TP Réseaux : Principes de l'adressage, du routage et du filtrage IP

(TP sur deux séances, dépôt moodle final et QCM en fin de la 2de séance – semaine 12)

Objectifs

- Observer et comprendre les principes d'adressage (configuration), de routage (aiguillage dans le réseau) et, s'il reste du temps, de filtrage (fonctions de pare-feu) IP
- Configurer le réseau sous Linux Mint et Windows
- Interconnecter des LAN

Données, outils et documentation

- salle réseau (salle D303 ou D304) avec tout ce qu'il y a à l'intérieur : câbles, hubs, switchs... et notamment le précâblage local privé (prises réseaux nommées Lxy, L comme Local)
- les câbles dans ces 2 salles ont été remplacés et sont de la catégorie 6 partout. Les codes couleurs **qu'il faut respecter et remettre en état à la fin des TP** est le suivant :
  - câbles **noirs** : servent, par défaut, à relier les machines étudiantes au réseau étudiant isolé 10.192.0.0/16, il faut veiller à ce que ce soit le cas à la fin du TP avant de quitter la salle. Cependant, pendant les TP ils peuvent provisoirement servir à d'autres interconnexions si besoin.
  - câbles **jaunes** : servent pour les manipulations et interconnexions provisoires le temps des TP, il faut les ranger à la fin des TP en les suspendant aux rails prévus à cet effet.
  - câbles **blancs/gris clairs** : câbles d'interconnexion dans les baies de brassage à destination des étudiants pour les manipulations de TP, ils ne servent que là.
  - câbles **bleus** : câbles d'interconnexion dans les baies de brassage à destination des enseignants et permettant de mettre en place les liaisons nécessaires au TP.
- machines physiques démarrées sous Linux
  - les machines physiques sont nommées arm001 à arm014 en D304 et sparc01 à sparc14 en D303
- VM sous Linux sur les machines avec 2 cartes réseau pour les passerelles/routeurs (machines normalement avec écran iiyama) et VM sous Windows sur les machines clientes avec 1 seule carte réseau pour les clients (machines normalement avec écran Dell)
  - les cartes principales sont toujours au milieu du panneau arrière
  - les VM portent les noms vmarm0xx en D304 et vmsparcxx en D303 (xx allant de 01 à 14)
- machine VM sous Linux pour les passerelles/routeurs et VM sous Windows hébergée sur une machine physique Linux (machines HP écran) pour les clients
  - les passerelles Linux utiliseront leurs 2 cartes réseaux et les clients Windows n'en utiliseront qu'une seule.
- identifiants de connexion :
  - Linux Mint : compte `admin` avec mot de passe `local192` pour se connecter et compte `root` avec mot de passe `local192` pour les configurations ayant besoin des privilèges administrateur (pour passer en mode `root` utiliser « `su -` »)
  - Windows : compte `Administrateur` ou `local192` avec mot de passe `local192`
- des petits commutateurs 8 ports (switchs Netgear), des hubs et switchs supplémentaires disponibles dans les salles ou dans leurs baies de brassage pour réaliser vos réseaux locaux
- outil Wireshark (voir TP Wireshark pour son utilisation)
- les pages du manuel Linux des différentes commandes à utiliser :
  - ifconfig : `man ifconfig`
  - route : `man route`
  - ping : `man ping`
  - traceroute : `man traceroute`
- **À la fin de la séance (1pt de pénalité pour les binômes qui ne feront pas la remise en état)**
  - **remettre le câblage initial**
  - **effacer vos fichiers scripts ou tout autre fichier créé (après sauvegarde sur un support personnel ou dépôt sur moodle)**

TP

Le TP est à effectuer en binôme.
Les passerelles/routeurs seront des VM sous Linux Mint et les clients seront des VM sous Windows. Dans ce qui suit, toutes la manipulations se feront sur l'une de ces 2 VM, disponibles sur les machines physiques des salles D303/D304. Si soucis avec les VM prévenez votre chargé de TP et passez sur la machine physique.

Dépôts moodle

Pour ce TP, vous aurez à remettre le relevé de manips obtenu à la fin de la 2de séance et à remettre le jour de celle-ci.

