บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

ในการศึกษาวิจัยเรื่องการปรับเปลี่ยนวิธีการเข้ารหัสลับของข้อมูลแบบอัลกอริทึมเดส ซึ่งในที่นี้จะเรียกอัลกอริทึมที่ปรับเปลี่ยนแล้วว่าอัลกอริทึมไอเดส สามารถสรุปผลได้ดังนี้ คือ

5.1.1 การวัดความซับซ้อน (Complexity) ของอัลกอริทึมไอเดสเมื่อเปรียบเทียบกับอัลกอริทึมเดส

จากการศึกษาสามารถสรุปได้ว่า อัลกอริทึมไอเดสมีความซับซ้อนกว่าอัลกอริทึมเดส โดยวัดจากปริมาณงานที่ต้องทำเพื่อค้นหาคีย์ที่ใช้สำหรับการเข้ารหัสลับให้พบ ซึ่งปริมาณงานเฉลี่ยในการค้นหาคีย์ของอัลกอริทึมไอเดส จะเท่ากับ $N + 1$ เท่าของปริมาณงานที่ต้องทำในการค้นหาคีย์ของอัลกอริทึมเดส และเฉพาะปริมาณงานเฉลี่ยที่ต้องทำเพื่อค้นหาคีย์ของอัลกอริทึมเดส จะเท่ากับการเปรียบเทียบ 2^{55} ครั้ง โดยที่ N คือ จำนวนวิธีการแบ่งกลุ่มที่เป็นไปได้ทั้งหมดของอัลกอริทึมไอเดส ซึ่งมีจำนวนวิธีมากมาย เพียงแค่รูปแบบการแบ่งกลุ่มที่นำมาทดสอบในบทที่ 4 จำนวน 9 รูปแบบ สามารถนำการแบ่งกลุ่มเหล่านั้นมาจัดลำดับได้วิธีการแบ่งกลุ่มเป็นแบบต่าง ๆ ถึง 1260 วิธี และยังมีวิธีการแบ่งกลุ่มแบบอื่นที่ไม่ได้กล่าวไว้อีกมากมาย ดังนั้น ปริมาณงานที่ต้องทำจึงมีปริมาณมาก ยากแก่การที่จะค้นหาคีย์ที่ใช้ในการเข้ารหัสลับได้

นอกจากนี้ยังมีความซับซ้อนที่เกิดจากการปรับเปลี่ยนไปของตาราง S-boxes คือต้องทำการทดสอบหาค่าของตารางที่ถูกต้อง ซึ่งจำนวนครั้งสูงสุดที่ต้องทดสอบ คือ 2^{32}

เมื่อรวมความซับซ้อนทั้งสองส่วน ทำให้การค้นหาค่าคีย์ที่ใช้ในการเข้ารหัสลับทำได้ยากขึ้น โอกาสที่จะค้นพบจะต้องสิ้นเปลืองทั้งเวลาและทรัพยากรมากมาย ทำให้ข้อมูลที่เข้ารหัสลับด้วยอัลกอริทึมไอเดสมีความปลอดภัย ซึ่งเราสามารถสรุปออกมาเป็นสมการได้ดังนี้ คือ

ให้ P = ปริมาณงานที่ต้องใช้เพื่อค้นหาค่าคีย์ของอัลกอริทึมเดส

P' = ปริมาณงานที่ต้องใช้เพื่อค้นหาค่าคีย์ของอัลกอริทึมไอเดส

N = จำนวนวิธีการแบ่งกลุ่มที่เป็นไปได้ทั้งหมดของอัลกอริทึมไอเดส

$$P' \simeq (N + 1)2^{32} P$$

5.1.2 การทดสอบการทำงานของอัลกอริทึมไอเดส

เนื่องจากอัลกอริทึมไอเดสมีการปรับเปลี่ยนมาจากอัลกอริทึมเดส ซึ่งเป็นอัลกอริทึมที่มีขั้นตอนการทำงานพื้นฐานที่มีประสิทธิภาพ ในการทดสอบเพื่อวิเคราะห์การทำงานของอัลกอริทึมไอเดส จะเป็นการเปรียบเทียบกับอัลกอริทึมเดส โดยทำการทดสอบหาความสัมพันธ์ของข้อมูลขาเข้าและขาออก และทดสอบหาความสัมพันธ์ของข้อมูลเข้ารหัสที่ขึ้นต่อข้อมูลเนื้อแท้

