บทที่ 3

การเข้ารหัสข้อมูล

3.1 มาตรฐานการเข้ารหัสข้อมูล (Data Cryptography)

ไม่ว่าคอมพิวเตอร์จะมีระบบความปลอดภัยอยู่ในระดับ C, B หรือ A ก็ตาม หากว่าข้อมูลที่เก็บอยู่ในเครื่องยังคงอยู่ในรูปข้อมูลปกติ หรือตัวอักษรธรรมดา (Plain Text) ความปลอดภัยของข้อมูลก็ยังต่ำมาก เนื่องจากเมื่อผู้อื่นรู้ชื่อผู้ใช้และรหัสผ่านที่ถูกต้องแล้ว ก็สามารถลักลอบนำข้อมูลที่เก็บไว้นี้ไปใช้ได้ทันที ซึ่งเท่ากับว่าความปลอดภัยของข้อมูลนี้ขึ้นอยู่กับรหัสผ่านเพียงอย่างเดียว ดังนั้นการเข้ารหัสข้อมูล หรือ Data Cryptography จึงเป็นอีกวิธีหนึ่งที่จะป้องกันไม่ให้ข้อมูลอันมีค่าของเราถูกผู้อื่นลักลอบนำไปใช้

การเข้ารหัสข้อมูล คือการนำข้อมูลปกติที่อยู่ในรูปของตัวอักษรธรรมดา มาผ่านการเข้ารหัสทำให้ได้ข้อมูลออกมาในอีกรูปแบบหนึ่ง ซึ่งจะไม่มีความหมาย อ่านไม่ออก และแปลกลับไปหาข้อมูลเดิมไม่ได้หากไม่มีรหัสลับที่ถูกต้อง ดังนั้นถึงแม้จะมีใครแอบก๊อปปี้ข้อมูลที่เข้ารหัสนี้ไปได้ แต่ถ้าไม่รู้รหัสลับก็จะไม่สามารถใช้ประโยชน์จากข้อมูลนั้นได้ ดังนั้นการเข้ารหัสข้อมูลจึงเป็นมาตรการสำหรับการรักษาความปลอดภัยของข้อมูล ที่เพิ่มขึ้นจากระบบความปลอดภัยของคอมพิวเตอร์ ซึ่งการเข้ารหัสข้อมูลจะแบ่งออกเป็น 2 ประเภท คือ Symmetric Cryptography และ Asymmetric Cryptography

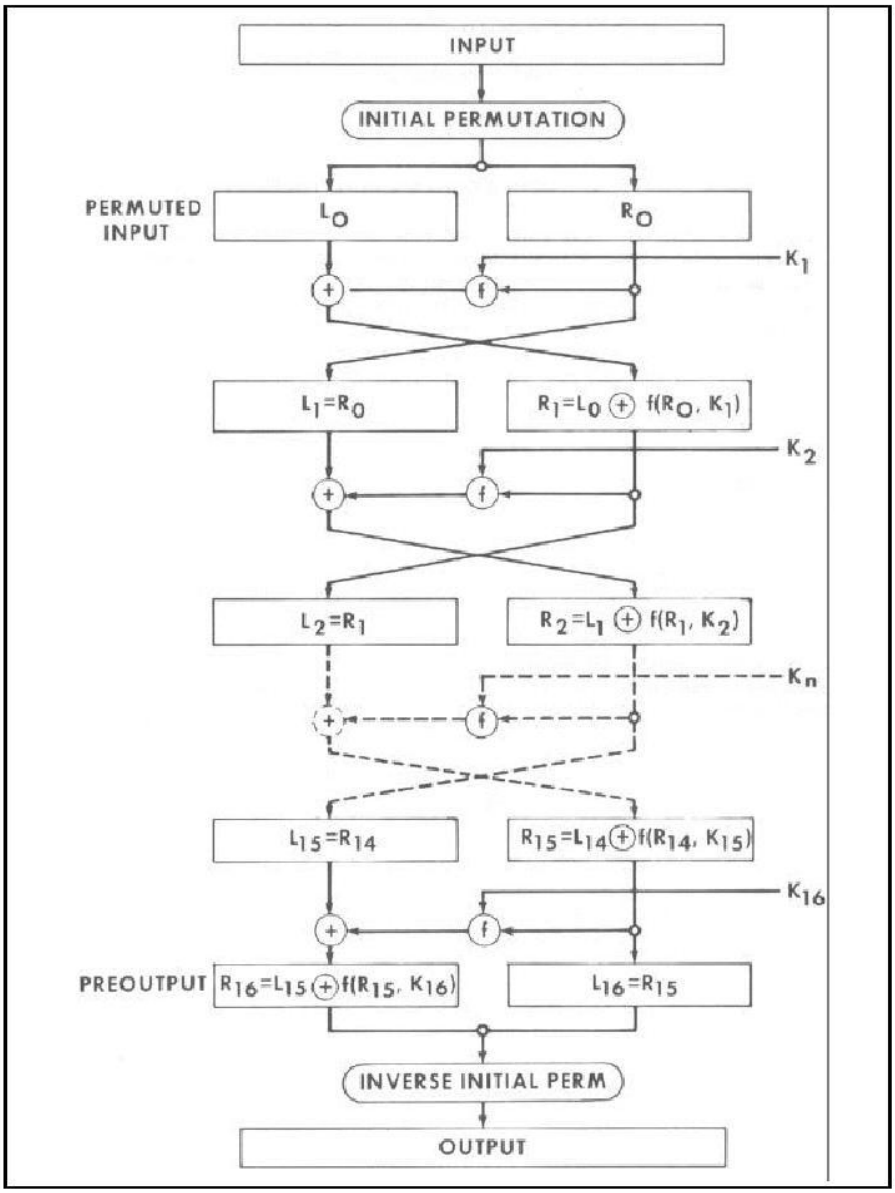
3.2 การเข้ารหัสแบบ DES (Data Encryption Standard)

