WebSphere MQ

MQ and SSL

Neil Kolban
IBM Corp
kolban@us.ibm.com

October 31st 2002

Overview

- Part I – Overview of security goals and SSL
- Part II – The MQ SSL story

Security

- Goals of security
  - Confidentiality
  - Message integrity
  - Endpoint Authentication

Encryption (1)

- Encryption
  - Data confidentiality
  - Plain text vs Cipher text

DAILY
DAILY

Plaintext

Cyphertext

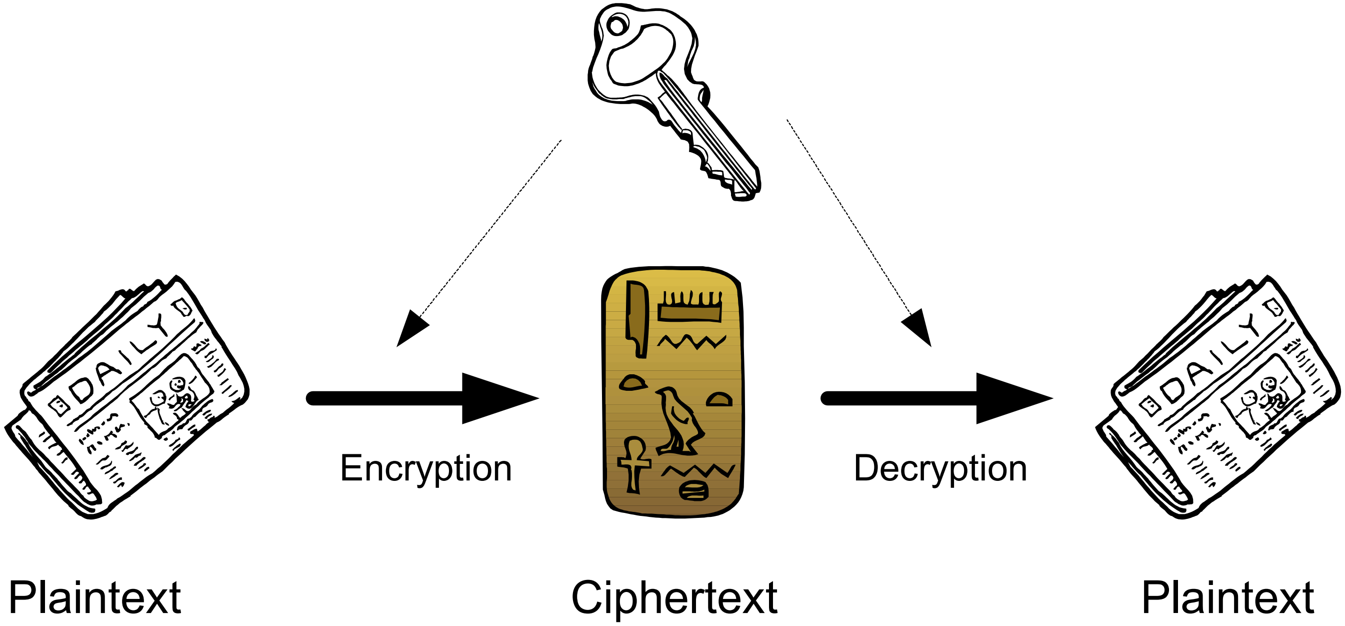
Plaintext

Encryption (2)

- Encryption
  - Data confidentiality
  - Plain text vs Cipher text
- Encryption
  - f_E(Plain) = Cipher
    - Example: f_E(“HEAD”) = “BQTN”
- Decryption
  - f_D(Cipher) = Plain
    - Example: f_D(“BQTN”) = “HEAD”

Plain	Cipher
A	T
B	M
C	I
D	N
E	Q
F	C
G	D
H	B
I	A
…	…
Z	R

Cipher keys (1)

