

TD : Routage statique

1. Principes de fonctionnement du routage statique

- Le routage statique consiste à définir manuellement les routes que doivent emprunter les paquets afin d'arriver à destination. Ce sont les administrateurs réseaux qui se chargent de ce travail en général. Les routes doivent être définies sur tous les routeurs et tous les hôtes du réseau pour que le routage fonctionne. Elles sont stockées dans des tables dites de routage. Tout hôte ou routeur possède donc une table de routage.
- Lorsque le nombre d'hôtes est important, on peut configurer leurs adresses IP automatiquement à l'aide du protocole DHCP. Ce protocole permet aussi de définir une passerelle par défaut, c'est-à-dire un routeur permettant de sortir du réseau local. Ainsi DHCP permet de ne pas avoir besoin de créer des routes dans les hôtes manuellement.
- Lorsque le nombre de routeurs est important, on peut utiliser un protocole de routage qui va remplir automatiquement les tables de routage des routeurs grâce à des échanges de messages. Plusieurs protocoles existent pour les réseaux de petites et moyennes tailles. Ils sont dits intra-domaines. On peut citer entre autres : RIP, IS-IS et OSPF.
- Un hôte peut être utilisé en tant que routeur lorsque l'*IP forwarding* est activé. Il aura alors en général deux interfaces réseau.

2. Mise en place

1. Ce TD utilise l'émulateur système **QEMU** avec le module noyau d'accélération matérielle **KVM** qui est installé sur les machines du CREMI. Pour activer **KVM** s'il n'est pas déjà installé, il faut taper les commandes :

```
ksu $USER
/usr/sbin/service qemu-kvm start
```
2. Créez-vous un répertoire de travail personnel `/espace/<login>` sur votre machine locale.
3. Récupérez l'image `debian1.qcow2.gz` pour les machines virtuelles hôtes, qui est située sur `/net/stockage/dmagoni` et copiez la sur votre machine locale dans votre répertoire `/espace/<login>`. Puis décompressez-la avec `gunzip`.
4. Ce TD utilise le logiciel `vswitch` pour émuler un commutateur Ethernet. Récupérez ce programme sur `/net/stockage/dmagoni` et copiez le dans votre répertoire de travail `/espace/<login>`. Placez les droits pour exécution avec `chmod u+x vswitch` puis lancez le par la commande : `./vswitch -d ./`.
5. Ce TD utilise quatre machines virtuelles émulant des PC standards exécutant une distribution **Debian** d'un système d'exploitation GNU/Linux. Elles serviront d'hôtes ou de routeurs selon les cas. Copiez le script `rout-stat.sh` depuis la page Web qui contient le sujet, dans votre répertoire `/espace/<login>`, placez les droits pour exécution avec `chmod u+x rout-stat.sh` puis lancez-le pour démarrer ces machines. Lorsqu'une machine a fini de démarrer, revenez sur la fenêtre du script et tapez sur la touche **Entrée**. Répétez ceci quatre fois afin de démarrer les quatre machines virtuelles.
6. Connectez-vous en tant qu'administrateur `root` sur chaque machine virtuelle en utilisant le mot de passe suivant : `root`.
7. Les commandes **UNIX** données ci-dessous sont à **compléter correctement**, grâce aux commandes données dans le fichier `TD-Instructions.pdf`, au `man` et au Web si besoin.
8. Pour chaque machine virtuelle, son interface `eth0` est connectée à l'Internet par SLIRP et elle est configurée par DHCP. Vous pouvez ainsi télécharger des paquets logiciels en utilisant la commande `apt-get install`.

3. Réseau sans routage

Un réseau interne simple sans routage est présenté sur la figure 1. Le réseau interne est un réseau local de type Ethernet. Utilisez le programme `vswitch` pour émuler le commutateur. Configurez vos quatre machines pour mettre en place cette topologie réseau.

Pour connecter le switch aux machines virtuelles, regardez dans le script `./rout-stat.sh` sur quels ports TCP écoutent les émulateurs QEMU et notez les.

