

# Routage IP

## Sous-réseaux

### TP N° 2

## 1 Rappels

L'ensemble des protocoles de la couche 3 du modèle OSI (appelée « couche réseau ») permet la communication entre les différents type de réseaux de niveau 2 (les « liens »). Plusieurs problèmes doivent être résolus, avec principalement :

- Hétérogénéité des réseaux
- Reconnaissance générale des machines (adressage universel)
- Acheminement des paquets (routage)

Dans Internet, des réseaux indépendants sont interconnectés et le protocole de la couche réseau IP fournit un service de communication universel. Ce service doit être indépendant de la structure et de la technologie utilisées localement sur chacun des réseaux.

Physiquement, deux réseaux (ou plus) sont interconnectés par l'intermédiaire d'une machine qui possède un point d'attache sur chacun des réseaux.

Cette machine qui joue un rôle particulier sur les réseaux s'appelle un *routeur* (ou passerelle ou en anglais *gateway*). Par opposition, les machines utilisateurs sont appelées *hôte*. Un routeur possède donc plusieurs interfaces réseaux et donc plusieurs adresses Internet. Voici un exemple comportant 2 réseaux locaux de type Ethernet sur lesquels sont connectés des hosts et 2 routeurs possédant chacun deux interfaces réseaux. Ces deux routeurs sont connectés via un réseau d'un autre type.

Le mécanisme permettant l'acheminement des paquets à bon port à travers un ensemble de réseau est appelé *routage*. Ce routage peut être plus ou moins perfectionné, par exemple "statique" (les routes sont fixées par un administrateur), ou encore "dynamique" (les routes peuvent varier dans le temps suivant différents critères : pannes, éventuellement charge sur les liens...). L'adresse Internet est partagée en deux parties :

- l'adresse du réseau sur lequel la machine est située ;
- l'adresse de cette machine sur ce réseau.

Le routage se fait au niveau des adresses des « réseaux ». C'est sur la première partie de l'adresse que se basent les routeurs pour faire parvenir les messages au réseau destination. Une fois les messages arrivés sur le réseau destination, c'est le protocole ARP qui permet d'acheminer les messages du dernier routeur vers les hôtes.

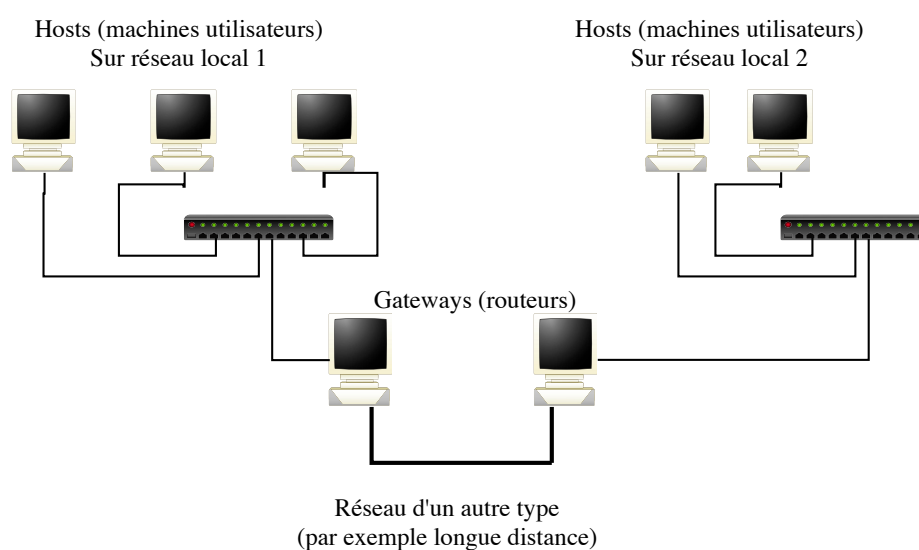


FIG. 1 –

Pour décider de la route à suivre, les routeurs gèrent une table appelée “table de routage” qui leur permet de répondre à la question : « d’après l’adresse destination du paquet, quel est le prochain routeur à qui envoyer le paquet pour qu’il arrive à destination ? ». Ce routeur est forcément un « voisin ». Une table de routage contient donc une liste de couples (adresse de réseau, adresse de routeur voisin) appelées routes. Le chemin complet n’est noté nulle part.

Si l’on prend l’analogie avec une voiture circulant sur un réseau routier. A chaque carrefour, le conducteur demande quelle est la route à prendre pour arriver au prochain carrefour sans jamais connaître le chemin complet jusqu’à la destination.

Les stations hôtes doivent également gérer une telle table de routage pour savoir à quel routeur il faut s’adresser sur son réseau local pour atteindre le réseau destination.

On peut distinguer deux fonctionnalités indépendantes dans le routage :

- la prise de décision sur la route à prendre au vue de la table de routage et de l’adresse destination contenu dans le paquet (entête IP), c’est l’aiguillage des paquets ;
- la mise à jour des tables de routage, basé sur une connaissance souvent partielle de la topologie du réseau.

