Computer Security

Session 4: Cryptographic Tools: Asymmetric Encryption

Dr. Fuad A. Ghaleb

Department: Computer Science

School of Computing, Faculty of Engineering, Universiti Teknologi Malaysia

2020-2021 (II)

Learning Objectives

After studying this session, you should be able to:

- ❖Explain the basic operation of asymmetric block encryption algorithms.
- ❖Apply the asymmetric block encryption algorithms in a variety of applications.
- ❖ Present an overview of the digital signature mechanism and explain the concept of digital envelopes.
- ❖ Explain the significance of random and pseudorandom numbers in cryptography.

Cryptographic Tools: Asymmetric Encryption

1. Public-Key Encryption
2. Digital Signatures and Key Management
3. Random and Pseudorandom Numbers
4. Practical Application: Encryption of Stored Data

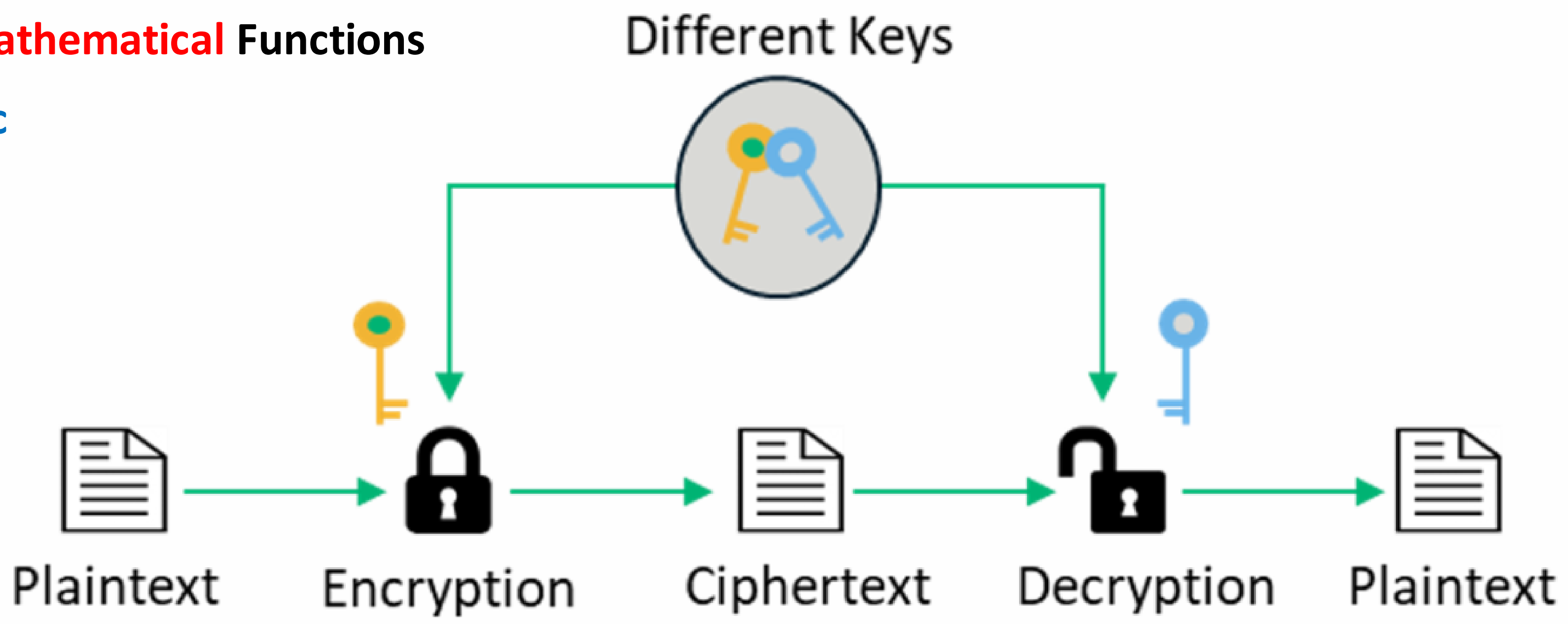
Why do we need asymmetric encryption?

Diffie-Hellman **Key Exchange - 1976**

Public-Key provides **encryption**, **authentication** and **digital signatures**