การทดสอบหาความสัมพันธ์ของข้อมูลขาเข้าและข้อมูลขาออก จะเป็นการดูการเปลี่ยนแปลงของข้อมูลเข้ารหัสเมื่อข้อมูลเนื้อแท้เปลี่ยนไป 1 บิต โดยใช้ระยะแฮมมิ่งในการวัดความเปลี่ยนแปลงของข้อมูล จากการทดสอบ เมื่อใช้อัลกอริทึมเดส ได้ระยะแฮมมิ่งประมาณ 31 และเมื่อใช้อัลกอริทึมไอเดสแบบต่าง ๆ ได้ระยะแฮมมิ่งอยู่ในช่วง 30 - 32 และเมื่อดูความเปลี่ยนแปลงของข้อมูลเข้ารหัสเมื่อคีย์สำหรับเข้ารหัสลับเปลี่ยนไป 1 บิต จะได้ระยะแฮมมิ่งเท่ากับ 33 สำหรับอัลกอริทึมเดส และ จะได้ระยะแฮมมิ่งอยู่ในช่วง 30 - 34 สำหรับอัลกอริทึมไอเดส จะเห็นได้ว่าระยะแฮมมิ่งที่ได้มีค่าใกล้เคียงกันทั้งของอัลกอริทึมเดสและไอเดส และมีค่าใกล้เคียงกับค่าเฉลี่ยระยะแฮมมิ่งที่ควรจะเป็น คือ 32 ซึ่งแสดงว่าข้อมูลมีการเปลี่ยนแปลงมากพอ เมื่อข้อมูลเข้ามีการเปลี่ยนแปลงเพียงบิตเดียว

ในส่วนการทดสอบหาความสัมพันธ์ของข้อมูลเข้ารหัสที่ขึ้นต่อข้อมูลเนื้อแท้ เป็นการวัดว่าในการทำงานของอัลกอริทึมเดส และไอเดส ซึ่งมีลักษณะเป็นการทำงานที่เป็นวงจรซ้ำ ๆ กัน 16 รอบ จะต้องใช้การทำงานกี่รอบจึงทำให้แต่ละบิตของข้อมูลเข้ารหัสขึ้นต่อข้อมูลเนื้อแท้ทุก ๆ บิต ผลจากการทดสอบพบว่าการทำงานที่เป็นวงจรซ้ำ ๆ กันในอัลกอริทึมเดสและอัลกอริทึมไอเดสกระทำเพียง 5 รอบ ก็ทำให้ข้อมูลเข้ารหัสขึ้นต่อข้อมูลเนื้อแท้ทุก ๆ บิตแล้ว และยังมีกรณีที่อัลกอริทึมไอเดสบางแบบใช้การทำงานเพียง 4 รอบก็เพียงพอ

ผลจากการทดสอบสรุปได้ว่า ประสิทธิภาพการทำงานของอัลกอริทึมไอเดสไม่ได้ด้อยไปกว่าอัลกอริทึมเดส แต่ยังมีความซับซ้อนมากกว่า

5.1.3 การวัดเวลาที่ใช้ในการประมวลผลข้อมูลของอัลกอริทึมทั้งสองแบบ

จากการทดสอบวัดเวลาที่ใช้ในการประมวลผลโดยใช้อัลกอริทึมทั้งสองแบบ จะใช้เวลาในการประมวลผลที่ค่อนข้างใกล้เคียงกัน ตัวอย่างเช่น การประมวลผลโดยการเข้ารหัสข้อมูลขนาด 11 ไบต์ อัลกอริทึมเดสจะใช้เวลา 1.700 วินาที ส่วนอัลกอริทึมไอเดสจะใช้เวลาอยู่ในช่วง 1.480 ถึง 1.820 วินาที หรือ ข้อมูลขนาด 85988 ไบต์ อัลกอริทึมเดสจะใช้เวลา 417 วินาที ส่วนอัลกอริทึมไอเดสจะใช้เวลาอยู่ในช่วง 395 ถึง 428 วินาที การที่ใช้เวลาในการประมวลผลที่แตกต่างกันไปบ้าง เนื่องมาจากความแตกต่างของจำนวนกลุ่มของข้อมูลที่ใช้ในการเปิดค่าจากตาราง S-boxes ถ้าจำนวนกลุ่มมาก จะใช้เวลามาก ถ้าจำนวนกลุ่มน้อย จะใช้เวลาน้อย

5.1.4 การวัดขนาดหน่วยความจำที่อัลกอริทึมไอเดสใช้

ขนาดของหน่วยความจำที่อัลกอริทึมไอเดส และอัลกอริทึมเดสใช้ จะขึ้นอยู่กับขนาดของตาราง S-boxes ถ้าตารางมีขนาดใหญ่ ขนาดของหน่วยความจำที่ใช้ก็จะมาก จากการทดสอบอัลกอริทึมเดสใช้ขนาดของหน่วยความจำเท่ากับ 41072 ไบต์ และมีขนาดตาราง S-boxes เท่ากับ 2546 ไบต์ ส่วนของอัลกอริทึมไอเดสแบบต่าง ๆ จะใช้ขนาดหน่วยความจำตั้งแต่ 41981 - 57144 ไบต์ และมีขนาดตารางตั้งแต่ 3874 - 60856 ไบต์ แต่ขนาดของตาราง S-boxes นี้มีขนาดใหญ่หรือเล็กจะขึ้นอยู่กับการแบ่งกลุ่มข้อมูล ถ้าข้อมูลแต่ละกลุ่มมีขนาดใหญ่ ขนาดของตาราง S-boxes ก็จะมีขนาดใหญ่ด้วย