DES คือ การเข้ารหัสข้อมูลโดยใช้อัลกอริทึมทางคณิตศาสตร์ โดยจะใช้ข้อมูลขนาด 64 บิต และกุญแจขนาด 64 บิต (โดย 8 บิตที่เพิ่มเข้ามา เพื่อนำมาใช้ตรวจสอบความผิดพลาด) ซึ่งในการเข้าและถอดรหัสนั้น ได้ใช้ระบบการเข้ารหัสแบบสมมาตร (Symmetric Cryptosystem) คือ ใช้กุญแจตัวเดียวกันในการเข้าและถอดรหัส สำหรับข้อมูลที่นำมาเข้ารหัสเรียกว่า เพลนเท็กซ์ (Plain Text) เมื่อผ่านการเข้ารหัสแล้ว จะเรียกข้อมูลชุดนั้นว่า ไซเฟอร์เท็กซ์ (Cipher Text) ซึ่งเป็นข้อมูลที่ไม่สามารถอ่านหรือตีความได้

ในกระบวนการถอดรหัส จะใช้กุญแจตัวเดียวกันกับที่ใช้ในการเข้ารหัส โดยจะทำกระบวนการย้อนกลับกับกระบวนการเข้ารหัส

การทำงานของอัลกอริทึม DES ได้แบ่งการทำงานออกเป็น 3 ส่วนที่สำคัญ คือ

1. การเรียงบิตใหม่ (Initial Permutation) เพื่อผลิต “permuted input” ซึ่งเป็นข้อมูลที่มีการสลับตำแหน่งจากเดิมไป
2. การเข้าฟังก์ชันการคำนวณ โดยมีกระบวนการทำงานแบบเดิมซ้ำกัน 16 ครั้ง ซึ่งรวมถึงการทำ และ ซึ่งผลลัพธ์ที่ได้ จะได้เป็นข้อมูลขนาด 64 บิตอยู่ชุดหนึ่งที่เรียกว่า PREOUTPUT
3. การเรียงบิตกลับ (Inverse Initial Permutation) ซึ่งจะได้ผลลัพธ์ออกมาเป็นข้อมูลที่ทำการเข้ารหัสแล้ว



รูปที่ 3-1 กระบวนการเข้ารหัสด้วย DES

3.2.1.การเรียงบิตใหม่ (Initial Permutation)

สำหรับขั้นตอนแรกของการเข้ารหัสข้อมูลแบบ DES นั้น เริ่มต้นโดยการสับเปลี่ยนตำแหน่งของบิต ซึ่งจะสับเปลี่ยนตำแหน่งของบิตตามตาราง

ในการใช้ตารางนั้น ตำแหน่งของตารางที่ปรากฏจะแทนตำแหน่งที่บิตนั้นๆอยู่หลังจากที่ทำการสลับตำแหน่งแล้วนั่นเอง โดยที่หมายเลขข้างในคือ หมายเลขที่แทนตำแหน่งข้อมูลก่อนการสลับตำแหน่ง