Based-on Mathematical Functions

Asymmetric



Confidentiality with Asymmetric Encryption

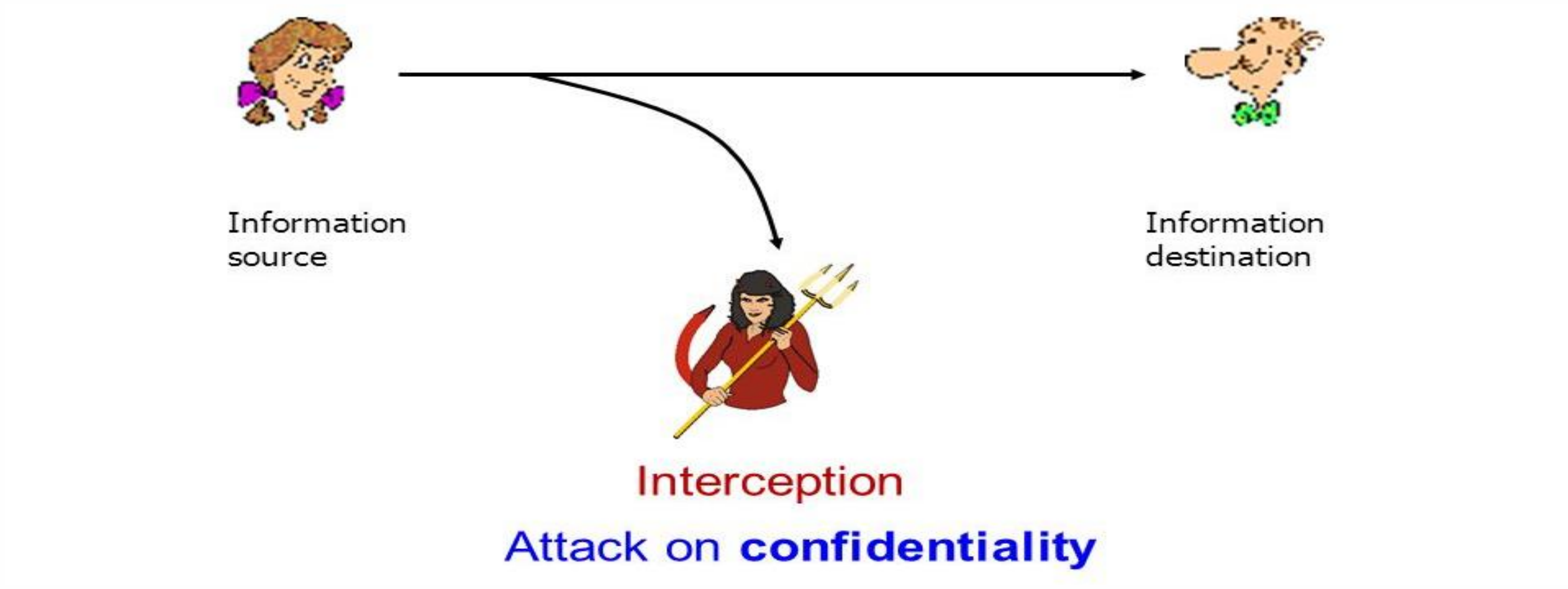
Public-Key Encryption

Symmetric Block Encryption Algorithms

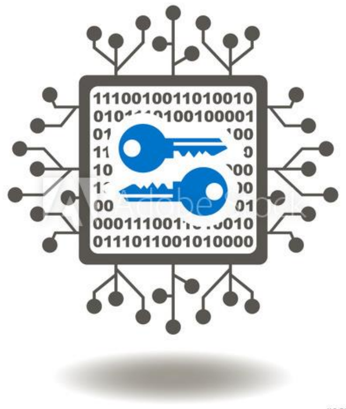
Stream Ciphers

Encryption

- Encryption protects against passive attack (eavesdropping).

