

เทคนิคการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล

Data Positioning Technique for Data Recovery

พงษ์ศักดิ์ สังขมณี^{1*}, ฉัตรตระกูล สมบัติธีระ²

Phongsak Sangkhamanee^{1*}, Chattrakul Sombattheera²

คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม^{1,2}

Faculty of Informatics at Mahasarakham University^{1,2}

E-Mail pongsak.s@msu.ac.th, chattrakul.s@msu.ac.th

บทคัดย่อ

งานวิจัยครั้งนี้มีวัตถุประสงค์เพื่อเพื่อ 1) ศึกษาแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล โดยใช้เทคนิคการ XOR 2) ค้นหาแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลด้วยวิธีการเขียนโปรแกรมคอมพิวเตอร์ 3) นำเสนอการทดสอบการกู้คืนข้อมูลของแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลด้วยการเขียนโปรแกรมคอมพิวเตอร์ เครื่องมือการวิจัยได้แก่ 1) คอมไพเลอร์ภาษาซี Turbo C++ Version 4.0 (Windows) และ 2) Apache Web Server Version 2.4.25

ผลการวิจัยพบว่า 1) ผลการศึกษารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลของงานวิจัยก่อนหน้า พบว่า รูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลขนาด 3×3 โดยใช้เทคนิคการ XOR เพียงรูปแบบเดียว คือ การแบ่งไฟล์ข้อมูลออกเป็น 6 ส่วน และสร้าง Parity อีก 3 ส่วน แล้วนำไฟล์ข้อมูลทั้ง 9 ส่วน ไปจัดวางในตำแหน่งรูปแบบการจัดวาง 3×3 เพียงรูปแบบเดียวเท่านั้น และการกู้ข้อมูลของงานวิจัยดังกล่าวจะสามารถทำได้ก็ต่อเมื่อข้อมูลหายไปไม่เกิน 3 กล่องที่อยู่ในแถวเดียวกัน หรือ 3 กล่องที่อยู่คอลัมน์เดียวกันเท่านั้น 2) ผลการวิจัยการค้นหาแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล ด้วยวิธีการเขียนโปรแกรมคอมพิวเตอร์ภาษา C++ พบว่า รูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลขนาด 3×3 โดยใช้เทคนิคการ XOR มีจำนวนมากกว่า 1 รูปแบบ ผลการทดลองจากเขียนโปรแกรมเพื่อค้นหาแบบการจัดวางตำแหน่งของกล่องข้อมูลขนาด 3×3 พบว่า มีรูปแบบการจัดวางข้อมูลมากถึง 2,592 รูปแบบที่สามารถกู้คืนข้อมูลได้ 3) ผลการวิจัยเพื่อทดสอบการกู้คืนข้อมูลตามรูปแบบการจัดวางข้อมูลทั้ง 2,592 รูปแบบ ด้วยการเขียนโปรแกรมโดยการจำลองกระจายข้อมูลไปเก็บยังเครื่องแม่ข่ายจำนวน 3 เครื่อง ต่างระบบปฏิบัติการเพื่อทดสอบการกู้คืนข้อมูล จากการทดสอบพบว่า สามารถกู้คืนข้อมูลที่สูญหายไปกลับคืนมาได้ 100 เปอร์เซ็นต์ ภายใต้เงื่อนไขข้อมูลสูญหายไปไม่เกิน 3 กล่องที่อยู่ในแถวเดียวกัน หรือ 3 กล่องที่อยู่คอลัมน์เดียว

คำสำคัญ: กล่องข้อมูล, รูปแบบการจัดวางข้อมูล, กู้คืนข้อมูล, เทคนิคการ XOR

Abstract

The objectives of this study are 1) to investigate the data positioning pattern of the recovery block through XOR technique, 2) to find out data positioning pattern through software development, and 3) to propose the recovery testing results of data positioning pattern of the recovery block.

The results reveal that 1) from the previous studies, XOR is the only one technique offered for data positioning which is stripping the data into 6 parts and creating 3 parity, then positioning all 9 parts into the 3×3 . However, the technique mentioned earlier can be done when

3 lost data is in the same row or the same column. 2) The results from C++ software development to find out the data positioning of 3x3 data block through XOR technique show that 2,592 data could be recovered. 3) To test the data recovery of 2,592 patterns from program development by stimulating 3 servers found that the data could be recovered for 100 % under the condition of less than 3 lost data in the same row or same column.

