

TP N° 1 de Réseaux

Interconnexions de réseaux : le problème du routage

Pascal Sicard

1 INTRODUCTION

Si ce n'est déjà fait, il est recommandé de commencer par la lecture de la documentation qui vous a été fournie sur le matériel (*Présentation de la plate-forme*).

Les manipulations que vous avez effectuées en L3 Miage étaient réalisées sur un seul réseau (dit "local"); c'est-à-dire l'interconnexion de plusieurs stations sur un même médium de communication. Le réseau Internet (INTERconnection NETwork) est une interconnexion de réseaux locaux.

Il existe une multitude de types de réseaux locaux. Ceci pour diverses raisons :

- Besoins spécifiques en termes de vitesse, débit... aboutissant à des réseaux de différentes natures (support physique, protocoles...).
- Contraintes physiques : pour un réseau comme Ethernet : limitation de la longueur du câble (quelques dizaines de mètres) et du nombre de machines reliées.
- Historique de développement et investissements effectués.
- ...

L'ensemble des protocoles de la couche 3 du modèle OSI permet la communication entre ces différents réseaux. Plusieurs problèmes doivent être résolus :

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

L'objectif de ce TP est d'étudier les moyens à mettre oeuvre pour interconnecter des réseaux, afin que toutes les stations des réseaux interconnectés puissent dialoguer entre elles.

Nous n'aborderons pas ici le problème de l'hétérogénéité des réseaux (les réseaux interconnectés sont tous du type Ethernet).

1.1 Rappel sur adresse Ethernet et adresse Internet :

Chaque interface Ethernet possède une adresse dite physique (ou Ethernet ou MAC) qui est fixée au moment de sa fabrication. Cette interface peut se situer sur la carte mère de l'ordinateur ou être ajoutée par la suite sous forme de carte amovible. Le protocole Ethernet est réalisé par *hard* (contrairement aux protocoles de niveau supérieur IP et TCP/UDP).

L'adresse Ethernet est composée de six octets. La notation habituelle pour ces adresses Ethernet consiste à écrire les six octets en hexadécimal et à les séparer par **:**.

Par exemple : **08 :00 :20 :40 :69 :d6**

Il existe d'autres types de réseaux au sens réseau physique et protocole de niveau 2. Ces réseaux utilisent d'autres types d'adresse. Il est donc nécessaire d'attribuer une adresse logique à chaque machine qui permet de faire abstraction de la nature des réseaux sous-jacents. Dans le monde Internet cela est fait au niveau de la couche IP (Internet Protocol : niveau réseau dans les couches OSI). Dans notre cas c'est l'**adressage Internet** (version 4 : IPv4) que l'on utilisera.

L'adresse IP est constituée de manière à identifier le réseau (au sens local) sur laquelle elle est connectée et à la distinguer des autres machines se trouvant aussi sur ce réseau. Deux parties distinctes dans une adresse IP :

- un numéro qui identifie le réseau sur lequel se trouve la machine : on parle de la partie **réseau** de l'adresse
- un numéro qui identifie la machine dans ce réseau : on parle de la partie **machine** de l'adresse.

Pour IPV4 cette adresse comporte quatre octets et est donnée sous la forme **$n_1.n_2.n_3.n_4$** où n_i est la valeur décimale d'un octet.

Historiquement, plusieurs **classes d'adresses** Internet existent suivant la taille de la partie réseau (un, deux, ou trois octets).

Voici en résumé, pour chaque classe, les bits réservés pour le codage de la partie réseau (bits r) et ceux réservés pour le codage de la partie machine sur le réseau (bits m).

La valeur des 3 premiers bits de l'octet de poids fort décide de la classe.

<i>Classe</i>	<i>Format des adresses</i>
A	0 rrrrrr.mmmmmmmm.mmmmmmmm.mmmmmmmm
B	10 rrrrr.rrrrrr.mmmmmmmm.mmmmmmmm
C	110 rrrr.rrrrrr.rrrrrr.mmmmmmmm

Exemples d'adresses :

- classe A (en décimal) : **55.22.45.12**
- classe B (en décimal) : **132.10.155.1**
- classe C (en décimal) : **195.1.10.41**

Adresse sans classe

Pour des raisons de pénurie et donc d'économie, les adresses IP peuvent (depuis 1990!) être attribuées sans tenir compte des classes. Il suffit de préciser le nombre de bit de la partie réseau de l'adresse.

