มาตรการเสริมสร้างความมั่นคงปลอดภัยทางไซเบอร์สำหรับระบบเว็บไซต์ของหน่วยงานขนาดเล็ก

Cybersecurity Enhancement Measures for Website Systems of Small Organizations

ศุภฤกษ์ ภัทรรัชต์ภาคย์
Supparerk Phudtararutchapark
คณะสถิติประยุกต์ (GSAS)
สถาบันบัณฑิตพัฒนบริหารศาสตร์
Graduate School of Applied Statistics
National Institute of Development Administration
กรุงเทพมหานคร ประเทศไทย
Bangkok Thailand
supparerkph@gmail.com

ปราโมทย์ กั่วเจริญ
Pramote Kuacharoen
คณะสถิติประยุกต์ (GSAS)
สถาบันบัณฑิตพัฒนบริหารศาสตร์
Graduate School of Applied Statistics
National Institute of Development Administration
กรุงเทพมหานคร ประเทศไทย
Bangkok Thailand
pramote@as.nida.ac.th

บทคัดย่อ — เว็บไซต์เป็นเครื่องมือสำคัญในการเผยแพร่ข้อมูลสาธารณะ โดยเฉพาะหน่วยงานภาครัฐที่ต้องจัดทำเว็บไซต์ตามข้อกำหนดการตรวจประเมินความโปร่งใส ITA อย่างไรก็ตาม หลายหน่วยงานขาดความพร้อมด้านเทคโนโลยี ความมั่นคงปลอดภัย และงบประมาณ ทำให้ตกเป็นเป้าหมายของการโจมตีและยึดครองเว็บไซต์เพื่อนำไปใช้ในกิจกรรมที่ผิดกฎหมาย

งานวิจัยนี้นำเสนอมาตรการเสริมสร้างความมั่นคงปลอดภัยทางไซเบอร์สำหรับเว็บไซต์ของหน่วยงานขนาดเล็ก ใช้การวิจัยเชิงคุณภาพโดยการศึกษาเอกสาร (Documentary Research) และคัดเลือกแนวปฏิบัติที่ดีจากมาตรฐานสากล เช่น มาตรฐานเว็บไซต์ภาครัฐ, NIST Cybersecurity Framework และ ISO/IEC 27001 ใช้การออกแบบสถาปัตยกรรมแบบ Single-node Multi-layer Defense ติดตั้ง Security Patch Update, ทำ Configuration Hardening, คัดเลือกซอฟต์แวร์โอเพ่นซอร์สที่จำเป็นสำหรับควบคุม ตรวจจับ และป้องกันการโจมตีทางไซเบอร์ เพิ่มระบบป้องกันการโจมตีเว็บไซต์ Web Application Firewall (WAF) และใช้ Automation ผ่าน Ansible ในการทำ Centralized Management ทำให้สามารถดำเนินการบนงบประมาณจำกัด แต่มีประสิทธิภาพสูง ต่อยอดการบริหารจัดการแบบรวมศูนย์ ลดความเสี่ยง ยกระดับความมั่นคงปลอดภัย และเสริมสร้างความน่าเชื่อถือของหน่วยงานภาครัฐ

คำสำคัญ — ความมั่นคงปลอดภัยทางไซเบอร์, เว็บแอปพลิเคชันไฟร์วอลล์, ระบบอัตโนมัติ, หน่วยงานภาครัฐ, การโจมตีเว็บไซต์

ABSTRACT — Websites are essential tools for disseminating public information, especially for government agencies that are required to maintain official websites under the Integrity and Transparency Assessment (ITA) regulations. However, many agencies lack sufficient technological readiness, cybersecurity measures, and budgetary resources, making them vulnerable to cyberattacks and website takeovers for illicit purposes.

This research proposes cybersecurity enhancement measures for websites of small-scale public sector organizations. A qualitative research methodology was adopted, using documentary research to identify best practices from international standards such as the Government Website Standard, the NIST Cybersecurity Framework, and ISO/IEC 27001. The proposed solution features Single-node Multi-layer Defense architecture, incorporating security patch updates, configuration hardening, and the selection of essential open-source software for monitoring, detection, and prevention of cyber threats. The system includes a Web Application Firewall (WAF) and centralized management through automation using Ansible.

The approach is designed to be cost-effective while delivering high security performance. It helps reduce cyber risks, enhances website security posture, and increases the credibility of public sector organizations.

Keywords — Cybersecurity, Web Application Firewall (WAF), Automation System, Government Agencies, Website Attacks

1. บทนำ

ในยุคที่เทคโนโลยีดิจิทัลมีบทบาทสำคัญในการขับเคลื่อนบริการสาธารณะและการบริหารจัดการภาครัฐ เว็บไซต์เป็นหนึ่งในช่องทางหลักสำหรับเผยแพร่ข้อมูลข่าวสารและให้บริการออนไลน์แก่ประชาชน ข้อมูลอ้างอิงจากรายงานของสำนักงานสถิติแห่งชาติ เรื่อง การสำรวจการมีการใช้เทคโนโลยีสารสนเทศและการสื่อสารในครัวเรือน พ.ศ. 2567 (ไตรมาส 3) ระบุว่า ผลสำรวจประชาชนอายุ 6 ปี ขึ้นไป ประมาณ 66 ล้านคน มีผู้ใช้อินเทอร์เน็ตมากถึงร้อยละ 89.7 [1] และสำนักงานคณะกรรมการป้องกันและปราบปรามการทุจริตแห่งชาติ (ปปช.) ซึ่งมีอำนาจหน้าที่เสริมสร้างความโปร่งใสในการปฏิบัติราชการ ได้กำหนดให้หน่วยงานภาครัฐต้อง

