ระบบรักษาความปลอดภัยบนเว็บไซต์การค้า

Commercial Web Security

โดย

นาย วุฒิมล สุภาพ

นาย สาธิต บุญเทือง

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

Commercial Web Security

BY

Mr. Wuttimol Supap

Mr. Sathit Boonthueang

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENT FOR THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2006

หัวข้อปริญญานิพนธ์	ระบบรักษาความปลอดภัยบนเว็บไซต์การค้า			
ชื่อนักศึกษา	นายวุฒิมล	สุภาพ	รหัสประจำตัว	46012195
	นายสาธิต	บุญเทือง	รหัสประจำตัว	46012203
อาจารย์ที่ปรึกษา	ผศ. บุณย์ชนะ ภู่ระหงษ์			
	ดร. สมเกียรติ อุดมหรรษากุล			
ระดับการศึกษา	ปริญญาวิศวกรรมศาสตร์บัณฑิต			
	สาขาวิศวกรรมสารสนเทศ			
ภาควิชา	วิศวกรรมสารสนเทศ			
ปีการศึกษา	2549			

ปริญญานิพนธ์ฉบับนี้ได้รับการอนุมัติเป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

.....

(ผศ. สมเกียรติ อุดมหรรษากุล)

อาจารย์ที่ปรึกษา

.....

(ผศ. บุณย์ชนะ ภู่ระหงษ์)

อาจารย์ที่ปรึกษา

หัวข้อปริญญานิพนธ์	ระบบรักษาความปลอดภัยบนเว็บไซต์การค้า			
ชื่อนักศึกษา	นายวุฒิมล	สุภาพ	รหัสประจำตัว	46012195
	นายสาธิต	บุญเทือง	รหัสประจำตัว	46012203
อาจารย์ที่ปรึกษา	ผศ. สมเกียรติ อุดมหรรษากุล			
	ผศ. บุณย์ชนะ ภู่ระหงษ์			
ระดับการศึกษา	ปริญญาวิศวกรรมศาสตร์บัณฑิต			
	สาขาวิศวกรรมสารสนเทศ			
ภาควิชา	วิศวกรรมสารสนเทศ			
ปีการศึกษา	2549			

บทคัดย่อ

ปัจจุบันเวิลด์ไวด์เว็บได้เข้ามาอำนวยความสะดวก และได้กลายเป็นส่วนหนึ่งในชีวิตประจำวันของเราอย่างหลีกเลี่ยงไม่ได้ และเราก็ได้มีการทำธุรกรรมมากมายบนระบบอินเตอร์เน็ต ซึ่งเราจะมั่นใจได้อย่างไรว่าระบบที่เราเข้าไปใช้นั้นมีความปลอดภัย จากการถูกโจรกรรมข้อมูลโดยผู้ไม่หวังดี ดังนั้นจึงจำเป็นต้องมีการสร้างระบบรักษาความปลอดภัยขึ้นมา

เนื้อหาของปริญญานิพนธ์นี้จะประกอบด้วยการศึกษาการทำงานของระบบรักษาความปลอดภัยบนเวิลด์ไวด์เว็บ ศึกษาถึงปัญหาด้านความปลอดภัย รวมไปถึงแนวทางป้องกันและแก้ไขปัญหา และจะนำความรู้ที่ได้จากการศึกษามาพัฒนาเว็บไซต์การค้า ที่มีระบบรักษาความปลอดภัยที่น่าเชื่อถือแก่ผู้ใช้บริการต่อไป

THESIS TITLE	Commercial Web Security
STUDENT	Mr. Wuttimol Supap No. 46012195
	Mr. Sathit Boonthueang No. 46012203
ADVISOR	Assoc.Prof. Somkait Udomhunsakul
	Assoc.Prof. Boonchana Purahong
Graduate Level	Bachelor Degree of Information Engineering
Department	Information Engineering
Academic Year	2006

Abstract

As of today, world wide web comes in to our life style to support our convenience. We barelyavoid to have contact with various internet transactions. However, we never know the security of the system that we are entering. The system can be hijack from unwelcome user. Therefore, the security protection support system must be created.

In this thesis content, we will study the security system of world wide web. Also reserch and study criticaly in detail area such as security problems, method of protection, and problem solve. We will take the reserch result in to account of improving the commercial website that has creditable security system for the customer.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี ต้องขอกราบขอบพระคุณบิดา มารดาที่ให้กำลังใจเสมอมา ขอขอบพระคุณ ผศ. บุณย์ชนะ ภู่ระหงษ์ และ ผศ. สมเกียรติ อุดมหรรษากุล อาจารย์ที่ปรึกษาที่ได้ช่วยเหลือให้คำชี้แนะ และได้ให้ความรู้ในการทำปริญญานิพนธ์ฉบับนี้เป็นอย่างดี

สุดท้ายนี้ทางคณะผู้จัดทำ ขอขอบคุณ อาจารย์ทุกท่านที่กรุณาประสิทธิ์ประสาทวิชาความรู้ รวมทั้งแนวทางความคิด แนวทางปฏิบัติ ให้แก่คณะผู้จัดทำ จนทำให้ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปด้วยดี

วุฒิมล สุภาพ

สาธิต บุญเทือง

สารบัญ

สารบัญ(ต่อ)

สารบัญรูป

สารบัญตาราง

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มา

เนื่องจากในปัจจุบันเวิลด์ไวด์เว็บได้เข้ามาอำนวยความสะดวก และได้กลายเป็นส่วนหนึ่งในชีวิตประจำวันของเราอย่างหลีกเลี่ยงไม่ได้

อย่างไรก็ตาม ถึงแม้จะช่วยอำนวยความสะดวก และความรวดเร็วแก่ผู้ใช้มากมายเพียงใด แต่เนื่องจากเวิลด์ไวด์เว็บเป็นบริการบนเครือข่ายขนาดใหญ่(Internetworking) ทุกๆ คนสามารถเข้าใช้บริการและให้บริการต่างๆ ได้อย่างเสรี ดังนั้นจึงไม่น่าแปลกใจที่จะแฝงไปด้วยภัยอันตรายต่างๆ มากมาย เช่น การละเมิดความเป็นส่วนตัว (Privacy) การโจรกรรมข้อมูลโดยแฮกเกอร์ (Hacker) และไวรัส(Virus) ยิ่งในปัจจุบันธุรกิจการค้าออนไลน์ในระบบอินเตอร์ได้รับความนิยมกันอย่างแพร่หลาย และเราก็ได้ทำธุรกรรมมากมายบนระบบนั้นเราจะสามารถมั่นใจได้อย่างไรว่าระบบที่เราใช้บริการอยู่นั้นมีความปลอดภัยจากอันตรายต่างๆ

อาจมีผู้ใช่หลายคนใช้งานระบบเวิลด์ไวด์เว็บมานานแล้วแต่ไม่เห็นจะมีปัญหาด้านความปลอดภัย นั้นเพราะบุคคลเหล่านั้นยังใช้ยังไม่มีความรู้ความเข้าใจ ทำให้มองไม่เห็นปัญหาที่เกิดขึ้นนั้นเอง หากมองจากปัญหาที่เกิดขึ้นแล้ว เราจะสามารถทำการจำแนกปัญหาได้เป็น 3 ประเภท คือ ปัญหาความปลอดภัยที่เว็บเซิร์ฟเวอร์ ปัญหาความปลอดภัยที่เว็บไคลเอนต์ และปัญหาความปลอดภัยของการสื่อสารระหว่างเว็บไคลเอนต์และเว็บเซิร์ฟเวอร์ ดังนั้นการที่จะสามารถป้องกันตนเองให้ปลอดภัยจากอันตรายต่างๆนั้นก่อนอื่นเราควรที่จะศึกษาถึงลักษณะของปัญหาต่างๆ ที่เกิดบนเวิลด์ไวด์เว็บดังที่กล่าวไว้ข้างต้นให้ครอบคลุม ถูกต้องและเข้าใจอย่างลึกซึ้ง หลังจากนั้นจะสามรถนำมาประยุกต์ใช้กับสถานการณ์จริงได้อย่างถูกต้องและเหมาะสม และเนื่องจากเวิลด์ไวด์เว็บมีการพัฒนาไปอย่างรวดเร็วปัญหาต่างๆ ก็สามารถเกิดขึ้นใหม่อยู่เรื่อยๆ ดั้นนั้นปัจจัยที่สำคัญที่จะทำให้ระบบความปลอดภัยบนเวิลด์ไวด์เว็บมีประสิทธิภาพก็คือ ต้องมีความรู้ความเข้าใจให้ทันต่อการพัฒนาอยู่ตลอดเวลานั้นเอง

อย่างไรก็ตีการทราบทฤษฎีอย่างเดียวนั้นไม่สามารถทำให้เข้าใจปัญหาความปลอดภัยต่างๆ ได้อย่างลึกซึ้ง ดั้งนั้นจึงจำเป็นอย่างยิ่งที่ต้องมีการนำทฤษฎีที่น่าสนใจมาทำการประยุกต์ใช้จริง เพื่อพิสูจน์ว่าสอดคล้องกับทฤษฎีหรือหลักการดังที่กล่าวไว้หรือไม่

1.2 วัตถุประสงค์

1.2.1 ศึกษาระบบรักษาความปลอดภัยของเว็บไซต์การค้าและแนวทางแก้ปัญหา

1.2.2 เพื่อพัฒนาระบบเว็บไซต์การค้าที่มีระบบรักษาความปลอดภัยที่ดี

1.3 ขอบเขตของโครงงาน

1.3.1 ศึกษาระบบความปลอดภัยสำหรับเว็บไซต์การค้า

1.3.1.1 ศึกษาการทำงาน ลักษณะของปัญหาและแนวทางการแก้ไขที่เว็บเซิร์ฟเวอร์

1.3.1.2 ศึกษาการทำงาน ลักษณะของปัญหาและแนวทางการแก้ไขที่เว็บเว็บไคลเอนต์

1.3.1.3 ศึกษาการทำงาน ลักษณะของปัญหาและแนวทางการแก้ไขสำหรับการสื่อสารระหว่างเว็บไคลเอนต์และเว็บเซิร์ฟเวอร์

1.3.2 จัดทำระบบจำลองการทำงานของเว็บไซต์การค้าที่มีระบบรักษาความปลอดภัยดังนี้คือ Encryption, CA (Certificate Authentication), SSL (Secure Socket Layer), Authentication

1.4 ขั้นตอนการดำเนินโครงงาน

1.4.1 ศึกษาระบบความปลอดภัยสำหรับเว็บไซต์การค้าและทฤษฎีที่เกี่ยวข้อง

1.4.2 จัดทำระบบจำลองการทำงานของเว็บไซต์การค้า

บทที่ 2
ทฤษฎี

2.1 โพรโตคอลHTTP

HTTP ย่อมาจาก Hypertext Transfer Protocol เป็นข้อกำหนดหรือวิธีการติดต่อสื่อสารหรือจัดการข้อมูลประเท Hypertext หรือที่เรียกกันโดยมากว่า HyperMedia ทั้งนี้เพราะตัวรูปแบบเอกสารเปลี่ยนไปจากเดิมที่เป็นเอกสารข้อความ(Text) เชื่อมโยงกันเป็นโครงข่ายแต่ปัจจุบัน ข้อมูลอาจเป็นทั้งภาพ เสียงหรืออื่นๆ(พวกมัลติมีเดีย Multimedia) ซึ่งเป็นโพรโตคอล HTTP เป็นพื้นฐานในการใช้งานหรือการทำงานสื่อสารของเครือข่ายเวิร์ลไวด์เว็ป โดยการทำงานของโพรโตคอลดังกล่าวนี้มีส่วนต่างๆ ที่ควรทราบ ดังนี้

การทำงานของโพโตคอล HTTP ซึ่งการทำงานจะแบ่งเป็น สามส่วนหลักๆ คือ ส่วนไคลเอนต์ เซิร์ฟเวอร์และเครือข่าย ตามหลักการพื้นฐานของวิธีการทำงานแบบไคลเอนต์-เซิร์ฟเวอร์ ฝั่งไคลเอนต์จะใช้งานโปรแกรมเว็บบราวเซอร์ติดต่อกับเว็บเซิร์ฟเวอร์ ซึ่งคือผู้ให้บริการเว็บโดยโปรแกรม ดังกล่าวจะเรียกว่า HTTPD (HyperText Transfer Protocol Daemon)ปกติแล้วจะทำงานที่พอร์ต 80 สำหรับหลักการของไคลเอนต์-เซิร์ฟเวอร์ คือ มีตัวไคลเอนต์เป็นผู้ให้บริการและมีเซิร์ฟเวอร์เป็นผู้ให้บริการ โดยวิธีการทำงานจะใช้วิธีร้องขอ(request) และตอบสนองการขอบริการ (Response/Reply) ซึ่งมีลักษณะดังนี้

2.1.1 การร้องขอบริการ

สำหรับ HTTP จะมีคำสั่งหลักๆในการจัดการดังนี้ OPTION, HEAD, PUT, DELETE, TRACE, GET และ POST แต่คำสั่งหลักๆ สำหรับผู้ร้องขอใช้บริการที่มักใช้บ่อยๆ คือ GET, POST และ HEADโดยรายละเอียดการใช้คำสั่งสำหรับระบบการร้องขอนั้นจะมีการระบุรายละเอียดไว้ในตัวโพรโตคอล HTTP

การร้องขอบริการจะมีความเกี่ยวข้องกับการระบุถึงสถานที่ที่ต้องการใช้บริการและรายละเอียดที่อยู่ของข้อมูลต่างๆ ตามหลักการของ URL (Uniform Resource Locator) ซึ่ง URL คือส่วนระบุเพิ่มเติมในการเข้าถึงข้อมูลต่างๆได้ตามวิธีการที่ควรจะเป็นหรือเหมาะสม เช่น http://www.kmitl.ac.th, ftp://.abcd.com นอกจากนั้นยังสามารถที่จะเรียกใช้ในลักษณะของแอพพลิเคชั่นอื่นๆ ได้ เช่น mailto:s0010065@ce.kmitl.ac.th เป็นต้น

จากการร้องขอบริการข้างต้นจะเป็นการเรียกโฮมเพจที่ชื่อ “Index.html” โดยตรง ซึ่งอาจจะระบุแบบส่วนขยาย URL ได้ เช่น /~s0010065/indcx.html เป็นต้น ส่วนสำคัญถัดมาของการร้องขอบริการคือส่วนของ User Agent ซึ่งเป็นการระบุรายละเอียดเกี่ยวกับสถานภาพของไคลเอนต์ เช่น โปรแกรมที่ใช้งานเป็นบราวเซอร์ของ Firefox หรือ IE ใช้งานกับวินโดว์ เป็นต้น ส่วนต่อมาคือโฮสต์(Host) ซึ่งจะเป็นไซต์ปลายทางและรายละเอียดนี้จะสัมพันธ์กับส่วนเพิ่มของ URL ดังเช่นจากตัวอย่างจะเป็นการระบุ URL เท่ากับ http://161.246.10.21/index.html หรือ http://www.kmitl.ac.th/index.html ซึ่งในกรณีที่สองจะต้อง

มีการนำค่าชื่อโฮสต์ www.kmitl.ac.th ไปแปลงเป็นไอพีแอดเดรสเท่ากับ 160.246.10.21 และ GET จะได้เท่ากับ

/index.html

2.1.2 การตอบกลับ/การให้บริการ

ในการตอบกลับการให้บริการของ HTTP นั้นจะมีรูปแบบในการใช้งานเหมือนโพรโตคอลอื่นๆ ตามหลักการของไคลเอนต์-เซิร์ฟเวอร์ทั่วไป คือ จะมีการส่งค่ากลับนพด้วยเหมือนหมายเลขที่เรียกว่า "Response Tag Number"หรือ "Status Code" และตามด้วยรายละเอียดข้อความซึ่งอธิบายหมายเลขนั้น จากนั้นส่วนท้ายสุดที่อาจจะมีส่งตามมาคือตัวของข้อมูลจริงๆในกรณีที่มีการให้ข้อมูล เช่น จากการร้องขอข้างต้น เราจะได้รับข้อมูลที่เว็บเซิร์ฟเวอร์ส่งกลับมาให้ดังนี้

หมายเลขการตอบสนอง (Response Number) จะเป็นค่ามาตรฐานแต่ข้อความที่ตามมานั้นอาจจะไม่เหมือนกันก็ได้ทั้งนี้ขึ้นอยู่กับผลิตภัณฑ์ของเว็บเซิร์ฟเวอร์ ดังนั้นในการเขียนโปรแกรมประเภทไคลเอนต์-เซิร์ฟเวอร์ หรือหากจะเขียนโปรแกรมบราวเซอร์เราต้องอาศัยการตรวจสอบกาทำงานของไคลเอนต์-เซิร์ฟเวอร์จากหมาเลขดังกล่าวไม่ควรใช้ข้าความเพราะอาจผิดพลาดได้ เช่น

```
:    :    :    :
If(Sresponse_number eq"200"){
:    :    :    :
}
```

ตัวเลขทุกหลักของ Response Tags Number ล้วนมีความหมาย เริ่มที่หลักแรกจะเป็นการแสดงความมีประสิทธิภาพของการร้องขอและตอบกลับว่าเป็นอย่างไร "200 OK" หรือ"500 Error" เป็นต้น ส่วนตัวเลขหลักที่สองจะแจ้งให้ทราบถึงชนิดของการทำงาน เช่น หากดกิด Error เป็นชนิดไหนและส่วนหลักสุดท้ายเป็นส่วนย่อยในประเภทของการทำงานนั้นๆ เป็นการขยายในส่วนของรายละเอียดของหลักที่สอง

ดารางที่ 2.1 รายละเอียดของหมายเลขสถานะการทำงานของโพรโตคอล HTTP

100	Continue
101	Switching
200	OK
201	Created
202	Accepted
203	Non-Authoritative Information
204	No Content
205	Reset Content
206	Partial Content
300	Multiple Choices
301	Moved Permanently
302	Moved Temporarily
303	See Other
304	Not Modified
305	Use Proxy
400	Bad Request
401	Unauthorize
402	Payment Required
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required
408	Required Time-out
409	Conflict
410	Gone
411	Length Required
412	Precondition Failed
413	Request Entity Too Large
414	Request-URI Too Large

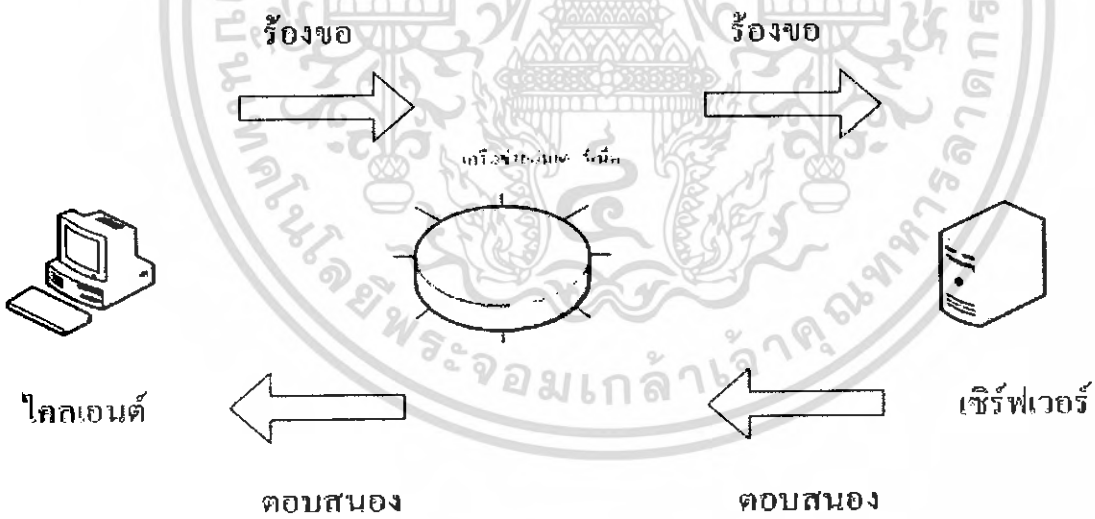
500	Internet Sever Error
501	Not Implemented
502	Bad Gateway
503	Device Unavailable
504	Gateway Time-out
505	HTTP Version not supported

2.1.3 การเชื่อมต่อระหว่างผู้ใช้บริการและผู้ให้บริการ

การเชื่อมต่อระหว่างผู้ใช้บริการและผู้ให้บริการสามารถแบ่งออกได้ 2 ลักษณะ คือ

2.1.3.1 การเชื่อมต่อโดยตรง

ลักษณะการทำงานก็คือ ผู้ขอใช้บริการจะติดต่อกับเซิร์ฟเวอร์หรือผู้ให้บริการโดยตรง ซึ่งส่วนของไคลเอนต์จะเรียกว่า User Agent โดยมีเว็บบราวเซอร์ทำหน้าที่นี้และส่วนเซิร์ฟเวอร์จะเรียกว่า origin ซึ่ง จะทำงานกับเว็บบราวเซอร์หรือ User Agent โดยตรง



รูปที่ 2.1 การเชื่อมต่อโดยตรง

2.1.3.2 การเชื่อมต่อผ่านตัวกลาง

ลักษณะการติดต่อแบบนี้ส่วนของ User Agent ไม่สามารถติดต่อกับ Origin ได้โดยตรง นั่นคือต้องติดต่อผ่านตัวกลางทุกครั้งที่มีการร้องขอบริการและการตอบสนองก็ต้องผ่านตัวกลาง เช่นกัน ดังนั้นการร้องขอหรือตอบสนองจะมีลักษณะเหมือนลูกโซ่โยงผ่านเป็นช่วงๆ เรียกว่า Request Chain/Response Chain ประเภทของตัวกลาง(Intermedia) ตามข้อกำหนดของHTTP มี 3ประเภท คือ

-Tunnel: ทำหน้าที่เชื่อมต่อเท่านั้น อาจจะมีไว้เพื่อความประสงค์อะไรบางอย่างแต่ตัวกลางนี้ จะไม่มีหน้าที่ หรือ อำนาจในการเปลี่ยนข้อมูลที่วิ่งผ่าน

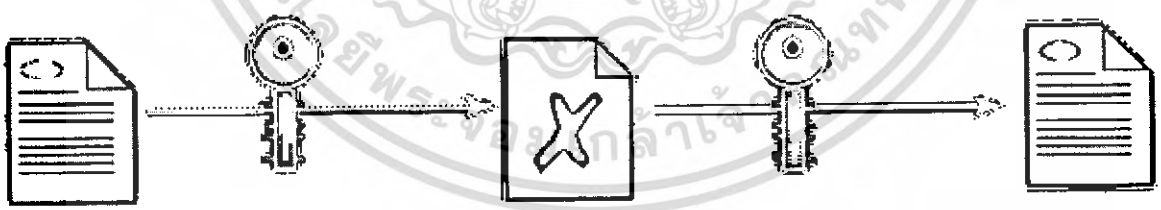
-Proxy: ส่วนนี้สามารถปรับปรุงรายละเอียด มีการประยุกต์ใช้งานได้ 2 ส่วน คือ ไคลเอนต์ และเซิร์ฟเวอร์ มักจะนำมาติดตั้งเป็น Cache Server หรือ Fire Wall

-Gateway: ส่วนนี้มักพบหน้าที่เชื่อมต่อ ในกรณีที่ไม่สามารถติดต่อหรือใช้งานกับเซิร์ฟเวอร์ได้โดยตรง

2.2 การเข้าและถอดรหัส(Cryptography)

2.2.1 การเข้ารหัสแบบคีย์เหมือน (Symmetric Encryption)

วิธีการเข้ารหัสโดยใช้กุญแจดัวเดียวในการเข้าและถอดรหัส คือระบบการเข้ารหัสลับและถอดรหัสด้วยกุญแจ(Secret Key)เพียงตัวเดียว หมายความว่ากุญแจที่ใช้ในการเข้าและถอดรหัสตัวเดียวกัน
สิ่งที่ควรสนใจสำหรับวีการข้างต้น คือทางฝั่งรับและส่ง ต้องรู้คีย์ตัวเดียวกัน ถ้ามีการส่งคีย์ข้ามเครือข่ายไปด้วยจะทำอย่างไรให้คีย์นั้นปลอดภัยการใช้งานนั้นรายละเอียดของกุญแจควรจะมีการปรับปรุงหรือเปลี่ยนบ่อยๆ ตัวอย่างของการเข้ารหัสด้วยคีย์เดียว คือ อัลกอริทึมของเดส (DES Algorithm)



