Chapter 2

Conventional Encryption Message Confidentiality

Henric Johnson
Blekinge Institute of Technology, Sweden
http://www.its.bth.se/staff/hjo/
henric.johnson@bth.se

Outline

- Conventional Encryption Principles
- Conventional Encryption Algorithms
- Cipher Block Modes of Operation
- Location of Encryption Devices
- Key Distribution

Conventional Encryption Principles

- An encryption scheme has five ingredients:
  - Plaintext
  - Encryption algorithm
  - Secret Key
  - Ciphertext
  - Decryption algorithm
- Security depends on the secrecy of the key, not the secrecy of the algorithm

Conventional Encryption Principles

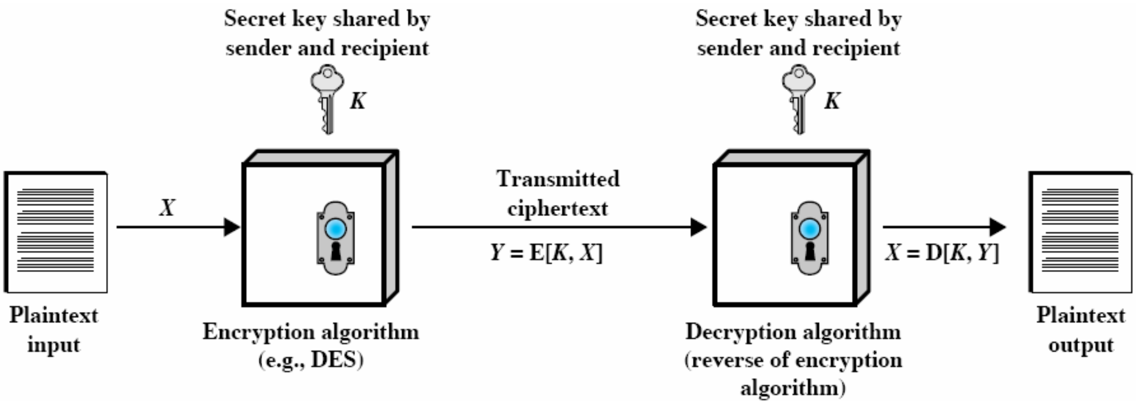


Figure 2.1 Simplified Model of Symmetric Encryption

Cryptography

- Classified along three independent dimensions:
  - The type of operations used for transforming plaintext to ciphertext
    - substitution
    - transposition
  - The number of keys used
    - symmetric (single key)
    - asymmetric (two-keys, or public-key encryption)
  - The way in which the plaintext is processed
    - Block cipher
    - Stream cipher

Cryptanalysis

- Cryptanalysis
  - The process of attempting to discover the plaintext or key
  - Attacks depend on the nature of the encryption scheme and the information available to the cryptanalyst.
    - Table 2.1
- Computationally secure

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

Average time required for exhaustive key search

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

Feistel Cipher Structure

- Virtually all conventional block encryption algorithms, including DES have a structure first described by Horst Feistel of IBM in 1973
- The realisation of a Fesitel Network depends on the choice of the following parameters and design features (see next slide):

Feistel Cipher Structure

- **Block size:** larger block sizes mean greater security
- **Key Size:** larger key size means greater security
- **Number of rounds:** multiple rounds offer increasing security
- **Subkey generation algorithm:** greater complexity will lead to greater difficulty of cryptanalysis.
- **Fast software encryption/decryption:** the speed of execution of the algorithm becomes a concern

