โปรแกรมสำรวจเครือข่ายและระบบคอมพิวเตอร์

NETWORK AND COMPUTER SYSTEM VIEWER

นายสรัญ ธุวโชคชัย

นายพงษ์สวัสดิ์ พฤกษ์เอก

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2548

โปรแกรมสำรวจเครือข่ายและระบบคอมพิวเตอร์

NETWORK AND COMPUTER SYSTEM VIEWER

โดย

นายสรัญ ธุวโชคชัย

นายพงษ์สวัสดิ์ พฤกษ์เอก

อาจารย์ที่ปรึกษา

อ. ธนา หงษ์สุวรรณ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ.2548

ปริญญานิพนธ์ปีการศึกษา 2548

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โปรแกรมสำรวจเครือข่ายและระบบคอมพิวเตอร์

NETWORK AND COMPUTER SYSTEM VIEWER

ผู้จัดทำ

1. นายสรัญ ธุวโชคชัย รหัสนักศึกษา 45010482

2. นายพงษ์สวัสดิ์ พฤกษ์เอก รหัสนักศึกษา 44010490

อาจารย์ที่ปรึกษา

(อาจารย์ ธนา หงษ์สุวรรณ)

โปรแกรมสำรวจเครือข่ายและระบบคอมพิวเตอร์

นายสรัญ ธุวโชคชัย	45010482
นายพงษ์สวัสดิ์ พฤกษ์เอก	45010490
อ.ธนา หงษ์สุวรรณ	อาจารย์ที่ปรึกษา
ปีการศึกษา 2548	

บทคัดย่อ

ในปัจจุบันนี้มีการใช้งานเครือข่ายการสื่อสารข้อมูลภายในองค์กรต่างๆอย่างแพร่หลายทั้งในด้านการสื่อสารข้อมูลภายในองค์กร และการสื่อสารข้อมูลระหว่างองค์กร การใช้เครือข่ายการสื่อสารข้อมูลจึงจำเป็นที่จะต้องมีประสิทธิภาพในการทำงานที่สูงเพื่อเป็นการสนับสนุนระบบการทำงานขององค์กร

โครงงานนี้เป็นการนำเสนอโครงงานพัฒนาโปรแกรมสำรวจเครือข่ายและระบบคอมพิวเตอร์ (Network and Computer System Viewer) ซึ่งเป็นโปรแกรมที่ใช้สำหรับผู้ดูแลระบบเครือข่ายเพื่ออำนวยความสะดวกในการแสดงให้เห็นสถานะปัจจุบันของระบบเครือข่าย และสภาพของระบบคอมพิวเตอร์ที่ใช้งานระบบเครือข่าย โดยนำเสนอเทคนิคต่างๆ ที่ใช้ในการรวบรวมข้อมูลของระบบคอมพิวเตอร์บนเครือข่าย, ส่วนที่ใช้ในการจัดเก็บข้อมูล และส่วนที่ใช้ในการจัดการเพื่อนำมาแสดงผลบนโปรแกรม

รวมทั้งมีรายละเอียดในการพัฒนาโปรแกรมเพื่อให้ผู้ที่สนใจสามารถนำไปเป็นแนวทางในการพัฒนาเครื่องมือที่ใช้อำนวยความสะดวกในการจัดการเครือข่ายอื่นๆ เพื่อลดการนำเข้าโปรแกรมต่างๆจากต่างประเทศ

NETWORK AND COMPUTER SYSTEM VIEWER

Sarun Tuvachokchai	45010482
Pongswasdi Pruek_ek	45010490
Thana Hongsuwan	Advisor
Academic Year 2005	

ABSTRACT

Presently, Communication Network is applied widely in many organizations include both internal communication and external communication. Therefore, Communication Network must have high performance to support the organization work.

This project describes the network and computer system viewer that is the administrator tool program for network monitoring . Including describes the use of enumeration technique for gathering network information, storing database and information processing to display. Addition developing details for developer.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้คงไม่สำเร็จลุล่วงไปด้วยดี หากปราศจากคำแนะนำและการให้คำปรึกษาจาก อาจารย์ ธนา หงษ์สุวรรณ ซึ่งเป็นอาจารย์ที่ปรึกษาวิทยานิพนธ์ และ อาจารย์ ธนัญชัย ตรีภาค ที่ให้ความเอาใจใส่ แนะนำ และช่วยเหลือเสมอมา ผู้วิจัยรู้สึกซาบซึ้งในความอนุเคราะห์จากท่านและขอขอบคุณเป็นอย่างสูง

ขอขอบคุณพี่ๆและเพื่อนๆห้องปฏิบัติการNETWORK และ ภาควิชาวิศวกรรมคอมพิวเตอร์ทุกคนที่ให้ความช่วยเหลือในเรื่องต่างๆ จนสำเร็จไปได้ด้วยดี

ขอขอบคุณตัวข้าพเจ้าทั้งสองที่ไม่ท้อแท้ไปเสียก่อนที่จะประสบความสำเร็จ ขอบคุณที่มีชีวิตมาถึงทุกวันนี้ และโอกาสดีๆ ที่ได้รับมา

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจและให้การสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

นายสรัญ ธุวโชคชัย

นายพงษ์สวัสดิ์ พฤกษ์เอก

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

สารบัญรูป

สารบัญรูป (ต่อ)

รูปที่ หน้า

สารบัญรูป (ต่อ)

รูปที่ หน้า

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มาของโครงงาน

ในปัจจุบันนี้การใช้งานระบบเครือข่ายคอมพิวเตอร์กำลังเป็นที่นิยมอย่างแพร่หลาย ทำให้จำนวนผู้ใช้ในระบบเพิ่มขึ้นอย่างรวดเร็วและมีผลให้ระบบเครือข่ายคอมพิวเตอร์มีขนาดใหญ่และมีความซับซ้อนมากขึ้น การดูแลระบบเครือข่ายจึงทำได้ยากขึ้นตามไปด้วย โดยในการบริหารระบบเครือข่ายจำเป็นที่จะต้องทราบรายละเอียดข้อมูลต่างๆของระบบเครือข่ายคอมพิวเตอร์ ทั้งในรูปแบบของปริมาณความหนาแน่นของข้อมูลที่ใช้งานในระบบเครือข่าย จำนวนคอมพิวเตอร์ที่เปิดใช้งานอยู่ ลักษณะการใช้งานของคอมพิวเตอร์ในเครือข่าย เพื่อนำข้อมูลเหล่านั้นมารวบรวมเป็นรายงานทางสถิติ และสามารถนำมาทำการวิเคราะห์เพื่อแก้ปัญหาที่เกิดขึ้นในระบบเครือข่ายได้

โครงงานนี้จึงมุ่งเน้นที่จะพัฒนาโปรแกรมสำรวจเครือข่าย และระบบคอมพิวเตอร์ ซึ่งสามารถสำรวจข้อมูลรายละเอียดต่างๆของระบบเครือข่ายคอมพิวเตอร์ และนำมารวบรวมเป็นรายงานทางสถิติในรูปแบบต่างๆ เช่น กราฟ แผนภูมิ ตาราง เพื่อนำไปใช้ในการวิเคราะห์ และบริหารจัดการเครือข่ายได้อย่างมีประสิทธิภาพ

1.2 วัตถุประสงค์ของโครงงาน

1. เพื่อพัฒนาต้นแบบของโปรแกรมที่ใช้ในสำรวจข้อมูลรายละเอียดต่างๆของระบบเครือข่ายคอมพิวเตอร์ให้สามารถอำนวยความสะดวกแก่ผู้ดูแลระบบ
2. เพื่อพัฒนาโปรแกรมที่ใช้ในการอำนวยความสะดวกในการศึกษา บริหารจัดการและตรวจสอบระบบเครือข่ายคอมพิวเตอร์
3. เพื่อนำไปใช้ทดแทนการนำเข้าโปรแกรมจากต่างประเทศ

1.3 ขอบเขตของโครงงาน

1. สามารถตรวจหาคอมพิวเตอร์ที่เปิดใช้งานเครือข่ายอยู่ในขณะนั้นได้
2. สามารถอ่านค่า SNMP ที่สนใจในการบอกลักษณะ และปริมาณความหนาแน่นของข้อมูลที่ใช้งานในระบบเครือข่ายได้
3. สามารถตรวจหาระบบปฏิบัติการคอมพิวเตอร์ที่เปิดใช้งานเครือข่ายอยู่ได้

4. สามารถดูบริการ หรือพอร์ตพื้นฐานของเครื่องปลายทางที่สนใจได้
5. สามารถตรวจสอบการเปิดใช้บริการที่ไม่เหมาะสมของเครื่องที่อยู่ในเครือข่ายได้ เช่น เครื่องติดโทรจัน โค้ดหนอน การเล่นเกมออนไลน์ได้
6. สามารถดูปริมาณความหนาแน่นของข้อมูลที่ใช้งานโดยรวมที่เข้าและออกจากเครือข่ายได้
7. สามารถนำข้อมูลโดยรวมของระบบเครือข่ายมาแสดงผลได้ในรูปของรายงาน กราฟและแผนภูมิได้
8. สามารถนำผลลัพธ์มาแสดงผลออกเป็นรายงานปริมาณความหนาแน่นของข้อมูลที่ใช้งานโดยรวม และคอมพิวเตอร์ที่เปิดใช้งานเครือข่ายอยู่ในขณะนั้นได้

1.4 วิธีการดำเนินการ

1. ศึกษาทฤษฎีที่เกี่ยวข้องกับโครงงาน
2. ศึกษาความเป็นไปได้ของโครงงาน
3. กำหนดขอบเขตของโครงงาน
4. ศึกษาเครื่องมือและภาษาที่ใช้เขียนโปรแกรม
5. ศึกษา source code ของโปรแกรมย่อยที่หามาได้ และนำมาประยุกต์ใช้
6. ออกแบบโครงสร้างของโปรแกรมหลัก และโปรแกรมย่อย
7. เขียนโปรแกรมย่อยในส่วนต่างๆ
8. เขียนและรวบรวมโปรแกรมย่อยตามที่ได้ออกแบบ
9. ทดสอบการทำงานของโปรแกรมและประเมินจุดแก้ไข
10. แก้ไขโปรแกรมในส่วนที่มีปัญหา
11. วิเคราะห์ผลการทดลอง และสรุป
12. จัดทำคู่มือการติดตั้งและใช้งาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. โปรแกรมต้นแบบเพื่อใช้ในสำรวจข้อมูลรายละเอียดต่างๆของระบบเครือข่ายคอมพิวเตอร์
2. ได้รับความรู้จากการศึกษาวิธีการตรวจสอบข้อมูลบนระบบเครือข่าย
3. ได้รับความรู้จากการศึกษา เกี่ยวกับเรื่องความปลอดภัย การเดินทางของข้อมูลในระบบเครือข่าย และการเขียนโปรแกรมทางด้านเครือข่าย
4. ช่วยลดการนำเข้าโปรแกรมที่ใช้ในการดูแลระบบเครือข่ายจากต่างประเทศ

1.6 ส่วนประกอบของปริญญานิพนธ์

ปริญญานิพนธ์ฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

บทที่ 1 กล่าวถึงความสำคัญและที่มาของโครงงาน วัตถุประสงค์ของโครงงาน ขอบเขตของโครงงาน วิธีการดำเนินการ ประโยชน์ที่คาดว่าจะได้รับ และส่วนประกอบของปริญญานิพนธ์

บทที่ 2 กล่าวถึงทฤษฎีที่ใช้ในโครงงาน ซึ่งประกอบด้วยทฤษฎีอะไรบ้าง รายละเอียดต่างๆของทฤษฎีที่เกี่ยวข้อง และได้ศึกษามา

บทที่ 3 กล่าวถึงชิ้นงานของโครงงานนี้ อธิบายการออกแบบ และพัฒนาโปรแกรม รายละเอียดของโครงสร้าง และการทำงานของโปรแกรม

บทที่ 4 กล่าวถึงการทดลอง และผลการทดลองการทำงานของโปรแกรม อธิบายการทดลองในส่วนต่างๆที่ได้ศึกษาทดลองมา

บทที่ 5 กล่าวถึงการวิเคราะห์ผลการทดลองและบทสรุปของโครงงาน ปัญหาอุปสรรคและแนวทางในการแก้ไข ข้อจำกัดของโปรแกรม แนวทางการพัฒนาต่อวิจารณ์สิ่งที่ได้รับจากโครงงาน และข้อเสนอแนะสำหรับเป็นแนวทางในการพัฒนาต่อ

บทที่ 2
ทฤษฎีพื้นฐานที่ใช้ในโครงงาน

2.1 ทฤษฎีการสแกนเพื่อตรวจสอบ

2.1.1 Network Scan : ICMP Ping Sweeps

หนึ่งในเทคนิคพื้นฐานของ ICMP Network Scan ที่เป็นที่นิยมก็คือ การ ping ไปยังเครื่องเป้าหมายจำนวนมากพร้อมๆกัน ในลักษณะคล้ายกับการกวาดหรือกราดยิง ซึ่งเราเรียกว่าการทำ ping sweep เพื่อตรวจสอบว่าเครื่องปลายทางใดบ้างที่ยังเปิดทำงานอยู่ โดยปกติหากใช้คำสั่ง ping ธรรมดาจะมีการส่ง แพ็กเก็ต ICMP ECHO (Type 8) ออกไปยังเครื่องปลายทางและรอคอย ICMP ECHO_REPLY (Type 0) ที่จะถูกส่งกลับมา ซึ่ง ping จะมีประโยชน์สำหรับการทดสอบว่าเครื่องปลายทางเปิดอยู่หรือไม่ โดยเทคนิคในการ ping sweep มีหลายเทคนิคแตกต่างกันไป เช่น การ ping ในลักษณะรอคอยการตอบสนองจากเครื่องทีละเครื่อง ก่อนจะเปลี่ยนไปทดสอบเครื่องอื่นๆ ถัดไปหรือจะเปลี่ยนเป็นการ ping ออกไปพร้อมๆกันในแบบขนานไปยังเครื่องปลายทางหลายๆ เครื่อง ในลักษณะคล้าย Round Robin คือ การ ping ไปที่เครื่อง 1,2,3,....,n ถึงเครื่องสุดท้าย แล้ววนกลับมาส่งแพ็กเก็ตไปที่เครื่อง 1,2,3 ใหม่ไปเรื่อยๆ แล้ววนกลับมาอีก โดยไม่จะเป็นต้องหยุดรอจากตอบสนองจากเครื่องแรก ดังนั้น จะทำงานได้รวดเร็วกว่าคำสั่ง ping ธรรมดามาก นอกจากนี้ยังมีเทคนิคอื่นในการที่จะตรวจสอบว่าเครื่องปลายทางยังเปิดทำงานอยู่หรือไม่โดยการส่งแพ็กเก็ตของโปรโตคอล ICMP ใน Type อื่นๆด้วย เช่น การส่ง ICMP Type TIME STAMP request (13) เพื่อสอบถามเวลาของเครื่องปลายทาง (เพื่อดูว่าเครื่องๆนั้นอยู่ในโซนเวลา (time zone) แถบใด) แล้วรอคอย ICMP Type TIME STAMP Reply (14) หรือการส่ง ICMP INFO request (15) แล้วรอคอย ICMP Type INFO Reply (16) กลับมา

กล่าวโดยสรุป ICMP NETWORK SCAN นี้จะทำให้เราสามารถตรวจสอบได้ว่าเครื่องปลายทางใดบ้างที่เราสามารถติดต่อได้โดยตรง ด้วยการส่งและรอรับแพ็กเก็ต ICMP ใน TYPE ต่างๆ แต่วิธีการนี้จะเหมาะสำหรับเครื่องที่อยู่บนเน็ตเวิร์คขนาดเล็กถึงขนาดกลางเท่านั้น มันจะไม่มีประสิทธิภาพเพียงพอที่จะนำมาใช้ตรวจสอบเครื่องที่อยู่บนเน็ตเวิร์คขนาดใหญ่ได้ เนื่องจากการตรวจสอบเครื่องที่อยู่ในเน็ตเวิร์คในองค์กรขนาดใหญ่ อาจกินเวลานานหลายชั่วโมงกว่าจะทราบผลและถึงแม้จะอยู่บนเน็ตเวิร์คขนาดเล็กหรือขนาดกลางก็ตามแต่หากที่เน็ตเวิร์คเป้าหมายได้มีการบล็อกแพ็กเก็ต ICMP ไม่ว่าจะเป็นการบล็อคที่เราเตอร์ของเน็ตเวิร์คปลายทางหรือการใช้โปรแกรม Personal ไฟร์วอลล์ที่เครื่องปลายทางก็ตามวิธีการนี้ก็จะหมดประสิทธิภาพทันที รวมทั้งในปัจจุบันวิธีการ Ping Sweep ถูกมองเป็นการโจมตีในรูปแบบของ DoS (Denial of

Service) เนื่องจาก Worm บางชนิดจะใช้วิธีการนี้ในการรบกวนแบนวิดของเน็ตเวิร์คเป้าหมายด้วย จึงทำให้วิธีการนี้ไม่เหมาะสมในการตรวจสอบเน็ตเวิร์คในปัจจุบัน

2.1.2 Network Scan : Port Scanning

จากเหตุผลที่กล่าวมาทำให้จำเป็นต้องหาเทคนิคหรือเครื่องมืออื่นที่ใช้ตรวจสอบได้ว่าเครื่องปลายทางใดบ้างที่เข้าถึงได้และเปิดทำงานอยู่ แต่อย่างไรก็ตาม มันอาจไม่ถูกต้องและรวดเร็วเท่ากับตรวจสอบด้วย ping sweep เทคนิคที่ว่านั้นก็คือ การสแกนพอร์ตนั่นเอง ซึ่งเป็นเทคนิคที่ใช้ถ้าเครื่องหรือเน็ตเวิร์คหรือปลายทางบล็อกแพ็กเก็ต ICMP ไว้ โดยการสแกนพอร์ตทั่วๆไปหรือสแกน

*พอร์ตสามัญบนแต่ละเครื่องเราจะสามารถคาดการณ์ได้ว่าเครื่องไหนเปิดอยู่บ้างด้วยการ*ส่ง TCP Packet หรือ UDP Packet ในรูปแบบต่างๆเพื่อคอนเน็กเข้าไปที่ TCP หรือ UDP port ของเครื่องปลายทางซึ่งทำให้สามารถตรวจสอบได้ว่าเครื่องปลายทางใดเชื่อมต่อกับเน็ตเวิร์คและเปิดทำงานอยู่ *ยกตัวอย่างเช่นพอร์ต* 80 เนื่องจากเป็นพอร์ตมาตรฐานที่ Access List หรือไฟร์วอลล์ของเราเตอร์ไซต์ส่วนใหญ่จะเปิดไว้ให้ผ่านเข้าไปยังเน็ตเวิร์คภายในและถึงแม้ เน็ตเวิร์คหรือเครื่องปลายทางจะบล็อกแพ็กเก็ต ICMP และพอร์ต 80 ไว้ก็อาจเปลี่ยนเป็นพอร์ตมาตรฐานที่มักพบบ่อย อย่างเช่น พอร์ต 25 (SMTP), 110 (POP),113 (AUTH),143 (IMAP) โดยถ้าหากเครื่องปลายทางนั้นได้เปิดพอร์ตที่สแกนไปนั้นก็จะมีการตอบกลับมาด้วย

นอกจากนั้นการสแกนพอร์ตยังสามารถใช้เพื่อค้นหาว่ามีพอร์ตอะไรบ้างที่ทำงานอยู่หรืออยู่ในสถานะ LISTENING การค้นหาพอร์ตที่เปิดอยู่เป็นเรื่องสำคัญทีเดียวในการตรวจสอบประเภทของระบบปฏิบัติการและแอปพลิเคชันที่ใช้งานอยู่ในระบบ เพราะ ระบบปฏิบัติการหรือเซอร์วิสที่รันอยู่อาจมีข้อบกพร่องบางอย่างเกี่ยวกับการเปิดพอร์ตที่อนุญาตให้ผู้ใช้ที่ไม่ได้ผ่านการตรวจสอบเข้าไปในระบบได้ หรือมีข้อบกพร่องเกี่ยวกับระบบการรักษาความปลอดภัยที่เป็นที่รู้กันดี โดยเฉพาะเซอร์วิสบางเวอร์ชันที่ยังไม่สมบูรณ์นั่นเอง เครื่องมือและเทคนิคในการสแกนพอร์ตได้รับการพัฒนาอย่างต่อเนื่องมาหลาย สำหรับรูปแบบของการสแกนพอร์ตนั้นแบ่งเป็นรูปแบบที่รู้จักกันดีได้ดังนี้

TCP connect scan เป็นการคอนเน็กไปที่พอร์ตที่ต้องการบนเครื่องปลายทาง แล้วขอเปิดคอนเน็กชันของโพรโตคอล TCP ด้วยกลไกมาตรฐานที่เรียกว่า TCP three-way handshake (SYN, SYN/ACK และ ACK) โดยรูปแบบการทำงานคือ

1. เครื่องต้นทางส่งแพ็กเก็ต TCP ที่เซตแฟลก SYN เป็น 1 ไว้ไปที่เครื่องปลายทาง
2. เครื่องปลายทางส่งแพ็กเก็ต TCP SYN/ACK กลับมาที่เครื่องต้นทาง
3. เครื่องต้นทางส่งแพ็กเก็ต TCP ACK กลับไปเพื่อยืนยันว่าได้รับแพ็กเก็ตในขั้นที่สองแล้ว

TCP SYN scan เทคนิคนี้บางครั้งเรียกว่า half-open scanning สาเหตุเพราะว่า คอนเน็กชันที่สมบูรณ์ของโพรโตคอล TCP ยังไม่ได้ถูกเปิดขึ้น เพราะเมื่อได้รับแพ็กเก็ต TCP ที่เซตค่า

แฟลก SYN และ ACK ไว้เป็น 1 (TCP SYN/ACK) นั่นก็เพียงพอแล้วที่จะสรุปว่า พอร์ตดังกล่าวอยู่ในสถานะ LISTENING แต่ถ้าพอร์ตดังกล่าวไม่ได้เปิดอยู่ แพ็กเก็ต TCP ที่เซตค่าแฟลก RST และ ACK ไว้เป็น 1 (TCP RST/ACK) จะถูกส่งกลับมาแทน

TCP FIN scan เทคนิคนี้จะส่งแพ็กเก็ต TCP ที่เซตค่าแฟลก FIN เป็น 1 (TCP FIN) ไปยังพอร์ตที่ต้องการ ถ้าไดร์เวอร์ของโพรโตคอล TCP/IP ที่เครื่องปลายทางได้ถูกพัฒนาขึ้นมาโดยมีฟีเจอร์ตามในมาตรฐาน RFC 793 เครื่องปลายทางจะส่งแพ็กเก็ต TCP RST ของทุกๆพอร์ตที่ปิดอยู่กลับมาให้ เทคนิคนี้มักใช้ได้กับเครื่องปลายทางที่ใช้ระบบปฏิบัติการยูนิกซ์หรือลินุกซ์

TCP Xmas Tree scan เทคนิคนี้จะส่งแพ็กเก็ต TCP ที่เซตแฟลก FIN ,URG และ PUSH ไปยังพอร์ตเป้าหมายที่เครื่องปลายทาง และอาศัยมาตรฐาน RFC 793 อีกเช่นกัน เครื่องปลายทางจะส่งแพ็กเก็ต TCP RST ของทุกพอร์ตที่ปิดอยู่กลับมาให้

TCP Null scan เทคนิคนี้จะส่งแพ็กเก็ต TCP ออกไปโดยเซตค่าของทุกๆแฟลกให้เป็น 0 หมด และ อาศัยมาตรฐาน RFC 793 อีกเช่นกัน เครื่องปลายทางจะส่งแพ็กเก็ต TCP RSTของทุกๆพอร์ตที่ปิดอยู่กลับมาให้

TCP ACK scan เทคนิคนี้จะถูกใช้เพื่อค้นหา "rule" และ "policy" ต่างๆที่เซตไว้ที่ไฟร์วอลล์เพื่อตรวจสอบดูว่าไฟร์วอลล์นั้นๆทำหน้าที่แค่เพียงกรองแพ็กเก็ตอย่างง่ายๆ หรือเป็นไฟร์วอลล์ที่มีความฉลาดพอสมควรและใช้เทคนิคการกรองแพ็กเก็ตขั้นสูง

TCP Window scan เทคนิคนี้จะตรวจสอบพอร์ตที่เปิดอยู่ รวมทั้งตรวจดูว่า พอร์ตใดบ้างที่ถูกฟิลเตอร์เอาไว้ไม่ให้ผ่านเข้าไปถึง และพอร์ตหมายเลขใดได้รับการอนุญาตไว้บ้าง โดยอาศัยช่องโหว่จากความผิดปกติบางอย่างในการแจ้งค่าของ TCP Window Size ของโพรโตคอลTCP/IP

TCP RPC scan เทคนิคนี้ใช้งานได้เฉพาะกับเครื่องปลายทางที่รันยูนิกซ์เท่านั้น มันถูกใช้เพื่อตรวจสอบดูว่ามีเซอร์วิสใดทำงานอยู่บนเซอร์วิส RPC บ้าง รวมทั้งตรวจดูเวอร์ชันของเซอร์วิสนั้นและโปรแกรมอื่นที่เกี่ยวข้อง

UDP scan เทคนิคนี้จะส่งแพ็กเก็ตของโพรโตคอล UDP ไปยังพอร์ตเป้าหมาย ถ้าเครื่องปลายทางตอบกลับมาด้วยแพ็กเก็ต ICMP type PORT UNREACHABLE นั่นหมายความว่า พอร์ตนั้นปิดอยู่ในทางตรงกันข้าม ถ้าเราไม่ได้รับแพ็กเก็ต ICMP type ดังกล่าว เราสามารถสรุปได้ว่าพอร์ตนั้นเปิดอยู่ เนื่องจากโพรโตคอล UDP เป็นโพรโตคอลลักษณะconnectionless คือไม่รับรองว่าแพ็กเก็ตที่ส่งไปจะถึงเครื่องปลายทางครบถ้วนหรือไม่ ดังนั้นความถูกต้องของผลลัพธ์ที่ได้จากเทคนิคนี้ก็อาจขึ้นกับปัจจัยอื่นๆ ด้วยเช่น ปริมาณทราฟฟิกในเน็ตเวิร์คและทรัพยากรบนเครื่องปลายทาง นอกจากนั้นมันยังเป็นเทคนิคที่ค่อนข้างช้าอีกด้วย

สำหรับไดร์เวอร์ของโพรโตคอล TCP/IP ของระบบปฏิบัติการบางตัวอาจส่งแพ็กเก็ต TCP RST กลับไปยังเครื่องต้นทางตลอกไม่ว่าพอร์ตๆ นั้นจะเปิดหรือปิดอยู่ ดังนั้นจึงเป็นไปได้ว่าผลลัพธ์ที่ได้อาจแตกต่างกันไปไม่แน่นอน แต่อย่างไรก็ดี การตรวจสอบเครื่องปลายทางด้วยการ Scan Port นั้นนับว่ามีประสิทธิภาพพอสมควร

2.1.3 การตรวจหาประเภทของระบบปฏิบัติการ : Impossible Flags และ OS Fingerprint

การตรวจสอบดูว่าเครื่องปลายทางเป็นระบบปฏิบัติการอะไร เวอร์ชั่นไหนเป็นการใช้ข้อบกพร่องของโปรโตคอลที่มิได้มีการกำหนดชัดเจนครอบคลุมทุกเงื่อนไขของการสื่อสารข้อมูล ดังนั้นเมื่อเหตุการณ์ดังกล่าวเกิดขึ้น จึงไม่มีมาตรฐานที่ทุกคนต้องยึดร่วมกัน การตอบสนองของระบบปฏิบัติการแต่ละชนิดจึงแตกต่างกันออกไปขึ้นอยู่กับผู้ผลิตระบบปฏิบัติการนั้นจะอิมพลีเมนต์ TCP/IP อย่างไร และแน่นอนว่าระบบปฏิบัติการแต่ละรุ่นก็มีลักษณะการตอบสนองแตกต่างกันออกไป การสแกนด้วยเทคนิคนี้ส่วนใหญ่จะกระทำบนโปรโตคอล TCP เนื่องจากมีส่วนที่ไม่ได้กำหนดรูปแบบทั้งหมดไว้ในโปรโตคอลอยู่นั่นคือ TCP Flag ซึ่งเป็นส่วนสำคัญที่ใช้ในการควบคุมการสื่อสารของ TCP แต่ TCP Flag เหล่านั้นก็ไม่ได้ถูกนำไปใช้งานทุกๆเงื่อนไขจึงมี Flag บางเงื่อนไขซึ่งไม่มีโอกาสเกิดขึ้นได้จริงในการทำงานปรกติ (Impossible TCP Flags) ซึ่งระบบ ปฏิบัติการแต่ละชนิดแต่ละเวอร์ชั่นก็จะตอบสนองต่อแพ็กเก็ตลักษณะนี้แตกต่างกันออกไปตามเหตุผลที่กล่าวไปแล้วและเมื่อรวบรวมผลลัพธ์ที่ตอบมาจากระบบปฏิบัติการแต่ละชนิดด้วย Impossible Flag หลายๆแบบแล้วจะพบว่ารูปแบบของข้อมูลเหล่านี้สามารถบ่งบอกถึงระบบ ปฏิบัติการได้ซึ่งเรียกรูปแบบของการตอบสนองของ Impossible Flags นี้ว่า OS Fingerprint ดังนั้นความแม่นยำของการทดสอบหาระบบปฏิบัติการด้วยวิธีนี้จะขึ้นอยู่กับจำนวนของแพ็กเก็ตที่ใช้ทดสอบ ยิ่งมีการทดสอบหลายแพ็กเก็ตผลการตอบรับมากก็จะยิ่งชี้ไปยังระบบปฏิบัติการรุ่นใดรุ่นหนึ่งได้อย่างแม่นยำมากขึ้นส่วนอีกปัจจัยหนึ่งคือความทันสมัยของการปรับปรุงฐานข้อมูลลายนิ้วมือให้ทันต่อระบบปฏิบัติการเวอร์ชั่น ใหม่ๆอยู่เสมอ

2.1.4 การตรวจหาประเภทของระบบปฏิบัติการ : Active Stack Fingerprinting

Stack fingerprinting เป็นเทคโนโลยีที่ช่วยทำให้มั่นใจได้ว่าประเภทของระบบปฏิบัติการที่ค้นหาได้นั้นเป็นระบบปฏิบัติการที่ถูกต้อง มีเปอร์เซ็นต์ความน่าเชื่อถือสูง โดยอาศัยหลักการที่ว่า ผู้ผลิตระบบปฏิบัติการแต่ละรายมักพัฒนาไดรเวอร์ของโพรโตคอล TCP/IP และเซอร์วิสที่ทำงานบนโพรโตคอล TCP/IP ให้มีเอกลักษณ์เฉพาะตัวเป็นของตัวเอง ซึ่งมักแตกต่างกับของระบบ ปฏิบัติการอื่นดังนั้น โดยการตรวจวัดความแตกต่างเหล่านี้ เราจะสามารถเริ่มคาดเดาได้อย่างมีเหตุ ผล แต่เพื่อให้มีความน่าเชื่อถือสูงสุด Stack fingerprinting จำเป็นต้องอาศัยการคอนเน็กไปยังพอร์ตที่เปิดอยู่อย่างน้อยหนึ่งพอร์ต แต่การทำงานของโครงงานนี้ สามารถคาดเดาได้อย่างมีเหตุผลถึงแม้ ว่าจะไม่ได้คอนเน็กไปที่พอร์ตใดเลย แต่ผลที่ได้ก็อาจยังไม่น่าเชื่อถือนักโดยอาศัยวิธีการดังนี้

Fin probe ใช้การส่งแพ็กเก็ตของโพรโตคอล TCP โดยเซตแฟลก Fin ให้เป็น 1 ไว้ ตามมาตรฐาน RFC 793 ระบุไว้ว่า พฤติกรรมที่ถูกต้องจะต้องไม่มีการส่งแพ็กเก็ตอะไรตอบสนองกลับไป แต่ทว่า ไดรเวอร์ของโพรโตคอล TCP/IP ในระบบปฏิบัติการบางตัว อย่างเช่น วินโดว์เอ็นที จะตอบสนองกลับมาด้วยแพ็กเก็ตของโพรโตคอล TCP ที่เซตแฟลก Fin และ ACK ให้เป็น 1 ซึ่งเครื่องมือส่วนมากได้นำเทคนิคนี้ไปใช้กัน

