

TP 1 — Réalisation et configuration de base d'un réseau

Le TP est à faire en binôme (si impossible, par groupe de 3). Chaque binôme rendra à l'enseignant un compte-rendu dans lequel seront notées les réponses aux questions, les observations éventuelles ainsi que les commandes exécutées en recopiant, si nécessaire, les sorties de ces commandes (ce que la commande affiche dans le terminal).

L'objectif de ce TP est de créer un petit réseau puis réaliser quelques opérations de configuration et de test : attribution d'adresses IP, configuration des interfaces, test d'accessibilité entre machines, mesures de débit, ... Le TP montre également comment utiliser SSH (Secure Shell) pour se connecter à distance sur une machine et copier des fichiers entre machines.

Quelques indications avant de commencer :

- Vous utiliserez l'image `mageia4`. Vous vous connecterez en utilisant le compte suivant :
identifiant : `etudiant`
mot de passe : `etudiant`
- Pensez à consulter le manuel (`man <nom-commande>`) ou Internet pour avoir une description plus complète des commandes utilisées.
- La plupart des commandes que nous utiliserons ne peuvent être exécutées que par l'utilisateur `root` (l'administrateur du système). De même, les fichiers de configuration que nous modifierons. La commande `su -` (pour *switch user*) permet d'ouvrir une session sous l'identité `root`. Le mot de passe de l'utilisateur `root` est `iutparis13`.
- Dans tout le TP on travaillera uniquement sur l'interface réseau connectée à la baie de brassage de la salle de TP. C'est normalement l'interface `eth1`. L'autre interface, `eth0`, est celle connectée au réseau de l'IUT et qui vous permet d'accéder à Internet.

Exercice 1 — Câblage des machines

Prendre un switch et y connecter les trois câbles Ethernet sortant de la baie de brassage et reliés aux trois machines du groupe.

Exercice 2 — Conception du réseau

Les trois machines du groupe reliées au switch via leurs interfaces `eth1` formeront un réseau IP.

- Q. 2.1** *Calcul du masque* On souhaite que le masque de ce réseau soit le plus long possible (c'est-à-dire contenant le plus de bits à 1) en laissant pour le `host-id` juste assez de bits pour coder les identifiants des trois machines du réseau. Quel sera alors le masque du réseau ?
- Q. 2.2** *Choix de l'adresse de réseau* Choisir une adresse pour le réseau de la forme `10.N.0.X` où `N` est un numéro attribué au groupe par l'enseignant et `X` un octet que vous choisirez différent de 0. Justifier le choix de la valeur de `X`. Quelle est l'adresse de diffusion de votre réseau ?
- Q. 2.3** *Choix des adresses IP* Choisir trois adresses IP pour les machines du réseau. Justifier ces choix.

Exercice 3 — Configuration des interfaces

Il y (au moins) deux façons de configurer une interface réseau : soit à l'aide de la commande `ifconfig`, soit en modifiant les fichiers de configuration des interfaces puis en utilisant les commandes `ifdown` et `ifup`. Nous utiliserons cette deuxième possibilité.

Cet exercice est à réaliser sur les trois machines du groupe.

- Q. 3.1** *Fichier de configuration* Supprimer le fichier de configuration actuel de l'interface `eth1` puis écrire un nouveau fichier afin que :
- l'adresse IP et le masque associés à cette interface soient ceux choisis à l'exercice précédent ;
 - l'interface soit activée au démarrage de l'ordinateur.
- Donner le contenu du fichier créé pour une des trois machines. Justifier le choix de la valeur du paramètre `BOOTPROTO`.
- Q. 3.2** *Activation de l'interface* Désactiver puis réactiver l'interface à l'aide des commandes `ifdown` et `ifup`. Vérifier ensuite à l'aide de la commande `ifconfig` que l'interface a bien été configurée.
- Q. 3.3** *Test de la connexion* Pour indiquer qu'une machine accepte de répondre aux demandes d'écho envoyées par la commande `ping`, il est nécessaire que les fichiers

- /proc/sys/net/ipv4/icmp_echo_ignore_all et
- /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts

contiennent tous les deux la valeur 0 (s'ils contiennent une valeur différente de 0 les demandes d'écho seront ignorées).

Utiliser la commande `echo` combinée avec une redirection pour que ces deux fichiers contiennent 0. À l'aide de la commande `ping`, vérifier ensuite que chaque machine peut être jointe par les autres machines du groupe.

- Q. 3.4** *Options de la commande ping* Effectuer les tests suivants pour essayer les différentes options de ping :
- (a) Envoi d'une demande d'écho aux deux autres machines simultanément (sans préciser les adresses des destinataires une à une).
 - (b) Envoi de 10 demandes d'écho.
 - (c) Envoi de 5 demandes d'écho à une fréquence de 1 paquet toutes les 2 secondes.

Exercice 4 — Connexion et envoi de fichiers à distance

Dans cet exercice, nous allons utiliser les commandes `ssh` et `scp` qui permettent de se connecter à distance sur une autre machine et de copier des fichiers d'une machine à l'autre. Une machine du trinôme doit jouer le rôle du serveur. C'est sur celle-ci que les deux autres machines (les clients) se connecteront et copieront des fichiers.

