Vietnam National University Ho Chi Minh City, University of Science
Department of Information Technology

Topic 5:

Modes of Operation & Padding Scheme

Assoc. Prof. Trần Minh Triết
PhD. Trương Toàn Thịnh

fit@hcmus

KHOA CÔNG NGHỆ THÔNG TIN
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

Contents

- Modes of Operation
- Padding Schemes

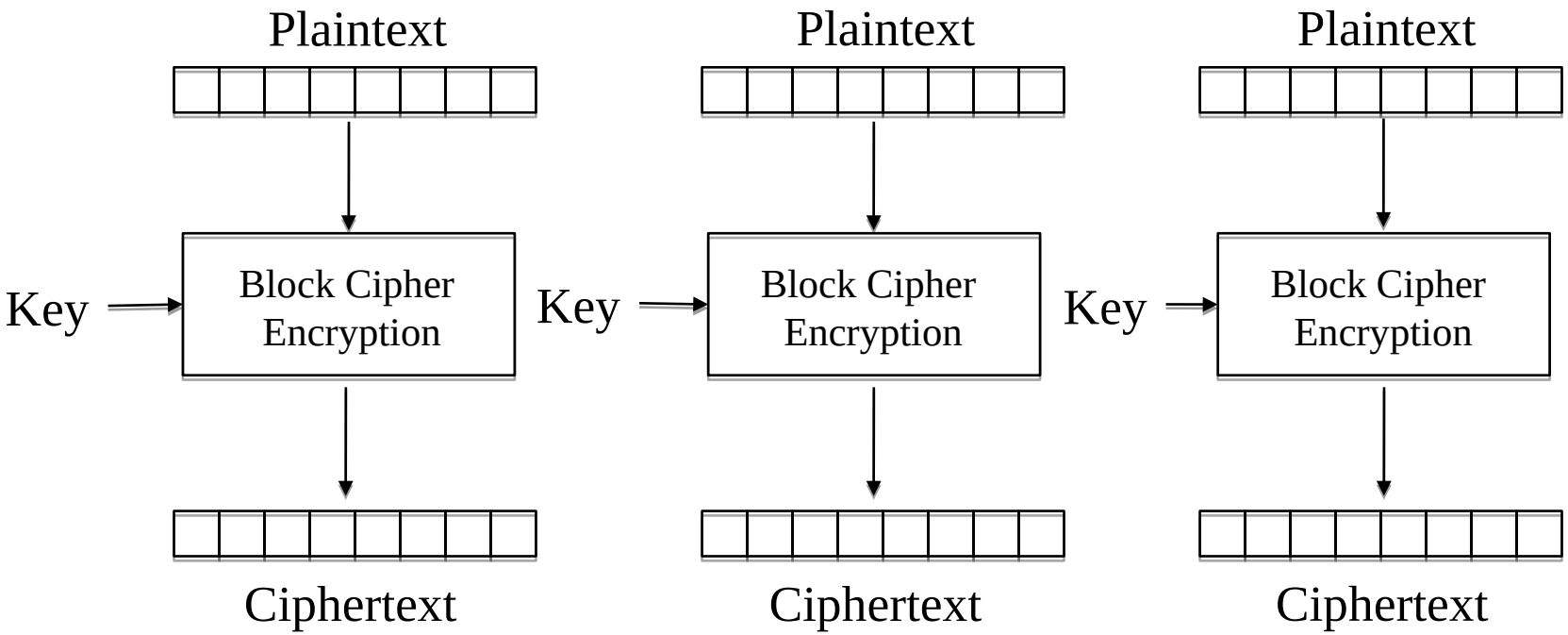
Modes of Operation

- In cryptography, data is often divided into chunks of fixed size (e.g., 64 or 128 bits).
- To encrypt the long messages (split into multiple blocks), **modes of operation** can be used.
- Some modes of operation (ECB, CBC, OFB, CFB) provide **confidentiality**, but do not ensure **message integrity**
- Some modes of operation (**CCM**, **EAX** and **OCB**) ensure **confidentiality** and **message integrity**.
- Some modes of operation are designed to encrypt sector on disc:
  - Tweakable narrow-block encryption –LRW
  - Wide-block encryption -CMC and EME

Electronic codebook (ECB)

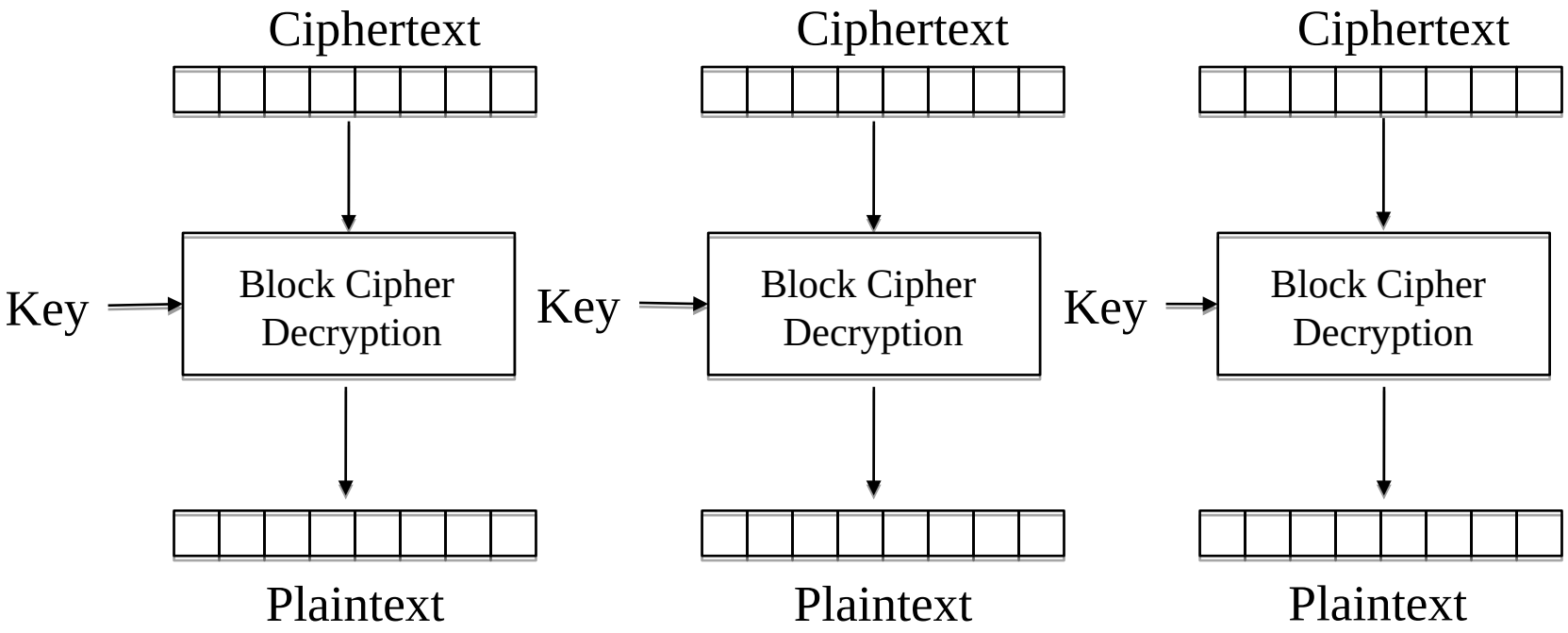
- Simple mode of operation is **electronic codebook** (ECB)
- The message to be encrypted is divided into segments, each segment is encrypted independently.
- Limitation: blocks with the same content, after encryption, also form identical result blocks → do not hide data pattern.
- The use of ECB in cryptographic protocols is not recommended

