RFC 7141 : Byte and Packet Congestion Notification

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 27 février 2014

Date de publication du RFC : Février 2014

https://www.bortzmeyer.org/7141.html

Lorsqu'un routeur de l'Internet doit stocker des paquets en attente d'envoi au routeur suivant, et que la file d'attente est pleine ou menace de l'être, que doit-il faire ? Il y a très longtemps, le routeur attendait le remplissage de la file et jetait les paquets arrivés ensuite (on appelle cette « méthode » le "*drop-tail*"). De nos jours, les routeurs ont en général une approche plus « intelligente », souvent nommée AQM ("*Active Queue Management*") qui consiste à utiliser divers algorithmes pour mieux réagir à cette situation, et limiter ainsi les risques de congestion. Il existe de nombreux algorithmes d'AQM comme, par exemple, RED (la section 1.1 en donne une liste : le RFC utilise en général RED comme exemple). Ce nouveau RFC ne décrit pas encore un algorithme de plus mais expose trois bonnes pratiques qui devraient être suivies par tous les algorithmes d'AQM :

- Les protocoles de transport devraient tenir compte de la taille des paquets, lorsqu'ils réagissent à des signaux indiquant l'arrivée de la congestion,
- Par contre, les routeurs ne devraient **pas** tenir compte de cette taille lorsqu'ils décident d'agir parce que les tuyaux se remplissent,
- Dans le cas spécifique de l'algorithme RED, son mode « octet », qui privilégie les petits paquets en cas de congestion, ne devrait **pas** être utilisé (contrairement à ce que recommandait le RFC 2309 [1], section 3, RFC depuis remplacé par le RFC 7567).

À noter que les actions prises par les routeurs lorsqu'ils détectent l'approche de la congestion sont variées : autrefois, il n'y en avait qu'une, laisser tomber des paquets mais, aujourd'hui, un routeur peut faire d'autres choix comme de marquer les paquets (ECN - RFC 3168 ou PCN - RFC 5670).

Le problème de la prise en compte (ou pas) de la taille des paquets lorsqu'on détecte ou réagit à la congestion a occupé les chercheurs depuis de nombreuses années. Les mécanisme pré-AQM, genre "*drop-tail*" (lorque la file d'attente est pleine, on jette les messages qui arrivent, ceux qui auraient normalement été mis à la fin de la file), traitaient tous les paquets pareils. Or, il y a des gros et des petits paquets et, en prime, les ressources réseau peuvent être pleines en nombre d'octets ("*bit-congestible*", maximum de b/s atteinte, la limite importante pour les réseaux lents, genre radio) ou en nombre de paquets ("*packet-congestible*", maximum de paquets/s atteint, un indicateur beaucoup moins cité que le précédent mais souvent crucial, par exemple pour la fonction de recherche d'une route dans un routeur). À l'époque, on ne se posait pas trop de questions. Aujourd'hui, on se demande :

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc2309.txt

— Lorsqu'on **mesure** la congestion, faut-il mesurer la longueur de la file d'attente en octets, en paquets (rappelez-vous qu'il y a des gros et des petits paquets, donc cela ne donnera pas le même résultat que l'indicateur précédent) ou en temps d'attente ? Le consensus actuel est que le mieux est de compter en temps d'attente mais, si cela n'est pas possible, de compter avec l'indicateur pour lequel le réseau est effectivement limité.
— Lorsqu'un routeur a détecté et **signale** la congestion (en marquant ou en jetant des paquets), doit-il tenir compte de la taille des paquets, et, par exemple, jeter/marquer préférentiellement les gros ? Il y avait nettement moins de consensus sur cette question, que notre RFC tranche en disant que, non, le routeur devrait ignorer la taille des paquets. Avant ce RFC, les discussions qui avaient mené au RFC 2309 (et qui n'ont pas forcément été documentées dans le RFC 2309, voir ce résumé <http://www-nrg.ee.lbl.gov/floyd/REDaveraging.txt>) recommandaient de marquer/jeter en fonction du nombre d'octets et donc de privilégier les petits paquets. Cela épargnait les importants paquets de contrôle (qui sont en général petits). L'une des raisons du changement de stratégie est que ce n'est pas au routeur de se préoccuper du débit de TCP, ce devrait être à TCP de gérer cela (RFC 2914).
— Lorsque les protocoles de transport comme TCP **agissent** lorsqu'ils voient un signal de congestion, doivent-ils tenir compte de la taille des paquets ? Il y avait nettement moins de consensus sur cette question, que notre RFC tranche en disant que, oui, TCP devrait prendre en compte la taille des paquets (contrairement au routeur).

