Computer Security

Concept of Computer Security

Computer Security The protection afforded to an automated information system in order to attain the applicable objectives of preserving the **integrity**, **availability**, and **confidentiality** of information system resources (includes hardware, software, firmware, information/data, and telecommunications).

Main Security Objectives — CIA Triad

Confidentiality: The requirement that private (privacy) or secret information not be disclosed to unauthorized individuals. Confidentiality protection applies to data in storage, during processing, and while in transit.

Integrity:

- Data integrity: The property that data has not been altered in an unauthorized manner while in storage, during processing, or while in transit.
- System integrity: The quality that a system has when performing the intended function in an unimpaired manner, free from unauthorized manipulation.

Availability: The requirement intended to assure that systems work promptly and service is not denied to authorized users.

Further Security Objectives

Accountability: The requirement that actions of an entity may be traced uniquely to that entity. Accountability is often an organizational policy requirement and directly supports non-repudiation, deterrence, intrusion detection and prevention.

Assurance: The basis for confidence that the security measures, both technical and operational, work as intended to protect the system and the information it processes. The other four security objectives (integrity, availability, confidentiality, and accountability) have been adequately met by a specific implementation.

Security Objective Interdependencies

- *Confidentiality is dependent on Integrity*: If the integrity of the system is lost, then there is no longer a reasonable expectation that the confidentiality mechanisms are still

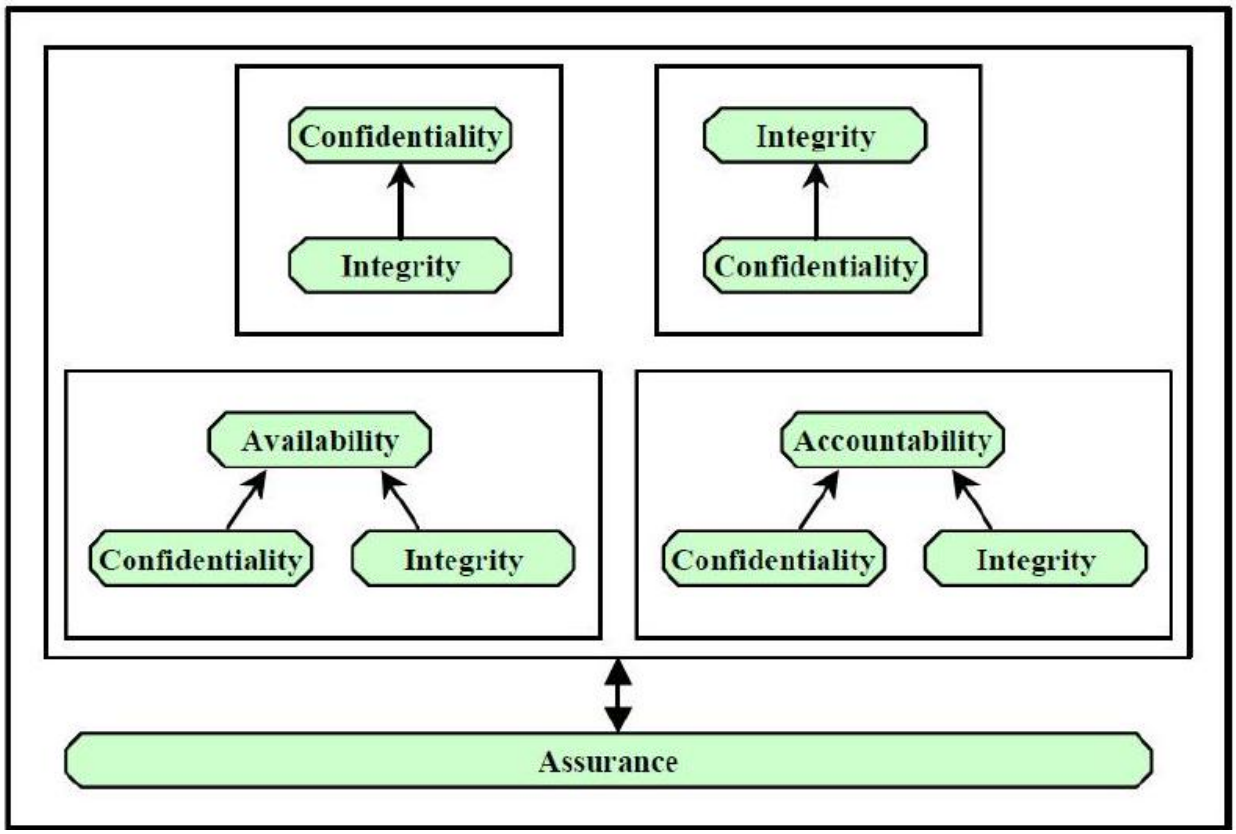

Figure 1: Security Objectives Dependencies

valid.

