



ระบบควบคุมระยะไกลสำหรับคอมพิวเตอร์ที่ใช้ Windows XP

A remote administrator for Windows-based system.



นายจิรวัดน์ พร้อมมิตร รหัส 46380188
นางสาวปัญชลีมา อรุณมาศ รหัส 46380190

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ..... 10/5 ค.ย. 2551
เลขทะเบียน..... 05100044
เลขเรียกหนังสือ.....
มหาวิทยาลัยนเรศวร

15093429. e.2

ม.ค.

95121

2550

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2550



ใบรับรองโครงการงานวิศวกรรม

หัวข้อโครงการ	ระบบควบคุมระยะไกลสำหรับคอมพิวเตอร์ที่ใช้ Windows XP		
ผู้ดำเนินโครงการ	นายจิรวัดน์	พร้อมมิตร	รหัส 46380188
	นางสาวปัญชลีมา	อรุณมาศ	รหัส 46380190
อาจารย์ที่ปรึกษา	ดร.ไพศาล มุณีสว่าง		
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2550		

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยราชภัฏสุรินทร์ อนุมัติให้โครงการฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะกรรมการสอบโครงการ

.....ประธานกรรมการ
(ดร.ไพศาล มุณีสว่าง)

.....กรรมการ
(ดร.ชัยรัตน์ พินทอง)

.....กรรมการ
(ดร.สมพร เรืองสินชัยวานิช)

หัวข้อโครงการ	ระบบควบคุมระยะไกลสำหรับคอมพิวเตอร์ที่ใช้ Windows XP
ผู้ดำเนินโครงการ	นายจิรวุฒิ พร้อมมิตร รหัส 46380188 นางสาวปัญชลิมา อรุณมาศ รหัส 46380190
อาจารย์ที่ปรึกษา	อาจารย์ไพศาล มุณีสว่าง
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2550

บทคัดย่อ

โครงการนี้เป็นการศึกษาเพื่ออำนวยความสะดวกให้กับผู้ดูแลระบบ เครื่องข่ายคอมพิวเตอร์ ในการจัดการปัญหาเกี่ยวกับคอมพิวเตอร์ในเครือข่ายที่มีคอมพิวเตอร์หลายเครื่อง เพื่อให้ระบบเครือข่ายทำงานได้อย่างมีประสิทธิภาพ โดยการปฏิบัติการทั้งหมด ผู้ดูแลระบบสามารถสั่งงานได้จากเครื่องของผู้ดูแลระบบเอง เช่น การส่งไฟล์ การสั่งเปิดโปรแกรม การสั่งปิดเครื่องลูกข่าย รีโมทอาร์ท หรือ ส่งข้อความโต้ตอบ เป็นต้น นอกจากนี้การรับส่งข้อความระหว่างกันนั้นจะมีฟังก์ชันการเข้ารหัสข้อมูล ซึ่งเป็นการเพิ่มความปลอดภัยและความรวดเร็วในการส่งข้อความหรือข้อมูลสำคัญให้กับผู้ดูแลระบบในการดูแลระบบเครือข่ายคอมพิวเตอร์

จากการทดสอบ โปรแกรมอำนวยความสะดวกให้กับผู้ดูแลระบบ ในการจัดการเครื่องลูกข่ายในระบบเครือข่ายคอมพิวเตอร์ พบว่าโปรแกรมสามารถทำงานได้ตามวัตถุประสงค์

Project title	A remote administrator for Windows-based system.		
Name	Mr. Chirawat Prommitt	ID. 46380188	
	Ms. Panchalima Aroonmas	ID. 46380190	
Project advisor	Mr. Paisarn Muncesawang		
Major	Computer Engineering.		
Department	Electrical and Computer Engineering.		
Academic year	2007		

.....

ABSTRACT

The purpose of this project is to facilitate a network system administrator to deal with computer 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 SEND FILE, OPEN PROGRAM, SHUT DOWN, RESTART and also to CHAT with other users or the client. Moreover, Function chat between administrator and client have a security by ENCRYPTION. This system reduces the time and complexity in the process of operation and security for administrator to manage the computer in the network system.

กิตติกรรมประกาศ

การทำโครงการวิศวกรรมคอมพิวเตอร์ในครั้งนี้สำเร็จลุล่วงไปได้ด้วยดี คณะผู้จัดทำขอกราบ
 ขอบพระคุณท่านอาจารย์ที่ปรึกษา ดร. ไพศาล มณีสว่าง ที่กรุณาให้คำปรึกษา แนะนำวิธีการในการ
 ทำงาน ตลอดจนการแนะนำแหล่งค้นคว้าหาข้อมูลพร้อมทั้งเสนอแนวทางการแก้ไขปัญหาที่เกิดขึ้น
 ตลอดระยะเวลาการทำงาน ขอบพระคุณท่านคณะกรรมการ ดร.ชัยรัตน์ พินทอง และ ดร.สมพร เรือง
 สิ้นชัยวานิช ที่ได้สละเวลาเพื่อทำการตรวจสอบการทำงานและชี้แนวทางในการแก้ไขปัญหาโครงการนี้
 และขอบพระคุณกลุ่มบุคคลตัวอย่างทุกท่านที่ได้สละเวลาในการตอบแบบสอบถาม ซึ่งทำให้ได้ข้อมูล
 ที่เป็นประโยชน์อย่างยิ่งในการทำโครงการนี้ รวมไปถึงให้ความช่วยเหลือในอีกหลาย ๆ ด้าน ทางผู้จัดทำ
 ขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ทั้งนี้ขอกราบขอบพระคุณอาจารย์ทุกท่านที่ได้อบรมสั่งสอนให้ความรู้ คำแนะนำและความ
 ช่วยเหลือเสมอมา รวมทั้งขอขอบพระคุณเพื่อน ๆ พี่ ๆ ทุกคนที่ได้ให้ความช่วยเหลืออย่างเต็มที่ไม่ว่า
 ด้านการจัดทำโปรแกรมหรือการจัดทำเอกสาร

ท้ายนี้ขอกราบขอบพระคุณ บิดา-มารดา ผู้ที่ทุ่มเททั้งชีวิตและจิตใจเพื่อให้เกิดการศึกษาที่ดีที่สุด
 และสิ่งต่าง ๆ ที่ดีที่สุดแก่ผู้จัดทำ และคอยดูความสำเร็จในแต่ละก้าวของผู้จัดทำนอกจากนี้ยังคอย
 ห่วงใยและให้กำลังใจเสมอมา จนสามารถทำงานครั้งนี้จนสำเร็จลุล่วงด้วยดี

นายจิรวัดน์ พร้อมมิตร
 นางสาวปัญชลีมา อรุณมาศ

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่ออังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญตาราง	ฉ
สารบัญรูป	ช
บทที่ 1 บทนำ	
1.1 ที่มาและความสำคัญของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 ขอบข่ายของโครงการ	2
1.4 ขั้นตอนของการดำเนินงาน	2
1.5 ผลที่คาดว่าจะได้รับ	3
1.6 งบประมาณของโครงการ	3
บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง	
2.1 ระบบควบคุมระยะไกล	4
2.2 Computer Network	6
2.3 โพรโทคอลของระบบเครือข่าย	10
2.4 IP Address	13
2.5 โครงสร้างของโปรโตคอล	14
2.6 ความรู้พื้นฐานเกี่ยวกับ Microsoft Visual Basic Version 6.0	16
2.7 ความรู้พื้นฐานเกี่ยวกับระบบรหัส (Cryptography)	21
2.8 RSA Algorithm	23

สารบัญ (ต่อ)

	หน้า
บทที่ 3 การศึกษาและพัฒนาโปรแกรม	
3.1 เครื่องมือที่ใช้ในการพัฒนาโปรแกรม.....	24
3.2 การเขียนโปรแกรมบน Client-Server	24
3.3 การออกแบบระบบ (Context Diagram)	26
3.4 แผนภาพกระแสข้อมูล (Data Flow Diagram).....	26
3.5 การออกแบบในส่วนของอินเทอร์เฟซที่ติดต่อกับผู้ใช้ของ โปรแกรม	31
3.5.1 ส่วนของการออกแบบ Controller.....	31
3.5.2 ส่วนของการออกแบบเครื่องลูกข่าย.....	31
บทที่ 4 การทดสอบและวิเคราะห์การทำงาน	
4.1 การทดลองระหว่างการพัฒนาโปรแกรม	32
4.2 การทดสอบส่วนควบคุม (Controller).....	33
4.3 การทดสอบส่วนของการรับคำสั่งและปฏิบัติงาน (Agent)	46
4.4 การทดสอบส่วนของการบันทึกประวัติการทำงานของโปรแกรม (Log File).....	48
บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ	
5.1 สรุปผลการดำเนิน โครงการ	50
5.2 ประโยชน์ที่ได้รับจากการทำโครงการ	50
5.3 ความสามารถของโปรแกรม	51
5.4 ข้อจำกัดของโปรแกรม.....	51
5.5 ข้อเปรียบเทียบและข้อแตกต่าง	51
5.3 แนวทางในการพัฒนาโปรแกรมในอนาคต.....	52
บรรณานุกรม.....	53
ประวัติผู้เขียนโครงการ	54

สารบัญตาราง

ตารางที่	หน้า
1.1 ขั้นตอนการดำเนินงาน.....	2
2.1 ตารางแสดงตัวอย่าง IP Address	13
2.2 ตารางแสดงลำดับชั้นของ Private IP Address	13
2.3 ตารางแสดงคุณสมบัติต่างๆ ของคอนโทรล Winsock	18
2.4 ตารางแสดงค่าต่างๆ สำหรับกำหนด Property ของ State.....	19
2.5 ตารางแสดงรายการของการเชื่อมต่อต่างๆ ที่สามารถใช้กับการควบคุม Winsock.....	19



สารบัญรูป

รูปที่	หน้า
2.1 แผนผังการควบคุมระยะไกล	5
2.2 เครือข่าย LAN แบบ Peer-to-Peer.....	8
2.3 เครือข่ายแบบ Client/Server มีเครื่องเซิร์ฟเวอร์เป็นศูนย์กลางเครือข่าย.....	9
2.4 TCP/IP และ OSI Model	15
2.5 แผนผังการ Symmetric encryption	21
2.6 แผนผังของ Asymmetric/Public Key.....	22
2.7 แผนผังของ Asymmetric/Public Key.....	22
3.1 Context Diagram.....	26
3.2 แสดงกระแสข้อมูลระดับ 1	27
3.3 กระแสของข้อมูลระดับ 2 ของโปรเซส Agent Remote.....	28
3.4 กระแสของข้อมูลระดับ 2 ของโปรเซส Controller Chat.....	29
3.5 กระแสของข้อมูลระดับ 2 ของโปรเซส API	29
3.6 กระแสของข้อมูลระดับ 2 ของโปรเซส Agent Chat.....	30
3.7 กระแสของข้อมูลระดับ 2 ของโปรเซส Open Program.....	30
3.8 กระแสของข้อมูลระดับ 2 ของโปรเซส Send File.....	30
3.9 ส่วนของการอินเตอร์เฟส Controller.....	31
3.10 ส่วนของการอินเตอร์เฟสเครื่องลูกข่าย (Agent)	31
4.1 การเชื่อมต่อจากเครื่องลูกข่าย	32
4.2 การเชื่อมต่อถึงตัวควบคุม.....	33
4.3 การทำงานของตัวควบคุม (Controller).....	33
4.4 แสดงการทำงานของคำสั่ง Open Program.....	34
4.5 แสดงการทำงานของคำสั่ง Open Program	35
4.6 แสดงการทำงานของคำสั่งเปิดโปรแกรม AutoCAD.....	35
4.7 รูปแสดงการทำงานของคำสั่ง Send File.....	36
4.8 รูปแสดงการ Browse File ในคำสั่ง Send File.....	37
4.9 รูปแสดงขั้นตอนในการส่งไฟล์ในคำสั่ง Send File	37
4.10 รูปแสดงไฟล์ที่ถูกส่งจากเครื่องผู้ดูแลระบบในเครื่องลูกข่าย.....	38
4.11 รูปแสดงฟังก์ชันการส่ง Message ของผู้ดูแลระบบ.....	39

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.12 รูปอธิบายฟังก์ชันการส่ง Message ของผู้ดูแลระบบ.....	39
4.13 รูปแสดงฟังก์ชันการส่ง Message ของผู้ดูแลระบบ.....	40
4.14 รูปแสดงการส่ง Message ของผู้ดูแลระบบในช่องแสดง Message	40
4.15 รูปแสดงฟังก์ชันการส่ง Message แบบ Msg. Box ของผู้ดูแลระบบไปยังเครื่องลูกข่าย.....	41
4.16 รูปแสดง Message ของเครื่องลูกข่ายติดต่อกับผู้ดูแลระบบ.....	41
4.17 รูปแสดงขั้นตอนการ Lock คอมพิวเตอร์เครื่องลูกข่าย.....	42
4.18 รูปแสดงหน้าจอของเครื่องลูกข่ายที่ถูกสั่ง Lock.....	43
4.19 รูปแสดงสถานะ IP ของเครื่องที่ถูก Lock.....	43
4.20 รูปแสดงการ Unlock เครื่องลูกข่าย.....	44
4.21 รูปแสดงการสถานะการ Lock และ Unlock เครื่องลูกข่าย	44
4.22 รูปแสดงการตั้งรีสตาร์ท.....	45
4.23 รูปแสดงการตั้ง Shutdown	45
4.24 การกำหนดค่า IP Address ที่ใช้ในการติดต่อตัวควบคุม	46
4.25 การเชื่อมต่อกับตัวควบคุม	46
4.26 การรอสัญญาณตอบรับจากเครื่องลูกข่าย สำหรับการรีสตาร์ท	47
4.27 การรอสัญญาณตอบรับจากเครื่องลูกข่าย สำหรับการปิดเครื่อง	47
4.28 การสื่อสารจากเครื่องลูกข่ายไปยังตัวควบคุม	48
4.29 รูปแสดงขั้นตอนการเปิด Log File เหตุการณ์ที่เกิดขึ้นในระบบ	49
4.30 รูปแสดง Log File ของระบบ	49

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงการ

เนื่องจากปัจจุบันเทคโนโลยีทางด้านเครือข่ายเข้ามามีบทบาทในชีวิตประจำวันมากขึ้น ทั้งในด้านการศึกษา การทำงานในสำนักงานหรือองค์กรต่างๆ จำนวนผู้ใช้และขนาดของเครือข่ายในแต่ละองค์กรมีการเพิ่มจำนวนและขยายตัวอย่างรวดเร็ว ส่งผลให้ระบบมีความซับซ้อนมากขึ้น

การบำรุงและดูแลรักษาระบบเครือข่ายให้เป็นปกติอยู่เสมอเป็นหน้าที่ของผู้ดูแลระบบ ซึ่งถ้าเครือข่ายมีขนาดใหญ่มาก ภาระหน้าที่ย่อมตกเป็นของผู้ดูแลระบบ ที่จะต้องเข้าไปดูและและแก้ปัญหาของคอมพิวเตอร์ในเครือข่ายที่ละเครื่องโดยตรง ส่งผลให้เกิดการสิ้นเปลืองเวลาและทรัพยากรเป็นอย่างมาก

เทคโนโลยีการควบคุมระบบคอมพิวเตอร์ระยะไกล (Remote Desktop) เป็นเทคโนโลยีหนึ่งที่สามารถนำมาประยุกต์ใช้ เพื่ออำนวยความสะดวกให้แก่ผู้ดูแลระบบ ในการตรวจสอบและวิเคราะห์ปัญหา อีกทั้งยังช่วยลดความสิ้นเปลืองในส่วนของเวลาและทรัพยากรได้

ทางคณะผู้นำเสนอโครงการ จึงมีความประสงค์จะพัฒนาโปรแกรม ที่สามารถอำนวยความสะดวกในการติดตั้งและควบคุมระบบคอมพิวเตอร์ระยะไกล (Remote Desktop) หลากๆ เครื่อง รวมถึงให้ความสะดวกและรวดเร็วแก่ผู้ดูแลระบบในการเข้าไปตรวจสอบและวิเคราะห์ปัญหาเบื้องต้น ตลอดจนสามารถแก้ปัญหาและบำรุงรักษาระบบเครือข่ายได้สำเร็จอย่างรวดเร็ว และเป็นแนวทางในการศึกษาแก่ผู้ที่สนใจในด้านนี้ต่อไป

1.2 วัตถุประสงค์ของโครงการ

1.2.1 พัฒนาโปรแกรมที่อำนวยความสะดวกและลดขั้นตอนการปฏิบัติงานในการตรวจสอบและการแก้ไขปัญหาในระบบเครือข่าย รวมถึงการควบคุมคอมพิวเตอร์ระยะไกล (Remote Desktop) ได้ หลากๆ เครื่อง ให้แก่ผู้ดูแลระบบได้ โดยผู้ดูแลระบบสามารถส่งคำสั่งจากเครื่องของผู้ดูแลระบบเอง

1.2.2 เพื่อให้มีความรู้ความเข้าใจเกี่ยวกับระบบเครือข่ายและการควบคุมระบบคอมพิวเตอร์ระยะไกล

1.3 ขอบข่ายของโครงการ

1.3.1 ศึกษาทฤษฎีเกี่ยวข้องกับสิ่งต่างๆ เหล่านี้ ได้แก่

1. ระบบเครือข่าย

2. การควบคุมคอมพิวเตอร์ระยะไกล

1.3.2 ออกแบบโปรแกรม Remote Administrator

1.3.3 พัฒนาโปรแกรม Remote Administrator

1.3.4 ทำการทดสอบการใช้งานของโปรแกรม

1.3.5 วิเคราะห์การทดลองและสรุปผลการทดลอง

1.4 กิจกรรมการดำเนินการ

กิจกรรม	ปี 2550		ปี 2551			
	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.
1.เขียนโครงการทำงานและการนำเสนอ	←→					
2.ออกแบบโปรแกรม remote administrator	←→					
3.พัฒนาโปรแกรม remote administrator	←→					
4.ทดสอบการใช้งานของโปรแกรม remote administrator			←→			
5.ตรวจสอบโปรแกรมปรับปรุงและแก้ไข ปัญหาที่เกิดขึ้น					←→	
6.วิเคราะห์การทดลองและสรุปผลการทดลอง					←→	
7.จัดทำรายงานการรวบรวมข้อมูลเพื่อเตรียมนำเสนอ					←→	
8.ส่งโครงการฉบับสมบูรณ์						←→

1.5 ผลที่คาดว่าจะได้รับ

1.5.1 โปรแกรมอำนวยความสะดวก ระบบควบคุมระยะไกลสำหรับระบบคอมพิวเตอร์ที่ใช้ Windows XP

1.5.2 มีความรู้ความเข้าใจเกี่ยวกับระบบเครือข่ายและการควบคุมระบบคอมพิวเตอร์ระยะไกล มีแนวทางในการพัฒนาโปรแกรมที่เกี่ยวข้องในระบบเครือข่าย

1.6 งบประมาณที่ใช้

1. ค่าอุปกรณ์และเอกสารต่างๆ 2,000 บาท



บทที่ 2

หลักการและทฤษฎีที่เกี่ยวข้อง

ในปัจจุบันการสื่อสารผ่านระบบเครือข่าย (Network) รวมทั้งระบบอินเทอร์เน็ต เป็นที่แพร่หลายมากในแง่ของการใช้งาน, ใช้บริการ, อำนวยความสะดวกในชีวิตประจำวันต่างๆ เนื่องจากระบบสื่อสารที่รวดเร็ว และสามารถเชื่อมโยงเข้ากับหลายๆระบบได้ทำให้สามารถนำมาประยุกต์ใช้งานผ่านระบบเครือข่ายได้หลากหลายรูปแบบ ซึ่งจะนำไปประยุกต์ใช้ในการควบคุมอุปกรณ์หรือโปรแกรมต่างๆ

เนื้อหาในบทนี้จะกล่าวถึงหลักการและทฤษฎีต่างๆที่สามารถนำมาประยุกต์ต่อการสร้างโปรแกรมที่อำนวยความสะดวก ด้านการลดขั้นตอนการทำงานให้กับผู้ดูแลระบบ และการควบคุมคอมพิวเตอร์ระยะไกลผ่านระบบเครือข่าย

2.1 ระบบควบคุมระยะไกล (Remote Control) [2]

การควบคุมระยะไกล หมายถึง การเข้าควบคุมคอมพิวเตอร์จากระยะไกล โดยเราจะต้องมีเครื่องคอมพิวเตอร์อย่างน้อย 2 เครื่องขึ้นไป ที่เชื่อมต่อกันอยู่

โปรแกรม Remote Administrator เป็นโปรแกรมควบคุมคอมพิวเตอร์ระยะไกลที่เชื่อมต่อกันโดยติดต่อผ่านทางโปรโตคอล TCP/IP ประกอบไปด้วย 2 ส่วนคือ

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

ใช้ในการปฏิบัติการของระบบ

ปกติแล้ว ระบบปฏิบัติการวินโดวส์ในเวอร์ชันก่อนๆ ที่ผ่านมามักจะสร้างปัญหาให้กับผู้ใช้งานเครือข่าย โดยเฉพาะในเรื่องการควบคุมคอมพิวเตอร์ระยะไกล ซึ่งทำได้ยาก ส่วนนี้จึงกลายเป็นช่องว่างสำหรับโปรแกรมยูทิลิตี้ของ Third Party ที่ได้เข้ามาอุดช่องว่างดังกล่าว ตัวอย่างเช่น โปรแกรม Symantec's PcAnywhere32 หรือ Virtual Network Computing (VNC) ซึ่งแน่นอนว่าการนำซอฟต์แวร์ยูทิลิตี้มาใช้นั้นย่อมเสียค่าใช้จ่ายราคาแพง โดยเฉพาะการใช้งานในระดับองค์กรที่มีเครือข่ายขนาดใหญ่

เมื่อมีระบบปฏิบัติการ Windows XP ขึ้นมา และได้เพิ่มเติมความสามารถของการควบคุมคอมพิวเตอร์ระยะไกล จึงกลับมาอุดช่องว่างของตัวเองที่เคยเกิดขึ้นกับวินโดวส์เวอร์ชันผ่านๆ มา

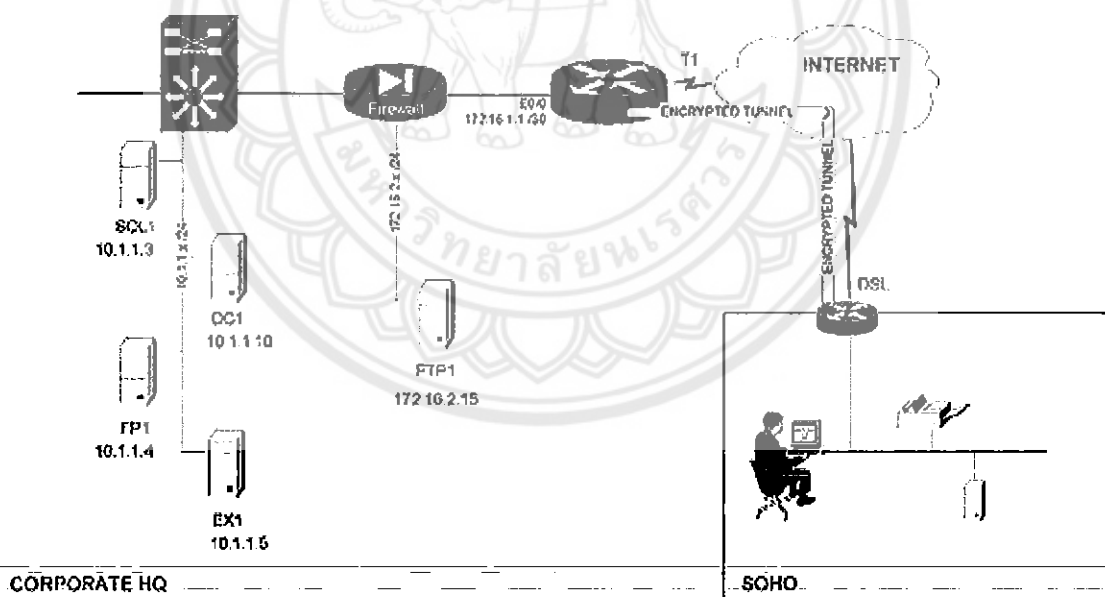
ในส่วนนี้จะพูดถึงการทำงานของระบบควบคุมระยะไกลของ Windows XP และ Virtual Network Computing (VNC)

2.1.1 Remote Desktop using Windows XP

เงื่อนไขในการเชื่อมต่อระบบ Remote Assistance นั้น ประการแรกจะต้องมีเครื่องพีซีที่ทำงานบนวินโดวส์ XP

ประการที่สอง เครื่องพีซีจะต้องมีการเชื่อมต่อกันเป็นเครือข่าย อย่างเช่นระบบแลนหรือว่าผ่านทางอินเทอร์เน็ต ถ้าเป็นการเชื่อมต่อผ่านทางอินเทอร์เน็ต ทาง Agent จะต้องแน่ใจว่าได้เซตค่า firewalls ให้สามารถผ่านข้อมูลที่เป็น RDP (TCP port 3389) เข้าไปสู่ระบบได้ แต่ถ้าทาง user ใช้วินโดวส์ XP ที่มีฟีเจอร์ Internet Connection Firewall (ICF) เอาไว้อยู่แล้ว การเซตค่าใน Firewalls ก็ไม่มีความจำเป็นอีกต่อไป เพราะวาระบบ ICF จะเข้าไปเปิดการเชื่อมต่อสำหรับ Remote Assistance โดยอัตโนมัติในทันทีที่ได้การร้องขอเข้ามา

