คำสั่งงานโครงงานรายวิชา Data Encryption

ชื่อโครงงาน โครงงานการออกแบบและประยุกต์ใช้การเข้ารหัสข้อมูลในระบบสารสนเทศ

คำอธิบายรายวิชา (ส่วนของโครงงาน)

โครงงานนี้มีวัตถุประสงค์เพื่อให้นิสิตเข้าใจหลักการของการเข้ารหัสข้อมูล (Data Encryption) ในเชิงลึก และสามารถนำไปประยุกต์ใช้ในการออกแบบระบบจริงที่เกี่ยวข้องกับความมั่นคงปลอดภัยของข้อมูล การสื่อสาร หรือการยืนยันตัวตน

คำสั่งงาน

ให้นิสิตทำงานเป็นกลุ่ม (กลุ่มละ <=6 คน) โดยออกแบบระบบที่ใช้กลไกการเข้ารหัสข้อมูลเพื่อแก้ปัญหาด้านความมั่นคงปลอดภัยในสถานการณ์จริง

นิสิตเป็นผู้กำหนด

- บริบทของระบบ
- ปัญหาที่ต้องการแก้ไข
- รูปแบบการใช้งาน
- เทคโนโลยีและกลไกการเข้ารหัสที่เหมาะสม

ตัวอย่างบริบท (เพื่อเป็นแนวคิดเท่านั้น)

- การสื่อสารของ Mobile Application หรือ Web Application
- ระบบยืนยันตัวตนของผู้ใช้หรืออุปกรณ์
- การยืนยันตัวตนของอุปกรณ์ IoT
- Digital Watermarking
- Secure Data Sharing
- อื่น ๆ ที่เกี่ยวข้องกับ Data Encryption

หมายเหตุ: ห้ามคัดลอกตัวอย่างมาใช้โดยตรง นิสิตต้องปรับบริบทหรือออกแบบระบบใหม่ด้วยตนเอง

ขอบเขตของโครงงาน

โครงงานต้องประกอบด้วย

- การวิเคราะห์ปัญหาและภัยคุกคาม (Threat Analysis)
- การออกแบบสถาปัตยกรรมของระบบ
- การเลือกและอธิบายกลไกการเข้ารหัสข้อมูล
- การจัดการกุญแจ (Key Management)
- การวิเคราะห์ข้อจำกัดและความเสี่ยงของระบบ
- การสาธิตหรือสร้างต้นแบบ (Prototype / Simulation)

สิ่งที่ต้องส่ง

- Proposal
- System Design Document
- Prototype / Demo (ถ้ามี)
- รายงานฉบับสมบูรณ์
- การนำเสนอผลงาน

เกณฑ์การประเมิน (โดยสังเขป)

- ความชัดเจนของปัญหาและการวิเคราะห์ภัยคุกคาม
- ความเหมาะสมของกลไกการเข้ารหัสที่เลือก
- ความเข้าใจด้านการจัดการกุญแจและความเชื่อถือ (Trust)
- ความสมเหตุสมผลของการออกแบบระบบ
- การตระหนักถึงข้อจำกัดและความเสี่ยง
- ความสามารถในการอธิบายแนวคิดทางวิชาการ

ข้อเสนอโครงงานรายวิชา 02204352 ประจำ ภาคปลาย ปีการศึกษา 2568

1. ชื่อโครงงาน (ภาษาไทย และภาษาอังกฤษ ถ้ามี)

การพัฒนาระบบรับส่งไฟล์เสริมความปลอดภัยภายในเครือข่ายท้องถิ่น

2. ที่มาและความสำคัญของปัญหา

ปัญหาที่เกิดขึ้นในสถานการณ์จริง

ในปัจจุบันมีการใช้เครือข่ายอินเตอร์เน็ตในการส่งไฟล์ต่าง ๆ เพิ่มมากขึ้นเป็นช่องทางปกติ และมีความสะดวกกว่าการส่งข้อมูลแบบดั้งเดิมซึ่งต้องใช้ฮาร์ดแวร์เสริมอยู่มาก แต่การส่งข้อมูลผ่านเครือข่ายเหล่านี้มีความเสี่ยงที่จะถูกผู้ไม่ประสงค์ดีดักจับข้อมูล และแอบแก้ไขข้อมูลระหว่างทาง (Man-in-the-Middle attack) ได้

เหตุผลที่จำเป็นต้องใช้การเข้ารหัสข้อมูล

เพื่อสร้างช่องทางการสื่อสารที่ปลอดภัยทำให้มั่นใจได้ว่ามีเพียงผู้รับที่ได้รับอนุญาตเท่านั้นที่สามารถเปิดอ่านไฟล์ได้ และสร้างความปลอดภัยในการส่งข้อมูลโดยจะไม่ถูกดักข้อมูลกลางทางโดย Man-in-the-Middle

