TP Réseaux : Analyse de trames et étude de protocoles avec Wireshark

(TP sur deux séances - remise réponses et captures et QCM en fin de la 2de séance)

Objectifs

- Capture de trames à l'aide de l'outil de scrutation du réseau Wireshark
- Observer et comprendre l'encapsulation des protocoles et le modèle en couches
- Comprendre le fonctionnement de certains protocoles courants
- Apprendre à consulter les documentations techniques

Données de base

- salle réseau avec tout ce qu'il y a à l'intérieur
- machine fonctionnant sous Linux Mint sur laquelle tournera la VM Linux qui servira à lancer Wireshark
  - il faut se connecter en tant qu'utilisateur `admin`
  - si vous avez des soucis avec la VM Linux accessible depuis le bureau de la machine physique essayer d'utiliser une VM sur view-linux (sur un des 2 pools R192)
  - si vous avez quand même des soucis de VM, faites le TP sur la machine physique **en ayant prévenu au préalable votre chargé de TP**
- outil Wireshark et ponctuellement des commandes Linux nécessaires aux manipulations ci-dessous

---

Documentation

Les **liens en gras** sont accessibles depuis n'importe où. Les autres liens ne sont valables que dans les salles isolées telles que D304 et D303. Pour plus ou pour d'autres d'informations, vous pouvez toujours faire des recherches sur Internet.

- Le présent sujet, votre cours, votre bon sens et les conseils de votre chargé de TP
- Utilisation de Wireshark : `man wireshark` ou sur **le site de Wireshark**
- Exemple d'utilisation des **filtres d'affichage sous Wireshark**
- La liste exhaustive des protocoles gérés et des filtres d'affichage disponibles est accessible sur **cette page**.
- Documentation sur le protocole HTTP : **page HTTP sur Wikipedia**, documentation HTTP sur ServerRX, autre documentation, directives HTTP
- Documentation sur le protocole ARP : **page ARP sur Wikipedia**, page sur ServerRX et autre page sur ServerRX
- Documentation sur le protocole ICMP : **document pdf de C. Pain-Barre (IUT Aix-en-Provence)** ou RFC initial définissant ICMP

Dépôt final (à déposer sur moodle à la fin de votre 2de séance)

Tout au long du TP vous devez tenir à jour un fichier texte au format Markdown (voir TP1) qui va contenir à la suite et en mode texte les réponses aux questions posées et les différents éléments des manipulations (captures) du TP, vous nommerez ce fichier `TPwireshark-nomEtudiant.md`. Ces éléments sont indiqués dans le sujet par l'indication « **À inclure dans le fichier de remise** ». Le fichier doit commencer par un préambule avec le nom, prénom et groupe de l'étudiant ainsi que l'adresse IP de la VM utilisée. Un fichier canevas exemple vous est fourni sur moodle.

Pour l'édition du fichier, vous aurez obligatoirement à utiliser l'éditeur vi ou vim en ligne de commande sur la machine physique depuis la VM. Ainsi, vous pourrez faire des copier/coller de texte depuis la VM et le fichier restera en permanence sur la machine physique et ne sera pas perdu en cas de crash de la VM. Voici quelques directives à suivre obligatoirement (ces manips permettent de configurer correctement le serveur ssh sur la machine physique pour pouvoir vous y connecter depuis la VM) :