Bogus Flag probe ใช้การส่งแพ็กเก็ตของโพรโตคอล TCP โดยเซตแฟลก SYN ให้เป็น 1 พร้อมทั้งเซตบิตตำแหน่งที่ยังไม่ได้ใช้งานให้เป็น 1 ด้วย บางระบบปฏิบัติการเช่น ลีนุกซ์ จะตอบสนองกลับมาด้วยการเซตแฟลกบางตัวในแพ็กเก็ต ว่ามักเป็นค่าอะไร อย่างไรก็ตามบางระบบปฏิบัติการจะทำการ reset connection เมื่อได้รับแพ็กเก็ตนี้

TCP Initial Sequence Number (ISN) sampling ใช้วิธีการตรวจหาแพทเทิร์นของ Sequence number ที่อยู่ในส่วนหัวของแพ็ตเก็ตที่ได้รับการตอบสนองต่อการร้องขอการเชื่อมต่อว่าเป็นค่าอะไร ซึ่งสามารถแบ่งได้หลายกลุ่มเช่น ถ้าเป็นแบบ traditional 64K จะพบใน UNIX รุ่นเก่า,แบบ Random increments จะพบใน Solaris, IRIX, FreeBSD, Digital UNIX, Cray ในเวอร์ชันใหม่, แบบ True "random" จะพบใน Linux 2.0.*, OpenVMS, AIX เวอร์ชันใหม่, ส่วน Windows จะเป็นแบบ "time dependent" model คือค่าของ ISN จะเพิ่มขึ้นจำนวนหนึ่งที่กำหนดไว้ในแต่ละช่วงเวลา, และในบางครั้งจะใช้ค่า ISN ค่าเดียวไม่เปลี่ยนเลย

"Don't fragment bit" monitoring บางระบบปฏิบัติการได้เซตบิต "Don't fragment bit" ไว้เพื่อเพิ่มความเร็วในการส่งข้อมูล ให้มอนิเตอร์บิตนี้เพื่อตรวจดูว่าระบบปฏิบัติการไหนเซตบิตนี้บ้าง

TCP initial window size ใช้วิธีดูขนาดของ TCP window เพราะบางระบบปฏิบัติการจะมีการล็อคค่าไว้เลยว่าขนาดของ TCP window เป็นเท่าไหร่ (เช่น AIX เป็นเพียงระบบปฏิบัติการเดียวที่มีขนาดของ window เท่ากับ 0x3F25 ส่วน window NT มีขนาด 0x402E) ค่านี้มักเป็นค่าเฉพาะตัวของแต่ละระบบปฏิบัติการด้วย จึงยิ่งทำให้ผลที่ได้มีความถูกต้องมากขึ้น

ACK value ระบบปฏิบัติการแต่ละระบบมักเซตค่าในฟิลด์ ACK ไม่เหมือนกัน บางระบบเซตค่าฟิลด์ ACK ให้สอดคล้องตามค่าของฟิลด์ SYN ที่เซตไว้ในฝั่งผู้ส่ง บางระบบจะเซตค่าฟิลด์ ACK ให้บวกจากค่าของฟิลด์ SYN ไปอีกหนึ่ง

ICMP error message quenching บางระบบปฏิบัติการอาจปฏิบัติตามมาตรฐาน RFC 1812 และมีการจำกัดอัตราการส่งข้อความแจ้งข้อผิดพลาดดังนั้น (ใน Linux kernel จะจำกัดการส่งข้อความแจ้งข้อผิดพลาดไปยังปลายทางที่อัตรา 80 แพ็กเก็ตต่อ 4±0.25 วินาที) โดยการส่งแพ็กเก็ตของโพรโตคอล UDP ไปยังหมายเลขพอร์ตสูงๆ แล้วค่อยนับจำนวนของ ICMP type PORTUNREACHABLE Message ที่ตอบกลับมาภายในช่วงเวลาที่กำหนด

ICMP message quoting เมื่อพบข้อผิดพลาดเกี่ยวกับโพรโตคอล TCP/IP ระบบปฏิบัติการแต่ละประเภทจะให้ข้อมูลและสาเหตุต่างๆมาในแพ็กเก็ตของ ICMP ไม่เท่ากัน บางระบบจะให้รายละเอียดมากบางระบบจะให้รายละเอียดน้อย (ใน Solaris จะส่งกลับมา a bit more และใน Linux จะส่งกลับมา even more than that) ดังนั้นโดยการสำรวจข้อมูลที่ได้มานี้ เราพอจะใช้ในการคาดเดาประเภทของระบบปฏิบัติการได้

ICMP error message-echoing integrity บางระบบปฏิบัติการอาจดัดแปลงส่วนหัวของแพ็กเก็ต IP เมื่อมีการส่ง ICMP error message ด้วยการตรวจสอบลักษณะของการดัดแปลงดังกล่าวนี้คุณสามารถนำมันมาใช้ในการคาดเดาได้

Type of service (TOS) ให้คุณสำรวจฟิลด์ที่ชื่อ Type of service ที่อยู่ในแพ็กเก็ต ICMP ประเภท port unreachable ซึ่งโดยปกติหลายๆระบบปฏิบัติการจะใช้ค่าศูนย์ แต่บางระบบอาจใช้ค่าอื่น เช่น Linux ใช้ 0xC0

Fragmentation handling แต่ละระบบปฏิบัติการจะจัดการกับแพ็กเก็ตที่ถูกแฟรกเมนต์หรือหั่นซอยไม่เหมือนกัน เมื่อมีการประกอบแพ็กเก็ตย่อยขึ้นมาเป็นแพ็กเก็ตที่สมบูรณ์ บางระบบจะเขียนทับแพ็กเก็ตย่อยอันเก่าด้วยแพ็กเก็ตย่อยอันใหม่หรือบางระบบก็ทำในทางตรงกันข้าม โดยการสังเกตพฤติกรรมตรงนี้ เราพอจะใช้ในการคาดเดาประเภทของระบบปฏิบัติการได้เช่นเดียวกัน

TCP options ได้ถูกกำหนดไว้ในมาตรฐาน RFC 793 และได้รับการปรับปรุงในมาตรฐาน RFC 1323 ในปัจจุบัน ผู้ผลิตหลายรายได้มีการอิมพลีเมนต์ออปชั่นพิเศษที่อยู่ใน RFC 1323 ไว้ในระบบปฏิบัติการของตน ดังนั้น ด้วยการส่งแพ็กเก็ตที่เซตออปชั่นหลายๆออปชั่นไปทดสอบอย่างเช่น no operation, maximum segment size, window scale factor และ timestamps มันเป็นไปได้ที่เราจะตั้งสมมติฐานเกี่ยวกับระบบปฏิบัติการที่รันที่เครื่องเป้าหมาย

ยกตัวอย่าง

Window Scale=10; NOP; Max Segment Size = 265; Timestamp; End of Ops;

บางระบบปฏิบัติการ เช่น FreeBSD จะ support ทุกออปชั่นข้างต้นขณะที่ Linux 2.0.X จะ support บางออปชั่น Linux 2.1.x support ทุกออปชั่นแม้ว่าหลายระบบปฏิบัติการจะ support ออปชั่นเดียวกันแต่บางทีเราก็สามารถแยกแยะได้โดยจากออปชั่น the_values_of เช่น ถ้าส่งค่า MSS เล็กๆไปที่ Linux โดยทั่วไปแล้วมันจะส่ง MSS echoกลับไป ส่วนระบบปฏิบัติการอื่นจะส่งค่าที่แตกต่างกันกลับไป แต่ถ้ายังได้ผลลัพธ์เหมือนกันอีกก็สังเกตลำดับของออปชั่นที่ได้รับมา เช่น Solaris จะเป็น 'NNTNWME' ซึ่งหมายความว่า <no op><no op><timestamp><no op><window scale><echoed MSS> ขณะที่ Linux 2.1.122 จะเป็นMENNTNW ซึ่งจะเห็นได้ว่าเป็นออปชั่นเดียวกัน ส่ง MSS echo เหมือนกัน แต่ลำดับที่ส่งมาไม่เหมือนกัน

SYN Flood Resistance บางระบบปฏิบัติการจะไม่ยอมรับการ connection ครั้งใหม่ ถ้ามีการส่งแพ็กเก็ต SYN ไปมากเกิน ซึ่งหลายระบบปฏิบัติการนั้นสามารถที่จะจัดการกับแพ็กเก็ตที่ส่งเข้ามาได้ไม่เกินครั้งละ 8 แพ็กเก็ต แต่ใน kernelของ Linux รุ่นใหม่มีวิธีในการป้องกันปัญหานี้ เช่น 64 การใช้ SYN cookies ดังนั้นเราก็สามารถรู้ได้ว่าเป็นระบบ ปฏิบัติการอะไรโดยการส่งแพ็กเก็ตไป 8 แพ็กเก็ตไปยังพอร์ตที่เปิดอยู่แล้วดูว่าสามารถ connect ไปยังพอร์ตนั้นได้หรือไม่

2.2 ทฤษฎี และหลักการในส่วนของ SNMP และ RMON

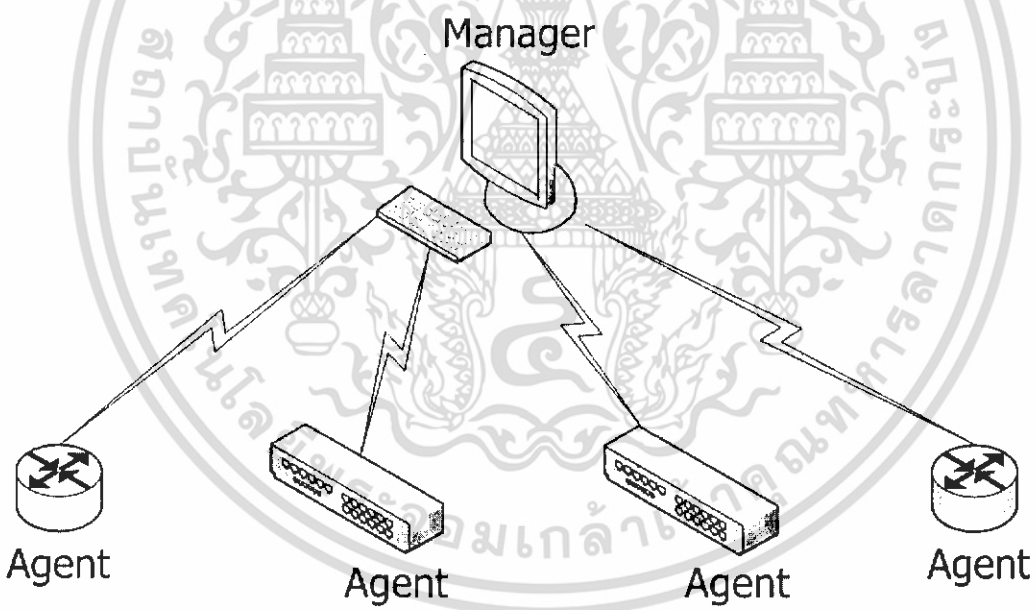
SNMP เป็นโปรโตคอลในระดับประยุกต์ (Application Layer) ที่กำหนดรูปแบบ และกรรมวิธีการจัดการเครือข่าย โดยมีสถานีจัดการเครือข่ายส่วนกลางทำหน้าที่ดูแล ตรวจสอบ และควบคุมการทำงานของอุปกรณ์เครือข่าย

2.2.1 พื้นฐานการบริหารเครือข่าย

การบริหารเครือข่าย คือ การตรวจ ควบคุม และวางแผนการใช้ทรัพยากรระบบเพื่อให้เครือข่ายทำงานได้อย่างมีประสิทธิภาพ และ สามารถตรวจหาจุดบกพร่องที่เกิดขึ้นเพื่อแก้ไขปัญหาได้อย่างรวดเร็ว

ในระบบเครือข่ายใดๆที่ ต้องใช้การจัดการเครือข่าย TCP/IP จะต้องประกอบด้วย 4 ส่วน (Model) ด้วยกัน คือ

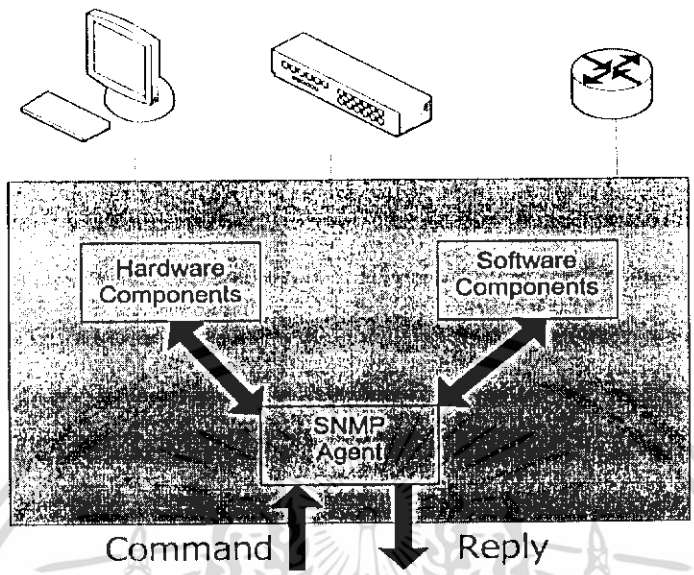
Management station : เป็นสถานีการจัดการเครือข่ายส่วนกลางซึ่งทำหน้าที่ดูแล ตรวจสอบ และ ควบคุม การทำงานของอุปกรณ์ในเครือข่าย ดังรูปที่ 2.1



รูปที่2.1 แสดงองค์ประกอบในระบบจัดการเครือข่าย

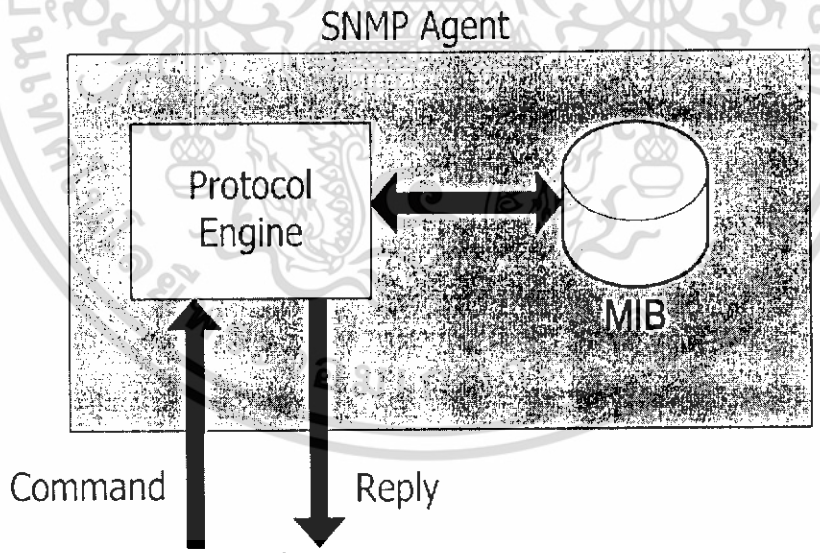
Management agent : เป็นสมาชิกในระบบการจัดการเครือข่าย ซึ่งมีฟังก์ชันที่ให้ตรวจสอบ และปรับเปลี่ยนการทำงานได้ อาจจะเป็น โฮสต์, บริดจ์ , สวิตซ์ , เราท์เตอร์, ฮับ หรืออุปกรณ์อื่นๆก็ได้ที่สามารถส่งข้อมูลสถานะของมันออกไปยังระบบได้ จะเป็นอุปกรณ์ ฮาร์ดแวร์ หรือ ซอฟท์แวร์ ก็ได้ แต่ต้องมี เอสเอ็นเอ็มพีเอเจนต์ อยู่ในตัวด้วย

โดยเอเจนต์จะนำการทำงานของซอฟต์แวร์ หรือฮาร์ดแวร์ เมื่อสถานีจัดการเครือข่าย ร้องขอข้อมูล และปรับเปลี่ยนการของซอฟต์แวร์ หรือฮาร์ดแวร์ เมื่อสถานีจัดการเครือข่ายสั่งงาน โดยมีการยืนยันสิทธิในรูปรหัสผ่านว่ามีอำนาจหน้าที่ในการร้องขอ และปรับค่าได้ ดังรูป 2.2



รูปที่ 2.2 เอสเอ็นเอ็มพีเอเจนต์

เอเจนต์ประกอบด้วยส่วนสำคัญ 2 ส่วน ด้วยกัน คือ โปรโตคอลเอ็นจิน (Protocol engine) และฐานข้อมูลสารสนเทศการจัดการ (Management information base) ดังรูป 2.3



รูปที่ 2.3 โครงสร้างของเอเจนต์

โปรโตคอลเอ็นจินทำหน้าที่ประมวลคำสั่งที่มาจากสถานีจัดการเครือข่ายได้แก่ รับคำสั่ง ถอดรหัสคำสั่ง ทำงานตามคำสั่ง และส่งผลตอบกลับ ฐานข้อมูลสารสนเทศการจัดการเป็นส่วนที่เก็บตัวแปร และค่ากำหนดการทำงานประจำอุปกรณ์

Management information base (MIB) : ฐานข้อมูลสารสนเทศการจัดการเป็นส่วนที่เก็บตัวแปร และค่ากำหนดการทำงานประจำอุปกรณ์

Network management protocol : ทำหน้าที่ติดต่อระหว่างสถานีจัดการ กับเอเจนต์ มีรูปแบบในการติดต่อ หลายรูปแบบด้วยกันตามวัตถุประสงค์ในการติดต่อ โดยการจัดการเครือข่ายใน TCP/IP จะอาศัยรูปแบบการจัดการมาตรฐานตามข้อกำหนดของโปรโตคอล SNMP ซึ่งเป็นโปรโตคอลประยุกต์ที่กำหนดรูปแบบ และกรรมวิธีจัดการเครือข่าย

2.2.2 Communities และ Community Names

การบริหารเครือข่ายจะถือได้ว่าเป็นการทำงานในลักษณะระบบกระจาย (Distributed application)รูปแบบหนึ่ง ซึ่งจะเห็นได้ว่าความสัมพันธ์ระหว่างสถานีจัดการเครือข่าย (Network Management Station : NMS) กับเอเจนต์ (Agent) จะเป็นในรูปแบบ many-to-many คือ สถานีจัดการเครือข่ายจะบริหารจัดเอเจนต์ได้หลายเครื่อง และในขณะเดียวกันเอเจนต์ก็สามารถถูกบริหารควบคุมจากสถานีจัดการเครือข่ายหลายเครื่องเช่นกัน

จากความสัมพันธ์ดังกล่าวจึงจำเป็นที่เอเจนต์แต่ละเครื่องจำเป็นที่จะต้องมีมาตรการควบคุมความปลอดภัยในการใช้งานฐานข้อมูลสารสนเทศการจัดการ (Management Information Base : MIB) ของตนเองโดยจะมีมุมมองทางด้านความปลอดภัย 3 ประการได้แก่

1. การพิสูจน์ตัวตน (Authentication service) จะเป็นการจำกัดให้เฉพาะสถานีจัดการเครือข่ายที่จะเข้ามาบริการควบคุม
2. นโยบายการเข้าถึง (Access policy) จะมีการกำหนดระดับการอนุญาตการเข้าถึงฐานข้อมูลสารสนเทศการจัดการให้แต่ละสถานีจัดการเครือข่ายไม่เท่ากันในแต่ละเครื่อง
3. การให้บริการ Proxy (Proxy service) เอเจนต์อาจทำหน้าที่เป็น Proxy ให้กับเอเจนต์เครื่องอื่นซึ่งจะรวมถึงการพิสูจน์ตัวตนและนโยบายการเข้าถึงของเอเจนต์ตัวอื่นที่อยู่ในระบบ Proxy

เอสเอ็นเอ็มพี (SNMP) ได้มีการกำหนดการทำงานเพื่อสนับสนุนมุมมองทางด้านทางปลอดภัยดังกล่าวในรูปแบบของ SNMP Community โดยการทำงานคือ เอเจนต์แต่ละเครื่องจะมีการสร้าง Community name เพื่อกำหนดให้กับสถานีจัดการเครือข่าย โดนในหนึ่ง Community name จะสามารถมีสถานีจัดการเครือข่ายมากกว่าหนึ่งตัว

เนื่องจาก Community name จะถูกกำหนดในแต่ละเอเจนต์จึงอาจเป็นไปได้ว่ามีการตั้งชื่อ Community name ซ้ำกันในแต่ละเอเจนต์ แต่สถานีจัดการเครือข่ายจะสามารถแยกความแต่ต่างของ Community ที่มีชื่อซ้ำกันเหล่านี้เองได้ถ้าอยู่ในคนละเอเจนต์กัน ดังนั้นจึงจำเป็นที่ว่าสถานีจัดการเครือข่ายจะต้องเก็บข้อมูลของ Community name และข้อมูลที่เกี่ยวข้องของแต่ละเอเจนต์เพื่อใช้ในการบริหารควบคุม

การพิสูจน์ตัวตน (Authentication service)

การพิสูจน์ตัวตนมีไว้เพื่อให้แน่ใจการติดต่อนั้นเป็นของแท้ ในกรณีของเอสเอ็นเอ็มพีการพิสูจน์ตัวตนจะมีไว้เพื่อให้แน่ใจว่า message ที่ได้รับมานั้นเป็นข้อความที่แท้จริง โดยในทุกๆ

message ของเอสเอ็นเอ็มพีจะมีการระบุ Community name ซึ่งจะมีหน้าที่เสมือนกับรหัสผ่าน (Password) ในการพิสูจน์ตัวตน นอกจากนั้นยังอาจจะมีการเข้ารหัส (Encryption) เพื่อเพิ่มความปลอดภัยในการพิสูจน์ตัวตนมากยิ่งขึ้น

นโยบายการเข้าถึง (Access policy)

การควบคุมการเข้าถึงในเอสเอ็นเอ็มพีจะประกอบด้วย 2 องค์ประกอบหลักที่เกี่ยวข้องคือ

1. SNMP MIB view: คือกลุ่มของอ็อบเจ็กในฐานข้อมูลสารสนเทศการจัดการที่ตั้งขึ้นโดย ในแต่ละกลุ่มอาจจะประกอบด้วยหลาย Sub tree ในฐานข้อมูลสารสนเทศการจัดการได้
2. SNMP access mode: คือรูปแบบของการเข้าถึงได้แก่ READ-ONLY และ READWRITE

ในเอเจนต์จะมีการกำหนด Access mode ให้แต่ละ MIB view ซึ่ง Access mode จะมีผลกับทุกๆObject ที่อยู่ในกลุ่มของ MIB view โดยทั้ง Access mode และ MIB view จะถูกรียกรวมกันว่า SNMP community profile ซึ่งจะถูกกำหนดในแต่ละ Community

ใน Access mode ของเอสเอ็นเอ็มพีจะมีการประนีประนอมต่อรองระดับการเข้าถึงกับระดับของการเข้าถึงของฐานข้อมูลสารสนเทศการจัดการ (MIB Access Category) ดังตารางที่ 2.1 (Access mode ของเอสเอ็นเอ็มพีกับระดับของการเข้าถึงของฐานข้อมูลสารสนเทศการจัดการเป็นคนละส่วนกัน) และจะเรียกรวม SNMP community และ SNMP community profile ว่า SNMP access policy

MIB Access Category	**SNMP Access Mode**	
	READ-ONLY	**READ-WRITE**
read-only	สามารถใช้คำสั่ง get และ trap ได้	
read-write	สามารถใช้คำสั่ง get และ trap ได้	สามารถใช้คำสั่ง get, set และ trap ได้
write-only	สามารถใช้คำสั่ง get และ trap แต่ต้องเป็นค่าที่เป็น implementation-specific	สามารถใช้คำสั่ง get, set และ trap ได้แต่ต้องเป็นค่าที่เป็น implementation-specific สำหรับคำสั่ง get และ trap
not accessible	ไม่สามารถเข้าถึงได้	

ตารางที่ 2.1 ความสัมพันธ์ระหว่าง MIB Access Category และ SNMP Access Mode

การให้บริการ Proxy (Proxy service)

แนวคิดของ Community จะสามารถสนับสนุนการทำ Proxy ได้ โดยเอเจนต์จะสามารถทำหน้าที่เป็นตัวแทนในการติดต่อกับสถานีจัดการเครือข่ายให้กับเอเจนต์หรืออุปกรณ์อื่นได้ ในกรณีที่อุปกรณ์เครื่องนั้นไม่สนับสนุน TCP/IP และเอสเอ็นเอ็มพีหรือในกรณีที่ต้องการลดปริมาณ

การติดต่อระหว่างอุปกรณ์นั้นกับสถานีจัดการเครือข่าย โดยเอเจนต์ที่เป็น Proxy จะสนับสนุน SNMP access policy ของเอเจนต์ที่อยู่ในระบบ Proxy ด้วย

2.2.3 MIB (Management Information Base)

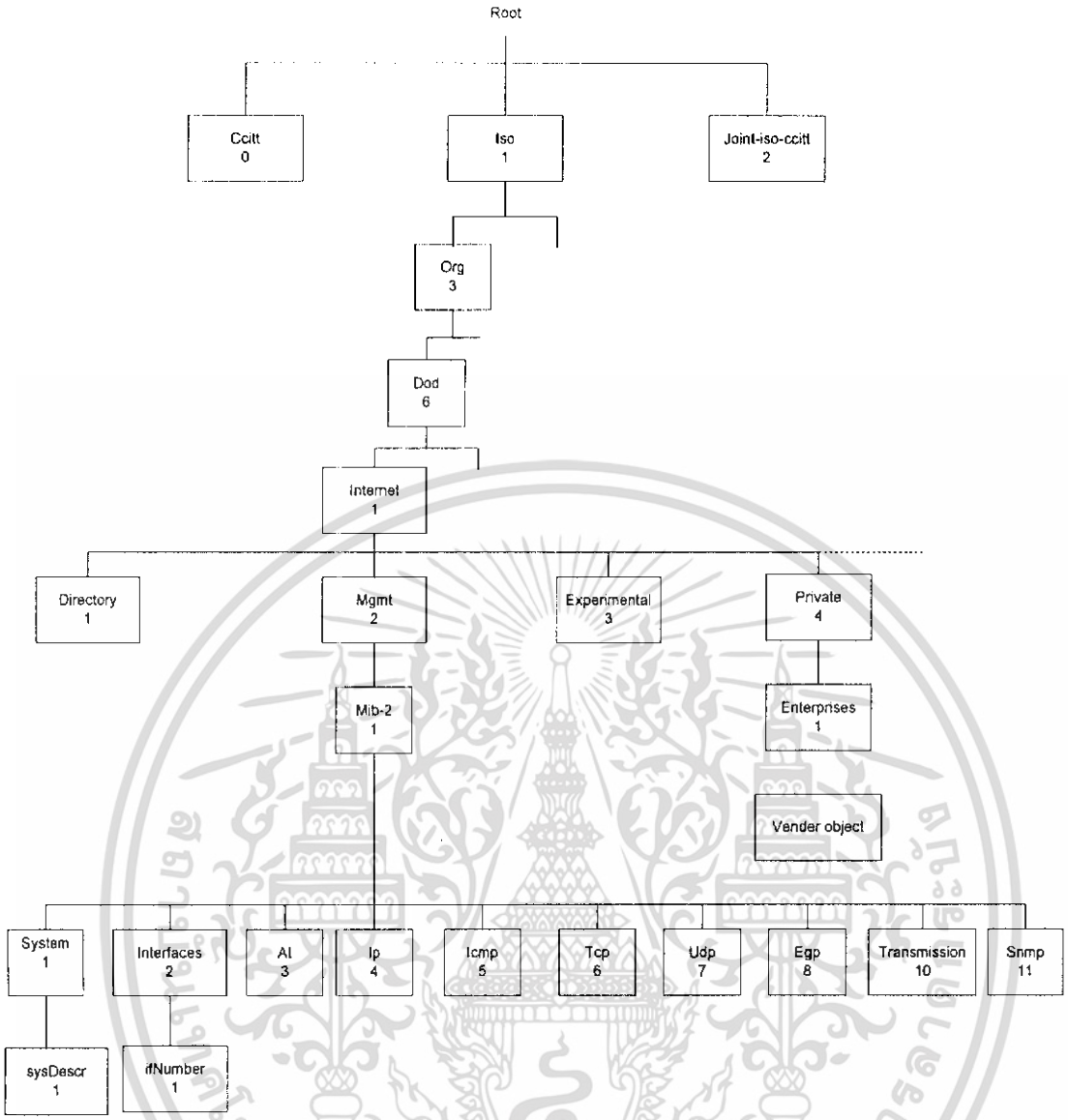
ภายใน MIB ประกอบด้วยตัวแปรจำนวนมากที่เรียกโดยทั่วไปว่า อ๊อบเจ็กจัดการ (managed object) อ๊อบเจ็กในความหมายนี้เป็นชื่อที่ใช้เรียกตัวแปรและลักษณะเฉพาะของตัวแปรในMIBโดยไม่เกี่ยวข้องกับ เชิงวัตถุพิสัย (object oriented) แต่อย่างใด โดยจะพิจารณาอ๊อบเจ็กใน SNMP มีลักษณะเช่น เดียวกับเรคอร์ดในฐานข้อมูล

แต่ละอ๊อบเจ็ก จะมีชื่อเรียกเฉพาะเรียกว่า อ๊อบเจ็กไอเด็นติไฟเออร์ (Object Identifier) หรือเรียกโดยย่อว่า ไอเด็นติไฟเออร์ (Identifier) เพื่อใช้อ้างอิงถึงอ๊อบเจ็กนั้น

อ๊อบเจ็กทุกตัวมีนิยามที่กำหนด ชื่อ แบบข้อมูล สิทธิ์การเข้าถึง คำอธิบายลักษณะ และค่าข้อมูลการนิยามอ๊อบเจ็กมีกฎเกณฑ์ตามข้อกำหนด โครงสร้างฐานข้อมูลสารสนเทศ (Structure of Management Information: SMI) [RFC 1155]

โครงสร้าง MIB

ข้อมูลประจำอุปกรณ์เครือข่ายชิ้นหนึ่งๆมีได้อย่างหลากหลาย อีกทั้งอุปกรณ์ต่างประเภทกันย่อมมีข้อมูลประจำอุปกรณ์แตกต่างกัน ดังนั้นการสอบถาม (อ่าน) หรือเปลี่ยนค่า (เขียน) ฐานข้อมูลจึงต้องมีรูปแบบมาตรฐานให้กับอุปกรณ์ทุกประเภท โครงสร้างต้นไม้แบบลำดับชั้นเป็นโครงสร้างที่เหมาะสมสำหรับใช้เป็นฐานข้อมูลเพื่อจัดเก็บตัวแปรเหล่านี้



รูปที่ 2.4 อ๊อบเจ็กไอเด็นติไฟเออร์ในโครงสร้างฐานข้อมูลสารสนเทศการจัดการ

ดังรูป 2.4 แสดงข้อมูลหรืออ๊อบเจ็กของเอสเอ็นเอ็มพีในโครงสร้างต้นไม้ซึ่งนิยมเรียกว่า มิบทรี(MIB Tree) แต่ละโหนดซึ่งแทนอ๊อบเจ็กหนึ่งๆมีชื่อพร้อมทั้งตัวเลขฐานสิบกำกับประจำโหนดเพื่อใช้อ้างอิง ยกเว้นรากซึ่งไม่มีชื่อกำกับ

ลำดับชั้นแรกจะมีโหนดหลัก 3 โหนดซึ่งกำหนดกลุ่มองค์กร 3 กลุ่มคือ ITU-T(0),ISO(1) และJoint-ISO-ITU-T(2) ภายใต้โหนด ISO มีโหนดลำดับที่ 3 คือ org(3) กำหนดองค์กรนานาชาติ และส่วนหนึ่งขององค์กรนี้คือ dod (6) หรือ Department of Defense และมี โหนด internet(1) เพื่อกำหนดกลุ่มการจัดการเครือข่ายใน internet

เมื่อต้องการอ้างอิงถึงโหนดใดในโครงสร้าง ให้เขียนหมายเลขจากรากไปตามเส้นทางถึงโหนดนั้นและคั่นด้วยจุด ลำดับตัวเลขนี้เรียกว่า อ๊อบเจ็กไอเด็นติไฟเออร์(Object Identifier) หรือ โอไอดี (OID)

ตัวอย่างเช่น 1.3.6.1.2.1.1 เป็นอ๊อบเจ็กไอเด็นติไฟเออร์โดยมีชื่อที่สมนัยกันคือ iso.org.dod.internet.mgmt.mib-2.system โหนดที่อยู่ภายใต้ 1.3.6.1.2.1 หรือในกลุ่ม mib-2 เป็นโหนดสำหรับใช้งานเอสเอ็นเอ็มพี แต่ละโหนจะมีโหนดย่อยเพื่ออ้างอิงถึงตัวแปร เช่น 1.3.6.1.2.1.1.1 คือตัวแปรsysDescr (System Description) ซึ่งเก็บคำอธิบายเกี่ยวกับอุปกรณ์นั้น

