# Quelques éléments à lire sur Netkit avant de commencer le TEA

## Pour lancer netkit

Placez-vous dans le répertoire $NETKIT_HOME et vérifiez la configuration du système en exécutant la commande
./check_configuration.sh
Des erreurs ou warning apparaissent : corrigez selon les informations affichées : export…
Relancez une nouvelle fois ./check_configuration.sh jusqu'à disparition des erreurs.

## Construire un réseau à partir d'un fichier lab.conf

Il est possible de décrire la structure du réseau dans un fichier nommé lab.conf. C'est ce que vous ferez d'une manière générale dans vos montages. La description d'une machine dans le lab.conf se fait de la manière suivante :

Nom_machine[0]=A
Nom_machine[1]=B
Nom_machine[2]=C

Ces lignes créent une machine appelée nom_machine, qui comporte 3 interfaces, eth0, eth1 et eth2, qui sont respectivement connectées aux switchs A, B et C.

## Il est nécessaire de créer dans le répertoire contenant le fichier lab.conf des dossiers dont le nom correspond à celui des machines virtuelles.

Par exemple, si votre fichier lab.conf contient :
secret[0]=A
public[0]=B
routeur[0]=A
routeur[1]=B
Alors vous devez créer 3 répertoires nommés : secret, public et routeur. Attention : la casse et l'orthographe des noms des répertoires doivent être identiques aux noms des machines décrites dans le fichier lab.conf.

Une fois les répertoires et le fichier lab.conf créés, on démarre les machines en tapant simplement lstart dans le répertoire contenant le lab.conf.

Pour arrêter les machines, il est conseillé de lancer dans chacun des terminaux, la commande shutdown -h now ou halt. Puis un lhalt dans le terminal de l'hôte. Si vous rencontrez des problèmes lors de l'arrêt des machines, lancez killall netktit-kernel sur l'hôte.

Attention ! Hormis les dossiers des noms des machines, le fichier lab.conf et les fichiers *.startup (voir juste après), vous ne devez placer dans le répertoire de travail aucun autre fichier ou répertoire.
Il est possible d'exécuter des commandes au démarrage des machines. Pour cela, il suffit de créer un fichier nom_machine.startup dans le même répertoire que le fichier lab.conf et d'y inscrire les commandes à exécuter. Il est notamment intéressant d'y placer les commandes de configuration des paramètres IP et de la table de routage.
Par exemple, le contenu du fichier secret.startup serait :

```
ifconfig eth1 172.30.0.1 netmask 255.255.0.0 up
route add default gw 172.30.0.254
```

Il faut toujours terminer le fichier par un retour à la ligne pour que la dernière commande soit exécutée.
*Remarques :*
*- Par défaut, vous êtes connecté en root sur la machine virtuelle. Le mot de passe du compte root est root.*
*- Au redémarrage des machines, les modifications réalisées dans les fichiers ne sont pas perdues.*

## Ecoute des paquets réseaux

Chaque machine virtuelle possède un répertoire /hosthome qui est lié à votre répertoire de votre machine hôte. Ainsi, tous les fichiers placés dans ce dernier sur la machine hôte sont accessibles depuis les machines virtuelles par le répertoire /hosthome, et inversement. Ceci vous permet de transférer simplement des fichiers entre l'hôte et les machines virtuelles.
En particulier, vous pouvez enregistrer les captures tcpdump dans un fichier que vous placerez dans /hosthome et que vous ouvrirez dans wireshark lancé depuis votre machine hôte. Dans ce but, la ligne de commande à taper sur la machine virtuelle est donc :

tcpdump –w /hosthome/capture.cap.

Votre répertoire sur la machine hôte contiendra le fichier capture.cap qu'il suffit de l'ouvrir dans wireshark. Vous bénéficiez ainsi de l'interface graphique pour l'analyse de vos captures.

***Plus de détails***

*Pour analyser le trafic sur l'interface eth0 d'une machine, il suffit de taper: tcpdump –i eth0*
*Le trafic capturé est alors affiché directement dans le terminal.*
*Pour enregistrer le trafic d'une machine virtuelle dans un fichier lisible par wireshark sur la machine hôte, il suffit de taper la commande tcpdump -i eth0 -w /hosthome/capture.cap*
*Le trafic est enregistré dans le fichier capture.cap que vous trouverez dans votre répertoire personnel sur la machine hôte. Vous pouvez alors l'ouvrir avec :*
*tcpdump -r /hosthome/capture.cap ou avec wireshark sur l'hôte.*

