	Module : Système d'Exploitation et Programmation Système <i>1ère Année Cycle Ingénieur</i> Mohamed BAKHOUYA, Abdelhak KHARBOUCH Projets	Année Universitaire 2019/2020
---	---	---

Fait par : EL HANAFI MAHA

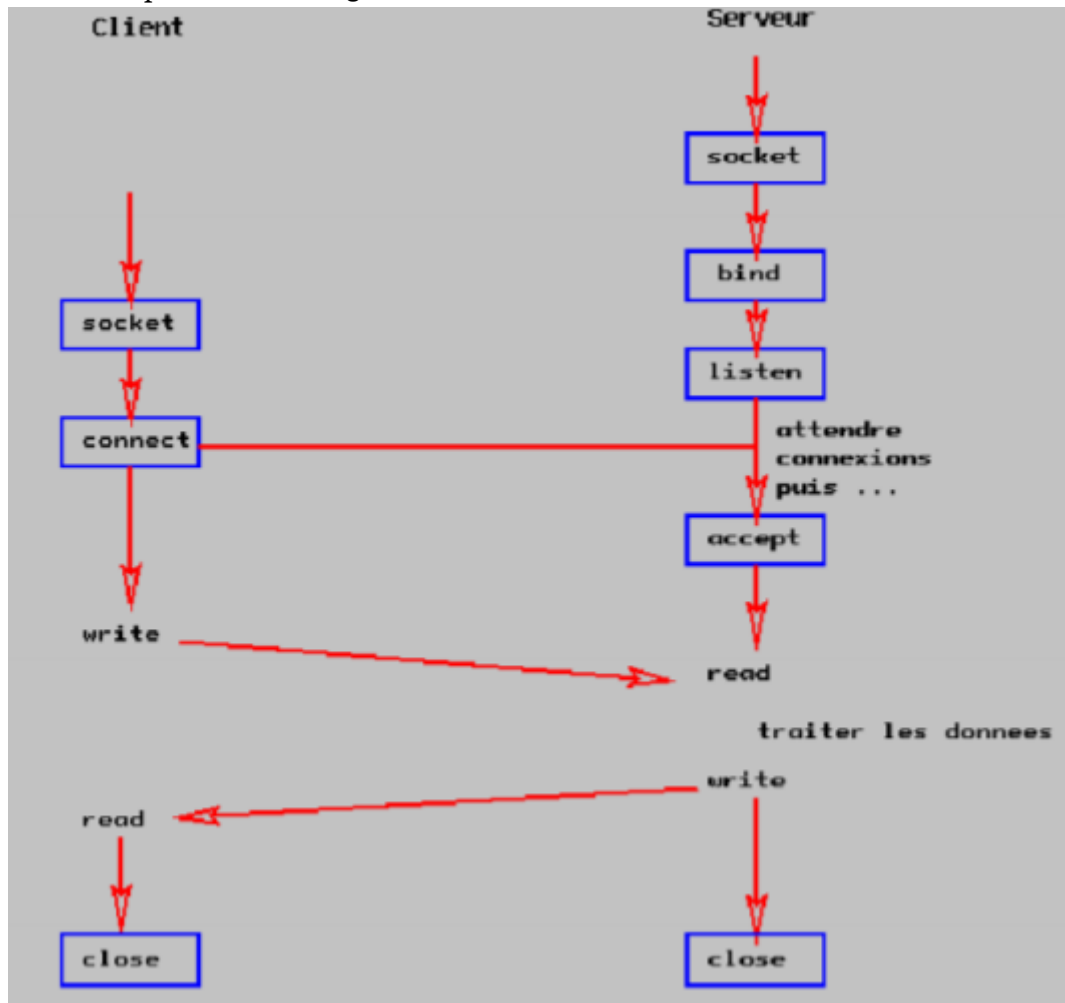
Projet 3

Sockets TCP (Client/Serveur) sous Unix



Objectif : Un socket est un point de communication par lequel un processus peut émettre et recevoir des données. Réaliser une application permettant la communication entre processus distants

Schéma explicatif : L'échange entre un client et un serveur TCP



• Côté Client (demandeur de la connexion) : le socket est dit actif

- crée un socket `socket()`
- se connecte à une `connect()`
- lit et écrit dans le socket `read()`, `recv()`; `write()`, `send()`
- ferme le socket `close()`

• Côté Serveur (en attente de connexion) : le socket est dit passif

- crée un socket `socket()`
- associe une adresse au socket `bind()`
- se met à l'écoute des connexions entrantes `listen()`
- accepte une connexion entrante `accept()`
- lit et écrit sur le socket `read()`, `recv()`; `write()`, `send()`
- ferme le socket `close()`

Programmes et Affichage de l'exécution :

