



<https://www.overleaf.com/project/6274c4f4d8f20acc3c43ab96>

# INFRASTRUCTURES RESEAUX

Claudio ANTONIO  
Yawavi Jeona-Lucie LATEVI

Mai 2022

Mise en place de VLANs, Tunnel L2TPv3 sécurisé

# Table des matières

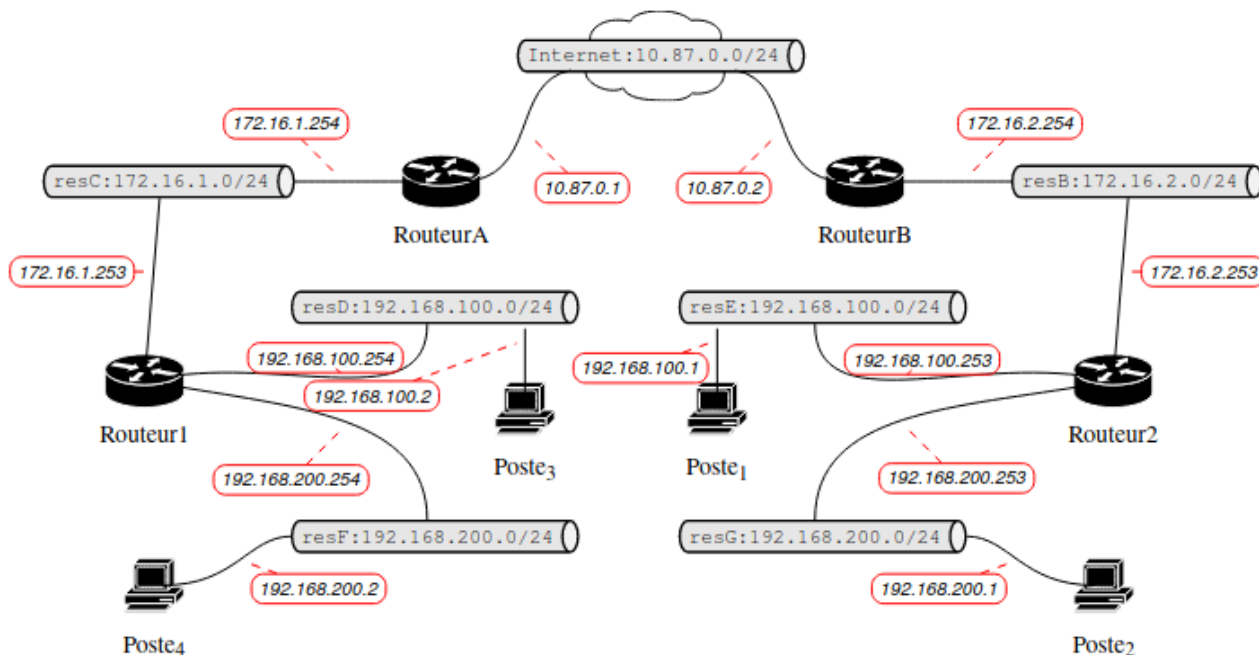
<b>I</b>	<b>Introduction</b>	<b>5</b>
<b>II</b>	<b>Mise en place du vlan</b>	<b>6</b>
II..1	Routeur 1 . . . . .	7
II..2	Routeur 2 . . . . .	7
II..3	Les postes . . . . .	7
II..4	Capture du fonctionnement de l'encapsulation VLAN . . . . .	8
<b>III</b>	<b>Mise en place du tunnel L2TPv3 en mode encapsulation IP</b>	<b>8</b>
III..1	Routeur 1 . . . . .	9
III..2	Routeur 2 . . . . .	10
<b>IV</b>	<b>Comparaison des tunnel L2tpv3 (ip ,udp) vs GrepTap</b>	<b>13</b>
IV..1	Teste de connectivité entre poste1 et 3 pour le tunnel en mode ip , que nous avons mis en dessus. . . . .	14
IV..2	La configuration du tunnel en mode udp est la suivante : . . . . .	15
IV..3	Routeur 1 . . . . .	15
IV..4	Routeur 2 . . . . .	15
IV..5	Teste de connectivité entre poste1 et 3 pour le tunnel en mode udp . . . . .	16
IV..6	Configuration des interface pour le tunnel GrepTap . . . . .	17
IV..7	Routeur 1 . . . . .	17
IV..8	Routeur 1 . . . . .	17
IV..9	Teste de connectivité entre poste1 et 3 pour le tunnel GreTap . . . . .	18
<b>V</b>	<b>Mise en place du chiffrement IPsec sur le tunnel l2TPv3 en encapsulation IP.</b>	<b>19</b>
V.I	Débit sans le chiffrement Ipsec . . . . .	19
V.II	Débit avec le chiffrement Ipsec . . . . .	20
<b>VI</b>	<b>Accès Internet « intelligent »</b>	<b>21</b>
<b>VII</b>	<b>interdiction du trafic entre VLAN 100 et VLAN 200</b>	<b>23</b>
<b>VIII</b>	<b>Description du protocole L2TPv3 et de la technologie des VXLANs</b>	<b>25</b>
VIII.I	Présentation rapide du protocole L2TPv3 et de la technologie des VXLANs . . . . .	25
VIII.I.1	L2TPv3 . . . . .	25
VIII.I.2	VXLANs . . . . .	25
VIII.II	Comparaison des deux solutions . . . . .	25
VIII.III	La mise en œuvre dans Linux des VXLANs dans Openv Switch . . . . .	26
VIII.IV	Les solutions de chiffrement du trafic L2TPv3 ou VXLAN . . . . .	27
VIII.IV.1	L2TP . . . . .	27
VIII.IV.2	VXLAN . . . . .	27
VIII.V	Comparaison avec MPLS ces deux technologies . . . . .	27
VIII.V.1	MPLS et L2TPv3 . . . . .	27
VIII.V.2	MPLS et VXLAN . . . . .	28



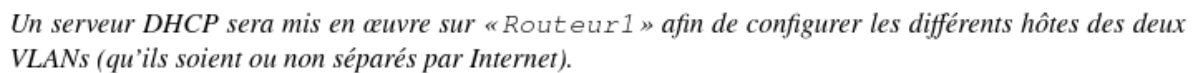
## I Introduction

Dans ce projet nous allons étudier le protocole L2TPv3, RFC 3931, pour faire des tunnels de niveau 2. Nous nous en servons pour faire du « trunking » de VLANs et de la mutualisation de service comme le DHCP. Enfin, à l'aide d'une configuration « intelligente », on limitera l'utilisation du tunnel lorsque les hôtes voudront se connecter à Internet en leur permettant de le faire directement de leur « côté d'Internet ».

On dispose du réseau suivant où les réseaux sont disjoints au niveau 2 :



On voudrait arriver au schéma suivant où les réseaux sont liés au niveau 2 :



Pour ce projet, nous avons mis en place 2 VLANs de port (VLAN 100 et VLAN 200) qui offrent une sécurité maximale tout en gardant la flexibilité des VLANs. Toute trame venant d'un VLAN sera étiquetée 'VLAN Tagging' (suivant le format 802.1Q). Par cette étiquette, on pourra savoir à quelle VLAN appartient une trame donnée. Nous allons mettre en place nos deux vlans sur les switchs resD et resE connectés respectivement au routeur 1 et au routeur 2. Dans notre projet, nous avons aussi des postes(poste 1, 2, 3, 4) qui vont nous servir d'hôte. Les switchs vont également permettre d'isoler les trafics de niveau 2 de ces réseaux pour qu'ils n'interfèrent pas entre eux : ARP, DHCP, Neighbor Discovery Protocol et de faire du VLAN trunking.

- \* Les postes 1 et 3 appartiendront au VLAN 100 mais seront respectivement placés sur le routeur 2 et 1.
- \* Les postes 2 et 4 appartiendront au VLAN 200 mais seront respectivement placés sur le routeur 2 et 1.