- vérifier qu'un serveur ssh est disponible sur votre machine physique avec `service ssh status`
  - si le service n'est pas installé, utiliser apt-get pour le faire
  - `sudo apt-get update`
  - `sudo apt-get install openssh-server`
- ajuster la configuration sur la machine physique (sans vous tromper dans la syntaxe)
  - s'il existe, copier le fichier `/etc/ssh/sshd_config.bak` dans `/etc/ssh/sshd_config`
  - recharger la configuration avec `service ssh reload`
- se connecter depuis votre VM avec `ssh admin@adresse_IP_machine_physique` (accepter la clé de chiffrement et saisir le mot de passe)
- dans ce même terminal de votre VM (**qui ne servira que pour l'édition compte rendu**), éditer votre fichier avec « `vim TPwireshark-nomEtudiant.md` »
  - à la fin de la 1re séance, penser à le sauvegarder sur un support ou espace personnel et à le supprimer de la machine étudiante
  - au début de la 2de séance pensez à y remettre le fichier réalisé lors de le 1re séance et le compléter pendant la séance
  - à la fin de la 2de séance, le dépôt sera ouvert jusqu'à la fin de la journée, penser à sauvegarder votre fichier sur un support ou espace personnel, le supprimer de la machine étudiante, éventuellement le compléter en dehors de la séance et le déposer avant la date limite
- dans vi ou vim, il y a deux modes :
  - le mode édition accessible en tapant '`i`' depuis le mode commande et permettant la saisie de texte et l'insertion de copier/coller (en utilisant la souris de préférence)
  - le mode commande accessible en tapant '`Esc`' depuis le mode insertion et permettant notamment la sauvegarde '`:w`', la sortie '`:q`', la recherche de texte '`/texte`' (vers l'avant) '`?texte`' (vers l'arrière)...

Le jour de **votre** 2de séance du TP Wireshark (la date et heure dépendent de votre groupe de TP), vous devez remettre sur moodle le fichier **après l'avoir converti en pdf** (pour convertir un .md en un .pdf voir TP1. Le fichier déposé doit être appelé obligatoirement **TPwireshark-nomEtudiant.pdf**

Séance 1 - travail à effectuer la 1re semaine (semaine 9) : Utilisation d'un scrutateur ou analyseur de réseau

Si vous n'avez pas le temps de finir durant la séance, vous pouvez terminer cette partie chez vous sur une VM de l'IUT accessible en Extranet (sur View Linux `https://view-linux.univ-lr.fr`). Attention ! Sur view-linux il y a beaucoup de trafic réseau, il faut donc bien rechercher le bloc correspondant à votre manip. Lors de la seconde séance, il faudra absolument vous consacrer au reste du TP. Si, au contraire, vous avez fini cette partie avant la fin de la séance, vous pouvez commencer les exercices de la séance 2.

Principe

Wireshark est un outil permettant de scruter le réseau et d'analyser les trames qui y circulent. Il donne la possibilité de capturer toutes les trames reçues par une des cartes réseaux de la machine utilisée et d'observer les échanges entre les divers protocoles. Pour lancer Wireshark sous Linux, tapez la commande « `wireshark &` » dans un terminal (il faut le faire avec les droits `root`).

L'observation se déroule en 4 étapes :

1. préparer l'expérimentation dont on veut observer l'échange dans un terminal ou via un application (telle que le navigateur Web)
2. lancer une capture dans Wireshark : menu `Capture`, puis `Start`
3. lancer l'expérimentation en validant la commande ou l'URL préparée (il faut d'abord lancer la capture est ensuite faire la manipulation dont on veut observer les échanges)
4. arrêter la capture (après quelques secondes, une fois que la manipulation testée à rendu ses résultats, bons ou mauvais)

