โปรแกรมอำนวยความสะดวกในการจัดการไวรัสสำหรับผู้ดูแลระบบ

Computer Virus Managing Program for Network System Administrator

นายจักเรศ จับแสงจันทร์ รหัส 43360395

นายเทพดล เกตุสุวรรณ รหัส 43360734

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2547

ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน	โปรแกรมอำนวยความสะดวกในการจัดการไวรัสสำหรับผู้ดูแลระบบ	
ผู้ดำเนินโครงงาน	นายจักเรศ จับแสงจันทร์	รหัสประจำตัว 43360395
	นายเทพดล เกตุสุวรรณ	รหัสประจำตัว 43360734
อาจารย์ที่ปรึกษา	อาจารย์ภาณุพงศ์ สอนคม	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์	
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์	
ปีการศึกษา	2547	

.....

คณะวิศวกรรมศาสตร์มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะกรรมการสอบโครงงานวิศวกรรม

.....ประธานกรรมการ
(อาจารย์ภาณุพงศ์ สอนคม)

.....กรรมการ
(อาจารย์สุรเดช จิตประไพกุลศาล)

.....กรรมการ
(อาจารย์แสงชัย มังกรทอง)

หัวข้อโครงงาน	โปรแกรมอำนวยความสะดวกในการจัดการไวรัสสำหรับผู้ดูแลระบบ		
ผู้ดำเนินโครงงาน	นายจักเรศ จับแสงจันทร์	รหัสประจำตัว	43360395
	นายเทพดล เกตุสุวรรณ	รหัสประจำตัว	43360734
อาจารย์ที่ปรึกษา	อาจารย์ภาณุพงศ์ สอนคม		
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2547		

บทคัดย่อ

โครงงานนี้จัดทำขึ้นเพื่ออำนวยความสะดวกให้กับผู้ดูแลระบบ เครือข่ายคอมพิวเตอร์ในการจัดการปัญหาเกี่ยวกับไวรัสคอมพิวเตอร์ เพื่อให้ระบบเครือข่ายทำงานได้อย่างมีประสิทธิภาพ โดยการปฏิบัติการทั้งหมด ผู้ดูแลระบบสามารถสั่งงานได้จากเครื่องของผู้ดูแลระบบเอง เช่น การตรวจสอบไวรัส การปรับปรุงโปรแกรมต่อต้านไวรัส รวมไปถึงการสั่งปิดเครื่องลูกข่าย เป็นต้น นอกจากนี้ ยังสามารถตรวจสอบผลการปฏิบัติงานย้อนหลังได้ ซึ่งเป็นการลดขั้นตอนการปฏิบัติงานให้กับผู้ดูแลระบบในการจัดการไวรัสคอมพิวเตอร์ในระบบเครือข่าย

จากผลการทดสอบโปรแกรมอำนวยความสะดวกให้กับผู้ดูแลระบบ ในการจัดการปัญหาด้านไวรัสคอมพิวเตอร์ พบว่าโปรแกรมสามารถทำงานได้ตามวัตถุประสงค์

Project Title	Computer Virus Managing Program for Network System Administrator
Name	Mr. Chakkarate Chubsangchan ID. 43360395
	Mr. Theppadol Getsuwan ID. 43360734
Project Advisor	Mr. Panupong Sonkhom
Major	Computer Engineering
Department	Electrical and Computer Engineering
Academic Year	2004

ABSTRACT

The purpose of this project is to facilitate a network system administrator to deal with computer virus problem in order to manage a network system efficiently. For all operations, the network system administrator is able to operate tasks from his own computer or the host computer such as virus inspection and anti virus program adjustment and also to shut down other users or the client. Moreover, the administrator is able to examine the previous operations. So that it is the way to reduce the process of operation for administrator to manage the computer viruses in the network system.

The outcome of the test on Computer Virus Managing Program for Network System Administrator is that the program can act properly.

กิตติกรรมประกาศ

โครงงานฉบับนี้สำเร็จลงได้ด้วยความอนุเคราะห์อันดียิ่งจาก อาจารย์ภาณุพงศ์ สอนคม ที่ได้กรุณามาเป็นอาจารย์ที่ปรึกษา ซึ่งให้แนวคิดและความช่วยเหลือตลอดจนสละเวลาอันมีค่าเพื่อตรวจทานแก้ไขข้อบกพร่องต่าง ๆจนกระทั่งงานสำเร็จลุล่วงไปได้ด้วยดี นอกจากนี้ยังได้รับความอนุเคราะห์จากอาจารย์สุรเดช จิตประไพกุลศาล ที่ได้สละเวลาอันมีค่าในการร่วมตรวจทานแก้ไขข้อบกพร่องต่าง ๆ รวมไปถึงให้ความช่วยเหลือในอีกหลาย ๆด้าน ทางผู้จัดทำขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ทั้งนี้ขอกราบขอบพระคุณอาจารย์ทุกท่านที่ได้อบรมสั่งสอนให้ความรู้ คำแนะนำและความช่วยเหลือเสมอมา รวมทั้งขอขอบพระคุณเพื่อน ๆพี่ ๆทุกท่านที่ได้ให้ความช่วยเหลืออย่างเต็มที่ไม่ว่าด้านการจัดทำโปรแกรมหรือการจัดทำเอกสาร

ท้ายนี้ขอกราบขอบพระคุณ บิดา-มารดา ผู้ที่ทุ่มเททั้งชีวิตและจิตใจเพื่อให้การศึกษาที่ดีที่สุดและสิ่งต่าง ๆที่ดีที่สุดแก่ผู้จัดทำ และคอยดูความสำเร็จในแต่ละก้าวย่างของผู้จัดทำนอกจากนี้ยังคอยห่วงใยและให้กำลังใจเสมอมา จนสามารถทำงานครั้งนี้จนสำเร็จลุล่วงด้วยดี

นายจักเรศ จับแสงจันทร์
นายเทพดล เกตุสุวรรณ

สารบัญ

หน้า

สารบัญ(ต่อ)

หน้า

สารบัญ(ต่อ)

หน้า

สารบัญรูป

รูปที่ หน้า

สารบัญรูป(ต่อ)

รูปที่ หน้า

สารบัญรูป(ต่อ)

สารบัญตาราง

ตารางที่ หน้า

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

เนื่องจากองค์กรต่างๆ มีระบบเครือข่ายคอมพิวเตอร์มีจำนวนมากขึ้น ซึ่งขนาดของระบบเครือข่ายจะขนาดใหญ่และซับซ้อนเพียงใดย่อมขึ้นอยู่กับขนาดขององค์กร การบำรุงและดูแลรักษาระบบเครือข่ายให้เป็นปกติอยู่เสมอ เป็นหน้าที่ของผู้ดูแลระบบ ในการทำงานย่อมมีปัญหาเกี่ยวกับการดูแลระบบหลากหลาย และหนึ่งในหลาย ๆ ปัญหาเกิดมาจากไวรัสคอมพิวเตอร์ อาจเป็นผลมาจากการเข้าเว็บไซต์หรือการดาวน์โหลดโปรแกรมต่างๆ ซึ่งเจ้าของเว็บไซต์หรือโปรแกรมนั้น ๆ ประสงค์จะสร้างความเดือดร้อนรำคาญจนกระทั่งสร้างความเสียหายแก่เครื่องที่เป็นเป้าหมาย บางครั้งปัญหาอาจลุกลามสร้างความเสียหายกับระบบเครือข่ายก็เป็นได้

การจัดการปัญหาดังกล่าวอาจทำให้ผู้ดูแลระบบใช้เวลาจัดการเป็นเวลานาน หากเป็นองค์กรขนาดใหญ่ที่มีเครื่องลูกข่ายมากมายด้วยแล้ว การจัดการปัญหาจำต้องใช้เวลามาก เพื่อลดขั้นตอนการทำงานของผู้ดูแลระบบในการจัดการปัญหาด้านไวรัสคอมพิวเตอร์ในระบบเครือข่าย ผู้ดูแลระบบควรมีความสามารถในการสั่งการจากเครื่องของผู้ดูแลระบบเอง และส่งคำสั่งที่ช่วยในการจัดการปัญหาแก่เครื่องในระบบเครือข่าย ครั้งละหลายๆเครื่อง และสามารถตรวจสอบรายงานที่ได้จากปฏิบัติงานได้ จากคุณสมบัติดังกล่าวจะสามารถช่วยลดขั้นตอนการทำงานให้ผู้ดูแลระบบ และจัดการปัญหาด้านไวรัสคอมพิวเตอร์ได้อย่างทั่วถึงอีกด้วย

1.2 วัตถุประสงค์ของโครงงาน

1. สร้างโปรแกรมที่อำนวยความสะดวกและลดขั้นตอนการปฏิบัติงานในการตรวจสอบและการจัดการไวรัสในระบบเครือข่าย ให้แก่ผู้ดูแลระบบได้ โดยผู้ดูแลระบบสามารถสั่งการได้จากเครื่องของผู้ดูแลระบบเอง

2. สามารถลดการแพร่กระจายของไวรัสคอมพิวเตอร์ภายในระบบเครือข่ายได้ โดยทำการตรวจสอบพร้อมทั้งจัดการปัญหาไวรัสคอมพิวเตอร์ในขั้นตอนเดียวกัน

1.3 ขอบข่ายของโครงงาน

1. สามารถตรวจสอบและจัดการกับไวรัสคอมพิวเตอร์ในระบบเครือข่าย
2. สามารถทำการปรับปรุงโปรแกรมต่อต้านไวรัส (Update Anti-Virus) ทั้งระบบเครือข่าย
3. สามารถรายงานผลของการตรวจสอบออกทางหน้าจอของผู้ดูแลระบบ

1.4 กิจกรรมดำเนินงาน

กิจกรรม	ปี 2546		ปี 2547									
	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.	พ.ค.	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.
เริ่มศึกษาและวางแผนการทำงาน	←	—	→									
ศึกษาระบบโปรแกรม Anti-Virus	←	—	—	—	→							
ศึกษาทฤษฎีที่เกี่ยวข้อง	←	—	—	—	—	→						
ศึกษาระบบเน็ตเวิร์ค			←	—	—	—	→					
ศึกษาแอพพลิเคชั่นที่เกี่ยวข้อง			←	—	—	→	→					
สร้างโปรแกรมจากระบบ			←	—	—	—	—	—	—	→		
ทดลอง ตรวจสอบ แก้ไขโปรแกรมจากปัญหา							←	—	—	→		
จัดทำเอกสาร			←	—	—	—	—	—	—	—	→	
ส่งโครงงานฉบับสมบูรณ์											←	→

1.5 ผลที่คาดว่าจะได้รับ

1. สามารถสร้างโปรแกรมอำนวยความสะดวกและลดขั้นตอนการทำงานสำหรับผู้ดูแลระบบในด้านการตรวจสอบและการจัดการไวรัสในระบบเครือข่ายและทำงานได้อย่างมีประสิทธิภาพ

2. ลดการแพร่กระจายของไวรัสคอมพิวเตอร์ในระบบเครือข่าย

บทที่ 2
หลักการและทฤษฎีที่เกี่ยวข้อง

ในปัจจุบันการสื่อสารผ่านระบบเครือข่าย (Network) รวมทั้งระบบอินเตอร์เน็ต เป็นที่แพร่หลายมากในแง่ของการใช้งาน, ใช้บริการ, อำนวยความสะดวกในชีวิตประจำวันต่างๆเนื่องจากระบบสื่อสารที่รวดเร็วและสามารถเชื่อมโยงเข้ากับหลาย ๆระบบได้ ทำให้สามารถนำมาประยุกต์ใช้งานผ่านระบบเครือข่ายได้หลากหลายรูปแบบ ซึ่งจะนำไปประยุกต์ใช้ในการควบคุมอุปกรณ์หรือโปรแกรมต่าง ๆ

เนื้อหาในบทนี้จะกล่าวถึงหลักการและทฤษฎีต่างๆ ที่สามารถนำมาประยุกต์ต่อการสร้างโปรแกรมที่อำนวยความสะดวกด้านการลดขั้นตอนการทำงานให้กับผู้ดูแลระบบ ในการจัดการปัญหาเกี่ยวกับไวรัสคอมพิวเตอร์

2.1 ความหมายของไวรัสคอมพิวเตอร์และโปรแกรมต่อต้านไวรัส

ไวรัสคอมพิวเตอร์ เป็นโปรแกรมที่ผู้เขียนโปรแกรมสร้างขึ้นมีเป้าหมายเพื่อสร้างความเดือดร้อนรำคาญแก่เครื่องที่เป็นเป้าหมาย เช่น ทำลายข้อมูลสำคัญหรือการแสดงข้อความขึ้นบนหน้าจออยู่ตลอดเวลา เป็นต้น โดยจะมีพฤติกรรมในการจำลองตัวเองไปยังไฟล์อื่น ๆและแพร่ กระจายไปยังไฟล์ข้างเคียงจนถึงข้อมูลทั้งหมด หากมีโอกาสจะสามารถแทรกเข้าไปประบาดสู่คอมพิวเตอร์เครื่องอื่นในระบบเครือข่าย การแพร่กระจายของไวรัสคอมพิวเตอร์นั้นอาจเกิดจากการนำดิสก์ที่ติดไวรัสจากเครื่องหนึ่งไปใช้ยังอีกเครื่องหนึ่ง หรืออาจเป็นการส่งผ่านข้อมูลที่ติดไวรัสทางระบบเครือข่ายรวมไปถึงข้อความในจดหมายอิเล็กทรอนิกส์ (E-mail) ที่แนบไฟล์ไวรัสและจะติดทันทีเมื่อเปิดอ่านเป็นต้น

โปรแกรมต่อต้านไวรัส (Anti-Virus) เป็นโปรแกรมที่เขียนขึ้นเพื่อกำจัดไวรัสที่แฝงตัวอยู่ในเครื่องคอมพิวเตอร์โดยโปรแกรมจะทำการตรวจสอบ (Scan) ข้อมูลภายในฮาร์ดดิสก์ เมื่อพบจะทำการกำจัดข้อมูลส่วนนั้นทิ้งไป ประสิทธิภาพการกำจัดไวรัสหรือไม่ก็ขึ้นอยู่กับโปรแกรมต่อต้านไวรัสว่าจะสามารถรู้จักไวรัสใหม่ ๆหรือพฤติกรรมของไวรัส ทั้งนี้ควรมีการปรับปรุงโปรแกรมต่อต้านไวรัส (Update Anti-Virus) อยู่เสมอด้วย

2.2 ความหมายของการควบคุมระยะไกล (Remote Control)

การควบคุมระยะไกล หมายถึง การเข้าควบคุมคอมพิวเตอร์จากระยะไกล โดยเราจะต้องมีคอมพิวเตอร์อย่างน้อย 2 เครื่องขึ้นไป ที่เชื่อมต่อกันอยู่

โปรแกรม Remote Administrator เป็นโปรแกรมควบคุมคอมพิวเตอร์ระยะไกลที่เชื่อมต่อกันอยู่โดยติดต่อผ่านทางโปรโตคอล TCP/IP ประกอบไปด้วย 2 ส่วนคือ

1. เครื่องลูกข่าย (Agent) คือเครื่องที่อยู่ห่างไกลที่จะถูกควบคุมโดยตัวควบคุม (Controller)
2. ตัวควบคุม (Controller) คือเครื่องที่จะส่งคำร้องขอไปที่เครื่องลูกข่ายเพื่อให้คำร้องขอนั้นไปใช้ในการปฏิบัติการของระบบ

2.3 เซิร์ฟเวอร์ (Server) และ ไคลเอนต์ (Client)

ไคลเอนต์จะเป็นเครื่องคอมพิวเตอร์ที่จะร้องขอบริการ หรือ ข้อมูล จากเครื่องเซิร์ฟเวอร์ โดยผ่านทางโปรแกรมต่างๆ เช่น Web Browser, Outlook Express, mIRC ซึ่งเครื่องเซิร์ฟเวอร์ จะเป็นเครื่องที่ให้บริการต่างๆแก่เครื่องที่เป็น Client เช่น Web Server, Mail Server, IRC Server โดยมากโปรแกรมบนอินเทอร์เน็ตจะถูกติดตั้งลงในเครื่องที่เป็นไคลเอนต์ ทำให้เครื่องที่เป็น ไคลเอนต์ มีความสามารถในการ รับและส่ง อีเมล์, เปิดชมเว็บไซต์, ทำการติดต่อสื่อสารกับกลุ่มข่าว (Newsgroups) ต่าง ๆ และสามารถโหลดไฟล์ที่ต้องการ ส่วนโปรแกรมอีกประเภทหนึ่ง จะต้องติดตั้งกับเครื่องที่เป็นเซิร์ฟเวอร์ โปรแกรมประเภทนี้จะให้บริการ แก่ไคลเอนต์ ที่ร้องขอใช้บริการ เครื่องเซิร์ฟเวอร์สามารถที่จัดการกับเครื่องไคลเอนต์ได้หลายเครื่อง และ พร้อมกันนั้นก็ยังสามารถจัดการกับงานต่างๆที่อยู่บนเครื่องไปพร้อมกัน เพราะว่าเทคโนโลยี Socket ที่มีใช้งานในปัจจุบันบนอินเทอร์เน็ตมีความเสถียรภาพ Socket ที่มีการใช้งานอยู่บน Windows จะถูกเรียกว่า Windows Socket หรือจะเรียกสั้นๆว่า Winsock ความหมายของ Socket คือ เครื่องมือของโปรแกรมที่จะถูกใช้ในการในการส่ง และ รับ ข้อมูลผ่านทางหมายเลขพอร์ทของ TCP/IP ที่กำหนด โปรแกรมจะสร้าง Socket ได้ตามที่ต้องการเพื่อใช้ในการทำงาน แต่ 1 Socket จะต้องทำงานกับ 1 พอร์ทของ TCP/IP เท่านั้น โปรแกรมฝั่งไคลเอนต์จะสร้าง Socket และทำการกำหนดหมายเลขพอร์ตโดยวิธีการสุ่มหมายเลขขึ้นมา แต่ทางฝั่งเซิร์ฟเวอร์จะไม่เป็นอย่างนั้น โปรแกรมฝั่งเซิร์ฟเวอร์จะต้องทำตามข้อกำหนด ที่ได้มีการกำหนดไว้ใน TCP/IP Ports ซึ่งจะเป็นมาตรฐาน ตัวอย่าง : หมายเลข TCP/IP Port สำหรับ FTP Server คือ 21, และ สำหรับ Web Server คือ 80 มีข้อกำหนดที่ครอบคลุม (Global Arrangement) สำหรับการเรียกใช้บริการ (Services) โดยจะมีการกำหนดหมายเลขของพอร์ตที่ไคลเอนต์ควรที่จะส่งคำร้องขอบริการ ดังตารางด้านล่าง

ไคลเอนต์จะเริ่มต้นสร้างการติดต่อผ่านเน็ตเวิร์ค(network sessions) กับเซิร์ฟเวอร์ โดยผ่านทาง network protocols ตัวใดตัวหนึ่ง แล้วจะสร้างซ็อกเก็ต และ กำหนดให้มันติดต่อไปยังเซิร์ฟเวอร์ที่ต้องการ เมื่อซ็อกเก็ตได้รับที่อยู่ (address) และ หมายเลขพอร์ต (Port) ของเซิร์ฟเวอร์แล้วมันก็จะติดต่อไปยังเซิร์ฟเวอร์นั้นทันที

ระบบเครือข่ายใช้โปรโตคอลมาตรฐานชื่อ TCP/IP ในการสื่อสารผ่านระบบเพื่อติดต่อกับเครื่องคอมฯอื่นๆ โปรโตคอล TCP/IP นั้นประกอบด้วยส่วนที่สำคัญ 2 ส่วนก็คือ

- TCP (Transmission Control Protocol)
- IP (Internet Protocol)

2.4 ความรู้พื้นฐานของ TCP/IP

โปรโตคอล(Protocol) คือ ข้อกำหนดในการสื่อสารระหว่างคอมพิวเตอร์ด้วยกัน ซึ่งมีอยู่ด้วยกันมากมายหลายชนิด แต่ละชนิดก็มีข้อดีข้อเสียและใช้ในโอกาสหรือสถานการณ์ที่แตกต่างกันออกไป ซึ่งการที่จะสื่อสารกันรู้เรื่องและเข้าใจระหว่างกันได้ จะต้องใช้ภาษาเดียวกันในการสื่อสาร ซึ่งภาษาในการสื่อสารกันก็คือ โปรโตคอลนั่นเอง ถ้ามีการใช้โปรโตคอลที่แตกต่างกัน (สื่อสารต่างภาษากัน) การจะเข้าใจกันต้องใช้ตัวกลางในการแปลงโปรโตคอลกลับไปกลับมา ซึ่งเรียกว่า Gate way ซึ่งมีทั้งที่เป็น

เซิร์ฟเวอร์แยกต่างหากสำหรับทำหน้าที่นี้โดยเฉพาะ หรืออาจเป็นโปรแกรมหรือไดรฟ์เวอร์ที่สามารถติดตั้งในเครื่องคอมพิวเตอร์ทั้งสองฝ่ายนั้นได้เลย

โปรโตคอล TCP/IP เป็นโปรโตคอลระบบหนึ่ง ทำหน้าที่เป็นสื่อกลางของการรับส่งข้อมูลในระบบอินเตอร์เน็ต ถ้าจะเปรียบเทียบเครือข่ายอินเตอร์เน็ตเหมือนกับเครือข่ายของรถไฟที่ต้องอาศัยการวางรางเหล็กไปตามเส้นทางต่างๆ ทั่วประเทศ เปรียบเสมือนเครือข่ายคอมพิวเตอร์ที่สร้างขึ้นมา สิ่งที่ใช้รับส่งสินค้าหรือข้อมูลให้ไปถึงปลายทาง ก็คือ TCP/IP ในกรณีที่มีข้อมูลเข้ามามาก จำเป็นต้องแบ่งข้อมูลออกเป็นส่วนๆ (packet) แล้วค่อยๆ แยกกันส่งไปจนถึงปลายทาง จากนั้นจึงรวมแพ็กเกจย่อยๆ เข้ามารวมกันเป็นข้อมูลที่เหมือนต้นฉบับอีกครั้ง แต่ข้อมูลที่ถูกแบ่งออกมาอาจมีแพ็กเกจอื่นมาในสายเดียวกันด้วย ดังนั้นเมื่อถึงปลายทางแล้ว แต่ละแพ็กเกจจะแยกแยะตามเป้าหมายและปลายทางได้อย่างไม่มีปัญหา การทำงานของ TCP/IP จะต้องอาศัยอุปกรณ์ Router เพื่อช่วยทำหน้าที่ในการส่งข้อมูลข้ามไปมาระหว่างระบบเครือข่ายย่อยๆ จำนวนมากที่เชื่อมต่อกันได้ หน้าที่ของ Router จะเด่นชัดมากเมื่อได้มีส่วนในเครือข่ายอินเตอร์เน็ต โดยจะทำหน้าที่เลือกเส้นทางการเดินไปหาปลายทางของ TCP/IP เนื่องจาก TCP/IP ไม่มีข้อมูลของเส้นทางที่จะไปให้ถึงปลายทางได้ ต้องให้ Router ชี้ทางไปให้ แต่กว่าจะถึงปลายทาง อาจต้องอาศัย Router หลายตัว บางครั้งข้อมูลอาจหลงทางจนกระทั่งสูญหายไประหว่างทางก็เป็นได้ ดังนั้น TCP/IP จึงมีกลวิธีในการจัดการกับปัญหาเหล่านี้ ซึ่งอาจมีการส่งซ้ำใหม่อีกครั้ง หรือ มีการยืนยันจากปลายทางว่าได้รับข้อมูลครบถ้วนแล้ว

IP Address และการอ้างอิงอุปกรณ์ในเครือข่ายเป็นสิ่งจำเป็น เพราะอุปกรณ์ในเครือข่ายทุกชิ้นจะต้องมีหมายเลขประจำตัวและควรต้องไม่ซ้ำกัน เพื่อว่าใช้ในการอ้างอิงถึงกัน สำหรับสิ่งที่ใช้อ้างอิงตัวตนในเครือข่ายของอ TCP/IP เรียกว่า IP Address ซึ่งจะต้องเป็นไปตามรูปแบบมาตรฐานที่กำหนดเท่านั้น ไม่สามารถกำหนดได้เองตามใจชอบความสำคัญของ IP Address เหมือนเลขที่บ้านหรือเลขประจำตัว ที่ใช้ในการติดต่อกัน เช่นหากผู้ใช้ต้องการโหลดไฟล์จากคอมพิวเตอร์เครื่องใดเครื่องหนึ่ง คุณต้องใช้โปรแกรมสำหรับการโอนถ่ายข้อมูลอย่างโปรแกรม FTP โดยระบุ IP Address ของเครื่องปลายทางที่จะไปดาวน์โหลดข้อมูล ในทางปฏิบัติ จะมีวิธีการใช้ตัวอักษรย่อแทนหมายเลข IP Address เรียกว่า Domain Name

รูปแบบของ IP Address จะเป็นเลขสี่ชุดที่คั่นด้วยจุด (.) เช่น 192.168.0.1 แต่การเก็บค่าในคอมพิวเตอร์จะเป็นเลขฐานสองและไม่มีจุดดังตัวอย่าง

ตาราง 2.1 ตารางแสดงตัวอย่าง IP Address

11000000	10101000	00000000	0000001
192	168	0	1

เพื่อป้องกันการซ้ำซ้อนของ IP Address ต้องมีหน่วยงานกลางที่มีหน้าที่จัดสรร IP Address ให้กับผู้ใช้ทั่วโลก คือ ICANN (Internet Corporation for Assigned Names and Number) แต่ส่วนใหญ่จะได้ IP ที่จัดสรรมาโดย ISP มาอีกต่อหนึ่ง ส่วนเครือข่ายของ TCP/IP ที่ใช้ตามบ้านหรือที่ทำงานไม่จำเป็นต้องขอ IP Address เนื่องจากเป็นเครือข่ายที่ไม่ได้เกี่ยวข้องกับอินเตอร์เน็ตโดยตรง แต่เพื่อป้องกัน

ความสับสน ขอแนะนำช่วงของ IP Address ที่สงวนไว้ใช้กับเครือข่ายภายใน ที่เรียกว่า Private IP Address ซึ่งมีดังนี้

ตารางที่ 2.2 ตารางแสดงลำดับชั้นของ Private IP Address

Class	Private IP Address
A	10.0.0.0
B	172.16.0.0 – 173.31.0.0
C	192.168.0.0 – 192.168.255.0

