การปกป้องข้อมูลส่วนบุคคลด้วยเทคนิคข้อมูลแฝงและข้อมูลนิรนามผ่านบริการพร็อกซี่บนระบบเครือข่าย

PROTECTING PERSONAL DATA THROUGH NETWORKED PROXY SERVICES BY PSEUDONYMISED AND ANONYMOUS TECHNIQUES

ขจรศักดิ์ ศิริศิลาศักดิ์[1]

ผศ.ดร.มัชฌิกา อ่องแตง[2]

บทคัดย่อ

โปรแกรมประยุกต์ทั้งที่มีอยู่ในปัจจุบันและที่จะได้รับการพัฒนาขึ้นในอนาคตอาจมีการประมวลผลและจัดเก็บข้อมูลส่วนบุคคลซึ่งต้องได้รับการคุ้มครองตาม พ.ร.บ.คุ้มครองข้อมูลส่วนบุคคลและกฎหมายอื่นที่เกี่ยวข้อง การปกป้องข้อมูลส่วนบุคคลจึงเป็นประเด็นสำคัญสำหรับการพัฒนาโปรแกรมต่าง ๆ อย่างไรก็ตาม การหลีกเลี่ยงการจัดเก็บและประมวลผลข้อมูลส่วนบุคคลโดยตรงด้วยการปรับเปลี่ยนโครงสร้างของข้อมูลส่วนบุคคลอาจทำให้การทำงานของโปรแกรมได้รับผลกระทบเป็นอย่างมาก เพราะบางโปรแกรมมีความสลับซับซ้อนและอาจส่งผลกระทบกับการใช้งานข้อมูลเพื่อประโยชน์ในเชิงวิเคราะห์อื่นๆด้วย

เพื่อแก้ไขปัญหาดังกล่าว ผู้เขียนจึงได้รวบรวมหลักการและตัวอย่างของ การเข้ารหัส/ถอดรหัส การทำข้อมูลให้เป็นข้อมูลแฝง (data pseudonymization) การทำข้อมูลให้เป็นข้อมูลนิรนาม (data anonymization) มาออกแบบและพัฒนา ซอฟต์แวร์แอปพลิเคชัน เพื่อประยุกต์เป็นพร็อกซี่สำหรับข้อมูลส่วนบุคคลซึ่งให้บริการปกปิดข้อมูลส่วนบุคคล ตามข้อกำหนดกฎหมาย โดยจำลองทดสอบการทำงานบนข้อมูลส่วนบุคคล เพื่อต่อยอดกับงานซอฟต์แวร์แอปพลิเคชันอื่นๆ ที่มีในระบบต่อไป ทั้งนี้ผู้เขียนได้ยกตัวอย่างการประยุกต์ใช้ระบบเพื่อส่งข้อมูลที่ผ่านการปกปิดแล้วเข้าระบบเครือข่ายผ่าน Apache Kafka ซึ่งเป็นที่นิยมและเป็นที่ยอมรับใช้งานอย่างแพร่หลายในปัจจุบัน

คำสำคัญ : การปกป้องข้อมูลส่วนบุคคล, ข้อมูลนิรนามผ่านบริการพร็อกซี่, ข้อมูลแฝงผ่านบริการพร็อกซี่

1. นักศึกษาสาขาวิศวกรรมคอมพิวเตอร์) วิทยาลัยนวัตกรรมด้านเทคโนโลยีและวิศวกรรมศาสตร์
2. อาจารย์ที่ปรึกษาสารนิพนธ์หลัก มหาวิทยาลัยธุรกิจบัณฑิตย์

ABSTRACT

Current software applications and those to be developed in the future may involve personal data processing and storage which require protection according to Personal Data Protection Act and related laws. Protection of personal data has become a prime concern for software development. Nevertheless, avoiding processing or storage of real personal data by modifying the structure of personal data may lead to extended impact because some applications are very complex. Such approach may also affect the use of the data for some further analytics.

To address this problem, the author has gathered some principles and examples of cryptography, data pseudonymization, and data anonymization to design and develop a software application that acts as a personal data proxy that provides services to conceal personal data in order to comply with personal data protection laws. It is simulated and tested on personal data to ensure that it can provide support to other software applications. In addition, the developed proxy application can also transmit the pseudonymized or anonymized data into the network via Apache Kafka, which is a popular and widely adopted today.

Keywords: Pseudonymised Proxy, Anonymous Proxy, Personal Data Protection

บทนำ

ปัจจุบันข้อมูลส่วนบุคคลของเรา รวมถึงของผู้เขียนเอง ได้ถูกนำมาใช้แพร่หลายบนเครือข่ายอินเตอร์เน็ต ตลอดจนถูกใช้ในโซเชียลมีเดีย ข้อมูลบัตรเครดิต ข้อมูลธนาคาร อื่นๆ ซึ่งในบางข้อมูลที่มีนัยสำคัญหากมีการจัดเก็บข้อมูลที่ไม่ถูกต้องอาจทำให้ มิจฉาชีพฉกฉวยหรือนำไปใช้ประโยชน์ในทางที่ผิดได้ รวมถึงปัจจุบันซอฟต์แวร์แอปพลิเคชันใหม่ๆ ได้ขยายตัวเพิ่มขึ้นอย่างต่อเนื่อง ซึ่งแน่นอนก็จะมีการใช้ ข้อมูลส่วนบุคคลเพิ่มมากขึ้นตามมา ด้วยการสนับสนุนจากทางภาครัฐและเอกชน ในการใช้ข้อมูลส่วนบุคคล เพื่อต่อยอดทางธุรกิจ หรือต่อยอดในการทำงานของหน่วยงานต่างๆ ให้เป็นไปตามข้อกำหนด สำนักงานคณะกรรมการคุ้มครองข้อมูลส่วนบุคคล จึงได้กำหนดกฎหมายการคุ้มครองข้อมูลอย่างเหมาะสม ตาม พ.ร.บ. คุ้มครองข้อมูลส่วนบุคคล พ.ศ. 2562 และ มีผลบังคับใช้ เมื่อวันที่ 27 พฤษภาคม 2562 หลังจากที่ถูกเลื่อนออกมาให้มีผลบังคับใช้ในวันที่ 1 มิถุนายน 2565 (สำนักงานคณะกรรมการคุ้มครองข้อมูลส่วนบุคคล, 2562) ปัจจุบันโปรแกรมประยุกต์ที่ถูกพัฒนาหากมีกฎหมายดังกล่าวบังคับใช้ก็จะมีการเปลี่ยนแปลงโครงสร้างของข้อมูลจึงหลีกเลี่ยงไม่ได้ที่จะเกิดผลกระทบกับโปรแกรมที่ทำงานอยู่ รวมถึงโปรแกรมที่กำลัง

พัฒนา ซึ่งทำให้ส่งผลกระทบกับการทำงานของโปรแกรม เป็นอย่างมาก บางโปรแกรมเป็นไปได้ยากหากมีการนำมาใช้เลยโดยไม่ศึกษา ทำให้มีความเสี่ยงสูงและมีโอกาสสร้างความเสียหายให้กับการทำงานของระบบได้

จากปัญหาดังกล่าว ผู้เขียนได้บทสรุปว่าต้องการแก้ปัญหา จึงได้ออกแบบและพัฒนาซอฟต์แวร์แอปพลิเคชันนี้เพื่อจำลอง ทดสอบ ข้อมูลส่วนบุคคลที่ถูกเปลี่ยนแปลงเพื่อต่อยอดในการศึกษาใช้งานกับซอฟต์แวร์แอปพลิเคชัน ที่มีในระบบต่อไป ซึ่งได้นำหลักการและตัวอย่าง การเข้ารหัส ถอดรหัส การใช้งานนามแฝง Pseudonymised ข้อมูลนิรนาม data anonymized มาประยุกต์เป็นพร็อกซี่ข้อมูลส่วนบุคคลให้บริการงานเข้ารหัสปกปิดข้อมูลที่แท้จริง เพื่อป้องกันข้อมูลส่วนบุคคลตามกฎหมายข้อกำหนด ทั้งนี้ผู้เขียนได้ยกตัวอย่างการประยุกต์การส่งข้อมูลเข้าระบบเครือข่ายซึ่งมีการใช้งานอย่างแพร่หลายในปัจจุบันด้วย Apache Kafka ที่กำลังเป็นที่ยอมรับและสามารถต่อยอดด้วยการรองรับเทคโนโลยีปัจจุบันและอนาคตได้หลากหลายอีกด้วย

