รายงานโครงการวิจัย
ฉบับสมบูรณ์

การตรวจวัดคุณลักษณะการใช้งานเครือข่ายอินเทอร์เน็ต

Measurement of Internet Traffic Characteristics

อัครินทร์ คุณกิตติ
AKHARIN KHUNKITTI

แสงเพ็ชร พระฉาย
SANGPETCH PRACHAI

เงินรายได้ประจำปีงบประมาณ 2547
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อโครงการวิจัย (ภาษาไทย)	**การตรวจวัดคุณลักษณะการใช้งานเครือข่ายอินเทอร์เน็ต**
(ภาษาอังกฤษ)	**Measurement of Internet Traffic Characteristics**

ชื่อหัวหน้าโครงการวิจัย	นาย อัครินทร์ คุณกิตติ
ชื่อผู้วิจัยร่วม	นาย แสงเพ็ชร พระฉาย
ชื่อหน่วยงานที่สังกัด	คณะเทคโนโลยีสารสนเทศ
	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

บทคัดย่อ

งานวิจัยนี้ได้นำเสนอกรอบและแนวคิดในการออกแบบเครื่องมือที่ใช้ตรวจวัดคุณลักษณะการสื่อสารข้อมูลบนเครือข่ายอินเทอร์เน็ตด้วยวิธีการที่ไม่ใช้การรับส่งข้อมูลระหว่างคู่ของการสื่อสาร ลักษณะการตรวจวัดใช้วิธีตรวจจับและกรองเฉพาะข้อมูลส่วนหัวของไอพีแพ็กเก็ตเวอร์ชั่นที่ 4 จากทั้งเครือข่ายภายในและภายนอก นำไปแบ่งแยกและจัดเก็บเป็นตารางข้อมูลที่เหมือนกัน 3 ชนิด คือ ตารางข้อมูลที่มีทิศทางการสื่อสารขาออก ตารางข้อมูลที่มีทิศทางการสื่อสารขาเข้า และตารางข้อมูลอื่น ๆ ตารางข้อมูลที่รวบรวมได้จะนำไปใช้กับการวิเคราะห์เวลาการสื่อสารไปกลับของข้อมูลชนิด ICMP และ TCP และวิเคราะห์ความล่าช้าและความสูญเสียของข้อมูลชนิด ICMP, TCP และ UDP โดยพิจารณาคู่แพ็กเก็ตในตารางข้อมูลที่มีความสัมพันธ์กัน ได้แก่ เวลาการสื่อสารไปกลับพิจารณาจากตารางข้อมูลที่มีทิศทางการสื่อสารตรงข้ามกันแต่อยู่ฝั่งเครือข่ายเดียวกัน ส่วนความล่าช้าและความสูญเสียพิจารณาจากตารางข้อมูลที่มีทิศทางการสื่อสารทางเดียวกันแต่อยู่ฝั่งเครือข่ายตรงข้ามกัน ในส่วนของการทดลองได้พัฒนาเครื่องมือและเปรียบเทียบผลการตรวจวัดเวลาไปกลับและความสูญเสียกับคำสั่ง ping และเปรียบเทียบความล่าช้ากับการหน่วงเวลาด้วย IPFirewall นอกจากนี้ได้นำเสนอผลการวิเคราะห์ความถูกต้องของการตรวจวัดความยาวและเวลาระหว่างการมาของแพ็กเก็ตเปรียบเทียบกับเครื่องมือที่พัฒนาจาก Corallib Library ผลการตรวจวัดในทุกหน่วยเวลาด้วยเครื่องมือวิจัยมีความแตกต่างกันเพียงเล็กน้อยกับเครื่องมือวัดทุกชนิดที่นำมาใช้ในการเปรียบเทียบกัน

Abstract

This research will present a framework and implementation of designed measurement tool for Internet traffic characteristics. The measurement has been designed using a passive traffic measurement method for capturing and filtering packets of Internet Protocol version 4 from internal and external networks. The collected data have been classified and stored into three types of tables as forward, reverse and other direction tables. The data in tables has been analyzed for response time of ICMP and TCP, delay and loss of ICMP, TCP and UDP packets. The response time has been derived from two opposite direction tables. The delay and loss has been derived from two same direction tables. The measurement tool has been developed on FreeBSD system and tested compared to the ping tool and emulated system under testing using IPFirewall of FreeBSD. There were also experiment for analysis of packet length and inter-arrival time compared to the Corallib Library tool. The results of experiments have been shown that the proposed measurement system has the same level of accuracy as the compared tools.

สารบัญ

หน้า

สารบัญ(ต่อ)

หน้า

สารบัญตาราง

ตารางที่ หน้า

สารบัญภาพ

ภาพที่ หน้า

สารบัญภาพ(ต่อ)

ภาพที่ หน้า

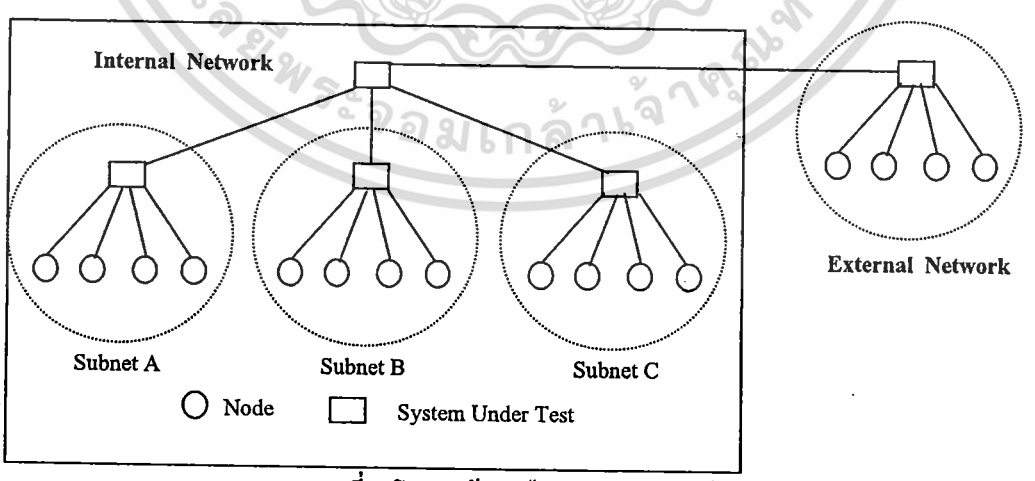
สารบัญภาพ(ต่อ)

ภาพที่ หน้า

บทที่ 1
บทนำ

1.1 ความเป็นมาของงานวิจัย

ระบบเครือข่ายอินเตอร์เน็ตเป็นเครือข่ายการสื่อสารข้อมูลที่มีขนาดใหญ่ การขยายและเชื่อมต่อเครือข่ายย่อย(Subnet)ย่อมส่งผลให้เกิดการแลกเปลี่ยนข้อมูลที่เพิ่มขึ้นตามไปด้วยและเป็นปัจจัยที่สำคัญอย่างหนึ่งที่ผู้บริหารเครือข่ายควรตรวจสอบหรือวิเคราะห์ประสิทธิภาพการสื่อสารข้อมูลอย่างต่อเนื่อง เพื่อปรับปรุงจุดบกพร่องของการสื่อสารข้อมูลให้อยู่ในสภาพที่เหมาะสมอยู่เสมอ ในสภาพของการจัดระบบเครือข่ายจริงภายในองค์กรหนึ่งสามารถเปรียบเทียบได้กับเครือข่ายขนาดใหญ่ที่ถูกย่อส่วนให้เล็กลง แต่ละองค์กรจะได้รับหมายเลขไอพี(Internet Protocol Address)อย่างน้อยหนึ่งชุดสำหรับการสื่อสารข้อมูลไปยังภายนอกเครือข่ายขององค์กร โดยผู้ดูแลขององค์กรต้องเป็นผู้จัดสรรหมายเลขไอพีเป็นชุดย่อยให้กับแต่ละฝ่ายภายในองค์กรตามความเหมาะสม ดังนั้นการสื่อสารข้อมูลภายในองค์กรจึงมีลักษณะเป็นลำดับชั้น(Hierarchy) โดยมีสมาชิกอย่างน้อย 1 โนด(Node)ทำหน้าที่รวบรวมและรับส่งข้อมูลไปในเส้นทางตามที่สมาชิกของโนดต้นทางกำหนด และขอนิยามกลุ่มของโนดที่ได้รับความสนใจในการสื่อสารข้อมูลผ่านแต่โนดเหล่านี้ว่าตัวระบบ(System Under Test) โดยนิยาม โนด หมายถึง อุปกรณ์อิเลคทรอนิกส์ที่ทำหน้าที่สื่อสารข้อมูลผ่านเครือข่ายอินเทอร์เน็ต เช่น คอมพิวเตอร์, เราท์เตอร์(Router Device) หรือ สวิชชิ่ง(Switching Device) เป็นต้น ดังแสดงลักษณะการสื่อสารข้อมูลโดยทั่วไปได้ดังภาพที่ 1



ภาพที่ 1 โครงสร้างเครือข่ายภายในองค์กร

จากภาพที่ 1 ปัญหาของการสื่อสารข้อมูลจะไม่เกิดขึ้นถ้าตัวระบบมีประสิทธิภาพเพียงพอและไม่สร้างปัญหาต่อการให้บริการรับส่งข้อมูลไปยังโนดปลายทาง แต่โดยทั่วไปแล้วมักไม่เป็นเช่นนี้ เนื่องจากพฤติกรรม การสื่อสารข้อมูลของแต่โนดไม่เท่าเทียมกัน ปริมาณของข้อมูลที่แต่ละโนดส่งผ่านตัวระบบ อาจทำให้เกิดความหนาแน่นขึ้นที่โนดใด ๆ บนตัวระบบ หรือเรียกปัญหานี้ว่า คอขวด(Bottle Neck) ผลกระทบของปัญหานี้ก็จะส่งผลกระทบต่อเนื่องจากโนดหนึ่ง ไปยังอีกโนดหนึ่ง และอาจมีผลไปถึงโนดต่าง ๆ ของผู้ให้บริการเครือข่ายอินเตอร์เน็ต(Internet Service Provider)ที่หน่วยงานต่อเชื่อมอยู่ ปัญหาคอขวดที่เกิดขึ้นนี้อาจจะก่อให้เกิดความล่าช้าหรือความสูญเสียข้อมูลขึ้นอย่างต่อเนื่องและขยายเป็นวงกว้างไปยังหน่วยงานต่าง ๆ ซึ่งขัดกับความต้องการของผู้ใช้ที่มีความต้องการให้ข้อมูลของตนส่งไปถึงผู้รับได้อย่างรวดเร็วและสมบูรณ์

ดังนั้นการออกแบบเครือข่ายที่ได้รับการวิเคราะห์อย่างเหมาะสมแล้วก็จะนำมาซึ่งค่าใช้จ่ายและประสิทธิภาพการสื่อสารข้อมูลที่เหมาะสมด้วย ซึ่งในความเป็นจริงการวิเคราะห์การใช้งานทรัพยากรหรืออุปกรณ์เครือข่ายอาจได้รับการวางแผนอย่างเหมาะสมแล้ว แต่ก็อาจเกิดปัญหานี้ขึ้นได้หากผู้ดูแลขาดการตรวจสอบและวิเคราะห์คุณลักษณะการสื่อสารภายในเครือข่ายอย่างต่อเนื่อง ซึ่งพฤติกรรมและความต้องการของผู้ใช้เครือข่ายนั้นสามารถเปลี่ยนแปลงได้ตลอดเวลา และผู้ดูแลระบบก็ไม่สามารถวิเคราะห์ได้จากการสอบถามผู้ใช้เพียงอย่างเดียวแต่ต้องสามารถวิเคราะห์ได้จากพฤติกรรมการสื่อสารข้อมูลจริงบนเครือข่ายเพื่อนำไปใช้วางแผนและออกแบบอุปกรณ์เครือข่ายที่เหมาะสม ดังนั้นการตรวจวัดการสื่อสารข้อมูลจริงบนเครือข่ายจึงเป็นความสำคัญอย่างยิ่งกับทุกหน่วยงานที่ต้องการสื่อสารข้อมูลที่มีสิทธิภาพ

1.2 ความมุ่งหมายและวัตถุประสงค์

เพื่อให้การวางแผนและออกแบบเครือข่ายได้รับข้อมูลที่ถูกต้องจากการสื่อสารข้อมูลจริงบนเครือข่าย ในงานวิจัยนี้จึงขอเสนอวิธีการเพื่อนำไปสู่การวัดและตรวจสอบการสื่อสารข้อมูลที่จำเป็นต่อการวางแผนจัดการเครือข่าย โดยมีแนวคิดในการศึกษาและวิจัยปัญหาที่สำคัญดังนี้

1.2.1 ออกแบบวิธีการตรวจวัดและวิเคราะห์การสื่อสารข้อมูลบนเครือข่ายอินเทอร์เน็ตด้วยวิธีการทางสถิติ เพื่อแจกแจงคุณลักษณะหรือพฤติกรรมของข้อมูล ที่มีผลกระทบต่อการสื่อสารผ่านตัวระบบ โดยมีความมุ่งหมายในการตรวจวัดข้อมูลต่าง ๆ ดังนี้

- ความยาวแพ็กเก็ต(Packet Size)
- เวลาระหว่างการมา(Inter-Arrival Time)
- เวลาการสื่อสารข้อมูลไปและกลับ(Response Time)

- ความล่าช้า(Delay)
- ความสูญเสียข้อมูล(Loss)

1.2.2 ศึกษาปัญหาและนำเสนอวิธีการประเมินความถูกของการตรวจวัดที่ออกแบบ

1.2.3 สร้างเครื่องมือและทดลองตรวจวัดจริงบนระบบเครือข่าย

1.3 สมมุติฐานของการศึกษาวิจัย

ผลจากการศึกษาและวิจัยสามารถสร้างเครื่องมือเพื่อการวิเคราะห์ข้อมูลเชิงสถิติที่ถูกต้อง สามารถนำข้อมูลไปใช้เพื่อการวางแผนจัดเตรียมเครือข่ายให้เกิดความเหมาะสมและมีประสิทธิภาพ ช่วยติดตามความผิดพลาดและขจัดปัญหาที่เกิดขึ้นจากการสื่อสารข้อมูลโดยตัวระบบ นอกจากนี้ คุณลักษณะการสื่อสารข้อมูลเชิงตัวเลขที่ตรวจวัดได้จากเครื่องมือสามารถสามารถนำไปใช้ร่วมกับวิธีการตรวจวัดโดยเชิงประมาณค่า(Approximation) หรือ การจำลองระบบ(Simulation)ได้ ซึ่งจะทำให้ผลของการออกแบบเครือข่ายมีความสมบูรณ์ยิ่งขึ้น

1.4 ทฤษฎีและแนวคิดที่นำมาใช้กับงานวิจัย

งานวิจัยได้ใช้ทฤษฎีในหลายส่วนรวมกันเพื่อนำไปสู่การสร้างเครื่องมือวัดบนระบบจริง โดยมีการศึกษาทฤษฎีที่สำคัญหลัก ๆ คือ ทฤษฎีการสื่อสารข้อมูล ที่ต้องศึกษาและวิเคราะห์รูปแบบการสื่อสารข้อมูลบนเครือข่ายอินเทอร์เน็ตทั้ง TCP, UDP และ ICMP โปรโตคอล ซึ่งแต่ละชนิดจะมีคุณลักษณะและขนาดของแพ็กเก็ตที่แตกต่างกัน ดังนั้นผลของเวลาในการสื่อสารข้อมูลจึงมีความแตกต่างกันตามพฤติกรรมการสื่อสารข้อมูลในแต่ละชนิดด้วย ทฤษฎีการโปรแกรมภาษา C บนระบบปฏิบัติการ FreeBSD โดยศึกษาการเขียนโปรแกรมให้ทำงานแบบ Multi-processing, การจัดเก็บและเรียกใช้ข้อมูลบนหน่วยความจำร่วม(Share Memory), การรับส่งข้อมูลผ่านเครือข่ายแบบ Socket, การโปรแกรมเพื่อจัดเก็บข้อมูลตามทฤษฎีโครงสร้างข้อมูลแบบ Binary Tree, การโปรแกรมร่วมกับ Libpcap และ Coral Library เพื่อตรวจจับข้อมูล ทฤษฎีวิเคราะห์ข้อมูลเชิงสถิติ โดยศึกษาวิธีกำหนดคุณลักษณะข้อมูลที่รวบรวมได้จากการตรวจจับบนเครือข่าย ปัจจุบันการสร้างเครื่องมือตรวจวัดประสิทธิภาพการสื่อสารข้อมูลจริงบนเครือข่ายมีอยู่ 2 วิธีที่นิยมใช้คือ การวัดที่ต้องใช้การรับส่งข้อมูลระหว่างคู่สื่อสาร(Active Traffic Measurement)และการวัดที่ไม่ใช้การรับส่งข้อมูลระหว่างคู่สื่อสาร (Passive Traffic Measurement) เครื่องมือวัดที่มีประสิทธิภาพจะต้องไม่ก่อให้เกิดผลกระทบหรือเพิ่มภาระงานให้กับระบบในขณะทำการวัด นอกจากนี้ผลลัพธ์ที่ได้ต้องสะท้อนถึงความเป็นจริงของการสื่อสารในสภาวะที่เป็นปัจจุบันและต่อเนื่อง ดังนั้นในงานวิจัยนี้จึงขอเสนอวิธีการสร้างเครื่องมือวัด

