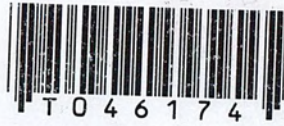


การประยุกต์ระบบความปลอดภัยสำหรับเว็บไซต์ไวด์เว็บ
World Wide Web Security Implementation



เลขหมู่.....
เลขทะเบียน... 46174
วัน, เดือน, ปี 20 ส.ค. 2546

b.....
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2544

341233-20

การประยุกต์ระบบความปลอดภัยสำหรับเว็บไซต์เว็บ

World Wide Web Security Implementation



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

ปริญญานิพนธ์ปีการศึกษา 2544

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การประยุกต์ระบบความปลอดภัยสำหรับเว็บไซต์เว็บ

World Wide Web Security Implementation

ผู้จัดทำ

- | | | |
|------------------------------|--------------|----------|
| 1. นายปรัชญา ชัยรัตน์ | รหัสประจำตัว | 41014256 |
| 2. นายสาริต ชัยวิวัฒน์ตระกูล | รหัสประจำตัว | 41014455 |



การประยุกต์ระบบความปลอดภัยสำหรับเว็ลด์ไวด์เว็บ

นายปรัชญา ชัยรัตน์ 41014256

นายสาธิต ชัยวิวัฒน์ตระกูล 41014455

อาจารย์ธนา หงษ์สุวรรณ อาจารย์ที่ปรึกษา

อาจารย์อัครเดช วัชรเทพพงษ์ อาจารย์ที่ปรึกษา

ปีการศึกษา 2544

บทคัดย่อ

ปัจจุบันแม้ว่าเว็ลด์ไวด์เว็บจะเข้ามาช่วยอำนวยความสะดวกและให้ประโยชน์ต่างๆ มากมายก็ตาม ผู้ที่เกี่ยวข้องนับตั้งแต่ผู้ใช้ธรรมดาจนไปถึงผู้ดูแลระบบ จำเป็นต้องตระหนักถึงปัญหาในด้านความปลอดภัยและผลกระทบต่างๆ ที่เกิดขึ้น ไม่ว่าจะเป็นปัญหาที่เกิดจากตัวผู้ใช้เองหรือไม่ว่าจะเป็นปัญหาที่เกิดจากการทำงานต่างๆ บนเว็ลด์ไวด์เว็บ การที่สามารถทราบลักษณะปัญหาและผลกระทบจะทำให้สามารถนำความรู้ที่ได้มาประยุกต์ใช้ให้ระบบที่มีอยู่ให้มีความปลอดภัยมากยิ่งขึ้น

เนื้อหาของปริญญานิพนธ์นี้จะประกอบด้วยการศึกษาการทำงานของเว็ลด์ไวด์เว็บที่ฝั่งเว็บเซิร์ฟเวอร์ ฝั่งเว็บไคลเอนต์และคอนเนกชัน ศึกษาถึงปัญหาด้านความปลอดภัยที่เกิดขึ้นในแต่ละส่วน รวมไปถึงแนวทางป้องกันและแก้ไขปัญหา และจะนำความรู้ที่ได้จากการศึกษามาพัฒนาเว็บไซต์ของห้องวิจัย ISAG ให้เป็นเว็บไซต์ที่มีความปลอดภัยสูงและเป็นแหล่งรวบรวมและเผยแพร่ความรู้ด้านระบบรักษาความปลอดภัยแก่สาธารณะชน

WORLD WIDE WEB SECURITY IMPLEMENTATION

Prachya Chairat

Sathit Chaiwiwattrakul

Thana Hongsuwan Advisor

Akkradach Watcharapupong Advisor

ABSTRACT

Currently, the World Wide Web has a consequent role in rendition people's life easier. Nevertheless, users who may concern with the World Wide Web must realize to any effects and problems those will occur to users. No matter where and what problems, it is potentially to reduce the risks and protect the problems if we know what and how the problems take place

The main subject of this thesis will conclude with an obligation of the Web Server, the Web Client and the Connection as well. This also includes the type of the problems and the protection for each type. In other parts, we use all of information from our research to develop our research lab's website which is secure website and publicize information about computer security.

กิตติกรรมประกาศ

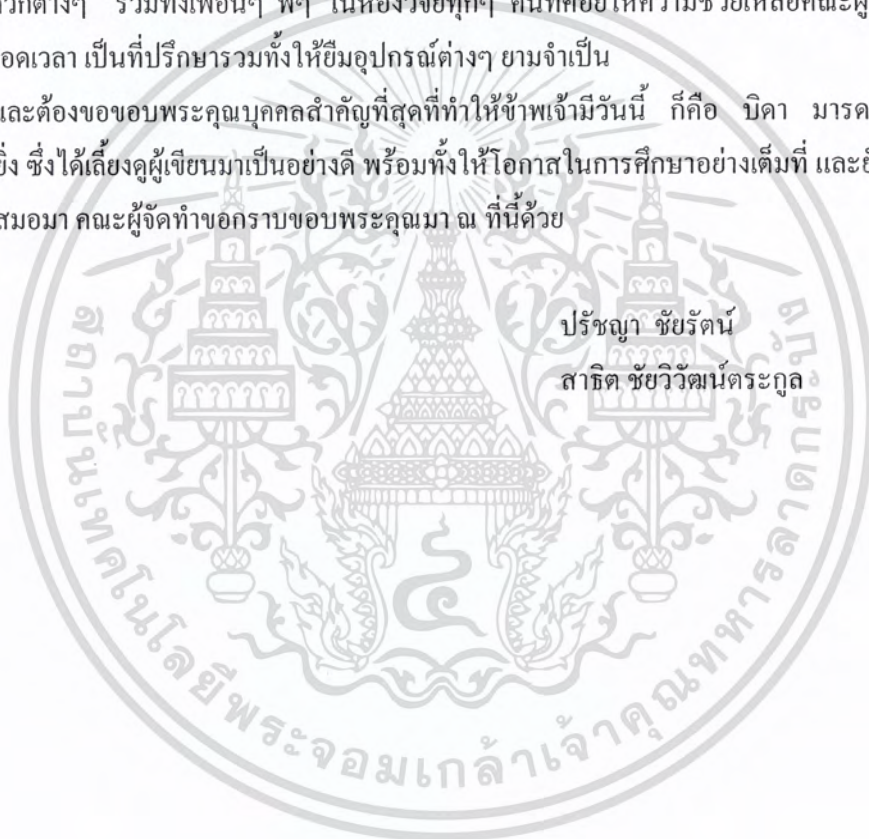
วิทยานิพนธ์ฉบับนี้คงไม่อาจเสร็จได้ด้วยดี หากไม่ได้รับความช่วยเหลือ และร่วมมือจากหลาย ๆ ฝ่ายด้วยกัน บุคคลแรกที่ต้องกล่าวถึงเพราะเป็นผู้จุดประกายความคิดให้เกิดหัวข้อวิทยานิพนธ์นี้ขึ้นมา นั่นก็คือ อาจารย์ธนา หงษ์สุวรรณและอาจารย์อัครเดช วัชรเทพวงษ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้ความเอาใจใส่ แนะนำ และช่วยเหลือเสมอมา ซึ่งต้องขอขอบพระคุณเป็นอย่างมาก

นอกจากนี้ต้องขอขอบพระคุณอาจารย์ทุกท่านในสถาบันที่ได้สั่งสอนคณะผู้จัดทำจนมีความรู้ความสามารถจนถึงทุกวันนี้ รวมทั้งภาควิชาวิศวกรรมคอมพิวเตอร์โดยเฉพาะห้องวิจัยและพัฒนากระบวนการรักษาความปลอดภัยข้อมูล (ISAG) ที่ได้เอื้อเฟื้อสถานที่ ให้คณะผู้จัดทำได้ทำการวิจัย และช่วยอำนวยความสะดวกต่างๆ รวมทั้งเพื่อนๆ พี่ๆ ในห้องวิจัยทุกๆ คนที่คอยให้ความช่วยเหลือคณะผู้จัดทำในการทำงานตลอดเวลา เป็นที่ปรกัษรวมทั้งให้ยืมอุปกรณ์ต่างๆ ยามจำเป็น

และต้องขอขอบพระคุณบุคคลสำคัญที่สุดที่ทำให้ข้าพเจ้ามีวันนี้ ก็คือ บิดา มารดา อันเป็นที่เคารพรักยิ่ง ซึ่งได้เลี้ยงดูผู้เขียนมาเป็นอย่างดี พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้กำลังใจเอาใจใส่เสมอมา คณะผู้จัดทำขอกราบขอบพระคุณมา ณ ที่นี้ด้วย

ปรีชญา ชัยรัตน์

สาริต ชัยวิวัฒน์ตระกูล



สารบัญ

หน้า

บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูปภาพ	VI
สารบัญตาราง	VIII
บทที่ 1 บทนำ	1
1.1 ความสำคัญและที่มา	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 เป้าหมายของโครงการ	1
1.4 ขอบเขตของโครงการ	2
1.5 วิธีการดำเนินงาน	3
บทที่ 2 โพรโตคอล HTTP	4
2.1 การร้องขอบริการ	4
2.2 การตอบกลับ/การให้บริการ	5
2.3 การเชื่อมต่อระหว่างผู้ใช้บริการและผู้ให้บริการ	7
บทที่ 3 เวิลด์ไวด์เว็บ	9
3.1 ส่วนประกอบที่สำคัญของเวิลด์ไวด์เว็บ	9
3.2 หลักการทำงานของเวิลด์ไวด์เว็บ	10
บทที่ 4 การเข้าและถอดรหัส	12
4.1 การเข้าและถอดรหัสแบบคีย์เหมือน	12
4.2 การเข้าและถอดรหัสแบบคีย์ต่าง	12
4.3 การแกะรหัส	15
บทที่ 5 โพรโตคอล SSL	17
5.1 องค์ประกอบของ SSL	18
5.2 การเข้ารหัสและถอดรหัสใน SSL	18
5.3 การตกลงเชื่อมต่อใน SSL	19
5.4 การยืนยันตัวเซิร์ฟเวอร์	21
5.5 การยืนยันตัวไคลเอนต์	21
บทที่ 6 ปัญหาความปลอดภัยที่เว็บเซิร์ฟเวอร์	24

สารบัญ (ต่อ)

	หน้า
6.1 Apache	24
6.2 CGI (Common Gateway Interface)	34
6.4 PHP	36
6.5 MySQL	50
บทที่ 7 ปัญหาความปลอดภัยที่เว็บไคลเอนต์	56
7.1 ปัญหาความปลอดภัยที่ตัวเว็บไคลเอนต์	56
บทที่ 8 ปัญหาความปลอดภัยในการสื่อสารระหว่างฝั่งเซิร์ฟเวอร์และฝั่งไคลเอนต์	94
8.1 ปัญหาความปลอดภัยสำหรับการสื่อสารระหว่างเว็บไคลเอนต์และเว็บเซิร์ฟเวอร์	94
บทที่ 9 การจัดตั้งเว็บไซต์ของห้องวิจัย ISAG ที่มีความปลอดภัย	98
9.1 วัตถุประสงค์ของเว็บไซต์ของห้องวิจัย	98
9.2 ไฟเจอร์ที่สำคัญของเว็บไซต์ของห้องวิจัย	98
9.3 สิทธิการใช้งานเว็บไซต์	99
9.4 โครงสร้างของเว็บไซต์	100
9.5 ตัวอย่างความปลอดภัยของเว็บไซต์ของห้องวิจัย	101
บทที่ 10 บทวิจารณ์และสรุป	111
10.1 สรุปผลการดำเนินงาน	111
10.2 แนวทางการพัฒนาต่อ	111
บรรณานุกรม	113

สารบัญรูปภาพ

หน้า

รูปที่ 1-1 แสดงขอบเขตของโครงการในส่วนของการศึกษาการทำงาน	2
ปัญหาและแนวทางการแก้ไขของเว็บเซิร์ฟเวอร์ เว็บไคลเอนต์ และคอนเนกชัน	
รูปที่ 2-1 แสดงการทำงานของโปรโตคอล HTTP	4
รูปที่ 3-1 เวิลด์ไวด์เว็บทำงานโดยใช้โปรโตคอล HTTP ใน Application Layer ที่ทำงานอยู่เหนือ	11
โปรโตคอล TCP/IP ใน Network Layer	
รูปที่ 4-1 แสดงการเข้าและถอดรหัส	12
รูปที่ 4-2 การเข้าและถอดรหัสโดยใช้กุญแจสาธารณะ: ปกปิดเป็นความลับ	14
รูปที่ 4-3 การเข้าและถอดรหัสโดยใช้กุญแจสาธารณะ: การพิสูจน์ตน	14
รูปที่ 4-4 การเข้าและถอดรหัสโดยใช้กุญแจสาธารณะ: ปกปิดเป็นความลับและการพิสูจน์ตน	15
รูปที่ 5-1 แสดงการทำงานของ SSL ในชั้น Application Layer และ Transport Layer	17
รูปที่ 5-2 แสดงโครงสร้างของ Handshake Protocol	20
รูปที่ 5-3 แสดงขั้นตอนการทำงานของการตรวจสอบในส่วนของเซิร์ฟเวอร์	22
รูปที่ 5-4 แสดงขั้นตอนการทำงานการตรวจสอบในส่วนของไคลเอนต์	22
รูปที่ 6-1 แสดงสถาปัตยกรรมคร่าวๆ ของ Apache Webserver	25
รูปที่ 6-2 สถาปัตยกรรมของ Apache Core	26
รูปที่ 6-3 สถาปัตยกรรมของ Apache Module	26
รูปที่ 6-4 แสดงถึงสถาปัตยกรรมโครงสร้างของ Apache อย่างละเอียด	29
รูปที่ 6-5 แสดงขั้นตอนการทำงานของ CGI	35
รูปที่ 6-6 ตัวอย่างแบบฟอร์มที่ใช้ในการรับส่งข้อมูล	35
รูปที่ 6-7 แสดงขั้นตอนการทำงานของ PHP	38
รูปที่ 6-8 ตัวอย่างแบบฟอร์ม	44
รูปที่ 7-1 แสดงหน้าเว็บเพจที่ทำปลอมขึ้นมา	64
รูปที่ 7-2 แสดง Properties ของหน้าเอกสารที่ได้ปลอมขึ้นมา	65
รูปที่ 7-3 แสดงโครงสร้างการทำงานของแอปพลิเคชันที่ใช้วีบีสคริปต์จัด	67
รูปที่ 7-4 แสดงส่วนต่างๆ ของเทคโนโลยี ActiveX	69
รูปที่ 7-5 แสดงส่วนของ ActiveX Control ที่ถูกดาวน์โหลดและทำการรันที่เว็บเบราว์เซอร์	71
รูปที่ 7-6 แสดงการตั้งค่าการทำงานในส่วนของ ActiveX	74
รูปที่ 7-7 แสดง Internet Explorer หลังจากได้ติดตั้งแพตช์แล้ว	79
รูปที่ 7-8 แสดงข้อมูลที่เก็บไว้ในเครื่องของผู้ใช้	83
รูปที่ 7-9 แสดงการปรับแต่งระดับความปลอดภัยในการติดตั้งลูกก็ใน IE	84

สารบัญรูปภาพ (ต่อ)

	หน้า
รูปที่ 7-10 แสดงการปรับแต่งระดับความปลอดภัยในการติดตั้งคุกกี้นใน Netscape Navigator	85
รูปที่ 7-11 การเพิ่มความปลอดภัยจากจาวาสคริปต์ จาวาแอปเพล็ตและ ActiveX	90
รูปที่ 7-12 รูปแบบการเช็คค่าความปลอดภัยของคุกกี้นและจาวาแอปเพล็ต	91
รูปที่ 7-13 รูปแบบการเช็คค่าความปลอดภัยของจาวาแอปเพล็ตใน Netscape	92
รูปที่ 8-1 แสดงการทำงานของเว็บสปูฟิง (Web Spoofing)	94
รูปที่ 8-2 แสกเกอร์ทำการดักลอกข้อมูล เพื่อนำข้อมูลไปใช้ประโยชน์	96
รูปที่ 8-3 แสดงการเกิดการเปลี่ยนแปลงข้อมูลในระหว่างการส่ง	97
รูปที่ 9-1 แสดงการการเข้ารหัสผ่านที่เก็บไว้ในฐานข้อมูลทำให้มีความปลอดภัยมากขึ้น	102
รูปที่ 9-2 แสดงการใช้เครื่องมือในการสแกนหาไฟล์ password ที่เก็บไว้ในเซิร์ฟเวอร์	102
รูปที่ 9-3 แสดงการเข้ารหัสไฟล์และไดเรกทอรีเพื่อป้องกันไฟล์และข้อมูลในไดเรกทอรีที่สำคัญ	103
รูปที่ 9-4 แสดงการใช้ชื่อและรหัสที่แตกต่างกัน ในการติดต่อกับฐานข้อมูลในแต่ละไฟล์ที่มีความสำคัญต่างกัน	103
รูปที่ 9-5 แสดงข้อผิดพลาดที่เกิดขึ้นโดยที่ไม่ได้ปิดการแสดงผลข้อผิดพลาดไว้ ทำให้ผู้บุกรุกทราบข้อมูล เกี่ยวกับระบบได้ ซึ่งถ้าหากมีการปิดการแสดงผลข้อผิดพลาดนี้ ผู้บุกรุกก็จะไม่ได้ข้อมูลอะไรไปเลย	104
รูปที่ 9-6 แสดงการใช้ล็อกไฟล์เพื่อให้สามารถตรวจสอบการทำงานของเว็บไซต์ได้	104
รูปที่ 9-7 แสดงไฟล์ที่อยู่ภายในไดเรกทอรีนั้นเมื่อไม่มีดีฟอลต์ไฟล์ ซึ่งทำให้ผู้บุกรุกสามารถดูไฟล์ทั้งหมดที่มีอยู่ได้	105
รูปที่ 9-8 แสดงการปรับแต่งค่าเพื่อไม่ให้แสดงชื่อไฟล์ในไดเรกทอรีเมื่อไม่มีดีฟอลต์ไฟล์ทำให้ระบบมีความปลอดภัยมากยิ่งขึ้น	105
รูปที่ 9-9 ตัวอย่างหน้าแรกของเว็บไซต์ซึ่งประกอบไปด้วยข่าวประกาศต่างๆ	106
รูปที่ 9-10 ตัวอย่างหน้า Sitemap ซึ่งจะแสดงแผนที่ของเว็บไซต์ทั้งหมด	106
รูปที่ 9-11 ตัวอย่างหน้าที่แสดงสมาชิกของห้องวิจัย	107
รูปที่ 9-12 ตัวอย่างหน้าที่แสดงบอร์ดความคิดเห็นต่างๆ	107
รูปที่ 9-13 ตัวอย่างหน้าที่แสดงโครงการงานของห้องวิจัย	108
รูปที่ 9-14 ตัวอย่างหน้าที่แสดงลิงค์ของห้องวิจัยแยกตามหมวดหมู่	108
รูปที่ 9-15 ตัวอย่างหน้าจอ login เข้าสู่ระบบสำหรับสมาชิกเท่านั้น	109
รูปที่ 9-16 ตัวอย่างหน้าที่แสดงข่าวประกาศภายในเมื่อสมาชิก login เข้ามาสู่ระบบแล้ว	109
รูปที่ 9-17 ตัวอย่างหน้าที่ใช้สำหรับประกาศข่าว	110
รูปที่ 9-18 ตัวอย่างหน้าที่ใช้สำหรับเพิ่มข้อมูลโครงการงาน	110

สารบัญตาราง

หน้า

ตารางที่ 2-1 แสดงรายละเอียดของหมายเลขสถานภาพการทำงานของโปรโตคอล HTTP	6
ตารางที่ 4-1 การเปรียบเทียบการเข้าและถอดรหัสแบบคีย์เหมือนกับการเข้าและถอดรหัสแบบคีย์ต่าง	13
ตารางที่ 4-2 แสดงระยะเวลาที่ใช้ในการแกะรหัส	16
ตารางที่ 9-1 แสดงโครงสร้างและสิทธิในการใช้งานของเว็บไซต์	100



บทที่ 1

บทนำ

1.1 ความสำคัญและที่มา

เนื่องจากในปัจจุบัน เวิลด์ไวด์เว็บได้เข้ามามีบทบาทสำคัญ และเป็นส่วนหนึ่งของการดำเนินชีวิตประจำวันของคนเรามากขึ้น โดยกิจกรรมต่างๆ ที่ดำเนินอยู่ในปัจจุบันจะมีเวิลด์ไวด์เว็บเข้ามาเกี่ยวข้องด้วยเสมอ เพื่อช่วยอำนวยความสะดวกและความรวดเร็วแก่ผู้ใช้งานมากมาย

แม้ว่าเวิลด์ไวด์เว็บจะช่วยอำนวยความสะดวกมากมายให้แก่ผู้ใช้ แต่ก็ยังแฝงไปด้วยภัยร้ายต่างๆ มากมายด้วยเช่นกัน ไม่ว่าจะเป็นการแอบอ้างผู้ใช้และละเมิดความเป็นส่วนตัว (Privacy) การถูกโจรกรรมข้อมูลหรือทำการแก้ไขข้อมูลโดยแฮกเกอร์ (Hacker) แครกเกอร์ (Cracker) หรือแม้กระทั่งปัญหาที่เกิดจากไวรัส (Virus) หรือเวิร์ม (Worm) ต่างๆ เป็นต้น ซึ่งอาจจะมาในลักษณะการโจมตีแบบเปิดเผยหรือการโจมตีโดยไม่ให้รู้ตัว ก่อให้เกิดความเสียหายต่อผู้ใช้ได้

ดังนั้นการที่จะสามารถป้องกันตนเองให้ปลอดภัยจากอันตรายต่างๆ ได้นั้น ควรที่จะต้องศึกษาถึงปัญหาและความปลอดภัยที่เกิดขึ้นบนเวิลด์ไวด์เว็บให้ถูกต้องครบถ้วนเสียก่อน จึงจะสามารถนำมาประยุกต์ใช้ให้เข้ากับระบบที่มีอยู่ได้อย่างถูกต้องและเหมาะสม

1.2 วัตถุประสงค์ของโครงการ

1. ศึกษาการทำงาน ลักษณะของปัญหา และแนวทางการรักษาความปลอดภัยที่ฝั่งเว็บเซิร์ฟเวอร์
2. ศึกษาการทำงาน ลักษณะของปัญหา และแนวทางการรักษาความปลอดภัยที่ฝั่งไคลเอนต์
3. ศึกษาการทำงาน ลักษณะของปัญหา และแนวทางการรักษาความปลอดภัยในการสื่อสารระหว่างฝั่งเซิร์ฟเวอร์และฝั่งไคลเอนต์
4. สามารถจัดตั้งเว็บไซต์ของห้องวิจัยที่มีความปลอดภัยสูง

1.3 เป้าหมายของโครงการ

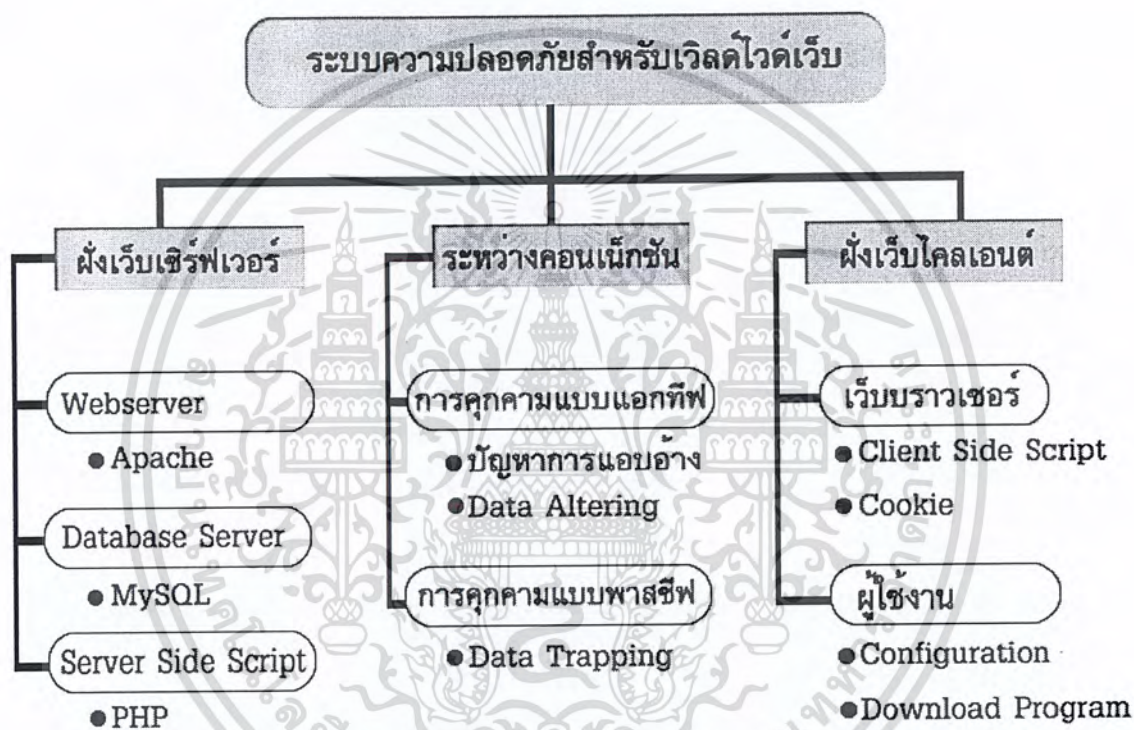
1. เป็นเอกสารอ้างอิงสำหรับระบบความปลอดภัยสำหรับเวิลด์ไวด์เว็บ
2. เป็นคู่มือแนะนำการดำเนินการจัดตั้งเว็บไซต์ที่มีความปลอดภัย
3. ดำเนินการจัดตั้งเว็บไซต์ของห้องวิจัยที่มีความปลอดภัยสูง

1.4 ขอบเขตของโครงการ

โครงการนี้ได้ทำการศึกษาถึงความปลอดภัยรวมถึงการแก้ปัญหาสำหรับเว็บไซต์เว็บ โดยจะทำการศึกษาและรวบรวมปัญหาให้ครอบคลุม ครบถ้วนและถูกต้องมากที่สุด โดยใช้เกณฑ์ในการพิจารณา สถานที่เกิด (Location) ของความไม่ปลอดภัย ซึ่งสามารถแบ่งได้ 3 ส่วนคือ

- ปัญหาความปลอดภัยที่ฝั่งเว็บเซิร์ฟเวอร์
- ปัญหาความปลอดภัยที่ฝั่งเว็บไคลเอนต์
- ปัญหาความปลอดภัยสำหรับการสื่อสารระหว่างเว็บไคลเอนต์และเว็บเซิร์ฟเวอร์

โดยขอบเขตของงานวิจัยสามารถแสดงรายละเอียดได้ดังแผนผังต่อไปนี้



รูปที่ 1-1 แสดงขอบเขตของโครงการในส่วนของการศึกษาการทำงาน ปัญหาและแนวทางการแก้ไขของเว็บเซิร์ฟเวอร์ เว็บไคลเอนต์ และคอนเนกชัน

นอกจากนี้ยังได้ทดสอบการทำงาน โดยทำการจัดตั้งเว็บไซต์ของห้องวิจัย ISAG ซึ่งจะเป็นการประยุกต์นำความรู้ทางด้านระบบรักษาความปลอดภัยมาใช้ในการจัดตั้งเว็บไซต์ที่มีความปลอดภัยและเผยแพร่ความรู้ทางด้านระบบรักษาความปลอดภัยแก่สาธารณะชน

โดยภาษาที่ได้เลือกมาใช้ในการทดสอบและทำการศึกษาในโครงการนี้คือ PHP เนื่องจากเป็นภาษาที่มีคุณลักษณะเด่นหลายประการ ไม่ว่าจะเป็น Open source, Cross platform มีการทำงานได้รวดเร็วและสนับสนุนโพรโตคอลมากมาย โดยทำงานบนระบบปฏิบัติการที่ใช้ OpenBSD ซึ่งใช้ Apache เป็นเว็บเซิร์ฟเวอร์และใช้ MySQL เป็นดาต้าเบสเซิร์ฟเวอร์

1.5 วิธีการดำเนินงาน

งานวิจัยในโครงการนี้จะเริ่มด้วยวิธีการศึกษาทฤษฎีและหลักการพื้นฐานต่างๆ ที่เกี่ยวกับงานวิจัยซึ่งจะมีเรื่องหลักๆ คือ โพรโทคอล HTTP, เวิลด์ไวด์เว็บ, การเข้ารหัสและการถอดรหัส(Cryptography), โพรโทคอล SSL(Secure Sockets Layer), เว็บเซิร์ฟเวอร์อาปาเช่ (Apache Webserver), MySQL Database Server และศึกษาการเขียนโปรแกรมเซิร์ฟเวอร์ไซด์สคริปต์ด้วยภาษา PHP จากนั้นก็จะเริ่มศึกษาและวิเคราะห์ปัญหาความปลอดภัยสำหรับเวิลด์ไวด์เว็บ โดยจะแบ่งออกเป็น 3 ส่วนด้วยกัน คือ ปัญหาความปลอดภัยที่ฝั่งเว็บเซิร์ฟเวอร์ ปัญหาความปลอดภัยที่ฝั่งเว็บไคลเอนต์ และปัญหาความปลอดภัยสำหรับการสื่อสารระหว่างเว็บไคลเอนต์และเว็บเซิร์ฟเวอร์

จากนั้นจะเป็นการนำความรู้ที่ได้รวมถึงปัญหาด้านความปลอดภัยจากการศึกษามาประยุกต์ใช้ในการติดตั้งเว็บเซิร์ฟเวอร์พร้อมดำเนินการจัดตั้งเว็บไซต์ของห้องวิจัยที่มีความปลอดภัยสูง ซึ่งก็จะใช้เป็นแหล่งเผยแพร่ข้อมูลทางด้านระบบรักษาความปลอดภัยสู่สาธารณะชนต่อไป



บทที่ 2

โพรโทคอล HTTP

HTTP ย่อมาจาก Hypertext Transfer Protocol เป็นข้อกำหนด หรือวิธีในการติดต่อสื่อสาร หรือจัดการข้อมูลประเภท HyperText หรือเรามักเรียกว่า HyperMedia ทั้งนี้เพราะตัวรูปแบบเอกสารเปลี่ยนไปจากเดิมที่เป็นเอกสารแบบข้อความ (Text) เชื่อมโยงกันเป็นเครือข่าย แต่ปัจจุบันข้อมูลอาจเป็นทั้งภาพ เสียง มัลติมีเดีย(Multimedia) และอื่นๆ ซึ่งโพรโทคอล HTTP เป็นพื้นฐานในการใช้งานหรือการทำงานสื่อสารของเครือข่ายเวิลด์ไวด์เว็บ โดยมีโครงสร้างและลักษณะการทำงานดังต่อไปนี้



รูปที่ 2-1 แสดงการทำงานของโพรโทคอล HTTP

จากภาพข้างต้นแสดงการทำงานของโพรโทคอล HTTP โดยสามารถแบ่งส่วนการทำงานออกเป็นสามส่วนหลักๆ คือ ส่วนเว็บไคลเอนต์ เว็บเซิร์ฟเวอร์ และเครือข่าย ตามหลักการพื้นฐานของวิธีการทำงานแบบไคลเอนต์/เซิร์ฟเวอร์ ซึ่งฝั่งไคลเอนต์จะใช้งานโปรแกรมเว็บเบราว์เซอร์ติดต่อกับเว็บเซิร์ฟเวอร์ ซึ่งคือผู้ให้บริการเว็บโดยโปรแกรมหาดังกล่าวจะเรียกว่า HTTPD (Hypertext Transfer Protocol Daemon) ปกติแล้วจะทำงานที่พอร์ต 80 สำหรับหลักการของไคลเอนต์-เซิร์ฟเวอร์ คือ มีตัวไคลเอนต์เป็นผู้ขอใช้บริการและมีเซิร์ฟเวอร์เป็นผู้ให้บริการ โดยวิธีการทำงานจะใช้วิธีการร้องขอ (Request) และการตอบสนองการขอบริการ (Response/Reply) ซึ่งมีลักษณะดังนี้

2.1 การร้องขอบริการ

สำหรับ HTTP จะมีคำสั่งหลักๆ ในการจัดการดังนี้ OPTION, HEAD, PUT, DELETE, TRACE, GET และ POST แต่คำสั่งหลักๆ สำหรับผู้ร้องขอใช้บริการที่มักใช้บ่อยคือ GET, POST และ HEAD โดยรายละเอียดการใช้คำสั่งสำหรับการร้องขอนั้นจะมีการบรรยายละเอียดไว้ในตัวของโพรโทคอล HTTP

```

Get/Index.html HTTP/1.1
Connection: Keep Alive
User-Agent: Mozilla/2.01
Pragma: no-cache
Host: 161.246.10.21
Accept: image/gif, image/jpeg, image/pjpeg, */*

```

การร้องขอบริการจะมีความเกี่ยวข้องกับการระบุสถานที่ที่ต้องการใช้บริการและรายละเอียดที่อยู่ของข้อมูลต่างๆ ตามหลักการของ URL (Uniform Resource Locator) ซึ่ง URL คือส่วนระบุเพิ่มเติมในการเข้าถึงข้อมูล โดยสามารถใช้วิธีการหรือโปรโตคอลได้หลายวิธี ดังนั้นข้อดีของ URL คือสามารถเข้าถึงแหล่งข้อมูลต่างๆ ได้ตามวิธีการที่ควรจะเป็นหรือเหมาะสม เช่น <http://www.kmitl.ac.th> นอกจากนี้ยังสามารถที่จะเรียกใช้ได้ในลักษณะของแอปพลิเคชันอื่นๆ ได้ เช่น <mailto:itsource@hotmail.com> เป็นต้น

จากการร้องขอบริการข้างต้นจะเป็นการเรียกขอโฮมเพจที่ชื่อ “index.html” โดยตรง ซึ่งอาจจะระบุแบบส่วนขยาย URL ได้ เช่น [/~s1014455/index.html](http://161.246.10.21/~s1014455/index.html) เป็นต้น ส่วนสำคัญถัดมาของการร้องขอบริการคือส่วนของ User Agent ซึ่งเป็นการระบุรายละเอียดเกี่ยวกับสถานะภาพของไคลเอนต์ เช่น โปรแกรมที่ใช้งานเป็นบราวเซอร์ของ Netscape หรือ Internet Explorer ใช้งานกับวินโดวส์ 2000 เป็นต้น ส่วนต่อมาก็คือโฮสต์ (Host) ซึ่งจะเป็นไต่ปลายทางและรายละเอียดนี้จะสัมพันธ์กับส่วนเพิ่มเติมของ URL ดังเช่นจากตัวอย่างจะเป็นการระบุ URL เท่ากับ <http://161.246.10.21/index.html> หรือ <http://www.kmitl.ac.th/index.html> ซึ่งในกรณีที่สองจะต้องมีการนำค่าชื่อโฮสต์ www.kmitl.ac.th ไปแปลงเป็นไอพีแอดเดรสเท่ากับ 161.246.10.21 และ GET จะได้เท่ากับ /index.html

2.2 การตอบกลับ/การให้บริการ

ในการตอบกลับการให้บริการของ HTTP นั้นจะมีรูปแบบการใช้งานเหมือนโปรโตคอลอื่นๆ ตามหลักการของไคลเอนต์-เซิร์ฟเวอร์ทั่วไป คือ จะมีการส่งค่ากลับไปด้วยหมายเลขที่เรียกว่า “Response Tag Number” หรือ “Status Code” และจะตามด้วยรายละเอียดข้อความซึ่งอธิบายหมายเลขนั้น จากนั้นส่วนท้ายสุดที่อาจจะมีส่วนมาคือตัวของข้อมูลจริงๆ ในกรณีที่มีการขอข้อมูล เช่น จากการร้องขอข้างต้น เราจะได้ข้อมูลที่เว็บเซิร์ฟเวอร์ส่งกลับมาให้ดังนี้

```

Trying 127.0.0.0 ... Connected to localhost.
Escape character is '^]'
HTTP/1.1 200 OK
Date: Sun, 13 Aug 2000 17:17:34 GMT
Server: Apache/1.3.20
Content-type: text/html

```


Content-length: 1339

Last-modified: Mon, 05 Aug 2000 22:39:GMT

Connection: Keep_Alive

Keep-Alive: timeout=15, max=5

<HTML>

<HEAD><TITLE>KMITL Website</TITLE></HEAD>

<BODY>

:

</BODY>

</HTML>

หมายเลขการตอบสนอง (Response Number) จะเป็นค่ามาตรฐานแต่ข้อความที่ตามมานั้นอาจจะไม่เหมือนกันก็ได้ทั้งนี้ขึ้นอยู่กับผลิตภัณฑ์ของเว็บเซิร์ฟเวอร์ ตัวเลขทุกหลักของ Response Tags Number ล้วนมีความหมาย เริ่มที่หลักแรกจะเป็นการแสดงความมีประสิทธิภาพของการร้องขอและตอบกลับว่าเป็นอย่างไร “200 OK” หรือ “500 Error” เป็นต้น ส่วนตัวเลขหลักที่สองจะแจ้งให้ทราบถึงชนิดของการทำงาน เช่น หากเกิด Error เป็นชนิดไหนและส่วนหลักสุดท้ายเป็นส่วนย่อยในประเภทของการทำงานนั้นๆ เป็นการขยายในส่วนรายละเอียดของหลักที่สอง

Response Tags Number	รายละเอียด
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	Unauthorized
402	Payment Required
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required
408	Request Time-out
409	Conflict
410	Gone
411	Length Required
412	Precondition Failed
413	Request Entity Too Large
414	Request-URI Too Large
415	Unsupported Media Type
500	Internal Server Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Time-out
505	HTTP Version not supported

ตาราง 2-1 แสดงรายละเอียดของหมายเลขสถานะภาพการทำงานของโปรโตคอล HTTP

2.3 การเชื่อมต่อระหว่างผู้ให้บริการและผู้ให้บริการ

การเชื่อมต่อระหว่างผู้ให้บริการและผู้ให้บริการสามารถแบ่งออกได้เป็น 2 ลักษณะ คือ

2.3.1 การเชื่อมต่อโดยตรง

ลักษณะการทำงานของการทำงานเชื่อมต่อโดยตรง คือ ผู้ขอใช้บริการจะติดต่อกับเซิร์ฟเวอร์หรือผู้ให้บริการโดยตรง ซึ่งส่วนของไคลเอนต์จะเรียกว่า User Agent โดยมีเว็บเบราว์เซอร์ทำหน้าที่นี้และส่วนเซิร์ฟเวอร์จะเรียกว่า Origin ซึ่งจะทำงานกับเว็บเบราว์เซอร์หรือ User Agent โดยตรง

2.3.2 การเชื่อมต่อผ่านตัวกลาง

ลักษณะการติดต่อแบบนี้ส่วนของ User Agent ไม่สามารถติดต่อกับ Origin ได้โดยตรง นั่นคือต้องติดต่อผ่านตัวกลางทุกครั้งที่มีการร้องขอบริการและการตอบสนองก็ต้องผ่านตัวกลางเช่นกัน ดังนั้นการร้องขอหรือการตอบสนองจะมีลักษณะเหมือนลูกโซ่โยงผ่านเป็นช่วงๆ เรียกว่า Request Chain/Response Chain

ประเภทของตัวกลาง (Intermediate) ตามข้อกำหนดของ HTTP มี 3 ประเภท คือ

- Tunnel: ทำหน้าที่เชื่อมต่อเท่านั้น อาจมีไว้เพื่อความประสงค์อะไรบางอย่างแต่ตัวกลางนี้จะไม่ทำหน้าที่หรืออำนาจในการเปลี่ยนข้อมูลที่วิ่งผ่าน
- Proxy: ส่วนนี้สามารถปรับปรุงรายละเอียด มีการประยุกต์ใช้งานได้ทั้งสองส่วน คือ โคลเอนต์และเซิร์ฟเวอร์ มักจะนำมาติดตั้งเป็น Cache Server หรือ FireWall
- Gateway: ส่วนนี้มักทำหน้าที่เชื่อมต่อ ในกรณีที่ไม่สามารถติดต่อหรือใช้งานเชื่อมต่อกับตัวเซิร์ฟเวอร์ได้โดยตรง



บทที่ 3

เวิลด์ไวด์เว็บ

เวิลด์ไวด์เว็บ (World Wide Web: WWW) เป็นรูปแบบการให้บริการแก่ผู้ใช้บริการอินเทอร์เน็ต โดยเป็นระบบที่เชื่อมต่อข้อมูลในคอมพิวเตอร์เครื่องต่างๆ ของอินเทอร์เน็ตเข้าไว้ด้วยกัน มีการแสดงข้อมูลเป็นหน้าเอกสาร (Webpage) ที่สามารถลิงก์เชื่อมต่อไปยังหน้าเอกสารอื่นที่อยู่ห่างไกลกันได้ ในหน้าเอกสารหนึ่ง ๆ จะประกอบด้วยข้อความ รูปภาพ เสียง ภาพเคลื่อนไหว ผู้ใช้สามารถอ่าน ข้อมูล ดาวน์โหลดไฟล์ ดูหนัง ฟังเพลง เล่นเกม ติดต่อสื่อสารกันได้ ทำให้เป็นสื่อที่ได้รับความสนใจอย่างแพร่หลาย และเติบโตอย่างรวดเร็ว ผู้ใช้สามารถใช้บริการเวิลด์ไวด์เว็บได้โดยใช้โปรแกรมซึ่งเรียกว่าเว็บเบราว์เซอร์

3.1 ส่วนประกอบที่สำคัญของเวิลด์ไวด์เว็บ มีดังนี้

1. เว็บเซิร์ฟเวอร์ (Web Server)

โปรแกรมที่รับและประมวลผลข้อมูลที่ร้องขอจากผู้ใช้บริการอินเทอร์เน็ตผ่านทางเว็บเบราว์เซอร์ เมื่อประมวลผลเสร็จแล้ว ข้อมูลก็จะถูกส่งกลับไปแสดงให้ผู้ใช้ผ่านทางเว็บเบราว์เซอร์นั่นเอง ตัวอย่างของเว็บเซิร์ฟเวอร์ ได้แก่ Apache, Personal Web Server, OmniHTTPd, Internet Information Server เป็นต้น

2. เว็บเบราว์เซอร์ (Web Browser)

โปรแกรมประยุกต์ที่ทำงานอยู่บนเครื่องผู้ใช้สำหรับอ่านข้อมูลที่อยู่ในเว็บไซต์ต่าง ๆ และแปลไฟล์เอกสารภาษา HTML เป็นข้อมูลที่แสดงบนหน้าจอ

3. เว็บไซต์ (Web Site)

เป็นแหล่งที่รวบรวมหน้าเว็บเพจต่างๆ หน้าขององค์กรแห่งนั้นไว้ ซึ่งเจ้าของจะเป็นผู้ดูแลและปรับปรุงข้อมูลเอง โดยเจ้าของเว็บไซต์ดังกล่าวอาจจะเป็นองค์กรของรัฐหรือเอกชนหรือเว็บไซต์ส่วนบุคคลก็ได้

4. พร็อกซีเซิร์ฟเวอร์ (Proxy Server)

เป็นเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นที่พักข้อมูลชั่วคราวเพื่อให้เครื่องไคลเอนต์ดาวน์โหลดข้อมูลจากพร็อกซีเซิร์ฟเวอร์โดยไม่จำเป็นต้องต่อเข้ากับเครือข่ายอินเทอร์เน็ต เพื่อประโยชน์ของเครื่องไคลเอนต์ที่จะดาวน์โหลดข้อมูลได้รวดเร็วยิ่งขึ้นหรือเพื่อจุดประสงค์ในการกั้นกรองข้อมูลที่จะผ่านเข้าสู่เครือข่ายหรือออกจากระบบ

5. ไอพีแอดเดรส (IP Address)

เนื่องจากการติดต่อสื่อสารในระบบเครือข่ายแบบอินเทอร์เน็ตใช้งานโปรโตคอลมาตรฐานที่เรียกว่า TCP/IP ซึ่งต้องกำหนดหมายเลขประจำตัวให้กับคอมพิวเตอร์ที่เชื่อมต่อกันนั้นให้แตกต่างกันออกไป หมายเลขประจำตัวที่ว่านี้เรียกว่า IP Address หรือหมายเลขไอพี ซึ่งหมายเลข

ไอพีนี้ประกอบด้วยตัวเลข 4 ชุด แต่ละชุดจะมีขนาด 8 บิต หรือมีค่าตั้งแต่ 0 ถึง 255 ตัวเลขแต่ละชุดจะคั่นด้วยเครื่องหมายจุด เช่น 161.246.10.21

6. โดเมนเนมเซิร์ฟเวอร์ (Domain Name Server หรือ DNS)

เครื่องคอมพิวเตอร์ที่ทำหน้าที่เก็บและค้นหาฐานข้อมูลซึ่งจะเก็บชื่อโดเมนเนม (Domain Name System) และหมายเลขไอพีแอดเดรส (IP Address) ของชื่อโดเมนนั้น เมื่อเครื่องไคลเอนต์เชื่อมต่อไปยังโฮสต์บนอินเทอร์เน็ตด้วยการกำหนดชื่อโดเมน โดเมนเนมเซิร์ฟเวอร์จะค้นหาหมายเลขไอพีแอดเดรสให้ว่าโฮสต์นั้นอยู่ที่ใดหรือถ้าไม่มีมันจะบอกให้ไปค้นหาหมายเลขไอพีแอดเดรสที่โดเมนเนมเซิร์ฟเวอร์เครื่องอื่น

7. ยูอาร์แอล (Uniform Resource Locator : URL)

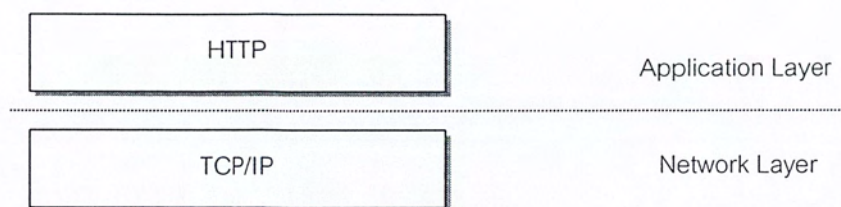
ตำแหน่งที่อยู่ของเว็บไซต์ ซึ่งทุกเว็บไซต์จะมีที่อยู่เป็นของตัวเองโดยเฉพาะ ถ้าผู้ใช้ยูอาร์แอลก็จะสามารถเข้าเว็บไซต์ได้อย่างรวดเร็ว โดยไม่ต้องเสียเวลาในการค้นหา โดยยูอาร์แอลจะมีส่วนประกอบหลักๆ 3 ส่วน ได้แก่ โพรโทคอลที่ใช้ แอดเดรสที่ต้องการติดต่อด้วยและพาท (Path) ของข้อมูลที่ต้องการ เช่น <http://www.thaimaster.com/cutjoke.html>

3.2 หลักการทำงานของเว็ลด์ไวด์เว็บ

เว็ลด์ไวด์เว็บมีหลักการทำงานโดยอาศัยโพรโทคอล HTTP ซึ่งสนับสนุนการทำงานแบบไคลเอนต์/เซิร์ฟเวอร์ (Client/Server) โพรโทคอล HTTP นี้ปัจจุบันยังผูกติดตัวเองกับการทำงานบนระบบเครือข่ายแบบทีซีพี/ไอพี (TCP/IP) ซึ่งเป็นโพรโทคอลพื้นฐานของอินเทอร์เน็ต ตัวเครื่องและคอมพิวเตอร์เป็นเว็บเบราว์เซอร์ (Web Browser) จะทำการเรียกใช้ข้อมูลข่าวสารและบริการจากผู้ให้บริการที่เชื่อมต่ออยู่ในอินเทอร์เน็ต ตัวเครื่องคอมพิวเตอร์และตัวเว็บเบราว์เซอร์จะทำงานเป็นเว็บไคลเอนต์ (Web Client) ส่วนเครื่องคอมพิวเตอร์ที่ให้บริการข้อมูลจะทำหน้าที่เป็นเว็บเซิร์ฟเวอร์ (Web Server)

การทำงานของเว็ลด์ไวด์เว็บนั้นจะเริ่มจากเครื่องที่ขอใช้บริการ (Web Client) ส่งคำร้องขอไปยังเว็บเซิร์ฟเวอร์ผ่านทางเว็บเบราว์เซอร์แล้วก็รอรับการตอบสนอง จากนั้นเว็บเซิร์ฟเวอร์จะจับคู่ระหว่างยูอาร์แอลที่ได้รับการร้องขอกับวัตถุ (Object) และข้อมูลที่ยูอาร์แอลนั้นอ้างถึง ซึ่งวัตถุที่อ้างนี้อาจจะอยู่ในรูปแบบภาษา HTML โปรแกรม หรือเป็นการร้องขอข้อมูลจากฐานข้อมูลก็ได้ แล้วส่งข้อมูลต่างๆ ไปยังเครื่องที่ขอใช้บริการ หรือถ้ามีความผิดพลาดเกิดขึ้นก็จะส่งข้อความที่แสดงความผิดพลาดนั้นไปยังเครื่องที่ขอใช้บริการด้วยโดยผ่านทางเว็บเบราว์เซอร์

ในปัจจุบันโปรแกรมที่ทำหน้าที่เป็นเว็บเบราว์เซอร์มีอยู่มากมายหลายโปรแกรม แต่ที่นิยมใช้กันอย่างแพร่หลายก็คือ IE (Internet Explorer) และ Netscape Navigator



รูปที่ 3-1 เวิลด์ไวด์เว็บทำงานโดยใช้โปรโตคอล HTTP ใน Application Layer ที่ทำงานอยู่เหนือ
โปรโตคอล TCP/IP ใน Network Layer



บทที่ 4

การเข้ารหัสและถอดรหัส

ในปัจจุบันมีการใช้งานระบบเครือข่ายอย่างแพร่หลาย ซึ่งข้อมูลข่าวสารเหล่านี้มีความสำคัญมากในการดำเนินงานขององค์กรทั้งของรัฐหรือของเอกชน ดังนั้นสิ่งสำคัญที่ต้องปฏิบัติคือ ป้องกันที่ตัวข้อมูลเหล่านี้ให้ปลอดภัยและถูกต้องอยู่ตลอดเวลา แต่ว่าข้อมูลอาจถูกเจาะจากผู้ไม่พึงประสงค์หรือไม่มีสิทธิในการเข้าอ่านข้อมูลได้และอาจทำให้เกิดผลเสียต่อเจ้าของข้อมูลได้ ถึงแม้ว่าระบบคอมพิวเตอร์ในปัจจุบันจะมีระบบรักษาความปลอดภัยในขั้นพื้นฐานอยู่แล้ว แต่ในกรณีที่ข้อมูลมีความสำคัญมากๆ ก็จำเป็นต้องมีมาตรการรักษาความปลอดภัยและการป้องกันการเข้าถึงข้อมูลของผู้ไม่มีสิทธิ ซึ่งกรณีนี้จะเกี่ยวข้องกับทั้งผู้ส่งและผู้รับข้อมูล และการกระทำอันใดที่เป็นการป้องกันระบบจากผู้ไม่เกี่ยวข้องนั้นย่อมมีผลโดยตรงกับรูปแบบการทำงานปกติ ดังนั้นจึงเกิดความคิดว่าถ้าให้การทำงานสามารถรับส่งข้อมูลกันได้อย่างปกติ แต่ทำให้เกิดความปลอดภัยในการรับส่งข้อมูลจากผู้ที่มีสิทธิเท่านั้น ส่วนผู้ที่ไม่เกี่ยวข้องไม่สามารถเข้าใจข้อมูลที่รับส่งจึงเป็นทางออกที่เหมาะสมที่สุด

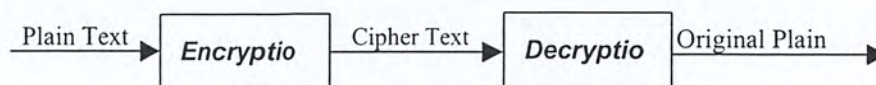
วิธีการที่นิยมนำมาใช้ในการปกป้องข้อมูลและมีประสิทธิภาพก็คือ การนำวิธีการเข้ารหัสและถอดรหัสข้อมูล (Cryptography) มาใช้เพื่อสร้างความปลอดภัยให้กับข้อมูล โดยสามารถแบ่งได้เป็น 2 ลักษณะ คือ

- การเข้ารหัสและถอดรหัสแบบคีย์เหมือน (Symmetric Cryptography)
- การเข้ารหัสและถอดรหัสแบบคีย์ต่าง (Asymmetric Cryptography)

4.1 การเข้ารหัสและถอดรหัสแบบคีย์เหมือน (Symmetric Encryption)

วิธีการเข้ารหัสโดยใช้กุญแจตัวเดียวในการเข้ารหัสและถอดรหัส คือระบบการเข้ารหัสลับและถอดรหัสด้วยกุญแจ (Secret Key) เพียงตัวเดียว หมายความว่ากุญแจที่ใช้ในการเข้ารหัสและถอดรหัสคือตัวเดียวกัน

สิ่งที่ควรสนใจสำหรับวิธีการข้างต้น คือทั้งฝั่งรับและฝั่งส่งต้องรู้คีย์ตัวเดียวกัน ถ้ามีการส่งคีย์ข้ามเครือข่ายไปด้วยจะทำอย่างไรให้คีย์หรือกุญแจที่ใช้นั้นปลอดภัย การใช้งานนั้นรายละเอียดของกุญแจควรมีการปรับปรุงหรือเปลี่ยนบ่อยๆ ตัวอย่างของการเข้ารหัสด้วยคีย์เดียว คือ อัลกอริทึมของเดส (DES Algorithm)



รูปที่ 4-1 แสดงการเข้ารหัสและถอดรหัส

4.2 การเข้ารหัสและถอดรหัสแบบคีย์ต่าง (Asymmetric Encryption)

หลักการของพับลิคคีย์ได้ถูกคิดขึ้นมาเพื่อที่จะแก้ไขปัญหาที่เกิดขึ้นในการเข้ารหัสแบบเดิมที่ได้กล่าวมาก่อนหน้านี้ซึ่งปัญหาที่พบมีอยู่ 2 อย่างคือ

ปัญหาแรก คือ เกี่ยวกับการจัดสรรคีย์ในการการเข้ารหัสและถอดรหัสแบบธรรมดาที่ไม่ใช้ฟังก์ชันคีย์ จะมีคีย์เพียงตัวเดียวซึ่งทั้งฝ่ายรับและฝ่ายส่งจะมีคีย์นี้ไว้ในครอบครองซึ่งดูจะไม่เป็นส่วนตัว คีย์สามารถที่จะรั่วไหลไปสู่ภายนอกได้ง่าย

