การพัฒนาโปรแกรมค้นหาและทำลายแอดแวร์และสปายแวร์ บนเครื่องคอมพิวเตอร์ส่วนบุคคล

ANTI ADWARE AND SPYWARE PROGRAM

นายจารุพงศ์ ว่องวุฒิพรชัย
นายบุญฤทธิ์ เพ็ญพิมล
นายวาทิต โลห์สิวานนท์

ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2548

ANTI ADWARE AND SPYWARE PROGRAM

JARUPONG VONGVUTHIPORNCHAI

BOONYARIT PHENPIMOL

WATIT LOHSIWANONT

A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR OF SCIENCE

DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE

FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2005

หัวข้อปัญหาพิเศษ	การพัฒนาโปรแกรมค้นหาและทำลายแอดแวร์และสปายแวร์บนเครื่องคอมพิวเตอร์ส่วนบุคคล ANTI ADWARE AND SPYWARE PROGRAM	
ชื่อนักศึกษา	นายจารุพงศ์ ว่องวุฒิพรชัย	45050462
	นายบุญฤทธิ์ เพ็ญพิมล	45050491
	นายวาทิต โลห์สิวานนท์	45050516
ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์	
สาขาวิชา	วิทยาการคอมพิวเตอร์	
อาจารย์ที่ปรึกษา	อ.ศังกรศรัณย์ ล่องชูผล	

ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้นับปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์ ประจำปีการศึกษา 2548

คณะกรรมการสอบ		ลายมือชื่อ
ประธานกรรมการ	อ. วิสันต์ ตั้งวงษ์เจริญ	
กรรมการ	รศ. ไพรบูลย์ พันธรักษ์พงษ์	
กรรมการและอาจารย์ที่ปรึกษา	อ.ศังกรศรัณย์ ล่องชูผล	

(รองศาสตราจารย์ ดร.วีระ บุญจริง)
หัวหน้าภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์

หัวข้อปัญหาพิเศษ	การพัฒนาโปรแกรมค้นหาและทำลายแอดแวร์และสปายแวร์บนเครื่องคอมพิวเตอร์ส่วนบุคคล ANTI ADWARE AND SPYWARE PROGRAM	
ชื่อนักศึกษา	นายจารุพงศ์ ว่องวุฒิพรชัย	45050462
	นายบุญฤทธิ์ เพ็ญพิมล	45050491
	นายวาทิต โลห์สิวานนท์	45050516
ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์	
สาขาวิชา	วิทยาการคอมพิวเตอร์	
ปีการศึกษา	2548	
อาจารย์ที่ปรึกษา	อ.ศังกรศรัณย์ ล่องชูผล	

บทคัดย่อ

ในทุกวันนี้ ระบบคอมพิวเตอร์ได้พัฒนาไปมาก โดยเฉพาะด้านการติดต่อสื่อสารและการเข้าถึงระหว่างกันในเครือข่าย แต่ในขณะเดียวกันความกังวลเกี่ยวกับความเป็นส่วนตัวและความปลอดภัยกลับเพิ่มสูงขึ้น เพราะในขณะนี้การเพิ่มขึ้นของมัลแวร์ ทั้งสปายแวร์และแอดแวร์ ได้เข้ามาคุกคามการทำงานของคอมพิวเตอร์ และกำลังค่อยๆ บ่อนทำลายความเชื่อมั่นของผู้ใช้งานในระบบอินเตอร์เน็ต

ในการศึกษานี้ เราจึงได้พยายามที่จะพัฒนาซอฟต์แวร์ ที่สามารถค้นหาและกำจัดมัลแวร์ประเภทสปายแวร์และแอดแวร์ในเครื่องคอมพิวเตอร์ส่วนบุคคลที่ใช้ระบบปฏิบัติการไมโครซอฟต์วินโดวส์ โดยในการศึกษานี้เราไม่เพียงแต่ตรวจสอบตัวไฟล์ที่เป็นสปายแวร์และแอดแวร์เพียงอย่างเดียวเท่านั้น เรายังได้ทำการเจาะลึกไปถึงการตรวจสอบสปายแวร์และแอดแวร์ที่ฝังตัวอยู่ในรีจิสตรี และคุ้กกี้ที่อาจเป็นที่แอบแฝงการทำงานของสปายแวร์และแอดแวร์โดยที่ผู้ใช้ไม่รู้ตัว ทั้งนี้ก็เพื่อสร้างความเชื่อมั่นและเพิ่มความปลอดภัยในระบบคอมพิวเตอร์ที่ดียิ่งขึ้น

Special Topic	ANTI ADWARE AND SPYWARE PROGRAM
Students	Mr.Jarupong Vongvuthipornchai 45050462
	Mr.Boonyarit Phenpimol 45050491
	Mr.Watit Lohsiwanont 45050516
Degree	Bachelor of Science
Department	Mathematics and Computer Science, Faculty of Science
Programme	Computer Science
Academic Year	2005
Special Project Advisor	Sungkornsarun Longchupol

ABSTRACT

Nowadays, Computer has increased connectivity and network accessibility. Privacy and security concerning has grown. An increasing variety of malware, such as spyware and adware, threatens computing and quietly undermines users' confidence in the Internet.

As increasing of spyware and adware, this study aims to develop software that scans spyware and adware and removes them in personal computers. We not only scan spyware and adware files, But also registry and cookie files that might be infected cause by spyware and adware without users' knowledge. All of above is for improving users' confidence and trust in online privacy and security.

กิตติกรรมประกาศ

โครงงานปัญหาพิเศษเรื่อง การพัฒนาโปรแกรมค้นหาและทำลายแอดแวร์และสปายแวร์บนเครื่องคอมพิวเตอร์ส่วนบุคคล สามารถสำเร็จลุล่วงไปได้ด้วยดี ด้วยความช่วยเหลือและความร่วมมือจากหลายๆ ท่าน คณะผู้จัดทำต้องขอขอบพระคุณ อ.ศังกรศรัณย์ ล่องชูผล ซึ่งเป็นอาจารย์ที่ปรึกษาโครงงานปัญหาพิเศษนี้ที่กรุณาให้คำแนะนำในการแก้ปัญหาต่างๆ คอยดูแลเอาใจใส่และให้การสนับสนุนทางด้านทางด้านซอฟต์แวร์และฮาร์ดแวร์ รวมทั้งเป็นผู้ตรวจสอบความถูกต้องของโครงงานพิเศษฉบับนี้

นอกจากนี้คณะผู้จัดทำต้องขอขอบพระคุณ บิดา มารดา ที่ได้ให้ความสนับสนุนทางด้านกำลังใจและทุนทรัพย์ จนการทำปัญหาพิเศษนี้สำเร็จลุล่วงไปด้วยดี รวมทั้งเพื่อนๆ พี่ๆ ทุกคนที่ให้ความช่วยเหลือในด้านต่างๆ เกี่ยวกับปัญหาพิเศษไว้ ณ ที่นี้

นายจารุพงศ์ ว่องวุฒิพรชัย
นายบุญฤทธิ์ เพ็ญพิมล
นายวาทิต โลห์สิวานนท์
มีนาคม 2549

สารบัญ

สารบัญรูป

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เครื่องคอมพิวเตอร์ส่วนบุคคลโดยปกติแล้วจะมีระดับความปลอดภัยต่ำ เนื่องจากระบบปฏิบัติการที่ใช้ส่วนใหญ่จะเป็นลักษณะที่เน้นความสะดวกต่อผู้ใช้เป็นหลัก โดยไม่ค่อยให้ความสนใจในเรื่องของความปลอดภัยในเครือข่ายมากนัก

ซึ่งเมื่อนำเครื่องคอมพิวเตอร์ส่วนบุคคลที่มีมาตรการการรักษาความปลอดภัยที่ต่ำ แต่นำมาเชื่อมต่อกับเครือข่ายอินเตอร์เน็ตโดยตรงนั้น เป็นการกระทำที่เสี่ยงเป็นอย่างมาก เพราะในทุกวันนี้โปรแกรมที่ไม่หวังดีต่อระบบนั้นเพิ่มมากขึ้นตลอดเวลา และทำให้ระบบเกิดความเสี่ยงที่จะถูกโปรแกรมเหล่านั้น มาสร้างปัญหาให้กับการทำงานของผู้ใช้ระบบ โดยการแฝงกายเข้ามาในรูปแบบต่างๆ โดยที่ผู้ใช้งานไม่รู้ตัว โดยจะเรียกโปรแกรมเหล่านี้รวมๆ ว่า มัลแวร์ (Malware)

จากปัญหาที่กล่าวมาข้างต้น ทำให้กลุ่มผู้พัฒนาเกิดแนวความคิด ที่จะทำการศึกษาและพัฒนาโปรแกรมที่มีความสามารถในการค้นหาและทำลายมัลแวร์ขึ้น แต่จากการที่มัลแวร์ได้มีความหลากหลายในการทำงาน จึงทำให้ทางกลุ่มผู้พัฒนาได้มีการคัดเลือกมัลแวร์เพียงบางประเภท ที่จะนำมาใช้เป็นต้นแบบในการค้นหามัลแวร์ที่ฝังตัวอยู่ภายในเครื่อง ซึ่งมัลแวร์ต้นแบบนี้จะถูกจัดอยู่ในกลุ่มของ แอดแวร์ (Adware) และสปายแวร์ (Spyware)

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

1.2.1 เพื่อลดการสูญเสียทรัพยากรทางคอมพิวเตอร์ อันเนื่องมาจากการทำงานของแอดแวร์และสปายแวร์ ทั้งการทำงานที่เข้ามาทำลายระบบและทำการทำงานโปรแกรมที่ไม่เป็นประโยชน์ต่อผู้ใช้

1.2.2 เพื่อเพิ่มความปลอดภัยในระบบคอมพิวเตอร์ ในด้านความเป็นส่วนตัวและความลับของข้อมูล

1.2.3 เพื่อให้เข้าใจถึงพฤติกรรมการทำงานของแอดแวร์และสปายแวร์

1.2.4 เพื่อให้ทราบถึงขั้นตอนและกรรมวิธีในการทำลายแอดแวร์และสปายแวร์

1.2.5 เพื่อเป็นแนวทางของการพัฒนาระบบรักษาความปลอดภัยให้แก่ผู้ที่ต้องการศึกษาและพัฒนาโปรแกรมที่เกี่ยวข้องต่อไป

1.3 ขอบเขตของปัญหาพิเศษ

โครงงานปัญหาพิเศษนี้ ได้มุ่งเน้นศึกษาในเรื่อง การพัฒนาโปรแกรมค้นหาและทำลายแอดแวร์และสปายแวร์บนเครื่องคอมพิวเตอร์ส่วนบุคคล และมีขอบเขตเพียงแค่ทำการค้นหาและทำลายเมื่อผู้ใช้ต้องการทำการค้นหาเท่านั้น

ในการออกแบบโปรแกรมค้นหาและทำลายแอดแวร์และสปายแวร์บนเครื่องคอมพิวเตอร์ส่วนบุคคลนี้ เราได้แบ่งส่วนออกเป็นส่วนใหญ่ๆ 3 ส่วนดังนี้

1.3.1 ส่วนของโปรแกรมหลักที่ใช้ในการค้นหาและทำลายแอดแวร์และสปายแวร์

ส่วนนี้เป็นส่วนที่ให้บริการผู้ใช้ทำการค้นหาและทำลายแอดแวร์และสปายแวร์ที่อยู่ในเครื่อง โดยมีส่วนประกอบย่อยเพื่อสนับสนุนการทำงาน เช่น การเก็บข้อมูลทางสถิติ การปรับแต่งการทำงานตามความเหมาะสม การอัพเดตฐานข้อมูลแอดแวร์และสปายแวร์ในเครื่อง และส่วนช่วยเหลือผู้ใช้งาน

1.3.2 ส่วนของเซิร์ฟเวอร์ผู้ให้บริการในการอัพเดตฐานข้อมูลแอดแวร์และสปายแวร์

ส่วนนี้เป็นส่วนที่ให้บริการผู้ใช้ที่ต้องการจะอัพเดตฐานข้อมูลแอดแวร์และสปายแวร์ โดยสามารถทำการอัพเดตฐานข้อมูลเหล่านั้นได้จากเซิร์ฟเวอร์ที่ให้บริการ

1.3.3 ส่วนของจัดการฐานข้อมูลแอดแวร์และสปายแวร์

ส่วนนี้เป็นส่วนที่สนับสนุนสำหรับผู้พัฒนาโปรแกรมเพื่อให้ผู้พัฒนาสามารถจัดการไฟล์ฐานข้อมูลแอดแวร์และสปายแวร์ได้สะดวกยิ่งขึ้น

1.4 ขั้นตอนการดำเนินงาน

1.4.1 ศึกษาและรวบรวมความต้องการของระบบ และเทคโนโลยีที่ใช้ในการพัฒนา

1.4.2 วิเคราะห์ความต้องการและออกแบบการทำงานของโปรแกรม

1.4.3 พัฒนาโปรแกรม

1.4.4 การทดสอบและปรับปรุงโปรแกรม

1.4.5 ประเมินผลงาน

1.4.6 การจัดการทำเอกสารประกอบการใช้งาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 เพิ่มความปลอดภัยบนเครื่องคอมพิวเตอร์ส่วนบุคคล

1.5.2 ลดค่าใช้จ่ายในการรักษาความปลอดภัยสำหรับเครื่องคอมพิวเตอร์ส่วนบุคคล

1.5.3 สามารถนำโปรแกรมที่ได้ทำการพัฒนาแล้ว ไปใช้ในการทำงานจริงได้อย่างมีประสิทธิภาพ

1.5.4 สามารถเข้าใจถึงการทำงานของแอดแวร์และสปายแวร์

1.5.5 สามารถนำความรู้ที่ได้ มาวิเคราะห์และแก้ไขปัญหาทางด้านความปลอดภัย

1.5.6 สามารถนำเอาแนวคิดไปพัฒนาโปรแกรมประยุกต์อื่นๆ ต่อไปได้

1.6 อุปกรณ์ที่ใช้ในการทำโครงงานพิเศษ

1.6.1 เครื่องคอมพิวเตอร์ไคลเอนต์

1.6.2 เครื่องคอมพิวเตอร์เซิร์ฟเวอร์

1.6.3 ฮาร์ดดิสก์

1.6.4 โมบายแลค

1.6.5 อุปกรณ์เชื่อมต่อเครือข่าย

1.6.6 โปรแกรม Microsoft Visual Studio .NET 2003

1.6.7 โปรแกรม Internet Information Services (IIS)

1.6.8 โปรแกรม HTML Help Workshop

บทที่ 2
ความรู้เบื้องต้น

2.1 ความรู้เบื้องต้นเกี่ยวกับโปรแกรมที่ไม่หวังดีต่อระบบ

2.1.1 ความหมายเบื้องต้นของโปรแกรมที่ไม่หวังดีต่อระบบ

โปรแกรมที่ไม่หวังดีต่อระบบ คือ โปรแกรมที่ไม่เป็นประโยชน์ต่อเครื่องคอมพิวเตอร์ และในบางโปรแกรมอาจถึงขั้นที่มีจุดประสงค์ร้ายต่อเครื่องคอมพิวเตอร์ รบกวนการทำงานทั้งเบื้องหน้าและเบื้องหลัง โดยเรามักเรียกโปรแกรมเหล่านั้นรวมๆ ว่า มัลแวร์

2.1.2 ประเภทของโปรแกรมที่ไม่หวังดีต่อระบบ

โปรแกรมที่ไม่หวังดีต่อระบบ สามารถจำแนกประเภทตามลักษณะการทำงานเป็น 10 ประเภท คือ

1) ไวรัส (Viruses)

ไวรัส คือ โปรแกรมคอมพิวเตอร์ประเภทหนึ่งที่ถูกออกแบบมาให้แพร่กระจายตัวเองจากไฟล์หนึ่งไปยังไฟล์อื่นๆ ภายในเครื่องคอมพิวเตอร์ ไวรัสจะแพร่กระจายตัวเองอย่างรวดเร็วไปยังทุกไฟล์ภายในคอมพิวเตอร์ หรืออาจจะทำให้ไฟล์เอกสารติดเชื้ออย่างช้าๆ แต่ไวรัสจะไม่สามารถแพร่กระจายจากเครื่องหนึ่งไปยังอีกเครื่องหนึ่งได้ด้วยตัวมันเอง โดยทั่วไปเกิดจากการที่ผู้ใช้เป็นพาหะ นำไวรัสจากเครื่องหนึ่งไปยังอีกเครื่องหนึ่ง เช่น เวลาที่ส่งอีเมล์โดยแนบเอกสาร หรือไฟล์ที่มีไวรัสไปด้วย การทำสำเนาไฟล์ที่ติดไวรัสไปไว้บนไฟล์เซิร์ฟเวอร์ การแลกเปลี่ยนไฟล์โดยใช้แผ่นดิสก์เก็ต เมื่อผู้ใช้ทั่วไปรับไฟล์หรือดิสก์มาใช้งาน ไวรัสก็จะแพร่กระจายภายในเครื่อง และจะเป็นวงจรในลักษณะนี้ต่อไป

2) หนอน (Worm)

หนอน เป็นสิ่งที่อันตรายต่อระบบมาก (สามารถทำความเสียหายต่อระบบได้จากภายในเหมือนกับหนอนที่กัดกินผลไม้จากภายใน) โดยทั่วไปก็จะคล้ายกับไวรัสคอมพิวเตอร์ และด้วยการอาศัยพฤติกรรมการทำงานของมนุษย์ยุคสารสนเทศ ในการแพร่กระจายตัวเองไปยังเครื่องคอมพิวเตอร์เครื่องอื่น หนอนร้ายเป็นโปรแกรมคอมพิวเตอร์ที่ถูกออกแบบมาให้สามารถแพร่กระจายตัวเองจากเครื่องคอมพิวเตอร์เครื่องหนึ่งไปยังอีกเครื่องหนึ่ง โดยอาศัยระบบเน็ตเวิร์ก หรืออีกนัยคือการกระจายผ่านจดหมายอิเล็กทรอนิกส์ ซึ่งการแพร่กระจายสามารถทำได้ด้วยตัวของมันเอง และจะแพร่กระจายได้อย่างรวดเร็วและทำความเสียหายรุนแรงกว่าไวรัสมาก

3) ม้าโทรจัน (Trojan)

ชื่อที่มาจากมหากาพย์อีเลียดที่กล่าวถึงเมืองทรอยของโฮมเมอร์ (Homer) ถูกนำมาใช้เป็นชื่อของโปรแกรมคอมพิวเตอร์ที่ถูกออกแบบมาให้แฝงตัว อำพรางตัวเองเข้าไปในระบบและจะทำงานโดยการดักจับเอารหัสผ่านเข้าสู่ระบบต่างๆ และส่งกลับไปยังผู้ประสงค์ร้าย เพื่อเข้าใช้หรือโจมตีระบบในภายหลัง ซึ่งอำพรางมาในหลายๆ รูปแบบ เช่นเกมส์ การ์ดอวยพรหรือจดหมายต่างๆ โปรแกรมโทรจันโดยทั่วไปจะอาศัยกลยุทธ์หลอกล่อผู้ใช้ให้เข้าถึง หรือติดกับดักที่ถูกส่งมาทางข้อความจดหมายอิเล็กทรอนิกส์ หรือการได้มาจากแหล่งที่มีความเสี่ยง โค้ดร้ายเหล่านี้จะถูกบรรจุ หรือส่งมากับข้อความที่ดูเหมือนจะธรรมดา แต่เมื่อเรียกใช้งานไฟล์เหล่านี้ โทรจันก็จะทำงาน และจะเปิดช่องทางต่างๆ ให้ผู้บุกรุกเข้าโจมตีระบบได้

4) ไวรัสข่าวหลอกลวง (Hoax)

ข่าวหลอกลวงเป็นวิธีการหรือการกุข่าวขึ้นมาสักเรื่องแล้วก็ส่งต่อๆ กันไปในระบบอินเทอร์เน็ตสักพักหนึ่งข่าวที่กุขึ้นมานี้ก็จะแพร่ไปในสังคมไอทีอย่างรวดเร็ว โดยรูปแบบอย่างหนึ่งที่ข่าวกุชอบใช้คือวิธี Social Engineer ที่อาศัยพฤติกรรมของคนเราเป็นตัวจักรในการสร้างความสับสน เช่น ข้อความแจ้งเตือน หรือจดหมายอิเล็กทรอนิกส์ ระบุว่ามีการค้นพบไวรัสคอมพิวเตอร์ในโทรศัพท์มือถือ และมีวิธีการทดสอบ เช่น หากคุณกดปุ่มตามที่อธิบายไว้ในจดหมาย และได้รับข้อความตามที่ระบุเครื่องคุณติดไวรัสให้รีบนำเครื่องเข้าศูนย์บริการ เป็นต้น ซึ่งจริงๆ แล้วการกดปุ่มบางอย่างของโทรศัพท์มือถือเป็นปุ่มคำสั่ง Service Code ซึ่งมีไว้สำหรับใช้ในศูนย์บริการเพื่อตรวจสอบการทำงานเบื้องต้นบางอย่างของโทรศัพท์เอง แต่สิ่งเหล่านี้สามารถสร้างความสับสนแก่ผู้คนจำนวนไม่น้อย

5) การฉ้อโกง (Scams)

รูปแบบของการฉ้อโกงไม่ได้ถูกจำกัดอยู่กับการฉ้อโกงแบบเดิม การฉ้อโกงสามารถเกิดขึ้นหรือกระทำขึ้นบนเครือข่ายอินเทอร์เน็ตได้เช่นกัน แต่ส่วนใหญ่วิธีการนี้มักเกิดขึ้นกับการทำธุรกรรมทางด้านการเงินหรือการใช้บัตรเครดิต ซึ่งวิธีการนี้เรียกว่า Phishing ตัวอย่างของ Phishing หรือการฉ้อฉลอันนี้คือการที่มีการสร้างจดหมาย ข้อความเลียนแบบ หรือรูปแบบการแจ้งข่าวสารของบริษัทที่มีชื่อเสียงอย่าง เช่น eBay เพื่อหลอกล่อเอาข้อมูลบางอย่าง จากผู้ใช้ที่หลงกลและกลุ่มผู้ไม่ประสงค์ดีก็จะสามารถเข้าถึงข้อมูลของผู้ใช้ได้ หรือการปลอมจดหมายของผู้ให้บริการบัตรเครดิต เพื่อให้ลูกค้ากรอกข้อมูลบางอย่าง เพื่อเป็นการยืนยันตนเอง แต่ข้อมูลที่กรอกกลับถูกส่งไปยังกลุ่มผู้ไม่ประสงค์ดี ซึ่งเรื่องเหล่านี้เป็นปัญหาที่นับวันจะสร้างความเสียหายมากขึ้น

6) ข้อความขยะ (Spam)

ข้อความขยะ คือ จดหมายอิเล็กทรอนิกส์ที่ถูกสร้างขึ้นเพื่อโฆษณา การให้บริการบางประเภท หรือบางผลิตภัณฑ์โดยพฤติกรรมคือ จะมีการส่งจดหมายอิเล็กทรอนิกส์ไปยังที่อยู่ต่างๆ ของผู้คนจำนวนมาก ซึ่งเมื่อวิธีการนี้เป็นที่นิยมก็ทำให้ผู้ที่ใช้จดหมายอิเล็กทรอนิกส์ทั่วไปเดือดร้อน โดยผู้ส่งจะไม่สนใจถึงผู้รับ ซึ่งผลเสียหายอาจเกิดขึ้น เช่น กล่องรับข้อความเต็มไปด้วยสแปม ต้องลบสแปมทิ้งทุกวัน หรือข้อความโฆษณาจำนวนมากถูกส่งเข้ามาทุกวัน

7) ดีลเลอร์ (Dialer)

ดีลเลอร์ เป็นอีกลักษณะหนึ่งของโปรแกรม ที่สร้างความเสียหายต่อผู้ใช้ในลักษณะของการถูกใช้งานโทรศัพท์ทางไกลข้ามประเทศโดยไม่รู้ตัว จะรู้ก็ตอนที่ได้รับใบเสร็จเรียกเก็บค่าใช้จ่ายโทรศัพท์ทางไกลระหว่างประเทศมา ดีลเลอร์จะแอบแฝงมากับการซอกแซกท่องอินเทอร์เน็ต โดยดีลเลอร์จะเชิญชวนให้ติดตั้งตัวโปรแกรมเข้าสู่ระบบพร้อมข้อเสนอมากมายในการให้ดาวน์โหลดโปรแกรมคลิปวีดิโอ หรืออื่นๆ เมื่อท่านเปิดเครื่องทิ้งไว้ตามลำพัง Dialer จะหมุนโทรศัพท์ไปยังหมายเลขปลายทางที่ระบุในโปรแกรม

8) สายลับคอมพิวเตอร์ (Spyware)

ในบางครั้ง สปายแวร์อาจถูกอ้างถึงในฐานะของ Spybot (Spy Robot) หรือซอฟต์แวร์ติดตาม สปายแวร์ใช้รูปแบบของซอฟต์แวร์ที่หลอกล่อ และโปรแกรมทำงานเองโดยอัตโนมัติซึ่งไม่ได้รับอนุญาติจากผู้ใช้ การทำงานโดยอัตโนมัติเหล่านี้เช่น การรวบรวมข้อมูลส่วนตัว และการเปลี่ยนการตั้งค่าของบราวเซอร์ หรือโปรแกรมค้นหาซึ่งก่อให้เกิดความรำคาญ และสปายแวร์เองสร้างปัญหาในแง่ของการลดประสิทธิภาพการทำงานของระบบคอมพิวเตอร์ลง หรือการก้าวล่วงสู่ความเป็นส่วนตัวของผู้ใช้เว็บไซต์ที่แจกจ่ายสปายแวร์นั้น ใช้กลอุบายแต่ละชนิดในการหลอกล่อ เพื่อทำให้ผู้ใช้ดาวน์โหลด และติดตั้งมันบนคอมพิวเตอร์ของพวกเขา กลอุบายเหล่านี้อาจเสนอรูปแบบของการให้บริการแลกเปลี่ยนกับการเข้าไปแวะเยี่ยมเยือนเว็บไซต์บางแห่งหรืออื่นๆ

9) แอดแวร์ (Adware)

แอดแวร์ คือ โปรแกรมประยุกต์ที่อาศัยการให้สิทธิในการใช้โปรแกรมฟรี แลกกับการมีพื้นที่โฆษณาในโปรแกรมเหล่านั้น ซึ่งหากผู้ใช้ยอมรับข้อตกลงโปรแกรมก็สามารถติดตั้งและใช้งานได้อย่างถูกต้องตามกฎหมาย และคุณก็จะได้เห็นโฆษณาไปด้วยระหว่างการใช้โปรแกรมเหล่านั้นอย่างไรก็ตาม ประกาศโฆษณาป๊อปอัปน่าจะกลายเป็นความรำคาญ และมีผลต่อการทำงานของระบบ นอกจากนี้ข้อมูลที่โปรแกรมประยุกต์เหล่านี้รวบรวม อาจก่อให้เกิดปัญหาความเป็นส่วนตัวของผู้ใช้ และในกรณีที่ผู้ใช้ไม่สนใจรายละเอียดในข้อความ ข้อตกลงเฉพาะในสัญญาการดำเนินการด้านสิทธิบัตรเพียงพอ (EULA)

10) คุกกี้อินเทอร์เน็ต (Cookies)

คุกกี้อินเทอร์เน็ต คือ แฟ้มเอกสารที่เก็บข้อมูลเว็บไซต์ต่างที่ผู้ใช้เข้าเยี่ยมชม และเก็บอยู่ในคอมพิวเตอร์ของผู้ใช้ และเก็บที่อยู่ที่เว็บไซต์ ข้อมูลของผู้ใช้ และข้อมูลต่างๆ ที่อาจมีการร้องขอหรือบันทึกไว้ขณะท่องอินเทอร์เน็ตคุกกี้ คือ เครื่องมือที่ชอบด้วยกฎหมายที่เว็บไซต์หลายแห่งใช้ในการแลกเปลี่ยนข้อมูล หรือใช้สำหรับเก็บข้อมูลพฤติกรรมการใช้บริการของลูกค้า เช่นผู้ใช้น่าจะเลือกซื้อของในร้านค้าออนไลน์ แต่ครั้งหนึ่งเขาหรือเธอเคยซื้อสินค้าใส่ในรถเข็นสินค้าออนไลน์ของพวกเขา หรือเพื่อการปรับเปลี่ยนหน้าร้าน การจัดหมวดหมู่ของสินค้าตามความสนใจของลูกค้า หรือเพื่อที่จะเก็บสถานะของผู้เข้าเยี่ยมชมเว็บไซต์ นักพัฒนาเว็บไซต์สามารถค้นคืนข้อมูลที่ถูกเก็บในคุกกี้ที่พวกเขาสร้างเท่านั้น และการกระทำเช่นนี้ต้องรับรองความเป็นส่วนตัวผู้ใช้ และป้องกันไม่ให้ผู้อื่นใช้คุกกี้ที่สร้างขึ้นบนเครื่องของผู้ใช้ แต่เป็นที่น่าสังเกตว่าผู้ใช้บางคนไม่รู้ว่าคุกกี้ที่ถูกเก็บอยู่ภายในเครื่องของเขาเองนั้น เป็นช่องทางให้กับนักพัฒนาเว็บไซต์บางกลุ่มใช้ในการติดตามพฤติกรรมการใช้อินเทอร์เน็ตของเรา นอกจากนี้คุกกี้ของเราสามารถค้นคืนข้อมูลและหาหลักฐานร่องรอยการใช้อินเทอร์เน็ตได้ และหากมันถูกนำไปใช้ในทางที่ผิดมันก็อาจสร้างปัญหาต่างๆ ให้กับระบบได้เช่นกัน แต่คุกกี้ก็เป็นสิ่งที่ยอมรับและใช้กันในอินเทอร์เน็ตอย่างแพร่หลาย หากเราพยายามปิดคุกกี้ อาจทำให้เราไม่สามารถหาข้อมูลที่ต้องการจากบางเว็บไซต์ได้เลยซึ่งนั่นหมายความว่าคุกกี้ยังคงมีความจำเป็นอยู่

2.1.3 ประเภทของโปรแกรมที่ไม่หวังดีต่อระบบที่สนใจ

จากโดยโปรแกรมที่ไม่หวังดีต่อระบบทั้ง 10 ประเภท เราได้เลือกตัวที่น่าสนใจได้แก่ 3 ประเภทหลัง คือ สายลับคอมพิวเตอร์หรือสปายแวร์ แอดแวร์ และคุกกี้ เพื่อมาทำการนำมาศึกษาและหาแนวทางที่จะกำจัดโปรแกรมที่ไม่หวังดีต่อระบบดังกล่าว

โดยโปรแกรมที่ไม่หวังดีต่อระบบทั้ง 3 ประเภทที่เราสนใจนี้ มีลักษณะการทำงานที่แตกต่างจากโปรแกรมที่ไม่หวังดีต่อระบบอื่นๆ ที่ไม่ทำการโจมตีผู้ใช้งานโดยตรง แต่กลับทำการแอบแฝงการทำงานเพื่อผลประโยชน์ทางด้านอื่น โดยแอดแวร์จะเป็นการแสดงโฆษณาต่างๆ ขึ้นมารบกวนผู้ใช้งาน ส่วนสปายแวร์และคุกกี้จะมีผลต่อการละเมิดสิทธิความเป็นส่วนตัวของผู้ใช้ โดยที่ผู้ใช้ไม่อาจจะรับรู้ได้เลย นอกจากนี้แอดแวร์และสปายแวร์ถือเป็นภัยคุกคามบนอินเตอร์เน็ตอย่างหนึ่งที่มีอัตราการเพิ่มขึ้นอย่างรวดเร็วในตลอด 2 ปีที่ผ่านมา

2.2 ระบบป้องกันภัยพื้นฐานที่นิยมใช้อยู่ในปัจจุบัน

ในหัวข้อนี้กล่าวถึงระบบป้องกันภัยพื้นฐานที่ปัจจุบันนิยมใช้ อันได้แก่ ไฟร์วอลล์ เพอร์ซันแนลไฟร์วอลล์ และระบบตรวจจับผู้บุกรุก ซึ่งลักษณะการทำงานของระบบป้องกันภัยพื้นฐานเหล่านี้ มีข้อดีข้อเสียที่แตกต่างกัน แต่ไม่ว่าจะมีระบบป้องกันภัยที่ดีเพียงใดก็ยังคงพบว่ามีโอกาสที่จะถูกโปรแกรมที่ไม่หวังดีต่อระบบนั้นสามารถเข้ามาสร้างปัญหาได้ทั้งสิ้น โดยจะขอกล่าวถึงการทำงานของแต่ละประเภทตามลำดับ ดังนี้

2.2.1 ความรู้เบื้องต้นเกี่ยวกับไฟร์วอลล์

2.2.1.1 ความหมายเบื้องต้นของไฟร์วอลล์

ไฟร์วอลล์ คือ เครื่องมือที่ใช้ป้องกันเน็ตเวิร์กจากการสื่อสารทั่วไปที่ไม่ได้รับอนุญาติ โดยปัญหาพื้นฐานที่สุดในเรื่องความปลอดภัยบนเน็ตเวิร์กก็คือ การเข้าถึงระบบหรือข้อมูลภายในผ่านทางเน็ตเวิร์ก หรือที่เรียกว่า ลอจิคัลแอคเซส (Logical Access) ซึ่งมักเกิดขึ้นได้ง่ายกว่าการเข้าถึงทางกายภาพ (Physical Access) คือ เข้ามาที่ตัวเครื่องจริงๆ การที่เรานำโฮสต์ใดๆ มาต่อเข้ากับเน็ตเวิร์ก จะหมายถึงโฮสต์ของเราสามารถถูกแอคเซสได้จากทุกๆ ที่ตลอดเท่าที่เน็ตเวิร์กนั้นจะครอบคลุมไปถึง แต่อย่างไรก็ตามลอจิคัลแอคเซสจะเกิดขึ้นได้นั้นก็ต่อเมื่อโฮสต์จะต้องสามารถสร้างการเชื่อมต่อกับโฮสต์เป้าหมายปลายทางได้ ซึ่งความสามารถในการสร้างลอจิคัลคอนเนคชั่นนั้นจะขึ้นอยู่กับโปรโตคอลที่ใช้งานอยู่เป็นสำคัญ บางโปรโตคอลสามารถสร้างลอจิคัลแอคเซสระหว่างโฮสต์ที่อยู่บนเซกเมนต์เดียวกัน บางโปรโตคอลสามารถสร้างลอจิคัลคอนเนคชั่นให้ข้ามเซกเมนต์ได้ แต่โปรโตคอลที่สำคัญที่สุดที่ต้องดูแลอย่างระมัดระวังก็คือ TCP/IP ซึ่งใช้งานอยู่บนอินเตอร์เน็ตในปัจจุบันเพราะสามารถสร้างลอจิคัลคอนเนคชั่นได้โดยไม่มีขีดจำกัดในเรื่องระยะทาง

2.2.1.2 หลักการทำงานของไฟร์วอลล์

ไฟร์วอลล์ เป็นเครื่องมือรักษาความปลอดภัยที่ทำงานในเชิงป้องกัน ซึ่งจะทำหน้าที่ควบคุมการเข้าถึงเน็ตเวิร์ก โดยอาศัยกฎเป็นพื้นฐาน (Rule base) สำหรับคุณสมบัติแต่ละอย่างของไฟร์วอลล์นั้นมีรายละเอียดดังนี้

1) การทำการป้องกัน (Protect)

ไฟร์วอลล์ เป็นเครื่องมือรักษาความปลอดภัยที่ทำงานในเชิงป้องกัน โดยแพ็คเก็ตที่สามารถผ่านเข้าออกเครือข่ายได้นั้น จะต้องเป็นแพ็คเก็ตที่ไฟร์วอลล์เห็นว่ามีความปลอดภัย แพ็คเก็ตใดที่ไฟร์วอลล์เห็นว่าไม่ปลอดภัยจะทำการควบคุม โดยการตัดสินใจของไฟร์วอลล์ว่าปลอดภัยหรือไม่ปลอดภัยจะขึ้นอยู่กับผู้ดูแลไฟร์วอลล์เป็นผู้กำหนดไว้ล่วงหน้า

2) การควบคุมการเข้าถึง (Access Control)

การควบคุมการเข้าถึง ในแต่ละระดับจะมีวิธีการแตกต่างกันออกไป ทำให้การควบคุมการเข้าถึงสำหรับแต่ละระดับแตกต่างกันตามไปด้วย ไฟร์วอลล์จึงมีการทำงานหลายลักษณะตามวิธีที่ไฟร์วอลล์ใช้ควบคุมการเข้าถึง

3) กฎที่ใช้ควบคุม (Rule Base)

ไฟร์วอลล์จะควบคุมการเข้าถึง โดยอาศัยการเปรียบเทียบคุณสมบัติของแพ็คเก็ตที่จะผ่านไฟร์วอลล์กับกฎของการเข้าถึงที่ได้กำหนดไว้ หากพบว่าไม่มีกฎที่ห้ามไว้ ก็จะอนุญาตให้แพ็คเก็ตนั้นผ่านไป แต่หากพบว่ามีกฎที่ห้ามไว้ ก็จะไม่อนุญาตให้แพ็คเก็ตนั้นผ่านไปได้

2.2.2 ความรู้เบื้องต้นเกี่ยวกับเพอร์ซันแนลไฟร์วอลล์ (Personal Firewall)