ในแบบที่ 2 ซึ่งเป็นวิธีที่ได้รับการนำไปใช้ในการสร้างเครื่องมืออย่างแพร่หลาย ตัวอย่าง เช่น การวัดข้อมูลโฟลว(Data Flow) ด้วย NetTraMet, NetFlow, Cflowd, FlowScan และ CoralReef และนอกจากข้อมูลโฟลวแล้วยังมีข้อมูลอื่นอีกหลายชนิดที่จะช่วยสนับสนุนการวิเคราะห์และช่วยสะท้อนปัญหาที่เกิดขึ้นกับครือข่ายการสื่อสารได้ เช่น ความยาวแพ็กเก็ต(Packet Length), เวลาระหว่างการมา(Inter-Arrival Time), เวลาการสื่อสารข้อมูลไปกลับ(Response Time), ความล่าช้า(Delay) และความสูญเสีย (Loss) ซึ่งข้อมูลใน 2 ลำดับแรกสามารถตรวจวัดได้ด้วย NetTraMet หรือ CoralReef ที่พัฒนาด้วย Corallib Library และพิจารณาเครื่องมือตรวจวัดเวลาการสื่อสารไปกลับได้จากคำสั่ง ping ที่ใช้ในการรับส่งข้อมูลชนิด ICMP โปรโตคอล ซึ่งวิธีการของคำสั่ง ping เป็นวิธีการวัดซึ่งจัดอยู่ในรูปแบบแรก คือ ใช้การตรวจวัดจากการรับส่งข้อมูลระหว่างคู่สื่อสาร(Active Traffic Measurement)

1.5 ขอบเขตของงานวิจัย

งานวิจัยนี้ได้เสนอผลงานวิจัยเพื่อออกแบบวิธีตรวจวัดและพัฒนาเครื่องมือวัดสำหรับวิเคราะห์คุณลักษณะการสื่อสารข้อมูลผ่านตัวระบบ ผลสรุปของงานวิจัยจะอภิปรายคุณลักษณะและปัญหาของการสื่อสารข้อมูลผ่านตัวระบบ โดยพิจารณากับตัวระบบที่มี 2 Interface และมีลักษณะการสื่อสารข้อมูลผ่านตัวระบบเป็นแบบพื้นฐานคือ รับและส่งต่อ(Store and Forward) โดยใช้ไอพีปลายทาง (Destination IP Address) เป็นเครื่องมือในการจำแนกทิศทางการสื่อสารข้อมูลในแต่ละฝั่งจาก Interface โดยที่แต่ละโนดบนตัวระบบต้องไม่มีระบบการจัดการไอพีแบบ NAT (Network Address Translation) ข้อมูลที่ผ่านเข้าและออกจากตัวระบบจะกรองเฉพาะแพ็กเก็ตที่มีชนิดของส่วนหัวเป็นไอพีเป็นรุ่นที่ 4(IP Header Version 4) และนำไปวิเคราะห์เฉพาะข้อมูลที่มีชนิดโปรโตคอลเป็น ICMP, TCP และ UDP เท่านั้น การทดลองสร้างเครื่องมือวัดจัดทำขึ้นบนระบบปฏิบัติการ FreeBSD 4.6 Release โดยใช้วิธีคำนวณเวลาของแพ็กเก็ต(Packet Time Stamp)ทั้งขาเข้าและออกจากตัวระบบด้วยคอมพิวเตอร์ที่ติดตั้งเครื่องมือวัดจากเครื่องเดียวกัน ดังนั้นความน่าเชื่อถือในการตรวจจับข้อมูลจึงขึ้นอยู่กับเวลาของเครื่องคอมพิวเตอร์และการตรวจจับข้อมูลโดย LIBPCAP Library ที่เรียกใช้และโปรแกรมด้วยภาษาซี (C Language)

1.6 ขั้นตอนการศึกษาวิจัย

การวิจัยนี้มีขั้นตอนการศึกษาและดำเนินการดังนี้

1.6.1 ศึกษาวิธีการตรวจวัดโดยเครื่องมือที่มีใช้ในปัจจุบัน

1.6.2 ออกแบบวิธีการตรวจวัดข้อมูลในงานวิจัย

1.6.3 สร้างเครื่องมือวัด

1.6.4 ทดลองและวิเคราะห์ความถูกต้องด้วยการเปรียบเทียบกับเครื่องมือชนิดอื่น ๆ

1.6.5 สรุปการวิจัย

บทที่ 2
การสื่อสารและการตรวจวัดข้อมูล

2.1 บทนำ

ผู้ใช้ ผู้บริหาร และนักออกแบบระบบคอมพิวเตอร์ต่างให้ความสนใจในการตรวจวัดประสิทธิภาพ โดยมีเป้าหมายที่จะได้รับหรือจัดเตรียมระบบคอมพิวเตอร์ที่มีประสิทธิภาพสูงแต่มีค่าใช้จ่ายต่ำด้วยกันทั้งสิ้น ผลการตรวจวัดต้องสามารถนำไปใช้ได้อย่างต่อเนื่องและทันต่อสถานภาพปัจจุบัน อีกทั้งมีผลสืบเนื่องให้สามารถขยายขีดความสามารถไปในอนาคตได้ ปัจจุบันคอมพิวเตอร์ส่วนบุคคล(Personal Computer)ได้มีการพัฒนาให้มีประสิทธิภาพที่สูงขึ้นและได้ทำการเชื่อมต่อให้เกิดการแลกเปลี่ยนการสื่อสารกันจนมีขนาดใหญ่เรียกว่า "เครือข่ายอินเทอร์เน็ต" การสื่อสารบนเครือข่ายใช้มาตรฐานเดียวกันเรียกว่า TCP/IP(Transmission Control Protocol / Internet Protocol) และมี "แพ็กเก็ต(Packets)" ทำหน้าที่ในการลำเลียงข้อมูลระหว่างคู่สื่อสารซึ่งอาจผ่านสายสัญญาณหลากหลายชนิด ปัจจุบันอัตราการเชื่อมต่อและแลกเปลี่ยนข่าวสารได้เพิ่มขึ้นอย่างต่อเนื่อง ปริมาณความต้องการของแต่ละหน่วยงานก็แตกต่างกัน ดังนั้นเทคโนโลยีการตรวจวัดเครือข่ายจึงเป็นปัจจัยหนึ่งที่ต้องนำมาใช้ในการพัฒนาประสิทธิภาพการสื่อสารของหน่วยงาน

การจัดเครือข่ายอาจทำขึ้นเฉพาะองค์กร(Campus Network) หรือเชื่อมต่อไปยังเครือข่ายอื่น ๆ ผ่านผู้ให้บริการเครือข่ายสาธารณะ(public Internet Service Provider) อุปกรณ์เครือข่ายและการเชื่อมต่อกับเครือข่ายภายนอกจึงเป็นองค์ประกอบหลักที่สำคัญขององค์กรที่จะต้องวิเคราะห์ความเหมาะสมเพื่อให้เกิดการสูญเสียค่าใช้จ่ายให้สอดคล้องกับประสิทธิภาพที่จะได้มาให้มากที่สุด ซึ่งปัจจัยที่สำคัญของการวิเคราะห์ประสิทธิภาพการสื่อสารข้อมูลที่ได้รับความนิยมได้แก่

ปริมาณการสื่อสาร(Traffic Quantity) หมายถึง จำนวนข้อมูลที่มีการแลกเปลี่ยนกันบนเครือข่ายคอมพิวเตอร์ ซึ่งจะมีผลกับหน่วยจัดเก็บข้อมูลบนอุปกรณ์เครือข่ายโดยตรง โดยรวมแล้วจำนวนและขนาดของข้อมูลจะมีความสัมพันธ์กัน สถานภาพเครือข่ายที่พบข้อมูลกระจายเป็นชิ้นเล็ก ๆ อาจสะท้อนให้เห็นถึงการสูญเสียเวลาในการรวมกลุ่มข้อมูลที่ปลายทางได้ หรือข้อมูลที่มีความยาวมากก็จะสะท้อนถึงอัตราการใช้พื้นที่จัดเก็บข้อมูลบนอุปกรณ์เครือข่ายด้วยในกรณีที่กำหนดพฤติกรรมการรับและส่งข้อมูลเป็นแบบ "Store and Forward"

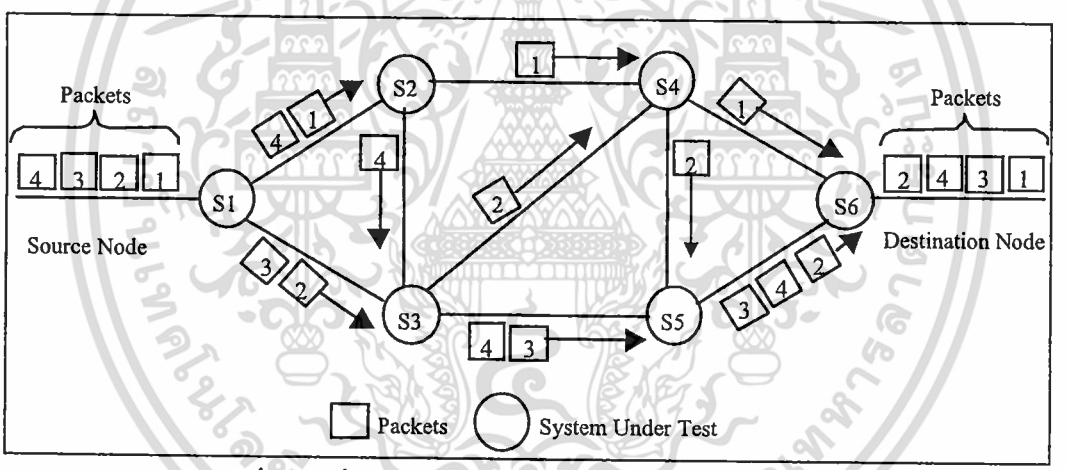
เวลาที่ใช้(Time Used) หมายถึง ระยะเวลาที่ใช้ไปในการแลกเปลี่ยนข่าวสาร กล่าวได้ว่าการสื่อสารข้อมูลใด ๆ ย่อมต้องการความรวดเร็วและถูกต้องของข้อมูลให้มากที่สุด การตอบรับการสื่อสารข้อมูลที่ใช้เวลานาน ๆ จะสะท้อนให้เห็นถึงความบกพร่องในการจัดนโยบายการสื่อสาร ข้อมูลโดย

ภาพรวมได้ หรือถ้าพิจารณาข้อกำหนดของการใช้ TCP(Transmission Control Protocol) เวลาที่นานเกินไปอาจทำให้ผู้ส่งต้องส่งข้อมูลใหม่(Retransmission)และซ้ำซ้อนไปยังปลายทางเป็นจำนวนมาก และอาจทำให้เกิดความหนาแน่นและสูญเสียข้อมูลต่อไปได้

ปัจจัยที่กล่าวในข้างต้นเป็นผลนำมาซึ่งการวิจัย และออกแบบเครื่องมือตรวจวัดพารามิเตอร์ต่าง ๆ ที่จำเป็นต่อการนำไปใช้พยากรณ์ความสามารถของระบบ และสามารถนำไปใช้ขจัดปัญหา รวมถึงการออกแบบได้อย่างเหมาะสม

2.2 เทคนิคการสื่อสารข้อมูล(Communication Techniques)

เทคนิคการสื่อสารข้อมูลบนเครือข่ายอินเทอร์เน็ตใช้วิธีการที่เรียกว่า Packet Switching ซึ่งเป็นวิธีการที่ยอมให้คอมพิวเตอร์ต่าง ๆ สามารถใช้เส้นทางหรือใช้บางส่วนของเส้นทางได้ในเวลาเดียวกัน ดังแสดงได้ในภาพที่ 2



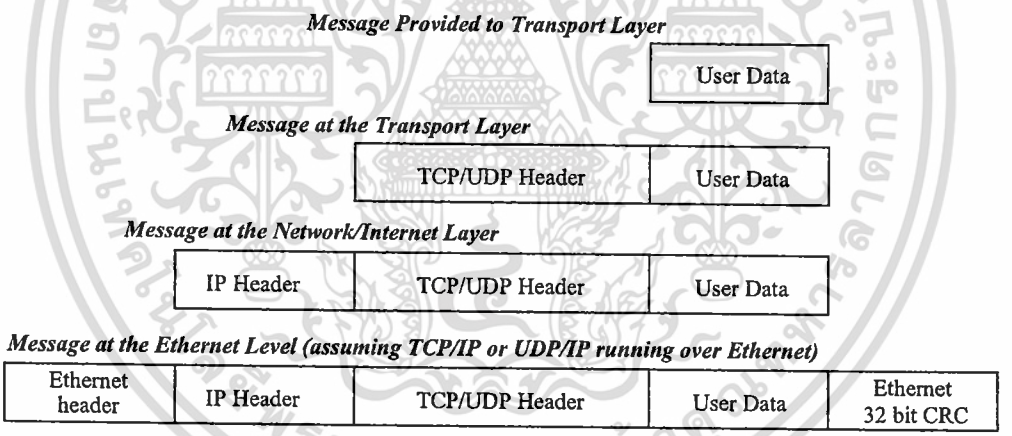
ภาพที่ 2 การสื่อสารข้อมูลแบบแพ็กเก็ตสวิชชิ่ง(Packet Switching)

แพ็กเก็ตจากโนดต้นทาง(Source Node) จะส่งผ่านไปบนสายสัญญาณเดียวกันเข้าสู่อุปกรณ์สวิชชิ่ง(Switching)หรือเราท์เตอร์(Router)ไปโนดปลายทาง(Destination Node) ซึ่งขอใช้นิยามโดยเรียกหน่วยบริการนี้ว่า "ตัวระบบ(System Under Test)" แทนความหมายที่รวมถึงอุปกรณ์การสื่อสารข้อมูลทุกชนิดบนเครือข่ายอินเทอร์เน็ต แพ็กเก็ตหนึ่ง ๆ ที่ไปถึงตัวระบบต้องถูกจัดเก็บให้สมบูรณ์ก่อนการตัดสินใจเพื่อส่งต่อให้กับตัวระบบที่อยู่ถัดไปและอาจไม่ตรงตามลำดับเมื่อแพ็กเก็ตทั้งหมดไปถึงโนดผู้รับปลายทาง เนื่องจากตัวระบบมีนโยบายการกำหนดเส้นทางที่แตกต่างกัน เช่น มาก่อนได้รับบริการก่อน(First Come First Serve) หรือ มาหลังได้รับบริการก่อน(Last Come First Serve) ดังนั้นตัว

ระบบจึงเป็นตัวแปรสำคัญที่ทำให้เกิดความความล่าช้าในการสื่อสารขึ้น โดยแปรผันตามความยาวของแพ็กเก็ตและคุณสมบัติในการเลือกเส้นทางของตัวระบบ ในความเป็นจริงความยาวของแพ็กเก็ตโดยมากมักมีค่าไม่เท่ากันเนื่องจากใช้กลไกควบคุมการสื่อสารข้อมูลด้วย TCP/IP(Transmition Control Protocol/Internet Protocol) นอกจากนี้การสื่อสารข้อมูลอาจขาดความสมบูรณ์ขึ้นได้เมื่อเกิดความสูญเสียขึ้นที่ตัวระบบเพราะพื้นที่จัดเก็บข้อมูลบนตัวระบบไม่เพียงพอต่ออัตราการมาของแพ็กเก็ต

2.3 คุณลักษณะของแพ็กเก็ต(Packets Characteristic)

ระบบการสื่อสารบนเครือข่ายคอมพิวเตอร์ได้มีการออกแบบมาตรฐาน เพื่อใช้อธิบายถึงการเชื่อมต่อและใช้เป็นข้อตกลงในการออกแบบโปรโตคอล(Protocol)เพื่อการสื่อสารข้อมูลเรียกว่า OSI (Open System Interconnection Model) โดยจะแบ่งหน้าที่การทำงานออกเป็น 7 ชั้น(Layer) แต่ละชั้นจะทำหน้าที่บรรจุข้อมูลที่เรียกว่า "แพ็กเก็ต(Packets)" เพื่อส่งต่อไปยังผู้รับปลายทาง รูปแบบการบรรจุข้อมูลจนกระทั่งเป็นแพ็กเก็ตในรูปแบบของ TCP/IP สามารถแสดงได้ในภาพที่ 3 ดังนี้

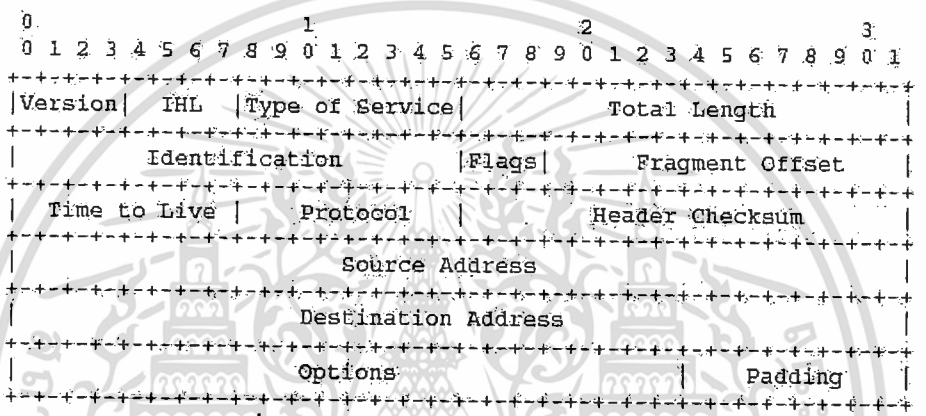


ภาพที่ 3 การห่อหุ้มแพ็กเก็ต(Packet Encapsulation)