## 1.1 Fichier `/etc/networks`

Tout réseau possède une adresse unique qui l’identifie. Cependant, il est plus facile de désigner les réseaux par des noms symboliques qu’on leur attribue. Ceci est rendu possible grâce au fichier de configuration `/etc/networks` qui permet de faire la correspondance entre les adresses réseaux et les noms qu’on leur donne. Les entrées du fichier `/etc/networks` sont de la forme suivante :

```
official-network-name network-number aliases
```

### 3 DÉROULEMENT DU TP

---

où **aliases** est un ensemble de surnoms qu'on peut donner au réseau en plus du nom officiel qu'on lui attribue.

## 2 Manipulation des tables de routage

Nous avons vu que toutes les stations, qu'elles soient hosts ou routeurs gèrent une table de routage. Nous avons la possibilité par l'intermédiaire de certains outils standards de manipuler cette table, comme par exemple afficher les entrées, ajouter ou supprimer une entrée... (voir la documentation sur les commandes systèmes).

Pour afficher la table de routage sous FreeBSD :

```
netstat -rn -f inet
```

Pour mettre à jour la table :

```
route add|delete destination gateway
```

Ex : `route add 172.16.0.0 192.168.2.4`

Pour rendre une machine routeur :

```
sysctl net.inet.ip.forwarding=1
```

Pour accéder au réseau du bâtiment F, le plus simple est d'appeler le script :

```
/var/backups/BackToNormal.
```

Pour configurer les routeurs Cisco, voir la documentation abrégée fournie.

## 3 Déroulement du TP

On pourra utiliser les fichiers **hosts** et **networks** permettant de manipuler des noms symboliques. Il est conseillé de respecter les numéros de réseau, de machine et d'interfaces apparaissant sur les différents montages pour pouvoir s'y retrouver facilement lors des observations.

### 3.1 Manipulation des tables de routage

Le but de cette manipulation est de vous remettre en mémoire les concepts de base liés au routage vus en première année. Nous utiliserons des routeurs (Cisco) possédant leur langage de configuration de commandes propre. Ces routeurs sont simplement des machines spécialisées, sans disque.

Supposons que l'on vous ait octroyé une seule adresse Internet de classe C (192.168.1/24 par exemple) pour le réseau local que vous avez à administrer. Vous voulez pour diverses raisons, découper votre réseau local en 3 sous-réseaux (imaginez ce que pourraient être ces raisons).

Vous voulez relier votre Intranet à l'ensemble d'Internet, pour cela vous louez une ligne spécialisée permettant de vous relier à Internet. L'adresse réseau de cette ligne vous est fournie : 10 (classe A) et l'on vous octroie l'adresse 10.0.0.1. On vous fournit aussi l'adresse du routeur –la station A en fait– par lequel vous devrez passer pour accéder au reste d'Internet : 10.0.0.2.

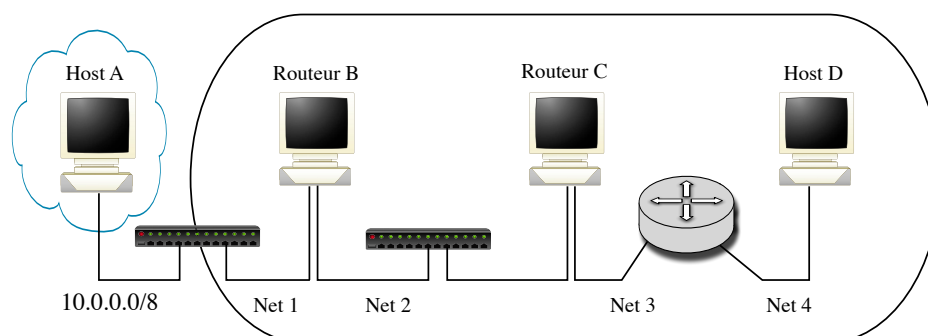


FIG. 2 – Le réseau

1. Reliez les quatre stations de la plate-forme comme en figure ?? : on pourra utiliser des câbles croisés pour économiser deux hubs.
2. **Faites un plan d'adressage de la plate-forme en notant précisément les adresses et les numéros d'interface.**  
Configurez les interfaces des machines et du routeur.
3. **Faites les configurations nécessaires de manière à ce que les quatre stations et le routeur puissent communiquer.**

**Remarques :** dans la commande `route` pour forcer une adresse « réseau », il faut rajouter l'option `-net` (ex : `route add -net 192.168.0.96 192.168.0.66`). Sinon l'utilitaire `route` prend l'adresse destination comme une adresse de machine. Pour spécifier un netmask différent des netmask standards associés aux classes (A, B, C), il faut ajouter l'option `-netmask` suivi du *netmask* en décimal ou spécifier le nombre de bits de la partie réseau.

Exemple :

```
route -v add -net 192.168.0.64 192.168.0.66 -netmask 255.255.255.192
ou bien route -v add -net 192.168.0.64/26 192.168.0.66
```

4. **Énumérez la liste des paquets échangés lors d'un ping de A vers D.**  
On supposera les tables ARP vides sur chaque machine. On donnera les adresses Ethernet et Internet apparaissant dans les entêtes respectives des paquets sur chaque réseau.
5. En utilisant la commande `tracert` mesurez les délais de transfert entre A et B, entre A et C, entre C et D et entre A et D. Capturez les paquets émis par cet utilitaire. Regardez en particulier le champ TTL de l'entête IP, avant et après la traversée d'un routeur.

