Symmetric Cryptographic Systems

Dr Basel Halak

Learning Outcomes

At the end of this unit you should be able to:

1. Discuss the basic principles of discrete probability and information theory
2. Describe the principle of Shannon perfect secrecy
3. Verify the security of traditional ciphers
4. Explain how to construct a secure block cipher.
5. Outline the basic operation principles of 3DES.
6. Encrypt and Decrypt using Advance Encryption Standard Algorithm(AES).
7. Describe the Cryptographic Modes of Block Ciphers.
8. Describe a number of cryptographic attacks on block ciphers

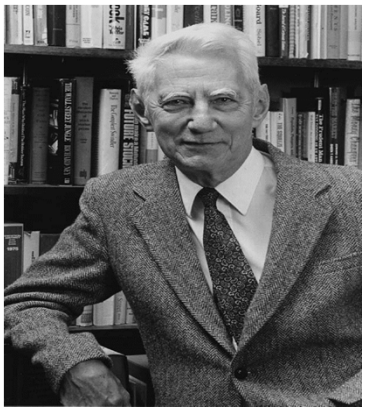
Learning Outcomes

At the end of this unit you should be able to:

1. Discuss the basic principles of discrete probability and information theory
2. Describe the principle of Shannon perfect secrecy
3. Verify the security of traditional ciphers
4. Explain how to construct a secure block cipher.
5. Outline the basic operation principles of 3DES.
6. Encrypt and Decrypt using Advance Encryption Standard Algorithm(AES).
7. Describe the Cryptographic Modes of Block Ciphers.
8. Describe a number of cryptographic attacks on block ciphers

Quantifying Information

- Shannon (1948) : "a signal that is totally predictable carries no information"
- In essence, the difference in the signal between what is predicted and actual values is a measure of its information content or *entropy*.
- Following Shannon the unit is the bit
- A system contains N-bits of information if it contains 2^N possible characters.



Information and Entropy

- Consider an information source with a set of symbols *ai*, where each symbol occurs in the data with probability *pi:*
- The entropy of this information source is:

$$H(X) = \sum_i p_i \log_2 (1 / p_i)$$

- Information theory **uses *entropy* as a measure of how much information is encoded in a message.**

1/27/2017

How much can we compress

The main theorem proved by Shannon says essentially that the minimum number of bits needed to binary code a message is given as follows:

$$\boldsymbol{B = n \cdot H(X)}$$

Where

n: the number of symbols in the message

H(X): The entropy of the message

1/27/2017

How much can we compress

- Example:

Let us consider the following message

A B A E S E D A

The probability of the character 'A' appearing in this manuscript is 0.375

The probability of the character 'B' appearing in this manuscript is 0.125

The probability of the character 'E' appearing in this manuscript is 0.25

The probability of the character 'D' appearing in this manuscript is 0.125

The probability of the character 'S' appearing in this manuscript is 01.25

The entropy of this message is:

$$H(X) = \sum_i p_i \log_2 (1 / p_i)$$

$$H(X) = 2.15 \; Bits$$

1/27/2017

How much can we compress

- Example:

According to Shannon the minimum number of bits (B) needed to encode this message is (n*H) bits, where

n : the number of symbols in the message

H: the message entropy

This means

B = 2.15* 8= 17.2 bits

Assuming we are using standard 8-bit ASCII characters to encode this message, we will need: 8*8= 66 bits

The difference between the 17.2 bits and the 64 bits used to encode the message is redundant data

This where the potential for data compression arises.

1/27/2017

Quiz

Calculate the information contents of this message:

X X X XY X X X X Z X X X XT X X X XW

Quiz

$$H(X) = \sum_i p_i \log_2 (1 / p_i)$$

$$H(X) = 1.121 \quad Bits$$

B = 1.121* 20= 22.43 bits

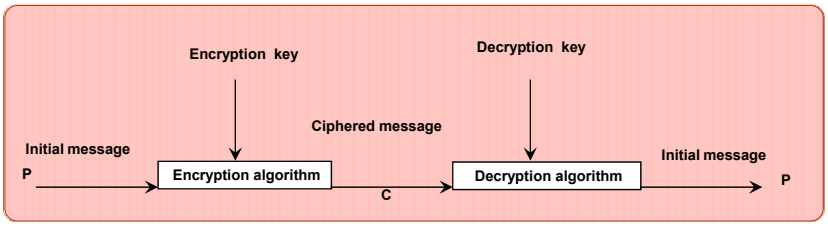
Learning Outcomes

At the end of this unit you should be able to:

1. Discuss the basic principles of discrete probability and information theory
2. Describe the principle of Shannon perfect secrecy
3. Verify the security of traditional ciphers
4. Explain how to construct a secure block cipher.
5. Outline the basic operation principles of 3DES.
6. Encrypt and Decrypt using Advance Encryption Standard Algorithm(AES).
7. Describe the Cryptographic Modes of Block Ciphers.
8. Describe a number of cryptographic attacks on block ciphers

Cryptography: Main Concepts

The general scheme of symmetric enciphering

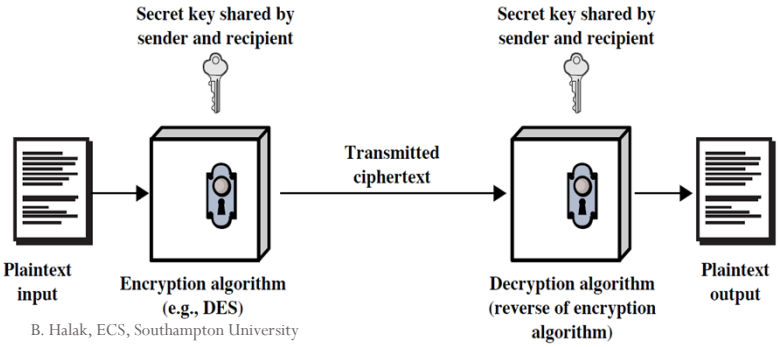


The encryption algorithm uses encryption key to transform a plaintext into a ciphertext.

The decryption algorithm uses decryption key to transform a ciphertext into a plaintext

Cryptography: Main Concepts

- **Shared Key Systems :** Both sender and receiver use the same key which must remain private. These systems are also called **symmetric**
- **Examples**: DES, Triple-DES, Twofish, Rijndael

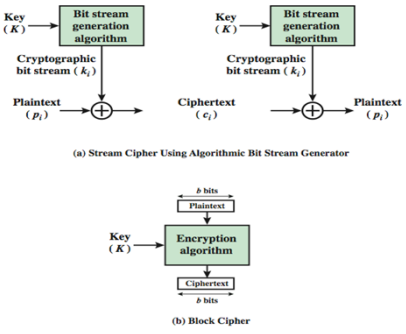


B. Halak, ECS, Southampton University

Stream Ciphers vs. Block Ciphers

- **Block ciphers:** in which a block of plaintext is treated as a whole and used to produce a ciphertext block of equal length. Typically, a block size of 128 bits or more is used. For Examples: AES.

- **Stream cipher: in which** the plaintext is encrypted/decrypted one bit or one byte at a time. For Examples includes, XOR cipher, **RC4** A5/1, A5/2,



Stream Ciphers vs. Block Ciphers

Cryptography: Main Concepts

Problems with Shared Key Systems

1. Compromised key means interceptors can decrypt any ciphertext they have acquired. Keys can be changed frequently to limit damage.
2. Distribution of keys is problematic: keys must be transmitted securely e.g. distribute key in pieces over separate channels.

Cryptography: Main Concepts

One very important principle in cryptography is **Kirchhoff's Principle**, which was documented in his book in (1883). Kirchhoff stated

"the cipher must not depend on the secrecy of the mechanism, it must not matter if it falls into the hands of the enemy."

Cryptography: Main Concepts

- **What parts of a cryptosystem must be kept secret?**

1. Encryption Algorithm
2. Decryption Algorithm
3. Keys
4. Ciphertext

Cryptography: Main Concepts

- **Symmetric Cipher Definition**: A cipher is defined over (K, M, C) is a pair of algorithms (E, D) where:

$$E : K \times M \longrightarrow C$$

$$D : K \times C \longrightarrow M$$

Such that: E and D satisfy the following consistency equation:

$$\forall\, m \in M, k \in K : D\big(k, E(k, m)\big) = m$$

Cryptography: Main Concepts

- **What is a secure cipher?**

Shannon (1949) basic idea : ciphertext reveals no info about plain text

- **Shannon Perfect Secrecy**:

A cipher (E, D) over (K, M, C) has perfect secrecy if:

$$\forall\, m1, m2 \in M, \quad \big(lengh(m1) = length(m2)\big)\ and\ \forall c\ \in C$$

$$\Pr[E(k, m1) = c] = \Pr[E(k, m2) = c]$$

Where k is uniform in K ($k \xleftarrow{R} K$)

Cryptography: Main Concepts

- **Shannon Perfect Secrecy: A cipher (E, D) over (K, M, C) has perfect secrecy if:**

$$\forall\, \boldsymbol{m1, m2} \in \boldsymbol{M},\ \big(\boldsymbol{lengh(m1) = length(m2)}\big)\ \boldsymbol{and}\ \forall \boldsymbol{c}\ \in \boldsymbol{C}$$

$$\boldsymbol{Pr[E(k, m1) = c] = Pr[E(k, m2) = c]}\ \textbf{ where } \quad (\boldsymbol{k \xleftarrow{R} K})$$

- This means given CT , I wont be able to deduce whether the message is m1 or m2 or any other m, therefore the most powerful attacker can learn nothing about the plaintext from the cipher text. This means **there is no cipher text only attack** (but other attacks may be possible).

