

TP 1 routage

Le but de ce TP est de vous faire comprendre et de vous faire manipuler sous Linux :

- La configuration des interfaces réseau,
- Les tables de routage.

Rappels : le routage

- Deux machines sur le même segment peuvent directement communiquer (ARP)
- Pour atteindre un autre réseau ou sous-réseau, on a besoin d'informations de routage :
 - statiques,
 - dynamiques.
- Chaque équipement possède une table de routage qui contient :
 - l'adresse du réseau local,
 - l'adresse de chacun des réseaux distants autorisés,
 - une entrée par défaut au cas où aucune entrée ne convient

Algorithme de routage

```
si (@IP_dest & masque == mon@IP & masque)
  alors envoi_direct (datagramme, @IP_dest)
sinon
  envoi_indirect(datagramme, @IP_dest, routeur(@IP_dest & masque))
```

Exemples :

- envoi de 192.168.30.1 vers 192.168.30.2
 - même réseau de classe C donc envoi direct
- envoi de 192.168.30.1 vers 192.168.31.1
 - deux réseaux différents de classe C donc envoi par le routeur qui est par exemple connecté aux deux réseaux (donc le routeur a **deux adresses IP**)
 - envoi de 192.168.30.1 au routeur (exemple 192.168.30.254)
 - envoi du routeur 192.168.31.254 vers 192.168.31.1
 - Mais attention il peut y avoir plusieurs routeurs entre deux réseaux !



Dans le cas où deux réseaux sont reliés par plusieurs routeurs, les datagrammes IP vont transiter de routeur en routeur jusqu'à la machine destination.
Les routeurs **ne connaissent pas les chemins complets**, ils connaissent simplement le routeur suivant sur la route.
Un routeur peut avoir plus de deux interfaces réseau et donc être connecté à plus de deux réseaux.

Commandes utiles

- Configuration des adresses IP des interfaces par la commande `ifconfig`
Chaque interface est identifiée par un nom :
 - `eth0` : première carte Ethernet
 - `lo` : *loopback* ou interface de bouclage

Liste des interfaces réseau configurées :

```
ifconfig
```

Pour configurer une interface réseau :

```
ifconfig interface adresse_IP netmask masque_de_réseau up
```

Exemple :

```
ifconfig eth0 192.168.10.1 netmask 255.255.255.0 up
```

- Configuration de la table de routage : la commande `route`

Affichage de la table de routage :

```
route -n
```

Pour ajouter une entrée de réseau à la table de routage :

```
route add -net adresse_réseau_IP netmask masque_de_réseau gw  
adresse_IP_routeur
```

Exemple :

```
route add -net 192.168.100.0 netmask 255.255.255.0 gw 192.168.10.254
```

Pour ajouter un routeur par défaut à la table de routage :

```
route add default gw adresse_IP_routeur
```

Exemple :

```
route add default gw 192.168.10.2
```

Pour supprimer une entrée de réseau de la table de routage :

```
route del -net adresse_réseau_IP netmask masque_de_réseau
```

Exemples :

```
route del -net 127.0.0.0 netmask 255.0.0.0  
route del -net 192.168.10.0 netmask 255.255.255.0
```

Pour supprimer la route par défaut de la table de routage :

```
route del default
```

Lancez Marionnet et créez un nouveau projet en allant dans le menu **Projet > Nouveau**, et donnez-lui un nom.

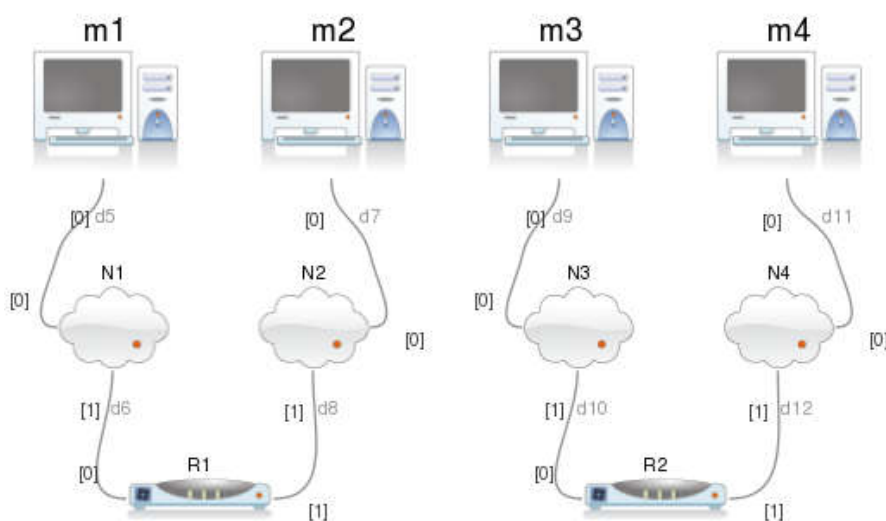
Vous utiliserez dans un premier temps 4 hôtes (*m1*, *m2*, *m3*, et *m4*) répartis dans les 4 réseaux suivants à raison d'un hôte par réseau (l'hôte *m1* dans le réseau *N1*, l'hôte *m2* dans le réseau *N2* ...)

