



## มาตรฐานความปลอดภัยของเว็บแอปพลิเคชัน

OWASP หรือ Open Web Application Security Project คือ มาตรฐานความปลอดภัยของเว็บแอปพลิเคชัน จัดทำขึ้นโดยองค์กรไม่แสวงหาผลกำไรที่ให้ความรู้เพื่อทำให้ระบบคอมพิวเตอร์มีความปลอดภัยมากยิ่งขึ้นและเน้นวิจัยทางด้าน Web Application Security โดยจะมี community เกี่ยวกับเอกสาร เครื่องมือและเทคโนโลยีความปลอดภัยของเว็บแอปพลิเคชัน

OWASP Top 10 คือ โครงการหนึ่งของ OWASP ที่จัดอันดับ 10 ความเสี่ยงทางด้านความปลอดภัย ปัจจุบันมีหลายโครงการที่จัดอันดับความเสี่ยง ได้แก่ ความเสี่ยงของเว็บไซต์, ความเสี่ยงของแอปพลิเคชันบนโทรศัพท์, IoT, Cloud และอื่นๆ

- 1. Injection** คือ การโจมตีโดยแทรกคำสั่ง(Code) เข้าไปที่แอปพลิเคชันเป้าหมาย จะมีได้ทั้งแบบ SQL หรือ NoSQL หรือแทรกผ่านคำสั่งระดับ OS และ การแทรกคำสั่งผ่าน LDAP
- 2. Broken Authentication** คือ การโจมตีที่เกี่ยวกับการ login โดยใช้ Username และ Password หรือ การใช้ Login Key หรือ การใช้ session token
- 3. Sensitive Data Exposure** คือ การโจมตีแบบเจาะจงมาที่การส่งชุดข้อมูลส่วนตัว, บัตรประชาชน หรือ บัตรเครดิต แล้วปลอมแปลงข้อมูลนั้นๆ โดยสามารถทำได้ทาง Web Application และ APIs (Application Programming Interface[API]: ช่องทางเชื่อมต่อระหว่างเว็บไซต์หนึ่งไปยังอีกเว็บไซต์หนึ่ง)
- 4. XML External Entities (XXE)** คือ การโจมตีผ่าน SOAP Web Service โดยการส่งคำสั่งเข้าไปใน XML document ของเครื่องเป้าหมาย เพื่อให้ฝั่ง Server ประมาณผลการโจมตีทาง XXE
- 5. Broken Access control** คือ การโจมตีผ่านทางสิทธิ์ของยูสเซอร์ในระบบ เช่น การให้สิทธิ์บางอย่างกับ User มากเกินไป จนทำให้สามารถเจาะได้ทั้งระบบ หรือ User ตัวเองไม่มีสิทธิ์ในการแก้ไขบางอย่างจึงไปใช้ User ของคนอื่นที่มีสิทธิ์มากกว่ามาทำ จนทำให้เกิดช่องโหว่ในการโจมตีจาก Hacker

**6. Security misconfigurations** คือ การโจมตีผ่านช่องโหว่การตั้งค่าระบบ แบบใช้ค่าเริ่มต้น ทำให้สามารถคาดเดาการตั้งค่าของระบบได้ หรือ การตั้งค่าระบบไม่ปลอดภัย ใส่ผิดตำแหน่ง(Path) กำหนดสิทธิ์ไม่ถูกต้อง หรือ การใส่ค่า Config เพิ่มเติมจากที่มีอยู่เพื่อทำงานบางอย่างโดยไม่คำนึงถึงความปลอดภัย การไม่ใส่ HTTP Headers การส่ง Output app log ที่มีข้อมูลส่วนบุคคลมากเกินไป เพราะฉะนั้นจึงต้อง กำหนดค่าระบบปฏิบัติการและแอปพลิเคชันทั้งหมดอย่างปลอดภัย และ ติดตั้ง/อัปเดต ในเวลาที่เหมาะสม

**7. Cross Site Scripting(XSS)** เป็นช่องโหว่ที่เกิดจากฝั่งหน้า Web browser ของ แอปพลิเคชันที่ยอมรับให้ Hacker สามารถส่ง data หรือ script (Java Script) ไป run หรือ execute ที่ Web browser ได้

-Reflected XSS คือการโจมตีแบบชั่วคราวโดยการส่ง Script ไปพร้อมกับ link URL เมื่อเป้าหมาย คลิก link ตัว Script ก็จะทำงานทันที