Learning Outcomes

At the end of this unit you should be able to:

1. Discuss the basic principles of discrete probability and information theory
2. Describe the principle of Shannon perfect secrecy
3. Verify the security of traditional ciphers
4. Explain how to construct a secure block cipher.
5. Outline the basic operation principles of 3DES.
6. Encrypt and Decrypt using Advance Encryption Standard Algorithm(AES).
7. Describe the Cryptographic Modes of Block Ciphers.
8. Describe a number of cryptographic attacks on block ciphers

Cryptography: Main Concepts

- **Shift Cipher Review:**

1. Mathematically, map the message letters to numbers

   a, b, c, …, x, y, z

   0, 1, 2, ….., 23, 24, 25

2. Encrypt by shifting each letter in the message by K positions (k is the key)

   $c = E_K(p) = (p + k) \bmod 26$

3. Example: m = *"HELLO"*; k = 1; c = *"IFMMP"*

Cryptography: Main Concepts

- **Does Shift Cipher satisfies Shannon perfect secrecy for messages with a length ≥ 2 ?**

Take m1 = "AC", m2 = "AZ", and c = "BD".

Now assuming all keys have the same probability:

$$\boldsymbol{Pr[E(k,m1) = c] = \frac{1}{26}}$$

However, for all k ∈ K we have E(k,m2) ≠ c, and hence

$$\boldsymbol{Pr[E(k,m2) = c] = 0}$$

and so the perfect secrecy requirement is violated, and the shift cipher is prone to cipher text only attack

- **Exercise: Prove that for a one-letter long message , shift cipher is immune to ciphertext only attack**

Cryptography: Main Concepts

- **One Time Pad Review**

OTP is defined over the following sets:

$M = C = \{0,1\}^n$

$K = \{0,1\}^n$

Encryption:

$c = E(k,m) = k\ xor\ m$

Decryption:

m= $D(k,c) = k\ xor\ c$

Cryptography: Main Concepts

- **Is One Time Pad secure?**

Quiz: Let $m \in M$ and $c \in C$. How many OTP keys map $\boldsymbol{m}$ to $\boldsymbol{c}$?

a) None

b) 1

c) 2

d) Depends on m

Cryptography: Main Concepts

- **Is One Time Pad secure?**

For OTP is defined over the following sets:

$M = C = \{0,1\}^n$

$K = \{0,1\}^n$

$\forall\, m, c \quad if\ E(k,m) = c\ this\ means$

$c = k\ xor\ m$ therefore $k = c\ xor\ m$

The size of $\{k \in K : E(k,m) = c\} = 1$

Therefore $\forall\, m, c \quad \boldsymbol{Pr[E(k,m) = c]} = \frac{\{k \in K:\ E(k,m)=c\}}{|\boldsymbol{K}|} = \frac{1}{|K|}$

Therefore OTP has perfect secrecy

Cryptography: Main Concepts

- The bad news: Shannon has proved later that in order for a cipher to achieve the perfect secrecy than:

$$|\mathcal{K}| \geq |\mathcal{M}|$$

This means the key must be at least as long as the message.

Cryptography: Main Concepts

- **Shannon's notion of perfect secrecy is too strong to be useful**
- **Shannon's notion of perfect secrecy is only related to ciphertexts attacks**
- **Solution**: relax this notion from information-theoretic secrecy to *computational* secrecy. While information-theoretic secrecy required that every given a ciphertext, every plaintexts are exactly as likely, the computational secrecy notion will ask only that no *efficient algorithm* can tell, given a ciphertext, and, say, any two messages that could potentially be plaintexts corresponding to this ciphertext, whether one of these messages is more likely than the other to be the actual plaintext. Other possible attacks should also be considered.

Cryptography: Main Concepts

- **Shannon Perfect Secrecy:** A cipher (E, D) over (K, M, C) has perfect secrecy if:

  $\forall\, m1, m2 \in M,\ \big(lengh(m1) = length(m2)\big)\ and\ \forall c\ \in C$

  $Pr[E(k, m1) = c] = Pr[E(k, m2) = c]$ where $(k \xleftarrow{R} K)$

- **Computational Perfect Secrecy :** A cipher (E, D) over (K, M, C) has perfect secrecy if:

  $\forall\, m1, m2 \in M,\ \big(lengh(m1) = length(m2)\big)\ and\ \forall c\ \in C$

  $Pr[E(k, m1) = c] - P\,r[E(k, m2) = c]$

  is "negligible " where $(k \xleftarrow{R} K)$

  but also need adversary to exhibit $m_1\ m_2 \in M$ explicitly

- Both notions assume that the attacker can only observe the ciphertexts.

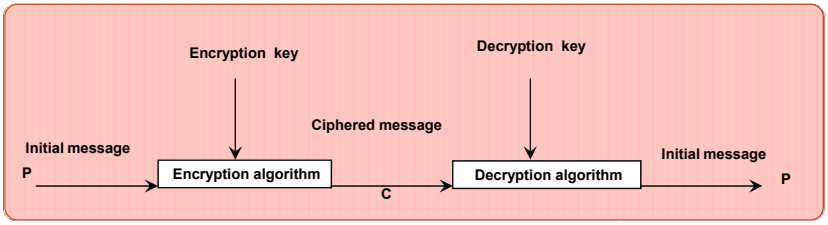
Learning Outcomes

At the end of this unit you should be able to:

1. Discuss the basic principles of discrete probability and information theory
2. Describe the principle of Shannon perfect secrecy
3. Verify the security of traditional ciphers
4. Explain how to construct a secure block cipher.
5. Outline the basic operation principles of 3DES.
6. Encrypt and Decrypt using Advance Encryption Standard Algorithm(AES).
7. Describe the Cryptographic Modes of Block Ciphers.
8. Describe a number of cryptographic attacks on block ciphers

Cryptography: Main Concepts

The general scheme of symmetric enciphering

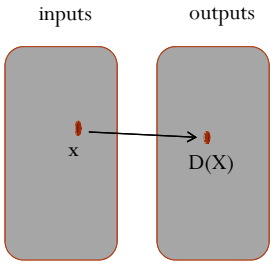


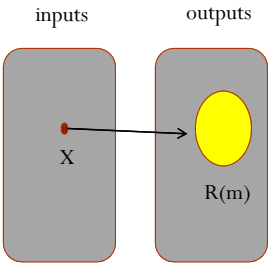
The encryption algorithm uses encryption key to transform a plaintext into a ciphertext

The decryption algorithm uses decryption key to transform a ciphertext into a plaintext

Deterministic vs Random Functions

- A function **D** is said to be deterministic if it always produces the same output of the input does not change.
- On the other hand a randomized function **R** may produce different outputs for the same inputs





PRPs and PRFs

- Pseudo Random Function (**PRF**) defined over (K,X,Y):

  $F: K \times X \rightarrow Y$

  such that exists "efficient" algorithm to evaluate F(k,x)

- **Pseudo Random Permutation (PRP) defined over (K,X):**

  $E: K \times X \rightarrow X$

  such that:

  1. Exists "efficient" deterministic algorithm to evaluate E(k, x)
  2. The function $E(k, \cdot)$ is one-to-one
  3. Exists "efficient" inversion algorithm D(k, y)

B. Halak, ECS, Southampton University

PRPs Examples

- Example PRPs: 3DES, AES, …

  AES: $K \times X \rightarrow X$ where $K = X = \{0,1\}^{128}$

  3DES: $K \times X \rightarrow X$ where $X = \{0,1\}^{64}$, $K = \{0,1\}^{168}$

- Functionally, any PRP is also a PRF.
  - A PRP is a PRF where X=Y and is efficiently invertible.

Semantic Security of PRP(one-time key)

- **Definition**: E is **semantically secure** if for all "efficient" A
  $\boldsymbol{Adv}[A,E]$ is negligible. Where:

$$\boldsymbol{Adv}[\boldsymbol{A},\boldsymbol{E}] = |\boldsymbol{EXP_A}[\boldsymbol{E}(\boldsymbol{k},\boldsymbol{m1})] - \boldsymbol{EXP_A}[\boldsymbol{E}(\boldsymbol{k},\boldsymbol{m2})]|$$

for all explicit m_0 , $m_1 \in M$ $\big(lengh(m1) = length(m2)\big)$

EXP the experiment that an adversary performs on encrypted messages.

In other words: the attacker cannot distinguish between encrypted messages.