จัดทำเว็บไซต์เพื่อการเปิดเผยข้อมูลการปฏิบัติงานต่อสาธารณะ [2] ส่งผลให้หน่วยงานขนาดเล็กที่มีอัตรากำลัง บุคลากร กระบวนการควบคุม การตรวจสอบ และงบประมาณจำกัด เช่น สถานีตำรวจ และองค์การบริหารส่วนตำบล (อบต.) จำเป็นต้องจัดทำเว็บไซต์ตามข้อกำหนด ทำให้ขาดอุปกรณ์ที่จำเป็น เช่น Network Firewall [3], Web Application Firewall (WAF), Vulnerability Scanner and Penetration Testing Tools เนื่องจากที่มีราคาสูง และขาดกระบวนการด้านความปลอดภัยและการเฝ้าระวังภัยทางไซเบอร์ ขาดการบริหารจัดการแบบรวมศูนย์ ส่งผลให้ระบบเว็บไซต์ตกเป็นเป้าหมายการโจมตีทางไซเบอร์ ถูกยึดครองเว็บไซต์เพื่อฝังลิงก์เปลี่ยนเส้นทาง การเปลี่ยนแปลงหน้าเว็บไซต์ เว็บการพนันออนไลน์ใช้โดเมนภาครัฐในการทำ Search Engine Optimization หรือแอบใช้ทรัพยากรทำเหมือง Cryptocurrency

งานวิจัยนี้มุ่งศึกษาเพื่อคัดเลือกมาตรการยกระดับความมั่นคงปลอดภัยทางไซเบอร์ที่จำเป็นสำหรับระบบเว็บไซต์ไม่ให้ถูกโจมตีและยึดครองจากผู้ไม่ประสงค์ดี รวมถึงศึกษาหาแนวทางการพัฒนาต่อยอดเพื่อสนับสนุนให้การบริหารจัดการระบบเว็บไซต์ของหน่วยงานขนาดเล็กแบบรวมศูนย์สามารถกระทำได้โดยง่าย ลดภาระของเจ้าหน้าที่ทั้งส่วนกลางและส่วนท้องถิ่น เพิ่มประสิทธิภาพในการดูแลระบบรักษาความมั่นคงปลอดภัย สามารถนำไปประยุกต์ใช้ได้จริงกับระบบเว็บไซต์ของหน่วยงานขนาดเล็กในประเทศไทย

2. วัตถุประสงค์ของการวิจัย

งานวิจัยนี้มีวัตถุประสงค์ดังนี้ (1) เพื่อพัฒนามาตรการเสริมสร้างความมั่นคงปลอดภัยทางไซเบอร์ที่สามารถนำไปประยุกต์ใช้กับระบบเว็บไซต์ของหน่วยงานขนาดเล็กในประเทศไทย เน้นการแก้ปัญหาจากข้อจำกัดของหน่วยงาน ความเชี่ยวชาญของบุคลากร และงบประมาณของหน่วยงานขนาดเล็กระดับท้องถิ่น (2) เพื่อออกแบบสถาปัตยกรรมและระบบรักษาความมั่นคงปลอดภัยเว็บไซต์บนเครื่องแม่ข่ายเดี่ยว (Single-node) เพื่อให้หน่วยงานขนาดเล็กดำเนินการได้เอง (3) เพื่อคัดเลือกมาตรการและมาตรฐานทางเทคนิคขั้นต่ำ เช่น มาตรฐานเว็บไซต์ภาครัฐ เวอร์ชัน 3 ของสำนักงานพัฒนารัฐบาลดิจิทัล [4] , NIST Cybersecurity Framework [5] , ISO/IEC 27001: 2022 [6], OWASP Top Ten [7] มาประยุกต์ใช้ให้เหมาะสมกับบริบทของหน่วยงานภาครัฐขนาดเล็กของประเทศไทย (4) เพื่อคัดเลือกซอฟต์แวร์โอเพนซอร์สของระบบเว็บไซต์และระบบรักษาความปลอดภัยสำหรับเว็บไซต์ ให้ติดตั้งและทำงานร่วมกันอยู่บนเครื่องแม่ข่ายเดี่ยว ทำงานเป็นระบบป้องกันภัยคุกคามหลายชั้น (Layer of Defense) รวมถึงมาตรการที่จำเป็นอื่น เช่น Network Firewall, Brute force protection, Web Application Firewall (WAF) และ Log rotate เป็นต้น โดยให้ครอบคลุมทั้งด้านการป้องกัน ตรวจจับ และตอบสนองต่อการโจมตี (5) เพื่อทดสอบประสิทธิภาพในการป้องกันภัยคุกคามทางไซเบอร์แบบ Unit test โดยใช้เครื่องมือ Tenable Nessus Vulnerability Scanner [8] ในการตรวจสอบช่องโหว่ในระบบ พร้อมทำ WAF Integration test บนสภาพแวดล้อมจริงเป็นเวลา 30 วันเพื่อวิเคราะห์การโจมตีที่ WAF ตรวจพบและป้องกันได้ (6) เพื่อศึกษาและนำระบบ Automation มาบริหารจัดการแบบอัตโนมัติและรวมศูนย์ ทำให้เจ้าหน้าที่ส่วนกลางสามารถติดตั้งระบบเว็บไซต์และปรับตั้งค่าให้ทำงานอย่างถูกต้องและมั่นคงปลอดภัย เป็นแนวทางสนับสนุนเพื่อให้เกิดการบริหารจัดการระบบเว็บไซต์จากส่วนกลางเป็นการลดภาระของบุคลากรในพื้นที่ และเพิ่มความรวดเร็วในการทำงานในภาพรวม

