บทที่ 3

แนวความคิดในการปรับเปลี่ยนอัลกอริทึมเดส

เนื่องจากผลกระทบของการพัฒนาทางเทคโนโลยี ทำให้ต้องการอัลกอริทึมสำหรับการเข้ารหัสลับข้อมูลที่มีประสิทธิภาพมากยิ่งขึ้น ซึ่งอาจจะทำได้โดยการออกแบบอัลกอริทึมสำหรับการเข้ารหัสลับใหม่ หรือ โดยการปรับเปลี่ยนแก้ไขอัลกอริทึมที่มีการใช้งานอยู่แต่เดิม ให้มีประสิทธิภาพมากยิ่งขึ้น

จากการศึกษางานวิจัยที่เกี่ยวข้องกับการพัฒนาอัลกอริทึมเดส พบว่าโครงสร้างการทำงานของอัลกอริทึมเดสมีลักษณะการทำงานที่มีประสิทธิภาพและมีความซับซ้อน โดยมีส่วนการทำงานที่สำคัญที่สุด คือ ฟังก์ชั่น f ซึ่งเป็นวงจรการทำงานที่ทำซ้ำ ๆ กัน 16 รอบ ประกอบด้วยวิธีการเข้ารหัสลับแบบพื้นฐานที่สำคัญ คือ การจัดลำดับตำแหน่งข้อมูล การแทนที่ข้อมูลด้วยข้อมูลอื่น และการเอกซ์คลูซีฟออร์ ทำให้ไม่สามารถหาความสัมพันธ์ระหว่างข้อมูลแต่ละบิตของข้อมูลเข้ารหัสและข้อมูลเนื้อแท้ได้ และเนื่องจากการที่อัลกอริทึมเดสได้รับการยอมรับให้เป็นอัลกอริทึมมาตรฐาน จึงทำให้อัลกอริทึมนี้ถูกพิจารณาตรวจสอบขั้นตอนการทำงานอย่างละเอียดจากนักวิชาการ และผู้เชี่ยวชาญด้านต่างๆ มากกว่าอัลกอริทึมอื่น ๆ แต่ก็ไม่พบจุดอ่อนที่สำคัญอันจะทำให้อัลกอริทึมเดสไม่ได้รับการยอมรับ โดยโครงสร้างพื้นฐานของอัลกอริทึมเดสยังคงมีศักยภาพในการป้องกันข้อมูลให้ปลอดภัยได้สูง

ดังนั้น จึงมีความเป็นไปได้ที่จะปรับเปลี่ยนอัลกอริทึมเดสซึ่งมีหลักการพื้นฐานการทำงานที่ดีอยู่แล้วให้มีประสิทธิภาพยิ่งขึ้นยากแก่การทำลาย สามารถป้องกันข้อมูลให้ปลอดภัยมากขึ้นโดยมีแนวความคิดในการปรับเปลี่ยนอัลกอริทึมเดส มีดังนี้ คือ

3.1 แนวความคิดในการปรับเปลี่ยนอัลกอริทึมเดส

จากการศึกษาพบว่าในส่วนของการแทนที่ข้อมูลด้วยข้อมูลที่ได้จากการเปิดค่าในตาราง S-boxes ซึ่งเป็นขั้นตอนหนึ่งในฟังก์ชั่น f สามารถที่จะปรับเปลี่ยนเพื่อให้ส่วนนี้มีการทำงานที่ซับซ้อนยิ่งขึ้นได้

ก่อนที่จะกล่าวถึงวิธีการในการปรับเปลี่ยนอัลกอริทึมเดส จะอธิบายการขั้นตอนการทำงานของอัลกอริทึมเดสเฉพาะในส่วนฟังก์ชั่น f ดังนี้

ขั้นตอนการทำงานส่วนของฟังก์ชั่น f ในอัลกอริทึมเดส หลังจากที่มีการจัดลำดับตำแหน่งบิตใหม่ตามตาราง Bit-selection E ซึ่งจะขยายข้อมูลจาก 32 บิต ไปเป็น 48 บิต แล้วนำมาเอกซ์คลูซีฟออร์กับสับคีย์ ได้ผลลัพธ์ออกมาซึ่งมี 48 บิตเท่าเดิม จะถูกแบ่งออกเป็น 8 กลุ่ม ๆ ละ 6 บิต ข้อมูลแต่ละกลุ่มนี้จะถูกใช้เพื่อเปิดค่าจากตาราง S-boxes ค่าที่ได้จากตาราง มีขนาด 4 บิต จะนำมาแทนที่ข้อมูลเข้า 6 บิต ที่ใช้ในการเปิดค่าจากตาราง โดยมีวิธีการ คือ

กำหนดให้ข้อมูลแต่ละกลุ่ม ๆ ละ 6 บิต คือ B_j จะถูกแทนด้วยค่าในตาราง S-boxes S_j ได้ผลลัพธ์ออกมามีขนาด 4 บิต วิธีการเลือกค่าในตาราง คือ

$$B_j = b_1\ b_2\ b_3\ b_4\ b_5\ b_6$$

การเลือกค่าจากตาราง S-boxes จะใช้บิตที่ 1 และบิตที่ 6 คือ b_1b_6 เพื่อกำหนดแถวที่จะเลือกในตาราง S-boxes และใช้บิต 2.3.4 และ 5 คือ $b_2b_3b_4b_5$ เพื่อกำหนดคอลัมน์ที่จะเลือกในตาราง S-boxes