วัตถุประสงค์การวิจัย

1. ใช้เป็นระบบบันทึกจัดเก็บข้อมูลส่วนบุคคลเพื่อนำมาเปลี่ยนข้อมูลส่วนบุคคลด้วยเทคนิคข้อมูลแฝง Pseudonymised และ ข้อมูลนิรนาม data anonymized
2. ใช้เป็นโปรแกรมต้นแบบในการทำ proxy ข้อมูลส่วนบุคคล ผ่านระบบเครือข่าย
3. ใช้เป็นโปรแกรม Data Encryption และ Decryption ได้
4. ใช้เป็นโปรแกรมสื่อสารเปรียบเทียบ สำหรับ Confirm requirement ในการจัดการข้อมูลส่วนบุคคล กับลูกค้าได้
5. ใช้เป็นโปรแกรมสื่อสารเปรียบเทียบ สำหรับ Developer กับ Designer
6. ใช้เป็นโปรแกรมทดสอบ prototype ข้อมูลที่แสดงผล หลังจากมีการทำ ข้อมูลแฝง Pseudonymised และ ข้อมูลนิรนาม data anonymized
7. ใช้ลดจำนวนการใช้งานโปรแกรมออนไลน์สำเร็จรูปในการเข้ารหัสถอดรหัสออนไลน์เพื่อป้องกันข้อมูลรั่วไหล

ขอบเขตและสถานที่ทำงานวิจัย

สามารถจำลองระบบเครือข่ายและประยุกต์ใช้โปรแกรมการแปลงข้อมูลบุคคลเป็น ข้อมูลตัวอย่างประเภทนามแฝง Pseudonymised ข้อมูลนิรนาม Data anonymizedโดยสามารถปรับใช้หลายอัลกอริธึมได้

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในงานวิจัยครั้งนี้ ผู้วิจัยได้ทำการศึกษาค้นคว้าหาข้อมูลทฤษฎี และงานวิจัยที่เกี่ยวข้องโดยมีลำดับเนื้อหาในส่วนของความหมายข้อมูลส่วนบุคคล การปกปิดข้อมูล ส่วนบุคคลของผู้ประกอบการ หรือ องค์กร รวมถึงการจำแนกกฎหมายที่ถูกกำหนดใช้ ที่มาของทฤษฎีการปกปิดข้อมูลส่วนบุคคล ตลอดจนตัวอย่างและเทคนิคที่นำมาประยุกต์ใช้กับงานวิจัย ดังนี้

1. ขอบเขตข้อมูลส่วนบุคคล

ศูนย์วิจัยกฎหมายและการพัฒนา คณะนิติศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย,(2561) ได้กล่าวถึงแนวคิดข้อมูลส่วนบุคคลไว้ว่า ข้อมูลส่วนบุคคลได้ถูกกำหนดให้เป็นข้อมูลใด ๆ ที่เกี่ยวข้องกับบุคคลธรรมดาที่ระบุหรือระบุตัวตนได้ บุคคลที่สามารถระบุตัวตนได้คือบุคคลที่สามารถระบุได้โดยตรงหรือโดยอ้อม โดยเฉพาะอย่างยิ่งโดยการอ้างอิงถึงหมายเลขประจำตัวหรือปัจจัยหนึ่งหรือหลายปัจจัยที่จำเพาะต่อเอกลักษณ์ทางร่างกาย สรีรวิทยา จิตใจ เศรษฐกิจ วัฒนธรรม หรือสังคมของเขา โดยมีตัวอย่างข้อมูลส่วนบุคคลและไม่ใช่ข้อมูลส่วนบุคคล ดังตารางที่ 2.1 และตารางที่ 2.2

ตารางที่ 2.1 แสดงตัวอย่างข้อมูลที่เป็นข้อมูลส่วนบุคคล

ลำดับ	รายการ
1	ชื่อ-นามสกุล หรือชื่อเล่น
2	เลขประจำตัวประชาชน, เลขหนังสือเดินทาง, เลขบัตรประกันสังคม, เลขใบอนุญาต ขับขี่, เลขประจำตัวผู้เสียภาษี, เลขบัญชีธนาคาร, เลขบัตรเครดิต (การ เก็บเป็นภาพส าเนาบัตรประชาชนหรือสำเนาบัตรอื่นๆที่มีข้อมูลส่วนบุคคลที่กล่าว มาย่อมสามารถใช้ระบุตัวบุคคลได้โดยตัวมันเอง จึงถือเป็นข้อมูลส่วนบุคคล)
3	ที่อยู่, อีเมล์, เลขโทรศัพท์
4	ข้อมูลอุปกรณ์หรือเครื่องมือ เช่น IP address, MAC address, Cookie ID
5	ข้อมูลทางชีวมิติ เช่น รูปภาพใบหน้า, ลายนิ้วมือ, ฟิล์มเอกซเรย์, ข้อมูลสแกนม่านตา, ข้อมูลอัตลักษณ์เสียง, ข้อมูลพันธุกรรม
6	ข้อมูลระบุทรัพย์สินของบุคคล เช่น ทะเบียนรถยนต์, โฉนดที่ดิน
7	ข้อมูลที่สามารถเชื่อมโยงไปยังข้อมูลข้างต้นได้ เช่น วันเกิดและสถานที่เกิด, เชื้อ ชาติ, สัญชาติ, น้ำหนัก, ส่วนสูง, ข้อมูลตำแหน่งที่อยู่ (location), ข้อมูลการแพทย์, ข้อมูลการศึกษา, ข้อมูลทางการเงิน, ข้อมูลการจ้างงาน
8	ข้อมูลหมายเลขอ้างอิงที่เก็บไว้ในไมโครฟิล์ม แม้ไม่สามารถระบุไปถึงตัวบุคคลได้ แต่หากใช้ร่วมกับระบบดัชนีข้อมูลอีกระบบหนึ่งก็จะสามารถระบุไปถึงตัวบุคคลได้ ดังนั้นข้อมูลในไมโครฟิล์มจึงเป็นข้อมูลส่วนบุคคล
9	ข้อมูลการประเมินผลการทำงานหรือความเห็นของนายจ้างกับลูกจ้าง
10	ข้อมูลบันทึกต่างๆ ที่ใช้ติดตามตรวจสอบกิจกรรมของบุคคล เช่น log file
11	ข้อมูลที่สามารถใช้ในการค้นหาข้อมูลส่วนบุคคลอื่นในอินเทอร์เน็ต

ตารางที่ 2.2 แสดงตัวอย่างข้อมูลที่ไม่เป็นข้อมูลส่วนบุคคล

ลำดับ	รายการ
1	เลขทะเบียนบริษัท
2	ข้อมูลสำหรับการติดต่อทางธุรกิจที่ไม่ได้ระบุถึงตัวบุคคล เช่น หมายเลขโทรศัพท์ หรือแฟกซ์ที่ทำงาน, ที่อยู่สำนักงาน, อีเมล์ที่ใช้ในการทำงาน, อีเมล์ของบริษัท เช่น info@companay.com เป็นต้น
3	ข้อมูลนิรนาม (Anonymized Data) หรือข้อมูลแฝง (Pseudonymous Data) หมายถึง ข้อมูลหรือชุดข้อมูลที่ถูกทำให้ไม่สามารถระบุตัวบุคคลได้อีกโดยวิธีการทางเทคนิค
4	ข้อมูลผู้ตาย