รูปที่ 2.2 การเข้ารหัสแบบคีย์เหมือน

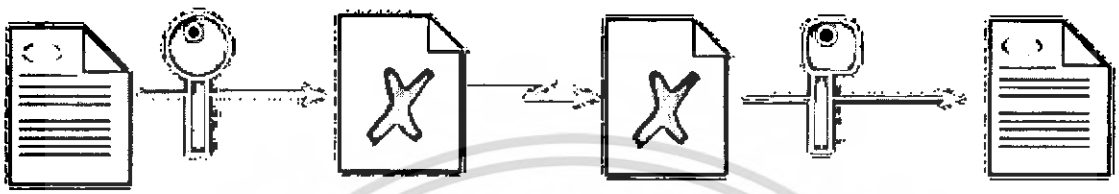
2.2.2 การเข้าและถอดรหัสแบบคีย์ต่างๆ(Asymmetric Encryption)

หลักการของพับลิกคีย์ได้ถูกคิดขึ้นมาเพื่อจะแก้ไขปัญหาที่เกิดในการเข้ารหัสแบบเดิมที่ได้กล่าวก่อนหน้านี้ ซึ่งปัญหาที่พบมีอยู่ 2อย่าง คือ

1. ปัญหาแรก คือ เกี่ยวกับการจัดสรรค์คีย์ในการเข้าและถอดรหัสแบบธรรมดาที่ไม่ใช้พับลิกคีย์ จะมีคีย์เพียงตัวเดียวซึ่งทั้งฝ่ายนับและฝ่ายส่งจะมีคีย์นี้ในการครอบครองซึ่งดูแล้วไม่เป็นส่วนตัว คีย์

สามารถรั่วไหลไปสู่ภายนอกได้ง่าย

2. ปัญหาที่สอง คือ เมื่อมีคนรู้คีย์แล้วนอกจากจะสามารถทำการถอดรหัสข้อมูลของเราได้แล้ว ยังสามารถที่จะทำการเข้ารหัสของเขาเองเมื่อรู้อัลกอริทึมและคีย์แล้วนำข้อมูลนั้นมาแทรกเข้ากับข้อมูลจริงๆ และถูกส่งไปยังผู้รับโดยไม่รู้ตัว ข้อมูลที่ถูกแทรกอาจเป็นโปรแกรมที่ไม่หวังดีก็ได้
เนื่องจากปัญหาทั้งสองนี้ Define และ Hellman ไตเสร้างพับลิกคีย์ขึ้นมาเพื่อจะแก้ปัญหาเหล่านี้



รูปที่ 2.3 การเข้ารหัสแบบคีย์ต่าง

วิธีการเข้าและถอดรหัสโดยใช้กุญแจสาธารณะ (Public Key Cryptography)จะมีลักษณะดังนี้คือ

1.แต่ละเอนด์ซิสเต็มในเครือข่ายจะสร้างคีย์มาเป็นคู่เพื่อที่จะทำการเข้าและถอดรหัส

2.คีย์ทั้งสองคีย์จะถูกเรียกว่า พับลิกคีย์ (Public Key) และไพรเวทคีย์(Private Key)

3.ถ้าA ต้องการที่จะส่งข้อมูลถึง B B จะต้องส่งพับลิกคีย์มาให้ A เพื่อใช้ในการเข้ารหัส จากนั้นก็จะส่งข้อมูลไปยัง B

4.เมื่อ B ได้รับข้อมูลก็จะทำการถอดรหัสโดยไพรเวทคีย์ของ B เองซึ่งมีเพียง B เท่านั้นที่รู้

ตารางที่ 2.2 เปรียบเทียบการเข้าและถอดรหัสแบบคีย์เหมือนกับการเข้าและถอดรหัสแบบคีย์ต่าง

การเข้าและถอดรหัสแบบคีย์เหมือน	การเข้าและถอดรหัสแบบคีย์ต่าง
1.ใช้อัลกอริทึมและคีย์เดียวกันในการเข้าและถอดรหัส	1.ใช้อัลกอริทึมเดียวกันในการเข้าและถอดรหัสแต่จะใช้คีย์คนละตัวกัน
2.ผู้รับและส่งจะต้องใช้อัลกอริทึมและคีย์ตัวเดียวกัน	2.ผู้รับและส่งจะต้องมีคีย์ที่เป็นความลับ 1 คีย์ คือ ไพรเวทคีย์
3.คีย์จะเป็นตัวเก็บความลับ	3.มีเพียงคีย์เดียวใน2 คีย์ที่เป็นตัวเก็บความลับ
4.ความรู้เฉพาะอัลกอริทึมและไซเฟอร์เท็กซ์ไม่เพียงพอที่จะใช้หาคีย์	4.ความรู้เรื่องอัลกอริทึมบวกกับคีย์ 1 ตัวบวกไซเฟอร์เท็กซ์ ไม่สามารถที่จะนำไปสู่การกาคีย์ต่อไปได้

จากลักษณะที่สำคัญของการเข้าและถอดรหัสโดยใช้กุญแจสาธารณะที่กล่าวมานั้นสามารถนำมาออกแบบโครงสร้างที่จะนำไปสู่การเข้าและถอดรหัส ซึ่งจะกล่าวใน 3 ลักษณะดังนี้

2.2.2.1 การเข้าและถอดรหัสโดยใช้กุญแจสาธารณะ : ปกปิดความลับ

วิธีการนี้แสดงให้เห็นถึงการเก็บความลับของไพรเวตคีย์ สมมติว่า A ต้องการส่งข้อมูลไปยัง B ข้อมูลก็จะทำการถอดรหัสโดยใช้พับลิกคีย์ของ B ซึ่ง B จะส่งมาให้เมื่อทำการเข้ารหัสเสร็จ A ก็จะส่งข้อมูลไปยัง B จากนั้น B ก็จะทำการถอดรหัสโดยใช้ไพรเวตคีย์ของ B เอง ซึ่งจะเห็นได้ว่า ไพรเวตคีย์ของBไม่มีทางที่จะแพร่งพรายออกสู่ภายนอกได้แต่ถ้าเป็นนักวิเคราะห์การเข้ารหัส(Cryptanalyst)สามารถที่จะเข้าไปเอาพับลิกออกมาได้ในตอนที่ B ส่งมาให้ A ก็จะรู้อัลกอริทึมไซเฟอร์เท็กซ์และสามารถที่จะนพเอาทั้งหมดนี้มาทำการรหัสโดยการเดาไพรเวตคีย์จนกระทั่งข้อความที่ถูกแปลออกมาอ่านรู้เรื่อนอกจากนี้นักวิเคราะห์การเข้ารหัสข้อมูลของตัวเองโดยใช้อัลกอริทึมและคีย์ที่กามาได้และทำการแทรกข้อมูลนั้นไปพร้อมๆ กับข้อมูลจริง ได้ ข้อมูลที่ถูกแทรกอาจเป็นอันตรายต่อระบบได้

2.2.2.2.การเข้าและถอดรหัสโดยใช้กุญแจสาธารณะ : การพิสูจน์ตน

วิธีที่จะแก้ปัญหาในข้อแรกคือ ไม่ให้นักวิเคราะห์การเข้ารหัสรู้ถึงคีย์ที่ใช้ในการเข้ารหัส และจะส่งกุญแจสาธารณะไปยังปลายทางเพื่อที่จะทำการถอดรหัส แสดงดังรูปที่ 2-8 ซึ่งก็ทำให้ไม่สามารถที่จะแทรกข้อมูลเข้าไปได้เพราะรู้แต่อัลกอริทึมแต่ไม่รู้ถึงคีย์ที่ใช้ในการเข้ารหัส แต่ข้อเสียก็คือกุญแจสาธารณะที่ถูกส่งไปยังปลายทางนักวิเคราะห์การเข้ารหัสสามารถที่จะล่วงรู้ได้ และเมื่อรู้อัลกอริทึมและไวเฟอร์เท็กซ์ก็สามารถนำไปสู่การหาไพรเวทคีย์ได้ ซึ่งไม่ปลอดภัย

2.2.2.3การเข้าและถอดรหัสโดยใช้กุญแจสาธารณะ : ปกปิดความลับและการพิสูจน์ตน

วิธีนี้จะเป็นการรวมกันระหว่างวิธีที่1และ2 เพื่อที่จะนำข้อดีของทั้งสองมาและกำจัดข้อเสียของทั้งสองแบบออก แสดงในรูปที่2-9 แต่วิธีนี้ก็มีข้อเสีย คือจะต้องทำการเข้ารหัสถึง 2 ครั้ง และเมื่อเวลาจะทำการถอดรหัสก็ต้องทำถึง 2 ครั้งเช่นกัน ทำให้ต้องใช้เวลาในส่วนนี้เพิ่มขึ้น

2.3 โพรโตคอล SSL

SSLเป็นโพรโตคอลที่ได้รับการพัฒนาจากบริษัทNetscapeและได้รับการเสนอให้เป็นมาตรฐานอุตสาหกรรมโดยองค์การเฉพาะกิจวิศวกรรมอินเตอร์เน็ต (IETF : Internet Engineering Task Force) โดย SSL ถูกออกแบบมาเพื่อใช้งานด้านการรักษาความปลอดภัยในตัวข้อมูล ซึ่งหากเปรียบเทียบกับ 7 เลเยอร์ กับ TCP/IP แล้วตัว SSL จะทำงานอยู่ระหว่างเลเยอชั้นสูง ซึ่งคือชั้น Application และ TCP/IP หรือเลเยอร์ที่เกี่ยวข้องกับการรับส่งข้อมูล (Transport)

กล่าวได้คือ การใช้งาน TCP/IP จะใช้ซอกเก็ต (Socket) ในการติดต่อสื่อสาร ดังนั้น SSL จึงทำการเข้ารหัสข้อมูลที่ใช้งานทั้งหมดจากเลเยอร์บนแล้วรับผ่าน ซอกเก็ตในเลเยอร์ล่าง นั่นคือ SSL สามารถใช้ได้กับโปรแกรมประยุกต์ทุกตัวที่ทำงานบนโปรโตคอล TCP/IP เช่น ในกรณีของเว็บคือ HTTP ในปัจจุบัน

ละเครือข่าย TCP/IP นั้นจะมีขั้นตอนการทำงานหลักๆ ดังนี้

-การพิสูจน์ตนของ SSL ณ เซิร์ฟเวอร์ (SSL server authentication): จะอนุญาตหรือให้ผู้ใช้สามารถตรวจสอบว่าเซิร์ฟเวอร์ที่ติดต้อเป็นเซิร์ฟเวอร์ที่ผู้ใช้ต้องการติดต่อด้วย โดย SSL-enabled Client S/W สามารถใช้เทคนิคมาตรฐานของการเข้าและถอดรหัสโดยใช้กุญแจสาธารณะ ในการตรวจสอบว่า certificate และ Public ID ของเซิร์ฟเวอร์ถูกต้องจริงและออกแบบให้โดย CA (Certificate authority) ซึ่งการแสดงการยืนยันนี้จะสำคัญต่อผู้ใช้ เช่น การที่ผู้ใช้จำเป็นต้องส่งหมายเลขบัตรเครดิตผ่านทางเครือข่าย (Network) และต้องการตรวจสอบและพิสูจน์เซิร์ฟเวอร์นั้น

-การพิสูจน์ตนของ SSL ณ ไคลเอนต์ (SSL client authentication): โดยเซิร์ฟเวอร์สามรถตรวจสอบผู้ใช้วิธีเดียวกับข้างบน นั่นคือ SSL-enabled Client S/W สามารถตรวจสอบว่า certificate และ ID ของผู้ใช้ว่าถูกต้องและใช้ได้และออกให้โดย CA ซึ่งการตรวจสอบนี้อาจจำเป็น เช่น เซิร์ฟเวอร์ต้องการตรวจสอบว่าผู้ใช้ที่ตนติดต่อด้วยจริงก่อนที่จะส่งข่าวสารเกี่ยวกับการเงินไปให้ผู้ใช้นั้นๆ

-การเข้ารหัสในการเชื่อมต่อแบบ SSL (An encrypted SSL connection): จะทำการเข้ารหัสข้อมูลข่าวสารที่ส่งระหว่างไคลเอนต์และเซิร์ฟเวอร์ (sending software ใช้ในการเข้ารหัสและ receiving software ใช้ในการถอดรหัส) ข้อมูลข่าวสารที่ส่งบน encrypted SSL connection จะถูกป้องกันโดยกลไกในการป้องกัน นั่นคือ จะทราบได้โดยอัตโนมัติว่าข้อมูลถูกเปลี่ยนแปลงหรือไม่ในระหว่างการส่ง

2.3.1 องค์ประกอบของSSL

-SSL record Protocol: กำหนดรูปแบบที่ใช้ในการส่งข้อมูล

-SSL handshake Protocol: จะเกี่ยวข้องกับการใช้ SSL record Protocol ในการแลกเปลี่ยนข้อความ ระหว่าง SSL-enable client ในตอนเริ่มต้นขั้นคอนการติดตั้งการเชื่อมต่อแบบ SSL การแลกเปลี่ยนข่าวสารถูกออกแบบมาเพื่อ

1) ตรวจสอบเซิร์ฟเวอร์ที่ไคลเอนต์ติดต่อ

2)ช่วยให้ไคลเอนต์และเซิร์ฟเวอร์แลกเปลี่ยนอัลกอริทึมที่ใช้ในการเข้าและถอดรหัสข้อมูลที่สนับสนุนการทำงาน

3) ตรวจสอบไคลเอนต์ที่เซิร์ฟเวอร์ดิดต่อ

4) การใช้เทคนิคการเข้ารหัสโดยใช้กุญแจสาธารณะในการสร้าง secret key ที่ใช้ร่วมกัน

5) ดิตตั้งการติดต่อสื่อสารแบบ SSL

2.3.2 การเข้าและถอดรหัสใน SSL(Ciphers Used with SSL)

SSL สนับสนุนอัลกอริทึมที่ใช้ในการที่ใช้ในการเข้าและถอดรหัสหลายๆชนิด ดังนี้

-DES(Data Encryption Standard) เป็นการเข้ารหัสใช้โดยรัฐบาลของประเทศสหรัฐอเมริกา

-DSA (Digital Signal Algorithm) เป็นส่วนหนึ่งของมาตรฐานของการตรวจสอบลายเซ็นดิจิตอล (Digital Authentication)

-RSA เป็นอัลกอริทึมที่ใช้ในการเข้าและถอดรหัสและการยืนยันตัวบุคคล

-Tripple-DES มีการทำงานที่คล้ายกับแบบ DES แต่จะมีการเข้ารหัสถึง 3ครั้ง

-KEA (Key Exchange Algorithm)

-MD5(Message Digest)

-RC2/RC4 (Rivest Encryption Ciphers)

-RSA Key Exchange

เป็นต้น

2.3.3การตกลงเชื่อมต่อใน SSL (SSL Handshake)

โพรโตคอล SSL จะใช้เทคนิคของการเข้ารหัสข้อมูลโดยใช้กุญแจสาธารณะและแบบคีย์เหมือน(Symmetric) ซึ่งการเข้ารหัสแบบกุญแจสาธารณะจะใช้เทคนิคในการตรวจสอบและยืนยันตัวบุคคล (Authentication) ที่ดีหว่าแบบคีย์เหมือน ในขั้นตอนการสื่อสารแบบ SSL จะเริ่มต้นด้วยการแลกเปลี่ยนข้อความ ที่เรียกว่า การตกลงการเชื่อมต่อนั้นจะอนุญาตให้เซิร์ฟเวอร์และไคลเอนต์ร่วมกันสร้างกุญแจหรือ Session Key ที่ใช้ในการส่งข้อมูลแก่กัน ซึ่งขั้นตอนการตกลงการเชื่อมต่อในการแลกเปลี่ยนข้อมูลมีดังนี้

1.ไคลเอนต์ส่งหมายเลขเวอร์ชั่นของ SSL (SSL Version Number) ของไคลเอนต์ / รูปแบบการเข้า/ถอดรหัส/วิธีการบีบอัดข้อมูลและข้อมูลอื่นๆ ให้แก่เซิร์ฟเวอร์ที่เซิร์ฟเวอร์ต้องการติดต่อกับไคลเอนต์โดยใช้ SSL

2. เซิร์ฟเวอร์ส่งหมายเลขเวอร์ชั่นของ SSL ของตัวเอง / รูปแบบการติดต่อ / วิธีการบีบอัดข้อมูล และ certificate ไปให้ไคลเอนต์เพื่อยืนยันตัวตนว่าเป็นเซิร์ฟเวอร์ซึ่งตนติดต่อด้วยจริง

3.ไคลเอนต์จะใช้ข้อมูลที่ได้รับมาจากเซิร์ฟเวอร์มาตรวจสอบเพื่อยืนยันว่าเซิร์ฟเวอร์นั้นจริง ถ้าเซิร์ฟเวอร์นั้นไม่สามารถถูกตรวจสอบได้ผู้ใช้จะได้รับการเตือนว่าเกิดปัญหาขึ้นและแสดงว่าการเข้ารหัส (Encrypted) และการยืนยันตัวบุคคล (Authentication) ไม่สามารถติดตั้งได้

4. ใส่ข้อมูลที่ทำการเชื่อมต่อ จากนั้นไคลเอนต์จะสร้าง “premaster secret” เพื่อสร้างเส้นทางการสื่อสารติดต่อและทำการเข้ารหัสโดยใช้กุญแจสาธารณะของเซิร์ฟเวอร์(ได้รับจากเซิร์ฟเวอร์ในข้อ 2) และทำการส่ง premaster secret ที่ถูกทำการเข้ารหัสให้เซิร์ฟเวอร์

5. ในกรณีที่เซิร์ฟเวอร์ต้องการตรวจสอบว่าไคลเอนต์นี้เป็นไคลเอนต์จริงหรือไม่ ไคลเอนต์ก็จะส่ง certificate ของไคลเอนต์ไร้อมกับ premaster secret ที่เข้ารหัสให้แก่เซิร์ฟเวอร์ไร้อมๆกัน

6. เซิร์ฟเวอร์ทำการตรวจสอบ certificate ของไคลเอนต์นั้น ถ้าการตรวจสอบล้มเหลวการสร้างเส้นทางการสื่อสารจะจบลงทันที ถ้าการตรวจสอบถูกต้องเซิร์ฟเวอร์จะใช้ไพรเวทคีย์ของตนทำการถอดรหัส premaster secret และทำการสร้าง master key ขึ้นมา

7. ไคลเอนต์และเซิร์ฟเวอร์จะใช้ master key สร้าง session key ซึ่งเป็น symmetric keys ที่ใช้ในการเข้าและถอดรหัส ข้อมูลข้าวสารที่ใช้แลกเปลี่ยนในช่วงการติดต่อสื่อสารของ SSL และใช้ในการตรวจสอบความถูกต้อง นั่นคือจะตรวจสอบการเปลี่ยนแปลงของข้อมูล(Data) ในช่วงเวลาที่มันส่งและใน

ช่วงเวลาที่รับข้อมูลผ่านการติดต้อแบบ SSL

8. ไคลเอนต์ส่งข้อความไปบอกแก่เซิร์ฟเวอร์ว่าข้อความถัดไปที่ไคลเอนต์จะส่งไปให้ จะถูกเข้ารหัสโดย session key จากนั้นไคลเอนต์จะส่งข้อความ(ที่ถูกเข้ารหัสโดย session key) เพื่อบอกแก่เซิร์ฟเวอร์ว่าในส่วนไคลเอนต์นั้นทำการตกลงเชื่อมต่อเสร็จสิ้น

9. เซิร์ฟเวอร์ส่งข้อความไปบอกแก่ไคลเอนต์ว่าข้อความต่อไปที่จะส่งมาเข้ารหัสโดย session key และเซิร์ฟเวอร์จะทำการส่งข้อความที่เข้ารหัสไว้(โดย session key) ไปให้แก่ ไคลเอนต์เพื่อบอกว่าการตกลงการเชื่อมต่อในส่วนของเซิร์ฟเวอร์เสร็จสิ้นแล้ว

10. การตกลงการเชื่อมต่อใน SSL เสร็จสิ้นและช่องการสื่อสารแบบ SSL ติดตั้งเสร็จและเริ่มทำงาน ไคลเอนต์และเซิร์ฟเวอร์จะใช้ session key ในการเข้าและถอดรหัสข้อมูลที่ไคลเอนต์และเซิร์ฟเวอร์ส่งระหว่างกันและกันและใช้ตรวจสอบความถูกต้อง

2.3.4 การยืนยันตัวเซิร์ฟเวอร์(Server Authentication)

เป็นการที่ไคลเอนต์ทำการตรวจสอบเซิร์ฟเวอร์โดยให้เซิร์ฟเวอร์ทำการยืนยันตัวเองเหมือนกันกับในขั้นตอนที่2 ในการตกลงเชื่อมต่อใน SSL โดยเซิร์ฟเวอร์ทำการส่ง certificate ของตัวเองไปให้แก่ไคลเอนต์และเมื่อได้รับข้อมูลเหล่านั้นก็จะทำการตรวจสอบดังในขั้นตอนที่ 3 ของการทำการตกลงการเชื่อมต่อใน SSL

ในการตรวจสอบการยืนยันที่ฝั่งเซิร์ฟเวอร์มีขั้นตอนดังนี้

1.ไคลเอนต์จะทำการตรวจสอบถึงช่วงเวลาที่สามารถใช้งานได้ของ certificate ของเซิร์ฟเวอร์ ซึ่งถ้าการตรวจสอบว่าช่วงวันและเวลาทำงานของ certificate ของเซิร์ฟเวอร์ยังสามารถใช้ได้ ก็จะทำการตรวจสอบในขั้นตอนต่อไป

2.ไคลเอนต์จะทำการตรวจสอบองค์กรที่ทำการออกใบ certificate ของเซิร์ฟเวอร์ว่าเป็นองค์กรที่เชื่อถือได้หรือไม่ ซึ่งในกรณีที่เป็นองค์กรที่เชื่อถือได้ก็จะทำการตรวจสอบในขั้นที่ 3 แต่ถ้าองค์กรที่ไม่สามารถเชื่อถือได้ก็จะไม่ทำการยืนยันในขั้นต่อไป

3.ไคลเอนต์จะใช้กุญแจสาธารระที่จาใบรับรองของ CA เพื่อตรวจสอบลายเซ็นดิจิตอล(Digital Signature) ของ CA บนcertificate ของเซิร์ฟเวอร์ ในกรณีที่ข้อมูลในลายเซ็นในลายเซ็นดิจิตอลได้รับการเปลี่ยนแปลงหรือกุญแจสาธารณะ ของCA บนcertificate ไม่สอดคล้องกับไพรเวทคีย์ที่ใช้โดยCA ในการสร้าง certificatc ไคลเอนต์ก็จะไม่รับรองการยืนยันของเซิร์ฟเวอร์

4.เป็นขั้นตอนการยืนยันว่าเซิร์ฟเวอร์ที่ไคลเอนต์กำลังทำการติดต่อด้วยนั้นมีโดเมนเนม (Domain Name) ภายในเครือข่ายตรงกับโดเมนเนมในใบ certificate หรือไม่ ซึ่งถ้าตรง แสดงว่าเป็นเซิร์ฟเวอร์ที่สามารถเชื่อถือได้ ดังรูปที่ 2-10

2.3.5 การยืนยันตัวไคลเอนต์(Client Authentication)

ในกรณีที่เซิร์ฟเวอร์ต้องการตรวจสอบหรือยืนยันจากไคลเอนต์โดยเซิร์ฟเวอร์จะร้องขอไปยังไคลเอนต์(ดูในขั้นตอนที่6 ของการติดต่อ SSL) โดยไคลเอนต์จะทำการส่ง certificate ของตัวเองและส่วน

ของลายเซ็นดิจิตอลไปให้แก่เซิร์ฟเวอร์ จากนั้นเซิร์ฟเวอร์จะทำการตรวจสอบโดยใช้ลายเซ็นดิจิตอลมาตรวจสอบความถูต้องของกุญแจสาธารณะใน certificate ที่ไคลเอนต์ส่งมา