Secure PRPs

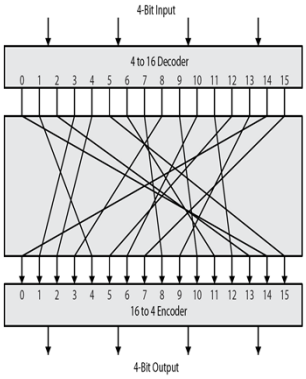
Quiz: Let P: $K \times X \rightarrow \{0,1\}^8$ be a secure PRP.
Is the following Y a semantically secure PRP?

$$Y(k, x) = \begin{cases} 001000101 \text{ if } x=0 \\ P(k, x) \quad \text{otherwise} \end{cases}$$

a) No, it is easy to distinguish Y from a random permutation
b) Yes, an attack on Y would also break P
c) It depends on P

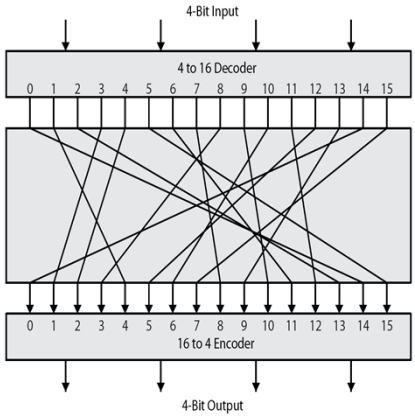
How to construct an ideal Block Cipher

- **One solution is to use a truly random permutation**
- Each block may be viewed as a gigantic character. The "alphabet" consists of 2^N gigantic characters.
- **Exercise**: Assume N=4, how many possible mapping exists between the input and the output? If the secret key indicate which mapping to use, how many bits do we need to represent this key ?



Principles of Modern Block Cipher

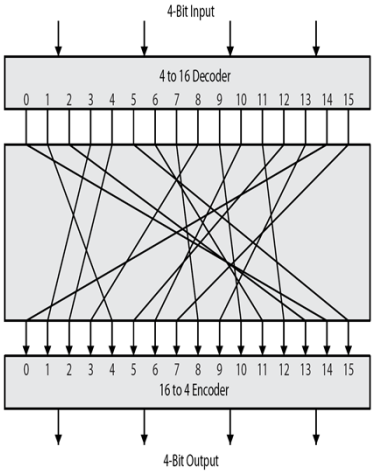
- Each particular cipher is a one-to-one mapping from the plaintext "alphabet" to the ciphertext "alphabet".
- There are $2^N!$ such mappings.
- A secret key indicates which mapping to use.



Ideal Block Cipher

- An ideal block cipher would allow us to use any of these $2^N!$ mappings.
  - The key space would be extremely large.
- But this would require a key of $\log_2(2^N!)$ bits.
- If N = 64,

  $\log_2(2^N!) \approx N \times 2^N \approx 10^{21}$ bits $\approx 10^{11}$ GB.
- Infeasible!



Learning Outcomes

At the end of this unit you should be able to:

1. Discuss the basic principles of discrete probability and information theory
2. Describe the principle of Shannon perfect secrecy
3. Verify the security of traditional ciphers
4. Explain how to construct a secure block cipher.
5. Outline the basic operation principles of 3DES.
6. Encrypt and Decrypt using Advance Encryption Standard Algorithm(AES).
7. Describe the Cryptographic Modes of Block Ciphers.
8. Describe a number of cryptographic attacks on block ciphers

Shannon's Confusion and Diffusion

- A cipher needs to completely obscure statistical properties of original message.
- Claude Shannon suggested using **both Substitution and Permutation** blocks to thwart cryptanalysis based on statistical analysis. He introduced two concepts:

1. **diffusion** –dissipates the redundancy of the plaintext by spreading it out over the cipher text. One of the easiest technique to achieve this is permutation
2. **confusion** : Obscures the relationship between ciphertext and the plaintext in order to hide any statistical patterns. One of the easiest technique to achieve this is substitution

Data Encryption Standard Algorithm

In 1971, IBM developed an algorithm, named LUCIFER which operates on a block of 64 bits, using a 128-bit key

↓

1973: NBS asks for block cipher proposals. IBM submits variant of Lucifer.

↓

1976: NBS adopts DES as a federal standard key-length = 56 bits ; block-length = 64 bits

↓

1997: DES broken by exhaustive search

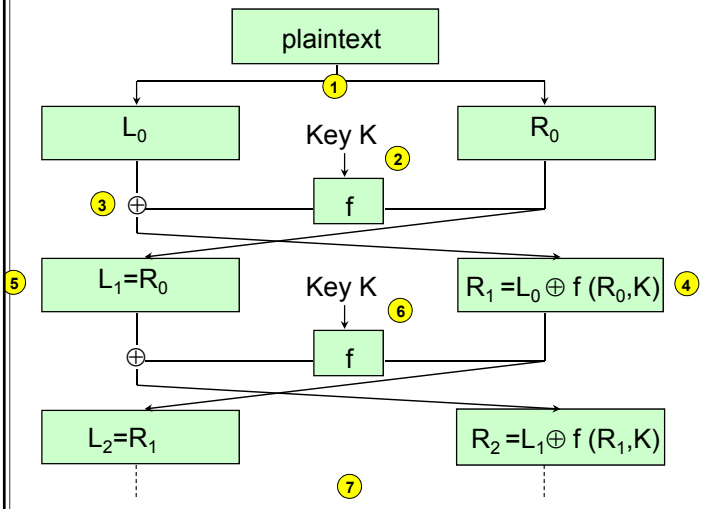
↓

2000: NIST adopts Rijndael as AES to replace DES

DES Core Idea (Feistel Structure)

- Horst **Feistel** proposed an approach in order **to avoid the complexity problem of ideal block ciphers**. His approach is based on Shannon s' diffusion and confusion.
- Feistel 's approach is based on the concept of "invertible product cipher" whereby **the execution of two or more simple ciphers in sequence** leads to a much more cryptographically secure solution than any of the component ciphers used. He employed the two primitive cryptographic operation: **substitution and permutation.**

Feistel Cipher Structure



$$R_i = F_i(R_{i-1}) + L_{i-1}$$
$$L_i = R_{i-1}$$

Feistel Cipher Structure

Claim: for all $f_1, \ldots, f_d$: $\{0,1\}^n \longrightarrow \{0,1\}^n$
Feistel network F: $\{0,1\}^{2n} \longrightarrow \{0,1\}^{2n}$ is invertible
In other word prove that given R_i, L_i ,
you can find R_{i-1}, L_{i-1}

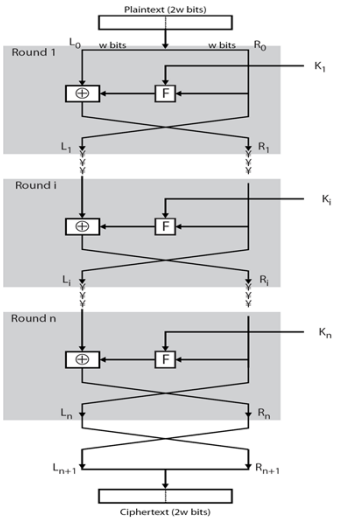
Proof: we have:

$$R_i = F_i(R_{i-1}, k) + L_{i-1}$$
$$L_i = R_{i-1}$$

This means : $R_{i-1} = L_i$
$L_{i-1} = ?$

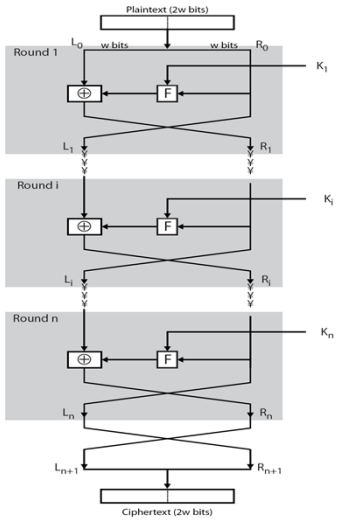
The Feistel Cipher Structure

Operation Principles(1):
The inputs to the algorithm are the plaintext (of length 2w bits) and a key K. The plaintext is split into two halves L and R, and the data is then passed through n "rounds "of processing and then recombined to produce the ciphertext.



The Feistel Cipher Structure

Operation Principles(2):
Each round has the same structure. The left halve of the data has a substitution performed. This requires a "round function" F to be performed on the right half of the data and then XORd with the left half. Finally a permutation is performed that requires the interchange of the two halves of the data.



The implementation of a Feistel Cipher

The implementation of a Feistel Cipher has the following key parameters:

1. **Block size**: A larger block size generally means greater security, but reduced speed. 64bit block sizes are very heavily used as being a reasonable trade-off- although AES now uses 128bits
2. **Key Size**: The same trade-off applies as for block size. Generally 64 bits is not now considered adequate and 128 bits is preferred.
3. **Number of rounds**: Each round adds additional security. A single round is inadequate, but 16 is considered standard
4. **Subkey generation**: the more complex this algorithm is, the more secure the overall system will be.
5. **Round function**: Greater complexity again means greater resistance to cryptanalysis.

