RFC 8953 : Coordinating Attack Response at Internet Scale 2 (CARIS2) Workshop Report

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 17 décembre 2020

Date de publication du RFC : Décembre 2020

https://www.bortzmeyer.org/8953.html

Voici le compte-rendu de la deuxième édition de l'atelier CARIS <https://www.internetsociety.org/events/caris2> ("*Coordinating Attack Response at Internet Scale*"), un atelier de l'ISOC consacré à la défense de l'Internet contre les différentes attaques possibles, par exemple les dDoS. Cet atelier s'est tenu à Cambridge en mars 2019. Par rapport au premier CARIS, documenté dans le RFC 8073 [1], on note l'accent mis sur les conséquences du chiffrement, désormais largement répandu.

Les problèmes de sécurité sur l'Internet sont bien connus. C'est tous les jours qu'on entend parler d'une attaque plus ou moins réussie contre des infrastructures du réseau. Ainsi, Google a été victime d'une grosse attaque en 2017 <https://www.zdnet.com/article/google-says-it-mitigated-a-2-54-tbps-> (mais qui n'a été révélée que des années après). Mais, pour l'instant, nous n'avons pas de solution miracle. L'idée de base des ateliers CARIS est de rassembler aussi bien des opérateurs de réseau, qui sont « sur le front » tous les jours, que des chercheurs, des fournisseurs de solutions de défense, et des CSIRT, pour voir ensemble ce qu'on pouvait améliorer. Pour participer, il fallait avoir soumis un article, et seules les personnes dont un article était accepté pouvait venir, garantissant un bon niveau de qualité aux débats, et permettant de limiter le nombre de participants, afin d'éviter que l'atelier ne soit juste une juxtaposition de discours.

La section 2 du RFC présente les quatorze papiers qui ont été acceptés. On les trouve en ligne <https://www.internetsociety.org/wp-content/uploads/2019/02/CARIS2-papers.zip>.

Le but de l'atelier était d'identifier les points sur lesquels des progrès pourraient être faits. Par exemple, tout le monde est d'accord pour dire qu'on manque de professionnel·le·s compétent·e·s en cybersécurité mais il ne faut pas espérer de miracles : selon une étude <https://cybersecurityventures.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc8073.txt

com/jobs/>, il manque trois millions de personnes dans ce domaine et il n'y a simplement aucune chance qu'on puisse les trouver à court terme. Plus réaliste, l'atelier s'est focalisé sur le déploiement du chiffrement (TLS 1.3, normalisé dans le RFC 8446, le futur - à l'époque - QUIC <https://www.bortzmeyer.org/quic.html>, et pourquoi pas le TCPcrypt du RFC 8548), déploiement qui peut parfois gêner la détection de problèmes, et sur les mécanismes de détection et de prévention. Une importance particulière était donnée au passage à l'échelle (on ne peut plus traiter chaque attaque individuellement et manuellement, il y en a trop).

Bon, maintenant, les conclusions de l'atelier (section 4). Première session, sur l'adoption des normes. C'est une banalité à l'IETF que de constater que ce n'est pas parce qu'on a normalisé une technique de sécurité qu'elle va être déployée. Beaucoup de gens aiment râler contre l'insécurité de l'Internet mais, dès qu'il s'agit de dépenser de l'argent ou du temps pour déployer les solutions de sécurité, il y a moins d'enthousiasme. (J'écris cet article au moment de la publication de la faille de sécurité SadDNS <https://www.saddns.net/>. Cela fait plus de dix ans qu'on a une solution opérationnelle contre la famille d'attaques dont SadDNS fait partie, DNSSEC et, pourtant, DNSSEC n'est toujours pas déployé sur la plupart des domaines.) Commençons par un point optimiste : certaines des technologies de sécurité de l'IETF ont été largement déployées, comme SSL (RFC 6101), remplacé il y a quinze ans par TLS (RFC 8446). L'impulsion initiale venait clairement du secteur du commerce électronique, qui voulait protéger les numéros des cartes de crédit. Lié à TLS, X.509 (RFC 5280) est aussi un succès. Cette fois, l'impulsion initiale est plutôt venue des États. (X.509 doit être une des très rares normes UIT survivantes sur l'Internet.)

Moins directement lié à la sécurité, SNMP (RFC 3410) est aussi un succès, même s'il est en cours de remplacement par les techniques autour de YANG comme RESTCONF. Toujours pour la gestion de réseaux, IPfix (RFC 7011) est également un succès, largement mis en œuvre sur beaucoup d'équipements réseau.

