

Secure Socket Layer (SSL) and Transport Layer Security (TLS)

Raj Jain

Washington University in Saint Louis
Saint Louis, MO 63130

Jain@cse.wustl.edu

Audio/Video recordings of this lecture are available at:

<http://www.cse.wustl.edu/~jain/cse571-07/>



- History and overview of SSL/TLS
- Products and Implementations
- Datagram Transport Layer Security (DTLS)
- Current TLS Issues and Extensions
- Secure Remote Password (SRP)

First part from the textbook. Remainder from Wikipedia and IETF

Key Features

- ❑ User level ⇒ Not operating system specific
- ❑ Uses TCP ⇒ Reliable transmission
(No retransmissions at application layer)
- ❑ Features:
 - Crypto negotiation
 - Key Generation for encryption and Integrity
 - Authentication:
 - ❑ Servers use Certificates
 - ❑ Clients use password or certificates

SSL/TLS Applications

- ❑ HTTPS = HTTP over port 443
- ❑ FTPS = FTP over SSL
(different from SFTP = FTP over SSH)
- ❑ NNTP over SSL
- ❑ OpenVPN

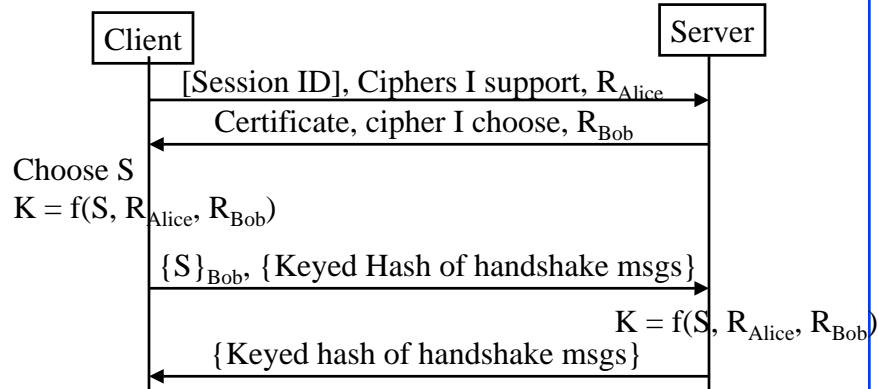
History

- ❑ Secure Socket Layer (SSL) V2 on Netscape Navigator 1.1 1995
- ❑ Private Communication Technology (PCT) by Microsoft fixed some bugs in SSL V2
- ❑ SSL v3 is most commonly deployed protocol
- ❑ Transport Layer Security (TLS) by IETF [RFC 2246 1999]
- ❑ TLS v1.1 [RFC 4346 2006]
- ❑ TLS v1.2 [draft-ietf-tls-rfc4346-bis-05.txt June 2007]

SSL v2 vs. v3

- ❑ **Downgrade Attack:** Crypto choices not protected in V2. Finished message in v3 contains digest of all previous messages
- ❑ **Truncation Attack:** V2 closes SSL on TCP connection close \Rightarrow Not protected. V3 added session finished message to close SSL session.

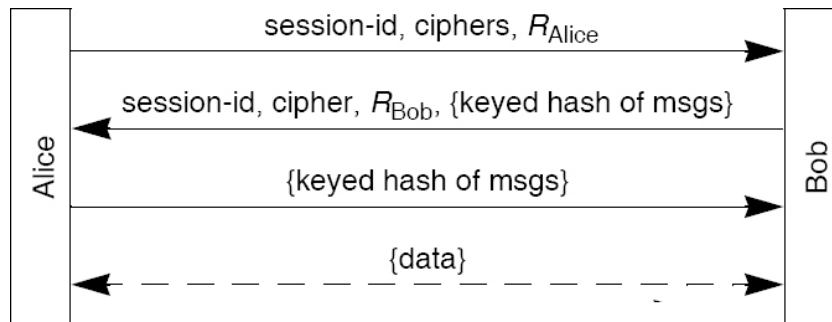
SSL/TLS Basic Protocol



- ❑ R's are 32B. First 4B = Unix time
- ❑ Secrets: Pre-master secret S , master secret K
- ❑ 6 Keys: Encryption, Integrity, IV (1 per direction)
- ❑ Authenticates server. Client authenticated by password.