5.2 ข้อควรพิจารณาในการใช้อัลกอริทึมไอเดสเพื่อเข้ารหัสลับข้อมูล

5.2.1 การเลือกวิธีการแบ่งกลุ่มข้อมูลที่นำมาใช้เพื่อเปิดค่าจากตาราง S-boxes

การเลือกวิธีการแบ่งกลุ่มข้อมูลที่นำมาใช้เพื่อเปิดค่าจากตาราง S-boxes ของอัลกอริทึมไอเดส จะพิจารณาใน 2 ส่วน คือ การกำหนดจำนวนกลุ่มในการแบ่งข้อมูล และการกำหนดขนาดของแต่ละกลุ่มทั้งขนาดของข้อมูลเข้าและขนาดของข้อมูลออก โดยที่การกำหนดจำนวนกลุ่ม และการกำหนดขนาดของแต่ละกลุ่มจะมีความสัมพันธ์กัน และมีข้อจำกัดอยู่ภายในเงื่อนไขว่า ข้อมูลเข้าทั้งหมดมี 48 บิต และให้ผลลัพธ์เป็นข้อมูลออกขนาด 32 บิต

ในการแบ่งข้อมูล ถ้ามีการกำหนดจำนวนกลุ่มน้อยจะทำให้ขนาดของแต่ละกลุ่มมีขนาดใหญ่ขึ้น ทำให้ใช้เวลาในการประมวลผลลดลง เนื่องจากจำนวนครั้งในการเปิดหาค่าจากตาราง S-boxes น้อยลง และมีจำนวนตารางน้อยลง แต่ตาราง S-boxes แต่ละ

ตารางจะมีขนาดใหญ่ขึ้น ซึ่งทำให้ต้องใช้ขนาดของหน่วยความจำมากขึ้น ในทางตรงกันข้าม ถ้ามีการกำหนดจำนวนกลุ่มข้อมูลมาก จะทำให้แต่ละกลุ่มมีขนาดเล็กลง และมีผลต่อการทำงานคือ ต้องใช้เวลาในการประมวลผลมากขึ้น มีจำนวนตาราง S-boxes มากขึ้น แต่ขนาดของแต่ละตารางจะเล็กลง

แต่อย่างไรก็ตาม จากการวิจัยของ Gordon และ Retkin ได้สรุปว่าขนาดของกลุ่มข้อมูลที่ใหญ่ขึ้น ทำให้ความน่าจะเป็นที่จะเกิดความสัมพันธ์ระหว่างข้อมูลเข้าและข้อมูลออกเป็นแบบเชิงเส้นมีค่าลดน้อยลงไปมาก การที่ความสัมพันธ์ระหว่างข้อมูลเข้าและข้อมูลออกไม่เป็นแบบเชิงเส้น จะทำให้การหากฎเกณฑ์ของความสัมพันธ์ที่เป็นระบบกระทำได้ยาก ทำให้การพยายามทำลายอัลกอริทึมไอเดสจะทำได้ยากขึ้น

ดังนั้น จะเห็นว่าถ้าเลือกวิธีการแบ่งกลุ่มขนาดใหญ่จะมีข้อดี คือโอกาสที่จะเกิดความสัมพันธ์แบบเชิงเส้นระหว่างข้อมูลเข้าและข้อมูลออกมีน้อยลง หรืออแทบจะไม่มีเลย และใช้เวลาในการประมวลผลน้อยลง แต่ก็จะทำให้ต้องสร้างตารางที่มีขนาดใหญ่ ทำให้เปลืองเนื้อที่ จึงควรจะพิจารณาเลือกวิธีการแบ่งกลุ่มที่เหมาะสม

5.2.2 การหลีกเลี่ยงการใช้วีคคีย์ (Weak Key) และเซมิ-วีคคีย์ (Semi-Weak Key)

เนื่องจากในการเข้ารหัสลับส่วนที่สำคัญมาก นอกจากความซับซ้อนของอัลกอริทึมที่ใช้ในการเข้ารหัสแล้ว ก็คือ การเลือกคีย์สำหรับการเข้ารหัสลับ จะต้องเป็นคีย์ที่ดี ควรหลีกเลี่ยงการใช้คีย์ที่เป็นพวกวีคคีย์ และ เซมิ-วีคคีย์ เพราะจะทำให้อัลกอริทึมมีความซับซ้อนลดลง