9. Pour chaque lien à créer de chaque machine i de 1 à 4, vous devez taper les trois commandes suivantes dans le `vswitch` :

```
% conn qemu <connect_id_i> tcp 127.0.0.1 * 127.0.0.1 <port_i>
% add <interf_id_i>
% bind <connect_id_i> <interf_id_i>
```

Une fois ceci fait, vous pouvez vérifier les liens avec la commande `show ep` et les interfaces avec `show if`. De manière générale, tapez `help` pour avoir de l'aide. Vous pouvez aussi taper ces commandes dans un fichier texte puis le charger dans le `vswitch` avec la commande `load`.

10. Il faut ensuite configurer les interfaces IP de chacune des machines virtuelles. Vérifiez la connectivité avec `ping`.

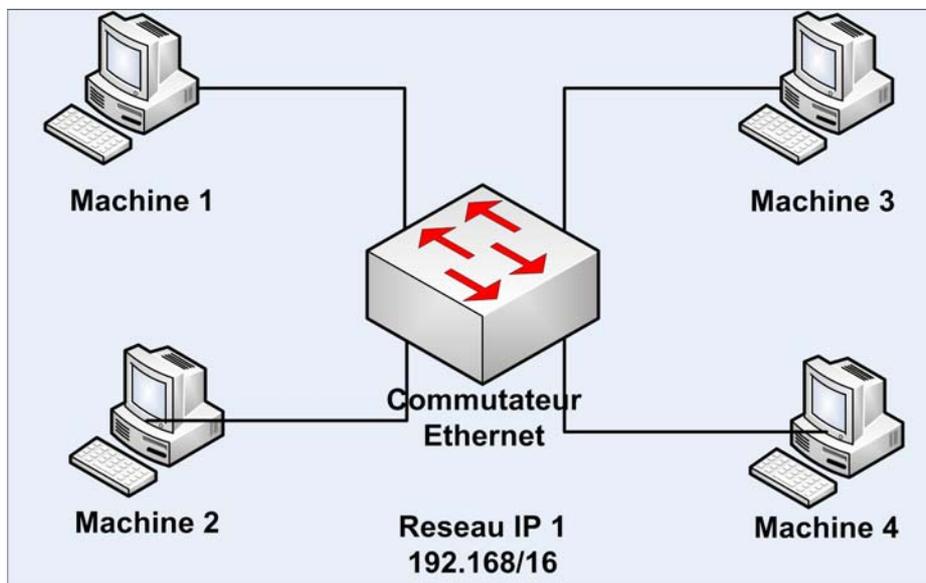


Figure 1. Réseau interne.

4. Configuration permanente des machines

La configuration réseau effectuée à la section précédente est volatile, c'est-à-dire qu'elle est perdue à chaque redémarrage du système. Afin de la rendre permanente, il est nécessaire de modifier un fichier de configuration du système qui permet la configuration systématique des interfaces réseaux utilisées parce qu'il est lu et interprété lors de l'exécution des scripts de démarrage.

11. Sur chaque machine, éditer le fichier `/etc/network/interfaces` à l'aide de l'éditeur `nano` ou `vi` pour y ajouter les lignes suivantes pour chaque interface :

```
auto eth<i>
iface eth<i> inet static
    address <address>
    netmask <netmask>
    network <network>
    broadcast <broadcast>
```

5. Réseau avec routage

Le but des manipulations proposées dans cette section est d'étudier et d'apprendre à utiliser le routage et les tables de routage d'IP. Configurer vos machines pour mettre en place la topologie du réseau de la figure 2. Par exemple, le premier réseau composé des trois premières machines sera adressé en **192.168/16** et interconnecté via la machine 3 jouant le rôle d'un routeur. Le deuxième réseau composé des deux dernières machines sera adressé en **10/8**.

