Tunnel L2 - VXLAN

1. Introduction : Pourquoi utiliser VXLAN ?

1.1. À quoi sert un tunnel VXLAN ?

Un tunnel **VXLAN (Virtual Extensible LAN)** permet de créer un réseau Ethernet virtuel (Layer 2) entre deux sites distants, comme s'ils étaient connectés au même switch. Contrairement aux tunnels L3 (GRE, IPIP), VXLAN transporte des **trames Ethernet complètes**, ce qui permet de bridger des réseaux distants. **Exemple concret :**

- Vous avez un réseau local derrière une box opérateur standard.
- Vous voulez une ou des IPs publiques directement accessibles sur vos machines, comme si elles étaient sur le même réseau que la plateforme.

1.2. Avantages de VXLAN

- Transport Layer 2 complet (trames Ethernet).
- Supporte jusqu'à 16 millions de réseaux virtuels (VNI 24 bits).
- Compatible avec la plupart des équipements réseau modernes.
- Fonctionne en UDP, facilement NAT-able.
- Peut être combiné avec d'autres tunnels L2 via un bridge.

1.3. Inconvénients

- Pas de chiffrement natif.
- Overhead plus important que GRE/IPIP (50 octets d'en-tête).
- Nécessite une configuration manuelle du bridging côté client.

---

2. Prérequis

2.1. Ce dont vous avez besoin

- Une **IP publique** (ou une redirection de port si derrière une box).
- Un **routeur ou serveur compatible VXLAN** (Linux, MikroTik, Cisco, Arista, Juniper, etc.).
- Un **service Tunnel-IP**.

2.2. Ports et NAT

- VXLAN utilise **UDP** (port par défaut : **4789**).
- Si le routeur final est derrière une box / NAT, il faut **rediriger le port UDP 4789** vers le routeur final.

---

3. Étape 1 : Créer le bridge et le tunnel sur Tunnel-IP.com

> Les tunnels L2 fonctionnent avec des **bridges** sur la plateforme. Un bridge est un switch virtuel qui regroupe vos tunnels L2 et vos subnets. Vous devez d'abord créer un bridge, puis créer le tunnel en l'associant à ce bridge.

3.1. Créer le bridge

1. Connectez-vous à panel.tunnel-ip.com
2. Allez dans **"Bridges"**.
3. Cliquez sur **"Ajouter un Bridge"** et donnez-lui un nom.
4. Validez. Le bridge sera créé sur la plateforme.

3.2. Créer le tunnel VXLAN

1. Allez dans **"Tunnels"** > **"VXLAN"**.
2. Remplissez les informations :
   - **Nom du tunnel** (ex : `<span class="editor-theme-code">VXLAN_Tunnel</span>`).
   - **Endpoint** (IP publique du routeur / box, ex : `<span class="editor-theme-code">203.0.113.1</span>`).
   - **Bridge** : Sélectionnez le bridge créé précédemment.
   - **MTU** : Taille maximale des paquets (-1 pour laisser la plateforme choisir).
3. Validez. La plateforme s'occupera de choisir le VNI et de configurer le tunnel côté Tunnel-IP.com.
4. Attendez que le tunnel soit créé
5. Cliquez sur "Accéder" pour récupérer les détails du tunnel

3.3. Créer le subnet

1. Allez dans **"Subnets"**

2. Copiez le bloc d'IP publique et collez le dans le formulaire

3. Sélectionnez le **bridge** (et non un tunnel directement) comme destination

4. Cliquez sur Créer

5. Une fois son statut à "Actif", la route est correctement installée sur les routeurs de la plateforme

> **Important :** La première IP du subnet sera utilisée comme **gateway** sur le bridge côté plateforme. Par exemple, pour le subnet `198.51.100.0/29`, la gateway sera `198.51.100.1`.

---

4. Étape 2 : Configurer le tunnel

> Contrairement aux tunnels L3, un tunnel VXLAN transporte des trames Ethernet (Layer 2). Côté client, vous devez créer l'interface VXLAN puis la **bridger** avec votre interface LAN (ou lui assigner directement une IP du subnet).

