แอปพลิเคชันสำหรับตรวจจับและป้องกันการโจมตีด้วยการเปลือยเอสเอสแอล

Application to Detect and Protect against SSL Stripping Attacks

สมนึก พ่วงพรพิทักษ์, ณัฐวุฒิ ศรีวิบูลย์
Somnuk Puangpronpitag, Nattavut Sriwiboon
Received: 18 May 2017; Accepted: 8 August 2017

บทคัดย่อ

HTTP Over TLS (HTTPS) เป็นโพรโทคอลที่ใช้เพื่อป้องกันการโจมตีดักจับข้อมูล โดยเข้ารหัสข้อมูลให้เป็นความลับ อย่างไรก็ตามการโจมตีแบบเปลือยเอสเอสแอล (SSL Stripping Attack) สามารถโจมตี HTTPS และดักจับข้อมูลได้ ดังนั้นงานวิจัยนี้จึงได้ประเมินปัญหาและเสนอแนวคิดทางเทคนิคในการแก้ไขปัญหา โดยมีการพัฒนาแอปพลิเคชันตรวจจับและป้องกันการโจมตี HTTPS ผลการทดสอบประสิทธิภาพและประสิทธิผลของแอปพลิเคชันแสดงให้เห็นว่ามีประสิทธิภาพรองรับการทำงานทั้งบนคอมพิวเตอร์ส่วนบุคคลและอุปกรณ์ Smartphone รวมถึงมีประสิทธิภาพป้องกันการโจมตีด้วยวิธี SSL Stripping Attack

คำสำคัญ: เอชทีทีพีเอส การโจมตีแบบเปลือยเอสเอสแอล การปลอมแปลงโพรโทคอลเออาร์พี

Abstract

HTTP Over TLS (HTTPS) is a protocol for protection eavesdropping attacks. It has encryption to provide data confidentiality. However, SSL Stripping Attack has been deployed by attackers to bypass the HTTPS and eavesdropping. This research has evaluated the problem and proposed a technique to solve it. After that, a solution has been implemented the application to provide detection and protection against HTTPS attack. The evaluation of our application shows that it effectively supports both the personal computer and smartphone. The experimental results have revealed favorable features regarding our solution.

Keywords: HTTPS SSL Stripping Attack ARP Poisoning

บทนำ

ระบบเว็บไซต์เป็นระบบทางเทคโนโลยีสารสนเทศที่มีการให้บริการระบบต่าง ๆ บนเครือข่ายอินเทอร์เน็ต โดยเฉพาะระบบเว็บไซต์ที่ให้บริการธนาคารทางอินเทอร์เน็ต ที่มีการให้บริการอย่างแพร่หลาย โดยระบบเว็บไซต์ส่วนใหญ่ทำงานบน Hyper Text Transfer Protocol (HTTP)[1] ซึ่งข้อมูลที่ส่งผ่านระหว่างไคลเอนต์กับเซิร์ฟเวอร์จะอยู่ในรูปของ Clear Text เมื่อเว็บไซต์ถูกโจมตีด้วยวิธีแทรกกลางการสื่อสาร (Man In The Middle Attack: MITM)[2] ผลของการโจมตีคือผู้โจมตีสามารถดักจับข้อมูลที่สื่อสารบน HTTP ซึ่งอาจเป็นข้อมูลสำคัญของผู้ใช้เช่นชื่อบัญชีผู้ใช้และรหัสผ่าน โดย HTTP Over TLS (HTTPS)[3] เป็นรูปแบบการสื่อสารข้อมูลบนระบบเว็บไซต์ที่เสนอเพื่อแก้ไขปัญหาการทำงานของ HTTP ที่ไม่มีความมั่นคงในการใช้งาน โดยการนำ Transport Layer Security (TLS)[4] เป็นโพรโทคอลที่ใช้ในการเข้ารหัสข้อมูลการสื่อสารระหว่างไคลเอนต์กับเซิร์ฟเวอร์ทำให้การสื่อสารมีความมั่นคงในการใช้งาน โดยรูปแบบการทำงานของ HTTPS มีการสร้างช่องทางการสื่อสารที่ปลอดภัยและเปลี่ยนข้อมูลต้นฉบับ (Clear Text) ให้เป็นข้อมูลที่ถูกเข้ารหัส (Cipher Text) เพื่อป้องกันปัญหาการโจมตีดักจับและเปลี่ยนแปลงข้อมูล

อย่างไรก็ตามการใช้งานเว็บไซต์บนโพรโทคอล HTTPS สามารถถูกโจมตีด้วยเทคนิคและวิธีการต่าง ๆ ที่มีพื้นฐานมาจากการโจมตีแบบ MITM โดยในปี ค.ศ.2009 Marlinspike และคณะ[5] ได้เสนอการโจมตีด้วยวิธีเปลือยเอสเอสแอล (SSL Stripping Attack) โดยใช้เครื่องมือ SSL Strip เป็นวิธีที่ใช้โจมตีเว็บไซต์ที่ทำงานบน HTTPS โดยผลการโจมตีบนเบราว์เซอร์ของเหยื่อไม่แสดงข้อความแจ้งเตือนเนื่องจากการทำงานบนเบราว์เซอร์ส่งข้อมูลบน HTTP ทำให้ผู้ใช้ไม่สามารถตรวจสอบการทำงานของระบบเว็บไซต์ว่าดำเนินบน HTTPS ที่ปลอดภัยหรือไม่

จากปัญหาการโจมตี SSL Stripping Attack ซึ่งเป็นปัญหาการโจมตี HTTPS ที่ส่งผลเสียต่อระบบเว็บไซต์เนื่อง

จากเบราว์เซอร์ไม่สามารถตรวจสอบและแสดงผลเมื่อมีการโจมตี มีงานวิจัยก่อนหน้านี้ที่เสนอวิธีการป้องกันแต่ประสบปัญหาเช่นความสะดวกในการใช้งานและประสิทธิภาพในการป้องกัน

ดังนั้นในงานวิจัยนี้จึงเสนอระบบป้องกันการโจมตีเว็บไซต์ที่ทำงานบน HTTPS จากการโจมตีแบบ MITM และการโจมตีด้วยวิธี SSL Stripping Attack มีการประเมินวิธีการของงานวิจัยก่อนหน้านี้และเสนอวิธีการป้องกันโดยการออกแบบและพัฒนาระบบต้นแบบที่สามารถตรวจจับและป้องกันการโจมตีโพรโทคอล HTTPS ในรูปของแอปพลิเคชันที่สามารถติดตั้งบนอุปกรณ์คอมพิวเตอร์ (Personal Computer: PC) และอุปกรณ์ Smartphone โดยมีการพัฒนาระบบต้นแบบและทดสอบระบบที่เสนอในงานวิจัยนี้เพื่อแสดงให้เห็นถึงประสิทธิภาพในการใช้งาน ประสิทธิภาพในการป้องกันการโจมตีเว็บไซต์จากการโจมตีแบบ MITM และการโจมตีด้วยวิธี SSL Stripping Attack

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

1. Secure Socket Layer (SSL)

Secure Socket Layer (SSL) ถูกพัฒนาขึ้นโดยบริษัทเน็ตสเคป (Netscape Communications) ในปี ค.ศ. 1994 เพื่อเป็นโพรโทคอลที่ให้บริการสำหรับการเข้ารหัสข้อมูลและการพิสูจน์ตัวตนในการสื่อสารระหว่างเซิร์ฟเวอร์ (Server) และไคลเอนท์ (Client) ทำให้การสื่อสารมีความปลอดภัยมากยิ่งขึ้น ซึ่งปกติแล้วข้อมูลที่ส่งไปมาระหว่างกันจะไม่มีการเข้ารหัสข้อมูลทำให้การดักจับข้อมูลเป็นไปได้โดยง่ายต่อมา SSL เวอร์ชัน 3.0 ในปี ค.ศ. 1997 ได้ถูกทำการทดสอบแล้วพบว่าไม่มีความปลอดภัยในการใช้งาน จึงมีการพัฒนา SSL ให้เกิดเป็นมาตรฐานกลางของโพรโทคอลบนอินเทอร์เน็ตที่นำไปสู่การออกแบบ Transport Layer Security (TLS) ที่ถูกกำหนดให้เป็นมาตรฐานอย่างเป็นทางการ ตามเอกสาร RFC 2246[6] ในปี ค.ศ.1999 โดยเป็นมาตรฐานการสื่อสารของ Internet Engineering Task Force (IETF) แต่โพรโทคอล TLS นั้นจะมีความแตกต่างจาก SSL เพียงเล็กน้อย โดยส่วนที่แตกต่างกันก็คือ Version, Cipher Suite, Alert Protocol, Handshake Protocol และ Record Protocol เป็นต้น ปัจจุบัน TLS ถูกพัฒนามาถึงเวอร์ชัน 1.2 ซึ่งเริ่มมีการใช้งานมาตั้งแต่ปี ค.ศ. 2008 เอกสาร RFC 5246[7]

