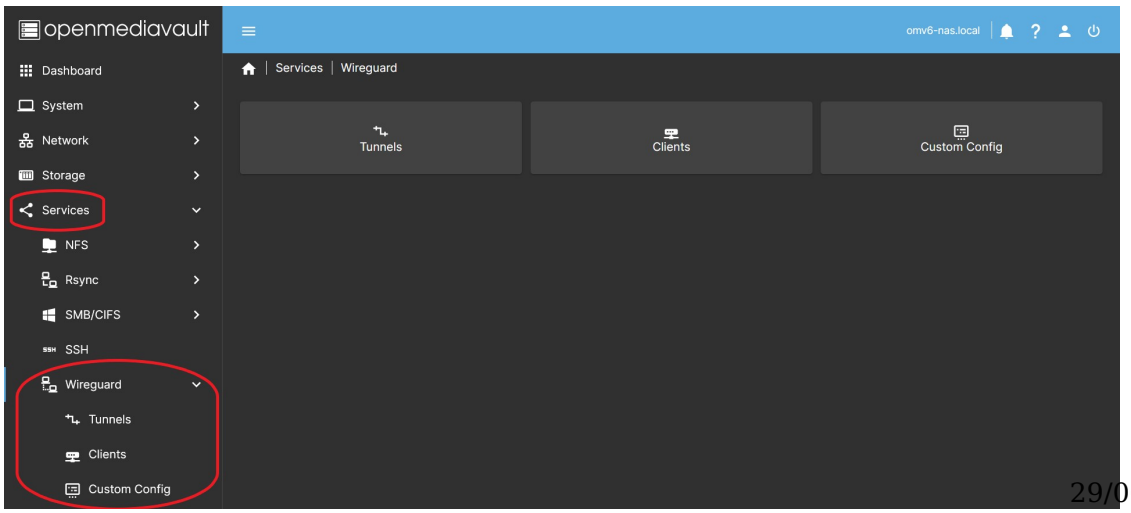
Plugin Wireguard pour OMV7

Plugin Wireguard pour OMV7

Sous **Services** > **Wireguard**



Résumé

- L'objectif principal de Wireguard est de faciliter la connexion sécurisée de deux appareils sur Internet.
- Openmediavault-wireguard intègre dans l'interface OMV via les onglets Tunnels et Clients la possibilité de générer un ou plusieurs réseaux de connexion VPN Wireguard cryptés point à site en deux clics.
    - La connexion point à site de Wireguard permet d'accéder à l'ensemble du réseau où se trouve le serveur.
        - Il s'agit de la configuration par défaut du plugin.
        - Vous pourrez accéder à tous vos dossiers partagés et à tous les services que vous avez configurés sur votre réseau local comme si vous y étiez.
        - Par défaut, tout le trafic client sera transmis via la connexion VPN (elle est configurable), garantissant ainsi la confidentialité via la connexion cryptée. Vous pouvez être connecté à un réseau Wi-Fi public et naviguer avec la sécurité que personne ne voit ce que vous faites.
- L'onglet Custom Config vous permet d'effectuer des configurations en fonction de besoins spécifiques. Vous pouvez utiliser cet onglet si vous devez connecter le serveur à un service VPN Wireguard externe, ou si vous pouvez implémenter n'importe quelle topologie de réseau Wireguard.
    - La connexion point à point permet la connexion entre deux serveurs, communiquant uniquement entre eux. Par exemple pour faire des sauvegardes à distance.
    - Le site à site fournit une connexion entre deux réseaux afin que toute adresse IP d'un réseau local puisse communiquer avec n'importe quelle adresse IP d'un autre réseau local.
    - Vous pouvez implémenter toute autre typologie dont vous avez besoin, Hub and Spoke…
- Le principe fondamental pour comprendre toute configuration de connexion Wireguard est que la connexion est établie d'égal à égal. Ce n'est pas vraiment une configuration client-serveur typique. Lorsque nous parlons d'un serveur Wireguard, nous parlons en réalité d'un seul homologue (serveur) qui établit des connexions simultanées sur le même réseau avec différents homologues (clients). Malgré cela et pour faciliter la compréhension, vous trouverez dans ce document des références aux *serveurs* et *aux clients* .

Remarque sur les logiciels tiers

Bien que ce plugin OMV facilite l'intégration du package wireguard dans openmediavault, le package wireguard lui-même a été créé par un tiers. Voir la page Web (https://translate.google.com/website?sl=auto&tl=fr&hl=fr&u=https://www.wireguard.com/) du projet/auteur → Wireguard pour des informations et une assistance plus détaillées.

WireGuard® est un VPN extrêmement simple mais rapide et moderne qui utilise une cryptographie de pointe. Il vise à être plus rapide, plus simple, plus simple et plus utile qu'IPsec, tout en évitant les énormes maux de tête. Il a l'intention d'être considérablement plus performant qu'OpenVPN. WireGuard est conçu comme un VPN à usage général pour fonctionner aussi bien sur des interfaces intégrées que sur des super-ordinateurs, adapté à de nombreuses circonstances différentes. Initialement publié pour le noyau Linux, il est désormais multiplateforme (Windows, macOS, BSD, iOS, Android) et largement déployable. Il est actuellement en développement intensif, mais il pourrait déjà être considéré comme la solution VPN la plus sécurisée, la plus facile à utiliser et la plus simple du secteur.

Conditions préalables

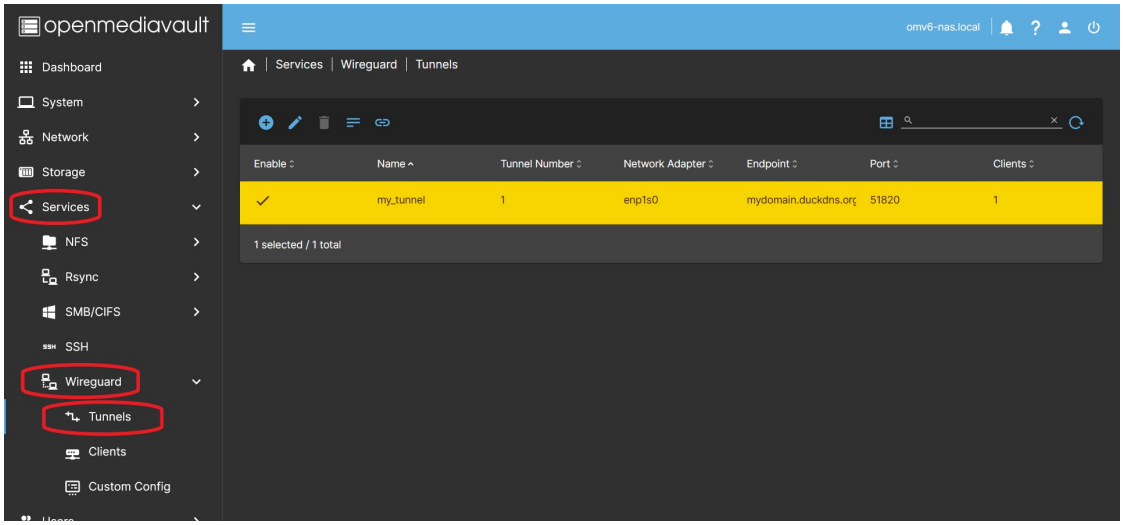
- OMV-Extras (https://wiki-omv--extras-org.translate.goog/doku.php?id=misc_docs:omv_extras&_x_tr_sl=auto&_x_tr_tl=fr&_x_tr_hl=fr) doit être préinstallé.