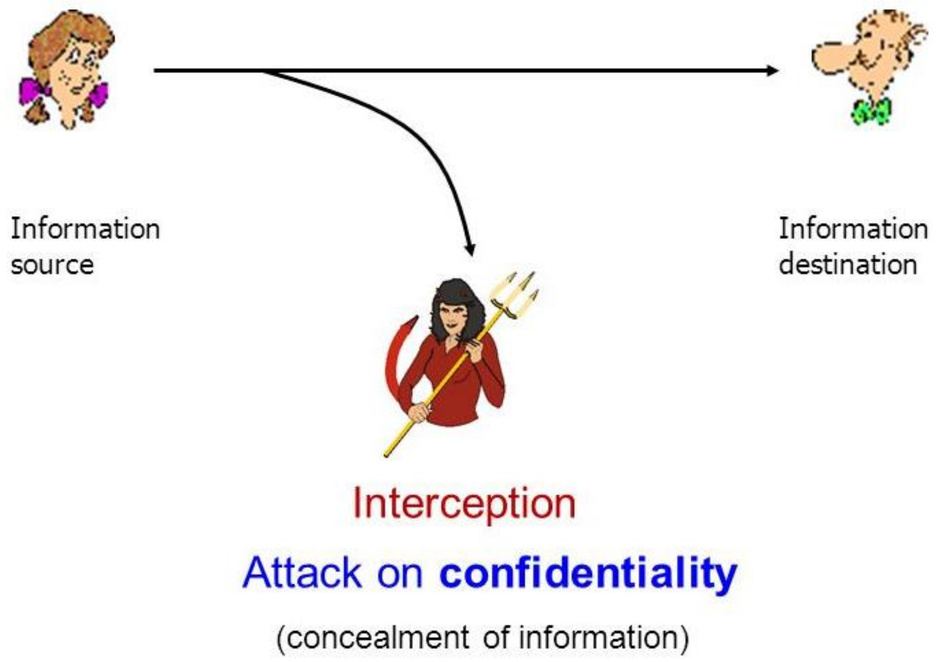


Public-Key Encryption

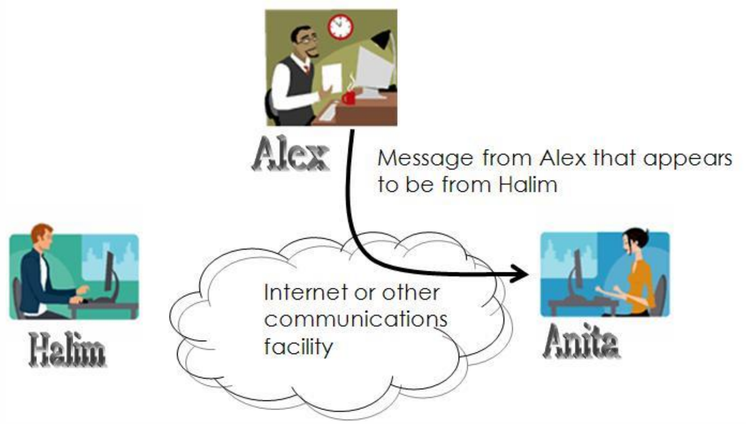


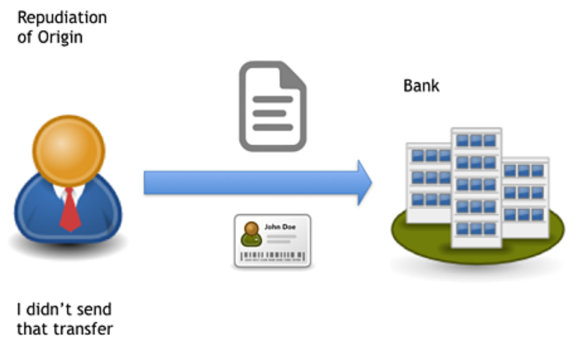
Public-Key Encryption

- Public-Key Encryption Structure
- Applications for Public-Key Cryptosystems
- Requirements for Public-Key Cryptography
- Asymmetric Encryption Algorithms



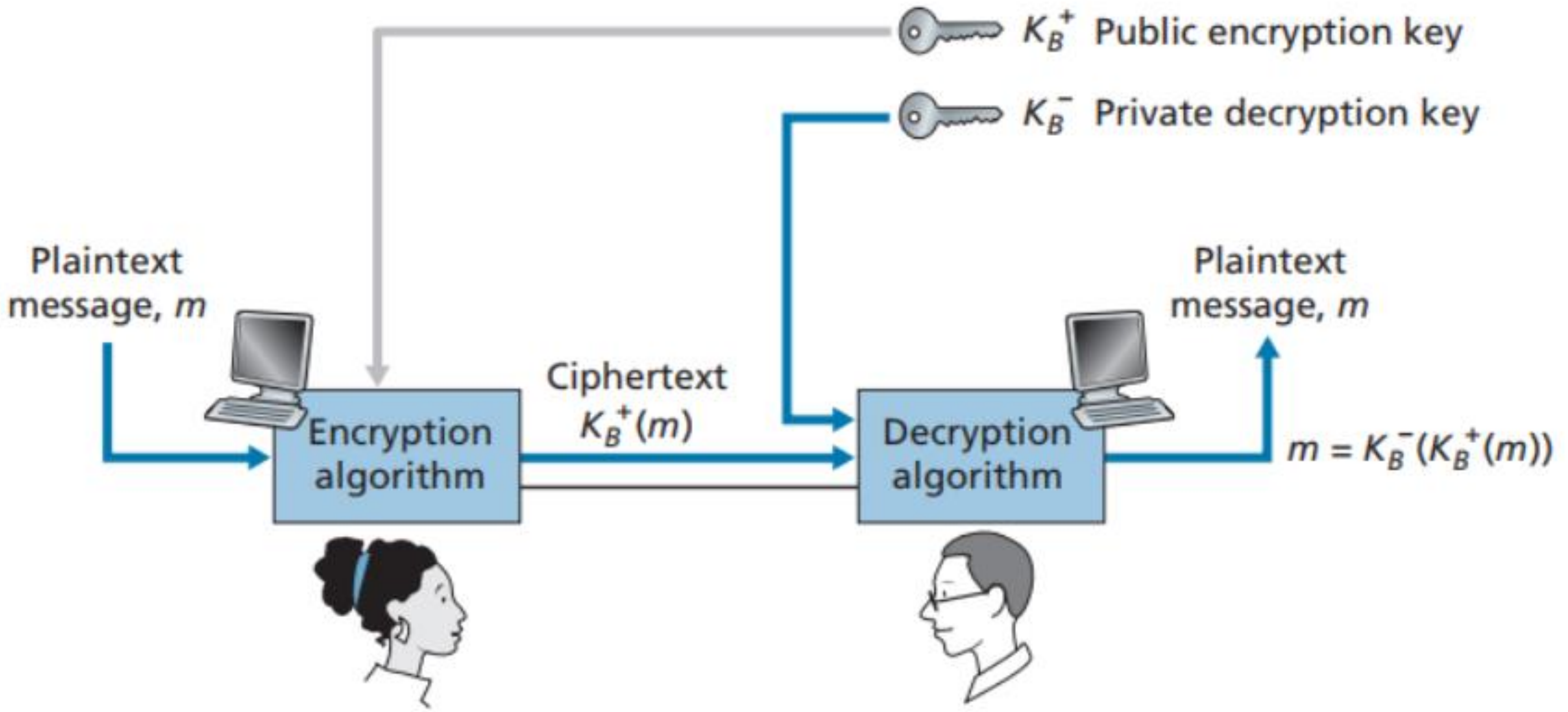
Active Attacks: Modification of Messages
like
dislike
Darth
Darth modifies message from Bob to Alice
Internet or other comms facility
Bob
Alice


Alex
Message from Alex that appears to be from Halim
Internet or other communications facility
Halim
Anita



Public-Key Encryption Structure

- **Public and private key**: a pair of keys that have been selected so if one is used for encryption, the other is used for decryption.