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	42	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

ตารางที่ 3-1 ค่าการสับเปลี่ยนตำแหน่งของบิตในตาราง IP (Initial Permutation)

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

ตารางที่ 3-2 ค่าการสับเปลี่ยนตำแหน่งของบิตในตาราง PC-1 (Permuted Choice 1)

ลำดับ	จำนวนครั้ง	ลำดับ	จำนวนครั้ง
1	1	9	1
2	1	10	2
3	2	11	2
4	2	12	2
5	2	13	2
6	2	14	2
7	2	15	2
8	2	16	1

ตารางที่ 3-3 จำนวนครั้งที่ทำการเลื่อนตำแหน่งในแต่ละรอบตาราง LS (Left Shift)

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

ตารางที่ 3-4 ค่าการสับเปลี่ยนตำแหน่งของบิตในตาราง PC-2 (Permuted Choice 2)

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

ตารางที่ 3-5 ค่าการสับเปลี่ยนตำแหน่งของบิตในตาราง E (Expansion)

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

ตารางที่ 3-6 ค่าการสับเปลี่ยนตำแหน่งของบิตในตาราง P (Permutation)

S_1

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

S_2

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

S_3

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

S_4

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

S_5

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

S_6

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

S_7

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

S_8

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

ตารางที่ 3-7 การหาค่าในตาราง S (Substitution)

ตาราง S นั้น จะมีอยู่ 8 ตาราง โดยข้อมูลที่นำมาเข้าในแต่ละตาราง S นั้น จะมีขนาด 6 บิต วิธีการใช้ตาราง S คือ นำค่าของบิตในตำแหน่งที่ 1 และ 6 มาเป็นค่าของส่วนแถว (Row) และค่าของบิตในตำแหน่งที่เหลือคือ 2, 3, 4 และ 5 นั้น นำมาเป็นค่าของส่วนคอลัมน์ (Column) แล้วนำค่าแถวและคอลัมน์มาเทียบในตาราง S ซึ่งจะได้ออกมาเป็นผลลัพธ์ของแต่ละตาราง S ที่มีขนาด 4 บิตออกมา

3.2.2. การเข้าฟังก์ชันการคำนวณ

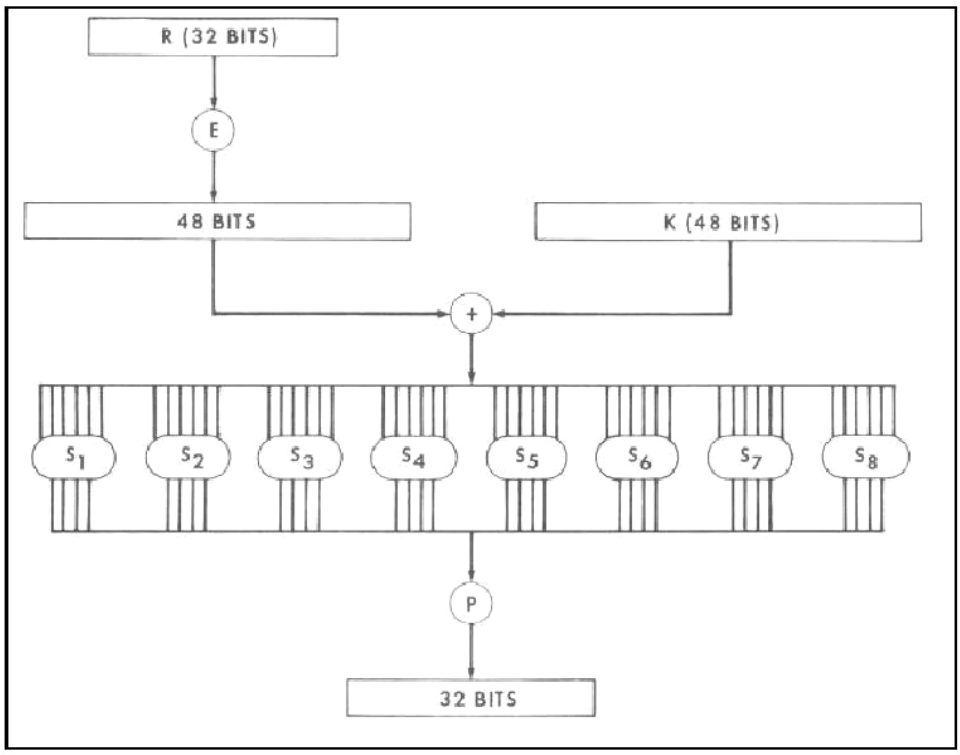
ในการเข้าฟังก์ชันการคำนวณ มีขั้นตอนดังนี้

