L'IETF, l'espionnage et les protocoles Internet

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 2 novembre 2013

https://www.bortzmeyer.org/ietf-securite-espionnage.html

Du 4 au 8 novembre 2013 se tient à Vancouver la 88ème réunion <https://www.ietf.org/meeting/88/index.html> de l'IETF. Elle verra notamment un grand nombre de discussions sur l'espionnage de masse auquel se livrent des organisations comme la NSA, et aux moyens d'adapter la sécurité de l'Internet à ce type de menaces.

Le terme important est **de masse**. La réalité de l'espionnage sur l'Internet est connue depuis longtemps, même si beaucoup préféraient faire l'autruche. Mais même les experts en sécurité dont tout le monde se moquait, en les traitant de « paranoïaques », n'osaient pas forcément envisager l'ampleur de l'espionnage mené par des organisations puissantes et placées en dehors de tout contrôle démocratique, comme la NSA. Grâce aux révélations d'Edward Snowden, on sait désormais que les paranoïaques ne l'étaient pas assez. Il ne s'agit pas de combattre un "*sniffer*" isolé qui écoute de temps en temps les communications (contre ce type d'attaques, l'IETF a déjà une large panoplie de techniques). Il s'agit de faire face à un attaquant très puissant, qui peut se permettre d'écouter beaucoup de gens, tout le temps. Ce type d'attaques, qui a déjà un nom en jargon IETF, le "*pervasive monitoring*", nécessite de repenser la sécurité de l'Internet.

Bien sûr, le problème est surtout politique, et c'est sur le champ politique qu'il va essentiellement se jouer. La technique seule ne peut pas grand'chose contre un attaquant qui contrôle autant de leviers. Mais, à une réunion IETF, il est normal qu'on se pose la question « Que peut faire l'IETF ? » Il y a deux axes importants :

— Défendre les normes IETF contre les attaques visant à affaiblir délibérement leur sécurité pour faciliter l'écoute. Parmi les révélations de Snowden, en effet, se trouvait la preuve que des processus de normalisation étaient subvertis par la NSA pour lui faciliter la tâche (notamment l'affaire <http://arstechnica.com/security/2013/09/the-nsas-work-to-make-crypto-worse-and-bett> du générateur aléatoire du NIST). Faut-il changer les processus IETF pour diminuer ce risque ?
— Rendre l'Internet plus résistant à l'espionnage.

Le premier axe a déjà été évoqué, par exemple par Jari Arkko <http://www.arkko.com/>, président de l'IETF, dans un exposé au RIPE à Athènes <https://ripe67.ripe.net/presentations/185-RIPE67-Per pdf> expliquant qu'il n'y avait sans doute pas trop de risque de subversion d'un processus de normalisation IETF. Contrairement à des organismes gouvernementaux comme le NIST, qui ne peut rien refuser à la NSA, l'IETF a un travail complètement ouvert, les normes sont relues par des tas de gens, peu diplomates et qui n'hésitent pas à poser des questions lorsque quelque chose n'est pas clair, ou à râler lorsque la sécurité semble en danger. Il serait difficile, dans une telle maison de verre, de glisser des modifications affaiblissant la sécurité.

Le second axe est un gros travail, déjà baptisé d'un nom de code IETF, « "*Internet hardening*" ». Il s'agit de durcir l'Internet, de le rendre moins aisé à écouter (ce qui voudra sans doute dire aussi, moins aisé à utiliser <https://www.bortzmeyer.org/crypto-debug.html>). Le projet a été exposé dans un article d'Arkko <http://www.ietf.org/blog/2013/10/plenary-on-internet-hardening/ >. À Vancouver, il fera l'objet notamment de la réunion plénière technique. Celle-ci, qui se tiendra le mercredi 8 novembre et sera diffusée en ligne <http://www.ietf.org/live/>, verra les exposés suivants :

- Bruce Schneier expliquera ce qu'on sait et ce qu'on ne sait pas sur la surveillance généralisée. Malgré les révélations de Snowden, nous n'avons pas encore tous les détails et nous devons définir des techniques contre des attaques que nous ne connaissons pas toutes.
- Brian Carpenter rappellera que l'IETF a déjà travaillé sur des sujets liés à l'espionnage et à la vie privée, par exemple dans les RFC 1984 [1] et RFC 2804 (où l'IETF dit clairement qu'il ne faut **pas** architecturer le réseau pour faciliter les écoutes, contrairement à ce que fait l'UIT avec son projet NGN qui inclut un volet LI - "*Legal Interception*"), le RFC 3365 (sur l'importance de prendre en compte la sécurité dans les protocoles Internet), et le RFC 6973, qui est le premier à décrire une politique globale de protection de la vie privée dans les protocoles IETF. Je ne sais pas si Carpenter en parlera mais les RFC ne sont pas tous aussi honorables et il faudrait peut-être aussi mentionner le RFC 3924...
- Stephen Farrell résumera le travail du groupe informel PERPASS <https://www.ietf.org/ mailman/listinfo/perpass>. Ce groupe n'est pas un groupe de travail officiel de l'IETF car ceux-ci ont toujours des tâches bien précises, avec un calendrier. PERPASS a une activité plutôt exploratoire, regarder ce qu'on peut faire pour améliorer la résistance de l'Internet à l'espionnage.