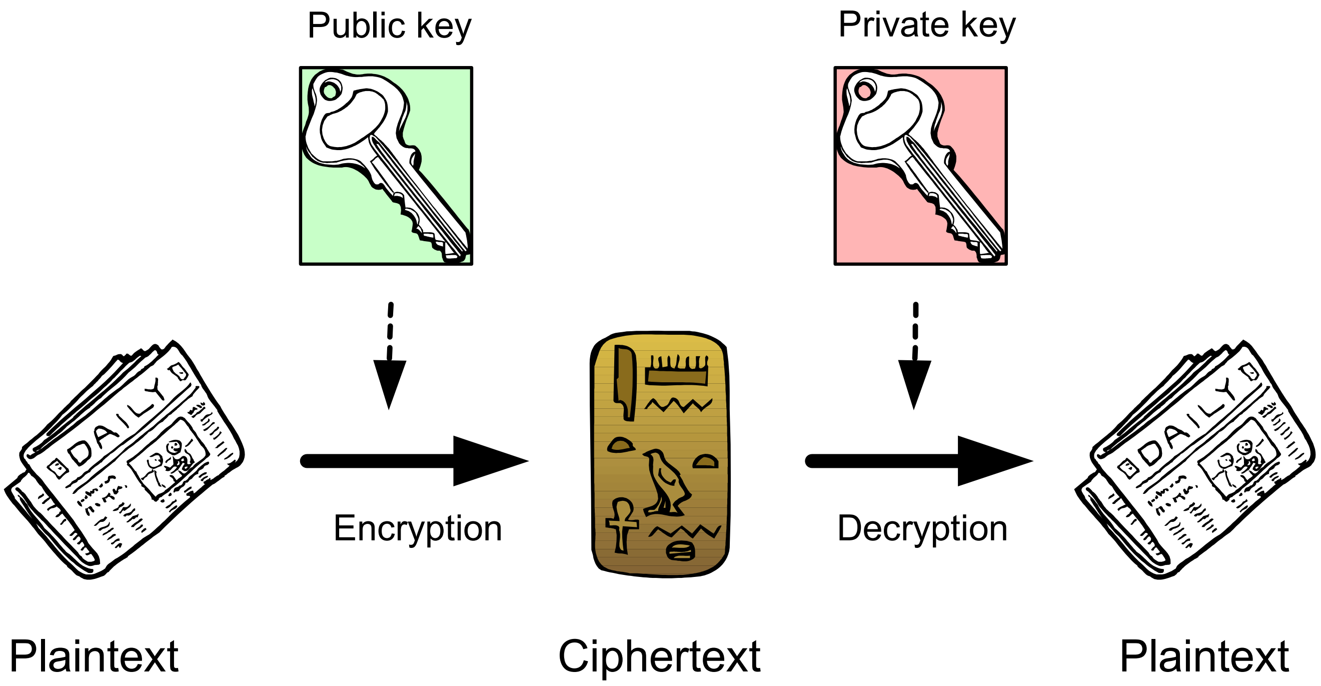


Cipher keys (2)

- Keys
  - –Shared secret key
  - –Symmetric cryptography
  - –Common algorithms
    - –DES
    - –RC2
    - –RC4
- Encryption
  - –f_E(Plain, Key) = Cipher
  - –f_E(“HEAD”, 2) = “LPNC”
- Decryption
  - –f_D(Cipher, Key) = Plain
  - –f_D(“LPNC”, 2) = “HEAD”

Plain	Cipher K=1	Cipher K=2	Cipher K=n
A	T	N	O
B	M	T	W
C	I	Y	E
D	N	C	T
E	Q	P	S
F	C	S	C
G	D	U	I
H	B	L	N
I	A	E	F
…	…	…	…
Z	R	M	H

Public Key Cryptography (1)



Public Key Cryptography (2)

- Two keys
  - One public (known to everyone)
  - One private (known only to you)
  - Common algorithms
    - RSA
    - Diffie-Hellman
  - Asymmetric cryptography
- $f_E(\text{Plain, Key}_{\text{public}}) = \text{Cipher}$
- $f_D(\text{Cipher, Key}_{\text{private}}) = \text{Plain}$
- Keys are asymmetric
- Relatively expensive to use

Security

- Goals of security
  - Confidentiality ☑
  - Message integrity
  - Endpoint Authentication

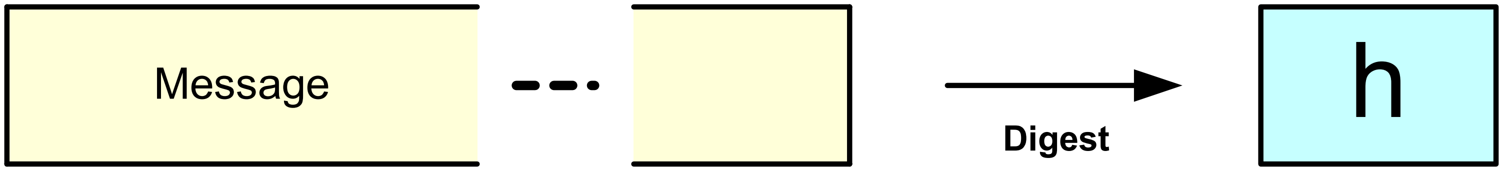
Message Digest (1)