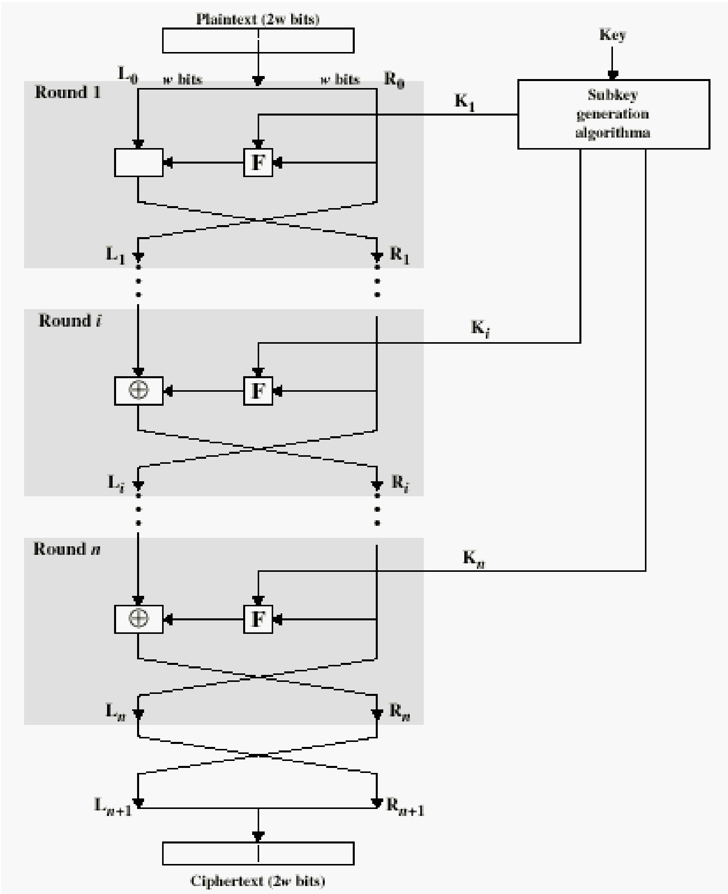
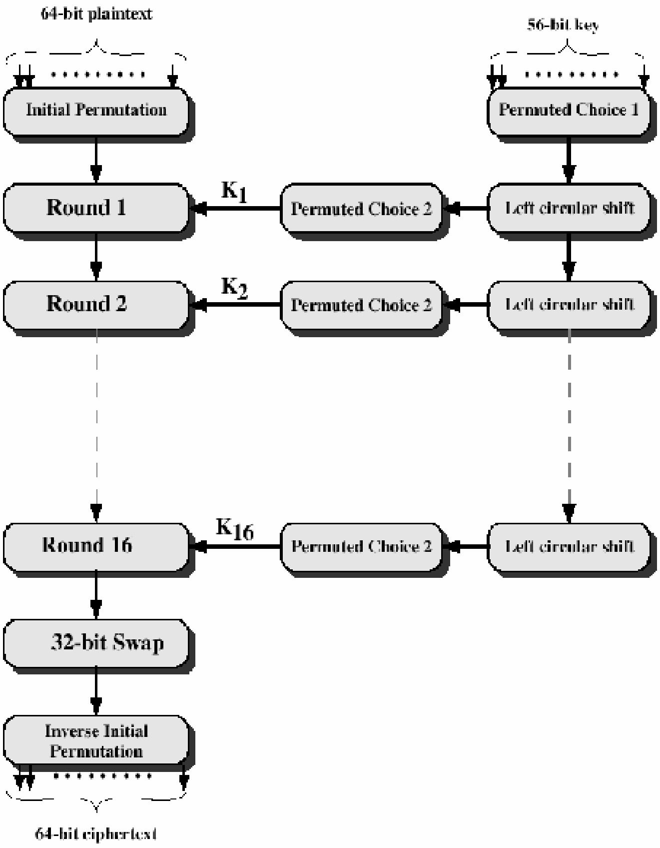


