Security in Computer Networks

Raj Jain
Washington University in Saint Louis
Saint Louis, MO 63130
Jain@wustl.edu

Audio/Video recordings of this lecture are available on-line at:

http://www.cse.wustl.edu/~jain/cse473-20/

1. Secret Key Encryption
2. Public Key Encryption
3. Hash Functions, Digital Signature, Digital Certificates
4. Secure Email

Not Covered:, SSL, IKE, WEP, IPSec, VPN, Firewalls, Intrusion Detection. These topics will not be included in the exam.

Note: This class lecture is based on Chapter 8 of the textbook (Kurose and Ross) and the figures provided by the authors.

Security Requirements

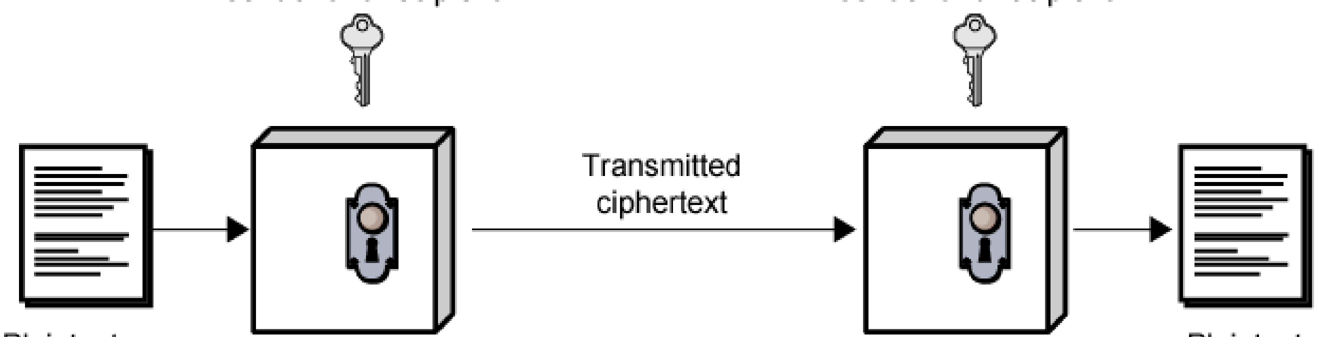
- **Integrity**: Received = sent?
- **Availability**: Legal users should be able to use.
  Ping continuously ⇒ No useful work gets done.
- **Confidentiality and Privacy**:
  No snooping or wiretapping
- **Authentication**: You are who you say you are.
  A student at Dartmouth posing as a professor canceled the exam.
- **Authorization** = Access Control
  Only authorized users get to the data
- **Non-repudiation**: Neither sender nor receiver can deny the existence of a message

Secret Key Encryption: Overview

1. Concept: Secret Key Encryption
2. Method: Block Encryption
3. Improvement: Cipher Block Chaining (CBC)
4. Standards: DES, 3DES, AES

Secret Key Encryption

- Also known as symmetric key encryption
- Encrypted_Message = Encrypt(Key, Message)
- Message = Decrypt(Key, Encrypted_Message)
- Example: Encrypt = division
- 433 = 48 R 1 (using divisor of 9)



Secret Key: A Simple Example

- **Substitution**: Substituting one thing for another
- **Monoalphabetic**: substitute one letter for another

plaintext: abcdefghijklmnopqrstuvwxyz

↓ ↓

ciphertext: mnbvcxzasdfghjklpoiuytrewq

E.g.:

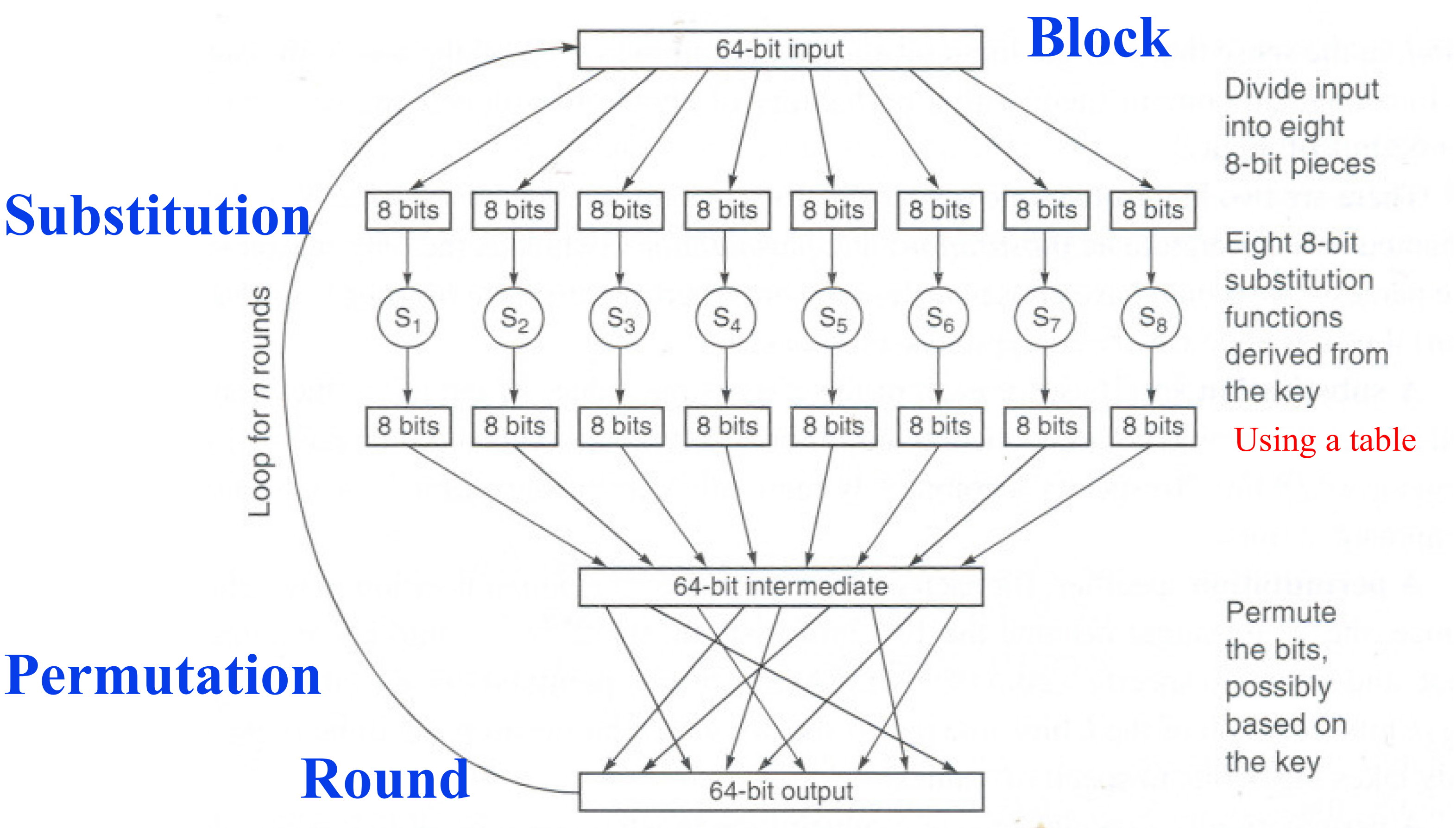
Plaintext: bob. i love you. alice

ciphertext: nkn. s gktc wky. mgsbc

- **Polyalphabetic**: Use multiple substitutions C1, C2, …
  Substitution selected depends upon the position
  ⇒Same letter coded differently in different position

