RFC 9065 : Considerations around Transport Header Confidentiality, Network Operations, and the Evolution of Internet Transport Protocols

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 20 juillet 2021

Date de publication du RFC : Juillet 2021

https://www.bortzmeyer.org/9065.html

La couche Transport n'est pas celle qui suscite le plus de passions dans l'Internet. Mais la récente normalisation du protocole QUIC <https://www.bortzmeyer.org/quic.html> a mis cette couche en avant et l'usage du chiffrement par QUIC a relancé le débat : quelles sont les conséquences d'un chiffrement de plus en plus poussé de la couche Transport ?

Traditionnellement, la couche Transport ne faisait pas de chiffrement (cf. RFC 8095 [1] et RFC 8922). On chiffrait en-dessous (IPsec) ou au-dessus (TLS, SSH). IPsec ayant été peu déployé, l'essentiel du chiffrement aujourd'hui sur l'Internet est fait par TLS. Toute la mécanique TCP est donc visible aux routeurs sur le réseau. Ils peuvent ainsi mesurer le RTT, découvrir début et fin d'une connexion, et interférer avec celle-ci, par exemple en envoyant des paquets RST ("*ReSeT*") pour mettre fin à la session. Cela permet de violer la vie privée (RFC 6973), par exemple en identifiant une personne à partir de son activité en ligne. Et cette visibilité de la couche Transport pousse à l'ossification : de nombreux intermédiaires examinent TCP et, si des options inhabituelles sont utilisées, bloquent les paquets. Pour éviter cela, QUIC <https://www.bortzmeyer.org/quic.html> chiffre une grande partie de la couche 4, pour éviter les interférences par les intermédiaires et pour défendre le principe de bout en bout et la neutralité du réseau <https://www.bortzmeyer.org/neutralite.html>. Comme souvent en sécurité, cette bonne mesure de protection a aussi des inconvénients, que ce RFC examine. Notons tout de suite que ce qui est un inconvénient pour les uns ne l'est pas forcément pour les autres : pour un FAI, ne pas pouvoir couper les connexions TCP de BitTorrent avec RST est un inconvénient mais, pour l'utilisateur, c'est un avantage, cela le protège contre certaines attaques par déni de service.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc8095.txt

On ne peut pas sérieusement aujourd'hui utiliser des communications non-chiffrées (RFC 7258). Personne n'ose dire publiquement le contraire. Par contre, on entend souvent un discours « je suis pour le chiffrement, mais » et, comme toujours avec ce genre de phrase, c'est ce qui est après le « mais » qui compte. Ce RFC essaie de documenter les avantages et les inconvénients du chiffrement de la couche Transport, mais, en pratique, est plus détaillé sur les inconvénients, ce qui était déjà le cas du RFC 8404.

La section 2 du RFC explique quel usage peut être fait des informations de la couche Transport par les équipements intermédiaires. En théorie, dans un modèle en couches idéal, il n'y en aurait aucun : la couche Transport est de bout en bout, les routeurs et autres équipements intermédiaires ne regardent rien au-dessus de la couche Réseau. Mais en pratique, ce n'est pas le cas, comme l'explique cette section. (Question pour mes lecteurices au passage : vous semble-t-il légitime de parler de DPI quand un routeur regarde le contenu de la couche Transport, dont il n'a en théorie pas besoin, ou bien doit-on réserver ce terme aux cas où il regarde dans la couche Application ?)

Première utilisation de la couche Transport par des intermédiaires : identifier des flots de données (une suite d'octets qui « vont ensemble »). Pourquoi en a-t-on besoin ? Il y a plusieurs raisons possibles, par exemple pour la répartition de charge, où on veut envoyer tous les paquets d'un flot donné au même serveur. Cela se fait souvent en prenant un tuple d'informations dans le paquet (tuple qui peut inclure une partie de la couche Transport, comme les ports source et destination) et en le condensant pour avoir un identificateur du flot. Si la couche Transport est partiellement ou totalement chiffrée, on ne pourra pas distinguer deux flots différents entre deux machines. En IPv6, l'étiquette de flot (RFC 6437) est une solution possible (RFC 6438, RFC 7098), mais je n'ai pas l'impression qu'elle soit très utilisée.

