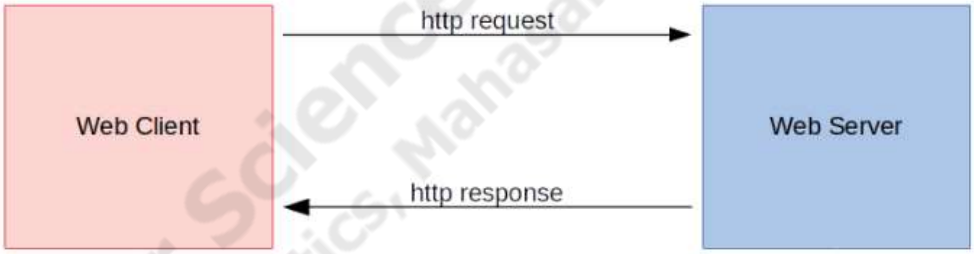
บทที่ 2
ทฤษฎี และงานที่เกี่ยวข้อง

2.1 งานวิจัยและทฤษฎีที่เกี่ยวข้อง

1) Hypertext Transfer Protocol (HTTP)

การสื่อสารบนเว็บไซต์ต่างๆ จำเป็นต้องอาศัยโพรโทคอลที่มีชื่อว่า Hypertext Transfer Protocol (HTTP) เป็นโพรโทคอลที่ใช้ในการสื่อสารข้อมูลบน browser ถูกกำหนดขึ้นโดย World Wide Web Consortium (W3C) และ Internet Engineering Task Force (IETF) โดยมีเอกสารสำคัญที่สุด คือ RFC 2616 ซึ่งเป็นข้อมูลเกี่ยวกับ HTTP Version 1.1 เป็นรุ่นที่ใช้กันอย่างกว้างขวางในปัจจุบัน HTTP Protocol ใช้ในการรับส่งข้อมูลระหว่างเครื่อง Client และเครื่อง Server โดยอาศัย Port 80 ในการรับ – ส่งข้อมูล ข้อมูลที่ส่งผ่านจะไม่มีการเข้ารหัส ในทางเทคนิคเรียกว่า Clear text การสื่อสารบน HTTP Protocol เป็นที่รู้กันโดยทั่วไปว่าการสื่อในรูปแบบนี้ไม่มั่นคง และไม่เป็นที่ยอมรับ แต่ก็ยังมีเว็บไซต์หลายแห่ง ที่สนับสนุนการใช้ HTTP Protocol อยู่ ทำให้ถูกดักจับข้อมูลและถูกโจมตีได้ง่าย



ภาพประกอบที่ 2.1 การรับ – ส่งข้อมูลบน HTTP Protocol

จากภาพประกอบที่ 2.1 การรับ – ส่งข้อมูลบน HTTPจะเห็นได้ว่า Client เรียกเว็บไซต์ปลายทางเป็น HTTP แล้ว Server ตอบกลับมาในรูปแบบ HTTP เช่นกัน การสื่อสารในรูปแบบนี้มีความมั่นคง เนื่องจากไม่มีการเข้ารหัส ในทางเทคนิค เรียกว่า Clear text

2) Hypertext Transfer Protocol Security (HTTPS)

Hypertext Transfer Protocol Security (HTTPS) คือ HTTP Protocol ที่ทำงานร่วมกับ Secure Socket Layer (SSL) Protocol ถูกกำหนดขึ้นในปีค.ศ. 1994 โดย Netscape Communications และถูกเผยแพร่อย่างเป็นทางการในเอกสารของ RFC 2818 ในปี ค.ศ. 2000 เป็นเทคโนโลยีการเข้ารหัสข้อมูล เพื่อให้เว็บไซต์ และแอพพลิเคชั่นต่างๆ สามารถรับส่งข้อมูลกันได้อย่างปลอดภัย และมีความน่าเชื่อถือ สามารถตรวจสอบความถูกต้องของข้อมูลที่รับส่งได้ ไม่ถูกปรับแก้ไข มีแหล่งที่มาต้นทางปลายทางอย่างถูกต้อง และไม่มีการเปลี่ยนแปลงข้อมูลระหว่างการรับส่งกันบน

เครือข่ายอินเตอร์เน็ต โดยใบรับรองความปลอดภัยนี้ถูกออกให้โดยหน่วยงานที่มีความน่าเชื่อถือ Certificate Authority (CA) ทำให้การสื่อสารนั้นมีความมั่นคงมากยิ่งขึ้น แต่ทว่าในปัจจุบัน HTTPS นั้นยังไม่มั่นคงเท่าที่ควร สามารถถูกโจมตีได้ด้วยวิธีที่หลากหลาย เช่น SSL Stripping Attack และ Downgraded Attack

