RFC 4814 : Hash and Stuffing: Overlooked Factors in Network Device Benchmarking

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 4 avril 2007

Date de publication du RFC : Mars 2007

https://www.bortzmeyer.org/4814.html

Un très intéressant RFC pour les amateurs de "*benchmarks*". Il indique deux points faibles courants dans ces tests de performance réseau.

On le sait, les "*benchmarks*" sont très difficiles à faire proprement (et la plupart des résultats publiés le sont dans un but commercial, donc l'auteur n'essaie même pas d'être honnête). Il est donc utile que ce RFC avertisse les réalisateurs de tests de performances de réseaux.

Plusieurs RFC ont déjà été écrits sur les tests de performance, comme les RFC 2544 [1] et RFC 2889. Ils sont l'œuvre du "*Benchmarking Methodology Working Group*" <http://home.comcast.net/~acmacm/BMWG/>.

Le premier problème est le fait que le traitement d'un paquet dépend souvent du résultat d'une fonction de hachage et que celle-ci dépend du contenu du paquet. Si les paquets du tests ne sont pas suffisamment variés, les collisions dans la fonction de hachage affecteront le résultat.

Par exemple, notre RFC cite le cas d'un commutateur Ethernet qui comprend huit "*network processors*" (les processeurs spécialisés qui vont traiter les paquets entrants) et où l'affectation d'un paquet entrant à un "*network processor*" donné est fait par un hachage des adresses MAC de source et de destination. Si le jeu de test génère des adresses trop semblables, la collision des résultats de la fonction de hachage fait qu'un seul processeur sera utilisé, et qu'on sous-estimera le débit du commutateur. Et le même phénomène peut se reproduire à d'autres couches, comme la couche réseau ou la couche transport.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc2544.txt

Notre RFC recommande donc de générer des paquets le plus aléatoires possibles.

Le deuxième problème est le fait que certains protocoles comme PPP doivent échapper certains motifs de bits qui pourraient être pris comme caractères de contrôle. Cet échappement rajoute des bits aux données, « faussant » ainsi les calculs de débit.

On notera que c'est pour cela qu'echoping `<http://echoping.sourceforge.net/>`, par défaut, envoie des données aléatoires (et dispose de l'option -f si on désire remplir avec des données fixes).

Il n'est pas évident de calculer en temps réel la taille effective des paquets et notre RFC recommande donc une approche probabiliste, le calcul de la probabilité qu'un échappement soit inséré, approche permettant d'arriver à des résultats presque exacts.