การระบุประเภทของ IP Address นั้นมีอยู่ 2 ประเภท

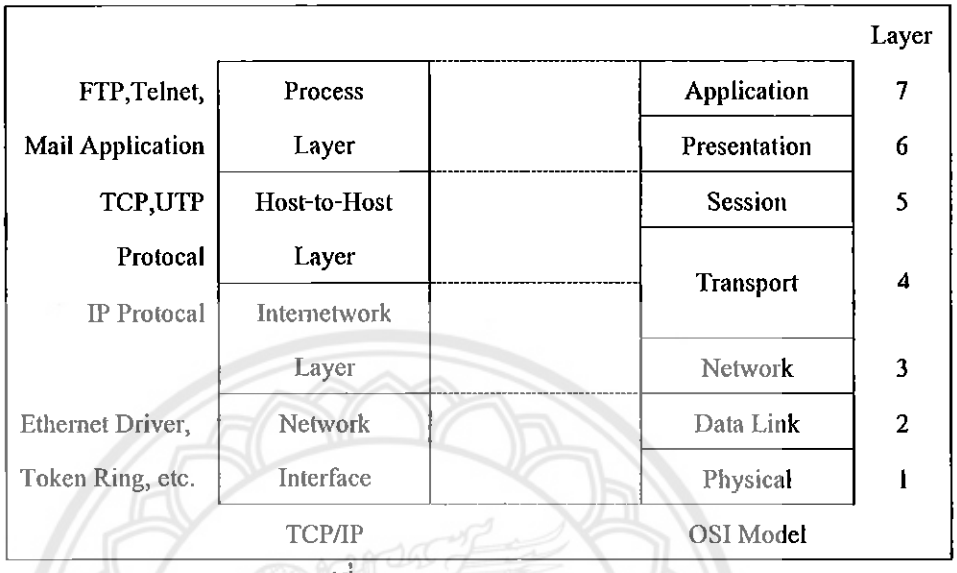
- Network Mask คือ การระบุว่า IP Address ที่เราใช้นั้นแบ่งย่อยเป็นอย่างไร มีกี่บิต ที่เป็นส่วนหนึ่งของหมายเลขเครื่องในเครือข่าย และกี่บิตที่เป็นหมายเลขเครื่อง หากจะใช้บิตใดเป็นเลขเครือข่ายก็ตั้งค่าบิตนั้นของ Network Mask ให้เป็น "1" เพื่อเป็นเสมือน "หน้ากาก" สำหรับบิตที่เหลือจะระบุให้เป็น "0" ซึ่งก็คือจำนวนบิตที่ใช้ในส่วนของหมายเลขเครื่อง โดยทั่วไปซอฟต์แวร์ของระบบปฏิบัติการจะกำหนดค่า Mask ให้เป็นอัตโนมัติ
- Subnet Mask จะใช้ในกรณีที่เราต้องการยืมบางส่วนของหมายเลขเครื่องมาใช้เป็นหมายเลขเครือข่าย เช่นขอยืม 1 บิตแรกของหมายเลขเครื่องมาเป็นหมายเลขของเครือข่าย เพื่อให้เกิดเป็นเครือข่ายย่อย หรือ Subnet ขึ้น เช่น เครือข่ายที่ใช้อยู่ คือ 128.1.x.x โดยที่ X มีค่าระหว่าง 1 – 254 ดังนั้น Network Mask คือ 255.255.0.0 อย่างไรก็ตามเราต้องการแบ่งเครือข่ายออกเป็นเครือข่ายย่อยสองเครือข่ายดังนั้นเราอาจจะยืม 1 บิตแรกของหมายเลขเครื่องช่วยในการกำหนดเลขเครือข่าย ซึ่งผลที่ได้คือ เครือข่าย 128.1.x.x จะถูกแบ่งเป็นสองเครือข่ายย่อย ได้แก่เครือข่ายย่อยที่ 1 มี IP Address อยู่ในช่วง 128.1.128.x – 128.1.254.x และเครือข่ายย่อยที่ 2 มี IP Address อยู่ในช่วง 128.1.0.x – 128.1.127.x โดยที่ X อยู่ในช่วง 1 - 254 และ Subnet Mask ที่ต้องกำหนดคือ 255.255.128.0

2.5 โครงสร้างของโปรโตคอล

โปรโตคอล TCP/IP มีการจัดแบ่งกลไกการทำงานออกเป็นชั้นๆ หรือ layer เหมือนกับมาตรฐาน OSI Model ในแต่ละ layer ของโปรโตคอล TCP/IP จะประกอบด้วย

- Process layer หรือ Application Layer
- Host-to-Host layer หรือ Transport Layer
- Internetwork layer
- Network Interface layer

โดยเมื่อเปรียบเทียบกับมาตรฐาน OSI model แล้วซึ่งเราจะเห็นว่าบาง Layer ของโปรโตคอล TCP/IP เทียบได้กับมาตรฐาน OSI model ถึงสอง Layer และบาง Layer ก็จะทำงานคาบเกี่ยวกับหลายๆ Layer ของ OSI model



รูปที่ 2.1 TCP/IP และ OSI Model

ลำดับชั้นการทำงานของโปรโตคอล TCP/IP เทียบกับมาตรฐาน OSI model นั้น ในชั้นบนสุดเรียกว่า Process layer ทำงาน 2 หน้าที่เทียบได้กับ Application layer และ Presentation layer ในชั้นนี้จะรองรับการทำงานของแอพพลิเคชั่นต่างๆ ที่ทำงานเป็นโปรเซส อยู่ในเครื่องเซิร์ฟเวอร์ ที่ให้บริการและเครื่องที่ขอใช้บริการ หรือ ไคลเอนต์ (Client) ซึ่งจะติดต่อกันผ่านโปรโตคอลเฉพาะแอพพลิเคชั่นอีกทีหนึ่ง ตัวอย่างเช่น เมื่อผู้ใช้งานอินเตอร์เน็ตต้องการถ่ายโอนไฟล์หรือ Download ข้อมูลจากเครื่องเซิร์ฟเวอร์ที่ให้บริการ โดยอาจจะเรียกใช้โปรแกรม FTP Client ทั่วไป เช่น โปรแกรม WS_ftp ติดต่อกับโปรเซส FTP ที่กำลังให้บริการอยู่ที่เครื่องเซิร์ฟเวอร์ จากนั้นตัวโปรเซส FTP ก็จะเรียกใช้โปรโตคอล FTP (File Transfer Protocol) เพื่อทำการถ่ายโอนไฟล์นี้ หรือถ้าผู้ใช้ต้องการเรียกใช้งานคอมพิวเตอร์เครื่องที่อยู่ห่างไกลออกไปด้วยการใช้โปรแกรม Telnet ที่เครื่องเซิร์ฟเวอร์ให้บริการตัวโปรเซส Telnet ที่ทำงานอยู่ก็จะเรียกใช้โปรโตคอล Telnet เพื่อติดต่อกัน หรือในกรณีที่มีการเรียกใช้โปรแกรม Web Browser เช่น Netscape Navigator เพื่อเรียกดูเว็บเพจในเว็บไซท์ CNN ที่เครื่องซึ่งให้บริการเว็บของ CNN ก็จะมีโปรเซส HTTP (HyperText Transfer Protocol) ทำงานอยู่และจะติดต่อกับผู้ใช้ผ่านโปรโตคอล HTTP เป็นต้น การทำงานของแอพพลิเคชั่นต่างๆ จะอยู่ที่ Process layer นี้ และมีการติดต่อกันตามแต่ละโปรโตคอลเฉพาะแล้วแต่แอพพลิเคชั่นที่ใช้งาน จากการที่ Process layer ของ TCP/IP รองรับให้โปรโตคอลอื่นทำงานได้หลายโปรเซสและหลายโปรโตคอลได้พร้อมกันนั้น ทำให้ผู้ใช้สามารถใช้งานได้หลายๆอย่างพร้อมกัน เช่น เปิดโปรแกรม Internet Explorer เพื่อเรียกดูเว็บเพจพร้อมกับใช้งานโปรแกรม Outlook Express เพื่อรับส่งอีเมล์ไปพร้อมกันได้โดยไม่ต้องรอให้ทำงานอย่างหนึ่งอย่างใดเสร็จก่อน หรือในปัจจุบันมีการพัฒนาโปรแกรม Web Browser ให้สามารถเรียกใช้งาน โปรโตคอลได้มากขึ้น ทำให้เรา

สามารถใช้โปรแกรม Web Browser โอนถ่ายไฟล์ข้อมูลที่ใช้ โปรโตคอล FTP ได้โดยไม่ต้องไปหาโปรแกรมอื่นมาใช้

โปรโตคอลหลักๆที่ทำงานใน Process layer ซึ่งผู้ใช้มักจะคุ้นเคยกันดีได้แก่ FTP (File Transfer Protocol) , Telnet, HTTP (HyperText Transfer Protocol) และ SMTP (Simple Mail Transfer protocol) นอกจากนี้ยังมีโปรโตคอลอื่นที่อยู่เบื้องหลัง ซึ่งทำงานโดยที่ผู้ใช้ไม่สามารถมองเห็นได้จากโปรแกรมหรือไม่ได้ทีการใช้งานโดยตรง เช่น

- โปรโตคอล DNS (Domain Name System) ที่ทำหน้าที่แปลงข้อมูลชื่อ domain name หรือชื่อเว็บไซท์ทั้งหลายให้เป็นหมายเลข IP Address
- โปรโตคอล SNMP (Simple network management Protocol) ใช้ในการควบคุมและตรวจสอบอุปกรณ์ที่อยู่ในเครือข่าย
- โปรโตคอล DHCP (Dynamic Host Configuration Protocol) ทำหน้าที่แจกจ่ายข้อมูลพารามิเตอร์ของเครือข่ายให้กับเครือลูกข่ายที่เชื่อมต่ออยู่

2.6 ความรู้พื้นฐานเกี่ยวกับ Microsoft Visual Basic Version 6.0

ภาษา Visual Basic เป็นเครื่องมือสำหรับเขียนโปรแกรมที่ได้รับความนิยมทั่วโลกรวมทั้งในเมืองไทย และเป็นภาษาที่เข้าใจได้ไม่ยากนัก ซึ่งความเป็นมาของภาษา Basic นั้นได้ถูกสร้างขึ้นในปี 1963 โดย Horn Keneny และ Thomas Kutz ที่มหาวิทยาลัย Dartmouth สร้างขึ้นเพื่อใช้ในการสอนการเขียนโปรแกรมโดยเน้นภาษาง่ายต่อการเข้าใจและการใช้งาน โดยที่เราสามารถทำการสร้างแอพพลิเคชั่นที่สามารถใช้งานกับระบบ Windows ได้ง่ายขึ้น ทำให้มีการสร้างการเขียนโปรแกรมแบบ Visual Programming กำเนิดขึ้นมา รูปแบบนี้ก็คือ การเขียนโปรแกรมพร้อมกับการเห็นผลลัพธ์ที่เกิดขึ้น Visual Basic เป็นหนึ่งใน Visual Programming ที่ไมโครซอฟท์สร้างขึ้นมา และด้วยความเรียบง่ายของภาษาและการเขียนโปรแกรมที่รวดเร็ว ทำให้ได้รับความนิยมสูงในปัจจุบัน

การเขียนโปรแกรมเพื่อสร้างแอพพลิเคชั่นจาก Visual Basic นั้นจะมีวิธีการสร้างที่แตกต่างจากโปรแกรมอื่นเพราะ Visual Basic จะทำงานแบบ Event-Driven ซึ่งก็คือการเขียนโปรแกรมแบบ " ถ้าเหตุการณ์เกิดขึ้น เราจะดำเนินการอย่างไร " กล่าวคือ ถ้าเหตุการณ์นี้เกิดขึ้น เราจะตอบสนองกับเหตุการณ์นี้อย่างไร เป็นโปรแกรมที่รองรับเหตุการณ์ Visual Basic เป็นโปรแกรมที่สนับสนุนการเขียนโปรแกรมแบบ Event-Driven โดยมีเครื่องมือต่าง ๆ ช่วยในการจัดการกับเหตุการณ์ต่าง ๆ ที่จะเกิดขึ้นการเขียนโปรแกรมบนระบบปฏิบัติการ Windows ให้สามารถติดต่อสื่อสารบนระบบเครือข่าย ปัจจุบันนี้เครื่องคอมพิวเตอร์ส่วนใหญ่หันมาใช้ระบบปฏิบัติการ Windows กันเนื่องจากมีเครื่องมือ (Tool) ที่ช่วยให้เราสามารถเขียนโปรแกรมติดต่อผ่านระบบเครือข่าย หนึ่งในเครื่องมือที่สามารถช่วยในการติดต่อสื่อสารผ่านทางระบบเครือข่าย คือMicrosoft Winsock Control 6 ซึ่งเป็นเครื่องมือที่อำนวยความสะดวกสำหรับการเขียนโปรแกรมติดต่อผ่านระบบเครือข่าย โดยโปรโตคอล TCP/IP

2.6.1 Microsoft Winsock Control 6

ส่วนที่ใช้ในการคอนโทรลที่ชื่อว่า Microsoft Winsock Control 6 ที่มากับ Visual Basic เป็น

เครื่องมือที่มีประโยชน์ในการพัฒนา โปรแกรมบน Internet เมื่อเปรียบเทียบกับคอนโทรล ActiveX ตัวอื่น ๆ Winsock จะเป็นคอนโทรลที่ถูกนำมาใช้งานมากคอนโทรลหนึ่ง แต่อย่างไรก็ตามถ้าต้องการที่จะคอนโทรลขึ้นมาเอง และไม่ต้องการที่จะใช้คอนโทรลของผู้พัฒนาคนอื่น ซึ่งอาจมีข้อผิดพลาดที่คุณไม่ต้องการ คุณก็จำเป็นจะต้องมีพื้นฐานที่เกี่ยวกับ Network Protocols และ หลักในการติดต่อสื่อสาร คุณสมบัติของ Winsock เหมาะสำหรับการใช้งานต่างๆ ดังนี้

- สร้างแอพพลิเคชันของไคลเอนต์ (Client) ที่รวบรวมสารสนเทศของผู้ใช้ก่อนทจะส่งคำสั่งไปยังเซิร์ฟเวอร์ (Server) ส่วนกลาง
- สร้างแอพพลิเคชันของเซิร์ฟเวอร์ (Server) เป็นศูนย์กลางในการรวบรวมข้อมูลจากผู้ใช้ต่างๆ จำนวนมาก
- สร้างแอพพลิเคชันสำหรับการพูดคุย (Chat)

การใช้ Control ของ Winsock

การควบคุม Winsock ที่ดีช่วยให้ TCP/IP ได้ง่ายและสะดวกขึ้นมาก ไมโครซอฟท์ได้ห่อหุ้ม API ของ Winsock และ inet เป็นแพ็กเกจ ที่สามารถรวมให้เป็นไฟล์เดียวเหมือนต้นฉบับภายในแอพพลิเคชันต่างๆ ของ Visual Basic ได้ง่าย การควบคุม Winsock จะช่วยในการสร้าง แอพพลิเคชันของไคลเอนต์หรือเซิร์ฟเวอร์ที่สามารถเชื่อมโยงกับตัวคอนโทรล Winsock และทำการแลกเปลี่ยนข้อมูลทางช่องทางนั้น ส่วนทางเซิร์ฟเวอร์จะยอมรับการติดต่อกันได้หลายไคลเอนต์ พร้อมทั้งรายงานสถานะของแต่ละเครื่องที่เชื่อมต่อกัน

คุณสมบัติต่างๆ ของการคอนโทรล Winsock

Winsock ช่วยในการสร้างไคลเอนต์และเซิร์ฟเวอร์โดยใช้คอนโทรลอย่างเดียวกัน การคอนโทรล Winsock ใช้คุณสมบัติต่างๆ ที่เหมือนกันไม่ว่าจะสร้างไคลเอนต์หรือเซิร์ฟเวอร์ คุณสมบัติของ Winsock จะช่วยสนับสนุนสารสนเทศ ในการกำหนดเงื่อนไขการเชื่อมโยงของ Socket , คุณสมบัติของโปรโตคอล จะช่วยระบุโปรโตคอลที่ใช้ในการสื่อสารเซิร์ฟเวอร์ของอินเตอร์เน็ตส่วนใหญ่ใช้โปรโตคอลของ TCP/IP

ตารางที่ 2.3 ตารางแสดงคุณสมบัติต่างๆ ของคอนโทรล Winsock

Property	คำอธิบาย	ตัวอย่าง
BytesReceived	ส่งกลับจำนวนไบต์ต่างๆ ที่ได้รับ	Var = Winsock1.BytesReceived
LocalHostName	ส่งกลับชื่อของคอมพิวเตอร์ท้องถิ่น	Var = Winsock1.LocalHostName
LocalIP	ส่งกลับแอดเดรสของ IP ของคอมพิวเตอร์ท้องถิ่น	Var = Winsock1.LocalIP
LocalPort	ส่งกลับเลขหมายพอร์ตที่ใช้บนคอมพิวเตอร์ท้องถิ่น	Winsock1.LocalPort = 1001
Protocol	ส่งกลับหรือกำหนดโปรโตคอล	Winsock1.Protocol = sckTCPProtocol
RemoteHostIP	ส่งกลับแอดเดรสของ IP ของ	Var = Winsock1.RemoteHostIP

	คอมพิวเตอร์ที่ห่างไกล	
RemoteHost	ส่งกลับหรือกำหนดชื่อหรือแอดเดรสของ IP ของคอมพิวเตอร์ที่อยู่ห่างไกล	Winsock1.RemoteHost = _ "100.0.0.1"หรือftp.microsotft.com
RemotePort	ส่งกลับหรือกำหนดพอร์ตที่ใช้บนคอมพิวเตอร์ที่ห่างไกล	Winsock1.RemotePort = 1001
SocketHandle	ส่งกลับสิ่งที่อ้างถึงการเชื่อมโยง API ของ Winsock	Var = Winsock1.SocketHandle
State	ส่งสถานะกลับคอนโทรล	Var = Winsock1.State

Property ของ State

State จะส่งกลับสถานะของตัวคอนโทรลที่ใช้การระบุค่าคงที่ลงไป Property ของ State จะทำการอ่านอย่างเดียวไม่สามารถทำการกำหนดค่าได้ในช่วงที่ออกแบบ Property ของ State จะเป็นตัวกำหนดการใช้เมธอด (method) หรือเหตุการณ์ต่างๆ (Event)

ตารางที่ 2.4 ตารางแสดงค่าต่างๆ สำหรับกำหนด Property ของ State

ชุดคำสั่ง	ค่าคงที่	คำอธิบาย
sckClose	0	ปิด socket
sckOpen	1	เปิด socket
sckListen	2	socket ฟัง
sckConnectionPending	3	รอคอยการเชื่อมโยง
sckResolvingHost	4	ชื่อ Host ที่ห่างไกลจะถูกเปลี่ยนแปลงเป็น IP
sckResolve	5	ชื่อ Host ที่อยู่ห่างไกลถูกแยกออกเป็น IP
sckConnecting	6	socket เชื่อมต่อคอมพิวเตอร์ที่ห่างไกล
sckConnected	7	socket เชื่อมต่อคอมพิวเตอร์ที่ห่างไกลแล้ว
sckClose	8	socket ปิดการเชื่อมต่อคอมพิวเตอร์ที่ห่างไกล
skcError	9	socket เกิดความผิดพลาด

และในตารางต่อไปนี้ จะแสดงรายการของการเชื่อมต่อต่างๆ ที่สามารถใช้กับการควบคุม Winsock

ตารางที่ 2.5 ตารางแสดงรายการของการเชื่อมต่อต่างๆ ที่สามารถใช้กับการควบคุม Winsock

Method	คำอธิบาย
Accept	ยอมรับการเชื่อมโยงที่ร้องของในเหตุการณ์ ConnectionRequest
Blind	เลือกอุปกรณ์ของเครือข่ายท้องถิ่นและพอร์ตท้องถิ่น

Close	ปิด socket รับฟัง สำหรับการเชื่อมโยงของ TCP และกำหนดคอนโทรลใหม่ เพื่อสามารถเปลี่ยนโปรโตคอล
Connect	สร้างการเชื่อมโยง TCP ให้ Host ที่ห่างไกล
Listen	รอคอยคอมพิวเตอร์อื่นทำการเชื่อมโยงกับ Server (เฉพาะโปรโตคอลของ TCP เท่านั้น)
PeekData	นำข้อมูลออกจากบัฟเฟอร์มาแสดงผล แต่ไม่เคลียร์หน่วยความจำของบัฟเฟอร์นั้น
SendData	ส่งข้อมูลไปให้คอมพิวเตอร์ที่ห่างไกล
DataArrival	เกิดขึ้นเมื่อได้รับข้อมูลจากคอมพิวเตอร์ที่ห่างไกล
Error	เกิดขึ้นเมื่อมีการผิดพลาด
SendComplete	เกิดขึ้นเมื่อการส่งข้อมูลเสร็จสิ้น
SendProgress	เกิดขึ้นระหว่างการส่งผ่านข้อมูล

2.6.2 ความรู้เกี่ยวกับ API

API(Application Programming Interface) ของ Windows เป็นกลุ่มของฟังก์ชันต่างๆ ที่ทาง Windows เปิดเผยให้แก่โปรแกรมเมอร์ ฟังก์ชันของระบบปฏิบัติการ Windows สามารถเรียกคำสั่งได้จาก Visual Basic เพื่อให้ดำเนินงานต่างๆ ที่มีสามารถเขียนโปรแกรมด้วยโค้ดของ Visual Basic มาตรฐาน API ของ Windows การใช้ฟังก์ชันต่างๆ ของ API บนระบบปฏิบัติการบนวินโดว์เป็นเรื่องค่อนข้างยุ่งยากมากกว่าการใช้ฟังก์ชันมาตรฐานต่างๆ ที่ผู้ใช้สร้างขึ้นเพื่อการใช้งาน โดยโปรแกรมเมอร์ของ Visual Basic อาจจะมองฟังก์ชัน API ของ Windows มีส่วนที่คล้ายกับฟังก์ชันต่างๆ ของ Visual Basic ที่บรรจุค่าพารามิเตอร์ต่างๆ เกี่ยวกับอินพุตและเอาท์พุตและบางครั้งมีการส่งค่ากลับ อย่างไรก็ตามฟังก์ชันต่างๆ ของ API ได้คอมไพล์เรียบร้อยแล้ว อยู่ในไฟล์แยกกัน ที่ทราบกันทั่วไปในนามสกุลของ *.DLL (Dynamic Link Library) และการใช้ฟังก์ชันต่างๆ เหล่านี้ โปรแกรมเมอร์ต้องเพิ่มค่าอีกเล็กน้อย เพื่อกำหนดฟังก์ชันภายนอกสำหรับ Visual Basic นอกจากนี้ การใช้ฟังก์ชันของ API เริ่มแรกจะต้องประกาศฟังก์ชันของ API ที่ท่านต้องการใช้ การประกาศของ API จะใส่ลงไปที่ส่วนประกาศทั่วไป (General Declarations) ของ module เช่นเดียวกับการประกาศตัวแปรต่างๆ จะต้องประกาศฟังก์ชันต่างๆ ของ API จนกระทั่งโค้ดจะเข้าถึงฟังก์ชันดังกล่าว การประกาศของ API ก็จะช่วยเชื่อมโยงไปยัง DLL ที่อยู่ภายใน ซึ่งปกติการประกาศ API จะอยู่ใน module แต่ก็สามารถให้ฟอร์มและคลาสต่างๆ โดยการเพิ่มคีย์เวิร์ด Private ตรงหน้าการประกาศนั้นๆ คำสั่ง Declare จะมีรายการพารามิเตอร์ต่างๆ ของฟังก์ชันของ API ตัวของ DLL จะถูกกำหนดและประเภทของข้อมูลของค่าที่จะส่งกลับ ไม่เหมือนกับฟังก์ชันของ Visual Basic ธรรมดา การประกาศของ API จะไม่มีโค้ดของฟังก์ชัน การประกาศของ API จะใช้คำสั่งบรรทัดเดียวเพื่อชี้ไปยังไฟล์ของ DLL ที่บรรจุฟังก์ชันความสัมพันธ์นั้นๆ โดยคำสั่ง Declare ที่มีอยู่มากมายได้รวบรวมพารามิเตอร์ Alias ซึ่ง alias จะช่วยระบุชื่อจริงของฟังก์ชัน API ที่บรรจุใน DLL ซึ่งอาจไม่สอดคล้องกับ

ชื่อที่กำหนดตั้งแต่แรกให้ฟังก์ชันในโปรแกรม ตัวอย่าง ฟังก์ชันชื่อ "_lopen" ปรากฏใน DLL ของ kernal32 แต่ "_lopen" จะเป็นชื่อฟังก์ชันที่ไม่ถูกต้องภายใน Visual Basic ในกรณีนี้ การประกาศของ API จะต้องมีรูปแบบดังนี้

```
Declare Function lopen Lib "kernal32" Alias
        "_lopen"(ByVal lpPathName As String,_
        ByVal iReadwrite As Long) As Long
```

Visual Basic จะเห็นฟังก์ชันที่ชื่อ "_lopen" แต่จะทราบจากพารามิเตอร์ Alias ของคำสั่ง Declare ที่ผ่านการเรียกใช้ไปยังฟังก์ชันชื่อ "_lopen" ที่อยู่ภายใน DLL ของ kernel32 ไฟล์ต่างๆ ของไลบรารีของ API ของวินโดว์ ซึ่งจะเป็นไฟล์ DLL ที่อยู่ในไดเรกทอรี \Windows\System โดยที่ไฟล์ DLL จะพบได้ทุกๆ เครื่อง PC ที่มีระบบปฏิบัติการวินโดว์ ไฟล์ที่จำเป็นจะติดตั้งไว้ ในขณะที่ติดตั้งระบบปฏิบัติการวินโดว์ต่างๆ ไฟล์ DLL ที่สำคัญของวินโดว์ได้แก่