Electronic codebook (ECB)



Electronic Codebook (ECB) mode encryption

Electronic codebook (ECB)

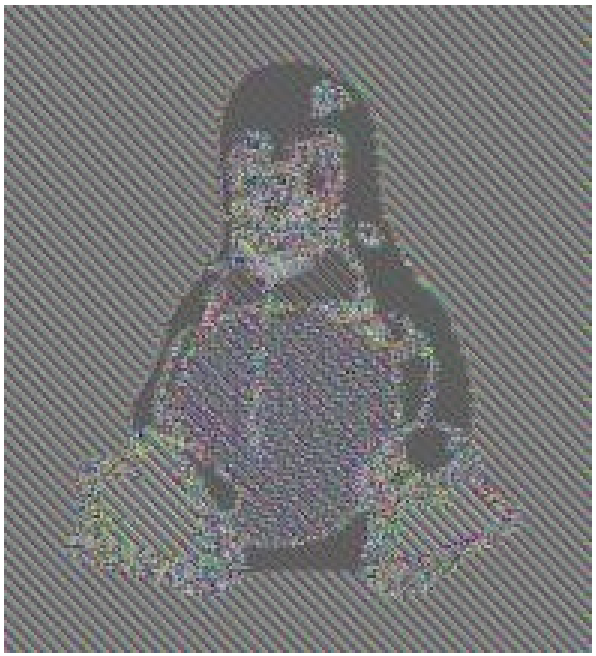


Electronic Codebook (ECB) mode decryption

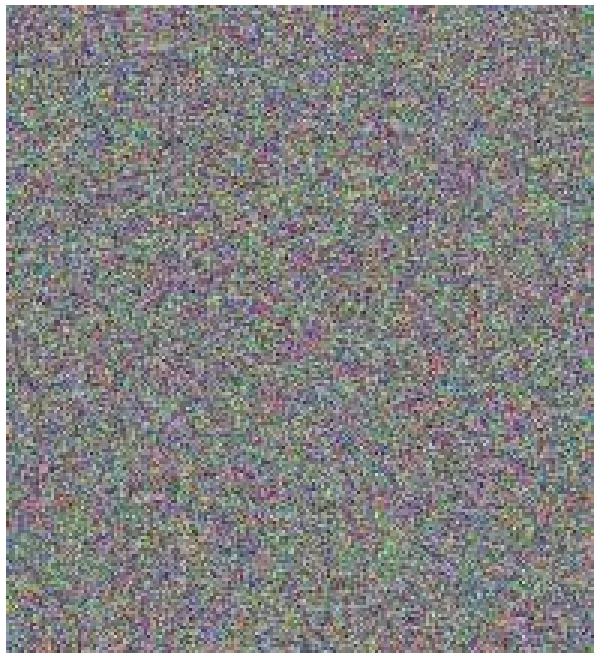
Electronic codebook (ECB)

- ECB can make the protocol less secure to protect information integrity (for example of replay attacks)