- le relevé de manips rédigé en markdown, de préférence sous vim, et reprenant toutes les **commandes de configuration** et de **test** exécutées **sur votre passerelle** depuis le paragraphe « Travail préliminaire » (ci-dessous) et jusqu'à la fin du sujet, ainsi que **les sorties terminal correspondantes**. Vous le compléterez au fur et à mesure des 2 séances du TP IP avec toutes les commandes de configuration que vous ferez.
  - chaque commande doit être précédée par un commentaire précisant ce que va faire la commande ou répondant à la question posée et suivie par sa sortie terminal.
  - les commandes ping doivent absolument utiliser l'option -c pour faire un nombre limité de tests
- ce document, sera obligatoirement nommé `TPIP-nomEtudiant1-nomEtudiant2-numeroLAN.md`, il vous servira également à reconfigurer votre machine lors de la 2de séance.
- le relevé de manip doit être converti en pdf lors du dépôt et renommé `TPIP-nomEtudiant1-nomEtudiant2-numeroLAN.pdf`, pour la conversion en pdf d'un fichier markdown voir TP wireshark.
- date limite du dépôt fin de journée de votre 2de séance du TP (pénalité 1pt par heure de retard)

Travail préliminaire

À partir du moment où vous commencerez à configurer votre réseau, vous n'aurez plus d'accès Internet jusqu'à la question 3 et pas d'accès au serveur de la salle D304 (serverrx) jusqu'à la question 2. Commencez par charger le sujet dans un navigateur de la machine physique et dans un navigateur de votre VM (VM Linux pour la passerelle, VW Windows pour le client).

Sur la VM passerelle effectuez les commandes suivantes (le # et ce qui le suit est un commentaire qui ne fait pas partie de la commande) :

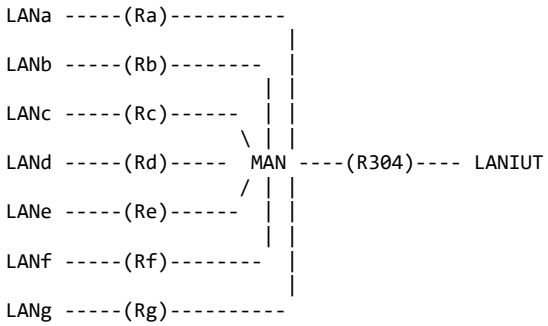
- dans le menu Périphériques->Réseau de la fenêtre de la VM Linux (qui vous sert de passerelle), associer la 2e carte réseau de la VM à la carte physique ens1 (au lieu de eno1)
- `service network-manager stop` # arrêt du NetworkManager, prcessus qui gère la configuration automatique de la machine
- `dhclient enp0s3` # récupération de la configuration IP via DHCP, si la carte réseau principale a un autre nom, utilisez le bon nom
- pour tester, faites un ping avec option -c3 sur le serveur 10.192.50.253 et vérifiez avec ifconfig que `enp0s3` a une adresse valide sur le réseau 10.192.0.0/16. Normalement, `enp0s3` est la carte intégrée active par défaut au milieu du panneau arrière (l'autre carte est juste à coté et s'appelle `enp0s8` sur les machines avec 2 cartes réseaux).
- sur les machines clientes sous Windows, désactivez le pare-feu (pour accéder à la fenêtre de configuration exécuter `firewall.cpl` dans une invite de commandes Windows)

Séance 1 : Création et configuration de réseaux

Chaque groupe de TP doit mettre en œuvre l'architecture réseau ci-dessous avec un LAN par binôme :

- LAN16 à LAN31 avec respectivement les @réseau 172.16.0.0/16 à 172.31.0.0/16
- les LAN16 à LAN23 (ne pas utiliser le LAN 17) seront mis en œuvre en salle D304 et les LAN24 à LAN31 dans la salle D303
- les numéros des LAN seront attribués par votre chargé de TP

L'objectif est de configurer la salle comme suit :