6

## II..1 Routeur 1

- \* Configuration du VLAN 100 sur l'interface rout1-eth1 avec comme id 100 et donc on a le VLAN 100 configuré sur rout1-eth1.100
- \* Configuration du VLAN 200 sur l'interface rout1-eth1 avec comme id 200 et donc on a le VLAN 200 configuré sur rout1-eth1.200

## II..2 Routeur 2

- \* Configuration du VLAN 100 sur l'interface rout2-eth1 avec comme id 100 et donc on a le VLAN 100 configuré sur rout2-eth1.100
- \* Configuration du VLAN 200 sur l'interface rout2-eth1 avec comme id 200 et donc on a le VLAN 200 configuré sur rout2-eth1.200

```
1 #rout1
2 ip netns exec rout1 ip link add link rout1-eth1 name rout1-eth1.100 type vlan id 100
3 ip netns exec rout1 ip link set dev rout1-eth1.100 up
4 ip netns exec rout1 ip link add link rout1-eth1 name rout1-eth1.200 type vlan id 200
5 ip netns exec rout1 ip link set dev rout1-eth1.200 up
6 ip netns exec rout1 ip a add dev rout1-eth1.100 192.168.100.254/24
7 ip netns exec rout1 ip a add dev rout1-eth1.200 192.168.200.254/24
8 #rout2
9 ip netns exec rout2 ip link add link rout2-eth1 name rout2-eth1.100 type vlan id 100
10 ip netns exec rout2 ip link set dev rout2-eth1.100 up
11 ip netns exec rout2 ip link add link rout2-eth1 name rout2-eth1.200 type vlan id 200
12 ip netns exec rout2 ip link set dev rout2-eth1.200 up
13 ip netns exec rout2 ip a add dev rout2-eth1.100 192.168.100.253/24
14 ip netns exec rout2 ip a add dev rout2-eth1.200 192.168.200.253/24
```

## II..3 Les postes

- \* Configuration du VLAN 100 sur l'interface poste1-eth0 avec comme id 100 et donc on a le VLAN 100 configuré sur poste1-eth0.100
- \* Configuration du VLAN 100 sur l'interface poste3-eth0 avec comme id 100 et donc on a le VLAN 100 configuré sur poste3-eth0.100
- \* De même avec le poste 2 et 4 mais avec comme id 200 puisqu'ils appartiennent au VLAN 200

```
1 ip netns exec poste1 ip link add link poste1-eth0 name poste1-eth0.100 type vlan id
  100
2 ip netns exec poste1 ip link set poste1-eth0.100 up
3 ip netns exec poste2 ip link add link poste2-eth0 name poste2-eth0.200 type vlan id
  200
4 ip netns exec poste2 ip link set poste2-eth0.200 up
5 ip netns exec poste3 ip link add link poste3-eth0 name poste3-eth0.100 type vlan id
  100
6 ip netns exec poste3 ip link set poste3-eth0.100 up
7 ip netns exec poste4 ip link add link poste4-eth0 name poste4-eth0.200 type vlan id
  200
8 ip netns exec poste4 ip link set poste4-eth0.200 up
```

Pour la configurations de postes de chaque routeur, nous allons utiliser un serveur DHCP plutôt que de configurer les adresse ip à la main ainsi que les routes par défaut dans le fichier build, par exemple pour configurer le poste3, avec les commandes suivantes.

```
1 ip netns exec rout1 dnsmasq -d -z -i rout1-eth1.100 -F
   192.168.100.1,192.168.100.150,255.255.255.0

1 ip netns exec poste3 sudo dhclient
```

## II.4 Capture du fonctionnement de l'encapsulation VLAN

Commande lancée depuis poste3 vers rout1 :[poste3] ping -c 3 172.16.1.253  
on sniffe sur rout1 avec la commande suivante : [rout1] : sudo tcpdump -nvveX -i rout1-eth1.100

```
1 09:06:13.713356 de:f6:d1:4e:b7:41 > 1a:c6:14:bb:0a:c0, ethertype 802.1Q (0x8100),
   length 102: vlan 100, p 0, ethertype IPv4, (tos 0x0, ttl 64, id 967, offset 0,
   flags [DF], proto ICMP (1), length 84)
2 192.168.100.115 > 172.16.1.253: ICMP echo request, id 6088, seq 1, length 64
3 0x0000: 4500 0054 03c7 4000 4001 63b9 c0a8 6473 E..T...@.c...ds
4 0x0010: ac10 01fd 0800 6656 17c8 0001 e5c8 7462 .....fV.....tb
5 0x0020: 0000 0000 56e2 0a00 0000 0000 1011 1213 ....V.....
6 0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!"#
7 0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 $%&'()*+,-./0123
8 0x0050: 3435 3637 4567
9
10 09:06:13.713910 1a:c6:14:bb:0a:c0 > de:f6:d1:4e:b7:41, ethertype 802.1Q (0x8100),
   length 102: vlan 100, p 0, ethertype IPv4, (tos 0x0, ttl 64, id 26384, offset 0,
   flags [none], proto ICMP (1), length 84)
11 172.16.1.253 > 192.168.100.115: ICMP echo reply, id 6088, seq 1, length 64
12 0x0000: 4500 0054 6710 0000 4001 4070 ac10 01fd E..Tg...@.p....
13 0x0010: c0a8 6473 0000 6e56 17c8 0001 e5c8 7462 ..ds..nV.....tb
14 0x0020: 0000 0000 56e2 0a00 0000 0000 1011 1213 ....V.....
15 0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!"#
16 0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 $%&'()*+,-./0123
17 0x0050: 3435 3637 4567
18
```

- \* **de :f6 :d1 :4e :b7 :41** : adresse mac source du poste 3
- \* **1a :c6 :14 :bb :0a :c0** : adresse mac destination du rout1
- \* **Ethertype** : 802.1Q
- \* le numéro du **vlan 100**
- \* protocole **ICMP**
- \* **ttl 64** , le paquet ne traverse pas le routeur
- \* ip src (celui du poste3) : **192.168.100.115**
- \* ip dst ( d'une des interface de rout1) : **172.16.1.253**

## III Mise en place du tunnel L2TPv3 en mode encapsulation IP

Un tunnel, Layer 2 Tunneling Protocol Version 3 , est un protocole d'encapsulation point à point d'un protocole quelconque de niveau 2 dans IP. C'est une évolution importante de L2TPv2 qui n'autorise que l'encapsulation du protocole PPP (source wikipedia).



Le tunnel L2TPv3 en mode encapsulation IP dans notre cas va permettre de créer une liaison point à point entre le routeur1 et le routeur2.

Voici la configuration du tunnel l2tpv3 :

### III.1 Routeur 1

```
1 # Cr ation du tunnel: l'interface IF2 ici l2tpeth0 de 172.16.1.253 -> 172.16.2.253
2 ip netns exec rout1 ip l2tp add tunnel remote 172.16.2.253 local 172.16.1.253 encap
   ip tunnel_id 3000 peer_tunnel_id 4000
3 ip netns exec rout1 ip l2tp add session tunnel_id 3000 session_id 1000
   peer_session_id 2000
4 # Activation de l'interface l2tpeth0
5 ip netns exec rout1 ip link set l2tpeth0 up
6 # ajout du bridge "tunnel"
7 ip netns exec rout1 brctl addbr tunnel
8 # ajout dans de l'interface l2tpeth0 au bridge
9 ip netns exec rout1 brctl addif tunnel l2tpeth0
10 # ajout dans de l'interface connect au r seau (rout1-eth1) au bridge
11 ip netns exec rout1 brctl addif tunnel rout1-eth1
12 # activation de l'interface correspondant au bridge
13 ip netns exec rout1 ip link set tunnel up
14 # configuration de l tiquettage VLAN pour le VLAN 100
15 ip netns exec rout1 ip netns exec rout1 ip link add link tunnel name rout1-eth1.100
   type vlan id 100
16 # activation de l'interface tunnel.100
17 ip netns exec rout1 ip link set rout1-eth1.100 up
18 # configuration de l tiquettage VLAN pour le VLAN 200
19 ip netns exec rout1 ip netns exec rout1 ip link add link tunnel name rout1-eth1.200
   type vlan id 200
20 # activation de l'interface tunnel.200
21 ip netns exec rout1 ip link set rout1-eth1.200 up
22 # configuration IP de l interface
23 ip netns exec rout1 ip addr add 192.168.100.254/24 dev rout1-eth1.100
24 ip netns exec rout1 ip addr add 192.168.200.254/24 dev rout1-eth1.200
```

Une fois le tunnel en place on peut remarquer les nouvelles interfaces.

```
1 [rout1] ip l
2 1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT
   group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
3 4: l2tpeth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1458 qdisc fq_codel master tunnel
   state UNKNOWN mode DEFAULT group default qlen 1000
   link/ether 06:a2:87:5d:8b:b0 brd ff:ff:ff:ff:ff:ff
4 5: tunnel: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1458 qdisc noqueue state UP mode
   DEFAULT group default qlen 1000
   link/ether 06:a2:87:5d:8b:b0 brd ff:ff:ff:ff:ff:ff
5 6: rout1-eth1.100@tunnel: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1458 qdisc noqueue
   state UP mode DEFAULT group default qlen 1000
   link/ether 06:a2:87:5d:8b:b0 brd ff:ff:ff:ff:ff:ff
6 7: rout1-eth1.200@tunnel: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1458 qdisc noqueue
   state UP mode DEFAULT group default qlen 1000
   link/ether 06:a2:87:5d:8b:b0 brd ff:ff:ff:ff:ff:ff
7 23: rout1-eth0@if22: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state
   UP mode DEFAULT group default qlen 1000
   link/ether 0a:bd:90:9c:2a:72 brd ff:ff:ff:ff:ff:ff link-netnsid 0
```

```

14 25: rout1-eth1@if24: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master
    tunnel state UP mode DEFAULT group default qlen 1000
15 link/ether 76:b3:dc:b5:15:bb brd ff:ff:ff:ff:ff:ff link-netnsid 0