2. HTTP Over TLS (HTTPS)

จากการเสนอ Transport Layer Security (TLS) ที่เป็นโพรโทคอลที่ทำงานในระดับของ Transport Layer ดังนั้นในการนำใช้งานจึงนำมาประยุกต์ใช้กับโพรโทคอลระดับแอปพลิเคชัน เช่น IMAP, POP3, LDAP, SSH และ FTP ในงานวิจัยนี้ได้ศึกษาถึงการนำ TLS มาประยุกต์ใช้กับโปรแกรมประยุกต์เว็บ ซึ่งปกติใช้โพรโทคอล HTTP ในการสื่อสารข้อมูลระหว่างกัน โดยข้อมูลที่รับส่งนี้ก็จะอยู่ในรูปแบบของข้อความปกติ (Clear Text) ที่มีความเสี่ยงต่อการถูกโจมตีด้วยการดักจับข้อมูล (Sniffing) จึงมีการเสนอให้นำ SSL มาประยุกต์ใช้ร่วมกับ HTTP สร้างเป็นการสื่อสารรูปแบบใหม่ขึ้นเรียกว่า HTTP Over TLS (HTTPS) เกิดจากแนวความคิดในการสร้างช่องทางการสื่อสารข้อมูลบนเว็บไซต์ที่มีการเข้ารหัสเพื่อความปลอดภัยในระหว่างการใช้งานเว็บไซต์ ซึ่งบนโพรโทคอล HTTPS นี้ เว็บเบราว์เซอร์จะแสดง URL ของเว็บไซต์เป็น https://www.sitename.com แทนซึ่งเดิมในโพรโทคอล HTTP ที่เคยใช้งานนั้น URL ดังกล่าวคือ http://www.sitename.com

3. ภัยคุกคามการใช้งาน HTTPS

ในปี ค.ศ. 2009 Marlinspike[8] ได้เสนอวิธีการโจมตี SSL ด้วยวิธี SSL Stripping Attack ในงาน Blackhat Conference โดยใช้ PYTHON scripts ซึ่งต่อมาเป็นเครื่องมือที่ถูกติดตั้งในระบบปฏิบัติการ Kali Linux ซึ่งเป็นระบบปฏิบัติการที่ถูกนำไปใช้งานด้านการทดสอบเจาะและประเมินความปลอดภัยของระบบเครือข่าย รวมถึงการถูกนำไปใช้โดยผู้ที่ไม่ประสงค์ดีเช่นเดียวกัน

การเปลือยเอสเอสแอล หรือ SSL Stripping Attack มีรูปแบบการโจมตีโดยอาศัยวิธีโจมตีแบบแทรกกลางการสื่อสารร่วมกับวิธีการโจมตีแบบ SSL Stripping Attack ควบคู่กันโดยการโจมตีเว็บไซต์ที่มีทำงานบนโพรโทคอล HTTPS ที่ถูกเข้ารหัสในระหว่างการสื่อสารนั้น เมื่อเหยื่อถูกโจมตีเว็บเบราว์เซอร์ก็จะถูกบังคับให้ใช้โพรโทคอล HTTP ไม่ปลอดภัยในการสื่อสารโดยข้อมูลต่างๆ ที่เหยื่อส่งไปที่เว็บเซิร์ฟเวอร์จะถูกส่งผ่านไปยังเครื่องของผู้โจมตีก่อน ซึ่งผู้โจมตีสามารถดักจับข้อมูลของเหยื่อได้ เนื่องจากข้อมูลที่สื่อสารกันบน HTTP จะอยู่ในรูปของ Clear Text ที่สามารถเข้าใจได้จากนั้นการทำงานในขั้นตอนต่อไปของ SSL Stripping Attack ก็จะนำข้อมูลของเหยื่อมาเข้ารหัสด้วยโพรโทคอล HTTPS แล้วส่งต่อไปที่เว็บเซิร์ฟเวอร์ ด้วยเหตุนี้จึงทำให้เว็บเซิร์ฟเวอร์ไม่สามารถตรวจสอบได้ว่าการสื่อสารที่เกิดขึ้นระหว่างเซิร์ฟเวอร์และไคลเอนท์มีการสื่อสารบนโพรโทคอล HTTP หรือ HTTPS รวมถึงผลของการโจมตีที่เว็บเบราว์เซอร์ของเครื่องเหยื่อก็จะไม่สามารถตรวจสอบหรือแสดงข้อความแจ้งเตือนความผิดพลาดได้อันมาเนื่องจากเครื่องของเหยื่อสามารถสื่อสารกับเว็บเซิร์ฟเวอร์ได้ตามปกติ เพียงแต่เป็นการสื่อสารที่ถูกบังคับให้อยู่บนโพรโท-คอล HTTP แทนที่จะเป็นโพรโทคอล HTTPS ซึ่งมีความปลอดภัยดัง Figure 1

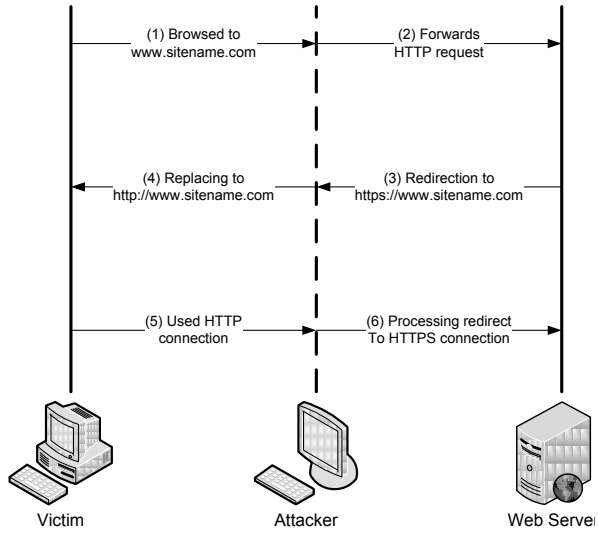


Figure 1 SSL Stripping Attack

4. งานวิจัยที่เกี่ยวข้อง

Fung และคณะ[9] ได้เสนอ SSLock เมื่อปี ค.ศ. 2010 เป็นวิธีบังคับใช้โพรโทคอล SSL กับเว็บไซต์ โดยใช้กระบวนการสำหรับพิจารณาแบ่ง Domain Name เพื่อกำหนดรายชื่อเว็บไซต์ที่บังคับใช้โพรโทคอล SSL ซึ่งการใช้งานผู้พัฒนาเว็บไซต์ต้องติดตั้ง API ที่พัฒนาขึ้นในงานวิจัย SSL Lock เสนอเพื่อตอบกลับ HTTP Header ชื่อ SSLock-Candidates ซึ่งทำหน้าที่จัดเก็บค่าของ Domain Name เช่น gmail.com พร้อมด้วย Javascript ที่ใช้อ่านค่า HTTP Header จากนั้นในการทำงานบนเว็บเบราว์เซอร์ของไคลเอนท์จะประมวลผล Script บนเว็บเบราว์เซอร์แล้วส่งคำขอไปที่เว็บเซิร์ฟเวอร์ด้วย URL ที่ถูกเปลี่ยนแปลงโพรโทคอลเป็น HTTPS แล้วเช่น https://secure.gmail.com