Installation

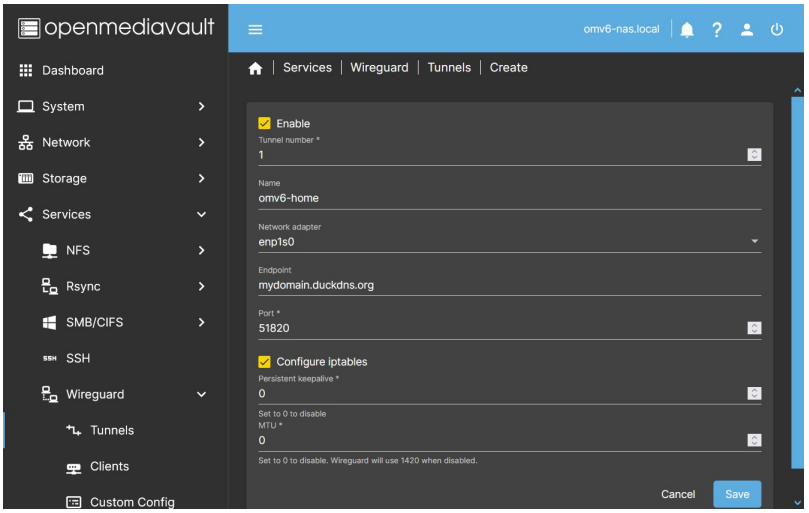
Dans l'interface graphique () d'OMV7 :
sous **System** > **Plugins** , recherchez et mettez en surbrillance **openmediavault-wireguard 7.X** , puis cliquez sur le bouton **d'installation** .

Gestion d'un tunnel

Sous **Services** > **Wireguard** > **Tunnels**



Configuration d'un tunnel



- Dans l' interface graphique () OMV, accédez à **Services** > **Wireguard** > **Tunnels.** Appuyez sur le bouton **Créer** .
  - Appuyez sur le bouton **Activer** pour activer le tunnel.
- **Configuration de base** Dans la boîte de dialogue, activez le tunnel et complétez les champs suivants :
  - **Nom** Vous pouvez nommer le tunnel pour l'identifier ultérieuren
  - **Adaptateu réseau** Cliquez sur le menu déroulant **Adaptateu réseau** et choisissez votre adaptateu
    - Si vou n'êt pas sûr de l'ad

dont vous disposez, vous pouvez accéder à **Réseau** > **Interfaces** pour le savoir.

- **Point de terminaison** Vous devez saisir l'adresse IP publique de votre routeur ou le nom d'un domaine qui pointe vers l'adresse IP publique de votre routeur. Ce point de terminaison dirigera le client vers l'adresse IP publique de votre routeur pour initier la connexion.
  - Si votre IP est dynamique (peut changer de manière inattendue) et que vous n'avez pas de domaine, vous pouvez obtenir un domaine gratuit sur Internet et configurer une mise à jour automatique de votre IP publique pour ce domaine. A la fin de ce document il y a une **procédure pour le faire en utilisant duckdns** .
  - Vérifiez votre adresse IP publique. Ce site peut être utile, whatismyip.com (https://translate.google.com/website?sl=auto&tl=fr&hl=fr&u=https://www.whatismyip.com/) Si ce n'est pas le même que celui établi dans votre routeur, c'est que vous êtes derrière CGNAT. Seul votre FAI peut le réparer.
  - Vérifiez que le domaine pointe vers votre IP. Ce site Web peut être utile, www.whatsmydns.net (https://translate.google.com/website?sl=auto&tl=fr&hl=fr&u=https://www.whatsmydns.net/) Vérifiez que votre domaine pointe vers votre IP publique.
- **Dans le champ Port** , saisissez le port que vous souhaitez utiliser pour la connexion, généralement 51820.
  - Vous pouvez choisir n'importe quel port disponible, il ne doit être occupé sur votre système par aucun service ou par un autre tunnel wireguard.
  - N'oubliez pas que vous devez ouvrir ce port dans le routeur et le diriger vers l'IP de votre serveur et avec le même port. Utilisez le protocole UDP. Si vous ne savez pas comment procéder, consultez le manuel de votre routeur.

- **Configuration avancée** Les champs précédents sont indispensables pour configurer un tunnel, si vous avez besoin d'autres configurations personnalisées elles peuvent être les suivantes (Si vous n'avez besoin de rien de tout cela, laissez les valeurs par défaut) :
  - **Configurer iptables** Par défaut, cela générera les paramètres dans iptables pour autoriser le trafic sur votre réseau interne depuis l'extérieur.
    - Si, pour une raison quelconque, vous avez besoin que le plugin ne configure pas iptables, vous pouvez le faire en décochant cette case.
    - Le paramètre par défaut est :
      - `PostUp = iptables -A FORWARD -i %i -j ACCEPT; iptables -A FORWARD -o %i -j ACCEPT; iptables -t nat -A POSTROUTING -o [NETWORK INTERFACE] -j MASQUERADE`
      - `PostDown = iptables -D FORWARD -i %i -j ACCEPT; iptables -D FORWARD -o %i -j ACCEPT; iptables -t nat -D POSTROUTING -o [NETWORK INTERFACE] -j MASQUERADE`
      - …
      - **Note**

Le plugin active le transfert IP sur l'hôte par défaut. Il n'est donc pas nécessaire d'ajouter une quelconque `sysctl -w net.ipv4.ip_forward=1` instruction de type dans l'interface.
        - …
    - **Keepalive persistant** Par défaut, il est désactivé. Définissez une valeur, telle que 25, afin qu'un bonjour soit envoyé via le tunnel toutes les 25 secondes.
        - Activez uniquement si cela est nécessaire pour une raison quelconque. L'un des principes de sécurité de Wireguard est de garder le silence sur les connexions.
    - **IP locale** Vous permet d'établir une plage réseau qui peut être choisie dans la configuration de chaque client pour diviser le trafic du tunnel sur ce client.
        - Un exemple courant pourrait être `192.168.1.0/24`
    - **MTU** Par défaut, il est désactivé. Généralement, cela équivaut à `MTU=1420` . Définissez une valeur si vous devez modifier ce paramètre. wg-quick ne prend pas en charge les valeurs inférieures à 1280. Si vous ne connaissez pas vos paramètres réseau, la valeur 1380 devrait fonctionner correctement dans la plupart des cas. La limite supérieure est de 9999.
        - Si aucune valeur n'est définie, wireguard la définira à partir de la configuration réseau existante. Généralement, cette valeur est de 1 500, donc wireguard sera automatiquement défini `MTU=1420` , puisque la longueur de l'en-tête utilisée par wireguard est soustraite (la plus longue est de 80 octets pour IPv6).
        - Vous devrez peut-être définir une valeur plus petite si votre connexion fonctionne via PPPoE ou VLAN, dans ce cas, vous devez soustraire la longueur d'en-tête ajoutée par ce réseau. Si vous ne le faites pas, les paquets seront fragmentés, davantage d'en-têtes seront ajoutés à ces paquets et le résultat sera une connexion plus lente.
        - Votre FAI peut également limiter cette valeur, vérifiez auprès de lui quelle est sa restriction.
        - Des valeurs plus élevées peuvent être définies pour les réseaux à plus haut débit.
    - Cliquez sur le bouton **Enregistrer** et acceptez les modifications. A ce moment, la connexion est configurée et active.