```

### III..2 Routeur 2

```

1 # Cr ation du tunnel: l'interface IF2 ici l2tpeth0 de 172.16.2.253 -> 172.16.1.253
2 ip netns exec rout2 ip l2tp add tunnel remote 172.16.1.253 local 172.16.2.253 encap
    ip tunnel_id 4000 peer_tunnel_id 3000
3 ip netns exec rout2 ip l2tp add session tunnel_id 4000 session_id 2000
    peer_session_id 1000
4 # activation de l'interface l2tpeth0
5 ip netns exec rout2 ip link set l2tpeth0 up
6 # ajout du bridge "tunnel"
7 ip netns exec rout2 brctl addbr tunnel
8 # ajout dans de l'interface l2tpeth0 au bridge
9 ip netns exec rout2 brctl addif tunnel l2tpeth0
10 # ajout dans de l'interface connect au r seau (rout2-eth1) au bridge
11 ip netns exec rout2 brctl addif tunnel rout2-eth1
12 # activation de l'interface correspondant au bridge
13 ip netns exec rout2 ip link set tunnel up
14 # configuration de l tiquettage VLAN pour le VLAN 100
15 ip netns exec rout2 ip netns exec rout2 ip link add link tunnel name rout2-eth1.100
    type vlan id 100
16 # activation de l'interface tunnel.100
17 ip netns exec rout2 ip link set rout2-eth1.100 up
18 # configuration de l tiquettage VLAN pour le VLAN 200
19 ip netns exec rout2 ip netns exec rout2 ip link add link tunnel name rout2-eth1.200
    type vlan id 200
20 # activation de l'interface tunnel.200
21 ip netns exec rout2 ip link set rout2-eth1.200 up
22 # configuration IP de l interface
23 ip netns exec rout2 ip addr add 192.168.100.253/24 dev rout2-eth1.100
24 ip netns exec rout2 ip addr add 192.168.200.253/24 dev rout2-eth1.200

```

Une fois le tunnel en place on peut remarquer les nouvelles interfaces.

```

1 [rout2] ip l
2 1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT
    group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
3 4: l2tpeth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1458 qdisc fq_codel master tunnel
    state UNKNOWN mode DEFAULT group default qlen 1000
    link/ether 56:e2:f3:fd:1e:2c brd ff:ff:ff:ff:ff:ff
4 5: tunnel: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1458 qdisc noqueue state UP mode
    DEFAULT group default qlen 1000
    link/ether 56:e2:f3:fd:1e:2c brd ff:ff:ff:ff:ff:ff
5 6: rout2-eth1.100@tunnel: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1458 qdisc noqueue
    state UP mode DEFAULT group default qlen 1000
    link/ether 56:e2:f3:fd:1e:2c brd ff:ff:ff:ff:ff:ff
6 7: rout2-eth1.200@tunnel: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1458 qdisc noqueue
    state UP mode DEFAULT group default qlen 1000
    link/ether 56:e2:f3:fd:1e:2c brd ff:ff:ff:ff:ff:ff
7 31: rout2-eth0@if30: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state
    UP mode DEFAULT group default qlen 1000

```

```

13 link/ether 42:43:78:ee:3a:12 brd ff:ff:ff:ff:ff:ff link-netnsid 0
14 33: rout2-eth1@if32: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master
    tunnel state UP mode DEFAULT group default qlen 1000
15 link/ether 5a:cb:33:b3:4e:d8 brd ff:ff:ff:ff:ff:ff link-netnsid 0

```

- a. Le protocole ARP fonctionne normalement pour la découverte des machines quelle que soit leur côté de connexion par rapport à Internet :

```

1 [rout1] arp -n
2 Adresse                TypeMap AdresseMat      Indicateurs
   Iface
3 192.168.100.115        ether    3e:8e:42:9d:7e:16      C
   rout1-eth1.100
4 172.16.1.254           ether    72:e6:7d:98:03:b2      C
   rout1-eth0
5 192.168.200.36         ether    3a:6c:1c:e5:7c:bf      C
   rout1-eth1.200
6 192.168.200.112        ether    22:91:4b:a1:7c:47      C
   rout1-eth1.200
7 192.168.100.93         ether    12:39:de:47:41:00      C
   rout1-eth1.100

```

On remarque bien que le protocole ARP fonctionne en visualisant l'adresse mac **12 :39 :de :47 :41 :00** et ip de **192.168.100.93** du poste1 dans la table ARP.

```

1 [poste1] ip a
2 1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
   default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
   inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
   inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
3 2: poste1-eth0.100@poste1-eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500
   qdisc noqueue state UP group default qlen 1000
   link/ether 12:39:de:47:41:00 brd ff:ff:ff:ff:ff:ff
   inet 192.168.100.93/24 brd 192.168.100.255 scope global poste1-eth0.100
       valid_lft forever preferred_lft forever
   inet6 fe80::1039:deff:fe47:4100/64 scope link
       valid_lft forever preferred_lft forever
14 15: poste1-eth0@if14: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue
   state UP group default qlen 1000
   link/ether 12:39:de:47:41:00 brd ff:ff:ff:ff:ff:ff link-netnsid 0
   inet6 fe80::1039:deff:fe47:4100/64 scope link
17   valid_lft forever preferred_lft forever

```

- b. le service DHCP que vous n'exécuterez que sur « Routeur1 » peut être utilisé par Poste1. Nous lançons le serveur DHCP sur le routeur 1 avec la commande ci-dessous, en spécifiant la bonne interface Vlan, dans ce cas rout1-eth1.100

```

1 ip netns exec rout1 dnsmasq -d -z -i rout1-eth1.100 -F
   192.168.100.1,192.168.100.150,255.255.255.0

```

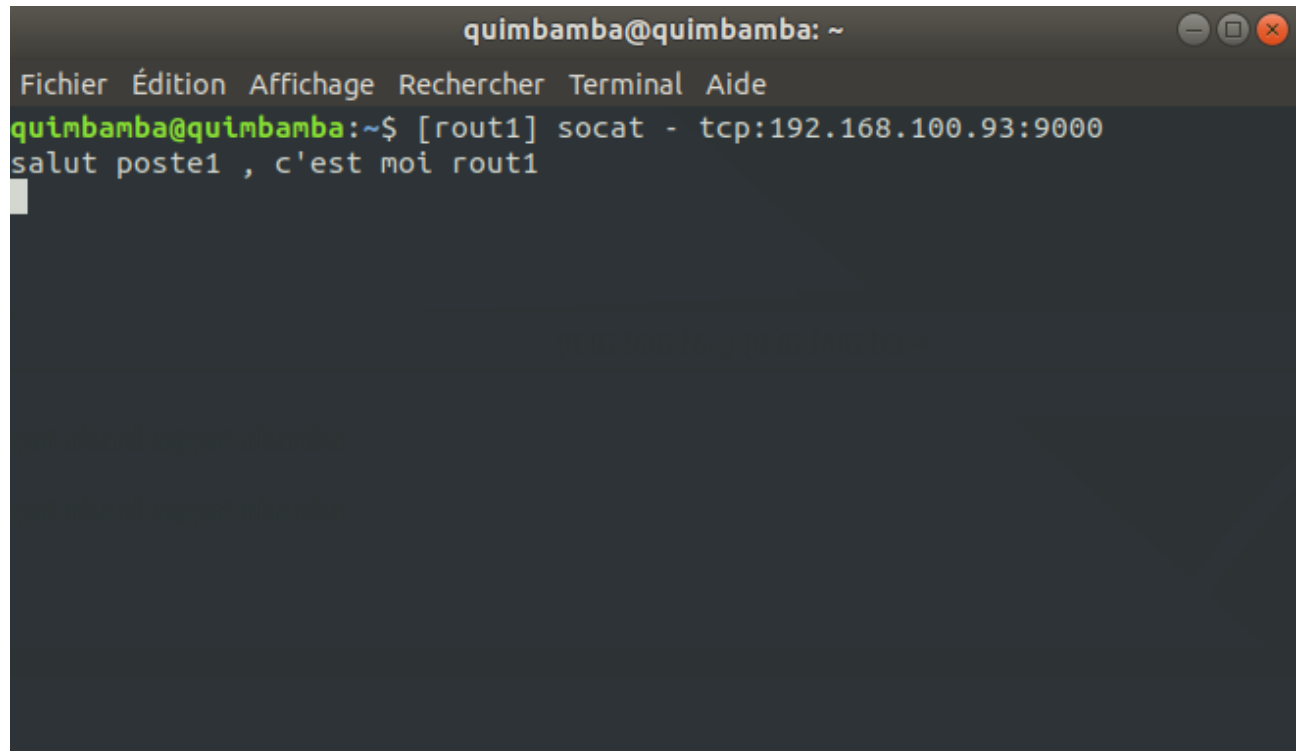
Etant donné que le DHCP a été lancé par le routeur 1 donc la route par défaut du poste1 est via rout1

```

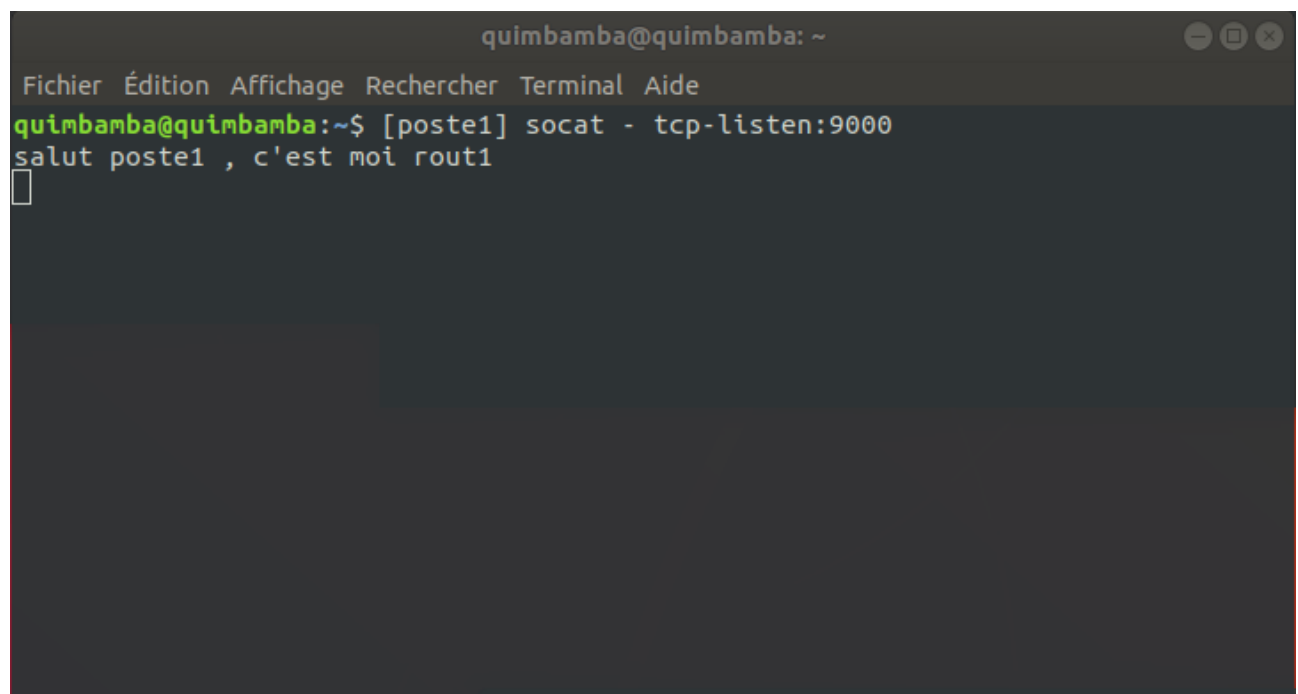
1 [poste1] ip r
2 default via 192.168.100.254 dev poste1-eth0.100
3 192.168.100.0/24 dev poste1-eth0.100 proto kernel scope link src
   192.168.100.93