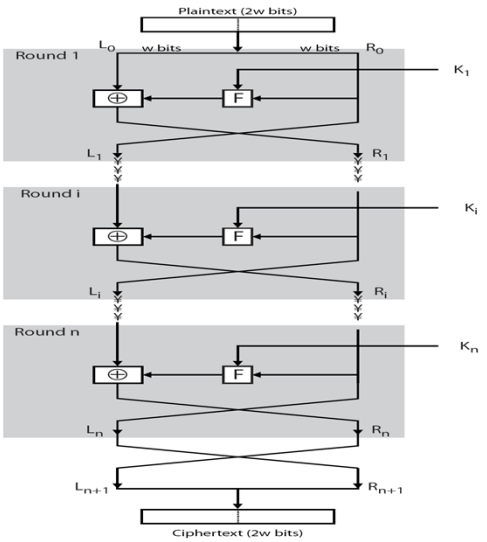
Data Encryption Standard Algorithm

Parameter	DES specification
Type of design	Feistel Cipher
Number of rounds	16
Block size	64
Length of key	56

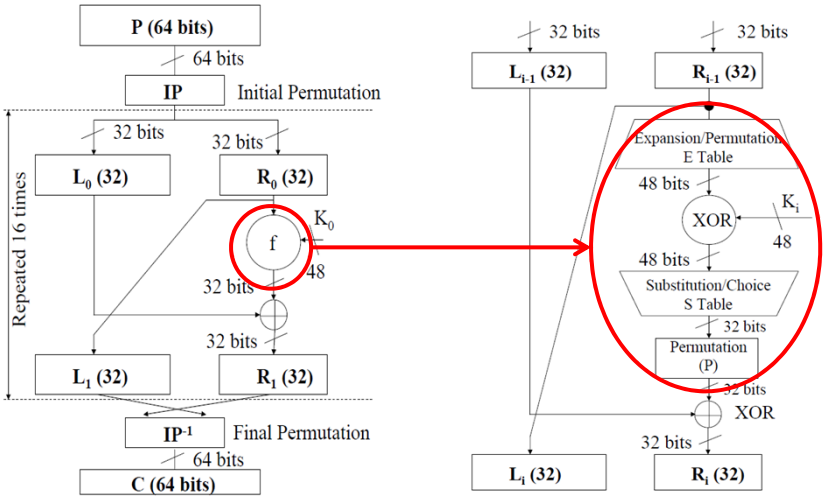
Data Encryption Standard Algorithm

- **What is specific to DES**

1. the design of the F function
2. how round keys are derived from the main key.



DES Coarse Structure



The F Function of DES

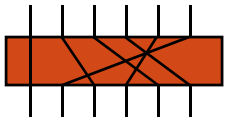
- The L and R each have 32 bits, and the round key K 48 bits.
- The F function, on input R and K, produces 32 bits:

$$F(R, K) = P\left(S\left(E(R) \oplus K\right)\right)$$

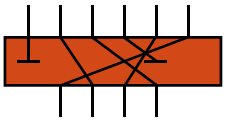
where E: expands 32 bits to 48 bits;
S: shrinks it back to 32 bits;
P: permutes the 32 bits.

DES Fine Structure Blocks

- **permutations**: Fixed known mapping 32-32 bits

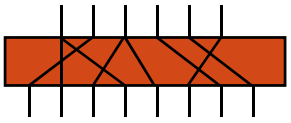


- **Compression Permutation(Permuted Choice):** Fixed known subset of 56 bit input mapped onto 48 bit output



DES Fine Structure Blocks

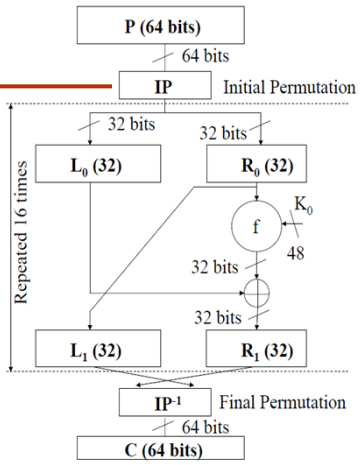
- **Expansion permutation:** 32 bit data shuffled and mapped (both operations fixed and known) onto 48 bits by duplicating 16 input bits. This makes diffusion quicker



- **Substitution**: 48 bits of data are divided into eight blocks of 6 bits. There are Eight S-boxes each map 6 to 4 bits

Initial Permutation IP

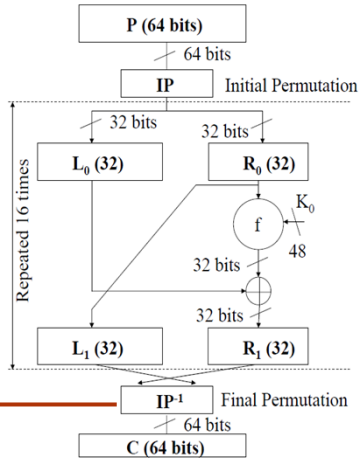
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7



This table specifies the input permutation on a 64-bit block. The meaning is as follows: the first bit of the output is taken from the 58th bit of the input; the second bit from the 50th bit, and so on, with the last bit of the output taken from the 7th bit of the input.

Inverse Permutation IP

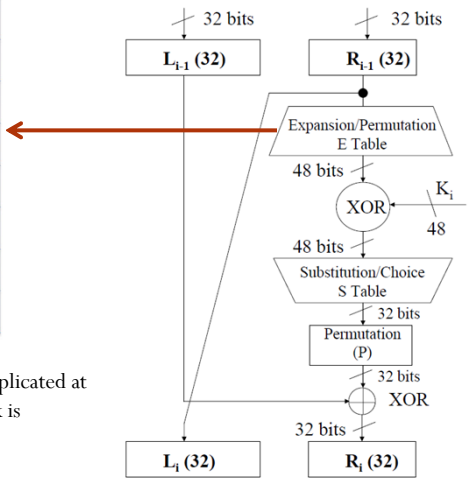
40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25



Note: $IP(IP^{-1}) = IP^{-1}(IP) = I$

The F Function
Expansion function (E)

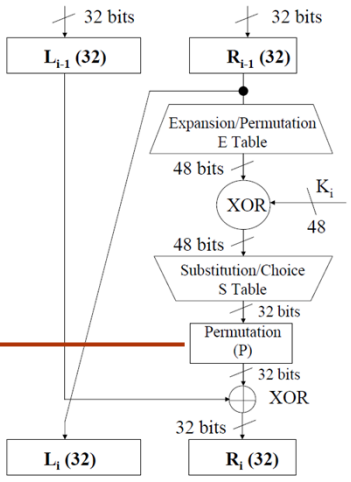
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1



Note that some bits from the input are duplicated at the output, therefore the 32-bit half-block is expanded to 48 bits.

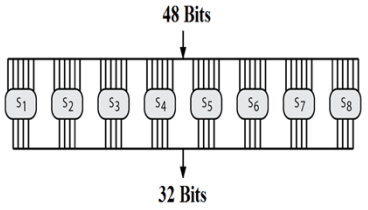
The F Function
Permutation P

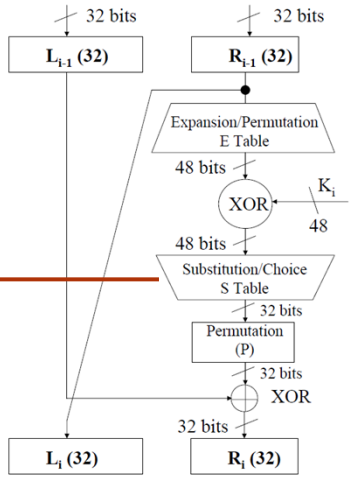
16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25



The P permutation shuffles the bits of a 32-bit half-block

The F Function
Substitution Boxes





- Eight S-boxes each map 6 to 4 bits

$$S_i: \{0,1\}^6 \longrightarrow \{0,1\}^4$$

The F Function
Substitution Boxes

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	6	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

- Each S-box is specified as a 4 x 16 table
  - each row is a permutation of 0-15
  - outer bits 1 & 6 of input are used to select one of the four rows
  - inner 4 bits of input are used to select a column
- All the eight boxes are different.

- For example, $S_1(101010) = 6 = 0110$.

The F Function
a bad S-box choice

If the S function is linear , then the entire DES cipher may become linear : this means there exists a fixed binary matrix such that:

$$DES(k, m) = \underset{64}{\overset{832}{A}} \cdot \begin{matrix} m \\ k_1 \\ k_2 \\ \vdots \\ k_{16} \end{matrix}^{832} = c \pmod 2$$

Given enough pairs of {c, m }, it is easy to calculate the secret key.

The F Function
S-boxes and P-box choice

- Choosing the S-boxes and P-box at random would result in an insecure block cipher as they may be close to linear functions which will allow key recovery after $\approx 2^{24}$ outputs) using one of the cryptanalysis techniques.
- **Therefore , a number of rules have been set for the choice of S-boxes**

1. No output bit should be close to a linear function of the input bits……

DES Design Controversy

Issues	Criticisms
Secret design criteria	Design criteria of round function and key schedules were secret. (although actual design public). However, Fear of trapdoors has proved baseless
Weak keys	Certain DES keys are **weak**. (encryption and decryption has same effect) Few such keys and their use easily avoided.
Inadequate key length	56 bits an inadequate key length. Criticized even in 1975 Unsubstantiated claims that NSA insisted on the "small" key length.

Strengths of DES

- **Avalanche effect:**
  - A small change in the plaintext or in the key results in a significant change in the ciphertext.
  - an evidence of high degree of diffusion and confusion
  - a desirable property of any encryption algorithm
- **DES exhibits a strong avalanche effect**
  - Changing 1 bit in the plaintext affects 34 bits in the ciphertext on average.
  - 1-bit change in the key affects 35 bits in the ciphertext on average.
- **Decryption** is performed by exactly the same procedure, except that the keys K1,… ,K16 are used in reverse order.

Attacks on DES

- **Exercise:** Suppose that we have a machine consisting of one million processors, each of which can test one million keys per second. How long is it likely to take before we find a DES key during an exhaustive key search?