1. ข้อมูลขนาด 64 บิต จะถูกแบ่งออกเป็น 2 ส่วน คือ ส่วนซ้าย L และส่วนขวา R โดยแต่ละส่วนจะมีขนาดเท่ากันคือ 32 บิต ซึ่งจะกำหนดค่าเริ่มต้นในแต่ละส่วนเป็น L0 และ R0
2. การทำงานของส่วน L และ R จะทำงานทั้งหมด 16 รอบการทำงาน โดยแต่ละรอบการทำงาน จะแทนด้วยตัว n ซึ่ง n จะมีค่าตั้งแต่ 1 ถึง 16 ซึ่งมีสูตรการคำนวณดังนี้

$$L0 = Rn-1$$

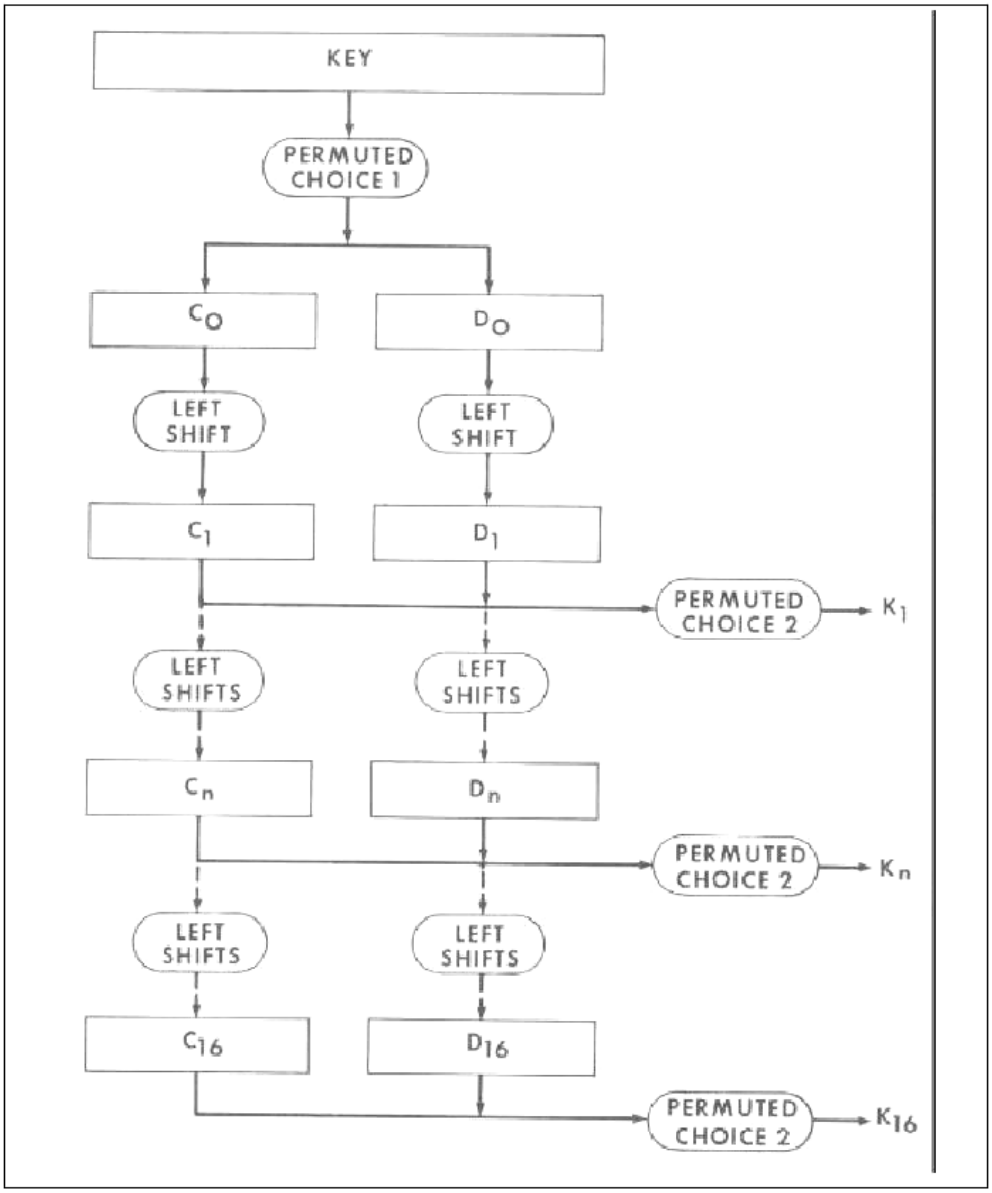
$$Rn = Ln \text{ XOR } f(Rn-1, Kn)$$

ในส่วนของฟังก์ชัน f จะเป็นการคำนวณเพื่อเพิ่มความยากของการเข้าและถอดรหัส ซึ่งจะถูกเรียกใช้คำนวณทั้งหมด 16 รอบ โดยในแต่ละรอบจะมีวิธีการคำนวณดังรูปที่ 2-8



รูปที่ 3-2 การทำงานในส่วนของฟังก์ชัน F

ในแต่ละรอบของกระบวนการเข้ารหัสข้อมูลด้วยกุญแจจำนวน 16 ชุดนั้น จะใช้ค่าของกุญแจแต่ละชุดที่มีขนาด 48 บิต โดยที่ค่าของกุญแจทั้ง 16 ชุดได้มาจากกุญแจที่เป็นข้อมูลขนาด 64 บิต โดยบิตตำแหน่งที่ 8, 16, …, 64 เป็นบิตที่ใช้ในการตรวจสอบความผิดพลาด



รูปที่ 3-3 การคำนวณในส่วนของหมายเลขกุญแจ

ในการทำการสับเปลี่ยนตำแหน่ง Permuted Choice 1 ตามตาราง PC-1 (ตารางที่ ก-2 ในภาคผนวก ก) จะทำการตัดในส่วนของบิตตรวจสอบความผิดพลาดออกแล้ว โดยมีการใช้ข้อมูลเพียงแค่ 56 บิตที่เหลือ