ผลกระทบหากไม่มีการป้องกันข้อมูล

หากปราศจากการเข้ารหัส ข้อมูลจะมีความเสี่ยงสูงต่อการถูกดักฟังและดัดแปลง (MITM) ส่งผลกระทบโดยตรงต่อความลับและความถูกต้องของข้อมูลรวมถึงเปิดช่องโหว่ให้อุปกรณ์ภายในเครือข่ายถูกโจมตีต่อเนื่องได

3. วัตถุประสงค์ของโครงงาน

- เพื่อออกแบบและพัฒนาระบบรับส่งไฟล์แบบ Peer-to-Peer ที่มีการป้องกันการดักจับข้อมูลเชิงรุก
- เพื่อความรู้ และนำกลไกการเข้ารหัสข้อมูลแบบมาประยุกต์ใช้ในการแก้ปัญหาความปลอดภัยในชีวิต
- เพื่อป้องกันความเสี่ยงจากการดักฟัง การแก้ไขข้อมูล และการโจมตีแบบ Man-in-the-Middle ในระหว่างการรับส่งไฟล์
- เพื่อเพิ่มความมั่นใจว่าข้อมูลที่รับส่งภายในเครือข่ายสามารถเข้าถึงได้เฉพาะผู้ใช้ที่ได้รับอนุญาต และรักษาความลับ ความถูกต้อง และความน่าเชื่อถือของข้อมูล

4. ขอบเขตของระบบ

ผู้ใช้งานหรืออุปกรณ์ที่เกี่ยวข้อง

1. ขอบเขตของระบบ (System Scope)
    - การค้นหา: ค้นหาอุปกรณ์ในวงแลน (LAN) เดียวกันโดยอัตโนมัติผ่านโพรโทคอล UDP/mDNS
    - ความปลอดภัย: เข้ารหัสข้อมูลแบบต้นทางถึงปลายทาง (End-to-End Encryption) ด้วย AES-256
    - การเชื่อมต่อ: รับส่งข้อมูลแบบจุดต่อจุด (P2P) ผ่าน TLS/HTTPS โดยมีตัวกลางเป็นเครือข่ายของผู้ใช้
    - การจัดการไฟล์: รองรับการรับส่งไฟล์ทุกประเภท (รูปภาพ, เอกสาร, วิดีโอ) และแสดงสถานะการส่ง (Progress Bar)
2. ผู้ใช้งานและอุปกรณ์ที่เกี่ยวข้อง (Users & Devices)
    - ผู้ใช้งาน: บุคคลทั่วไปที่เชื่อมต่ออยู่ในเครือข่ายท้องถิ่นเดียวกัน และต้องการแลกเปลี่ยนไฟล์อย่างปลอดภัย
    - อุปกรณ์ (Endpoints): คอมพิวเตอร์ (Linux, Windows, macOS) และสมาร์ทโฟน (Android, iOS) ที่รองรับ Web Browser
    - โครงสร้างเครือข่าย: อุปกรณ์กระจายสัญญาณ (Router/Switch) ที่รองรับมาตรฐาน TCP/IP และอยู่ใน Subnet เดียวกัน

ข้อมูลที่ต้องได้รับการป้องกัน

ข้อมูลที่ต้องได้รับการป้องกันครอบคลุมตั้งแต่ เนื้อหาไฟล์ต้นฉบับ, ข้อมูลประกอบไฟล์ (Metadata) เช่น ชื่อและขนาดไฟล์ ไปจนถึง ข้อมูลระบุตัวตนของอุปกรณ์ เพื่อให้มั่นใจว่าตลอดกระบวนการรับส่งจะไม่มีข้อมูลส่วนบุคคลหรือความลับของไฟล์รั่วไหลออกไปสู่บุคคลที่สามในเครือข่าย

สิ่งที่โครงงานครอบคลุม และไม่ครอบคลุม

- พัฒนาโปรแกรมสำหรับรับ–ส่งไฟล์ภายในเครือข่ายท้องถิ่น (LAN) ระหว่างอุปกรณ์ที่อยู่ในเครือข่ายเดียวกัน
- รองรับการค้นหาอุปกรณ์ปลายทางโดยอัตโนมัติผ่านโพรโทคอล UDP/mDNS และแสดงรายการอุปกรณ์ที่พร้อมเชื่อมต่อ
- เข้ารหัสข้อมูลแบบต้นทางถึงปลายทาง (End-to-End Encryption) ด้วยอัลกอริทึม AES-256 และสื่อสารผ่าน TLS/HTTPS เพื่อป้องกันการดักฟังและการโจมตีแบบ Man-in-the-Middle (MITM)
- รับส่งไฟล์ได้ทุกประเภท พร้อมแสดงสถานะความคืบหน้า (Progress Bar) และผลลัพธ์การส่งไฟล์ (สำเร็จ/ล้มเหลว)
- ทดสอบและประเมินประสิทธิภาพของระบบในด้านความเร็ว ความถูกต้อง และความปลอดภัยในการใช้งานจริง