4.1. Exemple pour Linux (Ubuntu/Debian)

Étapes :

1. **Créer l'interface VXLAN** :

   ```
   ip link add vxlan0 type vxlan id 100 remote 172.16.126.33 dstport 4789
   ip link set vxlan0 up
   ```

   - `vxlan0` : Nom de l'interface VXLAN.
   - `id 100` : VNI (Virtual Network Identifier), fourni par la plateforme.
   - `remote 172.16.126.33` : IP distante (Tunnel-IP.com).
   - `dstport 4789` : Port UDP de destination.
2. **Option A : Assigner directement une IP publique** :
   Si vous souhaitez que le routeur lui-même ait une IP publique :

   ```
   ip addr add 198.51.100.2/29 dev vxlan0
   ip route add default via 198.51.100.1 dev vxlan0
   ```

- `198.51.100.2/29` : Une IP du subnet (la première IP .1 est la gateway côté plateforme).
- `198.51.100.1` : Gateway (première IP du subnet, côté plateforme).

3. **Option B : Bridger avec une interface LAN** :
   Si vous souhaitez que des machines sur votre LAN aient directement les IPs publiques :

```
ip link add br0 type bridge
ip link set br0 up
ip link set vxlan0 master br0
ip link set eth1 master br0  # Interface LAN vers vos machines
```

   Les machines connectées à `eth1` pourront alors utiliser les IPs du subnet directement, avec `198.51.100.1` comme gateway.
4. **Activer le forwarding** (si nécessaire) :

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

   (Pour rendre permanent, ajoutez `net.ipv4.ip_forward=1` dans `/etc/sysctl.conf`.)

4.2. Exemple pour MikroTik

Étapes :

1. **Créer l'interface VXLAN** :

```
/interface/vxlan/add name=vxlan0 vni=100 port=4789
```

2. **Ajouter le VTEP distant (Tunnel-IP.com)** :

```
/interface/vxlan/vteps/add interface=vxlan0 remote-ip=172.16.126.33 port=4789
```

3. **Option A : Assigner une IP publique directement** :

```
/ip/address/add address=198.51.100.2/29 interface=vxlan0
/ip/route/add dst-address=0.0.0.0/0 gateway=198.51.100.1
```

4. **Option B : Bridger avec une interface LAN** :

```
/interface/bridge/add name=br-vxlan
/interface/bridge/port/add bridge=br-vxlan interface=vxlan0
/interface/bridge/port/add bridge=br-vxlan interface=ether2  # Interface LAN
```

4.3. Exemple pour Cisco (IOS/XE - NX-OS)

Étapes :

1. **Accéder au mode configuration** :

```
enable
configure terminal
```

2. **Créer le NVE (Network Virtualization Edge)** :

```
interface nve1
 source-interface Loopback0
 member vni 100
  ingress-replication protocol static
   peer-ip 172.16.126.33
 no shutdown
```

3. **Créer le VLAN et l'associer au VNI** :

```
vlan 100
 vn-segment 100

interface Vlan100
 ip address 198.51.100.2 255.255.255.248
 no shutdown
```

4. **Sauvegarder la configuration** :

```
write memory
```

> La configuration VXLAN sur Cisco est principalement disponible sur **NX-OS** (Nexus). Sur IOS-XE, le support VXLAN est limité à certains modèles (CSR1000v, Catalyst 9000, etc.).

4.4. Exemple pour Arista (EOS)

Étapes :

1. **Accéder au mode configuration** :

```
enable
configure terminal
```

2. **Créer l'interface VXLAN** :

```
interface Vxlan1
 vxlan source-interface Loopback0
 vxlan udp-port 4789
 vxlan vlan 100 vni 100
 vxlan flood vtep add 172.16.126.33
```

3. **Créer le VLAN et l'associer** :

```
vlan 100

interface Vlan100
 ip address 198.51.100.2/29
 no shutdown
```

4. **Sauvegarder la configuration** :

```
write memory
```

4.5. Exemple pour Juniper (JunOS)

Étapes :

1. **Accéder au mode configuration** :

```
edit
```

2. **Créer l'interface VXLAN et le bridge domain** :

```
set bridge-domains VXLAN-BD vlan-id 100
set bridge-domains VXLAN-BD vxlan vni 100
set bridge-domains VXLAN-BD vxlan ingress-node-replication

set interfaces lo0 unit 0 family inet address 10.0.0.1/32
set switch-options vtep-source-interface lo0.0
set switch-options remote-vtep 172.16.126.33

set bridge-domains VXLAN-BD routing-interface irb.100
set interfaces irb unit 100 family inet address 198.51.100.2/29
```

3. **Appliquer la configuration** :

```
commit
```

5. Étape 3 : Configurer les IPs publiques

5.1. Architecture L2

```
Internet → [Infrastructure Tunnel-IP.com] → [Bridge plateforme] → (Tunnel VXLAN) → [Votre Bridge/Interface] → [Vos Machines]
```

- La plateforme met la première IP du subnet comme gateway sur le bridge (ex: `198.51.100.1`).
- Vos machines utilisent les autres IPs du subnet.
- Comme le tunnel est L2, les trames Ethernet passent directement, **pas besoin de NAT ni de routage complexe**.