- *N1* : 192.168.10.0
- *N2* : 192.168.110.0
- *N3* : 192.168.20.0
- *N4* : 192.168.120.0

Vous utiliserez 2 routeurs, *R1* et *R2* pour relier les réseaux entre eux, le routeur *R1* interconnectant les réseaux *N1* et *N2* et le routeur *R2*, les réseaux *N3* et *N4*.

Dans chaque réseau, vous utiliserez la première adresse pour numéroté l'interface de l'hôte et la dernière adresse pour l'interface du routeur. La machine *m1* aura donc comme adresse IP 192.168.10.1 et le routeur *R1* l'adresse 192.168.10.254 dans le réseau *N1*, la machine *m2* et le routeur *R1* auront respectivement les adresses 192.168.110.1 et 192.168.110.254 dans le réseau *N2* et ainsi de suite.


Sous Marionnet, vous utiliserez le composant « nuage » pour représenter les réseaux et le composant « routeur », pour relier les réseaux entre eux.



Sous Marionnet, pensez à cocher la case « Show unix terminal » lorsque vous créez un routeur, pour que vous puissiez vous connecter dans un terminal comme avec les hôtes.

Veillez également à câbler vos routeurs sur les bons ports, sachant que le port0 correspond à eth0, port1 à eth1, port2 à eth2 ... Par exemple si vous reliez le port1 de N1 sur le port2 de R1 et que sur R1 vous définissez l'adresse IP 192.168.10.254 sur l'interface eth0, vous risquez d'avoir quelques petits soucis de liaison !!! (Question subsidiaire : sur quelle interface faudrait-il dans ce cas définir l'adresse IP ?)

Add router

 Name:

Label:

Hardware

Ports number:

Port 0 address: . . . /

Software

Distribution:

Variant:

Kernel:

Access

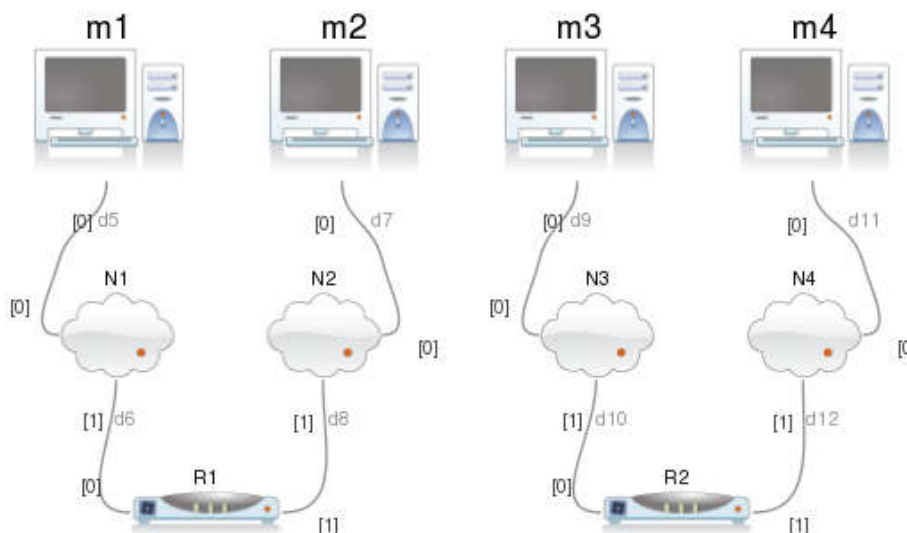
Show unix terminal:

Au moment d'ajouter les machines *m1* à *m4*, choisissez la distribution « guignol ».