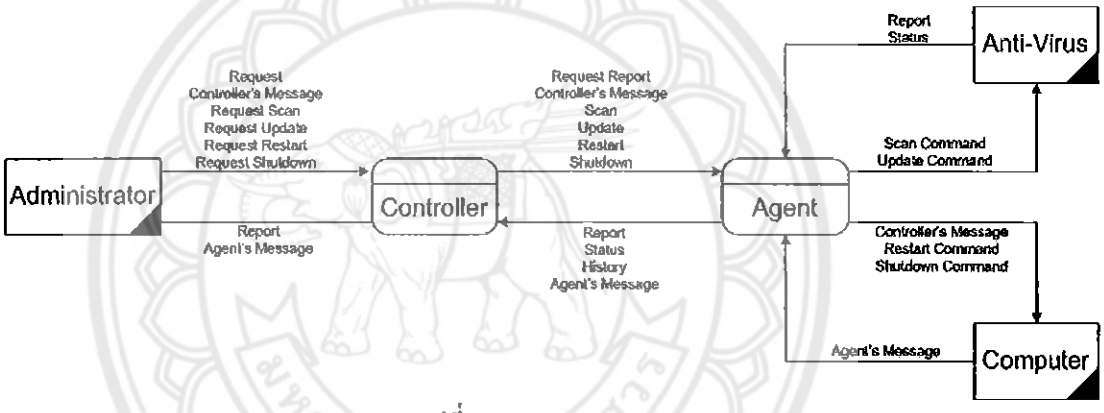
- USER32.DLL เป็นไลบรารีที่บรรจุฟังก์ชันต่างๆ ที่เกี่ยวข้องกับสภาพแวดล้อมของระบบวินโดว์ เช่น การเชื่อมโยงข่าวสารต่างๆ ระหว่างวินโดว์, การจัดการเคอร์เซอร์ต่างๆ, การจัดการบนเมนู และ การเชื่อมโยงกับฟังก์ชันอื่นๆ แต่ไม่ได้แสดงผลออกหน้าจอ
- KERNEL32.DLL เป็นไลบรารีที่บรรจุค่าต่าง ๆ ไว้มากมาย ที่ช่วยจัดการฟังก์ชันต่าง ๆ ของระบบปฏิบัติการในระดับต่ำ โดยหน้าที่ทั่วไป เช่น การจัดการเรื่องหน่วยความจำ, การเชื่อมโยงของแหล่งทรัพยากร, การจัดการเกี่ยวกับไฟล์และไดเรกทอรี และการจัดการเกี่ยวกับโมดูล
- GDI32.DLL (Graphics Device Interface) จะมีฟังก์ชันที่ช่วยจัดการเอาท์พุตให้อุปกรณ์อื่น ๆ โดยเฉพาะส่วนของจอภาพ

บทที่ 3
การศึกษาและพัฒนาโปรแกรม

3.1 เครื่องมือที่ใช้พัฒนาโปรแกรม

1. ภาษาที่ใช้พัฒนาโปรแกรม คือ Microsoft Visual Basic Version 6.0
2. Components ที่ใช้ในการสื่อสาร คือ Microsoft Winsock Control
3. โปรโตคอลที่ใช้ในการสื่อสาร คือ TCP/IP
4. โปรแกรม Anti-Virus ที่เลือกใช้ คือ โปรแกรม AntiVir

3.2 การออกแบบระบบ(Context Diagram)



รูปที่ 3.1 Context Diagram

3.3 แผนภาพกระแสข้อมูล (Data Flow Diagram)

เป็นแผนภาพที่ใช้ในการวิเคราะห์โครงสร้าง โดยแผนภาพจะแสดงความสัมพันธ์ระหว่างโปรเซสกับข้อมูลที่เกี่ยวข้อง เส้นทางการเดินของข้อมูลและเหตุการณ์ใดกับข้อมูลที่เข้าโปรเซสและผลตอบสนองที่ออกมาจากโปรเซสดังกล่าว

3.3.1 แผนภาพกระแสข้อมูลระดับที่ 1 (Data Flow Diagram Level-1)

จากข้อมูลของโปรแกรม Remote Admin สามารถรวบรวมและทำการวิเคราะห์เพื่อแสดงรายละเอียดของ Boundaries, Data และ Process ดังรายละเอียดดังต่อไปนี้

List of Boundaries

1. ผู้ดูแลระบบ (Admin) มีหน้าที่ในการป้อนคำสั่งที่ต้องการ เช่น การส่งคำร้องขอ การสั่งให้โปรแกรม Anti-Virus ทำงาน การสั่งให้เครื่องลูกข่ายทำการ Restart หรือ Shutdown ให้กับระบบและรับทราบรายงานที่ได้รับหลังการปฏิบัติงานระบบเสร็จสิ้น
2. โปรแกรม Anti-Virus ที่เลือกใช้ (AntiVir) จะถูกติดตั้งให้กับเครื่องลูกข่าย โดยจะรอรับคำสั่ง

จากเครื่อง Server เช่น คำสั่ง Scan หรือการ Update และทำการ Active เพื่อให้ได้ผลลัพธ์ตามที่ต้องการ หลังจากนั้นจึงส่งผลการรายงานค่าคืนกลับมายังระบบเพื่อทำการนำเสนอรายงานส่งให้ Admin ต่อไป

3. Computer จะถูกเตรียมพร้อมที่จะทำงานเพื่อรองรับการทำงานของ Anti-Virus และรอรับคำสั่ง Restart หรือ Shutdown เพื่อหยุดการทำงานของคอมพิวเตอร์หลังจากการปฏิบัติงานของระบบเสร็จสิ้นลง

List of Process

1. Controller Report มีหน้าที่ในการส่งคำร้องขอ (Request) และนำข้อมูลที่ได้รับมาทำการแยกประเภทของข้อมูลพร้อมทั้งการนำเสนอข้อมูลต่างๆ เพื่อเป็นรายงานนำส่งผู้ดูแลระบบต่อไป

2. Agent Remote มีหน้าที่ส่งคำสั่งที่ใช้ในการควบคุมคอมพิวเตอร์ เช่นการ Restart หรือการ Shutdown และส่งให้กับโปรเซสถัดไป เพื่อให้ดำเนินการต่อไป

3. Anti-Virus Remote มีหน้าที่ในการส่งคำสั่งที่ใช้ในการควบคุม Anti-Virus ให้ทำงานตามที่ผู้ดูแลระบบต้องการ เช่น การ Scan หรือ การ Update

4. Controller Chat มีหน้าที่ส่งข้อความจากเครื่องของผู้ดูแลระบบไปยังเครื่องลูกข่ายและรับข้อความที่ส่งมาจากเครื่องลูกข่ายและแสดงผลต่อผู้ดูแลระบบต่อไป

5. Agent Report มีหน้าที่ดึงข้อมูลจากส่วนเก็บรายงานเหตุการณ์ที่เกิดขึ้นจากการทำงานของ Anti-Virus และนำรายงานที่ได้ส่งไปสองส่วน ส่วนหนึ่งนำส่งไปยังโปรเซส Server Report อีกส่วนหนึ่งถูกนำข้อมูลไปสร้าง History ก่อนจะนำ History ที่ได้นำส่งไปยังโปรเซส Server Report เพื่อนำข้อมูลดังกล่าวนำเสนอเป็นรายงานที่จะให้ผู้ดูแลระบบทราบต่อไป

6. API มีหน้าที่ประสานการทำงานของระบบ โดยแบ่งการประสานงานออกเป็นสองส่วน ส่วนแรก ประสานงานเกี่ยวกับการควบคุมคอมพิวเตอร์ ซึ่งรับคำสั่งจาก Client Remote เช่น การ Restart หรือ การ Shutdown โดยสร้างคำสั่งเพื่อกระทำการดังกล่าว อีกส่วนทำการประสานงานเกี่ยวกับการดึงค่า Status ของการทำงาน Anti-Virus และนำค่า Status ที่ได้นำส่งให้โปรเซส Client Report เพื่อนำเสนอเป็นรายงานเพื่อให้ผู้ดูแลระบบทราบต่อไป

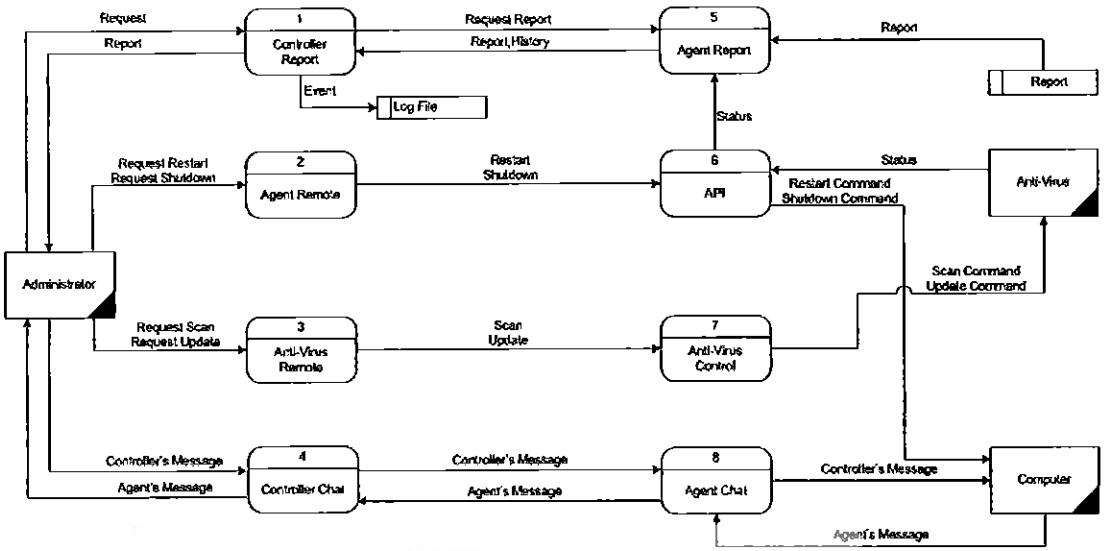
7. Anti-Virus Control มีหน้าที่ในการสั่งงานให้ Anti-Virus ที่เลือกใช้ (AntiVir) ทำงานตามที่ผู้ดูแลระบบสั่งการมา เช่น การ Scan หรือ การ Update

8. Agent Chat มีหน้าที่รับข้อความจากเครื่องของผู้ดูแลระบบ เพื่อทำการแสดงผลต่อเครื่องลูกข่าย และส่งข้อความจากเครื่องลูกข่ายนำไปแสดงผลต่อผู้ดูแลระบบต่อไป

List of Data

1. Log File มีหน้าที่เก็บค่าเหตุการณ์ที่เกิดขึ้นกับเครื่อง Agent

2. Report มีหน้าที่เก็บส่วนของรายงานผลการปฏิบัติงานของระบบ เช่น ผลของการตรวจสอบ โดยมีการเก็บค่าชนิด และจำนวนของไวรัสคอมพิวเตอร์ที่ตรวจพบ



รูปที่ 3.2 แสดงกระแสข้อมูลระดับ 1

3.3.2 แผนภาพแสดงข้อมูลระดับสอง (Data Flow Diagram Level-2)

เป็นการแสดงโปรเซสย่อย ที่สามารถแยกได้จากโปรเซสที่มีระดับ 1 **เพื่อการวิเคราะห์การทำงานภายในโปรเซสนั้นๆ**

โปรเซสที่ 1 Server Report มี 5 โปรเซสย่อย

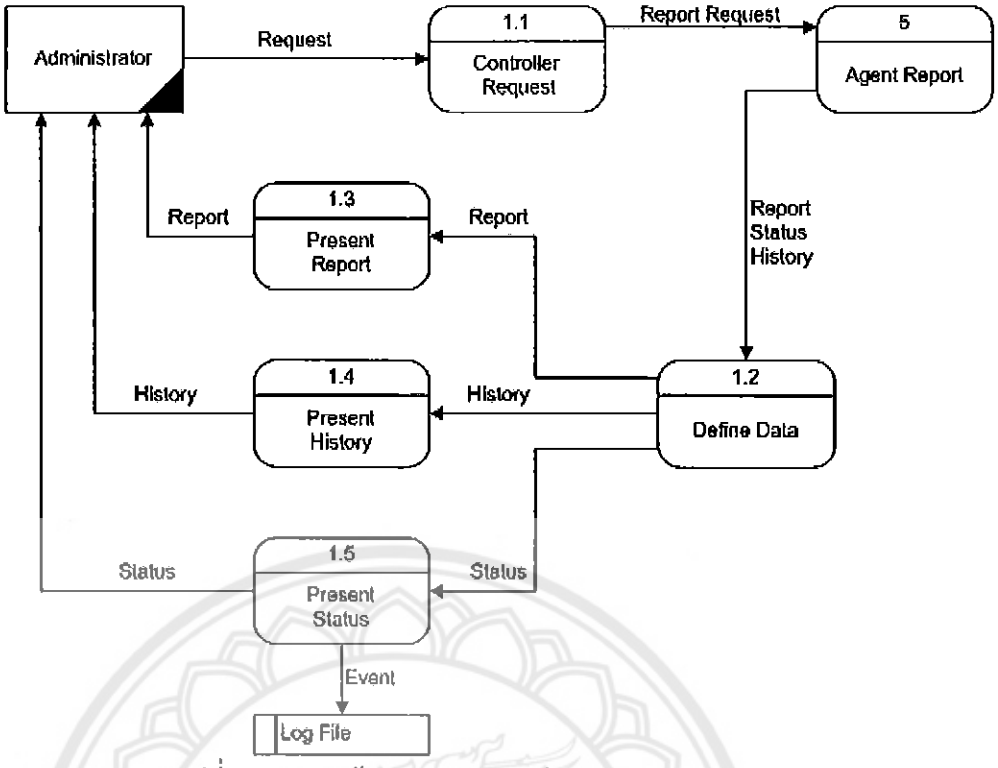
โปรเซสที่ 1.1 Server Request มีหน้าที่ส่งคำร้องขอเพื่อให้โปรเซส Agent Report **ส่งข้อมูลที่จำเป็นมา**ให้กับโปรเซส

โปรเซสที่ 1.2 Define Data มีหน้าที่แยกประเภทของข้อมูลที่ได้รับ

โปรเซสที่ 1.3 Present Report มีหน้าที่รวบรวมข้อมูล Report ที่ได้**รับมาเข้าสู่กระบวนการนำเสนอ** เพื่อรายงานให้ผู้ดูแลระบบทราบ

โปรเซสที่ 1.4 Present History มีหน้าที่รวบรวมข้อมูล History ที่ได้**รับมาเข้าสู่กระบวนการนำเสนอ** เพื่อรายงานให้ผู้ดูแลระบบทราบ

โปรเซสที่ 1.5 Present Status มีหน้าที่รายงาน Status ของเครื่องลูกข่าย **ให้แก่ผู้ดูแลระบบทราบ และเขียนเหตุการณ์ (Event) ลง Log File**

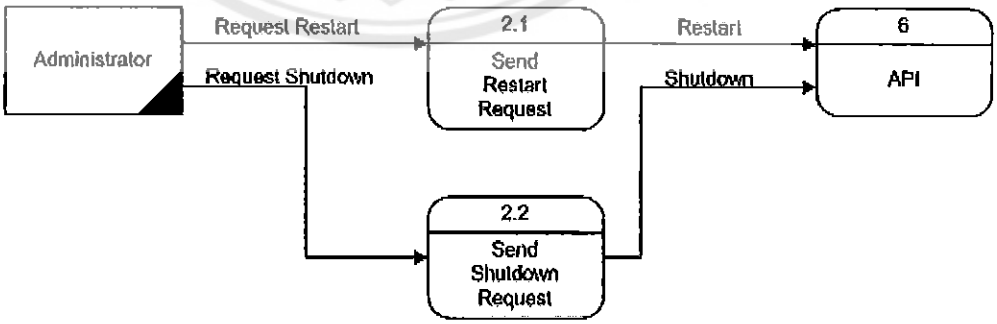


รูปที่ 3.3 กระแสข้อมูลระดับ 2 ของโปรเซส Controller Report

โปรเซสที่ 2 Agent Remote มี 2 โปรเซสย่อย

โปรเซสที่ 2.1 Send Restart Request มีหน้าที่สร้างคำขอสำหรับการ Restart เครื่อง Agent และส่งคำขอดังกล่าวเข้าสู่โปรเซส API ต่อไป

โปรเซสที่ 2.2 Send Shutdown Request มีหน้าที่สร้างคำขอสำหรับการ Shutdown เครื่อง Agent และส่งคำขอดังกล่าวเข้าสู่โปรเซส API ต่อไป

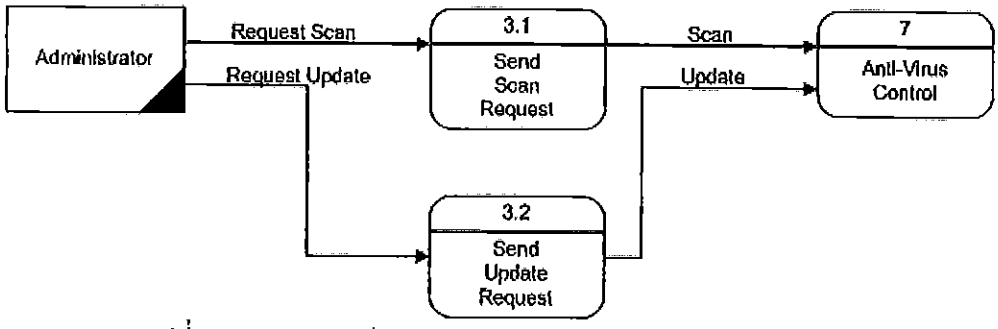


รูปที่ 3.4 กระแสของข้อมูลระดับ 2 ของโปรเซส Agent Remote

โปรเซสที่ 3 Anti-Virus Remote มี 2 โปรเซสย่อย

โปรเซสที่ 3.1 Send Scan Request มีหน้าที่สร้างคำขอสำหรับการตรวจสอบเครื่อง Agent โดยเป็นการสั่งให้ Anti-Virus ที่เลือกใช้ ทำการตรวจสอบเครื่องลูกข่ายดังกล่าว

โปรเซสที่ 3.2 Send Update Request มีหน้าที่สร้างคำขอสำหรับการปรับปรุงโปรแกรมต่อต้านไวรัส เครื่อง Agent โดยเป็นการสั่งให้ Anti-Virus ที่เลือกใช้ ทำการปรับปรุงโปรแกรมตัวเอง

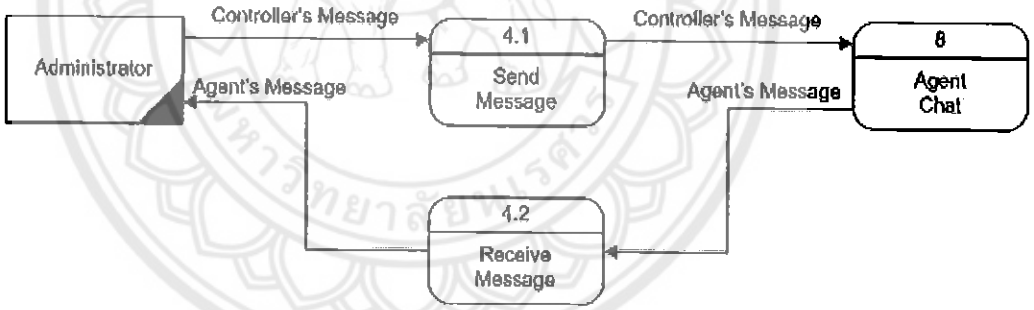


รูปที่ 3.5 กระแสของข้อมูลระดับ 2 ของโปรเซส Anti-Virus Remote

โปรเซสที่ 4 Controller Chat มี 2 โปรเซสย่อย

โปรเซสที่ 4.1 Send Message มีหน้าที่ส่งข้อความจากผู้ดูแลระบบและส่งข้อความดังกล่าวให้กับเครื่องลูกข่าย

โปรเซสที่ 4.2 Receive Message มีหน้าที่รับข้อความจากเครื่องลูกข่ายและส่งข้อความดังกล่าวให้กับผู้ดูแลระบบ

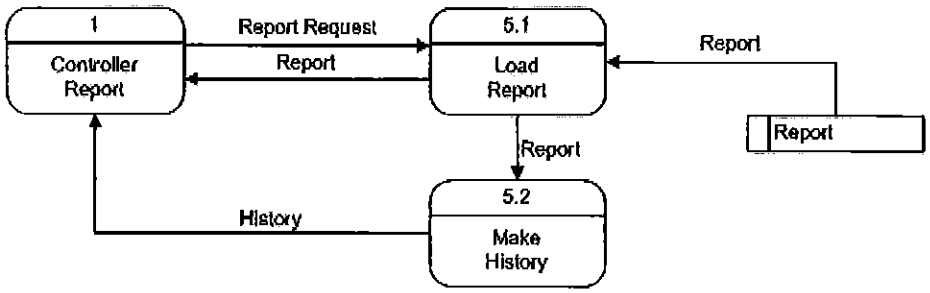


รูปที่ 3.6 กระแสของข้อมูลระดับ 2 ของโปรเซส Controller Chat

โปรเซสที่ 5 Agent Report มี 2 โปรเซสย่อย

โปรเซสที่ 5.1 Load Report มีหน้าที่ในการอ่านค่าข้อมูล Report ที่ได้บันทึกไว้ส่งเข้าโปรเซส Controller Report เพื่อนำไปทำรายงานแสดงผลแก่ผู้ดูแลระบบ

โปรเซสที่ 5.2 Make History มีหน้าที่รับค่า Report และทำการบันทึกเป็น History และส่งเข้าโปรเซส Controller Report เพื่อให้นำไปทำรายงานแสดงผลแก่ผู้ดูแลระบบ



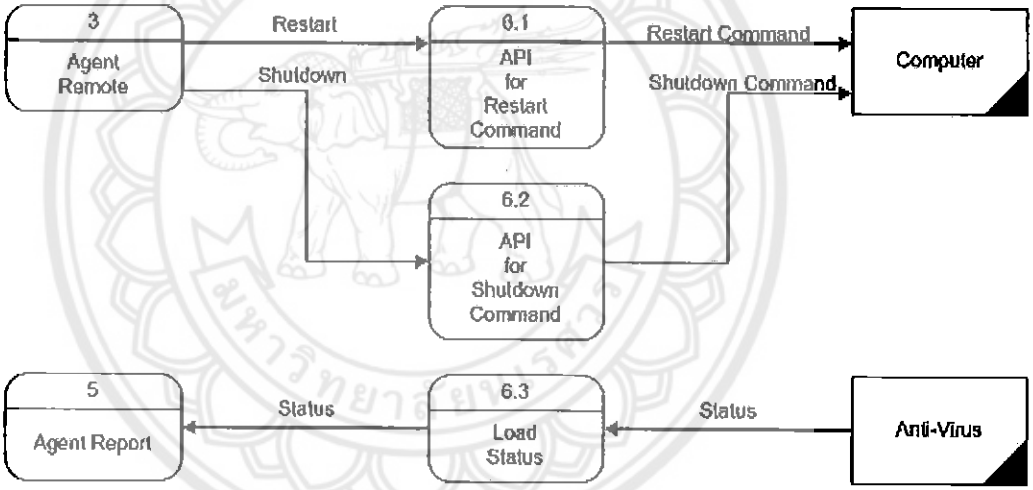
รูปที่ 3.7 กระแสของข้อมูลระดับ 2 ของโปรเซส Agent Report

โปรเซสที่ 6 API แบ่งการทำงานเป็นสองส่วน คือ ส่วนที่เป็นการควบคุมการทำงานของคอมพิวเตอร์และส่วนที่เป็นการอ่านค่า Status จาก Anti-Virus มี 3 โปรเซสย่อย

โปรเซสที่ 6.1 API for Restart Command มีหน้าที่สร้างคำสั่งในการ Restart เครื่องลูกข่าย

โปรเซสที่ 6.2 API for Shutdown Command มีหน้าที่สร้างคำสั่งการ Shutdown เครื่องลูกข่าย

โปรเซสที่ 6.3 Load Status มีหน้าที่อ่านค่า Title Bar เพื่ออ้างอิงการทำงานของระบบ

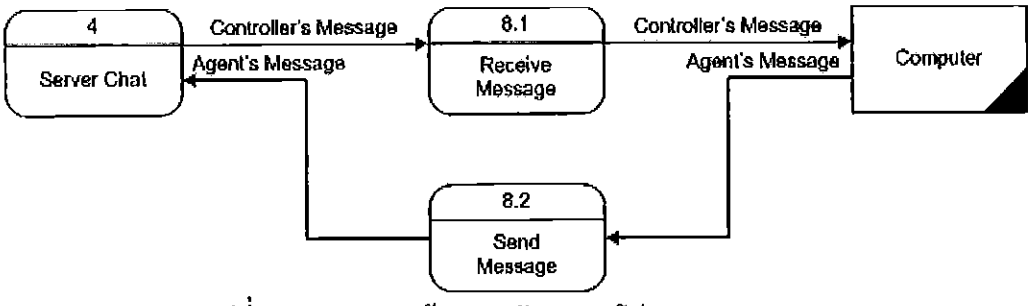


รูปที่ 3.8 กระแสของข้อมูลระดับ 2 ของโปรเซส API

โปรเซสที่ 8 Agent Chat มี 2 โปรเซสย่อย

โปรเซสที่ 8.1 Receive Message มีหน้าที่รับข้อความจากผู้ดูแลระบบและส่งข้อความดังกล่าวให้กับเครื่องลูกข่าย

โปรเซสที่ 8.2 Send Message มีหน้าที่ส่งข้อความจากเครื่อง Agent และส่งข้อความดังกล่าวให้กับผู้ดูแลระบบ



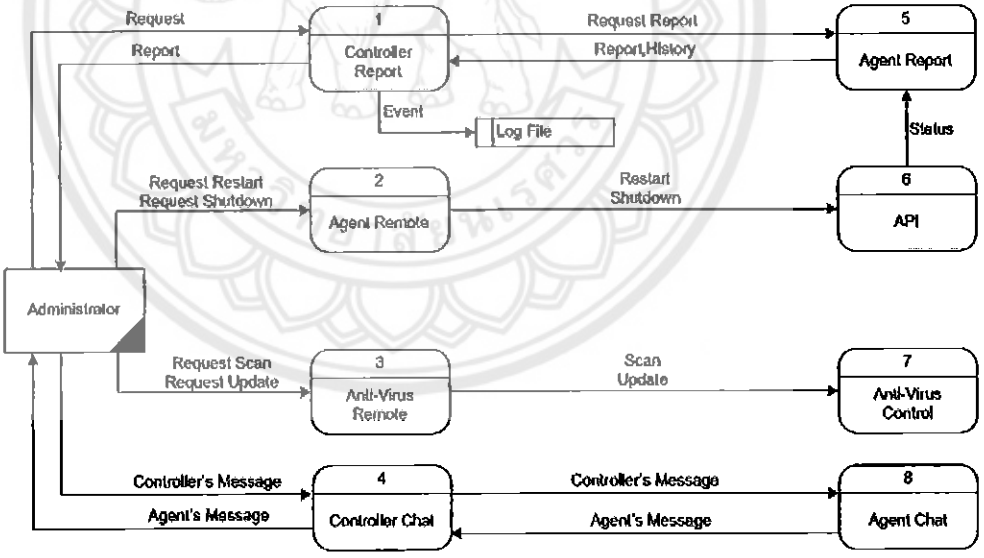
รูปที่ 3.9 กระแสของข้อมูลระดับ 2 ของโปรเซส Agent Chat