เมื่อข้อมูล(User Data) ส่งจากชั้นบนสุดคือ Application Layer ลงไปยังชั้นที่ต่ำกว่า แต่ละชั้นก็จะทำการเพิ่มข้อมูลลงไปบนส่วนด้านหน้าของข้อมูลจากชั้นก่อนหน้า เช่น TCP/UDP Header, IP Header และ Ethernet Header จะถูกเติมลงในชั้น Transport, Network และ Data Link ตามลำดับ โดยมีรูปแบบของข้อมูลส่วนหัว(Header Format)ที่พิจารณาตาม IPV4 Header ได้ดังนี้

2.3.1 ส่วนหัวของไอพี(Internet Protocol Header)

เป็นข้อมูลที่ถูกเพิ่มในส่วนของ Network Layer หรือเรียกส่วนนี้ว่า IP Datagram มีหน้าที่ในการเชื่อมต่อและรับประกันการส่งข้อมูลไปยังผู้รับปลายทางอย่างถูกต้อง นอกจากนี้ยังมีหน้าที่ในการแตกชิ้นส่วน(Fragmentation) และร่วมกลุ่มชิ้นส่วนของข้อมูล(Reassembly)ในกรณีที่ข้อมูลมีขนาดเกินกว่าค่าสูงสุดในการส่งข้อมูลในแต่ละหน่วย(Maximum Transmission Unit Size) โดยมีรูปแบบที่ประกอบด้วยข้อมูลดังแสดงได้ในภาพที่ 4



ภาพที่ 4 ส่วนประกอบของข้อมูลใน IP Datagram

โดยที่

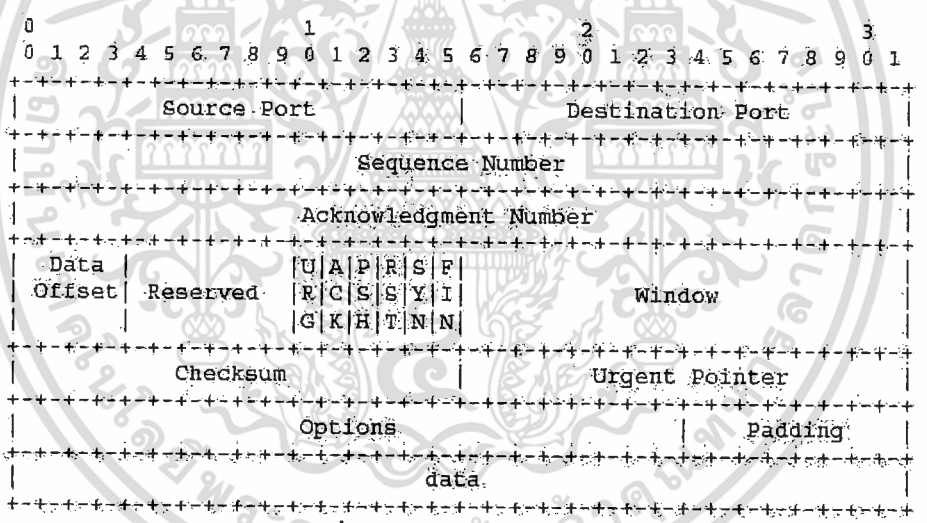
- Version ขนาด 4 bits ใช้แสดงรูปแบบของ Internet Header
- IHL ขนาด 4 bits ใช้บอกความยาวของ Internet Header ในชนิด 32 bit words
- Type of Service ขนาด 8 bits ใช้บ่งบอกคุณสมบัติของการให้บริการ
- Total length ขนาด 16 bits ใช้บอกความยาวของ datagram ที่รวมถึง Internet Header
- Identification ขนาด 16 bits ใช้บอกลำดับของข้อมูลที่ส่งไปยังปลายทาง
- Flags ขนาด 3 bits ใช้บอกสถานะของ datagram เมื่อมีการแยกชิ้นส่วน
- Time to LIVE ขนาด 8 bits ใช้บอกเวลาสูงสุดที่ datagram ยังคงอยู่ได้ในระบบเครือข่ายอินเทอร์เน็ต
- Protocol ขนาด 8 bits ใช้บอกประเภทของโปรโตคอลใน Transport Layer
- Header Checksum ขนาด 16 bits ใช้ตรวจสอบความถูกต้องของ Internet Header
- Source Address ขนาด 32 bits ใช้บอกที่อยู่ต้นทางของผู้ส่ง
- Destination Address ขนาด 32 bits ใช้บอกที่อยู่ปลายทางของผู้ส่ง
- Options ให้ใช้ประโยชน์โดยปรับเปลี่ยนได้ตามความต้องการ

2.3.2 ส่วนหัวใน Transport Layer

เป็นข้อมูลที่เพิ่มเติมต่อจาก Internet Header เพื่อใช้ทำหน้าที่รับประกันความน่าเชื่อถือของการรับและส่งข้อมูล ซึ่งในชั้นนี้สามารถแบ่งรูปแบบของข้อมูลออกเป็น 2 โปรโตคอลหลัก คือ TCP(Transmission Control Protocol) และ UDP(User Datagram Protocol) นอกจากนี้ยังมีอีกหนึ่งโปรโตคอลคือ ICMP(Internet Control Message Protocol) แต่โดยทั่วไปมักนิยมใช้นำเสนอให้อยู่ในชั้น Network โดยมีรูปแบบที่ประกอบด้วยข้อมูลต่าง ๆ ตามลำดับดังนี้

2.3.2.1 รูปแบบของ TCP Header

การสื่อสารข้อมูลด้วย TCP ผู้ส่งจะได้รับการประกันการสื่อสารข้อมูลคือ ข้อมูลครบถ้วนและถูกต้อง นอกจากนี้ยังมีส่วนในการควบคุมความคับคั่งและลดความเสี่ยงของการสูญเสียข้อมูลที่ส่งไปยังปลายทาง โดยมีรูปแบบข้อมูลที่สามารถแสดงได้ในภาพที่ 5



ภาพที่ 5 รูปแบบของ TCP Header

โดยที่

- Source Port ขนาด 16 bits บอกหมายพอร์ตที่ใช้ส่งข้อมูลจากต้นทาง
- Destination Port ขนาด 16 bits บอกหมายเลขพอร์ตที่ต้องใช้รับปลายทาง
- Sequence Number ขนาด 32 bits ใช้บอกหมายเลขลำดับของการส่งข้อมูลในแต่ละส่วน (Segment)
- Acknowledgment Number ขนาด 32 bits ถ้าสัญญาณ ACK ถูกกำหนดขึ้น Sequence Number ถัดไปของผู้ส่งจะใช้ในการคาดคะเนการส่งกลับมาของผู้รับ

- Data Offset ขนาด 4 bits ใช้บอกจุดเริ่มต้นของข้อมูลเป็น 32 bits ใน TCP Header
- Reserved ขนาด 6 bits สงวนไว้ใช้ในอนาคต
- Control bits ขาด 6 bits

  URG ใช้บอกสถานะความสำคัญของข้อมูล

  ACK ใช้บอกการตอบรับจากผู้ส่ง

  PSH ใช้บอกสถานะการส่งข้อมูล

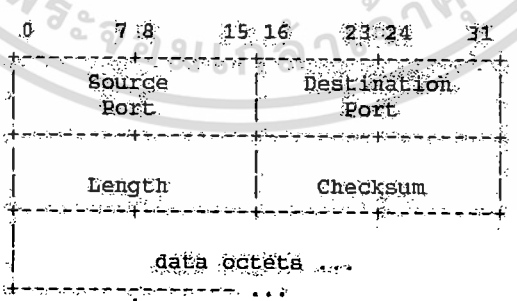
  RST ใช้บอกสถานะการยกเลิกการติดต่อ

  SYN ใช้บอกสถานะการเชื่อมต่อ

  FIN ใช้บอกสถานะปิดการเชื่อมต่อ
- Window ขนาด 16 bits ใช้ควบคุมการรับชิ้นส่วนข้อมูลจากผู้ส่ง
- Checksum ขนาด 16 bits ใช้ตรวจสอบความถูกต้องของ TCP Header
- Urgent Pointer ขนาด 16 bits ใช้บอก Sequence Number ที่มีลำดับความสำคัญในการรับส่งข้อมูล
- Options ให้ใช้ประโยชน์โดยปรับเปลี่ยนได้ตามความต้องการ
- Data ใช้บรรจุข้อมูลที่อยู่เหนือกว่า Transport Layer

2.3.2.2 รูปแบบของ UDP Header

การสื่อสารข้อมูลด้วย UDP นิยมใช้กับข้อมูลที่ต้องการส่งไปให้ถึงผู้รับด้วยความรวดเร็วโดยไม่คำนึงถึงความสมบูรณ์และถูกต้องเมื่อข้อมูลไปถึงฝั่งผู้รับ โดยมีรูปแบบข้อมูลที่สามารถแสดงได้ในภาพที่ 6



ภาพที่ 6 รูปแบบของ UDP Header

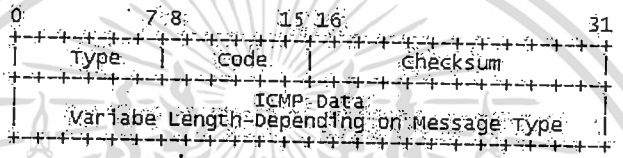
โดยที่

- Source Port ขนาด 16 bits บอกหมายพอร์ตที่ใช้ส่งข้อมูลจากต้นทาง

- Destination Port ขนาด 16 bits บอกหมายเลขพอร์ตที่ต้องใช้รับปลายทาง
- Length ขนาด 16 bits ใช้บอกความยาวของข้อมูลรวมถึง UDP Header
- Checksum ขนาด 16 bits ใช้ตรวจสอบความถูกต้องของ UDP Header

2.3.2.3 รูปแบบของ ICMP Header

การสื่อสารข้อมูลด้วย ICMP นิยมใช้เพื่อรายงานความผิดพลาดหรือสถานภาพของการสื่อสารบนเครือข่าย โดยมีรูปแบบข้อมูลที่สามารถแสดงได้ในภาพที่ 7



ภาพที่ 7 รูปแบบของ ICMP Header

โดยที่

- Type ขนาด 8 bits ใช้บอกรูปแบบของการรายงานสถานภาพเครือข่าย
- code ขนาด 8 bits เป็นหมายเลขบอกสถานภาพเครือข่าย
- Checksum ขนาด 16 bits ใช้ตรวจสอบความถูกต้องของ ICMP Header
- ICMP Data เพิ่มเติมได้เพื่อให้เกิดประโยชน์กับการรายงาน เช่นในกรณีของการตรวจสอบการไปถึงของข้อมูลอาจใช้สัญญาณสะท้อน(Echo) ที่ประกอบด้วยสัญญาณการร้องขอ(Request) และตอบรับ(Reply) ซึ่งการส่งข้อมูลอาจมีได้หลายครั้งจากผู้ส่งเดียวกัน ดังนั้นจึงต้องมี ICMP Identifier เพื่อระบุคู่สัญญาณให้ถูกต้องในแต่ละครั้ง นอกจากนี้ในแต่ละครั้งของการส่งข้อมูลสัญญาณอาจมีการส่งสัญญาณร้องขอได้หลายครั้ง ตัวอย่างเช่น การใช้ ping command บน Operating System ที่สามารถระบุจำนวนของ สัญญาณร้องขอได้ ดังนั้นการร้องขอแต่ละครั้งจึงต้องมี ICMP Sequence Number เพื่อบอกลำดับของการตรวจสอบภายใต้ ICMP Identifier หนึ่ง ๆ ด้วย

การสื่อสารข้อมูลบนเครือข่ายอินเทอร์เน็ตในปัจจุบันได้ใช้เทคนิคของการส่งแพ็กเก็ตไปบนเครือข่าย โดยที่แอพพลิเคชั่นแต่ละชนิดจะเลือกใช้ชนิดของแพ็กเก็ตตามความเหมาะสม และแพ็กเก็ตในแต่ละชนิดก็จะมีขบวนการรับส่งข้อมูลที่แตกต่างกัน

ดังนั้นรายงานงานวิจัยนี้จึงมีแนวคิดในการสร้างเครื่องมือวัดและวิเคราะห์คุณลักษณะของข้อมูลที่ผ่านตัวระบบเพื่อใช้สะท้อนถึงปัญหาต่าง ๆ ที่เกิดขึ้นภายในระบบเครือข่าย และนำไปสู่ขบวนการขจัดปัญหาหรืออุปสรรคของการสื่อสารข้อมูลได้

2.4 เทคนิคการตรวจจับข้อมูล(Capturing Techniques)

ปัจจุบันการตรวจจับข้อมูลได้กลายมาเป็นความสำคัญที่ต้องแสวงหาเครื่องมือ ที่จะนำมาวิเคราะห์ประสิทธิภาพการทำงานของเครือข่าย สิ่งที่สำคัญของการตรวจจับข้อมูลก็คือต้องไม่ทำให้เกิดความสูญเสียหรือมีผลกระทบต่อการสื่อสารข้อมูล โดยทั่วไปสามารถแบ่งวิธีการตรวจจับได้ 2 วิธี คือ การใช้เครื่องมือตรวจจับเป็น Hardware โดยตรง หรือการสร้าง Software ขึ้นโดยอาศัยคอมพิวเตอร์ส่วนบุคคลทั่วไป(PC Workstation) ให้ทำหน้าที่ตรวจจับแทน ซึ่งจะมีประสิทธิภาพที่ต่ำกว่าวิธีการแรก แต่ความสามารถในการปรับปรุงให้ตรงกับความต้องการของผู้ใช้จะทำได้ดีกว่า และสามารถพัฒนาขึ้นใช้เองได้

2.4.1 BPF บนระบบปฏิบัติการยูนิกซ์

จากที่นักวิจัย S.McCanne และ V.Jacobson ได้ทำการพัฒนา BPF(Bekeley Packet Filter) ขึ้นในมหาวิทยาลัยแคลิฟอเนีย(University of California) ทำให้นำไปสู่การสร้าง ไดร์เวอร์(Driver)ตรวจจับข้อมูล(Capture Driver) ที่มีประสิทธิภาพขึ้นบนระบบปฏิบัติการยูนิกซ์(Unix) เรียกว่า Libpcap Library และถูกนำไปสร้างคำสั่ง tcpdump บนระบบปฏิบัติการยูนิกซ์ที่แพร่หลายและใช้งานกันทั่วไป BPF Driver เป็นเครื่องมือที่ถูกเรียกใช้งานโดยโปรแกรมประยุกต์บนระบบปฏิบัติการยูนิกซ์เพื่อให้อ่าน แพ็กเก็ตผ่านอุปกรณ์เชื่อมต่อกับเครือข่าย(Network Adapter) และต่างจาก Driver ทั่วไปที่ไม่ได้ถูก ควบคุมโดยตรงจากอุปกรณ์เชื่อมต่อโดยมีองค์ประกอบที่สำคัญ 2 ส่วนคือ ส่วนเชื่อมต่อกับเครือข่าย (Network Tap) และส่วนกรองแพ็กเก็ต(Packet Filter)

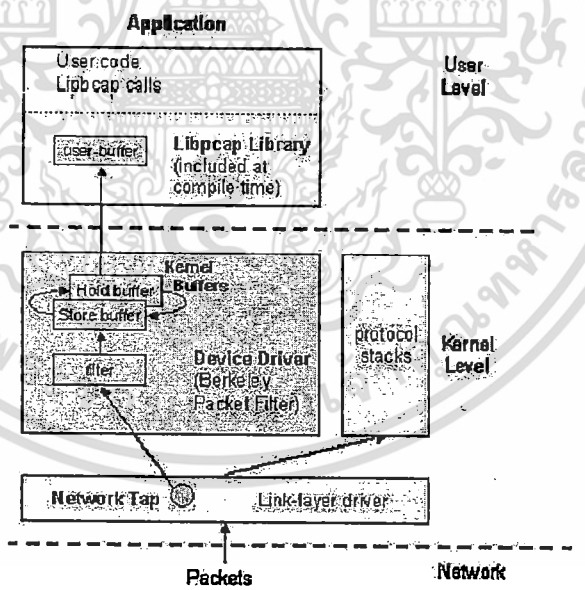
2.4.1.1 ส่วนเชื่อมต่อกับเครือข่าย(Network Tap) เป็นฟังก์ชั่นที่เขียนขึ้นเมื่อต้องการร้องขอให้ตรวจจับแพ็กเก็ตจากเครือข่ายและหลีกเลี่ยงจากการใช้งานกับ BPF โดยตรง ส่วนนี้จะถูกเรียกใช้โดยอุปกรณ์เชื่อมต่อ(Network Adapter) เมื่อตรวจพบการมาของแพ็กเก็ตและทำการคัดลอกแพ็กเก็ตส่งให้กับหน่วยรับฟังข้อมูลของโปรแกรมประยุกต์(Listening Application) ถ้าแพ็กเก็ตผ่านการตรวจสอบโดยเงื่อนไขของหน่วยกรองข้อมูล

