Elements of Cryptography. Symmetric Encryption.

NSE, sections 2.1-2.2
WSPC, chapter 3

Cryptography

- Cryptography is a collection of mathematical techniques for protecting information;

- Most important cryptographic technique is
  encryption/decryption

Cryptography for information protection

Level	What to protect	Method
3	Existence of message	Steganography
2	Metadata of message	Privacy-enhancing technologies
1	Content of message	Encryption
0	Nothing	None

Table by I.A. Goldberg

Encryption is used

- Directly at the level 1
- As an important ingredient at the levels 2 and 3

Two categories of encryption algorithms

- Symmetric encryption (or symmetric key encryption):
  - to encrypt and decrypt a message the *same key* (a piece of information; sequence of bits) is used

- Asymmetric encryption (or asymmetric key encryption):
  - One key is used for encryption (usually publicly known, *public key*);
  - Another key is used for decryption (usually *private*, or *secret* key)

Symmetric (conventional) encryption

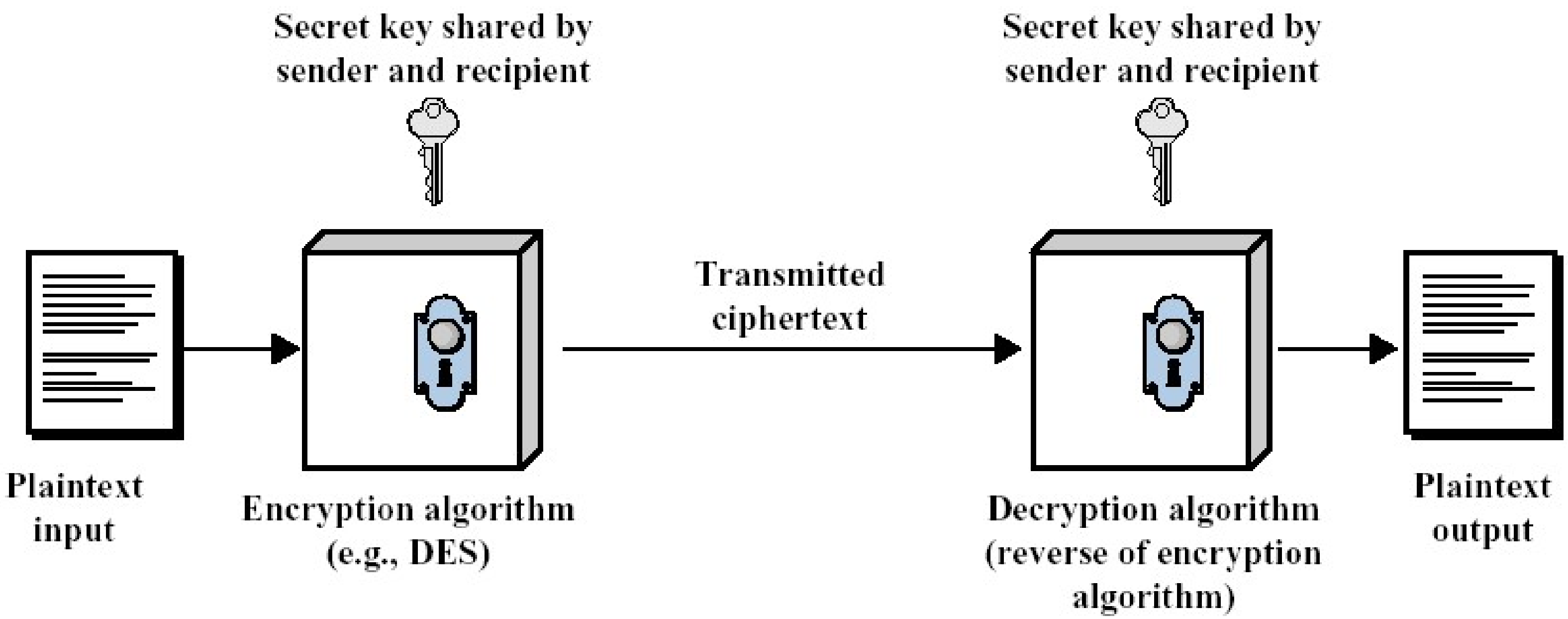


Figure 2.1 Simplified Model of Symmetric Encryption

Components of Symmetric Encryption

- Plaintext
- Encryption algorithm
- Secret key
- Ciphertext (encrypted text)
- Decryption algorithm

Security of symmetric encryption

- **Important principle:**
- security of symmetric encryption depends on
  - the secrecy of the key,
  - Not the secrecy of the algorithm

Why?
  - It is difficult to *invent* new algorithms and *keep* them in a secret;
  - Producing keys is much easier;

Requirements for symmetric encryption

- *Strong* encryption algorithm:
  - The adversary (opponent) should be unable to decrypt encrypted text, even if he/she knows several pairs

    (plaintext, encrypted plaintext)
- Sender and receiver must have obtained copies of the secret key in a secure way and *must keep the key secure*

Two more classifications of cryptosystems

- **Type of operations used**
  - Substitutions;
  - Transpositions;

- **The way in which plaintext is processed**
  - Block cipher: input block of elements (e.g. characters) is transformed to the output block at once;
  - Stream cipher: processes the input elements continuously, one element at a time.

