INTRODUCTION

Security Goals:

Confidentiality: Confidentiality is probably the most common aspect of information security. We need to protect our confidential an organization needs to guard against those malicious actions that endanger the confidentiality of its information. In the military, concealment of sensitive information is the major concern. In industry. hiding some information from competitors is crucial to the operation of the organization. In banking, customers' accounts need to kept secret. As we will see later in this chapter. confidentiality only applies to the storage of the information. it also applies to transmission of information- When we send a piece of information to store in a remote computer or when we retrieve a piece of information from a remote computer. we need to conceal it during transmission.

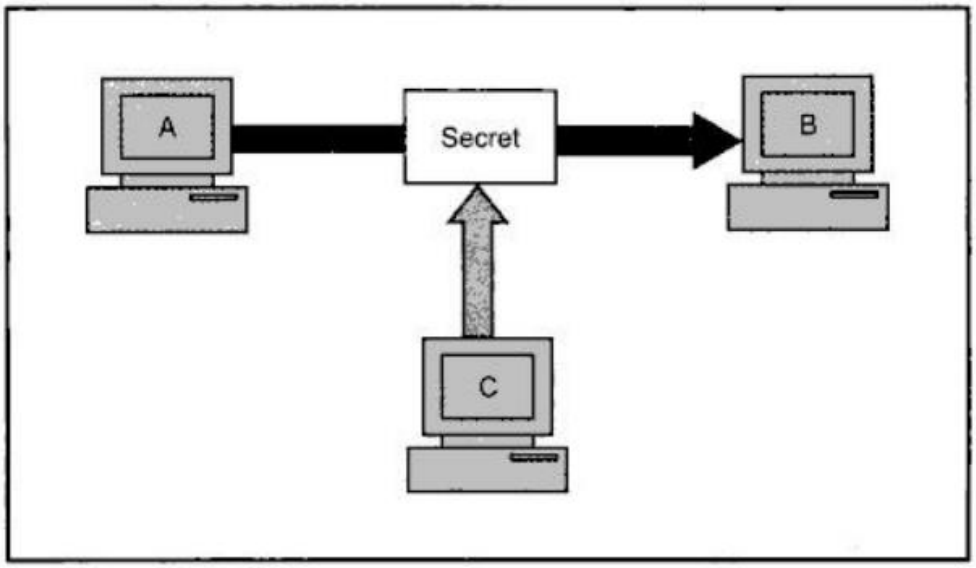

Fig. 1.4 *Loss of confidentiality*

Interception causes loss of message confidentiality.

Authentication: Authentication mechanisms help establish proof of identities. The authentication ensures tha the origin Of a electronic message or document is correctly identified. For instance, that user C sends an electronic document over the Internet to user B. However, the trouble is that user C had posed as user A when she sent this document to user B. How would user B know that the message has come from user C, who is posing as user A? A real-life example of this could be the case of a user C, posing as user A, sending a funds transfer request (from A's account to C's account) to bank B. The bank might happily transfer the funds from A's account to C's account after all, it would think that user A has requested for the funds transfer! This concept is shown in Fig. . This of attack is called as fabrication.

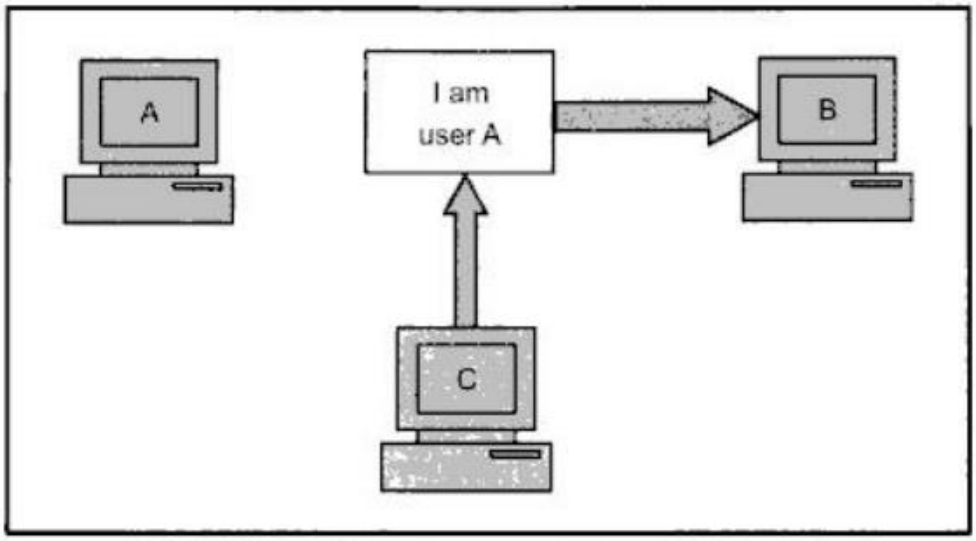

Fig. 1.5 *Absence of authentication*

Fabrication is possible in absence of proper authentication mechanisms.

