Master 1 - Informatique
MRI (Mécanismes des Réseaux informatiques)
TP1 et 2

OUTILS D'ADMINISTRATIONS BASIQUES & ANALYSE DE TRACES (ETHERNET, IP ET HTTP) AVEC WIRESHARK[1]

N.B. : Ces travaux pratiques font l'objet d'un Compte Rendu (électronique) qui doit être envoyé avant le début de séance suivante à l'enseignant responsable de ce TP. Ce CR fera état des réponses aux questions. Vous êtes libres d'ajouter tout élément d'information qui vous semblera judicieux de donner.

Vous pouvez utiliser des ressources extérieures (code, graphiques, explications, liens vers des articles wikipedia) dans la limite du raisonnable à condition de l'indiquer clairement (en donnant le lien). En particulier vous êtes encouragé à faire référence aux normes (e.g., RFC, javadoc) lorsque vous le pouvez. Dans le cas contraire, nous considérerons qu'il s'agit de plagiat (y compris si vous vous êtes contenté de traduire).

Votre CR doit contenir les traces prouvant les résultats obtenus et/ou le bon fonctionnement de vos programmes. Tout travail non justifié sera considéré comme non fait.

Les TPs sont notés A (excellent), B (bon travail), C (travail insuffisant), 0 (travail non rendu ou plagiat). Dans le cas de TPs de programmation, vous aurez deux notes : une pour le CR et une pour les résultats.

Si vous êtes bloqués sur une question vous pouvez faire appel à l'enseignant pour avoir des éléments de réponses ou utiliser les résultats d'un autre groupe à la condition expresse de l'indiquer, l'utilisation de travaux extérieur sera pris en compte dans la note. Deux résultats ou réponses étonnamment similaires dans deux CR de groupes différents sans explication seront considérés comme du plagiat.

Rappel: « Les enseignements de TP sont obligatoires. Deux absences non justifiées à une séance de TP entraînent la note 0 pour la note de TP de la matière au sein de l'UE. »

Objectifs

L'objectif de ce TP en deux séances est :

1. de se familiariser avec des commandes de base en mode utilisateur permettant d'avoir des informations sur une infrastructure Internet;
2. de faire une analyse de l'empilement protocolaire en utilisant le cas d'une application Web (http). Nous examinerons la structure des en-têtes des PDU (Protocol Data Unit) au niveau de la couche liaison (Ethernet) et de la couche réseau IP.
3. d'aborder le protocole HTTP dans le but d'en implémenter une version simplifiée lors des TP à venir. Vous apprendrez à vous servir de netcat.

[1] Une partie de ce TP est basé sur un TP de César Viho

PARTIE 1 : INTERROGATIONS D'UNE INFRASTRUCTURE DE RESEAU

INTRODUCTION

Nous avons dans le cours rapidement abordé les protocoles qui interviennent dans une infrastructure de réseaux Internet. L'objectif de ce TP est d'aborder quelques commandes qui servent à obtenir un certain nombre d'informations sur les composants, les mécanismes, bases de données et protocoles qui participent au fonctionnement global de l'architecture Internet. Dans cette première partie du TP, il s'agit d'utiliser quelques-unes de ces commandes pour :

- identifier et vérifier l'état de certaines machines qui composent les sous-réseaux de votre établissement (l'ISTIC) et de quelques autres machines, composants, service externes, etc.
- obtenir, lorsque c'est possible, des informations utiles au fonctionnement de ces sous-réseaux

Vous devez utiliser les commandes appropriées ou afficher les contenus des fichiers qui fournissent les réponses aux questions posées. Voici une liste non exhaustive des commandes utilisables (***ifconfig***, ***netstat***, ***nslookup***, ***dig***, ***ping***, ***traceroute***, etc.). Pour ces commandes, vous devez respecter la syntaxe précise et fournir les paramètres et les éventuelles options nécessaires. Vous avez deux moyens pour obtenir la syntaxe, les paramètres et les éventuelles options d'une commande :

- le manuel en ligne via la commande ***man***
- l'option ***–h*** ou ***-?*** qui affiche la liste des paramètres et options d'une commande.

Pour ce sujet TP, on suppose que l'on travaille en environnement Linux, mais sachez que des commandes analogues existent pour les autres systèmes d'exploitation tels que Windows, MacOS, etc.

Attention !!! Certaines commandes nécessitent de les préfixer par ***/sbin*** ou ***/usr/sbin*** et d'autres peuvent être inhibées dans les salles de TP. Dans ce dernier cas, vous pourrez utiliser des outils de diagnostic en ligne (e.g., http://network-tools.com). Vous devez à chaque fois indiquer la façon dont vous avez obtenu vos résultats et adapter vos commentaires en conséquences.

