RFC 6994 : Shared Use of Experimental TCP Options

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 24 août 2013

Date de publication du RFC : Août 2013

https://www.bortzmeyer.org/6994.html

Le protocole TCP dispose d'un champ « Options » permettant d'indiquer des fonctions spécifiques. Ce champ commence par un identificateur d'option (sur un octet) et deux valeurs sont réservées pour des options expérimentales, non enregistrées. Deux, c'est trop peu, et, en prime, il y a parfois collision entre deux options expérimentales différentes. Notre RFC propose donc un nouveau mécanisme éliminant ces collisions, en ajoutant un deuxième identificateur, l'**ExID**, dans les options expérimentales.

Le champ « Options » est décrit dans le RFC 793[1]. Il a deux formes mais la plus courante est composée d'un octet indiquant l'option, d'un octet indiquant la longueur de l'option et de données spécifiques à l'option (un TLV, donc). Les options sont enregistrées à l'IANA <https://www.iana.org/assignments/tcp-parameters/tcp-parameters.xhtml#tcp-parameters-1> et sont aujourd'hui au nombre d'une trentaine. 30 sur 256 possibles, cela justifie de les allouer avec prudence (RFC 2780) et les gens qui veulent juste essayer une idée amusante, sans enregistrer leur option, n'avaient pas de solution simple avant que le RFC 4727 ne réserve les options 253 et 254 pour les expérimentations. Si vous avez une idée qui nécessite une option TCP, et que vous voulez la mettre en œuvre, prenez un de ces deux numéros et mettez-le dans vos paquets TCP, vous serez dans la légalité (le RFC 3692 dit toutefois que cela ne doit pas être activé par défaut).

Mais ces expérimentations risquent de se marcher sur les pieds entre elles. Par exemple, le RFC 6013 s'alloue l'option 253. Si vous voulez tester une option en même temps que celle du RFC 6013, il ne vous reste plus qu'un numéro, ce qui peut être insuffisant (certains systèmes nécessitent plusieurs options). À noter que cet usage d'un numéro d'option expérimental est légal (c'est fait pour) mais qu'il existe aussi des squatteurs, qui utilisent des numéros normalement réservés sans les avoir enregistrés proprement. Les options 31 et 32 avaient ainsi été squattées autoritairement, comme 76 à 78, 33, 69, 70, et 76 à 78 (ces

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc793.txt

derniers dans des produits commerciaux). Le registre IANA les note comme « *"known unauthorized use without proper IANA assignment"* ».

Plusieurs approches ont été suggérées à l'IETF pour régler ce problème. Il a été proposé d'élargir l'espace des numéros d'options (*"Internet-Draft"* `draft-eddy-tcpm-addl-exp-options`) et/ou de libéraliser les règles d'allocation, actuellement très strictes (il faut un RFC sur le chemin des normes pour réserver une option). Ce RFC 6994 utilise une autre approche : il crée un « sous-espace » à l'intérieur des options expérimentales, dans lequel on peut (mais ce n'est pas obligatoire) enregistrer sans trop de formalités une expérimentation et son identificateur. C'est désormais l'approche recommandée pour les nouvelles options expérimentales TCP (et obligatoire si l'option expérimentale est déployée sur l'Internet public).

Le principe (section 3) est de remplacer la structure {numéro d'option, longueur, valeur} par {numéro d'option, longueur, ExID, valeur} où le ExID est l'identificateur d'une expérimentation. La taille des ExID (16 ou 32 bits) permet d'en avoir beaucoup et donc d'avoir une politique d'enregistrement libérale sans craindre d'épuiser une ressource finie. L'IANA les enregistre donc sur une base toute simple : « premier arrivé, premier servi » (RFC 5226 pour les différentes options d'allocation possibles). Le registre de ces ExID est en ligne <`https://www.iana.org/assignments/tcp-parameters/tcp-parameters.xhtml#tcp-exids`> (il compte actuellement trois expérimentations dont le *"Fast Open"* <`https://www.bortzmeyer.org/tcp-ouverture-rapide.html`> TCP qui a été décrit dans le RFC 7413).

