CRYPTOGRAPHY 1

Nineth Lecture –

AES Example

Assistant Professor Dr.

Sufyan Salim Mahmood

2024 - 2025

AES Example - Input (128 bit key and message)

Key in English: Thats my Kung Fu (16 ASCII characters, 1 byte each)

Translation into Hex:

T	h	a	t	s		m	y		K	u	n	g		F	u
54	68	61	74	73	20	6D	79	20	4B	75	6E	67	20	46	75

Key in Hex (128 bits): 54 68 61 74 73 20 6D 79 20 4B 75 6E 67 20 46 75

Plaintext in English: Two One Nine Two (16 ASCII characters, 1 byte each)

Translation into Hex:

T	w	o		O	n	e		N	i	n	e		T	w	o
54	77	6F	20	4F	6E	65	20	4E	69	6E	65	20	54	77	6F

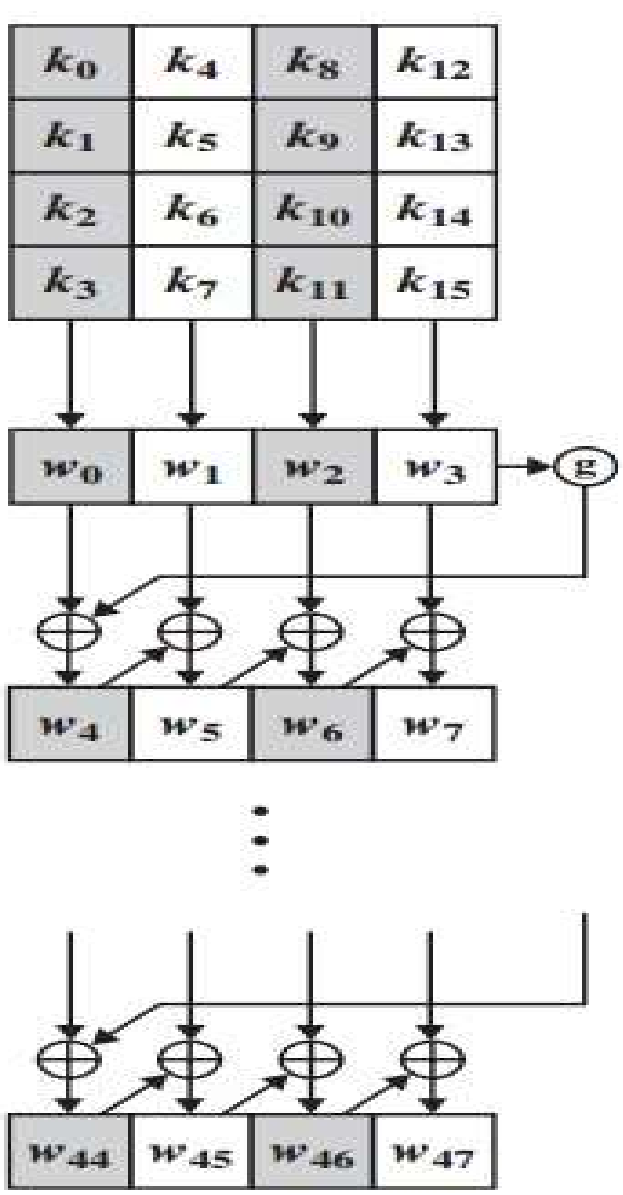
Plaintext in Hex (128 bits): 54 77 6F 20 4F 6E 65 20 4E 69 6E 65 20 54 77 6F