Original image



Encrypt with ECB



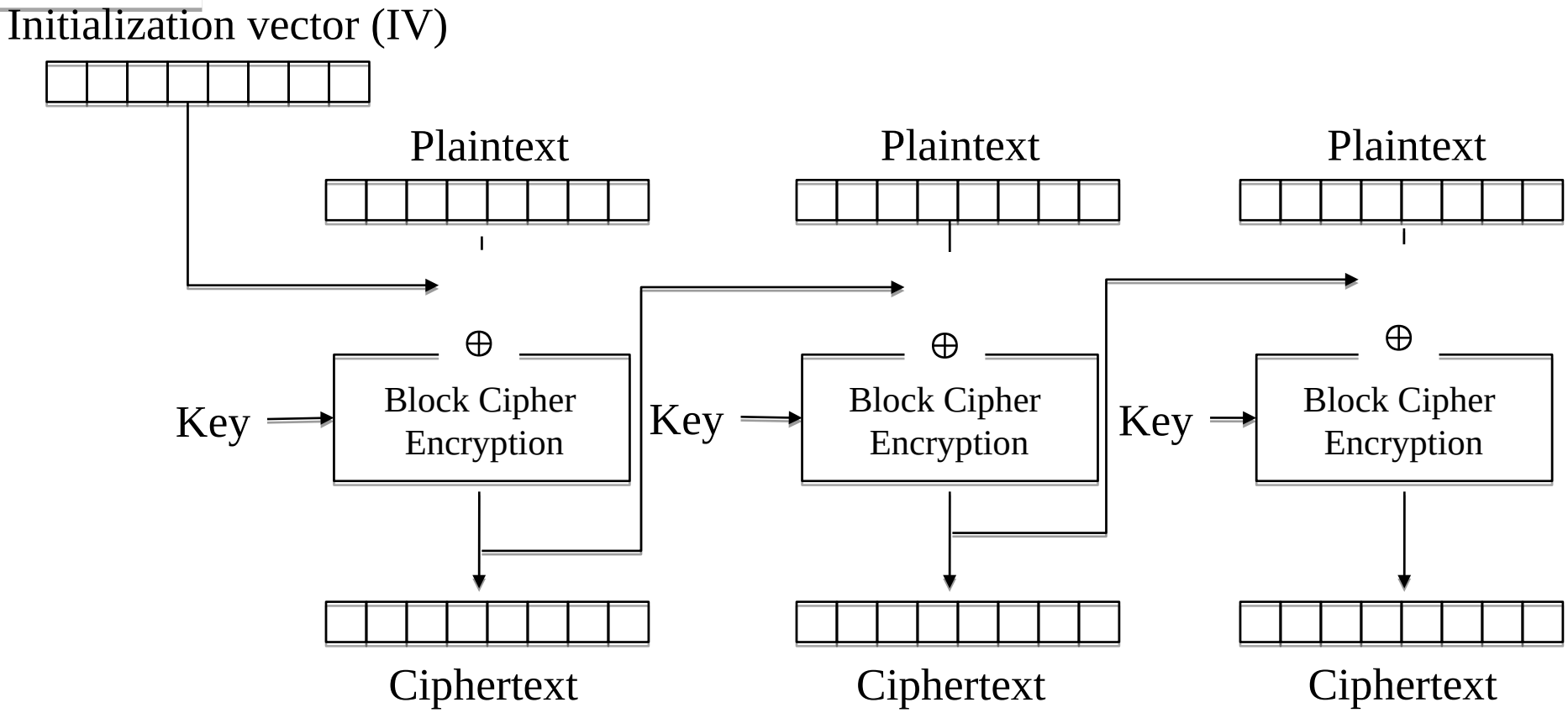
Encrypt with others

Cipher-block chaining (CBC)

- In **cipher-block chaining** (CBC):
  - Each plaintext block is XORed with the ciphertext block before being encrypted.
  - Thus, each ciphertext block depends on all the plaintext blocks that appear from the beginning up to that point
  - To ensure the uniqueness of each encrypted message, we use an additional initialization vector

Cipher-block chaining (CBC)

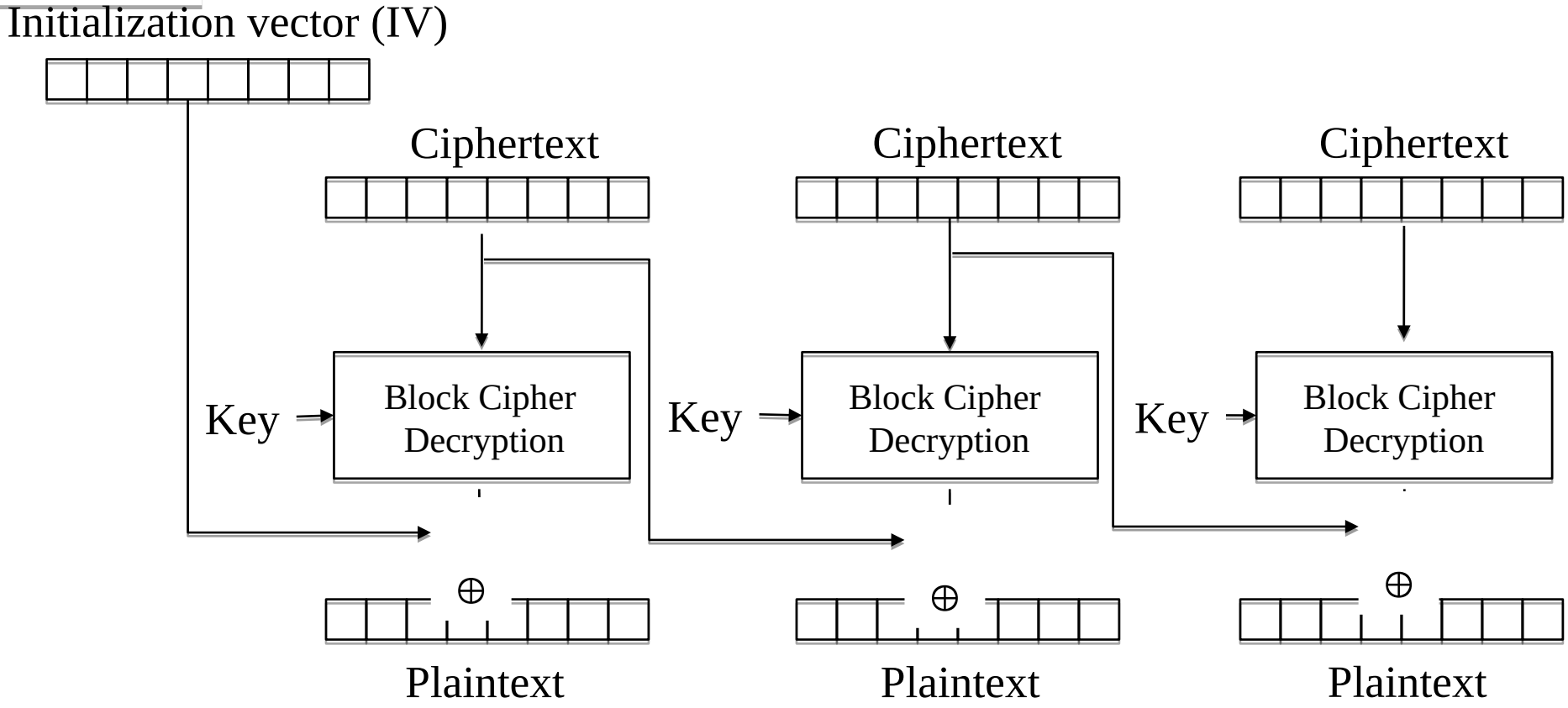
$$C_0 = IV$$
$$C_i = E_K(P_i \oplus C_{i-1})$$