สิ่งที่น่าสนใจเกี่ยวกับการบริการเครือข่ายพื้นฐานและการควบคุมระยะไกลเป็นต้นว่า สามารถควบคุมคอมพิวเตอร์อื่นๆ ในระยะไกล เสมือนไปนั่งอยู่หน้าจอคอมพิวเตอร์เครื่องนั้นเองในภาพประกอบข้างล่าง ผู้ควบคุมจะถูกมองเป็นเสมือนผู้บริหารระบบโดยอัตโนมัติเมื่อเกิดปัญหาที่เครื่องเป้าหมาย ผู้ควบคุมสามารถทำการแก้ไขหรือซ่อมแซมจากที่ไหนก็ได้ที่มีการเชื่อมต่อถึงกัน



รูปที่ 2.1 แผนผังการควบคุมระยะไกล

2.1.2 Remote Desktop using Windows XP

UltraVNC เป็นซอฟต์แวร์ที่ใช้งานได้ง่าย รวดเร็ว มีความสามารถในการแสดงจอภาพและควบคุมคอมพิวเตอร์เครื่องอื่น ๆ (ผ่านทางอินเทอร์เน็ต หรือเครือข่าย) บนจอ Monitor ของเราเอง เป็นโปรแกรมที่สามารถใช้เมาส์และคีย์บอร์ดของเราควบคุมเครื่องคอมพิวเตอร์อื่นๆได้ในระยะไกล นั่นคือสามารถทำงานบนเครื่องคอมพิวเตอร์ระยะไกล โดยที่ผู้ควบคุมนั่งทำงานอยู่ที่เครื่องของตัวเองราวกับว่าเราไปนั่งทำงานที่เครื่องเครื่องเป้าหมายที่ได้ลงโปรแกรมนี้ไว้ ผู้ควบคุมสามารถที่จะเข้าถึงเครื่องคอมพิวเตอร์ของลูกข่าย จากที่ใด ๆ ก็ได้ในโลก โดยที่เครื่องลูกข่ายจะต้องไม่ถอนการติดตั้งโปรแกรม ออกไปเสียก่อน UltraVNC ยังสามารถควบคุมคอมพิวเตอร์ระยะไกล โดยผ่านการเชื่อมต่อแบบ TCP/IP UltraVNC สามารถรันได้บนระบบปฏิบัติการ Windows 95, 98, Me, NT4, 2000, XP, 2003 UltraVNC มีความโดดเด่นมากกว่าการใช้ Remote Desktop ของ Windows XP ตรงที่สามารถเพิ่ม Plug-in ได้เช่น การเข้ารหัสข้อมูล ส่วนสนับสนุนวีดีโอ การโอนถ่ายข้อมูลที่รวดเร็ว และสามารถสนทนากันได้ระหว่างเครื่องเครือข่าย

2.2 Computer Network [7]

เครือข่ายคอมพิวเตอร์ หรือ Computer Network หมายถึง กลุ่มเครื่องคอมพิวเตอร์และอุปกรณ์คอมพิวเตอร์ชนิดต่างๆ ที่เชื่อมโยงเข้าด้วยกันอย่างเป็นระบบ มีระเบียบแบบแผนเพื่อให้กลุ่มคอมพิวเตอร์และอุปกรณ์เหล่านั้นสามารถติดต่อสื่อสารข้อมูลกันได้ หรือเพื่อให้เกิดการใช้ทรัพยากรร่วมกันอย่างมีประสิทธิภาพ สำหรับขนาดของเครือข่ายคอมพิวเตอร์อาจเริ่มต้นจากคอมพิวเตอร์ 2 เครื่องต่อเข้าด้วยกันจนไปถึงนับร้อยนับพันเครื่อง เครือข่ายคอมพิวเตอร์สามารถแบ่งออกเป็น 3 ประเภทหลักๆ ตามระยะทางการเชื่อมโยงเครือข่ายดังนี้

- เครือข่ายระดับท้องถิ่น (Local Area Network, Lan) เป็นการเชื่อมโยงคอมพิวเตอร์ในพื้นที่บริเวณเดียวกันหรือใกล้เคียงกันเข้าเป็นเครือข่าย เครือข่ายประเภทนี้มีระยะทำการใกล้และมักจะเชื่อมโยงกันด้วยความเร็วสูง เช่น ในห้องเดียวกัน ในชั้นอาคารเดียวกัน หรือภายในอาคารเดียวกัน เป็นต้น

- เครือข่ายระดับเมือง (Metropolitan Area Network, Man) ส่วนใหญ่จะเป็นการรวมกลุ่มเครือข่ายระดับท้องถิ่นหลายๆ เครือข่ายเข้าด้วยกัน ทำให้สามารถติดต่อสื่อสารข้อมูลข้ามเครือข่ายได้ และยังขยายขอบเขตพื้นที่การสื่อสารให้กว้างขึ้น ตัวอย่างเช่น เครือข่ายระดับ WAN ของมหาวิทยาลัย ซึ่งเชื่อมโยงเครือข่ายระดับ LAN ของแต่ละคณะหรือภาควิชาต่างๆ เข้ากับส่วนสนับสนุนการศึกษาอื่นๆ ซึ่งส่วนใหญ่อยู่คนละอาคาร โดยมีศูนย์กลางอยู่ที่ศูนย์เทคโนโลยีสารสนเทศของมหาวิทยาลัย เป็นต้น

- เครือข่ายบริเวณกว้าง (Wide Area Network, WAN) เป็นเครือข่ายคอมพิวเตอร์ที่รวมเครือข่ายระดับ LAN และ MAN เป็นเครือข่ายเดียวกัน สามารถครอบคลุมพื้นที่บริเวณกว้างอาจจะทั่วทั้งประเทศ

ข้ามประเทศ ระหว่างทวีป หรือทั่วโลกเลยก็ได้ ตัวอย่างของ WAN ที่เห็นได้ชัดเจนที่สุดก็คือ เครือข่ายอินเทอร์เน็ตนั่นเอง

2.2.1 ประโยชน์ของเครือข่ายคอมพิวเตอร์

เครือข่ายคอมพิวเตอร์เป็นระบบที่มีการใช้ทรัพยากรและใช้ข้อมูลร่วมกัน สามารถแบ่งประโยชน์ออกเป็น 5 ข้อคือ

1. การใช้โปรแกรม-ข้อมูลร่วมกัน (Shared-Applications)

สามารถจะทำให้ User หลายคนใช้โปรแกรม-ข้อมูลร่วมกันได้ และเป็นการประหยัดเวลาในการติดตั้งโปรแกรม โดยทำการติดตั้งโปรแกรมใช้งานไว้ที่ไฟล์เซิร์ฟเวอร์ และเก็บข้อมูลไว้ที่ไฟล์เซิร์ฟเวอร์เพียงตัวเดียว เครื่องเวิร์กสเตชัน ก็สามารถใช้งานโปรแกรมและข้อมูลจากเครื่องไฟล์เซิร์ฟเวอร์ได้

2. การใช้ฮาร์ดแวร์ร่วมกัน (Shared Hardware)

เช่น เครือข่ายคอมพิวเตอร์มีเครื่องเวิร์กสเตชันอยู่ 12 เครื่อง แต่ละเครื่องมีงานต้องพิมพ์บนเครื่องพิมพ์เลเซอร์ ถ้าไม่มีเครือข่ายคอมพิวเตอร์จะต้องใช้งบประมาณในการซื้อเครื่องพิมพ์อย่างน้อย 4-6 เครื่อง แต่สำหรับเครือข่ายคอมพิวเตอร์แล้ว สามารถใช้เครื่องพิมพ์เพียง 1-2 เครื่องโดยการทำเป็น Print Sever คอยรับงานพิมพ์จากเครื่องทั้ง 12 เครื่อง

3. การกระจายการประมวลผล (Distributed Processing)

เมื่อ User ของเครื่องเวิร์กสเตชันเครื่องหนึ่งต้องการใช้งานโปรแกรมที่อยู่บนเครื่องไฟล์เซิร์ฟเวอร์ เครื่องไฟล์เซิร์ฟเวอร์จะทำการ Copy โปรแกรมจากฮาร์ดดิสก์ของไฟล์เซิร์ฟเวอร์มาไว้ยังหน่วยความจำ RAM ของเวิร์กสเตชันเครื่องนั้น ต่อจากนี้จะเป็นหน้าที่ของ CPU ภายในเครื่องเวิร์กสเตชัน ทำการ Run โปรแกรมต่อไป

4. การติดต่อสื่อสารแบบรวดเร็ว (Rapid Communication)

ในเครือข่าย LAN นั้นความเร็วต่ำสุดสำหรับการส่งข้อมูลผ่านสายคู่ตีเกลียวจะประมาณ 1,000 Lbps หรือ 1 Mbps และจะส่งได้เร็วยิ่งขึ้นเมื่อใช้สายสื่อสารที่มีคุณภาพดีขึ้น ดังนั้นเครือข่ายคอมพิวเตอร์จะสามารถช่วยให้การสื่อสารข้อมูลภายในหน่วยงานรวดเร็วยิ่งขึ้น

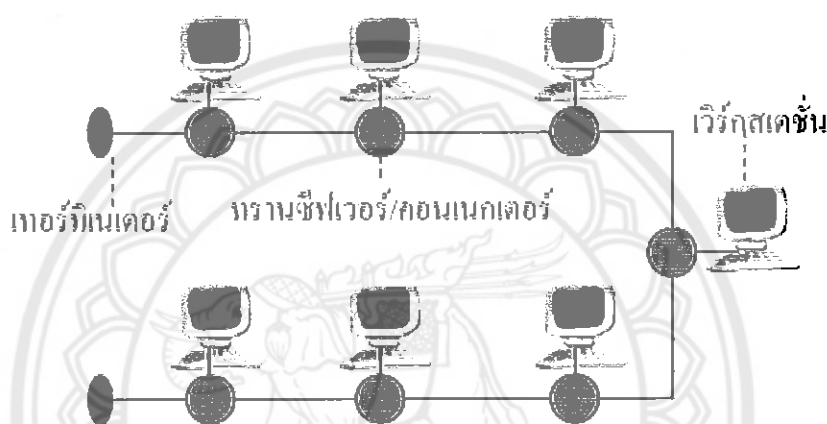
5. การติดต่อสื่อสารระหว่างเวิร์กสเตชัน (Interconnected Station)

เช่นการส่งข้อความ รายงาน หรือไฟล์ข้อมูล ถึงกันระหว่าง User สามารถจะช่วยลดขั้นตอนการส่งข่าวสาร ประหยัดเวลา และประหยัดกระดาษพิมพ์

2.2.2 เครือข่ายท้องถิ่น (Local Area Network, LAN)

วัตถุประสงค์ของการใช้ระบบเครือข่ายท้องถิ่นก็คือ การติดต่อร่วมกันของเทอร์มินัล เครื่องพิมพ์ ไมโครคอมพิวเตอร์ และอุปกรณ์อิเล็กทรอนิกส์ภายในสำนักงาน เพื่อสามารถแชร์การใช้ข้อมูล ซอฟต์แวร์ และฮาร์ดแวร์ เพื่อเป็นการใช้ “ทรัพยากร” ที่มีอยู่ให้คุ้มค่า เพื่อเพิ่มผลผลิตและประสิทธิภาพของงาน ที่สำคัญก็คือ เพื่อเป็นการลดค่าใช้จ่าย สามารถแบ่งประเภทของระบบ LAN ได้ 2 แบบ คือ

1. เครือข่ายแบบ Peer-to-Peer

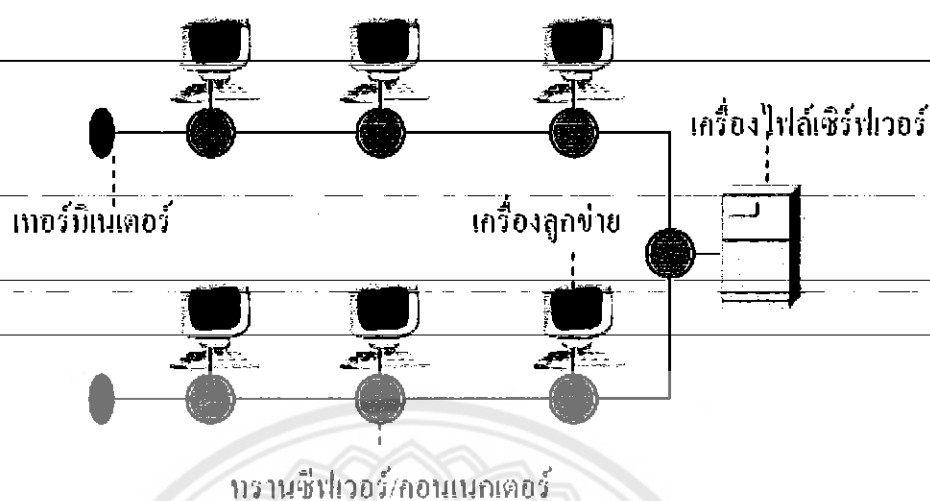


รูปที่ 2.2 เครือข่าย LAN แบบ Peer-to-Peer ทุกเครื่องสำคัญเท่าเทียมกัน

เป็นเครือข่าย LAN ขนาดเล็ก สามารถทำได้ง่ายโดยการติดตั้ง LAN Card บนเครื่องคอมพิวเตอร์แต่ละเครื่อง จากนั้นเชื่อมโยงคอมพิวเตอร์แต่ละเครื่องด้วยสายสัญญาณ เข้ากับ LAN CARD ต่อเนื่องกันไปทีละเครื่องจนครบวง

เครือข่าย Peer-to-Peer นี้จะไม่มีเครื่องคอมพิวเตอร์ศูนย์กลางทำหน้าที่ดูแลจัดการบริหารเครือข่าย ข้อมูลจะถูกสร้างและเก็บอยู่ในคอมพิวเตอร์แต่ละเครื่อง ผู้ใช้แต่ละคนสามารถดึงข้อมูลจากเครื่องอื่นมาใช้งานในเครื่องของตนได้โดยผ่านทางเครือข่าย แต่จะมีปัญหาเรื่องการรักษาความปลอดภัย เนื่องจาก User ทุกคนในเครือข่ายสามารถเข้าไปดูและทำการโอนถ่ายข้อมูลออกมาง่าย ๆ และในขณะที่มีการแลกเปลี่ยนข้อมูลหรือร่วมกันใช้ทรัพยากร สมรรถนะของเครื่องจะต่ำลง โปรแกรมที่มีความสามารถทาง Peer-to-Peer เช่น Windows for WorkGroup

2. เครือข่ายแบบ Client/Server



รูปที่ 2.3 เครือข่ายแบบ Client/Server มีเครื่องเซิร์ฟเวอร์เป็นศูนย์กลางเครือข่าย

เครือข่ายในรูปแบบนี้ จะมีคอมพิวเตอร์หลังเครื่องหนึ่งที่ทำหน้าที่เป็นศูนย์กลางของเครือข่าย เรียกว่า “เครื่องแม่ข่าย” หรือ “เซิร์ฟเวอร์ (Server)” ส่วนเครื่องของ User จะเรียกว่า “เครื่องลูกข่าย” หรือ “ไคลเอนต์ (Client)”

โครงสร้างพื้นฐานของ Client/Server ถูกแบ่งออกเป็น 3 ส่วน คือ Client, Middleware และ Server

- Client เป็นส่วนที่จะ Run Application โดยใช้ระบบ GUI (Graphical User Interface) เป็นการติดต่อกับ User ผ่านระบบกราฟิกซึ่งทำงานแบบเชิงวัตถุ (Object)

- Middleware เป็นส่วนที่ทำงานอยู่ระหว่าง Client และ Server เป็นเสมือนสะพานเชื่อมการทำงาน

- Server เป็นส่วนที่จะ Run Application ในการจัดการทรัพยากรต่างๆ สำหรับระบบ Client/Server สามารถแบ่งออกได้ 4 แบบด้วยกัน คือ

- ระบบฐานข้อมูล SQL
- ระบบจัดการ Transaction
- ระบบ Groupware
- ระบบ Distributed Objects

รูปแบบของ Client/Server ที่ใช้งานจะมีอยู่ 4 ชนิดด้วยกันคือ

1. Stand alone Client/Server การทำงานแบบนี้เซิร์ฟเวอร์จะอยู่บนเครื่องเดียวกับ ไคลเอนต์ ทำให้มีความเร็วในการติดต่อสื่อสารระหว่างผู้ให้บริการและผู้ขอใช้บริการสูงมาก แต่ประสิทธิภาพในการประมวลผลระบบฐานข้อมูลจะลดลงบ้าง ระบบนี้เรียกอีกอย่างว่า Tiny Client/Server

2. Department Client/Server หรือ LAN based single server การทำงานแบบนี้จะมีผู้ให้บริการเกี่ยวกับฐานข้อมูล แอปพลิเคชัน ฯลฯ อยู่บนเครื่องเซิร์ฟเวอร์ และผู้ขอใช้บริการทั้งหลายจะอยู่บนเครื่องไคลเอนต์ โดยจะเชื่อมต่อกันด้วยระบบเครือข่ายท้องถิ่น (LAN) และมีมิดเดิลแวร์ (Middleware) เป็นตัวกลางที่ทำงานอยู่ระหว่างไคลเอนต์และเซิร์ฟเวอร์ ภารติดต่อสื่อสารกันระหว่างผู้ให้บริการและผู้ขอใช้บริการจะช้ากว่าแบบ Stand alone เพราะจะต้องติดต่อผ่านระบบเครือข่าย ยิ่งถ้ามีผู้ขอใช้บริการเข้ามาดึงข้อมูลกันครั้งละมากๆ หลากๆ เครื่อง ประสิทธิภาพจะลดลงอย่างเห็นได้ชัดเจน วิธีเพิ่มประสิทธิภาพก็คือการเพิ่มเครื่องเซิร์ฟเวอร์ขึ้นในระบบ

3. Workgroup Client/Server การทำงานแบบนี้จะเป็นกลุ่มของเซิร์ฟเวอร์ที่หลากหลายแพลตฟอร์ม หลายผู้ผลิต มีความแตกต่างกันของเซิร์ฟเวอร์ แต่ทั้งหมดนี้จะเชื่อมต่อกันทางระบบเครือข่าย LAN และ WAN และใช้มิดเดิลแวร์มาตรฐานในการทำงาน

4. Enterprise Client/Server การทำงานแบบนี้จะทำให้มีการเชื่อมโยงเครื่องเซิร์ฟเวอร์หรือโฮสต์ต่างแพลตฟอร์มเข้าด้วยกัน ทำให้มีการใช้ทรัพยากรในระบบได้อย่างมีประสิทธิภาพสูงสุด โดยที่ไคลเอนต์สามารถจะเลือกใช้ทรัพยากร จากเซิร์ฟเวอร์เครื่องใดก็ได้ผ่านทางมิดเดิลแวร์

2.3 โพรโตคอลของระบบเครือข่าย (Network Protocol) [8]

โพรโตคอลของระบบเครือข่าย (Network Protocol) หรือที่นิยมเรียกกันว่า โพรโตคอล สแตก (Protocol stack) ก็คือชุดของกฎหรือข้อตกลงในการแลกเปลี่ยนข้อมูลผ่าน เครือข่ายคอมพิวเตอร์เพื่อให้แต่ละ Workstation ในเครือข่ายสามารถรับส่งข้อมูลระหว่างกันได้อย่างถูกต้อง โดยโพรโตคอลของระบบเครือข่ายส่วนมากจะทำงานอยู่ในระดับ Network Layer และ Transport Layer ใน OSI Reference Model และทำหน้าที่ในการประสานงานระหว่างแผงวงจรเชื่อมต่อเครือข่าย (NIC) กับระบบปฏิบัติการเครือข่าย (NOS)

ระบบเครือข่ายที่ใช้กันอยู่ในปัจจุบันจะมีโพรโตคอล สแตก ที่ได้รับความนิยมใช้งานกันอยู่หลายโพรโตคอลซึ่งแต่ละโพรโตคอลก็จะจัดการในงานของเครือข่ายที่คล้ายกัน และในกรณีที่ระบบเครือข่ายเชื่อมอยู่กับคอมพิวเตอร์หลายแบบ จะสามารถใช้งานหลาย ๆ โพรโตคอล สแตก พร้อมกันผ่านเครือข่ายได้ เช่น ใช้ IPX/SPX สำหรับ Netware และใช้ TCP/IP ในการติดต่อกับ UNIX ผ่าน LAN แบบ Ethernet พร้อมๆ กัน เป็นต้น

2.3.1 TCP/IP

หลายเทคโนโลยีที่เราใช้อยู่ทั่วไปมีจุดกำเนิดจากเทคโนโลยีการสงคราม IP เน็ตเวิร์กก็เป็นหนึ่งในนั้น เมื่อครั้งสงครามเย็นระหว่างสหรัฐอเมริกาและสหภาพโซเวียต กระทรวงกลาโหมภายใต้รัฐบาลกลางสหรัฐฯ - วิทยาลัยต่าง ๆ ทำวิจัยเพื่อสร้างเครือข่ายที่ทนต่อความล้มเหลว (ด้วยระเบิดนิวเคลียร์) สิ่งที่ได้คือโปรโตคอล TCP/IP เครือข่ายคอมพิวเตอร์ที่ใช้โปรโตคอลนี้เรียกสั้นๆ ว่า TCP/IP (Transmission Control Protocol/Internet Protocol) คือชุดของโปรโตคอลที่รวมกันเป็นกลุ่มให้ใช้งาน เช่น Internet Protocol (IP) , Address Resolution Protocol (ARP), Internet Control Message Protocol (ICMP), User Datagram Protocol (UDP) ฯลฯ แต่โปรโตคอลที่มีบทบาทสำคัญคือ Internet Protocol (IP) โดยมีหลักการทำงานคือ แบ่งเนื้อข้อมูลที่ต้องการส่งเป็นชิ้นเล็ก ๆ เรียกว่าแพ็กเก็ตส่งแพ็กเก็ตไปยังเส้นทางที่เหมาะสมเป็นทอดจนกว่าจะถึงปลายทาง แต่แต่ละแพ็กเก็ตอาจใช้เส้นทางคนละทิศขึ้นกับการพิจารณาของเราเตอร์ในช่วงต่าง ๆ หากเกิดข้อผิดพลาด ณ ช่วงการส่งใด เราเตอร์ที่รับผิดชอบการส่งช่วงนั้นจะจัดส่งแพ็กเก็ตชิ้นนั้นใหม่โดยอัตโนมัติ เมื่อถึงจุดหมายระบบปลายทางจะรวบรวมแพ็กเก็ตกลับให้เป็นเนื้อข้อมูลดั้งเดิม ซึ่งถ้าจะว่ากันตามทฤษฎีแล้ว TCP/IP นั้นจะประกอบด้วยส่วนสำคัญอยู่ 2 ส่วนด้วยกันก็คือ TCP หรือ Transmission Control Protocol และอีกส่วนก็คือ IP หรือ Internet Protocol นั่นเอง การแบ่งลักษณะในการทำงานก็จะแบ่งเป็น TCP มีหน้าที่ในการตรวจสอบการรับส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์ผู้รับ และเครื่องคอมพิวเตอร์ผู้ส่ง ให้ได้รับข้อมูลที่ถูกต้องและครบถ้วนหรือว่าหากมีการสูญหายของข้อมูลก็จะมีกระบวนการแจ้งเตือนให้ต้นทางที่ส่งข้อมูลมารับทราบแล้วให้ทำการส่งข้อมูลมาให้ใหม่

2.3.2 ลักษณะการทำงานของ TCP/IP

ลักษณะการทำงานของ IP นั้น จะทำหน้าที่ในการเลือกเส้นทางที่จะใช้ในการรับส่งข้อมูลในระบบเครือข่าย และทำการตรวจสอบที่อยู่ของผู้รับโดยการใช้ข้อมูลขนาด 4 Byte เป็นตัวกำหนดแอดเดรสหรือที่เราเรียกกันว่า IP Address ซึ่งโปรโตคอล TCP จะทำงานอยู่ในชั้น Transport Layer ตัวแพ็กเก็ต TCP จะประกอบด้วย ส่วนหัว (Header) และส่วนข้อมูล (Data) และโปรโตคอล IP จะทำงานอยู่ในชั้น Network Layer ตัวแพ็กเก็ต IP ประกอบด้วย 2 ส่วนใหญ่ ๆ คือ ส่วนหัว (IP Header) จะประกอบด้วย IP แอดเดรสของเครื่องต้นทางและปลายทาง และส่วนข้อมูล (IP Data) จะเป็นที่เก็บโปรโตคอล TCP เนื่องจากโปรโตคอล TCP/IP จะถูก Encapsulate ให้มาอยู่ในส่วนของแพ็กเก็ต IP