Cheng และคณะ[10] ได้เสนองานวิจัยเมื่อปี ค.ศ. 2010 โดยวิเคราะห์การทำงานของ HTTPS ในกระบวนการ SSL Handshake เพื่อนำผลการวิเคราะห์ใช้สำหรับกำหนดเกณฑ์การประเมินการทำงานของโพรโทคอล HTTPS โดยประเมินการโจมตี HTTPS ด้วยวิธี SSL Sniff และ SSL Strip ซึ่งในงานวิจัยนี้ได้สรุปปัญหาความมั่นคงของ HTTPS ได้ดังนี้ 1) User's habits คือพฤติกรรมการใช้งานเว็บไซต์ของผู้ใช้โดยปกติจะเรียกผ่านเว็บเบราว์เซอร์โดยระบุชื่อเว็บไซต์เช่น www.test.com ซึ่งพฤติกรรมของผู้ใช้จะไม่ระบุโพรโทคอลดังนั้นในกรณีนี้เว็บเบราว์เซอร์สื่อสารไปที่เว็บเซิร์ฟเวอร์ บนโพรโทคอล HTTP ทำให้ผู้โจมตีใช้วิธี SSL Strip โจมตีเหยื่อได้สำเร็จ 2) Application in practice คือในขั้นตอนการพัฒนาเว็บไซต์มีการกำหนดคำสั่ง Link หรือปุ่มกดเพื่อเชื่อมไปยัง Link ต่าง ๆ ด้วยโพรโทคอล HTTP ซึ่งในกรณีนี้คล้ายกับกรณี User's habits ที่เว็บเบราว์เซอร์สื่อสารไปที่เว็บเซิร์ฟเวอร์บนโพรโทคอล HTTP ทำให้ผู้โจมตีใช้วิธี SSL Strip โจมตีเหยื่อได้สำเร็จ เมื่อได้ผลวิเคราะห์ปัญหาความมั่นคงของ HTTPS แล้วงานวิจัยนี้เสนอให้ใช้วิธีป้องกัน 3 วิธีคือ 1) Static ARP วิธีนี้เป็นการกำหนด Static ARP บนเครื่องผู้ใช้ผ่านคำสั่งในระบบปฏิบัติการซึ่งวิธีการนี้สามารถป้องกันการโจมตีด้วยวิธี ARP Spoof ได้ 2) การสังเกต EVSSL certificate ซึ่ง EVSSL certificate คือแถบที่แสดงการใช้งานโพรโทคอล HTTPS บนเว็บเบราว์เซอร์ในกรณีที่ EVSSL certificate ไม่ปรากฏบนเว็บเบราว์เซอร์แสดงว่าเว็บไซต์ที่กำลังเข้าใช้งานสื่อสารด้วยโพรโท-คอล HTTP ซึ่งอาจมีสาเหตุมาจากการถูกโจมตีด้วยวิธี SSL Strip 3) Two-way authentication วิธีนี้เป็นวิธีที่สามารถพิสูจน์ตัวจริงระหว่างเซิร์ฟเวอร์และไคลเอนท์ได้ โดยปกติการใช้งาน SSL มีเพียงไคลเอนท์เท่านั้นที่สามารถพิสูจน์ตัวจริงได้ว่ากำลังสื่อสารกับเซิร์ฟเวอร์จริงหรือไม่ สำหรับวิธีการ Two-way authentication เป็นวิธีที่ เซิร์ฟเวอร์สามารถใช้เพื่อพิสูจน์ตัวจริงได้ว่ากำลังสื่อสารข้อมูลกับไคลเอนท์จริงหรือไม่

Fung และคณะ[11] ได้เสนอ HTTPSLock เมื่อปี ค.ศ. 2010 เป็นกลไกบังคับใช้โพรโทคอล HTTPS รองรับการทำงานกับเว็บไซต์ที่ใช้ Valid Certificate เท่านั้นพัฒนาเครื่องมือโดยใช้ภาษา JavaScript ของรับการตรวจเว็บไซต์ 2 กรณีประกอบด้วย 1) สำหรับตรวจสอบตรวจสอบ Certificate ที่ถูกใช้งานกับเว็บไซต์นั้นๆ ในกรณีที่เว็บไซต์ใช้งาน Invalid Certificate ระบบจะไม่อนุญาตให้ใช้งานเว็บไซต์และแสดงข้อความแจ้งเตือนความมั่นคงบนเว็บเบราว์เซอร์และ 2) สำหรับเว็บไซต์ที่มีการใช้งาน Valid Certificate โดยปกติการสื่อสารข้อมูลบนโพรโทคอล HTTPS การแสดงผลชื่อ URL ที่ช่อง Address Bar บนเว็บเบราว์เซอร์จะแสดงผลเป็น HTTPS อย่างไรก็ตามในกรณีที่เว็บไซต์แสดงโพรโทคอลเป็น HTTP อาจเนื่องมาจากการถูกโจมตีด้วยวิธี SSL Strip กลไกของ HTTPSLock จะแสดงการแจ้งเตือนความมั่นคงบนเว็บเบราว์เซอร์

Puangpronpitag และ Sriwiboon[12] ได้เสนอ ISAN-HTTPS Enforcer ในปี ค.ศ. 2012 โดยพัฒนาให้อยู่ในรูปของ API สำหรับติดตั้งบนเว็บเซิร์ฟเวอร์ด้วยภาษา JavaScript รองรับการทำงานกับทุกเว็บเบราว์เซอร์ที่รองรับการใช้งานภาษา JavaScript โดยรายละเอียดของขั้นตอนกลไกการทำงานมีดังนี้

1. ไคลเอนท์ส่งคำขอไปที่เว็บเซิร์ฟเวอร์โดยปกติแล้วผู้ใช้จะไม่ระบุโพรโทคอลใน URL ซึ่งการประมวลผลของเว็บเบราว์เซอร์จะคำขอด้วยโพรโทคอล HTTP
2. เว็บเซิร์ฟเวอร์ประมวลผลแล้วตอบกลับข้อมูลบนโพรโทคอล HTTPS

3. เมื่อไคลเอนท์ได้รับข้อความตอบกลับจากเว็บเซิร์ฟเวอร์ที่ประกอบด้วยเนื้อหาของเว็บไซต์และ ISAN-HTTPS Enforcer API ซึ่งเป็น File พัฒนาด้วยภาษา JavaScript ชื่อ ISAN-HTTPSEnforcer.js โดยการทำงานของ API จะตรวจสอบว่าโพรโทคอลที่ใช้สื่อสารข้อมูลกับระหว่างไคลเอนท์กับเว็บเซิร์ฟเวอร์เป็น HTTPS หรือไม่ ถ้าไม่ใช่ ISAN-HTTPS Enforcer จะประมวลผลอัตโนมัติให้เว็บเบราว์เซอร์ส่งคำขอไปที่เว็บเซิร์ฟเวอร์อีกครั้งโดยการระบุโพรโทคอล HTTPS บน URL

4. เว็บเซิร์ฟเวอร์ประมวลผลแล้วตอบกลับข้อมูลบนโพรโทคอล HTTPS

5. เมื่อ ISAN-HTTPS Enforcer API ตรวจสอบโพรโทคอลที่ใช้สื่อสารข้อมูลกับระหว่างไคลเอนท์กับเว็บเซิร์ฟเวอร์เป็น HTTPS ระบบจึงเริ่มขั้นตอน SSL Handshake โดยรายละเอียดการทำงานแสดงดัง Figure 2

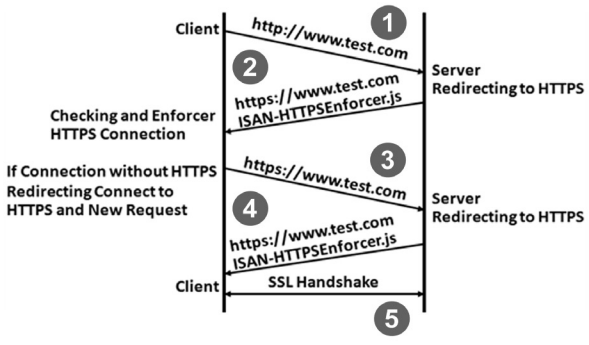


Figure 2 How ISAN-HTTPS Enforcer work

อย่างไรก็ตามจากงานวิจัยของสมนึก พ่วงพรพิทักษ์และอภิรักษ์ ทูลธรรม[13] ได้ประเมินวิธีที่ถูกใช้ในการป้องกันการโจมตีจากเทคนิค SSL Stripping Attack แสดงให้เห็นว่าระบบ ISAN-HTTP Enforcer ถึงแม้จะสามารถป้องกันการโจมตีได้บนทุกแพลตฟอร์ม และมีความเป็นมิตรกับผู้ใช้สูง แต่กลับถูกทำลายการป้องกันได้ด้วยการแก้ไขโค้ดที่ใช้ในการโจมตีเพียงไม่กี่บรรทัด ด้วยการอาศัยความสามารถพื้นฐานในการค้นหาและแทนที่ค่าของ Regular Expression บน Python ผลของการทดลองแสดงให้เห็นว่า ISAN-HTTP Enforcer ไม่สามารถบรรลุผลในด้านประสิทธิภาพของการป้องกัน