- LANIUT est le réseau étudiant isolé de l'IUT pour les TP réseaux d'@réseau 10.192.0.0/16, qui est le LAN par défaut des salles réseaux
- MAN, d'adresse réseau 192.168.222.0/24, est le réseau fédérateur de vos LAN individuels. Il est connecté au LANIUT par la passerelle (R304), le serveur Linux (serverrx) de la salle D304 avec 2 cartes réseaux : adresse IP 10.192.50.253 sur le réseau LANIUT et 192.168.222.253 sur le MAN. Le serveur de la salle D304 est déjà configuré et connecté sur les bons réseaux
- LANi (i allant de 16 à 31, voir ci-dessus) sont les réseaux locaux de chaque binôme connectés au réseau fédérateur MAN par leur routeurs/passerelles respectifs (Ri), les VM Linux. Les adresses réseaux des LANi sont 172.i.0.0/16 (où i est le numéro du LAN attribué par votre chargé de TP).
- Chaque LAN comprend une machine cliente (VM Windows) qui sera le client du LAN avec l'adresse 172.i.i.i et la machine VM Linux qui sera la passerelle du LAN avec l'adresse 172.i.0.1 sur le LANi et 192.168.222.i sur le MAN.

- La machine cliente et la passerelle d'un LANi donné doivent être interconnectées soit directement en utilisant un petit switch Netgear (il est possible d'utiliser le précâblage et la baie de brassage de chaque salle pour attendre un équipement présent sur une rangée éloignée)
- Les passerelles des LANi et la passerelle (R304) sont interconnectées en D304 à l'aide du switch ProCurve présent dans la baie de brassage et en D303 par le hub CentreCOM 3016TR.

Question 1 : configuration IP et routage local

1. Réalisez le câblage adéquat au moyen des câbles et des équipements mis à votre disposition. Pour les passerelles, la carte intégrée (`enp0s3` sous VM Linux, carte du milieu) connectera la machine à son LAN et la carte supplémentaire (`enp0s8` sous VM Linux) sera reliée au MAN (192.168.222.0/24).

2. Affectez manuellement (commande `ifconfig`) une adresse IP conforme au plan d'adressage ci-dessus à chaque interface de votre passerelle/routeur.

3. Configurez la machine cliente sous Windows via l'interface graphique avec l'adresse IP adéquate (pour l'accès rapide à la fenêtre de configuration exécuter `ncpa.cpl` dans une invite de commandes Windows).

4. Observez la table de routage de votre passerelle/routeur (commande `route`, pensez à la commande `man` pour l'aide Linux et notamment pour trouver l'option qui permette de ne pas faire la résolution DNS des @IP). Quelles sont les routes que votre table contient (ajoutez en commentaire dans votre fichier script le contenu en mode texte de la sortie terminal) ? Regardez quelles sont les informations principales contenues pour chaque route.

5. Vous devez observer 2 routes locales sur votre passerelle/routeur : une vers votre LAN et une vers le MAN. Si ce n'est pas le cas, si les routes locales n'ont pas été ajoutées automatiquement, ajoutez celles qui manquent à l'aide de la commande `route`. Le format général pour ajouter une route locale est (voir `man route` pour plus de précisions) :

   - `route add -net <@LANlocal> netmask <masqueLANdestinataire> <interface>`
     # < et > ne font pas partie de la commande et indiquent les paramètres à personnaliser
     # à noter que pour une route locale, `interface` doit être la carte qui est connectée directement sur le `LANlocal` concerné

6. Testez la connectivité avec la commande `ping` et commentez les résultats obtenus entre :

   - votre machine cliente et votre passerelle/routeur
   - votre passerelle/routeur et la passerelle/routeur R304 (celle du LANIUT)
   - votre machine cliente et la passerelle/routeur R304 (celle du LANIUT)

   Dites notamment ce qui marche ou pas et la cause de cette réussite ou de cet échec.

Question 2 : routage distant et tests d'interconnectivité

1. Activez le routage sur votre passerelle/routeur. Il faut mettre un 1 dans le fichier spécial : `/proc/sys/net/ipv4/ip_forward`
   - soit par `/sbin/sysctl -w net.ipv4.ip_forward=1`
   - soit par la commande `echo 1 > /proc/sys/net/ipv4/ip_forward`

2. Configurez la route par défaut sur votre passerelle/routeur pour qu'elle passe par la passerelle/routeur R304. Il faut en effet que votre passerelle/routeur sache où envoyer les paquets destinés à un réseau qu'elle ne connaît pas. La passerelle/routeur du R304 est configurée pour vous permettre d'accéder au reste d'Internet. Pour effectuer la configuration, il faut utiliser la commande `route`.