Keywords: Data block, Data positioning, Data recovery, XOR technique

บทนำ

มนุษย์ต้องการความสะดวกสบายในด้านความเป็นอยู่ การดำเนินชีวิต ตลอดจนการประกอบอาชีพ การทำธุรกิจที่ต้องมีการแข่งขันกันเป็นจำนวนมาก ไม่ใช่แค่การแข่งขันในระดับประเทศเท่านั้น แต่หมายถึงในระดับโลก ยิ่งในปัจจุบันนี้ความก้าวหน้าด้านเทคโนโลยีก้าวไกลไปมาก ทำให้การทำธุรกิจต่าง ๆ ต้องมีความรวดเร็ว ผลิตผลน้อยที่สุด หรือถ้าผลิตผลก็สามารถที่จะแก้ไขได้ในระยะเวลาอันน้อยนิด หรือแก้ไขได้ทันทีทันใด เพื่อให้การดำเนินธุรกิจหรือดำเนินงานในองค์กรดำเนินไปได้อย่างราบรื่น และต้องมีแผนสำรองกรณีที่เกิดข้อผิดพลาดหรือเหตุไม่คาดฝันเกิดขึ้นกับข้อมูลที่มีความสำคัญเป็นอย่างมาก ตัวอย่างเช่น การเก็บรักษาข้อมูลที่มีความสำคัญในระบบคอมพิวเตอร์ที่ต้องการความคงสภาพของข้อมูลให้อยู่ในสภาพที่ร้อยเปอร์เซ็นต์ [1] หมายถึง ไม่มีข้อมูลสูญหายไปนั่นเอง ดังนั้นการรักษาข้อมูลทางคอมพิวเตอร์จึงจำเป็นต้องมีการออกแบบระบบให้มีความรัดกุม และต้องมีระบบการกู้คืนข้อมูลเมื่อข้อมูลบางส่วนนั้นหายไป หรือสูญเสียไปจากสาเหตุอื่น ๆ ด้วยเวลาอันน้อยนิดหรือกู้คืนข้อมูลได้ทันที เพื่อประกันความคงสภาพของข้อมูลให้อยู่ในสภาพเดิมอย่าง 100 เปอร์เซ็นต์ ดังนั้นการออกแบบระบบการกู้คืนข้อมูลทางคอมพิวเตอร์จึงเป็นเรื่องที่สำคัญมากของการรักษาความมั่นคงของข้อมูลในระบบคอมพิวเตอร์ [2], [3]

แนวคิดของ Wiwat Sriphum [4] มีความน่าสนใจซึ่งได้มีการคิดค้นและนำเสนอการออกแบบบาร์โค้ดแบบสองมิติ สามารถกู้คืนข้อมูลได้โดยใช้หลักการ XOR [5] เข้ามาช่วยในการหาข้อมูลในส่วนที่สูญหายไป แนวคิดงานวิจัยดังกล่าว กล่าวถึงรูปแบบการกู้คืนข้อมูลที่สูญหายไป โดยการแบ่งไฟล์ข้อมูลออกเป็น 6 ส่วน และสร้าง Parity อีก 3 ส่วน แล้วนำไฟล์ข้อมูลทั้ง 9 ส่วน ไปจัดวางในตำแหน่งรูปแบบการจัดวาง 3×3 เพียงรูปแบบเดียวเท่านั้น และการกู้ข้อมูลของงานวิจัยดังกล่าวจะสามารถทำได้ก็ต่อเมื่อข้อมูลหายไปไม่เกิน 3 กล่องที่อยู่ในแถวเดียวกัน หรือ 3 กล่องที่อยู่คอลัมน์เดียวกันเท่านั้น

จากการศึกษาของงานวิจัยของ Wiwat Sriphum [4] ผู้วิจัยได้พบว่า รูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลโดยใช้เทคนิคการ XOR นั้นมีจำนวนมากกว่า 1 รูปแบบ ดังนั้นผู้วิจัยขอเสนอ “เทคนิคการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล” โดยใช้เทคนิคการ XOR ที่มากกว่า 1 รูปแบบ และนำเสนอโปรแกรมเพื่อทดสอบการกู้คืนข้อมูลของรูปแบบการจัดวางข้อมูลที่มีมากกว่า 1 รูปแบบดังกล่าว ซึ่งประโยชน์ของเทคนิคการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลโดยใช้เทคนิคการ XOR นี้ จะเป็นอีกทางเลือกที่จะนำไปใช้ประโยชน์ได้หลากหลาย เช่น การพัฒนาและปรับปรุงการสร้าง บาร์โค้ดแบบสองมิติ รุ่นใหม่ๆ ให้มีประสิทธิภาพมากยิ่งขึ้นสามารถกู้คืนข้อมูลได้ในหลายๆ รูปแบบ การนำแนวคิดนี้ไปออกแบบการทำระบบการจัดเก็บข้อมูลของเครื่องคอมพิวเตอร์ (Database Server) ให้รองรับการกู้คืนข้อมูลได้เมื่อเกิดเหตุฉุกเฉิน เช่น ไฟไหม้ ดึงถล่ม น้ำท่วม แผ่นดินไหว

1. วัตถุประสงค์การวิจัย

- 1.1 เพื่อศึกษารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล โดยใช้เทคนิคการ XOR
- 1.2 เพื่อค้นหารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล ด้วยวิธีการเขียนโปรแกรมคอมพิวเตอร์
- 1.3 เพื่อนำเสนอการทดสอบการกู้คืนข้อมูลของรูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล ด้วยการเขียนโปรแกรมคอมพิวเตอร์

2. เอกสารและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 Exclusive OR (XOR) [5] เป็นการกระทำทางตรรกศาสตร์ โดยมีหลักการ คือ มีขาวิ่งเข้าตั้งแต่ 2 ขาขึ้นไป เช่น A, B จะให้ขาออกเป็นลอจิก 0 เมื่อขาเข้ามีลอจิกเหมือนกัน จะให้ขาออกเป็นลอจิก 1 เมื่อ ขาเข้ามีลอจิกต่างกัน กล่าว คือ (บิตไหน) มีค่าเหมือนกันได้ 0 กับ มีค่าต่างกันได้ 1