กลุ่มในมิบ

มิบภายใต้ internet มีกลุ่มย่อยทั้งหมด 6 กลุ่มคือ

1. directory(1) สงวนไว้สำหรับใช้งานในอนาคต
2. mgmt(2) กลุ่มมิบที่ใช้ในการจัดการภายใต้เอสเอ็นเอ็มพีรุ่น 1
3. experimental(3) ใช้สำหรับการทดลอง
4. private(4) สำหรับผู้ผลิตกำหนดตัวแปรเฉพาะอุปกรณ์
5. security(5) ใช้ในระบบรักษาความปลอดภัย
6. SNMPv2(6) ใช้ในเอสเอ็นเอ็มพีรุ่น2

ภายใต้กลุ่ม mib-2(1.3.6.1.2) บรรจุกลุ่มย่อยที่ใช้ในเอสเอ็มเอ็นพีซึ่งประกอบด้วย interface, at, ip และอื่นๆ

ความหมายของแต่ละกลุ่มอธิบายไว้ในตารางที่ 2-2 แต่ละกลุ่มประกอบด้วยตัวแปรซึ่งมีแบบต่างๆกันไป

ลำดับ	ชื่อ	ความหมาย
1	system	ข้อมูลระบบ
2	interface	ข้อมูลอินเทอร์เฟสที่ใช้เชื่อมต่อ
3	at	ข้อมูลการแปลงแอดเดรส
4	ip	ข้อมูลไอพี
5	icmp	ข้อมูลไอซีเอ็มพี
6	tcp	ข้อมูลทีซีพี
7	udp	ข้อมูลยูดีพี
8	egp	ข้อมูลโปรโตคอลเกตเวย์ภายนอก
10	transmission	ข้อมูลสายสื่อสาร
11	SNMP	ข้อมูลเอสเอ็นเอ็มพี
16	RMON	ข้อมูลสำหรับใช้ตรวจสอบค่าประจำอุปกรณ์แบบรีโมต

ตารางที่ 2.2 กลุ่มย่อยภายใต้ mgnt

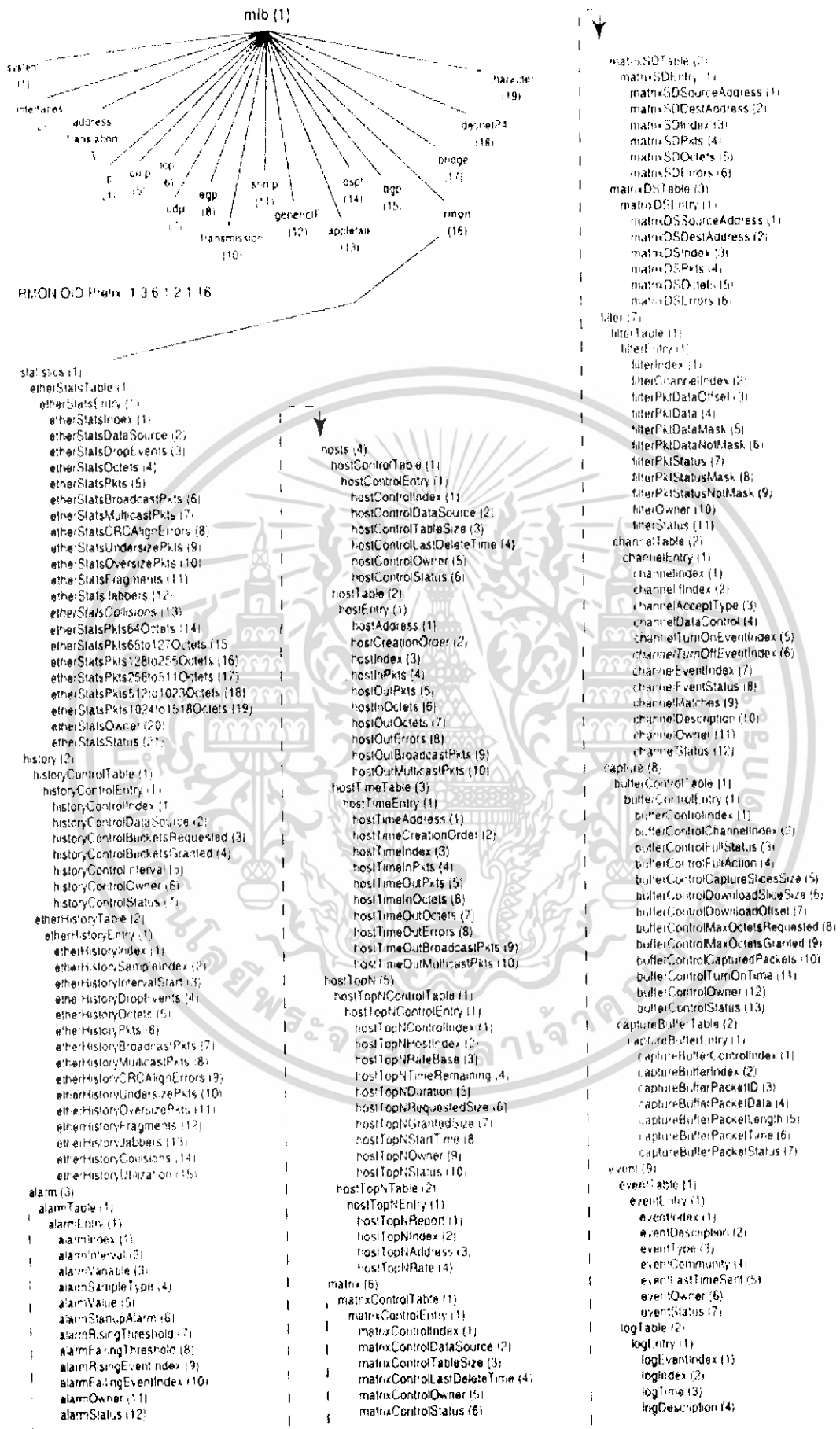
RMON

Remote network monitoring (RMON) เป็นมิบที่สำคัญชุดหนึ่ง ซึ่งรวมอยู่ในมาตรฐานของเอสเอ็นเอ็มพี ใช้ในการตรวจสอบค่าประจำอุปกรณ์แบบรีโมต และทำการบันทึกค่าสถิติของอุปกรณ์บนเครือข่ายได้

ภายใต้กลุ่ม rmon (1.3.6.1.2.1.16) บรรจุกลุ่มย่อยที่ใช้ในRMONซึ่งประกอบด้วย 9 กลุ่มหลักด้วยกัน ดังตารางที่ 2.3 และรูปที่ 2.5

ลำดับ	ชื่อ	ความหมาย
1	statistics	ข้อมูลทางสถิติ เช่น จำนวน และขนาดต่างๆของแพ็กเกต ที่ได้รับ ทั้ง broadcasts , collisions , fragments ฯลฯ
2	history	บันทึกสถิติเป็นช่วงๆเพื่อนำมาวิเคราะห์
3	alarm	เปรียบเทียบสถิติกับค่าที่ตั้งไว้ซึ่งหากตรวจพบว่าสถิติมีค่าเกินกว่าที่ตั้งไว้จะทำการส่งสัญญาณเตือน(Alarm)
4	hosts	บันทึกสถิติเป็นรายเครื่องที่ใช้บริการเน็ตเวิร์ครวมทั้ง MAC Address ของแต่ละเครื่องด้วย
5	hosTopN	แสดงรายงานของเครื่องในเน็ตเวิร์คที่มีค่าทางสถิติสูงที่สุด โดยสามารถเลือกได้ว่าจะใช้ค่าใดเป็นเกณฑ์ในการแสดงรายงาน
6	matrix	บันทึกสถิติของการสื่อสารระหว่างคู่ของเครื่องภายในระบบ
7	filter	ยอมรับให้แพ็คเก็ตที่ตรงกับเงื่อนไขที่ได้ตั้งไว้ถูกตรวจสอบ
8	capture	ทำการตรวจจับและส่งแพ็คเก็ตไปยัง Management console เมื่อแพ็คเก็ตตรงกับเงื่อนไขที่ได้ตั้งไว้
9	event	เก็บสถิติของ Event ที่สร้างโดย RMON Probe

ตารางที่ 2.3 กลุ่มย่อยภายใต้ rmon



รูปที่ 2.5 โครงสร้างของกลุ่มย่อยภายใต้ rmon

ชนิดของตัวแปรมิบ

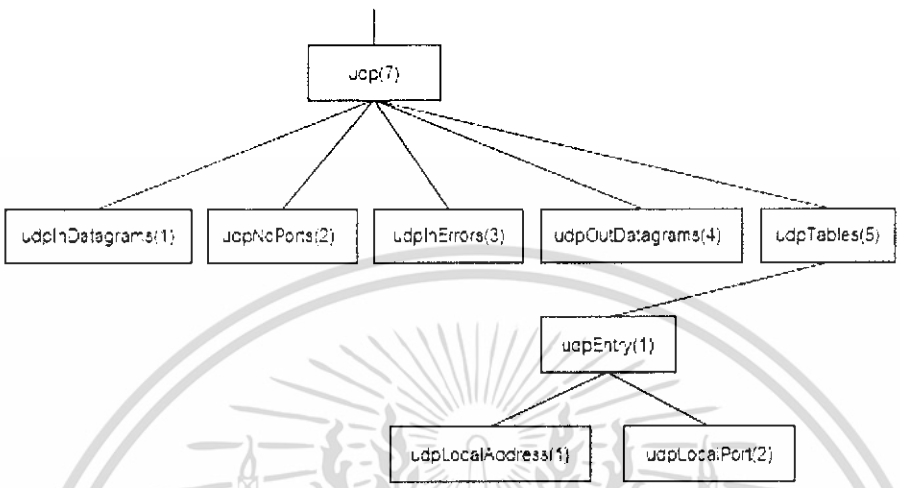
แต่ละตัวแปรในเอสเอ็นเอ็มพีมีข้อมูลประจำ แบบข้อมูลที่ใช้อยู่ในเอสเอ็นเอ็มพีมีดังนี้

- Integer: จำนวนเต็มเช่นหมายเลขพอร์ตของโปรโตคอลทีซีพีหรือยูดีพี มีค่าได้ตั้งแต่ 0 ถึง65535
- OctedString: สายอักขระขนาดตั้งแต่ 0 อ๊อกเท็ต แต่ละอ๊อกเท็ตมีค่าตั้งแต่ 0 ถึง 255ตัวอย่างแบบข้อมูลสายอักขระได้แก่ รหัสผ่าน
- DisplayString: สายอักขระตั้งแต่ 0 อ๊อกเท็ตแต่ละอ๊อกเท็ตต้องเป็นรหัสแอสกีเอ็นวีที ข้อมูลประเภทนี้มีความยาวตั้งแต่ 0 ถึง 255 ตัวอักษร
- Null: ใช้บอกว่าตัวแปรนั้นไม่มีค่าข้อมูลใดอยู่ เช่นเมื่อสอบถามข้อมูลด้วยคำสั่ง Get หรือGetNextRequest จะกำหนดแบบข้อมูลตัวแปรเท่ากับ null
- ObjectIdentifier: ชื่อตัวแปรในรูปแบบของการอ้างถึงแบบตัวเลขตามโครงสร้างมิบ
- IpAddress: สายอักขระ 4 อ๊อกเท็ต แต่ละอ๊อกเท็ตแทนไอพีแอดเดรสแต่ละตำแหน่ง
- PhysicalAddress: สายอักขระกำหนดฮาร์ดแวร์แอดเดรสเช่น อีเทอเน็ตแอดเดรสใช้สายอักขระ 6 อ๊อกเท็ต
- Counter: เลขจำนวนเต็มไม่คิดเครื่องหมาย มีค่าตั้งแต่ 0 ถึง 2 – 1 (4,294,967,295) การใช้ข้อมูล counter เป็นแบบเพิ่มค่าขึ้นอย่างเดียว เมื่อเพิ่มถึงค่ามากสุดจะกลับเป็น 0 ใหม่
- Gauge: เลขจำนวนเต็มไม่คิดเครื่องหมาย มีค่าตั้งแต่ 0 ถึง 2 -1 โดยสามารถเพิ่มหรือลดค่าได้ แต่เมื่อเพิ่มไปสูงสุดแล้วจะคงค่าไว้จนกว่าจะถูกปรับค่ากลับมาเป็น 0 อีกครั้ง ตัวอย่างตัวแปรที่ใช้ค่านี้เช่น จำนวนการเชื่อมโยงทีซีพีที่อนุญาตให้มีได้
- TimeTicks: เลขจำนวนเต็มใช้นับเวลาให้หน่วยเศษหนึ่งส่วนร้อยของวินาที เช่น เวลานับตั้งที่ระบบเริ่มทำงาน
- Sequence: โครงสร้างแบบเร็คคอร์ด หรือคล้ายกับแบบข้อมูล struct ในภาษาซี
- Sequence of: โครงสร้างแบบตารางหรือมองในรูปของอาร์เรย์ เช่น ตารางเลือกเส้นทางของไอพี

ตัวอย่างแบบข้อมูลอาร์เรย์

แบบข้อมูล sequence of ใช้กำหนดข้อมูลแบบเวกเตอร์ซึ่งสมาชิกทั้งหมดภายในมีข้อมูลแบบเดียวกัน หากสมาชิกมีแบบข้อมูลเบื้องต้น เช่น แบบจำนวนเต็มก็จะได้เวกเตอร์แบบมิติเดียว หรือหากสมาชิกมีข้อมูลแบบ sequence ก็จะได้อาร์เรย์ 2 มิติ (ตาราง)

ตัวอย่างของตัวแปรโครงสร้างอาร์เรย์ในมิบมีอยู่หลายตัวแปร แต่จะยกตัวอย่างเฉพาะตัวแปรที่มีขนาดเล็กเพื่อที่จะทำความเข้าใจได้ง่ายได้แก่ udpTable ซึ่งสังกัดอยู่ภายใต้กลุ่ม udp ตามโครงสร้างข้อมูล ดังรูปที่ 2.5



รูปที่ 2.6 กลุ่ม udp

udpTable มีแบบข้อมูล sequence of และภายใต้ udpTable มี udpEntry ซึ่งมีแบบข้อมูล sequence โดยประกอบด้วย udpLocalAddress และ udpLocalPort

udpLocalAddress มีแบบข้อมูล IpAddress ใช้กำหนดไอพีแอดเดรสที่รอให้บริการส่วนของ udpLocalPort กำหนดหมายเลขพอร์ตดังนั้น udpTable จึงมีโครงสร้างเป็นอาร์เรย์ 2 มิติหรือเขียนด้วย ตารางดังรูปที่2.6

	udpLocalAddress	udpLocalPort
udpEntry	(IpAddress)	(Integer)
udpEntry	...	...
udpEntry	...	...
udpEntry	...	...
udpEntry	...	...

รูปที่ 2.7 udpTable ในรูปอาร์เรย์ 2 มิติ (ตาราง)

การอ้างอิงถึงค่าในอ็อบเจ็ก (Instance Identification)

ในการอ้างอิงถึงค่าของอ็อบเจ็กในฐานข้อมูลสารสนเทศการจัดการของเอสเอ็นเอ็มพีนั้นจะอาศัย การอ้างอิงของอินสแตนท์ไอเด็นติไฟเออร์ (Instance Identifier)

โดยการอ้างถึงค่าของอ็อบเจ็กแบบปกติ (Simple Object Value) โดยทั่วไปจะใช้อินสแตนท์ไอเด็นติไฟเออร์ที่ประกอบไปด้วยค่าของ อ็อบเจ็กไอเด็นติไฟเออร์ (Object Identifier) แล้

วปิดท้ายด้วย 0 เช่น การอ้างถึงค่าในอ็อบเจ็ก sysDescr ด้วยค่าอินสแตนท์ไอเด็นติไฟเออร์คือ 1.3.6.1.2.1.1.1.0 ซึ่งก็ คือ จะประกอบด้วยค่าอ็อบเจ็กไอเด็นติไฟเออร์ของอ็อบเจ็ก sysDescr คือ 1.3.6.1.2.1.1.1 แล้วต่อท้ายด้วย 0 นอกจากนั้นยังอาจเขียนในรูปของชื่อได้ คือ iso.org.dod.internet.mgmt.mib-2.system.sysDescr.0 หรือเขียนสั้นๆเพียง sysDescr.0 ก็ได้

สำหรับการอ้างถึงค่าของอ็อบเจ็กในรูปแบบตาราง (Sequence of Object Value) นั้น เอสเอ็นเอ็มพีไม่อนุญาตให้อ้างอิงทั้งตารางได้ในคราวเดียว (เช่น ไม่สามารถอ้างเพียง udp เพื่อขอข้อมูลทั้งอาเรย์)การอ้างอิงจะต้องทำที่อ็อบเจ็กที่เป็นโหนดปลายเท่านั้น (Leaf object) หรือต้องเจาะจงถึงค่าในตารางนั้นเลย เช่น udpLocalAddress หรือ udpLocalPort และอ้างไปทีละค่า

ตัวอย่างเช่น เอเจนต์พร้อมที่จะรับยูดีพีเดทาแกรมทุกอินเทอร์เฟสสำหรับบริการ BOOTP (67) ,TFTP (69) และ SNMP (151) เอเจนต์จะจัดเก็บค่านี้ลงในมิบทรีที่โหนด udpTable จำนวน 3 ค่า ดังตารางที่ 2.4

udpLocalAddress	udpLocalPort
0.0.0.0	67
0.0.0.0	69
0.0.0.0	151

ตารางที่ 2.4 ตัวอย่างค่าใน udpTable

หาก สถานีจัดการ ต้องการถามว่าเอเจนต์พร้อมรับยูดีพีเดทาแกรมสำหรับบริการใดบ้างก็ให้ใช้คำสั่ง GetRequest หรือ GetNextRequest โดยระบุไอเด็นติไฟเออร์ที่อ้างถึง udpLocalAddress และ udpLocalPort ดังตารางที่ 2.5

อ็อบเจ็กไอเด็นติไฟเออร์	อ็อบเจ็กไอเด็นติไฟเออร์ แบบย่อ	ค่า
1.3.6.1.2.1.7.5.1.1.0.0.0.0.67	udpLocalAddress.0.0.0.0.67	0.0.0.0
1.3.6.1.2.1.7.5.1.1.0.0.0.0.69	udpLocalAddress.0.0.0.0.69	0.0.0.0
1.3.6.1.2.1.7.5.1.1.0.0.0.0.161	udpLocalAddress.0.0.0.0.161	0.0.0.0
1.3.6.1.2.1.7.5.1.1.0.0.0.0.67	udpLocalPort.0.0.0.0.67	67
1.3.6.1.2.1.7.5.1.1.0.0.0.0.69	udpLocalPort.0.0.0.0.69	69
1.3.6.1.2.1.7.5.1.1.0.0.0.0.161	udpLocalPort.0.0.0.0.161	161

ตารางที่ 2.5 อ็อบเจ็กไอเด็นติไฟเออร์อ้างอิงค่าใน udpTable

จะสังเกตว่าการอ้างอิงค่าด้วยอ็อบเจ็กไอเด็นติไฟเออร์ไม่ได้ใช้ดัชนีชี้ตำแหน่งเหมือนการอ้างอาเรย์ในภาษาคอมพิวเตอร์ หากแต่ใช้ "ค่าข้อมูล" ที่อยู่ในตารางมาเป็นตัวกำหนดดัชนี (Index)

ในขั้นตอนแรกของการอ้างอิงค่าจะใช้ไอเด็นติไฟเออร์ udpTable หาจุดเริ่มต้นของตาราง ก่อนด้วยคำสั่ง GetNextRequest (udpTable) เพื่อให้ได้ข้อมูลค่าแรก จากนั้นจึงนำค่าที่ได้มาใช้เป็นไอเด็นติไฟเออร์สำหรับค่าถัดไปด้วยคำสั่ง GetNextRequest (udpLocalAddress.0.0.0.0.67) และทำซ้ำเช่นนี้จนกระทั่งได้ค่าสุดท้ายในตาราง จุดสิ้นสุดของตารางก็ตรวจได้จากไอเด็นติไฟเออร์ที่เป็นชื่อใหม่

คำสั่ง GetNextRequest จะนำค่าถัดไปมาโดยไม่ต้องเจาะจงค่าโดยอำนวยความสะดวกอย่างมากต่อตัวแปรที่เป็นชนิด sequence และ sequence of แต่ลำดับการนำค่าของ GetNextRequest จะมีลำดับเริ่มที่คอลัมน์แรกจนสิ้นสุดทุกแถวก่อนที่จะขึ้นคอลัมน์ถัดไป ดังนั้นการใช้จาก GetNextRequest ในกรณีของค่าในรูปที่ จะมีลำดับของค่าที่ได้ดังตารางที่ 2.6

ลำดับคำสั่งสอบถามด้วย GetNextRequest	ค่าที่ได้
udpTable	udpLocalAddress.0.0.0.0.67=0.0.0.0
udpLocalAddress.0.0.0.0.67	udpLocalAddress.0.0.0.0.69=0.0.0.0
udpLocalAddress.0.0.0.0.69	udpLocalAddress.0.0.0.0.161=0.0.0.0
udpLocalAddress.0.0.0.0.161	udpLocalPort.0.0.0.0.67=67
udpLocalPort.0.0.0.0.67	udpLocalPort.0.0.0.0.69=69
udpLocalPort.0.0.0.0.69	udpLocalPort.0.0.0.0.161=161
udpLocalPort.0.0.0.0.161	egpInMsgs.0=161

ตารางที่ 2.6 การสอบถามค่าจากตารางด้วยGetNextRequest ตามลำดับคอลัมน์ก่อน

2.2.4 การแทนข้อมูลด้วย ASN.1

ไอเอสโอและซีซีไอทีทีกำหนดวิธีการนิยามชนิดของตัวแปรโดยใช้ไวยากรณ์ ASN.1 (Abstract Syntax Notation One) ซึ่งเป็นเป็นเสมือนภาษาอธิบายแบบข้อมูลที่ไม่ขึ้นกับฮาร์ดแวร์ต้นกำเนิดของ ASN.1 นำมาใช้นิยามชุดโปรโตคอลของไอเอสโอ แต่สำหรับเอสเอ็นเอ็มพีจำใช้เพียงไวยากรณ์เพียงบางส่วนของ ASN.1 เท่านั้น

การใช้ ASN.1 ช่วยให้ผู้นำมาตรฐานไปใช้เข้าใจถึงสิ่งที่ผู้สร้างมาตรฐานกำหนดไว้โดยไม่เกิด ความกำกวมในเรื่องของความหมายและการแทนข้อมูล เช่นการกำหนดตัวแปรชนิด integer จะต้องกำหนดให้แน่นอนว่ามีค่าอยู่ในช่วงใด เพราะว่าคอมพิวเตอร์ต่างชนิดกันอาจมีความแตกต่างกันในการแทนข้อมูลและช่วงของตัวแปรที่แตกต่างกัน มิบอ็อบเจ็กในเอสเอ็นเอ็มพีมีรูปแบบที่กำหนดด้วย ASN.1ตัวอย่างต่อไปนี้เป็นนิยามของอ็อบเจ็ก sysContact

```
sysContact OBJECT-TYPE
        SYNTAX  DisplayString (SIZE (0..255))
        ACCESS  read-write
```

```
STATUS  mandatory
DESCRIPTION
        "The textual identification of the contact person
        for this managed node, together with information
        on how to contact this person."
::= { system 4 }
```

นิยามข้างต้นมีความหมายดังนี้

- SYNTAX : กำหนดแบบข้อมูลของตัวแปรเช่นจำนวนเต็มหรืออักขระ ในที่นี้คือ DisplayString
- ACCESS : แบบการเข้าใช้ซึ่งอาจเป็น read-only(อ่านอย่างเดียว), read-write(อ่านเขียนได้),write-only(เขียนอย่างเดียว) หรือ not-accessible (ห้ามเข้าถึง) เป็นต้น
- STATUS : กำหนดสถานะของตัวแปรว่าจำเป็นต้องมีตัวแปรนี้หรือไม่ ค่าที่เป็นไปได้เช่น mandatory(จำเป็น),optional(ออพชั่นซึ่งมีหรือไม่ก็ได้), depreciate (จำเป็นต้องมีแต่อาจยกเลิกในรุ่นถัดไป), obsoleted (ไม่จำเป็นเนื่องจากยกเลิกไม่ใช้แล้ว)
- DESCRIPTION : ข้อความอธิบายตัวแปร
- บรรทัดสุดท้ายของนิยามกำหนดว่าตัวแปร sysContact จะเชื่อมกับโครงสร้างต้นไม้ต่อจากโหนด system และมีค่าเท่ากับ 4 ซึ่งแสดงถึงไอเด็นติไฟเออร์ประจำ sysContact คือiso.org.dod.internet.mgmt.mib-2.system.4 หรือ 1.3.6.1.2.1.1.4

อีกตัวอย่างหนึ่งต่อไปนี้แสดง โครงสร้างมิบบางส่วนของกลุ่มภายใต้โหนด mib-2 โปรดสังเกตว่าสัญลักษณ์ "--" ที่ปรากฏในนิยามใช้เป็นส่วนหมายเหตุแสดงคำอธิบาย

```
-- groups in MIB-II
        system      OBJECT IDENTIFIER ::= { mib-2 1 }
        interfaces  OBJECT IDENTIFIER ::= { mib-2 2 }
        at          OBJECT IDENTIFIER ::= { mib-2 3 }
        ip          OBJECT IDENTIFIER ::= { mib-2 4 }
        icmp        OBJECT IDENTIFIER ::= { mib-2 5 }
        tcp         OBJECT IDENTIFIER ::= { mib-2 6 }
        udp         OBJECT IDENTIFIER ::= { mib-2 7 }
        egp         OBJECT IDENTIFIER ::= { mib-2 8 }
```

```
        -- historical (some say hysterical)
        -- cmot     OBJECT IDENTIFIER ::= { mib-2 9 }
        transmission OBJECT IDENTIFIER ::= { mib-2 10 }
        snmp        OBJECT IDENTIFIER ::= { mib-2 11 }
        -- the System group
        -- Implementation of the System group is mandatory for all
        -- systems.  If an agent is not configured to have a value
        -- for any of these variables, a string of length 0 is
        -- returned.
sysDescr OBJECT-TYPE
        SYNTAX  DisplayString (SIZE (0..255))
        ACCESS  read-only
        STATUS  mandatory
DESCRIPTION
            "A textual description of the entity.  This value
            should include the full name and version
            identification of the system's hardware type,
            software operating-system, and networking
            software.  It is mandatory that this only contain
            printable ASCII characters."
        ::= { system 1 }
```

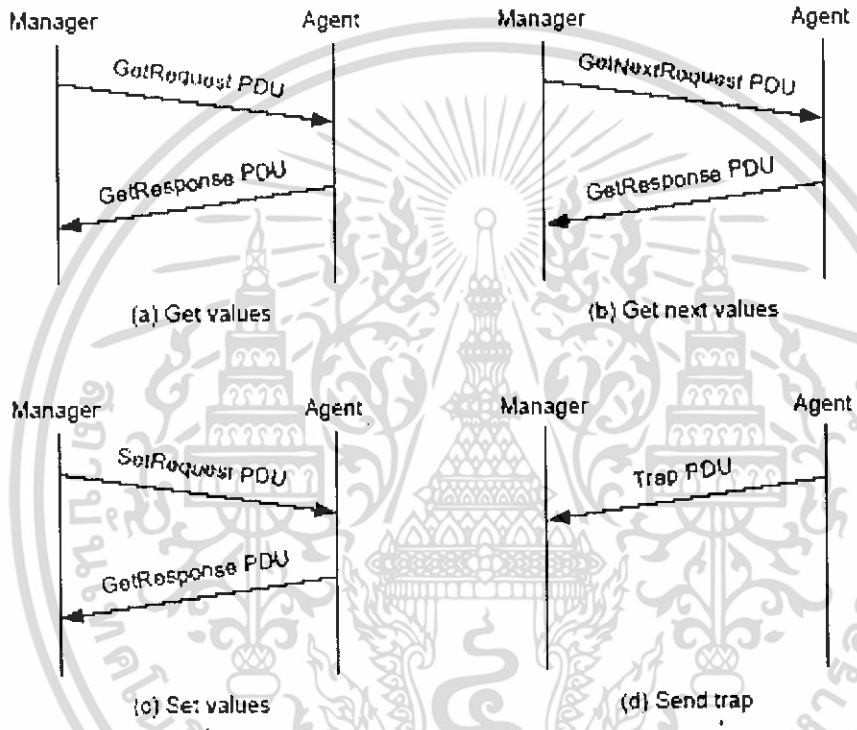
2.2.5 ลักษณะของโปรโตคอล *(Protocol Specification)*

การติดต่อระหว่างสถานีจัดการกับเอเจนต์มีรูปแบบในการติดต่อที่เรียกว่า protocol data units หรือ พีดียู (PDU) หลายรูปแบบด้วยกันตามวัตถุประสงค์ในการติดต่อ แบบของการติดต่อในเอสเอ็นเอ็มพีรุ่น 1 มี 5 แบบคือ

1. GetRequest PDU ใช้สอบถามข้อมูลจากตัวเอเจนต์ที่อยู่บนอุปกรณ์ที่ต้องการตรวจสอบในระบบเครือข่าย
2. GetNextRequest PDU ใช้สอบถามข้อมูลที่เรียงเป็นลำดับ เช่น ข้อมูลที่เก็บอยู่ในรูปตาราง หรือในกรณีที่ไม่ทราบชื่อตัวแปรที่แน่ชัด
3. GetResponse PDU เอเจนต์ส่งคำตอบกลับมายังผู้สอบถาม
4. SetRequest PDU ใช้เปลี่ยนแปลงค่าของอ็อบเจ็กที่เอเจนต์รับผิดชอบอยู่

5. Trap PDU ใช้แจ้งเหตุการณ์ที่เกิดขึ้นในระบบเครือข่าย เช่นการเริ่มต้นทำงานใหม่ของอุปกรณ์ หรือเส้นทางขัดข้อง

เอสเอ็นเอ็มพีอาศัยโปรโตคอลยูดีพี โดยเอเจนต์จะใช้พอร์ตหมายเลข 161 สำหรับการติดต่อในแบบที่ 1 ถึง 4 จากสถานีจัดการเครือข่ายและสถานีจัดการเครือข่ายจะใช้พอร์ตหมายเลข 162 สำหรับการติดต่อในแบบที่ 5 จากเอเจนต์ โดยจะมีลำดับการทำงานในการติดต่อทั้ง 5 รูปแบบดังรูปที่ 2.7



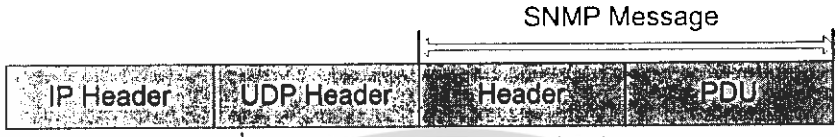
รูปที่ 2.8 ลำดับการทำงานของ SNMP PDU

ส่วนเอสเอ็นเอ็มพี รุ่น 2 ได้รับการปรับปรุงให้มีความสามารถเพิ่มขึ้นจากรุ่นแรก ฟังก์ชันที่เพิ่มขึ้นได้แก่ คำสั่ง GetBulkRequest เพื่อสอบถามค่าโดยกำหนดจำนวนอ็อบเจ็กที่ต้องการได้แทนที่จะต้องใช้ GetNextRequest ซ้ำหลายครั้ง นอกจากนี้ ยังมี InformRequest สำหรับการสื่อสารระหว่างสถานีการจัดการเครือข่าย เพื่อช่วยในการบริหารแบบกระจายภาระงาน จุดเด่นอีกส่วนหนึ่งที่มีในเอสเอ็นเอ็มพี รุ่น 2 ได้แก่ การรักษาความปลอดภัยโดยการเข้ารหัสข้อความ และการพิสูจน์ตัวจริง รวมทั้งการขยายกลุ่ม MIB เพิ่มเติม [RFC 1907]

การเอ็นแคปซูเลต (Encapsulation)

การเอ็นแคปซูเลตคำสั่งและข้อมูลในเอสเอ็นเอ็มพีมีวิธีตามรูปที่ 2.8 ฟอร์แมตของเอสเอ็นเอ็มพี ประกอบด้วย 2 ส่วนคือ เฮดเดอร์และพีดียู เฮดเดอร์ประกอบด้วยฟิลด์ย่อยสองฟิลด์คือ

