RFC 6576 : IPPM standard advancement testing

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 26 mars 2012

Date de publication du RFC : Mars 2012

https://www.bortzmeyer.org/6576.html

Pour déterminer si un protocole de communication est bien décrit dans les RFC qui le normalisent, l'IETF a une méthode simple, tester si les différentes mises en œuvre de ce protocole interopèrent proprement entre eux (par exemple, on vérifie qu'un client HTTP parle au serveur HTTP avec succès). Ce genre de tests est maintenant fréquent, et sert de base à l'avancement des RFC sur le chemin des normes (de simple proposition au statut de norme complète). Mais, lorsqu'un RFC spécifie, non pas un protocole mais une **métrique**, c'est-à-dire une grandeur mesurable, comment fait-on pour vérifier que le RFC est correct et est bien implémenté ? Ce nouveau RFC de l'inépuisable groupe de travail IPPM <http://tools.ietf.org/wg/ippm> propose une solution : on mesure le même phénomène avec les différentes mises en œuvre et les chiffres obtenus doivent être identiques (ou suffisamment proches pour être statistiquement équivalents). Cette méthode permettra aux RFC décrivant les métriques d'avancer sur le chemin des normes : si les résultats sont les mêmes, on considérera que cela semble indiquer que la métrique est définie clairement et sans ambiguité. (Si les résultats sont différents, cela pourra être dû à une bogue dans l'un des systèmes, ou à un RFC confus, qui n'a pas été compris de la même façon par tous les développeurs.)

Le chemin des normes, qui comportait trois étapes du temps du RFC 2026 [1], n'en a plus que deux depuis le RFC 6410. Un des critères d'avancement d'une norme est l'interopérabilité entre des mises en œuvre indépendantes. Cette interopérabilité devait autrefois se démontrer par des tests explicites (RFC 5657) mais, aujourd'hui, le simple déploiement massif suffit (nul besoin de tester HTTP pour voir tous les jours dans l'Internet que clients et serveurs se parlent...). Tout cela est très bien pour les protocoles de communication. Mais pour des métriques ? On peut avoir plusieurs mises en œuvre indépendantes, qui sont largement déployées, sans que cela ne prouve qu'elles soient cohérentes entre elles. La section 5.3 du RFC 5657 prévoit explicitement le cas des normes qui ne sont pas des protocoles de communication mais ne fournit guère de solutions. L'analyse du groupe IPPM est qu'il est quand même possible de tester explicitement des définitions de métriques et que l'identité (aux variations statistiques près) des

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc2026.txt

chiffres obtenus par deux mesures du même phénomène réseau est l'équivalent de l'interopérabilité. (Notez qu'IPPM a aussi quelques protocoles à son actif, comme ceux du RFC 4656 et RFC 5357, dont l'interopérabilité peut être testée par des moyens classiques.)

Pour les métriques, l'idée va donc être de générer un phénomène réseau contrôlé, avant de le faire mesurer par les différents systèmes. (Mesurer un phénomène réel, non contrôlé, est plus difficile car cela implique que les différentes mesures soient parfaitement synchronisées, pour observer exactement la même chose. Tous les systèmes de mesure n'ont pas forcément ce mécanisme de synchronisation.) Un exemple complet figure dans l'annexe A, avec la métrique « délai d'acheminement », et c'est une lecture très recommandée si tout cela vous semble trop abstrait.

La section 2 détaille cette idée de base. Inspirée du RFC 2330, elle part du principe qu'une mesure doit être **reproductible** (on mesure le même phénomène deux fois, et les résultats doivent être identiques, aux fluctuations statistiques près). En pratique, mesurer exactement « le même phénomène » implique de prêter attention à de nombreux détails, par exemple à la stabilité des routes (sur l'Internet, le délai d'acheminement entre deux points peut varier brusquement, si BGP recalcule les routes à ce moment). D'autre part, si la métrique a des options, les deux mesures doivent évidemment utiliser les mêmes options. Et, en raison des variations du phénomène, l'échantillon mesuré doit être assez grand pour que la loi des grands nombres rende ces variations négligeables. La section 2 dit clairement qu'on n'envisage pas de mesurer des singletons (par exemple, pour le délai d'acheminement, le délai d'un seul paquet).

