SSL/TLS & 3D Secure

BİL 448/548
Internet Security Protocols
Ali Aydın Selçuk
Brief History of SSL/TLS

• SSLv2
  – Released in 1995 with Netscape 1.1
  – Key generation algorithm kept secret
  – Reverse engineered & broken by Wagner & Goldberg

• SSLv3
  – Fixed and improved, released in 1996
  – Public design process

• TLS: IETF’s version; the current standard
  – Latest: v1.2 (RFC 5246, August 2008)

• DTLS (Datagram TLS): TLS over UDP
## SSL Architecture

- **Record Protocol**: Message encryption/authentication
- **Handshake P.**: Identity authentication & key exchange
- **Alert P.**: Error notification (cryptographic or otherwise)
- **Change Cipher P.**: Activate the pending crypto suite

### Protocol Stack

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SSL Record Protocol</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TCP</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IP</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SSL Alert Protocol**

**SSL Change Cipher Spec. Protocol**

**SSL Record Protocol**

**HTTP, etc.**
Basic SSL/TLS Handshake Protocol

Alice

hello, crypto offered, \( R_A \)

certificate, crypto selected, \( R_B \)

\( \{S\}_{\text{Bob}}, \{\text{keyed hash of messages}\} \)

\( (K = f(S, R_A, R_B)) \)

\( \{\text{keyed hash of messages}\} \)

session keys derived from \( K \)

Bob
SSL/TLS Handshake Protocol

• Client authentication:
  – Bob can optionally send “cert. req.” in msg 2.
  – Then, Alice will send her certificate in msg 3 as well as her signature on all messages so far.

• DH is supported too (fixed, ephemeral, or anon.)

• DH is gaining popularity (ECDH, in particular) over RSA key transport. Especially after Snowden.

• IETF is to drop RSA key transport from TLS 1.3.

http://www.theinquirer.net/inquirer/news/2343117/ietf-drops-rsa-key-transport-from-ssl
DH SSL/TLS Handshake Protocol

hello, crypto offered, $R_A$

$[g^x]_B$, certificate, crypto selected, $R_B$

$g^y$, {keyed hash of messages}

$(K = f(S, R_A, R_B), \text{where } S = g^{xy})$

{keyed hash of messages}

session keys derived from $K$
DH in SSL/TLS

• Main advantage: PFS

• Disadvantage: More costly
  – RSA key transport: One exponentiation per side
  – Ephemeral DH: Three exponentiations per side

• But, with new ECC algorithms (e.g., EC25519), the overhead has become almost negligible.
  https://www.imperialviolet.org/2013/05/10/fastercurve25519.html
SSL Session vs. Connection

• “Sessions” are relatively long-lived.
• Multiple “connections” (TCP) can be supported under the same SSL session.
• To start a connection, Alice can send an existing session ID.
• If Bob doesn’t remember the session ID Alice sent, he responds with a different value.
Session Resumption ("Connection")

- Alice sends session-id, crypto offered, $R_A$ to Bob.
- Bob responds with session-id, crypto selected, $R_B$, {keyed hash of messages}.
- Session keys are derived from $K$, $R_A$, $R_B$.
- Both parties agree on the session keys.
Key Computation

• “pre-master secret”: S
• “master secret”: $K = f(S, R_A, R_B)$
• For each connection, 6 keys are generated from $K$ and the nonces.
  (3 keys for each direction: encryption, auth., IV)
• Implicit IVs are abandoned in TLS 1.1 and later: 
Negotiating Crypto Suites

- **Crypto suite**: A complete package specifying the crypto to be used. (encryption algorithm, key length, integrity algorithm, etc.)
- 30+ predefined standard cipher suites.
- 256 values reserved for private use.
- Selection:
  - v2: Alice proposes a set of suites; Bob returns a subset of them; Alice selects one. (which doesn’t make much sense)
  - v3: Alice proposes a set of suites; Bob selects one.
The Trust Model

- PKI: Oligarchy model with X.509 certificates
- Browsers come configured with a set of trusted root CAs (VeriSign, AT&T, Entrust/Nortel, etc.) Additions to the root CA list by user is possible.
- Typically, only the server is authenticated. Client authentication is optional.
- Certificate revocation: Two alternative protocols:
  - CRL
  - OCSP
Secure Electronic Transaction (SET)

- Application-layer e-commerce protocol
- Developed by Visa & MasterCard consortium, 1996
- Provides security, authentication, order transaction, payment authorization, etc.
- Both the merchant & customer are authenticated by X.509 certificates
• Problems of e-commerce over SSL/TLS:
  – malicious merchants (stealing credit card numbers)
  – malicious customers (using stolen credit card no.s)

• SET solution:
  – Bank (B) acts as an intermediary between the customer (C) & the merchant (M)
  – M forwards C’s info. to B, encrypted with B’s key
  – B does:
    • authenticate C’s public key signature
    • decrypt the transaction info. (amount, card number, etc.)
    • issue payment authorization & send it to M
SET & 3D-Secure

• SET problem: All users are required to have public keys & “wallets”.
  – difficult to deploy & expensive
  – not convenient (user access from a single terminal)

• 3D-Secure solution:
  – No wallets are required.
  – B authenticates C by password (or, SMS-OTP).
  – M directs C to B, to which password is SSL-encrypted.
    (Problem: Malicious merchants can do m.i.t.m. attack, directing C to a fake page it controls.)
  – Officially launched in 2003, supported by Visa & MC.