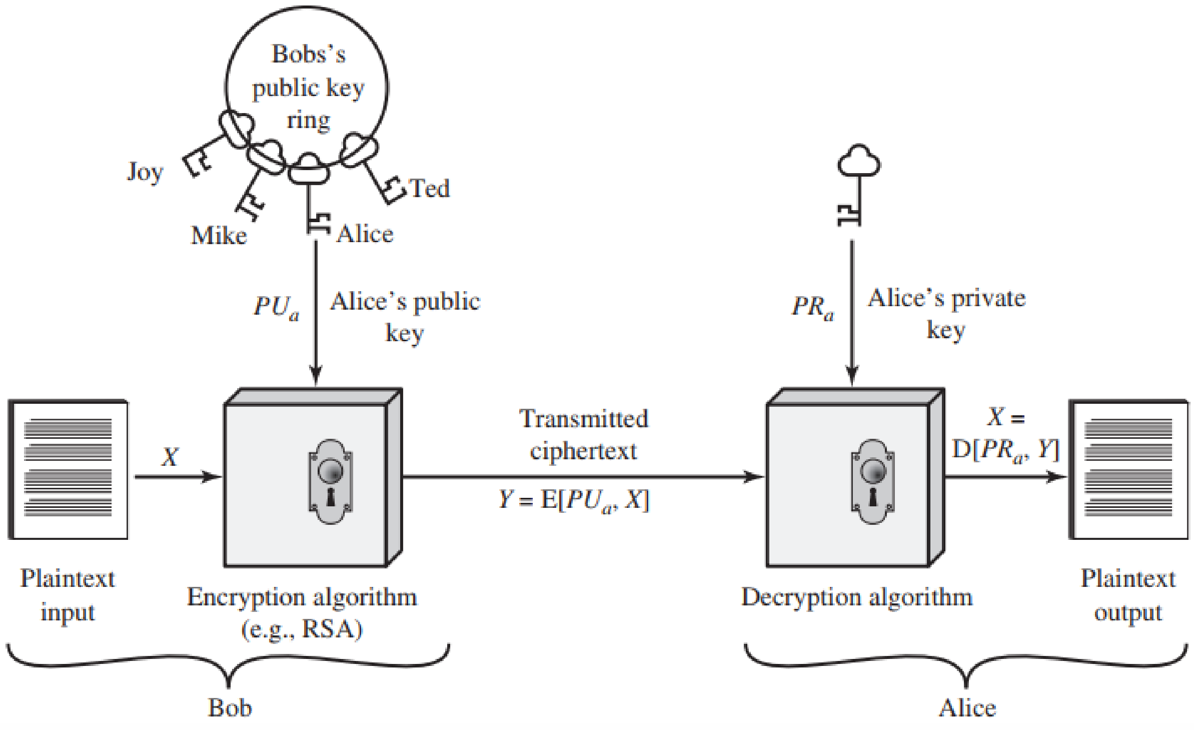


Steps of Creating and Using Public Keys

1. Each user generates a pair of keys to be used for the encryption and decryption of messages.
2. Each user places one of the two keys in a public register or other accessible file. This is the public key. The companion key is kept private. The private key is known only to its owner.
3. Any user example Bob can do the following:
   1. If Bob wishes to send a private message to Alice, Bob encrypts the message using Alice's public key. When Alice receives the message, she decrypts it using her private key. No other recipient can decrypt the message because only Alice knows Alice's private key.
   2. If Bob wishes to send a public message to all users, Bob encrypts the message using his private key. Anyone who knows the corresponding public key will then be able to decrypt the message.