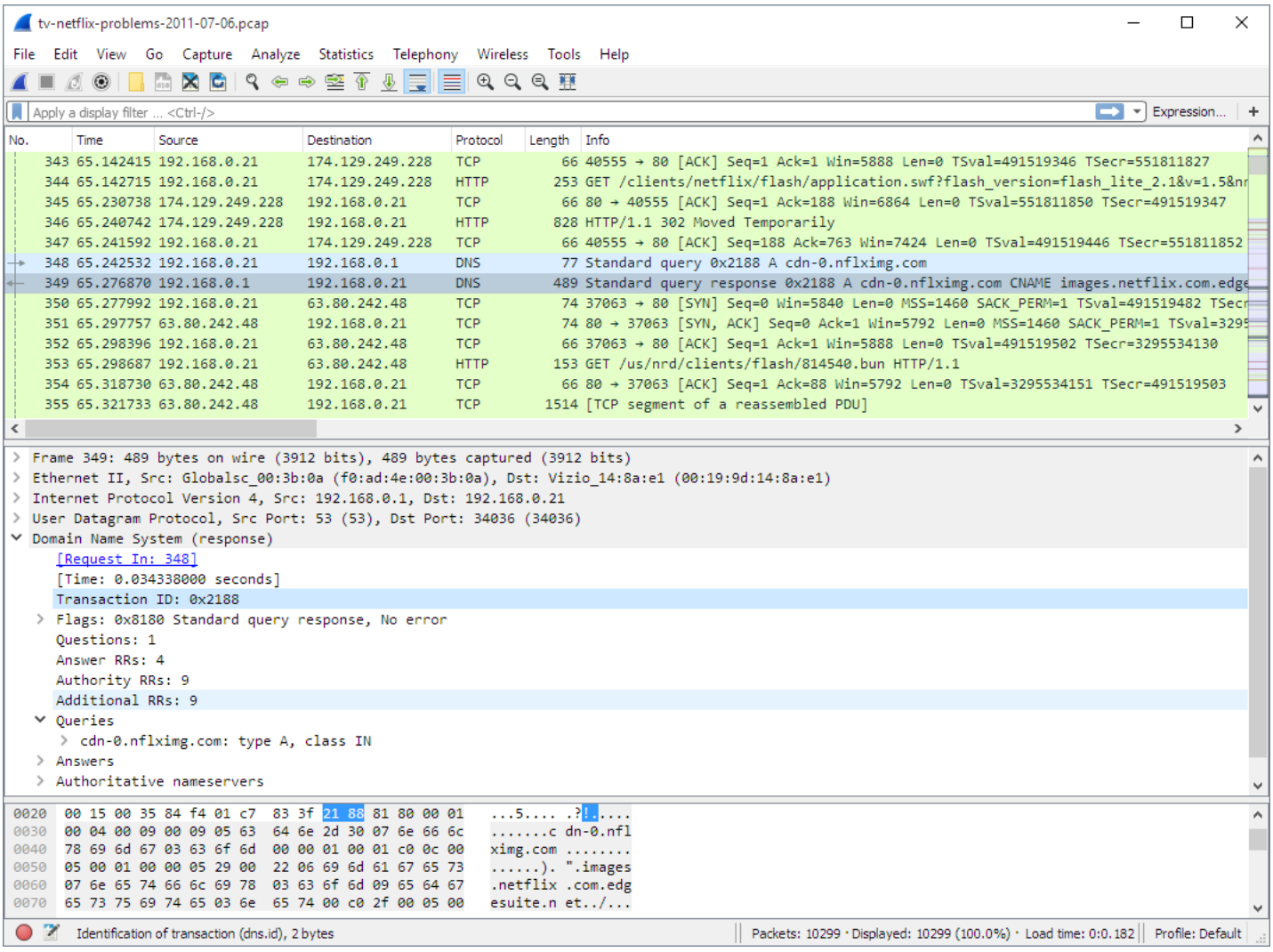
Il est possible de définir les options de la capture : menu `Capture`, puis `Options`. Il y a plusieurs options possibles lors du lancement d'une capture : choix de la carte réseau, choix du défilement en temps réel, choix du mode normal (on ne capture que ce qui nous est destiné) ou espion/promiscuous (on capture tout ce qui arrive sur la carte réseau)... Certaines options ne sont pas compatibles entre elles.

