โครงการจ้างพัฒนาระบบงาน
บริหารจัดการคุณภาพหนี้เพื่อยกระดับการให้บริการ
(Enhance Debt Management Service)
ธนาคารเพื่อการเกษตรและสหกรณ์การเกษตร

งวดงานที่ 6
Non-Functional Test (NFT)
- รายงานผลการทดสอบการเจาะระบบ (Penetration Test)

ditto DATA INTELLIGENCE บริษัท ดิทโต้ (ประเทศไทย) จำกัด (มหาชน)

ต้นฉบับ

ธนาคารเพื่อการเกษตรและสหกรณ์การเกษตร

รายงานผลการตรวจสอบความปลอดภัยของระบบ

การทดสอบเจาะระบบ Penetration test

โครงการสัญญาจ้างพัฒนาระบบงานบริหารจัดการคุณภาพหนี้เพื่อยกระดับการให้บริการ

รายงานเมื่อ วันที่ 10 พฤศจิกายน พ.ศ. 2568

โครงการสัญญาจ้างพัฒนาระบบงานบริหารจัดการคุณภาพหนี้เพื่อยกระดับการให้บริการ

ชื่อเอกสาร :	สัญญาจ้างพัฒนาระบบงานบริหารจัดการคุณภาพหนี้เพื่อยกระดับการให้บริการ	
คณะทำงาน :		
File name :	รายงานผลการตรวจสอบความปลอดภัยของระบบ Penetration test	
เลขที่ : 64006243	**Version no. :** 1.0	**Date :** 10 พฤศจิกายน 2568
ผู้เขียน : บริษัท บิซ โฟล์ค จำกัด		
ผู้ผลิตเอกสาร : บริษัท บิซ โฟล์ค จำกัด		
ประเภทเอกสาร :	**(/) เอกสารส่งมอบตามสัญญา** **( ) เอกสารภายในโครงการ** **( ) เอกสารส่งมอบเพิ่มเติม** **( ) รายงานประจำ** **( ) รายงานพิเศษ.....** **( ) อื่นๆ.....**	

Revision History

No.	Version (V.X.X)	Date of Change (YYYYMMDD)	A/M/D	Description of Change	Author
1	V1.0	20251110	A	Initial Version	Biz Folk

A = Added M = Modified D = Deleted

Project Sign off

บริษัท ดิตโต้ จำกัด (มหาชน)

ผู้จัดการโครงการ
10 พฤศจิกายน 2568

บริษัท บิซ โฟล์ค จำกัด

(ธนพล ไทรชมภู)
ผู้เชี่ยวชาญด้านความปลอดภัยไซเบอร์
10 พฤศจิกายน 2568

สารบัญ

สารบัญตาราง

สารบัญรูป

คำนำ

ความเป็นมา

ธนาคารเพื่อการเกษตรและสหกรณ์การเกษตร ซึ่งต่อไปนี้เรียกว่า “ธ.ก.ส.” มีความประสงค์จะบริหารจัดการลูกค้าที่ผิดนัดชำระหนี้อย่างมีประสิทธิภาพ เพื่อลดอัตราการเกิดหนี้เสียซึ่งกระทบกับต้นทุนโดยตรงและต้นทุนด้านความเสี่ยงของธนาคาร ความจำเป็นในการบริหารจัดการหนี้ด้วยกลยุทธ์ จากการวิเคราะห์ข้อมูลและเครื่องมือในการบริหารจัดการหนี้ที่เหมาะสมเป็นปัจจัยสำคัญที่จะช่วยปัญหาดังกล่าว ทางธนาคารสามารถใช้ทรัพยากร และบุคลากรที่มีอยู่ อย่างคุ้มค่าเพื่อลดความสูญเสียที่เกิดขึ้น โดยธนาคารสามารถบริหารจัดการหนี้ได้อย่างเป็นระบบและมีประสิทธิภาพ ปัจจุบันธนาคารได้ดำเนินโครงการพัฒนาระบบงานบริหารจัดการคุณภาพหนี้ (End To End Process) สำหรับช่วยงานในกระบวนการปรับปรุงโครงสร้างหนี้ที่กลุ่มลูกหนี้ที่เป็นหนี้ด้อยคุณภาพ (Non-Performing Loans: NPLs)

ธนาคารจึงได้มีแนวทางพัฒนาโครงการเพิ่มประสิทธิภาพระบบงานบริหารจัดการคุณภาพหนี้เพื่อยกระดับการให้บริการ (Enhance Debt Management Service) ซึ่งมีประกอบไปด้วย ระบบสามารถรองรับการทำงานได้ขณะมีสัญญาณอินเตอร์เน็ต (Online Mode) และไม่มีสัญญาณอินเตอร์เน็ต (Offline Mode) ให้สามารถทำผ่านช่องทางการให้บริการของธนาคาร เช่น ช่องทางคอมพิวเตอร์แท็บเล็ต (Tablet Computer) ของธนาคาร ช่องทางคอมพิวเตอร์บุคคล (Personal Computer) คอมพิวเตอร์โน้ตบุ๊ก (Laptop Computer) ของธนาคาร เชื่อมต่อกับฐานข้อมูลศูนย์ข้อมูลส่วนกลางของธนาคาร เพื่อให้สามารถจัดเก็บข้อมูลและการกำหนดตัวชี้วัดผลงาน (KPI) รวมถึงเชื่อมต่อกับระบบต่างๆ ของธนาคารได้ ด้วย API ที่เป็นอย่างน้อย โดยมีรายละเอียดขอบเขตของงาน (Terms of Reference: TOR) ตามที่แนบเอกสารฉบับนี้

วัตถุประสงค์ของโครงการ

1. เพื่อเพิ่มประสิทธิภาพระบบงานบริหารจัดการคุณภาพหนี้ สนับสนุนการขับเคลื่อนงานบริหารจัดการคุณภาพหนี้ให้บรรลุเป้าหมายตามอัตรากำหนด
2. เพื่อให้ธนาคารสามารถบริหารจัดการลูกหนี้ที่เป็นหนี้ด้อยคุณภาพได้อย่างมีประสิทธิภาพ
3. เพื่อสนับสนุนช่วยเจ้าหน้าที่ โดยการลดภาระค่าใช้จ่ายในการเดินทางมาติดต่อธนาคาร

บทที่ 1 บทนำ (Introduction)

1.1 ภาพรวมของการทดสอบ

การทดสอบเจาะระบบ (Penetration Test) เป็นกระบวนการที่มุ่งเน้นการประเมินความมั่นคงปลอดภัยของระบบโดยการจำลองการโจมตีระบบเพื่อค้นหาช่องโหว่และประเมินความทนทานของระบบต่อการโจมตีจริง ๆ ในขณะเดียวกัน การทดสอบเจาะระบบมุ่งเน้นการทดสอบด้วยพฤติกรรมของผู้โจมตีซึ่งมุ่งหมายร้ายและใช้วิธีการโจมตีทั้งตรงและทางอ้อมในการเข้าถึงข้อมูล การทดสอบนี้ช่วยในการค้นหาข้อบกพร่องและช่องโหว่ที่อาจเกิดขึ้นในระบบเพื่อให้ผู้ดูแลระบบสามารถปรับปรุงแก้ไขช่องโหว่เหล่านั้นเพื่อเสริมความปลอดภัยของระบบให้มีความทนทานมากยิ่งขึ้นต่อการโจมตีในอนาคต การทดสอบเจาะระบบเป็นเครื่องมือที่สำคัญในกระบวนการรักษาความปลอดภัยของระบบสารสนเทศและส่งเสริมให้เกิดการปรับปรุงความเชื่อถือได้ในระบบนั้น ๆ ในองค์กร

การทดสอบครั้งนี้เป็นการดำเนินการทดสอบเจาะระบบ (Penetration Test) รูปแบบ Grey-box และ Black-box Penetrationt test บนระบบงานบริหารจัดการคุณภาพหนี้เพื่อยกระดับการให้บริการ โดยประกอบไปด้วย การรักษาความปลอดภัยแอปพลิเคชัน (Application Security), การจัดการสิทธิ์ผู้ใช้ (Access Control), และความเสี่ยงด้านข้อมูลที่เกี่ยวข้องกับธุรกรรม โดยรายละเอียดเป้าหมายแสดงในตารางแสดงเครื่องเป้าหมายบนสภาพแวดล้อม UAT Environment

ตารางที่ : 1 รายละเอียดเป้าหมายสำหรับการทดสอบ

ที่	เป้าหมายการทดสอบ
1	https://e2e-m1-testapp.app.baac.or.th
2	https://e2e-m2-testapp.app.baac.or.th
3	https://e2e-uat-m3ui.app.baac.or.th
4	https://e2e-m4-testapp.app.baac.or.th
5	https://uat-efiling.app.baac.or.th

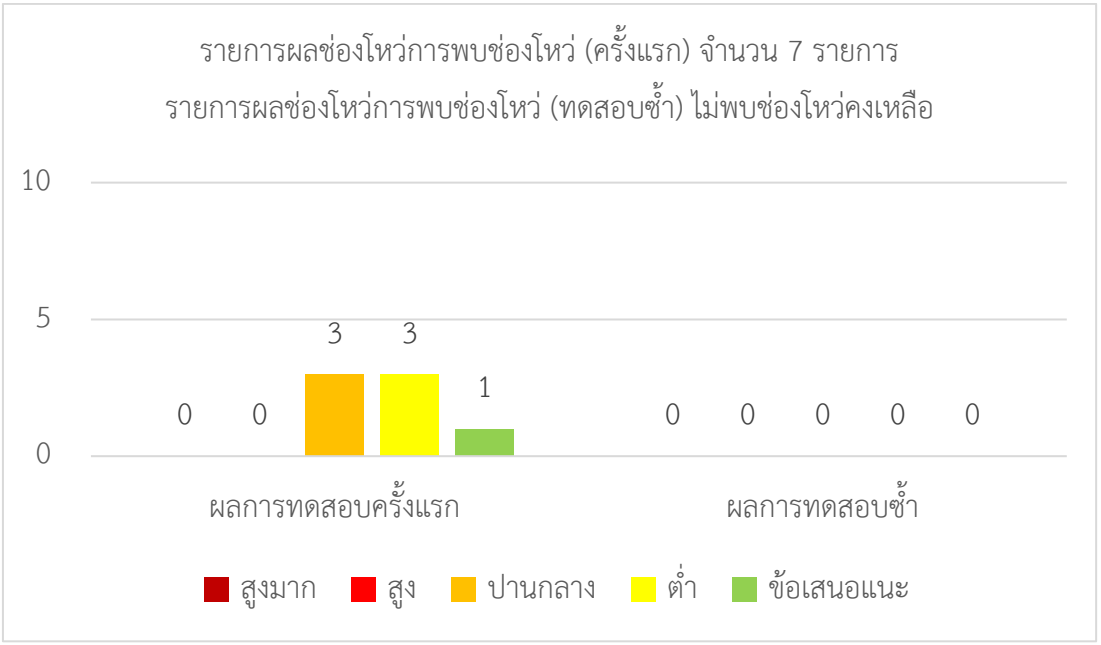
1.2 วัตถุประสงค์

วัตถุประสงค์ของการทดสอบ มีดังนี้

- เพื่อค้นหา ตรวจสอบ และระบุช่องโหว่ที่อาจมีอยู่ในระบบเป้าหมายขององค์กร เพื่อเปิดเผยจุดอ่อนที่อาจถูกผู้ไม่หวังดีใช้ประโยชน์ในการโจมตี
- เพื่อประเมินความเสี่ยงที่เกี่ยวข้องกับแต่ละช่องโหว่ที่ตรวจพบ โดยจำแนกตามระดับความรุนแรงของความเสี่ยงและจัดลำดับความสำคัญในการดำเนินการแก้ไขหรือบรรเทาผลกระทบที่อาจเกิดขึ้น
- เพื่อให้ข้อเสนอแนะในการเพิ่มประสิทธิผลในการป้องกันการโจมตีจากผู้บุกรุก และปรับปรุงมาตรการป้องกันและระบบความปลอดภัยตามที่ได้รับจากการทดสอบ

1.3 การวิเคราะห์ระดับความเสี่ยงตามรายการช่องโหว่ที่ตรวจพบ

ในการทดสอบครั้งแรกพบช่องโหว่ จำนวน 7 รายการ โดยแบ่งตามระดับความเสี่ยงได้ดังนี้ ความเสี่ยงปานกลาง จำนวน 3 รายการ ความเสี่ยงต่ำ จำนวน 3 รายการ และ บันทึกไว้ จำนวน 1 รายการ และได้รับการแก้ไขทั้งหมด โดยแสดงตามกราฟที่แสดงด้านล่าง



รูปที่: 1 กราฟแสดงจำนวนของระดับความเสี่ยงที่วิเคราะห์ตามรายการช่องโหว่ที่ตรวจพบ

บทที่ 2 ขอบเขตการดำเนินงาน

2.1 ระยะเวลาการทดสอบ

ตารางที่ : 2 แสดงการทดสอบได้ดำเนินการตามรายการ

ID	กิจกรรม	วันที่ดำเนินการ	หมายเหตุ
1	Penetration test Initial Testing	24 ตุลาคม 2568 - 6 พฤศจิกายน 2568	
2	Remediation by Developer	6 - 9 พฤศจิกายน 2568	
3	Penetration test Re-visit Testing	10 พฤศจิกายน 2568	
4	Deliver Re-visit Report	10 พฤศจิกายน 2568	

2.2 เครื่องมือที่ใช้ในการทดสอบ

เพื่อเพิ่มประสิทธิภาพและความครอบคลุมในการวิเคราะห์ช่องโหว่ ทีมทดสอบได้ใช้วิธีการแบบผสมผสานโดยใช้ทั้งเครื่องมือแบบอัตโนมัติและใช้คนเป็นผู้ทดสอบร่วมกัน เครื่องมือที่ใช้ในกระบวนการนี้มีดังนี้
ตู้ BAAC ATM (Automated Transfer Machine)

ตารางที่ : 3 แสดงเครื่องมือที่ใช้ทดสอบตามรายการ

ที่	ชื่อเครื่องมือ	คำอธิบาย	Tool type
1.	NMAP (Network Mapper)	เป็นเครื่องมือสำหรับสแกนและวิเคราะห์เครือข่ายที่ใช้ในการตรวจสอบและค้นหาเครื่องแม่ข่าย (hosts) และบริการ (services) ที่ทำงานอยู่บนเครือข่าย มันมีความสามารถในการสแกนพอร์ต (port scanning) เพื่อตรวจสอบว่าพอร์ตใดๆ บนเครื่องเป้าหมายถูกเปิดอยู่หรือไม่ รวมถึงการระบุระบบปฏิบัติการที่ทำงานอยู่บนเครื่องเหล่านั้นด้วย นอกจากนี้ NMAP ยังมีฟังก์ชันการสแกนแบบมาตรฐานและแบบลึกเพื่อตรวจสอบช่องโหว่และความปลอดภัยของระบบ เป็นเครื่องมือที่ถูกนำมาใช้ในการทดสอบความเสถียรและ	Automatic

ที่	ชื่อเครื่องมือ	คำอธิบาย	Tool type
		ความปลอดภัยของเครือข่ายในการทดสอบเจาะระบบและการประเมินความเสี่ยงของระบบคอมพิวเตอร์.	
2.	sslscan/testssl.sh	เป็นเครื่องมือสำหรับการทดสอบความปลอดภัยของเซิร์ฟเวอร์ที่ใช้โปรโตคอล SSL/TLS ซึ่งมักถูกใช้เพื่อตรวจสอบการกำหนดค่าและการใช้งานของ SSL/TLS บนเซิร์ฟเวอร์หรือเว็บไซต์ เครื่องมือเหล่านี้ช่วยในการค้นหาปัญหาที่อาจเกิดขึ้นเกี่ยวกับความปลอดภัย เช่น ช่องโหว่ในการกำหนดค่า SSL/TLS, การใช้งานของสัญญาณดิจิทัล (certificates) และการตรวจสอบความปลอดภัยของเว็บไซต์ที่ใช้ SSL/TLS รวมถึงปัญหาที่เกี่ยวข้องกับการเชื่อถือของผู้ออก Certificate Authority (CA) ด้วย	Automatic
3.	Nessus	เป็นเครื่องมือการตรวจสอบความปลอดภัยแบบอัตโนมัติ (Automated Security Scan) ซึ่งมีความสามารถในการสแกนและตรวจสอบหาช่องโหว่ที่อาจมีความเสี่ยงต่อระบบคอมพิวเตอร์และเครือข่าย	Automatic
4.	Wireshark	เป็นเครื่องมือวิเคราะห์โปรโตคอลเครือข่ายแบบโอเพนซอร์ส ใช้สำหรับตรวจจับและวิเคราะห์แพ็คเก็ตข้อมูลที่ส่งผ่านเครือข่าย สามารถดักจับและแสดงรายละเอียดของแพ็คเก็ตแบบเรียลไทม์ รวมถึงการกรองและค้นหาแพ็คเก็ตตามเงื่อนไขต่างๆ ช่วยในการแก้ไขปัญหาเครือข่าย การตรวจสอบความปลอดภัย และการวิเคราะห์โปรโตคอล	Manual
5.	Ettercap	เป็นเครื่องมือที่ใช้สำหรับการโจมตีเครือข่ายแบบ Man-in-the-middle Attack การตรวจจับและการบุกรุกเครือข่าย (Network sniffing and Network attack) รวมถึงการดักจับข้อมูลที่ถูกส่งผ่านเครือข่าย เช่น รหัสผ่าน (passwords),	Manual

ที่	ชื่อเครื่องมือ	คำอธิบาย	Tool type
		เฟรมเวิร์ก (network frames), และแพ็คเก็ตข้อมูลอื่น ๆ	

2.3 ทีมทดสอบ

ตารางที่ : 4 การทดสอบได้ดำเนินการโดยทีมทดสอบตามรายชื่อในตารางด้านล่าง

ที่	รายชื่อผู้ทดสอบ	Certification	ประสบการณ์การทำงาน
1	Warunyou Sunpachit	OCSP, CEH, CompTIA Security+, CompTIA Pentest+, CC	ดำเนินการประเมินช่องโหว่และทดสอบเจาะระบบในระดับ Mobile Application, Web Application, Network, Thick-client Application ในธุรกิจการเงินการธนาคาร โทรคมนาคม ประภัยภัย และอื่นๆ โดยมีประสบการณ์ด้านการประเมินช่องโหว่และทดสอบเจาะระบบมากกว่า 15 ปี
2	Pongsathon Sirithanyakul	OCSP, CEH, CompTIA Security+, CompTIA Pentest+, CC	ดำเนินการประเมินช่องโหว่และทดสอบเจาะระบบในระดับ Mobile Application, Web Application, Network, Thick-client Application ในธุรกิจการเงินการธนาคาร โทรคมนาคม ประภัยภัย และอื่นๆ โดยมีประสบการณ์ด้านการประเมินช่องโหว่และทดสอบเจาะระบบมากกว่า 7 ปี

2.4 ข้อจำกัด

การทดสอบนี้เป็นการทดสอบจุดในเวลาที่กำหนดเท่านั้น (point-in-time test) และการประเมินความปลอดภัยถูกจำกัดอยู่ในช่วงเวลาที่ตกลงกันไว้ เวลาที่เพิ่มเติมและการทดสอบเพิ่มเติมอาจจะเปิดเผยสิ่งที่สังเกตเห็นได้อีก ขอแนะนำให้ดำเนินการทดสอบช่องโหว่เพิ่มเติมรอบต่อไป รวมถึงการทดสอบการเจาะระบบบนระบบเป็นระยะหรือเมื่อมีการเปลี่ยนแปลงที่สำคัญ เช่น การเปลี่ยนเวอร์ชันซอฟต์แวร์ใหม่หรือการติดตั้งอุปกรณ์ใหม่ การเปลี่ยนแปลงที่สำคัญเช่นนี้อาจเป็นเหตุให้เกิดช่องโหว่ใหม่ ๆ ได้