Note

Il est tout à fait nécessaire et obligatoire que notre opérateur nous fournisse une adresse IP publique , si nous avons CG-NAT dans notre connexion nous ne pourrons pas y accéder de l'extérieur. C'est quelque chose de tout à fait normal et c'est la première chose dont vous devez vous assurer.
Bien sûr, si nous ouvrons un port sur le routeur, nous devons être sûrs à 100 % que le pare-feu de notre serveur n'interfère pas avec la communication.

Modification d'un tunnel

- Si vous devez modifier un tunnel pour une raison quelconque, vous pouvez sélectionner le tunnel et appuyer sur le bouton **Modifier** .
    - Vous pouvez le faire à tout moment, mais si vous avez déjà configuré les clients, **la configuration des clients sera également modifiée** .

Supprimer un tunnel

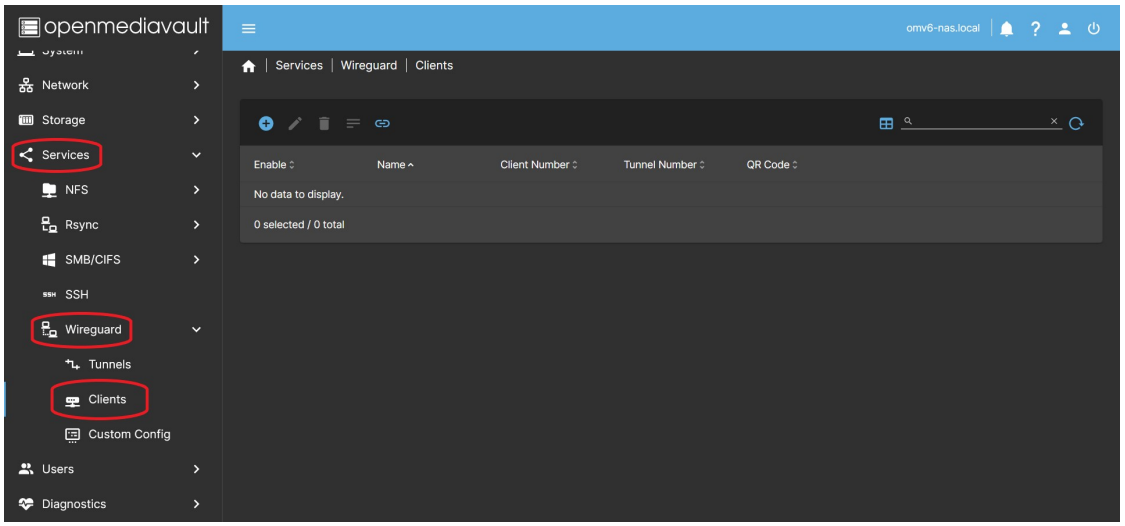
- Appuyez sur le bouton **Supprimer** pour supprimer un tunnel, sélectionnez-le au préalable.

Voir la configuration du tunnel

- Appuyez sur le bouton **Tunnel Config** pour voir la configuration du tunnel, sélectionnez-le au préalable.
    - Utile en cas d'utilisation du tunnel comme modèle pour une configuration personnalisée.

Gestion d'un Client

Sous **Services** > **Wireguard** > **Clients**

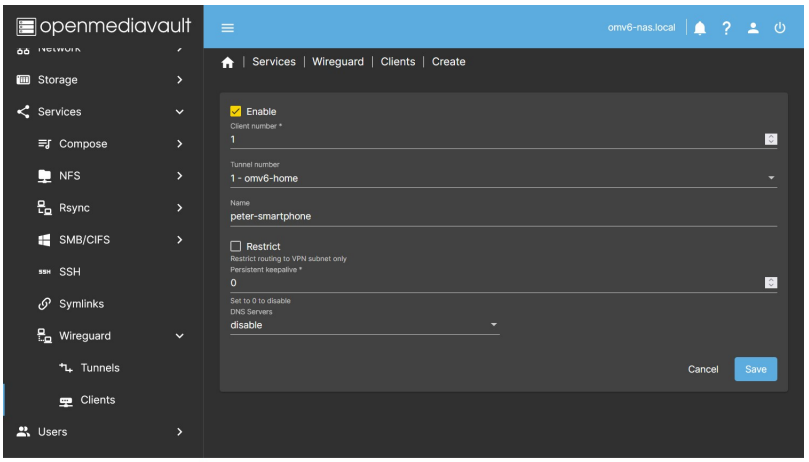


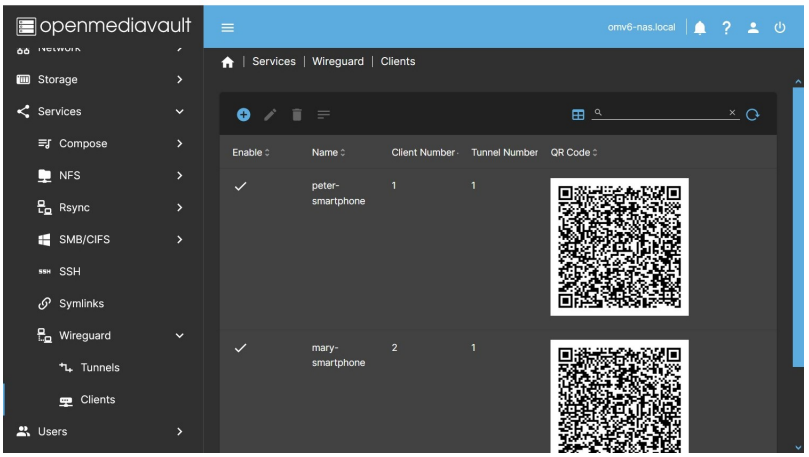
Configuration d'un client

- Dans l' interface graphique () OMV, accédez à **Services** > **Wireguard** > **Clients.** Appuyez sur le bouton **Créer** .
- **Configuration de base** Ce sont les champs nécessaires pour configurer un client, dans la boîte de dialogue activez le client et remplissez les données :
    - **Numéro de client** Il ne doit pas coïncider avec celui des autres clients.
    - **Numéro de tunnel** Vous devez affecter le client à l'un des tunnels créés précédemment.
    - **Nom** Vous pouvez nommer le client pour l'identifier ultérieurement.
- **Configuration avancée** Il s'agit d'options de configuration personnalisées qui ne sont pas nécessaires pour configurer un client, sauf pour des besoins particuliers. Si vous n'avez besoin de rien de tout cela, laissez les valeurs par défaut.
    - **Keepalive persistant** Le paramètre par défaut est de le laisser vide. Dans certains cas, il peut être nécessaire de définir une valeur ici pour maintenir la connexion active. Une valeur appropriée est généralement 25 (toutes les 25

secondes, le client enverra un paquet au serveur).