Maintenant, passons à la question de l'identification d'un flot. Était-ce un transfert de fichiers, de la vidéo, une session interactive ? Il faut déduire cette identification à partir des informations de la couche Transport (voir le RFC 8558). Mais pourquoi identifier ces flots alors que l'opérateur doit tous les traiter pareil, en application du principe de neutralité <https://www.bortzmeyer.org/neutralite.html> ? Cela peut être dans l'intérêt de l'utilisateur (mais le RFC ne donne pas d'exemple…) ou bien contre lui, par exemple à des fins de surveillance, ou bien pour discriminer certains usages (comme le réclament régulièrement certains politiciens et certains opérateurs), voire pour les bloquer complètement. Autrefois, on pouvait souvent identifier un service uniquement avec le numéro de port (43 pour whois, 25 pour le courrier, etc, cf. RFC 7605) mais cela n'a jamais marché parfaitement, plusieurs services pouvant utiliser le même port et un même service pouvant utiliser divers ports. De toute façon, cette identification par le numéro de port est maintenant finie, en partie justement en raison de cette discrimination selon les usages, qui pousse tout le monde à tout faire passer sur le port 443. Certains services ont un moyen simple d'être identifié, par exemple par un nombre magique, volontairement placé dans les données pour permettre l'identification, ou bien simple conséquence d'une donnée fixe à un endroit connu (RFC 3261, RFC 8837, RFC 7983…). Lors de la normalisation de QUIC <https://www.bortzmeyer.org/quic.html>, un débat avait eu lieu sur la pertinence d'un nombre magique permettant d'identifier du QUIC, idée finalement abandonnée.

Si les équipements intermédiaires indiscrets n'arrivent pas à déterminer le service utilisé, le flot va être considéré comme inconnu et le RFC reconnait que certains opérateurs, en violation de la neutralité de l'Internet, ralentissent ces flots inconnus.

L'étape suivante pour ceux qui veulent identifier à quoi servent les données qu'ils voient passer est d'utiliser des heuristiques. Ainsi, une visio-conférence à deux fera sans doute passer à peu près autant d'octets dans chaque sens, alors que regarder de la vidéo à la demande créera un trafic très asymétrique. Des petits paquets UDP régulièrement espacés permettent de soupçonner du trafic audio, même si on n'a pas pu lire l'information SDP (RFC 4566). Des heuristiques plus subtiles peuvent permettre d'en savoir plus. Donc, il faut se rappeler que le chiffrement ne dissimule pas tout, il reste une vue qui peut être plus ou moins précise (le RFC 8546 décrit en détail cette notion de vue depuis le réseau).

Autre motivation pour analyser la couche Transport, l'amélioration des performances. Inutile de dire que le FAI typique ne va pas se pencher sur les problèmes de performance d'un abonné individuel (si ça rame avec Netflix, appeler le support de son FAI ne déclenche pas de recherches sérieuses). Mais cela peut être fait pour des analyse globales. Là encore, les conséquences peuvent être dans l'intérêt de l'utilisateur, ou bien contre lui. Le RFC note que les mesures de performance peuvent amener à une discrimination de certains services (« QoS », qualité de service, c'est-à-dire dégradation de certains services). Que peut-on mesurer ainsi, qui a un impact sur les performances ? Il y a la perte de paquets, qu'on peut déduire, en TCP, des retransmissions. Dans l'Internet, il y a de nombreuses causes de pertes de paquets, du parasite sur un lien radio à l'abandon délibéré par un routeur surchargé (RFC 7567) en passant par des choix politiques de défavoriser certains paquets (RFC 2475). L'étude de ces pertes peut permettre dans certains cas de remonter aux causes.

