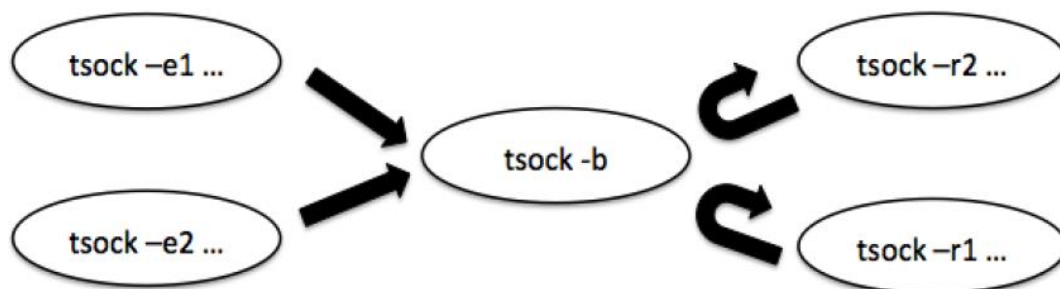


Compte rendu TD2

On souhaite étendre tsock pour lui donner des capacités de serveur de boîtes aux lettres (BAL) entre n émetteur et p récepteurs permettant :

- L'envoi par un émetteur d'une ou plusieurs lettres à destination d'un récepteur i.
- La réception et le stockage des lettres par le serveur de BAL dans une boîte aux lettres dédiée au récepteur i.

