

# TP1

## 1 Interfaces réseau et Adresse IP

- Observez les interfaces `eth0` et `lo` de votre machine avec `/sbin/ifconfig`. Indiquez les adresses IPv4 et IPv6 correspondantes, la taille de la partie réseau, les tailles maximales de paquets (MTU, Maximum Transmission Unit), etc. Utilisez également la commande `ip addr ls`, c'est la même chose, mais en version plus moderne.
- Pour les IPv4, utilisez `ipcalc l_adresse_ip/la_taille_du_prefixe` pour obtenir le détail des adresses appartenant au même réseau que celle de votre machine.
- D'après ses adresses IPv4 et IPv6, à quoi correspond l'interface `lo` ?
- Observez que la MTU n'est pas la même pour `lo` et pour `eth0`, pourquoi ?
- La commande `ping -4 uneIP` permet de tester la connectivité IPv4 par l'émission d'une requête echo ICMP. Vérifiez que vous êtes bien relié à la machine de votre voisin (utiliser `control-c` pour l'arrêter). De même, avec `ping6 uneIPv6` pour les adresses IPv6.

## 2 Netcat & Netstat

- Lancez `nc -l -p 12345` et laissez-le tourner. Observez (en passant par un autre terminal) dans `netstat -Ainet -Ainet6 -a` ou `ss -Ainet -a` l'apparition du service parmi les autres.
- Ajoutez à `netstat` ou `ss` l'option `-p` pour constater que c'est bien le programme `nc` qui est à l'écoute. C'est donc un mini-serveur que l'on a lancé, auquel on va maintenant se connecter.
- Dites à votre voisin de lancer `nc votremachine 12345`, pour jouer le rôle du client. Observez dans `netstat -Ainet -Ainet6` ou `ss -Ainet` la connexion établie entre client et serveur.
- Tapez des lignes d'un côté ou de l'autre, observez que c'est effectivement transmis de l'autre côté.
- Comment transmettre un fichier d'une machine à l'autre à l'aide de `nc` et des redirections de fichier bash ? (utilisez l'option `--send-only` côté émetteur)

## 3 Connexion à une machine distante avec SSH

Utilisez la commande `ssh -X une_autre_machine` pour vous connecter sur la machine du voisin.

Par exemple, dans la séquence ci-dessous l'utilisateur *toto* est connecté sur la machine *taraza* du CREMI et se connecte via SSH à la machine distante *infini1* :

```
toto@taraza:~$ ssh -X infin1
Last login: Wed Feb 12 10:48:13 2020 from 2001:660:6101:800:252::4
toto@infini1:~$
```

Lancez `xeyes`, sur quelle machine s'exécute-t-il ? Utilisez `ps` au pour vérifier. Lancez la commande `w` pour afficher les utilisateurs connectés sur cette machine.

Observez avec `echo $DISPLAY` que la variable `DISPLAY` permet aux programmes de savoir s'ils doivent produire leur affichage sur l'écran (:0), ou via ssh (:10.0, redirigé par ssh vers le :0 d'origine).

Comment l'affichage se retrouve-t-il sur votre machine ?

## 4 Configuration d'un réseau local

Vous allez utiliser un environnement permettant d'émuler un réseau de machines virtuelles (ou VMs). L'environnement que nous allons utiliser est *QemuNet*. Cet environnement s'appuie entièrement sur le projet *Qemu* pour émuler des machines virtuelles et sur *VDE* pour émuler des switchs Ethernet. Les VMs que nous utilisons sont des architectures x86\_64 sous Linux Debian 8. L'intérêt principal de cette approche tient au fait que vous pouvez tester facilement des topologies réseaux diverses. Par ailleurs, il vous est possible de disposer d'un compte administrateur (*root* sous Linux) sur les VMs, sans risquer de ne rien casser !

