Sicurezza sul Web

HTTP Fundamentals

- RFC 1945 – HTTP 1.0
- RFC 2616 – HTTP 1.1
- RFC 2396 – URL/URI syntax
- www.w3.org - World Wide Web Consortium (W3C) - Check this site regularly

Tim Berners-Lee

Biography
http://www.ibiblio.org/pioneers/lee.html
http://www.w3.org/People/Berners-Lee/

Interview With Christopher Lydon
http://media.skybuilders.com/Lydon/Berners-Lee.1.mp3

HTTP Fundamentals

- Traditional Client/Server Model
- Listens on port 80
- Glorified FTP server
- HTTP transmits resources rather than files
- Universal Resource Locator (URL) – a subset of URI

HTTP Fundamentals

- A request line has three parts, separated by spaces: a *method* name, the local path of the requested resource, and the version of HTTP being used.

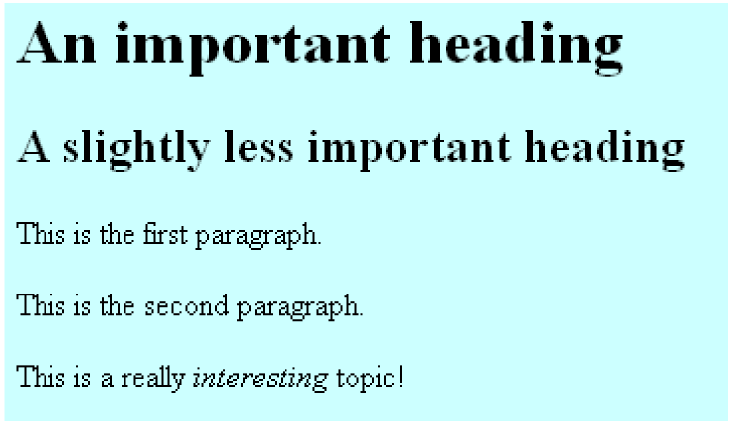
```
GET /path/to/file/index.html HTTP/1.0
```

- Other methods: HEAD and POST

HTML Fundamentals

- <h1>An important heading</h1>
- <h2>A slightly less important heading</h2>
- <p>This is the first paragraph.</p>
  <p>This is the second paragraph.</p>
- This is a really <em>**interesting**</em> topic!

HTML Fundamentals



An important heading

A slightly less important heading

This is the first paragraph.

This is the second paragraph.

This is a really *interesting* topic!

Famous Web Attacks

- "These cyber assaults have caused millions of Internet users to be denied services. At this time we are not aware of the motives behind these attacks. But they appear to be intended to disrupt legitimate electronic commerce." –*Janet Reno in response to a series of DoS attack in early 2000.*

Famous Web Attacks

- The Royal Canadian Mounted Police have charged a teenage computer hacker in one of the February cyber attacks that crippled several popular Web sites. The suspect is a 15-year-old boy known online by the nickname "Mafiaboy" – *FOX News, 4/19/2000*

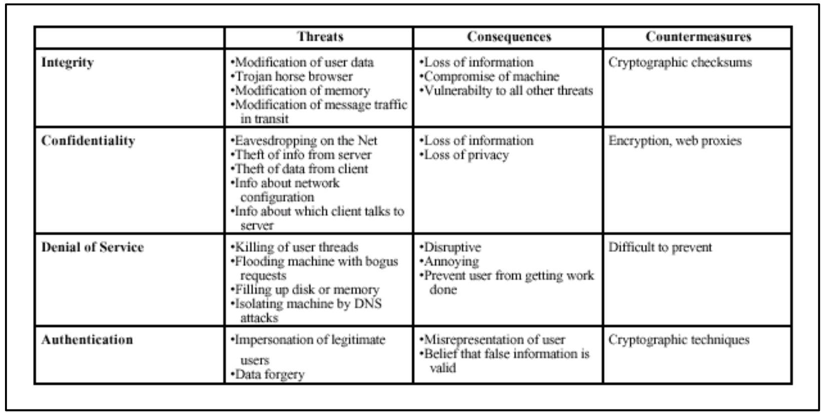
Famous Web Attacks

- A 17-year-old New Hampshire computer junkie known as "Coolio" may be charged in a handful of vandalism incidents at private and government Web sites according to U.S. federal law enforcement sources. Coolio hacked into and defaced three Web sites: D.A.R.E., an anti-drug organization; Internet security company RSA Security; and the U.S. government's Chemical Weapons Convention site, FBI sources said. – *Reuters, 3/3/2000*