- **()Serveurs DNS ()** Le paramètre par défaut est de le laisser vide. Dans certains cas, il peut être nécessaire d'établir un serveur DNS () pour que le client puisse communiq correctem sur le réseau local. La chose habituelle sera d'établir l'IP du routeur. Le menu affichera la valeur existante dans resolv.con au cas où vous souhaiteri la copier dans le champ de droite.
- **Bouton Restreind**





. Le paramètre par défaut est de le laisser décoché, cela définira AllowedIPs sur `0.0.0.0/0` et tout le trafic sera acheminé via le tunnel. Si vous devez diviser le trafic du tunnel sur ce client, vous pouvez appuyer sur le bouton pour accéder aux différentes options. L'utilisation de l'une ou l'autre de ces options supprimera la valeur `0.0.0.0/0` de la variable AllowedIPs. Les différentes options ajoutent des valeurs, que les autres options soient actives ou non.
    - **Bouton VPN** . Appuyer sur ce bouton supprimera la `0.0.0.0/0` plage réseau des paramètres AllowedIPs et ajoutera la plage réseau que le plugin a définie pour le VPN de ce tunnel.
    - **Bouton IP local** . Appuyer sur ce bouton supprimera la plage réseau `0.0.0.0/0` des paramètres AllowedIPs et ajoutera la plage réseau définie manuellement dans les paramètres du tunnel au champ IP locale.
    - **Champ Sous-réseau(s) supplémentaire(s)** . Vous permet d'ajouter manuellement une plage réseau au champ AllowedIPs sur ce client.
- Cliquez **sur Enregistrer** . À ce stade, si vous avez déjà activé le tunnel et le client, la connexion sera opérationnelle.
- En appuyant sur le bouton **Client Config,** vous pouvez voir le fichier de configuration du client, vous pouvez copier et coller le texte dans un fichier pour configurer la connexion dans le client. Si vous procédez de cette façon, ajoutez la terminaison « .conf » au fichier créé. Traitez ce fichier comme un mot de passe, c'est la clé d'accès à votre réseau. Une fois la connexion configurée, il est conseillé de supprimer ce fichier par sécurité.
- Un QR apparaîtra dans le tableau (si le client est activé), que vous pourrez scanner depuis un smartphone pour configurer la connexion sans avoir à copier de fichier. Si vous devez l'envoyer, vous pouvez prendre une photo. Traitez cette image comme un mot de passe, c'est la clé d'accès à votre réseau (au premier chargement de la page après configuration, le code QR n'apparaît toujours pas, veuillez recharger la page ou changer d'onglet et revenir pour voir le code QR).
- Utilisez une configuration client différente pour chaque client. Si vous configurez la même connexion sur plusieurs clients en même temps, ils ne pourront pas se connecter simultanément.

Modification d'un client

- En appuyant sur le bouton **Modifier** , vous pouvez modifier les paramètres ou désactiver le client. Sélectionnez-le préalablement.

Note

Toute modification apportée doit être à nouveau implémentée sur le client à l'aide du code QR ou du fichier de configuration pour que les modifications soient appliquées.

Supprimer un client

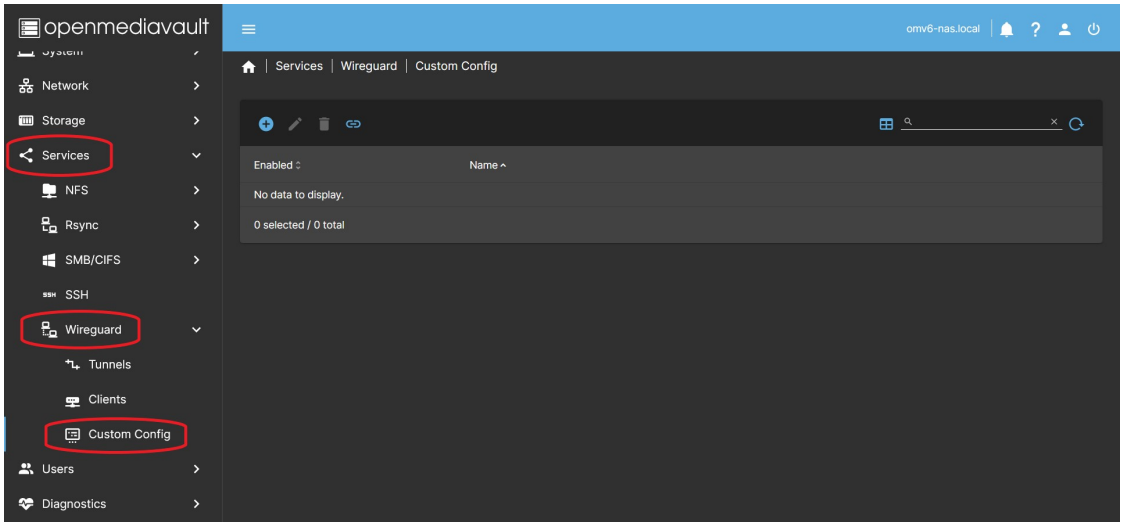
- Appuyer sur le bouton **Supprimer** supprimera le client du tunnel. Sélectionnez-le préalablement.
    - Pour supprimer un client dans un tunnel actif, vous devez d'abord désactiver le client.

Voir la configuration du client

- Appuyez sur le bouton **Client Config** pour voir la configuration du client, sélectionnez-la au préalable.
  - Utile en cas d'utilisation du client comme modèle pour une configuration personnalisée.
  - Utile en cas de configuration d'un client à l'aide d'un fichier au lieu de l'image QR.

Gestion d'une configuration personnalisée

Sous **Services** > **Wireguard** > **Configuration personnalisée**



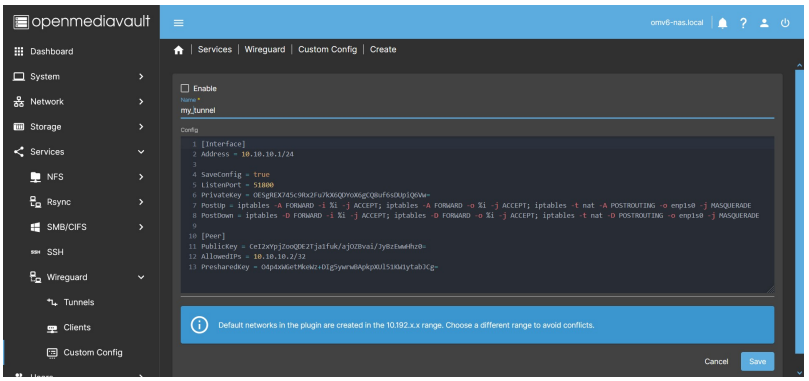
Cet onglet vous permet de créer un tunnel avec les paramètres personnalisés dont vous avez besoin. Il permet d'ouvrir une fenêtre d'édition où vous pouvez coller la configuration à partir d'un fichier texte, vous pouvez donc choisir les paramètres dont vous avez besoin pour le tunnel. Utilisez cet onglet si vous devez connecter le serveur à un service VPN Wireguard externe (commercial).

Si vous devez configurer un tunnel pour connecter le serveur point à point à un autre serveur, ou à toute autre topologie de réseau, vous devez le faire dans cet onglet, car il vous permet de définir manuellement les clés, les réseaux et autres paramètres nécessaires.