ผลลัพธ์จากการทำตามตาราง PC-1 แล้ว จะนำมาแบ่งออกเป็น 2 ส่วน ส่วนละ 28 บิต เรียกว่า C และ D ซึ่งนำมาใช้นำการหาค่าของกุญแจย่อยแต่ละชุด โดยที่

Ci = LSi(Ci-1)

Di = Lsi(Di-1)

ซึ่ง LS คือ การเลื่อนบิตไปทางซ้าย (Left Shift หรือ Circular Left Shift หรือ Rotation) โดยเลื่อนบิตไปเป็นจำนวนเท่ากับจำนวนครั้งที่กำหนดไว้ในตารางที่ ก-3 (ในภาคผนวก ก) โดยมี C0 และ D0 เป็นค่าเริ่มต้น

และค่าของกุญแจย่อยในแต่ละชุด จะได้จากการนำเอา Ci และ Di มาต่อกัน แล้วนำค่านั้นไปเข้าตาราง PC-2 (ตารางที่ ก-4 ในภาคผนวก ก)

Ki = PC_2(CiDi)

ในการทำงานนั้น เนื่องจากขนาดของกุญแจมี 48 บิต แต่ R มีขนาด 32 บิต ดังนั้น จึงต้องทำการขยายขนาดของ R ให้เป็น 48 บิต เพื่อให้สามารถนำ R และกุญแจมา exclusive OR กันได้ โดยนำ R ไปเข้าตาราง E (ตารางที่ ก-5 ในภาคผนวก ก)

หลังจากทำการขยายขนาดของ R แล้ว จึงนำเอา R และกุญแจที่มีขนาดเท่ากันมาทำการ XOR กัน หลังจากนั้น นำผลลัพธ์ที่ได้มาทำการแบ่งออกเป็น 8 ส่วน ส่วนละ 6 บิต แล้วนำไปเข้าตาราง S (ตารางที่ ก-7 ในภาคผนวก ก) ซึ่งจะได้ผลลัพธ์ออกมาส่วนละ 4 บิต รวมกันได้ 32 บิต

จากนั้น นำข้อมูลขนาด 32 บิตนี้ ไปทำการสับเปลี่ยนตำแหน่งในตาราง P (ตารางที่ ก-6 ในภาคผนวก ก)