2.4.1.2 ส่วนกรองแพ็กเก็ต(Packet Filter) เป็นหน่วยที่ทำหน้าที่แยกชนิดแพ็กเก็ตโปรแกรมประยุกต์ส่วนใหญ่ใช้ BPF ในการตรวจสอบและละทิ้ง(Reject) แพ็กเก็ตที่ไม่ได้รับความสนใจออกไปเพื่อเพิ่มประสิทธิภาพและความรวดเร็วในการวิเคราะห์ข้อมูล ส่วนแยกชนิดข้อมูลจะมี

ลักษณะเป็นฟังก์ชั่นที่ทำงานแบบตรรก(Boolean) นอกจากนี้ยังมีประโยชน์กับหน่วยสำรองข้อมูล (Buffer)ด้วย เนื่องจากในบางครั้งบางโปรแกรมประยุกต์อาจให้ความสนใจที่ส่วนหัวของแพ็กเก็ตโดยไม่สนใจในส่วนอื่นของข้อมูล ดังนั้นหน่วยแยกชนิดข้อมูลสามารถตัดเอาเฉพาะส่วนหัวคัดลอกไปจัดเก็บบนหน่วยสำรองข้อมูล เพื่อรอการส่งต่อให้กับหน่วยรับฟังของโปรแกรมประยุกต์ต่อไปได้ ซึ่งจะทำให้การบริหารพื้นที่จัดเก็บข้อมูลมีประสิทธิภาพ และสามารถลดความสูญเสียหรือละทิ้งบาง แพ็กเก็ตที่ไม่สามารถให้บริการการคัดลอกทันต่อการมาของแพ็กเก็ตลงได้

2.4.2 ภาพรวมการทำงานของ BPF

BPF จะมีส่วนสำรองข้อมูล(Buffer) 2 ส่วนที่มีความสำคัญต่อกระบวนการตรวจจับข้อมูลของทุกหน่วย การตรวจจับข้อมูลจะถูกเรียกโดยโปรแกรมประยุกต์ผ่าน BPF ผ่านไปยัง IOCTL หน่วยสำรองข้อมูลจะถูกเตรียมขึ้นโดย BPF ที่มีขนาด 4 KB ซึ่งหน่วยสำรองข้อมูลชุดแรกถูกเรียกว่า Store Buffer ทำหน้าที่จัดเก็บข้อมูลที่ตรวจพบจากอุปกรณ์เชื่อมต่อเครือข่าย(Network Adapter) หน่วยที่สองเรียกว่า Hold Buffer ทำหน้าที่คัดลอกแพ็กเก็ตเพื่อส่งต่อให้กับโปรแกรมประยุกต์ ปกติแล้ว Hold Buffer จะคอยการสลับที่หรือคัดลอกข้อมูลจาก Store Buffer ในกรณีที่ Store Buffer บรรจุข้อมูลเต็มแล้ว ซึ่งกระบวนการทั้งหมดนี้จะไม่อยู่รวมกับส่วนของ Adapter Device Driver ดังแสดงได้ในภาพที่ 8



ภาพที่ 8 โครงสร้าง BPF

เมื่อแพ็กเก็ตมาถึงเครือข่ายในระดับ Link Layer แพ็กเก็ตจะถูกส่งต่อไปยังแต่ละชั้นของโปรโตคอล(Protocol Stack) BPF จะทำการเรียกใช้ฟังก์ชั่นในส่วนของ Network Tap เพื่อส่งต่อไปยังหน่วยกรองข้อมูลที่ได้กำหนดเงื่อนไขและจำนวน(Packet Bytes)ที่ต้องการจัดเก็บข้อมูลจากระดับผู้ใช้

ไว้แล้ว(สามารถกำหนดเงื่อนไขจากการเรียกฟังก์ชั่นใน Libpcap Library) โดยในขณะที่ตรวจสอบนั้น ข้อมูลจะยังไม่ถูกคัดลอกไปในส่วนของ Kernel Level เนื่องจากจะทำให้พื้นที่หน่วยความจำต้องเสียไป ในกรณีที่แพ็กเก็ตนั้นถูกปฏิเสธจากเงื่อนไขในการยอมรับในระดับของผู้ใช้ ในกรณีที่ข้อมูลผ่านเงื่อนไขในการกรองแล้วก็จะถูกคัดลอกเพื่อจัดเก็บไว้ในส่วนของStore Buffer และสลับไปไว้ที่ Hold Buffer เมื่อพื้นที่จัดเก็บเต็ม ในขณะเดียวกันก็จะมีอีกกระบวนการหนึ่งที่คอยอ่านข้อมูลจาก Hold Buffer เพื่อส่งต่อให้กับระดับของผู้ใช้ การรับข้อมูลจาก BPF ในระดับผู้ใช้สามารถอ่านข้อมูลได้มากกว่า 1 แพ็กเก็ตในแต่ละครั้ง แต่ละครั้ง BPF จะทำการห่อหุ้มข้อมูลเพิ่มเติมลงไป(Encapsulate) ในส่วนหัวของข้อมูลได้แก่ เวลาที่มาถึง(Time Stamp), ความยาว และลำดับของข้อมูล(Offset of Data Alignment) ตามลำดับ

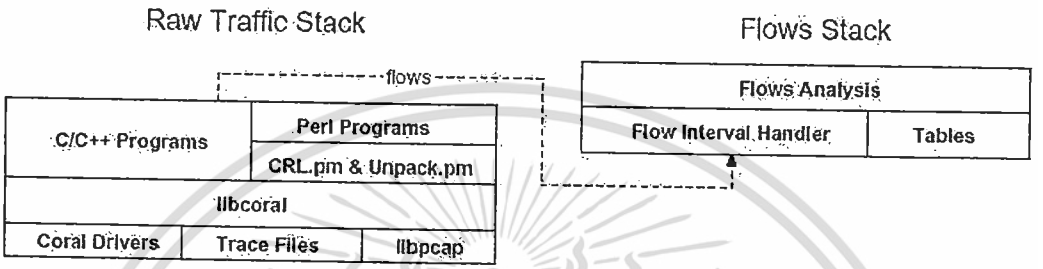
ปัจจุบัน Libpcap Library ได้ถูกนำไปใช้โดยแพร่หลายในการสร้างเครื่องมือวัด ซึ่งโดยทั่วไปวิธีการวัดพฤติกรรมการสื่อสารข้อมูลแบ่งออกได้ 2 วิธี คือ การวัดที่ต้องใช้การรับส่งข้อมูลระหว่างคู่สื่อสาร(Active Traffic Measurement)และการวัดที่ไม่ใช้การรับส่งข้อมูลระหว่างคู่สื่อสาร(Passive Traffic Measurement) เครื่องมือวัดที่มีประสิทธิภาพจะต้องไม่ก่อให้เกิดผลกระทบหรือเพิ่มภาระงานให้กับระบบในขณะทำการวัด นอกจากนี้ผลลัพธ์ที่ได้ต้องสะท้อนถึงความเป็นจริงของการสื่อสารในสภาวะที่เป็นปัจจุบันและต่อเนื่อง ดังนั้นวิธีการวัดแบบที่ 2 จึงเป็นวิธีที่นิยมนำไปใช้ในการสร้างเครื่องมือวัดกันมาก ตัวอย่างเช่น การวัดข้อมูลโฟลว(Data Flow) ด้วย NetTraMet, NetFlow, Cflowd, FlowScan และ CoralReef ซึ่งในเครื่องมือ 4 ชนิดแรกได้ถูกสร้างขึ้นจากการใช้ Libpcap Library ส่วน CoralReef ได้พัฒนาจาก Corallib Library ซึ่งพัฒนาต่อจาก Libpcap Library ในภายหลัง

2.5 CoralReef ซอฟท์แวร์ตรวจวัดการสื่อสารบนเครือข่ายอินเทอร์เน็ต

CoralReef เป็นนวัตกรรมทางซอฟท์แวร์ที่ประกอบด้วย 2 โครงสร้างหลัก คือ

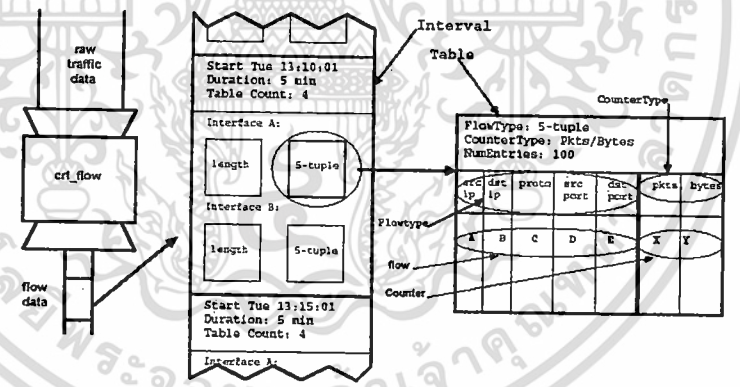
Raw Traffic เป็นส่วนที่ใช้ติดต่อกับ Protocol Stack (Packets หรือ Cells) โดยมีหน้าที่ในการอ่านข้อมูลจากเครือข่าย โครงสร้างประกอบด้วย Libcoral C library ให้นักเขียนโปรแกรมสามารถเรียกใช้และเขียนเป็นแอพพลิเคชั่นได้ด้วยภาษา C, C++ หรือ Perl ภายใน Libcoral จะประกอบด้วย API(Application Program Interface) สำหรับตรวจจับข้อมูลได้จากทั้ง ATM และ POS cards บางส่วนของ API สามารถอ่านแพ็กเก็ตจากแฟ้มข้อมูลได้หลากหลายรูปแบบ เช่น Coral format ,NLANR format, PCAP, DAG ATM และ POS format นอกจากนี้ยังมีส่วนช่วยสำหรับตรวจสอบโครงสร้างเครือข่าย เช่น โมดูล(Module)สำหรับตรวจสอบเพื่อกำหนดเส้นทางของ IP Address ด้วยการใช้ BGP(Border Gateway Protocol) ใน AS(autonomous system) หนึ่ง ๆ

Flows Stack เป็นส่วนที่ทำหน้าที่จัดการรวบรวมข้อมูลให้เป็นหน่วยข้อมูลที่เรียกว่า "Flow Intervals" และจัดเก็บเป็นตารางแจกแจงความถี่ข้อมูล มีหน่วยวัดข้อมูลอยู่ในรูปของผลรวมความยาว มีหน่วยเป็นไบต์(Bytes), จำนวนแพ็กเก็ต(packets), ข้อมูลโฟลวในแต่ละสถานี(Host), ประเภทของ IP Protocol, และ Port ดังแสดงได้ในภาพที่ 9



ภาพที่ 9 โครงสร้างของ CoralReef Software

ส่วนจัดเก็บข้อมูล(Flows Stack) จะทำหน้ารวบรวมข้อมูลให้อยู่ในรูปที่ หน่วยวิเคราะห์(Flows Analysis) สามารถนำไปวิเคราะห์ต่อได้ โดยมีขั้นตอนจัดเก็บข้อมูลดังแสดงได้ในภาพที่ 10



ภาพที่ 10 การจัดเก็บข้อมูลโฟลวด้วย CoralReef

จากภาพที่ 10 ข้อมูลที่ตรวจจับได้(Raw Traffic Data)จาก Interface จะถูกนำไปรวบรวมด้วยหน่วย Flow Stack ให้อยู่ในรูปตารางข้อมูลโฟลว(Flow Type) ที่ประกอบด้วย Source IP, Destination IP, Protocol, Source Port และ Destination Port นอกจากนี้ขบวนการจัดเก็บยังมีหน่วยนับ(CounterType)ที่สร้างตารางประมวลผลเพิ่มเติมได้ เช่น รวบรวมจำนวนแพ็กเก็ต ความยาวของแพ็กเก็ต และ อัตราการมาถึงหน่วยให้บริการ ที่สามารถนำไปสู่การสร้างรายงานได้ ดังแสดงได้ในภาพที่ 11 และ 12

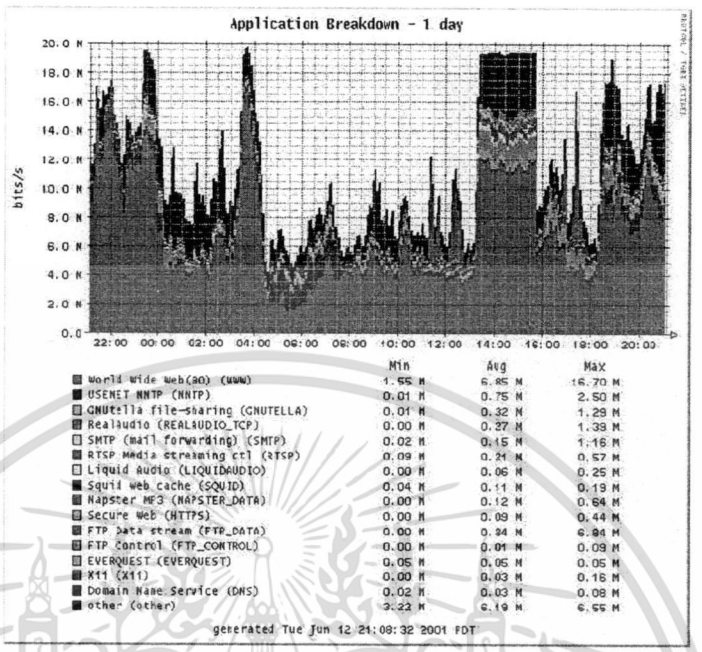
```
# begin Tuple Table ID: 0[131]
# expired flows
#src            dst          proto ok sport dport     pkts     bytes  flows
0.1.0.8          1.82.0.1        17  1    53    53        2       497      1
0.1.0.14         0.44.0.1         6  1    80  2223        4       646      1
0.3.0.148        1.95.0.1         6  1  1214 62772      125    187008      1
0.1.1.93         0.71.0.6         6  1 49200    80        3       565      1
0.1.1.93         0.71.0.6         6  1 49199    80        5       647      1
0.1.1.93         0.71.0.6         6  1 49198    80        5       647      1
0.1.1.93         0.71.0.6         6  1 49196    80        6       708      1
0.1.2.59         11.88.0.1        6  1 51643    80        6       817      1
...
# end of text table
```

ภาพที่ 11 ตัวอย่างรายงานข้อมูลโฟลว

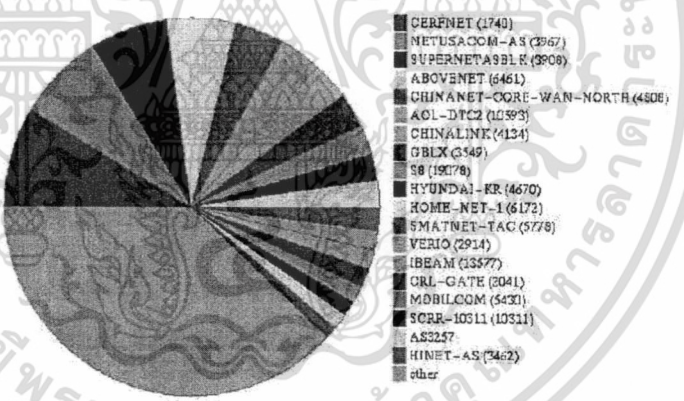
```
# time 1001975450.054545 (10.000000), packets lost: 0
# if[subif] v4pkts    v4bytes v6pkts v6bytes non_ip v4pkts/s v4bits/s v6pkts/s v6bits/s
0[135]          1         40      0       0      0     0.10    32.00     0.00     0.00
0[110]       2269    2536140      0       0      0    22.69     2.03M    0.00     0.00
0[169]       9397    3761410      0       0      0    93.97     3.01M    0.00     0.00
0[170]      40097   20640233      0       0      0   400.97    16.51M    0.00     0.00
0[130]       5659    1921566      0       0      0    56.59     1.54M    0.00     0.00
0[131]     118429   70553909      0       0      0     1.18k   56.44M    0.00     0.00
0[108]       1774      92307      0       0      0    17.74    73.85k    0.00     0.00
0   TOTAL  177626   99505605      0       0      0     1.78k   79.60M    0.00     0.00
```

ภาพที่ 12 ตัวอย่างรายงานจำนวนและอัตราการมาถึงของแพ็กเก็ต

จากการออกแบบ CoralReef ได้นำไปสู่การสร้างเครื่องมือวัดที่หลากหลายขึ้น โดยนำส่วนโครงสร้างในการจัดเก็บข้อมูลไปใช้สร้างกราฟเพื่อชี้ถึงคุณลักษณะการสื่อสารข้อมูลได้หลากหลายชนิด ยกตัวอย่างเช่น งานวิจัยของ David Moore ในเรื่อง "The CoralReef software suite as a tool for system and network administrators" ได้นำไปจำแนกชนิดของข้อมูลโฟลวดังแสดงได้ใน ภาพที่ 13 และ 14



