QUIC : le protocole de Google rend-il le web moins sûr ?

Entre HTTPS et QUIC, perd-on au change en matière de sécurité ? Oui… sur certains aspects, affirment des chercheurs chinois.

Le premier de ces protocoles est désormais bien implanté. La majorité des sites web l'ont adopté (environ 70 % selon W3Techs, 80 % selon Let's Encrypt…). Le second commence à peine à s'installer (5 %). Il faut dire qu'il est plus jeune. Son développement avait débuté en 2012 sous l'impulsion de Google. Le groupe américain en pousse actuellement la standardisation auprès de l'IETF. Il a franchi un cap l'an dernier en l'activant par défaut – à la place de HTTPS – dans son navigateur Chrome.

Comme HTTPS, QUIC implémente le protocole TLS pour sécuriser les échanges. En revanche, il ne s'appuie pas sur TCP, mais sur UDP. Objectif : réduire les délais de négociation client-serveur.

Multiplexage optimisé, meilleur contrôle du flux… Les chercheurs ne remettent pas en cause les performances supérieures de QUIC. Ils déplorent, en revanche, sa plus grande fragilité face au *fingerprinting*. C'est-à-dire la possibilité, pour des tiers, de déterminer les sites web visités en interceptant le trafic chiffré.

QUIC : une fragilité précoce

Ce n'est pas la première étude à traiter du sujet. Mais elle a, assurent ses auteurs, la particularité d'une approche en deux phases. Et cela change tout ou presque au niveau des résultats.

La phase qualifiée de « normale » comprend toutes les étapes de rendu d'une page web. Celle qualifiée d'« initiale » (*early*) n'en comprend qu'une partie. En l'occurrence, l'obtention du fichier HTML sur le serveur et l'analyse des balises CSS et JavaScript. Elle n'inclut pas les étapes ultérieures, à savoir l'obtention des fichiers CSS et JavaScript, leur analyse, leur combinaison avec le fichier HTML, puis l'analyse des balises multimédias et la récupération des contenus correspondants.

Principale conclusion : si les deux protocoles présentent un niveau de vulnérabilité similaire sur la phase « normale », QUIC se révèle plus exposé que HTTPS sur la phase « initiale ».

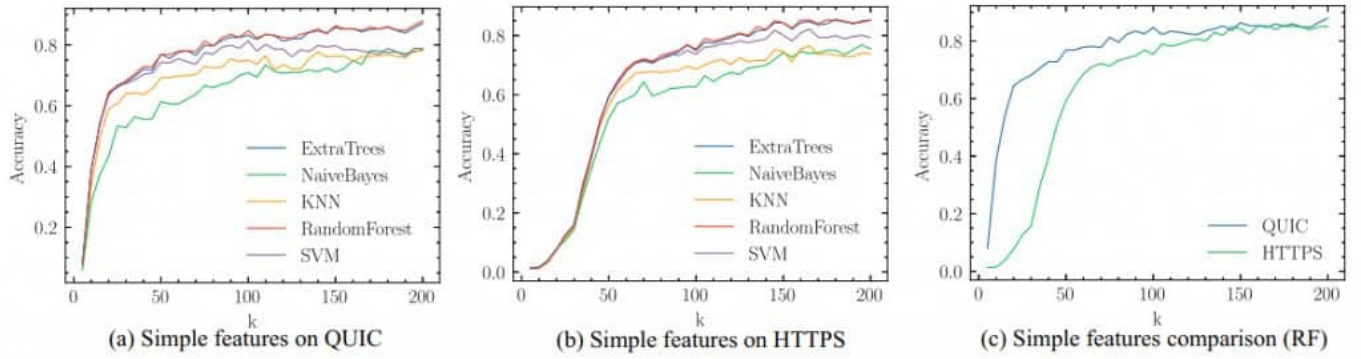
Les chercheurs reconnaissent les limites de leur méthodologie. Ils ont, en particulier, « assaini » le trafic qui leur a servi de base d'étude. Entre autres par le recours à un environnement fermé (LAN), à l'ouverture d'une page à la fois et à des adaptations de type désactivation du cache sur Chrome.