S_1

	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	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

ตาราง S-box S_1

ตัวอย่างเช่น ข้อมูล B_1 = 010011

b_1b_6 = 01 เลือกแถวที่ 1

$b_2b_3b_4b_5$ = 1001 เลือกคอลัมน์ที่ 9

ค่าที่ได้จากตาราง S-boxes S_1 แถวที่ 1 คอลัมน์ที่ 9 คือ 6 ในเลขฐานสิบ หรือเท่ากับ 0110 ในเลขฐานสอง ดังนั้น จะได้ผลลัพธ์ 0110 แทนค่า 010011 และ

ทำในลักษณะเดียวกันกับกลุ่มข้อมูลทั้ง 8 กลุ่ม ได้ผลลัพธ์แล้วนำมารวมกันจะได้ข้อมูล 32 บิต เท่าเดิม

จะเห็นได้ว่าอัลกอริทึมเดสจะแบ่งข้อมูล 48 บิต ออกเป็น 8 กลุ่ม ๆ ละ 6 บิต ซึ่งเป็นการแบ่งกลุ่มแบบคงที่ (fixed box size) เราสามารถที่จะปรับเปลี่ยนวิธีการแบ่งกลุ่มใหม่ โดยไม่จำเป็นที่จะต้องแบ่งในลักษณะนี้เท่านั้น นั่นคือ ขนาดของข้อมูลแต่ละกลุ่มสามารถแปรเปลี่ยนได้ตามต้องการ (variable box size) แต่ให้ผลลัพธ์ของแต่ละกลุ่มเมื่อรวมกันแล้วได้ 32 บิต

การปรับเปลี่ยนอัลกอริทึมเดสโดยการเปลี่ยนวิธีการแบ่งกลุ่มข้อมูลจะทำให้มีการเปลี่ยนแปลงขั้นตอนการทำงานในอัลกอริทึมเดสดังนี้ คือ มีการเปลี่ยนวิธีการแบ่งกลุ่มข้อมูลที่ถูกนำมาใช้เพื่อเปิดค่าจากตาราง S-boxes และมีการเปลี่ยนตาราง S-boxes ซึ่งจะได้อธิบายในรายละเอียดต่อไปนี้

3.1.1 การเปลี่ยนวิธีการแบ่งกลุ่มข้อมูล

เราสามารถแบ่งกลุ่มข้อมูลเป็นแบบต่าง ๆ ได้หลายแบบ โดยมีเงื่อนไข คือ ข้อมูลเข้าทั้งหมด 48 บิต ถูกแบ่งออกเป็นกลุ่มขนาดต่าง ๆ และให้ผลลัพธ์ที่ได้จากการเปิดค่าในตารางเมื่อรวมกันจะได้ 32 บิต จำนวนกลุ่มที่ได้จะขึ้นกับขนาดของกลุ่ม และการแบ่งข้อมูลแต่ละแบบยังสามารถนำมาจัดลำดับได้วิธีการแบ่งกลุ่มที่แตกต่างกันอีกหลายวิธี ซึ่งจำนวนวิธีขึ้นอยู่กับขนาดของกลุ่มที่แบ่ง จำนวนกลุ่มที่แบ่ง และขนาดของผลลัพธ์ที่ได้จากการเปิดตารางโดยมีวิธีการคำนวณดังนี้

สมมติ แบ่งกลุ่มข้อมูลทั้งหมดออกเป็น M กลุ่ม

- ถ้าแต่ละกลุ่มมีการแบ่งที่แตกต่างกันทั้งหมด เมื่อนำมาจัดลำดับจะได้วิธีการจัดที่แตกต่างกันเท่ากับ M! (M Factorial) วิธี
- ถ้าแต่ละกลุ่มมีการแบ่งที่เหมือนกัน หรือซ้ำกัน ซึ่งข้อมูลสองกลุ่มใด ๆ ที่เหมือนกัน จะต้องเหมือนกันทั้งขนาดของกลุ่มข้อมูลที่นำไปเปิดตารางและขนาดของผลลัพธ์ที่ออกมา มิฉะนั้นจะถือว่าเป็นการแบ่งกลุ่มที่ต่างกัน ตัวอย่างเช่น

การแบ่งกลุ่มข้อมูลเป็นกลุ่มละ 7 7 7 7 7 7 6

ผลลัพธ์ของข้อมูลแต่ละกลุ่ม 5 5 5 5 4 4 4

จะถือว่ากลุ่มข้อมูลที่มีขนาด 7 บิต ได้ผลลัพธ์ออกมา 5 บิต กับการกลุ่มข้อมูลที่มีขนาด 7 บิต และได้ผลลัพธ์ออกมา 4 บิต จะเป็นกลุ่มข้อมูลที่มีความแตกต่างกันเป็นต้น

ดังนั้น เมื่อการแบ่งกลุ่มมีกลุ่มที่ซ้ำกัน ในการจัดลำดับวิธีการแบ่งกลุ่มจะต้องมีการตัดจำนวนวิธีที่ซ้ำกันออกไป

สมมติว่า ข้อมูลถูกแบ่งกลุ่มได้ทั้งหมด M กลุ่ม

มีกลุ่มที่ซ้ำกันแบบที่ 1 เท่ากับ r_1 กลุ่ม

มีกลุ่มที่ซ้ำกันแบบที่ 2 เท่ากับ r_2 กลุ่ม

. .

