RFC 9620 : Guidelines for Human Rights Protocol and Architecture Considerations

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 17 septembre 2024

Date de publication du RFC : Septembre 2024

https://www.bortzmeyer.org/9620.html

Voici un RFC explicitement politique, puisqu'il documente la façon dont les concepteur·rices de protocoles à l'IETF devraient examiner les conséquences de leurs protocoles sur l'exercice des droits humains. Si vous concevez un protocole réseau (IETF ou pas), c'est une lecture recommandée.

Les protocoles ne sont pas neutres puisqu'ils ont des conséquences concrètes sur les utilisateurices, conséquences positives ou négatives. S'ils n'en avaient pas, on ne passerait pas du temps à les développer, et on ne dépenserait pas d'argent à les déployer. Ces conséquences ne sont pas forcément faciles à déterminer, surtout avant tout déploiement effectif, mais ce RFC peut guider la réflexion et permettre d'identifier les points qui peuvent avoir des conséquences néfastes. Il comprend de nombreux exemples tirés de précédents RFC. Il met à jour partiellement le RFC 8280[1], dont il reprend la section 6, et s'inspire de la méthode du RFC 6973, qui documentait les conséquences des protocoles sur la vie privée.

La question est évidemment complexe. Les protocoles n'ont pas forcément un pouvoir direct sur ce que les utilisateurices peuvent faire ou pas (c'est l'argument central de ceux qui estiment que les techniques sont neutres : HTTP transférera aussi bien un article de la NASA qu'un texte complotiste sur les extra-terrestres). Leur rôle est plutôt indirect, en ce qu'ils encouragent ou découragent certaines choses, plutôt que d'autoriser ou interdire. Et puis, comme le note le RFC, cela dépend du déploiement. Par exemple, pour le courrier électronique, des faits politiques importants ne s'expliquent pas par le protocole. Ce ne sont pas les particularités de SMTP (RFC 5321) qui expliquent la domination de Gmail par exemple. Il ne faut en effet pas tomber dans le déterminisme technologique (comme le font par exemple les gens qui critiquent DoH <https://www.bortzmeyer.org/doh-et-ses-adversaires.html>) : l'effet dans le monde réel d'un protocole dépend de bien d'autres choses que le protocole.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc8280.txt

Ah, et autre point dans l'introduction, la définition des droits humains. Notre RFC s'appuie sur la déclaration universelle des droits humains <http://www.un.org/en/documents/udhr/> et sur d'autres documents comme le "*International Covenant on Economic, Social and Cultural Rights*" <http://www.ohchr.org/EN/ProfessionalInterest/Pages/CESCR.aspx>.

Parmi les droits humains cités dans cette déclaration universelle, nombreux sont ceux qui peuvent être remis en cause sur l'Internet : liberté d'expression, liberté d'association, droit à la vie privée, non-discrimination, etc. L'absence d'accès à l'Internet mène également à une remise en cause des droits humains, par exemple parce que cela empêche les lanceurs d'alerte de donner l'alerte. Les atteintes aux droits humains peuvent être directes (censure) ou indirectes (la surveillance des actions peut pousser à l'auto-censure, et c'est souvent le but poursuivi par les acteurs de la surveillance ; cf. « "*Chilling Effects : Online Surveillance and Wikipedia Use*" <http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2769645> »). Et les dangers pour les individus ne sont pas seulement « virtuels », ce qui se passe en ligne a des conséquences physiques quand, par exemple, une campagne de haine contre des gens accusés d'attaques contre une religion mène à leur assassinat, ou quand un État emprisonne ou tue sur la base d'informations récoltées en ligne.

C'est pour cela que, par exemple, le conseil des droits humains de l'ONU mentionne que les droits qu'on a dans le monde physique doivent aussi exister en ligne <https://documents-dds-ny.un.org/doc/UNDOC/LTD/G16/131/89/PDF/G1613189.pdf?OpenElement>. [La propagande des médias et des politiciens en France dit souvent que « l'Internet est une zone de non-droit » et que « ce qui est interdit dans le monde physique doit aussi l'être en ligne », afin de justifier des lois répressives. Mais c'est une inversion complète de la réalité. En raison des particularités du numérique, notamment la facilité de la surveillance de masse, et de l'organisation actuelle du Web, avec un petit nombre d'acteurs médiant le trafic entre particuliers, les droits humains sont bien davantage menacés en ligne.] Sur la question de l'application des droits humains à l'Internet, on peut aussi lire les « "*10 Internet Rights & Principles¿*" <https://internetrightsandprinciples.org/campaign/> » et le « "*Catalog of Human Rights Related to ICT Activities*" <http://www.apig.ch/UNIGE%20Catalog.pdf> ».

