Chapter 2

Cryptographic Tools – Part 2

Course Instructor: Dr. Nausheen Shoaib

Asymmetric Encryption Algorithms

RSA (Rivest, Shamir, Adleman)	Developed in 1977	Most widely accepted and implemented approach to public-key encryption	Block cipher in which the plaintext and ciphertext are integers between 0 and *n*-1 for some *n*.
Diffie-Hellman key exchange algorithm	Enables two users to securely reach agreement about a shared secret that can be used as a secret key for subsequent symmetric encryption of messages	Limited to the exchange of the keys	
Digital Signature Standard (DSS)	Provides only a digital signature function with SHA-1	Cannot be used for encryption or key exchange	
Elliptic curve cryptography (ECC)	Security like RSA, but with much smaller keys		

Repudiation vs. Non Repudiation

- A repudiation attack happens **when an application or system does not adopt controls to properly track and log users' actions**, thus permitting malicious manipulation or forging the identification of new actions

- non-**repudiation translates into a method of assuring that something that's actually valid cannot be disowned or denied**

Digital Signatures & Key Management

- key management and distribution, there are at least three distinct aspects to the use of public-key encryption:

1. The secure distribution of public keys

2. The use of public-key encryption to distribute secret keys

3. The use of public-key encryption to create temporary keys for message encryption

Digital Signatures

- NIST FIPS PUB 186-4 defines a digital signature as:

  ”The result of a cryptographic transformation of data that, when properly implemented, provides a mechanism for verifying origin authentication, data integrity and signatory non-repudiation.”

- Thus, a digital signature is a data-dependent bit pattern, generated by an agent as a function of a file, message, or other form of data block

- FIPS 186-4 specifies the use of one of three digital signature algorithms:

  - Digital Signature Algorithm (DSA)
  - RSA Digital Signature Algorithm
  - Elliptic Curve Digital Signature Algorithm (ECDSA)