บทที่ 3 แนวทางการทดสอบ

3.1 วิธีการที่ใช้ในการทดสอบ

วัตถุประสงค์ของการประเมินนี้คือการตรวจพบช่องโหว่หรือจุดอ่อนที่ต้องการปรับปรุง และให้ข้อเสนอแนะในการแก้ไขปัญหาที่พบและรายละเอียดของความต้องการในเชิงปฏิบัติการ การประเมินครอบคลุมถึงสามประเภทของการทดสอบช่องโหว่และการทดสอบเจาะระบบตามตารางด้านล่าง

ตารางที่ : 5 ประเภทของการทดสอบหาช่องโหว่และการทดสอบเจาะระบบ

การทดสอบแบบกล่องดำ (Black box)	การทดสอบแบบกล่องเทา (Grey box)	การทดสอบแบบกล่องขาว (White box)
1) ไม่จำเป็นต้องให้ข้อมูลเกี่ยวกับเป้าหมายของระบบ 2) จำเป็นต้องยืนยันเป้าหมายก่อนทดสอบเจาะระบบ เพื่อป้องกันการทดสอบนอกขอบเขต 3) ไม่ให้บัญชีผู้ใช้สำหรับทดสอบเจาะระบบ	1) กำหนดระบบเป้าหมาย 2) ให้บัญชีผู้ใช้ในแต่กลุ่ม 3) ให้คำอธิบายรายละเอียดการใช้งานระบบแต่ละเมนู	1) ให้ source-codes ของระบบนั้น ๆ 2) ให้คำอธิบายรายละเอียดของแต่ละฟังก์ชันการใช้งาน

ทางทีมงานได้พัฒนากระบวนการการทดสอบเจาะระบบ จากกรอบการทดสอบต่าง ๆ (Framework) เพื่อนำมาประยุกต์ใช้ให้เหมาะสมกับสภาพแวดล้อมการทดสอบเจาะระบบนั้น ๆ เช่น กระบวนการทดสอบความปลอดภัยทั้งด้านความตระหนักการใช้งานสารสนเทศ (Human) ทางด้านกายภาพ (Physical) การทดสอบโครงข่ายไร้สาย (Wireless) จาก Open Source Security Testing Methodology Manual (OSSTMM) v3 การทดสอบเจาะระบบของระบบสารสนเทศบนเครื่องโทรศัพท์มือถือ Mobile Application Security Verification Standard (MASVS) และ OWASP Mobile Security Testing Guide (MSTG) การทดสอบเจาะระบบเว็บแอปพลิเคชัน OWASP Testing Guide (OTG) v4 เป็นต้น

3.2 ข้อมูลอ้างอิงที่ใช้ประเมินความเสี่ยง

3.2.1 OWASP Top 10 Application Security Risks - 2021

เป็นรายงานที่กำหนดและระบุปัญหาที่เกี่ยวข้องกับความปลอดภัยในการพัฒนาและดำเนินการแอปพลิเคชันที่พบบ่อยในการใช้งานจริง รายงานนี้ถูกเผยแพร่โดย OWASP (Open Web Application Security Project) ซึ่งมุ่งเน้นการเสริมสร้างความปลอดภัยในเว็บแอปพลิเคชันและซอฟต์แวร์ การระบุปัญหาความปลอดภัยที่สำคัญที่ถูกนำเสนอในรายงานนี้ช่วยให้นักพัฒนาและผู้ดูแลระบบสามารถระบุและแก้ไขปัญหาด้านความปลอดภัยในขั้นตอนการพัฒนาและดำเนินการแอปพลิเคชันได้อย่างมีประสิทธิภาพ

ตารางที่ : 6 รายงานของ OWASP Top 10 Application Security Risks – 2021 (https://owasp.org/Top10/)

OWASP Top 10 Application 2021	คำอธิบาย
A1: Broken Access Control	ความเสี่ยงที่เกี่ยวข้องกับปัญหาในการควบคุมการเข้าถึงข้อมูลหรือฟังก์ชันในระบบโดยไม่ถูกต้องหรือไม่เพียงพอ ซึ่งอาจทำให้ผู้ไม่ประสงค์ดีสามารถเข้าถึงข้อมูลหรือทำรายการที่ไม่ได้รับอนุญาต รายละเอียดการทำงานอาจรวมถึง: 1. การเข้าถึงข้อมูลที่ไม่ได้รับอนุญาต: ผู้ไม่ประสงค์ดีอาจสามารถเข้าถึงข้อมูลที่มีความลับหรือที่สำคัญโดยไม่ได้รับอนุญาต โดยเรียกใช้ข้อมูลที่ไม่ถูกคัดกรองหรือไม่มีการตรวจสอบการสิทธิ์การเข้าถึง 2. การเปลี่ยนแปลงข้อมูลที่ไม่ได้รับอนุญาต: ผู้ไม่ประสงค์ดีอาจทำการเปลี่ยนแปลงข้อมูลที่สำคัญหรือมีความสำคัญโดยไม่ได้รับอนุญาต ซึ่งอาจส่งผลให้เกิดความเสียหายหรือความผิดพลาดในระบบ 3. การทำรายการที่ไม่ได้รับอนุญาต: ผู้ไม่ประสงค์ดีอาจดำเนินการรายการที่ส่งผลกระทบที่ไม่พึงประสงค์ต่อระบบหรือข้อมูล โดยใช้ฟังก์ชันหรือความสามารถที่ไม่ได้รับอนุญาต การแก้ไขปัญหาที่เกี่ยวข้องกับรายงาน A1 สามารถทำได้โดยการใช้ระบบการควบคุมการเข้าถึงที่เข้มงวดและครอบคลุม รวมถึงการทดสอบและตรวจสอบความปลอดภัยของระบบอย่างสม่ำเสมอ

OWASP Top 10 Application 2021	คำอธิบาย
A2: Cryptographic Failures	ความเสี่ยงที่เกี่ยวข้องกับปัญหาในการใช้เทคนิคการเข้ารหัสที่ไม่ถูกต้องหรือไม่เพียงพอ ซึ่งอาจทำให้ข้อมูลที่เกี่ยวข้องกับความลับหรือความปลอดภัยถูกเปิดเผยหรือถูกแยกจากกันได้ รายละเอียดการทำงานอาจรวมถึง: 1. การใช้วิธีการเข้ารหัสที่ไม่เพียงพอ: การใช้วิธีการเข้ารหัสที่ไม่เหมาะสมหรือมีความเสี่ยงสูง เช่น การใช้วิธีการเข้ารหัสที่ถูกทำลายหรือเป็นที่รู้จักในวงการว่าไม่ปลอดภัย 2. ความยากจนของกุญแจ: การสร้างและการจัดการกับกุญแจการเข้ารหัสที่ไม่ปลอดภัย ซึ่งอาจทำให้ผู้ไม่ประสงค์ดีสามารถทำการระบาดไวรัสหรือเข้าถึงข้อมูลที่สำคัญได้ 3. การจัดการกับการรักษาความลับ: การเก็บรักษาและการจัดการกับข้อมูลที่เข้ารหัสอาจมีข้อบกพร่องหรือการจัดการที่ไม่เหมาะสมทำให้ข้อมูลถูกเปิดเผยหรือถูกแยกจากกัน 4. การบูรณะระบบเข้ารหัส: การประยุกต์ใช้ระบบการเข้ารหัสที่มีข้อบกพร่องหรือมีการเลือกใช้แบบไม่ถูกต้อง ซึ่งอาจทำให้ข้อมูลเสี่ยงต่อการโจมตี การแก้ไขปัญหาที่เกี่ยวข้องกับรายงาน A2 สามารถทำได้โดยการใช้เทคโนโลยีการเข้ารหัสที่มีความปลอดภัยและเหมาะสม รวมถึงการปรับปรุงกระบวนการสร้างและจัดการกับคีย์เวิร์ดการเข้ารหัสอย่างระมัดระวังและมีประสิทธิภาพ
A3: Injection	ความเสี่ยงที่เกี่ยวข้องกับการสั่งการไม่ประสงค์ดีที่ฝังเข้าไปในข้อมูลหรือคำสั่งที่รับเข้ามาในระบบโดยไม่ได้รับการตรวจสอบหรือกรองอย่างเหมาะสม ซึ่งอาจทำให้เกิดการโจมตีแบบ Injection เช่น SQL Injection, Command Injection, LDAP Injection, หรือ XML Injection รายละเอียดการทำงานของรายงาน A3 สามารถรวมถึง:

OWASP Top 10 Application 2021	คำอธิบาย
	1. SQL Injection: การฝังโค้ด SQL ที่ไม่ประสงค์ดีในข้อมูลที่ถูกส่งเข้ามาในแอปพลิเคชันเพื่อเข้าถึงหรือปรับเปลี่ยนข้อมูลในฐานข้อมูล 2. Command Injection: การฝังโค้ดคำสั่งระบบที่ไม่ประสงค์ดีในข้อมูลที่ถูกส่งเข้ามาในแอปพลิเคชันเพื่อรันคำสั่งในระบบหรือเข้าถึงข้อมูลที่ไม่ได้รับอนุญาต 3. LDAP Injection: การฝังโค้ด LDAP ที่ไม่ประสงค์ดีในข้อมูลที่ถูกส่งเข้ามาในแอปพลิเคชันเพื่อโจมตีบริการไดเรกทอรีและเข้าถึงข้อมูลที่สำคัญ 4. XML Injection: การฝังโค้ด XML ที่ไม่ประสงค์ดีในข้อมูล XML ที่ถูกส่งเข้ามาในแอปพลิเคชันเพื่อโจมตีระบบหรือเปิดเผยข้อมูลที่ละเมิดความเป็นส่วนตัว การป้องกันการโจมตีแบบ Injection สามารถทำได้โดยการใช้พื้นที่จำกัดสิทธิ์และกรองข้อมูลอย่างเหมาะสมก่อนที่จะส่งเข้าสู่ระบบ รวมถึงการใช้พื้นที่จำกัดสิทธิ์ในการเข้าถึงฐานข้อมูล และการใช้พื้นที่จำกัดสิทธิ์ในการรันคำสั่งระบบหรือเรียกใช้บริการจากภายนอก
A4: Insecure Design	ความเสี่ยงที่เกี่ยวข้องกับการออกแบบและสร้างแอปพลิเคชันโดยไม่คำนึงถึงมาตรฐานความปลอดภัยและหลักการออกแบบที่ปลอดภัยอย่างเพียงพอ ซึ่งอาจทำให้เกิดช่องโหว่และความไม่ปลอดภัยต่าง ๆ ในแอปพลิเคชัน รายละเอียดการทำงานของรายงาน A4 สามารถรวมถึง: 1. การขาดความสามารถในการจำกัดสิทธิ์: การขาดความสามารถในการจำกัดสิทธิ์และควบคุมการเข้าถึงข้อมูลหรือฟังก์ชันของแอปพลิเคชัน ซึ่งอาจทำให้ผู้ไม่ประสงค์ดีสามารถเข้าถึงข้อมูลที่ไม่เหมาะสมหรือทำการกระทำที่ไม่เหมาะสม 2. การไม่คำนึงถึงการป้องกันที่มีชั้นของข้อมูล: การออกแบบแอปพลิเคชันโดยไม่คำนึงถึงการป้องกันที่มีชั้นของข้อมูล ซึ่งอาจทำให้ข้อมูลที่สำคัญถูก

OWASP Top 10 Application 2021	คำอธิบาย
	เปิดเผยหรือถูกใช้โดยผู้ไม่ประสงค์ดี 3. การขาดความสามารถในการตรวจสอบและการติดตาม: การขาดความสามารถในการตรวจสอบและติดตามกิจกรรมที่เกิดขึ้นในระบบ เช่น การไม่มีการบันทึกล็อกหรือไม่มีการแจ้งเตือนเมื่อมีกิจกรรมที่เป็นไปไม่ได้ การแก้ไขปัญหาที่เกี่ยวข้องกับรายงาน A4 สามารถทำได้โดยการออกแบบและพัฒนาแอปพลิเคชันโดยให้คำนึงถึงความปลอดภัยตั้งแต่ต้น รวมถึงการใช้หลักการและเทคโนโลยีที่ปลอดภัย เพื่อลดความเสี่ยงของช่องโหว่และความไม่ปลอดภัยในแอปพลิเคชัน
A5: Security Misconfiguration	ความเสี่ยงที่เกี่ยวข้องกับการกำหนดค่าและการตั้งค่าระบบหรือแอปพลิเคชันโดยไม่ถูกต้องหรือไม่เพียงพอทำให้เกิดช่องโหว่ที่เปิดเผยข้อมูลหรือทำให้เกิดความไม่ปลอดภัย รายละเอียดการทำงานของรายงาน A5 สามารถรวมถึง: 1. ค่าเริ่มต้นที่ไม่ปลอดภัย: การใช้ค่าเริ่มต้นที่ไม่ปลอดภัยหรือค่าที่มีความเสี่ยง เช่น รหัสผ่านเริ่มต้นที่ถูกตั้งไว้, การเปิดใช้งานฟีเจอร์ที่ไม่จำเป็น 2. การกำหนดค่าสิทธิ์ไม่ถูกต้อง: การกำหนดค่าสิทธิ์หรือสิทธิ์ที่ไม่เพียงพอ ทำให้ผู้ไม่ประสงค์ดีสามารถเข้าถึงข้อมูลหรือฟังก์ชันที่ไม่ได้รับอนุญาตได้ 3. การทิ้งรายละเอียดการทำงานที่ไม่จำเป็น: การทิ้งรายละเอียดการทำงานหรือข้อมูลที่ไม่จำเป็นทำให้ผู้ไม่ประสงค์ดีสามารถใช้ข้อมูลนั้นในการโจมตีได้ 4. ขาดการอัปเดตและการจัดการเวอร์ชั่น: การขาดการอัปเดตซอฟต์แวร์หรือการจัดการเวอร์ชั่นที่สามารถทำให้เกิดช่องโหว่ที่มีชั้นความปลอดภัยต่ำหรือทำให้เกิดความไม่ปลอดภัยในระบบ การแก้ไขปัญหาที่เกี่ยวข้องกับรายงาน A5 สามารถทำได้โดยการตั้งค่าระบบและแอปพลิเคชันให้เป็นไปตามหลักการและมาตรฐานความปลอดภัย รวมถึง

OWASP Top 10 Application 2021	คำอธิบาย
	การทดสอบความปลอดภัยและการตรวจสอบการตั้งค่าเพื่อให้แน่ใจว่าไม่มีความเสี่ยงหรือช่องโหว่ใดๆ ที่เปิดเผยข้อมูลหรือทำให้เกิดความไม่ปลอดภัยในระบบ
A6: Vulnerable and Outdated Components	ความเสี่ยงที่เกี่ยวข้องกับการใช้งานส่วนประกอบของระบบหรือแอปพลิเคชันที่มีช่องโหว่หรือเวอร์ชันเก่าที่มีความไม่ปลอดภัย ซึ่งอาจทำให้เกิดความเสี่ยงและช่องโหว่ต่าง ๆ ในระบบ รายละเอียดการทำงานของรายงาน A6 สามารถรวมถึง: 1. การใช้งานส่วนประกอบที่มีช่องโหว่: การใช้งานและการนำเข้าส่วนประกอบหรือไลบรารีที่มีช่องโหว่ในระบบ เช่น ไลบรารีที่มีช่องโหว่ของชนิดซอฟต์แวร์ต่าง ๆ หรือชุดเครื่องมือที่ไม่ปลอดภัย 2. การใช้งานส่วนประกอบเวอร์ชันเก่า: การใช้งานส่วนประกอบหรือไลบรารีที่เป็นเวอร์ชันเก่าและมีช่องโหว่ที่ได้รับการแก้ไขในเวอร์ชันล่าสุด 3. การขาดการอัปเดตและการจัดการเวอร์ชั่น: การขาดการอัปเดตและการจัดการเวอร์ชั่นที่สามารถทำให้เกิดความไม่ปลอดภัยในระบบ และช่องโหว่ที่มีชั้นความปลอดภัยต่ำ การแก้ไขปัญหาที่เกี่ยวข้องกับรายงาน A6 สามารถทำได้โดยการทำการสแกนและตรวจสอบส่วนประกอบของระบบหรือแอปพลิเคชันเพื่อตรวจจับส่วนประกอบที่มีช่องโหว่หรือเวอร์ชันเก่า และทำการอัปเดตหรือแก้ไขส่วนประกอบเหล่านั้นเพื่อให้มีความปลอดภัยที่สูงขึ้น
A7: Identification and Authentication Failures	เป็นหัวหัวรายงานที่ให้ความสำคัญกับปัญหาเกี่ยวกับการระบุตัวตนและการรับรองตัวตน ซึ่งเป็นส่วนสำคัญของความปลอดภัยของระบบ เมื่อมีความผิดพลาดเกิดขึ้นในการระบุหรือการรับรองตัวตน ผู้ไม่ประสงค์ดีอาจมีโอกาสเข้าถึงข้อมูลหรือฟังก์ชันที่ไม่เหมาะสม หรือกระทำการในฐานะผู้ใช้ที่ถูกสมมติให้มีสิทธิ์สูง โดยไม่ได้รับการตรวจสอบอย่างเหมาะสม รายละเอียดที่เกี่ยวข้องอาจรวมถึง:

OWASP Top 10 Application 2021	คำอธิบาย
	1. Weak Password Policies: นโยบายที่ไม่เข้มงวดเกี่ยวกับการตั้งรหัสผ่านที่มีความปลอดภัย ทำให้ผู้ไม่ประสงค์ดีสามารถทำการทดลองเดารหัสผ่านหรือใช้วิธีการต่าง ๆ เพื่อบล็อกเข้าสู่ระบบ 2. Brute Force Attacks: การโจมตีโดยการทดลองเดารหัสผ่านจนกว่าจะพบรหัสที่ถูกต้อง ซึ่งอาจเป็นไปได้เมื่อระบบไม่มีการจำกัดจำนวนครั้งที่ลองเดารหัสเข้าสู่ระบบ 3. Weak Multifactor Authentication (MFA): การใช้วิธีการยืนยันตัวตนที่มีความปลอดภัยไม่เพียงพอ เช่น การใช้ SMS ในการยืนยันตัวตนซึ่งเป็นระบบที่มีช่องโหว่ 4. Session Management: ปัญหาที่เกี่ยวข้องกับการจัดการเซสชัน ซึ่งอาจทำให้ผู้ไม่ประสงค์ดีสามารถทำให้เซสชันหรือการตั้งค่าของผู้ใช้หมดอายุ หรือแยกจากการยืนยันตัวตน 5. Insufficient Account Lockout: การไม่มีการล็อคบัญชีหลังจากจำนวนครั้งที่พยายามเข้าสู่ระบบผิดพลาดเกินกำหนด อาจทำให้ผู้ไม่ประสงค์ดีสามารถดำเนินการโจมตีด้วยวิธีการทดลองเดารหัสผ่านได้โดยไม่มีข้อจำกัด การแก้ไขปัญหาในหัวข้อนี้จำเป็นต้องมีการสร้างนโยบายความปลอดภัยที่เข้มงวดเกี่ยวกับการระบุตัวตนและการรับรองตัวตน รวมถึงการใช้เทคโนโลยีที่มีความปลอดภัยมากขึ้น เช่น การใช้ MFA หรือการใช้วิธีการรับรองตัวตนที่มีความเข้มงวดมากขึ้น
A8: Software and Data Integrity Failures	ความเสี่ยงที่เกี่ยวข้องกับความผิดพลาดในซอฟต์แวร์และข้อมูลที่สามารถทำให้ข้อมูลหรือซอฟต์แวร์ถูกเปลี่ยนแปลงหรือทำลายโดยไม่ได้รับอนุญาต รายละเอียดการทำงานที่เกี่ยวข้องอาจรวมถึง: 1. ข้อบกพร่องในซอฟต์แวร์: การพัฒนาซอฟต์แวร์ที่มีข้อบกพร่องอาจทำให้มีช่องโหว่ที่ทำให้ผู้ไม่ประสงค์ดีสามารถเข้าถึงระบบหรือข้อมูลโดยไม่ได้รับ