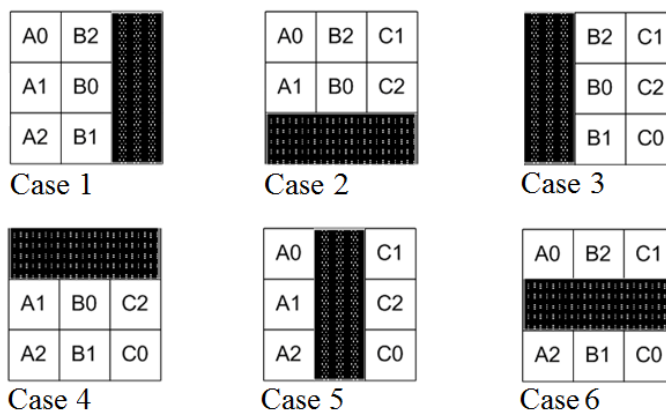
2.1.2 Data Striping [6] คือการแบ่งข้อมูลออกเป็นส่วน ๆ แล้วนำแต่ละส่วนไปเก็บใน Harddisk แต่ละตัว การทำ Striping จะช่วยให้การอ่าน หรือเขียนข้อมูลใน Disk array มีประสิทธิภาพมากขึ้น เพราะแต่ละไฟล์จะถูกแบ่งเป็นส่วนๆ กระจายไปเก็บในส่วนที่ต่างกันของ Harddisk หลายตัว โดย Harddisk เหล่านั้นทำงานไปด้วยกันแบบขนาน (Parallel) จึงทำให้การเข้าถึงข้อมูลนั้นเร็วกว่า Harddisk แบบตัวเดียว

2.1.3 RAID 5 [7] เป็นกระบวนการเก็บข้อมูลในฮาร์ดดิสก์เป็น Block คือการเอา Harddisk ตั้งแต่ 3 ลูกขึ้นไปมารวมกันโดยจะเขียนข้อมูลกระจายไปทั้ง Harddisk ทุกลูกและจะมี parity ในแต่ละลูกไว้สำหรับกู้ข้อมูลหาก Harddisk ลูกใดลูกหนึ่งเสียพื้นที่รวมจะได้ $3TB + 3TB + 3TB = 6TB$ ($A \times N - 1$; A ขนาดความจุต่อลูก, N จำนวน Harddisk)

2.1.4 กล่องข้อมูล (Data Block) [4] คือ กลุ่มของ Physical byte บนดิสก์โดยมีโครงสร้างของ Block ประกอบไปด้วย 1) Header (Common and Variable) คือส่วนที่เก็บข้อมูลของ Block นั้น ๆ เช่น Address, ประเภทของ Segment (Data, Index) ขนาดของ Header จะไม่คงที่ 2) Table Directory เก็บข้อมูลของ Table ที่มีข้อมูลอยู่ใน Block นั้น ๆ 3) Row Directory เป็นส่วนที่เก็บข้อมูลของแต่ละ Row ใน Block นั้น และ Address ที่เก็บ Row นั้น ๆ 4) Row Data เป็นส่วนที่เก็บข้อมูลจริง ๆ ของ Row กล่าวคือใน 1 Row อาจจะมีใหญ่กว่า Block ก็ได้ ก็คือมีข้อมูลอยู่ในหลายๆ Block 5) Free Space เป็นที่ว่างสำหรับใส่ข้อมูลของ Row ที่เพิ่มเข้ามา สำหรับข้อมูลที่จะ Update ใน Row เดิมที่อยู่ใน Block นั้น ๆ

2.2 งานวิจัยที่เกี่ยวข้อง

2.2.1 DR code (Data Restorable Code) [4] เป็นบาร์โค้ดสองมิติที่ทนทานกับการสูญหายของข้อมูลการออกแบบผู้วิจัย แสดงให้เห็นว่า DR code ของเราสามารถที่จะกู้คืนข้อมูลได้ดีกว่า QR code ที่นิยมใช้แพร่หลายในปัจจุบัน จากการศึกษา H-level QR code ซึ่งเป็น Model ของ QR code ที่สามารถกู้คืนข้อมูลได้มากที่สุด โดยจะสามารถกู้คืนข้อมูลได้ 30 เปอร์เซ็นต์ สำหรับความเสียหายที่ยอมให้เกิดขึ้นได้ใน 2 รูปแบบ จากการศึกษาพบว่า DR code สามารถที่จะกู้คืนข้อมูลได้ถึง 33 เปอร์เซ็นต์ สำหรับความเสียหายที่ยอมให้เกิดขึ้นได้ถึง 6 รูปแบบดังภาพที่ 1 การศึกษาใช้อัลกอริทึมของการกู้คืนข้อมูลที่ง่ายกว่า โดยการใช้ XOR แทนที่จะใช้ Reed-Solomon Error Correction ที่ซับซ้อน



ภาพที่ 1 Case ของความเสียหายของ DR code

การกู้ข้อมูลคืน เช่น Case 2 ที่ข้อมูลในแถบล่างเสียหายสามารถคำนวณหา A2 ได้จาก $A2 = B2 \text{ XOR } C2$ และสามารถคำนวณหา B1 ได้จาก $B1 = A1 \text{ XOR } C1$ และสามารถคำนวณหา C0 ได้จาก $C0 = A0 \text{ XOR } B0$ เป็นต้น