Block Encryption

- Block Encryption

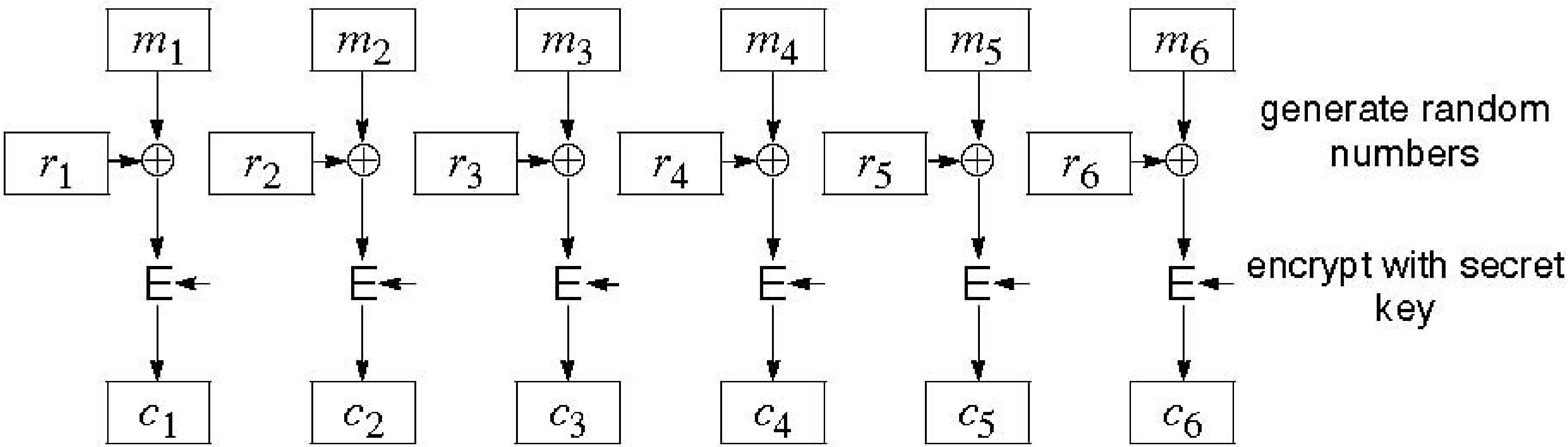


Block Encryption (Cont)

- Short block length ⇒ tabular attack
- 64-bit block
- Transformations:
  - Substitution: replace k-bit input blocks with k-bit output blocks
  - Permutation: move input bits around.
    1 → 13, 2 → 61, etc.
- Round: Substitution round followed by permutation round and so on. Diffusion + Confusion.
  Diffusion ⇒ 1 bit change in input changes many bits in output
  Confusion ⇒ Relationship between input and output is complex

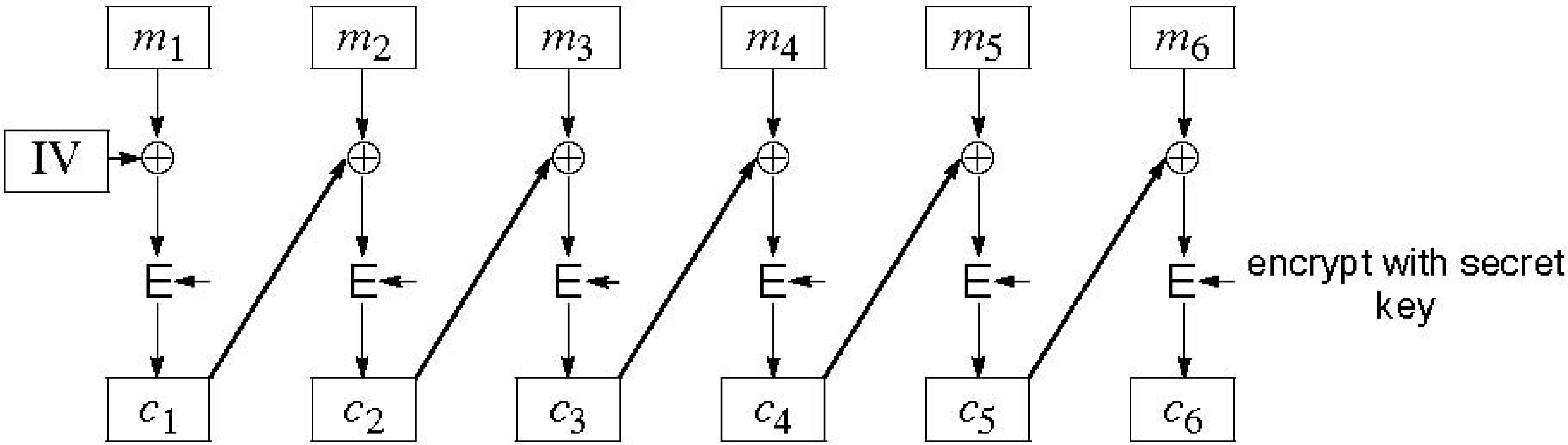
Cipher Block Chaining (CBC)

- Goal: Same message encoded differently
- Add a random number before encoding



CBC (Cont)

- Use C_i as random number for i+1



- Need Initial Value (IV)
- no IV ⇒ Same output for same message
  ⇒ one can guess changed blocks
- Example: Continue Holding, Start Bombing

Data Encryption Standard (DES)

- Published by NIST in 1977
- For commercial and *unclassified* government applications
- 8 octet (64 bit) key.
  Each octet with 1 odd parity bit ⇒ 56-bit key
- Efficient hardware implementation
- Used in most financial transactions
- Computing power goes up 1 bit every 2 years
- 56-bit was secure in 1977 but is not secure today
- Now we use DES three times ⇒ Triple DES = 3DES
  Cipher Text= DES(key1, DES(key2, DES(key1, Plain Text)))

Advanced Encryption Standard (AES)

- Designed in 1997-2001 by National Institute of Standards and Technology (NIST)
- Federal information processing standard (FIPS 197)
- Symmetric block cipher, Block length 128 bits
- Key lengths 128, 192, and 256 bits.
  Full key is used. No parity bit in the byte.
  Memory may use 9-bits to store a byte.

Secret Key Encryption: Review

1. Secret key encryption requires a shared secret key
2. Block encryption, e.g., DES, 3DES, AES break into fixed size blocks and encrypt
3. CBC is one of many modes are used to ensure that the same plain text results in different cipher text.

Ref: Section 8.1, 8.2.1, Review Questions R1, R2, R6, Problems P1-P6

Homework 8A

- [6 points] Consider 3-bit block cipher in the Table below

Plain	000	001	010	011	100	101	110	111
Cipher	110	111	101	100	011	010	000	001

- Suppose the plaintext is 100100100.

(a) Initially assume that CBC is not used. What is the resulting ciphertext?

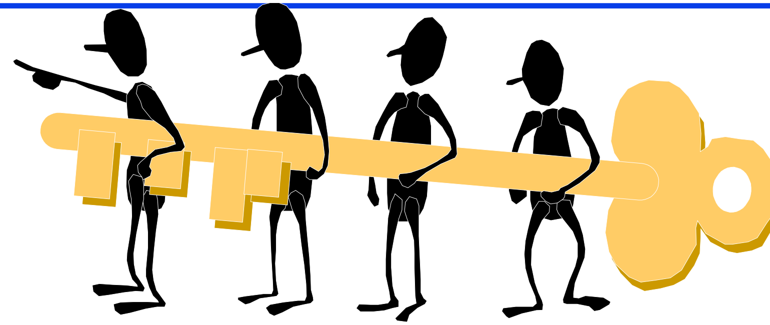
(b) Suppose Trudy sniffs the cipher text. Assuming she knows that a 3-bit block cipher without CBC is being employed (but doesn't know the specific cipher), what can she surmise?