Etape 1 : Socket_TCP_Client1.c

```
uir_student@ubuntu:~/Desktop/maha/Networkprojects$ gedit Socket_TCP_Client1.c
uir_student@ubuntu:~/Desktop/maha/Networkprojects$ gcc -o Socket_TCP_Client1 Socket_TCP_Client1.c
uir_student@ubuntu:~/Desktop/maha/Networkprojects$ ./Socket_TCP_Client1
Socket créée avec succès ! (3)
uir_student@ubuntu:~/Desktop/maha/Networkprojects$
```

Etape2 : Socket_TCP_Client2.c

```
uir_student@ubuntu:~/Desktop/maha/Networkprojects$ gedit Socket_TCP_Client2.c
uir_student@ubuntu:~/Desktop/maha/Networkprojects$ gcc -o Socket_TCP_Client2 Socket_TCP_Client2.c
uir_student@ubuntu:~/Desktop/maha/Networkprojects$ ./Socket_TCP_Client2
Socket créée avec succès ! (3)
connect: Connection refused
uir_student@ubuntu:~/Desktop/maha/Networkprojects$
```

Etape 3 : tester la connexion avec le serveur

```
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ gedit socketTcpClient
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ gcc -o socketTcpClient socketTcpClient2.c
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ ./socketTcpClient2
Socket créée avec succès ! (3)
Connexion au serveur réussie avec succès !
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$
TX packets:1411 errors:0 dropped:0 overruns:0 on interface: eth0
collisions:0 txqueuelen:1000
RX bytes:152061 (152.0 KB)
uir_student@ubuntu:~$ nc -l -p 5000
uir_student@ubuntu:~$
```

Etape4 : Echange de données : socket_TCP_Client3.c

```
Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.255.255.0
inet6 addr: ::1/128 Scope:Local
UP LOOPBACK RUNNING MTU:65536
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ ./socketTcpClient3
Socket créée avec succès ! (3)
Connexion au serveur réussie avec succès !
Message Hello world !
envoyé avec succès (14 octets)
Message reçu du serveur : ok
(3 octets)
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$
OK
uir_student@ubuntu:~$ nc -l -p 5000
Hello world !
ok
uir_student@ubuntu:~$
```

Etape5 : réalisation d'un serveur TCP, Socket_TCP_Serveur1.c

```

uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ gedit socketTcpServer
1.c
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ gcc -o socketTcpServe
r1 socketTcpServer1.c
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ ./socketTcpServer1
Socket créée avec succès ! (3)
Socket attachée avec succès !
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$

```

Attention, tout de même de bien comprendre qu'un numéro de port identifie un processus communiquant ! Exécutons deux fois le même serveur et on obtient alors :

```

uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ ./socketTcpServer1 &
./socketTcpServer1
[1] 5826
Socket créée avec succès ! (3)
Socket attachée avec succès !
Socket créée avec succès ! (3)
bind: Address already in use
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$

```

Etape6 : Mise En attente une connexion Socket_TCP_Serveur2.c

```

uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ gedit socketTcpServer
2.c
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ gcc -o socketTcpServe
r2 socketTcpServer2.c
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ ./socketTcpServer2
Socket créée avec succès ! (3)
Socket attachée avec succès !
Socket placée en écoute passive ...
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$

```

Etape7 : Accepter les demandes de la connexion Socket_TCP_Serveur3.c

```

uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ gedit socketTcpServer
3.c
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ gcc -o socketTcpServe
r3 socketTcpServer3.c
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ ./socketTcpServer3
Socket créée avec succès ! (3)
Socket attachée avec succès !
Socket placée en écoute passive ...
Attente d'une demande de connexion (quitter avec Ctrl-C)

```

On va exécuter deux clients à la suite :

```

uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$ ./socketTcpServer3
Socket créée avec succès ! (3)
bind: Address already in use
uir_student@ubuntu:~/Desktop/maha/Networkprojects/test$
uir_student@ubuntu:~$ nc -l -p 5000
Hello world !
uir_student@ubuntu:~$ ./socketTcpServer3
Socket créée avec succès ! (3)
Connexion au serveur réussie avec succès !
Message Hello world !
uir_student@ubuntu:~$
Message reçu du serveur : okk
uir_student@ubuntu:~$
Hello world !
uir_student@ubuntu:~$
uir_student@ubuntu:~$

```

Sources :

https://www.youtube.com/watch?v=IQ-3S4fJ0U&list=PLPyar5G9aNDvs6TtdpLcVO43_jvxp4eml

<https://lig-membres.imag.fr/donsez/cours/socketunix.pdf>

<http://sdz.tdct.org/sdz/les-sockets.html#Laselectiondesockets>

<http://tvaira.free.fr/reseaux/tp-sockets-tcp-linux.pdf>