AES Example - The first Roundkey

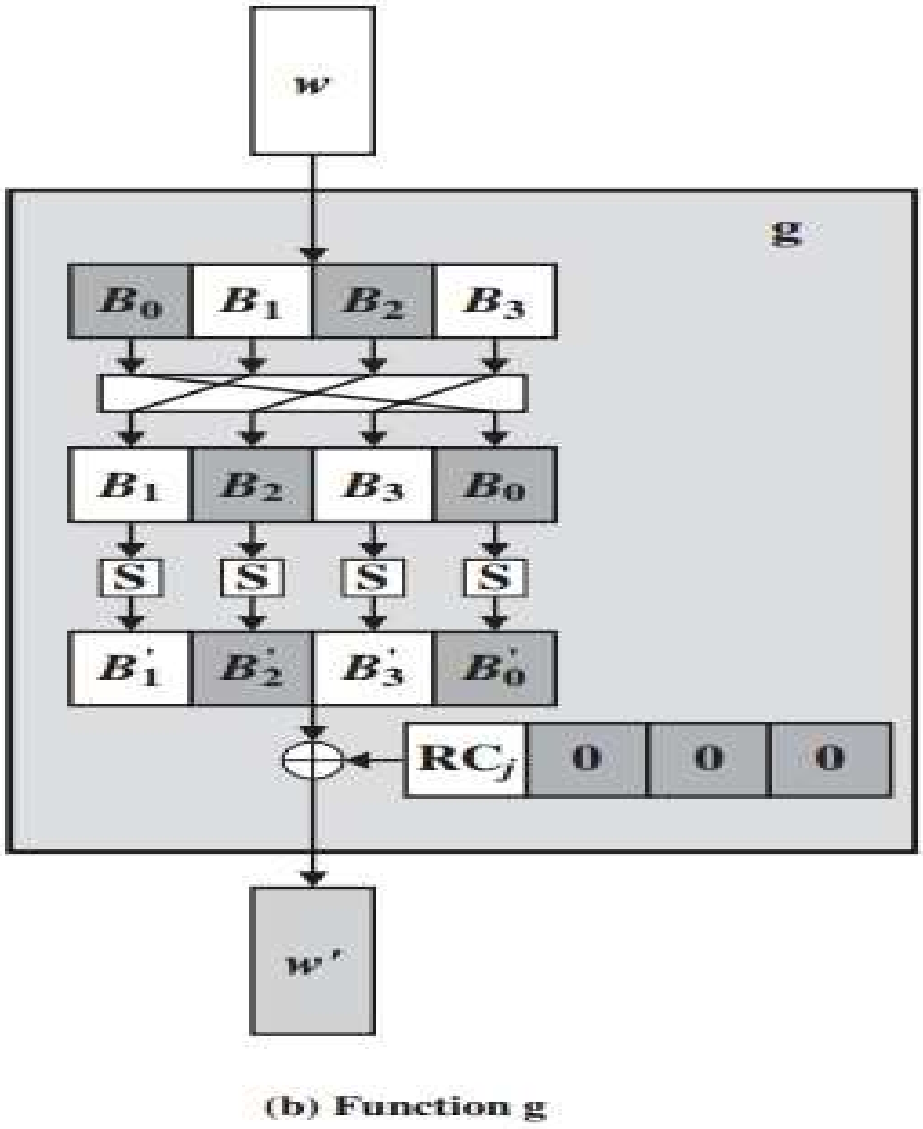
- Key in Hex (128 bits): 54 68 61 74 73 20 6D 79 20 4B 75 6E 67 20 46 75
- $w[0] = (54, 68, 61, 74), w[1] = (73, 20, 6D, 79), w[2] = (20, 4B, 75, 6E), w[3] = (67, 20, 46, 75)$
- $g(w[3])$:
  - circular byte left shift of $w[3]$: $(20, 46, 75, 67)$
  - Byte Substitution (S-Box): $(B7, 5A, 9D, 85)$
  - Adding round constant $(01, 00, 00, 00)$ gives: $g(w[3]) = (B6, 5A, 9D, 85)$
- $w[4] = w[0] \oplus g(w[3]) = (E2, 32, FC, F1)$:

0101 0100 1011 0110	0110 1000 0101 1010	0110 0001 1001 1101	0111 0100 1000 0101
1110 0010 E2	0011 0010 32	1111 1100 FC	1111 0001 F1

- $w[5] = w[4] \oplus w[1] = (91, 12, 91, 88), w[6] = w[5] \oplus w[2] = (B1, 59, E4, E6),$
  $w[7] = w[6] \oplus w[3] = (D6, 79, A2, 93)$
- first roundkey: E2 32 FC F1 91 12 91 88 B1 59 E4 E6 D6 79 A2 93



(a) Overall algorithm



(b) Function g

Figure 5.9 AES Key Expansion

AES Example - All RoundKeys

- Round 0: 54 68 61 74 73 20 6D 79 20 4B 75 6E 67 20 46 75
- Round 1: E2 32 FC F1 91 12 91 88 B1 59 E4 E6 D6 79 A2 93
- Round 2: 56 08 20 07 C7 1A B1 8F 76 43 55 69 A0 3A F7 FA
- Round 3: D2 60 0D E7 15 7A BC 68 63 39 E9 01 C3 03 1E FB
- Round 4: A1 12 02 C9 B4 68 BE A1 D7 51 57 A0 14 52 49 5B
- Round 5: B1 29 3B 33 05 41 85 92 D2 10 D2 32 C6 42 9B 69
- Round 6: BD 3D C2 B7 B8 7C 47 15 6A 6C 95 27 AC 2E 0E 4E
- Round 7: CC 96 ED 16 74 EA AA 03 1E 86 3F 24 B2 A8 31 6A
- Round 8: 8E 51 EF 21 FA BB 45 22 E4 3D 7A 06 56 95 4B 6C
- Round 9: BF E2 BF 90 45 59 FA B2 A1 64 80 B4 F7 F1 CB D8
- Round 10: 28 FD DE F8 6D A4 24 4A CC C0 A4 FE 3B 31 6F 26

AES Example - Add Roundkey, Round 0

- State Matrix and Roundkey No.0 Matrix:

$$\begin{pmatrix} 54 & 4F & 4E & 20 \\ 77 & 6E & 69 & 54 \\ 6F & 65 & 6E & 77 \\ 20 & 20 & 65 & 6F \end{pmatrix} \qquad \begin{pmatrix} 54 & 73 & 20 & 67 \\ 68 & 20 & 4B & 20 \\ 61 & 6D & 75 & 46 \\ 74 & 79 & 6E & 75 \end{pmatrix}$$

- XOR the corresponding entries, e.g., $69 \oplus 4B = 22$

$$\begin{array}{r} 0110\ 1001 \\ 0100\ 1011 \\ \hline 0010\ 0010 \end{array}$$

- the new State Matrix is

$$\begin{pmatrix} 00 & 3C & 6E & 47 \\ 1F & 4E & 22 & 74 \\ 0E & 08 & 1B & 31 \\ 54 & 59 & 0B & 1A \end{pmatrix}$$