งานวิจัยนี้มีการทบทวนวรรณกรรมและกรอบมาตรฐานด้านความมั่นคงปลอดภัย เพื่อใช้เป็นแนวทางในการออกแบบมาตรการที่จำเป็นดังนี้ (1) NIST Cybersecurity Framework (NIST CSF) มีการกำหนดการจัดการความมั่นคงปลอดภัย 6 ส่วน คือ Govern, Identify, Protect, Detect, Respond และ Recover โดยงานวิจัยนี้ใช้แนวคิดในเชิงโครงสร้างเพื่อจัดหมวดหมู่ของมาตรการ เช่น การศึกษาองค์ประกอบของระบบเว็บไซต์และการโจมตีที่อาจเกิดขึ้น (Identify) การใช้ WAF (Detect + Respond) (2) OWASP Top Ten เป็นการจัดกลุ่มช่องโหว่ด้านความปลอดภัยเว็บไซต์ที่พบบ่อย ใช้เป็นแนวทางการออกแบบเพื่อป้องกันการโจมตี และเป็นหัวข้อการทดสอบ WAF เช่น Injection, Broken Authentication, Security Misconfiguration และ Cross-site Scripting (XSS) (3) มาตรฐาน ISO/IEC 27001 เป็นกรอบแนวทางพัฒนาระบบบริหารจัดการความมั่นคงปลอดภัยของข้อมูล (ISMS) โดยเลือกใช้เฉพาะที่เกี่ยวข้องกับการควบคุมทางเทคนิค เช่น การควบคุมการเข้าถึง (Access Control) และการจัดการช่องโหว่ (Vulnerability Management) เพื่อป้องกันการโจมตี เช่น Fail2ban, การไม่ให้ root login โดยตรง, และกำหนดนโยบายรหัสผ่าน (4) มาตรฐานเว็บไซต์ภาครัฐของสำนักงาน DGA เป็นมาตรฐานที่หน่วยงานรัฐใช้เป็นแนวทางในการจัดทำเว็บไซต์ เช่น การจัดหมวดหมู่เนื้อหา การป้องกันการบุกรุก และการแจ้งเตือนความผิดปกติ งานวิจัยจึงนำแนวทางเหล่านี้มาบูรณาการผ่านเครื่องมือโอเพนซอร์สที่ติดตั้งบนเครื่องแม่ข่ายเครื่องเดียว และควบคุมจากศูนย์กลางโดยไม่ต้องพึ่งระบบเชิงพาณิชย์ ทำให้ระบบต้นแบบนี้มีความมั่นคงปลอดภัยและสนับสนุนการบริหารจัดการแบบรวมศูนย์ ภายใต้ข้อจำกัดด้านงบประมาณ

3. วิธีการดำเนินการวิจัย

ใช้วิธีการวิจัยเชิงคุณภาพโดยศึกษาเอกสาร (Documentary Research) จากแหล่งข้อมูลที่น่าเชื่อถือ เพื่อรวบรวมแนวปฏิบัติที่ดีจากกรอบการทำงานและมาตรฐานด้านความมั่นคงปลอดภัยทั้งของประเทศไทย และในระดับสากล ได้แก่ มาตรฐานเว็บไซต์ภาครัฐ เวอร์ชัน 3 ของสำนักงานพัฒนารัฐบาลดิจิทัล, NIST Cybersecurity Framework และ ISO/IEC 27001: 2022, และ OWASP Top Ten นำมาวิเคราะห์ สังเคราะห์ และประยุกต์ใช้ในการออกแบบระบบต้นแบบที่เหมาะสมกับบริบทของหน่วยงานขนาดเล็กในประเทศไทย เน้นการบริหารความเสี่ยง การติดตามและแก้ไขช่องโหว่ในระบบเว็บไซต์ การจำแนกประเภทของภัยคุกคาม และการวางแนวทางในการตรวจจับ ตอบสนอง และฟื้นฟูระบบภายใต้ข้อจำกัดของหน่วยงานภาครัฐระดับท้องถิ่น

กระบวนการออกแบบระบบเริ่มจากการศึกษาองค์ประกอบที่จำเป็นในระบบเว็บไซต์ (Identify) การคัดเลือกซอฟต์แวร์โอเพนซอร์สสำหรับการทำหน้าที่ในแต่ละองค์ประกอบ ศึกษาการโจมตีที่อาจเกิดขึ้นพร้อมหาแนวทางการตรวจจับ (Protect) ป้องกัน (Detect) และตอบสนอง (Response) เพื่อรับมือการโจมตีเหล่านั้น และเพื่อให้สอดคล้องกับงบประมาณที่จำกัด จึงกำหนดให้ส่วนประกอบทั้งหมดทำงานด้วยซอฟต์แวร์แบบโอเพนซอร์ส

ที่มีประสิทธิภาพสูง และมีเสถียรภาพในการทำงาน ได้รับการยอมรับโดยชุมชนผู้ใช้โอเพนซอร์ส (Opensource Software Community) และเป็นที่ยอมรับจากชุมชนด้านความมั่นคงปลอดภัยไซเบอร์ (Cybersecurity Community)

เพื่อให้นำไปปฏิบัติได้จริง ผู้วิจัยคัดเลือกหัวข้อการกำหนดค่า Security Configuration Guideline เท่าที่จำเป็นโดยอ้างอิงจาก CIS Benchmark [9] และ OWASP Secure Configuration ใช้กำหนดการตั้งค่าระบบเว็บไซต์เพื่อเสริมสร้างความปลอดภัย เช่น การปิดช่องโหว่ในระบบ ปิดบริการและพอร์ตบนเครือข่ายที่ไม่จำเป็น การลด Attack Surface [10] การจำกัดสิทธิ์การเข้าถึง การตั้งค่านโยบายรหัสผ่านและการเข้าถึงทางไกล การตั้งค่า HTTP Security Headers และออกแบบแนวทางการบริหารจัดการต่าง ๆ แบบรวมศูนย์และทำงานแบบอัตโนมัติ (Automation) โดยใช้เทคนิคการจัดการแบบ Idempotent [11] ที่สามารถเรียกใช้ซ้ำได้โดยไม่ทำให้ระบบเกิดความผิดพลาด เหมาะสมกับบริบทการดูแลระบบจำนวนมากจากหน่วยงานส่วนกลาง และสุดท้ายเป็นการประเมินประสิทธิภาพระบบต้นแบบ 2 ด้าน คือ (1) การตรวจสอบช่องโหว่โดยใช้เครื่องมือ Tenable Nessus Vulnerability Scanner ทำการตรวจสอบก่อนและหลังการดำเนินมาตรการที่กำหนดเพื่อเปรียบเทียบจำนวนช่องโหว่ (Vulnerability) และความรุนแรง (Severity) ที่เปลี่ยนแปลงไป (2) วิเคราะห์ความสามารถตรวจจับและป้องกันช่องโหว่ผ่านการโจมตีผ่านอินเทอร์เน็ตผ่าน HTTP protocol จาก logs ของ WAF บนเว็บไซต์ที่ใช้งานจริงในช่วงเวลา 30 วัน