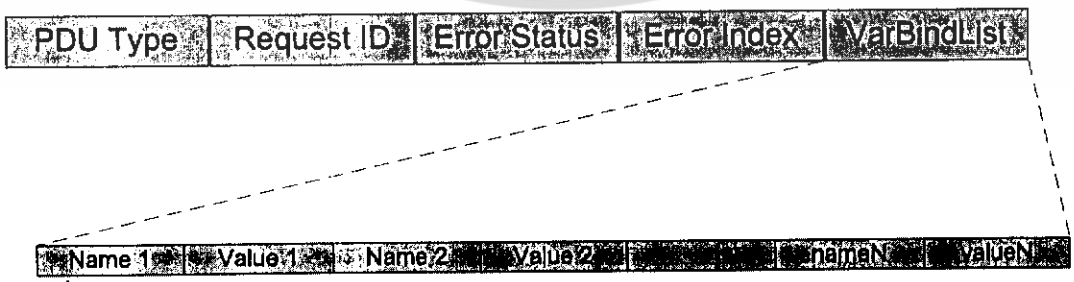
1. Version รุ่นของโพรโตคอลที่ใช้ ถ้าเป็นโพรโตคอลรุ่น 1 จะมีค่า 0 หากเป็นรุ่น 2 จะมีค่า 1
2. Community รหัสผ่านในรูปสายอักขระเพื่อให้เอเจนต์ใช้ในการพิสูจน์ตัวตนว่าข้อความที่ส่งมามีสิทธิ์ในการสอบถาม หรือเปลี่ยนแปลงข้อมูลหรือไม่ ซึ่งรายละเอียดได้อธิบายไว้ในหัวข้อ Communities และ Community Names ที่ผ่านมา



รูปที่ 2.9 การเอ็นแคปซูเลตเอสเอ็นเอ็มพี

ในส่วนของพีดียูประกอบด้วยฟิลด์ย่อยตามชนิดของข้อความ หากเป็นข้อความ GetRequest PDU, GetNextRequest PDU, GetResponse PDU และ SetRequest PDU จะมีโครงสร้างเดียวกัน รูปที่2.9 แสดงโครงสร้างของพีดียูโดยแต่ละฟิลด์มีความหมายดังนี้

- PDU type รูปแบบการติดต่อ (1 ถึง 5)
- Request ID กำหนดบอกหมายเลขข้อความเพื่อใช้จับคู่เมื่อรับคำตอบกลับมา
- Error Status สถานะความผิดพลาดที่เกิดขึ้นโดยรหัสความผิดพลาดและสถานะผิดพลาดที่ใช้ในเอสเอ็นเอ็มพีจะแสดงในตารางที่ 3 สำหรับข้อความGetRequest PDU, GetNextRequest PDU และ SetRequest PDU จะมีค่าในฟิลด์นี้เป็น 0 เสมอ
- Error Index ดรรชนีชี้ค่าผิดพลาดที่เกิดขึ้นเกิดจากตัวแปรตัวลำดับที่เท่าไหร่ของตัวแปรทั้งหมดที่สอบถามไปสำหรับข้อความ GetRequest PDU,GetNextRequest PDU และ SetRequest PDU จะมีค่าในฟิลด์นี้เป็น 0เสมอ
- VarBindList ค่าผูกพันตัวแปร(variable binding) แสดงอยู่ในรูปของการอ้างอิงตัวแปรหรืออ็อบเจ็ก (name) และค่าของตัวแปร (value) ต่อเนื่องกันไปเป็นรายการสำหรับข้อความ GetRequest PDU, GetNextRequest PDU และSetRequest PDU ค่าของตัวแปรจะมีค่าเป็น NULL เสมอ



รูปที่ 2.10 แสดงโครงสร้างของพีดียู GetRequest PDU, GetNextRequest PDU, GetResponse PDU และ SetRequest PDU

รหัสผิดพลาด	ชื่อ	คำอธิบาย
0	noError	ไม่มีข้อผิดพลาด
1	tooBig	เอเจนต์ไม่สามารถส่งคำตอบได้ในเฟรมเดียว
2	noSuchName	ไม่มีตัวอ็อบเจ็กที่ต้องการสอบถามอยู่ในฐานข้อมูล
3	badValue	ค่าที่กำหนดให้อ็อบเจ็กไม่ถูกต้อง
4	readOnly	เปลี่ยนค่าอ็อบเจ็กไม่ได้เพราะอ่านค่าได้เพียงอย่างเดียว
20	genErr	มีข้อผิดพลาดอื่นๆเกิดขึ้น

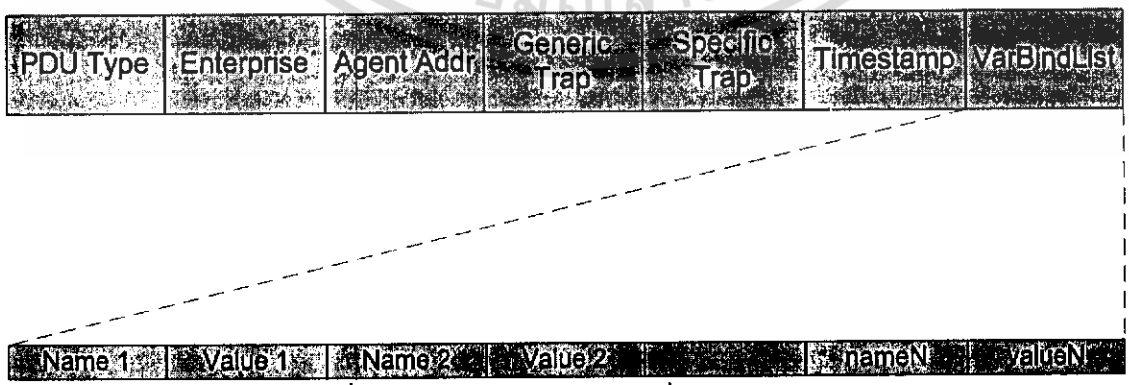
ตารางที่ 2.7 รหัสและสถานะความผิดพลาดในเอสเอ็นเอ็มพี

สำหรับข้อความ Trap PDU มีลักษณะแตกต่างกันออกไปดังรูปที่ 2.10 โดยแต่ละฟิลด์มีความหมายดังนี้

- Enterprise ชนิดของตัวแปรที่สร้าง Trap นี้ขึ้นมา โดยค่าในฟิลด์นี้จะอ้างอิงกับค่าในอ็อบเจ็ก sysObjectID
- Agent Addr ค่า IP Address ของอ็อบเจ็กที่สร้าง Trap นี้ขึ้นมา
- Generic Trap ชนิดของ Trap ที่เกิดขึ้นโดยชนิดของ Trap และรหัสของ Trap ที่ใช้ในเอสเอ็นเอ็มพีจะแสดงในตารางที่ 2.8
- Specific Trap ชนิดของเหตุการณ์ผิดปกติเกิดขึ้นกับ enterprise-specific
- Time-stamp เวลาที่ใช้ไปทั้งหมดตั้งแต่การเริ่มการติดต่อในเครือข่ายรวมถึงเวลาที่ใช้ในการสร้าง Trap โดยค่าดังกล่าวจะเก็บอยู่ในอ็อบเจ็ก sysUpTime

รหัส Trap	ชื่อ	คำอธิบาย
0	coldStart	มีการเริ่มต้นการทำงานใหม่ ด้วยสาเหตุของการเปลี่ยนแปลงค่าในเอเจนต์ หรือมีการเปลี่ยนแปลงโปรโตคอลเอ็นจิน เช่น การ restart อุปกรณ์หลังจากเกิดการล้มเหลวของระบบ (crash)
1	warmStart	มีการเริ่มต้นการทำงานใหม่ แต่ไม่ได้เกิดจากสาเหตุของการเปลี่ยนแปลงค่าในเอเจนต์ หรือโปรโตคอลเอ็นจิน เช่น การ restart routine
2	linkDown	การเชื่อมต่อของเอเจนต์มีปัญหา จะมีการค่าของอ็อบเจ็ก ifIndex เพื่อบอกจุดเชื่อมต่อ (interface) ที่มีปัญหา
3	linkUp	การเชื่อมต่อของเอเจนต์กลับมาใช้งานได้ จะมีการค่าของอ็อบเจ็ก ifIndex เพื่อบอกจุดเชื่อมต่อ (interface) ที่มีการกลับมาใช้งานได้
4	authenticationFailure	การพิสูจน์ตัวตนของ message ที่เข้ามาไม่ผ่านหรือมีการผิดพลาดเกิดขึ้น
5	egpNeighborLoss	โปรโตคอลเกตเวย์ภายนอก (External gateway protocol) มีปัญหา
6	enterproseSpecific	มีเหตุการณ์ผิดปกติเกิดขึ้นกับ enterprise-specific โดยจะมีการประกาศชนิดของเหตุการณ์ผิดปกติเกิดขึ้นในฟิลด์ specific-trap

ตารางที่ 2.8 รหัส และชนิดของ Trap ในเอสเอ็นเอ็มพี



รูปที่ 2.11 โครงสร้างพีดียูของคำสั่ง Trap PDU

โปรดสังเกตว่าในที่นี้ไม่ได้กล่าวขนาดความยาวของแต่ละฟิลด์ เพราะทุกฟิลด์ในเอสเอ็นเอ็มพีต้องเข้ารหัสและจะได้ขนาดของแต่ละฟิลด์ที่มีความยาวแตกต่างกันไปตามชนิดข้อมูล

กลไกในการส่ง SNMP Message (Transmission of an SNMP Message)

กลไกในการส่ง SNMP Message ของ PDU ทั้ง 5 แบบในส่วนของโปรโตคอลเอ็นจิน โดยการทำงานจะมีขั้นตอนดังต่อไปนี้

1. สร้าง PDU ตามโครงสร้างที่กำหนดไว้ใน ASN.1 (Abstract Syntex Notation One)
2. PDU ที่สร้างในขั้นตอนที่หนึ่งรวมค่า transport addresses ของต้นทางและปลายทาง และCommunity name จะถูกส่งไปให้ส่วนของการบริการพิสูจน์ตัวตน (Authentication service)ซึ่งจะทำการปรับเปลี่ยนข้อมูลที่จำเป็นในการแลกเปลี่ยน เช่น การเข้ารหัสลับ (encryption) หรือการสร้างโค้ดที่ใช้ในการพิสูจน์ตัวตน หลังจากการปรับเปลี่ยนเสร็จก็จะคืนค่ากลับมายังโปรโตคอลเอ็นจิน
3. โปรโตคอลเอ็นจินจะสร้าง message ของเอสเอ็นเอ็มพีโดยการรวมฟิลด์รุ่นของโพรโตคอล(Version), Community name, และผลลัพธ์ที่ได้ในขั้นตอนที่ 2
4. ทำการเข้ารหัส message เอสเอ็นเอ็มพีที่ได้จากขั้นตอนที่ 3 โดยวิธีการ Basic Encoding Rule (BER) แล้วส่ง message ที่เข้ารหัสแล้วไปยังโปรโตคอลในชั้น Transport ต่อไป

กลไกในการรับ SNMP Message ของ PDU ในส่วนของโปรโตคอลเอ็นจิน โดยการทำงานจะมีขั้นตอนดังต่อไปนี้

1. ตรวจสอบความถูกต้องทางไวยากรณ์ขั้นพื้นฐานของ message โดยจะกำจัด message ทิ้งถ้าพบความผิดพลาด
2. รุ่นของโพรโตคอล (Version) ที่ใช้ โดยจะกำจัด message ทิ้งถ้าพบว่าเป็นคนละรุ่นกัน
3. โปรโตคอลเอ็นจินจะนำค่าของ use name, ส่วนของ PDU และ transport addresses ของต้นทางและปลายทาง ส่งไปให้ส่วนของการบริการพิสูจน์ตัวตน (Authentication service)

- ถ้าการพิสูจน์ตัวตนล้มเหลว ส่วนของการบริการพิสูจน์ตัวตนจะส่งสัญญาณบอกไปยังโปรโตคอลเอ็นจินเพื่อให้ทำการสร้าง Trap และจะกำจัด message นั้นทิ้ง
- ถ้าการพิสูจน์ตัวตนสำเร็จ ส่วนของการบริการพิสูจน์ตัวตนจะคืนค่า PDU ที่อยู่ในโครงสร้างของ ASN.1 มาให้กับโปรโตคอลเอ็นจิน

4. ตรวจสอบความถูกต้องทางไวยากรณ์ขั้นพื้นฐานของ PDU โดยจะกำจัด PDU ทิ้งถ้าพบความผิดพลาด หลังจากนั้นจะนำชื่อของ Community name เพื่อจัดสรร

รูปแบบนโยบายการเข้าถึง (Access policy) ตาม Community ของ PDU นั้น และทำการประมวลผลตามคำสั่งใน PDU ต่อไป

2.2.6 การเข้ารหัสโดยใช้ BER (Basic Encoding Rules)

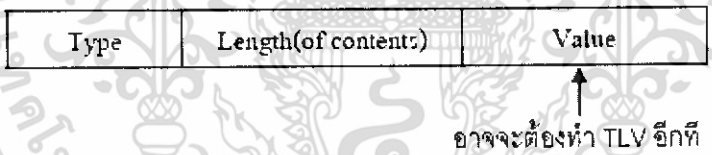
การส่งข้อมูลในSNMP ไม่ได้ส่งในรูปแบบเป็น ออกเทตของของตัวเลขโดยตรง หากแต่ฝ่ายส่งต้องเข้ารหัสข้อมูลเพื่อนำส่งข้อมูลออก และ ถอดรหัสที่ฝ่ายรับ

โครงสร้างการเข้ารหัส

การเข้ารหัสตามแบบ BER จะมีข่าวสารกำกับอยู่ในตัวว่าเป็นข้อมูลชนิดใด และ มีความยาวเท่าใด ฟิลด์ที่ผ่านการเข้ารหัสแล้วประกอบด้วยค่า 3 ส่วน ดังรูปที่ 2.11 คือ

- ชนิดของข้อมูล (type หรือ tag หรือ identifier)
- ความยาวของข้อมูล (length)
- ตัวข้อมูล (value หรือ contents)

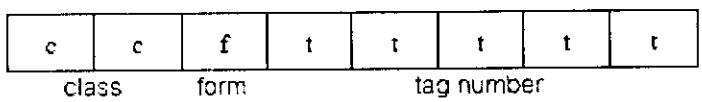
โครงสร้างรหัสเหล่านี้ เรียกว่า โครงสร้าง TLV (TLV : type-length-value structure) ค่าในส่วนของฟิลด์ value อาจจะต้องผ่านการเข้ารหัสค่าตามโครงสร้าง TLV ด้วยเช่นกัน



รูปที่ 2.12 โครงสร้างของ TLV

ฟิลด์กำหนดชนิดข้อมูล (Type)

การกำหนดค่าให้กับฟิลด์ Type ขนาด 8 bit มีการจัดวางตำแหน่งบิต เพื่อให้ได้ค่าตัวเลขที่ใช้แทนกลุ่มชนิดข้อมูล แบ่ง เป็น 3 ส่วนย่อย ดังรูปที่ 2.12

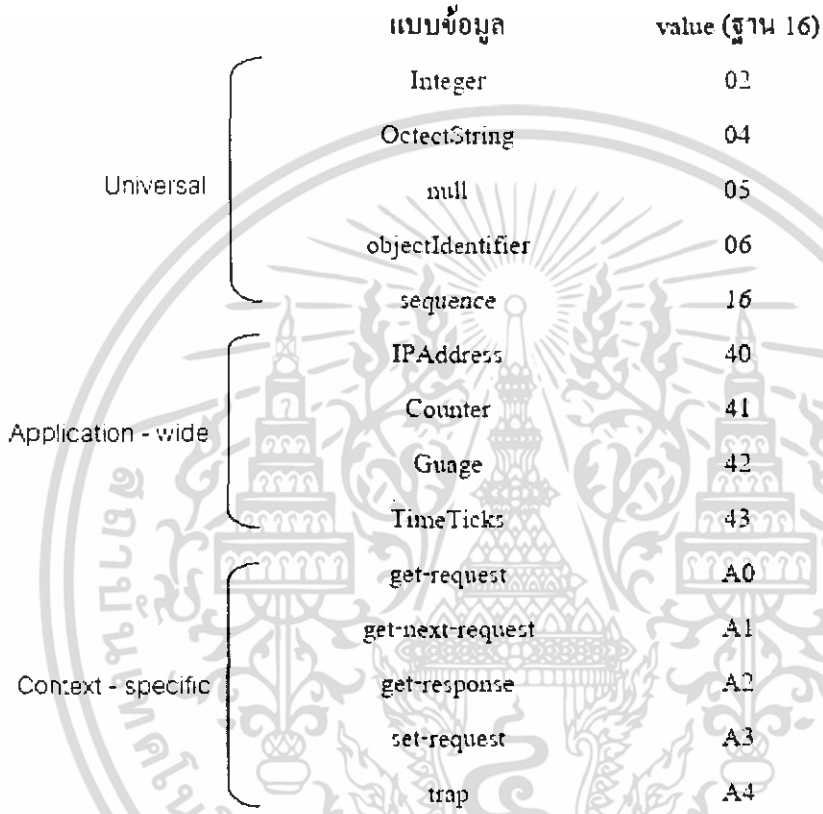


รูปที่ 2.13 รูปแบบการเข้ารหัสประเภทชนิดข้อมูล

1. ฟิลด์ class 2 bits แรก : กำหนดประเภทข้อมูล มี 4 แบบ คือ
   - 00 = ประเภท Universal เช่น ตัวเลขทั่วไป
   - 01 = ประเภท Application ข้อมูลสำหรับใช้กับ application program

- 10 = ประเภท Context Specific ข้อมูลเจาะจง เช่นคำสั่งใน SNMP
- 11 = ประเภท Private ข้อมูลสำหรับใช้กรณีเฉพาะ

2. บิตถัดมาคือฟิลด์ form กำหนดว่าแบบข้อมูลนั้นเป็นแบบ พื้นฐาน(Primitive) หรือ แบบโครงสร้าง(Constructed) เช่น ตัวอย่างแบบข้อมูลพื้นฐาน เช่น integer หรือ stringส่วนข้อมูลแบบโครงสร้าง ก็เช่น sequence
3. Tag number 5 bits สุดท้าย : เป็นส่วนกำหนดแบบข้อมูลซึ่งมีค่าได้ตั้งแต่ 0 ถึง 30



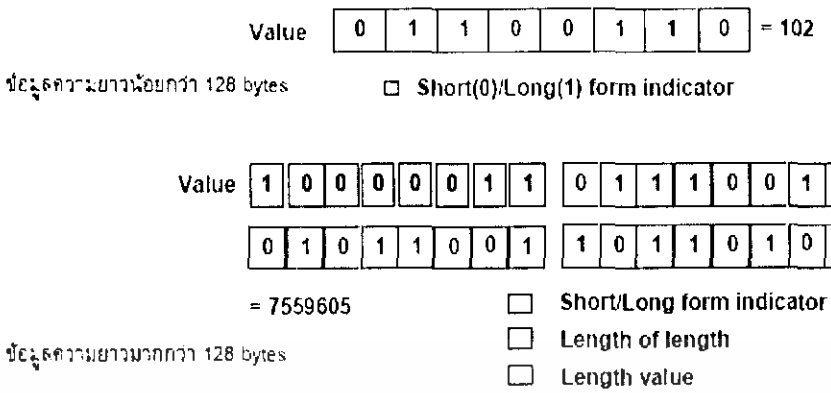
รูปที่ 2.14 รูปแบบข้อมูลที่ใช้ใน SNMP

ตัวอย่างเช่น ข้อมูล Integer จะมี type เท่ากับ 02 หรือแยกออกมาเป็น bit ได้เป็น 0000 0010 ซึ่งมาจาก 00(Universal) , 0(Primative) และ 0 0010(Tag Number)

ฟิลด์กำหนดความยาว (Length)

หากข้อมูลความยาวน้อยกว่า 128 bytes ฟิลด์นี้จะกินเนื้อที่เพียง 1 byte โดยมี bit ซ้ายสุดเป็น "0" และ 7 bit ที่เหลือกำหนดความยาว เช่น ข้อมูลยาว 10 byte ค่าในฟิลด์จะเท่ากับ 0x0A

หากข้อมูลมีความยาวตั้งแต่ 128 bytes ขึ้นไป ต้องใช้ฟิลด์ length หลายออกเทต โดยที่บิตซ้ายสุดจะมีค่าเป็น "1" และใช้ 7 bits ที่เหลือเป็นตัวนับจำนวนออกเทตที่กำหนดความยาว ถัดจากนั้นจึงเป็นไบท์กำหนดความยาว เช่น ข้อมูลยาว 1000 bytes (03 E8) ค่าในฟิลด์นี้จะเป็น 82 03 E8 โดยที่ 7 bits ขวาของB2 คือ 000 0010 ซึ่งเท่ากับ 2 หมายถึงใช้ 2 bytes กำหนดความยาว และ 2 bytes นั้นคือ 03 E8



รูปที่ 2.15 แสดงตัวอย่างการเข้ารหัสความยาว

ฟิลด์กำหนดข้อมูล (Value)

การเข้ารหัสฟิลด์ content จะแตกต่างกันไปตามชนิดของข้อมูล ดังนี้

รูปแบบการเข้ารหัส TLV (เฉพาะบางประเภทข้อมูลที่ใช้ใน SNMP)

1. ข้อมูลประเภทสายอักขระ ไม่ต้องเข้ารหัส ส่งไปตามลำดับของสายอักขระตามปกติ เช่น สายอักขระ "interfaces" จะอยู่ในรหัสTLV ดังนี้ 04 0A 'i' 'n' 't' 'e' 'r' 'f' 'a' 'c' 'e' 's' 04 คือ OctectString และ 0A คือความยาว
2. ข้อมูลตัวเลข

- ค่าไม่เกิน 127 จะใช้เพียง 1 byte เท่านั้น
- ค่าเกิน 127 จะต้องผ่านการเข้ารหัสอีกที คือ set bit ซ้ายสุดให้เป็น 1 และใช้ 7 bit ที่เหลือกำหนดจำนวน octet ที่ต้องใช้ จากนั้นจึงค่อยตามด้วยค่า octet ถัดต่อๆไป เช่น ค่าตัวเลข 130 เมื่อเข้ารหัสแล้วจะได้ค่า 02 02 81 02 โดยที่ 02 คือ integer และ 81 02 แทนค่า 130

3. Null ให้ส่งโดยเพียงแต่กำหนดค่าในฟิลด์ length เป็น 0 และไม่ต้องส่งค่าใดๆไปทั้งสิ้น
4. ข้อมูลประเภท objectIdentifier ต้องเข้ารหัสด้วยวิธีพิเศษ

ตัวอย่างSNMP Frame และการเข้ารหัส

การส่งคำสั่ง request (GetRequest PDU)

ลำดับ Byte เมื่อใช้คำสั่ง GetRequest PDU สอบถามค่า 1.3.6.1.2.1.1.1.0 หรือ sysDescr.0 แต่ละฟิลด์ของSNMP จะผ่านการเข้ารหัสตามโครงสร้างTLV

UDP	30	27	02	01	00	04	06	73	23	6E
	6D	70	21	A0	1A	02	02	22	FB	02
	01	00	02	01	00	30	0E	30	0C	06
	08	2B	06	01	02	01	01	01	00	05
	00									

รูปที่ 2.16 SNMP Frame ของ GetRequest PDU

									Type	Length	value	หมายเหตุ
30	27								sequence	39	-	มีข้อมูล 39 bytes ตามมา
02	01	00							integer	1	0	version = 1
04	06	73	23	6E	6D	70	21		string	6	#snmp!	community = #snmp!
A0	1A								context-spc.	26	-	get request 26 bytes
02	02	22	FB						integer	2	22FB	id = 22FB
02	01	00							integer	1	0	error status = 0
02	01	00							integer	1	0	error status = 0
30	0E								sequence	14	-	
30	0C								sequence	12	-	
06	08	2B	06	01	02	01	01	01	00	oid	8	sysDescr.0	1.3.6.1.2.1.1.1.0
05	00								null	0	0	รหัสปิดท้าย

รูปที่ 2.17 ความหมายของการเข้ารหัสข้อมูลของ GetRequest PDU

การตอบกลับคำสั่ง request (GetResponse PDU)

ลำดับ byte เมื่อ agent ใช้คำสั่ง GetResponse PDU ตอบค่า sysDescr.0 ส่งกลับไป แต่ละฟิลด์ที่ผ่านการเข้ารหัสตามโครงสร้าง TLV จะแสดงได้ดังนี้ (เนื่องจากสายอักขระที่ตอบกลับมามีความยาวมากกว่า 300 byte ดังนั้นจึงเลือกแสดงแค่บางส่วนเท่านั้น)

UDP	30	81	F9	02	01	00	04	06	73	23
	6E	6D	70	21	A2	81	EB	02	02	02
	FB	02	01	00	02	01	00	30	81	DE
	30	81	DB	06	08	2B	06	01	02	01
	01	01	04	81	CE	43	65	73	63	6F

รูปที่ 2.18 SNMP Frame ของ GetResponse PDU

30	81	F9								
02	01	00								
04	06	73	23	6E	6D	70	21			
A2	81	EB								
02	02	22	FB							
02	01	00								
02	01	00								
30	81	DE								
30	81	DB								
06	08	2B	06	01	02	01	01	01	00	
04	81	CE								
43	65	73	63	6F						

Type	Length	value	หมายเหตุ
sequence	377	-	มีข้อมูล 377 bytes ตามมา
integer	1	0	version = 1
string	6	#snmp!	community = #snmp!
context-spc.	363	-	get response 363 bytes
integer	2	22FB	id = 22FB
integer	1	0	error status = 0
integer	1	0	error index = 0
sequence	14	-	
sequence	12	-	
oid	8	sysDescr.0	1.3.6.1.2.1.1.1.0
OctectString	334	-	
			Cisco.....

รูปที่ 2.19 ความหมายของการเข้ารหัสข้อมูลของ GetResponse PDU

บทที่ 3
การออกแบบและพัฒนาโปรแกรม

3.1 รายละเอียดของการพัฒนา

ในการจัดทำโปรแกรมสำรวจเครือข่าย และระบบคอมพิวเตอร์จำเป็นต้องศึกษาทฤษฎี และข้อมูลที่ใช้ในการค้นหา และรวบรวมข้อมูลของคอมพิวเตอร์เครื่องอื่นที่อยู่ในเครือข่าย โดยในโปรแกรมของเราได้ใช้เทคนิคหลายๆอย่าง ดังนี้

- ใช้เทคนิค Port scanning (TCP connect scan) เพื่อค้นหาพอร์ตที่ต้องการบนเครื่องปลายทาง และยังนำมาใช้ในการค้นหาว่ามีเครื่องใดเปิดใช้งานอยู่บ้าง
- ใช้วิธีการของ SNMP เพื่อรวบรวมข้อมูลการใช้งานระบบเครือข่าย ซึ่งเมื่อนำมารวมกับข้อมูลของเครื่องปลายทางที่ได้รับมาจะทำให้สามารถพัฒนาโปรแกรมที่มีความสามารถในการวิเคราะห์ปริมาณการใช้งานระบบเครือข่ายในด้านต่างๆได้ในระดับนึง

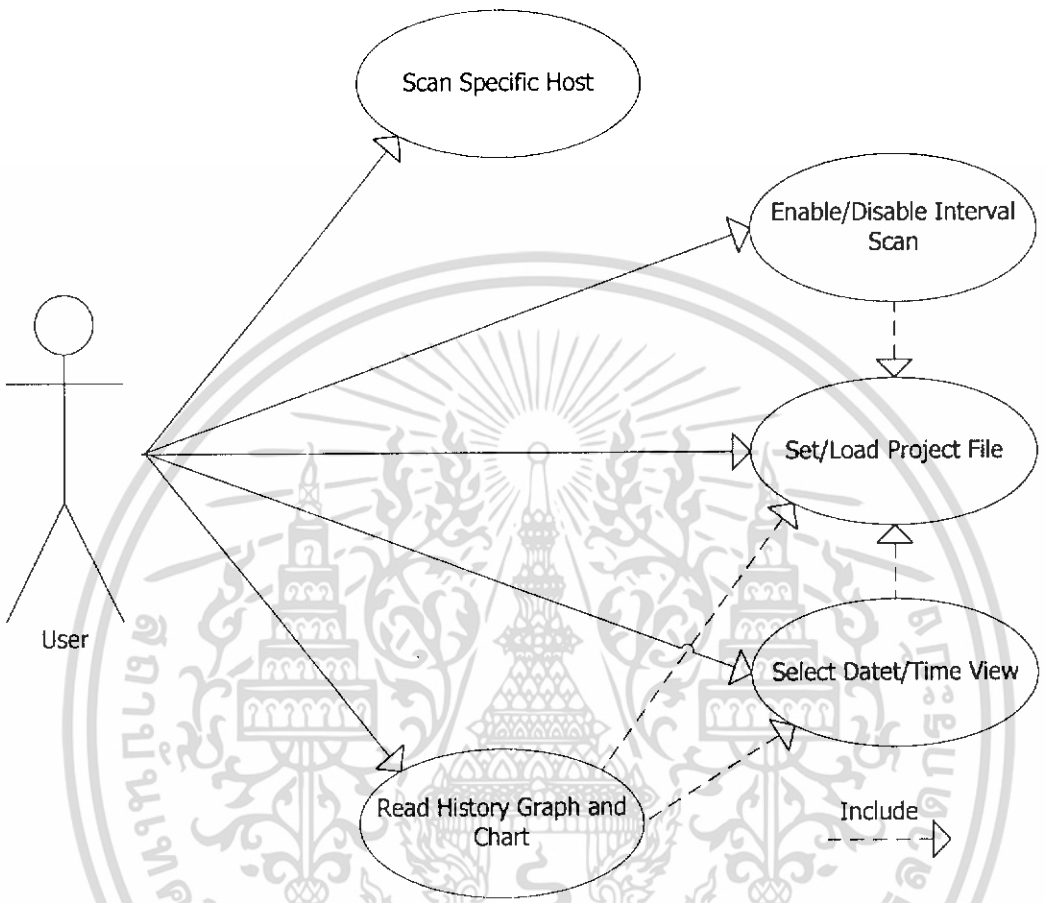
เครื่องมือต่าง ๆ ที่ใช้ในการพัฒนา

ได้แก่

- Microsoft Visual Studio .NET 2003
- Winpcap 3.1
- Nmap 3.95
- SNMP++.NET v. 1.16
- Chart Director for .NET v.4.0
- Microsoft Windows 2000 หรือ XP