Brute force Attacks on DES

Year	Technique	Implemented?	(Estimated) Cost in US$	(Estimated) Search time
1977	Diffie Hellman	No	20 million	20 hours
1993	Wiener	No	1 million	7 hours
1997	Rocke Verser et al.	Yes	Unknown	3 months
1998	Electronic Frontier Foundation [www.eff.org]	Yes DES Cracker machine was capable of testing over 90 billion keys per second	250, 000	56 hours
2008	COPACOBANA SciEngines GmbH	Yes (with cheap FPGA's)	10,000	Less than 24 hours

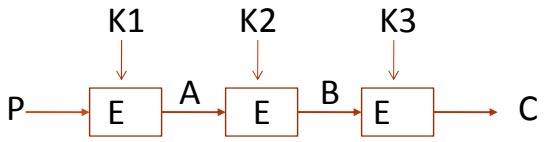
Multiple Encryption with DES

- DES is not secure enough.
- The once large key space, 2^{56}, is now too small.
- In 2001, NIST published the Advanced Encryption Standard (AES) as an alternative.
- But users in commerce and finance are not ready to give up on DES.
- Solution: to use multiple DES with multiple keys

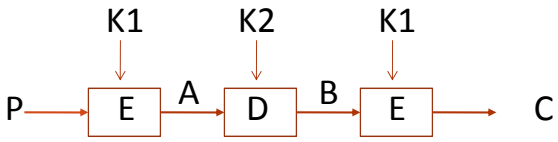
Triple DES with Two Keys

A straightforward implementation would be:

$C = E_{K1}(E_{K2}(E_{K3}(P)))$



- In practice: $C = E_{K1}(D_{K2}(E_{K1}(P)))$, **Why?**
- Also referred to as EDE encryption



Double-DES

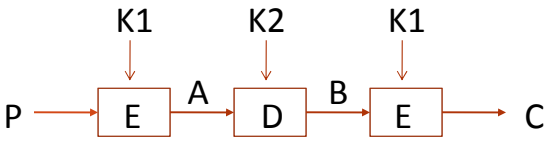
- Consider 2-DES with two keys:
  $C = E_{K2}(E_{K1}(P))$
- Decryption: $P = D_{K1}(D_{K2}(C))$
- Key length: 56 x 2 = 112 bits
- This should have thwarted brute-force attacks?
- Wrong!

Meet-in-the-Middle Attack on 2DES

- 2-DES: $C = E_{K2}(E_{K1}(P))$

- Given a known pair (P, C), attack as follows:
  - Encrypt P with all 2^{56} possible keys for K1.
  - Decrypt C with all 2^{56} possible keys for K2.
  - If $E_{K1'}(P) = D_{K2'}(C)$, try the keys on another (P', C').
  - If works, (K1', K2') = (K1, K2) with high probability.
  - Takes $O(2^{57})$ steps; not much more than attacking 1-DES.

Meet-in-the-Middle Attack on 3DES



1. For each possible key for K1, encrypt P to produce a possible value for A.
2. Using this A, and C, attack the 2DES to obtain a pair of keys (K2, K1').
3. If K1' = K1, try the key pair (K1, K2) on another (C',P').
4. If it works, (K1, K2) is the key pair with high probability.
5. It takes $O(2^{55} \times 2^{56}) = O(2^{111})$ steps on average.

Learning Outcomes

At the end of this unit you should be able to:

1. Discuss the basic principles of discrete probability and information theory
2. Describe the principle of Shannon perfect secrecy
3. Verify the security of traditional ciphers
4. Explain how to construct a secure block cipher.
5. Outline the basic operation principles of 3DES.
6. Encrypt and Decrypt using Advance Encryption Standard Algorithm(AES).
7. Describe the Cryptographic Modes of Block Ciphers.
8. Describe a number of cryptographic attacks on block ciphers

Where did AES Come from?

In the 1997, the U.S. National Institute of Standards and Technology (NIST), published a request for information regarding the creation of a new Advanced Encryption Standard (AES) for non-classified government documents. **AES would specify an unclassified, publicly disclosed encryption algorithm(s), available royalty-free, worldwide. The algorithm(s) must implement symmetric key cryptography as a block cipher and (at a minimum) support block sizes of 128-bits and key sizes of 128-, 192-, and 256-bits.**

⇩

9 Months later, fifteen different designs were created and submitted from several different countries.

⇩

In 1999, NIST announced the final 5: MARS, RC6, Rijndael, Serpent, and Twofish in AES2 conference

⇩

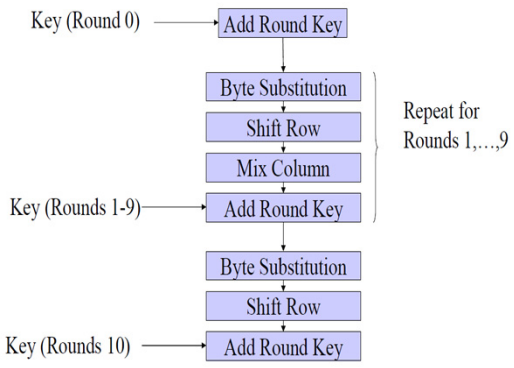
In April 2000 at AES3 a representative of each of the final five teams made a presentation arguing why their design should be chosen as the AES.

⇩

In October 2, 2000, NIST announced that Rijndael had been selected as the proposed AES

AES Encryption Algorithm

- AES is based on substitution-permutation network and not on Feistel network.
- Decryption is done by applying the inverse function of each step (i.e. all functions must be invertible).



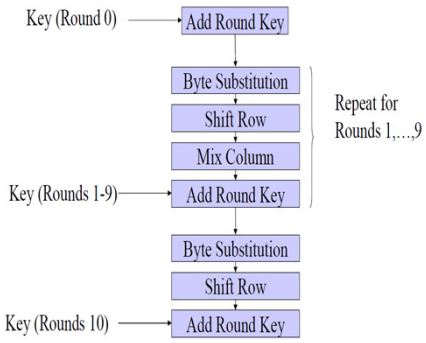
B. Halak, ECS, Southampton University

AES Algorithm Overview

- Each block consists of 128 bits, and these are divided into 16 bytes.
- Each of the operations acts upon these 8 bit bytes in

State :

$$\begin{pmatrix} a_{0,0} & a_{0,1} & a_{0,2} & a_{0,3} \\ a_{1,0} & a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,0} & a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,0} & a_{3,1} & a_{3,2} & a_{3,3} \end{pmatrix}$$



- State is a variable holding the current data block

AES Algorithm Overview

- Each ($a_{i,j}$) is an 8-bit byte, viewed as elements of the Galois filed GF(2^8).
- Eight bits in a byte is considered as a polynomial:

$$b_7x^7 + b_6x^6 + b_5x^5 + b_4x^4 + b_3x^3 + b_2x^2 + b_1x + b_0$$

For example: 1001001 = $x^7 + x^4 + 1$

- **Different arithmetic operations**
- **Addition** is XOR ($\oplus$)
- **Multiplication** is modular using a prime polynomial ($X^8 + X^4 + X^3 + X + 1$) i.e.(100011011). To multiply two elements, their representative polynomials are multiplied then divided by the prime polynomial above. The answer is the remainder of the division

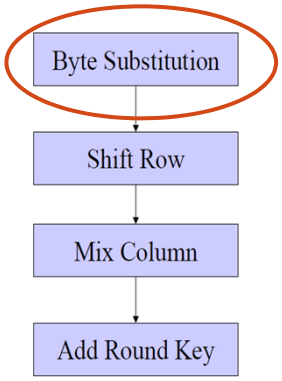
AES Encryption Algorithm
Byte Substitution

- Each byte ***a*** in the data block in the state matrix is substituted with another byte: $Sub(a) = Aa^{-1} + b$
- **Where**:

a^{-1}: is the multiplicative inverse of **a** in GF(2^8) mod ($X^8 + X^4 + X^3 + X + 1$)

$$A = \begin{pmatrix} 10001111 \\ 11000111 \\ 11100011 \\ 11110001 \\ 11111000 \\ 01111100 \\ 00111110 \\ 00011111 \end{pmatrix} \qquad b = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

(note: zero, which has no inverse, is set to zero).

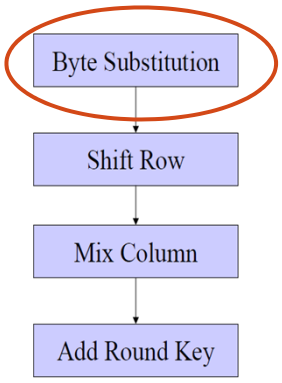


AES Encryption Algorithm
Byte Substitution

- This will generate the following S-box, which is represented here with hexadecimal notation:

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
10	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
20	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
30	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
40	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
50	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
60	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
70	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
80	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
90	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a0	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b0	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c0	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d0	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e0	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f0	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

- Here the column is determined by the least significant half byte, and the row is determined by the most significant half byte.