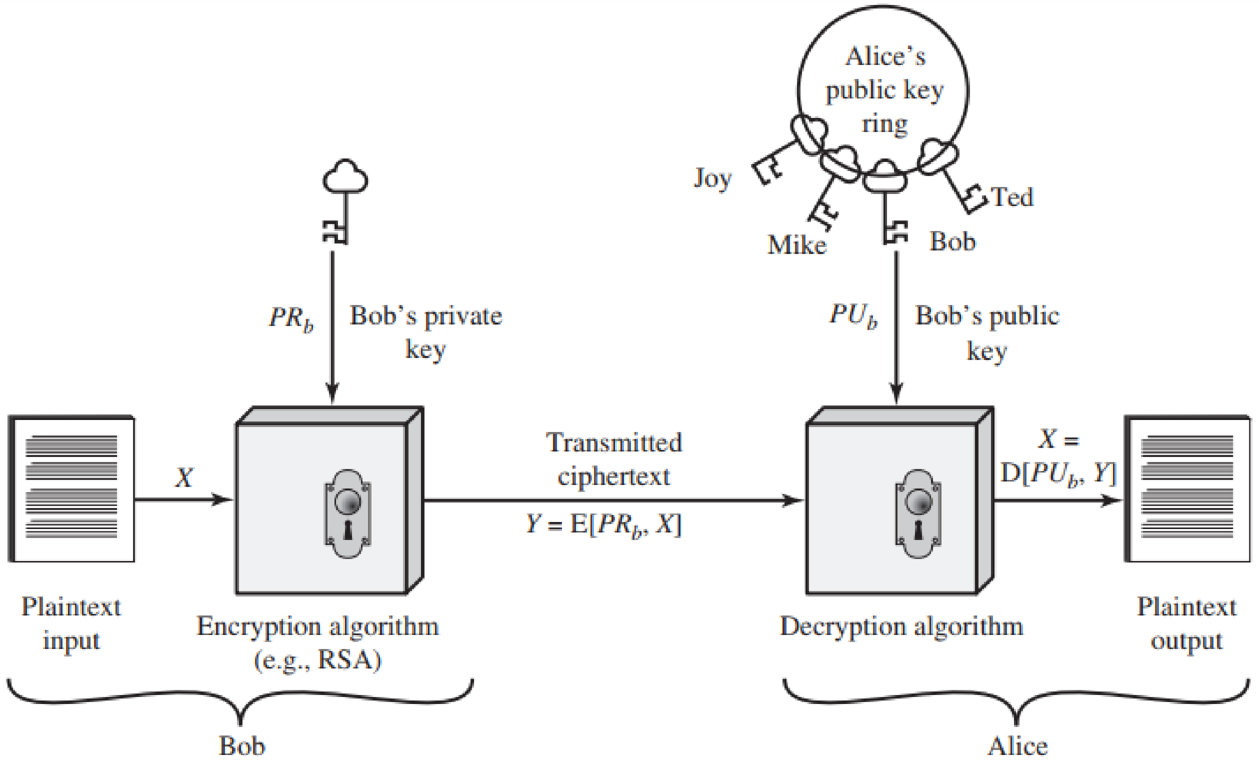
Public-Key Cryptography

Providing Confidentiality: Sending Private Message



Encryption with Public key

Providing authentication and/ or data integrity



Encryption with Private key

only Bob could encrypt the plaintext with Bob's private key.

Applications for Public-Key Cryptosystems

- Digital Signature
  - Public-Key Certificates
  - X.509 Scheme (IPsec, TLS, SSH
- Symmetric Key Distribution
  - Diffie-Hellman key Exchange
- Encryption of Secret keys
  - Digital Envelops

Asymmetric Encryption Algorithms

- **Diffie–Hellman key exchange**, 1976 or key agreement. The purpose of the algorithm is to enable two users to securely reach agreement about a shared secret that can be used as a secret key for subsequent **symmetric encryption of messages**.
- **RSA:** by Ron Rivest, Adi Shamir, and Len Adleman at MIT, 1977
- **Digital Signature Standard (DSS),** 1994, NIST, The DSS uses an algorithm that is designed to provide only the digital signature function. Unlike RSA, it cannot be used for encryption or key exchange.
- **Elliptic Curve Cryptography (ECC),** ECC provides the same cryptographic strength as the RSA-system, but with much smaller keys (faster). a 256-bit ECC key is the same as 3072-bit RSA key (which are 50% longer than the 2048-bit keys used today). ECC reduces processing overhead.

RSA public-key encryption Algorithm

- Most widely accepted and implemented approach to public-key encryption. RSA is a block cipher in which the plaintext and ciphertext are integers between 0 and n - 1 for some n.
- In April of 1994, a group working over the Internet and using over 1600 computers claimed the prize after only eight months of work. This challenge used a public-key size (length of n) of 129 decimal digits, or around 428 bits.
- A 1024-bit key size (about 300 decimal digits) is considered strong enough for virtually all applications.

RSA Algorithm

- RSA makes extensive use of arithmetic operations using modulo-n arithmetic.(19 mod 5 = 4)
- There are two interrelated components of RSA:
  - The choice of the public key and the private key
  - The encryption and decryption algorithm

Key Generator Algorithm

1. Choose two large prime numbers, p and q.
2. Compute $n = pq$ and $z = (p - 1)(q - 1)$.
3. Choose a number, e, less than n, that has no common factors with z.
4. Find a number, d, such that $ed - 1$ is exactly divisible by z. $ed \bmod z = 1$
5. The public key is the pair of numbers (n, e); his private key is the pair of numbers (n, d).

Encryption Algorithm

$$c = m^e \bmod n$$

Decryption Algorithm

$$m = c^d \bmod n$$

Lab Activity 1: Public Key Encryption

- Use the following online tool to create your **key pair** (Public/Private Key):
  - https://www.devglan.com/online-tools/rsa-encryption-decryption
- Share the **public key** with your classmates and keep your private key secret.
- Write a message and encrypt it using your Private Key,
- Share the message with your classmates and ask them to decrypt it using yours public key. (Discuss the applications of doing such activity?)
- Write another message and encrypt it using your friend Public Key.
- Share the message with your friend and ask hem to decrypt it using his Private key. (Discuss the applications of doing such activity?)

Lab Activity 2: Public Key Encryption

- Download PGP Encryption | gpg4win Kleopatra from the following link:
  - https://www.gpg4win.org/download.html
- **Create new key pair**, and share your public key with the class.
- Make a backup of your keys.
- **Import the public keys** of your classmates to the Kleopatra.
- Now you can encrypt data to achieve confidentiality, authentication or for integrity. How?
- Encrypt file using your public key (so you can only decrypt it using your private key)

