Washington System Center

RDS Training
Secure Socket Layer (SSL) Overview

z/Series Security (Mary Sweat, Greg Boyd)
Advanced Technical Support
Gaithersburg, MD

August 25, 2005

Purpose

- **Provide a communication protocol**
  - allows a session to be established between two parties, a client and a server
    - provide privacy (encryption), authentication of the communicating partner and data integrity of the information exchanged on the connection
      - security is based on negotiated agreement between these two parties
  - may be used on an application-by-application basis

SSL/TLS : Functions

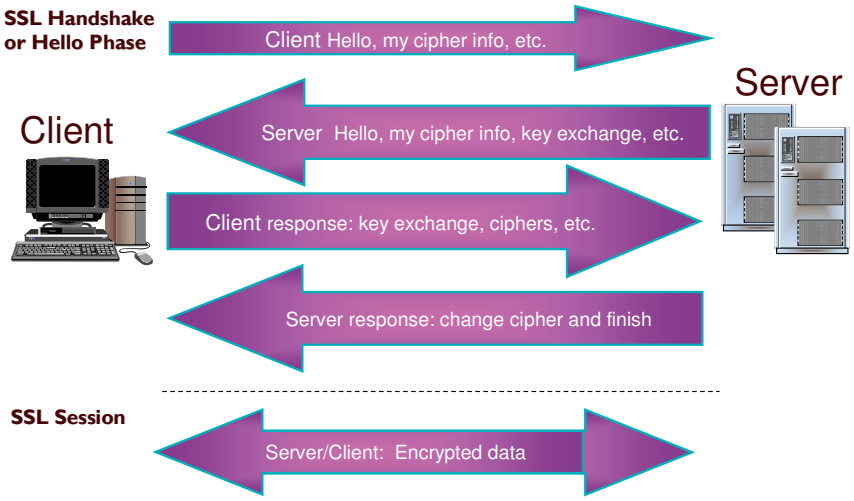
Server

1. provides information and data to the client at the client's request
2. decides what data should be protected
3. is usually an application written to provide data services outbound
4. has the responsibility to protect its identity (will prove its identity via a certificate)

Client

1. initiates the communications
2. generally selects the data to be provided by the Server
3. most are browsers but not necessarily
4. can prove its identity by also having a certificate

The SSL/TLS Session



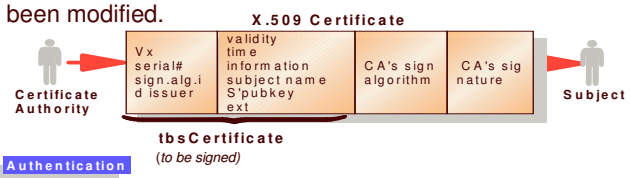
Certificates

- **Certificates are a way of securely identifying someone**
  - most are based on the standard structure X.509 v3
  - certificates are encoded using DER rules (X.209)
  - Contains;
    - Owner's distinguished name
    - Owner's public key
      - Signature algorithm with which the public key is used
    - issuers distinguished name
      - issuers signature

V#, SN , CA's signature, sgn-alg
Issuer name: CAxyz
Validity Dates and Time type
Subject name: Greg
Subject's Public Key , AlgoID
SignAlgo: RSA with SHA-1
Extensions

Certificate Authorities

- **Certificate authorities are trusted organizations who vouch for public keys**
  - a CA is an entity trusted by both the client and the owner
  - CA issues a credential (a certificate) to the owner, that associates the owner's name with the owner's public key
    - client trusts that the CA will not issue a certificate to an imposter
    - public keys are delivered via certificates, which are signed with the private key of the certificate authority
  - client can validate the certificate at any time
    - validating a certificate proves that it's authentic and has not been modified.