Il est possible de filtrer les trames soit durant la capture (filtre de capture ou filtre pré-capture : on ne récupère que les trames qui nous intéressent) soit après la capture (filtre d'affichage ou filtre post-capture : on capture toutes les trames mais on n'en présente qu'une sélection définie par le filtre). Dans les TP Réseaux, nous allons uniquement utiliser les filtres d'affichage post-capture. Le filtrage peut se faire suivant plusieurs critères (types de protocoles, destinataire ou source, contenu des champs...) éventuellement combinés à l'aide des opérateurs booléens. Par exemple :

- `dns` permet de ne garder que les trames ayant transporté le protocole DNS (en générale le nom du protocole en minuscule permet de filtrer sur le protocole choisi)
- `tcp.port==22` permet de ne garder que les trames dont le champ « port destination » ou « port source » de TCP est égal à 22 (ssh).
- `eth.type<=0x800` permet de ne garder que les trames dont la valeur du champ « type » est inférieure ou égale à 2048 (800 en hexadécimal)
- `http and ip.src==10.192.50.253` permet de ne garder que les trames issues de la machine dont l'@IP est 10.192.50.253 et transportant du protocole HTTP
- `!arp or !icmp` permet de garder les trames qui ne transportent ni ARP ni ICMP

La fenêtre principale de Wireshark est composée de trois parties (outre les menus, les icônes de lancement, la barre de filtre et la barre d'état en bas de la fenêtre) :

1. zone du haut : liste des trames capturées que nous appellerons « Liste »
2. zone intermédiaire : détail des champs de la trame sélectionnée que nous appellerons « Détail »
3. zone du bas : code héxadécimal de la trame capturée que nous appellerons « Contenu »



La barre de filtre, juste au dessus de la première zone, contient une zone de saisie pour le filtre post-capture. Le bouton « `Expression...` » vous aide notamment à constituer votre filtre si besoin.

Vous pouvez (et dans le cadre de ce TP devez) sauvegarder les captures au format texte quand cela vous est demandé. Il suffit d'aller dans le menu `Fichier->Exporter analyse des paquets->As Plain Text...`, sélectionner les critères de sauvegarde (paquets capturés ou juste ceux affichés, avec ou sans détails...) puis sauvegarder dans le bon répertoire. Vous pourrez ensuite copier/coller le contenu depuis cette sauvegarde vers votre compte-rendu des manips.

Pour les 2 séances, **les expérimentations doivent être lancées depuis la machine qui fait la capture à savoir la VM** (notamment, si l'expérimentation est dans un terminal, il faut que la session soit sur la VM et non un ssh sur la machine physique).

Question 1 : effectuez une capture pour l'expérimentation suivante (lisez l'ensemble de l'expérimentation avant de vous lancer) :

- capture en mode espion (*promiscuous*)
- expérimentation = `ping -c4 impll304.univ-lr.fr` :(vérifier au préalable que l'imprimante est allumée)
- Attention : si vous devez relancer l'expérimentation pour une raison ou une autre vous n'allez pas observer la même chose et notamment à cause des caches ARP et DNS. En cas d'un 2e test, faites le ping sur la machine `pandora.univ-lr.fr` ou une machine indiquée par votre chargé de TP
- arrêter la capture après la fin de l'affichage sur le terminal du résumé des temps de réponse du ping.

1. Analyse globale
   a. **À inclure dans le fichier de remise :** donnez la liste des 20 premières trames affichées (ou moins s'il y en a moins) sans détails des protocoles encapsulés ni le contenu hexadécimal (utilisez la sauvegarde en mode texte de Wireshark).
   b. quels sont les protocoles du niveau le plus haut capturés (protocoles affichés automatiquement par Wireshark dans la colonne Protocol) ? Le protocole du niveau haut dans une trame est celui qui est le plus encapsulé (dernière branche dans la zone Détail)
   c. quel est le protocole commun utilisé dans toutes les trames au niveau de la couche liaison de données (niveau le plus bas capturé) ? Le protocole du niveau bas est le protocole qui a généré la trame capturée (sur un même tronçon du réseau ce protocole est le même).

2. Filtrage
   a. avec le champ `eth.addr` filtrez les trames capturées pour ne garder que les trames issues de ou destinées à votre machine. Quel filtre avez vous utilisé ?
   b. **À inclure dans le fichier de remise :** donnez la liste des trames affichées après filtrage sans détails des protocoles encapsulés ni le contenu hexadécimal (utilisez la sauvegarde en mode texte de Wireshark).
   c. quels sont les 3 protocoles qui restent affichés et quel est le rôle de chacun d'eux ? Si vous en avez plus que 3 protocoles prévenez votre chargé de TP. Si vous avez moins que trois alors ce n'est pas la première fois que vous faites le test et quoi qu'il en soit vous devez en avoir au moins 1.