L'algorithme RED, tel que décrit dans le RFC 2309, a deux modes, nommés "*packet mode*" et "*byte mode*" et le second a un effet pervers, encourager les protocoles des couches supérieures à utiliser des petits paquets, qui ont moins de chances d'être jetés. (Notre RFC note que ce mode ne semble heureusement plus utilisé aujourd'hui.) La section 1.2 contient un tableau qui indique la différence de comportement pour deux flots de 48 Mb/s, un comportant uniquement des paquets de 60 octets (donc 100 kp/s) et un autre des paquets de 1 500 octets (donc 4 kp/s). En mode paquet (probabilité de perte de paquet constante), RED maintient le débit (certes, il dégomme davantage de petits paquets mais ceux-ci transportent moins). En mode octet (probabilité de perte d'un octet constante), le débit diminue davantage pour le flot ayant les gros paquets.

TCP, aujourd'hui, ne tient typiquement pas compte de la taille des paquets lorsqu'il prend des décisions comme de diminuer la taille de la fenêtre. En pratique, cela n'a jamais été un problème car, dans une connexion TCP donnée, tous les paquets dans une direction donnée ont la même taille. Mais, dans le futur, cela pourra changer et il est donc utile de rappeler aux auteurs de mises en œuvre de TCP qu'il est au contraire recommandé de considérer la congestion comme plus sérieuse lorsque des gros paquets sont perdus.

Est-ce que le système est limité en nombre de paquets/s ("*packet-congestible*") ou en nombre de bits/s ("*bit-congestible*") ? Tout réseau comporte des composants des deux types. Par exemple, si une machine a des tampons d'entrée-sortie de taille variable, ces tampons seront une ressource "*bit-congestible*" (trop de bits et toute la mémoire sera utilisée). S'ils sont de taille fixe, on sera plutôt "*packet-congestible*". Si tout est bien conçu, les deux limites devraient être approchées à peu près en même temps. En pratique, les ressources Internet sont en général actuellement limitées en bits/s (cf. RFC 6077, section 3.3). Il y a aussi des cas plus complexes par exemple un pare-feux à état qui serait limité en nombre de flots simultanés gérés et serait donc "*flow-congestible*".

La section 2 du RFC reprend les recommandations concrètes résumées plus tôt. (Elles sont à nouveau résumés dans la section 7 donc, si vous êtes pressés, vous pouvez vous contenter de lire cette section 7, qui est la conclusion du RFC.) D'abord, la mesure de la taille de la file d'attente. On l'a vu, l'idéal est lorsque cette mesure se fait en secondes (le temps que passera le paquet dans la file, qui peut être élevé en raison du "*bufferbloat*"). Mais on ne peut mesurer de manière exacte ce temps que lorsque le paquet quitte la file d'attente, ce qui est trop tard. Donc, en pratique, on mesure en général la file en octets ou en paquets. La première méthode est meilleure si la file d'attente mène à une ressource "*bit-congestible*", la seconde est meilleure si la ressource en aval est "*packet-congestible*". Ainsi, une file d'attente menant à

un lien réseau typique se mesurera en octets, une file d'attente menant au moteur de filtrage d'un pare-feu se mesurera plutôt en paquets. Traduit en vocabulaire RED, cela veut dire de choisir le mode octet pour les mesures. Et le RFC déconseille de rendre ce choix configurable, car le fait d'être "*bit-congestible*" ou "*packet-congestible*" est une propriété inhérente à la file d'attente, et ne dépend pas des opinions de l'administrateur système.