Cipher Block Chaining (CBC) mode encryption

Cipher-block chaining (CBC)

$$C_0 = \mathbf{IV}$$
$$P_i = D_K(C_i) \oplus C_{i-1}$$



Cipher Block Chaining (CBC) mode decryption

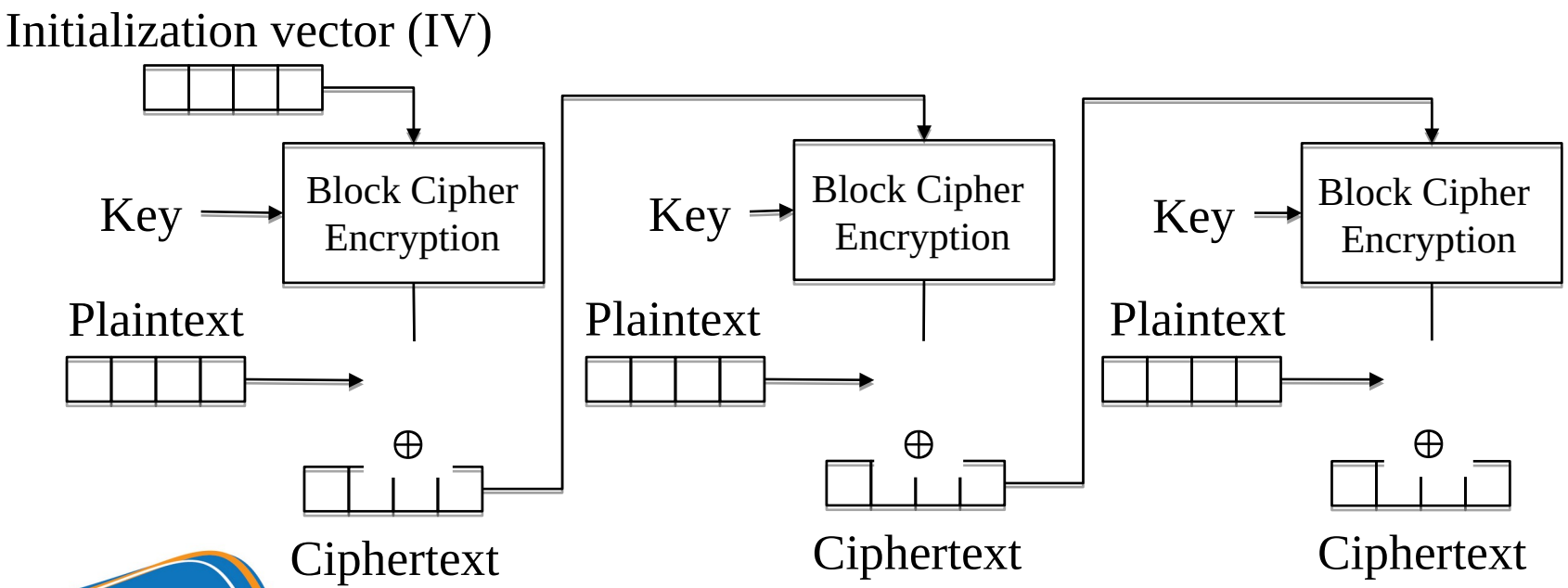
Propagating cipher-block chaining (PCBC)

- CBC is the most commonly-used type.
- Limitation: sequential processing, cannot be parallelized: counter mode solution can be chosen for parallel processing
- **Propagating cipher-block chaining** is designed to allow the influence is more pervasive in CBC.
  - P_0 = **IV,** C_0 = **0,** $C_i = E_K (P_i \oplus P_{i-1} \oplus C_{i-1})$
  - P_0 = **IV,** C_0 = **0,** $P_i = D_K (C_i) \oplus P_{i-1} \oplus C_{i-1}$
- PCBC commonly used in Kerberos and WASTE (besides, it's less common!)

Cipher feedback (CFB)