-Stored XSS คือ การโจมตีแบบถาวร คือ การฝัง Script ไว้ที่ Web browser และ Script สามารถเข้าไปประมวลผลได้

-DOM-based XSS คือ การแก้ไข Framework ของ DOM เพื่อรับข้อมูลจาก User ที่ทำงานปกติ มาแสดงที่ หน้า Browser ที่เราต้องการ

**8. Insecure Deserialization** คือ การโจมตีโดยการส่ง Remote Code เข้ามาเพื่อให้มีช่องโหว่ที่เกิดการเปลี่ยนแปลงข้อมูลหรือถอดรหัสข้อมูลใน Application ที่ผิดพลาดการโจมตีทำได้ดังนี้สามารถลบหรือเปลี่ยนแปลงข้อมูลที่ถูกเข้ารหัสไว้ โจมตีแบบ Injection ได้ทั้งแบบ SQL หรือ NoSQL หรือแทรกผ่านคำสั่งระดับ OS และ LDAP เพิ่มสิทธิ์การเข้าถึงของ User ให้สามารถ Insert/Update/Delete หรือ ได้สิทธิ์สูงสุดเท่า Admin

**9. Using Components with Known Vulnerabilities** เป็นการโจมตีผ่านช่องโหว่ของการใช้ Software และ Hardware ที่ถูกรายงานและเผยแพร่ไว้เป็นสาธารณะแล้ว โดยมาตรฐานจะอ้างอิงตามช่องโหว่ที่ประกาศที่ CVE

**10. Insufficient Logging and Monitoring** เป็นการโจมตีจากช่องบันทึก และ แสดงผล log ของการโจมตีที่ไม่เพียงพอสำหรับการวิเคราะห์หาความเสี่ยง และการตรวจสอบที่ไม่ครอบคลุมถึงเส้นทางการโจมตี จากกลุ่ม Hacker เช่น Hacker ใช้วิธีหลากหลายรูปแบบในการเจาะระบบ โดยการโจมตีนั้นๆจะมี access log รวมถึงวิธีการเข้าถึง ถ้าฝั่งของ Admin Server ไม่เข้ามาตรวจสอบ log ว่ามีใครพยายามเข้ามาในระบบบ้าง และไม่ทำการลบเส้นทางหรือ History log ของ Hacker ที่ทำไว้ ก็จะทำให้ Hacker โจมตีระบบได้มากยิ่งขึ้น

### ประโยชน์และความจำเป็น

1. เพื่อตรวจสอบช่องโหว่พื้นฐานก่อนเผยแพร่เว็บไซต์ สำหรับป้องกันการโจมตีจากผู้ไม่ประสงค์ดี
2. รักษาความปลอดภัยการเข้าถึงฐานข้อมูล การร้องขอข้อมูล การตั้งค่า การยืนยันตัวตน และการสื่อสาร
3. มีการเข้ารหัสของข้อมูลและป้องกันการถูกขโมยข้อมูล
4. ตรวจสอบความถูกต้องของข้อมูลที่ป้อนเข้ามา ให้ข้อมูลอยู่ในรูปแบบที่เหมาะสม
5. สามารถปกป้องข้อมูลจากทุกที่โดยเฉพาะข้อมูลที่สำคัญ เช่น รหัสผ่าน เลขบัตรเครดิต บันทึกรประวัติทางการแพทย์ ข้อมูลส่วนตัว เป็นต้น
6. ลดความเสี่ยงในการถูกโจมตี

### สามารถดูรายละเอียดเพิ่มเติมได้ที่

- (1) OWASP Top Ten : <https://owasp.org/www-project-top-ten/>
- (2) OWASP : <https://hrod-it.ipst.ac.th/2021/02/15/owasp/>
- (3) OWASP Top 10 2021 – 10 อันดับต้องเช็ค เพื่อเพิ่มความปลอดภัยให้เว็บ แอปพลิเคชัน :  
<https://www.cyfence.com/article/owasp-top-10-2021/>
- (4) OWASP Top 10 Web Application 2020 : <https://www.4x-treme.com/owasp-top-10-web-application-2020/>
- (5) Web App Security 101 ทำอย่างไรให้เว็บปลอดภัยโปรแกรมเมอร์ต้องรู้ :  
<https://www.cyfence.com/article/web-app-security-101-how-to-keep-the-web-safe-programmers-need-to-know/>