

Compte Rendu 1

1. Configuration réseau

1.1

Pour créer le réseau complet, nous avons lié les VM du tp1 et tp2 en ajoutant aux VM pouvant communiquer avec une nouvelle VM (VM1, VM1-6, VM3, VM3-6) une nouvelle interface à l'aide des vagrantfile. Cela permet à chacune des VM de pouvoir ping tout ces voisins.

1.2

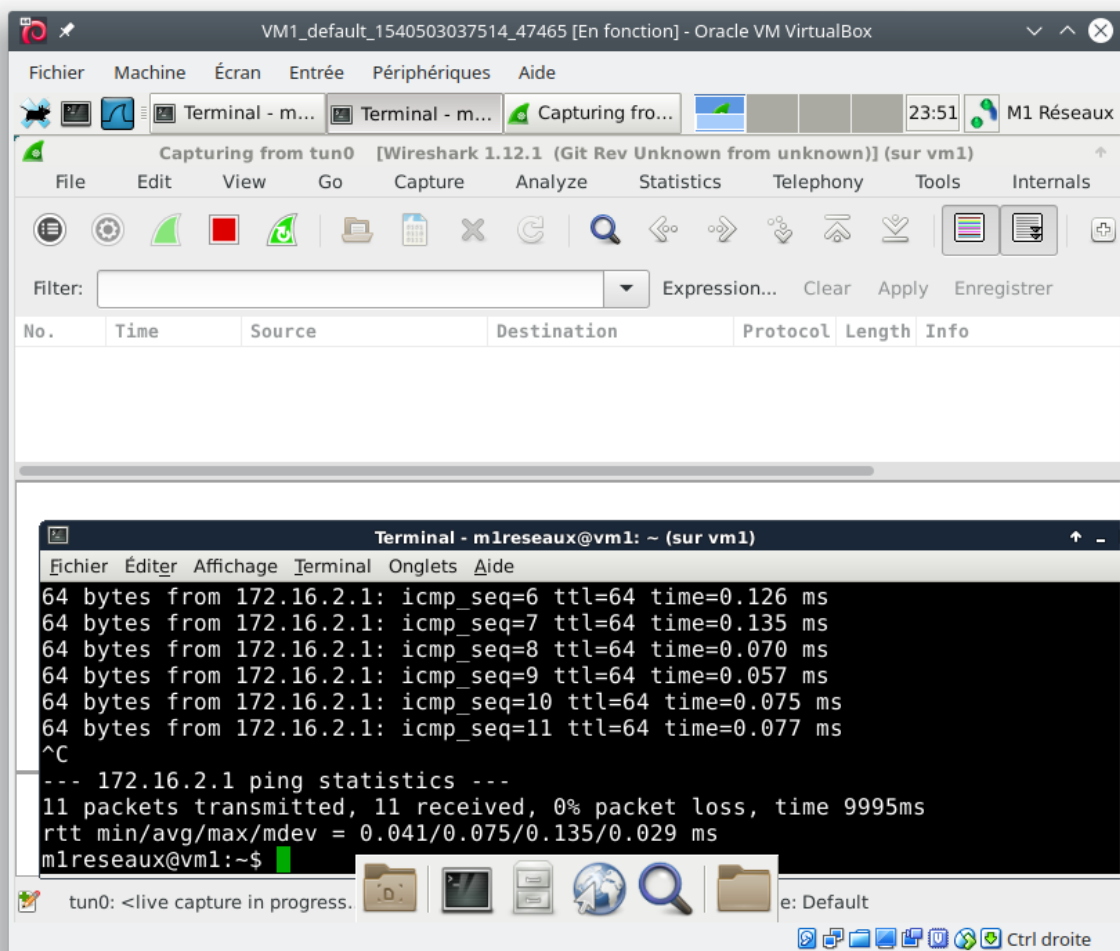
Pour que VM1 est de nouveau accès à VM3 il faudrait réussir à faire passer le message à travers le réseau de VM en ipv6.

