RFC 8546 : The Wire Image of a Network Protocol

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 29 avril 2019

Date de publication du RFC : Avril 2019

https://www.bortzmeyer.org/8546.html

Ce nouveau RFC de l'IAB décrit le très important concept de **vue depuis le réseau** ("*wire image*"), une abstraction servant à modéliser ce que voit, sur le réseau, une entité qui ne participe pas à un protocole, mais peut en observer les effets. Cela peut être un routeur, un boitier de surveillance, etc. Le concept n'était pas nécessaire autrefois, où tout le trafic était en clair. Maintenant qu'une grande partie est (heureusement) chiffrée, il est important d'étudier ce qui reste visible à ces entités extérieures.

Un protocole de communication, c'est un ensemble de règles que les participants doivent respecter, le format des messages, qui doit envoyer quoi et comment y répondre, etc. Par exemple, si on prend le protocole HTTP, il y a au moins deux participants, le client et le serveur, parfois davantage s'il y a des relais. Ces participants (par exemple le navigateur Web et le serveur HTTP) connaissent le protocole, et le suivent. (Du moins on peut l'espérer.) Mais, en plus des participants, d'autres entités peuvent observer le trafic. Cela va des couches basses de la machine (TCP, IP, Ethernet) aux équipements intermédiaires. Même si le routeur ne connait pas HTTP, et n'en a pas besoin pour faire son travail, il voit passer les bits et peut techniquement les décoder, en suivant le RFC. C'est ainsi que des applications comme Wireshark peuvent nous afficher une compréhension d'un dialogue auxquelles elles ne participent pas.

Cette fuite d'informations vers d'autres entités n'est pas explicite dans la spécification d'un protocole. Autrefois, avec le trafic en clair, elle allait de soi (« bien sûr que le routeur voit tout passer ! »). Aujourd'hui, avec le chiffrement, il faut se pencher sur la question « qu'est-ce que ces entités voient et comprennent du trafic ? » C'est la **vue depuis le réseau** qui compte, pas la spécification du protocole, qui ne mentionne pas les fuites implicites.

Prenons l'exemple de TLS (RFC 8446 [1]). TLS chiffre le contenu de la connexion TCP. Mais il reste des informations visibles : les couches inférieures (un observateur tiers voit le protocole TCP en action, les retransmissions, le RTT, etc), les informations sur la taille (TLS ne fait pas de remplissage,

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc8446.txt

par défaut, ce qui permet, par exemple, d'identifier la page Web regardée <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=6234422>), la dynamique des paquets (délai entre requête et réponse, par exemple). Tout ceci représente la **vue depuis le réseau**.

Le RFC prend un autre exemple, le protocole QUIC <https://www.bortzmeyer.org/quic.html>. Cette fois, la mécanique du protocole de transport est largement cachée par le chiffrement. QUIC a donc une « vue depuis le réseau » réduite. C'est le premier protocole IETF qui essaie délibérement de réduire cette vue, de diminuer le « rayonnement informationnel ». Cela a d'ailleurs entrainé de chaudes discussions, comme celles autour du "*spin bit*", un seul bit d'information laissé délibérement en clair pour informer les couches extérieures sur le RTT. En effet, diminuer la taille de la vue depuis le réseau protège la vie privée mais donne moins d'informations aux opérateurs réseau (c'est bien le but) et ceux-ci peuvent être frustrés de cette décision. Le conflit dans ce domaine, entre sécurité et visibilité, ne va pas cesser de si tôt.

Après cette introduction, la section 2 du RFC décrit formellement cette notion de « vue depuis le réseau ». La **vue depuis le réseau** ("*wire image*") est ce que voit une entité qui ne participe pas aux protocoles en question. C'est la suite des paquets transmis, y compris les métadonnées (comme l'heure de passage du paquet).

La section 3 de notre RFC discute ensuite en détail les propriétés de cette vue. D'abord, elle ne se réduit pas aux « bits non chiffrés ». On l'a vu, elle inclut les métadonnées comme la taille des paquets ou l'intervalle entre paquets. Ces métadonnées peuvent révéler bien des choses sur le trafic. Si vous utilisez OpenVPN pour chiffrer, et que vous faites ensuite par dessus du SSH ou du DNS, ces deux protocoles présentent une vue très différente, même si tout est chiffré. Mais un protocole chiffré, contrairement aux protocoles en clair (où la vue est maximale) peut être conçu pour changer volontairement la vue qu'il offre (la section 4 approfondira cette idée).