Classics: substitutions

- Each element of the plaintext (bit, letter, group of bits) is mapped to another element
- **Example:**

A -> B
B -> C
C -> D
….
Z -> A

Plaintext **"Knowledge is power"**
is transformed into
"Lopxmfehf jt rpxfs"

Classics: transposition

- Elements of the plaintext are re-arranged.
- Example: “Knowledge is power”
-

K	n	o	w	l
e	d	g	e	
i	s		p	o
w	e		r	

Is transformed into
“Keiwndseog weprl o “

Two remarks

- Most modern algorithms include multiple stages of *interleaving* substitutions and transpositions;
- The encryption uses a *key* (unlike simple examples on the previous slides)

Cryptanalysis and computationally secure schemes

- **Cryptanalysis:** The process of attempting to discover the plaintext or key;
- Depends very much on the information available;

- An encryption scheme is *computationally secure* if
  - The cost of breaking the scheme exceeds the value of the encrypted information;
  - The time required to break the scheme is more than lifetime of the information;

Types of Attacks (Cryptanalysis)

Table 2.1 Types of Attacks on Encrypted Messages

Type of Attack	Known to Cryptanalyst
Ciphertext only	•Encryption algorithm •Ciphertext to be decoded
Known plaintext	•Encryption algorithm •Ciphertext to be decoded •One or more plaintext-ciphertext pairs formed with the secret key
Chosen plaintext	•Encryption algorithm •Ciphertext to be decoded •Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key
Chosen ciphertext	•Encryption algorithm •Ciphertext to be decoded •Purported ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key
Chosen text	•Encryption algorithm •Ciphertext to be decoded •Plaintext message chosen by cryptanalyst, together with its corresponding ciphertext generated with the secret key •Purported ciphertext chosen by cryptanalyst, together with its corresponding decrypted plaintext generated with the secret key

Brute-Force Approach in Cryptanalysis

- If nothing else helps and there is no weakness in the encryption algorithms, brute-force approach may be applied;
- Try every possible key until correct translation of the encrypted text into plaintext is obtained;
- **Possible issue**: how does cryptanalyst recognize correct plaintext? Imagine it has been compressed before encryption;
- **Main issue:** time !!!

Time required for brute-force search

Key Size (bits)	Number of Alternative Keys	Time required at 1 encryption/μs	Time required at 10^6 encryptions/μs
32	$2^{32} = 4.3 \times 10^9$	2^{31} μs = 35.8 minutes	2.15 milliseconds
56	$2^{56} = 7.2 \times 10^{16}$	2^{55} μs = 1142 years	10.01 hours
128	$2^{128} = 3.4 \times 10^{38}$	2^{127} μs = 5.4×10^{24} years	5.4×10^{18} years
168	$2^{168} = 3.7 \times 10^{50}$	2^{167} μs = 5.9×10^{36} years	5.9×10^{30} years
26 characters (permutation)	$26! = 4 \times 10^{26}$	2×10^{26} μs = 6.4×10^{12} years	6.4×10^6 years

Analysis vs cryptanalysis

- One of the main goals when creating new encryption/decryption algorithm is to make it as difficult as possible for *cryptanalysis* (difficult to break);
- One the other hand, making an algorithm easy to *analyse* could be beneficial, because then
  - Analysis of the algorithm can provide with a higher level of assurance;