On peut aussi mesurer le débit. Bon, c'est facile, sans la couche Transport, uniquement en regardant le nombre d'octets qui passent par les interfaces réseaux. Mais l'accès aux données de la couche Transport permet de séparer le débit total du débit utile ("*goodput*", en anglais, pour le différencier du débit brut, le "*throughput*", cf. section 2.5 du RFC 7928, et le RFC 5166). Pour connaitre ce débit utile, il faut pouvoir reconnaitre les retransmissions (si un paquet est émis trois fois avant enfin d'atteindre le destinataire, il ne contribue qu'une fois au débit utile). Une retransmission peut se voir en observant les numéros de séquence en TCP (ou dans d'autres protocoles comme RTP).

La couche Transport peut aussi nous dire quelle est la latence <https://www.bortzmeyer.org/latence.html>. Cette information est cruciale pour évaluer la qualité des sessions interactives, par exemple. Et elle influe beaucoup sur les calculs du protocole de couche 4. (Voir l'article « "*Internet Latency : A Survey of Techniques and their Merits*" <http://bobbriscoe.net/projects/latency/latency_preprint.pdf> ».) Comment mesure-t-on la latence ? Le plus simple est de regarder les accusés de réception TCP et d'en déduire le RTT. Cela impose d'avoir accès aux numéros de séquence. Dans TCP, ils sont en clair, mais QUIC <https://www.bortzmeyer.org/quic.html> les chiffre (d'où l'ajout du "*spin bit*" <https://www.bortzmeyer.org/quic-spin-bit.html>).

D'autres métriques sont accessibles à un observateur qui regarde la couche Transport. C'est le cas de la gigue, qui se déduit des observations de la latence, ou du réordonnancement des paquets (un paquet qui part après un autre, mais arrive avant). L'interprétation de toutes ces mesures dépend évidemment du type de lien. Un lien radio (RFC 8462) a un comportement différent d'un lien filaire (par exemple, une perte de paquets n'est pas forcément due à la congestion, elle peut venir de parasites).

Le RFC note que la couche Réseau, que les équipements intermédiaires ont tout à fait le droit de lire, c'est son rôle, porte parfois des informations qui peuvent être utiles. En IPv4, ce sont les options dans l'en-tête (malheureusement souvent jetées par des pare-feux trop fascistes, cf. RFC 7126), en IPv6, les options sont après l'en-tête de réseau, et une option, "*Hop-by-hop option*" est explicitement prévue pour être examinée par tous les routeurs intermédiaires.

Outre les statistiques, l'analyse des données de la couche Transport peut aussi servir pour les opérations (voir aussi le RFC 8517), pour localiser un problème, pour planifier l'avitaillement de nouvelles ressources réseau, pour vérifier qu'il n'y a pas de tricheurs qui essaient de grapiller une part plus importante de la capacité <https://www.bortzmeyer.org/capacite.html>, au risque d'aggraver la congestion (RFC 2914). En effet, le bon fonctionnement de l'Internet dépend de chaque machine terminale. En cas de perte de paquets, signal probable de congestion, les machines terminales sont censées réémettre les paquets avec prudence, puisque les ressources réseau sont partagées. Mais une machine égoïste pourrait avoir plus que sa part de la capacité. Il peut donc être utile de surveiller ce qui se passe, afin d'attraper d'éventuels tricheurs, par exemple une mise en œuvre de TCP qui ne suivrait pas les règles habituelles. (Si on utilise UDP, l'application doit faire cela elle-même, cf. RFC 8085. Ainsi, pour

RTP, comme pour TCP, un observateur extérieur peut savoir si les machines se comportent normalement ou bien essaient de tricher.)

