POLITECNICO DI TORINO

Fundamentals of Information Systems Security

Student:

Gianmarco Michelini

Academic Year 2024/2025

All notes are derived from oral presentations and written papers.

Contents

List of Figures

List of Tables

Chapter 1

Cryptographic Techniques

[1]. Cryptography is the practice of securing communication and data by transforming it into a format that is unreadable to unauthorized users. It uses mathematical algorithms to encrypt (scramble) and decrypt (unscramble) information, ensuring confidentiality, integrity, authentication, and non-repudiation in digital communications.

The message in its original, unencrypted form is called plaintext (or cleartext), referred to as P. On the other hand, the message after being encrypted is called ciphertext, referred to as C.

1.1 Kerchoffs' Principle

> If the keys are kept secret, managed securely, and are of sufficient length, the system remains secure even if the encryption and decryption algorithms are publicly known. This is because, without access to the keys, an attacker cannot decrypt the data. In fact, making the algorithms public allows the cryptographic community to rigorously test and analyze them for potential weaknesses, improving the overall security of the system.

1.2 Symmetric Cryptography

Only one key, shared between the sender and the receiver, is used for both encryption and decryption.

Encrypt Equation

$$\begin{aligned} C &= \text{enc}(K, P) \\ &= \{P\}_K \end{aligned}$$

Decrypt Equation

$$\begin{aligned} P &= \text{dec}(K, C) \\ &= \text{enc}^{-1}(K, C) \end{aligned}$$

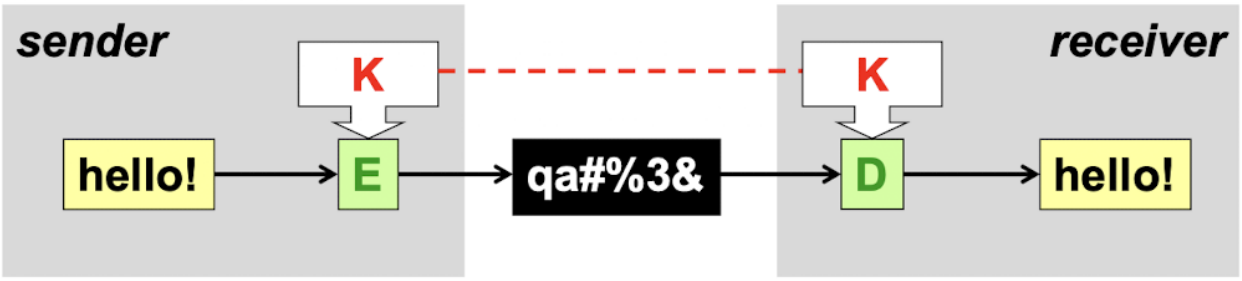

Figure 1.1: Example of symmetric encryption

1.3 Symmetric Block Encryption Algorithms

Block encryption refers to ciphers that process fixed-size blocks of data (e.g., 128 bits) at a time.

name	key (bit)	block (bit)	notes
DES	56	64	obsolete
3-DES (2 keys)	112	64	56...112-bit strength
3-DES (3 keys)	168	64	112-bit strength
IDEA	128	64	famous for PGP
RC2	8-1024	64	usually 64-bit key
Blowfish	32-448	64	usually 128-bit key
CAST	40-128	64	usually 128-bit key
RC5	0-2048	1-256	optimal when B=2W
AES	128-192-256	128	state-of-the-art

Figure 1.2: Famous Symmetric Block Encryption Algorithms

1.3.1 DES

(Data Encryption Standard - Obsolete because of short key length)

Is a symmetric-key block cipher that was widely used for data encryption in the past. It was developed in the 1970s by IBM.

The key features are:

- Block cipher: DES operates on 64-bit (8 bytes) blocks of data, meaning it encrypts 64 bits of plaintext at a time.
- Key length: DES uses a 56-bit (7 bytes) key.

- Efficient in hardware: Requires only XOR, shift and permutation.

1.3.2 Triple DES

(3DES or TDES)

Is a symmetric-key block cipher that was developed to enhance the security of the original DES.

The key features are:

- Block cipher: like DES (8 bytes).
- Key Length: 3DES can use either:
  - **Two keys** (2-key 3DES): The first and third keys are the same, resulting in a 112-bit effective key length.
  - **Three keys** (3-key 3DES): All three keys are distinct, resulting in a 168-bit effective key length.
- Triple encryption: 3DES applies the DES algorithm three times to each data block, using either two (2-key 3DES) or three (3-key 3DES) distinct keys.
- Is typically applied in the Encrypt-Decrypt-Encrypt (EDE) mode to achieve compatibility with DES when.