2. กฎหมายการคุ้มครองข้อมูลส่วนบุคคล

พระราชบัญญัติคุ้มครองข้อมูลส่วนบุคคล พ.ศ. 2562 ได้ระบุกฎหมายข้อมูลส่วนบุคคลไว้ว่า “บุคคลย่อมมีสิทธิในความเป็นอยู่ส่วนตัว เกียรติยศ ชื่อเสียง และครอบครัว การกระทำอันเป็นการละเมิดหรือกระทบต่อสิทธิของบุคคลตามวรรคหนึ่ง หรือ การนำข้อมูลส่วนบุคคลไปใช้ประโยชน์ไม่ว่าในทางใดๆ จะกระทำมิได้เว้นแต่โดยอาศัย อำนาจตามบทบัญญัติแห่งกฎหมายที่ตราขึ้นเพียงเท่าที่จำเป็นเพื่อประโยชน์สาธารณะ” ส่วนใน ประมวลกฎหมายแพ่งและพาณิชย์ มีบทบัญญัติ ที่เกี่ยวข้องอยู่ 2 มาตรา ดังนี้ “มาตรา 420 ผู้ใดจงใจหรือประมาทเลินเล่อ ทำต่อบุคคลอื่น โดยผิดกฎหมายให้ เขาเสียหายถึงแก่ชีวิตก็ดี แก่ร่างกายก็ดี อนามัยก็ดี เสรีภาพก็ดี ทรัพย์สินหรือสิทธิอย่าง หนึ่งอย่างใดก็ดี ท่านว่า ผู้นั้นทำละเมิด จำต้องใช้ค่าสินไหมทดแทนเพื่อการนั้น” และ “มาตรา 421 การใช้สิทธิซึ่งมีแต่จะให้เกิดเสียหายแก่บุคคลอื่นนั้น ท่านว่าเป็น การอันมิชอบด้วยกฎหมาย” จากประมวลกฎหมายแพ่งและพาณิชย์ มาตรา 420 และมาตรา 421 รวมถึงรัฐธรรมนูญแห่งราชอาณาจักรไทย พ.ศ.2560 มาตรา 32 มีกฎเกณฑ์การลงโทษ ผู้ประกอบการหรือองค์กรจึงควรต้องเพิ่มการระมัดระวัง รวมถึงการกำชับพนักงานให้คำนึงถึงความสำคัญในการเก็บข้อมูล หรือใช้ข้อมูลส่วนบุคคล จากกฎหมายข้างต้นของทั้ง GDPR และ PDPA มีบทลงโทษที่เห็นได้ชัดเจน จึงจำเป็นต้องตระหนักและมีความรอบครอบอย่างมากในการจัดเก็บข้อมูลส่วนบุคคล รวมถึงใช้เทคนิค ตามตารางที่ 2.2 ข้อ 3 “ข้อมูลนิรนาม (Anonymized Data) หรือข้อมูลแฝง (Pseudonymous Data)” ต่อมาจากการศึกษาประเภทของเทคนิคการทำ ข้อมูลนิรนาม (Anonymized Data) หรือ ข้อมูลแฝง (Pseudonymous Data) มีจำแนกวิธีดังตารางที่ 2.3

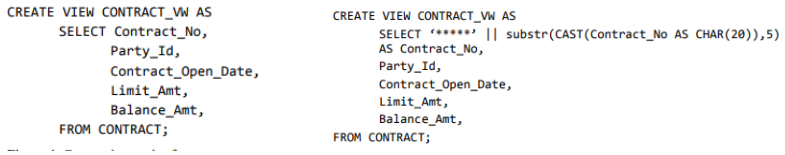
ตารางที่ 2.3 แสดงรายการเทคนิคการทำข้อมูลนิรนาม (Anonymized Data) หรือ ข้อมูลแฝง (Pseudonymous Data)

ลำดับ	รายการ	รายละเอียด
1.	การแทนที่ไดเร็กทอรี (Directory replacement)	การแก้ไขชื่อของบุคคลโดยที่ยังคงความสอดคล้องระหว่างค่าต่างๆ ไว้
2.	การสุ่ม (Scrambling)	การทำให้งงงวย บางครั้งกระบวนการสามารถย้อนกลับได้
3.	การกำบัง (Masking)	ซ่อนส่วนหนึ่งของข้อมูลด้วยอักขระสุ่ม ตัวอย่างเช่น การระบุนามแฝงด้วยข้อมูลประจำตัว
4.	การทำให้ไม่ระบุตัวตนบุคคล (Personalized anonymization)	การทำให้ไม่ระบุตัวตนแบบกำหนดเอง
5.	การเบลอ(Blurring)	ทำให้ความหมายของค่าข้อมูลล้าสมัยหรือไม่สามารถระบุค่าข้อมูลซ้ำได้
6.	การใช้นามแฝงรวมถึงการเข้ารหัสข้อมูล (Pseudonymization methods include data encryption)	การปลี่ยนข้อมูลต้นฉบับเป็นข้อความเข้ารหัส สามารถย้อนกลับได้ด้วยคีย์ถอดรหัส
7.	การปิดบังข้อมูล(Data masking)	การปกปิดข้อมูลในขณะที่ยังคงความสามารถในการใช้งานสำหรับฟังก์ชันต่างๆ

3. การปกปิดข้อมูลสำหรับผู้ประกอบการ หรือองค์กร

Shanika Wickramasinghe (2564)ได้กล่าวไว้ว่า สำหรับองค์กรที่ต้องการความน่าเชื่อถือสูงสุด ต้องการปกปิดข้อมูลสำคัญกับพนักงานภายในที่ไม่ได้รับอนุญาติ เพื่อป้องกันข้อมูลรั่วไหลออกไปภายนอกองค์กร ทั้งที่ตั้งใจและไม่ตั้งใจก็ตาม การทำ Data Masking จะช่วยป้องกันข้อมูลในส่วนนี้ได้ เช่นหน่วยงานที่จะต้องมีพนักงานหลายคนต้องเข้าถึงข้อมูลนี้ได้ หรือพนักงานจ้างชั่วคราว หรือ Outsource ก็จะถูกปิดบังข้อมูลจริงไว้ให้เห็นเฉพาะข้อมูลแฝงเท่านั้น สำหรับฝ่าย Software Development บางครั้งต้องการทดสอบการทำงานของ Application ด้วยข้อมูลใกล้เคียงข้อมูลจริงให้มากที่สุด ซึ่งไม่จำเป็นต้องใช้ข้อมูลจริงในการทดสอบ ก็สามารถทำ Data Masking กับข้อมูลเพื่อนำไปทดสอบก่อนที่จะ Production จริง ด้วยความสมเหตุสมผลและความเหมาะสมกับองค์กร ทางผู้เขียนสามารถเลือกเทคนิค Data Masking รวมถึงเทคนิค Data Encryption พื้นฐาน มาใช้ในงานวิจัยนี้ตัดสินใจเลือกเทคนิคลำดับที่ 6 และ 7 ดังตารางที่ 2.3 เรื่องทฤษฎีการปกปิดข้อมูลโดยใช้มุมมองบนตารางจริง เมื่อเราต้องการซ่อนข้อมูลบางส่วนจากผู้ใช้ เราสามารถสร้างเลเยอร์บนข้อมูลหลักได้ โดยเลเยอร์ที่สร้างมุมมองด้านบนของตารางจริง ในมุมมองเราสามารถกำหนดตารางที่เรากำลังใช้และวิธีการและคอลัมน์ที่เราเลือกจากตาราง วิธีง่ายๆในการซ่อนข้อมูลจากผู้ใช้คือการสร้างมุมมองโดยไม่มีคอลัมน์ที่ต้องซ่อน การทำเช่นนี้เราต้องรู้ว่าข้อมูล

สำหรับคอลัมน์อื่นจะยังใช้งานได้หลังจากนั้น ตัวอย่างเช่น ถ้าเราซ่อนมูลค่าจำนวนเงินจากข้อมูลการทำธุรกรรม ไม่น่าจะมีประโยชน์ แต่เมื่อเราซ่อนหมายเลขประกันสังคมลูกค้าจากตารางการทำธุรกรรมก็อาจจะเป็นที่ยอมรับ ซ่อนค่านิยมจำเป็นต้องวิเคราะห์ข้อมูลและข้อกำหนดบางประการเพื่อให้เข้าใจถึงความจำเป็นของข้อมูล แสดงดังภาพที่ 2.1



```
CREATE VIEW CONTRACT_VW AS
      SELECT Contract_No,
            Party_Id,
            Contract_Open_Date,
            Limit_Amt,
            Balance_Amt,
      FROM CONTRACT;
```

```
CREATE VIEW CONTRACT_VW AS
      SELECT '******' || substr(CAST(Contract_No AS CHAR(20)),5)
      AS Contract_No,
      Party_Id,
      Contract_Open_Date,
      Limit_Amt,
      Balance_Amt,
FROM CONTRACT;
```

ภาพที่ 2.1 แสดงการสร้างข้อมูลก่อน-หลังการใช้งานเทคนิคการปกปิดข้อมูล

Input data:

USER_ID	EN_FIRSTNAME	POSITION	TELEPHONE	ACTIVE_FLAG
210XX	First	DEVELOPER	0233456788	Y
210YY	Risa	SUPERVISER	021234555585	Y
210Y1	Michael	SUPERVISER	0276545555-7	Y
210Y2	Pitsanu	DEVELOPER	0276545558-10	Y
210ZZ	Thitiwat	DEVELOPER	063241xxxx	Y