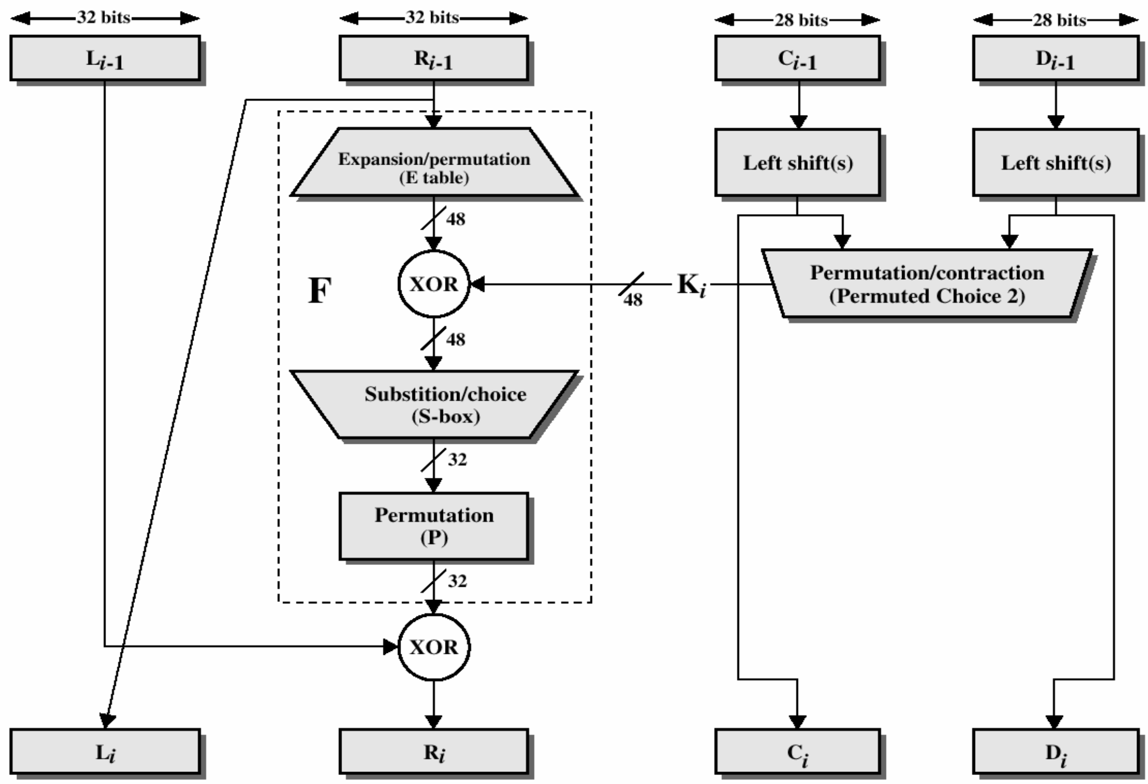
Figure 2.2 Classical Feistel Network

Conventional Encryption Algorithms

- Data Encryption Standard (DES)
  - The most widely used encryption scheme
  - The algorithm is reffered to the Data Encryption Algorithm (DEA)
  - DES is a block cipher
  - The plaintext is processed in 64-bit blocks
  - The key is 56-bits in length



General Depiction of DES Encryption Algorithm

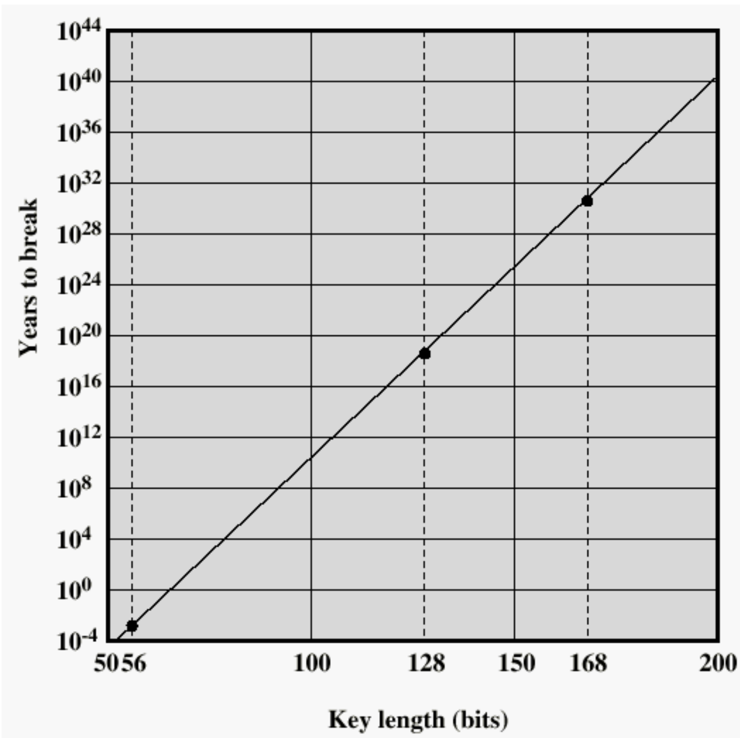


Single Round of DES Algorithm

DES

- **The overall processing at each iteration:**
  - $L_i = R_{i-1}$
  - $R_i = L_{i-1} \oplus F(R_{i-1}, K_i)$
- Concerns about:
  - The algorithm and the key length (56-bits)

Time to break a code (10^6 decryptions/µs)



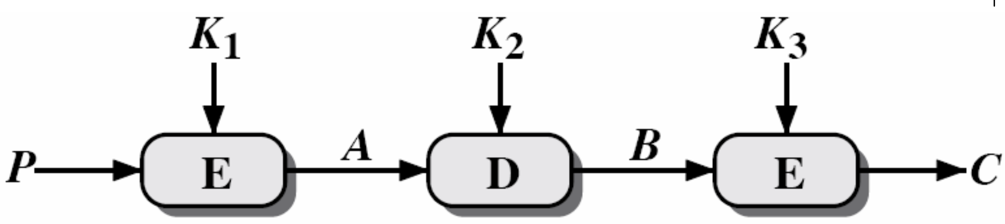
Triple DES

- Use three keys and three executions of the DES algorithm (encrypt-decrypt-encrypt)