2.2.2.1 ความหมายเบื้องต้นของเพอร์ซันแนลไฟล์วอลล์

เพอร์ซันแนลไฟร์วอลล์ คือ ระบบไฟล์วอลล์ที่ใช้สำหรับป้องกันเฉพาะเครื่องคอมพิวเตอร์ส่วนบุคคล โดยเพอร์ซันแนลไฟร์วอลล์มีความแตกต่างกับไฟร์วอลล์ทั่วไปพอสมควร เพราะไฟร์วอลล์ทั่วไปนั้นมีเป้าหมายหลักอยู่ที่ความปลอดภัยของเน็ตเวิร์ก และการควบคุมทราฟฟิคที่ผ่านเข้าออกระหว่างเน็ตเวิร์กภายในกับเน็ตเวิร์กภายนอก ซึ่งทำงานอยู่ระหว่างโปรโตคอลในชั้นทรานสปอร์ตเลเยอร์กับแอพพลิเคชั่นเลเยอร์ แต่สำหรับเพอร์ซันแนลไฟร์วอลล์แล้ว การทำหน้าที่เพียงควบคุมทราฟฟิคเข้าออกนั้นไม่เพียงพอที่จะคุ้มครองป้องกันผู้ใช้ให้ได้อย่างปลอดภัย เพราะภัยของเครื่องคอมพิวเตอร์ส่วนบุคคลนั้นได้ครอบคลุมไปในหลายเรื่อง และส่วนใหญ่ก็มีความจำเป็นต้องเข้าไปยุ่งเกี่ยวกับแอพพลิเคชั่นด้วยเติมนั้นเพอร์ซันแนลไฟร์วอลล์มีรากฐานมาจากไฟร์วอลล์ปกติ เพียงแต่เป็นการนำมารวมเข้าไว้ในเครื่องของผู้ใช้เสียเลย แทนที่จะต้องแยกเป็นไฟร์วอลล์ต่างหากอีกเครื่องหนึ่งซึ่งจะเป็นการสิ้นเปลืองโดยใช่เหตุ แต่ในเมื่อมีภัยหลายประการที่จะส่งผลกระทบต่อผู้ใช้ผลิตจึงได้นำโปรแกรมป้องกันหลายๆ ชนิดมารวมกันเป็นชุดของโปรแกรมรักษาความปลอดภัยสำหรับเครื่องคอมพิวเตอร์ส่วนบุคคล

2.2.2.2 หลักการทำงานของของเพอร์ซันแนลไฟล์วอลล์

จุดประสงค์หลักที่มีการผลิตของเพอร์ซันแนลไฟร์วอลล์ก็เพื่อรองรับผู้ใช้จำนวนมาก ที่ต้องการป้องกันตัวเอง แต่ต้องการลงทุนที่ในราคาไม่สูงจนเกินไป และได้ใช้งานอย่างเต็มที่เต็มความสามารถแทนที่จะต้องซื้อไฟร์วอลล์ธรรมดาแล้วได้ใช้งานเพียงส่วนน้อย และต้องปล่อยส่วนที่เหลือนั้นทิ้งไปโดยเปล่าประโยชน์ และราคาที่สูงมากนั้นก็เป็นถูกใช้ไปเพื่อส่วนเกินที่ไม่ได้นำมาใช้นั่นเอง

โดยการนำไฟร์วอลล์ทั่วไปมาทำหน้าที่เป็นเพอร์ซันแนลไฟร์วอลล์เพื่อให้บริการโฮสต์เพียงโฮสต์เดียว ซึ่งเป็นการใช้งานที่ไม่คุ้มค่าเลย เพราะมีทรัพยากรหลายชนิดที่ไม่ได้ถูกนำมาใช้ให้เกิดประโยชน์อย่างเต็มที่ ได้แก่ เน็ตเวิร์กอะแดปเตอร์บนโฮสต์ ซึ่งมีไว้สำหรับสื่อสารกับไฟร์วอลล์เท่านั้น ไม่ได้ใช้ประโยชน์อย่างอื่น เพราะไม่มีโฮสต์อื่นร่วมอยู่ในเน็ตเวิร์กด้วย ไฟร์วอลล์สามารถรองรับการทำงานได้หลายโฮสต์ แต่นำมาให้บริการกับโฮสต์เพียงโฮสต์เดียว และเราเตอร์ ที่สามารถเชื่อมเน็ตเวิร์กเข้าด้วยกัน แต่ก็ถูกนำมาใช้เพื่อเชื่อมโฮสต์เพียงโฮสต์เดียว จึงมีการพัฒนาเพอร์ซันแนลไฟร์วอลล์ที่ออกแบบมาสำหรับโฮสต์เดียว โดยมีหลักการทำกงานหลักๆ คือ

1) การควบคุมเอาต์บาวนด์แอคเซส

เอาต์บาวนด์แอคเซส คือการส่งข้อมูลจากโฮสต์ตนเองไปยังโฮสต์ผู้อื่น โดยเพอร์ซันแนลไฟร์วอลล์จะสามารถป้องกันทราฟฟิคประเภทนี้ได้อย่างไรนั้น ต้องพิจารณาถึงจุดกำเนิดของเอาต์บาวนด์แอคเซสเสียก่อนว่ามาจากที่ใดหากเป็นการพิจารณาทราฟฟิคบนเน็ตเวิร์กแล้วแน่นอนว่าเราจะพิจารณาจาก IP Address ต้นทางเป็นหลักหาก IP Address ต้นทางมาจากโฮสต์ใดก็ถือว่าโฮสต์นั้นเป็นจุดกำเนิดของทราฟฟิคโดยไม่คำนึงว่าจะมาจากแอพพลิเคชั่นใดบนโฮสต์นั้น แต่สำหรับเพอร์ซันแนลไฟร์วอลล์นั้นอยู่รวมภายในโฮสต์อยู่แล้วย่อมไม่ต้องทำการตรวจสอบจาก IP Address ให้เสียเวลา จุดสำคัญที่เพอร์ซันแนลไฟร์วอลล์สนใจ คือเป็นทราฟฟิคจากแอพพลิเคชั่นใด กล่าวคือ แอคเซสรูลของเพอร์ซันแนลไฟร์วอลล์จะไม่ได้เป็นการกำหนดว่าแพ็คเก็ตจากโฮสต์ใดที่จะสามารถผ่านออกไปได้ แต่จะเป็นการกำหนดว่าแพ็คเก็ตจากแอพพลิเคชั่นใดที่จะได้รับอนุญาตให้ผ่านออกไปบ้าง

2) การควบคุมอินบาวนด์แอคเซส

อินบาวนด์แอคเซส เป็นทราฟฟิคจากภายนอกที่มีปลายทางเข้ามายังเครื่องคอมพิวเตอร์ของผู้ใช้ ที่มาของอินบาวนด์แอคเซสนั้นไม่สามารถควบคุมได้ เพราะอาจมาจากโฮสต์ใดก็ตามที่อยู่ในอินเตอร์เน็ตที่สามารถระบุ IP Address ของผู้ใช้ได้อย่างถูกต้องตามที่กำหนดไว้ในโปรโตคอล TCP/IP เมื่อโฮสต์ใดๆ ก็ตามได้รับการติดต่อมาจากโฮสต์อื่นไม่ว่าในลักษณะใดก็จะต้องตอบสนองต่อการติดต่อนั้นอย่างถูกต้องตามที่กำหนดไว้ในโปรโตคอล แต่อย่างไรก็ตามด้วยลักษณะที่ไม่สามารถกำหนดต้นกำเนิดของอินบาวนด์แอคเซสเหล่านั้นได้ การตอบสนองต่อการติดต่อที่เข้ามานั้นอาจจะส่งผลร้ายต่อผู้ใช้ได้ เพอร์ซันแนลไฟร์วอลล์ได้เข้ามาช่วยผู้ใช้โดยจะทำการป้องกันไม่ให้อินบาวนด์แอคเซสที่ไม่เหมาะสม และอาจจะเป็นอันตรายกับผู้ใช้ไม่ให้เข้ามาได้ เพื่อไม่ให้เกิดความเสียหายอื่นลุกลามออกไป

ดังนั้นสำหรับเครื่องคอมพิวเตอร์ส่วนบุคคลในเบื้องต้น จึงไม่ควรจะอนุญาตอินบาวนด์แอคเซสใดๆ เข้ามาภายในเลยเพอร์ซันแนลไฟร์วอลล์จะทำการปิดกั้นอินบาวนด์แอคเซสทั้งหมดที่พยายามติดต่อเข้ามาโดยดีฟอลต์ ซึ่งจะเป็นการป้องกันไม่ให้แอพพลิเคชั่นใดๆ ก็ตามบนเครื่องคอมพิวเตอร์ของผู้ใช้ที่ทำตัวเป็นเซิร์ฟเวอร์สามารถใช้บริการต่อผู้ใดได้ เป็นการป้องกันอีกชั้นหนึ่ง

2.2.2.3 ความสามารถของไฟร์วอลล์และเพอร์ซันแนลไฟล์วอลล์

ไฟร์วอลล์และเพอร์ซันแนลไฟร์วอลล์ แม้จะมีความสามารถมากมายในการป้องกันระบบ แต่มันก็ยังคงมีส่วนที่ไม่สามารถป้องกันได้เช่นกัน ซึ่งสิ่งที่ไฟร์วอลล์และเพอร์ซันแนลไฟร์วอลล์ป้องกันได้และป้องกันไม่ได้

ด้วยคุณสมบัติที่มีอยู่ของไฟร์วอลล์ ทำให้ระบุได้ชัดเจนว่าภัยประเภทใดบ้างที่ไฟร์วอลล์สามารถป้องกันได้โดยตรง แต่ทั้งนี้จะต้องอยู่บนสมมติฐานว่าไฟร์วอลล์ได้มีการกำหนดกฎต่างๆเอาไว้อย่างถูกต้องด้วยจึงจะได้ผล

สิ่งที่ไฟร์วอลล์สามารถป้องกันได้มีดังนี้

1) การสแกนเน็ตเวิร์ก (Network Scanning)

การสแกนเน็ตเวิร์กเป็นสัญญาณเริ่มต้นของภัยอื่นๆ ที่จะติดตามมา ด้วยคุณสมบัติที่สามารถควบคุมการเข้าออกของแพ็คเก็ตได้ ทำให้ผู้ใช้มีโอกาสที่จะสามารถจำกัดปลายทางของแพ็คเก็ต ที่จะผ่านเข้าเฉพาะโฮสต์ที่อนุญาตให้ติดต่อได้จากภายนอกได้เท่านั้น แพ็คเก็ตที่ส่งเข้ามาเพื่อสำรวจเน็ตเวิร์กโดยการส่งไปยังโฮสต์อื่นๆ ในเน็ตเวิร์กจะไม่สามารถเล็ดลอดไปถึงเป้าหมาย และนำข้อมูลออกไปได้

2) การสแกนโฮสต์ (Host Scanning)

การสแกนโฮสต์ก็เป็นองค์ประกอบเริ่มต้นที่สำคัญ ของการเตรียมการของการเจาะระบบ เพราะถึงแม้จะมีไฟร์วอลล์ติดตั้งอยู่ด้วยก็ตาม มิได้หมายความเน็ตเวิร์กที่อยู่หลังไฟร์วอลล์จะถูกตัดขาดจากโลกภายนอก จะต้องมีโฮสต์อย่างน้อยหนึ่งตัวที่ต้องติดต่อกับโลกภายนอกได้ ซึ่งโฮสต์นั้นก็มีโอกาสที่จะถูกสแกนได้ การที่แฮคเกอร์สามารถสแกนโฮสต์ได้นั้นจะทำให้มีโอกาสค้นหาข้อบกพร่องต่างๆ ของโฮสต์ และนำไปเป็นข้อมูลเพื่อการเจาะเข้าไปยังโฮสต์ในภายหลังได้

3) อินบาวนด์แอคเซส (Inbound Access)

ทุกคนที่ต่อกับอินเตอร์เน็ตสามารถจะส่งข้อมูลออกไป (Outbound) ที่ใดก็ได้โดยไม่มีการควบคุม ไม่ว่าจะเป็นการต่อแบบตลอดเวลาโดยใช้ลีสไลน์ (Leased line) หรือต่อเป็นครั้งคราวโดยใช้โมเด็มต่อเข้าไปยัง ISP ทุกคนจะมีโอกาสเท่าเทียมกันในการส่งข้อมูลไปยังทุกๆที่ในโลก และไม่ว่าจะยินยอมหรือไม่ก็มีโอกาสที่จะได้รับข้อมูลเข้ามา (Inbound) ได้ทุกรูปแบบ และจากทุกๆที่ในโลก เช่น เครื่องมือของแฮคเกอร์ ไวรัส โปรแกรมม้าโทรจัน ก็จะมาถึงโฮสต์ผู้ใช้ได้ตราบใดที่วิธีการส่งยังเป็นวิธีที่โปรโตคอล TCP/IP การนำโฮสต์มาต่ออินเตอร์เน็ตจึงอยู่ในสภาพที่ทุกคนจะต้องป้องกันตัวกันเอาเอง

4) เอาต์บาวนด์แอคเซส (Outbound Access)

นอกจากการป้องกันการเข้ามาของข้อมูลภายนอกแล้ว ไฟร์วอลล์ยังสามารถป้องกันข้อมูลภายในไม่ให้ออกไปภายนอกได้ การลักลอบส่งข้มูล ในบางกรณีอาจจะไม่ได้เกิดจากการทำของพนักงานเอง แต่อาจจะเกิดจากคนภายนอกที่มีโอกาสเข้าถึงข้อมูลได้ ซึ่งถ้าไม่มีช่องทางในการส่งข้อมูลออกไปก็อาจจะไม่สามารถขโมยความลับไปได้ แต่ถ้ายังต่อกับเน็ตเวิร์กอยู่ ก็จะทำให้สามารถส่งข้อมูลออกไปได้ ซึ่งการใช้งานไฟร์วอลล์เป็นการป้องกันในส่วนนี้

5) การก่อกวนไม่ให้โฮสต์สามารถให้บริการได้ (Network Denial of Service)

ไฟร์วอลล์สามารถป้องกันการโจมตีโดยการก่อกวนไม่ให้โฮสต์สามารถให้บริการได้โดยใช้เทคนิคในระดับของเน็ตเวิร์กด้วยวิธีการต่างๆ ไม่ว่าจะเป็นการส่งอะนอมาลัสแพ็คเก็ต การทำให้เน็ตเวิร์กท่วมไปด้วยข้อมูล การส่งแพ็คเก็ตจำนวนมากไปยังโฮสต์เพื่อขอใช้บริการเพื่อรบกวนโฮสต์

สิ่งที่ไฟร์วอลล์ไม่สามารถป้องกันได้มี ดังนี้

1) แฮกกอร์ (Hacker)

คนหรือกลุ่มคนที่พยายามเจาะเข้าระบบคอมพิวเตอร์ต่างๆ โดยอาศัยทักษะทางคอมพิวเตอร์และข้อบกพร่องของระบบเป้าหมายที่ทำงานอยู่ โดยวัตถุประสงค์หลักของแฮคเกอร์ก็คือ ผ่านระบบรักษาความปลอดภัยเข้าไปให้ได้ไม่ว่าจะด้วยวิธีใดก็ตาม

2) การบริการที่ได้รับอนุญาต (Allowed Services)

ไฟร์วอลล์จะควบคุมการสื่อสารข้อมูลโดยใช้กฎเป็นสำคัญ ไม่มีกฎที่ผิด และกฎที่ถูกแน่นอนตายตัว กฎที่ผิดสำหรับเน็ตเวิร์กหนึ่งอาจถูกสำหรับอีกเน็ตเวิร์กหนึ่งได้ เพราะปัจจัยที่สำคัญในการกำหนดกฎก็คือ นโยบายความปลอดภัย และลักษณะการบริหารที่อยู่ในเน็ตเวิร์กนั้นการพิจารณากฎที่มีอยู่ในไฟร์วอลล์จึงมีเพียงความเหมาะสมของกฎเท่านั้น

3) โปรแกรมที่ไม่หวังดี (Malware)

ในความเป็นจริงไฟร์วอลล์นั้นสามารถป้องกันภัยประเภทนี้ได้บางส่วน โดยสามารถป้องกันไม่ให้โปรแกรมเหล่านั้นสามารถทำงานได้อย่างสมบูรณ์ แต่อย่างไรก็ตามไฟร์วอลล์นั้นไม่สามารถที่จะตรวจจับหรือเตือนผู้ใช้ไม่ให้รันโปรแกรมประเภทนี้ได้

2.2.3 ความรู้เบื้องต้นเกี่ยวกับระบบตรวจจับผู้บุกรุก (IDS: Instrusion Detection System)

2.2.3.1 ความหมายเบื้องต้นของระบบตรวจจับผู้บุกรุก

เมื่อพูดถึงระบบไฟร์วอลล์ (Firewall System) เกือบทุกคนที่อยู่ในวงการคอมพิวเตอร์รู้ถึงประโยชน์ และหน้าที่ของระบบนี้ ถ้าเปรียบเทียบระบบไฟร์วอลล์เหมือนกับยามที่เฝ้าระบบเครือข่าย เราก็สามารถเห็นข้อจำกัดต่างๆ ที่มีอยู่ในระบบไฟร์วอลล์ เช่น ระบบไฟร์วอลล์ไม่สามารถตรวจจับพฤติกรรมของบุคคล หรือระบบที่อยู่ภายในเครือข่ายได้ เพื่อแก้ปัญหานี้ เราจึงจำเป็นต้องมีอุปกรณ์หรือเครื่องมือที่ใช้ในการตรวจจับพฤติกรรมต่างๆ ที่มีอยู่ในเครือข่าย ถ้าเปรียบกับอาคารสำนักงาน อุปกรณ์เช่นนี้ก็ได้แก่ กล้องวีดีโอที่ติดตั้งตามจุดต่างๆ ภายในสำนักงานเพื่อบันทึกพฤติกรรมต่างๆ ที่เกิดขึ้นภายในสำนักงานนั้น หลังจากนั้นก็จะมีเจ้าหน้าที่มาทำการวิเคราะห์หาพฤติกรรมที่น่าสงสัย พร้อมกับจัดการกับเหตุการณ์นั้น ถ้าเป็นเครือข่ายคอมพิวเตอร์ ระบบที่ใช้จัดการกับปัญหาเช่นนี้ได้แก่ ระบบตรวจจับผู้บุกรุก (IDS) นั่นเอง

2.2.3.2 หลักการของระบบตรวจจับผู้บุกรุก

ระบบตรวจจับผู้บุกรุกนั้นจะทำการตรวจจับการบุกรุก และวิเคราะห์ข้อมูลที่อยู่บนเครือข่ายทั้งภายใน และภายนอกว่ามีพฤติกรรมที่เป็นความเสี่ยง และก่อความเสียหายต่อระบบงานภายในองค์กรหรือไม่ โดยระบบระบบตรวจจับผู้บุกรุกนั้นจะมีระบบแจ้งเตือนให้กับผู้ดูแลระบบทราบ และหยุดพฤติกรรมดังกล่าว

โดยมีสองวิธีการหลักที่ระบบตรวจจับผู้บุกรุก ใช้ในการตรวจจับการบุกรุก หรือความพยายามในการบุกรุก ได้แก่ การวิเคราะห์หลังเกิดเหตุการณ์ขึ้นแล้ว (Post-Event Analysis) และการวิเคราะห์ ณ เวลาจริง (Real-Time Analysis) โดยระบบตรวจจับผู้บุกรุกส่วนใหญ่ในปัจจุบันจะบันทึกเหตุการณ์อย่างเช่น เน็ตเวิร์กทราฟฟิกหรือการล็อกอินเข้าสู่โฮสต์เพื่อนำไปวิเคราะห์ในภายหลัง และมีอยู่ไม่มากที่จะวิเคราะห์เหตุการณ์ ณ เวลาจริง

1) **การวิเคราะห์ ณ เวลาจริง** (Real-Time Analysis)

การวิเคราะห์ประเภทนี้จะทำในสิ่งเดียวกันกับ Post-Event Analysis เพียงแต่ทำได้เร็วกว่า มีข้อจำกัดบางอย่างในแง่ที่ว่าการรวบรวมข้อมูล ณ เวลาจริงอาจเป็นไปได้ไม่ง่ายนัก นี่ไม่ได้หมายความว่ามันจะเกิดขึ้นไม่ได้ทางเทคนิค เราเพียงหมายความว่า ระบบ IDS แบบ ณ เวลาจริงสามารถตรวจจับได้เฉพาะในสิ่งที่มันสามารถมองเห็นได้ทัน

2) **การวิเคราะห์หลังเกิดเหตุการณ์ขึ้นแล้ว** (Post-Event Analysis)

ระบบตรวจจับผู้บุกรุกแบบนี้ ได้รับการออกแบบให้มีความกว้างขวางของสโคปการทำงาน โดยปกติแล้วพวกมันจะประกอบไปด้วยการทำออดิตล็อก หรือการทำการรวบรวมข้อมูลมาไว้ที่เครื่องศูนย์กลางที่จะทำหน้าที่วิเคราะห์ข้อมูลที่ได้รับเพื่อตรวจหาการบุกรุก โดยปกติ มันจะรับอินพุตมาจากระบบต่างๆ ที่สำคัญในเน็ตเวิร์กขององค์กรโดยใช้บางสิ่ง อย่างเช่น Syslog, EventLog, SNMP Traps หรือโปรโตคอลเฉพาะตัวอื่นๆ

2.2.3.3 ประเภทของการตรวจจับในระบบตรวจจับผู้บุกรุก

ปัจจุบันมีสองโมเดลหลักสำหรับการตรวจหาผู้บุกรุก ได้แก่ การตรวจจับพฤติกรรมที่ผิดแปลก และการตรวจจับการใช้งานในทางที่ไม่เหมาะสม โมเดลเหล่านี้มีรูปแบบการตรวจจับการบุกรุกที่แตกต่างกันออกไป แต่ผลลัพธ์ที่ได้ซึ่งก็คือ การตรวจหาผู้บุกรุกนั้นจะเหมือนกัน มันยังมีระบบไฮบริดที่ผนวกรวมเอาทั้งสองฟีเจอร์เข้าไว้ด้วยกัน

1) การตรวจจับโดยดูจากพฤติกรรม (Behavior-Based)

การตรวจจับโดยดูจากพฤติกรรมที่แปลกประหลาด โดยจะใช้วิธีการตรวจหาพฤติกรรมที่เบี่ยงเบนไป จากแพตเทิร์นการทำงานปกติ นี่มีพื้นฐานมาจากสมมุติฐานที่ว่ามีทราฟฟิกแพตเทิร์นที่ "ปกติ" สำหรับบริษัทของคุณ และพวกมันสามารถถูกแยกแยะได้ ทราฟฟิกที่ "ปกติ" เหล่านี้จะถูกเรียกว่า บรรทัดฐานปกติ(Baseline) จากนั้น เมื่อมีพฤติกรรมที่เบี่ยงเบนไปจาก "Baseline" โดย IDS จะประเมินมัน และทำการตัดสินใจต่อไปว่าจะทำอย่างไร สมมุติฐานพื้นฐานก็คือ การโจมตีระบบคอมพิวเตอร์สามารถถูกตรวจจับได้จากพฤติกรรมของระบบที่ไม่ปกติ

2) การตรวจจับโดยดูจากการใช้งานที่ไม่ถูกต้อง (Knowledge-Based)

การตรวจจับจากการใช้งานอย่างไม่ถูกต้อง จะชี้ให้เห็นถึงการบุกรุกโดยพิจารณา "การรู้จำแพตเทิร์นของการโจมตี (pattern recognition)" โดยหลักการนี้มีพื้นฐานจากข้อเท็จจริงที่ว่า แพตเทิร์นของการโจมตีที่เป็นที่รู้จักกันดีสามารถอธิบายได้ถึงการฉวยโอกาสโจมตีทางช่องโหว่ที่มีอยู่ นี่คือ ลักษณะการทำงานของระบบการตรวจจับในปัจจุบัน

แง่มุมในด้านบวกของระบบตรวจจับผู้บุกรุกประเภทนี้ก็คือ มันมีความน่าเชื่อถือได้เป็นอย่างมาก กล่าวคือ มันจะแจ้งเตือนให้คุณทราบถึงสิ่งที่คุณต้องการให้มันแจ้ง ถ้าคุณต้องการทราบว่าเมื่อใดที่มีใครบางคนที่พยายามเข้ามายังระบบของคุณด้วยรหัสผ่านว่างเปล่า (blank password) คุณสามารถเซตให้การใช้งานแบบนี้เป็น "แพตเทิร์นหรือสัญลักษณ์บ่งบอก" และมันจะแจ้งเตือนให้คุณทราบเมื่อพบแพตเทิร์นดังกล่าวนี้ ในลักษณะนี้มันสามารถบอกคุณได้เฉพาะในสิ่งที่คุณต้องการรู้

2.2.3.4 ความสามารถของระบบตรวจจับผู้บุกรุก (IDS)

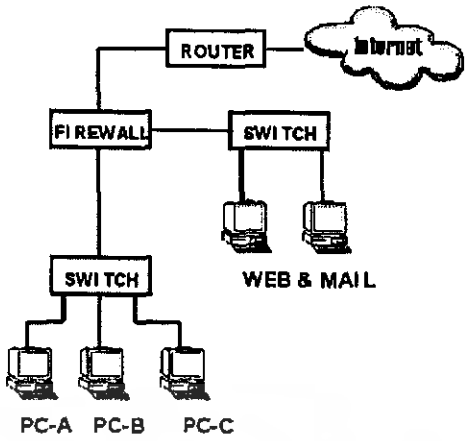
ถ้าหน่วยงานใดมีการติดตั้งเครือข่ายภายใน (LAN) หรือเครือข่ายอินเทอร์เน็ต เพื่อใช้งานในองค์กร ถึงแม้ว่าระบบจะมีการติดตั้งไฟล์วอลล์ เพื่อป้องกันการบุกรุกจากบุคคลภายนอกแล้วก็ตาม ระบบก็ยังมีช่องโหว่ที่ทำให้เกิดความเสี่ยงต่อการเสียหายได้ดังต่อไปนี้

ความเสี่ยงเนื่องจาก การเปิดบริการใช้งานบางประเภทที่ไฟร์วอลล์ ให้กับบุคคลภายนอกที่มาจากอินเทอร์เน็ต เช่น การบริการเวปแอพพลิเคชั่น การบริการรับส่งเมล์และการบริการถ่ายโอนข้อมูล (FTP) เป็นต้น เนื่องจาก ผู้ไม่ประสงค์ดี (Hacker) จะนิยมเข้ามาจู่โจมและก่อให้เกิดความเสียหายแก่

ระบบงานภายในองค์กร โดยเข้าทางที่ไฟล์วอลล์อนุญาติเป็นส่วนมาก ตัวอย่างของความเสี่ยงที่พบเห็นบ่อยๆ คือ

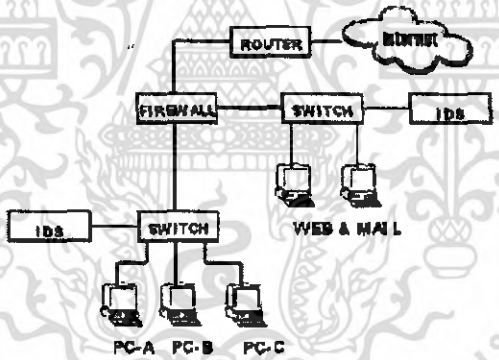
- **SMTP Overflow** เป็นความเสี่ยงที่จะก่อปัญหาให้กับระบบเมลล์เซิร์ฟเวอร์ทำงานช้าลง เพราะผู้ไม่ประสงค์ดีจะพยายามส่งเมล์จากภายนอกที่มีขนาดใหญ่มากๆ เข้ามาที่เมลล์เซิร์ฟเวอร์ภายในองค์กร
- **Web Attacks** เป็นความเสี่ยงที่เกิดจากการเข้ามาขโมยข้อมูล รหัสที่เวปเซอร์เวอร์โดยใช้ช่องทางพอร์ต 80 มาติดตั้งระบบ "Hand-Crafted URL" หลังจากนั้นก็จะดึงข้อมูลจากไฟล์ "/etc/shadow" ที่เก็บรหัสทั้งหมดไว้ไปใช้งานได้
- **ความเสี่ยงจากบุคคลภายใน** ซึ่งเป็นความเสี่ยงที่เป็นอันตรายมากกว่าบุคคลภายนอก เพราะว่าเครือข่ายภายในจะไม่มีระบบรักษาความปลอดภัยมาป้องกันเลย ดังนั้นทำให้คนภายในด้วยกันสามารถที่จะทำการแฮ็กกันได้ง่ายกว่าบุคคลภายนอก
- **Backdoor** เป็นความเสี่ยงที่เกิดจากการแอบนำเอาซอฟต์แวร์ไปทำการติดตั้งที่เครื่องที่ต้องการขโมยข้อมูล เมื่อติดตั้งเสร็จก็สามารถที่จะดึงข้อมูลที่อยู่บนเครื่องนั้นๆ ได้ และยังเฝ้าดูการใช้งานของเจ้าของเครื่องที่ถูกติดตั้งซอฟต์แวร์ดังกล่าวได้ เช่น การคีย์รหัสสำหรับแอพพลิเคชั่นที่เป็นความลับได้
- **ความพยายามเข้าไปใช้งานในระบบงานที่ไม่ได้อนุญาติไว้** เช่น ความพยายาม Telnet, FTP หรือพยายามติดต่อกับฐานข้อมูลของบริษัท ถ้าเราปล่อยให้บุคคลกลุ่มนี้ลองสุ่มใส่ รหัสไปเรื่อย หรือขโมยรหัสไปใช้ ก็จะเป็นอันตรายต่อระบบงานอย่างมาก
- **มีการแพร่กระจายไวรัสโดยไม่ตั้งใจ** เช่น การนำเอาอุปกรณ์คอมพิวเตอร์เข้ามาทดสอบหรือนำเอาเครื่องคอมพิวเตอร์จากภายนอกมาใช้ในองค์กร ซึ่งอาจก่อให้เกิดความเสียหายแก่ระบบงานได้

สรุปได้ว่าความเสี่ยงที่เกิดขึ้นดังที่กล่าวมานั้น มีทั้งความเสี่ยงที่ไม่ได้อยู่ในการบริการของไฟล์วอลล์ หรืออยู่ในการบริการของไฟล์วอลล์ ซึ่งล้วนแต่ก่อให้เกิดความสูญเสียแก่องค์กรไม่มากก็น้อย ขึ้นอยู่กับองค์ประกอบของแต่ละองค์กร แต่เราก็ควรที่จะป้องกันไม่ให้เกิดเหตุการณ์ดังกล่าวขึ้น ดังนั้นแนวทางในการแก้ไขปัญหาดังที่กล่าวมาก็คือ การนำระบบ Intrusion Detection System (IDS) มาใช้งานในองค์กร



รูปที่ 2.1 ระบบเครือข่ายทั่วไปที่มีไฟร์วอลล์

เมื่อมีการติดตั้งระบบ IDS ในองค์กรจะมีผลให้ระบบงานมีความปลอดภัยจากการบุกรุกทั้งบุคคลภายใน และจากอินเทอร์เน็ตได้มากขึ้น และมีประสิทธิภาพ ตลอดจนทำให้เราสามารถรู้ต้นสายปลายเหตุว่า อะไรทำให้ระบบงานมีประสิทธิภาพด้อยลง และเราควรที่จะปรับปรุงระบบงานของเราให้มีความปลอดภัยสูงขึ้นได้อย่างไร ทั้งหมดสามารถที่จะรู้ และทำได้โดยใช้ความช่วยเหลือของระบบ IDS



รูปที่ 2.2 ระบบเครือข่ายที่มีการติดตั้งระบบตรวจจับผู้บุกรุก

2.3 เทคนิคในการตรวจจับสิ่งแปลกปลอม

เทคนิคในการตรวจจับสิ่งแปลกปลอมได้ถูกพัฒนาขึ้น เพื่อตรวจจับไวรัสแตกต่างกันออกไป ซึ่งเทคนิคในการตรวจจับสิ่งแปลกปลอมโดยทั่วไป แบ่งได้ 4 เทคนิค คือ

2.3.1 การตรวจจับสิ่งไม่พึงประสงค์โดยใช้การตรวจหา (Scanning)

เป็นเทคนิคที่ใช้ตัวตรวจหา (Scanner) เข้าไปค้นหาไฟล์ที่ถูกบ่งบอกว่าถูกสิ่งไม่พึงประสงค์แฝงตัวอยู่ในหน่วยความจำ ส่วนเริ่มต้นในการบูต (Boot sector) และไฟล์ที่ถูกเก็บอยู่ในฮาร์ดดิสก์ โดยใช้หลักการ Checksum ซึ่งมีวิธีการทำงานคือ ในไฟล์ทุกไฟล์จะมีส่วนที่เก็บข้อมูลว่ามีจุดเริ่มต้นจุดสิ้นสุดของไฟล์ที่ตำแหน่งใด ตามด้วยข้อมูลของไฟล์ และปิดท้ายด้วยค่า Checksum ตัวตรวจหาจะคำนวณหาค่า Checksum ของแต่ละไฟล์ แล้วนำไปทำการเปรียบเทียบกับค่า Checksum ของไฟล์นั้นๆ ดังนั้นถ้าไฟล์ใดถูกสิ่งแปลกปลอมแฝงตัวก็จะทำให้ค่า Checksum ที่คำนวณได้จะไม่เท่ากับค่า Checksum ที่เป็นข้อมูลของไฟล์ดังกล่าว โปรแกรมป้องกันสิ่งแปลกปลอมทั่วๆ ไป จะมีวิธีการตรวจหา 2 ชนิด คือ

1) **การตรวจหาชนิดก่อนที่จะถูกโหลดเข้าหน่วยความจำ (On-Access Scanning)**

เป็นวิธีการตรวจหาไฟล์ก่อนที่จะถูกโหลดเข้าหน่วยความจำ เพื่อทำการเอ็กซิคิวต์

2) **การตรวจหาชนิด (On-Demand Scanning)**

เป็นวิธีการตรวจหาในหน่วยความจำหลัก ส่วนเริ่มต้นในการบูต และฮาร์ดดิสก์ผู้ใช้งานยังสามารถเรียกใช้งานวิธีการตรวจหาชนิดนี้ตามความต้องการได้

ข้อดีของเทคนิคนี้ก็คือ ตัวตรวจหาสามารถพบสิ่งแปลกปลอมก่อนที่จะทำการเอ็กซิคิวต์

2.3.2 การตรวจจับสิ่งไม่พึงประสงค์โดยใช้การตรวจสอบความคงอยู่ (Integrity Checking)

เทคนิคนี้อาศัยตัวตรวจสอบความคงอยู่ (Integrity Checker) ที่เก็บข้อมูลความคงอยู่ (Integrity Information) ของไฟล์สำคัญไว้สำหรับเปรียบเทียบ ตัวอย่างข้อมูล เช่น ขนาดไฟล์ เวลาแก้ไขครั้งล่าสุด และค่า Checksum เป็นต้น ส่วนมากจะใช้ค่าของ Checksum ในการเปรียบเทียบ เมื่อมีไฟล์เปลี่ยนแปลงที่มีสาเหตุอันเนื่องจากสิ่งแปลกปลอม หรือความผิดพลาดใดๆ จนทำให้ข้อมูลความคงอยู่ต่างจากข้อมูลเดิมที่เคยเก็บไว้ ระบบก็จะแจ้งให้ผู้ใช้งานทราบถึงความผิดปกติ และยังสามารถมีทางเลือกให้ผู้ใช้สามารถกู้ไฟล์ข้อมูลดังกล่าว คืนไปเป็นไฟล์ก่อนที่จะติดสิ่งแปลกปลอมได้

ข้อดีของเทคนิคนี้ คือ เป็นเทคนิคเดียวที่จะตรวจสอบว่ามีสิ่งแปลกปลอมทำลายไฟล์หรือไม่ และเกิดความผิดพลาดน้อย ตัวตรวจสอบความคงอยู่ในปัจจุบันมีความสามารถที่จะตรวจจับการทำลายข้อมูลชนิดต่างๆ ได้ เช่น ไฟล์ไม่สมบูรณ์ (Corruption) และยังสามารถกู้ไฟล์คืนได้

2.3.3 การตรวจจับสิ่งไม่พึงประสงค์โดยใช้การวิเคราะห์พฤติกรรม (Heuristic)

เป็นเทคนิคทั่วไปที่นิยมใช้ในการตรวจจับสิ่งแปลกปลอม โดยจะเปรียบเทียบการทำงานของสิ่งแปลกปลอมกับกฎ Heuristic (Rules Based System) ซึ่งคำว่า Heuristic เป็นคำที่มาจากภาษากรีซมาจากคำว่า Heuristic ซึ่งหมายความว่า "การค้นพบ" และชุดกฎ Heuristic ถูกพัฒนาให้สามารถแยกแยะพฤติกรรมการทำงานว่าเป็นการทำงานของสิ่งแปลกปลอมหรือไม่ มีการเก็บข้อมูลของสิ่งแปลกปลอมที่รู้จักเพื่อใช้ในการจับคู่แพตเทิร์น และชุดกฎนี้ถูกพัฒนาโดยผู้พัฒนาโปรแกรมป้องกันไวรัส

ยกตัวอย่างวิธีการตรวจจับไวรัสชนิดนี้ เช่น โปรแกรมป้องกันไวรัสรู้จักพฤติกรรมการทำงานของไวรัสทั่วไป (เช่น การอ่าน/เขียนลงใน Master Boot Record ซึ่งโปรแกรมทั่วๆ ไปจะไม่ทำเช่นนี้) เมื่อโปรแกรมป้องกันไวรัสตรวจพบว่ามีการทำงานที่ผิดปกติขึ้นในเครื่อง โปรแกรมป้องกันไวรัสจะใช้กฎ Heuristic เปรียบเทียบกับลักษณะดังกล่าว เพื่อที่จะระบุว่าเป็นพฤติกรรมการทำงานของไวรัสชนิดใด

ข้อดีของเทคนิคนี้ ก็คือ มีความยืดหยุ่นในการตรวจจับ และสามารถรู้จักสิ่งแปลกปลอมชนิดใหม่ๆ ได้เอง