MS_MAPPING_OUT

ตำแหน่ง Row1 |ตำแหน่ง Row2 | ตำแหน่ง Row3 |ตำแหน่ง Row4 |ตำแหน่ง Row5

005 003 003 | 002 004 004 | 003 001 005 | 004 005 001 | 001 002 002

ภาพที่ 2.2 แสดงตารางการ Shuffle masking (input) และบันทึกตำแหน่งลงตาราง

1. ทฤษฎีการปกปิดข้อมูลโดย Email masking การทำ MARKER ของ Email เมื่อนำมาใช้ในภาษา Java ทำการ Split data ลักษณะโดยการนำข้อมูล Email ที่ต้องการโดยใช้ @ มาแบ่งข้อมูลเป็นสองส่วน ทำการแปลงส่วนแรก ให้เป็นอักษร ‘X’ แล้วจึงต่อ String [0] + ‘@’ + String [2] Ex. Input: time@company.com Output: XXXX@company.com
2. ทฤษฎีการปกปิดข้อมูลโดย Shuffle masking การทำ Shuffle masking ของโปรแกรมสามารถกำหนดทำได้ 3 field คือ Position หมายถึงตำแหน่งงาน ,Telephone หมายถึงเบอร์โทรศัพท์ และ EnFirstname ชื่อจริง ซึ่งทั้ง 3 ส่วนสามารถ Random สลับค่ากันโดยมีการ เก็บบันทึกตำแหน่งหลักของแต่ละ Row ไว้ด้วยรหัส และบันทึกลง ตาราง MS_MAPPING_OUT จากนั้น random data สำหรับทำ Shuffle เพื่อแสดงผล output ดังภาพที่ 2.3

Output data :

USER_ID	EN_FIRSTNAME	POSITION	TELEPHONE	ACTIVE_FLAG
210XX	Thitiwat	SUPERVISER	0276545555-7	Y
210YY	Risa	DEVELOPER	0276545558-10	Y
210Y1	Michael	DEVELOPER	0233456788	Y
210Y2	Pitsanu	DEVELOPER	063241xxxx	Y
210ZZ	First	SUPERVISER	021234555585	Y

ภาพที่ 2.3 แสดงตารางการ Shuffle masking (output)

3. ทฤษฎีการปกปิดข้อมูลโดยวิธี Generalized masking การทำ GENERALIZED เป็นการนำมาประยุกต์ใช้โดยใช้ภาษา Java ซึ่งเป็นการเปลี่ยนแปลง ข้อมูลที่ไม่สามารถย้อนกลับได้เพื่อนำไปเป็นข้อมูลในการทดสอบหรือ Mock data เพื่อปกปิดข้อมูลจริงและนำไปใช้งานต่อได้อย่างปลอดภัย โดยตัวอย่างนี้คือ การใช้งาน PostgreSQL Anonymizer 0.5: ประเภท Generalization

```
SELECT * FROM patient;
     ssn     | firstname | zipcode |   birth    |    disease
-------------+-----------+---------+------------+--------------
 253-51-6170 | Alice     |   47012 | 1989-12-29 | Heart Disease
 091-20-0543 | Bob       |   42678 | 1979-03-22 | Allergy
 565-94-1926 | Caroline  |   42678 | 1971-07-22 | Heart Disease
 510-56-7882 | Eleanor   |   47909 | 1989-12-15 | Acne
```

ภาพที่ 2.4 แสดงตารางข้อมูลก่อนการทำ Generalized masking

```
CREATE MATERIALIZED VIEW generalized_patient AS
SELECT
  'REDACTED'::TEXT AS firstname,
  anon.generalize_int4range(zipcode,1000) AS zipcode,
  anon.generalize_daterange(birth,'decade') AS birth,
  disease
FROM patient;
```

ภาพที่ 2.5 แสดงการใช้ข้อมูล Generalized masking

```
SELECT * FROM generalized_patient;
 firstname |    zipcode    |          birth          |    disease
-----------+---------------+-------------------------+--------------
 REDACTED  | [47000,48000) | [1980-01-01,1990-01-01) | Heart Disease
 REDACTED  | [42000,43000) | [1970-01-01,1980-01-01) | Allergy
 REDACTED  | [42000,43000) | [1970-01-01,1980-01-01) | Heart Disease
 REDACTED  | [47000,48000) | [1980-01-01,1990-01-01) | Acne
```

ภาพที่ 2.6 แสดงตารางข้อมูลหลังทำการ Generalized masking

การแทนที่ข้อมูลด้วยค่าที่กว้างกว่าและแม่นยำน้อยกว่า ตัวอย่างเช่น แทนที่จะพูดว่า "ป๊อบอายุ 28 ปี" คุณสามารถพูดว่า "ป๊อบอายุระหว่าง 20 ถึง 30 ปี"

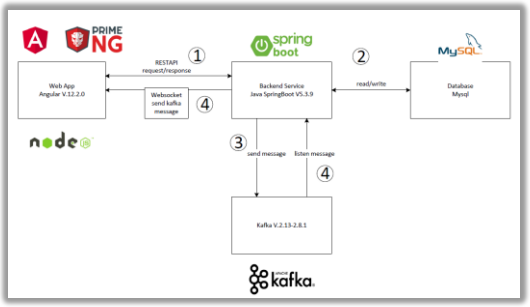
ปัจจุบันมีการนำข้อมูลนิรนาม (Anonymized Data) มาประยุกต์ใช้อย่างแพร่หลาย Proxy websites ซึ่งมีคุณสมบัติปกปิดความเป็นตัวตน,ปกปิด IP address ,ปกปิด location , ปกปิดข้อมูลส่วนบุคคล เมื่อมีการใช้งานมากขึ้นก็จะมีจำนวน Proxy websites มากขึ้นทุกวัน จึงสามารถมองเห็นแนวโน้มว่า การนำเทคนิค Anonymous มาใช้อย่างแพร่หลายและผู้คนให้ความสนใจกับการปกปิดข้อมูลเป็นอย่างมากเนื่องจากปัจจุบันมีมิจฉาชีพ,ผู้ก่อการร้ายหรือ ผู้ไม่หวังดีอยู่รอบตัวเราซึ่งมีความเสี่ยงเพิ่มสูงขึ้นทุกปี

การออกแบบและพัฒนาระบบ

จากแนวคิดและทฤษฎีที่เกี่ยวข้องจากบทที่แล้ว บทนี้ทางผู้เขียนจะกล่าวถึงสถาปัตยกรรมระบบ การออกแบบฐานข้อมูล การออกแบบหน้าจอ การส่งข้อมูลเข้าสู่เครือข่าย ตลอดจนแนวคิดการเลือกใช้ รวมถึงขั้นตอนการพัฒนาระบบ

สถาปัตยกรรมระบบ (System Architecture)

แนวคิดการออกแบบสถาปัตยกรรม มีข้อจำกัดความถนัดของผู้เขียนโปรแกรมคือ ระบบมีการทำงานด้วยภาษาจาวา สามารถส่งข้อมูลผ่านเครือข่ายด้วย JSON และสามารถทำงานร่วมกับหน้าจอโดยใช้ JavaScript รวมถึงเป็นภาษาที่ทันสมัยและมีใช้พัฒนาอยู่อย่างต่อเนื่องในปัจจุบัน โดยมีภาพรวมการทำงานของระบบ ดังภาพ ที่ 3.1



ภาพที่ 3.1 ภาพรวมการทำงานของระบบ

จากภาพรวมของระบบ ตามภาพที่ 3.1 เป็นการแนะนำ Framework ที่ใช้ เทคโนโลยี รวมถึงภาษาที่ใช้ ซึ่งมีการทำงานประกอบด้วย Front-end ,Back-end รวมถึงการส่งข้อมูล มีรายละเอียดโดยมี Front-end เป็นการทำงานโดยใช้ Framework Angular Version 12.2.0 รวมถึงใช้ PrimeNG ซึ่งทำงานอยู่บน NodeJS ใช้ภาษา JavaScript ในการจัดการและความสวยงามของหน้าจอ ส่วน Back-end เป็นการทำงานโดยใช้ Framework Spring-Boot ร่วมกับ ฐานข้อมูล MySQL และใช้ภาษา Java

ในการทำงาน ซึ่งใช้ Format type JSON ในการสื่อสารส่งข้อมูลไปยังหน้าจอ ระบบเครือข่าย รวมถึงค้นหา บันทึก แก้ไข และลบ ข้อมูลของระบบโดยโครงสร้างได้มีการเชื่อมต่อไปยัง Distributed message Queue โดยใช้ Framework Apache Kafka ในการทำงาน

การส่งข้อมูลตามลำดับหมายเลข หมายเลขที่ 1 เป็นการรับ และส่งข้อมูลจากหน้าจอ ส่งต่อไปยัง Services ที่ทาง Application ได้เตรียมไว้ให้ จากนั้นข้อมูลถูกส่งไปยัง หมายเลข 2 เพื่อทำหน้าที่ ค้นหา บันทึก แก้ไข หรือ ลบ ข้อมูลในฐานข้อมูล ตามที่ผู้ใช้งานร้องขอ ตามกระบวนการข้อมูลจะถูกส่งเข้าสู่ระบบ ด้วยหมายเลข 3 ซึ่งทำการ กดปุ่ม Send data ข้อมูล JSON Output ที่ได้จากการประมวลผลโดย Service ที่เตรียมไว้ให้ โดยข้อมูลจะถูกส่งไปยัง Topic บน Kafka ที่ได้ทำการ Configure ไว้ และ เมื่อ Kafka ได้รับ Message ที่ผูกไว้กับ Topic ที่กำหนด จึงได้ ส่งต่อข้อมูลดังกล่าวไปยัง เครือข่ายของ Kafka เพื่อให้ Listeners ทุกตำแหน่งได้รับ message หรือ data เดียวกัน ดังหมายเลข 4.