OWASP Top 10 Application 2021	คำอธิบาย
	อนุญาต 2. การสูญเสียข้อมูล: สูญเสียข้อมูลสำคัญหรือสำรองข้อมูลที่สำคัญอาจเกิดขึ้นเนื่องจากการขัดข้องในระบบหรือซอฟต์แวร์ 3. การเปลี่ยนแปลงข้อมูลที่ไม่ได้รับอนุญาต: การโจมตีที่เปลี่ยนแปลงข้อมูลที่สำคัญโดยไม่ได้รับอนุญาต เช่น การเขียนข้อมูลลงในฐานข้อมูลหรือไฟล์ที่สำคัญ 4. การบิดเบือนข้อมูล: การเปลี่ยนแปลงค่าข้อมูลที่ส่งผลกระทบต่อการประมวลผลหรือการตัดสินใจโดยไม่ได้รับอนุญาต 5. การสร้างซอฟต์แวร์ที่มีความไม่มั่นคง: การพัฒนาซอฟต์แวร์ที่มีข้อบกพร่องหรือช่องโหว่อาจสร้างความไม่มั่นคงในการทำงานของระบบและข้อมูล การจัดการความปลอดภัยของซอฟต์แวร์และข้อมูล รวมถึงการใช้เทคโนโลยีและมาตรการที่เหมาะสม เป็นสิ่งสำคัญในการป้องกันการสูญเสียข้อมูลและความไม่เสถียรของระบบในสถานการณ์ที่ไม่คาดคิด
A9: Security Logging and Monitoring Failures	ความเสี่ยงที่เกี่ยวข้องกับปัญหาในการบันทึกและตรวจสอบความปลอดภัยของระบบ ซึ่งอาจทำให้ระบบไม่สามารถตรวจจับหรือระบุกิจกรรมที่ไม่เหมาะสมหรือเกิดความผิดปกติในระบบได้อย่างทันท่วงที รายละเอียดการทำงานที่เกี่ยวข้องอาจรวมถึง: 1. ขาดความสามารถในการบันทึกเหตุการณ์: ระบบที่ไม่สามารถบันทึกเหตุการณ์หรือกิจกรรมที่เกิดขึ้นในระบบได้อย่างครอบคลุม ทำให้ไม่สามารถตรวจสอบหรือวิเคราะห์กิจกรรมที่เป็นอันตรายได้อย่างมีประสิทธิภาพ 2. ข้อบกพร่องในการตรวจจับการลักษณะเฉพาะของการโจมตี: ระบบที่ไม่สามารถตรวจจับและรายงานเหตุการณ์ที่อาจเป็นการโจมตีหรือกิจกรรมที่เกี่ยวข้องกับความมั่นคงของระบบ

OWASP Top 10 Application 2021	คำอธิบาย
	3. ข้อบกพร่องในการแจ้งเตือนเหตุการณ์: ระบบที่ไม่สามารถแจ้งเตือนผู้ดูแลระบบหรือผู้ใช้ที่เกี่ยวข้องเมื่อเกิดเหตุการณ์ที่สำคัญหรือเป็นอันตราย 4. ข้อบกพร่องในการควบคุมการเข้าถึงเข้าชม: การขาดความสามารถในการควบคุมการเข้าถึงและการเข้าชมไฟล์บันทึกเหตุการณ์ ซึ่งอาจทำให้ผู้ไม่ประสงค์ดีสามารถเข้าถึงและแก้ไขข้อมูลบันทึกเหตุการณ์ได้ 5. ข้อบกพร่องในการจัดเก็บข้อมูลเหตุการณ์: การบันทึกข้อมูลเหตุการณ์ที่ไม่เพียงพอหรือไม่ครอบคลุม ซึ่งอาจทำให้ไม่สามารถนำข้อมูลเหตุการณ์มาวิเคราะห์หรือใช้ในการตรวจสอบความมั่นคงของระบบได้อย่างมีประสิทธิภาพ การแก้ไขปัญหาด้านความสามารถในการบันทึกและตรวจสอบความปลอดภัยของระบบ รวมถึงการใช้เทคโนโลยีและการสร้างนโยบายที่เหมาะสม เป็นสิ่งสำคัญในการป้องกันการโจมตีและรักษาความมั่นคงของระบบไว้อย่างต่อเนื่อง
A10: Server-Side Request Forgery	ความเสี่ยงที่เกี่ยวข้องกับการโจมตีที่ผู้ไม่ประสงค์ดีสร้างคำขอ (request) ที่ส่งถึงเซิร์ฟเวอร์แล้วเกิดการประมวลผลคำขอนั้นในฝั่งเซิร์ฟเวอร์ ซึ่งอาจทำให้เกิดผลกระทบอันไม่พึงประสงค์ รวมถึงการเข้าถึงข้อมูลภายในระบบหรือระบบที่อยู่ในเครือข่ายภายใน หรือการโจมตีอื่น ๆ ที่เกี่ยวข้องกับการร้องขอจากฝั่งเซิร์ฟเวอร์ รายละเอียดการทำงานของ A10 อาจรวมถึงการโจมตีด้วยเทคนิคเช่นการส่งคำขอ HTTP ที่มี URL ที่ชี้ไปยังทรัพยากร์ในเครือข่ายภายใน หรือการใช้โปรโตคอลที่เรียกหรือส่งคำขอเพื่อทำงานภายในเครื่องเซิร์ฟเวอร์ที่ไม่ควรเข้าถึงได้ผ่านภายนอก การป้องกัน A10 สามารถทำได้โดยการตรวจสอบและกรองคำขอที่ส่งถึงเซิร์ฟเวอร์ และการใช้การกำหนดขีดจำกัดในการเข้าถึงทรัพยากรระดับระบบที่สูงขึ้น รวมทั้งการใช้การตรวจสอบและสร้างบันทึกเหตุการณ์เพื่อตรวจจับการโจมตีด้วยวิธีนี้ได้อย่างมีประสิทธิภาพ

3.2.2 CWE Top 25

ตารางที่ : 7 ความหมายของหัวข้อที่เกี่ยวข้องกับ Common Weakness enumuration

หัวข้อ CWE Top 25	คำอธิบาย
CWE-79: Cross-site Scripting (XSS)	ช่องโหว่ที่เกิดจากการไม่ตรวจสอบข้อมูลที่รับเข้ามาก่อนนำไปแสดงผลบนหน้าเว็บ ทำให้ผู้โจมตีสามารถแทรก script ที่เป็นอันตรายได้
CWE-787: Out-of-bounds Write	การเขียนข้อมูลเกินขอบเขตของ buffer ที่จัดสรรไว้ อาจทำให้เกิดการทำงานผิดพลาดหรือถูกใช้ในการโจมตีระบบ
CWE-89: SQL Injection	ช่องโหว่ที่เกิดจากการไม่ตรวจสอบข้อมูลที่รับเข้ามาก่อนนำไปใช้ใน SQL command ทำให้ผู้โจมตีสามารถแก้ไขคำสั่ง SQL ได้
CWE-352: Cross-Site Request Forgery (CSRF)	การโจมตีที่หลอกให้ผู้ใช้ที่ login อยู่ทำคำสั่งที่ไม่ได้ตั้งใจโดยไม่รู้ตัว
CWE-22: Path Traversal	ช่องโหว่ที่อนุญาตให้ผู้โจมตีเข้าถึงไฟล์หรือไดเรกทอรีนอกเหนือจากที่กำหนดไว้
CWE-125: Out-of-bounds Read	การอ่านข้อมูลเกินขอบเขตของ buffer ที่จัดสรรไว้ อาจทำให้เกิดการรั่วไหลของข้อมูลหรือระบบล่ม
CWE-78: OS Command Injection	ช่องโหว่ที่ทำให้ผู้โจมตีสามารถแทรกคำสั่งของระบบปฏิบัติการได้ผ่านทางแอปพลิเคชัน
CWE-416: Use After Free	การใช้งาน memory ที่ถูก free ไปแล้ว อาจทำให้เกิดพฤติกรรมที่ไม่คาดคิดหรือถูกใช้ในการโจมตี
CWE-862: Missing Authorization	การขาดการตรวจสอบสิทธิ์ในการเข้าถึงทรัพยากรหรือฟังก์ชัน
CWE-434: Unrestricted Upload of File with Dangerous Type	การอนุญาตให้อัปโหลดไฟล์ที่เป็นอันตรายโดยไม่มีการตรวจสอบประเภทไฟล์
CWE-94: Code Injection	ช่องโหว่ที่ทำให้ผู้โจมตีสามารถแทรกและรันโค้ดที่เป็นอันตรายได้
CWE-20: Improper Input Validation	การไม่ตรวจสอบความถูกต้องของข้อมูลที่รับเข้ามา
CWE-77: Command Injection	การแทรกคำสั่งที่เป็นอันตรายผ่านทาง input ของแอปพลิเคชัน

หัวข้อ CWE Top 25	คำอธิบาย
CWE-287: Improper Authentication	การตรวจสอบตัวตนที่ไม่เหมาะสมหรือไม่เพียงพอ
CWE-269: Improper Privilege Management	การจัดการสิทธิ์ที่ไม่เหมาะสม อาจทำให้ผู้ใช้มีสิทธิ์มากเกินไป
CWE-502: Deserialization of Untrusted Data	การแปลงข้อมูลที่ไม่น่าเชื่อถือกลับเป็น object อาจนำไปสู่การรันโค้ดที่เป็นอันตราย
CWE-200: Exposure of Sensitive Information	การเปิดเผยข้อมูลที่ละเอียดอ่อนให้กับผู้ที่ไม่มีสิทธิ์
CWE-863: Incorrect Authorization	การตรวจสอบสิทธิ์ที่ไม่ถูกต้อง
CWE-918: Server-Side Request Forgery (SSRF)	ช่องโหว่ที่ทำให้ผู้โจมตีสามารถบังคับให้ server ส่งคำขอไปยังปลายทางที่ไม่ได้ตั้งใจ
CWE-119: Improper Restriction of Operations within Memory Buffer	การจำกัดการทำงานภายใน memory buffer ที่ไม่เหมาะสม
CWE-476: NULL Pointer Dereference	การเข้าถึง pointer ที่มีค่าเป็น NULL ทำให้เกิด crash หรือพฤติกรรมที่ไม่คาดคิด
CWE-798: Use of Hard-coded Credentials	การใช้ username และ password ที่ฝังตายอยู่ในโค้ด
CWE-190: Integer Overflow or Wraparound	การล้นของตัวเลขจำนวนเต็มที่อาจนำไปสู่พฤติกรรมที่ไม่คาดคิด
CWE-400: Uncontrolled Resource Consumption	การใช้ทรัพยากรที่ไม่มีการควบคุม อาจทำให้เกิด Denial of Service
CWE-306: Missing Authentication for Critical Function	การขาดการตรวจสอบตัวตนสำหรับฟังก์ชันที่สำคัญ

3.2.3 CVSS Score

CVSS (Common Vulnerability Scoring System) คือระบบคะแนนมาตรฐานสำหรับประเมินความรุนแรงของช่องโหว่ โดยมีช่วงคะแนนดังนี้

None: 0.0

Low: 0.1 - 3.9

Medium: 4.0 - 6.9

High: 7.0 - 8.9

Critical: 9.0 - 10.0

หมายเหตุ: CWE เป็นการจัดหมวดหมู่ประเภทของช่องโหว่ ไม่ใช่ช่องโหว่เฉพาะเจาะจง ดังนั้นจึงไม่มีคะแนน CVSS ที่ตายตัว แต่จะมีช่วงคะแนนที่พบบ่อยในแต่ละประเภท โดยคะแนน CVSS ที่แท้จริงจะถูกกำหนดให้กับ CVE (ช่องโหว่เฉพาะ) แต่ละตัว

3.3 เมทริกซ์ของผลกระทบและความน่าจะเป็น (Impact and Likelihood Matrix)

มาตรฐาน OWASP Risk Rating Methodology standards (https://owasp.org/www-community/OWASP_Risk_Rating_Methodology) ถูกใช้ในการพิจารณาและประเมินความน่าจะเป็นของเหตุการณ์ที่เป็นอันตรายต่อระบบความปลอดภัย รวมถึงผลกระทบที่อาจเกิดขึ้นจากเหตุการณ์เหล่านั้น ที่เกิดจากการโจมตีหรือการละเมิดความปลอดภัยของระบบข้อมูล การวิเคราะห์ความน่าจะเป็นและผลกระทบมีเกณฑ์การให้คะแนนที่อธิบายโดยละเอียดดังต่อไปนี้

3.3.1 ปัจจัยที่ใช้ประเมินผลกระทบ (Impact Factors)

การวิเคราะห์ผลกระทบของข้อความจะถูกดำเนินการทั้งในด้าน "เชิงเทคนิค" และ "เชิงธุรกิจ" การให้คะแนนผลกระทบมีดังนี้

ตารางที่ : 8 คะแนนของผลกระทบในด้าน "เชิงเทคนิค" และ "เชิงธุรกิจ"

เกณฑ์ผลกระทบ	นิยามคะแนน
1. ปัจจัยของผลกระทบเชิงเทคนิค (Technical Impact Factor)	
การสูญเสียข้อมูลที่เป็นความลับ (Loss of Confidentiality)	2: ข้อมูลที่ไม่สำคัญเล็กน้อยถูกเปิดเผย 6: ข้อมูลที่สำคัญเล็กน้อยหรือข้อมูลที่ไม่สำคัญมากถูกเปิดเผย 7: ข้อมูลที่สำคัญมากถูกเปิดเผย 9: ข้อมูลทั้งหมดถูกเปิดเผย
การสูญเสียความสมบูรณ์หรือความน่าเชื่อถือของข้อมูล (Loss of Integrity)	1: ข้อมูลเสียหายเพียงเล็กน้อย 3: ข้อมูลเสียหายอย่างรุนแรงเพียงเล็กน้อย 5: ข้อมูลเสียหายเพียงเล็กน้อยอย่างกว้างขวาง 7: ข้อมูลเสียหายอย่างรุนแรงอย่างกว้างขวาง 9: ข้อมูลทั้งหมดเสียหายทั้งหมด
การสูญเสียความพร้อมใช้งาน (Loss of Availability)	1: Services ถูกหยุดชั่วคราวเพียงเล็กน้อย 5: Services ถูกหยุดชั่วคราวเพียงเล็กน้อย หรือ Services ถูกหยุด

เกณฑ์ผลกระทบ	นิยามคะแนน
	ชั่วคราวอย่างกว้างขวาง 7: Services ถูกหยุดชั่วคราวอย่างกว้างขวาง 9: ทุก Services สูญเสียไปอย่างสมบูรณ์
การสูญเสียความรับผิดชอบ (Loss of Accountability)	1: สามารถติดตามตัวได้ทั้งหมด 7: อาจจะสามารถติดตามตัวได้ 9: ไม่สามารถระบุตัวตนหรือติดตามตัวได้
2. ปัจจัยของผลกระทบเชิงธุรกิจ (Business Impact Factors)	
ความเสียหายทางการเงิน (Financial Damage)	1: มีน้อยกว่าค่าซ่อมแซมช่องโหว่ 3: มีผลกระทบต่อกำไรประจำปีเล็กน้อย 7: มีผลกระทบต่อกำไรประจำปีอย่างมีนัยสำคัญ 9: ล้มละลายการเงิน
ความเสียหายที่เกิดขึ้นกับชื่อเสียง หรือภาพลักษณ์ (Reputation Damage)	1: มีความเสียหายเล็กน้อย 4: สูญเสียลูกค้ารายสำคัญ 5: สูญเสียความน่าเชื่อถือ 9: มีความเสียหายต่อชื่อเสียงหรือแบรนด์
การไม่ปฏิบัติตามกฎหมาย หรือ นโยบาย หรือมาตรฐานที่กำหนดไว้ (Non-compliance)	2: การละเมิดเพียงเล็กน้อย 5: การละเมิดที่ชัดเจน 7: การละเมิดระดับสูง
การละเมิดความเป็นส่วนตัวของ บุคคลหรือข้อมูลส่วนบุคคล (Privacy Violation)	3: หนึ่งบุคคล 5: หลักร้อยคน 7: หลักพันคน 9: หลักล้านคน

3.3.2 ปัจจัยที่ใช้ประเมินความน่าจะเป็น (Likelihood Factors)

ความน่าจะเป็นประกอบด้วย ปัจจัยของ "ตัวแทนผู้ลักลอบ" และ "ช่องโหว่" การจัดระดับความน่าจะเป็นมีดังนี้

ตารางที่ : 9 คะแนนปัจจัยของ "ตัวแทนผู้ลักลอบ" และ "ช่องโหว่" ที่ใช้ประเมินความน่าจะเป็น

เกณฑ์ความน่าจะเป็น	นิยามคะแนน
1. ปัจจัยของตัวแทนผู้ลักลอบ (Threat Agent Factors)	
ระดับทักษะ (Skill Level)	1: ไม่มีทักษะทางเทคนิค 3: บางทักษะทางเทคนิค 5: ผู้ใช้คอมพิวเตอร์ระดับขั้นสูง 6: ทักษะในด้านเครือข่ายและโปรแกรมมิ่ง 9: ทักษะการเจาะระบบความปลอดภัย
เหตุผลหรือจุดมุ่งหมาย (Motive)	1: ต้องการรางวัลเล็กน้อยหรือไม่ต้องการ 4: ต้องการรางวัลที่เป็นไปได้ 9: ต้องการรางวัลขั้นสูง
โอกาสหรือสถานการณ์ที่เอื้อต่อตัวแทน (Opportunity)	0: ต้องการการเข้าถึงทั้งหมดหรือทรัพยากรที่มีค่าสูง 4: ต้องการการเข้าถึงพิเศษหรือทรัพยากรที่มีค่า 7: ต้องการการเข้าถึงบางส่วนหรือทรัพยากรบางประเภท 9: ไม่ต้องการการเข้าถึงหรือทรัพยากรใด
ขนาดของกลุ่มผู้ใช้หรือผู้เกี่ยวข้องที่เป็นเป้าหมายของการโจมตีหรือการทำลายระบบ (Size)	2: นักพัฒนาหรือผู้ดูแลระบบ 4: ผู้ใช้ภายในอินทราเน็ต 5: พาร์ทเนอร์ 6: ผู้ใช้ที่ได้รับการตรวจสอบสิทธิ์ 9: ผู้ใช้อินเทอร์เน็ตที่ไม่ระบุชื่อ
2. ปัจจัยของช่องโหว่ (Vulnerability Factors)	
ความง่ายในการค้นพบช่องโหว่ (Ease of discovery)	1: เกือบจะเป็นไปไม่ได้หรือมีความยากมากที่จะทำได้ในสภาวะปกติ 3: ยาก

เกณฑ์ความน่าจะเป็น	นิยามคะแนน
	7: ง่าย 9: มีเครื่องมือแบบอัตโนมัติที่ใช้ทำได้
ความง่ายในการใช้ช่องโหว่หรือข้อบกพร่องเพื่อโจมตีระบบ (Ease of exploit)	1: เป็นเพียงทฤษฎี 3: ยาก 5: ง่าย 9: มีเครื่องมือแบบอัตโนมัติที่ใช้ทำได้
ความตระหนักในเรื่องความปลอดภัยของข้อมูล (Awareness)	1: ไม่รู้ 4: ไม่เปิดเผย 6: ชัดเจน 9: รู้กันทั่วไป
การตรวจจับการบุกรุก หรือตรวจจับการกระทำที่เป็นภัยต่อระบบขององค์กร (Intrusion detection)	1: ตรวจจับเหตุการณ์ที่เกิดขึ้นทันทีในแอปพลิเคชัน 3: บันทึกและตรวจสอบ 8: บันทึกโดยไม่ตรวจสอบ 9: ไม่ได้ทำการบันทึก

3.4 ระดับการจัดอันดับความเสี่ยง (Risk rating level)

3.4.1 ความรุนแรงของความเสี่ยง (The Severity of the risk)

การประเมินความน่าจะเป็นและการประเมินผลกระทบจะประเมินร่วมกันเพื่อคำนวณความรุนแรงโดยรวมของความเสี่ยงนี้ การทำเช่นนี้จะทำโดยการกำหนดว่าความน่าจะเป็นมีระดับต่ำ กลาง หรือสูง และทำเช่นเดียวกันสำหรับผลกระทบด้วย เกณฑ์การให้คะแนนระดับ 0 ถึง 9 ถูกแบ่งออกเป็นสามส่วนดังนี้