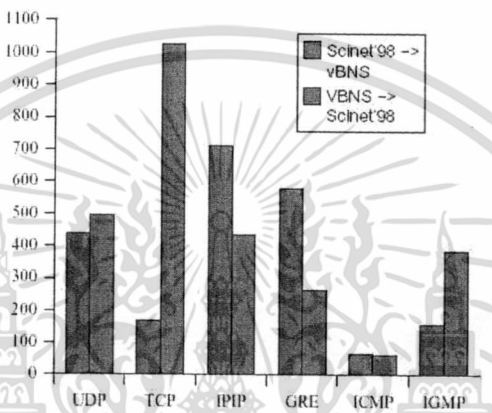
ภาพที่ 13 ตัวอย่างกราฟที่จำแนกคุณลักษณะข้อมูลในแต่ละชนิด



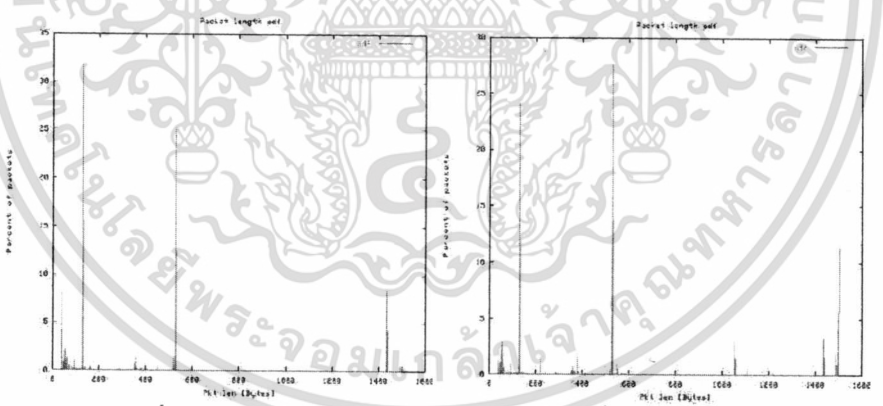
ภาพที่ 14 ตัวอย่างกราฟที่จำแนกอัตราส่วนข้อมูลโฟลวในแต่ละชนิด

นอกจากข้อมูลโฟลวแล้วยังมีข้อมูลอื่นอีกหลายชนิดที่จะช่วยสนับสนุนการวิเคราะห์และสะท้อนปัญหาที่เกิดขึ้นกับเครือข่ายการสื่อสารได้ ซึ่งในงานวิจัยของ Brynjar Age Viken เรื่อง "Passive monitor of Internet Traffic at Supercomputer'98" ได้นำ Coral Library ไปใช้ตรวจจับคุณลักษณะ ความยาว และเวลาระหว่างการมาของแพ็กเก็ต โดยใช้ร่วมกับอุปกรณ์ Optical Splitter คือ OC3 ทำงานกับหน่วยตรวจจับข้อมูลที่เป็นระบบปฏิบัติการ FreeBSD บนเครื่องคอมพิวเตอร์ที่มี CPU เป็น Intel Pentium II ความเร็ว 400 MHz มีหน่วยความจำสำรองขนาด 128 MB ตรวจจับข้อมูล

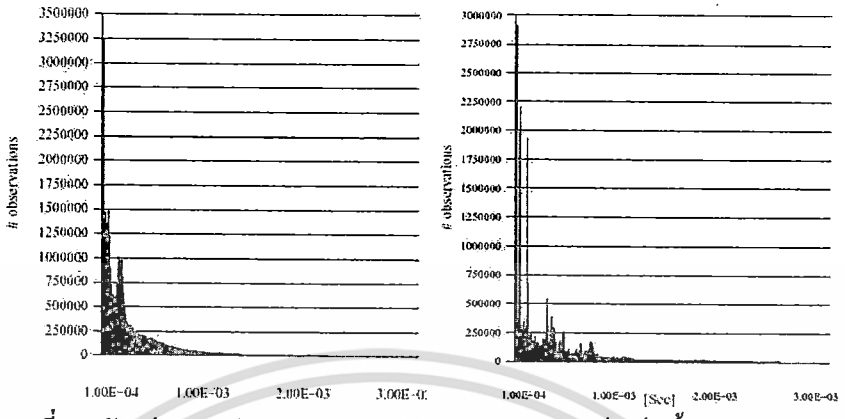
จาก 2 Interface ลงบน SCSI Hard Disk โดยพัฒนาโปรแกรมด้วยภาษา C++ ให้ทดลองตรวจจับข้อมูลจริงที่มีการสื่อสารข้อมูลระหว่าง SCiNet'98 และ vBNS ในรัฐฟอริดา สหรัฐอเมริกา ผลการตรวจวัดเครื่องมือไม่มีผลกระทบต่อตัวระบบที่จะวัดและสามารถนำข้อมูลที่ได้บน Hard Disk ไปใช้สร้างรายงานการสื่อสารข้อมูลแต่ละชนิด ที่สามารถจำแนกออกได้ 2 ทิศทางคือ การสื่อสารขาไปจาก SCiNet'98 ไปยัง vBNS และ ขากลับจาก vBNS ไปยัง SCiNet'98 ดังแสดงได้ในภาพที่ 15, 16 และ 17



ภาพที่ 15 ตัวอย่างกราฟวิเคราะห์ปริมาณของแพ็กเก็ตแต่ละชนิดทั้งขาไปและกลับ



ภาพที่ 16 ตัวอย่างกราฟแจกแจงขนาดแพ็กเก็ตทั้งขาไปและกลับ



ภาพที่ 17 ตัวอย่างกราฟแจกแจงเวลาระหว่างการมาของแพ็กเก็ตทั้งขาไปและกลับ

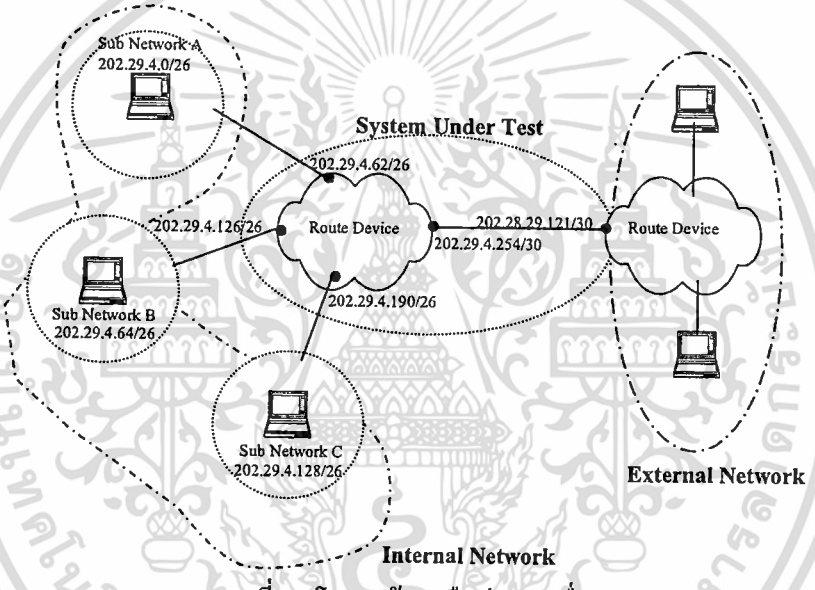
ในการตรวจวัดการสื่อสารข้อมูลยังมีอีกหลายปัจจัยที่สามารถนำไปใช้วิเคราะห์ความบกพร่องของปัญหาการสื่อสารข้อมูลได้ เช่น เวลาการสื่อสารไปกลับ ความล่าช้า และความสูญเสีย ซึ่งจากการศึกษาในส่วนของการตรวจวัด ความยาว เวลาระหว่างการมา และข้อมูลโฟรวสามารถตรวจวัดได้ด้วย NetTraMet หรือ CoralReef ที่พัฒนาจาก Corallib Library หรือเวลาการสื่อสารไปกลับจะมีกับเฉพาะเครื่องมือที่ต้องใช้การรับส่งข้อมูลระหว่างคู่สื่อสาร เช่น ping ที่ใช้การรับส่งข้อมูลด้วย ICMP โปรโตคอล แต่สิ่งสำคัญอีกสิ่งหนึ่งก็คือ เครื่องมือที่ได้รับการออกแบบที่ดีต้องสามารถนำไปสู่การพัฒนาต่อเนื่องและสามารถนำข้อมูลที่จัดเก็บไปใช้ประโยชน์ได้อย่างหลากหลาย

บทที่ 3
การออกแบบวิธีตรวจวัด

3.1 หลักการตรวจวัดข้อมูล

3.1.1 รูปแบบการวัดและหน่วยวัดการสื่อสารข้อมูล

โดยทั่วไปการเชื่อมต่อการสื่อสารข้อมูลบนเครือข่ายอินเทอร์เน็ต สามารถพิจารณาได้จากภาพที่ 18 ดังนี้



ภาพที่ 18 โครงสร้างเครือข่ายการสื่อสาร

จากภาพที่ 18 เป็นการเชื่อมต่อการสื่อสารข้อมูลระหว่างเครือข่ายภายใน(Internal Network) และเครือข่ายภายนอก(External Network) เมื่อมีการสื่อสารข้อมูลจากโนดที่อยู่ภายในเครือข่ายย่อย (Sub Network)ไปยังโนดอื่น ๆ ข้อมูลจะถูกส่งเป็นชิ้นเล็กเรียกว่า แพ็กเก็ต(Packet) ไปยังอุปกรณ์สืบค้นเส้นทางการสื่อสาร(Route Device)ก่อน เพื่อตรวจสอบเส้นทางจากหมายเลขไอพีที่ติดมากับ แพ็กเก็ต กับนโยบายสืบค้นเส้นทางที่ถูกกำหนดขึ้นโดยผู้ดูแลเครือข่าย ซึ่งแพ็กเก็ตอาจจะถูกส่งไปยังโนดปลายทางที่มีอยู่ทั้งภายในหรือภายนอกเครือข่าย

อุปกรณ์สืบค้นเส้นทางการสื่อสารจะมีลักษณะเช่นเดียวกับโนดสื่อสารอื่น ๆ คือ ต้องมีหมายเลขไอพี และอาจมีได้มากกว่า 1 หมายเลขเพื่อแทนค่าช่องทางการสื่อสารข้อมูล และต้องมีอย่างน้อย 1 หมายเลขที่ใช้สำหรับเชื่อมต่อกับเครือข่ายภายนอก และเช่นเดียวกันอุปกรณ์สืบค้นเส้นทางที่

อยู่ฝั่งเครือข่ายภายนอกก็จะต้องมีหมายเลขไอพีสำหรับเชื่อมต่อกับเครือข่ายภายในด้วย ดังนั้นด้วยลักษณะการจัดการเครือข่ายนี้เราสามารถจัดกลุ่มของสมาชิกโนดต่าง ๆ ได้ 3 กลุ่ม คือ

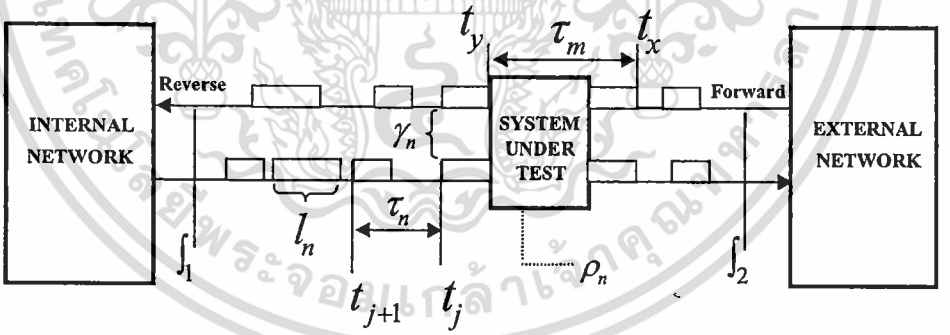
3.1.1.1 กลุ่มสมาชิกเครือข่ายภายใน(Internal Network Node) หมายถึง กลุ่มของโนดที่ได้รับความสนใจโดยมีหมายเลขไอพีอ้างอิงไปถึงอุปกรณ์สืบค้นเส้นทางเดียวกัน ได้แก่ กลุ่มเครือข่ายย่อย 202.29.4.0/26, 202.29.4.64/26 และ 202.29.4.128/26

3.1.1.2 กลุ่มที่ทำหน้าที่เป็นอุปกรณ์ตรวจสอบเส้นทาง(System Under Test Node) หมายถึง กลุ่มอุปกรณ์ที่ใช้หมายเลขไอพีที่ถูกอ้างอิงจากโนดเครือข่ายภายใน และรวมถึงหมายเลขไอพีที่ใช้เชื่อมต่อกับเครือข่ายภายนอก ได้แก่ 202.29.4.62, 202.29.4.126, 202.29.4.190, 202.29.4.254 และ 202.28.29.121

3.1.1.3 กลุ่มสมาชิกเครือข่ายภายนอก(External Network Node) หมายถึง กลุ่มของโนดที่ใช้หมายเลขไอพีแตกต่างจาก 2 กลุ่มแรก

3.1.2 พารามิเตอร์ที่ต้องการตรวจวัดกับทิศทางการสื่อสารข้อมูล

พิจารณาโมเดล(Model)ในการวิเคราะห์ที่มีการสื่อสารข้อมูลด้วยแพ็กเก็ตจากโนดต้นทางที่อยู่ฝั่งเครือข่ายภายใน(Internal Network) ส่งผ่านตัวระบบ(System Under Test) ไปยังโนดปลายทางที่อยู่ฝั่งเครือข่ายภายนอก(External Network) เมื่อทำการยุบรวมเครือข่ายย่อยแล้วจะสามารถนำไปพิจารณาพารามิเตอร์ที่เกี่ยวข้องกับการตรวจวัดในการวิจัย ดังแสดงได้ในภาพที่ 19



ภาพที่ 19 พารามิเตอร์ที่เกี่ยวข้องกับการตรวจวัด

จากภาพที่ 19 สามารถจำแนกวัตถุประสงค์ของพารามิเตอร์ที่ต้องการตรวจวัดได้ดังนี้

l_n หมายถึง ความยาวของแพ็กเก็ตใด ๆ ในงานวิจัยนี้กำหนดให้มีหน่วยการตรวจจับข้อมูลเป็นไบต์(Bytes)

τ_n หมายถึงผลต่างของเวลาระหว่างแพ็กเก็ตแรกกับแพ็กเก็ตถัดไป($|t_n, t_{n+1}|$) เดินทางถึงตัวระบบ ขอนิยามหน่วยเวลานี้ว่า เวลาระหว่างการมา(Inter-Arrival Times) มีหน่วยเป็นวินาทีและในงานวิจัยนี้กำหนดให้มีหน่วยเวลาตรวจจับเป็น มิลลิวินาที(MilliSecond)

γ_n หมายถึงผลต่างของเวลาระหว่างแพ็กเก็ตชนิดร้องขอ(Request) และตอบรับ(Response) ($|t_j, t_y|$) ขอนิยามหน่วยเวลานี้ว่า เวลาการสื่อสารไปและกลับ(Response Times) มีหน่วยเป็นวินาทีและในงานวิจัยนี้กำหนดให้ มีหน่วยเวลาตรวจจับเป็น มิลลิวินาที(MilliSecond)

τ_m หมายถึงผลต่างของเวลาที่แพ็กเก็ตเดียวกันผ่านเข้าและออกจากตัวระบบ ($|t_x, t_y|$) ขอนิยามหน่วยเวลานี้ว่า ความล่าช้า(Delay) มีหน่วยเป็นวินาทีและในงานวิจัยนี้กำหนดให้มีหน่วยเวลาตรวจจับเป็น มิลลิวินาที(MilliSecond)

ρ_n หมายถึงแพ็กเก็ตเดียวกันที่ไม่สามารถผ่านออกจากตัวระบบได้ ขอนิยามลักษณะการสื่อสารนี้ว่า ความสูญเสีย(Loss) และในงานวิจัยนี้กำหนดให้การตรวจจับและรวบรวมข้อมูลพิจารณาจากอัตราจำนวนของแพ็กเก็ตที่สูญเสียทั้งขาเข้าและออกจากตัวระบบ

$\int_1, \int_2$ หมายถึง จุดเชื่อมต่อเครือข่ายเพื่อการตรวจจับและนำข้อมูลไปใช้ในการวิเคราะห์ ซึ่งในงานวิจัยนี้กำหนดไว้ 2 จุด คือ ระหว่างทางเข้าและออกจากตัวระบบ

ในปัจจุบันพารามิเตอร์ที่ได้ความนิยมอีกชนิดหนึ่งก็คือ ข้อมูลโฟลว แต่การตรวจวัดข้อมูลโฟลวสามารถวิเคราะห์ได้จากเครื่องมือที่มีใช้แล้ว เช่น NetTraMet, NetFlow, Cflowd, FlowScan หรือ CoralReef ซึ่งข้อมูลโฟลวจะมีประโยชน์ต่อการนำไปใช้วิเคราะห์ จำนวน ความยาวรวมและเวลาของแพ็กเก็ตต่อการสื่อสารครั้งหนึ่ง ๆ ระหว่างผู้รับและส่งข้อมูลได้ และในงานวิจัยนี้ก็สามารถนำวิธีการที่ได้ออกแบบไปใช้กับการวิเคราะห์ข้อมูลโฟลวได้เช่นเดียวกัน