3.4 การออกแบบในส่วนของอินเตอร์เฟสที่ติดต่อกับผู้ใช้ของโปรแกรม

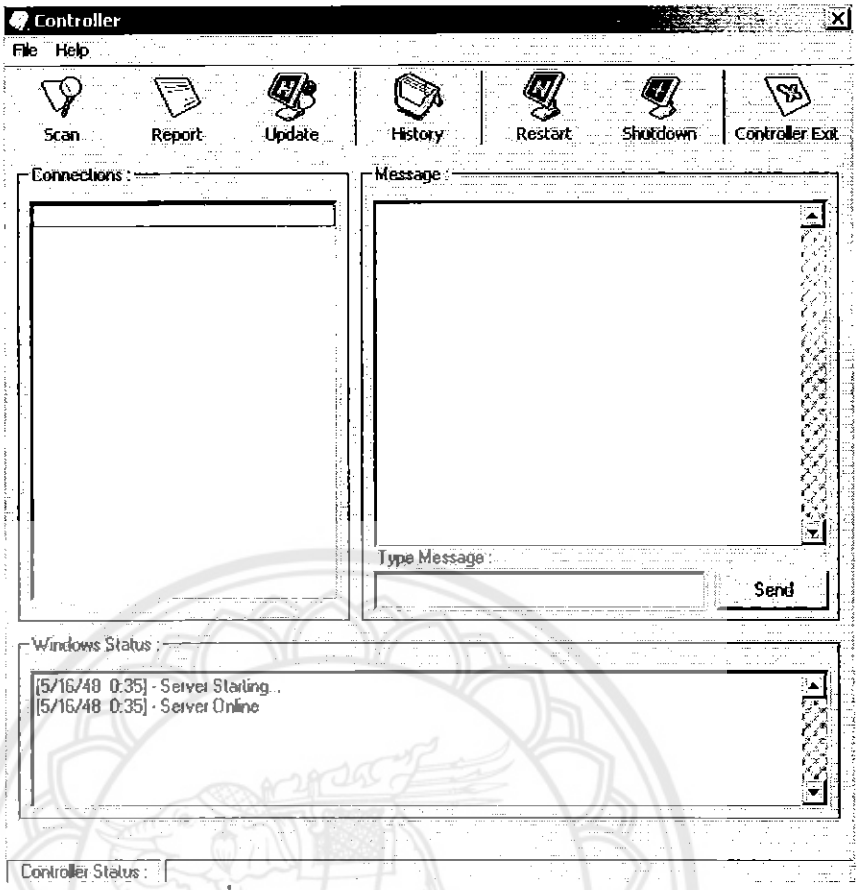
ในการออกแบบส่วนของการอินเตอร์เฟสได้แบ่งออกเป็นสองส่วน คือ ส่วนที่ติดต่อกับผู้ดูแลระบบและส่วนที่ติดต่อกับเครื่องลูกข่าย สามารถแสดงผลของการออกแบบอินเตอร์เฟสที่ติดต่อกับผู้ดูแลระบบและเครื่องลูกข่ายได้ดังนี้

3.4.1 ส่วนของการออกแบบ Controller

ผลของการออกแบบอินเตอร์เฟสที่ใช้ติดต่อผู้ดูแลระบบในส่วนของ Controller ดังรูป 3.11



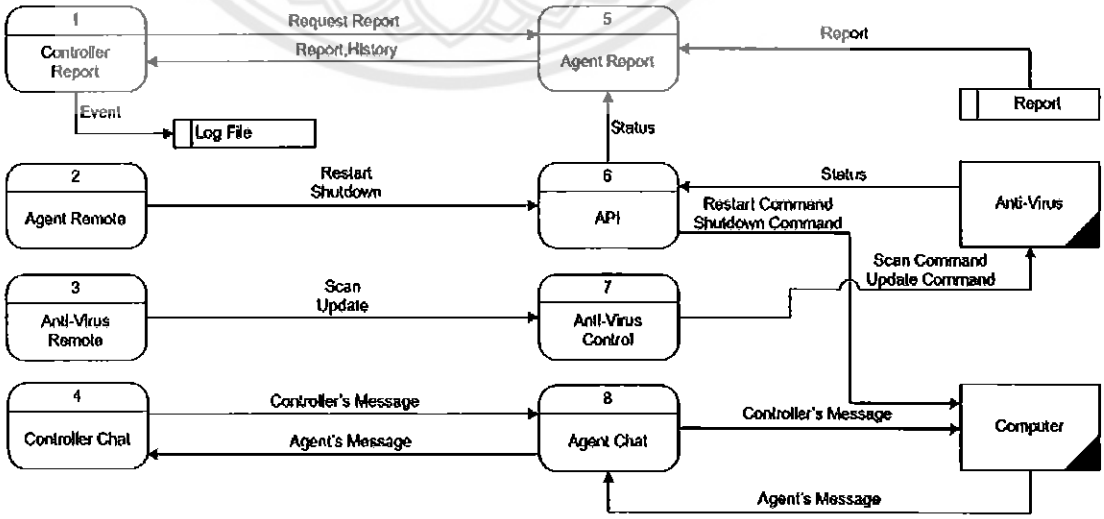
รูปที่ 3.10 การทำงานของ Controller



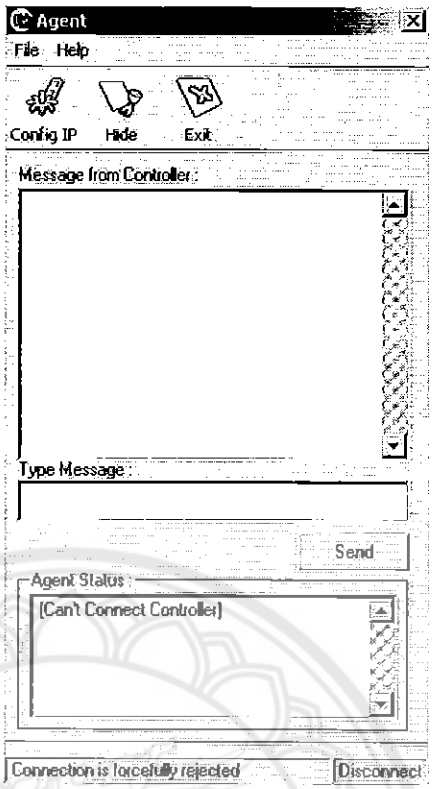
รูปที่ 3.11 ส่วนของการอินเตอร์เฟส Controller

3.4.2 ส่วนการออกแบบเครื่องลูกข่าย

ผลของการออกแบบอินเตอร์เฟสที่ใช้ติดต่อกับเครื่องลูกข่าย แสดงได้ดังรูป 3.13



รูปที่ 3.12 การทำงานของ Agent



รูปที่ 3.13 ส่วนของการอินเตอร์เฟสเครื่องลูกข่าย (Agent)

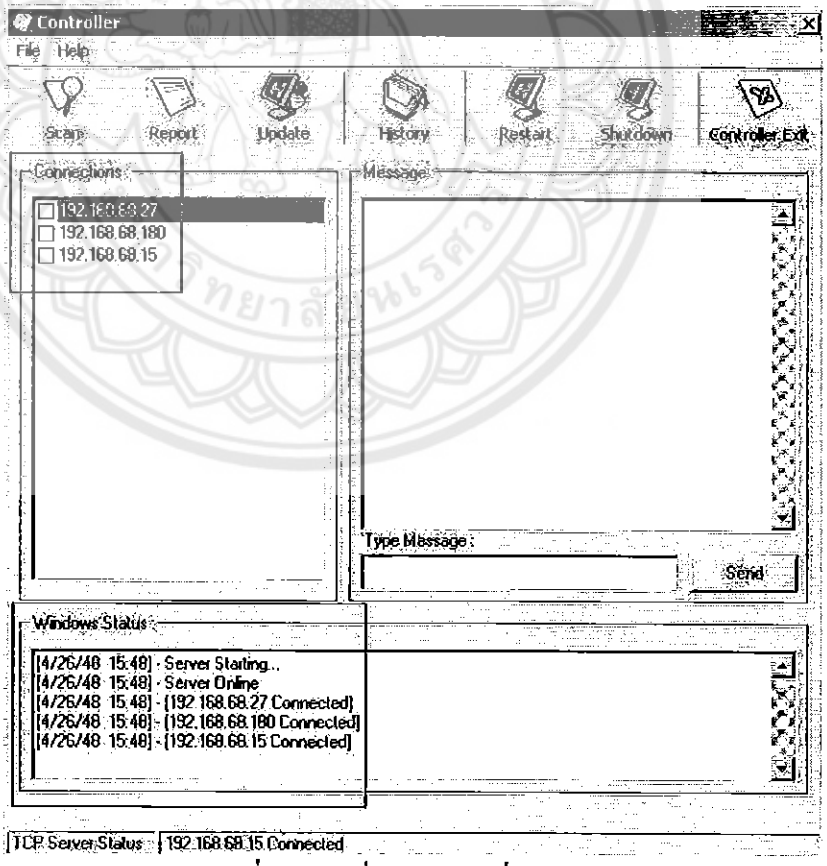
บทที่ 4
การทดสอบและวิเคราะห์การทำงาน

เนื้อหาในการทดสอบและวิเคราะห์การทำงานจริงของโปรแกรม เพื่อจะได้ทราบว่าโปรแกรมที่พัฒนาจะสามารถปฏิบัติงานได้ตามที่ต้องการหรือไม่ โดยการทดสอบนั้นสามารถแบ่งการทดสอบของโปรแกรมเป็นสองส่วนใหญ่ ๆ คือ

4.1 การทดลองระหว่างการพัฒนาโปรแกรม

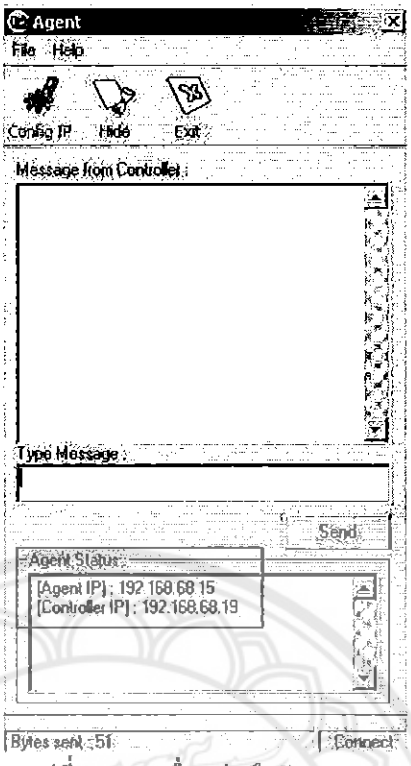
4.1.1 ผลทดสอบการเชื่อมต่อระหว่างตัวควบคุมกับเครื่องลูกข่าย

หลังจากกำหนดค่าพอร์ทและไอพีสำหรับการเชื่อมต่อเรียบร้อยแล้ว ให้ทำการรันตัวควบคุมและรอการเชื่อมต่อจากเครื่องลูกข่าย สิ่งที่แสดงการเชื่อมต่อสำเร็จนั้นตัวควบคุมสามารถรู้ได้โดยค่าไอพีของเครื่องลูกข่ายจะแสดงบนหน้าจอของตัวควบคุมและรายงานสถานะการเชื่อมต่อทางกล่องข้อความด้านล่างของตัวควบคุมดังรูป 4.1



รูปที่ 4.1 การเชื่อมต่อจากเครื่องลูกข่าย

ส่วนที่จะแสดงการเชื่อมต่อสำเร็จนั้นเครื่องลูกข่ายสามารถทราบได้จากการแสดงสถานะการเชื่อมต่อถึงตัวควบคุมโดยจะแสดงค่าไอพีของเครื่องและตัวควบคุมดังรูป 4.2



รูปที่ 4.2 การเชื่อมต่อถึงตัวควบคุม

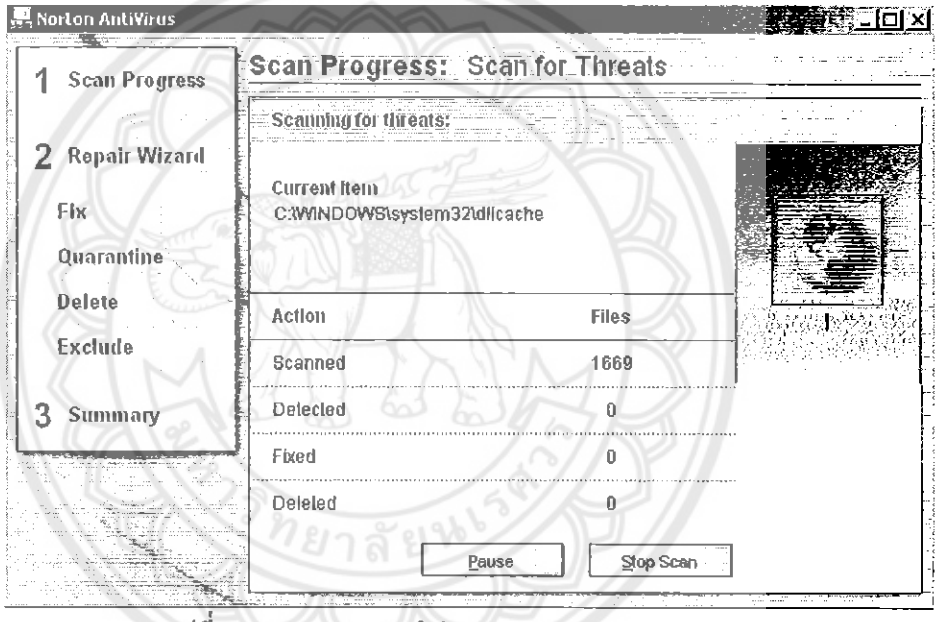
4.1.2 ผลการทดลองการสั่งงานจาก Agent ไปตัวเครื่อง

เป็นการทดสอบสั่งโปรแกรมต่อต้านไวรัสให้ทำงานผ่านทาง Command Line ในระบบดอส (Dos Mode) ว่าสามารถปฏิบัติงานตามได้หรือไม่ นอกจากนี้ยังทำการทดสอบดึงข้อมูลที่จำเป็นต่อการทำรายงานเสนอผู้ดูแลระบบ ว่าสามารถนำค่าดังกล่าวไปใช้ได้เลยหรือไม่ ได้ทำการทดสอบกับโปรแกรมต่อต้านไวรัส 3 โปรแกรมด้วยกัน คือ โปรแกรม Norton Anti-Virus 2005 โปรแกรม McAfee และโปรแกรม AntiVir โดยรายละเอียดการทดสอบมีดังต่อไปนี้

- โปรแกรม Norton Anti-Virus 2004

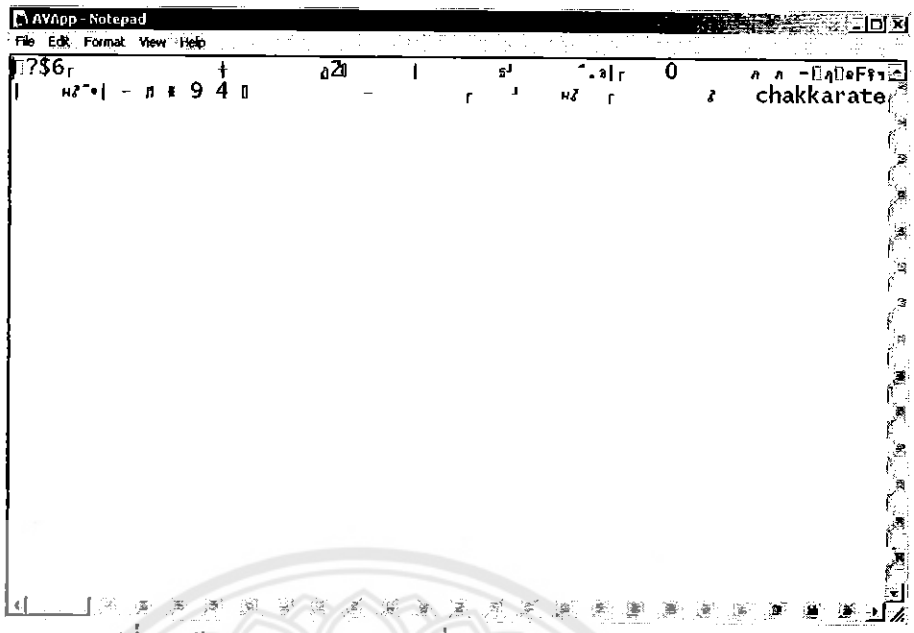
เป็นโปรแกรมต่อต้านไวรัสที่ได้รับความนิยมโปรแกรมหนึ่งโดยค่าย Symantec เป็นเจ้าของลิขสิทธิ์ ค่าใช้จ่ายเกี่ยวกับค่าลิขสิทธิ์นั้น เป็นราคา 24.97 เหรียญสหรัฐฯ (ประมาณ 1,000 บาท) การทดสอบสั่งงานผ่านทาง Command Line สามารถทำงานได้ดี โดยรูปแบบของคำสั่งเป็นไปตามรูปที่ 4.3

รูปที่ 4.3 การสั่งงานผ่าน Command Line ของโปรแกรม Norton Anti-Virus



รูปที่ 4.4 การทำงานของโปรแกรม Norton AntiVirus 2004

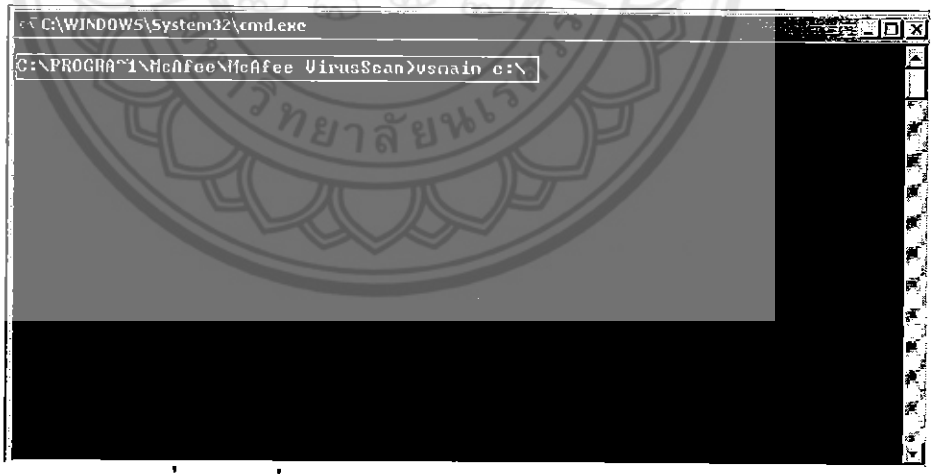
หลังจากการทดสอบสั่งงานผ่านทาง Command Line พบว่าสามารถทำงานได้ตามวัตถุประสงค์ต่อไป ทำการทดสอบดึงข้อมูลที่จำเป็นต่อการทำรายงานเพื่อนำเสนอต่อผู้ดูแลระบบต่อไป ในการทดสอบเกี่ยวกับเรื่องนี้ โปรแกรม Norton AntiVirus ไม่สนับสนุนการอ่านไฟล์แบบเปิด (Text File) ข้อมูลที่ได้ไม่สามารถนำไปทำเป็นรายงานต่อไปได้ ทำให้ผู้พัฒนาโปรแกรมจึงไม่เลือกใช้โปรแกรม Norton AntiVirus ข้อมูลที่ทำการดึงมานั้นแสดงไว้ดังรูป 4.5



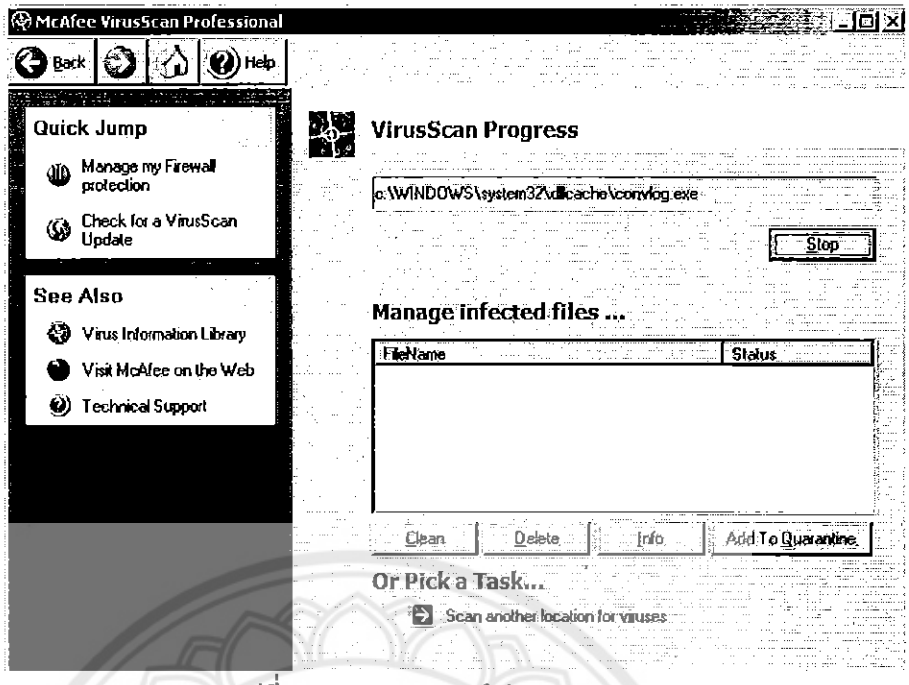
รูปที่ 4.5 ข้อมูลในการทำรายงานที่ได้จากโปรแกรม Norton AntiVirus

- โปรแกรม McAfee

เป็นโปรแกรมต่อต้านไวรัสที่ได้รับความนิยมอีกโปรแกรมหนึ่ง **โดยทาง** McAfeeเป็น**เจ้าของลิขสิทธิ์** ค่าใช้จ่ายเกี่ยวกับค่าลิขสิทธิ์นั้น มีราคาถึง 39.99 เหรียญสหรัฐฯ **(ประมาณ** 1,500 บาท) **การทดสอบสั่งงานผ่านทาง** Command Line สามารถทำงานได้ดี รูปแบบของคำสั่ง**เป็นไปตามรูปที่** 4.6

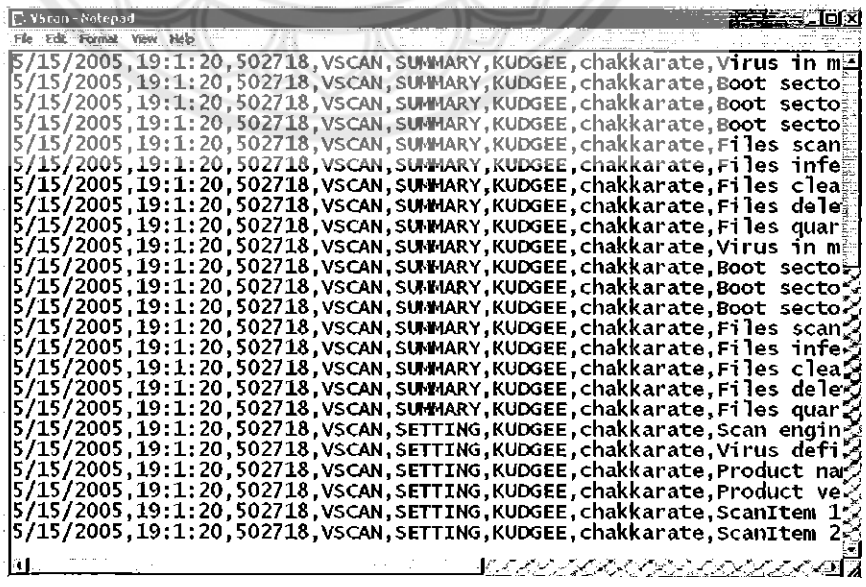


รูปที่ 4.6 การสั่งงานผ่าน Command Line ของโปรแกรม McAfee



รูปที่ 4.7 การทำงานของโปรแกรม McAfee

การทดสอบสั่งงานผ่านทาง Command Line พบว่าทำงานได้ตามวัตถุประสงค์เมื่อทำการทดสอบดึงข้อมูลที่จำเป็นต่อการทำรายงานเพื่อนำเสนอต่อผู้ดูแลระบบต่อไป ในการทดสอบเกี่ยวกับเรื่องนี้ โปรแกรม McAfee ข้อมูลที่ได้นั้นเป็นข้อมูลที่ไม่เอื้อประโยชน์ต่อการทำรายงาน เพราะข้อมูลที่ได้ไม่เป็นระเบียบมากนัก ทำให้ผู้พัฒนาไม่สามารถเลือกโปรแกรม McAfee นี้ให้เป็นโปรแกรมที่เหมาะสมกับระบบได้

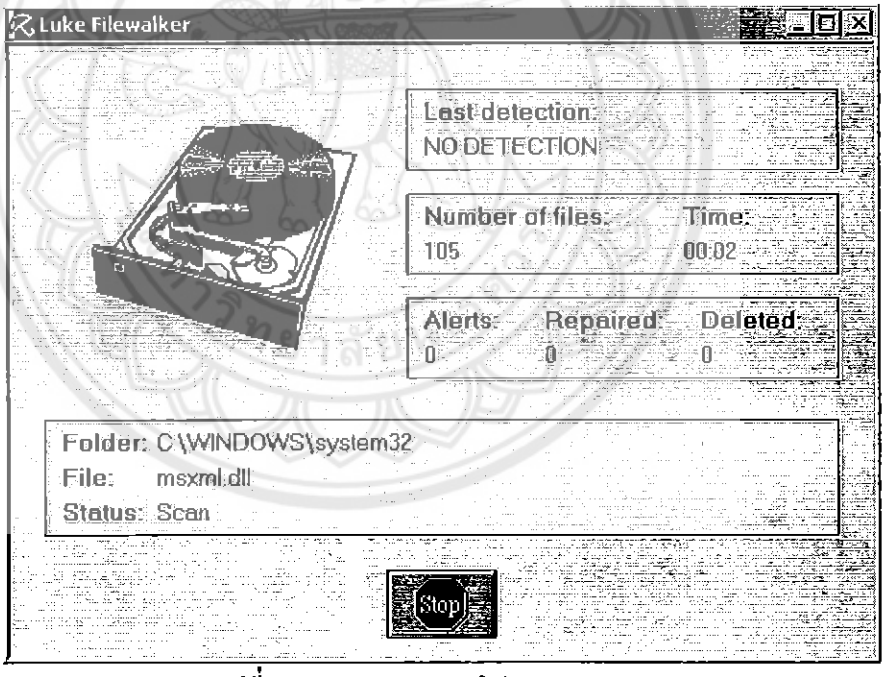


รูปที่ 4.8 ข้อมูลในการทำรายงานที่ได้จากโปรแกรม McAfee

- โปรแกรม AntiVir

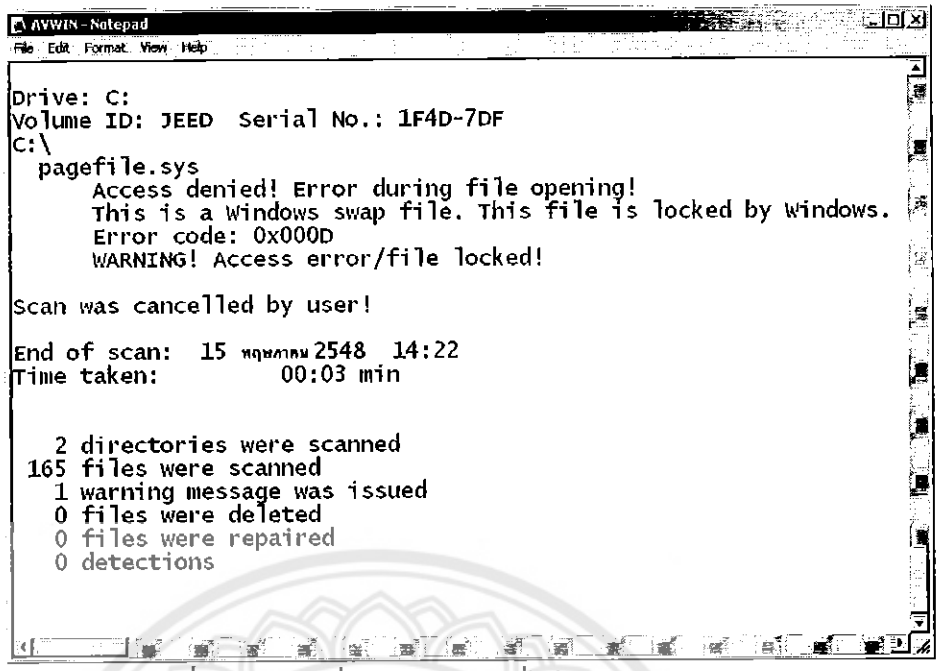
เป็นโปรแกรมต่อต้านไวรัสที่เปิดให้นำโปรแกรมไปใช้ โดยไม่เสียค่าลิขสิทธิ์แต่อย่างใด การทดสอบสั่งงานผ่านทาง Command Line สามารถทำงานได้ดี รูปแบบของคำสั่งเป็นไปตามรูปที่ 4.9