Symmetric Encryption. Feistel cipher. DES and AES

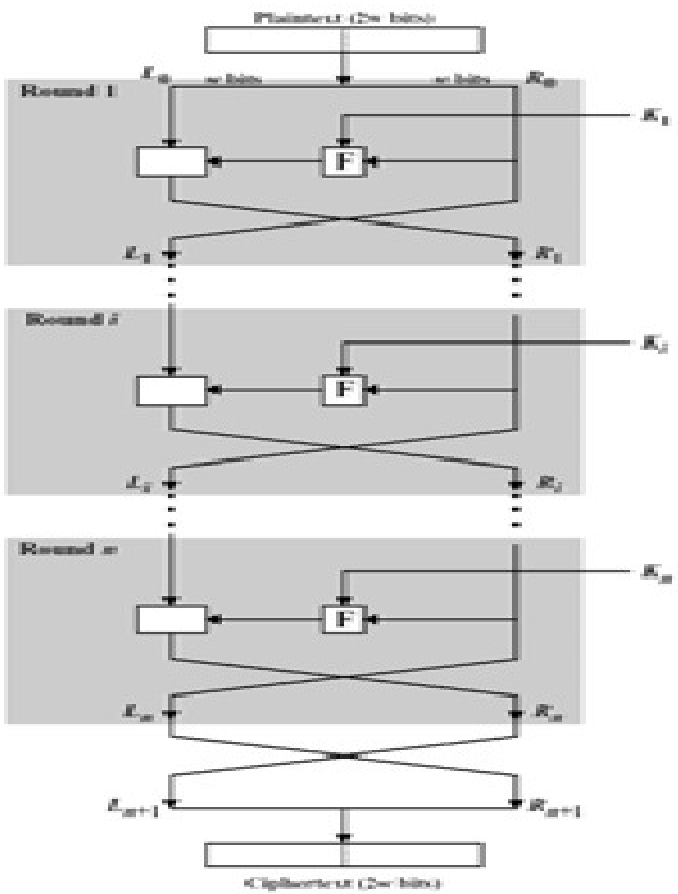
Block vs stream ciphers

- **The way in which plaintext is processed**
  - Block cipher: input block of elements (e.g. characters) is transformed to the output block at once;
  - Stream cipher: processes the input elements continuously, one element at a time.

Feistel cipher structure

- Most symmetric block encryption algorithms have a structure proposed by H.Feistel in 1973;
- The input is divided into the blocks of even numbers of elements;
- Then multiple stages of substitutions and transpositions is applied;
- Multiple keys (derived from a single key) are used at different rounds of the algorithm.

Feistel Cipher Structure



- Input is a plaintext block of the size 2w bits;
- The block is divided into two parts L_0 and R_0;
- Two parts going through n rounds of processing;
- At every round, a function F (round function) is applied to the right half using a (sub)key, the result is XOR'ed with the left half of the data;
- At every round a new (sub)key may be used; all sub(keys) are generated from the same secret key

Decryption in a Feistel cipher

- The same algorithm is used as for encryption;
- The only difference is subkeys should be applied in a reverse order:
  - If for encryption $K_1,\ldots K_n$ have been used at the rounds 1, …,n, then
  - $K_n,\ldots,K_1$ are used for decryption at the rounds 1,..n.

Choices in Feistel network scheme

- **Block size:** the larger the block size the more secure and slower the scheme is. 64 bits is a usual size;
- **Key size:** the larger key size means the greater security but slower the scheme. 128 bits is the most common key length;
- **Number of rounds:** more rounds means more security;
- **Subkey generation algorithm:** more complex algorithm generally means more difficult cryptanalysis;
- **Round function:** the same as above.

Symmetric Encryption Algorithms

Most important symmetric block ciphers

- DES (Data Encryption Standard);
- 3DES (triple DES);
- AES (Advanced Encryption Standard);

Data Encryption Standard (DES)

- Adopted in 1977 by National Bureau of Standards (now NIST);
- The algorithm itself is called Data Encryption algorithm (DEA);
- A variant of the Feistel schema;
- Blocks have a size 64-bits;
- The key is 56 bits long;
- Uses 16 rounds of processing;
- From the original 56-bit kay, 16 subkeys are generated, one for each round.

The weakness of DEA

- **Weakness:** the size of the key (56 bits).

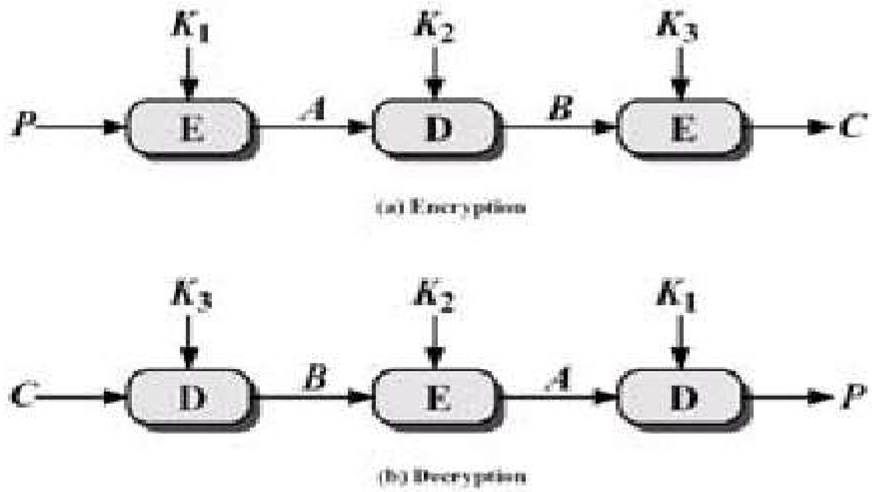
  Altogether there are $2^{56} \approx 7.2 \times 10^{16}$ different keys of such a length;
- The number is huge, but the special purpose machine "DES cracker" built in 1998 was able to break the algorithm in a less than 3 days using brute-force search;
- **Remedy:** increase the length of the key!! Increasing the length to 128 bits would increase the time of the brute-force search by "DES cracker" to 10^{18} years.

The strength of DEA

- The strength of DEA is based on the fact that no essentially better than brute-force search attack is known for DEA;
- In other words, no fatal weakness of DES itself has been discovered (only the weakness related to the small length of the key);
- But, no proof that an efficient attack is impossible.