Considerazioni su Web Security

- Internet is two way – a differenza delle tradizionali forme di pubblicazione di informazioni,
  - Ciò aumenta la vulnerabilità
- High visibility – determina l'immagine pubblica, la reputazione, è legata ai diritti d'autore
- Complex software – il protocollo è semplice, ma l'applicazione client/server è complessa
- Vulnerability point – un web server può essere il punto da cui lanciare ulterori attacchi

Web Security Threats

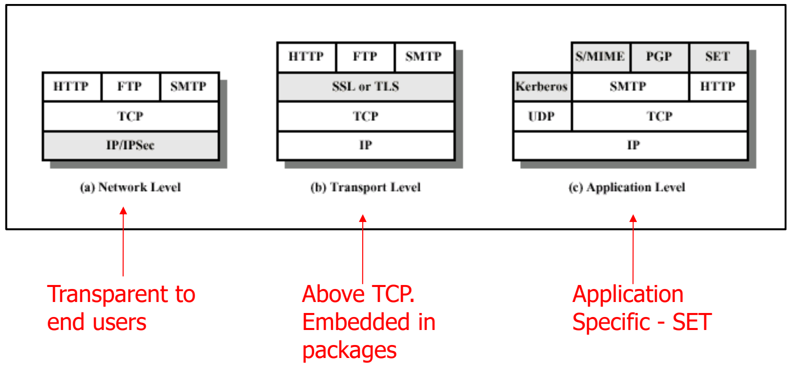


	Threats	Consequences	Countermeasures
Integrity	•Modification of user data •Trojan horse browser •Modification of memory •Modification of message traffic in transit	•Loss of information •Compromise of machine •Vulnerabilty to all other threats	Cryptographic checksums
Confidentiality	•Eavesdropping on the Net •Theft of info from server •Theft of data from client •Info about network configuration •Info about which client talks to server	•Loss of information •Loss of privacy	Encryption, web proxies
Denial of Service	•Killing of user threads •Flooding machine with bogus requests •Filling up disk or memory •Isolating machine by DNS attacks	•Disruptive •Annoying •Prevent user from getting work done	Difficult to prevent
Authentication	•Impersonation of legitimate users •Data forgery	•Misrepresentation of user •Belief that false information is valid	Cryptographic techniques

Web Traffic Security Approaches

- Classificare le minacce mediante la locazione: web server, web browser e network traffic
- Ci concentriamo sul *traffic*
- IPsec
- Secure Sockets Layer (SSL)
- Transport Layer Security (TLS)
- Secure Electronic Transaction (SET)

Web Security Approaches



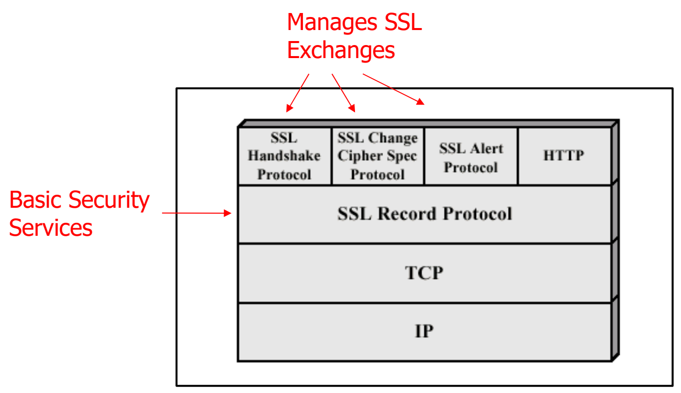
SSL Origins

- Originated by Netscape
- Competed with SHTTP
- Version 3 became Internet draft
- TLS (Transport Layer Security) is an attempt to develop a common standard
- SSLv3.1 = TLS