Autre utilisation de l'observation de la couche Transport pour l'opérationnel, la sécurité, par exemple la lutte contre les attaques par déni de service, l'IDS et autres fonctions. Le RFC note que cela peut se faire en coopération avec les machines terminales, si c'est fait dans l'intérêt de l'utilisateur. Puisqu'on parle de machines terminales <https://www.bortzmeyer.org/terminal-host.html>, puisque le chiffrement d'une partie de la couche Transport est susceptible d'affecter toutes les activités citées plus haut, le RFC rappelle la solution évidente : demander la coopération des machines terminales. Il y a en effet deux cas : soit les activités d'observation de la couche Transport sont dans l'intérêt des utilisateurs, et faites avec leur consentement, et dans ce cas la machine de l'utilisateur peut certainement coopérer, soit ces activités se font contre l'utilisateur (discrimination contre une application qu'il utilise, par exemple), et dans ce cas le chiffrement est une réponse logique à cette attaque. Bien sûr, c'est la théorie ; en pratique, certaines applications ne fournissent guère d'informations et de moyens de débogage. Les protocoles de transport qui chiffrent une bonne partie de leur fonctionnement peuvent aussi aider, en exposant délibérement des informations. C'est par exemple ce que fait QUIC <https://www.bortzmeyer.org/quic.html> avec son "*spin bit*" <https://www.bortzmeyer.org/quic-spin-bit.html> déjà cité, ou avec ses invariants documentés dans le RFC 8999.

Autre cas où le chiffrement de la couche Transport peut interférer avec certains usages, les réseaux d'objets contraints, disposant de peu de ressources (faible processeur, batterie qu'il ne faut pas vider trop vite, etc). Il arrive dans ce cas d'utiliser des relais qui interceptent la communication, bricolent dans la couche Transport puis retransmettent les données. Un exemple d'un tel bricolage est la compression des en-têtes, courante sur les liens à très faible capacité (cf. RFC 2507, RFC 2508, le ROHC du RFC 5795, RFC 6846, le SCHC du RFC 8724, etc). Le chiffrement rend évidemment cela difficile, les relais n'ayant plus accès à l'information. C'est par exemple pour cela que le RTP sécurisé du RFC 3711 authentifie l'en-tête mais ne le chiffre pas. (Je suis un peu sceptique sur cet argument : d'une part, les objets contraints ne vont pas forcément utiliser des protocoles de transport chiffrés, qui peuvent être coûteux, d'autre part un sous-produit du chiffrement est souvent la compression, ce qui rend inutile le travail des relais.)

Un dernier cas cité par le RFC où l'observation du fonctionnement de la couche Transport par les machines intermédiaires est utile est celui de la vérification de SLA. Si un contrat ou un texte légal prévoit certaines caractéristiques pour le réseau, l'observation de la couche 4 (retransmission, RTT…) est un moyen d'observer sans avoir besoin d'impliquer les machines terminales. (Personnellement, je pense justement que ces vérifications devraient plutôt se faire depuis les machines terminales, par exemple avec les sondes RIPE Atlas <https://atlas.ripe.net/>, les SamKnows <https://www.bortzmeyer.org/samknows.html>, etc.)

La section 3 du RFC décrit un autre secteur qui est intéressé par l'accès aux données de transport, la recherche. Par exemple, concevoir de nouveaux protocoles doit s'appuyer sur des mesures faites sur les protocoles existants, pour comprendre leurs forces et leurs faiblesses. C'est possible avec un protocole comme TCP, où l'observation passive permet, via notamment les numéros de séquence, de découvrir le RTT et le taux de perte de paquets. (Passive : sans injecter de paquets dans le réseau. Voir le RFC 7799.) Mais ces mêmes informations peuvent aussi servir contre l'utilisateur. Même s'il n'y a pas d'intention néfaste (par exemple de discrimination contre certains usages), toute information qui est exposée peut conduire à l'ossification, l'impossibilité de changer le protocole dans le futur. Une des motivations des protocoles chiffrés comme QUIC est en effet d'éviter l'ossification : une "*middlebox*" ne pourra pas prendre de décisions sur la base d'informations qu'elle n'a pas. QUIC affiche des données au réseau seulement s'il le veut (c'est le cas du "*spin bit*" <https://www.bortzmeyer.org/quic-spin-bit.html>). D'où également le choix délibéré de graisser, c'est-à-dire de faire varier certaines informations pour éviter que des programmeurs de "*middleboxes*" incompétents et/ou paresseux n'en déduisent que cette information ne change jamais (le graissage est décrit dans le RFC 8701).