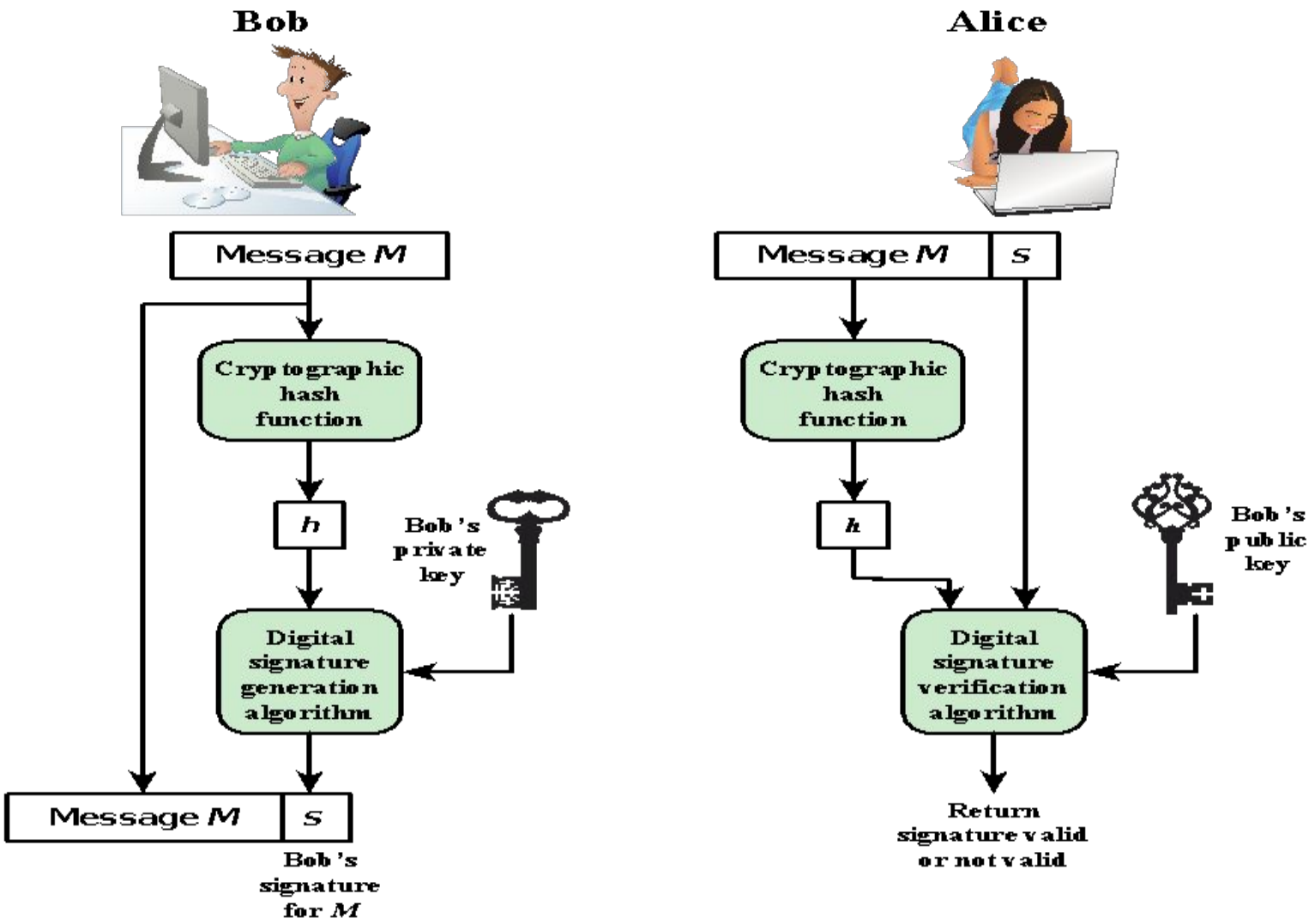


Figure 2.7 Simplified Depiction of Essential Elements of Digital Signature Process

Public Key Certificates

The key steps can be summarized as follows:

1. User software (client) creates a pair of keys: one public and one private.
2. Client prepares an unsigned certificate that includes the user ID and user's public key.

3. User provides the unsigned certificate to a CA in some secure manner. This might require a face-to-face meeting, the use of registered e-mail, or happen via a Web form with e-mail verification.

4. CA creates a signature as follows:

a. CA uses a hash function to calculate the hash code of the unsigned certificate. A hash function is one that maps a variable-length data block or message into a fixed-length value called a hash code, such as SHA family that we will discuss in Sections 2.2 and 21.1.
b. CA generates digital signature using the CA's private key and a signature generation algorithm.
.

Public Key Certificates

5. CA attaches the signature to the unsigned certificate to create a signed certificate

6. CA returns the signed certificate to client.

7. Client may provide the signed certificate to any other user.

8. Any user may verify that the certificate is valid as follows:

a. User calculates the hash code of certificate (not including signature).

b. User verifies digital signature using CA's public key and the signature verification algorithm.