AES Encryption Algorithm
Byte Substitution Table

- **Example:**

a(binary) = **0111 1010**
a (hex) = 7A

Sub (7A) = DA = 1101 1010

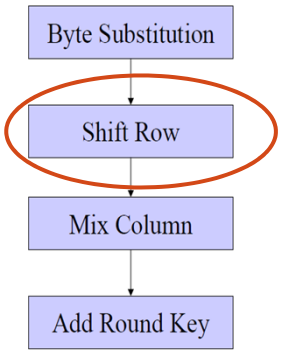
	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
00	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
10	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
20	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
30	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
40	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
50	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
60	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
70	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
80	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
90	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a0	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b0	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c0	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d0	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e0	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f0	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Sub (7A) = DA

AES Encryption Algorithm
Shift Row

- Cyclic shifts to the left with offsets of 0,1,2,3

$$\begin{pmatrix} c_{0,0} & c_{0,1} & c_{0,2} & c_{0,3} \\ c_{1,0} & c_{1,1} & c_{1,2} & c_{1,3} \\ c_{2,0} & c_{2,1} & c_{2,2} & c_{2,3} \\ c_{3,0} & c_{3,1} & c_{3,2} & c_{3,3} \end{pmatrix} = \begin{pmatrix} b_{0,0} & b_{0,1} & b_{0,2} & b_{0,3} \\ b_{1,1} & b_{1,2} & b_{1,3} & b_{1,0} \\ b_{2,2} & b_{2,3} & b_{2,0} & b_{2,1} \\ b_{3,3} & b_{3,0} & b_{3,1} & b_{3,2} \end{pmatrix}$$

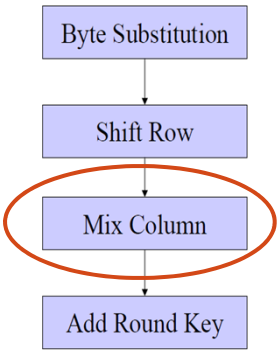


AES Encryption Algorithm
Mix Column

- We need to obtain the matrix d:

$$\begin{pmatrix} d_{0,0} & d_{0,1} & d_{0,2} & d_{0,3} \\ d_{1,0} & d_{1,1} & d_{1,2} & d_{1,3} \\ d_{2,0} & d_{2,1} & d_{2,2} & d_{2,3} \\ d_{3,0} & d_{3,1} & d_{3,2} & d_{3,3} \end{pmatrix} = \begin{pmatrix} '02' & '03' & '01' & '01' \\ '01' & '02' & '03' & '01' \\ '01' & '01' & '02' & '03' \\ '03' & '01' & '01' & '02' \end{pmatrix} * \begin{pmatrix} c_{0,0} & c_{0,1} & c_{0,2} & c_{0,3} \\ c_{1,0} & c_{1,1} & c_{1,2} & c_{1,3} \\ c_{2,0} & c_{2,1} & c_{2,2} & c_{2,3} \\ c_{3,0} & c_{3,1} & c_{3,2} & c_{3,3} \end{pmatrix}$$

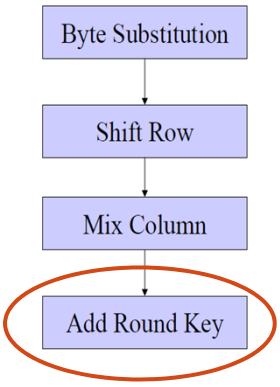
- Where:
  - **'01' = 00000001**
  - **'02' = 00000010**
  - **'03' = 00000011**
- All multiplications in this case are GF(2^8)
  This transformation is invertible



AES Encryption Algorithm
Add Round Key

- This is the final operation in each round:

$$\begin{pmatrix} e_{0,0} & e_{0,1} & e_{0,2} & e_{0,3} \\ e_{1,0} & e_{1,1} & e_{1,2} & e_{1,3} \\ e_{2,0} & e_{2,1} & e_{2,2} & e_{2,3} \\ e_{3,0} & e_{3,1} & e_{3,2} & e_{3,3} \end{pmatrix} = \begin{pmatrix} d_{0,0} & d_{0,1} & d_{0,2} & d_{0,3} \\ d_{1,0} & d_{1,1} & d_{1,2} & d_{1,3} \\ d_{2,0} & d_{2,1} & d_{2,2} & d_{2,3} \\ d_{3,0} & d_{3,1} & d_{3,2} & d_{3,3} \end{pmatrix} \oplus \begin{pmatrix} k_{0,0} & k_{0,1} & k_{0,2} & k_{0,3} \\ k_{1,0} & k_{1,1} & k_{1,2} & k_{1,3} \\ k_{2,0} & k_{2,1} & k_{2,2} & k_{2,3} \\ k_{3,0} & k_{3,1} & k_{3,2} & k_{3,3} \end{pmatrix}$$



AES Key Schedule

- Write original key as 4x4matrix with 4 columns: W(0), W(1), W(2), W(3).
- Key for round i is (W(4i), W(4i+1), W(4i+2), W(4i+3))

$$\underbrace{(w(0)\ w(1)\ w(2)\ w(3)}_{K_0}\ \underbrace{w(4)\ w(5)\ w(6)\ w(7)\ \ldots\ \ldots\ \ldots}_{K_1}\ \underbrace{w(40)\ w(41)\ w(42)\ w(43))}_{K_{10}}$$

Other columns defined recursively:

$$W(i) = W(i-4) \oplus \begin{cases} T(W(i-1)) & if i \mid 4 \\ W(i-1) & otherwise \end{cases}$$

$$W(i) = \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \xrightarrow{Shift\ and\ Sbox} \begin{pmatrix} e \\ f \\ g \\ h \end{pmatrix} \oplus \begin{pmatrix} r(i) \\ 0 \\ 0 \\ 0 \end{pmatrix} = T(W(i))$$

$$r(i) = (00000010)^{(i-4)/4}\ in\ GF(2^8)$$

Highly non-linear. Resists attacks at finding whole key when part is known

AES Encryption Algorithm
The Round Key

- **Original key 128 bits as 4 × 4 matrix of bytes**
- 4 columns W(0), W(1), W(2), W(3).
- Adjoin 40 columns W(4), . . . , W(43).
- **Round key for round- i consists of columns:**

(W(i),W(i +1),W(i + 2),W(i + 3)).

- **If i is a multiple of 4,** $W(i) = W(i-4) \oplus T(W(i-1))$

where T is a transformation of a, b, c, d in column *W(i - 1):*

1. **Shift cyclically** to get b, c, d, a.
2. **Replace each byte with S- box** entry using ByteSub, to get e, f, g, h.
3. Compute round constant r(i) = 00000010 $^{(i-4)/4}$ in *GF(2 ^{8}).*
4. *T(W(i - 1)) = (e ⊕ r(i), f, g, h)*

- If i is not a multiple of 4, $W(i) = W(i-4) \oplus W(i-1)$

AES Encryption Algorithm Animation

http://www.cs.bc.edu/~straubin/cs381-05/blockciphers/rijndael_ingles2004.swf

AES Software Implementation Options

Implementation	Code Size	Performance	Application
Pre-compute round functions (use look up table)	Largest	Fastest	High end machines (large powerful Servers)
Pre-Compute S-box only	Smaller	Slower	Desktop machine and tablets
No Pre-Computation	Smallest	Slowest	Smart Phones

AES Design Considerations

- Two rounds are sufficient to obtain full diffusion (obtained through substitution, Mix Column, ShiftRow).
- The number of rounds was chosen to be 10 because **there are attacks that are better than brute force up to six rounds.**
- The Key Schedule involves nonlinear mixing of the key bits, designed to resist attacks where the cryptanalyst knows part of the key and tries to deduce the remaining bits. Also, it aims to ensure that two distinct keys do not have a large number of round keys in common.
- AES Full Standard can be found on:

  http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

AES vs. DES

	DES	**AES**
Date	1976	1999
Block size	64	128
Key length	56	128, 192, 256
Number of rounds	16	10,12,14
Encryption primitives	Substitution, permutation	Substitution, shift, bit mixing
Cryptographic Operators	Confusion, diffusion	Confusion, diffusion
Design	Open	Open
Design rationale	Closed	Open
Selection process	Secret	Secret, but accept open public comment
Source	IBM, enhanced by NSA	Independent cryptographers

B. Halak, ECS, Southampton University

Learning Outcomes

At the end of this unit you should be able to:

1. Discuss the basic principles of discrete probability and information theory
2. Describe the principle of Shannon perfect secrecy
3. Verify the security of traditional ciphers
4. Explain how to construct a secure block cipher.
5. Outline the basic operation principles of 3DES.
6. Encrypt and Decrypt using Advance Encryption Standard Algorithm(AES).
7. Describe the Cryptographic Modes of Block Ciphers.
8. Describe a number of cryptographic attacks on block ciphers

How to use a block cipher securely?

- Block ciphers encrypt fixed size blocks
  - E.g. DES encrypts 64-bit blocks
- We need some way to encrypt arbitrary amounts of data
  - E.g. a message of 1000 bytes

Recall: Semantic Security of PRP(one-time key)

- **Definition**: E is **semantically secure** if for all "efficient" A
  Adv[A,E] is negligible. Where:

$$\boldsymbol{Adv[A,E] = |EXP_A[E(k,m1)] - EXP_A[E(k,m2)]|}$$

for all explicit m_0 , $m_1 \in M$ $\big(lengh(m1) = length(m2)\big)$

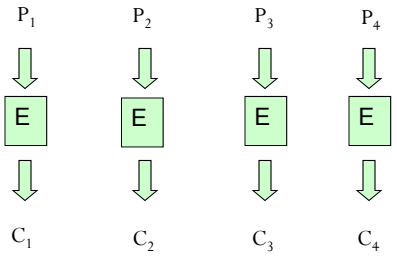
EXP the experiment that an adversary performs on encrypted messages.