AES Example - Round 1, Substitution Bytes

- current State Matrix is

$$\begin{pmatrix} 00 & 3C & 6E & 47 \\ 1F & 4E & 22 & 74 \\ 0E & 08 & 1B & 31 \\ 54 & 59 & 0B & 1A \end{pmatrix}$$

- substitute each entry (byte) of current state matrix by corresponding entry in AES S-Box
- for instance: byte 6E is substituted by entry of S-Box in row 6 and column E, i.e., by 9F
- this leads to new State Matrix

$$\begin{pmatrix} 63 & EB & 9F & A0 \\ C0 & 2F & 93 & 92 \\ AB & 30 & AF & C7 \\ 20 & CB & 2B & A2 \end{pmatrix}$$

- this non-linear layer is for resistance to differential and linear cryptanalysis attacks

AES Example - Round 1, Shift Row

- the current State Matrix is

$$\begin{pmatrix} 63 & EB & 9F & A0 \\ C0 & 2F & 93 & 92 \\ AB & 30 & AF & C7 \\ 20 & CB & 2B & A2 \end{pmatrix}$$

- four rows are shifted cyclically to the left by offsets of 0,1,2, and 3
- the new State Matrix is

$$\begin{pmatrix} 63 & EB & 9F & A0 \\ 2F & 93 & 92 & C0 \\ AF & C7 & AB & 30 \\ A2 & 20 & CB & 2B \end{pmatrix}$$

- this linear mixing step causes diffusion of the bits over multiple rounds

AES Example - Round 1, Mix Column

- Mix Column multiplies fixed matrix against current State Matrix:

$$\begin{pmatrix} 02\,03\,01\,01 \\ 01\,02\,03\,01 \\ 01\,01\,02\,03 \\ 03\,01\,01\,02 \end{pmatrix} \begin{pmatrix} 63 & EB & 9F & A0 \\ 2F & 93 & 92 & C0 \\ AF & C7 & AB & 30 \\ A2 & 20 & CB & 2B \end{pmatrix} = \begin{pmatrix} BA & 84 & E8 & 1B \\ 75 & A4 & 8D & 40 \\ F4 & 8D & 06 & 7D \\ 7A & 32 & 0E & 5D \end{pmatrix}$$

- entry BA is result of $(02 \bullet 63) \oplus (03 \bullet 2F) \oplus (01 \bullet AF) \oplus (01 \bullet A2)$:
  - $02 \bullet 63 = 00000010 \bullet 01100011 = 11000110$
  - $03 \bullet 2F = (02 \bullet 2F) \oplus 2F = (00000010 \bullet 00101111) \oplus 00101111 = 01110001$
  - $01 \bullet AF = AF = 10101111$ and $01 \bullet A2 = A2 = 10100010$
  - hence

$$\begin{array}{r} 11000110 \\ 01110001 \\ 10101111 \\ 10100010 \\ \hline 10111010 \end{array}$$

AES Example - Add Roundkey, Round 1

- State Matrix and Roundkey No.1 Matrix:

$$\begin{pmatrix} BA & 84 & E8 & 1B \\ 75 & A4 & 8D & 40 \\ F4 & 8D & 06 & 7D \\ 7A & 32 & 0E & 5D \end{pmatrix} \qquad \begin{pmatrix} E2 & 91 & B1 & D6 \\ 32 & 12 & 59 & 79 \\ FC & 91 & E4 & A2 \\ F1 & 88 & E6 & 93 \end{pmatrix}$$

- XOR yields new State Matrix

$$\begin{pmatrix} 58 & 15 & 59 & CD \\ 47 & B6 & D4 & 39 \\ 08 & 1C & E2 & DF \\ 8B & BA & E8 & CE \end{pmatrix}$$

- AES output after Round 1: 58 47 08 8B 15 B6 1C BA 59 D4 E2 E8 CD 39 DF CE

AES Example - Round 2

- after Substitute Byte and after Shift Rows:

$$\begin{pmatrix} 6A & 59 & CB & BD \\ A0 & 4E & 48 & 12 \\ 30 & 9C & 98 & 9E \\ 3D & F4 & 9B & 8B \end{pmatrix} \qquad \begin{pmatrix} 6A & 59 & CB & BD \\ 4E & 48 & 12 & A0 \\ 98 & 9E & 30 & 9B \\ 8B & 3D & F4 & 9B \end{pmatrix}$$

- after Mixcolumns and after Roundkey:

$$\begin{pmatrix} 15 & C9 & 7F & 9D \\ CE & 4D & 4B & C2 \\ 89 & 71 & BE & 88 \\ 65 & 47 & 97 & CD \end{pmatrix} \qquad \begin{pmatrix} 43 & 0E & 09 & 3D \\ C6 & 57 & 08 & F8 \\ A9 & C0 & EB & 7F \\ 62 & C8 & FE & 37 \end{pmatrix}$$

AES Example - Round 3

- after Substitute Byte and after Shift Rows:

$$\begin{pmatrix} 1A & AB & 01 & 27 \\ B4 & 5B & 30 & 41 \\ D3 & BA & E9 & D2 \\ AA & E8 & BB & 9A \end{pmatrix} \qquad \begin{pmatrix} 1A & AB & 01 & 27 \\ 5B & 30 & 41 & B4 \\ E9 & D2 & D3 & BA \\ A9 & AA & E8 & BB \end{pmatrix}$$

