

# M1 Informatique Réseaux et systèmes

## TP1 – TP2

Gaétan Richard  
gaetan.richard@unicaen.fr

09/2017

## 1 Introduction

L'objectif de ce TP est de se familiariser avec l'application **marionnet** que nous allons utiliser durant les TPs, d'effectuer une configuration réseau de base ainsi que des manipulations sur le système.

**Attention :** L'ensemble des TPs formera un tout qui sera évalué à la fin. Il vous est demandé de prendre, à côté, des notes au fur et à mesure et d'être capable de refaire (rapidement) la totalité des configurations. Comme vous travaillerez en *root*, le risque de perdre la totalité du projet existe.

## 2 Premier réseau

Le lancement de **marionnet** s'effectue depuis le terminal.

Il faut ensuite créer un nouveau projet pour pouvoir configurer un réseau virtuel.

Le premier réseau minimum à configurer ressemblera à celui de la figure 1 :

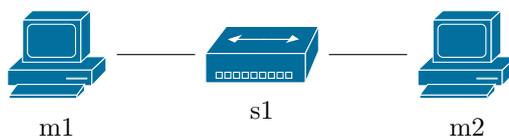


FIGURE 1 – Un embryon de réseau

*s1* est un switch, les liens sont des cordons droits, *m1* et *m2* des machines “standards”.

Les machines sont des *ubuntu* d'un âge certain. Dans un premier temps, créer un compte utilisateur avec la commande **useradd**. Dans une pratique normale, cet utilisateur devrait être systématiquement utilisé et les commandes nécessitant les droits de super-utilisateur devraient passer par un **sudo**.

Si on souhaite faire communiquer *m1* et *m2*, il faut leur fournir des configurations IP minimales. Éditer (par exemple avec **nano** ou **vim**), le fichier `/etc/network/interfaces`. Cette dernière solution aura l'avantage de mémoriser les adresses ... si vous pensez à sauvegarder une nouvelle version de votre projet ! N.B. : vous devrez néanmoins, avec cette solution, (re)lancer les services réseaux au moyen de la commande **service networking (re)start**.

Vérifiez, d'une machine, l'accessibilité de l'autre en tapant : **ping <adresse de l'autre machine>**

Nous allons maintenant tester la création de clefs pour *ssh*. Créer une bi-clef RSA sur *m1* et mettez la clef publique au bon endroit sur *m2* pour pouvoir faire un **ssh** sans avoir à taper un mot de passe.

## 3 Deux réseaux reliés par une passerelle

Nous allons maintenant reprendre une autre architecture. Vous devez pour cette section configurer un réseau ressemblant à celui de la figure 2. Cette architecture servira de base aux séances suivantes. L'accès à l'extérieur (nuage) sera mis en place par un *bridge*.

*m5* sera une machine avec trois interfaces Ethernet.

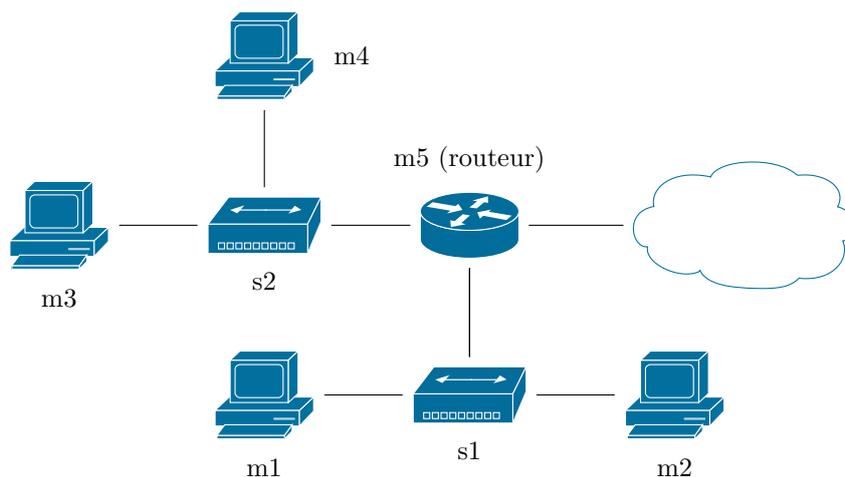


FIGURE 2 – Deux réseaux

Configurez en IPv4 chaque interface de chaque machine, pensez à donner des adresses réseaux différentes aux deux mini-réseaux ( $m1-m2-m5$ ) et ( $m3-m4-m5$ ).

Sur  $m5$ , pensez à activer le forwarding (autrement dit la fonction de routage) : En IPv4 à l'aide de `/proc/sys/net/ipv4/ip_forward` ou plus durablement en modifiant le fichier `/etc/sysctl.conf` (ceci sera valable aussi pour IPv6).

Testez la communication entre :

- $m5$  et les autres machines
- une machine du groupe ( $m1-m2$ ) et une machine du groupe ( $m3-m4$ )

Notez les succès et les échecs et expliquez. La visualisation des tables de routage au moyen de la commande `route -n` peut vous aider. Que faut-il ajouter pour que la communication soit complète ? Faites-le (un `man` de `route` peut être utile).

Configurez l'interface de cette nouvelle machine tournée vers l'extérieur de manière automatique (dhcp au lieu de static). En relançant les services réseaux, vous devez obtenir automatiquement une adresse dans le réseau `192.168.128.0/24`.

Essayez ensuite de faire communiquer cette machine avec celle de votre camarade d'en face (demandez-lui son adresse). Vérifiez également si vos deux adresses MAC sont bien différentes !

Configurer vos routes pour faire communiquer toute machine de votre réseau avec toute machine du sien. Essayer de rendre cette configuration pérenne (on pourra utiliser l'option `post-up` du fichier `/etc/network/interfaces`).

Pour simplifier la configuration, nous allons mettre en place une configuration `dhcp` sur le réseau  $m3-m4$ . Mettez en place un serveur dhcp sur  $m5$  et configurez les machines  $m3$  et  $m4$  en conséquence.

## Administration système

Pour cette partie (et cette partie seulement) nous allons reprendre une simple machine connectée à l'internet et faire un peu d'administration.

Dans un premier temps, trouver comment déterminer la version d'ubuntu utilisée ainsi que le noyau linux.

Regarder les outils de gestions `apt-get` et `aptitude`. À quoi sert la commande `moo` d'`apt-get`. Cette fonctionnalité est-elle présente sur `aptitude`.

Faire une mise à jour de la base de donnée (`update`), on demande de ne pas mettre à jour les paquets (ce qui consommerait inutilement du temps et de l'espace). Si vous avez néanmoins fait un `upgrade`, trouver un site pour apprendre à lire les consignes et consulter le.

Normalement, le test précédent ne devrait pas (ou mal) fonctionner. Ceci est du à notre architecture dans laquelle les adresses IPv6 d'interconnexion n'est pas routée. Un contournement consiste à mettre

à la main l'adresse du serveur de mise à jour dans le fichier `/etc/hosts` ou à forcer **apt-get** à utiliser l'IPv4.

Les deux outils précédent s'appuient sur l'outil bas-niveau **dpkg**. Quel est leur apport supplémentaire ?

Trouver un moyen de déterminer l'ensemble des paquets installés sur la machine, de rechercher l'existence d'un paquet.

Installer le paquet *fortunes-fr* et en tester le bon fonctionnement.