Travail à réaliser : étape ①

- Définition des adresses IP des interfaces réseau
- Configuration des tables de routage
- Test d'accessibilité avec la commande `ping`
 - client ↔ routeur
 - client ↔ client via routeur

La première étape du TP consiste à configurer les adresses IP et tables de routage des hôtes et des routeurs pour chaque groupe de 2 réseaux ($N1$ et $N2$, $N3$ et $N4$). Vous pouvez utiliser l'onglet « Interfaces » de Marionnet pour configurer les adresses IP des interfaces réseau (faites-le **AVANT** de démarrer les machines et routeurs).



Pour vérifier que la configuration est correcte utilisez la commande `ping` sur chaque machine pour essayer d'atteindre l'autre machine.

Exemple entre N1 et N2 :

Sur *m1* on veut vérifier qu'on atteint bien les deux adresses du routeur et la machine *m2* qui se trouve sur l'autre réseau. Dans ce cas, on lancera les commandes suivantes sur la machine *m1* :

- `ping 192.168.10.1` (vérifie que l'on peut s'atteindre soi-même)
- `ping 192.168.10.254` (vérifie que l'on peut atteindre le routeur)
- `ping 192.168.110.254` (vérifie que l'on peut atteindre la 2^{ème} interface du routeur)
- `ping 192.168.110.1` (vérifie que l'on peut atteindre la station sur l'autre réseau)

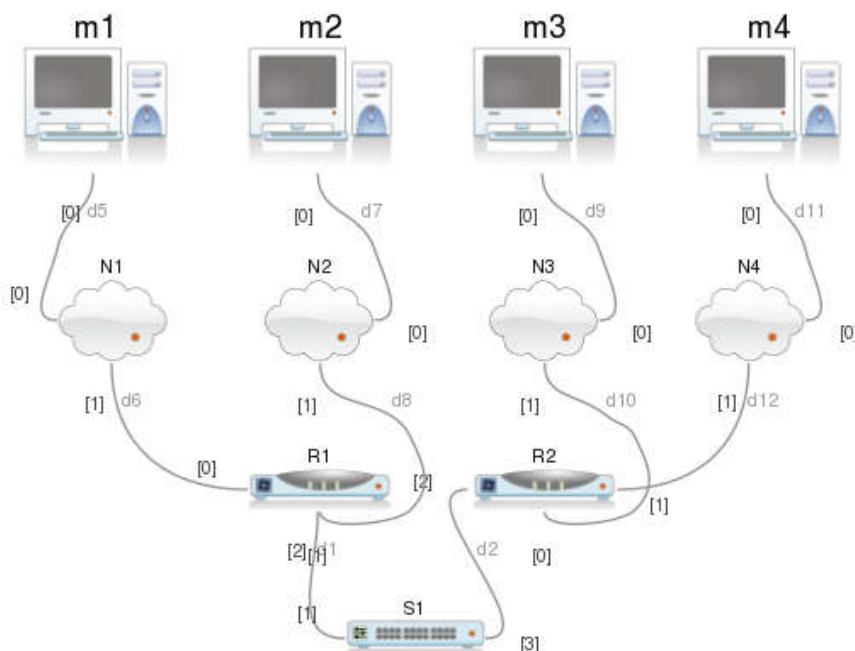
Travail à réaliser : étape ②

- Définition 3^{ème} adresse IP du routeur sur réseau 192.168.200.0
- Configuration des tables de routage pour atteindre les deux autres réseaux
- Test d'accessibilité avec la commande `ping`
 - $R1 \Leftrightarrow R2$
 - $m1, m2 \Leftrightarrow m3, m4$

La deuxième étape consiste à relier le routeur *R1* au routeur *R2* et à modifier la configuration des tables de routage pour que toutes les machines ainsi réunies puissent communiquer entre elles.

Le réseau qui relie les routeurs est le 192.168.200.0. Les routeurs devront utiliser les adresses suivantes :

- *R1* : 192.168.200.1
- *R2* : 192.168.200.2



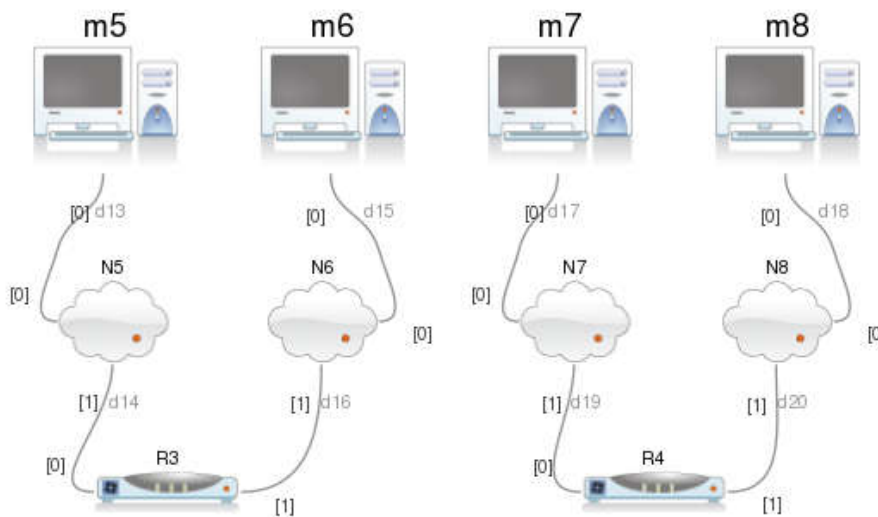
Vérifiez que depuis *m1*, vous pouvez faire un `ping` sur *m3* et *m4*.