การติดตั้ง Hardware และ Software

การออกแบบ Web Application แบ่งออกเป็น 2 ส่วน คือ ส่วน Hardware และส่วน Software เพื่อสามารถพัฒนา Web Applicationได้อย่างมีประสิทธิภาพ ด้วยระยะเวลาที่จำกัด ในการศึกษา ผู้เขียนจึงจำลองเครื่องแม่ข่าย และฐานข้อมูลไว้บนเครื่องเดียวกัน โดยกำหนดลักษณะของอุปกรณ์โดยยึดเครื่องจำลองของผู้เขียนเอง ซึ่งมีรายละเอียดความต้องการของ Hardware ที่ใช้งาน ดังนี้ การประมวลผล (CPU) : ชิปเซต Intel (R) Core™ (i7 ,2.30GHz) ,หน่วยความจำ (RAM) : 16GB (1x16GB) RDIMM, หน่วยบันทึกข้อมูล (Hard disk) : 1TB SSD และความต้องการ ขั้นต่ำ Software ที่ใช้งานในระบบ จะเป็นรายละเอียดการใช้งาน ภาษาและ Framework ที่ใช้พัฒนา Java Spring-boot, NodeJS, Angular+PirmeNG ระบบจัดการ deploy / Windows PowerShell Command line รวมถึง ระบบฐานข้อมูล MySql version 8.0.20 ส่วนเครื่องมือพัฒนาในการจัดการ API ผู้เขียรเลือกใช้ Eclipse ที่ทำงานอยู่บน Java เนื่องจากไม่เปลืองทรัพยากรณ์สูง สำหรับเครื่องมือพัฒนาจัดการงาน Front-end, Angular + PrimeNG ผู้เขียนเลือกใช้ Visual Studio Code เหตุผลเช่นเดียวกัน คือ ไม่เปลืองทรัพยากรณ์ในการทำงานสูง

การใช้งาน Web application บนเครื่องแม่ข่าย

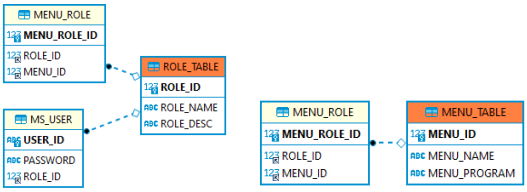
การทำงานบนเครื่องแม่ข่ายโดยใช้การทำงานของ Web application เพื่อใช้งานในการประมวลผล และแสดงข้อมูลบนเครื่องผู้ใช้งาน จะมีขั้นตอนการพัฒนาระบบ Back-End โดยใช้ภาษา Java ทำงานภายใต้ Framework Spring Boot เพื่อสร้าง Service เพื่อเตรียมพร้อม API ให้พร้อมทำงานอยู่บนระบบฐานข้อมูล MySQLสิ่งที่จำเป็นต้องมี ดังนี้ ความรู้พื้นฐานเว็บแอปพลิเคชัน

เบื้องต้น,ความรู้พื้นฐานการสร้าง Web Service เบื้องต้น,ซอฟต์แวร์: Java, NodeJS ,Apache Kafka, ฐานข้อมูล: MySQL version 8.0.20

การออกแบบระบบส่วนของฐานข้อมูล

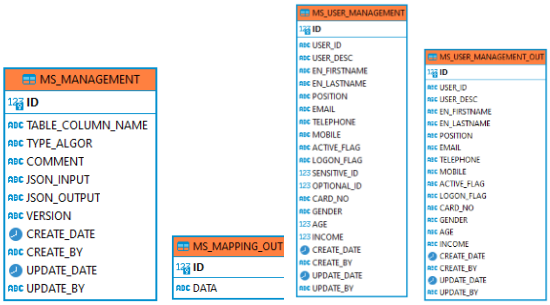
ฐานข้อมูลของระบบจะมีการออกแบบตามลักษณะและรูปแบบการใช้งานซึ่งมีตารางที่สำคัญรายละเอียดดังต่อไปนี้

จากภาพที่ 3.2 แสดงความสัมพันธ์ของระบบเข้าสู่ระบบ รวมถึงความสัมพันธ์กันกับหน้าที่ของแต่ละ User ซึ่งมีรายละเอียดแต่ละตารางดังนี้ ตาราง MS_USER ใช้ในการทำงาน login หน้าแรก เพื่อให้ทราบว่ามีสิทธิ์ในการใช้งานระบบ ซึ่งมีการเก็บ USER_ID มีการเก็บเป็นตัวอักษร รวมถึง PASSWORD ที่ผ่านการเข้ารหัส ซึ่งรวมกับตาราง MENU_ROLE ใช้ในการทำงานเก็บข้อมูลหน้าที่ของ User นั้นๆ ซึ่งจะรวมถึงสิทธิ์การเข้าถึงหน้าเมนู ที่สามารถใช้งานได้ และมีความสัมพันธ์ กับ ตาราง ROLE_TABLE ใช้เก็บข้อมูลของ หน้าที่การทำงาน



ภาพที่ 3.2 แสดงตารางที่เกี่ยวข้องกับระบบ Login รวมถึงความสัมพันธ์ของหน้าที่ของแต่ละ User

ตาราง MS_MANAGEMENT ใช้ในการจัดระบบ ซึ่งเป็นตารางหลักสำหรับใช้ในการ Configure ข้อมูลดังภาพที่ 3.4 มี column ที่มีความหมายดังต่อไปนี้ (Column:Table_column_name) ใช้เก็บข้อมูลชื่อตารางพร้อม Column เพื่อใช้ในการ Configure ได้อย่างถูกต้อง ตามตารางและ Column ที่ได้บันทึกไว้ (Column: Type_Algor) ใช้เก็บข้อมูลแสดงการเลือกชนิดของการเข้ารหัสหรือการทำ data masking (Column: Comment) ใช้เก็บข้อมูลรายละเอียดของ Column (Column:JSON_INPUT) ใช้เก็บข้อมูล Parameter ที่ใช้ในการ input ข้อมูลชนิด Camel (Column:VERSION) ใช้เก็บข้อมูลของ version ที่บันทึก (Column: CREATE_DATE, CREATE_BY, UPDATE_DATE, UPDATE_BY) ใช้บันทึกข้อมูลผู้ทำการเปลี่ยนแปลงข้อมูล ส่วน ตาราง MS_MAPPING_OUT ใช้เก็บข้อมูลแสดงความสัมพันธ์ สำหรับการใช้งาน Shuffle ข้อมูล ดังภาพที่ 3.4 ตาราง MS_USER_MANAGEMENT ใช้ในการเก็บข้อมูลส่วนบุคคลก่อนการเข้ารหัสหรือก่อนการปกปิดข้อมูล ดังภาพที่ 3.3 และ ตาราง MS_USER_MANAGEMENT_OUT ใช้ในการเก็บข้อมูลส่วนบุคคลหลังการเข้ารหัสหรือหลังการปกปิดข้อมูล ดังภาพที่ 3.3



ภาพที่ 3.3 แสดงรายละเอียดของ ตาราง MS_MANAGEMENT, MS_MAPPING_OUT, MS_USER_MANAGEMENT, MS_USER_MANAGEMENT_OUT

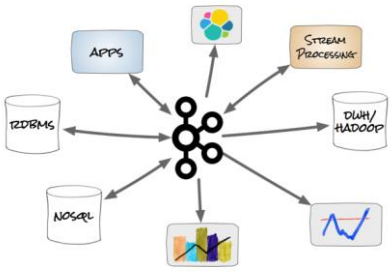
ข้อมูลส่วนบุคคล โดยจะมีข้อมูลที่เหมือนกันแสดงทั้ง 2 ตาราง โดยจะทำงานในลักษณะการเก็บข้อมูล input และ ข้อมูล output แสดงรายละเอียดดังตารางที่ 3.1