ปัญหาที่สอง คือ เมื่อมีคนรู้คีย์แล้วนอกจากจะสามารถทำการถอดรหัสข้อมูลของเราได้แล้ว ยังสามารถที่จะทำการเข้ารหัสข้อมูลของเขาเองเมื่อรู้อัลกอริทึมและคีย์แล้วนำข้อมูลนั้นมาแทรกเข้ากับข้อมูลจริงๆ และจะถูกส่งไปสู่ผู้รับโดยไม่รู้ตัว ข้อมูลที่ถูกแทรกเข้ามาอาจเป็นโปรแกรมที่ไม่หวังดีก็ได้

เนื่องจากปัญหาทั้งสองนี้ Diffie และ Hellman ได้สร้างฟังก์ชันคีย์ขึ้นมาเพื่อที่จะแก้ปัญหาเหล่านี้
วิธีการเข้ารหัสและถอดรหัสโดยใช้กุญแจสาธารณะ (Public Key Cryptography)

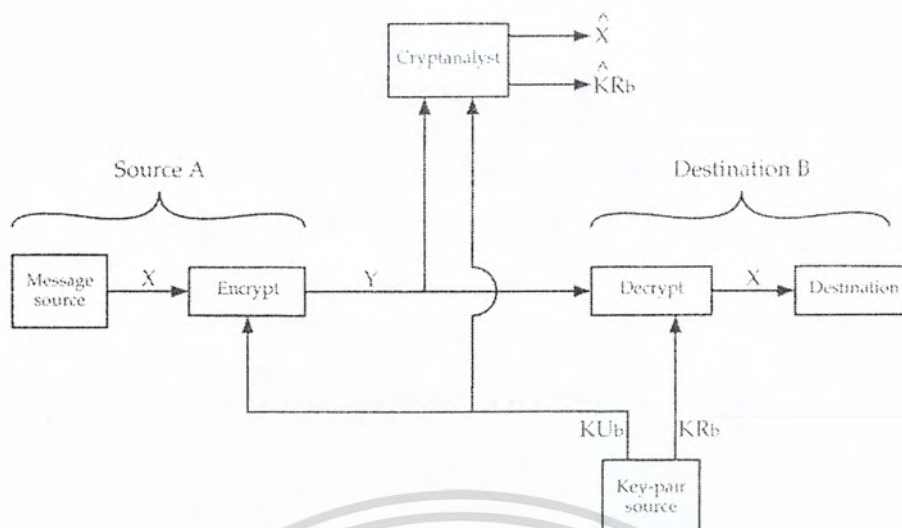
จะมีลักษณะสำคัญดังนี้คือ

- 1. แต่ละเอนด์ซิสเต็มในเครือข่ายจะสร้างคีย์มาเป็นคู่เพื่อที่จะทำการเข้ารหัสและถอดรหัส
- 2. คีย์ทั้งสองคีย์จะถูกเรียกว่า ฟังก์ชันคีย์ (Public Key) และไพรเวตคีย์ (Private Key)
- 3. ถ้า A ต้องการที่จะส่งข้อมูลถึง B B จะต้องส่งฟังก์ชันคีย์มาให้ A เพื่อใช้ในการเข้ารหัส จากนั้นก็จะส่งข้อมูลไปยัง B
- 4. เมื่อ B ได้รับข้อมูลก็จะทำการถอดรหัสโดยใช้ไพรเวตคีย์ของ B เองซึ่งมีเพียง B เท่านั้นที่รู้

การเข้ารหัสและถอดรหัสแบบคีย์เหมือน	การเข้ารหัสและถอดรหัสแบบคีย์ต่าง
1. ใช้ฟังก์ชันคีย์และคีย์เดียวกันในการเข้ารหัสและการถอดรหัส	1. ใช้ฟังก์ชันคีย์เดียวกันในการเข้ารหัสและการถอดรหัสแต่จะใช้คีย์คนละตัวกัน
2. ผู้รับและผู้ส่งจะต้องใช้ฟังก์ชันคีย์และคีย์ร่วมกัน	2. ผู้รับและผู้ส่งจะต้องมีคีย์ที่เป็นความลับ 1 คีย์ คือ ไพรเวตคีย์
3. คีย์จะเป็นตัวเก็บความลับ	3. มีเพียงคีย์เดียวในจำนวน 2 คีย์ที่เป็นตัวเก็บความลับ
4. ความรู้เฉพาะฟังก์ชันคีย์และไคเฟอร์เท็กซ์ไม่เพียงพอที่จะใช้หาคีย์	4. ความรู้เรื่องฟังก์ชันคีย์บวกกับคีย์ 1 ตัวบวกกับไคเฟอร์เท็กซ์ ไม่สามารถที่จะนำไปสู่การหาคีย์ต่อไปได้

ตารางที่ 4-1 การเปรียบเทียบการเข้ารหัสและถอดรหัสแบบคีย์เหมือน กับการเข้ารหัสและถอดรหัสแบบคีย์ต่าง

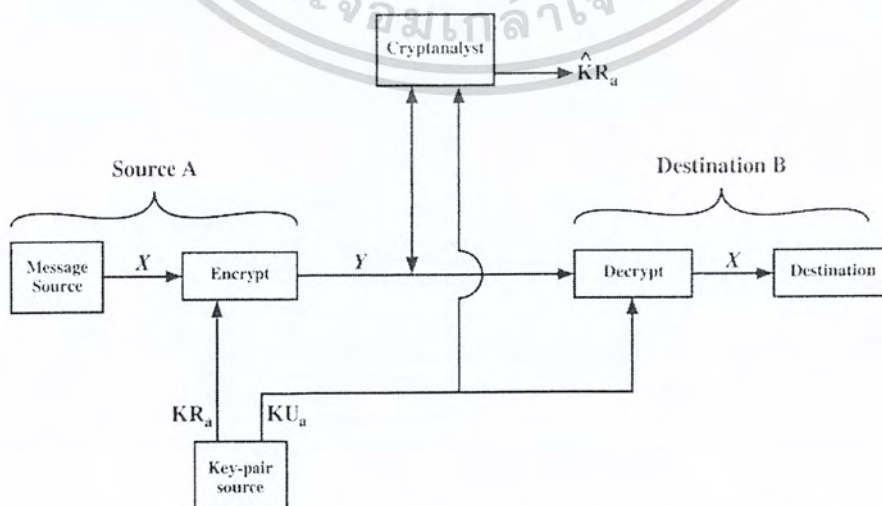
จากลักษณะที่สำคัญของการเข้ารหัสและถอดรหัสโดยใช้กุญแจสาธารณะที่กล่าวมานั้น สามารถนำมาออกแบบโครงสร้างที่จะนำไปสู่การเข้ารหัสและถอดรหัส ซึ่งจะกล่าวใน 3 ลักษณะดังนี้



รูปที่ 4-2 การเข้ารหัสและถอดรหัสโดยใช้กุญแจสาธารณะ: ปกปิดเป็นความลับ

4.2.1 การเข้ารหัสและถอดรหัสโดยใช้กุญแจสาธารณะ: ปกปิดเป็นความลับ

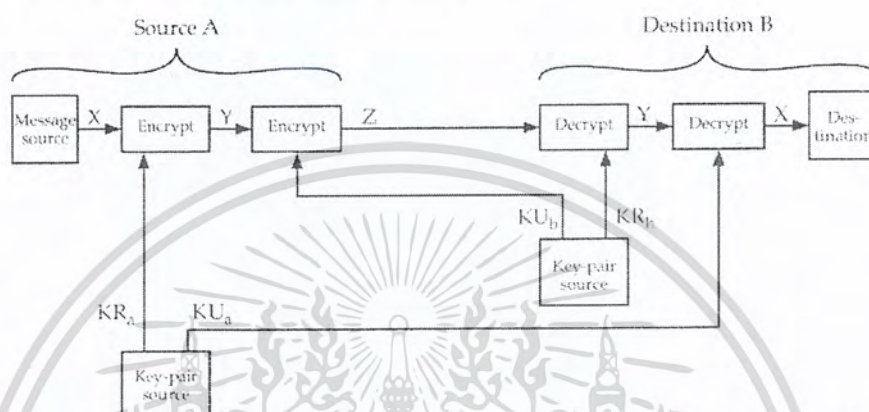
วิธีการนี้แสดงให้เห็นถึงการเก็บเป็นความลับของไพรเวตคีย์ แสดงดังรูปที่ 4-2 สมมติว่า A ต้องการส่งข้อมูลไปยัง B ข้อมูลจะถูกเข้ารหัสโดยใช้พับลิคคีย์ของ B ซึ่ง B จะส่งมาให้เมื่อทำการเข้ารหัสเสร็จ A ก็จะส่งข้อมูลไปยัง B จากนั้น B ก็จะทำการถอดรหัสโดยใช้ไพรเวตคีย์ของ B เอง ซึ่งจะเห็นได้ว่าไพรเวตคีย์ของ B ไม่มีทางที่จะแพร่กระจายออกไปสู่ภายนอกได้ แต่ถ้าเป็นนักวิเคราะห์การเข้ารหัส (Cryptanalyst) สามารถที่จะเข้าไปเอาพับลิคคีย์ออกมาได้ในตอนที่ B ส่งมาให้ A ก็จะรู้อัลกอริทึมไซเฟอร์เท็กซ์และสามารถที่จะนำเอาทั้งหมดนี้มาทำการถอดรหัสโดยการเอาไพรเวตคีย์จนกระทั่งข้อความที่ถูกแปลออกมาอ่านรู้เรื่อง นอกจากนี้นักวิเคราะห์การเข้ารหัสสามารถที่ทำการเข้ารหัสข้อมูลของตัวเองโดยใช้อัลกอริทึมและคีย์ที่หามาได้และทำการแทรกข้อมูลนั้นไปพร้อมกับข้อมูลจริงๆ ได้ ข้อมูลที่แทรกอาจจะเป็นอันตรายต่อระบบก็ได้



รูปที่ 4-3 การเข้ารหัสและถอดรหัสโดยใช้กุญแจสาธารณะ: การพิสูจน์ตน

4.2.2 การเข้ารหัสและถอดรหัสโดยใช้กุญแจสาธารณะ: การพิสูจน์คน

วิธีที่จะแก้ปัญหาในข้อแรกก็คือไม่ให้นักวิเคราะห์การเข้ารหัสรู้ถึงคีย์ที่ใช้ในการเข้ารหัส และจะส่งกุญแจสาธารณะไปยังปลายทางเพื่อที่จะใช้ทำการถอดรหัส แสดงดังรูปที่ 4-3 ซึ่งก็ทำให้ไม่สามารถที่จะแทรกข้อมูลเข้าไปได้เพราะรู้แต่อัลกอริทึมแต่ไม่รู้ถึงคีย์ที่ใช้ในการเข้ารหัส แต่ข้อเสียก็คือกุญแจสาธารณะที่ถูกส่งไปยังปลายทางนักวิเคราะห์การเข้ารหัสสามารถที่จะล่วงรู้ได้ และเมื่อรู้อัลกอริทึมและไซเฟอร์เท็กซ์ก็สามารถนำไปสู่การหาไพรเวตคีย์ได้ซึ่งไม่ปลอดภัย



รูปที่ 4-4 การเข้ารหัสและถอดรหัสโดยใช้กุญแจสาธารณะ: ปกปิดเป็นความลับและการพิสูจน์คน

4.2.3 การเข้ารหัสและถอดรหัสโดยใช้กุญแจสาธารณะ: ปกปิดเป็นความลับและการพิสูจน์คน

วิธีนี้จะเป็นการรวมกันระหว่างวิธีที่ 1 และ 2 เพื่อที่จะนำข้อดีของทั้งสองมาและกำจัดข้อเสียของทั้งสองแบบได้เช่นกัน แสดงในรูปที่ 4-4 แต่วิธีนี้ก็มีข้อเสียคือจะต้องมีการเข้ารหัสถึง 2 ครั้ง และเมื่อเวลาจะทำการถอดรหัสก็ต้องทำถึง 2 ครั้งเช่นกันทำให้ต้องใช้เวลาในส่วนนี้เพิ่มขึ้น

4.3 การแกะรหัส (Cryptanalysis)

ในระบบของการเข้ารหัสใดๆ จะต้องพิจารณาถึงความเป็นไปได้ที่จะมีการแกะรหัสที่ได้เข้าไปออกมา ว่ามีความยากง่ายแค่ไหน ก่อนจะนำไปใช้งาน โดยการแกะรหัสก็เช่นเดียวกับการเข้ารหัส คือมีหลายวิธี โดยขึ้นอยู่กับรูปแบบของการเข้ารหัสว่าเป็นแบบใด และมีข้อมูลให้ใช้มากน้อยแค่ไหน

โดยทั่วไปในการออกแบบอัลกอริทึมในการเข้ารหัสนั้น จะถือว่าประสบความสำเร็จในการรักษาความปลอดภัย เมื่อรหัสที่สร้างขึ้นมาโดยอัลกอริทึมนั้น สามารถบรรลุข้อกำหนดใดข้อกำหนดหนึ่งใน 2 ข้อกำหนดต่อไปนี้ คือ

- ค่าใช้จ่ายที่ต้องใช้ในการถอดรหัสจะต้องมากกว่ามูลค่าของข้อมูลที่เข้ารหัส
- เวลาที่ใช้ในการถอดรหัส จะต้องมากกว่าอายุการใช้งานของข้อมูลที่เข้ารหัส

ทั้งนี้เนื่องจากความต้องการข้อมูลที่เข้ารหัสก็เนื่องจากความต้องการนำข้อมูลนั้นไปใช้ประโยชน์ แต่หากค่าใช้จ่ายที่ต้องใช้ในการถอดรหัส มีมูลค่ามากกว่ามูลค่าตัวเองแล้ว ก็ไม่มีความคุ้มค่าในการถอดรหัส อย่างไรก็ตามการประเมินมูลค่าของข้อมูลนั้นมักพิจารณาได้ยาก ดังนั้นโดยส่วนใหญ่จึงมักประเมินความปลอดภัยของอัลกอริทึมเอาไว้ที่ข้อหลังมากกว่า เพราะเป็นสิ่งที่พิจารณาได้ง่าย ดังนั้นหากสมมุติว่า

อัลกอริทึมที่ออกแบบไม่มีจุดอ่อน ซึ่งหมายความว่า การถอดรหัสจะต้องใช้วิธีที่เรียกว่า Brute-Force เพียงอย่างเดียว

วิธีการที่เรียกว่า Brute-Force นั้นเป็นวิธีการที่ลองเดาคีย์ในทุกๆ ค่าที่เป็นไปได้ ซึ่งหมายความว่า จะต้องสามารถแกะได้อย่างแน่นอน สมมุติว่าโดยเฉลี่ยจะสามารถแกะคีย์ออกได้ เมื่อได้ลองคีย์ที่เป็นไปได้เท่ากับจำนวนครึ่งหนึ่งของคีย์ทั้งหมด โดยตารางที่ 4.1 จะแสดงว่า จะต้องใช้เวลาเท่าไร ในการแกะคีย์ที่มีความยาวต่างๆ กัน โดยสมมุติว่าในการแกะรหัส 1 ครั้ง จะใช้เวลาเท่ากับ 1 ไมโครวินาที ซึ่งถือว่าเป็นความเร็วที่ประมาณได้กับเครื่องคอมพิวเตอร์ระดับกลางในปัจจุบัน โดยจากตารางจะเห็นได้ว่าความยาวคีย์ที่เริ่มมีความปลอดภัย คือที่ 56 บิต แต่หากใช้คอมพิวเตอร์ความเร็วสูง เช่น เครื่องซูเปอร์คอมพิวเตอร์แบบ MPP ซึ่งสมมุติว่ามีความเร็วในการทำงานเท่ากับการแกะรหัส 1 ล้านครั้ง จะเห็นว่าความยาวของคีย์ที่เริ่มปลอดภัย คือ 128 บิต

Key Size(bits)	Number of Alternative Keys	Time Required at 1 Decryption/ μ s	Time Require at 10^6 Decrypt/ μ s
32	4.3×10^9	35.8 Minutes	2.15 Milliseconds
56	7.2×10^{16}	1142 Years	10 Hours
128	3.4×10^{38}	5.4×10^{24} Years	5.4×10^{18} Years
168	3.7×10^{50}	5.9×10^{36} Years	5.9×10^{30} Years

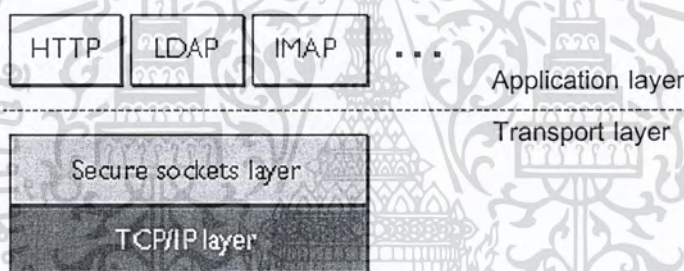
ตารางที่ 4-2 แสดงระยะเวลาที่ใช้ในการแกะรหัส

บทที่ 5

โพรโตคอล SSL

SSL (Secure Sockets Layer) เป็นโพรโตคอลที่ได้รับการพัฒนาโดยบริษัท Netscape และได้รับการเสนอให้เป็นมาตรฐานอุตสาหกรรมโดยองค์การเฉพาะกิจวิศวกรรมอินเทอร์เน็ต (IETF: Internet Engineering Task Force) โดย SSL ถูกออกแบบมาเพื่อใช้งานด้านการรักษาความปลอดภัยในข้อมูล ซึ่งหากเปรียบเทียบกับ 7 เลเยอร์กับ TCP/IP แล้วตัว SSL จะทำงานอยู่ระหว่างเลเยอร์สูง ซึ่งคือชั้นของ Application และ TCP/IP หรือเลเยอร์ที่เกี่ยวข้องกับการรับส่งข้อมูล (Transport)

กล่าวได้คือ การใช้งาน TCP/IP จะใช้ซ็อกเก็ต (Socket) ในการติดต่อสื่อสาร ดังนั้น SSL จึงทำการเข้ารหัสข้อมูลที่ใช้งานทั้งหมดจากเลเยอร์บนแล้วรับส่งผ่านซ็อกเก็ตในเลเยอร์ล่าง นั่นคือ SSL สามารถใช้ได้กับโปรแกรมประยุกต์ทุกตัวที่ทำงานบนโพรโตคอล TCP/IP เช่น ในกรณีของเว็บคือ HTTP ในปัจจุบัน



รูปที่ 5-1 แสดงการทำงานของ SSL ในชั้น Application Layer และ Transport Layer

ในการติดต่อกันระหว่างเว็บเซิร์ฟเวอร์และเว็บไคลเอนต์ผ่านเครือข่ายอินเทอร์เน็ตและเครือข่าย TCP/IP นั้นจะมีขั้นตอนการทำงานหลักๆ ดังนี้

- การพิสูจน์ตนของ SSL ณ เซิร์ฟเวอร์ (SSL server authentication): จะอนุญาตหรือให้ผู้ใช้ (User) สามารถตรวจสอบว่าเซิร์ฟเวอร์ที่ติดต่อเป็นเซิร์ฟเวอร์ที่ผู้ใช้ต้องการติดต่อด้วยโดย 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): โดยเซิร์ฟเวอร์สามารถตรวจสอบผู้ใช้โดยใช้วิธีเดียวกับข้างบน นั่นคือ SSL-enabled server S/W สามารถตรวจสอบว่า certificate และ ID ของผู้ใช้งานถูกต้องและใช้ได้และออกแบบให้โดย CA ซึ่งการตรวจสอบนี้อาจจำเป็น เช่น เซิร์ฟเวอร์ต้องการตรวจสอบว่าเป็นผู้ใช้ที่ตนติดต่อดำรงก่อนที่จะส่งข่าวสารเกี่ยวกับการเงินไปให้ผู้นั้นๆ

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

5.1 องค์ประกอบของ SSL

โพรโทคอล SSL แบ่งเป็น

- SSL record Protocol: กำหนดรูปแบบที่ใช้ในการส่งข้อมูล
- SSL handshake Protocol: จะเกี่ยวข้องกับการใช้ SSL record Protocol ในการแลกเปลี่ยนข้อความ (message) ระหว่าง SSL-enabled server กับ SSL-enabled client ในตอนเริ่มต้นการติดตั้งการเชื่อมต่อแบบ SSL

การแลกเปลี่ยนข่าวสารถูกออกแบบมาเพื่อช่วย

- 1) ตรวจสอบเซิร์ฟเวอร์ที่ไคลเอนต์ติดต่อ
- 2) ช่วยให้ไคลเอนต์และเซิร์ฟเวอร์เลือกอัลกอริทึมที่ใช้ในการเข้ารหัสและถอดรหัสข้อมูลที่สนั่นสนุนการทำงาน
- 3) ตรวจสอบไคลเอนต์ที่เซิร์ฟเวอร์ติดต่อ
- 4) การใช้เทคนิคการเข้ารหัส โดยใช้กุญแจสาธารณะในการสร้าง secret key ที่ใช้ร่วมกัน
- 5) ติดตั้งการติดต่อสื่อสารแบบ SSL

5.2 การเข้ารหัสและถอดรหัสใน SSL (Ciphers Used with SSL)

SSL สนับสนุนอัลกอริทึมที่ใช้ในการเข้ารหัสและถอดรหัสหลายๆ ชนิด ดังนี้

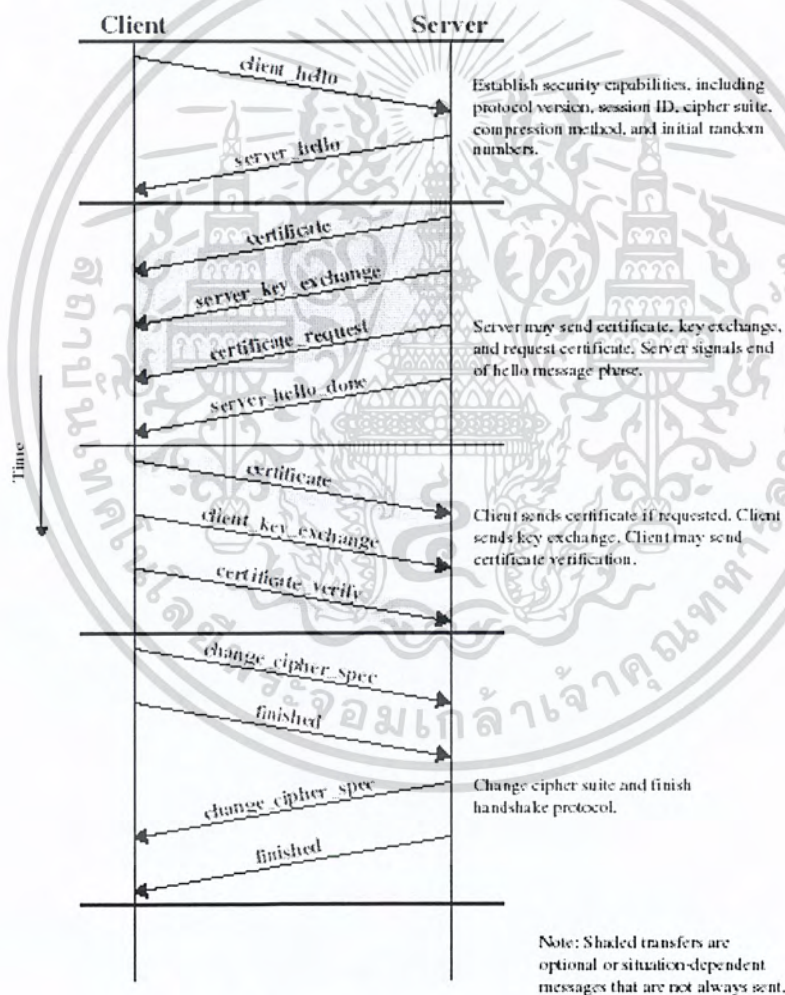
- DES (Data Encryption Standard) เป็นการเข้ารหัสใช้โดยรัฐบาลของประเทศสหรัฐอเมริกา
 - DSA (Digital Signature Algorithm) เป็นส่วนหนึ่งของมาตรฐานของการตรวจสอบลายเซ็นดิจิทัล (Digital Authentication)
 - RSA เป็นอัลกอริทึมที่ใช้ในการเข้ารหัสและถอดรหัสและการยืนยันตัวบุคคล
 - Triple-DES มีการทำงานที่คล้ายกับแบบ DES แต่จะมีการเข้ารหัสถึง 3 ครั้ง
 - KEA (Key Exchange Algorithm)
 - MD5 (Message Digest)
 - RC2/RC4 (Rivest Encryption Ciphers)
 - RSA Key Exchange
- เป็นต้น

5.3 การตกลงเชื่อมต่อใน SSL (SSL Handshake)

โพรโทคอล SSL จะใช้เทคนิคของการเข้ารหัสข้อมูลโดยใช้กุญแจสาธารณะและแบบคีย์เหมือน (Symmetric) ซึ่งการเข้ารหัสแบบคีย์เหมือนมีการเข้ารหัสที่รวดเร็วกว่าการเข้ารหัสแบบกุญแจสาธารณะ แต่การเข้ารหัสแบบกุญแจสาธารณะจะใช้เทคนิคในการตรวจสอบและยืนยันตัวตนบุคคล (Authentication) ที่ดีกว่าแบบคีย์เหมือน ในขั้นตอนการสื่อสารแบบ SSL จะเริ่มต้นด้วยการแลกเปลี่ยนข้อความ (message) ที่เรียกว่า การตกลงการเชื่อมต่อใน SSL โดยในการตกลงการเชื่อมต่อนั้นจะอนุญาตให้เซิร์ฟเวอร์และไคลเอนต์ทำการยืนยันตัวเองแก่ตัวไคลเอนต์โดยใช้เทคนิคของกุญแจสาธารณะ อีกทั้งเซิร์ฟเวอร์และไคลเอนต์ร่วมกันสร้างกุญแจหรือ Session Key ที่ใช้ในการส่งข้อมูลแก่กัน ซึ่งขั้นตอนของการตกลงการเชื่อมต่อในการแลกเปลี่ยนข้อมูล มีดังนี้

1. ไคลเอนต์ส่งหมายเลขเวอร์ชันของ SSL (SSL Version Number) ของไคลเอนต์ / รูปแบบการเข้ารหัส / ถอดรหัส / วิธีการบีบอัดข้อมูลและข้อมูลอื่นๆ ให้แก่เซิร์ฟเวอร์ที่เซิร์ฟเวอร์ต้องการติดต่อกับไคลเอนต์โดยใช้ SSL
2. เซิร์ฟเวอร์ส่งหมายเลขเวอร์ชันของ SSL ของตัวเอง / รูปแบบการติดต่อ / วิธีการบีบอัดข้อมูลและ certificate ไปให้ไคลเอนต์เพื่อยืนยันตัวตนว่าเป็นเซิร์ฟเวอร์ซึ่งตนติดต่อดำเนินการ
3. ไคลเอนต์จะใช้ข้อมูลที่ได้รับมาจากเซิร์ฟเวอร์มาตรวจสอบเพื่อยืนยันว่าเป็นเซิร์ฟเวอร์นั้นจริง ถ้าเซิร์ฟเวอร์นั้นไม่สามารถถูกตรวจสอบได้ผู้ใช้จะได้รับการเตือนว่าเกิดปัญหาขึ้น และแสดงว่าการเข้ารหัส (Encrypted) และการยืนยันตัวตนบุคคล (Authenticated) ไม่สามารถติดตั้งได้
4. ใส่ข้อมูลทั้งหมดในการทำการเชื่อมต่อ จากนั้นไคลเอนต์จะสร้าง “premaster secret” เพื่อสร้างเส้นทางการสื่อสารการติดต่อและทำการเข้ารหัส โดยใช้กุญแจสาธารณะของเซิร์ฟเวอร์ (ได้รับจากเซิร์ฟเวอร์ในข้อ 2) และทำการส่ง premaster secret ที่ถูกทำการเข้ารหัสให้แก่เซิร์ฟเวอร์
5. ในกรณีที่เซิร์ฟเวอร์ต้องการตรวจสอบว่าไคลเอนต์นี้เป็นไคลเอนต์จริงหรือไม่ ไคลเอนต์ก็จะส่ง certificate ของไคลเอนต์พร้อมกับ premaster secret ที่ถูกเข้ารหัสให้แก่เซิร์ฟเวอร์พร้อมๆ กัน
6. เซิร์ฟเวอร์ทำการตรวจสอบ certificate ของไคลเอนต์นั้น ถ้าการตรวจสอบล้มเหลวการสร้างเส้นทางการสื่อสารจะสิ้นสุดลง ถ้าการตรวจสอบถูกต้องเซิร์ฟเวอร์จะใช้ไพรเวตคีย์ของตนทำการถอดรหัส premaster secret และทำการสร้าง master key ขึ้นมา
7. ไคลเอนต์และเซิร์ฟเวอร์จะใช้ master key สร้าง session key ซึ่งเป็น symmetric keys ที่ใช้ในการเข้ารหัสและถอดรหัส ข้อมูลข่าวสารที่ใช้แลกเปลี่ยนในช่วงการติดต่อสื่อสารของ SSL และใช้ในการตรวจสอบความถูกต้อง นั่นคือจะตรวจสอบการเปลี่ยนแปลงของข้อมูล (Data) ในช่วงเวลาที่มันส่งและในช่วงเวลาที่รับข้อมูลผ่านการติดต่อแบบ SSL

8. ไคลเอนต์ส่งข้อความไปบอกแก่เซิร์ฟเวอร์ว่าข้อความถัดไปที่ไคลเอนต์จะส่งไปให้จะถูกเข้ารหัสโดย session key จากนั้นไคลเอนต์จะส่งข้อความ (ที่ถูกเข้ารหัสโดย session key) เพื่อบอกแก่เซิร์ฟเวอร์ว่าในส่วนของไคลเอนต์นั้นทำการตกลงการเชื่อมต่อเสร็จสิ้น
 9. เซิร์ฟเวอร์ส่งข้อความไปบอกแก่ไคลเอนต์ว่าข้อความต่อไปที่จะส่งมาจะเข้ารหัสด้วย session key และเซิร์ฟเวอร์จะทำการส่งข้อความที่เข้ารหัสไว้ (โดย session key) ไปให้แก่ไคลเอนต์เพื่อบอกว่าการตกลงการเชื่อมต่อในส่วนของเซิร์ฟเวอร์เสร็จสิ้นแล้ว
 10. การตกลงการเชื่อมต่อใน SSL เสร็จสิ้นและช่องทางการสื่อสารแบบ SSL ติดตั้งสำเร็จและเริ่มทำงาน ไคลเอนต์และเซิร์ฟเวอร์จะใช้ session keys ในการเข้ารหัสและถอดรหัสข้อมูลที่ไคลเอนต์และเซิร์ฟเวอร์ส่งระหว่างกันและกันและใช้ตรวจสอบความถูกต้อง
- สำหรับโครงสร้างของ Handshake Protocol นั้นแสดงได้ดังรูป 5.2



รูปที่ 5-2 แสดงโครงสร้างของ Handshake Protocol

5.4 การยืนยันตัวเซิร์ฟเวอร์ (Server Authentication)

เป็นการที่ไคลเอนต์ทำการตรวจสอบเซิร์ฟเวอร์โดยให้เซิร์ฟเวอร์ทำการยืนยันตัวเองเหมือนกันกับในขั้นตอนที่ 2 ในการตกลงการเชื่อมต่อใน SSL โดยเซิร์ฟเวอร์ทำการส่ง certificate ของตัวเองไปให้แก่ไคลเอนต์และเมื่อได้รับข้อมูลเหล่านั้นก็จะทำการตรวจสอบดังในขั้นตอนที่ 3 ของการทำการตกลงการเชื่อมต่อใน SSL

ในการตรวจสอบการยืนยันที่ฝั่งเซิร์ฟเวอร์มีขั้นตอนดังนี้

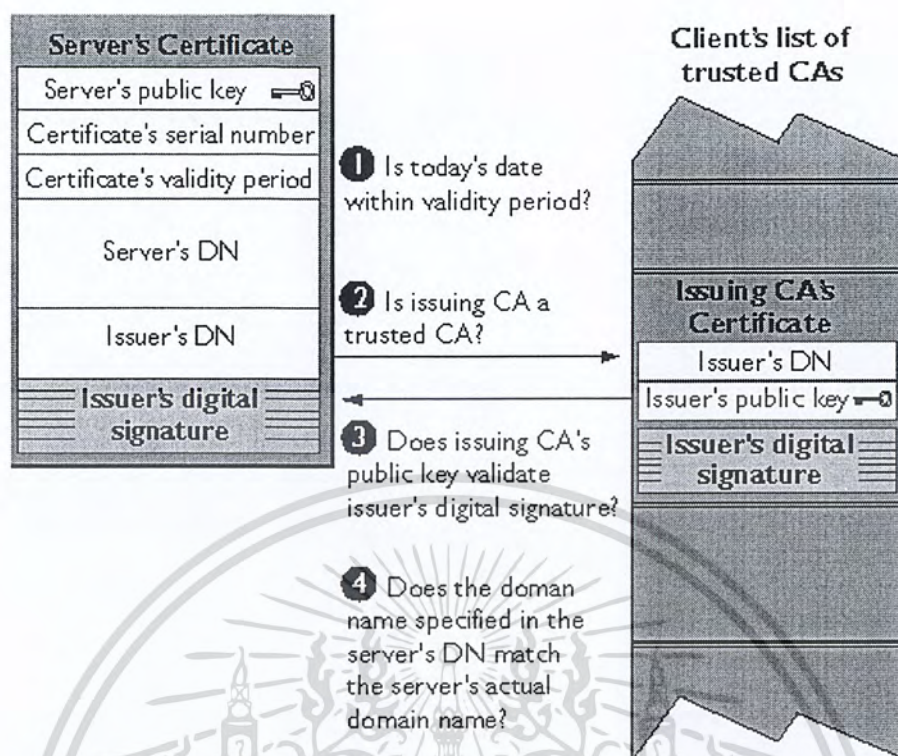
1. ไคลเอนต์จะทำการตรวจสอบถึงช่วงเวลาที่สามารถใช้งานได้ของ certificate ของเซิร์ฟเวอร์ซึ่งถ้าการตรวจสอบได้ว่าช่วงวันและเวลาในการทำงานของ certificate ของเซิร์ฟเวอร์ยังสามารถใช้ได้ก็จะทำการตรวจสอบในขั้นตอนต่อไป
2. ไคลเอนต์จะทำการตรวจสอบองค์ประกอบที่ทำการออกใบ certificate ของเซิร์ฟเวอร์ว่าเป็นองค์กรที่เชื่อถือได้หรือไม่ซึ่งในกรณีที่เป็้องค์กรที่เชื่อถือได้ก็จะทำการตรวจสอบต่อไปในขั้นที่ 3 แต่ถ้าเป็นองค์กรที่ไม่สามารถเชื่อถือได้ก็จะไม่ทำการยืนยันในขั้นต่อไป
3. ไคลเอนต์จะใช้กุญแจสาธารณะที่ได้จากใบรับรองของ CA (มีอยู่ใน list ของ CA ของไคลเอนต์) เพื่อตรวจสอบลายเซ็นดิจิทัล (Digital Signature) ของ CA บน certificate ของเซิร์ฟเวอร์ ในกรณีที่ข้อมูลในลายเซ็นดิจิทัลได้รับการเปลี่ยนแปลงหรือกุญแจสาธารณะของ CA บน certificate ไม่สอดคล้องกับไพรเวตคีย์ที่ใช้โดย CA ในการสร้าง certificate ไคลเอนต์ก็จะไม่รับรองการยืนยันของเซิร์ฟเวอร์
4. เป็นขั้นตอนที่ทำการยืนยันว่าเซิร์ฟเวอร์ที่ไคลเอนต์กำลังทำการติดต่อด้วยนั้นมีโดเมนเนม (Domain Name) ภายในเครือข่ายตรงกันกับโดเมนเนมในใบ certificate หรือไม่ ซึ่งถ้าตรงแสดงว่าเป็นเซิร์ฟเวอร์ที่สามารถเชื่อถือได้ ดังรูปที่ 5-3 แสดงขั้นตอนในการตรวจสอบเซิร์ฟเวอร์

5.5 การยืนยันตัวไคลเอนต์ (Client Authentication)

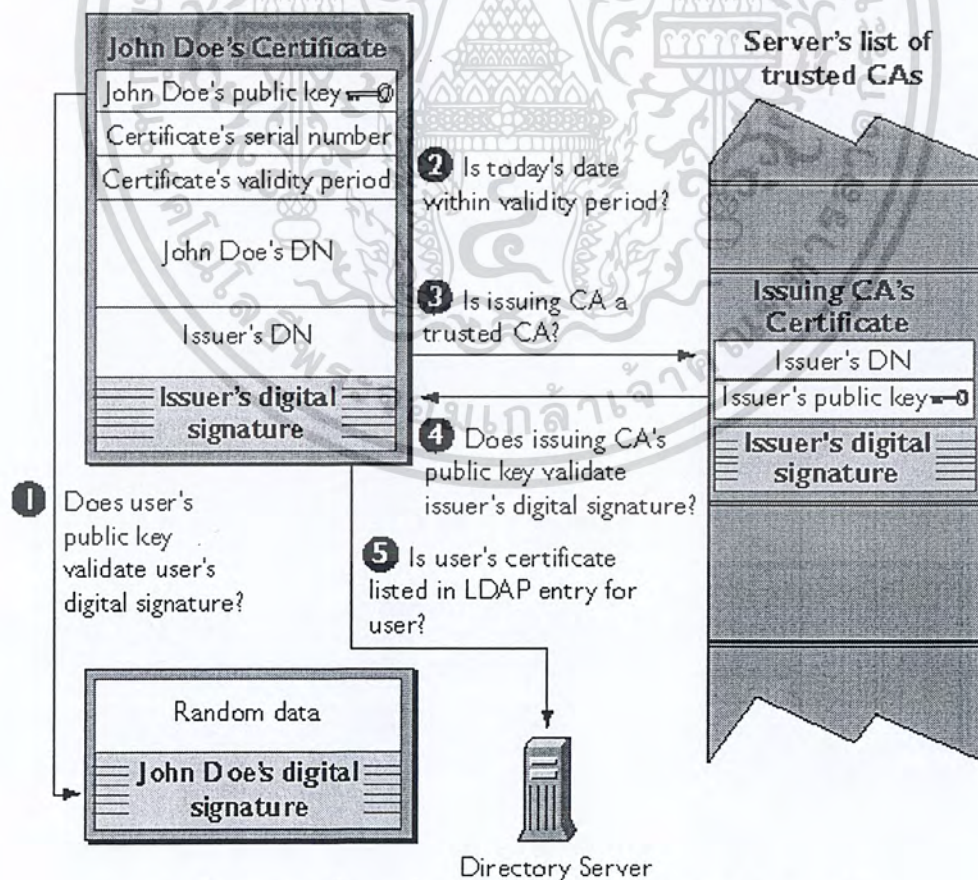
ในกรณีที่เซิร์ฟเวอร์ต้องการตรวจสอบหรือยืนยันจากไคลเอนต์โดยเซิร์ฟเวอร์จะร้องขอไปยังไคลเอนต์ (ดูในขั้นตอนที่ 6 ของการตกลงการเชื่อมต่อใน SSL) โดยไคลเอนต์จะทำการส่ง certificate ของตัวเองและส่วนของลายเซ็นดิจิทัลไปให้แก่เซิร์ฟเวอร์ จากนั้นเซิร์ฟเวอร์จะทำการตรวจสอบโดยใช้ลายเซ็นดิจิทัลมาตรวจสอบความถูกต้องของกุญแจสาธารณะใน certificate ที่ไคลเอนต์ส่งมา

โพรโตคอล SSL จะร้องขอให้ไคลเอนต์ทำการสร้างลายเซ็นดิจิทัลโดยใช้วิธี one-way hash โดยใช้ข้อมูลที่สุ่มขึ้นมาในขณะที่ทำการตกลงการเชื่อมต่อโดยรู้กันเฉพาะเซิร์ฟเวอร์กับไคลเอนต์นั้น ส่วนของข้อมูลที่ถูกทำการบีบอัด (Digest) จะถูกเข้ารหัสโดยไพรเวตคีย์ของฝั่งเซิร์ฟเวอร์ที่สอดคล้องกับกุญแจสาธารณะใน certificate ที่ส่งไปให้แก่เซิร์ฟเวอร์

ในการตรวจสอบการยืนยันที่ฝั่งไคลเอนต์ มีขั้นตอนดังนี้



รูปที่ 5-3 แสดงขั้นตอนการทำงานของ การตรวจสอบในส่วนของผู้ให้บริการ



รูปที่ 5-4 แสดงขั้นตอนการทำงานของการตรวจสอบในส่วนของผู้ใช้

- 1 เว็บเซิร์ฟเวอร์จะทำการตรวจสอบลายเซ็นดิจิทัลของเว็บไคลเอนต์ว่าสามารถใช้กับพับลิคคีย์ในใบรับรองดิจิทัลหรือไม่ ถ้าใช้ได้เว็บเซิร์ฟเวอร์จะสร้างการยืนยันพับลิคคีย์นี้เป็นของ John Doe ตามรูปที่ 5-4 ซึ่งตรงกับไพรเวตคีย์ที่ใช้ในการสร้างลายเซ็นดิจิทัลและข้อมูลไม่ได้ถูกดักหรือถูกเปลี่ยนแปลงตั้งแต่เริ่มทำการสร้างลายเซ็นดิจิทัล
- 2 เว็บเซิร์ฟเวอร์จะทำการตรวจสอบช่วงเวลาที่ใช้งานได้ของใบรับรองดิจิทัลของเว็บไคลเอนต์เพื่อตรวจสอบว่าวันและเวลาที่เว็บเซิร์ฟเวอร์ติดต่อกับเว็บไคลเอนต์นี้อยู่ในช่วงเวลาที่กำหนดไว้หรือไม่ ถ้าไม่อยู่ การยืนยันหรือการตรวจสอบจะไม่ทำอีกต่อไป แต่ถ้าอยู่ในช่วงก็จะไปยังขั้นตอนที่ 3
- 3 แต่ละ SSL-enabled ของเว็บเซิร์ฟเวอร์จะมีรายชื่อของใบรับรองดิจิทัลของ CA ที่น่าเชื่อถือ (อยู่ในรูปที่ 5-4 ส่วนที่แรกของเว็บเซิร์ฟเวอร์) โดยรายชื่อนี้จะแสดงใบรับรองดิจิทัลทั้งหมดที่เว็บเซิร์ฟเวอร์สามารถรับรองได้ ซึ่งถ้า DN (Distinguish Name) ของ CA บนใบรับรองดิจิทัลของเว็บไคลเอนต์ตรงกับรายชื่อใบรับรองดิจิทัลของ CA ที่เว็บเซิร์ฟเวอร์ จะถือว่าเชื่อถือได้ แต่ถ้าไม่มีอยู่ในใบรายชื่อ การตรวจสอบการยืนยันของเว็บไคลเอนต์จะถือว่าล้มเหลว
- 4 เว็บเซิร์ฟเวอร์จะใช้พับลิคคีย์ที่ได้จากใบรับรองของ CA (มีอยู่ใน list ของ CA ของเว็บเซิร์ฟเวอร์) เพื่อตรวจสอบลายเซ็นดิจิทัลของ CA บนใบรับรองดิจิทัลของเว็บไคลเอนต์ ในกรณีที่ข้อมูลในลายเซ็นดิจิทัลได้รับการเปลี่ยนแปลงหรือพับลิคคีย์ในใบรับรองดิจิทัลของ CA ไม่ตรงกันหรือไม่สอดคล้องกับไพรเวตคีย์ที่ใช้โดย CA ในการสร้างใบรับรองดิจิทัล เว็บเซิร์ฟเวอร์ก็จะไม่รับรองการยืนยันของเว็บไคลเอนต์
- 5 เป็นอีกหนทางหนึ่งสำหรับผู้ดูแลระบบในการถอนใบรับรองของเว็บไคลเอนต์ถึงแม้ว่าจะผ่านการตรวจสอบในขั้นตอนต่างๆ แล้วก็ตามโดยถ้าใบรับรองดิจิทัลของเว็บไคลเอนต์ในไดเรกทอรี (Directory) ตรงกับใบรับรองดิจิทัลของเว็บไคลเอนต์ในการทำการตกลงเชื่อมต่อใน SSL ก็จะไปยังขั้นตอนที่ 6 แต่ใบรับรองดิจิทัลของเว็บไคลเอนต์ไม่อยู่ในไดเรกทอรี ความน่าเชื่อถือ (Trusted) ของเว็บไคลเอนต์นั้นก็จะน้อยลง
- 6 เว็บเซิร์ฟเวอร์ทำการตรวจสอบว่าทรัพยากร (Resource) ใดหรือส่วนใดที่เว็บไคลเอนต์ได้รับอนุญาตในการเข้าใช้ ตาม ACLs (Access Control Lists) และทำการเชื่อมต่อการสื่อสารในรูปแบบที่เหมาะสม

บทที่ 6

ปัญหาความปลอดภัยที่ฝั่งเซิร์ฟเวอร์

ในโครงงานนี้เว็บเซิร์ฟเวอร์ที่ได้ทำการศึกษาคือ Apache เนื่องจากเป็นเว็บเซิร์ฟเวอร์ที่เป็นที่นิยมใช้กันอย่างแพร่หลายในปัจจุบัน

ในส่วนของการทำงานนั้นเว็บเซิร์ฟเวอร์จะเชื่อมต่อและคอยฟังการติดต่อผ่านทางแอดเดรสเครือข่ายและทางพอร์ต (Port) สำหรับเปิดการติดต่อกับเว็บไคลเอนต์ที่ทำการร้องขอ (Request) เข้ามาและทำการตอบสนองการร้องขอโดยการส่งไฟล์ เอกสารหรืออาจจะเป็นการรันโปรแกรมก็ได้ ขั้นตอนการทำงานจากเว็บไคลเอนต์ไปยังเว็บเซิร์ฟเวอร์มีดังนี้

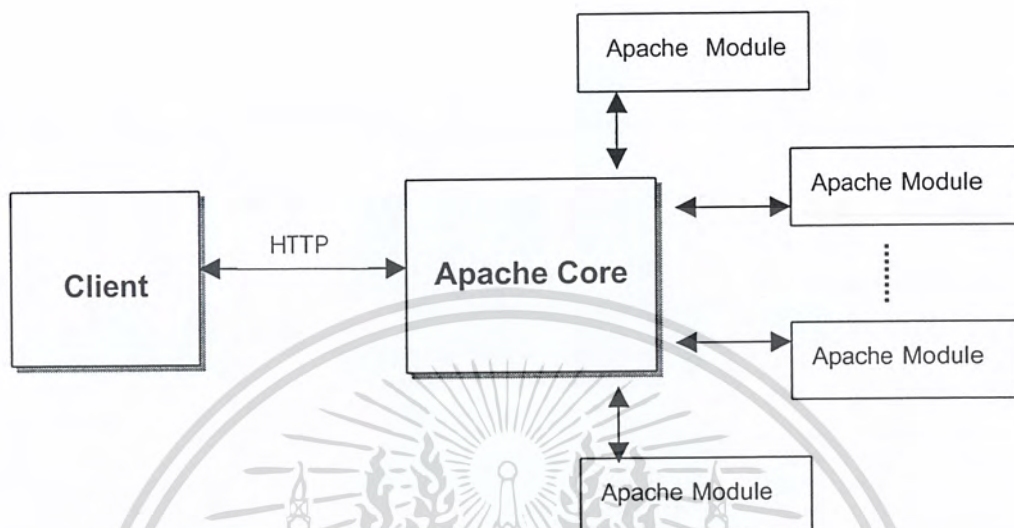
1. เว็บเซิร์ฟเวอร์จะทำการเชื่อมต่อกับเครือข่ายและเปิดพอร์ตการติดต่อมายังที่ TCP/IP แอดเดรส เมื่อมีการร้องขอเข้ามา
2. เมื่อมีการร้องขอมาจากเว็บไคลเอนต์ ซึ่งจะมีการใช้โปรโตคอล HTTP โดยคำร้องขอจะบอกว่าเว็บเซิร์ฟเวอร์จะต้องทำอะไรและมีการใช้โปรโตคอล HTTP เวอร์ชันใด
3. เว็บเซิร์ฟเวอร์จะทำการแปลความหมายหรือดูว่าคำร้องขอมีอะไรบ้าง
4. เว็บเซิร์ฟเวอร์จะเริ่มการทำงานตามคำร้องขอ โดยปกติแล้วคำร้องขอจะประกอบด้วยเอกสาร HTML และไฟล์ที่จะต้องส่งกลับ รวมถึงโปรแกรม CGI ที่ทำการเอ็กซีคิวต์ (Execute) ด้วย
5. เว็บเซิร์ฟเวอร์ตอบกลับโดยส่งเอกสาร HTML ไฟล์และไคเร็กทอรีต่างๆ รวมถึง HTML เพจที่สร้างจาก CGI ด้วย
6. เว็บเซิร์ฟเวอร์จะปิดการติดต่อสื่อสารในเครือข่าย ปิดไฟล์ต่างๆ จนกว่าจะมีการร้องขออีกครั้ง

6.1 Apache

Apache เป็นเว็บเซิร์ฟเวอร์ที่สามารถทำงานบนหลายๆ แพลตฟอร์ม รวมไปถึงบน Unix, Linux และ วินโดวส์ 95/98/Me/NT/2000 ด้วย แต่ไม่ว่าจะเป็นแบบใดก็ตาม Apache จะมีโครงสร้างสถาปัตยกรรมที่เหมือนกัน ซึ่งจะกล่าวในลำดับต่อไป

6.1.1 โครงสร้างและสถาปัตยกรรม

จากรูปที่ 6.1 แสดงโครงสร้างสถาปัตยกรรมของ Apache จะมีส่วนประกอบหลักๆ สองส่วนคือ ส่วนของ Apache Core และส่วน Modules ที่ทำหน้าที่ต่างๆ กันไป โดยปกติแล้ว Apache Core จะมีหน้าที่โดยรวม คือ การรับคำร้องขอเข้ามา (HTTP Request) พร้อมทั้งการจำแนกและติดตามการทำงานตามการร้องขอหรือการตอบกลับ (Reply) คำร้องขอที่เข้ามาต่อเนื่องกันไป ส่วน Modules นั้นจะเป็นหน่วยที่ให้บริการหน้าที่ต่างๆ ตามที่มีการร้องขอเข้ามา



รูปที่ 6-1 แสดงสถาปัตยกรรมคร่าวๆ ของ Apache Webserver

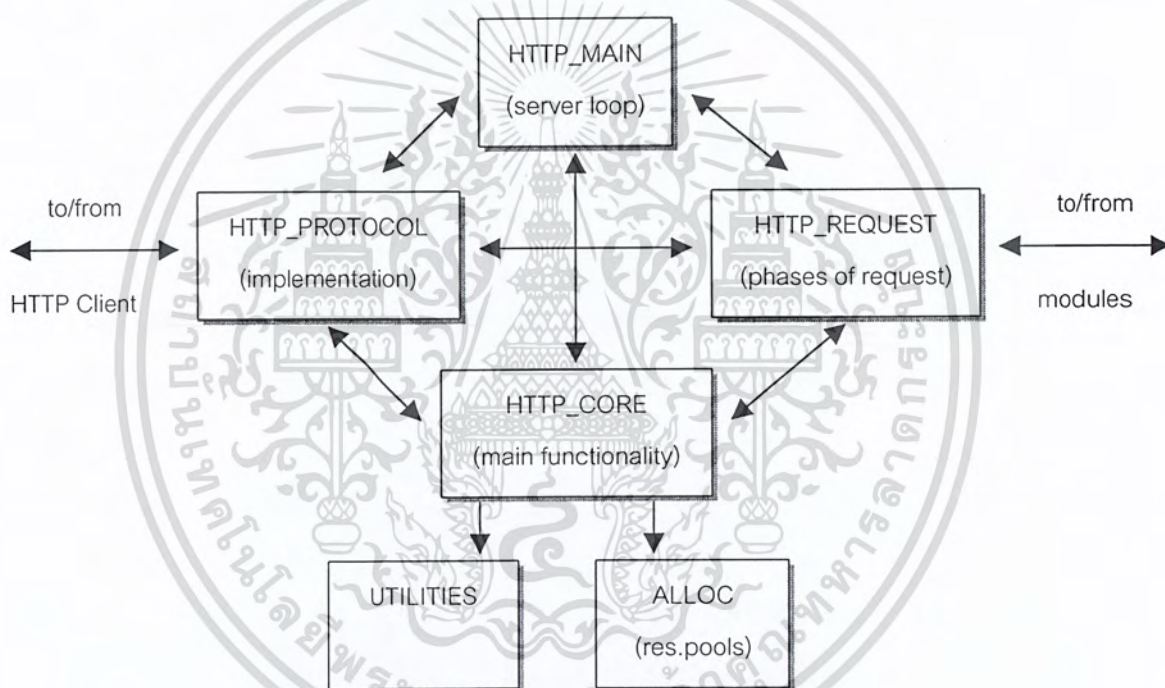
ในส่วนของ Apache Core นั้นจะแบ่งเป็นส่วนประกอบย่อยๆ ดังรูปที่ 6.2 ซึ่งแต่ละส่วนจะมีหน้าที่การทำงานดังต่อไปนี้

- **http_protocol** เป็นส่วนที่ทำหน้าที่ติดต่อสื่อสารโดยตรงกับเว็บไคลเอนต์ ผ่านทางโปรโตคอล HTTP
- **http_main** เป็นส่วนที่ทำหน้าที่เริ่มต้นการทำงานของเซิร์ฟเวอร์ คอยจัดการเกี่ยวกับการเปิดการเชื่อมต่อเครือข่ายเป็นระยะเมื่อมีการร้องขอจากเว็บไคลเอนต์เข้ามา รวมไปถึงอาจมีการสร้างโปรเซสลูกและการจัดตรวจสอบว่าเกิด Timeout หรือไม่
- **http_request** เป็นส่วนที่ทำหน้าที่จัดการเกี่ยวกับลำดับการร้องขอจากไคลเอนต์ พร้อมทั้งส่งลำดับเฟสที่ร้องขอมาไปให้แต่ละโมดูล นอกจากนี้ยังจัดการเกี่ยวกับ error handling
- **http_core** ทำหน้าที่ควบคุมและประสานการทำงานในหน้าที่หลักๆ ทั้งหมด
- **alloc** เป็นส่วนที่ทำหน้าที่ในการจัดสรรและจัดการเกี่ยวกับการใช้ทรัพยากรต่างๆ ภายในระบบ
- **utilities** เป็นส่วนที่ทำหน้าที่จัดการและอ่านไฟล์ต่างๆ ที่เป็นไฟล์ของระบบ (System File) เช่น การอ่านค่าจากไฟล์ configuration และจัดการข้อมูลในไฟล์นั้น นอกจากนี้ยังดำเนินการเกี่ยวกับการทำงานแบบ virtual host ด้วย