Ensuite, la notification de la congestion. Qu'elle se fasse par la méthode traditionnelle d'IP (jeter le paquet) ou bien par les méthodes plus douces comme l'ECN, elle ne devrait **pas** dépendre de la taille du paquet. Traduit en langage RED, cela veut dire d'utiliser le mode paquet pour la notification de la congestion (alors que le RFC recommandait l'autre mode, le mode octet, pour la mesure de la file d'attente, attention à ne pas mélanger les deux). Le but principal est de ne pas encourager les flots à multiplier les petits paquets. L'étude des implémentations publiée dans l'annexe A indique que tout le monde fait déjà comme cela, donc le RFC ne fait que documenter l'existant.

Et il reste le cas de la réaction à la congestion. Des paquets manquent, ou bien arrivent avec un bit ECN. Que doivent faire TCP ou les autres protocoles de transport ? Ils devraient considérer l'≪ intensité ≫ de la congestion comme étant proportionnelle à la taille des paquets. Comme si l'indication de la congestion portait sur chaque octet du paquet. Notez que le RFC ne dit pas ce que TCP doit faire mais comment il doit aborder le diagnostic. Ensuite, les actions éventuelles doivent être prises par TCP, pas par le réseau. Si, par exemple, on souhaite que deux flots TCP utilisant des paquets de taille différentes doivent tourner au même débit (mesuré en b/s), c'est TCP qui doit s'en occuper, et ne pas compter que le réseau ajustera les pertes de paquets pour cela. Même chose pour les paquets de contrôle (un paquet `SYN` par exemple).

C'est bon ? Vous avez bien compris ce que le RFC recommandait ? Alors, maintenant, pourquoi ? C'est le rôle de la section 3 que de justifier ces recommandations. D'abord, comme indiquer plus haut, ne pas donner des raisons à une implémentation de privilégier les petits paquets. Comme noté dans le RFC 3426 (ou le papier de Gibbens et Kelly <`http://www.statslab.cam.ac.uk/~frank/evol.html`>), toujours penser aux effets pervers des bonnes intentions. Favoriser les paquets de petite taille peut sembler raisonnable, mais cela encourage les protocoles de transport à réduire la taille des paquets, donc à augmenter la charge des routeurs. Pire, cela ouvre une voie à certaines attaques par déni de service (en envoyant beaucoup de petits paquets, on peut empêcher les gros de passer).

Ensuite, une des motivations pour l'ancienne recommandation de privilégier les petits paquets venait du fait que les paquets de contrôle (qu'il faut essayer de favoriser) sont en général petits (c'est le cas des paquets TCP `SYN` et `ACK` mais aussi des paquets de certaines applications par exemple les requêtes DNS et même les `GET` HTTP). Mais l'inverse n'est pas vrai, un petit paquet pouvant n'avoir pas de rôle dans le contrôle. Là encore, favoriser aveuglément tous les petits paquets encouragerait les protocoles des couches supérieures à tout découper en petits paquets, ce qui n'est pas le but recherché.

Une autre raison de donner la priorité aux petits paquets était d'égaliser le débit de deux flots TCP qui enverraient les données dans des segments de taille différente (pour deux flots, TCP maintient le même nombre de segments donc on a plus de débit avec des segments plus grands). Mais, même si on estime ce but souhaitable, il ne devrait pas être réalisé dans les routeurs, qui ne connaissent pas forcément le protocole de transport utilisé (même entre deux TCP, les résultats d'un paquet perdu vont dépendre de l'algorithme utilisé, par exemple NewReno - RFC 5681 - vs. Cubic), mais dans les protocoles de transport, qui peuvent toujours ajuster leur politique comme ils veulent.