La bonne solution pour récolter des données sans sacrifier la vie privée est, comme dit plus haut, de faire participer les extrémités, les machines terminales, ce qu'on nomme en anglais le "*endpoint-based logging*". Actuellement, malheureusement, les mécanismes de débogage ou de récolte d'information sur ces machines terminales sont trop réduits, mais des efforts sont en cours. Par exemple, pour QUIC, c'est la normalisation du format « qlog » d'enregistrement des informations vues par la couche Transport ("*Internet-Draft*" `draft-ietf-quic-qlog-main-schem`) ou bien le format Quic-Trace `<https://github.com/google/quic-trace>`. Mais le RFC note que la participation des machines terminales ne suffit pas toujours, notamment si on veut déterminer **où**, dans le réseau, se produit un problème.

Après qu'on ait vu les utilisations qui sont faites de l'analyse de la couche Trnsport par les équipements intermédiaires, la section 4 du RFC revient ensuite sur les motivations du chiffrement de cette couche. Pourquoi ne pas se contenter de ce que font TLS et SSH, qui chiffrent uniquement la couche Application ? L'une des premières raisons est d'empêcher l'ossification, ce phénomène qui fait qu'on ne peut plus faire évoluer la couche Transport car de stupides équipements intermédiaires, programmés avec les pieds par des ignorants qui ne lisent pas les RFC, rejettent les paquets légaux mais qui ne correspondent pas à ce que ces équipements attendaient. Ainsi, si un protocole de transport permet l'utilisation d'un octet dans l'en-tête, mais que cet octet est à zéro la plupart du temps, on risque de voir des "*middleboxes*" qui jettent les paquets où certains bits de ce champ sont à un car « ce n'est pas normal ». Tout ce qui est observable risque de devenir ossifié, ne pouvant plus être modifié par la suite. Chiffrer permet de garantir que les équipements intermédiaires ne vont pas regarder ce qui ne les regarde pas. Le RFC donne plusieurs exemples édifiants des incroyables comportements de ces logiciels écrits par des gens qui ne comprenaient qu'une partie d'un protocole :

— Pendant le développement de TLS 1.3 (qui mènera au RFC 8446), il a fallu concevoir 1.3 de manière à ce qu'il ressemble à 1.2, car certaines "*middleboxes*" rejettaient du TLS légal, mais différent de ce qu'elles attendaient.
— MPTCP (RFC 8684) a également dû être modifié pour tenir compte de boitiers intermédiaires qui observaient le fonctionnement de la fenêtre TCP et se permettaient de couper les connexions qui leur semblaient anormales.
— D'une manière générale, tout protocole qui permet des options est confronté à des "*middleboxes*" qui interfèrent dès qu'on utilise des options nouvelles. C'est le cas par exemple de TCP Fast Open (RFC 7413).
— Encore pire, si c'est possible, on a vu des équipements intermédiaires qui changeaient les numéros de séquence TCP, ce qui cassait les accusés de réception SACK (RFC 2018).

Il n'est donc pas étonnant que les concepteurs de protocole cherchent désormais à chiffrer au maximum, pour éviter ces interférences. Le RFC 8546 rappelle ainsi que c'est la vue depuis le réseau ("*wire image*"), c'est-à-dire ce que les équipements intermédiaires peuvent observer, pas la spécification écrite du protocole, qui détermine, dans le monde réel, ce qu'un intermédiaire peut observer et modifier. Il faut donc réduire cette vue au strict minimum ; **tout ce qui n'est pas chiffré risque fortement d'être ossifié, figé**. Et le RFC 8558 affirme lui qu'on ne doit montrer au réseau que ce qui doit être utilisé par le réseau, le reste, qui ne le regarde pas, doit être dissimulé.

Une autre motivation du chiffrement de la couche Transport est évidemment de mieux protéger la vie privée (RFC 6973). L'ampleur de la surveillance massive (RFC 7624) est telle qu'il est crucial de gêner cette surveillance le plus possible. Le RFC note qu'il n'y a pas que la surveillance passive, il y a aussi l'ajout de données dans le trafic, pour faciliter la surveillance `<https://www.bortzmeyer.org/enrichir-qui.html>`. Du fait de cet « enrichissement », il peut être utile, quand un champ doit être observable (l'adresse IP de destination est un bon exemple), d'utiliser quand même la cryptographie pour empêcher ses modifications, via un mécanisme d'authentification. C'est ce que fait TCP-AO (RFC 5925, mais qui semble peu déployé), et bien sûr le service AH d'IPsec (RFC 4302).