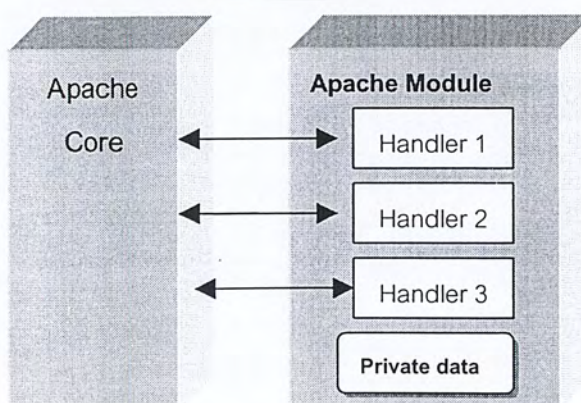
ในส่วนของโมดูลนั้น ภายในโมดูลจะประกอบไปด้วย Handlers ต่างๆ ดังแสดงในรูปที่ 6.3 ซึ่งโมดูลจะเป็นตัวที่กำหนดว่าจะใช้ Handlers ตัวไหน ซึ่ง Handlers จะถูกเรียกใช้เพื่อให้บริการเมื่อเฟสร้องขอเข้ามาและตรงกับหน้าที่การทำงานของ Handler ที่ได้กำหนดไว้ ยกตัวอย่างเช่น เมื่อมีการร้องขอไฟล์

เข้ามา Handler ที่ทำหน้าที่รับผิดชอบเกี่ยวกับไฟล์ ก็จะทำการเปิดไฟล์ อ่านข้อมูลในไฟล์ และทำการส่งไฟล์กลับไปให้ไคลเอนต์ โดยผ่านทาง Apache core ในบางเฟส เช่น การพิสูจน์ตน (Authentication) จะใช้เพียงแค่ Handler เพียงตัวเดียว หรือในเฟสอื่นๆ เช่น Logging จำเป็นต้องใช้ Handler หลายๆ ตัว หรือแม้กระทั่งในเฟสของการแปลง URI (URI Translation) จะมีการใช้ Handler จัดการหลายตัวเรียงตามลำดับการทำงาน เป็นต้น

ในการทำงานด้านอื่นๆ จะเป็นการทำงานผ่านโมดูล โดยเฉพาะอย่างยิ่งหน้าที่ของโมดูลการประยุกต์ (Implement) โอเวอร์ไรด์ (Override) และการทำงานที่ Apache Core ส่งมาให้ โดยปกติโมดูลจะไม่ทำการติดต่อกับโดยตรงแต่จะทำการสื่อสารผ่านทาง Apache Core ซึ่ง Apache Core จะสนับสนุนการเชื่อมต่อแบบไดนามิก (Dynamic Linking) นั่นคือ ปกติแล้วโมดูลจะไม่มีการสร้างการเชื่อมต่อและการเชื่อมต่อใดๆ จนกว่าจะมีการทำงานระหว่างโมดูลเกิดขึ้นผ่านทาง Apache Core



รูปที่ 6-2 สถาปัตยกรรมของ Apache Core



รูปที่ 6-3 สถาปัตยกรรมของ Apache Module

จากรูปที่ 6.3 จะเห็นว่าภายในโมดูลนั้นจะมีส่วนของ Private data อยู่ ซึ่งเป็นส่วนใช้สำหรับ save ข้อมูลที่มีการร้องขอเข้ามาในแต่ละครั้งเก็บไว้นั่นเอง

สำหรับ Apache นั้นจะมีโมดูลมาตรฐานส่วนหนึ่งไว้สำหรับการให้บริการต่างๆ ซึ่งสามารถแบ่งตามหมวดหมู่ได้ดังต่อไปนี้

Environment Creation

ทำการตั้งค่าตัวแปร environment ไปให้ CGI สคริปต์

Content Type Decisions

ทำการตรวจสอบประเภทของเอกสาร

URL Mapping

ทำการ mapping host filesystem ที่อยู่ใน document tree พร้อมทั้งทำการ redirection

Directory Handling

ทำหน้าที่ในการจัดการเกี่ยวกับ directory

Access Control

ทำหน้าที่ควบคุมและจัดการเกี่ยวกับการพิสูจน์สิทธิ์การใช้งาน

HTTP Response

ทำหน้าที่จัดการเกี่ยวกับ HTTP header

Dynamic Content

ทำหน้าที่เอ็กซีคิว CGI สคริปต์โดยขึ้นอยู่กับคำร้องขอจากไคลเอนต์

Internal Content Handlers

ทำหน้าที่แสดงสถานะของเซิร์ฟเวอร์พร้อมทั้งให้ข้อมูลเกี่ยวกับค่า Configuration ต่างๆ

Logging

ทำหน้าที่จัดการเกี่ยวกับกระบวนการ Logging ภายในระบบ

Miscellaneous

ทำหน้าที่จัดการเกี่ยวกับ Imagemap, caching proxy, loading module ต่างๆ

Development

แสดงข้อมูลเกี่ยวกับ API ต่างๆ

Obsolete

ทำการเซตตัวแปร environment ต่างๆ

เมื่อมีการส่งคำร้องขอจากไคลเอนต์เข้ามาแล้ว Apache จะทำการแบ่งแยกคำร้องขอนั้นออกเป็นเฟส (Phases) ต่างๆ ได้ดังนี้

1. จะทำการแปลง URI (Uniform Resource Identifier) เพื่อดูว่ามีการร้องขอหรืออ้างอิงการใช้ทรัพยากร (Resource) หรือออบเจกต์ใดในระบบไฟล์ของเว็บเซิร์ฟเวอร์
2. ทำการตรวจสอบการเข้าถึงจากแอดเดรสต้นทางที่มีการติดต่อเข้ามา
3. ตรวจสอบ ID จากคำร้องขอตามรูปแบบโปรโตคอล HTTP
4. ตรวจสอบสิทธิในการขอใช้ออบเจกต์
5. ตรวจสอบการร้องขอและสิทธิอื่นๆ ซึ่งแตกต่างกันตามคำร้องขอ
6. ดูว่ามีการใช้ MIME ชนิดใดในคำร้องขอ
7. ตอบกลับไปยังเว็บไคลเอนต์ ซึ่งปกติจะเป็นออบเจกต์ที่มีการร้องขอเข้ามา
8. ทำการยุติการร้องขอที่เข้ามา

ซึ่งในแต่ละเฟสนี้จะทำการส่งไปให้โมดูลที่เกี่ยวข้อง โดยภายในโมดูลก็จะมี Handler ซึ่งทำหน้าที่รับผิดชอบในการจัดการคำร้องขอในแต่ละเฟสที่ส่งเข้ามา ซึ่ง Handler ก็จะทำหน้าที่หนึ่งในสามอย่างต่อไปนี้คือ

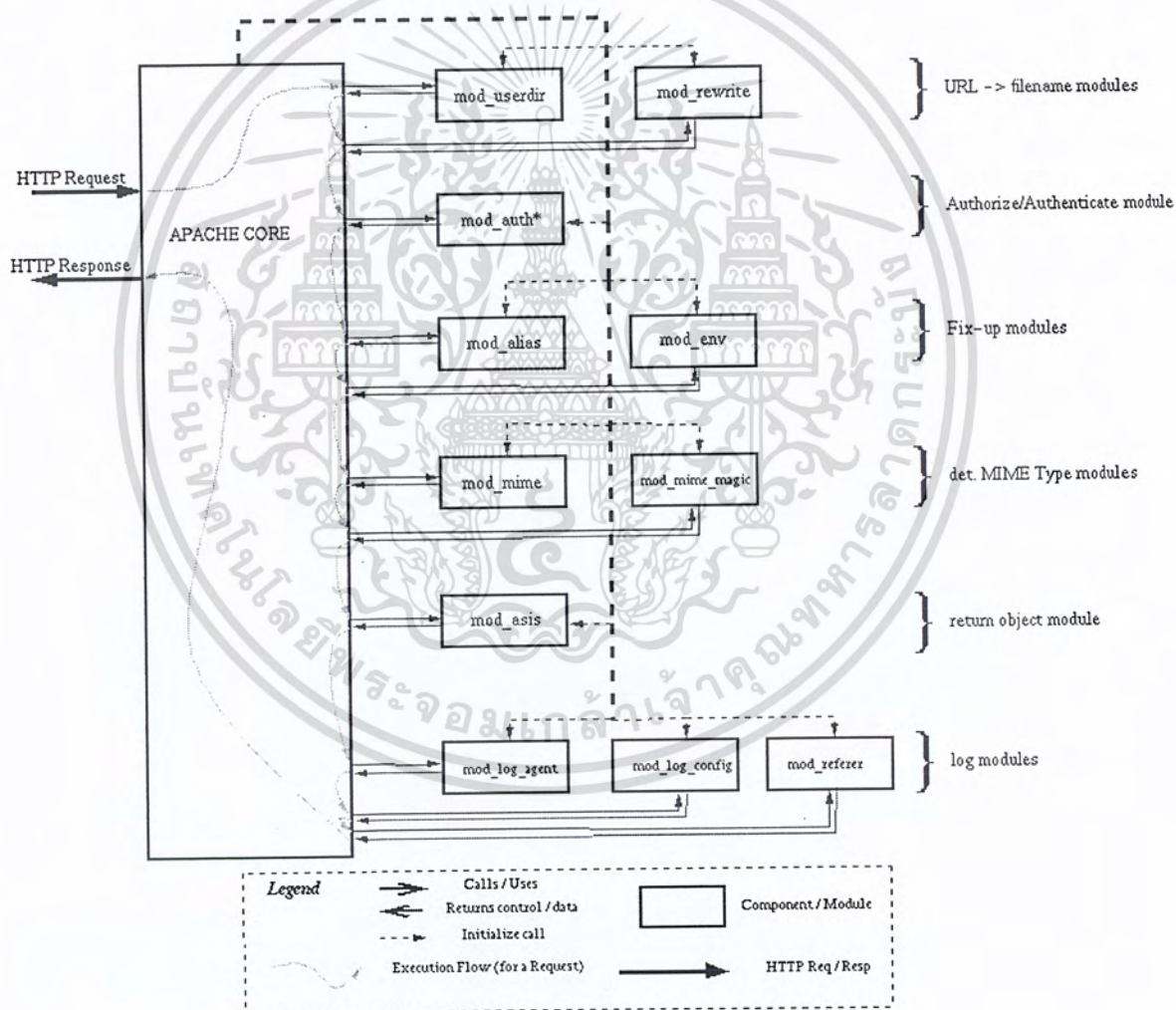
- ทำการจัดการให้ตามคำร้องขอ และส่ง magic constant ว่า OK กลับไปให้
- ไม่ยอมรับคำร้องขอ โดยส่ง magic integer constant ว่า DECLINED
- เกิดข้อผิดพลาดขึ้น โดยจะส่งข้อความ HTTP error code ออกไป

ในรูปแบบการทำงานด้านกายภาพ (Physical) ของ Apache จะมีกระบวนการทำงานที่สำคัญ เช่น การคอยฟังการร้องขอ การสร้างโพรเซสลูก (Child Process) รวมไปถึงการส่งคำร้องขอไปยังโพรเซสลูกเหล่านั้นให้ทำงานได้อย่างสมบูรณ์ ซึ่งโพรเซสลูกนี้จะถูกสร้างขึ้นมาทุกครั้งที่มีการร้องขอเข้ามา โดยจำนวนของโพรเซสลูกนี้จะมีจำนวนเท่าใดก็ขึ้นอยู่กับคำร้องขอที่เข้ามา ในการที่จะตอบสนองต่อคำร้องขอที่เข้ามานี้ โพรเซสลูกอาจมีการทำงานที่จำเป็นต้องสร้างโพรเซสอื่นๆ ขึ้นมาอีก เช่น โปรแกรม CGI หรือโพรเซสลูกอาจต้องการติดต่อสื่อสารกับโพรเซสอื่นๆ ที่มีหน้าที่ต่างกันไป เช่น โพรเซสที่ทำหน้าที่ในการจัดการเกี่ยวกับการตรวจสอบผู้ใช้ (User Authentication) เป็นต้น

ข้อดีและลักษณะเด่นของ Apache

1. สามารถทำงานบนระบบปฏิบัติการได้หลากหลาย
2. เป็น Free ware และ Open source
3. สนับสนุนการตรวจสอบโปรโตคอล HTTP ผ่านทาง Text File, DBM File และทาง SQL หรือจากโปรแกรมภายนอก (External Program)
4. สนับสนุน CGI และ Fast CGI
5. มีขั้นตอนการเซตไฟล์ต่างๆ ในระบบง่าย และสามารถปรับแต่งการทำงานได้หลากหลาย
6. สามารถเขียนโมดูล (Module) เพิ่มเติมเองได้ นอกจากโมดูลมาตรฐานที่มีอยู่แล้ว

7. สนับสนุน SSL (Secure Sockets Layer) คือ Apache SSL ที่เป็นแบบไม่เป็นทางการค้า (Noncommercial) และในเวอร์ชันที่เป็นแบบทางการค้า (Commercial)
 8. สนับสนุนการทำงานแบบ Virtual Host ทั้งแบบ IP-Based หรือ Name-Based
 9. มีความยืดหยุ่นในการควบคุม การตรวจสอบและยุติการสื่อสาร รวมถึงการตรวจสอบผ่านทางเว็บเบราว์เซอร์
 10. สนับสนุนโพรโตคอล HTTP 1.1 ซึ่งสามารถทำการรองรับการร้องขอที่เป็นแบบ Asynchronous
 11. มีความสามารถในการติดตามผู้ใช้จากคุกกี้ (Cookie)
 12. สนับสนุนการตรวจสอบในรูปแบบของ Message Digest
- ฯลฯ



รูปที่ 6-4 แสดงถึงสถาปัตยกรรมโครงสร้างของ Apache อย่างละเอียด

ข้อเสียและลักษณะด้อยของ Apache

1. ขาดรูปแบบการเชื่อมต่อแบบ GUI
2. ไม่มีการสนับสนุนจากผู้ขายในกรณีที่เป็นแบบ Free Ware

3. เกิดความไม่แน่นอนเมื่อมีการทำงานบนวินโดวส์แพลตฟอร์ม เนื่องจาก Apache มีการใช้ระบบ Multi-Threading อาจทำให้ทั้งระบบเกิดหยุดทำงานได้

6.1.2 ปัญหาทางด้านความปลอดภัยของ Apache

ในการศึกษาถึงปัญหาด้านความปลอดภัยของ Apache นั้น ได้จำแนกถึงชนิดของปัญหาด้านความปลอดภัยที่เกิดขึ้นได้ 4 กรณีหลักๆ ดังนี้

1. Input Validation Error
 2. Design Error
 3. Failure to Handle Exception Condition
 4. Configuration Error
1. **Input Validation Error:** เป็นปัญหาของระบบที่เกิดจากการที่ระบบได้รับอินพุตเข้ามาทำงานแล้วทำให้เกิดการทำงานที่ผิดพลาดขึ้น ตัวอย่างของปัญหาชนิดนี้มีดังนี้
 - **NCSA/Apache httpd ScriptAlias Source Retrieval Vulnerability**

ปัญหา: NCSA httpd ตั้งแต่ในเวอร์ชันที่ 1.5 ขึ้นไปและเว็บเซิร์ฟเวอร์ Apache ตั้งแต่เวอร์ชันที่ 1.0 ขึ้นไปจะมีบั๊กในส่วนของ ScriptAlias ซึ่งบั๊กตัวนี้จะทำให้ผู้ใช้สามารถที่จะทำการดูซอสโค้ดของ CGI โปรแกรมภายในเว็บเซิร์ฟเวอร์ได้ถ้าใครเรียกทอร์ ScriptAlias ถูกกำหนดอยู่ใน DocumentRoot และดัชนีของเว็บเซิร์ฟเวอร์ (Index Server) มีการกำหนดให้ใช้งานได้ ผู้ใช้ที่ติดต่อกับเว็บเซิร์ฟเวอร์ก็สามารถที่จะเข้าไปดูในไดเรกทอรีทั้งหมดของ CGI-BIN ได้

การแก้ไขและป้องกัน: ทำการดาวน์โหลดและเปลี่ยนเวอร์ชันของ Apache httpd ตัวใหม่ ซึ่งสามารถทำการดาวน์โหลดได้ที่ <http://httpd.apache.org/dist/httpd/>
 2. **Design Error:** เป็นความผิดพลาดที่เกิดขึ้นอันเนื่องมาจากการทำการพัฒนาและออกแบบในส่วน of Apache ที่ผิดพลาดซึ่งทำให้เกิดปัญหาและความไม่ปลอดภัยในการทำงาน โดยตัวอย่างของความผิดพลาดชนิดนี้มีดังต่อไปนี้
 - **Rewrite Module Arbitrary File Disclosure Vulnerability**

ปัญหา: ภายในเว็บเซิร์ฟเวอร์ Apache เวอร์ชัน 1.2 จะมีโมดูลตัวหนึ่งที่เรียกว่า mod_rewrite ซึ่งจะทำหน้าที่ในการเปลี่ยนจาก URL พิเศษไปเป็นชื่อไฟล์บนระบบไฟล์ของเว็บเซิร์ฟเวอร์ซึ่งโครงสร้างการทำงานภายในตัวโมดูลนี้จะมีคำสั่งที่เรียกว่ากฎซึ่งใช้ในการแปลงจาก URL ไปเป็นชื่อไฟล์หรือเรียกว่า Rewrite Rule ซึ่งกฎหรือ Rewrite Rule ของเว็บเซิร์ฟเวอร์ Apache 1.2 ได้รับการออกแบบมาซึ่งจะมีข้อบกพร่องในกรณีของบาง URL ที่กฎจะทำการแปลงไปเป็นชื่อไฟล์โดยการอ้างอิงสัญลักษณ์ (Expression) บางอย่าง และจะทำให้ผู้ที่ไม่ประสงค์ดีสามารถที่จะเข้าไปดูทุกไฟล์ที่อยู่ภายในระบบได้

การแก้ไขและป้องกัน: สามารถทำการดาวน์โหลดแพตช์สำหรับซอสโค้ดของ Apache ได้ที่ <http://bugs.apache.org/index.cgi/full/6671> ซึ่งจะทำให้การแก้ไขบั๊กต่างๆ และจำกัดหน้าที่ของโมดูล mod_rewrite บางอย่างออกไป

3. **Failure to Handle Exception Condition:** เป็นปัญหาและความผิดพลาดที่เกิดขึ้นคล้ายกับแบบ Design Error แต่ในกรณีนี้จะเป็นผลจากการที่ระบบไม่สามารถรองรับหรือแก้ไขปัญหาที่เกิดขึ้นได้ เกิดเป็นช่องโหว่ของระบบ ซึ่งทำให้ผู้ไม่ประสงค์ดีอาศัยจุดนี้ในการโจมตีระบบ ซึ่งส่วนใหญ่แล้ว ลักษณะของปัญหาที่เกิดขึ้นจะเป็นปัญหาที่ก่อให้เกิดการโจมตีแบบ Dos ดังตัวอย่างต่อไปนี้

- Apache Web Server Dos Vulnerability

ปัญหา: เว็บเซิร์ฟเวอร์ Apache ตั้งแต่เวอร์ชันที่ 1.2 ลงมาจะมีข้อบกพร่องที่ทำให้เกิดการโจมตีแบบ DoS ซึ่งเกิดจากการร้องขอในส่วนของคำสั่ง GET จากเว็บไคลเอนต์ไปยังเว็บเซิร์ฟเวอร์ ซึ่งจะทำให้การร้องขอนี้ทำให้เกิดอักขระ '/' เป็นจำนวนมากที่ฝั่งเว็บเซิร์ฟเวอร์ซึ่งจะทำให้ซีพียูไม่สามารถทำงานต่อไปได้และต้องทำการเริ่มต้นการทำงานของระบบใหม่อีกครั้งเพื่อให้สามารถทำงานได้ตามปกติ

การแก้ไขและการป้องกัน: สามารถทราบข้อมูลเพิ่มเติมและดาวน์โหลดแพตช์แก้ไขได้ที่

<http://httpd.apache.org/dist/httpd/>

4. **Configuration Error:** เป็นปัญหาที่เกิดขึ้นจากการติดตั้ง จัดการหรือจัดรูปแบบของไฟล์หรือไคลเอนต์ที่อยู่ในเว็บเซิร์ฟเวอร์ รวมไปถึงปัญหาที่เกิดขึ้นจากสคริปต์ต่างๆ ที่ถูกติดตั้งในระบบแล้วทำงาน และก่อให้เกิดข้อผิดพลาดขึ้น ตัวอย่างลักษณะของปัญหาดังกล่าวมีดังนี้

- SuSE Apache WebDAV Directory Listing Vulnerability

ปัญหา: WebDAV หรือ Web Distributed Authoring and Versioning จะเป็นโปรแกรมที่สนับสนุนการทำงานของโพรโตคอล HTTP ซึ่งจะทำให้ผู้ใช้สามารถที่จะทำการสร้าง แก้ไขและใช้เอกสารร่วมกันได้ โดยเฉพาะคำสั่ง REQUEST, METHOD และ PROPFIND จะช่วยให้ผู้ใช้สามารถดูคุณสมบัติ (Property) ของไฟล์ได้ เช่น ชื่อไฟล์ วันที่ทำการแก้ไขล่าสุด เป็นต้น ซึ่งถ้าเว็บเซิร์ฟเวอร์ Apache ได้ทำการติดตั้งโปรแกรม SuSE 6.4 ก็จะมีส่วนของ WebDAV ซึ่งจะทำให้ผู้ใช้สามารถที่จะเข้ามาดูโครงสร้างของไฟล์ทั้งหมดในเว็บเซิร์ฟเวอร์ได้

การแก้ไขและการป้องกัน: สามารถทำการดาวน์โหลดแพตช์เพื่อทำการแก้ไขได้ที่

<http://www.suse.com/us/support/download/updates/index.html>

6.1.3 การปรับแต่งค่าของ Apache ให้มีความปลอดภัย

การปรับแต่งค่าของระบบให้ถูกต้องนั้น จะมีส่วนช่วยในการป้องกันและลดความรุนแรงการโจมตีได้ โดยไคลเอนต์ที่สำคัญที่อยู่ในไฟล์ httpd.conf มีดังต่อไปนี้

LimitRequestBody

LimitRequestBody จะสามารถกำหนดได้ว่า ขนาดสูงสุดของคำร้องขอส่วนของบอดี (HTTP request body) มีขนาดเท่าไร โดยถ้ามีคำร้องขอที่มีขนาดมากกว่าที่กำหนดไว้ Apache ก็จะปฏิเสธคำร้องขอนั้นทันทีและส่งข้อความแสดงข้อผิดพลาดออกไป ซึ่งจะช่วยลดความรุนแรงของการโจมตีแบบการ

ปฏิเสธการให้บริการ (denial-of-service) ซึ่งอาจจะเกิดจากการส่งคำร้องขอต่อเนื่องมาเป็นจำนวนมาก ติดต่อกัน

โดยค่าของ `LimitRequestBody` ที่สามารถกำหนดได้จะเป็นตัวเลขที่อยู่ระหว่างช่วง 0 ถึง 2147483647 โดยที่ 0 หมายความว่า ไม่จำกัด ซึ่งโดยปกติก็จะเซตไว้เป็น 0 อยู่แล้ว ถ้าเราต้องการปรับเปลี่ยนค่านี้ จะต้องกำหนดขนาดให้สูงสุดให้พอดีกับขนาดของไฟล์ที่คาดว่าจะมีการร้องขอเข้ามา

LimitRequestFields

`LimitRequestFields` จะจำกัดจำนวนคำร้องขอของส่วนเซดเดอร์สูงสุดในการร้องขอแต่ละครั้ง ถ้ามีคำร้องขอส่วนเซดเดอร์ที่สูงกว่านี้ ก็จะปฏิเสธคำร้องขอนั้นทันที ซึ่งก็สามารถลดความรุนแรงของการโจมตีแบบการปฏิเสธการให้บริการได้ โดยอาจมีการส่งคำร้องขอเซดเดอร์เข้ามาเป็นจำนวนมากต่อเนื่องกันไป

โดยค่าของ `LimitRequestFields` ที่สามารถกำหนดค่าได้จะเป็นตัวเลขที่อยู่ระหว่าง 0 ถึง 32767 โดยที่ 0 หมายถึง ไม่จำกัด ซึ่งค่าปกติที่กำหนดไว้คือ 100 โดยค่าที่กำหนดไว้นี้ควรจะมากเพียงพอสำหรับรองรับการร้องขอเซดเดอร์ในแต่ละครั้งได้ ตัวอย่างของเซดเดอร์ ได้แก่ หมายเลขเซสชัน(session id), คุกกี (cookie), cache control headers, proxy headers, MIME headers เป็นต้น

LimitRequestFieldSize

`LimitRequestFieldSize` สำหรับจำกัดขนาดของการร้องขอเซดเดอร์ในแต่ละครั้ง โดยถ้าการร้องขอเซดเดอร์มีขนาดเกินกว่าที่กำหนดไว้ ก็จะปฏิเสธการร้องขอนั้นโดยทันที วิธีการนี้สามารถป้องกันการโจมตีโดยวิธี buffer-overflow ได้ โดยใคร่ที่พีพีสามารถปรับค่าได้ระหว่าง 0 ถึง 8190 โดยค่าดีฟอลต์จะเป็น 8190 ซึ่งสามารถป้องกันการโจมตีแบบ buffer-overflow ได้ โดยเราสามารถลดค่าดีฟอลต์นี้ลงมาได้ ถ้าพบว่ามีปัญหาการโจมตีแบบการปฏิเสธการให้บริการ

LimitRequestLine

`LimitRequestLine` จะจำกัดจำนวนบรรทัดในการร้องขอในแต่ละครั้ง ถ้าการร้องขอบรรทัดเกินจำนวนที่กำหนดนี้ คำร้องขอนั้นก็จะถูกปฏิเสธ ซึ่งวิธีนี้ก็สามารถป้องกันการโจมตีแบบ buffer-overflow ได้ ค่าที่ปรับได้ออยู่ระหว่าง 0 ถึง 8190 โดยค่าดีฟอลต์จะเป็น 8190 ซึ่งสามารถป้องกันการโจมตีแบบ buffer-overflow ได้ โดยเราสามารถลดค่าดีฟอลต์นี้ลงมาได้ถ้าพบว่ามีปัญหาการโจมตีแบบการปฏิเสธการให้บริการ

RLimitCPU

`RLimitCPU` จะจำกัดเวลาของโปรเซสในการทำงานซีพียูเป็นวินาที ซึ่งจะเป็นช่วงที่มีการสร้างโปรเซสใหม่ขึ้น เช่น การรัน CGI สคริปต์ ใคร่ที่พีพีสามารถกำหนดพารามิเตอร์ได้ 2 ค่าคือค่า minimum และ maximum ซึ่งค่าโดยดีฟอลต์จะขึ้นอยู่กับระบบปฏิบัติการเป็นหลัก โดยถ้าโปรเซสถูกใช้

เวลาของซีพียูเกินกว่าค่าที่กำหนดไว้นี้ โพรเซสซูก็นี้ก็จะถูกทำลายโดยทันที และจะแจ้งข้อความผิดพลาดออกไป

โดยไคเรกทีฟนี้สามารถป้องกันการเขียน CGI สคริปต์ที่ไม่ดีพอ ทำให้สิ้นเปลืองการทำงานของซีพียู ซึ่งก็สามารถป้องกันโดยการกำหนดเวลานั่นเอง

RlimitMem

RLimitCPU จะกำหนดจำนวนไบต์ของหน่วยความจำที่แต่ละโพรเซสสามารถจองไว้ใช้งานได้ ในแต่ละครั้งที่มีการสร้างโพรเซสลูกขึ้นมาใหม่ ไคเรกทีฟนี้สามารถกำหนดพารามิเตอร์ได้ 2 ค่าคือค่า minimum และ maximum ซึ่งค่าโดยดีฟอลต์จะขึ้นอยู่กับระบบปฏิบัติการเป็นหลัก โดยถ้าโพรเซสลูกใช้จำนวนของซีพียูเกินกว่าค่าที่กำหนดไว้นี้ โพรเซสซูก็นี้ก็จะถูกทำลายโดยทันที และจะแจ้งข้อความผิดพลาดออกไป

โดยไคเรกทีฟนี้สามารถป้องกันการเขียน CGI สคริปต์ที่ไม่ดีพอ ซึ่งเกิดจากการส่งข้อมูลจำนวนมากด้วยวิธี POST ไปที่เซิร์ฟเวอร์ เพื่อที่จะให้มีการใช้หน่วยความจำของระบบจำนวนมากจนเกิดข้อผิดพลาดขึ้น

RlimitNProc

RlimitNProc จะจำกัดจำนวนโพรเซสที่สามารถสร้างได้ของแต่ละโพรเซส โดยจะมีค่า minimum และค่า maximum ขึ้นอยู่กับระบบปฏิบัติการที่ใช้ โดยถ้ามีการสร้างโพรเซสเกินกว่าที่กำหนดไว้นี้ โพรเซสที่สร้างและโพรเซสลูกก็จะถูกทำลายทันทีและส่งข้อความผิดพลาดออกไป โดยวิธีนี้สามารถป้องกันการเขียน CGI สคริปต์ที่ไม่ดีพอ ซึ่งเกิดจากการส่งข้อมูลด้วยวิธี POST ซึ่งจะส่งคำสั่งเข้าไปรันในระบบได้ ซึ่งจะทำให้เกิดการสร้างโพรเซสขึ้นมา เราจึงสามารถตรวจสอบได้

KeepAlive

KeepAlive จะช่วยให้สามารถสร้างช่องสื่อสารถาวรได้ ซึ่งจะช่วยให้ประสิทธิภาพเนื่องจากช่วยให้ฝั่งไคลเอนต์สามารถร้องขอได้หลายๆ ครั้งโดยไม่ต้องเริ่มทำการเชื่อมต่อสื่อสารใหม่ทุกครั้ง โดยพารามิเตอร์ของไคเรกทีฟนี้จะมีแค่ on หรือ off เท่านั้น ซึ่งถ้าหากพบว่าเซิร์ฟเวอร์มีปัญหาในการถูกโจมตีแบบการปฏิเสธการให้บริการก็สามารถปรับให้เป็น off ได้เพื่อลดความรุนแรงจากการโจมตีให้น้อยลง

KeepAliveTimeout

KeepAliveTimeout จำกัดเวลาเป็นวินาทีว่าในแต่ละครั้งอาปาเซ่จะรอการร้องขอในคอนเน็กชันเดียวกันกี่วินาที โดยค่าดีฟอลต์ที่กำหนดไว้คือ 15 วินาที โดยสามารถป้องกันการโจมตีแบบการปฏิเสธการให้บริการโดยที่ผู้บุกรุกจะสร้างช่องทางการติดต่อสื่อสารถาวรขึ้น โดยที่ไม่ยอมปิด

MaxKeepAliveRequests

MaxKeepAliveRequests กำหนดจำนวนของคำร้องขอต่อการเชื่อมต่อในแต่ละครั้ง โดยค่าดีฟอลต์ที่กำหนดไว้คือ 100 ซึ่งถ้าปรับค่าลดลงให้น้อยกว่านี้ก็จะช่วยป้องกันการโจมตีแบบการปฏิเสธการให้บริการได้ ซึ่งเกิดจากผู้นุกรุกสร้างช่องทางสื่อสารถาวร และสั่งส่งคำร้องขอเป็น loop ที่ไม่จำกัด

ContentDigest

ContentDigest จะเป็นการคำนวณหา MD5 เพื่อตรวจสอบว่าข้อมูลที่ส่งไปนั้นถูกต้องหรือเปล่า โดยค่าที่เป็นไปได้มี 2 ค่าคือ on และ off ซึ่งโดยปกติจะเป็น off ซึ่งเราสามารถเซตให้เป็น on ได้ โดยแต่ละครั้งที่มีการ response จาก Apache จะมีการคำนวณ MD5 Message digest ของส่วน body และนำผลลัพธ์ที่ได้ไปเพิ่มไว้ใน Content-MD5 response header ซึ่งเมื่อไคลเอนต์ได้รับข้อมูลนี้แล้วสามารถตรวจสอบความถูกต้องได้ โดยทำการคำนวณหา MD5 ของ body แล้วนำมาเปรียบเทียบกับค่าที่ได้คำนวณมาแล้วจากฝั่งเซิร์ฟเวอร์ ซึ่งถ้าค่าที่ได้ตรงกันก็แสดงว่าเป็นข้อมูลที่ถูกต้อง แต่ถ้าค่าที่ได้ไม่ตรงกัน บราวเซอร์ก็จะสามารถรู้ได้ทันทีว่าข้อความที่ได้รับมานั้นมีการเปลี่ยนแปลงระหว่างทาง

Options Indexes FollowSymLinks MultiViews

โดยปกติแล้วภายในไคลเอนต์ของเว็บไซต์ จะมีการกำหนดดีฟอลต์ไฟล์ไว้คือ index.html ซึ่งหมายความว่าถ้ามีการเรียกไคลเอนต์นี้มาโดยไม่มีการอ้างไฟล์ใดๆ ก็จะหมายถึงกำลังเรียกไฟล์ index.html โดยอัตโนมัติ แต่ถ้าเราไม่มีไฟล์นี้เก็บไว้ อาปาเช่ก็จะทำการแสดงไคลเอนต์และไฟล์ทั้งหมดที่มีอยู่ภายในไคลเอนต์ที่เราอ้างถึงอยู่ ซึ่งถือว่าไม่ปลอดภัยเพราะจะทำให้ผู้ใช้เห็นไฟล์ทั้งหมดที่มีอยู่ ดังนั้นวิธีการป้องกันไม่ให้แสดงไคลเอนต์และไฟล์ในกรณีที่ไม่มีดีฟอลต์ไฟล์คือ การแทรกคอมเมนต์หน้าไคลเอนต์นี้ไว้ ดังนี้ #Options Indexes FollowSymLinks MultiViews

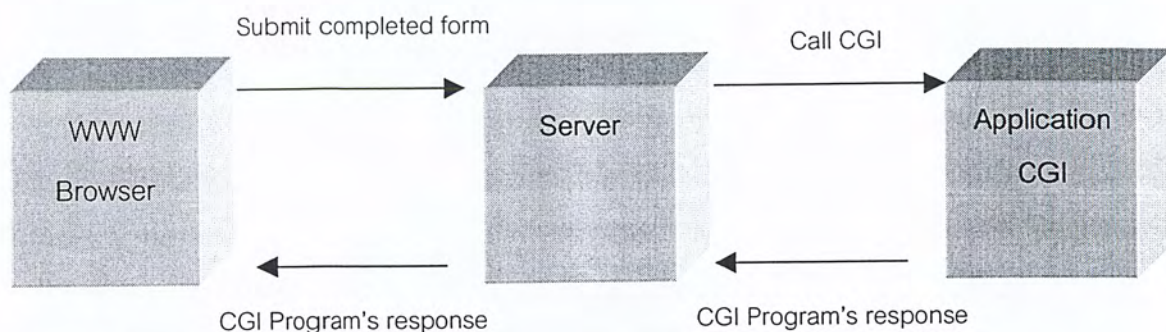
6.2 CGI (Common Gateway Interface)

CGI เป็นวิธีการมาตรฐานในการที่เว็บเซิร์ฟเวอร์รับคำร้องขอข้อมูลจากเว็บไคลเอนต์และนำข้อมูลไปประมวลผลพร้อมกับส่งผลลัพธ์ที่ได้กลับไปยังผู้ใช้

CGI เป็นส่วนหนึ่งของเว็บเซิร์ฟเวอร์ซึ่งสามารถทำการติดต่อกับโปรแกรมอื่นๆ ที่ทำงานอยู่บนเซิร์ฟเวอร์ได้ ด้วยคุณสมบัติของ CGI ทำให้เว็บเซิร์ฟเวอร์สามารถทำการเรียกโปรแกรมเหล่านั้น พร้อมทั้งส่งผ่านค่าพารามิเตอร์ต่างๆ ที่ส่งมาจากผู้ใช้ได้ ซึ่งโปรแกรมก็จะทำการประมวลผลข้อมูลที่ได้มาและเว็บเซิร์ฟเวอร์ก็จะทำการส่งผลลัพธ์ที่ได้จากการประมวลผลกลับไปให้เว็บเบราว์เซอร์

ยกตัวอย่างเช่น เมื่อผู้ใช้ทำการร้องขอเว็บไซต์ อาจจะโดยการคลิกลิงค์หรือการพิมพ์ที่อยู่ของเว็บไซต์ Server ก็จะทำการส่งหน้าที่ผู้ใช้ร้องขอมาให้และแสดงบนบราวเซอร์ สมมุติว่าหน้านั้นเป็นแบบฟอร์ม ในกรณีที่ผู้ใช้ทำการกรอกแบบฟอร์มบนเว็บและทำการส่งข้อมูลไปให้เซิร์ฟเวอร์ ซึ่งข้อมูลนี้ต้องการการประมวลผลโดยโปรแกรมที่อยู่ในเซิร์ฟเวอร์ซึ่งเว็บเซิร์ฟเวอร์ก็จะทำการส่งข้อมูลที่ได้จากฟอร์มไปให้โปรแกรมที่ทำหน้าที่ประมวลผลนี้ ซึ่งจะทำการประมวลผลข้อมูลและส่งค่าที่ได้กลับมา

หลักการทำงานอย่างคร่าวๆ ของ CGI มีดังต่อไปนี้



รูปที่ 6-5 แสดงขั้นตอนการทำงานของ CGI

CGI จะทำการประมวลผลข้อมูลที่ได้รับมาจากผู้ใช้ โดยแอปพลิเคชันส่วนใหญ่ที่สามารถทำการประมวลผลได้โดยใช้ CGI มีดังนี้

Forms

CGI นิยมนำมาใช้ในการประมวลผลแบบฟอร์มมากที่สุด โดยแบบฟอร์มนั้นเขียนขึ้นจากภาษา HTML โดยอนุญาตให้ผู้ใช้ใส่ข้อมูลต่างๆ ลงไปในฟอร์ม เมื่อผู้ใช้ทำการกรอกข้อมูลลงในฟอร์มเรียบร้อยแล้ว และทำการ submit ฟอร์มเพื่อส่งไปที่เซิร์ฟเวอร์โดย CGI จะทำการรับค่าที่อยู่ในฟอร์มแล้วนำไปประมวลผล แล้วส่งผลที่ได้กลับมาให้ผู้ใช้นั่นเอง ตัวอย่างของการใช้ฟอร์มมีดังต่อไปนี้

รูปที่ 6-6 ตัวอย่างแบบฟอร์มที่ใช้ในการรับส่งข้อมูล

Gateways

Web Gateways จะหมายถึงโปรแกรมหรือสคริปต์ที่ใช้ในการ access ข้อมูล ซึ่งไม่สามารถเรียกใช้งานได้โดยตรงจากไคลเอนต์ ยกตัวอย่างเช่น ในกรณีที่ต้องการนำเสนอข้อมูลบางอย่างที่เก็บอยู่ใน

Database ให้แก่ผู้ใช้ เราจะต้องใช้ CGI ในการเขียนคำสั่ง SQL เพื่อดึงอ่านข้อมูลที่ต้องการที่เก็บอยู่ใน Database เมื่อได้ข้อมูลที่ต้องการแล้วก็จะส่งข้อมูลนั้นมาให้ client ซึ่งในกรณีนี้ CGI ก็จะทำหน้าที่เป็น gateway ที่ทำการติดต่อไปยัง Database อีกที

Dynamic Document

Virtual Document หรือ Dynamic Document นั้นจัดเป็นหลักสำคัญของ CGI โดยจะเป็นการสร้าง output โดยที่ขึ้นอยู่กับค่าของ input ที่ได้รับเข้ามา โดย Dynamic Document อาจจะเป็นข้อความ ธรรมดา, รูปภาพ, เสียงต่างๆ หรืออื่นๆ

โปรแกรมที่เขียนเพื่อใช้งาน CGI นั้นสามารถเขียนได้หลายภาษา ซึ่งภาษาที่เป็นที่นิยมในปัจจุบัน เช่น C, C++, Java, Perl

6.2.1 Client-Side Script และ Server-Side Script

ในการพัฒนาเว็บแอปพลิเคชันในปัจจุบันนั้น นิยมนำภาษาสคริปต์มาใช้ในการพัฒนามากขึ้น ซึ่งภาษาสคริปต์ที่นำมาใช้ในการพัฒนาเว็บไซต์นี้ จะแบ่งออกเป็นสองประเภทคือ ภาษาสคริปต์ที่ทำงานที่ฝั่งไคลเอนต์ (Client-Side Script) และภาษาสคริปต์ที่ทำงานที่ฝั่งเซิร์ฟเวอร์ (Server-Side Script)

ภาษาสคริปต์ที่ทำงานที่ฝั่งไคลเอนต์นั้นจะถูกอ่านและแปลความหมายโดยเว็บเบราว์เซอร์ จะสคริปต์ประเภทนี้จะเขียนรวมอยู่กับเอกสาร HTML โดยส่วนใหญ่แล้วภาษาสคริปต์ที่ทำงานที่ฝั่งไคลเอนต์จะเขียนขึ้นเพื่อตรวจสอบหรือควบคุมการทำงานต่างๆ ที่ฝั่งไคลเอนต์ เช่น การตรวจสอบข้อมูลที่รับจากฟอร์ม การตอบสนองต่อการกระทำต่างๆ การควบคุมการทำงานของเบราว์เซอร์ เป็นต้น ภาษาสคริปต์ที่ทำงานที่ฝั่งไคลเอนต์ที่ได้รับความนิยมในปัจจุบัน มีหลายภาษา เช่น Java Script, JScript

ส่วนภาษาสคริปต์ที่ทำงานที่ฝั่งเซิร์ฟเวอร์นั้น เมื่อมีการร้องขอจากทางฝั่งไคลเอนต์ก็จะถูกคอมพิวเตอร์และรันที่ฝั่งเซิร์ฟเวอร์ และก็จะส่งผลลัพธ์ที่เป็นภาษา HTML กลับมาให้ทางฝั่งไคลเอนต์ ซึ่งภาษาสคริปต์ที่ทำงานที่ฝั่งเซิร์ฟเวอร์นี้จะช่วยให้สามารถพัฒนาเว็บแอปพลิเคชันที่มีความยืดหยุ่นและหลากหลายมากกว่าภาษาสคริปต์ที่ทำงานที่ฝั่งไคลเอนต์

ภาษาสคริปต์ที่ทำงานที่ฝั่งเซิร์ฟเวอร์นั้น จะถูกนำมาใช้งานหลายๆ ด้าน ไม่ว่าจะเป็น การเชื่อมต่อกับระบบฐานข้อมูล การแชร์ทรัพยากรระหว่างผู้ใช้ การเรียกใช้ไฟล์ที่อยู่บนเซิร์ฟเวอร์ การติดต่อสื่อสารกับโปรแกรมอื่นๆ หรือการสร้างเอกสารแบบไดนามิกที่ขึ้นอยู่กับความต้องการของผู้ใช้ เป็นต้น โดยภาษาที่นิยมใช้ในปัจจุบันมีหลายภาษา เช่น Perl, ASP, PHP, JSP , Cold Fusion

6.3 PHP

PHP นั้นย่อมาจาก PHP: Hypertext Preprocessor ซึ่งเป็นภาษาสคริปต์ที่ทำงานที่ฝั่งเซิร์ฟเวอร์ (Server Side Script) แบบเปิดเผยซอร์สโค้ด (Open-source) ซึ่งได้รับการออกแบบมาสำหรับการพัฒนาไดนามิกเว็บแอปพลิเคชัน (dynamic web application) รวมถึงการทำพาณิชย์อิเล็กทรอนิกส์ด้วย (e-commerce) โดยไดนามิกเว็บแอปพลิเคชันนั้น จะหมายถึงเว็บไซต์ที่สามารถมีการโต้ตอบกับผู้ใช้ได้ ซึ่งผู้

ที่เข้าไปเยี่ยมชมอาจจะสามารถกำหนดรูปแบบต่างๆ ตามต้องการได้ ซึ่งข้อมูลที่แสดงอาจจะสร้างมาจากข้อมูลที่อยู่ในดาต้าเบสหรือข้อมูลที่ได้รับผ่านแบบฟอร์มต่างๆ ก็ได้

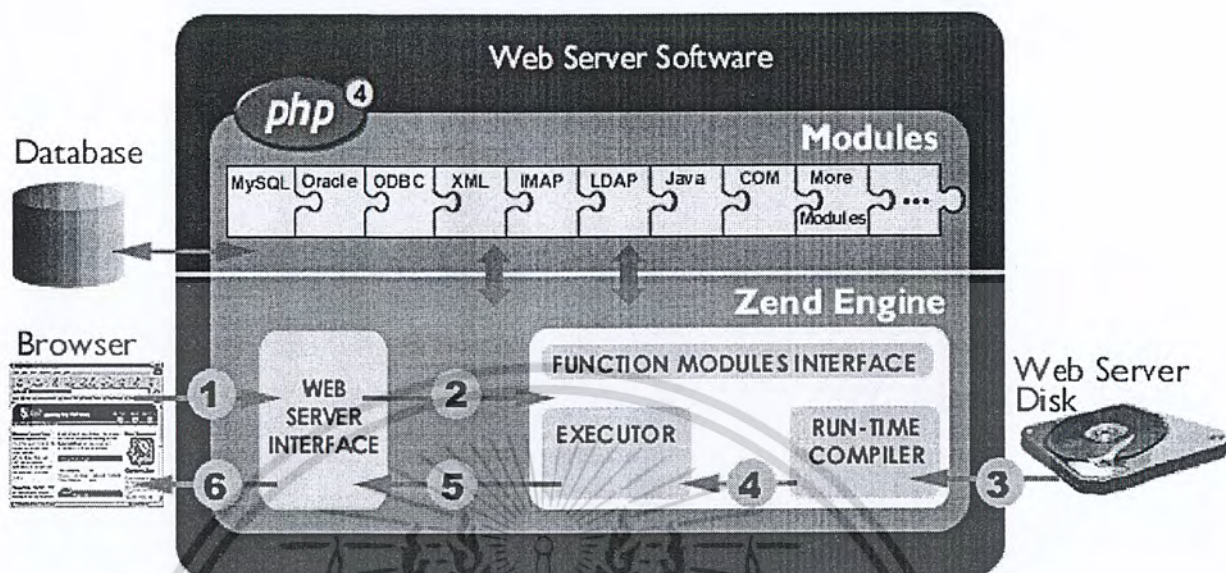
ในการพัฒนาเว็บแอปพลิเคชันด้วย PHP นั้นสามารถทำได้ง่ายมาก โดยสามารถเขียนโค้ดรวมไปกับภาษา HTML ได้เลย โดยไวยากรณ์ของภาษา PHP นั้นจะมีลักษณะคล้ายกับภาษา C หรือ Perl

PHP นั้นได้เริ่มมีการพัฒนาขึ้นมาใช้ครั้งแรกในปี 1994 โดยในปี 1998 ได้มีการพัฒนาไปเป็น PHP 3 และเริ่มเป็นที่นิยมอย่างรวดเร็ว จนปัจจุบัน PHP4 นั้นนับเป็นภาษาที่มีอัตราการเจริญเติบโตและจำนวนผู้ใช้สูงสุดในปัจจุบัน จากการสำรวจของ Netcraft และองค์กรที่มีชื่อเสียงต่างๆ ก็ได้เลือก PHP นี้ นำไปพัฒนาเว็บแอปพลิเคชันในองค์กรของตน อันได้แก่ Mitsubishi, Redhat, Der Spiegel, MP3-Lycos, Erricsson, NASA ฯลฯ

ข้อดีของ PHP

1. PHP มีการ Compile และ Execute ได้อย่างรวดเร็ว
2. สามารถพัฒนาได้ง่าย ในเวลาอันสั้น เนื่องจาก PHP นั้นเป็นการนำคุณสมบัติเด่นๆ ที่สำคัญของภาษา C, Perl และ Java นำมาพัฒนาให้มีประสิทธิภาพมากยิ่งขึ้นและง่ายต่อการใช้งาน ทำให้ง่ายต่อการบำรุงรักษาและเปลี่ยนแปลงแก้ไข โดยสามารถแทรกสคริปต์ที่ตำแหน่งใดๆ ก็ได้ในแท็กของ HTML
3. ไม่ขึ้นกับระบบใดๆ (Cross-platform) โดย PHP สามารถรองรับการทำงานได้บน Web Server ทั้งในระบบ Windows, Unix, Macintosh ไม่ว่าจะเป็น Apache, IIS, Netscape Enterprise, OmniHttpd และอื่นๆ
4. PHP สามารถรองรับระบบฐานข้อมูลที่หลากหลาย ที่มีใช้กันอยู่ในปัจจุบัน ไม่ว่าจะเป็น Empress, DB2, Informix, InterBase, mSQL, MySQL, MS-SQL, ODBC, Oracle, Sybase, PostgreSQL และอื่นๆ อีกมากมาย
5. PHP สามารถใช้งานทางด้านกราฟฟิคได้ ไม่ว่าจะเป็นการสร้างรูปเหลี่ยมต่างๆ กราฟแท่ง กราฟวงกลม และอื่นๆ อีกมาก
6. Open source โดยเปิดโอกาสให้โปรแกรมเมอร์ทั่วไปเข้ามามีส่วนร่วมในการพัฒนา ทำให้มีการพัฒนาได้เร็วขึ้นและมีคนใช้งานจำนวนมาก ข้อดีของ Open source ก็คือจะมีการแก้ไขข้อผิดพลาดและเพิ่มคุณสมบัติเด่นๆ ให้ตรงตามความต้องการของตลาดอยู่ตลอดเวลา PHP สามารถดาวน์โหลดได้ฟรีจากเว็บไซต์ <http://www.php.net>
7. Protocol Support โดย PHP นั้นสามารถเขียนเพื่อเชื่อมต่อและทำงานกับโปรโตคอลได้หลากหลาย ไม่ว่าจะเป็น IMAP, SNMP, NNTP, POP3, HTTP, Ftp, Socket ฯลฯ
8. Open API โดยผู้ใช้สามารถเขียนโปรแกรมที่เป็นโมดูลเพิ่มเติม ในการทำงานที่เฉพาะออกไปอีกได้

6.3.1 สถาปัตยกรรมของและ data flow แสดงการทำงานของ PHP 4



รูปที่ 6-7 แสดงขั้นตอนการทำงานของ PHP

ในแต่ละครั้งที่ผู้ใช้งานเรียกดูเว็บไซต์ที่ใช้ PHP 4 จะมีขั้นตอนการทำงานดังต่อไปนี้

1. เริ่มต้น Browser จะทำการร้องขอหน้าเว็บไซต์ไปที่ Web Server โดยจะส่ง request ไปยัง PHP Web Server Interface
2. Web server Interface จะทำการติดต่อไปยัง Zend Engine
3. Zend Engine จะเข้าไปดึงข้อมูลที่เป็น code ของ PHP จาก Web Server Disk และส่งต่อไปให้ Run-time Compiler
4. Zend Engine Run-time Compiler จะสร้าง compiled representation of the script ซึ่งจะส่งผ่านต่อไปยัง Zend Engine Executor
5. Zend Engine Executor จะทำการ generate ไฟล์ในรูปแบบ HTML ซึ่งสามารถ Browser สามารถมองเห็นและตีความได้ โดยในขั้นตอนนี้ถ้าพบว่าการเรียกใช้หรือติดต่อกับโมดูลอื่นๆ เช่น ทำการติดต่อกับ SQL database, XML หรือ Java Zend Engine ก็จะส่ง code ไปที่ PHP module ให้ทำการประมวลผลและส่งผลลัพธ์ที่ได้ไปให้ Web server Interface
6. Web server Interface จะทำการส่งผลลัพธ์ที่เป็น HTML ไปให้ Browser แสดงผล

6.3.2 Predefined variable

Predefined variable เป็นตัวแปรที่มีการกำหนดไว้ล่วงหน้าเรียบร้อยแล้ว ซึ่งผู้ใช้สามารถทำการเรียกใช้ได้ในทันทีภายในสคริปต์ที่กำลังใช้งานอยู่ ซึ่งตัวแปรเหล่านี้จะขึ้นอยู่กับเว็บเซิร์ฟเวอร์ที่ใช้งานอยู่ด้วย เพราะถ้าต่างชนิดกัน ตัวแปรก็จะแตกต่างกันออกไปแต่โดยภาพรวมแล้วก็จะมียกข้อยกเว้นที่คล้ายๆ กัน

ซึ่งในตัวอย่างที่ศึกษานี้จะเป็น Predefine variable ที่ใช้กับ Apache 1.3.6 โดยแสดงให้เห็นเพียงบางส่วนเท่านั้น ซึ่งในกรณีที่ต้องการดู Predefine variable ทั้งหมดสามารถดูได้โดยการเรียกใช้ฟังก์ชันดังนี้

```
<?php
phpinfo()
?>
```

ซึ่งฟังก์ชัน phpinfo() นี้จะแสดง Predefined variable ทั้งหมดพร้อมค่าต่างๆ ที่ได้ทำการเซตไว้ ซึ่งจะมีประโยชน์ในการติดตามและปรับแต่งค่าให้เหมาะสมยิ่งขึ้น แต่ควรใช้ด้วยความระมัดระวังเพราะว่าถ้าเกิดผู้บุกรุกได้เข้ามาเห็นการปรับแต่งค่าในไฟล์นี้อาจจะเป็นแนวทางในการหาวิธีเจาะเข้ามายังในระบบได้

Predefined variables นี้มีความสำคัญมากในการนำไปใช้งาน ซึ่งถ้าไม่มีการปรับแต่งที่เหมาะสม และการเรียกใช้งานอย่างถูกต้องแล้ว อาจจะเป็นช่องโหว่ที่ผู้บุกรุกจะเข้ามาภายในระบบได้ ซึ่งจะกล่าวรายละเอียดเกี่ยวกับเรื่องการปรับแต่งและการนำไปใช้งานต่อไปในภายหลัง

Predefined variables นี้จะแบ่งออกเป็น 2 ประเภทหลักๆ ด้วยกันคือ

1. Apache variables

ซึ่งตัวแปรเหล่านี้จะสร้างขึ้นมาโดยอพาเซในขณะที่เราเรียกใช้สคริปต์ ซึ่งตัวแปรที่สำคัญมีดังนี้