Deux protocoles, deux représentations

Deux dimensions entrent en jeu pour le *fingerprinting*. D'une part, le nombre de paquets interceptés. De l'autre, le type d'informations qu'on en tire. L'étude en distingue deux types. Le premier (« Simple ») inclut deux éléments qu'on trouve dans l'en-tête : la direction des paquets (vers le client ou vers le serveur) et leur taille. Le second (« Transfer ») englobe des métriques de type

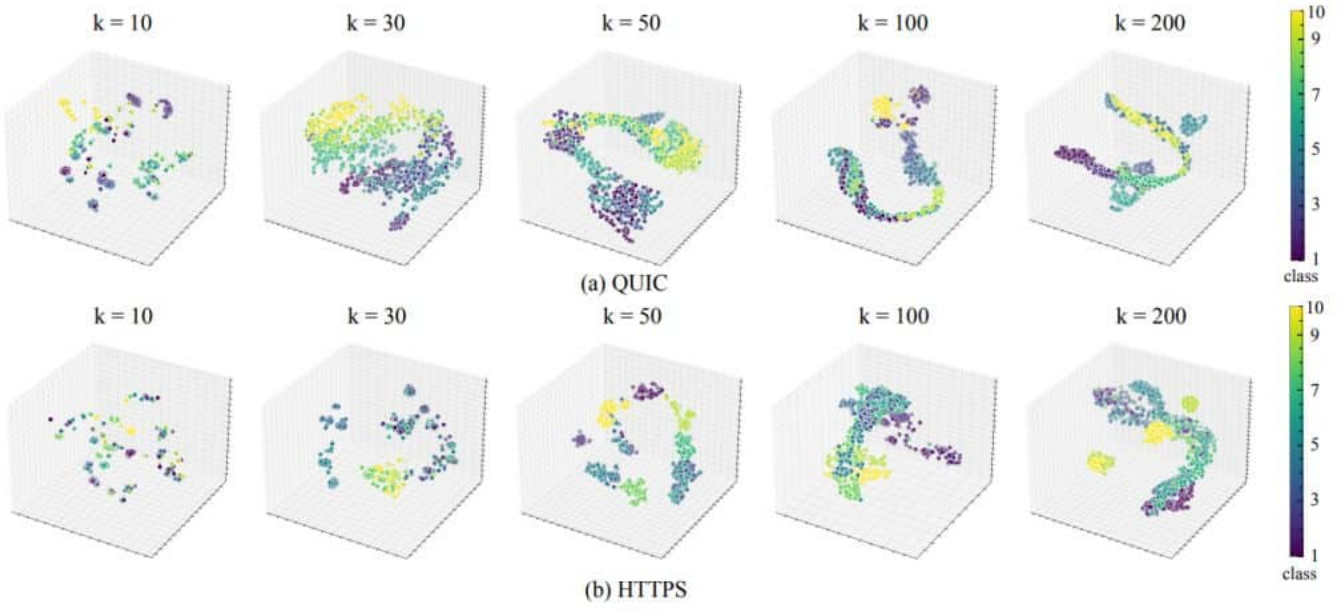
nombre de paquets de taille x, intervalle entre l'arrivée de deux paquets, nombre de paquets successifs dans la même direction, etc.

Le taux de précision (sites correctement prédits) se révèle plus élevé en phase amont sur la base des informations « Simple ». Il l'est aussi, bien que moins sensiblement, avec les métriques « Transfer ». L'écart se resserre ensuite à mesure qu'on intercepte des paquets.



(a) Simple features on QUIC (b) Simple features on HTTPS (c) Simple features comparison (RF)

Ce différentiel s'illustre aussi dans l'espace latent des deux protocoles. Il est plus aisé de distinguer les informations dans celui de QUIC.



(a) QUIC

(b) HTTPS

Dans ce contexte, les attaques contre QUIC peuvent être plus rapides, moins coûteuses… et laisser le temps de mettre en place d'autres attaques.

Illustration principale © Rawpixel.com – stock.adobe.com