สิ่งที่โครงงานไม่ครอบคลุม

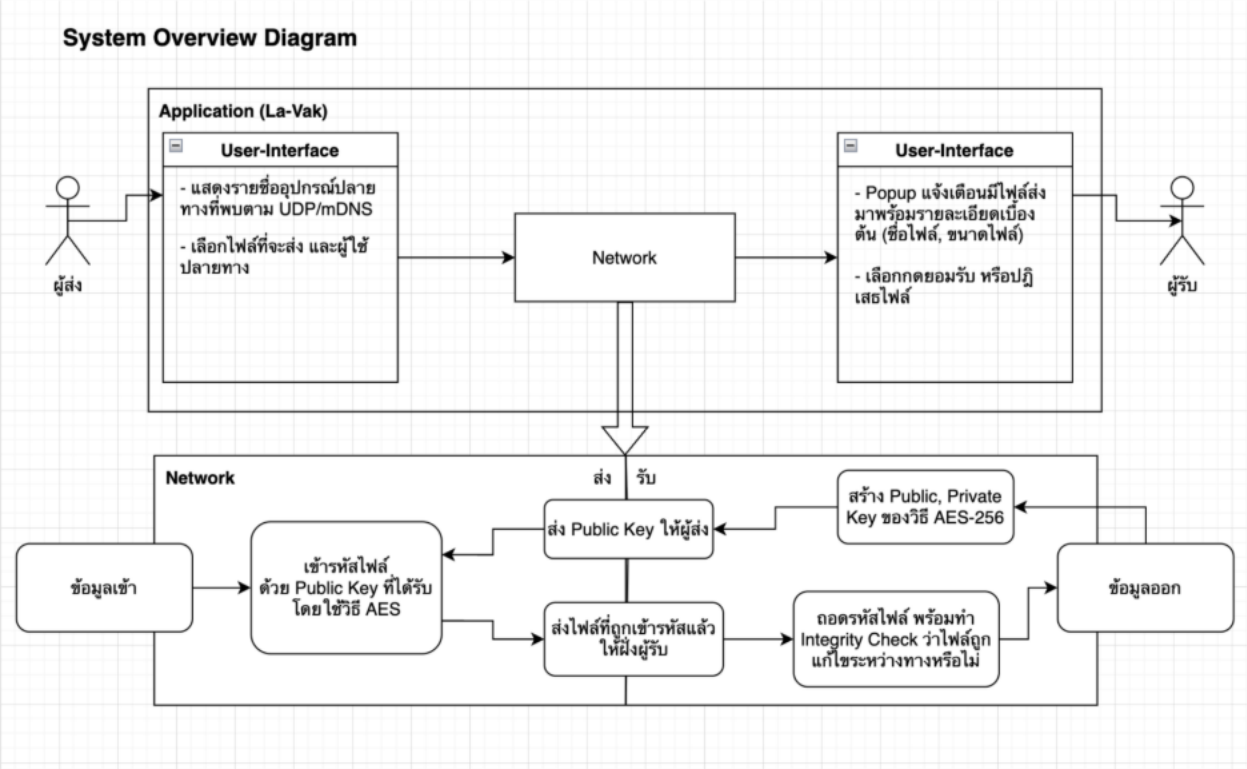
- การรับส่งไฟล์ผ่านเครือข่ายอินเทอร์เน็ตสาธารณะ (WAN/Cloud) หรือการเชื่อมต่อข้าม Subnet
- การจัดเก็บไฟล์ถาวรบนเซิร์ฟเวอร์กลาง หรือระบบคลาวด์สตอเรจ เช่น Google Drive หรือ Dropbox
- ระบบจัดการผู้ใช้ขั้นสูง เช่น ระบบสมาชิก ฐานข้อมูลผู้ใช้ หรือการกำหนดสิทธิ์แบบหลายระดับ (Role-based Access Control ขนาดใหญ่)
- การป้องกันภัยคุกคามจากมัลแวร์ ไวรัส หรือการสแกนไฟล์ด้านความปลอดภัยเชิงลึก (Antivirus/Threat Detection)

5. แนวคิดและภาพรวมของระบบ

อธิบายการทำงานของระบบโดยสังเขป

ระบบจะทำงานในรูปแบบ Peer-to-Peer (P2P) โดยเริ่มจากการค้นหาอุปกรณ์ในเครือข่ายท้องถิ่นผ่าน UDP/mDNS เมื่อผู้ส่งเลือกไฟล์และเลือกผู้รับ ระบบจะทำการแจ้งเตือนไปยังผู้รับ (Push Notification/Alert) เพื่อขอความยินยอม เมื่อผู้รับกดตกลง ระบบจะสร้างช่องทางสื่อสารที่เข้ารหัส (Encrypted Tunnel) เพื่อรับส่งไฟล์ข้อมูล โดยมีกระบวนการตรวจสอบความถูกต้องของไฟล์ที่ปลายทางเพื่อให้มั่นใจว่าข้อมูลไม่ถูกแก้ไขระหว่างการรับส่ง