Travail à réaliser : étape ③

De la même façon que vous l'avez fait lors des étapes (1) et (2), ajoutez 4 hôtes supplémentaires *m5*, *m6*, *m7* et *m8*, qui seront respectivement repartis dans les réseaux *N5*, *N6*, *N7* et *N8*, reliés 2 à 2 par les routeurs *R3* et *R4*.

- *N5* : 192.168.30.0
- *N6* : 192.168.130.0
- *N7* : 192.168.40.0
- *N8* : 192.168.140.0

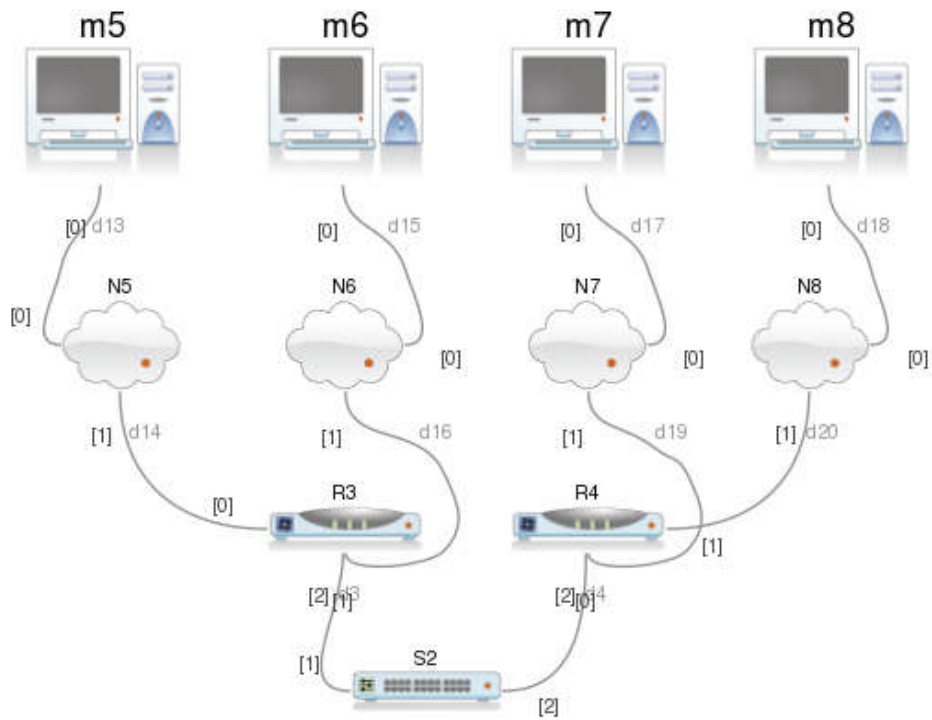
Commencez par configurer les adresses IP et tables de routage des hôtes et des routeurs pour chaque groupe de 2 réseaux (*N5* et *N6*, *N7* et *N8*). Dans chaque réseau, utilisez la première adresse pour numéroté l'interface de l'hôte et la dernière adresse pour l'interface du routeur.



Reliez ensuite le routeur *R3* au routeur *R4*, et modifiez la configuration des tables de routage pour que toutes les machines ainsi réunies puissent communiquer entre elles.