Comme les droits humains sont précieux, et sont menacés sur l'Internet, l'IETF doit donc veiller à ce que son travail, les protocoles développés, n'aggravent pas la situation. D'où la nécessité, qui fait le cœur de ce RFC, d'examiner les protocoles en cours de développement. Ces examens ("*reviews*", section 3 du RFC) devraient être systématiques et, évidemment, faits en amont, pas une fois le protocole déployé. [En pratique, ces examens ont assez vite été arrêtés <https://github.com/IRTF-HRPC/reviews>, et ce RFC ne reflète donc pas la situation actuelle.]

Vu la façon dont fonctionne l'IETF, il n'y a pas besoin d'une autorité particulière pour effectuer ces examens. Tout·e participant·e à l'IETF peut le faire. Ce RFC 9620 vise à guider les examinateurs ("*reviewers*"). L'examen peut porter sur le contenu d'un "*Internet Draft*" mais aussi être complété, par exemple, avec des interviews d'experts de la question (les conséquences de tel ou tel paragraphe dans un "*Internet Draft*" ne sont pas forcément évidentes à la première lecture, ou même à la seconde) mais aussi des gens qui seront affectés par le protocole en question. Si un futur RFC parle d'internationalisation, par exemple, il ne faut pas interviewer que des anglophones, et pas que des participants à l'IETF, puisque l'internationalisation concerne tout le monde.

Une grosse difficulté, bien sûr, est que le protocole n'est pas tout. Les conditions effectives de son déploiement, et son évolution dans le temps, sont cruciales. Ce n'est pas en lisant le RFC 5733 (ou le RFC 9083) qu'on va tout comprendre sur les enjeux de la protection des données personnelles des titulaires de noms de domaine ! Le RFC 8980 discute d'ailleurs de cette importante différence entre le protocole et son déploiement.

La plus importante section de notre RFC est sans doute la section 4, qui est une sorte de "*check-list*" pour les auteur·es de protocoles et les examinateurices. Idéalement, lors de la phase de conception d'un

protocole, il faudrait passer toutes ces questions en revue. Bien évidemment, les réponses sont souvent complexes : la politique n'est pas un domaine simple.

Premier exemple (je ne vais pas tous les détailler, rassurez-vous), les intermédiaires. Est-ce que le protocole permet, voire impose, des intermédiaires dans la communication ? C'est une question importante car ces intermédiaires, s'ils ne sont pas sous le contrôle des deux parties qui communiquent, peuvent potentiellement surveiller la communication (risque pour la confidentialité) ou la perturber (risque pour la liberté de communication). Un exemple est l'interception HTTPS (cf. « *"The Security Impact of HTTPS Interception"* <https://www.ndss-symposium.org/ndss2017/ndss-2017-programme/security-impact-https> »). Le principe de bout en bout (RFC 1958 ou bien « *"End-to-End Arguments in System Design"* <https://dl.acm.org/doi/10.1145/357401.357402> ») promeut plutôt une communication sans intermédiaires, mais on trouve de nombreuses exceptions dans les protocoles IETF (DNS, SMTP…) ou en dehors de l'IETF (ActivityPub), car les intermédiaires peuvent aussi rendre des services utiles. En outre, ces intermédiaires tendent à ossifier le protocole, en rendant plus difficile le déploiement de tout changement (cf. RFC 8446 pour les problèmes rencontrés par TLS 1.3).

Le RFC fait aussi une différence entre intermédiaires et services. Si vous êtes utilisateurice de Gmail, Gmail n'est pas un intermédiaire, mais un service car vous êtes conscient·e de sa présence et vous l'avez choisi. [L'argument me semble avoir des faiblesses : ce genre de services pose exactement les mêmes problèmes que les intermédiaires et n'est pas forcément davantage maitrisé.]

Un bon moyen de faire respecter le principe de bout en bout est de chiffrer au maximum. C'est pour cela que QUIC (RFC 9000) chiffre davantage de données que TLS. C'est aussi pour cela que l'IETF travaille au chiffrement du SNI (`draft-ietf-tls-esni`).

