
TD5 Contrôle de flux

Exercice 1 : Contrôle de flux dans le protocole à fenêtre coulissante

L'objectif est maintenant d'implanter un contrôle de flux similaire à celui utilisé par TCP en s'appuyant sur le simulateur nachos, et en particulier en ajoutant cette fonctionnalité au protocole de liaison étudié la semaine dernière.

Archivez et sauvegardez le contenu de votre répertoire `nachos_reseau`, puis détruisez-le et récupérez une nouvelle version de l'archive (`~pecher/TI/nachos_reseau_II.tar.gz`). Placez-vous dans le répertoire `nachos_reseau/code/reseau` et tapez `make`.

La version obtenue (`liaison-v4`) est une solution opérationnelle à la dernière question du TP 4 (fenêtre glissante sans rejet sélectif). Toutefois, elle souffre d'un problème ennuyeux : l'émetteur arrête d'émettre des trames uniquement lorsqu'il attend un acquittement qui lui permettrait de "faire de la place" dans son tampon d'émission, ou lorsque l'application ne lui donne plus de trames à émettre, bien évidemment. Autrement dit, lorsque l'application (côté récepteur) ne consomme pas suffisamment vite les trames reçues, le récepteur est obligé de jeter les trames alors qu'elles ont pourtant été correctement acheminées.

Lancez l'exécution du programme `nachos` en mode *debug* (option `-d 1`) et observez les messages de type *Trame xx détruite (tampon saturé)* correspondant à ce phénomène...

Pour résoudre ce problème, une solution est de renforcer le protocole en ajoutant un champ "taille de fenêtre" que le récepteur utilisera pour indiquer à l'émetteur le nombre d'emplacements libres dans son tampon (ou encore, et c'est équivalent, le numéro de trame à partir duquel l'émetteur devra temporairement s'arrêter d'émettre). Le récepteur peut ainsi *contrôler le flux* de trames émises par l'émetteur de manière à éviter les émissions inutiles.

Implantez ce protocole dans un fichier `liaison-v5.cc` et vérifiez que les destructions de trames n'apparaissent plus du côté du récepteur.