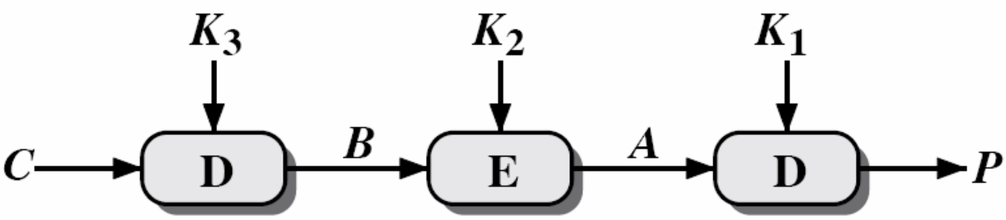
$$C = E_{K3}[D_{K2}[E_{K1}[P]]]$$

  - C = ciphertext
  - P = Plaintext
  - $E_K[X]$ = encryption of X using key K
  - $D_K[Y]$ = decryption of Y using key K
- Effective key length of 168 bits

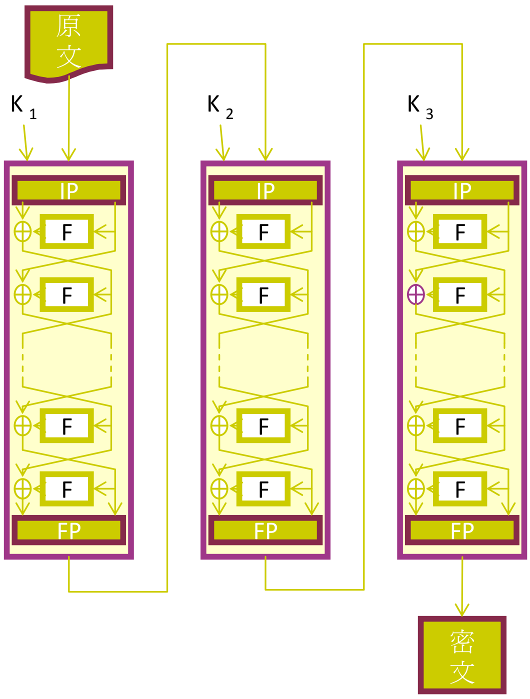
Triple DES



(a) Encryption



(b) Decryption



Advanced Encryption Standard (AES)

- 3DES's strength
  - 168-bit key length V.S. brute-force attack
  - Encryption algorithm in 3DES is the same as in DES
- 3DES's drawback
  - Relatively sluggish in software
  - 64-bit block size
- AES
  - A symmetric block cipher with a 128-bit block length
  - Key lengths of 128, 192, and 256 bits

Advanced Encryption Standard (AES)

- Evaluation criteria
  - Security
  - Computational efficiency
  - Memory requirements
  - Hardware and software suitability
  - Flexibility
- Comments on AES
  - It's **not a Feistel structure**.
  - Four different stages are used.

Substitute bytes	substitution
Shift rows	permutation
Mix columns	substitution
Add round key	substitution

Advanced Encryption Standard (AES)

- Only the **Add Round Key** stage makes use of the key.
- **The decryption algorithm is not identical to the encryption algorithm.**
- The final round of both encryption and decryption consists of only three stages.

Other Symmetric Block Ciphers

- **International Data Encryption Algorithm (IDEA)**
  - 128-bit key
  - Used in PGP
- **Blowfish**
  - Easy to implement
  - High execution speed
  - Run in less than 5K of memory

Other Symmetric Block Ciphers

- **RC5**
  - Suitable for hardware and software
  - Fast, simple
  - Adaptable to processors of different word lengths
  - Variable number of rounds
  - Variable-length key
  - Low memory requirement
  - High security
  - Data-dependent rotations
- **Cast-128**
  - Key size from 40 to 128 bits
  - The round function differs from round to round

