

TP 2 : mon premier réseau virtuel

Le but de ce TP est de vous familiariser avec l'outil de conception de réseaux virtuels [Marionnet](#).

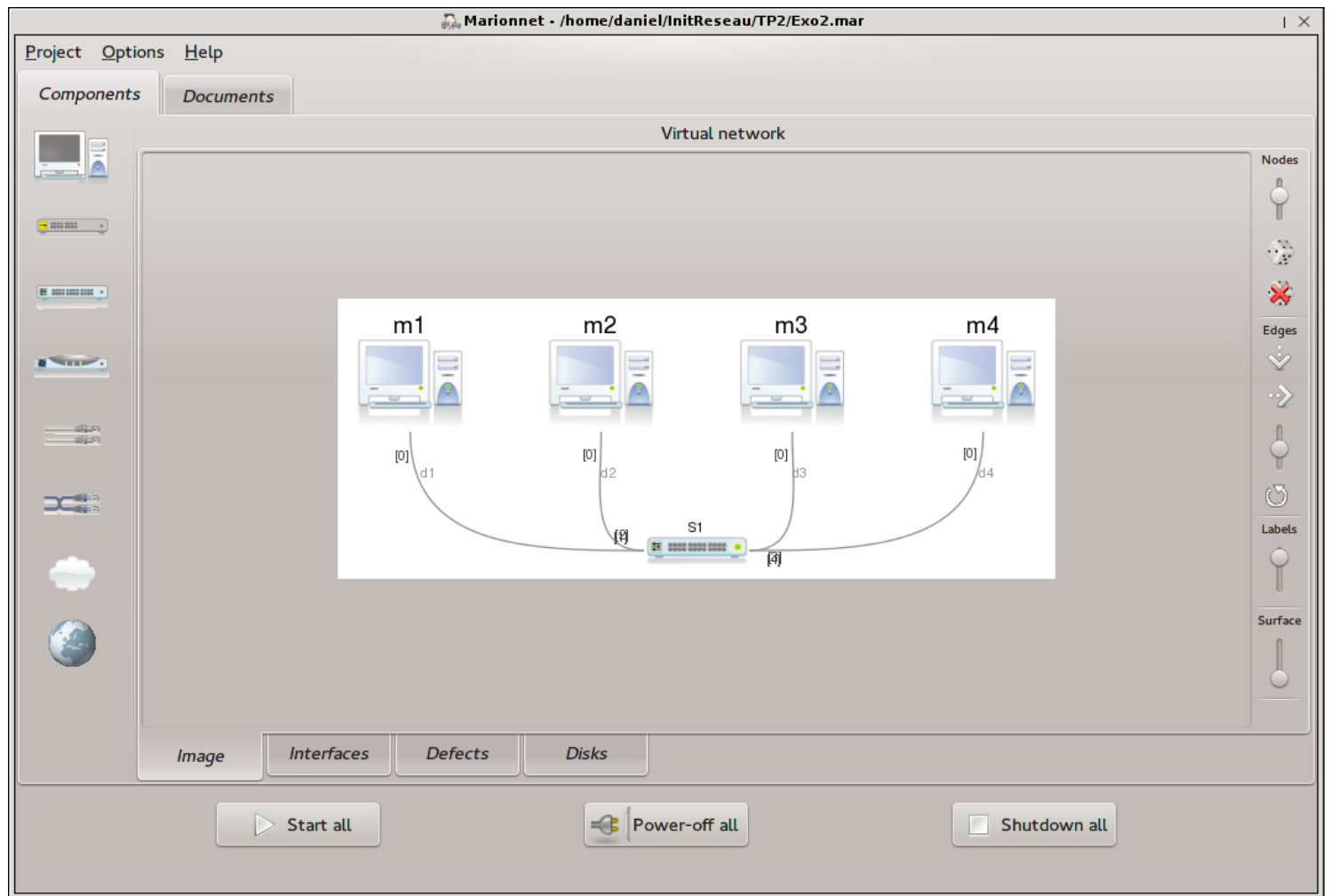


Figure 1: Aperçu du logiciel Marionnet

Cet outil a été conçu par [Jean-Vincent LODDO](#) pour les enseignements de réseau de l'IUT de Paris 13.

Il s'agit d'un [logiciel libre](#) : il peut être utilisé, étudié, modifié et diffusé à condition de respecter sa licence d'utilisation ([GNU GPL](#)).

