RFC 9146 : Connection Identifiers for DTLS 1.2

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 20 mars 2022

Date de publication du RFC : Mars 2022

https://www.bortzmeyer.org/9146.html

Ce RFC ajoute au protocole de sécurité DTLS 1.2 la possibilité d'identificateurs de connexion ("*connection ID*"), permettant de relier entre eux des paquets d'une même session DTLS, même si l'adresse IP source change. C'est une solution simple et minimale, DTLS 1.3 (RFC 9147[1]) l'utilise (avec quelques ajouts).

Dans le DTLS habituel, version de TLS qui tourne sur UDP et qui est normalisée dans le RFC 6347, lorsqu'un paquet arrive à une machine, celle-ci trouve l'association de sécurité appropriée (ce qui permet de trouver le matériel crptographique qui permettra de déchiffrer et de vérifier les signatures) en regardant le tuple {protocole, adresse IP source, adresse IP destination, port source, port destination}. Mais si la machine avec qui on correspond a changé d'adresse IP, par exemple parce qu'il s'agit d'un malinphone qui est passé de WiFi à 4G ? Ou bien si elle a changé de port car un routeur NAT a trouvé intelligent de considérer la session terminée, effaçant une entrée dans sa table de correspondance ? Dans ce cas, le paquet entrant va être considéré comme une nouvelle session, il faudra reprendre la négociation TLS, et c'est du temps perdu (la poignée de main cryptographique est coûteuse, et on souhaite amortir ce coût sur la plus longue durée possible).

Le RFC note que le problème est particulièrement sérieux pour les déploiements type Internet des Objets, où les objets peuvent se mettre souvent en sommeil pour économiser leur batterie, amenant le routeur NAT à oublier la session en cours.

Notez aussi qu'outre l'identificateur de connexion, notre RFC apporte quelques changements supplémentaires, notamment pour permettre le remplissage.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc9147.txt

La section 3 du RFC spécifie l'extension DTLS `connection_id` (numéro 54 dans le registre IANA <https://www.iana.org/assignments/tls-extensiontype-values/tls-extensiontype-values.xml#tls-extensiontype-values-1>) qui permet de spécifier des identifiants des connexions DTLS, et donc ainsi de construire une connexion qui résistera aux changements d'adresses IP et de ports. Dans son `ClientHello`, le client DTLS indiquera l'identifiant qu'il utilisera (idem pour le serveur DTLS dans son `ServerHello`). Par contre, on ne peut pas changer d'identifiant de connexion en cours de connexion (contrairement à ce que permettent, par exemple, QUIC <https://www.bortzmeyer.org/quic.html> ou, tout simplement, DTLS 1.3). Une fois l'utilisation de "*connection IDs*" négociée, les données sont envoyées dans une nouvelle structure, de type `tls12_cid` (numéro 25 dans le registre IANA <https://www.iana.org/assignments/tls-parameters/tls-parameters.xml#tls-parameter
(Pour DTLS 1.3, il faudra regarder le RFC qui lui sera consacré.) On peut ajouter des zéros avant le chiffrement, à des fins de remplissage. Voici la structure de données avant le chiffrement :

```
struct {
            opaque content[length];
            ContentType real_type;
            uint8 zeros[length_of_padding];
} DTLSInnerPlaintext;
```

Et une fois chiffrée, on envoie ça sur le réseau :

```
struct {
            ContentType outer_type = tls12_cid;
            ProtocolVersion version;
            uint16 epoch;
            uint48 sequence_number;
            opaque cid[cid_length];    // L'identifiant de
    // connexion est là.
            uint16 length;
            opaque enc_content[DTLSCiphertext.length];
} DTLSCiphertext;
```

Que se passe-t-il quand on reçoit un message pour un identifiant de connexion connu mais une nouvelle adresse IP ? Il ne faut pas lui faire une confiance aveugle (des méchants ont pu usurper l'adresse IP) et envoyer immédiatement des réponses à la nouvelle adresse IP. Il faut d'abord vérifier que le message est correctement signé, qu'il a une `epoch` plus récente que le précédent message (dans certains cas, comme l'ordre des messages n'est pas garanti, cela peut mener à ignorer un message valide), et l'application doit tester que le pair est toujours d'accord (la méthode dépend de l'application).

Les identifiants de connexion posent évidemment des questions de vie privée (section 8 du RFC). Ils doivent être en clair (puisqu'ils servent au récepteur à découvrir le matériel cryptographique de la connexion, qui servira au déchiffrement) et sont donc observables par quiconque est situé sur le trajet, ce qui améliore la traçabilité (ce qui est mauvais pour la vie privée). Et, contrairement à QUIC <https://www.bortzmeyer.org/quic.html>, on n'a qu'un identifiant, on ne peut pas en changer (c'est mieux en DTLS 1.3).