UTM UNIVERSITI TEKNOLOGI MALAYSIA

Digital Signature and Key Management

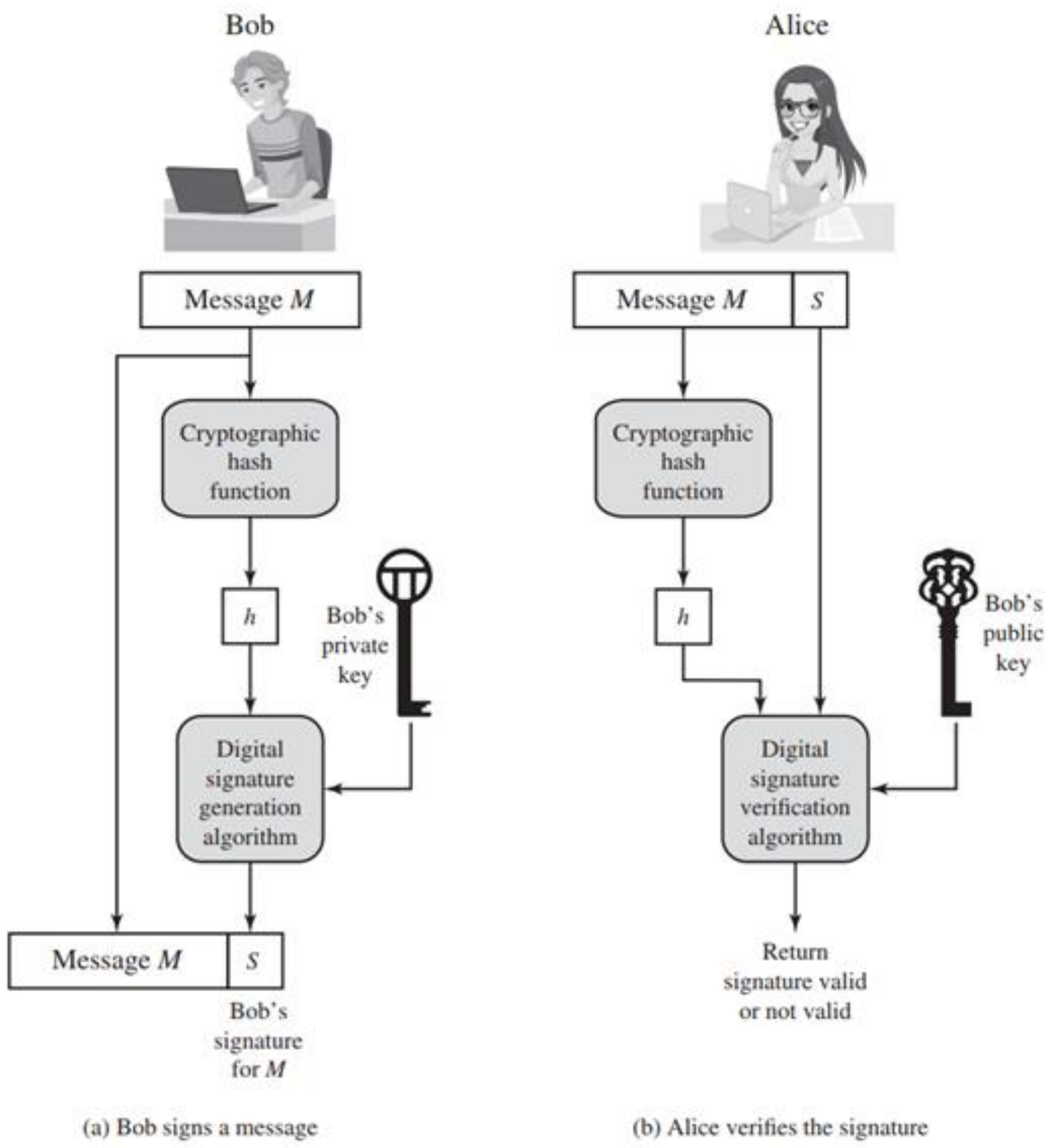
Digital signatures ensure that **signatures** are verified, authentic and legitimate.

Applications of Public-key Algorithms

- Public-key algorithms are used in a variety of applications. These applications fall into two categories:
  - **Digital signatures**,
  - **Key management and Distribution**
- With respect to key management and distribution, there are at least three distinct aspects to the use of public-key encryption in this regard:
  - The secure distribution of public keys
  - The use of public-key encryption to distribute secret keys
  - The use of public-key encryption to create temporary keys for message encryption

Digital Signature

- Public-key encryption can be used for authentication with a technique known as the digital signature.
- Digital signature provides a mechanism for verifying origin authentication, data integrity and signatory non-repudiation.


Bob
Alice
Message M
Message M
S
Cryptographic hash function
Cryptographic hash function
h
h
Bob's private key
Bob's public key
Digital signature generation algorithm
Digital signature verification algorithm
Message M
S
Bob's signature for M
Return signature valid or not valid
(a) Bob signs a message
(b) Alice verifies the signature

Digital Signature Example

Use Kleopatra to practice digital signature

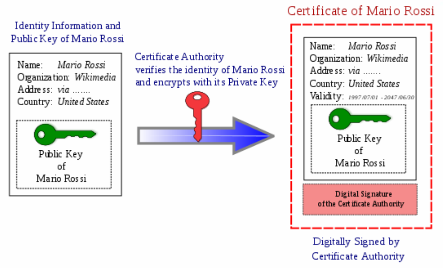
Kleopatra
Crypto Manager

1. Bob wants to send a message to Alice, he wants Alice to be certain that the message is indeed from him.
2. Bob uses a secure hash function, such as SHA-512, to generate a hash value for the message.
3. Bob use his private key to encrypt the hash (the signature) using digital signature generation algorithm.
4. Bob sends the message with the signature attached.
5. Alice receives the message plus signature
6. she (1) calculates a hash value for the message; (2) provides the hash value and Bob's public key as inputs to a digital signature verification algorithm.
7. If the algorithm returns the result that the signature is valid, Alice is assured that the message must have been signed by Bob.