4. ผลการวิจัย

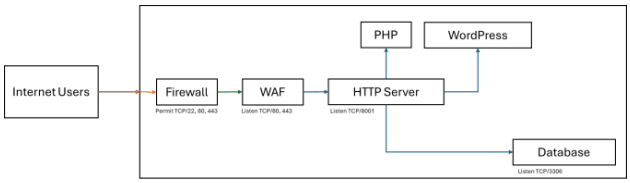
องค์ประกอบของระบบเว็บไซต์ที่ใช้ในหน่วยงานขนาดเล็ก มีองค์ประกอบที่สำคัญอยู่ 4 ส่วน คือ (1) ระบบปฏิบัติการของเครื่องแม่ข่าย เช่น Linux OS (2) Web Server เช่น Apache HTTP (3) ระบบ Content Management System (CMS) เช่น WordPress และ (4) ระบบฐานข้อมูล เช่น MySQL โดยองค์ประกอบทั้งหมดนี้ถูกติดตั้งอยู่บนเครื่องคอมพิวเตอร์แม่ข่ายเสมือน (Virtual Machine) เพียงเครื่องเดียว มีการโจมตีที่อาจเกิดขึ้นกับระบบดังนี้ (1) การโจมตีระบบปฏิบัติการ (OS Attack) เช่น การโจมตี Brute Force Password Attack เพื่อทำ Remote Access logon ผ่านบริการ Telnet หรือ SSH, การโจมตีผ่านบริการที่เข้าถึงได้บนเครือข่าย เช่น Bind DNS (2) การโจมตีผ่าน Web Server ที่ Misconfiguration ทำให้เรียกใช้งาน Directory listing เพื่อทำ Path Traversal เข้าถึงไฟล์สำคัญของระบบ เช่น /etc/passwd หรือไฟล์ที่ไม่ควรเข้าถึง เช่น phpinfo.php, และ Buffer Overflow ที่สามารถโจมตี Module ของ Apache ที่มีช่องโหว่ (3) การโจมตี Web Application (CMS + PHP) เช่น Code Injection โดยการส่งคำสั่ง ผ่าน HTTP Header หรือ Payload และการทำ Remote Code Execution (RCE) โดยใช้ช่องโหว่ของ CMS หรือ Plugin เพื่อรันคำสั่งบนเครื่องคอมพิวเตอร์แม่ข่าย, Cross-Site Scripting (XSS) เพื่อหลอกผู้ใช้หรือลักข้อมูล session, File Upload Vulnerability อัปโหลด Script เพื่อควบคุมระบบ, ใช้ช่องโหว่ของ CMS หรือ Plugin ที่มีช่องโหว่ (4) การโจมตีฐานข้อมูล MySQL เช่น SQL Injection ส่งคำสั่ง SQL ผ่าน Web Application ที่เชื่อมโยงกันเพื่อดึงข้อมูลหรือเข้าควบคุมฐานข้อมูล, Default Configuration Attack ใช้รหัสผ่านเริ่มต้นหรือ user ที่ไม่ได้จำกัดสิทธิ์, Remote MySQL Access โดยการโจมตีแบบ Brute Force, Password Spray หรือ Exploitation จาก Public IP โดยตรง ผลการวิเคราะห์รูปแบบการโจมตีระบบเว็บไซต์ข้างต้น สามารถกำหนดแนวทางป้องกันได้ดังนี้ (1) ใช้ Network Firewall เพื่อลด Attack Surface ของระบบป้องกันบริการที่เปิดไว้บนเครือข่ายโดยปริยายจากการสแกนและโจมตีโดยจำกัดให้เรียกใช้ได้เฉพาะบริการที่ตั้งใจเปิดให้เข้าถึง คือ SSH, HTTP, และ HTTPS ดังแสดงในตารางที่ 1 (2) ใช้ WAF ป้องกันการโจมตี Web Server เช่น Code / SQL Injection, XSS, การโจมตีผ่านช่องโหว่ของ CMS หรือ Plugin และป้องกันการโจมตีระบบฐานข้อมูล (3) การใช้ Fail2Ban [12] เพื่อป้องกันการโจมตีแบบ Brute Force ผ่านบริการ SSH โดย Source IP Address ที่ logon ไม่สำเร็จต่อเนื่องกัน 5 ครั้งภายในระยะเวลา 10 นาทีจะถูกแบนเป็นเวลา 10 นาที จึงจะเปิดให้ login ได้อีกครั้ง (4) การตั้งค่า SSH ไม่ให้ login ด้วย root ป้องกันการ Brute Force root Password ได้อย่างมีประสิทธิภาพ (5) การทำ Configuration Hardening เพื่อกำหนดนโยบายรหัสผ่านให้ซับซ้อนคาดเดาได้ยาก และทำงานควบคู่ไปกับ Fail2Ban จะทำให้การสุ่มทดลองรหัสผ่านแทบจะเป็นไปไม่ได้เลย, Log Rotate จะดำเนินการให้ System log ถูกจัดเก็บครบถ้วนตามระยะเวลาที่กำหนด เช่น 13 สัปดาห์หรือ 91 วัน และไม่ให้ log มีขนาดใหญ่เกินกำหนด (6) การ Update Security Patch จะปิดช่องโหว่ของระบบปฏิบัติการและองค์ประกอบต่าง ๆ ของระบบเว็บไซต์ การดำเนินการทั้ง 6 ข้อนี้เป็นเพียงมาตรการขั้นต่ำที่จำเป็นเพื่อลดความเสี่ยงในการถูกโจมตี แต่ยังไม่ได้ปิดช่องโหว่ทั้งหมดอย่างสมบูรณ์ เช่น ระบบฐานข้อมูลยังคงมีช่องโหว่แต่ไม่สามารถโจมตีได้โดยตรง เนื่องจากการเข้าถึงบริการฐานข้อมูลจากภายนอกได้รับการป้องกันจาก Firewall แล้ว