ตารางที่ 3.1 แสดงรายละเอียดของข้อมูลตาราง MS_USER_MANAGEMENT, MS_USER_MANAGEMENT_OUT

ลำดับ	รายการ	รายละเอียด
1	USER_ID	ใช้สำหรับเก็บข้อมูล ID ของผู้ใช้งานระบบ
2	USER_DESC	ใช้เก็บรายละเอียดข้อมูล ID ของผู้ใช้งานระบบ
3	EN_FIRSTNAME	ใช้เก็บชื่อจริงภาษาอังกฤษ
4	EN_LASTNAME	ใช้เก็บนามสกุลภาษาอังกฤษ
5	POSITION	ใช้เก็บข้อมูลตำแหน่งการทำงาน
6	EMAIL	ใช้เก็บข้อมูลจดหมายอิเล็กทรอนิกส์
7	TELEPHONE	ใช้เก็บข้อมูลโทรศัพท์
8	MOBILE	ใช้เก็บข้อมูลโทรศัพท์มือถือ
9	ACTIVE_FLAG	ใช้เก็บข้อมูลสถานะความสามารถใช้งานได้
10	LOGON_FLAG	ใช้เก็บข้อมูลสภานะการเข้าใช้ระบบ
11	SENSITIVE_ID	ใช้เก็บข้อมูล ID ของตาราง SENSITIVE
12	OPTIONAL_ID	ใช้เก็บข้อมูล ID ของตาราง OPTIONAL
13	CARD_NO	ใช้เก็บข้อมูลเลขบัตร
14	GENDER	ใช้เก็บข้อมูลเพศ
15	AGE	ใช้เก็บข้อมูลอายุ
16	INCOME	ใช้เก็บข้อมูลเงินเดือน

การเลือกระบบเครือข่ายเพื่อศึกษาการทำงานของบล็อคเชนเทคโนโลยี

การเลือกพิจารณา ระบบโครงข่าย เพื่อการศึกษาแนวทางการส่งข้อมูล และพัฒนาโปรแกรม เพื่อให้เป็นไปตามแนวทางซึ่งระบบสามารถต่อยอดได้ถึงการติดต่อกับ บล็อคเชนเทคโนโลยี แต่เนื่องจากที่ได้ศึกษาค้นคว้าแล้วพบว่า การใช้งานกับ Apache Kafka ถูกออกแบบมาเพื่อใช้งานง่ายและสอดคล้องกว่า



ภาพที่ 3.4 แสดงข้อมูล Apache Kafka มีคุณสมบัติครอบคลุมระบบฐานข้อมูล

ข้อจำกัดในการเลือกเครือข่าย เนื่องด้วย Apache Kafka มีคุณสมบัติรองรับการสื่อสารระหว่าง ฐานข้อมูลต่างๆ รวมถึง บล็อคเชนเทคโนโลยี และหลากหลาย ผู้เขียนได้พิจารณาอย่างเหมาะสมแล้ว เลือก Apache Kafka เพื่อนำมาพัฒนาเครือข่าย เนื่องจากในอนาคต Apache Kafka สามารถพัฒนาต่อยอดร่วมกับบล็อคเชนเทคโนโลยี เช่น Hyperledger หรือ Ethereum รวมถึงการทำงาน NoSQL และ RDBMS ซึ่งเป็นระบบดังเดิมซึ่ง จะเห็นได้ว่า Apache Kafka ครอบคลุมการทำงานสามารถติดต่อได้เกือบทุกค่าย แสดงดังภาพที่ รวมถึง ความคุ้มค่าในการศึกษาเพื่อต่อยอดการทำงานต่อไป

การทดลองและผลการวิจัย

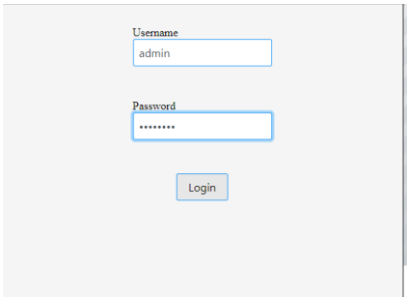
จากการออกแบบระบบจากบทที่แล้ว ในบทนี้ผู้เขียนจึงได้ Web Application เพื่อแสดงผลการทดลองและผลการวิจัย โดยมีขั้นตอนการใช้งาน รวมถึงผลการใช้งานในแต่ละส่วน

ผลการออกแบบ

ผลการทดลองและผลการวิจัยของการปกปิดข้อมูลส่วนบุคคลโดยเริ่มจากการกำหนดตำแหน่งและสิทธิ์เข้าถึงของแต่ละหน้ามีรายละเอียด ดังตารางที่ 4.1

ตารางที่ 4.1 แสดงสิทธิ์การเข้าถึงโปรแกรมการทำงานของแต่ละตำแหน่ง

Role/Page	Admin Control	Add Data Management	Structure Test	Data Production
Supervisor	/	/	/	/
Developer / Tester			/	/
User				/



ภาพที่ 4.1 แสดงหน้าจอ Login หน้าแรกของโปรแกรม

เริ่มต้นจากหน้า Login ดังภาพที่ 4.1 แสดงหน้า Login ทุก User สามารถใช้งานได้ที่การกรอก Username / Password เพื่อเข้าหน้าจอแรก โดยเริ่มต้นหน้าจอ Login จะแบ่งตาม Role หน้าที่ของแต่ละ ตำแหน่ง เมื่อใช้ Username: admin / Password: password แล้ว หน้าแรกของ Username: admin คือมี Role เป็น Supervisor ดังนั้นหน้าแรกที่ Login เข้ามาคือหน้า Admin Control

ภาพที่ 4.2 แสดงหน้าจอ Admin Control

รายการปุ่ม	**รายละเอียดและคำอธิบาย**
search	ค้นหา ข้อมูลการ Configure ตาม parameter ที่ต้องการค้นหาได้
add +	สามารถ เพิ่ม ข้อมูลที่ต้องการ Configure ได้
clear ×	สามารถ เคลียร์ข้อมูลหน้าจอค้นหาและ กลับสู่หน้าจอเริ่มต้นได้
generate	สร้างข้อมูลโดยใช้ หลักการ ตามที่กำหนดจากการ Configure ได้

send data	สามารถ ทดสอบส่งข้อมูลเข้าระบบเครือข่ายได้
delete	สามารถ ลบข้อมูล ในตารางได้

ภาพที่ 4.3 แสดงปุ่มบนหน้าจอ Admin Control ของระบบ

จากตารางที่ 4.1 ระดับ Supervisor จะมีสิทธิ์เข้าได้ถึง 4 เมนูด้วยกันได้แก่ ประกอบด้วย Admin Control , Add Data Management , Data Production ,Structure Test เริ่มต้นการใช้งานหน้า Admin Control วัตถุประสงค์ของหน้านี้คือสามารถ Configure data เพื่อกำหนดค่าเริ่มต้นการปกปิดข้อมูล โดยสามารถเลือกชนิดของ Algorithm กำหนด version และ comment ตัวอย่าง ดังภาพที่ 4.4 เป็นการ Configure Data 3 Rows ชนิด Shuffle ดังนี้ MS_USER_MANAGEMENT.POSITION, MS_USER_MANAGEMENT.TELEPHONE, MS_USER_MANAGEMENT.EN_FIRSTNAME

ในการกำหนดให้ข้อมูลบนตาราง MS_USER_MANAGEMENT และ COLUMN POSITION, TELEPHONE, EN_FIRSTNAME ให้มีการทำการ Data Masking ชนิด Shuffle หากมีข้อมูลมาบันทึกที่ TABLE และ COLUMN ดังกล่าว Data จะถูกเปลี่ยนเป็น ชนิด Shuffle คือเป็นการสลับข้อมูลแบบสุ่ม