แสดงแผนภาพสถาปัตยกรรม (System Overview Diagram)



6. แนวทางด้านความมั่นคงปลอดภัยของข้อมูล

รูปแบบการเข้ารหัสที่คาดว่าจะใช้ (เช่น Symmetric, Asymmetric, Hash, Signature ฯลฯ)

- รูปแบบที่เลือก: การเข้ารหัสแบบสมมาตร (Symmetric Encryption) เทคโนโลยีหลัก: AES-256 (Advanced Encryption Standard 256-bit)

เหตุผลในการเลือก

- ความมั่นคงปลอดภัย: มีความยาวกุญแจถึง 256 บิต ซึ่งถือว่าเป็นระดับความปลอดภัยสูงสุดที่ปัจจุบันยังไม่มีคอมพิวเตอร์เครื่องใดในโลกสามารถเจาะระบบด้วยการสุ่มรหัส (Brute-force) ได้สำเร็จ
- ประสิทธิภาพสูง: AES ถูกออกแบบมาให้ทำงานกับข้อมูลที่เป็นบล็อก (128 บิตต่อบล็อก) ทำให้ประมวลผลได้เร็วมาก เหมาะกับการส่งไฟล์ผ่านสายแลนหรือ Wi-Fi

สมมติฐานด้านความปลอดภัย (Security Assumptions)

โครงงานนี้ตั้งอยู่บนสมมติฐานด้านความมั่นคงปลอดภัยของข้อมูลดังต่อไปนี้

- อุปกรณ์ของผู้ใช้งานทั้งฝั่งผู้ส่งและผู้รับอยู่ภายใต้การควบคุมของผู้ใช้ที่เชื่อถือได้ และไม่มีมัลแวร์หรือซอฟต์แวร์อันตรายที่สามารถเข้าถึงข้อมูลหรือกุญแจเข้ารหัสภายในระบบได้
- กุญแจเข้ารหัสที่ใช้ในระบบถูกสร้างและจัดเก็บอย่างปลอดภัยภายในอุปกรณ์ของผู้ใช้งาน และไม่ถูกเปิดเผยให้กับบุคคลที่สามโดยเจตนาหรือโดยไม่เจตนา
- ผู้โจมตีสามารถเข้าถึงเครือข่ายท้องถิ่นและดักฟังการรับส่งข้อมูลได้ แต่ไม่สามารถถอดรหัสข้อมูลได้หากไม่มีกุญแจลับที่ถูกต้อง
- ผู้ใช้งานมีการยืนยันอุปกรณ์ปลายทางก่อนการรับส่งไฟล์ และไม่มีการเชื่อมต่อกับอุปกรณ์ที่ไม่ทราบแหล่งที่มาหรือไม่ได้รับความไว้วางใจ

7. ภัยคุกคามที่คาดว่าจะเกิดขึ้น (Threats) ระบุอย่างน้อย 3 ภัยคุกคาม พร้อมคำอธิบายสั้น ๆ

- การโจมตีแบบ Man-in-the-Middle (MITM)
  ผู้ไม่ประสงค์ดีแทรกตัวอยู่ระหว่างผู้ส่งและผู้รับเพื่อดักฟัง แก้ไข หรือปลอมแปลงข้อมูลระหว่างการรับส่งไฟล์ ทำให้ข้อมูลสูญเสียความลับและความถูกต้อง
- การดักฟังข้อมูลบนเครือข่าย (Packet Sniffing / Eavesdropping)
  ผู้โจมตีใช้เครื่องมือดักจับแพ็กเก็ตข้อมูลในเครือข่าย LAN เพื่ออ่านข้อมูลที่รับส่ง หากไม่มีการเข้ารหัสข้อมูลสำคัญหรือไฟล์ส่วนตัวอาจรั่วไหลได้
- การปลอมแปลงตัวตนอุปกรณ์ (Device Spoofing / Impersonation)
  ผู้โจมตีปลอมตัวเป็นอุปกรณ์หรือผู้ใช้งานที่ถูกต้อง เพื่อหลอกให้ผู้ใช้ส่งไฟล์ไปยังปลายทางที่ไม่ปลอดภัย ส่งผลให้ข้อมูลถูกขโมยหรือเข้าถึงโดยไม่ได้รับอนุญาต