Créer une configuration personnalisée

- Dans l' interface graphique () OMV, accédez à **Services** > **Wireguard** > **Configuration personnalisée** et appuyez sur le bouton **Créer** .
- Dans le champ **Nom** , saisissez le nom de votre tunnel personnalisé.
  - Le nom du service sera `wgnet_NAME` où `NAME` est le nom choisi pour le

tunnel.
    - En raison d'une restriction existante sur la longueur des noms, le nom choisi ne peut pas contenir plus de 9 caractères l' interface graphique () ne permet pas de saisir des noms plus longs.



- Dans le champ **Config** , écrivez le contenu de configuration de votre tunnel en suivant les règles Wireguard.
    - Vous pouvez voir comment procéder sur le site Web Wireguard (https://translate.g website?sl=auto&tl= www.wireg %23simple network-interface) . Ou utilisez un

modèle en suivant la suggestion au bas de cette section.
    - Si vous souhaitez vous connecter à un service VPN commercial, il vous fournira très probablement le modèle de configuration du tunnel. Dans ce cas, copiez et collez simplement ce modèle dans le champ Config.
    - Si vous avez besoin de topologies spéciales, vous pouvez les trouver sur le site procustodibus.com (https://translate.google.com/website?sl=auto&tl=fr&hl=fr&u=https://www.procustodibus.com/blog/2020/10/wireguard-topologies/) .
    - Notez que les réseaux créés par le plugin dans les onglets **Tunnel** et **Client** sont générés dans la `10.192.x.x` plage réseau. Par conséquent, le choix de réseaux dans cette plage peut provoquer des conflits. Dans ce cas, le service ne démarrera pas et le plugin générera une erreur.
- Cliquez sur **Activer** pour activer le tunnel.
- Appuyez sur le bouton **Enregistrer** et acceptez les modifications.
    - La configuration sera stockée dans la base de données OMV.
        - Si vous avez appuyé sur le bouton **Activer** , le service sera généré et le tunnel généré sera mis en service.
        - Si vous n'avez pas appuyé sur **Enable** , la configuration sera enregistrée mais le tunnel restera inactif.
        - Si le tunnel est actif, vous pouvez voir dans le `/etc/wireguard` dossier le fichier de configuration généré que le service utilise.
            - N'apportez aucune modification à ce fichier, le plugin l'écrasera. Si vous devez modifier la configuration, faites-le dans l' interface graphique () du plugin .

Astuce : utilisez des modèles pour générer automatiquement des clés.

Si vous créez un tunnel personnalisé à partir de zéro et souhaitez éviter de créer les clés manuellement, vous pouvez créer un tunnel et un client (ou plusieurs) et les utiliser comme modèles :
Créez un tunnel dans l' onglet **Tunnels** , activez-le et créez un client. dans l' onglet **Clients** et activez-le.
Dans l' onglet **Tunnels** , sélectionnez le tunnel que vous avez créé et cliquez sur le bouton **Tunnel Config** . Copiez le texte dans le presse-papiers et accédez à l' onglet **Configuration personnalisée** . Créez un nouveau tunnel et collez le texte dans la boîte de dialogue. Enregistrez les modifications sans activer le tunnel.
Accédez à l' onglet **Clients** et sélectionnez le client que vous avez créé. Cliquez sur le bouton **Client Config** et copiez le texte dans un fichier texte. Ce sera le point de départ avec les clés pour votre pair.
Supprimez le client et le tunnel générés que vous avez utilisés comme modèles. Vous êtes maintenant prêt à modifier votre tunnel personnalisé et à l'activer. Modifie la gamme de réseaux pour s'éloigner des réseaux générés par le plugin. Modifiez le reste des paramètres selon vos besoins.

Note

Dans la section procédures de ce document, vous trouverez une procédure pour créer un tunnel point à point.

Note

Le plugin active le transfert IP sur l'hôte par défaut. Il n'est donc pas nécessaire d'ajouter une quelconque `sysctl -w net.ipv4.ip_forward=1` instruction de type dans l'interface.

Modifier une configuration personnalisée

- Dans l' interface graphique () OMV, accédez à **Services** > **Wireguard** > **Custom Config** et sélectionnez le tunnel que vous souhaitez modifier (couleur jaune).
- Appuyez sur le bouton **Modifier** .
  - Dans la fenêtre, modifiez les paramètres dont vous avez besoin.
  - Appuyez sur le bouton **Enregistrer** et acceptez les modifications.

Supprimer un tunnel personnalisé

- Dans l' interface graphique () OMV, accédez à **Services** > **Wireguard** > **Custom Config** et sélectionnez le tunnel que vous souhaitez supprimer (couleur jaune).
- S'il est actif, vous devez le désactiver au préalable.
  - Appuyez sur le bouton **Modifier** .
  - Appuyez sur le bouton **Activer** pour le désactiver.
  - Appuyez sur le bouton **Enregistrer** et acceptez les modifications.
  - Sélectionnez à nouveau le tunnel, désormais inactif.
- Appuyez sur le bouton **Supprimer** .
- Cliquez sur **Oui** et acceptez les modifications.

Comment configurer un smartphone ou un PC

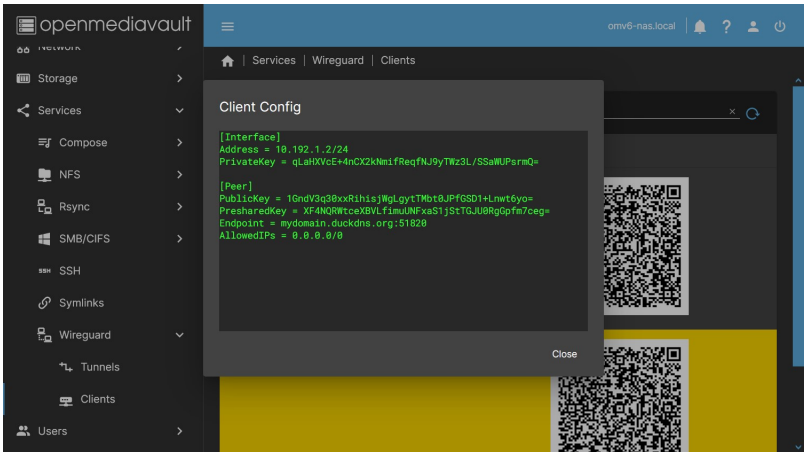
Si le client est un smartphone (android ou iOS)

- Installez l'application Wireguard sur votre smartphone.
- Ouvrez l'application et appuyez sur le bouton **+** pour ajouter une connexion. Appuyez sur l'option pour scanner un code QR. Alternativement, il peut être configuré à partir d'un fichier texte de la même manière qu'un PC (voir point suivant).
- Dans l'interface OMV rendez-vous dans **Services** > **Wireguard** > **Clients** Scannez le QR code du client correspondant depuis le smartphone.
- Tapez un nom pour votre connexion sur votre smartphone et appuyez sur **OK** .
- Votre client est configuré. Il vous suffit d'activer la connexion et vous aurez accès au réseau de votre serveur.

Note

Selon les paramètres de votre routeur, la connexion peut ne pas fonctionner si vous êtes connecté en Wi-Fi sur le même réseau que le serveur.
Dans ce cas, vous pouvez désactiver la connexion Wi-Fi du smartphone et vous connecter aux données du smartphone. De cette façon, vous pouvez tester votre connexion.