Session Resumption

- ❑ Similar to Phase 2 of IKE
- ❑ Multiple session keys from master secret K
- ❑ HTTP 1.0 used many TCP connections
- ❑ Server stores session ID and master secret



Version

- ❑ 0.2 ⇒ SSL v2
- ❑ 3.0 ⇒ SSL v3
- ❑ 3.1 ⇒ TLS v1
- ❑ V3 clients send v2 client-hello with version 3.0
- ❑ V2 servers respond with v2 server-hello
- ❑ V3 servers respond with a v3 server-hello

Cipher Suites

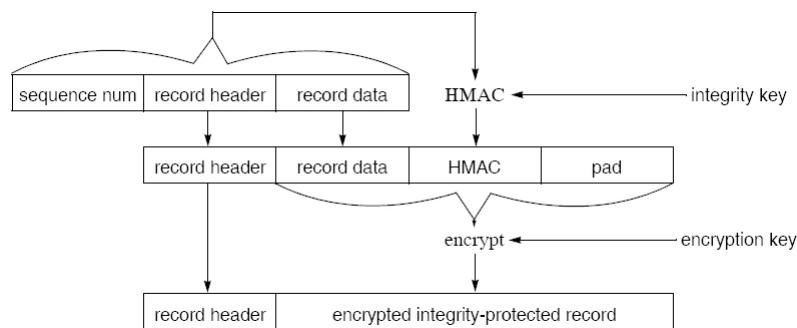
- ❑ V3 has a 2B field for cipher suite
- ❑ Standard numbers for 30 Cipher suites, e.g.,
SSL_RSA_EXPORT_WITH_DES40_CBC_SHA
- ❑ Server decides one of the choices offered by Client
- ❑ **Crypto Algorithms**
 - Key exchange: RSA, Diffie-Hellman, DSA, SRP, PSK
 - Symmetric ciphers: RC4, Triple DES, AES or Camellia.
 - Hash function: HMAC-MD5 or HMAC-SHA

Export Issues

- ❑ Only 40 bits keys allowed.
- ❑ Servers can encrypt keys using 512b RSA keys.
- ❑ Normally RSA keys are 1024b. 512b Ephemeral key.
- ❑ Server Gated Cryptography/Step-Up:
Financial transactions allowed to use longer keys.
- ❑ Server certificates signed by Verisign or Thawte contain SGC extension allowed.
- ❑ Initial handshake using 40b.
- ❑ Client would then send Change Cipher Spec message to renegotiate.

Encrypted Records

- ❑ Integrity is provided by HMAC using the integrity key
- ❑ Data prefixed by 64b sequence # but the sequence # not sent
- ❑ Block cipher \Rightarrow 40B padding in SSLv3, 44B in TLS.
- ❑ Final block of each record is used as IV for the next



Encoding

- ❑ All exchanges are in records up to 2^{14} B or 2^{16} -1B.
- ❑ Standard allows multiple messages in one record or multiple records.
- ❑ Most implementations use one message per record.
- ❑ Four Record Types:
 - 20 = Change Cipher Spec
 - 21 = Alerts (1 = Warning, 2 = Fatal)
 - 22 = Handshake
 - 23 = Application Data
- ❑ Record header:

| | | |
|-------------|-----------|--------|
| Record Type | Version # | Length |
| 1B | 2B | 2B |
- ❑ Each message starts with a 1B message-type and 3B message length.

Handshake Messages

- 1 = Client Hello: Version, R_{Alice} , Session ID, Cipher Suites, Compressions
- 2 = Server Hello: Version, R_{Bob} , Session ID, Chosen Cipher, Chosen Compression
- 14 = Server Hello Done
- 16 = Client Key Exchange: Encrypted pre-master key
- 12 = Server Key Exchange: Modulus p , Exponent g , Signature (export only)
- 13 = Certificate Request: CA Names (requested by server)
- 11 = Certificate: sent by server
- 15 = Certificate Verify: signature of Hash of messages
- 20 = Handshake Finished: MD5 and SHA Digest of message halves

TLS Message Exchange



15-15

Alerts

- 0 Close notify (warning or fatal)
- 10 Unexpected message (fatal)
- 20 Bad record MAC (fatal)
- 21 Decryption failed (fatal, TLS only)
- 22 Record overflow (fatal, TLS only)
- 30 Decompression failure (fatal)
- 40 Handshake failure (fatal)
- 41 No certificate (SSL v3 only) (warning or fatal)
- 42 Bad certificate (warning or fatal)
- 43 Unsupported certificate (warning or fatal)
- 44 Certificate revoked (warning or fatal)
- 45 Certificate expired (warning or fatal)

15-16

Alerts (Cont)

- 46 Certificate unknown (warning or fatal)
- 47 Illegal parameter (fatal)
- 48 Unknown CA (fatal, TLS only)
- 49 Access denied (fatal, TLS only)
- 50 Decode error (fatal, TLS only)
- 51 Decrypt error (TLS only) (warning or fatal)
- 60 Export restriction (fatal, TLS only)
- 70 Protocol version (fatal, TLS only)
- 71 Insufficient security (fatal, TLS only)
- 80 Internal error (fatal, TLS only)
- 90 User cancelled (fatal, TLS only)
- 100 No renegotiation (warning, TLS only)

SSL Products and Implementations

□ Acceleration:

- Offload public key encryption/decryption
- Sometimes all SSL message
- H/W from F5, Cisco, Nortel, Juniper, Radware, ...