2.3.4 การตรวจจับสิ่งไม่พึงประสงค์โดยการดักจับ (Interception)

เทคนิคนี้จะเริ่มต้นด้วยการที่โปรแกรมป้องกันสิ่งไม่พึงประสงค์จะสร้าง Virtual Machine ที่มีความอ่อนแอมากไว้ภายในเครื่อง คอยล่อให้โปรแกรมประเภทสิ่งไม่พึงประสงค์โจมตี และยังมีหน้าที่เฝ้าดูว่ามีโปรแกรมใดบ้างที่มีพฤติกรรมผิดปกติน่าสงสัยเข้ามาทำงานใน Virtual Machine ตัวอย่าง เช่น มีโปรแกรมที่ทำการติดตั้งตัวเอง รวมทั้งมีการส่ง Request ผิดปกติออกมาเพื่อทำให้เครื่องทำงานผิดพลาด เป็นต้น โปรแกรมที่ผิดปกติ หรือน่าสงสัยนี้อาจจะเป็นสิ่งไม่พึงประสงค์ก็ได้

ข้อดีของการใช้เทคนิคนี้ คือ จะหยุดการทำงานของสิ่งไม่พึงประสงค์ที่พยายามที่จะฝังตัวในหน่วยความจำได้ดี

2.4 เทคนิคทางด้านเครือข่ายและโปรโตคอล

2.4.1 ความหมายเบื้องต้นของโปรโตคอล

โปรโตคอล คือ เป็นตัวที่มีหน้าที่สำหรับคอยตกลงระเบียบวิธี ที่กำหนดขึ้นสำหรับสื่อสารข้อมูลโดยสามารถส่งผ่านข้อมูลไปยังปลายทางได้อย่างถูกต้อง เมื่อคอมพิวเตอร์เครื่องหนึ่งต้องการรับส่งข้อมูลกับคอมพิวเตอร์อีกเครื่องหนึ่งที่มีระบบแตกต่างกัน หรือคนละผู้ผลิตเป็นส่งที่ทำให้การติดต่อทำได้ยากมาก จึงต้องมีตัวที่เป็นมาตรฐานส่วนกลาง ที่จำเป็นต้องใช้ในการรับส่งข้อมูล

2.4.2 ความรู้เกี่ยวกับโมเดลเครือข่ายแบบโปรโตคอล TCP/IP

โปรโตคอล TCP/IP เป็นโปรโตคอลที่ใช้กันแพร่หลายที่สุด โดยเฉพาะเมื่อถูกนำไปใช้กับเครือข่ายบนอินเตอร์เน็ต TCP/IP มีการแบ่งโปรโตคอลสื่อสารออกเป็นชั้นๆ โดยจะมีการเรียกลำดับชั้นของ TCP/IP ว่า TCP/IP Stack โดย TCP/IP Stack มีทั้งหมด 4 ชั้น คือ

2.4.2.1 ลำดับชั้นของโมเดลเครือข่ายแบบโปรโตคอล TCP/IP

1) ลำดับชั้นที่ 4: Process Layer

จะเป็น Application Protocol ที่ทำหน้าที่เชื่อมต่อกับผู้ใช้ และให้บริการต่างๆ เช่น FTP, Telnet, SNMP เป็นต้น

2) ลำดับชั้นที่ 3: Host to Host Layer

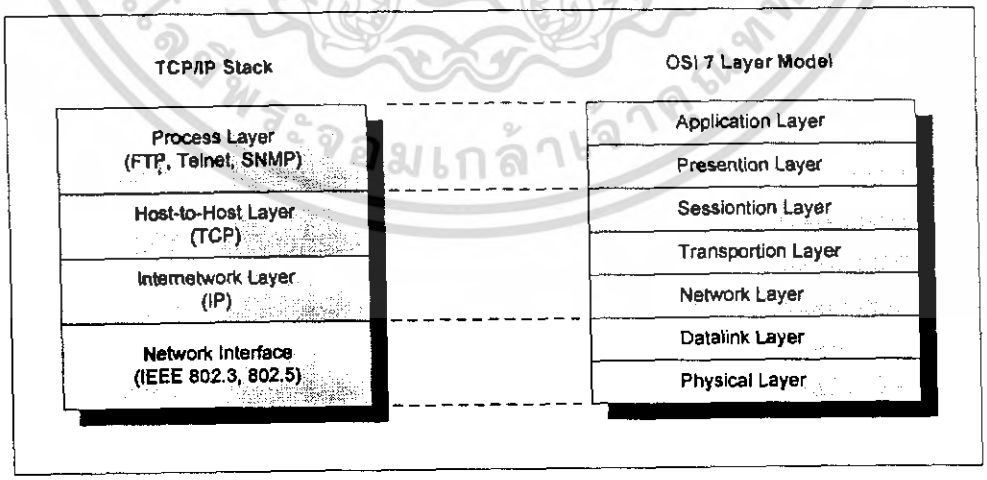
จะเป็น TCP หรือ UDP ที่ทำหน้าที่คล้ายกับ Layer4 ของ OSI Model ควบคุมการรับส่งข้อมูลจากปลายด้านส่งถึงปลายด้านรับข้อมูล

3) ลำดับชั้นที่ 2: Internetwork

ได้แก่ ส่วนของโปรโตคอล IP ซึ่งทำหน้าที่คล้ายกับ Layer3 ของ OSI Model เชื่อมต่อคอมพิวเตอร์ของด้านรับ และด้านส่งเข้าหากันผ่านระบบเครือข่ายพร้อมทั้งเลือก หรือกำหนดเส้นทางที่จะใช้ในการรับส่งข้อมูลระหว่างกัน และส่งผ่านข้อมูลที่ได้รับไปยังอุปกรณ์ในเครือข่ายต่างๆ จนกระทั่งถึงปลายทาง ข้อมูลที่รับส่งกันจะอยู่ในรูปแพ็คเก็ต หรือเฟรมข้อมูล

4) ลำดับชั้นที่ 1: Network Interface

ทำหน้าที่คล้ายกับ Layer1 ,Layer2 ของ OSI Model คือ เชื่อมต่อการรับส่งข้อมูลในระดับฮาร์ดแวร์ โดยทำหน้าที่แปลคำสั่งนั้นๆ ให้เป็นคำสั่งควบคุมฮาร์ดแวร์ และแก้ไขข้อผิดพลาดที่ตรวจพบนั้น ซึ่งที่ใช้กันอยู่จะเป็นตามมาตรฐาน IEEE ข้อมูลในชั้นนี้จะอยู่ในรูปเฟรม



รูปที่ 2.3 ภาพแสดงโมเดลเครือข่ายแบบโปรโตคอล TCP/IP

2.4.2.2 หลักการทำงานของโมเดลเครือข่ายแบบโปรโตคอล TCP/IP

แนวคิดหลักของระบบเครือข่ายคอมพิวเตอร์ ก็คือ การเชื่อมโยงอุปกรณ์เข้าด้วยกัน ไม่ว่าจะเป็นเครื่องเซิร์ฟเวอร์ และอุปกรณ์ในเครือข่ายอื่นๆ จึงจะต้องมีการ IP Address ที่เป็นค่าที่อยู่ไว้อ้างอิงถึงที่ที่จะส่งข้อมูลผ่าน

1) ไอพีแอดเดรส (IP Address)

ไอพีแอดเดรสถูกกำหนดขึ้นมาให้เป็นหมายเลขอ้างอิงประจำตัวของอุปกรณ์ต่างๆ ที่เชื่อมต่ออยู่ในเครือข่ายอินเตอร์เน็ต โดยการกำหนดไอพีแอดเดรสให้แต่ละเครื่อง หรือแต่ละอุปกรณ์นี้จะต้องไม่ซ้ำกัน ซึ่งไอพีแอดเดรสนี้จะไม่ถูกผูกติดกับตัวฮาร์ดแวร์ จึงสามารถกำหนดใหม่ หรือแก้ไขเปลี่ยนแปลงได้เมื่อมีการเปลี่ยนแปลงตัวฮาร์ดแวร์

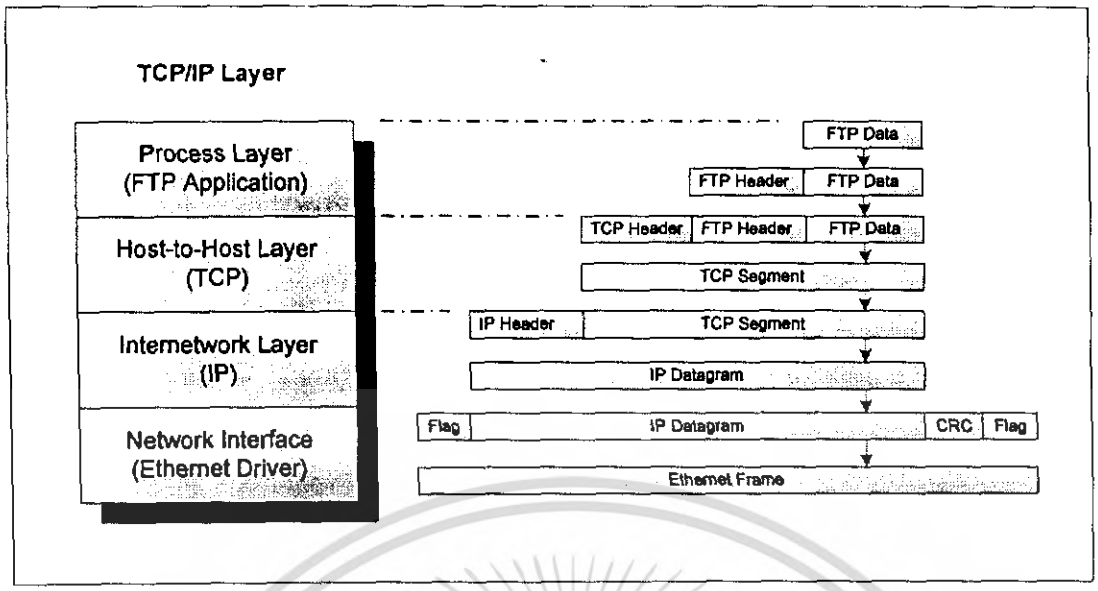
โปรโตคอล IP จำเป็นต้องอาศัยไอพีแอดเดรสเพื่อระบุถึงอุปกรณ์ต่างๆ ที่อยู่ในเครือข่ายไม่ว่าจะเป็นเว็บเซิร์ฟเวอร์ เมล์เซิร์ฟเวอร์ อุปกรณ์เราเตอร์ ไอพีแอดเดรสจะเป็นค่าตัวเลขขนาด 32 bit ถูกแบ่งออกเป็น 4 ส่วน ส่วนละ 8 bit และถูกคั่นแต่ละส่วนด้วยเครื่องหมายจุด

2) แพคเก็ตข้อมูล (Data Packet)

เมื่อมีการส่งข้อมูลบนเครือข่ายอินเตอร์เน็ตนั้น ที่แสดงออกมาทางบราวเซอร์ ข้อมูลจำเป็นต้องมีการทำให้มีขนาดเล็กลง โดยแบ่งออกเป็นย่อยๆ เรียกว่า Data Packet หรือ Datagram โดยข้อมูลจะถูกแบ่งออกเป็นย่อยๆ มีประโยชน์ คือ ทำให้เครือข่ายนั้นสามารถรองรับการติดต่อ และรับส่งข้อมูลกันได้อย่างราบรื่นไม่ติดขัด หรือถ้าเกิดปัญหาเครือข่ายทำงานช้า เมื่อมีการรับส่งข้อมูลขนาดใหญ่ เนื่องจากสายสัญญาณเชื่อมโยงเป็นสื่อที่ต้องแบ่งกันใช้ นอกจากนี้การแบ่งข้อมูลออกเป็นส่วนย่อยๆ ยังทำให้สามารถเพิ่มกระบวนการตรวจทานความถูกต้องของข้อมูลที่ปลายทาง และแก้ไขเมื่อข้อมูลผิดพลาดหรือตกหล่นได้โดยง่ายอีกด้วย

3) การห่อหุ้ม (Encapsulation)

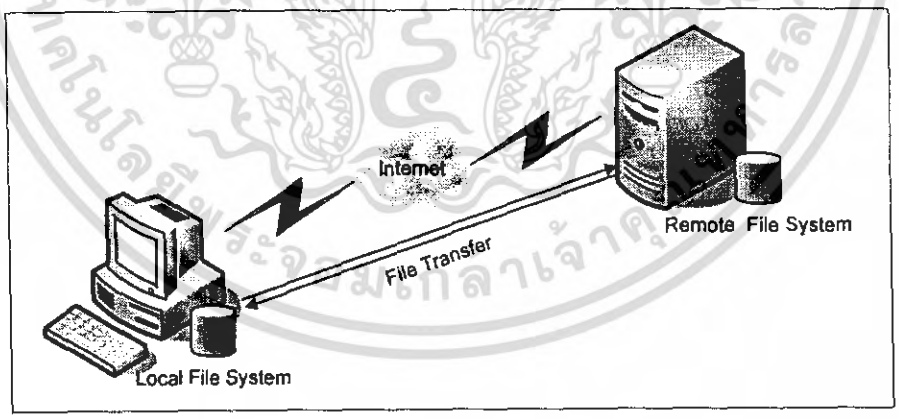
การผนึกข้อมูลหนึ่งให้ไปเป็นอีกรูปแบบหนึ่งนี้ จะเป็นกลไกที่สำคัญของการใช้งานโปรโตคอล TCP/IP มาก โดยที่ขบวนการที่ใช้จะมีขั้นตอนคร่าวๆ ดังรูป



รูปที่ 2.4 ภาพแสดง Header ในแต่ละลำดับชั้นของ TCP/IP

2.4.3 ความรู้เบื้องต้นเกี่ยวกับโปรโตคอล FTP

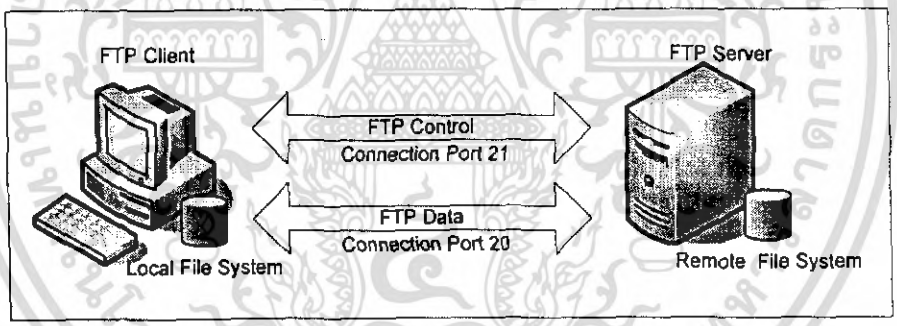
โปรโตคอล FTP (File Transfer Protocol) เป็นส่วนหนึ่งของชุดโปรโตคอล TCP/IP โดยเป็นโปรโตคอลสำหรับการถ่ายโอนไฟล์ระหว่างเครื่องสองเครื่อง ได้แก่ เครื่องคอมพิวเตอร์ที่รับบริการ (FTP Client) กับเครื่องที่เป็นเครื่องให้บริการ (FTP Server) โดยโปรโตคอล FTP นั้นมีมาพร้อมกับอินเตอร์เน็ตสมัยแรกๆ และยังคงเป็นโปรโตคอลที่เป็นที่นิยมในปัจจุบัน โดยโปรโตคอล FTP นั้นถูกอธิบายโดย RFC 959



รูปที่ 2.5 ภาพแสดงการถ่ายโอนไฟล์ระหว่างเครื่องด้วย FTP

ภาพนี้ได้แสดงลักษณะการถ่ายโอนไฟล์ระหว่างเครื่องด้วย FTP โดยทั่วไปเมื่อผู้ใช้ต้องการที่จะโอนไฟล์ระหว่างเครือข่าย ผู้ใช้ก็จะเปิดโปรแกรม FTP เสียก่อน โดยสิ่งที่ผู้ใช้จะต้องทำการระบุในการเชื่อมต่อครั้งแรกก็คือ ชื่อและที่อยู่ของ FTP เซิร์ฟเวอร์ ตามด้วยชื่อล็อกอินและรหัสผ่าน ซึ่งขั้นตอนนั้นจะเริ่มจากไคลเอนต์จะทำการเชื่อมต่อผ่าน TCP กับเซิร์ฟเวอร์ และทำการส่งข้อมูลเกี่ยวกับการล็อกอินเพื่อที่เซิร์ฟเวอร์จะได้ตรวจสิทธิ์ของผู้ใช้ และถ้าตรวจสอบสิทธิ์ผ่านผู้ใช้ก็สามารถอัพโหลดไฟล์หรือดาวน์โหลดไฟล์ระหว่างเครื่องของผู้ใช้และเซิร์ฟเวอร์ได้

โดยจะเห็นว่าโปรโตคอล HTTP ที่ใช้กันทั่วไปในระบบอินเตอร์เน็ตสามารถทำหน้าที่ได้คล้ายกับ FTP เพราะทั้งสองเป็นโปรโตคอลสำหรับการถ่ายโอนไฟล์และนอกจากนี้ทั้งคู่ยังใช้การเชื่อมต่อแบบ TCP เหมือนกัน แต่อย่างไรก็ตามทั้งสองโปรโตคอลมีข้อแตกต่างที่สำคัญ คือ โปรโตคอล FTP จะใช้การเชื่อมต่อ TCP ที่ขนานกันสองการเชื่อมต่อ โดยการเชื่อมต่อแรกจะใช้สำหรับการควบคุมการถ่ายโอนไฟล์ (Control Connection) ในขณะที่การเชื่อมต่อที่สองจะใช้สำหรับการถ่ายโอนข้อมูลหรือไฟล์ (Data Connection) ช่องการเชื่อมต่อควบคุมจะใช้สำหรับการส่งข้อมูล หรือ คำสั่งที่ใช้สำหรับควบคุมการถ่ายโอนไฟล์ระหว่างโฮส เช่น ชื่อล็อกอิน รหัสผ่าน คำสั่งสำหรับการเปลี่ยนไดเร็คทอรี คำสั่งสำหรับการดาวน์โหลด และคำสั่งสำหรับการอัพโหลดไฟล์ เป็นต้น ส่วนช่องทางการเชื่อมต่อข้อมูลนั้นใช้สำหรับการถ่ายโอนไฟล์ ในขณะที่โปรโตคอล HTTP นั้นจะใช้การเชื่อมต่อเดียวสำหรับทั้งการรับและส่งข้อมูล การควบคุมและเว็บเพจ



รูปที่ 2.6 ภาพแสดงช่องทางการเชื่อมต่อของ FTP

จากภาพนี้ได้แสดงถึงช่องทางการเชื่อมต่อของ FTP ซึ่งกระบวนการของการถ่ายโอนไฟล์ด้วย FTP นั้น จะเริ่มจากการที่ไคลเอนต์ สร้างการเชื่อมต่อ TCP กับทางเซิร์ฟเวอร์ผ่านทางพอร์ต 21 ซึ่งการเชื่อมต่อนี้จะเป็นช่องสำหรับการรับส่งข้อมูล รวมไปถึงการควบคุมและถ่ายโอนไฟล์ เมื่อสร้างการเชื่อมต่อเสร็จ ทางฝั่งไคลเอนต์ก็จะทำการส่งข้อมูลการล็อกอิน เช่น ชื่อผู้ใช้และรหัสผ่านไปให้ทางฝั่งเซิร์ฟเวอร์ตรวจสอบสิทธิ์ เมื่อเซิร์ฟเวอร์ตรวจสอบสิทธิ์ผ่าน ทางฝั่งไคลเอนต์ก็จะสามารถทำการดาวน์โหลดหรืออัพโหลดไฟล์ได้

ซึ่งในขั้นตอนการดาวน์โหลดหรืออัพโหลดไฟล์นั้น มีขั้นตอนดังนี้ เช่น เมื่อไคลเอนต์ต้องการดาวน์โหลดไฟล์ ไคลเอนต์ก็จะทำการส่งข้อมูลเกี่ยวกับไฟล์นั้น เช่น ชื่อไฟล์ที่ต้องการไปให้ยังเซิร์ฟเวอร์ ผ่านทางพอร์ต 21 เมื่อเซิร์ฟเวอร์ได้รับคำร้องขอก็จะทำการสร้างการเชื่อมต่อใหม่กับทางฝั่งไคลเอนต์ ซึ่งโดยทั่วไปมักจะสร้างพอร์ตที่ 20 ขึ้นมา และหลังจากนั้นไฟล์ที่ร้องขอนั้นจะถูกทำการถ่ายโอนผ่านทางพอร์ต 20 และเมื่อทำการถ่ายโอนเสร็จ ก็จะทำการปิดพอร์ตที่ 20 จนกว่าจะมีการร้องขอใหม่ ซึ่งอาจเป็นการอัพโหลดหรือดาวน์โหลดไฟล์ เซิร์ฟเวอร์ก็จะทำการสร้างการเชื่อมต่อ TCP ผ่านพอร์ต 20 ขึ้นมาใหม่ แต่ช่องการเชื่อมต่อสำหรับการควบคุมนั้นยังคงสภาพไว้จนกว่าทางฝั่งไคลเอนต์จะยกเลิกหรือออกจากระบบ

2.4.3.1 คำสั่งของโปรโตคอล FTP

ในหัวข้อนี้จะกล่าวถึงคำสั่งต่างๆ ที่รับส่งกันระหว่างไคลเอนต์และเซิร์ฟเวอร์ที่ใช้สำหรับการควบคุมการถ่ายโอนไฟล์ โดยคำสั่งเหล่านี้จะถูกส่งผ่านทางช่องควบคุมที่พอร์ต 21 ซึ่งคำสั่งของ FTP นั้นคนทั่วไปสามารถอ่านและเข้าใจความหมายได้ โดยคำสั่งจะอยู่ในรูปแบบ ASCII คำสั่งที่สำคัญมีดังนี้

คำสั่ง	คำอธิบาย
USER username	ใช้สำหรับลงชื่อสำหรับการล็อกอิน
PASS password	ใช้สำหรับส่งรหัสผ่าน
LIST	ใช้สำหรับไคลเอนต์ส่งการร้องขอให้เซิร์ฟเวอร์ส่งรายการของไฟล์และโฟลเดอร์ในไดเร็กทอรีปัจจุบัน
RETR filename	ใช้สำหรับการดาวน์โหลดไฟล์จากเซิร์ฟเวอร์จากไดเร็กทอรีปัจจุบัน
STOR filename	ใช้สำหรับการอัพโหลดไฟล์ไปยังไดเร็กทอรีปัจจุบันที่เซิร์ฟเวอร์

รูปที่ 2.7 ภาพแสดงตารางคำสั่งของโปรโตคอล FTP

2.4.3.2 การตอบกลับของเซิร์ฟเวอร์

จากคำสั่งด้านบนที่ไคลเอนต์ ได้ส่งผ่านช่องควบคุมไปยังเซิร์ฟเวอร์ โดยการตอบกลับของเซิร์ฟเวอร์ของแต่ละคำสั่งนั้นจะเริ่มต้นด้วยตัวเลขสามหลัก ซึ่งเป็นการบอกสถานภาพของเซิร์ฟเวอร์และอาจมีข้อความต่อท้ายหมายเลขดังกล่าวนี้เพื่ออธิบายรายละเอียดเพิ่มเติมเกี่ยวกับการตอบกลับนั้นๆ ตัวอย่างการตอบกลับจากเซิร์ฟเวอร์ เช่น

- 331 Username OK, Password required
- 125 Data connection already opened; transfer starting
- 425 Can't open data connection
- 452 Error Writing File

สำหรับรายละเอียดเพิ่มเติมเกี่ยวกับคำสั่งและการตอบกลับของเซิร์ฟเวอร์นั้น สามารถศึกษาเพิ่มเติมได้จากข้อมูลของ RFC 959

2.4.4 ความรู้เกี่ยวกับโปรโตคอลอื่นๆที่ใช้เสริมการทำงานในเครือข่าย

นอกจากนี้ TCP/IP ยังมีโปรโตคอลย่อยๆ อื่นๆ อีกที่ทำงานอยู่เบื้องหลัง ซึ่งจะทำงานโดยที่ผู้ใช้ไม่สามารถมองเห็นได้จากโปรแกรม หรือไม่ได้มีการใช้งานโดยตรง เช่น

1) **DNS (Domain Name System)** ที่ทำหน้าที่แปลงข้อมูลชื่อโดเมนเนมหรือชื่อเว็บไซต์ทั้งหลายให้เป็นหมายเลขไอพีแอดเดรส

2) **SNMP (Simple Network Management Protocol)** ใช้ในการควบคุมและตรวจสอบอุปกรณ์ที่อยู่ในเครือข่าย

3) **DHCP (Dynamic Host Configuration Protocol)** ทำหน้าที่แจกจ่ายข้อมูลพารามิเตอร์ของเครือข่ายที่เชื่อมต่ออยู่

โปรโตคอลที่ใช้	Port หรือ Socket ของ UDP หรือ TCP	โปรโตคอลในระดับ Host-to-Host	รายละเอียด
BootP	67	UDP	BOOTstrap Protocol ด้านเซิร์ฟเวอร์
BootP	68	UDP	BOOTstrap Protocol ด้านไคลเอนต์
DHCP	67	UDP	Dynamic Host Configuration Protocol ด้านเซิร์ฟเวอร์
DHCP	68	UDP	Dynamic Host Configuration Protocol ด้านไคลเอนต์
DNS	53	UDP/TCP	Domain Name System
FTP	21	TCP	File Transfer Protocol ด้านเซิร์ฟเวอร์ที่ควบคุม
FTP	20	TCP	File Transfer Protocol ด้านเซิร์ฟเวอร์ที่ส่งข้อมูล
HTTP	80	TCP/UDP	Hyper Text Transfer Protocol ด้านเซิร์ฟเวอร์
NetBT	138	UDP	NetBIOS datagram service
NetBT	139	TCP	NetBIOS session service
SMTP	25	TCP	Simple Mail Transfer Protocol ด้านเซิร์ฟเวอร์
SNMP	161	UDP	Simple Network Management Protocol ด้านเซิร์ฟเวอร์
SNMP	162	UDP	SNMP trap manager
Telnet	23	TCP	Teletype Network Protocol
TFTP	69	UDP	Trivial File Transfer Protocol
WINS	137	UDP	Windows Internet Names Service

รูปที่ 2.8 ตารางสรุปหมายเลข Port ที่ใช้งานโดย TCP

2.5 การทำงานบนระบบปฏิบัติการวินโดวส์

2.5.1 ความรู้ทั่วไปเกี่ยวกับรีจิสตรีในระบบปฏิบัติการวินโดวส์

2.5.1.1 โครงสร้างของรีจิสตรี

รีจิสตรีเป็นฐานข้อมูลที่วินโดวส์ ใช้เพื่อกำหนดวิธีการทำงานต่างๆ โดยการเปลี่ยนแปลงค่าของรีจิสตรีจากหนึ่งไปสู่อีกอย่างหนึ่ง ก็จะทำให้การทำงานของระบบต่างไปจากเดิม โดยข้อมูลที่เก็บอยู่ในรีจิสตรีบางข้อมูลก็ธรรมดามาก และบางข้อมูลก็อาจทำให้ระบบปฏิบัติการทำงานผิดพลาดไปเลย

ลักษณะการจัดเก็บข้อมูลของรีจิสตรี จะใช้รูปแบบที่ไม่เหมือนกับระบบฐานข้อมูลทั่วๆ ไป คือ จะใช้รูปแบบของ Tree หรือต้นไม้ พูดอย่างนี้ทำให้มองภาพไม่ออก จริงๆ แล้วเป็นสิ่งที่คุณคุ้นเคยอยู่แล้ว คือ รูปแบบของไฟล์ และโฟล์เดอร์ในวินโดวส์เอ็กพลอเรอร์ (Windows Explorer) แต่เปลี่ยนคำจำกัดความใหม่ จากที่เรียกว่าโฟล์เดอร์ก็เปลี่ยนเป็นเรียกว่า ไฮฟ์ (Hive) และส่วนที่เป็นไฟล์เปลี่ยนเป็นส่วนของข้อมูลที่ใช้กำหนดค่าต่างๆ

ในส่วนของข้อมูลก็ใช้รูปแบบของการจับคู่ง่ายๆ (หรือเรียกว่า Hash Table) โดยแบ่งเป็น 3 ส่วน คือ ชื่อของข้อมูล(Name), ชนิดของข้อมูล (Type), และค่ากำหนด (Value)

โดยทุกๆ ไฮฟ์จะมีการสร้างข้อมูลในแบบการจับคู่ (Hash) แบบนี้ทั้งสิ้น เพียงแต่ว่าจะกำหนดไว้มากหรือน้อยเท่านั้น การมีข้อมูลที่มากไม่ใช่มีความสำคัญมาก แต่คือมีฟังก์ชันใช้งานมาก ความสำคัญจะขึ้นอยู่กับว่า ไฮฟ์นี้มีหน้าที่ในการควบคุมการทำงานอะไร

2.5.1.2 โครงสร้างของรีจิสตรีไฮฟ์

ไฮฟ์ คือ การแบ่งประเภทใช้งานขององค์รวม แบ่งออกเป็นชนิด (Catagory) ได้ดังนี้

1) HKEY_USERS

ไฮฟ์นี้บรรจุค่ากำหนดที่เกี่ยวข้องกับการใช้งานของผู้ใช้ทุกๆ คนที่มีสิทธิในงานบนเครื่องๆ นี้ โดยผู้ใช้แต่ละแอคเคานต์ (ผู้ใช้ 1 คน สามารถมีได้มากกว่า 1 แอคเคานต์) จะถูกแยกออกเป็น SubHive Key ภายใต้ไฮฟ์นี้แบบ 1 แอคเคานต์ / ไฮฟ์ โดยใช้หมายเลขของแต่ละแอคเคานต์เป็นชื่อของ SubHive Key (Security ID + relative ID หรือ SID)

2) HKEY_CURRENT_USER

เป็นไฮฟ์ที่คัดลอกมาจาก HKEY_USER โดยเลือกเฉพาะผู้ที่กำลังใช้งานบนเครื่องๆ นี้ ข้อกำหนดต่างๆ ล้วนเกี่ยวกับผู้ใช้เท่านั้น ไม่มีเรื่องอื่นเช่นระบบหรือฮาร์ดแวร์ใดๆ มาเกี่ยวข้อง เช่น ซอฟต์แวร์ที่ผู้ใช้ติดตั้ง ,ภาษาที่ผู้ใช้เลือกทำงานบนคีย์บอร์ด หรือจอแสดงผล, การแสดงแสง/เสียงโต้ตอบจากการทำงานของผู้ใช้ หรือเลือก Logon Script ในการรันขณะบูตเครื่อง เป็นต้น

3) HKEY_CLASSES_ROOT

เป็นไฮฟ์ที่คัดลอก (Reflex) มาจาก HKEY_LOCAL_MACHINE ภายใต้ Software\Classes SubKey ไฮฟ์นี้บรรจุรายละเอียดของออปเจ็กต์ (Object) ต่างๆ ที่ใช้งานโดยแอพพลิเคชั่นต่างๆ ทั้งจาก

แอพพลิเคชั่นของวินโดวส์เอง และที่ติดตั้งเพิ่มเติม เช่น ในการเปิดหรือแก้ไขแอพพลิเคชั่นหนึ่งๆ (Open/Edit) วินโดวส์จะเลือกออปเจ็กต์ในการทำงาน หรือไอคอน (Icon) ของรีไซเคิลบิน (RecycleBin) เลือกจากออปเจ็กต์อะไร และมีออปชัน (Option) หรือทางเลือกอะไรบ้าง

4) HKEY_CURRENT_CONFIG

เป็นไฮฟ์ที่คัดลอก (หรือ Reflex) มาจาก HKEY_LOCAL_MACHINE ภายใต้ Software และ System SubKey ไฮฟ์นี้บรรจุรายละเอียดของการบริการ หรือการควบคุมการทำงานของระบบปฏิบัติการ ที่ยอมให้ได้กับผู้ใช้ที่กำลังใช้งานอยู่ เช่น ผู้ใช้นี้สามารถใช้โทรศัพท์ในการล็อกออน (Logon) สู่ระบบได้หรือไม่ เลือกมัลติมิเดียการ์ดไหนในการแสดงผล เป็นต้น

5) HKEY_LOCAL_MACHINE

เป็นซับทรีที่เก็บข้อมูลเกี่ยวกับระบบในเครื่องคอมพิวเตอร์ ได้แก่ ฮาร์ดแวร์และข้อมูลของระบบปฏิบัติการ เช่น ชนิดของระบบบัส ขนาดของหน่วยความจำ และ ซอฟต์แวร์ ไดรเวอร์ เป็นต้น

2.5.1.3 โครงสร้างของรีจิสตรีคาต้า

รีจีสตรีคาต้าประกอบด้วย 3 ส่วน คือ ชื่อ (Name), ประเภท (Type) และค่ากำหนด (Value)

1) ชื่อ (Name)

เป็นส่วนที่ใช้อธิบายหรือสื่อความหมาย ถึงเป้าหมายในการใช้งานนั้นๆ เช่น AttachedTo ใช้กับ Modem Hive คือโมเด็มตัวนี้ต่อเข้ากับ COM port อะไร หรือ NameServer ใช้อ้างอิงถึงแอดเดรสของเครื่องเซิร์ฟเวอร์ ที่ทำหน้าที่เป็น DNS (Domain Name Service) เป็นต้น ในทางปฏิบัติสามารถจัดแยกชนิดของชื่อได้ 3 กรณี

- **กรณีที่ 1** เมื่อผู้ใช้ติดตั้งระบบปฏิบัติการ หรือซอฟต์แวร์ วินโดวส์จะกำหนดค่าตั้งที่เป็นมาตรฐานให้ หรือเมื่อผู้ใช้ทำการแก้ไขหรือปรับปรุงผ่านทางกราฟิกโหมด ภายหลังที่ได้ติดตั้งซอฟต์แวร์แล้ว
- **กรณีที่ 2** เมื่อผู้ใช้ทำการแก้ผ่านทางกราฟิกโหมด และคลิกที่ปุ่ม Advance หรือมีเซอร์วิสที่รองรับได้ เช่น เน็ตเวิร์กที่ไม่ใช้ DHCP Server (Dynamic Host Configuration Protocol) ขณะติดตั้ง TCP จะไม่สามารถกำหนดค่าที่เกี่ยวกับ DHCP ได้ หรือแสดงออกในลักษณะที่เรียกว่า Gray Out หรือกระทำไม่ได้
- **กรณีพิเศษ** คือ วินโดวส์จะไม่สร้างให้ผู้ใช้ต้องทำการสร้างขึ้นเอง โดยมีหลายเหตุปัจจัย เช่น ด้านความปลอดภัย สร้างมาตรฐานใหม่ที่แตกต่างไปจากปกติ หรือเป็นกรณีเฉพาะที่ใช้วินโดวส์ร่วมกับระบบปฏิบัติการอื่น หรือกับอุปกรณ์บางอย่างที่ยังไม่มีความเหมาะสมกันทุกด้าน เช่น WallpaperOriginX คือ กำหนดตำแหน่งที่ใช้แสดงภาพบิตแมปบนเด็สก์ทอป หรือ LogonMsg.PL คือ ชื่อของ Perl Script ที่รันขณะล็อกออน เป็นต้น

2) **ประเภท (Type)**

เป็นส่วนที่ใช้แยกลักษณะการใช้งาน โดยขออธิบายในแต่ละประเภท สามารถแบ่งออกได้ดังนี้

- **REG_SZ** เป็นชนิดที่ผู้ใช้จะพบมากเป็นที่สอง แต่จะเกี่ยวข้องกับผู้ใช้มากที่สุด ค่ากำหนดสำหรับชนิด SZ จะเป็นอักขระ (String) รวมทั้งที่เป็นตัวเลขด้วย เช่น The World is my Oyster! และ 0 0 255 ในตัวอย่างตอนต้น
- **REG_EXPAND_SZ** เป็นชนิดที่คล้ายกับแบบ SZ คือ เป็นอักขระหรือ String แต่เพิ่มระดับการใช้งานให้มากขึ้น คือ สามารถบรรจุตัวแปรของระบบได้ เรียกว่า System Environment Variables เช่น การกำหนดเส้นทางของแอพพลิเคชั่น %SYSTEMDRIVE%\%SYSTEMROOT%\SYSTEM32 โดย %SYSTEMROOT% คือ ไดเรกทอรีที่ติดตั้งระบบปฏิบัติการ
- **REG_MULTI_SZ** เป็นชนิดที่เหมือนกับ SZ ทุกประการ แต่สามารถกำหนดค่าได้มากกว่าหนึ่งบรรทัด หรือใช้งานในแบบรายการ (List) เช่น รายชื่อของแอพพลิเคชั่นที่อนุญาตให้ผู้ใช้สามารถรันได้ รายการแอดเดรสของ WINS Server
- **REG_DWORD** เป็นชนิดที่พบมากที่สุด และค่อนข้างสับสน เนื่องจากค่ากำหนดสำหรับชนิดนี้ เป็นไปได้ทั้งเลขฐาน 10 และ 8 (Decimal \ Hexadecimal) โดยวินโดวส์ได้เตรียมทูลเพื่อให้ผู้ใช้สามารถแปลงกลับไปกลับมาได้ โดยการดับเบิลคลิกที่ชื่อนั้นๆ
- **REG_BINARY** เป็นชนิดที่มักพบใน Hardware and Security Hive เนื่องจากยากในการทำความเข้าใจโดยมนุษย์ แต่ง่ายสำหรับไมโครชิป ในการประมวลผล เพราะเป็นเลขฐานสอง และที่ไฮฟ์สองไฮฟ์ดังกล่าว เราก็ไม่ควรไปยุ่งอยู่แล้ว เพราะมักจะต้องขึ้นอยู่กับผู้ผลิตอุปกรณ์นั้นๆ โดยเฉพาะ
- **REG_RESOURCE_LIST** ใช้กับอุปกรณ์ประเภทดีไวซ์ไดรเวอร์ (Device Driver) เท่านั้น เช่น วิดีโอไดรเวอร์ (Video Driver) ,ซาวนด์ไดรเวอร์ (Sound Driver) หรือเน็ตเวอร์กไดรเวอร์ (Network Driver) และมีเฉพาะภายใน Hardware SubHive เท่านั้น
- **REG_FULL_RESOURCE_DESCRIPTOR** ใช้กับ Computer Hardware หรืออุปกรณ์ต่างๆ (Peripheral) เช่น ดิสก์คอนโทรลเลอร์ (Disk Controller) ,คีย์บอร์ด (Keyboard) โดยค่าที่กำหนดนี้จะถูกสร้างใหม่ทุกครั้งที่บูตเครื่อง และบันทึกลงรีจิสตรีโดยโปรแกรม NTDETECT
- **REG_RESOURCE_REQUIREMENTS_LIST** ใช้กับอุปกรณ์ประเภทปลั๊กแอนด์เพลย์
- **REG_NONE และ REG_UNKNOWN** ไม่มีค่า