Hodges และคณะ[14] ได้เสนอ HTTP Strict Transport Security (HSTS) ถูกประกาศเป็นมาตรฐานเมื่อ เดือนพฤศจิกายน ปี ค.ศ. 2012 โดยได้รับแนวคิดจาก Force-HTTPS[15] โดย HSTS เป็นกลไกสำหรับป้องกันการโจมตี HTTPS เช่นการโจมตีด้วยวิธี SSL Stripping Attack เป็นต้น โดยมีหลักการทำงาน คือส่งคำขอ ไปที่เว็บเซิร์ฟเวอร์ด้วยโพร-โทคอล HTTPS สำหรับเว็บไซต์ที่มีรายชื่ออยู่ใน HSTS List[16] และสำหรับเว็บไซต์ที่ไม่มีรายชื่อใน HSTS List ในกรณีที่เว็บเซิร์ฟเวอร์รองรับการทำงานโพรโทคอล HTTPS ผู้ใช้สามารถกำหนดชื่อเว็บไซต์ในเครื่องมือที่ติดตั้งบนเว็บเบราว์เซอร์เช่นใน Google Chrome เป็นต้น หรือสามารถพัฒนาระบบเว็บไซต์ให้รองรับการทำงาน HSTS ได้โดยการกำหนดคำสั่งในขั้นตอนการพัฒนาเว็บไซต์หรือกำหนดคำสั่งบนเว็บเซิร์ฟเวอร์[17] เพื่อเพิ่ม HTTP Header ชื่อ Strict-Transport-Security ที่ประกอบด้วยสอง Parameter หลักคือ max-age และ IncludeSubdomains โดย max-age จะแสดงจำนวนเวลาหน่วยเป็นวินาทีที่เว็บเบราว์เซอร์ควรส่งคำขอไปที่เซิร์ฟเวอร์ด้วย HTTPS ตัวอย่างเช่นเว็บไซต์ A.com มีการกำหนด max-age=1000 หมายถึงใน 1000 วินาทีเว็บเซิร์ฟเวอร์จะคำขอไปที่เซิร์ฟเวอร์ด้วย HTTPS ส่วน IncludeSubdomains คือการกำหนดให้ subdomains รองรับการทำงาน HSTS เมื่อ HTTP Header ตอบกลับถึงเครื่องไคลเอนท์แล้วเว็บเบราว์เซอร์จะอ่านค่า HTTP Header ชื่อ Strict-Transport-Security แล้วบันทึก Domain Name ในเครื่องไคลเอนท์เมื่อผู้ใช้เข้าใช้เว็บไซต์ในภายหลังขั้นตอนการส่งคำขอไปที่เว็บเซิร์ฟเวอร์จะถูกบังคับใช้โพรโทคอล HTTPS โดยการทำงานของ HSTS รองรับการทำงานกับเว็บเซิร์ฟเวอร์

Selvi[18] ได้เสนอวิธีการโจมตี HSTS ใน Blackhat Conference 2014 โดยมีการวิเคราะห์ข้อดีและข้อเสียของ HSTS แล้วเสนอวิธีการโจมตีโดยอาศัยการโจมตีที่ Network Time Protocol (NTP)[19] ของเครื่องเหยื่อเพื่อเปลี่ยนแปลงเวลาบนเครื่องเหยื่อให้เพิ่มมากขึ้นทำให้การทำงานของ HSTS มองว่าค่าใน Parameter ที่ชื่อ max-age หมดอายุ ดังนั้นเมื่อเว็บเบราว์เซอร์เปรียบเทียบเวลาที่ระบุใน max-age กับเวลาปัจจุบันบนเครื่องเหยื่อแล้วได้ผลว่าเวลาใน max-age หมดอายุ การประมวลผลของเว็บเบราว์เซอร์ก็จะไม่ทำตามเงื่อนไขของ HSTS แล้วผู้โจมตีสามารถใช้วิธีโจมตีแบบ SSL Stripping Attack ได้สำเร็จ โดยในงานวิจัยได้พัฒนาเครื่องมือชื่อ Delorean แล้วทดสอบโจมตีทั้งในระบบปฏิบัติการ Ubuntu Linux, Fedora Linux, Mac OS X Lion, Mac OS X Mavericks และ Microsoft Windows และกับเว็บเบราว์เซอร์ได้แก่ Safari, Firefox และ Google Chrome

อภิรักษ์ ทูลธรรมและสมนึก พ่วงพรพิทักษ์[20] ได้ดำเนินการประเมินปัญหาของการโจมตีด้วย SSL Stripping Attack ในมิติที่เป็นเชิงลึก บนแพลตฟอร์มของฮาร์ดแวร์ ระบบปฏิบัติการ และโปรแกรมเว็บเบราว์เซอร์ที่หลากหลาย ซึ่งพบว่าการโจมตีด้วยเทคนิคดังกล่าวเป็นปัญหาที่น่ากลัวของเว็บไซต์ที่ยังเป็นประเด็นปัญหาซึ่งต้องการการแก้ไข ในการ

ทดลองพบว่า แม้เว็บไซต์จะได้รับการปกป้องด้วย SSL และใช้เทคนิคการป้องกันรูปแบบอื่นเข้าร่วมก็ยังอาจเกิดปัญหาการโจมตีขึ้น แม้แต่รูปแบบของการใช้งาน SSL ในลักษณะต่างๆ อย่างเช่นการใช้ SSL เพื่อปกป้องระบบในตลอดขบวนการของการให้บริการ การใช้ SSL ปกป้องเฉพาะหน้า Login หรือการใช้ SSL ปกป้องกระบวนการส่งข้อมูลอยู่เบื้องหลังรวมถึงความแตกต่างทางด้านแพลตฟอร์มของฮาร์ดแวร์อย่างเว็บเบราว์เซอร์และระบบปฏิบัติการก็ไม่อาจหลีกเลี่ยงจากการโจมตีของเทคนิคการเปลือยเอสเอสแอลนี้ได้ ซึ่งจากการทดลองเพื่อประเมินปัญหาบน Test-bed ที่หลากหลายครั้งนี้ชี้ให้เห็นถึงพฤติกรรมและผลลัพธ์ของการโจมตีที่ละเอียดกว่าที่ผ่านมาอันเป็นแนวทางในการออกแบบวิธีรับมือกับ SSL Stripping Attack ที่มีประสิทธิภาพและครอบคลุมทุกแพลตฟอร์มในอนาคต ซึ่งการศึกษานี้เป็นเพียงการประเมินปัญหาการโจมตี SSL ในเชิงลึกและนำเสนอเพียงแนวคิดในการโจมตีวิธีป้องกันปัญหาเท่านั้น ยังไม่ได้ทำการทดสอบเพื่อประเมินวิธีการป้องกันการโจมตีจริง