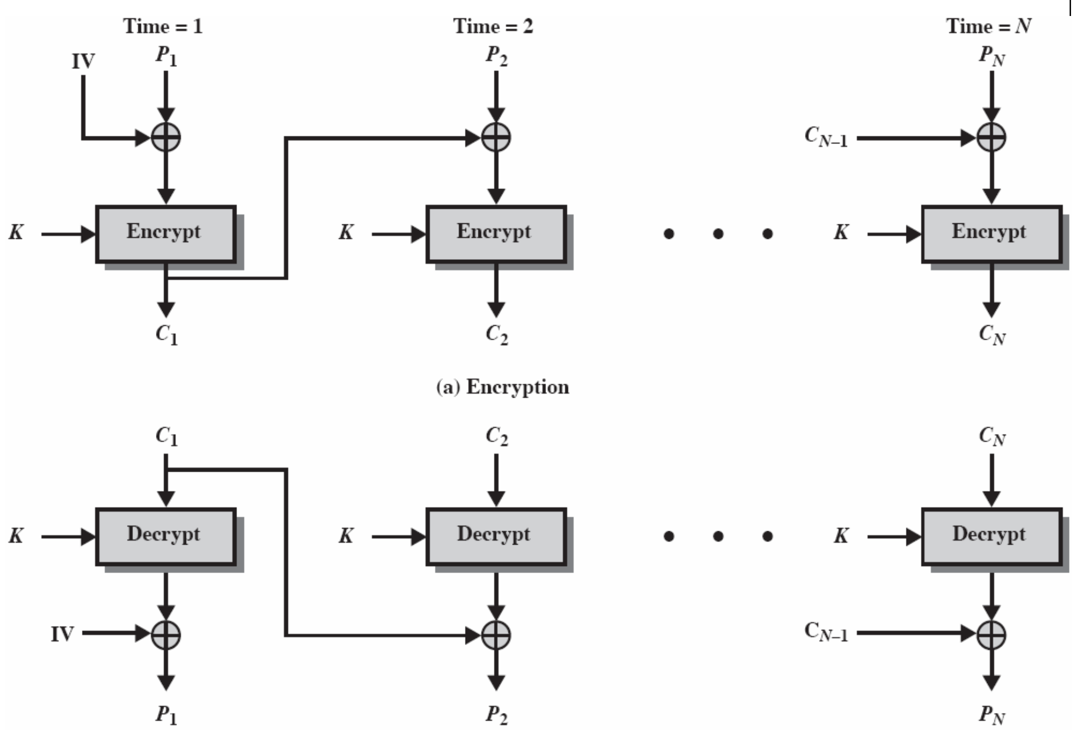
Cipher Block Modes of Operation

- Cipher Block Chaining Mode (CBC)
  - The input to the encryption algorithm is the XOR of the current plaintext block and the preceding ciphertext block.
  - Repeating pattern of 64-bits are not exposed

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

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

Cipher Block Chaining (CBC) Mode



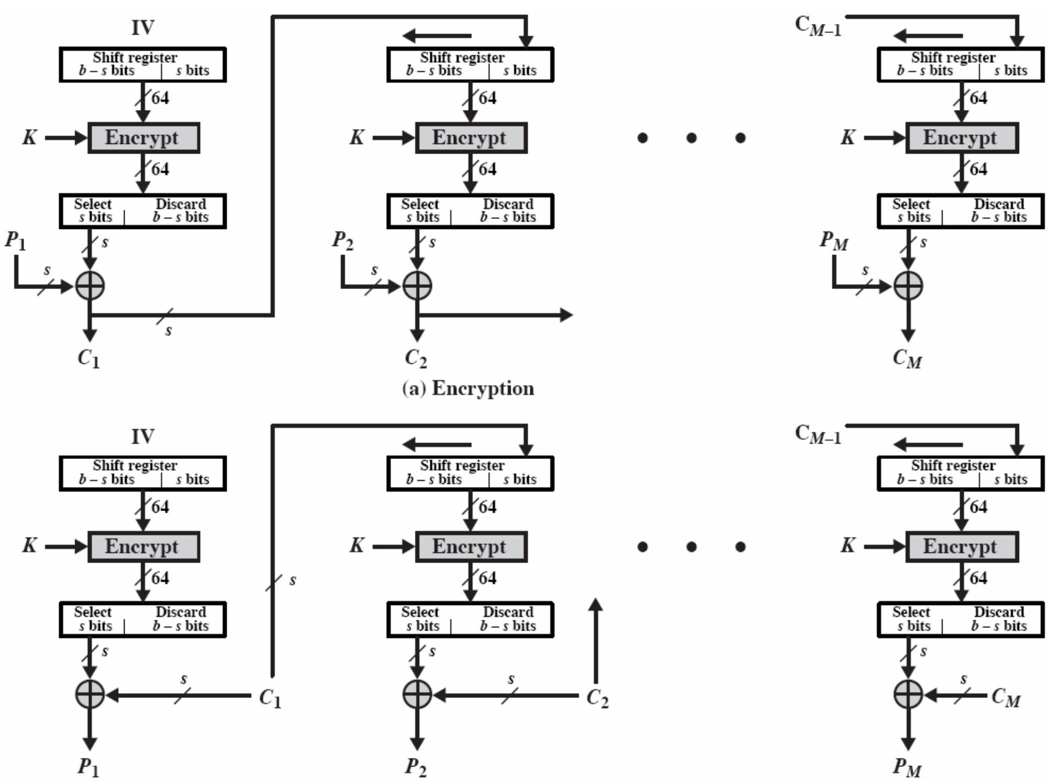
(a) Encryption

(b) Decryption

Cipher Feedback Mode (CFB)