- Input → arbitrary length message
- Output → fixed length string
- Attributes
  - Irreversibility
  - Collision resistance
- Other names for this
  - Hashing
  - Checksum
- Common algorithms
  - MD5
  - SHA

Message Digest (2)

- f_H(Message) = HashData
- f_H(Message1) ≠ f_H(Message2)

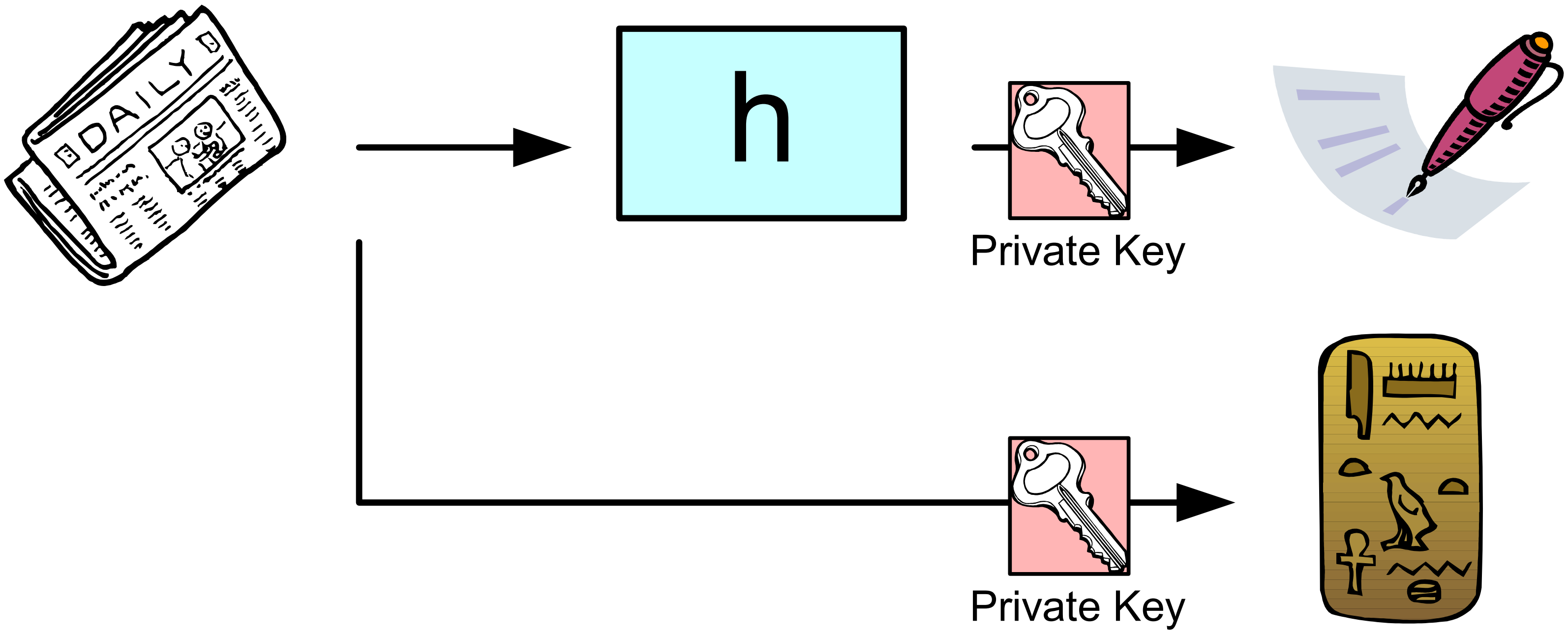