In other words: the attacker cannot distinguish between encrypted messages.

How not to use a block cipher?

- The plaintext is broken into blocks, P_1, P_2, P_3, …
- Each block is encrypted independently of the other blocks

  $C_i = E_K(P_i)$
- For a given key, this mode behaves like we have a gigantic codebook, in which each plaintext block has an entry, hence the name **Electronic Code Book**



Example:

- Consider the two messages:

m1= "hey hey", m2= "hey you" which are encrypted using ECB mode: $\mathrm{E}(\boldsymbol{k}, \boldsymbol{m}) = \{\mathrm{c1}, \mathrm{c2}\}$.

Assume we have an attacker with the following experiment:

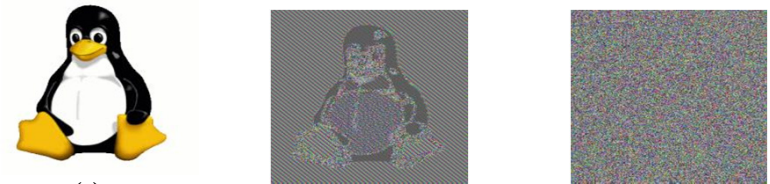
$$\boldsymbol{EXP_A[E(k,m) = if\ c1 = c2\ output\ 0,\quad else\ output\ 1]}$$

What is the advantage of the attacker?

$$\boldsymbol{Adv[A,E] = |EXP_A[E(k,m1)] - EXP_A[E(k,m2)]| = ?}$$

How not to use a block cipher?

- **Electronic Code Book Weakness:**
  - Repetitive information contained in the plaintext may show in the ciphertext, if aligned with blocks.
  - If the same message is encrypted (with the same key) and sent twice, their cipher texts are the same. So an attacker can learn that two encrypted files are the same, two encrypted packets are the same, etc. This may lead to significant attacks when message space M is small
  - You can only use ECB securely if the message length is equal to the data block length (e.g. 128 bit for AES) and you change the key for each message.



(a) (b) (c)

Figure1: (a) Original picture (b) ECB Encrypted (c) Encrypted using another method (Wikipedia)

How to use a block cipher securely without changing the key?

- When the key is used to encrypt many block , the adversary can see many ciphertexts with same key.
- This may make the system prone to attacks such as chosen-plaintext attack (CPA) where in the attacker can obtain the encryption of arbitrary messages of his choice and deduce the secret key (there are many variations of such an attack).
- In both DES and AES one pair of (c, m) is enough to find the key through exhaustive search

Semantic Security of PRP(many-time key)

- Definition: E is semantically secure under CPA if for all "efficient" Attackers
  Adv[A,E] is negligible. Where:

$$Adv[A,E] = |EXP_A[E(k,m1)] - EXP_A[E(k,m2)]|$$

  EXP is an experiment that an adversary performs on encrypted messages, to deduce their values

- *Adv*[A, E] represents the attacker ability to distinguish between the encryption of different messagesfor all explicit m_0 , $m_1 \in M$ $\left(lengh(m1) = length(m2)\right)$

- In other words: An encryption function E can only be semantically secure if the attackers cannot distinguish between encrypted messages, in which case whatever experiment they carry their advantage will be negligible (Adv(A, E)≈ 0)

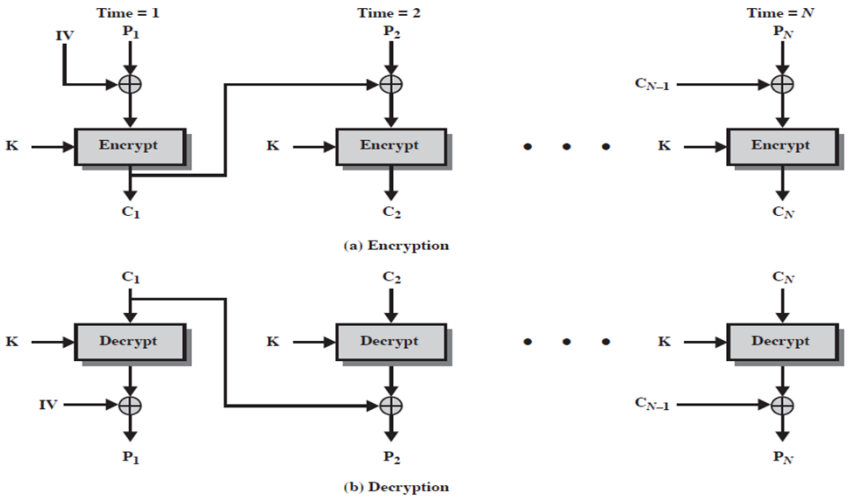
- One way to achieve this is to ensure that the encryption algorithm produce different outputs for the same message, for each encryption round

How to use a block cipher securely?

- If secret key is to be used multiple times than given the same plaintext message twice, encryption must produce different outputs. There are a number of methods which are approved by NIST:
  1. Cipher block chaining mode (CBC)
  2. Counter mode (CTR)
  3. Cipher feedback mode (CFB)
  4. Output feedback mode (OFB)
- For full list of all approved block cipher modes and to submit your own mode:

http://csrc.nist.gov/groups/ST/toolkit/BCM/index.html

Cipher Block Chaining (CBC)



Cipher Block Chaining (CBC)

- The plaintext is broken into blocks: $P_1, P_2, P_3, \ldots$
- Each plaintext block is XORed (chained) with the previous ciphertext block before encryption (hence the name):

$$C_i = \mathrm{E}_K(C_{i-1} \oplus P_i)$$

$$C_0 = \mathrm{IV}$$

- Use an Initial Vector (IV) to start the process.
- Decryption : $P_i = C_{i-1} \oplus \mathrm{D}_K(C_i)$
- Application : general block-oriented transmission.

Remarks on CBC

1. The same key is used for all blocks (prone to CPA).
2. Errors propagate.
3. Initialization Vector (IV) must be known to both the sender & receiver. Typically, IV is either a fixed value or is sent encrypted in ECB mode before the rest of message.
4. Serial processing (slow!)

Remarks on CBC

Theorem:
Assuming E is a secure encryption over (K,X), E_{CBC} is the CBC mode implementation of E. The upper bound o the advantage of the advantage of the attackers using chosen plain text attacks of E_{CBC} is given as follows.

$$\mathbf{Adv_{CPA}[A, E_{CBC}] \leq 2 \cdot Adv[A, E] + 2\, q^2 L^2 / |X|}$$

For all messages with length L >0

Where:

Adv[A, E]: is the upper bound of the advantage of the attackers (A) of E

q: the number of messages encrypted using the same encryption key (K)

L: the length of the max message

|X|: the size of the plaintext set

This means E_{CBC} is a semantically secure. under CPA as long as

$\mathbf{q^2 L^2 << |X|}$

Remarks on CBC

$$\mathbf{Adv_{CPA}[A, E_{CBC}] \leq 2 \cdot Adv_{PRP}[A, E] + 2\, q^2 L^2 / |X|}$$

Suppose we want $Adv_{CPA}[A, E_{CBC}] \leq 1/2^{40}$ and $Adv_{PRP}[A, E]$ is negligible

This means $q^2 L^2 / |X| < 1/ 2^{40}$

- For AES: $|X| = 2^{128} \Rightarrow q L < 2^{44}$

  So, after 2^{44} AES blocks, we must change key
- For 3DES: $|X| = 2^{64} \Rightarrow q L < 2^{12}$

  So, after 2^{12} 3DES blocks, we must change key

Counter Mode (CTR)

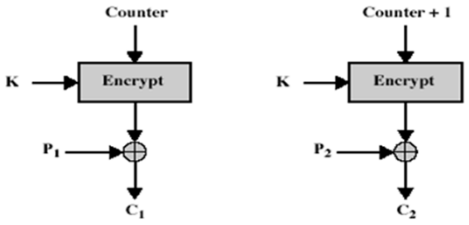
- A counter T is initialized to some IV and then incremented by 1 for each subsequent plaintext block.