รูปที่ 4.9 การสั่งงานผ่าน Command Line ของโปรแกรม AntiVir



รูปที่ 4.10 การทำงานของโปรแกรม AntiVir

หลังจากการทดสอบสั่งงานผ่านทาง Command Line พบว่าสามารถทำงานได้ตามวัตถุประสงค์ เมื่อนำข้อมูลส่วนที่จะไปทำเป็นรายงานนั้นสามารถสร้างเนื้อหาของการรายงานได้ มีความเป็นระเบียบ โดยในข้อมูลระบุเหตุการณ์ที่จำเป็นไว้ เช่น วันเวลาที่ทำการตรวจสอบ จำนวนไฟล์ที่ผ่านการตรวจ จำนวนไวรัสที่พบ เป็นต้น ด้วยคุณสมบัติที่เหมาะสมต่อโปรแกรมของระบบ ทางผู้พัฒนาจึงได้เลือกโปรแกรมต่อต้านไวรัสนี้



รูปที่ 4.11 ข้อมูลเพื่อการทำรายงานที่ได้จากโปรแกรม AntiVir

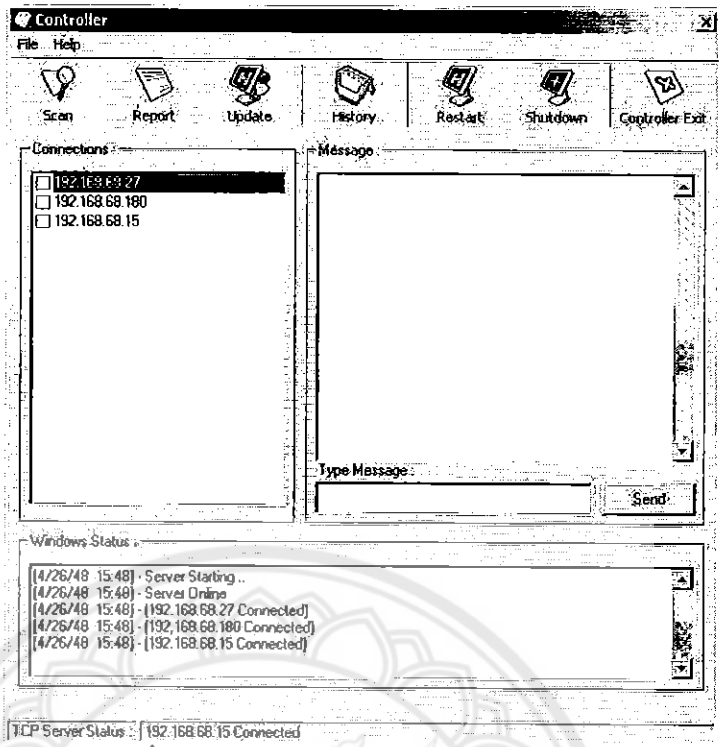
4.1.3 ผลการทดลองการใช้ระบบปฏิบัติการวินโดว์ Versionอื่น ๆ

โปรแกรมอำนวยความสะดวกในการจัดการไวรัสสำหรับผู้ดูแลระบบ สามารถรองรับระบบปฏิบัติการวินโดว์ 98/ME/2000/XP ในการทดสอบ ทางผู้พัฒนาได้ทดลองโปรแกรมกับระบบปฏิบัติการวินโดว์ XP กับรูปแบบคำสั่งโปรแกรมทำงานมีประสิทธิภาพดี

ในส่วนหลังจะเป็นการทดสอบโปรแกรม โดยกระทำเหมือนปฏิบัติงานจริงโดยจำลองสถานการณ์ที่เป็นไปได้ระหว่างการปฏิบัติงาน ทั้งนี้ได้รวบรวมปัญหาที่พบและแนวทางในการแก้ไขไว้แล้ว ผู้ศึกษาควรทำความเข้าใจเพื่อเป็นประโยชน์แก่การปรับแต่งโปรแกรมต่อไป

4.2 การทดสอบส่วนการควบคุม (Controller)

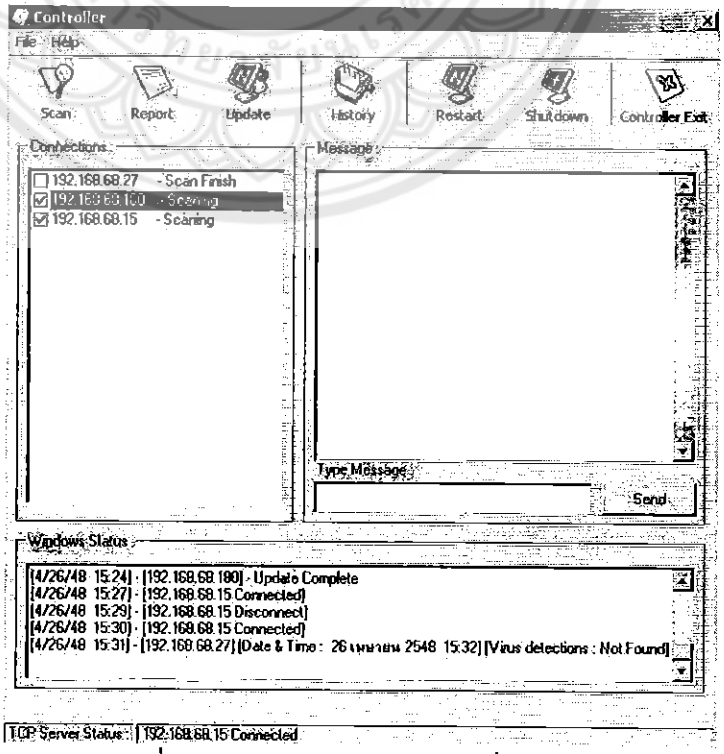
ในส่วนนี้เป็นการให้ตัวควบคุมทำงาน โดยตัวควบคุมจะเป็นส่วนที่คอยส่งคำสั่งต่าง ๆ ไปยังเครื่องลูกข่ายที่เป็นเครื่องเป้าหมาย และโปรแกรมยังสามารถรายงานสถานะของเครื่องลูกข่ายนั้น ๆ ว่าปฏิบัติงานในส่วนใดของคำสั่งอยู่ หรือมีการเปิดเครื่องเพื่อรอรับคำสั่งอยู่หรือไม่ แสดงได้ดังรูป 4.12 ซึ่งแสดงการเชื่อมต่อกันระหว่างตัวควบคุมกับเครื่องลูกข่าย



รูปที่ 4.12 การทำงานของตัวควบคุม (Controller)

4.2.1 การส่งคำสั่งตรวจสอบเครื่องลูกข่าย (Send Scan Command)

เมื่อเครื่องควบคุมส่งคำสั่ง Scan ไปยังเครื่องลูกข่าย จะมีการรายงาน**สถานะของเครื่องลูกข่าย**ออกทางหน้าจอของเครื่องควบคุม ดังรูป 4.13

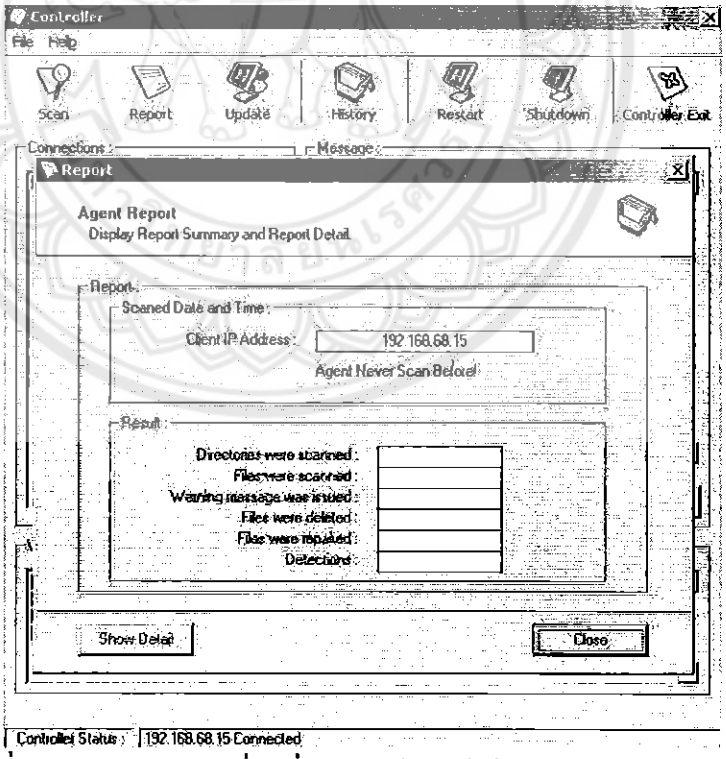


รูปที่ 4.13 การรายงานสถานะของเครื่องลูกข่าย

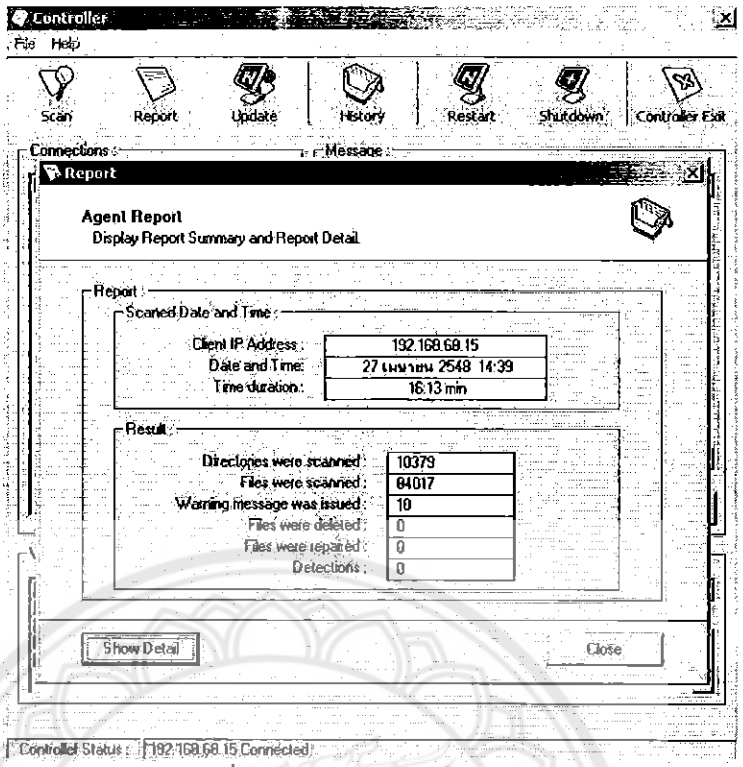
4.2.2 ส่วนของการรายงาน (Report)

ในส่วนนี้เป็นการดูผลของการปฏิบัติงานหลังจากการ Scan โดยเนื้อหาของการรายงานนั้นจะกล่าวถึงวันที่ทำการตรวจสอบ, เวลาของการตรวจสอบและบอกถึงการตรวจพบไวรัสในเครื่องลูกข่ายหรือไม่ ผลของการรายงานจะออกมา 2 ประการ ดังนี้

- หากเครื่องลูกข่ายไม่เคยมีการ Scan มาก่อน การประมวลรายงานจะไม่สามารถทราบข้อมูลที่จะนำมาแสดงผลได้ และจะรายงานไปทางเครื่องควบคุมว่า เครื่องลูกข่ายนั้นไม่เคยได้รับการตรวจสอบมาก่อน ผู้ดูแลระบบควรทำการตรวจสอบเครื่องลูกข่ายดังกล่าวเสียก่อน เพื่อจะได้รับรายงานที่ต้องการได้ เมื่อเครื่องลูกข่ายยังไม่ได้รับการตรวจสอบ การรายงานผลจะแสดง ดังรูป 4.14
- หากเครื่องลูกข่ายตรวจสอบมาแล้ว การรายงานจะแสดงค่าต่าง ๆที่โปรแกรมทำการตรวจสอบไว้ เช่น ไอพีเครื่องลูกข่ายนั้นๆ วันเวลาที่ทำการตรวจสอบ เวลาที่ใช้ในการตรวจสอบเครื่องลูกข่าย จำนวนไฟล์ที่ผ่านการตรวจสอบ จำนวนไวรัสที่ตรวจพบ เป็นต้น หน้าต่างที่แสดงนี้เป็นรายละเอียดเพียงพอสำหรับผู้ดูแลระบบที่จะทราบถึงสถานการณ์ หากผู้ดูแลระบบอยากทราบถึงรายละเอียดที่ลึกกว่านี้ ควรเลือกที่ไอคอน "Show Detail" จะเป็นการรายงานโดยละเอียดของการตรวจสอบ ผลการตรวจสอบนั้น แสดงไว้ดังรูป 4.15



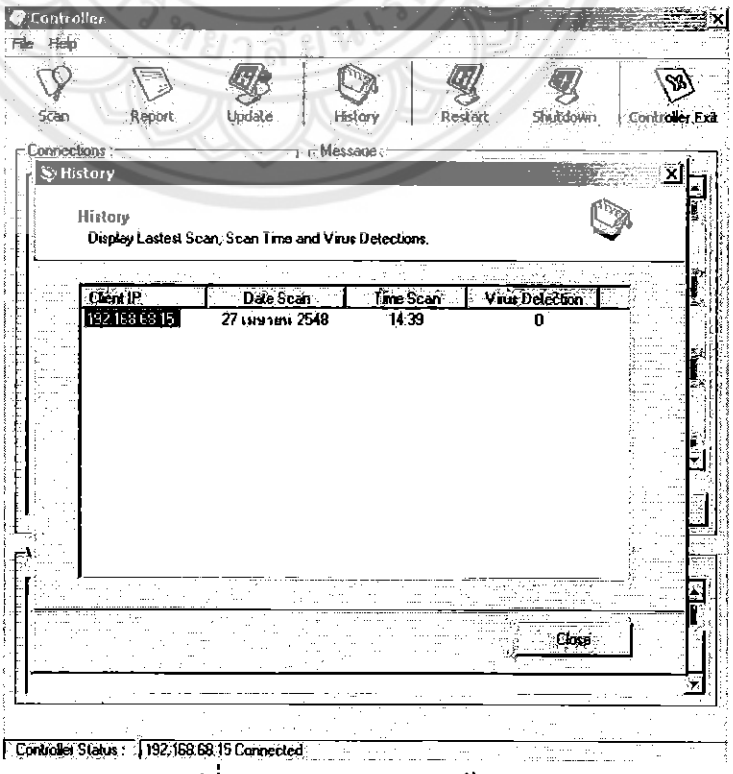
รูปที่ 4.14 ผลการรายงานเมื่อเครื่องลูกข่ายไม่เคยได้รับการตรวจสอบมาก่อน



รูปที่ 4.15 การรายงานเครื่องลูกข่าย

4.2.3 การเรียกดูผลตรวจสอบย้อนหลัง (History)

เป็นส่วนที่รายงานผู้ดูแลระบบให้ทราบถึงการที่เครื่องลูกข่ายที่เคยถูกตรวจสอบครั้งล่าสุด และผลของการตรวจสอบนั้นเป็นเช่นไร โดยเป็นการรายงานคร่าว ๆ ดังรูป 4.16



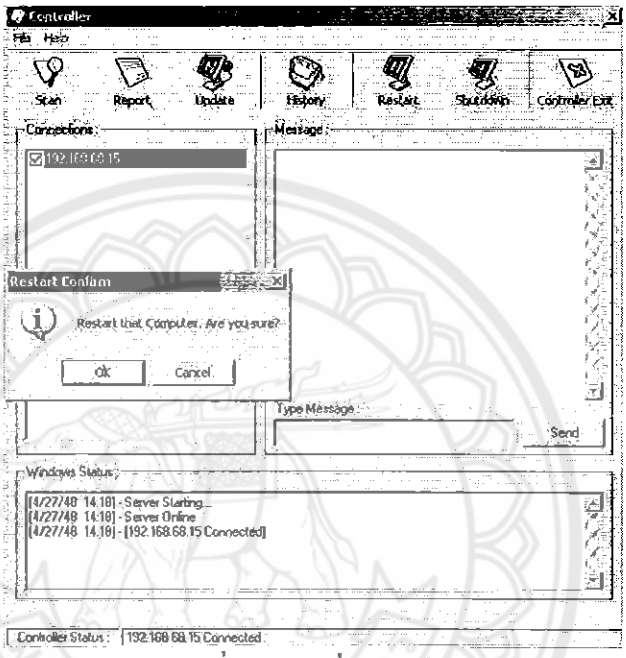
รูปที่ 4.16 ผลการตรวจสอบย้อนหลัง

4.2.4 การสั่งเครื่องลูกข่ายทำการรีสตาร์ท (Send Restart Command)

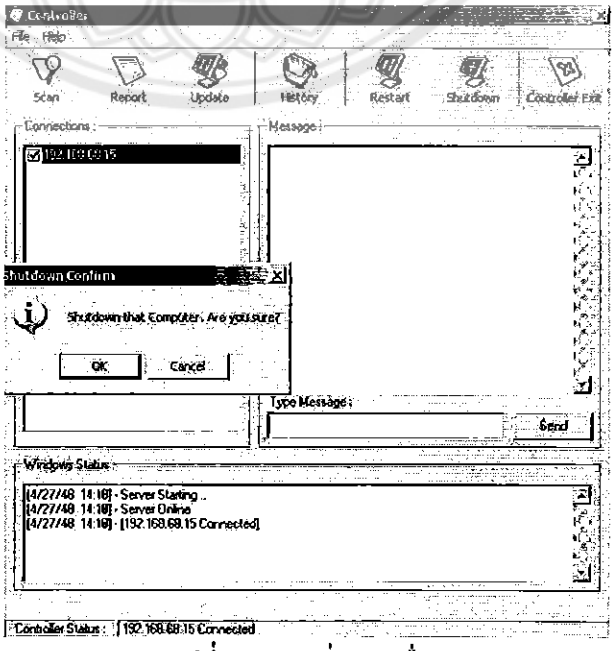
เป็นการสั่งให้เครื่องลูกข่ายกระทำการรีสตาร์ทเครื่องลูกข่ายนั้น โดยทำการขอคำยืนยันจากผู้ดูแลระบบก่อนที่จะส่งคำสั่งไปเครื่องลูกข่าย ดังรูป 4.17

4.2.5 การสั่งเครื่องลูกข่ายทำการปิดเครื่อง (Send Shutdown Command)

คล้ายกระบวนการรีสตาร์ท แต่ผลของปฏิบัติการเป็นการปิดเครื่อง ดังรูป 4.18



รูปที่ 4.17 การสั่งรีสตาร์ท



รูปที่ 4.18 การสั่งปิดเครื่อง

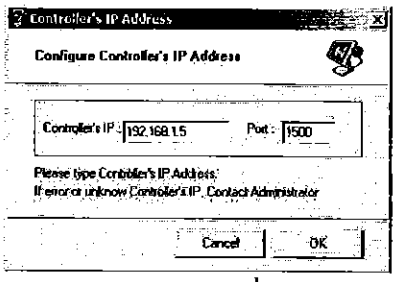
4.2.6 ระบบการสื่อสารภายในจากตัวควบคุม (Controller's Chat)

ใช้ในการติดต่อกับผู้ใช้ เมื่อผู้ดูแลระบบ ต้องการที่จะตรวจสอบหรือดูแลรักษาระบบของเครื่องลูกข่ายในขณะนั้น

รูปที่ 4.19 การสื่อสารจากตัวควบคุมไปยังเครื่องลูกข่าย

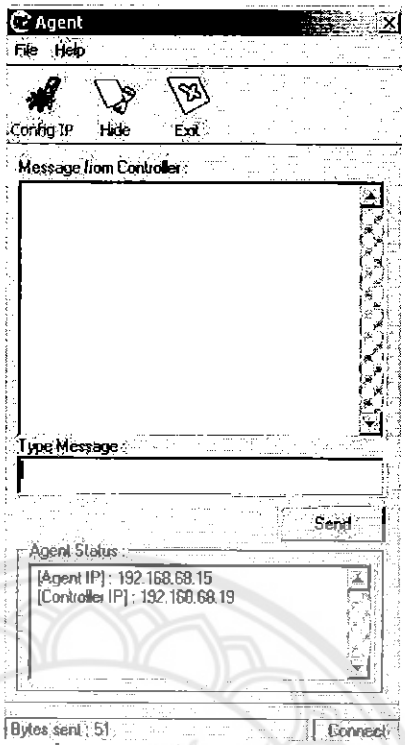
4.3 การทดสอบส่วนของการรับคำสั่งและปฏิบัติงาน (Agent)

เป็นการสังเกตการทำงานของส่วนการรับคำสั่งและปฏิบัติงาน โดยการทำงานแบ่งเป็นสองส่วน ส่วนแรกเป็นการปฏิบัติตามคำสั่งที่ได้รับการร้องขอมาจากตัวควบคุม และอีกส่วนเป็นการตรวจสอบสถานะการทำงานของโปรแกรม Anti-Virus เพื่อส่งค่าสถานะที่จำเป็นกลับไปยังตัวควบคุม โปรแกรมจะทำงานและทำการเชื่อมต่อกับตัวควบคุมได้ ต้องมีการระบุค่า IP Address ของตัวควบคุมเสียก่อน จึงจะสามารถทำการเชื่อมต่อกันและปฏิบัติตามคำร้องขอจากตัวควบคุมได้



รูปที่ 4.20 การกำหนดค่า IP Address ที่ใช้ในการติดต่อตัวควบคุม

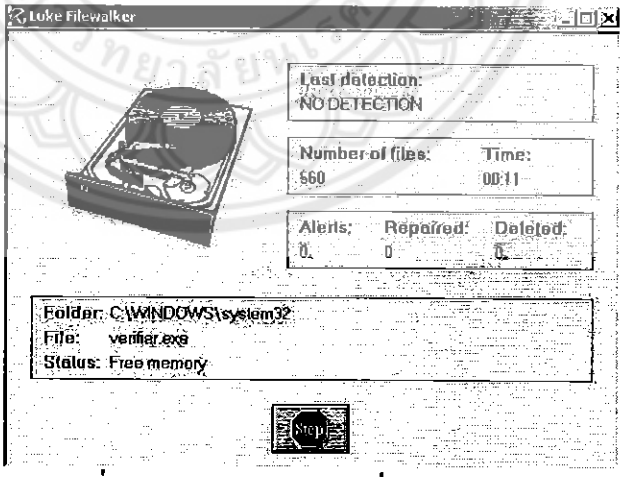
หลังจากการกำหนดค่า IP Address ของตัวควบคุมแล้ว จะเป็นการเชื่อมต่อกับตัวควบคุมอย่างสมบูรณ์ ดังรูป 4.21



รูปที่ 4.21 การเชื่อมต่อกับตัวควบคุม

4.3.1 ระบบการตรวจสอบ (Scan)

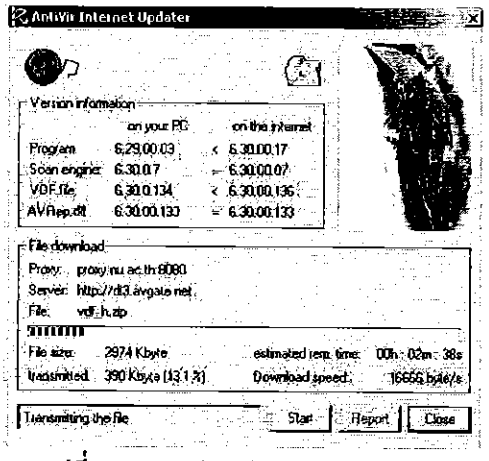
เป็นการรับคำร้องขอให้โปรแกรม Anti-Virus ที่ติดตั้งบนเครื่องลูกข่าย ทำการตรวจสอบไวรัสภายในเครื่องลูกข่ายนั้น ๆ และปฏิบัติงานจนเสร็จสิ้น แสดงดังรูป 4.22



รูปที่ 4.22 การตรวจสอบบนเครื่องลูกข่าย (Scan)

4.3.2 การปรับปรุงระบบโปรแกรม Anti-Virus (Update Anti-Virus)

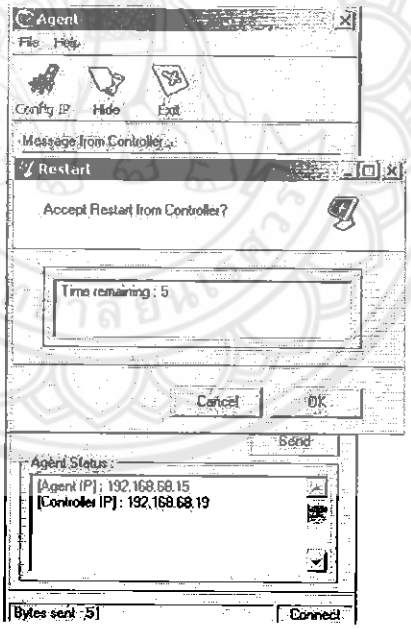
เป็นการรับคำร้องขอ ให้ทำการปรับปรุงระบบ Anti-Virus ให้สามารถจัดการกับไวรัสคอมพิวเตอร์ตัวใหม่ที่เกิดขึ้นตลอดเวลาในระบบอินเตอร์เน็ตได้ ดังรูป 4.23



รูปที่ 4.23 การปรับปรุงระบบ (Update)

4.3.3 การรับคำสั่งเพื่อการรีสตาร์ท (Restart)