```

- c. une connexion TCP à l'aide de socat entre Routeur1 et Poste1 est possible à travers le tunnel.



```
quimbamba@quimbamba: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
quimbamba@quimbamba:~$ [rout1] socat - tcp:192.168.100.93:9000  
salut poste1 , c'est moi rout1
```

A terminal window titled 'quimbamba@quimbamba: ~' with a menu bar containing 'Fichier', 'Édition', 'Affichage', 'Rechercher', 'Terminal', and 'Aide'. The prompt is 'quimbamba@quimbamba:~\$'. The command '[poste1] socat - tcp-listen:9000' has been entered. Below it, the received message 'salut poste1 , c'est moi rout1' is displayed, followed by a cursor on a new line.

```
quimbamba@quimbamba: ~
Fichier  Édition  Affichage  Rechercher  Terminal  Aide
quimbamba@quimbamba:~$ [poste1] socat - tcp-listen:9000
salut poste1 , c'est moi rout1
█
```

## IV Comparaison des tunnel L2tpv3 (ip ,udp) vs GrepTap

Comme mentionné en dessus , un tunnel L2tpv3 , est un protocole d'encapsulation point à point d'un protocole quelconque de niveau 2 dans IP. C'est une évolution importante de L2TPv2 qui n'autorise que l'encapsulation du protocole PPP, il utilise une session L2TPv3 entre deux équipements est appelée un pseudo-wire.

Plus complet que GRE, L2TPv3 peut être associé à une authentification AAA (Authentication, Authorization and Accounting), pour offrir des services de type VPN, apportant une alternative simple à MPLS.

Nous allons dans la suite mettre en place ces deux types de tunnel et étudier leur fonctionnement.

IV..1    Teste de connectivité entre poste1 et 3 pour le tunnel en mode ip , que nous avons mis en dessus.

```
quimbamba@quimbamba: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
quimbamba@quimbamba:~$ [poste3] socat - tcp:192.168.100.93:9000  
salut poste1  
  
...  
... 16: 83 bytes on wire (664 bits), 83 bytes captured (664 bits) on interface 0  
Ethernet II, Src: 3e:8a:42:9d:7e:16 (3e:8a:42:9d:7e:16), Dst: 12:39:de:47:41:08 (12:39:de:47:41:08)  
VLAN Virtual LAN, PRI: 0, DEI: 0, ID: 100  
Internet Protocol Version 4, Src: 192.168.100.115, Dst: 192.168.100.93  
Transmission Control Protocol, Src Port: 43326, Dst Port: 9000, Seq: 1, Ack: 1, Len: 13  
  
Destination Port: 9000  
[Stream index: 0]  
TCP Segment Len: 13  
Sequence number: 1 (relative sequence number)  
Next sequence number: 14 (relative sequence number)]  
Acknowledgment number: 1 (relative ack number)  
1000 .... = Header Length: 32 bytes (8)  
Flags: 0x018 (PSH, ACK)  
Window size value: 592  
  
12 39 de 47 41 08 3e 8a 42 9d 7e 16 81 00 00 64 ... 0-64 > .B ..... d
```

[illegible]

On remarque que la MTU, du tunnel en mode ip est la suivante :

```

quimbamba@quimbamba: ~
Fichier Édition Affichage Rechercher Terminal Aide
quimbamba@quimbamba:~$ [rout2] ip -d l show tunnel
5: tunnel: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1458 qdisc noqueue state UP mode DEFAULT group default qlen 1000
    link/ether 56:e2:f3:fd:1e:2c brd ff:ff:ff:ff:ff:ff promiscuity 0
    bridge forward_delay 1500 hello_time 200 max_age 2000 ageing_time 30000 stp_state 0 priority 32768 vlan_filtering 0 vlan_protocol
    802.1Q bridge_id 8000.56:e2:f3:fd:1e:2c designated_root 8000.56:e2:f3:fd:1e:2c root_port 0 root_path_cost 0 topology_change 0 topolo
    gy_change_detected 0 hello_timer 0.00 tcn_timer 0.00 topology_change_timer 0.00 gc_timer 86.14 vlan_default_pvid 1 vlan_st
    ats_enabled 0 group_fwd_mask 0 group_address 01:80:c2:00:00:00 mcast_snooping 1 mcast_router 1 mcast_query_use_ifaddr 0 mcast_querier
    0 mcast_hash_elasticity 16 mcast_hash_max 4096 mcast_last_member_count 2 mcast_startup_query_count 2 mcast_last_member_interval 100
    mcast_membership_interval 26000 mcast_querier_interval 25500 mcast_query_interval 12500 mcast_query_response_interval 1000 mcast_star
    tup_query_interval 3124 mcast_stats_enabled 0 mcast_igmp_version 2 mcast_mld_version 1 nf_call_iptables 0 nf_call_ip6tables 0 nf_call
    arptables 0 addrgenmode eui64 numtxqueues 1 numrxqueues 1 gso_max_size 65536 gso_max_segs 65535
quimbamba@quimbamba:~$ [rout2]

```

IV..2 La configuration du tunnel en mode udp est la suivante :

### IV..3 Routeur 1

```

1 # rout1 ==
2 # Cr ation du tunnel: l'interface IF2 ici l2tpeth0 de 172.16.1.253 -> 172.16.2.253
3 ip netns exec rout1 ip l2tp add tunnel remote 172.16.2.253 local 172.16.1.253 encaps
   udp tunnel_id 3000 peer_tunnel_id 4000 udp_sport 5050 udp_dport 9090
4 ip netns exec rout1 ip l2tp add session tunnel_id 3000 session_id 1000
   peer_session_id 2000
5 # Activation de l'interface l2tpeth0
6 ip netns exec rout1 ip link set l2tpeth0 up
7 # ajout du bridge "tunnel"
8 ip netns exec rout1 brctl addbr tunnel
9 # ajout dans de l'interface l2tpeth0 au bridge
10 ip netns exec rout1 brctl addif tunnel l2tpeth0
11 # ajout dans de l'interface connect au r seau (rout1-eth1) au bridge
12 ip netns exec rout1 brctl addif tunnel rout1-eth1
13 # activation de l'interface correspondant au bridge
14 ip netns exec rout1 ip link set tunnel up
15 # configuration de l'interface VLAN pour le VLAN 100
16 ip netns exec rout1 ip netns exec rout1 ip link add link tunnel name rout1-eth1.100
   type vlan id 100
17 # activation de l'interface tunnel.100
18 ip netns exec rout1 ip link set rout1-eth1.100 up
19 # configuration de l'interface VLAN pour le VLAN 200
20 ip netns exec rout1 ip netns exec rout1 ip link add link tunnel name rout1-eth1.200
   type vlan id 200
21 # activation de l'interface tunnel.200
22 ip netns exec rout1 ip link set rout1-eth1.200 up