3) **ค่ากำหนด (Value)**

เป็นส่วนที่ใช้ระบุจำนวน โดยทั่วไปสามารถแบ่งออกเป็นรูปแบบต่างๆ ดังนี้

- **แบบทางเลือก (Boolean)** มักพบใน REG_DWORD และ REG_SZ เช่น 0x00000000 (Hexadecimal) คือ ผิด และ 0x00000001 คือ ถูก และบางครั้งก็ใช้เป็นสตริง (String) เช่น TRUE หรือ FALSE หรือ YES หรือ NO
- **แบบรายการ (List or Mode or Option)** เช่น 0 คือ ค่ามาตรฐาน, 1 คือ ทางเลือกที่ 1 และ 2 คือ อีกทางเลือกหนึ่ง
- **แบบกำหนดเป็นช่วง (Range)** กำหนดค่าขอบเขตค่าต่ำสุดและสูงสุด โดยสามารถกำหนดเป็นเท่าไหร่ก็ได้ภายในช่วงนี้
- **แบบตาราง (Table Flags)** โดยจะใช้กับกรณีที่มีออปชั่นหลายๆ ออปชันให้เลือก และสามารถเลือกได้มากกว่า 1 ออปชัน หรือเรียกว่า Checkbox Group ดังกรณีของ Attributes = 00011 ในตัวอย่างตอนต้น การใช้ 0 และ 1 ก็คือ Checkbox แต่ละอันนั่นเอง
- **แบบข้อความ (String)** เช่น ไอพีแอดเดรส หรือ Computer NetBIOS Name
- **แบบข้อความที่บรรจุตัวแปรของระบบ (Environment)** เช่น ค่า %SYSTEMDRIVE% หรือค่า %USERNAME% เป็นต้น
- **แบบกำหนดค่าทีละหลายๆ ค่า** เช่น ใช้ Comma (,) หรือ White Space (ช่องว่าง) ในการแบ่งแยกค่าออกจากกัน เรียกว่า Comma Seperated Value (CVS) เช่น กำหนดอาร์กิวเมนต์อาร์เรย์ (Argument Array) ให้กับแอพพลิเคชั่น(Application) หรือสคริปต์ (Script) หรือกำหนดรายชื่อของแอพพลิเคชั่นที่ต้องการให้รัน
- **แบบกำหนดเป็นกลุ่มของสตริง (Order List)** โดยกำหนดเป็นสตริงแยกกันแต่ละบรรทัด ใช้กับกรณี REG_MULTI_SZ

2.5.2 ความรู้เกี่ยวกับการบูตระบบของระบบปฏิบัติการวินโดวส์

รีจิสตรีและขั้นตอนการบูตวินโดวส์สัมพันธ์กันอย่างไรในทางปฏิบัติแล้ว ผู้ใช้อาจไม่จำเป็นต้องรู้ แต่บางกรณีที่ที่เกิดความเสียหายกับระบบ ทำให้คุณล็อกออนไม่ได้ อาจเกิดในขั้นตอนของการโหลดรีจิสตรี

ดังนั้นการทำความเข้าใจขั้นตอนการโหลดรีจิสตรีระหว่างบูตเครื่อง จะทำให้คุณเข้าใจในเนื้อหาของวินโดวส์รีจีสตรีได้สมบูรณ์มากขึ้น และสามารถนำไปใช้ผสมผสานกับความรู้ในด้านอื่นได้

ในขั้นตอนการบูตเครื่อง สามารถแบ่งออกได้เป็น 2 ส่วนใหญ่ๆ คือ ส่วนที่ไม่เกี่ยวข้องกับรีจิสตรี (คิดเป็น 35 เปอร์เซนต์โดยประมาณ) และในส่วนของรีจิสตรีโดยตรง ซึ่งสำหรับส่วนแรกจะใช้เวลาน้อยกว่า เพราะเป็นการทำงานที่เกี่ยวข้องกับกิจกรรมไม่มาก

2.5.2.1 ขั้นตอนของการบูตระบบของระบบปฏิบัติการวินโดวส์

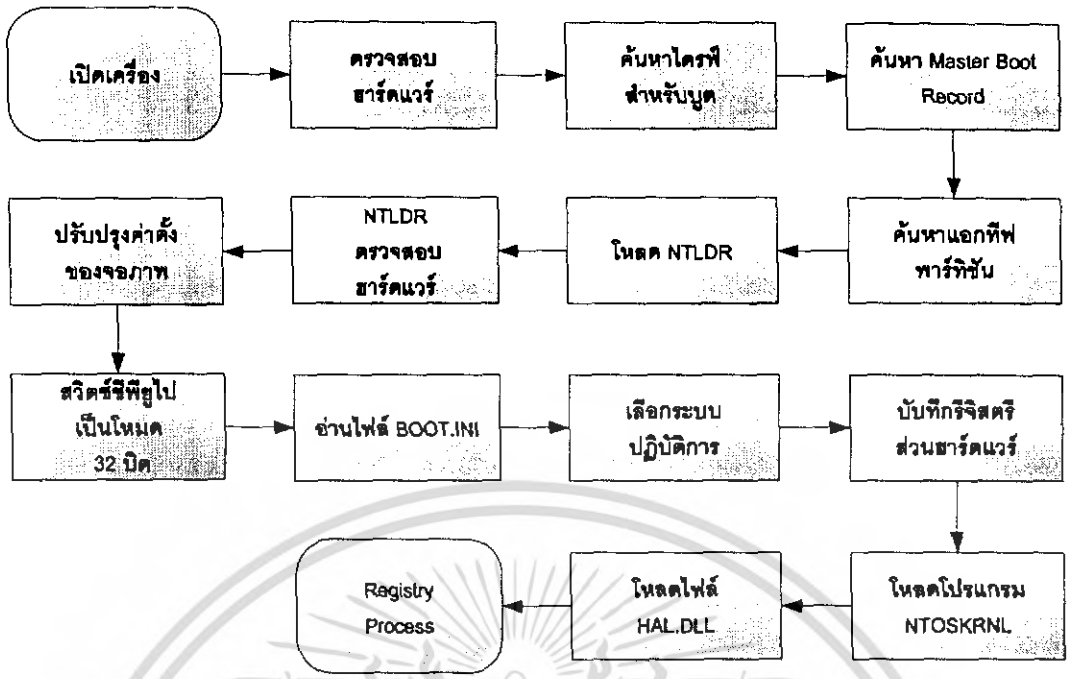
เมื่อผู้ใช้เปิดเครื่อง (Turn On) ระบบจะทำการตรวจสอบฮาร์ดแวร์พื้นฐานผ่านระบบ BIOS (Basic Input \ Output Service) เช่น อ่านค่าความจุของฮาร์ดดิสก์ ขนาดของแรม ชนิดของคีย์บอร์ด การ์ดหรืออะแดปเตอร์ ค้นหาไดรฟ์ (แบบ Physical) ที่ติดตั้งภายในเครื่องทั้งหมดและค้นหา MBR (Master Boot Record) เพื่อทราบการจัดแบ่งพาร์ทิชันในแต่ละไดรฟ์

ต่อมาทำการค้นหา Active Partition คือ พาร์ทิชันที่อยู่ติดกับ MBR คือ Cylinder 0 Header 1 Sector 1 โหลดโปรแกรม NTLDR (ไม่มีส่วนขยายอยู่ที่ C:\root dir มิฉะนั้นจะบูตไม่ได้) โดยโปรแกรม Ntldr จะทำการตรวจสอบฮาร์ดแวร์ทั้งหมดอย่างละเอียด

หลังจากนั้นโปรแกรม Ntldr ทำการกำหนดค่าให้การ์ดวิดีโอ (Video Card) ให้เป็นแบบ Alpha numeric mode (80*25 16-color) โปรแกรม Ntldr ทำการเปลี่ยนโหมดการทำงานของโปรเซสเซอร์ (ไมโครซอฟต์ยึดหลักการทำงานตามชีพียูของ Intel) จาก 8 – 16 บิต (real-mode) เป็นโหมด 32 บิต และทำการตรวจสอบว่าภายในเครื่องนั้น มีการใช้ระบบปฏิบัติการอื่นติดตั้งในเครื่องเดียวกันนี้หรือไม่ สุดท้าย โปรแกรม Ntldr จะโหลดโปรแกรมอื่นอีก 3 โปรแกรม ให้มาทำหน้าที่ในด้านอื่นๆ คือ NTDETECT.COM, Boot.ini และ BOTSECT.DOS (และ NTBOOTDD.SYS กรณีที่คุณใช้ฮาร์ดดิสก์แบบ SCSI ซึ่งทำงานแบบ Multi-Tasking)

ทำการอ่านไฟล์ Boot.ini โดยเนื้อหาภายในประกอบด้วยรายการของระบบปฏิบัติการ และตำแหน่งของพาร์ทิชัน แสดงทางเลือกของระบบปฏิบัติการที่อ่านจาก Boot.ini และโปรแกรม NTDETECT.COM รับผลการตรวจสอบฮาร์ดแวร์จาก Ntldr แล้วบันทึกลงในรีจิสตรี จากนั้นโปรแกรม Ntldr ทำโหลดโปรแกรม NTOSKRNL.EXE (อยู่ในโฟลเดอร์ System32) เข้าไปไว้ในแรม แต่ไม่เอ็กซีคิวต์ โปรแกรม NTOSKRNL.EXE ใช้เชื่อมโยงการทำงานต่างๆ ระหว่างวินโดวส์คอมโพเนนต์ด้วยกัน (เช่น Object Manager, ดีไวซ์ไดร์เวอร์)

สุดท้ายโปรแกรม Ntldr ทำการโหลดไฟล์ HAL.DLL (อยู่ในโฟลเดอร์ System32) เข้าไปไว้ในแรม ไฟล์ HAL.DLL ทำให้โปรแกรม NTOSKRNL.EXE สามารถติดต่อกับฮาร์ดแวร์ได้ และจะเข้าสู่กระบวนการของรีจิสตรี (Registry Process)

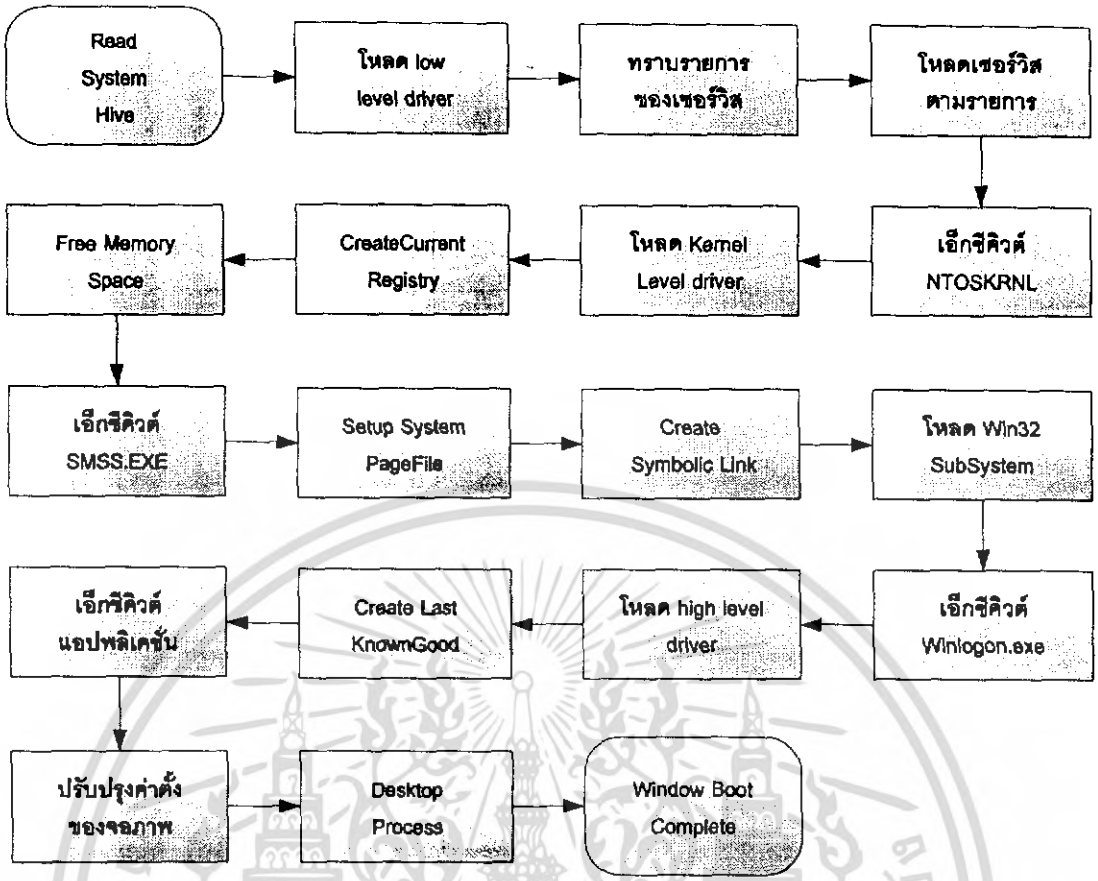


รูปที่ 2.9 ภาพแสดงขั้นตอนของการบูต

2.5.2.2 ขั้นตอนของการจัดการรีจิสตรีภายหลังการบูตระบบ

ขั้นตอนของรีจิสตรีในการบูตระบบในส่วนของรีจิสตรีนั้น เริ่มแรกจะทำการอ่านค่ากำหนดของ System Hive ทั้งหมด โหลดดีไวซ์ไดรเวอร์ที่มีค่ากำหนดของ (Name) Start เท่ากับ 0 ที่หน้าจอจะแสดงการพิมพ์ '.' โดยหนึ่งจุดต่อ 1 ไดรเวอร์ โดยไปที่คีย์ SYSTEM\CurrentControlSet\Control\Service GroupOrder ต่อมาโหลดเซอร์วิสจากค่ากำหนดของคีย์ในข้อ 3 โดยกระทำการโหลดเรียงลำดับจากบนลงล่าง หรืออีกนัยหนึ่งคือ จากซ้ายไปขวา และเอ็กซีคิวต์ NTOSKRNL.EXE โดยขั้นตอนนี้หน้าจอจะเปลี่ยนเป็นสีน้ำเงิน จากนั้นโหลดดีไวซ์ไดรเวอร์ที่มีค่ากำหนดของ (Name) Start เท่ากับ 1 ที่หน้าจอจะแสดงการพิมพ์ '.' โดยหนึ่งจุดต่อหนึ่งไดรเวอร์ โปรแกรม NTDETECT.COM บันทึกรีจิสตรีในส่วนของฮาร์ดแวร์ และเลือก ControlSet จากคีย์ System \ Select

ต่อมาทำการเคลียร์เมโมรี กรณีที่ทำงานร่วมกับระบบเน็ตเวิร์กอื่นๆ ที่ต้องการเมโมรีสเปซ (Memory Space) มากในการโหลดไฟล์ไดรเวอร์ของระบบปฏิบัติการนั้นๆ และเอ็กซีคิวต์ SMSS.EXE (Session Management Service)



รูปที่ 2.10 ภาพแสดงขั้นตอนของรีจิสตรีในการบูต

ทำการสร้าง PageFile.SYS โดยคัดลอกจากแรมไปสู่ดิสก์ (วินโดวส์บริหารหน่วยความจำของระบบผ่าน PageFile นี้ และดูได้ที่ซับคีย์ CurrentControlSet\Control\Session Manager\Memory Management) กำหนด Alias Name ให้กับ DOS Devices และโหลดโปรแกรม Win32 SubSystem จากนั้นทำการบันทึกรีจิสตรีในการบูตครั้งนี้ไปที่ CurrentControlSet และ CloneControlSet

จากนั้นทำการ Logon Process โดยเอ็กซีคิวต์ Winlogon.exe, LSASS>EXE (Local Security Authority SubSystem) และ SPOOLSS.EXE เพื่อรวบรวมระดับความปลอดภัย และบริการ Print Spool ของผู้ใช้ทั้งหมดจาก Domain Controller จากนั้นแสดง Logon Screen ให้ใส่ชื่อและพาสเวิร์ด และเอ็กซีคิวต์ SCREG.EXE (Service Controller) เพื่อโหลดดีไวซ์ไดรเวอร์ที่มีค่ากำหนดของ (Name) Start เท่ากับ 2 ในขั้นตอนนี้ถ้าไม่สามารถโหลดเซอร์วิสได้ จะเกิดไดอะล็อกซี้แจงและเขียน Message to Log บันทึกค่ารีจิสตรีที่ได้จากการบูตครั้งนี้ คือ CloneControlSet ไปที่คีย์ตรงกับค่า (Value) ของ LastKnownGood

สุดท้ายเอ็กซีคิวต์แอพพลิเคชั่นที่อยู่ในลิส (List) ของคีย์ SOFTWARE \ Microsoft \ Windows \ CurrentVersion \ Run Desktop Process เช่น ทาสก์บาร์ (Taskbar), ไอคอน (Icon), บิตแม็พ (Bitmap), Start Menu เป็นต้น

2.5.3 ความรู้เกี่ยวกับการจัดการระบบผู้ใช้ของระบบปฏิบัติการวินโดวส์

2.5.3.1 ยูสเซอร์โปรไฟล์ (User Profiles)

ยูสเซอร์โปรไฟล์ คือ การจำแนกผู้ใช้แต่ละคนโดยระบบจะรับค่าจาก Logon Dialog คือ ชื่อและรหัสผ่านในระหว่างการบูต จากนั้นจะแปลงชื่อและรหัสผ่านเป็น Security ID และ Relative ID [S-x-x-xxxx-x.....] เพื่อใช้ในการโหลดข้อมูลที่ไฮฟ์ HKEY_USER \ [S-x-x-xxxx-x.....] หรือ HKEY_CURRENT_USER และบันทึกลงไฟล์ NTUSER.DAT และใช้ข้อมูลจากไฟล์นี้เป็น Current User Profiles

โดยแสดงออกในรูปของ Symbolic Link ภายใต้โฟลเดอร์ %SystemRoot %\ Profiles \ "username" และ %SystemRoot%\Profiles\AllUsers และภายในโฟลเดอร์ Default User จะบรรจุ Symbolic Link ที่เป็นมาตรฐานที่สำหรับเป็นต้นแบบ ในกรณีที่มีการสร้างผู้ใช้ขึ้นมาใหม่ และโฟลเดอร์นี้เป็นข้อมูลที่สร้างมาจาก HKEY_USERS \ .DEFAULT

จากหลักการของสร้างยูสเซอร์โปรไฟล์ที่กล่าวมา จะเห็นว่ารีจีสตรีคีย์ที่น่าสนใจคือซับไฮฟ์ current user เพื่อกำหนดค่ารีจิสตรีที่ต้องการกับผู้ใช้ที่ท่านทำการเลือกได้ และ ที่ซับไฮฟ์ Default เหมาะสมกับการกำหนดเป็นมาตรฐานขั้นต่ำให้กับผู้ใช้ทั่วไปในเครือข่าย หรือ ในกรณีที่ท่านติดตั้งแอพพลิเคชั่นเพิ่มเติมใหม่ และ ต้องการให้ผู้ใช้ทุกคนเข้าถึงแอพพลิเคชั่นดังกล่าวได้ แทนที่ท่านจะสร้างชอร์ตคัต แล้วนำเอาไปใส่ในทุกโฟลเดอร์ของทุกๆผู้ใช้ ท่านเพียงนำไปใส่ใน All User ก็เพียงพอแล้ว

เช่นเดียวกันนี้ ท่านสามารถใช้วิธีการเดียวกันนี้กับซับไฮฟ์ .DEFAULT ก็เท่ากับว่าท่านกำลังจัดการกับผู้ใช้ทุกคนที่มีแอคเคานต์อยู่บนเครื่องๆนี้ และถ้าจัดการกับไฟล์แบบ Roaming จะทำให้ท่านควบคุมผู้ใช้ได้มากขึ้นไปอีก โดยการทำงานของ Roaming Profile เปรียบเสมือนการสร้าง Template Profile นั่นเอง

โดยยูสเซอร์โปรไฟล์เปรียบเสมือนการสร้างความจำได้ของระบบ สำหรับการใช้งานของผู้ใช้แต่ละคน โดยที่ระบบวินโดวส์ NT ได้เตรียมเครื่องมือให้ท่าน ในการที่สร้างการทำงานด้านนี้ โดยสามารถสร้างให้เป็นแบบที่ผู้ใช้สามารถกำหนดความเป็นตัวของตัวเองได้ (แต่ต้องอยู่ภายในกรอบของ System Policy) หรือแบบที่ถูกกำหนดไว้ในส่วนกลาง ที่ผู้ใช้จะไม่สามารถทำการเปลี่ยนแปลงให้เกิดขึ้นอย่างถาวรได้ (แต่สามารถทำขึ้นชั่วคราวในแต่ละครั้งที่ล็อกออนได้) โดยแบ่งได้ 3 แบบ

1) โปรไฟล์แบบ Local

โปรไฟล์แบบ Local คือ โปรไฟล์ที่ถูกสร้างบนเครื่องๆนั้น (โดย Account Manager) โฟลเดอร์ user และค่ารีจิสตรีจะบันทึกอยู่บนเครื่องๆนั้น

2) โปรไฟล์แบบ Mandatory

โปรไฟล์แบบ Mandatory คือ โปรไฟล์ที่ถูกสร้างบนเครื่องไคลแอนต์ทั่วไป แต่จะระบุที่อยู่ของ User profile ไปที่เครื่องเซิร์ฟเวอร์ (เครื่องใดก็ได้ที่สามารถเชื่อมโยงถึงกันได้ และ โฟลเดอร์ที่บรรจุ user profile นี้ต้อง share ให้กับ EveryOne ก่อน) โดยโปรไฟล์นี้ยังไม่เป็นโปรไฟล์แบบ Mandatory Profile จนกว่าคุณจะเปลี่ยน File Extension จาก NTUSER.DAT ให้เป็น NTUSER.MAN (ที่เซิร์ฟเวอร์ที่ใช้เก็บโปรไฟล์) โดนไฟล์นี้จะถูกสร้างในทุกๆโฟลเดอร์ user

3) โปรไฟล์แบบ Roaming

โปรไฟล์แบบ Roaming คือ Mandatory Profile นั่นเองแต่ Roaming จะใช้ระบุที่ ไคลแอนต์ และ Mandatory จะใช้ระบุที่เซิร์ฟเวอร์ หรือ อีกในหนึ่ง Mandatory คือ โปรไฟล์ที่สร้างขึ้นเพื่อใช้เป็น Template Profile และ อยู่บนเครื่องเซร์ฟเวอร์ ส่วน Roaming คือ โปรไฟล์ที่คัดลอกมาจาก Mandatory

2.5.3.2 ยูสเซอร์แอคเคานท์ (User Account)

ยูสเซอร์แอคเคานท์เป็นตัวที่จะใช้ล็อกออนเข้าสู่เซิร์ฟเวอร์ หรือ โดเมน ทำให้ยูสเซอร์สามารถเข้าใช้งานทรัพยากรต่างๆบนระบบได้ ปกติแล้ว User Account จะถูกสร้างขึ้นอัตโนมัติในขณะติดตั้งระบบปฏิบัติการ โดยจะมีทั้ง User Account ของระบบ เช่น Administrator ที่ใช้ล็อกออนเข้าสู่ระบบครั้งแรก Guset เป็นยูสเซอร์ชั่วคราวที่เข้าสู่ระบบ สามารถจะแบ่ง User Account ออกเป็น 4 ประเภทคือ

1) Administrator

เป็น Built-in User ที่คอยดูแลบริหารระบบ มีความสำคัญมากเพราะจะทำหน้าที่ดูแลทุกอย่างตั้งแต่สร้าง User Account การกำหนดสิทธิการเข้าใช้ทรัพยากร การจัดการบริหารทรัพยากรให้เหมาะสม การสร้างลบออบเจ็กต์ การดูแล และบำรุงรักษาฐานข้อมูล Active Directory การเพิ่ม-ลดสมาชิกในโดเมน

2) Guest

เป็น Built-in User แต่เป็นยูสเซอร์ที่ใช้เข้าสู่ระบบชั่วคราว สามารถจะดูข้อมูลธรรมดาได้เท่านั้น ไม่สามารถจะแก้ไขเปลี่ยนแปลงข้อมูลใดๆ ในทางปฏิบัติควรยกเลิกยูสเซอร์นี้เพื่อป้องกันการบุกรุกจากแฮกเกอร์และเพื่อความปลอดภัยของระบบ

3) User

เป็นยูสเซอร์ที่ถูกสร้างขึ้นโดยผู้ดูแลระบบหรือ Admin โดยจะกำหนดสิทธิในการเข้าถึง และใช้งานทรัพยากรต่างๆบนเซิร์ฟเวอร์ และ โดเมน ผู้ดูแลระบบยังสามารถกำหนดสิทธิ์ให้ยูสเซอร์ที่สร้างอยู่ในกรุ๊ป Adminstrators , Backup Operators , Print Operators , Replicator เพื่อช่วยทำหน้าที่สำคัญต่างๆ เหล่านี้ได้ด้วย

4) Computer

เป็นคอมพิวเตอร์แอคเคาน์ที่สร้างขึ้นบนเครื่องโดเมนคอนโทรลเลอร์ เพื่อเข้าใช้เป็นสมาชิกโดเมนไม่ต้องมีรหัสผ่านเหมือน user account

2.5.3.3 แอคทีฟไดเรคทอรีของผู้ใช้งาน (Active Directory)

เป็นเครื่องมือหลักในการแก้ไข เปลี่ยนแปลงฐานข้อมูลแอคทีฟไดเรคทอรีที่เกี่ยวกับยูสเซอร์ รายชื่อเครื่องคอมพิวเตอร์ การสร้างหรือลบออบเจกต่างๆ รวมทั้งคอนเทนนเนอร์ การสร้างแอคเคาน์ยูสเซอร์ และ คอมพิวเตอร์แอคเคาน์ การกำหนดสิทธิ์ต่างๆในการเข้าใช้ทรัพยากรบนโดเมน

1) Container

เป็นคอนเทนเนอร์ธรรมดาจะมี Builtin , Computer , ForeignSecurity Principles และ Users เอาไว้เก็บกรุ๊ปยูสเซอร์ (Adminstrators, Account Operators, Backup Operators, Print Operators ฯลฯ ซึ่งเป็น Builtin Domain Local Group) ชื่อเครื่องคอมพิวเตอร์ (ทั้งเครื่องโดเมนคอนโทรลเลอร์ เซิร์ฟเวอร์ และ ไคลเอนต์ รายชื่อยูสเซอร์ (Adminstrator, IUSR_, IWAM_, Guest ฯลฯ ซึ่งเป็น Builtin User Account)

2) Organizational Unit (OU)

เป็นคอนเทนเนอร์พิเศษเอาไว้เก็บออบเจกต่างๆ เช่น Users Account, Computer Account, Group User แต่ยังมีความสามารถในการบริหารจัดการ เช่น ถ้าต้องการกำหนด Security Policy ให้กับออบเจ็กต์แบบ User , Computer ก็ไม่ต้องเสียเวลาไปกำหนดทีละตัว เพียงแต่กำหนดนโยบายผ่าน OU เพียงตัวเดียวเช่นกัน นอกจากนี้ยังสามารถสร้าง OU ย่อยซ้อน OU หลักได้อีกด้วย OU ที่ถูกสร้างมาแบบ Builtin กับระบบคือ Domain Controllers ซึ่งจะเก็บรายชื่อเครื่องโดเมนคอนโทรลเลอร์ทุกตัวที่อยู่ในโดเมนแล้วยังสามารถบริหารจัดการได้เช่นกัน โดยจะเห็นว่าคอนเทนเนอร์ Users ยังเก็บรายชื่อ Builtin Global Group ที่ถูกสร้างขึ้นอัตโนมัติพร้อมกับการสร้างโดเมนคอนโทรลเลอร์

3) Domain Admins

จะใช้เก็บรายชื่อยูสเซอร์ที่เป็นสมาชิกในกรุ๊ป Admin ซึ่งมีหน้าที่บริหารระบบเอาไว้

4) Domain Computers

จะใช้เก็บรายชื่อเครื่องคอมพิวเตอร์ทั้งหมดบนโดเมนเอาไว้ และ ถ้ามีการสร้างรายชื่อ บนเครื่องคอมพิวเตอร์ขึ้นมาใหม่ รายชื่อเหล่านี้จะถูกเก็บไว้ใน Domain Computers เช่นกัน

5) Domain Controllers

จะใช้เก็บรายชื่อเครื่องโดเมนคอนโทรลเลอร์ทั้งหมดบนโดเมนเอาไว้

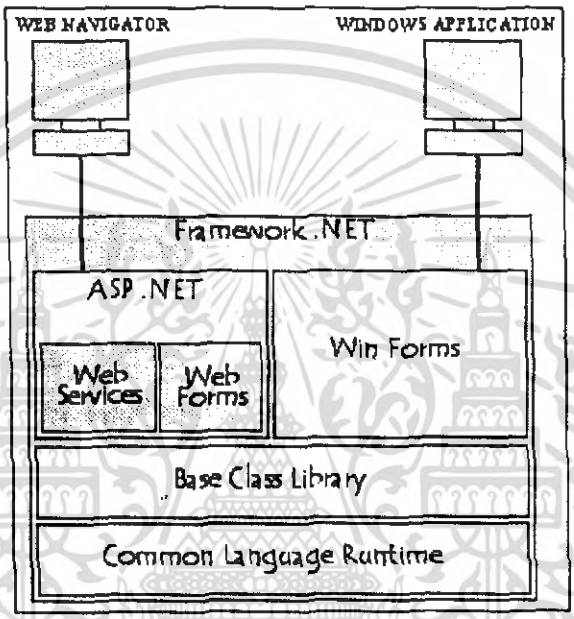
6) Domain Users

จะใช้เก็บรายชื่อยูสเซอร์ทุกคนบนโดเมนเอาไว้ และ ถ้ามีการสร้างรายชื่อยูสเซอร์ขึ้นมาใหม่ รายชื่อเหล่านี้จะถูกเก็บไว้ใน Domain Users เช่นกัน

2.6 การโปรแกรมเพื่อควบคุมการทำงานบนระบบปฏิบัติการวินโดวส์

2.6.1 ความรู้เกี่ยวกับไมโครซอฟต์ดอทเน็ตเฟรมเวิร์ค

ไมโครซอฟต์ดอทเน็ตเฟรมเวิร์คเป็นส่วนหนึ่งของไมโครซอฟต์ดอทเน็ตแพลตฟอร์ม ซึ่งเป็นแพลตฟอร์มของบริษัทไมโครซอฟต์ที่มีแนวคิด ที่ให้ทุกโปรแกรมสามารถที่จะทำงานร่วมกันและติดต่อสื่อสารระหว่างกันได้อย่างมีประสิทธิภาพ โดยใช้รูปแบบฐานข้อมูลมาตรฐานใหม่ที่เรียกว่า XML ในการสื่อสารระหว่างกัน โดยไมโครซอฟต์ดอทเน็ตเฟรมเวิร์คนั้นเป็นรากฐานสำคัญสำหรับไมโครซอฟต์ดอทเน็ตแพลตฟอร์ม โดยไมโครซอฟต์ดอทเน็ตเฟรมเวิร์คนั้นมีส่วนประกอบต่างๆซึ่งแสดงภาพดังนี้



รูปที่ 2.11 รูปแสดงโครงสร้างไมโครซอฟต์ดอทเน็ตเฟรมเวิร์ค

2.6.1.1 ส่วนประกอบของไมโครซอฟต์ดอทเน็ตแพลตฟอร์ม

1) Common Language Runtime (CLR)

เป็นส่วนที่ทำหน้าที่หลักในการรันโปรแกรมต่างๆ ที่ถูกสร้างขึ้นบนดอทเน็ตซึ่งประกอบไปด้วยหน้าที่ย่อยต่างๆ เช่น การควบคุมหน่วยความจำของระบบ การควบคุมการทำงาน ของโปรแกรมที่รันอยู่ในดอทเน็ต การควบคุมการทำงานระยะไกล และการควบคุมดูแลความปลอดภัย

โดยโปรแกรมที่ทำงานภายใต้ การทำงานของ CLR เราจะเรียกว่า managed code ในขณะที่โปรแกรมที่ทำงานอยู่นอกเหนือการทำงานของ CLR เราจะเรียกว่า unmanaged code

2) ดอทเน็ตเบสคลาสไลบรารี

ดอทเน็ตเบสคลาสไลบรารีเป็นคลาสไลบรารีพื้นฐาน ที่ช่วยให้เราสร้างแอพพลิเคชั่นบนแพลตฟอร์มดอทเน็ตได้สะดวกขึ้น โดยที่เบสคลาสไลบรารีนี้จะเป็นคลาสที่ทำการซ่อนรายละเอียดที่เป็นคำสั่งพื้นฐานของวินโดวส์ หรือที่เรียกว่า Windows API เอาไว้ ซึ่งเป็นการเตรียมให้นักพัฒนาสามารถนำคลาสนั้นไปสร้างโปรแกรมที่ทำงานบนวินโดวส์ต่อไป

3) รันไทม์โฮส

รันไทม์โฮส หมายถึง โปรแกรมที่พัฒนาขึ้นมาเพื่อรองรับไมโครซอฟต์ดอทเน็ตเฟรมเวิร์คให้สามารถนำไปใช้งานที่ไหนก็ได้ ซึ่งรันไทม์โฮสนี้ ได้แก่ พวกคอนโซลแอพพลิเคชั่น วินโดวส์แอพพลิเคชั่น และ ASP.NET เป็นต้น โดย ASP.NET นี้เป็นโครงสร้างในการพัฒนาโปรแกรมบนเว็บหรือที่เรียกว่า เว็บแอพพลิเคชั่น โดยจะมีส่วนของคอนโทรลต่างๆที่ใช้ในการพัฒนาเช่น กล่องข้อความ เลเบลข้อความ เมนู และโครงสร้างพื้นฐานในการพัฒนา ASP.NET เว็บเซอร์วิสจะอยู่ในส่วนนี้

2.6.1.2 **ส่วนประกอบของ Common Language Runtime (CLR)**

จากข้างต้นเราทราบว่า CLR เป็นส่วนที่ทำหน้าหลักในการรันโปรแกรมต่างๆ ที่ถูกสร้างขึ้นบนดอทเน็ตซึ่งในการทำหน้าที่นั้นได้เกิดมาจากส่วนประกอบต่างๆ ที่ทำงานร่วมกัน ดังนี้

1) Common Type System (CTS)

ส่วนนี้เป็นการกำหนดมาตรฐานของข้อมูลให้เป็นรูปแบบเดียวกัน ซึ่งทุกภาษาที่รองรับ CLR จะต้องยึดถือปฏิบัติตาม เพื่อเป็นการช่วยให้ภาษาที่มีความแตกต่างกัน สามารถที่จะติดต่อสื่อสารกันได้อย่างรู้เรื่องสอดคล้อง เช่น ถ้าหากมาตรฐานของ CTS กำหนดว่าข้อมูลแบบจำนวนเต็มมีขนาด 4 ไบต์และสามารถที่จะติดลบได้ ทุกภาษาในดอทเน็ต ก็ต้องกำหนดให้ข้อมูลแบบจำนวนเต็มมีมาตรฐานตามนั้น ซึ่งเป็นการช่วยให้มีการสอดคล้องทางด้านรูปแบบของข้อมูลระหว่างภาษาที่รองรับ CLR

2) Common Language Specification (CLS)

ส่วนนี้เป็นการกำหนดมาตรฐานเช่นกัน ซึ่งคล้ายกับ CTS แต่ CLS นั้นเป็นการกำหนดมาตรฐานให้กับภาษาต่างๆใน CLR ว่าต้องมีคุณสมบัติต่างๆ ที่เหมือนกัน เช่น โครงสร้างภาษา การตรวจสอบความผิดพลาดของภาษา คุณสมบัติทางด้านออบเจค เป็นต้น เพื่อช่วยให้ภาษาต่างๆ สามารถที่จะทำงานข้ามภาษาระหว่างกันได้

3) Common Intermediate Language (CIL)

โปรแกรมที่เขียนด้วยภาษาใดก็ตามในไมโครซอฟต์ดอทเน็ตเฟรมเวิร์คนั้น ก่อนนำไปรันจะต้องถูกแปลงเป็นภาษามาตรฐานของดอทเน็ตที่เรียกว่า CIL เสียก่อน จึงจะสามารถนำไปใช้งานกับฮาร์ดแวร์หรือซอฟต์แวร์ใดๆ ก็ได้ที่รองรับบนไมโครซอฟต์ดอทเน็ตเฟรมเวิร์คซึ่ง CIL บางครั้งจะถูกเรียกว่า MSIL ซึ่งย่อมาจาก Microsoft Intermediate Language

4) Just-In-Time Complier (JIT)

