



INF3405 – Réseaux informatiques

Hiver 2017

TP No. 3

Groupe 2

1792473 – Richer Archambault

1794745 – Kevin Pantelakis

POSTE : L4708-07

Soumis à : Fabien Berquez

16 février 2017

Les résultats de l'exécution du programme sont :

```
X:\Bureau\Lab3_Code_Source_H2017\Lab3_Code_Source\Lab3_Clients\Release\Lab301.exe
getaddrinfo a réussi!
Adresse # 1
  Flags: 0x0
  Famille: AF_INET6 (IPv6)
  Adresse IPv6: fe80::a997:1ef2:bafa:7da9
  Taille de cette adresse: 28 octets
  Nom canonique: (null)
Adresse # 2
  Flags: 0x0
  Famille: AF_INET6 (IPv6)
  Adresse IPv6: fe80::c142:aefd:8d71:467b
  Taille de cette adresse: 28 octets
  Nom canonique: (null)
Adresse # 3
  Flags: 0x0
  Famille: AF_INET6 (IPv6)
  Adresse IPv6: fe80::d5ee:6cfe:31c7:a974
  Taille de cette adresse: 28 octets
  Nom canonique: (null)
Adresse # 4
  Flags: 0x0
  Famille: AF_INET6 (IPv6)
  Adresse IPv6: fe80::bd23:1415:4849:7f28
  Taille de cette adresse: 28 octets
  Nom canonique: (null)
Adresse # 5
  Flags: 0x0
  Famille: AF_INET (IPv4)
  Adresse IPv4: 132.207.29.107
  Taille de cette adresse: 16 octets
  Nom canonique: (null)
Adresse # 6
  Flags: 0x0
  Famille: AF_INET (IPv4)
  Adresse IPv4: 192.168.44.64
  Taille de cette adresse: 16 octets
  Nom canonique: (null)
Adresse # 7
  Flags: 0x0
  Famille: AF_INET (IPv4)
  Adresse IPv4: 192.168.233.1
  Taille de cette adresse: 16 octets
  Nom canonique: (null)
Adresse # 8
  Flags: 0x0
  Famille: AF_INET (IPv4)
  Adresse IPv4: 192.168.142.1
  Taille de cette adresse: 16 octets
  Nom canonique: (null)
Adresse # 9
  Flags: 0x0
  Famille: AF_INET6 (IPv6)
  Adresse IPv6: 2002:84cf:1d6b::84cf:1d6b
  Taille de cette adresse: 28 octets
  Nom canonique: (null)
Appuyer une touche pour finir...
```

1)

Ce poste détient 9 adresses

2)

Un NAT donne une adresse IPv4 et IPv6 par interface réseau qui supporte l'a norme.

3)

Il y a 5 classes d'adresse IPv4 :

Classe A : 0.0.0.0 à 127.255.255.255 et le masque : 255.0.0.0

Classe B : 128.0.0.0 à 191.255.255.255 et le masque : 255.255.0.0

Classe C : 192.0.0.0 à 223.255.255.255 et le masque : 255.255.255.0

Classe D : 224.0.0.0 à 239.255.255.255 et aucun masque

Classe E : 240.0.0.0 à 255.255.255.255 et aucun masque

4)

L'adresse 132.207.29.107 correspond à une classe B

5)

a) L'adresse : 10000100. 11001111. 00011101. 01101011


Le masque : 11111111. 11111111. 00000000. 00000000

b)

Binaire : 10000100. 11001111. 00000000. 00000000

Décimale : 132.207.0.0

6)

 X:\Bureau\Lab3_Code_Source_H2017\Lab3_Code_Source\Lab3_Clients\Release\Lab301.exe

```
getaddrinfo a reussi!
Adresse # 1
    Flags: 0x0
    Famille: AF_INET (IPv4)
    Adresse IPv4: 132.207.29.103
    Taille de cette adresse: 16 octets
    Nom canonique: (null)
Appuyer une touche pour finir...
```

7)

Oui, car ils sont dans le même réseau, et partagent le même 3^e octet. Donc on peut supposer que tous les ordinateurs qui ont une adresse correspondant au masque 255.255.29.0 seront dans le même sous-réseau.