  → Message1 ≠ Message2



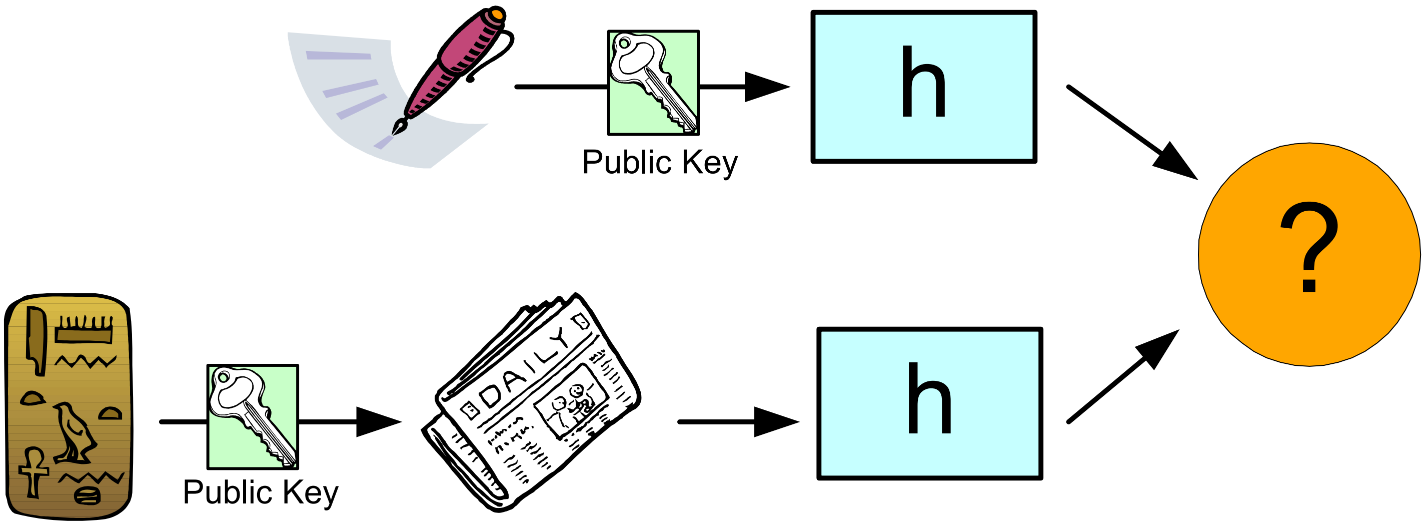
Digital Signature (1)

- Digital Signature built from
  - Message Digest
  - Public key encryption
- Used to prove that a message has not been tampered with.

Digital Signature (2)



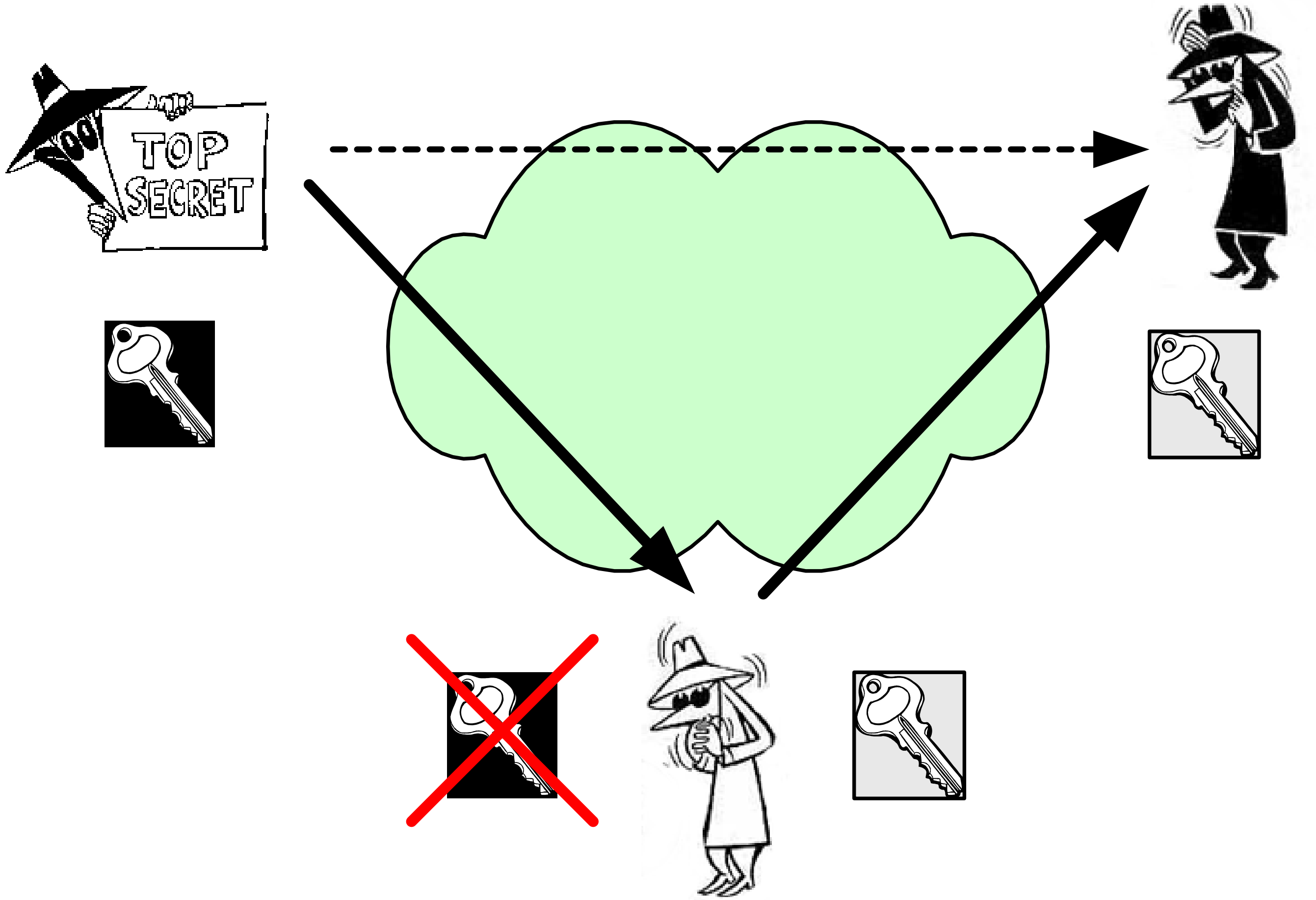
Digital Signature (3)