โพรโตคอล SSL จะร้องขอให้ไคลเอนต์ทำการสร่งลายเซ็นดิจิตอลโดยวิธี one-way hash โดยใช้ข้อมูลที่ส่งมาในขณะที่ทำการตกลงการเชื่อมต่อโดยรู้กันเฉพาะเซิร์ฟเวอร์กับไคลเอนต์เท่านั้น ส่วนของข้อมูลที่ถูกทำการบีบอัด (Digest) จะถูกเข้ารหัสโดยไพรเวทคีย์ของฝั่งเซิร์ฟเวอร์ที่สอดคล้องกับกุญแจสาธารณะใน certificate ที่ส่งไปให้แก่เซิร์ฟเวอร์ โดยมีขั้นตอน ดังนี้

1.เว็บเซิร์ฟเวอร์จะทำการตรวจสอบลายเซ็นดิจิตอลของเว็บไคลเอนต์ว่าสามรถใช้กับพับลิกคีย์ใน ใบรับรองดิจิตอลหรือไม่ ถ้าใช้ได้เว็บเซิร์ฟเวอร์จะสร้างดารยืนยันพับลิกคีย์นี้เป็นของ John Doe ตามรูปที่ 2-11ซึ่งตรงหับไพรเวทคีย์ที่ใช้ในการสร้างลายเซ็นดิจิตอลและข้อมูลไม่ได้ถูกดักหรือเปลี่ยนแปลงตั้งแต่เริ่มทำการสร้างลายเซ็นดิจิตอล

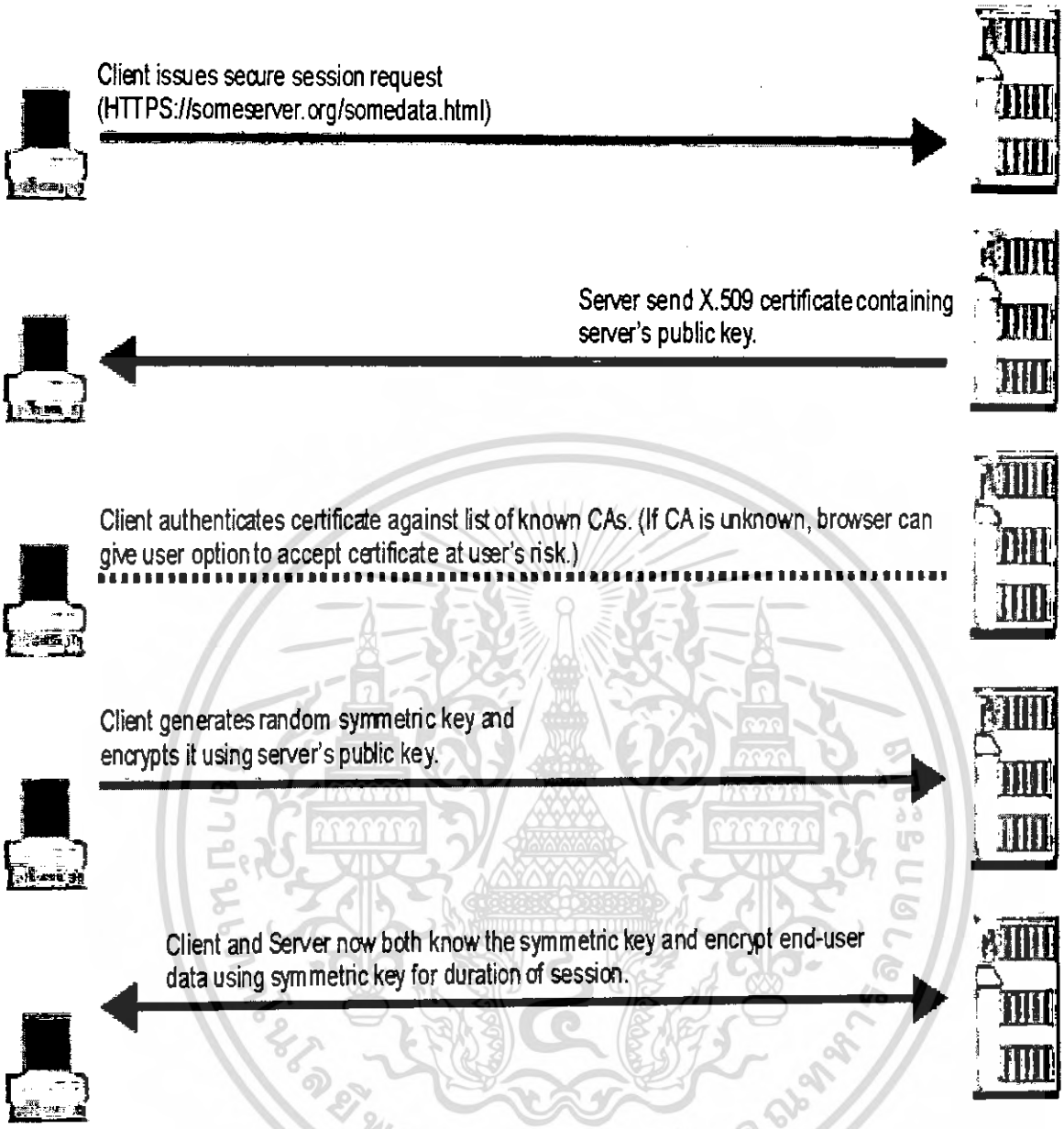
2. เว็บเซิร์ฟเวอร์จะทำการตรวจสอบช่วงเวลาที่ใช้งานได้ของใบรับรองดิจิตอลของเว็บไคลเอนเพื่อตรวจสอบว่าวันเวลาที่เว็บเซิร์ฟเวอร์ติดต่อกับเว็บไคลเอนต์นี้อยู่ในช่วงเวลาที่กำหนดเอาไว้หรือไม่ ถ้าไม่อยู่ การยืนยันการตรวจสอบจะไม่ทำอีกต่อไป แต่ถ้าอยู่ก็จะไปยังขั้นตอนที่ 3

3. แต่ละ SSL-enabled ของเว็บเซิร์ฟเวอร์จะมีรายชื่อใบรับรองดิจิตอลของ CA ที่น่าเชื่อถือ (อยู่ในรูปที่ 2-11 ในส่วนที่แรเงาของเว็บเซิร์ฟเวอร์) โดยรายชื่อนี้จะแสดงใบรับรองดิจิตอลทั้งหมดที่เว็บเซิร์ฟเวอร์สามรถรับรองได้ ซึ่งถ้า DN(Distinguish Name) ของ CA บนใบรับรองดิจิตอลของเว็บไคลเอนต์ตรงกับรายชื่อใบรับรองดิจิตอลของ CA ที่เว็บเซิร์ฟเวอร์ จะถือว่าเชื่อถือได้ แต่ถ้าไม่มีอยู่ในใบรายชื่อ การตรวจสอบการยืนยันของเว็บไคลเอนต์จะถือว่าล้มเหลว

4. เว็บเซิร์ฟเวอร์จะใช้พับลิกคีย์ที่ได้มาจากใบรับรองของ CA (ที่มีอยู่ในลิส) เพื่อตรวจสอบลายเซ็นดิจิตอลของ CA บนใบรับรองดิจิตอลของเว็บไคลเอนต์ ในกรณีที่ข้อมูลภายในลายเซ็นดิจิตอลได้รับการเปลี่ยนแปลงหรือพับลิกคีย์ในใบรับรองดิจิตอลของ CA ไม่ตรงกันหรือไม่สอดคล้องกับไพรเวทคีย์ที่ใช้โดย CA ในการสร้างใบรับรองดิจิตอล เว็บเซิร์ฟเวอร์ก็จะไม่รับรองการยืนยันของเว็บไคลเอนต์

5.เป็นอีกหนทางหนึ่งสำหรับผู้ดูแลระบบในการถอนใบรับรองของเว็บไคลเอนต์ถึงแม้จะผ่านการตรวจสอบในขั้นตอนต่างๆ แล้วก็ตามโดยถ้าใบรับรองดิจิตอลของเว็บไคลเอนต์ในไดเร็กทอรี่ (Directory) ตรงกับใบรับรองดิจิตอลของเว็บไคลเอนต์ในการทำการตกลงเชื่อมต่อใน SSL ให้จะไปยังขั้นตอนที่6 แต่ใบรับรองดิจิตอลของเว็บไคลเอนต์ไม่อยู่ในไดเร็กทอรี่ ความน่าเชื่อถือ(Trusted) ของเว็บไคลเอนต์นั้นก็จะน้อยลง

6.เว็บเซิร์ฟเวอร์ทำการตรวจสอบว่าทรัพยากร(Resource)ใดหรือส่วนที่เว็บไคลเอนต์ได้รับอนุญาตในการเข้าใช้ ตาม ACLs (Access Control Lists) และทำการเชื่อมต่อการสื่อสารในรูปแบบที่เหมาะสม



2.4 การทำงานของ SSL

2.4 กุญแจสาธารณะ(PKI)

โครงสร้างมูลฐานกุญแจสาธารณะ(PKI) เป็นระบบที่ใช้สำหรับการเผยแพร่กุญแจสาธารณะและข้อมูลของคีย์นั้นในกระบวนการของการเข้าและถอดรหัสโดยใช้กุญแจสาธารณะเพื่อความปลอดภัยของข้อมูลที่ทำการรับส่งกันบนระบบเครือข่าย โดย PKI จะเอื้อประโยชน์ต่างๆ ดังนี้

- การจัดการคีย์(Manage Keys):ช่วยให้การสร้างคีย์ การแก้ไขและระงับการใช้คีย์สามารถทำได้โดยง่าย

- เผยแพร่คีย์(Public Keys):ช่วยให้เกิดความสะดวกสบายในการรับส่งคีย์และข้อมูลของคีย์ระหว่างการสื่อสาร

- การใช้งานคีย์(User Keys):ช่วยให้ผู้ใช้สามารถใช้คีย์ได้ง่ายผ่านทางแอพพลิเคชั่นต่างๆ

2.4.1โครงสร้างของ PKI

โครงสร้างของ PKIจะประกอบด้วย

1.ใบรับรองดิจิตอล(Digital Certificate):ใบรับรองดิจิตอลตามมาตรฐาน X.509 ซึ่งเป็นมาตรฐานสำหรับกำหนดรูปแบบของใบรับรองดิจิตัล สามารถนำไปใช้ในด้านต่างๆ เช่น Client Certificate, Server Certificate และ Email Certificate เป็นต้น

2. Certificate Authority (CA):เป็นองค์กรที่ทำหน้าที่ออกใบรับรองดิจิตอล ซึ่งสามารถแบ่งได้ 2 ประเภท ตามลักษณะขององค์กรคือ

- Internal CA คือ CA ที่ติดตั้งขึ้นมาภายในองค์กรและทำหน้าที่ออกใบรับรองดิจิตอลให้กับเฉพาะบุคคลและหน่วยงานภายในองค์กร

-Trusted Third Party คือ CA ที่เป็นองค์กรภายนอกหรือที่เรียกว่าองค์กรที่สาม ซึ่งทำหน้าที่ออกใบรับรองให้กับบุคคลและองค์กรอื่นๆ ทั่วไป โดยองค์กรเหล่านั้นต้องมีความน่าเชื่อถือสูง เช่น Verisign, Inc.,US Postal Service และ Thawte เป็นต้น

3. Registration Authority (RA): เป็นหน่วยงานที่ทำหน้าที่ตรวจสอบบุคคลที่มาทำการขอใบรับรองดิจิตอล ซึ่งบางครั้งอาจจะถูกรวมไว้กับ CA ก็ได้

4. Directory Service: เป็นที่เก็บข้อมูลของผู้ร้องขอใบรับรองดิจิตอลรวมทั้งกุญแจสาธารณะด้วย

5. Software: เป็นโปรแกรมหรือแอพพลิเคชั่นที่ช่วยในการจัดการและสนับสนุนการใช้งานใบรับรองดิจิตอล

2.4.2 กลไกการทำงานของ PKI

กลไกการทำงานของ PKI แบ่งได้ 2 ลักษณะ ด้วยกัน

-Certification: เป็นกระบวนการเชื่อมกุญแจสาธารณะเข้ากับข้อมูลต่างๆ ของเจ้าของคีย์นั้น เช่น ข้อมูลส่วนตัว องค์กร และสิทธิในการใช้งาน เป็นต้น

-Validation: เป็นกระบวนการในการตรวจสอบว่าใบรับรองดิจิตอลนั้นยังมีความถูกต้อง

หรือไม่

2.4.2.1 Certification

Certification เป็นหน้าที่พื้นฐานของ PKI เกี่ยวหับวิถีทางในการเผยแพร่หรือกระจายค่าของกุญแจสาธารณะและข้อมูลที่เกี่ยวข้องกับคีย์นั้น ดังนั้นอาจกล่าวได้ว่าใบรับรองดิจิตอล ก็คือกุญแจสาธารณะและข้อมูลของคีย์นั่นเอง นอกจากนี้คุณลักษณะสำคัญของใบรับรองดิจิตอลอีกประการหนึ่งคือจะมีการลงลายมือชื่อดิจิตอลของผู้ออกอยู่ด้วย โดยการลงลายมือชื่อดิจิตอลของผู้ออกจะอาศัยการเข้ารหัสโดยใช้ไพรเวทคีย์ของผู้ออกเพื่อยืนยันว่าใบรับรองดิจิตอลถูกออกโดยผู้ออกดังกล่าวจริง

ผู้ที่ทำหน้าที่ออกใบรับรองดิจิตอลนั้นเราจะเรียกว่า Certificate Authority (CA) สามารถแสดงตัวอย่างในการใช้งาน PKI ซึ่งเกี่ยวกับ Certification ได้ดังนี้

สมมติว่า Alice ต้องการสร้างช่องทางการสื่อสารที่ปลอดภัยกับ Bob โดยใช้กระบวนการเข้าและถอดรหัสโดยใช้กุญแจสาธารณะ ซึ่ง Ash จำเป็นจะต้องทราบกุญแจสาธารณะของ Bob ถ้าไม่มีการใช้ PKI Alice จำเป็นจะต้องมีความรู้โดยตรงเกี่ยวกับกุญแจสาธารณะของ Bob เช่น Bob ต้องการส่งมาให้ Alice ผ่านทางช่องทางการสื่อสารที่ปลอดภัยและถ้า Ash ต้องการติดต่อกับผู้อื่น เช่น Edd Alice ก็จำเป็นจะต้องได้รับกุญแจสาธารณะของ Edd โดยตรงก่อนเช่นกัน แต่เมื่อมีการประยุกต์ใช้ PKI Aliceเพียงแต่มีความรู้เกี่ยวกับกุญแจสาธารณะของ CA เท่านั้น โดย CA จะออกใบรับรองดิจิตอลสำหรับกุญแจสาธารณะของ Bob และ Edd ดังนั้น เมื่อ Alice ต้องการจะสร้างช่องทางการสื่อสารที่ปลอดภัยกับ Bob และ Edd Alice ก็จะสามารถได้กุญแจสาธารณะที่ถูกต้องจากใบรับรองดิจิตอลของบุคคลทั้งสอง โดยเราจะเรียก Alice ว่า Certificate User ในขณะที่ Bob และ Edd จะเป็น Certificate Subject

หากจะกล่าวถึงความสัมพันธ์ระหว่าง CA, Certificate User และ Certificate Subject แต่ละฝ่ายจะเป็น Entity เดียวกัน ซึ่งอาจจะเกี่ยวข้องกันหรือไม่ก็ได้ และความเชื่อถือระกว่างทั้งสามฝ่ายก็เป็นคุณลักษณะประการหนึ่งของ PKI จากตัวอย่างข้างต้น Alice ต้องมีความเชื่อถือในใบรับรองดิจิตอลของ CA แต่ถ้า Alice และ CA ไม่เกี่ยวข้องกัน Alice จะเชื่อถือ CA นั้นได้อย่างไร ซึ่งจะกล่าวต่อไป

2.4.2.1.1 CA Arrangement

เนื่องจากเป็นไปไม่ได้ที่จะมี CA เพียงองค์กรเดียวเพื่อให้บริการใบรับรองดิจิตอลแก่บุคคลหรือหน่วยงานทั้งหมด ดังนั้น PKI จึงอนุญาตให้ CA สามารถทำการรับรอง CAs อื่นๆได้ กล่าวคือ CA กำลังบอกผู้ใช้ว่าพวกเขาสามารถให้ความเชื่อถือ CA อื่น (Second CA) ซึ่งได้ระบุไว้ในใบรับรองดิจิตอลของ CA ได้จากตัวอย่างข้างต้น Alice, Bob และ Edd จะมีใบรับรองดิจิตอลของตนเองที่ออกให้โดย CA ต่างๆกัน เมื่อ Alice ต้องการจะติดต่อกับBob Alice จะต้องรับรู้ถึงใบรับรองดิจิตอลของ CA ของ Bob หรือใบรับรองดิจิตอลของ CA ของ Alice ถ้า CA ของAlice เป็นผู้ออกใบรับรองดิจิตอลให้ CA ของ Bob ซึ่งในกรณีหลังนี้ Alicc จะได้กุญแจสาธารณะของ Bob อย่างปลอดภัยเพียงแค่ทราบใบรับรองดิจิตอลของ CA ของ Alice เราจะเรียกใบรับรองดิจิตอลที่ออกให้แก่ Alice และ Bob ว่า “End-user Certificate” และเรียกใบรับรองดิจิตอลที่ออกโดย CA ของ Alice ให้แก่ CA ของ Bob ว่า “CA

Certificate"

โดยปกติแล้วจะมี CA อื่นๆ ระหว่างการติดต่อสื่อสารระหว่าง Alice และ Bob

การที่จะได้มาซึ่งกุญแจสาธารณะของ Bob Alice ต้องทำการตรวจสอบใบรับรองดิจิตอลของ CA ต่างๆ ตลอดเส้นทางจนกว่าจะได้รับใบรับรองดิจิตอลของ Bob โดยกระบวนการนี้เรียกว่า "Certificate Path Validation" ความยาวของ Certificate Path จะเท่ากับจำนวนของ CA ระหว่าง Alice และ Bob หรือเท่ากับไนวนใบรับรองดิจิตอลที่ Alice ต้องใช้ตรวจสอบจนกว่าจะได้กุญแจสาธารณะของ Bob จากรูปที่ 2-13 จะมีทั้งหมด 3 ใบรับรองดิจิตอล กล่าวคือ ใบรับรองดิจิตอลที่1 เป็น CA Certificate ที่ออกโดย CA X ให้แก่ CA Y ออก CA Certificate ให้แก่ CA Z ซึ่งเป็นผู้ออก end user Certificate ให้กับ Bob โดย Alice จะต้องเริ่มต้นด้วยกุญแจสาธารณะของ CA X เพื่อใช้ตรวจสอบใบรับรองดิจิตอลใบที่ 1 แล้วจึงจะใช้กุญแจสาธารณะของ CA Y ที่ได้รับใบรับรองดิจิตอลของ CA Y ไปตรวจสอบใบรับรองดิจิตอลใบที่ 2 ซึ่งจะได้กุญแจสาธารณะของ Z แล้วนำไปตรวจสอบใบรับรองดิจิตอลใบที่ 3 และจะได้กุญแจสาธารณะของ Bob ในที่สุด การจัดการ CA อาจจะเป็นแบบ "general hierarchy" จากรูปวงกลมแทน CA สี่เหลี่ยมแทนผู้ใช้ (end user) ส่วนลูกศรแสดงกาออกใบรับรองดิจิตอลให้กันเป็นโหมดพ่อ (parent) และโหมดลูก (child) และยังมี cross certificate คือ certificate ที่ CA ตั้งแต่ 2CA ขึ้นไปร่วมกันออกหรือกล่าวได้ว่าได้รับการรับรองจาก CA ตั้งแต่ 2CA ขึ้นไปได้อีกด้วย ดังเช่น โหมด Q เป็นต้น นอกจากนี้อาจจะมีโครงสร้างเป็นแบบ "Top down hierarchy" CA ทำหน้าที่ออกใบรับรองดิจิตอลให้กับเฉพาะโหมดลูก (child mode) เท่านั้น โดยปกติแล้ว Certificate path length ไม่ควรจะยาวเกินไปเพื่อให้ง่ายต่อการค้นพบและการตรวจสอบ เพราะถ้ายาวเกินไปอาจก่อให้เกิดช่องโหว่ของการแอบอ้างและปลอมแปลงได้ อย่างไรก็ดีเราสามารถใช้ Cross Certificate เพื่อช่วยลด Certificate path length ได้

2.4.2.2 Validation

เนื่องจากข้อมูลในใบรับรองดิจิตอลสามารถเปลี่ยนได้ตลอดเวลา ดังนั้นผู้ที่จะใช้ใบรับรองดิจิตอลใดๆ จำเป็นที่จะต้องแน่ใจในความถูกต้องของข้อมูลในใบรับรองดิจิตอลนั้นๆก่อน นั่นคือ ผู้ใช้จำเป็นต้องมีการตรวจสอบใบรับรองดิจิตอลนั้นๆ เสียก่อน ซึ่งอาจแบ่งได้ 2 ลักษณะ คือ

- ติดต่อกับ CA โดยตรง เกี่ยวกับความถูกต้องของใบรับรองดิจิดอลทุกครั้งที่ใช้งาน เรียกว่า online validation

- CA จะกำหนดช่วงเวลาที่ใบรับรองดิจิตอลได้รับรองความถูกต้องในแต่ละใบรับรองดิจิตอลอยู่แล้ว เรียกว่า offline validation

นอกจากนี้ยังมีกระบวนการซึ่งเรียกว่า "Certificate revocation" ซึ่งเป็นกระบวนการที่บอกให้ผู้ที่ใช้งานรู้ถึงข้อมูลของใบรับรองดิจิดอลที่ถูกระงับใช้งานแล้ว ซึ่งสาเหตุอาจจะมาจาก ไพรเวทคีย์โดนขโมยหรือมีการเปลี่ยนใบรับรองดิจิตอลใหม่ เป็นต้น ซึ่งกระบวนการนี้มักใช้ร่วมกับ offline validation โดย CA จะทำการออกรายชื่อของใบรับรองดิจิตอลที่ถูกระงับการใช้งานก่อนเวลาอันควรซึ่งเรียกว่า CRLs (Certificate Revocation Lists)

แต่อย่างไรก็ตามการ CRLs ก็จะพบปัญหาได้ใน 2 กรณี คือ

1. ในด้านความเหลื่อมล้ำของเวลาที่ใบรับรองดิจิตอลถูกระงับการใช้งานโดย CA กับเวลาที่ CA ออก CRLs อันใหม่ออกมา เรียกว่า "CRLs time granularity problem"

2. ขนาดของ CRLs เนื่องจาก CRL จะเป็นรายชื่อของ ใบรับรองดิจิตอลทั้งหมดที่ถูกระงับการใช้งานซึ่งจะมีขนาดใหญ่ขึ้นเรื่อยๆ เมื่อเวลาผ่านไปทำให้เป็นการยากที่ผู้ใช้จะเอามาใช้ได้ วิธีการแก้ไขปัญหาดังกล่าวอาจทำได้การออก CRL แยกเป็นประเภทๆ ตามสาเหตุของการถูกระงับการใช้งานหรือแบ่งตามผู้ใช้งานว่าเป็น CA หรือผู้ใช้ ก็จะช่วยลดขนาดของ CRLs ได้ส่วนในเรื่องของความเหลื่อมล้ำของเวลาก็อาจทำให้การออก "delta-CRLs" ซึ่งเป็น Crl ที่แสดงรายชื่อของใบรับรองดิจิตอลที่ถูกระงับการใช้งานภายหลังการออก CRLs ครั้งล่าสุดไปและ สามารถออก delta-CRLs ได้บ่อยครั้งอีกด้วยเนื่องจากเป็นเพียงรายชื่อที่เปลี่ยนแปลงไปจาก CRLs ฉบับปัจจุบัน นอกจากนี้ delta-CRLs ยังมีส่วนในการช่วยลดขนาดของ CRLs อีกด้วย กล่าวคือ เริ่มออก full CRLs เพียงครั้งแรกครั้งเดียวแล้วก็มาทำหารออก delta-CRLs ให้บ่อยขึ้น จนถึงช่วงเวลาหนึ่งแล้วค่อยทำการปรับปรุง (update) full CRLs

2.4.3 การประยุกต์การใช้งาน PKI

- Secure E-mail
- Authentication

2.4.4 X.509

X.509 เป็นกรอบหรือรูปแบบของการตรวจสอบ (Authentication Framewaork) ที่ถูกออกแบบมาให้สนับสนุนโครงสร้าง X.500 โดยทั้ง X.509 และ X.500 เป็นส่วนหนึ่งของมาตรฐาน X-Series ซึ่งกำหนดโดย ISO และ ITU ซึ่งมาตรฐาน X.500 ถูกออกแบบมาสำหรับการให้บริการเครือข่ายขนาดใหญ่

2.4.4.1 X.500

โครงสร้างของ X.500 มีลักษณะคล้ายๆ กับสมุดโทรศัพท์ กล่าวคือ เมื่อได้ชื่อของคนๆหนึ่งมาก็จะสามารถหาข้อมูลของบุคคลนั้นได้ อย่างไรก็ตาม X.500 ให้ข้อมูลมากกว่า ชื่อ ที่อยู่และหมายเลขโทรศัพท์ ได้แก่ชื่อหน่วยงานที่คนๆนั้นทำงานอยู่ อาชีพ และอีเมล์แอดเดรส เป็นต้น นอกจากนี้ X.500 สามารถแสดงสิ่งต่างๆ ในโลกโดยไม่จำเป็นต้องเป็นตัวบุคคล เช่น คอมพิวเตอร์ และ บริษัท เป็นต้น

เพื่อสนับสนุนการค้นกาข้อมูล จึงได้มีการกำหนดชื่อที่มีความเป็นเอกลักษณ์ (Distinguished Name:DN) ซึ่งมีรูปแบบต่างๆกันไป ดังนั้น เพื่อยืนยันว่าชื่อที่ตั้งขึ้นมามีความเป็นเอกลักษณ์ X.500 จึงมีการจัดการในลักษณะ เป็นลำดับขั้นขึ้นมา ซึ่งเรียกว่า Directory Information Tree (DIT)

ในแต่ละโหนดยกเว้นรูท (Root) จะถูกกำหนดด้วย RDN (Relative Distinguished Name) ซึ่งจะมีความเป็นเอกลักษณ์ในระดับเดียวกันแล้ว DN จะเกิดการนำ RDN มาเชื่อมต่อกันโดยจะเริ่มตั้งแต่รูทไล่ลงมาตรมลำดับขั้นจนถึงโหนดที่เราสนใจ โหนดที่อยู่ในระดับถัดจากรูทลงมาจะเป็นชื่อของประเทศ ซึ่ง RDN ของแต่ละประเทศจะถูกกำหนดด้วยรหัสของอักษร 2 ตัวตามมาตรฐาน ISO เช่น CA = Canada,

TH=Thailand เป็นต้น ส่วนในลำดับต่อไปจะเป็นองค์กรภายในประเทศนั้นซึ่ง RDN จะเป็นชื่อองค์กรนั่นเองและลำดับถัดลงมาจะเป็นพนักงานในองค์นั้นๆ ซึ่ง RDN ก็คือชื่อของพนักงาน DN ของ Mr.Riel จะได้ เป็น

Country = CA Organization = Bombardier Inc.