TESTS ET COMPTE-RENDU

Vous fournirez des explications sur ce que vous observerez dans un compte-rendu qui sera fourni à votre enseignant de TP avant le début de la séance suivante. Ces explications devront montrer que vous avez compris le fonctionnement des éléments et protocoles intervenant dans une architecture Internet. Ce compte-rendu devra comporter des réponses à chacune des questions suivantes. Vous prendrez soin de donner les commandes utilisées et les résultats obtenus dans le compte-rendu.

Q1 Indiquez le nom complet, l'adresse IP de votre machine le masque de son sous-réseau.

Q2 Donnez les informations sur le(s) protocole(s) de niveau liaison de données utilisé(s) sur votre machine. On donnera également les autres informations que l'on peut déduire de la lecture des résultats fournis par la(les) commande(s) utilisée(s).

Q3 Donnez le plan d'adressage de la salle. Peut-on en déduire le plan d'adressage des autres salles du bâtiment ? Si oui, on indiquera comment, sinon on dira pourquoi.

Q4 Testez si les 5 machines suivantes sont actives et donnez leur nom complet :

148.60.10.15, 148.60.4.3, 148.60.12.7, 148.60.2.200 et 148.60.1.39

Q5 Donnez les numéros de ports associés à ces services : ***http, ftp, telnet, smtp*** et ***ssh***. On indiquera les moyens/commandes utilisés pour obtenir ces informations.

Q6 Donnez les principales sessions Internet en cours sur votre machine.

Q7 Donnez la table de routage de votre machine.

Q8 Donnez les serveurs DNS des machines/services suivants : votre station, yasuragi.irisa.fr, www.google.fr, www.mit.edu, www.tahi.org, www.etsi.org. On donnera également les autres informations que l'on peut déduire de la lecture des résultats obtenus.

Q9 Déterminez le chemin suivi par les paquets pour atteindre les machines/services suivants: votre station, yasuragi.irisa.fr, www.google.fr, www.mit.edu, www.tahi.org, www.etsi.org. Si vous utilisez des outils externe précisez l'origine des paquets émis.

Q10 Produire un récapitulatif court des commandes utilisées dans cette partie avec leur usage principal.