Autre exemple, la connectivité (section 4.2). Car, si on n'a pas accès à l'Internet du tout, ou bien si on y a accès dans des conditions très mauvaises, tout discussion sur les droits humains sur l'Internet devient oiseuse. L'accès à l'Internet est donc un droit nécessaire (cf. la décision du Conseil Constitutionnel sur Hadopi <https://www.conseil-constitutionnel.fr/decision/2009/2009580DC.htm>, qui posait le principe de ce droit d'accès à l'Internet). Pour le protocole, cela oblige à se pencher sur son comportement sur des liens de mauvaise qualité : est-ce que ce protocole peut fonctionner lorsque la liaison est mauvaise ?

Un sujet délicat (section 4.4) est celui des informations que le protocole laisse fuiter (l'« image vue du réseau » du RFC 8546). Il s'agit des données qui ne sont pas chiffrées, même avec un protocole qui fait de la cryptographie, et qui peuvent donc être utilisées par le réseau, par exemple pour du traitement différencié (de la discrimination, pour dire les choses franchement). Comme recommandé par le RFC 8558, tout protocole doit essayer de limiter cette fuite d'informations. C'est pour cela que TCP a tort d'exposer les numéros de port, par exemple, que QUIC va au contraire dissimuler.

L'Internet est mondial, on le sait. Il est utilisé par des gens qui ne parlent pas anglais et/ou qui n'utilisent pas l'alphabet latin. Il est donc crucial que le protocole fonctionne pour tout le monde (section 4.5). Si le protocole utilise des textes en anglais, cela doit être de manière purement interne (le *"GET"* de HTTP, le *"Received :"* de l'IMF, etc), sans être obligatoirement montré à l'utilisateurice. Ce principe, formulé dans le RFC 2277, dit que tout ce qui est montré à l'utilisateur doit pouvoir être traduit. (En pratique, il y a des cas limites, comme les noms de domaine, qui sont à la fois éléments du protocole, **et** montrés aux utilisateurs.)

Si le protocole sert à transporter du texte, il doit évidemment utiliser Unicode, de préférence encodé en UTF-8. S'il accepte d'autres encodages et/ou d'autres jeux de caractère (ce qui peut être dangereux

pour l'interopérabilité), il doit permettre d'étiqueter ces textes, afin qu'il n'y ait pas d'ambiguité sur leurs caractéristiques. Pensez d'ailleurs à lire le RFC 6365.

Un contre-exemple est le vieux protocole whois (RFC 3912), qui ne prévoyait que l'ASCII et, si on peut l'utiliser avec d'autres jeux de caractères, comme il ne fournit pas d'étiquetage, le client doit essayer de deviner de quoi il s'agit. (Normalement, whois n'est plus utilisé, on a le Web et RDAP, mais les vieilles habitudes ont la vie dure <https://www.afnic.fr/observatoire-ressources/papier-expert/trouver-les-informations-sociales-associees-a-un-nom-de-domaine/>.)

Toujours question étiquetage, notre RFC rappelle l'importance de pouvoir, dans le protocole, indiquer explicitement la langue des textes (RFC 5646). C'est indispensable afin de permettre aux divers logiciels de savoir quoi en faire, par exemple en cas de synthèse vocale.

Le RFC parle aussi de l'élaboration des normes techniques (section 4.7). Par exemple, sont-elles dépendantes de brevets (RFC 8179 et RFC 6701) ? [Personnellement, je pense que c'est une question complexe : les brevets ne sont valables que dans certains pays et, en outre, la plupart des brevets logiciels sont futiles, brevetant des technologies banales et déjà connues. Imposer, comme le proposent certains, de ne normaliser que des techniques sans brevet revient à donner un doit de veto à n'importe quelle entreprise qui brevète n'importe quoi. Par exemple, le RFC 9156 n'aurait jamais été publié si on s'était laissé arrêter par le brevet.]

Mais un autre problème avec les normes techniques concerne leur disponibilité. Si l'IETF, le W3C et même l'UIT publient leurs normes, ce n'est pas le cas de dinosaures comme l'AFNOR ou l'ISO, qui interdisent même la redistribution de normes qu'on a légalement acquises. Si les normes IETF sont de distribution libre, elles dépendent parfois d'autres normes qui, elles, ne le sont pas.

Un peu de sécurité informatique, pour continuer. La section 4.11 traite de l'authentification des données (ce que fait DNSSEC pour le DNS, par exemple). Cette possibilité d'authentification est évidemment cruciale pour la sécurité mais le RFC note qu'elle peut aussi être utilisée négativement, par exemple avec les menottes numériques.