Digital Signature Algorithms

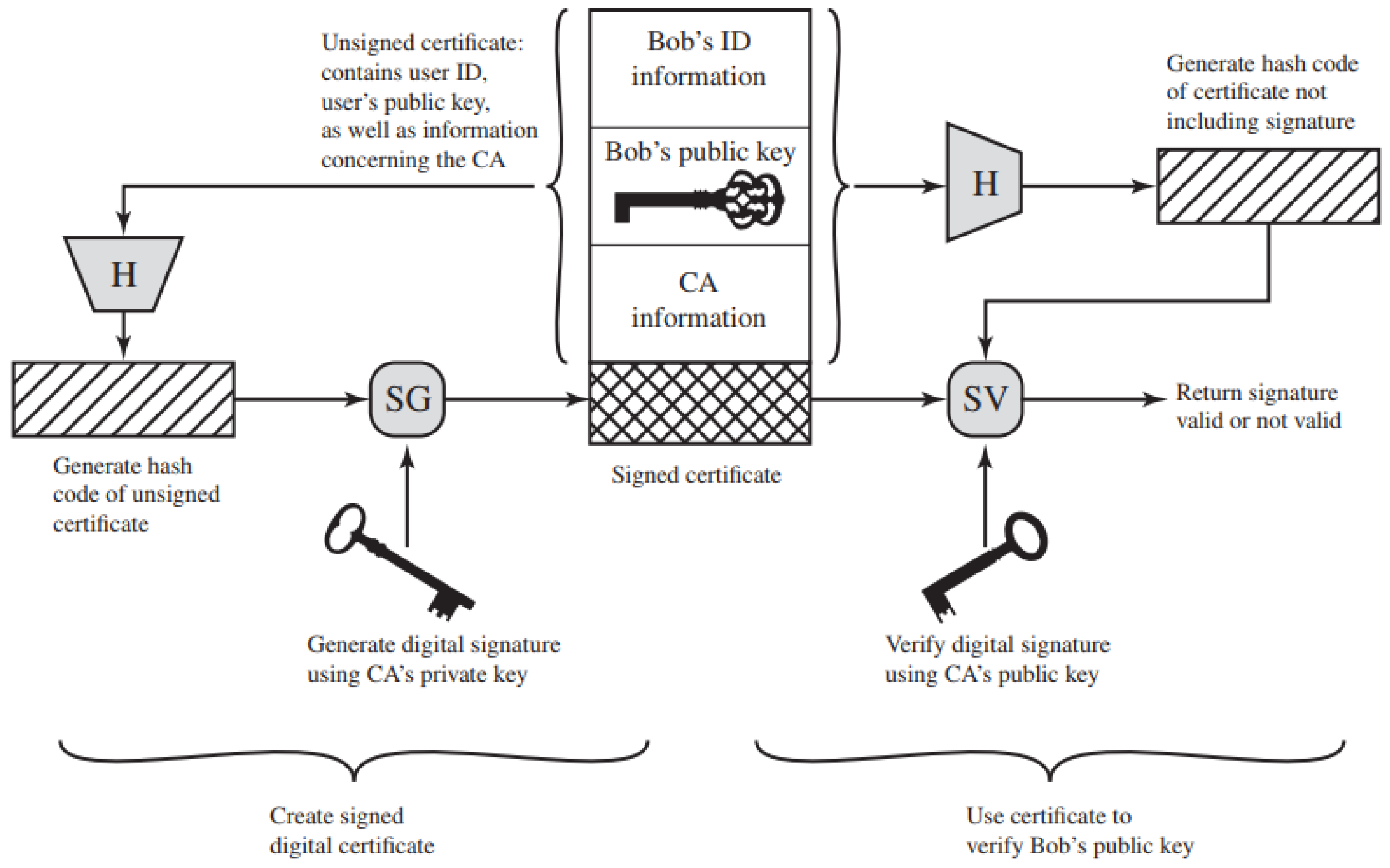
- **Digital Signature Algorithm** (DSA): The original NIST-approved algorithm, which is based on the difficulty of computing discrete logarithms.
- **RSA Digital Signature Algorithm**: Based on the RSA public-key algorithm.
- **Elliptic Curve Digital Signature Algorithm** (ECDSA): Based on elliptic-curve cryptography..

