Bessa
Alexandre
21306128
alexandrebessa26@gmail.com

Jacquette
Pierrick
21305551
pierrick.jacquette@gmail.com

Rapport de projet
Deep inspection packet

1) Comment utiliser notre programme ?

Tout d'abord il vous faut libpcac, pour l'installer lancer la commande :

```
sudo apt-get install libpcap-dev
```

Pour lancer le programme il vous suffit de commencer par le compiler avec le makefile puis de lancer telnet :

- make
- ./telnet [nom_du_fichier] [-v]

Le nom du fichier est le nom de la trace a analyser, le -v est une option et donc n'est pas obligatoire
Par défaut l'affichage est en décimal, avec l'option -v on affiche la commande exacte

2) Objectif

Nous devons réaliser une analyse de trafic de réseau du protocole telnet
Le programme doit extraire les différents champs du protocoles et les afficher en sortie.

L'entrée est un fichier pcap contenant les paquets à analyser, ces paquets peuvent être des paquets telnet ou non. En cas de paquets ne faisant pas partie de telnet seront ignorés.

La sortie est l'affichage du contenu des différents champs du protocoles.

3) Comment nous avons fait ?

Le programme a été développé en C.
Tout d'abord nous avons commencé par consulter la RFC de Telnet, puis la documentation de la librairie "libpcap".
Cela nous a permis de comprendre les différentes commande utilisable dans Telnet. Et aussi les différentes possibilités d'utilisation de la librairie.
Ensuite nous avons consulté wireshark avec différents fichier pcap trouvé sur internet pour pouvoir visualiser les données et les différents champs.
Enfin nous avons aussi chercher des exemples de "deep packet inspection" pour pouvoir s'en inspirer et le modifier pour notre utilisation.

Nous avons d'abord commencé avec les commandes :

- WILL
- DO
- DON'T
- WON'T

Une fois ces commandes traités, nous avons décidé de nous attaquer à la dernière commande, celle que nous avons trouvé la plus complexe : SB

4) Architecture

Il y a 8 entités (fichier .c et son .h) un fichier main et un fichier permettant de définir les structures et les includes (core.h)

- Annexe — lire, verifier une sous-commande et l'afficher
- Paquet — récupère, parse et appelle l'affichage
- Pcap — ouvre le fichier, filtre et appelle la fonction
- Tcp — affiche les différents codes tcp
- Telnet — affiche les différents code telnet
- Telnet_message — affiche les messages correspondant aux commandes
- Telnet_option — affiche les messages correspondant aux sous-commandes
- Telnet_sb — gère l'option sb

5) Documentation

La Documentation est générée via Doxygen est est disponible ici :
https://www.dropbox.com/sh/fowou31kdfi8dnp/AAAWWaw_FQ3sOjEqvt5Kf1Kva?dl=1
Vous pouvez l'avoir sous forme d'un site internet en allant dans le dossier html puis index.html

6) Exemple

L'exemple pour montrer que le programme fonctionne correctement est sur le fichier telnet2.pcap
Les captures d'écran sont dans le dossier screenshot