Public Key Cryptography – Mathematically Related

■ Generate 2 prime numbers (each over 100 digits long)	P = 7 Q = 17
■ Multiply primes to get modulus, N	N = 7 x 17 = 119
■ Select odd number, E, that will be the second part of the public key	E = 5
■ Public Key (N E)	119 5
■ Compute second part of private key, D	
(P-1) x (Q-1) x (E-1)	(7-1) x (17-1) x (5-1) = 384
Add 1 to result	384 + 1 = 385
Divide by E to get D	D = 385/5 = 77
■ Private Key (N D)	119 77

Encipher Message – 'SELL'

■ P = 7; Q = 17; N = 119; E = 5; D = 77	
■ Public Key (N E)	119 5
■ Private Key (N D)	119 77
■ Convert characters to numeric	
• E.g. a=1, b=2, c=3 ….	
• Plaintext 'SELL' becomes 19 5 12 12	
■ Raise that character value to power E	('S' => 19**5 => 2476099)
■ Divide by first part of Public Key	2476099 / 119 = 20807
■ And get the remainder	66 = eKP(S)
■ Ciphertext	66 31 3 3

Decipher Message – ’66 31 3 3’

- P = 7; Q = 17; N = 119; E = 5; D = 77
- Public Key (N E) 119 5
- Private Key (ND) 119 77

- Raise to power D 66 ** 77 = 1273…..
- Divide result by modulus N 1273….. / 119 = 1069
  And get remainder 19
- Remainder is numeric equivalent of character sent 19 = “S”
- Plaintext 19 5 12 12 or C’S E L L’

Where are the SSL functions executed?

Function	z800/z900	z890/z990	z9 109
Handshake Phase			
CSNDPKD – Public Key Decrypt	PCICA, PCICC, CCF	PCICA, Crypto Express2, Software	Crypto Express2 (Accelerator/Coprocessor), Software
CSNDPKE – Public Key Encrypt	PCICC, CCF	PCICA, Crypto Express2, Software	Crypto Express2 (Accelerator/Coprocessor), Software
CSNDDSV – Digital Signature Verify	PCICC, CCF	PCICA, Crypto Express2, PCIXCC,	Crypto Express2 (Accelerator/Coprocessor), Software
Record Layer			
DES/TDES	CCF	CPACF or Software	CPACF or Software
AES	Software	Software	Software or CPACF with z/OS V1R8
RC2 or RC4	Software	Software	Software
SHA-1 (Hash)	Software	CPACF	CPACF
MD5 (Hash)	Software	Software	Software

Could be lots of different places

Hardware Decisions

- **Some IBM product code that can take advantage of cryptographic hardware includes a software crypto engine. If the hardware is not properly installed and setup the software will be used to perform the encryption.**
  - code is written to detect whether there is crypto hardware and if ICSF is active
    - check is done at session startup time
    - when base hardware crypto and ICSF conditions are met, an indicator is set
      - examples of products; IBM WebSphere, System SSL, TN3270
- **Products that *optionally* allow cryptographic functions usually do not provide a software crypto engine and require the presence of active IBM base crypto and ICSF**
  - example: VTAM Session Level Encryption

SSL Exploiters

CICS
LDAP
Firewall Technologies
WebSphere
MQ Series
Tivoli Access Manager for Business Integration Host Edition
Policy Director Authorization Services
Secure TN3270
IMS
PKI Services
EIM
Sendmail
Secure FTP
IBM HTTP Server

References

- **SSL, Secure Sockets Layer http://wp.netscape.com/eng/ssl3/draft302.txt**
- **TLS, Transport Layer Security http://www.ietf.org/rfc/rfc2246.txt**
- **Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and CRI Profile (RFC 3279) http://www.faqs.org/rfcs/rfc3279.html**
- **X.509 certificate, certificate revocation list, and certificate extensions http://www.ietf.org/rfc/rfc2469.txt**
- **Signatures**
  - http://www.itl.nist.gov/div897/pubs/fip186.htm (DSS)
  - http://www.rsasecurity.com/rsalabs/pkcs/pkcs-1/index.html (RSA)
- **Hashing**
  - http://www.itl.nist.gov/fipspubs/fip180-1.htm (SHA-1)
  - http://www.ietf.org/rfc/rfc1321.txt?number=1321 (MD5)
- **Key Exchange**
  - http://www.ietf.org/internet-drafts/draft-ietf-pkix-rfc2510bis-08.txt

Questions