- *Integrity is dependent on Confidentiality*: If the confidentiality of certain information is lost (e.g., the superuser password), then the integrity mechanisms are likely to be by-passed.
- *Availability and Accountability are dependent on Confidentiality*: If confidentiality is lost for certain information (e.g., superuser password), the mechanisms implementing Availability and Accountability are easily bypassable.
- *Availability and Accountability are dependent on Integrity*: If system integrity is lost, then confidence in the validity of the mechanisms implementing Availability and Accountability is also lost.

Physical Security Control

Physical security controls include such things as data center perimeter fencing, locks, guards, access control cards, biometric access control systems, surveillance cameras and intrusion detection sensors.

Malware

A program that is inserted into a system, usually covertly, with the intent of compromising the confidentiality, integrity, or availability of the victim's data, applications, or operating system or otherwise annoying or disrupting the victim.

- Malicious software = malware
- Classification: modes of **propagation** and **payloads**.
  - Propagation mechanisms include those used by viruses, worms, and Trojans. Payloads include system corruption, bots, phishing, spyware, and rootkits.

Classification

Propagation mechanisms

- infection of existing executable (virus)
- exploit of software vulnerabilities (worm)
- social engineering attacks (to install trojans, to respond to phishing attacks, spams)

Host program

- is needed (virus)
- is not needed (worm)

Replication

- does not replicate(trojans, spam e-mail)

- does replicate (virus, worm)

Payload actions

- corruption of system or data files
- theft of service (in order to make the system a zombie agent of the attack as part of a botnet)
- theft of information (passwords, personal details)

Encryptions

	Symmetric	Asymmetric
Secrecy of the keys	keys are kept secret (K)	(PK, SK) public and secret key
Handling the keys	key exchange algorithms are needed	Public Key Infrastructure
Computational time	fast algorithms	slow algorithms
Size of messages	large size	small size
Examples	TDES, AES	RSA, EIGamal, elliptic curve encryption

Symmetric Encryption Scheme

Definition A triple $SE =$ (Key, Enc, Dec) is a symmetric encryption scheme, if

- Key: a key-generation algorithm that outputs a key $K \in \mathcal{K}$ for a security parameter k.
- Enc: an encryption algorithm that takes as input a key $K \in \mathcal{K}$ and a plaintext message $m \in \mathcal{P}$ and outputs a ciphertext $c \in \mathcal{C}$.

$$c = \mathrm{Enc}_K(m)$$

- Dec: a decryption algorithm that takes as input a key K and a ciphertext c and outputs a plaintext m.

$$m = \mathrm{Dec}_K(c)$$

Asymmetric Encryption Scheme

Definition A triple $AE =$ (Key, Enc, Dec) is an asymmetric encryption scheme, if

- Key: a key-generation algorithm that for a security parameter k outputs a key pair (PK, SK), where PK is the public and SK is the secret key.

- Enc: an encryption algorithm that takes as input a public key PK and the a plaintext message $m \in \mathcal{P}$ and outputs a ciphertext $c \in \mathcal{C}$.

$$c = \text{Enc}_{PK}(m)$$

- Dec: a decryption algorithm that takes as input a secret key SK and a ciphertext c and outputs a plaintext m.

