

L'objectif de ce TP est de vous familiariser avec la configuration réseau des ordinateurs sous Linux. Pour cela nous allons utiliser **netkit**, un environnement qui permet de réaliser des maquettes de réseau basées sur des images Linux virtualisées.

1. Un premier réseau : configuration des interfaces

Pour les questions suivantes, nous allons mettre en place un réseau tout simple avec deux machines correspondant à la figure 1. Ces machines ont une adresse IP sur le réseau 192.168.1.0/24.

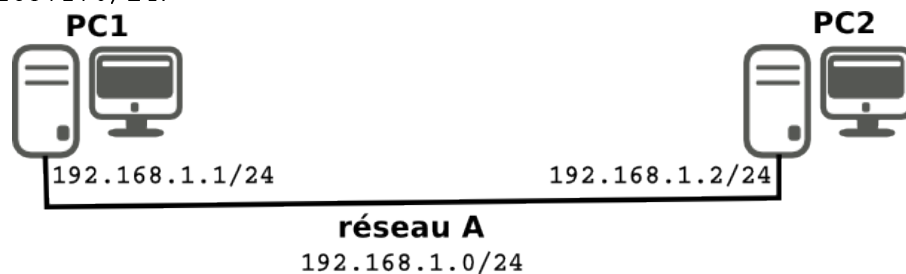


FIGURE 1 – Un premier réseau de deux machines

- (a) Nous allons maintenant démarrer nos machines, pour cela, vous aller ouvrir un terminal et exécuter la commande suivante : `vstart pc1 --eth0=A`

La commande `vstart` permet de démarrer une machine virtuelle Linux. Le nom de cette machine sera `pc1`. Cette machine aura une carte réseau nommée `eth0` (correspondant à la première, et unique, carte Ethernet). Cette carte est reliée au réseau désigné par la lettre `A` grâce à l'argument `--eth0=A`.

- (b) Démarrez maintenant la machine `pc2` en veillant à ce quelle soit reliée au même réseau. Vous devez maintenant avoir deux terminaux correspondant aux ordinateurs démarrés (figure 2). Sur chaque machine lancée vous avez accès au shell avec un compte `root`.

```

pc1
Virtual machine "pc1" booting up...
[ ok ] Setting kernel variables ...done.
[ ok ] Setting up resolvconf...done.
[ ok ] Configuring network interfaces...done.
INIT: Entering runlevel: 2
[info] Using makefile-style concurrent boot in runlevel 2.

--- Starting Netkit phase 1 init script ---
Mounting /home/petery on /hoshome...
Configuring host name...
--- Netkit phase 1 initialization terminated ---

[ ok ] Starting enhanced syslogd: rsyslogd.

--- Starting Netkit phase 2 init script ---
--- Netkit phase 2 initialization terminated ---

pc1 login: root (automatic login)
Linux rootstrap 3.2.54-netkit-ng-K3.2 #2 Fri May 23 15:45:58 CEST 2014 i686
Welcome to Netkit
root@pc1:~#

pc2
Virtual machine "pc2" booting up...
[ ok ] Setting kernel variables ...done.
[ ok ] Setting up resolvconf...done.
[ ok ] Configuring network interfaces...done.
INIT: Entering runlevel: 2
[info] Using makefile-style concurrent boot in runlevel 2.

--- Starting Netkit phase 1 init script ---
Mounting /home/petery on /hoshome...
Configuring host name...
--- Netkit phase 1 initialization terminated ---

[ ok ] Starting enhanced syslogd: rsyslogd.

--- Starting Netkit phase 2 init script ---
--- Netkit phase 2 initialization terminated ---

pc2 login: root (automatic login)
Linux rootstrap 3.2.54-netkit-ng-K3.2 #2 Fri May 23 15:45:58 CEST 2014 i686
Welcome to Netkit
root@pc2:~#
  
```

FIGURE 2 – Les deux machines virtuelles sont démarrées

- (c) **Configuration manuelle (non persistente)**

- Pour que ces deux machines puissent communiquer, il faut configurer leurs interfaces réseau. Pour cela nous allons utiliser la commande `ip` fournie par le paquet `iproute2`. Cette commande remplace la commande historique `ifconfig` que

vous pouvez trouver sur d'autres Unix¹. La commande `ip` couvre tout les éléments de la configuration réseau. De ce fait, son manuel est réparti sur plusieurs pages en fonction des fonctionnalités. Pour la configuration des interfaces, lisez la page de manuel `ip-address`. Trouvez la commande permettant de configurer l'interface `eth0` du `pc1`.

- ii. Configurez les interfaces réseau de vos deux ordinateurs, puis vérifiez votre configuration au moyen de la commande `ip address show`. Vous devez obtenir un résultat similaire à la figure 3. Vous remarquerez qu'au niveau liaison, l'interface n'est pas démarrée (`state DOWN`). Pour la mettre en route, utilisez la commande `ip link set dev eth0 up`.

```

root@pc1:~# ip address show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST> mtu 1500 qdisc pfifo_fast state DOWN qlen 1000
    link/ether 9a:97:0f:c2:1a:68 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.1/24 brd 192.168.1.255 scope global eth0
root@pc1:~#
  
```

FIGURE 3 – Consultation de la configuration des interfaces

- iii. Afin de tester la communication, on peut utiliser la commande `ping`. Celle-ci envoie des paquets à la machine destination, qui lui répond en retour. Aidez vous du manuel pour tester la communication avec cette commande. Quelle commande avez vous utilisé ?