CN = Louis Riel

2.4.4.2 X.509 V 3

X.509ถูกสร้างมาเพื่อสนับสนุนการพิสูจน์ตนโดยใช้ข้อมูลมาตรฐาน X.500 โดยมี X.509 V2 ซึ่งเป็นใบรับรองดิจิตอลตามรูปแบบมาตรฐานทางราชการ ซึ่งมีส่วนประกอบต่างๆ ดังรูปที่ 2-17 รูปที่ 2-17 แสดงว่าส่วนประกอบของใบรับรองดิจิตอลตามรูปแบบมาตรฐาน

- Version:เวอร์ชั่นของ X.509 ที่ใช้เป็นรูปแบบในการสร้างใบรับรองดิจิตอล

- Serial number: เลขที่ที่ความเป็นเอกลักษณ์ซึ่งกำหนดให้กับใบรับรองดิจิตอลที่ออกโดย CA

-CA Signature algorithm: เป็นอัลกอริทึมและพารามิเตอร์ต่างๆ ที่CA ใช้ในการลงลายมือชื่ออิเล็กโทรนิกส์

-Issue name: ชื่อตามรูปแบบของ X.500 ของ CA

-Validity period: ช่วงวันเวลาคู่หนึ่งซึ่งแสดงว่าใบรับรองดิจิตอลได้รับการรับรองความถูกต้อง

-Subject name: ชื่อตามรูปแบบของ X.500 ของผู้ที่ถือไพรเวทคีย์ซึ่งสอดคล้องกับ P กุญแจสาธารณะบนใบรับรองดิจิตอล

-Subject public key information: ค่าของกุญแจสาธารณะของเจ้าของใบรับรองดิจิตอลรวมไปถึงอักอริทึมที่ใช้ในการเข้ารหัสคีย์

-Issues unique identifier: ตัวอักษรจำนวนหนึ่งที่ช่วยทำตามรูปแบบของ X.500 ของ CA มากำกวม

-Subject unique identifier: ตัวอักษรจำนวนหนึ่งที่ช่วยทำให้ชื้อตามรูปแบบของ X.500 ของเจ้าของใบรับรองดิจิตอลนี้ไม่กำกวม

สำหรับมาตรฐาน X.509 V3 จะมีส่วนขยาย ดังนี้

-Certificate policies and policy mapping: กำหนดจุดประสงค์และนโยบายของการใช้งานใบรับรองดิจิตอล

-Alternative Name: ชื่อของ CA หรือเจ้าของใบรับรองดิจิตอลโดยไม่ต้องตรงตามรูปแบบของ X.500

-Subject directory attributes: สามารถเพิ่มข้อมูลอื่นๆ นอกจากนอกเหนือจากที่กำหนดใน X.500

-Certificate path constraints: CA สามารถแสดง Certificate path ขององค์กรได้โดยสามารถกำหนดเงื่อนไขและข้อบังคับของพาธ (path)

นอกจากนี้ใบรับรองดิจิตอลแล้ว X.509 ยังเป็นการกำหนดรูปแบบของ CRLs อีกด้วยสามารถแสดงได้และสำหรับในมาตรฐาน X.509 V3 จะมีส่วนขยายต่างๆ เพิ่มขึ้น ดังนี้

- CRL number and reason code: เพิ่มลำดับหมายเลขของใบรับรองดิจิตอลที่ถูกระงับการใช้งาน

- CRL distribution point: ทำการกระจายออก CRL ไปยังจุดต่างๆ แทนที่จะถูก full-CRL ที่จุดเดียวเพื่อลดขนาดของ CRL และภาระการโหลด CRL ของผู้ใช้

-Delta CRL: ใช้วิธีการ Delta CRL เพื่อลดความเหลื่อมล้ำระหว่างเวลาในการระงับการใช้งานใบรับรองดิจิตอลกับเวลาที่ออก CRL นอกจากนี้ยังช่วยลดขนาดของ CRL ด้วย ดังที่กล่าวไปเบื้องต้น

- Indirect CRL: การอนุญาตให้องค์กรอื่นๆ ที่มาได้เป็นผู้ออกใบรับรองดิจิตอลสามารถทำการออก CRL แล้วจึงทำการรวบรวม CRLs ทั้งหมดมาแล้วทำการกระจายที่จุดเดียว

2.5 ปัญหาความปลอดภัยที่เว็บไคลเอนต์

ปัญหาที่เกิดขึ้นที่ตัวเว็บไคลเอนต์ส่วนใหญ่เป็นปัญหาที่ส่งผลกระทบต่อข้อมูล ไฟล์และรีซอร์ส (Rosource) ต่างๆ ที่เว็บไว้ในฮาร์ดดิสก์ของผู้ใช้บริการ นอกจากนี้ยังรวมไปถึงการถูกละเมิดความเป็นส่วนตัวของผู้ใช้ในขณะทำการติดต่อกับเว็บไซต์อยู่ทั้งจากแฮกเกอร์และแม้กระทั้งเจ้าของเว็บไซต์เอง สามารถแบ่งพิจารณาได้ดังนี้

2.5.1 Client Side Script

ไคลเอนต์ไซต์สคลิปต์เป็นสคลิปต์ที่เขียนขึ้นด้วยโปรแกรมภาษาต่าง โดยสคลิปต์เหล่านี้จะถูกทำการประมวลผลที่ฝั่งไคลเอนต์ ซึ่งในที่นี้คือเว็บบราวเซอร์ โดยไคลเอนต์ไซต์สคลิปต์จะช่วยเพิ่มความสามารถในการติดต่อและโต้ตอบกันของข้อมูลหรือการทำงานต่างๆ ระหว่างผู้ใช้และเว็บเซิร์ฟเวอร์ผ่านสคริปต์ที่ทำงานโดยเว็บบราวเซอร์ เช่น สคลิปต์ที่นำมาใช้ในการตรวจสอบอินพุตของผู้ใช้ในการกรอกข้อมูลลงบนแบบฟอร์มผ่านเว็บบราวเซอร์ ซึ่งสคลิปต์ที่ทำงานนี้สามารถทำการตรวจสอบว่ามีความผิดพลาดเกิดขึ้นหรือไม่ก่อนที่จะส่งข้อมูลกลับไปยังเซิร์ฟเวอร์ ไคลเอนต์ไซต์สคลิปต์ ที่จะกล่าวถึงคือ วีบีสคลิปต์ (VB Script)

2.5.1.1 จาวาสคริปต์ (Java Script)

จาวาสคริปต์เป็นภาษายุคใหม่สำหรับการเขียนโปรแกรมบนระบบอินเตอร์เน็ตที่กำลังได้รับความนิยมอย่างสูง เราสามารถทำการเขียนโปรแกรมจาวาสคริปต์เพิ่มเข้าไปในเว็บเพจเพื่อใช้ประโยชน์สำหรับด้านต่างๆ ทั้งการคำนวณ การแสดงผล การรับส่งข้อมูล และที่สำคัญคือสามารถโต้ตอบกับผู้ใช้ได้อย่างทันทีทันใด นอกจากนี้ยังมีความสามารถด้านอื่นๆ อีกหลายประการที่ช่วยสร้างความน่าสนใจให้แก่เว็บเพจของเราเป็นอย่างมาก

จาวาสคริปต์ถือกำเนิดมาจากบริษัท Netscape Communication ถูกเปิดตัวครั้งแรกในชื่อ LiveScript เพื่อใช้สร้างเว็บเพจที่สามารถแลกเปลี่ยนข้อมูลกับเซิร์ฟเวอร์แบบ Liveware ได้โดยจาวา

สคริปต์เป็น "ภาษาคริปต์เชิงวัตถุ" ที่ช่วยให้เราสามารถควบคุมเว็บเพจได้อย่างง่ายดาย สามารถทำงานข้ามแพลตฟอร์มได้ ทำหน้าที่เป็นตัวประสานระหว่างเว็บเพจ HTML, จาวาแอพเพล็ต (Java Applet) และเว็บบราวเซอร์ทั้งทางฝั่งไคลเอนต์และฝั่งเซิร์ฟเวอร์ และสามารถใช้กับเทคโนโลยีอื่นๆ เช่น ActiveX, CGI, Plug-in, จาวาและอื่นๆ ซึ่งช่วยให้เว็บเพจที่บรรจุจาวาสคริปต์มีความน่าสนใจและสมบูรณ์แบบมากกว่าเว็บเพจทั่วไป

ตัวอย่างต่อไปนี้เป็นส่วนหนึ่งของโปรแกรมที่เขียนขึ้นจากจาวาสคริปต์และผลของการรันสคริปต์ ดังนี้

```
<Font Color="Maroom" Size="+1"> This is your first script example. Today's date is
        <SCRIPT LANGIAGE="JAVASCRIPT">
            <!
                d= new Date( );
                document.write( )
            // – >
        </SCRIPT>
</Font>
```

จากตัวอย่างเป็นโปรแกรมที่ถูกเขียนขึ้นด้วยจาวาสคริปต์ซึ่งสังเกตได้จากการประกาศการใช้ในคำสั่ง LANGUAGE โดยผลของการรันจาวาสคริปต์จะเป็นการนำเอาวันที่ที่ได้ไปเก็บไว้ในตัวแปล d ด้วยฟังชั่น new Date ซึ่งผลการรันเป็นดังนี้

```
Today's date is Tue Dec 13:20.22 UTC+700 2006
```

ตัวอย่างต่อไปนี้จะเป็นตัวอย่างตรวจสอบอินพุตซึ่งในที่นี้ก็คือความยาวของชื่อของผู้ใช้ (UserName) ว่ามีจำนวนตัวอักษรน้อยกว่า 5 ตัวหรือไม่และตรวจสอบว่าชื่อที่พิมพ์เข้าไปนั้นครงตามกฎหรือถูกต้องหรือไม่และในส่วนสุดท้ายก็จะเป็นการตรวจสอบรหัสผ่าน(Password)ดังนี้

```
<SCRIPT LANGIAGE="JAVASCRIPT"><! -
//perform client-side input valiiddation
//before returning to server
function thisPage_onbeforeserverevent( ){
        //check for minimun username length
        if(document.thisForm.Textbox1.value.length < 5){
                alert("UserName must br at least 5 characters!");
                thisPage.cancleEvent=true;
                return;}
        //check for valid username
        if(document.thisForm.Textbox1.value!="VIUSER"){
                alert("Invalid UserName!");
                thisPage.cancleEvent=true;
                return;}
        //check for valid password
        if(document.thisForm.Textbox2.value!="SECRET"){
                alert("Invalid Password");
                thisPage.cancleEvent=true;
                return;}}
//-->     </SCRIPT>
```

จากตัวอย่างโปรแกรมที่แสดงข้างต้นจะทำการรันเมทธอด(Method) thisPage_onbeforeserverevent และตรวจสอบอินพุตของผู้ใช้ในแบบฟอร์ม ซึ่งถ้าไม่ผ่านการตรวจสอบค่าของ thisPage.cancleEvent จะมีค่าเป็น true ซึ่งจะทำให้เว็บบราวเซอร์ยกเลิกการส่งข้อมูลไปยังเซร์ฟเวอร์ และทำการแจ้งต่อผู้ใช้เป็นต้น

2.5.1.1.1 ความปลอดภัยของจาวาสคริปต์ (Java Script Secuerity)

โปรแกรมที่เขียนขึ้น โดยจาวาสคริปต์จะมีความปลอดภัยมากกว่าโปรแกรมโปรแกรมที่เขียนขึ้นด้วยจาวาหรือโปรแกรมอื่นๆ เนื่องจากเหตุผลดังนี้

- ไม่มีเมทธอทใดๆ ของจาวาสคริปต์ ที่สามารถเข้าถึงระบบไฟล์ของเครื่องคอมพิวเตอร์ที่ฝั่งไคลเอนต์ได้โดยตรง
- ไม่มีเมทธอทใดๆ ของจาวาสคริปต์ที่จะสามารถทำการเปิดหรือสร้างการติดต่อสื่อสารไปยังเครื่องคอมพิวเตอร์บนเครือข่ายได้โดยตรง

แต่จาวาสคริปต์ก็เหมือนกับส่วนอื่นของเว็บ คือ ได้รับการพัฒนาและเปลี่ยนแปลงตลอดเวลา ซึ่งยิ่งพัฒนาก็จะมีจุดอ่อนเพิ่มขึ้นเรื่อยๆ โดยปัญหาทางด้านความปลอดภัยของจาวาสคริปต์สามารถ

จำแนกได้ 2 ประการ คือ ปัญหาความปลอดภัยเนื่องจากการจู่โจมแบบ Dos (Denial of Services Attack) และปัญหาความปลอดภัยจากการจู่โจมแบบสพูฟิ่ง (Spoofing Attack)

(1) ปัญหาความปลอดภัยเนื่องจากการจู่โจมแบบ Dos (Denial of Services)

ปัญหาความปลอดภัยที่สำคัญของจาวาสคริปต์ คือความยากในการป้องกันการจู่โจมแบบ Dos เนื่องมาจากผู้ที่ทำการพัฒนาจาวาสคริปต์ไม่ได้คำนึงถึงความสำคัญหรือข้อบกพร่องจากการจู่โจมในลักษณะนี้

การจู่โจมแบบ Dos จะส่งผลกระทบ ต่อผู้ที่ใช้เว็บด้วยเช่นกัน โดยโปรแกรมการจู่โจมจะถูกฝัง (Embedded) เป็นส่วนหนึ่งของโปรแกรมจาวาสคริปต์

การจู่โจมแบบ Dosจะมีลักษณะและผลของการจู่โจมที่แตกต่างกันออกไป ในส่วนนี้จะกล่าวถึงชนิดของการจู่โจมแบบ Dos ซึ่งจะเกี่ยวข้องกับเว็บบราวเซอร์ที่จะทำการรันสคริปต์ต่างๆ เหล่านั้นโดยจะกล่าวถึงเฉพาะ Internet Explorer ชนิดของการจู่โจมแบบ Dos มีดังนี้

- CPU and Stack Attack
- Swap Space Attack
- Window System Attack

(1.1) CPU and Stack Attack

โปรแกรมที่เขียนโดยจาวาสคริปต์สามารถที่จะทำให้เครื่องคอมพิวเตอร์หยุดการทำงานได้โดยโปรแกรมจะไปร้องขอการใช้งานของซีพียูและหน่วยความจำในปริมาณมากทำให้ซีพียูและหน่วยความจำไม่สามารถทำงานอื่นได้ตัวอย่างโปรแกรมต่อไปนี้เป็นโปรแกรมที่เขียนด้วยจาวาสคริปต์ซึ่งเป็นโปรแกรมที่ทำการคำนวณเลขทางคณิตศาสตร์ ซึ่งผลการทำงานจะใช้ทรัพยากรของซีพียูและหน่วยความทรงจำจนหมด

```
<html>
<head><title>Fibonacci Test Page</title>
</head>
<body>
<h1>The Fibonacci Series</h1>
<script>
function fibonacci (n)
        {
                if(n>1)return fibonacci(n-1)+fibonacci(n-2);
                        if(n=>0) return 0;
                        retun 1;
                }
                for(I=0;I<100000;I++){
                        document.write("Fibonacci number"+I+"is"+Fibonacci(I)+"<br>");
                }

                </script>
                </body>
                </html>
```

(1.2) Swap Space Attack

เป็นลักษณะของโปรแกรมที่เมื่อทำการรันแล้วพยายามที่จะจัดสรรหน่วยความจำจำนวนมาก เพื่อรองรับการทำงาน ซึ่งเสมือนกับการเคลื่อนย้ายไฟล์ (Swap) ไปยังหน่วยความจำสำรอง (Hard Disk) ทั้งหมดภายในเครื่อง

โปรแกรมต่อไปนี้เป็นโปรแกรมที่เป็นการจู่โจมแบบ Swap Space

```
Public true mouseDown(Event evt,int x, int y)
{
        String big="This is going to be really big.";
        Int I;
        For (I=0;I<100000;I++){
                big=big+big;
        }
        return true;
}
```

การจู่โจมแบบ Swap Space จะแตกต่างกับแบบ Stack Attack ซึ่งการจู่โจมในแบบ Swap Space

นี้จะส่งผลกระทบอย่างมากต่อประสิทธิภาพของเว็บบราวเซอร์และทุกๆ โปรแกรมที่กำลังทำงานอยู่บนเครื่องคอมพิวเตอร์ นั่นก็เป็นเพราะว่าคอมพิวเตอร์ถูกสั่งให้ทำการเคลื่อนย้ายไฟล์ ไปมาอยู่ตลอดเวลา ซึ่งในกระบวนการเคลื่อนย้ายไฟล์นี้จะทำให้หน่วยความจำสำรองและซีพียูไม่สามารถทำงานอื่นได้

(1.3) Window System Attack

จาวาสคริปต์ที่มีการดาวน์โหลดโค้ด (Code) ในการสร้างและจัดการหน้าต่าง (window) บนเครื่องผู้ใช้ การทำงานในส่วนของ GUI (Graphic User Interface) จะใช้ทรัพยากรของระบบบนเครื่องเป็นจำนวนมากโดยจะทำการสร้างหน้าต่างขึ้นมาจำนวนมาก ซึ่งทำให้ไม่สามารถทำงานอื่นได้ ตัวอย่างการจู่โจมชนิดนี้ เช่น การที่ผู้ใช้เข้าไปยังเว็บไซต์ที่ไม่ปลอดภัย เว็บบราวเซอร์จะทำการรันโปรแกรมในส่วนของจาวาสคริปต์ ซึ่งจะส่งผลให้เกิดการเปิดหน้าต่างใหม่ขึ้นเรื่อยๆ ทำให้ผู้ใช้ไม่สามรถหยุดโปรแกรมได้จนกว่าทรัพยากรบนเครื่องจะหมดไป สิ่งที่สามารถทำได้คือ การกด Ctrl+Alt+Delete ซึ่งจะส่งผลให้เครื่องต้องหยุดและเริ่มต้นการทำงานใหม่

(1.4) การป้องกันการจู่โจมแบบ Dos

ในความเป็นจริงแล้วไม่มีวิธีการใดในการป้องกันการจู่โจมทาง Dos ได้ 100 เปอร์เซ็นต์ แต่ผู้ใช้สามารถทำการป้องกัน โดยลดความเสี่ยงต่างๆ ที่เป็นสาเหตุการจู่โจมแบบ Dos โดยสาเหตุส่วนใหญ่ของการจู่โจมจะมาจากบั๊กของตัวโปรแกรมเอง ซึ่งแนวทางในการลดความเสี่ยงมีดังนี้

- จำกัดและติดตามการใช้ทรัพยากรภายในตัวเครื่องเนื่องจากการันโปรแกรมบนเครื่อง โดยทำการติดตามว่าโปรแกรมที่ทำการดาวน์โหลดหรือรันอยู่นั้นมีการใช้ซีพียูหรือหน่วยความจำมากน้อยแค่ไหน
- ทำการปรับค่าในบราวเซอร์ไม่ให้มีการใช้งานในส่วนของจาวาสคริปต์ (Disable) ของ Internet Explorer แต่ในกรณีที่ผู้ใช้ต้องตระหนักว่าจะไม่สามารถทำการรันงานบางอย่างหรือเปิดเว็บเพจได้ถ้าไม่มีการใช้งานจาวาสคริปต์

(2) ปัญหาความปลอดภัยจากการจู่โจมแบบสปูฟิง (Spoofing Attack)

โปรแกรมที่เขียนด้วยจาวาสคริปต์สามารถที่จะทำให้ผู้ใช้เครื่องสับสนได้เมื่อมีการเปิดเว็บเพจผ่านบราวเซอร์

Spoofing Browser Status with Java Script

โปรแกรมที่เขียนด้วยจาวาสคริปต์สามารถที่จะสร้างส่วนที่หลอกผู้ใช้บนหน้าเว็บเพจซึ่งสามารถที่จะสร้างส่วนที่หลอกผู้ใช้ได้ เช่น

- จาวาสคริปต์สามารถสร้างข้อความปลอมขึ้นมาหลอกผู้ใช้ได้
- จาวาสคริปต์สามารถทำการหลอกผู้ใช้โดยการเปลี่ยน URL ใน Status Line ได้
- จาวาสคริปต์สามรถทำการซ่อนส่วนของ "Goto:" ของเว็บบราวเซอร์ได้และทำงานด้วยแบบฟอร์มอื่นที่สร้างขึ้นเองจากจาวาสคริปต์

เช่น URL ที่ผู้ใช้ต้องการจะติดต่อ คือ http://www.shopping.com/order-entry.html แต่เมื่อผู้ใช้

ทำการคลิก เพื่อเข้าเว็บไซต์ เว็บเพจที่แสดงจริงกลับเป็น URL ที่ http://www.attacker.org/trapped.html โดยทำการเขียนจาวาสคริปต์ ดังนี้

```
<a href= "http://www.attacker.org/trapped.html"
onMouseerover="window.status='http://www.shopping.com/order/order-entry.html';
return true ">Click Here to enter your credit card number</a>
```