Triple DES

- Triple DES (3DES) is a standard introduced in 1985;
- 3DES algorithm does what its name says: it runs DES (rather DEA) algorithm 3 times;
- It uses three keys, one for each execution of DEA;



(a) Encryption

(b) Decryption

Encryption and Decryption in 3DES

Encryption: $C = E_{K_3}[D_{K_2}[E_{K_1}[P]]]$

Decryption: $P = D_{K_1}[E_{K_2}[D_{K_3}[C]]]$

Where

- C is ciphertext
- P is plaintext
- $E_K[X]$ is encryption of X using key K
- $D_K[Y]$ is decryption of Y using key K

3DES is compatible with DES: $C = E_{K_1}[D_{K_1}[E_{K_1}[P]]] = E_{K_1}[P]$

Advanced Encryption Standard

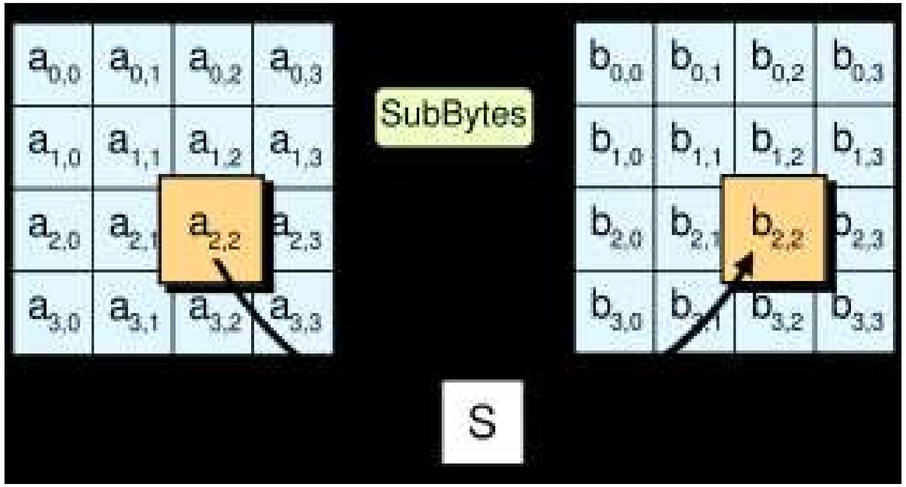
- Designers: J. Daemen, Vincent Rijmen
- First published: 1998
- Became effective as a NIST standard May, 2002
- A variant of substitution-permutation network
- Key size is 128, 192 or 256 bits
- Number of rounds is 10, 12, or 14

Advanced Encryption Standard

- Design uses theory of finite fields, a branch of algebra;
- Every block of 128 bits is presented as 4 by 4 array of bytes
- Key Expansion: Key → Round keys

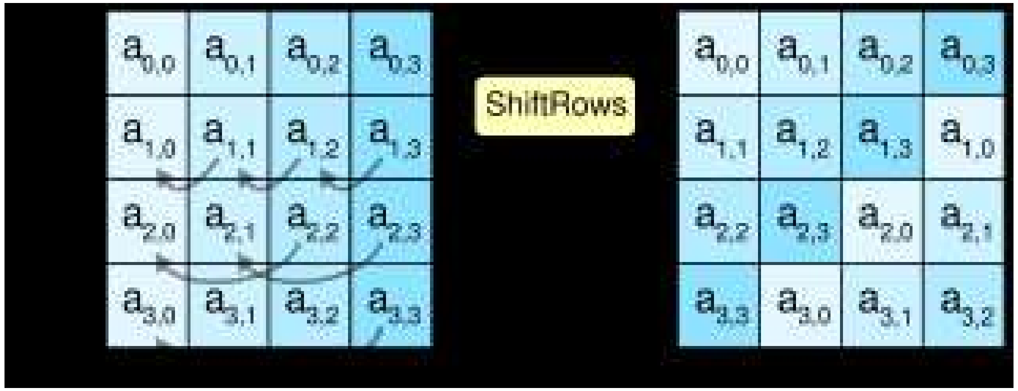
Steps in AES processing, I

- Every round includes the following steps:
  - **Substituiton**: each byte is replaced with another based on lookup table