3.2 การออกแบบโครงสร้างของโปรแกรม

Use case Diagram

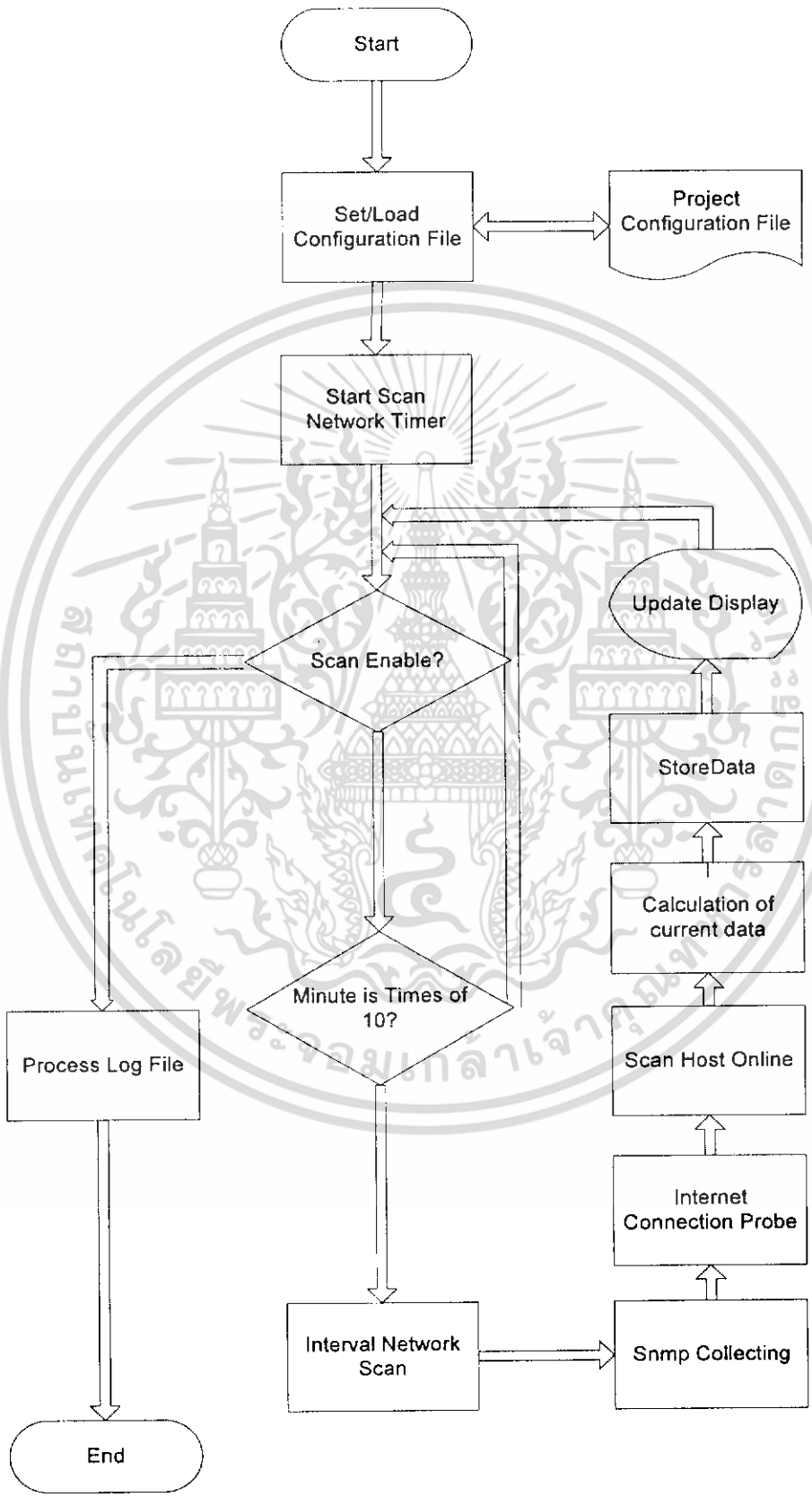


รูปที่ 3.1 Use case Diagram ของโปรแกรม

Set/Load Project File	ทำการตั้งค่า Project File ต่างๆเพื่อเป็นข้อมูลอ้างอิงการทำงาน
Enable / Disable interval scan	สั่งให้โปรแกรมเริ่มทำการแสกนแบบเป็นช่วงเวลาเพื่อเก็บข้อมูล
Scan Specific Host	สั่งให้โปรแกรมสแกนเครื่องปลายทางที่สนใจ
Select Date/Time View	เลือกวันที่และเวลาที่จะทำการเขียน Graph และ Chart
Read History Graph and Chart	อ่านข้อมูล Graph และ Chart ข้อมูลต่างๆที่ได้จากการแสกนและคำนวณ

ตารางที่ 3.1 แสดงคำอธิบาย Use case Diagram ของโปรแกรม

Flow Chart



รูปที่ 3.2 Flow Chart Diagram ของส่วนการสแกนระบบเครือข่ายโปรแกรม

Start	เริ่มต้นโปรแกรม
Set / Load Configuration File	ตั้งค่าเริ่มต้นการทำงาน หรือนำค่าเริ่มต้นที่มีอยู่แล้วมาใช้งาน
Project Configuration File	ไฟล์ซึ่งมีค่าเริ่มต้นการทำงานเก็บเอาไว้
Start Scan Network Timer	เริ่มจับเวลาในก่อนที่จะ scan
Scan Enable	สามารถทำการ scan ได้หรือไม่
Minute is Times of 10 ?	ครบ 10 นาทีแล้ว ใช่หรือไม่
Interval Network Scan	เริ่มทำการ scan
Snmp Collecting	รวบรวมข้อมูล SNMP
Internet Connection Probe	สำรวจสถานะการเชื่อมต่อระหว่างระบบเครือข่ายกับ Internet
Scan Host Online	ตรวจหาเครื่องคอมพิวเตอร์ที่เปิดใช้งานภายในเครือข่าย
Calculation of current data	ทำการคำนวณข้อมูลที่มีอยู่
Store Data	เก็บข้อมูลที่ได้มา
Update Display	แสดงผลลัพธ์ออกทางหน้าจอโปรแกรม
Process Log File	จัดเก็บ Log File
End	สิ้นสุดการทำงานของโปรแกรม

ตารางที่ 3.2 แสดงคำอธิบาย Flow Chart Diagram ของโปรแกรม

3.3 โครงสร้าง และการทำงานของโปรแกรม

โครงสร้างการทำงานของโปรแกรมจะประกอบด้วยส่วนประกอบที่สำคัญ ได้แก่

ส่วนติดต่อกับผู้ใช้ โดยในขั้นตอนนี้จะได้รับ input จากผู้ใช้ถึงรายละเอียดและขอบเขตของเครือข่ายย่อยที่ผู้ใช้ต้องการให้โปรแกรมสำรวจซึ่งแบ่งออกเป็น 3 ส่วนย่อยๆ คือ

1. รายละเอียดของอุปกรณ์เครือข่ายที่จะทำการสำรวจค่าภายใต้ Protocol SNMP
2. IP Range ของ เครือข่ายย่อยแต่ละเครือข่าย
3. รายชื่อของ URL หรือ IP ที่จะใช้เป็นตำแหน่งอ้างอิงในการสำรวจสถานะของการเชื่อมต่อกับ Internet

เมื่อได้รับรายละเอียดต่างๆครบแล้วก็จะทำการบันทึกไว้เป็น Project File ซึ่งจะนำไปใช้ในการทำงาน

ส่วนสำรวจระบบเครือข่าย ซึ่งแบ่งออกเป็น 4 ส่วนย่อยๆคือ

- การสำรวจปริมาณการใช้ Bandwidth ของเครือข่ายย่อย
- การสำรวจสถานะของเครื่องคอมพิวเตอร์ภายในเครือข่ายย่อย
- การสำรวจสถานะของการเชื่อมต่อระหว่างระบบเครือข่ายกับ Internet
- การประมวลผลที่ได้จาก 3 ส่วนย่อย และส่งกลับไปให้

ส่วนแสดงผล เพื่อแสดงข้อมูลในรูปแบบต่างๆให้กับผู้ใช้นั่นเอง

นอกจากส่วนการทำงานหลักแล้วยังมีส่วนประกอบย่อยอีก 1 ส่วน คือ

ส่วนการสำรวจข้อมูลเฉพาะเครื่อง

รายละเอียดของทั้ง 4 ส่วน มีดังนี้

3.3.1 ส่วนติดต่อกับผู้ใช้ เป็นส่วนที่จะรับ Input จากผู้ใช้เพื่อนำไปใช้งานในส่วนหลักต่อไป สำหรับ Input ที่รับมาจากผู้ใช้แบ่งออกเป็น 3 ส่วนย่อยคือ

1. รายละเอียดของอุปกรณ์ในเครือข่ายที่รองรับ SNMP ที่สนใจได้แก่ IP ของ Interface ของอุปกรณ์ที่ตอบรับ Protocol SNMP และ Read Community ของอุปกรณ์นั้นๆ
2. IP Range ของเครือข่ายย่อย ซึ่งมีข้อจำกัดคือต้องสอดคล้องกับ Interface ของหนึ่งในอุปกรณ์เครือข่ายที่รองรับ SNMP ที่สนใจไม่เกิน 2 port โดย IP Range แต่ละชุดจะกลายเป็นเครือข่ายย่อย 1 เครือข่ายนั่นเอง
3. รายชื่อของ URL หรือ IP ที่จะใช้เป็นตำแหน่งอ้างอิงในการสำรวจสถานะของการเชื่อมต่อกับ Internet ซึ่งควรจะเป็น URL หรือ IP ภายนอกระบบที่มีความน่าเชื่อถือพอสมควรโดย Default แล้วจะมีมาให้ 1 URL คือ www.google.co.th

ตัวอย่างข้อมูลที่ได้รับ

เครือข่ายย่อยที่ 1 :

IP Range 161.246.5.91 - 161.246.5.130 สอดคล้องกับ Interface Fast Ethernet 0/13 และ 0/14 ของอุปกรณ์เครือข่ายที่รองรับ SNMP ที่ Interface IP 161.246.5.253 และมี Read Community เป็น cereadonly

เครือข่ายย่อยที่ 2 :

IP Range 161.246.5.131 - 161.246.5.170 สอดคล้องกับ Interface Fast Ethernet 0/6 และ 0/7 ของอุปกรณ์เครือข่ายที่รองรับ SNMP ที่ Interface IP 161.246.5.253 และมี Read Community เป็น cereadonly

3.3.2 ส่วนสำรวจระบบเครือข่าย เป็นส่วนที่นำข้อมูลจากส่วนแรกมาทำการสำรวจและหาข้อมูลต่างๆของระบบเครือข่าย และเก็บข้อมูลที่ได้นั้นประมวลผลและส่งให้กับส่วนสุดท้ายเพื่อนำไปแสดงผล ประกอบด้วย การทำงาน 4 ส่วนคือ

1. การสำรวจปริมาณการใช้ Bandwidth ของเครือข่ายย่อย โดยจะทำการดึงข้อมูลปริมาณการใช้ระบบเครือข่ายจากอุปกรณ์ SNMP และทำการบันทึกลงใน Log File แยกตามเครือข่ายย่อย
2. การสำรวจสถานะของเครื่องคอมพิวเตอร์ภายในเครือข่ายย่อย โดยจะทำการ Scan Port ของเครื่องคอมพิวเตอร์ และทำการบันทึกลง Log File แยกตามเครือข่ายย่อย
3. การสำรวจสถานะการเชื่อมต่อระหว่างระบบเครือข่ายกับ Internet โดยจะทำการ Scan Port ไปยังเครื่องเป้าหมายที่จะสุ่มออกมาจาก URL หรือ IP ที่อยู่ใน List ของตำแหน่งอ้างอิงและทำการบันทึกลงใน Log File
4. ส่วนประมวลผล ซึ่งจะนำข้อมูลที่ได้จากส่วนที่ 3.3.2.1 และ 3.3.2.2 มาประมวลผลร่วมกันได้ปริมาณการใช้งานโดยเฉลี่ยของแต่ละเครื่องแยกตามเครือข่ายย่อยและทำการบันทึกลงใน Log File

3.3.3 ส่วนแสดงผล จะทำหน้าที่นำข้อมูลที่อยู่ใน Log File มาจัดแสดงเป็น กราฟ, แผนภูมิแท่ง หรือรายงานประกอบไปด้วย 3 ส่วน คือ

1. ส่วนการแสดงผลแผนภูมิแท่งซึ่งมีหัวข้อหลักในการแสดงคือ

- ปริมาณ Bandwidth รวม
- ปริมาณ Bandwidth โดยเฉลี่ยต่อวินาที

และมีหัวข้อย่อยคือ

- Download และ Upload (รวมทั้งหมด)
- Download อย่างเดียว
- Upload อย่างเดียว

โดยทั้งหมดจะแสดง 5 ลำดับ ของเครื่องคอมพิวเตอร์ในเครือข่ายย่อยที่มีการใช้งานสูงสุด

2. ส่วนการแสดงผลกราฟ 3 มิติซึ่งมีหัวข้อหลักในการแสดงคือ

- ปริมาณ Bandwidth ในช่วงเวลาต่างๆใน 1 วัน
- จำนวนของเครื่องคอมพิวเตอร์ที่ Online ในช่วงเวลาต่างๆใน 1 วัน

ซึ่งมีหัวข้อย่อย คือ

- Download และ Upload
- Download อย่างเดียว
- Upload อย่างเดียว

โดยทั้งหมดสามารถเลือกแสดงเฉพาะเครือข่ายย่อยบางเครือข่ายเพื่อเปรียบเทียบปริมาณการใช้งานได้ชัดเจนและยังสามารถเลือกวันที่ใช้ในการแสดงผลได้ นอกจากนี้ในส่วนนี้ยังมี

หน้าต่างย่อยเพื่อตรวจสอบว่า ณ เวลาที่กำหนดนั้นมีเครื่องคอมพิวเตอร์เครื่องใดในเครือข่ายย่อย Online อยู่บ้างอีกด้วย

3. ส่วนการแสดงลำดับของเครื่องคอมพิวเตอร์ที่มีการใช้งาน Bandwidth สูงสุด 3 ส่วนย่อย คือ

- Download และ Upload
- Download อย่างเดียว
- Upload อย่างเดียว

โดยทั้งหมดจะถูกแสดงทีละเครือข่ายย่อยตามที่ได้เลือกไว้

นอกจากส่วนการทำงานหลักแล้วยังมีส่วนประกอบย่อยอีก 1 ส่วน คือ

3.3.4 ส่วนการสำรวจข้อมูลเฉพาะเครื่อง รายละเอียดจะแบ่งออกเป็น 3 ส่วน ดังนี้

1. ส่วนรับ IP และ Host Name
2. ส่วนรับของรายละเอียดที่ต้องการตรวจสอบแบ่งออกเป็น

- Service Scan : จะทำการตรวจสอบ IP/Hostname ที่กำหนดว่ามีการเปิดบริการใดอยู่บ้าง
- Trojan Scan : จะทำการตรวจสอบ IP/Hostname ที่กำหนดว่ามีการติด Trojan หรือ Worm ที่อาจเป็นอันตรายต่อระบบเครือข่ายหรือไม่
- Specific Scan : จะทำการตรวจสอบ IP/Hostname ที่กำหนดตาม Port Number ที่ผู้ใช้ได้ป้อนเข้ามา

3. ส่วนที่แสดงผลการตรวจสอบ IP / Hostname ที่กำหนด

3.4 การพัฒนาโปรแกรม

3.4.1 ศึกษาและทดลองการดึงค่า SNMP/RMON จาก อุปกรณ์เครือข่ายโดยใช้ Library SNMP++ ด้วยภาษา C#

อุปสรรค

- การ Handler Error ทำได้ยาก
- ผู้ใช้ไม่สามารถจดจำได้ว่า Object ID ใดทำหน้าที่ใด
- อุปกรณ์ทั่วไปรองรับเพียง SNMP พื้นฐานโดยการรองรับ RMON นั้นก็มีเพียงบางส่วนของ Object ID ทั้งหมด

วิธีการแก้ไข

- ทำการบังคับข้อมูลที่ผู้ใช้ป้อนเพื่อลดความผิดพลาดจาก Library และลดภาระของผู้ใช้ในการค้นหา Object ID ที่ถูกต้อง

- อาศัยการดึงข้อมูลจาก SNMP พื้นฐานเป็นหลักเพื่อลดความผิดพลาดในกรณีที่อุปกรณ์เครือข่ายที่ต้องการไม่รองรับ Object ID ที่กำหนดไว้

3.4.2 ศึกษาและทดลองการสร้าง Child Process เพื่อเรียกโปรแกรม Nmap ในการ Scan Port ที่สนใจ ด้วยภาษา C#

อุปสรรค

- Child Process ที่เรียกออกมาจะไม่ฟ้อง Error เมื่อเกิดความผิดพลาดและจะปิดตัวลง
- ในบางกรณี Child Process จะไม่ยอมปิดตัวเองทำให้ไม่สามารถสร้าง Child Process อีกครั้งเมื่อต้องการได้

วิธีการแก้ไข

- มีการตรวจสอบว่ายังคงมี Child Process เดิมอยู่หรือไม่ถ้าหากพบว่ามีก็จะทำการ Terminate ทิ้งก่อนจะมีการสร้าง Child Process ใหม่

3.4.3 ศึกษาและทดลองการแตก Thread ให้กับ Function เพื่อลดภาวะ Not Responding ของโปรแกรมในภาษา C#

อุปสรรค

- วิธีการใช้ Thread นั้นอาศัยการเรียก Method Name ที่ต้องการแตกย่อยออกมาโดยไม่อนุญาตให้มีการ Pass หรือ Return Parameter จึงทำให้โครงสร้างของโปรแกรมเปลี่ยนไปจากเดิม
- การเรียกใช้ค่าจาก Class อื่นจากภายใน Thread ก็เป็นผลให้เกิดความผิดพลาดขึ้นได้

วิธีการแก้ไข

- รวบ Method หลักๆเข้ามาอยู่ใน Class เดียวกันเพื่อลดความผิดพลาดในการส่งผ่านค่าข้าม Class
- มีการใช้การ Locking ช่วยในกรณีที่มีการใช้ Resource ร่วมกันระหว่าง Thread
- การใช้ Thread Call Method ข้าม Class นั้นจะอาศัยการสื่อสารกับโปรแกรมหลักผ่าน File แทน

3.4.4 ศึกษาและทดลองการสร้าง Graph และแผนภูมิต่างๆ ด้วย Library Chart Director ในภาษา C#

อุปสรรค

- ลักษณะของข้อมูลที่ Chart Director รับมาสร้าง Graph และแผนภูมิต่างๆนั้นเป็นตัวแปรในรูปแบบของ Double[] ซึ่งเดิมใช้ Long
- เป็น Library ที่ไม่ได้รองรับการทำงานที่มีการ Update ข้อมูลในช่วงสั้นๆ

วิธีการแก้ไข

- สร้าง Method สำหรับการแปลงค่าที่จากเดิมเป็นลักษณะของ Double[]
- อาศัยการเขียน Graph และแผนภูมิใหม่ในกรณีที่ต้องการ Update ถึงแม้จะมีอาการกระตุกบ้างแต่ก็ยอมรับได้

3.4.5 ศึกษาและทดลองการทำงานแบบ Real-time ด้วยภาษา C#

อุปสรรค

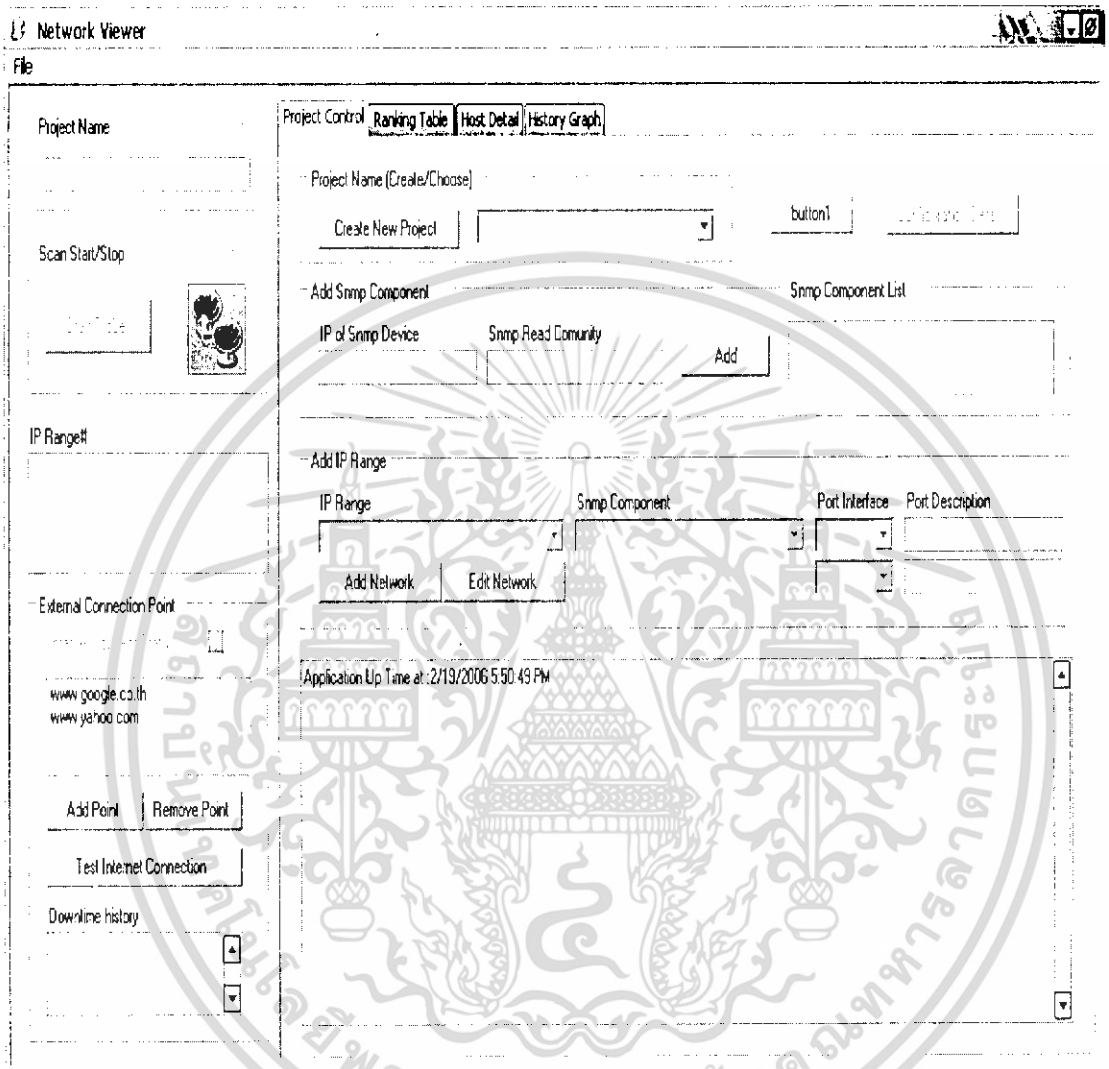
- หากมีการตั้ง Refresh Rate สั้น Library SNMP++ จะมีการใช้งาน CPU ที่สูงมากเป็นผลให้ระบบโดยรวมทำงานช้าลงอย่างเห็นได้ชัด
- หากมีการตั้ง Refresh Rate สั้น Library Chart Director จะมีการจองตัวแปรและไม่มีการลบทิ้งเป็นผลให้มีการใช้งาน Physical Memory สูงขึ้นเรื่อยๆจนถึงขีดจำกัดที่ OS กำหนดให้เป็นผลให้โปรแกรม crash ทันที
- การใช้ Nmap แสกนจำเป็นต้องใช้เวลาช่วงหนึ่งทำให้หาก Refresh Rate สั้น อาจเกิดการคาบเกี่ยวของการแสกนได้

วิธีการแก้ไข

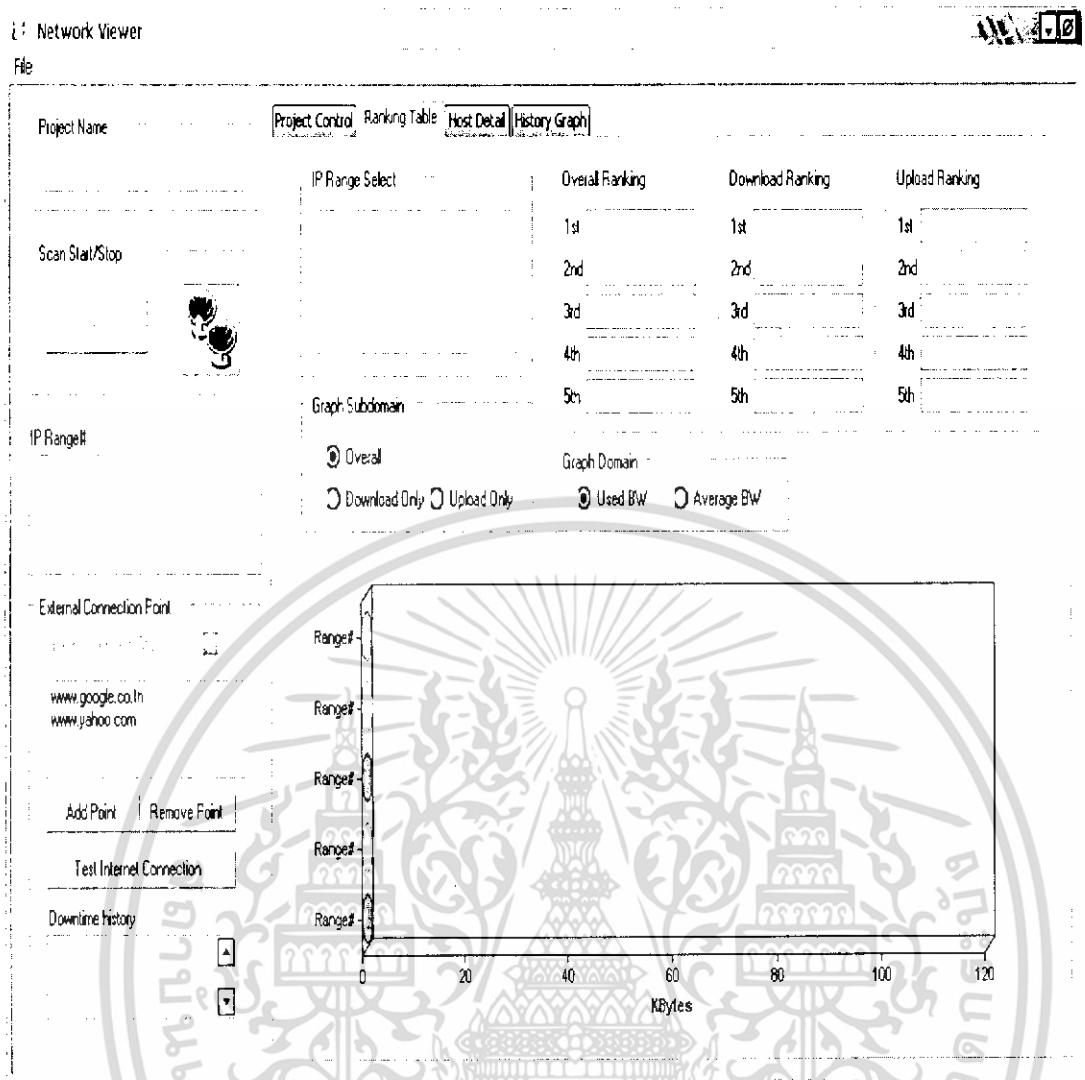
- ตั้ง Refresh Rate ให้มากขึ้นและเนื่องด้วยเหตุผลนี้จึงจำเป็นต้องไม่ให้ผู้ใช้แก้ไขค่า Refresh Rate เองได้

3.5 ตัวอย่างส่วนติดต่อกับผู้ใช้

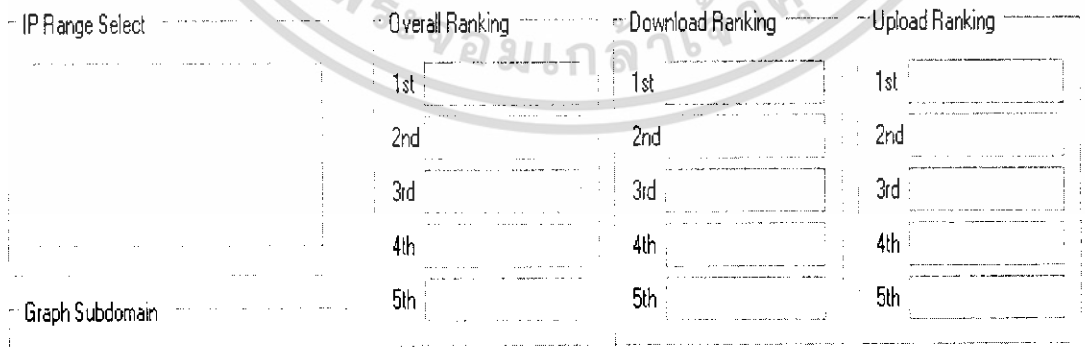
User Interface



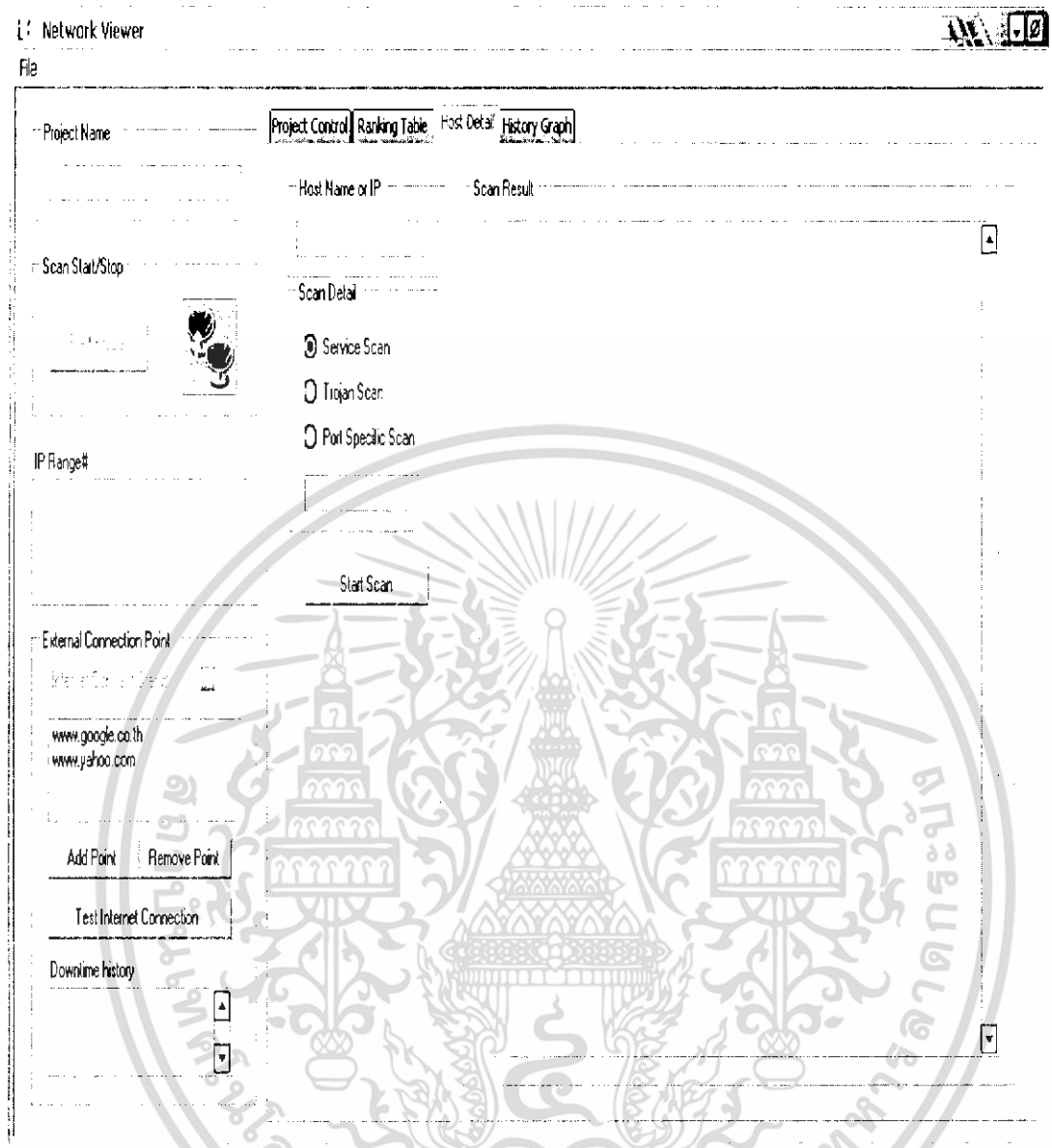
รูปที่ 3.3 User Interface ของการ Configuration ค่าเริ่มต้น