- To convert any block cipher into a stream cipher by CFB mode.
- To eliminate the need to pad a message to be an integral number of blocks.
- The ciphertext is of the same length as the plaintext.
- Note that it is the *encryption* function that is used, not the *decryption* function.

s-bit Cipher Feedback (CFB) Mode



Location of Encryption Device

- **Link encryption:**
  - A lot of encryption devices
  - High level of security
  - Decrypt each packet at every switch
- **End-to-end encryption**
  - The source encrypt and the receiver decrypts
  - Payload encrypted
  - Header in the clear
- **High Security:** Both link and end-to-end encryption are needed (see Figure 2.9)

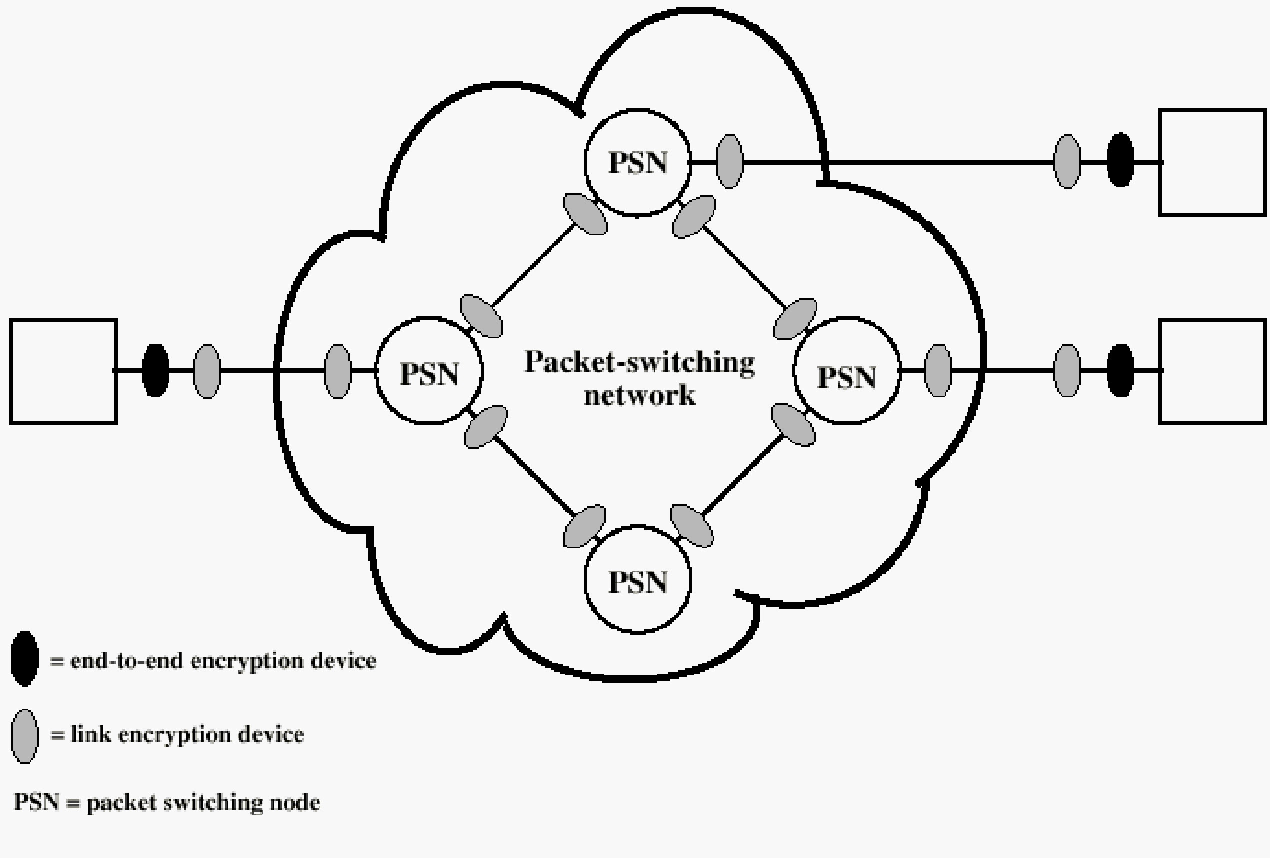


Figure 2.9 Encryption Across a Packet-Switching Network