ผู้ใช้ส่วนใหญ่จะเชื่อถือและไว้ใจในการรันโปรแกรมที่ทำการดาวน์โหลดจากเว็บไซต์ที่น่าเชื่อถือ (Well Trusted Domain) อย่างไรก็ตามก็ยังมีอีกหลายกลยุทธ์ในการหลอกล่อผู้ใช้

(2.1) การป้องกันการโจมตีแบบสปูฟิง

ในการลดความเสี่ยงจากการจู่โจมแบบนี้โดยการป้องกันการสปูฟิง ซึ่งไม่สามรถเขียนด้วยจาวาสคริปต์ และวิธีการแก้ไขที่ดีที่สุดคือการตรวจสอบใบ Certificate ของทางฝั่งเซิร์เวอร์ที่ทำการติดต่อด้วย

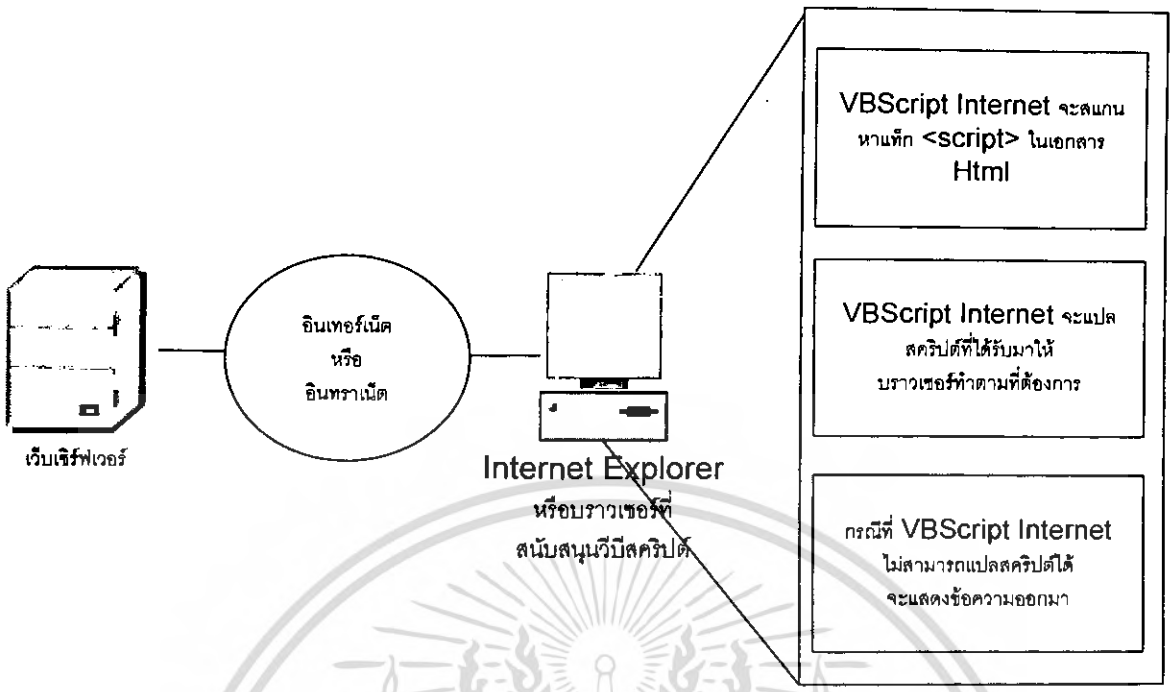
2.5.1.2 วีบีสคริปต์(VB Script)

วีบีสคริปต์ ถือได้ว่าเป็นสับเซ็ตของวิชวลเบสิก (Visual Basic) คือนำเอารูปแบบภาษาการเขียนโปรแกรมในแบบวิชวลเบสิกมาเขียนคำสั่งให้แอพพลิเคชันสำหรับอินเทอร์เน็ตหรือสั่งงานให้บราวเซอร์ทำงานได้ตามต้องการ โดยเพิ่มความน่าสนใจให้กับแอพพลิเคชั่นที่สร้างขึ้น

(1) องค์ประกอบของแอพพลิเคชั่นที่ใช้งานแอพพลิเคชั่น

แอพพลิเคชั่นที่นำความสามารถของแอพพลิเคชั่นไปใช้งานมักประกอบด้วย

- คำสั่งของภาษา HTML จะเป็นส่วนที่บรรจุข้อความในภาษา HTML ให้ทุกบราวเซอร์เข้าใจและแสดงผลได้อย่างตรงกัน
- VBScript Delimiter เป็นสิ่งที่ใช้แยกวีบีสคริปต์ออกจากภาษา HTML โดยจะใช้แท็ก <secret> ครอบส่วนที่เป็นคำสั่งในวีบีสคริปต์ และมักใช้แท็ก Comment (<!—กับ --!>) ครอบส่วนที่เป็นวีบีสคริปต์ภายในอีกชั้นหนึ่ง ซึ่งจะมีข้อดีคือ ถ้าแอพพลิเคชั่นนี้ถูกเรียกใช้งานโดยบราวเซอร์ที่ไม่สนับสนุนวีบีสคริปต์ก็ยังใช้งานได้อย่างต่อเนื่อง
- VBScript Subroutine or Function คือความสามารถในการสั่งโปรแกรมย่อยของวีบีสคริปต์ ซึ่งจะเหมือนกับการเขียนโปรแกรมในแบบโครงสร้าง (Structure Programming) ในกรณีของ Function เราสามารถเขียนขึ้นมาเองได้หรืออาจจะใช้งานฟังชั่นที่วีบีสคริปต์เตรียมไว้ก็ได้



รูปที่ 2.5 โครงสร้างการทำงานของแอพพลิเคชั่นที่ใช้วีบีสคริปต์จัดการ

- VBScript Built-in Object

วีบีสคริปต์เองมีออบเจ็กต์อยู่จำนวนหนึ่งซึ่งพร้อมถูกนำมาใช้งานร่วมกันกับคำสั่งในวีบีสคริปต์ เช่น Dictionary Object, File System Object, Error Object เป็นต้น

ตัวอย่างต่อไปนี้เป็นโปรแกรมอย่างง่ายที่เขียนขึ้นโดยวีบีสคริปต์และ HTML ซึ่งเป็นการสร้างปุ่มข้อความ (Massage Box) ที่มีข้อความว่า"Test" อยู่บนปุ่ม ดังนี้

```
<HTML>
        <HEAD><TITLE>A Simple First Page</TITLE>
        <SCRIPT LENGUAGE="VBScript">
        <!--
        Sub Button1_OnClick
                MsgBox"Test"
        End Sub
        -->
        </SCRIPT>
        </HEAD>
        <BODY>
        <H3>A Simple First Page</H3><HR>
        <FORM><INPUT NAME="Button1"TYPE="BUTTON"
        VALUE="Click Here"></FORM></BODY>
</HTML>
```

(2) วีบีสคริปต์กับการพัฒนาแอพพลิเคชั่นแบบ Event-Driven

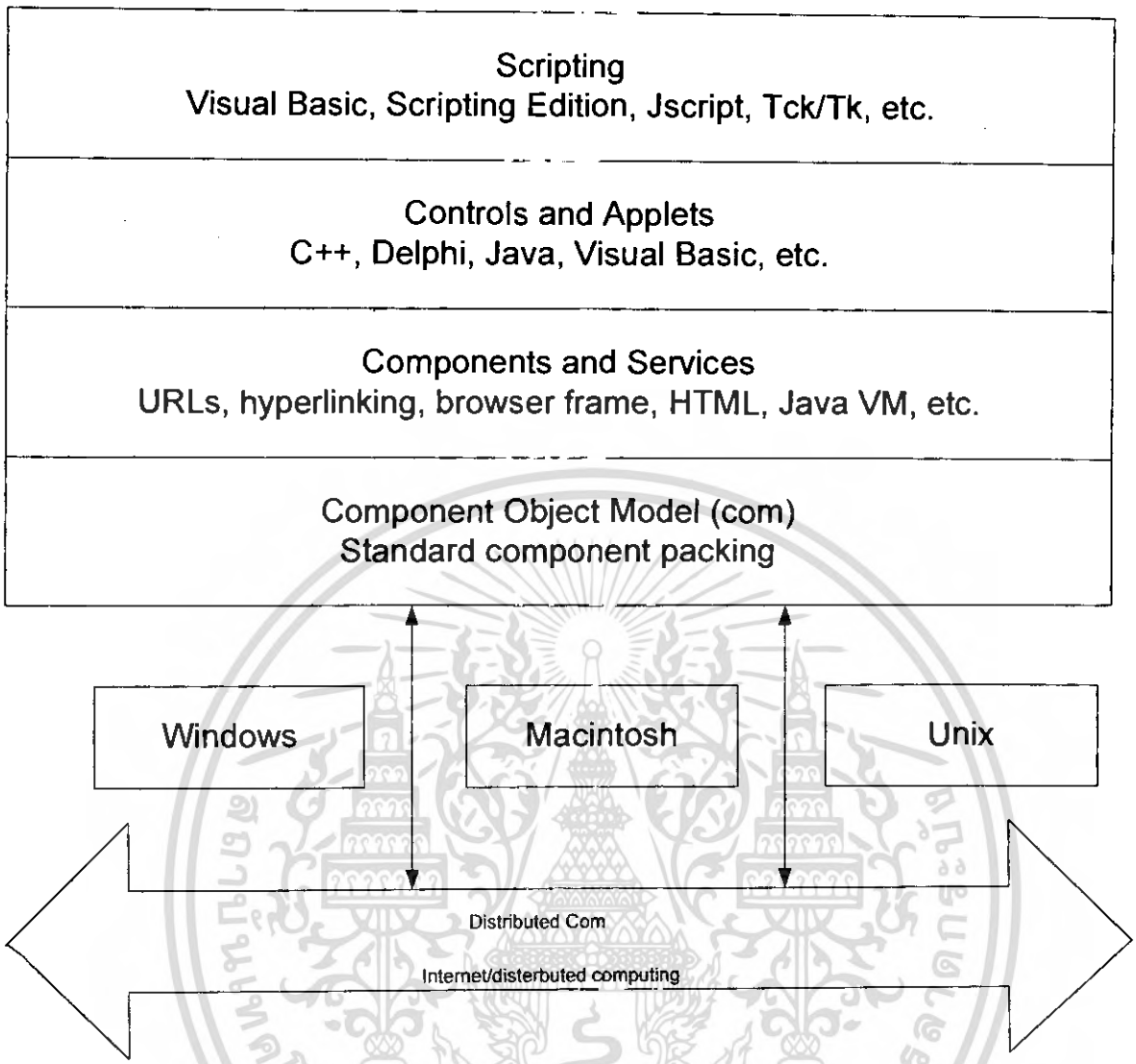
สำหรับวีบีสคริปต์แล้ว เรียกได้ว่าเป็นวิชวลเบสิกฉบับย่อซึ่งสิ่งหนึ่งที่ติดมาจากวิชวลเบสิกและทำการพัฒนาแอพพลิเคชั่นสำหรับอินเทอร์เน็ตมีประสิทธิภาพขึ้น ก็คือการใช้หลักของ Event-Driven (เหตุการณ์พาไป) มาใช้ทำให้แอพพลิเคชั่นอินเทอร์เน็ตที่สร้างขึ้นสามารถตอบสนองต่อเหตุการณ์ที่จะเกิดขึ้น ได้อย่างเหมาะสมโดยใช้รูปแบบภาษาเช่นเดียวกับวิชวลเบสิก

(3) ความปลอดภัยของวีบีสคริปต์(VBScript Security)

เนื่องจากวีบีสคิปต์มีการนำเอาเทคโนโลยีของ ActiveX มาใช้ในการทำงานร่วมกัน ทำให้การทำงานต่างๆ ของเว็บเพจที่เขียนด้วยวีบีสคริปต์มีความปลอดภัยเนื่องมาจากการใช้ ActiveX

- Component ที่รันบนฝั่งไคลเอนต์

ส่วนประกอบของเทคโนโลยีของ ActiveX แสดงได้ดังรูปที่ 2-4



รูปที่2.6 แสดงส่วนต่างๆของเทคโนโลยี ActiveX

2.5.1.3ActiveX

Active X เป็นคอมโพเนนต์สำหรับ Internet Explorer ช่วยทำให้เว็บเพจให้มีคุณภาพดี น่าใช้และช่วยให้ใช้ง่ายยิ่งขึ้น เช่น สามารถเพิ่มเมนูป๊อปอัพ(Pop-up menu) ทันทีที่ทำการคลิกเมาส์ซึ่งจะช่วยให้ผู้เลือกรายการที่ต้องการได้อย่างรวดเร็ว นอกจากนี้ยังสามารถใส่ ActiveX เพิ่มเข้าไปบังภาพเคลื่อนไหวหรือข้อมูลจากโปรแกรมอื่นๆ เช่น Microsoft Excel หรือ Microsoft Word ที่อยู่ในเว็บเพจได้

(1) Active Scripting

มีองค์ประกอบอยู่ 2 ชนิด คือ

1. ActiveX Scripting Hosts

ActiveX Scripting Hosts จะจัดหาแพลตฟอร์มไว้เพื่อให้ ActiveX Scripting Engines ทำงาน จะเห็นได้ว่า ActiveX Scripting Hosts หลักที่สำคัญคือ Microsoft Internet Explorer อย่างไรก็ตามยังมี scripting hosts อื่นที่มีศักยภาพภายใต้ ActiveX อีกเช่นกันได้แก่

- เว็บบราวเซอร์อื่นๆ
- Internet Authoring Tools
- เว็บเซิร์ฟเวอร์ (Server-base Scripting)

2. ActiveX Scripting Engines

โดยพื้นฐานแล้ว ActiveX scripting engine ก็คือภาษาที่สามารถปฏิบัติการบน ActiveX Scripting Hosts ได้ ActiveX Scripting Engine ชนิดแรกได้แก่ วีบีสคริปต์ (เป็นส่วนหนึ่งของ Visual Basic) อย่างไรก็ตามยังมี environment อื่นๆ อีกเช่น Perl, Lisp, Delphi, Scheme

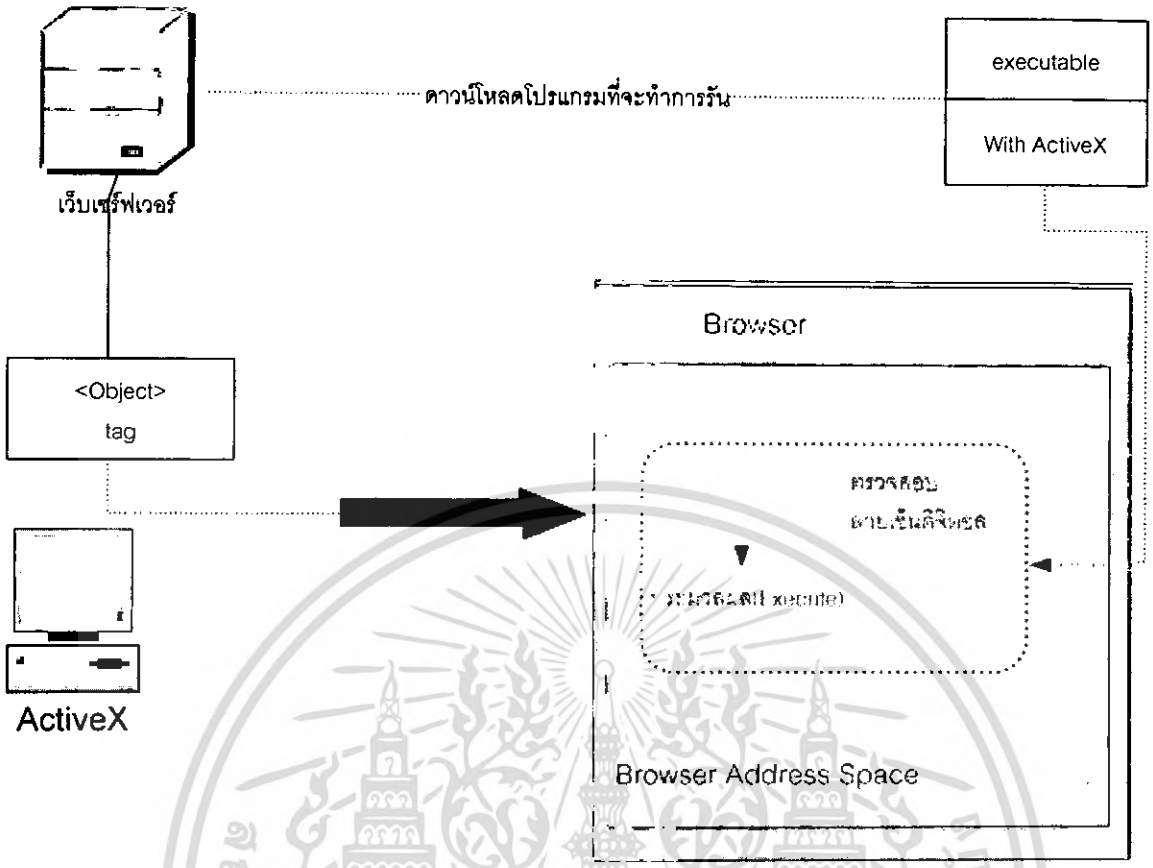
(2) ActiveX Documents

Active Documents เป็นชุดเครื่องมือ (tools) ที่ใช้ในการสร้าง dynamic content ซึ่งจะสนับสนุนการสร้าง ActiveX Control ได้ดีพอๆ กับ developer ชนิดอื่นๆ เช่น Visual Basic 5.0 และ ActiveX Document เป็น software Object ที่ถูกดาวน์โหลดและทำงานภายใน ActiveX Container อย่างเช่น Internet Explorer ได้ ดังนั้น ActiveX Documents จึงนับว่าเป็นเครื่องทุ่นแรงนักพัฒนาได้อย่างสากในการสร้างแอพพลิเคชั่นทางอินเตอร์เน็ต โดย ActiveX Documents จะกำหนดส่วนที่ใช้ในการเข้าถึง (access) เมทธอท ต่างๆ ที่สร้างขึ้นภายใน developer ไว้ในรูปแบบที่สามารถทำการดาวน์โหลดได้ ผลก็คือทำให้ผู้ใช้สามารถดูและแก้ไขเอกสารที่ไม่ใช่ HTML ผ่านทางบราวเซอร์ได้

(3) ActiveX Control

ActiveX Control ได้ถูกแนะนำเพื่อการรวมสองส่วนที่แยกกันวิวัฒนาการของเทคโนโลยีคอมพิวเตอร์ Custom Control กับส่วนที่คิดใหม่ของไอเดียพื้นฐานเกี่ยวกับ OLE และ OOP

คอนโทรลตัวใหม่นี้เป็นคอนโทรล OLE ที่อยู่บนพื้นฐานของ DCOM (Distributed Component Object Model) ตัวแรก ดังที่ได้กล่าวมาแล้วว่าการติดต่อพื้นฐานของ OLE อยู่บนพื้นฐานของ COM เพราะฉะนั้น ActiveX คือ คอนโทรล OLE อย่างแท้จริงตัวแรก



รูปที่ 2.7 ส่วนของ ActiveX Control ที่ถูกดาวน์โหลดและทำการรันที่เว็บบราวเซอร์

(4) ความปลอดภัยของ ActiveX(ActiveX Security)

ActiveX สามารถนำเอาแอพพลิเคชันและคอนโทรลต่างๆ ไปวางไว้บนเว็บเพจได้ และเมื่อต้องการที่จะใช้ก้ต้องทำการดาวน์โหลดและจะต้องลงทะเบียนกับระบบปฏิบัติการ ซึ่งสามารถถูกใช้โดยแอพพลิเคชันอื่น ที่อยู่บนระบบได้ด้วย นอกเหนือจากใช้บราวเซอร์ กระบวนการนี้เรียกว่า การดาวน์โหลดคอมโพเนนต์(Component Downloading) แต่ยังมีข้อจำกัดบางประการและคำถามสำคัญที่มีอยู่ซึ่งต้องถามเพื่อป้องกันในแต่ละการดาวน์โหลดคอนโทรล ActiveX มายังเครื่อง

- คอนโทรล ActiveX บางตัวไมสามารถนำมาใช้หรือเผยแพร่ได้ถ้าไม่มีใบอนุญาตจากนักพัฒนาคอนโทรลนั้นๆ
- จะมีการเซ็นรหัส (Code Signing) และมีการใช้เทคโนโลยีในการตรวจสอบบุคคล (Authentication Technology)มาใช้ในการยืนยันการสร้างคอนโทรลว่ามีความปลอดภัยและมาจากแหล่งที่น่าเชื่อถือ ซึ่งจะช่วยในการป้องกันคอนโทรลที่อาจบรรจุไวรัสไว้ได้

ปัญหาอีกอย่างคือยากในการที่จะทำการควบคุมและติดตามการทำงานต่างๆ ในส่วนของคอนโทรลที่มีการทำงานอย่างสลับซับซ้อน ตัวอย่างเช่น การส่งข้อมูลข่าวสารที่เป็นความลับที่เกี่ยวกับการติดตั้ง (Configuration information) จากคอมพิวเตอร์ของผู้ใช้ไปยังเซิร์ฟเวอร์ผ่านอินเทอร์เน็ต ซึ่งอาจจะทำให้มีไวรัสเกิดภายในวง LAN เป็นต้น

นอกจากนี้ที่ตัว ActiveX Control ยังได้รับรายงานถึงปัญหาด้านความปลอดภัยที่เกิดขึ้น จากการทำงานของตัวคอนโทรลสองตัว คือ scriptlet.typelib และ eyedog ซึ่งผลเสียโดยรวมที่เกิดขึ้นคือเจ้าของเว็บไซต์สามารถที่จะทำการเขียนเว็บเพ็จที่จะสามารถทำการบางอย่างกับเครื่องคอมพิวเตอร์จของผู้ที่เรียกใช้เว็บเพจนั้น ซึ่งตัวคอนโทรลทั้งสองถูกออกแบบมาเพื่อใช้ทำงานดังนี้

- Scriptlet.typelib ถูกออกแบบมาเพื่อให้นักพัฒนาโปรแกรมใช้สร้าง Type Libraies สำหรับ Window Script Component(WSC)ซึ่ง Type Libraries เหล่านี้ถูกใช้โดยเครื่องมือที่ใช้ในการพัฒนาทั้งหลาย เช่น Microsoft Visual InterDev เพื่อสร้างคุณสมบัติบางอย่าง เช่น ตัวช่วยเหลือในการทำงาน (Tool-tip help)
- Eyedog นั้นถูกออกแบบมาเพื่อให้โปรแกรมที่ใช้ในการวิเคราะห์ระบบใช้ในการรวบรวมข้อมูลเกี่ยวกับฮาร์ดแวร์ทั้งหลายบนเครื่องที่โปรแกรมเหล่านั้นกำลังทำงานอยู่

ปัญหาก็คือตัวคอนโทรลทั้งสองถูกกำหนดไว้ว่าปลอดภัยในภาษาสคริปต์(Save for Scripting) นั้นคือได้รับการรับรองจากทางไมโครซอฟท์แล้วว่าปลอดภัยแต่การทำงานของเครื่องคอมพิวเตอร์ของผู้ใช้ ดังนั้นจึงสามารถทำงานโดยไม่ต้องร้องขอความยินยอมจากผู้ใช้ก่อน ซึ่งทำให้เกิดอันตรายได้ดังนี้

- Scriptlet.typelib ทำให้เว็บเพจสามารถแก้ไขหรือลบไฟล์บนเครื่องคอมพิวเตอร์ของผู้ใช้ได้ โดยแก้ไขไฟล์ของระบบ โดยเจ้าของเว็บไซต์สามารถสั่งคำสั่งหรือโปรแกรมใดที่เข้าต้องการให้ทำงานขึ้นมาก็ได้
- Eyedog ทำให้เว็บเพจสามารถเก็บข้อมูลต่างๆ จากเครื่องคอมพิวเตอร์ของผู้ใช้ได้ เช่น ค่าในรีจิสตรี การกำหนดค่าต่างๆ ของระบบ และทำกาส่งข้อมูลนี้กลับไปยังเซิร์ฟเวอร์

การป้องกันปัญหาทางด้านความปลอดภัยของ ActiveX

ส่วนในกรณีของปัญหาที่เกิดจาคอลโทรลเลอร์ต่างๆนั้นสามารถแก้โดยจะการดาวน์โหลดแพตซ์มาอัพเดท