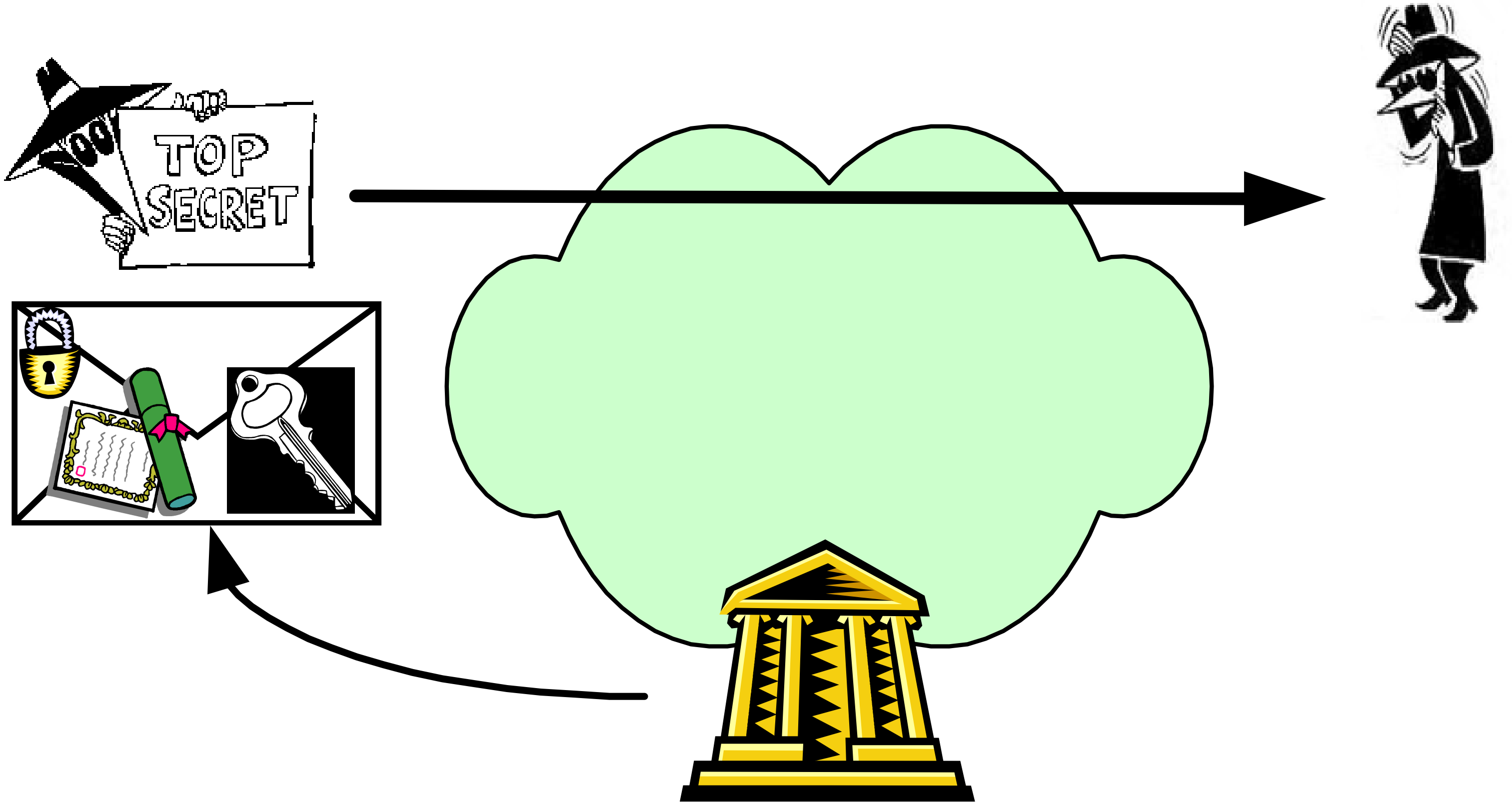
Security

- Goals of security
  - Confidentiality ☑
  - Message integrity ☑
  - Endpoint Authentication ☑