Il ne faut pas oublier non plus que le déploiement d'AQM est incomplet : certains routeurs font de l'AQM, d'autres pas. Les protocoles de transport doivent donc de toute façon gérer leur propre politique, ils ne peuvent pas être sûrs que le réseau le fera pour eux. C'est une variante de l'argument classique en faveur d'un contrôle de bout en bout (cf. J.H. Saltzer, D.P. Reed et David Clark, ≪ "*End-To-End*

Arguments in System Design" <http://web.mit.edu/saltzer/www/publications/endtoend/endtoend.pdf> ») : ne faites pas dans le réseau ce que vous pouvez faire aux extrémités.

Une critique plus détaillée des anciennes recommandations, notamment à propos de RED, figure en section 4. Très intéressante lecture pour les étudiants qui veulent comprendre la congestion et les moyens de la combattre. Le papier originel de RED (Floyd et Van Jacobson, « "*Random Early Detection (RED) gateways for Congestion Avoidance*" <http://www.icir.org/floyd/papers/red/red.html> ») introduisait les deux modes dont j'ai parlé, mode octet et mode paquet. Mais il ne recommandait pas un mode spécifique. Quand RED est devenu la recommandation officielle dans le RFC 2309, cinq ans plus tard, il n'y avait pas encore une recommandation claire (depuis, RED a perdu son statut, cf. RFC 7567). C'est dans les échanges de courrier cités plus haut <http://www-nrg.ee.lbl.gov/floyd/REDaveraging.txt> que s'était forgée petit à petit l'idée qu'il fallait distinguer la **mesure** de la congestion (en octets ou en paquets) et l'**action** (même probabilitité d'abandon pour tous les paquets, ou bien une probabilité qui dépend de leur taille).

La mesure de la congestion n'est pas triviale. Par exemple, un routeur peut avoir une file d'attente "*packet-congestible*" (par exemple parce que le routeur utilise un tampon de taille fixe par paquet en attente) mais qui donne accès à une ressource "*bit-congestible*". Que choisir dans ce cas ? Notre RFC dit qu'il faut quand même compter en octets : c'est la congestion de la ressource la plus en aval qu'il faut compter. Cette règle simple à l'avantage de marcher aussi si la file d'attente est plus compliquée, se servant de plusieurs jeux de tampons (des petits, des grands...), chaque jeu étant d'une taille donnée (ce que Cisco nomme "*buffer carving*").

Et s'il n'y a pas de file d'attente à proprement parler ? Par exemple, l'alimentation électrique d'un engin qui communique par radio est en général "*bit-congestible*" car la consommation électrique dépend du nombre de bits transmis. Mais il n'existe pas de file d'attente des paquets attendant que la batterie se remplisse. En fait, AQM n'impose pas une file d'attente. D'autres moyens que la longueur de cette file permettent de mesurer la congestion, comme expliqué dans « "*Resource Control for Elastic Traffic in CDMA Networks*" <http://www.ics.forth.gr/netlab/publications/resource_control_elastic_cdma.html> » dans le cas de l'exemple de l'engin sans fil.

Le problème d'une discrimination envers les grands paquets n'est pas spécifique aux mécanismes « avancés » d'AQM. Le bête et traditionnel "*tail-drop*" (jeter les nouveaux paquets entrants quand la file est pleine) tend à rejeter surtout les grands paquets (s'il y a un peu de place, on n'acceptera que les petits paquets : donc, dès que la file d'attente « avance », les petits paquets la rempliront avant qu'elle n'ait eu le temps de se vider assez pour un grand paquet). Même chose lorsqu'on a plusieurs jeux de tampons et qu'on autorise les petits paquets à occuper les grands tampons.

Et du côté des protocoles de transport, qu'est-ce qui a été fait ? Suivant la ligne du RFC 5348, le protocole TFRC-SP, spécifié dans le RFC 4828 vise à favoriser les petits paquets mais au bon endroit, dans le protocole de transport aux extrémités, pas dans les routeurs.