$$m = \text{Dec}_{SK}(c)$$

Digital Signature

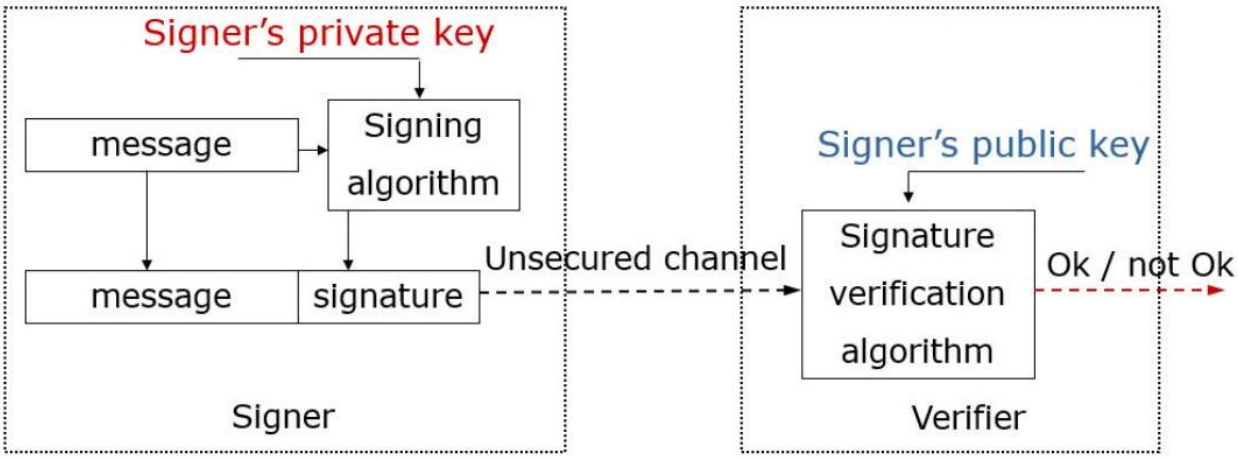

Figure 2: Digital Signature Schema

Properties:

- data integrity
- authenticity of the message
- non-repudiation

Definition

A signature scheme is a tuple of three algorithms DS = (Key, Sign, Ver) satisfying the following:

- Key: The key-generation algorithm Key takes as input a security parameter k and outputs a pair of keys (PK,SK). These are called the public key and the secret key, respectively.
- Sign: The signing algorithm Sign takes as input a secret key SK and a message $m \in \{0,1\}^*$. It outputs signature $s = \text{Sign}_{SK}(m)$.

- Ver: The deterministic verification algorithm Ver takes as input a public key PK, a message m, and a signature s. It outputs TRUE meaning valid or FALSE meaning invalid.

Hash Functions

By a hash function, we mean a map $H : \{0,1\}^* \to \{0,1\}^n, n \in \mathbb{N}$.

Thus, hash functions map arbitrarily long strings to strings of fixed length.

- Verifying data integrity: Cryptographic hash functions can be used to check whether a file has been changed. The hash value of the file is stored separately. The integrity of the file is checked by computing the hash value of the actual file and comparing it with the stored hash value. If the two hash values are the same, then the file is unchanged.
- hash value is called message digest
- **Avalanche effect**: If an input is changed slightly (for example, flipping a single bit), the output changes significantly (e.g., half the output bits flip)

Requirements

- Hash functions are never injective.

Definition A collision of H is a pair $(x, x') \in \{0,1\}^*$ for which $x \neq x'$ and $h(x) = h(x')$

Typically, a cryptographic hash function $H : X \to Y$ has three properties:

- **Preimage resistance**: Given $y \in Y$, it's computationally infeasible to find $x \in X$ such that $H(x) = y$.
- **Second preimage resistance** (weak collision resistant): Given x, it's computationally infeasible to find another x' such that $x \neq x'$ and $H(x) = H(x')$.
- **Collision resistance** (strong collision resistant): It's computationally infeasible to find any two distinct values $x, x' \in X$ such that $H(x) = H(x')$

RSA Algorithm

Asymmetric encryption scheme: $AE =$ (Key, Enc, Dec)

- Key:
  - Randomly choose two large primes: p, q. (Using Miller-Rabin for example)
  - Calculate RSA modulus: $n = p \cdot q$.
  - Calculate Euler totient: $\phi(n) = (p-1)(q-1)$.
  - Randomly choose an integer $e : 1 < e < \phi(n)$ and $\text{GCD}(e, \phi(n)) = 1$. (e is the encryption exponent)

  - Calculate $d : 1 < d < \phi(n)$ and $ed \equiv 1 (\mathrm{mod} \phi(n))$. (d is the decryption exponent) $PK = (n, e), SK = d$ and $\phi(n), p, q$ are secret parameters $\mathcal{P} = \mathcal{C} = \mathbb{Z}_n$
- $\mathrm{Enc}_{PK}(m) = m^e (\mathrm{mod} n) \forall m \in \mathcal{P}$ and for a $PK = (n, e)$.
- $\mathrm{Dec}_{SK}(c) = c^d (\mathrm{mod} n) \forall c \in \mathcal{C}$ and for a $SK = d$.