SSL Architecture

- Dipende da TCP per quanto riguarda l’affidabilità end-to-end
- Due livelli dei protocolli:
  - SSL Record Protocol – fornisce servizi di sicurezza di base ai livelli superiori
  - Three higher layer protocols - used in the management of SSL exchanges

SSL Protocol Stack



SSL Architecture/Concepts

- Connection – relazione *peer-to-peer* nel *transport layer*. Ogni connessione è associata a una sessione
- Session – *un'associazione* tra un client e un server creata da *Handshake Protocol*
  - Definisce un insieme di parametric per la crittografia, condivisi tra più connessioni
  - Evita la negoziazione di nuovi parametri per ogni connessione

SSL Statefullness

- Più connessioni sicure all'interno di una sessione
- Numero degli stati associato ad ogni sessione
- Current operating state for read and write (receive and send)
- Pending read and write states created during Handshake Protocol

Session State

- Session identifier – arbitrary byte sequence chosen by the server
- Peer certificate – X.509.v3 digital certificate of peer; may be null
- Compression method
- Cipher spec – algorithms used (AES, MD5)
- Master secret – 48 byte shared key
- Is resumable – session can be used to initiate new connections

Connection State

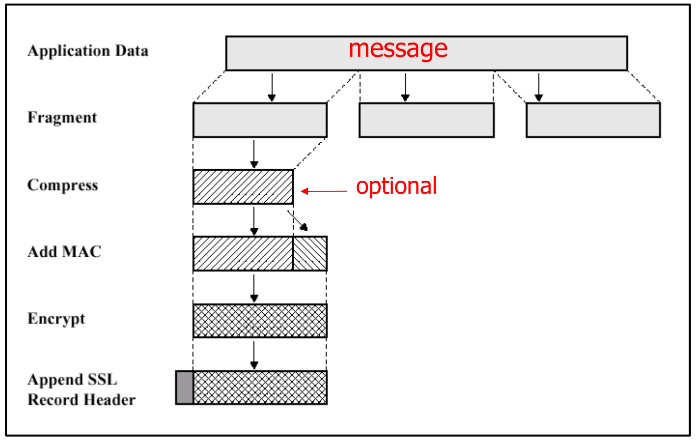
- Server and client random – byte sequences chosen for each connection
- Server/Client write MAC secret – secret key used in MAC operations on data sent by the server/client
- Server/Client write key – conventional encryption key
- Initialization vectors – needed for CBC mode
- Sequence numbers – separate for xmit & recv

SSL Record Protocol

Provides ***two important services*** for SSL connections:

- Confidentiality – Handshake Protocol defines a secret key for conventional encryption of SSL payloads
- Integrity – Handshake Protocol defines a shared secret key used to form a message authentication code (MAC)

SSL Record Protocol Ops



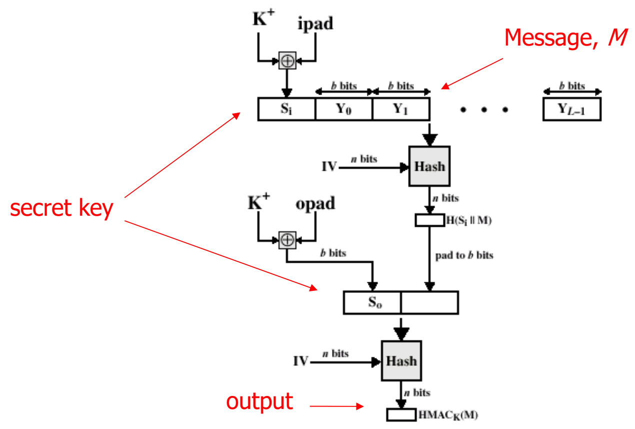
SSL Record Protocol Ops

- Fragmentation – block of 16K bytes or less
- Compression – optional, must not increase content length beyond 1024 bytes
- Message authentication code (MAC) – uses shared secret key, similar to HMAC algorithm

Recall: HMAC

- Effort to develop a MAC derived from a cryptographic hash code
- Executes faster in software
- No export restrictions
- Relies on a secret key
- RFC 2104 list design objectives
- Used in IPsec

HMAC Structure



By passing S_i and S_o through the hash algorithm, we have pseudoradomly generated two keys from *K*.