Et il y a bien sûr la confidentialité (section 4.12 mais aussi RFC 6973), impérative depuis toujours mais qui était parfois sous-estimée, notamment avant les révélations Snowden. Les auteur·es de protocoles doivent veiller à ce que les données soient et restent confidentielles et ne puissent pas être interceptées par un tiers. Il y a longtemps que tout RFC doit contenir une section sur la sécurité (RFC 3552), exposant les menaces spécifiques à ce RFC et les contre-mesures prises, entre autre pour assurer la confidentialité. L'IETF refuse, à juste titre, toute limitation de la cryptographie, souvent demandée par les autorités (RFC 1984). Les exigences d'accès par ces autorités (en invoquant des arguments comme la lutte contre le terrorisme ou la protection de l'enfance) ne peuvent mener qu'à affaiblir la sécurité générale puisque ces accès seront aussi utilisés par les attaquants, ou par un État qui abuse de son pouvoir.

Le modèle de menace de l'Internet, depuis longtemps, est que tout ce qui est entre les deux machines situées aux extrémités de la communication doit être considéré comme un ennemi. Pas parce que les intermédiaires sont forcément méchants, loin de là, mais parce qu'ils ont des possibiliés techniques que certains exploiteront et il faut donc protéger la communication car on ne sait jamais ce que tel ou tel intermédiaire fera (RFC 7258 et RFC 7624). Bref, tout protocole doit chiffrer le contenu qu'il transporte (RFC 3365). Aujourd'hui, les principales exceptions à ce principe sont le très vieil whois (RFC 3912) et surtout le DNS qui a, certes, des solutions techniques pour le chiffrement (RFC 7858 et RFC 8484) mais qui sont loin d'être universellement déployées.

Ce chiffrement doit évidemment être fait de bout en bout, c'est-à-dire directement entre les deux machines qui communiquent, afin d'éviter qu'un intermédiaire n'ait accès à la version en clair. Cela pose un problème pour les services "*store-and-forward*" comme le courrier électronique (RFC 5321). De même, chiffrer lorsqu'on communique en HTTPS avec Gmail ne protège pas la communication contre Google, seulement contre les intermédiaires réseau ! Relire le RFC 7624 est recommandé.

Question vie privée, le RFC recommande également de faire attention aux métadonnées et à l'analyse de trafic. Les conseils du RFC 6973, section 7, sont ici utiles.

Un sujet encore plus délicat est celui de l'anonymat et du pseudonymat. On sait qu'il n'y a pas réellement d'anonymat sur l'Internet (quoiqu'en disent les politiciens malhonnêtes et les journalistes avides de sensationnalisme), le numérique permettant au contraire de récolter et de traiter beaucoup de traces de la communication. Néanmoins, le protocole doit permettre, autant que possible, de s'approcher de l'anonymat. Par exemple, les identificateurs persistents sont évidemment directement opposés à cet objectif puisqu'ils rendent l'anonymat impossible (rappel important : anonymat [Caractère Unicode non montré [2]] pseudonymat). Au minimum, il faudrait permettre à l'utilisateur de changer facilement et souvent ces identificateurs. Et, bien sûr, ne pas imposer qu'ils soient liés à l'identité étatique. Des exemples d'identificateurs qui sont parfois utilisés sur le long terme sont les adresses IP, les "*Connection ID*" de QUIC (un bon exemple d'un protocole qui permet leur changement facilement), les biscuits de HTTP, et les adresses du courrier électronique, certainement très difficiles à changer. Comme le montre l'exemple de ces adresses, les identificateurs stables sont utiles et on ne peut pas forcément les remplacer par des identificateurs temporaires. Ici, le souci de vie privée rentre en contradiction avec l'utilité des identificateurs, une tension courante en sécurité. Le fait qu'on ne puisse en général pas se passer d'identificateurs, à relativement longue durée de vie, est justement une des raisons pour lesquelles il n'y a pas de vrai anonymat sur l'Internet.

Notons que les politiciens de mauvaise foi et les journalistes incompétents parlent parfois d'anonymat dès qu'un identificateur stable n'est pas l'identité étatique (par exemple quand je crée un compte Gmail « anonymous652231 » au lieu d'utiliser le nom qui est sur ma carte d'identité). Mais tout identificateur stable peut finir par se retrouver lié à une autre identité, peut-être aussi à l'identité étatique, par exemple si deux identificateurs sont utilisés dans le même message. Et certains identificateurs sont particulièrement communs, avec plusieurs usages, ce qui les rend encore plus dangereux pour la vie privée. Le numéro de téléphone, que certaines messageries instantanées imposent, est particulièrement sensible et est donc déconseillé.

Donc, s'il faut utiliser des identificateurs stables, ils doivent au moins pouvoir être des pseudonymes.