Une fois ces précautions comprises, comment détermine-t-on la conformité d'une mesure à la spécification de la métrique (section 3) ? D'abord, dans la grande majorité des cas, les critères de comparaison vont dépendre de la métrique, et doivent donc être définis spécifiquement pour chaque métrique (l'annexe A fournit un exemple complet). Ensuite, on va comparer deux mesures (qui peuvent avoir été faites dans le même laboratoire, ou bien dans des endroits différents.) La méthode statistique de comparaison recommandée est le test d'Anderson-Darling K. Entre deux mesures d'un même phénomène, on veut un niveau de confiance d'Anderson-Darling K d'au moins 95 %.

Selon l'environnement de test, il va falloir faire attention à des techniques comme la répartition de charge qui peuvent envoyer le trafic par des chemins distincts, une bonne chose pour les utilisateurs, mais une plaie pour les métrologues (cf. RFC 4928).

Compte-tenu de tous les pièges que détaille la section 3 (le gros du RFC), il est nécessaire de documenter soigneusement (RFC 5657 et section 3.5 de notre RFC) l'environnement de test : quelle était la métrique mesurée, quelle était la configuration de test, tous les détails sur le flot de paquets (débit, taille, type, etc).

Si on trouve des différences entre deux mesures de la même métrique, sur le même phénomène, alors il faudra se lancer dans un audit soigneux pour déterminer où était la racine du problème : bogue dans une implémentation ou manque de clarté de la spécification, cette seconde raison étant un bon argument pour ne pas avancer la métrique sur le chemin des normes et pour l'améliorer.

Si vous voulez en savoir plus que le test d'Anderson-Darling, il est documenté dans le rapport technique 81 de l'Université de Washington, par Scholz, F. et M. Stephens, « "*K-sample Anderson-Darling Tests of fit, for continuous and discrete cases*" <http://www.stat.washington.edu/research/reports/1986/tr081.pdf> ». Le test Anderson-Darling K est faisable en R : voir le paquetage « "*adk : Anderson-Darling K-Sample Test and Combinations of Such Tests*" <http://cran.r-project.org/web/packages/adk/> ». Une implémentation d'Anderson-Darling K en C++ figure dans l'annexe B du RFC.

L'annexe A contient un exemple complet de test d'une métrique, en l'occurrence le délai d'acheminement d'un paquet, tel que défini par le RFC 2679 (le test complet sera documenté dans un futur RFC, actuellement l'"*Internet-Draft*" « "*Test Plan and Results for Advancing RFC 2679 on the Standards Track*" <http://tools.ietf.org/id/draft-morton-ippm-testplan-rfc2679-01> »). Par exemple, on met le délai d'attente maximum (une option de la métrique) à 2 secondes, on mesure un flot de paquets à qui on impose un délai d'une seconde, grâce à un dispositif qui introduit délibérement des problèmes réseaux (un "*impairment generator*" ; ce genre de traitement peut se faire sur un Unix courant <https://www.bortzmeyer.org/tester-protocoles-reseaux-avec-pertes.html>), on vérifie qu'ils sont tous comptés, on passe le délai imposé à 3 secondes, et on vérifie qu'on n'obtient plus aucun résultat (la grandeur « délai d'acheminement » est indéfinie si le paquet n'arrive pas avant le délai maximum, RFC 2679, section 3.6).

Ce n'est évidemment pas le seul test. L'annexe A propose également de tester le délai d'acheminement (qui va jusqu'à la réception du dernier bit du paquet, RFC 2679, section 3.4) en faisant varier la taille des paquets, mettons à 100 et 1500 octets. Les implémentations testées doivent mesurer la même augmentation du délai d'acheminement (qui ne dépendra que du temps de transmission supplémentaire, que devra supporter le plus gros des paquets). Autre test, mesurer le délai, puis imposer un retard N, les différentes mesures doivent voir la même augmentation du délai.

Le premier RFC à avoir testé cette démarche a été le RFC 2679, dont le rapport d'avancement a été publié dans le RFC 6808.