Par contre, il y a des semi-échecs et des échecs. Le format de description d'incidents de sécurité IODEF (RFC 7970) ne semble pas très répandu. (Il a un concurrent en dehors de l'IETF, STIX - "*Structured Threat Information eXpression*", qui ne semble pas mieux réussir.) IODEF est utilisé par des CSIRT mais souffre de son niveau de détail (beaucoup d'opérationnels veulent des synthèses, pas des données brutes) et, comme toutes les techniques d'échange d'information sur les questions de sécurité, souffre également des problèmes de confiance qui grippent la circulation de l'information. Autre technique de sécurité excellente mais peu adoptée, DANE (RFC 7671). Malgré de nombreux efforts de promotion (comme , le blog que vous lisez a une note de 93 %, car la configuration TLS est tolérante) et même avec une reconnaissance légale partielle en Allemagne, DANE reste très minoritaire.

Un autre cas fameux de non-succès, même s'il n'est pas directement lié à la sécurité, est IPv6 (RFC 8200).

Deuxième session, les protocoles nouveaux. L'atelier s'est penché sur le format MUD ("*Manufacturer Usage Description*", RFC 8520) qui pourrait aider à boucher une petite partie des trous de sécurité de l'Internet des objets. Il a également travaillé l'échange de données et les problèmes de confiance qu'il pose. Comme à CARIS 1, plusieurs participants ont noté que cet échange de données reste gouverné par des relations personnelles. La confiance ne passe pas facilement à l'échelle. L'échange porte souvent sur des IOC et un standard possible a émergé, MISP.

Une fois le problème détecté, il reste à coordonner la réaction, puisque l'attaque peut toucher plusieurs parties. C'est encore un domaine qui ne passe guère à l'échelle. L'Internet n'a pas de mécanisme (technique mais surtout humain) pour coordonner des centaines de victimes différentes. Des tas d'obstacles à la coordination ont été mentionnés, des outils trop difficiles à utiliser en passant par les obstacles frontaliers à l'échange, les obligations légales qui peuvent interdire l'échange de données, et bien sûr le

problème récurrent de la confiance. Vous vous en doutez, pas plus qu'au premier atelier, il n'y aura eu de solution parfaite découverte pendant les sessions.

La session sur la surveillance a vu plusieurs discussions intéressantes. Ce fut le cas par exemple du problème de la réputation des adresses IP. Ces adresses sont souvent des IOC et on se les échange souvent, ce qui soulève des questions liées à la vie privée. (Un des papiers de l'atelier est « "*Measured Approaches to IPv6 Address Anonymization and Identity Association*" », de David Plonka et Arthur Berger , qui explique la difficulté de l'« anonymisation » des adresses IP si on veut qu'elles restent utiles pour les opérationnels.) L'exploitation correcte de ces adresses IP nécessite de connaitre les plans d'adressage utilisés (si une adresse IPv6 se comporte mal, faut-il bloquer tout le préfixe /64 ? Tout le /48 ?). Il n'y a pas de ressources publiquement disponibles à ce sujet, qui permettrait de connaitre, pour une adresse IP donnée, l'étendue du préfixe englobant. (Je ne parle évidemment pas du routage, pour lequel ces bases existents, mais de la responsabilité.) Une des suggestions était d'étendre les bases des RIR. Une autre était de créer une nouvelle base. Le problème est toujours le même : comment obtenir que ces bases soient peuplées, et correctement peuplées ?

Une des questions amusantes lorsqu'on essaie de déboguer un problème de communication entre deux applications est de savoir quoi faire si la communication est chiffrée. Il n'est évidemment pas question de réclamer une porte dérobée pour court-circuiter le chiffrement, cela créerait une énorme faille de sécurité. Mais alors comment faire pour savoir ce qui se dit ? On a besoin de la coopération de l'application. Mais toutes les applications ne permettent pas facilement de journaliser les informations importantes et, quand elles le font, ce n'est pas dans un format cohérent. D'où une suggestion lors de l'atelier de voir s'il ne serait pas envisageable de mettre cette fonction dans les compilateurs, pour avoir un mécanisme de journalisation partout disponibles.

Pendant qu'on parle de chiffrement, une autre question est celle de l'identification d'une machine ou d'un protocole par le "*fingerprinting*", c'est-à-dire en observant des informations non chiffrées (taille des paquets, temps de réponse, variations permises par le protocole, etc). Le "*fingerprinting*" pose évidemment des gros risques pour la vie privée et beaucoup de travaux sur des protocoles récents (comme QUIC) visaient à limiter son efficacité.

Pour résumer l'atelier (section 6 du RFC), plusieurs projets ont été lancés pour travailler sur des points soulevés dans l'atelier. À suivre, donc.