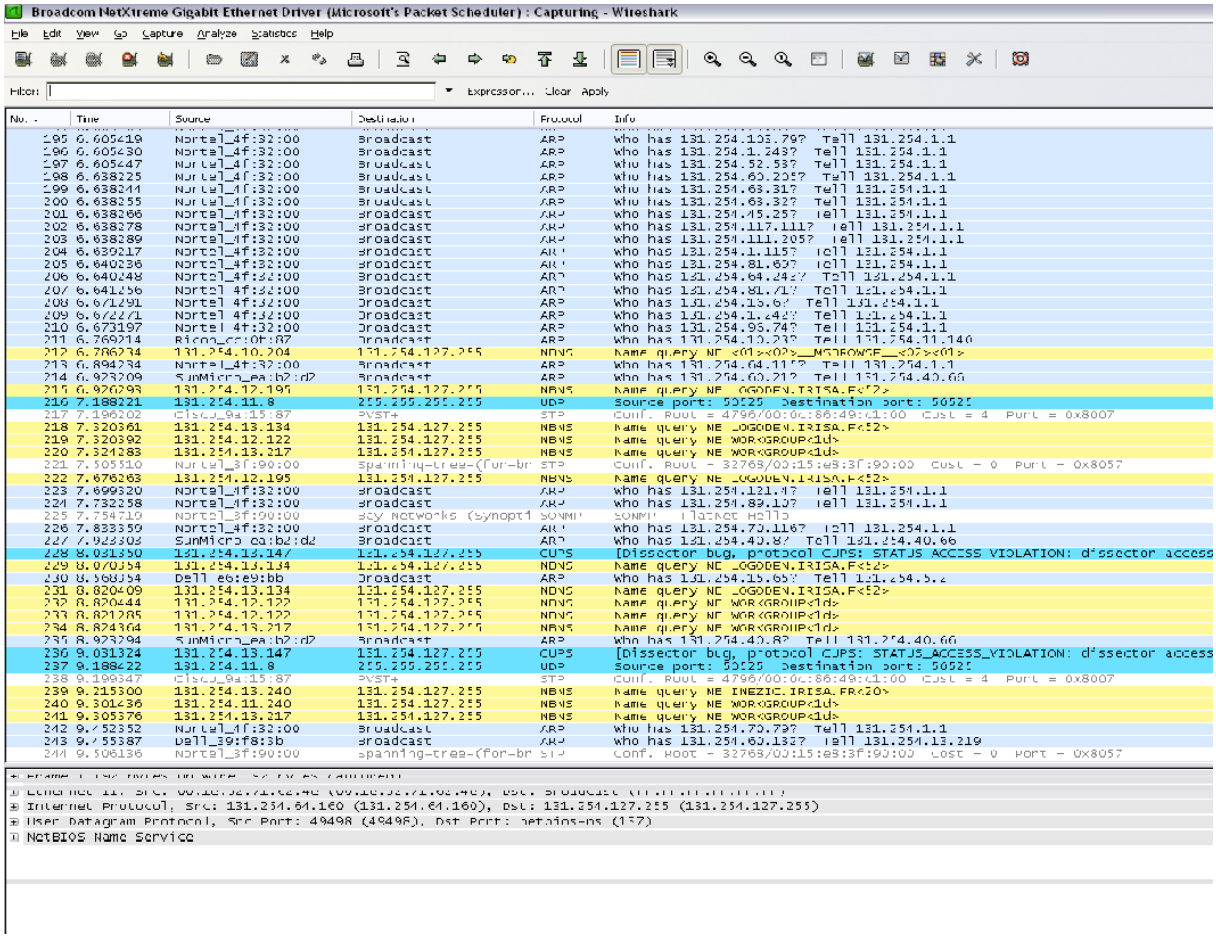
PARTIE 2 : ANALYSE DE TRACES ETHERNET ET IP AVEC WIRESHARK

INTRODUCTION

L'analyseur de trafic est un outil essentiel pour comprendre les mécanismes de fonctionnement des protocoles de communication sur les réseaux actuels. Un analyseur de trames peut récupérer (et analyser) les trames en entier ou les premiers octets de celle-ci au choix, ce qui permet de donner des indications sur les en-têtes de protocole sans récupérer l'intégralité des données, ce qui est suffisant pour analyser le réseau et les problèmes éventuels. Nous avons choisi d'utiliser Wireshark qui présente une interface graphique très conviviale et qui analyse de nombreux protocoles. Il existe d'autres analyseurs de trafic plus complexe ou en mode console que nous vous encourageons à essayer (tcpdump, tshark, ...).

INTERFACE UTILISATEUR

L'interface de l'analyseur se décompose en plusieurs barres et fenêtres



Barre de menus

On y retrouve la liste classique de menus. Voici une liste des fonctions accessibles à partir de ces menus.

- Le menu *File* sert à sauvegarder ou charger un fichier de capture réseau. Une capture peut très bien avoir été réalisée sur une sonde distante ou avec un autre outil et être analysée avec Wireshark a posteriori.
- Le menu *Capture* sert à fixer les paramètres d'une nouvelle capture réseau.

- Le menu *Statistics* sert à effectuer différents calculs sur les volumes de données et les protocoles concernés.

Barre des icônes

Cette barre regroupe tous les raccourcis sur les manipulations d'une capture.

Barre de filtrage

Cette barre sert à saisir l'expression de filtrage a posteriori d'une capture pour isoler tout ou partie d'un échange réseau.

Fenêtre contenant la liste des trames capturées

Sur chaque ligne on retrouve :

- Le numéro du paquet,
- Son temps de capture,
- Sa source,
- Sa destination,
- Le protocole de plus haut niveau décodé,
- Le résumé des champs caractéristiques de ce protocole.

Fenêtre d'affichage de la pile des protocoles décodés pour la trame sélectionnée

Avant toute opération de développement des champs d'un ou plusieurs protocoles, cette fenêtre donne la liste la pile de protocoles décodés allant du niveau physique (en haut) jusqu'au niveau le plus haut reconnu (en bas). Le protocole de niveau le plus haut reconnu est celui qui apparait dans la colonne protocole de la Fenêtre contenant la liste des trames capturées.

- La première ligne ou niveau Frame correspond à une pseudo couche physique. Comme il n'est pas possible de réaliser la capture directement à partir des composants électroniques qui pilotent l'interface réseau sans perturber le fonctionnement du système, l'opération a lieu au niveau liaison à l'aide de la bibliothèque libpcap. A ce niveau, les informations disponibles sont la quantité de bits capturés et la date de capture.
- La deuxième ligne correspond au niveau liaison. On y détaille le type et les champs de la trame et les adresses physiques.
- La troisième ligne correspond au niveau réseau. On y détaille les champs du protocole réseau reconnu : adresses logiques et indicateurs d'état.
- La quatrième ligne correspond au niveau transport. On y détaille les champs du protocole de transport reconnu : état de la connexion, numéros de ports utilisés et diverses options.
- La cinquième ligne correspond au niveau application. On y trouve les données utilisateur. Pour le développement de chacun des champs de la trame, il faut cliquer sur le triangle situé à gauche au niveau de chaque couche.