ภาพประกอบที่ 2.2 การรับ – ส่งข้อมูลบน HTTPS Protocol

จากภาพประกอบที่ 2.2 การรับ – ส่งข้อมูลบน HTTPS Protocol จะเห็นได้ว่า Client เรียกเว็บไซต์ปลายทางในรูปแบบ HTTP โดยอาศัย port 80 เป็นช่องทางในการสื่อสาร และ Server ตอบกลับมาในรูปแบบ HTTPS โดยอาศัย port 443 เป็นช่องทางในการสื่อสาร การสื่อสารโดย HTTPS Protocol มีการเข้ารหัสเพื่อให้การรับส่ง - ส่งข้อมูลนั้นมั่นคง

3) Salted - Hash Password [9]

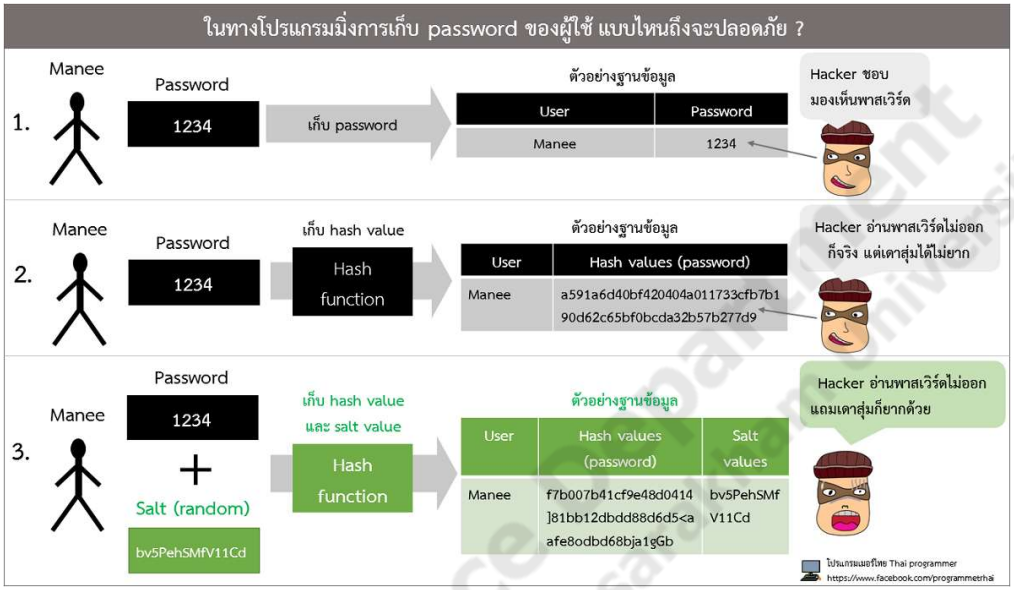
Cryptographic Hash หรือที่เรียกกันว่า Hashing คือการสร้างข้อมูลที่เป็นตัวแทนของข้อมูลที่ต้องการ ซึ่งอาจจะเป็นรหัสผ่าน หรือข้อมูลส่วนบุคคลอื่นๆ และนำไปจัดเก็บในฐานข้อมูลหรือใน Text file หรือในที่อื่นๆ ซึ่งข้อดีของการทำ Hash คือจะไม่สามารถถอดรหัส หรือกระทำการใดๆ เพื่อที่จะ Reverse ให้ออกมาเป็นข้อความต้นฉบับ ซึ่งในปัจจุบันมีวิธีการ Hash มากมาย เช่น MD5, SHA1, SHA256, SHA512 และ SHA3 เป็นต้น

MD5 Hashing & Cracking เป็นการทำ Hash ที่พื้นฐานที่สุด และเมื่อหลายปีที่ผ่านมามีข่าวออกมาว่ามีผู้ที่สามารถ Crack ได้สำเร็จ ทำการ Hash ที่มีการเกิดการซ้ำกันของค่า Hash เนื่องจากการมีคุณสมบัติแทนข้อมูลที่ต้องการ ซึ่งค่า Hash ที่สร้างขึ้นจะมีความยาวที่เท่ากันเสมอ ซึ่งสำหรับ MD5 ก็จะมีความยาว 16 bytes (128 bits) ซึ่งค่า hash ที่เป็นไปได้ทั้งหมดก็จะมีค่า 256^{16} หรือ 2^{128}

Rainbow Table เป็นการเก็บข้อมูล Hash โดยมีข้อมูลต้นฉบับจากการ Brute Force เพื่อความรวดเร็วในการตรวจสอบ ซึ่งในปัจจุบัน GPU ระดับปานกลางหลายๆ รุ่นจะสามารถคำนวน Hash ได้ในระดับ 10 ล้าน Hash ต่อวินาที ซึ่งในปัจจุบันมีผู้ยอมเสียเวลาเพียงครั้งเดียวเพื่อสร้าง Hash ที่มีควายาวมากๆ และมีความซับซ้อน