Voici la configuration réseau utilisée pour le TP1, un réseau local (LAN) de 4 machines interconnectés via le switch Ethernet *s1*.

```

opeth grave
  \   /
   [s1]
  /   \
syl immortal
```

Afin de démarrer ce réseau virtuel au CREMI, il convient de taper la commande suivante :

```
/net/ens/qemunet/qemunet.sh -x -s /net/ens/qemunet/demo/lan0.tgz
```

Si cela ne se lance pas, vérifiez que vous n'avez pas `virtualbox` lancé (qui vient en conflit pour la virtualisation, il faut donc le fermer), et que vous le lancer bien directement sur votre machine.

Une fois le script lancé, vous disposez pour chaque VM d'un simple terminal en mode texte : c'est bien suffisant pour faire tout ce que nous voulons... Il n'est donc pas question de lancer un programme graphique ! Comme éditeur de texte, vous disposez au besoin de *nano* ou *jed*<sup>1</sup>. Notez que pour copier/coller, comme toujours sous Linux, vous pouvez

---

1. Un clone léger de Emacs qui dispose de raccourcis clavier similaires : c-x c-f (open file), c-x c-s (save file), c-x c-c (exit), ...

simplement sélectionner d'un côté, et faire un clic-milieu de l'autre côté.

- Une fois que les 4 terminaux des VM sont apparus (Attention, les 4 fenêtres peuvent être empilées) : connectez-vous en tant que root sur *immortal* (sans password).
- A l'aide de la commande `ifconfig -a` (man `ifconfig`), donnez la liste des interfaces réseaux.
- On décide de configurer l'interface `eth0` de telle sorte que la machine *immortal* possède l'adresse `192.168.0.1/24`. Quelle est l'adresse du réseau ? Quel est le masque du réseau ? Quelle est la plage d'adresse IP de ce réseau.
- Configurez *immortal* à l'aide de la commande `ifconfig eth0 192.168.0.1/24`. Configurez de manière analogue les 3 autres machines.
- Vérifiez vos configurations à l'aide de la commande `ping`. Quel est le protocole utilisé par le programme `ping` ?
- Mettez en évidence que le ping fonctionne à l'aide de la commande `tcpdump -i eth0` qui permet d'afficher tout le trafic réseau entrant et sortant d'une certaine machine (sur l'interface `eth0`).
- Essayez de pinguer une IP autre que celles configurées. On voit passer des requêtes ARP, on en reparlera plus tard.
- Essayez un ping avec l'adresse de broadcast du réseau. Que se passe-t-il ? Les requêtes sont-elles reçues par toutes les machines ? Est-ce qu'elles y répondent ? Corrigez le problème en tapant cette commande sur tous les postes :  
`sysctl net.ipv4.icmp_echo_ignore_broadcasts=0`  
Re-essayer un ping avec l'adresse de broadcast du réseau. Que constatez-vous ?
- Avec la commande `reboot`, redémarrez la machine *immortal*. Vous notez que l'interface réseau a perdu sa configuration ! Pour remédier à ce problème, il faut éditer le fichier `/etc/network/interfaces` et y donner la configuration de l'interface `eth0` : `iface eth0 inet static`. Cherchez sur Internet et dans le man (man `interfaces`) comment configurer ce fichier. N'oubliez pas de mettre une ligne `auto eth0` Notez que ce fichier est interprété seulement au démarrage de la machine, ou lorsque vous appelez explicitement le script `/etc/init.d/networking restart`. Testez votre configuration.
- Clôturez votre session *QemuNet*. Pour ce faire, vous pouvez fermer toutes les VMs proprement avec la commande `poweroff` ou plus simplement faire un `Ctrl-C` dans le terminal où vous avez démarré votre session.
- Bonus : Configurez les machines en IPv6 dans le réseau : `2001:db8::/48`. Faites-le à la fois en direct avec `ifconfig` et dans `/etc/network/interfaces`.
- Bonus : Configurez à nouveau le réseau en IPv4 en utilisant la commande `ip`, une version plus moderne et plus puissante de la commande `ifconfig`...