Notation : 192.0.0.193/26

Le /26 indique que 26 bits de poids fort sont réservés pour la partie réseau. Il reste donc 6 bits pour la partie machine avec un /26.

ATTENTION : certaines applications ou commandes système tiennent compte encore des classes d'adresses. Par exemple *ifconfig* attribue par défaut le nombre de bit de la partie réseau associé à la classe de l'adresse si l'on ne précise pas ce nombre.

Notion de Netmask (masque de réseau)

Il permet aussi de préciser les parties machine et réseau d'une adresse IP. Il comporte des 1 en binaire sur la partie réseau. Par exemple le netmask (en décimal) de l'adresse 200.0.1.193/26 est 255.255.255.192. A noter que toutes les adresses /26 possède ce netmask.

Il permet de forcer à 0 la partie machine par un simple AND booléen et de calculer ainsi l'adresse du réseau auquel appartient une machine (une adresse). Par exemple $200.0.1.193 \text{ AND } 255.255.255.192 = 200.0.1.192$.

A SAVOIR : Il existe une deuxième version du protocole IP : IPversion6 ou IPv6. Ce protocole utilise des adresses sur 16 octets. La notation est en hexadécimal séparé par des " : ". Le découpage de l'adresse en deux parties machine et réseau existe toujours, ainsi que la notation /x pour le nombre de bit de la partie réseau.

Exemple d'adresse IPV6 : 2001 :0db8 :0000 :85a3 :0000 :0000 :ac1f :8001/56

1.2 "Philosophie" Internet

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.

La seule contrainte pour permettre à deux réseaux de communiquer est l'existence d'un chemin reliant les deux. Ce chemin peut être direct ou via d'autres réseaux.

Nous avons déjà vu comment connecter des machines sur un réseau. La question qui se pose : comment interconnecter des réseaux ?

Physiquement, deux réseaux (ou plus) sont interconnectés par l'intermédiaire d'une machine qui possède un point d'attache (une interface) 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 **HOTES** (**HOST** en anglais).

Un routeur possède plusieurs interfaces réseaux et donc plusieurs adresses Internet.

La figure 1 montre 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.

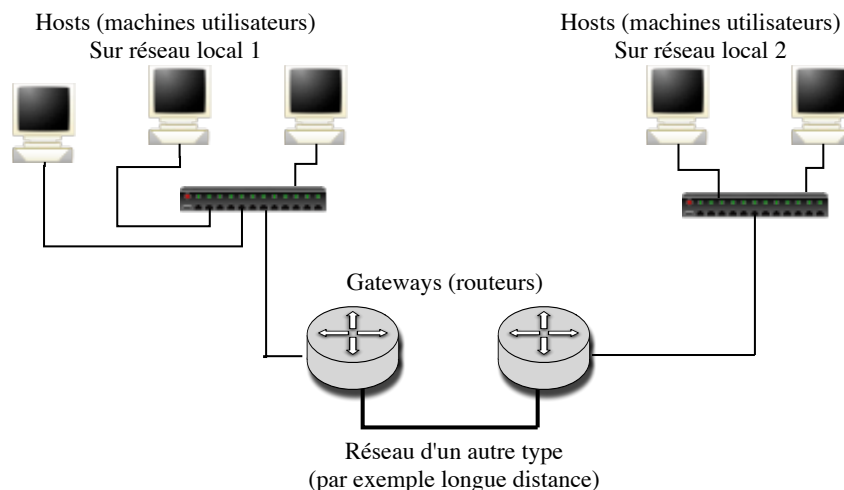


FIGURE 1 – Exemple d'interconnexion de réseaux

Le mécanisme permettant l'acheminement des paquets à bon port à travers un ensemble de réseau est appelé **ROUTAGE**.

Rappel : 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 hosts.

Pour décider de la route à suivre, les routeurs gèrent une table appelée "**table de routage**" qui leur permet de répondre à la question : *" Au vue de 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 (adresse de réseau, Netmask, adresse de routeur voisin). Le netmask permet de déterminer la partie réseau de l'adresse destination. Le chemin complet n'est pas noté dans les tables de routage.

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 "hosts" 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).
- la mise à jour des tables de routage.

