

# Socket

Frédéric Hamel   Aldo Lamarre

5 mars 2018

# Qu'est-ce qu'un sockets ?

## Définition

C'est un objet qui permet d'établir la connexion entre plusieurs processus sur différente machine. (Connexion)

# Les différents modes d'un socket

## Définition

- E/S : Écrire ou lire des bytes. Le socket a un comportement similaire à un fichier.
- Écoute : Attente pour une connexion sur un port (serveur mode).

# Libraries posix

```
#include <netdb.h>
```

```
#include <sys/socket.h>
```

```
#include <sys/types.h>
```

```
#include <unistd.h>
```

# Fonctions utiles

- `htons` et `htonl` : Convertie l'entier non signé de l'ordre des bytes de l'hôte vers l'ordre des bytes sur le réseau.
- `socket` : Création d'un objet socket.
- `bind` : Associe une adresse à un objet socket.
- `listen` : Met le socket en mode écoute. (socket serveur)
- `accept` : Accepte la connexion d'un client à un socket serveur.
- `connect` : Connecte le socket à un serveur et permet d'effectuer des opérations E/S avec les fonction `recv` et `send`. (le socket utilisé pour communiquer avec le serveur)
- `close` : Fermer un fichier ou un socket.

# Structure utile

Cette structure est définie dans la librairie `netinet/in.h`.

## Codes

```
struct sockaddr_in {  
    // Adresse du socket.  
    struct in_addr sin_addr;  
  
    // Famille du socket. (AF_INET utilisé dans le cours)  
    sa_family_t sin_family;  
  
    // Entier 0-65535 qui indique le port réseau.  
    in_port_t sin_port;  
  
    // Padding.  
    unsigned char sin_zero;  
};
```

# TCP Socket côté serveur

## Création d'un socket server

- Créer un socket tcp avec la fonction `socket`.
- Associer le socket à une adresse avec la fonction `bind`.
- Mettre le socket en mode écoute avec la fonction `listen`.

Pour accepter une connexion au serveur, il suffit d'appeler la fonction `accept` sur le socket serveur.

# Socket serveur (exemple)

## Code

```
int server_socket;
struct sockaddr_in addr;

// Create socket.
server_socket = socket(AF_INET, SOCK_STREAM | SOCK_NONBLOCK, 0);

memset(&addr, 0, sizeof(addr));
addr.sin_family = AF_INET;
// Set port to 2018.
addr.sin_port = htons(2018);
// Set address to 127.0.0.1.
addr.sin_addr.s_addr = htonl(INADDR_LOOPBACK);

// Set socket address.
bind(server_socket, &addr, sizeof(addr));

// Set the socket mode to listen.
listen(server_socket, 5);
```

## Note

Le code ci-dessus ne gère pas les erreurs liées aux fonctions socket, bind et listen.



# TCP Socket côté client

## Connexion à un serveur

- Créer un socket tcp avec la fonction `socket`.
- Connect le socket au serveur avec la fonction `connect`.

# Socket client (exemple)

## Code

```
int client_socket = socket(AF_INET, SOCK_STREAM, 0);
struct sockaddr_in addr;

addr.sin_family = AF_INET;
addr.sin_port = htons(2018);
addr.sin_addr.s_addr = htonl(INADDR_LOOPBACK);

// Connect client socket to server.
connect(client_socket, &addr, sizeof(addr));
```

## Note

Le code ci-dessus ne gère pas les erreurs liées aux fonctions socket et connect.