ACIS Professional Center[21] ได้เสนอ SSL Strip Guard ซึ่งเป็นแอปพลิเคชันเพื่อแจ้งเตือนผู้ใช้จากการถูกโจมตีด้วยวิธี SSL Stripping Attack บน Tablet และ Smartphone บนระบบปฏิบัติการ Android และ Apple iOS ทำการป้องกันผู้ใช้งานในระหว่างการเชื่อมต่อเข้ากับ Rogue Wi-Fi, HOT-SPOT และ Access Point สาธารณะโดยลักษณะของการทำงานก็คือ ในขณะเชื่อมต่อกับเครือข่ายไร้สายใดๆ SSL Strip Guard จะทำการติดต่อไปยังเซิร์ฟเวอร์ที่ถูกกำหนดเอาไว้ ซึ่งมีการป้องกันโดยใช้งานโพรโทคอล SSL จากนั้นจึงทำการตรวจสอบการติดต่อสื่อสารไคลเอนต์กับเซิร์ฟเวอร์ว่ายังคงใช้โพรโทคอล HTTPS อยู่หรือไม่ ด้วย JavaScript ถ้าหากไม่ได้ใช้งานโพรโทคอล HTTPS นั่นหมายความว่าเครือข่ายไร้สายดังกล่าวไม่มีความมั่นคงปลอดภัยเนื่องจากถูกโจมตีจาก SSL Stripping Attack แต่ถ้าโพรโทคอลที่ตอบกลับมาเป็น HTTPS แสดงว่าเครือข่ายไร้สายดังกล่าวไม่ได้ถูกโจมตีจาก SSL Stripping Attack ซึ่งผู้ใช้สามารถใช้งานได้อย่างปลอดภัย ดังแสดงใน Figure 3 แต่วิธีการป้องกันการโจมตีของ SSL Strip Guard นี้ยังมีข้อเสียคือสามารถทำได้เพียงการตรวจสอบการโจมตีเท่านั้น แต่ไม่สามารถป้องกันการโจมตีด้วยวิธี SSL Stripping Attack ได้ และจะต้องมีการเรียกใช้งานโปรแกรม SSL Strip Guard เพื่อทำการตรวจสอบเครือข่ายไร้สายที่เชื่อมต่อก่อนการใช้งาน รวมถึงลักษณะการทำงานของโปรแกรมเป็นแบบเว็บเบราว์เซอร์ซึ่งมีการใช้งานไฟล์ HTML ที่ถูกกำหนดเอาไว้แล้ว โดยให้ติดต่อไปยัง https://www.acisonline.net/iPhone/acis.html จากนั้นจึงทำการ Redirection ไปเรียกใช้งาน JavaScript ในไฟล์ check.html เพื่อตรวจสอบสถานะ ซึ่งการทำงานลักษณะดังกล่าวสามารถปลด Tag ที่ใช้ในการป้องกันออกได้ หรือหลอกโดยการ Strip ทุก site ยกเว้น site ที่ SSL Strip Guard กำหนดไว้ ดังเปิดเผยโดย สมนึก พ่วงพรพิทักษ์และอภิรักษ์ ทูลธรรม[13]

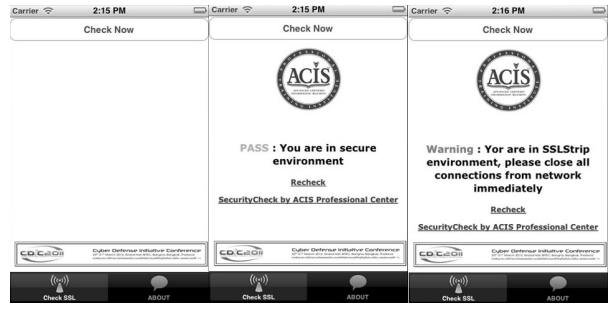


Figure 3 ACIS SSL Strip Guard

วิธีดำเนินการวิจัย

กระบวนการวิจัยแบ่งออกเป็น 3 ส่วนประกอบด้วย (1) การวิเคราะห์ปัญหาความมั่นคงของงานวิจัยก่อนหน้านี้ (2) การออกแบบและพัฒนาแอปพลิเคชัน (3) การประเมินประสิทธิภาพแอปพลิเคชันตรวจจับและป้องกันการโจมตี SSL Stripping Attack

1. การวิเคราะห์ปัญหาความมั่นคงของงานวิจัยก่อนหน้านี้

ผลการประเมินวิธีแก้ไขการโจมตี HTTPS จากเทคนิค SSL Stripping Attack สามารถสรุปได้ดังใน Table 1

Table 1 Previous Solution Discussion

Solution	Experimental Results
HSTS	- ป้องกันการโจมตีเว็บไซต์ที่อยู่ใน List ถ้านอกเหนือจาก List นั้นต้อง add เพิ่มหรือผู้ดูแลระบบต้องกำหนดการใช้งาน HSTS หรือผู้พัฒนาเว็บไซต์ต้องกำหนดการใช้งาน HSTS ในขั้นตอนการพัฒนาเว็บไซต์ - ป้องกันแฮ็กเกอร์ได้ทั้งระดับ Script Kiddies และแฮ็กเกอร์ที่มีความเชี่ยวชาญ ซึ่งสามารถแก้ไขโค้ด Python เพื่อโจมตีได้ - มีงานวิจัยของ Selvi[18] ที่เสนอเพื่อโจมตี HSTS
ISAN-HTTPS Enforcer	- ป้องกันเว็บไซต์จากการโจมตี SSL - มีความเป็นมิตรสูง เนื่องจากผู้ใช้ไม่ต้องทำอะไรเลยระบบจะจัดการป้องกันให้โดยอัตโนมัติ - ป้องกันได้เพียง Script Kiddies เนื่องจากแฮ็กเกอร์ที่มีความเชี่ยวชาญสามารถแก้ไขโค้ด Python เพื่อโจมตีได้ - รองรับการทำงานได้ทุกแพลตฟอร์ม
SSL Strip Guard	- สามารถทำได้เพียงตรวจสอบการโจมตี SSL เท่านั้น ไม่สามารถป้องกันการโจมตีได้ - มีความเป็นมิตรกับผู้ใช้ที่ต่ำ เนื่องจากต้องติดตั้งโปรแกรมและผู้ใช้ต้องให้ความร่วมมือในการตรวจสอบ - ป้องกันได้เพียง Script Kiddies เนื่องจากแฮ็กเกอร์ที่มีความเชี่ยวชาญสามารถแก้ไขโค้ด Python เพื่อโจมตีได้ - ถูกพัฒนาให้รองรับเพียงแพลตฟอร์มที่เป็น Mobile Device

2. ออกแบบและพัฒนาระบบ

ในงานวิจัยนี้เสนอการออกแบบระบบแอปพลิเคชันตรวจจับและป้องกันการโจมตี SSL Stripping Attack และเสนอแนวคิดสำหรับการพัฒนาระบบให้รองรับการทำงานทั้ง PC Desktop และอุปกรณ์ Smartphone โดยมีรายละเอียดการออกแบบระบบดังนี้

1) แอปพลิเคชันตรวจจับการโจมตี SSL Stripping Attack แอปพลิเคชันตรวจจับการโจมตี SSL Stripping Attack ในงานวิจัยนี้ได้กำหนดชื่อแอปพลิเคชันคือ ISAN - Detect SSL Strip มีหน้าที่ตรวจสอบการทำงานของเครื่องผู้ใช้ว่ากำลังใช้งานอินเทอร์เน็ตปลอดภัยจากการโจมตีด้วยวิธีแทรกกลางการสื่อสารและวิธี SSL Stripping Attack หรือไม่ โดยการออกแบบแอปพลิเคชันตรวจจับการโจมตี SSL Stripping Attack มีรายละเอียดดังนี้

Public Key Server	คือ	กุญแจสาธารณะของเซิร์ฟเวอร์
Private Key Server	คือ	กุญแจลับของเซิร์ฟเวอร์
Session Key	คือ	กุญแจที่ถูกสุ่มขึ้นโดยไคลเอนต์
Local User List	คือ	รายชื่อเว็บไซต์ที่ถูกเก็บบนเครื่องมือตรวจจับการโจมตีโพรโทคอล HTTPS
Server URL List	คือ	รายชื่อเว็บไซต์ที่ถูกส่งมาจากเซิร์ฟเวอร์โดยผ่านการเข้ารหัสด้วย Session Key
User URL	คือ	ชื่อเว็บไซต์ที่ผู้ใช้กำหนด
Log	คือ	ข้อมูลที่ถูกเข้ารหัสด้วย Session Key เพื่อรายงาน ไปที่เซิร์ฟเวอร์ในกรณีถูกโจมตี

โดยหลักการทำงานของแอปพลิเคชันตรวจจับการโจมตี SSL Stripping Attack มีรายละเอียดดังนี้