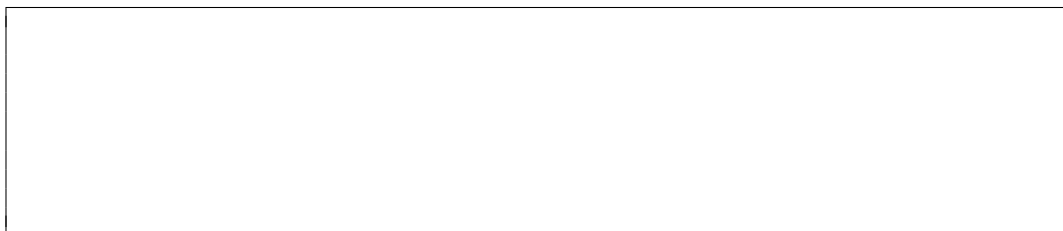
(d) Configuration persistente (fichiers de configuration)

La configuration que nous avons réalisée jusqu'à maintenant fonctionne, mais il sera nécessaire de la refaire à chaque démarrage de la machine. Afin de la rendre permanente, nous allons aborder les fichiers de configuration des interfaces réseau. La configuration permanente est basée sur le fichier `/etc/network/interfaces`.

1. Plus généralement, le paquet `iproute2` fournit des remplacements pour toutes les commandes du paquet `net-tools`

Celui-ci est utilisé au démarrage de la machine ainsi que lors de l'activation manuelle via la commande de haut niveau `ifup`. Cette commande, ainsi que la commande `ifdown` sont fournies par le paquet `ifupdown`.

- i. Arrêtez vos deux machines (commande `halt`) puis relancez les. Vérifiez que votre configuration de réseau a disparu.
- ii. Sur chaque machine, modifiez le fichier `/etc/network/interfaces` afin de définir la configuration de l'interface `eth0`. Pour cela, aidez vous de la page suivante et du manuel pour les plus courageux².



- iii. Activez votre interface sur chaque machine (commande `ifup`)³. Puis vérifiez qu'elle possède bien une adresse IP correcte.
- iv. Vérifiez la communication à l'aide de la commande `ping`.

2. Un second réseau : le routage

Pour cette deuxième partie nous allons utiliser une infrastructure composée de deux réseaux reliés par un matériel intermédiaire (figure 4).

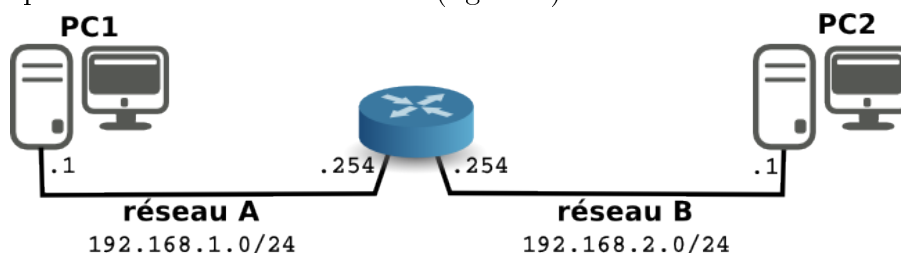


FIGURE 4 – Une infrastructure avec deux réseaux

- (a) Plutôt que de lancer les différentes machines individuellement, nous allons utiliser le *lab netkit* disponible sur Moodle. Pour utiliser ce lab, placez vous dans le répertoire `reseau2` après l'avoir décompressé. Utilisez ensuite la commande `lstart` pour démarrer toutes les machines de l'infrastructure.
- (b) Vous pouvez maintenant configurer les interfaces de `pc1` et `pc2` avec la commande `ip` en fonction du schéma figure 4 (vous devrez combiner l'adresse de réseau et le numéro de machine indiqués sur le schéma). Suite à cela, vérifiez que sur chaque pc vous pouvez joindre la machine `rtr`.
- (c) Essayez maintenant de *pinguer* `pc2` depuis `pc1`. Que se passe t-il ? Pourquoi ? Dans le doute essayez de *pinguez* l'adresse `192.168.2.254` de la machine `rtr`. Le résultat est-il différent ?

2. `man interfaces`

3. vous pouvez ignorer le message d'erreur relatif à `resolv.conf`.

- (d) Pour consulter les réseaux connus de `pc1`, utilisez la commande `ip route show`. Ce que vous obtenez est la table de routage de la machine, c'est à dire les destinations qu'elle connaît et comment les atteindre. Quel est son contenu ?

- (e) Consultez la page de manuel `ip-route` puis configurez `pc1` de manière à ce qu'il envoie les paquets à destination du réseau `192.168.2.0/24` à l'adresse `192.168.1.254` (la passerelle ou routeur de sortie du réseau local). Quelle commande avez-vous utilisée ?

- (f) affichez à nouveau la table de routage pour vérifier votre travail.
- (g) *Pinguez* maintenant l'adresse `192.168.2.254` de la machine `rtr`. Cela fonctionne-t-il ?
- (h) *Pinguez* `pc2`. Cela fonctionne-t-il ? Pourquoi ? Quelle est la différence avec `rtr` ?

- (i) Complétez votre configuration de manière à ce que les `ping` fonctionnent depuis `pc1` et `pc2`.
- (j) Arrêtez toutes les machines avec la commande `lcrash`. Relancez le lab et refaites une configuration permanente fonctionnelle (via `/etc/network/interfaces`).

3. A l'issue de ce TP :

- ☐ je sais utiliser la commande bas niveau `ip` pour activer les interfaces réseau et leur attribuer une adresse IP.
- ☐ je sais configurer de manière permanente les interfaces réseau via le fichier `/etc/network/interfaces` et les (dés)activer via `ifup/ifdown`.
- ☐ j'ai une première expérience de la table de routage et je sais la consulter via la commande `ip`.
- ☐ je sais fixer la passerelle par défaut qui me permet d'atteindre des machines hors du réseau local ou définir une route spécifique.