12. Afin de permettre à un hôte de jouer le rôle d'un routeur interconnectant deux réseaux, positionner la variable `ip_forward` du noyau système à l'aide de la commande suivante :
`sysctl -w net.ipv4.ip_forward=1`
13. Pour que l'*IP forwarding* soit permanent :
 - a. modifier le fichier `/etc/sysctl.conf` avec la ligne `net.ipv4.ip_forward = 1`
 - b. puis tapez la commande `sysctl -p /etc/sysctl.conf`
14. Configurer les tables de routage avec la commande `route` pour créer le réseau de la figure 2.

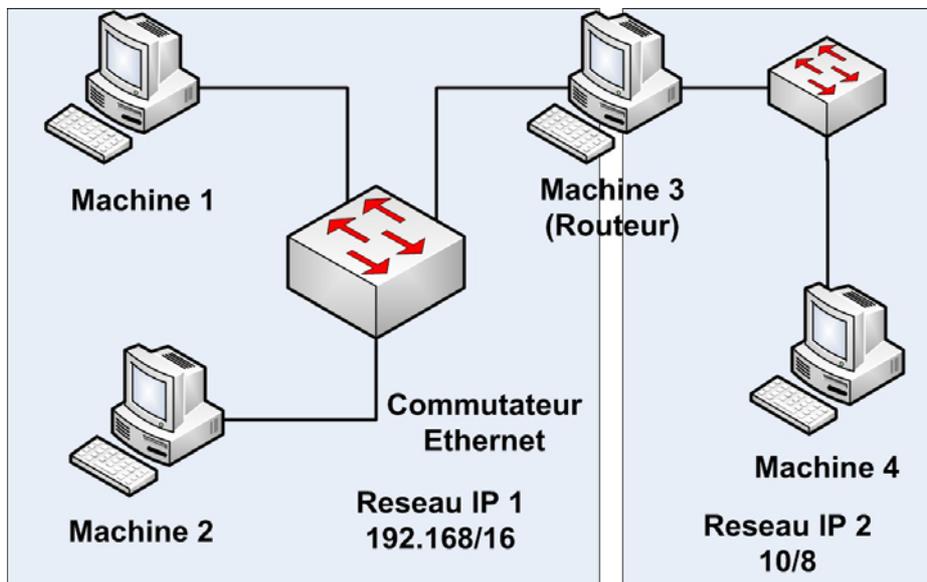


Figure 2. Réseau avec routage.

6. Sous réseaux avec routage

Le but des manipulations proposées dans cette section est d'étudier et d'apprendre à utiliser le sous-adressage IP et ses relations avec le routage. Sous réseau IP 1 et 2 sont deux sous-réseaux du réseau initial **192.168/16** de taille équivalente. Le trafic de sous réseau 1 est routé via le sous réseau 2. Dans la figure 3 et pour le réseau **192.168/16**, les machines 1 et 2 sont supposées appartenir à deux sous-réseaux différents d'une même organisation alors que la machine 3 est considérée être le routeur d'accès de cette organisation vers un opérateur offrant l'accès à un plus grand réseau, voire à l'Internet. La machine 4 est considérée comme le premier routeur de l'opérateur rencontré vers le reste du réseau, voire le reste du monde.