ภาพที่ 4.4 แสดงการทำงาน Shuffle Masking

ข้อมูลก่อนการทำ Shuffle จะมีลักษณะข้อมูลปกติ แสดงดังภาพที่ 4.5

ID	USER_ID	USER_DESC	EN_FIRSTNAME	EN_LASTNAME	POSITION	EMAIL	TELEPHONE	MOBILE	ACTIVE_FLAG
1	210XX	Developer	First	Sripratanarat	DEVELOPER	overtime@company.com	0233456788	0987654321	Y
3	210YY	System Analys	Risa	Stamp	SUPERVISER	risa.email@gmail.com	021234555585	0861234599	Y
4	210Y1	Technical lead	Michael	Wongsarod	SUPERVISER	ScoreBoard@apple.com	0276545555-7	0874565555	Y
5	210Y2	Developer2	Pitsanu	Nampratum	DEVELOPER	ScoreBoardFirst@apple.com	0276545558-10	0874565557	Y
6	210Y3	Tester1	Manut	Onsretep	DEVELOPER	testemail1@hotmail.com	0233454444	0233454444	Y
7	210ZZ	Developer3	Thitiwat	Wuthimapa	DEVELOPER	thitiwat.wuthima@outlook.com	063241xxxx	063241hyyy	Y

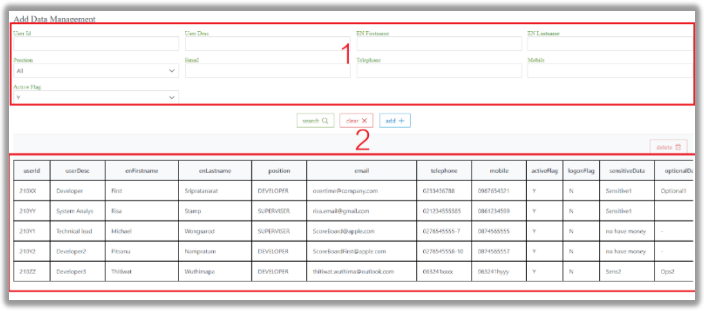
ภาพที่ 4.5 แสดงข้อมูลก่อนการทำ Shuffle Masking

ID	USER_ID	USER_DESC	EN_FIRSTNAME	EN_LASTNAME	POSITION	EMAIL	TELEPHONE	MOBILE	ACTIVE_FLAG
1	210XX	Developer	Michael	mgWG2SCONt/mGKqs9IUSs	SUPERVISER	XXXXXXX@company.com	021234555585	0987654321	Y
3	210YY	System Analys	Pitsanu	RP0efdeUKIg=	DEVELOPER	XXXXXXXXX@gmail.com	0233456788	0861234599	Y
4	210Y1	Technical lead	Thitiwat	xsQhvNeX0SpIdJpsrrJJvA==	DEVELOPER	XXXXXXXXXX@apple.com	063241xxxx	0874565555	Y
5	210Y2	Developer2	Risa	LC+vmHueJqAp8UfQ3ITXMQ==	DEVELOPER	XXXXXXXXXXXXXX@apple.com	0276545558-10	0874565557	Y
7	210ZZ	Developer3	First	Cj5mVg02jZT2dtdK7rjAFg==	SUPERVISER	XXXXXXXXXXXXXXX@outlook.com	0276545555-7	063241hyyy	Y

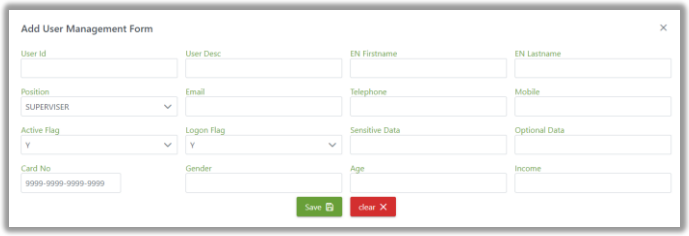
ภาพที่ 4.6 แสดงข้อมูลหลังการทำ Shuffle Masking

หลังจากกรอก data เรียบร้อยแล้วให้ทำการ กดปุ่ม generate เพื่อให้ระบบ generate output ได้ผลลัพธ์ คือข้อมูลหลังการทำ Shuffle จะมีลักษณะข้อมูลสุ่ม ดังภาพที่ 4.6 ซึ่งจะต่างกัน

เพราะสลับแบบสุ่มจากข้อมูลเดิม เปรียบเทียบภาพที่ 4.5 และจากได้ data output แล้วเราสามารถนำข้อมูลที่ได้ไปใช้พัฒนาต่อยอดเช่น นำไปเป็นข้อมูลทดสอบโดยเป็นข้อมูลเสมือนจริง ซึ่งจะทำให้ระบบทดสอบเพิ่มความน่าเชื่อถือได้มากขึ้น ส่วนการใช้งาน หน้า Add Data Management วัตถุประสงค์คือการเพิ่มข้อมูลเพื่อให้เป็นข้อมูลตั้งต้นก่อนการทำ Data Masking หากไม่กรอกหน้านี้จะไม่สามารถเริ่มการทำงานโปรแกรมได้อย่างถูกต้องเนื่องจากไม่มีข้อมูล



ภาพที่ 4.7 แสดงหน้าจอหลัก Add Data Management



ภาพที่ 4.8 แสดงหน้าจอ Pop-Up Add User Management Form

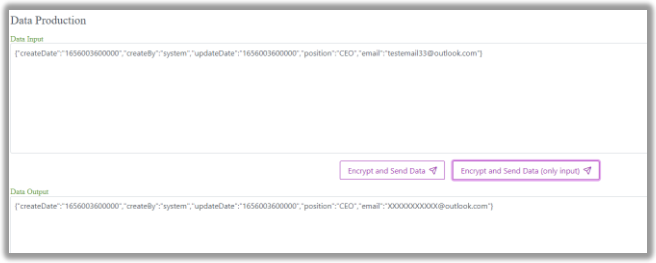
ตารางที่ 4.2 แสดงส่วนประกอบหน้า Add Data Management

รายการ	รายละเอียดและคำอธิบาย
ส่วนที่ 1 (Textbox)	สำหรับกรอกข้อมูลเพื่อเตรียมค้นหาต่อไป
ส่วนที่ 2 (Table+Columns)	Header กำหนดหัวข้อของข้อมูลรายการที่บันทึก Body ส่วนแสดงรายการข้อมูลตาม Header

การทำงานของปุ่มแต่ละปุ่ม หน้า Add Data Management ดังภาพที่ 4.7 ซึ่งความสามารถของปุ่มดังตารางที่ 4.3

ตารางที่ 4.3 แสดงการทำงานของปุ่ม หน้า Add Data Management

รายการปุ่ม	รายละเอียดและคำอธิบาย
search	ค้นหาข้อมูล ตาม parameter ที่ต้องการค้นหาได้
add +	สามารถ เปิดหน้า Add User Management Form เพื่อ บันทึกข้อมูลเพิ่มได้
clear	สามารถ เคลียร์ข้อมูลหน้าจอค้นหาและ กลับสู่หน้าจอเริ่มต้นได้
delete	สามารถ ลบข้อมูล ในตารางได้



ภาพที่ 4.8 แสดงหน้าจอ Data Production

สามารถนำ JSON String มาตรวจสอบที่โปรแกรม Data Production เพื่อ Format ที่ถูกต้องในการ เข้ารหัส รวมถึงสามารถเพิ่มเติม parameter ที่ขาดหากต้องการส่งข้อมูลที่ครบถ้วน ส่วนหน้า Structure Test คือหน้าจอสำหรับตรวจโครงสร้างของข้อมูลที่มีการ Configure จากหน้า Admin Control และสามารถ Convert Data จาก Encrypt เป็น Decrypt รวมถึงสามารถจัด Format JSON ได้

ภาพที่ 4.9 หน้าจอจัด Format JSON ใช้สำหรับทดสอบโครงสร้างของข้อมูล

การใช้งาน Algorithm Masking ข้อมูลทำกำหนดให้มี CARD_NO Type 16 Digit Masking, EMAIL Type Email Masking, EN_LASTNAME Type 3DES

ID	USER_ID	USER_DESC	EN_FIRSTNAME	EN_LASTNAME	POSITION	EMAIL	CARD_NO	TELEPHONE	MOBILE
1	210XX	Developer	First	Sripratanarat	DEVELOPER	overtime@company.com	1234-5677-8990-1234	0233456788	0987654321
3	210YY	System Analys	Risa	Stamp	SUPERVISER	risa.email@gmail.com	4444-3333-2222-2111	021234555585	0861234599
4	210Y1	Technical lead	Michael	Wongsarod	SUPERVISER	ScoreBoard@apple.com	4321-4533-3334-3333	0276545555-7	0874565555
5	210Y2	Developer2	Pitsanu	Nampratum	DEVELOPER	ScoreBoardFirst@apple.com	4321-4533-3334-3333	0276545558-10	0874565557
6	210Y3	Tester1	Manut	Onsretep	DEVELOPER	testemail1@hotmail.com	4321-4533-3334-3333	0233454444	0233454444
7	210ZZ	Developer3	Thitiwat	Wuthimapa	DEVELOPER	thitiwat.wuthima@outlook.com	5432-6789-2131-2213	063241xxxx	063241hyyy