Integrity: When the contents of a message are changed after the sender sends it, but before it reaches the intended recipient, we say that the integrity of the message is lost. For example, suppose you write a check for $ 100 to pay for the goods bought from the US. However, when you see your next account statement, you are startled to see that the check resulted in a payment of $10! This is the case for loss of message integrity. Conceptually, this is shown in Fig. 1.6. Here, user C tampers with a message originally sent by user A, which is actually destined for user B. User C somehow manages to access it, change its contents and send the changed message to user B. User B has no way of knowing that the contents of the message were changed after user A had sent it. User A also does not know about this change. This type of attack is called as modification.

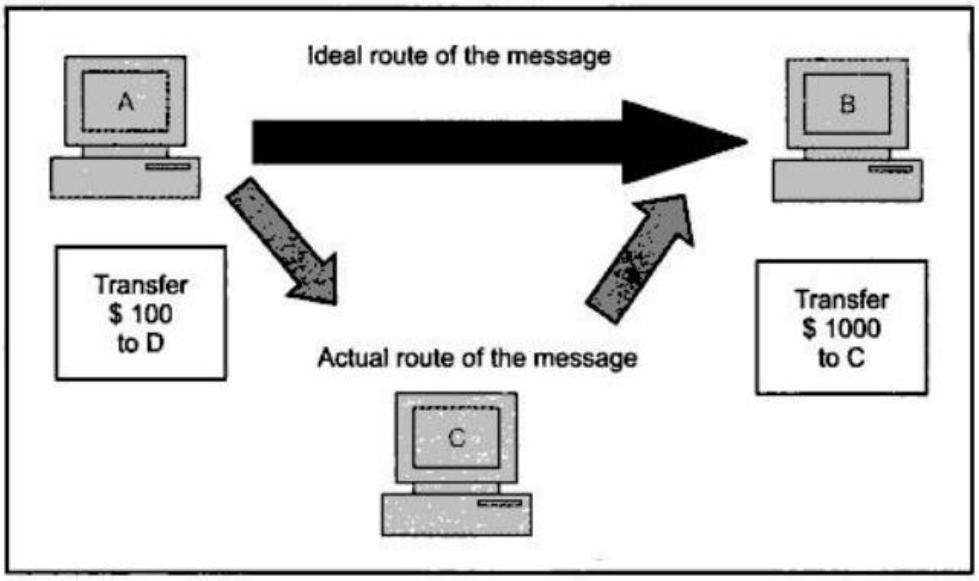

Fig. 1.6 *Loss of integrity*

Modification causes loss of message integrity

Non-repudiation: There are situations where a user sends a message and later on refuses that she had sent that message. For instance, user A could send a funds transfer request to bank B over the Internet. After the bank performs the funds transfer as per A's instructions, A could claim that she never sent the funds transfer instruction to the bank! Thus, A repudiates or denies, her funds transfer instruction. The principle of non-repudiation defeats such possibilities of denying something, having done it. This is shown in Fig. Non-repudiation does not allow the sender of a message to refute the claim of not sending that message.

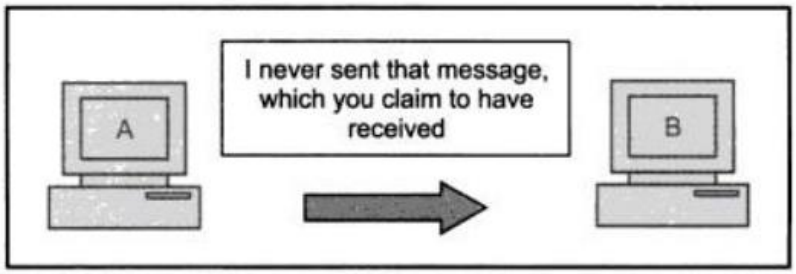

Fig. 1.7 *Establishing non-repudiation*

Access Control: The principle of access control determines who should be able to access what. For instance, we should be able to that user A can view the records in a database, but cannot update them. However, user B might be allowed to make updates as well. An access control mechanism can be set up to ensure this. Access control is broadly related to two areas: role management and rule management. Role management concentrates on the user side (which user can do what), whereas rule management focuses on the resources side (which resource is accessible and under what circumstances). Based on the decisions taken here, an access control matrix is prepared, which lists the users against a list of items they can access (e.g. it can say that user A can write to file X, but can only update files Y and Z). An Access Control List (ACL) is a subset of an access control matrix.

