# RFC 8773 : TLS 1.3 Extension for Certificate-Based Authentication with an External Pre-Shared Key

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 31 mars 2020

Date de publication du RFC : Mars 2020

https://www.bortzmeyer.org/8773.html

---

L'authentification dans TLS se fait typiquement, soit à partir d'un certificat, soit par une clé partagée à l'avance. Ce nouveau RFC spécifie une extension de TLS qui permet d'utiliser ces deux méthodes en même temps.

Rappelons d'abord qu'il y a deux sortes de clés partagées à l'avance (PSK, pour "*Pre-Shared Key*") : celles qui ont été négociées dans une session précédentes ("*resumption PSK*") et celles qui ont été négociées par un mécanisme extérieur (envoi par pigeon voyageur sécurisé…), les "*external PSK*". Ce RFC ne concerne que les secondes. Les certificats et les clés partagées à l'avance ont des avantages et des inconvénients. Les certificats ne nécessitent pas d'arrangement préalable entre client et serveur, ce qui est pratique. Mais il faut se procurer un certificat auprès d'une AC. Et les certificats, comme ils reposent sur des algorithmes comme RSA ou ECDSA, sont vulnérables aux progrès de la cryptanalyse, par exemple en utilisant un (futur) ordinateur quantique. Utiliser une clé partagée à l'avance n'est pas forcément commode (par exemple quand on veut la changer) mais cela peut être plus sûr. Or, la norme TLS actuelle (RFC 8446 [1]) ne permet d'utiliser qu'une seule des deux méthodes d'authentification. Si on les combinait ? L'ajout d'une clé externe permettrait de rendre l'authentification plus solide.

Le principe est simple : notre RFC spécifie une extension à TLS, `tls_cert_with_extern_psk` (valeur 33 <https://www.iana.org/assignments/tls-extensiontype-values/tls-extensiontype-values.xml#tls-extensiontype-values-1>.) Le client TLS l'envoie dans son `ClientHello`. Elle indique la volonté de combiner certificat et PSK. Elle est accompagnée d'extensions indiquant quelle est la clé partagée à utiliser. Si le serveur TLS est d'accord, il met l'extension `tls_cert_with_extern_psk`

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc8446.txt`

dans son message `ServerHello`. (Le serveur ne peut pas décider seul de l'utilisation de cette extension, il faut que le client ait demandé d'abord.)

Les clés ont une identité, une série d'octets sur lesquels client et serveur se sont mis d'accord avant (PSK = "*Pre-Shared Key*", clé partagée à l'avance.) C'est cette identité qui est envoyée dans l'extension `pre_shared_key`, qui accompagne `tls_cert_with_extern_psk`. La clé elle-même est bien sûr un secret, connu seulement du client et du serveur (et bien protégée : ne la mettez pas sur un fichier lisible par tous.) Voyez la section 7 du RFC pour une discussion plus détaillée de la gestion de la PSK.

Une fois que client et serveur sont d'accord pour utiliser l'extension, et ont bien une clé en commun, l'authentification se fait via le certificat (sections 4.4.2 et 4.4.3 du RFC 8446) **et** en utilisant ensuite, non pas seulement la clé générée (typiquement par Diffie-Hellman), mais la combinaison de la clé générée et de la PSK. L'entropie de la PSK s'ajoute donc à celle de la clé générée de manière traditionnelle.

Du point de vue de la sécurité, on note donc que cette technique de la PSK est un strict **ajout** à l'authentification actuelle, donc on peut garantir que son utilisation ne diminuera pas la sécurité.

Il n'y a apparemment pas encore de mise en œuvre de cette extension dans une bibliothèque TLS.

Notez qu'il s'agit apparemment du premier RFC à mentionner explicitement les calculateurs quantiques, et les risques qu'ils posent pour la cryptographie.