*Pour pouvoir écrire des commandes dans le terminal tout en capturant le trafic, ajoutez un & à la fin de la commande tcpdump :*
*tcpdump -i eth0 -w /hosthome/capture.cap &*
*Avant d'ouvrir le fichier dans wireshark, il faut arrêter le processus tcpdump. Pour cela, exécutez la commande : killall tcpdump*
*Vous pouvez rediriger les erreurs via la redirection 2> /dev/null*

## Connexion vers l'extérieur

Pour accéder à l'extérieur, nous créons généralement un routeur disposant d'une interface eth2 (interface qui sera reliée à une interface de type TAP du système hôte). De plus, grâce à ce routeur, nous pourrons ainsi faire communiquer l'ensemble des machines virtuelles vers l'extérieur et pourrons ainsi installer de nouveaux paquets.

*Remarque : Interfaces **TAP***
*La création d'une machine virtuelle disposant d'une interface TAP nécessite que l'utilisateur dispose des droits root sur le système hôte.*

**Exemple (ne pas réaliser) :**
La commande suivante vstart routeur --eth1=tap,192.168.0.1,192.168.0.2 --eth0=A
permet de créer :
- sur l'hôte une interface nk_tap_root avec l'adresse IP 192.168.0.1,
- une interface eth1 avec l'IP 192.168.0.2 sur la machine virtuelle routeur qui pourra maintenant joindre l'extérieur,
- une interface eth0 connectée au switch A

Netkit a en fait effectué les opérations suivantes sur le système hôte:

-- ajout d'une règle de translation d'adresse;
- ajout d'une règle autorisant les flux émis par notre interface TAP;
- activation du routage:
- une règle de routage sur le routeur précisant que la passerelle par défaut est l'interface TAP du système hôte.

Il ne reste plus qu'à finir de configurer le DNS sur le routeur (fichier /etc/resolv.conf), son adresse interne connectée au switch A (ex : 172.30.0.250) et de s'assurer que le routage est activé (il l'est par défaut sur toutes les machines Netkit).

Cela ne suffit pas pour que les machines autres que le routeur puissent joindre l'extérieur. Le système hôte sera en effet incapable de router les paquets retour. Il n'a en effet aucune connaissance du plan d'adressage internet des machines Netkit connectées au switch A (dans le cas de l'exemple *172.30.0.0/16).* La solution est de réaliser une seconde translation d'adresse au niveau du routeur pour que les paquets qu'il émet son l'interface TAP (eth2) le soient avec l'adresse IP 192.168.0.2.

La commande est la suivante : iptables -t nat -A POSTROUTING -o eth2 -j MASQUERADE

**En résumé, pour qu'une machine puisse se connecter à l'extérieur, il faut créer une interface spéciale (ici eth2) sur une machine router, dont la syntaxe est la suivante :**
routeur[2]=tap, 192.168.0.1,192.168.0.2
**Ajoutez ensuite sur l'hôte la commande :**
iptables -t nat -A POSTROUTING -s 192.168.0.0/24 -j MASQUERADE
**Ajoutez sur le routeur (tout ce qui sort est masqué)**
iptables -t nat -A POSTROUTING -o eth2 -j MASQUERADE

# ATELIER TEA : Partie 1 – Des erreurs se sont glissées dans la configuration du réseau. A vous de les trouver !

Soit le réseau présenté sur la figure 1 vu en cours et TD. Le dossier de configuration (tpl3netE) de ce réseau doit être téléchargé du serveur Moodle. Attention : dans chaque segment, seule une machine est lancée, celle correspondant à l'adresse IP la plus petite dans le segment.

- Consultez les différents fichiers de configuration, lab.conf, et *.startup. Puis lancez le laboratoire avec la commande lstart.

- Complétez les tables de routage directement sur les terminaux des machines virtuelles de telle façon que les routeurs R2 et R3 puissent communiquer DIRECTEMENT avec R5 et R6. Indication: aidez-vous des commandes ping et traceroute pour diagnostiquer le problème; vous avez 2 routes à écrire sur deux routeurs. Choisissez-les bien !

- Vérifiez avec la commande traceroute que pc65 communique maintenant avec pc130 et pc161.

- Modifiez les fichiers *.startup pour prendre en compte ces nouvelles routes. Dans un terminal sur l'hôte, lancez la commande killall netkit-kernel, puis relancez le labo. Vérifiez que tout fonctionne.