ภาพที่ 4.10 แสดงก่อนการเข้ารหัสและการทำ Masking

ข้อมูลหลังการกดปุ่ม Generate 

ID	USER_ID	USER_DESC	EN_FIRSTNAME	EN_LASTNAME	POSITION	EMAIL	CARD_NO	TELEPHONE	MOBILE
1	210XX	Developer	Michael	mgWG2SCONt/mGKqs9IUsQ==	SUPERVISER	XXXXXXXX@company.com	****-****-****-1234	021234555585	0987654321
3	210YY	System Analys	Pitsanu	RP0efdeUKIg=	DEVELOPER	XXXXXXXXXX@gmail.com	****-****-****-2111	0233456788	0861234599
4	210Y1	Technical lead	Thitiwat	xsQhvNeX05pIdJpsrrJJvA==	DEVELOPER	XXXXXXXXXX@apple.com	****-****-****-3333	063241xxxx	0874565555
5	210Y2	Developer2	Risa	LC+vmHueJqAp8UfQ3ITXMQ==	DEVELOPER	XXXXXXXXXXXXXX@apple.com	****-****-****-3333	0276545558-10	0874565557
7	210ZZ	Developer3	First	Cj5mVg02jZT2dtdK7rjAFg==	SUPERVISER	XXXXXXXXXXXXXXX@outlook.com	****-****-****-2213	0276545555-7	063241hyyy

ภาพที่ 4.11 แสดงหลังการเข้ารหัสและการทำ Masking

ผลการทดลอง คือ EN_LASTNAME ถูกเข้ารหัส ด้วย 3DES ,EMAIL ถูกทำการ MARKER เรียบร้อย ,CARD_NO ถูกทำการปกปิดข้อมูลเรียบร้อย

ระดับ Developer / Tester สามารถ ตรวจสอบ Structure ด้วย JSON format และจัดเรียงความสวยงามด้วยรวมสามารถ Decrypt ได้บางชนิดการเข้ารหัส ดังภาพที่ 4.9

ระดับ User สามารถเข้าโปรแกรมได้เมื่อต้องการทดสอบข้อมูลบนระบบว่า ทำงานถูกต้องตรงไปตรงมาหรือไม่ หรือในที่นี้ ค่าที่ส่งเข้าไป คือ {enFirstname :"KING" , "enLastname":Test} จะสามารถเข้ารหัสได้ในทันที ทำงานเป็น Proxy Convert Data จาก input to Encrypt, Anonymized Pseudonyms Data ได้ หน้าจอจะเป็นดังภาพที่ 4.8

รายละเอียดการวิจัย

ตารางที่ 4.4 รายละเอียดการทดสอบจริงจาก developer 3 คน และนักศึกษาฝึกงาน 2 คน

ตัวชี้วัด	ก่อนการวิจัย	ผลลัพธ์	หมายเหตุ
1. เวลาที่ใช้ในการทำความเข้าใจความหมายไม่ตรงกันของพนักงานกับลูกค้า คุยกันถกเถียงเรื่อง parameter API	ใช้เอกสาร และประชุม ใช้เวลา 1.0 mandays (ประชุมเช้า – บ่าย)	ใช้ program ทดสอบ ใช้เวลา 0.25 mandays (ไม่ถึง 2 hr.)	ลดลง 75%
2. เวลาที่ใช้ในการสื่อสารภายในทีม (Developer กับ System Analyst)	ใช้เอกสาร และประชุมคุยกัน ใช้เวลา 0.5 mandays (ประชุมเช้า)	ใช้ program ทดสอบ ใช้เวลา 0.125 mandays (ไม่ถึง 1 hr)	ลดลง 75%
3. เวลาที่ใช้ในการสื่อสารภายในทีม (Tester กับ Developer)	ใช้เอกสาร และประชุมคุยกัน ใช้เวลา 0.5 mandays (ประชุมเช้า)	ใช้ program ทดสอบ ใช้เวลา 0.125 mandays (ไม่ถึง 1 hr)	ลดลง 75%
4. ความเข้าใจของนักศึกษาฝึกงาน รวมถึง Developer ที่เข้ามาใหม่	ใช้เอกสาร และประชุมเรียนรู้งาน ใช้เวลา 25 mandays	ใช้ program สอนและอธิบาย ใช้เวลา 7 mandays	ทดสอบโดยให้ Developer เข้ารหัสชนิดต่างๆ
5.การใช้งาน Encrypt/Decrypt online	100%	20%	ทดสอบ 5 คนเหลือใช้ Web online 1 คน

สรุปผลการศึกษาและวิจัย

จากการวิจัยข้างต้น สรุปข้อมูลการใช้งานเว็บแอปพลิเคชัน การปกป้องข้อมูลส่วนบุคคลด้วยเทคนิคข้อมูลแฝงและ ข้อมูลนิรนามผ่านบริการพร็อกซี่บนระบบเครือข่าย

ผลที่เกิดขึ้นหลังงานวิจัย

ลดเวลาที่ใช้ในการทำความเข้าใจ ความหมายไม่ตรงกันของพนักงานกับลูกค้า คุยถกเถียงเรื่อง parameter API, การสื่อสารภายในทีม (Developer กับ System Analyst) ,การสื่อสารภายในทีม (Tester กับ

Developer) ,รวมถึงสามารถสอนนักศึกษาฝึกงาน รวมถึง Developer เข้ามาใหม่ มีความเข้าใจเกี่ยวกับเรื่องการเข้ารหัสถอดรหัสมากขึ้น และลดการใช้งาน Encrypt/Decrypt จาก website online

ข้อเสนอแนะจากการวิจัย

สามารถพัฒนาต่อยอดการทำงานของระบบ เว็บแอปพลิเคชันให้มีประสิทธิภาพมากขึ้น เช่น ทำระบบ log เพื่อประมวลผลและแสดงรายงาน เพื่อทำบทสรุปและการวิเคราะห์ผลของเว็บแอปพลิเคชันได้โดยใช้ข้อมูลที่มีอยู่ในระบบ และสามารถออกรายงานได้ รวมถึง เพิ่มประสิทธิภาพในการนำเข้าข้อมูล การจัดเก็บข้อมูล การประมวลผล การรายงานผล รูปแบบการแสดงผล การเชื่อมโยงกับฐานข้อมูลที่มีอยู่ มีการทำงานของระบบ และสามารถ เพิ่มระบบทดสอบระบบรักษาความปลอดภัยของฐานข้อมูลและอื่น ๆ

บรรณานุกรม

ศูนย์วิจัยกฎหมายและการพัฒนา จุฬาลงกรณ์มหาวิทยาลัย “แนวปฏิบัติเกี่ยวกับการคุ้มครองข้อมูลส่วนบุคคล”,2561 https://www.law.chula.ac.th/wp-content/uploads/2019/06/tdpg.pdf

พระราชบัญญัติคุ้มครองข้อมูลส่วนบุคคล พ.ศ. 2562 (2562). ราชกิจจานุเบกษา. เล่ม 136 ตอนที่ 69 ก. หน้า 52-94.

Baeldung. (2022). Java Base64 Encoding and Decoding. Retrieved from https://www.baeldung.com/java-base64-encode-and-decode

Boris Terzic. (2008). How do I use 3DES encryption/decryption in Java. Retrieved from https://stackoverflow.com/questions/20227/how-do-i-use-3des-encryption-decryption-in-java

Cédric Burton and Sára Hoffman. (2015). Personal Data, Anonymization, and Pseudonymization in the EU. Retrieved from https://www.wsgrdataadvisor.com/2015/09/personal-data-anonymization-and-pseudonymization-in-the-eu/

Dhiraj Rai. (2018). RSA Encryption and Decryption in Java. Retrieved from https://www.devglan.com/java8/rsa-encryption-decryption-java

Geeksforgeeks.org. (2022). SHA-256 Hash in Java. Retrieved from https://www.geeksforgeeks.org/sha-256-hash-in-java/

Koba Molenaar. (2022). Top 12 Proxy Websites for Anonymous Browsing in 2022. Retrieved from https://influencermarketinghub.com/proxy-websites/#toc-1Krzysztof Majewski “3DES in Java”, https://www.baeldung.com/java-3des

Shanika Wickramasinghe. (2021). What’s Data Masking? Types, Techniques & Best Practices. Retrieved from https://www.bmc.com/blogs/data-masking