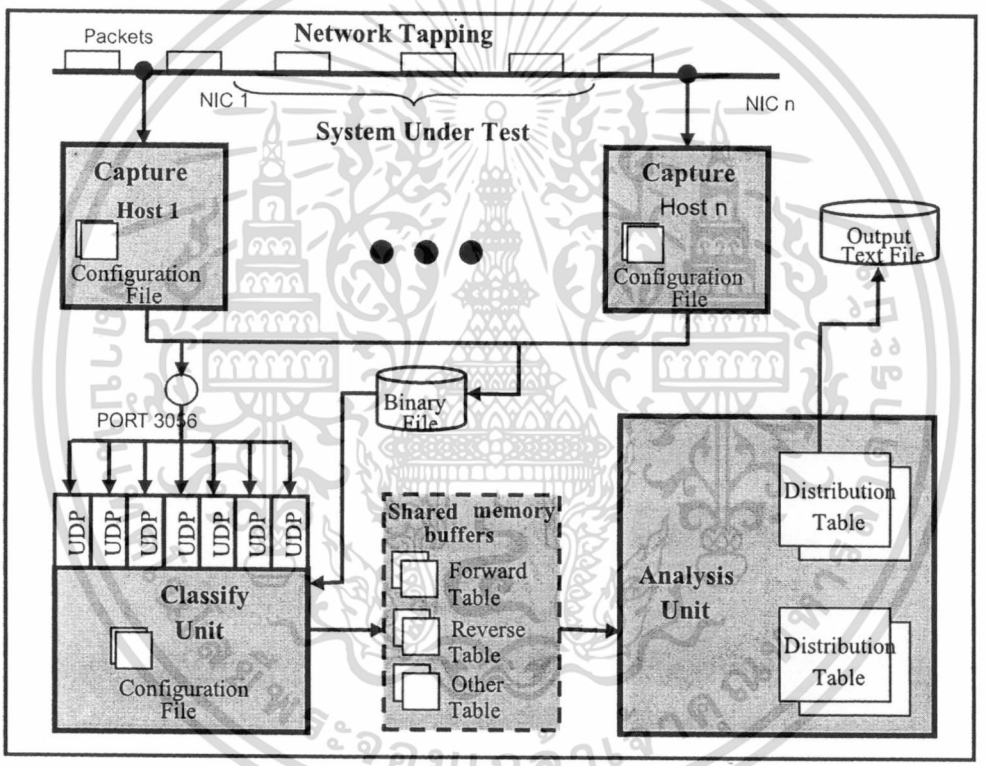
ในงานวิจัยนี้กำหนดให้พารามิเตอร์ในแต่ละชนิดสามารถจำแนกตามทิศทางการสื่อสารข้อมูลได้ 2 ลักษณะ คือ

3.1.2.1 พารามิเตอร์ที่ตรวจวัดได้จากข้อมูลในทิศทางเดียว(One way direction parameters) หมายถึง พารามิเตอร์ที่สามารถตรวจวัดได้จากกลุ่มแพ็กเก็ตสื่อสารตามทิศทางใดทิศทางหนึ่งทั้งจากขาไปหรือขากลับ(Forward หรือ Reverse) ได้แก่ ความยาว, เวลาระหว่างการมา, ความล่าช้า, ความสูญเสีย และ ข้อมูลโฟลว

3.1.2.2 พารามิเตอร์ที่ตรวจวัดได้จากข้อมูลในทั้งสองทิศทาง(Two way direction parameters) หมายถึง พารามิเตอร์ที่ต้องนำข้อมูลทั้งสองทิศทางไปประมวลผลร่วมกัน ได้แก่ เวลาการสื่อสารข้อมูลไปและกลับ

3.1.3 โครงสร้างวิธีการตรวจวัดข้อมูล

เครื่องมือวัดในงานวิจัยได้ออกแบบให้สามารถตรวจวัดข้อมูลจริงบนเครือข่ายได้ ดังนั้นจึงออกแบบให้มีการทำงานแยกหน่วยทำงานที่สำคัญออกจากกัน แต่ละหน่วยทำงานต้องแยกเป็นอิสระต่อกันและสามารถทำงานเป็นแบบคู่ขนานได้ ด้วยคุณสมบัตินี้จึงจัดทำเครื่องมือตรวจวัดขึ้นบนระบบปฏิบัติการ FreeBSD ที่มีลักษณะการทำงานเหมือนกับระบบปฏิบัติการ Unix ทั่วไป และมีข้อดีในการสร้างโปรแกรมด้วยรูปแบบ การแบ่งเวลาประมวลผล(Multi-Processing) ได้ ดังแสดงแต่ละหน่วยทำงานได้ในภาพที่ 20



ภาพที่ 20 หน่วยประมวลผลข้อมูล

โครงสร้างของเครื่องมือวัดได้ออกแบบให้แบ่งหน่วยทำงานสำคัญ ให้มีหน้าที่หลักดังนี้

3.1.3.1 หน่วยรวบรวมข้อมูล(Capture Unit) มีหน้าที่ตรวจจับและรวบรวมแพ็กเก็ตบนเครือข่ายเพื่อส่งต่อให้กับหน่วยแยกชนิดข้อมูล(Classify Unit) ซึ่งสามารถทำได้ 2 รูปแบบ คือ ส่งข้อมูลผ่านเครือข่ายทันทีโดยใช้แพ็กเก็ตข้อมูลชนิด UDP(User Datagram Protocol) หรือบันทึกข้อมูลเก็บไว้บนดิสค์เป็น Binary File เพื่อรอการประมวลผลจากหน่วยแยกชนิดในภายหลัง

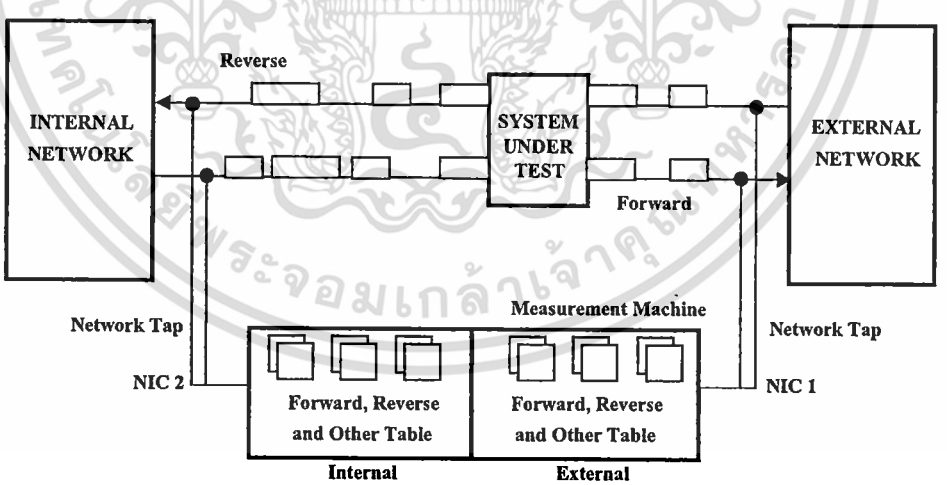
3.1.3.2 หน่วยแยกชนิดข้อมูล(Classifying Unit) ทำหน้าที่แยกชนิดข้อมูลที่ได้รับมาจากหน่วยรวบรวมข้อมูล โดยจัดเก็บในรูปแบบตารางฐานข้อมูลแยกตามทิศทางการสื่อสารบนหน่วยความจำที่สามารถใช้ร่วมกันได้(Shared memory buffers)จากหน่วยประมวลผลอื่น ๆ ตารางข้อมูลจะแบ่งออกเป็น 2 ส่วนหลักตามจำนวน Interface ที่มีการตรวจจับข้อมูล คือ

- ตารางจัดเก็บข้อมูลตามทิศทางขาออก(Forward Table) หมายถึง ตารางที่ใช้จัดเก็บข้อมูล ซึ่งทำหน้าส่งจากเครือข่ายภายใน(Internal Network)ผ่านตัวระบบไปยังเครือข่ายภายนอก (External Network)

- ตารางจัดเก็บข้อมูลในทิศทางขาเข้า(Reverse Table) หมายถึง ตารางที่ใช้จัดเก็บข้อมูล ซึ่งทำหน้าที่ส่งจากเครือข่ายภายนอก ผ่านตัวระบบ ไปยัง เครือข่ายภายใน

- ตารางจัดเก็บข้อมูลอื่น ๆ (Other Table) หมายถึง ตารางที่ใช้จัดเก็บข้อมูล ซึ่งทำหน้าที่นอกเหนือจาก 2 ตารางแรก

ดังนั้นจำนวนตารางข้อมูลจะขึ้นอยู่กับจำนวนจุดเชื่อมต่อของหน่วยตรวจจับข้อมูล เช่น ถ้ามีจุดเชื่อมต่อ 2 จุด ทำหน้าที่ตรวจจับข้อมูลฝั่งขาเข้า(Internal)และฝั่งขาออก(External)จากตัวระบบ แต่ละฝั่งก็จะทำการแยกชนิดข้อมูลบนตารางข้อมูลทั้ง 3 ชนิดตามที่ได้ออกแบบไว้ จากนั้นหน่วยวิเคราะห์ข้อมูลก็จะนำข้อมูลที่ได้ไปใช้ประมวลผลรวมกันเพื่อวิเคราะห์ถึงผลกระทบต่าง ๆ ที่เกิดจากจากการสื่อสารข้อมูลผ่านตัวระบบต่อไป ดังแสดงได้ในภาพที่ 21



ภาพที่ 21 การแบ่งตารางฐานข้อมูลฝั่งขาเข้าและออกจากตัวระบบ

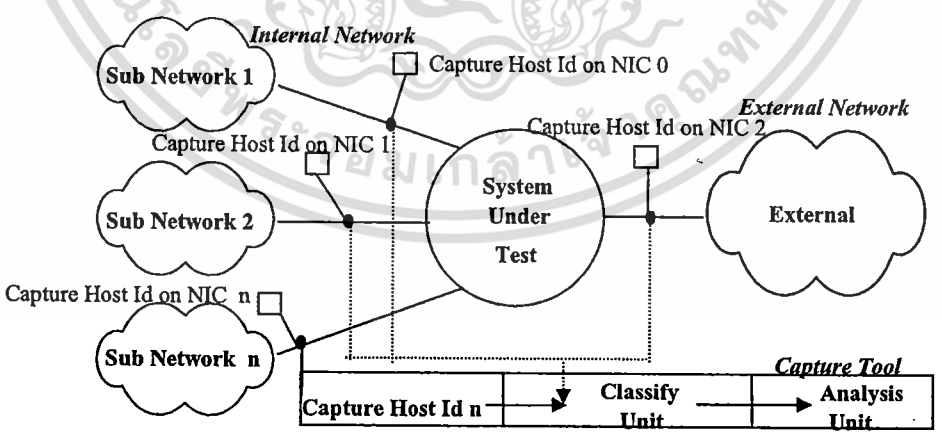
จากภาพที่ 21 หน่วยรวบรวมและแยกชนิดข้อมูลอาจติดตั้งอยู่บนคอมพิวเตอร์เครื่องเดียวกัน (Measurement Machine) การตรวจจับข้อมูลจะขึ้นอยู่กับจำนวนของ NIC(Network Interface Card) ที่

ติดตั้งบนเครื่องคอมพิวเตอร์ แต่ละ NIC จะทำการตรวจจับข้อมูลและนำไปแบ่งแยกเพื่อจัดเก็บบนตารางข้อมูลในแต่ละชนิด คือ ตารางจัดเก็บข้อมูลขาออก(Forward Table), ตารางจัดเก็บข้อมูลขาเข้า (Reverse Table) และ ตารางจัดเก็บข้อมูลอื่น ๆ(Other Table)

3.1.3.3 หน่วยวิเคราะห์ข้อมูล(Analysis Unit) มีหน้าที่อ่านข้อมูลจากตารางฐานข้อมูลที่ถูกแยกชนิดแล้วโดยหน่วยแยกชนิด และจัดเก็บไว้บนหน่วยความจำร่วม ข้อมูลจะถูกนำไปวิเคราะห์พฤติกรรมการสื่อสารข้อมูลตามพาราเตอร์ต่าง ๆ และจัดเก็บในรูปของตารางแจกแจงความถี่ (Distribution Table)บนหน่วยความจำ เมื่อผู้ใช้หยุดการตรวจวัดข้อมูลตารางแจกแจงความถี่ทั้งหมดจะถูกนำไปบันทึกไว้ในแฟ้มข้อมูลเพื่อนำไปใช้วิเคราะห์ผลทางสถิติต่อไป ขบวนการภายในของหน่วยวิเคราะห์จะมีการแบ่งแยกเป็นโพรเซสย่อย(Child Process) ให้ทำหน้าที่วิเคราะห์พารามิเตอร์แต่ละตัวไปพร้อม กัน ซึ่งพามิเตอร์ที่สามารถตรวจวัดได้ คือ ความยาวแพ็กเก็ต เวลาระหว่างการมา เวลาการสื่อสาร ข้อมูลไปกลับ ความล่าช้า ความสูญเสีย และข้อมูลโฟลว ซึ่งการแยกเป็นหน่วยวัดเป็นโพรเซสย่อยนั้น ทำให้มีข้อดีที่ทำให้ผู้วิเคราะห์สามารถเลือกตรวจวัดเฉพาะพามิเตอร์ใดก็ได้

3.2 การตรวจจับข้อมูลบนเครือข่าย

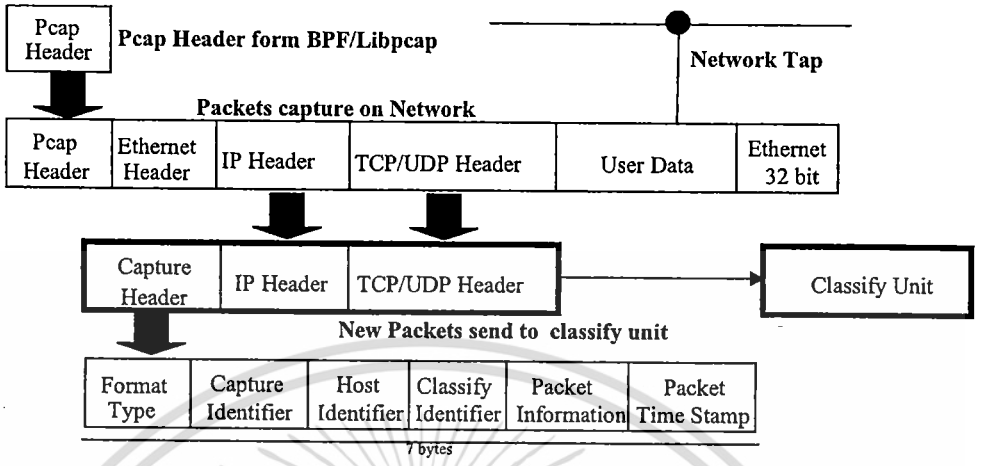
เนื่องจากหน่วยรวบรวมข้อมูลได้ถูกออกแบบให้มีได้หลายหน่วยและสามารถตรวจจับข้อมูลได้เท่ากับจำนวนของ NIC ที่มีอยู่บนตัวระบบ จึงมีข้อดีทำให้หน่วยรวมรวมข้อมูลสามารถกระจายจุดเชื่อมต่อบนเครือข่าย(Network Tap)ได้ ซึ่งจุดเชื่อมต่ออาจอยู่ได้ทั้งขาเข้าและขาออกหรืออาจไม่อยู่ในพื้นที่เดียวกันก็ได้ ดังแสดงได้ในภาพที่ 22



ภาพที่ 22 การเชื่อมต่อหน่วยตรวจจับข้อมูล

จากภาพที่ 22 หน่วยตรวจจับข้อมูล(Capture Host)ได้ออกแบบเผื่อไว้ให้มีได้มากกว่า 1 หน่วย แต่ท้ายที่สุดเครื่องมือวิจัยจะต้องรวบรวมทุกหน่วยให้เหลือเพียง 2 หน่วยคือ หน่วยที่ทำหน้าที่ตรวจจับข้อมูลจากเครือข่ายภายใน และหน่วยที่ทำหน้าที่ตรวจจับข้อมูลจากภายนอก หรือทุกหน่วยอาจได้รับการตรวจจับข้อมูลอยู่ภายในหน่วยเดียวกันก็ได้ คือ อยู่รวมกันทั้งหน่วยรวบรวมข้อมูล หน่วยแยกชนิด (Classify Unit)และหน่วยวิเคราะห์ข้อมูล(Analysis Unit) ซึ่งจากภาพหน่วยตรวจจับข้อมูลอยู่บนเครื่องคอมพิวเตอร์ที่ต่างกัน แต่ละเครื่องติดตั้ง NIC จำนวน 1 หน่วย ในงานวิจัยอาจกำหนดให้ NIC หมายเลข 0,1 และ n(Capture Host Id on NIC 0,1 และ n)ทำหน้าที่รวบรวมข้อมูลที่จัดอยู่ในฝั่งเครือข่ายภายใน(Internal Network) ส่วนหน่วยตรวจจับ NIC หมายเลข 2(Capture Host on NIC Id 2)ทำหน้าที่รวบรวมข้อมูลที่จัดอยู่ฝั่งเครือข่ายภายนอก(External Network) ซึ่งหน่วยแยกชนิดข้อมูลจะต้องทราบข้อมูลเหล่านี้ก่อนการประมวลผล ด้วยการสร้างข้อตกลงร่วมกันระหว่างหน่วยประมวลผลต่าง ๆ นอกจากนี้แต่ละ NIC บนหน่วยตรวจจับข้อมูลยังมีผลกระทบต่อเวลาบนเครื่องคอมพิวเตอร์ที่แตกต่างกัน ดังนั้นในงานวิจัยนี้จึงออกแบบให้ใช้วิธีที่หน่วยตรวจจับข้อมูลทำงานอยู่บนคอมพิวเตอร์เครื่องเดียวกันไปก่อน เพื่อรับประกันการกำกับเวลา(Time Stamp)ข้อมูลจากการทำงานบนเครื่องคอมพิวเตอร์ที่ต่างกัน ข้อมูลจาก NIC ที่ทุกหน่วยตรวจจับได้รับจะถูกกรองและนำไปใช้ในการวิเคราะห์เฉพาะแพ็กเก็ตที่มีชนิดเป็น IP รุ่นที่ 4 เท่านั้น ก่อนที่หน่วยตรวจจับจะทำการปรับปรุงทุกแพ็กเก็ตให้อยู่ในรูปที่สามารถประมวลผลได้โดยหน่วยแยกชนิดข้อมูลและหน่วยวิเคราะห์ข้อมูลต่อไป