1. เมื่อแอปพลิเคชันตรวจจับการโจมตี SSL Stripping Attack ทำงานจะตรวจสอบว่าตอนนี้การใช้งานอินเทอร์เน็ตปลอดภัยจากการโจมตีด้วยวิธีแทรกกลางการสื่อสาร ถ้าไม่ปลอดภัยก็จะแสดงข้อความเตือนผู้ใช้ แต่ถ้าระบบปลอดภัยจากการแทรกกลางการสื่อสาร ระบบตรวจจับการโจมตีโพรโท-คอล HTTPS มีกระบวนการทำงานดังต่อไปนี้

2. ไคลเอนต์สมัครสมาชิกและเข้าสู่ระบบโดยกระบวนการส่งข้อมูลไปที่เซิร์ฟเวอร์ในขั้นตอน Login ไคลเอนต์จะสร้าง Session Key แล้วเข้ารหัสข้อมูล Username, Password และ Session Key ด้วย Public Key Server

3. เซิร์ฟเวอร์ถอดรหัสข้อมูลด้วย Private Key Server จะได้เป็น Username และ Password เพื่อใช้ในกระบวนการพิสูจน์ตัวตนเข้าใช้งานระบบและ Session Key เพื่อใช้ในการเข้ารหัสข้อมูล

4. เซิร์ฟเวอร์เข้ารหัส Server URL List ด้วย Session Key แล้วตอบกลับมาที่ ไคลเอนต์

5. ไคลเอนต์ถอดรหัส ข้อมูลได้ผลลัพธ์จากการถอดรหัสเป็น Server URL List

6. ไคลเอนต์ร้องขอไปที่ชื่อเว็บไซต์ที่ถูกสุ่มมาจาก Server URL List และ User URL ที่เป็นชื่อเว็บไซต์ที่ถูกกำหนดโดยผู้ใช้ ซึ่งในขั้นตอนนี้ หากระบบไม่สามารถติดต่อกับเซิร์ฟเวอร์ได้ ระบบจะไปอ่านค่า Local User List แล้วสุ่มชื่อเว็บไซต์เพื่อใช้ในการตรวจสอบโพรโทคอล

7. เมื่อแอปพลิเคชันตรวจจับการโจมตี SSL Stripping Attack ได้รับค่าตอบกลับแล้วตรวจสอบว่า Local

User List, Server URL List และ User URL ทำงานบนโพรโทคอล HTTPS ก็จะรายงานผลว่าการใช้งานอินเทอร์เน็ตปลอดภัยจากการโจมตีด้วยวิธีแทรกกลางการสื่อสารและวิธี SSL Stripping Attack แต่หากมี URL ที่ส่งเพื่อตรวจสอบรายงานผลว่าเว็บไซต์ไม่ได้ทำงานบนโพรโทคอล HTTPS ระบบก็จะส่ง Log ไปที่เซิร์ฟเวอร์ประกอบด้วย IP Address ของเครื่องที่ถูกโจมตี วันเวลา ชื่อเว็บไซต์ที่ถูกโจมตีและชื่อผู้ใช้

2) แอปพลิเคชันป้องกันการโจมตี SSL Stripping Attack

จากการศึกษาการทำงานของโพรโทคอล HTTPS เมื่อผู้ใช้เข้าใช้งานเว็บไซต์จะร้องขอด้วยโพรโทคอล HTTP เมื่อเซิร์ฟเวอร์ได้รับการร้องขอจากไคลเอนต์จึงเริ่มขั้นตอน SSL Handshake แล้วเปลี่ยนโพรโทคอลในการสื่อสารเป็น HTTPS จากการวิเคราะห์กระบวนการทำงานของโพรโทคอล HTTPS และวิธีการโจมตีแบบ SSL Stripping Attack พบว่าหลักการทำงานของ SSL Strip จะโจมตีสำเร็จเมื่อผู้ใช้ร้องขอข้อมูลไปที่เซิร์ฟเวอร์บนโพรโทคอล HTTP

ดังนั้นแนวคิดในการออกแบบแอปพลิเคชันป้องกันการโจมตี SSL Stripping Attack โดยในงานวิจัยนี้ได้กำหนดชื่อแอปพลิเคชันคือ ISAN - Secure Browsing HTTPS มีหน้าที่ทำให้การใช้งานเว็บไซต์ปลอดภัยจากการโจมตีเว็บไซต์ที่ทำงานบนโพรโทคอล HTTPS โดยหลักการทำงานของระบบ มีขั้นตอนดังต่อไปนี้

1. ผู้ใช้กำหนดชื่อเว็บไซต์
2. การทำงานของระบบจะกำหนดโพรโทคอล HTTPS ให้เว็บไซต์ที่ผู้ใช้กำหนดแล้วระบบจะเรียกใช้งานเว็บเบราว์เซอร์เพื่อเข้าใช้งานเว็บไซต์

3. การกำหนดเกณฑ์สำหรับประเมินประเมินประสิทธิภาพ

การประเมินประสิทธิภาพการทำงานของระบบป้องกันการโจมตี HTTPS ด้วยวิธี SSL Stripping Attack รูปแบบใหม่มีรายละเอียดดังนี้

1) ประสิทธิผลในการป้องกันการโจมตี SSL Stripping Attack ทดสอบโดยใช้เครื่องมือ SSL Strip ที่ติดตั้งในระบบปฏิบัติการ Kali Linux

2) ความเป็นมิตรต่อผู้ใช้ (User Friendliness) ระบบการป้องกันมีความยุ่งยากหรือซับซ้อนในการใช้งานหรือไม่

3) ระบบสามารถป้องกันการโจมตี HTTPS จากผู้โจมตีระดับ Script Kiddies ที่ใช้เครื่องมือ SSL Strip หรือระบบสามารถป้องกันการโจมตีจากวิธีที่ผู้โจมตีมีทักษะในการปรับปรุงโค้ดโปรแกรมที่ใช้ในการจู่โจมที่เรียกว่า Bypass HTTPS

4) ความครอบคลุมในแพลตฟอร์ม (Platform Coverage) ระบบรองรับทำงานบนแพลตฟอร์มฮาร์ดแวร์ระบบปฏิบัติการ และเว็บเบราว์เซอร์ที่มีความหลากหลายได้หรือไม่

เครื่องมือและสภาพแวดล้อมที่ถูกกำหนดให้เป็น Test-bed ประกอบด้วย

1) เครื่องคอมพิวเตอร์ส่วนบุคคล (Personal Computer : PC) หรือ PC Desktop ที่ใช้ทดสอบระบบต้นแบบคือ Intel ® Core ™ 2 Duo 2.66 GHz RAM 4 GB ติดตั้งระบบปฏิบัติการ Kali Linux 64 bit และ Windows โดยทดสอบในระบบปฏิบัติการ Windows 8

2) Macbook Air คือ Intel Core i5 1.6 GHz RAM 4 GB ติดตั้งระบบปฏิบัติการ OS X

3) Samsung Galaxy Tab 10.1 ติดตั้งระบบปฏิบัติการ Android 4.4

4) Web Browser ใช้ Google Chrome ที่สามารถรองรับทั้งระบบปฏิบัติการ Windows, Linux, OS X, iOS และ Android

ผลการวิจัย

จากการศึกษางานวิจัยก่อนหน้านี้ที่พัฒนาเครื่องมือ SSL Strip Guard โดยได้วิเคราะห์และสรุปผลการทดสอบโจมตี SSL Strip Guard ว่ามีปัญหาในกระบวนการตรวจสอบโพรโทคอลในการสื่อสาร ที่มีการกำหนดชื่อเว็บไซต์ของเซิร์ฟเวอร์เพียง 1 ชื่อเว็บไซต์เท่านั้นในการตรวจสอบ เมื่อผู้โจมตีแก้ไขคำสั่งในเครื่องมือ SSL Strip เพื่อไม่โจมตีชื่อเว็บไซต์ที่กำหนดใน SSL Strip Guard เครื่องมือดังกล่าวก็จะรายงานผลว่า การใช้งานอินเทอร์เน็ตปลอดภัย

โดยผลการพัฒนาระบบในงานวิจัยนี้ พัฒนาให้รองรับทั้งระบบปฏิบัติการที่ทำงานบน PC Desktop อย่างเช่น ระบบปฏิบัติการ Windows และระบบปฏิบัติการ Linux และรองรับการทำงานระบบปฏิบัติการที่ทำงานบนอุปกรณ์ Tablet และ Smartphone อย่างเช่นระบบปฏิบัติการ Android