ตารางที่ 1. IP traffic ที่ถูกควบคุมด้วย ufw / iptables Firewall

Source	Destination	Port	Firewall Action
Any IP	Web IP	TCP/22	Permit Internet to web
Any IP	Web IP	TCP/80	Permit Internet to web
Any IP	Web IP	TCP/443	Permit Internet to web
Any IP	Web IP	ANY	Implicit Deny (by default)

การคัดเลือกซอฟต์แวร์โอเพนซอร์สตามหน้าที่ของแต่ละองค์ประกอบจากความนิยมในการใช้งานเนื่องจากซอฟต์แวร์ที่มีจำนวนผู้ใช้งานจำนวนมากจะมีข้อมูลสนับสนุน มีการแก้ไขปัญหา และถูกตรวจพบช่องโหว่ได้มากกว่า ดังนี้ (1) Operating System เลือกใช้ Linux Ubuntu 24.04 LTS ซึ่งเป็นรุ่นล่าสุด (2) Network Firewall เลือกใช้ ufw หรือ iptables ที่มาพร้อมระบบปฏิบัติการ (3) WAF เลือกใช้ ModSecurity v3 ทำงานร่วมกับ Nginx (4) HTTP Server เลือกใช้ Nginx ซึ่งทำงานร่วมกับ ModSecurity V3 (5) ระบบ CMS เลือกใช้ WordPress ซึ่งมีผู้ใช้งานมากที่สุด (6) ระบบฐานข้อมูลใช้ MySQL ซึ่งมีสัดส่วนผู้ใช้งานมากและรองรับการทำงานร่วมกับ WordPress (7) ระบบป้องกันการโจมตีแบบ brute-force attack เลือกใช้ Fail2Ban และ (8) ระบบ Log rotate สำหรับบริหารจัดการบันทึกของระบบ (Log files) ให้หมุนเวียนอยู่ในขนาดและระยะเวลาที่เหมาะสม ทั้งหมดนี้ถูกติดตั้งบนระบบปฏิบัติการ Linux ซึ่งมีประสิทธิภาพสูง มีเสถียรภาพในการทำงาน และได้รับการสนับสนุนจากชุมชนผู้ใช้โอเพนซอร์สในวงกว้าง และกำหนดแนวคิดการออกแบบสถาปัตยกรรมแบบ Single-node Multi-layer Defense ตามที่แสดงในภาพที่ 1 โดยใช้เครื่องแม่ข่ายเพียงเครื่องเดียว และรองรับการป้องกันหลายชั้น (defense in depth)

ภาพที่ 1. สถาปัตยกรรม Single-node Multi-layer Defense



การทดสอบเพื่อประเมินประสิทธิภาพของมาตรการใช้การทดสอบแบบ Unit test แยกเป็นส่วน และทดสอบการทำงานจริงในภาพรวม แบบ Integration test โดยดำเนินการ Unit test ดังนี้ (1) ตรวจสอบช่องโหว่ด้วยเครื่องมือ Nessus บนระบบที่ไม่ได้ติดตั้ง Security Patch Update พบช่องโหว่ระดับ Critical 2 รายการ High 5 รายการ Medium 12 รายการ และ Low 2 รายการ เปรียบเทียบผลหลังติดตั้ง Security Patch Update พบว่าช่องโหว่ระดับ Critical, High และ Medium ถูกปิดครบถ้วน คงเหลือเพียงช่องโหว่ระดับ Low จำนวน 1 รายการ โดยการ Patch ทำให้องค์ประกอบปรับเป็นเวอร์ชันใหม่ขึ้น และ Nessus มี Visibility ในการตรวจสอบ Informationได้มากขึ้น แต่ไม่ใช่ช่องโหว่ของระบบ ดังแสดงในตารางที่ 2

ตารางที่ 2. ผลการตรวจสอบช่องโหว่หลังทำ Security Patch Update

Vulnerability	Before Patch	After Patch
Severity: Critical	2	0
Severity: High	5	0
Severity: Medium	12	0
Severity: Low	2	1
Severity: Information	50	51

(2) การทดสอบ WAF ป้องกันการโจมตีทางเว็บ ใช้เครื่องมือทดสอบ Burp Suite [13] เป็น Interception Proxy เพื่อกำหนดค่า HTTP request จำลองการโจมตี และตรวจสอบ HTTP response จากเว็บไซต์ พบว่าสามารถป้องกันการโจมตีพื้นฐาน เช่น Code injection, SQL injection, Cross-site Scripting (XSS), Path Traversal ได้ดังแสดงในภาพที่ 3