La cryptographie peut aussi servir à garantir l'intégrité de la vue (empêcher les modifications), même si on ne chiffre pas. En revanche, toutes les parties de la vue qui n'utilisent pas la cryptographie peuvent être non seulement observées mais encore changées par des intermédiaires. Ainsi, un FAI sans scrupules peut changer les en-têtes TCP pour ralentir certains types de trafic. (Beaucoup de FAI ne respectent pas le principe de neutralité <https://www.bortzmeyer.org/neutralite.html>.)

Notez que la vue depuis le réseau dépend aussi de l'observateur. S'il ne capture qu'un seul paquet, il aura une vue réduite. S'il observe plusieurs paquets, il a accès à des informations supplémentaires, et pas seulement celles contenues dans ces paquets, mais également celles liées à l'intervalle entre paquets. De même, si l'observateur ne voit que les paquets d'un seul protocole, il aura une vue limitée de ce qui se passe alors que, s'il peut croiser plusieurs protocoles, sa vue s'élargit. Un exemple typique est celui du DNS : très majoritairement non chiffré, contrairement à la plupart des protocoles applicatifs, et indispensable à la très grande majorité des transactions Internet, il contribue beaucoup à la vue depuis le réseau (RFC 7626). Si vous voyez une requête DNS pour `imap.example.net` juste avant un soudain trafic, il est facile de suspecter que le protocole utilisé était IMAP. Élargissons encore la perspective : outre le trafic observé, le surveillant peut disposer d'autres informations (le résultat d'une reconnaissance faite avec nmap, par exemple), et cela augmente encore les possibilités d'analyse de la vue dont il dispose.

Puisqu'on parle de vue ("*image*"), le RFC note également que le terme n'est pas uniquement une métaphore, et qu'on pourrait utiliser les techniques de reconnaissance d'images pour analyser ces vues.

Notez que, du point de vue de l'IETF, l'Internet commence à la couche 3. Les couches 1 et 2 contribuent également à la vue depuis le réseau, mais sont plus difficiles à protéger, puisqu'elles n'opèrent pas de bout en bout.

Pour un protocole, il est difficile de réduire la vue qu'il offre au réseau. On ne peut pas rendre les paquets plus petits, ni diminuer l'intervalle entre deux paquets. Une des solutions est d'envoyer volontairement des informations fausses, pour « noyer » les vraies. (Voir le livre de Finn Brunton et Helen Nissenbaum, « Obfuscation » <https://www.bortzmeyer.org/obfuscation.html>, chez C&F Éditions en français.) On ne peut pas réduire les paquets, mais on peut les remplir, par exemple. Ou bien on peut ajouter de faux paquets pour brouiller les pistes. Mais il n'y a pas de miracle, ces méthodes diminueront la capacité <https://www.bortzmeyer.org/capacite.html> utile du réseau, ou ralentiront les communications. (Par exemple, utiliser le Web via Tor est bien plus lent.) Bref, ces méthodes ne sont vraiment acceptables que pour des applications qui ne sont pas trop exigeantes en performance.

J'ai dit plus haut qu'on pouvait assurer l'intégrité de certains champs du protocole, sans les chiffrer. Cela permet d'avoir des informations fiables, non modifiables, mais visibles, ce qui peut être utile pour certains équipements intermédiaires. Notez que cette protection a ses limites : on ne peut protéger que des bits, pas des données implicites comme l'écart entre deux paquets. Et la protection est forcément par paquet puisque, dans un réseau à commutation de paquets, comme l'Internet, on ne peut pas garantir l'arrivée de tous les paquets, ou leur ordre.

