Exported Authenticators

Nick Sullivan
Exported Authenticators

• New concept along the lines of TLS exporters (RFC 5705)
• Allows binding of new certificates to existing TLS connections
• Inspired by HTTP/2 secondary authentication
HTTP/2 Reactive Client Authentication

• draft-bishop-httpbis-http2-additional-certs-02

• HTTP 1.1 reactive client authentication
  • TLS 1.2 uses renegotiation
  • TLS 1.3 uses post-handshake authentication

• HTTP/2 does not interact well with renegotiation due to multiplexing

• Add certificate authentication into HTTP/2 using special frames
  • Need a way to do certificate proof and link certificate proof to TLS connection
HTTP/2 Connection Coalescing

• TLS sessions are reused if both the following are fulfilled
  • IP address of both domains match
  • Certificate covers both Subject Alternative Names

• ORIGIN frame allows bypass of IP restriction

• HTTP/2 Secondary Authentication allows bypass of certificate restriction
Initial attempt

- draft-sullivan-tls-post-handshake-auth-00
- Generalization of existing TLS post-handshake authentication
  - Allowing both spontaneous and elicited post-handshake client and server authentication

| *CertificateRequest → | ← Certificate, CertificateVerify, Finished |
| ← *CertificateRequest | Certificate, CertificateVerify, Finished → |
Initial attempt

• Criticisms
  • Additional complexity during initial negotiation
  • Acknowledging CertificateRequest creates buffering, additional work to calculate finished
    • NOTE: This criticism also applies to post-handshake client authentication
Exported Authenticators

- Like the post-handshake auth, but messages are exported instead of sent as TLS messages
- No state kept in TLS handshake, only exported messages

- Message structure
  Certificate, CertificateVerify, Finished
TLS Tunnel
Open Questions

• Formal security proof needed along lines of Krawczyk’s SIGMAC
• Is it an issue that this does not get updated when there is a key update?
Exported Authenticators

Nick Sullivan