Le réseau qui relie les routeurs est le 192.168.200.0. Les routeurs devront utiliser les adresses suivantes :

- *R3* : 192.168.200.3
- *R4* : 192.168.200.4

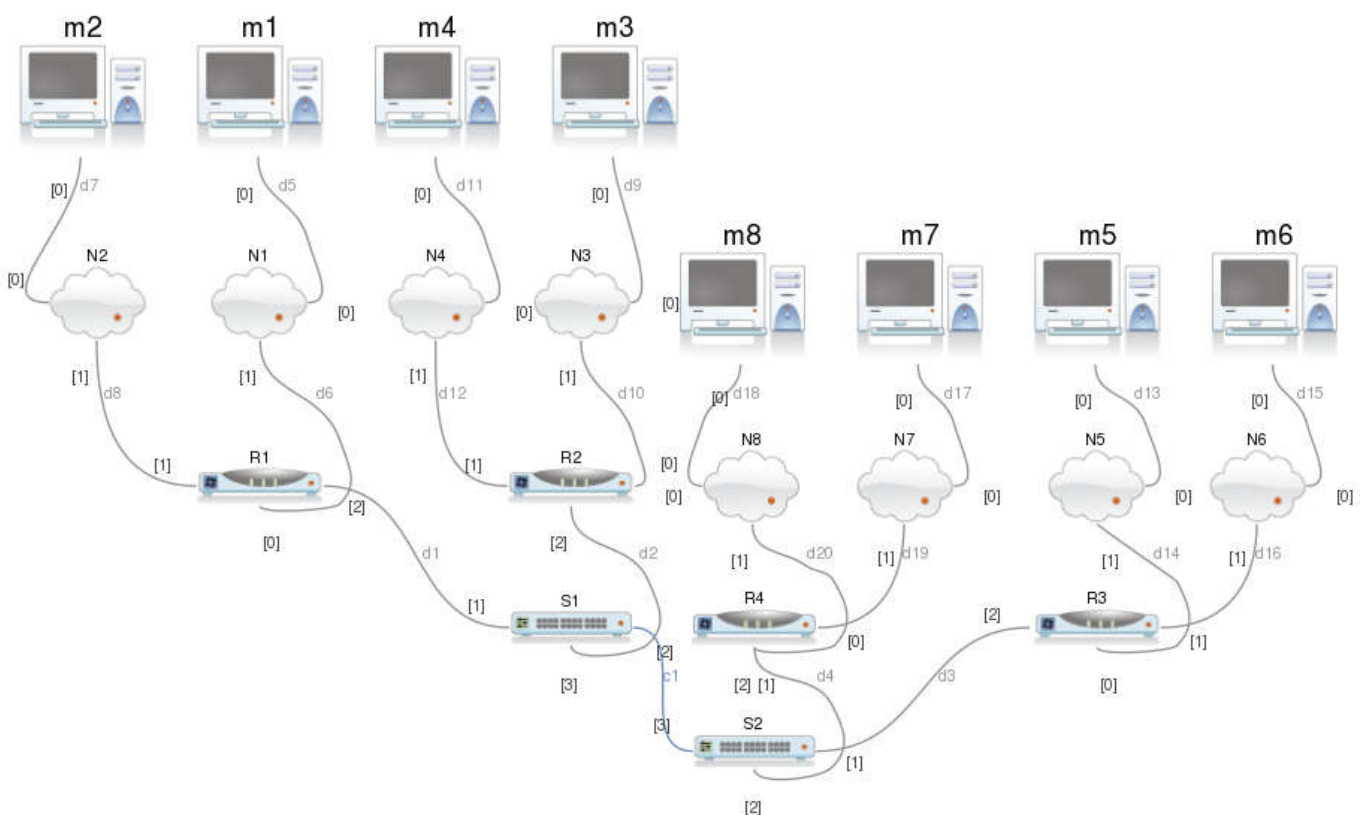


Vérifiez que depuis *m5*, vous pouvez faire un ping sur *m7* et *m8*.

Travail à réaliser : étape 4

- Configuration des tables de routage pour atteindre tous les réseaux
- Test d'accessibilité avec la commande `ping`

La dernière étape consiste à interconnecter tous les réseaux et à modifier la configuration des tables de routage pour que les 8 machines puissent communiquer entre elles.



Pour atteindre chacun des réseaux il est nécessaire de préciser le routeur à utiliser (*gateway*)

Exemple :

```
route add -net 192.168.10.0 netmask 255.255.255.0 gw 192.168.200.1
```

Vous pouvez utiliser la commande `tracert -n adresse_IP` pour vérifier que les paquets passent bien par les chemins prévus. Par exemple entre *m1* et *m8* : `tracert -n 192.168.140.1 sur m1`.

Travail à réaliser : étape ⑤

Modifiez la topologie physique du réseau en remplaçant les « nuages » et tous les commutateurs ou hubs présents par un seul et unique commutateur (pensez à prévoir suffisamment de ports). Vérifiez que la topologie logique du réseau n'a pas été affectée.