Marionnet fonctionne uniquement sur Linux, car il utilise une fonctionnalité spécifique de ce système d'exploitation pour créer des machines virtuelles ([User Mode Linux](#)).

L'installation de Marionnet sur un système Linux est non triviale : elle nécessite des connaissances avancées en administration système pour être effectuée correctement.

Si vous souhaitez utiliser Marionnet sur votre machine personnelle, nous vous conseillons l'utilisation du [DVD live](#) qui permet d'exécuter une ancienne version du logiciel sans installer quoi que ce soit sur votre machine ou d'utiliser ce logiciel via une [machine virtuelle VirtualBox](#).

Les commandes à utiliser dans les consoles

Marionnet possède une interface graphique permettant de créer divers composants du réseau (ordinateurs, switches, routeurs) et de les câbler. Les machines administrables doivent être gérées à l'aide de la console, en se connectant comme super-utilisateur (login: root, mot de passe: root).

Les commandes suivantes vous seront utiles pour les prochains TPs.

ping

Il s'agit d'une commande permettant de vérifier si une machine est atteignable sur le réseau. Elle utilise un protocole particulier au dessus de IP (ICMP) pour demander à une machine de renvoyer un paquet écho de la demande. Cette commande indique le temps écoulé entre la demande d'écho et le retour de la machine visée. Cette information peut être utile pour détecter un réseau congestionné.

```
$ ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1): 56 data bytes
64 bytes from 127.0.0.1: icmp_seq=0 ttl=64 time=0.049 ms
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.132 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.057 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.056 ms
^C
--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.049/0.073/0.132/0.034 ms
```

Remarque: Certains administrateurs réseaux configurent certaines machines (via un pare-feu) pour ne pas répondre au ping. Dans l'environnement windows le pare-feu permet de filtrer l'ICMP. Les paquets ICMP Echo Request sont ignorés (DROP) au niveau IP mais la résolution d'adresse ARP garde une trace dans le cache ARP. la commande `$arp -a` nous donne le contenu du cache. On peut aussi vider le cache ARP en cas d'empoisonnement (fausse correspondance entre @IP et @mac) Pour vider le cache ARP `$arp -d @IP` Pour modifier le cache ARP `$arp -s @IP @(Fausse adresse Mac)`

ifconfig

Nous avons utilisé `ifconfig` lors du premier TP pour récupérer l'adresse IP et l'adresse MAC d'une machine. Cette commande, lorsqu'elle est utilisée par un utilisateur "normal", est purement informative.

Cependant, utilisée par un "super utilisateur", elle permet de configurer les paramètres réseau de la machine (adresse IP, masque de sous-réseau).

On utilisera les commandes suivantes :

```
$ ifconfig eth0 192.168.27.2/24 up
```

pour associer l'adresse IP 192.168.27.2 et un masque de sous-réseau de 24 bits à l'interface eth0 et activer le réseau sur cette interface.

```
$ ifconfig eth0 down
```

pour désactiver le réseau sur l'interface eth0.

route

Sous Unix (Linux, Mac OS X), la table de routage peut être visualisée avec la commande `netstat -rn`. Sous Linux, il est aussi possible d'utiliser la commande `route`.

Pour déclarer une passerelle par défaut sur une machine, on utilisera la commande suivante :

```
route add default gw @IP
```

wireshark

Cet outil permet de visualiser les informations qui transitent sur une interface réseau donnée. / Il se lance en ligne de commande : `wireshark`

tcpdump

Cet outil permet de visualiser les informations qui transitent sur le réseau directement sur la console. Contrairement à wireshark, il peut être totalement utilisé en ligne de commande : c'est la solution la plus simple pour découvrir ce qui se passe sur le réseau.

Il se lance en ligne de commande : `tcpdump -n`

Présentation du fonctionnement de Marionnet

Marionnet vous permet de créer des composants réseau et de les connecter les uns aux autres.

Pour démarrer Marionnet, ouvrir un terminal sous Linux, se placer dans le répertoire `/tmp` (taper la commande `cd /tmp`) et lancer Marionnet (taper la commande `marionnet`).

Les ordinateurs



Figure 2: Une machine Marionnet

Il s'agit de petits systèmes linux, avec 48 Mio de RAM par défaut, et le minimum de commandes disponibles. Il est possible de lancer un navigateur web (firefox par exemple) pour vérifier un accès HTTP externe. Les ordinateurs sont disponibles en deux variantes de Linux : Debian Linux et Mandriva Linux.