(c) Now suppose that CBC is used with IV-111. What is the resulting ciphertext?

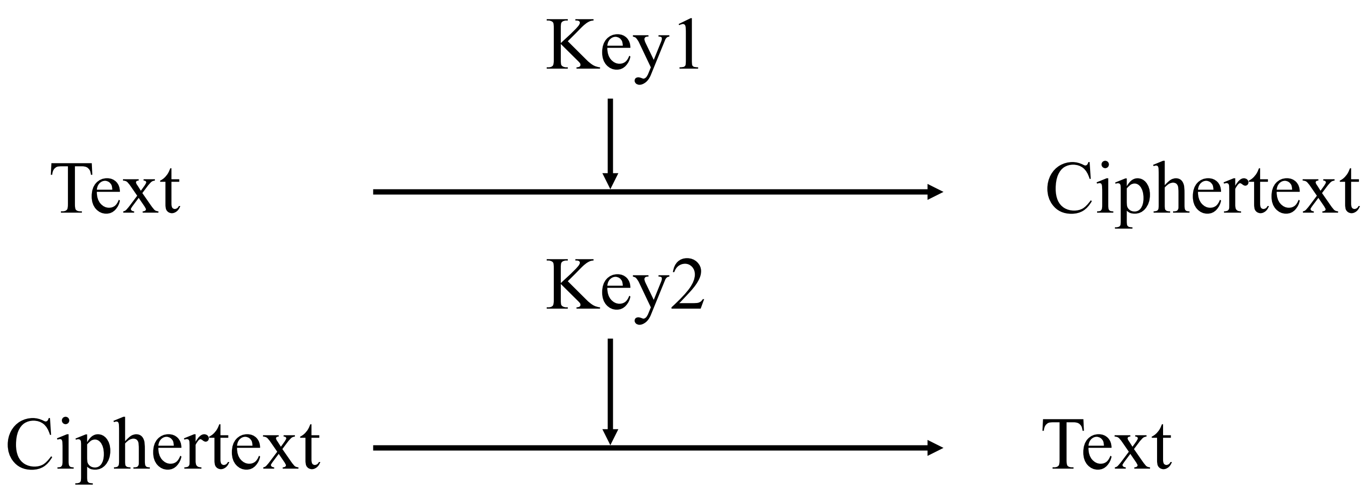
Public Key Encryption

1. Public Key Encryption
2. Modular Arithmetic
3. RSA Public Key Encryption

Public Key Encryption

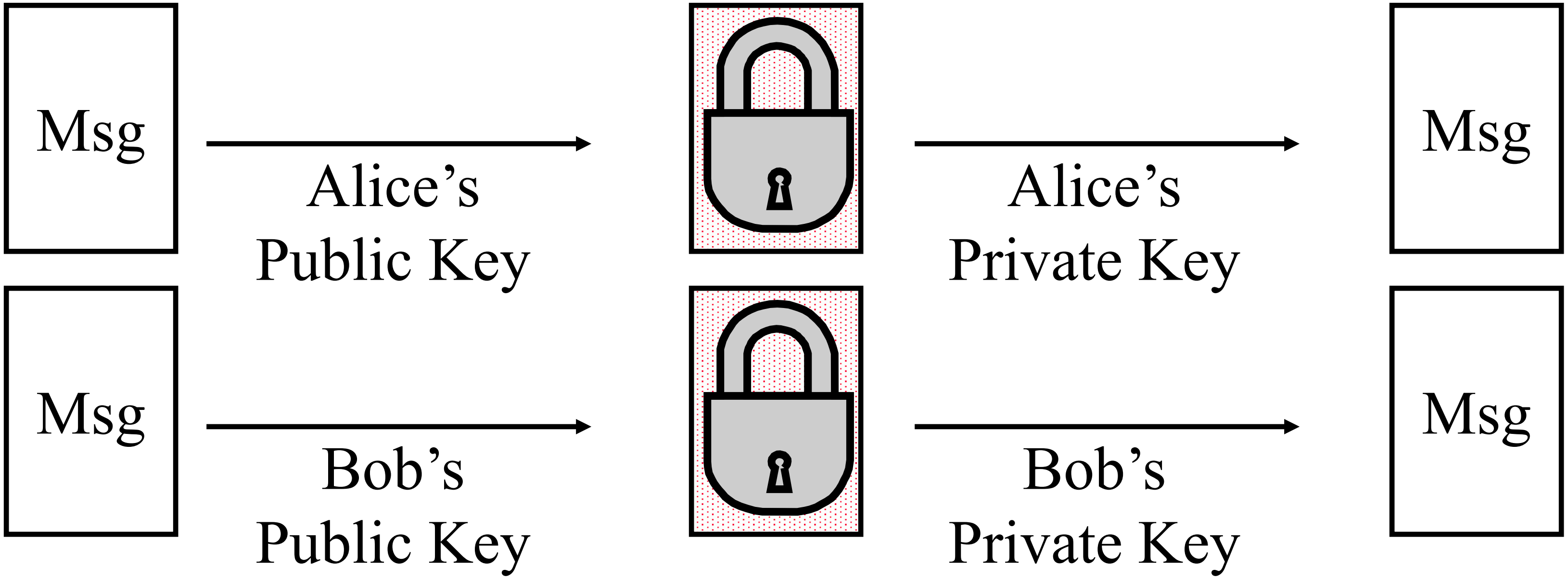


- Invented in 1975 by Diffie and Hellman
- Encrypted_Message = Encrypt(Key1, Message)
- Message = Decrypt(Key2, Encrypted_Message)



Public Key (Cont)

- One key is private and the other is public
- Message = Decrypt(Public_Key, Encrypt(Private_Key, Message))
- Message = Decrypt(Private_Key, Encrypt(Public_Key, Message))
- Encrypted with public key can be decrypted by private key
  Encrypted with private key can be decrypted by public key



Public Key Encryption Method

- Rivest, Shamir, and Adelson (RSA) method
- Example: Key1 = <3,187>, Key2 = <107,187>
- $Encrypted\_Message = m^3 \bmod 187$
- $Message = Encrypted\_Message^{107} \bmod 187$
- Message = 5
- Encrypted Message = $5^3 = 125 \bmod 187 = 125$
- Message = $125^{107} \bmod 187 = 5$
  $= 125^{(64+32+8+2+1)} \bmod 187$
  $= \{(125^{64} \bmod 187)(125^{32} \bmod 187)\ldots$
  $(125^2 \bmod 187)(125 \bmod 187)\} \bmod 187$

Modular Arithmetic

- $xy \bmod m = (x \bmod m)\,(y \bmod m) \bmod m$
- $x^4 \bmod m = (x^2 \bmod m)(x^2 \bmod m) \bmod m$
- $x^{ij} \bmod m = (x^i \bmod m)^j \bmod m$
- $125 \bmod 187 = 125$
- $125^2 \bmod 187 = 15625 \bmod 187 = 104$
- $125^4 \bmod 187 = (125^2 \bmod 187)^2 \bmod 187$
  $= 104^2 \bmod 187 = 10816 \bmod 187 = 157$
- $125^8 \bmod 187 = 157^2 \bmod 187 = 152$
- $125^{16} \bmod 187 = 152^2 \bmod 187 = 103$
- $125^{32} \bmod 187 = 103^2 \bmod 187 = 137$
- $125^{64} \bmod 187 = 137^2 \bmod 187 = 69$
- $125^{107} = 125^{64+32+8+2+1} \bmod 187$
  $= 69 \times 137 \times 152 \times 104 \times 125 \bmod 187$
  $= 18679128000 \bmod 187 = 5$
- Need to be able to do additions to convert 107 to 64+32+8+2+1