ในการติดต่อสื่อสารกันจริงๆ แล้วเราคงจะไม่สามารถเห็นขั้นตอนการทำงานของระบบ ได้ เพราะเป็นการทำงานของ Software & Hardware แต่เราจะอธิบายเพื่อความเข้าใจของ โปรโตคอล TCP/IP ให้ดูกันดังมีส่วนประกอบดังนี้

- IP Address : สำหรับการรับส่งข้อมูลในระบบ Internet จะถูกกำหนดและอ้างอิงด้วยหมายเลขประจำเครื่องนั้นก็คือเครื่อง IP Address ซึ่งในระบบ Internet จะมีเครื่องคอมพิวเตอร์เป็นจำนวน

มากที่อยู่ในระบบ ในการที่จะใช้ IP Address อาจจะไม่สะดวก จึงได้มีการเปลี่ยนมาใช้เป็นชื่อ ในความเข้ากันก็คือ Domain name โดยทั้งหมดนี้อยู่ในระบบ Name Services ซึ่งเป็นการอ้างอิงชื่อแทนหมายเลขนั่นเอง

- Routing Configuration : ข้อดีของโปรโตคอล TCP/IP ก็คือในการกำหนดเส้นทางสำหรับการรับส่ง ที่สามารถเลือกเส้นทางในการรับส่งข้อมูลได้อย่างอัตโนมัติหากถ้าเกิดเส้นทางบางเส้นทางเสียหาย ระบบกลไกในการกำหนดเส้นทางสำหรับการรับส่งข้อมูลของโปรโตคอล TCP/IP ก็จะเลือกเส้นทางที่เหมาะสมถูกต้องให้สามารถรับส่งข้อมูลได้

- Protocol, Ports, Sockets : เป็นช่องทางสำหรับกำหนดทิศทางของการรับส่งข้อมูล นอกเหนือ จากที่จะต้องกำหนดหลังจาก IP Address

2.3.3 จุดเด่นของโปรโตคอล TCP/IP

- สามารถนำส่งข้อมูล ไปถึงจุดหมายได้แม้เส้นทางบางที่เสียหาย : เป็นจุดประสงค์หลักที่ช่วยให้ทนต่อความล้มเหลว โดยหากระหว่างการสื่อสารข้อมูลและมีเส้นทางใดเสียหายหรือล้มเหลว IP เนตเวิร์กจะปรับใช้เส้นทางอื่นที่ทดแทนได้เพื่อนำส่งข้อมูลให้ไปถึงปลายทางอย่างอัตโนมัติ ผู้ส่งและผู้รับข้อมูลไม่จำเป็นต้องรับรู้หรือปรับแก้แต่ประการใด

- ไม่ขึ้นกับแพลตฟอร์มใดๆ : ไม่ว่าเครื่องข่ายนั้นเป็นเครือข่ายท้องถิ่นหรือเครือข่ายระหว่างภูมิภาค เป็นไฟล์/พริ้นต์เซิร์ฟเวอร์หรือไคลเอ็นต์/เซิร์ฟเวอร์ เป็นระบบปฏิบัติการใด เนตเวิร์กอินเทอร์เฟซเป็นแบบใดก็ตาม ในมุมมองของโปรโตคอล TCP/IP ก็คือ IP เนตเวิร์ก

2.3.4 จุดอ่อนของ IP มี 2 ประเด็น

1. รับส่งโดยไม่มีการรักษาความปลอดภัยเนื้อข้อมูล : การรับส่งข้อมูลด้วย IP แพ็กเก็ตไม่มีทั้งการเข้ารหัสข้อมูลและป้องกันการปลอมแปลงใด ๆ การไม่เข้ารหัสข้อมูลอาจทำให้ผู้ไม่ประสงค์ดีระหว่างเส้นทางที่ IP แพ็กเก็ตผ่านดักข้อมูลอย่างง่ายดาย แม้ว่าเราอาจสามารถบังคับเส้นทางของ IP แพ็กเก็ตได้ ก็ไม่อาจมั่นใจได้ว่าระหว่างทางมีการดักข้อมูลหรือไม่ในเรื่องปัญหาการปลอมแปลงแบ่งออกเป็นสองกรณีคือ การปลอมแปลงหรือดัดแปลงเนื้อข้อมูล และการปลอมแปลงส่วนหัวของ IP แพ็กเก็ต ทั้งสองกรณีให้ผลเหมือนกันคือผู้รับได้ข้อมูลที่ผิดจากความเป็นจริง ทว่าจุดประสงค์ต่างกัน หากเป็นกรณีแรกนั้น ผู้ไม่หวังดีต้องการหลอกหรือกลั่นแกล้งให้ได้ข้อมูลผิด ๆ หากเป็นกรณีหลัง ผู้ไม่หวังดีต้องการแอบอ้างว่าข้อมูลนั้นมาจากแหล่งที่ผู้รับไว้วางใจหรือแหล่งอื่นที่กลายเป็นเหยื่อของการแอบอ้างโดยไม่รู้ตัว

2. รับส่งโดยไม่คำนึงถึงคุณภาพการให้บริการ : การรับส่งต่อ IP แพ็กเก็ต ระหว่างเครือข่ายย่อยไปเป็นทอดนั้นใช้หลักการใครมาก่อนได้ก่อน ฉะนั้นจึงคาดเดาไม่ได้ว่าข้อมูลที่นำส่งไปจะไปถึงปลายทางเมื่อใด แม้ว่า IP เนตเวิร์กใช้หลักการเลือกเส้นทางที่เหมาะสมที่สุดในขณะนั้นก็ตามหากแต่

ความเหมาะสมนั้นผู้ส่งและผู้รับไม่อาจคาดการณ์หรือมีส่วนร่วมตัดสินใจได้เลยว่าจะช้าเร็วหรือมีโอกาสที่ข้อมูลผิดพลาดมากน้อยเพียงไร

2.4 IP Address [7]

IP Address และการอ้างอิงอุปกรณ์ในเครือข่ายเป็นสิ่งจำเป็น เพราะอุปกรณ์ในเครือข่ายทุกชิ้นจะต้องมีหมายเลขประจำตัวและควรต้องไม่ซ้ำกัน เพื่อใช้ในการอ้างอิงถึงกัน สำหรับสิ่งที่ใช้อ้างอิงตัวตนในเครือข่ายของ TCP/IP เรียกว่า IP Address ซึ่งจะต้องเป็นไปตามรูปแบบมาตรฐานที่กำหนดเท่านั้น ไม่สามารถกำหนดได้เองตามใจชอบความสำคัญของ IP Address เหมือนเลขที่บ้านหรือเลขประจำตัว ที่ใช้ในการติดต่อกัน เช่นหากผู้ใช้ต้องการโหลดไฟล์จากคอมพิวเตอร์เครื่องใดเครื่องหนึ่งคุณต้องใช้โปรแกรมสำหรับการโอนถ่ายข้อมูลอย่างโปรแกรม FTP โดยระบุ IP Address ของเครื่องปลายทางที่จะไปดาวน์โหลดข้อมูล ในทางปฏิบัติ จะมีวิธีการใช้ตัวอักษรย่อแทนหมายเลข IP Address เรียกว่า Domain Name

รูปแบบของ IP Address จะเป็นเลขสี่ชุดที่กั้นด้วยจุด (.) เช่น 192.168.0.1 แต่การเก็บค่าในคอมพิวเตอร์จะเป็นเลขฐานสองและไม่มีจุดตั้งตัวอย่าง

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 เนื่องจากเป็นเครือข่ายที่ไม่ได้เกี่ยวข้องกับอินเทอร์เน็ตโดยตรง แต่เพื่อป้องกันความสับสน 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 โดยทั่วไปซอฟต์แวร์ของระบบปฏิบัติการ จะกำหนดค่า Mask ให้เป็นอัตโนมัติ

- Subnet Mask จะใช้ในกรณีที่เราต้องการยืมบางส่วนของหมายเลขเครื่องมาใช้เป็นหมายเลขเครือข่าย เช่น ยืม 1 บิตแรกของหมายเลขเครื่องมาเป็นหมายเลขของเครือข่าย เพื่อให้เกิดเป็นเครือข่ายย่อย หรือ Subnet ขึ้น เช่น เครือข่ายที่ใช้คือ 128.1.x.x โดยที่ X มีค่าระหว่าง 1 – 254 ดังนั้น Network Mask คือ 255.255.0.0 อย่างไรก็ตามเราต้องการแบ่งเครือข่ายออกเป็นเครือข่ายย่อยสองเครือข่าย ดังนั้นเราอาจจะยืม 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 [8]

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 ก็จะทำหน้าที่เกี่ยวกับหลายๆ Layer ของ OSI model

				Layer
FTP, Telnet, Mail Application	Process Layer		Application	7
			Presentation	6
TCP, UDP Protocol	Host-to-Host Layer		Session	5
IP Protocol	Internetwork Layer		Transport	4
			Network	3
Ethernet Driver, Token Ring, etc.	Network Interface		Data Link	2
			Physical	1
	TCP/IP		OSI Model	

รูปที่ 2.4 TCP/IP และ OSI Model

ลำดับชั้นการทำงานของโปรโตคอล TCP/IP เทียบกับมาตรฐาน OSI model นั้น ในชั้นบนสุด เรียกว่า Process layer ทำงาน 2 หน้าที่เทียบได้กับ Application layer และ Presentation layer ในชั้นนี้จะรองรับการทำงานของแอปพลิเคชันต่างๆ ที่ทำงานเป็นโปรเซส อยู่ในเครื่องเซิร์ฟเวอร์ ที่ให้บริการและเครื่องที่ขอใช้บริการ หรือ โคลเอนต์ (Client) ซึ่งจะติดต่อกันผ่านโปรโตคอลเฉพาะแอปพลิเคชันอีกทีหนึ่ง ตัวอย่างเช่น เมื่อผู้ใช้งานอินเทอร์เน็ตต้องการถ่ายโอนไฟล์หรือ 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 เพื่อเรียกดูเว็บเพจในเว็บไซต์ CNN ที่เครื่องซึ่งให้บริการเว็บของ CNN ก็จะมีโปรเซส HTTP (HyperText Transfer Protocol) ทำงานอยู่และจะติดต่อกับผู้ใช้ผ่านโปรโตคอล HTTP เป็นต้น การทำงานของแอปพลิเคชันต่างๆ จะอยู่ที่ Process layer นี้ และมีการติดต่อกันตามแต่ละโปรโตคอลเฉพาะแล้วแต่แอปพลิเคชันที่ใช้งาน จากการที่ Process layer ของ TCP/IP รองรับให้โปรโตคอลอื่นทำงานได้หลายโปรเซสและหลายโปรโตคอลได้พร้อมกันนั้น ทำให้ผู้ใช้สามารถใช้งานได้หลายๆอย่างพร้อมกัน เช่น เปิดโปรแกรม Internet Explorer เพื่อเรียกดูเว็บเพจพร้อมกับใช้งานโปรแกรม Outlook Express เพื่อรับส่งอีเมลล์ไปพร้อมกันได้โดยไม่ต้องรอให้ทำงานอย่างหนึ่งอย่างใดเสร็จก่อน หรือในปัจจุบันมีการพัฒนาโปรแกรม Web Browser ให้สามารถเรียกใช้

งาน โพรโทคอลได้มากขึ้น ทำให้เราสามารถใช้โปรแกรม Web Browser โอนถ่ายไฟล์ข้อมูลที่ใช้ โพรโทคอล FTP ได้โดยไม่ต้องไปหาโปรแกรมอื่นมาใช้

โพรโทคอลหลักๆที่ทำงานใน Process layer ซึ่งผู้ใ้ใช้มักจะคุ้นเคยกันดีได้แก่ FTP (File Transfer Protocol), Telnet, HTTP (HyperText Transfer Protocol) และ SMTP (Simple Mail Transfer protocol) นอกจากนี้ยังมีโพรโทคอลอื่นที่อยู่เบื้องหลัง ซึ่งทำงานโดยที่ผู้ใช้ไม่สามารถมองเห็นได้จากโปรแกรม หรือไม่ได้ที่การใช้งานโดยตรง เช่น

- โพรโทคอล DNS (Domain Name System) ที่ทำหน้าที่แปลงข้อมูลชื่อ domain name หรือชื่อเว็บไซต์ทั้งหลายให้เป็นหมายเลข IP Address

- โพรโทคอล SNMP (Simple network management Protocol) ใช้ในการควบคุมและตรวจสอบอุปกรณ์ที่อยู่ในเครือข่าย

- โพรโทคอล DHCP (Dynamic Host Configuration Protocol) ทำหน้าที่แจกจ่ายข้อมูลพารามิเตอร์ของเครือข่ายให้กับเครื่องลูกข่ายที่เชื่อมต่ออยู่

2.6 ความรู้พื้นฐานเกี่ยวกับ Microsoft Visual Basic Version 6.0 [3]

ภาษา Visual Basic เป็นเครื่องมือสำหรับเขียนโปรแกรมที่ได้รับความนิยมทั่วโลก รวมทั้งในประเทศไทย และเป็นภาษาที่เข้าใจได้ไม่ยากนัก ซึ่งความเป็นมาของภาษา Basic นั้นได้ถูกสร้างขึ้นในปี 1963 โดย Horn Keneny และ Thomas Kutz ที่มหาวิทยาลัย Dartmouth สร้างขึ้นเพื่อใช้ในการสอนการเขียนโปรแกรมโดยเน้นภาษาง่ายต่อการเข้าใจและการใช้งาน โดยที่เราสามารถทำการสร้างแอปพลิเคชันที่สามารถใช้งานกับระบบ Windows ได้ง่ายขึ้น ทำให้มีการสร้างการเขียนโปรแกรมแบบ Visual Programming กำเนิดขึ้นมา รูปแบบนี้ก็คือ การเขียนโปรแกรมพร้อมกับการเห็นผลลัพธ์ที่เกิดขึ้น Visual Basic เป็นหนึ่งใน Visual Programming ที่ไม่ใคร่ซอพท์สร้างขึ้นมา และด้วยความเรียบง่ายของภาษาและการเขียนโปรแกรมที่รวดเร็ว ทำให้ได้รับความนิยมสูงในปัจจุบัน

การเขียนโปรแกรมเพื่อสร้างแอปพลิเคชันจาก Visual Basic นั้นจะมีวิธีการสร้างที่แตกต่างจากโปรแกรมอื่นเพราะ Visual Basic จะทำงานแบบ Event-Driven ซึ่งก็คือการเขียนโปรแกรมแบบ “ถ้าเหตุการณ์เกิดขึ้น เราจะดำเนินการอย่างไร” — กล่าวคือ ถ้าเหตุการณ์นี้เกิดขึ้น เราจะตอบสนองกับเหตุการณ์นี้อย่างไร เป็นโปรแกรมที่รองรับเหตุการณ์ Visual Basic เป็นโปรแกรมที่สนับสนุนการเขียนโปรแกรมแบบ Event-Driven โดยมีเครื่องมือต่าง ๆ ช่วยในการจัดการกับเหตุการณ์ต่าง ๆ ที่จะเกิดขึ้น การเขียนโปรแกรมบนระบบปฏิบัติการ Windows ให้สามารถติดต่อสื่อสารบนระบบเครือข่าย ปัจจุบันนี้เครื่องคอมพิวเตอร์ส่วนใหญ่หันมาใช้ระบบปฏิบัติการ Windows กันเนื่องจากมีเครื่องมือ (Tool) ที่ช่วยให้เราสามารถเขียนโปรแกรมติดต่อผ่านระบบเครือข่าย หนึ่งในเครื่องมือที่สามารถช่วยในการ

ติดต่อสื่อสารผ่านทางระบบเครือข่าย คือ Microsoft Winsock Control 6 ซึ่งเป็นเครื่องมือที่อำนวยความสะดวกสำหรับการเขียนโปรแกรมติดต่อผ่านระบบเครือข่าย โดยโปรโตคอล TCP/IP

2.6.1 Microsoft Winsock Control 6

ส่วนที่ใช้ในการคอนโทรลที่ชื่อว่า Microsoft Winsock Control 6 ที่มากับ Visual Basic เป็นเครื่องมือที่มีประโยชน์ในการพัฒนา โปรแกรมบน Internet เมื่อเปรียบเทียบกับคอนโทรล ActiveX ตัวอื่น ๆ Winsock จะเป็นคอนโทรลที่ถูกนำมาใช้งานมากคอนโทรลหนึ่ง แต่อย่างไรก็ตามถ้าต้องการที่จะคอนโทรลขึ้นมาเอง และไม่ต้องการใช้คอนโทรลของผู้พัฒนาคนอื่น ซึ่งอาจมีข้อผิดพลาดที่คุณไม่ต้องการ คุณก็จำเป็นจะต้องมีพื้นฐานที่เกี่ยวกับ Network Protocols และหลักในการติดต่อสื่อสาร คุณสมบัติของ Winsock เหมาะสำหรับการใช้งานต่างๆ ดังนี้

- สร้างแอปพลิเคชันของไคลเอนต์ (Client) ที่รวบรวมสารสนเทศของผู้ใช้ก่อนจะส่งคำสั่งไปยังเซิร์ฟเวอร์ (Server) ส่วนกลาง
- สร้างแอปพลิเคชันของเซิร์ฟเวอร์ (Server) เป็นศูนย์กลางในการรวบรวมข้อมูลจากผู้ใช้ต่างๆ จำนวนมาก
- สร้างแอปพลิเคชันสำหรับการพูดคุย (Chat)

2.6.2 การใช้ Control ของ Winsock

การควบคุม Winsock ที่ดีช่วยให้ TCP/IP ได้ง่ายและสะดวกขึ้นมาก ไมโครซอฟท์ได้ห่อหุ้ม API ของ Winsock และ inet เป็นแพ็คเกจ ที่สามารถรวมให้เป็นไฟล์เดียวเหมือนต้นฉบับภายในแอปพลิเคชันต่างๆ ของ Visual Basic ได้ง่าย การควบคุม Winsock จะช่วยในการสร้าง แอปพลิเคชันของไคลเอนต์หรือเซิร์ฟเวอร์ที่สามารถเชื่อมโยงกับตัวคอนโทรล Winsock และทำการแลกเปลี่ยนข้อมูลทางช่องทางนั้น ส่วนทางเซิร์ฟเวอร์จะยอมรับการติดต่อกันได้หลายไคลเอนต์ พร้อมทั้งรายงานสถานะของแต่ละเครื่องที่เชื่อมต่อกัน

- คุณสมบัติต่างๆ ของการคอนโทรล Winsock

Winsock ช่วยในการสร้างไคลเอนต์และเซิร์ฟเวอร์โดยใช้คอนโทรลอย่างเดียวกัน การคอนโทรล Winsock ใช้คุณสมบัติต่างๆ ที่เหมือนกันไม่ว่าจะสร้างไคลเอนต์หรือเซิร์ฟเวอร์ คุณสมบัติของ Winsock จะช่วยสนับสนุนสารสนเทศ ในการกำหนดเงื่อนไขการเชื่อมโยงของ Socket, คุณสมบัติของโปรโตคอล จะช่วยระบุโปรโตคอลที่ใช้ในการสื่อสารเซิร์ฟเวอร์ของอินเทอร์เน็ตส่วนใหญ่ใช้โปรโตคอลของ 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 = sekTCPProtocol
RemoteHostIP	ส่งกลับแอดเดรสของ IP ของคอมพิวเตอร์ที่ห่างไกล	Var = Winsock1.RemoteHostIP
RemoteHost	ส่งกลับหรือกำหนดชื่อหรือแอดเดรสของ IP ของคอมพิวเตอร์ที่อยู่ห่างไกล	Winsock1.RemoteHost = _ "100.0.0.1"หรือftp.microsoft.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	รอกอยการเชื่อมโยง
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

และในตารางต่อไปนี้จะแสดงรายการของการเชื่อมต่อต่างๆ ที่สามารถใช้ในการควบคุม Winsock

ตารางที่ 2.5 ตารางแสดงรายการของการเชื่อมต่อต่างๆ ที่สามารถใช้ในการควบคุม Winsock

Method	คำอธิบาย
Accept	ยอมรับการเชื่อมโยงที่ร้องขอในเหตุการณ์ ConnectionRequest
Bind	เลือกอุปกรณ์ของเครือข่ายท้องถิ่นและพอร์ตท้องถิ่นที่ทำการเชื่อมต่อกันในเครือข่าย
Close	ปิด socket รับฟัง สำหรับการเชื่อมโยงของ TCP และกำหนดคอนโทรลใหม่ เพื่อสามารถเปลี่ยน โปรโตคอล
Connect	สร้างการเชื่อมโยง TCP ให้ Host ที่ห่างไกล
Listen	รอกอยคอมพิวเตอร์อื่นทำการเชื่อมโยงกับ Server (เฉพาะ โปรโตคอลของ TCP เท่านั้น)
PeekData	นำข้อมูลออกจากบัฟเฟอร์มาแสดงผล แต่ไม่เคลื่อนหน่วยความจำของบัฟเฟอร์นั้น
SendData	ส่งข้อมูลไปให้คอมพิวเตอร์ที่ห่างไกล
DataArrival	เกิดขึ้นเมื่อได้รับข้อมูลจากคอมพิวเตอร์ที่ห่างไกล
Error	เกิดขึ้นเมื่อมีการผิดพลาด
SendComplete	เกิดขึ้นเมื่อการส่งข้อมูลเสร็จสิ้น
SendProgress	เกิดขึ้นระหว่างการส่งผ่านข้อมูล