Key Distribution

1. A key could be selected by A and physically delivered to B.
2. A third party could select the key and physically deliver it to A and B.
3. If A and B have previously used a key, one party could transmit the new key to the other, encrypted using the old key.
4. If A and B each have an encrypted connection to a third party C, C could deliver a key on the encrypted links to A and B.

Key Distribution (See Figure 2.10)

- **Session key:**
  - Data encrypted with a one-time session key. At the conclusion of the session the key is destroyed
- **Permanent key:**
  - Used between entities for the purpose of distributing session keys

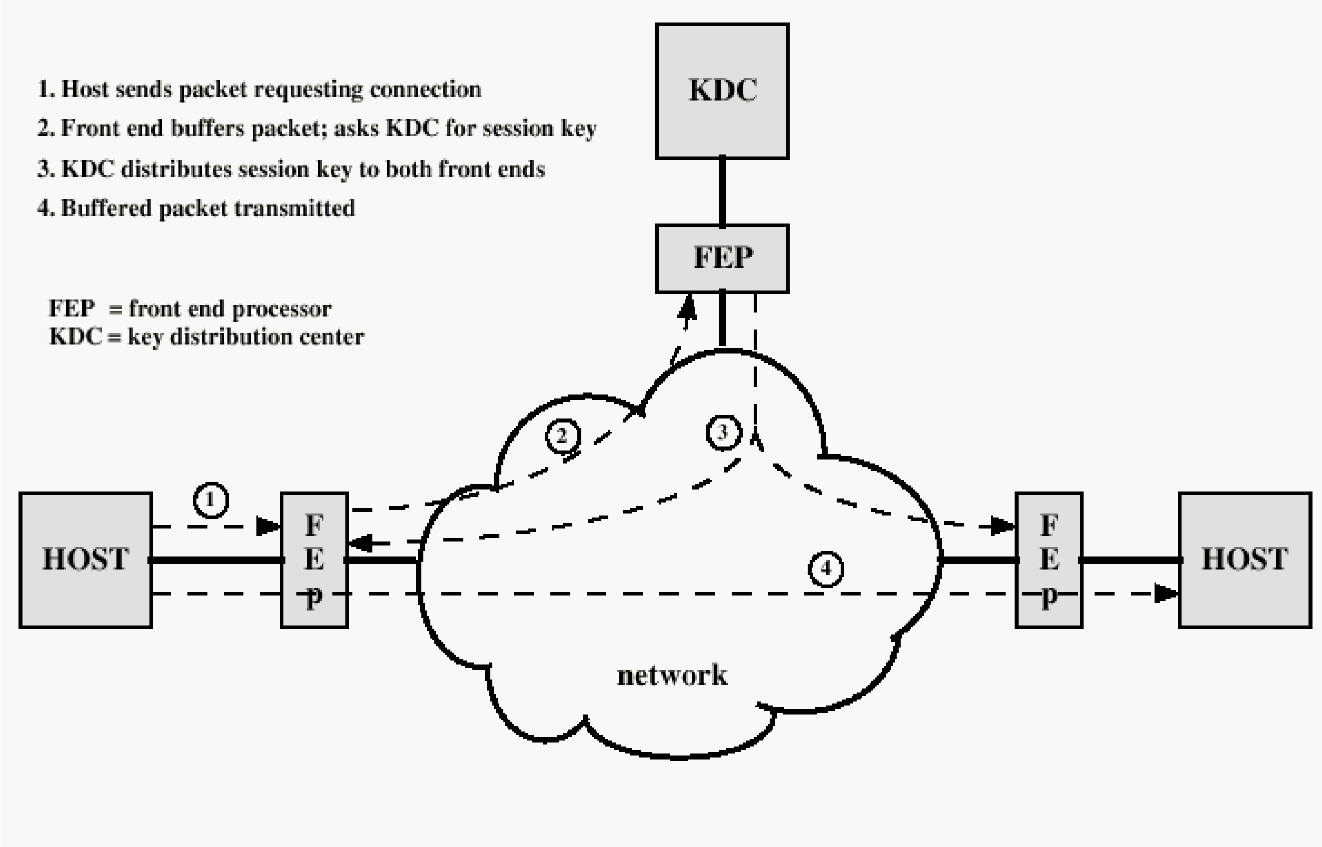
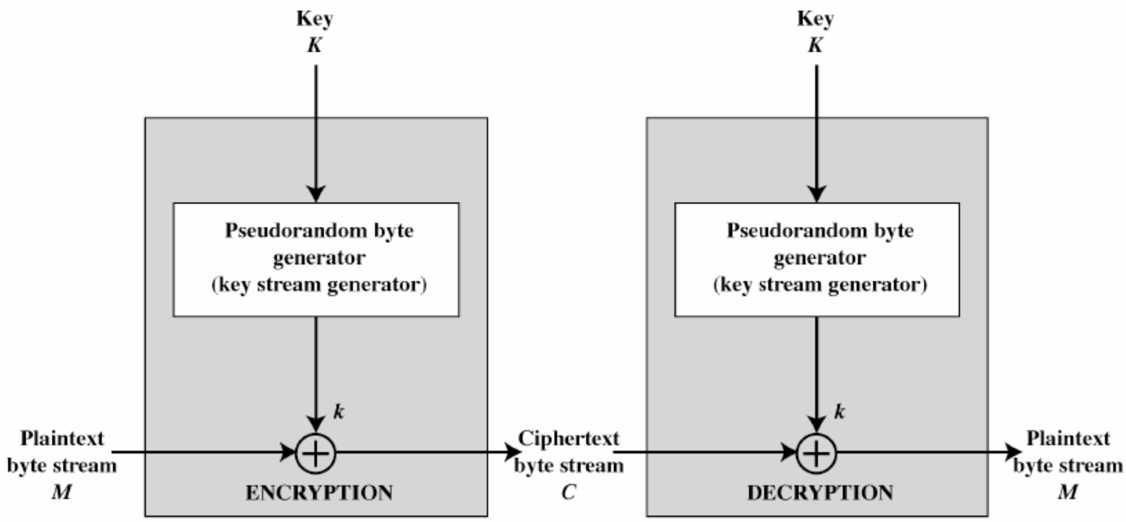