- after Mixcolumns and after Roundkey:

$$\begin{pmatrix} AA & 65 & FA & 88 \\ 16 & 0C & 05 & 3A \\ 3D & C1 & DE & 2A \\ B3 & 4B & 5A & 0A \end{pmatrix} \qquad \begin{pmatrix} 78 & 70 & 99 & 4B \\ 76 & 76 & 3C & 39 \\ 30 & 7D & 37 & 34 \\ 54 & 23 & 5B & F1 \end{pmatrix}$$

AES Example - Round 4

- after Substitute Byte and after Shift Rows:

$$\begin{pmatrix} BC & 51 & EE & B3 \\ 38 & 38 & EB & 12 \\ 04 & FF & 9A & 18 \\ 20 & 26 & 39 & A1 \end{pmatrix} \qquad \begin{pmatrix} BC & 51 & EE & B3 \\ 38 & EB & 12 & 38 \\ 9A & 18 & 04 & FF \\ A1 & 20 & 26 & 39 \end{pmatrix}$$

- after Mixcolumns and after Roundkey:

$$\begin{pmatrix} 10 & BC & D3 & F3 \\ D8 & 94 & E0 & E0 \\ 53 & EA & 9E & 25 \\ 24 & 40 & 73 & 7B \end{pmatrix} \qquad \begin{pmatrix} B1 & 08 & 04 & E7 \\ CA & FC & B1 & B2 \\ 51 & 54 & C9 & 6C \\ ED & E1 & D3 & 20 \end{pmatrix}$$

AES Example - Round 5

- after Substitute Byte and after Shift Rows:

$$\begin{pmatrix} C8 & 30 & F2 & 94 \\ 74 & B0 & C8 & 37 \\ D1 & 20 & DD & 50 \\ 55 & F8 & 66 & B7 \end{pmatrix} \qquad \begin{pmatrix} C8 & 30 & F2 & 94 \\ B0 & C8 & 37 & 74 \\ DD & 50 & D1 & 20 \\ B7 & 55 & F8 & 66 \end{pmatrix}$$

- after Mixcolumns and after Roundkey:

$$\begin{pmatrix} 2A & 26 & 8F & E9 \\ 78 & 1E & 0C & 7A \\ 1B & A7 & 6F & 0A \\ 5B & 62 & 00 & 3F \end{pmatrix} \qquad \begin{pmatrix} 9B & 23 & 5D & 2F \\ 51 & 5F & 1C & 38 \\ 20 & 22 & BD & 91 \\ 68 & F0 & 32 & 56 \end{pmatrix}$$

AES Example - Round 6

- after Substitute Byte and after Shift Rows:

$$\begin{pmatrix} 14 & 26 & 4C & 15 \\ D1 & CF & 9C & 07 \\ B7 & 93 & 7A & 81 \\ 45 & 8C & 23 & B1 \end{pmatrix} \qquad \begin{pmatrix} 14 & 26 & 4C & 15 \\ CF & 9C & 07 & D1 \\ 7A & 81 & B7 & 93 \\ B1 & 45 & 8C & 23 \end{pmatrix}$$

- after Mixcolumns and after Roundkey:

$$\begin{pmatrix} A9 & 37 & AA & F2 \\ AE & D8 & 0C & 21 \\ E7 & 6C & B1 & 9C \\ F0 & FD & 67 & 3B \end{pmatrix} \qquad \begin{pmatrix} 14 & 8F & C0 & 5E \\ 93 & A4 & 60 & 0F \\ 25 & 2B & 24 & 92 \\ 77 & E8 & 40 & 75 \end{pmatrix}$$

AES Example - Round 7

- after Substitute Byte and after Shift Rows:

$$\begin{pmatrix} FA & 73 & BA & 58 \\ DC & 49 & D0 & 76 \\ 3F & F1 & 36 & 4F \\ F5 & 9B & 09 & 9D \end{pmatrix} \qquad \begin{pmatrix} FA & 73 & BA & 58 \\ 49 & D0 & 76 & DC \\ 36 & 4F & 3F & F1 \\ 9D & F5 & 9B & 09 \end{pmatrix}$$

- after Mixcolumns and after Roundkey:

$$\begin{pmatrix} 9F & 37 & 51 & 37 \\ AF & EC & 8C & FA \\ 63 & 39 & 04 & 66 \\ 4B & FB & B1 & D7 \end{pmatrix} \qquad \begin{pmatrix} 53 & 43 & 4F & 85 \\ 39 & 06 & 0A & 52 \\ 8E & 93 & 3B & 57 \\ 5D & F8 & 95 & BD \end{pmatrix}$$

AES Example - Round 8

- after Substitute Byte and after Shift Rows:

$$\begin{pmatrix} ED & 1A & 84 & 97 \\ 12 & 6F & 67 & 00 \\ 19 & DC & E2 & 5B \\ 4C & 41 & 2A & 7A \end{pmatrix} \qquad \begin{pmatrix} ED & 1A & 84 & 97 \\ 6F & 67 & 00 & 12 \\ E2 & 5B & 19 & DC \\ 7A & 4C & 41 & 2A \end{pmatrix}$$

