TP Introduction aux Réseaux

Dino Lopez Pacheco dino.lopez@univ-cotedazur.fr

1 Introduction

Dans ce TP, vous trouverez quelques exercices qui vous permettront de réaffirmer vos connaissances acquises à propos de l'architecture physique des réseaux, ainsi que les phénomènes observés lors de l'échange de paquets (messages).

2 Architecture Internet

1. Les switches sont utilisés pour fournir un accès aux clients des réseaux locaux filaires. Supposez que vous avez des commutateurs (switches) avec 5 ports chacun. Nous réserverons cependant un port pour l'accès Internet (ce qui laisse donc 4 ports pour les clients). Quelle topologie réseau construire afin de donner accès internet à 20 clients ? combien de switches vous faut-il au total ? Dessinez votre topologie.
2. Quelle profondeur doit avoir une topologie arbre pour pouvoir donner service à n clients si on possède de switches à p ports uniquement ?
3. Les routeurs sont de dispositifs qui permettent d'accéder à d'autres réseaux. De ce fait, chaque réseau doit posséder au moins 1 routeurs afin d'accéder à l'Internet. Ajoutez un routeur à votre topologie réseau de la question 1.
4. Supposez que l'architecture réseau faite dans le point 3 doit être répliqué 4 fois, car par exemple, nous avons 4 départements indépendants dans des bâtiments séparés. Si nous savons que chaque routeur doit avoir au moins 2 liaisons vers d'autres routeurs afin d'éviter les isolements en cas de panne, proposez une topologie valable pour l'interconnexions de vos 4 sites.
5. En partant du schéma entier d'interconnexion des 4 sites, identifiez les topologies réseaux qui ont été utilisés.
6. Les réseaux n'ont pas une capacité infinie. Décrivez comment un réseau à commutation de circuit et un réseau à commutation de paquets peuvent expérimenter des phénomènes de congestion.
7. Combien de serveurs au total on pourrait héberger dans un réseau k-6 fat tree ? dessinez le réseau.

3 Capture de trafic réseau

Pour debugger ou analyser un protocole ou une application repartie, il est souvent nécessaire de capturer le trafic transitant par le réseau pour s'assurer que les choses se passent comme prévu. Voici quelques exercices qui ont pour but de vous apprendre à maîtriser l'outil de capture tcpdump.

Pour les exercices de ce TP, la page web suivante peut s'avérer très utile https://docs.netgate.com/pfsense/en/latest/book/packetcapture/using-tcpdump-from-the-command-line.html

8. Dans votre machine virtuel Linux
    - Trouvez le nom de l'interface avec l'adresse IP « 10.0.X.X/24 ». Donnez aussi l'adresse MAC (adresse disponible dans la ligne « link/ether ».
    - En mode super utilisateur (e.g. en exécutant « sudo su »), exécutez tcpdump de la manière suivante « tcpdump -i *yourIface* -w /tmp/test1.1.pcap », où *yourIface* est le nom de l'interface que vous avez trouvé dans le point précédent. Laissez le programme tourner, puis, visitez un site internet quelconque (e.g. twitter.com) à l'aide de votre navigateur.
    - tcpdump est un outil pour capturer le trafic réseau (un *sniffer*). A l'aide de la man page, expliquez à quoi sert chaque option passez en paramètre.
    - Arrêtez tcpdump avec « Ctrl-C » dans le terminal, puis ouvrez le fichier pcap avec la commande « wireshark /tmp/test1.1.pcap ». Expliquez brièvement les informations disponibles dans chaque division de la fenêtre Wireshark. Vous devez faire un screenshot de votre fenêtre Wireshark et indiquer dessus les informations qu'on y retrouve.
9. Comme vous l'avez sans doute compris, tcpdump a capturé tout le trafic réseau disponible sur votre interface, ce qui rend les choses très compliqués à la lecture et peut conduire à un fichier de trace trop volumineux. Il faut donc utiliser des filtres de capture afin de mieux cibler le trafic réseau à enregistrer.

    Quelques mots clés pour écrire des filtres de capture :

    « port X » permet de capturer le trafic en provenance ou à destination du port numéro X ; « tcp » capture le trafic TCP ; « udp » le trafic UDP ; « host adresse_IP_ou_nom_de_machine » capture le trafic en provenance ou à destination de l'adresse ou nom de machine donné ; « net » capture le trafic en provenance ou à destination de l'adresse réseau spécifiée.

    Il est possible de précéder les mots clés « port », « host », « net » par les mots clés « src » et « dst » pour capturer uniquement le trafic en provenance et à destination respectivement de l'argument donné.

    Les opérateurs logiques « and » et « or » permettent de créer un filtre avec de multiples expressions, et « not » de trouver le complément d'une expression. L'utilisation de parenthèses permet également de définir l'ordre d'évaluation de filtres.

    - Expliquez à quoi sert donc la commande « tcpdump "tcp port 80" -i *yourIface* -w /tmp/test2.1.pcap », où « tcp port 80 » est en fait le filtre de capture.
    - Quel filtre de capture permet de capturer uniquement le trafic DHCP ? quel filtre utiliser pour capturer le trafic DNS ?
    - Comment capturer le trafic en provenance ou à destination de la machine www.unice.fr ?
    - Comment capturer le trafic à destination du réseau 134.59.1.0/24 ?
    - Comment capturer tout le trafic qui n'est pas de type ICMP (ICMP est le protocole qui vous permet de faire un ping –entre autres–) ?

10. Donnez un filtre pour capturer un trafic HTTP et HTTPS à destination de la machine cliente. Prouvez que ça marche correctement en déployant la topologie topo-1lan-4hosts.imn avec CORE. Copiez sur « n3 » les fichiers « cert.pem », « key.pem » et « https.py ». Donnez les permissions 644 et 600 à « cert.pem » et « key.pem » respectivement.
    - Lancez le serveur HTTPS sur « n3 » avec la commande « python3 https.py ». Le passphrase est "azerty", sans les "".
    - Lancez le serveur HTTP sur « n4 » avec la commande « python3 -m http.server 80 »
    - Ouvrez un terminal sur « n1 » et lancez tcpdump avec votre filtre de capture.
    - Ouvrez un 2ème terminal sur « n1 » et lancez firefox. Visitez le serveur HTTP puis le serveur HTTPS. La première fois que vous visiterez le serveur HTTPS, vous devez dire à firefox qu'il n'y a aucun problème de sécurité en acceptant le certificat auto-signé.
    - Fournissez une capture d'écran de wireshark montrant le trafic HTTP et HTTPS à destination de « n1 ».

4 Traffic et délai

11. Nous vous proposons maintenant d'expérimenter l'impact du trafic sur le délai observé entre 2 hosts. En réutilisant le réseau topo-1lan-4hosts.imn
    - Lancez un ping depuis « n2 » vers « n4 » (« ping *adresse_ip_de_n4* ») et laissez tourner la commande
    - déployez sur « n3 » un serveur iperf (« iperf -s -i0.2 »)
    - sur « n1 », déployez le client iperf (« iperf -c *adresse_ip_de_n3* -t15 »)
    - montrez graphiquement l'évolution du délai (vous pouvez utiliser LibreOffice Calc) et identifiez l'intervalle pendant laquelle iperf a été exécuté et expliquez vos observations.