EDE Mode

In this mode, the data is first encrypted using the first key, then decrypted using the second key, and finally encrypted again using the third key (if a third key is present). This sequence of operations helps mitigate the vulnerabilities of DES by applying encryption multiple times.

Meet-in-the-Middle attack

Double application of encryption algorithms is subject to a known plaintext attack named meet-in-the-middle, which allows decrypting data with at most 2^{N+1} attempts (the key is N-bit long). For more details, refer to Appendix A.

1.4 Application of Block Algorithms

We should answer a question:

Is the size of the encrypted data smaller or larger than the algorithm's block size?

If the answer is:

- Size of the encrypted data > algorithm's block size: Use Electronic Code Book (ECB) or Cipher Block Chaining (CBC).
- Otherwise: Padding, Cipher FeedBack (CFB), Output FeedBack (OFB) or Counter Mode (CTR).

1.4.1 Electronic Code Book

(ECB)

Simple mode of operation for block ciphers. In this mode, the plaintext is divided into fixed-size blocks, and each block is encrypted independently using the same key.

$$C_i = enc\ (K, P_i)$$

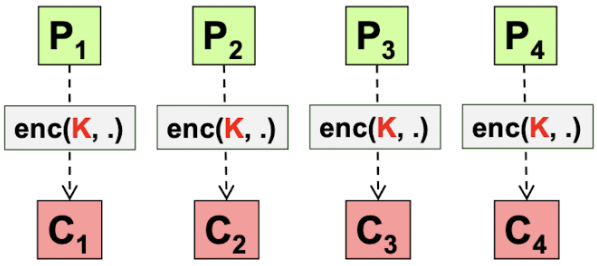

Figure 1.3: ECB encrypting.

$$P_j = \mathrm{enc}^{-1}(K, C_j)$$

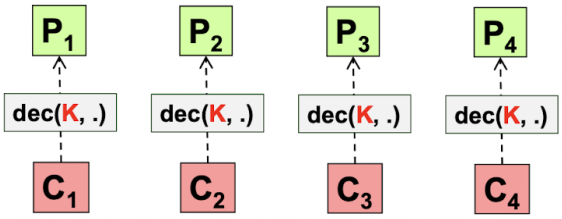

Figure 1.4: ECB decrypting.

Key features:

- Should not be used for long messages because swapping two blocks of ciphertext goes undetected, and identical plaintext blocks generate identical ciphertext blocks, making it vulnerable to known-plaintext attacks.
- The plaintext is divided into blocks of equal size (typically 64 or 128 bits, depending on the cipher)
- Each block is encrypted separately with the block cipher using the same key.
- The encrypted blocks are concatenated to produce the ciphertext.
- Identical plaintext blocks produce identical ciphertext blocks, which can lead to patterns in the ciphertext that may be exploited in cryptanalysis.
- If there is an error in one ciphertext block C_j , only the corresponding plaintext block P_j will be affected during decryption.

1.4.2 Cipher Block Chaining

(CBC)

Mode of operation for block ciphers that provides better security than Electronic Codebook (ECB). It involves chaining the encryption of each block with the previous block's ciphertext, which helps obscure patterns in the ciphertext.

$$C_i = enc\ (K, P_i \oplus C_i - 1)$$

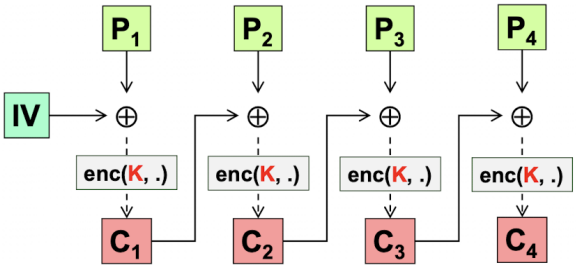

Figure 1.5: ECB encrypting.

$$P_j = \text{enc}^{-1}(K, C_j) \oplus C_{j-1}$$

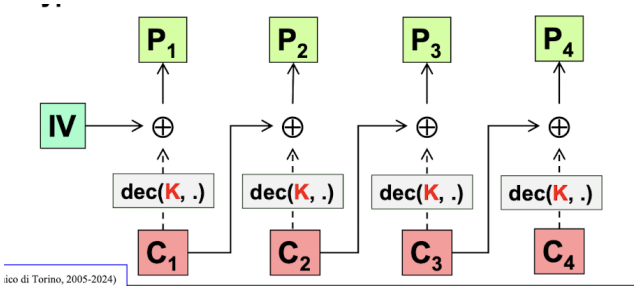

Figure 1.6: ECB decrypting.

