Armaan Shamsaasef

Encryption Technologies

Encryption Technologies - SY0-701 1.4

Purpose: To protect data by converting it into an unreadable format (ciphertext) that can only be converted back (decrypted) with the correct key.

1. Symmetric Encryption

- Concept: Uses a single, shared secret key for both encryption and decryption.
- Analogy: A single key that locks and unlocks the same door.
- Strengths: Very fast and efficient for bulk data encryption.
- Weakness: The Key Distribution Problem - how do you securely share the single secret key with the intended party?
- Common Algorithms:
  - AES (Advanced Encryption Standard): The modern global standard. Very strong and efficient.
  - DES (Data Encryption Standard): Now considered obsolete and insecure due to its short 56-bit key.
  - 3DES (Triple DES): A slower, more secure successor to DES, but now also deprecated.
  - RC4: A stream cipher, now considered insecure and should not be used.
  - Blowfish / Twofish: Fast and flexible algorithms, with Twofish being a strong AES finalist.

2. Asymmetric Encryption

- Concept: Uses a mathematically related key pair: a Public Key and a Private Key.
  - Public Key: Is shared publicly and is used to encrypt data.
  - Private Key: Is kept completely secret and is used to decrypt data.
- Analogy: A locked mailbox with a public slot. Anyone can drop a letter in (encrypt with the public key), but only the person with the private key can open the box to read it (decrypt).
- Strengths: Solves the key distribution problem. No need to pre-share a secret.

- Weakness: Computationally slow. Not suitable for encrypting large amounts of data.
- Primary Use Cases:
  - Key Exchange: (e.g., Diffie-Hellman) Securely establishing a symmetric session key.
  - Digital Signatures: Proving the authenticity and integrity of a message.
- Common Algorithms:
  - RSA (Rivest, Shamir, Adleman): One of the first and most common. Used for encryption and digital signatures.
  - Elliptic Curve Cryptography (ECC): Provides similar security to RSA but with much smaller keys, making it faster and more efficient.
  - PGP / GPG: Standards that use asymmetric encryption to secure email and files.

3. Lightweight Cryptography

- Concept: Encryption algorithms designed for devices with limited processing power, memory, or battery life.
- Use Cases: Internet of Things (IoT) devices, embedded systems, sensors, RFID tags.
- Example Algorithm: A common standard is ASCON, which was selected by NIST in 2023 for this purpose.

4. Homomorphic Encryption

- Concept: Allows computations to be performed on encrypted data without decrypting it first. The result of the computation, when decrypted, matches the result as if the operations had been performed on the raw, plaintext data.
- Video Example:
  1. You have the number `7` and encrypt it to get `XYZ`.
  2. You send `XYZ` to a cloud server.
  3. The cloud server performs an operation (e.g., adds `3` to `XYZ`) without knowing what the original data is.
  4. You get back the encrypted result, decrypt it, and get the correct answer: `10`.
- Use Case: Enables secure outsourcing of data processing to third-party cloud services without sacrificing privacy.

Summary

- Symmetric: One shared key. Fast for data. Problem: Key distribution.
- Asymmetric: Public/Private key pair. Slow, but solves key distribution. Used for key exchange and digital signatures.
- Lightweight: For low-power devices (IoT).
- Homomorphic: Compute on encrypted data; a cutting-edge technology for privacy in the cloud.