Public Key Certificates

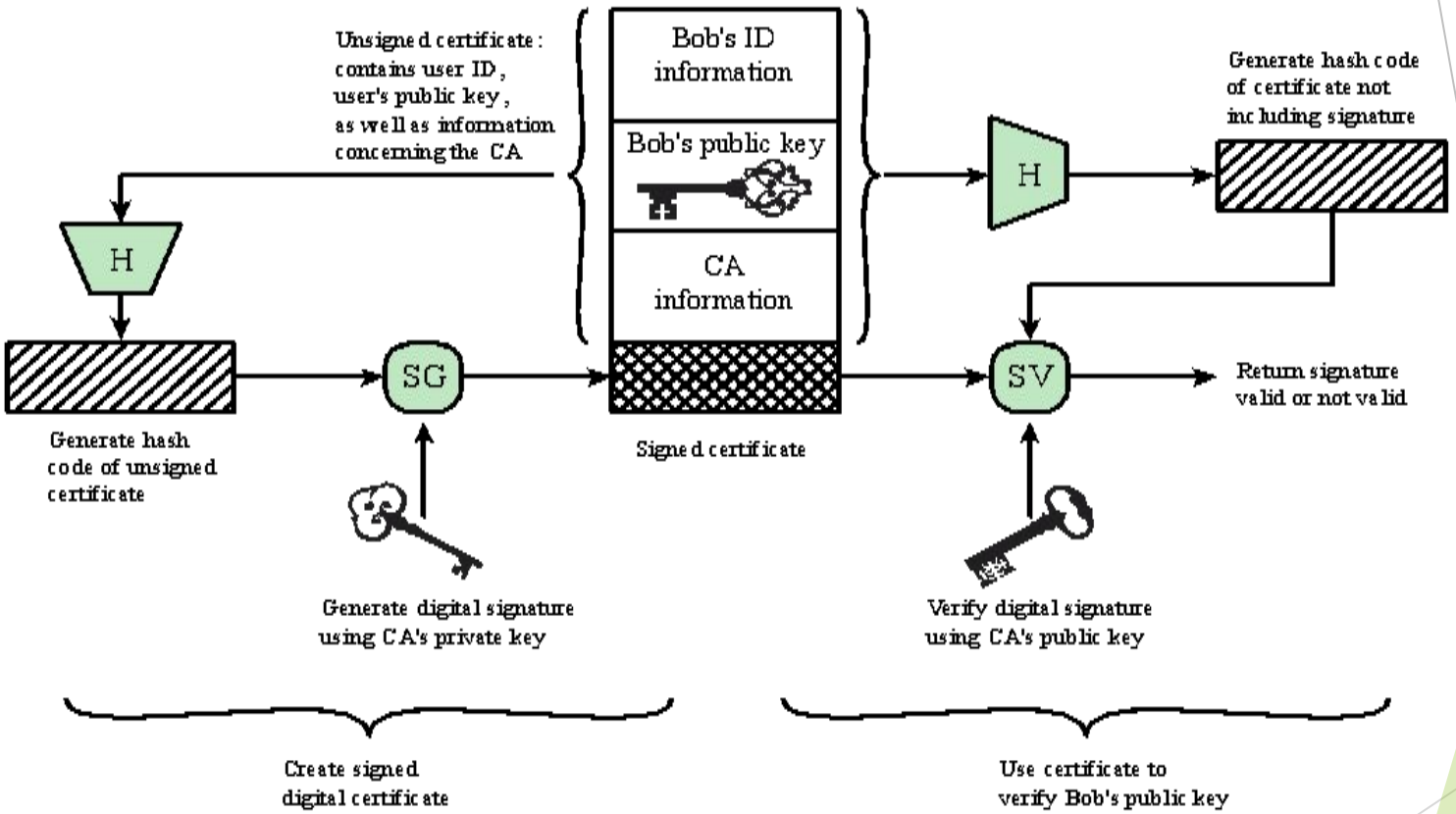


Figure 2.8 Public-Key Certificate Use

Symmetric Key Exchange Using Public-Key Encryption

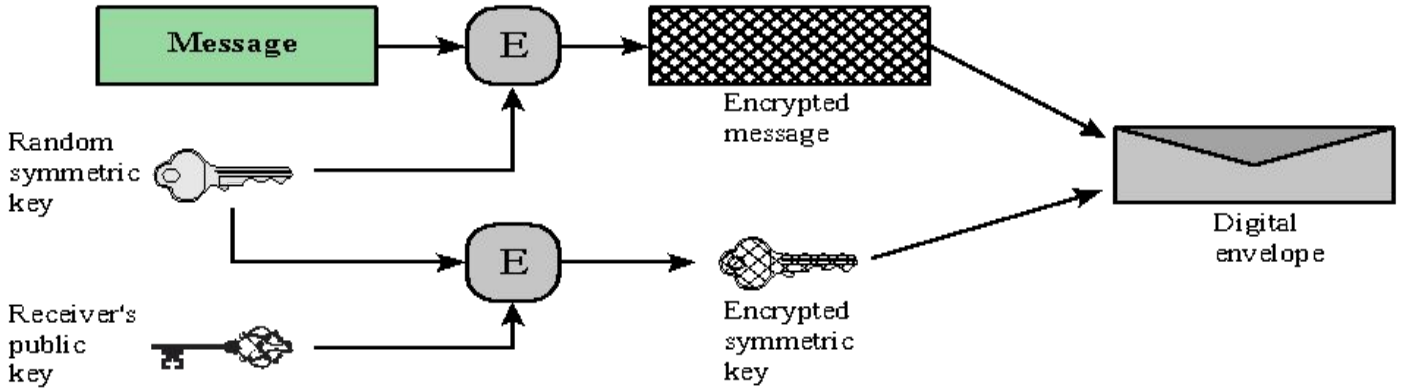
With symmetric encryption, a fundamental requirement for two parties to communicate securely is that they share a secret key.

