

RFC 2681 : A Round-trip Delay Metric for IPPM

Stéphane Bortzmeyer

<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 25 septembre 2008

Date de publication du RFC : Septembre 1999

<https://www.bortzmeyer.org/2681.html>

Les mesures de performances sur l'Internet sont un sujet passionnant mais complexe. Comme il faut bien commencer par définir précisément ce qu'on mesure, ce RFC ne spécifie pas un protocole mais uniquement une métrique, une grandeur qu'on veut mesurer et qui doit faire l'objet d'une définition rigoureuse. Cette métrique est le temps d'aller-retour (RTT) d'un paquet IP.

Cette mesure du RTT est connue grâce à l'outil ping. Mais ping a, entre autres défauts, celui de ne pas reposer sur une métrique rigoureuse, ce qui rend souvent difficile de savoir ce qu'il mesure exactement. Notre RFC s'attache donc à une définition précise d'une telle métrique. Il s'appuie pour cela sur le cadre et les définitions du RFC 2330¹, et suit de près le RFC 2679, qui décrivait une telle métrique pour les allers simples.

La section 1.1 du RFC commence par expliquer pourquoi une telle métrique est utile : notamment parce que certaines applications, notamment les plus interactives, fonctionnent mal si le RTT est trop élevé, que sa valeur minimale donne une idée des caractéristiques intrinsèques du lien et que des augmentations de ce RTT peuvent indiquer une congestion du lien. Elle est donc pertinente dans beaucoup de cas.

Mais pourquoi, continue cette section 1.1, mesurer le temps d'aller-retour et pas seulement celui d'aller simple comme le fait le RFC 2679 ? En effet, le RTT a plusieurs défauts : en cas de routage asymétrique, le temps de parcours aller peut être très différent du temps retour (et le RTT ne permet pas de voir cette différence). Même chose si des mécanismes divers (par exemple l'asymétrie des débits en ADSL) font qu'une direction a des caractéristiques très différentes d'une autre. C'est vrai, mais le RTT a aussi des avantages, notamment le fait qu'il est beaucoup plus facile à mesurer que l'aller simple, puisqu'il ne

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc2330.txt>

nécessite pas que les horloges des deux machines soient synchronisées (c'est sur ce principe que fonctionne ping).

Les lecteurs étant désormais convaincus de l'intérêt de cette métrique, la section 2 passe donc à sa définition : baptisée du nom scientifique de "*Type-P-Round-trip-delay*", le temps d'aller-retour a pour paramètre les adresses IP de la source et de la destination, le temps T de la mesure, pour unité la seconde et pour définition (section 2.4) le délai entre la mise du premier bit de la question sur le câble et la réception du dernier bit de la réponse. "*Type-P*" signifie simplement qu'il peut dépendre du type du paquet (par exemple TCP vs. UDP, ou bien le numéro de port, tout ce qui fait qu'un routeur peut traiter ce paquet rapidement ou pas). Ainsi, les mesures de ping ont pour "*Type-P*" « ICMP avec des paquets echo ». Ce terme de "*Type-P*" est défini dans la section 13 du RFC 2330.

Cette définition est donc très simple sur le papier. En pratique, elle pose quelques problèmes, signalés par la section 2.5 : le temps T de la mesure dépend, lui, d'une bonne synchronisation des horloges, la définition n'indique pas à partir de quand on renonce à attendre un paquet perdu (le résultat de la mesure est alors indéfini), etc.

Ce n'est pas tout : même lorsque la métrique est parfaite, les ordinateurs sont des machines physiques ayant plein de limitations, qui limitent la précision de la mesure. Il est symptomatique du mauvais état de la métrologie sur Internet qu'on voit souvent citer des résultats sans indiquer les marges d'erreur ou d'incertitude. La section 2.7 cite certaines de leurs causes :