- Q. 4.1** *Démarrage du service sshd* Pour pouvoir utiliser `ssh` et `scp` il faut que le service `sshd` qui traite les connexions des clients soit lancé sur la machine serveur. Démarrer ce service puis vérifier qu'il est bien lancé.
- Q. 4.2** *Visualisation des processus* Lancer, toujours sur la machine serveur, la commande `ps aux`. Celle-ci affiche la liste des processus en cours d'exécution avec leurs pids, leurs consommations en mémoire, leurs heures de lancement, . . . Retrouver la ligne correspondant au processus `sshd`. Quel est son pid ? Dans quel état est le processus ? Pourquoi ?
- Q. 4.3** *Copie à distance* Sur les machines clientes, créer un fichier vide ou contenant une suite de caractères quelconque. Copier ensuite, sous l'identité étudiant, ce fichier dans le répertoire `/tmp` de la machine serveur. Vérifier ensuite, sur la machine serveur, que ce fichier a bien été créé.
- Q. 4.4** *Connexion à distance* Depuis les machines clientes, ouvrir une connexion `ssh` sur le serveur sous l'identité étudiant. Quel est le répertoire courant à l'ouverture de la session sur la machine serveur ? Effectuer ensuite ces deux tests dans la session ouverte par `ssh` :
- (a) Utiliser la commande `wall` suivie d'une phrase quelconque (ex : `wall salut à toi`) puis vérifier dans le terminal de la machine serveur que cette phrase a bien été affichée.
 - (b) Créer un fichier quelconque dans le répertoire `/tmp` de la machine serveur. Vérifier ensuite dans le terminal de la machine serveur que le fichier a bien été créé.
- Q. 4.5** *Affichage des connexions ouvertes* Proposer une commande combinant `ps aux` avec un tube afin de n'afficher que les lignes contenant le mot `sshd`. Exécuter cette commande sur la machine serveur. Pour chaque connexion ouverte, on devrait voir un processus dont la commande est `sshd: étudiant@pts/XXX`. Ce processus est le terminal ouvert chez le client. Proposer puis exécuter une commande permettant d'interrompre une des deux connexions ouvertes. Vérifier sur la machine cliente correspondant que la connexion a bien été interrompue.
- Q. 4.6** *Fermeture de la session SSH* Fermer la session `ssh` ouverte sur l'autre machine cliente en utilisant la commande `exit` ou en tapant `Ctrl+D`.

Exercice 5 — Mesure de performances

Nous allons mesurer les temps de transfert de données sur notre réseau. Cet exercice est à réaliser sur deux machines du groupe : la machine cliente (l'émetteur des données) et la machine serveur (le récepteur).

- Q. 5.1** *Débit de la liaison* À l'aide de la commande `ethtool` (décrite en annexes, page 3), déterminer le débit de l'interface `eth1`.
- Q. 5.2** *Temps de transfert théorique* Avec le débit observé quel est le temps théorique de transfert de 100 Mo de données entre deux machines du réseau ?
- Q. 5.3** *Temps de transfert effectif* Utiliser la commande `iperf` (décrite en annexes, page 3) pour mesurer le temps de transfert de 100 Mo entre deux machines du groupe. Comparer au temps théorique calculé à la question précédente. Comment expliquer la différence entre ces deux temps ?
- Q. 5.4** *Changement du mode de transmission* Utiliser `ethtool` pour passer la liaison au switch en half-duplex (à la fois sur le client et sur le serveur). Reprendre le test de la question précédente. Quel temps observe-t-on ? Lancer la commande `ifconfig`. Quelle information affichée par celle-ci permet d'expliquer la différence ?

Annexe 1 — Mesurer le débit/temps de transfert : iperf

La commande `iperf` est utilisée pour mesurer les performances d'un réseau. Elle permet de mesurer le temps de transfert entre deux machines. Une des deux machines doit jouer le rôle de serveur. Elle recevra les données envoyées par la deuxième machine qui jouera le rôle de client.

Sur la machine serveur, il faut d'abord démarrer `iperf` en lui passant l'option `-s` pour passer en mode serveur. `iperf` attend alors des connexions de clients et affiche un message à chaque fois qu'un client se connecte.

Sur la machine cliente, il faut ensuite appeler `iperf` avec l'option `-c <ip-du-serveur>` pour passer en mode client et préciser le serveur à contacter et avec l'option `-n <nb-d-octets>` pour préciser le nombre d'octets à transférer. Les données transférées et n'ont aucune signification.

Annexe 2 — Obtenir/modifier les caractéristiques d'une liaison : ethtool

On peut appeler la commande `ethtool` en lui passant en argument le nom d'une interface pour qu'elle affiche les caractéristiques de transmission de cette interface, comme ceci :

```
# ethtool <nom-de-l-interface >
```

Une liaison Ethernet est caractérisée par un code de la forme DCT où

- D est le débit de la liaison en Mbit/s ;
- C est la méthode de codage utilisée pour envoyer les signaux sur le câble (généralement base pour du codage en bande de base, ce qui correspond à des signaux carrés) ;
- T donne des informations sur le type de câble utilisé ainsi que sur le mode de transmission (full ou half duplex).

Par exemple :

- 10baseT-FD = débit de 10 Mbit/s, avec codage en bande de base sur de la paire torsadée en full duplex
- 10baseT-HD = même chose mais en half duplex
- 100base-FX = débit de 100 Mbit/s, avec codage en bande de base sur de la fibre optique

On peut également modifier ces caractéristiques en utilisant l'option `-s` (pour set) suivie des nouvelles caractéristiques :

```
# ethtool -s <nom-de-l-interface > <nouvelles-caracteristiques >
```

Les principales caractéristiques que l'on peut modifier sont le débit en Mbit/s (paramètre `speed <nouveau-debit-en-Mbit/s>`) et le mode (paramètre `duplex <full-ou-half>`).