CASTEL Alexian

DIBON Josselin

FERRARI Axel

**Rapport Projet réseau**

**Etude des options de sockets**

1. ***OPTION SO\_REUSEADDR***
2. Principe

Cette option permet de réutiliser l’adresse locale sur un autre socket

1. Impact

Cette option permet d’éviter l’erreur « Address already in use » lors de l’utilisation d’un deuxième socket.

1. Implémentation

setsockopt(sock, SOL\_SOCKET, SO\_REUSEADDR, &valOption, sizeof(valOption));

1. Analyse
2. ***OPTION SO\_RCVBUF***
3. Principe

Cette option modifie la taille maximum du buffer de réception, la taille minimum est de 256 octets.

1. Impact

SO\_RCVBUF permet de limiter les données reçu provenant du serveur TCP.

1. Implémentation

setsockopt(sock, SOL\_SOCKET, SO\_RCVBUF, &valOption, sizeof(valOption));

1. Analyse
2. ***OPTION SO\_SNDBUF***
3. Principe

Cette option modifie la taille maximum du buffer d’envoi, la taille minimum est de 256 octets.

1. Impact

SO\_SNDBUF permet de limiter les données voulant être envoyé sur le serveur TCP.

1. Implémentation

setsockopt(sock, SOL\_SOCKET, SO\_SNDBUF, &valOption, sizeof(valOption));

1. Analyse
2. ***OPTION SO\_KEEPALIVE***
3. Principe

Le concept du keepalive est très simple, lorsque l’on crée une connexion TCP, on lance un chronomètre. Ce temps permet d’utiliser la procédure keepalive de TCP. Quand la durée maximale du keepalive est atteinte, on adresse à  l'hôte distant un paquet ne contenant aucune donnée sauf le bit ACK afin de maintenir la connexion avec le client et de savoir si le client est toujours disponible.

1. Impact

Cette option permet de maintenir une connexion TCP lorsqu’il n’y aucune activité sur celle-ci.

1. Implémentation

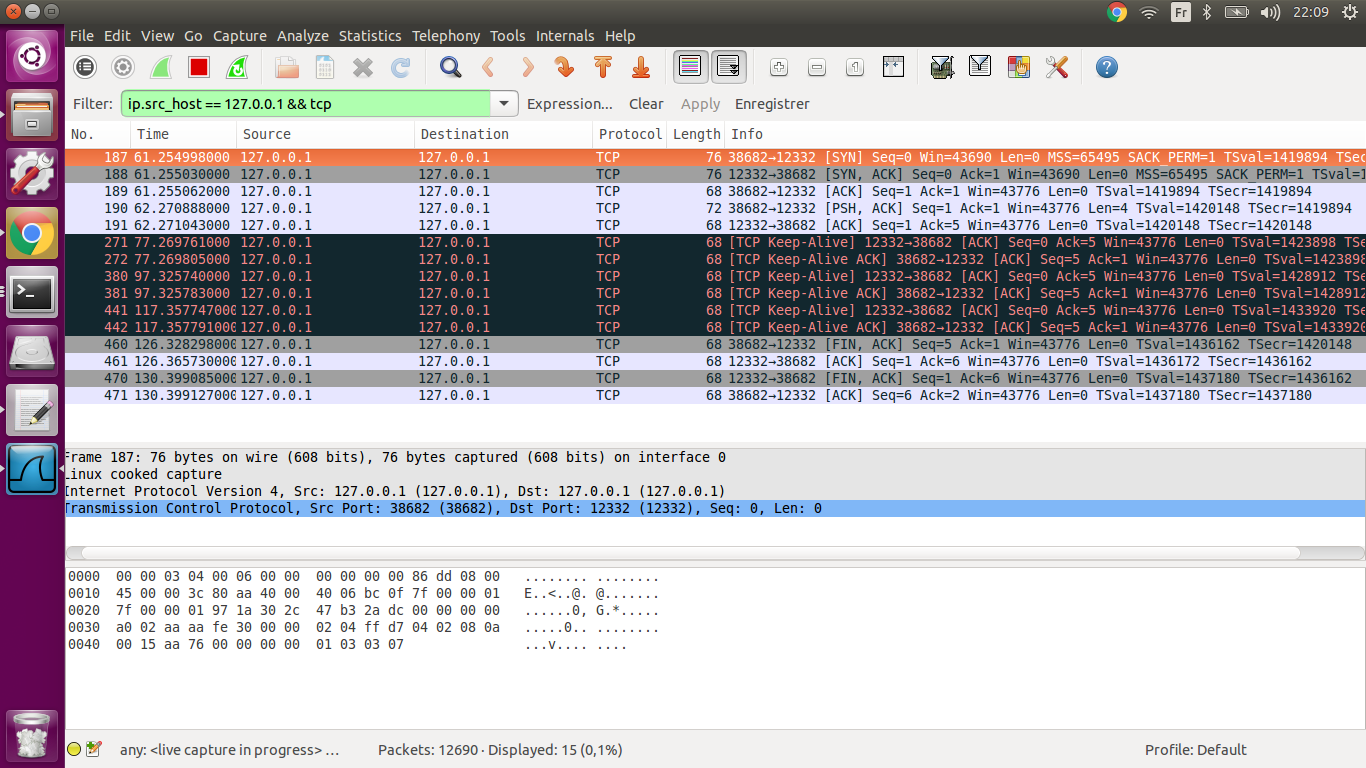
setsockopt(sock, SOL\_SOCKET, SO\_KEEPALIVE, &valOption, sizeof(valOption));

setsockopt(sock, IPPROTO\_TCP, TCP\_KEEPIDLE, &idle, sizeof(int));

setsockopt(sock, IPPROTO\_TCP, TCP\_KEEPINTVL, &interval, sizeof(int));

setsockopt(sock, IPPROTO\_TCP, TCP\_KEEPCNT, &maxpkt, sizeof(int));

1. Analyse



Sur cette capture on peut voir que le keepalive s’exécute 15 secondes après la dernière trame et qu’il se répète toutes les 20 secondes. Ces paramètres sont réglables grâce aux options de tcp : TCP\_KEEPIDLE (durée à attendre avant de déclencher), TCP\_KEEPINTVL (durée intervalle répétition), TCP\_KEEPCNT (nombre d’occurrence avant de fermer la connexion).

1. ***OPTION SO\_ERROR***
2. Principe

Rapporte les erreurs rencontrées sur le socket puis les supprime.

1. Impact
2. Implémentation

setsockopt(sock, SOL\_SOCKET, SO\_ERROR, &valOption, sizeof(valOption));

1. Analyse
2. ***OPTION SO\_DEBUG***
3. Principe

Permet de suivre les informations du débogage.

1. Impact
2. Implémentation

setsockopt(sock, SOL\_SOCKET, SO\_DEBUG, &valOption, sizeof(valOption));

1. Analyse
2. ***OPTION SO\_OOBINLINE***
3. Principe

Quand l'option SO\_OOBINLINE est active, les données urgentes sont mises dans le flux de données normal.

1. Impact
2. Implémentation

setsockopt(sock, SOL\_SOCKET, SO\_OOBINLINE, &valOption, sizeof(valOption));

1. Analyse