Les concentrateurs (hub)



Figure 3: Un concentrateur Marionnet

Il s'agit de composants réseaux qui répètent sur tous leurs ports les signaux reçus en entrée. Ils permettent de relier plusieurs ordinateurs entre eux. Marionnet permet de définir le nombre de ports disponibles.

Si en pratique l'utilisation des concentrateurs a disparu au niveau des liaisons filaires, on retrouve ce comportement lorsque l'on utilise des bornes WiFi.

Les commutateurs (switch)



Figure 4: Un commutateur Marionnet

Il s'agit de composants réseaux intelligents capables de limiter la diffusion des trames Ethernet aux seuls ports concernés. Comme dans le cas du hub, le nombre de ports est paramétrable. En pratique, on utilise presque toujours des switches pour relier des machines entre elles dans un même sous-réseau.

Les routeurs (passerelle IP)

Il s'agit du composant réseau le plus important d'Internet : il permet de relier des sous-réseaux distincts.

Il contient comme le concentrateur et le commutateur plusieurs ports. Cependant, ces ports disposent chacun d'une interface réseau, donc d'une adresse MAC. On peut leur associer une adresse IP.

Généralement, un routeur a beaucoup moins de ports qu'un commutateur : on a besoin d'un port sur chaque sous-réseau à connecter.



Figure 5: Un routeur Marionnet

Les câbles droits



Figure 6: Un câble droit Marionnet

Il s'agit d'une représentation virtuelle des câbles Ethernet classiques avec une prises RJ45 de chaque coté. Ces câbles sont utilisés pour connecter un ordinateur à un hub ou a un switch, c'est-à-dire une prise RJ45 avec une interface réseau et une prise RJ45 sans interface réseau.

Les câbles croisés



Figure 7: Un câble croisé Marionnet

Il s'agit d'une représentation virtuelle des câbles Ethernet croisés avec une prises RJ45 de chaque coté. Ces câbles sont généralement utilisés pour connecter directement deux ordinateurs entre eux, ou un ordinateur et un routeur, c'est-à-dire deux prises RJ45 associées à une interface réseau.

Mon premier réseau local entre deux machines

Il est recommandé de créer un nouveau projet Marionnet pour chaque exercice. Le temps de démarrage des machines est de l'ordre de quelques dizaines de secondes, selon les machines. Évitez de redémarrer les machines durant un exercice si ce n'est pas nécessaire. Sauvegardez votre projet Marionnet AVANT de démarrer les machines.

Dans cet exercice, nous allons créer le plus petit réseau non trivial possible : un réseau contenant deux ordinateurs.

1. Créez deux ordinateurs m1 et m2 avec la configuration standard.
2. Démarrez les deux machines : vous devez voir apparaître pour chaque ordinateur une console. Se connecter sur chacune d'elle en mode super-utilisateur (root/root).
3. Vérifiez que sur chacune des machines vous obtenez un echo sur l'adresse IP 127.0.0.1 à l'aide de la commande `ping`.
4. Associez la machine m1 à l'adresse IP 192.168.21.18/28 et m2 à l'adresse 192.168.21.19/28 à l'aide de la commande `ifconfig`.
5. Vérifiez que sur chacune des machines vous obtenez un echo sur leur adresse respective à l'aide de la commande `ping`.
6. Essayez de contacter une machine à partir de l'autre machine. Que se passe-t-il ?
7. Connectez les deux machines à l'aide d'un câble droit. Essayez de contacter une machine à partir de l'autre machine. Que se passe-t-il ?
8. Remplacer le câble droit par un câble croisé. Essayez de contacter une machine à partir de l'autre machine. Que se passe-t-il ?

Au delà de deux machines

Lorsque vous devez connecter plus de deux machines entre elles, il faut utiliser un équipement réseau supplémentaire : un concentrateur ou un commutateur. Le but de cet exercice est de mettre en évidence des différences de fonctionnement entre ces deux équipements.