Notation:
x = y mod z
or
x = y (mod z)
or
x mod z = y

RSA Public Key Encryption

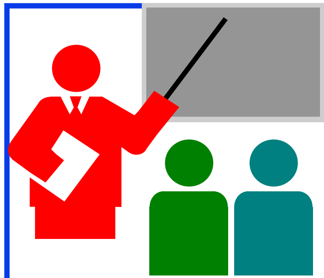
- Ron Rivest, Adi Shamir, and Len Adleman at MIT 1978
- Both plain text M and cipher text C are integers between 0 and n-1.
- Key 1 = {e, n},
  Key 2 = {d, n}
- $C = M^e \bmod n$
  $M = C^d \bmod n$
- How to construct keys:
  - Select two large primes: p, q, $p \neq q$
  - $n = p \times q$
  - Calculate $z = (p-1)(q-1)$
  - Select e, such that $\gcd(z, e) = 1$; $0 < e < z$
  - Calculate d such that $de \bmod z = 1$

RSA Algorithm: Example

- Select two large primes: p, q, $p \neq q$
  $p = 17, q = 11$
- $n = p \times q = 17 \times 11 = 187$
- Calculate $z = (p-1)(q-1) = 16 x 10 = 160$
- Select e, such that $\gcd(z, e) = 1; 0 < e < z$
  say, $e = 7$
- Calculate d such that de mod z = 1
  - $160k+1 = 161, 321, 481, 641$
  - Check which of these is divisible by 7
  - 161 is divisible by 7 giving $d = 161/7 = 23$
- Key 1 = {7, 187}, Key 2 = {23, 187}

Confidentiality and Non-Repudiation

- User 1 to User 2:
- Encrypted_Message
  = Encrypt(Public_Key2,
  Encrypt(Private_Key1, Message))
- Message = Decrypt(Public_Key1,
  Decrypt(Private_Key2, Encrypted_Message)
  ⇒ Authentic and Private



Public Key Encryption: Review

1. Public Key Encryption uses two keys: Public and Private
2. Either key can be used to encrypt. Other key will decrypt.
3. RSA public key method is based on difficulty of factorization

Ref: Section 8.2.2, Review exercises:R3, R7, Problems: P7, P9, P10

Homework 8B

[6 points] Consider RSA with p=5, q=11

A. what are n and z

B. let e be 3. Why is this an acceptable choice for e?

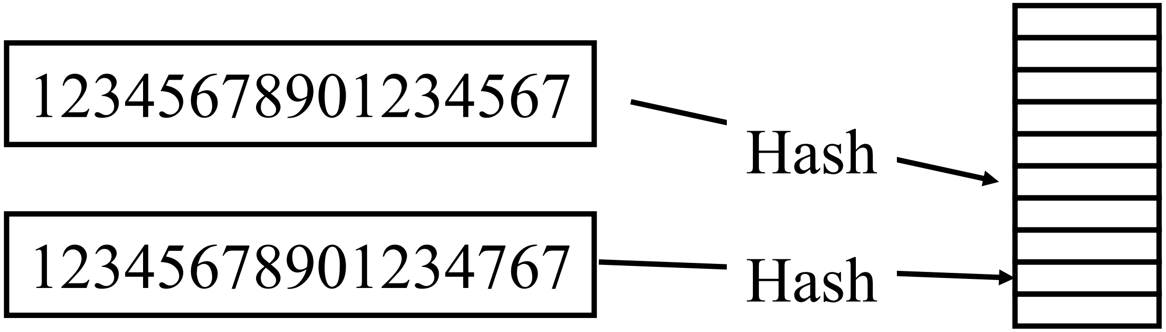
C. Find d such that de=1(mod z) and d<160

D. Encrypt the message m=8 using the key (n,e). Let c be the corresponding cipher text. Show all work including decryption.

Hash, Signatures, Certificates

1. Hash Functions
2. MD5 Hash
3. SHA-1 Algorithm
4. Message Authentication Code (MAC)
5. Digital Signature
6. Digital Certificates
7. End Point Authentication

Hash Functions



Example: CRC can be used as a hash (not recommended for security applications)

Requirements:

1. Applicable to any size message
2. Fixed length output
3. Easy to compute
4. Difficult to Invert ⇒ Can't find x given H(x) ⇒ One-way
5. Difficult to find y, such that H(x) = H(y) ⇒ Can't change msg
6. Difficult to find *any* pair (x, y) such that H(x) = H(y) ⇒ Strong hash

MD5 Hash

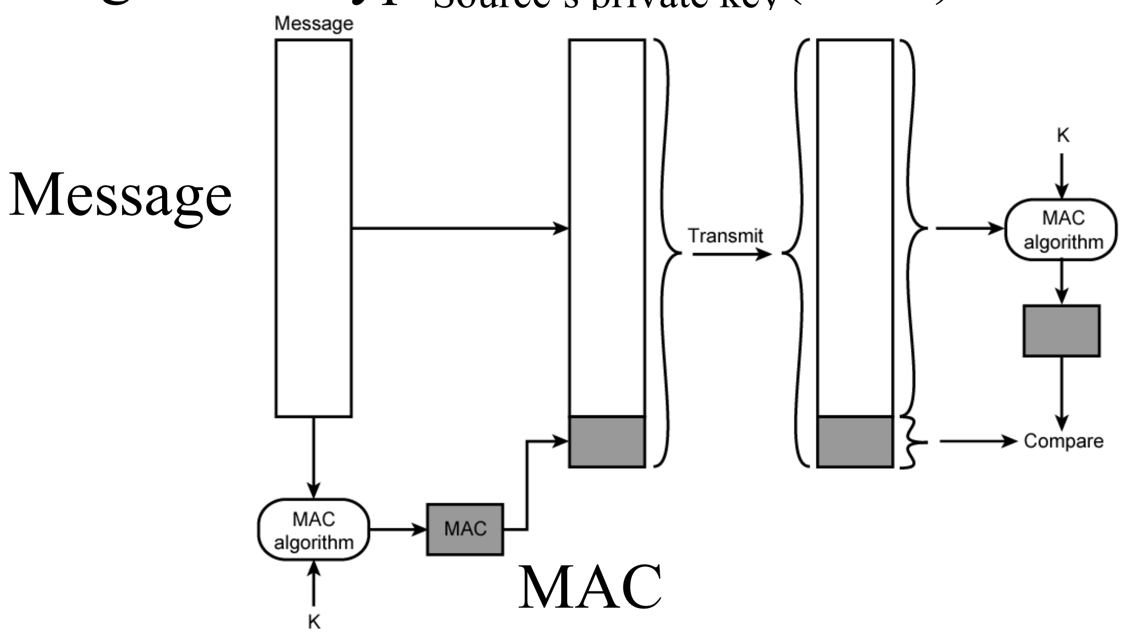
- 128-bit hash using 512 bit blocks using 32-bit operations
- Invented by Ron Rivest in 1991
- Described in RFC 1321
- Commonly used to check the integrity of files (easy to fudge message and the checksum)
- Also used to store passwords

SHA-1 Algorithm

- 160 bit hash using 512 bit blocks and 32 bit operations
- Five passes (4 in MD5 and 3 in MD4)
- Maximum message size is 2^{64} bit

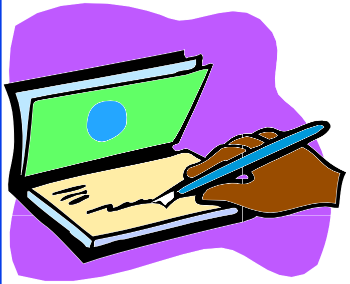
Message Authentication Code (MAC)