2.6.3 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 บนระบบปฏิบัติการบนวินโดวส์เป็นเรื่องค่อนข้างยุ่งยากมากกว่าการใช้ฟังก์ชันมาตรฐานต่างๆ ที่ผู้ใช้สร้างขึ้นเพื่อการใช้งาน โดยโปรแกรมเมอร์ของ Visual Basic อาจมองฟังก์ชัน API ของ Windows มีส่วนที่คล้ายกับฟังก์ชันต่างๆ ของ Visual Basic ที่บรรจุตัวพารามิเตอร์ต่างๆ เกี่ยวกับอินพุตและเอาต์พุตและบางครั้งมีการส่งค่ากลับ อย่างไรก็ตามฟังก์ชันต่างๆ ของ API ได้คอมไพล์เรียบร้อยแล้ว อยู่ในไฟล์แยกกัน ที่ทราบกันทั่วไปในนามสกุลของ *.DLL (Dynamic Link Library) และการใช้ฟังก์ชันต่างๆ เหล่านี้ โปรแกรมเมอร์ต้องเพิ่มค่าอีกเล็กน้อยเพื่อกำหนดฟังก์ชันภายนอกสำหรับ 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 ซึ่งอาจไม่สอดคล้องกับชื่อที่กำหนดตั้งแต่แรกให้ฟังก์ชันในโปรแกรม ตัวอย่างฟังก์ชันชื่อ “_lopen” ปรากฏใน DLL ของ kernel32 แต่ “_lopen” จะเป็นชื่อฟังก์ชันที่ไม่ถูกต้องภายใน Visual Basic ในกรณีนี้ การประกาศของ API จะต้องมีรูปแบบดังนี้

```
Declare Function lopen Lib "kernel32" Alias
```

```
    "_lopen"(ByVal IpPathName 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 ของวินโดวส์ ซึ่งจะเป็นไฟล์ DLL ที่อยู่ในไดเรกทอรี \Windows\System โดยที่ไฟล์ DLL จะพบได้ทุกๆ เครื่อง PC ที่มีระบบปฏิบัติการวินโดวส์ ไฟล์ที่จำเป็นจะติดตั้งไว้ ในขณะที่ติดตั้งระบบปฏิบัติการวินโดวส์ต่างๆ ไฟล์ DLL ที่สำคัญของวินโดวส์ได้แก่

- USER32.DLL เป็นไลบรารีที่บรรจุฟังก์ชันต่างๆ ที่เกี่ยวข้องกับสภาพแวดล้อมของระบบวินโดวส์ เช่น การเชื่อมโยงข่าวสารต่างๆ ระหว่างวินโดวส์, การจัดการเคอร์เซอร์ต่างๆ, การจัดการบนเมนู และการเชื่อมโยงกับฟังก์ชันอื่นๆ แต่ไม่ได้แสดงผลออกหน้าจอ

- KERNEL32.DLL เป็นไลบรารีที่บรรจุค่าต่างๆ ไว้มากมาย ที่ช่วยจัดการฟังก์ชันต่างๆ ของระบบปฏิบัติการในระดับต่ำ โดยหน้าที่ทั่วไป เช่น การจัดการเรื่องหน่วยความจำ, การเชื่อมโยงของแหล่งทรัพยากร, การจัดการเกี่ยวกับไฟล์และไดเรกทอรี และการจัดการเกี่ยวกับโมดูล

- GDI32.DLL (Graphics Device Interface) จะมีฟังก์ชันที่ช่วยจัดการเอาท์พุทให้อุปกรณ์อื่นๆ โดยเฉพาะส่วนของจอภาพ

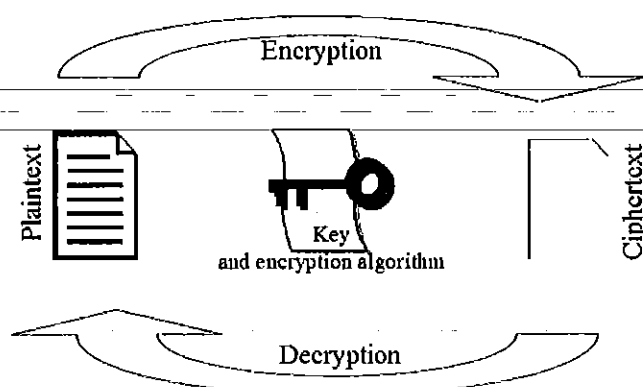
2.7 ความรู้พื้นฐานเกี่ยวกับระบบรหัส (Cryptography) [6]

ระบบรหัส มีขึ้นเพื่อเปลี่ยนข้อมูลที่เป็นข้อความปกติ(plain text)ให้เป็นข้อมูลที่เข้ารหัส(cipher text)ด้วยการ Encryption และถูกแปลงกลับเป็น ข้อความที่อ่านได้โดยการ Decryption แบ่งเป็น 2 ประเภท คือ

- แบบกุญแจสมมาตร(Symmetric/Secret Key)
- แบบกุญแจอสมมาตร(Asymmetric/Public Key)

2.7.1 แบบกุญแจสมมาตร(Symmetric/Secret Key)

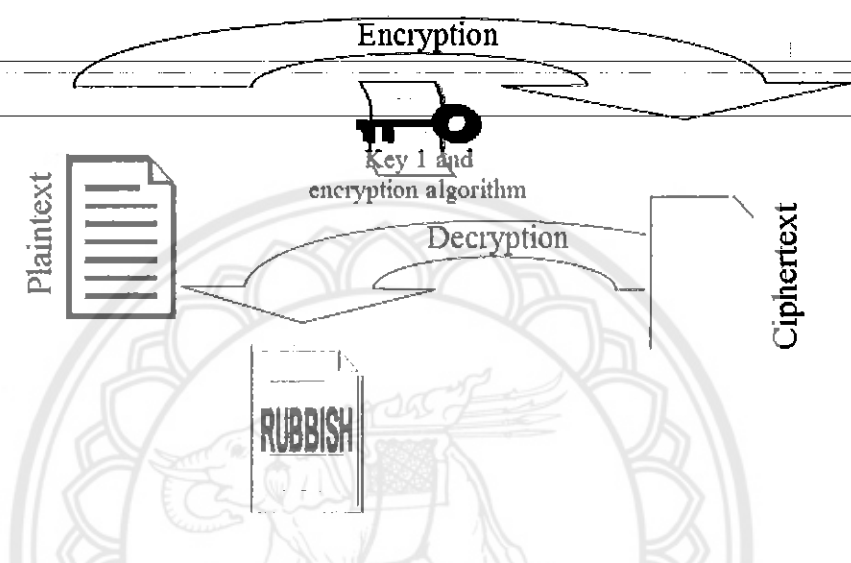
ผู้ส่งและผู้รับต้องตกลงกันเรื่องรูปแบบการรับ-ส่งข้อมูล ใช้กุญแจที่เหมือนกันทั้งในการเข้ารหัสและถอดรหัส เรียกว่า กุญแจลับ(secret key) ข้อดีของแบบกุญแจสมมาตรคือ รวดเร็ว ฟังก์ชันคณิตศาสตร์ที่ใช้ไม่ซับซ้อนมากนัก ข้อมูลที่เข้ารหัสแล้วไม่ใหญ่กว่าเดิมมากนัก ข้อเสียคือกุญแจลับเฉพาะเจาะจงกับตัวบุคคลที่เราติดต่อ ถ้าหากมีการแจกจ่ายกุญแจให้หลายฝ่ายที่แม้จะเกี่ยวข้องกับงาน ก็ทำให้ความปลอดภัยลดลง



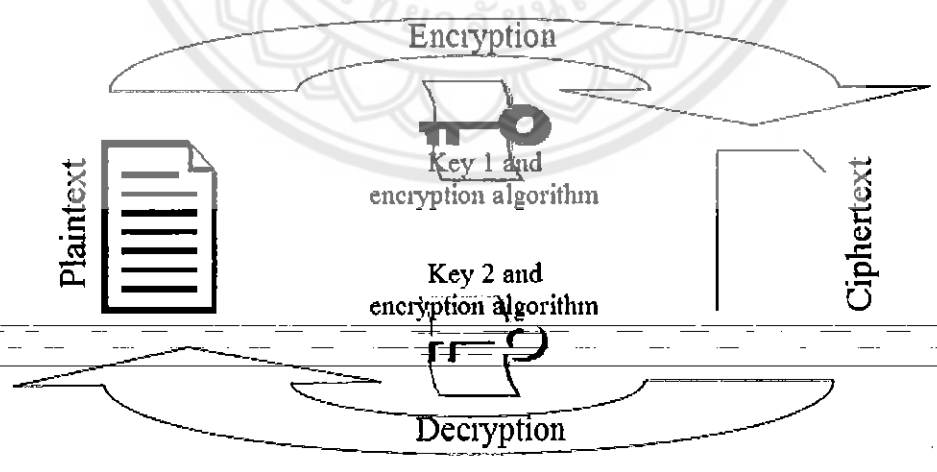
รูปที่ 2.5 แผนผังการ Symmetric encryption

2.7.2 แบบกุญแจอสมมาตร(Asymmetric/Public Key)

ใช้หลักการกุญแจคู่ประกอบด้วย กุญแจสาธารณะ(public key)ซึ่งเปิดเผยได้และกุญแจส่วนตัว(private key)เก็บอยู่ที่ตัวเจ้าของเท่านั้น ใช้ฟังก์ชันทางคณิตศาสตร์ที่ซับซ้อน กุญแจทั้งสองเป็นคู่กัน แต่ไม่ใช่อันเดียวกัน ข้อดีคือการจัดการกุญแจง่ายกว่า เป็นประโยชน์ในการสร้างลายมือชื่ออิเล็กทรอนิกส์ แก้ปัญหาการกระจายของกุญแจ ข้อเสียคือใช้เวลาเข้ารหัส,ถอดรหัสนาน มีฟังก์ชันซับซ้อน ข้อมูลที่เข้ารหัสแล้วมีขนาดใหญ่



รูปที่ 2.6 แผนผังของ Asymmetric/Public Key



If Key 1 = private, Key2 must be corresponding *public*
 If Key 1 = public, Key2 must be corresponding *private*

รูปที่ 2.7 แผนผังของ Asymmetric/Public Key

ในการเข้ารหัสกุญแจนั้นด้วยระบบการเข้ารหัสของ RSA ซึ่งเป็นการผลิตพื้นฐานจาก 2 เลขจำนวนเฉพาะขนาดใหญ่ ซึ่งได้จากการเข้ารหัสจากตัวจริง ซึ่งยากที่จะทำให้ factor ของเลขขนาดใหญ่ทำการแปลงเป็นชนิดที่ตรงกันข้ามกับอย่างเดิมจะเห็นได้ว่าระบบรหัสแบบกุญแจสมมาตรมีประสิทธิภาพในการรักษาความปลอดภัยให้กับข้อมูลมากกว่า

2.8 RSA Algorithm [6]

เป็นการสร้าง Ciphertext คือ การสร้างรหัสข้อความ โดยการใช้ลักษณะเฉพาะ 1 ลักษณะ ซึ่งได้จากการเข้ารหัสจากกุญแจสาธารณะ การเข้ารหัสถูกกระทำด้วยกุญแจสาธารณะเท่านั้น ซึ่งในการที่จะถอดรหัสได้นั้นต้องใช้กับกุญแจส่วนตัวที่เป็นคู่กันของกุญแจสาธารณะนี้ ซึ่งให้ผู้รับที่สามารถถอดรหัสและอ่านข้อความต้นฉบับก่อนที่จะมีการเข้ารหัสได้ ซึ่งสมการ การเข้ารหัสจะเป็นดังนี้

$$c = m^e \pmod n$$

โดยที่ m คือ ข้อความที่เข้ารหัส และ c คือผลที่ได้จากการเข้ารหัส การแสดงการดำเนินการโดยเฉพาะ คือ การยกกำลัง $m^e \pmod n$ เมื่อ e และ n เป็นกุญแจสาธารณะของผู้รับ ของ Ciphertext ซึ่งสมการการถอดรหัสจะเป็นดังนี้

$$m = c^d \pmod n$$

โดยที่ c คือ ข้อความรหัส และ m คือผลที่ได้จากการถอดรหัส การแสดงการดำเนินการโดยเฉพาะ คือ การยกกำลัง $c^d \pmod n$ เมื่อ d และ n เป็นกุญแจส่วนตัวของผู้รับ ตัวอย่างเช่น ให้

$$p = 61, q = 53, n = p, q = 3233, e = 17, d = 2753$$

Public key ในที่นี้คือ $(e, n) = (17, 3233)$ และ Private key ในที่นี้คือ $(d, n) = (2753, 3233)$

การเข้ารหัส ข้อความที่มีค่า 123 คือ

$$\text{Encrypt}(123) = 123^{17} \pmod{3233} = 855$$

การถอดรหัส Ciphertext ที่มีค่า 855 คือ

$$\text{Decrypt}(855) = 855^{2753} \pmod{3233} = 123$$

บทที่ 3

การศึกษาและพัฒนาโปรแกรม

การพัฒนาโปรแกรมระบบควบคุมระยะไกลสำหรับคอมพิวเตอร์ที่ใช้ Windows XP

โปรแกรมระบบควบคุมระยะไกลสำหรับคอมพิวเตอร์ที่ใช้ Windows XP ประกอบด้วย
โปรแกรม 2 โปรแกรมด้วยกันคือ โปรแกรมควบคุมและโปรแกรมลูกข่าย

1. โปรแกรมควบคุม เป็นโปรแกรมหลักที่ถูกติดตั้งและถูกเรียกใช้บนเครื่องคอมพิวเตอร์ของผู้ดูแลระบบซึ่งทำหน้าที่เป็นเครื่องแม่ข่าย (server) ของระบบเครือข่าย โปรแกรมนี้จะทำหน้าที่ควบคุมเครื่องคอมพิวเตอร์ลูกข่ายและคอยตรวจสอบการทำงานต่างๆ โดยโปรแกรมจะทำหน้าที่รอรับสัญญาณเชื่อมต่อจากโปรแกรมลูกข่าย เมื่อโปรแกรมลูกข่ายถูกเรียกให้ทำงาน (จะถูกเรียกให้ทำงานทุกครั้งเมื่อเปิดระบบปฏิบัติการวินโดวส์) โปรแกรมควบคุมจะสามารถรู้ได้ว่าเครื่องคอมพิวเตอร์ของลูกข่ายเครื่องใดบ้างที่ติดต่อเข้ามา โดยโปรแกรมนี้จะแสดงหมายเลข IP ของเครื่องคอมพิวเตอร์ลูกข่ายนั้นๆ ให้ทราบ

