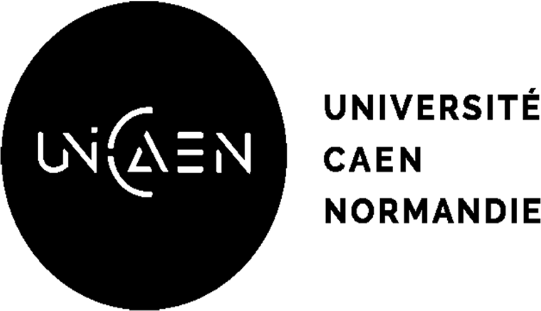
****

**Dm Systéme**

Fait par :

- **Kermezian Axel:22008897**

- **Ihannouba Nihal:22009099**

- **Krimi ibrahim:22011592**

**1-Synthèse du protocole FTP :**

ftp (File TRANSFERT Protocol)

Un protocole qui sert à servir l’envoi d’une fichier entre un client et un serveur (chargement et téléchargement ), deux port sont nécessaire a l’utilisation du cette protocole

Un canal de control : pour le transfert de commande et l’authentification , port 21

Un canal pour transfère des donnes ,Port 20

Il y a deux mode de transfère de donnes Mode actif mode passif

Pour le mode actif , le canal de donnes sera normalement défini sur le canal 20

Pour le mode passif il sera sur un port aléatoire

En mode actif le client connecte d’abord d’un port aléatoire ou le port 21 sur le serveur via le canal de commande , il indique au serveur a quel port client il doit se connecter pour le transfert de données ensuite le serveur se connecte au port auquel le client lui a dit

Enmode passif le client se connecte toujours au port 21 ou d’un port aléatoire

Mais la seule différence est que le serveur ftp sera celui qui dira au client quel port aléatoire est ouvert pour le transfert de donnes

Autrement dit en mode actif le serveur initie la connexion et en mode passif le client qui initie la connexion , le protocole ftp est parmi les protocole de transfère le plus connus mais il a un grand inconvénient ce que FTP n'a pas été conçu pour être un protocole sécurisé.et aussi Le cryptage des données ne se fait pas en FTP. File Transfer Protocol s’exécute dans la couche application de la pile des protocoles TCP/IP. Il est ainsi sur la même couche que le http.

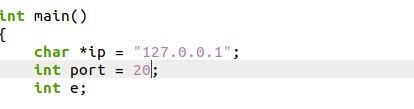
**2-Interprétation de codes et tests :**

1-Récupérez un ou 2 gros fichiers pour les tests genre 1G.

## 

La taille de la fichier file3.txt égale a 1G.

2-Lisez le code et les explications données

3-Compilez le code et modifiez (changez le port !) ou si un serveur FTP tourne sur la VM tuez le.

Changement de port

Une image contenant texte

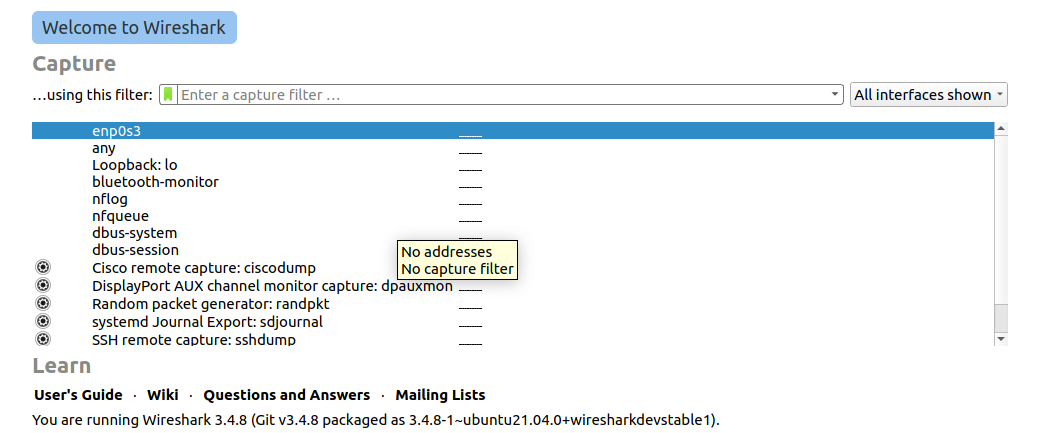
Description générée automatiquementCompilation de Code

Une image contenant texte

Description générée automatiquement

4. Lancez wireshark et sélectionnez l’interface adéquat plus les bons paramètres

configuration au premier temps on doit sélectionner l’interface adéquat au ftp et tcp

 Apres on doit cliquer sur la bouton de Capture

ICI LA CAPTURE DE FTP PAS ENCORE !!!( Toujours le tcp qui s affiche)

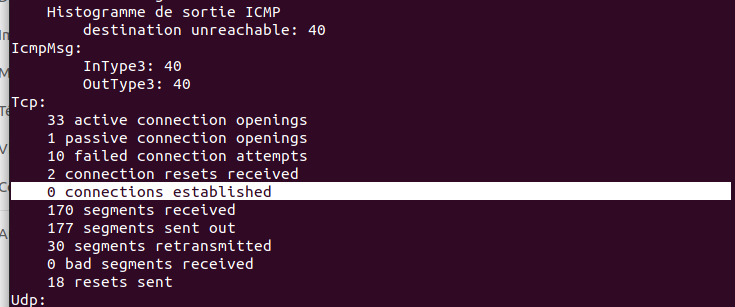
5. Lancez le serveur et expliquez ce qui se passe, Utilisez la commande netstat avec les bons paramètres.