SSERVER_NAME

เป็นชื่อของเซิร์ฟเวอร์ที่สคริปต์ที่เราเรียกใช้กำลังใช้งานอยู่ เช่น "ISAG16"

SSERVER_PROTOCOL

เป็นข้อมูลเกี่ยวกับ protocol ของหน้าที่เว็บไซต์ทำการร้องขอ เช่น "HTTP/1.1"

SREQUEST_METHOD

จะแสดงข้อมูลว่าวิธีการใดที่ใช้ในการร้องขอเพื่อดูเว็บไซต์นั้น เช่น "GET", "HEAD", "POST", "PUT"

SDOCUMENT_ROOT

จะแสดง root directory ที่สคริปต์นั้นกำลังทำงานอยู่

SHTTP_CONNECTION

จะแสดง Connection header ของหน้าเพจปัจจุบันที่เรียกดู เช่น "keep Alive"

SREMOTE_ADDR

แสดง IP address ของผู้ใช้ที่เรียกดูข้อมูลจากเว็บไซต์

SREMOTE_PORT

แสดงถึง port ที่เครื่องของผู้ใช้ในการติดต่อกับ web server

SSERVER_PORT

แสดงถึง port บนเครื่อง server โดยปกติจะใช้ port 80 ซึ่งอาจจะมีการเปลี่ยนไปใช้ port อื่นได้ เช่น ในกรณีที่ต้องการติดตั้ง secure HTTP เพื่อใช้ SSL อาจเปลี่ยนไปใช้ port อื่นได้ตามความเหมาะสม

2. Environment variable

ตัวแปรเหล่านี้จะสร้างขึ้นโดย PHP เองซึ่งจะถือเป็นตัวแปรโกลบอลสามารถเรียกใช้ได้ทุกที่ในขณะที่เรียกใช้สคริปต์ ซึ่งตัวแปรที่สำคัญมีดังนี้

`$argv`

เป็นอะเรย์ของ argument ต่างๆ ที่ทำการส่งมาให้กับ PHP สคริปต์

`$PHP_SELF`

เป็นชื่อไฟล์ของสคริปต์ที่กำลังเรียกใช้อยู่ในปัจจุบัน

`$HTTP_COOKIE_VARS`

เป็นอะเรย์ของตัวแปรที่ส่งมาให้สคริปต์ผ่านทาง HTTP cookies

`$HTTP_GET_VARS`

เป็นอะเรย์ของตัวแปรที่ส่งมาให้สคริปต์โดยวิธี GET ผ่านทาง HTTP

`$HTTP_POST_VARS`

เป็นอะเรย์ของตัวแปรที่ส่งมาให้สคริปต์โดยวิธี POST ผ่านทาง HTTP

`$HTTP_ENV_VARS`

เป็นอะเรย์ของตัวแปรที่ส่งมาให้สคริปต์ผ่านทาง parent environment

`$HTTP_SERVER_VARS`

เป็นอะเรย์ของตัวแปรที่ส่งมาให้สคริปต์โดย HTTP server

`$HTTP_POST_FILES`

เป็นอะเรย์ของตัวแปรเกี่ยวกับการ upload file โดยวิธี POST ผ่านทาง HTTP ที่ส่งมาให้สคริปต์

สำหรับ `$HTTP_*_VARS` จะสามารถใช้งานได้ก็ต่อเมื่อมีการเซตให้ “track_var = on” ในไฟล์ `php.ini` ซึ่งเมื่อมีการเซตแล้วก็มีตัวแปรเกิดขึ้นแม้ว่าจะไม่มีการส่งตัวแปรมาให้สคริปต์ก็ตาม ก็จะมีการกำหนดเป็น empty อะเรย์ไว้ เพื่อป้องกันผู้ที่ไม่หวังดีทำการปลอมตัวแปรและค่าของตัวแปรได้ ซึ่ง `track_vars` นี้จะมีค่า on เสมอใน PHP 4.03 เป็นต้นไป แม้ว่าจะมีการ set ค่าไปเป็นอย่างอื่นก็ตาม

ในไฟล์ `php.ini` ถ้ามีการเซตค่า “register_global=on” ตัวแปรเหล่านี้ทุกตัวจะมีการกำหนดและสามารถเรียกใช้ได้เสมือนเป็นตัวแปร global ซึ่งแตกต่างจาก `HTTP_*_VARS` โดยสามารถเรียกใช้ได้เลยเสมือนตัวแปรทั่วไป เช่น `$name`, `$surname` ซึ่งค่า `register_global` นี้ควรใช้ด้วยความระมัดระวัง ถ้าเป็นไปได้ควรเซตให้เป็น off ซึ่งตัวแปร `$HTTP_*_VARS` จัดได้ว่าเป็นตัวแปรที่มีความปลอดภัยเพราะเป็นตัวแปรที่มีค่าจริงๆ ในขณะที่ตัวแปร global ธรรมดาสามารถเปลี่ยนค่าใหม่โดยผู้ใช้ได้ซึ่งนับเป็นจุดที่ควรระมัดระวังเป็นอย่างมาก ซึ่งถ้ามีสิ่งให้ “register_global=on” จะต้องมีการตรวจสอบอินพุตให้แน่ใจก่อนว่าเป็นค่าที่รับมาอย่างถูกต้องจริงๆ ก่อนจะนำไปใช้งานต่อไป

6.3.3 Regular Expression

Regular Expression หรือ Pattern Matching เป็นการประมวลผลและจัดการรูปแบบเฉพาะของข้อมูลในข้อความต่างๆ ซึ่งจะมีประโยชน์ดังต่อไปนี้

1. ใช้สำหรับตรวจสอบข้อมูลบางอย่างที่ต้องการ เพื่อให้ตรงกับความต้องการของข้อมูลที่จะนำไปใช้ เช่น การตรวจสอบอีเมลล์ของผู้ใช้ที่พิมพ์เข้ามา ว่าต้องอยู่ในรูปแบบของอีเมลล์จริงๆ การตรวจสอบรูปแบบตัวอักษร ตัวเลข หรือสัญลักษณ์ต่างๆ
2. ใช้สำหรับค้นหาหรือแทนที่ข้อมูลที่มีปริมาณมากๆ เช่น ตัวแปรที่เก็บข้อความไว้มากมาย หรือข้อความในไฟล์ทั้งไฟล์

ฟังก์ชันที่ใช้ในการเปรียบเทียบที่สำคัญคือ `ereg` ซึ่งมีรูปแบบการใช้งาน ดังนี้

`ereg` (รูปแบบที่ต้องการค้นหา, รูปแบบที่มีอยู่)

โดยฟังก์ชัน `ereg` จะคืนค่า `true` ถ้าตรวจพบรูปแบบที่ต้องการค้นหา และคืนค่า `false` ถ้าไม่พบรูปแบบที่ต้องการค้นหา โดยในการนำไปใช้งานนั้น จะนำไปใช้ร่วมกับ `if` เพื่อทำการเปรียบเทียบ เช่น

```
<?php
    $str = "Computer Engineering";
    if (ereg("Computer",$str)) {
        print "Yes found";
    } else {
        print "Not found";
    }
?>
```

จากตัวอย่างโปรแกรมนั้น ได้ทำการค้นหาคำว่า `Computer` ในตัวแปร `$str` ว่ามีอยู่หรือไม่ ซึ่งผลการค้นหาพบว่ามีอยู่ ผลลัพธ์จึงพิมพ์คำว่า `Yes found` ออกมา

พื้นฐาน Pattern Matching

Pattern จะหมายถึงคำหรือข้อความที่เราต้องการที่จะค้นหา ซึ่งจากตัวอย่างข้างบน Pattern จะเป็นคำว่า `Computer` แต่ในการใช้งานจริงๆ แล้ว เราสามารถกำหนดรูปแบบของ Pattern ได้ว่าจะให้รูปแบบของการค้นหานั้นออกมาเป็นอย่างไร โดยจะมีการนำเครื่องหมายเข้ามาช่วย เพื่อให้ Pattern ในการค้นหามีความยืดหยุ่นมากขึ้น

ในการตรวจสอบตัวอักษรจำนวน 1 ตัวว่ามีตัวอักษรนี้อยู่หรือไม่จะใช้รูปแบบ [...] เข้ามาช่วย เช่น

[a]	ตรวจสอบว่ามีตัวอักษร a อยู่หรือไม่
[R]	ตรวจสอบว่ามีตัวอักษร R อยู่หรือไม่
[a-z]	ตรวจสอบว่ามีตัวอักษร a-z ตัวใดตัวหนึ่งอยู่หรือไม่
[A-Z]	ตรวจสอบว่ามีตัวอักษร A-Z ตัวใดตัวหนึ่งอยู่หรือไม่
[a-zA-Z0-9]	ตรวจสอบว่ามีตัวอักษร a-z หรือ A-Z หรือ 0-9 ตัวใดตัวหนึ่งอยู่หรือไม่
[\\+\\-*\\/]	ตรวจสอบว่ามีเครื่องหมาย +, -, *, / ตัวใดตัวหนึ่งอยู่หรือไม่

การตรวจสอบจุดเริ่มต้นและจุดสิ้นสุด จะใช้รูปแบบ `^...$` โดยเครื่องหมาย `^` จะใช้ในการตรวจสอบจุดเริ่มต้นและ `$` ใช้ในการตรวจสอบจุดสิ้นสุด ดังตัวอย่างต่อไปนี้

<code>^computer</code>	ตรวจสอบว่าข้อความนั้นขึ้นต้นด้วยคำว่า computer หรือเปล่า
<code>network\$</code>	ตรวจสอบว่าข้อความนั้นขึ้นลงท้ายด้วยคำว่า network หรือเปล่า
<code>^network programming\$</code>	ตรวจสอบว่าข้อความนั้นเป็นคำว่า network programming หรือเปล่า

การตรวจสอบตัวอักษรที่ซ้ำกัน จะใช้เครื่องหมาย `{X}` โดย X คือจำนวนของตัวอักษรที่ซ้ำกันได้ ตัวอย่าง เช่น

<code>^g{3}\$</code>	ตรวจสอบว่าข้อความนั้นต้องเป็น ggg
<code>g{3}</code>	ตรวจสอบว่าข้อความนั้นต้องประกอบด้วย ggg โดยอยู่ที่ตำแหน่งใดก็ได้
<code>t{2,4}</code>	ตรวจสอบว่าข้อความนั้นมี t, tt, ttt ตัวใดตัวหนึ่งอยู่หรือไม่

การใช้เครื่องหมาย `.` และเครื่องหมาย `+` โดยเครื่องหมาย `.` จะหมายถึงตัวอักษรใดๆ ก็ได้ ส่วนเครื่องหมาย `+` จะหมายถึงว่าสามารถมีตัวอักษรนั้นได้มากกว่า 1 ตัว ตัวอย่าง เช่น

<code>^.{2}\$</code>	ตรวจสอบว่ามีตัวอักษร 2 ตัวใดๆ ก็ได้ เช่น ab, rr, ty
<code>^+S</code>	ตรวจสอบข้อความอะไรก็ได้ (ยกเว้นการขึ้นบรรทัดใหม่)

ตัวอย่างการนำความรู้ทางด้าน Regular Expression ไปประยุกต์ใช้ เช่น การตรวจสอบอีเมลที่รับเข้ามาอย่างง่าย โดยทั่วไปอีเมลจะอยู่ในรูป `XXX@XXX.XXX` ซึ่งสามารถเขียน Regular Expression ตรวจสอบได้ดังนี้คือ `^.+@.+\\.$` โดยมีตัวอย่างโปรแกรมดังนี้

```
<?php
    if (ereg("^.+@.+\\.$", $email)) {
        print "Email is correct";
    } else {
        print "Email not correct";
    }
?>
```


6.3.4 ปัญหาทางด้านความปลอดภัยของ PHP และแนวทางป้องกัน

ในการศึกษาถึงปัญหาด้านความปลอดภัยของการพัฒนาเว็บแอปพลิเคชันด้วย PHP นั้น สามารถจำแนกถึงชนิดปัญหาด้านความปลอดภัยที่เกิดขึ้นได้เป็น 4 หัวข้อหลักๆ ดังต่อไปนี้

1. **Global Variable** เป็นปัญหาที่เกิดจากการใช้ตัวแปร โกลบอลในการประมวลผลแบบฟอร์มต่างๆ
2. **Input Validation** เป็นปัญหาที่เกิดขึ้นจากการประมวลผลของ Script ที่ได้รับ input เข้ามาแล้วทำให้เกิดปัญหาขึ้น ซึ่งเกิดจากการไม่มีการตรวจสอบ input ที่รับเข้ามาอย่างละเอียดเพียงพอ ทำให้เกิดช่องโหว่และทำให้ผู้บุกรุกสามารถเข้ามาก่อความเสียหายได้ หรืออาจจะทำให้รันสคริปต์บางอย่างได้
3. **User Authentication** เป็นปัญหาที่เกิดจากการเขียนสคริปต์ที่ใช้ในการตรวจสอบและพิสูจน์สิทธิ์การใช้งานของผู้ใช้ที่ยังไม่รัดกุมพอ ทำให้เกิดช่องโหว่ภายในระบบได้
4. **Design Error** เป็นความผิดพลาดที่เกิดขึ้นอันเนื่องมาจากการพัฒนาและออกแบบระบบภายในของ PHP เอง ทำให้เกิดปัญหาและความไม่ปลอดภัยในการใช้งาน
5. **Configuration** เป็นปัญหาที่เกิดจากการติดตั้งหรือการจัดรูปแบบไฟล์หรือโครงสร้างใดเร็กทอรีภายในเว็บเซิร์ฟเวอร์ที่ยังไม่ดีพอ รวมไปถึงการปรับแต่งค่าต่างๆ ของระบบรวมทั้งสคริปต์ต่างๆ ที่ถูกติดตั้งเข้ามาในระบบที่ไม่เหมาะสม

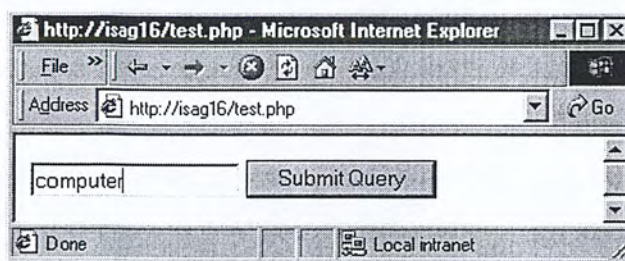
1. ปัญหาในการเรียกใช้งานตัวแปรโกลบอล (Global variables)

ตัวแปรใน PHP นั้นไม่ต้องการประกาศ แต่จะมีการสร้างขึ้นโดยอัตโนมัติเมื่อมีการเรียกใช้ โดยที่ไม่ได้มีการกำหนดประเภทของตัวแปร เพราะประเภทของตัวแปรนั้นจะขึ้นอยู่กับค่าภายในตัวแปรนั้น ซึ่งทำให้เกิดความสับสนและรวดเร็วในการพัฒนาแอปพลิเคชันมาก โดยเมื่อมีการสร้างตัวแปรขึ้นมาแล้วจะสามารถเรียกใช้ได้ทุกที่ในโปรแกรม (ยกเว้นฟังก์ชัน ถ้าฟังก์ชันจะเรียกใช้จะต้องกำหนดคำว่า global นำหน้าตัวแปรก่อนถึงเรียกใช้ได้) ซึ่งจากคุณสมบัตินี้เองทำให้โปรแกรมเมอร์นั้นแทบจะไม่ได้คำนึงถึงในประกาศตัวแปรต่างๆ ด้วยตัวเองเลย เมื่อมีการสร้างตัวแปรในตอนต้นจะมีค่าเป็น empty

ฟังก์ชันส่วนใหญ่ของ PHP นั้น ส่วนใหญ่จะเป็นฟังก์ชันที่รับค่า input บางส่วนจากผู้ใช้ (เช่น form variables, upload file, cookies, etc) ซึ่งจะมีการประมวลผล input จากผู้ใช้และส่งค่า output กลับมาโดยขึ้นอยู่กับค่า input ที่ได้รับ โดยวิธีการที่ง่ายที่สุดในการรับ input จากผู้ใช้ก็ทำได้โดยการใช้ฟอร์ม ดังตัวอย่างต่อไปนี้

```
<FORM METHOD="GET" ACTION="test.php">
<INPUT TYPE="TEXT" NAME="hello">
<INPUT TYPE="SUBMIT">
</FORM>
```


โดยผลที่ได้จากฟอร์มนี้เมื่อแสดงผลใน Browser คือ



รูปที่ 6-8 ตัวอย่างแบบฟอร์ม

โดยฟอร์มนี้จะแสดง Text box และปุ่ม submit โดยเมื่อผู้ใช้พิมพ์ข้อความลงไปใน Text box และกดปุ่ม submit ก็จะเกิดตัวแปรขึ้นมาหนึ่งตัวคือ Shello ซึ่งตัวแปรนี้จะเก็บค่าจาก Text box ที่ผู้ใช้พิมพ์ลงไป แล้วส่งตัวแปรนี้ไปให้ไฟล์ test.php ซึ่งไฟล์ test.php นี้ก็จะรับ input จากผู้ใช้ไปประมวลผลต่อไป จากตัวอย่างในรูปเมื่อผู้ใช้พิมพ์คำว่า “computer” และกดปุ่ม submit ก็จะเกิดตัวแปร Shello=“submit” ขึ้นแล้วส่งไปให้ไฟล์ test.php ต่อไป ซึ่งส่วนนี้เองเป็นส่วนสำคัญโดยผู้บุกรุกสามารถสร้างตัวแปร global อะไรก็ได้ที่ต้องการ โดยแทนที่เขาคจะเรียกใช้ไฟล์ฟอร์มข้างต้นเพื่อส่งตัวแปร Shello พร้อมค่าไปให้ไฟล์ test.php โดยตรง เขาก็จะเรียก URL ปกติพร้อมใส่ค่าตัวแปรพร้อมค่าลงไป เช่น <http://server/test.php?hello=hi&user=admin> ซึ่งไม่เพียงแต่จะส่งตัวแปร Shello=“hi” เท่านั้น แต่ยังสามารถกำหนดตัวแปรอะไรที่ต้องการก็ได้ ซึ่งในที่นี้จะเห็นได้ว่าการกำหนด Suser=“admin” มาด้วย ซึ่งจะเห็นได้ว่าผู้บุกรุกนั้นสามารถกำหนดตัวแปรอะไรก็ได้ตามใจชอบแล้วส่งมาให้ไฟล์ test.php

สำหรับปัญหาและข้อเสียที่เห็นได้จากตัวอย่างข้างต้นนี้คือ กรณี Script ที่ใช้สำหรับ พิสูจน์สิทธิการใช้งานก่อนที่จะอนุญาตให้ผู้ใช้เข้าไปชมในส่วนที่แสดงเนื้อหาที่สำคัญในเว็บไซด์ เช่น

```
<?php
if ($pass == "hello")
    $auth = 1;
.....
if ($auth == 1)
    echo "This is important information";
?>
```

ตัวอย่างข้างบนโดยทั่วไปจะทำการเช็ค Password ก่อนที่ user จะได้รับ authenticate และทำการเช็ค ว่า ถ้า authenticate แล้วถึงจะให้เข้าไปดูหน้าที่สำคัญได้ ซึ่งตัวอย่างนี้จะสมมุติว่าตัวแปร \$auth ไม่ได้เก็บค่าไว้จนกว่าจะมีการกำหนดค่ามันให้เป็น 1 โดยเมื่อ Password ที่รับเข้ามาถูกต้องถึงจะทำการกำหนด

ค่าให้ Sauth=1 ซึ่งผู้บุกรุกสามารถสร้างตัวแปร global นี้ได้โดยการพิมพ์ตัวแปรลงไปที่ Address bar ได้เลย ดังเช่น <http://server/test.php?auth=1> ซึ่งวิธีนี้จะทำให้ไม่ต้องใส่ password ก็สามารถ authenticate และเข้าสู่ข้อมูลที่สำคัญได้

ซึ่งปัญหานี้กล่าวโดยสรุปก็คือว่าผู้เขียนไม่ได้มีการตรวจสอบ Input ที่รับเข้ามาอย่างชัดเจน ซึ่งวิธีการแก้ปัญหานี้จะต้องทำการตรวจให้แน่ชัดว่า input มาจากที่ใดและผ่านการตรวจอย่างละเอียดหรือเปล่า โดยเฉพาะในกรณีที่มีการติดต่อกับตัวแปรจำนวนมากจะต้องมีการตรวจสอบให้ละเอียดและครบถ้วนก่อนจึงจะอนุญาตให้ประมวลผลได้

แนวทางแก้ไข

การแก้ไขปัญหาวัดแปรโกลบอลสามารถทำได้โดยยกเลิกการใช้ตัวแปรโกลบอลโดยการเช็คค่า “register_global=off” ซึ่งจะทำให้ไม่สามารถเรียกใช้ตัวแปรโกลบอลได้เลย รวมทั้งการส่งค่าตัวแปรผ่านทาง url ของเว็บไซต์ ซึ่งเมื่อมีการเช็คดังกล่าวแล้วเวลาที่ต้องการอ้างถึงตัวแปรต่างๆ ก็สามารถอ้างถึงตัวแปรอะไรที่ส่งมาจาก HTTP แทน เช่น เดิมเรามีตัวแปร Smyvar ซึ่งส่งมาจากแบบฟอร์มที่ผู้ใช้ได้กรอกผ่านวิธี POST เมื่อทำการเช็คค่า “register_global = off” แล้ว เวลาจะเรียกใช้ตัวแปรนี้ก็จะเรียกใช้ได้เป็น HTTP_POST_VARS[‘myvar’] แทน

2. ปัญหาในการตรวจสอบอินพุตจากผู้ใช้ (Input Validation)

ปัญหาในการตรวจสอบอินพุตจากใช้นั้นนับเป็นปัญหาที่สำคัญอย่างหนึ่งที่ต้องให้ความสำคัญเป็นอย่างมาก เนื่องจากเว็บไซต์ในปัจจุบันจะมีส่วนที่รับอินพุตจากผู้ใช้เป็นจำนวนมาก เช่น แบบฟอร์มต่างๆ ที่ให้ลงทะเบียน, เว็บบอร์ด, สมุดเยี่ยมชม ฯลฯ ซึ่งถ้าไม่มีการตรวจสอบข้อมูลอย่างถูกต้องอาจจะก่อให้เกิดปัญหาภายหลังได้ เนื่องจากข้อมูลที่ผู้ใช้กรอกลงมาอาจจะไม่ใช่ข้อมูลจริงๆ แต่อาจจะเป็นโปรแกรม หรือ คำสั่งต่างๆ ที่อาจก่อให้เกิดความเสียหายหรือเข้าไปดูข้อมูลภายในระบบได้

ปัญหาที่เกิดจากการรับข้อมูลที่ไม่มีการตรวจสอบมาจากผู้ใช้ มีดังนี้

1. ปัญหาการแสดงผลแท็ก HTML เนื่องจากในแบบฟอร์มที่รับข้อมูลจากผู้ใช้ส่วนใหญ่นั้น ผู้ใช้สามารถกรอกข้อมูลอะไรลงไปก็ได้ ซึ่งอาจจะเป็นข้อมูลจริงๆ หรืออาจจะเป็นโค้ดภาษา HTML ลงไปก็ได้ ซึ่งโดยปกติแล้ว ถ้าไม่มีการป้องกันแท็ก HTML เอาไว้ เมื่อผู้ใช้พิมพ์แท็ก HTML ลงไปจะทำให้ Browser ตีความหมายและแสดงผลแท็กนั้นได้ แต่ในกรณีที่ผู้ใช้เขียน Tag ไม่รอบคอบพอก็จะทำให้เกิดปัญหาได้ เช่น ในกรณีที่ผู้ใช้ทำการเปิดแท็ก HTML ไว้แต่ไม่ปิดแท็ก HTML ทำให้แท็กนั้นมีผลต่อทั้งเอกสาร ยกตัวอย่างเช่น ผู้ใช้พิมพ์ข้อมูลต่อไปนี้ลงไป

<FONR COLOR="#000000" size=7> Test

จากแท็กที่ผู้ใช้พิมพ์ลงไป โดยที่ไม่มีการปิดแท็กด้วย `</FONT></B>` จะทำให้เอกสารที่แสดงผลที่ผู้ใช้พิมพ์ลงไปนี้เกิดเป็นตัวหนาและตัวอักษรสีดำและมีขนาด 7 ทั้งเอกสาร ซึ่งจะ ทำให้เอกสารดูไม่สวยงามและยากต่อการอ่าน ซึ่งวิธีการป้องกัน HTML

2. ปัญหาการ Query ในฐานข้อมูล โดยปกติแล้วงานบางอย่าง จะต้องมีการรับข้อมูลบางส่วน จากผู้เข้ามา และนำเอาข้อมูลนั้นไปทำการ Query ในฐานข้อมูล ยกตัวอย่าง เช่น เว็บไซต์ที่ อนุญาตให้ผู้ใช้สามารถเปลี่ยน Username ได้เมื่อทำการ Login เข้าไปในระบบแล้ว โดยจะมีฟอร์มให้ผู้ใช้ทำการพิมพ์ชื่อ Username ใหม่ที่ต้องการเปลี่ยน เมื่อมีการส่งค่าตัวแปรที่เก็บ Username ใหม่ที่ต้องการเปลี่ยน เพื่อนำไป Query ในฐานข้อมูล คำสั่งที่ใช้ในการ Query นั้นจะอยู่ในรูปแบบดังนี้

```
UPDATE users SET Name='Susename' WHERE ID=1;
```

ในคำสั่งนี้ จะเห็นได้ว่าการส่งตัวแปร Susename ที่เป็นชื่อใหม่ที่ต้องการเปลี่ยนเข้ามา สมมุติว่าผู้ใช้ต้องการเปลี่ยน Username ใหม่เป็น Hanako ก็จะพิมพ์ชื่อนี้ลงในฟอร์ม ซึ่งคำสั่งที่ใช้ในการ Query จะได้ดังนี้

```
UPDATE users SET Name='Hanako' WHERE ID=1;
```

จะเห็นได้ว่าตัวอย่างข้างต้นนั้นสามารถทำงานได้โดยไม่มีปัญหาอะไร แต่ในกรณีที่ผู้ใช้ พิมพ์ Username ที่ต้องการเปลี่ยนใหม่เป็น Hanako',Admin='1 คำสั่งที่ได้ใน Query จะเป็นดังนี้

```
UPDATE users SET Name='Hanako',Admin='1' WHERE ID=1;
```

จะเห็นได้ว่าผู้ใช้นอกจากจะสามารถเปลี่ยน Username ของตัวเองได้แล้ว ยังสามารถทำการ เปลี่ยนค่า Admin ที่อยู่ในฐานข้อมูลได้ด้วย ซึ่งจะเป็นผลเสียในกรณีที่ฐานข้อมูลของ เว็บไซต์มีใช้เช็คค่า Admin ให้เป็น 1 หรือ 0 เพื่อตรวจสอบว่า Username นี้เป็น Admin หรือไม่

3. ปัญหาการตรวจสอบข้อมูลทั่วไป เป็นการตรวจสอบข้อมูลทั่วไปในแบบฟอร์ม ซึ่งผู้ใช้งาน คนอาจจะพิมพ์ค่าที่ไม่ตรงกับที่แบบฟอร์มต้องการได้ เช่น แบบฟอร์มต้องการหมายเลข ประจำตัว 8 หลัก แต่ผู้ใช้กลับพิมพ์เป็นตัวอักษร เป็นต้น ซึ่งวิธีนี้อาจไม่ก่อให้เกิดความเสียหายมากนัก แต่จะทำให้มีข้อมูลที่ไม่ถูกต้องเก็บลงในฐานข้อมูล

แนวทางแก้ไข

1. การป้องกันผู้ใช้พิมพ์แท็ก HTML เข้ามา สามารถป้องกันได้โดยการใช้ฟังก์ชัน `htmlspecialchars()`; ซึ่งฟังก์ชันนี้จะทำการเปลี่ยนสัญลักษณ์ต่างๆ เหล่านี้ ได้แก่ `&`, `"`, `'`, `<`, `>` ให้เป็นชื่อรหัสแทน ดังนี้

`'&' (ampersand)` เปลี่ยนเป็น `'&'`

`"" (double quote)` เปลี่ยนเป็น `'"'`

`" (single quote)` เปลี่ยนเป็น `'''`

`'<' (less than)` เปลี่ยนเป็น `'<'`

`'>' (greater than)` เปลี่ยนเป็น `'>'`

ซึ่งรหัสพวกนี้จะเป็นรหัสที่ใช้ในการแสดงผลเท่านั้น โดยเบราว์เซอร์เมื่ออ่านเจอรหัสเหล่านี้ จะไม่ทำการตีความหมายว่าเป็นแท็กที่ใช้แสดงผลอะไร แต่จะแสดงแท็กนั้นออกมาเลย เช่น เมื่อผู้ใช้พิมพ์คำว่า

`<B> This is BOLD </B>`

เมื่อใช้ฟังก์ชัน `htmlspecialchars()` แล้ว จะได้เป็น `<B> This is BOLD </B>`

เมื่อ Browser อ่านเจอรหัสพิเศษนี้ เวลาแสดงผลก็จะแสดงเป็น Tag ออกมาเลยคือ

`<B> This is BOLD </b>` โดยที่ไม่ได้แสดงเป็น **This is BOLD**

2. การป้องกันปัญหาในการ Query ข้อมูลที่ผิดพลาดลงในฐานข้อมูลสามารถทำได้โดย การใช้ฟังก์ชัน `addslashes()` ทุกครั้ง โดยตัวแปรทุกตัวที่รับมาจากผู้ใช้นั้นจะเอาไปใช้กับคำสั่ง Query ในฐานข้อมูลควรจะใช้ฟังก์ชัน `addslashes()` กับข้อมูลนั้นก่อน ซึ่งฟังก์ชัน `addslashes()` นี้จะทำการใส่เครื่องหมาย `\` (backslash) เข้าไปในตัวแปรนั้น ถ้าภายในตัวแปรมีการใช้เครื่องหมาย `'` (single quote), `""` (double quote), `\` (backslash), และค่าที่เป็น Null เพื่อเป็นการแบ่งแยกระหว่างคำสั่งที่ใช้ในการ Query ในฐานข้อมูลกับค่าของตัวแปรที่รับเข้ามาให้ชัดเจน และไม่ให้เกิดปัญหา
3. การตรวจสอบข้อมูลทั่วไป ว่าเป็นข้อมูลที่ถูกต้องตรงตามที่เว็บไซต์ต้องการนั้น สามารถทำได้โดยการใช้ Regular Expression ในการตรวจสอบ

3. ปัญหาที่เกิดจากการพิสูจน์สิทธิ์การใช้งานของผู้ใช้ (User Authentication)

ในการพิสูจน์สิทธิ์การใช้งานของผู้ใช้นั้น สามารถทำได้หลายวิธี โดยที่แต่ละวิธีก็มีความยากง่าย และประสิทธิภาพในการพิสูจน์สิทธิ์ผู้ใช้ที่แตกต่างกัน ซึ่งถ้าใช้วิธีที่ไม่เหมาะสม หรือวิธีที่มีข้อบกพร่องจนเกินไป ก็อาจทำให้เกิดปัญหาการบุกรุกเข้ามาภายในระบบได้ โดยในหัวข้อนี้จะนำเสนอเกี่ยวกับการพิสูจน์สิทธิ์ผู้ใช้โดยใช้ตัวแปรเซสชันซึ่งนับเป็นวิธีการหนึ่ง ที่มีความปลอดภัยสูงมากในปัจจุบัน

Session ในที่นี้จะหมายถึง กระบวนการในการตรวจสอบสถานะของผู้ใช้แต่ละคนในขณะที่กำลังเยี่ยมชมอยู่ภายในเว็บไซต์ ในช่วงเวลาหนึ่ง ซึ่งโดยปกติแล้ว HTTP จะไม่มีกระบวนการในการตรวจสอบผู้ใช้ในแต่ละครั้งที่ผู้ใช้ร้องขอทรัพยากรต่างๆ เพราะในการร้องขอแต่ละครั้งเมื่อได้รับการตอบสนอง

แล้ว ก็จะหยุดการเชื่อมต่อ เมื่อมีการร้องขอใหม่ ทางฝั่งเซิร์ฟเวอร์ก็ไม่มีทางที่จะรู้ว่าผู้ใช้นี้เป็นคนเดียวกับที่เคยเข้ามาหรือไม่ ซึ่งในการที่จะติดตามการทำงานของผู้ใช้แต่ละคนว่าทำอะไรไปบ้างนั้น สามารถทำได้โดยการใช้คุณสมบัติของตัวแปรเซสชัน ซึ่งคุณสมบัติใหม่ที่สำคัญที่เพิ่มเข้ามาใน PHP 4.0

หลักการทำงานโดยรวมของเซสชันคือ เมื่อมีการเริ่มการทำงานโดยใช้เซสชันนั้นผู้ใช้แต่ละคนที่เข้ามาใช้บริการหรือร้องขอข้อมูลต่างๆ ภายในเว็บไซต์ แต่ละคนจะได้รับหมายเลขเซสชันเก็บไว้ในเครื่องของแต่ละคน โดยหมายเลขเซสชันที่ทำการสร้างให้แต่ละคนนั้น จะเป็นหมายเลขที่ไม่ซ้ำกันและมีการเข้ารหัสไว้เพื่อเป็นการพิสูจน์ความเป็นบุคคลของแต่ละคน ซึ่งหมายเลขเซสชันนี้จะมีการเก็บไว้ในที่เซิร์ฟเวอร์ด้วย โดยเมื่อผู้ใช้มีการเรียกใช้หรือติดต่อกับตัวแปรตัวใด ก็จะทำให้ตัวแปรตัวนั้นเป็นตัวแปรเซสชัน ซึ่งจะมีการเก็บค่าไว้ในที่เซิร์ฟเวอร์ด้วย โดยถ้ามีการเรียกใช้ข้อมูลหรือตัวแปรแต่ละครั้งก็จะมีการตรวจสอบว่าหมายเลขเซสชันในเครื่องของผู้ใช้กับหมายเลขเซสชันในเครื่องเซิร์ฟเวอร์ตรงกันหรือไม่ ถ้าตรงกันก็จะอนุญาตให้มีการเรียกใช้ตัวแปรต่างๆ ได้ ซึ่งจะสามารถป้องกันการปลอมแปลงผู้ใช้เป็นผู้อื่นได้ เนื่องจากหมายเลขเซสชันนั้นจะไม่เหมือนกันในแต่ละคนและมีการเข้ารหัสไว้ด้วย

ตัวอย่างของหมายเลขเซสชันที่มีการเข้ารหัสไว้คือ 526as329mu4747y123xfr45874s198e9

ในการตรวจสอบสิทธิการใช้งานของผู้ใช้นี้ ก็จะให้ดำเนินการอยู่ภายใต้การทำงานของเซสชันทั้งหมด ซึ่งในการล็อกอินเข้าไปทำงานในระบบแต่ละครั้งก็จะมีการเข้ารหัสข้อมูล โดยใช้โปรโตคอล SSL เพื่อเข้ารหัสข้อมูลในการส่งข้อมูลระหว่างไคลเอนต์กับเซิร์ฟเวอร์ นอกจากนี้ในตรวจสอบรหัสผ่านของผู้ใช้ที่อยู่ในฐานข้อมูลก็ยังมีมีการเข้ารหัสเพื่อเพิ่มความปลอดภัยในอีกระดับหนึ่ง และในขณะที่ผู้ใช้กำลังใช้บริการจากเว็บไซต์ก็จะได้รับหมายเลขเซสชันที่มีการเข้ารหัสอีกเช่นกัน เพื่อความปลอดภัยในการทำงานและใช้บริการต่างๆ ผ่านทางเว็บไซต์

4. ปัญหาที่เกิดจากโครงสร้างภายในของ PHP เอง (Design Error)

เป็นความผิดพลาดที่เกิดขึ้นอันเนื่องมาจากการพัฒนาและออกแบบระบบภายในของ PHP เอง ทำให้เกิดปัญหาและความไม่ปลอดภัยในการใช้งาน ตัวอย่าง ปัญหาที่พบ เช่น

1. การแสดงซอร์สโค้ดของ PHP มีการพบว่าในบางครั้ง มีการแสดงผลไฟล์ PHP ที่บราวเซอร์เลย แทนที่จะถูกเอ็กซีคิวให้แปลงให้เป็นไฟล์ HTML ก่อน ซึ่งทำให้ผู้ใช้สามารถมองเห็นซอร์สโค้ดการทำงานภายในได้
2. มีรายงานเข้ามาว่าในบางครั้งสามารถทำการเอ็กซีคิวไฟล์ PHP นอกเหนือไปจากไคลเอนท์ที่กำหนดไว้ได้ แม้ว่าจะมีการปรับค่าไฟล์คอนฟิกูเรชันเพื่อป้องกันไว้แล้วก็ตาม
3. จากการที่ PHP นั้นสนับสนุนการ upload ไฟล์ผ่านทางฟอร์มโดยการร้องขอด้วยวิธี POST ซึ่งส่วนหนึ่งของฟังก์ชันที่รองรับการทำงานในด้านนี้คือ `php_mime_split()` ซึ่งยังมีข้อบกพร่องอยู่โดยอาจจะถูกใช้โดยผู้บุกรุกในการส่งเอ็กซีคิวไฟล์ที่ต้องการภายในระบบได้ ซึ่งข้อบกพร่องนี้ไม่ได้พบเฉพาะใน PHP4 เท่านั้นแต่ยังพบในรุ่นก่อนหน้าด้วย ซึ่งในกรณีนี้จะต้องปิดการ upload ไฟล์ก่อนด้วยการเซตไฟล์ `php.ini` ให้ `file_uploads = off`

แนวทางแก้ไข

การแก้ไขปัญหานี้ สามารถทำได้โดยการดาวน์โหลดแพตช์ ได้ที่เว็บไซต์ <http://www.php.net> หรือทำการ update ให้เป็นเวอร์ชันที่สูงกว่า ซึ่งจะมีการแก้ปัญหาต่างๆ ที่ได้รับการรายงานเข้ามาเรียบร้อยแล้ว

5. ปัญหาที่เกิดจากการปรับแต่งค่าระบบไม่ถูกต้อง (Configuration)

เป็นปัญหาที่เกิดจากการปรับแต่งค่า Configuration ของระบบที่ไม่เหมาะสม โดยทั่วไปเมื่อมีการติดตั้งโปรแกรมแล้ว ผู้ใช้ส่วนใหญ่ไม่ค่อยจะไปปรับค่าต่างๆ นัก เนื่องจากเกรงว่าเมื่อไปปรับแต่งแล้วอาจทำให้ระบบมีปัญหา หรือเกิดข้อผิดพลาดได้ ดังนั้นจึงใช้ค่าดีฟอลต์ที่มากับโปรแกรมแทน ซึ่งในการใช้งานจริงๆ ควรที่จะมีการปรับแต่งให้ถูกต้อง เพื่อให้ระบบที่ทำงานนั้นมีประสิทธิภาพสูงสุด และเพื่อความปลอดภัยของระบบด้วย

แนวทางแก้ไข

เมื่อทำการติดตั้ง PHP ลงในเครื่องเรียบร้อยแล้ว จะมีไฟล์อยู่ไฟล์หนึ่งคือ `php.ini` ซึ่งเป็นไฟล์ที่ใช้สำหรับปรับแต่งค่า Configuration ต่างๆ ทั้งหมด ภายในระบบ โดยเมื่อติดตั้งเรียบร้อยแล้ว จะต้องมีการปรับแต่งค่าให้เหมาะสม ซึ่งการปรับแต่งที่ควรพิจารณาเนื่องจากจะมีผลต่อความปลอดภัยของการใช้สคริปต์ในเซิร์ฟเวอร์ มีดังนี้ คือ

1. ปรับค่าให้ `register_globals = off`

เพื่อป้องกันไม่ให้มีการสร้างตัวแปร โกลบอลจากอินพุตของผู้ใช้ขึ้นมา เช่น เมื่อผู้ใช้มีการส่งตัวแปร `hello` ผ่านแบบฟอร์มโดย Post Method แล้ว เราจะไม่สามารถเรียกใช้ตัวแปรผ่านทาง Shell ได้ แต่สามารถเรียกใช้ได้โดย `HTTP_POST_VARS['hello']` เท่านั้น

2. ปรับค่าให้ `safe_mode = on`

เพื่อป้องกันและไม่อนุญาตให้ทำการ `execute` ไฟล์ต่างๆ ภายในระบบ เช่น การใช้ฟังก์ชัน `exec()` รวมไปถึงการป้องกันการเข้าไปเรียกใช้ไฟล์ที่ไม่ได้รับอนุญาตได้

3. ปรับค่าให้มีการใช้ `open_basedir`

เพื่อป้องกันการดำเนินการเกี่ยวกับไฟล์ใด ในไดเรกทอรีอื่น หรือนอกเหนือจากไดเรกทอรีที่กำหนดไว้

4. ปรับค่าให้ `display_errors = off` และ `log_errors = on`

เป็นการป้องกันไม่ให้แสดงข้อความผิดพลาดออกมา ซึ่งถ้าหากมีการแสดงข้อมูลที่ผิดพลาดออกมา ระหว่างการเยี่ยมชมเว็บไซต์นั้น จะทำให้สามารถทราบชื่อไฟล์ ฟังก์ชันที่ใช้ และบรรทัดที่มีข้อผิดพลาด อันจะก่อให้เกิดปัญหาในการบูกักต่อไป

5. การใช้ `include` ไฟล์ที่ถูกต้อง โดยทั่วไป มักจะมีการใช้ `include` ไฟล์สำหรับการติดต่อกับฐานข้อมูล และรวมฟังก์ชันที่สำคัญต่างๆ ที่ต้องการนำมาใช้ โดยส่วนที่ใช้ในการติดต่อฐาน

ข้อมูลนั้นจะมีการระบุชื่อโฮส username และ password ที่ได้ และโดยส่วนใหญ่ผู้ใช้งานมักจะทำการ save ไฟล์ include เป็น .inc ซึ่งการเซฟเป็นนามสกุลนี้ ถือได้ว่าไม่ปลอดภัย เพราะถ้าผู้ใช้พิมพ์ชื่อไฟล์ include นี้ก็จะเห็นเป็น plain text ทั้งหมดที่เก็บไว้ได้ การแก้ไขปัญหาเรื่อง include ไฟล์ สามารถทำได้โดย save เป็นไฟล์ .inc.php

6. การซ่อนไฟล์ PHP ภาษาที่นิยมใช้ในการเขียนสคริปต์ในปัจจุบันไม่ว่าจะเป็น Perl, PHP, ASP, JSP ฯลฯ ซึ่งในปัจจุบันแต่ละภาษาอาจจะมีจุดอ่อนและวิธีการโจมตีที่แตกต่างกันบ้าง โดยเมื่อผู้บุกรุกได้เห็นนามสกุลไฟล์ที่แสดงในเว็บเบราว์เซอร์ เขาก็จะรู้ทันทีว่าใช้ภาษาอะไรเขียน และควรโจมตีแบบไหน ดังนั้นวิธีการป้องกันอย่างหนึ่งคือการเปลี่ยนนามสกุลไฟล์ PHP นั้นให้เป็นนามสกุลอื่นอะไรก็ได้ โดยอาจจะเปลี่ยนเป็น .asp, .ce, .isag หรือ .html ก็ได้ โดยวิธีการเปลี่ยนนั้น ทำได้ดังนี้

1. ในไฟล์ php.ini ให้เซต `expose_php = off`
2. ทำการเปลี่ยนไฟล์ configuration ของ Apache ให้เป็นนามสกุลไฟล์ที่ต้องการ เช่น `AddType application/x-httpd-php .asp .py .pl` ในที่นี้จะเปลี่ยนจาก .php ให้เป็น .asp หรือ .py หรือ .pl

7. ในไฟล์คอนฟิกของ php นั้น ถ้ามีการกำหนดให้ `sql.safe_mode = on` จะหมายถึงว่า Username และ Password ที่ใช้ในการติดต่อกับฐานข้อมูลนั้นเป็นของผู้ที่ติดตั้ง PHP ในระบบ โดยถ้ามีการใช้ Username อื่นจะไม่สามารถ connect ฐานข้อมูลได้ ดังนั้นในกรณีที่มิใช่อื่นๆ ในระบบด้วยก็สามารถเซตให้เป็น off ได้

6.4 MySQL

MySQL เป็น SQL (Structured Queries Language) database server ที่เป็นที่ยอมรับมากในปัจจุบัน เหมาะสำหรับแอปพลิเคชันที่มีขนาดเล็กและปานกลาง สนับสนุน standard SQL (ANSI) นอกจากนี้ยังสามารถใช้ได้หลายแพลตฟอร์มของคอมพิวเตอร์

ข้อดีของ MySQL

1. มีประสิทธิภาพสูงและทำงานได้รวดเร็วเมื่อเปรียบเทียบกับฐานข้อมูลชนิดอื่นๆ
2. เป็น open source license ซึ่งไม่ต้องเสียค่าใช้จ่ายใดๆ และสามารถดาวน์โหลดได้ฟรี
3. สามารถติดตั้งและใช้งานได้ง่ายและรองรับในหลายแพลตฟอร์ม ไม่ว่าจะเป็นวินโดวส์, ยูนิกซ์
4. มีฟังก์ชันที่ช่วยอำนวยความสะดวกในการทำงานมากมายและรองรับการทำงานแบบ multi-threaded

6.4.1 ระบบตรวจสอบสิทธิ์การใช้งาน

หน้าที่หลักของระบบตรวจสอบสิทธิ์ คือ การตรวจสอบสิทธิ์ของผู้ใช้ที่เชื่อมต่อ MySQL server เช่น การใช้คำสั่ง query พื้นฐาน เช่น select insert update delete นอกจากนั้นระบบตรวจสอบสิทธิ์ยังใช้ในการจำกัดการใช้สิทธิ์ของผู้ใช้ที่เป็น anonymous การเพิ่มสิทธิ์การใช้ฟังก์ชันพิเศษ เช่น LOAD DATA INFILE และการบริหารระบบ MySQL

เมื่อมีการใช้คำสั่ง GRANT เพื่อกำหนดสิทธิ์การใช้งานให้กับผู้ใช้โดย ROOT นั้น คำสั่งนี้จะมีผลต่อฐานข้อมูลชื่อ mysql ซึ่งข้อมูลเกี่ยวกับสิทธิ์การใช้งานภายในฐานข้อมูลทั้งหมดจะเก็บอยู่ใน 5 ตารางที่อยู่ในฐานข้อมูลนี้ โดยในการกำหนดสิทธิ์การใช้งานให้กับผู้ใช้นี้ ควรระมัดระวังเกี่ยวกับการให้ผู้ใช้เข้าถึงฐานข้อมูล mysql เนื่องจากเป็นฐานข้อมูลที่กำหนดข้อมูลเกี่ยวกับสิทธิ์การใช้งานทั้งหมด โดยภายในฐานข้อมูลนี้มีทั้งหมด 5 ตาราง ดังนี้

User	กำหนดว่าผู้ใช้คนไหนสามารถเชื่อมต่อเข้ามายัง MySQL Server ได้บ้าง
Db	กำหนดว่าฐานข้อมูลไหนที่ผู้ใช้สามารถใช้งานได้
Host	กำหนดว่าผู้ใช้สามารถเชื่อมต่อเข้ามาผ่านทางโฮสต์ไหนได้บ้าง
table_priv	กำหนดตารางในฐานข้อมูลว่าผู้ใช้สามารถใช้งานตารางใดได้บ้าง
Columns_priv	กำหนดคอลัมน์ในตารางว่าผู้ใช้สามารถใช้งานคอลัมน์ใดได้บ้าง

ตารางที่สำคัญที่ต้องพิจารณาทุกครั้งในการกำหนดสิทธิ์การใช้งานมี 3 ตารางดังนี้

ตาราง user

ใช้ในการกำหนดสิทธิ์ว่าผู้ใช้ใดสามารถใช้งานไคลเอนต์ได้จากโฮสต์ใด และสามารถทำอะไรได้บ้าง โดยมีฟิลด์ต่าง ๆ ดังนี้

- Host - สามารถติดต่อจากไคลเอนต์ใดได้บ้าง
- User – ผู้ใช้คนใดสามารถติดต่อได้
- Select_priv – สามารถใช้ คำสั่ง query SELECT ได้หรือไม่ (Y/N)
- Insert_priv – สามารถใช้คำสั่ง query INSERT ได้หรือไม่ (Y/N)
- Update_priv – สามารถใช้คำสั่ง query UPDATE ได้หรือไม่ (Y/N)
- Delete_priv – สามารถใช้คำสั่ง query DELETE ได้หรือไม่ (Y/N)
- Create_priv – สามารถใช้คำสั่ง query CREATE ได้หรือไม่ (Y/N)
- Drop_priv – สามารถใช้คำสั่ง query DROP ได้หรือไม่ (Y/N)
- Reload_priv - สามารถใช้คำสั่ง query RELOAD ได้หรือไม่ (Y/N)
- Shutdown_priv - สามารถใช้คำสั่ง query SHUTDOWN MySQL Server ได้หรือไม่ (Y/N)
- Process_list_priv – สามารถสั่ง list คำสั่งที่กำลังประมวลผลบนเซิร์ฟเวอร์ได้หรือไม่ (Y/N)
- File_priv – ผู้ใช้สามารถเขียนไฟล์ลงบนเซิร์ฟเวอร์ได้หรือไม่ (Y/N)

ตาราง db

ใช้ในการกำหนดสิทธิ์ว่าฐานข้อมูลใดสามารถเข้าใช้ได้โดยผู้ใด จากไคลเอนต์ใดและสามารถทำอะไรได้บ้าง โดยมีฟิลด์ต่าง ๆ ดังนี้

- Host – สามารถติดต่อจากไคลเอนต์ใดได้บ้าง
- Db – สามารถใช้ฐานข้อมูลอะไรได้
- User – ผู้ใช้คนใดสามารถติดต่อได้
- Select_priv – สามารถใช้ คำสั่ง query SELECT ได้หรือไม่ (Y/N)
- Insert_priv – สามารถใช้คำสั่ง query INSERT ได้หรือไม่ (Y/N)
- Update_priv – สามารถใช้คำสั่ง query UPDATE ได้หรือไม่ (Y/N)
- Delete_priv – สามารถใช้คำสั่ง query DELETE ได้หรือไม่ (Y/N)
- Create_priv – สามารถใช้คำสั่ง query CREATE ได้หรือไม่ (Y/N)
- Drop_priv – สามารถใช้คำสั่ง query DROP ได้หรือไม่ (Y/N)

ตาราง host

ใช้ในการกำหนดสิทธิ์ว่าโฮสต์ใดมีสิทธิ์ใช้งานฐานข้อมูลใดบ้าง โดยมีฟิลด์ดังนี้

- Host – สามารถติดต่อจากไคลเอนต์ใดได้บ้าง
- Db – สามารถใช้ฐานข้อมูลอะไรได้
- Select_priv – สามารถใช้ คำสั่ง query SELECT ได้หรือไม่ (Y/N)
- Insert_priv – สามารถใช้คำสั่ง query INSERT ได้หรือไม่ (Y/N)
- Update_priv – สามารถใช้คำสั่ง query UPDATE ได้หรือไม่ (Y/N)
- Delete_priv – สามารถใช้คำสั่ง query DELETE ได้หรือไม่ (Y/N)
- Create_priv – สามารถใช้คำสั่ง query CREATE ได้หรือไม่ (Y/N)
- Drop_priv – สามารถใช้คำสั่ง query DROP ได้หรือไม่ (Y/N)

ตัวอย่างการใช้งานคำสั่ง GRANT โดยจะเป็นการเพิ่มผู้ใช้ชื่อ j1 และรหัสผ่านคือ jk12CE45 เพื่อสามารถใช้งานคำสั่ง select, insert, delete และ update ได้ทุกๆ ตารางในฐานข้อมูล books ดังนี้

```
mysql > grant select, insert, delete, update
> on books.*
> to j1 identified by 'jk12CE45';
```


6.4.2 การใช้งาน MySQL ให้มีความปลอดภัย

1. เมื่อเราติดตั้ง MySQL เสร็จแล้ว MySQL จะสร้างฐานข้อมูลและตารางขึ้นมาให้อัตโนมัติส่วนหนึ่ง ซึ่งเป็นตารางเกี่ยวกับการกำหนดสิทธิ์การใช้งาน จะมีตารางชื่อ user อยู่ในฐานข้อมูลชื่อ mysql ในตาราง user นี้จะเก็บ username พาสเวิร์ด และสิทธิ์ที่ผู้ใช้แต่ละคนมี ซึ่งจะมี row ที่ข้อมูลในฟิลด์ password ว่างอยู่ ฉะนั้นเราจะต้องลบ row นี้ทิ้งไปโดยใช้คำสั่งดังนี้ คือ

```
mysql> DELETE FROM user
      -> WHERE Host='localhost' AND User='';

mysql> QUIT

mysql> FLUSH PRIVILEGES;    //สั่งให้เซิร์ฟเวอร์ได้อ่านข้อมูลจากตารางต่าง ๆ อีกครั้งเพราะ
                           //เราได้เปลี่ยนแปลงข้อมูลในตาราง user
```

2. ตั้งพาสเวิร์ดสำหรับผู้ใช้ที่เป็น root โดยใช้คำสั่งดังนี้ คือ

```
shell> mysql -u root mysql
mysql> UPDATE user
      -> SET Password=PASSWORD('new_password')
      -> WHERE user='root';

mysql> FLUSH PRIVILEGES;
```

ไม่ควรนำเอาค่าศัพท์ที่พบในดิกชันนารีมาตั้งเป็นพาสเวิร์ดเพราะจะทำให้ง่ายต่อการเดาหรือการแกะรหัสพาสเวิร์ด

3. การกำหนดสิทธิ์การใช้งานของผู้ใช้

เราไม่ควรกำหนดสิทธิ์การใช้งานให้กับผู้ใช้นักเกินความจำเป็น โดยเฉพาะอย่างยิ่งไม่ควรกำหนดให้ผู้ใช้ที่ไม่ใช่ root มีสิทธิ์ใช้งานในตาราง user ในฐานข้อมูล mysql เพราะว่าถ้าเขาได้รับสิทธิ์ในตาราง user เขาก็จะมีสิทธิ์ในทุกฐานข้อมูลและทุกตาราง ฉะนั้นควรกำหนดให้เฉพาะที่เขาจำเป็นต้องใช้เท่านั้น เช่น เราจะกำหนดให้นาย top ซึ่งใช้ password='Mhall4589' มีสิทธิ์ในการ insert delete update select ข้อมูลเฉพาะในตาราง friend ซึ่งอยู่ในฐานข้อมูล sample เท่านั้น เราก็บอกให้ root จัดการให้ โดย root จะใช้คำสั่งดังนี้


```

shell> mysql -u root -p          //root ทำการ login เข้าใช้งาน mysql
Enter password : *****       //พาสเวิร์ดของ root )
Mysql > GRANT INSERT, UPDATE, DELETE, SELECT
Mysql > ON Sample.friend
Mysql > TO tum@isag16           //ระบุให้ใช้ เครื่อง isag16 ในการติดต่อ MySQL server)
Mysql > IDENTIFIED BY 'Mhall4589'; //ใช้ password = Mhall4589)