Man in the middle attack



Certificate Authority



Certificates

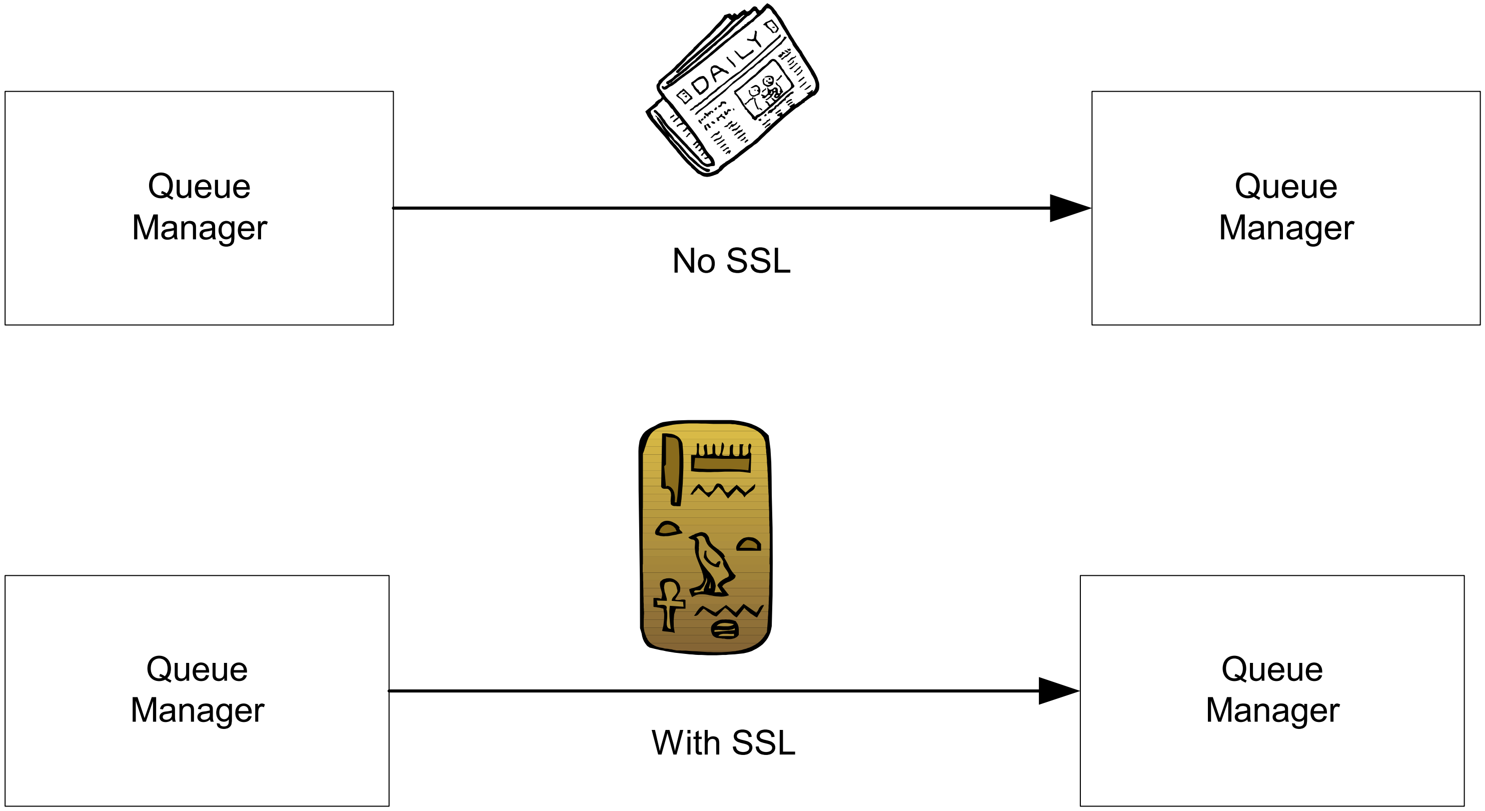
- Issued by CA
  - –VeriSign
  - –Entrust
  - –CyberTrust
  - –etc
- Contains
  - –Subject Name
  - –Issuer Name
  - –X.500 distinguished names
- X.509
  - –Common certificate exchange format

Security

- Goals of security
  - Confidentiality ☑
  - Message integrity ☑
  - Endpoint Authentication ☑
- Implement this design and you have SSL!!