2.2.2 Recover Of Failure Data In Data Integrity Checking With Erasure Code and Replica Server Management System [8] งานวิจัยนี้กล่าวถึงการบริหารจัดการข้อมูลบนระบบ Cloud Computing ที่ซึ่งเมื่อมีไฟล์ข้อมูลในระบบ Cloud ถูกการโจมตีและทำให้ข้อมูลสูญหายไป และกระบวนการเข้ารหัสและถอดรหัสใช้เวลานาน จึงทำให้เกิดปัญหาด้านการรักษาความปลอดภัยในระบบ Cloud [9] งานวิจัยนี้จึงเสนอวิธีการวางระบบจัดเก็บข้อมูลที่มีประสิทธิภาพและเมื่อเกิดปัญหาสามารถที่จะกู้ไฟล์ข้อมูลกับคืนมาได้โดยวิธีการดังนี้

1) ผู้ใช้อัปโหลดดาวน์โหลดข้อมูลจากระบบ Cloud โดยผ่านกระบวนการระบุตัวตนด้วยชื่อผู้ใช้และรหัสผ่าน เพื่อเข้าสู่ระบบพื้นที่จัดเก็บข้อมูล

2) กระบวนการแยกและการเข้ารหัสข้อมูลที่ส่งมาจากผู้ใช้ ในกระบวนการนี้ ผู้วิจัยได้นำเสนอการแยกพาร์ทิชันย่อย Sub Cloud Server ออกเป็น 4 บล็อก หรือ Cloud Server (Cloud1, Cloud2, Cloud3, Cloud4) และเมื่อผู้ใช้อัปโหลดข้อมูลขึ้นในระบบ ระบบก็จะทำการแยกข้อมูลออกเป็นส่วนย่อย ๆ เพื่อแยกการจัดเก็บข้อมูลตามพาร์ทิชันย่อย Sub Cloud Server พร้อมทั้งทำการเข้ารหัสข้อมูลด้วย RSA algorithm ซึ่งเป็นการเข้ารหัสด้วยกุญแจสาธารณะ (Public Key)

3) กระบวนการจำลอง Cloud Server ที่มีการแยกพาร์ทิชันเท่ากับชุดข้อมูล Sub Cloud Server (Replica Server 1 - 4) ข้อมูลที่ถูกส่งไปเก็บยัง Replica Server จะถูกแปลงไปเป็นข้อมูลที่อยู่ในรูปแบบ Binary เมื่อมีข้อมูลสูญหายในส่วนของ Cloud Server [9] ระบบก็จะสามารถดึงข้อมูลที่อยู่ใน Replica Server ขึ้นมาได้

วิธีดำเนินการวิจัย

1. เครื่องมือการวิจัย

1.1 คอมไพเลอร์ภาษาซี Turbo C++ Version 4.0 (Windows)

1.2 Apache Web Server Version 2.4.25

2. ขั้นตอนการดำเนินการวิจัย

งานวิจัยนี้แบ่งออกเป็น 3 เป้าหมายตามวัตถุประสงค์ ดังนี้

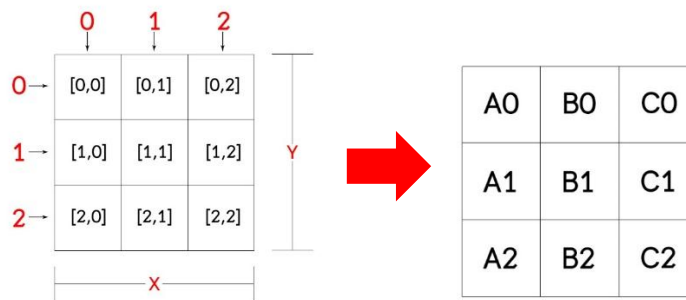
1. ศึกษารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล โดยใช้เทคนิคการ XOR

การศึกษารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลของงานวิจัยก่อนหน้านี้ [4] พบว่ารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลขนาด 3×3 โดยใช้เทคนิคการ XOR เพียงรูปแบบเดียว คือ การแบ่งไฟล์ข้อมูลออกเป็น 6 ส่วน และสร้าง Parity อีก 3 ส่วน แล้วนำไฟล์ข้อมูลทั้ง 9 ส่วนไปจัดวางในตำแหน่งรูปแบบการจัดวาง 3×3 เพียงรูปแบบเดียวเท่านั้น และการกู้ข้อมูลของงานวิจัยดังกล่าวจะสามารถทำได้ก็ต่อเมื่อข้อมูลหายไปไม่เกิน 3 กล่องที่อยู่ในแถวเดียวกัน หรือ 3 กล่องที่อยู่คอลัมน์เดียวกันเท่านั้น

2. ค้นหาแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล

เขียนโปรแกรมด้วยภาษา C++ เพื่อหารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลโดยมีขั้นตอนดังต่อไปนี้

2.1 กำหนดรูปแบบกล่องข้อมูลขนาด 3×3 ทั้งแกน X และ Y (Array 3×3) พร้อมกำหนดหมายเลขกำกับเริ่มต้นให้แต่ละกล่องข้อมูล จำนวน 9 กล่อง