3. Récupération d'information
   a. donnez l'adresse MAC et l'adresse IP de votre machine et dites comment vous les avez trouvées dans votre capture
   b. donnez l'adresse MAC et l'adresse IP de la machine `impll304.univ-lr.fr` et dites comment vous les avez trouvées dans votre capture
   c. **À inclure dans le fichier de remise :** donnez les détails complets de la trame ayant transporté le premier paquet ICMP ainsi que son contenu en hexa.
   d. à votre avis pourquoi le champ de contrôle d'erreur du niveau de la couche liaison de données n'est pas présenté par Wireshark ?

Question 2 : Avant tout,

- modifiez les préférences d'analyse du protocole HTTP (menu `Éditer->Préférences...->Protocols->HTTP`) en décochant les 2 premiers boutons (`Reassemble HTTP headers/bodies...`)
- du protocole TCP (menu `Éditer->Préférences...->Protocols->TCP`) en décochant le bouton `Analyze TCP sequence numbers`
- exécutez `sudo ethtool -K enp0s3 gro off` si VM salle D304 ou D303, ou `sudo ethtool -K ens160 gro off` si VM view-linux

Effectuez la capture pour l'expérimentation suivante :

- capture en mode normal (*non promiscuous*)
- expérimentation = chargement de la page `http://serverrx.univ-lr.fr/`. Si la page était déjà chargée actualisez l'affichage ou videz le cache au préalable.
- arrêter vite la capture après le chargement de la page.

1. Analyse globale
   a. quels sont les protocoles du niveau le plus haut capturés ?
   b. quel est le protocole utilisé pour envoyer la requête et recevoir la réponse entre le navigateur et le serveur
   c. quel est le protocole qui a encapsulé le protocole précédent (la requête de votre navigateur et la réponse du serveur Web) ?
   d. quel est le filtre qui permet d'isoler les trames transportant le protocole de la question précédente ?
   e. quelles sont les adresses MAC et IP du serveur Web ?
   f. **À inclure dans le fichier de remise :** donnez la liste des trames affichées après filtrage sans détails des protocoles encapsulés ni le contenu hexadécimal (utilisez la sauvegarde en mode texte de Wireshark).

2. Analyse fine
   a. combien de trames ont transporté la réponse du serveur ?
   b. quelle est la taille de la page Web récupérée et dites comment vous l'obtenez ?
   c. quelle est la requête envoyée au serveur Web ?
   d. quelles sont les 3 premières directives de la requête Web et quels sont leur rôles respectifs ?
   e. **À inclure dans le fichier de remise :** donnez le détail des protocoles encapsulés dans la première trame de la réponse du serveur (utilisez la sauvegarde en mode texte de Wireshark).

Séance 2 - travail à effectuer la 2de semaine (semaine 10 ou avant si vous avez terminé le travail de la séance 1) : étude de protocoles courants

Vous devez absolument vous consacrer aux exercices suivants, même si vous n'avez pas fini la partie précédente. Vous pouvez commencer cette partie chez vous sur une VM de l'IUT accessible en Extranet (sur View Linux `https://view-linux.univ-lr.fr`). Attention, dans ce cas les captures, même courtes, récupèrent énormément de trames, il faut donc rechercher le bloc correspondant à vos expérimentations.