15. Déterminer la plage d'adresse de chaque sous-réseau et configurer les tables de routages.

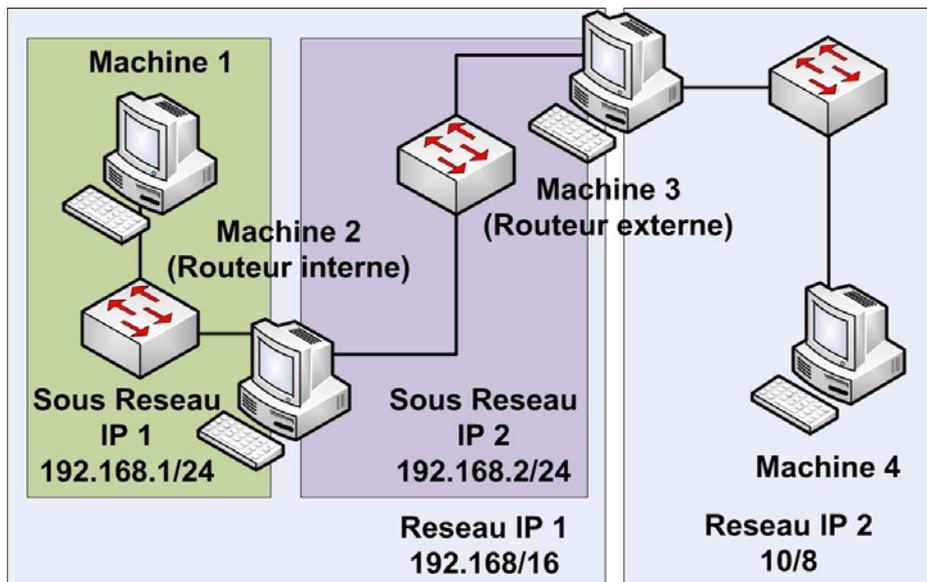


Figure 3. Sous réseaux avec routage.

7. Observation des protocoles

L'observation des protocoles réseau se fait grâce à un outil de capture de trames appelé **wireshark**. Cet outil écoute une interface réseau et décode / interprète les champs du protocole identifié. On y retrouve le format d'une trame telle qu'elle a été envoyée / reçue par l'interface réseau.

16. Installez **wireshark** dans la machine 3.

17. Lancez le mode graphique avec **startx** puis lancez **wireshark** et identifier l'échange lors de l'utilisation de la commande **ping** entre la machine 1 et la machine 4.

8. Sous réseaux avec routage global

Le but de cette section est d'interconnecter votre réseau précédent à un commutateur Ethernet général, géré par le chargé de TD, et de remplir vos tables de routage afin de pouvoir atteindre tous les réseaux des autres étudiants de la salle de TD. Comme indiqué sur la figure 4, vous devrez donc choisir les nombres **w**, **x**, **y** et **z** de manière unique dans la salle de TD afin de ne pas créer de conflit d'adresses.

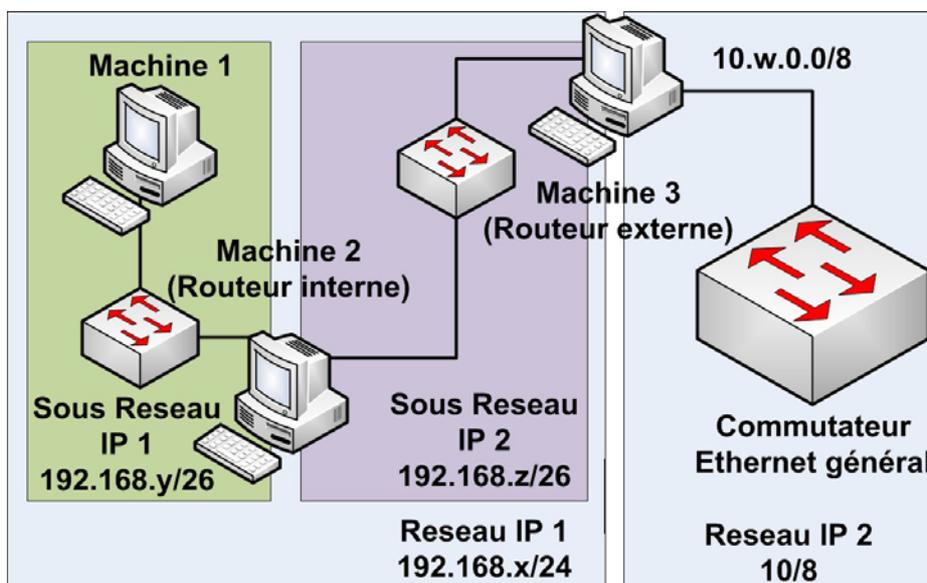


Figure 4. Sous réseaux avec routage global.