ภาพที่ 3. การจำลองโจมตีเว็บไซต์ด้วย Burp Suit Interception Proxy

(3) ทำ Integration test กับเว็บไซต์ที่ให้บริการจริง โดยติดตั้งระบบ WAF ที่หน้า Web Server เป็นเวลา 30 วัน และนำ log ของ WAF ไปวิเคราะห์ผลการทำงาน พบว่า WAF ตรวจจับและป้องกันการโจมตีได้ 103,384 เหตุการณ์ ดังแสดงในตารางที่ 3 โดยจำแนกเป็น DoS via PCRE (ReDoS) 64%, Path Traversal 25%, SQL Injection 7% และ Cross-site Scripting (XSS) 4% โดยประมาณ และยังตรวจพบการสแกนระบบ เช่น SQLmap, curl และ bot อีกด้วย

เพื่อให้ระบบที่ออกแบบได้รับการบำรุงรักษาและมีความมั่นคงปลอดภัยอย่างต่อเนื่องในระยะยาว ลดความเสี่ยงจากความผิดพลาดในการดูแลระบบของเจ้าหน้าที่ จึงนำเสนอแนวทางการบริหารจัดการแบบรวมศูนย์ โดยให้หน่วยงานส่วนกลางที่มีความพร้อมด้านทรัพยากรบุคคลและความเชี่ยวชาญ เป็นผู้สนับสนุนและบริหารจัดการระบบจากส่วนกลางผ่านอินเทอร์เน็ต

ตารางที่ 3. ผลวิเคราะห์ log จากการทดสอบกับเว็บไซต์จริง 30 วัน

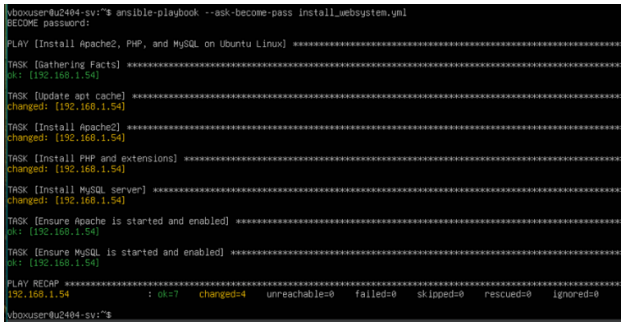
รูปแบบการโจมตี	ลักษณะในเชิงเทคนิค	ร้อยละ
DoS / PCRE Limits	พบข้อความ TX:MSC_PCRE_LIMITS_EXCEEDED แสดงว่า regex rule ถูกออกแบบให้ตรวจสอบลึกเกินไป จนเกิดภาระต่อระบบ อาจสะท้อนพฤติกรรม ReDoS	64.07
Path Traversal	พบรูปแบบ ../, etc/passwd จาก URI พยายามเข้าถึงไฟล์บนระบบปฏิบัติการ	24.51
SQL Injection	พบ payload ที่มีคำสั่ง SQL statement เช่น SELECT, UNION, SLEEP()	7.03
Cross-Site Scripting (XSS)	พบคำที่มักปรากฏใน XSS attack เช่น <script>, onerror=	4.24
Suspicious User-Agent / Bot	พบ HTTP request ที่มาจาก curl, SQLmap, python scripts	0.15

ดังแสดงในภาพที่ 4 ใช้ระบบ Ansible สั่งการติดตั้ง ปรับแต่ง และบำรุงรักษาระบบเว็บไซต์ เนื่องจากเป็นที่ใช้งานแพร่หลาย ไม่ต้องติดตั้ง agent ในเครื่องเป้าหมาย และเขียน Playbook script ได้ง่าย มีแนวทางการใช้ Ansible เพื่อบริหารจัดการแบบรวมศูนย์ ดังนี้ (1) ทำงานผ่าน SSH Key-based authentication ลดความเสี่ยงจากการใช้รหัสผ่านและการโจมตี Brute Force (2) มี Playbook สำหรับติดตั้งระบบเว็บไซต์อัตโนมัติ (HTTP Server, CMS, Database) (3) มี Playbook ตรวจสอบสถานะ และ Restart service ที่สำคัญของระบบเว็บไซต์ เช่น WAF, HTTP Server, MySQL (4) มีPlaybook สำหรับ Security Patch Management และอัปเดตระบบปฏิบัติการให้เป็นปัจจุบัน (5) มี Playbook สำหรับ Configuration Hardening ตามแนวทาง CIS Benchmark และ (6) มี Playbook สำหรับเรียกดูและวิเคราะห์ไฟล์ log

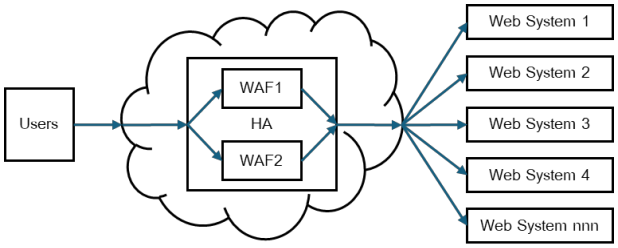
ภาพที่ 4. แผนภาพการออกแบบระบบ Ansible Automation

การใช้ระบบ Automation จะเปลี่ยนการติดตั้งองค์ประกอบต่าง ๆ ทีละส่วนแบบ manual ให้เป็นการสั่งการเครื่องเป้าหมายเป็นกลุ่มใหญ่แบบรวมศูนย์ได้พร้อม ๆ กันด้วยคำสั่งเพียงบรรทัดเดียว เช่นการทำ Security Patch Update การติดตั้งระบบเว็บไซต์ ดังแสดงในภาพที่ 5 และยังประหยัดทรัพยากรได้โดยใช้ WAF ขนาดใหญ่เพียงชุดเดียว หรือออกแบบให้มี Redundancy ทำหน้าที่เป็น Reverse Proxy ให้ทุกเว็บไซต์แบบรวมศูนย์ และทำการบริหารจัดการจากศูนย์กลางดังภาพที่ 6