Part II MQ and SSL

Data movement between queue managers



Queue Manager

No SSL

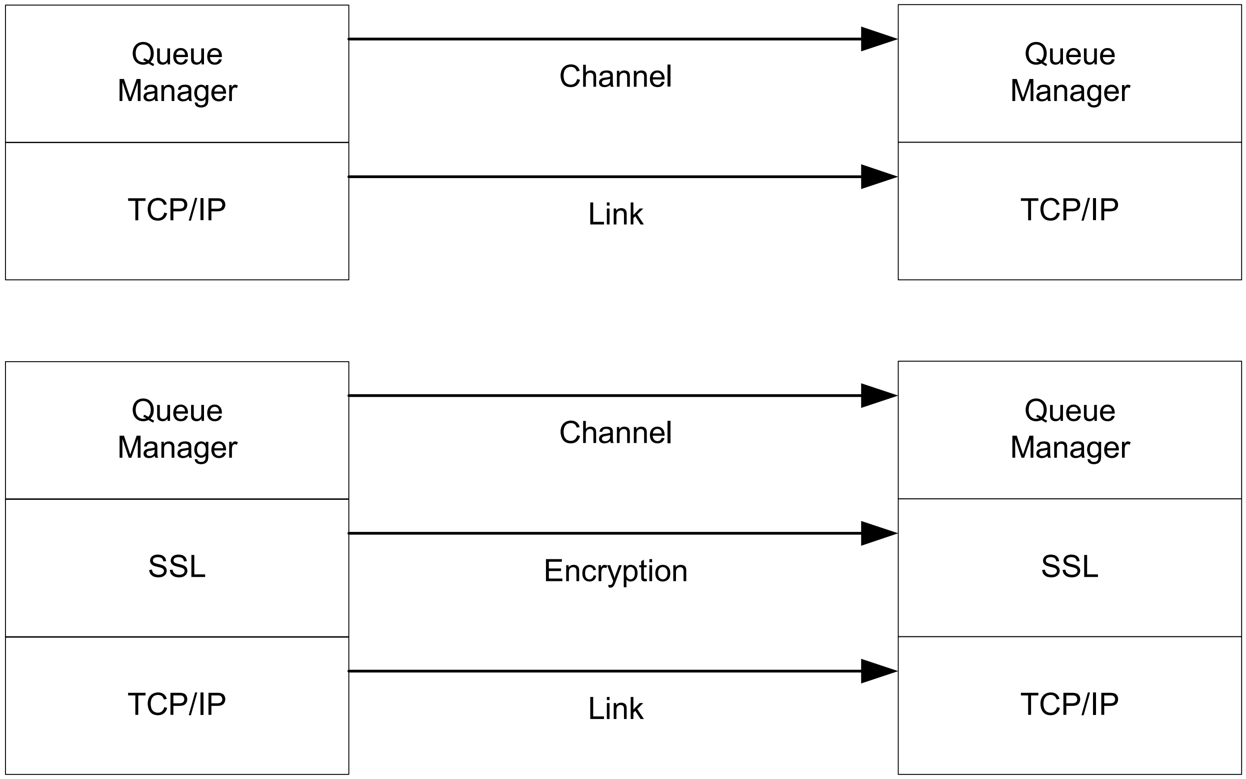
Queue Manager

Queue Manager

With SSL

Queue Manager

Adding SSL Support



MQ SSL Implementations

- Supports SSL V3.0
- Implemented using:

Java	JSSE (Java Secure Socket Extension)
Windows	SChannel
Unix	???
z/OS	System SSL

Channel Security

- SSL can be used across channels
- All kinds of channels supported
  - Sender
  - Receiver
  - Cluster
  - Client
  - Etc
- Specified on a per channel basis

Key questions

- Which CipherSpec shall be used?
  - Cost of security
  - Performance characteristics
- Is client authentication required?
  - Uni or bidirectional authentication
- Names of accepted peers.
  - Limit the names of channel initiators (SSL clients)

