RFC 7098 : Using the IPv6 Flow Label for Load Balancing in Server Farms

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 17 janvier 2014

Date de publication du RFC : Janvier 2014

https://www.bortzmeyer.org/7098.html

Le champ « "*Flow label*" » dans l'en-tête des paquets IPv6 est un des mystères de l'Internet. Bien qu'il ait désormais un RFC entièrement pour lui, le RFC 6437 [1], et qu'il soit obligatoire, selon le RFC 6434, personne n'a l'air de savoir à quoi il peut vraiment servir en pratique. Lorsqu'on observe des paquets IPv6, ce champ est souvent mis à zéro. Des gens ont déjà réfléchi à des utilisations possibles et ont documenté leurs idées dans le RFC 6294. Notre nouvel RFC, lui propose un usage : faciliter la répartition de charge dans les fermes de serveurs. Comment ? Pas si vite, d'abord un petit rappel sur le "*flow label*".

La section 2 de notre RFC rafraichit la mémoire de ceux qui ont lu les RFC 2460 et RFC 6437 trop vite. Le champ "*flow label*" fait 20 bits et doit avoir une valeur fixe pour un flot donné (par exemple pour une connexion TCP donnée). Sa valeur doit être uniformément répartie parmi les valeurs possibles (et, pour des raisons de sécurité, il vaut mieux qu'elle soit imprévisible de l'extérieur, cf. le cours de sécurité de Gont <https://www.bortzmeyer.org/hacking-ipv6.html>). Si le nœud de départ n'a pas mis une valeur non nulle dans ce champ, un nœud ultérieur (par exemple un routeur) a le droit de le faire. Le "*flow label*" est à une position fixe par rapport au début du paquet, ce qui le rend facilement lisible, même à grande vitesse, contrairement à l'en-tête TCP, qui peut se situer à une distance quelconque du début du paquet <https://www.bortzmeyer.org/analyse-pcap-ipv6.html>. Comme IPv6, contrairement à IPv4, n'a pas de champ qui indique la longueur des en-têtes de couche 3 (permettant de les sauter facilement pour arriver à l'en-tête de couche 4), le "*flow label*" pourrait être un meilleur moyen d'identifier un flot donné, d'autant plus qu'il est répété dans tous les paquets (alors que, en cas de fragmentation, l'en-tête de couche 4 ne sera que dans un seul paquet). Mais c'est purement théorique : aujourd'hui, il est rare, dans la nature, de rencontrer des paquets IPv6 avec un "*flow label*" non nul (1,4 % des paquets HTTP/IPv6 entrant sur mon blog ont un "*flow label*" non nul - vu avec tshark). Vingt bits de perdus pour rien.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc6437.txt

Avant d'utiliser le "*flow label*" pour faire de la répartition de charge, la section 3 de notre RFC résume les techniques qui existent aujourd'hui pour faire cette répartition. (On peut aussi lire à ce sujet une synthèse d'un des auteurs du RFC <http://1wt.eu/articles/2006_lb/>.) Les exemples sont tous empruntés à HTTP mais la plupart des ces techniques s'appliquent à tous les protocoles. Première méthode, donc, mettre plusieurs adresses IP dans le DNS. Si le serveur faisant autorité, ou le résolveur, renvoient ces adresses dans un ordre aléatoire, cela assurera une répartition égalitaire entre toutes les adresses. (Le RFC ne mentionne pas que l'usage des enregistrement SRV du RFC 2782 résoudrait bien des limites du DNS. Mais cela dépend de clients bien faits et certains gérants de serveurs préfèrent donc contrôler tout le processus de répartition de charge.) Cette méthode était indépendante du protocole applicatif utilisé. Une méthode spécifique à HTTP serait d'avoir un relais inversé ("*reverse proxy*") en face des vrais serveurs, qui redirige vers un des serveurs. Les clients ne verraient alors qu'une seule adresse IP. Bien sûr, le relais inversé sera seul à la tâche (alors que le but était de faire de la répartition de charge) mais relayer une connexion TCP est bien moins de travail que de servir une page Web complexe. Une variante de cette technique est utilisée pour HTTPS où le répartiteur de charge est également la terminaison de la session TLS, relayant ensuite en clair le trafic jusqu'au vrai serveur.

Mais la technique la plus répandue est sans doute d'avoir un répartiteur de charge opérant en couches 3 et 4, indépendamment du protocole de couche 7. Cela peut être un boîtier fermé spécialisé, un PC ordinaire avec, par exemple IPVS, une fonction embarquée sur le routeur, etc. L'adresse IP du répartiteur est publiée dans le DNS et les clients n'ont donc rien à faire. Comme la solution précédente, elle ne dépend donc pas du comportement du client. Certains de ces répartiteurs sont sans état, relayant chaque paquet IP indépendemment, typiquement en utilisant comme clé un condensat d'informations faciles à trouver dans le paquet (comme l'adresse IP source). Et d'autres sont à état, gardant en mémoire les connexions TCP en cours, et envoyant tous les paquets d'une même connexion vers le même serveur.