Le groupe PERPASS a déjà vu passer plusieurs documents <http://datatracker.ietf.org/ doc/search/?name=perpass&activedrafts=on&sort=>. Ainsi, , de Hannes Tschofenig décrit les menaces. Il différencie les attaques exploitant une faille du protocole, celles exploitant une faiblesse de la cryptographie, celles exploitant une bogue dans une mise en œuvre, et celles enfin qui s'appuient sur des mauvais choix lors du déploiement. Les premières sont les seules à être complètement du ressort de l'IETF. Par exemple, la faille BEAST <https://www.bortzmeyer.org/beast-tls.html> était une faille du protocole TLS et a été réparée par le RFC 4346. Dans ce cas, la correction était assez simple. C'est plus difficile lorsque des choix d'architecture ont été faits : on ne peut pas en général les remettre en cause. Ainsi, le fait que les serveurs DNS (même ceux de la racine) reçoivent le nom complet du site auquel on veut accéder est certainement une faiblesse, pour la protection de la vie privée, mais ne peut pas être modifié, sauf à remplacer le DNS par un protocole nouveau et incompatible. Le "*draft*" cite aussi d'autres responsabilités de l'IETF : par exemple, peu de fournisseurs de voix sur IP ont déployé des mécanismes de protection cryptographiques mais l'IETF ne leur a certainement pas facilité la tâche en ayant pas moins de trois mécanismes de sécurisation concurrents. Et puis, déployer la sécurité a posteriori sur un protocole qui ne l'avait pas au début, est toujours difficile. (D'un autre côté, les protocoles qui ont la sécurité dès le début ne sont en général pas déployés du tout, car ils sont trop pénibles à utiliser.)

Mais les failles menaçant la vie privée ne sont pas en général dans les protocoles et le problème dépasse donc les seuls moyens de l'IETF. Ainsi, les protocoles IETF sécurisés dépendent en général

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc1984.txt

de la cryptographie mais l'IETF elle-même ne fait pas de cryptographie, elle s'appuie sur des normes extérieures existantes. Par exemple, l'IETF a souvent suivi les recommandations du NIST, alors qu'on sait maintenant que cet organisme sabotait ses propres normes sur ordre de la NSA. La plupart des protocoles IETF sont « "*crypto-agiles*" » ce qui signifie que le protocole n'est pas lié à un seul algorithme cryptographique mais peut en changer. Cela permet d'abandonner un algorithme dont on sait qu'il est cassé mais cela ne dit pas quels algorithmes sont dangereux, car affaiblis délibérement par la NSA.

Et, bien sûr, il y a la mise en œuvre du protocole. Il y a nettement plus de bogues dans les programmes que de failles dans les protocoles. Écrire ces programmes n'est pas le rôle de l'IETF mais celle-ci a quand même une responsabilité : offrir aux programmeurs des spécifications claires, sans ambiguité et lisibles. Outre les bogues accidentelles, les réalisations concrètes des protocoles IETF ont aussi des faiblesses dues à des mauvais choix de sécurité. Ainsi, livrer une machine avec un mot de passe par défaut, dont on sait bien que la plupart des utilisateurs ne le changeront pas, n'est pas formellement une bogue mais cela a un effet désastreux sur la sécurité de l'Internet. Le "*draft*" fait aussi remarquer qu'évidemment, les programmes dont le code source n'est pas publié sont bien plus dangereux, d'autant plus qu'ils peuvent contenir des portes dérobées (trois ont été découvertes dans des routeurs dans les deux semaines précédant la réunion IETF).

Enfin, le "*draft*" de Tschofenig fait remarquer que même avec des protocoles parfaits, de la cryptographie sans faille et des programmes sans bogues, le déploiement effectif peut introduire des vulnérabilités et il cite le cas d'architectures où plusieurs serveurs sont utilisés pour contribuer à rendre le service ("*multi-tier*") et où les communications entre ces serveurs ne sont pas sécurisées (ce qui était le cas de Google il y a peu, et avait fait l'objet du programme MUSCULAR <http://www.washingtonpost.com/world/national-security/nsa-infiltrates-links-to-yahoo-google-data-centers-worldwide-2013/10/30/e51d661e-4166-11e3-8b74-d89d714ca4dd_story.html> de la NSA).