ระดับความน่าจะเป็นและผลกระทบ	
0 ถึง < 3	ต่ำ (Low)
3 ถึง < 6	กลาง (Medium)
6 ถึง 9	สูง (High)

การให้คะแนนความเสี่ยงถูกคำนวณโดยการคูณผลกระทบและความน่าจะเป็นของอุปสรรคต่าง ๆ คะแนนจะถูกกำหนดตามเกณฑ์ดังนี้:

Impact	Likelihood: Low	Likelihood: Medium	Likelihood: High
High	Medium	High	Critical
Medium	Low	Medium	High
Low	Information	Low	Medium

คำจำกัดความของการจัดอันดับความเสี่ยง (Risk Rating Definitions)

ระดับความเสี่ยง	คำอธิบาย
สูงมาก (Critical)	ระดับความรุนแรง "สูงมาก" เป็นระดับที่ยอมรับไม่ได้และควรพิจารณาการแก้ไขปัญหาโดยทันที เพื่อให้ความเสี่ยงอยู่ในระดับที่ยอมรับได้ตามนโยบายและมาตรการด้านความปลอดภัยของข้อมูลขององค์กร และเพื่อลดความเสี่ยงด้านความปลอดภัยของข้อมูลรวมถึงความลับ ความสมบูรณ์ และความพร้อมใช้งาน รวมถึงความรับผิดชอบในการบรรลุวัตถุประสงค์ขององค์กร การดำเนินงาน การเงิน ชื่อเสียง ความเป็นไปตามข้อกำหนด และความเป็นส่วนตัวของบุคคล.
สูง (High)	ระดับความรุนแรง "สูง" เป็นระดับที่ยอมรับไม่ได้และควรพิจารณาการแก้ไขข้อบกพร่องโดยนำมาใช้ในทางปฏิบัติภายในระยะเวลาที่เหมาะสม และจัดการความเสี่ยงให้อยู่ในระดับที่ยอมรับได้ตามนโยบายและมาตรการด้านความปลอดภัยของข้อมูลขององค์กร เพื่อลดความเสี่ยงด้านความปลอดภัยของข้อมูลรวมถึงความลับ ความถูกต้อง และความพร้อมใช้งาน รวมถึงความรับผิดชอบในการบรรลุวัตถุประสงค์ขององค์กร การดำเนินงาน การเงิน ชื่อเสียง ความเป็นไปตามข้อกำหนด และความเป็นส่วนตัวของบุคคล.
ปานกลาง (Medium)	ระดับความรุนแรง "ปานกลาง" ควรพิจารณาการแก้ไขข้อบกพร่องและนำมาใช้ในทางปฏิบัติภายในระยะเวลาที่เหมาะสมตามนโยบายและมาตรการด้านความปลอดภัยของข้อมูลขององค์กรเพื่อลดความเสี่ยงด้านความปลอดภัยของข้อมูลรวมถึงความลับ ความถูกต้อง และความพร้อมใช้งาน รวมถึงความรับผิดชอบในการบรรลุวัตถุประสงค์ขององค์กร การดำเนินงาน การเงิน ชื่อเสียง ความเป็นไปตามข้อกำหนด และความเป็นส่วนตัวของบุคคล อย่างไรก็ตาม การยอมรับความเสี่ยงอาจถูกพิจารณา และความเสี่ยงควรถูกควบคุมเพื่อป้องกันไม่ให้เกิดการเพิ่มขึ้นไปสู่ระดับที่ยอมรับไม่ได้
ต่ำ (Low)	ระดับความรุนแรง "ต่ำ" ควรพิจารณาว่าการทำการแก้ไขนั้นยังจำเป็นหรือไม่ หรือตัดสินใจที่จะยอมรับความเสี่ยงตามนโยบายและมาตรการด้านความปลอดภัยของข้อมูลขององค์กรเพื่อลดความเสี่ยงด้านความปลอดภัยของข้อมูลรวมถึงความลับ ความถูกต้อง และความพร้อมใช้งาน รวมถึงความรับผิดชอบในการบรรลุวัตถุประสงค์ขององค์กร การดำเนินงาน การเงิน ชื่อเสียง ความเป็นไปตามข้อกำหนด และความเป็นส่วนตัวของบุคคล
เสนอแนะ (Information)	"เสนอแนะ" อาจมีการพัฒนาเพิ่มเติมเพื่อเสริมความแข็งแกร่งของการป้องกันสินทรัพย์ข้อมูลขององค์กรต่อไป ทั้งนี้ขึ้นอยู่กับการวิเคราะห์ขององค์กรเกี่ยวกับปัจจัยที่มีผลต่อความสมเหตุสมผลของต้นทุนสำหรับการกระทำการแก้ไขต่อไป

บทที่ 4 สรุปช่องโหว่ที่ตรวจพบ

หัวข้อนี้ได้ถูกจัดทำขึ้นเพื่อบรรยายสรุปช่องโหว่ที่ตรวจพบและความเสี่ยงที่เกี่ยวข้องในระหว่างการการทดสอบเจาะระบบ ซึ่งมีดังต่อไปนี้

ตารางที่ : 10 สรุปช่องโหว่ที่ตรวจพบจากการเจาะระบบและความเสี่ยงที่เกี่ยวข้อง

ID	Vulnerabilities name	Risk	Fixing Status	Remark
การทดสอบประเภท PENETRATION TEST แบบ GREY-BOX				
1	ตรวจพบว่า JSON Parameter ไม่ตรวจสอบ payload injection	**ปานกลาง**	ปิดโดยการแก้ไข	
2	ระบบไม่มี Session Timeout หลังจากมีการ Logout จากระบบ	**ปานกลาง**	ปิดโดยการแก้ไข	
3	สามารถเข้าถึงข้อมูลได้โดยไม่ต้องมี Access Token	**ปานกลาง**	ปิดโดยการแก้ไข	
4	ตรวจพบมีการใช้งาน Session Token บน URL Path	**ต่ำ (Low)**	ปิดโดยการแก้ไข	
5	ตรวจพบ Secret/Credential Key ใน source code	**ต่ำ (Low)**	ปิดโดยการแก้ไข	
การทดสอบประเภท PENETRATION TEST แบบ BLACK-BOX				
6	ตรวจพบการใช้งาน Software ที่มีช่องโหว่	**ต่ำ (Low)**	ปิดโดยการแก้ไข	
7	ตรวจพบการตั้งค่าไม่ปลอดภัยบน Software Header	**บันทึกไว้ (Notes)**	ปิดโดยการแก้ไข	

บทที่ 5 รายละเอียดของช่องโหว่จากการทดสอบเจาะระบบ

หัวข้อนี้ถูกออกแบบขึ้นเพื่อให้มีข้อมูลที่เป็นรายละเอียดเกี่ยวกับผลการตรวจพบพร้อมกับคำแนะนำในการแก้ไข เพื่อเสริมสร้างความปลอดภัยของระบบ

5.1 ตรวจพบว่า JSON Parameter ไม่ตรวจสอบ payload injection

ผลการทดสอบซ้ำ

วิธีการแก้ไข	ระดับความเสี่ยง	สถานะช่องโหว่	หมายเหตุ
ตรวจพบว่า JSON Parameter มีการตรวจสอบ payload injection	ปานกลาง	ปิดโดยการแก้ไข	

ประเด็นข้อตรวจพบและผลกระทบ

การที่ระบบไม่ตรวจสอบ Payload Injection ในพารามิเตอร์ (Parameter) ใด ๆ เป็นช่องโหว่ประเภท Input Validation ที่มีการตรวจสอบที่ล้มเหลว ซึ่งสามารถนำไปสู่การโจมตีได้หลายรูปแบบเช่น XSS , SQLi เป็นต้น ซึ่งตัวระบบยังคงมีการป้องกัน Special Character ได้บางตัวอักษรอยู่บ้าง แต่ยังไม่สมบูรณ์

Attacker สามารถอ่าน, แก้ไข, หรือลบข้อมูล ทั้งหมด ในฐานข้อมูลได้ หากมีการใช้งาน Payload Injection ไปจนกว่าจะมีข้อมูลสำคัญหรือผลลัพธ์ที่ต้องการ

ที่	เป้าหมายที่กระทบ
(1)	• 'https://e2e-m4-testapp.app.baac.or.th/BAACLegalService/rest/internal/app/legal/req_case_request_group_list_to • https://e2e-m4-testapp.app.baac.or.th/BAACLegalService/rest/internal/app/legal/req_case_request_group_contract_704_list_to • https://e2e-m4-testapp.app.baac.or.th/BAACLegalService/rest/internal/app/legal/req_case_request_group_contract_write_off_list_706_to • https://e2e-m4-testapp.app.baac.or.th/BAACCoreService/rest/internal/app/auth/auth_session_manage_update • https://e2e-m4-testapp.app.baac.or.th/BAACLegalService/rest/internal/app/legal/legal_lawyer_name_list_to

ระดับความเสี่ยง

ระดับความเสี่ยง	ปานกลาง	**ความน่าจะเป็น**	ต่ำ
		ผลกระทบ	สูง
อ้างอิง	A03:2021 CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H (10.0 Critical) CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:H/SI:H/SA:H (10.0 Critical)		
คำอธิบายเพิ่มเติม	ระบบ SAP นี้มีการเข้าใช้งานยากในการเข้าถึงระบบ เนื่องจากต้องติดตั้งซอฟต์แวร์ในการต่อ VPN ที่มีการยืนยันตัวตนด้วย 2FA และต้องผ่านการเปิดใช้งานหน้า Web Console ที่เป็นแบบ captive portal บนเครื่องคอมพิวเตอร์หรือ Client ก่อน ทำให้นักทดสอบเจาะระบบ (Pentester) มองว่ามี likelihood (ความเป็นไปได้) ในการถูกโจมตีหลังจากที่พิจารณาองค์ประกอบทั้งหมดแล้วอยู่ในระดับต่ำเนื่องจากเหตุผลดังนี้: 1) ความซับซ้อนในการเข้าถึง ● การที่ต้องติดตั้ง VPN และซอฟต์แวร์ในการทำ 2FA เพิ่มความซับซ้อนในการเข้าถึงระบบ ทำให้ผู้โจมตีต้องใช้เวลาและความพยายามมากขึ้นในการเข้าถึงระบบ 2) การใช้ VPN ● การใช้VPN ช่วยเพิ่มความปลอดภัยในการสื่อสารระหว่างผู้ใช้และเซิร์ฟเวอร์ ทำให้การโจมตีแบบ Man-in-the-Middle (MitM) ยากขึ้นอย่างมาก		

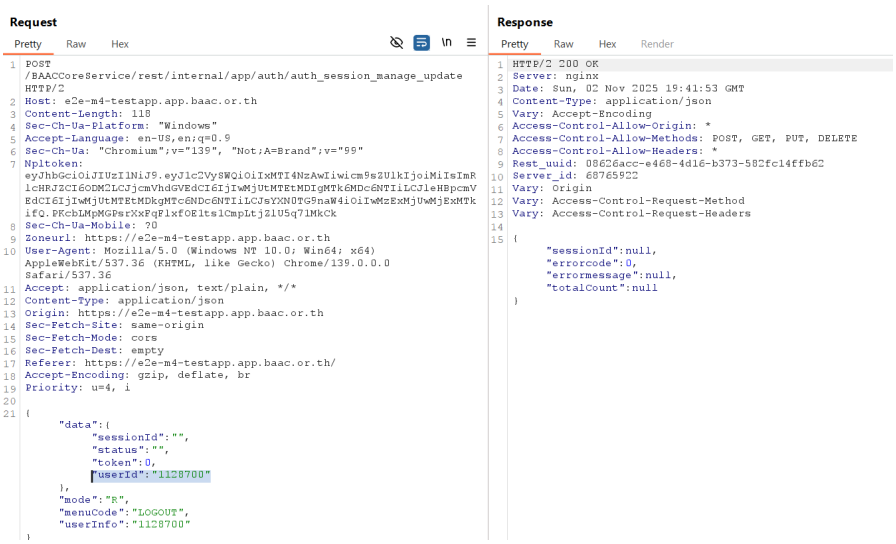
คำแนะนำในการแก้ไข

คำอธิบาย
หลักการป้องกันช่องโหว่ Injection คือ ไม่เชื่อถือข้อมูลนำเข้าจากผู้ใช้ (Never Trust User Input) ดังนี้: 1. **Input Validation (การตรวจสอบความถูกต้อง):** - Whitelist: อนุญาตให้รับเฉพาะรูปแบบ (Format) ที่ถูกต้องและจำเป็นเท่านั้น (เช่น ถ้าคาดหวังตัวเลข ต้องตรวจสอบว่าเป็นตัวเลขจริง ๆ, ถ้าคาดหวังชื่อ ต้องอนุญาตเฉพาะตัวอักษร) - Sanitization: ลบอักขระพิเศษหรือโค้ดอันตรายออกไป

คำอธิบาย
2. Output Encoding / Escaping (การเข้ารหัสข้อมูลแสดงผล): - ก่อนนำข้อมูลจากพารามิเตอร์ไปแสดงผลบนหน้าเว็บ ต้องเข้ารหัส (Encode) อักขระพิเศษ (เช่น < เป็น <) เพื่อให้เบราว์เซอร์มองเห็นเป็นข้อความธรรมดา ไม่ใช่ โค้ด HTML กรณีใช้ Nginx สามารถใช้งาน Module ModSecurity ทำการตรวจสอบเนื้อหาของ HTTP Request (เช่น URL, Query Parameters, Body) เทียบกับชุดกฎ (Rule Sets) ที่ถูกออกแบบมาเพื่อตรวจจับรูปแบบของ Payload Injection ที่เป็นอันตราย หรือใช้งานระบบป้องกันอย่าง Web Application Firewall (WAF)/Web Application and API Protection (WAAP)

ตัวอย่างผลของการทดสอบ

1) ทดสอบการ POST ข้อมูลผ่าน auth_session_manage_update



Request

Pretty Raw Hex

```
POST /BAACCoreService/rest/internal/app/auth/auth_session_manage_update HTTP/2
Host: e2e-m4-testapp.app.baac.or.th
Content-Length: 118
Sec-Ch-Ua-Platform: "Windows"
Accept-Language: en-US,en;q=0.9
Sec-Ch-Ua: "Chromium";v="139", "Not;A=Brand";v="99"
Npltoken:
eyJhbGciOiJIUzI1NiJ9.eyJlc2VyS1dQiOiIxMTI4NzAwIiwicm9sZUlkIjoiMiIsImRlcHRJZCI6ODM2LCJjcmVhdGVEdCI6IjIwMjUtMTEtMDIgMTk6MDc6NTIiLCJleHBpcmVEdCI6IjIwMjUtMTEtMDkgMTc6NDc6NTIiLCJsYXN0TG9naW4iOiIwMzExMjUwMjExMTkifQ.PKcbLMpMGPsrXxPqPlxfOEltslCmpLtj2lU5q7lMkCk
Sec-Ch-Ua-Mobile: ?0
Zoneurl: https://e2e-m4-testapp.app.baac.or.th
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/139.0.0.0 Safari/537.36
Accept: application/json, text/plain, */*
Content-Type: application/json
Origin: https://e2e-m4-testapp.app.baac.or.th
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://e2e-m4-testapp.app.baac.or.th/
Accept-Encoding: gzip, deflate, br
Priority: u=4, i

{
    "data":{
        "sessionId":"",
        "status":"",
        "token":0,
        "userId":"1128700"
    },
    "mode":"R",
    "menuCode":"LOGOUT",
    "userInfo":"1128700"
}
```

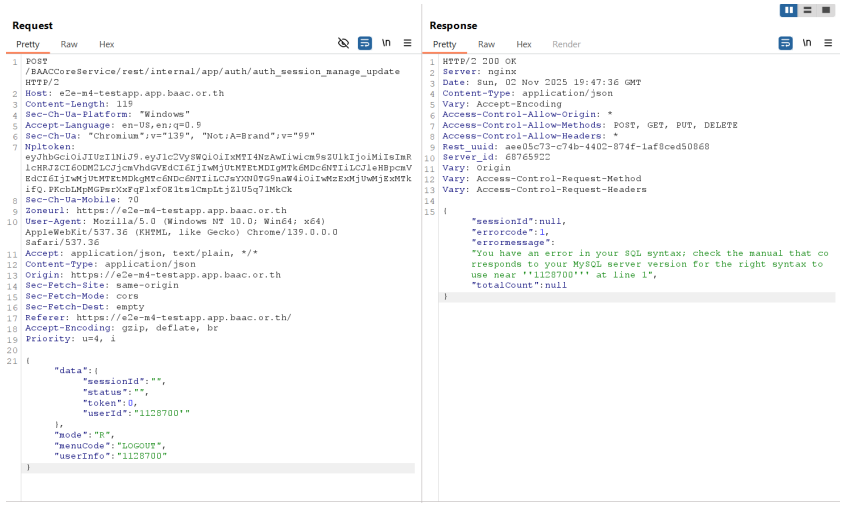
Response

Pretty Raw Hex Render

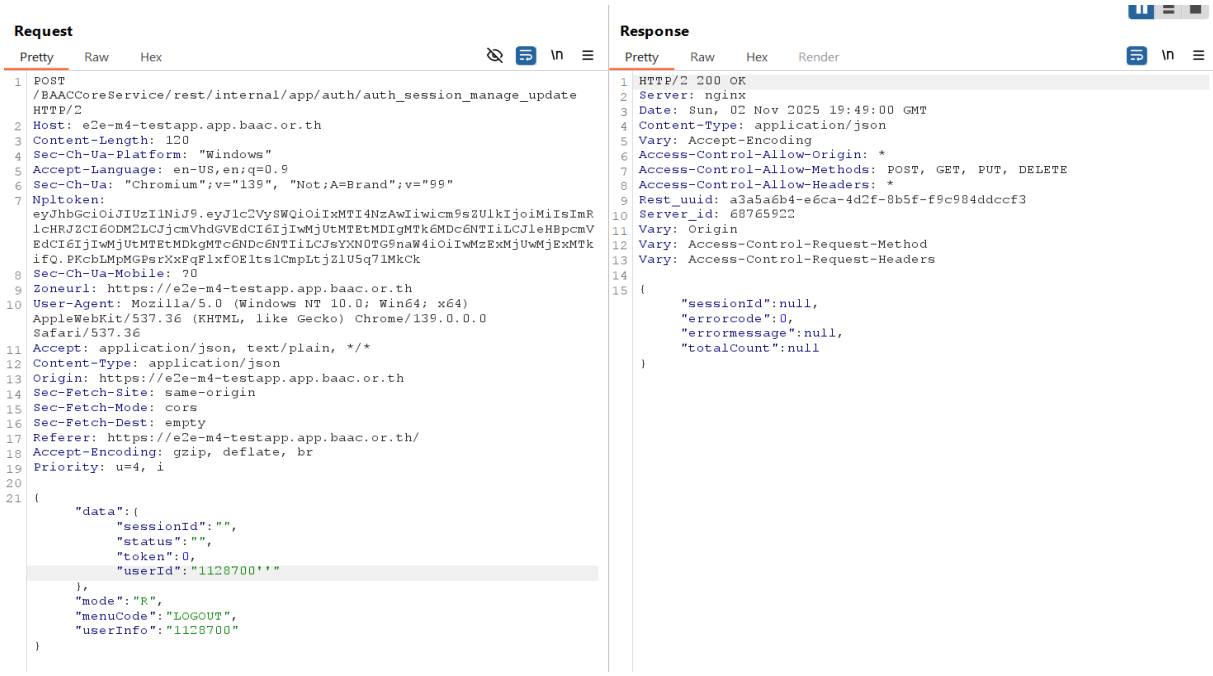
```
HTTP/2 200 OK
Server: nginx
Date: Sun, 02 Nov 2025 19:41:53 GMT
Content-Type: application/json
Vary: Accept-Encoding
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: POST, GET, PUT, DELETE
Access-Control-Allow-Headers: *
Rest_uuid: 08626acc-e468-4d16-b373-582fc14ffb62
Server_id: 68765922
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers

{
    "sessionId":null,
    "errorcode":0,
    "errormessage":null,
    "totalCount":null
}
```