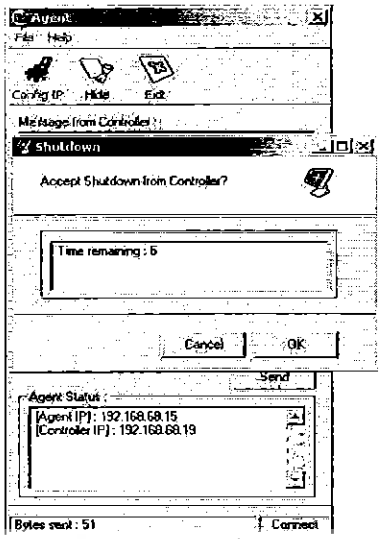
เป็นการปฏิบัติงานที่สั่งให้เครื่องทำการรีสตาร์ทภายหลังจากที่ตัวควบคุมเห็นว่าไม่มีกระบวนการทำงานบนเครื่องนั้นอีก ระบบจะทำการขอคำยอมรับจากเครื่องที่จะทำการรีสตาร์ท หากไม่มีการโต้ตอบภายใน 10 วินาทีระบบจะทำการรีสตาร์ททันที ดังรูป 4.24



รูปที่ 4.24 การรอสัญญาณตอบรับจากเครื่องลูกข่าย สำหรับการรีสตาร์ท

4.3.4 การรับคำสั่งเพื่อการปิดเครื่อง (Shutdown)

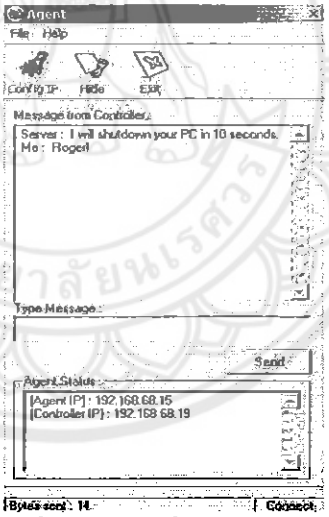
เป็นการปฏิบัติงานที่สั่งให้เครื่องทำการปิดเครื่อง ภายหลังจากที่ผู้ดูแลระบบเห็นว่าไม่มีกระบวนการทำงานบนเครื่องนั้นอีก เมื่อได้รับคำร้องขอดังกล่าว ระบบจะทำการขอคำยอมรับจากเครื่องที่จะทำการปิดเครื่อง เป็นระบบการนับถอยหลัง หากไม่มีโต้ตอบโดยการเลือก "ตกลง" หรือ "ยกเลิก" ภายใน 10 วินาทีระบบจะทำการปิดตัวลง ดังรูป 4.25



รูปที่ 4.25 การรอสัญญาณตอบรับจากเครื่องลูกข่าย สำหรับการปิดเครื่อง

4.3.5 ระบบการสื่อสารภายในจากตัวควบคุม (Agent's Chat)

เป็นการสื่อสารจากผู้ใช้ถึงผู้ดูแลระบบ เมื่อผู้ใช้ต้องการติดต่อกับผู้ดูแลระบบ เพื่อที่จะสนทนาหรือแจ้งปัญหาต่าง ๆ



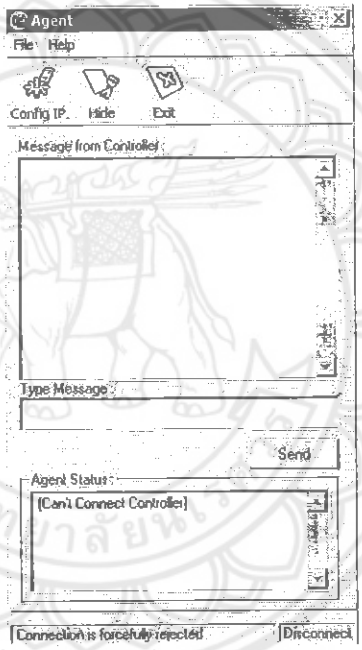
รูปที่ 4.26 การสื่อสารจากเครื่องลูกข่ายไปยังตัวควบคุม

4.4 ปัญหาที่พบขณะปฏิบัติการทดสอบและวิธีแก้ไข

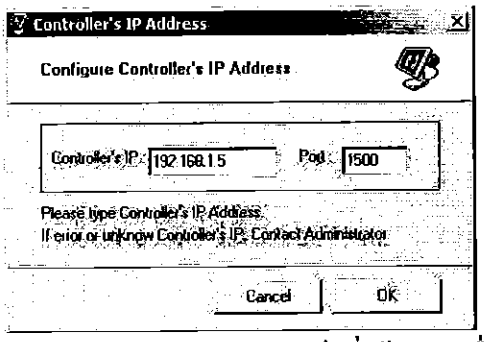
กรณีที่ 1 เมื่อโปรแกรมบนเครื่องบนลูกข่ายทำงานแล้ว ไม่สามารถติดต่อกับตัวควบคุมได้โดยโปรแกรมบนเครื่องลูกข่ายทำการรายงานดังรูป 4.24 สาเหตุมาจากการ กำหนดค่า IP ที่ใช้ติดต่อกับตัวควบคุมไม่ตรงกัน หรือ ตัวควบคุมยังไม่ทำงาน

การแก้ไข 1. ตรวจสอบการทำงานของตัวควบคุมว่าทำงานอยู่หรือไม่

2. ทำการตรวจสอบ IP และ Portของโปรแกรมบนเครื่องลูกข่ายว่าตรงกับ IP ของเครื่องควบคุมหรือไม่ หากไม่ตรงกันควรแก้ไข โดยเลือก "Config IP" บนโปรแกรมของเครื่องลูกข่าย และทำการระบุ IP ที่ถูกต้องอีกครั้ง ดังรูป 4.28



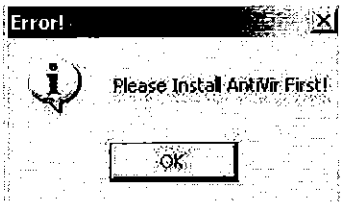
รูปที่ 4.27 การไม่เชื่อมต่อกันของโปรแกรม



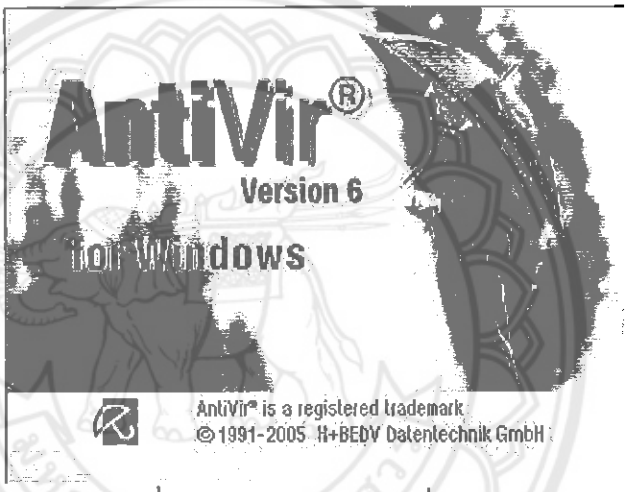
รูปที่ 4.28 การกำหนดค่าไอพีและพอร์ทที่ใช้ในการเชื่อมต่อ

กรณีที่ 2 หากเครื่องลูกข่ายยังไม่ติดตั้งโปรแกรม Anti-Virus ที่เลือกใช้แล้ว เมื่อโปรแกรมทำงานแล้วจะแสดงกล่องข้อความแจ้งขึ้นมา ดังรูป 4.29

การแก้ไข เครื่องลูกข่ายควรติดตั้งโปรแกรม Anti-Virus ที่เหมาะสมกับระบบเสียก่อนซึ่งโปรแกรม Anti-Virus ที่เลือกใช้นี้ คือ โปรแกรม AntiVir



รูปที่ 4.29 กล่องข้อความแจ้งความผิดพลาด

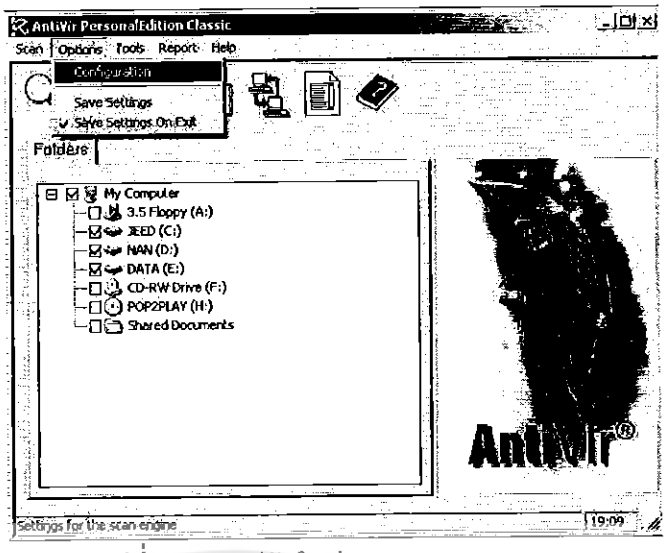


รูปที่ 4.30โปรแกรม AntiVir ที่เลือกใช้

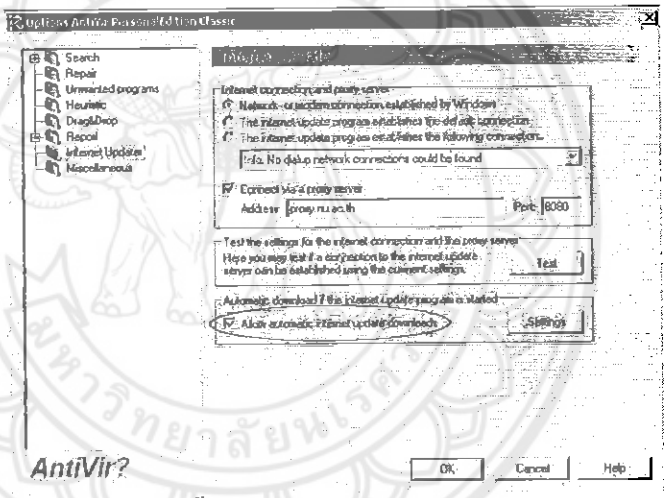
กรณีที่ 3 ส่วนของการส่งคำร้องขอเพื่อปรับปรุงระบบ Anti-Virus (Update Anti-Virus) เครื่องลูกข่ายยังไม่ทำการ Update สาเหตุมาจากไม่ได้มีการปรับ Configure ของโปรแกรม Anti-Virus ให้เป็นการปรับปรุงโปรแกรมอัตโนมัติ

การแก้ไข ควรเข้าไปปรับในส่วนของ Configuration ของโปรแกรม Anti-Virus เพื่อเปลี่ยนระบบของโปรแกรมให้มีการปรับปรุงระบบอัตโนมัติดังขั้นตอนดังต่อไปนี้

1. เข้าไปในส่วนของโปรแกรม AntiVir แล้วเลือก Option --> Configuration จะได้หน้าตาของโปรแกรมดังรูป 4.31
2. เลือกในส่วน Internet Updater ทำการเลือกหัวข้อ "Allow automatic internet update downloads" ดังนี้จะทำให้โปรแกรม Anti-Virus ได้ทำการ Update ระบบอัตโนมัติดังรูป 4.32



รูปที่ 4.31 การปรับในส่วนของ Configuration

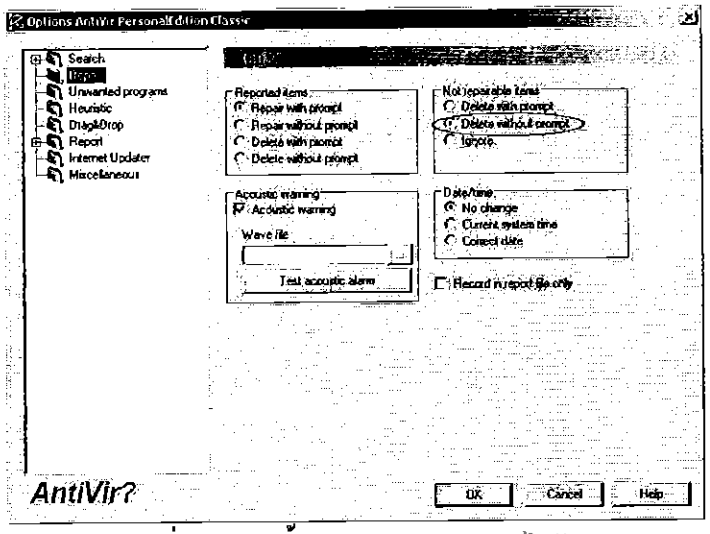


รูปที่ 4.32 การตั้งค่าให้โปรแกรมทำการปรับปรุงอัตโนมัติ

กรณีที่ 4 ในส่วนของการตรวจสอบระบบ (Scan) หากระบบตรวจพบไวรัสที่ไม่สามารถจัดการได้การตรวจสอบก็จะหยุด และไม่สามารถปฏิบัติงานต่อได้

การแก้ไข 1. เข้าไปในส่วนของโปรแกรม AntiVir แล้วเลือก Option --> Configuration จะได้หน้าตาของโปรแกรมดังรูป 4.31 เช่นกัน

2. เลือกในส่วน Repair จะพบหน้าต่างสำหรับจัดการไวรัสดังรูป 4.33 เลือกในส่วนของ "Not repairable item" ทำการเลือกหัวข้อ "Delete without prompt" เพื่อเป็นทำการลบข้อมูลในส่วนที่ตรวจพบนั้นอัตโนมัติ จะทำให้ระบบยังคงทำงานต่อไปได้

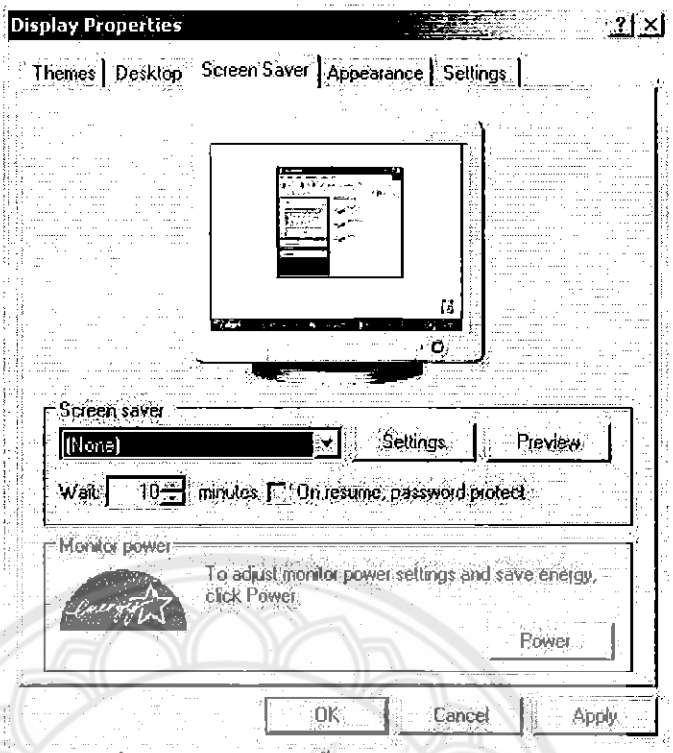


รูปที่ 4.33 การตั้งค่าสำหรับการจัดการไวรัส

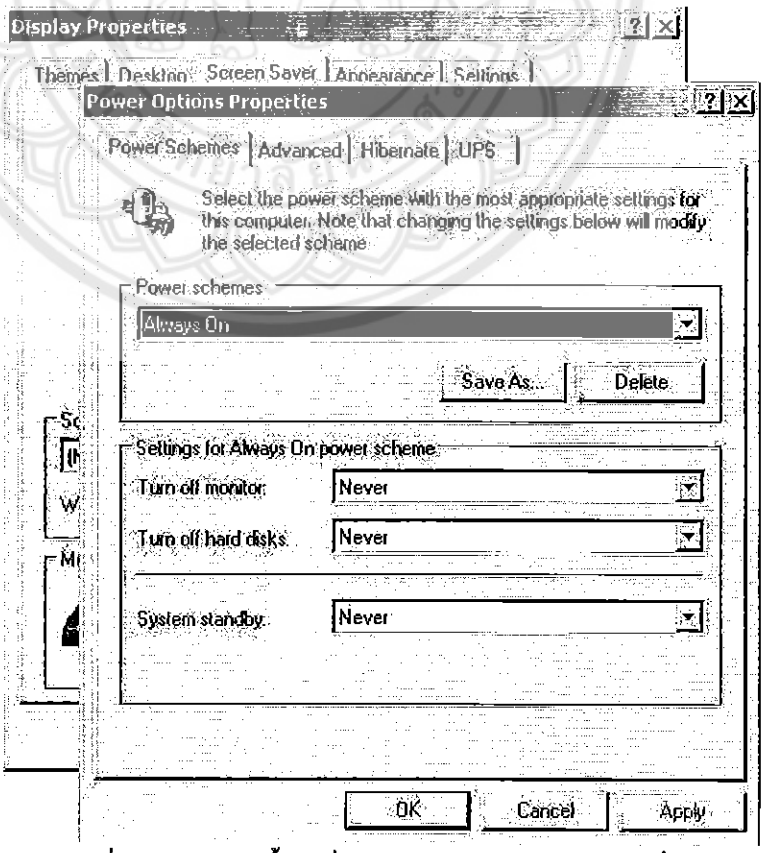
กรณีที่ 5 หากเครื่องลูกข่ายมีการปรับให้เครื่องมี Screen Saver เพื่อการพักหน้าจอ บางครั้งอาจมีการปรับให้มีการปิดหน้าจอหรือทำการปิดเครื่องตามเวลาที่กำหนดไว้ ด้วยเหตุการณ์ดังกล่าวทำให้การหยุดปฏิบัติงานของโปรแกรม มีผลให้การดูแลรักษาทำงานได้ไม่เต็มที่

การแก้ไข 1. เข้าไปเลือกในส่วนของ Display Properties โดยการคลิ้กขวาแล้วทำการเลือก Properties-->Screen Saver หรือ Start-->Settings-->Display-->Screen Saver จะสามารถเข้าสู่หน้าต่างเดียวกันดังรูป 4.34 ได้ เลือกดูในหัวข้อ Screen Saver ควรปรับให้เป็น "None" เพื่อไม่ให้เครื่องลูกข่ายเข้าสู่ระบบ Screen Saver ได้

2. ต่อไปเลือกดูหัวข้อ "Monitor power" และคลิ้กที่ "Power..." จะเข้าไปในส่วนหน้าต่างของ Power Option Properties เลือกในส่วน "Settings for Always On power scheme" ปรับให้เป็น "Never" ทั้งหมดเพื่อปรับให้ไม่มีการปิดหน้าจอ, ไม่มีการปิดเครื่องและไม่มีการเข้าสู่ระบบ Stand by ดังรูป 4.35 เมื่อเรียบร้อยแล้วตอบตกลงทั้งสองหน้าต่าง เพียงเท่านี้เครื่องลูกข่ายจะทำงานตลอดเวลาจนกว่าผู้ดูแลระบบจะส่งคำร้องขอในการปิดเครื่อง



รูปที่ 4.34 หน้าต่างการตั้งค่าเกี่ยวกับ Screen Saver



รูปที่ 4.35 หน้าต่างตั้งค่าเกี่ยวกับการปิดหน้าจอและปิดเครื่อง

บทที่ 5
สรุปผลการดำเนินงานและข้อเสนอแนะ

เนื้อหาในบทนี้ กล่าวถึงการสรุปผลของการดำเนินงานที่ได้จากการศึกษาและการจัดทำโปรแกรมอำนวยความสะดวกในการจัดการไวรัสสำหรับผู้ดูแลระบบ จนสามารถสร้างโปรแกรมที่ผู้พัฒนาคาดว่ามีประสิทธิภาพเพียงพอต่อการดูแลรักษาระบบเครือข่าย ด้านการจัดการเกี่ยวกับไวรัสคอมพิวเตอร์ ซึ่งจะช่วยให้ลดขั้นตอนการปฏิบัติงานและอำนวยความสะดวกให้กับผู้ดูแลระบบมากขึ้น

5.1 สรุปผลการดำเนินโครงงาน

จากผลการดำเนินงาน จัดทำโปรแกรมอำนวยความสะดวกในการจัดการไวรัสสำหรับผู้ดูแลระบบ มีจุดประสงค์เพื่อช่วยให้ผู้ดูแลระบบสามารถดูแลรักษาระบบเครือข่าย ทางด้านการจัดการไวรัสคอมพิวเตอร์ ส่งผลให้ระบบเครือข่ายสามารถทำงานได้อย่างมีประสิทธิภาพ และยังช่วยลดขั้นตอนการปฏิบัติงานที่เคยจัดการทีละเครื่อง เปลี่ยนเป็นการจัดการหลาย ๆ เครื่องกับระบบเครือข่ายโดยผ่านทางหน้าจอของผู้ดูแลระบบเอง ทั้งนี้ยังลดปัญหาการแพร่กระจายของไวรัสในระบบเครือข่าย เนื่องจากโปรแกรมสามารถสั่งการตรวจสอบทั้งระบบเครือข่ายได้โดยความสามารถของโปรแกรม Anti-Virus ทำการจัดการไวรัสในขั้นตอนของกระบวนการตรวจสอบ ได้โดยอัตโนมัติ (โปรแกรม Anti-Virus นั้นต้องได้รับการตั้งค่าที่ถูกต้องแล้ว)

จากผลการดำเนินงานและการทดสอบพบว่า โปรแกรมอำนวยความสะดวกในการจัดการไวรัสสำหรับผู้ดูแลระบบ สามารถทำงานได้ตรงตามวัตถุประสงค์ของโครงงาน

5.2 ประโยชน์ที่ได้รับจากการทำโครงงาน

1. ได้รับความรู้ความเข้าใจเกี่ยวกับการติดต่อสื่อสารผ่านทางโปรโตคอล TCP/IP
2. ได้รับความรู้ความเข้าใจเกี่ยวกับการพัฒนาโปรแกรมด้วย Microsoft Visual Basic
3. ได้รับความรู้ความเข้าใจเกี่ยวกับการพัฒนาโปรแกรมผ่านระบบเครือข่ายโดยการใช้โปรแกรม Microsoft Winsock Control
4. ได้รับความรู้ความเข้าใจเกี่ยวกับการเรียกใช้ฟังก์ชันวินโดวส์เอพีไอ (Windows API)

5.3 ความสามารถของโปรแกรมอำนวยความสะดวกในการจัดการไวรัสสำหรับผู้ดูแลระบบ

1. สามารถสั่งงานให้โปรแกรมต่อต้านไวรัสบนเครื่องลูกข่ายทำการตรวจสอบไวรัสในเครื่องดังกล่าวได้
2. สามารถสั่งงานให้เครื่องลูกข่ายทำการปรับปรุงโปรแกรมต่อต้านไวรัสในเครื่องดังกล่าวได้
3. สามารถสั่งการให้เครื่องลูกข่ายทำการรีสตาร์ทและชัตดาวน์เครื่องดังกล่าวได้
4. สามารถแสดงรายงานการตรวจสอบไวรัสในปัจจุบันของเครื่องลูกข่ายแต่ละเครื่องได้

5. สามารถแสดงประวัติการตรวจสอบไวรัสในระบบเครือข่ายได้
6. สามารถทำการสั่งงานครั้งละหลาย ๆ เครื่องพร้อมกันได้

5.4 ข้อจำกัดของโปรแกรมอำนวยความสะดวกในการจัดการไวรัสสำหรับผู้ดูแลระบบ

1. สามารถทำงานได้บนระบบปฏิบัติการ Windows 98/ME/2000/XP เท่านั้น
2. จำเป็นต้องมีโปรแกรมต่อต้านไวรัสติดตั้งบนเครื่องลูกข่าย ซึ่งโปรแกรมนี้สามารถทำงานร่วมกับโปรแกรม AntiVir ได้เท่านั้น เนื่องจากโปรแกรม AntiVir สามารถสั่งงานผ่านทาง Command Line ได้ และสามารถส่งค่าที่จำเป็นต่อการทำรายงานในรูปแบบของแฟ้มข้อมูลตัวอักษรได้
3. ในระหว่างการปฏิบัติงานจะต้องไม่มีใช้งานโปรแกรมอื่น ๆ ในเครื่องลูกข่าย เนื่องจากจะทำให้รบกวนการทำงานของโปรแกรมนี้ เพราะการทำงานของโปรแกรมนี้นั้นจะมีการตรวจสอบการทำงานของ Active Windows อยู่ตลอดเวลา
4. โปรแกรมไม่สามารถสั่งการให้เครื่องลูกข่ายทำการปรับปรุงโปรแกรมต่อต้านไวรัสผ่านไฟล์ได้ เนื่องจากโปรแกรม AntiVir ไม่สนับสนุนการทำงานในรูปแบบนี้
5. การทำงานของโปรแกรมอยู่ในรูปแบบสั่งการทางเดียว คือ เมื่อสั่งการลงไปที่ Agent แล้ว จะไม่สามารถยกเลิกคำสั่งดังกล่าวได้ เช่น คำสั่งในการตรวจสอบไวรัส (Scan), Update โปรแกรม Anti-Virus
6. โปรแกรมอำนวยความสะดวกในการจัดการไวรัสสำหรับผู้ดูแลระบบนี้ สามารถรองรับเครื่องลูกข่ายเข้าสู่ระบบได้ไม่เกิน 255 เครื่อง

5.5 ข้อเปรียบเทียบและข้อแตกต่าง

1. โปรแกรมที่พัฒนาขึ้นมานี้มีหลักการทำงานคล้ายกับโปรแกรม Corporate Edition ในส่วนที่มีตัวควบคุมเป็นศูนย์กลางและลักษณะการปฏิบัติงาน แต่โปรแกรมที่พัฒนาขึ้นนี้มีข้อดีในหลายส่วน เช่น ค่าใช้จ่ายในการใช้งานน้อย เพราะเป็นโปรแกรมที่พัฒนาขึ้นมาเอง และแม้ว่าจะเลือกใช้โปรแกรมต่อต้านไวรัสที่ไม่ต้องเสียค่าลิขสิทธิ์แต่ประสิทธิภาพโดยรวมของโปรแกรมใกล้เคียงกับโปรแกรมที่มีขายอยู่ทั่วไป
2. โปรแกรมที่พัฒนาขึ้นมานี้จะมีหลักการสั่งงานคล้ายกับโปรแกรม PC Anywhere แตกต่างกันที่โปรแกรม PC Anywhere มีการควบคุมการทำงานของเครื่องลูกข่ายได้เต็มประสิทธิภาพ สามารถเข้าถึงข้อมูลของเครื่องลูกข่าย เสมือนว่านั่งอยู่หน้าเครื่องนั้น แต่โปรแกรมที่พัฒนาขึ้นเป็นโปรแกรมแฝงที่รอรับคำร้องขอจากตัวควบคุมและปฏิบัติตามคำร้องขอนั้น โดยไม่สามารถเข้าควบคุมการทำงานของเครื่องลูกข่ายได้มากนัก ทั้งนี้โปรแกรม PC Anywhere เข้าถึงและควบคุมเครื่องลูกข่ายได้ครั้งละหนึ่งเครื่องเท่านั้น ซึ่งเหมือนกระบวนการทำงานทีละเครื่องเพียงแต่ไม่ต้องเดินไปสั่งงาน ในขณะที่โปรแกรมที่พัฒนาขึ้น สามารถสั่งการให้เครื่องลูกข่ายปฏิบัติงานพร้อมกันได้ครั้งละหลาย ๆเครื่อง ทำให้ลดขั้นตอนการทำงานไปได้มาก

5.6 แนวทางในการพัฒนาโปรแกรมในอนาคต

1. พัฒนาโครงสร้างของโปรแกรมอำนวยความสะดวกในการจัดการไวรัสสำหรับผู้ดูแลระบบให้สามารถใช้งานร่วมกับโปรแกรมต่อต้านไวรัสตัวอื่น ๆ ได้

2. พัฒนาโปรแกรมให้สามารถควบคุมการสั่งงานได้ดีขึ้น เช่น สามารถยกเลิกคำสั่งระหว่างปฏิบัติการได้ สามารถแจ้งสถานะการทำงานได้ละเอียดมากขึ้น เป็นต้น

3. สามารถปรับแต่งค่าต่าง ๆ ของโปรแกรมได้ละเอียดมากขึ้น

4. พัฒนาโปรแกรมให้สามารถสั่งปรับปรุงโปรแกรมต่อต้านไวรัสผ่านไฟล์ได้ (เมื่อโปรแกรมต่อต้านไวรัสนั้นสนับสนุนความสามารถนี้)

เอกสารอ้างอิง

[1] กิตติ ภัคดีวัฒนกุล , จำลอง ครูอุตสาหะ. Visual Basic 6 ฉบับโปรแกรมเมอร์. กรุงเทพฯ: ไทยเจริญการพิมพ์. 2542.