- Tapez les deux commandes nécessaires pour que votre réseau puisse *voir* la salle (voir section « Connexion vers l'extérieur »). La première commande se lancera sur l'hôte, la seconde sur R1. Vérifiez que pc130 peut sortir sur le réseau de la salle. De pc130, connectez-vous à une machine de la salle en ssh. Ajoutez une route pour que pc3 puisse accéder lui aussi sur une machine de la salle.

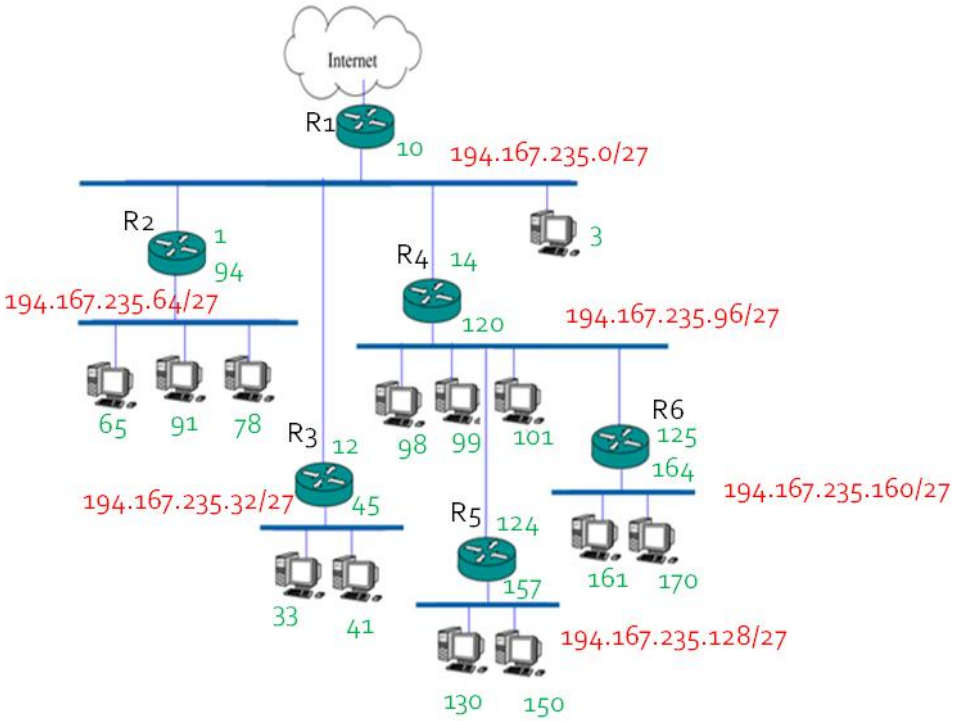
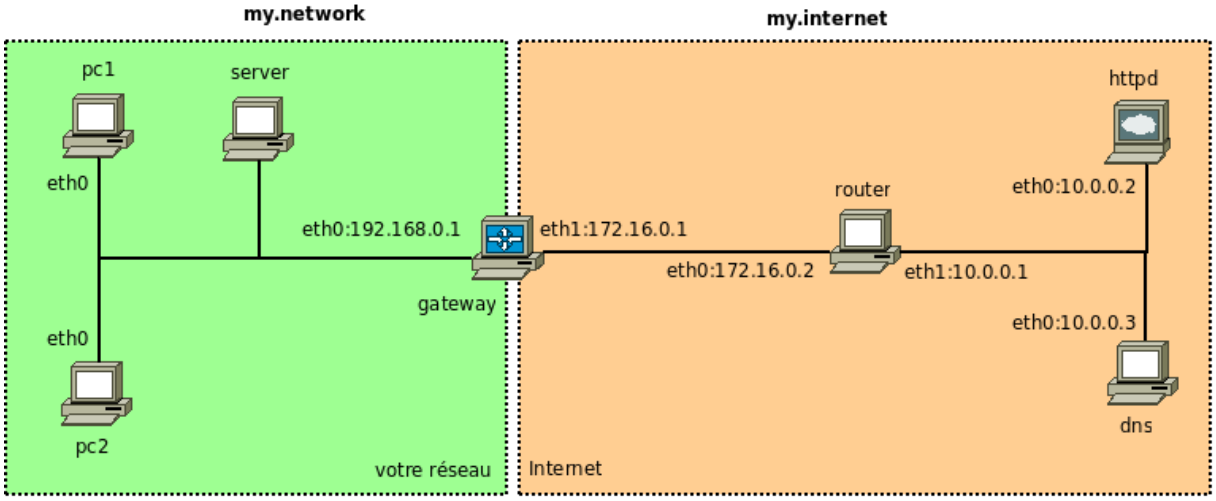



**Figure 1. réseau constitué de 6 routeurs et de 13 postes. Dans le fichier de configuration donné, en plus des 6 routeurs, seuls les postes pc65, pc3, pc98, pc161 pc33 et pc130 seront lancés**

# ATELIER TEA. Partie 2 - Découverte d'un réseau LAN et de ses services

Soit l'architecture ci-dessous.




