# Les conséquences techniques de l'interception HTTPS en entreprise

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 28 février 2017

https://www.bortzmeyer.org/https-interception.html

---

Le 28 février 2017, à la conférence NDSS 17 <https://www.internetsociety.org/events/ndss-symposium/ndss-symposium-2017> à San Diego, Zakir Durumeric a présenté les conclusions de la recherche faite avec Zane Ma, Drew Springall, Richard Barnes, Nick Sullivan, Elie Bursztein, Michael Bailey, J. Alex Halderman et Vern Paxson, « "*The Security Impact of HTTPS Interception*" <https://jhalderm.com/pub/papers/interception-ndss17.pdf> ». Ils montraient que la pratique, très répandue en entreprise, de l'interception (en fait, le détournement) des sessions HTTPS des utilisateurs, outre son côté immoral, a de graves conséquences pour la sécurité.

De quoi s'agit-il ? Normalement, la session HTTPS est protégée de bout en bout. Le client se connecte au serveur et le protocole TLS se fait directement entre client et serveur. Les intermédiaires (par exemple les routeurs) ne voient qu'un flux chiffré qu'ils ne peuvent pas comprendre (confidentialité via le chiffrement), ni modifier (intégrité). Dans un certain nombre d'entreprises et d'organisations, par exemple étatiques, la direction souhaite au contraire pouvoir examiner le trafic des employés, par exemple pour savoir qui écrit au Canard Enchaîné. La technique pour cela consiste à intercepter le trafic HTTPS, le déchiffrer, et le rechiffrer avec le serveur. Lorsqu'il sera en clair, dans l'équipement de surveillance, on pourra l'examiner.

Mais, attendez, dit la lectrice qui connait TLS (RFC 5246 [1]) : ça ne peut pas marcher. La session n'est pas juste chiffrée, elle est authentifiée. Le serveur doit présenter un certificat, et il chiffre avec la clé contenue dans ce certificat. L'équipement d'interception ne peut pas montrer un certificat qui convienne.

Le truc réside dans une énorme faille du système X.509 : n'importe quelle autorité de certification peut produire un certificat pour n'importe quel nom de domaine, même si le titulaire de ce nom a choisi une autre AC. Il suffit donc d'avoir une AC à sa disposition dans le magasin de certificat de la machine.

1. Pour voir le RFC de numéro NNN, https://www.ietf.org/rfc/rfcNNN.txt, par exemple https://www.ietf.org/rfc/rfc5246.txt

C'est ainsi par exemple que le ministère des finances avait fait un vrai/faux certificat <http://www.01net.com/actualites/comment-le-ministere-des-finances-espionne-le-trafic-web-de-ses-
html> pour gmail.com...

Je ne parlerai pas ici des aspects moraux, politiques ou juridiques de la surveillance via interception HTTPS. L'article des chercheurs qui a fait l'objet de l'exposé d'aujourd'hui se focalisait sur les problèmes techniques. Ils ont étudié cette interception des deux côtés : en observant le trafic chez certains serveurs Web, et en étudiant certaines des boîtes noires qui font cette interception. Côté serveur, pour voir si le trafic est intercepté, ils regardaient surtout s'il y avait une différence entre le `User-Agent` : HTTP (RFC 7231, section 5.5.3) annoncé et les paramètres TLS. Si un navigateur se présente comme étant Firefox mais annonce l'extension « battement de cœur » du RFC 6520, on est sûr qu'il s'agit d'une interception : Firefox n'a jamais mis en œuvre cette extension. Des tas d'autres heuristiques peuvent être utilisées, comme l'ordre des extensions TLS dans le `ClientHello` du client.

Leur étude montre que plus de 10 % des sessions HTTPS vers Cloudflare (6 % pour des sites divers de commerce en ligne) sont interceptées, ce qui est assez inquiétant.

Mais il y a bien pire : le système d'interception, on l'a vu, termine la session TLS et en commence une autre avec le serveur. Ce faisant, la totalité des systèmes testés font d'énormes erreurs TLS : ils annoncent des algorithmes de chiffrement abandonnés (RC4, cf. RFC 7465), ou qui n'auraient jamais dû être utilisées (les algorithmes « exportation », délibérement affaiblis), acceptent des certificats expirés, et, parfois, ils ne valident même pas le certificat du serveur ! Ils sont en outre vulnérables à plusieurs attaques TLS connues. Cela est dû au fait que ces boîtes noires utilisent des versions anciennes de bibliothèques TLS, et qu'elles ne les configurent pas proprement. (Ce problème avait déjà été démontré avec les anti-virus <https://www.bortzmeyer.org/killed-by-proxy.html>.)

Rien d'étonnant à cela : ces boîtes noires sont achetées par des gens qui n'y connaissent rien, qui n'évaluent pas le logiciel, et pour qui la sécurité est un produit qu'on achète. (Je viens de lire un guide « L'essentiel de la sécurité numérique pour les dirigeants » qui recommande de dépenser « de 3 % à 10 % du budget informatique [pour la] cybersécurité », comme si la sécurité dépendait de l'argent dépensé !)

Autre point à noter : ces boîtes noires de surveillance sont toutes en logiciel privateur (et sont donc populaires auprès de dirigeants qui se disent « le logiciel libre, ce n'est pas professionnel et "*enterprise-grade*" ») mais utilisent très souvent du logiciel libre en dessous (sans prendre la peine d'utiliser des versions récentes).

Donc, employés, la prochaine fois que vous entendez dire qu'on a déployé l'interception HTTPS pour votre bien, « pour des raisons de sécurité », méfiez-vous : cette pratique **diminue** la sécurité.

Dans la série des bonnes lectures, notez que l'ANSSI a un guide sur l'interception HTTPS <https:
//www.ssi.gouv.fr/administration/guide/recommandations-de-securite-concernant-lanalyse
>. Notez que le US-CERT a également sonné l'alarme <https://www.us-cert.gov/ncas/alerts/
TA17-075A> contre ces interceptions HTTPS. La recherche présentée à NDSS portait sur le côté client des boîtiers intercepteurs mais leur côté serveur est tout aussi bogué comme le montre l'exemple BlueCoat <https://bugs.chromium.org/p/chromium/issues/detail?id=694593>.