- after Mixcolumns and after Roundkey:

$$\begin{pmatrix} E8 & 8A & 4B & F5 \\ 74 & 75 & EE & E6 \\ D3 & 1F & 75 & 58 \\ 55 & 8A & 0C & 38 \end{pmatrix} \qquad \begin{pmatrix} 66 & 70 & AF & A3 \\ 25 & CE & D3 & 73 \\ 3C & 5A & 0F & 13 \\ 74 & A8 & 0A & 54 \end{pmatrix}$$

AES Example - Round 9

- after Substitute Byte and after Shift Rows:

$$\begin{pmatrix} 33 & 51 & 79 & 0A \\ 3F & 8B & 66 & 8F \\ EB & BE & 76 & 7D \\ 92 & C2 & 67 & 20 \end{pmatrix} \qquad \begin{pmatrix} 33 & 51 & 79 & 0A \\ 8B & 66 & 8F & 3F \\ 76 & 7D & EB & BE \\ 20 & 92 & C2 & 67 \end{pmatrix}$$

- after Mixcolumns and after Roundkey:

$$\begin{pmatrix} B6 & E7 & 51 & 8C \\ 84 & 88 & 98 & CA \\ 34 & 60 & 66 & FB \\ E8 & D7 & 70 & 51 \end{pmatrix} \qquad \begin{pmatrix} 09 & A2 & F0 & 7B \\ 66 & D1 & FC & 3B \\ 8B & 9A & E6 & 30 \\ 78 & 65 & C4 & 89 \end{pmatrix}$$

AES Example - Round 10

- after Substitute Byte and after Shift Rows:

$$\begin{pmatrix} 01 & 3A & 8C & 21 \\ 33 & 3E & B0 & E2 \\ 3D & B8 & 8E & 04 \\ BC & 4D & 1C & A7 \end{pmatrix} \qquad \begin{pmatrix} 01 & 3A & 8C & 21 \\ 3E & B0 & E2 & 33 \\ 8E & 04 & 3D & B8 \\ A7 & BC & 4D & 1C \end{pmatrix}$$

- after Roundkey (Attention: no Mix columns in last round):

$$\begin{pmatrix} 29 & 57 & 40 & 1A \\ C3 & 14 & 22 & 02 \\ 50 & 20 & 99 & D7 \\ 5F & F6 & B3 & 3A \end{pmatrix}$$

- ciphertext: 29 C3 50 5F 57 14 20 F6 40 22 99 B3 1A 02 D7 3A

CRYPTOGRAPHY 1

Tenth Lecture –

Lightweight Algorithm

Assistant Professor Dr.

Sufyan Salim Mahmood

2024 - 2025

Introduction

- One of the emerging areas of cryptography in the last many years is lightweight cryptography. It is the scaled down version of traditional cryptography.

Introduction

- In fact, it is a combination of two fields: cryptography and hardware technology. Lightweight cryptography has assumed a greater significance with the evolution of a large number of small ubiquitous computing devices e.g. Radio-frequency IDentification devices (RFIDs).

Introduction

- These devices are resource constrained so it is not possible for them to run traditional cryptographic algorithms which require a large memory and greater processing power than these devices.
- Accordingly, there is need for innovative design of the cipher to suit the need of these devices.

Introduction

- Generally, the number of rounds is kept higher than a traditional block cipher to compensate for the simplicity of the round function.
- Besides the constraint of the resources, the designer also has to keep in mind the most important requirement i.e. the security of the cipher.
- The other important factor of a lightweight algorithm is the cost and unit called a gate-equivalent (GE).

Introduction

- Regarding the criteria, there are many criteria used by the authors of lightweight algorithms in order to achieve the required balance

Introduction

- Decrease of the main algorithm's parameters: block size, key length (within reasonable limits) and the algorithm's internal state;
- Simplifying the layers of transformations by decreasing the ROM requirements by using 4 × 4 S-boxes;
- Using low-cost (in implementation) but effective elements, such as data-dependent bit permutations, shift registers etc.;

Introduction

There are many lightweight block cipher algorithms such as:

- PRINCE
- KLEIN
- LED algorithm
- Lblock algorithm
- PRINT cipher
- PRESENT
- HIGHT

KLEIN Algorithm

Zheng Gong and his colleagues proposed this algorithm in 2012. It was designed to work with constrained devices such as wireless sensors and RFID tags. Compared to the related proposals, KLEIN has an advantage in relation to software performance and also its hardware implementation is compact.

KLEIN Algorithm

KLEIN is a family of block ciphers, with a 64-bit block size and a variable key length - 64, 80 or 96-bits. According to the different key length, it is denoted as KLEIN-64/80/96, respectively. It is well-known that the key length and the block size are two important factors for a block cipher in the trade-off between security and performance.

KLEIN Algorithm

. The structure of the KLEIN algorithm is a SPN as shown in the next figure. This structure has been used by many block ciphers like the AES and PRESENT. The KLEIN has three sets for its number of rounds which is dependent on the key size. For example, KLEIN-64 has 12 rounds while KLEIN-80 has 16 rounds and KLEIN-96 has 20 rounds.