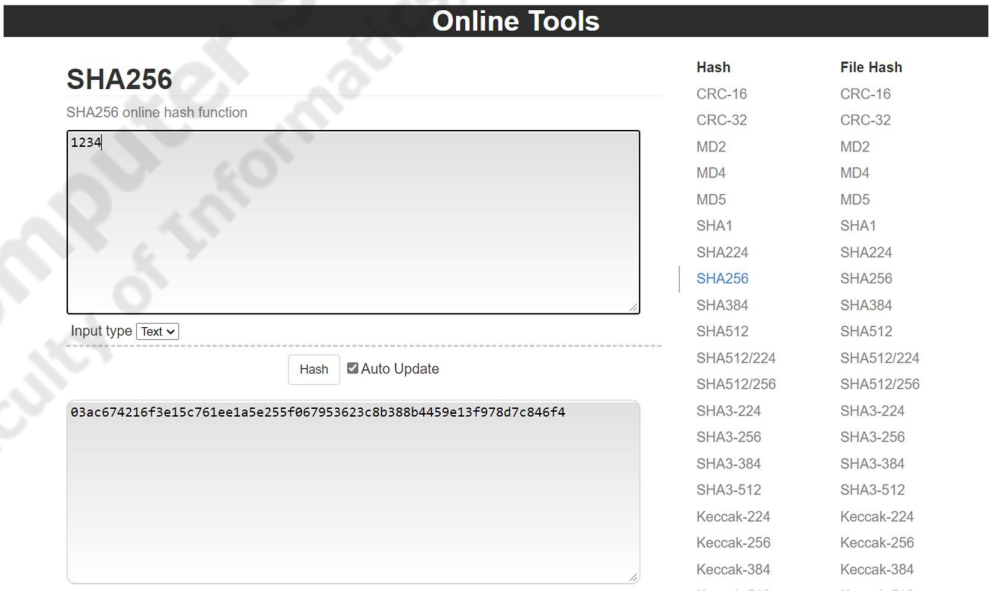
Salted - Hash Password (SHP) เป็นการเข้ารหัสโดยอาศัยแนวคิดของ Salting คือ การให้ค่าของเกลือที่ต่างกัน แม้จะใช้รหัสผ่านเดียวกันแต่ค่าของการ Hash จะต่างกัน ช่วยเพิ่มความมั่นคง

ให้กับข้อมูลที่เป็นความลับ โดยเฉพาะอย่างยิ่งกับเทคนิคการโจมตี Brute Force Attack คือการผสมทักตัวอักษรที่เป็นไปได้จนกว่าจะพบรหัสผ่าน แนวคิดของ Salted - Hash Password ก็คือการเติมเกลือที่ส่วนท้ายของรหัสผ่านจากนั้นจึงแฮชรหัสผ่านจะทำให้ขั้นตอนการถอดรหัสรหัสผ่านซับซ้อนขึ้น



ภาพประกอบที่ 2.3 กรณีตัวอย่าง Salted - Hash Password

[HTTPS://www.patanasongsivilai.com/blog/Password-hash-function-salt-values/]



ภาพประกอบที่ 2.4 ตัวอย่างการเข้ารหัสด้วยเทคนิค Hash ด้วย SHA256

[HTTPS://emn178.github.io/online-Tools/sha256.html]

ภาพประกอบที่ 2.5 การเข้ารหัสด้วยเทคนิค Hash ด้วย SHA256

[HTTPS://md5decrypt.net/en/Sha256/#answer]

จากภาพประกอบที่ 2.3 กรณีตัวอย่าง Salted - Hash Password สามารถแบ่งกรณีตัวอย่างออกได้เป็น 3 กรณี คือ

1) ไม่มีการเข้ารหัส (Clear text) Password ถูกจัดเก็บในฐานข้อมูลโดยตรง สามารถถูกดักจับข้อมูลได้โดยง่าย

2) เข้ารหัสด้วยฟังก์ชัน Hash ปกติ หากข้อมูลนั้นง่ายก็การสุ่มเดามากเกินไป Hacker ก็จะสามารถสุ่มเดาได้ และนำมาเทียบได้เนื่องจากผลลัพธ์ที่ได้จากการ Hash function เดียวกัน ข้อความเดิมในแต่ละครั้งจะได้ผลลัพธ์เดียวกัน ดังภาพประกอบที่ 2.4 ตัวอย่างการเข้ารหัสด้วยเทคนิค Hash ด้วย SHA256 และภาพประกอบที่ 2.5 การเข้ารหัสด้วยเทคนิค Hash ด้วย SHA256