3. Faites un ping depuis votre passerelle/routeur vers l'imprimante de la salle D304 (10.192.50.49). Est-ce que ça marche ?

4. Pour qu'une machine cliente d'un LAN puisse communiquer avec d'autres réseaux, il faut ajouter dans sa configuration IP la passerelle/routeur par défaut. Cette passerelle/routeur est la machine qui connecte le LAN en question au reste du monde. Configurez la passerelle par défaut de votre machine cliente (une passerelle par défaut doit obligatoirement avoir une @IP sur le LAN de la machine configurée et c'est cette adresse qu'il faut utiliser). Testez en faisant un ping depuis votre client vers l'imprimante de la salle D304 (10.192.50.49).

5. Faites un ping vers le serveur wushu.univ-lr.fr en utilisant son @IP : 10.2.50.62. Est-ce que ça marche ?
   Donnez 2 raisons possibles expliquant pourquoi ça ne marche pas.

6. Pour pouvoir atteindre une machine sur un LAN distant, il faut que tous les routeurs sur le chemin entre l'émetteur et le destinataire sachent vers où il faut envoyer les paquets concernés. Ajoutez sur votre passerelle, les routes explicites vers les autres LAN de votre salle (autres que le vôtre) et faites en sorte de pouvoir faire un ping sur toutes les machines clientes de tous les LAN des autres binômes de votre salle. Effectuez les tests adéquats et décrivez-les.

7. Utilisez la commande `traceroute` (voir `man`) sous Linux et `tracert` (`/?` pour l'aide) sous Windows pour vérifier par où passent vos paquets et où ça bloque en cas de problème.Vérifiez notemment le chemin entre votre client et le client d'un autre LAN puis entre votre client et l'imprimante de la salle D304.

8. Regardez le contenu de la table de routage de votre passerelle/routeur. Observez les 3 types de routes présentes. Ajoutez le contenu final de votre table de routage en commentaire dans votre script.

Question 3 : accès Internet

Faites en sorte d'avoir accès à Internet d'abord sur le client, puis sur la passerelle :

- Le serveur DNS du réseau étudiant est 10.2.40.230.
  - sous Windows le DNS se configure dans les paramètres de la carte réseau
  - sous Linux la configuration DNS se fait dans le fichier `/etc/resolv.conf`. Il faut y rajouter `nameserver @IP-serveurDNS`. Il faut aussi arrêter le service `avahi-daemon` (`service avahi-daemon stop`).
- Le proxy de l'université est `wwwcache.univ-lr.fr` et il écoute sur le port 3128. Le proxy se configure dans le navigateur utilisé dans la section des paramètres réseaux avancés.

Séance 2 : Analyse de l'interconnexion

En début de la séance, il faut remettre en place l'architecture commencée en séance 1.
Avant de poursuivre le sujet, il faut absolument terminer les questions 2 et 3 de la séance 1.
Pour ce qui suit, afin d'intégrer des tableaux en markdown dans votre relevé de manips, vous pouvez par exemple utiliser un générateur de tableau Markdown en ligne.

Question 4 : analyse de fonctionnement du routage

1. Remplissez le tableau qui suit. Notez les @MAC et @IP du client et des deux interfaces de la passerelle/routeur de votre LAN et du LAN immédiatement suivant le votre. Si vous gérez le LANi alors il faut noter ces adresses pour le LANi et le LANi+1. Si vous avez le dernier LAN de votre salle, le LAN suivant sera le 1er LAN de la salle. En cas de souci (si le LAN en question n'existe pas par exemple), votre chargé de TP vous en attribuera un.

	LANi	**LANi+1**
@MAC client		
@MAC routeur enp0s3		
@MAC routeur enp0s8		

2. Lancez 2 fois `wireshark` sur votre passerelle/routeur et faites une capture sur chacune de ses interfaces Ethernet. Faites un ping entre votre machine cliente et la machine cliente identifiée à la question précédente. Arrêtez la capture.
   - Après avoir filtré les trames qui vous concernent sous `wireshark`, remplissez le tableau ci-dessous avec ce qui est observé dans la capture.

	interface de capture	**@IP interface**	**@MAC émetteur trame**	**@MAC destinataire trame**	**@IP émetteur paquet**	**@IP destinataire paquet**
Requête ping	enp0s3					
	enp0s8					
Réponse ping	enp0s3					
	enp0s8					

Le tableau suppose que enp0s3 est raccordée à votre réseau local. Si ce n'est pas le cas, corrigez.