```

### IV..4 Routeur 2

```

1 #rout2 ==
2 # Cr ation du tunnel: l'interface IF2 ici l2tpeth0 de 172.16.2.253 -> 172.16.1.253
3 ip netns exec rout2 ip l2tp add tunnel remote 172.16.1.253 local 172.16.2.253 encaps
   udp tunnel_id 4000 peer_tunnel_id 3000 udp_sport 9090 udp_dport 5050
4 ip netns exec rout2 ip l2tp add session tunnel_id 4000 session_id 2000
   peer_session_id 1000
5 # activation de l'interface l2tpeth0

```

```

6 ip netns exec rout2 ip link set l2tpeth0 up
7 # ajout du bridge "tunnel"
8 ip netns exec rout2 brctl addbr tunnel
9 # ajout dans de l'interface l2tpeth0 au bridge
10 ip netns exec rout2 brctl addif tunnel l2tpeth0
11 # ajout dans de l'interface connect au r seau (rout2-eth1) au bridge
12 ip netns exec rout2 brctl addif tunnel rout2-eth1
13 # activation de l'interface correspondant au bridge
14 ip netns exec rout2 ip link set tunnel up
15 # configuration de l'interface VLAN pour le VLAN 100
16 ip netns exec rout2 ip netns exec rout2 ip link add link tunnel name rout2-eth1.100
    type vlan id 100
17 # activation de l'interface tunnel.100
18 ip netns exec rout2 ip link set rout2-eth1.100 up
19 # configuration de l'interface VLAN pour le VLAN 200
20 ip netns exec rout2 ip netns exec rout2 ip link add link tunnel name rout2-eth1.200
    type vlan id 200
21 # activation de l'interface tunnel.200
22 ip netns exec rout2 ip link set rout2-eth1.200 up

```

#### IV..5 Teste de connectivité entre poste1 et 3 pour le tunnel en mode udp

```

quimbamba@quimbamba: ~
Fichier Édition Affichage Rechercher Terminal Aide
quimbamba@quimbamba:~$ [rout1] sudo ip l2tp show tunnel
Tunnel 3000, encap UDP
  From 172.16.1.253 to 172.16.2.253
  Peer tunnel 4000
  UDP source / dest ports: 5050/9090
  UDP checksum: disabled
quimbamba@quimbamba:~$ [rout1]

```

```

quimbamba@quimbamba: ~
Fichier Édition Affichage Rechercher Terminal Aide
quimbamba@quimbamba:~$ [poste3] ip a show dev poste3-eth0.100
2: poste3-eth0.100@poste3-eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
  noqueue state UP group default qlen 1000
  link/ether da:55:b2:28:c6:1e brd ff:ff:ff:ff:ff:ff
  inet 192.168.100.20/24 brd 192.168.100.255 scope global poste3-eth0.100
    valid_lft forever preferred_lft forever
  inet6 fe80::d855:b2ff:fe28:c61e/64 scope link
    valid_lft forever preferred_lft forever
quimbamba@quimbamba:~$ [poste3]

```

```

quimbamba@quimbamba: ~
Fichier Édition Affichage Rechercher Terminal Aide
quimbamba@quimbamba:~$ [poste1] ping -c 3 192.168.100.20
PING 192.168.100.20 (192.168.100.20) 56(84) bytes of data.
64 bytes from 192.168.100.20: icmp_seq=1 ttl=64 time=2.85 ms
64 bytes from 192.168.100.20: icmp_seq=2 ttl=64 time=0.248 ms
64 bytes from 192.168.100.20: icmp_seq=3 ttl=64 time=0.240 ms

--- 192.168.100.20 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2025ms
rtt min/avg/max/mdev = 0.240/1.113/2.851/1.228 ms

```

La MTU du tunnel en mode udp est :



```

quimbamba@quimbamba:~$ [rout1] ip l show dev tunnel
5: tunnel: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1446 qdisc noqueue state UP mod
e DEFAULT group default qlen 1000
    link/ether 62:98:bb:54:cd:ce brd ff:ff:ff:ff:ff:ff
quimbamba@quimbamba:~$ [rout1]

```

#### IV..6 Configuration des interface pour le tunnel GrepTap

La configuration du tunnel GrepTap

#### IV..7 Routeur 1

```

1 # rout1
2 ip netns exec rout1 ip link add mon_tunnel type gretap remote 172.16.2.253 local
   172.16.1.253 nopmtudisc
3 ip netns exec rout1 ip link set mon_tunnel up
4 ip netns exec rout1 brctl addbr tunnel_gre
5 ip netns exec rout1 brctl addif tunnel_gre mon_tunnel
6 ip netns exec rout1 brctl addif tunnel_gre rout1-eth1
7 ip netns exec rout1 ip link set tunnel_gre up
8 ip netns exec rout1 ip link add link tunnel_gre name rout1-eth1.100 type vlan id 100
9 ip netns exec rout1 ip link set rout1-eth1.100 up
10 ip netns exec rout1 ip link add link tunnel_gre name rout1-eth1.200 type vlan id 200
11 ip netns exec rout1 ip link set rout1-eth1.200 up
12 ip netns exec rout1 ip addr add 192.168.100.254/24 dev rout1-eth1.100
13 ip netns exec rout1 ip addr add 192.168.200.254/24 dev rout1-eth1.200

```

#### IV..8 Routeur 1

```

1 # rout2
2 ip netns exec rout2 ip link add mon_tunnel type gretap remote 172.16.1.253 local
   172.16.2.253 nopmtudisc
3 ip netns exec rout2 ip link set mon_tunnel up
4 ip netns exec rout2 brctl addbr tunnel_gre
5 ip netns exec rout2 brctl addif tunnel_gre mon_tunnel
6 ip netns exec rout2 brctl addif tunnel_gre rout2-eth1
7 ip netns exec rout2 ip link set tunnel_gre up
8 ip netns exec rout2 ip link add link tunnel_gre name rout2-eth1.100 type vlan id 100
9 ip netns exec rout2 ip link set rout2-eth1.100 up
10 ip netns exec rout2 ip link add link tunnel_gre name rout2-eth1.200 type vlan id 200
11 ip netns exec rout2 ip link set rout2-eth1.200 up
12 ip netns exec rout2 ip addr add 192.168.100.253/24 dev rout2-eth1.100
13 ip netns exec rout2 ip addr add 192.168.200.253/24 dev rout2-eth1.200

```

#### IV..9 Teste de connectivité entre poste1 et 3 pour le tunnel GreTap

```
quimbamba@quimbamba: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
RTNETLINK answers: Cannot assign requested address  
quimbamba@quimbamba:~$ [poste3] ping 192.168.100.93  
PING 192.168.100.93 (192.168.100.93) 56(84) bytes of data. length 126  
64 bytes from 192.168.100.93: icmp_seq=1 ttl=64 time=2.71 ms  
64 bytes from 192.168.100.93: icmp_seq=2 ttl=64 time=0.337 ms  
64 bytes from 192.168.100.93: icmp_seq=3 ttl=64 time=0.353 ms  
64 bytes from 192.168.100.93: icmp_seq=4 ttl=64 time=0.333 ms  
64 bytes from 192.168.100.93: icmp_seq=5 ttl=64 time=0.298 ms  
64 bytes from 192.168.100.93: icmp_seq=6 ttl=64 time=0.278 ms  
64 bytes from 192.168.100.93: icmp_seq=7 ttl=64 time=0.256 ms  
64 bytes from 192.168.100.93: icmp_seq=8 ttl=64 time=0.313 ms  
64 bytes from 192.168.100.93: icmp_seq=9 ttl=64 time=0.301 ms  
64 bytes from 192.168.100.93: icmp_seq=10 ttl=64 time=0.349 ms  
64 bytes from 192.168.100.93: icmp_seq=11 ttl=64 time=0.301 ms  
64 bytes from 192.168.100.93: icmp_seq=12 ttl=64 time=0.294 ms  
64 bytes from 192.168.100.93: icmp_seq=13 ttl=64 time=0.296 ms  
64 bytes from 192.168.100.93: icmp_seq=14 ttl=64 time=0.342 ms
```

```
quimbamba@quimbamba: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
2022-05-07 19:47:27.937383 IP (tos 0x0, ttl 64, id 33554, offset 0, flags [none], proto GRE (47), length 126)  
172.16.1.253 > 172.16.2.253: GREv0, Flags [none], length 106  
IP (tos 0x0, ttl 64, id 63710, offset 0, flags [DF], proto ICMP (1), length 84)  
192.168.100.115 > 192.168.100.93: ICMP echo request, id 15080, seq 1, length 64  
0x0000: 4500 007e 8312 0000 402f 9a24 ac10 01fd E...XV13t...  
0x0010: ac10 02fd 0000 6558 9e03 578e 23c2 7631 .....eX.W.#.v1  
0x0020: 3374 c95f 8100 0064 0800 4500 0054 f8de 3t.....E..T..  
0x0030: 4000 4001 f7a8 c0a8 6473 c0a8 645d 0800 @.0.....ds..d..  
0x0040: 57ea 3ae8 0001 afb0 7662 0000 0000 7246 W:.....vb....rF  
0x0050: 0e00 0000 0000 1011 1213 1415 1617 1819 .....  
0x0060: 1a1b 1c1d 1e1f 2021 2223 2425 2627 2829 .....!"#$%&'()  
0x0070: 2a2b 2c2d 2e2f 3031 3233 3435 3637 *+,-./01234567  
2022-05-07 19:47:27.937948 IP (tos 0x0, ttl 62, id 34059, offset 0, flags [none], proto GRE (47), length 126)  
172.16.2.253 > 172.16.1.253: GREv0, Flags [none], length 106  
IP (tos 0x0, ttl 64, id 34559, offset 0, flags [none], proto ICMP (1), length 84)  
192.168.100.93 > 192.168.100.115: ICMP echo reply, id 15080, seq 1, length 64  
0x0000: 4500 007e 850b 0000 3e2f 9a2b ac10 02fd E...XV13t...  
0x0010: ac10 01fd 0000 6558 7631 3374 c95f 9e03 .....eXV13t...  
0x0020: 578e 23c2 8100 0064 0800 4500 0054 86ff W.#.....d..E..T..  
0x0030: 0000 4001 a988 c0a8 645d c0a8 6473 0000 .0.....d]..ds..  
0x0040: 5fea 3ae8 0001 afb0 7662 0000 0000 7246 -:.....vb....rF  
0x0050: 0e00 0000 0000 1011 1213 1415 1617 1819 .....  
0x0060: 1a1b 1c1d 1e1f 2021 2223 2425 2627 2829 .....!"#$%&'()  
0x0070: 2a2b 2c2d 2e2f 3031 3233 3435 3637 *+,-./01234567  
2022-05-07 19:47:28.937168 IP (tos 0x0, ttl 64, id 33660, offset 0, flags [none], proto GRE (47), length 126)  
172.16.1.253 > 172.16.2.253: GREv0, Flags [none], length 106  
IP (tos 0x0, ttl 64, id 63816, offset 0, flags [DF], proto ICMP (1), length 84)
```