ภาพที่ 2 กำหนดรูปแบบกล่องข้อมูลขนาด 3×3 ทั้งแกน X และ Y

2.2 หารูปแบบการจัดวางใหม่ กำหนดตัวแปรรอบในการวนลูป r0, r1, r2, r3, r4, r5, r6, r7, r8 จะทำทั้งหมด 9 รอบ และลดจำนวนรอบการหาเลขตามเงื่อนไขของรอบก่อนหน้าตามลำดับ และในแต่ละรอบจะต้องมีกระบวนการตรวจสอบการซ้ำของหมายเลขที่จะนำไปจัดวางในกล่องข้อมูล

2.3 ตรวจสอบการซ้ำของหมายเลขในกล่องข้อมูล การตรวจสอบการซ้ำของหมายเลข โดยการเอาจำนวนที่ได้จากรอบแรกมาเทียบกับหมายเลขที่ได้จากรอบถัดไป จำนวนรอบมากขึ้นตัวเปรียบเทียบก็จะเยอะขึ้นตามลำดับจำนวนรอบ เพื่อป้องกันการได้หมายเลขการจัดวางข้อมูลในกล่องข้อมูลที่ไม่ซ้ำกัน อย่างเช่น ถ้าได้ 0 ในรอบแรก และรอบที่สอง ถ้ามี 0 อีกก็ทำการตัดออก

2.4 ตรวจสอบการกู้คืนข้อมูล โดยเมื่อได้ข้อมูลตัวเลขจากข้อที่ 2.3 แล้ว โดยยกตัวอย่างดังนี้

| r0 = 0 | r1 = 1 | r2 = 2 | r3 = 4 | r4 = 5 | r5 = 3 | r6 = 8 | r7 = 6 | r8 = 7 |

จากข้อมูลด้านบนการทดสอบการกู้คืนข้อมูลมีกระบวนการคือ จาก 9 จำนวน เลือกข้อมูลมาทดสอบ 3 จำนวน เช่น

r0 = 0	r1 = 1	r2 = 2
r3 = 4	r4 = 5	r5 = 3
r6 = 8	r7 = 6	r8 = 7

โดยเลือกมาทดสอบแถวละ 1 ข้อมูล ก็จะได้ $r_0 = 0$, $r_3 = 4$, $r_6 = 8$ เพื่อไปทดสอบในชุดที่ 1 โดย ค่าทั้ง 3 ที่ถูกเลือกไปจะนำไปผ่านกระบวนการหาว่าจำนวนที่ 1 อยู่บรรทัดไหน จำนวนที่ 2 อยู่บรรทัดไหน จำนวนที่ 3 อยู่บรรทัดไหน และถ้าจำนวนใดจำนวนหนึ่งของทั้ง 3 จำนวนอยู่บรรทัดเดียวกัน นั่นก็หมายความว่าชุดข้อมูลที่จะนำมาทดสอบนั้น ไม่สามารถที่จะนำมาผ่านกระบวนการกู้คืนข้อมูลได้ “สรุปคือหมายเลข $r_0 = 0$, $r_3 = 4$, $r_6 = 8$ ต้องไม่อยู่บรรทัดเดียวกัน” เป็นต้น

2.5 จัดวางตำแหน่ง รูปแบบการจัดวางตำแหน่งของกล่องข้อมูล ตัวอย่างข้อมูลดังข้อที่ 2.4 ในกระบวนการจัดตำแหน่งรูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล โดยจะอ้างอิงจากรูปแบบที่เป็นค่าเริ่มต้นของกล่องข้อมูล มีลักษณะดังภาพที่ 3

0	1	2
3	4	5
6	7	8



A0	B0	C0
A1	B1	C1
A2	B2	C2

ภาพที่ 3 เริ่มต้นกล่องข้อมูล

จากตัวอย่างข้อมูลดังข้อที่ 2.4 เมื่อนำมาจัดรูปแบบการจัดวางตำแหน่งของกล่องข้อมูลแล้วจะได้กล่องข้อมูล ดังภาพที่ 4

A0	B0	C0
A1	B1	C1
A2	B2	C2



A0	B0	C0
B1	C1	A1
C2	A2	B2

ภาพที่ 4 รูปแบบของกล่องข้อมูลที่จัดตำแหน่งแล้ว

3. ทดสอบการกู้คืนข้อมูลของรูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล

ในการพัฒนาโปรแกรมเพื่อพิสูจน์การกู้คืนข้อมูลจากรูปแบบการจัดวางตำแหน่งของกล่องข้อมูล มีขั้นตอนดังต่อไปนี้

3.1 อ่านไฟล์ข้อมูลที่ใช้ในการทดสอบ เช่น .TXT, .DOCX, .PDF แล้วทำการคำนวณขนาด แบ่งไฟล์เป็นส่วน ๆ จำนวน 6 ส่วน การคำนวณขนาดนั้นเป็นการคำนวณเพื่อหาจำนวนไบต์ของไฟล์ต้นฉบับ กล่าวคือ ไฟล์ต้นฉบับต้องมีจำนวนไบต์ที่เมื่อแบ่งไฟล์ออกมาจำนวน 6 ส่วนแล้ว ทั้ง 6 ส่วนมีจำนวนไบต์ที่เท่ากัน ซึ่งต้องมีกระบวนการ Padding เพื่อเติมขนาดไบต์เข้าไปยังไฟล์ต้นฉบับ