Au passage, une fois la décision prise de joindre tel ou tel serveur parmi tous ceux présents dans la ferme, comment le répartiteur redirige-t-il les paquets ? Si le répartiteur et les serveurs sont sur le même LAN, changer l'adresse MAC peut être suffisant (« liaison directe »). Les paquets de retour seront alors transmis sans passer par le répartiteur, qui aura moins de travail. Autre solution, mettre chaque serveur derrière un routeur et utiliser de l'"*anycast*" (contrairement à ce que croient certains, l'"*anycast*" n'est pas spécifique à BGP). Mais le RFC ne détaille pas cette solution (l'"*anycast*" ne se marie pas bien avec la nécessité de maintenir les paquets d'une même connexion TCP épinglés au même serveur).

Troisième solution pour la transmission des paquets vers le « vrai » serveur, donner à chaque serveur sa propre adresse IP et faire un tunnel (GRE, par exemple, car c'est le plus simple et le plus facile à déboguer) vers le serveur, encapsulant le paquet destiné à l'adresse IP du service. Là aussi, le paquet de retour n'a pas besoin de passer par le répartiteur. Selon la taille des paquets utilisés pour l'encapsulation, on pourra rencontrer des problèmes de MTU. Enfin, dernier cas, le NAT où le répartiteur fait une traduction d'adresse IP vers celle du serveur. Attention, dans ce cas, il faut bien configurer le serveur pour qu'il renvoie les paquets de retour vers le répartiteur, pour que celui-ci puisse faire la traduction d'adresse dans l'autre sens. Avec cette solution, le répartiteur, et son état, devient un point de défaillance unique.

Sur toutes ces questions de répartition de charge, on peut consulter l'exposé et l'article d'Alexandre Simon aux JRES 2011 <https://2011.jres.org/archives/110/index.htm>.

Comme les relais travaillant au niveau 7 doivent de toute façon analyser tous les paquets, l'utilisation du "*flow label*" a peu de chances d'améliorer leurs performances. Donc, le reste du RFC se limite aux redirecteurs travaillant aux niveaux 3 et 4.

Qu'est-ce qui est proposé pour ceux-ci (section 4) ? **C'est le cœur de ce RFC, la recommandation pour les répartiteurs de charge d'utiliser cette étiquette des flots.**

— Si le "*flow label*" est à zéro (la grande majorité aujourd'hui), continuer comme maintenant,
— Si le "*flow label*" a été mis, comme spécifié par le RFC 6437, utiliser le couple {adresse IP source, "*flow label*"} pour la répartition de charge : un répartiteur sans état condense ce couple (RFC 6438), on utilise le condensat comme clé, et on envoie tous les paquets ayant la même clé vers le même serveur. Cela marchera même si les paquets IPv6 sont fragmentés alors qu'un redirecteur classique sans état, qui ne peut donc pas réassembler les paquets, ne pourra pas fonctionnner car il ne trouvera pas le port source dans les fragments (sauf le premier). Un répartiteur avec état dirige le premier paquet du flot vers un serveur et mémorise l'association {adresse IP source, "*flow label*"} -¿ serveur. Si les paquets IPv6 comprennent des en-têtes d'extension, l'utilisation du "*flow label*" sera nettement plus rapide <`https://www.bortzmeyer.org/analyse-pcap-ipv6.html`> que de chercher l'en-tête de transport, et cela malgré le test supplémentaire (≪ est-ce que le "*flow label*" est nul ? ≫).
— Les redirecteurs qui font du NAT vers le serveur n'ont aucun intérêt à utiliser le "*flow label*" puisqu'ils doivent de toute façon atteindre la couche transport afin de trouver les ports.

Certaines configurations pourraient empêcher un bon fonctionnement de cette méthode. Par exemple, s'il y a partage massif d'adresses derrière un CGN, les risques de collision des "*flow labels*" augmentent. Comme il n'y a guère de bonnes raisons de faire un tel partage massif en IPv6, on peut espérer que cela ne soit pas un problème. (Un rappel : en IPv6, traduction d'adresses n'implique pas partage d'adresses, cf. RFC 6296.)

Des problèmes de sécurité avec cette approche ? La section 5 note que le "*flow label*" n'est pas spécialement sécurisé (à part si on utilise IPsec) mais c'est également le cas de tous les champs utilisés par les répartiteurs de charge. Néanmoins, comme le rappelle le RFC 6437, un répartiteur paranoïaque peut vérifier que, par exemple, les paquets ne sont pas envoyés de manière déséquilibrée vers un seul serveur car un petit malin modifie les "*flow label*" en cours de route. En pratique, en raison de l'utilisation de l'adresse IP source dans la clé, d'une fonction de condensation cryptographique comme SHA-1 et, possiblement, d'un secret concaténé au couple {adresse IP source, "*flow label*"} avant sa condensation, une telle attaque ne serait pas triviale.

À noter qu'aujourd'hui, il ne semble pas qu'il existe déjà de répartiteur de charge qui mette en œuvre les préconisations de ce document. (Par exemple, apparemment rien dans l'IPVS de Linux en 3.1.7.)

Merci à Alexandre Simon pour sa relecture.