Si le client est un PC (Linux, MacOS ou Windows)



- Installez l'application Wireguard sur le PC. Sur la page officielle Wireguard, (https://translate.google.website?sl=auto&tl=fr&hl=www.wireguard.cinstall/) vous trouverez des instructions pour toutes les plateformes.
- Dans l'interface OMV, accédez à **Services** > **Wireguard** > **Clients,** cliquez sur le bouton **Client Config** . Une fenêtre avec les paramètres du client s'ouvrira. Copiez et collez le texte dans un fichier et ajoutez-y l'extension « .conf », enregistrez-le sur le bureau du PC que vous souhaitez configurer.
- Sur le PC sur lequel vous allez configurer la connexion

Wireguard, ouvrez l'application Wireguard et cliquez sur **Ajouter une connexion à partir du fichier** . Sélectionnez le fichier sur votre bureau et cliquez sur **ok** .
    - S'il n'y a pas d'option d'ajout de connexion à partir du fichier, vous pouvez **créer une nouvelle connexion** et coller le contenu du fichier à l'intérieur.
- Votre client est configuré. Il vous suffit d'activer la connexion et vous aurez accès au réseau de votre serveur.

Note

Selon les paramètres de votre routeur, la connexion peut ne pas fonctionner si le PC est connecté au même réseau que le serveur.
Si le PC dispose du Wi-Fi, vous pouvez partager une connexion de données via Wi-Fi depuis le smartphone et vous y connecter depuis le PC. De cette façon, vous pouvez tester votre connexion.

Comment diviser le trafic du tunnel

Utile si vous vous connectez à un réseau distant mais que vous souhaitez en même temps accéder à votre réseau local. Ceci peut être réalisé si les plages de réseau sont différentes sur ces deux réseaux.

- Si vous devez diviser le trafic du tunnel pour une raison quelconque, vous pouvez modifier le champ **AllowedIPs** sur le client.
    - Le champ **AllowedIPs** filtre les adresses qui navigueront à travers le réseau Wireguard. Les adresses en dehors de la plage définie ne seront pas acheminées via le tunnel Wireguard.
    - Vous pouvez modifier ce champ sur le client, cela n'affectera que le client, la configuration sur le serveur ne changera pas, le serveur recevra seulement plus ou moins de trafic.
    - `0.0.0.0/0` désigne toutes les adresses IP, c'est-à-dire tout le trafic.
- Changer la valeur `0.0.0.0/0` en autre chose limitera le trafic du tunnel à la plage IP spécifiée. Par exemple, si vous souhaitez uniquement transférer le trafic pour accéder à votre réseau 192.168.1.x, vous pouvez spécifier `192.168.1.0/24` (cela inclut toutes les adresses IP comprises entre 192.168.1.1 et 192.168.1.254)
    - Avec cette configuration, le client détournera le trafic adressé à n'importe quelle adresse IP comprise entre 192.168.1.1 et 192.168.1.254 via le tunnel Wireguard, le reste du trafic suivra son cours normal via une autre interface réseau disponible.

Comment configurer un tunnel point à point

Point à point. Tunnel standard.

La configuration point à point de Wireguard crée une connexion à distance entre deux pairs afin qu'ils puissent communiquer uniquement entre eux, sans partager d'autres points sur le réseau local de chaque homologue.

C'est une configuration simple, chaque homologue ne peut voir que l'adresse IP de l'autre homologue. Les autres IP du réseau local sont restreintes à cet égard. Logiquement, chaque homologue peut accéder à tous les ports de l'autre homologue, il peut donc accéder à tous ses services sur cette IP.

L'un des deux homologues est celui qui ouvre la connexion avec l'autre homologue et la maintient ouverte afin que l'autre homologue puisse également établir une liaison. Nous le verrons plus tard.

La configuration point à point est utile, par exemple, dans les situations où vous devez créer des tâches de sauvegarde à distance d'un serveur à un autre. Par exemple, un homologue peut accéder à un module rsync créé par l'autre homologue s'il dispose des informations d'identification nécessaires pour le faire.

La procédure est la suivante :

Configuration du premier serveur (pair 1)

- Utilisez un tunnel et un client comme modèles pour générer les clés. Vous pouvez voir comment procéder à la fin de la section **Créer une configuration personnalisée** .
- Modifiez les paramètres dans l' onglet **Configuration personnalisée** comme suit :

```
[Interface]
PrivateKey = Uses the previously generated TUNNEL PRIVATE KEY
ListenPort = 51500
Address = 10.15.15.1/32

[Peer]
PublicKey = Uses the previously generated CLIENT PUBLIC KEY
AllowedIPs = 10.15.15.2/32
```

- Vous pouvez adapter le champ *ListenPort* à vos besoins. Utilisez un port libre sur votre serveur.
- Vous pouvez adapter le champ *Adresse* à vos besoins. Assurez-vous d'utiliser `/32` pour Netmask. Cela garantira que le homologue accédera uniquement à l'adresse IP du serveur.
- Appuyez sur le bouton **Activer** pour activer le tunnel.
- Appuyez sur le bouton **Enregistrer** et acceptez les modifications.

Configuration du deuxième serveur (pair 2)

- Sur le deuxième serveur, accédez à **Services** > **Configuration personnalisée** dans l' interface graphique () et cliquez sur **Créer**
- Collez dans la fenêtre le contenu du fichier client que vous avez généré sur le premier serveur.
- Débarrassez-vous des paramètres dont vous n'avez pas besoin et conservez les clés, vous vous retrouvez donc avec cette configuration :

```
[Interface]
PrivateKey = Uses the previously generated CLIENT PRIVATE KEY
ListenPort = 51500
Address = 10.15.15.2/32

[Peer]
PublicKey = Use the previously generated TUNNEL PUBLIC KEY
AllowedIPs = 10.15.15.1/32
Endpoint = mydomain.com:51500
PersistentKeepalive = 25
```

- Si vous avez ajusté les valeurs du premier homologue, du port et du sous-réseau, répétez-le dans cette configuration.
- Ce pair sera celui qui établira la connexion et la maintiendra ouverte.
    - Pour cela, nous devons indiquer un domaine auquel se connecter. Vous pouvez en créer un gratuitement en suivant cette procédure : **Comment créer un domaine gratuit avec duckdns fix Dynamic IP**
    - Nous devons également définir la valeur *PersistentKeepalive* . Cela enverra un paquet minimum toutes les 25 secondes (ou quelle que soit la valeur que vous choisissez, 25 suffit) afin que le homologue 1 sache où diriger sa communication s'il souhaite prendre contact.
- Appuyez sur le bouton **Activer** pour activer le tunnel.
- Appuyez sur le bouton **Enregistrer** et acceptez les modifications.

Il vous suffit de vous assurer que le port défini est ouvert sur les deux routeurs et que le point de domaine défini sur le peer 2 pointe vers l'IP publique du peer 1.

Si vous êtes sûr de ce qui précède, le tunnel point à point est établi. Vous pouvez vérifier cela dans la CLI à partir du premier homologue en exécutant `ping 10.15.15.2` et à partir du deuxième homologue en exécutant `ping 10.15.15.1` ou quel que soit le sous-réseau que vous avez défini. La réponse devrait ressembler à ceci :

```
ping 10.15.15.1 (10.15.15.1) 56 (84) bytes of data.
64 bytes from 10.15.15.1: icmp_seq = 1 ttl = 64 time = 30.6 ms
64 bytes from 10.15.15.1: icmp_seq = 2 ttl = 64 time = 30.7 ms
64 bytes from 10.15.15.1: icmp_seq = 3 ttl = 64 time = 29.6 ms
64 bytes from 10.15.15.1: icmp_seq = 4 ttl = 64 time = 28.9 ms
^ C
--- 10.15.15.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 8ms
rtt min / avg / max / mdev = 28.877 / 29.934 / 30.677 / 0.775 ms
```

Si cela ne fonctionne pas, vérifiez votre domaine et vérifiez que les ports sont ouverts sur les deux routeurs.