KLEIN Algorithm

$sk^1 \leftarrow$ KEY;
STATE $\leftarrow$ PLAINTEXT;
for $i = 1$ *to* N_R do
  $AddRoundKey$(STATE, sk^i);
  $SubNibbles$(STATE);
  $RotateNibbles$(STATE);
  $MixNibbles$(STATE);
  $sk^{i+1} = KeySchedule(sk^i, i)$;
end for
CIPHERTEXT $\leftarrow$ $AddRoundKey$(STATE, sk^{N_R+1});

KLEIN Algorithm

This algorithm uses a one 4-bit S-box and it is repeated 16 times. The KLEIN S-box is an involutive S-box as described in the next Table. By choosing an involutive S-box, it can save the implementation costs for its inverse.

Input	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Output	7	4	A	9	1	F	B	0	C	3	2	6	8	E	D	5

KLEIN Algorithm

The cost of this algorithm as follows:

- KLEIN 64 → 1981
- KLEIN 80 → 2097
- KLEIN 96 → 2213

LBLOCK Algorithm

Wu and his colleagues proposed this algorithm in 2011. This algorithm is like many other lightweight block ciphers and the structure of the LBlock is a Feistel network and the authors considered security and efficiency when they designed this algorithm.

LBLOCK Algorithm

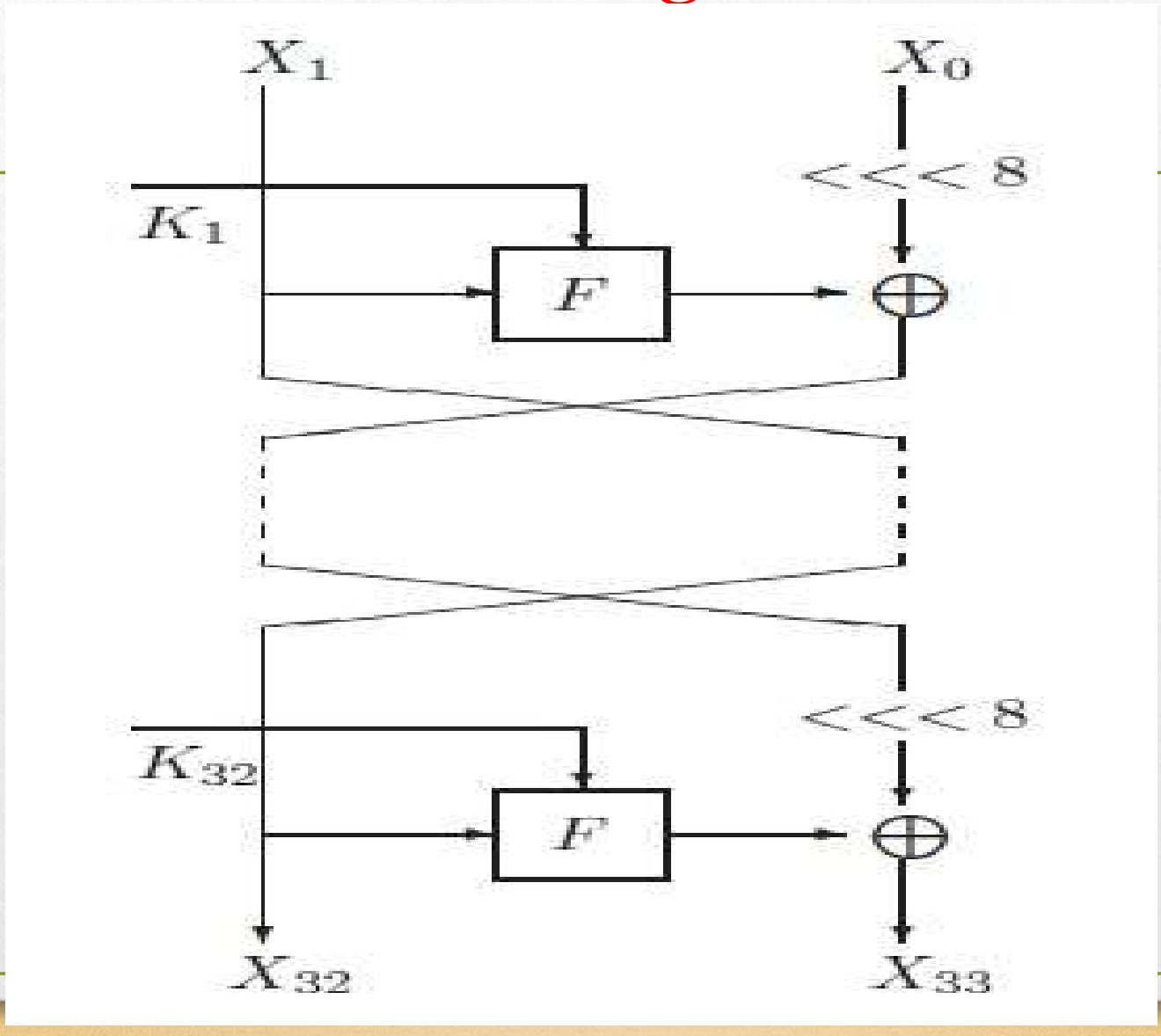
The LBlock has a 64-bit plaintext and the key size is 80-bits. It employs a variant Feistel structure and consists of 32 rounds. The block cipher 64-bit is divided into two parts, a left side 32-bit and right side 32-bit.