[2] พ.อ. เจนวิทย์ เหลืองอร่าม , ปิยวิทย์ เหลืองอร่าม.การเขียนโปรแกรมสำหรับ Client/Server ด้วย Visual Basic 6 และ ASP, VBScript, Access, SQLServer. กรุงเทพฯ: บริษัท ธรรมสาร จำกัด

[3] บริษัท อัปเปอร์ แมเนจเมนต์ เอ็กซ์เซลเลนซ์ จำกัด.เทคนิคและการประยุกต์ Visual Basic 6.0 Win32 API.กรุงเทพฯ : บริษัท ฟิล สไตล์ จำกัด

[4] สัจจะ จรัสรุ่งระวีวร. คู่มือพัฒนาแอพพลิเคชั่นด้วย Visual Basic 6.0. กรุงเทพฯ : ด่านสุธาการพิมพ์

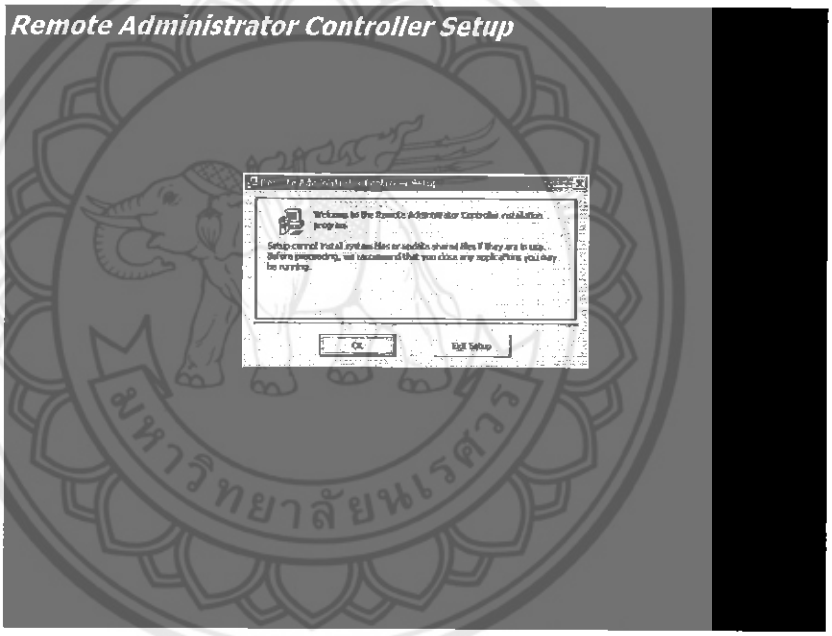
[5] สัจจะ จรัสรุ่งระวีวร. คู่มือการเขียนโปรแกรมและการใช้งาน Visual Basic 6.0. กรุงเทพฯ : ด่านสุธาการพิมพ์

ภาคผนวก

ในส่วนนี้จะเป็นการอธิบายถึงขั้นตอนการติดตั้งโปรแกรมอำนวยความสะดวกในการจัดการไวรัสสำหรับผู้ดูแลระบบทั้งในส่วนของตัวควบคุม (Controller) ส่วนที่ติดตั้งบนเครื่องลูกข่าย (Agent) รวมไปถึงคู่มือการใช้งาน รูปแบบคำสั่งที่ใช้ในการควบคุมวินโดว์เอพีไอ และคำสั่งที่ใช้ในการควบคุมโปรแกรมต่อต้านไวรัส ซึ่งมีรายละเอียดดังต่อไปนี้

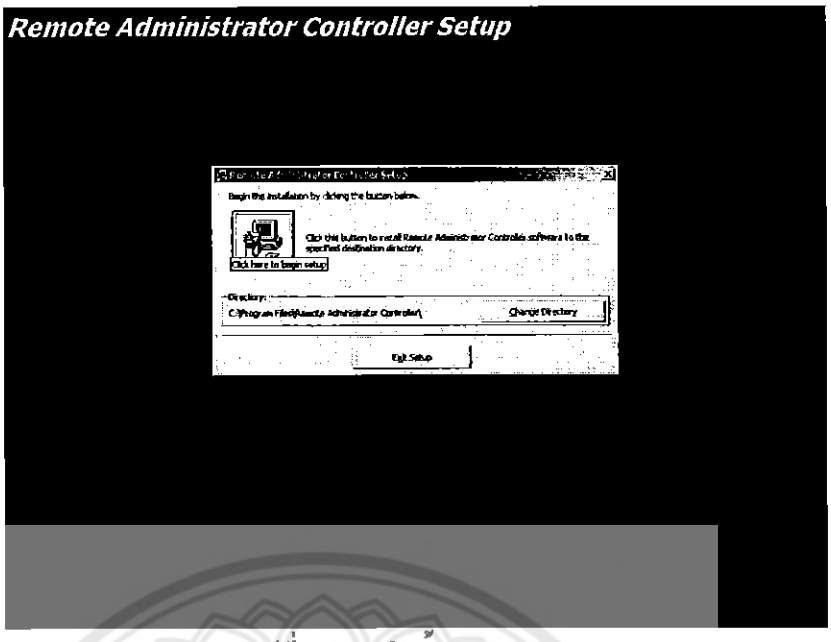
ขั้นตอนการติดตั้งโปรแกรมในส่วนของตัวควบคุม (Controller)

เมื่อทำการรันตัวเซตอัพ (Setup) ของโปรแกรมในส่วนของตัวควบคุมจะเป็นการเข้าสู่หน้าจอการติดตั้งโปรแกรมดังรูป 6.1



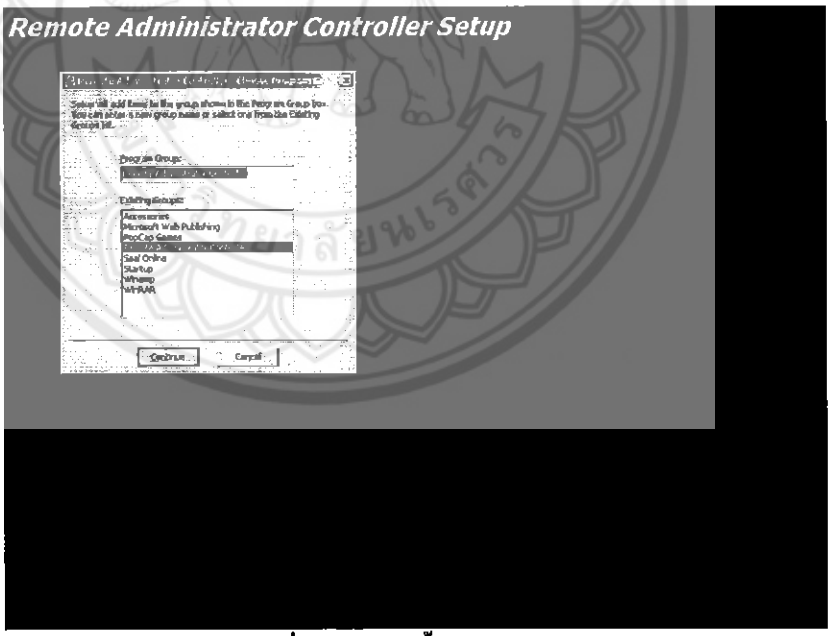
รูปที่ 6.1 การติดตั้งตัวควบคุม(1)

เมื่อต้องการดำเนินการติดตั้งต่อไปควรเลือกที่ OK จากนั้นจะเป็นการเข้าสู่หน้าจอเพื่อเลือกปลายทางที่ต้องการติดตั้งโปรแกรม ดังรูป 6.2



รูปที่ 6.2 การติดตั้งตัวควบคุม(2)

เมื่อเลือกปลายทางที่ต้องการติดตั้งโปรแกรมแล้ว ต่อไปเลือกไอคอนตามรูปที่ 6.2 ก็จะเข้าสู่หน้าจอดังรูป 6.3 ซึ่งจะเป็นการติดตั้งโปรแกรมลงบนสตาร์ทเมนู (Start Menu)



รูปที่ 6.3 การติดตั้งตัวควบคุม(3)

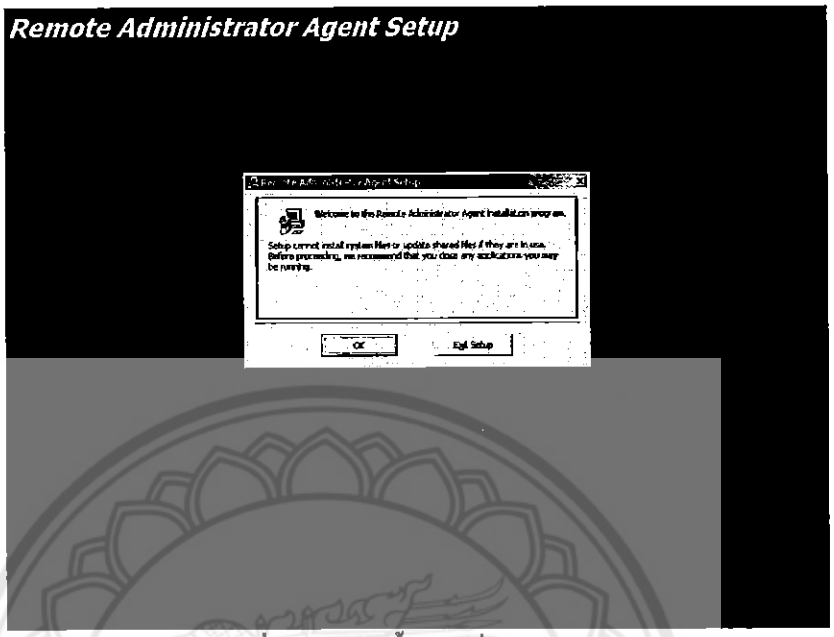
จากนั้นเลือกไอคอน "Continue" หลังจากนั้นจะเริ่มกระบวนการติดตั้ง รอจนกระบวนการต่าง ๆ เสร็จสิ้น การติดตั้งส่วนของตัวควบคุมก็เป็นอันเรียบร้อย และจะเข้าสู่หน้าจอแสดงผลการติดตั้ง ดังรูป 6.4

รูปที่ 6.4 การติดตั้งตัวควบคุม(4)

เมื่อทำการติดตั้งเรียบร้อยแล้ว การเรียกใช้โปรแกรมให้เข้าไปที่ Start-->Program--> Remote Administrator Controller ซึ่งจะแสดงหน้าจอของตัวควบคุมดังแสดงไว้ตอนต้น หลังจากนั้นควรทำการตั้งค่าที่ถูกต้องต่อการทำงาน

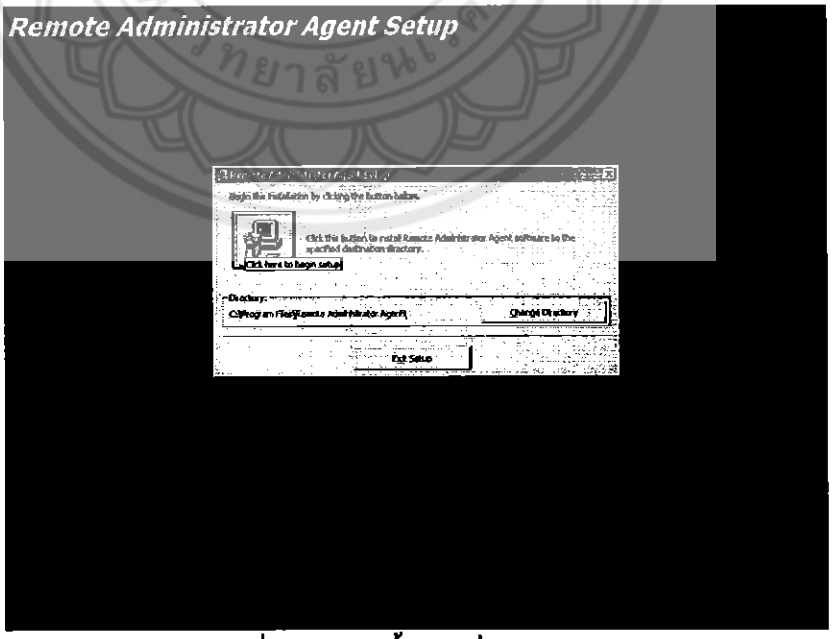
ขั้นตอนการติดตั้งในส่วนที่ติดตั้งบนเครื่องลูกข่าย (Agent)

เมื่อทำการรันตัวเซตอัพ (Setup) ของโปรแกรมจะเป็นการเข้าสู่หน้าต่างการติดตั้งดังรูป 6.5



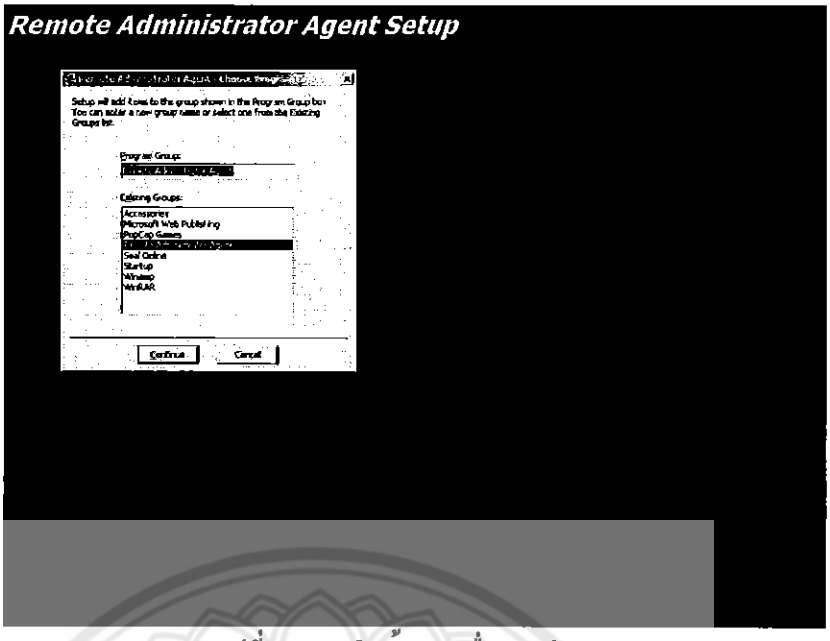
รูปที่ 6.5 การติดตั้งบนเครื่องลูกข่าย(1)

เมื่อต้องการดำเนินการติดตั้งต่อไปควรเลือกที่ OK จากนั้นจะเป็นการเข้าสู่หน้าจอเพื่อเลือกปลายทางที่ต้องการติดตั้งโปรแกรม ดังรูป 6.6



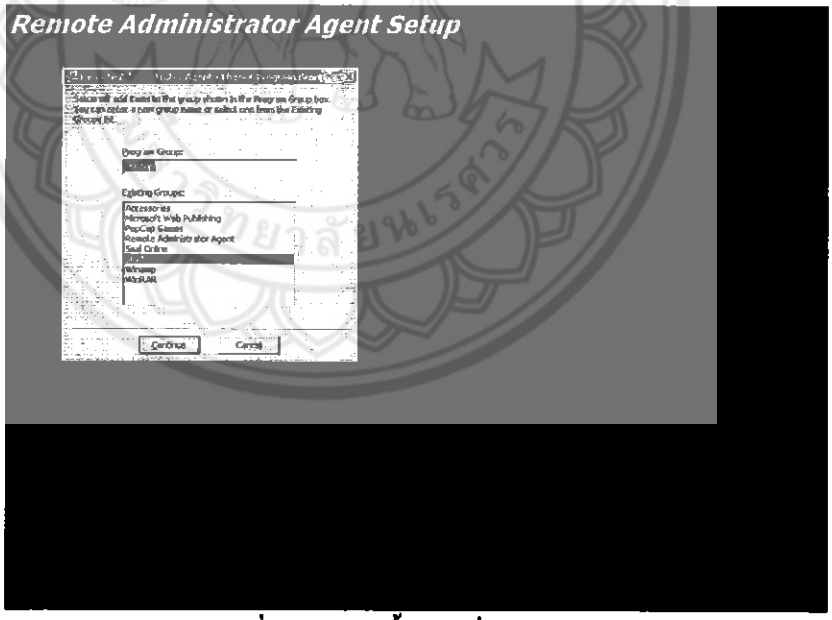
รูปที่ 6.6 การติดตั้งบนเครื่องลูกข่าย(2)

เมื่อเลือกปลายทางที่ต้องการติดตั้งโปรแกรมแล้ว ต่อไปเลือกไอคอนตามรูปที่ 6.6 ก็จะเข้าสู่หน้าจอดังรูป 6.7 ซึ่งจะเป็นการติดตั้งโปรแกรมลงบนสตาร์ทเมนู (Start Menu)



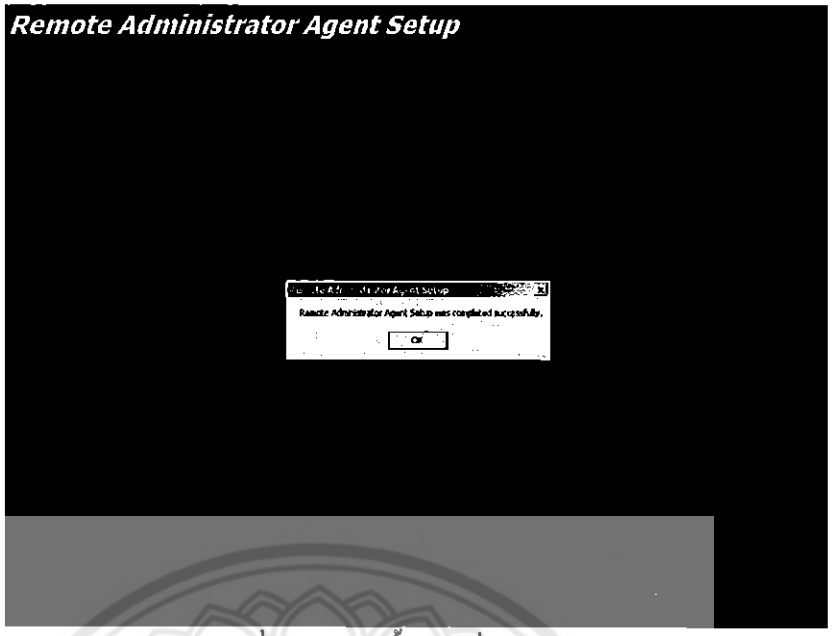
รูปที่ 6.7 การติดตั้งบนเครื่องลูกข่าย(3)

เลือก "Continue"แล้วจะมีหน้าจอคล้ายกับรูปที่ 6.7 แต่ในกระบวนการนี้โปรแกรมถูกติดตั้งลงบน Startup เพื่อให้โปรแกรมบนเครื่องลูกข่ายทำงานทันทีเมื่อเครื่องทำงาน ดังรูป 6.8



รูปที่ 6.8 การติดตั้งบนเครื่องลูกข่าย(4)

หลังจากเลือก "Continue" อีกครั้งแล้วจะเริ่มกระบวนการติดตั้ง รอจนกระบวนการต่าง ๆเสร็จสิ้นการติดตั้งส่วนโปรแกรมบนเครื่องลูกข่ายก็เป็นอันเรียบร้อย และจะเข้าสู่หน้าจอแสดงผลการติดตั้ง ดังรูป 6.9



รูปที่ 6.9 การติดตั้งบนเครื่องลูกข่าย(5)

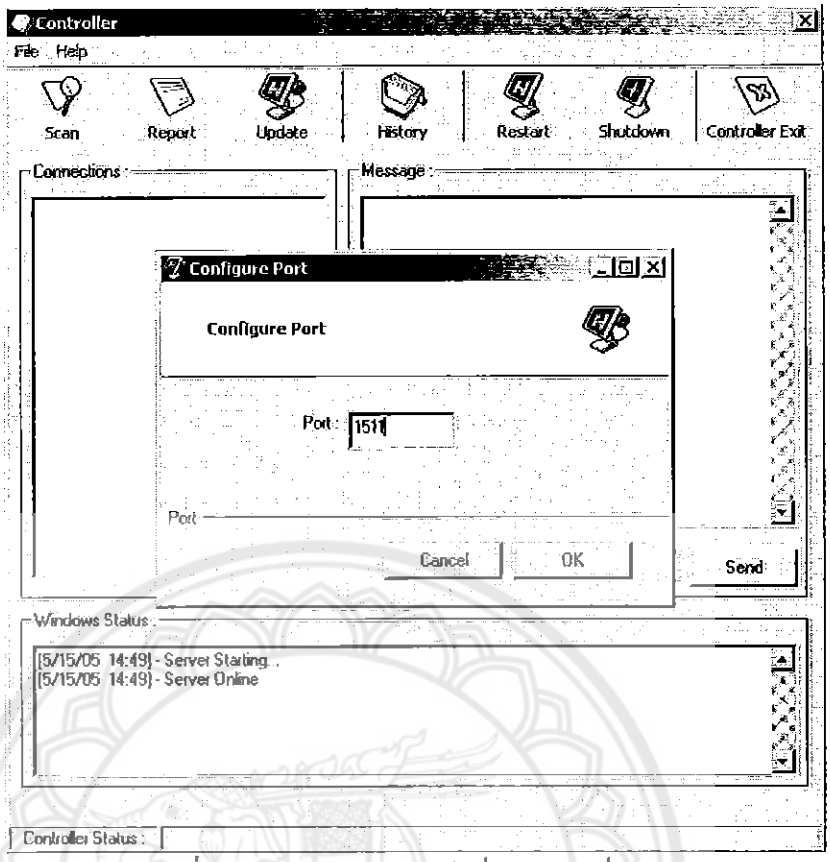
เมื่อทำการติดตั้งโปรแกรมบนเครื่องลูกข่ายแล้ว โปรแกรมจะทำงานอัตโนมัติหลังจากที่เครื่องลูกข่ายนั้นทำงาน หากโปรแกรมถูกปิด หรือไม่ทำงาน ให้เลือกไปที่ Start-->Program--> Remote Administrator Agent ดังนี้แล้วโปรแกรมจะทำงานอีกครั้ง และทำการตั้งค่าที่ถูกต้องต่อการเชื่อมต่อเพื่อให้โปรแกรมรอรับคำร้องขอจากตัวควบคุมต่อไป

คู่มือการใช้งาน (User Manual)

ความต้องการของระบบ

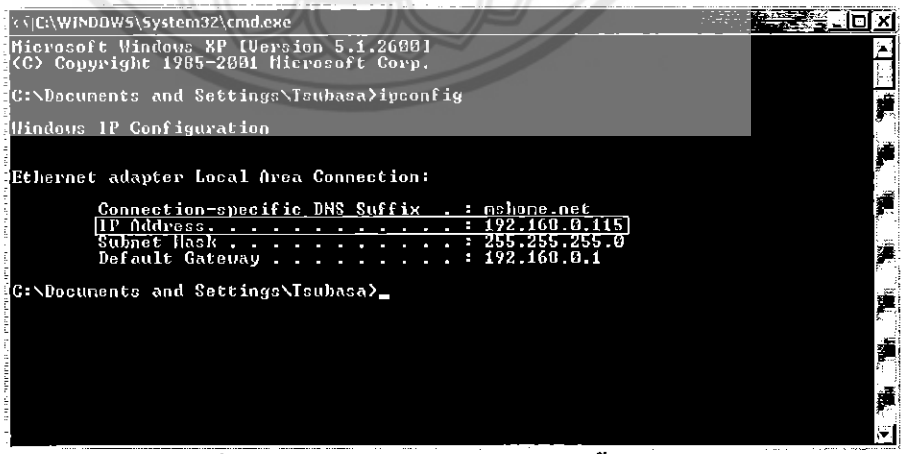
- ระบบปฏิบัติการ Windows 98/ME/2000/XP
- เครื่องลูกข่ายติดตั้งโปรแกรมต่อต้านไวรัส AnitVir
- ในระบบเครือข่ายติดตั้งตัวควบคุมและโปรแกรมบนเครื่องลูกข่ายแล้ว

โปรแกรมควรได้รับการตั้งค่าที่ถูกต้องสำหรับการเชื่อมต่อเสียก่อน เริ่มที่ส่วนของตัวควบคุม การตั้งค่าจะมีส่วนเดียว คือ การกำหนดค่าพอร์ท (Port) ที่ใช้เชื่อมต่อกันเสียก่อน โดยเข้าไปในส่วนของ File-->Config Port และทำการกำหนดค่าพอร์ท ดังรูป 6.10