Access control specifies and controls who can access what.

Availability: The principle of availability states that resources (i.e. information) should available to authorized parties at all times. For example, due to the intentional actions of an unauthorized user C, an authorized user A may not be able to contact a server computer B, as shown in Fig. 1.8. This would defeat the principle of availability. Such an attack is called as interruption.

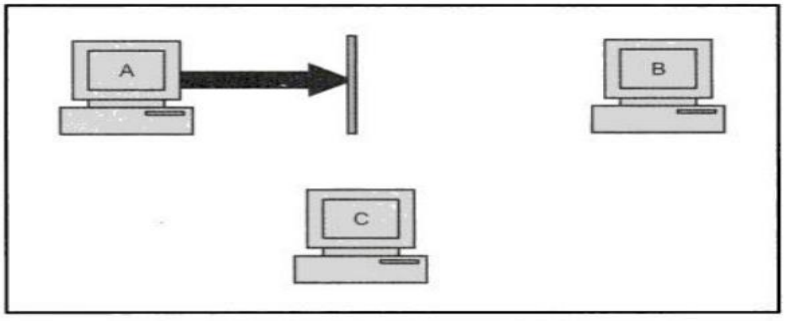

Fig. 1.8 *Attack on availability*

Interruption puts the availability of resources in danger.

Cryptography: Cryptography is the art and science of making a cryptosystem that is capable of providing information security. It refers to the design of mechanisms based on mathematical algorithms that provide fundamental information security services.

Cryptanalysis: The art and science of breaking the cipher text is known as cryptanalysis. Cryptanalysis is the sister branch of cryptography and they both co-exist . It involves the study of cryptography mechanism with intention to break them.

Cryptography concerns with the design of cryptosystem while cryptanalysis studies the breaking of cryptosystem.

Types of Cryptography: In general there are three types of cryptography –

1. Symmetric key cryptography:
   It is an encryption system where the sender and receiver of message use a single common key to encrypt and decrypt message . The most popular Symmetric key cryptography system is **DES(Data Encryption System)**.
2. Asymmetric key Cryptography:
   In this system a pair of key is used to encrypt and decrypt information. A public key is used for encryption and a private key is used for decryption. Public key and private key are different . Even if the public key is known by everyone the private key is only known by the intended receiver.
3. Hash Function:
   In hash Function there is no usage of any key . A hash value with fixed length is calculated as per the plain text which makes it impossible for contents of plain text to be recovered.

Advantages of Cryptography:

I. Cryptography can be used for access control to ensure that only parties with proper permission have access to a resource . Only those with the correct key can access the resources.
II. For secure online communications , cryptography is crucial. It offers secure mechanism for transmitting private information like passwords, bank details and other sensitive data.
III. Cryptography can assists firms in meeting a variety of legal requirements, including data protection and privacy legislation.

IV. Cryptography aids in the defense against various types of assaults , including replay and man-in-middle attacks.

Application of Cryptograph:

The some of the applications of cryptography are given below –

i. Cryptography is widely utilize in computer security, particularly when creating and maintain passwords.
ii. To safeguard transactions and prevent fraud complex algorithms and cryptography key are used to safeguard transactions.
iii. Cryptography is used for authentication in many different situations.
iv. Online browsing security is provide by the use of cryptography.
v. Digital Signature are created using cryptography.

Question- Answers

Difference Between Symmetric and Asymmetric Key Encryption

Symmetric Key Encryption: Encryption is a process to change the form of any message in order to protect it from reading by anyone. In Symmetric-key encryption the message is encrypted by using a key and the same key is used to decrypt the message which makes it easy to use but less secure. It also requires a safe method to transfer the key from one party to another.
Asymmetric Key Encryption: Asymmetric Key Encryption is based on public and private key encryption techniques. It uses two different key to encrypt and decrypt the message. It is more secure than the symmetric key encryption technique but is much slower.

Symmetric Key Encryption	Asymmetric Key Encryption
It only requires a single key for both encryption and decryption.	It requires two keys, a public key and a private key, one to encrypt and the other one to decrypt.
The size of cipher text is the same or smaller than the original plain text.	The size of cipher text is the same or larger than the original plain text.
The encryption process is very fast.	The encryption process is slow.
It is used when a large amount of data is required to transfer.	It is used to transfer small amounts of data.