D'autres façons de désanonymiser existent, par exemple quand les gens ont bêtement cru que condenser un identificateur n'était pas réversible (cf. l'article « "*Four cents to deanonymize : Companies reverse hashed email addresses*" <https://freedom-to-tinker.com/2018/04/09/four-cents-to-deanonymize-compani
> »).

Notre RFC rappelle ainsi les discussions animées qu'avait connu l'IETF en raison d'un mécanisme d'allocation des adresses IPv6, qui les faisaient dériver d'un identificateur stable, l'adresse MAC, qui permettait de suivre à la trace un utilisateur mobile. Depuis, le RFC 8981 (et le RFC 7217 pour les cas où on veut une stabilité limitée dans l'espace) ont résolu ce problème (le RFC 7721 résume le débat). À noter que l'adresse MAC peut aussi devenir variable (RFC 9724).

2. Car trop difficile à faire afficher par LaTeX

Autre exemple où un protocole IETF avait une utilisation imprudente des identificateurs, DHCP, avec ses identificateurs stables qui, certes, n'étaient pas obligatoires mais, en pratique, étaient largement utilisés (RFC 7844).

Une autre question très sensible est celle de la censure. Le protocole en cours de développement a-t-il des caractéristiques qui rendent la censure plus facile ou au contraire plus difficile (section 4.16) ? Par exemple, si le protocole fait passer par des points bien identifiés les communications, ces points vont certainement tenter les censeurs (pensez aux résolveurs DNS <https://www.bortzmeyer.org/resolveur-dns.html> et à leur rôle dans la censure <https://www.bortzmeyer.org/uptobox-orange.html>...). Les RFC 7754 et RFC 9505 décrivent les techniques de censure. Elles sont très variées. Par exemple, pour le Web, le censeur peut agir sur les résolveurs DNS mais aussi bloquer l'adresse IP ou bien, en faisant du DPI, bloquer les connexions TLS en regardant le SNI. Certains systèmes d'accès au contenu, comme Tor, ou de distribution du contenu, comme BitTorrent, résistent mieux à la censure que le Web mais ont d'autres défauts, par exemple en termes de performance. L'exemple du SNI montre en tout cas très bien une faiblesse de certains protocoles : exposer des identificateurs aux tierces parties (qui ne sont aucune des deux parties qui communiquent) facilite la censure. C'est pour cela que l'IETF développe ECH (cf. `draft-ietf-tls-esni/`).

Le protocole, tel que normalisé dans un RFC, c'est bien joli, mais il faut aussi tenir compte du déploiement effectif. Comme noté au début, ce n'est pas dans le RFC 5321 qu'on trouvera les causes de la domination de Gmail ! Les effets du protocole dans la nature sont en effet très difficiles à prévoir. La section 4.17 se penche sur cette question et demande qu'on considère, non seulement le protocole mais aussi les effets qu'il aura une fois déployé. [Ce qui, à mon avis, relève largement de la boule de cristal, surtout si on veut tenir compte d'effets économiques. Et les exemples du RFC ne sont pas géniaux, comme de reprocher au courrier sa gratuité, qui encourage le spam. L'exemple de l'absence de mécanisme de paiement sur le Web, qui pousse à développer des mécanismes néfastes comme la publicité, est meilleur.] La section 4.21 traite également ce sujet des conséquences, parfois inattendues, du déploiement d'un protocole.

Le RFC a aussi un mot (section 4.18) sur les questions d'accessibilité, notamment aux handicapés. Cette question, très présente dans les discussions autour des couches hautes du Web (cf. les réunions Paris Web) semble plus éloignée de ce que fait l'IETF mais le RFC cite quand même l'exemple du RFC 9071, sur l'utilisation de RTP dans des réunions en ligne, avec une alternative en texte pour les personnes malentendantes.

L'Internet est aujourd'hui très, trop, centralisé, notamment pour ce qui concerne les services (la connectivité, quoique imparfaitement répartie, est moins dépendante d'un petit nombre d'acteurs). Il est donc utile, lors de la conception d'un protocole, de réfléchir aux caractéristiques du protocole qui risquent d'encourager la centralisation (par exemple par la création d'un, ou d'un petit nombre de points de contrôle). Le RFC 3935 donne explicitement à l'IETF un objectif de promotion de la décentralisation.

En conclusion, même si l'activité organisée d'examen des futurs RFC n'a pas pris, ce RFC reste utile pour réfléchir à l'impact de nos protocoles sur les droits des humains.