```

เมื่อทำคำสั่งดังกล่าวเสร็จแล้ว ข้อมูลจะถูก insert ลงไปในตาราง user ซึ่งอยู่ในฐานข้อมูล mysql โดยมีค่าในคอลัมน์ต่าง ๆ ดังนี้

Host = isag16, User = top, Password = 59590f3c0b8a6e43 (Mhall4589 ที่ถูกเข้ารหัสแล้ว) ส่วนสิทธิ์การใช้งานตาราง user นี้มีค่าเป็น 'N' หมดเลย

4. ก่อนที่จะเริ่มการใช้งาน MySQL นั้นจะต้องใช้คำสั่ง mysqld ก่อนเพื่อให้ Server นั้นไปโหลดข้อมูลจากตารางที่กำหนดสิทธิ์ของผู้ใช้ เพื่อจะจำกัดสิทธิ์ของผู้ใช้ตามที่กำหนดไว้ได้ โดยออพชันของ mysqld ซึ่งเกี่ยวข้องกับความปลอดภัยที่ควรพิจารณามีดังนี้

-- skip-show-database

เมื่อใช้ออพชันนี้แล้วจะทำให้ใช้คำสั่ง SHOW DATABASES ไม่ได้ ทำให้ผู้ใช้นั้นไม่มีสิทธิ์ที่จะดูรายชื่อฐานข้อมูลทั้งหมดที่มีอยู่

-- safe-show-database

เมื่อใช้ออพชันนี้แล้วจะทำให้ใช้คำสั่ง SHOW DATABASES ได้ โดยถ้าผู้ใช้นั้นมีสิทธิ์ในการใช้งานฐานข้อมูลใดก็จะเห็นเฉพาะฐานข้อมูลนั้น

5. โดยปกติแล้วงานบางอย่าง จะต้องมีการรับข้อมูลบางส่วนจากผู้ใช้งาน และนำเอาข้อมูลนั้นไปทำการ Query ในฐานข้อมูล ยกตัวอย่าง เช่น เว็บไซต์ที่อนุญาตให้ผู้ใช้สามารถเปลี่ยน Username ได้เมื่อทำการ Login เข้าไปในระบบแล้ว โดยจะมีฟอร์มให้ผู้ใช้ทำการพิมพ์ชื่อ Username ใหม่ที่ต้องการเปลี่ยน เมื่อมีการส่งค่าตัวแปรที่เก็บ Username ใหม่ที่ต้องการเปลี่ยน เพื่อนำไป Query ในฐานข้อมูล คำสั่งที่ใช้ในการ Query นั้นจะอยู่ในรูปแบบดังนี้

```
UPDATE users SET Name='Susurname' WHERE ID=1;
```

ในคำสั่งนี้ จะเห็นได้ว่าการส่งตัวแปร Susurname ที่เป็นชื่อใหม่ที่ต้องการเปลี่ยนเข้ามา สมมุติว่าผู้ใช้ต้องการเปลี่ยน Username ใหม่เป็น Hanako ก็จะพิมพ์ชื่อนี้ลงในฟอร์ม ซึ่งคำสั่งที่ใช้ในการ Query จะได้ดังนี้


```
UPDATE users SET Name='Hanako' WHERE ID=1;
```

จะเห็นว่าตัวอย่างข้างต้นนั้นสามารถทำงานได้โดยไม่มีปัญหาอะไร แต่ในกรณีที่ผู้ใช้พิมพ์ Username ที่ต้องการเปลี่ยนใหม่เป็น Hanako',Admin='1 คำสั่งที่ได้ใน Query จะเป็นดังนี้

```
UPDATE users SET Name='Hanako',Admin='1' WHERE ID=1;
```

จะเห็นว่าผู้ใช้นอกจากจะสามารถเปลี่ยน Username ของตัวเองได้แล้ว ยังสามารถทำการเปลี่ยนค่า Admin ที่อยู่ในฐานข้อมูลได้ด้วย ซึ่งจะเป็นผลเสียในกรณีที่ฐานข้อมูลของเว็บไซต์มีใช้เซตค่า Admin ให้เป็น 1 หรือ 0 เพื่อตรวจสอบว่า Username นี้เป็น Admin หรือไม่

การป้องกันปัญหาในการ Query ข้อมูลที่ผิดพลาดลงในฐานข้อมูลสามารถทำได้โดยใส่เครื่องหมาย apostrophe ครอบค่าคงที่ที่เป็นตัวเลข เช่น ใช้ SELECT * FROM table WHERE ID = '234' แทนที่จะใช้ SELECT * FROM table WHERE ID = 234 MySQL นั้นจะแปลง string เหล่านี้ไปเป็นตัวเลข และเอาเครื่องหมายที่ไม่ใช่ตัวเลขออก นอกจากนี้ควรใช้ฟังก์ชัน addslashes() ทุกครั้ง โดยตัวแปรทุกตัวที่รับมาจากผู้ใช้ก่อนที่จะเอาไปใช้กับคำสั่ง Query ในฐานข้อมูลควรจะใช้ฟังก์ชัน addslashes() กับข้อมูลนั้นก่อน ซึ่งฟังก์ชัน addslashes() นี้จะทำการใส่เครื่องหมาย \ (backslash) เข้าไปในตัวแปรนั้น ถ้าภายในตัวแปรมีการใช้เครื่องหมาย '(single quote)', ""(double quote), \ (backslash), และค่าที่เป็น Null เพื่อเป็นการแบ่งแยกระหว่างคำสั่งที่ใช้ในการ Query ในฐานข้อมูลกับค่าของตัวแปรที่รับเข้ามาให้ชัดเจน และไม่ให้เกิดปัญหา

6. การเข้ารหัส รหัสผ่านของผู้ใช้หรือข้อมูลอื่นๆ ที่สำคัญที่เก็บอยู่ในฐานข้อมูลนั้น จะต้องมีการเข้ารหัสไว้เพื่อป้องกันการลักลอบอ่าน สำหรับผู้ที่อาจจะกรุกเข้ามาภายในระบบได้ โดยการเข้ารหัสอาจจะใช้ hash function เป็นต้น
7. การกำหนดผู้ใช้หลายระดับในการเข้าถึงข้อมูลในฐานข้อมูล โดยปกตินั้นภายในเว็บไซต์ที่มีการติดต่อกับฐานข้อมูลนั้น จะมีข้อมูลอยู่หลายๆ ส่วนด้วยกัน เช่น ในส่วนของผู้ใช้ทั่วไปที่มีสิทธิ์ดูข้อมูลต่างๆ ภายในเว็บไซต์ได้อย่างเดียว ในที่นี้เราจะใช้ชื่อและรหัสผ่านที่ใช้ connect เข้าฐานข้อมูลที่มีสิทธิ์เฉพาะแค่ดูข้อมูลในฐานข้อมูลเท่านั้น ในกรณีนี้ถ้าหากว่ามีผู้บุกรุกเข้ามาและได้ชื่อและรหัสผ่านนี้ไป ก็จะสามารถ select ดูข้อมูลในฐานข้อมูลเท่านั้น แต่เนื้อหาบางส่วนของเว็บไซต์เช่น ส่วนของสมาชิกภายในที่สามารถเพิ่มหรือแก้ไขข้อมูลภายในเว็บไซต์ได้ เราก็จะใช้ชื่อและรหัสผ่านที่ใช้ connect เข้าฐานข้อมูลที่มีสิทธิ์ในการ update ข้อมูลในฐานข้อมูลด้วย เป็นต้น

บทที่ 7

ปัญหาความปลอดภัยที่เว็บไคลเอนต์

7.1 ปัญหาความปลอดภัยที่ตัวเว็บไคลเอนต์

ปัญหาที่เกิดขึ้นที่ตัวเว็บไคลเอนต์ส่วนใหญ่เป็นปัญหาที่ส่งผลกระทบต่อข้อมูล ไฟล์และรีซอร์ส (Resource) ต่างๆ ที่เก็บไว้ในฮาร์ดดิสก์ของผู้ใช้บริการ นอกจากนี้ยังรวมไปถึงการถูกละเมิดความเป็นส่วนตัวของผู้ใช้ในขณะทำการติดต่อกับเว็บไซด์อยู่ทั้งจากแฮกเกอร์และแม้กระทั่งเจ้าของเว็บไซด์เอง สามารถแบ่งพิจารณาเป็นกรณีย่อยๆ ได้ดังนี้

- Client Side Script

ไคลเอนต์ไซด์สคริปต์เป็นสคริปต์ที่เขียนขึ้นด้วยภาษาโปรแกรมมิ่งต่างๆ กัน โดยที่สคริปต์เหล่านั้นจะถูกทำการประมวลผลที่ฝั่งของไคลเอนต์ ซึ่งในที่นี้ก็คือเว็บเบราว์เซอร์ โดยไคลเอนต์ไซด์สคริปต์จะช่วยเพิ่มความสามารถในการติดต่อและโต้ตอบกันของข้อมูลหรือการทำงานต่างๆ ระหว่างผู้ใช้และเว็บเซิร์ฟเวอร์ผ่านสคริปต์ที่ทำงานโดยเว็บเบราว์เซอร์ ตัวอย่างเช่น สคริปต์ที่นำมาใช้ในการตรวจสอบอินพุตของผู้ใช้ในการกรอกข้อมูลลงบนแบบฟอร์มผ่านเว็บเบราว์เซอร์ ซึ่งสคริปต์ที่ทำงานนี้สามารถทำการตรวจสอบว่ามีความผิดพลาดเกิดขึ้นหรือไม่ก่อนที่จะส่งข้อมูลกลับไปยังเซิร์ฟเวอร์

ไคลเอนต์ไซด์สคริปต์ที่จะกล่าวถึงต่อไปนี้เป็นสคริปต์ที่ใช้กันอย่างแพร่หลายในปัจจุบัน คือ จาวาสคริปต์ (Java Script) และวีบีสคริปต์ (VB Script) ซึ่งจะมีโครงสร้างการทำงานและข้อดีข้อเสียที่ต่างกันออกไป ซึ่งจะกล่าวถึงต่อไปในภายหลัง

โดยปกติแล้วสคริปต์จะถูกรันโดยเว็บเบราว์เซอร์ ซึ่งเว็บเบราว์เซอร์แต่ละตัวก็จะรองรับการทำงานของแต่ละสคริปต์ต่างกันออกไป นั่นก็就会有มีความเสี่ยงทางด้านความปลอดภัยที่ต่างกันด้วย ซึ่งผู้ใช้จะต้องตระหนักถึงผลกระทบเหล่านั้นต่อทรัพยากรบนเครื่องของผู้ใช้ด้วย

จาวาสคริปต์ (Java Script)

จาวาสคริปต์เป็นภาษาชุดใหม่สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ตที่กำลังได้รับความนิยมอย่างสูง เราสามารถทำการเขียนโปรแกรมจาวาสคริปต์เพิ่มเข้าไปในเว็บเพจเพื่อใช้ประโยชน์สำหรับงานด้านต่างๆ ทั้งการคำนวณ การแสดงผล การรับ-ส่งข้อมูล และที่สำคัญคือสามารถโต้ตอบกับผู้ใช้ได้อย่างทันทีทันใด นอกจากนี้ยังมีความสามารถด้านอื่นๆ อีกหลายประการที่ช่วยสร้างความน่าสนใจให้แก่เว็บเพจของเราเป็นอย่างมาก

จาวาสคริปต์ถือกำเนิดมาจากบริษัท Netscape Communication ถูกเปิดตัวครั้งแรกในชื่อ LiveScript เพื่อใช้สร้างเว็บเพจที่สามารถแลกเปลี่ยนข้อมูลกับเซิร์ฟเวอร์แบบ LiveWare ได้โดยจาวาสคริปต์เป็น “ภาษาสคริปต์เชิงวัตถุ” ที่ช่วยให้เราสามารถควบคุมเว็บเพจได้อย่างง่ายดาย สามารถทำงานข้ามแพลตฟอร์มได้ ทำหน้าที่เป็นตัวประสานระหว่างเอกสาร HTML, จาวาแอปเพล็ต (Java Applet) และเว็บเบราว์เซอร์ทั้งทางฝั่งไคลเอนต์และฝั่งเซิร์ฟเวอร์ และสามารถใช้กับเทคโนโลยีอื่นๆ เช่น ActiveX,

CGI, Plug-in, จาวาและอื่นๆ ซึ่งช่วยให้เว็บเพจที่บรรจุจาวาสคริปต์มีความน่าสนใจและสมบูรณ์แบบมากกว่าเว็บเพจทั่วไป

ตัวอย่างต่อไปนี้เป็นส่วนหนึ่งของโปรแกรมที่เขียนขึ้นจากจาวาสคริปต์และผลของการรันสคริปต์ ดังนี้

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

จากตัวอย่างเป็นโปรแกรมที่ถูกเขียนขึ้นด้วยจาวาสคริปต์ซึ่งสังเกตได้จากการประกาศการใช้ในคำสั่ง LANGUAGE โดยผลของการรันจาวาสคริปต์จะเป็นการนำเอาวันที่ที่ได้ไปเก็บไว้ในตัวแปร d ด้วยฟังก์ชัน new Date ซึ่งผลของการรันเป็นดังนี้

Today's date is Tue Mar 13 13:20:21 UTC+0700 2001

ตัวอย่างต่อไปจะเป็นตัวอย่างตรวจสอบอินพุต ซึ่งในที่นี้ก็คือความยาวของชื่อของผู้ใช้ (UserName) ว่ามีจำนวนตัวอักษรน้อยกว่า 5 ตัวหรือไม่และตรวจสอบว่าชื่อที่พิมพ์เข้าไบนั้นตรงตามกฎหรือถูกต้องหรือไม่และในส่วนสุดท้ายก็จะเป็นการตรวจสอบรหัสผ่าน (Password) ดังนี้

จากตัวอย่าง โปรแกรมที่แสดงข้างต้นจะทำการรันในส่วนของเมธอด (Method) **thisPage_onbeforeserverevent** และตรวจสอบอินพุตของผู้ใช้ในแบบฟอร์ม ซึ่งถ้าไม่ผ่านการตรวจสอบค่าของ **thisPage.cancelEvent** จะมีค่าเป็น **true** ซึ่งจะทำให้เว็บเบราว์เซอร์ยกเลิกการส่งข้อมูลไปยังเซิร์ฟเวอร์และทำการแจ้งต่อผู้ใช้ เป็นต้น


```

<SCRIPT LANGUAGE = javascript>

<!--

// perform client-side input validation
// before returning to server
function thisPage_onbeforeserverevent ( ) {

    // check for minimum username length
    if (document.thisForm.Textbox1.value.length < 5) {
        alert ("UserName must be at least 5 characters!");
        thisPage.cancelEvent=true;
        return;
    }

    // check for valid username
    if (document.thisForm.Textbox1.value!="VIUSER") {
        alert ("Invalid UserName!");
        thisPage.cancelEvent=true;
        return;
    }

    // check for valid password
    if (document.thisForm.Textbox2.value!="SECRET") {
        alert ("Invalid Password!");
        thisPage.cancelEvent=true;
        return;
    }

}

// -->

</SCRIPT>

```

ความปลอดภัยของจาวาสคริปต์ (Java Script Security)

โปรแกรมที่เขียนขึ้นโดยจาวาสคริปต์จะมีความปลอดภัยมากกว่าโปรแกรมที่เขียนขึ้นโดยจาวาหรือโปรแกรมอื่นๆ เนื่องจากเหตุผลดังนี้

- ไม่มีเมธอดใดๆ ของจาวาสคริปต์ที่สามารถเข้าถึงระบบไฟล์ของเครื่องคอมพิวเตอร์ที่ฝังไคลเอนต์ได้โดยตรง
- ไม่มีเมธอดใดๆ ของจาวาสคริปต์ที่จะสามารถทำการเปิดหรือสร้างการติดต่อสื่อสารไปยังเครื่องคอมพิวเตอร์บนเครือข่ายได้โดยตรง

แต่จาวาสคริปต์ก็เหมือนกับส่วนอื่นของเว็บ คือ ได้รับการพัฒนาและเปลี่ยนแปลงอยู่ตลอดเวลา ซึ่งยิ่งพัฒนา ก็จะยิ่งมีจุดอ่อนเพิ่มขึ้นเรื่อยๆ โดยปัญหาทางด้านความปลอดภัยของจาวาสคริปต์สามารถจำแนกได้ 3 ประการ คือ ปัญหาความปลอดภัยเนื่องจากการโจมตีแบบ DoS (Denial of Service Attack) ปัญหาความปลอดภัยจากการโจมตีแบบสปูฟิง (Spoofing Attack) และปัญหาที่เกิดจากการเข้าถึงไฟล์ได้โดยตรง (Direct File Access)

ปัญหาความปลอดภัยเนื่องจากการโจมตีแบบ DoS (Denial of Service Attack)

ปัญหาด้านความปลอดภัยที่สำคัญของจาวาสคริปต์ คือ ความยากในการป้องกันการโจมตีแบบ DoS เนื่องมาจากผู้ที่ทำการพัฒนาจาวาสคริปต์ไม่ได้คำนึงถึงความสำคัญหรือข้อบกพร่องจากการโจมตีในลักษณะนี้

การโจมตีแบบ DoS จะส่งผลกระทบต่อผู้ใช้เว็บด้วยเช่นกัน โดยโปรแกรมการโจมตีจะถูกฝัง (Embedded) เป็นส่วนหนึ่งของโปรแกรมจาวาสคริปต์

การโจมตีแบบ DoS จะมีลักษณะและผลของการโจมตีที่แตกต่างกันออกไป ในส่วนนี้จะกล่าวถึงชนิดของการโจมตีแบบ DoS ซึ่งจะเกี่ยวข้องกับเว็บเบราว์เซอร์ที่จะทำการรันสคริปต์ต่างๆ เหล่านั้นโดยจะกล่าวถึงเฉพาะ Netscape และ Internet Explorer ชนิดของการโจมตีแบบ DoS มีดังนี้

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

CPU and Stack Attack

โปรแกรมที่เขียนโดยจาวาสคริปต์สามารถที่จะทำให้เครื่องคอมพิวเตอร์หยุดการทำงานได้โดยโปรแกรมจะไปร้องขอการใช้งานของซีพียูและหน่วยความจำในปริมาณมากทำให้ซีพียูและหน่วยความจำไม่สามารถทำงานอื่นได้ ตัวอย่างโปรแกรมต่อไปนี้ เป็นโปรแกรมที่เขียนด้วยจาวาสคริปต์ซึ่งเป็นโปรแกรมที่ทำการคำนวณเลขทางคณิตศาสตร์ ซึ่งผลการทำงานจะใช้ทรัพยากรของซีพียูและหน่วยความจำจนหมด


```

<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;

            return 1;

        }

        for (I=0;I<100000; I++) {

            document.write("Fibonacci number "+I+" is "+fibonacci(I)+"<br>");

        }

    </script>

</body>

</html>

```

ในความเป็นจริงแล้วทั้ง Netscape และ Internet Explorer จะสามารถแก้ไขปัญหที่เกิดขึ้นได้แต่จะต่างกันออกไป โดย Netscape จะแสดงข้อความ (Error Message) คือ “Lengthy JavaScript” และข้อความให้ผู้ใช้ในการปิดโปรแกรม แต่สำหรับส่วนของ Internet Explorer จะทำการรันโปรแกรมและจะหยุดการทำงานเมื่อค่าของ Fibonacci เท่ากับ 22 ซึ่งจะทำให้เกิด Stack Overflow ขึ้น

Swap Space Attack

เป็นลักษณะของโปรแกรมที่เมื่อทำการรันโดย Netscape และ Internet Explorer แล้วพยายามที่จะจัดสรรหน่วยความจำจำนวนมากเพื่อรองรับการทำงาน ซึ่งเสมือนกับการเคลื่อนย้ายไฟล์ (Swap) ไปยังหน่วยความจำสำรอง (Hard Disk) ทั้งหมดภายในเครื่อง

โปรแกรมต่อไปนี้เป็นโปรแกรมที่เป็นการจู่โจมแบบ Swap Space

การโจมตีแบบ Swap Space Attack จะแตกต่างกับแบบ Stack Attack ซึ่งการโจมตีแบบ Swap Space นี้จะส่งผลกระทบต่อประสิทธิภาพของเว็บเบราว์เซอร์และทุกๆ โปรแกรมที่กำลังทำงานอยู่บนเครื่องคอมพิวเตอร์ นั่นก็เป็นเพราะว่าคอมพิวเตอร์ถูกสั่งให้ทำการเคลื่อนย้ายไฟล์ (Swap) ไปมาอยู่ตลอดเวลา ซึ่งในกระบวนการเคลื่อนย้ายไฟล์นี้จะทำให้หน่วยความจำสำรองและซีพียูไม่สามารถทำงานอื่นได้

```
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;
}
```

Window System Attack

จาวาสคริปต์ที่มีการดาวน์โหลดโค้ด (Code) ในการสร้างและจัดการหน้าต่าง (Window) บนเครื่องของผู้ใช้ การทำงานในส่วนของ GUI (Graphic User Interface) จะใช้ทรัพยากรของระบบบนเครื่องเป็นจำนวนมากโดยจะทำการสร้างหน้าต่างขึ้นมาจำนวนมาก ซึ่งทำให้ไม่สามารถทำงานอื่นได้บนเครื่องคอมพิวเตอร์ ตัวอย่างของการโจมตีชนิดนี้ เช่น การที่ผู้ใช้เข้าไปยังเว็บไซต์ www.bcl.net/~hpokindm/security/testbrowser.html เว็บเบราว์เซอร์จะทำการรันโปรแกรมในส่วนของจาวาสคริปต์ ซึ่งจะส่งผลให้เกิดการเปิดหน้าต่างใหม่ขึ้นมาเรื่อยๆ ทำให้ผู้ใช้ไม่สามารถหยุดยั้งได้จนกว่าทรัพยากรบนเครื่องจะถูกใช้หมดไป สิ่งที่ผู้ใช้สามารถทำได้คือการกด Ctrl+Atl+Delete ซึ่งจะส่งผลให้เครื่องหยุดและต้องเริ่มการทำงานใหม่

การป้องกันการโจมตีแบบ DoS

ในความเป็นจริงแล้วไม่มีวิธีใดในการป้องกันการโจมตีแบบ DoS ได้ 100 เปอร์เซ็นต์ แต่ผู้ใช้สามารถทำการป้องกันโดยการลดความเสี่ยงต่างๆ ที่เป็นสาเหตุการโจมตีแบบ DoS โดยสาเหตุส่วนใหญ่ของการโจมตีจะมาจากการที่โปรแกรมเอง ซึ่งแนวทางในการลดความเสี่ยงมีดังนี้

- จำกัดและติดตามการใช้ทรัพยากรภายในเครื่องเนื่องจากการรันโปรแกรมบนเครื่อง โดยทำการติดตามว่าโปรแกรมที่ทำการดาวน์โหลดหรือรันอยู่นั้นมีการใช้ซีพียูหรือหน่วยความจำมากน้อยแค่ไหน

- ทำการปรับค่าในบราวเซอร์ไม่ให้มีการใช้งานในส่วนของจาวาสคริปต์ (Disable) ทั้งของ Netscape และ Internet Explorer แต่ในกรณีนี้ผู้ใช้อาจต้องตระหนักว่าจะไม่สามารถทำการใช้งานบางอย่างหรือเปิดเว็บเพจได้ถ้าไม่มีการใช้งานจาวาสคริปต์

ปัญหาความปลอดภัยจากการโจมตีแบบสปูฟิง (Spoofing Attack)

โปรแกรมที่เขียนด้วยจาวาสคริปต์สามารถที่จะทำให้ผู้ใช้เครื่องสับสนได้เมื่อมีการเปิดเว็บเพจผ่านบราวเซอร์

Spoofing Browser Status with Java Script

โปรแกรมที่เขียนด้วยจาวาสคริปต์สามารถที่จะสร้างส่วนที่หลอกผู้ใช้นบนหน้าเว็บเพจ ซึ่งสามารถที่จะสร้างส่วนที่หลอกผู้ใช้ได้ เช่น

- จาวาสคริปต์สามารถสร้างข้อความปลอมขึ้นมาหลอกผู้ใช้ได้
- จาวาสคริปต์สามารถทำการหลอกผู้ใช้โดยการเปลี่ยน URL ใน Status Line ได้
- จาวาสคริปต์สามารถทำการซ่อนส่วนของ “Goto:” ของบราวเซอร์ได้และทำงานด้วยแบบฟอร์มอื่นที่สร้างขึ้นเองโดยจาวาสคริปต์

เช่น URL ที่ผู้ใช้อาจต้องการจะติดต่อ คือ <http://www.shopping.com/order-entry.html> แต่เมื่อผู้ใช้คลิกเพื่อเข้าไปยังเว็บไซต์ เว็บเพจที่แสดงจริงกลับเป็น URL ที่ <http://www.attacker.org/trapped.html> โดยทำการเขียนโปรแกรมจาวาสคริปต์ ดังนี้

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

ผู้ใช้งานส่วนใหญ่จะเชื่อถือและไว้วางใจในการรันโปรแกรมที่ทำการดาวน์โหลดจากเว็บไซต์ที่น่าเชื่อถือ (Well Trusted Domain) อย่างไรก็ตามก็มีหลายกลยุทธ์ในการหลอกล่อผู้ใช้งานผ่านเว็บเบราว์เซอร์ ในการดาวน์โหลดโปรแกรมจากโดเมนเนมหนึ่งแต่ในความเป็นจริงแล้วกลับเป็นการดาวน์โหลดจากโดเมนเนมอื่น ตัวอย่างเช่นผู้ใช้อาจต้องการดาวน์โหลดไฟล์ *SETUP.EXE* จากไมโครซอฟท์

- ผู้ใช้อาจจะคิดว่าไฟล์ที่กำลังทำการดาวน์โหลดจาก *MICROSOFT.COM* แต่ในความเป็นจริงแล้วดาวน์โหลดจาก *MICROSOFT.COM* (ตัวอักษร “O” เปลี่ยนเป็น “0”)
- ผู้ใช้อาจจะเห็นว่าไฟล์ที่กำลังดาวน์โหลดนั้นมาจาก *MICROSOFT.CO.FI* โดยที่ไม่รู้ว่าเป็นโดเมนเนมที่เกี่ยวข้องกับไมโครซอฟท์หรือไม่
- เว็บเบราว์เซอร์อาจจะแสดงเฉพาะส่วนต้นและส่วนท้ายแต่จะไม่แสดงอักขระตรงส่วนกลางของไฟล์ที่จะทำการดาวน์โหลด เช่น บราวเซอร์อาจแสดงแอดเดรส ดังนี้ <http://www>

.microsoft.co../setup.exe แต่ในความเป็นจริงแล้วทำการดาวน์โหลดจาก <http://www.microsoft.com.attacker.org/guests/users/hacker/setup.exe>

ตัวอย่างในการปลอมเว็บไซต์ขึ้นมาโดยสร้างเอกสาร HTML ขึ้นมาชื่อ c:\site_imperson.html โดยใช้ข้อมูลที่เราต้องการขึ้นมาเอง สามารถทำได้ดังโค้ดข้างล่างนี้

```
<html>
<head>
<title> Site Impersonation </title>
<script language = "JavaScript">
<!--

function openWin()
{

url = prompt("Enter a fully qualified domain name:");

win = document.open(url, "urlWin", "top=200, left=200, width=300, height = 250, resizable=yes,
location=yes, status=yes, toolbar=yes");

win.document.write("You entered the domain name of " + url + ", but I'd rather you see this:<br><br>
<h1>Hello World!</h1>");

}

-->
</script>
</head>
<body bgcolor="#FFFFFF" onLoad="openWin()">

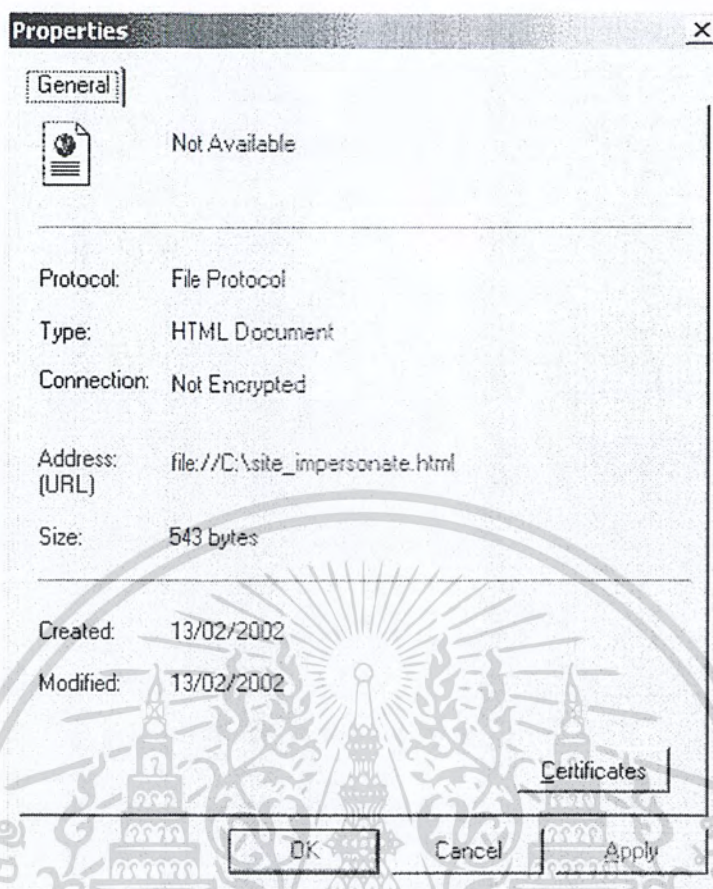
</body>
</html>
```


เมื่อเรียกเว็บเพจนี้ขึ้นมาทำงานและใส่ชื่อโดเมนแบบเดิมในหน้าต่างที่แสดงขึ้นมา สมมุติว่าได้ใส่ address ว่า <http://www.devarticles.com> แล้วสิ่งที่เกิดขึ้นเมื่อจาวาสคริปต์เปิดหน้าต่างขึ้นมาใหม่โดยจะเป็นดังรูปข้างล่างนี้



รูป 7-1 แสดงหน้าเว็บเพจที่ทำปลอมขึ้นมา

การป้องกันการโจมตีแบบสปูฟิง
 เพื่อความปลอดภัยแล้ว ควรที่จะเลื่อนเมาส์ให้อยู่ในหน้าต่างสกรีนนั้นแล้วคลิกเมาส์ปุ่มขวา แล้วเลือกที่ Properties แล้วดูว่า address ที่แสดงให้คุณนั้นใช่ address ที่เราได้กรอกไปไหม ข้างล่างนี้เป็นรูปของ จากโค้ดด้านบน



รูป 7-2 แสดง Properties ของหน้าเอกสารที่ได้ปลอมขึ้นมา

จะเห็นว่า address แสดงเป็นชื่อไฟล์ที่ของเว็บเพจที่เราสร้างขึ้นมา แทนที่จะเป็น address ของเว็บไซต์ที่เราได้กรอกเข้าไป ถ้าใช้ IE6 และต้องกรอกข้อมูลที่สำคัญ เช่น หมายเลขบัตรเครดิต แล้วก็ต้องตรวจสอบ address ของเว็บไซต์นั้นก่อนที่จะกรอกข้อมูลด้วย

ปัญหาที่เกิดจากการเข้าถึงไฟล์ได้โดยตรง (Direct File Access)

เป็นการขโมยเนื้อหาในไฟล์จากเครื่องของผู้ใช้ แล้วส่งกลับไปยังเว็บเซิร์ฟเวอร์ เพื่อประมวลผล และวิเคราะห์จาวาสคริปต์มีฟังก์ชัน document.open ใช้สำหรับเปิดเอกสาร HTML ขึ้นมาในหน้าต่างใหม่ ใน IE6 มีช่องโหว่ตรงที่สามารถแสดงเนื้อหาของไฟล์บนเครื่องของผู้ใช้ได้โดยใช้ฟังก์ชัน document.open ข้างล่างนี้เป็นโค้ดที่แสดงให้เห็นว่าสามารถนำเนื้อหาในไฟล์จากเครื่องมาแสดงในเว็บเบราว์เซอร์ได้ สร้างไฟล์ชื่อ c:\file_read.html แล้วใส่โค้ดดังข้างล่างนี้ลงไป

```
<html>
```

```
<head>
```

```
<title> Local File Reading </title>
```



```

<script language="JavaScript">
<!--
function readFile()
{
// What file do we want to read?
fileLoc = 'c:/winnt/setuplog.txt';
file = document.open(fileLoc, "fileWin", "top=5000, left=5000, width=1, height=1");
contents = file.document.body.innerText;
file.close();

document.write("<b>I can see the contents of " + fileLoc + ": </b><br><br>");
document.write("<textarea style='width:400; height:300'>" + contents + "</textarea>");
}
-->
</script>
</head>
<body bgcolor="#FFFFFF" onLoad="readFile()">
</body>
</html>

```

โค้ดจาวาสคริปต์ข้างบนนั้นเป็นการดึงเอาเนื้อหาในไฟล์ c:\winnt\setuplog.txt มาแสดงในกล่องข้อความของเอกสาร HTML โดยตำแหน่งของไฟล์ภายในเครื่องนั้นจะถูกเก็บไว้ในตัวแปร fileLoc และไฟล์นี้จะเป็นไฟล์อะไรก็ได้ตามที่ต้องการ แต่โค้ดนี้จะใช้ไฟล์ได้ 2 ประเภท คือ HTML และ เท็กซ์ไฟล์ และจะต้องอยู่ในเครื่องของผู้ใช้เท่านั้น ยิ่งไปกว่านั้นยังสามารถถอดเงื้อมมือของจาวาสคริปต์ในการควบคุมการส่ง (submit) ข้อมูลกลับไปยังเว็บเซิร์ฟเวอร์ระยะไกล โดยที่ผู้ใช้ที่เป็นเจ้าของเครื่อง (client) นั้นไม่รู้เรื่องเลยว่ามีเกิดอะไรขึ้น

การป้องกันปัญหาที่เกิดจากการเข้าถึงไฟล์ได้โดยตรง

วิธีที่ดีที่สุดในการป้องกันช่องโหว่นี้คือ การยกเลิกการทำงานของสคริปต์ที่ฝั่งไคลเอนต์ (client-side scripting) การปรับระดับของความปลอดภัยให้เป็นแบบปลอดภัยสูง (high) ตอนนี้ทางไมโครซอฟท์ก็ได้พยายามออกแพตช์ (patch) มาแก้ไขช่องโหว่ที่มีอยู่เหมือนกัน มันก็ขึ้นอยู่กับผู้ใช้แล้วว่า จะใช้ IE6 เป็นเว็บเบราว์เซอร์หรือไม่

วีบีสคริปต์ (VB Script)

วีบีสคริปต์ คือได้ว่าเป็นสับเซตของวิซวลเบสิก (Visual Basic) คือนำเอารูปแบบภาษาการเขียนโปรแกรมในแบบวิซวลเบสิกมาเขียนคำสั่งให้แอปพลิเคชันสำหรับอินเทอร์เน็ตหรือสั่งงานให้เบราว์เซอร์ทำงานได้ตามต้องการ โดยจะเพิ่มความน่าสนใจให้กับแอปพลิเคชันที่สร้างขึ้น

แม้วีบีสคริปต์จะมีความสามารถจำกัดกว่าวิซวลเบสิกในหลายเรื่อง แต่ด้วยความเรียบง่ายของวีบีสคริปต์ ทำให้ทางไมโครซอฟท์เลือกนำมาใช้ในการเพิ่มความสามารถให้แอปพลิเคชันสำหรับอินเทอร์เน็ต รวมทั้งเทคโนโลยีการจัดการเว็บเพจและเว็บเซิร์ฟเวอร์อย่าง ASP (Active Server Page) องค์ประกอบของแอปพลิเคชันที่ใช้งานวีบีสคริปต์

แอปพลิเคชันที่นำความสามารถของวีบีสคริปต์ไปใช้งานมักจะประกอบด้วย

◆ คำสั่งของภาษา HTML

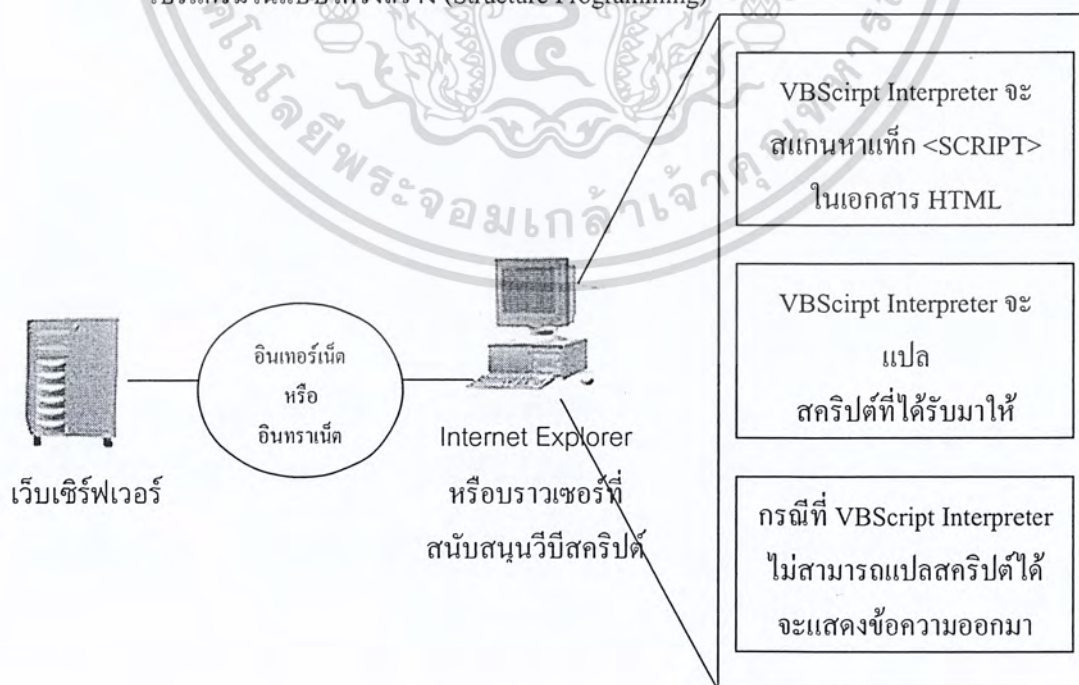
จะเป็นส่วนที่บรรจุข้อความในภาษา HTML ให้ทุกเบราว์เซอร์เข้าใจและแสดงผลได้อย่างตรงกัน

◆ VBScript Delimiter

เป็นสิ่งที่ใช้แยกวีบีสคริปต์ออกจากภาษา HTML โดยจะใช้แท็ก <SCRIPT> ครอบส่วนที่เป็นคำสั่งในวีบีสคริปต์ และมักจะใช้แท็ก Comment (<!-- กับ -->) ครอบส่วนที่เป็นวีบีสคริปต์ภายในอีกชั้นหนึ่ง ซึ่งจะมีข้อดีคือ ถ้าแอปพลิเคชันนี้ถูกเรียกใช้งานโดยเบราว์เซอร์ที่ไม่สนับสนุนวีบีสคริปต์ก็ยังใช้งานได้อย่างต่อเนื่อง

◆ VBScript Subroutine or Function

คือความสามารถในการสั่งให้ทำโปรแกรมย่อยของวีบีสคริปต์ ซึ่งจะเหมือนกับการเขียนโปรแกรมในแบบโครงสร้าง (Structure Programming)



รูปที่ 7-3 แสดงโครงสร้างการทำงานของแอปพลิเคชันที่ใช้วีบีสคริปต์จัดการ

ในกรณีของฟังก์ชัน (Function) เราสามารถเขียนขึ้นมาใช้เองได้หรืออาจจะใช้งานฟังก์ชันที่วีบีสคริปต์เตรียมไว้ก็ได้

◆ VBScript Built-in Object

วีบีสคริปต์เองมีออบเจกต์อยู่จำนวนหนึ่งซึ่งพร้อมถูกนำมาใช้งานร่วมกับคำสั่งในวีบีสคริปต์ เช่น Dictionary Object, File System Object, Err Object เป็นต้น ซึ่งออบเจกต์เหล่านี้ต่างมีความสามารถในตัวและทำให้วีบีสคริปต์มีความน่าใช้งานมากขึ้น

ตัวอย่างต่อไปนี้เป็นโปรแกรมอย่างง่ายที่เขียนขึ้นโดยวีบีสคริปต์และ HTML ซึ่งเป็นการสร้างปุ่มข้อความ (Message Box) ที่มีข้อความว่า “Test” อยู่บนปุ่ม ดังนี้

```
<HTML>

<HEAD><TITLE> A Simple First Page </TITLE>
<SCRIPT LANGUAGE="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>
```

วีบีสคริปต์กับการเขียนพัฒนาแอปพลิเคชันแบบ Event-Driven

สำหรับวีบีสคริปต์แล้ว เรียกได้ว่าเป็นวิซวลเบสิกฉบับย่อซึ่งสิ่งหนึ่งที่คิดมาจากวิซวลเบสิกและทำการพัฒนาแอปพลิเคชันสำหรับอินเทอร์เน็ตมีประสิทธิภาพขึ้น ก็คือการใช้หลักของ Event-Driven (เหตุการณ์พาไป) มาใช้ ทำให้แอปพลิเคชันอินเทอร์เน็ตที่สร้างขึ้นสามารถตอบสนองต่อเหตุการณ์ที่จะเกิดขึ้น (จากการใช้งานของผู้ใช้งาน) ได้อย่างเหมาะสมโดยใช้รูปแบบภาษาเช่นเดียวกับวิซวลเบสิก

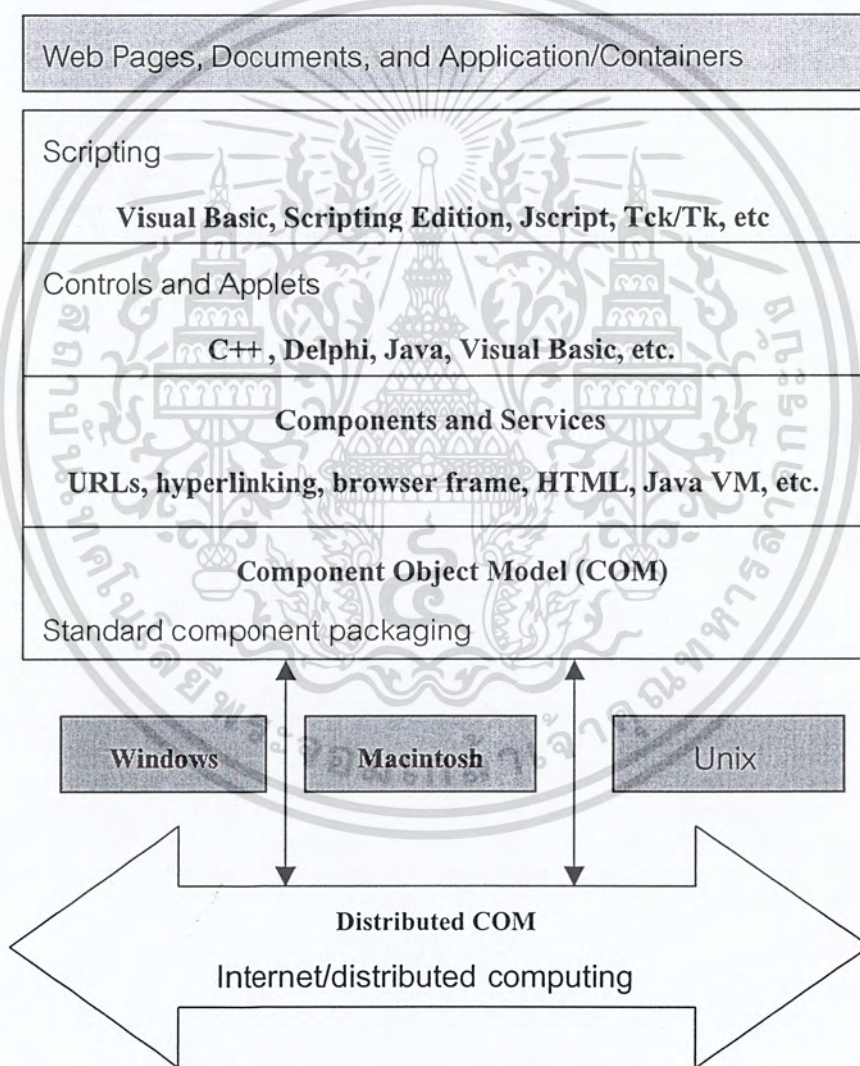
ความปลอดภัยของวีบีสคริปต์ (VBScript Security)

เนื่องจากว่าวีบีสคริปต์มีการนำเอาเทคโนโลยีของ ActiveX มาใช้ในการทำงานร่วมกัน ทำให้การทำงานต่างๆ ของเว็บเพจที่เขียนขึ้นด้วยโปรแกรมวีบีสคริปต์มีความปลอดภัยเนื่องมาจากการใช้ ActiveX ด้วย เช่น เทคนิคของการ Signing Authority และการ Authenticode ซึ่งเป็นเทคนิคที่ใช้เฉพาะในส่วนของ ActiveX โดยการทำงานและความปลอดภัยของ ActiveX จะกล่าวในภายหลัง

- Component ที่รันบนฝั่งไคลเอนต์

ActiveX

ส่วนประกอบของเทคโนโลยีของ ActiveX แสดงได้ดังรูปที่ 7-2



รูปที่ 7-4 แสดงส่วนต่างๆ ของเทคโนโลยี ActiveX

ActiveX Scripting

Netscape ได้เพิ่มลักษณะเด่นที่มีประโยชน์มากมายให้แก่ Navigator เรียกว่าจาวาสคริปต์ โดยปกติแล้วเว็บเบราว์เซอร์จะมีตัวตีความที่ทำหน้าที่เป็นตัวแปลสำหรับสคริปต์นี้ ส่วนการใช้สคริปต์

ActiveX ที่มีสำหรับ Internet Explorer นั้นก็คล้ายคลึงกันแต่จะเป็นบริการที่มากกว่า กล่าวคือ ทำให้มีความสามารถเพิ่มด้านสคริปต์และความสามารถในการลักษณะเดียวกับ OLE ให้กับโปรแกรมได้ สคริปต์ของ ActiveX จะจัดหาแพลตฟอร์มสำหรับผู้พัฒนาภาษา แต่สำหรับตัวภาษาสคริปต์ ไวยากรณ์ และรูปแบบการทำงานนั้นจะหลากหลายแตกต่างกันไปขึ้นอยู่กับการออกแบบของนักพัฒนา

นอกจากนี้ ActiveX ยังเป็นคอมโพเนนต์ที่ช่วยในการทำเว็บเพจให้มีคุณภาพดี น่าใช้และช่วยให้ใช้งานง่ายยิ่งขึ้น เช่น สามารถเพิ่มเมนูป๊อปอัพ (Pop-up menu) พื้นที่ทำการคลิกเมาส์ซึ่งจะช่วยให้ผู้เลือกรายการที่ต้องการได้อย่างรวดเร็ว นอกจากนี้ยังสามารถใส่ ActiveX เพิ่มเข้าไปยังภาพเคลื่อนไหวหรือข้อมูลจากโปรแกรมอื่นๆ เช่น Microsoft Excel หรือ Microsoft Word ที่อยู่ในเว็บเพจได้

ActiveX Scripting มีองค์ประกอบ 2 ชนิด คือ

1. ActiveX Scripting Hosts

ActiveX Scripting Host จะจัดหาแพลตฟอร์มไว้เพื่อให้ ActiveX Scripting Engines ทำงาน จะเห็นได้ว่า ActiveX scripting host หลักที่สำคัญคือ Microsoft Internet Explorer อย่างไรก็ตามยังมี scripting host อื่นที่มีศักยภาพภายใต้ ActiveX อีกเช่นกัน ได้แก่

- เว็บเบราว์เซอร์อื่นๆ (เช่น Netscape)
- Internet Authoring Tools
- เว็บเซิร์ฟเวอร์ (Server-based Scripting)

2. ActiveX Scripting Engines

โดยพื้นฐานแล้ว ActiveX scripting engine ก็คือภาษาที่สามารถปฏิบัติการบน ActiveX Scripting Host ได้ ActiveX Scripting Engine ชนิดแรกได้แก่ วิบีสคริปต์ (เป็นส่วนหนึ่งของ Visual Basic) อย่างไรก็ตามยังมี environment อื่นๆ อีก เช่น Perl, Lisp, Delphi, Scheme

ActiveX Component

เนื่องจากได้มีการปรับปรุงเทคโนโลยีในการผลิตเบราว์เซอร์ให้ดีขึ้น ทำให้สามารถใช้ ActiveX Component ซึ่งเป็นเทคโนโลยีที่มีพื้นฐานมาจาก Microsoft's Component Object Model (COM) ในการเพิ่มแพลตฟอร์มต่างๆ ให้กับเบราว์เซอร์ได้อย่างอิสระ ขอบข่ายของ ActiveX Component จะเป็นได้ตั้งแต่ Fancy Controls อย่างเช่น spinner และ slider ไปจนถึง nonvisual components จึงทำให้เกิดการเข้าถึงข้อมูลและเพิ่มประสิทธิภาพให้กับการรับส่งไปรษณีย์อิเล็กทรอนิกส์ (E-mail) โดยคอมโพเนนต์เหล่านี้จะทำให้เว็บเพจใน Internet Explorer มีความน่าสนใจและมีฟังก์ชันใช้งานได้หลากหลาย แต่จะไม่สามารถใช้ได้ในระบบที่ไม่สนับสนุน ActiveX

ActiveX Documents

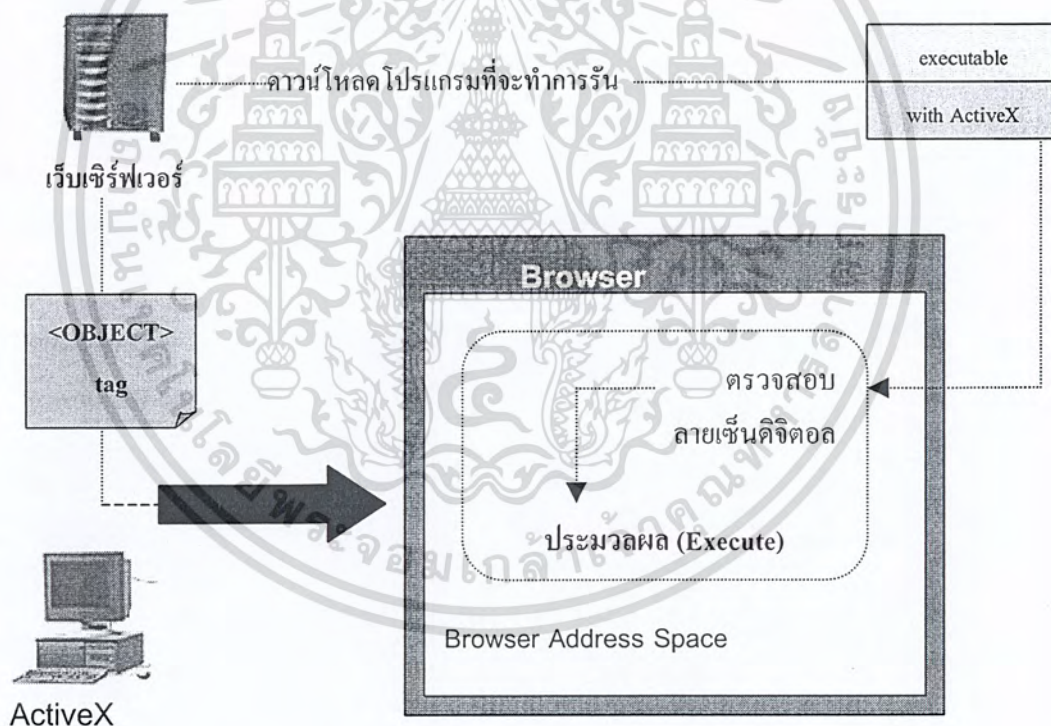
ActiveX Documents เป็นชุดเครื่องมือ (tools) ที่ใช้ในการสร้าง dynamic content ซึ่งจะสนับสนุนการสร้าง ActiveX Control ได้ดีพอๆ กับ developer ชนิดอื่นๆ เช่น Visual Basic 5.0 และ ActiveX

Documents เป็น Software Object ที่ถูกดาวน์โหลดและทำงานภายใน ActiveX Container อย่างเช่น Internet Explorer ได้ ดังนั้น ActiveX Documents จึงนับว่าเป็นเครื่องมือที่ทุนแรงนักพัฒนาได้อย่างมากในการสร้างแอปพลิเคชันทางอินเทอร์เน็ต โดย ActiveX Documents จะกำหนดส่วนที่ใช้ในการเข้าถึง (access) คุณสมบัติ (properties) เมธอด (method) ต่างๆ ที่สร้างขึ้นภายใน developer ไว้ในรูปแบบที่สามารถทำการดาวน์โหลดได้ ผลก็คือทำให้ผู้ใช้สามารถดูและแก้ไขเอกสารที่ไม่ใช่ HTML ผ่านทางเบราว์เซอร์ได้

ActiveX Control

ActiveX Control ได้ถูกแนะนำเพื่อการรวมสองส่วนที่แยกกันวิวัฒนาการของเทคโนโลยีคอมพิวเตอร์ Custom Control กับส่วนที่คิดใหม่ของไอดีพื้นฐานเกี่ยวกับ OLE และ OOP

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



รูปที่ 7-3 แสดงส่วนของ ActiveX Control ที่ถูกดาวน์โหลดและทำการรันที่เว็บเบราว์เซอร์

ความปลอดภัยของ ActiveX (ActiveX Security)

เทคโนโลยี ActiveX ที่ใช้บนเครื่องนั้นสามารถนำมาใช้บนอินเทอร์เน็ตได้ในเว็บเพจและสามารถที่จะนำเอาคอนโทรลต่างๆ หรือแอปพลิเคชันทั้งอันที่เป็น ActiveX โปรแกรมไปวางไว้บนเว็บ