Symmetric Key Encryption	Asymmetric Key Encryption
It only provides confidentiality.	It provides confidentiality, authenticity, and non-repudiation.
The length of key used is 128 or 256 bits	The length of key used is 2048 or higher
In symmetric key encryption, resource utilization is low as compared to asymmetric key encryption.	In asymmetric key encryption, resource utilization is high.
It is efficient as it is used for handling large amount of data.	It is comparatively less efficient as it can handle a small amount of data.
Security is less as only one key is used for both encryption and decryption purpose.	It is more secure as two keys are used here- one for encryption and the other for decryption.
The Mathematical Representation is as follows- P = D (K, E(P)) where K –> encryption and decryption key P –> plain text D –> Decryption E(P) –> Encryption of plain text	The Mathematical Representation is as follows- P = D(Kd, E (Ke,P)) where Ke –> encryption key Kd –> decryption key D –> Decryption E(Ke, P) –> Encryption of plain text using encryption key Ke . P –> plain text
Examples: 3DES, AES, DES and RC4	Examples: Diffie-Hellman, ECC, El Gamal, DSA and RSA

Block Cipher and Stream Cipher

Block Cipher and Stream Cipher are two types of symmetric key ciphers used for encrypting data. Here's a brief overview of each:

Block Cipher:

- Encrypts data in fixed-size blocks (e.g., 64, 128, or 256 bits) using a shared, secret key.
- Examples: AES, DES, 3DES, Serpent, and Blowfish.
- Uses both "confusion" and "diffusion" principles for the conversion required for encryption and decryption.
- Suitable for encrypting large amounts of data and providing strong security.

Stream Cipher:

- Encrypts data one byte (or bit) at a time using a shared, secret key.
- Examples: RC4, Salsa20, Rabbit, and HC-256.
- Works with an infinite stream of pseudorandom bits (keystream) generated by a key.
- Suitable for encrypting data streams and providing fast encryption.

Illustration:

Suppose we have a plaintext message "HELLO". In a block cipher, the message is divided into fixed-size blocks (e.g., 2 characters per block) and encrypted using a secret key. In a stream cipher, each character of the message is encrypted one at a time using the secret key and a pseudorandom keystream.

In summary, block ciphers encrypt data in fixed-size blocks, providing strong security for large amounts of data, while stream ciphers encrypt data one byte (or bit) at a time, offering fast encryption for data streams.

Stream Cipher generation technique LFSR

Linear Feedback Shift Register (LFSR) is a commonly used technique for generating a stream of pseudorandom bits in stream ciphers. It is a shift register that operates on binary data, where the output bit is derived from a linear combination of the shift register's contents. LFSR is defined by its feedback polynomial, which determines the shift register's behavior.

Here's how LFSR works as a stream cipher generation technique:

1. Initialization: Start with an initial seed value, which is the initial state of the shift register. The seed value should be a nonzero value.
2. Feedback Polynomial: Define a feedback polynomial that determines the tap positions within the shift register. The tap positions are the positions from which the bits are extracted to form the output bit.
3. Shift and Feedback: At each clock cycle, the bits in the shift register are shifted to the right, and the rightmost bit is output as the keystream bit. The feedback polynomial is then applied to determine the new bit to be entered into the leftmost position.
4. Repeat: The shifting and feedback process is repeated for each clock cycle to generate a stream of pseudorandom bits, also known as the keystream.

Example: Let's consider an LFSR with a 4-bit shift register and the feedback polynomial x^4 + x + 1 (which corresponds to the taps at positions 4 and 1).

Initialization:

- Set the initial state of the shift register to a nonzero seed value, e.g., 1010.

Shift and Feedback:

- Clock Cycle 1: The rightmost bit (0) is output as the keystream bit. The feedback polynomial is applied, resulting in a new bit (1) to be entered into the leftmost position. The shift register becomes 0101.
- Clock Cycle 2: The rightmost bit (1) is output. The feedback polynomial is applied, resulting in a new bit (0). The shift register becomes 1010.
- Clock Cycle 3: The rightmost bit (0) is output. The feedback polynomial is applied, resulting in a new bit (1). The shift register becomes 0101.

- Clock Cycle 4: The rightmost bit (1) is output. The feedback polynomial is applied, resulting in a new bit (1). The shift register becomes 1011.
- Clock Cycle 5: The rightmost bit (1) is output. The feedback polynomial is applied, resulting in a new bit (1). The shift register becomes 1101.

This process continues, generating a stream of pseudorandom bits (the keystream) as the rightmost bits are output at each clock cycle. The keystream can then be combined with the plaintext using bitwise XOR operation to encrypt the message.

(Note that the strength of an LFSR-based stream cipher depends on the properties of the feedback polynomial, the size of the shift register, and the initialization process.)