TLS alternatives – faster and more secure

Tanja Lange
Technische Universiteit Eindhoven

26 April 2015
TLS: 4 roundtrips

1. DNS lookup.
2. TCP three-way handshake establishes random initial sequence number:
   - Weak authenticator/liveness check
   - Address late packet arrival
3–4. Negotiate cipher suite and establish ephemeral keys

Slide credit: Mike Petullo.
TLS: 4 roundtrips

1. DNS lookup.

Slide credit: Mike Petullo.
TLS: 4 roundtrips

1. DNS lookup.

2. TCP three-way handshake establishes random initial sequence number:
   - Weak authenticator/liveness check
   - Address late packet arrival

3–4. Negotiate cipher suite and establish ephemeral keys

Application data

Slide credit: Mike Petullo.
TLS: 4 roundtrips

3–4. Negotiate cipher suite and establish ephemeral keys

Slide credit: Mike Petullo.
TLS (abbreviated): 2 roundtrips

Slide credit: Mike Petullo.
TLS: 4 roundtrips

Slide credit: Mike Petullo.
CurveCP: Usable security for the Internet, DJB, 2010

\[
\begin{align*}
\text{ClientHello (} C', 0, \text{Box}[0'](C' \rightarrow S)) \\
\text{Resumption Cookie (Box}[S', K](S \rightarrow C')) \\
\text{Initiate, vouch, & data (} C', K, \text{Box}[C, V, N, \ldots](C' \rightarrow S')) \\
\text{Application data (Box}[\ldots](S' \rightarrow C')) \\
\text{Application data (Box}[\ldots](C' \rightarrow S')) \\
\end{align*}
\]

\(c', C':\) client’s short-term private & public key
\(s, S:\) server’s long-term private & public key
\(s', S':\) server’s short-term private & public key
\(\text{Box}[m](C' \rightarrow S):\) encrypt \(m\) using DH key obtained from \(c'\) and \(S\)
\(K = \text{Box}[C', s'](t):\) encrypt \(m\) under minute key \(t\).
If no long-term \(C:\) skip \(V\), else \(V = \text{Box}[C'](C \rightarrow S)\).
MinimaLT: Minimal Latency Tunneling, CCS’13

(Petullo, Zhang, Solworth, Bernstein, Lange implementation for linux and ethos involves more UIC students.)

MinimaLT objects

- **Public keys**: identify client users and servers
- **Ephemeral keys**: time-based and cryptographically protect identifying traffic
- **Tunnels**: an encrypted channel between two hosts which multiplexes connections
- **Connections**: user-authenticated two-way communication within a tunnel
- **Directory and name service**: resolve hostnames to IP addresses and keys
MinimaLT roundtrips

Obtaining $D$’s ephemeral key (only at boot time):

$$
\begin{array}{c}
C' & \text{Conn., req. ephemeral key} & D \\
\text{Ephemeral key} & \leftarrow & T1
\end{array}
$$

DNS-like lookup (only if tunnel does not yet exist):

$$
\begin{array}{c}
C' & \text{Conn., req. server information} & D' \\
\text{IP address, UDP port, key, ephemeral key of $S$} & \leftarrow & T2
\end{array}
$$

Connection establishment:

$$
\begin{array}{c}
C' & \text{Connect, application-to-service RPC} & S' \\
\end{array}
$$

Plus: Lots of cool crypto in rekeying, proof-of-work puzzles, resumption at different IP address.
QUIC: Quick UDP Internet Connections

Design Document and Specification Rationale, Roskind
QUIC Crypto, Langley, Chang

- 0 RTT achieved similar to current 0 RTT proposals:
  - Client tries previous (stored) server key.
  - If key still active, get 0 RTT. Else data in first packet (or prior to ACK) is lost.
  - Server can reply with fresh key (1 RTT, 2 RTT worst case).

- Con: Less radical change than MinimaLT.

- Con: Favors long key validity, so bad for forward secrecy.

- Pro: Actually in use for Chrome – Google connections.
  Chromium Blog: Results so far are positive, with the data showing that QUIC provides a real performance improvement over TCP

- Less cool stuff on rekeying.