Comme on le voit, il y a une tension, voire une lutte, entre les opérateurs réseau et les utilisateurs. On pourrait se dire que c'est dommage, qu'il vaudrait mieux que tout le monde travaille ensemble. Cela a été discuté à l'IETF, avec des expressions comme « un traité de paix entre machines terminales et boitiers

intermédiaires ». Pour l'instant, cela n'a pas débouché sur des résultats concrets, en partie parce qu'il n'existe pas d'organisations représentatives qui pourraient négocier, signer et faire respecter un tel traité de paix. On en reste donc aux mesures unilatérales. Les machines terminales doivent chiffrer de plus en plus pour maintenir le principe de bout en bout. Comme dans tout conflit, il y a des dégâts collatéraux (le RFC 8922 en décrit certains). Le problème n'étant pas technique mais politique, il est probable qu'il va encore durer. La tendance va donc rester à chiffrer de plus en plus de choses.

À noter qu'une autre méthode que le chiffrement existe pour taper sur les doigts des boitiers intermédiaires pénibles, qui se mêlent de ce qui ne les regarde pas, et s'en mêlent mal : c'est le graissage. Son principe est d'utiliser délibérément toutes les options possibles du protocole, pour habituer les "*middleboxes*" à voir ces variations. Le RFC 8701 en donne un exemple, pour le cas de TLS.

Déterminer ce qu'il faut chiffrer, ce qu'il faut authentifier, et ce qu'il vaut mieux laisser sans protection, autorisant l'observation et les modifications, n'est pas une tâche facile. Autrefois, tout était exposé parce qu'on avait moins de problèmes avec les boitiers intermédiaires et que les solutions, comme le chiffrement, semblaient trop lourdes. Aujourd'hui qu'on a des solutions réalistes, on doit donc choisir ce qu'on montre ou pas. Le choix est donc désormais explicite (cf. RFC 8558).

Au passage, une façon possible d'exposer des informations qui peuvent être utiles aux engins intermédiaires est via un en-tête d'extension. Par exemple en IPv6, l'en-tête "*Hop-by-hop*" (RFC 8200, section 4.3) est justement fait pour cela (voir un exemple dans le RFC 8250, quoiqu'avec un autre type d'en-tête). Toutefois, cet en-tête "*Hop-by-hop*" est clairement un échec : beaucoup de routeurs jettent les paquets qui le portent (RFC 7872), ou bien les traitent plus lentement que les paquets sans cette information. C'est encore pire si cet en-tête porte des nouvelles options, inconnues de certaines "*middleboxes*", et c'est pour cela que le RFC 8200 déconseille (dans sa section 4.8) la création de nouvelles options "*Hop-by-hop*".

Mais, bon, le plus important est de décider quoi montrer, pas juste comment. Le RFC rappelle qu'il serait sympa d'exposer explicitement des informations comme le RTT ou le taux de pertes vu par les machines terminales, plutôt que de laisser les machines intermédiaires le calculer (ce qu'elles ne peuvent de toute façon plus faire en cas de chiffrement). Cela permettrait de découpler l'information de haut niveau des détails du format d'un protocole de transport. Pourquoi une machine terminale ferait-elle cela, au risque d'exposer des informations qu'on peut considérer comme privées ? Le RFC cite la possibilité d'obtenir un meilleur service, sans trop préciser s'il s'agit de laisser les opérateurs offrir un traitement préférentiel aux paquets portant cette information, ou bien si c'est dans l'espoir que l'information exposée serve à l'opérateur pour améliorer son réseau. (Comme le note le RFC 8558, il y a aussi le risque que la machine terminale mente au réseau. Au moins, avec le chiffrement, les choses sont claires : « je refuse de donner cette information » est honnête.)

