Un bel exemple de logiciels de sécurité dangereux

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 17 juillet 2016

https://www.bortzmeyer.org/killed-by-proxy.html

On répète souvent aux utilisateurs que l'Internet est un lieu dangereux (ce n'est pas faux) et qu'il faut utiliser des logiciels qui les protègent magiquement : anti-virus, logiciels de contrôle parental, etc. Mais ce sont des logiciels, ils ont donc des bogues et ils ne sont pas mieux écrits que la moyenne des logiciels. Leurs bogues peuvent sérieusement affecter la sécurité de la machine. Morale : ajouter du logiciel de sécurité n'améliore pas forcément la sécurité.

Pour ceux qui ne seraient pas convaincus de ces évidences, je recommande très fortement la lecture de l'excellent article « "*Killed by Proxy : Analyzing Client-end TLS Interception Software*" <http://users.encs.concordia.ca/~mmannan/publications/ssl-interception-ndss2016.pdf> ». Les auteurs, Mohammad Mannan et Xavier de Carné-Carnavalet, ont testé en labo un certain nombre de logiciels qui font de l'« interception TLS » et découvert que la plupart ouvraient des boulevards à des attaquants. Qu'est-ce qu'un logiciel d'interception TLS ? C'est un logiciel qui est un relais TLS, entre le logiciel de l'utilisateur (typiquement un navigateur Web) et le vrai serveur. L'intercepteur se fait passer pour le vrai serveur auprès du navigateur Web et pour le client auprès du vrai serveur. Pour ne pas lever d'alerte de sécurité dans le navigateur, il présente un certificat valable. Ce genre de logiciel est donc un détournement délibéré du modèle de sécurité de TLS : il casse la sécurité exprès. Il n'est donc pas étonnant qu'ils ouvrent des failles graves.

Pour comprendre ces failles, un petit mot sur le fonctionnement des ces logiciels : ils tournent sur la machine de l'utilisateur (contrairement aux relais TLS que les grandes entreprises et les administrations installent souvent <https://blog.mozilla.org/security/2013/12/09/revoking-trust-in-one-anssi-ce> près de l'accès Internet, pour surveiller les "*malwares*" et espionner les employés), ils détournent le trafic TLS (typiquement HTTPS) et ils présentent un certificat valable pour le nom de domaine demandé. Ce certificat a pu être généré en usine ou bien à l'installation du logiciel. Ce certificat est parfois protégé par une phrase de passe. Pour que le certificat soit accepté, ils mettent leur propre AC dans le magasin du système, avec une période de validité allant jusqu'à 20 ans. (En général, ils n'expliquent pas clairement à l'utilisateur ce qu'ils font, ce qui augmente encore le danger.) Le logiciel d'interception reçoit les connexions locales, venant du navigateur Web, et se connecte lui-même aux vrais serveurs distants. Ces logiciels sont presque toujours privateurs, aucun accès au code source (bien que tous utilisent, sans honte, un logiciel libre, OpenSSL), aucun moyen de les vérifier.

La **totalité** des logiciels testés par les auteurs a au moins une faille TLS. Inutile donc de chercher le « bon antivirus » ou le « bon logiciel de contrôle parental ». Voici une liste non limitative de ces failles :

— Acceptation de certificats **externes** (présentés par le vrai serveur) signés par l'AC du logiciel d'interception. Un pirate qui peut faire signer ces faux certificats peut donc faire accepter n'importe quoi aux machines ayant installé ces logiciels.
— Comportements bizarres lors de l'expiration de la licence, comme d'accepter soudainement tous les certificats externes, sans les valider.
— Non-suppression de leur AC du magasin lorsqu'on exécute le programme de désinstallation du logiciel. On ne peut donc jamais réellement faire marche arrière.
— Clés privées sécurisant le certificat stockées sans protection, ou avec une protection identique pour toutes les installations du logiciel, permettant à un logiciel malveillant (sans privilèges Administrateur) d'y accéder et de faire donc ensuite accepter n'importe quel serveur TLS mensonger.
— Modification des erreurs TLS lorsque le certificat **externe** (celui du vrai serveur) est invalide. Par exemple, un certificat dont le nom (le sujet) ne correspond pas devient un certificat signé par une AC inconnue, ce qui sera le message affiché par le navigateur.
— Bien pire, la plupart des logiciels acceptent presque tous les certificats externes invalides, masquant toute erreur.
— Sécurité cryptographique plus faible qu'un vrai navigateur (par exemple, acceptation de certificats qui utilisent MD5, vulnérabilité à Poodle, etc).
— Acceptation d'AC qui auraient dû être retirées depuis longtemps, comme DigiNotar.

Pourquoi les sociétés qui écrivent ces logiciels feraient des efforts pour la sécurité ? Leurs utilisateurs ne sont pas des connaisseurs, des tas de gens auto-proclamés experts en sécurité servent de vendeurs pour ces logiciels en répétant aux utilisateurs « pensez à installer un anti-virus » et la non-disponibilité du code source rend difficile toute analyse de ces logiciels.