ภาพที่ 5 ตัวอย่างหน้าจอ Ansible Playbook บน Control Node ขณะติดตั้ง Security Patch Update พร้อมติดตั้งองค์ประกอบของระบบเว็บไซต์



ภาพที่ 6 แผนภาพการออกแบบ WAF แบบรวมศูนย์และ Redundancy



5. สรุป

การศึกษาครั้งนี้ได้พัฒนามาตรการเสริมสร้างความมั่นคงปลอดภัยทางไซเบอร์สำหรับเว็บไซต์ของหน่วยงานขนาดเล็ก โดยเน้นการออกแบบระบบที่สามารถติดตั้งใช้งานได้จริงภายใต้ข้อจำกัดด้านบุคลากรและงบประมาณ โดยสรุปผลการศึกษาเป็นประเด็นที่สำคัญได้ดังนี้ (1) สามารถออกแบบสถาปัตยกรรมระบบแบบ Single-node Multi-layer Defense ที่ใช้เทคโนโลยี Open Source ทั้งระบบจาก Linux, Nginx, WordPress, MySQL, ModSecurity และ Fail2ban โดยไม่จำเป็นต้องมีเครื่องแม่ข่ายหลายเครื่อง ลดต้นทุนและความซับซ้อนในการดูแลระบบได้อย่างมีประสิทธิภาพ (2) สามารถปิดช่องโหว่ในระบบด้วยการติดตั้ง Security Patch Update โดยผลทดสอบจากเครื่องมือ Nessus พบช่องโหว่ระดับ Critical, High และ Medium ได้ถูกปิดทั้งหมด (3) สามารถตรวจจับป้องกันช่องโหว่และการโจมตีผ่านอินเทอร์เน็ตได้มากกว่า 100,000 ครั้ง ด้วย Firewall และ WAF (ModSecurity v3 + OWASP CRS) โดยพบรูปแบบการโจมตีที่สำคัญใน OWASP Top Ten เช่น SQL Injection, Path Traversal, และ Cross-site Scripting ได้มากกว่า 2,600 ครั้งในช่วงเวลา 30 วัน (4) สามารถรวบรวม log และตรวจสอบพฤติกรรมการโจมตีเชิงลึกได้ โดยจัดกลุ่มตามลักษณะเชิงพฤติกรรม เช่น ReDoS, SQLi, XSS และ Path Traversal ซึ่งสามารถนำไปใช้ปรับปรุงนโยบายการรักษาความมั่นคงปลอดภัย และออกแบบ Use case สำหรับระบบ Security Information and Event Management (SIEM) ได้ในอนาคต

ผลการวิจัยทั้งหมดแสดงให้เห็นว่าแนวทางที่นำเสนอสามารถเสริมสร้างความมั่นคงปลอดภัยให้กับระบบเว็บไซต์ของหน่วยงานขนาดเล็กได้อย่างมีประสิทธิภาพ ทั้งในด้านการลดช่องโหว่ การตรวจจับภัยคุกคาม และความสามารถในการบริหารจัดการจากศูนย์กลาง ซึ่งเป็นประเด็นสำคัญในการแก้ปัญหาเชิงโครงสร้างของหน่วยงานภาครัฐในระดับพื้นที่ ลดความเสี่ยงจากช่องโหว่ด้านความมั่นคงปลอดภัยที่ตรวจพบได้ทั้งหมดในระดับ Critical, High และ Medium ข้อมูลจาก log ของระบบ WAF ยังแสดงให้เห็นถึงแนวโน้มของภัยคุกคามที่หน่วยงานระดับท้องถิ่นต้องเผชิญ โดยมีการโจมตีจำนวนมากที่มุ่งเน้นไปที่ช่องโหว่ที่มักเกิดขึ้นในระบบ CMS ที่ได้รับความนิยม เช่น WordPress ซึ่งแสดงให้เห็นว่ามาตรการที่ออกแบบตามแนวทาง NIST Cybersecurity Framework และ OWASP Top Ten มีความครอบคลุมและมีประสิทธิภาพ การเลือกใช้ซอฟต์แวร์โอเพนซอร์ส เช่น ModSecurity, Fail2ban และ iptables ร่วมกับแนวทางการกำหนดค่าอย่างเป็นระบบ ทำให้ระบบสามารถตอบสนองต่อการโจมตีรูปแบบต่าง ๆ ได้โดยอัตโนมัติ

การใช้ Ansible เพื่อการบริหารจัดการแบบรวมศูนย์ มีประสิทธิภาพในการลดเวลาและความซับซ้อนในการดำเนินงาน และสามารถดำเนินการต่าง ๆ ไปยังเครื่องแม่ข่ายพร้อม ๆ กันอย่างรวดเร็วและไม่เกิดข้อผิดพลาด สนับสนุนแนวทางการบริหารจัดการจากศูนย์กลาง (Centralized Management) ที่เหมาะสมกับโครงสร้างการกำกับดูแลของหน่วยงานรัฐในประเทศไทย สามารถกระจายการติดตั้งระบบไปยังหน่วยงานในพื้นที่ห่างไกลได้โดยใช้ทรัพยากรบุคลากรจากส่วนกลางเพียงจำนวนน้อย

จากการวิเคราะห์เชิงเปรียบเทียบก่อนและหลังการติดตั้ง พบว่าระบบสามารถเพิ่มระดับความมั่นคงปลอดภัยของเว็บไซต์ในเชิงเทคนิคและสนับสนุนการบริหารจัดการได้ดี ดังแสดงในตารางที่ 4 สะท้อนให้เห็นถึงความเป็นไปได้ในการขยายผลระบบต้นแบบนี้ไปสู่หน่วยงานขนาดเล็กประเภทอื่น ๆ สามารถนำผลการวิจัยนี้ไปเป็นแนวทางเพื่อประยุกต์ใช้ในหน่วยงานขนาดเล็ก เช่น อบต. หรือสถานีตำรวจ เป็นต้น รวมถึงการสนับสนุนเชิงนโยบายให้มีการผลักดันการนำแนวทางนี้ไปใช้ในระดับกรมหรือกระทรวง โดยเฉพาะหน่วยงานกลางที่มีหน้าที่กำกับดูแลการปฏิบัติด้านเทคโนโลยีดิจิทัลของภาครัฐ