Point à point. Variante avec deux Endpoints, silence dans la connexion.

Wireguard se caractérise par être une connexion silencieuse. Dans le cas ci-dessus, un seul des deux homologues peut initier la connexion puisque l'autre homologue n'a pas de point de terminaison. Par conséquent, pour permettre l'initiation de la connexion depuis les deux pairs, l'un d'eux doit maintenir la connexion ouverte en envoyant un paquet de

temps en temps (PersistentKeepalive).

Nous pouvons faire en sorte que le tunnel se comporte silencieusement lorsqu'il n'y a pas de communication et que les deux pairs puissent initier la connexion, pour cela nous devons établir un point de terminaison dans chacun des deux pairs. Bien entendu pour y parvenir nous devons disposer d'un domaine pointant vers chaque serveur. Nous les appellerons dans ce cas peer1.mydomain.com et peer2.mydomain.com

La configuration précédente ressemblerait à ceci :

Pair 1.

```
[Interface]
PrivateKey = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
ListenPort = 51500
Address = 10.15.15.1/32

[Peer]
PublicKey = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
AllowedIPs = 10.15.15.2/32
Endpoint = peer2.mydomain.com:51500
```

Pair 2.

```
[Interface]
PrivateKey = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
ListenPort = 51500
Address = 10.15.15.2/32

[Peer]
PublicKey = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
AllowedIPs = 10.15.15.1/32
Endpoint = peer1.mydomain.com:51500
```

Point par point. Configuration du pare-feu à l'aide d'iptables.

La configuration ci-dessus permet aux homologues de se parler sur n'importe quel port. Si vous en avez besoin, vous pouvez restreindre ces communications à un ou plusieurs ports en utilisant iptables comme pare-feu. Cela amènera chacun des pairs à rejeter toute communication en dehors de ce que nous devons établir.

Supposons que nous configurions ce tunnel pour effectuer la synchronisation rsync à partir d'un module prenant en charge les connexions sur le port 873 (vous pouvez le personnaliser). Dans ce cas, il serait souhaitable que chaque serveur autorise uniquement les connexions à ce port, refusant les connexions à tout autre service.

Supposons que nous ayons appelé l'interface de chaque homologue pp_peer1 dans l' interface graphique () du plugin . Le plugin finira par appeler cette interface wgnet_pp_peer1. De même, dans le homologue 2, l'interface s'appellera wgnet_pp_peer2.

La configuration de la section précédente ressemblerait à ceci après application des règles iptables :

Pair 1.

```
[Interface]
PrivateKey = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
ListenPort = 51500
Address = 10.15.15.1/32

# Firewall
PreUp = iptables -A INPUT -i wgnet_pp_peer1 -m state --state ESTABL
ISHED,RELATED -j ACCEPT
PreUp = iptables -A INPUT -i wgnet_pp_peer1 -m state --state NEW -p
tcp --dport 873 -j ACCEPT
PreUp = iptables -A INPUT -i wgnet_pp_peer1 -j REJECT
PostDown = iptables -D INPUT -i wgnet_pp_peer1 -m state --state EST
ABLISHED,RELATED -j ACCEPT
PostDown = iptables -D INPUT -i wgnet_pp_peer1 -m state --state NEW
-p tcp --dport 873 -j ACCEPT
PostDown = iptables -D INPUT -i wgnet_pp_peer1 -j REJECT

[Peer]
PublicKey = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
AllowedIPs = 10.15.15.2/32
Endpoint = peer2.mydomain.com:51500
```

Pair 2.

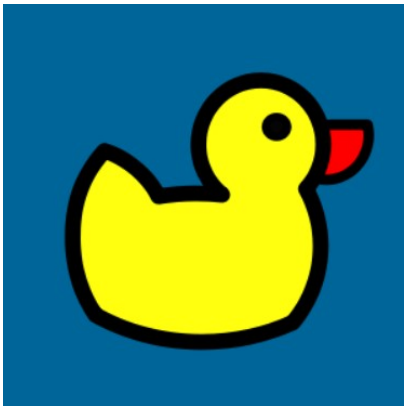
```
[Interface]
PrivateKey = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
ListenPort = 51500
Address = 10.15.15.2/32

# Firewall
PreUp = iptables -A INPUT -i wgnet_pp_peer2 -m state --state ESTABL
ISHED,RELATED -j ACCEPT
PreUp = iptables -A INPUT -i wgnet_pp_peer2 -m state --state NEW -p
tcp --dport 873 -j ACCEPT
PreUp = iptables -A INPUT -i wgnet_pp_peer2 -j REJECT
PostDown = iptables -D INPUT -i wgnet_pp_peer2 -m state --state EST
ABLISHED,RELATED -j ACCEPT
PostDown = iptables -D INPUT -i wgnet_pp_peer2 -m state --state NEW
-p tcp --dport 873 -j ACCEPT
PostDown = iptables -D INPUT -i wgnet_pp_peer2 -j REJECT

[Peer]
PublicKey = xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
AllowedIPs = 10.15.15.1/32
Endpoint = peer1.mydomain.com:51500
```

Assurez-vous de désactiver l'interface avant d'apporter des modifications à iptables, puis de la rétablir pour éviter que les règles orphelines ne restent actives.

Si vous devez ajouter plus d'un port au tunnel, par exemple les ports 80 et 443 sur peer1, vous pouvez remplacer les deuxième et cinquième lignes d'iptables par celles-ci :

```
PreUp = iptables -A INPUT -i wgnet_pp_peer1 -m state --state NEW -p
tcp -m multiport --dports 80,443 -j ACCEPT
PostDown = iptables -D INPUT -i wgnet_pp_peer1 -m state --state NEW
-p tcp -m multiport --dports 80,443 -j ACCEPT
```

Comment créer un domaine gratuit avec duckdns. Corriger l'IP dynamique



Si vous avez besoin d'un domaine, il existe de nombreuses façons de l'obtenir. L'un d'eux est fourni gratuitement par duckdns.org . (https://translate.google.com/website?sl=auto&tl=fr&hl=fr&u=https://www.duckdns.org/)

Si votre IP est dynamique, ce qui est le plus courant, elle peut changer à tout moment. Si cela se produit, vous perdrez la connexion car le domaine sera redirigé vers une adresse IP qui n'est plus celle de votre serveur. Pour résoudre ce problème, duckdns fournit également un système simple de mise à jour dynamique de l'adresse IP publique.

Obtenez un domaine dans duckdns

- Accédez au site Web duckdns.org (https://translate.google.com/website?sl=auto&tl=fr&hl=fr&u=https://www.duckdns.org/) et connectez-vous.
  - En haut vous pouvez voir le token qui a été attribué à votre compte, c'est l'identifiant du compte que vous avez créé.
- Vous pouvez obtenir jusqu'à 5 domaines différents avec un seul compte de connexion. Choisissez un domaine disponible et ajoutez-le à votre compte.
  - Ces domaines auront le format MY_DOMAIN.duckdns.org où MY_DOMAIN est choisi par vous chaque fois qu'il est disponible.