Figure 2.10 Automatic Key Distribution for Connection-Oriented Protocol

Stream Ciphers

- process the message bit by bit (as a stream)
- typically have a (pseudo) random **stream key**
- combined (XOR) with plaintext bit by bit
- randomness of **stream key** completely destroys any statistically properties in the message
  - $C_i = M_i \text{ XOR } StreamKey_i$
- what could be simpler!!!!
- **but must never reuse stream key**
  - otherwise can remove effect and recover messages

Stream Cipher Diagram



Stream Cipher Properties

- some design considerations are:
  - long period with no repetitions
  - statistically random
  - depends on large enough key

RC4

- a proprietary cipher owned by RSA DSI
- another Ron Rivest design, **simple but effective**
- variable key size, byte-oriented stream cipher
- widely used (web SSL/TLS, wireless WEP)
- key forms random permutation of all 8-bit values
- uses that permutation to scramble input info processed a byte at a time

RC4 Key Schedule

- starts with an array S of numbers: 0..255
- use key to well and truly shuffle
- S forms **internal state** of the cipher
- given a key k of length l bytes

```
/* Initialization */
for i = 0 to 255 do
    S[i] = i;
    T[i] = K[ i mod keylen ];

/* Initial Permutation of S */
j = 0
for i = 0 to 255 do
    j = (j + S[i] + T[i]) mod 256;
    swap (S[i], S[j]);
```

RC4 Encryption

- encryption continues shuffling array values
- sum of shuffled pair selects "stream key" value
- XOR with next byte of message to en/decrypt

```
/* Stream Generation */
i, j = 0 ;
while (true)
    i = (i + 1) mod 256;
    j = (j + S[i]) mod 256;
    swap(S[i], S[j]);
    t = (S[i] + S[j]) mod 256;
    k = S[t]

    Ci = Mi XOR S[t]
```


RC4 Security

- claimed secure against known attacks
  - have some analyses, none practical
- result is very non-linear
- since RC4 is a stream cipher, must **never reuse a key**
- have a concern with WEP, but due to key handling rather than RC4 itself

Recommended Reading

- Stallings, W. *Cryptography and Network Security: Principles and Practice, 2nd edition*. Prentice Hall, 1999
- Scneier, B. *Applied Cryptography*, New York: Wiley, 1996
- Mel, H.X. Baker, D. *Cryptography Decrypted*. Addison Wesley, 2001