1.3 Manipulation des tables de routage

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

Pour afficher la table de routage sous free BSD : **netstat -rn -f inet**

Pour mettre à jour la table : **route add|delete destination gateway**

Exemple :

route add 195.1.1.0 192.2.2.4

pour aller au réseau *195.1.1.0*, il faut passer par le routeur voisin *192.2.2.4*

Attention il faut spécifier les quatres octets de l'adresse du réseau en mettant la partie "machine" à 0.

A noter que l'on peut aussi spécifier un netmask particulier (différent de la classe standard);

Exemple : *route add 196.4.0.0/16 130.2.2.4*

2 DEROULEMENT DU TP

Ce TP est à faire en 2 séances de 3 heures.

Nous vous conseillons dans l'ensemble des manipulations qui suivent d'utiliser le fichier `/etc/hosts` afin de nommer chaque interface de vos machines. On pourra utiliser les conventions **netI** pour le réseau numéro I et **KJ** pour l'adresse de la machine K sur le réseau J.

Il est fortement conseillé de respecter exactement les différents montages apparaissant sur les figures de l'énoncé (numéro de réseau, de machine) et de faire **un plan d'adressage précis** (avant le montage) en notant sur les figures les noms des interfaces (em0 ...) et les adresses IP utilisées.

2.1 Manipulation de la table de routage

Le but de ces premières expérimentations est de vous familiariser avec les concepts de base de communication inter-réseaux.

 Reliez trois stations de la plate-forme suivant la figure 2.

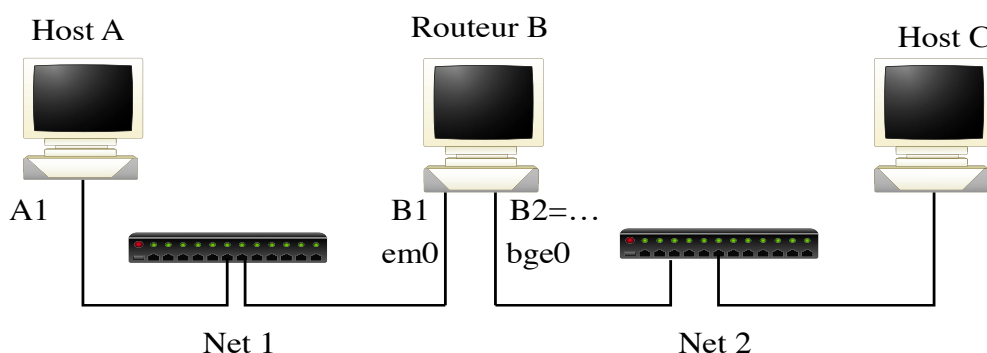


FIGURE 2 – Montage 2.1

Cette topologie faisant apparaître deux réseaux distincts, les adresses réseaux de A et C doivent être différentes. La machine centrale est utilisée comme un routeur.

Remarque : On peut pour économiser des hubs/switchs utiliser des câbles croisés.

 – 1 Faites un plan d'adressage mentionnant les adresses et le nom des interfaces réseaux que vous allez utiliser.



Configurez la ou les interfaces des trois stations.

Consultez la table de routage (**netstat -rn -f inet**). Expliquez précisément son contenu.

A l'aide de **ping**, déterminez les stations qui ne peuvent pas communiquer entre elles ; expliquez brièvement pourquoi.



Dans la table de routage de la station A, introduisez une entrée permettant d'atteindre le réseau 2 (**route add ...**).

Vérifiez le contenu de la table de routage par **netstat -rn -f inet** (en particulier le netmask associé à net2). On pourra visualiser les noms des machines (/etc/hosts) en enlevant l'option n à netstat (**netstat -r -f inet**).



Sur la station C, lancez l'utilitaire **pong** dont le rôle est de vous prévenir quand la station reçoit une requête de **ping** (un message de la forme "**echo request from...**" apparaît à l'écran...).



Recommencez **ping** de A vers C. Vous devriez constater que **ping** sur A bloque (mais n'affiche plus de message d'erreur), et que **pong** sur C ne réagit pas. Ceci est dû au fait que le routeur ne fait pas suivre la requête du réseau 1 vers le réseau 2.



Sur le routeur, lancez la commande suivante : **sysctl net.inet.ip.forwarding=1**

Le but de cette commande est de modifier le comportement de la machine, de façon à ce qu'elle permette le passage des paquets entre les deux réseaux lorsque cela est nécessaire.