2) ทดสอบเพิ่ม Payload ใน Parameter: userId



3) ทดสอบปรับเพิ่ม Payload ใน Parameter: userId



4) ทดสอบปรับเปลี่ยน Payload ใน Parameter: userId

Request

Pretty Raw Hex

```
POST /BAACCoreService/rest/internal/app/auth/auth_session_manage_update HTTP/2
Host: e2e-m4-testapp.app.baac.or.th
Content-Length: 148
Sec-Ch-Ua-Platform: "Windows"
Accept-Language: en-US,en;q=0.9
Sec-Ch-Ua: "Chromium";v="139", "Not;A=Brand";v="99"
Npltoken:
eyJhbGciOiJIUzI1NiJ9.eyJlc2VySWQiOiIxMTI4NzAwIiwicm9sZUlkIjoiMiIsImRlcHRJZCI6ODM2LCJjcmVhdGVEdCI6IjIwMjUtMTEtMDIgMTk6MDc6NTIiLCJleHBpcmVEdCI6IjIwMjUtMTEtMDkgMTc6NDc6NTIiLCJsYXN0TG9naW4iOiIwMzExMjUwMjExMTkifQ.PKcbLMpMGPsrXxFqFlxfOE1ts1CmpLtjZlU5q71MkCk
Sec-Ch-Ua-Mobile: ?0
Zoneurl: https://e2e-m4-testapp.app.baac.or.th
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/139.0.0.0 Safari/537.36
Accept: application/json, text/plain, */*
Content-Type: application/json
Origin: https://e2e-m4-testapp.app.baac.or.th
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://e2e-m4-testapp.app.baac.or.th/
Accept-Encoding: gzip, deflate, br
Priority: u=4, i

{
    "data":{
        "sessionId":"",
        "status":"",
        "token":0,
        "userId":"'+(select*from(select(sleep(20)))a)+'"
    },
    "mode":"R",
    "menuCode":"LOGOUT",
    "userInfo":"1128700"
}
```

Response

Pretty Raw Hex Render

```
HTTP/2 200 OK
Server: nginx
Date: Sun, 02 Nov 2025 19:51:06 GMT
Content-Type: application/json
Vary: Accept-Encoding
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: POST, GET, PUT, DELETE
Access-Control-Allow-Headers: *
Rest_uuid: 513d5946-b4b7-4058-a080-2610701e0802
Server_id: 68765922
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers

{
    "sessionId":null,
    "errorcode":0,
    "errormessage":null,
    "totalCount":null
}
```

5.2 ระบบไม่มี Session Timeout หลังจากมีการ Logout จากระบบ

ผลการทดสอบซ้ำ

วิธีการแก้ไข	ระดับความเสี่ยง	สถานะช่องโหว่	หมายเหตุ
มีการปรับให้แต่ละ Module ดำเนินการทำ Session timeout เพื่อให้ session หลุด	ปานกลาง	ปิดโดยการแก้ไข	

ประเด็นข้อตรวจพบและผลกระทบ

พบว่าหลังจากกด Logout แล้ว แต่ยังสามารถเข้าถึงหน้าเพจจากเมนูในระบบ EDMS และยังไม่ได้ใช้งานใน Tab เดียวกัน ซึ่งจะเป็นการ New Tab Page ขึ้นมาโดยไม่ต้อง Login ใหม่ ซึ่งเกิดจากกระบวนการ Logout ที่ไม่สมบูรณ์ (Incomplete Logout Process) ซึ่งถือเป็นช่องโหว่ความปลอดภัยในส่วนของ Session Management

Attacker สามารถใช้เครื่องมือดึง Session ID เก่าที่ค้างอยู่และนำ Session ID นั้นไปใช้สวมรอยเป็นผู้ใช้คนแรกได้ทันที เพราะ Server ยังมองว่า Session นี้ถูกต้อง (Session Hijacking) ซึ่งช่องโหว่นี้จะทำให้ Session ID ที่ขโมยมามีประโยชน์นานขึ้นมาก แม้ผู้ใช้จะ Logout ไปแล้ว

ที่	เป้าหมายที่กระทบ
(1)	'https://e2e-m1-testapp.app.baac.or.th/ https://e2e-m2-testapp.app.baac.or.th/ https://e2e-uat-m3ui.app.baac.or.th/ https://e2e-m4-testapp.app.baac.or.th/ https://uat-efiling.app.baac.or.th

ระดับความเสี่ยง

ระดับความเสี่ยง	ปานกลาง	ความน่าจะเป็น	ปานกลาง
		ผลกระทบ	ปานกลาง
อ้างอิง	A07:2021 CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:L (9.4 Critical) CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:L/SC:N/SI:N/SA:N (9.3		

	Critical)
คำอธิบายเพิ่มเติม	ระบบ SAP นี้มีการเข้าใช้งานยากในการเข้าถึงระบบ เนื่องจากต้องติดตั้งซอฟต์แวร์ในการต่อ VPN ที่มีการยืนยันตัวตนด้วย 2FA และต้องผ่านการเปิดใช้งานหน้า Web Console ที่เป็นแบบ captive portal บนเครื่องคอมพิวเตอร์หรือ Client ก่อน ทำให้นักทดสอบเจาะระบบ (Pentester) มองว่ามี likelihood (ความเป็นไปได้) ในการถูกโจมตีหลังจากที่พิจารณาองค์ประกอบทั้งหมดแล้วอยู่ในระดับต่ำเนื่องจากเหตุผลดังนี้: 1) ความซับซ้อนในการเข้าถึง ● การที่ต้องติดตั้ง VPN และซอฟต์แวร์ในการทำ 2FA เพิ่มความซับซ้อนในการเข้าถึงระบบ ทำให้ผู้โจมตีต้องใช้เวลาและความพยายามมากขึ้นในการเข้าถึงระบบ 2) การใช้ VPN ● การใช้VPN ช่วยเพิ่มความปลอดภัยในการสื่อสารระหว่างผู้ใช้และเซิร์ฟเวอร์ ทำให้การโจมตีแบบ Man-in-the-Middle (MitM) ยากขึ้นอย่างมาก

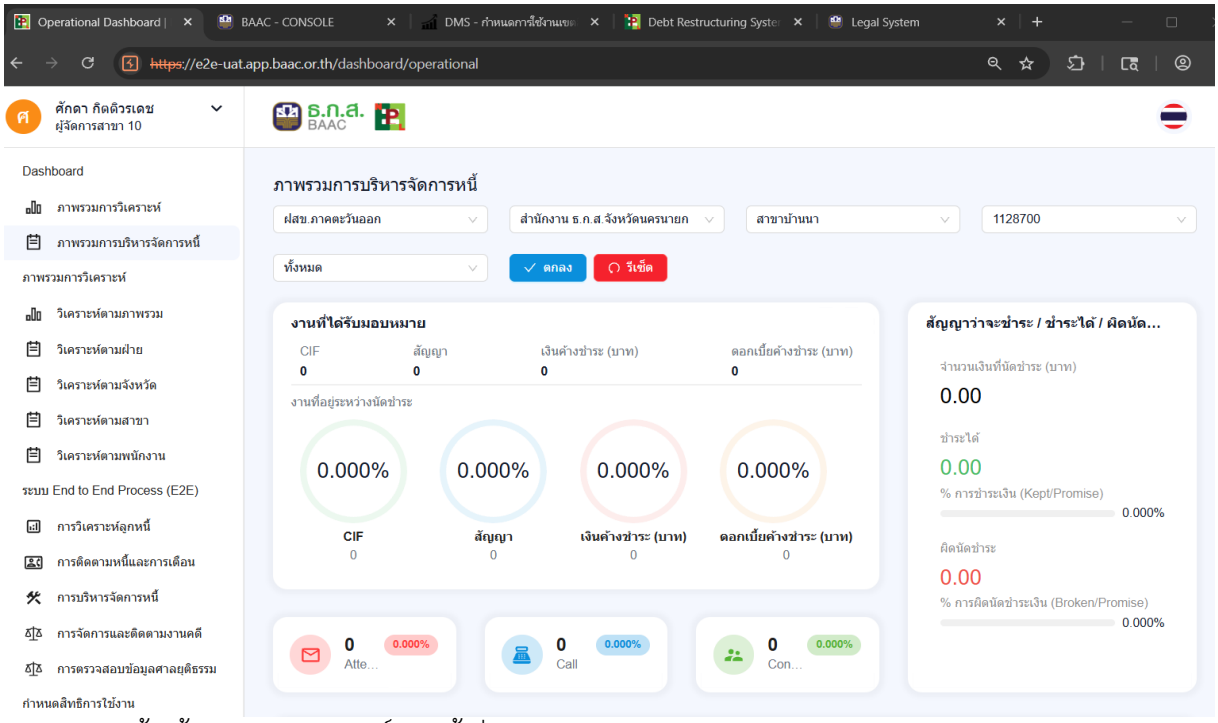
คำแนะนำในการแก้ไข

คำอธิบาย

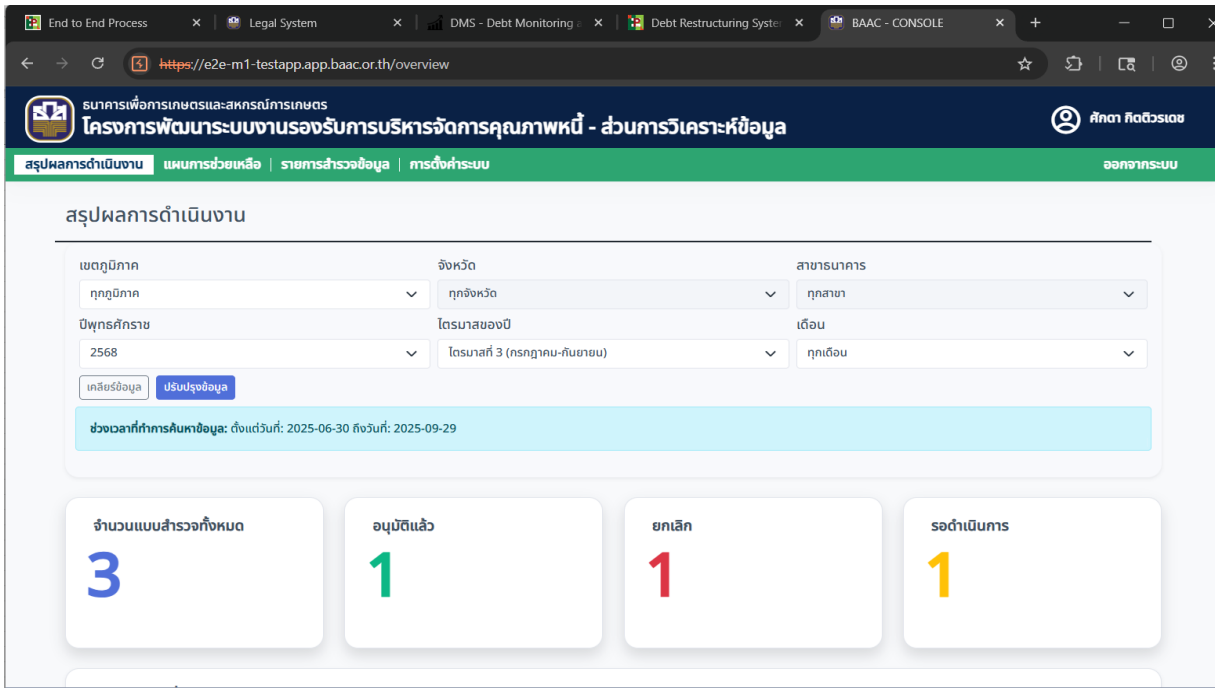
วิธีแก้ไขใน Nginx โดยการ เพิ่ม HTTP Headers เพื่อสั่งปิด Cache ใน location block
ปรับ location ~ ^/(admin|dashboard|profile) ให้ตรงกับ URL Path
location ~ ^/(admin|dashboard|profile) {
...
# --- ส่วนที่เพิ่มเข้ามา --- 	
# สั่งไม่ให้เบราว์เซอร์และ Proxy ใดๆ เก็บ Cache เลย 	
# no-store: คือคำสั่งที่แรงที่สุด คือห้ามเก็บลงดิสก์หรือเมมโมรี่เลย 	
# no-cache: บังคับให้ถามเซิร์ฟเวอร์ทุกครั้งก่อนใช้งาน 	
# must-revalidate: ต้องตรวจสอบความถูกต้องเสมอ 	
# private: อนุญาตให้แค่เบราว์เซอร์ส่วนตัวแคชได้ (แต่ no-store จะแรงกว่า) 	
add_header 'Cache-Control' 'no-store, no-cache, must-revalidate, private';
# สั่งให้หมดอายุทันที 	
add_header 'Expires' '0';
# --- จบส่วนที่เพิ่ม --- 	
...
}

ตัวอย่างผลของการทดสอบ

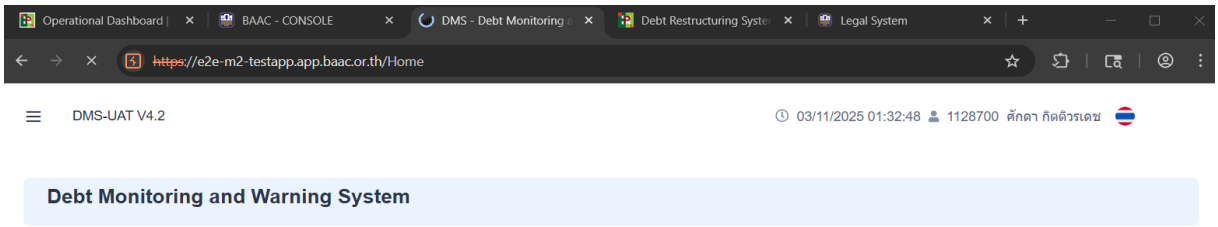
1) ทดสอบโดยใช้ Account: 1128700 ผ่านระบบ EDMS



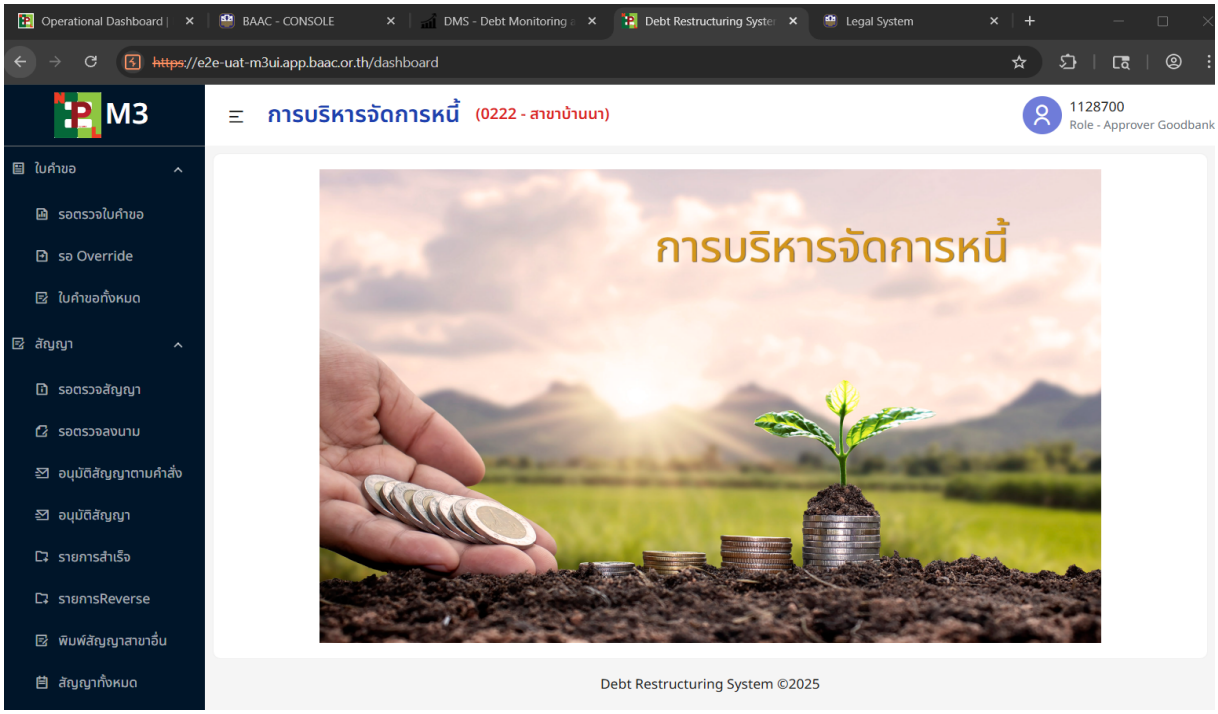
2) ทดสอบเข้าใช้งาน: การวิเคราะห์ลูกหนี้ ผ่านระบบ EDMS



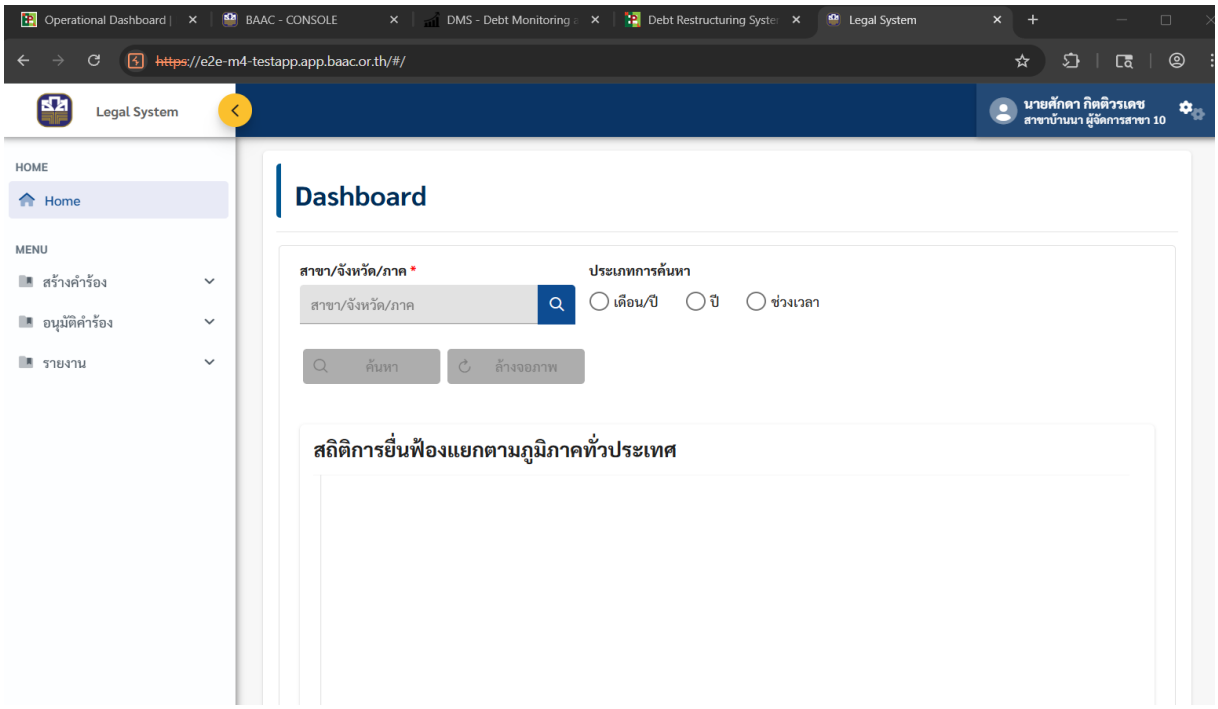
3) ทดสอบเข้าใช้งาน: การติดตามหนี้และการเตือน ผ่านระบบ EDMS