2. โปรแกรมลูกข่าย เป็นโปรแกรมที่ถูกติดตั้งไว้ที่เครื่องคอมพิวเตอร์ลูกข่ายซึ่งทำหน้าที่เป็นเครื่องลูกข่าย(client) ของระบบเครือข่าย โปรแกรมนี้จะถูกสั่งให้ทำงานทันทีหลังจากที่ลูกข่ายเปิดระบบปฏิบัติการวินโดวส์ ซึ่งเมื่อโปรแกรมนี้ทำงานก็จะส่งสัญญาณเชื่อมต่อกับโปรแกรมควบคุมเพื่อให้เครื่องคอมพิวเตอร์ควบคุมรู้ว่าเครื่องคอมพิวเตอร์ของลูกข่ายติดต่อเข้ามา โดยโปรแกรมนี้ทำหน้าที่รับคำสั่งต่างๆจากโปรแกรมควบคุมมาทำงานอีกทีหนึ่ง โดยโปรแกรมทั้ง 2 โปรแกรมมีองค์ประกอบที่สำคัญในการเขียนโปรแกรมดังนี้

3.1 เครื่องมือที่ใช้พัฒนาโปรแกรม

1. ภาษาที่ใช้พัฒนาโปรแกรม คือ Microsoft Visual Basic 6.0
2. Components ที่ใช้ในการสื่อสาร คือ Microsoft Winsock Control
3. โปรโตคอลที่ใช้ในการสื่อสาร คือ TCP/IP
4. - Algorithm ที่ใช้ในการเข้ารหัสข้อมูลคือ RSA Algorithm

3.2 การเขียนโปรแกรมบน Client-Server

โดยหน้าที่พื้นฐานของ TCP/IP นั้นจะช่วยในเรื่องของการสร้างและคงสถานะในการติดต่อไปยังเครื่องลูกข่ายต่างๆ สิ่งแรกที่เราควรจะต้องรู้เมื่อเรากำลังสร้าง Network Application สำหรับเครื่อง Client นั้น คือ เราต้องรู้ หมายเลข IP ของเครื่อง Server หรือชื่อเครื่อง Server สิ่งที่ 2 คือ ต้องรู้ Port ที่เครื่อง Server ใช้ในการติดต่อ เมื่อทำการสร้าง Server Application เราจำเป็นต้องเปิด Port เพื่อทำ

ปี ๒๕๖๓

การรองรับฟังการติดต่อจากเครื่อง Client เมื่อเครื่อง Client ทำการติดต่อเข้าโดยตั้งค่าที่ตรงกับค่าที่กล่าวมาของเครื่อง Server แล้ว Application ทั้งสองจะสามารถติดต่อสื่อสารและแลกเปลี่ยนข้อมูลกันได้ และ Application ทั้งสองจะสิ้นสุดการติดต่อเมื่อ Server Application หรือ Client Application ปิดช็อกเกตและหยุดการเชื่อมต่อหรือการปิดโปรแกรมฝ่ายใดฝ่ายหนึ่งนั่นเอง การเขียน โปรแกรมติดต่อสื่อสารผ่านระบบเครือข่ายกันโดยใช้โปรโตคอล TCP นั้นถูกพัฒนาขึ้นมาตามโครงสร้างของ Client-Server model นั่นคือ เมื่อ โปรแกรมสอง โปรแกรมทำหน้าที่เชื่อมต่อกัน โปรแกรมหนึ่งจะทำหน้าที่เป็นโปรแกรมควบคุมซึ่งถือได้ว่าโปรแกรมนี้เป็นโปรแกรมฝั่ง server โดยโปรแกรมที่เป็นโปรแกรมควบคุมนั้นจะเริ่มทำการเชื่อมต่อด้วยการสร้างช็อกเกตของตนขึ้นมาแล้วพยายามส่งสัญญาณข้อมูลร้องขอไปยังโปรแกรมที่ทำหน้าที่เป็นโปรแกรมลูกข่ายซึ่งถือได้ว่าเป็นโปรแกรมฝั่ง client โดยก่อนหน้านั้น โปรแกรมที่ทำหน้าที่เป็นโปรแกรมลูกข่ายจะต้องสร้างช็อกเกตขึ้นมาเพื่อส่งสัญญาณไปยังโปรแกรมควบคุมที่ส่งสัญญาณติดต่ออยู่ก่อนแล้วเมื่อโปรแกรมควบคุมเปิดสัญญาณการเชื่อมต่อและ โปรแกรมลูกข่ายก็ร้องขอการเชื่อมต่อการตอบรับการเชื่อมต่อจะเกิดขึ้นเป็นผลให้เส้นทางระหว่าง 2 โปรแกรมนั้นถูกจัดตั้งขึ้นและพร้อมจะรับส่งข้อมูลระหว่างกันและกันเราสามารถสรุปขั้นตอนการเชื่อมต่อข้อมูลของ โปรแกรมที่ทำหน้าที่เป็นโปรแกรมควบคุมได้ดังต่อไปนี้

1. สร้าง server ช็อกเกต
2. กำหนดหมายเลข service port ที่ต้องการให้เครื่องคอมพิวเตอร์ลูกข่ายติดต่อเข้ามา
3. รอฟังสัญญาณการเชื่อมต่อจาก โปรแกรมลูกข่าย
4. ตอบรับการเชื่อมต่อเมื่อมีเครื่องคอมพิวเตอร์ลูกข่ายติดต่อเข้ามา
5. รับส่งข้อมูล
6. ปิดช็อกเกตและหยุดการเชื่อมต่อ

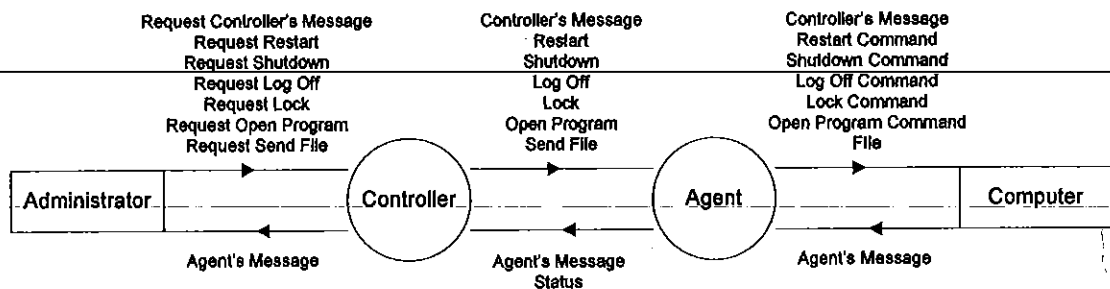
มก
จศ 2๕
มวอ
๕๒

สำหรับโปรแกรมที่เป็นโปรแกรมลูกข่ายสามารถสรุปขั้นตอนได้ดังนี้

1. สร้าง client ช็อกเกต
2. กำหนด IP address และหมายเลข service port ให้ตรงกับโปรแกรมควบคุมกำหนดไว้
3. ส่งสัญญาณการเชื่อมต่อกับโปรแกรมควบคุม
4. รับส่งข้อมูล
5. ปิดช็อกเกตและหยุดการเชื่อมต่อ

3.3 การออกแบบระบบ (Context Diagram)

Context Diagram ของโปรแกรม ระบบควบคุมระยะไกลสำหรับคอมพิวเตอร์ที่ใช้ Windows XP



รูปที่ 3.1 Context Diagram

3.4 แผนภาพกระแสข้อมูล (Data Flow Diagram)

เป็นแผนภาพที่ใช้ในการวิเคราะห์โครงสร้าง โดยแผนภาพจะแสดงความสัมพันธ์ระหว่างโปรแกรมกับข้อมูลที่เกี่ยวข้อง เส้นทางการเดินของข้อมูลและเหตุการณ์ใดกับข้อมูลที่เข้าโปรแกรม และผลตอบสนองที่ออกมาจากโปรแกรมดังกล่าว

3.4.1 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 การสั่งให้เครื่องลูกข่ายทำการ Restart หรือ Shutdown ให้กับระบบและส่งไฟล์ไปยังเครื่องลูกข่าย
2. Computer จะถูกเตรียมพร้อมที่จะทำงานเพื่อรอรับคำสั่ง Restart, Shutdown, Log off, Lock, หรือ รับส่งไฟล์ ที่ส่งมาจาก Controller

List of Process

1. Agent Remote มีหน้าที่ส่งคำสั่งที่ใช้ในการควบคุมคอมพิวเตอร์ เช่นการ Restart หรือ การ Shutdown และส่งให้กับโปรแกรมถัดไป เพื่อให้ดำเนินการต่อไป
2. Controller Chat มีหน้าที่ส่งข้อความจากเครื่องของผู้ดูแลระบบไปยังเครื่องลูกข่ายและรับข้อความที่ส่งมาจากเครื่องลูกข่ายและแสดงผลต่อผู้ดูแลระบบต่อไป

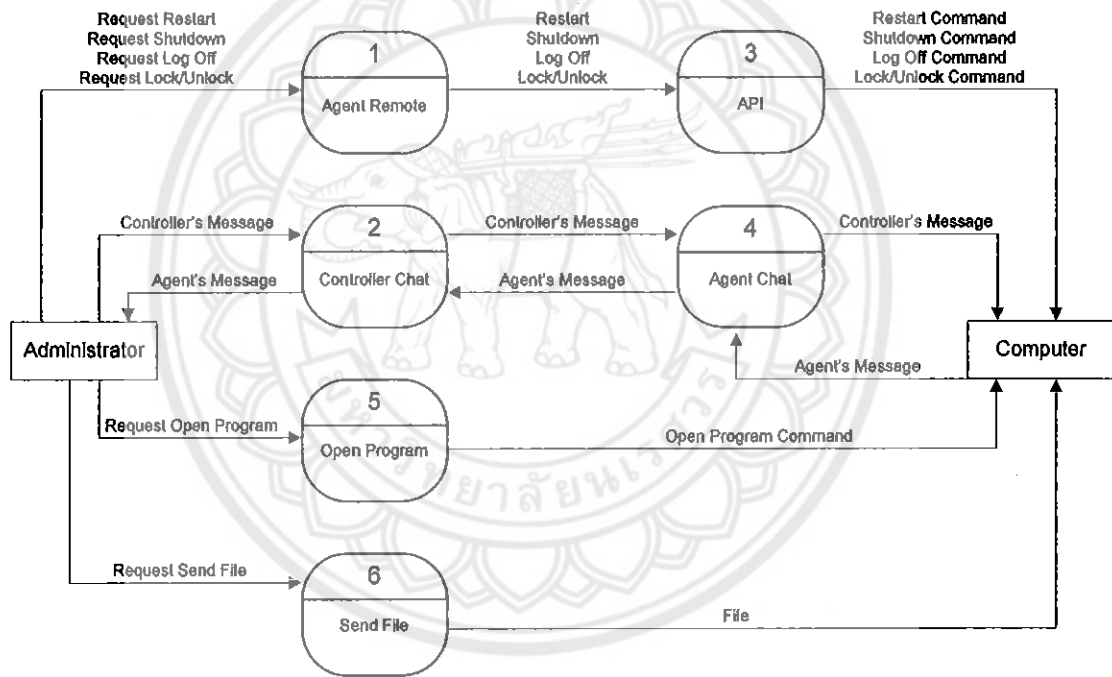
3. API มีหน้าที่ประสานการทำงานจากระบบ โดยแบ่งการประสานงานออกเป็นสองส่วน ส่วนแรก ประสานงานเกี่ยวกับการควบคุมคอมพิวเตอร์ ซึ่งรับคำสั่งจาก Agent Remote เช่น การ Restart หรือ การ Shutdown โดยสร้างคำสั่งเพื่อกระทำการดังกล่าว

4. Agent Chat มีหน้าที่รับข้อความจากเครื่องของผู้ดูแลระบบ เพื่อทำการแสดงผลต่อเครื่องลูกข่าย และส่งข้อความจากเครื่องลูกข่ายนำไปแสดงผลต่อผู้ดูแลระบบต่อไป

5. Open Program มีหน้าที่ส่งชื่อโปรแกรมที่ต้องการเปิด ไปยังเครื่องคอมพิวเตอร์ลูกข่าย เพื่อให้เครื่องคอมพิวเตอร์ลูกข่าย เปิดโปรแกรมที่ต้องการ

5. Send File มีหน้าที่ส่งไฟล์จากเครื่องของผู้ควบคุม ไปยังเครื่องคอมพิวเตอร์ลูกข่าย

List of data



รูปที่ 3.2 แสดงกระแสข้อมูลระดับ 1

3.4.2 แผนภาพแสดงข้อมูลระดับสอง (Data Flow Diagram Level-2)

เป็นการแสดงโปรเซสย่อย ที่สามารถแยกได้จากโปรเซสที่มีระดับ 1 เพื่อการวิเคราะห์การทำงานภายในโปรเซสนั้นๆ

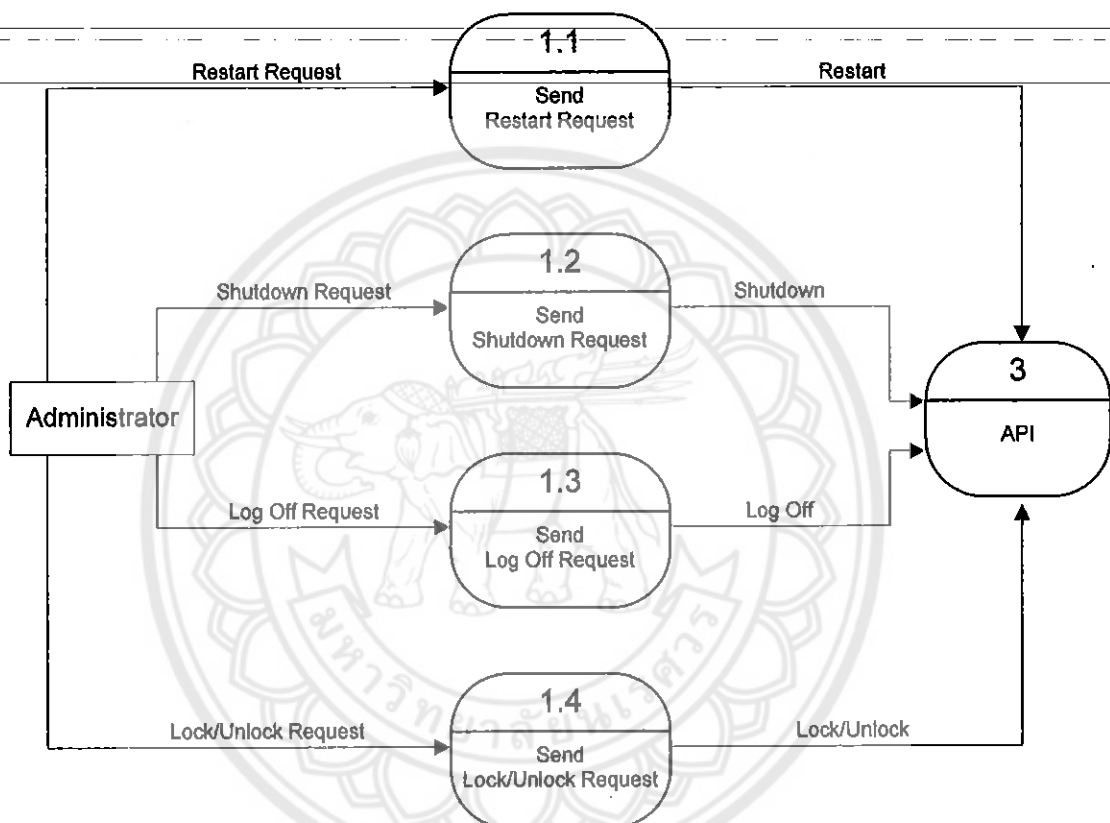
โปรเซสที่ 1 Agent Remote มี 4 โปรเซสย่อย

โปรเซสที่ 1.1 Send Restart Request มีหน้าที่สร้างคำขอสำหรับการ Restart เครื่อง Agent และส่งคำขอดังกล่าวเข้าสู่โปรเซส API ต่อไป

โปรเซสที่ 1.2 Send Shutdown Request มีหน้าที่สร้างคำขอสำหรับการ Shutdown เครื่อง Agent และส่งคำขอดังกล่าวเข้าสู่โปรเซส API ต่อไป

โปรเซสที่ 1.3 Send Log Off Request มีหน้าที่สร้างคำขอสำหรับการ Log Off เครื่อง Agent และส่งคำขอดังกล่าวเข้าสู่โปรเซส API ต่อไป

โปรเซสที่ 1.4 Send Lock/Unlock Request มีหน้าที่สร้างคำขอสำหรับการ Lock/Unlock เครื่อง Agent และส่งคำขอดังกล่าวเข้าสู่โปรเซส API ต่อไป

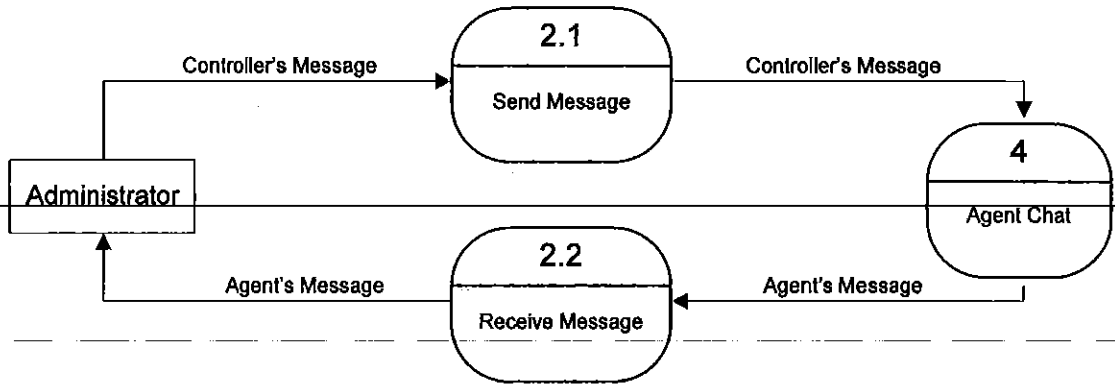


รูปที่ 3.3 กระแสของข้อมูลระดับ 2 ของโปรเซส Agent Remote

โปรเซสที่ 2 Controller Chat มี 2 โปรเซสย่อย

โปรเซสที่ 2.1 Send Message มีหน้าที่ส่งข้อความจากผู้ดูแลระบบและส่งข้อความดังกล่าวให้กับเครื่องลูกข่าย

