TLS 1.3 Extension for Certificate-based Authentication with an External Pre-Shared Key

draft-housley-tls-tls13-cert-with-extern-psk

Russ Housley

TLS WG at IETF 102

July 2018

TLS 1.3 Authentication and Key Schedule

Initial Handshake:

Authentication:	**Key Schedule Secret Inputs:**
Signature and Certificate	(EC)DHE

Subsequent Handshake:

Authentication:	**Key Schedule Secret Inputs:**
Resumption PSK	Resumption PSK + (EC)DHE
Resumption PSK	(EC)DHE

This Extension Adds Another Choice

Initial Handshake:

Authentication:	Key Schedule Secret Inputs:
Signature and Certificate	(EC)DHE
Signature and Certificate	External PSK + (EC)DHE

Subsequent Handshake:

Authentication:	Key Schedule Secret Inputs:
Resumption PSK	Resumption PSK + (EC)DHE
Resumption PSK	(EC)DHE

External PSK for Quantum Protection

- Open question whether a large-scale quantum computer is feasible, and if so, when it might happen
- If it happens, (EC)DHE becomes vulnerable
- The concern …
  - Today: Adversary saves TLS 1.3 handshake and the associated ciphertext
  - Someday: Decrypt communications when a large-scale quantum computer becomes available
- The solutions …
  - Near-term: Strong external PSK as an input to the TLS 1.3 key schedule
  - Long-term: Quantum-resistant public-key cryptographic algorithms (the winners of NIST competition)

Extension Overview

Extension Syntax

- The successful negotiation of the "tls_cert_with_extern_psk" extension requires the TLS 1.3 key schedule processing to include *both* the selected external PSK and the (EC)DHE shared secret value; it also requires the server to send the Certificate and CertificateVerify messages in the handshake
- The "tls_cert_with_extern_psk" extension is always be used along with the already defined "key_share", "psk_key_exchange_modes", and "pre_shared_key" extensions
- The "psk_key_exchange_modes" extension will always offer psk_dhe_ke
- The "pre_shared_key" extension used with obfuscated_ticket_age of zero

- Inclusion of the extension is willingness to authenticate the server with a certificate and include an external PSK in the key schedule processing:

```
struct {
    select (Handshake.msg_type) {
        case client_hello: Empty;
        case server_hello: Empty;
    };
} CertWithExternPSK;
```

Allow Certificates with External PSK

- TLS 1.3 does not permit the server to send a CertificateRequest message when a PSK is being used; this restriction is removed when the "tls_cert_with_extern_psk" extension is negotiated
  - Allows external PSK, and
  - Allow client and server authentication with certificates
- TLS 1.3 does not permit an external PSK to be used in the same fashion as a resumption PSK; this extension does not alter those restrictions
- Likewise, a certificate *still* MUST NOT be used with a resumption PSK

A Few Thoughts About External PSKs

- Group external PSKs must be distributed in a manner that does not depend on current public key cryptography
- Pairwise external PSKs for every client and server is not feasible
- A group, such as an enterprise or organization, can manage an external PSK
  - Invention of a large-scale quantum computer means that the group members might be able to perform decryption
  - Parties outside the group remain unable to decrypt
- External PSKs are more suitable for some applications of TLS 1.3 than others

The Ask

- TLS WG adopt the Internet-Draft: draft-housley-tls-tls13-cert-with-extern-psk

- Review and comment on the Internet-Draft