Bon, mais l'ExID fait 16 ou 32 bits, alors ? C'est subtil : les 16 premiers bits doivent être uniques pour identifier une expérimentation. Les 16 bits optionnels suivants servent de nombre magique, pour éviter le risque de collision avec des mises en œuvre de TCP qui ne connaissent pas ce RFC (ou bien l'ignorent) et mettent leurs données juste après l'octet de longueur. Si on veut économiser de la place, on choisit un ExID de 16 bits, si on veut minimiser le risque de collision, on prend 32 bits.

Que va donc faire une mise en œuvre de TCP qui rencontre ces options (section 3.2) ? Si l'option est 253 ou 254, elle doit lire l'ExID et ignorer l'option s'il s'agit d'une expérimentation qui lui est inconnue.

Mais les collisions, alors ? Elles peuvent toujours exister. Si une implémentation ancienne de TCP utilise l'option 253 et met au début des données qu'elle envoie la valeur 0x15df, et que le récepteur TCP, conforme à ce nouveau RFC 6994, croit qu'il s'agit de l'expérimentation 5599, qu'il connait, et se met à analyser l'option comme telle ? Il n'y a pas de solution magique. La section 4 de notre RFC recommande aux receveurs TCP d'être robustes face à de telles collisions, ou bien de mettre en œuvre un mécanisme de contrôle de la cohérence des données, pour rejeter les faux positifs.

Un problème classique (et non spécifique à TCP) avec les options expérimentales est qu'un jour, elles seront peut-être normalisées et qu'il faudra alors gérer une transition depuis l'option expérimentale vers l'option standard. La section 5 couvre ce cas mais ne recommande pas une stratégie particulière. Elle se contente de dire qu'envoyer les deux options (l'expérimentale et la standard) dans le même paquet est une mauvaise idée (l'espace des options est limité en TCP).

La section 6 discute des autres approches qui **auraient** pu être choisies. Par exemple, au lieu des ExID, on aurait pu marquer les options avec un identificateur de l'organisation qui dirige l'expérience, en utilisant les OUI de l'IEEE ou bien les *"Enterprise Numbers"* <`https://www.iana.org/assignments/enterprise-numbers`> de l'IANA (RFC 1155). Mais les OUI sont très chers (comme souvent à l'IEEE), plus de 1 800 dollars. Les PEN (*"Private Enterprise Numbers"*) sont gratuits mais ne sont normalement pas allouables à un individu. Et, dans les deux cas, ces nombres sont plus grands que l'ExID (les PEN n'ont pas de taille définie actuellement mais ils sont déjà trop nombreux pour 16 bits) alors que l'espace est limité dans le champ Options de TCP, puisque tout octet prend la place d'un octet utile aux données.

Si on regarde les versions de développement de Linux et de FreeBSD, on ne trouve rien dans ce dernier (`sys/netinet/tcp_input.c` dans `tcp_dooptions()`), qui ne connait pas, par défaut, les options expérimentales. Pour Linux, seul "*Fast Open*" est reconnu. Le code dans `net/ipv4/tcp_input.c`, `tcp_parse_options()` dit :

```
case TCPOPT_EXP:
        /* Fast Open option shares code 254 using a
         * 16 bits magic number. It's valid only in
         * SYN or SYN-ACK with an even size.
         */
        if (opsize < TCPOLEN_EXP_FASTOPEN_BASE ||
            get_unaligned_be16(ptr) != TCPOPT_FASTOPEN_MAGIC ||
            foc == NULL || !th->syn || (opsize & 1))
                break;
        foc->len = opsize - TCPOLEN_EXP_FASTOPEN_BASE;
        if (foc->len >= TCP_FASTOPEN_COOKIE_MIN &&
            foc->len <= TCP_FASTOPEN_COOKIE_MAX)
                memcpy(foc->val, ptr + 2, foc->len);
        ...
}
```

Où `TCPOPT_FASTOPEN_MAGIC` vaut 0xF989 (l'ExID sur 16 bits dans le registre IANA `<https://www.iana.org/assignments/tcp-parameters/tcp-parameters.xhtml#tcp-exids>`).