. .

. .

มีกลุ่มที่ซ้ำกันแบบที่ k เท่ากับ r_k กลุ่ม

จะได้จำนวนการแบ่งกลุ่มที่แตกต่างกันเท่ากับ $\dfrac{M!}{r_1!\, r_2! \ldots r_k!}$

จากตัวอย่างข้างต้น มีการแบ่งกลุ่มออกเป็น 7 กลุ่ม

การแบ่งกลุ่มที่มีขนาดกลุ่มละ 7 บิต ได้ผลลัพธ์กลุ่มละ 5 บิต มี 4 กลุ่ม

การแบ่งกลุ่มที่มีขนาดกลุ่มละ 7 บิต ได้ผลลัพธ์กลุ่มละ 4 บิต มี 2 กลุ่ม

การแบ่งกลุ่มที่มีขนาดกลุ่มละ 6 บิต ได้ผลลัพธ์กลุ่มละ 4 บิต มี 1 กลุ่ม

จะได้จำนวนการแบ่งกลุ่มที่แตกต่างกันเท่ากับ $\dfrac{7!}{4!\, 2!\, 1!} = 105$ วิธี

ต่อไปนี้เป็นตัวอย่างการแบ่งกลุ่มข้อมูลแบบต่าง ๆ และจำนวนวิธีการจัดลำดับของการแบ่งกลุ่มในแต่ละแบบ ดังนี้

- กรณีแบ่งกลุ่มข้อมูลเป็นกลุ่มละ 6 6 6 6 12 12

  ผลลัพธ์ที่ได้จากการแทนค่าในตาราง 4 4 4 4 8 8

ในการแบ่งกลุ่มลักษณะนี้ คือแบ่งข้อมูลออกเป็น 6 กลุ่ม กลุ่มละ 6 บิต จำนวน 4 กลุ่ม และกลุ่มละ 12 บิต จำนวน 2 กลุ่ม ได้ผลลัพธ์ออกมาเป็น กลุ่มละ 4 บิต จำนวน 4 กลุ่ม และกลุ่มละ 8 บิต จำนวน 2 กลุ่ม ตามลำดับ

การแบ่งกลุ่มลักษณะข้างต้น สามารถนำมาจัดลำดับใหม่ได้วิธีการแบ่งกลุ่มที่แตกต่างกันอีกเท่ากับ $\dfrac{6!}{4!\, 2!} = 15$ วิธี มีตัวอย่างการจัดลำดับการแบ่งกลุ่มดังนี้ คือ

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

- กรณีแบ่งกลุ่มข้อมูลเป็นกลุ่มละ 5 5 5 5 5 5 5 5 8
  ผลลัพธ์ที่ได้จากการแทนค่าในตาราง 3 3 3 3 3 3 3 3 8

ในการแบ่งกลุ่มลักษณะนี้ คือแบ่งข้อมูลออกเป็น 9 กลุ่ม กลุ่มละ 5 บิต จำนวน 8 กลุ่ม และกลุ่มละ 8 บิต จำนวน 1 กลุ่ม ได้ผลลัพธ์ออกมาเป็น กลุ่มละ 3 บิต จำนวน 8 กลุ่ม และกลุ่มละ 8 บิต จำนวน 1 กลุ่ม ตามลำดับ

การแบ่งกลุ่มลักษณะข้างต้น สามารถนำมาจัดลำดับใหม่ได้วิธีการแบ่งกลุ่มที่แตกต่างกันอีกเท่ากับ $\frac{9!}{8!\ 1!} = 9$ วิธี

- กรณีแบ่งกลุ่มข้อมูลเป็นกลุ่มละ 5 6 10 8 9 10
  ผลลัพธ์ที่ได้จากการแทนค่าในตาราง 3 4 7 5 6 7

ในการแบ่งกลุ่มลักษณะนี้ คือแบ่งข้อมูลออกเป็น 6 กลุ่ม ที่มีความแตกต่างกันทั้งหมด สามารถนำมาจัดลำดับใหม่ได้วิธีการแบ่งกลุ่มที่แตกต่างกันอีกเท่ากับ 6! หรือเท่ากับ 720 วิธี

- กรณีแบ่งกลุ่มข้อมูลเป็นกลุ่มละ 7 10 12 8 11
  ผลลัพธ์ที่ได้จากการแทนค่าในตาราง 5 6 8 6 7
  ในการแบ่งกลุ่มลักษณะนี้ คือแบ่งข้อมูลออกเป็น 5 กลุ่ม ที่มีความแตกต่างกันทั้งหมด สามารถนำมาจัดลำดับใหม่ได้วิธีการแบ่งกลุ่มที่แตกต่างกันอีกเท่ากับ 5! หรือเท่ากับ 120 วิธี

- กรณีแบ่งกลุ่มข้อมูลเป็นกลุ่มละ 12 12 12 12
  ผลลัพธ์ที่ได้จากการแทนค่าในตาราง 8 8 8 8
  ในการแบ่งกลุ่มลักษณะนี้ คือแบ่งข้อมูลออกเป็น 4 กลุ่ม กลุ่มละ 12 บิต จำนวน 4 กลุ่ม ได้ผลลัพธ์ออกมาเป็น กลุ่มละ 8 บิต จำนวน 4 กลุ่ม ตามลำดับ
  การแบ่งกลุ่มลักษณะข้างต้น สามารถนำมาจัดลำดับใหม่ได้วิธีการแบ่งกลุ่มที่แตกต่างกันอีกเท่ากับ $\frac{4!}{4!} = 1$ วิธี