La commande **sysctl -A** permet d'avoir l'état d'un grand nombre de paramètres système et surtout réseau (taille des tampons, timers...).




Recommencez **ping** de A vers C. Que constatez-vous sur les stations A et C ?


Expliquez pourquoi, et remédiez à ce problème (faites en sorte que le **ping** de A vers C fonctionne).

 Sur les stations A, B et C, lancez **arp -d -a** pour nettoyer leur table ARP.


Sur la station B, lancez l'utilitaire **Wireshark** afin de capturer les paquets sur les deux interfaces.


-  – **2** Lancez ping de A vers C. Analysez les paquets capturés, en particulier :
- les paquets ARP qui circulent. A quoi servent-ils ? Quelles machines sont concernées par ces paquets ARP ?
 - les adresses Ethernet et Internet des paquets de type ICMP.


Décrivez et expliquez l'enchaînement dans le temps des paquets échangés à l'aide d'un chronogramme.
Expliquez en détail le comportement et le travail du protocole IP dans le routeur B.

-  – **3** Quel doit être le contenu de la table de routage d'une station grâce à laquelle on veut pouvoir communiquer avec le "reste du monde" ?

Il est possible d'ajouter dans une table de routage une adresse d'un routeur par défaut. C'est-à-dire que l'on donne l'adresse d'un routeur pour toutes les adresses réseaux que l'on ne connaît pas explicitement (voir la commande **route** dans la documentation sur les commandes systèmes).

-  – **4** Pourquoi une route par défaut consiste à donner une adresse destination égale à 0.0.0.0 et un netmask égal à 0.0.0.0 ?

 Refaites la manipulation précédente en mettant dans les tables de routage de A et C seulement l'adresse du routeur B par défaut. On laissera les lignes "direct" (ou "link") des réseaux directement connectés.

-  – **5** Si l'on fait un ping depuis A vers une adresse inconnue (autre que celles que vous avez choisies), que se passe-t-il sur le réseau ? Quelle est la différence avec la configuration sans *défaut*.

 – 6 Quel est l'intérêt/inconvénient de ce routeur par défaut ? Conclusion ?

2.2 Observation du routage automatique

Le but de cette manipulation est de découvrir le routage automatique RIP (Routing Internet Protocol) par l'intermédiaire de processus particulier (appelés démons) qui effectue la mise à jour automatique des tables de routage.

Le rôle des **démons** RIP sur les différentes stations est de se communiquer les routes permettant d'atteindre tous les réseaux locaux. Le démon s'appelle **routed** et peut être lancé avec deux options :

- **-s** (supply) pour informer les autres stations des réseaux qu'elles peuvent atteindre en transitant par la station sur laquelle il est lancé.
- **-q** (quiet) pour prendre en compte les chemins qui lui sont communiqués sans informer les stations des chemins qu'il connaît.

*En résumé, sur les routeurs, on lance généralement **routed** avec l'option **-s**, et sur les hosts avec l'option **-q**.*


Les démons **RIP** utilise UDP pour la communication de ses messages.

Remarque : Des paquets IGMP (gestion du multicast) peuvent apparaître lors des manipulations. Vous n'en tiendrez pas compte, il n'interfère pas ici avec RIP. Pour résumer ils permettent à RIPv1 de dialoguer éventuellement avec RIPv2 qui utilise le multicast pour diffuser ses paquets (au lieu du broadcast)

 Gardez la dernière topologie réalisée dans la manipulation précédente, puis :

- Nettoyez les tables de routages sur les stations A et C (par **route flush -inet**) ;
- Lancez le démon **routed -q -P no_rdisc** sur les stations A et C ;
- Lancez le démon **routed -s -P no_rdisc** sur la station B.

Note : dans la manipulation précédente, la station B a été paramétrée de façon à faire passer les paquets d'un réseau à un autre. Il est donc inutile de la paramétrer de nouveau ; mais notez bien que ce paramétrage est indispensable, et que ce n'est pas le démon qui le réalise.


 – 7 Demandez par **netstat -rn -f inet** le contenu des tables de routage des trois stations. Que concluez-vous ?

Vérifiez par un ping que les stations A et C peuvent communiquer.


 – **8** Sur la station B capturez 2 paquets sur chaque réseau.


L'opération ne devrait pas durer plus de 1 ou 2 minutes. Analysez les paquets capturés. Comment les routeurs se comportent-ils vis à vis des hosts ? (quelles sont les informations communiquées avec le protocole RIP ?)