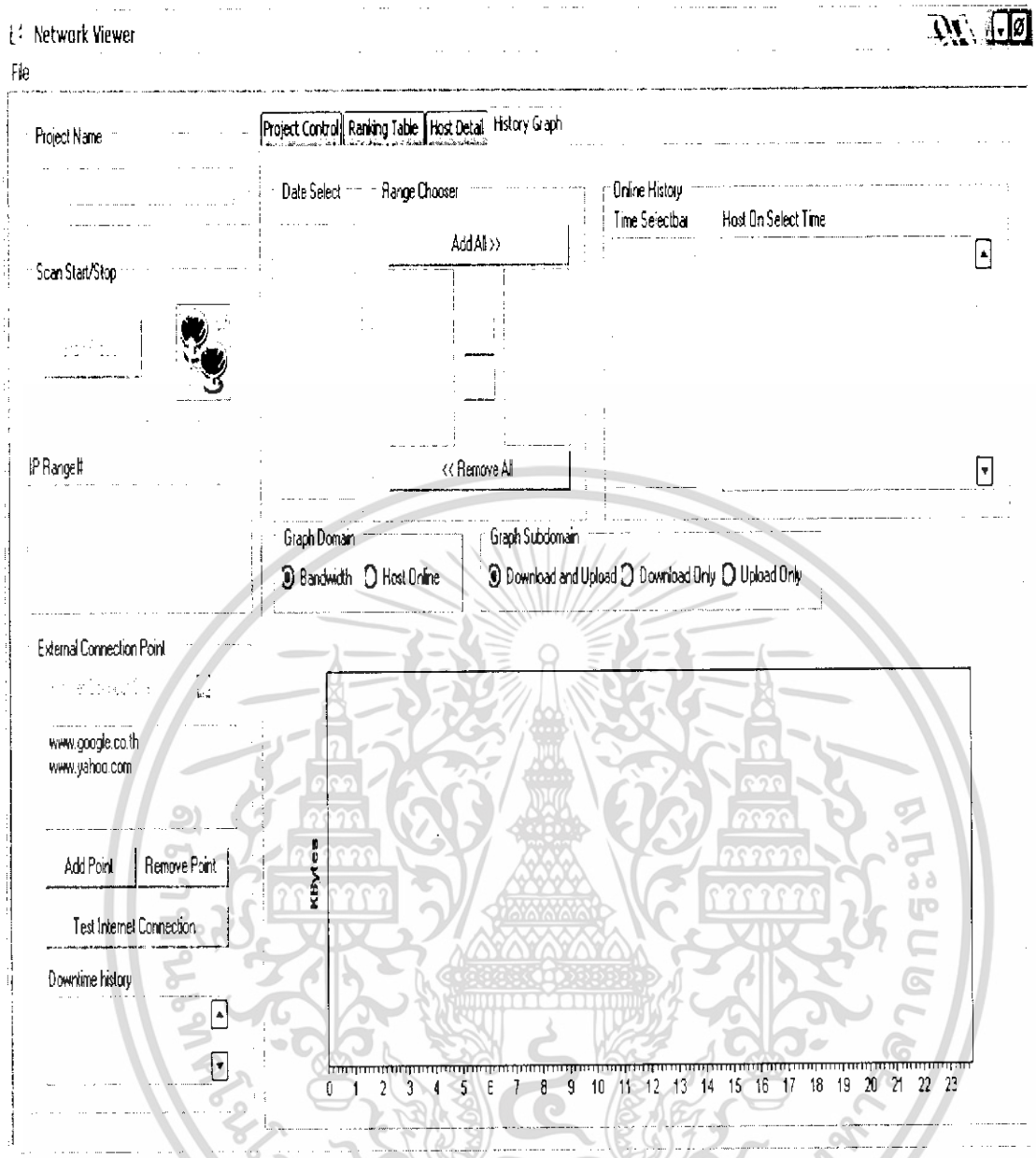
รูปที่ 3.4 User Interface ส่วนการแสดงผลแผนภูมิแท่งปริมาณ Bandwidth รวม และปริมาณ Bandwidth โดยเฉลี่ยต่อวินาที



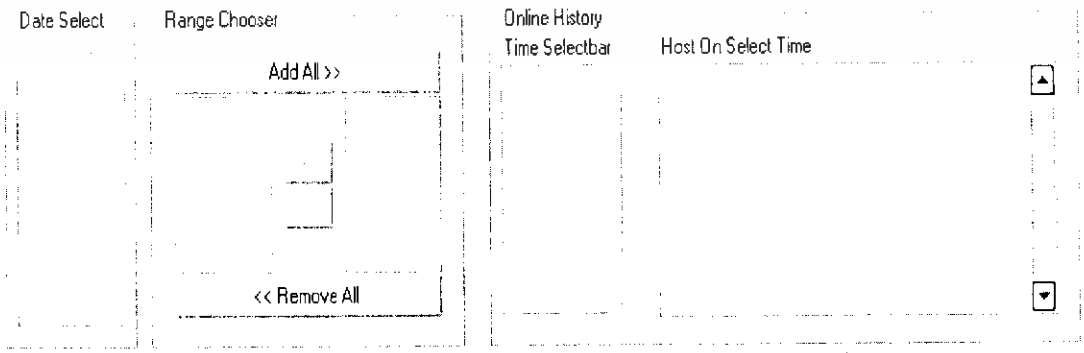
รูปที่ 3.5 User Interface แสดงลำดับของเครื่องคอมพิวเตอร์ในเครือข่ายย่อยที่มีการใช้งานสูงสุด



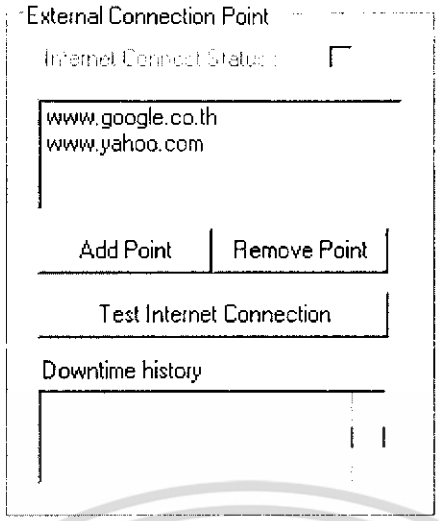
รูปที่ 3.6 User Interface ของการสำรวจข้อมูลเฉพาะเครื่อง



รูปที่ 3.7 User Interface ส่วนของการแสดงกราฟปริมาณ Bandwidth ในช่วงเวลาต่างๆใน 1 วัน และจำนวนของเครื่องคอมพิวเตอร์ที่ Online ในช่วงเวลาต่างๆใน 1 วัน



รูปที่ 3.8 User Interface ที่แสดงว่า ณ เวลาที่กำหนดมีเครื่องคอมพิวเตอร์เครื่องใดในเครือข่ายย่อย Online อยู่บ้าง



รูปที่ 3.9 User Interface ที่แสดงการสำรวจสถานะของการเชื่อมต่อระหว่างระบบเครือข่ายกับ Internet

บทที่ 4
การทดลองและผลการทดลอง

4.1 ด้านการสแกนเพื่อตรวจสอบ

ICMP Ping Sweep

ทดลองการใช้วิธีการ Ping Sweep ภายในเน็ตเวิร์คภาคทั้งภายใน Subnet เดียวกันและคนละ Subnet เพื่อค้นหาเครื่องปลายทางในขอบเขตที่สนใจที่เชื่อมต่อกับเน็ตเวิร์คของภาควิชาและเปิดใช้งานอยู่ ซึ่งจากผลการทดลองแสดงให้เห็นว่าไม่สามารถให้ผลที่ถูกต้องได้ ไม่ว่าจะใช้ ICMP Packet ใน Type ใดๆก็ตามอันเนื่องมาจากในปัจจุบันได้มีการใช้โปรแกรม Personal Firewall อย่างแพร่หลายและแม้กระทั่งภายในโปรแกรม Anti-virus หลายๆชนิดก็มีการรวมเอา Personal Firewall เข้าเป็นส่วนประกอบหนึ่งไปแล้วเป็นผลให้เครื่องปลายทางในขอบเขตที่สนใจไม่ทำการตอบรับ Packet ICMP ที่ส่งไปถึงแม้ว่าในตัวเน็ตเวิร์คอนุญาตให้ Packet ICMP วิ่งผ่านได้ก็ตาม

Port Scanning

ทดลองการใช้วิธีการสแกน Port ด้วยวิธีการตั้งค่า Flag ต่างๆไปยังพอร์ตพื้นฐานที่ระบบปฏิบัติการต่างๆเปิดไว้เช่นพอร์ต 25, พอร์ต 110 และพอร์ต 143 ทั้งใน Subnet เดียวกันและคนละ subnet เพื่อค้นหาเครื่องปลายทางในขอบเขตที่สนใจที่เชื่อมต่อกับเน็ตเวิร์คของภาควิชาและเปิดใช้งานอยู่ ซึ่งจากผลการทดลองถึงแม้ว่าจะไม่สามารถให้ผลที่ถูกต้องสมบูรณ์ได้เนื่องจากเครื่องปลายทางที่ใช้ระบบปฏิบัติการ Linux หรือ Unix นั้นสามารถปิดพอร์ตมาตรฐานได้ทำให้เครื่องปลายทางไม่ตอบรับกับการสแกนพอร์ตแต่นอกจากกรณีนี้แล้วผลที่ได้มีความถูกต้องสูงกว่าการใช้ ICMP Ping Sweep

สำหรับรูปแบบการสแกนพอร์ตทั้งหมดพบว่าวิธีการในรูปแบบของ TCP Connect scan จะเหมาะสมที่สุดเนื่องจากการสแกนในรูปแบบอื่นๆนอกจาก TCP connect scan นั้นเครื่องปลายทางที่มีการติดตั้งโปรแกรม Personal Firewall บางตัวจะทำให้ผลที่ได้จากการสแกนได้ว่าพอร์ตนั้นเป็น closed หรืออาจทำให้ผลที่ได้จากการ Scan นั้นกลายเป็น Open | Filtered แทนซึ่งทำให้ผลของการใช้การสแกนพอร์ต ค้นหาเครื่องปลายทางนั้นผิดเพี้ยนไปรวมทั้งการใช้วิธีสแกนพอร์ตในการค้นหาพอร์ตที่เปิดที่เครื่องปลายทางเพื่อตรวจสอบเซอร์วิสที่เปิดอยู่นั้นก็จะผิดเพี้ยนไปด้วย

การตรวจหาประเภทของระบบปฏิบัติการ

ทดลองการตรวจหาประเภทของระบบปฏิบัติการโดยวิธีการของ Impossible Flags ร่วมกับ OS Fingerprint และวิธีการของ Active Stack Fingerprinting ซึ่งจากผลการทดลองแสดงให้เห็นว่าวิธีการนี้ไม่สามารถชี้ชัดได้ว่าเครื่องปลายทางเป็นระบบปฏิบัติการประเภทใดได้หากว่าเครื่องปลายทางมีการติดตั้งโปรแกรม Personal Firewall

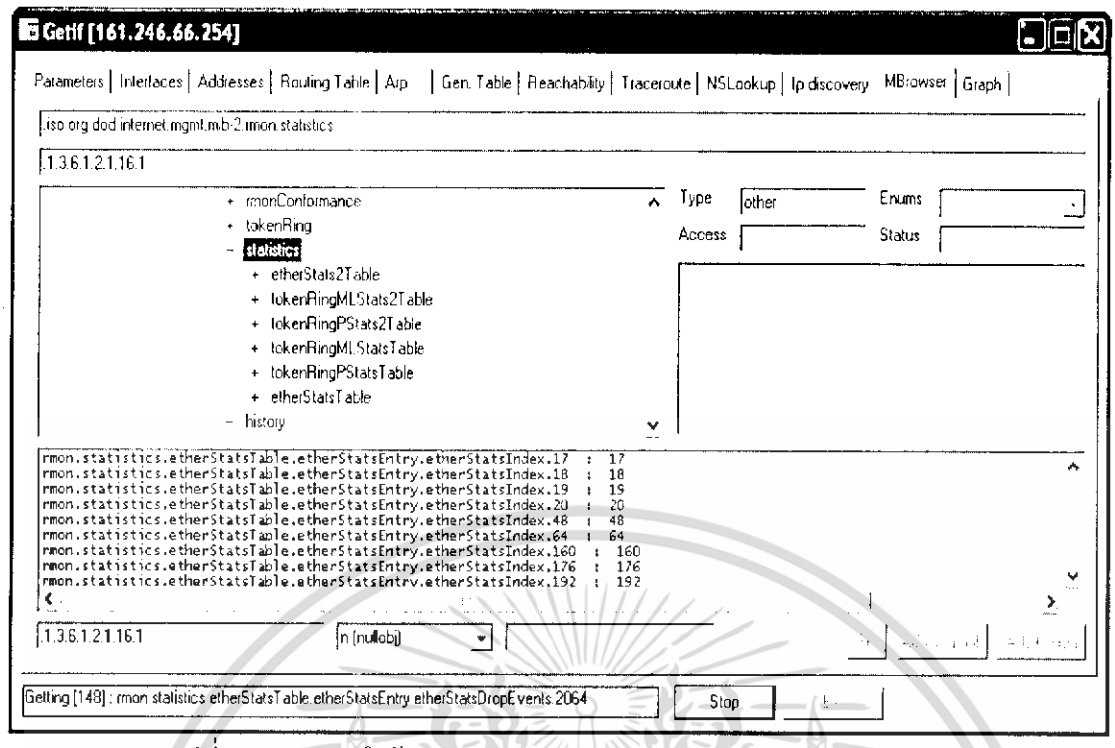
4.2 ด้าน SNMP และ RMON

การตรวจสอบคุณสมบัติของอุปกรณ์บนระบบเครือข่ายของภาควิชา

ทดลองจากการศึกษารายละเอียดของอุปกรณ์บนเครือข่ายของภาควิชา โดยจากการสอบถามอาจารย์ และจากรายละเอียดของอุปกรณ์ในรุ่นนั้นๆ ซึ่งจากผลการทดลองในภาควิชาจะมีอุปกรณ์ที่มีคุณสมบัติของ SNMP และ RMON อยู่หลายรุ่นด้วยกันโดยอุปกรณ์ที่ใช้เชื่อมต่อเครือข่ายในแต่ละชั้นของภาควิชาจะเป็น สวิตซ์ Cisco Systems Catalyst 3500 Series XL ซึ่งสนับสนุน RMON 4 กลุ่ม ด้วยกัน คือ History, Statistics,Alarms, และ Events และ เกตเวย์ของภาควิชาจะเป็น สวิตซ์ Cisco Systems Catalyst 4000 Series ซึ่งสนับสนุนทั้ง RMON ,RMON II ซึ่งจากคุณสมบัติดังกล่าวนี้ของสวิตซ์ทั้ง 2 รุ่น ทำให้สามารถนำข้อมูลที่ได้จาก RMON ทั้ง 4 กลุ่มมาทำการตรวจสอบได้

การทดลองดึงข้อมูล RMON จากอุปกรณ์

ทดลองดึงค่า RMON ทั้ง 4 กลุ่ม คือ history ,statistic ,alarm และevent มาตรวจสอบว่าสามารถดึงข้อมูลมาตรวจสอบได้ หรือไม่ จากผลการทดลอง หากเปิดบริการของ SNMP ที่สวิตซ์ตัวใดแล้ว และทราบ SNMP Parameters(Read/Write Community) ของอุปกรณ์ตัวนั้นจะสามารถดึงข้อมูลของ RMON ในกลุ่มที่ สวิตซ์ตัวนั้นสนับสนุนอยู่ได้ ดังรูปที่ 4.1 เป็นรูปตัวอย่างการทดลองดึงข้อมูลของ RMON ในกลุ่มของ statistics มาตรวจสอบ จากสวิตซ์ที่เป็นเกตเวย์ของภาควิชา(IP Address 161.246.66.254) โดยใช้โปรแกรม Getif 2.3.1



รูปที่ 4.1 ทดลองดึงข้อมูล RMON ในกลุ่มของ statistics มาตรวจสอบ

4.3 การทดลองใช้งานโปรแกรมสำรวจเครือข่าย และระบบคอมพิวเตอร์

แสดงการทดลองใช้งาน โปรแกรมสำรวจเครือข่าย และระบบคอมพิวเตอร์โดยการกำหนดค่าเริ่มต้น ดังนี้

เครือข่ายย่อยที่ 1 : (Network Lab.)

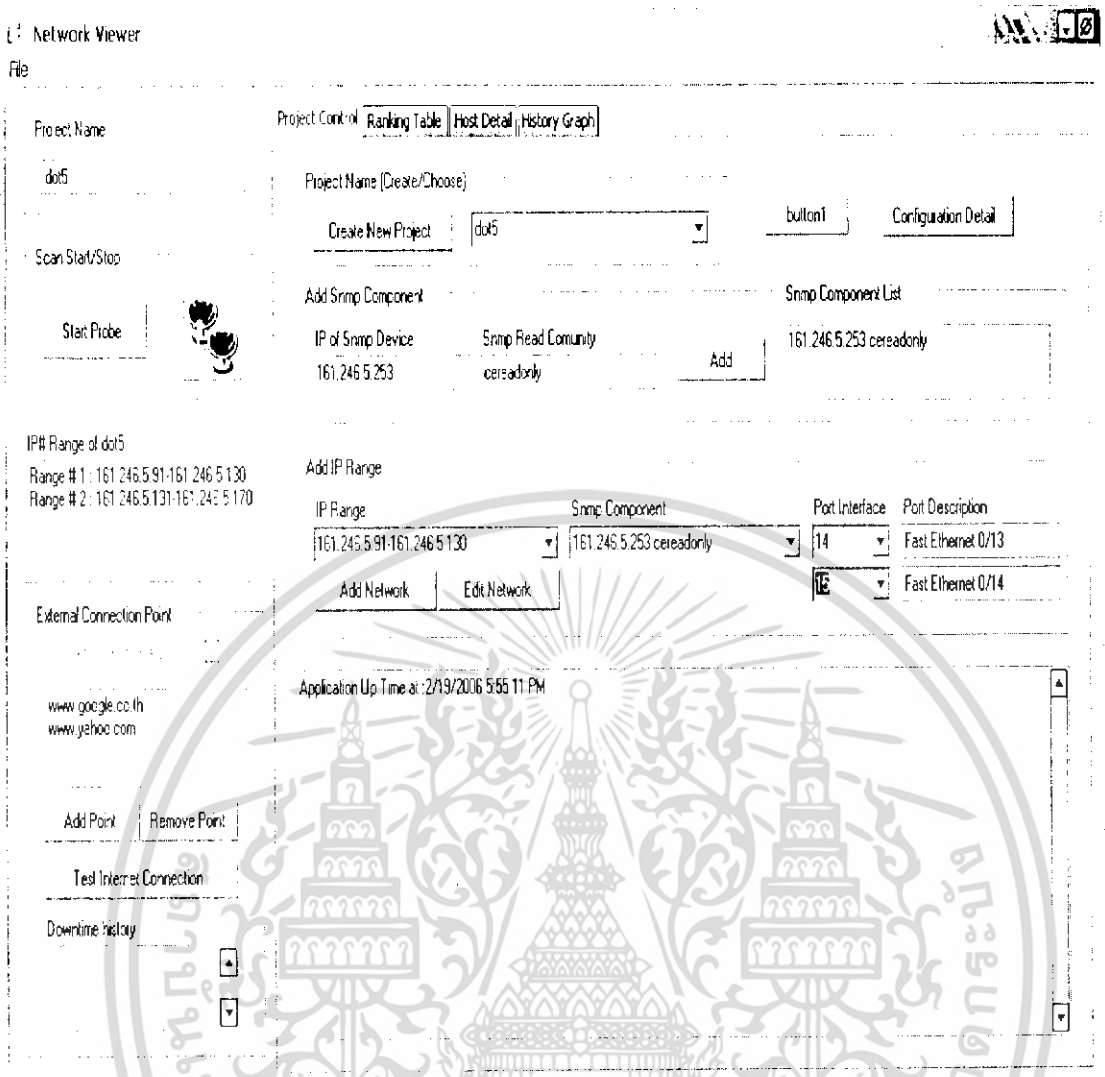
IP Range 161.246.5.91 - 161.246.5.130 สอดคล้องกับ Interface Fast Ethernet 0/13 และ 0/14 ของอุปกรณ์เครือข่ายที่รองรับ SNMP ที่ Interface IP 161.246.5.253 และมี Read Community เป็น cereadonly

เครือข่ายย่อยที่ 2 : (ESL Lab.)

IP Range 161.246.5.131 - 161.246.5.170 สอดคล้องกับ Interface Fast Ethernet 0/6 และ 0/7 ของอุปกรณ์เครือข่ายที่รองรับ SNMP ที่ Interface IP 161.246.5.253 และมี Read Community เป็น cereadonly

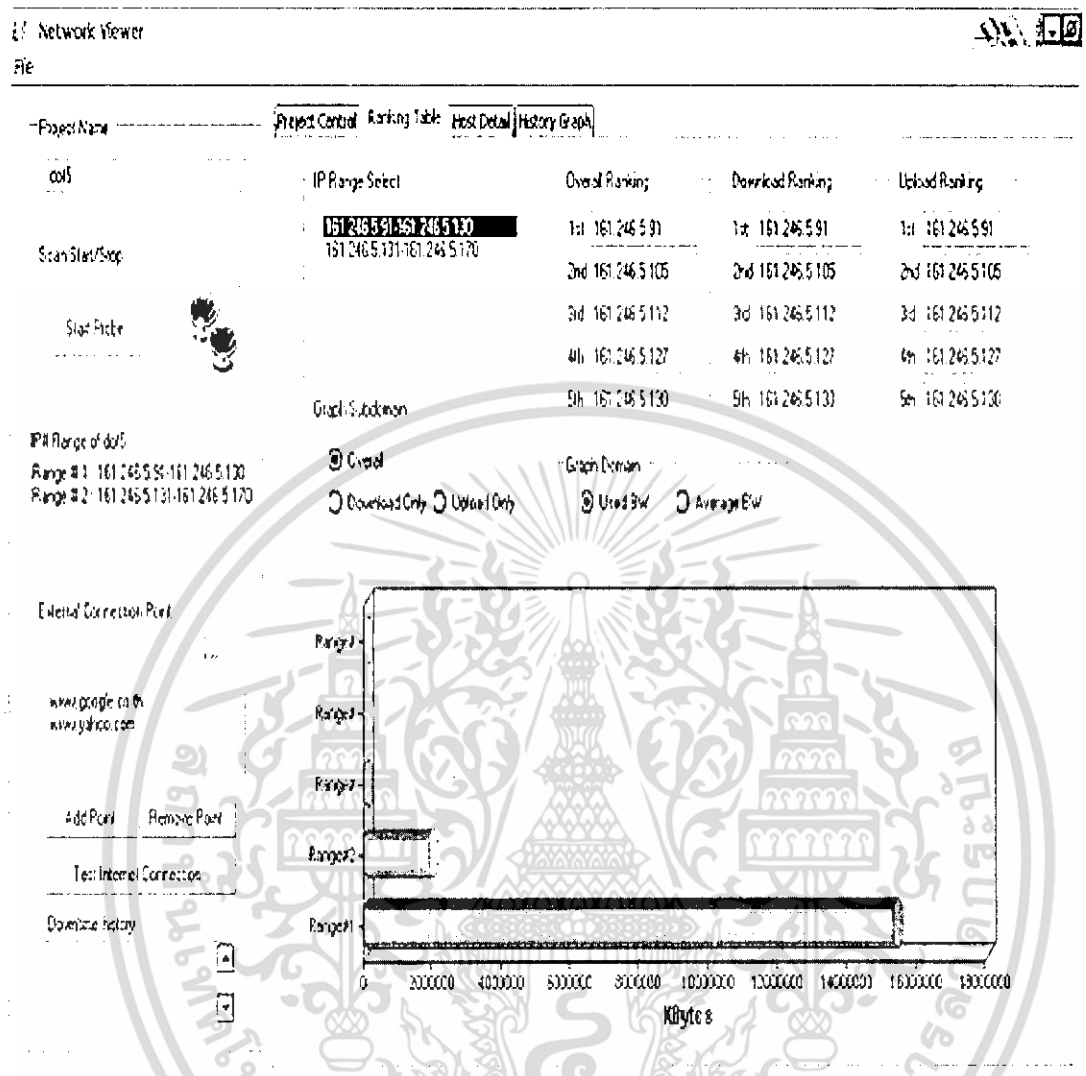
โดยให้รายชื่อของ URL ที่จะใช้เป็นตำแหน่งอ้างอิงในการสำรวจสถานะของการเชื่อมต่อกับ Internet คือ www.google.co.th และ www.yahoo.com

เมื่อทำการกำหนดค่าเริ่มต้นทั้งหมดลงไปในส่วนของ User Interface ของการ Configuration ได้ดังรูปที่ 4.2

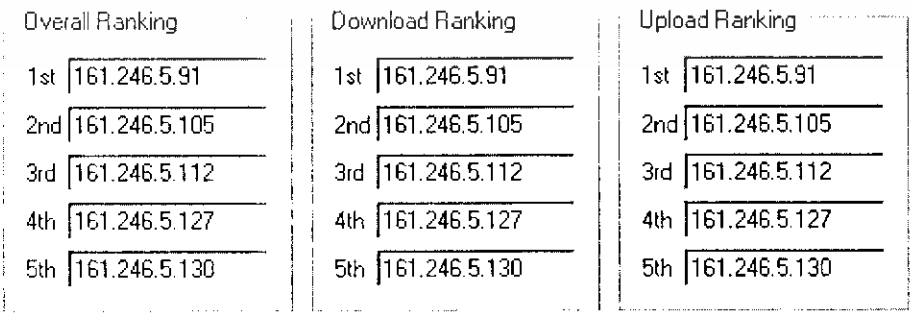


รูปที่ 4.2 แสดงการกำหนดค่าเริ่มต้นต่างๆให้กับโปรแกรม

จากนั้นจึงเริ่มใช้งานโปรแกรมให้ผลการทดลองแสดงออกมา ดังรูป



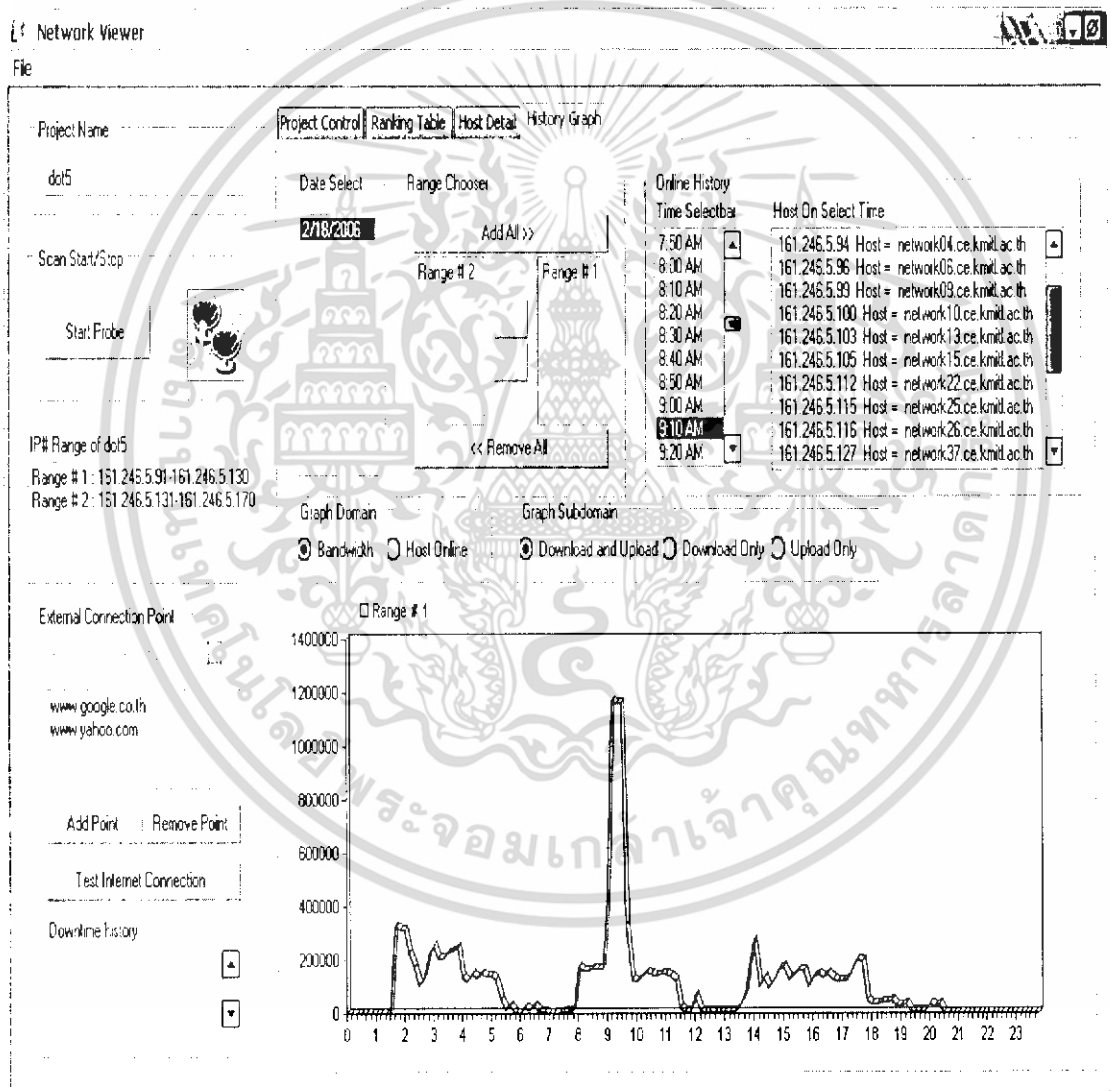
รูปที่ 4.3 แสดงผลของแผนภูมิแท่งปริมาณ Bandwidth รวม และปริมาณ Bandwidth โดยเฉลี่ยต่อวินาที ของทั้ง 2 เครือข่ายย่อย



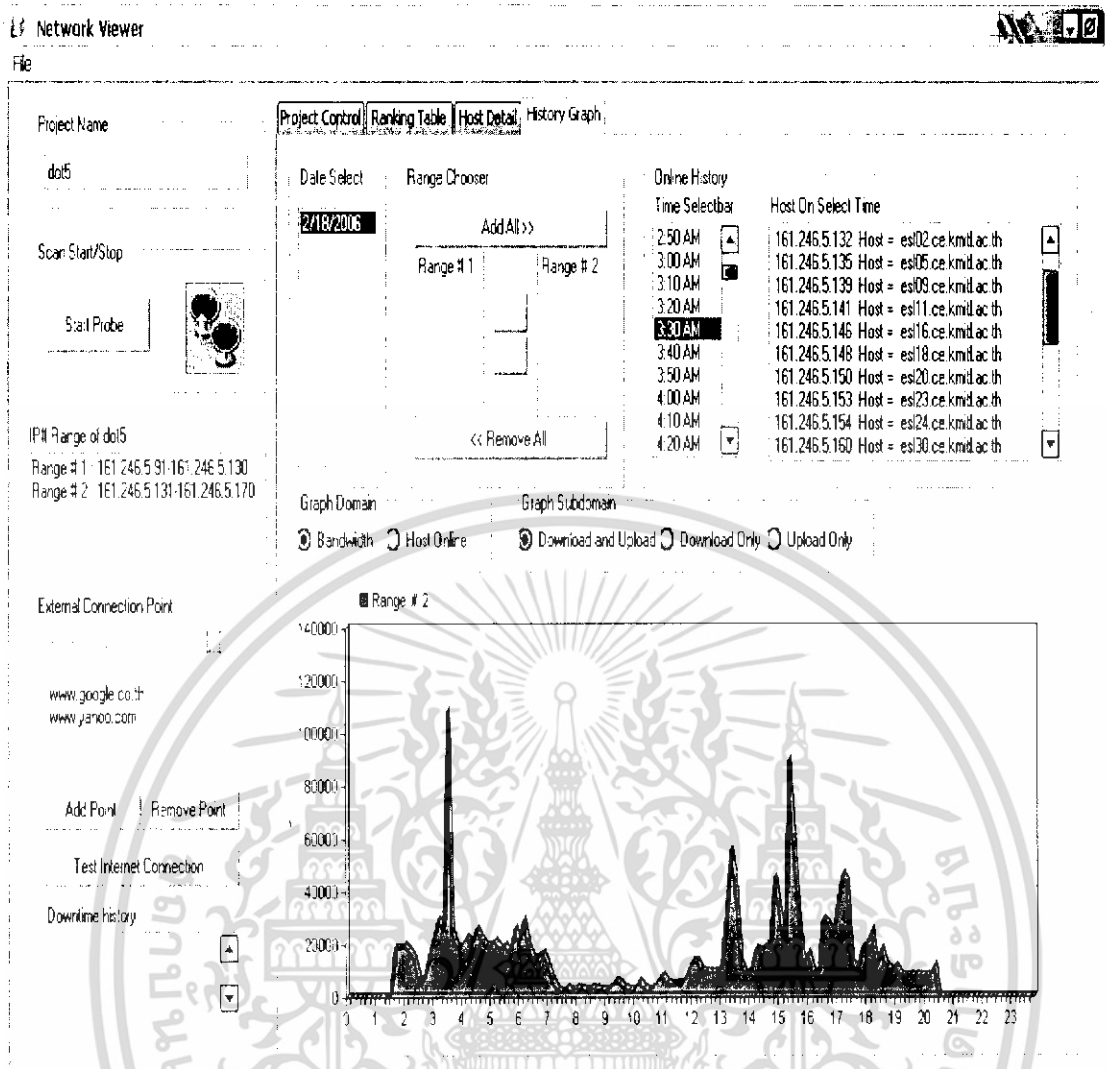
รูปที่ 4.4 แสดงลำดับของเครื่องคอมพิวเตอร์ในเครือข่ายย่อยที่ 1 ที่มีการใช้งานสูงสุด