3. Analysez ce que vous avez noté et constatez comment fonctionne la retransmission ou le « forwarding » (traversée d'un routeur ou d'une passerelle). **Cette analyse vous servira pour certaines questions du QCM en fin de séance 2**.

Compléments s'il reste du temps : Initiation au filtrage IP et au NAT/PAT

Dans ce qui suit, la passerelle va avoir un rôle de pare-feu.

Rappels de fonctionnement de NetFilter (n'hésitez pas à demander des précisions à votre chargé de TP)

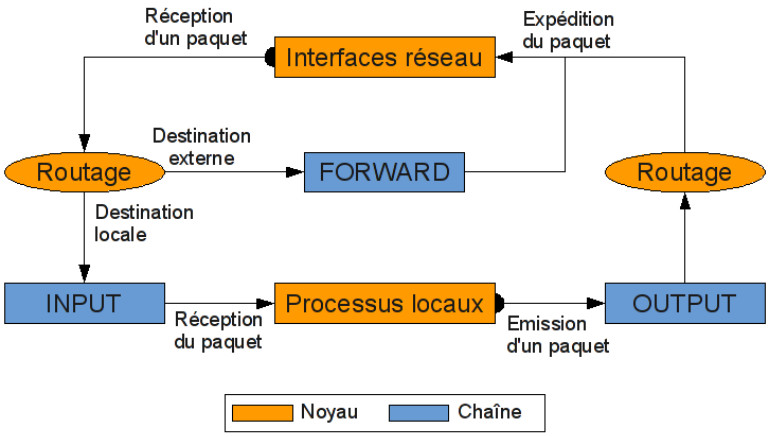
Les principes décrits s'appliquent à une machine qu'elle soit cliente ou passerelle à partir du moment où NetFilter est installé.

Un *datagramme* est un terme générique pour désigner les PDU des protocoles TCP/IP, couches transport et réseau. Tous les datagrammes qui transitent par un pare-feu NetFilter traversent un ensemble de tables (ou *files d'attentes*) qui sont manipulables sous Linux par la commande `iptables`. Chaque file est constituée par un ensemble de *chaînes* qui permettent de filtrer ou de transformer des datagrammes particuliers. Une chaine étant un ensemble de *règles* qui s'appliquent successivement à un datagramme reçu. Une *règle* définit des critères de sélection des datagrammes et une cible de destination du datagramme répondant à ces critères. Enfin, une *cible* est la décision appliquée au datagramme définissant le traitement que subira le datagramme répondant aux critères correspondant.

On trouve essentiellement trois types de tables (à noter qu'une 4e table existe, la table **raw**, non abordée ici) :

- la table **mangle** qui se charge de marquer ou de changer la qualité de service des datagrammes
- la table **filter** qui se charge du filtrage des datagrammes
- la table **nat** qui se charge de la translation d'adresse (mascarade ou *masquerading* en anglais) et de port

Celle qui nous intéresse ici est la table **filter** mais les principes sont similaires aux 2 autres tables avec une organisation des flux de traitement et des chaines propres à chaque table. **filter** est la table par défaut et elle est constituée de trois chaînes de filtrage :



- la chaîne **INPUT** sert à filtrer les datagrammes **destinés nommément à la machine** en provenance de l'extérieur (sous ensemble de tous les datagrammes qui arrivent à la machine)
- la chaîne **FORWARD** sert pour le filtrage des datagrammes qui transitent par la machine et ne concerne que les datagrammes qui ne sont pas nommément destinés à la machine ; cette chaine n'a de sens que si la machine est une passerelle (c'est à dire avec la fonction de routage activée)
- la chaîne **OUTPUT** sert pour le filtrage des datagrammes à destination de l'extérieur en provenance **explicitement** de la machine (à ne pas confondre avec tous les datagrammes qui sortent de la passerelle dont une partie représentent les datagrammes qui ne font que transiter).