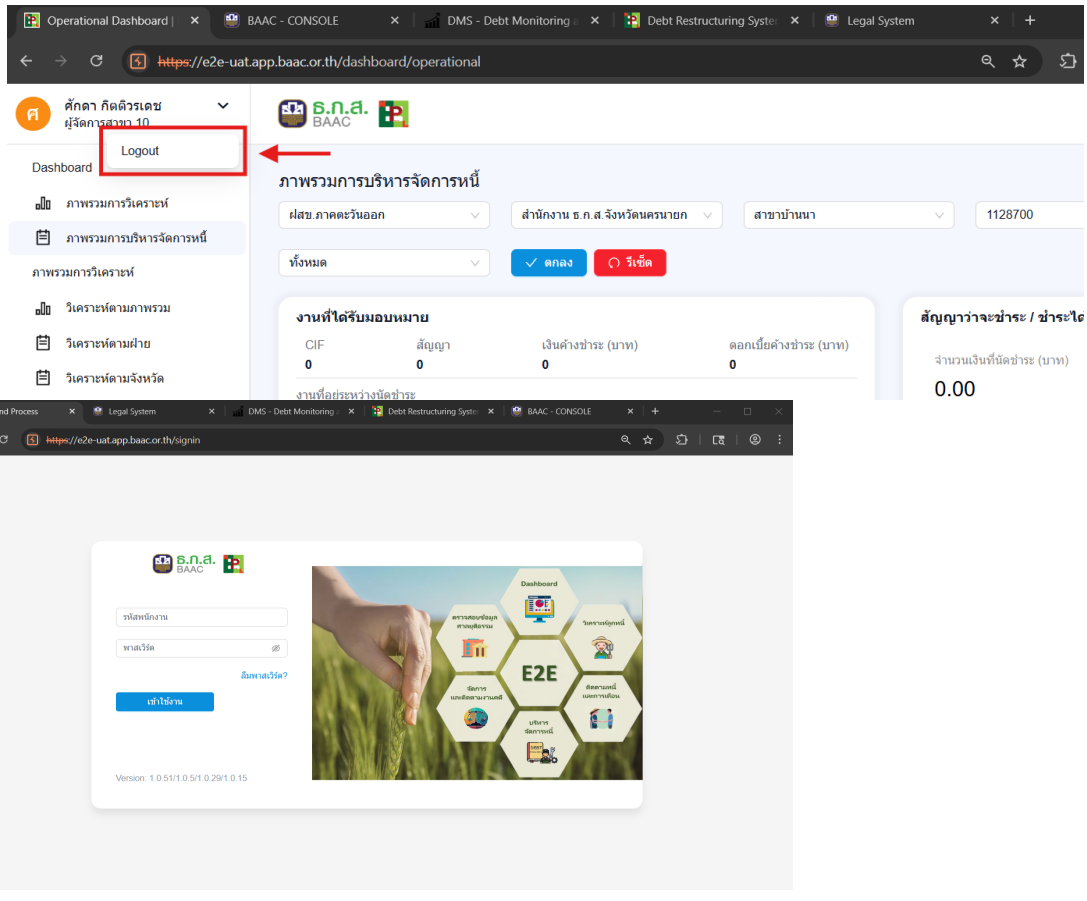
4) ทดสอบเข้าใช้งาน: การบริหารจัดการหนี้ ผ่านระบบ EDMS



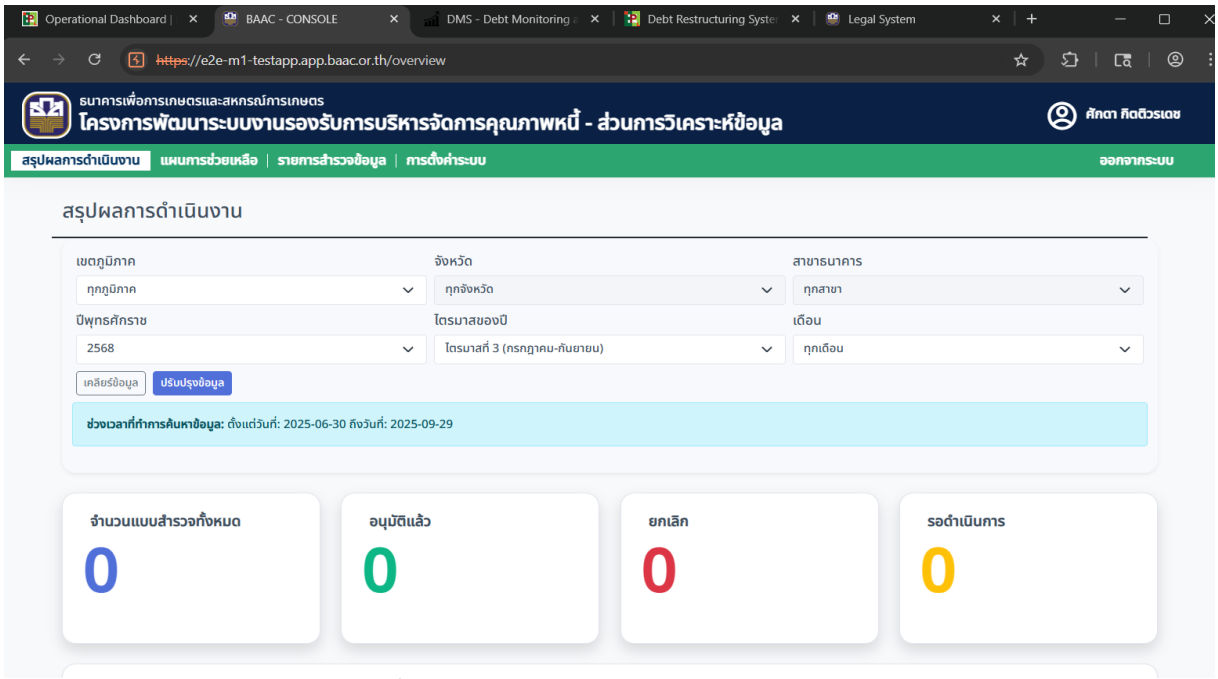
5) ทดสอบเข้าใช้งาน: การจัดการและติดตามคดี ผ่านระบบ EDMS



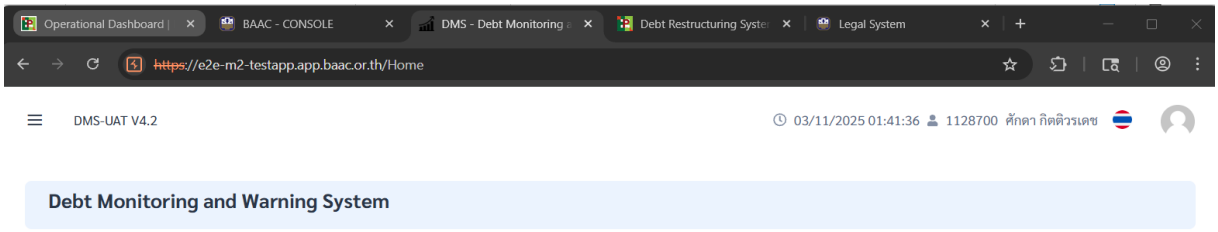
6) ทดสอบทำการ Logout ระบบ EDMS ผ่านระบบ EDMS



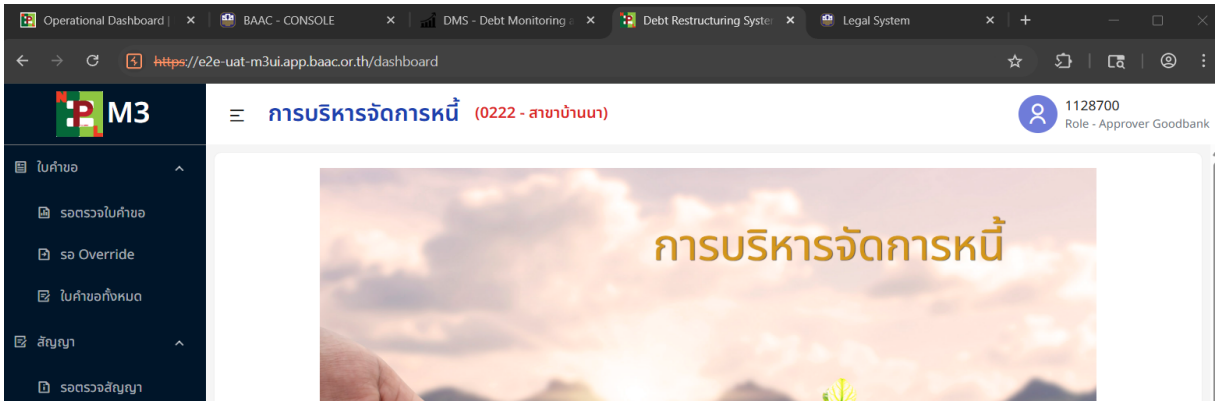
7) ทดสอบเข้าใช้งาน: การวิเคราะห์ลูกหนี้ ยังไม่ Logout ตาม



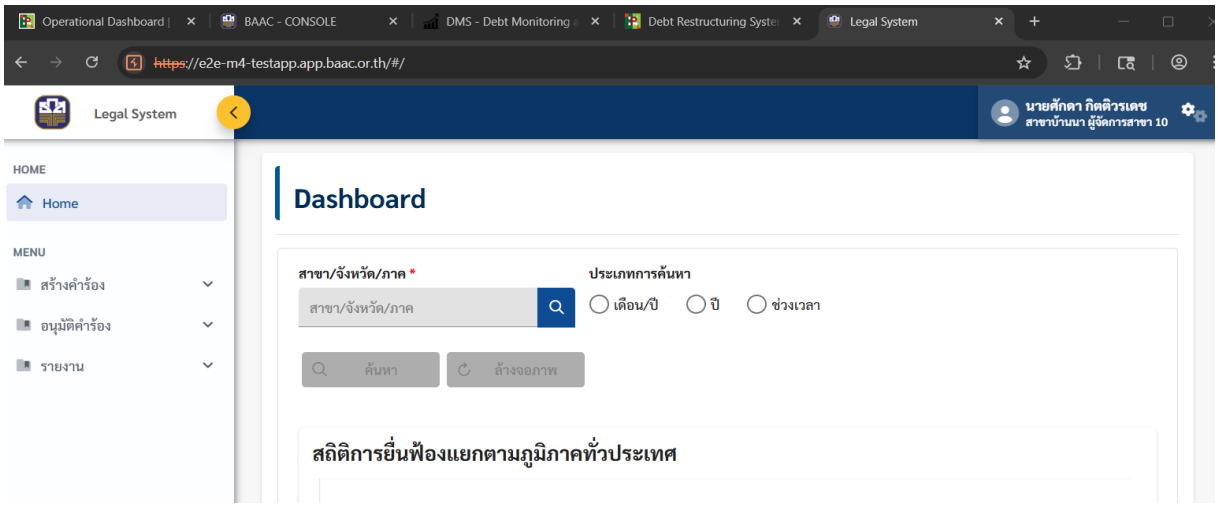
8) ทดสอบเข้าใช้งาน: การติดตามหนี้และการเตือน ยังไม่ Logout ตาม



9) ทดสอบเข้าใช้งาน: การบริหารจัดการหนี้ ยังไม่ Logout ตาม



10) ทดสอบเข้าใช้งาน: การจัดการและติดตามคดี ยังไม่ Logout ตาม



5.3 สามารถเข้าถึงข้อมูลได้โดยไม่ต้องมี Access Token

ผลการทดสอบซ้ำ

วิธีการแก้ไข	ระดับความเสี่ยง	สถานะช่องโหว่	หมายเหตุ
มีการ Encrypted ข้อมูลที่ทำการส่งโดยผู้ทดสอบจะไม่สามารถที่จะนำเอา Session token ไปใช้ต่อได้	ปานกลาง	ปิดโดยการแก้ไข	

ประเด็นข้อตรวจพบและผลกระทบ

การเข้าถึงข้อมูลโดยไม่มี Access Token โดยไม่มีการตรวจสอบว่าคำขอ (Request) ที่เข้ามา มีข้อมูลยืนยันตัวตนที่ถูกต้อง หรือ Access Token หรือไม่ และระบบยอมให้เข้าถึงข้อมูลได้เลย ซึ่งแสดงข้อมูลรายชื่อสาขา ซึ่งการส่ง Access Token ใน Request Body ผ่าน multipart/form-data เป็นแนวทางที่ ไม่เป็นไปตามแนวทางปฏิบัติที่ดี (Not a Best Practice) และก่อให้เกิดความเสี่ยงด้านความปลอดภัยหลายประการ

Attacker สามารถเข้าถึงข้อมูลส่วนตัวทั้งหมดของผู้ใช้ทุกคน ข้อมูลทางการเงิน ข้อมูลลับขององค์กร หรือข้อมูลอื่น ๆ และสามารถแก้ไขหรือลบข้อมูลทั้งหมดในระบบได้โดยไม่ต้อง Login

ที่	เป้าหมายที่กระทบ
(1)	'https://e2e-m1-testapi.app.baac.or.th/ConsoleMaster/BranchUnitList

ระดับความเสี่ยง

ระดับความเสี่ยง	ปานกลาง	ความน่าจะเป็น	สูง
		ผลกระทบ	ต่ำ
อ้างอิง	A01:2021 API1:2023 API2:2023 API5:2023 CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H (10.0 Critical) CVSS:4.0/AV:N/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:H/SI:H/SA:H (10.0 Critical)		

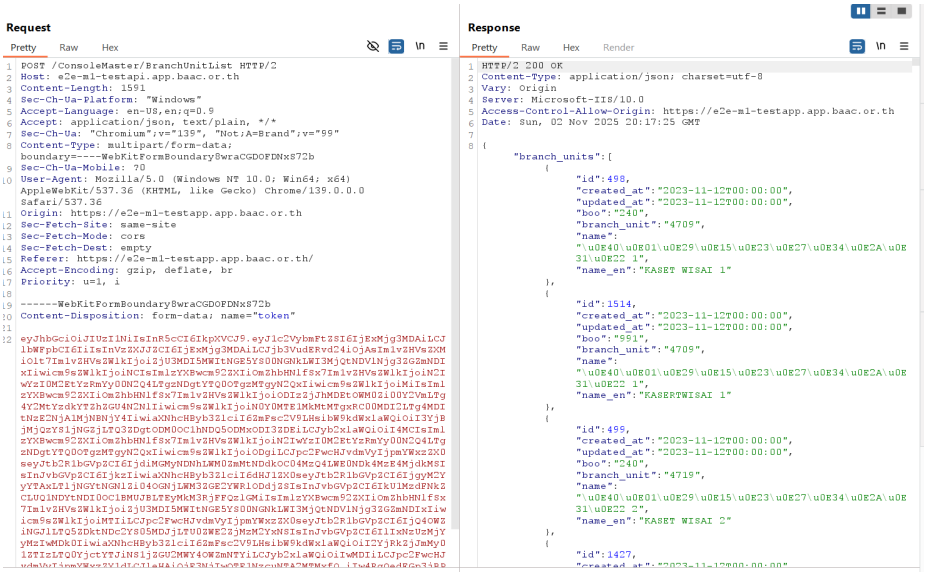
คำแนะนำในการแก้ไข

คำอธิบาย

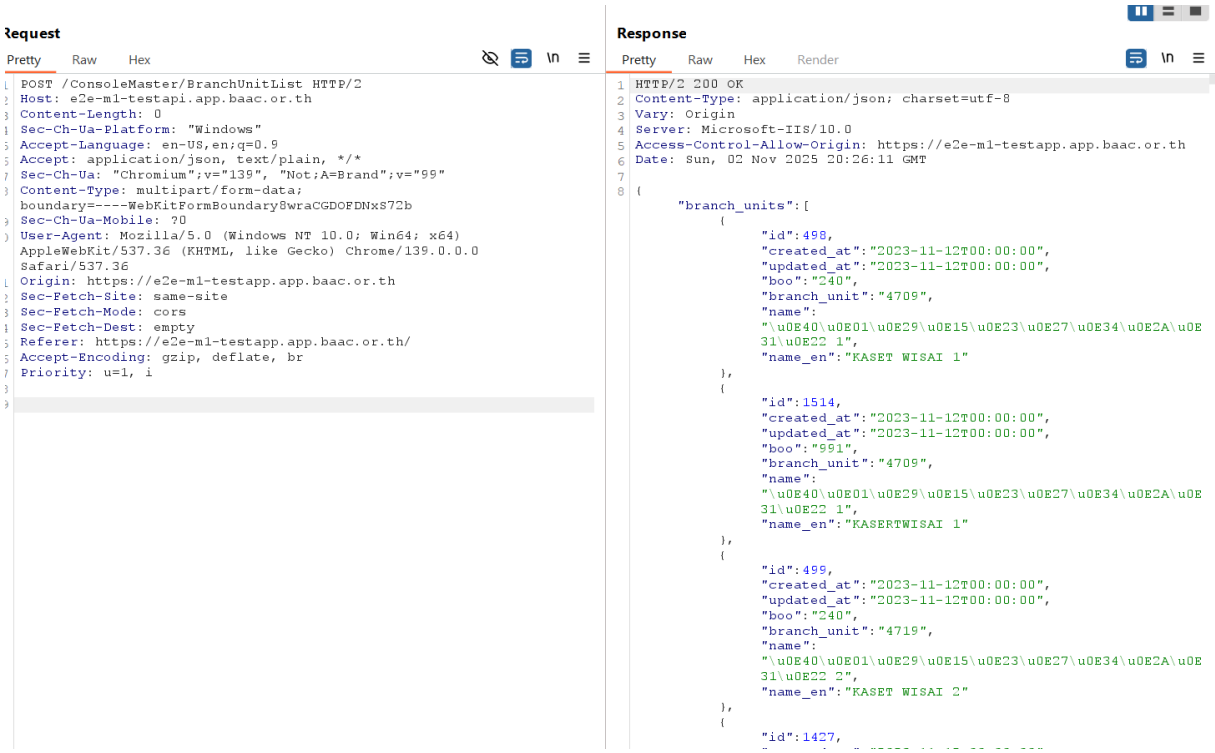
แนวทางการแก้ไขและป้องกัน
1. ยกเลิกการใช้งาน Access Token ใน Request Body ผ่าน multipart/form-data
2. เปลี่ยนการส่ง Access Token เป็น HTTP Authorization Header
- Nginx Config สำหรับบังคับใช้ Header
location /api/upload/ {
# 1. ตรวจสอบว่ามี Authorization Header หรือไม่ 	
if ($http_authorization = "") {
return 401; # ปฏิเสธทันทีถ้าไม่มี
}
# หากมี Header: ส่งคำขอไปที่ Backend (ที่ควรจะตรวจสอบ Token ต่อ) 	
proxy_pass http://your_api_backend;
}
3. หากจำเป็นต้องรับ Token จาก Request Body จริง ๆ (ซึ่งทางผู้ทดสอบไม่แนะนำ) ต้องบังคับใช้ Authentication Middleware ทุก API Endpoint ที่มีการเข้าถึงข้อมูลต้อง ถูกหุ้มด้วย Middleware หรือ Filter ที่บังคับตรวจสอบ Token โค้ดต้องมีการตรวจสอบที่รัดกุมและมั่นใจว่า Token ถูกนำไป Validate ก่อนอนุญาตให้โค้ดทำงานต่อไปได้

ตัวอย่างผลของการทดสอบ

1) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList



2) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList โดยไม่ใช้งาน Access Token



5.4 ตรวจพบมีการใช้งาน Session Token บน URL Path

ผลการทดสอบซ้ำ

วิธีการแก้ไข	ระดับความเสี่ยง	สถานะช่องโหว่	หมายเหตุ
มีการ Encrypted ข้อมูลที่ทำการส่งโดยผู้ทดสอบจะไม่สามารถที่จะนำเอา Session token ไปใช้ต่อได้	ต่ำ	ปิดโดยการแก้ไข	

ประเด็นข้อตรวจพบและผลกระทบ

การเปิดเผย Session Token บน URL Path ทำให้ Token นั้นมีความเสี่ยงที่จะถูกขโมยได้ง่ายกว่าการส่งผ่าน HTTP Cookie หรือ HTTP Header มาก Session Token จะถูกบันทึกไว้ใน Access Logs ของ Web Server (เช่น Nginx) ,Browser History และ Backend Server อย่างถาวร หาก Attacker เข้าถึงไฟล์ Log เหล่านี้ได้ ก็จะสามารถรวบรวม Session Token ที่ยังไม่หมดอายุทั้งหมด