โปรเซสที่ 2.2 Receive Message มีหน้าที่รับข้อความจากเครื่องลูกข่ายและส่งข้อความดังกล่าวให้กับผู้ดูแลระบบ



รูปที่ 3.4 กระแสของข้อมูลระดับ 2 ของโปรเซส Controller Chat

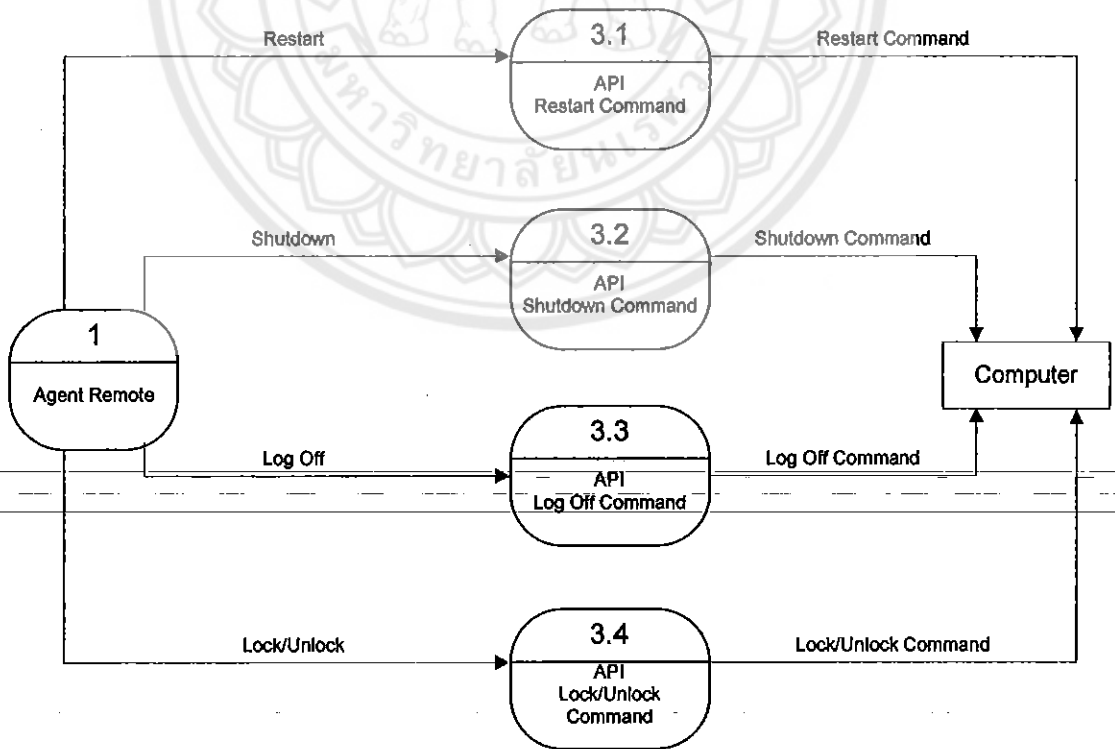
โปรเซสที่ 3 API เป็นการควบคุมการทำงานของคอมพิวเตอร์มี 4 โปรเซสย่อย

โปรเซสที่ 3.1 API for Restart Command มีหน้าที่สร้างคำสั่งในการ Restart เครื่องลูกข่าย

โปรเซสที่ 3.2 API for Shutdown Command มีหน้าที่สร้างคำสั่งการ Shutdown เครื่องลูกข่าย

โปรเซสที่ 3.3 API for Log Off Command มีหน้าที่สร้างคำสั่งการ Log Off เครื่องลูกข่าย

โปรเซสที่ 3.4 API for Lock/Unlock Command มีหน้าที่สร้างคำสั่งการ Lock/Unlock เครื่องลูกข่าย

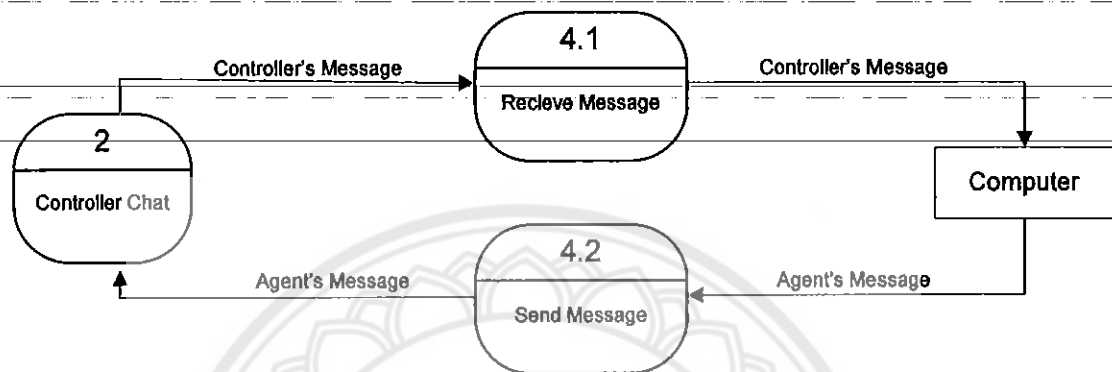


รูปที่ 3.5 กระแสของข้อมูลระดับ 2 ของโปรเซส API

โปรเซสที่ 4 Agent Chat มี 2 โปรเซสย่อย

โปรเซสที่ 4.1 Receive Message มีหน้าที่รับข้อความจากผู้ดูแลระบบและส่งข้อความดังกล่าวให้กับเครื่องลูกข่าย

โปรเซสที่ 4.2 Send Message มีหน้าที่ส่งข้อความจากเครื่อง Agent และส่งข้อความดังกล่าวให้กับผู้ดูแลระบบ



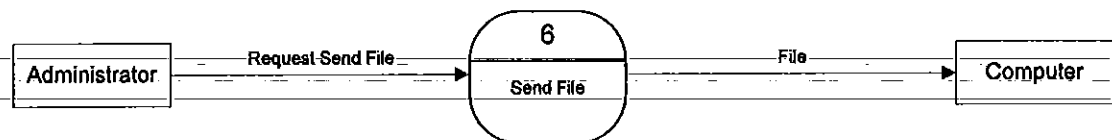
รูปที่ 3.6 กระแสของข้อมูลระดับ 2 ของโปรเซส Agent Chat

โปรเซสที่ 5 Open Program ทำหน้าที่ส่งชื่อโปรแกรมที่ต้องการให้เครื่องลูกข่ายเปิด ร้องขอไปยังเครื่องลูกข่ายให้เปิดโปรแกรมนั้นๆ



รูปที่ 3.7 กระแสของข้อมูลระดับ 2 ของโปรเซส Open Program

โปรเซสที่ 6 Send File ทำหน้าที่ส่งไฟล์ หรือเพิ่มเอกสารทุกชนิด ไปยังเครื่องคอมพิวเตอร์ลูกข่าย



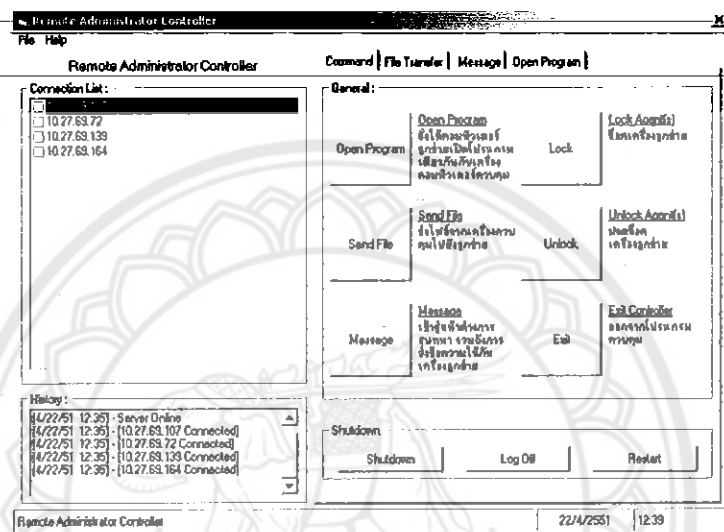
รูปที่ 3.8 กระแสของข้อมูลระดับ 2 ของโปรเซส Send File

3.5 การออกแบบในส่วนของอินเทอร์เน็ตเฟสที่ติดต่อกับผู้ใช้ของโปรแกรม

ในการออกแบบส่วนของการอินเทอร์เน็ตเฟสได้แบ่งออกเป็นสองส่วน คือ ส่วนที่ติดต่อกับผู้ดูแลระบบและส่วนที่ติดต่อกับเครื่องลูกข่าย สามารถแสดงผลของการออกแบบอินเทอร์เน็ตเฟสที่ติดต่อกับผู้ดูแลระบบและเครื่องลูกข่ายได้ดังนี้

3.5.1 ส่วนของการออกแบบ Controller

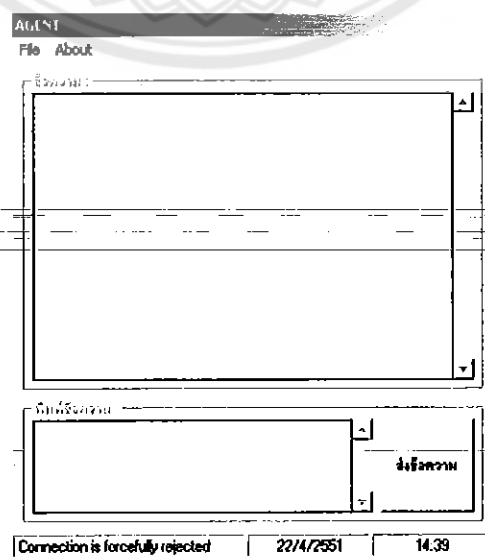
ผลของการออกแบบอินเทอร์เน็ตเฟสที่ใช้ติดต่อกับผู้ดูแลระบบในส่วนของการ Controller ดังรูป 3.9



รูปที่ 3.9 ส่วนของการอินเทอร์เน็ตเฟส Controller

3.5.2 ส่วนของการออกแบบ Agent

ผลของการออกแบบอินเทอร์เน็ตเฟสที่ใช้ติดต่อกับผู้ดูแลระบบในส่วนของการ Agent ดังรูป 3.10



รูปที่ 3.10 ส่วนของการอินเทอร์เน็ตเฟสเครื่องลูกข่าย (Agent)

บทที่ 4

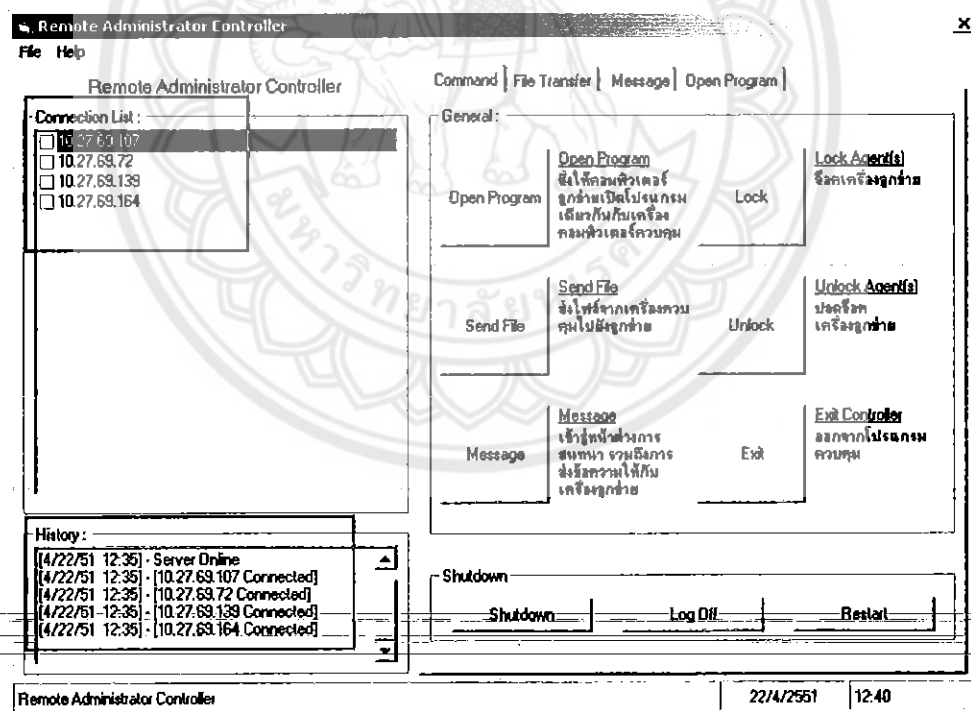
การทดสอบและวิเคราะห์การทำงาน

เนื้อหาในการทดสอบและวิเคราะห์การทำงานจริงของโปรแกรม เพื่อจะได้ทราบว่า โปรแกรมที่พัฒนาจะสามารถปฏิบัติงานได้ตามที่ต้องการหรือไม่ โดยการทดสอบนั้นสามารถแบ่งการทดสอบของโปรแกรมเป็นสองส่วนใหญ่ ๆ คือ

4.1 การทดลองระหว่างการพัฒนาโปรแกรม

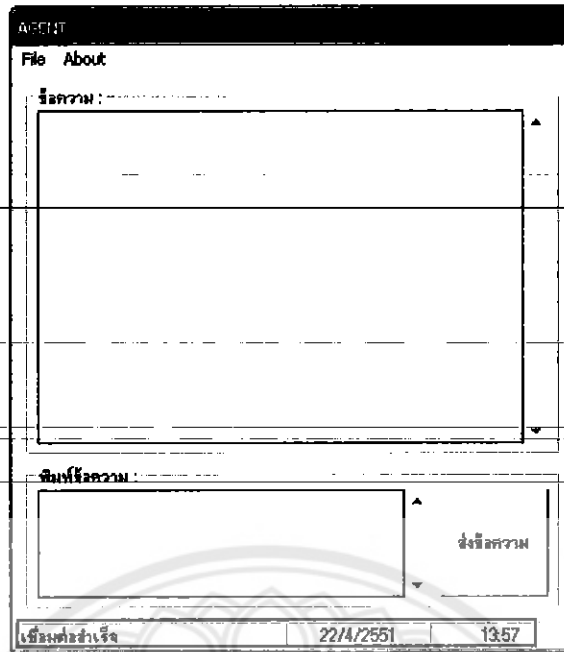
4.1.1 ผลทดสอบการเชื่อมต่อระหว่างตัวควบคุมกับเครื่องลูกข่าย

หลังจากกำหนดค่าพอร์ทและไอพีสำหรับการเชื่อมต่อเรียบร้อยแล้ว ให้ทำการรันตัวควบคุมและรอการเชื่อมต่อจากเครื่องลูกข่าย สิ่งที่แสดงการเชื่อมต่อสำเร็จนั้นตัวควบคุมสามารถรู้ได้โดยค่าไอพีของเครื่องลูกข่ายจะแสดงบนหน้าจอของตัวควบคุมและรายงานสถานะการเชื่อมต่อทางกล่องข้อความด้านล่างของตัวควบคุมดังรูป 4.1



รูปที่ 4.1 การเชื่อมต่อจากเครื่องลูกข่าย

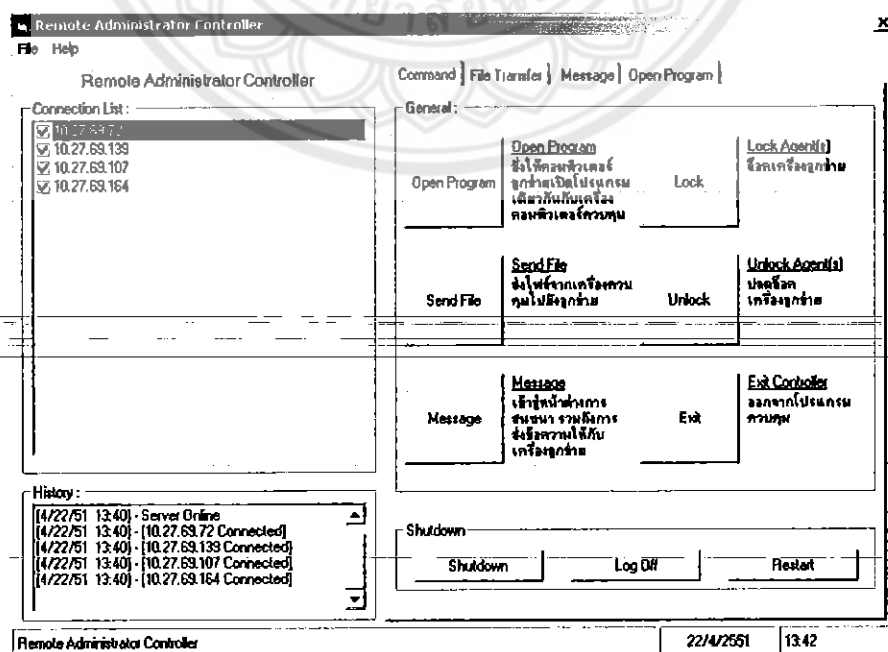
ส่วนที่จะแสดงการเชื่อมต่อที่เครื่องลูกข่ายสามารถทราบได้จากการแสดงสถานะการเชื่อมต่อถึงตัวควบคุม โดยจะแสดงข้อความการเชื่อมต่อสำเร็จ แสดงวันที่และเวลาที่เชื่อมต่อกับเครื่องของผู้ดูแลระบบดังรูป 4.2



รูปที่ 4.2 การเชื่อมต่อถึงตัวควบคุม

4.2 การทดสอบส่วนการควบคุม (Controller)

ในส่วนนี้เป็นการให้ตัวควบคุมทำงาน โดยตัวควบคุมจะเป็นส่วนที่คอยส่งคำสั่งต่าง ๆ ไปยังเครื่องลูกข่ายที่เป็นเครื่องเป้าหมาย และ โปรแกรมยังสามารถรายงานสถานะของเครื่องลูกข่ายนั้น ๆ ว่าปฏิบัติงานในส่วนใดของคำสั่งอยู่ หรือมีการเปิดเครื่องเพื่อรอรับคำสั่งอยู่หรือไม่ แสดงได้ดังรูป 4.3 ซึ่งแสดงการเชื่อมต่อกันระหว่างตัวควบคุมกับเครื่องลูกข่าย



รูปที่ 4.3 การทำงานของตัวควบคุม (Controller)

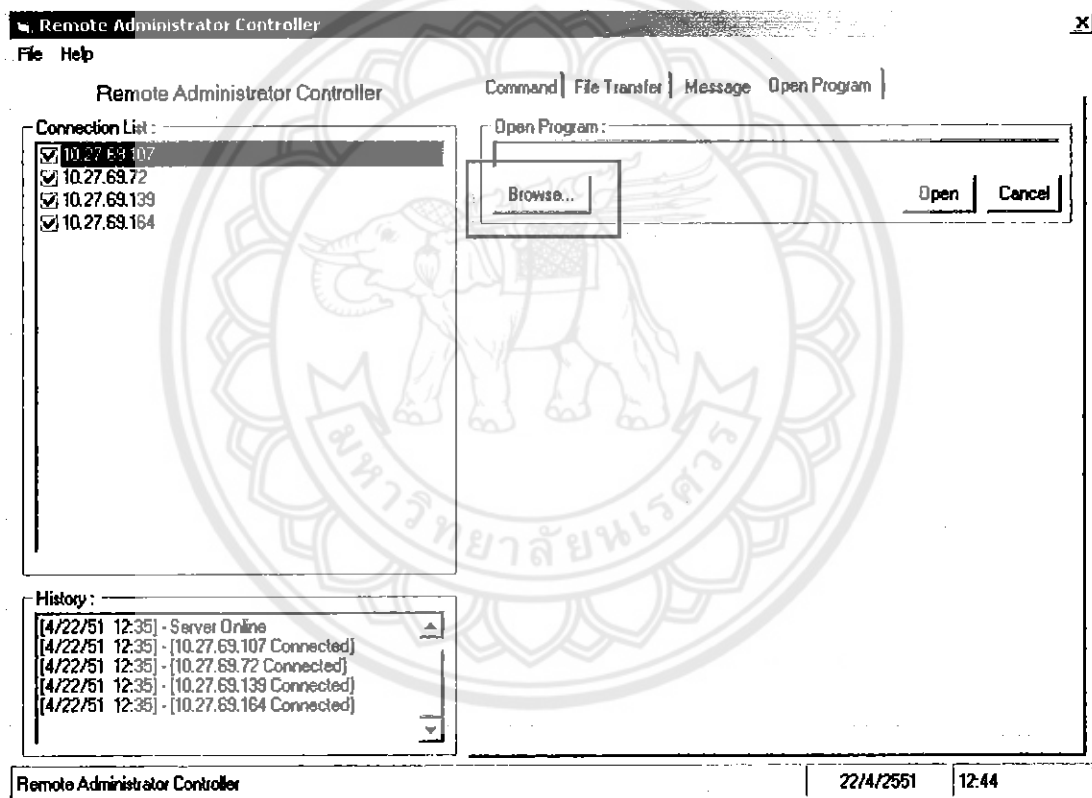
4.2.1 การส่งคำสั่งเปิดโปรแกรมที่เครื่องลูกข่าย (Open Program)

ในส่วนของการดำเนินการนี้เป็นการสั่งให้เครื่องลูกข่าย เปิดโปรแกรมที่ต้องการใช้ โดยสั่งเปิดโปรแกรมจากเครื่องของผู้ดูแลระบบ

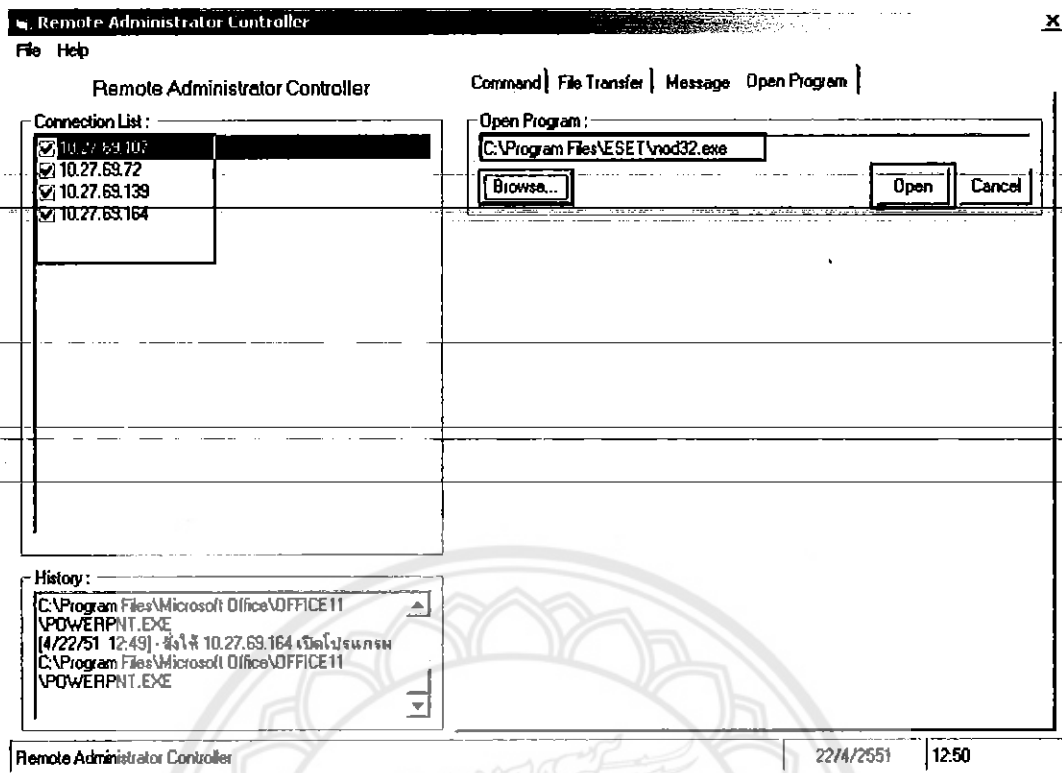
- เมื่อเครื่องควบคุมส่งคำสั่ง Open Program ไปยังเครื่องลูกข่าย จะมีหน้าจอสำหรับ Browse ไปหาโปรแกรมที่ต้องการเปิดในเครื่องลูกข่ายออกทางหน้าจอของเครื่องควบคุม ดังรูป 4.4

- เมื่อคลิกปุ่ม Browse แล้วให้ผู้ดูแลระบบเลือกโปรแกรมที่ต้องการเปิด แล้วกดที่ปุ่ม Open ดังรูป 4.5

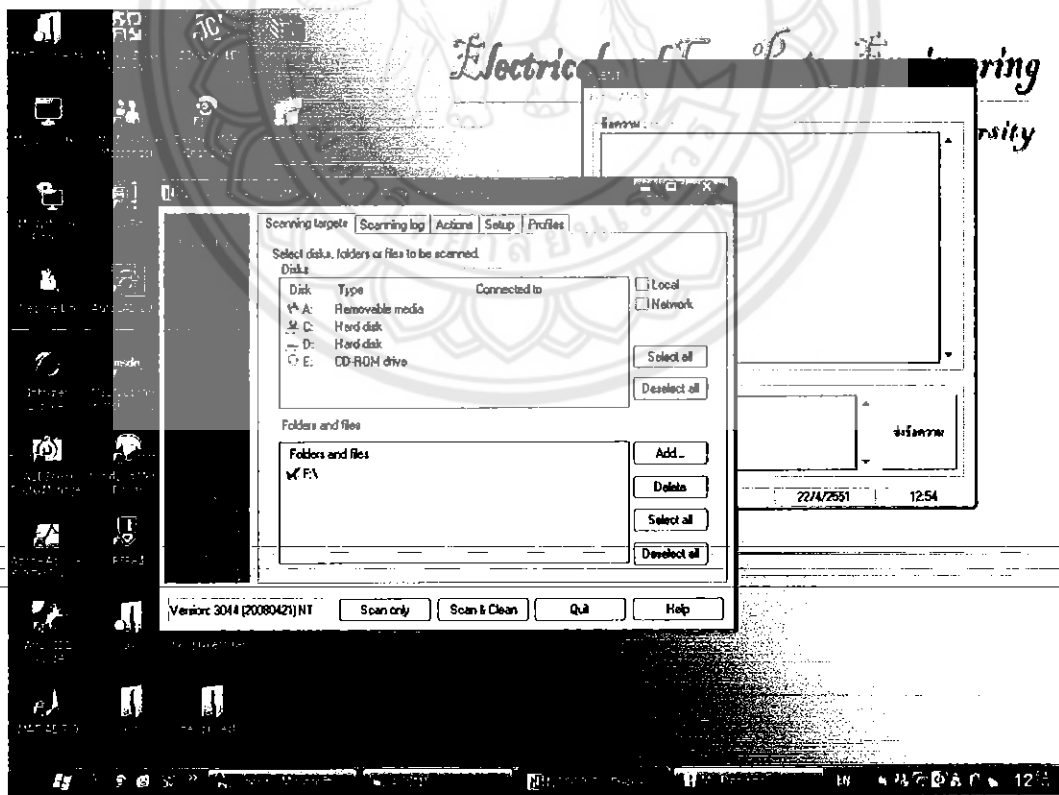
- เมื่อคลิกปุ่ม Open ก็จะทำให้การเปิดโปรแกรมดังกล่าวที่เครื่องลูกข่าย ในที่นี้ได้ทำการสั่งให้เครื่องลูกข่ายเปิดโปรแกรม NOD32 ดังรูป 4.6



รูปที่ 4.4 แสดงการทำงานของคำสั่ง Open Program



รูปที่ 4.5 แสดงการทำงานของคำสั่ง Open Program



รูปที่ 4.6 แสดงการหน้าจอเครื่องลูกข่ายที่ถูกสั่งเปิด โปรแกรม NOD32

4.2.2 การส่งคำสั่งส่งไฟล์ข้อมูลไปยังเครื่องลูกข่าย (Send File)

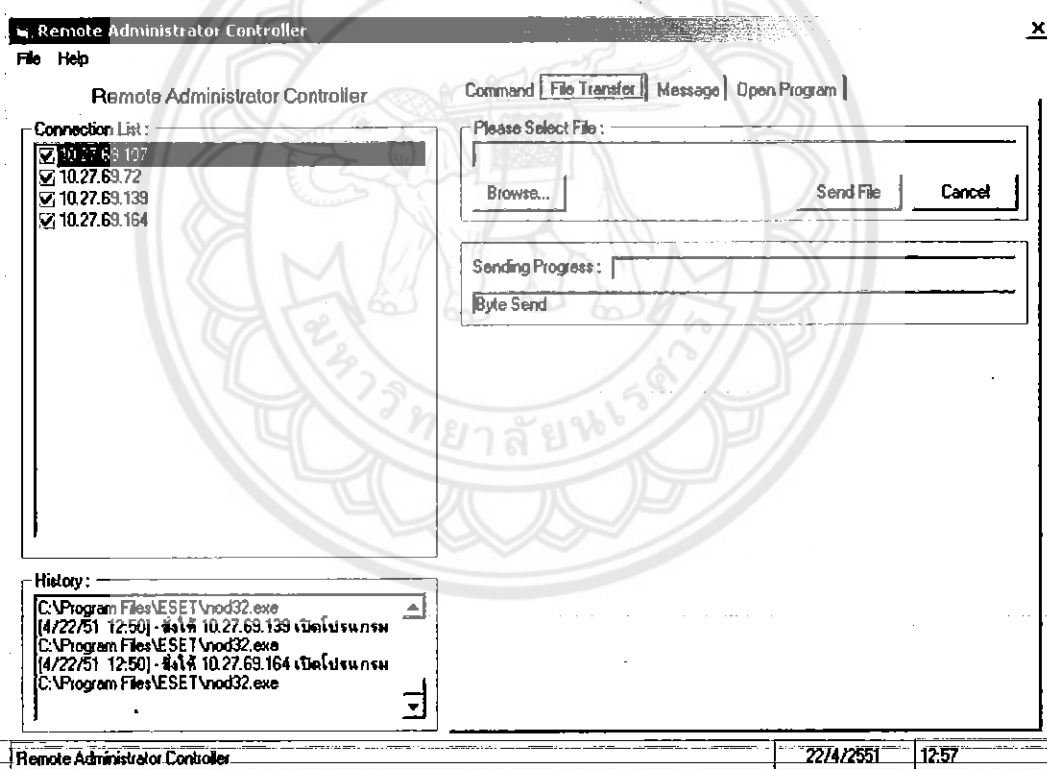
ในส่วนนี้เป็นการส่งไฟล์ต่างๆ จากเครื่องของผู้ดูแลระบบไปยังเครื่องลูกข่าย โดยสามารถส่งได้พร้อมกันที่หลายๆเครื่อง

- เมื่อเครื่องควบคุมส่งคำสั่ง Send File ไปยังเครื่องลูกข่าย จะมีหน้าจอสำหรับ Browse ไปหาไฟล์ที่ต้องการส่งให้เครื่องลูกข่ายออกทางหน้าจอของเครื่องควบคุม ดังรูป 4.7

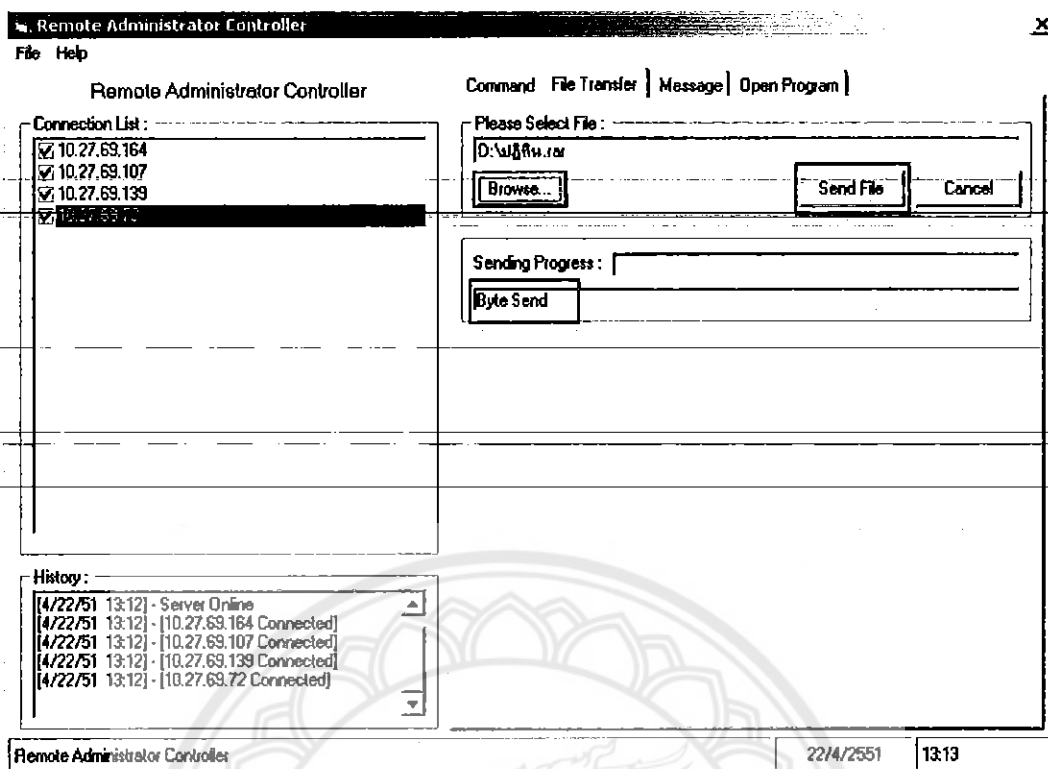
- เมื่อกดปุ่ม Browse แล้วให้ผู้ดูแลระบบเลือกไฟล์ที่ต้องการส่ง แล้วกดที่ปุ่ม Sent File ด้านล่างจะมีแถบบอกสถานะการส่งไฟล์คือแถบ Sending Progress ดังรูป 4.8

- เมื่อส่งไฟล์เสร็จแล้วแถบสถานะ Sending Progress จะบอกสถานะว่า Send Complete ดังรูปที่ 4.9

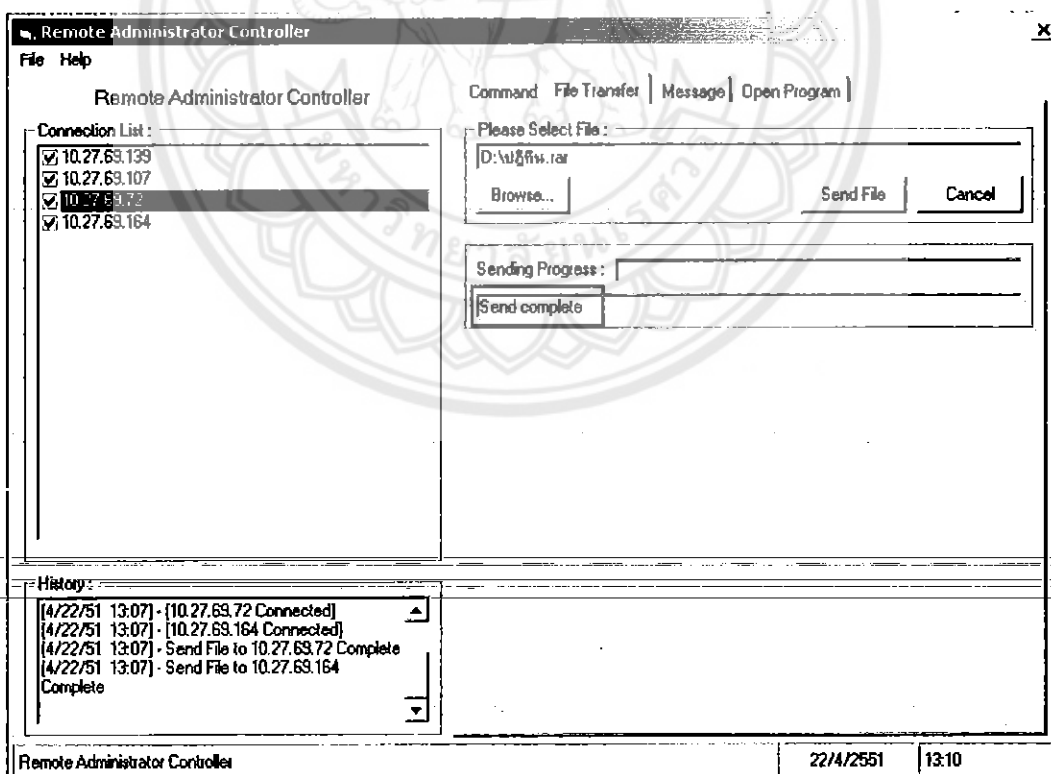
- ไฟล์ที่ผู้ดูแลระบบส่งไปนั้นจะถูกเก็บอยู่ที่ Folder Agent ของเครื่องลูกข่าย ดังรูปที่ 4.10



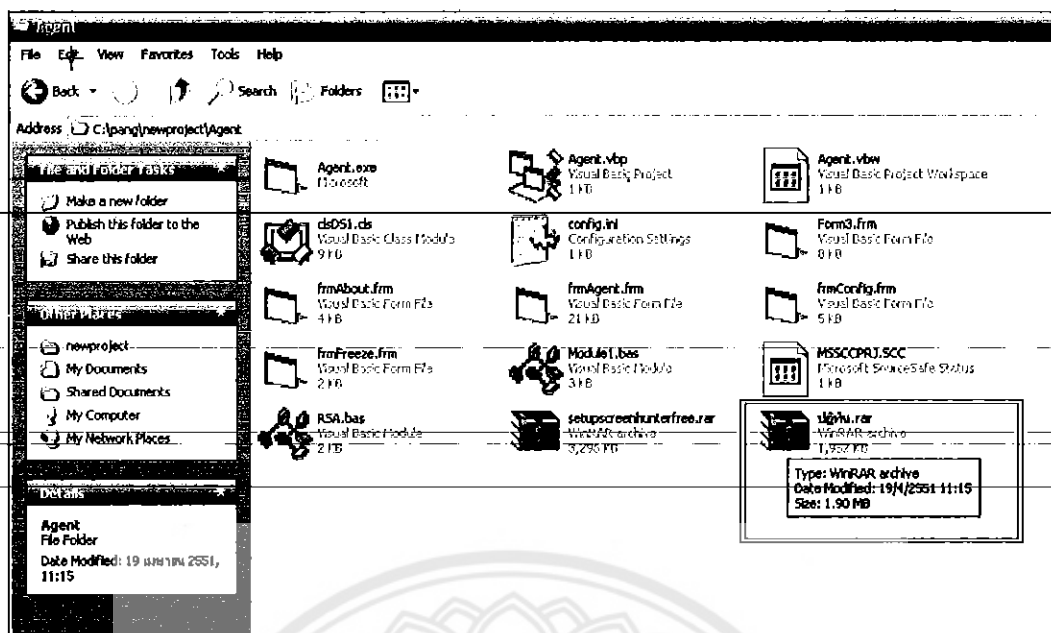
รูปที่ 4.7 รูปแสดงการทำงานของคำสั่ง Send File



รูปที่ 4.8 รูปแสดงการ Browse File ในคำสั่ง Send File



รูปที่ 4.9 รูปแสดงขั้นตอนในการส่งไฟล์ในคำสั่ง Send File

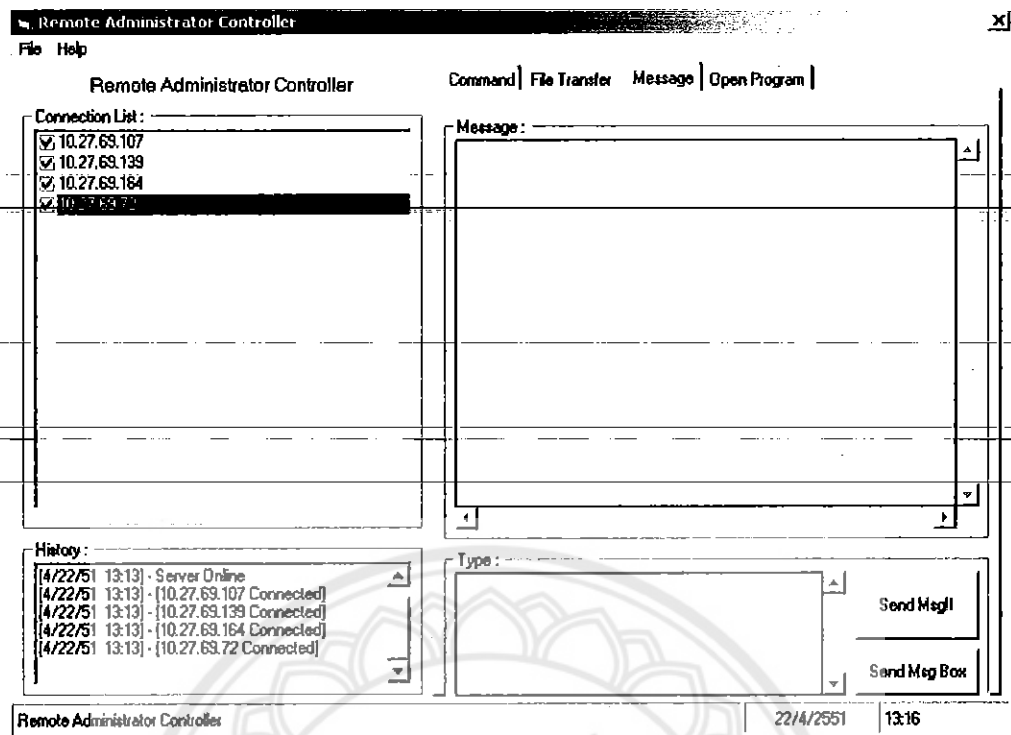


รูปที่ 4.10 รูปแสดงไฟล์ที่ถูกส่งจากเครื่องผู้ดูแลระบบมายังเครื่องลูกข่าย

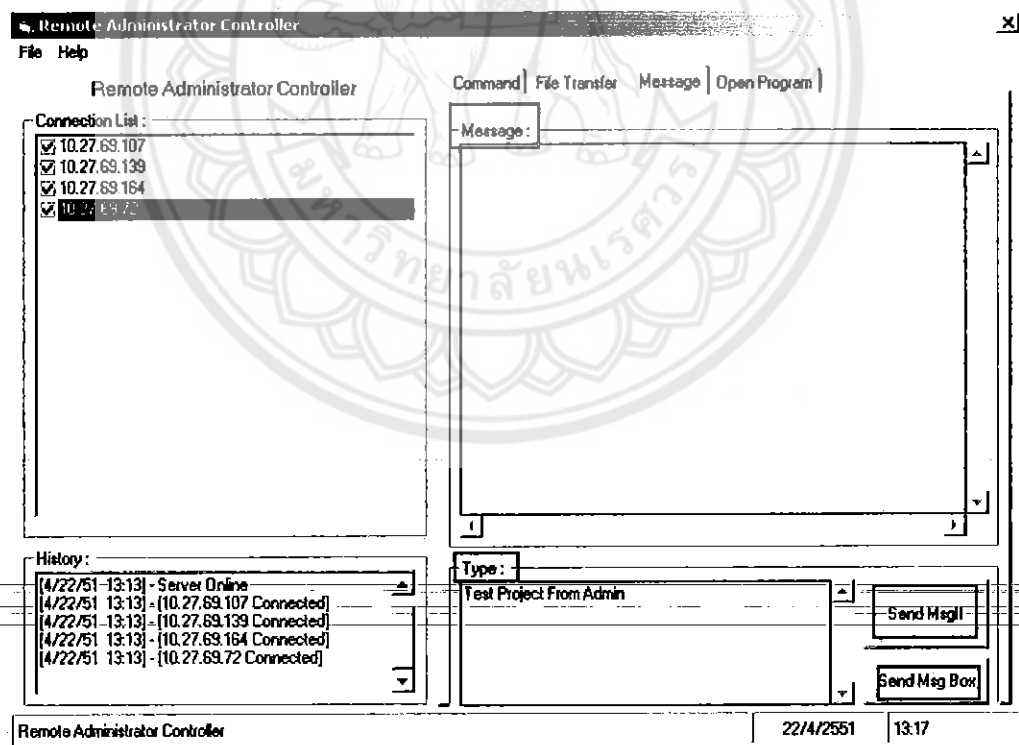
4.2.3 ระบบการสื่อสารภายในจากตัวควบคุม (Controller's Chat)

ใช้ในการติดต่อกับผู้ใช้ เมื่อผู้ดูแลระบบ ต้องการที่จะตรวจสอบหรือดูแลรักษาระบบของเครื่องลูกข่ายในขณะนั้น