- Les faiblesses des horloges (section 2.7.1) sont à prendre en compte. Si la mesure du RTT ne nécessite pas de synchronisation de celles-ci (sauf pour déterminer le temps T de la mesure), elle demande par contre un peu de stabilité. Une dérive ("*skew*") très forte ou, plus probable, un changement brusque de l'heure par l'opérateur ou par un logiciel peut fausser un résultat. De même, la résolution des horloges peut être trop mauvaise pour la rapidité des réseaux modernes.
- La définition du RTT est lié au concept de « mise sur le câble » des bits (section 2.7.2). Un programme typique n'a pas accès à l'information sur l'instant exact où le bit est envoyé. Le temps qu'il mesure est celui de la machine, pas celui de la carte réseau. (Mon programme echoping <<http://echoping.sourceforge.net/>>, qui mesure le RTT, n'essaie même pas de déterminer ce "*wire time*", je ne pense pas qu'il existe une méthode portable sur Unix pour cela.)
- Le temps de réaction de la machine visée est souvent difficile à prédire (section 2.7.3). Les routeurs Cisco, par exemple, ne génèrent pas la réponse aux paquets ICMP echo sur la carte ("*fast path*") mais dans leur processeur généraliste, bien plus lent. Avec de telles machines, le temps mesuré par ping est largement dominé par le temps de réponse du routeur et pas par le réseau proprement dit. Sur une machine Unix, ICMP est mis en œuvre dans le noyau, ce qui fournit certaines garanties, notamment que le "*swapping*" n'interviendra pas. Mais cela ne garantit pas, par exemple, qu'une surcharge du processeur de la machine ne va pas ajouter un délai inattendu. Si le "*Type-P*" est TCP avec le port 80, comme HTTP est typiquement mis en œuvre en mode utilisateur, les incertitudes sont bien plus grandes ("*swapping*", temps de réponse de l'application - par exemple Apache, ...)

Une fois ces problèmes surmontés, il faut indiquer à l'utilisateur le résultat des mesures. C'est l'objet de la section 2.8 qui insiste sur la **nécessité** de transmettre au dit utilisateur toutes les informations permettant d'interpréter les données : le "*Type-P*" (pour echoping, il faut utiliser l'option `-v` et encore, l'affichage du "*Type-P*" n'est pas vraiment clair), la façon dont une réponse en retard est considérée comme perdue, les résultats de l'étalonnage et le chemin suivi par les paquets (à noter que cette information n'est en général pas accessible au programme de mesure, à part éventuellement l'interface réseau de sortie).

Une fois cette première métrique établie, le RFC, dans sa section 3, définit un échantillonnage de cette mesure, selon une distribution de Poisson. Cette seconde métrique, définie de 3.1 à 3.4, a deux paramètres supplémentaires, le temps de fin de la mesure et le taux d'envoi des paquets (le lambda de la distribution de Poisson).

La section 3.5 discute du choix du paramètre lambda, en citant le RFC 2330. En gros, comme il s'agit d'une mesure **active**, plus lambda est élevé et plus la mesure est précise, mais plus on a perturbé le réseau avec tous ces paquets de test.

Munis de cette deuxième métrique, le RFC cite quelques statistiques intéressantes (section 4) qui peuvent être obtenues :

- Le Xème centile (section 4.1), c'est-à-dire la valeur du RTT pour laquelle X % des mesures tombent en dessous. Ainsi, le 90ème centile est la valeur du RTT telle que 90 % des mesures donnaient une RTT plus faible. Le 50ème centile est presque équivalent à la médiane <<https://www.bortzmeyer.org/mediane-et-moyenne.html>>, décrite en section 4.2.
- Le délai d'aller-retour minimum (section 4.3) qui reflète le cas idéal (réseau peu ou pas chargé, machine distante peu ou pas chargée et qui répond donc aux mieux de ses capacités).

Enfin, la section 5 est consacrée aux questions de sécurité. Comme toutes les mesures actives, celles du RTT doivent garder à l'esprit la nécessité de ne pas surcharger le réseau (ne pas utiliser l'option `-f` de ping). Mais ce point concerne la sécurité du réseau vis-à-vis des mesures. Il y a aussi un problème de sécurité des mesures vis-à-vis du réseau. En effet, les équipements sur le trajet peuvent fausser la mesure en retardant ou en donnant la priorité à certains paquets. En cas de mesure d'un réseau franchement hostile, il peut être nécessaire de recourir à des techniques cryptographiques pour préserver l'intégrité des mesures.

Un protocole utilisant cette métrique a été normalisé, TWAMP (RFC 5357).