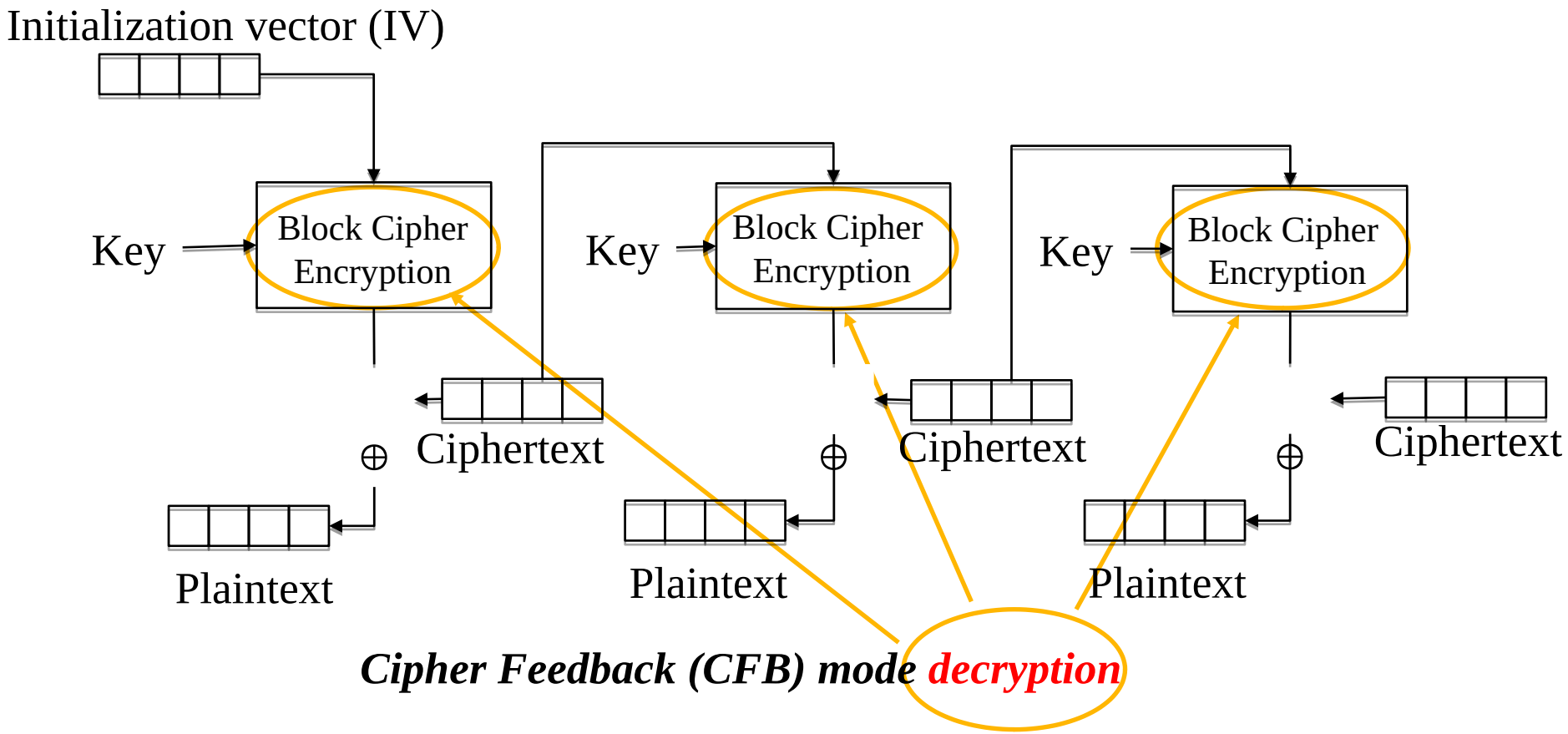
$$C_0 = \mathbf{IV}$$
$$C_i = P_i \oplus E_K(C_{i-1})$$

- Properties:
  - Plaintext is NOT encrypted by the algorithm in question
  - Plaintext is encrypted by XORing a string generated by the encryption algorithm.
  - Turn block-cipher into stream cipher



Cipher Feedback (CFB) mode encryption

Cipher feedback (CFB)

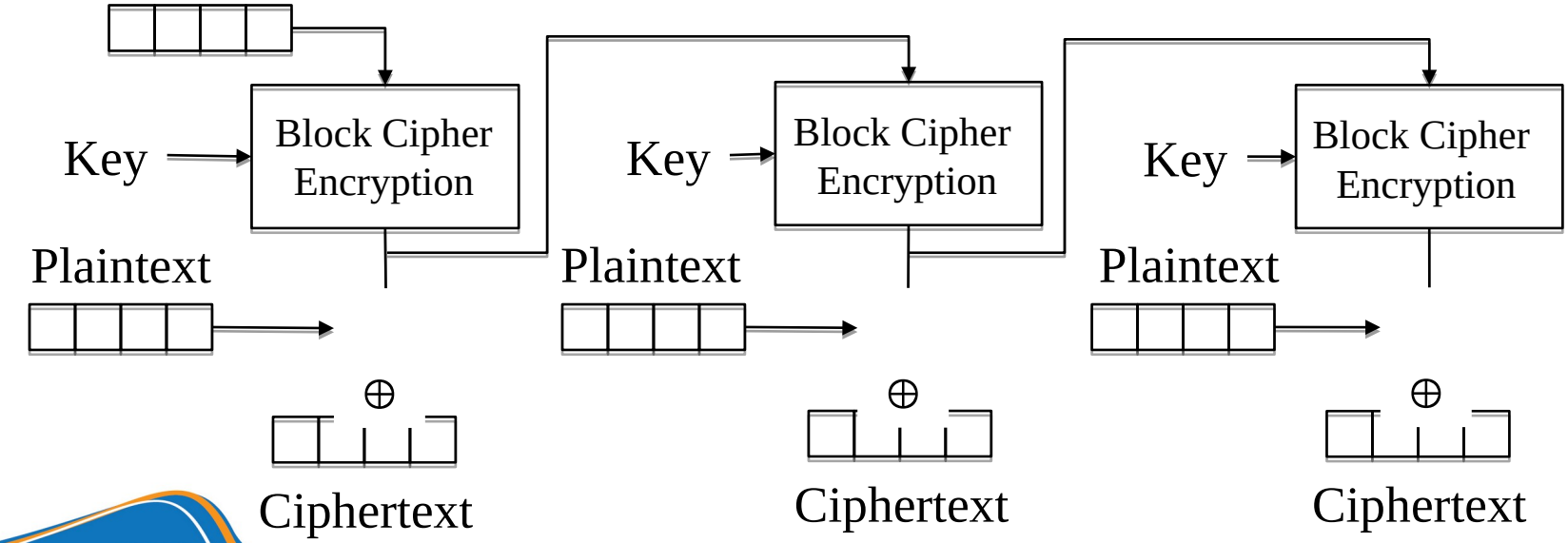


Output feedback (OFB)

$$O_0 = \mathbf{IV}$$
$$O_i = E_K(O_{i-1})$$
$$C_i = P_i \oplus O_i$$

- Properties:
  - Plaintext is NOT encrypted by the algorithm in question
  - Plaintext is encrypted by XORing a string generated by the encryption algorithm.
  - Turn block-cipher into stream cipher

Initialization vector (IV)



