pas de handshake	Vérifier l'Endpoint (IP/domaine) et que le port 51820/UDP est bien redirigé sur la box. Vérifier que vous avez bien une IPv4 Full Stack.
Ping vers 10.10.10.1 échoue	Vérifier les règles nftables sur le serveur (sudo nft list ruleset).
Sites externes non accessibles	Vérifier AllowedIPs = 0.0.0.0/0, ::0 sur le client.
DNS ne fonctionne pas	Vérifier DNS = 192.168.1.200 sur le client.
Connexion 4G bloque le VPN	Activer PersistentKeepalive = 25 sur le client.

Commandes utiles :

Action	Commande (Serveur)	Commande (Client)
Démarrer	sudo wg-quick up wg0	sudo wg-quick up wg0

Fiche Pratique N°20a : Configurer WireGuard sur YunoHost pour un accès VPN sécurisé V1.0

Action	Commande (Serveur)	Commande (Client)
Arrêter	<code>sudo wg-quick down wg0</code>	<code>sudo wg-quick down wg0</code>
Statut	<code>sudo wg show</code>	<code>sudo wg show</code>
Logs	<code>sudo journalctl -u wg-quick@wg0 -f</code>	<code>sudo journalctl -u wg-quick@wg0 -f</code>

8. Sécurité et maintenance

8.1. Sauvegarder les configurations

- **Serveur** : `/etc/wireguard/wg0.conf`, `/etc/nftables.conf`
- **Clients** : Fichiers `.conf` exportés

8.2. Surveiller les logs

bash

```
sudo journalctl -u wg-quick@wg0 -n 50 --no-pager
```

8.3. Mettre à jour WireGuard

bash

```
sudo apt update
```

```
sudo apt upgrade wireguard wireguard-tools -y
```

C'est terminé ! Vous avez désormais un VPN WireGuard opérationnel, qui servira de base sécurisée pour les fiches suivantes (SearXNG et AdGuard Home).