2. L'interface virtuelle TUN

2.2

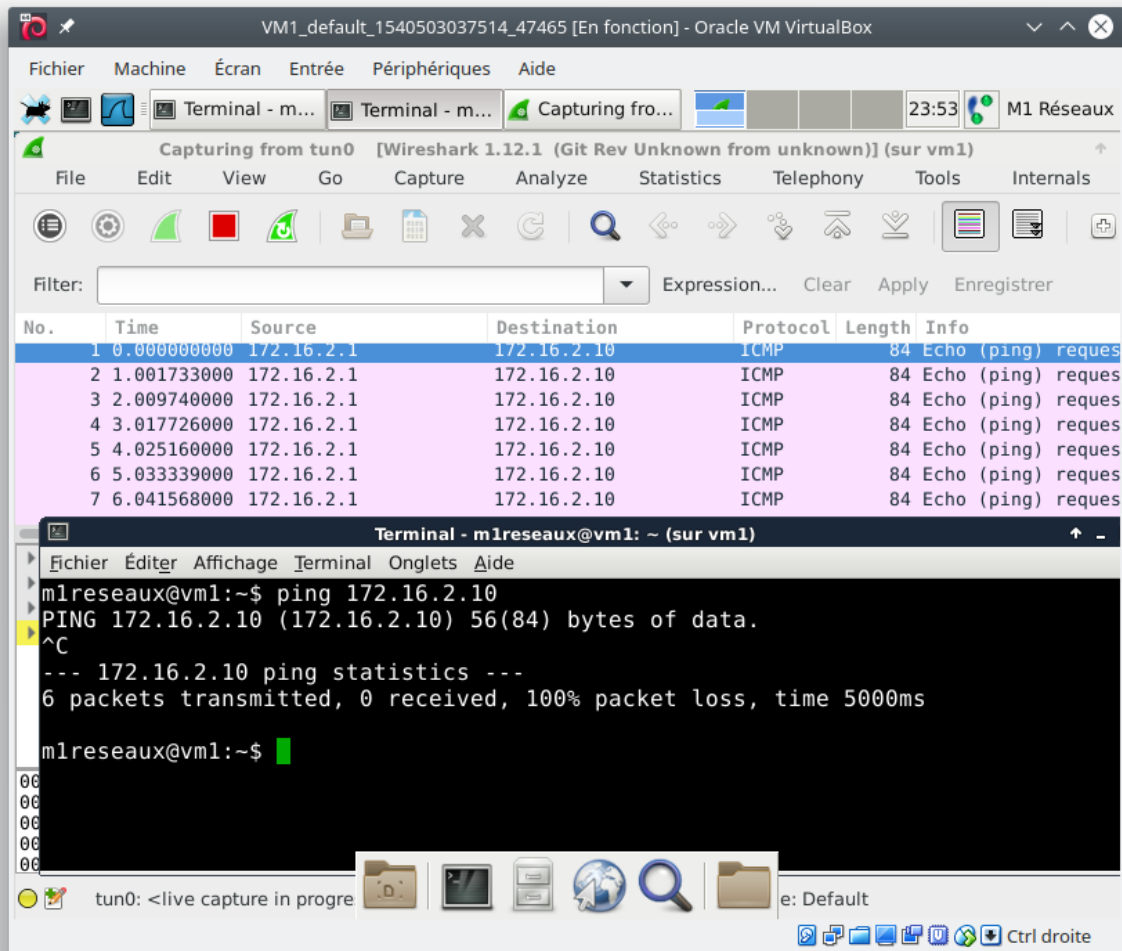
1) Pour configurer le tunnel nous avons utiliser la commande `ip addr add $1 dev tun0` elle permet à l'aide de \$1 représentant l'adresse de pouvoir la modifier plus facilement si l'on doit réutiliser le script plus tard.

2)



On peut voir que le ping ne traverse pas l'interface tun0 même si l'on obtient la réponse à la requête ping.

3)



On peut voir que le ping bien que n'ayant pas de réponse traverse l'interface tun0.

4) On peut voir que lorsque l'on ping 172.16.2.10 les ping sont perçus par wireshark même si dans le terminal ayant lancer la commande les ping sont sans réponses.

5)

Pour le ping de 172.16.2.1 on voit que le ping fonctionne même si il n'est pas affiché à partir de wireshark. On peut possiblement expliquer ca par le fait que la réponse se fait avant de rentrer dans le sous réseau de tun0.

Pour le ping de 172.16.2.10 il est normal qu'il soi perdu car i n'existe aucune machine dans le réseau ayant cette adresse. On peut les voir passer à travers l'interface tun0 car il s'agit du même sous réseau, cette interface est utilisé pour communiquée avec cette partie du réseau.