1. ผลการพัฒนา ISAN - Detect SSL Strip

จากปัญหาของ SSL Strip Guard ในงานวิจัยนี้ได้ออกแบบระบบตรวจจับการโจมตีโพรโทคอล HTTPS โดยมีแนวคิดในการส่งชื่อเว็บไซต์ในลักษณะ Dynamic เพื่อตรวจสอบโพรโทคอลในการสื่อสาร โดยผลการพัฒนา ISAN - Detect SSL Strip แบ่งออกเป็น 5 ส่วนคือ

1) ส่วนการตรวจสอบการแทรกกลางการสื่อสาร

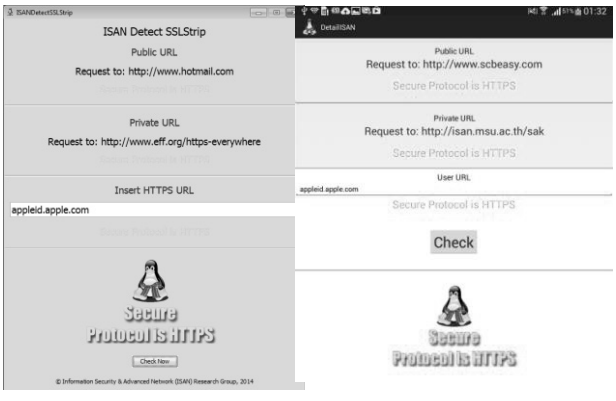
2) ส่วนสมัครสมาชิก ส่วนเข้าใช้งานระบบและกำหนด Session Key

3) ส่วนของเซิร์ฟเวอร์ที่ทำหน้าที่ในการเข้ารหัส URL ที่เก็บอยู่บนเซิร์ฟเวอร์ก่อนส่งไปที่ไคลเอนต์

4) ส่วนของไคลเอนต์ที่ทำหน้าที่ในการตรวจสอบโพรโทคอลในการสื่อสาร

5) ส่วนของเว็บไซต์แสดงสถิติ URL และหมายเลข IP Address ของเหยื่อที่ ถูกโจมตีรวมถึงเป็นส่วนของการใช้งานเพื่อให้ผู้ดูแลระบบกำหนดชื่อเว็บไซต์ที่ใช้ในกระบวนการตรวจสอบโพรโทคอลในการสื่อสาร

โดยแสดงตัวอย่างส่วนการทำงานของแอปพลิเคชันที่ทดสอบบน PC Desktop ดัง Figure 4 (a) และบนอุปกรณ์อุปกรณ์ Smartphone ที่รองรับระบบปฏิบัติการ ANDROID ดัง Figure 4 (b)



(a) (b)

Figure 4 (a) ISAN - Detecting SSL Strip on a PC (b) ISAN - Detecting SSL Strip on a Smartphone

2. ผลการพัฒนา ISAN - Secure Browsing HTTPS

การพัฒนาแอปพลิเคชัน ISAN - Secure Browsing HTTPS บนระบบปฏิบัติการที่ทำงานบนอุปกรณ์ PC Desktop และ Smartphone ในงานวิจัยนี้ มีการพัฒนาแอปพลิเคชันต้นแบบโดยกับระบบปฏิบัติการ Android โดยใช้หลักการพัฒนาจากระบบที่ต้นแบบที่พัฒนาบนเครื่องคอมพิวเตอร์ส่วนบุคคล เรื่องจากการพัฒนาแอปพลิเคชันบนระบบปฏิบัติการ Android ใช้ใช้หลักการพัฒนามาจากภาษา Java โดยแสดงตัวอย่างส่วนการทำงานของแอปพลิเคชันที่ทดสอบบน PC Desktop ดัง Figure 5 และบนอุปกรณ์อุปกรณ์ Smartphone ที่รองรับระบบปฏิบัติการ Andr oid ดัง Figur e 6

Figure 5 ISAN-Secure Browsing on a PC

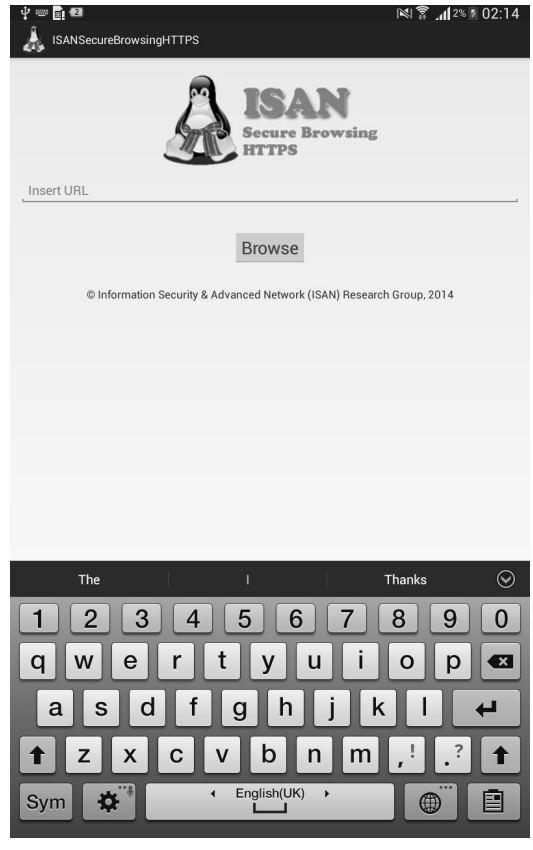


Figure 6 ISAN-Secure Browsing **on a** Smartphone

ผลการประเมินประสิทธิภาพ

1. ผลการทดสอบ ISAN - Detect SSL Strip

ผลการทดสอบ ISAN-Detect SSL Strip เปรียบเทียบกับ SSL Strip Guard เพื่อเปรียบเทียบประสิทธิภาพการตรวจจับการโจมตี SSL Stripping Attack แสดงดัง Table 2 และการทดสอบส่งข้อมูลในขั้นตอนการตรวจสอบการโจมตี HTTPS เพื่อแสดงประสิทธิผลการใช้งานแอปพลิเคชันทั้งบน PC Desktop และ Smartphone โดยทดสอบส่งข้อมูลจำนวน 30 ครั้ง ต่อ 1 การทดสอบ ผลการทดสอบแสดงค่าเฉลี่ยของเวลาหน่วยเป็น millisecond (ms) แสดงดัง Figure 7

Table 2 Performance of ISAN - Detect SSL Strip

วิธีการโจมตี	ISAN - Detect SSL Strip		SSL Strip Guard
	ระบบปฏิบัติการ		
	Windows	Android	Android
SSL Strip	✓	✓	✘
Bypass HTTPS	✓	✓	✓

✓ป้องกันการโจมตีได้ ✘ไม่สามารถป้องกันการโจมตี

จากการโจมตีเครื่องมือที่ใช้ตรวจจับการโจมตี SSL Stripping Attack ที่ถูกเสนอในงานวิจัยนี้เปรียบเทียบกับ SSL Strip Guard พบว่าเมื่อผู้โจมตีใช้วิธี Bypass HTTPS เพื่อไม่โจมตีด้วยวิธี SSL Strip กับเว็บไซต์ที่ใช้ตรวจสอบการทำงานโพรโทคอล HTTPS ของเครื่องมือ SSL Strip Guard ระบบก็จะรายงานว่าเครือข่ายปลอดภัย ซึ่งสำหรับ ISAN - Detect SSL Strip ทั้งวิธี SSL Strip และวิธี Bypass HTTPS ไม่สามารถโจมตีได้เนื่องจากเว็บไซต์ที่ใช้ตรวจสอบการทำงานโพรโทคอล HTTPS ถูกสุ่มขึ้นมาเพื่อตรวจสอบและถูกกำหนดรายชื่อเว็บไซต์โดยเซิร์ฟเวอร์ รวมถึง ISAN - Detect SSL Strip สามารถตรวจสอบการโจมตีด้วยวิธี ARP Spoof ได้ และเครื่องมือสามารถรองรับการทำงานได้ทั้งระบบปฏิบัติการ Android และ Windows