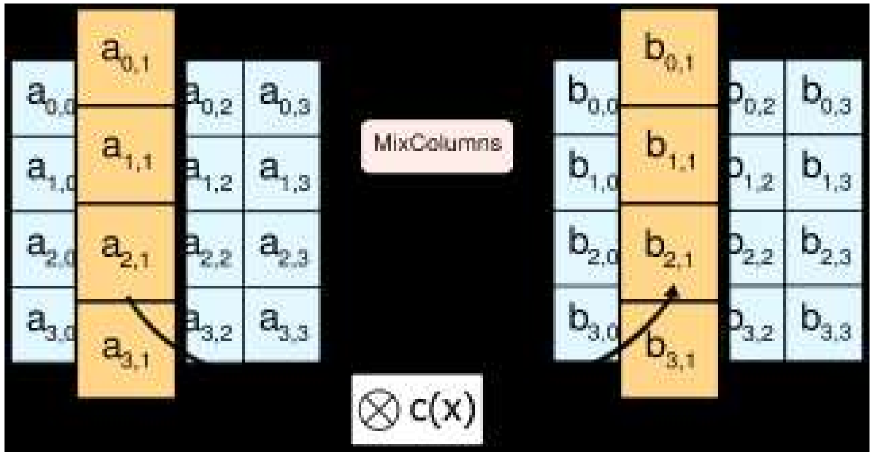
Steps in AES processing, II

- **ShiftRows**: each row is shifted cyclically certain amount of steps



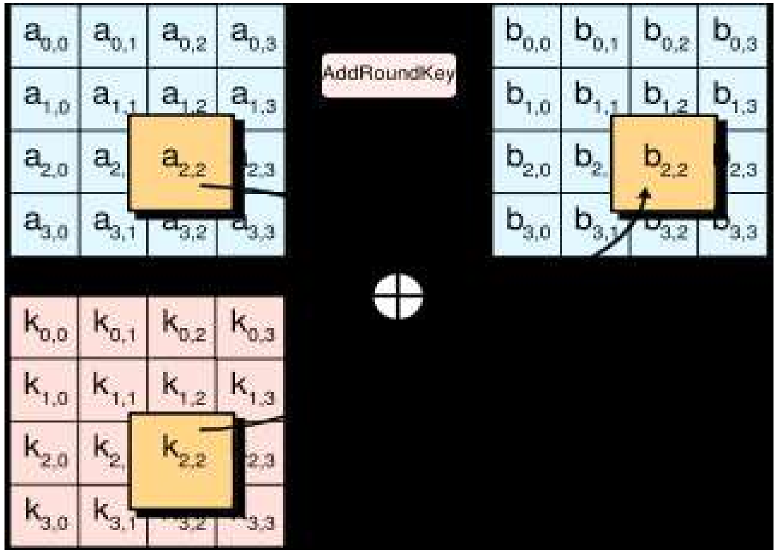
Steps in AES processing, III

- **MixColumns:** mixing operation on the columns (defined in terms of computations in a finite field).



Steps in AES processing, IV

- **AddRoundKey-** each byte is combined with the round key



Security of AES

- Considered secure for use for classified information, secret and top secret level;
- However, there are some concerns related to the algebraic foundations of algorithm – underlying algebraic structure might be used in the attacks in some clever way;
- The above is for Black Box setting; rather efficient Side Channels attacks have been discovered recently

Symmetric Encryption. Part 2

Block ciphers modes

- Block ciphers may be used in different modes. Most common modes are

- Electronic Codebook Mode (ECB)
- Cipher Block Chaining (CBC)
- Cipher Feedback Mode (CFB)

Electronic Codebook Mode (ECB)

- **Simple mode**: each block, say of size 64 bits is encrypted with the same key;
- For a given block of the plaintext and a given key the result of encryption is unique;
- If a block of plaintext is repeated several times, the result of encryption contains several copies of the same ciphertext;
- So, the encryption of the lengthy (regular) messages might be insecure.