Quel est l'intervalle de temps qui sépare l'émission automatique de deux paquets RIP consécutifs sur un même réseau ?


 – **9** Sur la station A, lancez l'utilitaire **check-route** (il affiche à intervalles de 10 secondes le contenu de la table de routage), puis débranchez la station A du réseau 1 (afin de simuler une rupture du réseau).

Notez le changement qui doit normalement intervenir dans la table de A quelques minutes après sa déconnexion. Quelle est la durée du timer d'effacement de RIP ?

 Reconnectez la station sur le réseau, et notez un nouveau changement dans la table qui doit normalement intervenir dans les 30" qui suivent.

 – **10** Expliquez le fonctionnement du protocole RIP en ce qui concerne la mise à jour de la table de routage.

 – **11** Tuez le démon de routage sur B (**killall routed**), capturez les paquets RIP qui circule à ce moment là sur le réseau. Expliquez le contenu de ces paquets et leur rôle.

 – **12** Relancez le démon de routage sur B. Tuez et relancez le démon sur C, capturez les paquets RIP qui circule sur le réseau au moment où on relance le démon. Expliquez le contenu de ces paquets et leur rôle.

2.3 Troisième manipulation

Sur la topologie précédente, insérez la quatrième station D afin d'obtenir la configuration donnée dans la figure 3 (faisant apparaître un nouveau réseau).

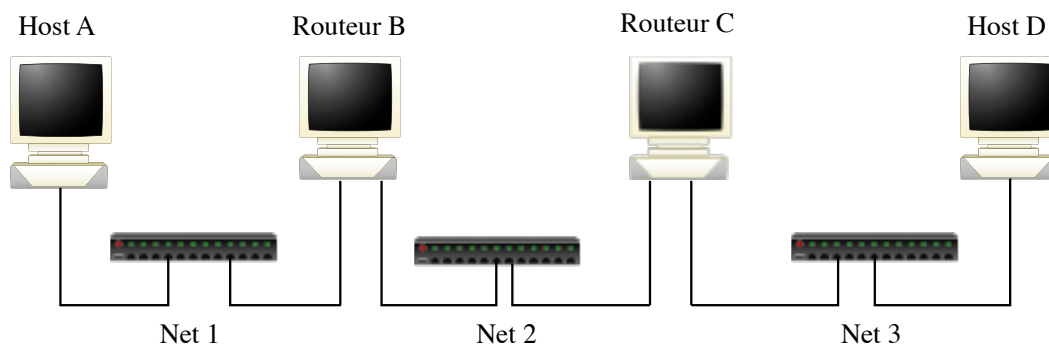





FIGURE 3 – Montage 2.3

 – **13** Faites un plan d’adressage mentionnant les adresses et le nom des interfaces réseaux que vous allez utiliser.


2.3.1 Routage statique

 Supprimez tous les démons, puis nettoyez toutes les tables de routage (**route flush -inet**).

 – **14** Donnez le contenu des tables de routages permettant que toutes les machines communiquent.

 Remplissez ces tables ”à la main”.


Faites les manipulations nécessaires de manière à ce que les quatre stations puissent communiquer entre elles.


 – **15** Videz les tables ARP sur toutes les machines. Enumérez la liste des paquets échangés lors d’un ping de A vers D à l’aide d’un chronogramme.


2.3.2 Routage automatique


 Lancez sur chaque station le démon RIP.

ATTENTION, vous prendrez soin de nettoyer les tables de routage de toutes les stations auparavant !

 – **16** Observez les paquets émis lors du lancement des démons. A quoi servent-ils ?


 – **17** Enumérez la liste des paquets RIP échangés et détaillez les informations contenues dans ces paquets (depuis le lancement des démons jusqu'à la stabilisation des tables où « tout le monde connaît tous les réseaux »).
Est ce que les routeurs attendent toujours 30 secondes entre l'émission de chaque paquet RIP, pourquoi ?

 – **18** Pourquoi les paquets RIP envoyés par B à C ne contiennent-ils pas le réseau net3 (ou le contiennent avec une métrique de 16) alors qu'il apparaît dans la table de routage de B ?

 Lancez une capture sur les deux interfaces de C. Tuez ensuite le demon RIP sur B.


 – **19** Observez et expliquez l'utilité des paquets qui ont circulé à ce moment là sur le reseau 2 et 3.