Channel definitions

- SSL either enabled or disabled by channel definition
- New parameters for channel definitions
  - Cypher spec (SSLCIPH)
  - DN’s allowed (SSLPEER)
  - Client authentication required (SSLCAUTH)

SSLCipherSpec (SSLCIPH) – Channel attribute

- Name of the Cipher specification to use
- If blank, no SSL
- Same attribute value required on both ends of the channel

CipherSpec name	**Hash algorithm**	**Encryption algorithm**	**Encryption bits**
NULL_MD5	MD5	None	0
NULL_SHA	SHA	None	0
RC4_MD5_EXPORT	MD5	RC4	0
RC4_MD5_US	MD5	RC4	40
RC4_SHA_US	SHA	RC4	128
RC2_MD5_EXPORT	MD5	RC2	128
DES_SHA_EXPORT	SHA	DES	40
RC4_56_SHA_EXPORT1024	SHA	RC4	56
DES_SHA_EXPORT1024	SHA	DES	56
TRIPLE_DES_SHA_US	SHA	3DES	128
TLS_RSA_WITH_AES_128_CBC_SHA	SHA	AES	128
TLS_RSA_WITH_AES_128_CBC_SHA	SHA	AES	256

SSLClientAuth (SSLCAUTH) - Channel attribute

- Requestor to form channel considered the SSL Client
- Defines if certificate from client is needed to form channel
- Values:
  - Required – Client authentication required
  - Optional – Client authentication optional

SSLPeerName (SSLPEER) - Channel attribute

- Distinguished names of the allowed partners

Obtaining certificates

- Certificates obtained from Commercial CA
- Certificates for test environments
  - OpenSSL
  - MakeCert
  - Java 1.4 Keytool
  - IKeyMan

Certificate Stores

- Certificates stored in *key repositories*
- Queue manager SSLKeyRepository (SSLKEYR) attributes specifies Queue Manager's location of its own certificate
- MQ Client uses the MQSSLKEYR environment variable to specify location of certificate store

The amqmcert command

- Used to manage MQSeries certificate store
- Adds certificates to store
- Removes certificates from store
- Lists certificates in store
- Assigns certificate to queue manager

Performance

- Nothing for nothing …
- Extra CPU overhead for encrypted data
- No *official* IBM numbers yet published
- Performance expected to be equivalent to moving same quantity of data over base SSL implementation
  - Possibly better due to single handshake and reuse
  - Overhead based on ciphersuite employed

References

- MQ Security Manual
- SSL and TLS – Eric Rescorta
- Java Secure Socket Extension (JSSE) Reference Guide
- Web sites

  http://home.netscape.com/eng/ssl3/ssl-toc.html

IBM

WebSphere MQ

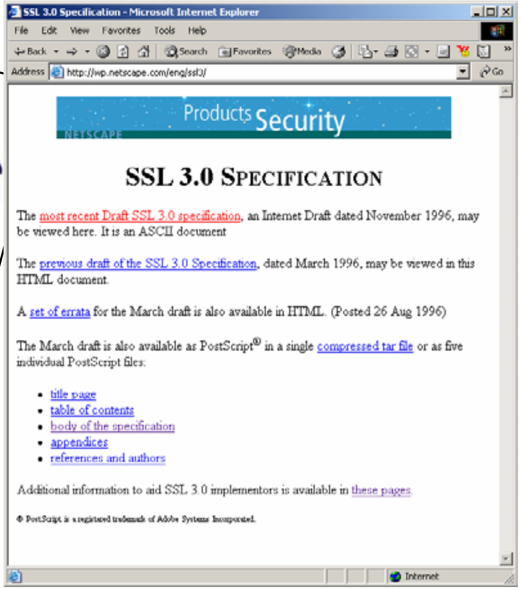
Security

Version 53

SSL and TLS

Designing and Building Secure Systems

Eric Rescorla



SSL 3.0 Specification - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://wp.netscape.com/eng/ssl3/
Go
Products Security
NETSCAPE
Internet

SSL 3.0 SPECIFICATION

The most recent Draft SSL 3.0 specification, an Internet Draft dated November 1996, may be viewed here. It is an ASCII document

The previous draft of the SSL 3.0 Specification, dated March 1996, may be viewed in this HTML document.

A set of errata for the March draft is also available in HTML. (Posted 26 Aug 1996)

The March draft is also available as PostScript® in a single compressed tar file or as five individual PostScript files:

- title page
- table of contents
- body of the specification
- appendices
- references and authors

Additional information to aid SSL 3.0 implementors is available in these pages.

® PostScript is a registered trademark of Adobe Systems Incorporated.