Pour réaliser le filtrage des *règles iptables* sont associées à chaque chaîne. Ces règles s'appliquent les unes après les autres et filtrent un type particulier de datagramme en fonction de :

- l'interface réseau d'où provient le paquet (eth0, eth1, eth2, enp0s1, enp0s2...)
- l'adresse IP source
- l'adresse IP destination
- le protocole intermédiaire utilisé au dessus de IP : UDP, TCP, ICMP
- le port source
- le port destination
- les drapeaux TCP (*flags*): SYN, ACK, ...
- ...

Chaque règle est un ensemble de critères ou conditions qui, si elles sont vérifiées, vont aboutir à une décision pour le datagramme traité. Ces décisions sont appelées des *cibles*. Les principales cibles sont :

- **ACCEPT** qui accepte le datagramme
- **DROP** qui rejette le datagramme en mode silencieux en éliminant le datagramme
- **REJECT** qui rejette le datagramme en prévenant la source du datagramme de ce rejet par un paquet ICMP
- **LOG** qui enregistre dans les journaux systèmes le passage du datagramme dans la chaine correspondante
- ...

La disponibilité ou non d'une cible dépend de la table NetFilter manipulée. Certaines cibles n'apparaissent que dans certaines tables. En outre, les cibles peuvent être :

- terminales : celles qui provoquent l'arrêt de l'analyse du datagramme lorsque celui-ci est sélectionné par les critères de la règle en cours de traitement, la cible est appliquée et le traitement s'interrompt et passe au datagramme suivant (c'est le cas de ACCEPT, DROP et REJECT)
- non-terminales : celles qui n'interrompent pas l'analyse du datagramme, la cible est appliqué au datagramme qui correspond aux critères de sélection et le datagramme passe à la règle suivante (c'est le cas de LOG et de certaines autres cibles)

Les règles d'une chaine sont examinées séquentiellement et dès que les conditions de sélection (critères) correspondent au datagramme traité, la cible de la règle est appliquée. Si la cible est terminale, le processus de filtrage pour le datagramme s'arrête (les règles suivantes ne seront pas examinées), si la cible n'est pas terminale, le datagramme continue à être analysé par les règles suivante de la chaine considérée.
Remarque : une chaîne peut faire appel à une autre chaîne.

Les chaines et règles de filtrage du pare-feu NetFilter sous Linux se manipulent avec la commande `iptables`. La commande `man iptables` donne toutes les options utilisables. Voici la syntaxe générique simplifiée de la commande `iptables`. Ce qui est entre crochets est optionnel. Ce qui est entre chevrons doit être remplacé par une valeur valide. **Les crochets et les chevrons ne font pas partie de la commande !!!!**

```
user@PC-1 ~ $ iptables [-t <nom-table>] <commande> <nom-chaîne> [-i <if-in>] [-o <if-out>] [-s <@IP-src>] [-d <@IP-dst>]
[-p <protocole>] [<parametre> <option>]* -j <cible> [<parametre> <option>]
```

- `<nom-table>` est l'une des trois tables citées plus haut (par défaut c'est filter)
- `<commande>` peut-être -A, -D, -I, -P... voir `iptables -h` pour une aide minimale ou une des docs ci-dessus
- `<nom-chaîne>` est la chaîne des règles concernée de la table considérée : pour filter c'est INPUT, OUTPUT ou FORWARD, pour nat c'est PREROUTING, POSTROUTING ou OUTPUT...
- `<if-in>`/`<if-out>` est le nom de l'interface par laquelle le datagramme arrive à/sort de la machine
- `<@IP-src>`/`<@IP-dst>` est l'adresse IP de la source/du destinataire (peut être une adresse réseau, peut être en notation slashée précisant la taille du masque)
- `<protocole>` précise le protocole au dessus de IP (tcp, udp ou icmp) contenu dans le datagramme, chacun des protocoles définit un contexte différent pouvant donner accès à des paramètres de sélection supplémentaires
- `<parametre>` paramètre supplémentaire souvent dépendant du protocole sélectionné. Par exemple avec `-p tcp` on accède aux paramètres `--dport` et `--sport` permettant de spécifier le numéro de port TCP destination ou source
- `<option>` valeur du paramètre supplémentaire
- `*` précise qu'on peut avoir plusieurs occurrences des paires `<parametre> <option>` pour ajouter plusieurs critères supplémentaires
- `<cible>` désigne ce qu'on fait avec le datagramme correspondant aux critères précédents (ACCEPT, DROP, REJECT, LOG...).
- la paire `<parametre> <option>` qui suit la cible apporte des précisions sur le comportement de la cible. Par exemple, `--reject-with` permet de choisir le message ICMP qui sera envoyé à l'émetteur en cas de rejet (pour les types possibles consulter le paragraphe sur le cible REJECT de la documentation)