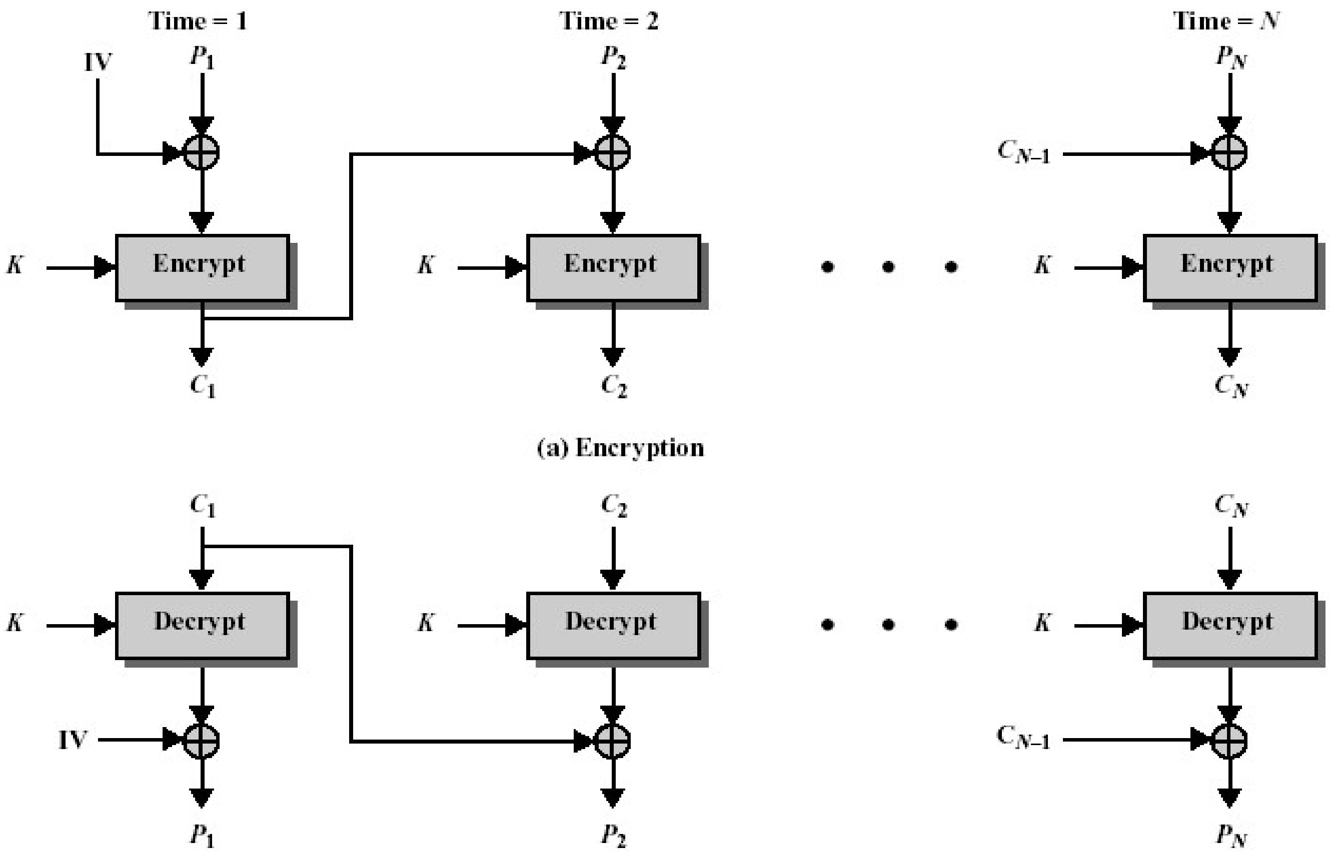
Cipher Block Chaining Mode (CBC)

- CBC mode fixes abovementioned disadvantage of ECB mode: here the **same** blocks of plaintext may produce **different** blocks of ciphertext;
- **Simple idea:** before encryption a block of the plaintext is XOR'ed with the result of encryption of the previous block;
- For the first block encryp … vector (IV) is used;

$$C_i = E_K[C_{i-1} \oplus P_i]$$

- It is better to keep both a key and IV secret.