Question 3 : protocole ARP

1. À quoi sert le protocole ARP et comment marche-t-il (en 2-3 lignes) ?
   - vous pouvez voir le contenu d'une table ARP avec la commande `arp -a` et vous pouvez supprimer des entrées de cette table avec `arp -d`. Sans paramètre, la commande vous donne l'aide et l'utilisation de la commande.
2. Faites l'expérimentation suivante :
   a. regardez le contenu du cache ARP de votre machine (commande `arp`) et choisissez une machine voisine de la votre qui ne se trouve pas dans le cache (donnez son @IP). Si vous faites cette manip chez vous sur view-linux vous pouvez essayer avec serverrx, 10.192.50.253, ou l'imprimante, 10.192.50.49, celle qui n'est pas dans le cache.
   b. faites un `ping -c3` sur l'@IP de la machine repérée et capturez les échanges correspondants en mode normal (non promiscuous).
   c. filtrez sur ARP (il serait peut-être utile d'affiner le filtre pour n'afficher que les trames qui vous concernent, voir Q1.2.a). Avez vous des trames ARP ? Pourquoi ?
   d. refaites les questions b. et c. ci-dessus (sur la même machine).
3. Regardez le chronogramme d'un échange ARP (sous Wireshark aller dans `Statistiques->Graphique des flux`)
4. Notez la pile de protocoles observés de ARP
5. **À inclure dans le fichier de remise :** Donnez le contenu des champs ARP de la première trame ARP *capturée* (pensez à utiliser la sauvegarde en mode texte de Wireshark)

Question 4 : protocole ICMP

Afin que les expérimentations se passent correctement, il faut modifier le routeur par défaut de votre machine car les résultats attendus dépendent de la configuration particulière du serveur de la salle D304 . Sous Linux, il faut faire :

- `route del default` pour supprimer la route par défaut courante
- `route add default gw 10.192.50.253` pour ajouter la route par défaut vers le serveur serverrx.univ-lr.fr

1. À quoi sert le protocole ICMP (en 2-3 lignes) ?
2. Faites les expérimentations suivantes (en mode normal et non espion : pas *promiscuous*). Pour chacune d'elles donnez les champs type et code du message ICMP, le rôle du message ICMP et sa cause probable (pensez à filtrer sur ICMP). Un temps d'attente de quelques secondes est nécessaire pour les expérimentations b. c. et d. (il faut attendre d'avoir une trame ICMP). Pour l'expérimentation c., il faut sortir de ftp avant de continuer. Toutes les expérimentations ci-dessous provoquent des messages ICMP différents, si ce n'est pas le cas, prévenez votre chargé de TP.
   a. `ping -c1 192.168.56.3`
   b. `ssh 10.200.2.3`
   c. `ftp 192.168.56.17`
   d. `telnet 192.168.56.17`
   e. `ftp 10.100.1.1`
   f. `ping -c1 -t1 10.2.40.230`
3. **À inclure dans le fichier de remise :** donnez la liste des trames ICMP capturées dans cet exercice après filtrage, sans détails des protocoles encapsulés ni le contenu hexadécimal (utilisez la sauvegarde en mode texte de Wireshark).
4. Regardez le chronogramme pour l'échange a. (premier échange) ci-dessus.
5. Notez la pile de protocoles observés de ICMP

Question 5 : TELNET

1. Rappelez les 2 rôles de `telnet`. (en 2-3 lignes)
2. Effectuez une capture avec l'expérimentation suivante :
   - capture en mode normal
   - expérimentation = faire un `telnet` sur `10.192.50.253`, authentification avec nom d'utilisateur : `invite` et mot de passe : `invite`
   - listez le répertoire distant (commande `ls`)
   - déconnectez vous avec la commande `exit` et arrêtez la capture.

   a. filtrez pour n'avoir que les trames transportant le protocole TELNET. Combien de trames y a-t-il ?
   b. à partir des trames capturées, regardez comment le protocole TELNET envoie les commandes ?
   c. que pensez vous de la sécurité de TELNET ? Justifiez votre réponse en fournissant les détails nécessaires.
3. **À inclure dans le fichier de remise :** les 10 premières trames de la capture au format texte obtenue dans cet exercice (après le filtrage) qui doit contenir uniquement la liste des trames capturées et après filtrage de TELNET, sans détails et sans contenu hexadécimal.