8. แผนการดำเนินงานโดยสังเขป

ตารางหรือรายการขั้นตอน

ลำดับ	ขั้นตอน	ระยะเวลาในการทำโครงงาน			
		Week 1	Week 2	Week 3	Week 4
1	เสนอหัวข้อ				
2	จัดทำข้อเสนอโครงงานรายวิชา				
3	ออกแบบ UI/UX และโครงสร้างโปรแกรม				
4	พัฒนาระบบค้นหาอุปกรณ์และรับ–ส่งไฟล์ (UDP/mDNS + P2P)				
5	พัฒนาระบบเข้ารหัสข้อมูล (AES-256 + TLS/HTTPS)				
6	ทดสอบระบบ				
7	ส่ง prototype				

System Design Document

1. ภาพรวมของระบบ (System Overview)

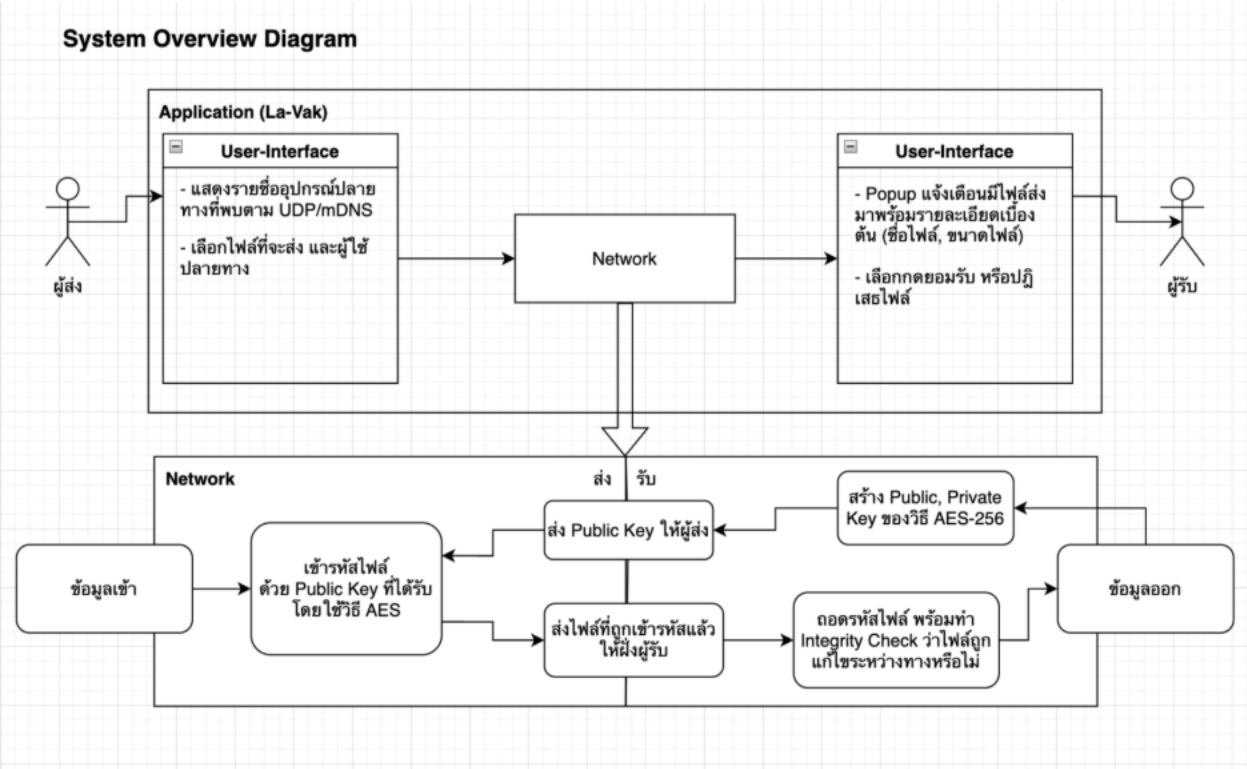
โครงงานนี้เป็นการพัฒนาระบบรับส่งไฟล์ที่มีความปลอดภัยสูงภายในเครือข่ายท้องถิ่น (LAN) โดยอาศัยสถาปัตยกรรมแบบ จุดต่อจุด (Peer-to-Peer Architecture) ซึ่งช่วยให้อุปกรณ์สองเครื่องสามารถสื่อสารและแลกเปลี่ยนข้อมูลกันได้โดยตรง ไม่ต้องผ่านเซิร์ฟเวอร์กลาง (Centralized Server) เพื่อลดความหน่วงและเพิ่มความเป็นส่วนตัวของข้อมูล

อธิบายภาพรวมของระบบและวัตถุประสงค์ด้านความมั่นคงปลอดภัย

ระบบถูกแบ่งออกเป็น 3 ส่วนหลักที่ทำงานประสานกัน ดังนี้:

- ส่วนการค้นหาอุปกรณ์ (Discovery Module): ทำหน้าที่ตรวจสอบและแสดงรายชื่ออุปกรณ์ที่ออนไลน์อยู่ในวงแลนเดียวกัน โดยใช้โพรโทคอล UDP Multicast เพื่อให้ผู้ใช้สามารถเลือกโหนดปลายทางได้อย่างรวดเร็วโดยไม่ต้องระบุหมายเลขไอพี (IP Address) ด้วยตนเอง
- ส่วนการรักษาความปลอดภัย (Security Module): เป็นหัวใจสำคัญของโครงงาน โดยใช้มาตรฐานการเข้ารหัสแบบสมมาตร AES-256 (Advanced Encryption Standard) ในการแปลงไฟล์ต้นฉบับ (Plaintext) ให้เป็นข้อมูลเข้ารหัส (Ciphertext) ก่อนส่งออกไป และทำหน้าที่ถอดรหัสที่ฝั่งผู้รับ
- ส่วนการรับส่งข้อมูล (Transmission Module): ทำหน้าที่จัดการการเชื่อมต่อผ่าน TCP Sockets เพื่อสร้างช่องทางในการส่ง Stream ข้อมูลระหว่างอุปกรณ์ โดยมีการจัดการระบบคิว (Queue) และการแสดงสถานะการรับส่ง (Progress Status) ให้ผู้ใช้งานทราบ

2. สถาปัตยกรรมของระบบ (System Architecture)