Fenêtre d'affichage brut de la trame sélectionnée

Cette fenêtre affiche tous les octets de la trame en hexadécimal.

SYNTAXE DU FILTRAGE

La granularité de la syntaxe de filtrage est très importante. Elle peut donc s'avérer très complexe à manipuler. Wireshark offre plusieurs solutions pour rendre l'apprentissage de cette syntaxe interactif. Tout d'abord, l'opération précédente de filtrage simplifié n'était qu'un cas particulier de saisie interactive de filtre de capture. Il est aussi possible d'utiliser la syntaxe du langage C pour les tests :

• == : égalité,

• != : différence,

• >= : supérieur ou égal,

• <= : inferieur ou égal.

ETUDE DES PROTOCOLES ETHERNET ET IP

Procédures

- Lancer Wireshark (Démarrer -> Programmes ->Emulateurs-> Wireshark).
- Lancer Internet Explorer ou Firefox
- Dans le menu capture de Wireshark, cliquer sur start.
- Entrer l'adresse suivante : http://www.istic.univ-rennes1.fr dans le navigateur, dès que la page est chargée, arrêter la capture dans Wireshark.
- Dans le champ filtre, tapez TCP (les paquets concernant cette connexion sont affichés en vert)

En utilisant les bonnes commandes et options de Wireshark,

Q1 Identifier l'adresse Ethernet et l'adresse IP de votre machine, et vérifier avec les réponses aux questions 1 et 2 de la partie 1.

Q2 Quel est le contenu du champ ***type*** du paquet Ethernet ?

Q3 Identifier les adresses de destination Ethernet et IP ?

Q4 Quel est l'identifiant du constructeur de la carte réseau ?

Q5 Quelle est la taille de l'en-tête du paquet IP ? Quelle est la taille totale du paquet?

Q6 Identifier le champ du type du protocole transporté par le paquet IP ?

Q7 Quelle est la valeur du champ TTL ?

Q8 Pour un des échanges entre votre ordinateur et un site web, dessinez le chronogramme d'un échange. Vous porterez attention au début et à la fin de la capture, vous pouvez résumé les paquets intermédiaires si il y'en a bcp. Assurez-vous que la connexion que vous décrivez a bien été fermée.

PARTIE 3 : ANALYSE DE PROTOCOLES DE LA COUCHE APPLICATION

Dans cette partie, il est important de justifier vos réponses par des traces. MAIS, nous vous encourageons à sélectionner les informations.

PROTOCOLE HTTP

Analyse avec wireshark

Dans cette partie, vous allez vous concentrer sur le protocole HTTP. Le protocole est décrit dans les RFC 1945, 2068 et 2616.

Connectez-vous de nouveau à http://www.istic.univ-rennes1.fr et intéressez-vous maintenant au contenu de la communication. Dans wireshark, vous pouvez avoir une vue globale des échanges avec la commande Analyse/Follow TCP stream. Elle permet d'analyser le contenu d'un échange TCP. Attention toutefois, il y'a parfois plusieurs objets chargés séparément dans des échange TCP différents. Nous vous encourageons à utiliser les filtres pour filtrer le protocole (ex "http and not udp") ou filtrer les adresses cibles et sources.

Q1 Sur quel protocole de la couche transport s'appuie le HTTP ? Quel est le modèle de communication mis en œuvre par ce protocole ? Qui prend l'initiative de la communication ? Comment sont représentées les informations ?

Q2 Essayez de vous connecter au site http://156.35.213.25 (ne dois pas pointer sur un serveur HTTP existant). Une connexion TCP est-elle initialisée (utilisez un filtre ip.dst ==156.35.213.25 si nécessaire) ?

Q3 Essayez de vous connecter au site cesitenexistepas.com. Une requête est-elle émise ? Une connexion est-elle initialisée ? Expliquez comment le navigateur fait pour déterminer l'existence du site... Sur quel protocole se base la résolution de nom? Donnez une capture.

Q4 Indiquez de façon générique dans l'ordre les étapes nécessaires à la consultation d'une page web http://nom.de.domaine.tld par le navigateur au niveau de la couche transport et des protocoles utilisées (HTTP et ...) lors de la consultation d'une nouvelle page (dont l'adresse n'est pas résolue).