ผลกระทบหลักคือ Session Hijacking ซึ่งหมายถึงการที่ Attacker สามารถขโมย Session ID ไปสวมรอยเป็นผู้ใช้ได้

ที่	เป้าหมายที่กระทบ
(1)	'https://e2e-uat-m3ui.app.baac.or.th/login-iauth/username/ https://e2e-m1-testapp.app.baac.or.th/login https://e2e-m1-testapi.app.baac.or.th/landing/ https://e2e-m2-testapp.app.baac.or.th/

ระดับความเสี่ยง

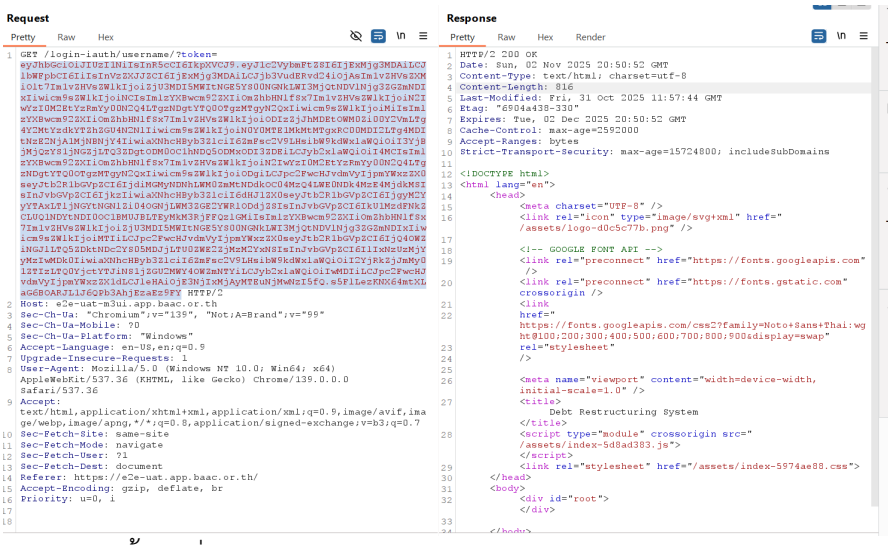
ระดับความเสี่ยง	ต่ำ	ความน่าจะเป็น	ต่ำ
		ผลกระทบ	ปานกลาง
อ้างอิง	A07:2021 CVSS:3.1/AV:A/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:N (8.6 High) CVSS:4.0/AV:A/AC:L/AT:N/PR:N/UI:N/VC:H/VI:H/VA:N/SC:N/SI:N/SA:N (8.5 High)		

คำแนะนำในการแก้ไข

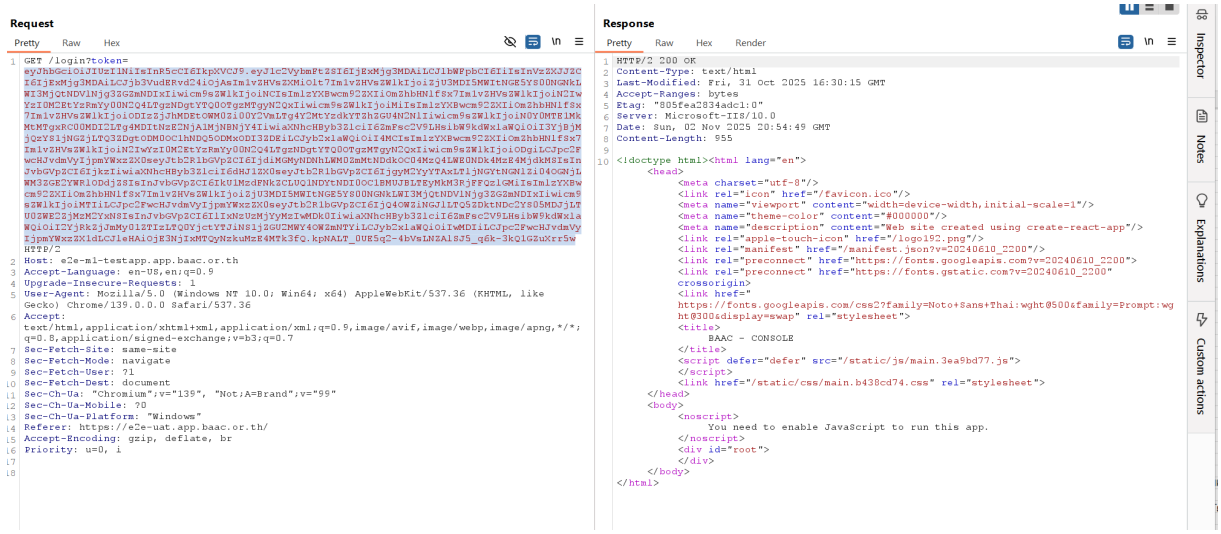
คำอธิบาย
1. ห้ามส่ง Session Token ผ่าน URL Path 2. ใช้ HTTP Authorization Header: สำหรับ API ควรส่ง Access Token หรือ Session Token ผ่าน Header (Authorization: Bearer <token> เป็นต้น) 3. กรณี Nginx หรือ IIS ต้องปิดคุณสมบัติ URL-based Session Tracking ที่ระดับ Application Backend , การ Rewrite URL (เพื่อกำจัด Session ID) และบังคับให้ใช้ HTTP Cookies (Secure,HTTPOnly, SameSite=Strict/Lax)

ตัวอย่างผลของการทดสอบ

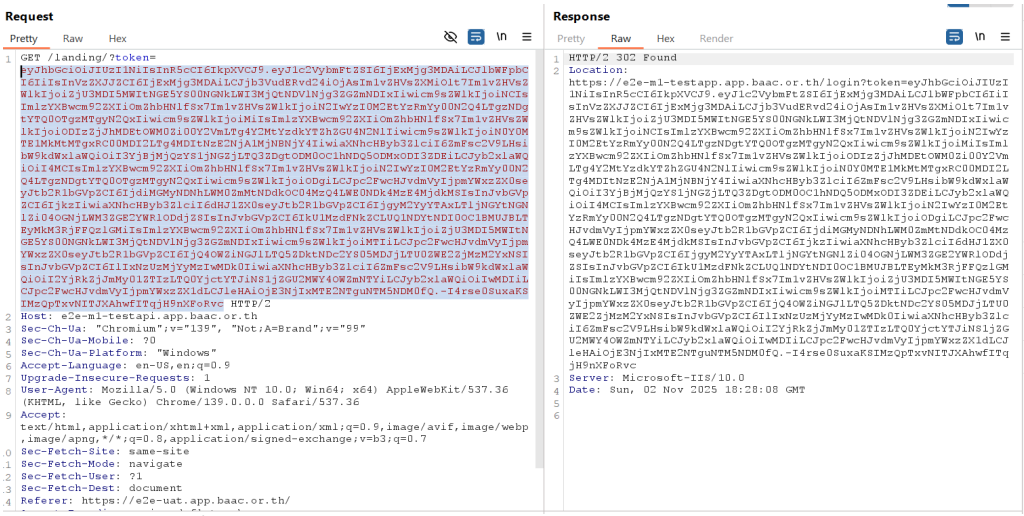
1) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList



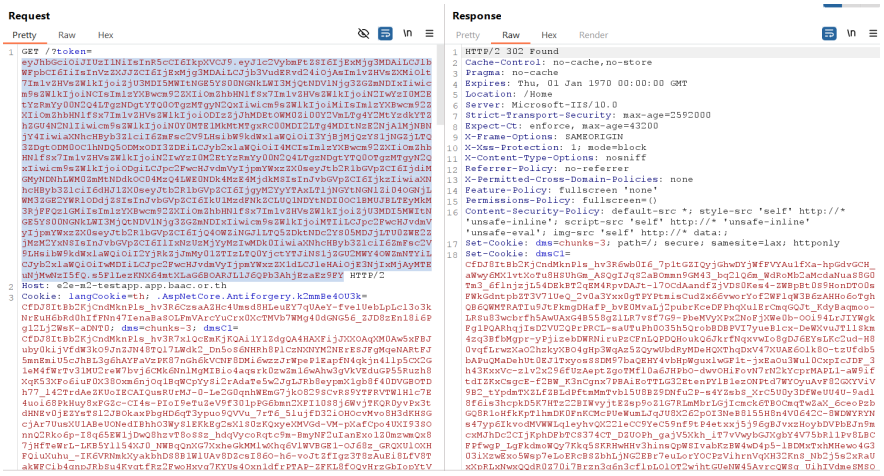
2) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList



3) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList



4) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList



5.5 ตรวจพบ Secret/Credential Key ใน source code

ผลการทดสอบซ้ำ

วิธีการแก้ไข	ระดับความเสี่ยง	สถานะช่องโหว่	หมายเหตุ
มีการนำเอา Secret และ Credential ออกจาก Source code	ต่ำ	ปิดโดยการแก้ไข	

ประเด็นข้อตรวจพบและผลกระทบ

การฝัง Secret Keys ไว้ใน Source Code ที่อาจถูกเปิดเผยสู่สาธารณะ (Sensitive Data Exposure) นั้นเกือบจะเทียบเท่ากับ Remote Code Execution (RCE) หรือ Full Access Bypass ปัจจุบันข้อมูล Key ที่พบสามารถเข้าถึงได้เลยโดยไม่มีต้องมี Access token/Credential และยังระบุไม่ได้ว่าใช้สำหรับเข้าถึงระบบหรือข้อมูลชุดใด จึงยังประเมินความเสี่ยงในการไปใช้งานต่อค่อนข้างยาก

การฝัง Secret Keys ไว้ใน Source Code ที่อาจถูกเปิดเผยสู่สาธารณะ (Sensitive Data Exposure) นั้นเกือบจะเทียบเท่ากับ Remote Code Execution (RCE) หรือ Full Access Bypass ปัจจุบันข้อมูล Key ที่พบสามารถเข้าถึงได้เลยโดยไม่มีต้องมี Access token/Credential และยังระบุไม่ได้ว่าใช้สำหรับเข้าถึงระบบหรือข้อมูลชุดใด จึงยังประเมินความเสี่ยงในการไปใช้งานต่อค่อนข้างยาก

ที่	เป้าหมายที่กระทบ
(1)	'https://e2e-uat-m3ui.app.baac.or.th/assets/index-5d8ad383.js https://e2e-m4-testapp.app.baac.or.th/static/js/main.76b30312.js

ระดับความเสี่ยง

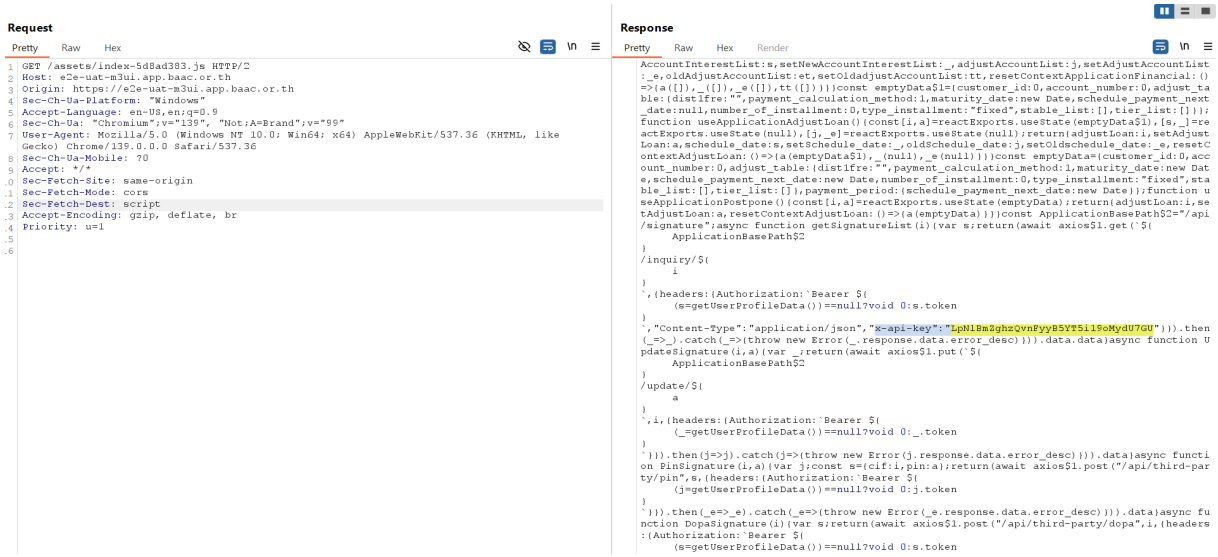
ระดับความเสี่ยง	ต่ำ	ความน่าจะเป็น	ปานกลาง
		ผลกระทบ	ต่ำ
อ้างอิง	A05:2021 A04:2021 CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:C/C:H/I:H/A:H (8.5 High) CVSS:4.0/AV:N/AC:H/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:H/SI:H/SA:H (8.6 High)		

คำแนะนำในการแก้ไข

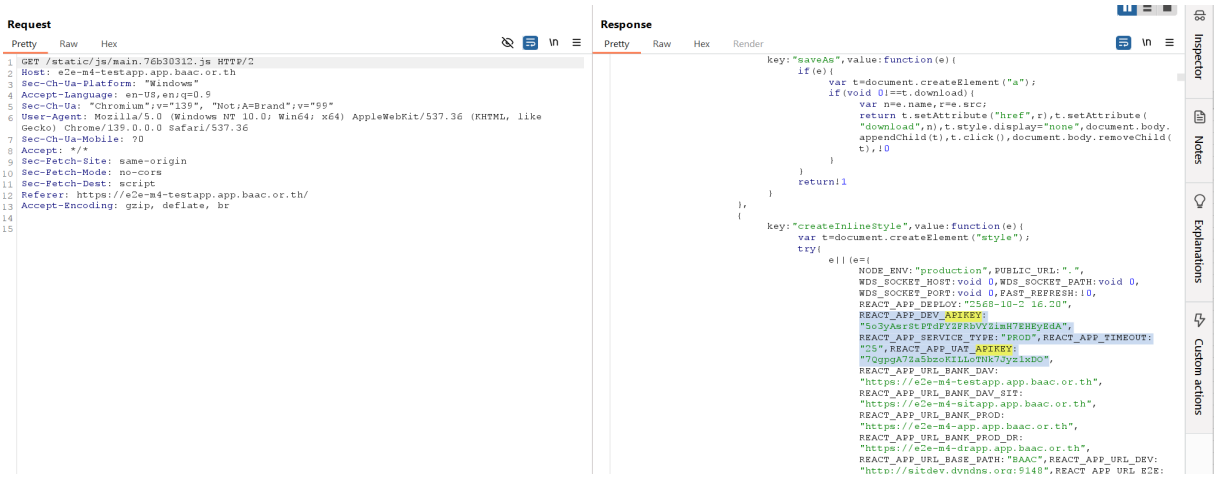
คำอธิบาย
แนวทางการแก้ไขและป้องกัน 1. ห้าม Hardcode โดยเด็ดขาด: ห้ามใส่ Secret Keys ใด ๆ ในไฟล์ Source Code (ไม่ว่าจะเป็นภาษาใดก็ตาม) โดยเด็ดขาด 2. ใช้ Environment Variables: สำหรับการตั้งค่าที่ไม่ละเอียดอ่อนมากนัก ควรใช้ Environment Variables ในการโหลด Key ระหว่างการ Deploy 3. ใช้ Secret Manager ดังนี้ - HashiCorp Vault - AWS Secrets Manager - Azure Key Vault - Google Cloud Secret Manager 4. ใช้ Git Secret Scanning: ใช้เครื่องมือเช่น GitGuardian, TruffleHog, หรือ GitHub's Secret Scanning เพื่อตรวจสอบ Repository และป้องกันไม่ให้นักพัฒนาเผลอ Commit Secret Keys เข้าไปตั้งแต่แรก

ตัวอย่างผลของการทดสอบ

1) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList



2) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList



5.6 ตรวจพบการใช้งาน Software ที่มีช่องโหว่

ผลการทดสอบซ้ำ

วิธีการแก้ไข	ระดับความเสี่ยง	สถานะช่องโหว่	หมายเหตุ
มีการอัพเดท Software ที่ใช้	ต่ำ	ปิดโดยการแก้ไข	

ประเด็นข้อตรวจพบและผลกระทบ

การตรวจพบการใช้งาน Software ที่มีช่องโหว่ (Vulnerable Software) ดังนี้

jQuery Validation Plugin - v1.19.5 มีช่องโหว่ดังนี้

- CVE-2025-3573 (XSS)

JSZip v3.1.3 มีช่องโหว่ดังนี้

- CVE-2021-23413 (DoS)
- CVE-2022-48285 (Directory Traversal)

Axios v1.6.2 มีช่องโหว่ดังนี้

- CVE-2024-39338 (Server-Side Request Forgery)
- CVE-2025-27152 (Server-Side Request Forgery)

ผลกระทบของการใช้ซอฟต์แวร์ที่มีช่องโหว่

jQuery Validation Plugin

Attacker สามารถขโมย Session Cookie หรือ Access Token ของผู้ใช้ และส่งไปยังเซิร์ฟเวอร์ของผู้โจมตี (Session Hijacking) การเปลี่ยนหน้าเว็บ: สามารถแก้ไขเนื้อหาของหน้าเว็บที่ผู้ใช้เห็น (Defacement) หรือหลอกให้ผู้ใช้ป้อนข้อมูลส่วนตัว (Phishing)

JSZip

Attacker ทำการการประมวลผลไฟล์ ZIP ที่มีความซับซ้อนจะทำให้แอปพลิเคชันหรือเซิร์ฟเวอร์ที่ประมวลผลไฟล์ ZIP ล่มหรือทำงานช้าลงอย่างมาก

Axios

Attacker สามารถใช้ Server ของคุณเป็น Proxy เพื่อสแกนหรือโจมตีระบบภายในเครือข่ายส่วนตัว (Internal Network) ที่ไม่สามารถเข้าถึงได้จากภายนอก

ที่	เป้าหมายที่กระทบ
(1)	'https://e2e-m2-testapp.app.baac.or.th/js/jquery.validate.min.js https://e2e-m2-testapp.app.baac.or.th/js/jszip.min.js

	https://e2e-m1-testapp.app.baac.or.th/static/js/main.3ea9bd77.js

ระดับความเสี่ยง

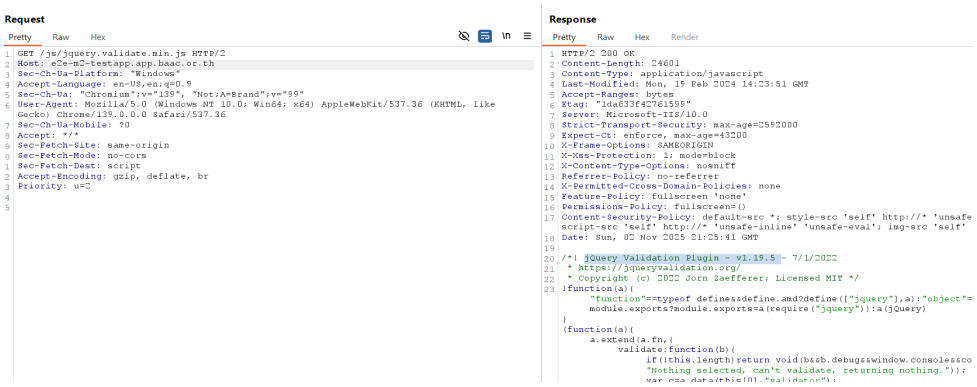
ระดับความเสี่ยง	ต่ำ	ความน่าจะเป็น	ต่ำ
		ผลกระทบ	ปานกลาง
อ้างอิง	A06:2021		