Cela ne résout pas le problème des paquets de contrôle, qui sont en général de petite taille (mais rappelez-vous que l'inverse n'est pas vrai). Une solution possible est de rendre TCP plus solide face à la perte de paquets de contrôle. Cela a été décrit dans les RFC 5562 et RFC 5690. Ces deux RFC disent d'ailleurs qu'il vaudrait peut-être mieux que les routeurs ne laissent pas tomber les petits paquets. Mais notre RFC 7141 dit au contraire qu'il faut résoudre le problème aux extrémités, pas dans les routeurs. D'autres propositions de réforme de TCP ont été proposées, parfois assez rigolotes comme la proposition de Wischik <http://rsta.royalsocietypublishing.org/content/366/1872/1941.full.pdf+html> de dupliquer systématiquement les trois premiers paquets de toute connexion TCP, ce qui ajouterait peu de trafic (ces paquets sont petits) mais rendrait TCP bien plus résistant aux pertes

des petits paquets, ou encore une proposition similaire <http://doi.acm.org/10.1145/2486001.2486014> qui a été testée sur les serveurs de Google.

La section 5 du RFC est consacrée aux questions non répondues. Notre RFC est clair dans ses recommendations (que le routeur ne tienne pas compte de la taille des paquets mais que le transport le fasse). Mais cela ne répond pas à tout :

— Comment gérer le cas des routeurs à AQM existants, s'ils mettent en œuvre le mode octet désormais déconseillé ?
— Comment déployer les nouvelles recommandations dans les protocoles de transport ?

La première question est relativement facile car l'étude documentée dans l'annexe A montre que le mode octet a été peu déployé. On peut donc ignorer le problème. Pour la deuxième, il existe des modifications expérimentales des protocoles de transport (RFC 4828) mais rien de systématique.

Il n'y a donc pas trop de problèmes si l'Internet reste comme il est, essentiellement composés de ressources "*bit-congestible*". Mais s'il se peuplait petit à petit de ressources "*packet-congestible*" ? La question avait déjà été posée dans la section 3 du RFC 6077 mais n'a pas encore de réponse.

Ces recommandations ont-elles des conséquences sur la sécurité de l'Internet ? La section 6 de notre RFC dit que oui, dans un sens positif. En effet, l'ancienne recommandation, de jeter en priorité les gros paquets, facilitait certaines attaques par déni de service. Après tout, une des raisons de déployer AQM était d'éviter le biais en faveur des petits paquets qui était un conséquence du "*tail drop*". Le mode octet d'AQM pouvait donc être vu comme une réintroduction d'une vulnérabilité qu'AQM voulait justement éviter.

L'annexe A rend compte de l'étude de 2007 qui montre qu'aucun des vendeurs d'équipement (routeurs, pare-feux, etc) ayant répondu à l'enquête (16 répondants) ne mettait en œuvre le mode octet pour décider de laisser tomber les paquets (donc, ils n'auront rien à faire, ce RFC 7141 rend juste compte de leur pratique actuelle). À noter que la plupart des vendeurs n'ont pas voulu être identifiés (les deux exceptions sont Alcatel-Lucent et Cisco, plus Linux qui a pu être étudié sans le vendeur, puisqu'en logiciel libre). Autre sujet d'amusement, deux vendeurs n'étaient pas sûrs et devaient vérifier. Les justifications données étaient plutôt égoïstes : prendre en compte la taille des paquets avant de décider de les jeter aurait compliqué le code. Maintenant, qu'est-ce qui est déployé dans le vrai Internet ? Bien qu'il n'y ait pas d'étude à ce sujet, les auteurs du RFC pensent que bien des files d'attente sur l'Internet sont encore "*tail drop*", pré-AQM, notamment dans les équipements de bas de gamme. Les équipements qui font de l'AQM mais avec des tampons de taille fixe, comme ceux en "*tail drop*" continuent à favoriser (légèrement) les paquets de petite taille mais il est difficile de mesurer l'exacte ampleur de leur déploiement.