- Authentic Message = Contents unchanged + Source Verified
- May also want to ensure that the time of the message is correct
- Encrypt$_{\text{secret key}}$ {Message, CRC, Time Stamp}
- Message + Encrypt$_{\text{secret key}}$(Hash)
  Or, Message + Encrypt$_{\text{Source's private key}}$(Hash)



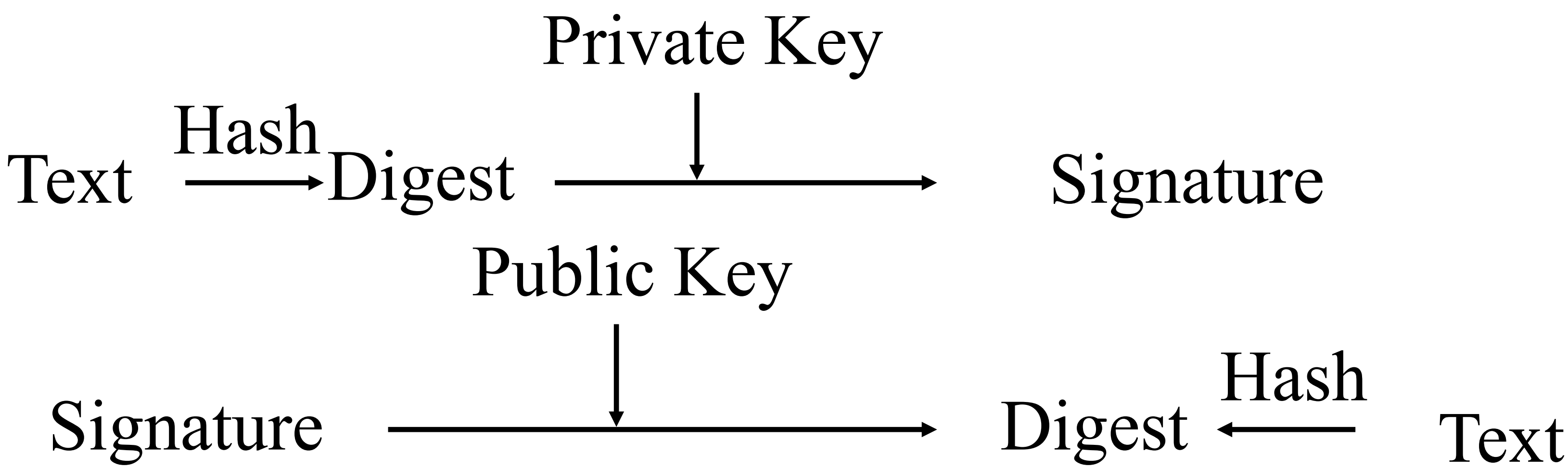
HMAC Overview

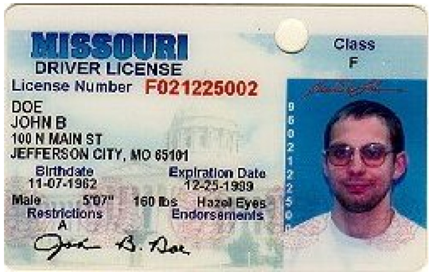
- Keyed Hash $\Rightarrow$ includes a key along with message
- HMAC is a general design. Can use any hash function $\Rightarrow$ HMAC-MD5, HMAC-AES
- Uses hash functions without modifications
- Has well understood cryptographic analysis of authentication mechanism strength



Digital Signature

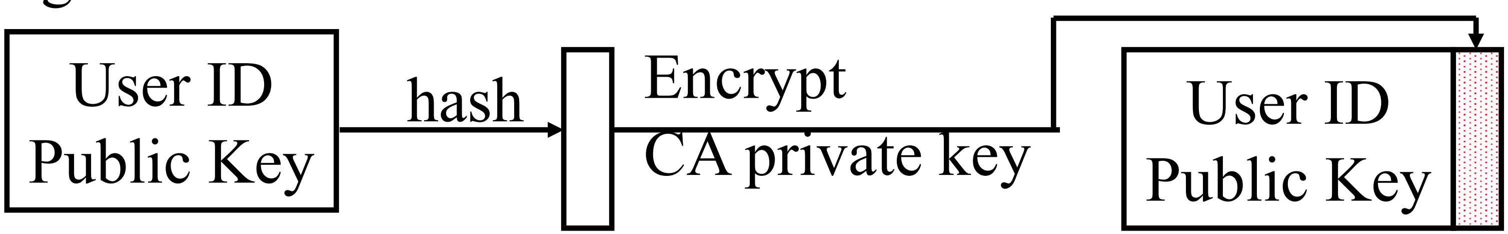
- Message Digest = Hash(Message)
- Signature = Encrypt(Private_Key, Hash)
- Hash(Message) = Decrypt(Public_Key, Signature)
  ⇒ Authentic
- Also known as Message *authentication* code (MAC)