La MTU pour le tunnel gretap est la suivante

```

quimbamba@quimbamba: ~
Fichier  Édition  Affichage  Rechercher  Terminal  Aide

quimbamba@quimbamba:~$ [rout2] ip -d l show mon_tunnel
7: mon_tunnel@NONE: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1462 qdisc fq_codel master tunnel_gre sta
te UNKNOWN mode DEFAULT group default qlen 1000
    link/ether 9a:4f:88:bd:6a:90 brd ff:ff:ff:ff:ff:ff promiscuity 1
    gretap remote 172.16.1.253 local 172.16.2.253 ttl inherit nopmtudisc
    bridge_slave state forwarding priority 32 cost 100 hairpin off guard off root_block off fastlea
ve off learning on flood on port_id 0x8001 port_no 0x1 designated_port 32769 designated_cost 0 desi
gnated_bridge 8000.9a:4f:88:bd:6a:90 designated_root 8000.9a:4f:88:bd:6a:90 hold_timer 0.00 mess
age_age_timer 0.00 forward_delay_timer 0.00 topology_change_ack 0 config_pending 0 proxy_arp
off proxy_arp_wifi off mcast_router 1 mcast_fast_leave off mcast_flood on neigh_suppress off group_
fwd_mask 0x0 group_fwd_mask_str 0x0 vlan_tunnel off addrngenmode eui64 numtxqueues 1 numrxqueues 1 g
so_max_size 65536 gso_max_segs 65535
quimbamba@quimbamba:~$ [rout2]

```

## V Mise en place du chiffrement IPsec sur le tunnel l2TPv3 en encapsulation IP.

En utilisant la fiche du TP2, nous avons mis en place un chiffrement IPsec sur le tunnel l2TPv3 en encapsulation IP, avec la configuration suivante. Avec la commande **iperf**, nous allons calculer le débit sans et avec le chiffrement IPsec.

### V.I Débit sans le chiffrement Ipsec

```

1 [rout1] iperf -c 192.168.100.93 -f
2 iperf: option requires an argument -- f
3 -----
4 Client connecting to 192.168.100.93, TCP port 5001
5 TCP window size: 85.0 KByte (default)
6 -----
7 [ 3] local 192.168.100.254 port 38980 connected with 192.168.100.93 port 5001
8 [ ID] Interval      Transfer      Bandwidth
9 [ 3] 0.0-10.0 sec  1.94 GBytes  1.66 Gbits/sec

```

Le débit du tunnel sans le chiffrement IPsec est d'environ 1.66 Gbits/sec

#### Configuration du chiffrement IPsec

```

1 #rout1 ==
2 # rout1
3 # Flush the SAD and SPD
4 ip netns exec rout1 ip xfrm state flush
5 ip netns exec rout1 ip xfrm policy flush
6
7 # AH SAs using 256 bit long and ESP SAs using 160 bit long keys
8 ip netns exec rout1 ip xfrm state add src 172.16.1.253 dst 172.16.2.253 proto esp
   spi 0x12345678 reqid 0x12345678 mode tunnel auth sha256 0
   x323730ed6f1b9ff0cb084af15b197e862b7c18424a7cdfb74cd385ae23bc4f17 enc "rfc3686(
   ctr(aes))" 0x27b90b8aeca1ee32a8150a664e8faac761e2d305b
9 ip netns exec rout1 ip xfrm state add src 172.16.2.253 dst 172.16.1.253 proto esp
   spi 0x12345678 reqid 0x12345678 mode tunnel auth sha256 0
   x44d65c50b7581fd3c8169cf1fa0ebb24e0d55755b1dc43a98b539bb144f2067f enc "rfc3686(
   ctr(aes))" 0x9df7983cb7c7eb2af01d88d36e462b5f01d10bc1

```

```

10
11
12 # Security policies
13 ip netns exec rout1 ip xfrm policy add src 172.16.2.253 dst 172.16.1.253 dir in
    tmpl src 172.16.2.253 dst 172.16.1.253 proto esp reqid 0x12345678 mode tunnel
14 ip netns exec rout1 ip xfrm policy add src 172.16.1.253 dst 172.16.2.253 dir out
    tmpl src 172.16.1.253 dst 172.16.2.253 proto esp reqid 0x12345678 mode tunnel
15
16
17 # rout2 ==
18 #rout2
19 # Flush the SAD and SPD
20 ip netns exec rout2 ip xfrm state flush
21 ip netns exec rout2 ip xfrm policy flush
22
23 # AH SAs using 256 bit long and ESP SAs using 160 bit long keys
24 ip netns exec rout2 ip xfrm state add src 172.16.2.253 dst 172.16.1.253 proto esp
    spi 0x12345678 reqid 0x12345678 mode tunnel auth sha256 0
    x323730ed6f1b9ff0cb084af15b197e862b7c18424a7cdfb74cd385ae23bc4f17 enc "rfc3686(
    ctr(aes))" 0x27b90b8aec1ee32a8150a664e8faac761e2d305b
25 ip netns exec rout2 ip xfrm state add src 172.16.1.253 dst 172.16.2.253 proto esp
    spi 0x12345678 reqid 0x12345678 mode tunnel auth sha256 0
    x44d65c50b7581fd3c8169cf1fa0ebb24e0d55755b1dc43a98b539bb144f2067f enc "rfc3686(
    ctr(aes))" 0x9df7983cb7c7eb2af01d88d36e462b5f01d10bc1
26
27
28 # Security policies
29 ip netns exec rout2 ip xfrm policy add src 172.16.1.253 dst 172.16.2.253 dir out
    tmpl src 172.16.1.253 dst 172.16.2.253 proto esp reqid 0x12345678 mode tunnel
30 ip netns exec rout2 ip xfrm policy add src 172.16.2.253 dst 172.16.1.253 dir in tmpl
    src 172.16.2.253 dst 172.16.1.253 proto esp reqid 0x12345678 mode tunnel

```

## V.II Débit avec le chiffrement Ipsec

```

1 iperf -c 192.168.100.93 -f
2 iperf: option requires an argument -- f
3 -----
4 Client connecting to 192.168.100.93, TCP port 5001
5 TCP window size: 85.0 KByte (default)
6 -----
7 [ 3] local 192.168.100.254 port 38984 connected with 192.168.100.93 port 5001
8 [ ID] Interval      Transfer      Bandwidth
9 [ 3] 0.0-10.0 sec   1.85 GBytes  1.59 Gbits/sec

```

Le débit du tunnel avec le chiffrement ipsec est d'environ 1.59 Gbits/sec

## VI Accès Internet « intelligent »

On active une interface du switch dans notre machine et on donne à cette interface une adresse du sous-réseau internet1

```
1 ip link set internet1 up
2 ip address add 10.87.0.3/24 dev internet1
```

On ajoute sur routA une route par défaut vers l'interface que l'on vient d'activer.

```
1 ip netns exec routA ip r add default via 10.87.0.3
```