ภาพที่ 5 แยกไฟล์เป็น 6 ส่วน

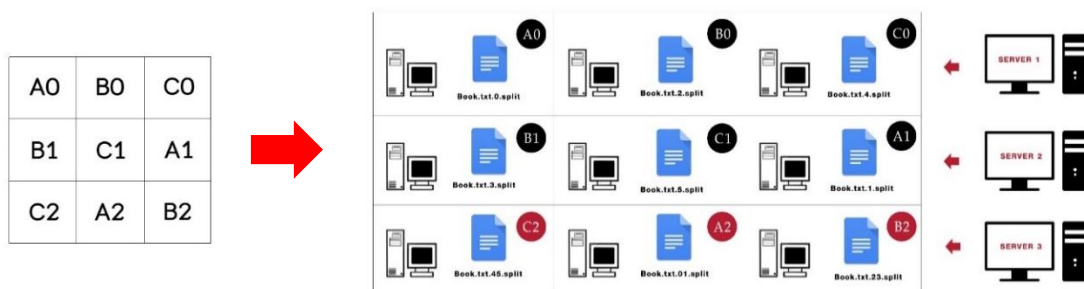
3.2 หา Parity ไฟล์ จำนวน 3 ไฟล์ ในการหา Parity ไฟล์ทำได้โดยการแบ่งไฟล์เป็น 6 ส่วนเท่า ๆ กัน แล้วนำเอาไฟล์ที่ได้มาทำการ XOR กัน เป็นคู่ทั้งหมด 3 คู่ คือ

Book.txt.0.split XOR Book.txt.1.split → Book.txt.01.split

Book.txt.2.split XOR Book.txt.3.split → Book.txt.23.split

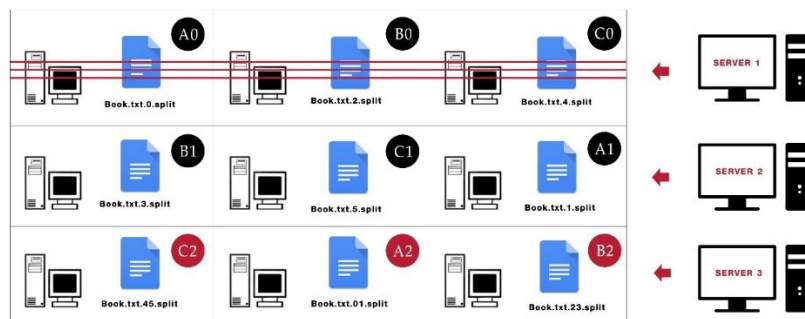
Book.txt.4.split XOR Book.txt.5.split → Book.txt.45.split

ได้ไฟล์ข้อมูลทั้ง 9 ไฟล์ แล้วก็นำไฟล์ข้อมูลไปจัดเก็บลงในกล่องข้อมูลที่ได้จากการหาในข้อที่ 2.5 จะได้ดังภาพที่ 6



ภาพที่ 6 จัดเก็บลงในกล่องข้อมูลตามรูปแบบการจัดวาง

3.3 การกู้คืนข้อมูลจากการจำลอง



ภาพที่ 7 จำลองการสูญเสียไฟล์เนื่องจากการถูกลบไฟล์

จากภาพที่ 7 สามารถกู้คืนข้อมูลที่สูญเสียไปกลับคืนมาได้โดยการนำเอาไฟล์ข้อมูลที่ยังเหลืออยู่ใน Server 2 และ Server 3 มาทำการ XOR กันเพื่อให้ได้ข้อมูลที่สูญเสียไปกลับคืนมา ดังนี้

- 1) กู้ไฟล์ข้อมูลในกล่องข้อมูลหมายเลข A0 นำไฟล์ A1 มา XOR ไฟล์ A2 (Parity)
- 2) กู้ไฟล์ข้อมูลในกล่องข้อมูลหมายเลข B0 นำไฟล์ B1 มา XOR ไฟล์ B2 (Parity)
- 3) กู้ไฟล์ข้อมูลในกล่องข้อมูลหมายเลข C0 นำไฟล์ C1 มา XOR ไฟล์ C2 (Parity)

3. สถิติที่ใช้ในการวิจัย

ได้แก่ ค่าเฉลี่ย ร้อยละ และส่วนเบี่ยงเบนมาตรฐาน

ผลการวิจัย

1. ผลการศึกษารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล โดยใช้เทคนิคการ XOR

การศึกษารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลของงานวิจัยก่อนหน้านี้ [4] พบว่ารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลขนาด 3×3 โดยใช้เทคนิคการ XOR มีเพียงเพียงรูปแบบเดียวเท่านั้น และการกู้ข้อมูลของงานวิจัยดังกล่าวจะสามารถทำได้ก็ต่อเมื่อข้อมูลหายไปไม่เกิน 3 กล่องที่อยู่ในแถวเดียวกัน หรือ 3 กล่องที่อยู่คอลัมน์เดียวกันเท่านั้น

2. ผลการค้นหารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล

จากการทดสอบการหารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลโดยการเขียนโปรแกรมด้วยโปรแกรมภาษา C++ ผู้วิจัยพบว่า มีรูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลแบบ 3×3 จำนวน 2,592 รูปแบบ จาก 46,652 รูปแบบ ซึ่งแต่ละรูปแบบที่ได้มีการจัดวางตำแหน่งของกล่องข้อมูลที่ไม่ซ้ำกัน ผู้วิจัยได้นำตัวอย่างรูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลบางส่วนมาแสดงดังนี้

```
A0 B1 C2
A1 B2 C0
A2 B0 C1
r0=0,r1=1,r2=2,r3=4,r4=5,r5=3,r6=8,r7=6,r8=7, count=1, count_nr=5
```