1. Créez trois ordinateurs m1, m2 et m3 avec la configuration standard.
2. Ajoutez un concentrateur (hub) à 4 ports sur votre réseau.
3. Connectez les trois ordinateurs au hub à l'aide de câbles droits.
4. Démarrez l'ensemble des équipements.
5. Associez les adresses IP suivantes aux équipements : 192.168.21.18/28 pour m1, 192.168.21.19/28 pour m2, 192.168.21.20/28 pour m3.
6. Lancez `tcpdump -n` sur la machine m3. Surveillez son unique interface `eth0`.
7. Lancez la commande `ping 192.168.21.18` sur m1. Qu'observez-vous sur la console de m3 ?
8. Lancez la commande `ping 192.168.21.19` sur m1. Qu'observez-vous sur la console de m3 ?
9. Lancez la commande `ping 192.168.21.20` sur m1. Qu'observez-vous sur la console de m3 ?
10. Éteignez le concentrateur. Ajoutez un commutateur. Déplacer les câbles droits du concentrateur au commutateur.
11. Allumez le commutateur.
12. Reprenez les manipulations précédentes : quelle différence de comportement observez-vous ?
13. Que se passe-t-il si vous remplacez un câble droit par un câble croisé ?

La limite des concentrateurs

Dans cet exercice, vous devez créer 6 machines et les relier à un même concentrateur. Pour éviter de vous connecter sur chaque machine pour fixer son adresse IP, **celle-ci peut être directement fournie dans l'onglet Interfaces de Marionnet avant de démarrer les machines.**

1. Créez 6 machines m1 à m6 et leur donner une adresse IP du réseau 192.168.2.0/24
2. Créez un concentrateur h1 de 6 ports et y connecter les 6 machines.
3. Allumez les machines m1 et m2.
4. Allumez le concentrateur
5. Lancez une commande `ping` sur m1 pour vérifier l'accès à m2. Notez le temps de réponse de m2.
6. Démarrez les quatre autres machines.
7. Lancez une commande `ping` sur m3 et m5 pour vérifier l'accès à m4 et m6 respectivement.
8. Qu'observez-vous ?

Importance des masques de réseau

Jusqu'à présent, toutes les machines étaient situées sur le même sous-réseau. Que se passe-t-il si deux machines sur deux réseaux différents essaient de communiquer.

1. Créez deux ordinateurs m1 et m2 d'adresses IP 192.168.1.2/24 et 192.168.2.2/24. Notez que ces deux machines sont sur des réseaux différents.
2. Reliez ces deux machines par un câble croisé.
3. Démarrez ces machines, et essayez d'atteindre la machine m1 à partir de la machine m2. Qu'observez-vous ?
4. Pour relier deux réseaux IP différents, il nous faut une passerelle IP, c'est-à-dire un routeur. Créez un routeur ayant comme adresse IP de port 0 192.168.1.1/24.
5. A l'aide de l'onglet Interfaces, associez l'adresse IP 192.168.2.1/24 au port 1.
6. Reliez les deux ordinateurs au routeur **à l'aide de câbles croisés** : m1 doit être branché sur le port 0 et m2 doit être branché sur le port 1.
7. Vérifiez que vous pouvez atteindre les interfaces du routeur dans les deux réseaux locaux (ping sur les adresses IP des interfaces sur le même réseau).
8. Essayez d'atteindre la machine m1 à partir de la machine m2. Qu'observez-vous ?
9. Il est nécessaire d'indiquer sur les machines m1 et m2 qu'une passerelle existe sur le réseau. Lancez les commandes `route add default gw 192.168.1.1` sur m1 et `route add default gw 192.168.2.1` sur m2 pour cela.
10. Essayez d'atteindre la machine m1 à partir de la machine m2. Qu'observez-vous ?

Switch et masques de sous-réseau

Que se passe-t-il si des machines ne se trouvant pas sur le même réseau sont connectées à un commutateur ?

1. Créez quatre ordinateurs m1, m2, m3 et m4 avec la configuration standard.
2. Ajoutez un commutateur (switch) à 4 ports sur votre réseau.
3. Connectez les quatre ordinateurs au commutateur à l'aide de câbles droits.
4. Démarrez l'ensemble des équipements.

5. Associez les adresses IP suivantes aux équipements : 192.168.0.2/24 pour m1, 192.168.0.3/24 pour m2, 192.168.1.2/24 pour m3, 192.168.1.3/24 pour m4.
6. Vérifiez que les machines d'un même sous-réseau se voient.
7. Que se passe-t-il quand une machine du réseau 192.168.0.0/24 cherche à contacter une machine sur le réseau 192.168.1.0/24 ?
8. En vous inspirant des instructions de l'exercice précédent, ajoutez un routeur pour relier ces deux réseaux. **Attention, un routeur est vu comme un ordinateur par un switch, il faut donc les relier avec un câble droit, pas un câble croisé comme dans l'exercice précédent.**