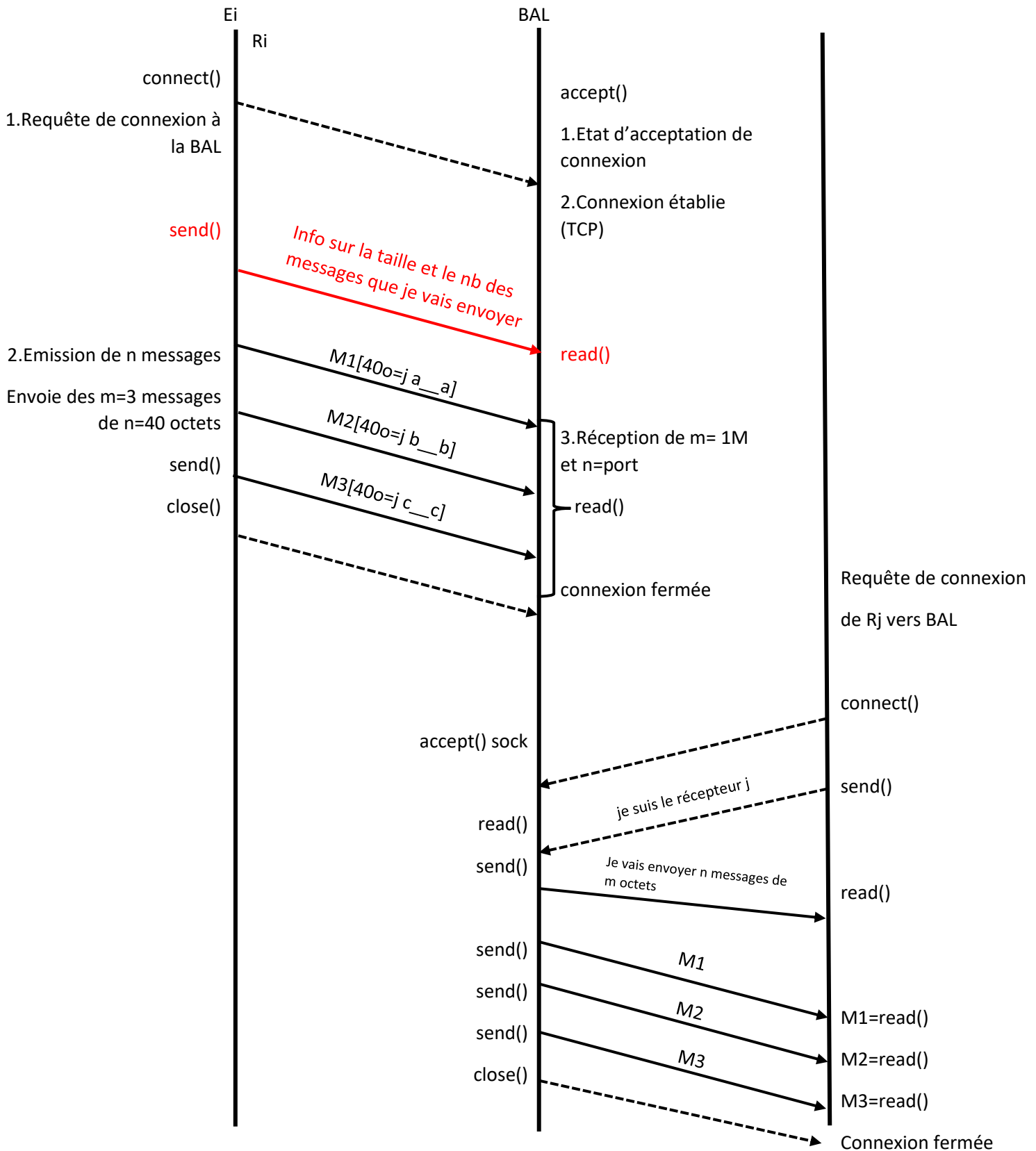


- e1 est un émetteur qui veut envoyer vers le récepteur r1.
- e2 est un émetteur qui veut envoyer vers le récepteur r2.

Rappel des hypothèses :

- Tous les échanges se font via des connexions TCP
- Un émetteur E est client de la connexion TCP à établir avec le serveur de BAL. Il ferme la connexion une fois la / les lettres envoyées
 - Entre chaque émetteur et la communication avec la boîte aux lettres, puisque c'est une communication TCP, il faut une réclamation de la connexion par l'émetteur.
- Un récepteur R est client de la connexion TCP à établir avec le serveur de BAL en vue de la récupération de ses lettres.
- Le serveur de BAL est serveur des connexions TCP en provenance des émetteurs / récepteurs, via un unique socket dont le numéro de port supposé connu de tous. Il ferme la connexion avec un R une fois la / les lettres envoyées à R.
 - Lorsque récepteur se connecte, le récepteur initie la communication et le système de boîte aux lettres envoie le message et ferme la communication TCP.

Communication entre un émetteur et récepteur avec le système de boîte aux lettres :



Problème lors de l'envoi des message de E1:

Envoie d'une série de message dont on ne connaît pas la taille

Résolution :

Avant d'envoyer les messages M1, M2, M3, on envoie un premier message contenant le nombre de message et la taille des messages envoyés.

Programmation de l'émetteur :

- Création du socket local (On laissera l'OS lui attribué un port tout seul) :
 - o `sock=socket(AF_INET,SOCK_STREAM,0)`
- Réinitialisation de la structure d'adresse :
 - o `memset((char *)&adr_distant, 0, sizeof(adr_distant)) ; /* reset */`
- Affectation du n° port du socket de la boite aux lettres et récupération de l'adresse IP :
 - o `adr_distant.sin_family = AF_INET ; /* domaine Internet */`
 - o `adr_distant.sin_port = atoi(argv[argc-1]) ; /* n° de port */`
 - o `hp = gethostbyname(argv[argc-2]) ; /* récupération de l'adresse IP de la boite aux lettres */`
 - o `memcpy((char*)&(adr_distant.sin_addr.s_addr), hp->h_addr, hp->h_length) ; /* Affectation de l'adresse IP récupéré à notre structure d'adresse */`
- Connection à la boite aux lettres
 - o `Connect(sock, &adr_distant, lg_adr_distant); /* Connexion à la boite aux lettres */`
- Envoie de la configuration :
 - o `strcpy(config, "%c%d,%d,%d", 'e', n°recepteur,nb_messages,lg_messages) ; /* création du message de config */`
 - o `write(sock, config, sizeof(config)); /* envoi de la config des messages à la boite aux lettres, au travers du port socket créé */`
- Envoie du message
 - o `Write(sock,M,lg_m); /*envoi des n messages de longueur lg_m */`
- Fermeture du socket :

- `close(sock);`

Programmation du récepteur :