ตารางที่ 1 ข้อมูลค่าเฉลี่ยการหารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล

จำนวนรอบการค้นหา	กู้คืนข้อมูลได้	กู้คืนข้อมูลไม่ได้
46,652	2,592	44,060
เปอร์เซ็นต์ (ร้อยละ)	5.5	94.5

3. ผลการทดสอบกู้คืนข้อมูลที่สูญหายตามรูปแบบการจัดวางตำแหน่งของกล่องข้อมูล

จากการจำลองเครื่องแม่ข่าย จำนวน 3 เครื่อง ได้แก่ Server 1, Server 2, Server 3 และทำการทดสอบจัดวางไฟล์ข้อมูลที่ผ่านการประมวลผลแยกไฟล์ออกเป็น 6 ส่วน และไฟล์ที่ได้จากการคำนวณ Parity file 3 ไฟล์นำมาจัดวางตามรูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล ในการทดสอบโดยการจำลองสถานการณ์เครื่องแม่ข่ายล้ม ฮาร์ดดิสก์พัง และทดสอบการกู้คืนข้อมูลโดยใช้เทคนิคการ XOR ไฟล์ที่เหลือจากเครื่องแม่ข่ายที่ยังใช้งานได้ โดยผู้วิจัยได้ทำการทดสอบการกู้คืนข้อมูลตามรูปแบบการจัดวางตำแหน่งของกล่องข้อมูลจำนวน 2,592 รูปแบบ ผลของการกู้คืนข้อมูลทั้ง 2,592 รูปแบบ คือสามารถกู้คืนข้อมูลได้ 100 เปอร์เซ็นต์จากการทดสอบจำลองสถานการณ์เครื่องแม่ข่ายล้ม ดังตารางที่ 2

ตารางที่ 2 ตารางการทดสอบการกู้คืนข้อมูลตามรูปแบบการจัดวาง 2,592 รูปแบบ จำลองเครื่องแม่ข่ายล้ม

Server 1	Server 2	Server 3	ผลการกู้คืนข้อมูล
x	/	/	กู้คืนได้
/	x	/	กู้คืนได้
/	/	x	กู้คืนได้

อภิปรายผลการวิจัย

เทคนิคการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล สามารถอภิปรายผลการวิจัยได้ดังนี้

1. ในการศึกษารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลโดยใช้เทคนิคการ XOR [4] พบว่า ผู้วิจัยในงานวิจัยดังกล่าว ได้ออกแบบรูปแบบการจัดวางตำแหน่งของกล่องข้อมูลที่ใช้งานกับบาร์โค้ดสองมิติ DR Code ที่เมื่อข้อมูลบางส่วนสูญหายไป สามารถที่จะกู้คืนข้อมูลกลับคืนมาได้โดยใช้เทคนิคการ XOR ผลการศึกษาคือรูปแบบการจัดวาง 3×3 เพียงรูปแบบเดียวเท่านั้น และงานวิจัยดังกล่าวจะสามารถทำได้ก็ต่อเมื่อข้อมูลหายไปไม่เกิน 3 กล่องที่อยู่ในแถวเดียวกัน หรือ 3 กล่องที่อยู่คอลัมน์เดียวกันเท่านั้น

2. การค้นหารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูล ด้วยการเขียนโปรแกรมประกอบไปด้วยองค์ประกอบ 5 ส่วน ในการค้นหาคือ 1) สร้างกล่องข้อมูลขนาด 3×3 พร้อมกำหนดหมายเลขกำกับ 2) หารูปแบบการจัดวางใหม่ 3) ตรวจสอบการซ้ำของหมายเลขในกล่องข้อมูล ถ้าหมายเลขซ้ำก็ทำการตัดออก 4) ตรวจสอบการกู้คืนข้อมูลคือ ถ้าจำนวนใดจำนวนหนึ่งของทั้ง 3 จำนวนที่เลือกมาทดสอบอยู่บรรทัดเดียวกัน รูปแบบนั้นก็จะไม่สามารถกู้คืนข้อมูลได้ 5) จัดวางตำแหน่งรูปแบบการจัดวางตำแหน่งของกล่องข้อมูล

จากองค์ประกอบในการค้นหารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลทุกองค์ประกอบ มีการตรวจสอบด้วยเงื่อนไขอย่างรัดกุมเพื่อให้ได้กล่องข้อมูลที่สามารถกู้คืนข้อมูลได้จริง และส่งผลให้การทดสอบในการกู้ข้อมูลในกระบวนการถัดไป สามารถทำได้จริงตามทฤษฎีการกู้คืนข้อมูลโดยใช้เทคนิคการ XOR ซึ่งผลการค้นหารูปแบบการจัดวางตำแหน่งของกล่องข้อมูลพบรูปแบบการจัดวางถึง 2,592 รูปแบบที่สามารถกู้คืนข้อมูลได้ สอดคล้องกับงานวิจัยของ Wiwat Sriphum [4] ได้วิจัยเรื่องบาร์โค้ดสองมิติที่ทนทานกับการสูญหายของข้อมูลโดยใช้เทคนิคการ XOR เช่นเดียวกันแต่นำเสนอแค่ 1 รูปแบบการจัดวางเท่านั้น