Un autre document PERPASS, pas encore officiellement publié, est le `draft-huitema-perpass-analthreat` de Christian Huitema. Il fait le tour des menaces, analysant ce qu'il est possible de faire en exploitant les protocoles Internet, notamment les fameuses métadonnées. Par exemple, pour le traditionnel téléphone, la simple connaissance de qui parle à qui, même sans accéder au contenu des communications, permet déjà de connaître le graphe des relations. La quantité de trafic, elle, permet de savoir que quelque chose se prépare. Bref, la collecte des métadonnées est une vraie menace pour la vie privée (section 2 du "*draft*"). Dans le monde IP, des données comme celles récoltées par NetFlow (RFC 3954), adresses IP source et destination, quantité de données échangées, présentent les mêmes dangers.

Huitema analyse aussi la liaison entre une adresse IP et une personne. Traditionnellement, c'était fait en demandant au FAI de fournir les informations « lequel de vos abonnés avait l'adresse `192.0.2.49` hier à 20 :13 ? » Mais il note bien (section 4) qu'il existe d'autres méthodes de lier l'adresse IP à une identité, par exemple l'examen des sites Web visités, qui, ensemble, forment une véritable signature d'un individu. Autre exemple, l'analyse des en-têtes `Received:` dans les courriers électroniques permet de récupérer ce lien entre une adresse IP (en général explicitement indiquée) et une adresse de courrier donc sans doute un individu.

Huitema envisage aussi les solutions (section 5). S'il n'y a guère d'espoir d'empêcher techniquement l'espionnage, on peut en tout cas le rendre plus pénible et plus coûteux. D'abord, il faut développer le chiffrement. L'analyse des en-têtes `Received:` mentionnée plus haut serait plus difficile si SMTP utilisait systématiquement TLS (RFC 3207). On pourrait aussi, même si cela rendra le débogage plus difficile, rendre ces en-têtes moins bavards, par exemple en n'indiquant pas l'adresse IP de l'émetteur original. Peut-être aussi faudrait-il revenir de la tendance à avoir des adresses IP stables : si elles changeaient souvent, cela serait moins pratique pour l'utilisateur mais cela rendrait le traçage plus difficile. On pourrait aussi utiliser le "*multi-homing*" de manière plus créative, en envoyant une partie des paquets d'un flot par un lien et le reste par un autre lien, rendant la reconstitution du flot plus difficile. Il y a aussi des

idées plus futuristes (et qui ne font pas l'objet pour l'instant d'une spécification rigoureuse) comme de modifier l'architecture de l'Internet pour le rendre « sans source ». L'en-tête des paquets IP ne porterait plus que l'adresse destination, pas la source, celle-ci serait contenue plus loin dans le paquet, chiffrée, pour les cas où on veut une réponse (la plupart des usages).

Parmi les documents PERPASS, le "*Internet draft*" `draft-hardie-perpass-touchstone` de Ted Hardie choisit une autre question : comment évaluer la sécurité des systèmes ? Un système peut être sûr pour protéger une information triviale d'un "*script-kiddie*" mais pas pour protéger des informations cruciales contre la NSA. Et, entre les deux, il y a plein d'intermédiaires. Hardie propose donc un test de Litmus pour discuter des futurs protocoles IETF : « est-ce qu'un homosexuel en Ouganda peut l'utiliser sans craindre pour sa sécurité ? » L'Ouganda a été choisi pour deux raisons : c'est un pays où il y a une répression active (y compris par la loi) de l'homosexualité, et c'est un pays pauvre, où on peut être tenté de sacrifier la sécurité aux performances. Et pourquoi l'homosexualité ? Parce que, comme le note l'auteur, la surveillance mène à l'auto-censure, au repli sur soi, et que les jeunes homosexuels qui n'arrivent pas à rejoindre une communauté qui les soutient courent énormément de risques (dont le suicide). Si la réponse au test est Oui, cela ne signifie pas que le système résistera à la NSA ou à un autre attaquant déterminé, mais qu'il sera suffisant pour une très large variété d'usages.

Autres lectures sur cette réunion et sur le rôle de l'IETF :

— Bon résumé <http://www.techdirt.com/articles/20131029/09004225049/ietf-begins-to-wo
shtml> à partir du discours d'Arkko à l'IGF à Bali.
— Une analyse très fouillée <http://policyreview.info/articles/news/technical-community-deb
215> de Monika Ermert.
— Mon compte-rendu de la plénière technique et du groupe perpass <https://www.bortzmeyer.
org/ietf-securite-espionnage-bis.html>.
— Le RFC 7258, publié quelques mois après, et qui résume la politique de l'IETF sur cette question.