5.2.3 ข้อจำกัดของการใช้ระยะแฮมมิ่ง (Hamming Distance)

การใช้ระยะแฮมมิ่งในการวัดความแตกต่างระหว่างบิตต่อบิตของข้อมูล 2 ชุด ระยะแฮมมิ่งไม่ได้บอกอะไรนอกจากจะบอกถึงความแตกต่างของข้อมูลแต่ละบิตในตำแหน่งที่ตรงกันเท่านั้น และไม่ได้ให้คำตอบที่ตายตัวว่าข้อมูลทั้งสองชุดมีความแตกต่างกันโดยสิ้นเชิง ตัวอย่างเช่น ถ้าข้อมูลทั้งสองมีความแตกต่างกันในทุก ๆ บิต แสดงว่าข้อมูลทั้งสองเป็นคอมพลีเมนต์กัน (Complementation) ดังนั้น ระยะแฮมมิ่งที่วัดได้ ควรจะมีค่าเฉลี่ยประมาณครึ่งหนึ่งของจำนวนบิตทั้งหมด แต่ในกรณีนี้ก็อาจจะเป็นในลักษณะที่ครึ่งหนึ่งของข้อมูลเหมือนกันและอีกครึ่งหนึ่งต่างกันทั้งหมด ดังนั้นจะเห็นได้ว่า ในการวัดระยะแฮมมิ่ง จึงควรดูลักษณะข้อมูลประกอบไปด้วยกัน

5.3 ข้อเสนอแนะ

5.3.1 การเข้ารหัสลับที่ใช้ในการประมวลผลเวลาจริง (Real Time Processing)

การเข้ารหัสลับโดยใช้อัลกอริทึมเดส หรืออัลกอริทึมไอเดส จะเป็นการเข้ารหัสทีละกลุ่ม ๆ ละ 64 บิต แต่ในทางปฏิบัติจริง เราต้องการทำงานในลักษณะใช้การประมวลผลเวลาจริง คือ เมื่อรับข้อมูลแล้ว ซึ่งข้อมูลที่รับมาอาจจะเป็นทีละ 1 บิต หรือทีละ 1 ตัวอักษร หรืออาจจะเป็นหลาย ๆ บิต หลาย ๆ ตัวอักษรก็ได้ แล้วนำข้อมูลนั้นมาทำการเข้ารหัสลับ และส่งไปยังสายสื่อสารทันที ไม่ต้องคอยจนกว่าจะรับข้อมูลได้ครบ 64 บิตเสียก่อน การแก้ปัญหาสามารถกระทำได้โดยการนำเทคนิค Cipher Feedback หรือ CFB (รายละเอียด ดู (Davies and Price, 1984)) มาช่วยในการเข้ารหัสลับจะทำให้สามารถเข้ารหัสได้ทีละ 1 บิต หรือหลาย ๆ บิต แต่ไม่เกิน 64 บิต

5.3.2 การวิเคราะห์อัลกอริทึมไอเดสด้วยทฤษฎีทางคณิตศาสตร์

ในการวิจัยนี้ ได้เสนอวิธีการต่าง ๆ เพื่อแสดงถึงความซับซ้อนและประสิทธิภาพในการทำงานของอัลกอริทึมไอเดส ตัวอย่างเช่น การวัดปริมาณงานที่ต้องทำเพื่อค้นหาค่าคีย์สำหรับเข้ารหัสลับ และการทดสอบวิธีการต่าง ๆ เพื่อดูประสิทธิภาพในการทำงานของอัลกอริทึมซึ่งได้ข้อสรุปว่า อัลกอริทึมไอเดสมีความซับซ้อนกว่าอัลกอริทึมเดส ส่วนในการทดสอบเพื่อดูประสิทธิภาพการทำงาน ก็พบว่า ประสิทธิภาพการทำงานไม่ด้อยไปกว่าอัลกอริทึมเดส

แต่อย่างไรก็ตาม ยังมีวิธีการอื่น ๆ อีกที่สามารถนำมาใช้เพื่อพิสูจน์ถึงประสิทธิภาพในการทำงานของอัลกอริทึมไอเดส และวิธีการหนึ่ง ก็คือ การวิเคราะห์อัลกอริทึมไอเดสด้วยทฤษฎีทางคณิตศาสตร์ นั่นคือ เป็นการพยายามพิสูจน์ด้วยทฤษฎีทางคณิตศาสตร์ เพื่อเป็นการยืนยันในอีกทางหนึ่งว่าอัลกอริทึมไอเดสมีประสิทธิภาพ มีความซับซ้อน มีข้อดีที่ควรปฏิบัติ และด้อยที่ควรหลีกเลี่ยงอย่างไรบ้างในเชิงคณิตศาสตร์