SSL Record Protocol Ops

- Message authentication code (MAC) – two pads are concatenated in SSLv3 but XORed in HMAC
- SSLv3 was based on original internet draft for HMAC, which used concatenation
- hash(secret_key || 0x5C_pad || hash(secret_key || 0x36_pad || seq_num || compress_type || length || fragment))

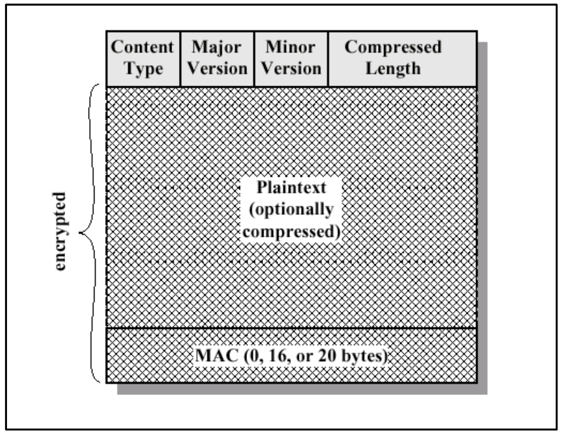
SSL Record Protocol Ops

- Compressed message plus the MAC are encrypted using symmetric encryption
- Can’t increase content length by more than 1K bytes
- May use padding – for cipher block
- IDEA, DES, 3DES, Fortezza (NSA product)

SSL Record Protocol Ops

- Final step is to prepend a header with following fields:
  - Content type – the higher layer protocol used to process the enclosed fragment
  - Major version – SSLv3
  - Minor version – value of 0
  - Compressed length – plaintext fragment length in bytes

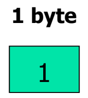
SSL Record Format



Content Types

Four types:

- Change Cipher Spec – simplest protocol consists of a single byte message that *causes the pending state to be copied into the current state* which *updates cipher suite* to be used

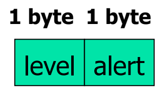


Change Cipher Spec Protocol

Content Types

Four types:

- Alert – 2 byte protocol used to convey SSL related alerts to the peer entity. 1st byte is either a warning or fatal, which terminates the connection. 2nd byte indicates specific alert



Alert Protocol

Content Types

Four types:

- Application Data – this is opaque data to SSL. No distinction made among the various applications

≥1 byte
opaque content

Other upper-layer protocol (e.g., HTTP)

Content Types

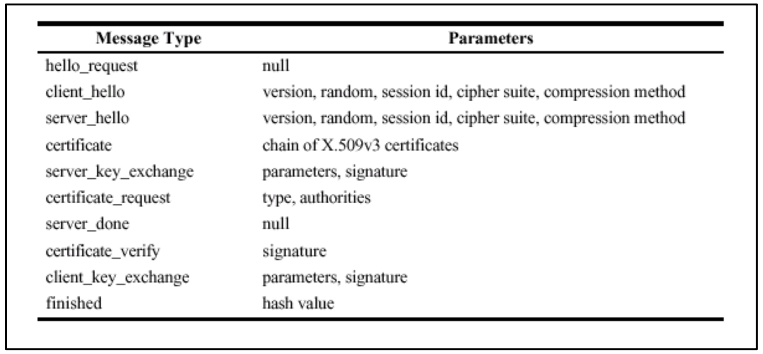
Four types:

- Handshake – allows server and client to authenticate each other and negotiate and encryption and MAC algorithm. Used before any application data is transmitted. Consists of a series of messages