3) เข้ารหัสด้วยเทคนิค Salted - Hash Password

4) SSL Stripping Attack

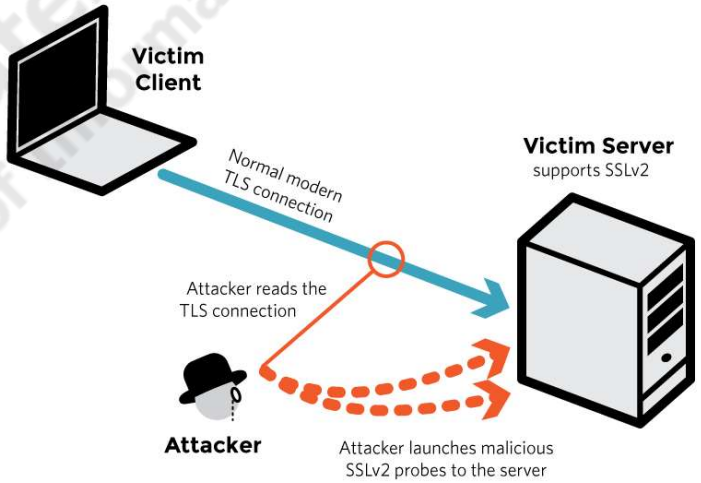
เทคนิคการเปลือยเอสเอสแอล (SSL Stripping Attack) ถูกเสนอขึ้นในปี ค.ศ. 2009 โดย Moxie Marlinspike ที่งาน Black hat DC ซึ่ง SSL Strip นั้นเป็นการโจมตีในรูปแบบ Man in the middle Attack (MITM) คือการแทรกกลางการสื่อสาร โดยปกติเมื่อ Client ร้องขอไปที่ Server ในรูปแบบของ HTTP แล้ว Server จะทำการตอบกลับมาในรูปแบบของ HTTPS ซึ่งจะมีความปลอดภัยในการสื่อสาร แต่ทว่าการโจมตีแบบ MITM นั้นเมื่อ Client ร้องขอไปที่ในรูปแบบของ HTTP แล้ว Server จะทำการตอบกลับมาในรูปแบบ HTTPS ซึ่งทำให้การสื่อสารนั้นปลอดภัย แต่เทคนิคดังกล่าวจะทำการแทรกกลางการสื่อสาร และเปลี่ยนกลับมาเป็น HTTP อีกครั้งหลังจากที่ server นั้นเปลี่ยนเป็น HTTPS แล้วจึงทำให้

Client ที่ร้องขอไปในรูปแบบของ HTTP ไม่รู้ตัวว่าถูกแทรกกลางการสื่อสารแล้ว ดังนั้น SSL Strip นั้นสามารถโจมตี HTTPS ได้ หาก Client ไม่สังเกตเห็นถึงความผิดปกติก็จะทำให้ถูกดักจับข้อมูลที่สำคัญต่าง ๆ ได้เนื่องจาก HTTP ไม่มีการเข้ารหัส (Clear text)

5) DROWN Attack

Decrypting RSA with Obsolete and Weakened Encryption Attack (DROWN Attack) เป็นช่องโหว่ที่มีผลต่อ HTTPS และ service อื่น ๆ ที่ทำงานร่วมกับ Secure Socket Layer (SSL), Transport Layer Security (TLS) และโพรโทคอลการเข้ารหัสที่สำคัญอื่น ๆ ซึ่งโพรโทคอลเหล่านี้อนุญาตให้ทุกคนเบราซ์เว็บไซต์ ส่งอีเมลล์ ซื้อของออนไลน์ และสื่อสารออนไลน์โดยไร้ซึ่งบุคคลที่สาม (Third - Parties) ถูกเผยแพร่ครั้งแรกในปี 1995

DROWN Attack อนุญาตให้แฮคเกอร์หยุดการเข้ารหัสโดยการกำหนดค่าผิดให้รองรับ SSLv2 ซึ่งเป็น TLS รุ่นก่อนปี 1990 นอกจากปัญหาที่เกิดใน SSLv2 ยังเกิดจากช่องโหว่ของ RSA อีกด้วยซึ่งทำให้แฮคเกอร์สามารถอ่าน หรือขโมยข้อมูลที่เป็นความลับได้ เช่น รหัสผ่าน เลขบัตรเครดิต ความลับทางการค้า หรือข้อมูลการเงิน เป็นต้น เมื่อเดือนมีนาคม ปี 2016 วัดค่าได้ 33% ของ HTTPS server มีช่องโหว่ที่ทำให้สามารถโจมตีได้ แต่ในปัจจุบันช่องโหว่ดังกล่าวมีให้พบเห็นน้อยลงมาก ในปี 2019 แลป SSL มีการประมาณการว่ามีช่องโหว่บน HTTPS server เพียง 1.2% แม้ว่า TLS ในเวอร์ชันล่าสุดจะไม่อนุญาตให้มีการเชื่อมต่อโดย SSLv2 แล้วก็ตาม หากแต่ผู้ดูแลระบบกำหนดค่าบางอย่างเพื่อให้เกิดความยืดหยุ่นต่อการใช้งานนั่นจึงกลายเป็นช่องโหว่ที่ทำให้ผู้ไม่หวังดีสามารถนำมาใช้เพื่อโจมตีระบบได้ ในการป้องกันการโจมตีเบื้องต้นสามารถทำได้โดยปิดการใช้งาน SSLv2 เพื่อในแน่ใจว่า Private Key จะไม่ถูกใช้บนโพรโทคอล SSLv2