เพจได้ เนื่องจากมีคอนโทรล ActiveX มากมายให้ใช้บนอินเทอร์เน็ต และเมื่อต้องการที่จะใช้ก็ต้องการการดาวน์โหลดแล้วคอนโทรลก็จะลงทะเบียนกับระบบปฏิบัติการ ซึ่งจะสามารถถูกใช้โดยแอปพลิเคชันอื่นๆ ที่อยู่บนระบบได้ด้วย นอกเหนือจากใช้บนเบราว์เซอร์ กระบวนการนี้เรียกว่า การดาวน์โหลดคอมโพเนนต์ (Component Downloading)

อย่างไรก็ตามยังมีข้อจำกัดบางประการและคำถามสำคัญที่มีอยู่ซึ่งต้องถามเพื่อป้องกันในแต่ละการดาวน์โหลดคอนโทรล ActiveX มายังเครื่อง

- คอนโทรล ActiveX บางตัวไม่สามารถนำมาใช้หรือเผยแพร่ได้ถ้าไม่มีใบอนุญาตจากนักพัฒนาของคอนโทรลนั้น ใบอนุญาตที่ทำให้สามารถใช้คอนโทรลนั้นในเว็บเพจของตัวเอง ผู้ใช้สามารถดาวน์โหลดคอนโทรลและลงทะเบียนนั้นกับระบบของผู้ใช้ แต่ผู้ใช้ไม่อาจضمنได้ถ้าไม่มีใบอนุญาตสำหรับมัน เหตุผลอื่นที่คอนโทรล ActiveX ต้องมีใบอนุญาตเพื่อเป็นทางหนึ่งของการแจ้งว่าคอนโทรลนั้นมีความปลอดภัยที่จะใช้บนเว็บเพจ ตัวอย่างของคอนโทรลมักเป็นเช่นนั้นซึ่งทำงานภายใต้เบราว์เซอร์ในการสร้างไฟล์ของผู้ใช้และการทำงานอื่นๆ ที่คล้ายกัน
- จะมีการเซ็นรหัส (Code Signing) และมีการใช้เทคโนโลยีในการตรวจสอบบุคคล (Authentication Technology) มาใช้ในการยืนยันการสร้างคอนโทรลว่ามีความปลอดภัยและมาจากแหล่งที่น่าเชื่อถือ ซึ่งจะช่วยในการป้องกันคอนโทรลที่อาจบรรจุไวรัสได้

ปัญหาด้านความปลอดภัยของ ActiveX (ActiveX Security Hole)

ปัญหาด้านความปลอดภัยที่สำคัญของ ActiveX คือความยากในการที่จะทำการควบคุมและติดตามการทำงานต่างๆ ในส่วนของคอนโทรลที่มีการทำงานอย่างสลับซับซ้อน ตัวอย่างเช่น การส่งข้อมูลข่าวสารที่เป็นความลับเกี่ยวกับการติดตั้ง (configuration information) จากคอมพิวเตอร์ของผู้ใช้ไปยังเซิร์ฟเวอร์ผ่านอินเทอร์เน็ต ซึ่งอาจจะทำให้มีไวรัสเกิดภายในวง LAN เป็นต้น

นอกจากนี้ที่ตัว ActiveX Control ยังได้รับการรายงานถึงปัญหาทางด้านความปลอดภัยที่เกิดขึ้นจากการทำงานของตัวคอนโทรลทั้งสอง คือ scriptlet.typeilib และ eyedog ซึ่งผลเสียโดยรวมที่เกิดขึ้นคือเจ้าของเว็บไซต์สามารถที่จะทำการเขียนเว็บเพจที่จะกระทำการการบางสิ่งบางอย่างกับเครื่องคอมพิวเตอร์ของผู้ที่เรียกใช้เว็บเพจนั้น ซึ่งตัวคอนโทรลทั้งสองถูกออกแบบมาเพื่อใช้ทำงานดังนี้

- Scriptlet.typeilib ถูกออกแบบมาเพื่อให้ นักพัฒนาโปรแกรมใช้สร้าง Type Libraries สำหรับ Window Script Component (WSC) ซึ่ง Type Libraries เหล่านี้ถูกใช้โดยเครื่องมือที่ใช้ในการ พัฒนาทั้งหลาย เช่น Microsoft Visual InterDev เพื่อสร้างคุณสมบัติบางอย่าง เช่น ตัวช่วยเหลือในการทำงาน (Tool-tip help)
- Eyedog นั้นถูกออกแบบมาเพื่อให้โปรแกรมที่ใช้ในการวิเคราะห์ระบบใช้ในการรวบรวมข้อมูลเกี่ยวกับฮาร์ดแวร์ทั้งหลายบนเครื่องที่โปรแกรมเหล่านั้นกำลังทำงานอยู่

ปัญหาก็คือตัวคอนโทรลทั้งสองถูกกำหนดไว้ว่าปลอดภัยในภาษาสคริปต์ (Save for Scripting) นั้นหมายความว่าตัวคอนโทรลนี้ได้รับการรับรองจากทางไมโครซอฟท์แล้วว่าไม่สามารถทำอันตรายบน

เครื่องคอมพิวเตอร์ของผู้ใช้ได้ ดังนั้นจึงสามารถทำงานโดยไม่ต้องร้องขอความยินยอมจากผู้ใช้ก่อน สิ่งนี้เป็นข้อผิดพลาดของตัวคอนโทรลทั้งสอง เพราะมันสามารถทำให้เกิดการกระทำที่เป็นอันตรายได้ ดังนี้

- Scriptlet.typelib ทำให้เว็บเพจสามารถแก้ไขหรือลบไฟล์บนเครื่องคอมพิวเตอร์ของผู้ใช้ได้โดยการแก้ไขไฟล์ของระบบ โดยเจ้าของเว็บไซต์สามารถสั่งคำสั่งหรือโปรแกรมใดที่เขาต้องการให้ทำงานขึ้นมาก็ได้
- Eyedog ทำให้เว็บเพจสามารถเก็บข้อมูลต่างๆ จากเครื่องคอมพิวเตอร์ของผู้ใช้ได้ เช่น ค่าที่กำหนดในรีจิสทรี ชื่อยูสเซอร์เนม การกำหนดค่าต่างๆ ของฮาร์ดแวร์ เป็นต้น และทำการส่งข้อมูลเหล่านี้กลับไปยังเว็บเซิร์ฟเวอร์

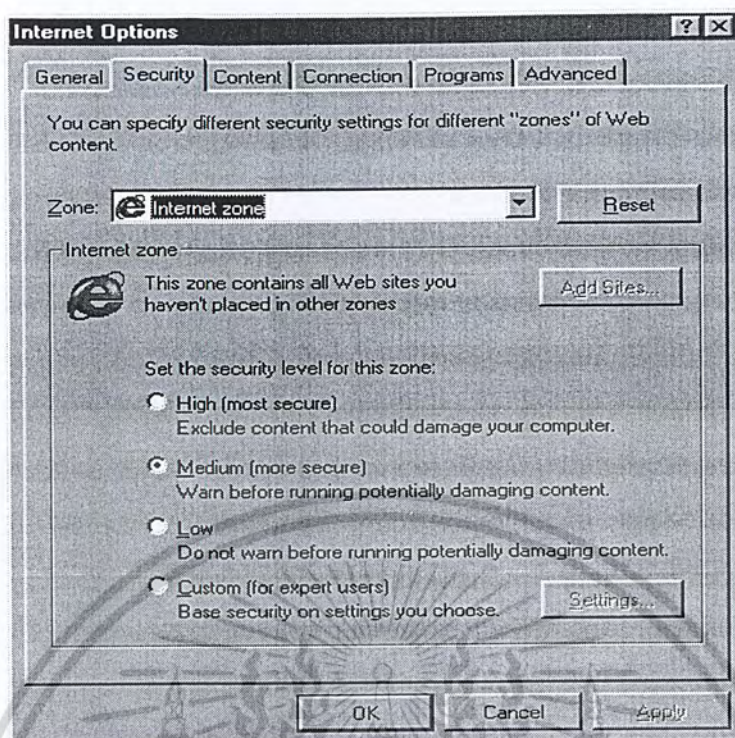
การป้องกันปัญหาทางด้านความปลอดภัยของ ActiveX

สำหรับในกรณีของปัญหาที่เกิดจากตัวคอนโทรลทั้งสองนั้นสามารถทำการแก้ไขได้โดยทำการดาวน์โหลดแพช โดยจะใช้วิธีที่ต่างกันในการแก้ไขปัญหาของตัวคอนโทรลทั้งสองนี้

- สำหรับ scriptlet.typelib นั้น แพชจะยกเลิกการกำหนดว่า ปลอดภัยสำหรับใช้ในภาษาสคริปต์ อย่างไรก็ตามตัวคอนโทรลนี้ยังสามารถใช้ใน Internet Explorer ได้ แต่จะทำงานได้ก็ต่อเมื่อผู้ใช้นิยามเท่านั้น
- สำหรับ eyedog นั้น แพชจะมีการตั้งค่าบิตที่ชื่อว่า Kill Bit ซึ่งจะทำให้มันไม่สามารถใช้ใน Internet Explorer ได้อีกต่อไป

ส่วนในการป้องกันที่ปลอดภัยนั้น ก็สามารถทำได้โดยการไม่อนุญาตให้มีการใช้งานในส่วนของ ActiveX ของบราวเซอร์ ซึ่งในที่นี้จะกล่าวถึงการเซต Internet Explorer ดังนี้

1. เลือก Internet Options และเลือกที่ Security ใน Internet Security
2. เลือก High Security ซึ่งจะเป็นการที่จะไม่ให้มีการรัน ActiveX ใดๆ เลยบนบราวเซอร์
3. เลือก Medium Security เพื่อทำการเตือนผู้ใช้ก่อนทำการดาวน์โหลดและรัน ActiveX ซึ่งถ้าผู้ใช้นิยามให้มีการรัน ActiveX จะต้องดูในส่วนของ Authenticode Certificate ให้รอบคอบ
4. เลือก Low Security ในกรณีที่ยินยอมให้มีการรัน ActiveX ได้โดยไม่มีการเตือน ดังรูป



รูปที่ 7-6 แสดงการตั้งค่าการทำงานในส่วนของ ActiveX

จาวาแอปเพล็ต (Java Applet)

แอปเพล็ต (Applet) คือ โปรแกรมขนาดเล็กที่สร้างขึ้นด้วยภาษาจาวา สามารถถูกเรียกจากในเอกสาร HTML ให้ทำงานเป็นส่วนหนึ่งของเว็บเพจนั้น กล่าวอีกอย่างได้ว่าเป็นโปรแกรมที่ถูกส่งไปกับเอกสาร HTML เพื่อไปทำงานภายใต้บราวเซอร์ ที่มีตัวแปลจาวา (Java Interpreter) อยู่บนเครื่องที่รับเอกสาร HTML นั้นไป สังเกตว่าโปรแกรมภาษาจาวาที่เป็นแอปเพล็ตจะทำงานภายใต้ตัวแปลจาวาโดยไม่ต้องอาศัยโปรแกรมอื่น (จึงเรียกว่า Standalone โปรแกรม) และสามารถควบคุมการดำเนินงานของตัวเอง แต่โปรแกรมที่เป็นแอปเพล็ตจะทำงานภายใต้บราวเซอร์โดยไม่สามารถควบคุมการดำเนินงานของตัวเองได้หมด

โปรแกรมของแอปเพล็ตเมื่อถูกคอมไพล์แล้วจะได้หนึ่งไฟล์เป็น .class แต่จะไม่สามารถทำงานได้โดยตัวแปลจาวา (Java Interpreter: java.exe) การเรียกใช้แอปเพล็ตทำงานนั้นต้องทำจากภายในเอกสาร HTML โดยใช้แท็กของแอปเพล็ต (Applet Tag) และแอปเพล็ตนั้นจะทำงานในสภาวะแวดล้อมของบราวเซอร์ อย่างเช่น Netscape Navigator, Microsoft Internet Explorer หรือ Hot Java ซึ่งทั้งหมดเป็นบราวเซอร์ที่มีตัวแปลจาวาอยู่ภายในหรือโปรแกรมใช้สำหรับทดสอบแอปเพล็ตที่มากับ JDK คือ appletviewer แต่ appletviewer ไม่ได้ถูกสร้างขึ้นเพื่อเป็นบราวเซอร์ มันจึงไม่จัดการกับข้อความส่วนอื่นๆ ในเอกสาร HTML และจะทำงานเฉพาะส่วนของแอปเพล็ตเท่านั้น อีกทั้งยังไม่สามารถติดต่อกับระบบเครือข่าย (Network) ได้เหมือนกับบราวเซอร์ทั่วไป

การทำงานและความปลอดภัยของจาวาแอปเพล็ต (Java Applet Security)

เมื่อผู้ใช้ส่งคำร้องขอข้อมูลเว็บเพจไปยังเว็บเซิร์ฟเวอร์ หากภายในเว็บเพจนั้นมีรหัสคำสั่ง `<APPLET>..</APPLET>` สำหรับกำหนดจาวาแอปเพล็ตที่ต้องการนำมาใช้งาน เซิร์ฟเวอร์ก็จะเพียงทำหน้าที่ส่งข้อมูลต่างๆ เกี่ยวกับแอปเพล็ตนั้นไปยังเครื่องคอมพิวเตอร์ของผู้ใช้ จากนั้นเว็บเบราว์เซอร์ที่เครื่องคอมพิวเตอร์ของผู้ใช้จะนำไบต์โค้ดที่ได้นั้นไปเรียกใช้งานต่อไป

ส่วนควบคุมการทำงานของจาวา (Java Runtime)

โดยปกติจะฝังตัวอยู่ในโปรแกรมเว็บเบราว์เซอร์เพื่อทำหน้าที่ควบคุมการทำงานของโปรแกรมจาวาที่ส่งมาจากเซิร์ฟเวอร์ และเมื่อเว็บเบราว์เซอร์นั้นพบเว็บเพจที่มีรหัสคำสั่ง `<APPLET>` ข้อมูลต่างๆ ที่ใช้ในการทำงานร่วมกับแอปเพล็ตนั้นจะถูกส่งคืนมายังเบราว์เซอร์ของผู้ใช้ให้โดยอัตโนมัติ การจัดการกับจาวาแอปเพล็ตจะเรียบร้อยเมื่อเราได้เห็นภาพและได้ยินเสียงตามที่ได้มีการกำหนดมา

เนื่องจากจาวาถูกสร้างขึ้นมาเพื่อใช้งานร่วมกับระบบเครือข่าย จึงถูกเน้นให้มีระบบรักษาความปลอดภัยที่รัดกุมเพื่อป้องกันอันตรายที่อาจเกิดขึ้นจากไวรัสคอมพิวเตอร์ หรือโปรแกรมคอมพิวเตอร์ที่มีโอกาสทำงานผิดพลาดได้ ภาษาจาวาจึงไม่มีคุณสมบัติในการเข้าถึงหน่วยความจำของระบบในระดับลึก เพื่อตัดปัญหาให้ภาษาจาวาเป็นอันตรายต่อระบบการทำงานของเครื่องคอมพิวเตอร์ที่เรียกใช้มันได้ และเนื่องจากจาวาต้องทำงานผ่านเว็บเบราว์เซอร์ ดังนั้นเว็บเบราว์เซอร์จึงทำหน้าที่เป็นผู้ตรวจทานรหัสของคำสั่งจาวาก่อนว่า ไม่มีการเขียนรหัสคำสั่งที่เป็นอันตรายต่อระบบ จากนั้นจึงผ่านให้ Java Class Loader เพื่อสั่งให้โปรแกรมทำงานต่อไป ฉะนั้นจาวาจึงเหมาะกับการใช้งานที่ต้องการความปลอดภัยสูงผ่านระบบเน็ตเวิร์กหรือทำธุรกิจในระบบอินเทอร์เน็ตโดยจาวาได้มีการแบ่งระบบการรักษาความปลอดภัยไว้ 3 ระดับดังนี้

- การตรวจสอบความถูกต้องของรหัสคำสั่ง
- การกำหนดไฟล์ที่สามารถใช้งานได้
- การตรวจสอบขณะเรียกใช้งาน

ข้อจำกัดของแอปเพล็ต (Applet Restrictions)

เนื่องจากแอปเพล็ตเป็นโปรแกรมที่จะถูกแจกจ่ายไปในระบบเครือข่ายเพื่อไปทำงานบนเครื่องของผู้ใช้ซึ่งอยู่ในที่ต่างๆ กัน ดังนั้นจึงต้องมีข้อจำกัดบางอย่างในการสร้างแอปเพล็ตเพื่อไม่ให้ผู้ประสงค์ร้ายบางคนสามารถสร้างแอปเพล็ตที่ทำลายโปรแกรมหรือข้อมูลบนเครื่องคอมพิวเตอร์ของผู้ที่รับแอปเพล็ตนั้นไปทำงานเช่น ทำการลบไฟล์ หรือส่งจดหมายอิเล็กทรอนิกส์ปลอมโดยใช้ชื่อของเจ้าของเครื่องหรือแอบใช้เครื่องคอมพิวเตอร์นั้นเป็น Remote File Server เป็นต้น

โดยปกติแล้วเบราว์เซอร์จะถือว่าแอปเพล็ตที่รับมาจากเครื่องอื่นในระบบเครือข่ายเป็นโปรแกรมที่ไม่ปลอดภัยไว้ก่อนซึ่งเรียกว่า Untrusted Applets และจะให้แอปเพล็ตเหล่านี้ทำงานภายใต้สภาพแวดล้อมที่ปลอดภัย ซึ่งมีการติดตั้ง Security Manager ให้คอยตรวจสอบการทำงานของแอปเพล็ตและไม่ยอมให้มีการทำงานที่ไม่สมควรเกิดขึ้น ทำให้ Untrusted applets ถูกจำกัดด้วยกฎเกณฑ์ต่อไปนี้

1. ห้ามออกคำสั่งกับระบบไฟล์ของเครื่องที่รับแอปเพล็ตนั้นไปทำงาน โดยมีรายละเอียดดังนี้

- ห้ามเขียน อ่าน หรือลบไฟล์
 - ห้ามสร้างไคเรกทอรีหรือควาในไคเรกทอรีใดมีอะไรบ้าง
 - ห้ามตรวจสอบว่ามีไฟล์หรือไคเรกทอรีชื่อหนึ่งในเครื่องนั้นหรือไม่
 - ห้ามตรวจสอบว่าชื่อหนึ่งเป็นไฟล์หรือไคเรกทอรี
 - ห้ามขอข้อมูล ขนาด หรือเวลาที่ทำการเปลี่ยนแปลงไฟล์นั้น
 - ห้ามเปลี่ยนชื่อไฟล์หรือเปลี่ยนแปลงค่า file descriptor
2. ห้ามออกคำสั่งแก่ระบบเครือข่ายดังนี้
- ห้ามติดต่อไปยังคอมพิวเตอร์เครื่องอื่นที่ไม่ใช่เครื่องที่ให้แอปเพล็ตนั้นมา
 - ห้าม listen หรือ accept ใน port ใดๆ ที่มีเลขน้อยกว่าหรือเท่ากับ 1024
 - ห้ามใช้ multicast sockets
 - ห้ามสร้าง instance ของคลาส SocketImplFactory, URL StreamHandlerFactory, หรือ ContentHandlerFactory
3. ห้ามไม่ให้ใช้ฟังก์ชันบางประเภทของระบบ เช่น
- ห้ามออกจากตัวแปลจาวา (Java Interpreter) ด้วยคำสั่ง System.exit() หรือ Runtime.exit()
 - ห้ามสร้างและทำงานโปรเซสใหม่ด้วยคำสั่ง Runtime.exec()
 - ห้ามทำการโหลด native code ด้วยคำสั่ง load() หรือ loadLibrary() ของคลาส Runtime หรือคลาส System
4. ไม่สามารถใช้งานคลาสบางตัวของ AWT ได้เหมือนปกติทั่วไป
- ทุกๆ หน้าต่าง (Window) ที่ถูกสร้างขึ้นด้วยแอปเพล็ตจะมีข้อความใดๆ ให้เห็นว่าไม่ปลอดภัย (Insecure)
 - ห้ามสร้าง print job
 - ห้ามอ้างถึง System Clipboard หรือ AWT event queue
5. ห้ามอ้างถึงค่าคุณสมบัติ (Properties) ของระบบ เช่น ห้ามเรียกคำสั่ง System.getProperty() และไม่สามารถแก้ไขหรือเพิ่มค่าคุณสมบัติใน System Properties List แต่แอปเพล็ตสามารถเรียกคำสั่ง System.getProperty() อ่านค่าคุณสมบัติแต่ละตัวได้ โดยมีข้อแม้ว่าต้องได้รับอนุญาตจาก appletviewer โดยทั่วไป appletviewer อนุญาตให้สามารถอ่านค่าคุณสมบัติ (Properties) ต่างๆ เหล่านี้ได้ java.version, java.class.version, java.vendor, java.vendor.url, os.name, os.version, os.arch, file.separator, path.separator, line.separator
6. ห้ามสร้างหรืออ้างถึงเธรด (Thread) หรือกลุ่มของเธรด (Thread Group) ที่ไม่ได้อยู่ในกลุ่มของเธรดของแอปเพล็ตนั้น
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 ยกเว้นแต่คลาสนั้นถูกโหลดมาจากเซิร์ฟเวอร์เดียวกันกับที่ให้อาพเพล็ตนั้นมา
9. มีข้อจำกัดในการใช้ java.security package
- ห้ามจัดการกับ Security Identities ไม่ว่ากรณีใดๆ
 - ห้ามอ่านหรือเขียนค่า Security Properties
 - ห้าม list, lookup, insert, หรือ remove security providers
 - ห้ามสร้าง instance ของคลาส ClassLoader หรือ Security Manager ขึ้นมาใหม่

ในกรณีใช้ appletviewer ทำงาน แอพเพล็ตที่อยู่ในเครื่องเดียวกัน ข้อจำกัดบางอย่างข้างต้นอาจถูกละเว้น แต่แอพเพล็ตที่ถูกโหลดจะยังคงผ่านกระบวนการรักษาความปลอดภัยที่ appletviewer นั้นสร้างขึ้น

ปัญหาด้านความปลอดภัยของเว็บเบราว์เซอร์

เป็นที่ทราบกันดีแล้วว่าในเครือข่ายอินเทอร์เน็ตนั้น โครงสร้างการทำงานนั้นจะประกอบไปด้วยเซิร์ฟเวอร์และไคลเอนต์ โดยที่ฝั่งเซิร์ฟเวอร์ก็จะมีหน้าที่ในการให้บริการต่างๆ ส่วนที่ฝั่งไคลเอนต์ก็จะทำการร้องขอบริการต่างๆ ที่ต้องการ ซึ่งในการติดต่อกันระหว่างไคลเอนต์และเซิร์ฟเวอร์นั้นจะผ่านโปรแกรมที่ฝั่งไคลเอนต์ นั่นก็คือเบราว์เซอร์ โดยหน้าที่ของเบราว์เซอร์ก็คือทำการเชื่อมต่อการทำงานระหว่างเซิร์ฟเวอร์และผู้ใช้ ทำการแปลเอกสาร HTML และแสดงผลในรูปแบบของเว็บเพจ อีกทั้งยังสามารถทำการรันโปรแกรมบางอย่างจากฝั่งเซิร์ฟเวอร์ เช่น Java Script, Java Applet, ActiveX เป็นต้น

ในที่นี้ความหมายของไคลเอนต์คือเว็บเบราว์เซอร์ โดยเว็บเบราว์เซอร์ที่สำคัญที่จะกล่าวถึงคือ Internet Explorer ของทางไมโครซอฟท์และ Netscape Navigator ของทางเน็ตสเคป

ในการทำงานของเว็บเบราว์เซอร์นั้น จากที่กล่าวมาแล้วว่าหน้าที่ของเว็บเบราว์เซอร์ก็คือการแปลเอกสาร HTML แล้วแสดงออกมาเป็นหน้าเว็บเพจรวมไปถึงความสามารถในการรันโปรแกรมบางอย่างที่ฝั่งไคลเอนต์เพื่อเพิ่มประสิทธิภาพให้กับเว็บเพจ ซึ่งจากการทำงานดังกล่าว ที่เว็บเบราว์เซอร์จะทำการรันโปรแกรมที่ฝั่งไคลเอนต์ อาจทำให้เกิดความเสี่ยงของปัญหาและการละเมิดความเป็นส่วนตัวที่จะเกิดขึ้นต่อคอมพิวเตอร์ของผู้ใช้ ซึ่งปัญหาที่เกิดขึ้นนี้จะเกิดจากตัวเว็บเบราว์เซอร์เอง ซึ่งอาจจะเป็นบั๊กหรือข้อบกพร่องที่เกิดจากการออกแบบของตัวเว็บเบราว์เซอร์เอง ไม่ได้เกิดจากโปรแกรมที่ถูกรันโดยเบราว์เซอร์ และปัญหาที่เกิดจากโปรแกรมที่รันโดยเว็บเบราว์เซอร์

ปัญหาที่เกิดจากเว็บเบราว์เซอร์

- ปัญหาเนื่องจากข้อบกพร่องในการออกแบบเว็บเบราว์เซอร์

ปัญหาที่เกิดจากโปรแกรมที่รันโดยเว็บเบราว์เซอร์

- ปัญหาของเว็บเบราว์เซอร์เนื่องจากคุกกี้ (Cookie)

- ปัญหาของเว็บเบราว์เซอร์เนื่องจากโปรแกรมจาวา จาวาสคริปต์ และจาวาแอปเพล็ต
- ปัญหาของเว็บเบราว์เซอร์เนื่องจากการรัน ActiveX

ปัญหาเนื่องจากข้อบกพร่องในการออกแบบเว็บเบราว์เซอร์

ในการใช้งานคอมพิวเตอร์ในปัจจุบันนั้น ไม่ว่าจะใช้วินโดวส์ 9x, ME, 2000 หรือ XP ส่วนใหญ่ผู้ใช้ทั่วไปจะติดตามการแก้ไขบั๊กของโปรแกรมต่าง ๆ ไม่ทัน วิธีที่ปลอดภัยที่สุดก็คือการ ไม่เชื่อมต่ออินเทอร์เน็ตเลย แต่มันจะเป็นการลดความน่าสนใจในการใช้งานคอมพิวเตอร์เป็นอย่างมาก

ไมโครซอฟท์นั้นรีบออก IE6 มาเร็วเกินไป ยังไม่ทันจะตรวจสอบข้อบกพร่องของโปรแกรม และทดสอบด้านความปลอดภัย และรู้โหวให้ถ่วงก่อนก็รีบออก IE6 มาแล้ว

ปัญหาที่พบใน IE6 ได้แก่

1. การขโมยคุกกี้
2. ผู้บุกรุกเข้ามาอ่านไฟล์บนเครื่องของผู้ใช้
3. ผู้บุกรุกสั่งรันไฟล์ที่สามารถรัน (execute) ได้บนเครื่องของผู้ใช้
4. มีการเปลี่ยนชื่อไฟล์ที่แสดงให้เห็นใน download dialogue box

ปัญหาที่พบใน IE5.5 ได้แก่

1. ผู้บุกรุกเข้ามาอ่านไฟล์บนเครื่องของผู้ใช้
2. มีการเปลี่ยนชื่อไฟล์ที่แสดงให้เห็นใน download dialogue box

สาเหตุของปัญหาที่เกิดขึ้นใน IE6 และ IE5.5

1. การขโมยคุกกี้ เกิดจากการที่ ไมโครซอฟท์ได้นำจาวาสคริปต์ ซึ่งเป็นสคริปต์ที่รันที่ฝั่งไคลเอนต์มาใช้ และมันก็มีช่องโหว่ตรงที่ผู้ใช้สามารถใช้ฟังก์ชันของจาวาสคริปต์ เช่น document.open และ document.write เพื่อปลอมแปลงไปยังอีกเว็บไซต์หนึ่งและเมื่อผู้ใช้เข้าไปเว็บไซต์นั้นเว็บไซต์นั้นก็จะขโมยคุกกี้ที่อยู่บนเครื่องของผู้ใช้โดยใช้ฟังก์ชันของจาวาสคริปต์
2. ผู้บุกรุกเข้ามาอ่านไฟล์บนเครื่องของผู้ใช้ เกิดจากการที่ไมโครซอฟท์ได้นำจาวาสคริปต์มาใช้ และผู้บุกรุกใช้ฟังก์ชันของจาวาสคริปต์สั่งให้มาอ่านไฟล์บนเครื่องของผู้ใช้
3. ผู้บุกรุกสั่งรันไฟล์ที่สามารถรัน (execute) ได้บนเครื่องของผู้ใช้ เกิดจากการที่ผู้บุกรุกแก้ไขค่า Content-Disposition และ Content-Type ใน HTML Header ซึ่งผู้บุกรุกสามารถสร้างเว็บเพจไว้เพื่อให้ผู้ใช้มาเปิดดูหรือส่งจดหมายอิเล็กทรอนิกส์ประเภท HTML ไปยังเครื่องของผู้ใช้ ซึ่งจะสามารถสั่งรันไฟล์ที่สามารถรันได้บนเครื่องของผู้ใช้
4. มีการเปลี่ยนชื่อไฟล์ที่แสดงให้เห็นใน download dialogue box เกิดจากผู้ไม่ประสงค์ดีเปลี่ยนแปลงชื่อของไฟล์ที่ปรากฏใน dialogue box ของการดาวน์โหลดไฟล์ ซึ่ง dialogue box นี้อาจถูกเรียกจากเว็บเพจ หรือจดหมายอิเล็กทรอนิกส์ก็ได้

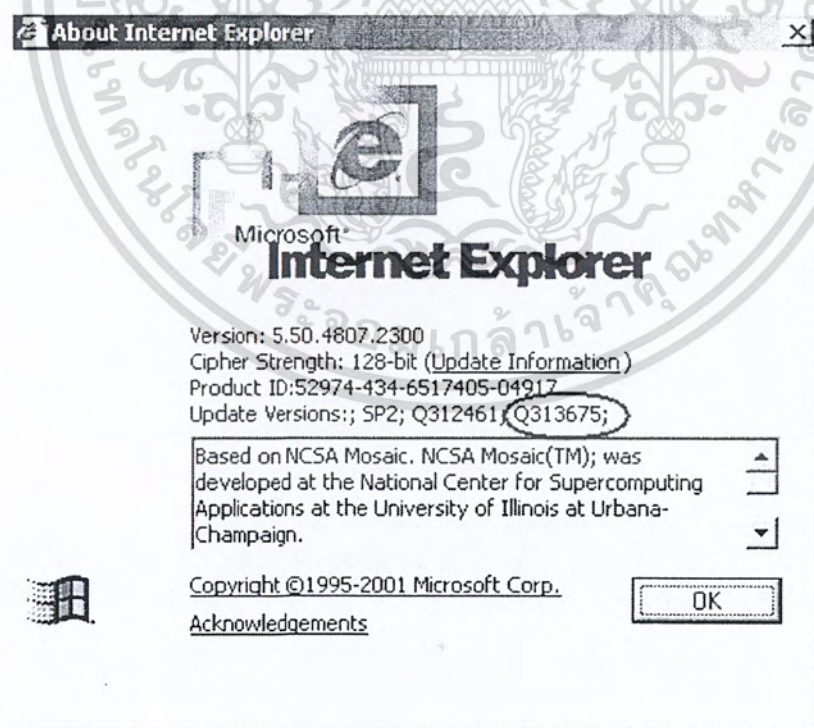
การแก้ปัญหาข้อบกพร่องของ IE5.5 และ IE6

ไมโครซอฟต์ได้ให้ผู้ใช้ไปดาวน์โหลดแพชมาจากที่ <http://www.microsoft.com/windows/ie/downloads/critical/Q313675/default.asp> มาใช้งานและเมื่อดาวน์โหลดมาเสร็จแล้วก็ให้ไปยกเลิก active scripting ในส่วน setting ของบราวเซอร์

ข้อมูลเพิ่มเติม

- สำหรับ IE 5.5 ควรติดตั้งแพชภายหลังการติดตั้ง IE 5.5 Service Pack 2 เรียบร้อยแล้ว
- สำหรับ IE 6.0 สามารถติดตั้งแพชนี้ได้ทันที
- โปรแกรมแพชนี้จะถูกรวบรวมไว้เป็นส่วนหนึ่งของ IE 5.5 service pack3 และ IE 6 service pack1
- การติดตั้งแพชจำเป็นต้องรีบูตเครื่องใหม่เพื่อให้การติดตั้งเสร็จสมบูรณ์

สำหรับการตรวจสอบว่าได้ติดตั้งแพชนี้เรียบร้อยแล้ว จะต้องมิกำว่า Q313675 อยู่ใน Update Versions จาก About Internet Explorer (เปิด IE, เลือก Help, เลือก About Internet Explorer) ดังตัวอย่างในภาพด้านล่าง



รูป 7-7 แสดง Internet Explorer หลังจากได้ติดตั้งแพชแล้ว

วิธีแก้ไข Active Scripting ใน Internet Explorer

- คลิกที่ Tools ในเมนูของบราวเซอร์
- เลือกแท็บ (tab) Security
- เลือกออปชัน Custom Level
- เมื่ออยู่ใน Setting box แล้วก็เลื่อนลงไปในส่วน Scripting แล้วคลิก disable ในส่วน Active scripting
- คลิก disable ในส่วน Scripting of Java applets
- กดปุ่ม OK เพื่อยืนยันการเปลี่ยนแปลง
- กดปุ่ม OK อีกครั้ง

นอกจากนั้นไมโครซอฟท์ IE ยังมีช่องโหว่ buffer overflow ในการจัดการออบเจกต์ซึ่งฝังอยู่ในข้อมูล HTML โดยผู้โจมตีอาจใช้ช่องโหว่นี้ในการรันโค้ดต่าง ๆ บนระบบของผู้เคราะห์ร้าย เมื่อผู้เคราะห์ร้ายเข้าเยี่ยมชมเว็บไซต์หรืออ่านจดหมายอิเล็กทรอนิกส์ที่เป็น HTML

IE สนับสนุนไคเร็กทีฟ <EMBED> ซึ่งใช้ในการฝังออบเจกต์ต่าง ๆ ใน HTML โดยออบเจกต์ดังกล่าวอาจได้ไฟล์จากมัลติมีเดีย, JAVA Applets หรือ ActiveX controls และ SRC จะกำหนด path ของแหล่งที่มาและชื่อของออบเจกต์นั้น ๆ ตัวอย่าง เช่น ข้อมูล MIDI อาจจะถูกฝังในเว็บเพจโดยใช้โค้ด HTML ดังต่อไปนี้

```
<EMBED TYPE TYPE = "audio/midi"
```

```
SRC="/path/sound.mid" AUTOSTART="TRUE">
```

การแก้ไขปัญหา buffer overflow

- นำแพชมาใช้

ไมโครซอฟท์ได้ออกแพชสำหรับ IE เพื่อแก้ไขช่องโหว่นี้รวมทั้งช่องโหว่อื่น ๆ ข้อมูลเพิ่มเติมเกี่ยวกับแพช และช่องโหว่ ท่านสามารถดูได้ที่ <http://www.microsoft.com/technet/security/buletin/MS02-005.asp>

- ปิด ActiveX Control และปลั๊กอิน

ปลั๊กอินของ IE ถูกใช้ในการดูหรือเริ่มทำงานกับ Object ที่ฝังอยู่ โดยการทำงานของ Object นั้นจะถูกควบคุมโดยการเลือก "Run ActiveX Control and Plug ins" ซึ่งการปิด option นี้ก็จะเป็นการป้องกันไม่ให้ Object เริ่มการทำงาน และนั่นคือการป้องกันการเข้าใช้จุดอ่อนนี้นั่นเอง อย่างน้อยที่สุดควรทำการปิด "Run ActiveX Control and Plug ins" โดยการปรับ security setting ให้เป็น High สำหรับ Internet Zone

ปัญหาที่เกิดจากการใช้งานคุกกี้ (Cookies)

คุกกี้จะเป็นแฟ้มข้อมูลขนาดเล็กที่อาศัยอยู่ในฮาร์ดไดรฟ์ (Hard Drive) ซึ่งจะเก็บข้อมูลข่าวสารบางอย่างเกี่ยวกับตัวของผู้ใช้ที่บ่งบอกถึงเว็บไซต์ต่างๆ ที่ผู้ใช้ได้เข้าไปเยี่ยมชมหรือทำธุรกรรมทั้งที่ผ่านมาและเว็บไซต์ปัจจุบันที่ผู้ใช้งานกำลังเล่น โดยข้อมูลส่วนใหญ่ที่คุกกี้ได้นั้นจะได้อาจมาจาก “ผู้ใช้” เอง คุกกี้นี้เป็นสิ่งที่ช่วยอำนวยความสะดวกในการทำงานบางอย่างบนอินเทอร์เน็ต เช่น ในบางเว็บไซต์ที่รับสมัครสมาชิกโดยให้ผู้ใช้กรอก ชื่อผู้ใช้ รหัสผ่าน และรายละเอียดส่วนตัว และมีอปชัน remind me for later ให้คลิกเลือกว่าจะให้เก็บข้อมูลไว้ในคุกกี้ไหมเพื่อที่จะล็อกอินให้โดยอัตโนมัติเมื่อผู้ใช้เข้ามายังเว็บไซต์นี้ โดยไม่ต้องกรอกรหัสผ่านอีก บ่อยครั้งที่คุกกี้จะใช้เพื่อลดการทำงานบางอย่างที่เกี่ยวข้องกับเบราว์เซอร์หรืออาจใช้ในการส่งข้อมูลส่วนบุคคลก็ได้ ตัวอย่างเช่น ในกรณีที่ผู้ใช้เข้าไปในไซต์ของ "TV Guide" เพื่อที่จะทำการหาข้อมูลเกี่ยวกับบัญชีโทรทัศน์ที่ผู้ใช้สนใจ ผู้ใช้ก็จำเป็นต้องใส่ข้อมูลว่าผู้ใช้อยู่ที่ไหน ใช้บริการแบบใด (Broadcast, cable, satellite, etc...) และชนิดต่างๆ ที่ผู้ใช้สนใจ ซึ่งข้อมูลเหล่านี้จะถูกบันทึกในคุกกี้ นั่นคือ เมื่อครั้งต่อไปที่ผู้ใช้จะเข้าไปดูข้อมูลเกี่ยวกับบัญชีโทรทัศน์ ข้อมูลต่างๆ ที่เกี่ยวกับตัวผู้ใช้จะถูกบันทึกไว้ซึ่งผู้ใช้เพียงแค่ใส่ข้อมูลเพียงครั้งเดียว นั่นคือเฉพาะครั้งแรกที่ผู้ใช้ได้เข้าไปดู โดยไม่จำเป็นต้องใส่ข้อมูลทุกครั้ง

ในอีกด้านหนึ่งคุกกี้ก็จะบอกถึงร่องรอย (Track) หรือการใช้อินเทอร์เน็ตของผู้ใช้ เช่น แสดงถึงว่าเว็บไซต์ต่างๆ ที่ผู้ใช้ได้เข้าไปเยี่ยมชมหรือทำธุรกรรม ซึ่งบ่อยครั้งในจุดนี้เองที่จะใช้ในด้านการตลาด (Marketing) และด้านการโฆษณา (advertising) โดยเว็บไซต์สามารถเข้าไปดูที่คุกกี้ว่าผู้ใช้ได้เคยเข้าไปในเว็บไซต์ไหนมาก่อน และทำการคาดการณ์ว่าผู้ใช้จะเข้าไปในเว็บไซต์ไหนต่อไป ซึ่งแบนเนอร์ก็จะถูกส่งมายังเบราว์เซอร์ของผู้ใช้ โดยผู้ใช้อาจสังเกตได้ว่าในกรณีที่ผู้ใช้เข้าไปยังเว็บไซต์ Search Engine เช่น Yahoo หรือ HotBot และทำการค้นหาก็จะปรากฏแบนเนอร์หลายๆ แบนเนอร์ในหลายๆ วินโดวส์ขึ้นที่เบราว์เซอร์ของผู้ใช้ซึ่งจะเกี่ยวข้องกับสิ่งที่ผู้ใช้ต้องการจะค้นหาใน Search Engine เช่น ผู้ใช้เข้าไปใน HotBot และทำการค้นหาคำว่า "used car prizes" ก็จะปรากฏแบนเนอร์ที่หน้าต่างใหม่ ซึ่งอาจจะจะเป็นของเว็บไซต์ "AutoTrader.com" ก็ได้

ตัวอย่างของคุกกี้ที่ถูกสร้างขึ้นหลังจากการเข้าไปยังไซต์ของ TV Guide ที่แสดงถึงข้อมูลส่วนตัวของผู้ใช้บางอย่าง:

.tvguide.com	TRUE/FALSE	205112224000	SITESERVER	ID=AC386
				91bb2db02e13188901d46e2bc75
.tvguide.com	TRUE/FALSE	15778336801	ServiceID	64536

ซึ่งคุกกี้ส่วนใหญ่จะเป็นการเข้ารหัสไว้ และอาจจะมีความหมายบางอย่างต่อเฉพาะบางเซิร์ฟเวอร์ที่ใช้คุกกี้ แต่จากตัวอย่างจะบ่งบอกถึงข้อมูลที่สำคัญบางอย่าง คือ โดเมนเนม (.tvguide.com) และชื่อที่เกี่ยวข้องกับค่าต่างๆ ของคุกกี้ (SITESERVER and ServiceID.) โดยชื่อจะบ่งบอกว่าเป็นใครและมีค่าอะไร อย่างไร

ซึ่งจากตัวอย่างของคุณก็ดูเหมือนว่าจะบ่งบอกถึง เว็บเซิร์ฟเวอร์ (หรือเว็บไซต์) และแสดงถึง User ID ที่เข้ามาใช้เซิร์ฟเวอร์นั้น เป็นต้น

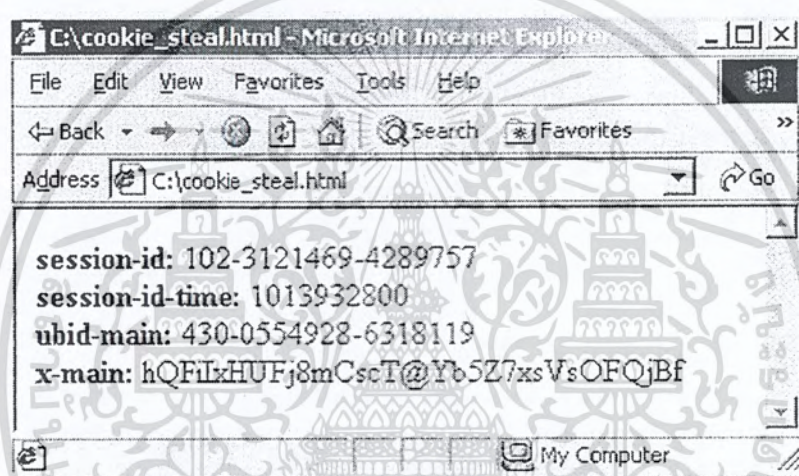
คุณก็นำไปใช้ในทางที่ไม่ดีได้เหมือนกัน เช่น ถ้ามีเว็บไซต์ที่ใช้ฟังก์ชันของจาวาสคริปต์ที่ไปขโมยคุกกี้ซึ่งเก็บหมายเลขผู้ใช้ และรหัสผ่านของบัญชีธนาคารแบบออนไลน์ในเครื่องของผู้ใช้ดังตัวอย่างโปรแกรมข้างต้น และแอบถอนเงินของผู้ใช้คนนั้น โดยที่ผู้ใช้ซึ่งเป็นเจ้าของบัญชีไม่รู้ตัว นอกจากนั้นคุณก็อาจถูกเว็บไซต์อื่นขโมยไปได้อย่างง่ายดายเหมือนกัน

ต่อไปนี้เป็นตัวอย่างของการขโมยคุกกี้จากเครื่องผู้ใช้ โดยผู้ใช้ได้สร้างบัญชีผู้ใช้ (account) ของตัวเองในเว็บไซต์หนึ่งแล้ว โดยชอรัสโค้ดของโปรแกรม c:\cookie_steal.html เป็นดังข้างล่างนี้

```
<html>
<head>
<title> Mmmm Gimme Cookie!</title>
<script language="JavaScript">
<!--
function getCookie()
{
    url = prompt("Enter a fully qualified domain name :");
    win = document.open(url, "urlWin", "top=5000, left=5000, width=1, height=1");
    cookie = win.document.cookie;
    // Close the window, we're done with it
    win.close();
    cookies = cookie.split(';');
    // How many are there?
    num = cookies.length;
    for (i = 0; i < num; i++)
    {
        // Get the cookie from the name/value pair
        curCookie = cookies[i].split('=');
        // Write its name
        document.write("<b>" + curCookie[0] + ": </b>");
        // Write its value
        document.write(curCookie[1] + "<br>");
    }
}
```

```
-->
</script>
</head>
<body bgcolor="#FFFFFF" onLoad="getCookie()">
</body>
</html>
```

เมื่อเรียกโปรแกรมนี้ขึ้นมาทำงานและใส่ชื่อโดเมนแบบเต็ม (ใส่ส่วน http:// ด้วย) ของเว็บไซต์ลงในกล่องสี่เหลี่ยมที่แสดงขึ้นมา แล้วผลลัพธ์ที่ได้ในบราวเซอร์จะเป็นดังรูปข้างล่างนี้



รูป 7-8 แสดงข้อมูลที่เก็บไว้ในเครื่องของผู้ใช้

การป้องกันปัญหาจากคุกกี้

มีอยู่หลายวิธีในการในการป้องกันการสร้างและเก็บคุกกี้ในเครื่องของผู้ใช้ โดยวิธีที่ง่ายที่สุดคือการยกเลิกการใช้คุกกี้ในบราวเซอร์ของผู้ใช้ แต่ถ้าใช้ IE6 วิธีแก้ก็คือ การไม่คลิกเลือกตัวเลือก remember me for later ซึ่งจะไม่ได้เก็บคุกกี้เอาไว้ในเครื่องผู้ใช้ และไปกำหนดตัวเลือกของความปลอดภัยใน IE6 ว่าเป็นเมื่อมีการสร้างคุกกี้ขึ้นมา และยอมให้มีการสร้างเฉพาะคุกกี้ที่ไม่ได้เก็บข้อมูลสำคัญส่วนตัว โดยข้อมูลที่อนุญาตให้เก็บได้ก็เช่น หมายเลขเซชันที่เข้ารหัสแล้ว หรือตัวแปร

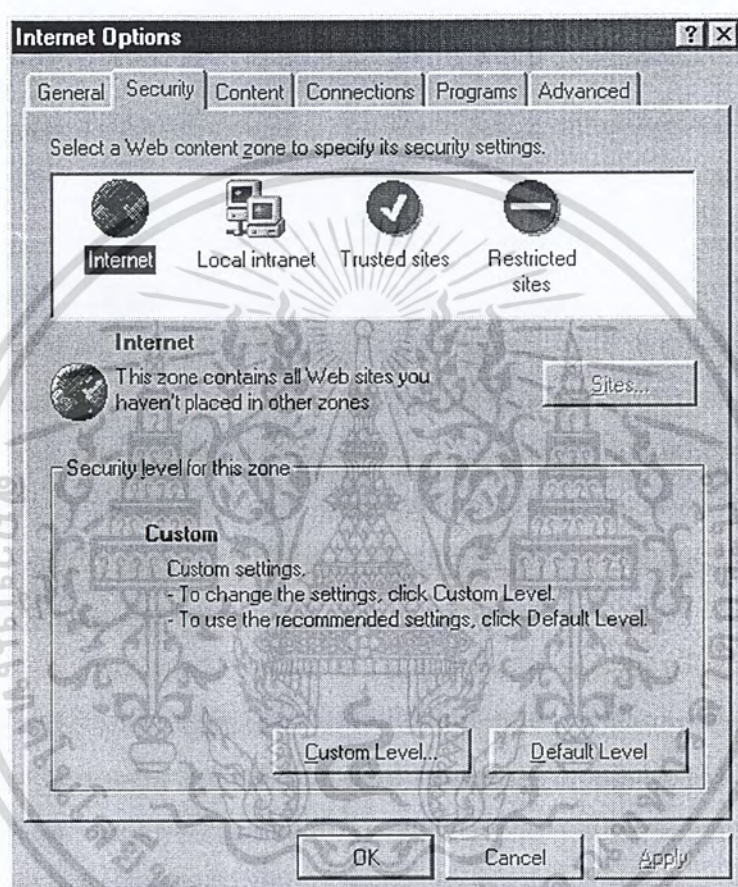
- การแก้ไขใน Internet Explorer:

- เลือกเมนู "Tools" ที่ทูลบาร์ จากนั้นทำการเลือก Internet Options
- เลือกแท็บ Security และเลือกใน Internet โชนจะมีทางเลือกในการติดตั้งคุกกี้อยู่ 3 ระดับดังนี้

Enable - อนุญาตให้มีการติดตั้งทุกคุกกี้และข้อมูลต่างๆ ซึ่งจะเป็น Default ในกรณีที่เราทำการเซต ความปลอดภัยในระดับ Medium, Medium-Low และ Low

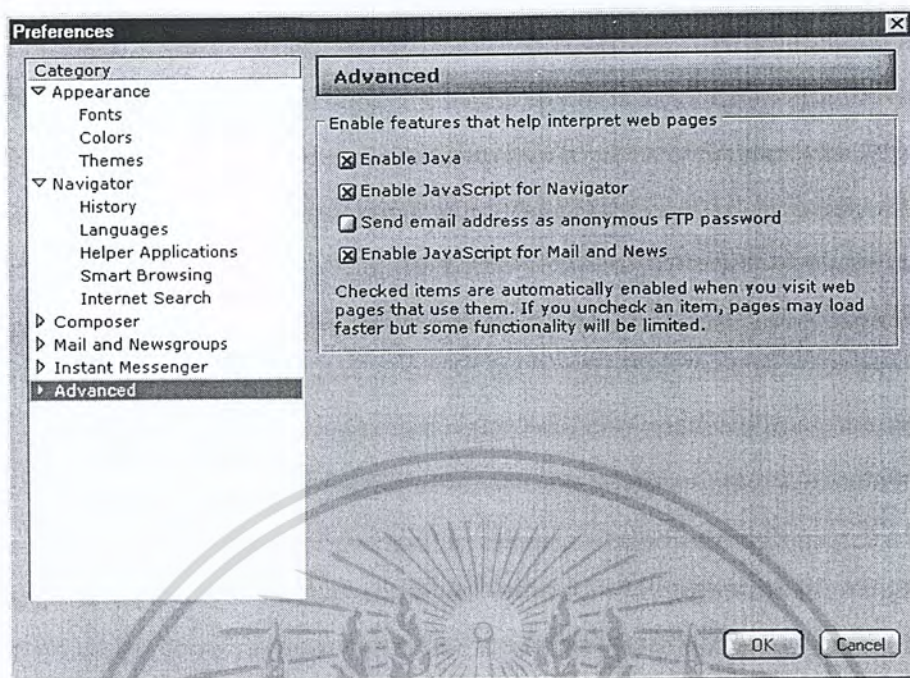
Disable – จะไม่อนุญาตให้มีการติดตั้งลูกก๊ี้ใดๆ ในเครื่อง โดยจะเป็น Default ในกรณีที่เราทำการเช็คความปลอดภัยในระดับ **High**

Prompt – จะทำการเตือนผู้ใช้ทุกๆ ครั้งที่จะมีการทำงานใดๆ ที่เกี่ยวข้องกับลูกก๊ี้ ซึ่งจะไม่เป็น Default ในระดับใดๆ ในกรณีที่ทำการเลือกในระดับ **"Prompt"** ผู้ใช้จะเสียเวลาส่วนใหญในการปฏิเสธ เพื่อทำการ Reject การขอติดตั้งลูกก๊ี้ เพราะเว็บไซต์ส่วนใหญ่จะใช้ลูกก๊ี้



รูปที่ 7-9 แสดงการปรับแต่งระดับความปลอดภัยในการติดตั้งลูกก๊ี้ใน IE

- การแก้ไขใน Netscape Navigator
 - จากเมนู Edit ทำการเลือกที่ Preferences
 - จาก Preferences ทำการเลือกที่ Advanced



รูปที่ 7-10 แสดงการปรับแต่งระดับความปลอดภัยในการติดตั้งคุกกี้ใน Netscape Navigator

ในกรณีของ Netscape จะเพิ่มหัวข้อในการติดตั้งคุกกี้ที่แตกต่างจาก Internet Explorer คือ:

"Accept only cookies that get sent back to the originating server"

ซึ่งผู้ใช้สามารถมั่นใจได้ว่าเว็บไซต์ที่ผู้ใช้กำลังติดต่ออยู่ด้วยนั้นจะได้รับข้อมูลของคุกกี้เหล่านั้นโดยไม่ใช่นุคคลอื่นหรือนุคคลที่สาม ซึ่งจะไม่ละเมิดความเป็นส่วนตัวของผู้ใช้ (Privacy)

ผู้ใช้จะต้องระลึกไว้เสมอว่าในกรณีที่ผู้ใช้ทำการยกเลิกการใช้คุกกี้ ผู้ใช้จะไม่สามารถทำการเซฟรูปแบบการติดตั้งใดๆ ได้เลย และที่แย่กว่าก็คือ ผู้ใช้จะไม่สามารถเข้าเยี่ยมชมบางเว็บไซต์ได้ถ้าปราศจากการใช้คุกกี้

ในด้านความปลอดภัย คุกกี้เป็นสิ่งที่น่าเป็นห่วงน้อยที่สุด นั่นคือ ในส่วนของจาวา จาวาสคริปต์ และ ActiveX เป็นสิ่งที่ต้องคำนึงในด้านความปลอดภัยมากกว่า

ปัญหาความปลอดภัยและผลกระทบจากจาวา จาวาสคริปต์ และ ActiveX

จาวา จาวาสคริปต์ และ ActiveX จะมีความเสี่ยงต่อความปลอดภัยของบราวเซอร์มากกว่า เหตุผลก็คือจาวาและ ActiveX จะเป็นโปรแกรมที่คุณทำการดาวน์โหลดและรันบนเครื่องของคุณ และจาวาสคริปต์จะเป็นภาษาสคริปต์ ที่คุณทำการดาวน์โหลดซึ่งเป็นส่วนหนึ่งของเว็บเพจ และจะบอกบราวเซอร์ของคุณถึงวิธีการทำงานหรือปฏิบัติต่อข้อมูลหรือบางสิ่งบางอย่าง โดยในปัจจุบันบางบราวเซอร์สามารถทำงานกับหลายๆ ภาษาสคริปต์ ซึ่งไม่ใช่เพียงแค่กับจาวาสคริปต์เท่านั้น อย่างไรก็ตามภาษาสคริปต์เหล่านั้นก็มีคุณสมบัติคล้ายกับจาวาสคริปต์