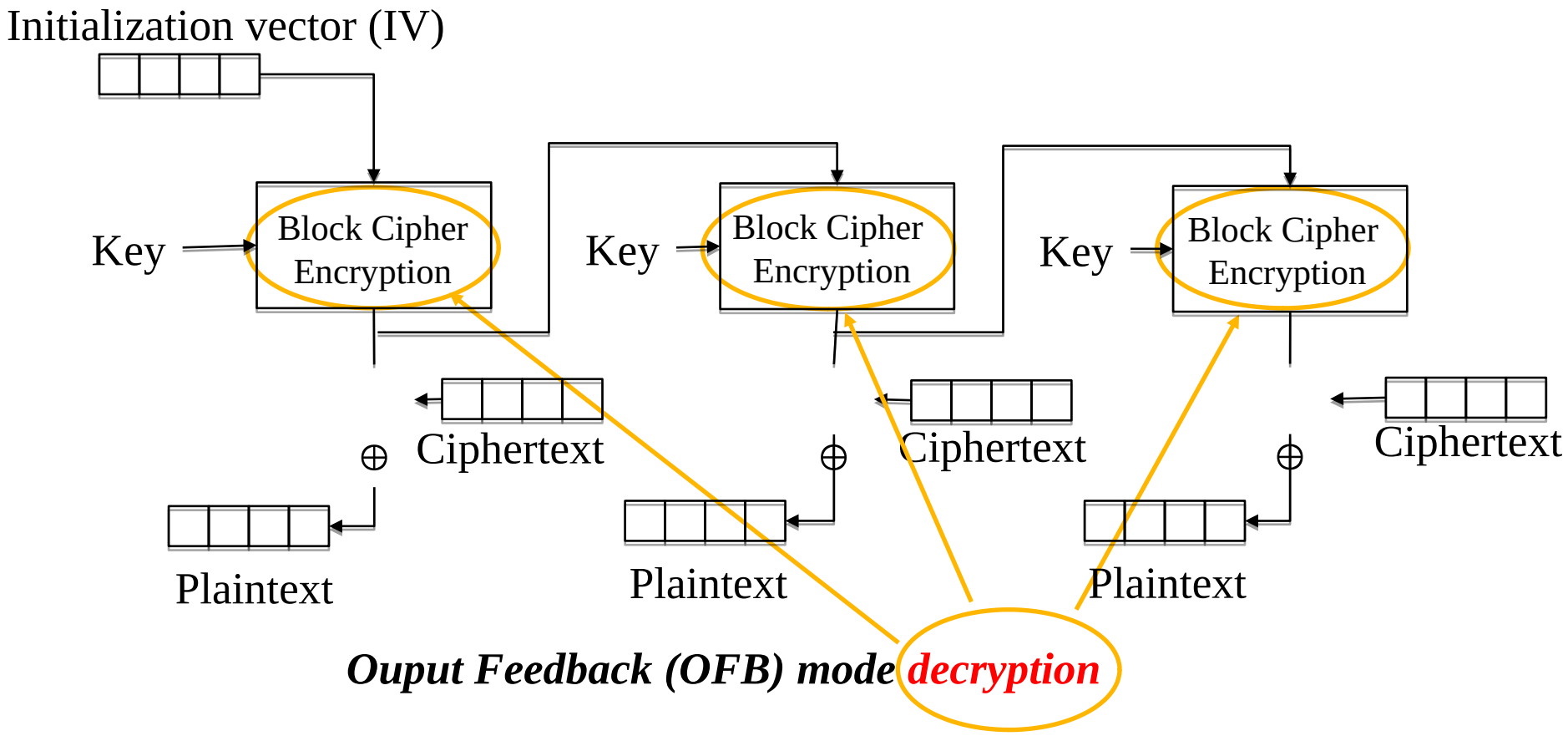
Output Feedback (OFB) mode encryption

Output feedback (OFB)