ภาพประกอบที่ 2.6 ลักษณะการโจมตีด้วยเทคนิค DROWN Attack

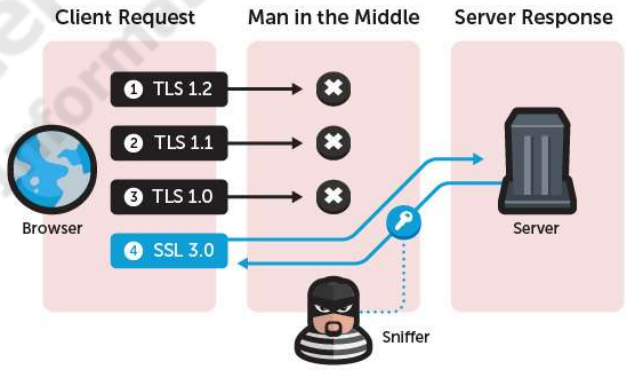
[HTTPS://drownAttack.com/]

ภาพประกอบที่ 2.6 Client ใช้โพรโทคอล TLS Version ปัจจุบัน ร้องขอไป Server โดยมี Hacker เป็นคนกลางในการสื่อสารนี้ และเห็นช่องโหว่ของ Server ที่สนับสนุน SSL Version 2 ซึ่งเป็นช่องโหว่ที่สามารถถูกโจมตีได้ด้วยเทคนิค DROWN Attack จากนั้น Hacker ทำการ Downgraded การเชื่อมต่อของ Client ลงมาเป็น SSL Version 2 นั่นเอง

6) Padding Oracle On Downgraded Legacy Encryption (POODLE) Attack

Padding Oracle On Downgraded Legacy Encryption (POODLE) เป็นช่องโหว่ของ Man in the middle (MITM) ตัวแรกของโลก ถูกค้นพบโดย Thai Duong และ Krzysztof Kotowicz นักวิจัยด้านความปลอดภัยของ Google และถูกเผยแพร่ในช่วงปลายปี 2014 POODLE Attack เกิดจากการเชื่อมต่อ TLS ซึ่งเกิดการ Downgraded ให้เป็น SSLv3 โดยช่องโหว่นี้ทำให้แฮคเกอร์สามารถดักจับการสื่อสารที่เข้ารหัสได้ ช่องโหว่ใน SSLv3 จะไม่พบใน TLS สามารถทำได้โดยดร็อป Session ซ้ำ ๆ จนกระทั่ง Client ทดลองใช้ SSL ในเวอร์ชันเก่ากว่า ซึ่งการเข้ารหัสบน SSLv3 นั้นใช้การเข้ารหัสแบบ RC4 Stream Cipher หรือ Block Cipher ในโหมด CBC ปัญหาของ Server ที่เข้ารหัสด้วย CBC ใน SSLv3 คือ Block Cipher ที่ไม่ครอบคลุมถึง MAC (Message Authentication Code) เมื่อถอดรหัส Padding ที่สมบูรณ์จะไม่สามารถตรวจสอบได้อย่างถี่ถ้วน Padding ที่มีความยาวตั้งแต่ 1 ถึง L ไบต์ (L คือ ขนาดของ Block ที่มีหน่วยเป็นเป็นไบต์)

ในปี 2020 Web Application Vulnerability รายงานว่ามีมากกว่า 3.9% ของเว็บเซิฟเวอร์ที่ยังสนับสนุน SSLv3 ซึ่งเป็นช่องโหว่ของ POODLE แต่ในปัจจุบันแม้จะใช้ TLS Protocol ก็ยังมีมากกว่า 30% ที่จะถูกโจมตี เนื่องจากสนับสนุน TLS 1.0 ซึ่งเป็นช่องโหว่ที่ BEAST Attack สามารถโจมตีได้



ภาพประกอบที่ 2.7 ลักษณะการโจมตีด้วยเทคนิค POODLE Attack

[HTTPS://www.trendmicro.com/en_us/research/14/j/poodle-vulnerability-puts-online-transactions-at-risk.html]

ภาพประกอบที่ 2.7 ลักษณะการโจมตีด้วยเทคนิค POODLE Attack มี Hacker เป็นคนกลางการสื่อสาร โดยจะ Block ไม่การเชื่อมต่อจาก TLS Protocol และดร็อป Session ซ้ำๆ จนกระทั่ง Client ยอมใช้ SSL Version 3 แล้วจึงทำการดักจับข้อมูล