เมื่อเราเรียกโปรแกรมที่ทำงานบนดอทเน็ตเฟรมเวิร์ค ขึ้นมาทำงานนั้นจะต้องมีการทำการแปลงภาษา CIL ให้เป็นภาษาเครื่องที่ตรงกับฮาร์ดแวร์หรือซอฟต์แวร์ของเครื่องนั้นๆ เสียก่อน จึงจะสามารถทำงานได้ ซึ่ง JIT เป็นผู้ให้การช่วยเหลือทำงานในส่วนนี้ โดยในส่วนกระบวนการทำงานนี้จะเกิดขึ้นโดยอัตโนมัติ และทำให้การรันโปรแกรมในครั้งแรกจะเสียเวลายาวนานกว่าปกติ แต่หลังจากนั้นในการรันครั้งต่อไปจะไม่มีการทำการ JIT ซ้ำอีก ซึ่งจะช่วยให้โปรแกรมทำงานเร็วเป็นปกติ

5) Virtual Execute System (VES)

หลังจากที่กระบวนการ JIT ทำงานจบลงเรียบร้อยแล้ว เมื่อเราทำการรัน โปรแกรมจะเริ่มทำงานภายใต้การดูแลของ Virtual Execute System นี้ ซึ่งมันจะคอยตรวจสอบการทำงานของโปรแกรมและให้การช่วยเหลือเมื่อโปรแกรมมีปัญหา

2.6.1.3 ประโยชน์ของ Common Language Runtime (CLR)

1) จัดการหน่วยความจำโดยอัตโนมัติ

จากที่เราทราบว่า CLR นั้นช่วยในการจัดการหน่วยความจำต่างๆ ให้โดยอัตโนมัติ โดยเฉพาะการจัดการหน่วยความจำของโปรแกรม ซึ่งมีประโยชน์มาก โดย CLR นั้นได้มีการนำคุณสมบัติ Garbage Collection มาใช้งานแล้ว ซึ่งทำให้เราสามารถที่จะทำลายออบเจคที่ไม่ใช้งานแล้วโดยอัตโนมัติเพื่อป้องกันหน่วยความจำรั่ว (Memory Leakage) ซึ่งเกิดจากการที่เราประกาศตัวแปรเอาไว้ขึ้นมาใช้งาน แต่หลังจากการที่เราใช้งานเสร็จสิ้นลงคือไม่มีการใช้งานออบเจคตัวนั้นอีกต่อไปแล้ว และเราลืมกำหนดค่าให้ค่าตัวแปรนั้นให้เป็นค่าว่าง (nothing หรือ null) ซึ่งทำให้เสียพื้นที่ของหน่วยความจำโดยไม่จำเป็น

2) สนับสนุนการสร้างโปรแกรมหลายภาษา

CLR นั้นได้ถูกออกแบบขึ้นมาเพื่อรองรับและสนับสนุนภาษาหลายๆภาษา ทำให้การพัฒนาโปรแกรมด้วยภาษาที่สนับสนุนดอทเน็ต นั้นเป็นไปโดยง่ายดาย เนื่องจากระบบ Common Type System และ Common Intermediate Language ที่ทำให้ทุกๆภาษามีการกำหนดรูปแบบของข้อมูลและโครงสร้างภาษาเป็นไปในรูปแบบหรือทิศทางเดียวกัน ทำให้ CLR อนุญาตให้ภาษาต่างๆ บนดอทเน็ต สามารถทำงานร่วมกันได้อย่างที่ไม่เคยเกิดขึ้นมาก่อน

สำหรับการสร้างแอพพลิเคชั่นด้วยภาษาหลายๆภาษาในไมโครซอฟต์ดอทเน็ตเฟรมเวิร์ค นั้นทำได้ไม่ยากนัก ตัวอย่างเช่น การสืบทอดคลาสระหว่างกัน โดยเราแทบไม่ต้องมีขั้นตอนที่ซับซ้อนเลย เพราะคลาสที่เราสร้างด้วยภาษาVB.NET อาจสืบถอดมาจากคลาสที่เขียนในภาษา C++ หรือ COBAL ก็ย่อมได้ ซึ่งทำให้เกิดข้อดีในการทำงานเป็นทีมก็คือ เราสามารถที่จะมีทีมโปรแกรมเมอร์ที่ชำนาญในภาษาที่แตกต่างกันสามารถทำงานอยู่ทีมเดียวกันได้ โดยที่ไม่ต้องทำการให้ผู้ที่เก่งภาษาใดภาษาหนึ่งต้องไปเรียนรู้

เพิ่มอีกภาษาหนึ่งเพื่อทำงานร่วมกัน ซึ่งถือเป็นข้อดีที่เกิดจากการ Cross-Language Inheritance นั่นเอง ซึ่งไม่เคยมีเฟรมเวิร์คใดที่สามารถทำได้เช่นเดียวกับไมโครซอฟต์ดอทเน็ตเฟรมเวิร์คนี้

3) การส่งมอบโปรแกรมที่ง่ายและปลอดภัยยิ่งขึ้น

ในอดีตนั้นเมื่อเราจะทำแอพพลิเคชั่นที่ใช้ COM คอมโพเนนต์ ซึ่งเป็นคอมโพเนนต์ของไมโครซอฟต์นั้นไปใช้งาน จำเป็นที่จะต้องมีการทำการลงทะเบียนลงในวินโดวส์รีจิสตรีเสียก่อน เพื่อทำให้ระบบปฏิบัติการรู้จักคอมโพเนนต์นั้นๆ ก่อนนำไปใช้งาน แต่ในดอทเน็ตเฟรมเวิร์คนั้นเราไม่จำเป็นที่จะต้องทำเช่นนั้น เนื่องจากแอพพลิเคชั่นที่สร้างบน ดอทเน็ตเฟรมเวิร์คสามารถที่จะติดตั้งด้วยการคัดลอกไฟล์ที่จำเป็นลงในเครื่อง ก็สามารถนำไปใช้งานได้ทันที

2.6.1.4 ความรู้เกี่ยวกับคลาสพื้นฐานของดอทเน็ต

ไมโครซอฟต์ดอทเน็ตเฟรมเวิร์คนั้นได้มีการจัดเตรียมบริการและคลาสต่างๆ ที่จำเป็น เช่น คลาสที่ใช้ในการจัดการข้อมูล การอ่านและเขียนไฟล์ คลาสด้านความปลอดภัย และอื่นๆ ซึ่งเป็นคลาสพื้นฐานที่จำเป็นในการสร้างแอพพลิเคชั่น นอกจากนี้ยังรวมไปถึง ADO ที่เป็นส่วนที่ใช้ในการติดต่อฐานข้อมูลที่ได้ถูกพัฒนาเป็น ADO.NET ก็เข้าไปร่วมเป็นส่วนหนึ่งของดอทเน็ตเฟรมเวิร์คเช่นกัน อีกทั้งยังมีการรวมโมดูลที่ช่วยให้เราสามารถที่จะติดต่อและทำงานร่วมกับ XML ได้ดียิ่งขึ้น

นอกจากนี้ในส่วนของฟังก์ชั่นในการทำงานเฉพาะบางอย่างที่เคยเป็นส่วนหนึ่งสำหรับบางภาษา ก็ได้เข้าไปรวมเป็นส่วนหนึ่งในคลาสของดอทเน็ต เช่น ฟังก์ชั่น sqrt ซึ่งเป็นฟังก์ชั่นที่ใช้ในการทำการหารากที่สองซึ่งมีในภาษา Visual Basic ก็ได้ถูกนำมาเป็นส่วนหนึ่งของดอทเน็ตในคลาสที่ชื่อว่า System.Math เป็นฟังก์ชั่น System.Math.Sqrt ซึ่งทุกๆภาษาที่สนับสนุนดอทเน็ตนั้น ก็จะเข้าถึงคลาสไลบรารีต่างๆ เหล่านั้นได้ โดยอัตโนมัติ เช่น ฟังก์ชั่น System.Math.Sqrt ก็สามารถที่จะใช้ฟังก์ชั่นเดียวกันนี้ได้กับทุกๆ ภาษา

โดยการจัดการคลาสของไมโครซอฟต์ดอทเน็ตเฟรมเวิร์คนั้นจะมีการทำการจัดกลุ่มของคลาสที่มีการทำงานที่คล้ายคลึงกันมาจัดเป็นกลุ่มเดียวกัน โดยจะเรียกกลุ่มเหล่านั้นว่า เนมสเปซ (Namespace) ซึ่งการทำงานพื้นฐานในดอทเน็ตเฟรมเวิร์คนั้นจะอยู่ในเนมสเปซที่ชื่อว่า System ตัวอย่างที่พบได้ทั่วไป เช่น

Namespace	คำอธิบาย
System.Data	เป็นคลาสที่ทำงานเกี่ยวกับการจัดการข้อมูลพื้นฐานเช่น DataSet, DataTable
System.Diagnostics	เป็นคลาสที่ทำงานเกี่ยวกับการดีบักและตรวจสอบข้อผิดพลาดของแอพพลิเคชั่น
System.IO	เป็นคลาสที่ทำงานเกี่ยวกับการอ่านและเขียนไฟล์ เช่น File, FileStream เป็นต้น
System.Math	เป็นคลาสที่ทำงานเกี่ยวกับการคำนวณทางคณิตศาสตร์

รูปที่ 2.12 รูปภาพแสดงตารางคลาสพื้นฐานของดอทเน็ต

2.6.1.5 การสร้างส่วนติดต่อกับผู้ใช้ในดอทเน็ต

ในการสร้างส่วนติดต่อกับผู้ใช้ในไมโครซอฟต์ดอทเน็ตเฟรมเวิร์คนั้นสามารถทำได้อย่างง่ายดาย เนื่องจากดอทเน็ตได้มีการเตรียมคลาสที่ใช้ในการจัดการส่วนติดต่อกับผู้ใช้ซึ่งคลาสที่พบเห็นได้บ่อยมี 2 ลักษณะ คือ

1) วินโดวส์ฟอร์มจากคลาส WinForms

เป็นคลาสที่ช่วยใช้ในการสร้างส่วนของแอพพลิเคชั่นบนวินโดวส์ โดยทุกภาษาในดอทเน็ตจะใช้คลาส WinForms นี้ในการสร้างส่วนติดต่อกับผู้ใช้ ซึ่งภายในคลาส WinForms จะประกอบด้วยคอนโทรลต่างๆ มากมายที่ใช้ในการสร้างโปรแกรมบนวินโดวส์รวมทั้งยังมีฟังก์ชั่นที่ใช้ในการวาดรูป ซึ่งคลาส WinForms นี้เป็นส่วนหนึ่งของดอทเน็ตในเนมสเปซที่ชื่อว่า System.WinForms

2) เว็บฟอร์มจากคลาส WebForms

เป็นคลาสที่เป็นส่วนหนึ่งของ ASP.NET ซึ่งเป็นคลาสที่ใช้ในการสร้างเว็บฟอร์มที่ใช้ในการพัฒนาเว็บแอพพลิเคชั่นที่ทำงานบนเว็บ ซึ่งจะมีลักษณะการใช้งานเหมือนกับการสร้างฟอร์มบนวินโดวส์ โดยเราสามารถที่จะนำคอนโทรลมาวางและเขียนโปรแกรมตอบสนองได้เหมือนกับวินโดวส์ฟอร์ม

2.6.2 ความรู้เกี่ยวกับการทำงานของ Win32 API

2.6.2.1 ความหมายเบื้องต้นของ Win32 API

Win32 API คือ ชื่อของ API (Application Programming Interface) ที่รวบรวมฟังก์ชันที่ใช้สำหรับพัฒนาแอพพลิเคชั่นบนวินโดวส์ทั้งหมดเอาไว้ หากพัฒนาแอพพลิเคชั่นโดยติดต่อกับ Win32 API โดยตรงจะสามารถควบคุมการทำงานของแอพพลิเคชั่นได้ทุกอย่างตามต้องการ

อย่างไรก็ตามหากต้องการที่จะพัฒนาระบบงานใหญ่ๆ ให้สำเร็จทันเวลาในภาวะที่มีการแข่งขันกันสูงในปัจจุบันคงไม่เหมาะสมหากจะพัฒนาโดยใช้ Win32 API ระดับล่างล้วนๆ เพราะแทนที่จะเสร็จทันเวลาอาจเกิดความล้มเหลวในที่สุด จึงมีการนำเอาเฟรมเวิร์คเข้ามาช่วยเหลือในการพัฒนา และเรียกใช้ Win32 API เมื่อจำเป็นเท่านั้น

2.6.2.2 หลักการการแฮนเดิล (Handle)

ฟังก์ชัน API จะใช้แฮนเดิลในการอ้างถึงออปเจคต่างๆ ที่กำลังเปิดใช้งานอยู่แทนการใช้ชื่อของออปเจค โดยแฮนเดิลเปรียบเสมือน ID ที่อ้างอิงไปยังออปเจคต่างๆ ที่กำลังทำงานอยู่ในระบบปฏิบัติการวินโดวส์ซึ่ง ID ดังกล่าวจะเป็นตัวเลขแบบ 32 บิต บนระบบปฏิบัติการแบบ 32 บิต และแฮนเดิลจะถูกกำหนดให้กับออปเจคใดๆ โดยอัตโนมัติ เมื่อออปเจคนั้นๆ ถูกสร้างขึ้น โดยไม่ตายตัวในแต่ละออปเจค นอกจากนี้ค่าของแฮนเดิลของออปเจคที่กำลังทำงานอยู่จะไม่มีวันซ้ำกัน

2.6.2.3 หลักการระบบเมสเสจในระบบปฏิบัติการ Win32

แอพพลิเคชั่นบน Win32 จะใช้หลักการของ Event-Driven ในการรับอินพุต ภาพรวม คือ แอพพลิเคชั่นจะรอจนกว่าจะมีการส่งอินพุตมาให้ ซึ่งอินพุตเหล่านั้นจะถูกส่งไปยังวินโดวส์ต่างๆ ภายในแอพพลิเคชั่นอีกทีหนึ่ง

ทุกวินโดวส์จะมีฟังก์ชันควบคุมการทำงานของมันเรียกว่า วินโดวส์โพรซีเยอร์ (Window Procedure) ฟังก์ชันนี้มีไว้เพื่อตอบสนองต่ออินพุตต่างๆ ที่ถูกส่งมาโดยระบบปฏิบัติการจะตรวจสอบก่อนว่าอินพุตนั้นเกิดที่วินโดวส์ตัวใด และวินโดวส์นั้นเป็นของเธรดใด แล้วจึงส่งอินพุตที่เกิดขึ้นไปยังเธรดที่เป็นเจ้าของวินโดวส์ตัวนั้น เพื่อให้เธรดนั้นจัดส่งอินพุตไปยังฟังก์ชันควบคุมการทำงานของวินโดวส์ตัวนั้นอีกทีหนึ่ง และฟังก์ชันควบคุมการทำงานของวินโดวส์ตัวนั้นจะดำเนินการตามกระบวนการต่างๆ เป็นการตอบสนอง

กระบวนการดังกล่าวนั้นจะเกิดขึ้น ก็ต่อเมื่อมีอินพุตประเภทการกดคีย์บอร์ด หรือการขยับเมาส์เกิดขึ้น ซึ่งจริงๆ แล้ววินโดวส์ไม่ได้ส่งอินพุตไปยังเธรดโดยตรง แต่จะสร้างเมสเสจ ขึ้นมาจากอินพุตเหล่านั้นก่อน แล้วจึงส่งเมสเสจไปแทน โดยเมสเสจเป็นเพียงค่าๆ หนึ่งที่ส่งไปยังวินโดวส์พร้อมๆ กับรายละเอียดอื่นๆ ซึ่งฟังก์ชันควบคุมการทำงานของวินโดวส์จะตีความค่านั้นรวมถึงรายละเอียดที่ส่งมาเพื่อตัดสินใจว่าจะเข้าไปดำเนินการในกระบวนการใด

2.6.2.4 หลักการกระบวนการส่งเมสเสจในระบบปฏิบัติการ Win32

แอพพลิเคชั่นบนระบบปฏิบัติการ Win32 อาจประกอบไปด้วย กระบวนการตั้งแต่ 1 กระบวนการขึ้นไปซึ่งแต่ละกระบวนการ อาจประกอบไปด้วยเธรดตั้งแต่ 1 ขึ้นไปเช่นกัน ขึ้นอยู่กับลักษณะการทำงาน และความซับซ้อนภายในตัวมัน

2.6.2.5 หลักการกระบวนการ (Process)

กระบวนการเป็นหน่วยหนึ่งของระบบ ระบบประกอบไปด้วยกระบวนการต่างๆ เช่น กระบวนการของระบบปฏิบัติการทำงานโปรแกรมของระบบ กระบวนการของผู้ใช้ ทำงานโปรแกรมผู้ใช้ กระบวนการเหล่านี้จะทำงานประสานกันไป (Concurrent) โดยระบบปฏิบัติการ จัดสรรหน่วยประมวลผลกลางสลับให้ แต่ละกระบวนการทำงาน ซึ่งจะทำให้ระบบคอมพิวเตอร์มีประสิทธิผลสูงขึ้น

เราอาจเรียกโปรแกรมที่กำลังทำงานอยู่ว่าเป็น กระบวนการ การทำงานของกระบวนการต้องเป็นแบบลำดับ หรืออีกนัยหนึ่ง ณ เวลาใดๆ จะมีเพียงอย่างมากหนึ่งคำสั่งที่กำลังดำเนินการอยู่ในนามของกระบวนการนี้ โดยกระบวนการไม่ได้หมายความเพียงโปรแกรม และการทำงานเท่านั้น แต่รวมถึงพวกรีจิสเตอร์ของซีพียู ,ข้อมูลชั่วคราว (Stack), เก็บตัวแปรแบบโกลบอล(Data Section)

แม้ว่าอาจมีกระบวนการ 2 กระบวนการ ทำงานบนโปรแกรมเดียวกัน ก็ยังต้องนับว่าเป็นการทำงานตามลำดับแยกกัน เป็นเรื่องปกติที่กระบวนการหลัก จะสร้างกระบวนการย่อยหลายๆ กระบวนการในขณะทำงาน

2.6.2.6 หลักการทำงานแบบเธรด (Thread)

ในระบบปฏิบัติการแบบมัลติเธรดจะสามารถแบ่งโปรเซสออกเป็นหน่วยที่เล็กกว่าซึ่งเรียกว่าเธรด หรือบางทีเรียกว่า Light-Weight Process ระบบปฏิบัติการแบบนี้จะสามารถสร้างหลายเธรดในหนึ่งโปรเซส โดยปกติเธรดหนึ่ง คือ Execution Flow ของงานอย่างหนึ่ง หากนำหลายๆ เธรดมารวมกันในโปรเซสหนึ่งจะช่วยให้โปรเซสนั้น สามารถทำงานได้มากกว่าหนึ่งอย่างไปพร้อมๆ กันโดยไม่ต้องมีการทำ Context Switching โปรแกรมจึงทำงานได้เร็ว และเป็นการใช้งานหน่วยประมวลผลได้อย่างมีประสิทธิภาพมากขึ้น

ในระบบโดยปกติหนึ่งโปรเซสจะมีหนึ่ง PCB ส่วนในระบบ Multitasking หนึ่งโปรเซสมีหนึ่ง PCB เช่นกัน แต่อาจมีหลายเธรด โดยที่เธรดเหล่านั้นมีส่วนที่เป็นโปรแกรม (Code Segment) ร่วมกัน และใช้หน่วยความจำสำหรับเก็บข้อมูลบางส่วนร่วมกัน

แต่ละเธรดจะมีโปรแกรมของตัวเองซึ่งมีจุดเริ่มต้น และจุดสิ้นสุดแตกต่างกันไป เธรดแต่ละตัวมักจะทำงานเป็นอิสระต่อกันทำให้เธรดหลายๆ ตัวสามารถ ทำงานพร้อมกันได้ ถ้าเครื่องคอมพิวเตอร์นั้นมีหน่วยประมวลผลหลายตัวก็อาจแบ่งให้หน่วยประมวลผลแต่ละตัวทำงาน แต่ละเธรดได้พร้อมกันในรูปแบบเส้นขนาน ซึ่งจะทำให้โปรแกรมทำงานเร็วขึ้น

2.7 ลักษณะการทำงานของแอดแวร์และสปายแวร์

2.7.1 การทำงานของแอคทีฟคอนเทนต์

เทคโนโลยีของเว็บได้มีการพัฒนาปรับปรุงมาในทิศทางที่เปลี่ยนจากแพสสีฟคอนเทนต์ ให้เป็นเนื้อหาที่เปลี่ยนแปลงได้ตลอด เพื่อตอบสนองความต้องการของผู้ใช้เซิร์ฟเวอร์จึงทำหน้าที่เปลี่ยนไปจากการจัดเตรียม และอำนวยความสะดวกในการอ่านไฟล์ เป็นการประมวลผลคำสั่งและสร้างเว็บเพจที่สามารถตอบสนองต่อผู้ใช้ได้อย่างทันทีทันใด ในฝั่งของไคลเอนต์เองก็มีการพัฒนาเพิ่มมากขึ้น เพื่อให้ผู้ใช้สามารถขยายขอบเขตการทำงานได้มากกว่าการเรียกดูเว็บเพจซึ่งเสมือนการสื่อสารทางเดียวให้เป็นการสื่อสารแบบสองทาง ซึ่งผู้ใช้สามารถส่งข้อมูลกลับไปยังเว็บเซิร์ฟเวอร์ได้ด้วย

ในส่วนของภาษา HTML เดิมนั้นมีช่องทางให้ผู้ใช้ส่งข้อมูลกลับไปได้บ้างแต่อยู่ในขอบเขตที่จำกัดมาก และไม่เพียงพอต่อการใช้งานที่มีความสลับซับซ้อน ดังนั้นจึงมีการพัฒนาภาษาที่ทำงานได้บนไคลเอนต์ เพื่อให้ทำได้มากกว่าการแปลภาษา HTMLอย่างเดียว คือ เพิ่มความสามารถที่จะประมวลผล

ตามคำสั่งได้ด้วย เช่น ActiveX ,JavaScript ซึ่งเทคโนโลยีประเภทนี้จะเป็นการขยายขอบเขตของการใช้งานเว็บบราวเซอร์ และเว็บเซิร์ฟเวอร์ให้ทำงานได้กว้างขวางยิ่งขึ้น

โดยหลักการพื้นฐานภาษาเหล่านี้มีวัตถุประสงค์เหมือนกัน คือ เซิร์ฟเวอร์จะส่งชุดคำสั่งที่เตรียมไว้แล้วสำหรับหน้าเว็บเพจนั้นมายังบราวเซอร์พร้อมกับ HTML โดยที่ HTML จะเป็นเสมือนยูสเซอร์อินเตอร์เฟสของแอพพลิเคชั่น และคำสั่งซึ่งส่งมาด้วยจะเป็นเสมือนส่วนประมวลผลที่เซิร์ฟเวอร์ต้องการให้ไคลเอนต์ทำงาน คำสั่งซึ่งส่งมา และทำให้บราวเซอร์สามารถทำงานได้นั้นเรียกว่า แอคทีฟคอนเทนต์ (ActiveContent) บราวเซอร์ที่จะสามารถทำงานได้ก็จะต้องมีตัวแปลภาษาอยู่ด้วยเพื่อจะได้ทำงานตามที่เซิร์ฟเวอร์สั่งได้

โดยหากพิจารณาแล้ว การที่มีโปรแกรมมาทำงานที่ไคลเอนต์ก็น่าจะเป็นสิ่งที่ช่วยให้การทำงานของบราวเซอร์ และเซิร์ฟเวอร์ก้าวหน้าไปอีกระดับหนึ่งสามารถรองรับเว็บแอพพลิเคชั่นได้เป็นอย่างดี อย่างไรก็ตามปัญหาของแอคทีฟคอนเทนต์ที่ควรคำนึงถึงมี ก็คือ

ขอบเขตของการทำงานของแอคทีฟคอนเทนต์ มีขอบเขตการทำงานที่จำกัดเฉพาะเรื่องเกี่ยวกับการโต้ตอบกับผู้ใช้หรือยูสเซอร์อินเตอร์เฟส การคำนวณ บวก ลบ คูณ หาร และการตรวจสอบเงื่อนไขทั่วไปเท่านั้น ไม่ควรที่จะสามารถสั่งงานให้ทำงานออกไปนอกขอบเขตของบราวเซอร์ได้มากนัก เช่น เรียกคำสั่งของระบบ (System Call) ซึ่งผู้ผลิตก็ได้คำนึงในส่วนนี้ และพยายามป้องกันไว้แล้วแต่ก็ทำได้ไม่ดีนัก จึงมักพบว่าแอคทีฟคอนเทนต์เหล่านั้นสามารถทำงานได้หลายอย่างบนเครื่องคอมพิวเตอร์ที่บราวเซอร์ทำงานอยู่ เช่น แอบอ่านข้อมูลของผู้ใช้ส่งกลับไปยังเว็บเซิร์ฟเวอร์โดยที่ผู้ใช้ไม่รู้ตัว

ความน่าเชื่อถือของเว็บไซต์ การตั้งเว็บไซต์ไม่ได้เป็นเรื่องยากเย็นแสนเข็ญแต่ประการใดมีเงินเพียงเล็กน้อย ก็สามารถเปิดเว็บไซต์ด้วยชื่อโดเมนของตัวเองได้ หรือหากไม่ต้องการเสียเงินก็ฝากชื่อไว้กับโดเมนคนอื่น ซึ่งก็มีผู้ให้บริการอยู่มากมาย ด้วยความง่ายดังนี้ การเปิดเว็บไซต์ใหม่ และการปิดตัวของเว็บไซต์เก่าจึงเกิดขึ้นตลอดเวลาอาจพูดได้ว่าบนอินเตอร์เน็ตนั้นเป็นสังคมที่วุ่นวายที่สุด และไม่รู้ว่าใครเป็นใครทุกอย่างสามารถปลอมแปลง หรือหลอกลวงได้โดยง่าย

2.7.2 การทำงานของไลบรารี่ DLL

ไลบรารี่ DLL หรือ Dynamic Link Libraries เป็นวิธีการรวบรวมโค้ดของโปรแกรมที่ออกแบบมาให้ใช้ร่วมกันไว้ในไฟล์ที่มีลักษณะเหมือนกับไฟล์ .EXE แต่มีไว้สำหรับให้แอพพลิเคชั่นต่างๆ เรียกใช้ เราสามารถสร้าง DLL สำหรับทำงานเฉพาะอย่างไว้ใช้กับแอพพลิเคชั่นที่พัฒนาด้วยภาษาใดๆก็ได้

ไลบรารี่ DLL เป็นไลบรารีที่ประกอบด้วยฟังก์ชันหรือโพรซีเยอร์ ซึ่งสามารถนำไปใช้ร่วมกับแอพพลิเคชั่นต่างๆ ได้ ประโยชน์ที่ได้จากการใช้ DLL มีดังนี้

- เป็นไฟล์ที่ทำงานแยกต่างหากจากแอพพลิเคชั่น ทำให้การแก้ไขไลบรารีทำได้ง่าย โดยที่ไม่ต้องคอมไพล์แอพพลิเคชั่นใหม่

- เราสามารถแบ่งโปรแกรมใหญ่ๆ ออกเป็นแอพพลิเคชั่นเล็กๆ โดยใช้สิ่งที่ใช้ร่วมกันมาเขียนเป็นฟังก์ชัน หรือโพรซีเยอร์ไว้ใน DLL เพื่อเรียกได้จากทุกแอพพลิเคชั่น
- เพิ่มประสิทธิภาพในการใช้เนื้อที่หน่วยความจำ เพราะแอพพลิเคชั่นสามารถเรียก DLL ขึ้นมาใช้และเมื่อทำงานเสร็จแล้วก็เลิกสั่งได้ทุกเมื่อ ทำให้สามารถใช้เนื้อที่ในหน่วยความจำ และทรัพยากรของระบบอย่างมีประสิทธิภาพ นอกจากนี้แล้วถ้าในขณะหนึ่งมีแอพพลิเคชั่นหลายตัวใช้ DLL เดียวกันก็สามารถใช้ DLL ที่ถูกเรียกใช้อยู่ในหน่วยความจำแล้วได้โดยไม่ต้องโหลดขึ้นมาใหม่

2.7.3 การทำงานของคุกกี้

คุกกี้ เป็นข้อมูลขนาดเล็กที่เว็บเซิร์ฟเวอร์ส่งมาให้เว็บบราวเซอร์ เพื่อใช้ในการอ้างอิงระหว่างบราวเซอร์กับเว็บเซิร์ฟเวอร์ได้มีการกำหนดกลไกที่ใช้เพื่อให้บราวเซอร์ และเซิร์ฟเวอร์จดจำซึ่งกันและกันได้ โดยการที่เซิร์ฟเวอร์จะส่งข้อมูลขนาดเล็กชุดหนึ่งให้กับบราวเซอร์เพื่อเป็นเครื่องหมายที่จดจำว่าบราวเซอร์ได้โดยติดต่อกับเว็บเซิร์ฟเวอร์ไว้แล้วในสถานะใด เมื่อเว็บเซิร์ฟเวอร์ต้องการทราบข้อมูลก็ทำการขออ่านคุกกี้ที่บราวเซอร์เก็บไว้ เช่น อย่างน้อยที่สุดก็สามารถบอกได้ว่าเพจก่อนหน้าเพจปัจจุบันนั้นบราวเซอร์ได้ขอเว็บเพจใดไป ทำให้มีการรับรู้สถานะ และความต่อเนื่องของบราวเซอร์ได้ คุกกี้จึงถูกนำมาใช้เป็นสิ่งที่ใช้ควบคุมเซสชั่น และความต่อเนื่องของบราวเซอร์กับเซิร์ฟเวอร์บราวเซอร์เมื่อได้รับคุกกี้ก็จะทำการจัดเก็บเป็นไฟล์ขนาดเล็กไว้บนเครื่องของตนและจะทำการอ่านค่าและส่งไปให้เมื่อเซิร์ฟเวอร์ต้องการ

ต่อมาได้มีการประยุกต์คุกกี้มาใช้งานในการเก็บข้อมูลหลายอย่างของผู้ใช้ที่เกี่ยวข้องกับเว็บเซิร์ฟเวอร์นั้นได้ เช่น เพื่อให้เว็บเซิร์ฟเวอร์ดูเหมือนฉลาด และตอบสนองผู้ใช้ได้ดีก็จะทำการเก็บคุกกี้ที่มีข้อมูลเกี่ยวกับลักษณะของผู้ใช้ไว้เมื่อเปิดเว็บเพจมาดูเซิร์ฟเวอร์ก็สามารถจดจำผู้ใช้ได้จากคุกกี้ และทำการจัดหน้าเว็บเพจให้เหมือนเดิมตามที่ผู้ใช้ชอบ หรือการที่เว็บเซิร์ฟเวอร์ทำการเติมชื่อผู้ใช้ให้อัตโนมัติ เมื่อผู้ใช้เข้ามาดูในเว็บเพจนั่นก็มาจากการที่เว็บเซิร์ฟเวอร์สามารถจำผู้ใช้ได้จากคุกกี้ที่เก็บอยู่นั่นเองคุกกี้เลยได้มีวิวัฒนาการมาใช้สำหรับเก็บข้อมูลอื่นๆ นอกจากข้อมูลของเซสชั่น เช่น ชื่อผู้ใช้ รหัสผ่าน หมายเลขบัตรเครดิต และข้อมูลอื่นๆ อีกมากมายที่ผู้ใช้เคยป้อนเข้าไปยังเว็บเซิร์ฟเวอร์แล้ว จริงอยู่ว่าบราวเซอร์เองก็มีกลไกรักษาความปลอดภัยของคุกกี้เองในระดับหนึ่ง แต่ปัจจัยที่สำคัญก็ขึ้นอยู่กับการกำหนดระดับความปลอดภัยของผู้เขียนโปรแกรมที่ส่งคุกกี้จากเซิร์ฟเวอร์ด้วย หากไม่ได้มีการกำหนดพารามิเตอร์ที่รัดกุมก็จะทำให้เว็บไซต์อื่นที่ไม่ประสงค์ดีมาหลอกอ่านคุกกี้ในเครื่องของผู้ใช้ไปได้

2.8 รูปแบบในการจัดเก็บข้อมูล

2.8.1 ความรู้เบื้องต้นเกี่ยวกับภาษาข้อมูล XML

2.8.1.1 ความหมายเบื้องต้นเกี่ยวกับภาษาข้อมูล XML

XML คือเป็นภาษาที่ถูกออกแบบมาเพื่อรองรับการแลกเปลี่ยนข้อมูลท่ามกลางมาตรฐานที่แตกต่างของภาษาต่างๆในปัจจุบันและรองรับกับการพัฒนาแอพพิเคชั่นบนเว็บ ในยุคต่อไปของไมโครซอฟท์ที่เรียกว่า สถาปัตยกรรม .net ซึ่งจะมีแนวความคิดหลักก็คือ การแลกเปลี่ยนข้อมูลซึ่งกันและกันต่างแพลตฟอร์ม

ภาษา XMLนั้นไม่ได้เน้นที่การแสดงผลหรือควบคุมการทำงาน แต่ภาษาXMLเน้นที่การรองรับการแลกเปลี่ยนข้อมูลระหว่างกันถ้าจะบอกว่าXMLทำหน้าที่เป็นล่ามที่แสนดี ก็ไม่ผิดสักเท่าใด ซึ่งการศึกษาภาษาXMLจึงแตกต่างจากภาษาอื่นๆ ตรงที่ภาษาXMLไม่สามารถทำได้ทุกๆอย่างตามที่ต้องการแต่เราจะใช้ความได้เปรียบของภาษาXMLมาช่วยเสริมหรือเพิ่มประสิทธิภาพในการทำงานโดยเฉพาะอย่างยิ่งสามารถใช้งานร่วมกับภาษาเว็บอื่นๆได้เช่นVB.net ASP.net , HTML , CSS หรือ DHTML , XSL , JavaScript

2.8.1.2 ส่วนประกอบของภาษาข้อมูล XML

"การสื่อสารข้อมูลเป็นคำที่มีความหมายกว้าง และแทบจะเปล่าประโยชน์ในการพยายามหาคำจำกัดความหรือหาข้อกำหนดที่สามารถครอบคลุมได้อย่างทั่วถึง จากเหตุผลข้างต้นทำให้มีข้อกำหนดที่เกี่ยวพันกันจำนวนหนึ่ง (บางอันก็ยังอยู่ในส่วนของการพัฒนาระยะแรกๆ) ซึ่งทำงานควบคู่กันไปในกลุ่มของ XML ซึ่งแต่ละข้อกำหนดจะครอบคลุมในแต่ละด้านของการสื่อสารข้อมูล ต่อไปนี้คือข้อกำหนดที่สำคัญบางเรื่อง

1) XML เวอร์ชั่น 1.0

เป็นข้อกำหนดพื้นฐานซึ่งเกี่ยวข้องกับการสร้างของกลุ่มของเทคโนโลยี XML กล่าวถึงไวยากรณ์ซึ่งเอกสาร XML ต้องยึดถือ และกฎเกณฑ์ที่ XML Parser จำต้องยึดถือ รวมถึงทุกๆ สิ่งที่จำเป็นต่อการอ่านหรือเขียนเอกสาร XML (นอกจากนี้ยังกำหนดในเรื่องของ DTDs ด้วย แม้ในบางครั้งสองเทคโนโลยีนี้จะถูกแยกจากกันก็ตาม)

เนื่องมาจากการที่เราสามารถกำหนดโครงสร้างและชื่อของอิลิเมนต์ ขึ้นมาสำหรับใช้ในเอกสารของเราได้ จึงจำเป็นต้องมี DTD และสกีมา (Schemas) เพื่อช่วยในการจัดสร้างเทมเพลตสำหรับประเภทเอกสารขึ้นมา เพื่อที่จะตรวจสอบได้ว่าเอกสารต่างๆ เป็นไปตามที่กำหนดไว้ในเท็มเพลต และนักพัฒนาคนอื่นๆ จะสามารถสร้างเอกสารที่เข้ากันได้กับเท็มเพลตนั้นๆ

2) เนมสเปซ (Namespace)

ช่วยให้สามารถแยกแยะ XML Vocabulary แต่ละอันออกจากกันได้ ทำให้เราสามารถสร้างเอกสารที่ใช้งานได้กว้างขึ้นด้วยการรวมเอา Vocabulary หลายๆ อันไว้ในประเภทของเอกสารเดียวกัน

3) ภาษากำหนดเส้นทาง (XPath)

ใช้กำหนดรายละเอียดของภาษาที่ใช้ในการสืบค้นข้อมูลบางส่วนของเอกสาร XML ทำให้แอพพลิเคชั่นต่างๆ สามารถค้นหาข้อมูลบางส่วนของเอกสาร XML ที่ต้องการได้ แทนที่จะต้องนำข้อมูลทั้งหมดของเอกสารออกมาใช้งาน ตัวอย่างเช่น XPath สามารถใช้ค้นหา "นามสกุลทั้งหมดที่มีอยู่ในเอกสารได้

ในบางกรณีเราอาจจำเป็นที่จะแสดงผลเอกสาร XML ของเรา สำหรับในกรณีที่ไม่ซับซ้อนเราสามารถใช้ Cascading Style Sheet (CSS) เพื่อแสดงผลเอกสารของเราได้ และถ้าเอกสารนั้นมีความซับซ้อนเราก็สามารถใช้ Extensible Style Sheet Language (XSL) ที่เกิดมาจาก XSLT ซึ่งสามารถแปลงเอกสารของเราจากประเภทหนึ่งไปสู่อีกประเภทหนึ่งได้ รวมถึง Formatting Object ที่ทำงานเกี่ยวกับการแสดงผลเช่นกัน

4) ภาษากำหนดการเชื่อมโยง (XLink และ Xpointer)

