Travaux Pratiques
Routage statique IPv4

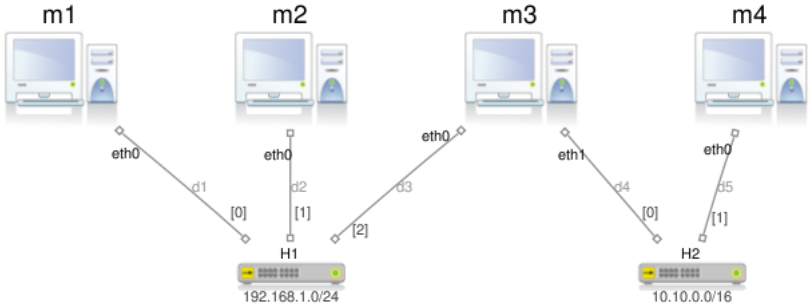
Séance de TP entièrement effectuée avec le logiciel Marionnet. Durée estimée : 1h30 - 2h.

Prérequis. Notions de routage, tables de routage et passerelle.

1 Câblage et configuration des réseaux locaux

On utilise 4 machines, m_1, m_2, m_3 et m_4, dont une en particulier, m_3, équipée de 2 interfaces réseau $eth0$ et $eth1$. Construisez un premier réseau local $LAN_1 = \{m_1, m_2, m_3\}$ en 192.168.1.0/24, sur lequel m_3 sera branché par l'interface $eth0$. Construisez un deuxième réseau local $LAN_2 = \{m_3, m_4\}$ en 10.10.0.0/16, sur lequel m_3 sera branché par l'interface $eth1$. Les deux réseaux locaux seront réalisés par des hubs [1].

Distributions GNU/Linux. Utilisez une distribution permettant de lancer des applications graphiques (p.e. *debian-wheezy*) sur les machines "espions" m_1 et $m4$. Vous avez le choix pour les autres machines.



Attribution des IP. Par simplicité, la machine m_i aura l'adresse 192.168.1.i ou 10.10.0.i selon son réseau d'appartenance, à l'exception de m_3 qui aura, dans les deux réseaux, la dernière adresse possible.

2 Configuration du réseau étendu

La machine Linux m_3 sera utilisée comme routeur pour relier LAN_1 et LAN_2 et réaliser ainsi un réseau étendu $LAN_{12} = LAN_1 \cup LAN_2 = \{m_1, m_2, m_3, m_4\}$:

1. activez la fonctionnalité *routage* du noyau Linux sur m_3 ;
2. modifiez les tables de routage des machines du LAN_1 de façon qu'elle prennent connaissance de l'existence du LAN_2 et réciproquement ; pour ce faire, utilisez la commande `route` (cf. `man route`) :

   `route add -net` *adresse-réseau* `gw` *adresse-ip*

   où *adresse-réseau* peut être spécifié avec la notation CIDR (recommandé) ou par une syntaxe longue en utilisant le mot `netmask` :

adresse-réseau	::=	*adresse-ip/entier*	`(notation CIDR)`
	|	*adresse-ip* `netmask` *masque-réseau*	`(notation longue)`

   Dans le premier cas on écrira par exemple "`123.45.67.0/24`", dans le second on écrira par exemple "`123.45.67.0 netmask 255.255.255.0`".
   Faut-il modifier la table de routage de m_3 ? Pourquoi ?
3. testez la réussite d'un ping entre m_1 et $m4$; si cette communication fonctionne, passez au point suivant ;

1. Pour des raisons purement pédagogiques, c'est-à-dire pour avoir la possibilité d'espionner confortablement, donc d'étudier, le trafic dans ces réseaux.

4. éliminez toutes les routes ajoutées au point 2. toujours par la commande `route` :

   `route del -net` *adresse-réseau* `gw` *adresse-ip*

   (il suffit dans chaque terminal de reprendre la commande `route add -net ...` utilisée précédemment en changeant le mot `add` par `del`).

5. Faites à nouveau la configuration des routes , comme dans le point 2., mais cette fois vous définirez la passerelle comme étant celle par défaut :

   `route add default gw` *adresse-ip*

   Est-ce que cela fait une différence dans notre réseau ? Que faudrait-il faire pour observer une différence entre cette manière de configurer les machines "périphériques" et celle utilisée au point 2. ?

3 Le rôle de ICMP dans le routage IP : un cas de figure

Éteignez (proprement, sans débrancher) toutes les machines virtuelles et faites-les redémarrer pour recommencer l'exercice depuis le début [2]. Cette fois, vous commencerez par lancer en tâche de fond (`&` en fin de ligne) un nouveau terminal sur m_1 :

```
m1#  xterm &
```

Dans le nouveau terminal de m_1, essayez de faire tourner un `ping` vers m_4 :

```
m1#  ping 10.10.0.4
connect: Network is unreachable
```

vous aurez droit à un message d'erreur qui vous indique que le système ne sait pas quelle direction doit prendre ce paquet IP. Lancez à présent la commande `tcpdump -i eth0` à la fois sur m_2 pour espionner le trafic sur LAN_1, et sur m_4 à partir d'un nouveau terminal (`xterm&` depuis m_4). Vous êtes maintenant en mesure de constater, pas à pas, la progression de votre configuration. Essayez alors, dans l'ordre, les actions suivantes :

- sur m_1, configurez l'IP de l'interface $eth0$
  - testez à nouveau le ping $m_1 \rightarrow m_4$
  - observez que *rien* ne se passe sur le LAN_1 depuis l'espion m_2 : pourquoi ?
- sur m_1, configurez m_3 comme passerelle par défaut
  - testez à nouveau le ping $m_1 \rightarrow m_4$ (laissez-le boucler [3])
  - observez que *quelque chose* se passe sur le LAN_1 depuis l'espion m_2 : quoi ? quels protocoles ?
- sur m_3, configurez les deux interfaces $eth0$ et $eth1$
  - observez que *quelque chose* de différent se passe sur le LAN_1 depuis l'espion m_2 : quoi ? quels protocoles ?
- sur m_3, activez le routage
  - observez que *quelque chose* se passe
    - sur le LAN_1 depuis l'espion m_2, dans le sens de communication $m_3 \rightarrow m_1$: quoi ? quels protocoles ? S'agit-il d'un message **ICMP** de type différent de 0 (REPLY) et de 8 (ECHO) ?
    - sur le LAN_2 depuis le `tcpdump` tournant dans la deuxième fenêtre de terminal de m_4 : quoi ? quels protocoles ?
- sur m_4, configurez l'interface $eth0$
  - que y a t'il de différent par rapport à la situation précédente ? que se passe t'il sur les deux réseaux ?
- sur m_4, configurez m_3 comme passerelle par défaut
  - le ping $m_1 \rightarrow m_4$ devrait finalement fonctionner !

4 Exercices complémentaires

Modifier la topologie et/ou la configuration des systèmes de façon à pouvoir provoquer et observer (depuis un espion) :

- un message ICMP de type REDIRECTION (type 5)
- un message ICMP de type TTL EXPIRED (type 11, code 0)
- les deux en même temps (par un seul et simple `ping`)

Vérifiez avec `wireshark` quelle passerelle est suggérée par le routeur émetteur du message ("reproche") REDIRECTION.

2. L'historique des commandes sera conservée d'un démarrage à l'autre et vous pourrez donc retaper facilement toutes le commandes dans chaque terminal.
3. Ce ping ne fonctionnera pas immédiatement, mais laissez-le tourner tout de même, de façon à observez les progrès réalisés pas à pas au cours de la configuration