Public-Key Certificates

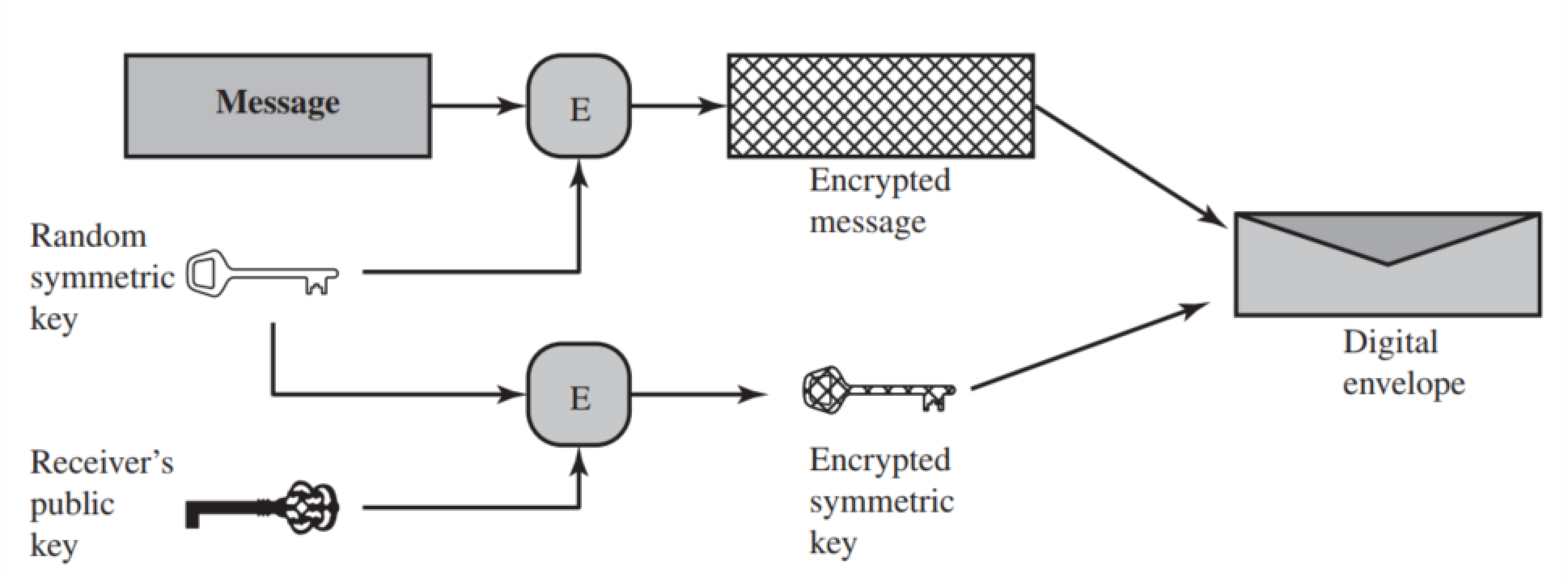

Identity Information and Public Key of Mario Rossi
Name: Mario Rossi
Organization: Wikimedia
Address: via ....
Country: United States
Public Key of Mario Rossi
Certificate Authority verifies the identity of Mario Rossi and encrypts with its Private Key
Certificate of Mario Rossi
Name: Mario Rossi
Organization: Wikimedia
Address: via ....
Country: United States
Validity: 1997/07/01 - 2047/06/30
Public Key of Mario Rossi
Digital Signature of the Certificate Authority
Digitally Signed by Certificate Authority

- Anyone can forge a public announcement. That is, some user could pretend to be Bob and send a public key to another participant or broadcast such a public key.
- The solution to this problem is the public-key certificate. In essence, a certificate consists of a public key plus a user ID of the key owner, with the whole block **signed by a trusted third party**.
- The third party is a **certificate authority (CA)** that is trusted by the user community, such as a government agency or a financial institution.
- A user can present his or her public key to the authority in a secure manner and obtain a signed certificate.
- The user can then publish the certificate. Anyone needing this user's public key can obtain the certificate and verify that it is valid by means of the attached trusted signature.

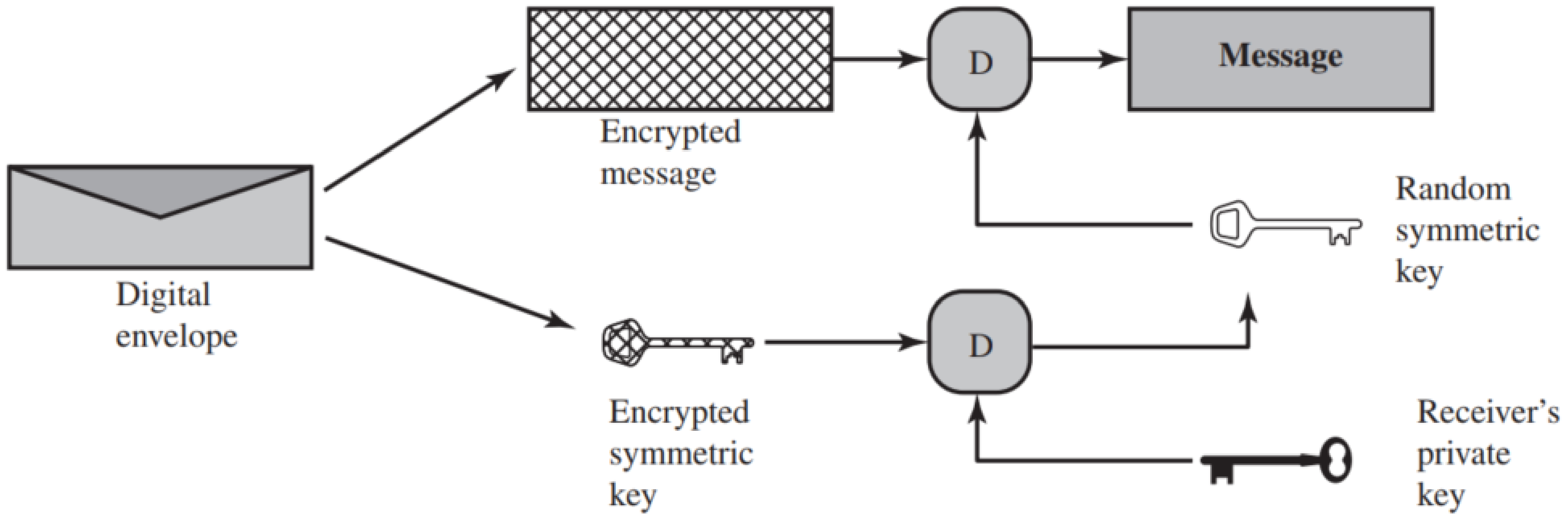
Public-Key Certificate


Unsigned certificate: contains user ID, user's public key, as well as information concerning the CA
Bob's ID information
Bob's public key
CA information
H
Generate hash code of certificate not including signature
H
Generate hash code of unsigned certificate
SG
Signed certificate
SV
Return signature valid or not valid
Generate digital signature using CA's private key
Verify digital signature using CA's public key
Create signed digital certificate
Use certificate to verify Bob's public key

Digital Envelopes - Creation of a digital envelope



Digital Envelopes - Opening a digital envelope



Reading Topics

- Application of Public Key Encryption
- Symmetric Key Exchange Using Public-Key Encryption
- Digital Envelopes
- The Use of Random Numbers for Public Key

Next Session

- **User Authentication.**