□ Software:

- OpenSSL: C library of SSL/TLS
- GnuTLS: C Library under GNU Public license
- Java Secure Socket Extension (JSSE)
- Network Security Services (NSS): Open source security library includes SSL also

Datagram Transport Layer Security

- ❑ TLS runs on TCP
 - ⇒ Suitable for stream-oriented applications
 - ⇒ Not suitable for datagram applications
- ❑ DTLS uses UDP
- ❑ Need timeout, retransmission, fragmentation
- ❑ Some state is kept in the messages
- ❑ Explicit sequence number
- ❑ As close to TLS as possible
- ❑ RFC 4347, April 2006

TLS: Current Issues

- ❑ TLS V1.2
- ❑ Transport Layer Security (TLS) Extensions:
Extension Definitions
- ❑ Using Secure Remote Password (SRP) protocol for
TLS Authentication
- ❑ Using OpenPGP keys for TLS authentication
- ❑ TLS Elliptic Curve Cipher Suites with SHA-256/384
and AES Galois Counter Mode
- ❑ RSA based AES-GCM Cipher Suites for TLS

TSL V1.1

- ❑ RFC 4346, April 2006
- ❑ IV = Final Block of each record (in V1). Implicit IV to prevent CBC attacks
- ❑ Padding errors ⇒ Bad Record MAC alert
⇒ Prevents CBC attacks
- ❑ Sessions resumeable after premature TCP closes
- ❑ Informational notes on TLS attacks

TLS V1.2

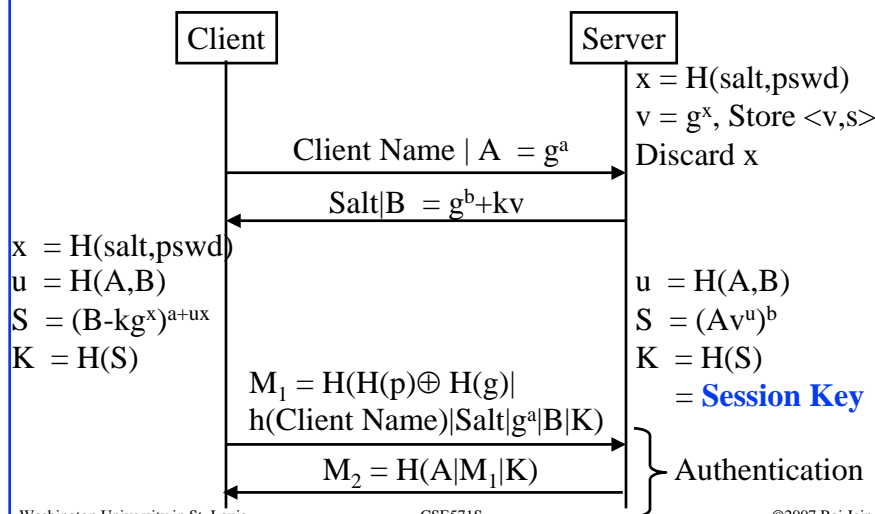
- ❑ draft-ietf-tls-rfc4346-bis-05.txt, Sep 2007
- ❑ Merged TLS extensions
- ❑ Replacement of MD5/SHA-1 combination
- ❑ Client specifies hash functions choices
- ❑ Server selects hash function
- ❑ Authenticated encryption with additional data modes
- ❑ Tighter checking of encrypted pre master secret version numbers
- ❑ Info on implementation pitfalls

TLS Extensions

- ❑ draft-ietf-tls-rfc4366-bis-00.txt, June 2007
- ❑ Server Name Indication: Clients can indicate the virtual server they are contacting
- ❑ Maximum Fragment Length Negotiation:
- ❑ Client Certificate URLs
- ❑ Trusted CA Indication: from clients
- ❑ Truncated HMAC: Save bandwidth
- ❑ Certificate Status Request: Send OCSP URL

Secure Remote Password (SRP)

All clients and server know $g, p, k = H(p, g)$



SRP

- ❑ Resistant to dictionary attacks
- ❑ Does not require trusted third party
- ❑ No client certificates
- ❑ Currently SRP V6 being standardized in IEEE 1363. V3 described in RFC 2945, Sept 2000.

Summary



- ❑ SSLv3 allows crypto negotiation, server authentication and key exchange. Uses PKI.
- ❑ TLS extensions allow using SRP and shared secrets
- ❑ DTLS = TLS over UDP \Rightarrow Allows UDP applications
- ❑ Secure remote password (allows) authentication is stronger than simple password hashes

Homework 15

- ❑ Read chapter 19 of the textbook and Wikipedia
- ❑ Submit answer to the following exercise
- ❑ **Exercise 19.3:** What is the advantage, in the exportable SSLv3 case, of hashing the 40-bit secret with two non-secret values to produce a 128-bit key? How many keys would have to be tested to brute-force break a single session?