$O_0 = \mathbf{IV}$

$O_i = E_K(O_{i-1})$

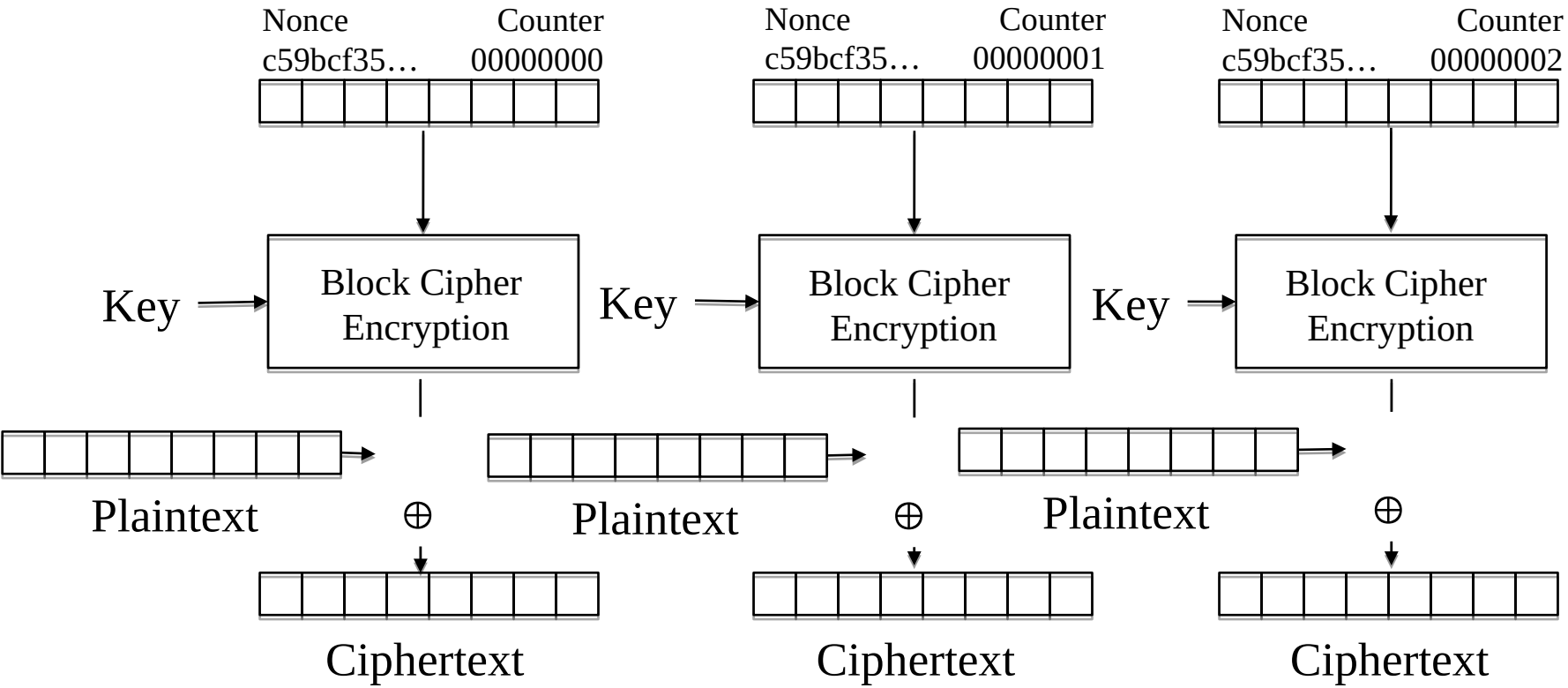
$P_i = C_i \oplus O_i$



Counter (CTR)

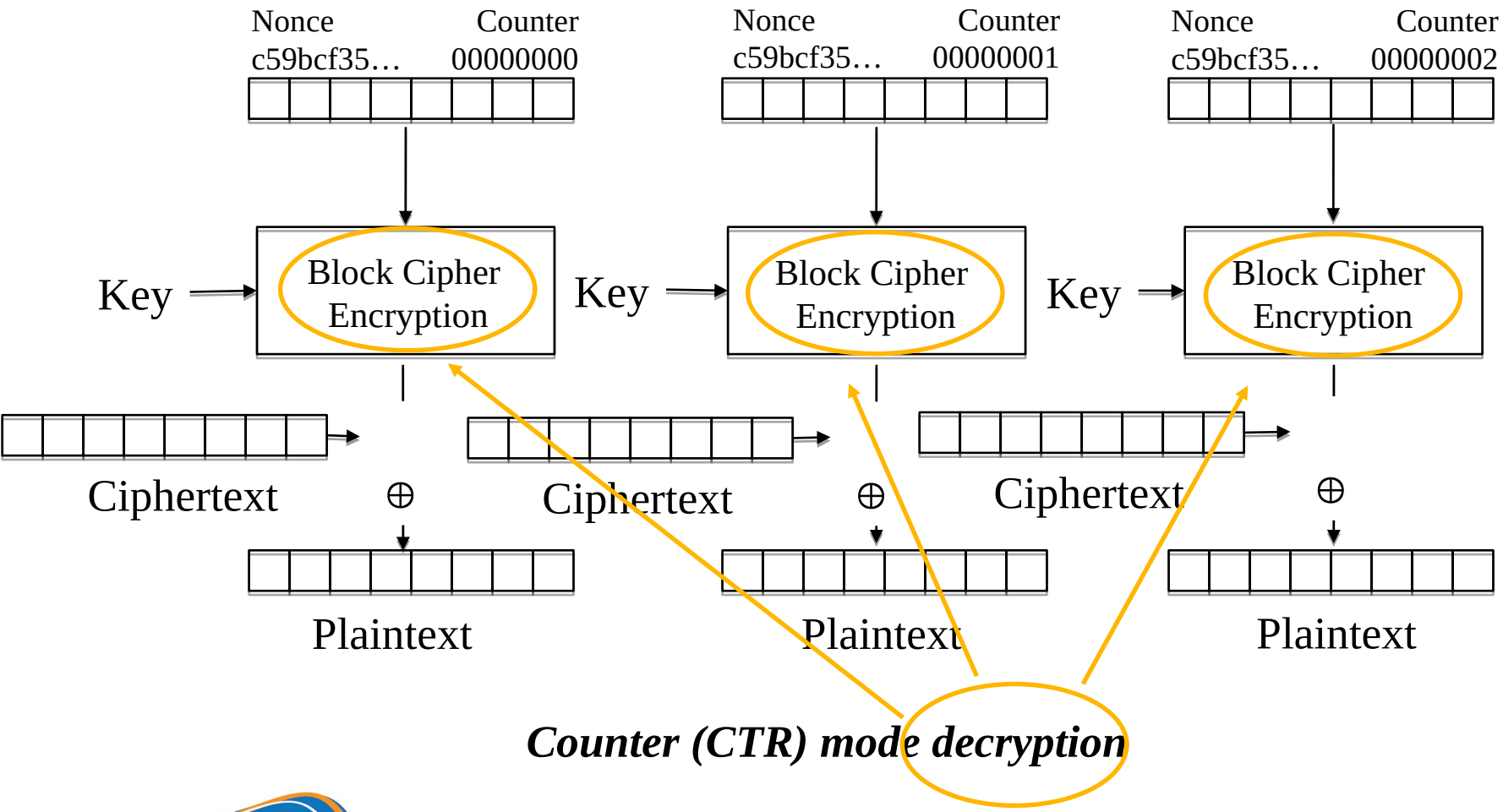
- CTR is called Segmented Integer Counter (SIC)
- Similar to OFB, Counter converts block cipher to stream cipher.
  - Create next block keystream by encrypting next value of "counter".
- Counter can be any function generating a string of distinct numbers in a long-enough duration.
- CTR has properties similar to OFC,
- CTR allows to randomly decrypt any cipher-text block
- Note: Role of nonce is the same as initialization vector (*IV*)
- *IV*/nonce and value counter can be joined, added or XORed to create a unique string of bits corresponding to each value counter

Counter (CTR)