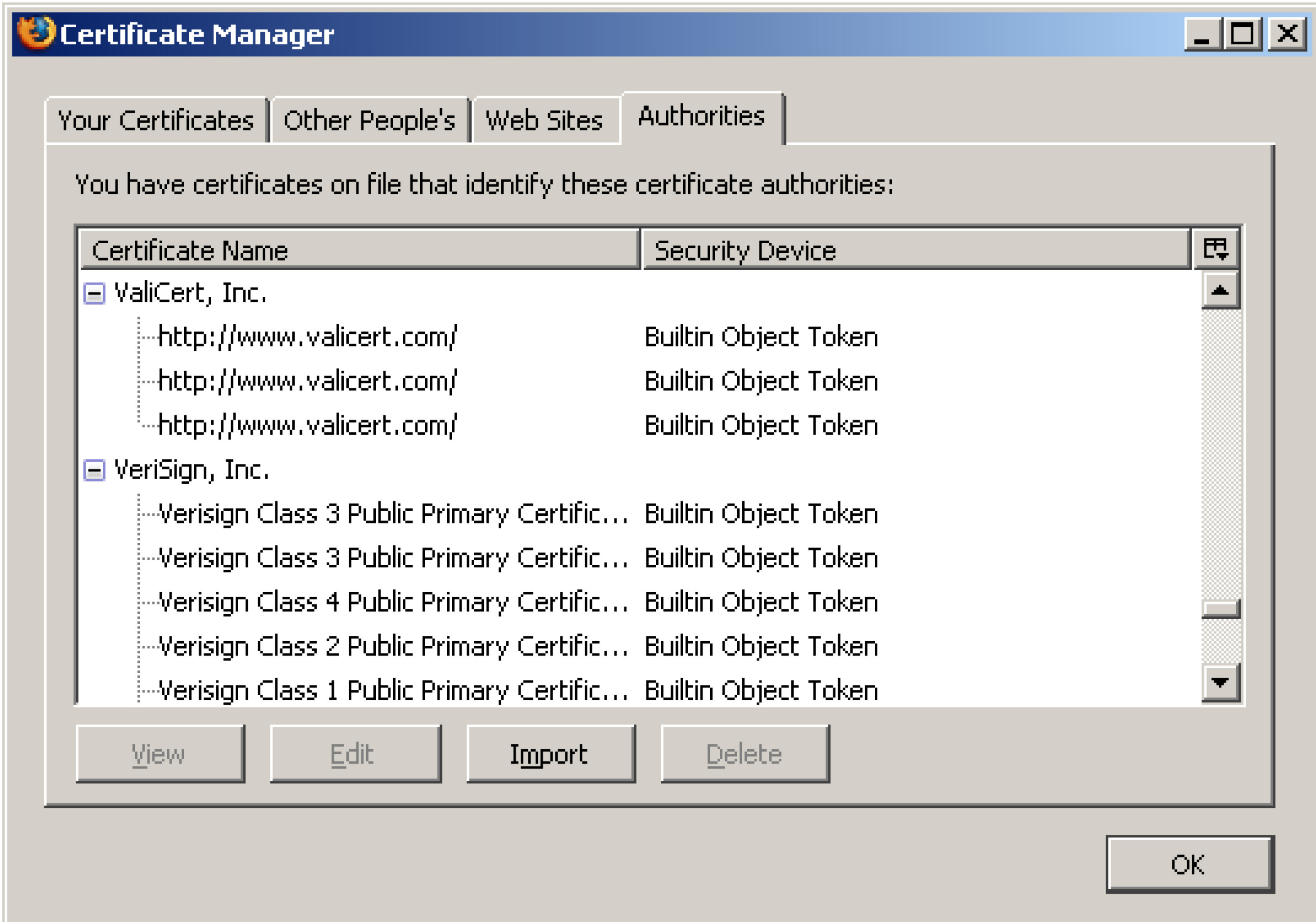


Digital Certificates

- Like driver license or passport
- Digitally signed by Certificate authority (CA) - a trusted organization
- Public keys are distributed with certificates
- CA uses its private key to sign the certificate
  ⇒ Hierarchy of trusted authorities
- X.509 Certificate includes: Name, organization, effective date, expiration date, public key, issuer's CA name, Issuer's CA signature



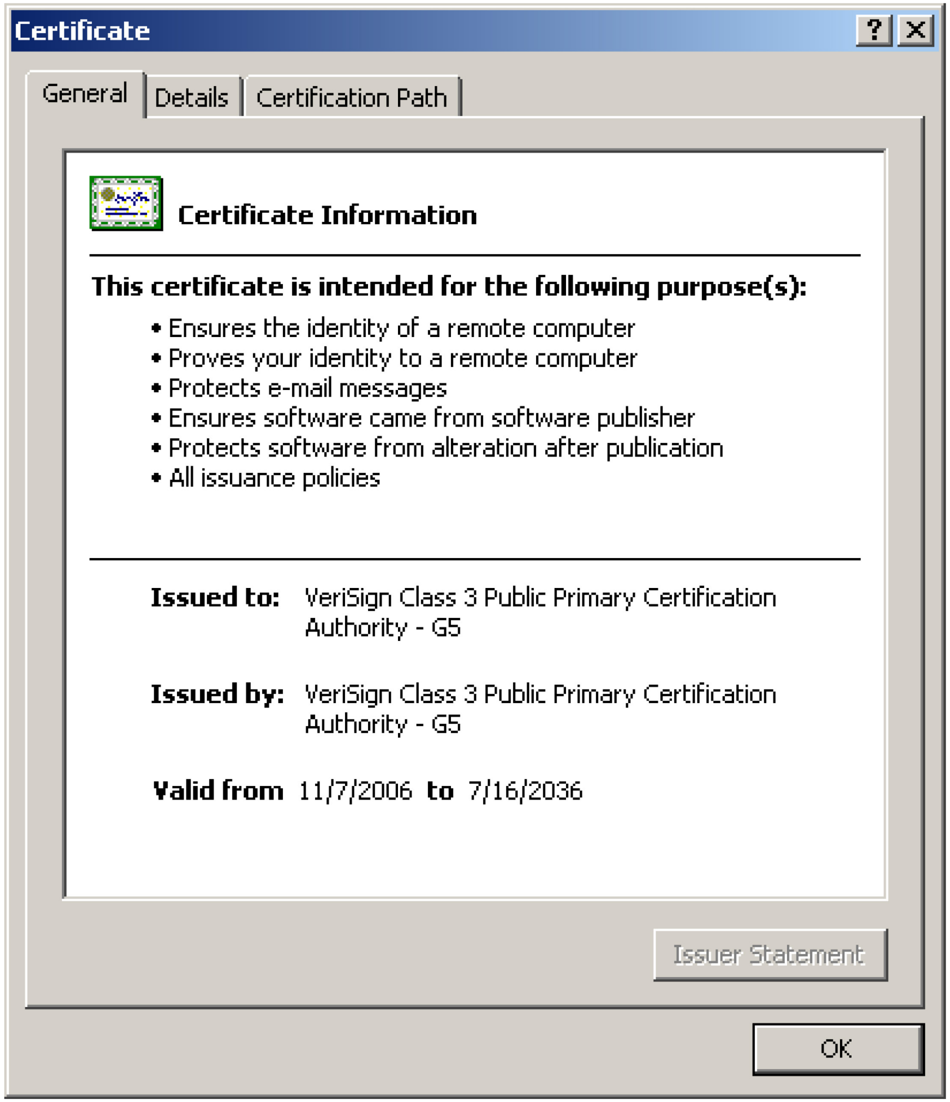
Oligarchy Example



Ref: Windows: http://smallbusiness.chron.com/see-security-certificates-stored-computer-54732.html
MAC: https://superuser.com/questions/992167/where-are-digital-certificates-physically-stored-on-a-mac-os-x-machine

Sample X.509 Certificate

- Certmgr.msc in Windows



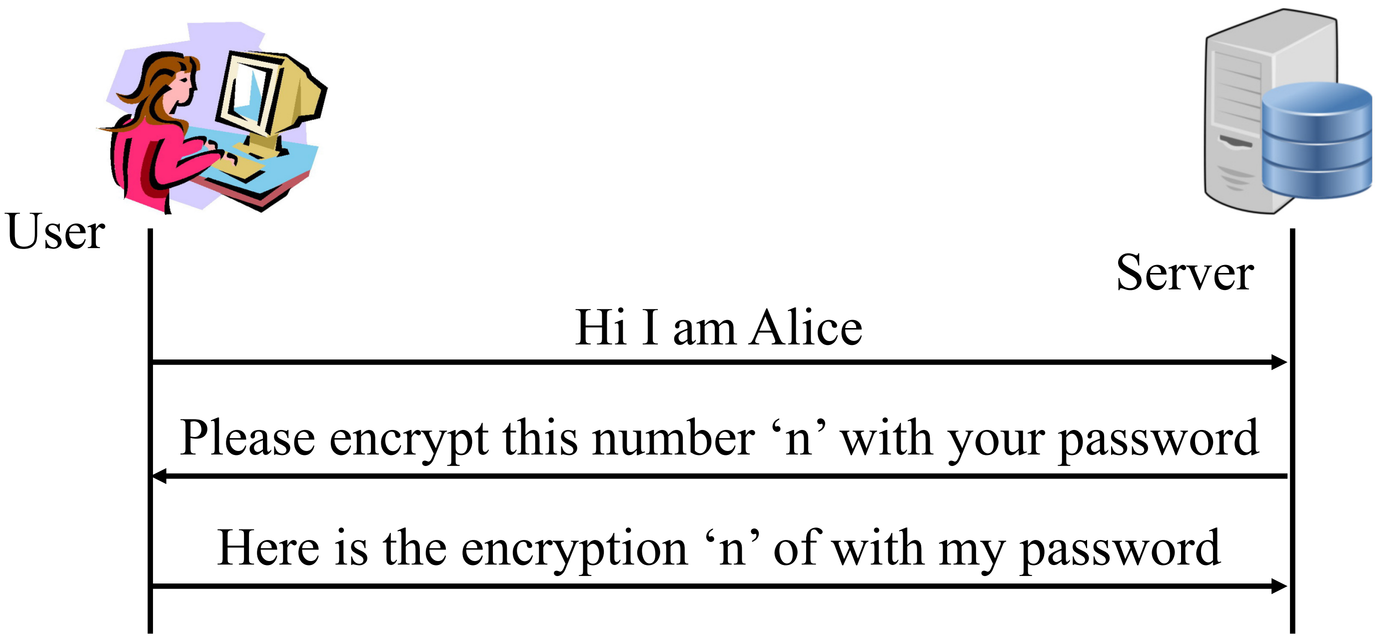
X.509 Sample (Cont)