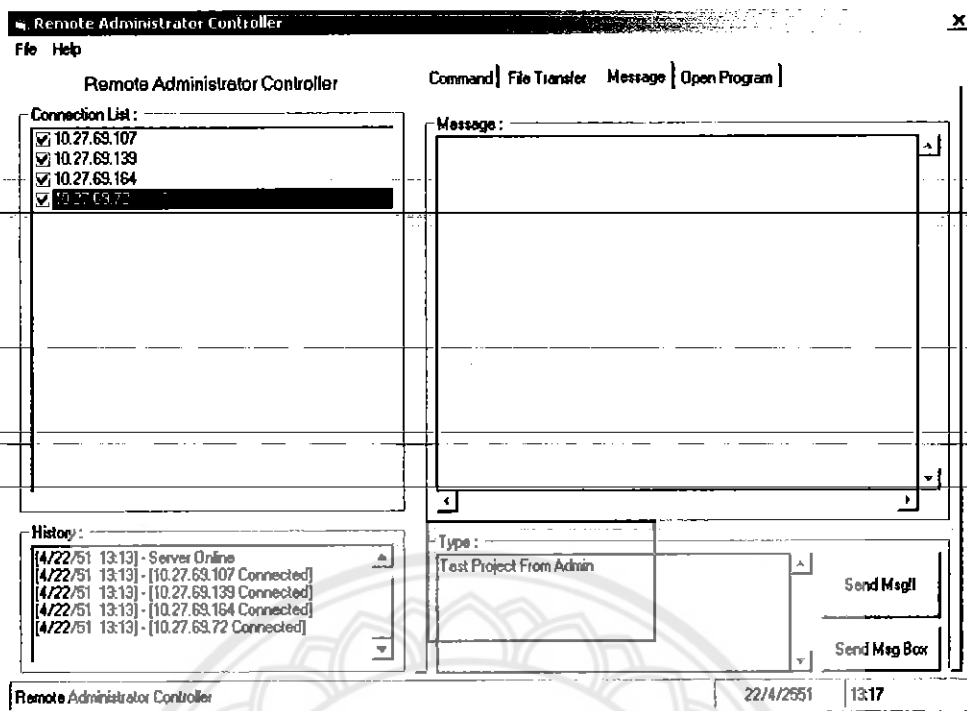
- เมื่อผู้ดูแลระบบกดปุ่มคำสั่ง Message โปรแกรมก็จะแสดงหน้าจอสำหรับส่ง Message ไปยังเครื่องลูกข่ายขึ้นมา ดังรูป 4.11
- ในส่วนของฟังก์ชัน Message จะมีหน้าจอหลักคือ Message, Type, Send Msg., Send Msg. Box ดังรูป 4.12
- ในส่วนของ Type คือช่องสำหรับพิมพ์ข้อความของผู้ดูแลระบบ ไปยังเครื่องลูกข่าย ดังรูป 4.13
- ในส่วนของ Message จะเป็นตัวบอกข้อความต่างๆ ที่ผู้ดูแลระบบติดต่อกับเครื่องลูกข่ายในระบบดังรูป 4.14
- คำสั่ง Send-Msg!! เป็นคำสั่งส่งข้อความที่ผู้ดูแลระบบต้องการส่งไปยังเครื่องลูกข่าย เมื่อผู้ดูแลระบบพิมพ์ข้อความในช่อง Type แล้ว
 - คำสั่ง Send Msg Box เป็นคำสั่งส่งข้อความเตือนต่างๆ ไปยังเครื่องลูกข่าย โดยจะขึ้นข้อความเป็น Message Box ดังรูป 4.15
 - เมื่อผู้ดูแลระบบส่ง Message ไปยังเครื่องลูกข่าย เครื่องลูกข่ายสามารถตอบกลับมายังเครื่องของผู้ดูแลระบบได้ ดังรูป 4.16



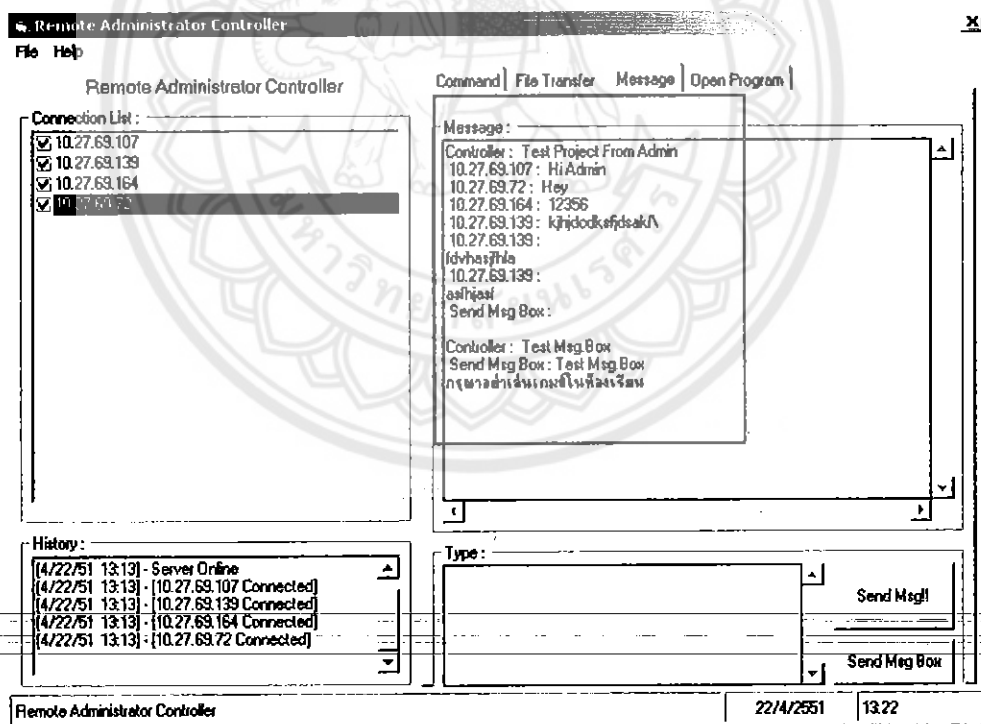
รูปที่ 4.11 รูปแสดงฟังก์ชันการส่ง Message ของผู้ดูแลระบบ



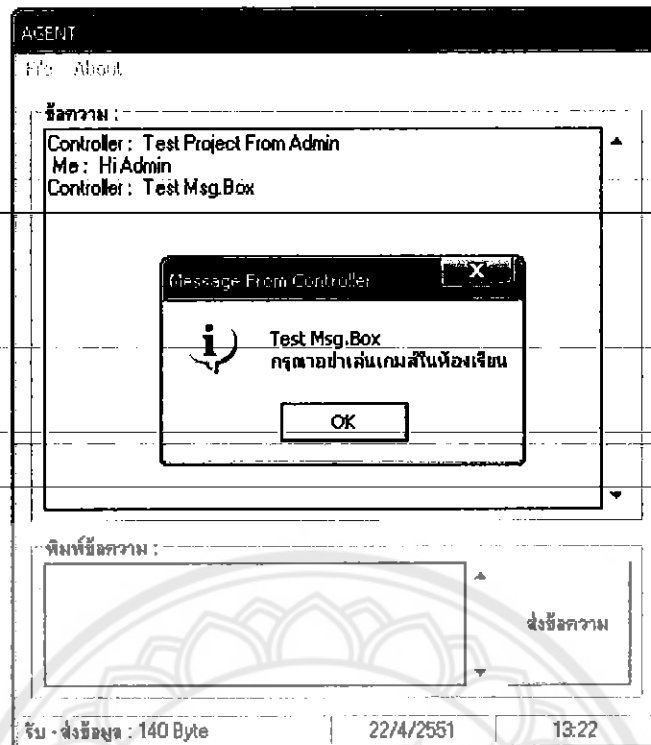
รูปที่ 4.12 รูปอธิบายฟังก์ชันการส่ง Message ของผู้ดูแลระบบ



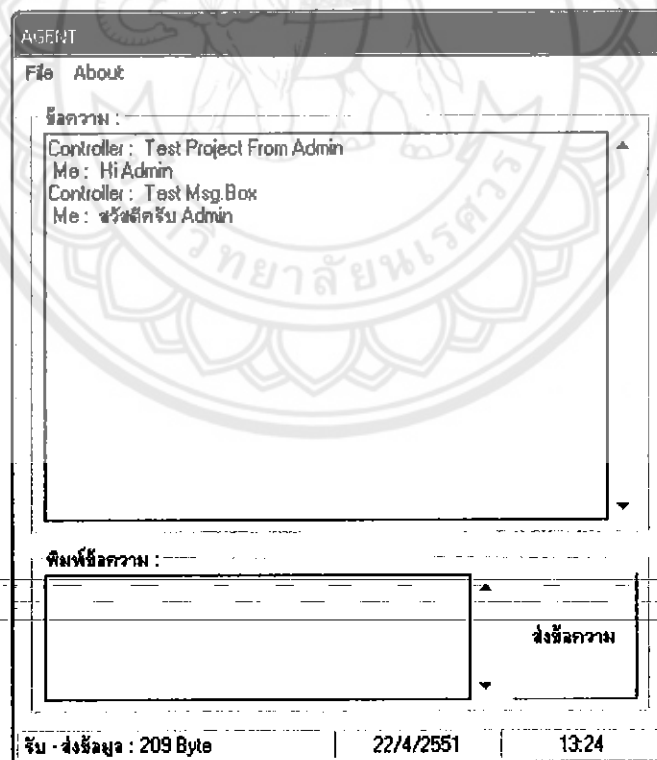
รูปที่ 4.13 รูปแสดงฟังก์ชันการส่ง Message ของผู้ดูแลระบบ



รูปที่ 4.14 รูปแสดงการส่ง Message ของผู้ดูแลระบบในช่องแสดง Message



รูปที่ 4.15 รูปแสดงฟังก์ชันการส่ง Message แบบ Msg Box ของผู้ดูแลระบบไปยังเครื่องลูกข่าย



รูปที่ 4.16 รูปแสดง Message ของเครื่องลูกข่ายติดต่อกับผู้ดูแลระบบ

4.2.4 การตั้งค่า Lock และ UnLock ไปยังเครื่องลูกข่าย

เป็นการสั่งระบบการใช้เครื่องลูกข่ายเครื่องดังกล่าวไม่ให้ใช้งานได้ชั่วคราว โดยเมื่อผู้ดูแลระบบสั่งฟังก์ชันนี้ไปยังเครื่องลูกข่ายใดๆ เครื่องลูกข่ายเครื่องดังกล่าวจะไม่สามารถใช้งานอะไรได้เลย จนกว่าผู้ดูแลระบบจะสั่งปลดล็อกเครื่องลูกข่ายเครื่องดังกล่าว

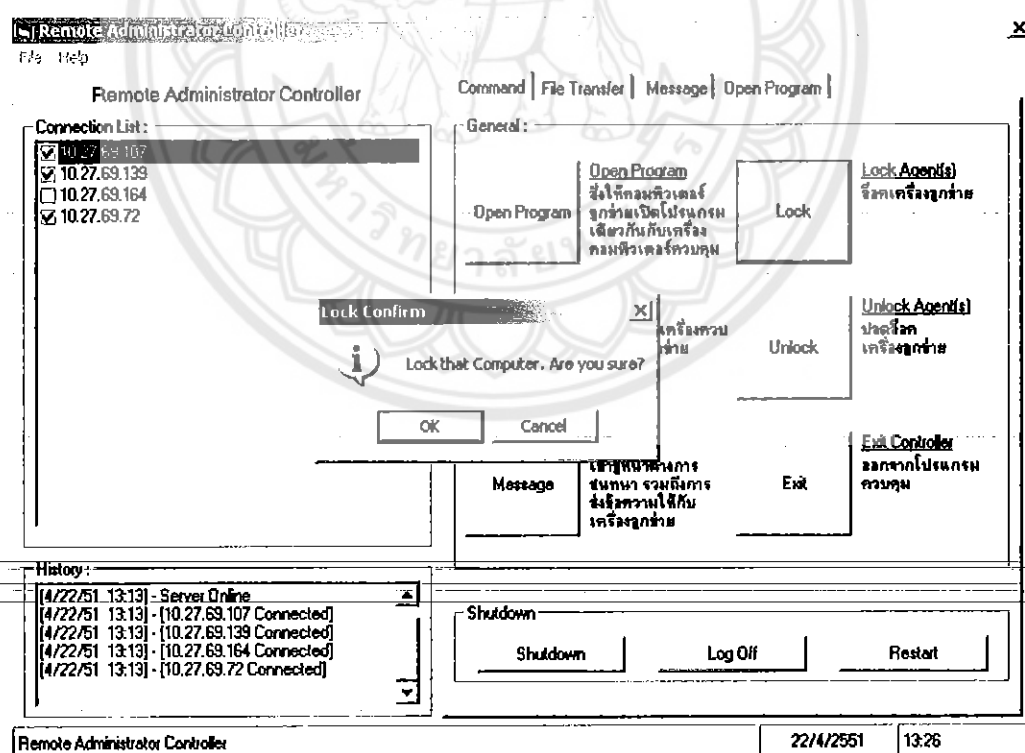
- เมื่อคลิกปุ่ม Lock เครื่องลูกข่ายเครื่องใดๆ จะมี Msg. Box ถามว่าต้องการทำการ ล็อกเครื่องลูกข่ายเครื่องดังกล่าวหรือไม่ ดังรูป 4.17

- เมื่อผู้ดูแลระบบคลิกปุ่ม OK เครื่องลูกข่ายเครื่องดังกล่าวจะถูกสั่งการใช้งานในทันที ดังรูป 4.18

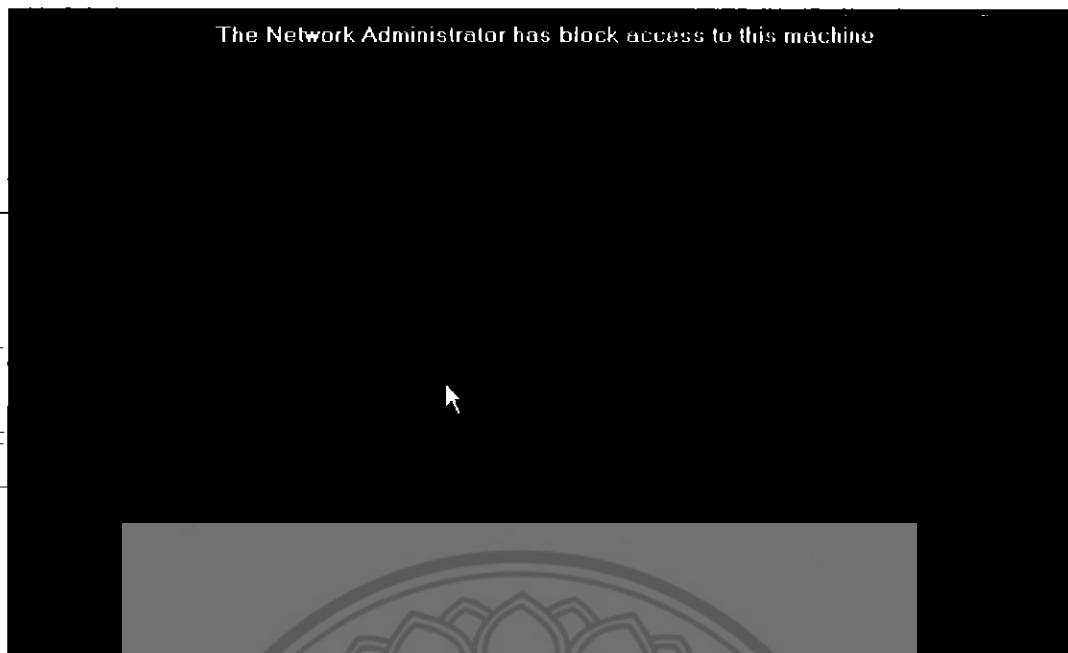
- แถบแสดง IP ของเครื่องลูกข่ายจะมีข้อความบอกว่าเครื่องลูกข่ายเครื่องนั้นๆ ถูก ล็อกการใช้งานไว้ ดังรูป 4.19

- เมื่อผู้ดูแลระบบคลิกปุ่ม Unlock จะมี Msg. Box ถามว่าต้องการจะ Unlock เครื่องลูกข่ายเครื่องนั้นๆ หรือไม่ ดังรูป 4.20

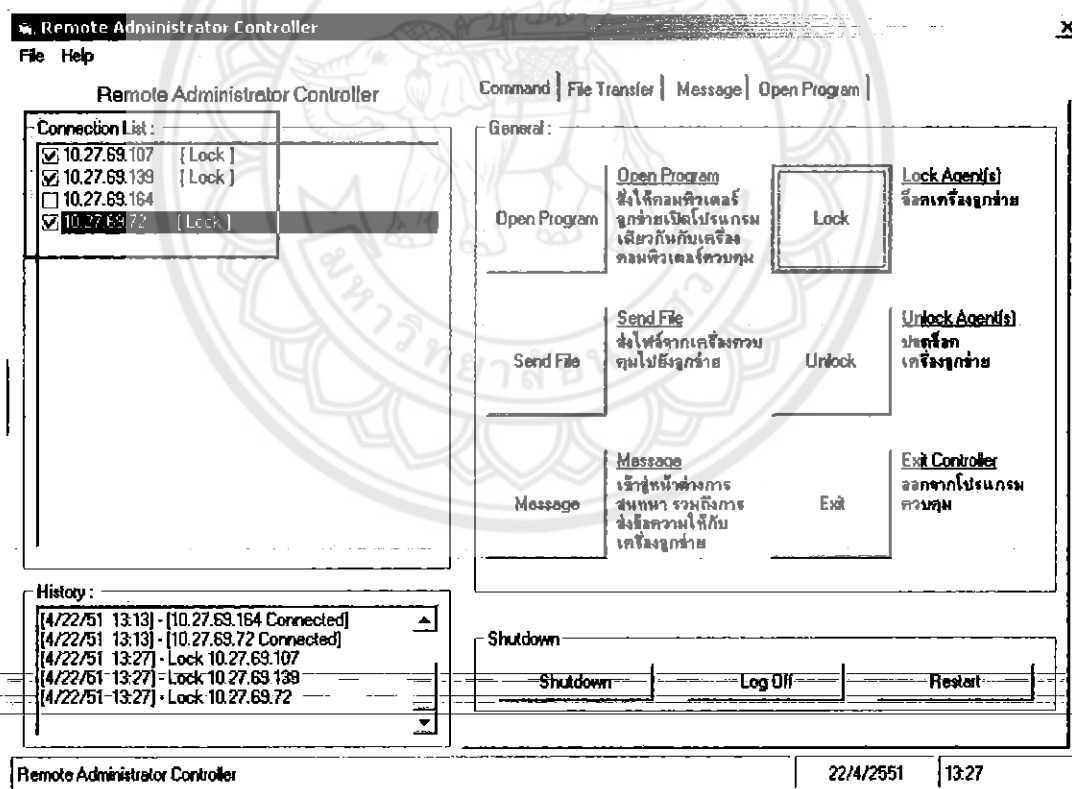
- เมื่อผู้ดูแลระบบคลิกปุ่ม OK เครื่องลูกข่ายที่ถูกล็อกไว้จะกลับมาใช้งานได้ตามปกติ แถบสถานะ IP จะบอกว่าเครื่องลูกข่ายเครื่องดังกล่าวได้ทำการ Unlock แล้ว ดังรูป 4.21



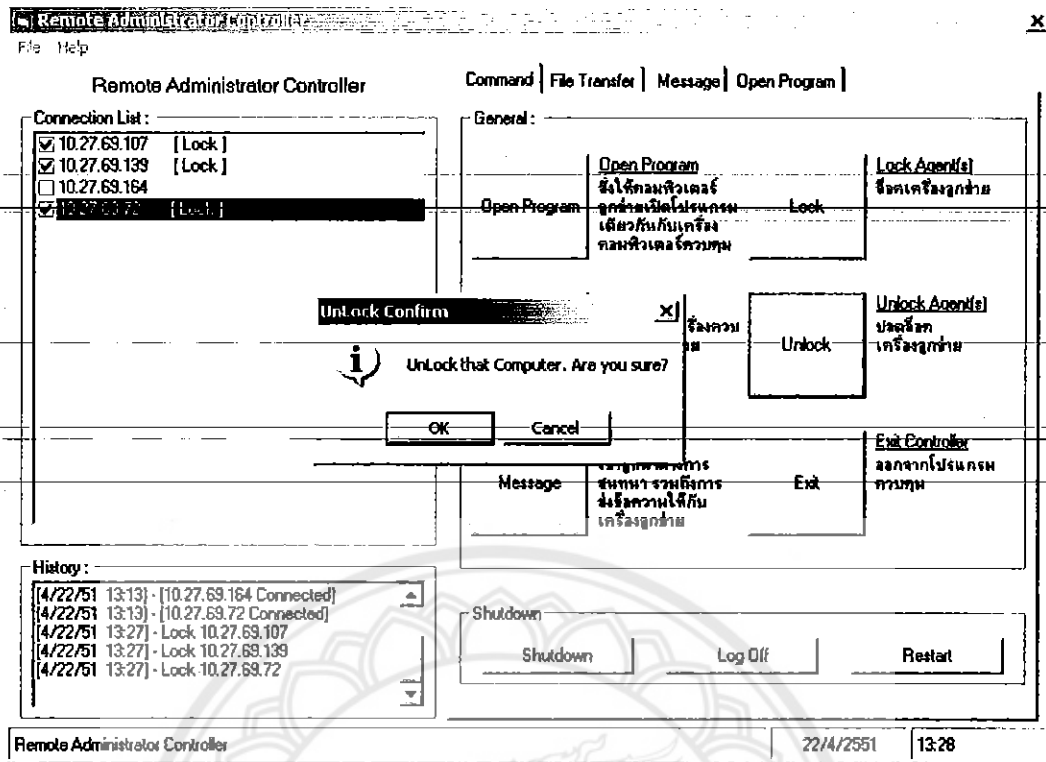
รูปที่ 4.17 รูปแสดงขั้นตอนการ Lock คอมพิวเตอร์เครื่องลูกข่าย



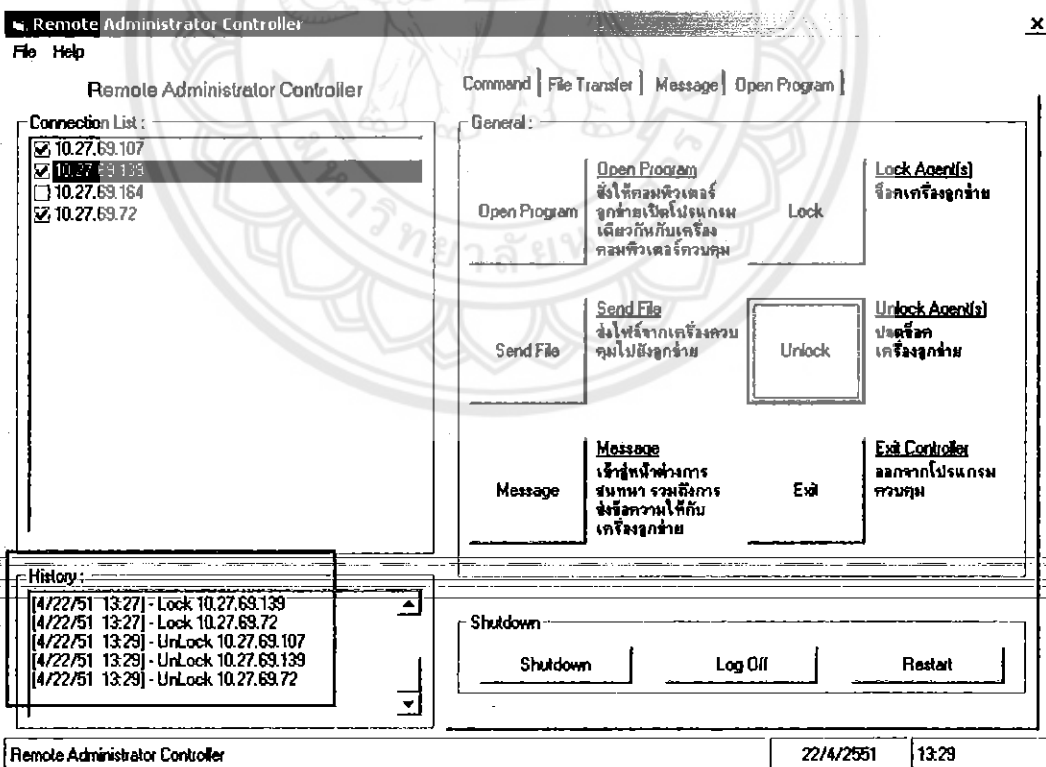
รูปที่ 4.18 รูปแสดงหน้าจอของเครื่องลูกข่ายที่ถูกตั้ง Lock



รูปที่ 4.19 รูปแสดงสถานะ IP ของเครื่องที่ถูก Lock



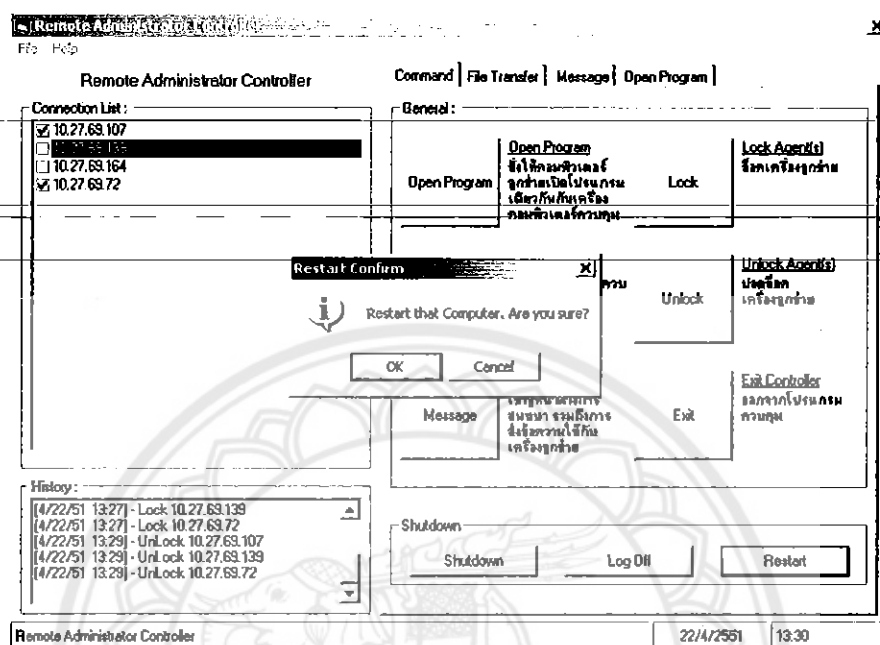
รูปที่ 4.20 รูปแสดงการ Unlock เครื่องลูกข่าย



รูปที่ 4.21 รูปแสดงการ Lock และ Unlock เครื่องลูกข่าย

4.2.5 การสั่งเครื่องลูกข่ายทำการรีสตาร์ท (Send Restart Command)

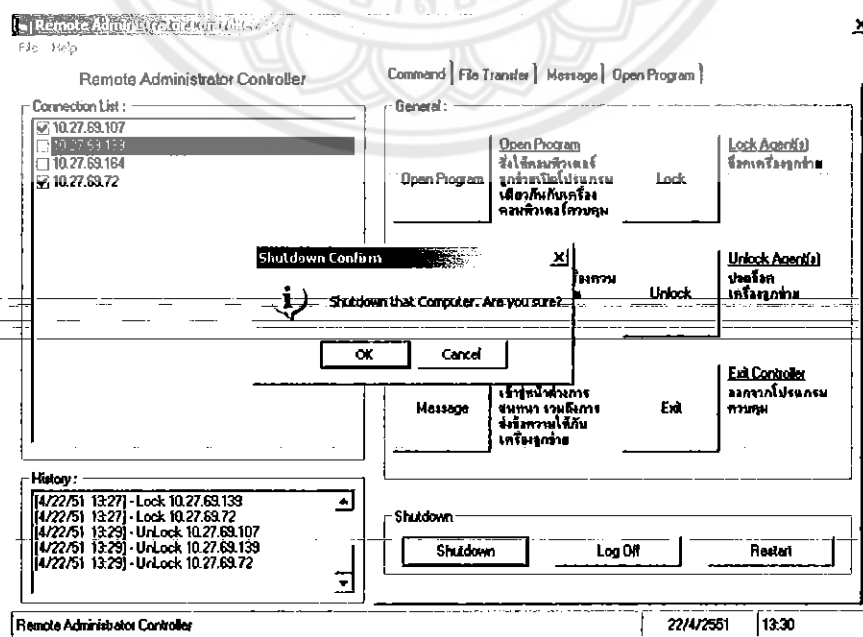
เป็นการสั่งให้เครื่องลูกข่ายกระทำการรีสตาร์ทเครื่องลูกข่ายนั้น โดยทำการขอคำยืนยันจากผู้ดูแลระบบก่อนที่จะส่งคำสั่งไปเครื่องลูกข่าย ดังรูป 4.22



รูปที่ 4.22 รูปแสดงการสั่งรีสตาร์ท

4.2.6 การสั่งเครื่องลูกข่ายทำการปิดเครื่อง (Send Shutdown Command)

คล้ายกระบวนการรีสตาร์ท แต่ผลของปฏิบัติการเป็นการปิดเครื่อง ดังรูป 4.23



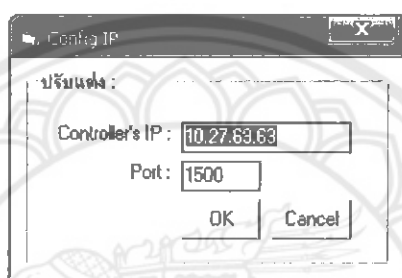
รูปที่ 4.23 รูปแสดงการสั่ง Shutdown

4.2.6 การสั่งเครื่องลูกข่ายทำการ Log Off

คล้ายกระบวนการรีสตาร์ทและปิดเครื่อง แต่ผลของปฏิบัติการเป็นการ Log Off

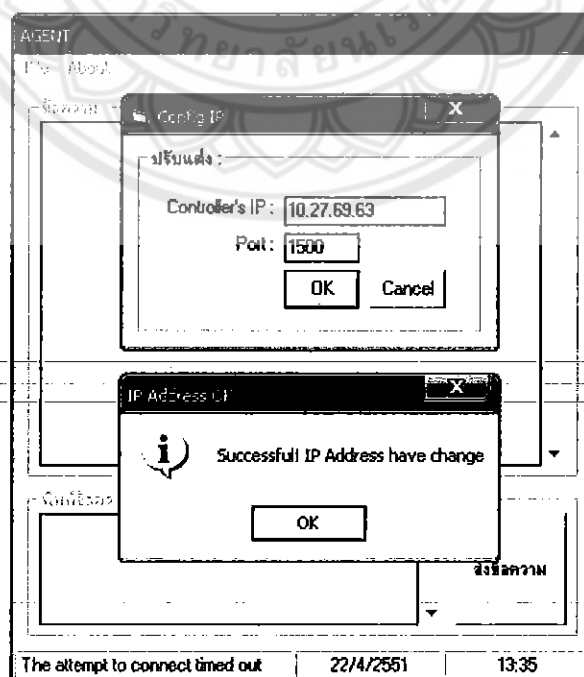
4.3 การทดสอบส่วนของการรับคำสั่งและปฏิบัติงาน (Agent)

เป็นการสังเกตการทำงานของส่วนการรับคำสั่งและปฏิบัติงาน โดยการทำงานเป็นการปฏิบัติตามคำสั่งที่ได้รับการร้องขอมาจากตัวควบคุม เพื่อส่งค่าสถานะที่จำเป็นกลับไปยังตัวควบคุม โปรแกรมจะทำงานและทำการเชื่อมต่อกับตัวควบคุมได้ ต้องมีการระบุค่า IP-Address ของตัวควบคุมเสียก่อน จึงจะสามารถทำการเชื่อมต่อกันและปฏิบัติตามคำสั่งร้องขอจากตัวควบคุมได้



รูปที่ 4.24 การกำหนดค่า IP Address ที่ใช้ในการติดต่อตัวควบคุม

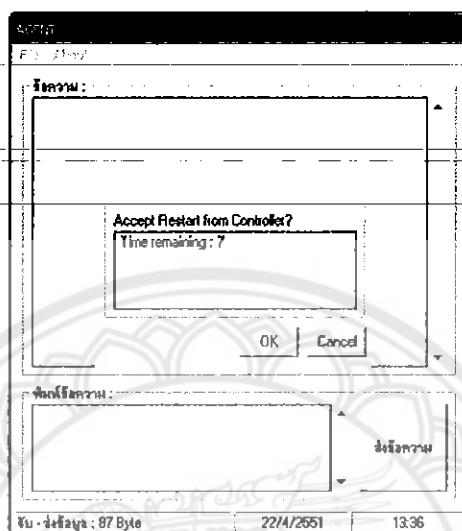
หลังจากการกำหนดค่า IP Address ของตัวควบคุมแล้ว จะเป็นการเชื่อมต่อกับตัวควบคุมอย่างสมบูรณ์ ดังรูป 4.25



รูปที่ 4.25 การเชื่อมต่อกับตัวควบคุม

4.3.1 การรับคำสั่งเพื่อการรีสตาร์ท (Restart)

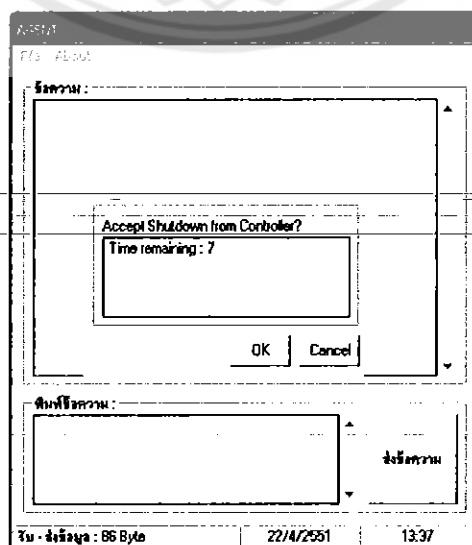
เป็นการปฏิบัติงานที่สั่งให้เครื่องทำการรีสตาร์ทภายหลังจากที่ตัวควบคุมเห็นว่าไม่มีกระบวนการทำงานบนเครื่องนั้นอีก ระบบจะทำการขอคำยอมรับจากเครื่องที่จะทำการรีสตาร์ท หากไม่มีกรตอบ ภายใน 10 วินาทีระบบจะทำการรีสตาร์ททันที ดังรูป 4.26



รูปที่ 4.26 การรอสัญญาณตอบรับจากเครื่องลูกข่าย สำหรับการรีสตาร์ท

4.3.2 การรับคำสั่งเพื่อการปิดเครื่อง (Shutdown)

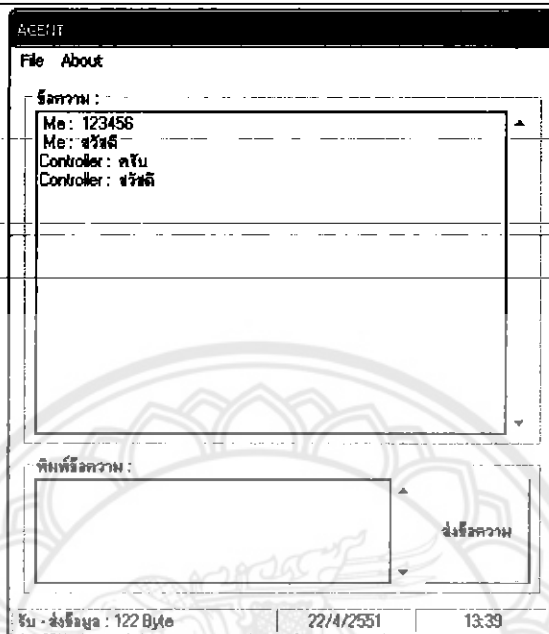
เป็นการปฏิบัติงานที่สั่งให้เครื่องทำการปิดเครื่องภายหลังจากที่ผู้ดูแลระบบเห็นว่าไม่มีกระบวนการทำงานบนเครื่องนั้นอีก เมื่อได้รับคำสั่งขอดังกล่าว ระบบจะทำการขอคำยอมรับจากเครื่องที่จะทำการปิดเครื่อง เป็นระบบการนับถอยหลัง หากไม่มีได้ตอบ โดยการเลือก “OK” หรือ “Cencle” ภายใน 10 วินาทีระบบจะทำการปิดตัวลง ดังรูป 4.27



รูปที่ 4.27 การรอสัญญาณตอบรับจากเครื่องลูกข่าย สำหรับการปิดเครื่อง

4.3.3 ระบบการสื่อสารภายในจากตัวควบคุม (Agent's Chat)

เป็นการสื่อสารจากผู้ใช้ถึงผู้ดูแลระบบ เมื่อผู้ใช้ต้องการติดต่อกับผู้ดูแลระบบ เพื่อที่จะสนทนาหรือแจ้งปัญหาต่าง ๆ

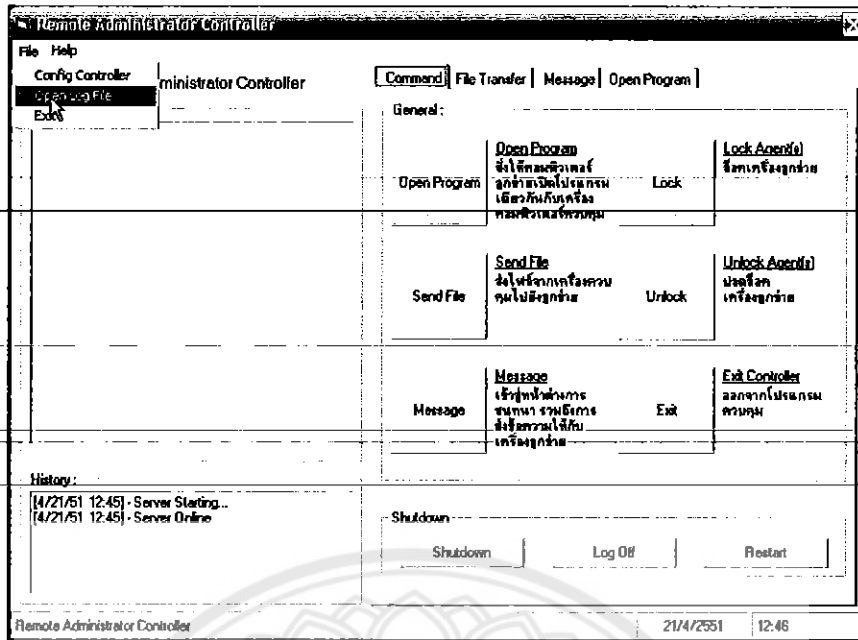


รูปที่ 4.28 การสื่อสารจากเครื่องลูกข่ายไปยังตัวควบคุม

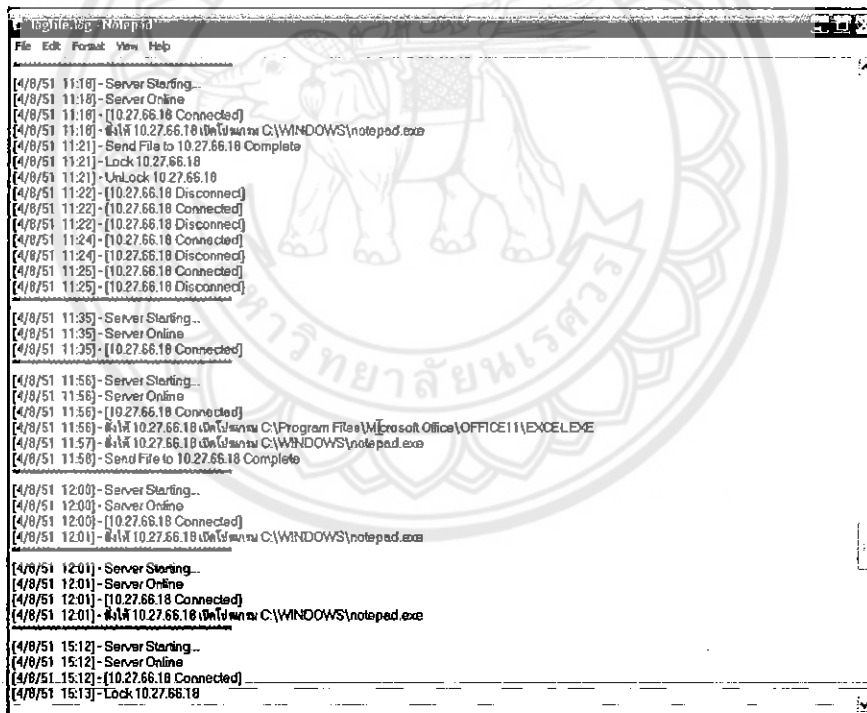
4.4 การทดสอบส่วนของการบันทึกประวัติการทำงานของโปรแกรม (Log File)

เป็นส่วนของการบันทึกการทำงานหรือคำสั่งต่างๆที่ผู้ดูแลระบบกระทำกับเครื่องลูกข่ายใดๆ โดยแสดงเป็นลักษณะของ Text File โดยบอก วันที่ เวลา และเหตุการณ์ที่เกิดขึ้นในระบบ

- เมื่อผู้ดูแลระบบต้องการดูเหตุการณ์ต่างๆ ที่เกิดขึ้นในระบบ สามารถคลิกที่ File > Open Log File ดังรูป 4.29
- โปรแกรมจะทำการเปิด Log File ขึ้นมา ดังรูป 4.30



