

TP n° 12: À la découverte du routage

Exercice 1.

Réseau et ligne de commande

Dans cet exercice, on va (re)découvrir les bases du réseau (TCP/IP) sous Linux, par la pratique. À cet effet, on va bien sûr utiliser les machines qui sont à disposition dans cette salle de TP (mais ça devrait aussi fonctionner sur vos machines persos).

En rappels théoriques, vous avez bien évidemment le cours de Michael, en particulier https://perso.ens-lyon.fr/michael.rao/ASR2/cours_slides_11.pdf et https://perso.ens-lyon.fr/michael.rao/ASR2/cours_slides_12.pdf.

1. Rappelez le nombre et le nom des différentes couches du modèle / de la suite TCP/IP. Rappelez en quelques mots à quoi sert chaque couche. Par la suite on s'intéresse plus particulièrement à la couche « Liaison » et à la couche « Réseau ».
2. La programme « ip » permet de manipuler et d'interagir avec les périphériques réseaux, les interfaces réseaux, et les routes sous Linux (entre autres). Quelle est la différence entre une interface et un périphérique réseau ? À quoi sert une route ?
3. En utilisant les commandes « ip route », « ip address » et « ip link », déterminez quelles sont les caractéristiques (interfaces, routes, périphériques, adresses IP et adresses physiques) des machines de TP. Il peut être intéressant de faire un petit schéma de ces machines pour la suite.
4. Les différentes entités interagissant au sein de la couche liaison sont identifiées par des adresses MAC (pour *Medium Access Control*), aussi nommées adresses physiques. Les différentes entités interagissant au sein de la couche réseau sont identifiées par des adresses IP. Pour faire la correspondance entre ces deux types d'adresses, un protocole et une table de correspondance sont utilisés, possédant le nom ARP (pour *Address Resolution Protocol*). Utilisez la commande « arp » (et les options adaptées) pour déterminez la correspondance entre les deux.
5. Quel format possède une adresse IPv4 ? Qu'est ce qu'un masque de sous-réseau ? Qu'est ce que la notation CIDR ? À quel intervalle d'adresses IP correspond 192.168.3.5/21 ?
6. Connectez physiquement en utilisant un câble avec deux embouts RJ45¹ deux machines. Une fois cette liaison (physique) établie, déterminez quelles étapes doivent être effectuées pour permettre une communication (par exemple avec ping) entre les deux machines. En particulier, attribuez une adresse IPv4 (avec « ip address add ... ») à chaque machine, et ajoutez une route sur chaque machine expliquant comment accéder à l'autre machine (avec « ip route add ... »).
Testez la connexion entre les deux machines avec ping.
Vérifiez que la correspondance ARP est bien mise à jour.
7. Connectez trois machines ensemble (qu'on nommera A, B et C), soit avec un switch², soit en utilisant plusieurs cartes réseaux si les machines en possèdent plusieurs.
8. Établissez une connexion bidirectionnelle au niveau IP entre les machines A et B, et entre les machines B et C (attribuez donc des adresses IP à chacune de ces machines). Vérifiez que la machine A ne peut pas communiquer avec la machine C directement (par exemple en vérifiant qu'il n'est pas possible de ping C depuis A).
9. Configurez la machine B comme une passerelle (i.e. un routeur) permettant à la machine A de communiquer avec C en passant par B, et C de communiquer avec A en passant par A.

1. On appelle ça câble ethernet à tort, car un câble n'est pas foncièrement lié au protocole avec lequel on l'utilise

2. Un switch est un équipement réseau qui inspecte l'adresses MAC de destination d'une frame, et transmet à la bonne machine sur le bon port

Pour cela, autorisez la machine B à transférer du trafic avec la commande « `sysctl -w net.ipv4.ip_forward=1` », et utilisez une commande du type « `ip route add TRUC via MACHIN` ».

Testez la connexion (et vérifiez avec Wireshark que le trafic passe bien par B), et vérifiez qu'il faut plus de temps pour pinguer C depuis A qu'il n'en faut pour pinguer B depuis A.

10. Tentez d'assigner une même adresse IP à deux machines connectées à un switch, et utilisez une troisième machine connectée à ce switch pour pinguer ces deux machines. Que se passe-t-il ?

Exercice 2.

Système Autonome

1. On veut pouvoir identifier facilement les machines du groupe une fois qu'elles seront connectées ensemble. Sur chacune des machines, lancez un serveur web qui sert une page contenant votre nom et toutes les adresses IP de la machine (de manière dynamique, sinon c'est trop simple). On utilisera judicieusement la commande `ip`.³
Vous devez donc modifier un serveur http existant (comme par exemple `https://remy.grunblatt.org/teaching/ASR2/http_server.c`) pour renvoyer les données à afficher.
2. Avec votre voisinage (groupe de tables), mettez-vous d'accord sur un /24 IPv4⁴ que vous allez utiliser ensemble. Les différents voisinages seront plus tard interconnectés. Pour éviter des collisions d'IP, on veut avoir des intervalles d'adresse IP disjoints entre groupes différents. Comment (quasi-)garantir des intervalles disjoints sans communiquer entre les groupes ?
À partir de maintenant, nous dirons que votre voisinage est un AS⁵ (Autonomous System). Trouvez un nom pour votre AS.
3. Ajoutez le nom de votre AS à votre serveur web, ainsi que son préfixe.
4. Au sein de votre AS, répartissez-vous les adresses IPv4⁶ des préfixes que vous avez choisis. Il est tout à fait possible d'avoir plusieurs adresses IP par machine.
5. Connectez les machines de votre AS ensemble en utilisant des switches.
6. Écrivez un script (Python/bash par exemple) qui teste que votre machine arrive à se connecter aux serveurs HTTP de toutes les autres machines de l'AS. Utilisez `wget/curl/python-request` et vérifiez leur code de retour.

Exercice 3.

Peering

1. Connectez votre AS avec un autre AS (appelé peer), en utilisant un seul lien. Faites en sorte qu'une machine de votre AS puisse accéder à au moins une machine de l'autre AS, et réciproquement. Ces machines jouent le rôle d'un routeur pour permettre les échanges entre l'AS locale et les autres AS.
2. Configurez vos tables de routage pour que n'importe qui dans votre AS puisse accéder à n'importe qui dans l'autre, et réciproquement. On utilisera la machine faisant office de routeur.⁷
3. Connectez-vous à au moins un autre AS (qui n'est pas un peer de l'un de vos peers), et répétez les questions précédentes.
4. Faites en sorte de pouvoir accéder aux machines qui sont dans un AS qui est un peer de l'un de vos peers, mais pas un peer direct.
5. Arrangez-vous avec les autres AS pour que les AS forment un graphe connexe. Vérifiez que toute machine peut accéder à toute autre machine.

3. Dans la réalité, il existe un protocole un peu vieux qui sert à ça : le protocole Finger décrit dans la RFC 742.

4. si vous avez fini le TP, refaites les questions avec un /64 IPv6

5. https://fr.wikipedia.org/wiki/Autonomous_System

6. et IPv6 si vous avez fini le TP

7. Il est possible de créer des routes « default », qui seront utilisées lorsque qu'aucune route plus précise ne correspond à la requête.