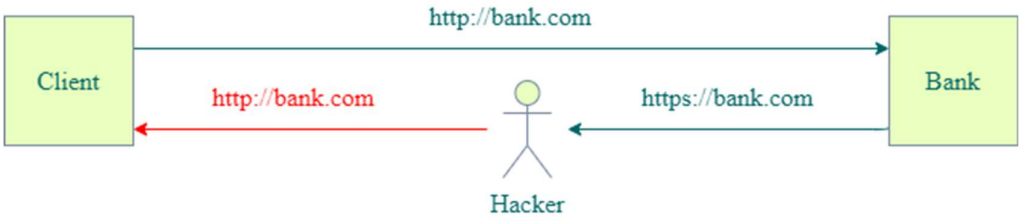
7) BEAST Attack

Browser Exploit Against SSL/TLS (BEAST) เป็นการโจมตีช่องโหว่บน network ใน TLS 1.0 และ SSL เวอร์ชันเก่า ๆ BEAST Attack ถูกค้นพบเมื่อปี 2011 โดย Thai Duong และ Juliano Rizzo ซึ่งเป็นนักวิจัยด้าน Security ที่ทำงานให้กับ Google ช่องโหว่ดังกล่าว ในทางทฤษฎีนั้นถูกค้นพบตั้งแต่ปี 2002 โดย Phillip Rogaway นักวิจัยด้านวิทยาการรหัสลับ และจากการวิจัยในปี 2020 โดย Acunetix Web Application Vulnerability พบว่า 30.7% ที่ทำการตรวจสอบยังมีช่องโหว่ TLS 1.0 ที่ทำให้ BEAST Attack สามารถโจมตีได้ ซึ่งแสดงให้เห็นถึงปัญหาด้านความปลอดภัยยังต่อธุรกิจ

ไม่ว่าจะมีการนำคุณลักษณะใหม่ ๆ ที่ปรับปรุงด้านความปลอดภัยมาใช้ในซอฟต์แวร์ ปัญหาช่องโหว่ของ SSL/TLS ก็ยังคงเป็นปัญหาสำคัญที่ทำให้การโจมตีในรูปแบบเก่าไม่ว่าจะเป็น BEAST, POODLE หรือ Open SSL และ Heartbleed นั้นยังสามารถโจมตีได้

8) Man in the middle (MITM)

การโจมตีแบบ Man in the middle หรือ MIMT เป็นการโจมตีโดยที่มีผู้ไม่หวังดีเข้ามาแทรกกลางระหว่างการสนทนาระว่าง Web Client และ Web Server แล้วทำหน้าที่เป็นตัวกลางในการรับ - ส่งข้อมูลของคู่สนทนา โดยที่คู่สนทนาไม่ทราบว่าได้มีผู้อื่นเป็นผู้รับผู้ส่งกับคู่สนทนาของตนอยู่ ทำให้ผู้ไม่หวังดีใช้รูปแบบการโจมตีในลักษณะดักรับหรือเปลี่ยนแปลงข้อมูลที่ทั้ง 2 ฝั่งสื่อสารอยู่ ซึ่งการโจมจีนี้ถูกนำมาประยุกต์ใช้กับการสื่อสารต่าง ๆในระบบคอมพิวเตอร์และอุปกรณ์ Wi-Fi access Point เพื่ออ่านปลอมแปลง หรือแก้ไขข้อมูลที่รับส่งระหว่างคอมพิวเตอร์ทั้ง 2 เครื่อง ซึ่งการเข้ารหัสเพียงอย่างเดียวไม่สามารถป้องกันการโจมตีนี้ได้เสมอไป ถ้าผู้รับผู้ส่งไม่ได้มีกลไกใด ๆ ในการยืนยันคู่สนทนาได้อย่างถูกต้อง การโจมตีแบบ MITM สามารถใช้การโจมตีการสื่อสารข้อมูลของระบบต่าง ๆ ในเครือข่ายอินเทอร์เน็ตได้โดยง่าย เนื่องจากรูปแบบมาตรฐานของการสื่อสารข้อมูลต่าง ๆ ในระบบอินเทอร์เน็ตไม่ได้ถูกออกแบบมาให้มีการรักษาความปลอดภัยของข้อมูล เช่น การสื่อสารข้อมูลผ่านโพรโทคอล HTTP สำหรับเรียกดูข้อมูลเว็บไซต์ต่าง ๆส่วนใหญ่ไม่มีการเข้ารหัสลับทำให้ผู้โจมจีสามารถใช้โปรแกรมสำหรับดักจับข้อมูลในเครือข่ายได้ เช่น โปรแกรม Wireshark, TCPDump, Ettercap หรือ BetterCap



ภาพประกอบที่ 2.8 ลักษณะการโจมตีด้วยเทคนิค MITM

จากภาพประกอบที่ 2.8 ลักษณะการโจมตีด้วยเทคนิค MITM จะเห็นได้ว่า Client ร้องขอเว็บไซต์ HTTP จากนั้น Server จะทำการตอบกลับมาในรูปแบบ HTTPS แต่ Hacker ที่เป็นคนกลางการสื่อสารระหว่าง Client – Server ทำการเปลี่ยนกลับมาในรูปแบบ HTTPทำให้ Client ที่ร้องขอไปในรูปแบบของ HTTP นั้นไม่รู้ตัวว่าถูกโจมตีแล้ว