AES (Advanced Encryption Standard)

The features of AES are as follows:

- Symmetric key symmetric block cipher
- 128-bit data, 128/192/256-bit keys
- Stronger and faster than Triple-DES
- Provide full specification and design details

Interestingly, AES performs all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Each of these rounds uses a different 128-bit round key, which is calculated from the original AES key.

Each round comprises of four sub-processes

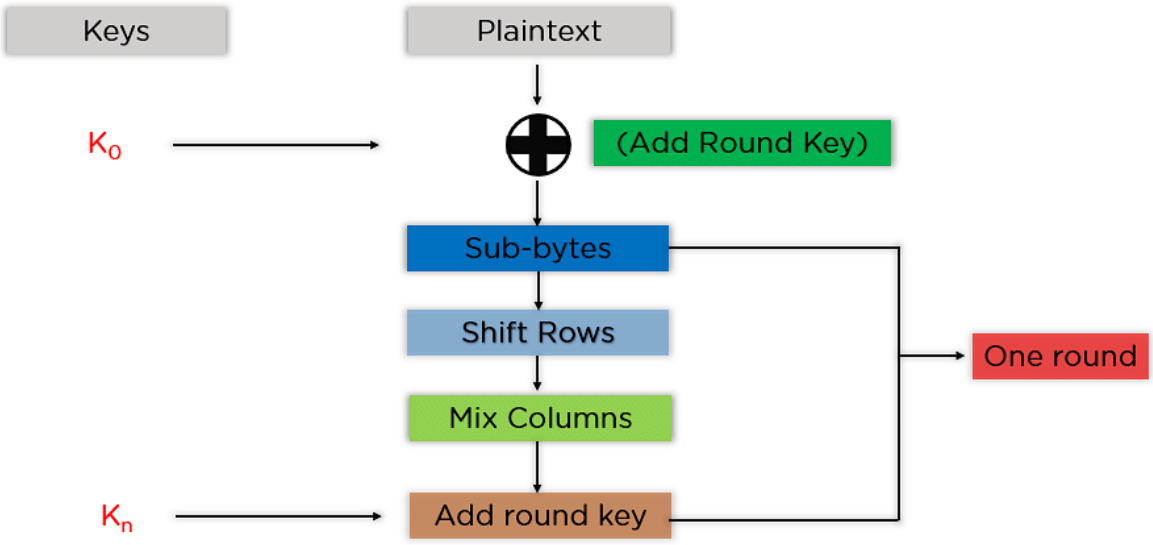

Figure 3: Process of AES

Byte Substitution (SubBytes) — The 16 input bytes are substituted by looking up a fixed table (S-box) given in design. The result is in a matrix of four rows and four columns.

Shift Rows — Each of the four rows of the matrix is shifted to the left. Any entries that 'fall off' are re-inserted on the right side of row. Shift is carried out as follows:

- First row is not shifted.
- Second row is shifted one (byte) position to the left.
- Third row is shifted two positions to the left.
- Fourth row is shifted three positions to the left.
- The result is a new matrix consisting of the same 16 bytes but shifted with respect to each other.

Mix Columns — Each column of four bytes is now transformed using a special mathematical function. This function takes as input the four bytes of one column and outputs four completely new bytes, which replace the original column. The result is another new matrix consisting of 16 new bytes. It should be noted that this step is not performed in the last round.

Add Round Key — The 16 bytes of the matrix are now considered as 128 bits and are XORed to the 128 bits of the round key. If this is the last round then the output is the ciphertext. Otherwise, the resulting 128 bits are interpreted as 16 bytes and we begin another similar round.

The process of decryption of an AES ciphertext is similar to the encryption process in the reverse order. Each round consists of the four processes conducted in the reverse order.