- สำหรับ Scriptlet.typelib นั้นแพตซ์จะทำการยกเลิกความปลอดภัยสำหรับภาษาสคลิป์ ทำให้ต้องได้รับการอนุญาตจากผู้ใช้ก่อนถึงจะทำงานได้
- สำหรับ Eyedog นั้นแพตซ์จะทำการตั้งค่าบิตที่ชื่อว่า Kill Bit ซึ่งจะทำให้ไม่สามารถใช้ใน Internet Explorer ได้อีกต่อไป

2.5.1.4 จาวาแอพเพล็ต(Java Applet)

แอพเพล็ต (Applet) คือโปรแกรมขนาดเล็กที่สร้างขึ้นด้วยภาษาจาวา สามารถถูกเรียกจากใจ HTML เพจให้ทำงานเป็นส่วนหนึ่งของเว็บเพ็จนั้น

การทำงานแลความปลอดภัยของจาวาแอพเพล็ต(Java Applet Security)

เมื่อผู้ใช้ส่งคำร้องขอข้อมูลเว็บเพจไปยังเว็บเซิร์ฟเวอร์ หากภายในเว็บเพจนั้นมีคำสั่ง <APPLET>...</APPLET> สำหรับกำหนดจาวาอแพล็ตที่ต้องการนำมาใช้งาน เซิร์ฟเวอร์ก็จะเพียงทำหน้าที่ส่งข้อมูลต่างๆ เกี่ยวกับแอพเพล็ตนั้นไปยังเครื่องคอมพิวเตอร์ของผู้ใช้ จากนั้นเว็บบราวเซอร์ของ

ผู้ใช้จะนำไบต์โค๊ดที่ได้นั้นไปเรียกใช้งานต่อไป

เนื่องจากจาวาถูกสร้างขึ้นมาเพื่อใช้งานร่วมกันกับระบบเครือข่าย จึงถูกเน้นให้มีระบบรักษาความปลอดภัยที่รัดกุมเพื่อป้องกันอันตรายที่อาจเกิดขึ้น ภาษาจาวาจึงไม่มีคุณสมบัติในการเข้าถึงหน่อยความจำของระบบในระดับลึกเพื่อดัดปัญหามิให้ภาษาจาวาเป็นอันตรายต่อระบบการทำงานของเครื่องคอมพิวเตอร์ที่เรียกใช้มัน และจาวาทำงานผ่านเว็บบราวเซอร์ ดังนั้นเว็บบราวเซอร์จึงทำหน้าที่เป็นผู้ตรวจทานรหัสของคำสั่งก่อนว่า ไม่มีคำสั่งที่เป็นอันตรายต่อระบบ จากนั้นจึงผ่านให้ Java Class Loader เพื่อสั่งให้โปรแกรมทำงานต่อไป ฉะนั้นจาวาจึงเหมาะกับการใช้งานที่ต้องการความปลอดภัยสูงผ่านระบบเน็ตเวิร์กหรือทำธุรกิจในระบบอินเตอร์เน็ต

2.5.1.4.1ข้อจำกัดของแอพเพล็ต(Applet Restrictions)

เพื่อป้องกันไม่ให้ผู้ประสงค์ร้ายสามารถสร้างแอพเพล็ตที่ไปทำลายโปรแกรมหรือข้อมูลบนเครื่องคอมพิวเตอร์ของผู้ที่รับแอ็พเพล็ตไปใช้งาน จึงมีข้อจำกัดบ้างประการไว้ โดยบราวเซอร์จะถือว่าแอพเพล็ตที่รับมาจากเครื่องอื่นในระบบเครือข่ายเป็นโปรแกรมที่ไม่ปลอดภัยไว้ก่อน ละจะให้ทำงานในสภาพแวดล้อมที่ปลอดภัย ซึ่งจะมีการติดตั้ง Security Manager ให้คอยตรวจสอบการทำงานของแอพเพล็ตและไม่ยอมให้มีการทำงานที่ไม่สมควรเกิดขึ้น ด้วยกฎเกณฑ์ตังต่อไปนี้

1. ห้ามออกคำสั่งกับระบบไฟล์ของเครื่องที่รับแอ็พเพล็ตนั้นไปทำงาน โดยมีรายละเอียดดังนี้
- ห้ามเขียน อ่าน หรือลบไฟล์
- ห้ามสร้างไดเรกทอรีหรือดูว่าในไดเรกทอรี่มีอะไรบ้าง
- ห้ามตรวจสอบว่ามีไฟล์หรือไดรเรกทอรีชื่อหนึ่งในเครื่องนั้นหรือไม่
- ห้ามตรวจสอบว่าชื่อหนึ่งเป็นไฟล์หรือไดเรกทอรี
- ห้ามขอข้อมูล ขนาด หรือเวลาที่ทำการเปลี่ยนแปลงไฟล์นั้น
- ห้ามเปลี่ยนชื่อไฟล์หรือเปลี่ยนแปลงค่า file descriptor

2. ห้ามออกคำสั่งแก่ระบบเครือข่ายดังนี้
- ห้ามติดต่อไปยังคอมพิวเตอร์เครื่องอื่นที่ไม่ใช่เครื่องที่แอพเพล็ตนั้นมา
- ห้าม listen หรือ accept ใน port ใดๆ ที่มีเลขน้อยกว่าหรือเท่ากับ 1024
- ห้ามใช้ multicast sockets
- ห้ามสร้าง instance ของคลาส SocketImplFactory, URL StreamHandlerFactory,หรือ ContentHandlerFactory

3. ห้ามไม่ให้ใช้ฟังก์ชั่นบางประเภทของระบบ เช่น
- ห้ามออกจากตัวแปลจาวา (Java Interpreter) ด้วยคำสั่ง System.exit() หรือ Runtime.exit()
- ห้ามสร้างและทำงานโพรเซสใหม่ด้วยคำสั่ง Runtime.exec()
- ห้ามทำการโหลด native code ด้วยคำสั่ง load() หรือ loadLibrany() ของคลาส Runtime หรือคลาส System

4. ไม่สามารถใช้งานคลาสบางตัวของ AWT event queue

- ทุกๆ หน้าต่าง (Window) ที่ถูกสร้างขึ้นด้วยแอพเพล็ตจะมีข้อความให้เห็นว่าไม่ปลอดภัย
- ห้ามสร้าง print job
- ห้ามอ้างถึง System Clipboard หรือ AWT event queue

5. ห้ามอ้างถึงค่าคุณสมบัติ (Properties) ของระบบ เช่น ห้ามเรียกคำสั่ง System.getProperty() และสามารถแก้ไขหรือเพิ่มค่าคุณสมบัติใน System Properties List แต่แอพเพล็ตสามารถเรียกคำสั่ง System.getProperty() อ่านค่าคุณสมบัติแต่ละตัวได้ โดยมีข้อแม้ว่าต้องได้รับอนุญาตจาก appletviewer โดยทั่วไป appletviewer อนุญาตให้สามารถอ่านค่าคุณสมบัติ (Properties) ต่างๆ เหล่านี้ได้ java.version, java.class,vertion, java.vendor, java.vendor.url, os.namc, os.version, os.arch, file.seperator, path.separator, line.separator
6. ห้ามสร้างหรืออ้างถึงเธรด(Thread) หรือกลุ่มของเธรด(Thread Group) ที่ไม่ได้อยู่ในกลุ่มของเธรดของแอพเพล็ตนั้น
7. มีข้อจำกัดในการโหลดและกำหนดคลาส ดังนี้

- ห้ามโหลดคลาสจาก sun.*packages
- ห้ามกำหนดคลาสใหม่ให้อยู่ภายใต้ java.* หรือ sun.*package.
- ห้ามสร้าง Class Loader หรือเรียกคำสั่งใดๆ ใน Class Loader

8. ห้ามใช้คลาส Java.lang.Class เพื่อทำการ reflection ในการขอข้อมูลของสมาชิกของคลาสที่ไม่ใช่ public ยกเว้นแต่คลาสนั้นถูกโหลดมาจากเซิร์ฟเวอร์เดียวกันกับที่ให้แอพเพล็ตนั้นมา
9. มีข้อจำในการใช้ java.security package

- ห้ามจัดการกับ Security Identities ไม่ว่ากรณีใดๆ
- ห้ามอ่านหรือเขียนค่า Security Properties
- ห้าม list, lookup, insert หรือ remove security provides
- ห้ามสร้าง instance ของคลาส ClassLoader หรือ Security Mangerขึ้นมาใหม่

ในกรณีใช้ appletviewer ทำงาน แอพเพล็ตที่อยู่ในเครื่องเดียวกัน ข้อจำกัดบางอย่างข้างต้นอาจถูกละเว้น แต่แอพเพล็ตที่ถูกโหลดจะยังคงผ่านกระบวนการรักษาความปลอดภัยที่ appletview นั้นสร้างขึ้น

2.5.2 ปัญหาด้านความปลอดภัยของเว็บบราวเซอร์

เป็นที่ทราบกันแล้วว่าเว็บบราวเซอร์นั้น จากที่กล่าวมาแล้วว่าหน้าที่ของเว็บบราวเซอร์ก็คือการแปลเอกสาร HTML แล้วแสดงออกมาเป็นหน้าเว็บเพจรวมไปถึงความสามารถในการรันโปรแกรมบางอย่างที่ฝั่งไคลเอนต์เพื่อเพิ่มประสิทธิภาพให้กับเว็บเพจ ซึ่งการทำงานดังกล่าวอาจทำให้เกิดการเสี่ยงของปัญหาและการละเมิดความเป็นส่วนตัว ซึ่งอาจจะเป็นบั๊กหรือข้อบกพร่องที่เกิดจากการออกแบบเว็บ

บราวเซอร์เอง ไม่ได้เกิดจากโปรแกรมที่ถูกรันโดยบราวเซอร์เอง ซึ่งเราสามารถแบ่งสาเหตุของปัญหาได้ดังนี้

2.5.2.1 ปัญหาที่เกิดจากการทำงานของคุกกี้(Cookies)

คุกกี้เป็นเท็กซ์ไฟล์ขนาดเล็กที่อาศัยอยู่ในฮาร์ดไดร์ฟ ซึ่งจะเก็บข้อมูลบางอย่างเกี่ยวกับผู้ใช้ ที่ได้เข้าไปเยี่ยมชมหรือทำธุรกรรมทั้งที่ผ่านมาและเว็บไซต์ปัจจุบันที่ผู้ใช้กำลังเล่นอยู่ เพื่อลดการทำงานบางอย่างที่เกี่ยวข้องกับบราวเซอร์ แต่ในอีกด้านหนึ่งก็บอกถึงร่องรอยการใช้งานอินเตอร์เน็ต การทำธุรกรรมต่างๆของผู้ใช้

2.5.2.2 ปัญหาความปลอดภัยและผลกระทบจากสคริปต์ จาวา และActiveX

ทั้งจาวาและ ActiveX เป็นโปรแกรม (Applets) ซึ่งสามารถทำการดาวน์โหลดและรันบนเครื่องของคุณผ่านทางบราวเซอร์ที่คุณใช้ ซึ่งเมื่อเปรียบเทียบระหว่างจาวาและ ActiveX แล้ว ActiveX สามารถสร้างอันตรายและส่งผลเสียมากกว่า นั้นคือ ActiveX สามารถส่งผลต่อระบบ(System Calls) โดยทำให้ไฟล์บนไดร์ฟของคุณมีปัญหาได้

ส่วนจาวาแอ็พเพล็ดจะทำงานภายใต้ข้อจำกัดของบราวเซอร์ การจำกัดด้านความปลอดภัย ที่ป้องกันบราวเซอร์ของคุณจากแอ็พเพล็ต ดังนี้

1. จะป้องกันการเข้าถึงหรือการเปลี่ยนแปลงไฟล์โดยตรงในเครื่องของคุณ
2. จะป้องกันการเข้าถึงอุปกรณ์ต่างๆ ในเครื่องของคุณ
3. จะป้องกันการสร้างเครือข่ายการติดต่อกับบุคคลอื่นๆ ที่ไม่ใช่เว็บไซต์ที่คุณกำลังติดต่อด้วย

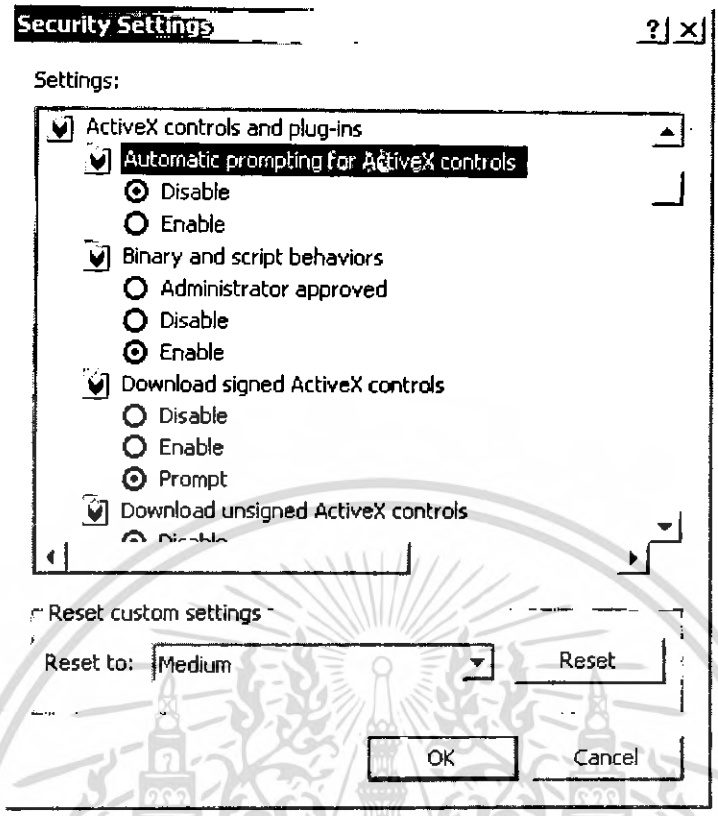
ทั้งนี้ยังอาจมีอันตรายต่อบราวเซอร์ของคุณ ขึ้นอยู่กับการออกแบบและข้อจำกัดต่างๆ ของบราวเซอร์ที่ได้พัฒนาขึ้นมา

2.5.2.3 Internet Explorer

-จากเมนู Tools ทำการเลือก Internet Options

-เลือกที่แท็บ Security

-ทำการเลือกระดับความปลอดภัย (Security Level) ซึ่งคุณสามารถเลือกระดับความปลอดภัยทั้ง Low, High และ medium โดยเลือกที่แท็บ “Custom Level” ดังรูป



รูป 2.8 การเซ็ตค่าความปลอดภัยของคุกกี้และจาวาแอพเพล็ต

ใน "Custom Level":

High จะทำการป้องกันทุกอย่างไม่ว่าจะเป็นคุกกี้ ActiveX และแอพเพล็ตแต่บางเว็บไซต์จะไม่สามารถทำงานได้ถ้าปราศจากจาวาสริปต์หรือคุกกี้

Medium จะเป็นการเซ็ตแบบ Default นั่นคือบราวเซอร์จะทำงานตามปกติและจะถามในกรณีของ ActiveX หรือเนื้อความต่างๆ ที่ไม่มีความปลอดภัย

Low จะอนุญาตให้การงานทุกอย่าง ทุกรูปแบบเกิดขึ้นได้ โดยไม่มีการรัษาความปลอดภัย

จากตัวอย่างข้างล่างเป็นการเซ็ตระดับการรักษาความปลอดภัยในระดับ "High" โดยจะยกเลิกการทำงานทุกๆอย่าง ยกเว้น ActiveX ที่มีความหน้าเชื่อถือ (Trusted) และใช้ untrusted สำหรับจาวาแอพเพล็ต ดังรูป

2.5.2.4 ปัญหาที่เกิดจากผู้ใช้เว็บบราวเซอร์

ในกรณีปัญหาที่เกิดจากตัวของผู้ใช้เว็บบราวเซอร์จะเป็นปัญหากรณีเฉพาะ หมายความว่าปัญหาจะเกิดขึ้นหรือไม่นั่นอยู่ที่ตัวผู้ใช้ และในกรณีที่เกิดปัญหาใดๆขึ้น การที่จะแก้ไขปัญหาก็จะอยู่ที่ตัวของผู้ใช้เองด้วยเช่นกัน ดังนั้นในการที่จะป้องกันไม่ให้เกิดปัญหาผู้ใช้จะต้องมีความตระหนัก (awareness)รัดกุม(concieness)และมีสติ(Consciousness)ในการใช้เว็บบราวเซอร์เพื่อทำธุรกิจทางอินเตอร์เน็ต

2.6 ปัญหาความปลอดภัยที่เว็บเซิร์ฟเวอร์

ปัญหาที่เกิดขึ้นจะมาในหลายรูปแบบซึ่งโดยส่วนใหญ่มีจุดประสงค์เพื่อทำให้เว็บไซต์นั้นถูกเปลี่ยนแปลงหรือทำให้เซร์ฟเวอร์ไม่สามารถทำการได้ตามปกติ สามารถแบ่งย่อยๆ ได้ดังนี้

2.6.1 ความอ่อนแอของรหัสผ่าน (Password) ที่จะใช้ในการแก้ไขเว็บไซต์

การเข้าไปแก้ไขเว็บเพจโดยการแฮกรหัสผ่าน โดยวิธีการที่ง่ายที่สุดในการเจาะผ่านเข้าระบบอินเตอร์เน็ตก็คือการเดารหัสผ่านหรือพาสเวิร์ด ซึ่งในการเข้าไปแก้ไขเว็บเพจจะต้องพิมพ์ชื่อ(User Name) และรหัสผ่านจึงจะเข้าไปในระบบได้ ดังนั้นถ้าหากว่าผู้ใช้ใช้ชื่อไม่ดีพอ เช่น Jack และใช้รหัสผ่านว่า Jack123 ก็อาจจะมีคนสามารถเดาได้ และก็มีโปรแกรมประเภท Brute Force ซึ่งจะทำการเดารหัสผ่านไปเรื่อยจนกว่าจะเจอ

การหารหัสผ่านอีกวิธีที่นิยมกันมากคอการดูที่ไฟล์ /etc/password ซึ่งจะอยู่ในระบบยูนิกซ์รุ่นเก่า

พวกแฮกเกอร์สามารถใช้โปรแกรมค้นหา เช่น Crack เพื่อค้นหารหัสผ่านจากไฟล์นี้ได้ แต่ในระบบยูนิกซ์รุ่นใหม่ๆ จะใช้รหัสผ่านเงา (Shadow password)ซึ่งทำให้การค้นหารหัสผ่านทำได้ยากขึ้น

ตัวอย่างของรายชื่อใน /etc/password

Root : x : 0 : 1 : Sys. Admin : / : / bin / sh

โดยตำแหน่งที่ตัว x จะเป็นพาสเวิร์ดที่มีการเข้ารหัสไว้และพาสเวิร์ดที่เข้ารหัสไว้จะถูกเก็บไดว้ในไฟล์ชื่อ /etc/shadow ซึ่งแต่ละแถวในนั้นจะมีรูปดังนี้

Root : XyfgFekJ95Fpq

อีกวิธีคือการใช้ฟังชั่นที่ชื่อว่า pwdauth() ซึ่งจะรับชื่อและรหัสผ่านเข้าไปในตัวฟังก์ชันจากนั้นมันจะเข้ารหัสเจ้าตัวพาสเวิร์ดแล้วไปเปรียบเทียบกับรหัสที่เก็บไว้ในรหัสผ่านเงา ซึ่งถ้าตรงกันก็จะทำให้แฮกเกอร์รู้ว่านี้คือรหัสผ่านที่ใช้ได้

เมื่อแฮกเกอร์สามารถทำการขโมยรหัสผ่านของผู้ใช้หรือเจ้าของเว็บไซต์ได้ ส่วนใหญ่จะเข้าไปทำการแก้ไขหน้าเว็บเพจเพื่อเป็นการกลั่นแกล้งหรือเพื่อเป็นการข้ำเตือนถึงความอ่อนแอของระบบนั้น

2.6.2 การโจมตีเพื่อการปิดให้บริการ(Denial of Service:Dos) และ (Distributes Denial of service:DDos)

เป็นการทำให้เว็บเซิร์ฟเวอร์ไม่สามารถให้บริการบางอย่างได้หรือไม่สามารถให้บริการต่อไปได้อีก โดยทั่วไปจะโจมตีที่พอร์ตทีซีพี/ไอพี จึงเป็นการโจมตีการให้บริการของระบบนั้นเอง และอาจมีผลให้ระบบนั้นไม่สามารถให้บริการบางอย่างได้ หรือไม่สามารถให้บริการใดๆ ได้เลย แบ่งได้เป็นหัวข้อได้ดังนี้

2.6.2.1 การส่งแพ็กเก็ตจำนวนมาก (Amount of Packers Sending)

การโจมตีแบบนี้เป็นการส่งแพ็กเก็ตปริมาณมากเข้าไปยังเว็บเซิร์ฟเวอร์อาจทำให้เว็บเซิร์ฟเวอร์ไม่สามารถให้บริการบางอย่างหรือไม่สามารถทำงานต่อได้ ลักษณะของแพ็กเก็ตได้แก่

- แพ็กเก็ตข้อมูล (Data Packets)
- แพ็กเก็ตสำหรับการควบคุม (Control Packets)

ตัวอย่างการโจมตีแบบนี้ ได้แก่ การทำ SYN Flooding ปกติการเชื่อมต่อแบบ 3-way handshake มีช่วงโหว่ คือเครื่องที่ขอบริการส่งสัญญาณ SYS ไป แต่เมื่อได้รับสัญญาณ ACK จากเครื่องที่ให้บริการแล้วไม่ส่งสัญญาณ SYN ตอบกลับไป ทำให้เครื่องที่ให้บริการต้องเปิดการเชื่อมต่อรอการตอบกลับ ซึ่งการเปิดการเชื่อมต่อรอเอาไว้นี้ต้องใช้ทรัพยากรของระบบส่วนหนึ่ง และหากมีการส่งสัญญาณในลักษณะนี้มากๆ และทรัพยากรของระบบมีไม่เพียงพอ อาจทำให้ระบบไม่สามารถทำให้บริการอย่างอื่นๆ หรือให้บริการกับผู้รองขอรายอื่นได้

2.6.2.2 ความผิดปกติของแฟร็กเมนต์ (Abnormal Fragmentation)

การโจมตีนี้อาศัยหลักการแฟร็กเมนต์เตชันและรีแอสเซมเบิ้ล โดยการทำให้แพ็กเก็ตนั้นมีการรีแอสเซมเบิ้ล ซึ่งปกติการรีแอสเซมเบิ้ลทั้งหมดนั้นต้องสามารถเชื่อมต่อกันได้สนิท ดังรูป แต่แพ็กเก็ตที่ผู้บุกรุกส่งไปมีการแก้ไขข้อมูลในบางฟิลด์ ทำให้เกิดความผิดปกติในกระบวนการรีแอสเซมเบิ้ล ซึ่งการโจมตีในลักษณะนี้ แบ่งได้ดังต่อไปนี้

- **การส่งแพ็กเก็ตที่มีลำดับผิดปกติ(Abnormal Sequence of Packets Sending)**

ปกติการส่งแพ็กเก็ตมักเรียงตามลำดับกันไป หากไม่เรียงลำดับก็ต้องรอจนกว่าแพ็กแก็ตก่อนหน้านี้มาถึงเพื่อเรียงลำดับแพ็กเก็ตที่เครื่องรับ แต่การโจมตีแบบนี้กลับส่งเฉพาะแพ็กเก็ตสุดท้ายเพื่อให้ระบุเป้าหมายรอแพ็กเก็ตก่อนหน้า และส่งไปเป็นปริมาณมากๆ เพื่อให้ระบบเป้าหมายไม่สามารถให้บริการอย่างอื่นได้

โดยปกติแล้วการโจมตีในรูปแบบนี้ผู้โจมตีจะแก้ไขข้อมูลในฟิลด์แสดงลำดับของแพ็กเก็ต (Fragment Offset) ของแพ็กเก็ตไอ้พี ซึ่งเป็นส่วนที่แสดงลำดับของข้อมูลหลังจากกระบวนการแฟร็กเมนต์เตชันโดยแก้ให้ส่งแพ็กเก็ตสุดท้ายหรือแพ็กเก็ตหลังๆ เพียงแพ็กเก็ตเดียวเลย ทำให้ระบบเป้าหมายต้องรอแพ็กเก็ตก่อนหน้านี้