การแบ่งกลุ่มข้อมูลเพื่อปรับเปลี่ยนอัลกอริทึมเดส เมื่อรวมวิธีต่าง ๆ ที่เป็นไปได้ในการแบ่งกลุ่มทั้งหมดจากตัวอย่างที่ยกมาข้างต้น มีจำนวนวิธีทั้งหมดถึง 970 วิธี โดยที่แต่ละวิธีมีการแบ่งกลุ่มที่แตกต่างกัน

ดังนั้น จะเห็นว่าเราสามารถเลือกวิธีการในการแบ่งกลุ่มได้มากมายหลายวิธี และเฉพาะที่ยกมาเป็นตัวอย่างข้างต้น มีวิธีต่าง ๆ ที่ให้เลือกใช้ถึง 970 วิธี นอกจากนี้ยังมีวิธีในการแบ่งกลุ่มอื่น ๆ อีกนอกเหนือจากที่ยกเป็นตัวอย่างมา แล้วแต่ผู้พัฒนาโปรแกรมจะกำหนดโดยมีเงื่อนไขให้มีข้อมูลเข้า 48 บิต และได้ข้อมูลออกเมื่อรวมแล้วได้ 32 บิต

3.1.2 การเปลี่ยนตาราง S-boxes

เมื่อมีการเปลี่ยนวิธีการแบ่งกลุ่มจะต้องมีการปรับเปลี่ยนตาราง S-boxes ให้เหมาะสมกับวิธีการแบ่งกลุ่มและเหมาะกับผลลัพธ์ที่ได้จากแต่ละกลุ่มที่ได้เปลี่ยนไป ในกรณีของอัลกอริทึมเดสซึ่งแบ่งข้อมูลออกเป็น 8 กลุ่ม ๆ ละ 6 บิต ได้ผลลัพธ์ออกมากลุ่มละ 4 บิต เมื่อพิจารณาตาราง S-boxes ของอัลกอริทึมเดสจะเห็นว่ามีการแบ่งเป็นตารางย่อย 8 ตาราง กำหนดให้ข้อมูล 1 กลุ่ม สำหรับตาราง 1 ตาราง ตัวอย่างเช่น ข้อมูลกลุ่มที่ 1 คือ B_1 จะเปิดค่าจากตาราง S-boxes S_1 หรือ ข้อมูลกลุ่มที่ 5 คือ B_5 จะเปิดค่าจากตาราง S-boxes S_5 เป็นต้น และใน 1 ตารางย่อยจะมี 4 แถว แต่ละแถวมี 16 ค่า มีค่าอยู่ในช่วง 0 - 15 (ดูตาราง S-boxes S1)

การที่ตารางย่อยของตาราง S-boxes มีลักษณะเช่นนี้ จะถูกกำหนดมาจากลักษณะการแบ่งกลุ่มของอัลกอริทึมเดส นั่นคือ ข้อมูลเข้า 6 บิต ได้ผลลัพธ์ 4 บิต จะใช้บิตที่ 2, 3, 4, 5 จำนวน 4 บิต เพื่อกำหนดจำนวนคอลัมน์ของตาราง หรืออีกนัยหนึ่งคือการกำหนดค่าที่เป็นไปได้จากการเปิดตาราง ดังนั้นจะมีค่าทั้งหมดเท่ากับ 16 ค่า อยู่ในช่วง 0 - 15 หรือ 0000 - 1111 ในเลขฐานสอง และจะใช้บิตที่ 1 และ บิตที่ 6 ขนาด 2 บิต เพื่อกำหนดจำนวนแถวที่เป็นไปได้ทั้งหมด ซึ่งมีค่าเท่ากับ 4 ดังนั้น 1 ตารางย่อยของตาราง S-boxes จะมี 4 แถว คือแถวที่ 0 ถึงแถวที่ 3 และแต่ละแถวมี 16 ค่า อยู่ในช่วง 0 - 15 โดยที่แต่ละค่าจะไม่ซ้ำกัน