คำแนะนำในการแก้ไข

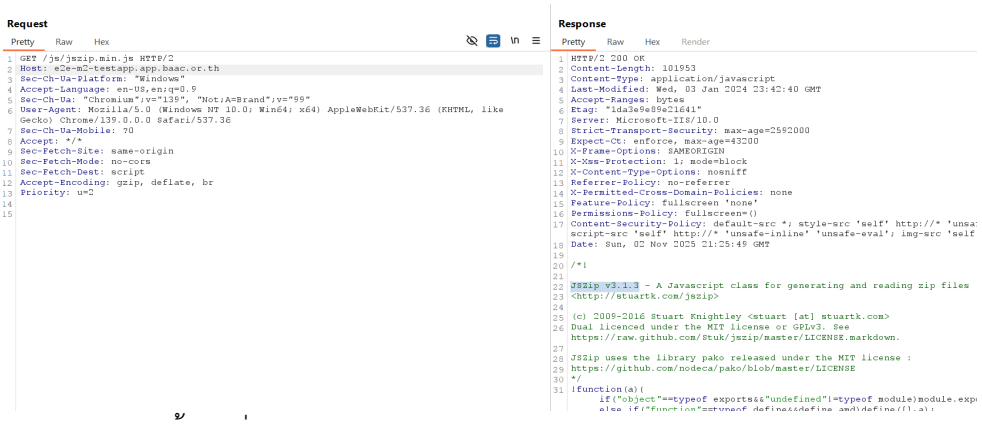
คำอธิบาย
jQuery Validation Plugin อัปเดตเป็น v1.20.0 หรือสูงกว่า JSZip อัปเดตเป็น v3.1.4 หรือ v3.10.1 หรือสูงกว่า Axios อัปเดตเป็น v1.7.0 หรือสูงกว่า

ตัวอย่างผลของการทดสอบ

1) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList



2) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList



3) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList

4) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList

5.7 ตรวจพบการใช้งาน Software บน Header

ผลการทดสอบซ้ำ

วิธีการแก้ไข	ระดับความเสี่ยง	สถานะช่องโหว่	หมายเหตุ
มีการปรับแก้ไข Response Header	**บันทึกไว้**	ปิดโดยการแก้ไข	

ประเด็นข้อตรวจพบและผลกระทบ

1) Improper Information leakage

ช่องโหว่นี้จัดอยู่ในกลุ่ม Information Leakage แม้จะดูไม่เกี่ยวกับการโจมตีหรือการระบุช่องโหว่โดยตรง แต่การเปิดเผยข้อมูล Software เป็นการให้ ข้อมูลนำเข้า (Input) แก่ Attacker ซึ่งสามารถนำไปใช้ในการโจมตีที่ซับซ้อนขึ้นได้

Attacker ไม่ต้องเสียเวลาในการคาดเดาหรือใช้เครื่องมือ Fingerprinting ที่ซับซ้อนเพื่อหาเวอร์ชันที่แน่นอน ซึ่งช่วยประหยัดเวลาและทรัพยากรในการสร้าง Payload โจมตี

ข้อมูลที่ตรวจพบมีการใช้งานดังนี้

1. Nginx
2. Microsoft IIS/10.0

2) Missing security headers

ช่องโหว่นี้คืออะไร

- X-Content-Type-Options

  บอกเบราเซอร์ไม่ให้ “เดา” ชนิดไฟล์จากเนื้อหาเพื่อป้องกัน MIME sniffing

- X-XSS-Protection

  กลไกป้องกัน XSS แบบเก่าในบางเบราเซอร์

- Strict-Transport-Security (HSTS)

  บังคับใช้ HTTPS ทุกครั้ง

โดยส่งผลกระทบแยกตามประเภทได้ดังนี้

- ขาด X-Content-Type-Options: เบราเซอร์อาจตีความไฟล์ผิดชนิด นำไปสู่โค้ดอันตรายถูกรัน
- ขาด X-XSS-Protection: ลดโอกาสกรอง XSS อัตโนมัติในเบราเซอร์ที่ยังรองรับ
- ขาด HSTS: ผู้ใช้ถูกดาวน์เกรดไป HTTP ได้ ทำให้เสี่ยง MITM และการดักฟัง

3) SSL/TLS configuration issue: ‘notAfter’

ช่องโหว่นี้คืออะไร

○ ‘notAfter’ คือวันหมดอายุของใบรับรองดิจิทัล (TLS certificate). หากหมดอายุ หรือใกล้หมดอายุจะทำให้การเชื่อมต่อ HTTPS ไม่น่าเชื่อถือโดยเกิดขึ้นได้หลายสาเหตุ เช่น ใบรับรองหมดอายุแล้วแต่ยังไม่ต่ออายุ, ติดตั้งใบรับรองผิดชุด เช่น ขาด intermediate CA ทำให้เบราเซอร์มองว่าไม่เชื่อถือ, ใช้ใบรับรองที่ออกให้โดเมนอื่น หรือชื่อโฮสต์ไม่ตรง

• โดยส่งผลกระทบคือ ผู้ใช้จะเห็นคำเตือน “ไม่ปลอดภัย” และอาจปฏิเสธการเชื่อมต่อและ เสี่ยงต่อการโจมตีแบบดักกลางทาง (MITM) เพราะผู้ใช้บางส่วนอาจยอมข้ามคำเตือน ทำให้ข้อมูลถูกดักฟังหรือแก้ไขได้

ที่	เป้าหมายที่กระทบ
(1)	https://e2e-m1-testapp.app.baac.or.th https://e2e-m1-testapi.app.baac.or.th https://e2e-m4-testapp.app.baac.or.th https://uat-efiling.app.baac.or.th

ระดับความเสี่ยง

ระดับความเสี่ยง	บันทึกไว้	ความน่าจะเป็น	ต่ำ
		ผลกระทบ	ต่ำ
อ้างอิง	A02:2021 – Cryptographic Failures AV:A/AC:L/PR:N/UI:R/S:U/C:L/I:L/A:N (3.9 Low)		

คำแนะนำในการแก้ไข

คำอธิบาย
1) Improper Information leakage การซ่อน Header ทั้งหมด (Defense in Depth) หากต้องการความปลอดภัยสูงสุด ต้องทำการลบ Header ที่เปิดเผยซอฟต์แวร์ออกไปทั้งหมดก่อนส่ง Response ให้ Client **สำหรับ Nginx** location / { ... proxy_hide_header Server; # ซ่อน Header Server (ต้องมีการตั้งค่าใน Backend ให้ทำงานร่วมกัน)

คำอธิบาย
... } **สำหรับ IIS** 1. ไปที่คีย์: HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\HTTP\Parameters 2. สร้างค่า DWORD (32-bit) ใหม่ชื่อว่า DisableServerHeader 3. กำหนดค่า (Value Data) ของ DisableServerHeader เป็น 1 4. รีสตาร์ทบริการ HTTP Service

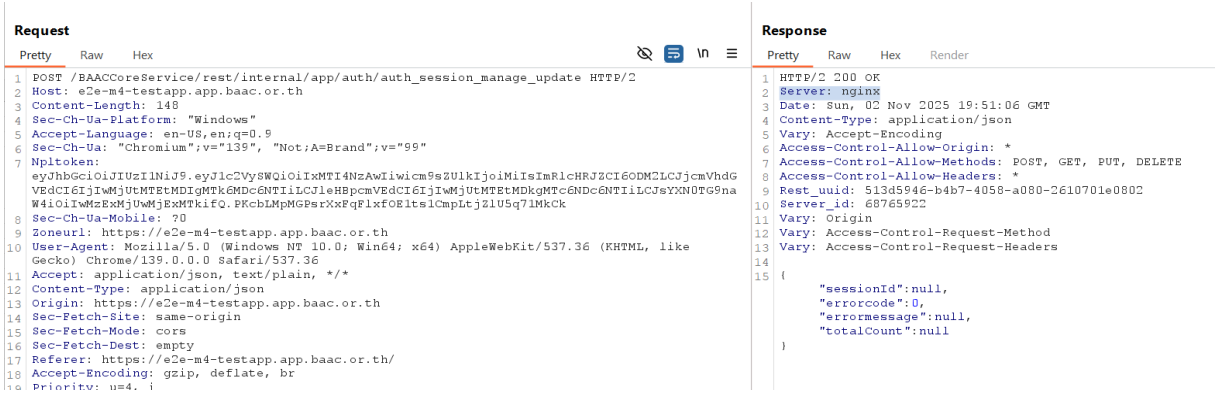
2) Missing security headers
ตั้งค่า X-Content-Type-Options:
• กำหนด X-Content-Type-Options: nosniff ใน header เพื่อสั่งให้เบราว์เซอร์ปฏิบัติตาม MIME type ที่กำหนดและไม่เดาชนิดไฟล์ .
• เปิดใช้งาน X-XSS-Protection (สำหรับเบราว์เซอร์เก่า):
• การตั้งค่า X-XSS-Protection: 1; mode=block จะเปิดฟิลเตอร์ XSS และบล็อกเพจเมื่อพบการโจมตี XSS .อย่างไรก็ตาม ฟีเจอร์นี้ถูกยกเลิกในเบราว์เซอร์ใหม่และควรใช้ Content-Security-Policy (CSP) แทน .
เปิดใช้ Strict-Transport-Security (HSTS):
• กำหนด header Strict-Transport-Security: max-age=63072000; includeSubDomains; preload เพื่อบังคับให้ผู้ใช้เชื่อมต่อผ่าน HTTPS และป้องกันการ downgrade attack
• ปรับค่าบนเว็บเซิร์ฟเวอร์:
• Nginx: เพิ่มในไฟล์ nginx.conf ภายในบล็อก server:
NGINX
add_header X-Content-Type-Options "nosniff" always;
add_header X-XSS-Protection "1; mode=block" always;
add_header Strict-Transport-Security "max-age=63072000; includeSubDomains; preload" always;

3) SSL/TLS configuration issue: 'notAfter' 1. ระบุใบรับรองที่หมดอายุ – ตรวจสอบใบรับรองดิจิทัลทั้งหมดในโครงสร้างพื้นฐานและหาจุดที่ยังใช้งานใบรับรองหมดอายุ.

คำอธิบาย
2. ต่ออายุใบรับรองทันที – บทความของ SSL Store แนะนำให้เปลี่ยนใบรับรองใหม่โดยเร็ว เนื่องจากใบรับรองหมดอายุมีผลเสียต่อภาพลักษณ์และความน่าเชื่อถือขององค์กร . 3. ติดตั้งใบรับรองใหม่ – ติดตั้งใบรับรองบนเซิร์ฟเวอร์ให้ถูกต้องตามวิธีของระบบ (เช่น Apache, Nginx หรือ IIS)

ตัวอย่างผลของการทดสอบ

1) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList



Request

Pretty Raw Hex

```
POST /BAACCoreService/rest/internal/app/auth/auth_session_manage_update HTTP/2
Host: e2e-m4-testapp.app.baac.or.th
Content-Length: 148
Sec-Ch-Ua-Platform: "Windows"
Accept-Language: en-US,en;q=0.9
Sec-Ch-Ua: "Chromium";v="139", "Not;A=Brand";v="99"
Nptoken:
eyJhbGciOiJIUzI1NiJ9.eyJ1c2VyS1QiOiIxMTI4NzAwIiwicm9sZUlkIjoiMiIsImRlcHRJZCI6ODM2LCJjcmVhdGVEdCI6IjIwMjUtMTEtMDIgMTk6MDc6NTIiLCJleHBpcmVEdCI6IjIwMjUtMTEtMDkgMTc6NDc6NTIiLCJsYXN0TG9naW4iOiIwMzExMjUwMjExMTkifQ.PKcbLMpMGPsrXxPqFlxfOE1tslCmpLtj2lU5q71MkCk
Sec-Ch-Ua-Mobile: ?0
Zoneurl: https://e2e-m4-testapp.app.baac.or.th
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/139.0.0.0 Safari/537.36
Accept: application/json, text/plain, */*
Content-Type: application/json
Origin: https://e2e-m4-testapp.app.baac.or.th
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://e2e-m4-testapp.app.baac.or.th/
Accept-Encoding: gzip, deflate, br
Priority: u=4, i
```

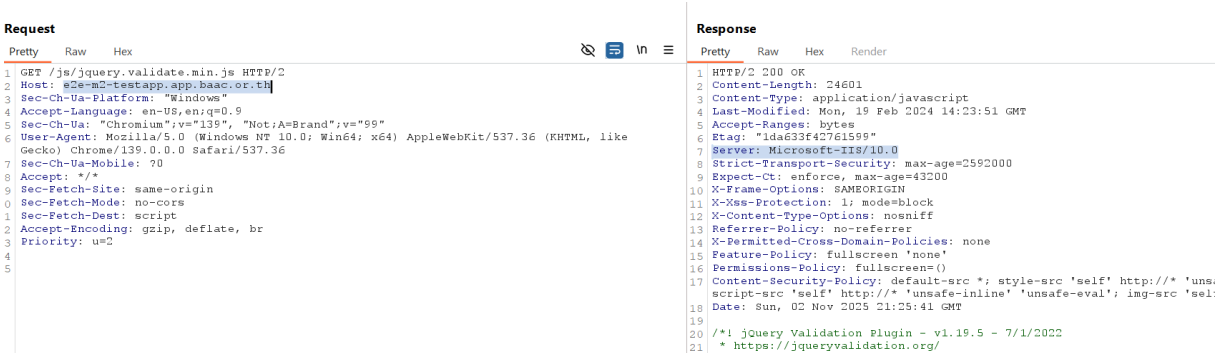
Response

Pretty Raw Hex Render

```
HTTP/2 200 OK
Server: nginx
Date: Sun, 02 Nov 2025 19:51:06 GMT
Content-Type: application/json
Vary: Accept-Encoding
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: POST, GET, PUT, DELETE
Access-Control-Allow-Headers: *
Rest_uuid: 513d5946-b4b7-4058-a080-2610701e0802
Server_id: 68765922
Vary: Origin
Vary: Access-Control-Request-Method
Vary: Access-Control-Request-Headers

{
    "sessionId":null,
    "errorcode":0,
    "errormessage":null,
    "totalCount":null
}
```

2) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList



Request

Pretty Raw Hex

```
GET /js/jquery.validate.min.js HTTP/2
Host: e2e-m2-testapp.app.baac.or.th
Sec-Ch-Ua-Platform: "Windows"
Accept-Language: en-US,en;q=0.9
Sec-Ch-Ua: "Chromium";v="139", "Not;A=Brand";v="99"
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/139.0.0.0 Safari/537.36
Sec-Ch-Ua-Mobile: ?0
Accept: */*
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: script
Accept-Encoding: gzip, deflate, br
Priority: u=2
```

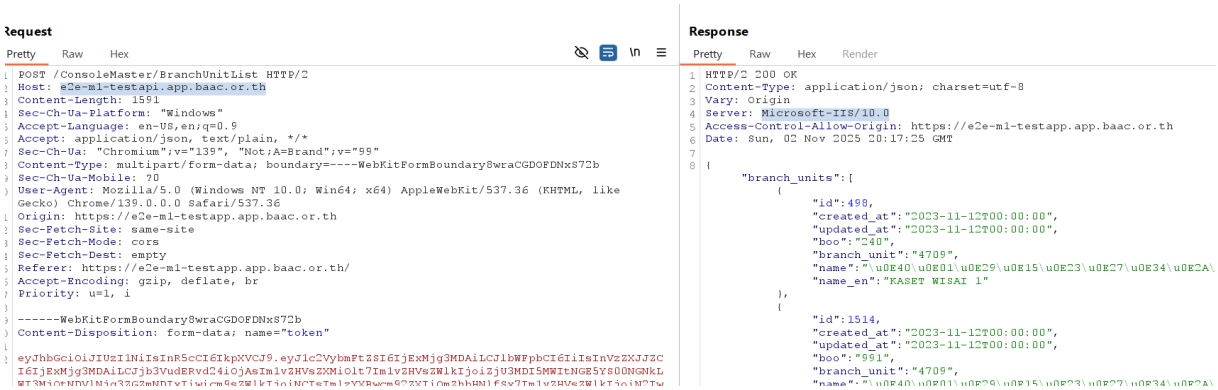
Response

Pretty Raw Hex Render

```
HTTP/2 200 OK
Content-Length: 24601
Content-Type: application/javascript
Last-Modified: Mon, 19 Feb 2024 14:23:51 GMT
Accept-Ranges: bytes
Etag: "1da633f42761:599"
Server: Microsoft-IIS/10.0
Strict-Transport-Security: max-age=2592000
Expect-Ct: enforce, max-age=43200
X-Frame-Options: SAMEORIGIN
X-Xss-Protection: 1; mode=block
X-Content-Type-Options: nosniff
Referrer-Policy: no-referrer
X-Permitted-Cross-Domain-Policies: none
Feature-Policy: fullscreen 'none'
Permissions-Policy: fullscreen=()
Content-Security-Policy: default-src *; style-src 'self' http://* 'unsa
script-src 'self' http://* 'unsafe-inline' 'unsafe-eval'; img-src 'sel
Date: Sun, 02 Nov 2025 21:25:41 GMT

/*! jQuery Validation Plugin - v1.19.5 - 7/1/2022
 * https://jqueryvalidation.org/
```

3) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList



Request

Pretty Raw Hex

```
POST /ConsoleMaster/BranchUnitList HTTP/2
Host: e2e-ml-testapi.app.baac.or.th
Content-Length: 1591
Sec-Ch-Ua-Platform: "Windows"
Accept-Language: en-US,en;q=0.9
Accept: application/json, text/plain, */*
Sec-Ch-Ua: "Chromium";v="139", "Not;A=Brand";v="99"
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary8wraCGDOFDNxS72b
Sec-Ch-Ua-Mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/139.0.0.0 Safari/537.36
Origin: https://e2e-ml-testapp.app.baac.or.th
Sec-Fetch-Site: same-site
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: https://e2e-ml-testapp.app.baac.or.th/
Accept-Encoding: gzip, deflate, br
Priority: u=1, i

------WebKitFormBoundary8wraCGDOFDNxS72b
Content-Disposition: form-data; name="token"

eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VybmFtZSI6IjExMjg3MDAiLCJlbWFpbCI6IiIsInVzZXJJZC
I6IjExMjg3MDAiLCJjb3VudERvd24iOjAsImlvZHVsZXMiOlt7ImlvZHVsZWlkIjoiZjU3MDI5MWItNGE5YS00NGNkL
```

Response

Pretty Raw Hex Render

```
HTTP/2 200 OK
Content-Type: application/json; charset=utf-8
Vary: Origin
Server: Microsoft-IIS/10.0
Access-Control-Allow-Origin: https://e2e-ml-testapp.app.baac.or.th
Date: Sun, 02 Nov 2025 20:17:25 GMT

{
    "branch_units":[
        {
            "id":498,
            "created_at":"2023-11-12T00:00:00",
            "updated_at":"2023-11-12T00:00:00",
            "boo":"240",
            "branch_unit":"4709",
            "name":"\u0E40\u0E01\u0E29\u0E15\u0E23\u0E27\u0E34\u0E2A\
            "name_en":"KASET WISAI 1"
        },
        {
            "id":1514,
            "created_at":"2023-11-12T00:00:00",
            "updated_at":"2023-11-12T00:00:00",
            "boo":"991",
            "branch_unit":"4709",
            "name":"\u0E40\u0E01\u0E29\u0E15\u0E23\u0E27\u0E34\u0E2A\
```

4) ทดสอบการ POST ขอข้อมูลผ่าน /ConsoleMaster/BranchUnitList

Request

Pretty Raw Hex

```
GET /static/js/main.3ea9bd77.js HTTP/2
Host: e2e-ml-testapp.app.baac.or.th
Sec-Ch-Ua-Platform: "Windows"
Accept-Language: en-US,en;q=0.9
Sec-Ch-Ua: "Chromium";v="139", "Not;A=Brand";v="99"
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/139.0.0.0 Safari/537.36
Sec-Ch-Ua-Mobile: ?0
Accept: */*
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: no-cors
Sec-Fetch-Dest: script
Accept-Encoding: gzip, deflate, br
```

Response

Pretty Raw Hex Render

```
HTTP/2 200 OK
Content-Type: application/javascript
Last-Modified: Fri, 31 Oct 2025 16:30:15 GMT
Accept-Ranges: bytes
Etag: "805fea2834adc1:0"
Server: Microsoft-IIS/10.0
Date: Sun, 02 Nov 2025 21:25:57 GMT
Content-Length: 4103262

/*! For license information please see main.3ea9bd77.js.LICEN
(()=>{
    var e={
        7762:(e,t,n)=>{
            "use strict";
            n.r(t),n.d(t,{
                default:()=>s
            }
```