จาวา และ ActiveX

ทั้งจาวาและ ActiveX เป็นโปรแกรม (Applets) ซึ่งสามารถทำการดาวน์โหลดและรันบนเครื่องของคุณผ่านทางเบราว์เซอร์ที่คุณใช้ ซึ่งเมื่อเปรียบเทียบระหว่างจาวาและ ActiveX แล้ว ActiveX จะอันตรายและส่งผลเสียมากกว่า นั่นคือ ActiveX สามารถส่งผลกระทบต่อระบบ (System Calls) โดยทำให้ไฟล์บนไคลฟ์ของคุณมีปัญหาได้

ในกรณีของ ActiveX นี้ ไฟล์ใหม่อาจถูกสร้างหรือไฟล์เก่าอาจถูกเขียนทับหรือถูกดัดแปลงใหม่ก็ได้ลองคิดดูว่าจะเป็นอย่างไรถ้าเกิดในกรณีที่ไฟล์ autoexec.bat ในเครื่องของคุณถูกเขียนทับโดยเวอร์ชันอื่น ในปี 1997 ทางด้านสมาคมคอมพิวเตอร์ของเยอรมันแสดงตัวอย่างของการใช้ ActiveX ในการควบคุมเครื่องคอมพิวเตอร์ของคุณทำการโอนเงินจากธนาคารในบัญชีของคุณไปยังธนาคารอื่นในบัญชีใดก็ได้ที่พวกเขาต้องการ อย่างไรก็ตาม ActiveX จะไม่มีผลต่อความเสี่ยงด้านความปลอดภัยต่อ Netscape เพราะ Netscape ไม่รองรับ (Support) ActiveX

ในอีกทางหนึ่งจาวาแอปเพล็ตจะทำงานภายใต้ข้อจำกัดของเบราว์เซอร์ การจำกัดด้านความปลอดภัยจะเกี่ยวข้องกับ "Java Sandbox" โดยลักษณะสำคัญของ Java Sandbox ที่ป้องกันเบราว์เซอร์ของคุณจากแอปเพล็ต ดังนี้

- จะป้องกันการเข้าถึงหรือการเปลี่ยนแปลงไฟล์โดยตรงในเครื่องของคุณ
- จะป้องกันการเข้าถึงอุปกรณ์ต่างๆ ในเครื่องของคุณ และ
- จะป้องกันการสร้างเครือข่ายการติดต่อกับบุคคลอื่นๆ ที่ไม่ใช่เว็บไซต์ที่คุณกำลังติดต่อด้วย

อย่างไรก็ตามจาวาก็ยังอาจมีอันตรายต่อเบราว์เซอร์ของคุณ ทั้งนี้ขึ้นอยู่กับการออกแบบและข้อจำกัดต่างๆ ของเบราว์เซอร์ที่ได้รับการพัฒนาขึ้นมา ตัวอย่างหนึ่งของความไม่ปลอดภัยที่เกิดจากการพัฒนาเบราว์เซอร์ที่ไม่ดีพอ คือ ในกรณีคู่แข่งของจาวาแอปเพล็ต คือ "CrashCom405" ที่เมื่อทำการรันบน Netscape แล้วจะส่งผลให้เบราว์เซอร์หยุดทำการทันที โดยแอปเพล็ตตัวนี้จะมีผลด้านความปลอดภัยเฉพาะกับ Netscape เท่านั้น ไม่มีผลใดๆ ต่อ Internet Explorer ทั้งสิ้น

ในกรณีของจาวาสคริปต์ (หรือภาษาสคริปต์อื่นๆ) จะมีปัญหาด้านความปลอดภัยน้อยกว่า ซึ่งจะมีผลกับบางเว็บเพจเท่านั้น ซึ่งในทางปกติเมื่อมีการใช้หรือรันสคริปต์เหล่านี้อาจจะสร้างความรำคาญให้แก่คุณก็ได้ นั่นคือจะเกิดการเปิดหน้าต่างใหม่ของเบราว์เซอร์ขึ้นมาเรื่อยๆ โดยที่ทุกครั้งที่คุณพยายามจะปิดหน้าต่างนั้น ก็จะมีหน้าต่างใหม่เกิดขึ้นมาเรื่อยๆ เช่นกัน โดยผลที่แยกว่านั้นก็คือ ทรัพยากรของระบบ (เช่น Ram) อาจถูกใช้หมดและส่งผลให้เครื่องของคุณไม่สามารถทำงานเลยก็ได้ ผลสุดท้ายก็คือคุณก็จะต้องทำการปิดเครื่องและเริ่มต้นการทำงานใหม่ซึ่งจะไม่ส่งผลกระทบใดๆ ต่อเครื่องของคุณ

ตัวอย่างปัญหาด้านความปลอดภัยและการป้องกันของเว็บเบราว์เซอร์

JavaScript Cookie Exploit

ข้อบกพร่องนี้ได้รับการรายงาน ยืนยันและตรวจสอบว่ามีผลต่อ Netscape ในเวอร์ชัน 4.72 และเวอร์ชันก่อนๆ โดยผู้ที่ไม่ประสงค์ดีสามารถที่จะทำการอ่านลิงก์ (link) ในไฟล์ Bookmark ของคุณได้ในกรณีที่ผู้ไม่ประสงค์ดีนั้นทราบ Directory Path ของ Profile Name และ ของการติดตั้ง Netscape ซึ่งผลจากการวิเคราะห์แสดงให้เห็นว่าในกรณีที่ทราบ Directory Path นอกจากจะสามารถอ่านลิงก์ในไฟล์

Bookmark ได้แล้วยังสามารถที่จะอ่านแอททริบิวต์บางอย่างของไฟล์ HTML ได้ด้วย เช่น ไตเติล (Title) และลิงก์อื่นๆ ของไฟล์ HTML ในฮาร์ดดิสก์ของคุณ (แต่จะไม่สามารถอ่านได้ทั้งหมด) ซึ่งปัญหาที่เกิดขึ้นนี้ถือว่าเป็นปัญหาด้านความปลอดภัยและการละเมิดสิทธิส่วนบุคคล ซึ่งปัญหาดังกล่าวได้รับการแก้ไขใน Netscape เวอร์ชัน 4.73 ซึ่งสามารถทำการดาวน์โหลดได้ที่ <http://home.netscape.com/download/>

สาเหตุของปัญหาดังกล่าว เกิดจากสภาวะการณ์ต่างๆ ดังนี้

- ผู้ไม่ประสงค์ดีทราบ Directory หรือ Path ของการติดตั้ง Netscape ภายในเครื่องของคุณรวมไปถึง Profile Name ด้วย (เช่น เช็ตเป็น Default)
- ทำการเซตให้สามารถรับจาวาสคริปต์ได้
- อนุญาตให้มีการติดตั้งทุกก็ภายในเครื่อง

แต่จากปัญหาที่เกิดขึ้นไม่ได้รับการรายงานว่ามี การคุกคามกันทางเว็บและถึงแม้ว่าผู้ไม่ประสงค์ดีจะสามารถทำการอ่านไฟล์ HTML ได้ก็ตามแต่ก็ไม่สามารถทำการอ่านได้ทั้งหมด รวมไปถึงยังไม่สามารถทำการคัดแปลงระบบหรือข้อมูลของคุณได้

การป้องกันปัญหาที่เกิดขึ้น

- ทำการเปลี่ยนชื่อ Profile ของคุณเป็นอย่างอื่นที่ไม่ใช่เป็นการใช้ Default และไม่อนุญาตให้ผู้อื่นนำไว้วางใจหรือเว็บที่ไม่ปลอดภัยทราบถึงชื่อที่คุณได้ทำการเปลี่ยน ซึ่งขั้นตอนมีดังนี้
 1. เลือก "Start" และ ไปยัง "Programs" จากนั้น เลือก "Netscape Communicator" และคลิกที่ "Utilities" และ "User Profile Manager" ตามลำดับ
 2. ทำการเลือกชื่อ Profile ที่คุณต้องการจะเปลี่ยน เช่น "Default"
 3. คลิกที่ปุ่ม "Rename"
 4. พิมพ์ชื่อใหม่ที่ต้องการใช้
 5. หลังจากนั้น ห้ามเปิดเผยชื่อ Profile นี้ให้แก่เว็บหรือบุคคลใดที่ไม่น่าเชื่อถือ
- ทำการตรวจสอบและปรับการติดตั้งทุกก็ในเครื่องของคุณให้เป็นแบบ "Warn me before accepting a cookie," และปฏิเสธการขอติดตั้งทุกก็จากเว็บไซต์ที่ไม่น่าเชื่อถือ
 1. เลือกที่เมนู "Edit" และคลิกที่ "Preferences"
 2. คลิกที่ "Advanced"
 3. ตรวจสอบดูว่าการปรับเปลี่ยนการติดตั้งทุกก็เป็นแบบ "Warn me before accepting a cookie" หรือยัง ถ้ายัง ให้ทำการเลือกที่หัวข้อนี้
 4. ในกรณีที่มิเว็บไซต์ใดขอทำการติดตั้งทุกก็ในเครื่องของคุณ ถ้าหากว่าเป็นเว็บไซต์ที่ไม่น่าเชื่อถือก็ทำการปฏิเสธ โดยเลือกที่ "Cancel" ซึ่งจะทำการเตือนคุณก่อนทุกครั้งที่จะทำการติดตั้งทุกก็ในเครื่องของคุณ

ปัญหาความอ่อนแอด้านความปลอดภัยของจาวา

โดยทางด้าน Netscape ได้ทำการเตือนไปยังผู้ใช้วินโดวส์ ยูนิกซ์ และ Mac ถึงความอ่อนแอด้านความปลอดภัยเนื่องจากการพัฒนาจาวา ซึ่งจะมีผลกระทบต่อผู้ใช้ระบบปฏิบัติการดังที่ได้กล่าวมาข้างต้น

กับ Netscape Communicator และ Netscape Navigator เวอร์ชันตั้งแต่ 4.0 ขึ้นไป ซึ่งปัญหาดังกล่าวค้นพบโดย นาย Karsten Sohr และนาย Ed Felton ได้ทำการรายงานไปยังบริษัท Sun Microsystems ต่อไป

ทั้งนี้ทางบริษัท Sun Microsystems ได้ทำการตรวจสอบและยืนยันถึงปัญหาดังกล่าวที่เกิดขึ้นจริง ซึ่งจากความอ่อนแอดังกล่าวสามารถก่อให้เกิดการรั่วจาวาแอปเพล็ตที่ไม่ประสงค์ดีจากเว็บที่ไม่น่าเชื่อถือ ซึ่งอาจส่งผลกระทบต่อความปลอดภัยของข้อมูลหรือระบบต่างๆ ภายในเครื่องของคุณ โดยทาง Netscape เองได้แนะนำว่าคุณควรจะให้มีการเตือนจากเว็บไซค์กรณีที่คุณ ไม่ทราบแน่ชัดว่าน่าเชื่อถือหรือไม่รวมไปถึงในขณะที่คุณอนุญาตให้มีการรั่วจาวาจากเว็บที่น่าเชื่อถือด้วยเช่นกัน

คุณสามารถทำการป้องกันเบื้องต้นได้โดยการ ไม่อนุญาตให้มีการรั่ว จาวาดังขั้นตอนข้างต้น

1. ใน Communicator ทำการเลือกที่ "Preference" จากเมนู "Edit"
2. ใน "Preference" เลือกที่ "Advanced"
3. จากนั้นทำการยกเลิกหรือคลิกเครื่องหมายถูกหน้าหัวข้อ "Enable Java" ออก
4. คลิกที่ OK เป็นการเสร็จ

ความอ่อนแอทางด้านเฟรมสปูฟิง (Frame-Spoofing)

ทางด้าน Netscape ได้ทำการแจ้งเตือนถึงปัญหาด้านความอ่อนแอนี้แก่ผู้ใช้ Netscape Navigator ทุกเวอร์ชันที่แพลตฟอร์มมีการรองรับการใช้เฟรม (Frames) โดยผู้ที่ไม่ประสงค์ดีสามารถใช้ประโยชน์จากความอ่อนแอนี้ในการที่จะสร้างเฟรมหรือข้อความที่เป็นของผู้ไม่ประสงค์ดี โดยเฟรมที่ปรากฏจะดูเหมือนว่าเป็นเฟรมที่เป็นส่วนหนึ่งของเว็บไซค์ที่คุณกำลังเปิดอยู่ ซึ่งในการทำเช่นนี้ ผู้ไม่ประสงค์ดีสามารถทำให้คุณผู้ซึ่งกำลังท่องเว็บไซค์อยู่เกิดการหลงหรือเข้าใจผิดและจะหลอกให้คุณกรอกข้อมูลลงในแบบฟอร์ม โดยจะทำให้คุณเชื่อว่า คุณกำลังติดต่อกับเว็บไซค์ที่มีความน่าเชื่อถือ และผู้ไม่ประสงค์ดียังสามารถสร้างข้อมูลที่ไม่เหมาะสมหรือน่าอับอายให้ปรากฏบนเว็บไซค์ได้ซึ่งการทำเช่นนี้แล้วปกติผู้ที่ไม่ประสงค์ดีจะไม่ใส่ข้อมูลเหล่านั้นบนไซค์เป้าหมายของทางฝั่งเซิร์ฟเวอร์โดยตรง แต่จะทำให้ดูเหมือนว่าข้อมูลเหล่านั้นมาจากเว็บไซค์เป้าหมาย

ในการที่จะก่อให้เกิดการจู่โจมในลักษณะเช่นนี้ ผู้ไม่ประสงค์ดีจะต้องทำสิ่งใดสิ่งหนึ่งดังนี้

- ชักชวนให้คุณทำการคลิกลิงก์จากเว็บไซค์ของผู้ไม่ประสงค์ดีนั้น หรือ
- ทำการส่งอีเมลล์หลอกล่อไปยังทางฝั่งไคลเอนต์ที่มีการใช้จาวาสคริปต์ เช่น Netscape Messenger และทำการล่อลวงให้คุณคลิกที่ไฮเปอร์ลิงก์ (Hyperlink) ในเมลนั้น (สมมุติว่าเป็นไซค์เป้าหมาย)

ถึงแม้ว่าการจู่โจมแบบเฟรมสปูฟิงนี้จะไม่มีความจำเป็นต้องให้ทางฝั่งไคลเอนต์เซตให้มีการรั่วจาวาสคริปต์แต่ในความเป็นจริงแล้วการที่จะจู่โจมในลักษณะนี้ให้สำเร็จก็ต้องให้ทางฝั่งไคลเอนต์อนุญาตให้สามารถรันจาวาสคริปต์ได้ เพราะถ้าไม่อนุญาตให้มีการรัน นั่นหมายความว่าจู่โจมได้ก็ต่อเมื่อมีหน้าต่าง (window) ในฝั่งของไซค์เป้าหมาย (targeted site) ถูกเปิดไว้ก่อนที่ผู้ไม่ประสงค์ดีจะทำการล่อลวงให้คุณคลิกที่ไฮเปอร์ลิงก์ในเมลที่จู่โจมหรือบนเว็บไซค์ของผู้ไม่ประสงค์ดีนั่นเอง

คุณสามารถทำการป้องกันผลกระทบเหล่านี้ได้โดย:

1. หลีกเลี่ยงหน้าต่าง หน้าจอหรือเว็บไซค์ที่ไม่น่าเชื่อถือ

2. ใช้ Communicator 4.5 และไม่อนุญาตให้มีการรันหรือใช้จาวาสคริปต์ในเมลล์ตามขั้นตอนดังต่อไปนี้
 - 2.1 เลือกที่ "Preferences" จากเมนู "Edit"
 - 2.2 ใน "Preferences" คลิกที่ "Advanced"
 - 2.3 ทำการยกเลิกการใช้จาวาสคริปต์ในช่อง "JavaScript for Mail and News"
 - 2.4 คลิกที่ OK

The JavaScript Cache Browsing Bug

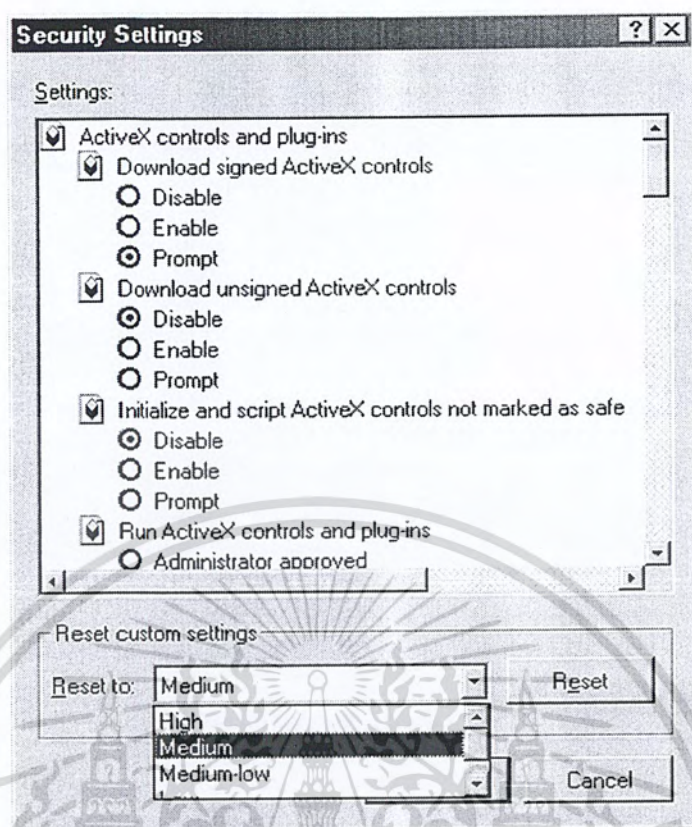
ปัญหาของ Cache Browsing นี้จะมีผลต่อวินโดวส์เวอร์ชันของ Navigator 3.04 และ 4.07 และ Netscape Communicator 4.5 (แต่ไม่มีผลต่อระบบปฏิบัติการ Mac และยูนิกซ์) โดยบั๊ก (Bug) หรือจุดอ่อนนี้สามารถถูกใช้โดยเว็บไซต์ที่มีการรันโปรแกรมที่มุ่งร้าย (malicious) หรืออันตราย ในการที่จะทำการอ่าน URL จากแคชที่อยู่ในเครื่องของคุณ ภายใต้สภาวะการณ่ปกติแล้วทางด้าน Netscape เชื่อว่าคุณจะไม่สามารถทราบหรือรู้ตัวว่ามีการอ่าน URL จากแคชในเครื่องคุณอยู่ นั่นก็หมายความว่าถ้าคุณกำลังติดต่อกับเว็บไซต์ที่คุณไม่ทราบหรือไม่น่าเชื่อถือ ก็เป็นไปได้ว่าคุณอาจกำลังถูกอ่านข้อมูลที่อยู่ภายในแคชของคุณอยู่ก็ได้

จะเห็นได้ว่าปัญหาของบั๊กเกิดขึ้นอยู่กับจาวาสคริปต์ที่ทำการเปิดข้อมูล ซึ่งคุณสามารถทำการหลีกเลี่ยงปัญหาดังกล่าวได้โดยการปิดจาวาสคริปต์เมื่อติดต่อกับเว็บที่ไม่รู้ที่มาหรือไม่น่าเชื่อถือ ซึ่งการจะทำการยกเลิกการใช้จาวาสคริปต์นั้นก็ได้กล่าวมาแล้วข้างต้น

จากตัวอย่างของปัญหาที่เกิดขึ้นจะเห็นได้ว่าเราสามารถทำการลดความเสี่ยงของสาเหตุที่ทำให้เกิดปัญหาได้โดยการเซตค่าต่างๆ ทั้งในส่วนของคุณก็จาวาสคริปต์ จาวาแอปเพล็ตและ ActiveX ซึ่งสามารถสรุปแนวทางการป้องกันทั้งของ Internet Explorer และ Netscape ได้ดังนี้

- Internet Explorer

- จากเมนู Tools ทำการเลือก **Internet Options**
- เลือกที่แท็บ **Security**
- ทำการเลือกระดับความปลอดภัย (Security Level) ซึ่งคุณสามารถเลือกระดับความปลอดภัยทั้ง Low, High และ Medium โดยเลือกที่แท็บ "Custom Level" ดังรูป



รูปที่ 7-11 การเพิ่มความปลอดภัยจากจาวาสคริปต์ จาวาแอปเพล็ตและ ActiveX

ใน "Custom Level":

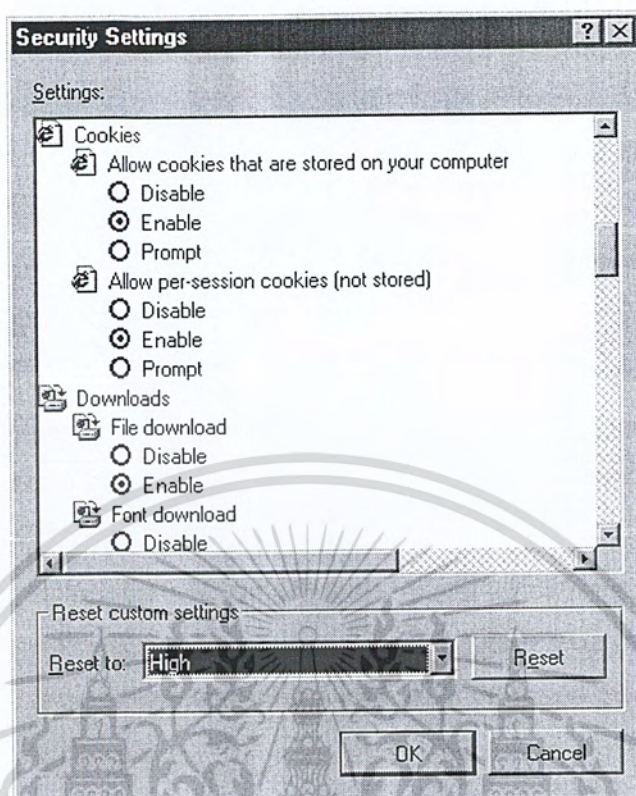
High จะทำการป้องกันทุกอย่างไม่ว่าจะเป็นลูกที่ ActiveX และแอปเพล็ตแต่บางเว็บไซต์จะไม่สามารถทำงานได้ถ้าปราศจากจาวาสคริปต์หรือลูกที่

Medium จะเป็นการเซตแบบ Default นั่นคือบราวเซอร์จะทำงานตามปกติและจะถามในกรณีของ ActiveX หรือเนื้อหาต่างๆ ที่ไม่มีความปลอดภัย

Medium Low จะให้บราวเซอร์ทำงานไปตามปกติแต่จะไม่ถามคุณในกรณีที่เนื้อหาไม่มีความปลอดภัย

Low จะอนุญาตให้การทำงานทุกอย่าง ทุกรูปแบบเกิดขึ้นได้ โดยไม่มีการรักษาความปลอดภัย

จากตัวอย่างข้างล่างเป็นการเซตระดับการรักษาความปลอดภัยในระดับ "High" โดยจะยกเลิกการทำงานทุกๆ อย่างยกเว้น ActiveX ที่มีความน่าเชื่อถือ (trusted) และใช้ untrusted สำหรับจาวาแอปเพล็ต ดังรูป

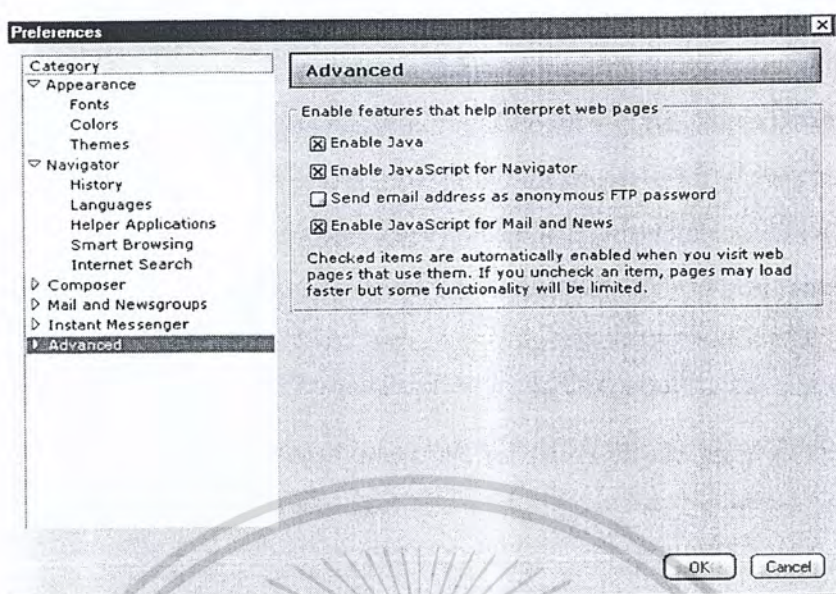


รูปที่ 7-12 รูปแบบการตั้งค่าความปลอดภัยของคุกกี้และจาวาแอปเพล็ต

- Netscape Navigator

- จากเมนู Edit ทำการเลือก Preferences
- จาก Preferences เลือกหัวข้อ Advanced

ในกรณีของ Netscape คุณเพียงแค่เลือกว่าจะให้มีการใช้ (Enable) หรือยกเลิกการใช้ (Disable) จาวาและจาวาสคริปต์โดยไม่มีให้เลือกลำหรับ "Prompt" ซึ่งเฉพาะคุกกี้เท่านั้นที่มีหัวข้อของ "Prompt" ดังรูป



รูปที่ 7-13 รูปแบบการตั้งค่าความปลอดภัยของจาวาแอปเพล็ตใน Netscape

ปัญหาที่เกิดจากผู้ใช้งานเว็บเบราว์เซอร์

ในกรณีปัญหาที่เกิดจากตัวของผู้ใช้เว็บเบราว์เซอร์จะเป็นปัญหากรณีเฉพาะ หมายความว่า ปัญหาจะเกิดขึ้นหรือไม่ขึ้นอยู่กับตัวของผู้ใช้ และในกรณีที่เกิดปัญหาใดๆ ขึ้น การที่จะแก้ปัญหาก็จะอยู่ที่ตัวของผู้ใช้เองด้วยเช่นกัน ดังนั้นในการที่จะป้องกันไม่ให้เกิดปัญหา ผู้ใช้จะต้องมีความตระหนัก (awareness) รัดกุม (conciseness) และมีสติ (consciousness) ในการใช้เว็บเบราว์เซอร์เพื่อการทำธุรกรรมทางอินเทอร์เน็ต

ปัญหาที่เกิดขึ้นจากผู้ใช้งานเว็บเบราว์เซอร์สามารถแบ่งได้ 2 ลักษณะใหญ่ๆ ดังนี้

I). ปัญหาที่เกิดโดยผู้ใช้งานการปรับค่าคอนฟิกต่างๆ ทั้งโดยความตั้งใจและไม่ตั้งใจ

ปัญหาที่เกิดขึ้นในลักษณะนี้จะเกิดจากการที่ผู้ใช้งานการปรับเปลี่ยนค่าคอนฟิกูเรชันบางอย่าง และส่งผลให้เกิดผลกระทบต่อทรัพยากรภายในของระบบหรือการล่วงละเมิดความเป็นส่วนตัวของผู้ใช้ หรือปัญหาอื่นๆ ไม่ว่าจะเป็นการปรับเปลี่ยนคุกกี้ (cookies) จาวาสคริปต์ จาวาแอปเพล็ตและรวมไปถึง ActiveX Control ซึ่งผู้ใช้อาจจะทราบถึงผลกระทบที่เกิดขึ้นหากไม่ทำการยกเลิกในบางตัวเลือกแต่ถึงกระนั้นก็ตามที่ ผู้ใช้ก็ไม่สามารถทำได้เพราะจะทำให้เบราว์เซอร์ไม่สามารถเรียกใช้บริการเว็บเพจได้ อย่างเช่นในกรณีของคุกกี้ที่ผู้ใช้งานสามารถรักษาความเป็นส่วนตัวได้ร้อยละเก้าสิบห้าถ้าทำการยกเลิกคุกกี้ทุกอย่าง แต่ผลกระทบก็คือ บางเว็บเพจจะไม่สามารถทำงานได้เลยถ้าไม่มีการยอมให้ติดตั้งคุกกี้ เป็นต้น

II). ปัญหาการดาวน์โหลดและรันโปรแกรมที่มีผลต่อความเสี่ยงที่เครื่องคอมพิวเตอร์ของผู้ใช้

เป็นปัญหาที่เกิดจากกรณีที่ผู้ใช้งานการดาวน์โหลดโปรแกรมจากอินเทอร์เน็ตโดยไม่ได้ทำการตรวจสอบว่าโปรแกรมที่ตนทำการดาวน์โหลดมานั้นมีไวรัสหรือเป็นโปรแกรมที่เมื่อรันแล้วจะส่งผลกระทบต่อระบบของเราหรือไม่ ตัวอย่างของปัญหานี้ได้แก่ เว็บไซต์แห่งหนึ่งได้เปิดให้ผู้ใช้งานสามารถดาวน์โหลดไฟล์

ภาพโป๊ไปชมได้ฟรี แต่ต้องดูด้วยโปรแกรมวิวเวอร์ซึ่งรันผ่านเว็บ ด้วยความไม่รู้ของผู้ใช้เมื่อรันโปรแกรมนี้แล้ว โปรแกรมนี้จะทำการตัดขาดการติดต่อกับ ไอเอสพี(Internet Service Provide: ISP) ของผู้ใช้งาน และทำการเชื่อมต่อให้กับไอเอสพีที่อยู่ห่างไกลให้แทน เช่น ต่างประเทศ เป็นต้น จากเหตุการณ์ดังกล่าว หากผู้ใช้หลงระเริงกับการชมภาพโดยไม่รู้ตัว ก็จะพบว่ามีค่าโทรศัพท์มหาศาลที่จะคอยหลอกหลอนตามมา นอกจากนี้ยังเกิดกรณีดังกล่าวกับนักดาวน์โหลดโปรแกรมบนเว็บที่มีการติดตั้ง (Install) หรือรันโปรแกรมผ่านเว็บ นอกจากนี้ยังอาจจะนำมาซึ่งไวรัสมากมายได้อีกด้วย



บทที่ 8

ปัญหาความปลอดภัยสำหรับการสื่อสารระหว่าง ฝั่งเซิร์ฟเวอร์และฝั่งไคลเอนต์

8.1 ปัญหาความปลอดภัยสำหรับการสื่อสารระหว่างเว็บไคลเอนต์และเว็บเซิร์ฟเวอร์

ปัญหาที่เกิดขึ้นกับการสื่อสารระหว่างเว็บไคลเอนต์และเว็บเซิร์ฟเวอร์ โดยส่วนใหญ่ผู้มั่งร้ายจะมุ่งเน้นความสนใจไปยังข้อมูลที่ทำกรสื่อสารกันระหว่างเว็บไคลเอนต์และเว็บเซิร์ฟเวอร์ ทั้งในรูปแบบของการดักเอาข้อมูลไปใช้ประโยชน์ การเปลี่ยนแปลงข้อมูลและการแอบอ้างตัวบุคคล โดยเฉพาะประการหลังจะทำให้เกิดปัญหาการไม่ไว้วางใจกันระหว่างเว็บไคลเอนต์และเว็บเซิร์ฟเวอร์ ซึ่งจะส่งผลไปถึงการลดอัตราการให้บริการบนเว็ลด์ไวด์เว็บในเชิงพาณิชย์ตามมา สามารถแยกพิจารณาเป็นกรณีย่อยๆ ได้ดังนี้

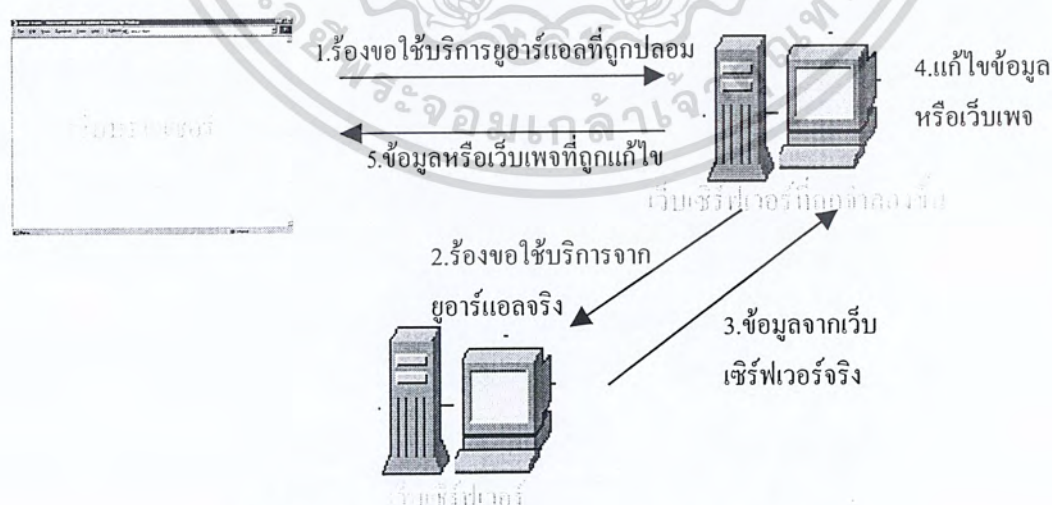
- ปัญหาในการแอบอ้าง

ปัญหาการแอบอ้างที่สำคัญ ได้แก่ เว็บสปูฟิง (Web Spoofing) คือ การจำลองเว็บที่มีลักษณะเหมือนเว็บต้นแบบทุกประการ (shadow copy) เพื่อให้ผู้ใช้ติดต่อเข้ามาโดยไม่รู้ว่าเป็นเว็บที่ถูกจำลองขึ้นมา โดยมีวัตถุประสงค์เพื่อตรวจจับ แก๊ไข เปลี่ยนแปลงข้อมูลต่างๆ ที่สื่อสารกันระหว่างเว็บไคลเอนต์กับเว็บเซิร์ฟเวอร์จริงที่ผู้ใช้ต้องการติดต่อด้วย

ลักษณะการทำงานของเว็บสปูฟิง

เว็บเซิร์ฟเวอร์ที่จำลองขึ้นมา (Spoofed Web Server) จะทำหน้าที่เป็นตัวกลางการติดต่อสื่อสารระหว่างผู้กับเว็บเซิร์ฟเวอร์จริง โดยที่ผู้ใช้และเว็บเซิร์ฟเวอร์จริงไม่รู้ตัว

สามารถแสดงการทำงานได้ ดังรูปที่ 8-1



รูปที่ 8-1 แสดงการทำงานของเว็บสปูฟิง (Web Spoofing)

การสร้างเว็บที่ถูกจำลองขึ้นมาอาจจะหลงเหลือหลักฐานต่างๆ ไว้ได้ เช่นที่แถบแสดงสถานะ (Status Bar) ที่แถบแสดงที่อยู่ (Location Line) แสดงที่อยู่ของเว็บไซต์ที่เราจะติดต่อด้วยไม่ถูกต้อง เป็นต้น แต่หลักฐานเหล่านี้สามารถปกปิดได้โดยการใช้จาวาสคริปต์ (JavaScript)

สิ่งสำคัญที่จะทำให้เว็บสปูฟิงเกิดผลจริง ก็คือ ทำอย่างไรให้มีผู้เข้ามายังเว็บนี้ให้ได้ โดยวิธีส่วนใหญ่ที่ใช้กันอยู่ เช่น

- การฝากลิงก์ที่จะมายังเว็บที่ถูกจำลองขึ้นมาในเว็บที่มีชื่อเสียง
- ส่งไปในรูปของเมล (e-mail) โดยเป็นลักษณะของเมลที่มีลิงก์ไปยังเว็บไซต์เว็บได้ (Web-enabled email)

เทคนิคที่แฮกเกอร์นิยมใช้ในการทำเว็บสปูฟิง คือ การทำดีเอ็นเอสสปูฟิง (DNS Spoofing) ซึ่งเป็นการอาศัยจุดอ่อนของโพรโตคอลทีซีพี/ไอพีโดยจะเป็นการจู่โจมไปที่เนมเซิร์ฟเวอร์ กล่าวคือ

1. จู่โจมให้เนมเซิร์ฟเวอร์ไม่สามารถทำงานได้ชั่วคราว
2. ตอบข้อมูลไอพีแอดเดรสที่ผิดกลับไปก่อนที่จะเนมเซิร์ฟเวอร์จริงจะตอบ
3. โดยสิ่งที่จำเป็นต้องทราบก็คือ หมายเลขของแพ็กเก็ตที่จะตอบรับคำร้องขอของเว็บไคลเอนต์ (id := id+1) โดยอาศัยวิธีการสนิฟฟิง (Sniffing)

ตัวอย่าง 161.246.5.27 : 3128 → [? www.hotmail.com ID = 123] → 161.246.4.3 : 53

ผลลัพธ์

161.246.5.27 ← ----- 161.246.4.3

161.246.5.27 ← [IP for www.hotmail.com is 1.2.3.4 ID = 124] ← ns.attacker.com : 53

โดยผู้จู่โจมจะต้องส่งหมายเลขแพ็กเก็ตตอบรับให้ถูกต้อง ($124 = 123 + 1$) ไอพีแอดเดรส 1.2.3.4 จะเป็นไอพีแอดเดรสของเว็บไซต์ที่จำลองเว็บไซต์ www.hotmail.com ขึ้นมา

นอกจากนี้ยังมีปัญหาที่เกิดขึ้นกับเนมเซิร์ฟเวอร์อีกประการหนึ่ง ซึ่งเป็นการแฮกเข้าไปที่เรดเทเบิ้ล (Route Table) แล้วแก้ไขไฟล์คอนฟิก (Config file) ที่แม้ระหว่างโดเมนเนมกับไอพีแอดเดรส เช่น www.pantip.com 168.100.10.129 → ถูกแก้เป็น → www.pantip.com 161.246.10.21 ดังนั้นใครก็ตามที่เรียกใช้บริการเว็บไซต์เว็บผ่านเนมเซิร์ฟเวอร์ตัวนี้ เมื่อใดที่จะลิงก์ไป www.pantip.com ก็จะโดนลิงก์ไปยัง www.kmitl.ac.th แทน ปัญหาดังกล่าว เรียกว่า เว็บไฮจกิง (Web Hijacking) ดังตัวอย่างที่เว็บไซต์ www.sanook.com เคยประสบมา

นอกจากนี้ปัญหาการแอบอ้างยังเกิดได้จากแฮกไปยังหน่วยงานที่รับผิดชอบโดเมนเนม โดยแต่เดิมหน่วยงานรับผิดชอบโดเมนเนมจะมีอยู่ทีเดียวนั่นคือ www.InterNIC.net ซึ่งเป็นผู้มีสิทธิในการแจกจ่ายชื่อโดเมนเนมให้กับผู้ขอ แต่เมื่ออินเทอร์เน็ตขยายกว้างขึ้น InterNIC เลยแบ่งการทำงานลงไปให้ดูแลเป็นโซน (zone) อย่างของประเทศไทยเราจะอยู่โซนเอเชียแปซิฟิก (Asia-Pacific) ผู้ดูแลโดเมนเนมในโซนนี้คือ APNIC (Asia-Pacific Network Information Center) ในไทยเองก็มีผู้ดูแลในประเทศชื่อ THINIC (Thailand Network Information Center) เป็นผู้ดูแลโดเมนเนมที่ลงท้ายด้วย .th ทั้งหมด สำหรับ

การแยกดังกล่าว สามารถทำได้โดยการปลอมแปลงเอกสารทางราชการต่างๆ ที่จำเป็นเพื่อยืนยันว่าเป็นเจ้าของโดเมนเนมนั้นจริง แล้วส่งไปให้ยังหน่วยงานที่โดเมนเนมนั้นจดทะเบียนอยู่ แล้วทำทีเป็นว่ามาขอทำการแก้ไขข้อมูลต่างๆ เช่น บอกว่าลืมพาสเวิร์ด เป็นต้น ซึ่งฟังดูแล้วก็ไม่น่าเป็นไปได้แต่ก็เคยเกิดเป็นเหตุการณ์พิพาทครั้งใหญ่มาแล้ว เช่น www.thailand.com เป็นต้น

- ปัญหาข้อมูลที่ทำให้การรับส่งถูกดักลอบไป (Data Trapping)

ข้อมูลที่ทำให้การสื่อสารระหว่างเว็บไคลเอนต์และเว็บเซิร์ฟเวอร์จะถูกดักลอบโดยแฮกเกอร์ ซึ่งข้อมูลที่ถูกดักลอบไปนี้จะถูกแฮกเกอร์นำไปใช้ประโยชน์ ตัวอย่างการถูกดักลอบ เช่น กรณีที่ผู้ใช้ทำการสั่งซื้อสินค้าผ่านทางอินเทอร์เน็ต (E-Commerce) ซึ่งผู้ซื้อจะต้องทำการใส่หมายเลขบัตรเครดิต หรือ พาสเวิร์ด ซึ่งในกรณีที่ข้อมูลเหล่านี้ถูกดักลอบไปจะเกิดความเสียหายต่อเจ้าของเป็นอย่างมาก ดังรูปที่ 8-2

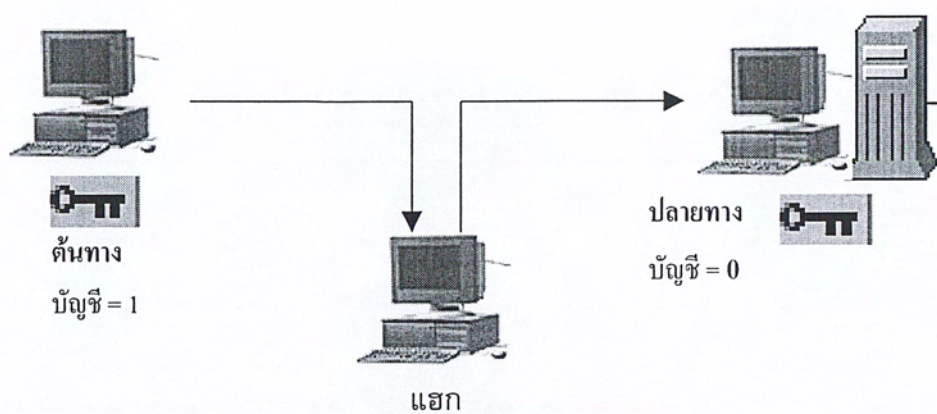
แฮกเกอร์สามารถทำการดักจับข้อมูลโดยใช้โปรแกรมในการดักจับข้อมูล เช่น สนิฟเฟอร์ (Sniffer) ซึ่งเป็นโปรแกรมที่ใช้ในการดักจับข้อมูลที่ถูกส่งมาในรูปของแพ็กเก็ตที่ไหลผ่านในเครื่องคอมพิวเตอร์ตัวที่ติดตั้งโปรแกรมสนิฟเฟอร์ไว้ไม่ว่าข้อมูลจะถูกกระโดดไฟแอดเดรสปลายทางเป็นอะไรก็ตาม ตัวโปรแกรมสนิฟเฟอร์สามารถดักข้อมูลที่อยู่ในแพ็กเก็ตนั้นมาอ่านได้ แม้ว่าข้อมูลนั้นไม่ได้ส่งถึงมันก็ตาม



รูปที่ 8-2 แฮกเกอร์ทำการดักลอบข้อมูล เพื่อนำข้อมูลไปใช้ประโยชน์

- ปัญหาข้อมูลที่ทำให้การรับส่งถูกเปลี่ยนแปลง (Data Altering)

ประเภทของของความเสียหายต่อระบบความปลอดภัยอีกแบบหนึ่งคือ แฮกเกอร์ไม่ได้ทำการดักเอาข้อมูลไปแต่เป็นการเปลี่ยนแปลงเป็นข้อมูลตัวใหม่ลงไปแทนซึ่งข้อมูลตัวใหม่ก็ถูกเข้ารหัสเหมือนกันและยังถูกต้องตามวิธีการทุกอย่าง เช่นเดียวกับข้อมูลเดิมที่ได้ทำไว้แต่รายละเอียดของข้อมูลเปลี่ยนไป ซึ่งรูปแบบการเปลี่ยนแปลงข้อมูลนี้จะทำให้ปลายทางได้รับข้อมูลที่ผิดทำให้การทำงานผิดพลาดและอาจให้บริการกับผู้ใช้ (User) ผิดคนและทำให้การสื่อสารผิดพลาด สามารถแสดงได้ดังรูปที่ 8-3



รูปที่ 8-3 แสดงการเกิดการเปลี่ยนแปลงข้อมูลในระหว่างการส่ง



บทที่ 9

การจัดตั้งเว็บไซต์ของห้องวิจัย ISAG ที่มีความปลอดภัย

9.1 วัตถุประสงค์ของเว็บไซต์ของห้องวิจัย

เว็บไซต์ของห้องวิจัย ISAG จัดทำขึ้นเพื่อเป็นแหล่งข้อมูลและศูนย์กลางในการเผยแพร่ความรู้เกี่ยวกับระบบรักษาความปลอดภัยในคอมพิวเตอร์ โดยภายในเว็บไซต์จะมีข้อมูลข่าวสารต่างๆ มากมาย อันได้แก่ ข่าวความเคลื่อนไหวต่างๆ กิจกรรมที่น่าสนใจ บทความ เอกสารและไฟล์ต่างๆ ข้อมูลโครงการของห้องวิจัยและข้อมูลอื่นๆ ที่น่าสนใจ นอกจากนี้ยังเป็นแหล่งพบปะระหว่างสมาชิกของห้องวิจัยด้วย

9.2 ฟีเจอร์ที่สำคัญของเว็บไซต์ของห้องวิจัย

- ระบบสมาชิก สามารถเพิ่มสมาชิก หรือลบสมาชิกออกจากระบบได้ โดยสมาชิกสามารถกรอกข้อมูลเพื่อใช้ในการติดต่อกันระหว่างรุ่นได้
- ส่วนประกาศข่าว สามารถประกาศข่าวในหน้าแรกของเว็บไซต์ได้ โดยข่าวนี้ยังแยกเป็น ข่าวประกาศภายนอกซึ่งบุคคลทั่วไปสามารถดูได้และข่าวประกาศภายในซึ่งเฉพาะสมาชิกที่ล็อกอินเข้ามาในเว็บไซต์เท่านั้นที่สามารถดูได้
- ส่วนเขียนบทความ โดยสมาชิกที่ล็อกอินเข้ามาสามารถเขียนบทความต่างๆ ที่น่าสนใจ โดยสามารถแบ่งแยกบทความเป็นหมวดหมู่ได้
- ส่วนเพิ่มข้อมูลโปรเจกต์ โดยสมาชิกสามารถกรอกข้อมูลโปรเจกต์ที่ทราวมทั้งอัปโหลดไฟล์โปรเจกต์เพื่อเผยแพร่โปรเจกต์ของห้องวิจัยผ่านทางเว็บไซต์ได้
- ส่วนเพิ่มข้อมูลบทความวิจัยตีพิมพ์ โดยสมาชิกสามารถกรอกข้อมูลและอัปโหลดไฟล์เพื่อเผยแพร่บทความวิจัยตีพิมพ์ผ่านทางเว็บไซต์ได้
- ส่วนเพิ่มโปรแกรมและไฟล์ โดยสมาชิกสามารถอัปโหลดไฟล์และโปรแกรมจากเครื่องตัวเองมาที่เซิร์ฟเวอร์เพื่อเผยแพร่ไฟล์ต่างๆ ที่สำคัญได้ โดยจะแบ่งเป็นไฟล์ที่ให้ผู้ทั่วไปดาวน์โหลดได้ และไฟล์ภายในซึ่งให้สมาชิกที่ล็อกอินเข้ามาในเว็บไซต์เท่านั้นดาวน์โหลดได้
- ส่วนเวบบอร์ดสำหรับแสดงความคิดเห็นโดยจะแบ่งเป็นเวบบอร์ดสำหรับบุคคลทั่วไป และเวบบอร์ดภายในสำหรับสมาชิกที่ล็อกอินเข้ามาในระบบแล้วเท่านั้น
- ส่วนเพิ่มลิงค์ต่างๆ ที่น่าสนใจ โดยสมาชิกสามารถเพิ่มลิงค์ที่น่าสนใจ โดยสามารถสร้างหมวดลิงค์ และแบ่งประเภทของลิงค์เป็นหมวดหมู่ตามที่ต้องการได้
- ส่วนของสมุดเยี่ยมชม (Guestbook) สำหรับแนะนำติชมต่างๆ
- ส่วนของระบบแจ้งข่าวสารสำหรับสมาชิก (Mailing list)
- ส่วนของคำถามที่ถามบ่อย (Frequently Asked Questions : FAQ) โดยผู้ดูแลระบบสามารถสร้างหมวดพร้อมเพิ่มทั้งคำถาม-คำตอบในแต่ละหมวดได้ นอกจากนี้ยังสามารถแก้ไขหรือลบได้

- ส่วนของผู้ดูแลระบบที่สามารถติดตามการทำงานและจัดการข้อมูลทั้งหมดภายในเว็บไซต์ได้ ไม่ว่าจะเป็นการเพิ่มข้อมูล การลบข้อมูล การตรวจสอบว่าผู้ใช้งานไหนทำอะไรภายในเว็บไซต์บ้าง การเพิ่มหรือลบผู้ใช้งานจากระบบ การกำหนดรหัสผ่านให้ผู้ใช้ใหม่ การดูแลสถิติการใช้งานของเว็บไซต์รวมทั้งล็อกไฟล์ของเว็บไซต์ด้วย

9.3 สิทธิการใช้งานเว็บไซต์

เอกสารในแต่ละหน้าภายในเว็บไซต์นั้นจะสามารถกำหนดสิทธิการใช้งานได้ว่าผู้ใช้ระดับไหนมีสิทธิในการดูเอกสารนี้ โดยที่เวลาร้องขอเอกสารแต่ละหน้าของเว็บไซต์นั้นจะมีการตรวจสอบว่าผู้ที่ร้องขอนั้นมีสิทธิ์ดูข้อมูลในหน้านั้นหรือเปล่าถ้ามีสิทธิ์ก็จะแสดงข้อมูลในหน้านั้นออกมาทันที แต่ถ้าไม่มีสิทธิ์ก็จะแสดงแบบฟอร์มเพื่อให้ผู้ใช้ได้ล็อกอิน โดยผู้ใช้จะต้องใส่ชื่อและรหัสผ่านที่ถูกต้องจึงจะสามารถดูข้อมูลในหน้านั้นได้ สำหรับสิทธิการใช้งานเว็บไซต์ของผู้ใช้นั้น จะแบ่งสิทธิการใช้งานออกเป็น 3 ระดับดังต่อไปนี้

- **ผู้ใช้ทั่วไป**
สามารถดูข้อมูลทั่วไปจากเว็บไซต์ได้ เช่น โครงการของห้องวิจัย ข่าวประกาศ บทความ คาวนั โหลดโปรแกรม ข้อมูลของห้องวิจัย สมาชิกของห้องวิจัย แสดงความคิดเห็นต่างๆ เป็นต้น
- **สมาชิกห้องวิจัย**
สามารถดูข้อมูลได้เหมือนผู้ใช้ทั่วไป นอกจากนี้ยังสามารถเพิ่มหรือแก้ไขข้อมูลต่างๆ ในเว็บไซต์ได้ เช่น สามารถประกาศข่าวได้ เขียนหรือแก้ไขบทความได้ เพิ่มหรือแก้ไขลิงค์ เพิ่มหรือแก้ไขข้อมูลโปรเจกต์ได้ เปลี่ยนรหัสผ่านได้ เป็นต้น โดยสมาชิกจะต้องล็อกอินเข้าไปก่อน ถึงจะสามารถใช้คุณสมบัตินี้ได้
- **ผู้ดูแลระบบ**
มีสิทธิ์ทุกอย่างที่สมาชิกห้องวิจัยสามารถทำได้ นอกจากนี้ยังมีสิทธิ์พิเศษเพิ่มเติม ได้แก่ การเพิ่มผู้ใช้เข้ามาในระบบหรือลบผู้ใช้งานจากระบบ การแก้ไขหรือลบข้อมูลใดๆ อย่างภายในเว็บไซต์ได้ เช่น ลบกระทู้จากเว็บบอร์ด หรือลบบทความ สามารถดูสถิติหรือล็อกไฟล์ของเว็บไซต์ได้ เป็นต้น

9.4 โครงสร้างของเว็บไซต์

โดยโครงสร้างทั้งหมดของเว็บไซต์ แสดงดังตารางดังต่อไปนี้

	ผู้จัดทำไป	สมาชิกห้องวิจัย	ผู้ดูแลระบบ
ข่าวความเคลื่อนไหว	ดูได้	ประกาศหรือแก้ไขได้	ลบได้
บทความ	ดูได้	เขียนหรือแก้ไขได้	ลบได้
บทความตีพิมพ์	ดูได้	เพิ่มหรือแก้ไขได้	ลบได้
โครงการห้องวิจัย	ดูได้	เพิ่มหรือแก้ไขได้	ลบได้
จัดเก็บไฟล์ ทั่วไป	ดาวน์โหลดได้	เพิ่มหรือแก้ไขได้	ลบได้
จัดเก็บไฟล์ ภายใน	-	ดาวน์โหลด เพิ่มหรือแก้ไขได้	ลบได้
Webboard ทั่วไป	แสดงความคิดเห็นได้	แสดงความคิดเห็นได้	ลบกระทู้ได้
Webboard ภายใน	-	แสดงความคิดเห็นได้	ลบกระทู้ได้
ข้อมูลสมาชิกห้องวิจัย	ดูได้	แก้ไขข้อมูลได้	ลบได้
ข้อมูลอาจารย์	ดูได้	ดูได้	แก้ไขหรือลบได้
เพิ่มหรือแก้ไขข้อมูลผู้ใช้ในระบบ	-	-	เพิ่มหรือลบได้
เปลี่ยนรหัสผ่าน	-	ได้	ได้
กำหนดรหัสผ่านใหม่หรือกำหนดสิทธิใหม่	-	-	ได้
Login – logout	-	ได้ (ส่วนของสมาชิก)	ได้ (ส่วนของผู้ดูแลระบบ)
Link	ดูได้	เพิ่มหรือแก้ไขได้	ลบหรือย้ายหมวดได้
ค้นหาข้อมูล	ค้นหาได้	ค้นหาได้	ค้นหาได้
FAQ	ดูได้	ดูได้	เพิ่ม แก้ไขหรือลบได้
Logfile	-	-	ดูได้
สถิติของเว็บไซต์	-	-	ดูได้
ข้อมูลติดต่อ	ดูได้	ดูได้	แก้ไขได้
สมุดเยี่ยมชม	เขียน-ดูได้	เขียน-ดูได้	ลบได้
Mailing list	สมัคร-รับข่าวได้	สมัคร-รับข่าวได้	เขียนข่าวได้
รายงานข้อผิดพลาด	รายงานได้	รายงานได้	ตรวจสอบ ลบหรือแก้ไขได้
Review paper	-	Review หรือดูได้	ลบได้