2.2 Software ที่เกี่ยวข้อง

1) Kali Linux Operating System [10]

Kali เป็นระบบปฏิบัติการ (Linux OS) ระบบปฏิบัติการณ์หลายตัว เช่น MacOS, Windows, DOS และ Linux แต่ Kali เป็น Linux OS ที่ออกแบบมาสำหรับงานด้านความมั่นคงระบบไอที โดยการที่ติดตั้ง Software ต่างๆ ที่มักจะถูกใช้งาน ในการทำงานเอาไว้เรียบร้อยแล้ว หรือยังไม่ได้ติดตั้ง แต่สามารถติดตั้งได้โดยง่าย ผ่านระบบติดตั้งโปรแกรมที่มีให้เรียกว่า Software Repository ของ Kali โดยเฉพาะ

Kali มักถูกมองว่าเป็นระบบของ Hacker ติดตั้งแล้วจะเป็น Hacker เจาะระบบ แต่ในความเป็นจริงแล้ว ความสำคัญของการทำงานด้านความมั่นคง จะเป็นความรู้เกี่ยวกับการใช้งานมากกว่าการมีโปรแกรม เหมาะแก่การใช้งานไว้ทดลอง ทดสอบโปรแกรมด้านความมั่นคงต่างๆ เหมาะสำหรับมือใหม่และผู้ไม่อยากเสียเวลาลงโปรแกรมเอง ที่คนทำงานด้านไอทีทุกคนควรศึกษาและลองใช้เพื่อศึกษาโปรแกรมต่างๆ ที่ Hacker อาจจะใช้ในการทดสอบระบบ

2) Ettercap

Ettercap เป็น tool ที่ใช้ทำในเรื่องของการดักจับข้อมูลหรือใช้สำหรับโจมตีในรูปแบบของ Man in the middle ที่อยู่ภายในเครือข่ายเดียวกัน โดยตัว tool นั้นสามารถที่จะทำการโจมตีได้ในหลากหลายรูปเช่น ARP poisoning, ICMP redirect หรือ DHCP snooping ซึ่งในครั้งนี้เราจะทำการทดสอบการโจมตีด้วยวิธี DNS spoofing เป็นการเปลี่ยนข้อมูลของ DNS ให้วิ่งไปที่ IP Address ปลายทางที่อื่นที่ไม่ใช่ของจริง และวิธีการโจมตีแบบนี้จะสังเกตเห็นความผิดปกติได้ยาก เนื่องจากใน Address Bar ของเบราว์เซอร์จะแสดง URL ที่ถูกต้อง แต่เว็บไซต์ปลายทางนั้นไม่ใช่เว็บไซต์ที่แท้จริง จุดประสงค์หลักๆ ของการโจมตีด้วยวิธีนี้อาจจะเป็นการขโมยข้อมูลส่วนตัวของผู้ถูกโจมตีเช่น เว็ปไซต์

ของธนาคารอาจจะมีการปลอมแปลงเว็ปไซต์ของธนาคารขึ้นมาเพื่อให้เรากรอกข้อมูลส่วน Username, Password ลงไป ก็ทำให้ Attacker ได้ข้อมูลของเราไปได้ง่าย

3) Python QT Designer

Python QT Designer เป็นเครื่องมือที่ใช้ในการสร้าง และออกแบบ Graphical User Interface (GUI) สามารถทำงานบน Desktop PC, Smart phone, Embedded system สามารถทำงานได้หลายระบบปฏิบัติการ (Operating System) หรือ Cross – platform เมื่อมีโปรแกรมที่ทำงานบน OS หนึ่ง สามารถนำไป Compile เพื่อให้สามารถทำงานบน OS อื่นโดยที่ไม่ต้องแก้โปรแกรม QT Designer สามารถเลือกใช้ API, Library ต่างๆ ที่เขียนด้วยภาษา C++ ทำให้สามารถพัฒนาแอปพลิเคชันแบบ GUI และยังสนับสนุนการพัฒนาทั้ง C++, Java, Python, Peal และ PHP