1 byte	3 bytes	≥0 bytes
type	length	content

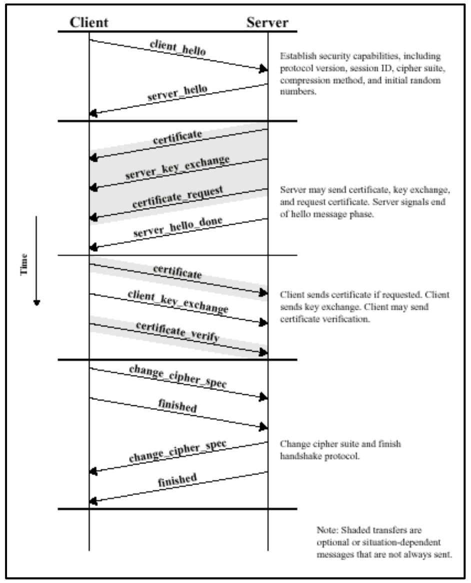
Handshake Protocol

Handshake Protocol Message Types



Message Type	Parameters
hello_request	null
client_hello	version, random, session id, cipher suite, compression method
server_hello	version, random, session id, cipher suite, compression method
certificate	chain of X.509v3 certificates
server_key_exchange	parameters, signature
certificate_request	type, authorities
server_done	null
certificate_verify	signature
client_key_exchange	parameters, signature
finished	hash value

Handshake Protocol Action



Establish security capabilities, including protocol version, session ID, cipher suite, compression method, and initial random numbers.

Server may send certificate, key exchange, and request certificate. Server signals end of hello message phase.

Client sends certificate if requested. Client sends key exchange. Client may send certificate verification.

Change cipher suite and finish handshake protocol.

Note: Shaded transfers are optional or situation-dependent messages that are not always sent.

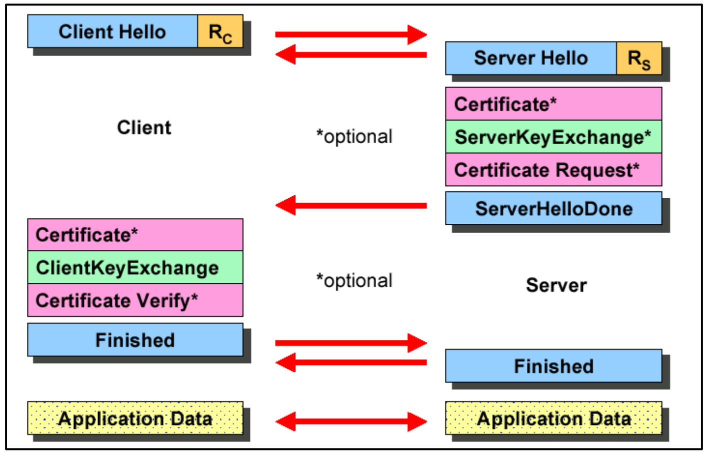
Phase 1

Phase 2

Phase 3

Phase 4

Handshake Protocol



Handshake Protocol – Phase 1

- Initiate a logical connection and establish security capabilities
- Client send client_hello message with nonce, session ID, cipher suite (decreasing order of preference), compress method
- Server returns server_hello message with nonce and selection of proposed parameters
- Key exchanges: RSA | fixed, ephemeral, or anonymous Diffie-Hellman | Fortezza

Handshake Protocol – Phase 2

- Most of this is optional
- Server sends it's certificate (X.509s) if it needs to be authenticated
- server_key_exchange message is sent. This is a hash which includes nonces to prevent replay attacks
- Server can send a certificate_request message to the client
- Finally the server_done message (no parms) is always sent by the server to indicate the end of hello, authentication and exchange message
- Server waits for client response

Handshake Protocol – Phase 3

- Client now verifies the certificate if requested and checks parameters
- A certificate message is sent if server requests it
- client_key_exchange message sent to exchange keys
- certificate_verify message may be sent to verify the client's ownership of the private key for the client certificate

Handshake Protocol – Phase 4

- Completes the setting up of a secure connection
- Client sends a change_cipher_spec message and copies the pending CipherSpec into the current CipherSpec
- Client sends finished message under the new algorithm, keys and secrets
- In response to these two messages, the server does the same
- Handshake is complete and the client and server may begin to exchange application layer data

Cryptographic Computations

- Master Secret Creation – two stages: pre-master-secret exchange (RSA or Diffie-Hellman) and master secret computation by both sides
- Generation of Cryptographic Parameters – the master-secret is a seed value for functions that generate the client/server MAC secret, keys, and IV