เป็นภาษาที่ใช้สำหรับเชื่อมโยงเอกสาร XML เข้าด้วยกัน ในลักษณะที่ใกล้เคียงกับไฮเปอร์ลิงก์

5) Document Object Model (DOM)

เป็นแอพพลิเคชั่นที่มักใช้ในการติดต่อกับเอกสาร XML กันอย่างแพร่หลายรวมถึงทางเลือกอย่างหนึ่งสำหรับโปรแกรมเมอร์ที่จะใช้ติดต่อกับเอกสาร XML จากโค้ดของ XML เอง เรียกว่า Simple API for XML (SAX)

2.8.2 ขั้นตอนการสร้างภาษาข้อมูล XML

ความโดดเด่นของภาษาข้อมูล XML นั้นอยู่ที่ความสามารถในการสร้างเอกสารเพื่ออธิบายหรือให้คำจำกัดความข้อมูลใดๆ ก็แล้วแต่ที่เราต้องการ อีกทั้งยังสามารถปรับเปลี่ยนได้ตามโครงสร้างข้อมูลที่เราออกแบบอีกด้วย แต่ในที่สุดแล้ว เรายังคงต้องการที่จะจัดการกับการออกแบบข้อมูลอย่างละเอียดของเราให้เรียบร้อย เพื่อที่จะกำหนดได้ว่า "หากต้องการที่จะให้ข้อมูลอยู่ในรูปแบบ XML แล้วจะต้องจัดโครงสร้างข้อมูลดังนี้มิใช่ปรับ XML ตามโครงสร้างข้อมูล"

ตัวอย่างเช่น เมื่อเราสร้าง <name> ดังข้างต้น เราได้สร้างข้อมูลแบบโครงสร้างขึ้นมา ซึ่งก็เนื่องมาจากว่าเราไม่ได้เพียงแต่นำเอาข้อมูลทุกๆ อย่างที่เกี่ยวข้องกับชื่อไปรวมกันไว้เฉยๆ แต่ในข้อมูลแบบลำดับชั้นของเรานั้น มีข้อมูลเกี่ยวกับความสัมพันธ์ระหว่างส่วนต่างๆ ของข้อมูลด้วย เช่นภายใต้ <name> จะมี <first> อยู่ด้วย เป็นต้น

นอกเหนือไปจากนั้น เรายังได้สร้างองค์ประกอบเฉพาะขึ้นมาชุดหนึ่ง ซึ่งเรียกว่า คำศัพท์ และคำศัพท์เหล่านั้นนั่นเองที่เราใช้กำหนดอิลิเมนต์ต่างๆ เพื่อใช้สำหรับกำหนดรายละเอียดของชื่อขึ้นมา: คือ <name>, <first>, <middle> และ <last> นั่นเอง

แต่เหนือสิ่งอื่นใดทั้งหมดสิ่งสำคัญที่สุดที่เราจำเป็นต้องสร้างขึ้นมาก็คือ ประเภทของเอกสาร เมื่อเราจะสร้างเอกสารพิเศษขึ้นมา โดยการจัดโครงสร้างแบบพิเศษ เพื่อใช้อธิบายข้อมูลประเภทพิเศษของเราเองแม้ว่าเราจะไม่ได้กำหนดกฎเกณฑ์ของข้อมูลอย่างแน่นอนออกมา ก็ยังคงมีกฎทั่วๆ ไปซึ่งแต่ละอิลิเมนต์ของคำศัพท์ของเราจำเป็นต้องยึดถือเอาไว้ เพื่อที่จะให้เอกสาร <name> เป็นไปตามประเภทของเอกสารของเรา โดยข้อกำหนดการประกาศค่าต่างๆ มีดังนี้

2.8.2.1 การประกาศค่าอิลิเมนต์ในเอกสาร XML

รูปแบบของการประกาศค่าอิลิเมนต์เป็นดังนี้

```
<!ELEMENT ชื่ออิลิเมนต์ (เนื้อหาภายในอิลิเมนต์)>
```

กฎอิลิเมนต์ของ XML

- ทุกๆ แท็กเริ่มต้นจะต้องมีแท็กสิ้นสุดที่เข้าคู่เหมาะสมกัน
- แท็กต่างๆ จะคาบเกี่ยวกันไม่ได้
- เอกสาร XML จะมีรูทอิลิเมนต์ได้เพียงหนึ่งเดียวเท่านั้น
- ชื่อของอิลิเมนต์ต้องเป็นไปตามกฎของชื่อใน XML
- XML ให้ความสำคัญกับตัวหนังสือเล็ก – ใหญ่
- XML จะไม่ตัดส่วนที่เป็น White Space (การเว้นวรรค)

ในข้อความออกเนื้อหาภายในอิลิเมนต์หนึ่งๆมีโอกาสเป็นไปได้ 3 ลักษณะ คือ

1) เป็นอิลิเมนต์อื่นๆ
2) เป็นข้อความปกติ
3) เป็นแท็กว่าง (ไม่มีอะไรอยู่เลย)

ตัวอย่างแบบที่มีอิลิเมนต์อื่นๆอยู่ภายใน

```
<Book>
        <Title> เริ่มคิด-เริ่มสร้าง-เริ่มใช้ XML</Title>
        <Publisher>วิตตี้ กรุ๊ป</Publisher>
</Book>
```

2.8.2.2 ประกาศค่าแอตทริบิวต์ในเอกสาร XML

การประกาศแอตทริบิวต์ มีรูปแบบมาตรฐานดังนี้

<!ATTLIST ชื่ออิลิเมนต์ ชื่อแอตทริบิวต์ ชนิดข้อมูลของแอตทริบิวต์ (#REQUIRED| #IMPLIED| #FIXED) ค่าปกติของแอตทริบิวต์>

สรุปคีย์เวิร์ด #REQUIRED, #IMPLIED, #FIXED

คีย์เวิร์ด	คำอธิบาย
#REQUIRED	หากระบุตามหลังชนิดข้อมูลของแอตทริบิวต์ใด หมายความว่าต้องมีแอตทริบิวต์นั้นเสมอ
#IMPLIED	หากระบุตามหลังชนิดข้อมูลของแอตทริบิวต์ใด หมายความว่าจะมีแอตทริบิวต์นั้นหรือไม่ก็ได้
#FIXED ตามด้วยค่าปกติของแอตทริบิวต์	จะถือว่ามีแอตทริบิวต์นี้เสมอ ไม่ว่าจะระบุแอตทริบิวต์นี้ไว้หรือไม่ระบุก็ตาม คือค่าแอตทริบิวต์จะเท่ากับค่าปกติที่อยู่หลังคำว่า #FIXED เสมอ
ไม่ระบุคีย์เวิร์ด	จะถือว่ามีแอตทริบิวต์นี้หรือไม่ก็ได้

รูปที่ 2.13 รูปภาพแสดงตารางค่าแอตทริบิวต์

ตัวอย่างเช่น <!ATTLIST Document lang CDATA #FIXED "TH">

- Document คือ ชื่ออิลิเมนต์
- Lang คือ ชื่อแอตทริบิวต์
- Document lang คือการประกาศแอตทริบิวต์ชื่อ lang ซึ่งอยู่ภายในอิลิเมนต์ Document
- CDATA คือการประกาศชนิดของค่าแอตทริบิวต์ ว่าเป็นข้อมูลประเภทตัวอักษร
- #FIXED "TH" บอกให้ทราบว่า กำหนดค่าแอตทริบิวต์ไว้ตายตัวเป็นคำว่า "TH"

2.8.2.3 การประกาศค่าเอนทิตี้ในเอกสาร XML

เอนทิตี้ จะใช้สำหรับอ้างถึง "ทรัพยากร" หรือข้อมูลที่เรานิยามไว้ก่อน ทำให้สามารถนำทรัพยากรหรือข้อมูลนั้นกลับมาใช้ใหม่ (reuse) โดยไม่ต้องเขียนบ่อยๆ ดังนั้น เอนทิตี้ก็คือ ค่าคงที่ต่างๆที่เราสร้างขึ้นมาการประกาศเอนทิตี้ใน DTD มีรูปแบบดังนี้

```
<!ENTITY ชื่อเอนทิตี้ ทรัพยากร>
```

2.8.3 การทำงานร่วมกับภาษาข้อมูล XML

XML เป็นเทคโนโลยีที่ไม่ขึ้นกับแพลตฟอร์ม หรือ ภาษาซึ่งหมายความว่ามันจะไม่มีความแตกต่างกันในแต่ละเครื่องคอมพิวเตอร์เลย

ตัวอย่างเช่น การใช้ร่วมกับ Visual Basic ในระบบปฏิบัติการของไมโครซอฟต์ และเครื่องอื่นที่ใช้ระบบยูนิกซ์กับโค้ดที่เป็นภาษาจาวา จริงๆ แล้วหากโปรแกรมบนคอมพิวเตอร์เครื่องหนึ่งจำเป็นที่จะต้องมีการติดต่อกับโปรแกรมอื่น ๆ XML จะเป็นตัวเลือกที่เหมาะสมกับการแลกเปลี่ยนข้อมูล ต่อไปนี้จะเป็นตัวอย่างบางส่วนของการนำ XML ไปใช้งาน อันได้แก่

1) ลดการทำงานของเซิร์ฟเวอร์

แอพพลิเคชั่นที่ทำงานบนเว็บ สามารถใช้ XML เพื่อลดภาระในการทำงานให้กับเว็บเซิร์ฟเวอร์ได้ โดยการเก็บข้อมูลเอาไว้ในเครื่องไคลเอนต์ให้นานที่สุดเท่าที่จะนานได้ หลังจากนั้นจึงค่อยทำการส่งในรูปแบบของเอกสาร XML ขนาดใหญ่

2) เนื้อหาของเว็บไซต์

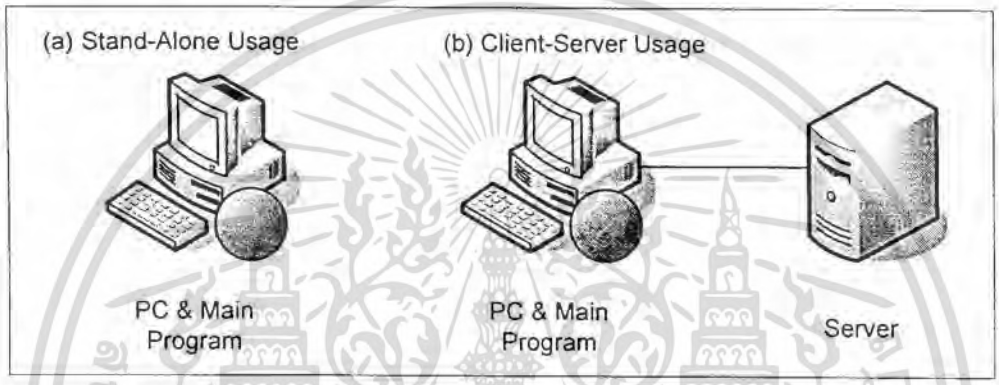
ในเว็บของ W3C เองก็มีการใช้ XML ในการแสดงข้อกำหนดต่าง ๆ เอกสาร XML เหล่านั้นจะถูกแปลงมาเป็น HTML เพื่อการแสดงผล (โดยใช้ XSLT) หรือ แปลงมาเป็นรูปแบบการนำเสนออื่น ๆ ได้อีกจากที่เคยใช้ HTML ในการจัดการกับเนื้อหาของเว็บไซต์

ขณะนี้บางเว็บไซต์ได้มีการนำเอา XML เข้ามาจัดการกับเนื้อหาทั้งหมดของตนเองแล้ว โดย XML อาจจะถูกแปลงให้เป็น HTML โดย XSLT หรือ แสดงผลโดยตรงผ่านเบราเซอร์เลยด้วย CSS ในความเป็นจริงเว็บเซิร์ฟเวอร์สามารถตรวจสอบได้ว่าเบราเซอร์ประเภทใดที่กำลังรับข้อมูลอยู่ และตัดสินใจได้ว่าควรทำอะไร (เช่น แปลง XML ไปเป็น HTML และส่งไปยังเบราเซอร์เก่าๆ หรือส่ง XML ตรงๆ ไปยังเบราเซอร์ใหม่ๆ เลยเพื่อลดภาระของเซิร์ฟเวอร์เอง)

บทที่ 3
วิธีดำเนินการวิจัย

3.1 รูปแบบโครงสร้างที่ใช้ในการดำเนินการวิจัย

โครงสร้างที่เราใช้ในการดำเนินการวิจัยนี้ เราได้แบ่งออกเป็น 2 โครงสร้างในการทดลอง โดยโครงสร้างที่หนึ่งเป็นโครงสร้างการใช้งานโดยลำพังหรือแบบเอกเทศ ในขณะที่โครงสร้างที่สองมีลักษณะแบบไคลเอนต์-เซิร์ฟเวอร์ คือมีเครื่องหนึ่งทำหน้าที่เป็นไคลเอนต์ ในขณะที่อีกเครื่องหนึ่งทำหน้าที่เป็นเซิร์ฟเวอร์ ทำการแสดงโครงสร้างทั้งสองแบบได้ดังภาพ



รูปที่ 3.1 รูปภาพแสดงรูปแบบโครงสร้างที่ใช้ในการทดลอง

โดยโครงสร้างแบบที่หนึ่งที่เป็นโครงสร้างแบบเอกเทศ เราได้ใช้โครงสร้างนี้เพื่อพัฒนาโปรแกรมหลักที่ใช้ในการค้นหาและทำลายแอดแวร์และสปายแวร์บนเครื่องคอมพิวเตอร์ส่วนบุคคล ซึ่งโดยปกตินั้นการใช้งานแบบเอกเทศนี้ก็เพียงพอต่อการใช้งานแล้ว แต่เพื่อสนับสนุนคุณสมบัติในด้านการอัพเดตฐานข้อมูลแอดแวร์และสปายแวร์ของโปรแกรม จึงได้มีส่วนเพิ่มเติมและพัฒนาเป็นโครงสร้างที่สองที่มีลักษณะการทำงานเป็นแบบไคลเอนต์-เซิร์ฟเวอร์ เพื่อให้เซิร์ฟเวอร์ให้บริการในการอัพเดตฐานข้อมูลได้

3.2 ลักษณะการทำงานของโปรแกรมที่ทำการทดลอง

ในการพัฒนาโปรแกรมค้นหาและทำลายแอดแวร์และสปายแวร์บนเครื่องคอมพิวเตอร์ส่วนบุคคลนี้ เราได้แบ่งส่วนการทำงานออกเป็นส่วนใหญ่ๆ เพื่อให้ง่ายต่อการออกแบบและการพัฒนา โดยแบ่งการทำงานออกเป็น 3 ส่วน ดังนี้

3.2.1 ส่วนของโปรแกรมหลักที่ใช้ในการค้นหาและทำลายแอดแวร์และสปายแวร์

ส่วนนี้เป็นส่วนที่ให้บริการผู้ใช้ทำการค้นหาและทำลายแอดแวร์และสปายแวร์ที่อยู่ในเครื่อง ซึ่งถือว่าเป็นส่วนหลักของการดำเนินการวิจัยนี้ โดยส่วนนี้จะเป็นส่วนที่ให้ทำการติดตั้งบนเครื่องผู้ใช้ทั่วไป โดยสามารถใช้งานได้แม้ไม่ต้องทำการเชื่อมต่อกับเครือข่าย

โดยส่วนนี้นอกจากการค้นหาและทำลายแอดแวร์และสปายแวร์แล้ว ยังคงมีส่วนประกอบย่อยอื่นๆ เพื่อสนับสนุนการทำงาน เช่น การเก็บข้อมูลทางสถิติ การปรับแต่งการทำงานตามความเหมาะสม การอัพเดตฐานข้อมูลแอดแวร์และสปายแวร์ในเครื่อง และส่วนช่วยเหลือผู้ใช้งาน

3.2.2 ส่วนของเซิร์ฟเวอร์ผู้ให้บริการในการอัพเดตฐานข้อมูลแอดแวร์และสปายแวร์

ส่วนนี้ถือเป็นส่วนเสริมจากส่วนของโปรแกรมหลัก โดยในส่วนนี้ได้ถูกเพิ่มเข้ามาเพื่อสนับสนุนโปรแกรมหลักในการที่จะอัพเดตฐานข้อมูลแอดแวร์และสปายแวร์ในเครื่องของผู้ใช้งาน เพื่อให้ฐานข้อมูลเหล่านั้นมีความทันสมัย

โดยในส่วนนี้เราได้ทำการทดลองโดยให้เครื่องคอมพิวเตอร์เครื่องหนึ่งจำลองการทำงานเป็นเครื่องเซิร์ฟเวอร์ที่ให้บริการในการอัพเดตฐานข้อมูล

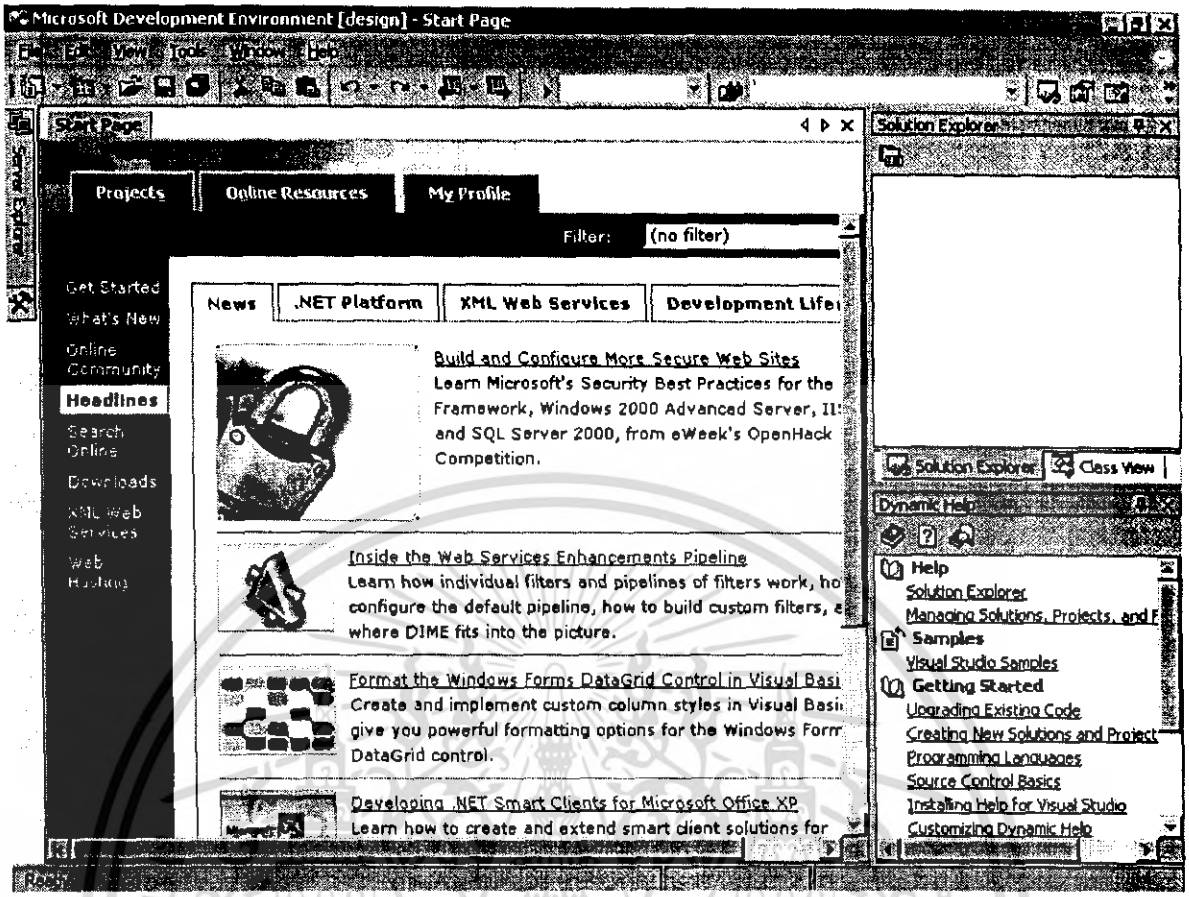
3.2.3 ส่วนของโปรแกรมจัดการฐานข้อมูลแอดแวร์และสปายแวร์

ส่วนนี้เป็นโปรแกรมเพิ่มเติมสำหรับผู้พัฒนาโปรแกรม โดยมีลักษณะการทำงานเป็นตัวแก้ไขไฟล์ข้อมูล XML เพื่อให้ผู้พัฒนาสามารถจัดการไฟล์ฐานข้อมูลแอดแวร์และสปายแวร์ที่มีลักษณะการจัดเก็บในลักษณะ XML ได้สะดวกยิ่งขึ้น

3.3 โครงสร้างของโปรแกรมหลักที่ใช้ในการค้นหาและทำลายแอดแวร์และสปายแวร์

ส่วนนี้เป็นส่วนหลักของการดำเนินการวิจัยนี้ โดยส่วนนี้ได้ใช้ภาษา VB.Net และได้ใช้โปรแกรม Microsoft Visual Studio 2003 เป็นเครื่องมือที่ใช้ในการพัฒนาโปรแกรมหลักนี้ โดยจะขอกล่าวถึงการใช้งานโปรแกรม Microsoft Visual Studio 2003 ก่อน ซึ่งถือเป็นโปรแกรมหลักที่ใช้ในการพัฒนาโปรแกรมครั้งนี้

โปรแกรม Microsoft Visual Studio 2003 เป็นเครื่องมือที่ช่วยอำนวยความสะดวกในการแก้ไขและปรับปรุงโค๊ดโปรแกรมได้อย่างมีประสิทธิภาพ โดยในที่นี้เราได้ใช้โค๊ดภาษา VB.Net ซึ่งถือว่าเป็นภาษาใหม่ที่ทำงานบนไมโครซอฟต์ดอทเน็ตเฟรมเวิร์ค ซึ่งสนับสนุนการทำงานข้ามแพลตฟอร์มได้ (แต่ต้องเป็นแพลตฟอร์มที่สนับสนุนไมโครซอฟต์ดอทเน็ตเฟรมเวิร์คเท่านั้น)



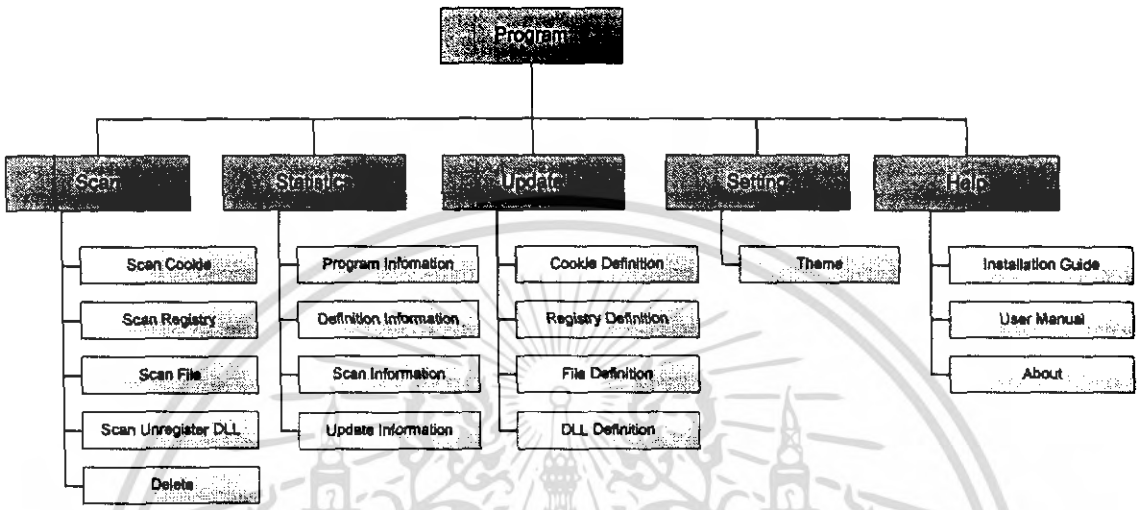
รูปที่ 3.2 รูปภาพแสดงโปรแกรม Microsoft Visual Studio 2003

อย่างไรก็ตามเราภาษา VB.Net เราไม่จำเป็นต้องทำการคอมไพล์โปรแกรมที่เราพัฒนาโดยใช้โปรแกรม Microsoft Visual Studio 2003 เพียงอย่างเดียว เราสามารถที่จะทำการคอมไพล์โปรแกรมด้วยการเรียกคำสั่ง vbc ผ่านทางคอมมานไลน์ได้อีกด้วย ตัวอย่างเช่น หากต้องการคอมไพล์ Module1.vb เป็นไฟล์ AntiMalware.exe สามารถคอมไพล์ด้วยคำสั่งดังนี้

```
vbc /out:AntiMalware.exe Module1.vb
```

โดยคำสั่ง vbc นี้คือการเรียกใช้โปรแกรม Visual Basic Complier ซึ่งเป็นโปรแกรมที่สามารถแจกจ่ายให้นำไปใช้ได้โดยไม่เสียค่าใช้จ่ายแต่อย่างใด แต่เพื่อความสะดวกในการพัฒนาโปรแกรมที่มีความซับซ้อนและมีการทำงานหลายส่วนร่วมกัน อย่างโปรแกรมค้นหาและทำลายแอดแวร์และสปายแวร์นี้ เราจึงเลือกใช้ Microsoft Visual Studio 2003 เป็นเครื่องมือในการพัฒนา

โดยในการพัฒนาส่วนของโปรแกรมหลักที่ใช้ในการค้นหาและทำลายแอดแวร์และสปายแวร์นี้เราได้แบ่งส่วนการทำงานออกเป็นส่วนย่อยๆ ได้อีก อันได้แก่ ส่วนการค้นหาและทำลายแอดแวร์และสปายแวร์ ส่วนการเก็บข้อมูลทางสถิติ ส่วนการอัพเดตฐานข้อมูล ส่วนการปรับแต่งลักษณะ และส่วนช่วยเหลือผู้ใช้งาน โดยสามารถแสดงแต่ละส่วนออกได้เป็นดังภาพ



รูปที่ 3.3 รูปภาพแสดงโครงสร้างของโปรแกรมหลัก

จากภาพจะเห็นว่าโปรแกรมได้ถูกแบ่งการทำงานออกเป็น 5 ส่วน คือ โดยจะขออธิบายเป็นส่วนๆ ในหัวข้อย่อยถัดไป

3.3.1 ส่วนของการค้นหาและทำลายแอดแวร์และสปายแวร์ (Scan)

เป็นส่วนที่มีหน้าที่ในการค้นหาและทำลายสปายแวร์และแอดแวร์ที่อาศัยอยู่ภายในเครื่อง โดยทางโปรแกรมจะทำการเลือกค้นหาเฉพาะจุดที่คาดว่าจะเกิดความเสี่ยง โดยแบ่งการทำงานเป็น 5 ระยะ (Phase) คือ

1. การค้นหามัลแวร์จากคุ้กกี้ (Scan Cookie)
2. การค้นหามัลแวร์ที่ฝังตัวในรีจิสตรี (Scan Registry)
3. การค้นหามัลแวร์จากไฟล์ข้อมูล (Scan File)
4. การค้นหามัลแวร์ที่ฝังตัวในไดนามิกลิงก์ (Scan DLL)
5. การทำลายมัลแวร์ (Delete)

3.3.2 ส่วนการเก็บข้อมูลทางสถิติ (Statistics)

เป็นส่วนที่มีหน้าที่ในการแสดงรายละเอียดของโปรแกรม ซึ่งประกอบไปด้วย

1. Program Information เป็นส่วนที่แสดงข้อมูลของตัวโปรแกรม
2. Definition Information เป็นส่วนที่แสดงข้อมูลของฐานข้อมูล
3. Scan Information เป็นส่วนที่แสดงข้อมูลการใช้งานโปรแกรมของผู้ใช้
4. Update Information เป็นส่วนที่แสดงข้อมูลการอัพเดตโปรแกรมของผู้ใช้

3.3.3 ส่วนการอัพเดตฐานข้อมูลแอดแวร์และสปายแวร์ (Update)

เป็นส่วนที่มีหน้าที่ในการปรับปรุงฐานข้อมูลให้มีความทันสมัยอยู่เสมอ โดยทางโปรแกรมจะทำการปรับปรุงข้อมูลทันทีเมื่อผู้ใช้ทำการอัพเดต ซึ่งมีการอัพเดตทั้งหมด 4 ส่วน คือ

1. Cookie Definition ฐานข้อมูลคุกกี้ไฟล์
2. Registry Definition ฐานข้อมูลรีจิสตรี
3. File Definition ฐานข้อมูลไฟล์แอดแวร์และสปายแวร์
4. DLL Definition ฐานข้อมูลไฟล์ DLL ไลบรารี

3.3.4 ส่วนการปรับแต่งคุณลักษณะ (Setting)

เป็นส่วนที่มีหน้าที่ ในการอำนวยความสะดวกให้กับผู้ใช้ โดยมีการปรับปรุงรูปแบบ หน้าตาตามความต้องการของผู้ใช้มากที่สุด ซึ่งประกอบไปด้วย

1. Theme เป็นการปรับแต่งการแสดงผลการใช้งานที่เหมาะสมกับผู้ใช้

3.3.5 ส่วนการช่วยเหลือผู้ใช้งาน (Help)

เป็นส่วนที่มีหน้าที่ในการช่วยเหลือผู้ใช้ ให้มีความเข้าใจวิธีและขั้นตอนในการใช้โปรแกรม ซึ่งประกอบไปด้วย

1. Installation Guide ส่วนแนะนำการติดตั้งโปรแกรมให้กับผู้ใช้
2. User Manual ส่วนคู่มือการใช้งานเพื่อสนับสนุนผู้ใช้
3. About ส่วนแสดงข้อมูลโปรแกรมและรายชื่อคณะผู้จัดทำ

โดยในส่วนการช่วยเหลือผู้ใช้งานนี้ เราได้ใช้โปรแกรม HTMLHelp Workshop ซึ่งเป็นโปรแกรมที่สามารถโหลดใช้งานได้โดยไม่เสียค่าใช้จ่าย ซึ่งเป็นโปรแกรมที่ช่วยทำให้สามารถที่จะสร้างเอกสารการช่วยเหลือผู้ใช้งานได้อย่างสะดวกยิ่งขึ้น อีกทั้งยังเป็นการสร้างเอกสารการช่วยเหลือผู้ใช้งานอย่างมีมาตรฐานอีกด้วย

รูปที่ 3.4 รูปภาพแสดงโปรแกรม HTML Help Workshop

โดยในการใช้งานโปรแกรมนี้เราต้องทำการสร้างไฟล์คำอธิบายต่างๆ เป็นเอกสาร HTML และนำมาทำการเชื่อมโยงโดยใช้โปรแกรม HTML Help Workshop นี้แล้วทำการคอมไพล์ ซึ่งจะได้เป็นไฟล์ช่วยเหลือมาตรฐานนามสกุล *.chm ซึ่งเราจะใช้ภาษา VB.Net มาเขียนโปรแกรมเพื่อทำการเชื่อมโยงกับตัวโปรแกรมอีกครั้ง

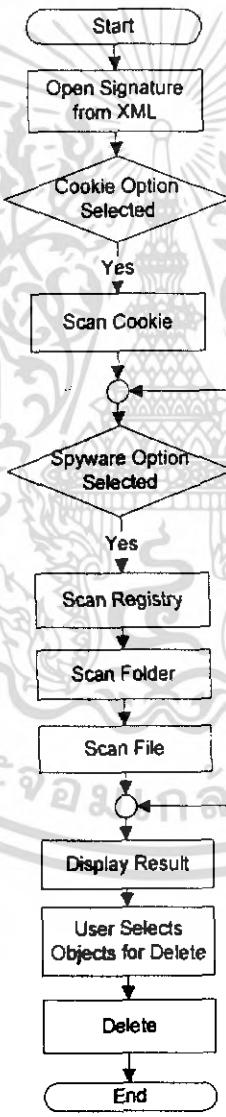
3.4 การทำงานของโปรแกรมหลักที่ใช้ในการค้นหาและทำลายแอดแวร์และสปายแวร์

การทำงานในส่วนของโปรแกรมหลักนี้ จะทำการแยกอธิบายออกเป็น 3 หัวข้อตามลักษณะการทำงานดังนี้ คือ ส่วนแรกคือส่วนการค้นหาและทำลายแอดแวร์และสปายแวร์ ส่วนที่สองคือส่วนของการอัพเดตฐานข้อมูล และส่วนที่สามคือส่วนของข้อมูลและการแสดงผล ซึ่งจะเป็นส่วนที่เกี่ยวกับส่วนการเก็บข้อมูลทางสถิติและส่วนการปรับแต่งลักษณะ ซึ่งทั้งสองมีหลักการทำงานที่คล้ายกันจึงขอกล่าวในหัวข้อเดียวกัน

3.4.1 ส่วนการค้นหาและทำลายแอดแวร์และสปายแวร์

ในขั้นตอนการค้นหาและทำลายแอดแวร์และสปายแวร์สามารถแสดงลำดับการทำงานได้ดังภาพด้านล่าง โดยการทำงานจะแบ่งออกเป็นสองส่วนใหญ่ๆ คือ ทำการค้นหาและทำการทำลาย โดยในขั้นตอนการค้นหาจะเป็นการให้ผู้ใช้เลือกว่าต้องการทำการค้นหาส่วนใดบ้าง โดยมีทั้งหมดสองส่วนคือ ส่วนของการค้นหาคุกกี้ และค้นหาแอดแวร์สปายแวร์ ซึ่งในส่วนของการค้นหาแอดแวร์สปายแวร์จะประกอบไปด้วยการค้นหาส่วนย่อย 3 ส่วนด้วยกันคือ ค้นหาในรีจิสตรี โฟลเดอร์ และไฟล์

เมื่อทำการค้นหาเสร็จสิ้นก็จะให้ผู้ใช้งานเลือกว่าต้องการลบตัวใดบ้างหรือไม่ต้องการลบเลย โดยในขั้นตอนการลบนี้ ในบางกรณี โปรแกรมต้องทำการปิดการทำงานของแอดแวร์และสปายแวร์ก่อน โดยจะทำการหยุดโปรเซสและถอนรีจิสตรีออก แล้วจึงทำการลบตามปกติ



รูปที่ 3.5 ภาพแสดงขั้นตอนการทำการตรวจจับ (Scan)

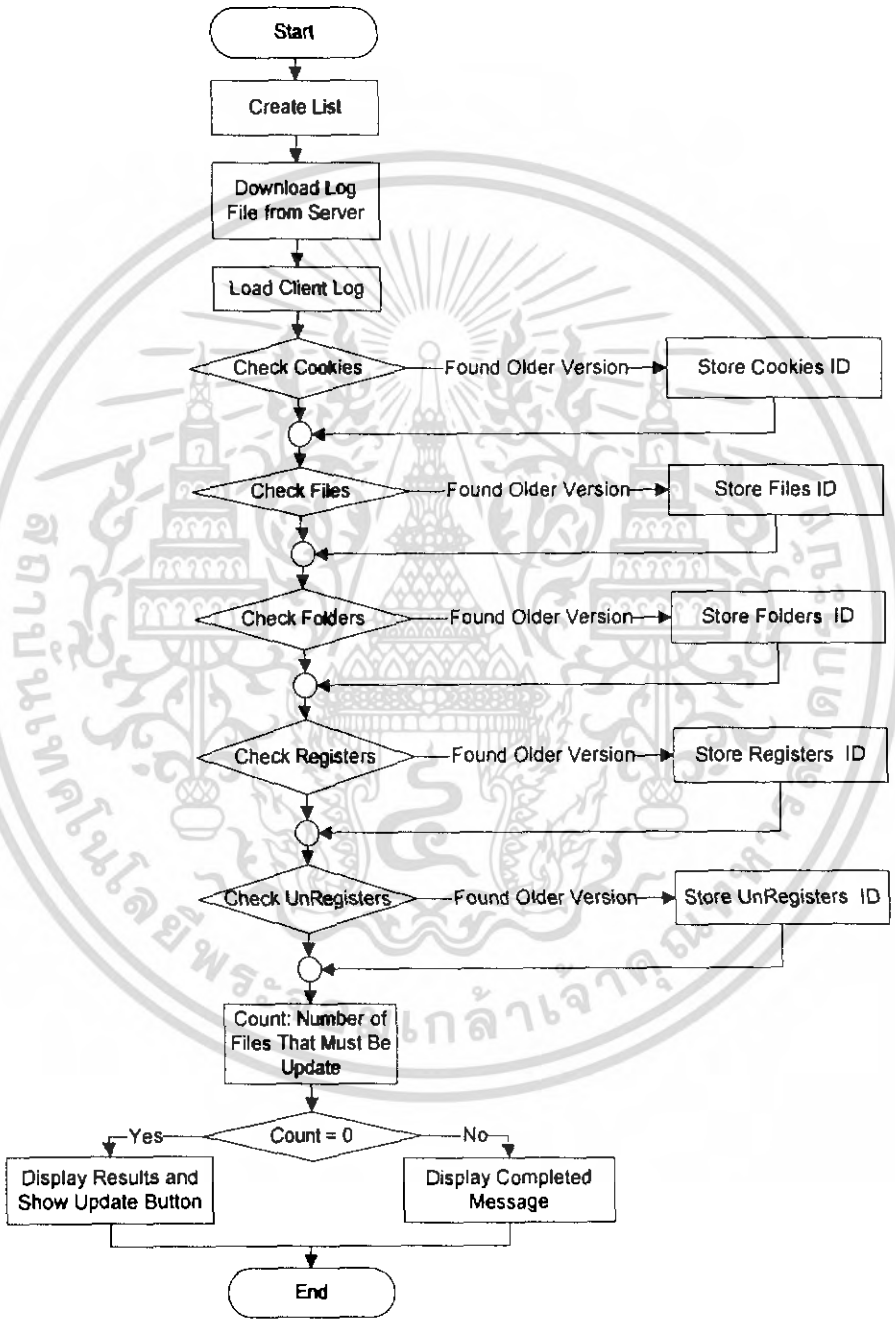
โดยในขั้นตอนการค้นหาไฟล์ต่างๆ นี้ หลักการในการค้นหาทั้ง 4 ส่วนคือ คุกกี้ รีจิสตรี โฟลเดอร์ และไฟล์นั้น จะมีหลักการเหมือนกันคือ จะทำการเปิดลายเซนต์ซึ่งเป็นฐานข้อมูลว่าแอดแวร์และสปายแวร์เหล่านั้น มีรูปแบบอย่างไร เพื่อนำมาทำการเปรียบเทียบและทำการยืนยันว่าวัตถุที่เราพบเหล่านั้นเป็นแอดแวร์และสปายแวร์จริง และเมื่อพบก็จะทำการบันทึกเป็นรายงานต่อไป