ดังนั้น เมื่อมีการเปลี่ยนวิธีการแบ่งกลุ่มใหม่ ก็จะต้องมีการสร้างตาราง S-boxes ใหม่ ให้เหมาะสมกับการแบ่งกลุ่มและผลลัพธ์ที่ได้จากแต่ละกลุ่ม ตัวอย่างเช่น ถ้าแบ่งกลุ่มให้แต่ละกลุ่มมีขนาด 6, 6, 12, 12, 6, 6 บิต และให้ได้ผลลัพธ์ของแต่ละกลุ่มเท่ากับ 4, 4, 8, 8, 4, 4 บิต ตาราง S-boxes จะต้องประกอบด้วยตารางย่อย 6 ตาราง เนื่องจากข้อมูลถูกแบ่งเป็น 6 กลุ่ม โดยที่ตารางย่อย S_1, S_2, S_5, และ S_6 จะมีลักษณะเช่นเดียวกับตารางย่อยของตาราง S-boxes ของอัลกอริทึมเดส แต่ตารางย่อยของ S_3 และ S_4 จะต้องมีการปรับเปลี่ยนใหม่ให้เหมาะสม เนื่องจากข้อมูลกลุ่มที่ 3 และ 4 มีขนาด 12 บิต ให้ผลลัพธ์ออกมา 8 บิต ดังนั้น ค่าที่เป็นไปได้ทั้งหมดจากการเปิดตารางเท่ากับ 256 ค่า มีค่าอยู่ในช่วง 0 - 255 หรือ 00000000 - 11111111 ในเลขฐานสอง และค่าที่ใช้เพื่อกำหนดจำนวนแถวทั้งหมดคือ บิต 1, 2, 11 และ 12 ของข้อมูล มีขนาด 4 บิต ซึ่งจะได้จำนวนแถวเท่ากับ 16 แถว คือ แถวที่ 0 ถึงแถวที่ 15 และแต่ละแถวมี 256 ค่า มีค่าอยู่ในช่วง 0 - 255 ซึ่งแต่ละค่าจะไม่ซ้ำกัน การสร้างตารางนี้ทำได้โดยใช้วิธีการสุ่มตัวเลข

เพื่อให้สะดวกในการกล่าวอ้างถึงอัลกอริทึมเดสที่ได้ปรับเปลี่ยนแล้ว ต่อไปนี้จะเรียกอัลกอริทึมเดสที่ได้ปรับเปลี่ยนแล้วว่า อัลกอริทึมไอเดส (Improved DES Algorithm) และจะเห็นได้ว่าอัลกอริทึมไอเดสที่ได้ปรับเปลี่ยนมาจากอัลกอริทึมเดส จะไม่เป็นอัลกอริทึมที่เป็นมาตรฐาน เนื่องจากจะมีการกำหนดวิธีการแบ่งกลุ่ม และค่าของตาราง S-boxes จะไม่เป็นค่าคงที่ แต่จะเปลี่ยนไปแล้วแต่ความต้องการของผู้สร้างโปรแกรม

3.2 การวิเคราะห์เชิงประสิทธิภาพของอัลกอริทึมไอเดส

ในการเปรียบเทียบอัลกอริทึม 2 อัลกอริทึม ที่ใช้แก้ปัญหาเดียวกันว่า อัลกอริทึมใดจะมีประสิทธิภาพมากกว่ากัน วิธีหนึ่งที่เราสามารถนำมาใช้ได้ คือ การวัดความซับซ้อนของอัลกอริทึม (Complexity) ความซับซ้อนของอัลกอริทึม จะหมายถึงปริมาณงานที่อัลกอริทึมจะต้องทำเพื่อใช้แก้ปัญหาให้ได้คำตอบที่ต้องการ โดยวัดจากการดำเนินการหลักของอัลกอริทึมนั้น ตัวอย่างเช่น ต้องการค้นหา X จากรายชื่อทั้งหมด การดำเนินการหลักของอัลกอริทึมก็คือ การเปรียบเทียบ X กับทุก ๆ ชื่อในรายการนั้น เป็นต้น แต่ในการวิจัยนี้ต้องการวัดประสิทธิภาพของอัลกอริทึมสำหรับการเข้ารหัสลับข้อมูล ซึ่งอัลกอริทึมที่มีประสิทธิภาพ หมายถึง อัลกอริทึมที่ถูกทำลายได้ยาก หรือต้องใช้เวลาและทรัพยากรมากมายในการค้นหาคีย์ที่ใช้ในการรหัส ดังนั้น ความหมายของคำว่าประสิทธิภาพของอัลกอริทึม จะวัดด้วยความซับซ้อนของอัลกอริทึม ซึ่งมีความหมายเดียวกับ การวัดปริมาณงานที่ต้องทำเพื่อค้นหาคีย์ที่ใช้ในการเข้ารหัสลับ หรือปริมาณงานที่ต้องทำเพื่อทำลายอัลกอริทึม

ในรายงานการวิจัยนี้ การวัดประสิทธิภาพของอัลกอริทึมไอเดสโดยเปรียบเทียบกับอัลกอริทึมเดส จะทำโดยการวัดความซับซ้อนของอัลกอริทึม หรือวัดปริมาณงานที่ต้องทำเพื่อค้นหาคีย์ที่ใช้ในการเข้ารหัสลับ โดยวัดจากการดำเนินการหลัก คือ การเข้ารหัสลับข้อมูลและทำการเปรียบเทียบผลลัพธ์จากการเข้ารหัส กับข้อมูลเข้ารหัสที่ทราบค่าแล้วว่าตรงกันหรือไม่ จะต้องมีการเปรียบเทียบกี่ครั้งจึงจะพบคีย์ที่ต้องการ ซึ่งมีรายละเอียด ดังนี้