Cette configuration possède deux ensembles :
– votre réseau (my.network) : les machines server, pc1 et pc2 représentent un réseau domestique derrière une passerelle (gateway). Chacune des machines est sous votre contrôle.
– Internet (WAN) (my.internet) : router, httpd et dns sont des machines extérieures à votre réseau (vous ne devez pas modifier leur configuration).

Les machines sont les suivantes :

- pc1 et pc2 représentent des postes clients dans votre réseau
- server est une machine fournissant le service DNS et DHCP dans votre réseau.
- gateway représente votre passerelle pour l'accès à Internet
- dns (my.internet) est un serveur DNS
- httpd est un serveur HTTP
- router représente un routeur sur Internet

La configuration de votre réseau local est sensiblement identique à celle d'un réseau domestique connecté par une free/orange/sfr … box à Internet.

➔ Avant de lancer le labo, modifiez les différents fichiers de configuration du laboratoire pour que la partie LAN virtuel (my.network) prenne l'adresse de sous réseau 192.168.1.0/24. Les fichiers de configurations à modifier se trouvent dans le dossier *server/etc/* (il y a en 3 !). Justifiez. Contrôlez pc1.startup et pc2.startup. Attention, server et gateway doit être aussi dans ce LAN ! En conséquence quels autres fichiers doivent être aussi modifiés ?
➔ Configurez lab.conf pour que router puisse se connecter à l'hôte via une interface de type tap (eth2) ayant pour adresse 192.168.3.1. L'hôte prendra alors l'adresse 192.168.3.2.
➔ Configurez la gateway du sous réseau pour que celle-ci puisse masquer les adresses du LAN (MARSQUERADE). La machine gateway doit être routeur.

- ➔ Même chose pour la machine router. La sortie eth2 doit être configurée en NAT et masquée les adresses des machines sortant par cette interface.
- ➔ A l'aide de Netkit, lancez le laboratoire (commande lstart dans le répertoire du lab) : host> lstart -s
- ➔ Lancez dhclient eth0 sur PC1 et PC2. Vérifiez que les 2 PC prennent les bonnes adresses.
- ➔ Vérifiez que PC1 et PC2 peuvent attendre les différentes machines du LAN, du WAN virtuels (my.internet), ainsi que du LAN physique.
- ➔ Vérifiez que le serveur apache2 est bien lancé sur le pc httpd (/etc/init.d/apache2 status). Avec la commande wget sur pc1, connectez-vous sur la page index.html du site.
- ➔ Configurez les routes sur l'hôte pour atteindre le sous réseau 172 .16.0.0 du WAN virtuel.
- ➔ Sur PC1, utilisez la commande nmap pour explorer/scanner le réseau LAN. Quels sont les ports ouverts sur la gateway ? Quels sont les ports ouverts sur httpd et dns ?
- ➔ On souhaite placer quelques règles anti-scans sur gateway. Justifiez leur utilité.

```
iptables -t filter -A INPUT -p tcp --tcp-flags ALL NONE -j DROP

iptables -t filter -A INPUT -p tcp --tcp-flags ALL ALL -j DROP

iptables -t filter -A INPUT -p tcp --tcp-flags ALL SYN,FIN -j DROP

iptables -t filter -A INPUT -p tcp --tcp-flags ALL URG,PSH,FIN -j DROP

iptables -t filter -A INPUT -p tcp --tcp-flags ALL FIN -j DROP

iptables -t filter -A INPUT -p tcp --tcp-flags SYN,RST SYN,RST -j DROP
```

# ATELIER TEA. Partie 3 - Etude d'une attaque construite sur l'ARP poisonning

Soit le réseau présenté sur la figure 2. Le dossier de configuration (arpoisoning) de ce réseau doit être téléchargé du serveur moodle. Editez le fichier lab.conf et modifiez-le pour que toutes les machines présentes dans le réseau local puissent se lancer. Lancez le laboratoire et vérifiez que les adresses des interfaces sont bien positionnées. Vérifiez également l'adresse de l'hôte (nk_tap_root).

Ajoutez sur l'hôte la commande iptables -t nat -A POSTROUTING -s 192.168.1.0/24 -j MASQUERADE.