รูปที่ 4.29 รูปแสดงขั้นตอนการเปิด Log File เหตุการณ์ที่เกิดขึ้นในระบบ



รูปที่ 4.30 รูปแสดง Log File ของระบบ

บทที่ 5

สรุปผลการดำเนินงานและข้อเสนอแนะ

เนื้อหาในบทนี้ กล่าวถึงการสรุปผลของการดำเนินงานที่ได้จากการศึกษาและการจัดทำโปรแกรมอำนวยความสะดวกในการจัดการเครื่องลูกข่ายสำหรับผู้ดูแลระบบ จนสามารถสร้างโปรแกรมที่ผู้พัฒนาคาดว่ามีประสิทธิภาพเพียงพอต่อการดูแลรักษาระบบเครือข่ายด้านการจัดการเกี่ยวกับคอมพิวเตอร์ ซึ่งจะช่วยให้ลดขั้นตอนการปฏิบัติงานและอำนวยความสะดวกให้กับผู้ดูแลระบบมากขึ้น

5.1 สรุปผลการดำเนินโครงการ

จากผลการดำเนินงาน จัดทำโปรแกรมอำนวยความสะดวกในการจัดการเครื่องลูกข่ายสำหรับผู้ดูแลระบบ ที่มีจุดประสงค์เพื่อช่วยให้ผู้ดูแลระบบสามารถดูแลรักษาระบบเครือข่ายทางด้านการจัดการคอมพิวเตอร์ ส่งผลให้ระบบเครือข่ายสามารถทำงานได้อย่างมีประสิทธิภาพ และยังช่วยลดขั้นตอนการปฏิบัติงานที่เคยจัดการทีละเครื่อง เปลี่ยนเป็นการจัดการหลาย ๆ เครื่องกับระบบเครือข่ายโดยผ่านทางหน้าจอของผู้ดูแลระบบเอง ทั้งนี้ยังลดปัญหาการถูกเจาะข้อมูลในระบบเครือข่าย เนื่องจากโปรแกรมสามารถทำการ Encrypt ข้อมูลที่รับส่งกันในระบบเครือข่ายได้โดยอัตโนมัติ

จากผลการดำเนินงานและการทดสอบพบว่า โปรแกรมอำนวยความสะดวกในการจัดการเครื่องลูกข่ายสำหรับผู้ดูแลระบบ สามารถทำงานได้ตรงตามวัตถุประสงค์ของโครงการ

5.2 ประโยชน์ที่ได้รับจากการทำโครงการ

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

5.3 ความสามารถของโปรแกรมอำนวยความสะดวกในการจัดการคอมพิวเตอร์เครื่องลูกข่ายสำหรับผู้ดูแลระบบ

1. สามารถสั่งการให้เครื่องลูกข่ายทำการรีสตาร์ทและชัตดาวน์เครื่องดังกล่าวได้
2. สามารถสั่งงานโปรแกรมให้ส่งไฟล์ข้อมูลไปยังเครื่องลูกข่ายเครื่องดังกล่าวได้
3. สามารถสั่งล็อกและปลดล็อกเครื่องลูกข่ายดังกล่าวได้
4. สามารถสั่งเปิดโปรแกรมต่างๆที่มีเหมือนกันระหว่างเครื่องของผู้ดูแลระบบและเครื่องลูกข่ายเครื่องดังกล่าวได้
5. สามารถส่งข้อความตอบรับระหว่างเครื่องของผู้ดูแลระบบและเครื่องลูกข่ายเครื่องดังกล่าวได้
6. สามารถแสดงประวัติการสั่งงานโปรแกรมระหว่างเครื่องของผู้ดูแลระบบและเครื่องลูกข่ายได้
7. สามารถทำการสั่งงานครั้งละหลาย ๆ เครื่องพร้อมกันได้

5.4 ข้อจำกัดของโปรแกรมอำนวยความสะดวกในการการคอมพิวเตอร์เครื่องลูกข่ายสำหรับผู้ดูแลระบบ

1. สามารถทำงานได้บนระบบปฏิบัติการ Windows XP เท่านั้น
2. จำเป็นต้องมีโปรแกรม Controller สำหรับเครื่องผู้ดูแลระบบ และมีโปรแกรม Agent บนเครื่องลูกข่าย
3. โปรแกรมไม่สามารถมองเห็นหน้าจอของเครื่องลูกข่ายได้
4. โปรแกรมอำนวยความสะดวกในการจัดการคอมพิวเตอร์เครื่องลูกข่ายสำหรับผู้ดูแลระบบนี้ สามารถรองรับเครื่องลูกข่ายเข้าสู่ระบบได้ไม่เกิน 255 เครื่อง

5.5 ข้อเปรียบเทียบและข้อแตกต่าง

1. โปรแกรมที่พัฒนาขึ้นมานี้มีหลักการทำงานคล้ายกับโปรแกรม Corporate Edition ในส่วนที่มีตัวควบคุมเป็นศูนย์กลางและลักษณะการปฏิบัติงาน แต่โปรแกรมที่พัฒนาขึ้นมานี้มีข้อดีในหลายส่วน เช่น ค่าใช้จ่ายในการใช้งานน้อย เพราะเป็น โปรแกรมที่พัฒนาขึ้นมาเอง และแม้ว่าจะเลือกใช้โปรแกรมดูแลระบบที่ไม่ต้องเสียค่าลิขสิทธิ์แต่ประสิทธิภาพโดยรวมของโปรแกรมใกล้เคียงกับโปรแกรมที่มีขายอยู่ทั่วไป
2. โปรแกรมที่พัฒนาขึ้นมาจะมีหลักการสั่งงานคล้ายกับโปรแกรม PC Anywhere แตกต่างกันที่โปรแกรม PC Anywhere มีการควบคุมการทำงานของเครื่องลูกข่ายได้เต็มประสิทธิภาพ สามารถเข้าถึงข้อมูลของเครื่องลูกข่าย เสมือนว่านั่งอยู่หน้าเครื่องนั้น แต่โปรแกรมที่

พัฒนาขึ้นเป็น โปรแกรมแฝงที่รองรับคำสั่งจากตัวควบคุมและปฏิบัติตามคำสั่งของมัน โดยไม่สามารถเข้าควบคุมการทำงานของเครื่องลูกข่ายได้มากนัก ทั้งนี้โปรแกรม PC Anywhere เข้าถึงและควบคุมเครื่องลูกข่ายได้ครั้งละหนึ่งเครื่องเท่านั้น ซึ่งเหมือนกระบวนการทำงานที่ละเครื่องเพียงแต่ไม่ต้องเดินไปสั่งงาน ในขณะที่โปรแกรมที่พัฒนาขึ้น สามารถส่งการให้เครื่องลูกข่ายปฏิบัติงานพร้อมกันได้ครั้งละหลาย ๆ เครื่อง ทำให้ลดขั้นตอนการทำงานไปได้มาก

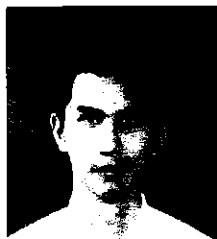
5.6 แนวทางในการพัฒนาโปรแกรมในอนาคต

1. พัฒนาโครงสร้างของ โปรแกรมอำนวยความสะดวกในการจัดการคอมพิวเตอร์เครื่องลูกข่ายสำหรับผู้ดูแลระบบให้สามารถใช้งานร่วมกับระบบปฏิบัติการอื่นๆ ได้
2. พัฒนาโปรแกรมให้สามารถควบคุมการสั่งงานได้ดีขึ้น เช่น สามารถมองเห็นหน้าจอของเครื่องลูกข่ายได้จากเครื่องของผู้ดูแลระบบเอง สามารถแจ้งสถานะการทำงานได้ละเอียดมากขึ้น เป็นต้น
3. สามารถปรับแต่งค่าต่าง ๆ ของโปรแกรมได้ละเอียดมากขึ้น
4. พัฒนาโปรแกรมให้สามารถ Install โปรแกรมต่างๆ ลงในเครื่องลูกข่าย โดยสั่งงานจากเครื่องของผู้ดูแลระบบ

บรรณานุกรม

- [1] กิตติ ภัคคีวัฒนนกุล , จำลอง ทรูอดุทธาหะ. **Visual Basic 6 ฉบับโปรแกรมเมอร์**. กรุงเทพฯ: ไทย
เจริญการพิมพ์. 2542.
- [2] พ.อ. เจนวิทย์ เหลืองอร่าม , ปิยวิทย์ เหลืองอร่าม. **การเขียนโปรแกรมสำหรับ Client/Server
ด้วย Visual Basic 6 และ ASP, VBScript, Access, SQLServer**. กรุงเทพฯ: บริษัท ชรรมสาร
จำกัด
- [3] บริษัท อับเปอร์ แมเนจเม้นต์ เอ็กซ์เซลเลนซ์ จำกัด. **เทคนิคและการประยุกต์ Visual Basic 6.0
Win32 API**. กรุงเทพฯ : บริษัท ฟีด สไตล์ จำกัด
- [4] สัจจะ จรัสรุ่งระวีวร. **คู่มือพัฒนาแอปพลิเคชันด้วย Visual Basic 6.0**. กรุงเทพฯ : ด้านสุทธการ
พิมพ์
- [5] สัจจะ จรัสรุ่งระวีวร. **คู่มือการเขียนโปรแกรมและการใช้งาน Visual Basic 6.0**. กรุงเทพฯ :
ด้านสุทธการพิมพ์
- [6] **บทเรียนผ่านเว็บไซต์ความรู้เบื้องต้นเกี่ยวกับระบบรหัส Cryptography [Online]**. จุฬาลงกรณ์
มหาวิทยาลัย : <http://progmeth.eng.chula.ac.th/ReviseLab/index.html>.
- [7] วิทยา เรื่องพรวิสุทธิ. **เรียนรู้อินทราเน็ต ระบบเครือข่ายองค์กรยุคใหม่**. กรุงเทพฯ : ซีเอ็ด
ยูเคชั่น, 2542.
- [8] **บทเรียนผ่านเว็บไซต์ความรู้เบื้องต้นเกี่ยวกับเครือข่ายคอมพิวเตอร์**. [Online]. Available:
<http://www.cybered.co.th/warnuts/wbi/wbi3/web/page14.htm>

ประวัติผู้เขียนโครงการ



ชื่อ นายจิรวัดน์ พร้อมมิตร
 ภูมิลำเนา 43/1 หมู่ 1 ต.บ้านตึก อ.ศรีสัชนาลัย จ. สุโขทัย
 64130

ประวัติการศึกษา

- จบระดับมัธยมศึกษาตอนปลายจากโรงเรียนอุดรดิตถ์
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4
 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยนเรศวร

E-mail Lp_pang@hotmail.com



ชื่อ นางสาวปิยชลิมา อรุณมาศ
 ภูมิลำเนา 24/80 ถ.สระหลวง ต.ในเมือง อ.เมือง จ.พิจิตร
 66000

ประวัติการศึกษา

- จบระดับมัธยมศึกษาตอนปลายจากโรงเรียนสระหลวง
 พิจิตร
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4
 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
 มหาวิทยาลัยนเรศวร

E-mail Chobits_amp@hotmail.com