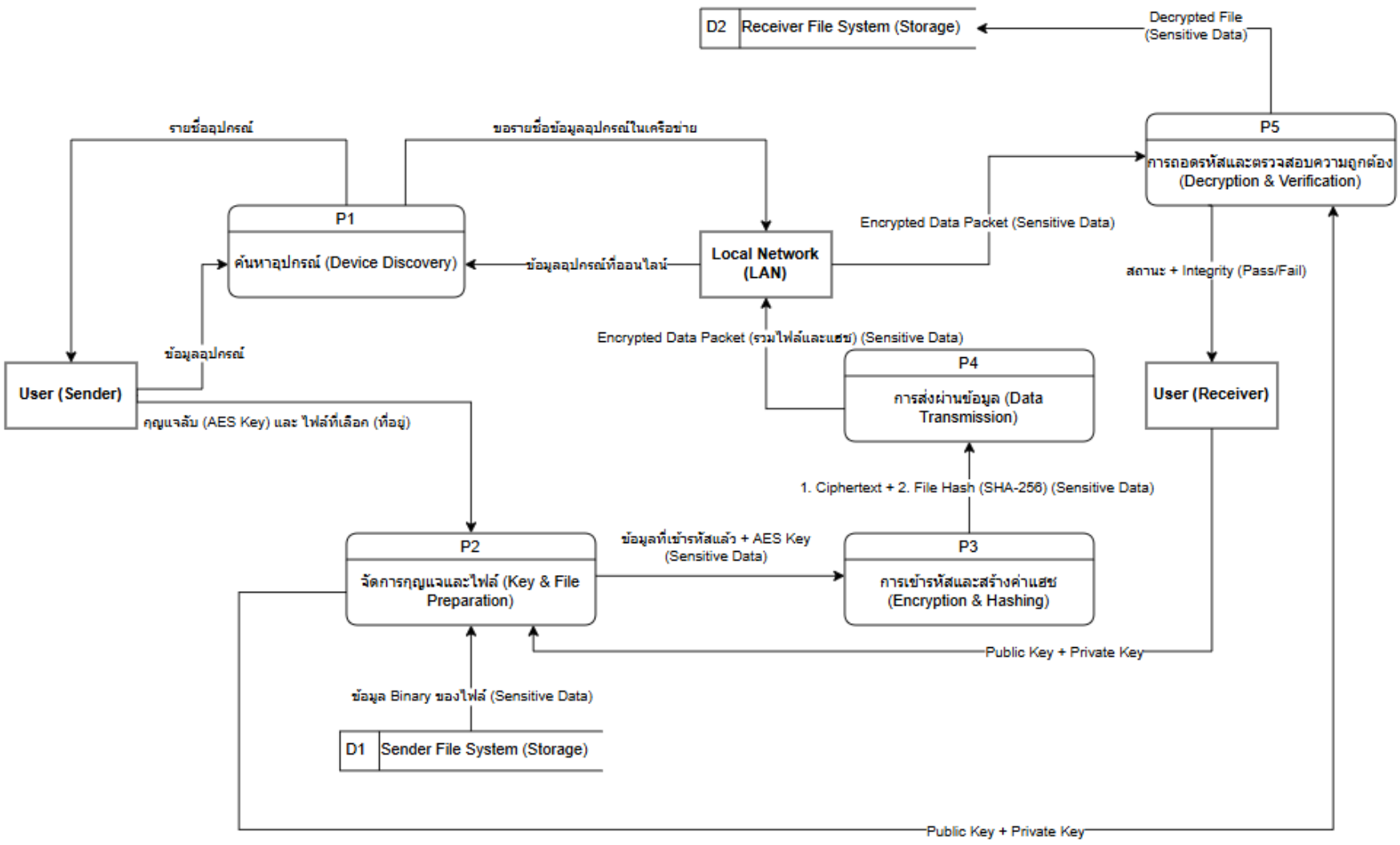


- User (Sender): ผู้ส่งที่เลือกไฟล์และระบุกุญแจลับ
- User (Receiver): ผู้รับที่รอกดรับไฟล์และใส่กุญแจเพื่อถอดรหัส
- Local Network (LAN): ตัวกลางที่ข้อมูล (ที่เข้ารหัสแล้ว) วิ่งผ่าน

Processes

- P1: ค้นหาอุปกรณ์ (Device Discovery): รับข้อมูลจาก LAN เพื่อแสดงรายชื่อเครื่องที่ออนไลน์อยู่
- P2: จัดการกุญแจและไฟล์ (Key & File Preparation): รับ "กุญแจลับ" และ "ไฟล์ต้นฉบับ" จาก User เพื่อเตรียมเข้าสู่ระบบ
- P3: การเข้ารหัสข้อมูล (Encryption Process): นำไฟล์มาประมวลผลด้วยอัลกอริทึม AES-256 จนกลายเป็น Ciphertext
- P4: การส่งผ่านข้อมูล (Data Transmission): ส่งไฟล์ที่ล็อกแล้วผ่าน Socket ไปยัง LAN
- P5: การรับและถอดรหัสข้อมูล (Reception & Decryption): ฝั่งผู้รับรับข้อมูลมาจาก LAN และใช้กุญแจลับในการถอดรหัสคืนค่าเป็นไฟล์เดิม

3. การไหลของข้อมูล (Data Flow)



4. กลไกการเข้ารหัสข้อมูล

อัลกอริทึมที่ใช้

ระบบเลือกใช้ AES (Advanced Encryption Standard) แบบสมมาตร

รูปแบบการใช้งาน (Mode, Key Size)

- Key Size: 256 บิต - ให้ระดับความปลอดภัยสูงมาก และทนทานต่อการโจมตีแบบ Brute-force ในทางปฏิบัติ
- Mode of Operation - ใช้ AES-256-GCM (Galois/Counter Mode)
- Authentication Tag - ใช้ Authentication Tag ที่มากับ GCM เพื่อตรวจสอบความถูกต้องของข้อมูลเมื่อถึงปลายทาง

5. การจัดการกุญแจ (Key Management)

วิธีการสร้างกุญแจ (Key Generation)

การสร้างกุญแจต้องเน้นที่ความสุ่มแบบคาดเดาไม่ได้ (Randomness) เพื่อป้องกันการโจมตีเชิงสถิติ:

- อัลกอริทึม: ใช้เครื่องมือสร้างตัวเลขสุ่มที่มีความปลอดภัยทางรหัสผ่าน

  (CSPRNG - Cryptographically Secure Pseudo-Random Number Generator)
- ขนาดกุญแจ: สร้างข้อมูลสุ่มขนาด 256 บิต (32 ไบต์) เพื่อให้สอดคล้องกับมาตรฐาน AES-256
- แหล่งที่มาของความสุ่ม: Library มาตรฐานของภาษาที่ใช้
  (เช่น secrets ใน Python หรือ crypto ใน Node.js) เพื่อให้แน่ใจว่ากุญแจแต่ละดอกมีความเป็นเอกลักษณ์สูง

การจัดเก็บและการแจกจ่ายกุญแจ (Key Storage & Distribution)

การจัดเก็บกุญแจ (Key Storage)

ระบบเน้นการลดความเสี่ยงจากการรั่วไหลของกุญแจ โดยกำหนดให้

- จัดเก็บกุญแจไว้ใน หน่วยความจำชั่วคราว (RAM) ในรูปแบบตัวแปรภายในโปรแกรมระหว่างที่แอปพลิเคชันทำงาน

- ไม่บันทึกกุญแจลับลงในหน่วยความจำถาวร (Hard Drive / Persistent Storage) เพื่อลดความเสี่ยงกรณีอุปกรณ์สูญหาย ถูกขโมย หรือถูกเข้าถึงไฟล์ระบบ

การแจกจ่ายกุญแจ (Key Distribution)

ระบบใช้แนวคิด Hybrid Encryption เพื่อแลกเปลี่ยนกุญแจอย่างปลอดภัย โดยมีขั้นตอนดังนี้

- ผู้รับส่ง Public Key ให้กับผู้ส่ง
- ผู้ส่งใช้ Public Key เข้ารหัสกุญแจ AES-256 (Session Key)
- ส่งกุญแจที่ถูกเข้ารหัสกลับไปยังผู้รับผ่านเครือข่าย
- ผู้รับใช้ Private Key ของตนถอดรหัสและนำกุญแจ AES ไปใช้งาน