และในการทดลองค้นหาไฟล์เหล่านี้ เราเลือกทำการทดลองหาเพียงบางจุดเพื่อความรวดเร็วในการค้นหา โดยเราเลือกเฉพาะจุดที่มักพบแอดแวร์และสปายแวร์เป็นประจำมาทำการค้นหา ซึ่งมีดังต่อไปนี้

1) **การค้นหาคุกกี้** เราจะทำการค้นหาที่ไฟล์ในโฟลเดอร์ดังต่อไปนี้
   - <DRIVE>\Documents and Settings\<USER>\Cookies\ ซึ่งเป็นการค้นหาเฉพาะในโฟลเดอร์คุกกี้ส่วนบุคคล
2) **การค้นหารีจิสตรี** เราจะทำการค้นหาที่ไฮฟ์ดังต่อไปนี้
   - HKEY_LOCAL_MACHINE
   - HKEY_LOCAL_MACHINE\SOFTWARE\MICROSOFT\WINDOWS\CURRENTVERSION\RUN
   - HKEY_LOCAL_MACHINE\SOFTWARE\MICROSOFT\WINDOWS\CURRENTVERSION\RUNONCE
   - HKEY_CLASSES_ROOT
   - HKEY_CURRENT_USER
3) **การค้นหาโฟลเดอร์** เราจะทำการค้นหาที่โฟลเดอร์ในโฟลเดอร์ดังต่อไปนี้
   - <DRIVE>\Documents and Settings\<USER>\Application Data\
   - <DRIVE>\Documents and Settings\<USER>\Favorites\
   - <DRIVE>\Program Files\
   - <DRIVE>\Windows\
   - <DRIVE>\Windows\Downloaded Program Files\
4) **การค้นหาไฟล์** เราจะทำการค้นหาไฟล์ในโฟลเดอร์ดังต่อไปนี้
   - <DRIVE>\Documents and Settings\<USER>\Application Data\
   - <DRIVE>\Windows\System32\
   - <DRIVE>\Program Files\
   - <DRIVE>\Windows\
   - <DRIVE>\Windows\Downloaded Program Files\

3.4.2 ส่วนการอัพเดตฐานข้อมูลแอดแวร์และสปายแวร์

ในส่วนของการอัพเดตฐานข้อมูลนี้ เป็นส่วนที่ต้องทำการติดต่อกับเซิร์ฟเวอร์เพื่อรับบริการในการอัพเดตฐานข้อมูล โดยในส่วนนี้แบ่งออกเป็น 2 ขั้นตอนใหญ่ๆ คือ ขั้นตอนการตรวจสอบการอัพเดตว่ามีไฟล์ใดที่ต้องทำการอัพเดตบ้างโดยดูจากวันที่และเวลา เมื่อมีไฟล์ที่จำเป็นต้องอัพเดต ก็จะทำการให้ผู้ใช้เลือกที่จะเข้าสู่ขั้นตอนที่สอง ซึ่งคือขั้นตอนการอัพเดตไฟล์



รูปที่ 3.6 รูปภาพแสดงขั้นตอนการตรวจสอบการอัพเดต

3.4.2.1 ขั้นตอนการตรวจสอบการอัพเดต

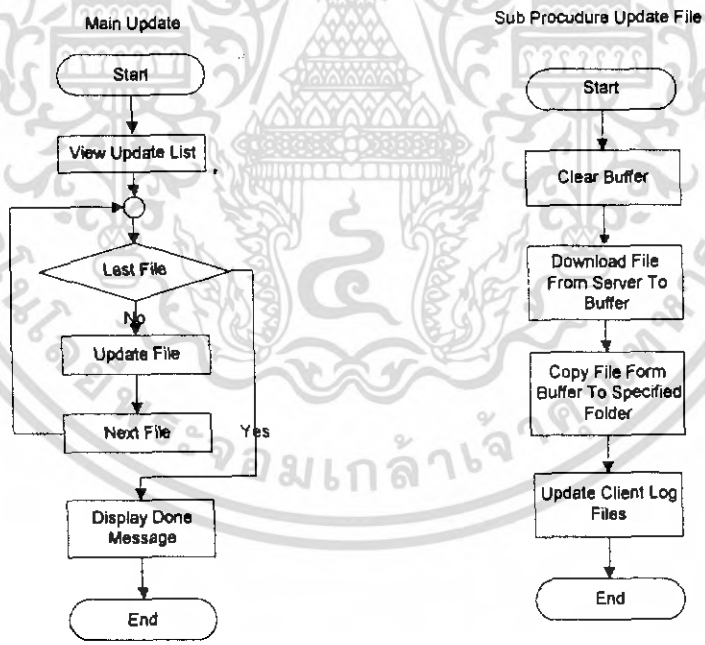
ในส่วนขั้นตอนการตรวจสอบการอัพเดตนี้จะเริ่มต้นจากการดูว่าว่ามีไฟล์ใดที่ต้องทำการอัพเดตบ้างโดยดูจากวันที่และเวลา โดยจะทำการดาวน์โหลดไฟล์ที่เป็นล็อกไฟล์จากเซิร์ฟเวอร์มาทำการเปรียบเทียบกับล็อกไฟล์จากไคลเอนท์ เมื่อมีไฟล์ที่จำเป็นต้องอัพเดต เช่น เมื่อไฟล์ที่เซิร์ฟเวอร์มีไฟล์ที่มีวันที่ที่ใหม่กว่าหรือมีไฟล์ใหม่เข้ามา ก็จะทำการบันทึกรหัสไฟล์ที่ต้องการอัพเดตเหล่านั้นเก็บไว้

เมื่อทำการตรวจสอบไฟล์ทั้งหมดแล้ว ถ้าพบว่ามีไฟล์ที่ต้องการการอัพเดตก็จะทำการให้ผู้ใช้เลือกที่จะเข้าสู่ขั้นตอนที่สอง ซึ่งคือขั้นตอนการอัพเดตไฟล์ต่อไป

3.4.2.2 ขั้นตอนการอัพเดต

ในส่วนขั้นตอนการอัพเดตไฟล์นี้ เราจะทำการนำไฟล์ที่ได้จากขั้นตอนการตรวจสอบการอัตเดต ไปทำการวนลูปอัพเดตไฟล์ไปทีละไฟล์จนกระทั่งถึงไฟล์สุดท้าย

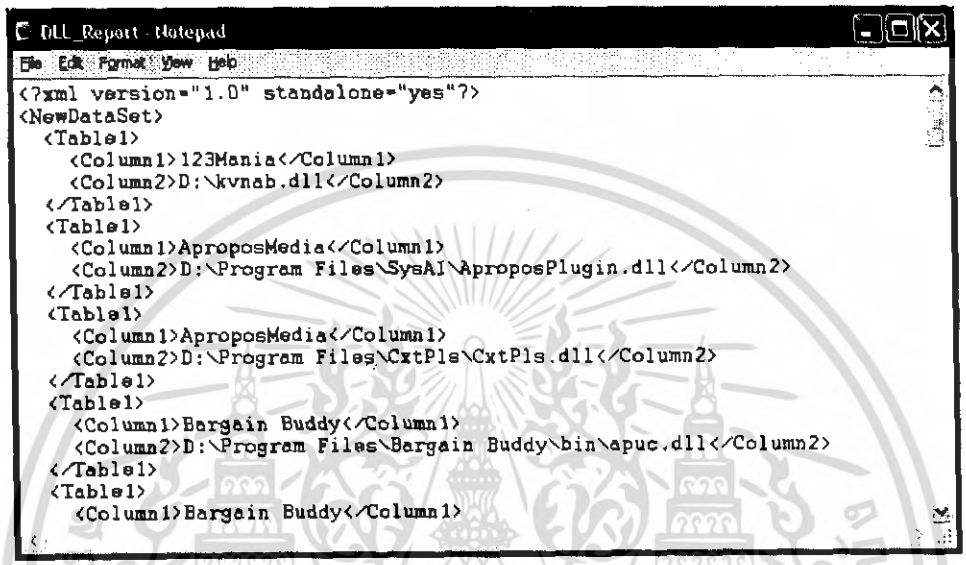
โดยจะทำการอัพเดตแต่ละไฟล์นั้นจะเป็นการโหลดไฟล์จากเซิร์ฟเวอร์มาเก็บยังไคลเอนต์ ซึ่งในขั้นตอนนี้เราพบว่ามักมีปัญหาในเรื่องของการโหลดไฟล์จากเซิร์ฟเวอร์ได้ไม่สมบูรณ์ เราจึงทำการโหลดไฟล์เหล่านั้นมาเก็บยังบัฟเฟอร์ก่อนเพื่อตรวจสอบ เมื่อพบว่าดาวน์โหลดได้สำเร็จ จึงค่อยนำไฟล์นั้นไปยังสถานที่โฟลเดอร์ที่ระบุ และสุดท้ายทำการปรับปรุงล็อกไฟล์ในแต่ละไฟล์



รูปที่ 3.7 รูปภาพแสดงขั้นตอนการอัพเดต

3.4.3 ส่วนของข้อมูลและการแสดงผล

การทำงานในส่วนนี้เป็นส่วนที่เกี่ยวกับส่วนการเก็บข้อมูลทางสถิติและส่วนการปรับแต่งลักษณะ ซึ่งทั้งสองมีหลักการทำงานที่คล้ายกันคือ เราจะใช้ไฟล์ทำการเก็บบันทึกข้อมูลเหล่านั้นเอาไว้ เมื่อมีการเปลี่ยนแปลง เช่น เมื่อผู้ใช้ทำการเปลี่ยนลักษณะของโปรแกรมหรือทำการอัพเดตไฟล์ ข้อมูลการเปลี่ยนแปลงต่างๆ เหล่านั้น ก็จะถูกทำการจัดการเก็บลงบนไฟล์



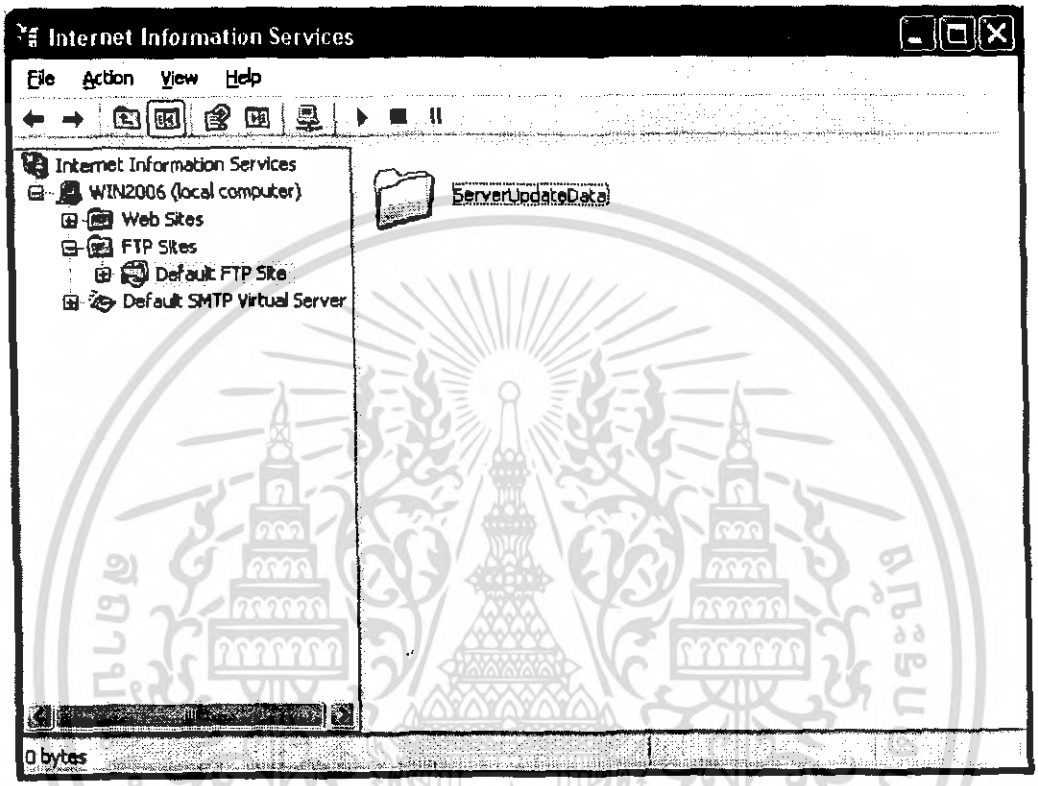
รูปที่ 3.8 ตัวอย่างรายงานข้อมูลไฟล์ที่ถูกตรวจพบ

นอกจากนี้ลักษณะการทำงานนี้ยังรวมไปถึงการสร้างรายงานการตรวจพบแอดแวร์และสปายแวร์ต่างๆ อีกด้วย โดยเมื่อโปรแกรมมีการตรวจพบจะทำการบันทึกข้อมูลของไฟล์ที่พบ จัดเก็บเป็นรายงาน โดยรายงานอาจจะประกอบไปด้วย ชื่อแอดแวร์และสปายแวร์ที่ถูกตรวจพบและตำแหน่งที่พบ เป็นต้น โดยจะทำการจัดเก็บแยกตามประเภทที่พบ

3.5 การทำงานของเซิร์ฟเวอร์ผู้ให้บริการในการอัพเดตฐานข้อมูล

ในส่วนการทำงานของโปรแกรมหลักไม่อาจจะสมบูรณ์ได้ หากไม่มีส่วนสนับสนุนการอัพเดตฐานข้อมูลแอดแวร์และสปายแวร์ ซึ่งก็คือส่วนของเซิร์ฟเวอร์ผู้ให้บริการอัพเดตฐานข้อมูล โดยในส่วนนี้เราได้ทำการทดลองโดยให้เครื่องคอมพิวเตอร์เครื่องหนึ่งจำลองการทำงานเป็นเครื่องเซิร์ฟเวอร์ โดยได้เลือกใช้โปรแกรมจำลองเซิร์ฟเวอร์ที่ชื่อว่า Internet Information Services (IIS) ซึ่งเป็นโปรแกรมที่ใช้จำลองเซิร์ฟเวอร์บนระบบปฏิบัติการไมโครซอฟต์วินโดวส์

โดยแม้ว่าในการพัฒนาโปรแกรมนั้น เราได้พัฒนาโปรแกรมสามารถที่จะสนับสนุนเซิร์ฟเวอร์ในหลายๆ รูปแบบ ทั้งการสนับสนุนเซิร์ฟเวอร์บนระบบปฏิบัติการ UNIX และบนระบบปฏิบัติการไมโครซอฟต์วินโดวส์ แต่เราเลือกใช้ Internet Information Services เนื่องจากความสะดวกในการจัดหาและการใช้งาน



รูปที่ 3.9 รูปภาพแสดงโปรแกรม Internet Information Services

โดยที่นี้เราได้ทำการจำลองเครื่องคอมพิวเตอร์เป็น FTP เซิร์ฟเวอร์ เพื่อทำการสนับสนุนในการโอนไฟล์ โดยไคลเอนท์ที่เป็นโปรแกรมใช้งานจะทำการล็อกอินเข้ามายังเซิร์ฟเวอร์ และทำการโหลดไฟล์ที่ต้องการจากเซิร์ฟเวอร์ไปเก็บยังเครื่องของไคลเอนท์

3.6 การทำงานของโปรแกรมจัดการฐานข้อมูลแอดแวร์และสปายแวร์

ส่วนนี้เป็นโปรแกรมเพิ่มเติมสำหรับผู้พัฒนาโปรแกรม โดยมีลักษณะการทำงานเป็นตัวแก้ไขไฟล์ข้อมูล XML เพื่อให้ผู้พัฒนาสามารถจัดการไฟล์ฐานข้อมูลแอดแวร์และสปายแวร์ที่มีลักษณะการจัดเก็บในลักษณะ XML ได้สะดวกยิ่งขึ้น

บทที่ 4
ผลการวิจัย

4.1 เครื่องมือที่ใช้ในการทดสอบโปรแกรม

ในขั้นตอนการทำการทดสอบโปรแกรมค้นหาและทำลายแอดแวร์และสปายแวร์ที่เราพัฒนาขึ้นนั้นเราได้ใช้เครื่องมือในการทดสอบโปรแกรมดังต่อไปนี้

4.1.1 เครื่องมือทางด้านฮาร์ดแวร์

1) เครื่องคอมพิวเตอร์ Pentium 4 ความเร็วซีพียู 2.4 GHz
2) แรม 256 MB
3) พื้นที่ว่างของฮาร์ดดิสก์ขนาด 40 MB เพื่อใช้การติดตั้ง
4) อุปกรณ์พื้นฐานได้แก่ หน้าจอ คีย์บอร์ด และเมาส์
5) อุปกรณ์เชื่อมต่อเครือข่ายเพื่อใช้ในการอัพเดตไฟล์

4.1.2 เครื่องมือทางด้านซอฟต์แวร์

1) ระบบปฏิบัติการที่รองรับไมโครซอฟต์ดอทเน็ตเฟรมเวิร์ค เช่น ไมโครซอฟต์วินโดวส์ 2000 และไมโครซอฟต์วินโดวส์ XP เป็นต้น
2) ไมโครซอฟต์ดอทเน็ตเฟรมเวิร์ค 1.0

4.2 ขั้นตอนการติดตั้งโปรแกรม

ในขั้นตอนการติดตั้งโปรแกรมที่เราพัฒนา เราจำเป็นต้องติดตั้งไมโครซอฟต์ดอทเน็ตเฟรมเวิร์ค 1.0 ก่อนแล้วจึงสามารถติดตั้งโปรแกรมเพื่อเรียกใช้งานได้ทันที

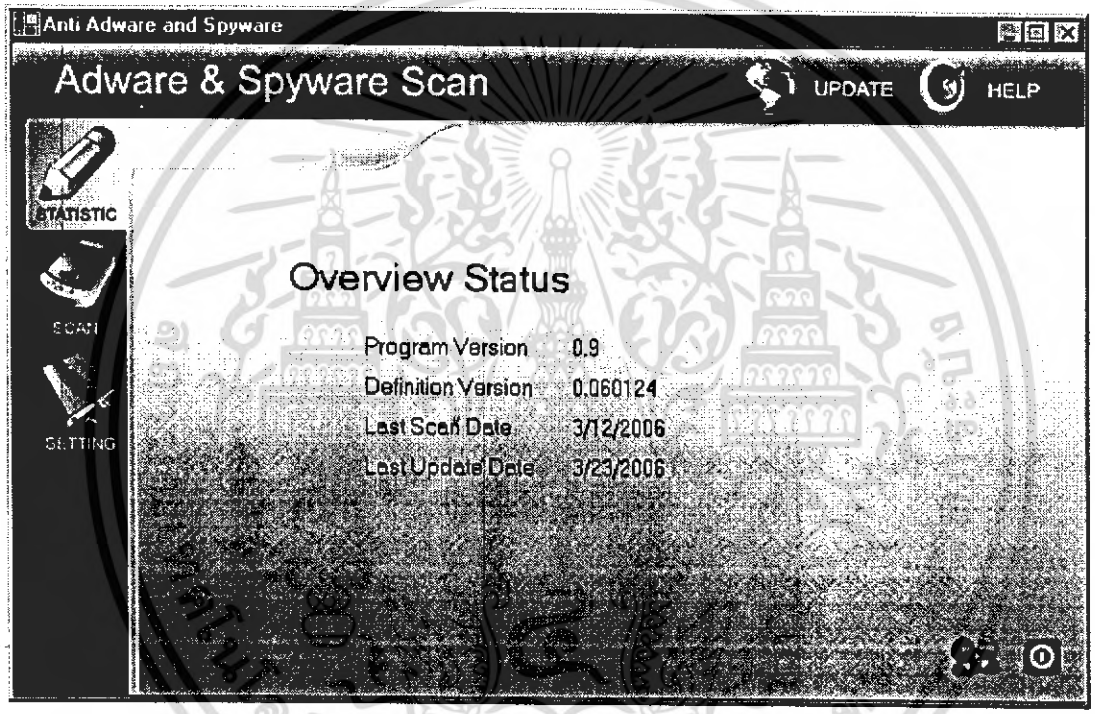
4.3 ลักษณะของโปรแกรมและการใช้งาน

ในหัวข้อนี้จะกล่าวถึงลักษณะของโปรแกรมค้นหาและทำลายแอดแวร์และสปายแวร์ที่เราพัฒนาว่ามีลักษณะและการใช้งานอย่างไร โดยจะกล่าวถึงเป็นส่วนๆ ตามหัวข้อการทำงาน ดังนี้ คือ การแสดงข้อมูลภาพรวม การค้นหาและทำลายแอดแวร์และสปายแวร์ การปรับแต่งโปรแกรมตามความเหมาะสม การอัพเดตฐานข้อมูลแอดแวร์และสปายแวร์ในเครื่อง ส่วนช่วยเหลือผู้ใช้งาน และส่วนขั้นตอนการปิดโปรแกรม

4.3.1 หน้าต่างการแสดงข้อมูลภาพรวม

การทำงานของโปรแกรมจะมีหน้าหลักคือหน้า Statistic ซึ่งเป็นหน้าต่างที่แสดงภาพรวมของโปรแกรม โดยในหน้าต่างนี้จะเป็นการบอกข้อมูลสถานะของโปรแกรมโดยรวม แบ่งข้อมูลออกเป็น 4 ส่วนคือ

- **Program Version** คือ รุ่นของโปรแกรม
- **Definition Version** คือ รุ่นของฐานข้อมูล
- **Last Scan Date** คือ บอกวันที่ทำการค้นหาแอดแวร์และสปายแวร์ครั้งสุดท้าย
- **Last Update Date** คือ บอกวันที่ทำการอัพเดตฐานข้อมูลครั้งสุดท้าย

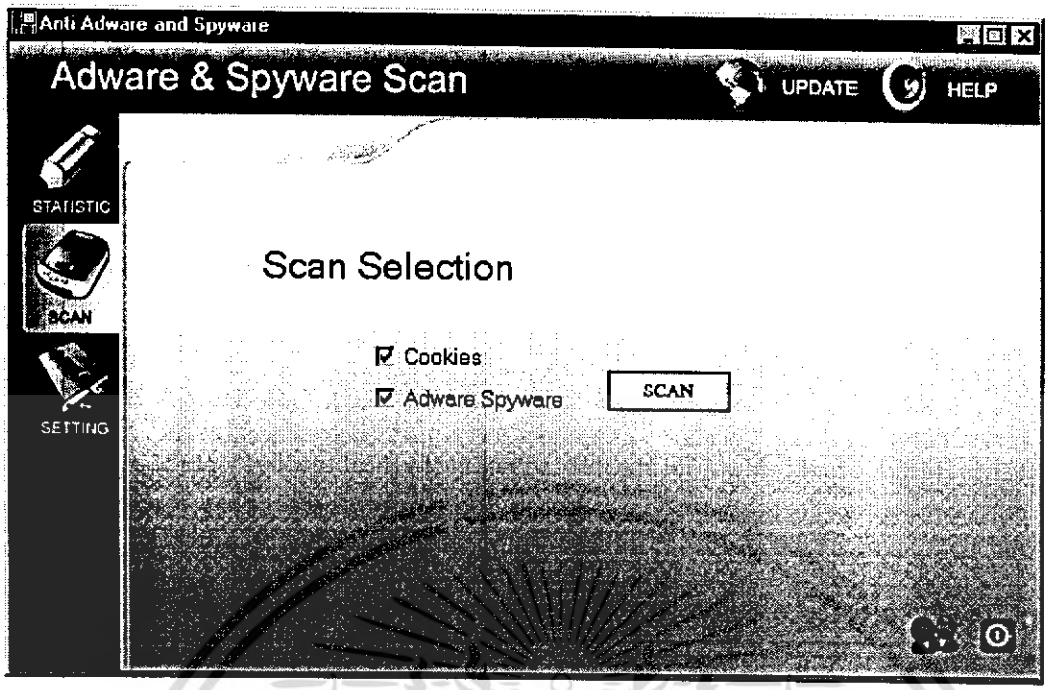


รูปที่ 4.1 หน้าต่างแสดงข้อมูลภาพรวม

4.3.2 หน้าต่างการค้นหาและทำลายแอดแวร์และสปายแวร์

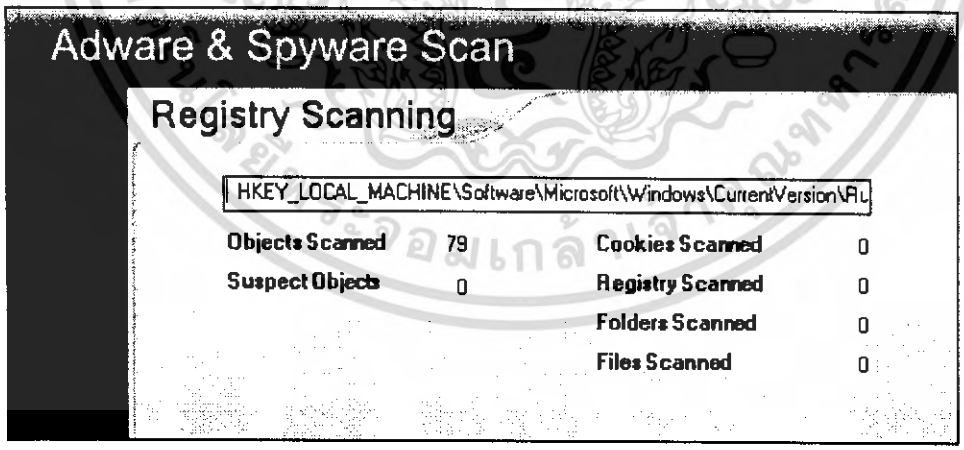
ในหน้าต่างค้นหาและกำจัดแอดแวร์และสปายแวร์เป็นส่วนที่ให้ผู้ใช้ทำการสั่งการทำงานให้โปรแกรมทำการตรวจจับแอดแวร์และสปายแวร์ โดยในหน้าต่างนี้จะเป็นการให้ผู้ใช้เลือกว่าต้องการทำการตรวจสอบส่วนใดบ้างซึ่งมี 2 ส่วนด้วยกัน คือ

1) เลือกให้โปรแกรมทำการตรวจสอบคุกกี้
2) เลือกให้โปรแกรมทำการตรวจสอบแอดแวร์และสปายแวร์



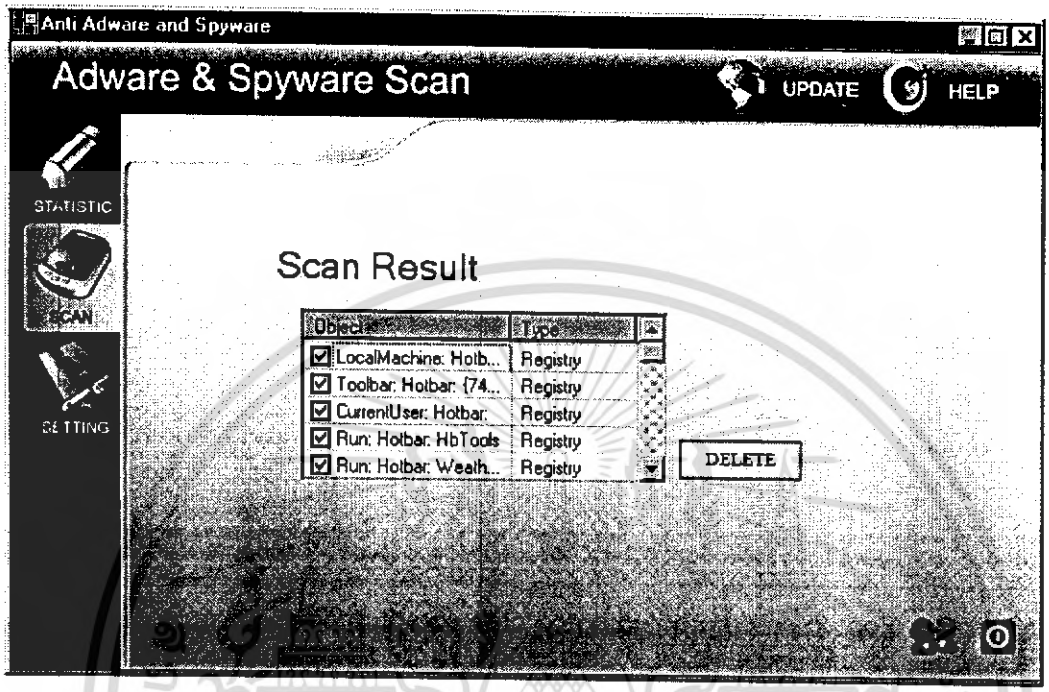
รูปที่ 4.2 หน้าต่างค้นหาและกำจัดแอดแวร์และสปายแวร์

โดยผู้ใช้หากต้องการทำการตรวจสอบประเภทใดก็ให้เลือกเฉพาะส่วนที่ต้องการทำการตรวจสอบ ซึ่งถ้าหากว่าผู้ใช้ไม่ต้องการที่จะให้ตรวจสอบส่วนใดก็ให้ผู้ใช้คลิกที่เครื่องหมายถูกออก โปรแกรมก็จะไม่ทำการตรวจสอบในส่วนนั้นแล้ว โดยเมื่อเลือกส่วนที่จะให้ตรวจสอบเรียบร้อยแล้ว ให้คลิกที่ปุ่ม SCAN หลังจากนั้นโปรแกรมจะการแสดงหน้าต่างการค้นหาโดยอัตโนมัติ



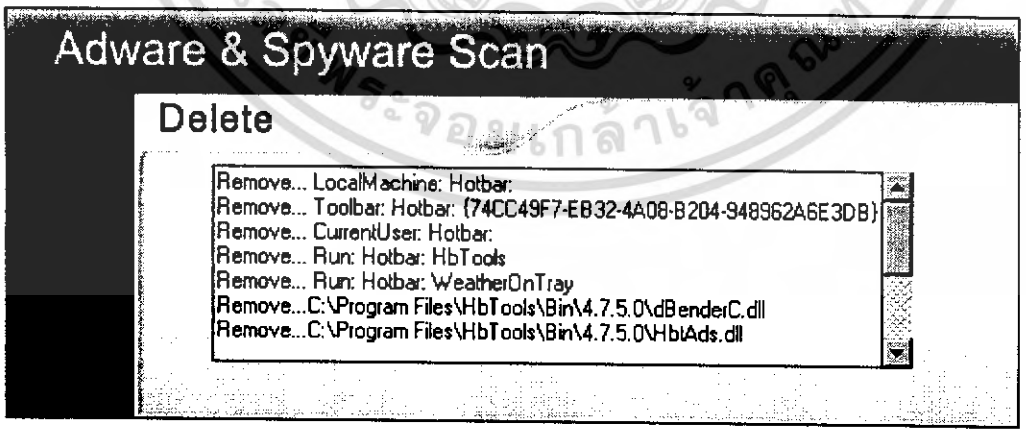
รูปที่ 4.3 หน้าต่างค้นหาแอดแวร์และสปายแวร์

หลังจากนั้นจะมีหน้าต่างแสดงผลว่าตรวจพบแอดแวร์และสปายแวร์ใดบ้าง โดยถ้าผู้ใช้สามารถทำเครื่องหมายถูกหน้าชื่อแอดแวร์และสปายแวร์เพื่อเลือกตัวที่ต้องการลบและทำการกดปุ่ม DELETE เพื่อทำการลบแอดแวร์และสปายแวร์ที่ตรวจพบ



รูปที่ 4.4 หน้าต่างแสดงแอดแวร์และสปายแวร์ที่ตรวจพบ

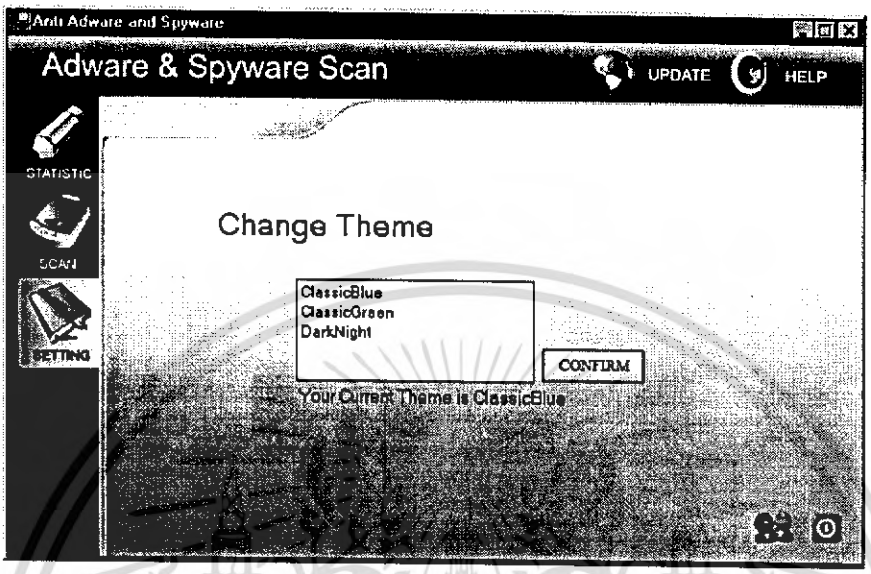
เมื่อลบทำการเลือกทำการลบ โปรแกรมจะทำการลบและทำการแสดงรายละเอียดรายงานการลบออกมา ดังภาพ



รูปที่ 4.5 หน้าต่างเลือกแอดแวร์และสปายแวร์ที่ต้องการกำจัด

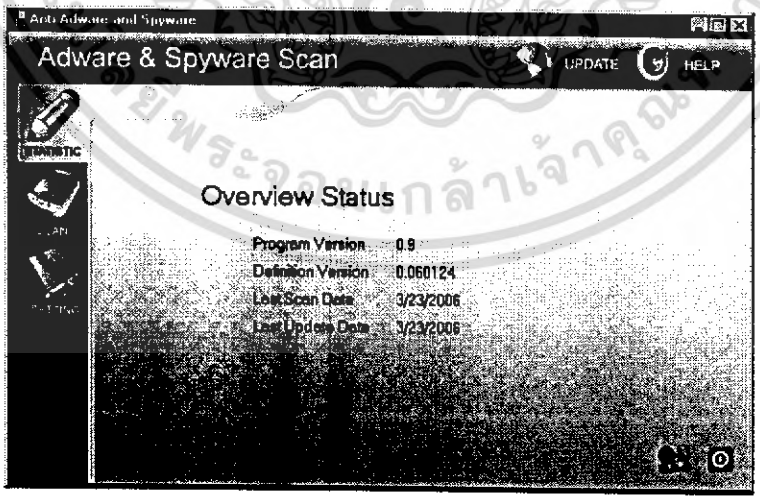
4.3.3 หน้าต่างปรับแต่งโปรแกรม

ในหน้าต่างปรับแต่งโปรแกรมนี้ เป็นส่วนที่ใช้ในการปรับเปลี่ยนคุณสมบัติตามความพึงพอใจของผู้ใช้งานโปรแกรม

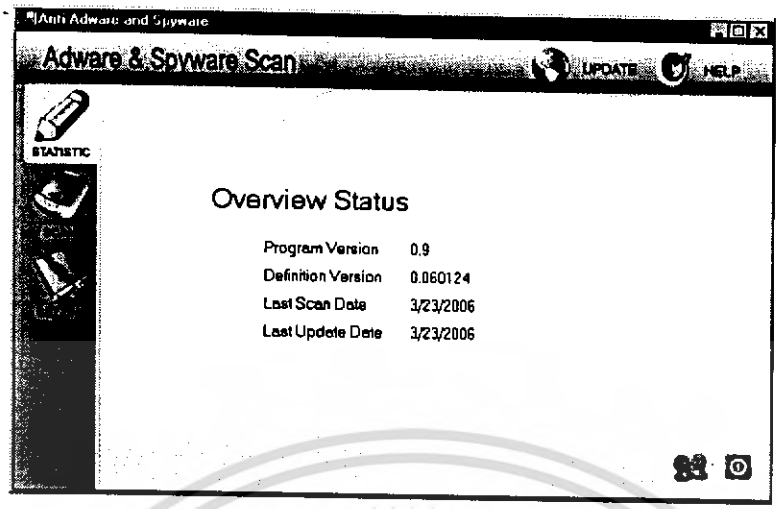


รูปที่ 4.6 หน้าต่างการปรับแต่งโปรแกรม

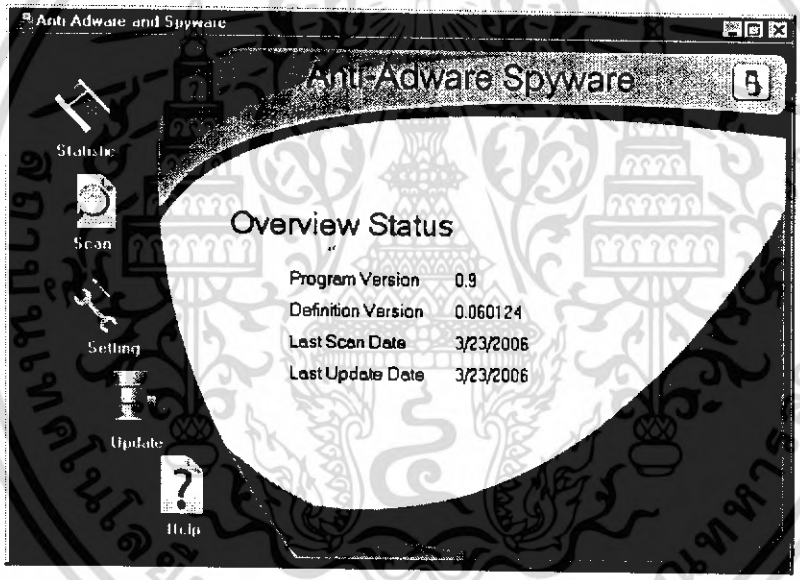
โดยการเปลี่ยนรูปแบบนั้นสามารถทำได้โดยการเลือกรูปแบบการแสดงผล และทำการกดปุ่ม CONFIRM เพื่อเปลี่ยน Theme แล้วให้ทำการรีสตาร์ทโปรแกรมนี้ โดยรูปแบบการแสดงผลนี้มีหลายรูปแบบให้เลือก โดยจะแสดงรูปแบบตัวอย่าง 3 แบบ คือ