Overall Ranking		Download Ranking		Upload Ranking	
1st	161.246.5.141	1st	161.246.5.141	1st	161.246.5.141
2nd	161.246.5.146	2nd	161.246.5.146	2nd	161.246.5.146
3rd	161.246.5.148	3rd	161.246.5.148	3rd	161.246.5.148
4th	161.246.5.153	4th	161.246.5.153	4th	161.246.5.153
5th	161.246.5.154	5th	161.246.5.154	5th	161.246.5.154

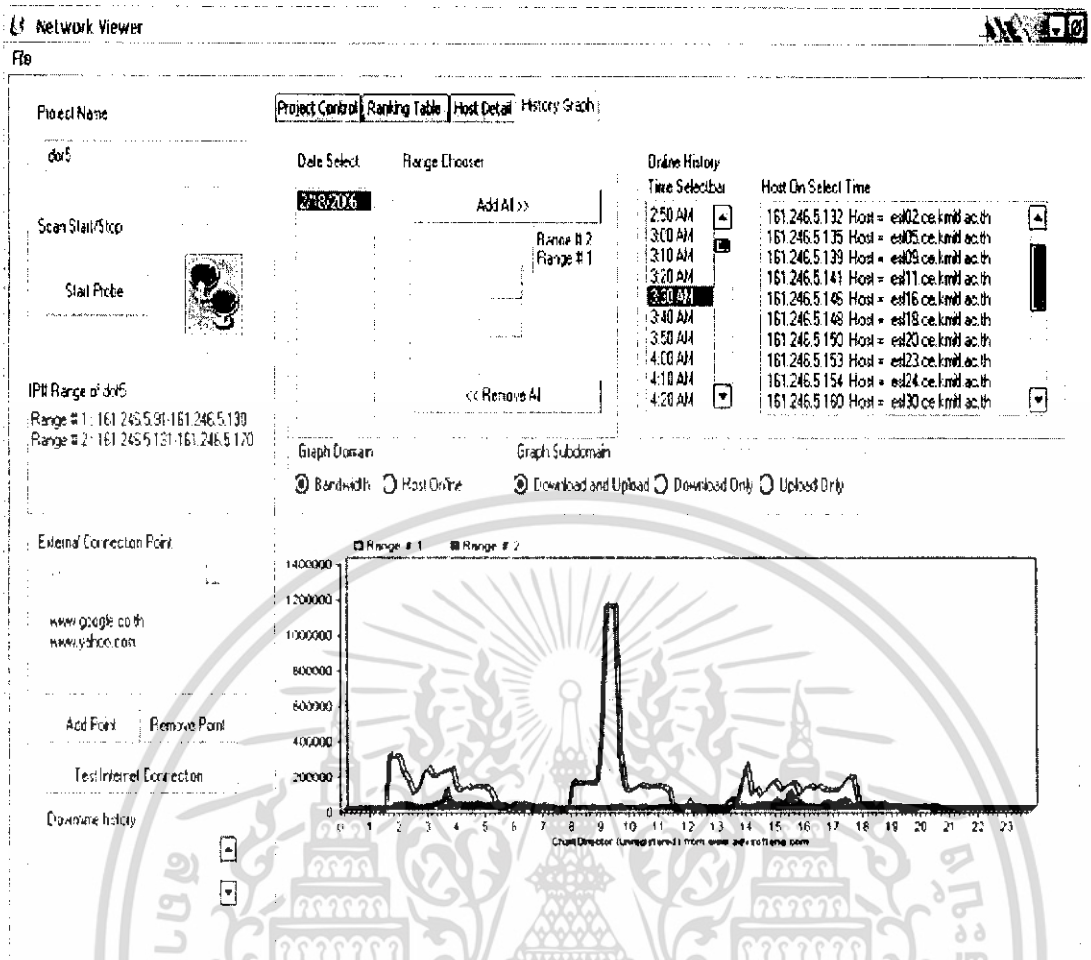
รูปที่ 4.5 แสดงลำดับของเครื่องคอมพิวเตอร์ในเครือข่ายย่อยที่ 2 ที่มีการใช้งานสูงสุด



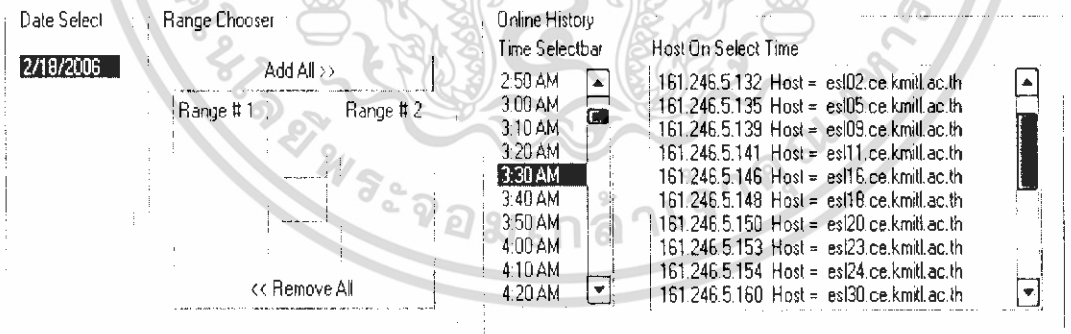
รูปที่ 4.6 แสดงผลกราฟปริมาณ Bandwidth ในช่วงเวลาต่างๆใน 1 วัน และจำนวนของเครื่องคอมพิวเตอร์ที่ Online ในช่วงเวลาต่างๆใน 1 วัน ของเครือข่ายย่อยที่ 1



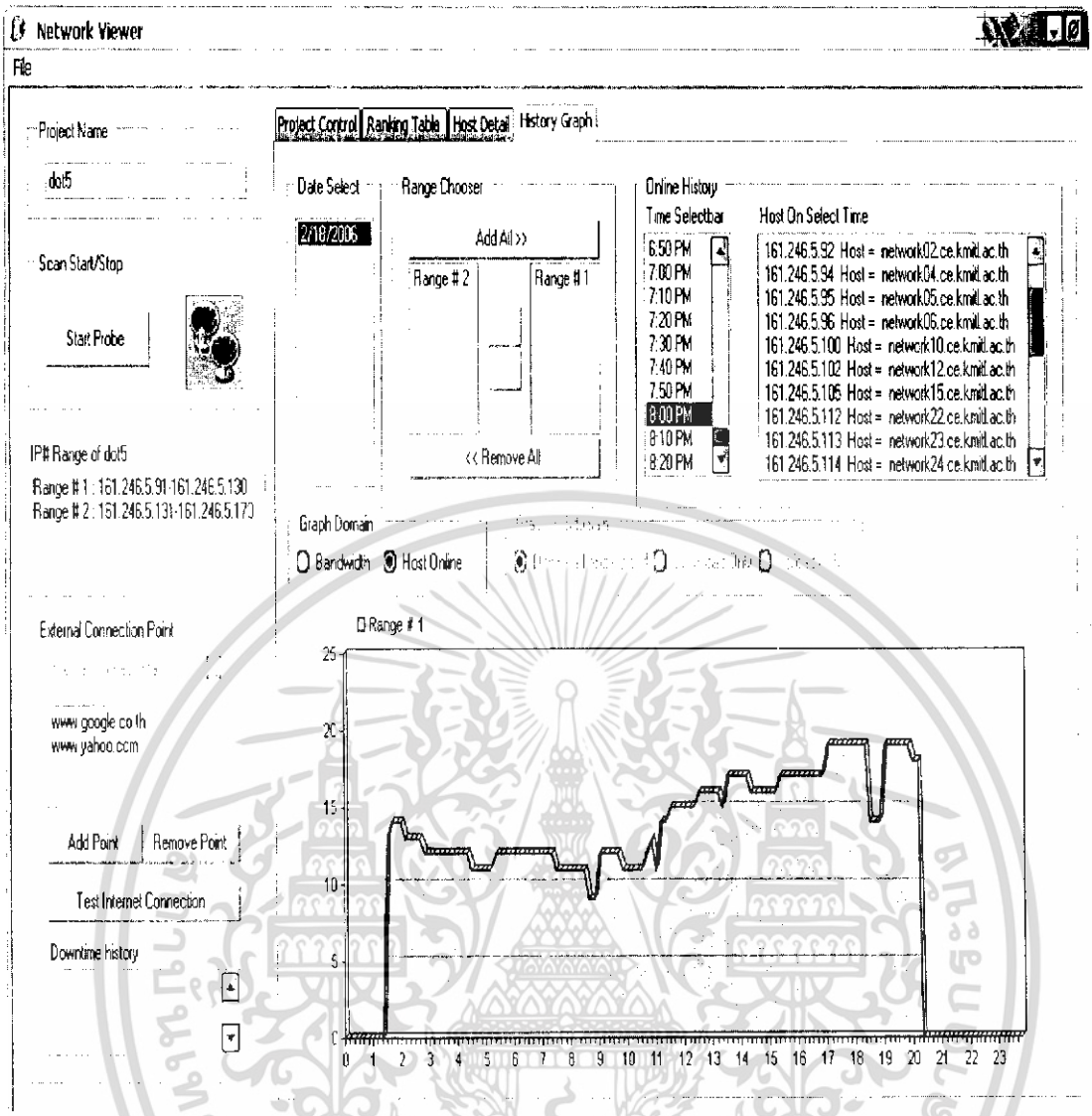
รูปที่ 4.7 แสดงผลกราฟปริมาณ Bandwidth ในช่วงเวลาต่างๆใน 1 วัน และจำนวนของเครื่องคอมพิวเตอร์ที่ Online ในช่วงเวลาต่างๆใน 1 วัน ของเครือข่ายย่อยที่ 2



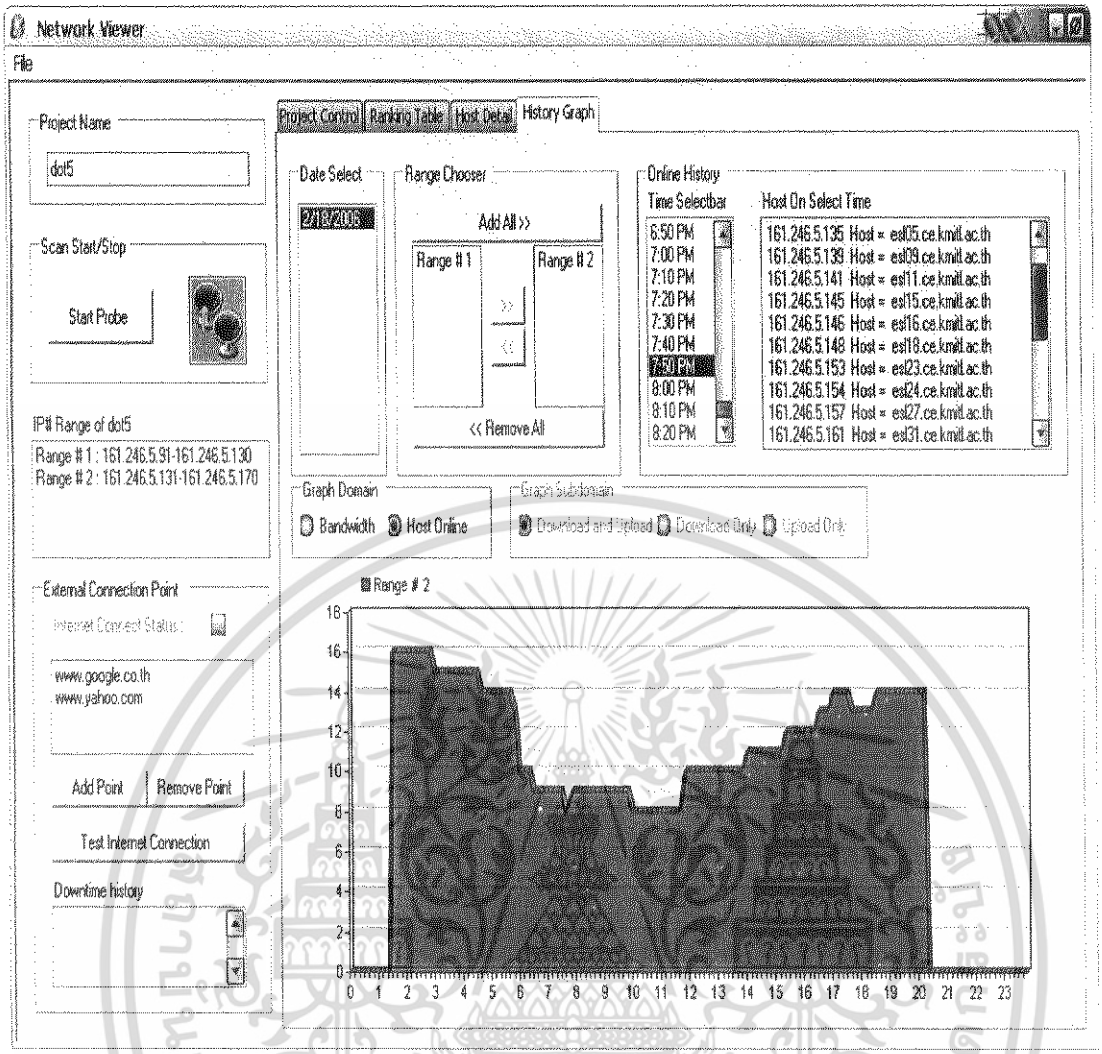
รูปที่ 4.8 แสดงผลกราฟเปรียบเทียบปริมาณ Bandwidth ในช่วงเวลาต่างๆใน 1 วัน ของเครือข่ายย่อยทั้งสอง



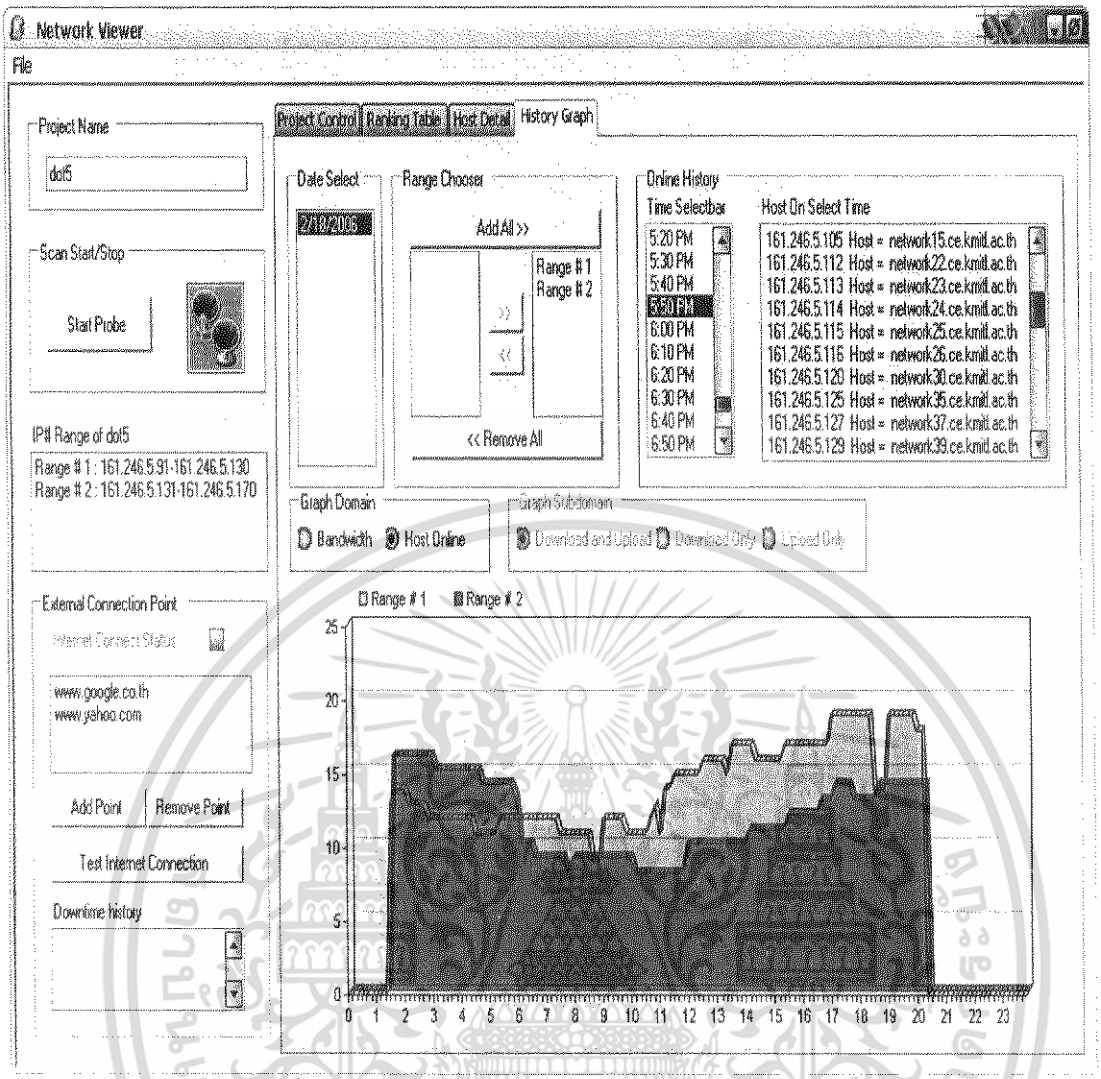
รูปที่ 4.9 ตัวอย่างการแสดงผลว่า ณ เวลาที่กำหนดมีเครื่องคอมพิวเตอร์เครื่องใดในเครือข่ายย่อย Online อยู่บ้าง



รูปที่ 4.10 แสดงผลกราฟปริมาณ Host ที่ Online ในช่วงเวลาต่างๆใน 1 วัน และจำนวนของเครื่องคอมพิวเตอร์ที่ Online ในช่วงเวลาที่กำหนด ของเครือข่ายย่อยที่ 1

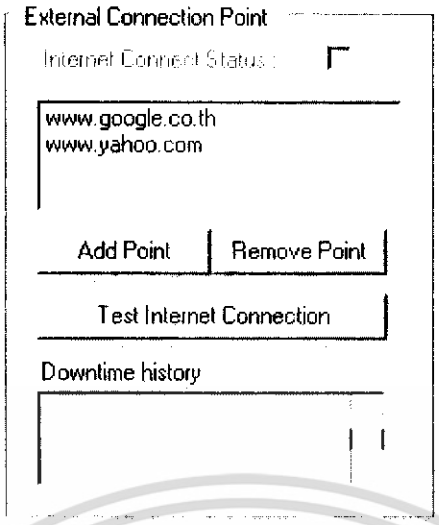


รูปที่ 4.11 แสดงผลกราฟปริมาณ Bandwidth ในช่วงเวลาต่างๆใน 1 วัน และจำนวนของเครื่องคอมพิวเตอร์ที่ Online ในช่วงเวลาที่กำหนด ของเครือข่ายย่อยที่ 2



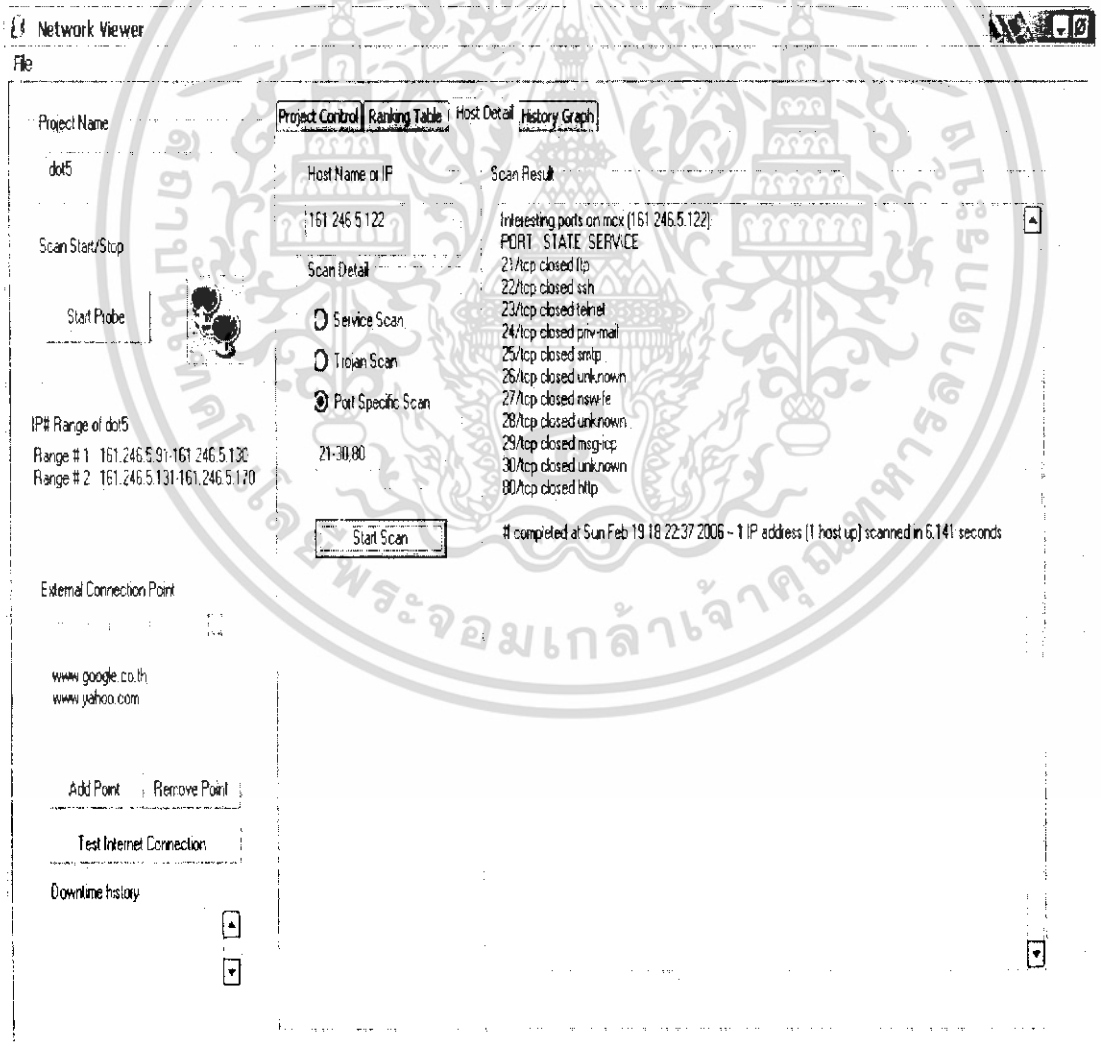
รูปที่ 4.12 แสดงผลกราฟเปรียบเทียบปริมาณ Host ที่ Online ในช่วงเวลาต่างๆใน 1 วัน ของเครือข่ายย่อยทั้งสอง

ส่วนของการสำรวจสถานะการเชื่อมต่อระหว่างระบบเครือข่ายกับ Internet โดยจะทำการ Scan Port ไปยังเครื่องเป้าหมายที่จะสุ่มออกมาจาก URL ที่อยู่ใน List ของตำแหน่งอ้างอิง ดังรูปที่ 4.13

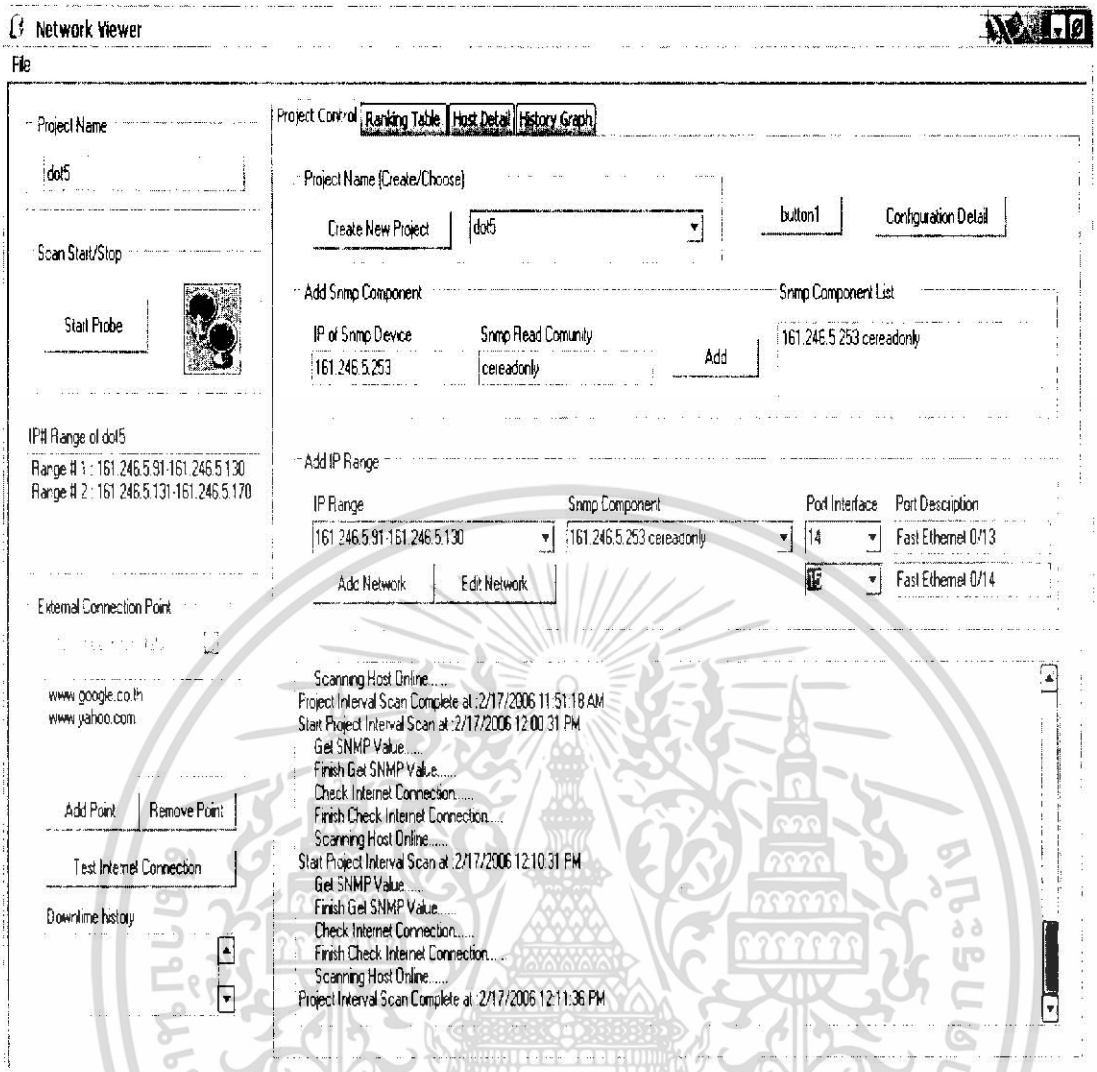


รูปที่ 4.13 แสดงผลการสำรวจสถานะของการเชื่อมต่อระหว่างระบบเครือข่ายกับ Internet

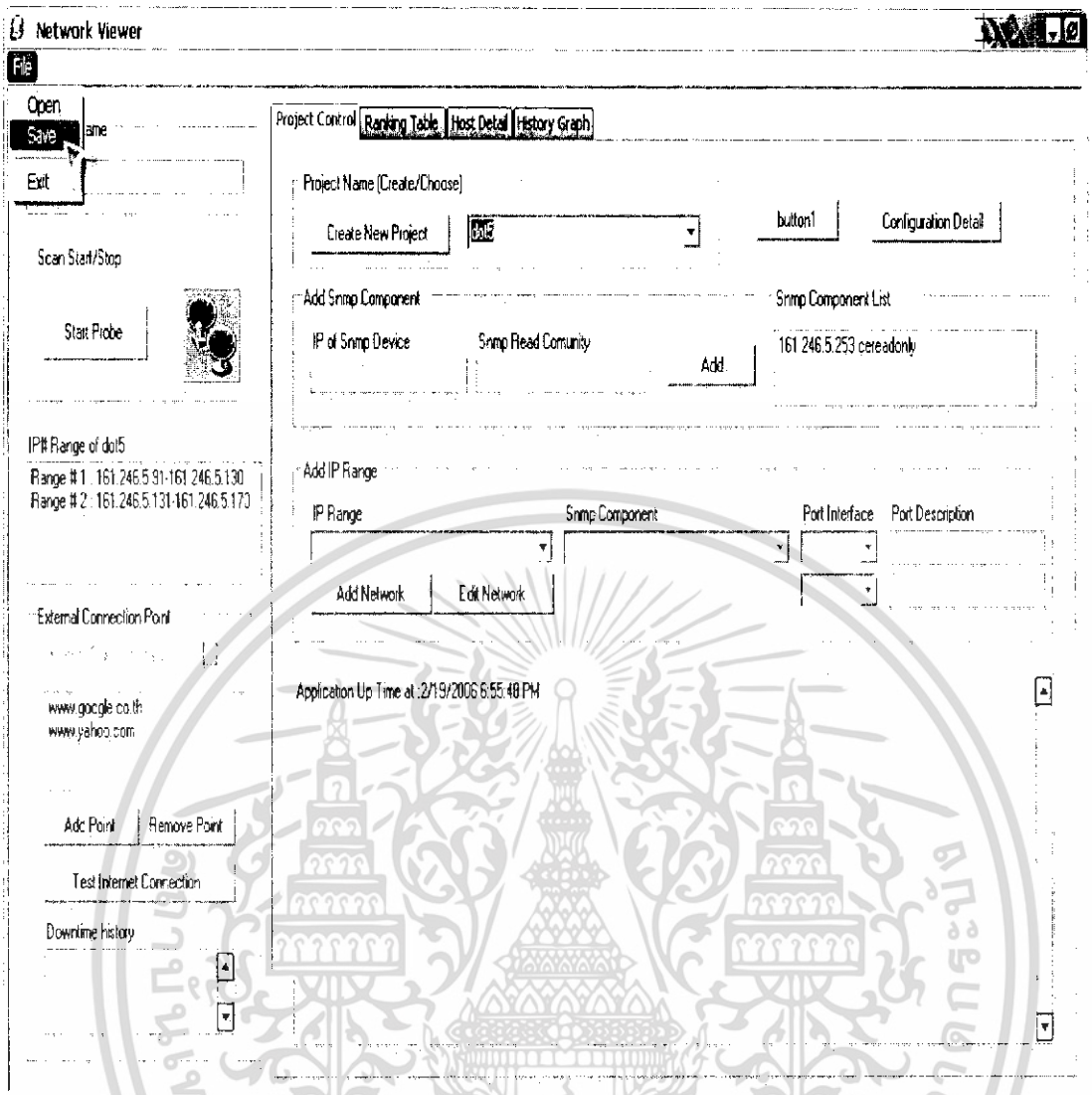
ส่วนการสำรวจข้อมูลเฉพาะเครื่อง โดยยกตัวอย่างทดสอบการ scan พอร์ตที่ระบุ คือ พอร์ต 21-23 และ 80 ของเครื่องผู้ใช้ IP Address 161.246.5.122 ได้ผลดังแสดงในรูป 4.14