ข้อมูลที่ตรวจจับได้จากเครือข่ายอาจมีความยาวที่ไม่เท่ากัน ขึ้นอยู่กับแพ็กเก็ตเหล่านั้นว่าทำหน้าที่ใดไปยังปลายทาง แพ็กเก็ตที่ทำหน้าที่ลำเลียงข้อมูลอาจมีความยาวที่สูงกว่าแพ็กเก็ตชนิดอื่น ๆ ดังนั้นเพื่อให้เกิดความรวดเร็วในการประมวลผลก่อนที่จะส่งต่อไปยังหน่วยแยกชนิดข้อมูลจึงต้องออกแบบให้มีเฉพาะส่วนที่จำเป็นต่อการวิเคราะห์เท่านั้น คือ นำไปใช้เฉพาะส่วนหัวของแพ็กเก็ต (Packet Header) นอกจากนี้เพื่อให้หน่วยแยกชนิดสามารถจำแนกที่มาของข้อมูลว่ามาจาก NIC บนหน่วยตรวจจับข้อมูลใด จึงต้องผนวกส่วนหัวที่ประกอบด้วยรายละเอียดของหน่วยตรวจจับข้อมูล (Capture Header)ไปด้วย โดยแพ็กเก็ตที่ตรวจจับได้จากเครือข่ายด้วยขบวนการของ BPF/libpcap ซึ่งประกอบด้วย Ethernet Header, IP Header, TCP หรือ UDP Header, User Data และ Ethernet 32 bit แต่ละส่วนจะถูกนำไปสร้างเป็นแพ็กเก็ตใหม่ในงานวิจัยโดยนำไปผนวกเฉพาะส่วน IP Header และ TCP/UDP Header เท่านั้น ก่อนการส่งต่อให้กับหน่วยแยกชนิด ซึ่งงานวิจัยได้กำหนดส่วนหัวใหม่ที่ต้องผนวกเพิ่มโดยออกแบบให้ใช้เนื้อที่ขนาด 7 ไบต์เท่านั้น เพื่อบรรจุรายละเอียดของหน่วยตรวจจับข้อมูล ดังแสดงได้ในภาพที่ 23



ภาพที่ 23 รูปแบบของแพ็กเก็ตที่ส่งให้กับหน่วยแยกชนิด

จากภาพที่ 23 ส่วนหัวที่ประกอบขึ้นใหม่ในงานวิจัยจะประกอบด้วยข้อมูลที่สำคัญดังนี้

Format Type มีขนาด 8 bits ใช้บอกรูปแบบของข้อมูลที่ต้องการจัดส่งให้กับหน่วย แยกชนิด ในงานวิจัยนี้กำหนดไว้เพียงรูปแบบเดียวคือ ส่งเป็น Binary โดยกำหนดให้ Format Type มีค่าเป็น 0

Capture Identifier มีขนาด 8 bits ใช้บอกตำแหน่งของหน่วยตรวจจับข้อมูล ซึ่งจะถูกนำไปใช้ในหน่วยแยกชนิด เพื่อระบุว่าข้อมูลนี้เป็นข้อมูลที่ตรวจจับได้จากฝั่งเครือข่ายภายใน(Internal Network) หรือฝั่งเครือข่ายภายนอก(External Network) หรือในทางอุปกรณ์จะใช้แทนค่าของ Interface Card ที่เชื่อมต่อกับเครือข่าย

Host Identifier มีขนาด 8 bits ใช้บอกที่มาของข้อมูลว่ามาจากคอมพิวเตอร์เครื่องใด

Classifying Identifier มีขนาด 8 bits ใช้บอกสถานะ การแยกชนิดของข้อมูล ในงานวิจัยนี้กำหนดไว้ 3 สถานะ คือ 0 หมายถึง ยังไม่มีการแยกชนิด, 1 หมายถึง เป็นข้อมูลที่มีทิศทางการสื่อสารขาออก(Forward Direction), 2 หมายถึง เป็นข้อมูลที่มีทิศทางการสื่อสารขาเข้า(Reverse Direction) และ 255 หมายถึง เป็นข้อมูลอื่น ๆ ที่แตกต่างจากสถานภาพ 1 และ 2

Packet Information มีขนาด 8 bits ใช้กำหนดรูปแบบการส่งข้อมูลไปยังหน่วย แยกชนิด โดยมีค่าแสดงสถานภาพต่าง ๆ ดังนี้

- 0 หมายถึง ส่งเฉพาะ Capture Information Header
- 1 หมายถึง ผนวกส่วน IP Header ไปด้วย
- 3 หมายถึง ผนวกส่วน TCP/UDP Header ไปด้วย

Packet Time Stamp มีขนาด 16 bytes ใช้บอกเวลาที่แพ็กเก็ตถูกตรวจจับได้จากขบวนการของ Libpcap ซึ่งมีโครงสร้างดังนี้

```
struct pcap_pkthdr {
        struct timeval ts;      time stamp
        bpf_u_int32 caplen;     length of portion present
        bpf_u_int32 len;        length this packet (off wire)
}
```

โดยที่ ts จัดเก็บการกำกับเวลาในหน่วยวินาทีตามโครงสร้าง timeval ขนาด 16 ไบต์

caplen ความยาวรวมที่ pcap ตรวจจับข้อมูลได้

length ความยาวของแพ็กเก็ตที่ตรวจจับได้

ts จะถูกนำไปแทนค่าให้กับ Packet Time Stamp ในส่วนหัวที่ประกอบขึ้นใหม่ ตามโครงสร้าง timeval ซึ่งประกอบด้วยลักษณะดังนี้

```
struct timeval {
    u_int32 tv_sec;
    u_int32 tv_usec;
};
```

โดยที่ tv_sec เป็นส่วนที่ใช้เก็บหน่วยเวลาในรูปของวินาที(รวม ชั่วโมง นาที และวินาที) ณ ปัจจุบัน

tv_usec เป็นส่วนที่ใช้เก็บหน่วยเวลาที่ต่ำกว่า วินาที คือเป็น ไมโครวินาที (Micro Second) ณ ปัจจุบัน

3.3 การแยกชนิดข้อมูลเครือข่าย

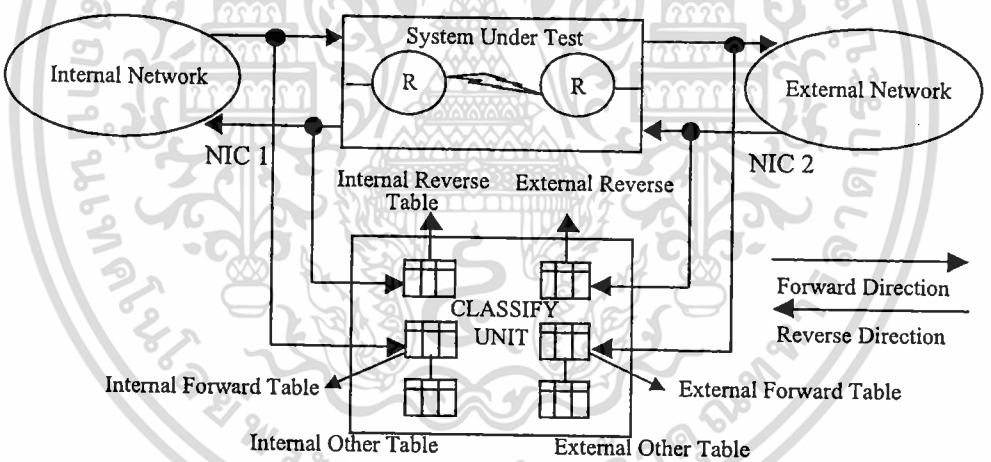
3.3.1 การออกแบบหน่วยแยกชนิดข้อมูล

หน่วยแยกชนิดข้อมูล(Classifying Unit)เป็นหน่วยที่ทำหน้าที่ในการรวบรวมข้อมูลจากแต่ละ NIC บนหน่วยตรวจจับข้อมูล หรือข้อมูลอาจได้มาจากแต่ละ NIC ด้วยการรับส่งข้อมูลผ่านเครือข่ายโดยการเปิดพอร์ตสำหรับรับส่งข้อมูลชนิด UDP หมายเลขเดียวกับหน่วยตรวจจับข้อมูล หรืออาจได้จากการอ่านข้อมูลบนแฟ้มที่สะสมจากแต่ละ NIC ซึ่งมีรูปแบบการจัดเก็บตามข้อตกลงเดียวกันระหว่างหน่วยตรวจจับและแยกชนิดข้อมูล ก่อนการวิเคราะห์จะต้องกำหนดก่อนว่าแต่ละ NIC ทำงานตรวจจับข้อมูลในฝั่งใดระหว่างตัวระบบ และขอนิยามเครือข่ายที่ได้รับความสนใจว่า เครือข่ายภายใน

(Internal Network) ส่วนเครือข่ายอื่นจากอีกด้านหนึ่งของตัวระบบ เรียกว่า เครือข่ายภายนอก(External Network)ข้อมูลดิบที่ตรวจจับได้ในแต่ละ NIC จะถูกนำไปแบ่งแยกเพื่อจัดเก็บลงตารางข้อมูล 3 ชนิด โดยแยกตามทิศทางการสื่อสารข้อมูลคือ ขาออก(Forward Direction) และ ขาเข้า(Reverse Direction) ดังนี้

ส่วนของ NIC ที่ตรวจจับข้อมูลด้านเครือข่ายภายใน แบ่งเป็น ตารางจัดเก็บข้อมูลตามทิศทางการสื่อสารข้อมูลขาออกในฝั่งเครือข่ายภายใน(Internal Forward Table) ตารางจัดเก็บข้อมูลตามทิศทางการสื่อสารข้อมูลขาเข้าในฝั่งเครือข่ายภายใน(Internal Reverse Table) และตารางจัดเก็บข้อมูลอื่น ๆ(Internal Other Table)

ส่วนของ NIC ที่ตรวจจับข้อมูลฝั่งเครือข่ายภายนอกแบ่งเป็น ตารางจัดเก็บข้อมูลตามทิศทางการสื่อสารข้อมูลขาออกในฝั่งเครือข่ายภายนอก(External Forward Table) ตารางจัดเก็บข้อมูลตามทิศทางการสื่อสารข้อมูลขาเข้าในฝั่งเครือข่ายภายนอก(External Reverse Table) และตารางจัดเก็บข้อมูลอื่น ๆ(External Other Table) ดังแสดงได้ในภาพที่ 24



ภาพที่ 24 หน่วยแยกชนิดและตารางจัดเก็บข้อมูล

3.3.2 โครงสร้างและตารางจัดเก็บข้อมูล

ตารางข้อมูลในงานวิจัยได้ออกแบบให้จัดเก็บข้อมูลบนหน่วยความจำของเครื่องคอมพิวเตอร์ รูปแบบการจัดเก็บของทุกตารางข้อมูลมีลักษณะที่เหมือนกัน โดยรวบรวมข้อมูลเฉพาะส่วนที่มีความสำคัญต่อการนำไปใช้ในการวิเคราะห์เท่านั้น ซึ่งข้อมูลส่วนใหญ่ได้มาจากส่วนหัวของแพ็กเก็ตและบอกคุณลักษณะข้อมูลที่อยู่ในชั้น Network และ Transport Layer เป็นหลัก ข้อมูลในทุกระเบียน(Record) เมื่อถูกแยกชนิดแล้วจะถูกจัดเก็บไว้บนบนตารางข้อมูลแต่ละชนิดโดยใช้การเขียน

โปรแกรมจัดการแบบหน่วยความจำร่วม(Share Memory) เพื่อรอการเรียกใช้จากหน่วยวิเคราะห์ข้อมูล แต่ละระเบียนมีโครงสร้างข้อมูลหลักที่ประกอบไปด้วยคุณลักษณะดังแสดงได้ในตารางที่ 1 ดังนี้

ตารางที่ 1 โครงสร้างตารางข้อมูลแต่ละชนิด

Size (Bytes)	Name	Source From	Definition
0 – 1	ordinal	New Define	Sequence Number of Record
2 – 9	Timeinvl	New Define	Inter-Arrival Time
10 – 25	timestamp	Capture Header	Time Stamp with Libpcap Library
26 – 27	Iplen	IP Header	IP Packet Length
27 – 28	Ipid		IP Packet Identifier
29 – 29	ippro		IP Protocol
30 – 33	ipsrc		IP Source Address
34 – 37	ipdst		IP Destination Address
38 – 38	icmptype	TCP / UDP Header And ICMP Header	ICMP Type
39 – 39	icmpcode		ICMP Code
40 – 41	icmpid		ICMP Identifier Number for ECHO request / response
42 – 43	icmpseq		ICMP sequence Number for ECHO request / response
44 – 45	tcpsport		TCP Source Port
46 – 47	tcpdport		TCP Destination Port
48 – 51	tcpseq		TCP Sequence Number
52 – 55	tcpack		TCP Acknowledgement Number
56 – 56	tcpflags		TCP Flag such as ACK, FIN, PUSH
57 – 58	udpsport		UDP Source Port
59 – 60	udpdport		UDP Destination Port
61 – 62	udplen		UDP Packet Length
63 – 63	flags	New Define	Status for reuse memory

ผลรวมขนาดของแต่ละระเบียนในทุกตารางข้อมูลจะมีขนาดเท่ากับ 63 ไบต์(Bytes) แต่ละตารางจองพื้นที่จัดเก็บข้อมูลไว้บนหน่วยความจำเป็นจำนวนคงที่ คือ 10,000 ระเบียน ดังนั้นผลรวมของการใช้หน่วยความจำสำหรับจัดเก็บข้อมูลแต่ละชนิดจึงมีขนาดเท่ากับ 3,780,000 Bytes ในทุกตาราง ซึ่งทำให้การวิเคราะห์ข้อมูลด้วยเครื่องมือวิจัย สามารถติดตั้งบนเครื่องคอมพิวเตอร์ที่มีหน่วยความจำต่ำสุดเพียงแค่ 16 MB ก็สามารถทำงานได้

3.3.3 ขั้นตอนการแยกชนิดข้อมูล

การแบ่งแยกทิศทางการสื่อสารและจัดเก็บข้อมูลลงบนตารางแต่ละชนิด สามารถแสดงได้ดังภาพที่ 25

```
For i = 1 to Number of Packet in Interface Table {
    If (pkt_i ∈ IP packet version 4) {
      If (pkt_i.ip_dst ∉ SUT && pkt_i.ip_dst ∉ INT)
          Save packet information to Forward Table;
      else if ((pkt_i.ip_src ∉ SUT && pkt_i.ip_src ∉ INT) &&
            pkt_i.ip_dst ∈ INT)
          Save packet information to Reverse Table;
      else  Save packet information to Other Table;
    } else  Save packet information to Other Table;
}
/* SUT set of IP Address in System Under Test */
/* INT set of IP Address in Internal Network */
```

ภาพที่ 25 ขั้นตอนการแบ่งแยกชนิดข้อมูล

พิจารณาข้อมูลที่ตรวจจับได้โดย NIC จากเครือข่าย(Packet in Interface Table)จะถูกกรองไปใช้เฉพาะที่มีชนิดเป็นไอพีแพ็กเก็ตเท่านั้น ไอพีต้นทางและปลายทางของแพ็กเก็ตจะเป็นส่วนสำคัญที่นำไปทดสอบเงื่อนไขเพื่อแยกจัดเก็บข้อมูลตามตารางต่าง ๆ โดยจะมีเงื่อนไขที่สำคัญดังนี้

เงื่อนไขที่ 1 กรณีที่ไอพีปลายทางของแพ็กเก็ตไม่เป็นส่วนหนึ่งของไอพีที่ใช้อยู่บนตัวระบบหรือใช้สัญลักษณ์เรียกว่า "SUT(System Under Test)" และไอพีต้นปลายทางไม่เป็นส่วนหนึ่งของไอพีที่ใช้อยู่บนเครือข่ายภายใน หรือใช้สัญลักษณ์เรียกว่า "INT(Internal Network)" ให้ถือว่าเป็นแพ็กเก็ตที่ทำ

หน้าที่ส่งข้อมูลออกจากเครือข่ายภายใน ดังนั้นต้องจัดเก็บข้อมูลไว้บนตารางข้อมูลในทิศทางขาออก (Forward Table)