LBLOCK Algorithm

In the left side, there is a copy of 32-bit to make the right hand side. Another copy of 32-bit is XOR'ed with a 32-bit key and then as input to 8 S-boxes 4-bit. While in the right side, there is one operation called rotate left 8-bit for 32-bits and then XOR'ed with the output of 8 S-boxes in the left hand side to make the left hand side.

The same operation is repeated for 32 rounds as shown in the following figure.

LBLOCK Algorithm



LBLOCK Algorithm

This algorithm used 10 S-boxes as shown in the following table. There are 8 S-boxes out of 10 S-boxes used for encryption and decryption algorithm while 2 S-boxes only are used for key scheduling

LBLOCK Algorithm

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S0(x)	E	9	F	0	D	4	A	B	1	2	8	3	7	6	C	5
S1(x)	4	B	E	9	F	D	0	A	7	C	5	6	2	8	1	3
S2(x)	1	E	7	C	F	D	0	6	B	5	9	3	2	4	8	A
S3(x)	7	6	8	B	0	F	3	E	9	A	C	D	5	2	4	1
S4(x)	E	5	F	0	7	2	C	D	1	8	4	9	B	A	6	3
S5(x)	2	D	B	C	F	D	0	9	7	A	6	3	1	8	4	5
S6(x)	B	9	4	E	0	F	A	D	6	C	5	7	3	8	1	2
S7(x)	D	A	F	0	E	4	9	B	2	1	8	3	7	5	C	6
S8(x)	8	7	E	5	F	D	0	6	B	C	9	A	2	4	1	3
S9(x)	B	5	F	0	7	2	9	D	4	8	1	C	E	A	3	6

LBLOCK Algorithm

The cos of this algorithm is 1320 GE.

CRYPTOGRAPHY 1

Eleven Lecture –

S-box

Assistant Professor Dr.

Sufyan Salim Mahmood

2024 - 2025

Definition

- S-box substitution is a vital process that helps enhance security through nonlinear substitution in symmetric key encryption.

Definition

- S-Box substitution is used to enhance the security of encryption algorithms by introducing non-linearity in the substitution process, which helps bring about concepts such as confusion and diffusion making it harder for attackers to decode the encrypted data.

Definition

- The S-Box is an important part of symmetric key encryption algorithms. It's a substitution table used to do nonlinear substitutions in the course of encryption. Each byte of the input is substituted for another byte according to a fixed table which strengthens the confusion and diffusion properties of the encryption.

Definition

- In general, an S-box takes some number of input bits, *m*, and transforms them into some number of output bits, *n*, where *n* is not necessarily equal to *m*. Fixed tables are normally used, as in the Data Encryption Standard (DES), but in some ciphers the tables are generated dynamically from the key (e.g. the Blowfish and the Two fish encryption algorithms).

S-box Types

- 4-bit S-box.
- 6-bit S-box.
- 8-bit S-box.

4-bit S-box

- Most of the lightweight block cipher algorithms use this type of S-box.
- For example the PRESENT algorithm uses the same type of S-box.
- Also there is special type of 4-bit S-box called an involutive S-box. This special type is used by the KLIEN algorithm. The advantage of an involutive S-box is to save the cost on the decryption side (i.e. no need to generate the inverse S-box)

4-bit S-box

PRESENT S-box

X	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S(X)	C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2

KLIEN S-box

X	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
S(X)	7	4	A	9	1	F	B	0	C	3	2	6	8	E	D	5

6-bit S-box

- This type is used with a DES. This algorithm uses 8 S-boxes. There are 6-bits input, 2-bits to select the row and 4-bits to select the column. The output will be 4 bits. For example, an input 45 (101101) has outer bits "11"3 and inner bits "0110"6 and the corresponding output would be "1"

6-bit in, 4-bit out S-box

S_1		Middle bits															
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Outer bits	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	9	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

8-bit S-box

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

CRYPTOGRAPHY 1

Twelve Lecture –

P-box

Assistant Professor Dr.

Sufyan Salim Mahmood

2024 - 2025

Definition

- S-box substitution is a vital process that helps enhance security through nonlinear substitution in symmetric key encryption.

Permutation Box

P-boxes are permutation boxes which are usually one of the main components of a modern block cipher. They are also known as **D-boxes** or diffusion boxes.

A **p-box** (permutation box) is used to transposition the characters for the particular input of characters. In simple words, it transposes the bits. Here, 1,2,3,4,5 refers to the position of the bit, and the bits in those positions are transpositioned using the predefined p-box.

Permutation Box

For example, in the p-box, if the input is 1,2,3,4,5, the output might be 3,4,2,1,5. This means the values of 1,2,3,4,5 are being arranged in the order of 3,4,2,1,5.

