Projet Programmation

Visualisateur de trafic réseau

Présentation

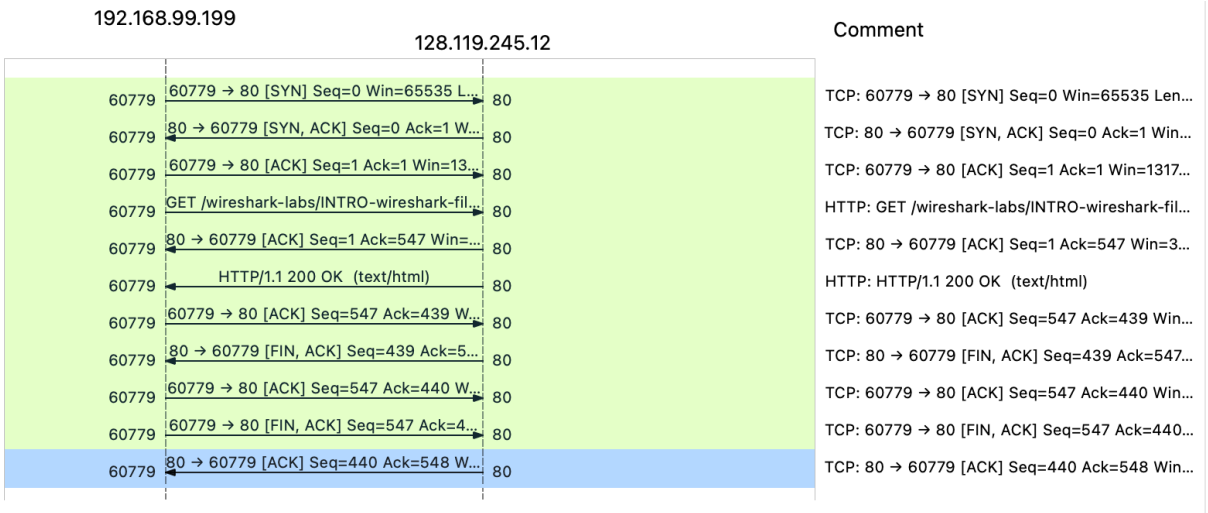
L'objectif de ce projet est de programmer un visualisateur des flux de trafic réseau. Un flux de trafic fait référence aux trames échangées dans le cadre d'un protocole exécuté à l'initiative de deux machines, chacune identifiée par une adresse MAC, une adresse IP et éventuellement par un numéro de port.

Le visualisateur prendra en entrée un fichier trace au format texte contenant les octets capturés préalablement sur un réseau Ethernet. Votre programme peut s'exécuter dans une fenêtre de commande (de type terminal) ou s'afficher dans une interface graphique.

La liste des protocoles que votre analyseur sera en mesure de comprendre sont les suivants :

- Couche 2: Ethernet
- Couche 3: IPv4
- Couche 4: TCP
- Couche 7: HTTP

Un exemple de résultat attendu est donné ci-dessous.



Le visualisateur affichera l'ensemble des trames par ordre chronologique qui correspond à leur ordre d'apparition dans le fichier trace. Pour chaque trame, le visualisateur affichera les informations suivantes :

- L'adresse IP des deux machines impliquées.
- Le numéro de port utilisé.
- Les informations pertinentes concernant le protocole de la couche la plus haute encapsulé.

A chaque exécution, le résultat du visualisateur doit être sauvegardé dans un ficher texte ou pdf formaté de façon à faciliter sa lecture.

Soumission

- Ce projet sera réalisé en binôme.
- Vous êtes libres de choisir le langage de programmation.
- Date de soumission : **Vendredi 09 décembre 23:59:00**.
- Documents à soumettre :
    1. Une **archive zip** à soumettre sur le Moodle de l'UE :
        a. votre **code source**,
        b. un **fichier binaire** ou **makefile** pour lancer l'exécution de votre analyseur,
        c. un **fichier readme** qui décrit la structure de votre code,
        d. un **fichier howto** qui explique comment installer et lancer votre programme.
    2. Une **presentation vidéo préenregistrée de 10 minutes** postée sur Youtube : Lien d'ajout de vidéo. (Votre vidéo sera ajouté à une liste de lecture privée). Ne cliquer pas sur "cette vidéo a été conçue pour les enfants".

        Dans cette vidéo, vous présenterez :

        1. un **aperçu complet** de votre projet,
        2. une description de **vos choix, réalisations et contributions personnelles**,
        3. une **démonstration** de votre visualisateur en action.

Instructions à suivre

1/ En entrée

Votre programme prend en entrée un fichier trace (format texte) contenant les octets ‘bruts’, tels que capturés sur le réseau. Ces octets sont présentés comme dans Wireshark dans le panneau ‘Octets capturés’. Ce fichier pourra contenir plusieurs trames Ethernet ordonnées par ordre chronologique (sans préambule ni champ FCS) :

- Chaque octet est codé par deux chiffres hexadécimaux.
- Chaque octet est délimité par un espace.
- Chaque ligne commence par l’offset du premier octet situé à la suite sur la même ligne. L’offset décrit la positon de cet octet dans la trace.
- Chaque nouvelle trame commence avec un offset de 0 et l’offset est séparé de trois espaces des octets capturés situés à la suite.
- L’offset est codé sur quatre chiffres hexadécimaux.
- Les caractères hexadécimaux peuvent être des majuscules ou minuscules.

2/ En sortie

Le résultat de votre programme doit être similaire aux informations produites par Wireshark dans l’outil 'Flow Graph’.

Le visualisateur affichera l’ensemble des trames par ordre chronologique qui correspond à leur ordre d’apparition dans le fichier trace. Pour chaque trame, le visualisateur affichera les informations suivantes :

- L’adresse IP des deux machines impliquées.
- Le numéro de port qu’elles utilisent.
- Les informations que vous jugerez pertinentes concernant le protocole de la couche la plus haute encapsulé. Ces informations comprendront entre autres le type de message, les numéros de séquence ou d’identification ainsi que les valeurs des champs d’entête significatifs.

Le visualisateur offrira un ensemble de filtres pour visualiser un des flots réseaux en sélectionnant les adresses IP des machines à l’origine d’un flot et/ou d’un protocole en particulier.

A chaque exécution, le résultat du visualisateur sera sauvegardé dans un ficher texte ou pdf formaté de façon à faciliter sa lecture.