Essayez de vous connecter aux sites suivants (Attention, vous veillerez à utiliser http et non https dans votre requête):

- http://example.org
- http://www.google.com/images/errors/logo_sm.gif
- http://fr.wikipedia.org/azdqsj
- http://google.fr

Q5 Quelle est la forme d'une requête ? Vous décrirez les champs qui vous paraissent important en vous appuyant sur la RFC. Vous accorderez une attention particulière à la première ligne de la requête.

Q6 Quelle est la forme de la réponse ? Vous commenterez particulièrement le cas de la requête pour google.fr ? Que se passe-t-il ? Combien de requêtes sont émises pour obtenir effectivement le contenu HTML ? Quel protocole est utilisé après la première requête ?

Q6 L'échange est-il fondamentalement différent pour les images ? Quels sont les champs importants ?

Q7 A quoi sert le champs Accept-Encoding dans la requête? Quel est le pendant à ce champs dans la réponse ? Trouvez un site qui utilise gzip et montrer une courte traces ...

Q8 A quoi sert l'encodage des caractères ? Comment est géré l'encodage en HTTP? Donnez les champs de la requête et de la réponse qui permettent d'indiquer l'encodage (donnez une trace des en-têtes).

Le cache

Q9 Examinez les requêtes émises par le navigateur et réponses fournies par le serveur dans les trois situations suivantes pour la page http://example.org :

a) clic sur le bouton *recharger*,

b) idem avec touche *majuscule* (shift) appuyée,

c) Se rendre sur google.fr puis clic sur le bouton précédent.

Quels sont les principales différences dans le comportement du navigateur et dans les requêtes ?

Q10 Dans quel cas le cache du navigateur est-il utilisé (vous pouvez vider le cache en cherchant dans les options de votre navigateur) ? Où est précisément situé le cache? Pouvez-vous lire ces fichiers (essayez about:cache dans firefox)?

Q11 Quels sont les champs importants pour la gestion du cache dans la requête et dans la réponse ?

Q12 Quel est l'intérêt du cache ? Décrivez rapidement les étapes suivies par le navigateur et par le serveur lors de l'utilisation du cache (comparaison de date, etc ...).

Les scripts

Q13 Rendez-vous sur la page http://www.random.org/integers/ et demandez la génération de nombre aléatoire. Le résultat est-il transmis dans la réponse du serveur (vous pouvez faire afficher la source de la page web par le navigateur pour le vérifier)?

Q14 Redemandez la génération de nombre aléatoire, une nouvelle requête est-elle émise par le navigateur ? Qui réalise le traitement ?

Q15 Répondez aux mêmes questions pour le site http://www.thonky.com/random-number/.

Q16 Comparer les avantages/inconvénients de l'exécution coté serveur et de l'exécution coté client.

Simulation avec netcat

Q17 Parcourez la page de manuel de l'outil netcat et telnet. Utilisez-les pour envoyer une requête HTTP sur un serveur de google.fr. Préparez votre requête dans un éditeur de texte. Vous demanderez la page racine. Quels sont les éléments de la requête HTTP absolument indispensable ?

Q18 Suivez les éventuelles redirections (en omettant le https).

Q19 Utilisez l'outil pour attendre une requête exemple sur le port 8080. Préparez votre réponse dans un éditeur de texte. Vous préparez deux réponses : 302 pour une redirection vers le site de yahoo.fr et 404 avec un message adéquat.

AUTRE PROTOCOLE

Q20 Connectez-vous au site ftp://ftp.mozilla.org/ avec un navigateur moderne. Quel est le protocole utilisé ? Sur quel port votre navigateur se connecte ?

Q21 Observez les commandes échangée entre le serveur et votre client. Pourquoi la liste des fichiers n'apparaît pas dans la requête ? Essayez de déterminer comment la liste de fichier est effectivement transmise.

Q22 En vous appuyant sur la RFC 959 de FTP, décrivez les commandes échangées entre le serveur et le client. Quel est le rôle de la commande PASV ? Essayez de comprendre son fonctionnement.

Q23 Essayez de vous connecter au site ftp://ftp.mozilla.org/ avec la commande ftp. Vous pouvez lire son manuel avec man (pensez à vous mettre en mode passif).

Q24 Essayez de simuler une connexion à mozilla via netcat (attention le serveur n'est pas patient. Il faut préparer vos réponses).

Q25 Essayez de simuler un serveur FTP sur votre machine avec netcat et de vous connecter dessus avec la commande FTP. Vous utiliserez les commandes découvertes plus tôt pour répondre. Essayer d'aller le plus loin possible dans l'échange. Si vous avez le temps vous pouvez même simuler une connexion à deux canaux.