5.2. Attribution des IPs

Un `/29` contient **8 adresses IP** :

- `198.51.100.0` : Adresse réseau (inutilisable).
- `198.51.100.1` : **Gateway plateforme** (réservée automatiquement).
- `198.51.100.2` à `198.51.100.6` : IPs utilisables pour vos machines.
- `198.51.100.7` : Adresse broadcast (inutilisable).

5.3. Configuration sur les machines

Sur chaque machine qui doit utiliser une IP publique :

Linux :

```
ip addr add 198.51.100.2/29 dev eth0
ip route add default via 198.51.100.1
```

MikroTik :

```
/ip/address/add address=198.51.100.2/29 interface=ether1
/ip/route/add dst-address=0.0.0.0/0 gateway=198.51.100.1
```

>

> Si vous avez bridgé l'interface VXLAN avec votre LAN, les machines peuvent êtreconfigurées directement avec les IPs publiques et la gateway `198.51.100.1`, comme si elles étaient sur un réseau local classique.

---

6. Commandes spécifiques par constructeur

Constructeur	Commande pour vérifier le tunnel	Commande pour vérifier le bridge
Linux	`ip -d link show vxlan0`	`bridge link show`
MikroTik	`/interface vxlan print`	`/interface bridge port print`
Cisco	`show nve peers`	`show vlan`
Arista	`show interfaces Vxlan1`	`show vlan`
Juniper	`show bridge-domain`	`show interfaces irb`

7. Vérification et Dépannage

7.1. Tester la connectivité

- **Depuis votre routeur / machine** :

```
ping 198.51.100.1  # Ping de la gateway plateforme
```

- **Vérifier l'interface VXLAN (Linux)** :

```
ip -d link show vxlan0  # Vérifier les paramètres VXLAN
bridge fdb show dev vxlan0  # Vérifier la table MAC
```

- **Depuis une machine locale** :

```
ping 8.8.8.8  # Vérification internet
curl -4 ifconfig.me  # Vérification de l'IP sortante
```

7.2. Problèmes courants

Symptôme	Cause Possible	Solution

Le tunnel ne répond pas	Port UDP 4789 bloqué	Vérifiez la redirection du port sur votre box
Pas de connectivité L2	VNI incorrect	Vérifiez que le VNI correspond à celui fourni par la plateforme
La gateway ne répond pas	Subnet non associé au bridge	Vérifiez que le subnet est bien associé au bridge sur la plateforme
Latence élevée / Pertes de paquets	MTU trop grande	Réduisez la MTU : `ip link set mtu 1450 dev vxlan0` (overhead VXLAN = 50 octets)
Les IPs publiques ne fonctionnent pas	Mauvaise gateway	Utilisez la première IP du subnet comme gateway (ex: `198.51.100.1`)

---

8. Bonnes Pratiques

1. **Sécurité** :
   - Filtrez le trafic entrant avec un pare-feu (ex: `iptables` ou ACL).
   - VXLAN n'offre aucun chiffrement natif. Envisagez IPsec ou Wireguard en complément si le chiffrement est nécessaire.
2. **Performance** :
   - Réduisez la MTU à `1450` pour éviter la fragmentation (overhead VXLAN = 50 octets).
   - Vérifiez que la MTU de votre lien WAN est suffisante pour encapsuler les paquets VXLAN.
3. **Bridging** :
   - Vous pouvez combiner plusieurs tunnels L2 (VXLAN, EoIP, GRETAP) sur le même bridge, aussi bien sur la plateforme que côté client.
   - Attention aux boucles de bridge : activez le STP si nécessaire.

Revision #1
Created 2026-03-30 14:41:58 UTC by Landry JUGE
Updated 2026-03-30 14:42:26 UTC by Landry JUGE