D’après la capture il y a 1 connexion opening (1 ouvertures de connexion passives)

Ce celle de la serveur , et il y a 0 connexions établies ( c normal parceque on a pas lancer le

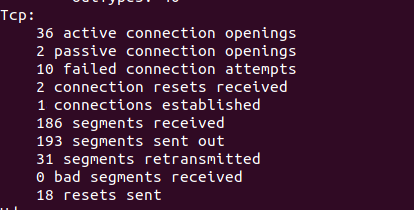
Client)

On peut voir que le serveur a crier la socket pour la transfère de ficher et aussi il est entrain de écouter sur le canal de connexion ( plutôt il est en train d’attendre le client ).



6. lancez le client et expliquez ce qui se passe.

la socket de le client a été criée avec succès et la connexion entre le serveur et le client est établi

La ficher file.txt a été bien envoyer vers le client et à la fin le serveur et le client a déconnecte de la canal de connexion.

7- Complétez le serveur de manière qu’il devienne un processus Daemon quand on le lance (vous pouvez vous inspirer du code du 1er TP sur les processus) Expliquez en “gros” les différentes lignes ajoutées.

la fonction log\_message serre a afficher un msg ( on l utilise dans la fonction signal\_handler)

la fonction signal\_handler c est pour traite les signal (catch , ignore ou handle un signal)

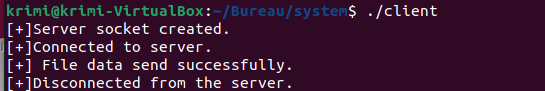
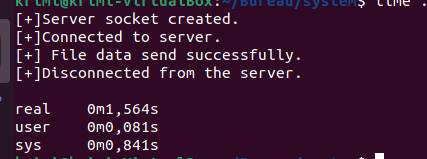
la fonction daemonize() : la fonction daemonize serre a criée le processus daemon , on cree la processus fils avec l’appelle fork() ,au premier temps on test si le processus est déjà daemon (getppid()==1) apres on crée la processus fils avec l’appelle fork(), si le proccesus n été pas criée correctement (i<0) alors on sort sinon (i>0) on commence les traitement de ftp ,

apres que on obtenir un nouveau groupe de processus , on ferme tous les descripteurs et on définit les autorisations de fichier nouvellement créées ,on change le répertoire en cours d'exécution, âpres on teste si il sont verrouiller ou pas , si il son pas verrouiller alors la première instance continue, on enregistrer le pid dans le fichier de verrouillage , on ignorer l'enfant SIGCHILD, on ignorer les signaux SIGTSTP , après on attrape le signal de raccrochage et a la fin on attrape le signal d'arrêt .

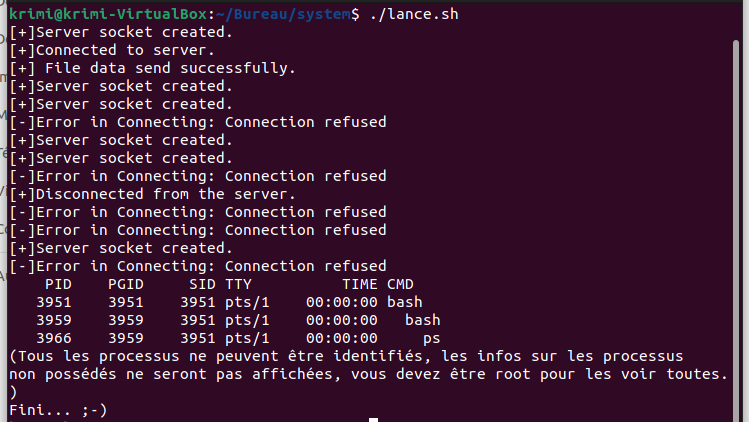
8- Lancez le client et vérifiez que tout se passe bien. Mesurez le temps d’exécution à l’aide de la commande time (voir le code du serveur parallèle du TP socket)

Une image contenant texte

Description générée automatiquement

$

9- Lancez plusieurs clients et répondez à ces questions (vous pouvez écrire un script shell) comme celui du TP socket.

 Résultat :

1. Il y a juste le premier client qui il a la bonne affiche sinon le reste de client n’as pas la bonne affichage.
2. A faire facile

10. Complétez le serveur de manière qu’il devienne un serveur parallèle

A faire diffecile

11. Compilez le et lancez le

A faire facile

12. Lancez 3 clients et vérifiez que tout se passe bien. Mesurez le temps d’exécution à l’aide de la commande time (voir le code du serveur parallèle du TP socket) (vous pouvez écrire un script shell comme celui du TP socket)

A faire facile

13. Comparez le temps mesuré avec le temps mesuré précédemment

A faire facile

14. Concluez

A faire facile

**3-Conclusion : Conclusion 5 ligne sur sockets, port, ftp, protocol et daemon**