- Création du socket local :
 - `sock=socket(AF_INET,SOCK_STREAM,0)`
- Réinitialisation de la structure d'adresse :
 - `memset((char *)& adr_local, 0, sizeof(adr_local)) ; /* reset */`
- Affectation d'un n° de port + choix de l'adresse IP sur laquelle on va recevoir (dans ce cas, n'importe laquelle des adresses associées à une de nos cartes réseaux) :
 - `adr_local.sin_family = AF_INET ;`
 - `adr_local.sin_port = atoi(argv[argc-1]) ;`
 - `adr_local.sin_addr.s_addr = INADDR_ANY;`
 - `hp = gethostbyname(argv[argc-2]) ; /* récupération de l'adresse IP de la boite aux lettres */`
 - `memcpy((char*)&(adr_distant.sin_addr.s_addr), hp->h_addr, hp->h_length) ;`
`/* Affectation de l'adresse IP récupéré à notre structure d'adresse */`
- Associer le n° de port choisi au socket :
 - `bind (sock, (struct sockaddr *)&adr_local, lg_adr_local)`
- Connection à la boite aux lettres
 - `Connect (sock, &adr_distant, lg_adr_distant); /* Connexion à la boite aux lettres */`
- Envoie de la configuration :
 - `strcpy(config, "%e%d ", 'r', n°recepteur) ;`
 - `write(sock, config, sizeof(config)); /* envoi du numéro de récepteur à la boite aux lettres */`
- Acceptation des machines voulant se connecter :
 - `Listen(sock,n) /* On met en attente la connexion de n machines au maximum */`
 - `sock2=accept (sock, &adr_to_register, &lg_adr_to_register); /* Création du socket d'écoute */`

- Réception du message de configuration :
 - Read(sock2, config, sizeof(config)) /* Read de la config des messages qui vont être reçues */
 - sscanf(config, "%d%d", &nb_messages, &lg_messages)
- Réception des messages:
 - Read(sock2, M, lg_message) /* lecture des messages */
- Fermeture du socket :
 - close(sock2);
 - close(sock1);
- Traitement des messages recues:
 - Découpage du message reçu grâce au message de config reçu avant

Programmation de la boîte aux lettres :

- Création du socket local :
 - sock=socket(AF_INET,SOCK_STREAM,0)
- Réinitialisation de la structure d'adresse :
 - memset((char *)& adr_local, 0, sizeof(adr_local)) ; /* reset */
- Affectation d'un n° de port + choix de l'adresse IP sur laquelle on va recevoir (dans ce cas, n'importe laquelle des adresses associées à une de nos cartes réseaux) :
 - adr_local.sin_family = AF_INET ;
 - adr_local.sin_port = atoi(argv[argc-1]) ;
 - adr_local.sin_addr.s_addr = INADDR_ANY;
- Associer le n° de port choisi au socket :
 - bind (sock, (struct sockaddr *)&adr_local, lg_adr_local))
- Acceptation des machines voulant se connecter :
 - Listen(sock,n) /* On met en attente la connexion de n machines au maximum */
 - sock2=accept (sock, &adr_to_register, &lg_adr_to_register); /* Création du socket d'écoute */

- Réception du message de configuration :
 - Read(sock2, config, sizeof(config)) /* Read de la config des messages qui vont être reçues */
 - sscanf(config, "%c", &entite) /* recuperation de l'entité */
- Boite aux lettres si réception d'un émetteur :
 - Fin de la réception du message de configuration :
 - sscanf(config, « %d,%d,%d », &n°recepteur, &nb_messages, &lg_message)
 - Réception des messages:
 - Read(sock2, M, lg_message) /* lecture des messages */
 - Traitement des messages :
 - Découpe des messages
 - Placement des messages dans la boite aux lettres correspondante
 - Fermeture du socket :
 - close(sock2);
 - close(sock1);
- Boite aux lettres si réception d'un émetteur :
 - Fin de la réception du message de configuration :
 - sscanf(config, « %d », &n°recepteur)
 - Récupération des messages dans la boite aux lettres :
 - Récupération des messages dans la boite aux lettres correspondante au n°recepteur reçu
 - Envoie de la configuration :
 - strcpy(config, "%d,%d", nb_messages, lg_messages) ; /* création du message de config */
 - write(sock, config, sizeof(config)); /* envoi de la config des messages à la boite aux lettres, au travers du port socket créé */
 - Envoie du message
 - write(sock, M, lg_m); /* envoi des n messages de longueur lg_m */

JUMIN – CHANFREAU
FISA-AE

- Fermeture du socket :
 - Close(sock2);
 - close(sock);