On fait du nat sur notre machine pour avoir une réponse en retour

```
1 iptables -t nat -A POSTROUTING -s 10.87.0.0/24 -j MASQUERADE
```

A cette stade , le route A et B peuvent pinger **8.8.8.8**

```
quimbamba@quimbamba: ~
Fichier Édition Affichage Rechercher Terminal Aide
quimbamba@quimbamba:~$ [routA] ping -c 3 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=119 time=14.9 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=119 time=20.8 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=119 time=15.2 ms

--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 14.999/17.056/20.899/2.719 ms
quimbamba@quimbamba:~$ [routA]
```

```
quimbamba@quimbamba: ~
Fichier Édition Affichage Rechercher Terminal Aide
quimbamba@quimbamba:~$ [routB] ping -c 3 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=119 time=14.8 ms
From 10.87.0.1: icmp_seq=2 Redirect Host(New nexthop: 10.87.0.3)
64 bytes from 8.8.8.8: icmp_seq=2 ttl=119 time=15.3 ms
From 10.87.0.1: icmp_seq=3 Redirect Host(New nexthop: 10.87.0.3)
64 bytes from 8.8.8.8: icmp_seq=3 ttl=119 time=14.3 ms

--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2002ms
rtt min/avg/max/mdev = 14.316/14.834/15.353/0.434 ms
quimbamba@quimbamba:~$ [routB]
```

Pour activer l'accès sur internet au routeur 1 et aux réseaux du VLAN , on utilise la même logique qu'en dessus.

On active le nat sur notre machine pour avoir la réponse vers le réseau **172.16.1.0/24** le sous réseau où se trouve le route 1

```
1 iptables -t nat -A POSTROUTING -s 172.16.1.0/24 -j MASQUERADE
```

Et sur le routeur 1 on fait pareil, pour avoir la réponse vers les sous réseaux du VLAN.

```
1 ip netns exec rout1 iptables -t nat -A POSTROUTING -s 192.168.100.0/24 -j MASQUERADE
2 ip netns exec rout1 iptables -t nat -A POSTROUTING -s 192.168.200.0/24 -j MASQUERADE
```

c. vous vérifierez que le trafic de Poste1 passe bien par Routeur1;  
ping depuis poste3

```
quimbamba@quimbamba: ~
Fichier Édition Affichage Rechercher Terminal Aide
quimbamba@quimbamba:~$ [poste1] ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=117 time=16.3 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=117 time=15.2 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=117 time=14.6 ms
^C
--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 14.636/15.417/16.374/0.720 ms
```

```
quimbamba@quimbamba:~$ [rout1] sudo tcpdump -lnvvsX host 8.8.8.8
tcpdump: listening on l2tpeth0, link-type EN10MB (Ethernet), capture size 262144 bytes
06:35:30.437289 IP (tos 0x0, ttl 64, id 60488, offset 0, flags [DF], proto ICMP (1), length 84)
 192.168.100.93 > 8.8.8.8: ICMP echo request, id 5975, seq 1, length 64
    0x0000: 4500 0054 ec48 4000 4001 194b c0a8 645d E..T.H@.@..K..d]
    0x0010: 0808 0808 0800 5b7f 1757 0001 9248 7762 .....[.W...Hwb
    0x0020: 0000 0000 b6aa 0600 0000 0000 1011 1213 .....
    0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!"#
    0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 $%&'()*+,-./0123
    0x0050: 3435 3637 4567
06:35:31.438665 IP (tos 0x0, ttl 64, id 60566, offset 0, flags [DF], proto ICMP (1), length 84)
 192.168.100.93 > 8.8.8.8: ICMP echo request, id 5975, seq 2, length 64
    0x0000: 4500 0054 ec96 4000 4001 18fd c0a8 645d E..T..@.@.....d]
    0x0010: 0808 0808 0800 1778 1757 0002 9348 7762 .....X.W...Hwb
    0x0020: 0000 0000 f9b0 0600 0000 0000 1011 1213 .....
    0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!"#
    0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 $%&'()*+,-./0123
    0x0050: 3435 3637 4567
```

Dans la capture en dessus on remarque bien que le paquet du poste1 passe par route 1 , en effet , cela arrive puisque le poste1 a eu sa configuration depuis router 1, on peut rediriger les postes Poste1 et Poste2, lors de leur configuration par DHCP, vers Routeur 2 au lieu de Routeur 1 pour optimiser l'accès à Internet. L'option à ajouter sur la commande **dnsmasq** pour faire cette redirection, est **-dhcp-option= option :router, adresse ip du router par default** l'option **-dhcp-option** peut être simplifié en **-O**

```
1 dnsmasq -d -z -i rout1-eth1.100 -F 192.168.100.1,192.168.100.150,255.255.255.0 -O
  option:router,192.168.100.253
```

On remarque que la route par default du poste1 est le routeur 2 et que le ping vers 8.8.8.8 passe par le routeur 2.

```
quimbamba@quimbamba: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
quimbamba@quimbamba:~$ [poste1] sudo dhclient  
quimbamba@quimbamba:~$ [poste1] ip r  
default via 192.168.100.253 dev poste1-eth0.100  
192.168.100.0/24 dev poste1-eth0.100 proto kernel scope link src 192.168.100.93  
quimbamba@quimbamba:~$ [poste1]
```

```
quimbamba@quimbamba: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
quimbamba@quimbamba:~$ [rout2] sudo tcpdump -lnvvSX host 8.8.8.8  
tcpdump: listening on l2tpeth0, link-type EN10MB (Ethernet), capture size 262144 bytes  
08:23:57.279387 IP (tos 0x0, ttl 117, id 0, offset 0, flags [none], proto ICMP (1), length 84)  
8.8.8.8 > 192.168.100.93: ICMP echo reply, id 10825, seq 1, length 64  
0x0000: 4500 0054 0000 0000 7501 1094 0808 0808 E..T....u.....  
0x0010: c0a8 645d 0000 a325 2a49 0001 fd61 7762 ..d]...%*I...awb  
0x0020: 0000 0000 fbfb 0300 0000 0000 1011 1213 .....  
0x0030: 1415 1617 1819 1a1b 1c1d 1e1f 2021 2223 .....!"#  
0x0040: 2425 2627 2829 2a2b 2c2d 2e2f 3031 3233 $%&'()*+,-./0123  
0x0050: 3435 3637 4567
```

## VII interdiction du trafic entre VLAN 100 et VLAN 200

a. à l'aide de règles « iptables » ;

```
1 ip netns exec rout1 iptables -A FORWARD --in-interface rout1-eth1.100 --out-  
   interface rout1-eth1.200 -j REJECT  
2 ip netns exec rout1 iptables -A FORWARD --in-interface rout1-eth1.200 --out-  
   interface rout1-eth1.100 -j REJECT  
3  
4 ip netns exec rout2 iptables -A FORWARD --in-interface rout2-eth1.100 --out-  
   interface rout2-eth1.200 -j REJECT  
5 ip netns exec rout2 iptables -A FORWARD --in-interface rout2-eth1.200 --out-  
   interface rout2-eth1.100 -j REJECT
```

On remarque bien que le ping entre le poste3 et 4 ne marche pas.

```
quimbamba@quimbamba: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
quimbamba@quimbamba:~$ [poste3] ping -c 3 192.168.200.36  
PING 192.168.200.36 (192.168.200.36) 56(84) bytes of data.  
From 192.168.100.254 icmp_seq=1 Destination Port Unreachable  
From 192.168.100.254 icmp_seq=2 Destination Port Unreachable  
From 192.168.100.254 icmp_seq=3 Destination Port Unreachable  
  
--- 192.168.200.36 ping statistics ---  
3 packets transmitted, 0 received, +3 errors, 100% packet loss, time 2025ms
```

```
quimbamba@quimbamba: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
quimbamba@quimbamba:~$ [poste4] ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: poste4-eth0.200@poste4-eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc  
    noqueue state UP group default qlen 1000  
    link/ether 6e:ac:ad:96:79:3c brd ff:ff:ff:ff:ff:ff  
    inet 192.168.200.36/24 brd 192.168.200.255 scope global poste4-eth0.200  
        valid_lft forever preferred_lft forever  
    inet6 fe80::6cac:adff:fe96:793c/64 scope link  
        valid_lft forever preferred_lft forever  
13: poste4-eth0@if12: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue s  
    tate UP group default qlen 1000  
    link/ether 6e:ac:ad:96:79:3c brd ff:ff:ff:ff:ff:ff link-netnsid 0  
    inet6 fe80::6cac:adff:fe96:793c/64 scope link  
        valid_lft forever preferred_lft forever  
quimbamba@quimbamba:~$ [poste4]
```

b. à l'aide de la « Policy Routing ».