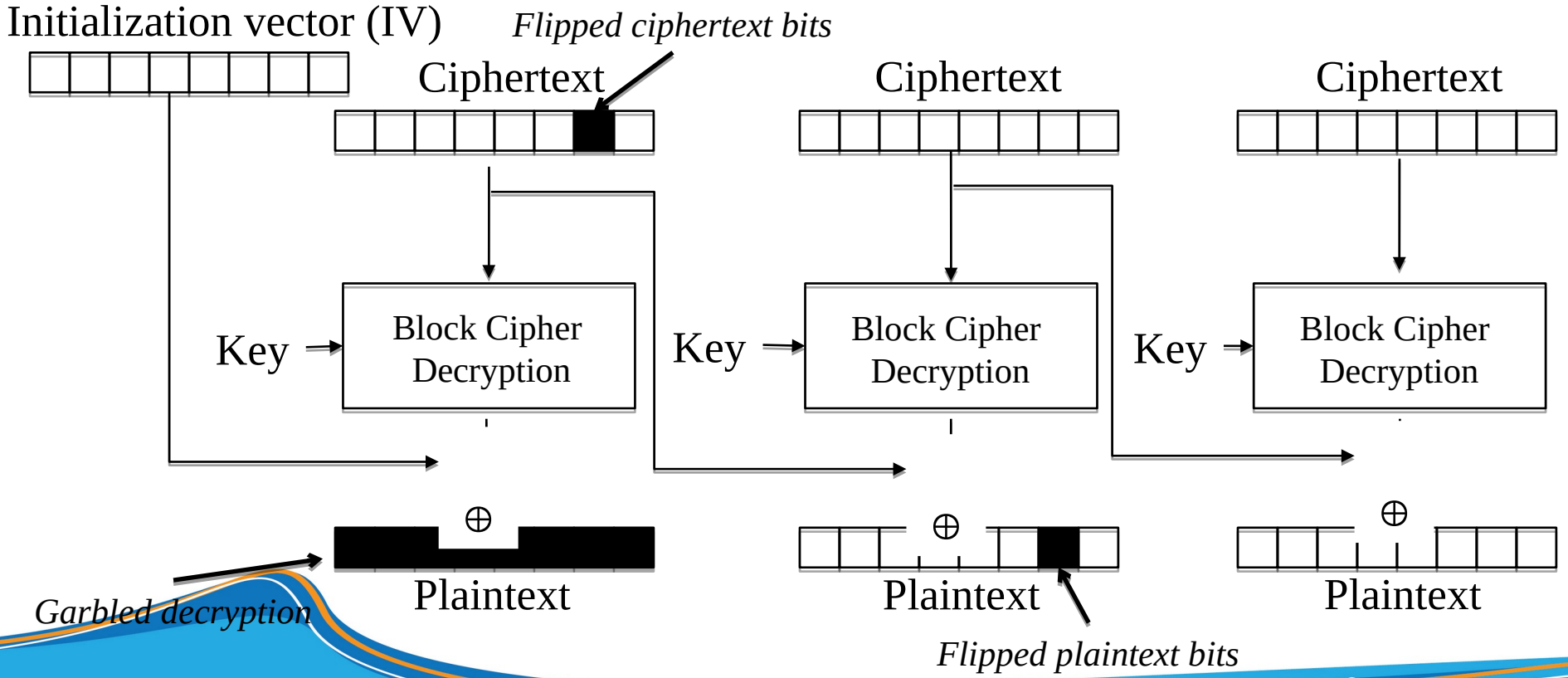
Counter (CTR) mode encryption

Counter (CTR)


Nonce
c59bcf35…
Counter
00000000
Nonce
c59bcf35…
Counter
00000001
Nonce
c59bcf35…
Counter
00000002
Key
Block Cipher Encryption
Key
Block Cipher Encryption
Key
Block Cipher Encryption
Ciphertext
⊕
Ciphertext
⊕
Ciphertext
⊕
Plaintext
Plaintext
Plaintext
Counter (CTR) mode decryption

Error propagation

- Limiting error-propagation: a criterion for evaluating Mode of operation
- Example: investigate error-propagation when decrypting information with CBC



Modification attack or transmission error for CBC

Initialization vector (IV)

- All modes of operation (except ECB) use *initialization vector - IV.*
- Role of *IV*:
  - Dummy block so that the processing of the first block is not different from the processing of successive blocks
  - Increase the randomness of the encryption process.
  - No need to keep it a secret
  - Be sure to limit the reuse of the same *IV* value with the same key.
- With CBC and CFB, reusing *IV* values leaks information.
- With OFB and CTR, reusing the *IV* completely breaks the security of the system
- *IV* in CFB must be randomly generated and kept secret until the contents of the first plaintext block are ready for encryption

Padding schemes

- **Padding Scheme:** additional information so that the data block is the right size for encryption
- Requirements:
  - Data block after addition has a size suitable for encryption
  - Can easily recover the exact data after decryption (exactly cut off the extra data)
- Basic methods:
  - **Bit Padding: see RFC1321** (http://www.faqs.org/rfcs/rfc1321.html)
  - **Byte Padding: see RFC1319** (http://www.faqs.org/rfcs/rfc1319.html)

Padding schemes

- **Bit Padding:**
  - "Standard" data block-size: n bits
  - Original data block M has size of m bits ($m \leq n$)
  - Data block after padding

m bits 1 bit (L – 1) bits

$L = n - (m \bmod n)$

What if $m = n$?

Padding schemes

- **Byte Padding (PKCS5):**
  - "Standard" data block-size: n bytes ($n < 256$)
  - Original data block M has size of m bytes ($m \leq n$)
  - Data block after padding

m bytes

L bytes

$$L = n - (m \bmod n)$$

What if $m = n$?

Find out more

- OAEP (**O**ptimal **A**symmetric **E**ncryption **P**adding)
- CCM (**C**ounter with **C**BC-**M**AC mode)
- EAX (**E**ncryption-then-**A**uthentication-then-Translate mode)
- OCB (**O**ffset **c**ode**b**ook mode)