รูปที่ 6.10 การกำหนดค่าพอร์ทที่ใช้ในการเชื่อมต่อ

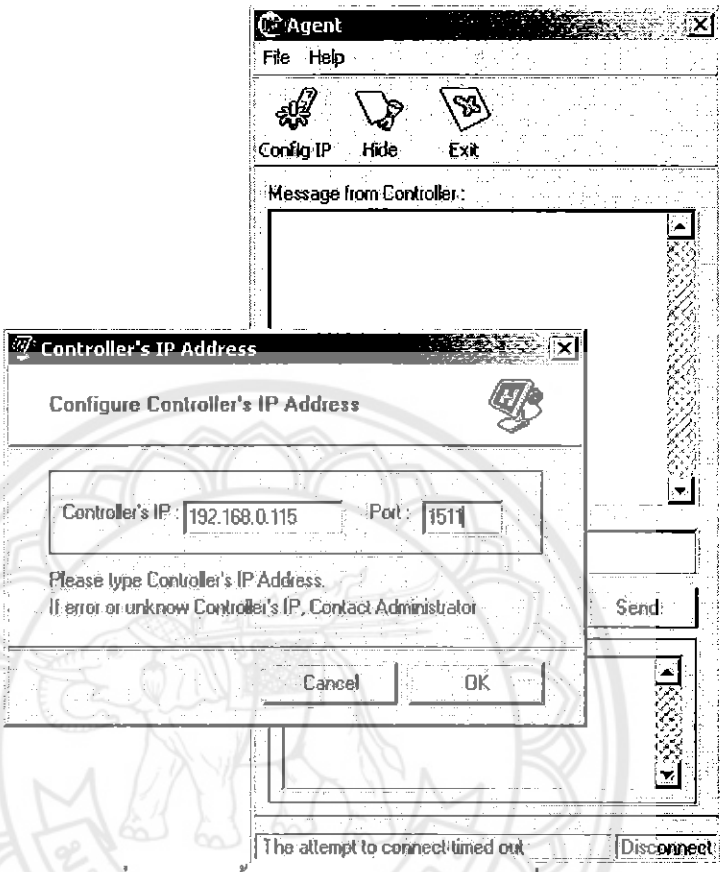
หลังจากนั้นทำการหาค่าไอพี แอดเดรส (IP Address) ของเครื่องที่ทำการติดตั้งตัวควบคุมโดยเข้าไปที่ Start-->Run พิมพ์คำว่า "cmd" หรือ "Command" จะเข้าสู่หน้าต่างของระบบดอส (Dos Mode) หลังจากนั้นพิมพ์คำว่า "ipconfig" และ Enter จะมีหน้าต่างค่าไอพี เพื่อทำการหาค่าไอพีของเครื่องตัวควบคุม ดังรูป 6.11



รูปที่ 6.11 การหาค่าไอพีของเครื่องที่ติดตั้งตัวควบคุม

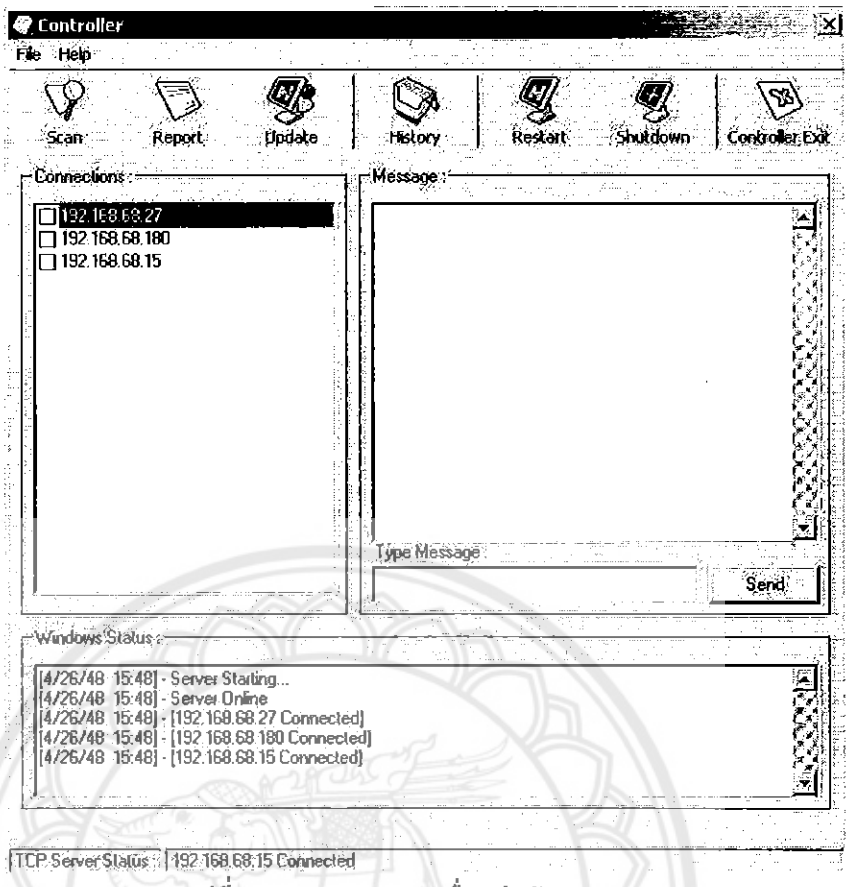
หลังจากนั้นนำค่าไอพีและพอร์ทดังกล่าว ไประบุให้กับโปรแกรมที่ติดตั้งบนเครื่องลูกข่าย โดยเริ่มจากการเข้าไปในส่วนของโปรแกรมของเครื่องลูกข่าย เข้าไปในส่วนของ "Config IP" จะแสดง

หน้าต่างสำหรับการตั้งค่าไอพีและพอร์ท ที่ใช้ในการเชื่อมต่อและนำค่าไอพีและพอร์ทที่ทราบจากขั้นตอนก่อนหน้านี้ระบุลงไป เลือก "OK" จะเป็นการตั้งค่าสำหรับเครื่องลูกข่าย ดังรูป 6.12



รูปที่ 6.12 การตั้งค่าสำหรับโปรแกรมบนเครื่องลูกข่าย

เมื่อตั้งค่าที่ถูกต้องให้กับระบบแล้ว การทำงานของตัวควบคุมและโปรแกรมบนเครื่องลูกข่ายจะแสดงการเชื่อมต่อกันดังรูป 6.13



รูปที่ 6.13 ภาพแสดงการเชื่อมต่อกันของระบบ

เมื่อเชื่อมต่อกันได้แล้ว ตัวควบคุมสามารถสั่งให้เครื่องลูกข่ายปฏิบัติตามคำร้องขอได้ เช่น คำสั่งในการตรวจสอบ(Scan) การเรียกดูผลการตรวจสอบ (Report) การปรับปรุงโปรแกรมต่อต้านไวรัส (Update Anti-Virus) การดูผลการตรวจสอบย้อนหลัง (History) รวมไปถึงการควบคุมเครื่องลูกข่ายให้กระทำการรีสตาร์ทหรือชัตดาวน์

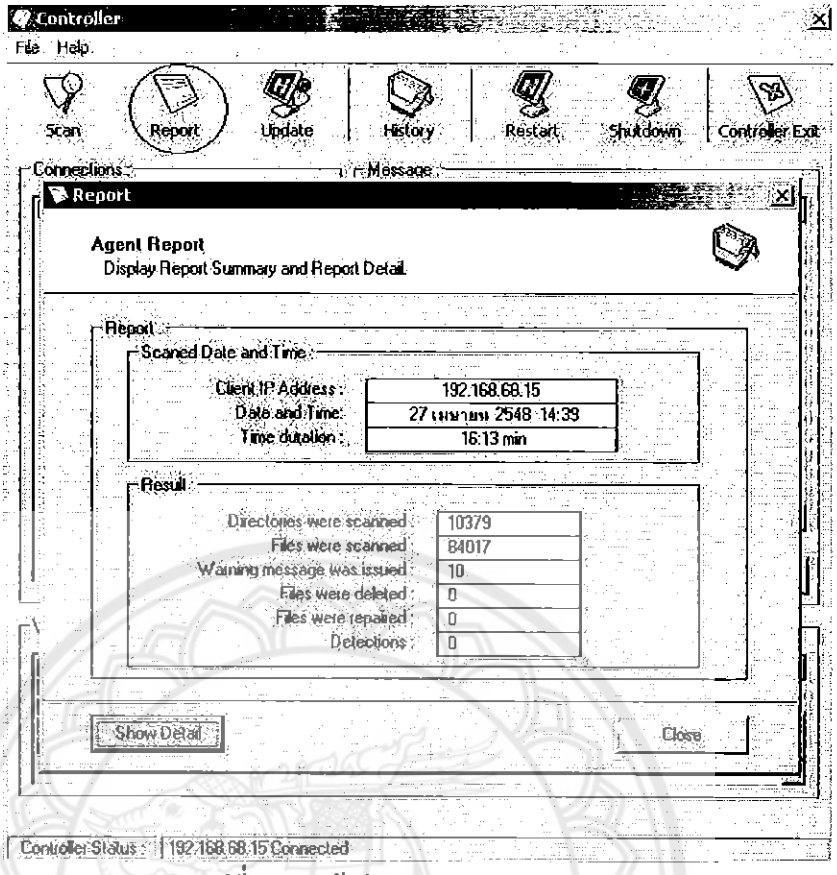
คำร้องขอในการตรวจสอบ (Scan)

เมื่อเครื่องลูกข่ายทำการเชื่อมต่อและแสดงขึ้นในหน้าต่างของตัวควบคุมแล้ว หากต้องการส่งคำร้องขอในการตรวจสอบ ให้ทำการเลือกเครื่องลูกข่ายที่แสดงอยู่บนหน้าต่างนั้นทำเครื่องหมายถูกที่กล่องทำเครื่องหมาย (Check Box) ของเครื่องลูกข่ายที่ต้องการ โดยสามารถสั่งการได้ครั้งละหลาย ๆเครื่องพร้อมกัน เมื่อเลือกเครื่องลูกข่ายเรียบร้อยแล้ว เลือกไอคอน “Scan” เพื่อส่งคำร้องขอให้กับเครื่องลูกข่าย เครื่องลูกข่ายจะทำการตรวจสอบเครื่องตัวเอง หลังจากทำการตรวจสอบเสร็จสิ้นแล้ว เครื่องลูกข่ายจะรายงานกลับมาเป็นรายงานคร่าว ๆ เช่นวันเวลาที่ทำการตรวจสอบ ผลการตรวจสอบพบหรือไม่ ดังรูป 6.14

รูปที่ 6.14 ขั้นตอนการส่งคำร้องขอในการตรวจสอบ

การเรียกดูผลการตรวจสอบ (Report)

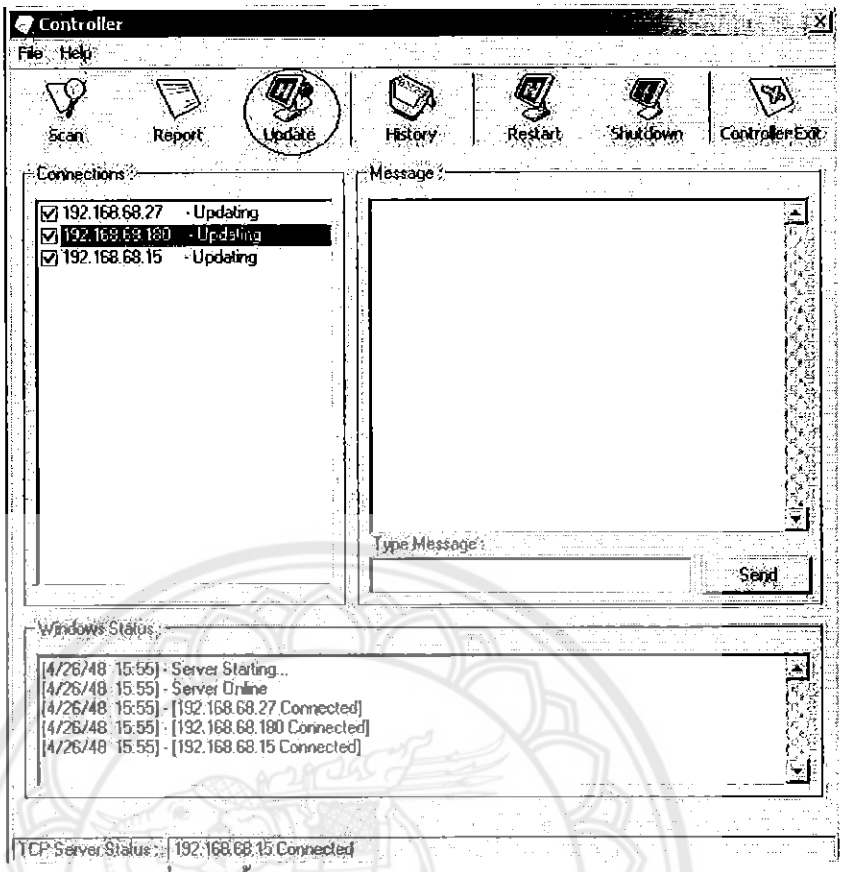
เมื่อเครื่องลูกข่ายทำการเชื่อมต่อและแสดงขึ้นในหน้าต่างของตัวควบคุมแล้ว หากต้องการส่งคำร้องขอในการเรียกดูผลการตรวจสอบ ให้ทำการเลือกเครื่องลูกข่ายที่แสดงอยู่บนหน้าต่างนั้นทำเครื่องหมายถูกที่กล่องทำเครื่องหมาย ของเครื่องลูกข่ายที่ต้องการ โดยสามารถสั่งการได้ครั้งละหนึ่งเครื่อง โดยเนื้อหาในรายงานจะแสดงผลของการตรวจสอบโดยละเอียด เช่น วันเวลาการตรวจสอบ การใช้เวลาในการตรวจสอบ จำนวนไฟล์ที่ทำการตรวจสอบ ไวรัสคอมพิวเตอร์ที่พบเป็นต้น โดยรายงานจะมีเนื้อหาที่สมบูรณ์ได้นั้น เครื่องลูกข่ายควรผ่านกระบวนการตรวจสอบมาก่อนแล้ว การแสดงผลของการตรวจสอบแสดงดังรูป 6.15



รูปที่ 6.15 หน้าต่างแสดงผลการตรวจสอบ

การปรับปรุงโปรแกรมต่อต้านไวรัส (Update Anti-Virus)

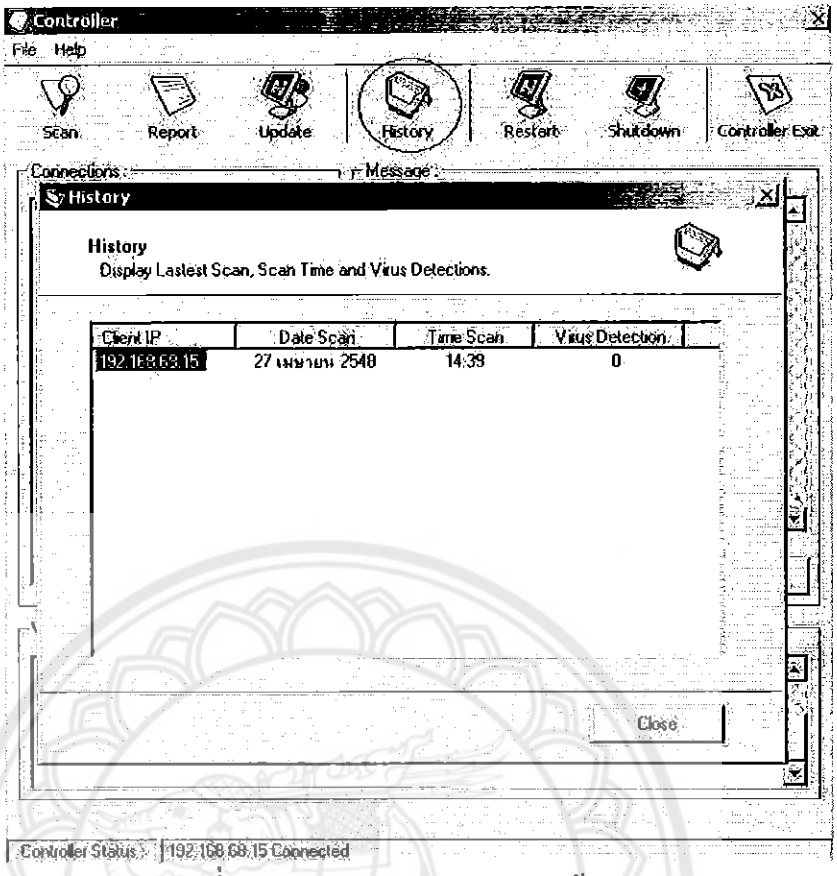
มีกระบวนการคล้ายกับการส่งคำร้องขอในการตรวจสอบ สามารถสั่งให้เครื่องลูกข่ายทำการปรับปรุงโปรแกรมต่อต้านไวรัสได้ครั้งละหลายๆ เครื่อง โดยโปรแกรมต่อต้านไวรัสจะทำการติดต่อไปยังโฮส (Host) ซึ่งมีไฟล์ที่จำเป็นต่อการปรับปรุงโปรแกรมต่อต้านไวรัส และทำการดาวน์โหลดไฟล์ดังกล่าวเพื่อทำการปรับปรุง รอจนกระบวนการดังกล่าวเสร็จสิ้น จะมีการรายงานว่าได้ทำการปรับปรุงเรียบร้อยแล้ว การตั้งค่าของโปรแกรมต่อต้านไวรัสที่ถูกต้องควรศึกษาเนื้อหาในบทที่ 4 หัวข้อที่ 4.3 ปัญหาที่พบขณะปฏิบัติการทดสอบและวิธีแก้ไข



รูปที่ 6.16 ขั้นตอนในการปรับปรุงโปรแกรมต่อต้านไวรัส

การดูผลการตรวจสอบย้อนหลัง (History)

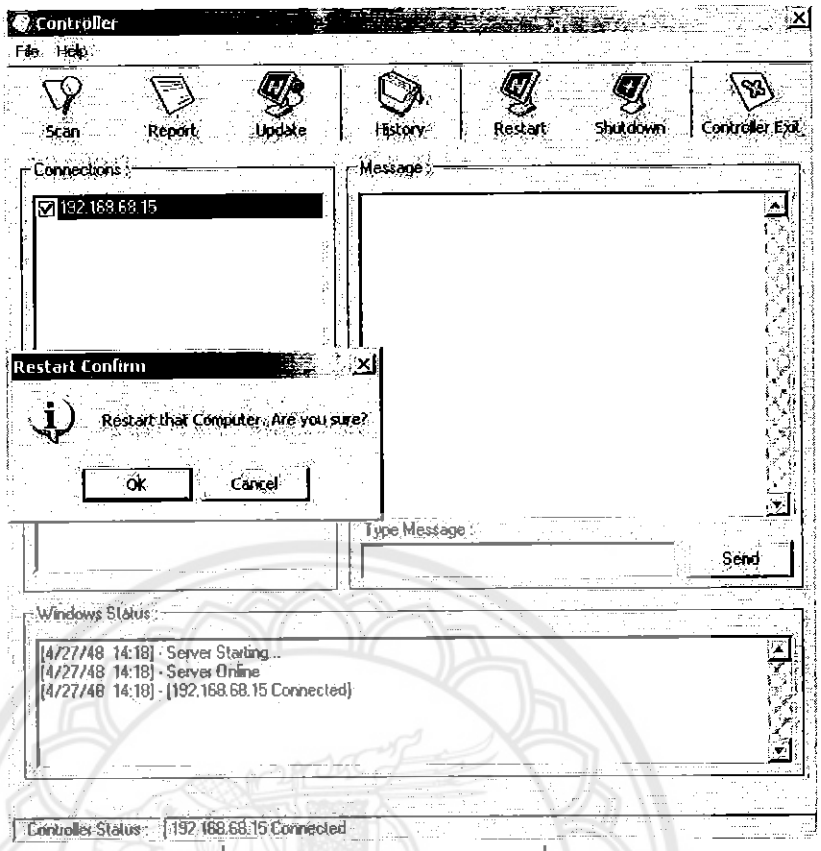
เป็นการเรียกดูผลการตรวจสอบล่าสุดและผลการตรวจสอบว่าพบไวรัสกี่ตัว โดยไม่ต้องทำการเลือกเครื่องลูกข่าย เมื่อเครื่องลูกข่ายทำการเชื่อมต่อแล้ว จะสามารถดูผลการตรวจสอบย้อนหลังได้ทันที โดยเลือกไอคอน "History" ของตัวควบคุมก็เข้าสู่หน้าต่างผลการตรวจสอบย้อนหลัง



รูปที่ 6.17 ภาพแสดงผลการตรวจสอบย้อนหลัง

การควบคุมเครื่องลูกข่ายให้กระทำการรีสตาร์ทหรือชัตดาวน์

เมื่อผู้ดูแลระบบเห็นว่าเครื่องลูกข่ายนั้นได้เสร็จสิ้นกระบวนการต่างๆแล้วและเห็นควรว่าสามารถทำการปิดระบบบนเครื่องลูกข่ายได้แล้ว ก็จะทำการเลือกเครื่องลูกข่ายดังกล่าวในที่นี้อาจเลือกเครื่องลูกข่ายได้หลายเครื่อง เมื่อทำการเลือกเรียบร้อยแล้ว เลือกไอคอน "Restart" หรือ "Shutdown" หากกดเลือกแล้วจะมีกล่องข้อความแสดงเพื่อยืนยันการปฏิบัติงาน เมื่อทำการยืนยันแล้วอีก 10 วินาทีเครื่องลูกข่ายจะปฏิบัติตามคำร้องขอนั้น และตัดการเชื่อมต่อออกไป



รูปที่ 6.18 ภาพแสดงการควบคุมเครื่องลูกข่าย

***หม**ายเหตุ การตั้งค่าและข้อกำหนดต่าง ๆควรศึกษาจากเนื้อหาข้างต้นเสียก่อน

คำสั่งที่ใช้ในโปรแกรม (Command for Operation in Program)

ตารางที่ 6.1 ตารางแสดงคำสั่งที่ใช้โปรแกรม

คำร้องขอ	รูปแบบคำสั่งในโปรแกรม	หน้าที่ของคำสั่ง
Report	"FileLog " & strSize	Agent ทำการอ่านรายงานและส่งรายงานคืนให้ Controller
Restart	InitiateShutdownMachine GetMyMachineName, _True, True,_True, 10, _	Agent ใช้ฟังก์ชันของ Windows API ในการสั่งให้เริ่มการทำงานของเครื่องลูกข่ายอีกครั้ง
Shutdown	InitiateShutdownMachine GetMyMachineName, _True, False,_True, 10, _	Agent ใช้ฟังก์ชันของ Windows API ในการสั่งปิดเครื่องลูกข่าย
Scan	Call Shell ("C:\ProgramFiles\AVPersonal\avwin.exe /ah/ns/b" , vbMaximizedFocus)	Agent สั่งให้โปรแกรม AntiVir ทำการตรวจสอบเครื่องลูกข่าย
Update	Call Shell ("C:\Program Files\AVPersonal\inetupd.exe", vbNormalFocus)	Agent สั่งให้โปรแกรม AntiVir ทำการปรับปรุงตัวเอง
Chat (Message)	"SMS " + sText	การส่งข้อความระหว่าง Controller กับ Agent

คำสั่งสำหรับวินโดว์เอพีไอ (Command for Windows API)

ตารางที่ 6.2 ตารางแสดงคำสั่งสำหรับวินโดว์เอพีไอ

หน้าที่ของฟังก์ชัน	รูปแบบการประกาศฟังก์ชันในโปรแกรม
เพิ่ม Icon ไปที่ System Tray	Declare Function Shell_NotifyIcon Lib "shell32" Alias "Shell_NotifyIconA" (ByVal dwMessage As Long, pnid As NOTIFYICONDATA) As Boolean
ฟังก์ชันสำหรับเปลี่ยนขนาดของวินโดว์ที่กำลังทำงานอยู่ (Active Windows)	Declare Function SetWindowPos Lib "user32.dll" _ (ByVal hWnd As Long, _ ByVal hWndInsertAfter As Long, _ ByVal X As Long, _

	ByVal Y As Long, _ ByVal cx As Long, _ ByVal cy As Long, _ ByVal uFlags As Long) As Long
ฟังก์ชันสำหรับการรับค่าของวินโดว์ที่กำลังใช้งานอยู่	Declare Function GetForegroundWindow Lib "user32.dll" () As Long
ฟังชันสำหรับคัดลอกข้อความภายในวินโดว์ที่กำลังทำงานระบุไปเก็บไว้ที่บัพเฟอร์	Declare Function GetWindowText Lib "user32.dll" _ Alias "GetWindowTextA" _ (ByVal hWnd As Long, _ ByVal lpString As String, _ ByVal nMaxCount As Long) As Long
ฟังชันสำหรับเขียนข้อความลงในเมสเสจคิวของเทรดที่เป็นเจ้าของวินโดว์ที่กำลังทำงานอยู่	Declare Function PostMessage Lib "user32.dll" _ Alias "PostMessageA" _ (ByVal hWnd As Long, _ ByVal Msg As Long, _ ByVal wParam As Long, _ lParam As Any) As Long

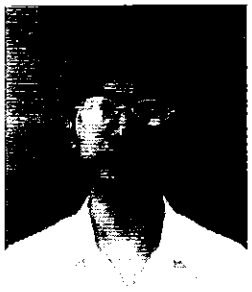
คำสั่งสำหรับโปรแกรมต่อต้านไวรัส AntiVir (Command for AntiVir Program)

เมื่อต้องการใช้คำสั่งผ่านระบบปฏิบัติการดอส เข้าไปในของโปรแกรม AntiVir เช่น "C:\Program Files\AVPersonal\avwin.exe /ah/ns/b" คำขีดเส้นใต้เป็นค่าของพารามิเตอร์ โดยสามารถเลือกปรับให้เหมาะสมกับการทำงาน ดังตารางที่ 6.3

ตารางที่ 6.3 ตารางแสดงคำสั่งโปรแกรมต่อต้านไวรัส

คำสั่งที่ใช้	หน้าที่ของคำสั่ง
...avwin.exe /ah...	เป็นคำสั่งที่ให้ทำการตรวจสอบทั้งฮาร์ดดิสก์ของเครื่องนั้น
...avwin.exe /an...	เป็นคำสั่งใช้ทำการตรวจสอบโฟลเดอร์และฮาร์ดดิสก์ที่แชร์อยู่ในเครือข่าย
...avwin.exe /ah/ns/b	เป็นการสั่งให้โปรแกรมต่อต้านไวรัสทำงานอัตโนมัติหลังจากสั่งงานแล้ว

ประวัติผู้เขียนโครงงาน



ชื่อ นายจักเรศ จับแสงจันทร์

ภูมิลำเนา 6/2 หมู่ 4 ตำบลหนองกุลา อำเภอบางระกำ จังหวัดพิษณุโลก

ประวัติการศึกษา

- จบระดับมัธยมศึกษาตอนปลายจากโรงเรียนจุฬาภรณ์ราชวิทยาลัย
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 5 สาขาวิชาวิศวกรรมศาสตร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : kudgee@hotmail.com

ชื่อ นายเทพดล เกตุสุวรรณ

ภูมิลำเนา 71 หมู่ 7 ตำบลพญาแมน อำเภอพิชัย จังหวัดอุตรดิตถ์

ประวัติการศึกษา

- จบระดับมัธยมศึกษาตอนปลายจากโรงเรียนพิชัย
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 5 สาขาวิชาวิศวกรรมศาสตร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : ozamunaki@hotmail.com