One approach is the use of Diffie–Hellman key exchange. This approach is indeed widely used. However, it suffers the drawback that, in its simplest form, Diffie– Hellman provides no authentication of the two communicating partners. There are variations to Diffie–Hellman that overcome this problem

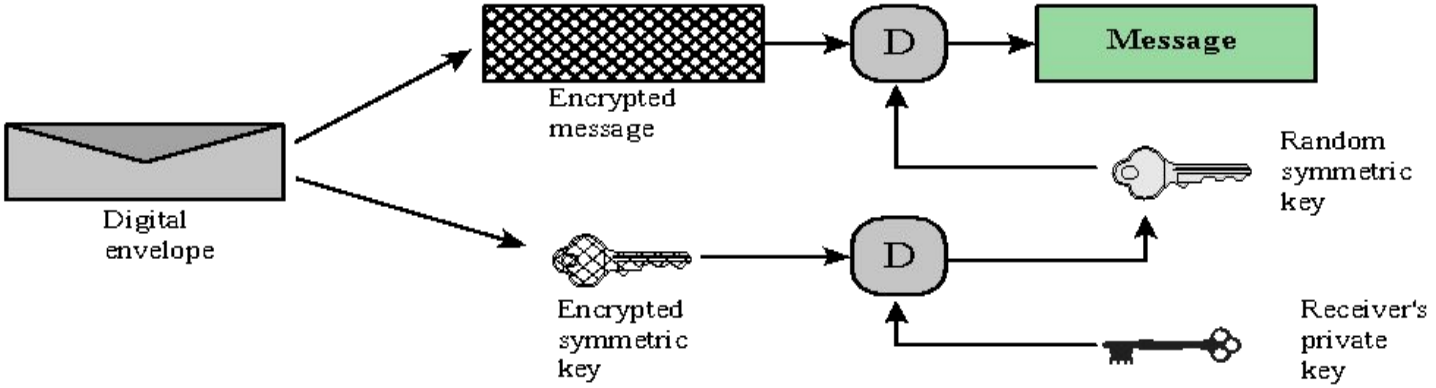
Symmetric Key Exchange Using Public-Key Encryption

- **Digital Envelope:** Another application in which public-key encryption is used to protect a symmetric key is the digital envelope.

- sealed envelope containing an unsigned letter.



(a) Creation of a digital envelope



(b) Opening a digital envelope

Figure 2.9 Digital Envelopes

Random Numbers

Uses include generation of:

- Keys for public-key algorithms
- Stream key for symmetric stream cipher
- Symmetric key for use as a temporary session key or in creating a digital envelope
- Handshaking to prevent replay attacks
- Session key

Random Number Requirements

Randomness

- Criteria:
  - Uniform distribution
    - Frequency of occurrence of each of the numbers should be approximately the same
  - Independence
    - No one value in the sequence can be inferred from the others

Unpredictability

- Each number is statistically independent of other numbers in the sequence
- Opponent should not be able to predict future elements of the sequence on the basis of earlier elements

Random versus Pseudorandom

Cryptographic applications typically make use of algorithmic techniques for random number generation

- Algorithms are deterministic and therefore produce sequences of numbers that are not statistically random

Pseudorandom numbers are:

- Sequences produced that satisfy statistical randomness tests
- Likely to be predictable

True random number generator (TRNG):

- Uses a nondeterministic source to produce randomness
- Most operate by measuring unpredictable natural processes
  - e.g. radiation, gas discharge, leaky capacitors
- Increasingly provided on modern processors

Practical Application: Encryption of Stored Data

Common to encrypt transmitted data

Much less common for stored data

- There is often little protection beyond domain authentication and operating system access controls
- Data are archived for indefinite periods
- Even though erased, until disk sectors are reused data are recoverable

Approaches to encrypt stored data:

- Use a commercially available encryption package
- Back-end appliance
- Library based tape encryption
- Background laptop/PC data encryption

Summary

- Confidentiality with symmetric encryption
  - Symmetric encryption
  - Symmetric block encryption algorithms
  - Stream ciphers
- Message authentication and hash functions
  - Authentication using symmetric encryption
  - Message authentication without message encryption
  - Secure hash functions
  - Other applications of hash functions
- Random and pseudorandom numbers
  - The use of random numbers
  - Random versus pseudorandom
- Public-key encryption
  - Structure
  - Applications for public-key cryptosystems
  - Requirements for public-key cryptography
  - Asymmetric encryption algorithms
- Digital signatures and key management
  - Digital signature
  - Public-key certificates
  - Symmetric key exchange using public-key encryption
  - Digital envelopes
- Practical Application: Encryption of Stored Data