ตาราง 9-1 แสดงโครงสร้างและสิทธิในการใช้งานของเว็บไซต์

9.5 ตัวอย่างความปลอดภัยของเว็บไซต์ห้องวิจัย

- ส่วนตรวจสอบสิทธิ์การใช้งานของผู้ใช้ โดยที่เอกสารแต่ละหน้าจะสามารถกำหนดได้ว่าผู้ใช้ระดับใดมีสิทธิ์ดูข้อมูลภายในเอกสารนี้ได้
- ความปลอดภัยในการจัดเก็บไฟล์และโปรแกรมต่างๆ ภายในเว็บไซต์ โดยจะใช้วิธีการเข้ารหัสชื่อไฟล์และไคลเอนทอร์ไว้ ทำให้ผู้ใช้ทั่วไปไม่สามารถรู้ได้ว่าเราจัดเก็บไฟล์ไว้ที่ไหน
- ความปลอดภัยในการรับส่งข้อมูลด้วย Session โดยตัวแปรต่างๆ ที่อยู่ใน Session จะมีการจัดเก็บไว้ที่ฝั่งเซิร์ฟเวอร์ ซึ่งในการติดต่อเพื่อเข้าถึงตัวแปรนี้จากฝั่งไคลเอนต์จะติดต่อโดยใช้ Session ID ที่มีการเข้ารหัสไว้ นอกจากนี้ Session ID นี้ยังมีการเข้ารหัสอีกครั้งด้วยโพรโทคอล SSL จึงเป็นการยากที่จะดักจับข้อมูล Session นี้ได้
- ความปลอดภัยของระยะเวลาในการใช้งานภายในเว็บไซต์ โดยเมื่อสมาชิก Login เข้ามาใช้งานภายในเว็บไซต์แล้ว โดยที่ทั้งหน้าจอไว้และไม่ได้ทำอะไร ระบบจะสามารถตรวจสอบและ Logoff ให้โดยอัตโนมัติ โดยสามารถกำหนดเวลาได้
- การกำหนดระดับของผู้ใช้ในการติดต่อกับฐานข้อมูล โดยที่เว็บไซต์แต่ละหน้าจะมีการใช้คำสั่ง Query ที่แตกต่างกัน ดังนั้นการกำหนด Username และ Password ที่ใช้ในการติดต่อกับฐานข้อมูลในระดับที่แตกต่างกัน ทำให้เว็บไซต์มีความปลอดภัยมากยิ่งขึ้น
- ความปลอดภัยของฐานข้อมูล โดยการกำหนดสิทธิ์ให้กับผู้ใช้โดยชื่อและรหัสผ่านที่กำหนดให้ นั้นมีผลเฉพาะฐานข้อมูลฯ หนึ่งเท่านั้น ซึ่งถ้าหากผู้บุกรุกได้ชื่อและรหัสผ่านนี้ไป ก็ไม่สามารถทำอะไรกับข้อมูลในฐานข้อมูลอื่นๆ ได้
- การป้องกันการส่งค่าตัวแปรมาทาง URL โดยวิธี post method สามารถป้องกันได้โดย การยกเลิกการสร้างตัวแปรโกลบอลที่ส่งมาทางโพรโทคอล HTTP โดยการปรับแต่งค่าในไฟล์คอนฟิกูเรชันของ PHP ทำให้สามารถป้องกันการส่งค่าตัวแปรมาได้
- การตรวจสอบ input ที่รับมาจากผู้ใช้โดยใช้ Regular Expression เพื่อป้องกันการพิมพ์ค่า input ที่แปลกๆ มาหรือส่งโปรแกรมเข้ามารันที่เซิร์ฟเวอร์
- หยุดการทำงานของ Script ทันทีเมื่อเกิดข้อผิดพลาดในการทำงาน ทำให้ Browser ส่งค่าว่างเปล่ากลับไปให้ผู้ใช้ โดยผู้บุกรุกจะไม่ได้รับอะไรไปเลย ไม่มีการแสดงผลข้อผิดพลาดต่างๆ
- การตรวจสอบว่าผู้ที่เข้ามาใช้งานในส่วนของผู้ดูแลระบบต้องมาจาก IP Address ภายในที่กำหนดไว้เท่านั้น
- การป้องกันการ Browser Directory ของ Apache
- การจัดเก็บ Logfile เพื่อติดตามการทำงานของผู้ใช้ ว่าใครทำอะไร ที่ไหน เมื่อไหร่ เวลาไหน
- การป้องกันการโจมตีแบบปฏิเสธการให้บริการ โดยการปฏิเสธคำร้องขอที่มีเข้ามาเป็นจำนวนมากในช่วงเวลาเดียวกันจากที่ใดที่หนึ่ง

161.246.5.02 - SecureCRT

File Edit View Options Transfer Script Window Help

mysql> select ID,Fullname,Username,password from accesslist;

ID	Fullname	Username	password
1	ÈÒ,Òp ¢ÑÇÒÇÑ²:ìpÀÐ;ÙÀ	itsource	2916a8566839c32f
37	ÇÒÀÇÑ²:ì ÈÀÒpDÀÒÀÑÀ	weerawat	73c1ed5e659dca90
17	»ÀÑ²-Ò	top	77987b801d8a9bd9
22	,Ò ÈSÈìÈØÇÀÀ²	thana	0e3a7bdf16b32cea
25	Kor Kittikorn	kor	5ae34a172dee9757
26	Phat R	tum	7a82fc8649fcafea
27	ÈÑ;´Òì²ÑÀ ¢ÒÀÈØÇÀÀ²	mint_gozzi	47d470e909e79a6d
30	ฆØ²ÈÀÒÇØ, À,´Òì²Ñ²:ì	o_isaG	650f3628774f7d0f
31	Apichon Witayungkurn	maykonovic	220437646d806acf
32	,²Ñ²-²ÑÀ pÀÒÀÒM	robiuz	0f264ee96914071b
33	Chawdee Baramee	chaw	7d434d616ab3a440
34	Phanita	kay+	66a3ad0c1f1d29cb
35	àÀ,Ò ¢ØÀÙ²²Ò	NeungBoy	0bd5a7f070def35d
36	Kanoksak Ratchapat	toon	0c40053d0be21d9f
38	ÇÑ²À pÑÈS²²Ò²ÒÀÒ²²:ì	pongkemon	4e6d087259d33fdc
39	Mr. Anan	ministry	47ad073909e9a9d5
41	àÇÀÀ²²:ì ÍÒÀÑ²ÀÍÑS;ÙÀ	ejel	61f0fbbcl0e05230
42	ÇÑ²ÀÑ;È²:ì à;àÀÈ	Ammy	78e3a5325f53d3f9
43	ÍÑSÈ²Ò ÇSÈìÀÑp²ÇÒ²ÒpÀ	Okeydoke	707cba7b6363620f
44	pÇS²ÀÒ²-ì àÈÒÀSÈ²²ÑÈ²	gatealcus	6ca938d12df7c1c4
45	²Ñ²À²Ò ÀÑÈ²-À²S	dear	2514998a1740cdb5
46	,ÒÀ²ÀSÈìÈÑ;´Òì àÈS²ÇÀÈÀÒ	kae	056f259573ecd51d
47	ÀÑp²²ÀÑ ÇÒàÈÈÀÈS	s1014357	400281027d482a60

Ready | ssh1:3DES | 31, 8 | 28 Rows, 78 Cols | VT100 | NUM

รูปที่ 9-1 แสดงการการเข้ารหัสผ่านที่เก็บไว้ในฐานข้อมูลทำให้มีความปลอดภัยมากขึ้น

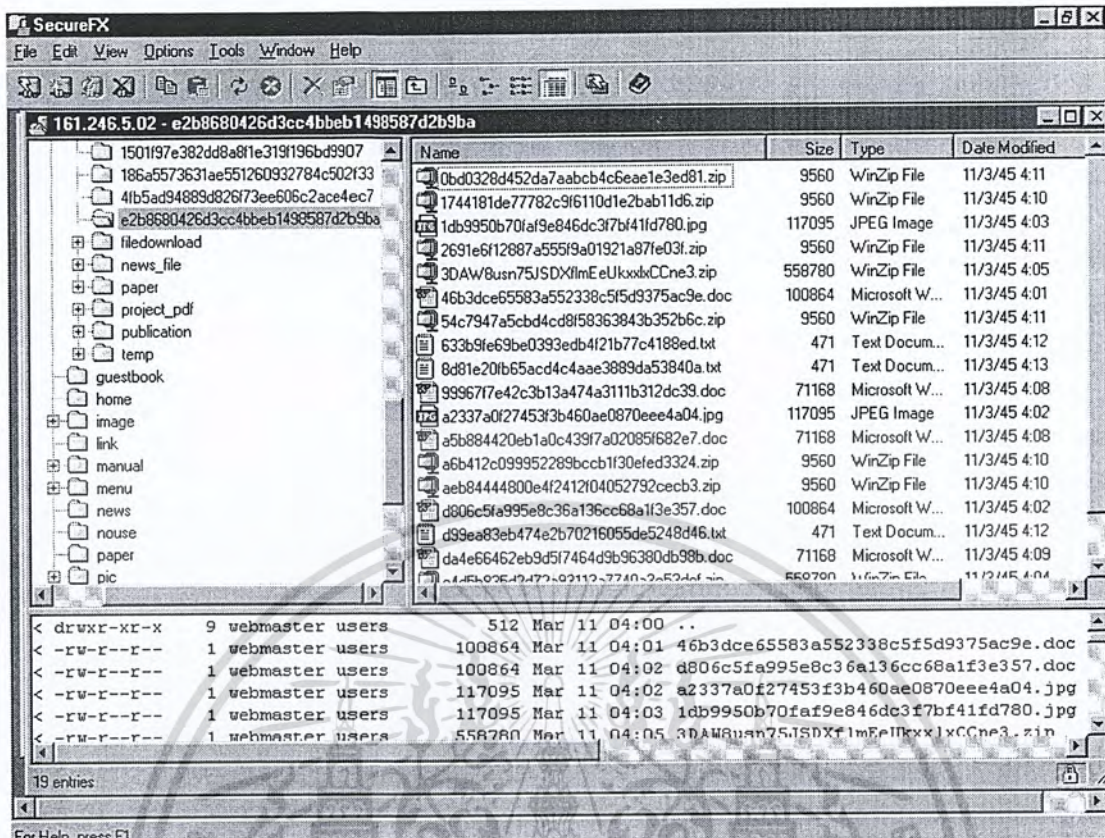
EditPlus - [C:\Program Files\Apache Group\Apache\logs\error.log]

File Edit View Search Document Project Tools Window Help

7846	[Thu Nov 15 16:26:54 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/password.dat
7847	[Thu Nov 15 16:26:54 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/password.data
7848	[Thu Nov 15 16:26:55 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/password.db
7849	[Thu Nov 15 16:26:55 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/password.dbf
7850	[Thu Nov 15 16:26:55 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/password.exe
7851	[Thu Nov 15 16:26:55 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/password.htm
7852	[Thu Nov 15 16:26:55 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/password.html
7853	[Thu Nov 15 16:26:55 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/password.htx
7854	[Thu Nov 15 16:26:55 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/password.ini
7855	[Thu Nov 15 16:26:56 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/password.lst
7856	[Thu Nov 15 16:26:56 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.php3
7857	[Thu Nov 15 16:26:56 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/password.pl
7858	[Thu Nov 15 16:26:56 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/password.txt
7859	[Thu Nov 15 16:26:57 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.asp
7860	[Thu Nov 15 16:26:57 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.cfg
7861	[Thu Nov 15 16:26:57 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.cgi
7862	[Thu Nov 15 16:26:57 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.dat
7863	[Thu Nov 15 16:26:57 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.data
7864	[Thu Nov 15 16:26:57 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.db
7865	[Thu Nov 15 16:26:58 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.dbf
7866	[Thu Nov 15 16:26:58 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.exe
7867	[Thu Nov 15 16:26:58 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.htm
7868	[Thu Nov 15 16:26:58 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.html
7869	[Thu Nov 15 16:26:58 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.htx
7870	[Thu Nov 15 16:26:58 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.ini
7871	[Thu Nov 15 16:26:58 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.lst
7872	[Thu Nov 15 16:26:58 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.php3
7873	[Thu Nov 15 16:26:59 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.pl
7874	[Thu Nov 15 16:26:59 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/passwords.txt
7875	[Thu Nov 15 16:26:59 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/store.asp
7876	[Thu Nov 15 16:26:59 2001] [error] [client 161.246.5.12] File does not exist: c:/isagweb/cgiis/store.cnf

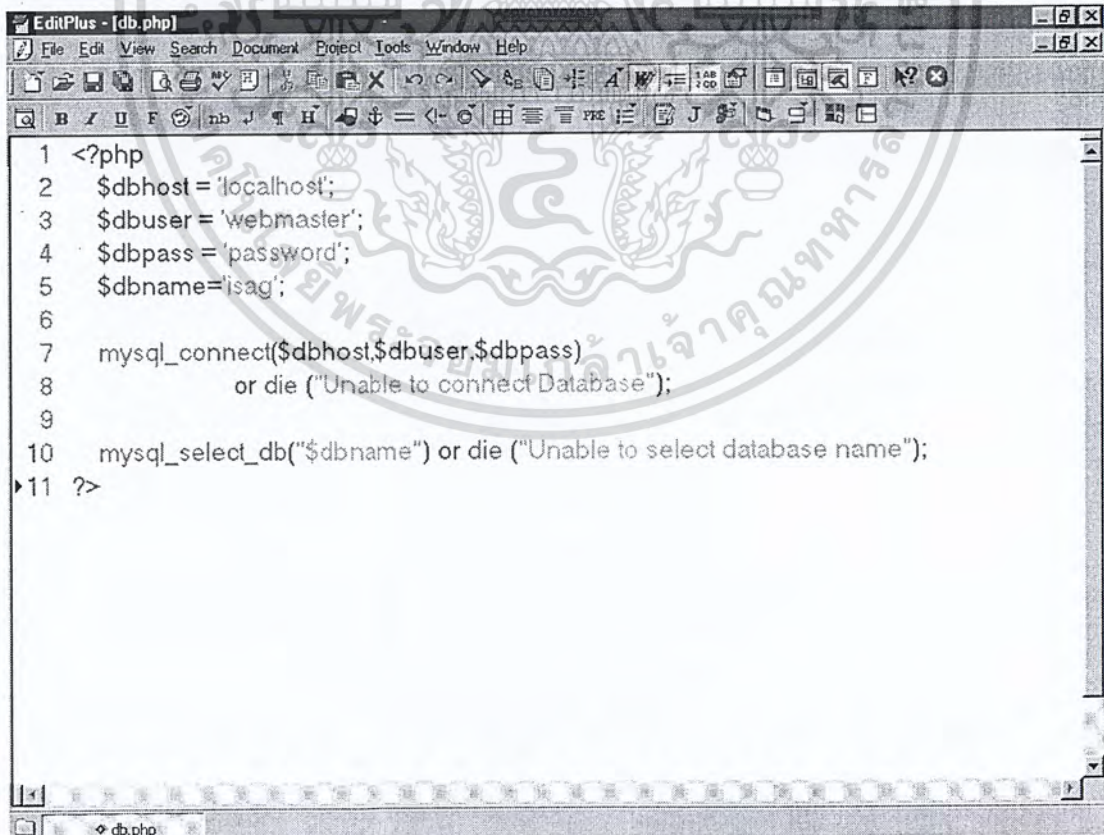
test.php | error.log

รูปที่ 9-2 แสดงการใช้เครื่องมือในการสแกนหาไฟล์ password ที่เก็บไว้ในเซิร์ฟเวอร์

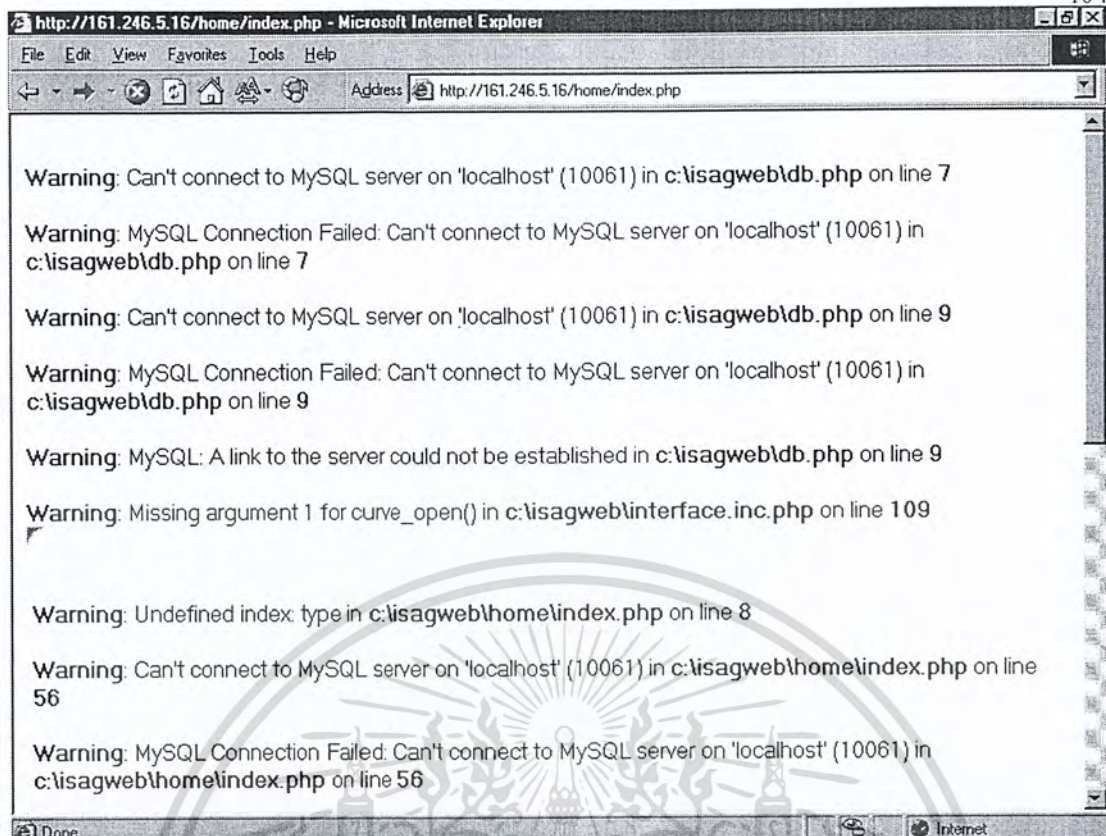


For Help, press F1

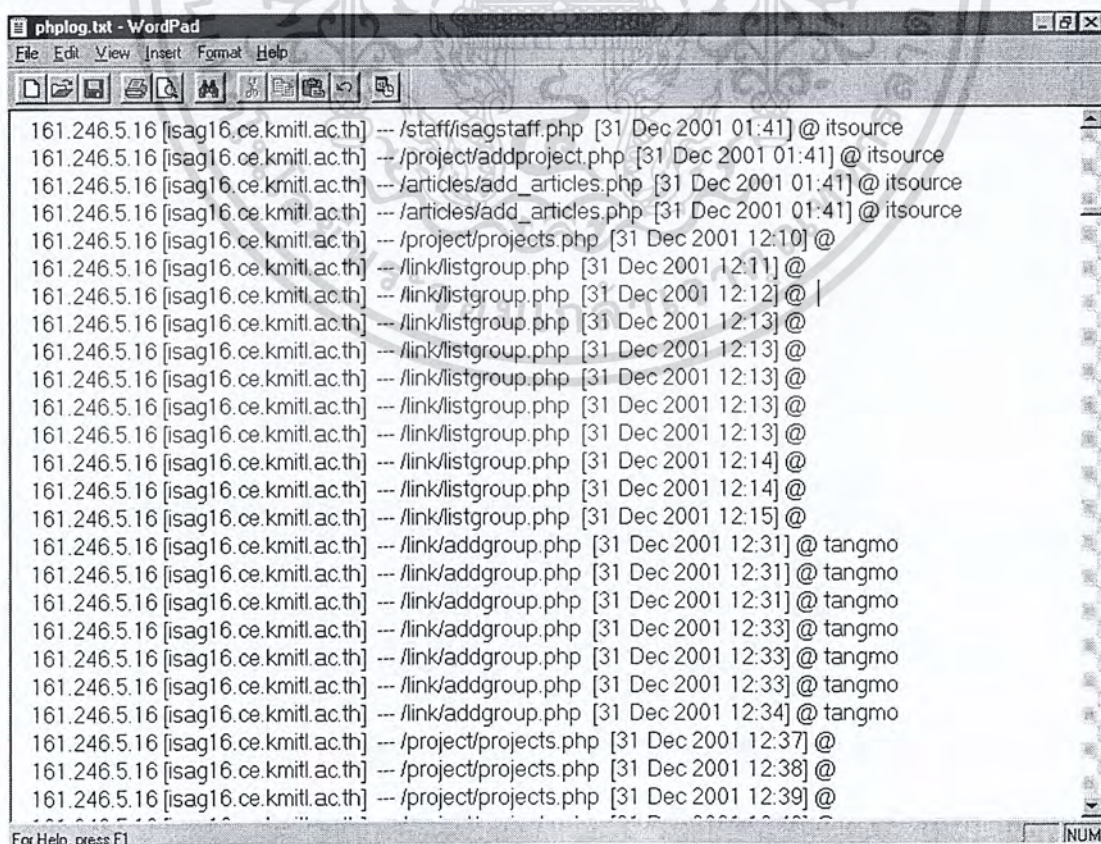
รูปที่ 9-3 แสดงการเข้ารหัสไฟล์และไดเรกทอรีเพื่อป้องกันไฟล์และและข้อมูลในไดเรกทอรีที่สำคัญ



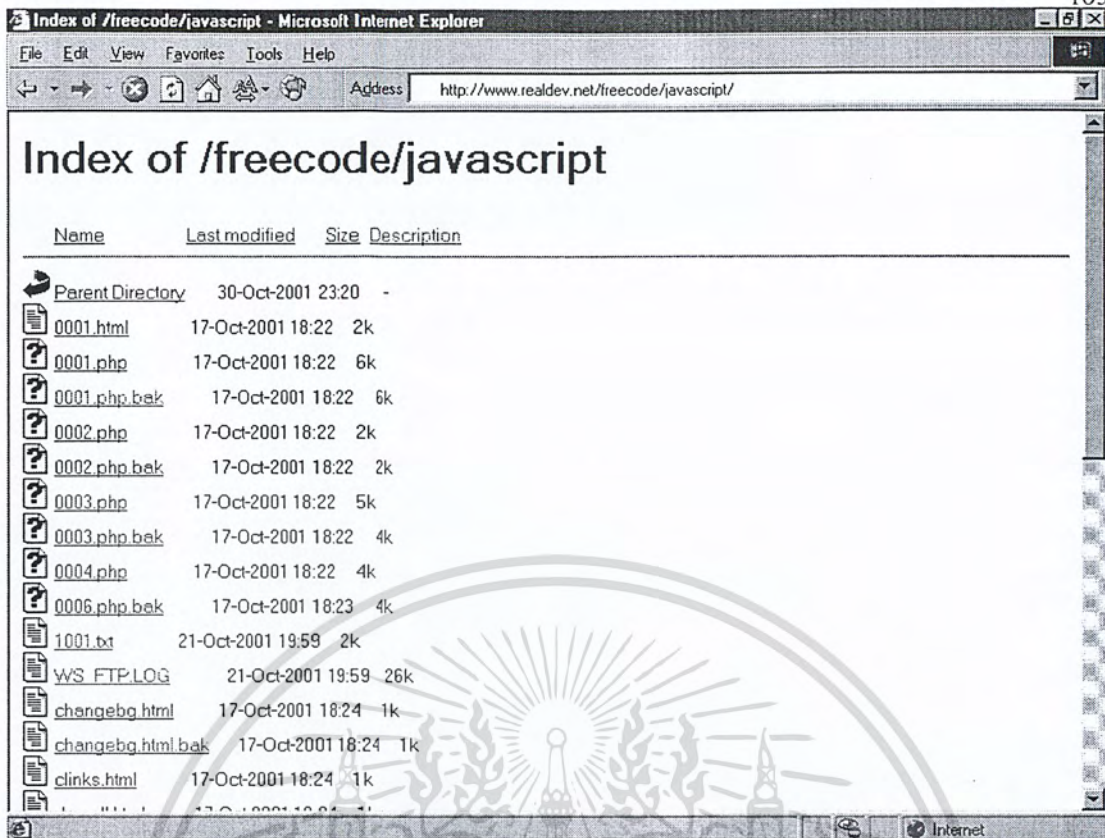
รูปที่ 9-4 แสดงการใช้ชื่อและรหัสที่แตกต่างกันในการติดต่อกับฐานข้อมูลในแต่ละไฟล์ที่มีความสำคัญต่างกัน



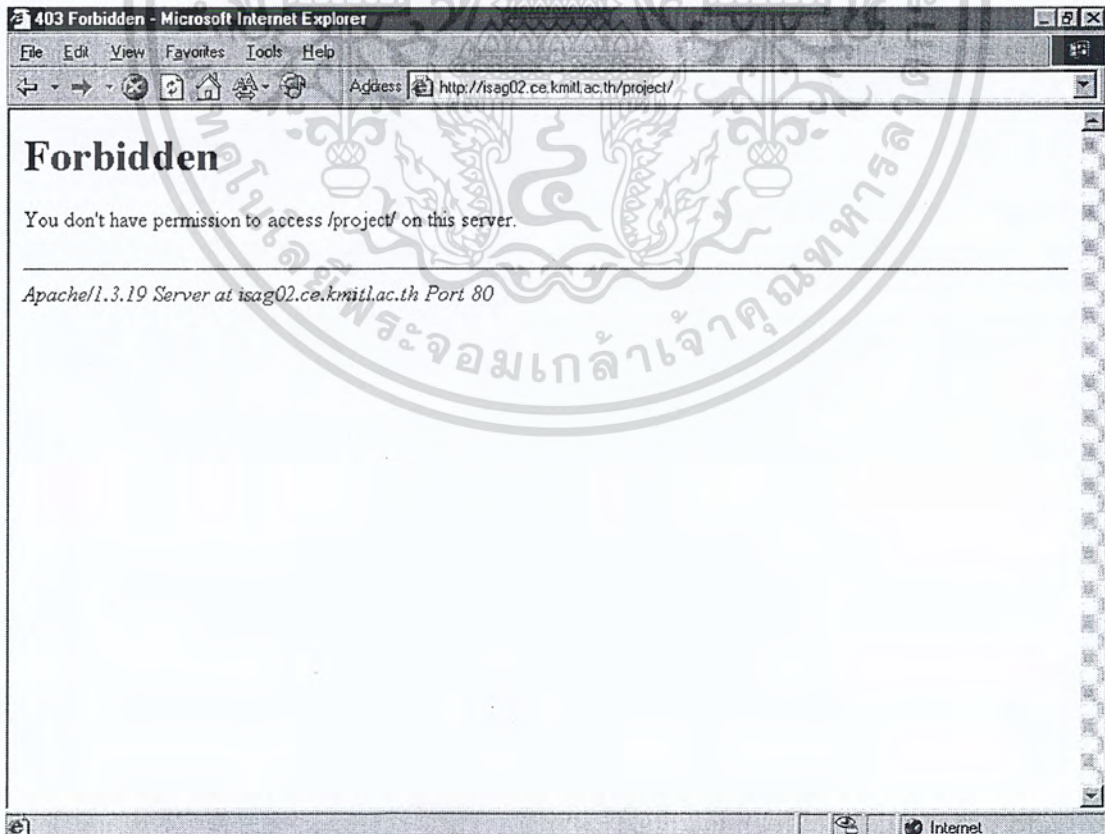
รูปที่ 9-5 แสดงข้อผิดพลาดที่เกิดขึ้นโดยที่ไม่ได้ปิดการแสดงผลข้อผิดพลาดไว้ ทำให้ผู้บุกรุกทราบข้อมูลเกี่ยวกับระบบได้ ซึ่งถ้าหากมีการปิดการแสดงผลข้อผิดพลาดนี้ ผู้บุกรุกก็จะไม่ได้ข้อมูลอะไรไปเลย



รูปที่ 9-6 แสดงการใช้ล็อกไฟล์เพื่อให้สามารถตรวจสอบการทำงานของเว็บไซต์ได้



รูปที่ 9-7 แสดงไฟล์ที่อยู่ภายในไดเรกทอรีนั้นเมื่อไม่มีดีฟอลต์ไฟล์ ซึ่งทำให้ผู้บุกรุกสามารถดูไฟล์ทั้งหมดที่มีอยู่ได้



รูปที่ 9-8 แสดงการปรับแต่งค่าเพื่อไม่ให้แสดงชื่อไฟล์ในไดเรกทอรีเมื่อไม่มีดีฟอลต์ไฟล์ทำให้ระบบมีความปลอดภัยมากยิ่งขึ้น

Information Security Advisory Group (ISAG) - Microsoft Internet Explorer

Address: http://isag02.ce.kmit.ac.th/staff/staff.php?type=2544

ISAG Information Security Advisory Group

ห้องวิจัยระบบรักษาความปลอดภัยข้อมูล Information Security Advisory Group (ISAG)
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

Home | Login | Sitemap | Resource | Community | Search | Contact | FAQ

Community

Members
Webboard
Guestbook
Mailing list

Resource

News
Events
Projects
Publication
Download
Links
Articles

About ISAG

FAQ
Contact

นักศึกษาปริญญาตรี ปีการศึกษา 2544

ล้นคิ ชีระชัยเสนา	เด	tangmo@hotmail.com
ปรีชญา ชัยรัตน์	ท็อป	top_newcastle@hotmail.com
พิศสัย จายสุวรรณ	หมิ่น	mint_gozzi@hotmail.com
คุณศราวุธ ฤทธิพันธ์	oyOyo	o_litnant@yahoo.com
Apichon Witayungkum	Maykonovic	maykonovic@hotmail.com
ชาตรี บารมี	chaw	chawdee@thai.com
Phanita	key+	tweety@thai.com
วัชร ตั้งหาญยานนท์	ฟ้ง	vachara@doraimail.com
วิวัฒน์ ศรีทะลาลัย	เน็ค	weerawatnet@hotmail.com
วันลักษณ์ ไนงศ	แอม	ammandmacka@hotmail.com
อังสนา วงศ์รัตนวิจิตร	จิม	a-jip@lycos.com

สมาชิกห้องวิจัย

อาจารย์
นักศึกษาปริญญาโท
นักศึกษาปริญญาตรี
- 2545
- 2544

รูป 9-11 ตัวอย่างหน้าที่แสดงสมาชิกของห้องวิจัย

ISAG Webboard -> Display Topic - Microsoft Internet Explorer

Address: http://isag02.ce.kmit.ac.th/webboard/boardlist.php

ISAG Information Security Advisory Group

ห้องวิจัยระบบรักษาความปลอดภัยข้อมูล Information Security Advisory Group (ISAG)
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

Home | Login | Sitemap | Resource | Community | Search | Contact | FAQ

ISAG ->> General Webboard (จำนวนกระทู้ทั้งหมด 8)

New Topic

Page [1]

Thread	Starter	Replies	Views	Last Post
๑ ปรับปรุง	ธนา	1	5	13 Feb 2002, (14:23) by สาวิศ
๑ แนะนำเขียนเขียน	บุ๋ม	1	11	11 Feb 2002, (20:25) by ^-^
๑ test	top	1	12	8 Feb 2002, (12:08) by Thana
๑ ชีตถาบันการศึกษาไทยไม่หนุนเทคโนโลยีฮาร์ดแวร์	สำนักข่าว IT	1	8	31 Jan 2002, (00:07) by Tong+
๑ รายงานข่าวเกี่ยวกับไปรษณีย์ ของ อาจารย์ วรวัฒน์	คัมเม้งคอง	5	60	30 Jan 2002, (23:42) by คัมเม้งคอง
๑ ต้องการหาเพื่อนที่มีความสนใจทำไปรษณีย์เกี่ยวกับdatabase	พัน2P	6	23	11 Feb 2002, (16:17) by test3
๑ update ข้อมูลคีย์เก่า	ธ	5	35	26 Jan 2002, (05:52) by Idempotent
๑ คำเตือน เกี่ยวกับ อะมีบ้า และ แลคโตบาซิลลัส	สำนักข่าวรอยเตอร์	3	24	27 Jan 2002, (04:08) by glg

Page [1]

รูปที่ 9-12 ตัวอย่างหน้าที่แสดงบอร์ดความคิดเห็นต่างๆ

Information Security Advisory Group (ISAG) - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://isag02.ce.kmitl.ac.th/project/projects.php?id=62

ISAG
Information Security Advisory Group

ห้องวิจัยระบบรักษาความปลอดภัยข้อมูล Information Security Advisory Group (ISAG)
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

Home Login Sitemap Resource Community Search Contact FAQ

Community
Members
Webboard
Guestbook
Mailing list

Resource
News
Events
Projects
Publication
Download
Links
Articles

About ISAG
FAQ
Contact

โครงการวิจัยปีการศึกษา 2544

โปรแกรมจำลองการตั้งค่าอุปกรณ์เราเตอร์
Router Configuring Simulator

ระดับปริญญาตรี
อาจารย์ที่ปรึกษา
อาจารย์ธนา หงษ์สุวรรณ thana@ce.kmitl.ac.th
อาจารย์อัครเดช วัชรพงษ์ adek@ce.kmitl.ac.th

ผู้รับผิดชอบโครงการ
นางสาวพนิศา ตั้งวิทย์ไมโย รหัส 41014292 tweety@thai.com
นางสาวณัฏฐา ดินนามทอง รหัส 41014126 dear@csqbbs.student.kmitl.ac.th

รายละเอียดโครงการ
การติดตั้งเราเตอร์ไม่ใช่วiệcที่ยากเกินไปสำหรับนักศึกษาในระดับปริญญาตรี หรือชั้นสูงกว่านี้ แต่เนื่องจากเราเตอร์ในปัจจุบันมีราคาสูงมาก ทำให้โอกาสในการฝึกหัดใช้ให้เกิดความคุ้นเคยมีน้อยลง มีผลให้ผู้ที่มีความสามารถและความชำนาญในการติดตั้งเราเตอร์มีน้อยไปด้วย ทั้งๆ ที่ความต้องการมีมาก โดยตั้งเป้าหมายให้โปรแกรมนี้มีความสามารถที่จะติดตั้งค่าให้เราเตอร์ได้หลายตัว โดยไม่จำเป็นต้องมีเราเตอร์จริงๆ หรืออาจจะมีเราเตอร์จริงๆ เพียงตัวเดียวเพื่อฝึกฝนกับเครื่องจริงๆ ในขั้นสุดท้าย

ทำโปรแกรมจำลองการติดตั้งค่าเราเตอร์โดยอ้างอิงกับค่าส่งของจริงให้

ปีการศึกษา

2544
2543
2542
2541

Internet

รูป 9-13 ตัวอย่างหน้าที่แสดงโครงการของห้องวิจัย

Information Security Advisory Group (ISAG) - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://isag02.ce.kmitl.ac.th/link/listgroup.php?group=67

ISAG
Information Security Advisory Group

ห้องวิจัยระบบรักษาความปลอดภัยข้อมูล Information Security Advisory Group (ISAG)
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

Home Login Sitemap Resource Community Search Contact FAQ

Community
Members
Webboard
Guestbook
Mailing list

Resource
News
Events
Projects
Publication
Download
Links
Articles

About ISAG
FAQ
Contact

link หมวด Security News

◇ HERT
Hacker Emergency Response Team. The HERT goal is to provide accurate information about computer security vulnerabilities, provide incident response services to sites that have been the victims of attacks, publish security alerts and find new vulnerabilities.
by itsource (6 Feb 2002, 15:50)

◇ Security Beat
Security Beat, the portal for Security and Computer Security news and information
by itsource (6 Feb 2002, 15:50)

◇ Computer Security News Daily
Daily selections of news and other information pertaining to computer/network security.
by itsource (6 Feb 2002, 15:49)

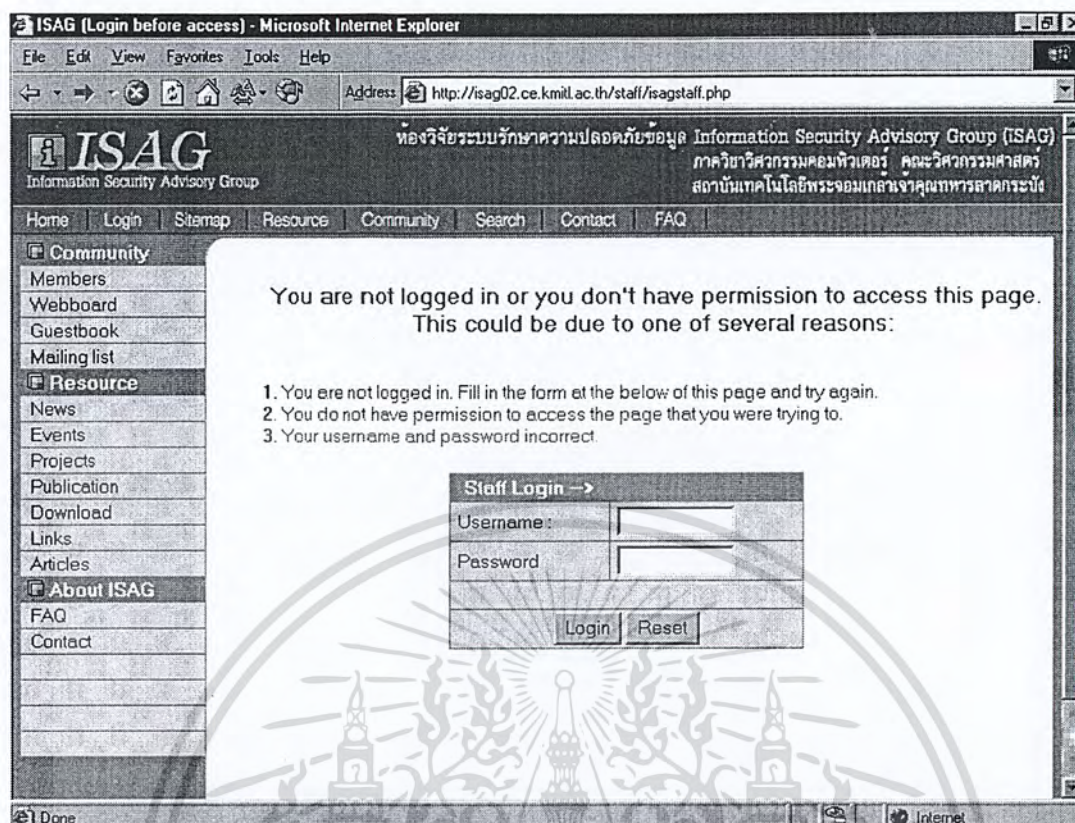
◇ Infoworld Weekly Security Review
This is a weekly article in infoworld that covers security news happenings and access to the archives.

Links

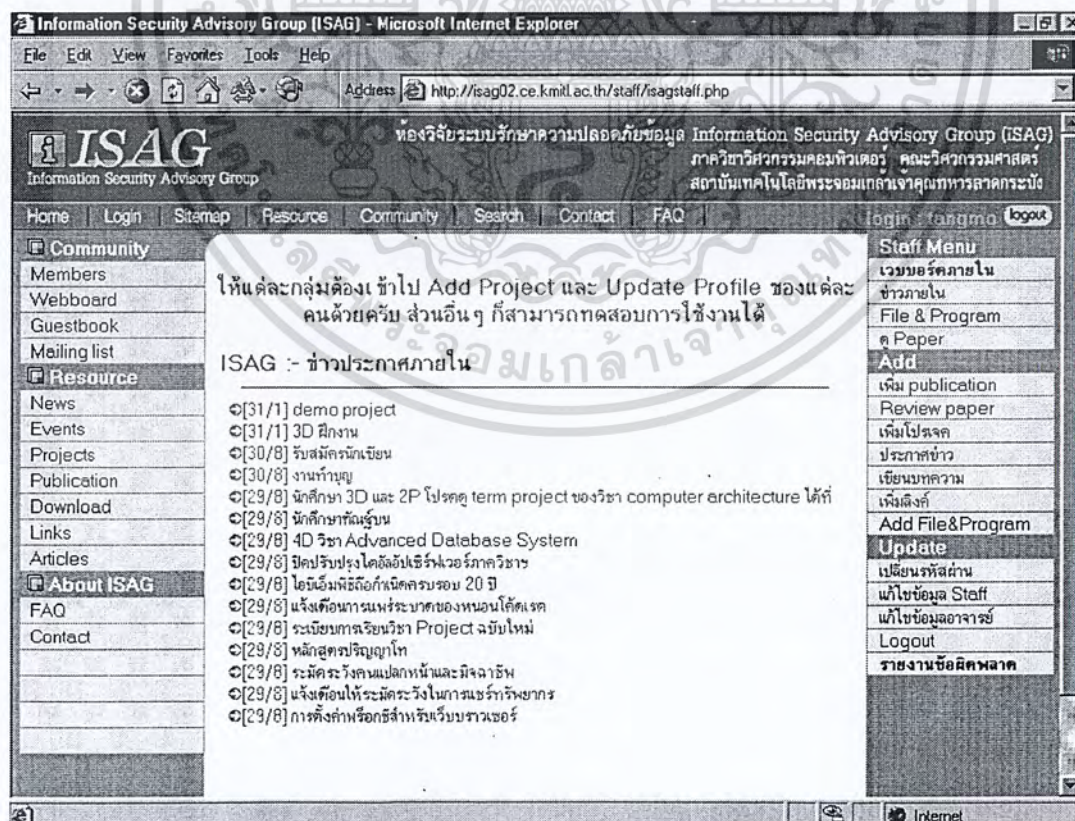
Authentication	2
Java	1
Public Key Infrastructure	4
Security News	10
windows2000	2

Done Internet

รูป 9-14 ตัวอย่างหน้าที่แสดงลิงค์ของห้องวิจัยแยกตามหมวดหมู่



รูป 9-15 ตัวอย่างหน้าจอ login เข้าสู่ระบบสำหรับสมาชิกเท่านั้น



รูป 9-16 ตัวอย่างหน้าจอที่แสดงข่าวประกาศภายในเมื่อสมาชิก login เข้าสู่ระบบแล้ว

Information Security Advisory Group (ISAG) - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://isag02.ce.kmitl.ac.th/news/add_news.php

ISAG
Information Security Advisory Group

ห้องวิจัยระบบรักษาความปลอดภัยข้อมูล Information Security Advisory Group (ISAG)
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

Home Login Sitemap Resource Community Search Contact FAQ

Community
Members
Webboard
Guestbook
Mailing list

Resource
News
Events
Projects
Publication
Download
Links
Articles

About ISAG
FAQ
Contact

Staff Menu
เว็บบอร์ดภายใน
ข่าวภายใน
File & Program
ด Paper
Add
เพิ่ม publication
Review paper
เพิ่มโปสเตอร์
ประกาศข่าว
เขียนบทความ
เพิ่มลิงค์
Add File&Program
Update
เปลี่ยนรหัสผ่าน
แก้ไขข้อมูล Staff
แก้ไขข้อมูลอาจารย์
Logout
รายงานข้อผิดพลาด

ประกาศข่าว

ประเภทข่าว *
หัวข้อข่าว *
เนื้อหาข่าว *

ไฟล์ประกอบข่าว (Maximum 30 MB) Browse... (ถ้าไม่มีแล้ว)

ประกาศถึงวันที่ *
ประกาศข่าวจนถึงวันที่ * 2002
(สำหรับข่าวที่มีกำหนดวันสิ้นสุดแน่นอน)

Links ที่เกี่ยวข้อง (ถ้าไม่มีแล้ว)

วันที่ประกาศข่าว 15/02/2002 เวลาประกาศข่าว 00:26

Done Internet

รูป 9-17 ตัวอย่างหน้าที่ใช้สำหรับประกาศข่าว

Information Security Advisory Group (ISAG) - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Address http://isag02.ce.kmitl.ac.th/project/addproject.php

ISAG
Information Security Advisory Group

ห้องวิจัยระบบรักษาความปลอดภัยข้อมูล Information Security Advisory Group (ISAG)
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

Home Login Sitemap Resource Community Search Contact FAQ

Staff Menu
เว็บบอร์ดภายใน
ข่าวภายใน
File & Program
ด Paper
Add
เพิ่ม publication
Review paper
เพิ่มโปสเตอร์
ประกาศข่าว
เขียนบทความ
เพิ่มลิงค์
Add File&Program
Update
เปลี่ยนรหัสผ่าน
แก้ไขข้อมูล Staff
แก้ไขข้อมูลอาจารย์
Logout
รายงานข้อผิดพลาด

เพิ่มโครงการวิจัย

ชื่อโครงการวิจัย (Thai) ระดับ ปฏิญญาไตร
ชื่อโครงการวิจัย (Eng) ปริญญาตรี
อาจารย์ที่ปรึกษา อาจารย์ธนา พงษ์สุวรรณ E-mail thana@ce.kmitl.ac.th
อาจารย์ที่ปรึกษา อาจารย์อัครเดช วัชรพงษ์ E-mail adek@ce.kmitl.ac.th
ผู้จัดทำโครงการวิจัย (ไม่ต้องใส่เลขลำดับ) รหัส E-mail
รหัส E-mail
ความสำคัญและที่มา (ไม่เกิน 5 บรรทัด) (ไม่ต้องย่อหน้า)
(เว้นช่องว่าง 1 บรรทัด)
รายละเอียดของโครงการ (ไม่เกิน 5 บรรทัด)
ไฟล์เนื้อหา PDF

Done Internet

รูป 9-18 ตัวอย่างหน้าที่ใช้สำหรับเพิ่มข้อมูลโครงการ

บทที่ 10

บทวิจารณ์และสรุป

10.1 สรุปผลการดำเนินงาน

ในการที่จะป้องกันปัญหาความปลอดภัยที่เกิดขึ้นนั้น จำเป็นที่จะต้องทราบลักษณะของปัญหา และสาเหตุที่แท้จริงเพื่อที่จะสามารถป้องกันและแก้ไขได้อย่างมีประสิทธิภาพ ซึ่งผู้จัดทำโครงการนี้ได้ตระหนักและทราบถึงผลกระทบที่เกิดขึ้น จึงได้ทำการศึกษาค้นคว้าการทำงานและลักษณะปัญหาที่เกิดขึ้นทั้งทางด้านฝั่งเซิร์ฟเวอร์ ฝั่งไคลเอนต์ และคอนเนกชันระหว่างไคลเอนต์และเซิร์ฟเวอร์ ซึ่งจากการศึกษานั้นทำให้ได้ทราบถึงลักษณะปัญหาในรูปแบบต่างๆ พร้อมทั้งแนวทางแก้ไขปัญหา ซึ่งผู้จัดทำก็ได้ใช้เป็นแนวทางในการพัฒนาเว็บไซต์ของห้องวิจัยระบบรักษาความปลอดภัยข้อมูล ซึ่งเป็นเว็บไซต์ที่เผยแพร่ข้อมูลข่าวสารทางด้านระบบรักษาความปลอดภัยให้แก่ผู้ที่สนใจ โดยการจัดตั้งเว็บไซต์ขึ้นมานี้ก็เน้นถึงความปลอดภัยของระบบที่จัดตั้งขึ้นมาด้วย โดยจะเป็นการนำความรู้เรื่องระบบรักษาความปลอดภัยที่ได้ทำการศึกษามาทั้งหมดมาประยุกต์ใช้ ตั้งแต่เว็บเซิร์ฟเวอร์ ดาต้าเบสเซิร์ฟเวอร์ และเซิร์ฟเวอร์ไซด์สคริปต์ การเข้ารหัสแบบต่างๆ ซึ่งก็จะเป็นเว็บไซต์ที่มีความปลอดภัยสูงภายในตัว

สำหรับภายในเว็บไซต์ที่จัดตั้งขึ้นนั้นก็ก็จะประกอบด้วยข้อมูลข่าวสารต่างๆ บทความที่น่าสนใจ ไฟล์และโปรแกรมที่สำคัญ บทความตีพิมพ์ โครงการของห้องวิจัย จดหมายข่าว ลิงค์ที่น่าสนใจรวมทั้งกระดานแสดงความคิดเห็น และอื่นๆ ที่น่าสนใจ ซึ่งเป็นประโยชน์ต่อผู้ที่สนใจเป็นอย่างยิ่ง ซึ่งก็ประสบผลสำเร็จเป็นที่น่าพอใจ เพราะนอกจากเว็บไซต์จะปลอดภัยในระดับหนึ่งแล้ว ยังเป็นแหล่งข้อมูลสำคัญในการเผยแพร่ข้อมูลเกี่ยวกับความปลอดภัยให้กับบุคคลที่สนใจด้วย

10.2 แนวทางการพัฒนาต่อ

เว็บไซต์ที่ได้จัดทำขึ้นนั้นได้มีการเปิดให้บริการ โดยให้สมาชิกในห้องวิจัยทดสอบการทำงานในส่วนต่างๆ เพื่อหาข้อผิดพลาดที่อาจเกิดขึ้นได้อันจะส่งผลกระทบต่อระบบ ซึ่งผู้จัดทำก็ได้ทำการแก้ไขเป็นระยะๆ ซึ่งการที่จะทราบข้อผิดพลาดหรือปัญหาที่เกิดขึ้นนั้นภายในระบบได้นั้น ส่วนหนึ่งก็เกิดจากการทดสอบและใช้งานไปเรื่อยๆ จากผู้ใช้อย่างหลายๆ คน สำหรับแนวทางการพัฒนาเว็บไซต์เพื่อให้มีความปลอดภัยยิ่งขึ้น มีดังต่อไปนี้ คือ

1. การจัดเก็บข้อมูลเซชันลงฐานข้อมูล ซึ่งจากเดิมที่เก็บไว้เป็นไฟล์ โดยการจัดเก็บลงฐานข้อมูลนี้จะมีประโยชน์ในการจัดการกับข้อมูลเซชันมากขึ้น เช่น การตรวจสอบว่ามีการล็อกอินเข้ามาในเว็บไซด์จากผู้ใดคนหนึ่งหรือหลายคนหลายที่หรือไม่ การตรวจสอบว่าขณะนี้มีผู้ที่ล็อกอินเข้ามาในเว็บไซด์อยู่กี่คน เป็นต้น
2. การจัดทำ logfile ที่สมบูรณ์มากขึ้น โดยที่จากเดิมจะเป็นการตรวจสอบเพียงแค่ว่าไอพีแอดเดรสจากที่ไหน ชื่อเครื่องอะไร เข้ามาร้องขอไฟล์ไหน วันเวลาใด และได้ login เข้ามาหรือไม่ ก็จะเขียนให้ตรวจสอบข้อมูลอย่างละเอียดมากขึ้นในส่วนของสมาชิกที่สามารถเข้ามา

เพิ่มข้อมูลลงในฐานข้อมูลได้ ว่าเพิ่มข้อมูลสำเร็จหรือเปล่าหรือว่าแค่เข้ามาดูในหน้านั้นเฉยๆ รวมถึงสามารถตรวจสอบผู้ไ้รายบุคคลได้ ว่าผู้ไ้คนนี้เคยเข้ามาทำอะไรในเว็บไซต์บ้าง

3. การตรวจสอบผู้บุกรุก โดยทำการตรวจสอบว่ามีผู้ไ้คนไหนกำหนดตัวแปรและส่งตัวแปรผ่านทางยูอาร์แอล หรือว่ามีการร้องขอไฟล์ที่ไม่มีอยู่ในระบบ ก็จะทำการบันทึกข้อมูลของผู้ไ้คนนั้นเก็บไว้เพื่อตรวจสอบต่อไป
4. การตรวจสอบและจัดทำสถิติการใช้งานเว็บไซต์ของผู้ไ้
5. การนำ PGP Public key มาประยุกต์ใช้กับระบบแจ้งข่าวสารของเว็บไซต์โดยอัตโนมัติ โดยขึ้นอยู่กับผู้ที่ล็อกอินเข้ามาภายในเว็บไซต์เป็นหลัก
6. ปรับปรุงการทำงานในส่วนต่างๆ ของสมาชิก โดยการออกแบบเพื่อให้มีรูปแบบการใช้งานที่ง่ายขึ้น เช่น ในส่วนของการเขียนบทความลงแบบฟอร์มบนเว็บ โดยอาจจะออกแบบในรูปแบบวิช่วลเพื่อให้่ายต่อการใช้งาน เป็นต้น



บรรณานุกรม

- [1] Aviell Rubin and Geer and Marcus Ranum, "Web Security Sourcebook", John Wile and Sons, 1997
- [2] Bellovin, Steve and Bill Chesweek, "Firewalls and Internet Security", Addison-Wesley, 1994
- [3] Carroll, John M., "Computer Security", Butterworth Publisher, 1987
- [4] Denning, Dorothy E.R., "Cryptography and Data Security", Addison-Wesley, 1983
- [5] Denning, Peter J., "*Computer Under Attack: Intruders, Worms and Viruses*", ACM Press/Addison-Wesley, 1990
- [6] Drew Dean and Edward W.Felten, "*Java Security: >From HotJava to Netscape and Beyond*", Oakland, 1996
- [7] Jesus Castagnetto, Harish Rawat, Sascha Schumann, Chris Scollo, Deepak Veliath, "Professional PHP Programming", Wrox, 1999
- [8] Lincoln Stein, "*Web Security: A step-by-step*", Addison-Wesley Longman, 1998
- [9] Luke Welling, Laura Thomson, "PHP and MySQL Web Development", SAMS, 2001
- [10] Mark Arnode, "Administering Apache", McGraw-Hill, 2000
- [11] Stephen A. Thomas, "HTTP Essential : Protocols for Secure, Scaleable Web Sites", Wiley Computer Publishing 2001
- [12] William Stallings, "Network Security Essentials : Application and Standards", Prentice Hall, 2000
- [13] ณรงค์ชัย นิมิตรบุญอนันต์, "*Computer Security for E-Commerce*", SUM Publishing Department, 2000
- [14] <http://www.apache.org>
- [15] <http://www.cert.org>
- [16] <http://developer.netscape.com/docs/manuals/security/sslin/contents.htm>
- [17] <http://developer.netscape.com/docs/manuals/security/pkin/contents.htm>
- [18] <http://www.devshed.com>
- [19] <http://www.google.com>
- [20] <http://www.microsoft.com>
- [21] <http://www.mysql.com>
- [22] <http://www.openbsd.org>
- [23] <http://www.php.net>

- [24] <http://www.securityfocus.com>
- [25] <http://security.e-matters.de/>
- [26] <http://www.w3c.org>
- [27] <http://www.webmasterbase.com>
- [28] <http://www.whatis.com>
- [29] <http://www zend.com>