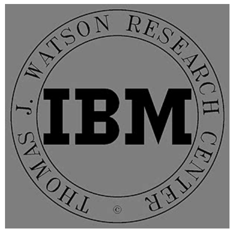
Transport Layer Security

- TLS is an Internet standard to replace SSLv3
- Defined in RFC 2246
- Record format is the same as SSL Record Format
- TLS makes use of HMAC (padding bytes are XORed)

Transport Layer Security

- PRF, pseudorandom function, expands small shared secrets into longer blocks of data. Uses two hash functions (RSA & SHA-1) for added security
- Similar alert codes to SSL with a few new additions
- Cipher suites are the same except for Fortezza (not supported)

Digital Watermarks



Watermark

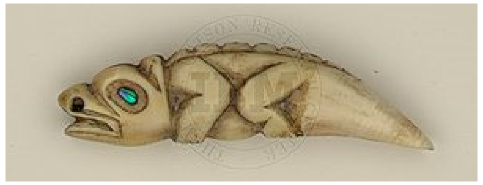


Image with watermark

Digital Watermarks

- Complements the cryptographic processes
- Visible or invisible identification code that is permanently embedded in the multimedia data
- Removal of the watermark is virtually impossible
- Composed of a bit pattern distributed throughout the data based on noise theory
- Causes no visual aural degradation of the image

Important URLs

- http://docs.sun.com/source/816-6156-10/contents.htmIntroduction to SSL from Netscape
- http://www.openssl.org/
  A very good open source version
- http://www.ietf.org/html.charters/tls-charter.htmlIETF TLS WOrkgroup
- http://www.forensics.nl/digital-watermarking
  Good collection of digital watermarking papers

Network Security

Web Security – Part 2

Secure Electronic Transaction

- Matercard & Visa – 1996
- SET is an open encryption and security specification designed to protect credit card transactions on the Internet
- Microsoft, Netscape, RSA, Versign
- 1998 – first set of SET compliant products

Secure Electronic Transaction

- SET is not a payment system
- Set of security protocols enabling the use of the existing credit card payment infrastructure over the Internet in a secure fashion
- Three services:
  - Secure communications channel
  - Trust through X.509v3 certificates
  - Ensures privacy

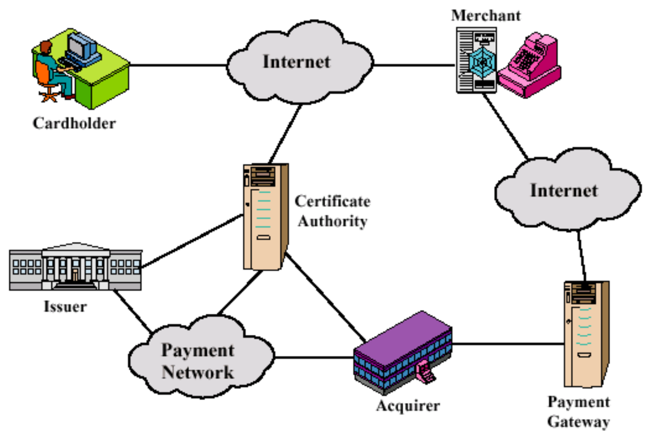
SET Requirements – Book 1

- Provide confidentiality of payment & ordering – encryption
- Ensure integrity of data – digital signatures
- Verify cardholder is legitimate user of a valid account – signatures and certificates
- Ensure use of best security practices – well tested specification
- Protocol is independent of transport security mechanisms – “raw” TCP/IP, IPSec, or SSL
- Interoperability among software & network providers – independent of platforms & OS

SET Features

- Confidentiality of information – prevents the merchant from learning the cardholder’s credit card number; conventional encryption
- Integrity of data – guarantees that message contents are not altered in transit; RSA digital signatures
- Cardholder account authentication – merchants can verify that cardholder is a legitimate user; X509 certificates
- Merchant authentication – cardholders can verify that a merchant has a relationship with a financial institution

Secure Electronic Commerce Components



3-D Secure

- 3-D Secure is a XML-based protocol to allow authentication of cardholders of credit card companies in ePayment transactions. The protocol was developed by Visa and was adopted under the names Verified By Visa and Mastercard Secure Code.
- Visa 3-D Secure Payment Program