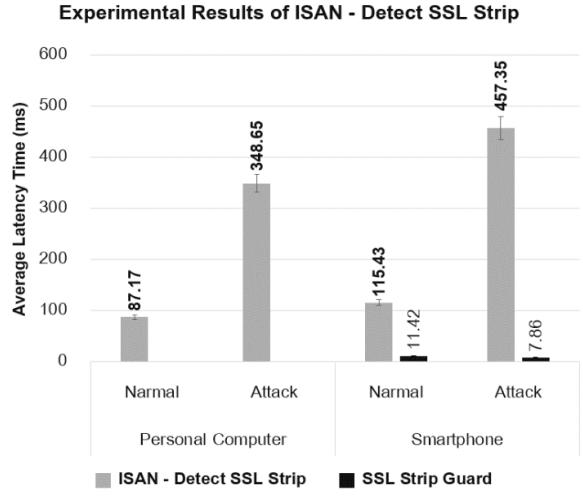


Figure 7 Experimental Results

การรับส่งข้อมูลรหัสผ่านของ ISAN - Detect SSL Strip ใช้เวลาในการรับส่งข้อมูลในการตรวจจับการโจมตีโพรโทคอล HTTPS มากกว่า SSL Strip Guard เนื่องจาก ISAN - Detect SSL Strip มีการส่งข้อมูลไปตรวจสอบจำนวน 3 เซิร์ฟเวอร์ และ ISAN - Detect SSL Strip มีส่วนของการทำงานในระบบตรวจสอบการโจมตีด้วยวิธี ARP Spoof อย่างไรก็ตามสามารถแก้ไขปัญหาของ SSL Strip Guard ที่สามารถถูกโจมตีโดยการเพิ่มโค้ดที่ใช้ในการ Bypass การตรวจจับการโจมตี SSL Stripping Attack และสามารถตรวจสอบการโจมตีด้วยวิธี ARP Spoof

2. ผลการทดสอบ ISAN - Secure Browsing HTTPS

ผลการทดสอบโจมตี ISAN - Secure Browsing HTTPS เปรียบเทียบกับ HSTS และ SSL Strip Guard โดยทดสอบโจมตี Stripping Attack และ Bypass HTTPS ด้วยเครื่องมือ SSL Strip ซึ่งจากผลการทดสอบพบว่า ISAN - Secure Browsing HTTPS สามารถป้องกันการโจมตีเว็บไซต์ที่สื่อสารบน HTTPS จากการโจมตีด้วยวิธี SSL Strip และ Bypass HTTPS เนื่องจากมีการกำหนดให้สื่อสารบนโพรโทคอล HTTPS ตั้งแต่ขั้นตอนการร้องขอของไคลเอนต์ และสามารถนำไปใช้ได้กับทุกเว็บเบราว์เซอร์แก้ปัญหาการใช้งาน HSTS ที่รองรับเฉพาะกับเว็บเบราว์เซอร์ Chrome, Firefox และ Opera เท่านั้น โดยการใช้งาน ISAN - Secure Browsing มีความปลอดภัยทั้งในระบบปฏิบัติการ Windows และระบบปฏิบัติการ Android รายละเอียดแสดงดัง Table 3

Table 3 Solution Comparison

ระบบที่ใช้ทดสอบ	วิธีการโจมตี	
	SSL Strip	Bypass HTTPS
ISAN Secure Browsing	✓	✓
ISAN-HTTPS Enforcer Javascript API	✓	✘
HSTS	✓*	✓*

✓ป้องกันการโจมตีได้ ✘ไม่สามารถป้องกันการโจมตี

* ป้องกันการโจมตีได้บางกรณี

สรุปผลและข้อเสนอแนะ

ระบบเว็บไซต์ใช้ช่องทางการสื่อสารที่มีความมั่นคงบนโพรโทคอล HTTPS อย่างไรก็ตามจากที่ได้กล่าวมาแล้ว HTTPS ถูกโจมตีเพื่อดักจับข้อมูลสำคัญที่สื่อสารระหว่างผู้ใช้กับเว็บเซิร์ฟเวอร์ โดยผู้โจมตีอาศัยวิธีการโจมตีแบบ MITM และการโจมตีด้วยวิธี SSL Stripping Attack

ดังนั้นในงานวิจัยนี้จึงเสนอวิธีการเพื่อป้องกันการโจมตี HTTPS จากการโจมตีแบบ MITM และการโจมตีด้วยวิธี SSL Stripping Attack โดยผลการออกแบบระบบ

แอปพลิเคชัน ISAN - Detect SSL Strip และ ISAN - Secure Browsing HTTPS แสดงให้เห็นว่าสามารถตรวจจับและป้องกันการโจมตีเว็บไซต์ที่สื่อสารบน HTTPS และผลการทดสอบแสดงให้เห็นว่าระบบมีประสิทธิภาพในการใช้งานรองรับการทำงานทั้ง PC Desktop และ Smartphone รวมถึงมีประสิทธิภาพในการป้องกันการโจมตีเว็บไซต์จากการโจมตีแบบ MITM และการโจมตีด้วยวิธี SSL Stripping Attack

กิตติกรรมประกาศ

โครงการนี้ได้รับการสนับสนุนการวิจัย งบประมาณรายได้คณะวิทยาการสารสนเทศ ประจำปีงบประมาณ 2559 มหาวิทยาลัยมหาสารคาม ขอขอบคุณเจ้าหน้าที่กรมสอบสวนคดีพิเศษ สำหรับข้อมูลคดีด้านการโจมตี HTTPS

เอกสารอ้างอิง

1. Fielding R, Gettys J, Mogul J, et al. Hypertext Transfer Protocol -- HTTP/1.1: RFC 2616, IETF June 1999.
2. Burkholder P. SSL Man-in-the-Middle Attacks.
3. Rescorla E. HTTP Over TLS: RFC 2818, IETF May 2000.
4. Dierks T, Rescorla E. The Transport Layer Security (TLS) Protocol Version 1.2: RFC 5246, IETF August 2008.
5. SSLsplit - transparent and scalable SSL/TLS interception.
6. Dierks T, Allen C. The TLS Protocol Version 1.0: IETF, January 1999.
7. Dierks T, Rescorla E. The Transport Layer Security (TLS) Protocol Version 1.2: IETF, August 2008.
8. Marlinspike M. New Tricks For Defeating SSL In Practice.
9. Fung A, Cheung K. SSLock: sustaining the trust on entities brought by SSL. Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security;pp. 204-213.
10. Cheng K, Gao M, Guo R. Analysis and Research on HTTPS Hijacking Attacks. Proceedings of the 2010 Second International Conference on Networks Security, Wireless Communications and Trusted Computing; pp. 223 - 226.
11. Fung A, Cheung K. HTTPSLock: Enforcing HTTPS in Unmodified Browsers with Cached Javascript. Proceedings of the 4th Network and System Security;pp. 269-274.
12. Puangpronpitag S, Sriwiboon N. Simple and Light-weight HTTPS Enforcement to Protect Against SSL Striping Attack. Proceedings of 4th International Conference on Computational Intelligence, Communication Systems and Networks;pp. 229-234.
13. สมนึก พ่วงพรพิทักษ์ และ อภิรักษ์ ทูลธรรม. Experimental Evaluation of SSL Stripping Attack Solutions. Information Technologies Journal;10:pp. 37-47.
14. Hodges J, Jackson C, Barth A. HTTP Strict Transport Security (HSTS): RFC 6797, IETF November 2012.
15. Jackson C, Barth A. ForceHTTPS: Protecting High-Security Web Sites from Network Attacks. Proceedings of the 17th International World Wide Web Conference;pp. 525-534.
16. Domain to include in HSTS list.
17. HTTP Strict Transport Security.
18. Selvi J. Bypassing HTTP Strict Transport Security. Proceedings of Black Hat Europe 2014.
19. Mills D, Delaware U, Burbank J, et al. Network Time Protocol Version 4: Protocol and Algorithms Specification: RFC 5905, IETF June 2010.
20. อภิรักษ์ ทูลธรรม และ สมนึก พ่วงพรพิทักษ์. The Evaluation of the SSL Stripping Attack Problem. Proceedings of The National Conference on Computer Information Technologies;pp. 43-48
21. SSLSTRIPGuard: ACIS Professional Center.