เมื่อทำขั้นตอนในส่วนของการเข้าฟังก์ชันครบ 16 ครั้งแล้ว จะได้ข้อมูลชุดหนึ่งที่มีขนาด 64 บิต โดยเรียกว่า PREOUTPUT หลังจากนั้น จึงนำข้อมูลนี้ไปทำการเรียงบิตกลับ

3. การเรียงบิตกลับ (Inverse Initial Permutation)

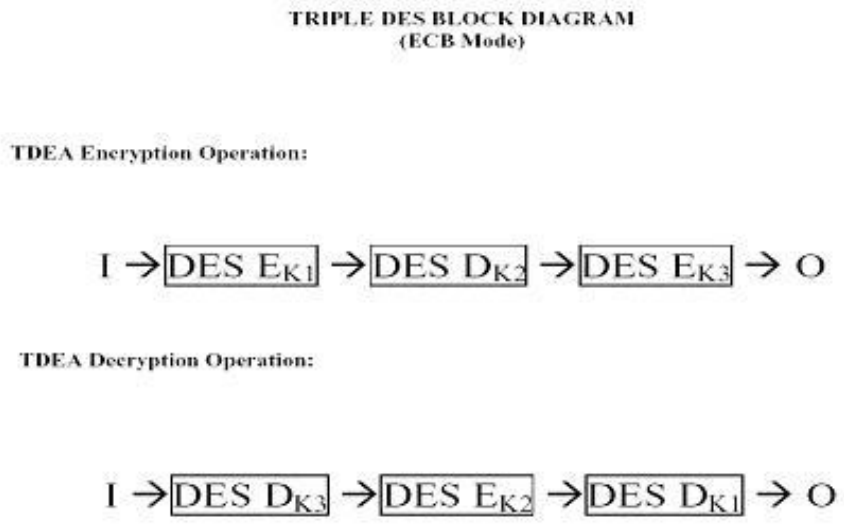
เมื่อได้ข้อมูลที่เป็น PREOUTPUT ออกมาแล้ว จะนำข้อมูลนั้น มาทำการสับเปลี่ยนตำแหน่งกลับจากในตอนแรก ซึ่งจะได้เป็นข้อมูลที่ผ่านการเข้ารหัสแล้ว (Cipher Text)

สำหรับกระบวนการถอดรหัสนั้น ทำเช่นเดียวกันกับกระบวนการเข้ารหัส คือ ใส่ข้อมูลที่ถูกเข้ารหัสข้อมูลมาแล้วที่มีขนาด 64 บิต พร้อมกับกุญแจตัวเดิมที่ใช้ในการเข้ารหัส แล้วทำตามขั้นตอนต่างๆ เหมือนกับการเข้ารหัสข้อมูล แต่ต้องทำการเปลี่ยนลำดับของกุญแจใหม่ โดยใช้กุญแจจากลำดับท้ายสุดเรียงมาจนถึงลำดับแรกสุด คือ K16, K15,…, K1

3.3 การเข้ารหัสแบบ 3DES (Triple Data Encryption Standard)

และการใช้งานจริงหากเราต้องการใช้ข้อมูลมีความปลอดภัยมากขึ้นสามารถทำได้โดยเราสามารถเอาอัลกอริทึม DES มาประยุกต์ใช้งาน

การเข้ารหัสแบบ 3DES นั้น เป็นการเข้ารหัสอีกแบบหนึ่ง โดยอิงมาตรฐานมาจากการเข้ารหัสแบบ DES ซึ่งได้ทำการเข้าและถอดรหัสแบบ DES เป็นจำนวน 3 ครั้ง และใช้กุญแจเป็นจำนวน 3 ตัวในการทำงาน



รูปที่ 3-4 กระบวนการเข้าและถอดรหัสด้วย 3DES

โดยมีขั้นตอนในการเข้ารหัสข้อมูล ดังนี้คือ

1. ทำการเข้ารหัสข้อมูลโดยใช้กุญแจตัวแรก
2. ทำการถอดรหัสข้อมูลโดยใช้กุญแจตัวที่สอง และ
3. ทำการเข้ารหัสข้อมูลโดยใช้กุญแจตัวที่สาม

ในส่วนของขั้นตอนในการถอดรหัสข้อมูล มีดังนี้คือ

1. ทำการถอดรหัสข้อมูลโดยใช้กุญแจตัวแรก
2. ทำการเข้ารหัสข้อมูลโดยใช้กุญแจตัวที่สอง และ
3. ทำการถอดรหัสข้อมูลโดยใช้กุญแจตัวที่สาม