3.2.1 การวัดความซับซ้อนของอัลกอริทึมไอเดส

เนื่องจากอัลกอริทึมเดสมีขั้นตอนการทำงานที่แน่นอนตายตัว มีจำนวนกลุ่มและขนาดของกลุ่มคงที่ และจากการวิเคราะห์โครงสร้างอัลกอริทึมเดสในบทที่ 2 จะมีวิธีการที่มีโอกาสจะทำลายอัลกอริทึมเดสได้โดยใช้วิธีการค้นหาคีย์สำหรับการเข้ารหัสลับโดยตรง จะต้องค้นหาคีย์ที่เป็นไปได้ทั้งหมด และเมื่อมีการปรับเปลี่ยนอัลกอริทึมเดสโดยการเปลี่ยนวิธีการแบ่งกลุ่มของข้อมูล เราสามารถเลือกวิธีการแบ่งกลุ่มในลักษณะที่มีการแปรเปลี่ยนขนาดของแต่ละกลุ่มได้หลายวิธี ดังนั้น ผู้ที่ต้องการนำอัลกอริทึมที่มีการปรับเปลี่ยนแล้ว ไปพัฒนาสร้างเป็นโปรแกรมสำหรับเข้ารหัสลับ สามารถเลือกวิธีการแบ่งกลุ่มที่ต้องการได้ และซ่อนวิธีการแบ่งกลุ่มนี้ไว้เป็นความลับ

ผู้ที่ต้องการทำลายอัลกอริทึม หรือ ผู้ที่ต้องการค้นหาคีย์ที่ใช้สำหรับการเข้ารหัสลับจะต้องค้นหาว่าโปรแกรมเข้ารหัสเลือกใช้วิธีการแบ่งกลุ่มแบบใด โดยการทดลองเลือกวิธีการแบ่งกลุ่มจากวิธีการที่เป็นไปได้ก่อน แล้วจึงทำการค้นหาค่าคีย์สำหรับเข้ารหัสลับ จากค่าคีย์

ที่เป็นไปได้ทั้งหมดจนกว่าจะพบคีย์ที่ต้องการ ซึ่งจำนวนวิธีการแบ่งกลุ่มมีมากมายหลายวิธี ขึ้นอยู่กับการเลือกของผู้ใช้ เราสามารถวัดปริมาณงานที่ผู้ทำลายอัลกอริทึมต้องทำเมื่อใช้อัลกอริทึมที่มีการปรับเปลี่ยนแล้วโดยเปรียบเทียบกับการทำลายอัลกอริทึมเดสได้ดังนี้ คือ

ในการทำลายอัลกอริทึมเดสเพื่อค้นหาค่าคีย์ เนื่องจากค่าคีย์มีขนาด 56 บิต จะมีจำนวนคีย์ที่เป็นไปได้ทั้งหมด คือ 2^{56} คีย์ การค้นหาค่าคีย์จะมีสมมติฐานว่ารู้ข้อมูลเนื้อแท้และข้อมูลเข้ารหัสที่คู่กันบางส่วนแล้ว ในการค้นหาคีย์ต้องทำการทดสอบโดยการเข้ารหัสลับข้อมูลเนื้อแท้ด้วยค่าคีย์ที่เป็นไปได้ แล้วนำข้อมูลเข้ารหัสที่ได้มาเปรียบเทียบกับข้อมูลเข้ารหัสเดิมที่มีอยู่แล้ว ถ้าข้อมูลเข้ารหัสตรงกันแสดงว่าคีย์ที่ใช้ทดสอบเป็นคีย์ที่แท้จริง แต่ถ้าไม่ตรงกันก็จะทำการทดสอบกับคีย์อื่น ๆ ต่อไป การหาจำนวนครั้งในการเปรียบเทียบจนกว่าจะพบค่าคีย์ที่ต้องการกรณีที่ต้องทดสอบมากที่สุด (Worst-case) คือจะต้องทดสอบคีย์ทุกคีย์ ซึ่งเท่ากับ 2^{56} ครั้ง แต่ถ้าคิดเป็นค่าเฉลี่ยของจำนวนครั้งที่ต้องเปรียบเทียบจะประมาณ 2^{55} ครั้ง

ในส่วนของอัลกอริทึมไอเดส สามารถเลือกวิธีการแบ่งกลุ่มได้หลายวิธี ถ้าจำนวนวิธีการแบ่งกลุ่มที่เป็นไปได้เท่ากับ N วิธี ผู้ที่ต้องการค้นหาคีย์จะต้องทำการทดลองเลือกวิธีการแบ่งกลุ่มมา 1 วิธี และทำการทดสอบหาค่าคีย์ ถ้าวิธีการแบ่งกลุ่มที่เลือกไม่ตรงกับการแบ่งกลุ่มที่แท้จริง จะต้องทำการทดสอบคีย์ถึง 2^{56} ครั้ง แล้วต้องเลือกวิธีการแบ่งกลุ่มวิธีอื่นต่อไป และทำการทดสอบคีย์ต่อไป จนกว่าจะพบวิธีการแบ่งกลุ่มที่ถูกต้อง และค้นหาค่าคีย์ที่ต้องการพบ จำนวนครั้งในการทดสอบเพื่อหาวิธีการแบ่งกลุ่มที่ถูกต้อง อย่างมากที่สุดจะเท่ากับ N เมื่อคิดโดยเฉลี่ยแล้วจะเท่ากับ $\frac{N + 1}{2}$ ครั้ง ดังนั้น **ปริมาณงานโดยเฉลี่ยที่ใช้ในการค้นหาคีย์ที่ถูกต้องเมื่อใช้อัลกอริทึมไอเดสจะเท่ากับ $\frac{N + 1}{2} \times 2^{56}$ หรือ เท่ากับ $(N + 1) \times 2^{55}$**