Le pare-feu NetFilter peut également faire le suivi des connexions transport avec le module *conntrack* (fonctionnement en mode *statefull* ou suivi d'états TCP). Cependant, cette notion n'est pas utile au cas d'étude traité et ne sera par conséquent pas décrit plus en détail.

Question 5 : Filtrage IP

Réalisez les expériences qui suivent :

a. Trouvez la commande qui affiche le contenu de vos tables de filtrage et vérifiez que la table est vide.
b. Tapez la commande : `iptables -A INPUT -i enp0s3 -p ICMP -j REJECT`
   - D'après vous, à quoi sert-t-elle (analysez la commande) ?
   - Faites des tests adéquats pour vérifier qu'elle est bien prise en compte. Il y a au moins 3 tests à effectuer :
     - test sur `enp0s3` de la passerelle
     - test sur `enp0s8` de la passerelle
     - test qui traverse la passerelle
   - Observez notamment avec Wireshark quels paquets ICMP sont bloqués (rappelez-vous que le ping c'est un aller-retour, là il y a donc 2 tests à faire).
     - test sur la passerelle et
     - test depuis la passerelle
   - Supprimez la règle ci-dessus (trouvez la commande qui l'enlève de la table de filtrage). Vérifiez que le ce qui ne passait pas ci-dessus passe à nouveau.
c. Tapez la commande : `iptables -A FORWARD -i enp0s3 -o enp0s8 -p ICMP -j REJECT`
   - D'après vous, à quoi sert-t-elle (analysez la commande) ?
   - Que se passe-t-il à présent si vous recommencez les tests de la question précédente ? Pouvez-vous encore faire des requêtes ping au sein de votre mini-réseau ? Est-ce normal ? Observez notamment avec Wireshark quels paquets ICMP sont bloqués.
   - Supprimez la règle ci-dessus (donnez la commande qui l'enlève de la table de filtrage).
d. Testez séparément et donnez la différence entre : `iptables -A FORWARD -p ICMP -j DROP` et `iptables -A FORWARD -p ICMP -j REJECT`.
   - Pensez à utilisez `wireshark` lors des tests pour constater ces différences.
   - Si les deux règles sont présentes simultanément, vérifiez laquelle est appliquée en premier (pensez à regarder dans les rappels de cours et de tester les diverses combinaisons).
   - Ne supprimez pas les règles !
e. La passerelle/routeur du MAN (R304) est aussi un serveur Web (port 80), un serveur telnet (port 23) et un serveur ssh (port 22). Testez que vous avez accès par défaut à ces 3 services depuis votre passerelle et depuis votre client.
f. Tapez la commande : `iptables -A FORWARD -p TCP --dport 23 -j REJECT`
   D'après vous, à quoi sert-t-elle (analysez la commande) et regardez quel service est bloqué ? Faites les tests adéquats et décrivez-les.
g. Trouvez la commande qui permet d'empêcher la retransmission par votre routeur du trafic HTTP et qui insère ce filtre en tête de votre liste de règles.
h. Trouvez la (ou les) commande(s) permettant d'empêcher toute session `ssh` de ou vers votre passerelle/routeur.
i. Donnez le contenu actuel de votre table filter
j. Trouvez comment remettre à zéro les tables de filtrage de votre routeur. Vérifier que votre mini-réseau est à nouveau perméable.

k. Assurez-vous que votre table de filtrage est vide. Appliquez une commande qui empêche toutes les retransmissions (rejet sans notification de tous les flux FORWARD) par votre routeur et qui définit ce fonctionnement comme étant le comportement par défaut. Vérifiez que plus rien ne traverse votre mini-réseau.