CBC encryption and decryption



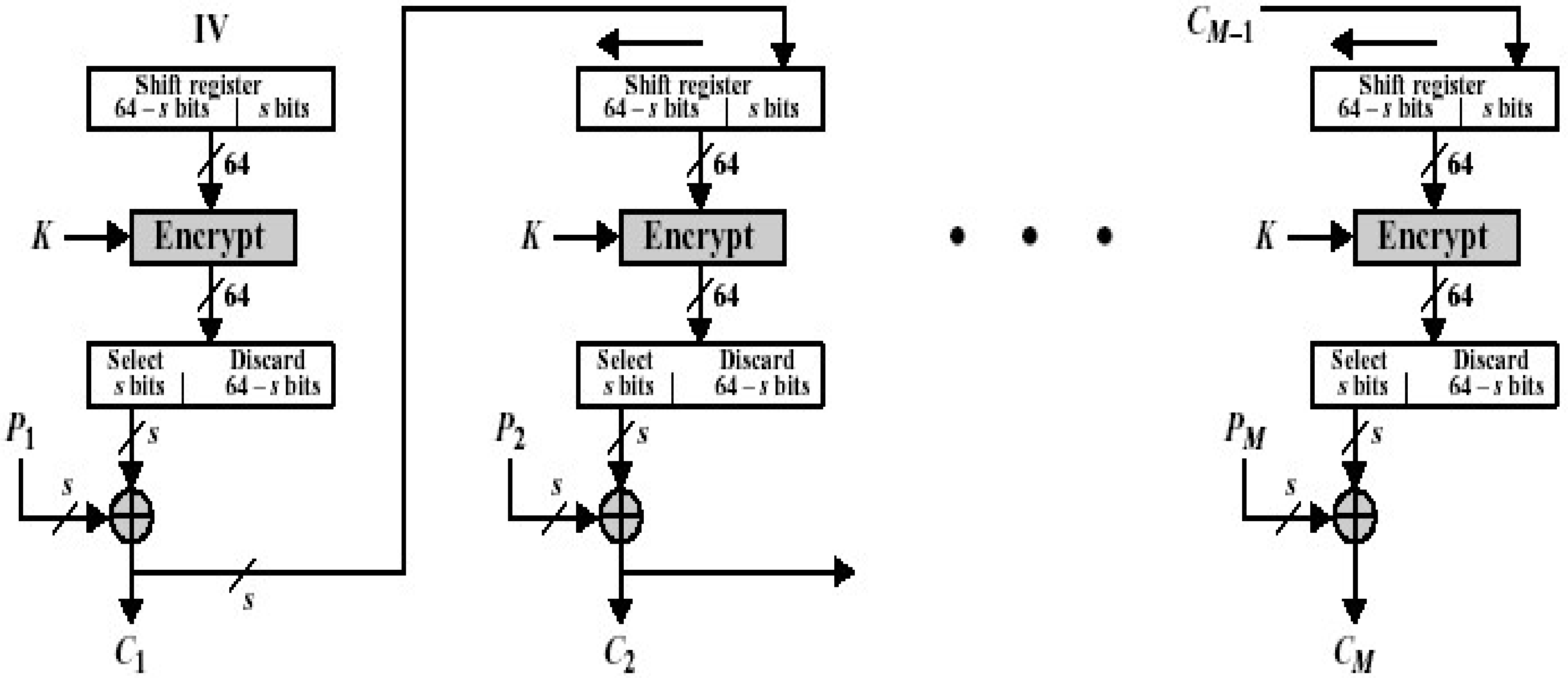
(a) Encryption

(b) Decryption

Cipher Feedback Mode (CFB)

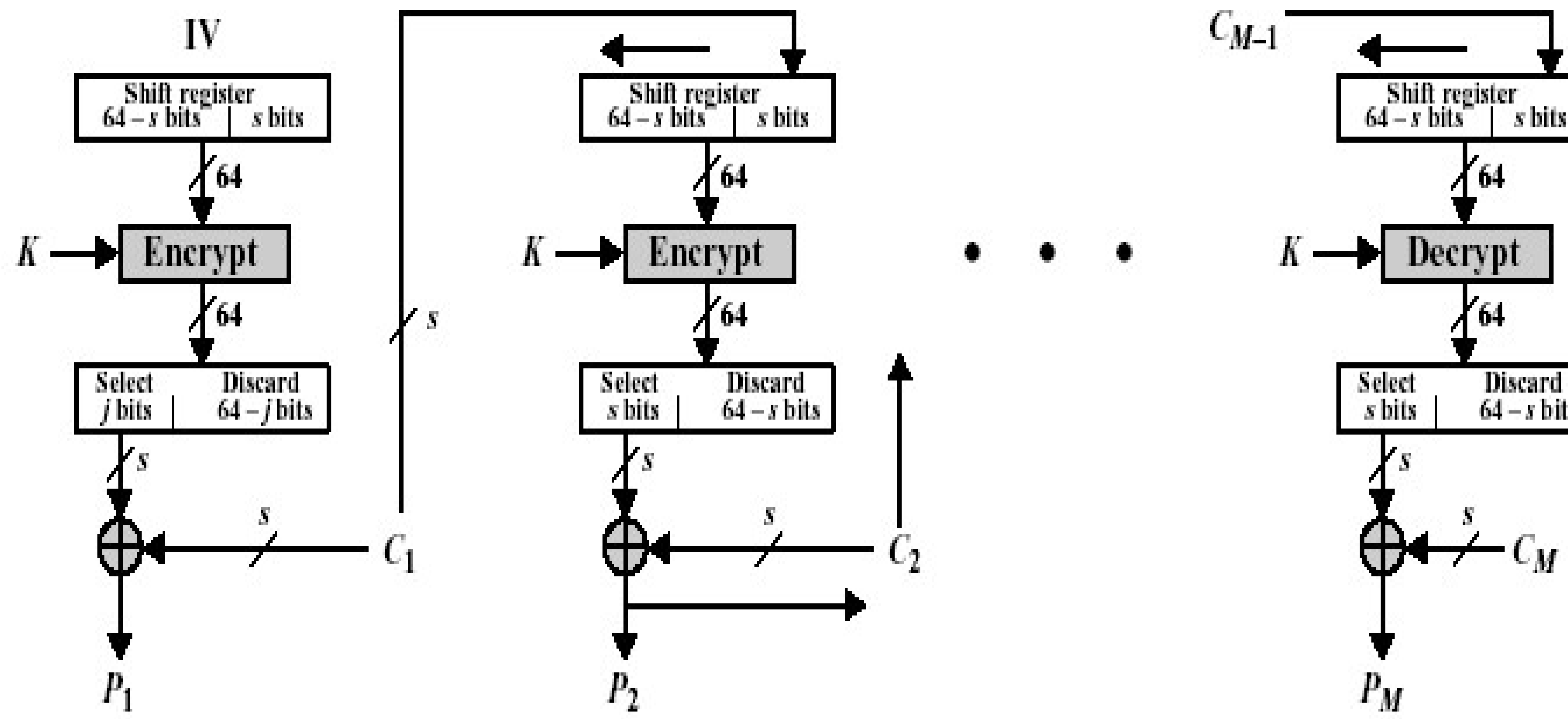
- CFB mode may be used to transform a block cipher to the stream cipher;
- It has a parameter *s* (the size of transmission unit); if 8-bit characters are used as transmission unit, then *s = 8;*
- Shift register of the size equal to the size of the block of the block cipher is used (typically it is 64 bits);
- Again, an initialisation vector is needed.

s-bits CFB encryption



(a) Encryption

s-bits CFB decryption



(b) Decryption

Key distribution

- **From requirements for symmetric encryption:**
- “Sender and receiver must *have obtained copies of the secret key* in a secure way and must keep the key secure”
- **Important issue**: how to distribute secret keys?