หรืออาจจะกล่าวได้ว่า ปริมาณงานโดยเฉลี่ยในการค้นหาคีย์เมื่อใช้อัลกอริทึมไอเดส จะเท่ากับ (N + 1) เท่า ของปริมาณงานโดยเฉลี่ยในการค้นหาคีย์เมื่อใช้อัลกอริทึมเดส และจำนวนวิธีในการแบ่งกลุ่มที่เป็นไปได้อย่างน้อยที่สุดจากตัวอย่างที่ยกมาข้างต้นก็มีจำนวนถึง 970 วิธี ดังนั้นการที่จะพยายามทำลายอัลกอริทึมไอเดสจะทำได้ยากมากขึ้น ปริมาณงานโดยเฉลี่ยในการค้นหาคีย์จะต้องใช้เวลาและทรัพยากรมากมาย

3.2.2 การเพิ่มความซับซ้อนของอัลกอริทึมไอเดสจากอัลกอริทึมเดส เนื่องจากต้องค้นหาค่าจากตาราง S-boxes ที่เปลี่ยนไป

เมื่อเลือกวิธีการแบ่งกลุ่มแล้ว จะต้องมีการปรับตาราง S-boxes ให้เหมาะสมกับวิธีการแบ่งกลุ่ม ซึ่งตารางนี้อาจจะใช้วิธีการสร้างโดยวิธีการสุ่มตัวเลข เมื่อสร้างตารางนี้เสร็จแล้วจะมีการซ่อนตารางนี้ไว้เป็นความลับ ซึ่งตรงกันข้ามกับอัลกอริทึมเดสที่ตาราง S-boxes จะเปิดเผยให้ทราบทั่วไป เมื่อมีการซ่อนตารางไว้จะเป็นการเพิ่มความซับซ้อนให้กับอัลกอริทึมขึ้นอีก

สมมติว่า ในการเลือกวิธีการแบ่งกลุ่ม กำหนดให้แบ่งข้อมูลออกเป็น M กลุ่ม และให้มีผลลัพธ์จากแต่ละกลุ่มมีขนาดเท่ากับ O_i บิต โดยที่ i = 1, 2, ..., m นั่นคือ O_1 เป็นขนาดของผลลัพธ์จากกลุ่มที่ 1 O_2 เป็นขนาดของผลลัพธ์จากกลุ่มที่ 2 เป็นลักษณะนี้เรื่อยไปจนถึงกลุ่มที่ m เมื่อผู้ทำลายอัลกอริทึมได้ทดลองเลือกวิธีการแบ่งกลุ่มแล้ว เมื่อทำงานถึงขั้นตอนที่ต้องเปิดค่าจากตาราง S-boxes ซึ่งเป็นตารางที่ถูกซ่อนไว้ไม่ทราบค่า ผู้ทำลายอัลกอริทึมจะต้องทำการทดสอบเพื่อหาผลลัพธ์จากค่าที่เป็นไปได้ของแต่ละกลุ่ม ซึ่งผลลัพธ์ของข้อมูลกลุ่มที่ i จะมีขนาดเท่ากับ O_i ดังนั้นค่าของผลลัพธ์ที่เป็นไปได้ทั้งหมดเท่ากับ 2^{O_i} ค่า และมีค่าอยู่ในช่วง $0 - (2^{O_i} - 1)$ จำนวนครั้งที่ต้องทดสอบว่าค่าใดเป็นผลลัพธ์ที่แท้จริง จะขึ้นอยู่กับการเลือกวิธีการแบ่งกลุ่ม ถ้าเลือกวิธีการแบ่งที่ไม่ถูกต้อง จำนวนครั้งสูงสุดที่ต้องทดสอบ คือ 2^{O_i} และข้อมูลทั้งหมดมีอยู่ m กลุ่ม ดังนั้น จำนวนครั้งสูงสุดที่จะต้องทดสอบเพื่อหาค่าที่แท้จริงจะเท่ากับ $2^{O_1} \times 2^{O_2} \times \ldots \times 2^{O_m} = \prod_{i=1}^{m} 2^{O_i}$ ซึ่งจะทำให้การทำงานของอัลกอริทึมยิ่งมีความซับซ้อนมากขึ้น ตัวอย่างเช่น

ถ้าแบ่งข้อมูลเป็นกลุ่มละ 8 8 8 8 8 8

ได้ผลลัพธ์เป็นกลุ่มละ 6 6 6 6 4 4

จะได้ข้อมูล 6 กลุ่ม ๆ ละ 8 บิต ได้ผลลัพธ์เป็นข้อมูลกลุ่มละ 6 บิต จำนวน 4 กลุ่ม และ กลุ่มละ 4 บิต จำนวน 2 กลุ่ม ดังนั้นการหาค่าที่เป็นไปได้จากการเปิดตาราง S-boxes ที่มีการซ่อนไว้ จะต้องทดสอบจากค่าที่เป็นไปได้ จากตัวอย่างข้างต้นผลลัพธ์มีขนาด 6 บิต ค่าที่เป็นไปได้มีทั้งหมด 2^6 หรือ 32 ค่า และอยู่ในช่วง 0 - 31 ซึ่งมีทั้งหมด 4 กลุ่ม และยังมีผลลัพธ์ที่มีขนาด 4 บิต จำนวน 2 กลุ่ม ค่าที่เป็นไปได้ในแต่ละกลุ่มคือ 2^4 หรือ 16 และอยู่ในช่วง 0 - 15 ดังนั้นจำนวนครั้งสูงสุดที่จะต้องทดสอบเพื่อหาค่าของผลลัพธ์เท่ากับ $2^6 \times 2^6 \times 2^6 \times 2^6 \times 2^4 \times 2^4$ หรือ เท่ากับ 2^{32}