Key features:

- Requires an Initialization Vector (IV=C_0).
- XOR between ciphertexts and plaintexts.
- If there is an error in a ciphertext block C_j it will affect the decryption of two blocks of plaintext:
  - The current block P_j will be corrupted because the error will be decrypted into some incorrect plaintext.
  - The next block P_{j+1} will also be corrupted because the error will propagate into the XOR operation with the next ciphertext block C_{j+1}.

1.4.3 Padding

(aligning, filling)

Padding modes are used in block cipher encryption schemes to handle plaintext that is not an exact multiple of the block size. Since block ciphers operate on fixed-size blocks, any plaintext that does not fit into the required block size needs to be padded.

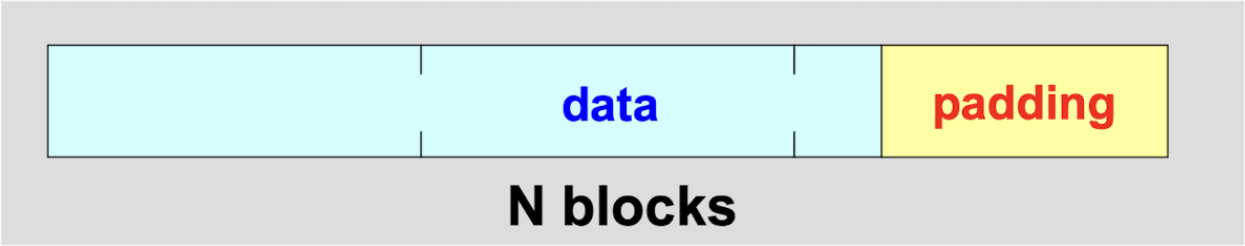

Figure 1.7: Padding Mode

Padding Techniques

- Original DES Padding: bit pattern that started with a 1 bit, followed by many 0 bits.
- One byte with value 128 (0x80) followed by null bytes.
- Last byte indicates padding length.
- Padding with explicit length:
  - Null bytes, with the last byte indicating the padding length.
  - Only null bytes. Schneier
  - Bytes with value equal to Length. SSL/TLS
  - Random bytes. SSH2
  - Sequential numbers starting from 0x01. IPSec/ESP
  - Each byte is the $Length - 1$.

Some Keynotes

- Minimal integrity control: If the key is incorrect or data is manipulated, the padding bytes will become incoherent.
- Typically applied to large data, on the last fragment.
- If the data length D is less than the block size B, ad-hoc techniques like CFB, OFB, or CTR are preferred instead of padding.
- Even if the plaintext is an exact multiple of the block size, padding is still required to avoid errors in the interpretation of the last block.

1.5 CTS

(Cipher Text Stealing)

Allows the use of block algorithms without padding.

- The last (partial) block is filled with bytes from the second-to-last block, which are removed from the second-to-last block (making it partial).

- After encryption, the positions of the last and second-to-last blocks are swapped.

This method is particularly useful when the size of the data cannot be increased after encryption, such as in storage encryption.

However, the computation time increases slightly.

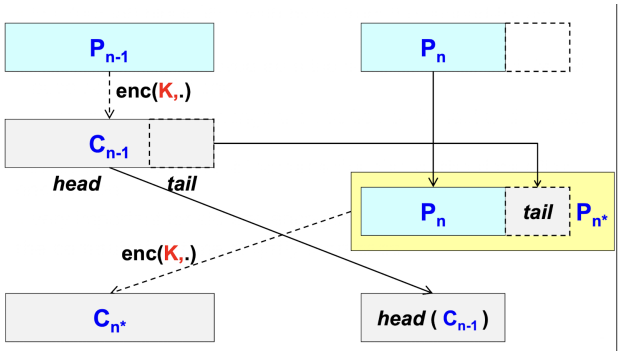

Figure 1.8: CTS with ECB

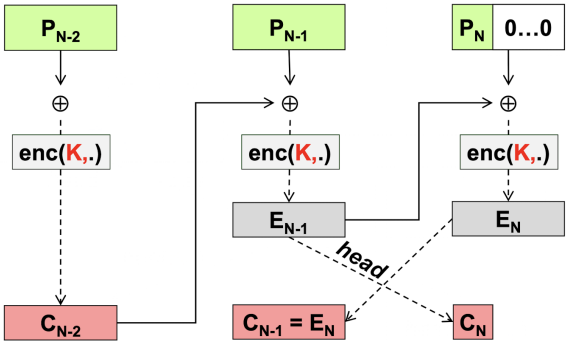

Figure 1.9: CTS with CBC

Bibliography

[1] Antonio Lioy. Cryptographic techniques for cybersecurity, 2024.