**Quel est le principe de cet outil de mesure ?**

## 3.2 Fonctionnement du protocole RIP (Routing Information Protocol)

Le but de cette manipulation est de rappeler le principe de fonctionnement d'un des protocoles permettant le routage dans Internet : le protocole RIP.

1. Supprimer toutes les entrées statiques des tables de routages des stations et du routeur. Lancez les démons `routed` sur les stations et configurez le routeur pour qu'il utilise RIP. On lancera ces démons sur l'ensemble des stations avec l'option `-s` (*supply*) (`routed -s -P no_rdisc`) qui spécifie au démon de publier ses routes à ses voisins. l'option `-q` (*quiet*) démarre un démon silencieux, mais pas sourd.

**Capturez des paquets sur le réseau entre B et C.**

**Enumérez la liste des paquets RIP échangés et détaillez les informations contenues dans ces paquets.**

**Est ce que cette version de RIP implémente la méthode appelée « *split horizon* » (« horizon coupé ») ?**

2. Est ce que le contenu des paquets vous paraît satisfaisant étant donné le découpage en sous-réseaux de nous utilisons ?

**Remarque :** La première version de RIP était adaptée lorsque les tailles de préfixe pouvaient être déduite du début de l'adresse, ou si le préfixe était subdivisé en sous-réseaux tous de même taille. Voir l'annexe pour plus de détails

3. Lancez la version 2 de RIP sur les machines  
`routed [-q|-s] -P ripv2 -P no_rdisc`  
et les routeurs. Observer quelques paquets RIP. Quelles sont les informations supplémentaires circulant dans ces paquets ? À partir de maintenant nous utilisons la version 2 de RIP.

**Capturez les paquets circulant sur chaque sous réseau, en analysant les modifications qu'ils induisent aux tables de routage.**

4. Introduisez l'adresse du routeur B par défaut dans la table de C. Introduisez une adresse inconnue via le routeur B dans la table de C.  
Capturez des paquets RIP émis par C. Conclusion sur RIP et les routes statiques et par défaut ?
5. Lancez l'utilitaire `check-route` sur D. Lancez une capture de paquets entre le routeur et D. Tuez le demon `routed` sur C (`killall routed`). Puis relancez le.  
Que s'est-il passé ? Expliquez.
6. Faites cette fois un (`killall -9 routed` sur C, et attendez qu'un changement se produise sur D. Expliquez. (l'option `-9` supprime immédiatement le processus.)
7. Rajoutez un réseau entre A et D. Lancez RIP en mode *supply* sur A et D. Capturez les paquets RIP échangés sur ce réseau.

**Donnez les contenus des nouvelles tables de chaque machine. Expliquez en rappelant l'algorithme des démons de routages RIP ce qui s'est passé.**

### 3.3 Router Discovery

Lancez `routed` sur une station sans spécifier l'option `-P no_rdisc`.

1. Observez l'effet sur les stations voisines.
2. **Capturez les paquets correspondant. Expliquez leurs adresses de niveau 2 et de niveau 3**

### 3.4 Traceroute en vrai

Sur une réseau reliée au réseau de l'UFR, lancez un `traceroute`<sup>1</sup> à destination d'une machine distante (en dehors de l'université).

**Capturez les paquets correspondant et commentez le résultat : en particulier les délais affichés peuvent souvent être assez parlant.**

## Annexe : RIPv1 et les sous-réseaux

À la réception d'un paquet RIP regarde chaque ligne (`@net`, `métrie`). Dans le cas où il doit ajouter cette ligne à la table de routage (voir algo. de RIP), il détermine le `netmask` associé comme ceci :

```
Si (@net && netmask_standard_classe(@net) ) ==
  (@net_associé_à_une_interface_du_routeur && netmask_standard_classe(@net))
  alors prendre le netmask associé à cette interface.
Sinon
  prendre le netmask_standard_classe(@net) .
```

Le même algorithme est utilisé pour calculer les `@net` à mettre dans un paquet RIP qui sera envoyé sur une interface donnée. Pour chaque adresse `@net` de la table de routage pouvant être mis dans un paquet RIP sur l'interface d'adresse `@Interface` faire :

```
Si (@net && netmask_standard_classe(@net)) ==
  (@interface && netmask_standard_classe(@net))
  alors Adresse_mis_dans_le_paquet_RIP= @net
Sinon
```

```
  Adresse_mis_dans_le_paquet_RIP= @net && netmask_standard_classe(@net) .
```

Par exemple, pour une `@net` 192.168.0.128/26. Sur une interface d'adresse 172.16.0.1/24, le préfixe publié sera 192.168.0.0. Par contre, sur une interface d'adresse 192.168.0.190/25, le préfixe émis sera 192.168.0.128.

**Pouvez-vous imaginer un cas où RIPv1 pourrait être mis en défaut ?**

<sup>1</sup>utilisez l'option `-n` de façon à éviter un important trafic de résolution de nom.