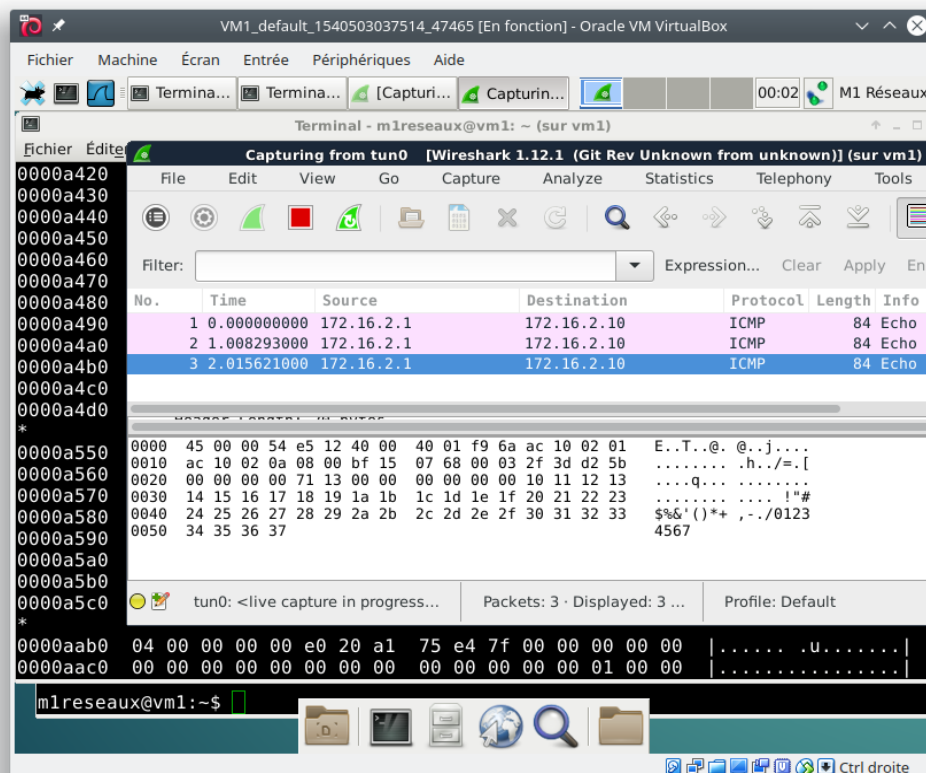
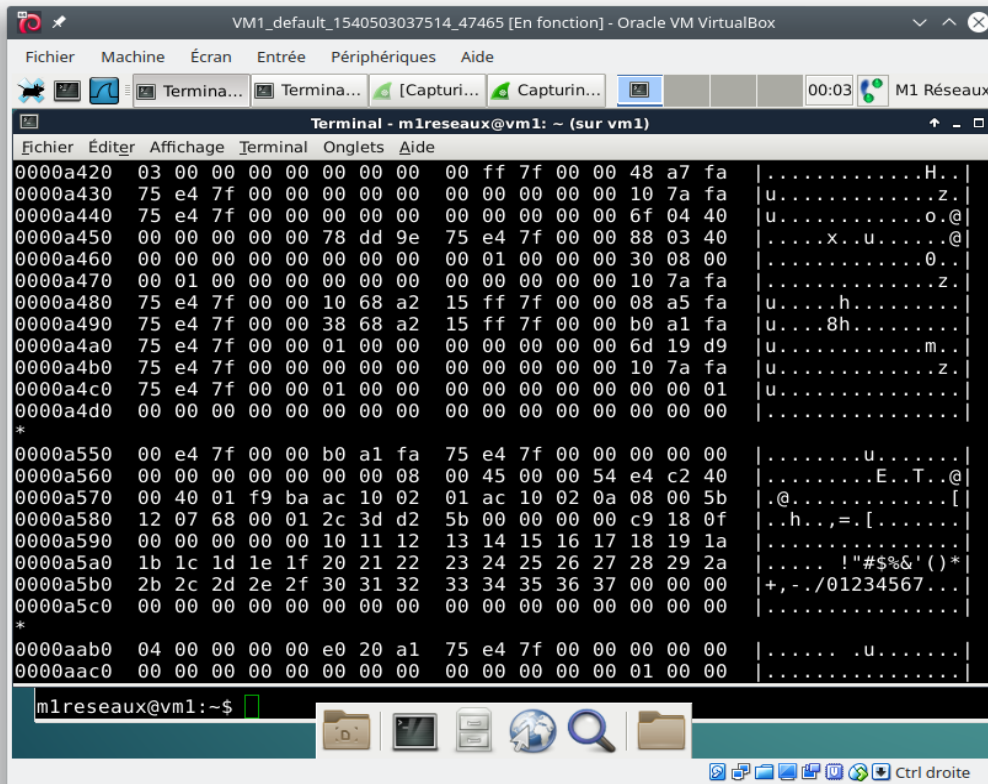
2.3

1)

2)

La réception de données à le même comportement qu'avec le ping c'est à dire qu'aucun affichage n'est déclenché lorsqu'il est la cible de la réception de ses données. L'affichage ci-dessus provient d'un ping à destination de 172.16.2.10.

3)



On peut remarquer que notre programme a la même sortie que ce que Wireshark montre. Il s'agit donc de la trame du ping vers 172,16,2,10.

5)

L'option IFF_NO_PI va servir à retirer les headers Ethernet qui ne sont pas nécessaire dans cette utilisation.