Performances de MPTCP sur différentes topologies virtuelles

Encadrant : **S. Secci, Y. Benchaïb, M. Coudron,**

Etudiants : **R. Ly, K. Lam, Q. Dubois, S. Ravier**

Table des matières

1 Cahier des charges

1.1 Objectifs

Les objectifs du projet sont de

- mesurer les performances de MPTCP sur différentes topologies de réseaux virtuels.
- modifier l'ordonnanceur de MPTCP pour priviliéger une répartition équilibrée sur les différents sous-flots.

1.2 Contexte

MPTCP est une extension de TCP qui permet pour une connexion TCP donnée d'utiliser plusieurs chemins pour l'échange de données. La multiplicité des sous-flots a pour but d'améliorer le débit et d'augmenter la résilience de la connexion [1–3].

L'utilisation de MPTCP ne doit en aucun cas diminuer les performances de l'utilisateur en dessous des performances d'une simple connexion TCP ou diminuer le débit des autres utilisateurs sur le même réseau. Les performances de MPTCP dépendent en partie de l'algorithme utilisé pour la répartition des données entre les différents sous-flots ouverts [4]. Pour caractériser les performances de l'ordonnanceur de MPTCP, nous allons le tester dans différents réseaux virtualisés en utilisant dans un premier temps l'algorithme par défaut.

L'utilisation exclusive d'un chemin dans une connexion TCP permet la possibilité à un attaquant d'écouter la totalité des données échangées, d'utiliser des attaques de type rejeu ou *man in the middle*. L'emploi de MPTCP améliorerait la sécurité si les données transitaient de manière équilibrée entre les différents sous-flots. Le débit global de la connexion serait affecté car les chemins les plus lents vont ralentir le débit des chemins les plus rapides. Nous allons, dans un premier temps, modifier l'ordonnanceur afin de garantir la répartition équitable des charges au détriment du débit puis analyser l'influence de cette modification sur les performances de MPTCP dans les topologies réseaux utilisés précédemmment. Si nous avons le temps, nous réfléchirons s'il est possible d'obtenir un algorithme pouvant garantir une sécurité et un débit supérieur à celle d'une simple connexion TCP pour que cela soit conforme aux objectifs premiers de MPTCP.

1.3 Méthodes

La simulation des réseaux sera effectué via mininet qui va nous permettre de simuler un réseau OpenFlow. Pour caractériser l'influence de l'ordonnanceur sur les performances, nous utiliserons des réseaux simples où les différents sous-flots sont asymétriques et diffèrent par une propriété à la fois : latence, débit, pertes ...

Nous testerons différents algorithmes de répartition de charge entre les sous-flots : l'algortihme implémenté par défaut (LIA), celui qui satisfait l'optimum de pareto par rapport aux objectifs de MPTCP (OLIA, [4]) ou celui qu'on va développer qui permet de répartir de manière équilibréé les charges dans les différents sous-flots.

Nous utiliserons des scripts python pour automatiser la création des sous-flots ainsi que les tests de performance.

Pour implémenter l'algorithme de balancement équilibré (favorisant la sécurité), nous modifierons le code source de MPTCP : `https://github.com/multipath-tcp/mptcp/tree/mptcp_v0.88/net/mptcp`. Dans un premier temps,

- nous privilégierons une répartition équitable entre les différents sous-flots.
- il est aussi nécessaire de gérer les problèmes liés à la retransmission, pour éviter que les paquets soient réexpédiés sur le même sous-flot en cas de congestion ou de panne d'un sous-flot, qui dans ce cas renderait l'avantage de l'agorithme nulle.
- et ensuite mesurer les performances de notre algorithme.

1.4 Répartition

Nous allons répartir la charge de travail en deux groupes de deux personnes. Le premier groupe va édifier les réseaux virtuels ainsi que les outils d'analyses. Le deuxième groupe personnes s'occupera du code de l'ordonnanceur sans exclure l'aide ou la concertation entre les deux groupes.

Références

[1] A. Ford, C. Raiciu, M. Handley, S. Barre, and J. Iyengar, "Architectural guidelines for multipath tcp development," *RFC 6182*, March 2011.

[2] A. Ford, C. Raiciu, M. Handley, and O. Bonaventure, "Tcp extensions for multipath operation with multiple addresses," *RFC 6824*, January 2013.

[3] M. Coudron, S. Secci, G. Pujolle, P. Raad, and P. Gallard, "Cross-layer cooperation to boost multipath tcp performance in cloud networks," in *Cloud Networking (CloudNet), 2013 IEEE 2nd International Conference on*, pp. 58–66, IEEE, 2013.

[4] R. Khalili, N. Gast, M. Popovic, U. Upadhya, and J.-Y. Le Boudec, "Mptcp is not pareto-optimal : Performance issues and a possible solution," *Networking, IEEE/ACM Transactions on*, vol. 21, no. 5, pp. 1651–1665, 2013.