- **การส่งแพ็กเก็ตแบบวนลูป (Looping)**

คือ การส่งโดยกำหนดค่าแอดเดรสต้นทาง(Source Address) และแอดเดรสปลายทาง (Destination Address) ให้เหมือนกัน ทำให้เกิดการรับวนส่งไปมาอยู่ที่เครื่องเป้าหมาย ซึ่งเป็นโปรแกรมโจมตีที่มีการกำหนดแอดเดรสต้นทางและแอดเดรสปลายทางเป็นค่าเดียวกัน คือ เป็นแอดเดรสของเครื่องเป้าหมายนั้นเองทำให้เกิดการส่งวนไปมาอยู่ที่เครื่องเป้าหมาย

2.7 ปัญหาความปลอดภัยสำหรับการสื่อสารระหว่างเว็บไคลเอนต์และเซิร์ฟเวอร์

ปัญหาที่เกิดขึ้น โดยส่วนใหญ่ผู้มุ่งร้ายจะมุ้งเน้นความสนใจไปยังข้อมูลที่ทำการสื่อสารกันระหว่างเว็บไคลเอนต์และเซิร์ฟเวอร์ ทั้งในรูปแบบของการคัดเอาข้อมูลไปใช้ประโยชน์ การเปลี่ยนแปลง

ข้อมูลและการแอบอ้างตัวบุคคล ทั้งในรูปแบบของการตักเอาข้อมูลไปใช้ประโยชน์ การเปลี่ยนแปลงข้อมูลและการแอบอ้างตัวบุคคล โดยเฉพาะประการหลังจะทำให้เกิดปัญหาการไม่ไว้ใจกันระหว่างเว็บไคลเอนต์และเซิร์ฟเวอร์ ซึ่งจะส่งผลไปถึงการลดอัตราการใช้บริการในเชิงพาณิชย์ สามารถแยกพิจารณาเป็นกรณีได้ดังนี้

2.7.1 ปัญหาในการแอบอ้าง

ปัญหาการแอบอ้างที่สำคัญ ได้แก่ เว็บสปูฟิง (Web Spoofing) คือ การจำลองเว็บที่มีลักษณะเหมือนเว็บต้นแบบทุกประการ (shadow copy) เพื่อให้ผู้ใช้ติดต่อเข้ามาโดยไม่รู้ว่าเป็นเว็บที่ถูกจำลองขึ้นมา โดยมีวัตถุประสงค์เพื่อตรวจจับ แก้ไข เปลี่ยนแปลงข้อมูลต่างๆ ที่สื่อสารกันระหว่างเว็บไคลเอนต์และเซิร์ฟเวอร์จริงที่ผู้ใช้ต้องการติดต่อด้วย

การสร้างเว็บที่ถูกจำลองขึ้นมาอาจจะหลงเหลือหลักฐานต่างๆ ไว้ได้ เช่นที่แถบแสดงสถานะ (Status Bar) ที่แถบแสดงที่อยู่(Location Bar) แสดงที่อยู่ของเว็บไซต์ที่เราจะติดต่อด้วยไม่ถูกต้อง เป็นต้น แต่หลักฐานเหล่านี้สามารถปกปิดได้โดยการใช้จาวาสคลิปต์(Javascript) สิ่งสำคัญที่จะทำให้เว็บสปูฟฟิงเกิดผลจริง ก็คือ ทำอย่างไรให้มีผู้เข้ามายังเว็บนี้ให้ได้ โดยวิธีส่วนใหญ่ที่ใช่กัน เช่น

- การฝากลิงค์ที่จะมายังเว็บที่ถูกจำลองขึ้นมาในเว็บที่มีชื่อเสียง
- ส่งไปในรูปของเมลล์ โดยเป็นลักษณะของเมลล์ที่มีลิงค์ไปยังหน้าเว็บได้

เทคนิคที่แฮกเกอร์นิยมใช้ในการทำเว็บสปูฟฟิง คือ การทำ DNS Spoofing ซึ่งเป็นการอาศัยจุดอ่อนของโพรโตคอล TCP/IP โดยจะเป็นการจู่โจมไปที่เนมเซิร์ฟเวอร์

- จู่โจมให้เนมเซิร์ฟเวอร์ไม่สามารถทำงานได้ชั่วขณะ
- ตอบข้อมูล IP Address ที่ผิดกลับไปก่อนที่เนมเซิร์ฟเวอร์จริงจะตอบ
- โดยสิ่งจำเป็นต้องทราบก็คือ หมายเลขของแพ็กเก็ตที่จะตอบรับคำร้องขอของเว็บไคลเอ็นต์ โดยใช้วิธีการ Sniffing

นอกจากนี้ยังมีปัญหาที่เกิดกับเนมเซิร์ฟเวอร์ โดยเรียกว่า เว็บไฮแจ็กกิ้ง (Web Hijacking) เป็นการแฮกไปที่ Routing Table แล้วแก้ไขไฟล์คอนฟิก (Config file) ที่ map ระหว่างโดเมนเนมกับ IP Address เช่น

www.sanook.com 161.246.10.21 ถูกแก้เป็น www.sanook.com 161.246.34.11

ดังนั้นเมื่อผู้ใช้ทำการเรียกใช้เว็บดังกล่าวผ่านทางเนมเซิร์ฟเวอร์ตัวนี้ ก็จะโดนลิงก์ไปยัง www.kmitl.ac.th แทน

2.7.2 ปัญหาข้อมูลที่ทำการรับส่งถูกลักลอบไป (Data Trapping)

ข้อมูลที่ทำการสื่อสารระหว่างเว็บไคลเอนต์และเซิร์ฟเวอร์จะถูกลักลอบ เช่น การทำการสั่งสินค้าผ่านทางเว็บ E-Commerce ซึ่งผู้ใช้จะต้องทำการใส่หมายเลขบัตรเครดิต หรือ พาสเวิร์ดซึ่งในกรณีที่ข้อมูลเหล่านี้ถูกลักลอบไปจะเกิดความเสียต่อเจ้าของเป็นอย่างมาก

แฮกเกอร์สามารถทำการดักจับข้อมูลโดยใช้โปรแกรมในการดักจับข้อมูล เช่น Sniffer ซึ่งเป็น

โปรแกรมที่ใช้ในการดักจับข้อมูลที่ถูกส่งมาในรูปของแพ็กเก็ตที่ไหลผ่านในเครื่องคอมพิวเตอร์ตัวที่ติดตั้งโปรแกรมเอาไว้ ไม่ว่าข้อมูลจะถูกระบุไอพีแอดเดรสปลายทางเป็นอะไรก็ตาม ตัวโปรแกรม Sniffer จะสามารถถอดข้อมูลที่อยู่ในแพ็กเก็ตนั้นมาอ่านได้ แม้ว่าข้อมูลนั้นไม่ได้ส่งถึงมันก็ตาม

2.7.3 ปัญหาข้อมูลที่ทำการรับส่งถูกเปลี่ยนแปลง (Data Altering)

ประเภทของความเสี่ยงต่อระบบความปลอดภัยอีกแบบหนึ่งคือ แฮกเกอร์ไม่ได้ทำการดักเอาข้อมูลไปแต่เป็นการเปลี่ยนแปลงเป็นข้อมูลตัวใหม่ลงไปแทนซึ่งข้อมูลตัวใหม่ถูกเข้ารหัสเหมือนกันและยังถูกต้องตามวิธีกาทุกอย่าง เช่นเดียวกับข้อมูลเดิมที่ได้ทำไว้แต่รายละเอียดของข้อมูลเปลี่ยนไป ซึ่งรูปแบบการเปลี่ยนแปลงข้อมูลนี้จะทำให้ปลายทางได้รับข้อมูลที่ผิดทำให้การทำงานผิดพลาดและอาจให้บริการกับผู้ใช้ (User) ผิดคนและทำให้การสื่อสารผิดพลาด

2.8 การเข้ารหัส

เกิดจากหลักคณิตศาสตร์ที่เรียกว่า ฟังก์ชันทางเดียว (one-way function) กลุ่มของฟังก์ชันทางเดียวส่วนหนึ่งมีความเกี่ยวข้องกับเลขจำนวนเฉพาะ (prime number) เลขที่หารได้เฉพาะ 1 และตัวมันเอง ถ้าเอาเลขจำนวนเฉพาะสองตัวมาคูณกัน สมมติเป็น 5 กับ 7 ได้ 35 .. ทีนี้ลองหาตัวประกอบของ 35 ดูสิครับ เราจะได้ว่ามีเพียง 5 และ 7 เท่านั้นที่เป็นตัวประกอบ (ไม่นับ 1x35 นะครับ) ทีนี้ลองหาอีกซักจำนวนนึง 11,927 x 20,903 คำตอบคือ 249,310,081 ..การคูณ 11,927 กับ 20,903 นี้ง่ายกว่าหาตัวประกอบของ 249,310,081 ใช่มั้ยครับ ? ยิ่งเลขจำนวณเฉพาะมีค่ามากเท่าไหร่ยิ่งจะแยกตัวประกอบยากขึ้นเท่านั้น ทีนี้นึกออกยังครับว่ามันเกี่ยวอะไรกับ public key ? ..ใบ้อีกหน่อยสมมติว่า 249,310,081 เป็นข้อมูลที่เราได้รับและถ้าเรารู้จำนวนเฉพาะตัวนึง เราจะหาอีกตัวนึงได้อย่างง่ายดาย แต่ถ้าเราไม่รู้ล่ะก็กว่าจะหาได้ก็นานทีเดียว..ใช่แล้ว คุณสมบัตินี้สามารถเอามาใช้เก็บความลับได้ !! แนวคิดเรื่อง public key ไม่ได้เป็นของใหม่ครับ คนแรกที่คิดเรื่องนี้คือ วิธฟิลด์ ดิฟฟี่ (Whitfield Diffie) และ มาร์ติน เฮลแมน (Martin Hellman) ซึ่งเสนอวิธีการแบบ public key นี้ใน National Computer Conference ปี 1976 และตีพิมพ์ใน IEEE Transaction on Information Theory หลังจากนั้นไม่กี่เดือนต่อมา algorithm ที่ทั้งสองคิดขึ้นมาอธิบายได้เป็น

1. A และ B กำหนดค่า n และ g โดยที่ $1 < g < n$ ..เลขทั้งสองไม่จำเป็นต้องเป็นความลับ
2. A สุ่มเลขที่มีค่ามากๆ มาตัวนึง กำหนดให้เป็นค่า x และหาค่า $X = g^x \bmod n$ เก็บค่า x เป็นความลับ
3. B ทำเหมือนกัน สุ่มเลขที่มีค่ามากๆ มาตัวนึง กำหนดให้เป็นค่า y และหาค่า $Y = g^y \bmod n$ เก็บค่า y เป็นความลับ
4. ทีนี้ A กับ B แลกค่า X และ Y กัน
5. A คำนวณหาค่า $k = Y^x \bmod n$
6. B คำนวณหาค่า $k' = X^y \bmod n$

7. ค่า k และ k' จะมีค่าเท่ากัน และเท่ากับ $g^{xy} \bmod n$

ค่า k และ k' ที่ว่านี้นอกจาก A และ B แล้วคนอื่นไม่มีทางหาได้เพราะค่าที่คนอื่นมีโอกาสรู้มีเพียง n, g, X และ Y โอกาสที่จะหาค่า x จาก X (หรือ y จาก Y) ทำได้ด้วยการหา inverse ของ X ซึ่งเรียกว่า discrete logarithm

2.8.1 RSA

RSA cryptosystem คิดค้นโดย รอน ริเวสต์ (Ron Rivest), อาดิ ชาร์เมียร์ (Adi Shamir), และ เลียวนาร์ด เอเดิลแมน (Leonard Adleman) ในปี 1978 ความปลอดภัยของ algorithm นี้ขึ้นกับความยากในการแยกตัวประกอบของเลขจำนวนเฉพาะที่มีค่ามากๆ ทั้งสามคิดว่าวิธีการนี้ปลอดภัยมากและเชื่อว่าต้องใช้เวลานับล้านปีกว่าจะแยกตัวประกอบของเลขจำนวน 129 หลักออกไม่ว่าจะใช้คอมพิวเตอร์ที่ทรงพลังขนาดไหนก็ตาม ปัญหาการแยกตัวประกอบ 129 ตัวนี้เป็นที่รู้จักกันดีในวงการนักคณิตศาสตร์และคอมพิวเตอร์ว่า "RSA 129" เลขที่ว่านี้คือ

"114 381 625 757 888 867 669 235 779 967 146 612 010 218 296 721 242 362 562 561 842 935 706 935 245 733 897 830 597 123 563 958 705 058 989 075 147 599 290 026 879 543 541"

เลขนี้ตีพิมพ์เป็นปริศนาในคอลัมน์ Mathematical Games ใน Scientific American โดย มาร์ติน การ์ดเนอร์ ในปี 1977 (ก่อน RSA จะตีพิมพ์) ไม่มีใครในยุคนั้นสามารถหาคำตอบได้เลยจนกระทั่งในปี 1993 จึงมีคนพยายามแก้ปริศนานี้อีกครั้ง พอล เลย์แลนด์ (Paul Leyland), ไมเคิล กราฟฟ์ (Michael Graff) และ เดเรค แอตกินส์ (Derek Atkins) เป็นผู้ที่พยายามจะแก้ปัญหานี้โดยได้รับการสนับสนุนจากอาสาสมัครมากกว่า 600 คนทั่วโลกให้ run โปรแกรมที่เขียนโดย เค. เลนสตรา (K. Lenstra) ในเวลากลางคืนเพื่อช่วยกันหาคำตอบผ่านทางอินเทอร์เน็ตในที่สุดในเดือนเมษายนปี 1994 ปริศนาก็ถูกแก้ออกเป็นเลขจำนวนเฉพาะขนาด 64 และ 65 หลักคือ

"3 490 529 510 847 650 949 147 849 619 903 898 133 417 764 638 493 387 843 990 820 577"

"32 769 132 993 266 709 549 961 988 190 834 461 413 177 642 967 992 942 539 798 288 533"

และยังถอดรหัสออกมาเป็นข้อความได้ว่า "The magic words are Squeamish and Ossifrage" อย่างไรก็ตามปัจจุบัน RSA ใช้กุญแจเป็นตัวเลขขนาด 1024 bits (ประมาณ 309 หลัก) เป็นอย่างน้อย จึงยอมรับกันว่า RSA cryptosystem ปลอดภัยสูงพอ

RSA Cryptosystem
Public Key:

- n คำนวณจากเลขจำนวนเฉพาะสองตัว p และ q คูณกัน - ทั้ง p และ q ต้องเก็บเป็นความลับ ปกติจะทำลายทิ้งหลังจากหา key ได้เพราะไม่ได้ใช้ในการเข้า/ถอดรหัส
- e เป็นจำนวนที่ไม่มีตัวประกอบร่วมกับ (p-1)(q-1)

Private Key: $d = e^{-1}(\text{mod}(p-1)(q-1))$
Encrypting: $c = m^e(\text{mod } n)$
Decrypting: $m = c^d(\text{mod } n)$

2.9 การเข้ารหัส RSA

2.8.2 MD5

Rivest เป็นผู้พัฒนาเช่นกันโดยพัฒนาต่อจาก MD4 เพื่อให้มีความปลอดภัยที่สูงขึ้น ถึงแม้จะเป็นที่นิยมใช้งานกันอย่างแพร่หลาย ทว่าในปี 1996 ก็มีผู้พบจุดบกพร่องของ MD5 (เช่นเดียวกับ MD4 (ปัญหาการชนกันของไดเจสต์มีโอกาสเกิดขึ้นได้ไม่น้อย ซึ่งผู้บุกรุกอาจใช้ประโยชน์จากจุดอ่อนนี้เพื่อทำการแก้ไขข้อความตั้งต้นที่ส่งมาให้ได้)) จึงทำให้ความนิยมเริ่มลดลง MD5 ผลิตไดเจสต์ที่มีขนาด 128 บิต

2.8.3 DES

DES ย่อมาจาก Data Encryption Standard อัลกอริทึมนี้ได้รับการรับรองโดยรัฐบาลสหรัฐอเมริกาในปี ค.ศ. 1977 ให้เป็นมาตรฐานการเข้ารหัสข้อมูลสำหรับหน่วยงานของรัฐทั้งหมด ในปี 1981 อัลกอริทึมยังได้รับการกำหนดให้เป็นมาตรฐานการเข้ารหัสข้อมูลในระดับนานาชาติตามมาตรฐาน ANSI (American National Standards) DES เป็นอัลกอริทึมแบบบล็อกซึ่งใช้กุญแจที่มีขนาดความยาว 56 บิตและเป็นอัลกอริทึมที่มีความแข็งแกร่ง แต่เนื่องด้วยขนาดความยาวของกุญแจที่มีขนาดเพียง 56 บิต ซึ่งในปัจจุบันถือได้ว่าสั้นเกินไป ผู้บุกรุกอาจใช้วิธีการลองผิดลองถูกเพื่อค้นหากุญแจที่ถูกต้องสำหรับการถอดรหัสได้ ในปี 1998 ได้มีการสร้างเครื่องคอมพิวเตอร์พิเศษขึ้นมาซึ่งมีมูลค่า 250,000 เหรียญสหรัฐเพื่อใช้ในการค้นหากุญแจที่ถูกต้องของการเข้ารหัสข้อมูลหนึ่งๆ ด้วย DES และพบว่าเครื่องคอมพิวเตอร์นี้สามารถค้นหากุญแจที่ถูกต้องได้ภายในระยะเวลาไม่ถึงหนึ่งวัน

2.8.3 .1 Triple-DES

Triple-DES เป็นอัลกอริทึมที่เสริมความปลอดภัยของ DES ให้มีความแข็งแกร่งมากขึ้นโดยใช้อัลกอริทึม DES เป็นจำนวนสามครั้งเพื่อทำการเข้ารหัส แต่ละครั้งจะใช้กุญแจในการเข้ารหัสที่แตกต่างกัน ดังนั้นจึงเปรียบเสมือนการใช้กุญแจเข้ารหัสที่มีความยาวเท่ากับ 56*3 = 168 บิต Triple-DES ได้ถูก ใช้งานกับสถาบันทางการเงินอย่างแพร่หลาย รวมทั้งใช้งานกับโปรแกรม Secure Shell (ssh) ด้วย การใช้อัลกอริทึม DES เพื่อเข้ารหัสเป็นจำนวนสองครั้งด้วยกุญแจสองตัว (56*2=112 บิต) ยังถือได้ว่าไม่ปลอดภัยอย่างพอเพียง การเข้ารหัสโดยคีย์เดี่ยว 3DES แบบ mode CBC

ECB คือการเข้ารหัสทีละบล็อก โดยไม่มีความสัมพันธ์ระหว่างบล็อกเลย

ถ้าเขียนเป็นสูตรก็

C1 = E(Key, M1)

C2 = E(Key, M2)

ถ้าสลับ M1 กับ M2 เราก็จะได้ C1 กับ C2 เหมือนเดิม ก็เลยมีการสร้าง Relative key (Initialization vector) ขึ้นมา เพื่อใส่ผสมเข้าไปในแต่ละบล็อกเพื่อสร้างความปั่นป่วนใน cipher ให้อ่านยาก ๆ ลบร่องรอย (finger print) ของการทำ block cipher ให้หายไป ทำให้เรามองไม่เห็นขนาดของ block ไม่รู้ขนาด key และลบร่องรอยของ encryption algorithm เองด้วย

Cipher = Encrypt(Message, IV)

C1 = E(Key, M1, IV)

C2 = E(Key, M2, IV + 1)

รอบนี้ถ้าสลับ M1 กับ M2 เราจะไม่ได้ C1 กับ C2 แต่จะได้ Cxx มาแทน

2.8.4 AES

อัลกอริทึมนี้ได้รับการพัฒนาโดย Joan Daemen และ Vincent Rijmen ในปี 2000 อัลกอริทึมได้รับการคัดเลือกโดยหน่วยงาน National Institute of Standard and Technology (NIST) ของสหรัฐอเมริกาให้เป็นมาตรฐานในการเข้ารหัสชั้นสูงของประเทศ อัลกอริทึมมีความเร็วสูงและมีขนาดกะทัดรัดโดยสามารถใช้กุญแจที่มีความยาวขนาด 128, 192 และ 256 บิต

บทที่ 3
การวิเคราะห์และการออกแบบระบบ

ระบบเป็นโปรแกรมจำลองการทำงานของเว็บไซต์ เพื่อให้บริการจำหน่ายซอฟต์แวร์ โดยการทำงานผ่านทาง Command Line ซึ่งถูกสร้างขึ้นโดยใช้ภาษา Java เพื่อให้ง่ายต่อการนำไปประยุกต์ใช้และพัฒนาต่อเพื่อสร้างเว็บไซต์ผ่านทางภาษา JSP (Java Server Pages) ต่อไป

3.1. การออกแบบระบบ

3.1.1 ส่วนที่ใช้ติดต่อกันระหว่างลูกค้าและร้านค้า

3.1.1 การที่ลูกค้าจะเข้าไปใช้บริการดาวน์โหลดซอฟต์แวร์ ได้นั้น จะต้องทำการจ่ายเงินเสียก่อน จากนั้นทางร้านค้าจะทำการกำหนด Username/Password และ Certificate ของลูกค้าแต่ละคนขึ้นมาและทำการส่งข้อมูลเหล่านั้นกับไปให้ลูกค้า โดย Username/Password จะถูกเก็บอยู่ในระบบฐานข้อมูลของทางร้านค้า

- ระบบฐานข้อมูลจะถูกเก็บอยู่ในรูปของ Text file โดย Password จะถูกเข้ารหัสโดยอัลกอริทึม RSA
- การเชื่อมต่อระหว่างลูกค้าและร้านค้า จะถูกรักษาความปลอดภัยโดย SSL โปรโตคอล โดยกุญแจสาธารณะ (Public Key) จาก certificate จะถูกนำมาใช้ในการเข้ารหัสในการส่งสื่อสารและการพิสูจน์ตัวตนกันระหว่างระหว่างทั้งสองฝ่าย
- ในระบบนี้ไม่ได้ทำการออกแบบกระบวนการตรวจสอบการจ่ายเงินของลูกค้า แต่จะสมมุติให้ลูกค้าแต่ละคนทำการจ่ายเงินเรียบร้อยแล้ว

3.1.2 เมื่อลูกค้าได้รับ Username/Password และ Certificate แล้วถึงจะสามารถ Login เข้าสู่ระบบเพื่อทำการดาวน์โหลดซอฟต์แวร์ ได้เมื่อลูกค้าทำการดาวน์โหลดซอฟต์แวร์ เรียบร้อยแล้ว ซอฟต์แวร์ที่ถูกดาวน์โหลดมา จะอยู่ถูกผนึกอยู่ในรูปของ Jar/Zip files ซึ่งจะถูกกำกับไว้ด้วยกุญแจเข้ารหัสจากทางร้านค้า และทางร้านค้าก็จะทำการส่งกุญแจเข้ารหัส SHA-Digest ของซอฟต์แวร์ตัวนั้นๆมาให้กับลูกค้า เพื่อทำการเปรียบเทียบกับกุญแจเข้ารหัส SHA-Digest ที่มาพร้อมกับซอฟต์แวร์ ถ้ากุญแจทั้งสองตรงกันก็แสดงว่าซอฟต์แวร์ ตัวนั้นมีลิขสิทธ์ถูกต้องตามกฎหมายและสามารถนำไปใช้ได้ แต่ถ้าไม่ต้องตรงกัน ก็จะไม่สามารถนำซอฟต์แวร์ ตัวนั้นไปใช้ได้

- กระบวนการนี้ได้ประยุกต์ การเข้ารหัส, Certificate, Jar/Zip files, SSL, Authentication
- SHA-Digest จะถูกส่งผ่านทาง Cipher Stream โดยใช้กุญแจสาธารณะ (Public Key) จาก Certificate ของลูกค้าเพื่อเป็น session key และ secure random number ในการสร้าง Cipher Stream