3.3 แนวความคิดอื่นในการปรับเปลี่ยนอัลกอริทึมเดส

นอกจากแนวความคิดในการปรับเปลี่ยนอัลกอริทึมเดส ซึ่งได้ผลลัพธ์เป็นอัลกอริทึมไอเดส ดังที่ได้เสนอมาแล้ว ยังมีแนวความคิดอื่น ๆ ที่สามารถปรับเปลี่ยนอัลกอริทึมเดสได้ ตัวอย่างเช่น **การเปลี่ยนวิธีการคำนวณค่าสับคีย์ (Subkey)** ทำได้ดังนี้ คือ

ในระบบการเข้ารหัสลับแบบอัลกอริทึมเดสนี้ จะต้องมีค่าคีย์ที่ใช้สำหรับเข้ารหัสลับ ค่าคีย์นี้จะถูกนำมาสร้างเป็นสับคีย์ 16 คีย์ ดูขั้นตอนการคำนวณค่าสับคีย์นี้ได้จากรูปที่ 2.4 ส่วนที่เราสามารถปรับเปลี่ยนการคำนวณได้ คือ การเปลี่ยนการเลื่อนตำแหน่งบิตแบบหมุนเวียนไปทางซ้าย (circular left shift) ที่อัลกอริทึมเดสใช้อยู่ ไปเป็นการเลื่อนตำแหน่งบิตแบบหมุนเวียนไปทางขวา (circular right shift) แทน ซึ่งจะได้ผลลัพธ์เป็นค่าสับคีย์ที่แตกต่างกัน แม้ว่าจะมีบางค่าที่เหมือนกัน แต่เมื่อเข้ารหัสแล้วจะได้ผลลัพธ์เป็นข้อมูลเข้ารหัสที่แตกต่างกัน จะเห็นว่าวิธีการนี้จะเป็นวิธีหนึ่งที่สามารถนำมาปรับเปลี่ยนอัลกอริทึมเดส ให้มีการทำงานที่ซับซ้อนยิ่งขึ้น แต่ว่าในการปรับเปลี่ยนอัลกอริทึมเดส ถ้าใช้วิธีการนี้เพียงอย่างเดียวจะเป็นการเพิ่มความซับซ้อนจากของเดิมไม่มากนัก เพราะเป็นการเพิ่มปริมาณงานที่ต้องทำเพื่อทำลายอัลกอริทึมขึ้นเป็น 2 เท่าของปริมาณที่ต้องทำเพื่อทำลายอัลกอริทึมเดสเท่านั้น ดังนั้น เพื่อให้การทำงานมีประสิทธิภาพยิ่งขึ้น เราสามารถนำวิธีการนี้มาใช้ประกอบกับอัลกอริทึมไอเดสที่ได้เสนอมาข้างต้น ก็จะได้วิธีการเข้ารหัสลับที่มีความซับซ้อนมาก

กล่าวโดยสรุปแล้วจะเห็นได้ว่า อัลกอริทึมเดสไอเดสมีความซับซ้อนกว่าอัลกอริทึมเดสมาก โดยดูได้จากปริมาณงานที่ต้องทำเพื่อทำลายอัลกอริทึม ผู้ที่ต้องการทำลายอัลกอริทึมไอเดสจะต้องทำงานเท่ากับ (N + 1) เท่าของปริมาณงานที่อัลกอริทึมเดสต้องทำ และยังมีความซับซ้อนในการค้นหาค่าที่เปิดจากตาราง S-boxes ยิ่งทำให้มีความยากในการทำลายอัลกอริทึมมากยิ่งขึ้นเท่ากับ 2^{32} คิดเป็นปริมาณงานที่ต้องทำเพื่อทำลายอัลกอริทึมไอเดสจะเท่ากับ (N + 1) 2^{32} เท่าของอัลกอริทึมเดส นอกจากนี้เรายังสามารถนำการเปลี่ยนวิธีการคำนวณค่าสับคีย์มาประกอบกับอัลกอริทึมไอเดสได้ และเพื่อดูประสิทธิภาพของการทำงานของอัลกอริทึมไอเดส ว่ามีความแตกต่างจากการทำงานของอัลกอริทึมเดสอย่างไร โดยจะทำการศึกษาในรายละเอียดของการทำงานของอัลกอริทึมทั้งสองแบบ เช่น ดูผลกระทบของอัลกอริทึมต่อข้อมูลทั้งหมดในระบบ ซึ่งประกอบด้วยข้อมูลเนื้อแท้ ข้อมูลเข้ารหัส และคีย์สำหรับการเข้ารหัส ว่ามีความสัมพันธ์กันอย่างไร โดยจะยกตัวอย่างเป็นกรณีศึกษา เพื่อใช้ในการทดสอบและเปรียบเทียบประสิทธิภาพของการทำงานของอัลกอริทึมไอเดสกับอัลกอริทึมเดส ดังจะได้กล่าวในรายละเอียดในบทต่อไป