Field	Value
Version	V3
Serial number	18 da d1 9e 26 7d e8 bb 4a 21…
Signature algorithm	sha1RSA
Issuer	VeriSign Class 3 Public Primary …
Valid from	Tuesday, November 07, 2006 …
Valid to	Wednesday, July 16, 2036 6:…
Subject	VeriSign Class 3 Public Primary …
Public key	RSA (2048 Bits)

Version	V3
Serial number	18 da d1 9e 26 7d e8 bb 4a 21…
Signature algorithm	sha1RSA
Issuer	VeriSign Class 3 Public Primary …
Valid from	Tuesday, November 07, 2006 …
Valid to	Wednesday, July 16, 2036 6:…
Subject	VeriSign Class 3 Public Primary …
Public key	RSA (2048 Bits)

End Point Authentication

- Passwords can not be exchanged in clear
  Nonce = random **n**umber used only **once**
- Also done using certificates



Requires the server to store passwords in clear.

Hashes, Signatures, Certificates

1. Hashes are one-way functions such that it difficult to find another input with the same hash like MD5, SHA-1
2. Message Authentication Code (MAC) ensures message integrity and source authentication using hash functions
3. Digital Signature consists of encrypting the hash of a message using private key
4. Digital certificates are signed by root certification authorities and contain public keys

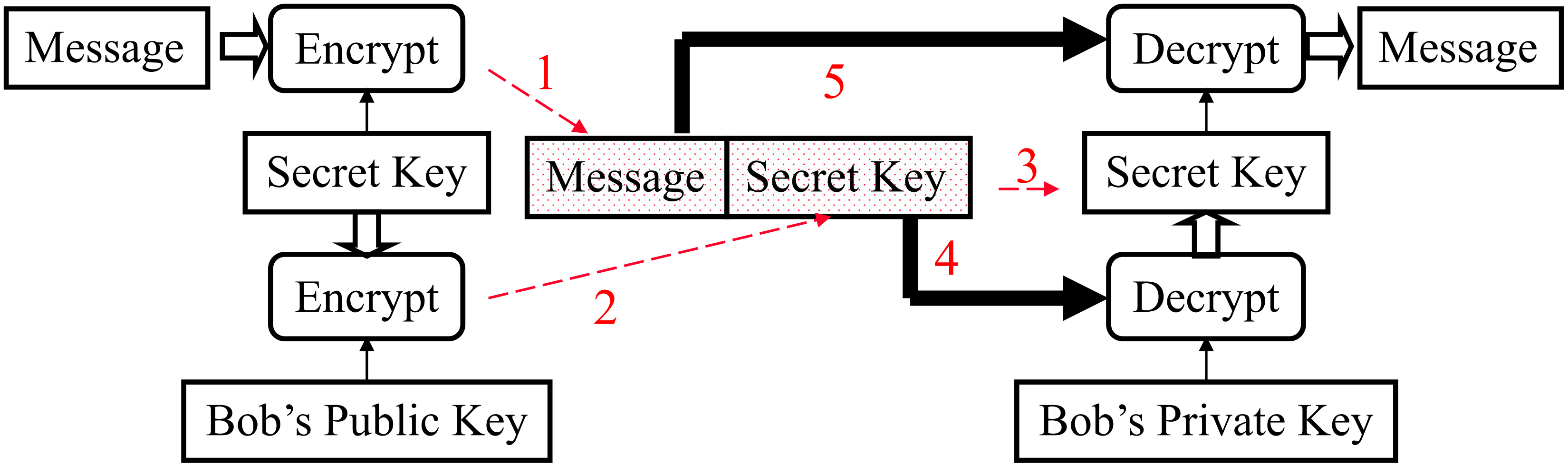
Ref: Section 8.3-8.4, Review questions R9-18

Secure Email

1. Secure E-Mail
2. Signed Secure E-Mail
3. Pretty Good Privacy (PGP)

Secure E-Mail

- Alice wants to send confidential e-mail, m, to Bob.



- **<u>Alice:</u>**

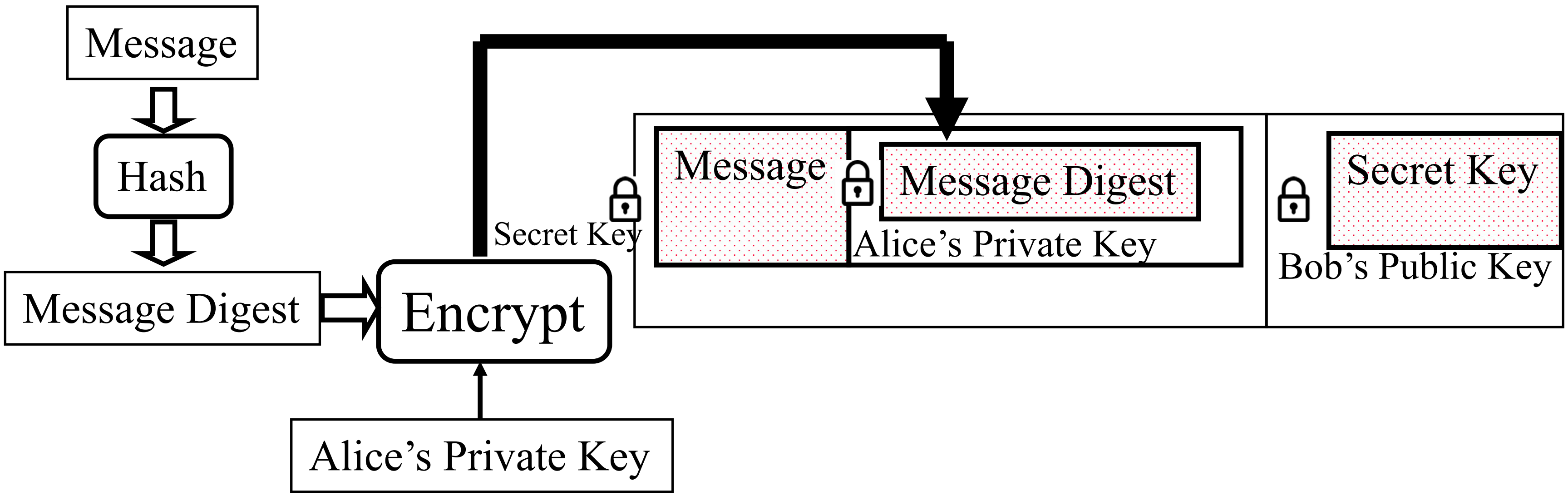
0. Generates random *secret* key, K_S.
1. Encrypts message with K_S (for efficiency)
2. Also encrypts K_S with Bob's public key.
3. Sends both $K_S(m)$ and $K_B(K_S)$ to Bob.

- **<u>Bob:</u>**

4. Bob uses his private key to recover K_s
5. Bob decrypts message

Signed Secure E-Mail

- Alice wants to provide secrecy, sender authentication, message integrity.



- Alice uses three keys: her private key, Bob's public key, newly created secret key
- Bob uses his private key to recover the secret key
- Bob uses Alice's public key to verify that the message came from Alice and was not changed.