3. การทดสอบกู้คืนข้อมูลที่สูญหายตามรูปแบบการจัดวางตำแหน่งของกล่องข้อมูลด้วยการเขียนโปรแกรมประกอบไปด้วยองค์ประกอบ 3 ส่วนในการทดสอบคือ 1) อ่านไฟล์ข้อมูลที่ใช้ในการทดสอบแล้วทำการคำนวณขนาด แบ่งไฟล์เป็นส่วน ๆ จำนวน 6 ส่วน ถ้าจำนวนไบต์คำนวณไม่ลงตัวให้ทำการ Padding เพื่อเติมขนาดไบต์เข้าไปแล้วค่อยแบ่งไฟล์ออกเป็น 6 ส่วน 2) หา Parity ไฟล์ จำนวน 3 ไฟล์ ในการหา Parity ไฟล์ทำได้โดยการแบ่งไฟล์เป็น 6 ส่วนเท่า ๆ กัน แล้วนำเอาไฟล์ที่ได้มาทำการ XOR กัน เป็นคู่ทั้งหมด 3 คู่ 3) การกู้คืนข้อมูลที่สูญหายไปโดยใช้เทคนิคการ XOR โดยการนำข้อมูลที่เลือกอยู่มา XOR เพื่อให้ได้ข้อมูลที่สูญหายไปกลับคืนมา

การทดสอบการกู้คืนข้อมูลตามรูปแบบการจัดวางตำแหน่งของกล่องข้อมูล จำนวน 2,592 รูปแบบ ผลของการกู้คืนข้อมูลทั้ง 2,592 รูปแบบ คือสามารถกู้คืนข้อมูลได้ 100 เปอร์เซ็นต์ จากการทดสอบจำลองสถานการณ์เครื่องแม่ข่ายล่ม สอดคล้องกับการทำงานของ Raid 5 [7] ที่ถ้ามีข้อมูลในฮาร์ดดิสก์สูญหายไประบบก็จะทำการคำนวณหาข้อมูลสูญหายไปด้วยเทคนิคการ XOR เช่นเดียวกันกับงานวิจัยชิ้นนี้

ข้อเสนอแนะ

จากการวิจัยในครั้งนี้เมื่อผู้อ่านเข้าใจถึงกระบวนการ ศึกษา ค้นหา ทดสอบรูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลขนาด 3×3 แล้ว ดังนั้นแนวความคิดถัดไปควรศึกษา ค้นหา ทดสอบรูปแบบการจัดวางของกล่องข้อมูลเพื่อกู้คืนข้อมูลที่มีขนาดมากกว่าแบบ 3×3 ว่ามีความเป็นไปได้มากน้อยเพียงใดและเมื่อเพิ่มขนาดมากกว่าแบบ 3×3 แล้วจะมีรูปแบบการจัดวางตำแหน่งของกล่องข้อมูลเพื่อกู้คืนข้อมูลที่รูปแบบ และหาวิธีการทดสอบให้มีความหลากหลายมากกว่านี้เพื่อประโยชน์ของการนำแนวคิดการงานวิจัยนี้ไปใช้งานจริงตามสถานการณ์จริงต่อไป

เอกสารอ้างอิง

- [1] ลัขณา อินชัยวงศ์. (2559). *ความคงสภาพของข้อมูล Data Integrity*. สืบค้นจาก <https://www.scribd.com/doc/8757181/บทที่-10-การคงสภาพของข้อมูล-Data-Integrity>
- [2] จตุชัย แพงจันทร์. (2553). *Master in Security* (2nd edition). นนทบุรี: ไอทีซีฯ.
- [3] Pfleeger P., Charles, and Pfleeger L. Shari. (2007) *Security in Computing* (4th ed.). upper SaddleRiver, N.J.: PrenticeHall.
- [4] Wiwat, S. (2013). *DR Code : The Two Dimensions Barcode supporting high rate data recovery*. IEEE 16th International Conference on Computational Science and Engineering 2013. (DOI : 10.1109/CSE 2013.180).
- [5] Ping-Hsun H., Ing-Yi Chen, Yu-Ting Lin, & Sy-Yen Kuo. (2004). *An XOR based Reed-Solomon algorithm for Advanced Raid System*. Proceedings 19th IEEE International Symposium 2004. (DFT2004 : 165-172).
- [6] Overclockzone. (2558). *RAID*. สืบค้นจาก <https://www.overclockzone.com/spin9/raid/>
- [7] Vmodtech. (2558). *RAID เทคโนโลยีสูงสุดของการป้องกันความเสียหายของข้อมูลและความเร็วในฮาร์ดดิสก์*. สืบค้นจาก <http://www.vmodtech.com/main/article/raid/all/1/>
- [8] Department of Computer Science and Engineering. (2015). Recover of Failure Data In Data Integrity Checking With Erasure Code and Replica Server Management System. *International Journal of Emerging Technology & Research*, 2(1).
- [9] CAT Cyfence. (2012). *Cloud Computing Security*. Retrieved from <https://www.catcyfence.com/it-security/article/cloud-computing-security>