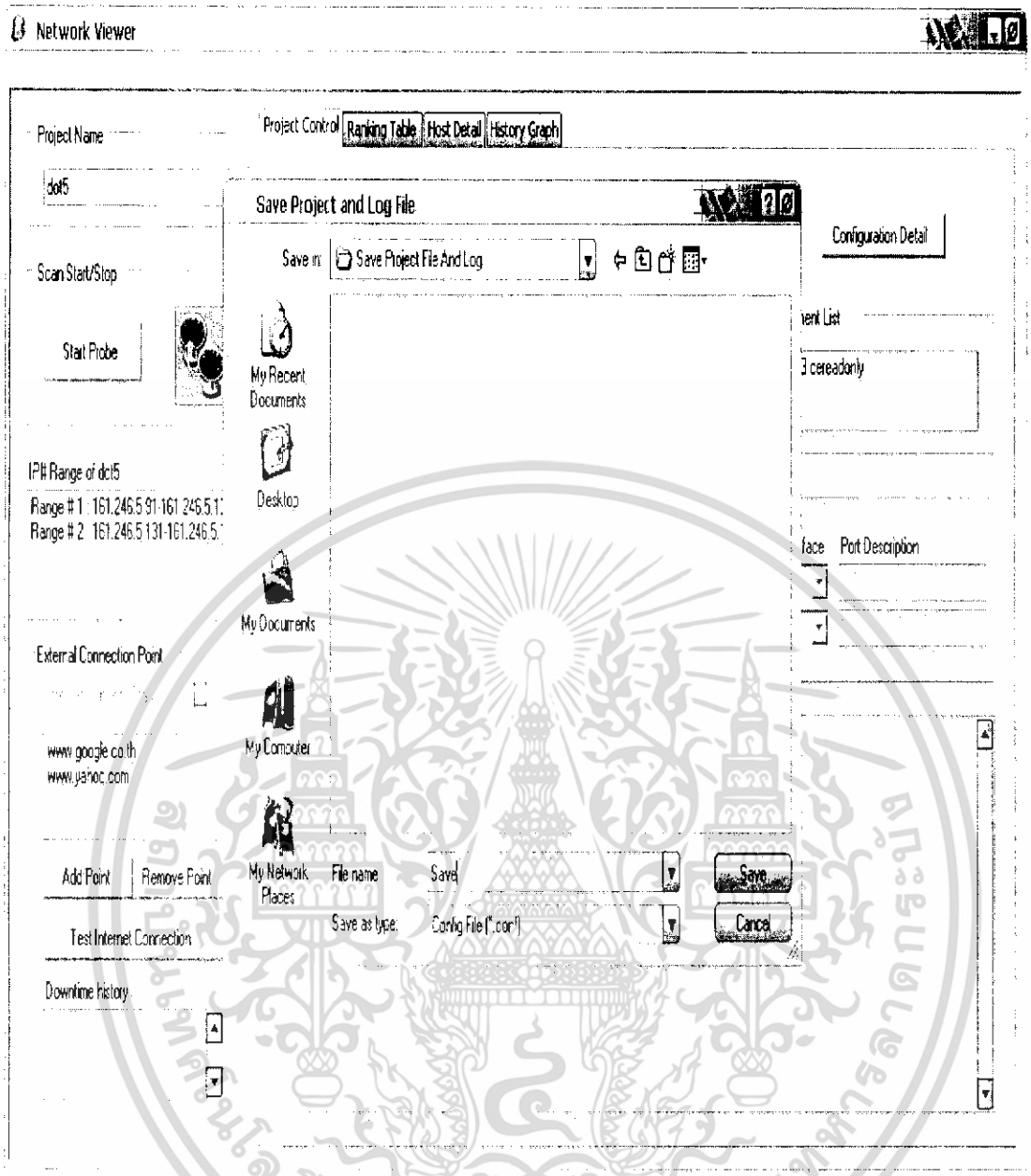
รูปที่ 4.14 ตัวอย่าง แสดงผลการสำรวจข้อมูลเฉพาะเครื่อง โดยscan พอร์ต 21-30 และ 80 ของผู้ใช้ IP Address 161.246.5.122



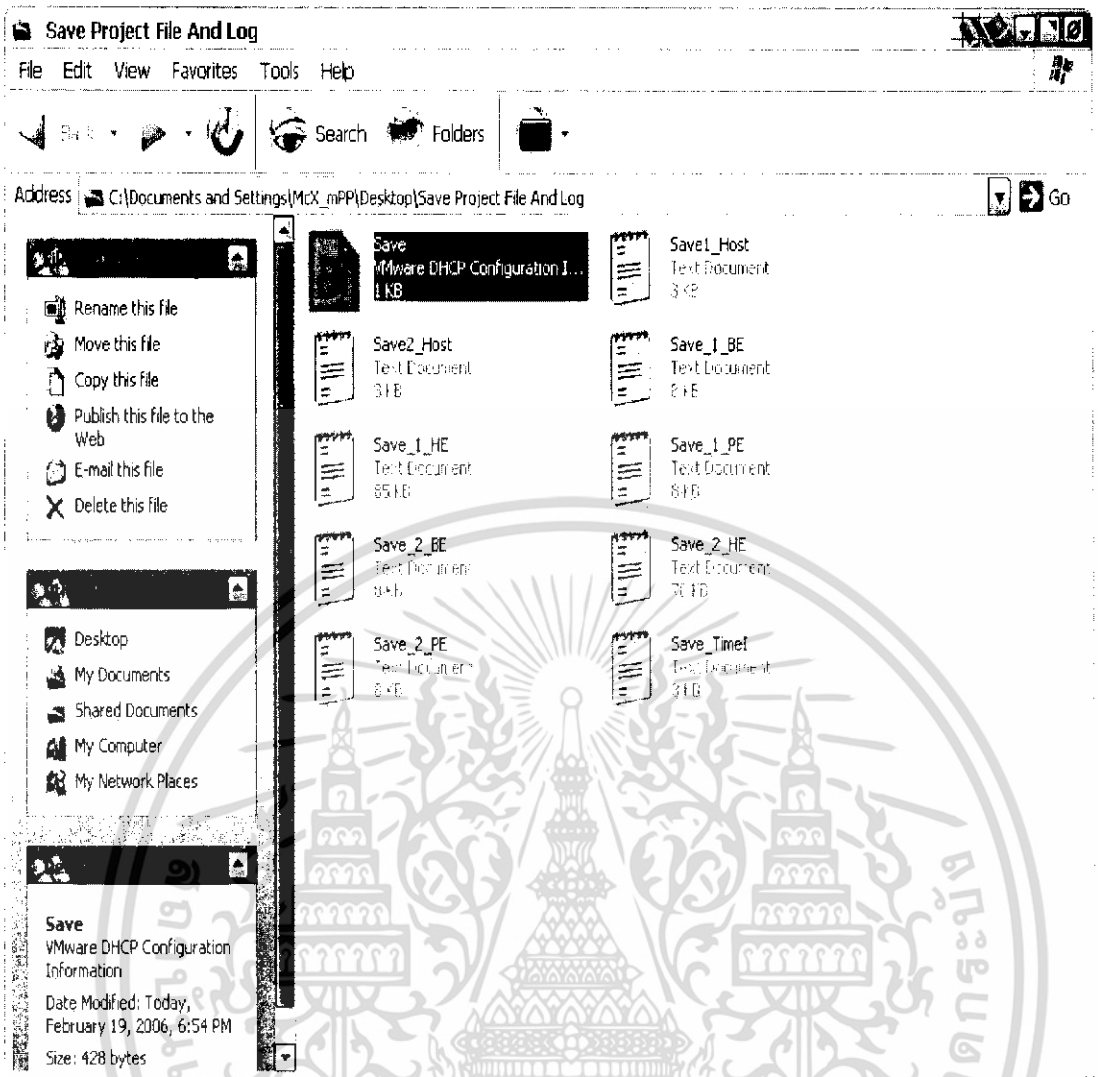
รูปที่ 4.15 แสดงส่วนของ Log File ที่โปรแกรมบันทึกไว้



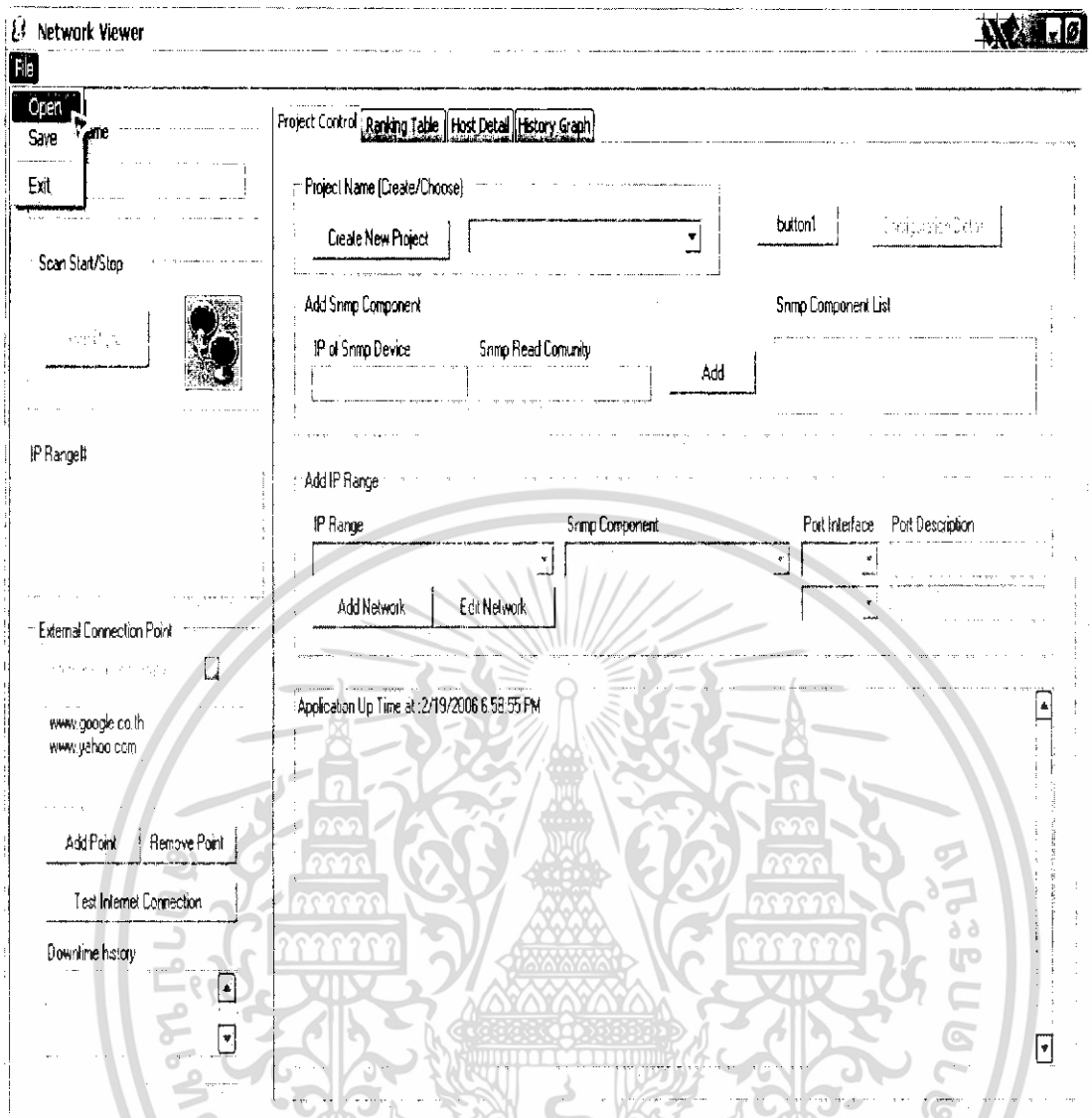
รูปที่ 4.16 แสดงการ Save configuration file และ Log file จากการใช้งานโปรแกรม (1)



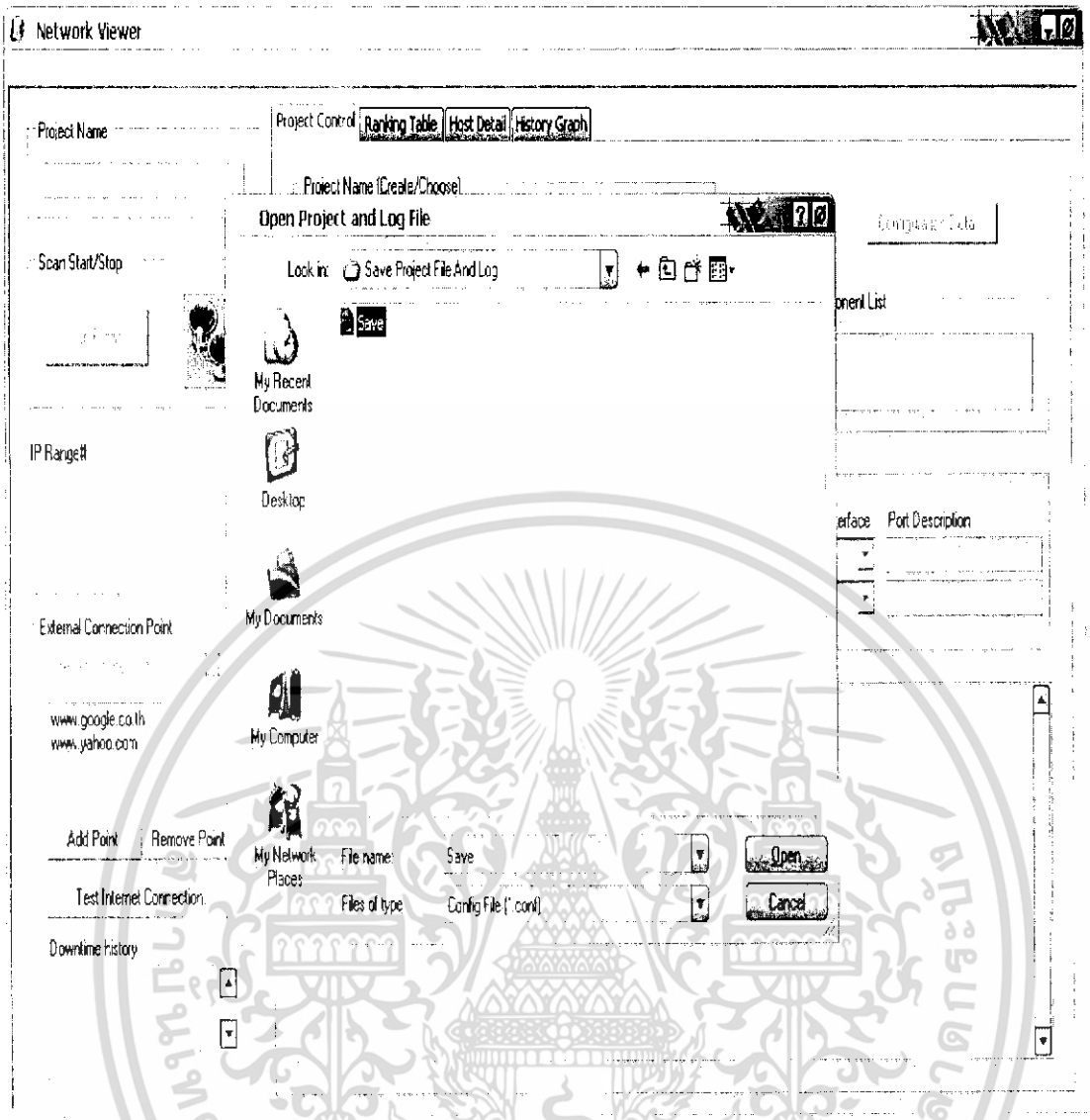
รูปที่ 4.17 แสดงการ Save configuration file และ Log file จากการใช้งานโปรแกรม (2)



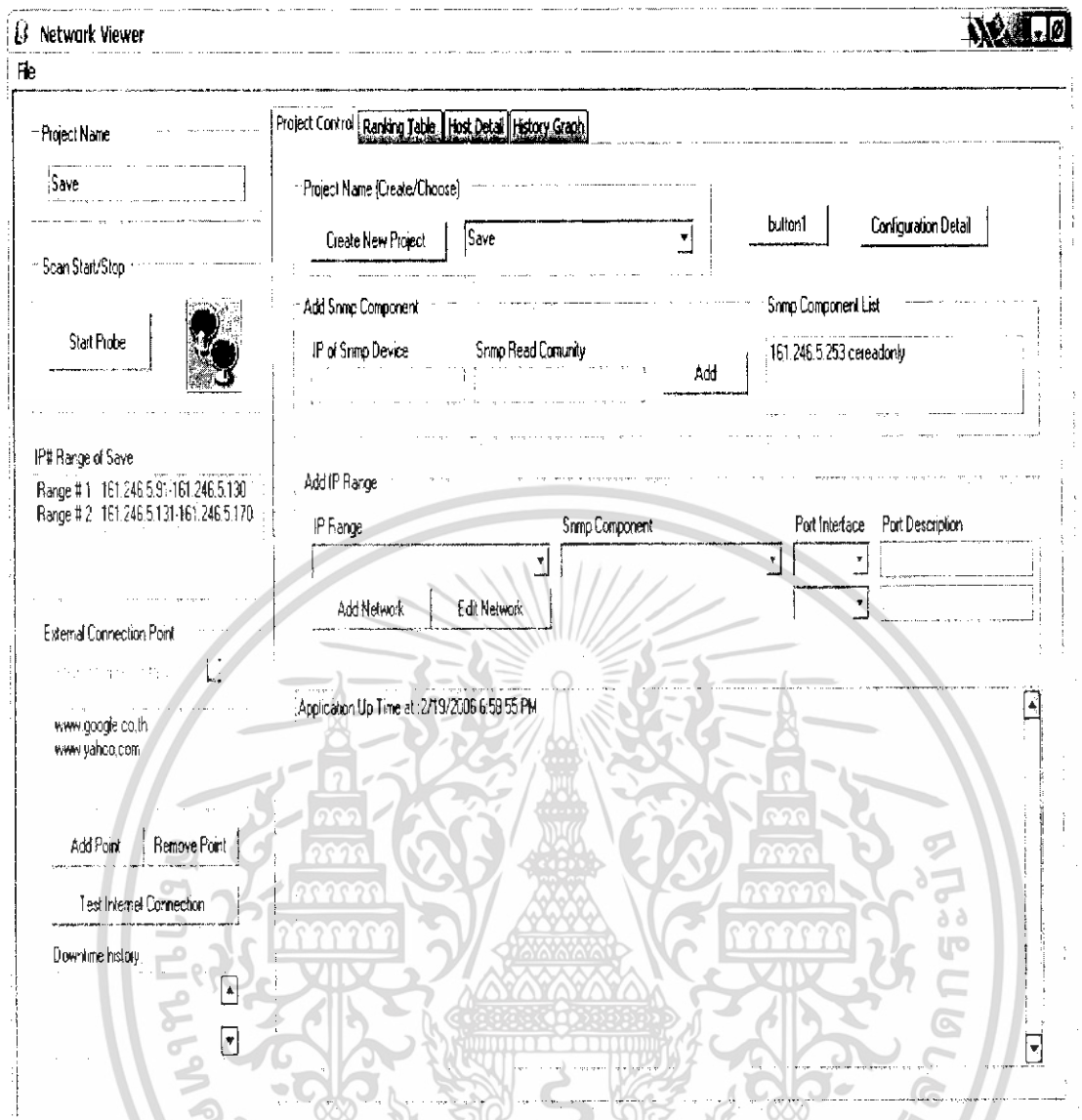
รูปที่ 4.18 แสดงการ Save configuration file และ Log file จากการใช้งานโปรแกรม (3)



รูปที่ 4.19 แสดงการ นำไป Open กับตัวโปรแกรมที่เครื่องอื่น หรือในโอกาสที่ต้องการดูข้อมูล (1)



รูปที่ 4.20 แสดงการ นำไป Open กับตัวโปรแกรมที่เครื่องอื่น หรือในโอกาสที่ต้องการดูข้อมูล (2)



รูปที่ 4.21 แสดงการ นำไป Open กับตัวโปรแกรมที่เครื่องอื่น หรือในโอกาสที่ต้องการดูข้อมูล (3)

บทที่ 5
วิเคราะห์ผลการทดลองและสรุป

5.1 บทสรุป

การใช้งานระบบเครือข่ายคอมพิวเตอร์ (Computer Network) เป็นที่นิยมอย่างแพร่หลาย ดังจะเห็นได้จากปริมาณการใช้งานเครือข่ายและจำนวนผู้ใช้ที่เพิ่มมากขึ้นตลอดเวลาขึ้น ทำให้เกิดความพยายามในสร้างโปรแกรมที่จะนำมาตรวจสอบและวัดตัวแปรสำคัญทั้ง 2 เพื่อนำข้อมูลที่ได้ไปวิเคราะห์ซึ่งนำไปสู่การจัดการและดูแลระบบเครือข่ายให้มีประสิทธิภาพสูงสุดโดยโปรแกรมเหล่านี้ได้แก่ โปรแกรมจัดการระบบเครือข่าย (Network Management Program) หรือ โปรแกรมตรวจสอบระบบเครือข่าย (Network Viewer Program) และโปรแกรมตรวจสอบเครื่องคอมพิวเตอร์ในเครือข่าย (Host Investigate Program) แต่เนื่องด้วยความเปลี่ยนแปลงของสภาพแวดล้อมในเครือข่ายปัจจุบันที่มุ่งเน้นความปลอดภัยมากขึ้นทำให้โปรแกรมเหล่านั้นไม่สามารถทำงานได้อย่างเต็มประสิทธิภาพและถึงแม้โปรแกรมบางโปรแกรมยังคงสามารถทำงานได้แต่ก็มีราคาแพงหรือไม่ก็สามารถแสดงข้อมูลได้เพียงบางส่วนทำให้เป็นหน้าที่ของผู้ดูแลระบบ (Network Administrator) ต้องนำข้อมูลจากโปรแกรมหลายโปรแกรมมาวิเคราะห์ด้วยตนเองก่อให้เกิดความเสียเวลาและอาจเกิดความผิดพลาดได้ ดังนั้น จึงต้องสร้างโปรแกรมที่มีความสามารถในการแสดงข้อมูลที่มีประโยชน์ต่อการดูแลและจัดการเครือข่ายรวมทั้งประมวลผลข้อมูล

โปรโตคอลเอสเอ็มเอ็นพี (SNMP Protocol) เป็นหนึ่งในโปรโตคอลที่มีมานานและอุปกรณ์ทางด้านระบบเครือข่ายเกือบทั้งหมดรองรับซึ่งสามารถบอกถึงปริมาณข้อมูลที่วิ่งผ่านอุปกรณ์ระบบเครือข่ายนั้นๆได้อย่างแน่นอน และการวิธีการพอร์ตแสกน (Port Scan) นอกจากจะสามารถตรวจสอบบริการทางระบบเครือข่าย (Network Service) ได้แล้วยังสามารถนำมาใช้แทนการPing ในการตรวจสอบสถานะของเครื่องคอมพิวเตอร์ในเครือข่ายได้อย่างมีประสิทธิภาพ ดังนั้นการนำวิธีการทั้งสองนี้มารวมไว้ในโปรแกรมเดียวกันและให้ภาระหน้าที่ในการประมวลผลไปถึงการวิเคราะห์ข้อมูลเบื้องต้นเป็นหน้าที่ของโปรแกรมแล้วย่อมลดภาระและความผิดพลาดของผู้ดูแลระบบได้

5.2 วิเคราะห์ผลการทดลอง

ความสามารถเกี่ยวกับระบบเครือข่าย

- สามารถตรวจสอบปริมาณ Traffic ของระบบเครือข่ายย่อยที่กำหนดและบันทึกสถิติได้

- สามารถตรวจสอบสถานะการเชื่อมต่อออกไปยัง Internet และเก็บสถิติไว้ได้

ความสามารถเกี่ยวกับเครื่องคอมพิวเตอร์ในเครือข่าย

- สามารถตรวจสอบสถานะเปิดปิดและเก็บสถิติได้
- สามารถตรวจสอบบริการทางเครือข่ายที่มีการเปิดใช้ได้
- สามารถตรวจสอบได้ว่ามีการติด Trojan หรือไม่

ความสามารถเกี่ยวกับการวิเคราะห์และแสดงผล

- สามารถแบ่งการตรวจสอบออกเป็นระบบเครือข่ายย่อยในช่วง IP Range ที่กำหนดและให้แต่ละช่วงมีการเก็บข้อมูล SNMP จากอุปกรณ์เครือข่ายคนละตัวได้
- สามารถแสดงปริมาณ Traffic ของระบบเครือข่ายทั้งการ Download หรือ Upload หรือทั้ง 2 อย่างในรูปแบบกราฟ 3 มิติตามวันที่กำหนดเพื่อง่ายต่อการเปรียบเทียบระหว่างระบบเครือข่ายย่อยได้
- สามารถแสดงปริมาณ Traffic ทั้งหมดตั้งแต่เริ่มตรวจสอบของระบบเครือข่ายย่อยทั้งการ Download หรือ Upload หรือทั้ง 2 อย่างในรูปแบบกราฟ 3 มิติตามวันที่กำหนดเพื่อง่ายต่อการเปรียบเทียบระหว่างระบบเครือข่ายย่อยได้
- สามารถแสดงปริมาณ Traffic เฉลี่ยของของระบบเครือข่ายทั้งการ Download หรือ Upload หรือทั้ง 2 อย่างในรูปแบบกราฟ 3 มิติเพื่อง่ายต่อการเปรียบเทียบระหว่างระบบเครือข่ายย่อยได้
- สามารถแสดงปริมาณเครื่องคอมพิวเตอร์ที่เชื่อมต่อกับเครือข่ายในรูปแบบกราฟ 3 มิติเพื่อง่ายต่อการเปรียบเทียบระหว่างระบบเครือข่ายย่อยได้
- สามารถแสดงเครื่องคอมพิวเตอร์ที่เชื่อมต่อกับระบบและปริมาณ Traffic ของระบบเครือข่ายย่อยในช่วงเวลาที่กำหนดได้
- สามารถจัดลำดับเครื่องคอมพิวเตอร์ในระบบเครือข่ายย่อยที่มีปริมาณ Traffic เฉลี่ยสูงสุด 5 ลำดับทั้งการ Download และ Upload และทั้ง 2 อย่าง
- สามารถ Save ข้อมูลที่ได้จากการตรวจสอบระบบเครือข่ายเพื่อถ่ายโอนไปแสดงผลโดยนำไป Open กับตัวโปรแกรมที่เครื่องอื่นได้ง่าย
- สามารถแสดง System Log ของโปรแกรมได้

ความสามารถเฉพาะของโปรแกรม

- สามารถทำงานทั้งหมดที่กล่าวมาแม้ในสภาพแวดล้อมที่เครื่องคอมพิวเตอร์ในเครือข่ายมี
- การติดตั้ง Personal Firewall ก็ตาม
- สามารถถ่ายโอนโปรเจ็คการตรวจสอบระบบไปยังเครื่องคอมพิวเตอร์อื่นได้

5.3 ปัญหาอุปสรรคและแนวทางในการแก้ไข

1. อุปกรณ์ในเครือข่ายโดยส่วนใหญ่จะมีการรองรับ SNMP ในระดับพื้นฐานเหมือนกันแต่หากเป็นความสามารถของ SNMP อื่นๆจะรองรับแตกต่างกันขึ้นกับระดับของอุปกรณ์รวมไปถึงยี่ห้อของอุปกรณ์ด้วย จึงไม่สามารถนำความสามารถระดับสูงของ SNMP เช่น การดึงสถิติของการเชื่อมต่อระหว่าง Host, การดึงสถิติของ Traffic ในระดับ Port มาใช้ได้เพราะจะทำให้กลายเป็นโปรแกรมเฉพาะอุปกรณ์รวมไปถึงความที่ SNMP นั้นไม่ค่อยถูกใช้ทำให้มีผู้เข้าใจจริงน้อย การแก้ไขคือมีการดึงสถิติมาเฉพาะในส่วนของ SNMP พื้นฐานและทำให้ในส่วนของ การตั้งค่า SNMP น้อยที่สุดและมีเฉพาะส่วนที่จำเป็นจริงๆ

2. เนื่องจากสภาพแวดล้อมของเครือข่ายมีการใช้งาน Personal Firewall ทำให้การตรวจสอบอ้างอิงอยู่บนการ Scan Port มีผลให้การตรวจสอบระบบในแต่ละครั้งใช้เวลานานและมีความเปราะบางหากมีการโปรแกรมอื่นๆเข้ามาทำงานพร้อมๆกับการตรวจสอบระบบอาจทำให้ข้อมูลผิดพลาดหรือสูญหายได้ การแก้ไขคือมีการตัดความสามารถต่างๆของโปรแกรมชั่วคราวและไม่ใช้โปรแกรมที่มีการใช้งานระบบเครือข่ายหนักๆในช่วงที่ทำการตรวจสอบระบบ

3. เนื่องจากสภาพแวดล้อมของเครือข่ายมีการใช้งาน Personal Firewall ทำให้ไม่สามารถตรวจสอบ Traffic ต่อเครื่องคอมพิวเตอร์ในระบบได้โดยตรง การแก้ไขคืออาศัยค่าเฉลี่ยระหว่าง Traffic ของเครือข่ายย่อยกับจำนวนเครื่องที่มีสถานะเชื่อมต่อกับเครือข่ายในขณะนั้นเพื่อนำมาจัดลำดับเครื่องที่มีการใช้งานระบบเครือข่ายซึ่งจะได้ความแม่นยำระดับหนึ่ง

4. เนื่องจากวิธีการจัดวางหรือระบบโครงสร้างของระบบเครือข่ายในที่ต่างๆไม่เหมือนกันจึงอาจเป็นปัญหาในการตั้งค่าต่างๆสำหรับการดึงค่า SNMP ในช่วงที่ตรวจสอบระบบได้ การแก้ไขคือมีการตั้งค่าของระบบเครือข่ายย่อยได้ถึงระดับ Port ของอุปกรณ์และมีการแสดง Port ทั้งหมดของอุปกรณ์ออกมาเพื่อป้องกันการ ตั้งค่าผิด

5.4 ข้อจำกัดของโปรแกรม

1. ลำดับที่ได้จากโปรแกรมอาจมีความผิดพลาดเนื่องจากความจำกัดทางข้อมูล
2. การตั้งค่าช่วงของการตรวจสอบระบบถูกจำกัดไว้ที่ 10 นาทีเนื่องจากระยะเวลาของการตรวจสอบระบบในแต่ละครั้งมีเวลานานและมีการใช้งานทรัพยากรของเครื่องสูง
3. จำเป็นต้องมีการตรวจสอบระบบเป็นระยะเวลาหนึ่งจึงจะได้ข้อมูลด้านสถิติที่ถูกต้องมากขึ้น
4. การตรวจสอบระบบอาจมีปัญหาหากเครื่องที่ตั้งมีโปรแกรม Personal Firewall อยู่

5. ไม่สามารถตรวจสอบเครื่องปลายทางที่ทำการปิดพอร์ต เช่น เครื่องปลายทางที่ใช้ระบบปฏิบัติการ Linux และทำการปิดเซอร์วิสทั้งหมดได้
6. ไม่สามารถตรวจสอบการเปิดโปรแกรม P2P หรือ เกมส์ที่ทำการเปลี่ยนพอร์ตพื้นฐานของโปรแกรมได้
7. หากต้องการให้โปรแกรมสามารถเข้าไปดึงข้อมูล SNMP ในสวิตซ์หรือ เราเตอร์ได้ ผู้ดูแลระบบจำเป็นที่จะต้องทำการเปิดใช้บริการ SNMP ที่สวิตซ์หรือ เราเตอร์ตัวนั้นก่อน และทราบ SNMP Parameters(Read/Write Community) ของอุปกรณ์ตัวนั้น

5.5 แนวทางการพัฒนาต่อ

1. พัฒนาในส่วนของระบบการ Scan Port ขึ้นมาเองเพื่อนำมาใช้งานในโปรแกรมโดยเฉพาะเพื่อลดความล่าช้าและความผิดพลาดที่อาจจะเกิดขึ้นจากการเรียกใช้งานโปรแกรม Nmap
2. พัฒนาในส่วนของ Database ให้มีการ Share แบบ Online และมีการนำข้อมูลจากเครื่องอื่นๆมาวิเคราะห์ร่วมกันเพื่อให้ได้ข้อมูลที่ชัดเจนยิ่งขึ้น

บรรณานุกรม

[1] สุวัฒน์ ปุณณชัยยะ, ต้น ตันฑ์สุทธิวงศ์, สุพจน์ ปุณณชัยชนะ, "**เปิดโลกของ TCP/IP และโปรโตคอลของอินเตอร์เน็ต**", โปรวิชั่น, 2545

[2] เรืองไกร รังสิพล, "**เจาะระบบ TCP/IP จุดอ่อนของโปรโตคอลและวิธีป้องกัน**", โปรวิชั่น , 2544

[3] เรืองไกร รังสิพล : "**เปิดโลก Firewall และ Internet Security**", โปรวิชั่น , 2545

[4] สุรศักดิ์ สงวนพงษ์ : "**สถาปัตยกรรม และโปรโตคอลทีซีพี/ไอพี**" , ซีเอ็ดยูเคชั่น , 2545

[5] Mark a. Miller : "**Managing Internet with SNMP** " , M&T BOOKS , 1993

[6] William Stallings : " **SNMP ,SNMPv2 ,and RMON Practical Network Management** " , Addison Wesley, 1996

[7] Searchsmb.com "Ping Sweep ." [Online]. Available : http://searchsmb.techtarget.com/sDefinition/0,,sid44_gci802721,00.html

[8] Security.com "Ping Sweep." [Online]. Available : http://www.linuxsecurity.com/content/view/117111/141/

[9] Insecure.org "ICMP Ping Sweep Detection." [Online]. Available : http://seclists.org/lists/focus-ids/2003/Oct/0070.html

[10]Insecure.org "Nmap network security scanner man page." [Online]. Available : http://www.insecure.org/nmap/data/nmap_manpage.html

[11]Insecure.org "Nmap Remote OS Detection." [Online]. Available : http://www.insecure.org/nmap/nmap-fingerprinting-article.html

[12]Thaicert "เทคนิคการ Scan Port และวิธีป้องกัน." [Online]. Available :
http://thaicert.nectec.or.th/paper/auditing/portscan.php