Résumé du protocole HIP

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 21 décembre 2010

https://www.bortzmeyer.org/hip-resume.html

La parution régulière de RFC sur HIP ("*Host Identity Protocol*") me donne à penser qu'un petit résumé de ce protocole est une bonne idée (ce texte est très inspiré du chapitre 3 du RFC 6079 [1]). HIP est un protocole de séparation entre l'identificateur et le localisateur <https://www.bortzmeyer.org/separation-identificateur-localisateur.html>. Avant HIP, les adresses IP servaient les deux rôles à la fois : comme localisateur, elles identifient une position de la machine sur l'Internet (pas une localisation physique <https://www.bortzmeyer.org/hostname-physical-location.html>, bien sûr), et changent donc si la machine se déplace ou si le réseau change de FAI. Comme identificateur, elles sont utilisées par les protocoles des couches supérieures, notamment TCP pour identifier une connexion en cours et, si elles changent, la connexion est cassée. Cette dualité de rôle des adresses IP rend certains problèmes comme la mobilité, le renumérotage (cf. RFC 5887) ou le "*multihoming*" très durs à traiter.

HIP, normalisé dans le RFC 7401, résoud le problème en limitant les adresses IP au rôle de localisateur. Ainsi, un paquet HIP est un paquet IP normal et les routeurs n'ont pas besoin d'être modifiés. HIP est entièrement mis en œuvre dans la machine terminale. Les identificateurs sont, eux, des clés publiques, permettant une authentification des machines (leur nom officiel est HI, pour "*Host Identifier*"). Comme ces clés sont souvent très longues, et, pire, de taille variable, HIP introduit également un condensat cryptographique des clés, le HIT ("*Host Identifier Tag*"), qui a la taille d'une adresse IPv6 et peut donc être présenté aux couches supérieures comme TCP, sans trop les traumatiser. (Sur FreeBSD, vous trouverez le HIT de votre machine sous `/etc/hip`.)

Les HIT sont rangés dans un préfixe IPv6 spécial, nommé "*Orchid*" (RFC 7343), `2001:20::/28`, avant d'éviter toute collision avec les adresses IPv6 « normales ».

Pour établir une connexion, HIP utilise un échange de quatre paquets (comme SCTP, alors que TCP n'en utilise que trois). Pour envoyer ces paquets, le pair à l'initiative de la connexion doit connaître

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc6079.txt`

le localisateur (l'adresse IP) du répondeur. Il peut la trouver dans le DNS (RFC 8005) ou bien via un serveur de rendez-vous (RFC 8004). En pair-à-pair, une DHT pourrait être un bon moyen de résoudre un identificateur en localisateur, et le RFC 6537 explore cette piste. (Sur FreeBSD, on peut même mettre HIT et adresse - identificateur et localisateur - du pair dans `/etc/hosts`.) HIP dispose également d'un mécanisme pour pouvoir fonctionner à travers les NAT (RFC 5770). Une fois la connexion établie, les localisateurs peuvent changer (RFC 5206), la connexion continue.

HIP dispose d'une forte sécurité : protection contre les usurpations d'identificateur par le fait que ceux-ci sont une clé cryptographique et que les messages sont signés, protections contre les DoS au moment de l'établissement de la connexion (une faiblesse traditionnelle de TCP lorsqu'il est utilisé seul). Le « certificat » que représente cette clé est auto-signé par défaut (et est donc accepté par TOFU "*Trust On First Use*" comme dans SSH) mais, si on veut avoir encore plus de sécurité, on peut tout à fait avoir un serveur central qui alloue les identificateurs et les lie à des identificateurs des applications. HIP n'a pas qu'un seul modèle de sécurité pour authentifier les pairs.

Qu'est-ce que HIP change pour les applications ? Une application traditionnelle peut tout à fait utiliser HIP (cf. RFC 5338) mais une API standard figure dans le RFC 6317 pour celles qui veulent aller plus loin.

Il existe des mises en œuvres de HIP pour FreeBSD <`http://hip4inter.net/`> (le développement a stoppé en 2008) et Linux <`http://www.openhip.org/`> mais aucun des deux ne semble proche d'une intégration dans le système officiel (pour FreeBSD, rien n'est prévu). Le projet OpenHIP <`http://www.openhip.org/`> adapte également des logiciels comme Wireshark pour qu'ils aient un support HIP. InfraHIP <`http://infrahip.hiit.fi/`> travaille également à l'infrastructure HIP et à des implémentations <`http://infrahip.hiit.fi/hipl/manual/index.html`>. Ils ont réalisé une bonne explication de HIP en une page <`http://infrahip.hiit.fi/index.php?index=how`> qui concurrence sérieusement cet article. Un compte-rendu des expériencs pratiques avec HIP se trouve dans le RFC 6538.