8)

La fonction retourne à la fois les informations relatives aux host passés en paramètre (adresse IP, flags, ...) ainsi qu'une valeur pour indiquer le bon fonctionnement de la fonction.

9) Les lignes sont :

```
46 // On indique le nom et le port du serveur auquel on veut se connecter
47 //char *host = "L4708-XX";
48 //char *host = "L4708-XX.lerb.polymtl.ca";
49 //char *host = "add_IP locale";
50 char *host = "132.207.29.XXX";
51 char *port = "5000";
```

Le port utilisé est 5000

10)

La famille AF_INET

11)

Le type choisi précise qu'il y aura une connexion en TCP.

Le résultat obtenu de l'exécution est la transmission d'un mot du client vers le serveur, puis du serveur vers le client

12)

La fonction send()

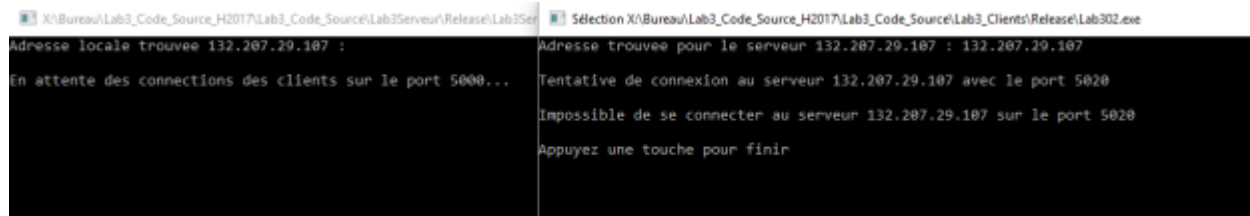
13)

La fonction recv()

14)

Le mot envoyé est 123456 et le mot reçu est 654321 (mot de 6 lettres comme le code ne fonctionne pas pour 7). Le serveur reçoit donc le mot à l'envers.

15)



```
X:\Bureau\Lab3_Code_Source_H2017\Lab3_Code_Source\Lab3Serveur\Release\Lab3Ser
Adresse locale trouvee 132.207.29.107 :
En attente des connections des clients sur le port 5000...

Sélection X:\Bureau\Lab3_Code_Source_H2017\Lab3_Code_Source\Lab3_Clients\Release\Lab302.exe
Adresse trouvee pour le serveur 132.207.29.107 : 132.207.29.107
Tentative de connexion au serveur 132.207.29.107 avec le port 5020
Impossible de se connecter au serveur 132.207.29.107 sur le port 5020
Appuyez une touche pour finir
```

Il est impossible de se connecter au serveur, car le port est différent.

16)

L'utilité du port est de diriger les flux de donnée entre les différentes applications qui utilisent une même interface réseau.

17)

PROMT_IP:

```
printf("Saisir une adresse IP ");
gets_s(host);

char end[2];

if (strlen(host) == 14) {
    end[0] = host[12];
    end[1] = host[13];

    if (!(atoi(end) > 1 && atoi(end) < 28)) {
        printf("Adresse IP invalide\n");
        goto PROMT_IP;
    }
}
```

```

else {
    printf("Adresse IP invalide\n");
    goto PROMT_IP;
}

```

18)

Ce mécanisme est les threads sur lesquels sont utilisés des sockets.

19)

```

SOCKET sd = accept(ServerSocket, (sockaddr*)&sinRemote,
&nAddrSize);

```

```

if (sd != INVALID_SOCKET) {
    cout << "Connection acceptee De : " <<
    inet_ntoa(sinRemote.sin_addr) << ":" <<
    ntohs(sinRemote.sin_port) << "." <<
    endl;

```

```

    DWORD nThreadId;

```

```

    CreateThread(0, 0, EchoHandler, (void*)sd, 0,
&nThreadId);
}

```

```

else {
    cerr << WSAGetLastError("Echec d'une
connection.") <<
    endl;

```

```
// return 1;  
}
```

20)

Protocole	Port(s) par défaut
HTTP	80
FTP	20 et 21
HTTPS	443
SSH	22