การจัดการเมื่อกุญแจหมดอายุหรือรั่วไหล (Key Revocation & Expiration)

เป็นการวางแผนรับมือสถานการณ์ฉุกเฉินและกำหนดอายุการใช้งาน:

- กุญแจหมดอายุ (Expiration): ใช้กลไก "กุญแจต่อเซสชัน" (Ephemeral Session Key) คือการสร้างกุญแจ AES ชุดใหม่ทุกครั้งที่มีการเริ่มส่งไฟล์ชุดใหม่ เมื่อการส่งเสร็จสิ้นหรือปิดแอปพลิเคชัน กุญแจจะถูกลบออกจากหน่วยความจำทันที
- การรั่วไหล (Key Compromise): หากระบบตรวจพบว่าการเชื่อมต่อมีความผิดปกติ (เช่น Integrity Check ไม่ผ่านซ้ำๆ) ระบบจะทำการ ยกเลิกเซสชัน (Terminating Session) ทันที ผู้ใช้งานต้องทำการ "จับคู่" และตกลงกุญแจใหม่ทั้งหมด (Re-handshake) เพื่อสร้างกุญแจชุดใหม่ที่ไม่เกี่ยวข้องกับกุญแจเดิม

6. การวิเคราะห์ภัยคุกคาม (Threat Model)

ผู้โจมตีที่เป็นไปได้

- ผู้ไม่ประสงค์ดีภายในเครือข่ายท้องถิ่น (Internal Attacker)
- ผู้โจมตีแบบดักฟังข้อมูล (Passive Attacker)
- ผู้โจมตีแบบเชิงรุก (Active Attacker)

วิธีการโจมตี

- Man-in-the-Middle (MITM)
- การดักฟังข้อมูลบนเครือข่าย (Packet Sniffing / Eavesdropping)
- การปลอมแปลงตัวตนอุปกรณ์ (Device Spoofing / Impersonation)

แนวทางการลดความเสี่ยง

- การเข้ารหัสข้อมูลแบบต้นทางถึงปลายทาง (End-to-End Encryption)
- การสื่อสารผ่านช่องทางที่ปลอดภัย (TLS/HTTPS)
- การยืนยันอุปกรณ์ปลายทางก่อนการรับส่งไฟล์
- การตรวจสอบความถูกต้องของข้อมูล (Integrity Check)

ข้อจำกัดของระบบ (System Limitations)

- จำกัดการทำงานเฉพาะภายในเครือข่ายท้องถิ่น (LAN)
- ไม่ครอบคลุมการป้องกันมัลแวร์ในไฟล์

7. ข้อจำกัดของระบบ

ข้อจำกัดด้านเทคนิค (Technical Limitations)

- ขอบเขตเครือข่าย: ระบบค้นหาอัตโนมัติ (mDNS) ทำงานได้ดีเฉพาะในวง LAN เดียวกัน (Layer 2) ไม่รองรับการใช้งานข้าม Subnet
- ใบรับรองความปลอดภัย: การใช้ HTTPS ใน LAN ต้องใช้ Self-Signed Certificate ซึ่งทำให้ Browser แจ้งเตือนความไม่ปลอดภัยในครั้งแรก
- การตั้งค่า Firewall: การค้นหาและการรับส่งไฟล์อาจถูกบล็อกโดย OS Firewall หรือ Antivirus หากผู้ใช้ไม่ตั้งค่าอนุญาต

ข้อจำกัดด้านประสิทธิภาพ (Performance Limitations)

- ภาระการประมวลผล: การเข้ารหัส AES-256 ใช้ทรัพยากร CPU สูง อาจทำให้การส่งไฟล์ช้าลงในอุปกรณ์สเปคต่ำ
- ความเร็วเครือข่าย: ความเร็วในการรับส่งจริงถูกจำกัดด้วย Bandwidth ของ WiFi และสัญญาณรบกวนในขณะนั้น

8. สรุป

โครงงานนี้ได้นำหลักการเข้ารหัสข้อมูลและการจัดการกุญแจที่ได้มาตรฐานมาประยุกต์ใช้กับระบบรับ–ส่งไฟล์ภายในเครือข่ายท้องถิ่น โดยมุ่งเน้นการรักษาความลับ ความถูกต้อง และความน่าเชื่อถือของข้อมูล แม้ว่าจะยังมีข้อจำกัดด้านเครือข่ายและความปลอดภัยของอุปกรณ์ปลายทาง แต่ระบบที่พัฒนาขึ้นสามารถลดความเสี่ยงจากการดักฟังและการโจมตีในระดับเครือข่ายได้อย่างมีประสิทธิภาพ