Inscrivez dans un tableau les correspondances machine, adresse IP et adresse MAC et service.

Complétez la configuration de la machine dns pour qu'elle puisse adresser un serveur de google : 208.67.222.222. Vérifiez que vous pouvez maintenant, de victim, atteindre ce serveur.

Visualisez à l'aide de la commande ip n show le contenu du cache ARP de victim. Videz-le avec la commande ip n flush all. Connectez-vous ensuite à l'aide de la commande links à facebook.com. Visualisez à nouveau le cache ARP. Que constatez-vous ?

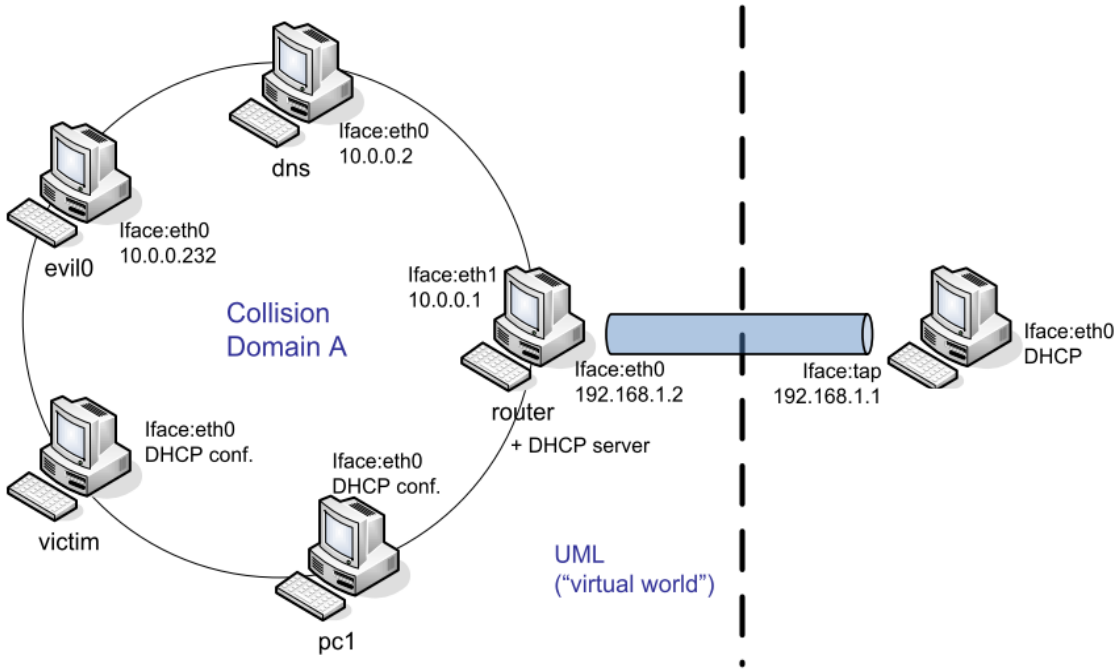



Figure 2. Réseau "Arppoisoning"

Nous utilisons l'utilitaire scapy (qui vient d'être installé sur evil0) pour effectuer l'ARP poisonning.

Dans un premier temps, visualisez :

- les fichiers /etc/resolv.conf de evil0 et de dns.

- le fichier de configuration evil0.startup ainsi que son fichier /etc/hosts

Que constatez-vous ? Quelle est d'après vous l'attaque mise en place ?

Sur evil0, lancez les commandes suivantes (attention, vérifiez bien l'adresse de victim et corrigez si nécessaire:

```
evil0:$ scapy

>>ips="10.0.0.2"
>>ipd="10.0.0.101"
>>hs="00:00:00:00:00:FF"
>>hd="00:00:00:00:00:AA"

>>a=Ether(src=hs,dst=hd)
>>b=ARP(op=2,psrc=ips,pdst=ipd,hwdst=hd,hwsrc=hs)
>>p=a/b
>>sendp(p,loop=1,inter=1)
```

Précisez l'objectif.

Visualisez à nouveau le cache ARP de victim. Que se passe t-il ?

Faites une requête à l'aide de links vers www.facebook.com

Expliquez l'attaque en vous appuyant sur le schéma ci-dessous.

Analysez les différents fichiers de configuration afin d'être encore plus précis dans votre analyse (table mangle, fichier hosts,…). Vous pouvez relancer l'attaque, le fichier attack.py scapy est directement accessible à la racine de evil0 (lancez python attack.py).