3.1.2 ส่วนของร้านค้า

- สามารถสร้าง Username/Password สำหรับลูกค้า

- สามารถสร้าง Keystore, Shop's Key, Customer's Key

- สามารถสร้าง Certificate และสามารถนำ Certificate ที่ออกขึ้นมาไปเก็บไว้ใน Shop's Key, Customer's Key ได้

- สามารถสร้างและเปิด Jar file และสามารถกำกับ Jar file ด้วย Key จาก Keystore

3.1.3 รายชื่อไฟล์ของระบบ

-CallBackHandler.java

-LoginModule.java

-auth.conf

-Cert.java

-Constant.java

-CreateCert.java

-Customer.java

-Digest.java

-JarUtil.java

-ReadFile.java

-ReadKeyStore.java

-Shop.java

-SimplePrinciple.java

-VerifyCertificate.java

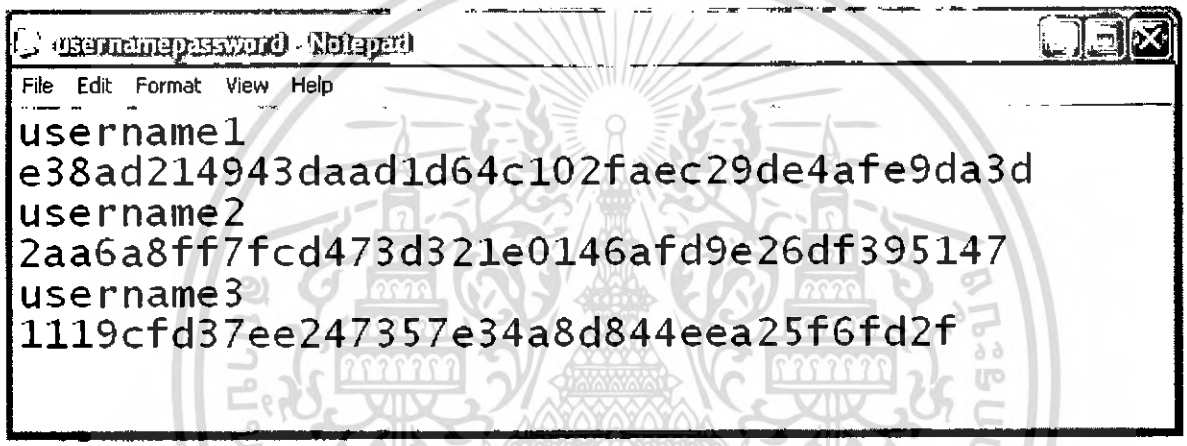
บทที่ 4

ผลการทำงานของระบบ

4.1 การทำงานของระบบส่วนที่ใช้ติดต่อกันระหว่างลูกค้าและร้านค้า

4.1.1 เมื่อลูกค้าต้องการดาวน์โหลดซอฟท์แวร์ จะต้องทำการ Login ด้วย Username/Password ตัว Login โดยตัว Login Module จะเป็นตัวทำการตรวจสอบว่าถูกต้องหรือไม่ ถ้าถูกต้องระบบก็ทำงานต่อไป

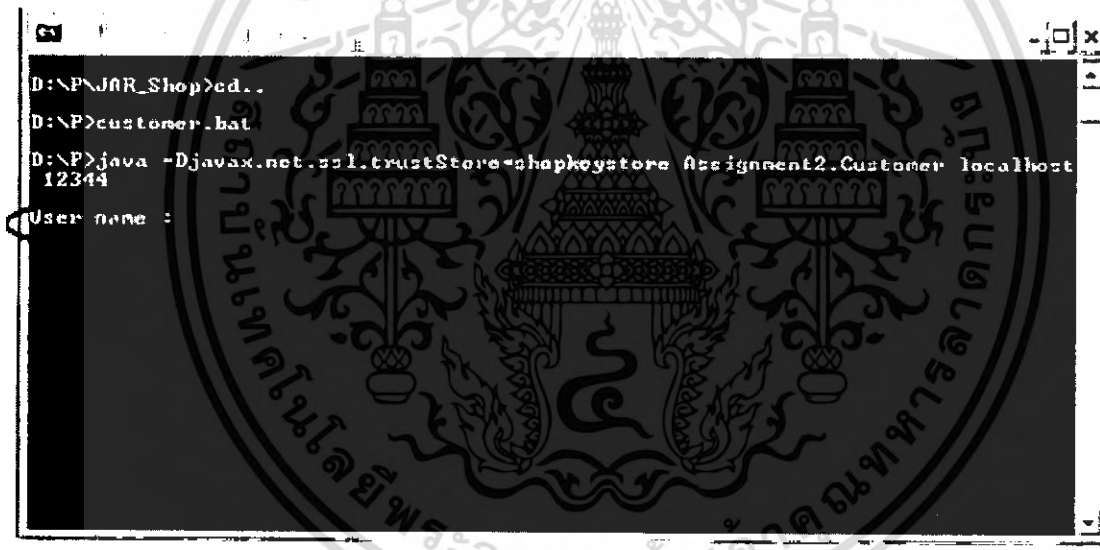
แต่ถ้าไม่ถูกต้องการเชื่อมต่อระหว่างร้านค้าและลูกค้าก็จะถูกยกเลิกทันที



รูปที่ 4.1 แสดง Username/Password

เซิร์ฟเวอร์

ไคลเอนต์



รูปที่ 4.2 แสดงหน้า Login เข้าสู่ระบบ

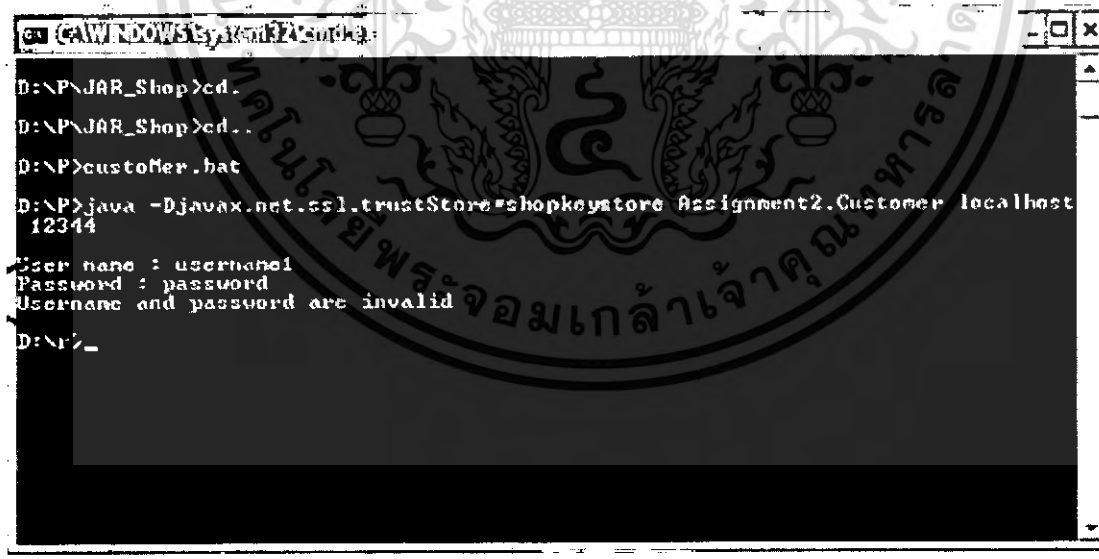
เซิร์ฟเวอร์

ไคลเอนต์

รูปที่ 4.3 แสดงการทำงานเมื่อใส่ Username/Password ถูกต้อง

เซิร์ฟเวอร์

ไคลเอนต์

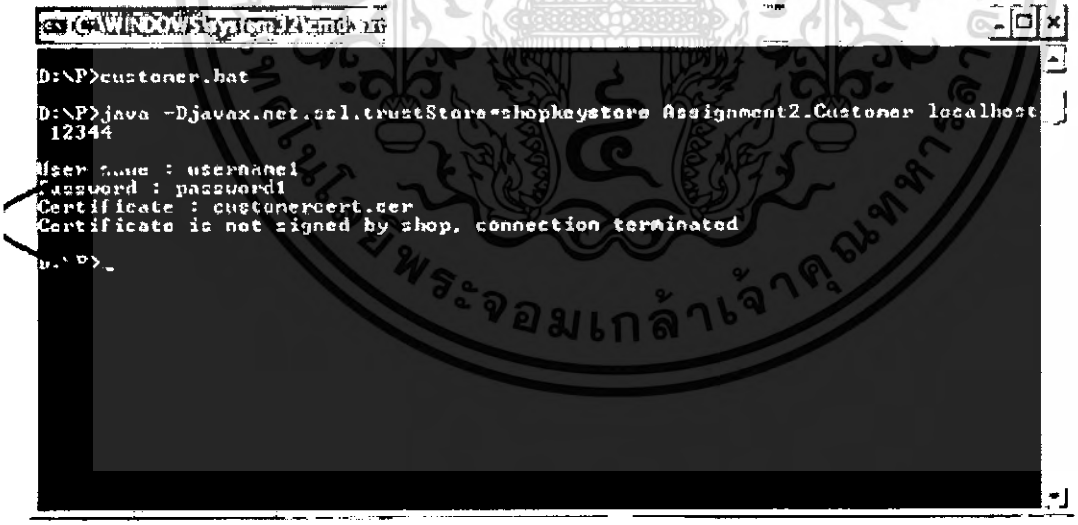


รูปที่ 4.4 แสดงการทำงานเมื่อ Username/Password ผิด

4.1.2 ระบบก็จะให้ใส่ Certificate ซึ่ง Certificate ตัวนี้จะอยู่ในรูปของ customerCert.cer ซึ่งเป็นเอกสารรับรองที่ทางร้านจะเป็นคนออกให้ โดยถ้าได้รับการตรวจสอบว่าถูกต้องทางร้านก็จะทำการแสดงรายการซอฟต์แวร์ที่มีอยู่ให้กับลูกค้า แต่ถ้าผิดระบบก็จะทำการตัดการเชื่อมต่อทันที

เซิร์ฟเวอร์

ไคลเอนต์



รูปที่ 4.5 รูปแสดงการทำงานเมื่อใส่ Certificate ผิด

4.1.3 ระบบจะแสดงรายการสินค้าที่มีอยู่ โดยลูกค้าสามารถดาวน์โหลดซอฟต์แวร์ตัวไหนก็ได้ตามต้องการ โดยการพิมพ์ตัวเลขของของซอฟต์แวร์ตัวที่ต้องการลงไป

4.2 การทำงานของระบบในส่วนของร้านค้า

ในส่วนของร้านค้าสามารถกำหนดค่าต้องได้ตามคำสั่งดังนี้

command used to create keystore and key of shop

D:\P>keytool -genkey -alias shopkey -keystore shopkeystore -storepass passshopkeystore

command used to create JAR file, and also view content in JAR file

D:\P>jar cf SPE01.jar SPE01

D:\P>jar tf SPE01.jar

command used to sign JAR file by key stored in specified keystore

D:\P>jarsigner -keystore shopkeystore -storepass passshopkeystore -keypass passshopkeystore -
signedjar SPE01Target.jar
SPE01.jar shopkey

command used to create keystore and key of customer

D:\P>keytool -genkey -alias customer -keystore customerkeystore -storepass
passcustomerkeystore

command used to create certificate from Certificate, customerCert.cer

D:\P>keytool -export -file customerCert.cer -keystore customerkeystore -storepass
passcustomerkeystore -alias customer
Certificate stored in file <customerCert.cer>

command used to import certificate (customerCert.cer) to shop keystore

D:\P>keytool -import -alias customer -f
ile customerCert.cer -keypass passcustomerkeystore -keystore shopkeystore -storepass
passshopkeystore

command used to import certificate (customerCert.cer) to customer keystore

D:\P>keytool -import -alias customerCer
t -file customerCert.cer -keypass passcustomerkeystore -keystore customerkeystor
e -storepass passcustomerkeystore

Certificate already exists in keystore under alias <customer>

Do you still want to add it? [no]: yes

Certificate was added to keystore

command used to create key with RSA algorithm

D:\P>keytool -genkey -alias custRSA -ke

yalg RSA -keysize 1024 -sigalg MD5withRSA -keystore customerkeystore -storepass

passcustomerkeystore

command used to create certificate, customerCertRSA.cer, with RSA algorithm

D:\P>keytool -export -file customerCertRSA.cer -keystore customerkeystore -storep

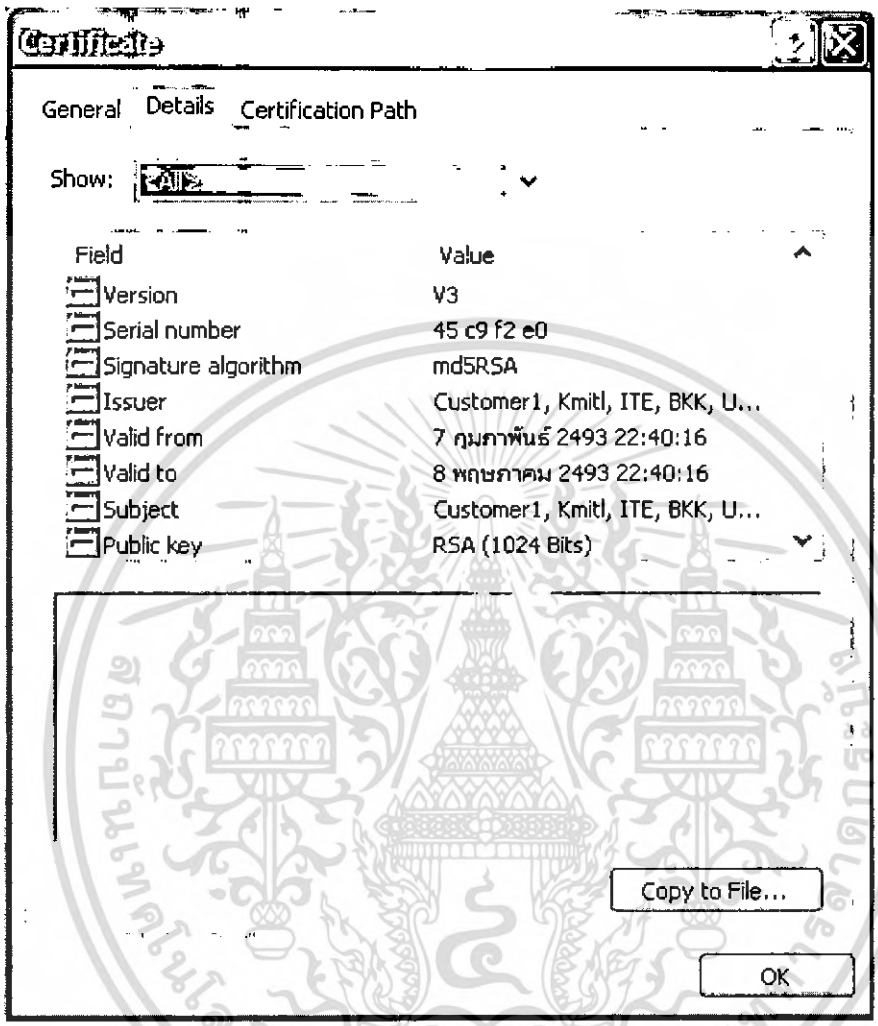
ass passcustomerkeystore -alias custRSA

Certificate stored in file <customerCertRSA.cer>

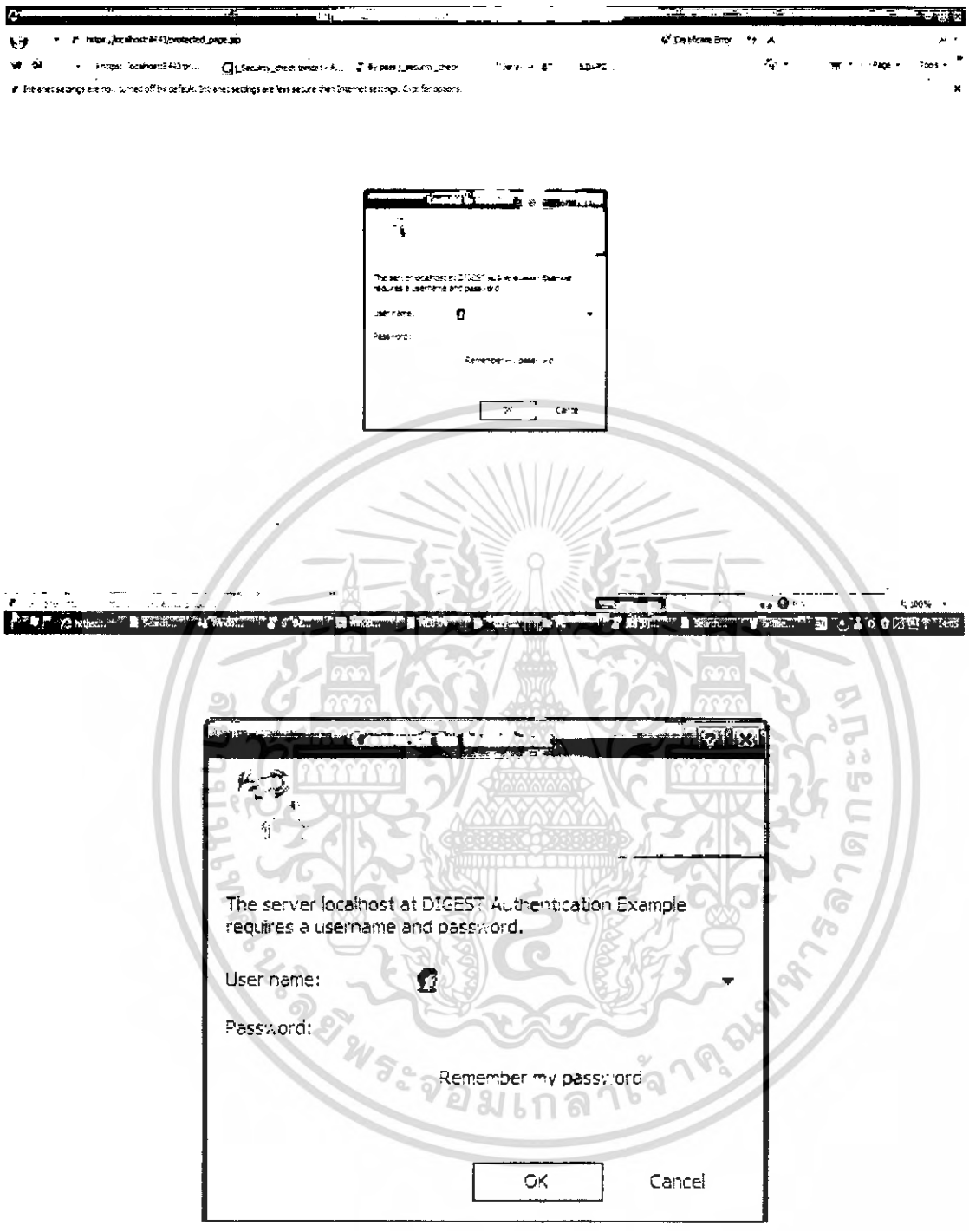
command used to import certificate to keystore

D:\P>keytool -import -alias customerCertRSA -file customerCertRSA.cer -keypass

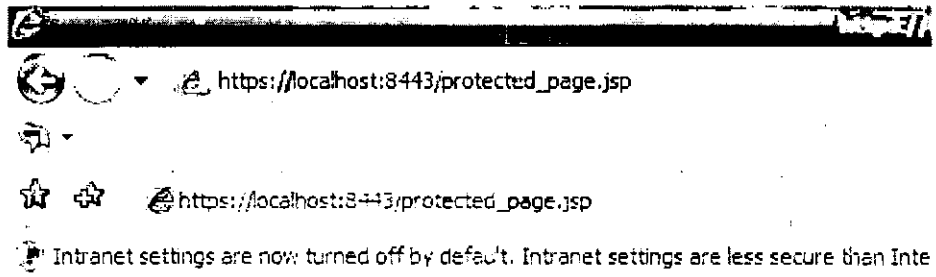
passcustomerkeystore -keystore customerkeystore -storepass passcustomerkeystore



รูปที่ 4.6 ตัวอย่าง Certificate



รูปที่ 4.7 การนำไปใช้บนเว็บ



รูปที่ 4.8 เมื่อทำการ login ผ่าน

บทที่ 5
สรุปผลโครงงาน

5.1 สรุปผลการพัฒนาโครงงาน

การสรุปผลการพัฒนาโครงงาน จะเป็นการสรุปผลการทำงานของระบบโดยทำการเปรียบเทียบกับวัตถุประสงค์ เป้าหมาย และการออกแบบระบบที่ได้มีการออกแบบไว้ก่อนหน้า โดยมีรายละเอียด ดังนี้

5.1.1 ระบบมีระบบรักษาความปลอดภัยป้องกันในส่วนของการติดต่อกันระหว่างลูกค้าและร้านค้าโดยโปรโตคอล SSL และมีการพิสูจน์ตัวตนกันระหว่างทั้งสองทาง

5.1.2 ระบบในส่วนของร้านค้าสามารถทำการสร้างค่าต่างๆ ได้ เช่นการสร้างคีย์และทำการสุ่มคีย์ การสร้าง Certificate โดยสามารถกำผู้ใช้สามรถเลือกใช้อัลกอริทึมในการเข้ารหัสได้

5.1.3 ผู้พัฒนาได้ทำการสร้างตัว bat file ขึ้นมาเพื่อให้ต่อการนำไปใช้งาน โดยไม่ต้องติดตั้งโปรแกรมใหม่ให้ยุ่งยาก

5.2 ปัญหาด้านการพัฒนา

5.2.1 เนื่องระบบดังกล่าวเป็นเพียงระบบจำลองการทำงานของเว็บไซต์การค้า ซึ่งยังไม่ใช่เว็บไซต์จริง

5.2.2 เนื่องจากระบบไม่ได้ออกแบบในส่วนของการชำระเงินของลูกค้าขึ้นมา แต่เป็นการจำลองว่าลูกค้าทุกคนได้ชำระเงินกับทางร้านค้าแล้ว

5.2.2 เนื่องจากระบบที่ออกแบบมาเป็น ทำงานอยู่ในรูปของ Command Line ซึ่งอยากต่อการใช้สำหรับผู้ไม่คุ้นเคยมาก่อน

5.3 แนวทางในการพัฒนาต่อ

5.3.1 ระบบระบบเป็นเพียงระบบการจำลองการทำงานของเว็บไซต์เท่านั้น ถ้าจะนำไปใช้งานเป็นเว็บไซต์จริงๆ นั้น จะต้องมีการพัฒนาต่อเพิ่มเติม โดยต้องนำไปประยุกต์กับภาษา JSP เพื่อทำเป็นเว็บไซต์จริงต่อไป

5.3.2 เพื่อให้ง่ายต่อการใช้งานจะต้องมีการทำในส่วนของ Interface เพิ่มเติมขึ้นมาให้เหมาะกับการใช่งานในรูปแบบต่างๆ

บรรณานุกรม

[1] Carroll, John M., "Computer Security", Butterworth Publisher , 1987

[2] Denning, Dorothy E. R., "Cryptography and Data Security", Addison-Wesley, 1983

[3] Drew Dean and Edward W. Felten," Java Security:> From HotJava to Netscape and Beyond", Oakland, 1997

[4] Lincoln Stein, "Web Security: A step-by-step:, Addison-Wesley Longman, 1998

[5] Pfleeger, Charles P., "Security in Computing: Second Edition", Prentice Hall, 1996

[6] Simon Garfinkle with Gene Spafford," Web Security and Commerce ", O'reilly &Associates, 1997

[7] Stalling, William ,"Network and International Security: Principles and Practice" , Prentice Hall , 1995

[8] ณรงค์ชัย นิมิตบุญอนันต์, "Computer Security for E-commerce" , SUM Publishing Department, 2000

[9] http://www.apache.org

[10] http://www.genome.wi.mit.edu/WWW/faqs/www-security-fag.html

[11] http://www.google.com

[12] http://www.hacker.co.za

[13] http://www.howstuffworks.com

[14] http://www.microsoft.com

[15] http://www.openmarket.com/security

[16] http://www.securityfocus.com

[17] http://www.modperl.com/perl_conference/apache_security/index.html