 Lancez une capture sur les deux interfaces de C. Relancer le demon RIP sur B.

 – **20** Est ce que C propage sur net3 l'adresse de net1 tout de suite après l'avoir reçu depuis B ? En d'autres termes est ce que cette version de RIP implémente la méthode dites de la *mise à jour déclenchée* dans ce cas ?
Complétez la partie de l'algorithme de RIP correspondant.

2.4 Quatrième manipulation

Le but de cette manipulation ainsi que de la suivante est de compléter le principe de fonctionnement du protocole RIP que vous avez décrit à la fin de la deuxième manipulation de ce TP.

 – **21** Faites un plan d'adressage mentionnant les adresses et le nom des interfaces réseaux que vous allez utiliser.

 Connectez les quatre stations comme indiqué dans la figure 4.

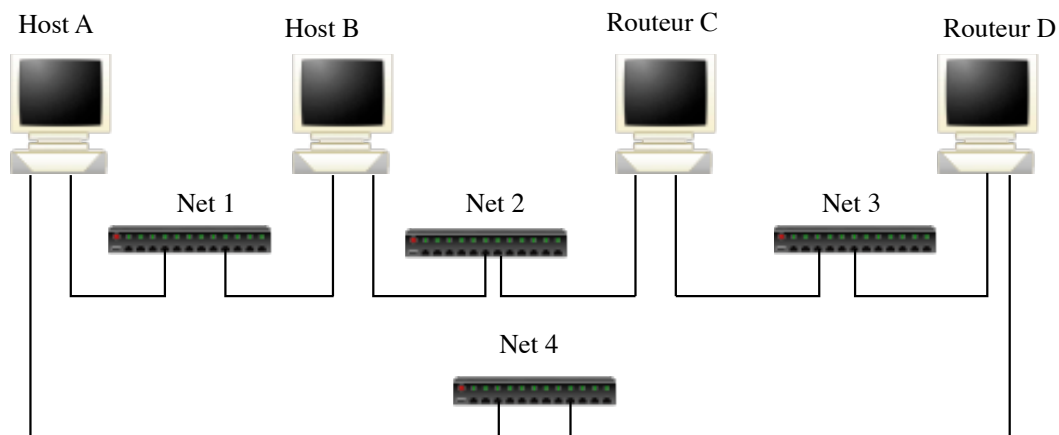




FIGURE 4 – Montage 2.4

- Supprimez les éventuels démons de routage qui pourraient tourner sur les stations, puis nettoyez toutes les tables de routage.
- Sur les stations A et B, lancez le démon **routed -q -P no_rdisc**.
- Sur les stations C et D, lancez le démon **routed -s -P no_rdisc**.


 – **22** Consultez la table de routage de A, et notez le chemin qui lui permet d'accéder au réseau 2. Justifiez le choix de ce chemin. Donnez le contenu des paquets RIP circulant sur chaque réseau. Expliquez leur contenu.

Le réseau 1 est-il connu de C et D ? Pourquoi ?

 – **23** Supprimez le démon de routage sur la station B, puis relancez le en mode ”**supply**” (-s). Quel chemin permet désormais d’accéder au réseau 2, sur A ?


Pour quelle raison ce nouveau chemin a-t-il été choisi ?

Pour vous aider à répondre capturez un paquet sur chaque interface de A et analysez les informations qu’ils contiennent.

 – **24** Complétez le principe de fonctionnement du protocole RIP décrit à la fin de la deuxième manipulation.

2.5 Cinquième manipulation


Nous allons profiter de cette dernière expérimentation pour s’initier à la configuration d’un routeur CISCO. Si vous n’avez pas assez de temps (voir avec l’enseignant) continuez à utiliser des PCs.

 – **25** Faites un plan d’adressage mentionnant les adresses et le nom des interfaces réseaux que vous allez utiliser.

Remplacez le PC routeur B par un vrai routeur CISCO. Consultez la documentation associée à ces routeurs pour pouvoir configurer les interfaces et mettre en place le routage.

 Connectez les quatre machines comme indiqué dans la figure 5.

- Supprimez les éventuels démons de routage qui pourraient tourner sur les stations, puis nettoyez toutes les tables de routage.
- Lancez **routed -q -P no_rdisc** sur la station C, et **routed -s -P no_rdisc** sur les autres stations (ou routeur).