Mise à jour IP dynamique avec duckdns sur OMV

Les instructions originales peuvent être consultées ici. www.duckdns.org/install.jsp (https://translate.google.com/website?sl=auto&tl=fr&hl=fr&u=https://www.duckdns.org/install.jsp) . La procédure installe un petit fichier qui doit être exécuté de temps en temps. A chaque exécution, il envoie l'adresse IP actuelle et la compare avec celle configurée dans le domaine, si elle est différente, il la met à jour.

Au lieu de cela, nous utiliserons l'exécution de tâche planifiée dans l' interface graphique () OMV pour exécuter l'instruction directement.

Pour le configurer, procédez comme suit :

- Dans l' interface graphique () OMV, accédez à **Système** > **Tâches planifiées** et appuyez sur le bouton **Créer .**
  - Vous devez l'avoir `curl` installé sur votre système. Si vous ne l'avez pas installé, vous pouvez le faire à partir d'ici. Si vous l'avez déjà installé, ignorez les sous-étapes suivantes.

        - À installer `curl` :
            - Tapez dans le champ **Commande** `apt install curl` la commande.
            - Appuyez sur le bouton **Enregistrer**
            - Sélectionnez la tâche planifiée que vous venez de créer (couleur jaune)
            - Appuyez sur le bouton **Exécuter**
            - Cela aura installé le `curl` package sur le système. Sélectionnez maintenant à nouveau la tâche et appuyez sur le bouton **Modifier** . Supprimez la `apt install curl` commande que vous avez écrite plus tôt.
    - Tapez la commande suivante dans le champ **Commande** de la boîte de dialogue.

```
echo url="https://www.duckdns.org/update?domains=MY_DOMAIN&token=MY_TOKEN&ip=" | curl -k -o /var/log/duck.log -K -
```

- Remplacez `MY_DOMAIN` par le sous-domaine que vous avez choisi dans « `MY_DOMAIN` .duckdns.org ».
- Remplacez `MY_TOKEN` par le jeton qui a été attribué à votre compte duckdns.
- Dans le champ **Heure d'exécution** , choisissez l'option Horaire.
    - Cela exécutera la commande toutes les heures. Si votre adresse IP change fréquemment, vous pouvez la modifier pour qu'elle fonctionne pendant des périodes plus courtes. Toutes les 5 minutes peuvent être raisonnables.
- Assurez-vous que l' option **Activé** est cochée.
- Dans le champ **Balises,** vous pouvez éventuellement rédiger une description, par exemple `Duckdns_DDNS` .
- Appuyez sur le bouton **Enregistrer** .
- Exécutez la tâche une fois pour vérifier le fonctionnement. Sélectionnez la tâche et appuyez sur le bouton **Exécuter** .
    - Vous pouvez vérifier sur le site duckdns que votre domaine a été mis à jour et pointe désormais vers l'IP publique de votre serveur.
        - Vérifiez votre adresse IP publique sur le site Web Quelle est mon IP (https://translate.google.com/website?sl=auto&tl=fr&hl=fr&u=https://www.whatismyip.com/) . Si vous avez tout configuré correctement et que cela ne correspond pas, vous êtes probablement derrière CGNAT. Vérifiez auprès de votre fournisseur d'accès Internet une solution à ce problème.

Note

Cette tâche créera un fichier journal dans /var/log/duck.log.

Problèmes communs

Je n'arrive pas à me connecter au réseau depuis l'extérieur

- Si vous testez la connexion depuis un appareil connecté au même réseau par Wi-Fi ou par câble, cela ne fonctionnera pas, déconnectez-vous de ce réseau. Par exemple, avec un smartphone, désactivez le Wi-Fi et connectez-vous à Internet

avec une connexion de données mobile.
- Assurez-vous que votre adresse IP publique est accessible sur Internet. Recherchez sur votre routeur l'adresse IP publique que vous utilisez. Comparez cette IP avec celle qui apparaît sur n'importe quel site du type whatismyip, par exemple www.whatismyip.net (https://translate.google.com/website?sl=auto&tl=fr&hl=fr&u=https://www.whatismyip.net/) . S'ils sont différents, votre réseau n'est pas accessible. Vérifiez auprès de votre FAI si vous êtes dans CGNAT, si c'est le cas, demandez à votre FAI de vous supprimer de CGNAT si possible.
- Vérifiez que votre domaine pointe vers votre IP publique. Tapez votre domaine sur ce site www.whatsmydns.net (https://translate.google.com/website?sl=auto&tl=fr&hl=fr&u=https://www.whatsmydns.net/) et vérifiez s'il pointe vers votre IP.
- Assurez-vous d'avoir établi un tunnel avec les valeurs par défaut et vérifiez le fonctionnement. Une fois vérifié, procédez aux personnalisations.

La connexion fonctionne, je reçois des données, mais je ne parviens pas à accéder au réseau

Définissez la valeur `AllowedIPs = 0.0.0.0/0` et vérifiez si vous y avez accès. Si tel est le cas, vous pouvez maintenant procéder à la personnalisation de la plage réseau. Si après avoir personnalisé la portée du réseau, vous perdez l'accès, vous n'avez pas bien fait les choses.

J'ai défini les mêmes paramètres sur le smartphone et l'ordinateur portable et cela ne fonctionne que sur un seul

Si vous avez besoin de deux accès depuis deux clients ou plus, vous devez configurer une connexion différente pour chaque client. Si vous établissez la même configuration sur différents clients, un seul d'entre eux fonctionnera, ils ne fonctionneront jamais simultanément.

Je ne peux pas accéder à mes dossiers partagés

Parfois, la résolution de noms de domaine peut ne pas fonctionner. Si tel est le cas, essayez d'accéder via l'adresse IP de votre serveur au lieu du nom de domaine.

Je ne parviens pas à accéder à Internet depuis mon client

Si vous pouvez accéder à votre réseau local à partir du client mais que vous ne pouvez pas accéder à Internet, essayez de diviser le trafic du tunnel. Les connexions dirigées vers votre réseau local passeraient par le tunnel Wireguard, le reste des connexions passerait par l'interface réseau standard de votre smartphone accessible par l'ordinateur portable. Pour ce faire, vous devez procéder comme ceci :

- Dans la configuration client remplacez la ligne `AllowedIPs = 0.0.0.0/0` par ceci `AllowedIPs = 192.168.1.0/24` (en supposant que la portée de votre réseau local soit celle-là, adaptez-la à votre cas)

Code source

→ ouvrirmediavault-wireguard (https://translate.google.com/website?sl=auto&tl=fr&hl=fr&u=https://github.com/OpenMediaVault-Plugin-Developers/openmediavault-wireguard)

Un mot de clôture

Nous, qui soutenons le projet openmediavault, espérons que vous avez trouvé ce guide utile et que vous trouverez votre serveur openmediavault efficace, facile à utiliser et agréable.

Si vous avez trouvé ce guide de plugin utile, veuillez envisager un modeste don pour prendre en charge les frais d'hébergement de ce serveur.

OMV-Extras.org

Donate

Venmo : ryecoaaron