Key distribution, manual delivery

- For two parties A and B:
- A key could be created by A and delivered physically to B (or vice versa);
- A key could be created by the third trusted party C and delivered physically to A and B;

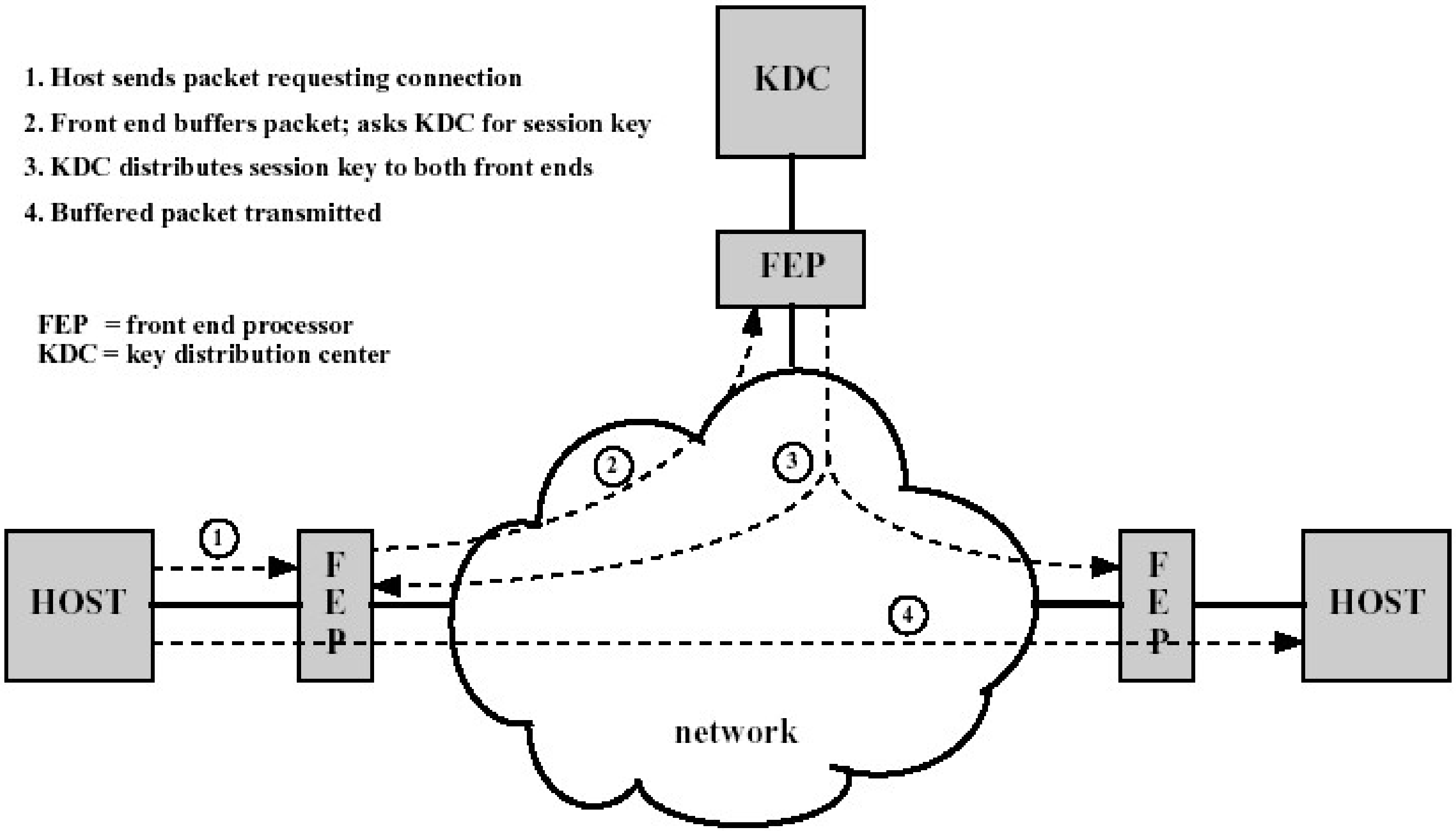
- Difficult to use in wide area distributed systems, when dynamic connections are needed.

Key distribution, further techniques

- If A and B have used recently a secret key, one of them could create a new secret key and send it to the partner using old key;
- *Potential problem: once an attacker learned one key, he can disclose all keys afterwards*

- There is a third trusted party C connected by encrypted channels with both A and B. Then C creates a key and distributes it among A and B using encrypted channels;

Automated key distribution



Finally

- The option we will discuss next time:

- *Both parties use public-key cryptographic techniques*