Permutation Box

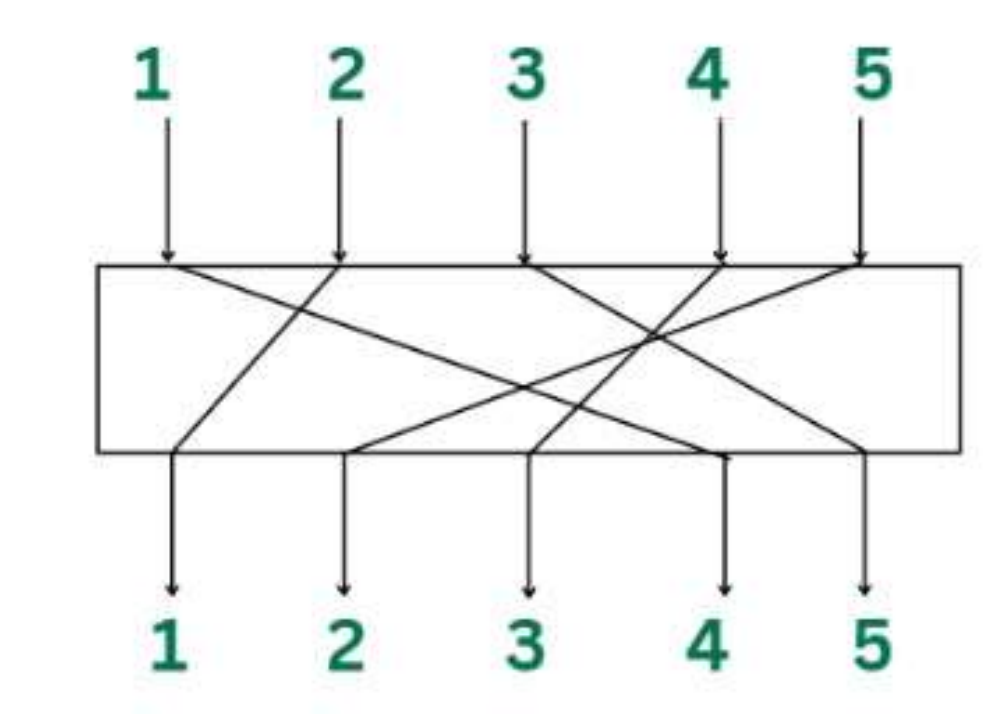
There are three types pf P-Box

- **Straight P-box**
- **Expansion P-box**
- **Compression P-box**

Straight P-box

In this type of p-box, the number of inputs and output is the same. If inputs are n and outputs are m, then **m=n**. The positions of the arrangement are shown in the figure, where an equal number of bits are transposed to different places.

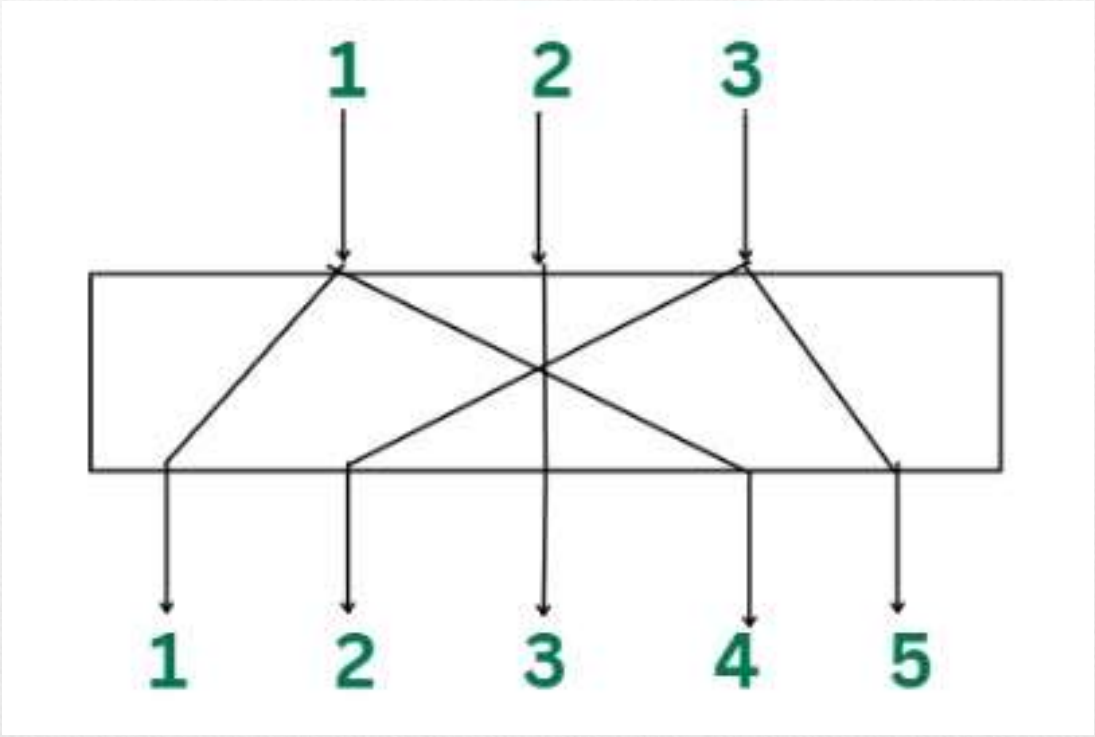
Straight P-box



Expansion P-box

An expansion p-box is a p-box with n inputs and m outputs where **m>n**, i.e., the number of outputs is more than the number of inputs. In this type of p-box, the values get repeated as for one input there is a possibility for more than one output.

Expansion P-box



Compression P-box

A compression p-box is a p-box with n inputs and m outputs where **m<n**, i.e., the number of outputs is less than the number of inputs. In this type of p-box, few bits are dropped as not all input bits are considered for output.

Compression P-box