License Comparison Chart

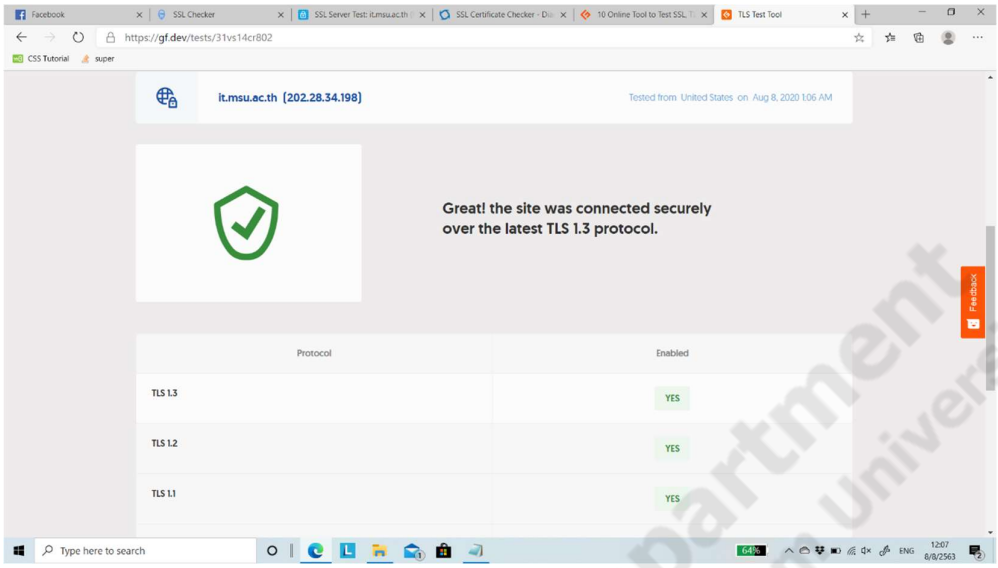
	Commercial	LGPL	GPL
License cost	License fee charged	No license fee	No license fee
Must provide source code changes to Qt	No, modifications can be closed	Source code must be provided	Source code must be provided
Can create proprietary applications	Yes - No source code must be disclosed	Yes, in accordance with the LGPL v. 2.1 terms	No, applications are subject to the GPL and source code must be made available
Updates provided	Yes, immediate notice sent to those with a valid support and update agreement	Yes, made available	Yes, made available
Support	Yes, to those with a valid support and update agreement	Not included but available separately for purchase	Not included but available separately for purchase
Charge for Runtimes	Yes, for some embedded uses	No	No

ภาพประกอบที่ 2.9 Software license QT ทั้ง 3 แบบ

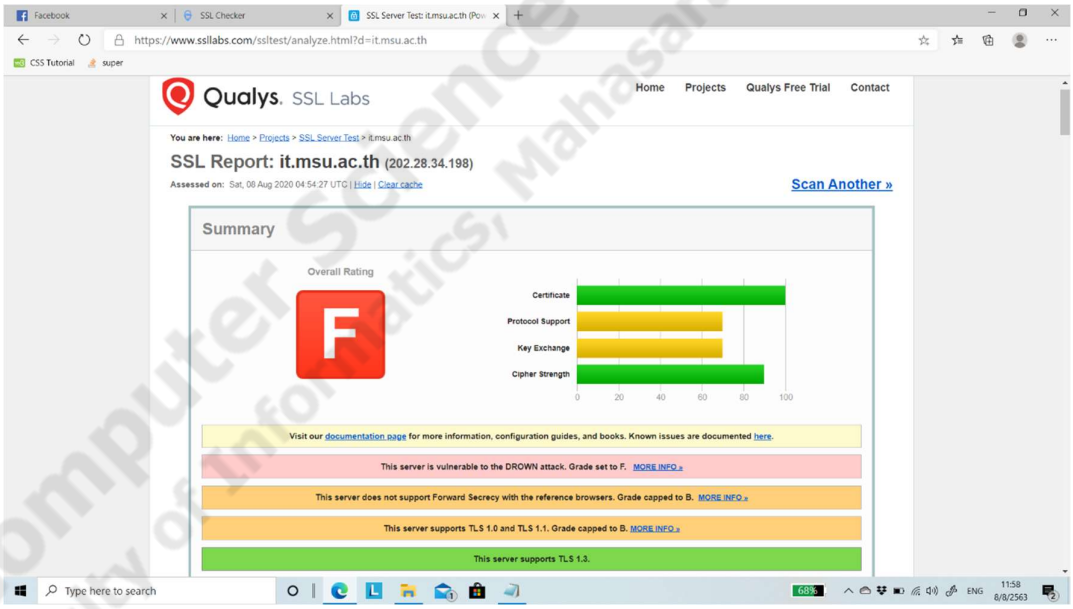
[HTTPS://bit.ly/3dgW33Z]

4) บริการที่ใช้ทดสอบช่องโหว่ของ SSL (SSL Tester)

SSL Tester เป็นเครื่องมือที่ให้บริการในการตรวจสอบหาช่องโหว่ของ SSL/TLS บนเว็บไซต์คอยติดตามความคืบหน้า และการกำหนดค่า SSL, Certificate Authentication (CA) เป็นต้น เว็บไซต์ที่ให้บริการในการตรวจสอบช่องโหว่ SSL/TLS บนเว็บไซต์มีหลากหลายในท้องตลาดเช่น SSL labs by qualys, SSL Checker, Digicert SSL Certificate checker และ Gleekflare เป็นต้น



ภาพประกอบที่ 2.10 ตัวอย่างเว็บไซต์ที่ให้บริการตรวจสอบช่องโหว่ของ SSL/TLS



ภาพประกอบที่ 2.11 ตัวอย่างเว็บไซต์ที่ให้บริการตรวจสอบช่องโหว่ของ SSL/TLS