Voici les commandes pour la Policy Routing ,on peut également utiliser l'option backhole pour ne pas envoyer un packet de retour.

```
1 ip netns exec rout1 ip rule add from 192.168.100.0/24 to 192.168.200.0/24 prohibit  
2 ip netns exec rout1 ip rule add from 192.168.200.0/24 to 192.168.100.0/24 prohibit  
3  
4 ip netns exec rout2 ip rule add from 192.168.100.0/24 to 192.168.200.0/24 prohibit  
5 ip netns exec rout2 ip rule add from 192.168.200.0/24 to 192.168.100.0/24 prohibit
```

On remarque que le paquet sont filtré par notre réglé.

```
quimbamba@quimbamba: ~  
Fichier Édition Affichage Rechercher Terminal Aide  
quimbamba@quimbamba:~$ [poste3] ping -c 3 192.168.200.36  
PING 192.168.200.36 (192.168.200.36) 56(84) bytes of data.  
From 192.168.100.254 icmp_seq=1 Packet filtered  
From 192.168.100.254 icmp_seq=2 Packet filtered  
From 192.168.100.254 icmp_seq=3 Packet filtered  
  
--- 192.168.200.36 ping statistics ---  
3 packets transmitted, 0 received, +3 errors, 100% packet loss, time 2027ms
```



Comme nous l'avons vu au TD2 dans la partie **Les tables de routage multiples** , nous pouvons également interdire le trafic entre les deux Vlan en utilisant cette notion.

```
1 # ajouter une table pour router le trafic vers un Vlan
2 ip rule add from 192.168.100.0/24 to 192.168.200.0/24 prio 16000 table Matable
3 # interdire le trafic dans cette table
4 ip rule add prohibit 192.168.200.0/24 from 192.168.100.0/24 table Matable
```

## VIII Description du protocole L2TPv3 et de la technologie des VXLANs

### VIII.I Présentation rapide du protocole L2TPv3 et de la technologie des VXLANs

#### VIII.I.1 L2TPv3

Le protocole L2TP est largement utilisé , notamment pour la collecte des abonnés ADSL et transmission des données au FAI auprès duquel elles sont enregistrées (l'opérateur de collecte, celui qui a accès aux PBX (Private Branch Exchange) n'étant pas toujours le FAI). Cette RFC spécifie la nouvelle version, 3. Le principe de L2TP est simple : un protocole de contrôle permet d'établir des sessions L2TP au dessus d'un protocole non connectes permettant ensuite d'encapsuler et de désencapsuler les données du protocole de niveau 2 (typiquement Ethernet ou PPP) qui sont transportées. Le grand changement par rapport à la version 3 de L2TP est précisément une plus grande indépendance vis-à-vis du protocole transporté. L2TP peut fonctionner sur de l'UDP ou directement sur IP.

#### VIII.I.2 VXLANs

VXLAN signifie Virtual eXtensible Local Area Network, et est un moyen de résoudre les défis de mise à l'échelle des réseaux VLAN dans un environnement multi-tenant. VXLAN est un réseau superposé qui transporte un réseau L2 sur un réseau L3 existant.

### VIII.II Comparaison des deux solutions

Du point de vue de la commutation de paquets, VXLAN consiste simplement à coller une encapsulation au-dessus d'une trame L2 : quelque chose que d'autres protocoles font également.

La vraie différence se situe au niveau de la couche de contrôle et de gestion.

Les VXLANs sont utilisés dans les data center pour créer des réseaux de overlay au-dessus du réseau physique, les utilisateurs finaux du même segment de réseau ou de segments de réseau différents peuvent communiquer entre eux via les tunnels VXLAN. Le L2TPv3 est utilisé pour établir des connexions point à point, mais pas des connexions point à multipoint.

L2TPv3 ne peut être utilisé que lorsque les périphériques aux deux extrémités du tunnel se connectent aux VLAN ou utilisent des interfaces AC.

Un seul VLAN peut être configuré pour un tunnel L2TPv3.

### VIII.III La mise en œuvre dans Linux des VXLANs dans Openv Switch

Nous avons utilisé la procédure décrite sur le site suivante  
<https://purplepalmdash.github.io/2015/06/08/openvswitch-and-vxlan-how-to/>

#### VM Netorking Configuration

For VM1:

```
root@OpenVSwitchVM1:~# ovs-vsctl add-br br0
root@OpenVSwitchVM1:~# ovs-vsctl add-br br1
# ovs-vsctl add-port br0 eth0
# ifconfig eth0 0 up
# ifconfig br0 10.94.94.11
# route add default gw 10.94.94.1 br0
# ifconfig br1 172.10.0.1
```

For VM2:

```
# ovs-vsctl add-br br0
# ovs-vsctl add-br br1
# ovs-vsctl add-port br0 eth0
# ifconfig eth0 0 up && ifconfig br0 10.94.94.12
# route add default gw 10.94.94.1
# ifconfig br1 172.10.1.1
```

Ping each other, we could see br1 is not OK.

#### VXLAN Setup

On VM1, do following operation, to set the vx1:

```
root@OpenVSwitchVM1:~# ovs-vsctl add-port br1 vx1 -- set interface vx1 typ
root@OpenVSwitchVM1:~# ovs-vsctl show
a1e9afb6-345a-4f79-8e0b-131cd43cfb67
    Bridge "br0"
        Port "eth0"
        Interface "eth0"
        Port "br0"
        Interface "br0"
            type: internal
    Bridge "br1"
        Port "br1"
        Interface "br1"
            type: internal
        Port "vx1"
        Interface "vx1"
            type: vxlan
            options: {remote_ip="10.94.94.12"}
    ovs_version: "2.3.0"
```

```
On VM2, do following operation, to set vx1

root@OpenVSwitchVM2:~# ovs-vsctl add-port br1 vx1 -- set interface vx1 typ
root@OpenVSwitchVM2:~# ovs-vsctl show
bce3f2b5-9b77-41dc-8130-b8922dd7ac9e
  Bridge "br1"
    Port "vx1"
      Interface "vx1"
        type: vxlan
        options: {remote_ip="10.94.94.11"}
    Port "br1"
      Interface "br1"
        type: internal
  Bridge "br0"
    Port "br0"
      Interface "br0"
        type: internal
    Port "eth0"
      Interface "eth0"
  ovs_version: "2.3.0"

So now you could ping each other via the br1 address.
```

## VIII.IV Les solutions de chiffrement du trafic L2TPv3 ou VXLAN

### VIII.IV.1 L2TP

L2TP sur UDP ou sur IP peuvent l'un et l'autre être sécurisés avec IPsec, la méthode recommandée, pour sécuriser L2TPv2 et le L2TPv3 possède des caractéristiques identiques à l'égard d'IPsec pour la sécurisation du tunnel.

### VIII.IV.2 VXLAN

VXLAN apporte une isolation entre les différents locataires mais le trafic circule en clair. La solution la plus simple pour le chiffrer est IPsec.

## VIII.V Comparaison avec MPLS ces deux technologies

### VIII.V.1 MPLS et L2TPv3

L2TPv3 fonctionne sur le réseau IP et MPLS PW nécessite un réseau MPLS.

Les deux sont similaires dans le type de services qu'ils fournissent, y compris l'interfonctionnement et la QoS.

L2TPv3 nécessite une taille MTU plus élevée tout au long du chemin sur le réseau IP, sinon nous pouvons configurer le routeur d'entrée pour fragmenter les paquets.

Pour MPLS PW, nous devons configurer MPLS MTU sur une valeur plus élevée tout au long du chemin.

MPLS PW nécessite un LDP ciblé entre les routeurs PE, l'encapsulation est effectuée par l'étiquette MPLS et pour L2TPv3, l'encapsulation est effectuée par les en-têtes L2TPv3 du paquet IP.

### VIII.V.2 MPLS et VXLAN

MPLS a été conçu pour accélérer le traitement des paquets. Le transfert est basé sur l'étiquette dans l'en-tête MPLS

VXLAN a été conçu pour permettre au trafic L2 d'être transféré au-dessus d'une structure IP. Le trafic est transféré en fonction de l'adresse de destination incluse dans l'en-tête IP externe. Cette adresse est l'adresse du point de terminaison de tunnel VXLAN de destination (VTEP). Un protocole de routage tel que BGP peut être utilisé pour annoncer les adresses VTEP.

## IX Conclusion

Ce projet nous a permis de mettre en place des tunnels grâce à des protocoles de tunneling ainsi que la manipulation des protocoles d'encapsulation sécurisés et des VLANs. Nous avons étudié en long et en large le protocole L2TPv3 et fait des comparaisons avec d'autres protocoles de tunneling. Par le biais des règles de firewall, nos netns ont pu communiquer sur internet et dans les VLANs. Nous avons mis en place tout ce qui a été demandé dans le projet.

## Bibliographie

- <http://abcdrfc.free.fr/rfc-vf/pdf/rfc3931.pdf>
- <https://networklessons.com/cisco/ccie-routing-switching-written/l2tpv3-layer-2-tunnel-protocol>
- [https://www.reddit.com/r/networking/comments/3sw4s6/vxlan\\_vs\\_l2tpv3\\_vs\\_something\\_else/](https://www.reddit.com/r/networking/comments/3sw4s6/vxlan_vs_l2tpv3_vs_something_else/)
- <https://networkengineering.stackexchange.com/questions/46151/vxlan-vs-vlan-over-layer-3>
- <https://support.huawei.com/enterprise/en/doc/ED0C1000174118/12541fbc/understanding-l2tpv3>
- <https://docs.openvswitch.org/en/latest/faq/vxlan/>