Pretty Good Privacy (PGP)

- Used RSA and IDEA (RSA patent in US until 2000)
- V2.6.2 became legal for use within US and can be downloaded from MIT
- A patent-free version using public algorithm has also been developed
- Code published as an OCRable book
- Initially used web of trust- certificates issued by people
- Certificates can be registered on public sites, e.g., MIT
- hushmail.com is an example of PGP mail service
- OpenPGP standard [RFC 4880]
- MIME=Multipurpose Internet Mail Extenstion. Allows non-ascii characters to be encoded in ASCII

Ref: http://en.wikipedia.org/wiki/Pretty_Good_Privacy , https://en.wikipedia.org/wiki/MIME

Lab 8: Secure Email

[20 points] You will receive a "signed" email from the TA. Reply to this email with a "encrypted and signed" email to TA.

If outlook says "*There is a problem with the signature on the TA's message*" then click on the signature icon on the top right of the message and accept Actalis's certificate. The warning will go away.

- You can reply to the TA's email with a signed encrypted message. Content of the the reply should be the contents of the "**Enhanced key usage**" field in your new certificate.
- Before sending the reply, on the outlook message window, Select View → Options → (More Options →) Security Settings
  Select encryption and signature. Now send the message.
- **Outlook is required** for both Windows and Mac

Lab 8 (Cont)

- To sign your email with a private key you need your digital certificate. To send an encrypted email you need TA's public key.
- TA's public key is attached with his/her email.
- The steps to obtain a free certificate and use it for email depend upon your email software and your operating system.
- Instructions for Outlook on Windows 10 are as included next. If you do not have windows, you can do it using remote desktop to a Wash U windows computer.
- Instructions for Mac are similar. Further details for Mac are in the references cited below.

Ref: https://support.apple.com/guide/mail/use-personal-certificates-mlhlp1179/mac
https://knowledge.digicert.com/solution/SO6722.html

Lab 8 (Cont)

1. Getting your Certificate:

- In any browser, go to https://extrassl.actalis.it/portal/uapub/freemail?lang=en
- Enter your wustl.edu email address. Leave everything else blank. and click on "Send Verification Email." Leave the page open.
- Check your email. You will receive a verification code in email within a few minutes. Enter the received verification code on the previous page. Enter Captcha. Accept the first three conditions. You do not need to accept the last two (marketing) conditions. Submit request. It will send the certificate in an email and present a password on the screen. Copy and paste the password in some text file. Also print the page to pdf (as a backup) to save the password.
- You will receive a zip file in email. Unzip it.

Lab 8 (Cont)

2. Installing your Certificate in Outlook:

- Now open the Outlook App (not the website and follow the following click sequence:
- File → Options → Trust Center→ Trust Center Settings → Email Security → Digital IDs import/export
- Import the certificate file and enter the password that was given by Actalis. Click OK.
- Now, you can digitally sign an email by selecting the "Options" tab in the composing a message window, and clicking the "Sign" button.

Ref: https://www.thesslstore.com/knowledgebase/email-signing-support/install-e-mail-signing-certificates-outlook/

Lab 8 (Cont)

4. Sending Encrypted Emails:

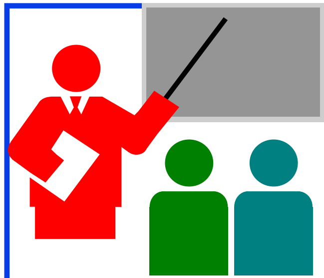
- The recipient may see "There is a problem with the signature" when they receive the signed message for the first time. This is because they may not have included Actalis as a trusted Certificate Authority. To fix this they need to click on the signature icon on the right-top of the message and accept Actalis's certificate. After this the problem message will go away. (In my case, Actalis was already in my system).
- The recipient can also get a Actalis certificate and send a signed message to you. When you open that message, the recipient's public key is automatically installed in your outlook.
- After both of you have each other's public key, you can send encrypted emails to each other. You can send such messages by by selecting the dropdown menu on the "Encrypt" button (right next to the "Sign" button), and selecting "Encrypt with S/MIME".

Lab 8 (Cont)

5. Examining your certificate: From the references below.

- In Windows, use Run → Certmgr.msc
- In the window that opens, look for Personal → Certificates
- Click on the new certificate issued by Actalis. Go to details tab. Scroll down to find "Enhanced Key Usage". Click on it to see the results in the bottom pane. Copy and paste it to your email reply to the TA email.
- Before clicking send, remember to click options and select encryption.
- The process on MAC is in the 2nd reference below but has not been verified.

Ref: https://www.top-password.com/blog/view-installed-certificates-in-windows-10-8-7/
https://www.digicert.com/kb/code-signing/mac-verifying-code-signing-certificate.htm

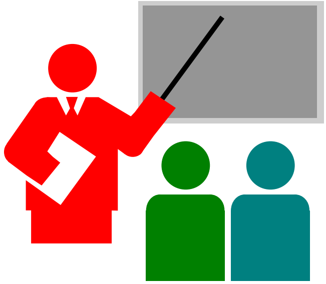


Secure Email: Review

1. Email provide confidentiality using a secret key
2. Public key and Certificates are used to:
    1. Sign the message
    2. To send the secret key

Ref: Section 8.5, Review question R19

Summary



1. Network security requires confidentiality, integrity, availability, authentication, and non-repudiation
2. Encryption can use one secret key or two keys (public and private)
3. Public key is very compute intensive and is generally used to send secret key
4. Digital certificate system is used to certify the public key
5. Secure email uses confidentiality using a secret key, uses certificates and public keys to sign the email and to send the secret key

Ref: Sections 8.1 through 8.5

Acronyms

- 3DES Triple DES
- AES Advanced Encryption Standard
- CA Certificate authority
- CBC Cipher Block Chaining (CBC)
- CRC Cyclic Redundancy Check
- DES Data Encryption Standard (DES)
- FIPS Federal Information Processing standard
- HMAC
- ID Identifier
- IDEA
- IKE Internet Key Exchange
- IPSec Secure IP
- IV Initialization Vector
- MAC Message Authentication Code
- MD4 Message Digest 4
- MD5 Message Digest 5

Acronyms (Cont)

❑	NIST	National Institute of Standards and Technology
❑	OCR	Optical Character Recognition
❑	OpenPGP	Open PGP
❑	PGP	Pretty Good Privacy
❑	RFC	Request for Comment
❑	RSA	Rivest, Shamir, Adleman
❑	SHA	Secure Hash
❑	SSL	Secure Socket Layer
❑	TA	Teaching Assistant
❑	US	United States
❑	VPN	Virtual Private Network
❑	WEP	Wired Eqivalent Privacy
❑	XOR	Exclusive OR

Scan This to Download These Slides

Raj Jain

http://rajjain.com

http://www.cse.wustl.edu/~jain/cse473-20/i_8sec.htm

Related Modules

CSE 567: The Art of Computer Systems Performance Analysis
https://www.youtube.com/playlist?list=PLjGG94etKypJEKjNAa1n_1X0bWWNyZcof

CSE473S: Introduction to Computer Networks (Fall 2011),
https://www.youtube.com/playlist?list=PLjGG94etKypJWOSPMh8Azcgy5e_10TiDw

CSE 570: Recent Advances in Networking (Spring 2013)
https://www.youtube.com/playlist?list=PLjGG94etKypLHyBN8mOgwJLHD2FFIMGq5

CSE571S: Network Security (Spring 2011),
https://www.youtube.com/playlist?list=PLjGG94etKypKvzfVtutHcPFJXumyyg93u

Video Podcasts of Prof. Raj Jain's Lectures,
https://www.youtube.com/channel/UCN4-5wzNP9-ruOzQMs-8NUw