- Encryption:

$T_1 = IV$

$T_i = T_{i-1} + 1$

$C_i = P_i \text{ XOR } E_K(T_i)$



Remark on CTR

Weaknesses:

1. The same key is used for all blocks (prone to CPA).
2. IV should not be reused.

Strengths:

1. Needs only the encryption algorithm
2. Fast encryption/decryption; blocks can be processed (encrypted or decrypted) in parallel; good for high speed links

Remarks on CTR

Theorem:

Assuming E is a secure encryption over (K,X), E_{CTR} is the CTR mode implementation of E. The upper bound o the advantage of the advantage of the attackers using chosen plain text attacks of E_{CTRC} is given as follows.

$$\mathbf{Adv_{CPA}\,[A, E_{CTR}] \leq 2 \cdot Adv\,[A, E] + q^2\,L\,/\,|X|}$$

For all messages with length L >0

Where:

Adv[A, E]: is the upper bound of the advantage of the attackers (A) of E

q: the number of messages encrypted using the same encryption key (K)

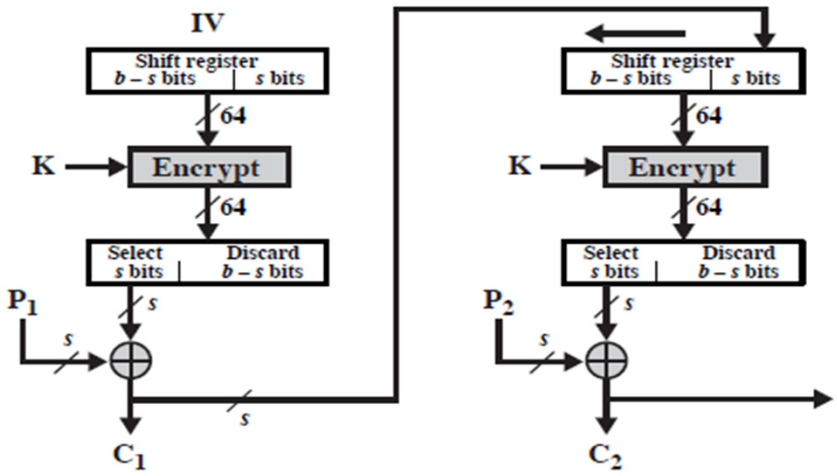
L: the length of the max message

|X|: the size of the plaintext set

This means E_{CTR} is a semantically secure. under CPA as long as

$\mathbf{q^2L << |X|}$

Cipher Feedback (CFB) Mode Encryption

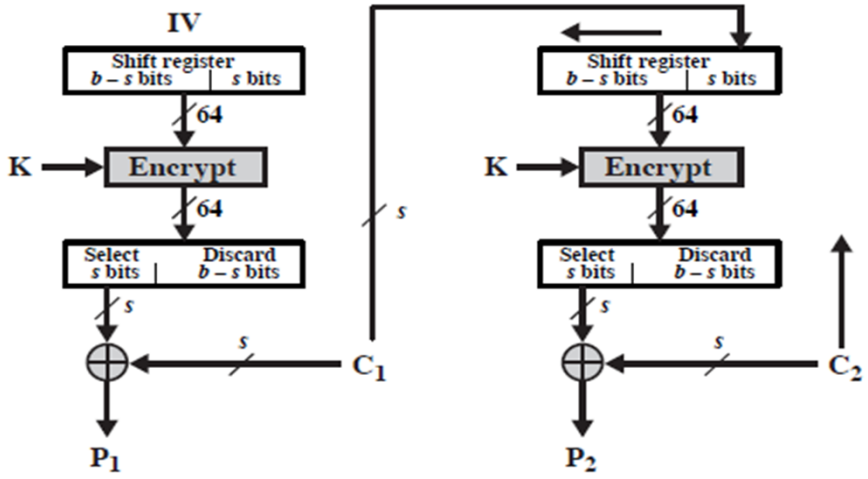


Cipher feedback mode

- Plaintext blocks: $p_1, p_2, \ldots$
- Key: k
- Basic idea: construct key stream $k_1, k_2, k_3, \ldots$
- Encryption:

$$\begin{cases} c_0 = \text{IV} \\ k_i = E_k(c_{i-1}), \quad \text{for } i \geq 1 \\ c_i = p_i \oplus k_i, \quad \text{for } i \geq 1 \end{cases}$$

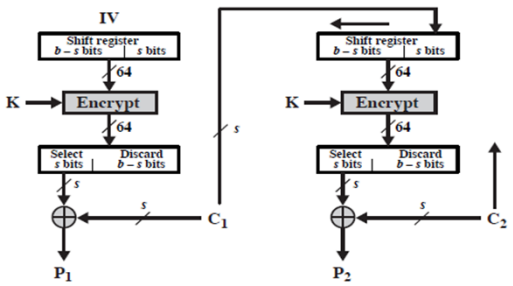
Decryption in CFB Mode



Decryption in CFB Mode

- Generate key stream $K_1, K_2, K_3, K_4, \ldots$ the same way as for encryption.
- Then decrypt each ciphertext segment as:

$$P_i = C_i \oplus K_i$$



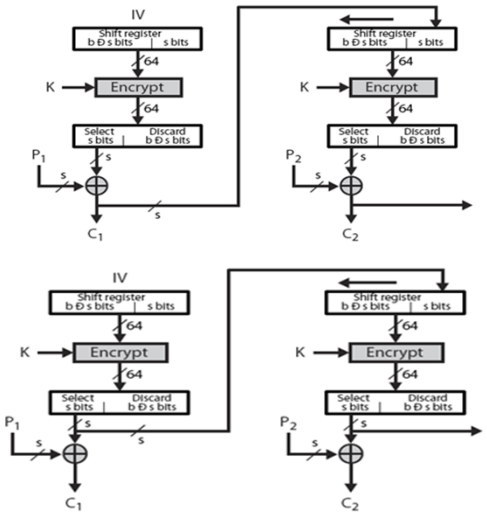
Remark on CFB

- The block cipher is used as a stream cipher.
- Appropriate when data arrives in bits/bytes.
- s can be any value; a common value is s = 8.
- A ciphertext segment depends on the current and all preceding plaintext segments.
- A corrupted ciphertext segment during transmission will affect the current and next several plaintext segments.

Output Feedback (OFB) Mode

Cipher Feedback

Output Feedback



Learning Outcomes

At the end of this unit you should be able to:

1. Discuss the basic principles of discrete probability and information theory
2. Describe the principle of Shannon perfect secrecy
3. Verify the security of traditional ciphers
4. Explain how to construct a secure block cipher.
5. Outline the basic operation principles of 3DES.
6. Encrypt and Decrypt using Advance Encryption Standard Algorithm(AES).
7. Describe the Cryptographic Modes of Block Ciphers.
8. Describe a number of cryptographic attacks on block ciphers

Types of Attacks

The objective of the following attacks is to systematically recover plaintext from ciphertext, or even more drastically, to deduce the decryption key.

A ciphertext-only attack

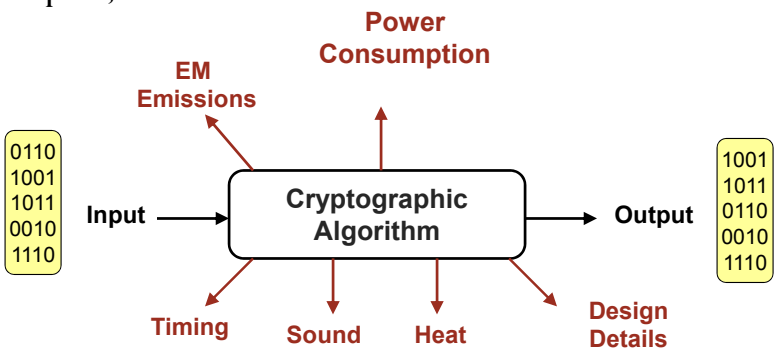
- It is one where the adversary (or cryptanalyst) tries to deduce the decryption key or plaintext by only observing ciphertext. Any encryption scheme vulnerable to this type of attack is considered to be completely insecure (e.g. frequency analysis on a shift cipher).

A known-plaintext attack

- It is one where the adversary has a quantity of plaintext and corresponding ciphertext. This type of attack is typically only marginally more difficult to mount. (e.g. meet in the middle attack on DES)
- Another variation of this is : chosen-plaintext attack where the adversary chooses plaintext and is then given corresponding ciphertext. Subsequently, the adversary uses any information deduced in order to recover plaintext corresponding to previously unseen cipher text.

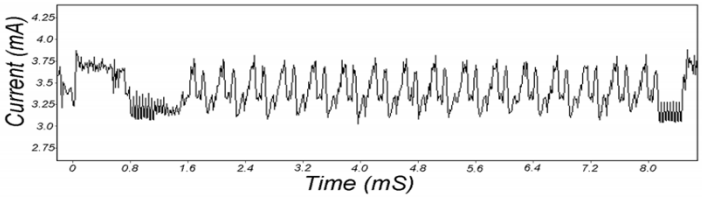
Side Channel Attacks

- **These are based on the analysis of information leaked from the physical implementation of the system such as electromagnetic radiation, power consumption, sounds..**



Side Channel Attacks

- **Example:** Power Analysis Attack which exploits the correlation between of power consumption of the cipher hardware and the data being processed.



Power Consumption Trace of DES [Kocher, Jaffe, Jun, 1998]

Quantum attacks

- Generic search problem:
  Let $f: X \longrightarrow \{0,1\}$ be a function.
  Goal: find $x \in X$ s.t. $f(x)=1$.
- Classical computer: best generic algorithm time $= O(|X|)$
- Quantum computer [Grover '96] : time $= O(|X|^{1/2})$
- Can quantum computers be built: latest attempt(DWave) has failed (see link below):

http://arstechnica.com/science/2014/01/dwaves-updated-quantum-optimizer-gets-beaten-by-a-classical-computer/

Quantum attacks

Given $m, c=E(k, m)$ define

$$f(k) = \begin{cases} 1 & \text{if } E(k, m) = c \\ 0 & \text{otherwise} \end{cases}$$

Grover $\Rightarrow$ quantum computer can find k in time $O(|K|^{1/2})$

DES: time $\approx 2^{28}$, AES-128: time $\approx 2^{64}$

quantum computer $\Rightarrow$ 256-bits key ciphers (e.g. AES-256)

B. Halak, ECS, Southampton University