Enfin, la dernière section de notre RFC, la section 4, explore les moyens par lesquels un protocole peut tromper un éventuel surveillant, en modifiant la vue qu'il offre au réseau. Une fois qu'on a ce concept de vue depuis le réseau, on peut bâtir des choses utiles sur ce concept. Par exemple, il aide à comprendre des questions d'ossification (la difficulté à déployer de nouveaux services ou protocoles, et qui rend, par exemple, nécessaire de faire passer même le DNS sur HTTPS, comme spécifié dans le RFC 8484). En effet, tout ce qui est visible sera regardé, tout ce qui n'est pas protégé sera modifié. Les boitiers intermédiaires, ou plutôt les entreprises et les États qui les conçoivent et les déploient, n'ont aucun scrupule et ne connaissent aucune restriction. Cela veut dire que si un protocole laisse une information visible, celle-ci sera utilisée par les boitiers intermédiaires et donc il sera difficile de changer sa sémantique par la suite, même si toutes les machines terminales sont d'accord.

Prenons l'exemple de TCP (normalisé dans le RFC 793). TCP envoie un certain nombre de signaux involontaires et implicites. Par exemple, l'observation des numéros de séquence permet de mesurer le RTT. Et TCP permet également de modifier ces signaux. Comme l'explique le RFC 8558, des équipements sont vendus aujourd'hui avec des fonctions de surveillance et tripotage des en-têtes TCP. Le RFC fournit deux exemples d'utilisation de cette surveillance :

— Déterminer la joignabilité et le consentement. Si on voit des réponses respectant le protocole TCP (notamment les numéros de séquences, cf. RFC 6528), cela indique que les deux machines sont d'accord pour communiquer, l'une n'est pas en train d'attaquer l'autre. Cette conclusion peut être utilisée par un pare-feu.
— Mesurer la latence <https://www.bortzmeyer.org/latence.html> et le taux de pertes de paquets. Cela peut se faire, comme indiqué plus haut, en regardant les numéros de séquence dans les paquets et les accusés de réception, et ou en regardant ECN (RFC 3168) et l'estampillage (RFC 7323).

Dans le cas de TCP, cette exposition d'information est « involontaire ». Le but de TCP n'est pas que tout le monde sur le trajet puisse regarder, et encore moins modifier, ces informations. Mais c'est quand même ce qui arrive. Avec un protocole qui réduit consciemment la vue, comme QUIC <https://www.bortzmeyer.org/quic.html>, ne serait-ce pas une bonne idée que de donner un peu à manger aux équipements intermédiaires, afin qu'ils puissent optimiser leurs décisions ? Ce fut tout le débat dans le groupe de travail QUIC à l'IETF sur le "*spin bit*" <https://www.bortzmeyer.org/quic-spin-bit.html>, un bit uniquement conçu pour agrandir un peu la vue dont disposent les équipements du réseau, mais qui était un peu en conflit avec le principe d'en dire le moins possible, et ossifiait un peu le protocole (une fois QUIC déployé avec le "*spin bit*", on ne peut plus le supprimer, sous peine de mettre en colère les "*middleboxes*".)

Les informations accessibles dans la vue sont en pratique difficiles à changer puisque les boitiers intermédiaires vont s'habituer à compter dessus. Au moins, on pourrait les rendre explicites plutôt

qu'implicites, et documenter clairement ces **invariants**, ces informations présentes dans la vue et que les concepteurs du protocole s'engagent à garder dans les évolutions futures. Typiquement, les invariants sont des données stables, et simples. Pour un protocole qui a la notion de version, et de négociation de version, cette négociation a intérêt à être déclarée invariante (RFC 8999 pour le cas de QUIC). Mais attention : une fois qu'on a figé certaines parties de la vue, en les déclarant invariantes, il ne faut pas s'imaginer que les équipements du réseau vont s'arrêter là : ils vont sans doute utiliser d'autres parties de la vue pour prendre leur décision, et ces autres parties risquent donc de devenir des invariants de fait. Le RFC recommande donc que toutes les parties qui ne sont pas explicitement listées comme invariantes soient chiffrées, pas pour la confidentialité, mais pour éviter qu'elles ne deviennent invariantes du fait de leur utilisation par les intermédiaires.

Enfin, le RFC rappelle que les équipements intermédiaires ne peuvent pas savoir ce que les deux parties qui communiquent ont décidé entre elles, et que la véracité de la vue depuis le réseau n'est jamais garantie.