Dernière note, cet ajout d'informations utiles pour l'OAM peut être faite par la machine terminale mais aussi (section 6 du RFC) par certains équipements intermédiaires.

En conclusion ? La section 7 du RFC reprend et résume les points importants :

— Le chiffrement et l'authentification dans la couche de Transport sont une bonne chose. Personne n'ose dire ouvertement qu'il faudrait rester à des protocoles de transport non sécurisés. C'est un point sur lequel le document a beaucoup évolué. Dans les versions antérieures, comme le notait Christian Huitema, « "*Much of the draft reads like a lamentation of the horrible consequences of encrypting transport headers*" <https://mailarchive.ietf.org/arch/msg/tsvwg/ctPi-nysGSrUNRl_slM8HNYNl60/> », reflétant unilatéralement le point de vue des opérateurs réseau, et des vendeurs de "*middleboxes*". Une relecture par Christopher Wood <https://datatracker.ietf.org/doc/review-ietf-tsvwg-transport-encrypt-01-secdir-early-wood-2018-12-27/> au début du projet avait déjà pointé ce problème, notant que le document était très anti-chiffrement. Cette question a, fort logiquement, été le principal point de discussion à l'IETF.

— Le RFC, officiellement, ne tranche pas sur la pertinence et l'éthique des pratiques qu'il décrit, il explique juste ce qui se fait. (Le même argument, que je trouve un peu hypocrite, avait été utilisé pour le très contestable RFC 8404.)
— Comme souvent en sécurité, il n'y a pas de solution idéale, il faudra trouver un compromis, par exemple entre la vie privée et l'OAM. Le RFC cite le "*spin bit*" de QUIC, qui avait été très chaudement discuté, comme un exemple de compromis, en tout cas par le sérieux de l'analyse de ses coûts et de ses bénéfices.
— Le RFC reconnait que tout ce qui est exposé au réseau s'ossifiera et deviendra une spécification de fait, qu'on ne pourra plus changer. Qu'un protocole choisisse d'exposer beaucoup ou au contraire très peu, il doit de toute façon faire ce choix explicitement.
— Même le chiffrement de la couche Transport ne cache pas tout, et les couches inférieures exposent toujours des métadonnées. Un surveillant déterminé n'est donc pas désarmé. (Même si des organisations comme Interpol prétendent que le chiffrement rend la police « aveugle ».)
— Les opérationnels se sont habitués depuis longtemps à disposer de certaines informations, que le chiffrement de la couche Transport peut rendre inutilisables. Il faudra donc changer certaines pratiques et certains outils, par exemple avec davantage de coopération des machines terminales (sinon, les opérations seront affectées).
— Le RFC rappelle aussi qu'il existe différents types de réseaux, et qui n'ont pas forcément les mêmes contraintes et les mêmes buts. Entre le réseau d'une entreprise qui veut contrôler tout ce que font les employés et le réseau d'un FAI qui doit respecter (en théorie…) le principe de neutralité <https://www.bortzmeyer.org/neutralite.html>, il n'est pas du tout sûr qu'on puisse trouver des solutions qui plaisent à tout le monde. (A priori, l'IETF travaille et normalise pour l'Internet ouvert, pas forcément pour chaque réseau connecté à l'Internet avec ses règles spécifiques.)
— L'Internet est un réseau partagé et son bon fonctionnement dépend donc du respect de certaines règles par tous. Par exemple, un protocole de transport doit penser aux autres, en ne noyant pas le réseau sous les paquets de retransmission. Si un fournisseur de logiciels était tenté de développer un protocole de transport égoïste, qui tente d'obtenir plus que sa part de la capacité du réseau, la tricherie pourrait se détecter en observant le fonctionnement de ce protocole. Le chiffrement de la couche Transport rend évidemment la vérification plus complexe.
— Le bon fonctionnement de l'Internet sur le long terme dépend également d'une activité de recherche et développement, qui
s'appuie sur des mesures <https://www.research-collection.ethz.ch/handle/20.500.11850/314604>, que le chiffrement de la couche Transport peut gêner. (C'est bien, de se préoccuper des chercheurs.)