ตารางที่ 4. ผลเปรียบเทียบมาตรการรักษาความมั่นคงปลอดภัย

มาตรการควบคุม	ก่อนปรับปรุง	หลังปรับปรุง
ระบบ WAF	ไม่มีระบบ WAF	ModSecurity บน Nginx
ระบบ Network Firewall	ไม่มี Firewall	iptables/ufw filter traffic TCP 22, 80, 443
การทำ Security Patch Update ให้ระบบ	ทำแบบ Manual	ทำได้สะดวกโดยใช้ Ansible automation
การป้องกัน Brute force	ไม่มีการป้องกัน	ใช้ Fail2ban
การควบคุม root SSH	ไม่มีการป้องกัน	ไม่ให้ root login SSH
การเก็บบันทึกของระบบ (System Log)	ไม่มีการจัดการ / ตาม OS default	Log Rotation 90 วัน ตามกฎหมาย

อย่างไรก็ตาม มีข้อจำกัดของการวิจัยนี้จากการทดสอบบางส่วนที่อยู่ในสภาพแวดล้อมจำลองและมีการควบคุมในระดับหนึ่ง และยังไม่ได้ครอบคลุมกรณีที่ซับซ้อน เช่น การโจมตีจากภัยคุกคามขั้นสูง (Advanced Persistent Threat: APT) หรือการโจมตีที่มุ่งเป้าด้วย social engineering จึงควรมีการต่อยอดการศึกษาในอนาคตที่เน้นการประเมินความยืดหยุ่น

ของระบบในการเผชิญกับภัยคุกคามที่เปลี่ยนแปลงอย่างรวดเร็ว และศึกษาแนวทางการบูรณาการระบบความมั่นคงปลอดภัยเข้ากับนโยบายภาครัฐอย่างเป็นระบบ

กิตติกรรมประกาศ

การศึกษาเรื่องมาตรการเสริมสร้างความมั่นคงปลอดภัยทางไซเบอร์สำหรับระบบเว็บไซต์ของหน่วยงานขนาดเล็กนี้สำเร็จได้ด้วยการสนับสนุนจากหลายฝ่าย ขอขอบคุณนักวิชาการ คณาจารย์ และผู้เชี่ยวชาญทุกท่านที่ได้เผยแพร่ความรู้และแนวปฏิบัติที่เป็นประโยชน์ต่อสาธารณะ ผู้เขียนหวังเป็นอย่างยิ่งว่าผลการศึกษาจะถูกนำไปประยุกต์ใช้เพื่อเพิ่มความมั่นคงปลอดภัยและลดความเสี่ยงในการถูกโจมตีจากภัยคุกคามทางไซเบอร์ได้อย่างมีประสิทธิภาพและเป็นประโยชน์ต่อไป

เอกสารอ้างอิง

[1] สำนักงานสถิติแห่งชาติ, “การสำรวจการมีการใช้เทคโนโลยีสารสนเทศและการสื่อสารในครัวเรือน พ.ศ. 2567 (ไตรมาส 3),” กองสถิติพยากรณ์ สำนักงานสถิติแห่งชาติ, กรุงเทพมหานคร, 2024.

[2] สำนักประเมินคุณธรรมและความโปร่งใส, “คู่มือการประเมินคุณธรรมและความโปร่งใสในการดำเนินงานของหน่วยงานภาครัฐประจำปีงบประมาณ พ.ศ. 2567,” สำนักประเมินคุณธรรมและความโปร่งใส สำนักงานคณะกรรมการป้องกันและปราบปรามการทุจริตแห่งชาติ, นนทบุรี, 2024.

[3] UBUNTU Linux Documentation, “Ubuntu documentation,” [ออนไลน์]. Available: https://help.ubuntu.com/community/UFW.

[4] สำนักงานพัฒนารัฐบาลดิจิทัล (องค์การมหาชน), “มาตรฐานสำนักงานพัฒนารัฐบาลดิจิทัล (องค์การมหาชน) ว่าด้วยมาตรฐานเว็บไซต์ภาครัฐ เวอร์ชัน 3.0,” 29 September 2023. [ออนไลน์]. Available: https://standard.dga.or.th/standard/dga-std/8091/.

[5] National Institute of Standards and Technology, “The NIST Cybersecurity Framework (CSF) 2.0,” 26 February 2024. [ออนไลน์]. Available: https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.29.pdf.

[6] International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC), “Information security, cybersecurity and privacy protection — Information security management systems — Requirements,” ISO, Geneva, 2022.

[7] OWASP, “OWASP Top Ten,” September 2024. [ออนไลน์]. Available: https://owasp.org/www-project-top-ten/.

[8] NESSUS, “NESSUS 10.9 User Manual,” Tenable, 5 August 2025. [ออนไลน์]. Available: https://docs.tenable.com/nessus/Content/GettingStarted.htm.

[9] Center for Internet Security, “CIS Benchmark List,” [ออนไลน์]. Available: https://www.cisecurity.org/cis-benchmarks.

[10] A. H. P. และ D. G., “System Hardening using CIS Benchmarks,” ใน *International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)*, Chennai, India, 2024.

[11] Ansible project contributors, “Using Ansible Playbook,” [ออนไลน์]. Available: https://docs.ansible.com/ansible/latest/playbook_guide/.

[12] S. G. Brester, “Github,” [ออนไลน์]. Available: https://github.com/fail2ban/fail2ban.

[13] “Burp Suite Application Security,” [ออนไลน์]. Available: https://portswigger.net/burp.