รูปที่ 4.7 รูปแบบ ClassicBlue



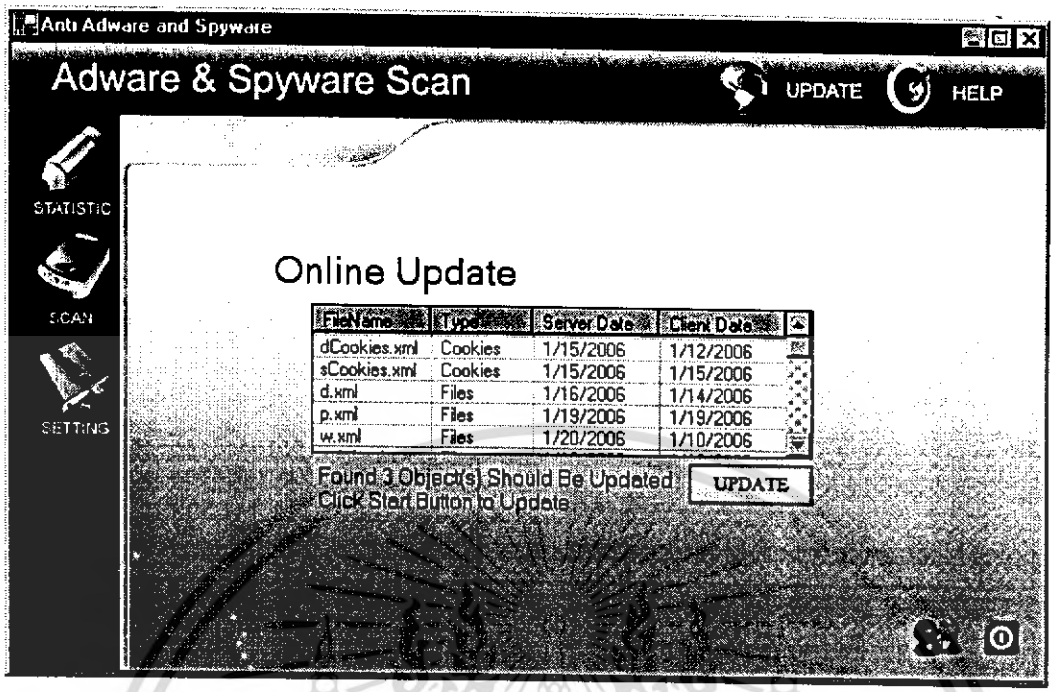
รูปที่ 4.8 รูปแบบ ClassicGreen



รูปที่ 4.9 รูปแบบ DarkNight

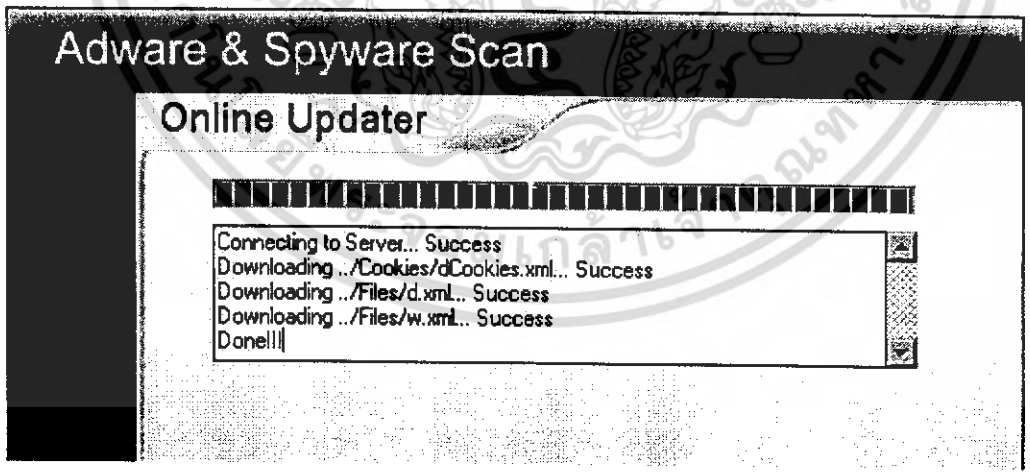
4.3.4 หน้าต่างการอัพเดตฐานข้อมูล

หน้าต่างการอัพเดตนี้เป็นหน้าต่างที่ทำการอัพเดตตัวไฟล์ที่ใช้ในโปรแกรม โดยอาจจะเป็นไฟล์ฐานข้อมูลแอดแวร์และสปายแวร์ ไฟล์ระบบ เป็นต้น โดยเมื่อผู้ใช้ที่เลือกทำการอัพเดตโปรแกรมจะทำการติดต่อกับเซิร์ฟเวอร์ว่ามีไฟล์ใดที่ต้องการอัพเดตบ้าง โดยถ้าหากผู้ใช้ไม่ได้ทำการเชื่อมต่อเครือข่าย ระบบจะทำการแจ้งเตือนผู้ใช้ แต่ถ้าหากผู้ใช้เชื่อมต่อเครือข่ายเรียบร้อยโปรแกรมจะทำการแสดงรายชื่อไฟล์ที่ต้องการอัพเดต



รูปที่ 4.10 หน้าต่างแสดงไฟล์ที่ต้องทำการอัพเดต

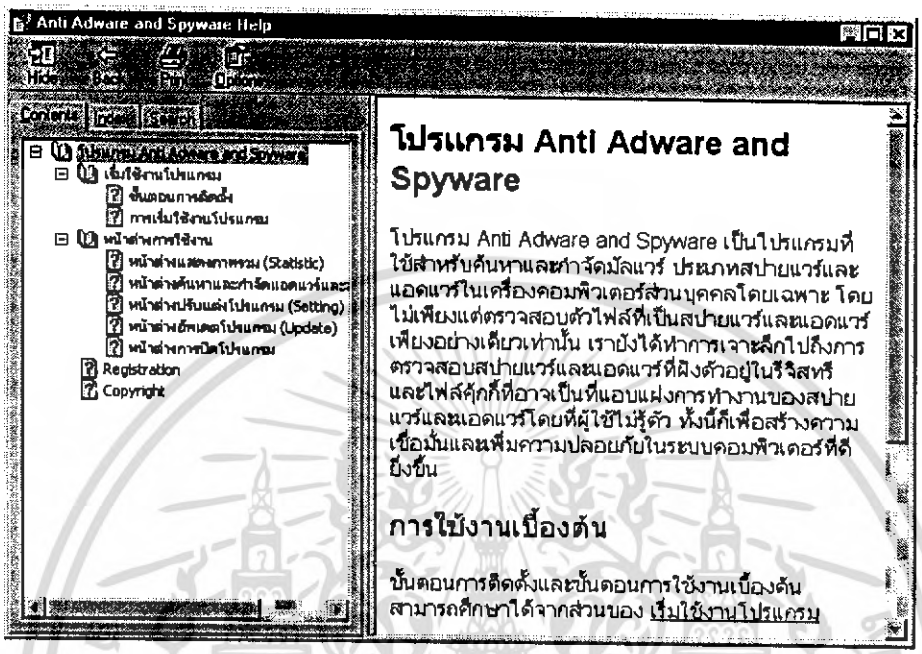
โดยหากมีไฟล์ใดที่ต้องการอัพเดต โปรแกรมจะแสดงปุ่ม UPDATE เพื่อให้ผู้ใช้งานทำการเข้าสู่หน้าจอการอัพเดตต่อไป โดยขั้นตอนการอัพเดตจะเป็นการอัพเดตไปทีละไฟล์ เมื่ออัพเดตเสร็จสิ้นจะทำการแสดงข้อความว่าเสร็จสิ้น



รูปที่ 4.11 หน้าต่างขณะอัพเดต

4.3.5 หน้าต่างการช่วยเหลือผู้ใช้งาน

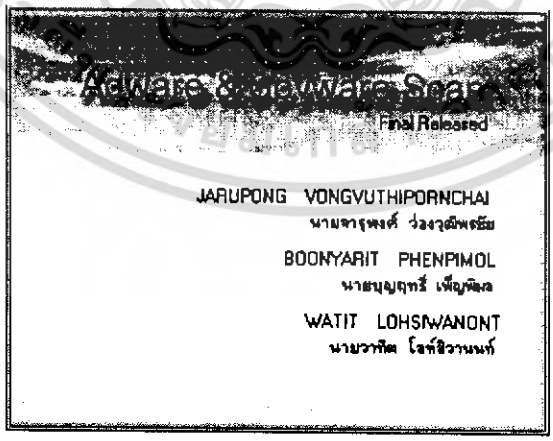
เมื่อผู้ใช้ต้องการที่จะขอความช่วยเหลือ ผู้ใช้สามารถทำการกดปุ่ม F1 ที่คีย์บอร์ด หรือทำการกดที่ปุ่ม HELP แล้วโปรแกรมจะทำการแสดงหน้าต่างการช่วยเหลือออกมาดังภาพ



รูปที่ 4.12 หน้าต่างการช่วยเหลือผู้ใช้งาน

4.3.6 หน้าต่างรายชื่อผู้จัดทำ

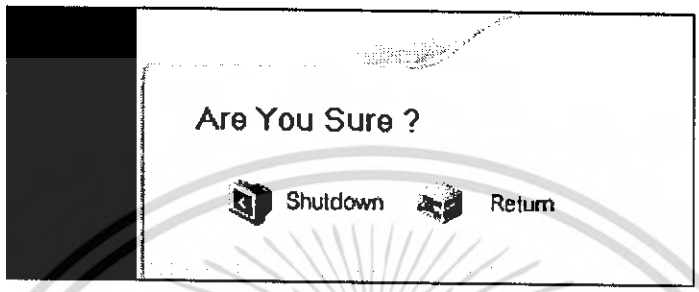
ผู้ใช้งานสามารถที่จะแสดงรายชื่อคณะผู้จัดทำได้ โดยทำการกดปุ่ม บนหน้าจอ แล้วโปรแกรมจะทำการแสดงรายชื่อผู้จัดทำ ดังภาพ



รูปที่ 4.13 หน้าต่างรายชื่อผู้จัดทำ.

4.3.7 หน้าต่างการปิดโปรแกรม

เมื่อผู้ใช้ทำงานเสร็จแล้วเรียบร้อยแล้วผู้ใช้ต้องการที่จะออกจากโปรแกรมให้ผู้ใช้กดที่ปุ่มปิดโปรแกรม แล้วโปรแกรมจะทำการแสดงหน้าต่างจะมีหน้าต่างขึ้นมาถามว่าคุณแน่ใจที่จะออกจากโปรแกรมหรือไม่ โดยถ้าต้องการออกให้เลือก Yes แต่ถ้าไม่ต้องการออกให้เลือก No โดยหน้าต่างการปิดโปรแกรม แสดงดังภาพ



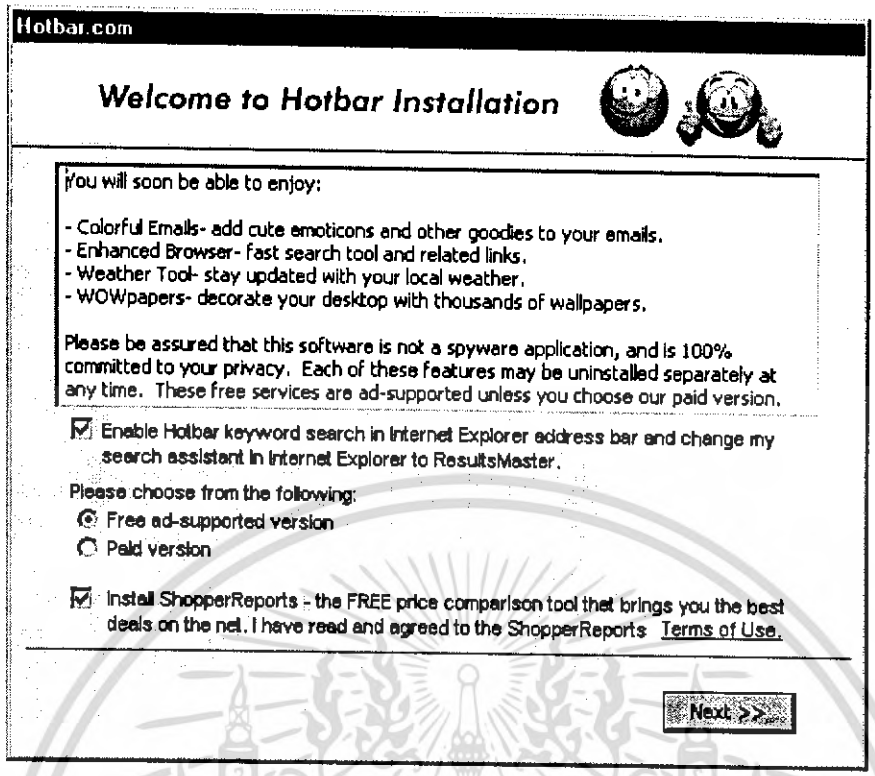
รูปที่ 4.14 หน้าต่างเลือกปิดโปรแกรม

4.4 การทดสอบโปรแกรม

การทดสอบโปรแกรม เราจะใช้โปรแกรม HotBar ซึ่งเป็นรูปแบบหนึ่งของแอดแวร์และสปายแวร์มาใช้ในการทดสอบความสามารถของโปรแกรม อันเนื่องมาจาก HotBar มีการฝังการทำงานและมีรูปแบบการทำงานแบบแอดแวร์และสปายแวร์ โดยมันจะฝังการทำงานทั้งใน คุกกี้ รีจิสตรี ไฟล์และโฟลเดอร์

4.4.1 การติดตั้งโปรแกรม HotBar

ในการติดตั้งโปรแกรม HotBar เราได้ทำการดาวน์โหลดโปรแกรม HotBar และทำการติดตั้ง ซึ่งโปรแกรมจะทำการฝังการทำงานทั้งในคุกกี้ รีจิสตรี ไฟล์และโฟลเดอร์



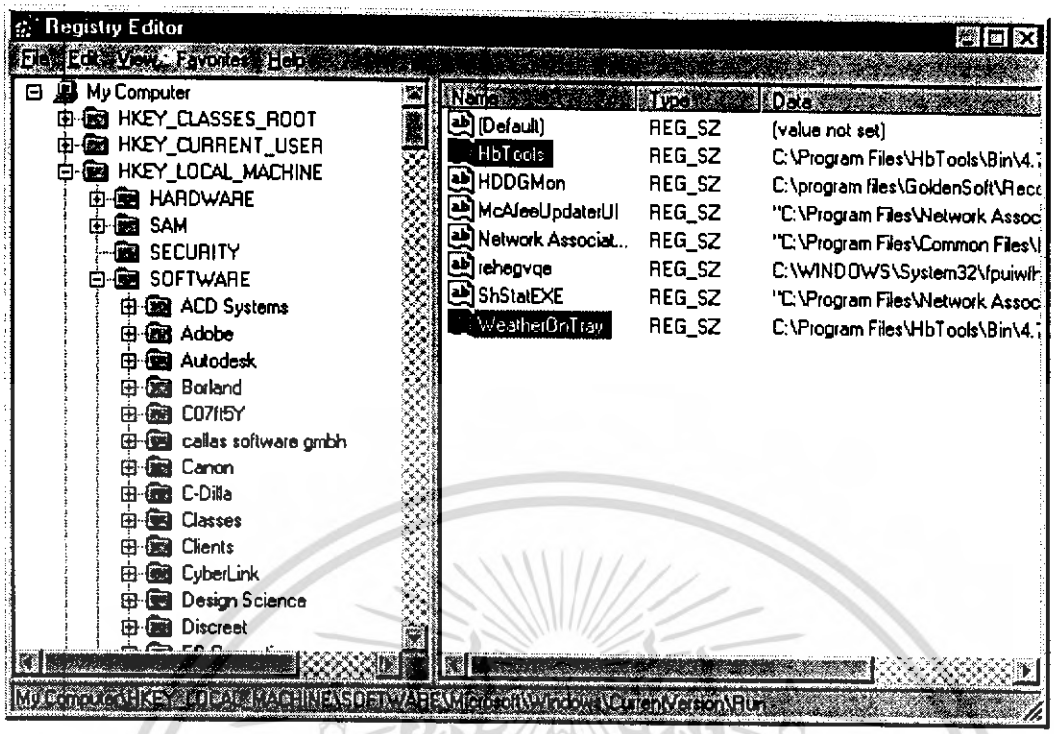
รูปที่ 4.15 ขั้นตอนการติดตั้งโปรแกรม HotBar

โดยโปรแกรม HotBar จะฝังการทำงานในส่วนต่างๆ ที่สามารถพบเห็นได้ ดังต่อไปนี้

1) การฝังการทำงานในรีจิสตรี

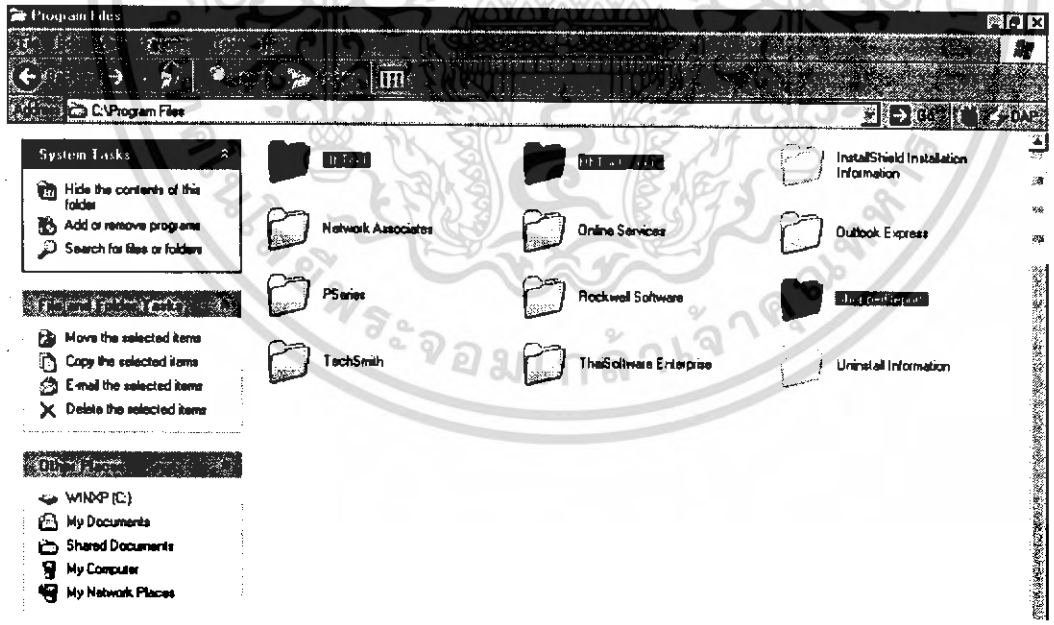
โปรแกรม HotBar จะทำการฝังค่าในรีจิสตรีไว้ ทั้งในส่วนของ HKEY_LOCAL_MACHINE, HKEY_CURRENT_USERและในส่วนของHKEY_LOCAL_MACHINE\SOFTWARE\MICROSOFT\WINDOWS\CURRENTVERSION\RUN

โดยโปรแกรม HotBar จะทำการฝังค่าในรีจิสตรีไว้ 2 ค่าคือ HbTools และ WeatherOnTray เพื่อเป็นประโยชน์ในการรันโปรแกรมต่างๆ ของ HotBar โดยมันจะฝังอยู่ใน HKEY_LOCAL_MACHINE\SOFTWARE\MICROSOFT\WINDOWS\CURRENTVERSION\RUN



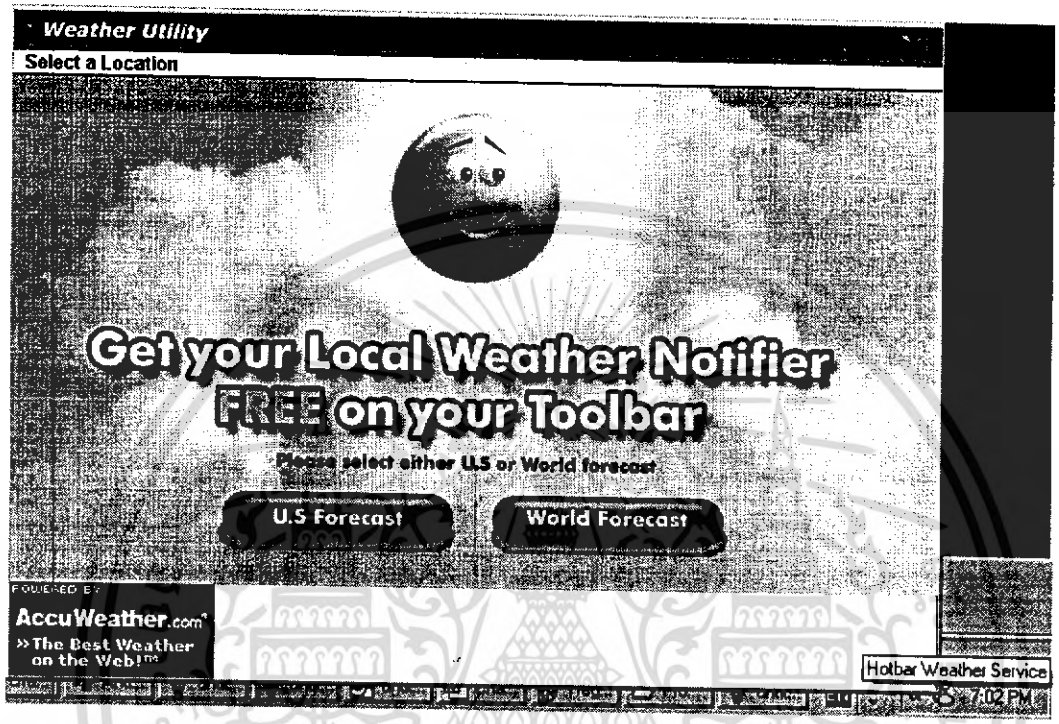
รูปที่ 4.16 ภาพแสดงการฝังการทำงานในรีจิสตรีของ HotBar

2) การฝังการทำงานในโฟลเดอร์และไฟล์



รูปที่ 4.17 ภาพแสดงการฝังการทำงานในโฟลเดอร์และไฟล์

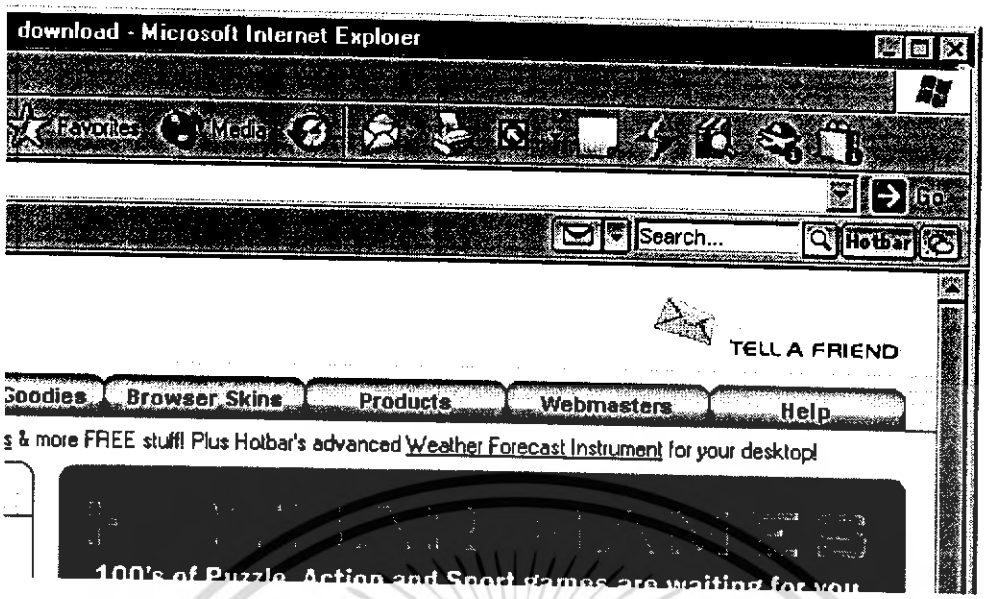
โปรแกรม HotBar จะทำการจะทำการสร้างไฟล์และโฟลเดอร์ขึ้นมา โดยจะเป็นไฟล์โปรแกรมต่างๆ ของ HotBar ซึ่งโปรแกรมต่างๆเหล่านั้นจะทำการรันโดยอัตโนมัติโดยค่าจากรีจิสตรีที่ฝังเอาไว้ก่อนหน้านั้น จากภาพด้านล่างเป็นโปรแกรม WeatherOnTray จากโปรแกรม HotBar โดยมันจะฝังการทำงานอยู่ภายใน Tray และบางครั้งจะแสดงหน้าต่างขึ้นมารบกวนผู้ใช้งาน ดังภาพ



รูปที่ 4.18 ภาพแสดงโปรแกรม WeatherOnTray ที่ถูกทำงานโดยอัตโนมัติ

3) การฝังการทำงานในไฟล์ไลบรารี DLL

ในไฟล์ไลบรารี DLL ถือเป็นไฟล์ที่มีความสำคัญเพราะเป็นไฟล์ไลบรารีที่ถูกเรียกใช้งานอยู่เสมอจากโปรแกรมทั่วไปต่างๆ โดยในโปรแกรม HotBar ได้มีการสร้างไฟล์ไลบรารี DLL ต่างๆ มากมาย โดยที่เห็นได้ชัดคือ การมีการฝังทูลบาร์บน Internet Explorer



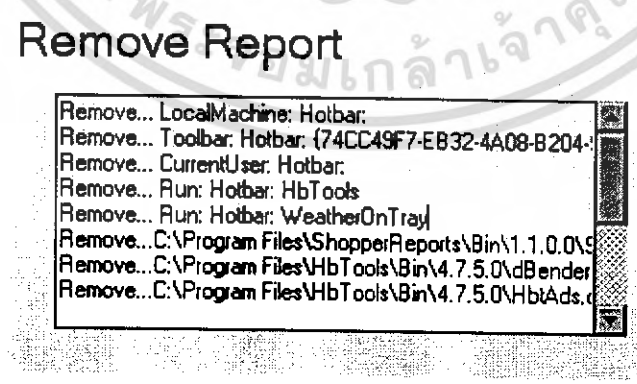
รูปที่ 4.19 ภาพแสดง HotBar ทูลบาร์ที่ถูกฝังในโปรแกรม Internet Explorer

4) การฝังการทำงานในคุกกี้

โปรแกรม HotBar นอกจากมีการฝังการทำงานในส่วนของรีจิสตรี ไฟล์และโฟลเดอร์ต่างๆ แล้ว ยังคงมีการสร้างไฟล์คุกกี้ เพื่อเก็บประวัติผู้ใช้ต่างๆ อีกด้วย

4.4.2 การทดสอบความสามารถโปรแกรมที่พัฒนา

หลังจากที่เราทำการติดตั้งโปรแกรม HotBar ซึ่งเป็นแอดแวร์และสปายแวร์เป้าหมาย เราก็จะทำการรันโปรแกรมค้นหาและทำลายแอดแวร์และสปายแวร์ที่เราพัฒนา แล้วจะทำการสแกนโดยภายหลังการทดสอบโปรแกรมทำการให้โปรแกรมทำการค้นหาและทำลายแอดแวร์และสปายแวร์ โดยเราพบว่าเราสามารถตรวจพบวัตถุเป้าหมาย 13 ตำแหน่งด้วยกัน ดังนี้



รูปที่ 4.20 ภาพแสดงรายงานไฟล์ที่ถูกตรวจพบ

โดยรายชื่อวัตถุที่พบทั้งหมด แสดงได้ดังนี้

- LocalMachine: Hotbar
- Toolbar: Hotbar: {74CC49F7-EB32-4A08-B204-948962A6E3DB}
- CurrentUser: Hotbar
- Run: Hotbar: HbTools
- Run: Hotbar: WeatherOnTray
- C:\Program Files\ShopperReports\Bin\1.1.0.0\ShprRprt.dll
- C:\Program Files\HbTools\Bin\4.7.5.0\dBenderC.dll
- C:\Program Files\HbTools\Bin\4.7.5.0\HbtAds.dll
- C:\Program Files\HbTools\Bin\4.7.5.0\HbtHostOL.dll
- C:\Program Files\HbTools\Bin\4.7.5.0\HbtInstIE.dll
- C:\Program Files\HbTools\Bin\4.7.5.0\HbtWallpaper.dll
- C:\Program Files\ShopperReports/cs
- C:\Program Files\ShopperReports/bin
- C:\Program Files\ShopperReports

โดยจากการทดลอง เราสามารถทำการกำจัดไฟล์ต่างๆเหล่านั้นได้ทั้งหมด โดยเราพบว่าค่าในรีจิสตรี ทูลบาร์และโปรแกรมที่ฝังอยู่ ได้ถูกกำจัดออกไปจากเครื่องแล้ว

รูปที่ 4.21 ภาพแสดง Internet Explorer และ Task Bar ภายหลังทำการทำลายสปายแวร์

4.5 สรุปผลการทำงาน

จากการทำการทดลองเราพบว่า เราสามารถที่จะกำจัดการทำงานส่วนต่างๆ ของ HotBar ได้อย่างเต็มความสามารถก็ต่อเมื่อเรามีฐานข้อมูลที่ทันสมัย เพราะวัตถุที่เราตรวจพบและทำลายได้นั้น จะเป็นไฟล์ที่มีข้อมูลอยู่ในฐานข้อมูลเท่านั้น และเราพบว่าลักษณะของแอดแวร์และสปายแวร์นั้นมักไม่มีแพร่กระจาย ดังนั้น เราจึงสามารถตรวจจับไฟล์แอดแวร์และสปายแวร์ได้โดยดูจากจุดที่แอดแวร์และสปายแวร์นั้นฝังอยู่เป็นประจำได้

โดยปัญหาสำคัญที่เรามักพบคือ แอดแวร์และสปายแวร์มักมีการฝังการทำงานที่ซับซ้อนและมีการทำงานประกอบกันหลายส่วนเพื่อให้แอดแวร์และสปายแวร์เหล่านั้นสามารถที่จะฝังการทำงานกับโปรแกรมที่ผู้ใช้ใช้อยู่เป็นประจำและสามารถที่จะทำงานได้โดยอัตโนมัติ ทำให้การที่จะลบไฟล์หนึ่งๆ ได้นั้น มักมีปัญหาที่ไม่คาดคิดอยู่เสมอ โดยเฉพาะอย่างยิ่งการแก้ไขค่ารีจิสตรีและการทำงานร่วมกับโปรเซสบนระบบปฏิบัติการ แต่อย่างไรก็ตามโปรแกรมนี้เป็นเพียงแนวทางเริ่มต้นเท่านั้น ซึ่งเราควรต้องมีการปรับปรุงเพิ่มเติม เพื่อรองรับแอดแวร์และสปายแวร์ที่กำลังเพิ่มขึ้นอย่างรวดเร็วในปัจจุบันต่อไป

บทที่ 5
สรุปผลและข้อเสนอแนะ

5.1 สรุปผลปัญหาพิเศษ

จากการศึกษาปัญหาพิเศษการพัฒนาระบบค้นหาและทำลายแอดแวร์และสปายแวร์ในรูปของโปรแกรมที่ทำงานบนเครื่องคอมพิวเตอร์ส่วนบุคคลร่วมกับระบบปฏิบัติการไมโครซอฟต์วินโดวส์ XP นั้น มีความสามารถหลักดังนี้

5.1.1 โปรแกรมที่พัฒนาสามารถตรวจจับคุ้กกี้ แอดแวร์และสปายแวร์ ไฟล์เอกสาร ไฟล์ไลบรารี DLL และจากค่ารีจิสตรีได้

5.1.2 โปรแกรมมีความสามารถ ในการยับยั้งการทำงานของโปรแกรมแอดแวร์และสปายแวร์โดยการกำจัดแอดแวร์และสปายแวร์เหล่านั้นเมื่อตรวจพบ

5.1.3 โปรแกรมสามารถที่จะทำการค้นหาและทำลายแอดแวร์และสปายแวร์ตัวใหม่ๆ ได้

5.1.4 โปรแกรมสามารถที่จะทำการอัพเดตฐานข้อมูล แอดแวร์และสปายแวร์จากเครือข่ายอินเตอร์เน็ตได้

5.1.5 โปรแกรมสามารถทำการจัดการการเชื่อมต่อเครือข่ายผ่านโปรโตคอล FTP เพื่ออัพเดตฐานข้อมูลได้ โดยที่ผู้ใช้ไม่ต้องจัดการการเชื่อมต่อเอง

5.2 ข้อจำกัดปัญหาพิเศษ

ปัญหาพิเศษการพัฒนาระบบค้นหาและทำลายแอดแวร์และสปายแวร์ มีข้อจำกัดในการทำงาน ดังนี้

5.2.1 โปรแกรมสามารถทำหน้าที่ค้นหาและทำลายแอดแวร์และสปายแวร์บนเครื่องคอมพิวเตอร์แบบสแตนอโลนเท่านั้น

5.2.2 โปรแกรมไม่สามารถที่จะเรียนรู้แอดแวร์และสปายแวร์ตัวใหม่ๆ ได้ด้วยตนเอง จำเป็นต้องมีการอัพเดตข้อมูลจากอินเตอร์เน็ต

5.2.3 โปรแกรมที่พัฒนาขึ้นสามารถใช้งานได้บนระบบปฏิบัติการที่สนับสนุนไมโครซอฟต์ดอทเน็ตเฟรมเวิร์คเท่านั้น เช่น ระบบปฏิบัติการไมโครซอฟต์วินโดวส์ 2000 และไมโครซอฟต์วินโดวส์ XP

5.3 ข้อเสนอแนะและแนวทางการศึกษาต่อ

5.3.1 โปรแกรมควรที่จะสามารถรองรับการทำงานบนหลายระบบปฏิบัติการ ไม่ควรที่จะรองรับเฉพาะระบบปฏิบัติการที่สนับสนุนไมโครซอฟต์ดอทเน็ตเฟรมเวิร์คเท่านั้น

5.3.2 ด้านข้อมูลแอดแวร์และสปายแวร์ ควรจะมีโปรแกรมเสริมที่ทำหน้าที่แปลงรูปแบบข้อมูลของแอดแวร์และสปายแวร์จากแหล่งข้อมูลอื่นๆ ให้อยู่ในรูปแบบข้อมูลที่ระบบเข้าใจ เพื่อที่จะรวบรวมข้อมูลจากแหล่งข้อมูลเหล่านั้นให้นำมาใช้ในโปรแกรมที่ถูกพัฒนาได้

5.3.3 โปรแกรมควรที่จะสามารถทำการค้นหาและทำลายแอดแวร์และสปายแวร์ผ่านทางเครือข่ายได้

บรรณานุกรม

หนังสือ

[1] เรืองไกร รังสิพล, "เปิดโลก firewall ฉบับสมบูรณ์", โปรวิชั่น จำกัด, 2544

[2] ค็อกซ์ ฟิลลิป, "คู่มือความปลอดภัยบน Windows 2000", ซีเอ็ดยูเคชั่น, 2545

[3] สุวัฒน์ ปุณณชัยยะ, ต้น ตัณฑ์สุทธิวงศ์, สุพจน์ ปุณณชัยยะ, "เปิดโลก TCP/IP และโปรโตคอลของอินเตอร์เน็ต", 2nd Edition, โปรวิชั่น

[4] สแคมเบรย์ โจเอล, "Hacking Exposed ปิดทางแฮกเกอร์", ซีเอ็ดยูเคชั่น, 2544

[5] บริษัท อัปเปอร์ แมเนจเมนท์ เอ็กซ์เซลเลนซ์ จำกัด, "Visual Basic 6.0 Win32 API เทคนิคและการประยุกต์", 2543

[6] พิเชษฐ์ ศิริรัตนไพศาลกุล, "ระบบปฏิบัติการ", ซีเอ็ดยูเคชั่น, 2546

[7] กมลมาศ กำจรกิจการ, "คู่มือ Borland Delphi 5 ฉบับสมบูรณ์", โปรวิชั่น จำกัด, 2543

[8] เรืองรัตน์ กรุงทองพัฒนา "Windows NT Registry ความลับที่ไม่ลับของ Windows NT", ซีเอ็ดยูเคชั่น, 2543

[9] สุวัฒนา สุขสมจินตน์, "คัมภีร์การใช้ XML ฉบับสมบูรณ์", ซีเอ็ดยูเคชั่น, 2545

บทความ

[1] สาร NECTEC ประจำเดือนพฤศจิกายน - ธันวาคม 2547

[2] กิตติศักดิ์ จิรวรรณกูล, "โปรแกรมป้องกันไวรัสทำงานกันอย่างไร", http://thaicert.nectec.or.th/paper/virus/antivirus1.php, เผยแพร่เมื่อ : 27มกราคม 2546