 – **26** Notez la route qui permet d’accéder au réseau 3 à partir de A, et faites un ping de A vers C pour vérifier que ces deux stations peuvent effectivement communiquer. Pourquoi ce chemin a-t-il été choisi ?

 Déconnectez la station A du réseau emprunté pour communiquer avec C.

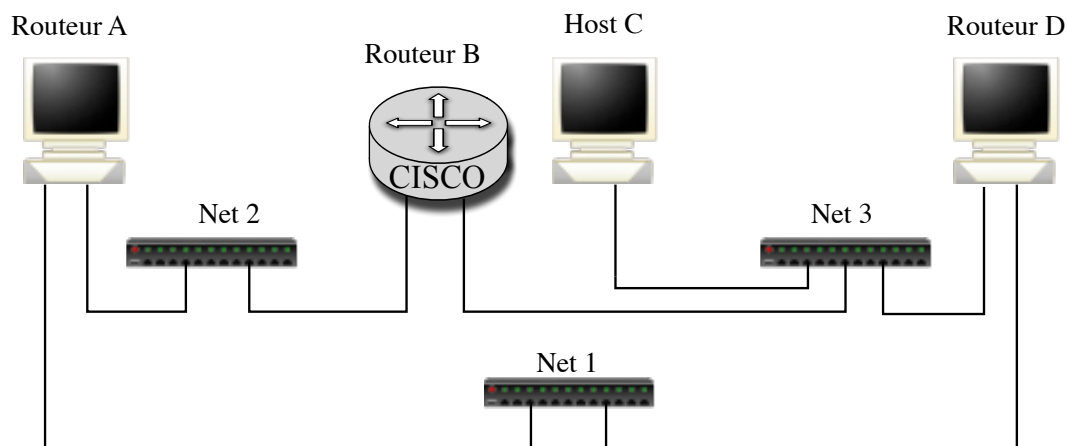




FIGURE 5 – Montage 2.5


 – **27** Refaites un ping de A vers C. Pourquoi ne fonctionne-t-il plus ? Laissez « tourner » ce **ping** jusqu’à ce qu’il fonctionne à nouveau (cela peut prendre quelques minutes).

Reconsultez la table de routage de la station A. Que constatez-vous ? Expliquez ce qu’il s’est passé.

 – **28** Au vue de l’ensemble de ces observations répondez aux questions suivantes :

- Quel est le contenu des paquets RIP émis par un routeur ?
- Est ce le même paquet RIP qui est émis sur toutes les interfaces ?
- RIP mémorise t-il plusieurs routes quand cela est possible ?
- Comment un routeur calcule-t-il la partie “réseau” de l’adresse des réseaux auxquels il appartient ?
- Les paquets RIP contiennent-ils les netmasks associés aux adresses qu’ils transportent ?
- Comment RIP (version 1) calcule-t-il les netmasks de l’adresse des réseaux qu’il met dans la table de routage ?
- Le traitement d’une ligne contenue dans un paquet RIP est-il le même quelle que soit l’adresse du routeur émetteur du paquet RIP ?

- Que fait RIP si un routeur ré-annonce une route connue mais avec un coût supérieur à celui précédemment annoncé ?
- Que fait RIP à la réception d'un paquet RIP « Request » ?
- Que fait RIP lorsque le démon est arrêté sur une machine ?

 – **29** Compléter de façon informelle le corps de l'algorithme suivant dans les deux cas : démons en mode -q (non routeur) et -s (routeur) :

tantque vrai **faire**

attendre (événement)

- **si** événement est “réception d'un paquet RIP response”
Pour chaque adresse contenue dans le paquet RIP faire :
...
finsi
- **si** événement est “réception d'un paquet RIP request”
...
finsi
- **si** événement est expiration du timer associé à une entrée dans la table de routage
...
finsi
- **si** événement est expiration du timer d'émission de paquet RIP (seulement pour un routeur)
attention au choix des adresses à envoyer suivant l'interface d'émission...
finsi
- ... (autres événements ?)

fin tantque

3 Pour ceux qui veulent aller un peu plus loin

Essayer sur la dernière configuration de votre plate-forme de faire tourner la version 2 de RIP. La commande est **routed -s -P no_rdisc -P ripv2**.

- Faites des captures sur les réseaux.
- Quelles sont les différences majeures entre les paquets RIP version 1 et 2 ?
- Conclusions ?