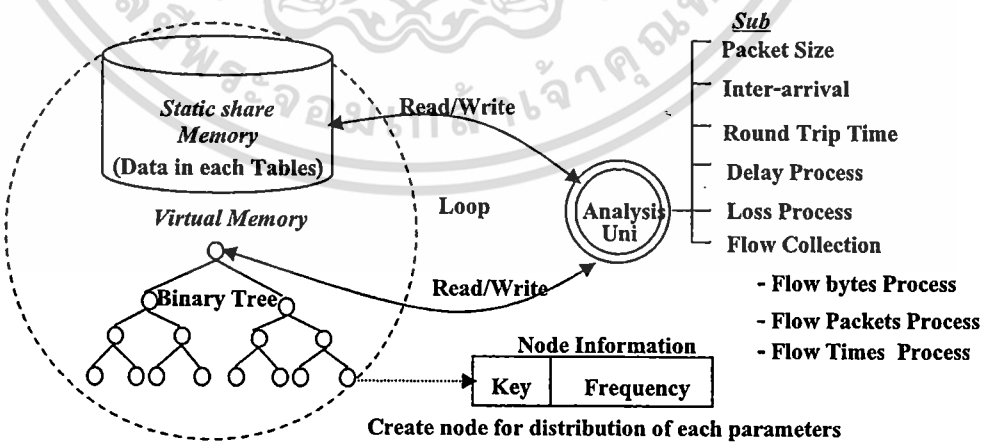
เงื่อนไขที่ 2 กรณีที่ไอพีต้นทางของแพ็กเก็ตไม่เป็นส่วนหนึ่งของไอพีที่ใช้อยู่บนตัวระบบและเครือข่ายภายใน รวมถึงไอพีปลายทางของแพ็กเก็ตไม่เป็นส่วนหนึ่งของไอพีที่อยู่บนเครือข่ายภายในด้วย ให้ถือว่าเป็นแพ็กเก็ตที่ทำหน้าที่ส่งข้อมูลมาจากภายนอกเครือข่าย ดังนั้นต้องจัดเก็บข้อมูลไว้บนตารางข้อมูลในทิศทางขาเข้า(Reverse Table)

นอกจากทั้งสองเงื่อนไขก่อนหน้า ให้ถือว่าเป็นไอพีแพ็กเก็ตอื่น ๆ ที่ยังไม่ได้รับความสนใจ เช่น ไอพีแพ็กเก็ตที่รับส่งข้อมูลกันเฉพาะภายในเครือข่าย หรืออาจเป็นไอพีแพ็กเก็ตที่สื่อสารกันระหว่างเครือข่ายภายในกับตัวระบบ ข้อมูลเหล่านี้ให้เก็บไว้ในตารางข้อมูลอื่น ๆ(Other Table) เพื่อรอการวิเคราะห์ต่อไป

3.4 การวิเคราะห์ข้อมูลเครือข่าย

3.4.1 การออกแบบหน่วยวิเคราะห์ข้อมูล

หน่วยวิเคราะห์ข้อมูลได้ออกแบบให้แบ่งการทำงานเพื่อวิเคราะห์พารามิเตอร์ต่าง ๆ เป็นโพรเซสย่อย(Sub Process) แต่ละโพรเซสจะอ่านข้อมูลจากหน่วยความจำร่วมเพื่อผ่านการประมวลผลด้วยกระบวนการที่แตกต่างกัน ตารางผลลัพธ์ที่แต่ละโพรเซสสร้างขึ้นใช้วิธีการบริหารและจัดการหน่วยความจำแบบต้นไม้(Binary Tree) ทุกโพรเซสจะทำงานไปพร้อมกันจนกว่าจะได้รับสัญญาณหยุดการทำงานจากผู้ใช้ จึงจะเขียนตารางผลลัพธ์ทั้งหมดในรูปแบบแฟ้มข้อความ(Text File) บนดิสก์ ดังแสดงได้ในภาพที่ 26



ภาพที่ 26 ขบวนการวิเคราะห์ข้อมูล

หน่วยวิเคราะห์ข้อมูลจะประกอบด้วยโพรเซสย่อยดังนี้

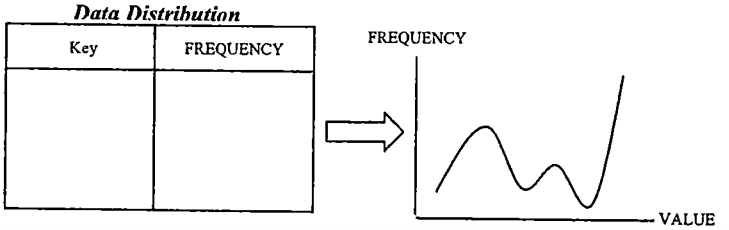
- โพรเซสวิเคราะห์ความยาวแพ็กเก็ต(Packet Size Process)
- โพรเซสวิเคราะห์เวลาระหว่างการมา(Interarrival Process)
- โพรเซสวิเคราะห์เวลาการสื่อสารไปและกลับ(Round Trip Time Process)
- โพรเซสวิเคราะห์ความล่าช้า(Delay Process)
- โพรเซสวิเคราะห์ความสูญเสีย(Loss Process)
- โพรเซสวิเคราะห์ข้อมูลโฟลว(Flow Collection Process) ประกอบด้วยโพรเซสย่อย ๆ อีก 3 หน่วย คือ โพรเซสวิเคราะห์ความยาวโฟลว(Flow Bytes Process) โพรเซสวิเคราะห์จำนวนแพ็กเก็ตในโฟลว(Flow Packets Process) และโพรเซสวิเคราะห์เวลาโฟลว(Flow Times Process)

แต่ละโพรเซสจะมีหน้าที่ในการอ่านข้อมูลและสร้างตารางผลลัพธ์ที่มีคุณสมบัติตามชื่อ โพรเซส ด้วยกระบวนการแบบต้นไม้ โดยมีโครงสร้างของโนดที่แสดงได้ในภาพที่ 27

```
typedef struct _node {
    unsigned short key;
    int frequency;
    struct _node *left;
    struct _node *right;
} NODE;
```

ภาพที่ 27 โครงสร้างโนด

แต่ละโนดได้ออกแบบให้สอดคล้องกับตารางแจกแจงความถี่ข้อมูล(Data Distribution Table) ตามกระบวนการทางสถิติที่ประกอบด้วยข้อมูล 2 ชนิด คือ Value หมายถึง มูลค่าของข้อมูล ในที่นี้ใช้ตัวแปรแทนความหมายด้วย Key และ ความถี่(Frequency) หมายถึง จำนวนเหตุการณ์ที่เกิดมูลค่าของข้อมูลขึ้นซ้ำ ๆ กัน ตารางแจกแจงความถี่สามารถนำไปใช้ประโยชน์ได้หลายด้าน ได้แก่ การนำไปใช้สร้างแผนภูมิ (Histogram) เพื่อแสดงคุณลักษณะของการเกิดเหตุการณ์ หรือวิเคราะห์ค่าเป็นตัวเลขทางสถิติเพื่อบอกความหมายต่าง ๆ เช่น ค่าต่ำสุด(Minimum) , ค่าสูงสุด(Maximum), ค่าเฉลี่ย(Average), ค่าความแปรปรวน(Variance) และ ค่าเบี่ยงเบนมาตรฐาน(Standard Division) เป็นต้น ดังแสดงได้ในภาพที่ 28



ภาพที่ 28 ตัวอย่างการแจงแจงความถี่และกราฟ

ตารางผลลัพธ์ได้ออกแบบเพื่ออธิบายคุณลักษณะข้อมูลได้หลากหลายชนิด สำหรับงานวิจัยได้จำแนกผลลัพธ์ออกตามทิศทางการสื่อสารข้อมูลด้วยการวิเคราะห์จากตารางข้อมูลในแต่ละชนิด โดยแบ่งผลลัพธ์ออกเป็น 2 ลักษณะ ดังนี้

3.4.1.1 ตารางผลลัพธ์ที่วิเคราะห์ได้จากตารางข้อมูลเดียว (One Table Parameters)

หมายถึง คุณลักษณะของข้อมูลที่รวบรวมจากทุกแพ็กเก็ตที่มีการสื่อสารข้อมูลไปในทิศทางเดียวกันหรืออยู่ในตารางข้อมูลเดียวกัน คือ จากตารางข้อมูลขาออก(Forward Table) หรือจากตารางข้อมูลขาเข้า(Reverse Table) พารามิเตอร์ที่วิเคราะห์คุณลักษณะข้อมูลเหล่านี้ ได้แก่ ขนาดแพ็กเก็ต, เวลาระหว่างการมา และ ข้อมูลโฟลว

3.4.1.2 ตารางผลลัพธ์ที่วิเคราะห์ได้จาก 2 ตารางข้อมูล (Two Table Parameters)

หมายถึง คุณลักษณะของข้อมูลที่รวบรวมขึ้นด้วยการนำข้อมูลจากทั้งสองตารางมาพิจารณาร่วมกัน โดยแบ่งที่มาของการวิเคราะห์ผลลัพธ์ออกเป็น 2 รูปแบบคือ

รูปแบบที่ 1 วิเคราะห์ตามทิศทางเดียวกันจากข้อมูลตามฝั่งขาเข้าหรือขาออก โดยการเปรียบเทียบข้อมูลจากทั้ง 2 ตารางที่มีทิศทางการสื่อสารทางเดียวกัน เช่น การวิเคราะห์ความล่าช้า และความสูญเสียที่เกิดขึ้นจากตัวระบบตามทิศทางการสื่อสารข้อมูลขาออกสามารถพิจารณาได้จากข้อมูลใน 2 ตารางคือ ตารางการสื่อสารข้อมูลฝั่งเครือข่ายภายในและภายนอกตามทิศทางขาออก (Internal and External Forward Table)

รูปแบบที่ 2 วิเคราะห์ทิศทางตรงข้ามกันจากข้อมูล 2 ตาราง โดยการเปรียบเทียบข้อมูลจาก 2 ตารางที่มีทิศทางการสื่อสารตรงกันข้ามกัน เช่น เวลาการสื่อสารข้อมูลไปกลับ สามารถพิจารณาได้จากตารางการสื่อสารข้อมูลขาออก(Forward Table) กับตารางการสื่อสารข้อมูลขาเข้า (Reverse Table)

3.4.2 ขั้นตอนการตรวจวัดขนาดและเวลาระหว่างการมา

สำหรับขั้นตอนการวิเคราะห์ความยาว เวลาระหว่างการมาของแพ็กเก็ตในงานวิจัยนี้ได้ทำการเปรียบเทียบกับเครื่องมือที่สร้างจาก Corallib Library โดยใช้ขบวนการโปรแกรมคอมพิวเตอร์ด้วยภาษา C เช่นเดียวกับการทำงานด้วย Libpcap Library ให้ทำการตรวจจับขนาดแพ็กเก็ตและเวลาระหว่างการมาในทุกโปรโตคอล ที่ผ่าน NIC แล้วจึงนำค่าความยาวและค่าที่คำนวณเวลาระหว่างมาของแพ็กเก็ตปัจจุบันกับแพ็กเก็ตก่อนหน้าไปจัดเก็บบนแฟ้มข้อมูลแบบเรียงลำดับ จากนั้นจึงนำข้อมูลที่ได้จากการเก็บด้วยขบวนการจาก Corallib Library ไปเปรียบเทียบกับการตรวจวัดจากเครื่องมือวิจัยที่สร้างจาก Libpcap Library ด้วยขบวนการเดียวกัน ผลของการตรวจวัดได้เปรียบเทียบความถูกต้องระหว่างทั้งสองเครื่องมือแบบแพ็กเก็ตต่อแพ็กเก็ต โดยใช้หลักสถิติพื้นฐานได้แก่ ผลต่างที่ตรวจวัดได้จากเครื่องมือทั้ง 2 ชนิด(Difference) อัตราเฉลี่ย(Average) และความเชื่อมั่น(Confidence Interval) เป็นหลัก

3.4.3 ขั้นตอนการตรวจวัดเวลาการสื่อสารข้อมูลไปกลับ

ทำการวิเคราะห์เฉพาะแพ็กเก็ตที่โปรโตคอลมีชนิดเป็น TCP และ ICMP(Echo)นี้เท่านั้น โดยขอนิยามข้อมูลที่ได้รับการพิจารณาเหล่านี้ว่า Considered Packet แต่ละแพ็กเก็ตจะถูกนำไปเปรียบเทียบหาข้อมูลที่เป็นคู่การสื่อสารกันจากตารางข้อมูลที่มีทิศทางการสื่อสารตรงข้ามกันและอยู่ในฝั่งเครือข่ายเดียวกัน และขอนิยามเรียกแพ็กเก็ตที่เป็นคู่การสื่อสารกันนี้ว่า Corresponding Packet โดยมีขั้นตอนดังภาพที่ 29

```
For i = 1 to Number of Packet in Response Table {
    if (Res_i is Considered Packet) {
    t_i = TimeStamp of Res_i;
    For j = 1 to Number of Packet in Request Table {
        t_j = TimeStamp of Req_i
        if (Req_j Corresponding to Res_i)
        Compute γ_n ; γ_n = (t_res - t_req) = t_y - t_j
    }
}
```

ภาพที่ 29 ขั้นตอนการตรวจวัดเวลาการสื่อสารข้อมูลไปกลับ

โดยที่

Res_i หมายถึง แพ็กเก็ตฝ่ายตอบที่เป็นคู่การสื่อสารกับแพ็กเก็ตฝ่ายเรียก

Req_i หมายถึง แพ็กเก็ตฝ่ายเรียกที่เป็นคู่การสื่อสารกับแพ็กเก็ตฝ่ายตอบ

Response Table หมายถึง ตารางข้อมูลขาออก(Forward Table)หรือขาเข้า(Reverse Table)ก็ได้ที่พิจารณาเฉพาะแพ็กเก็ตทำหน้าที่ตามโปรโตคอลนั้น ๆ ว่าเป็นฝ่ายตอบ

Request Table หมายถึง ตารางข้อมูลขาออก(Forward Table)หรือขาเข้า(Reverse Table)ก็ได้ที่พิจารณาเฉพาะแพ็กเก็ตทำหน้าที่ตามโปรโตคอลนั้น ๆ ว่าเป็นฝ่ายเรียก

จากภาพที่ 35 การตรวจวัดใช้วิธีเปรียบเทียบโดยนำแพ็กเก็ตของฝ่ายตอบ(Res_i)ไปค้นหาแพ็กเก็ตของฝ่ายเรียก(Req_i) ในตารางข้อมูลที่มีทิศทางการสื่อสารตรงข้ามกัน(Forward or Reverse Table) ซึ่งแพ็กเก็ตของฝ่ายตอบอาจอยู่ในรูปของตารางข้อมูลขาออก หรือตารางข้อมูลขาเข้าก็ได้ ในที่นี้พิจารณาแพ็กเก็ตฝ่ายตอบจากตารางข้อมูลที่มีทิศทางการสื่อสารเป็นขาเข้าและนำไปพิจารณาแพ็กเก็ตฝ่ายเรียกในตารางที่มีทิศทางการสื่อสารเป็นขาออก โดยจะนำแพ็กเก็ตที่เป็นคู่สื่อสารกันไปคำนวณหาเวลาของการสื่อสารข้อมูลไปกลับจากสมการ

$$\gamma_n = t_y - t_j \qquad (1)$$

โดยที่ t_y หมายถึง เวลาของแพ็กเก็ตฝ่ายตอบ

t_j หมายถึง เวลาของแพ็กเก็ตฝ่ายเรียก

การตรวจวัดเวลาไปกลับของคู่แพ็กเก็ต ต้องแยกพิจารณากลุ่มการสื่อสารของข้อมูลที่ต่างชนิดกันดังนี้

3.4.3.1 การตรวจวัดเวลาการสื่อสารไปกลับของข้อมูล ICMP

ICMP โปรโตคอลโดยทั่วไปถูกใช้เพื่อรายงานสถานภาพของเครือข่าย 2 ลักษณะคือ

- แจ้งความผิดพลาดที่เกิดขึ้นจากการสื่อสารข้อมูลบนเครือข่าย เช่น ข้อมูลไม่ถึงปลายทาง(Destination Unreachable), ต้นทางระงับการทำงาน(Source Quench), การรอเกินขอบเขตของเวลา(Time Exceeded) และ ปัญหาของพามิเตอร์ (Parameter Problem)
- สอบถามสถานะ(Query Message) เช่น การตอบสนอง(Echo Request and Reply), ความต้องการด้านข้อมูล(Information Request and Replay) และ การกำกับเวลา (Timestamp Request and Replay)

ในส่วนของงานวิจัยจะพิจารณาเฉพาะ ICMP แพ็กเก็ตที่ทำหน้าที่ในส่วนของ การตอบสนองเท่านั้น(Echo Request and Reply) ขั้นตอนของการสื่อสารที่เกิดขึ้นกรณีนี้คือ คอมพิวเตอร์ต้นทางส่ง ICMP แพ็กเก็ต เรียกว่า แพ็กเก็ตฝ่ายเรียก(Echo Request) โดยมีโครงสร้างที่สำคัญ คือ ชนิด(Type) เป็น 0x00(ฐาน 16) และสุ่มหมายเลขลำดับของ แพ็กเก็ต (Identifier) ไปยังคอมพิวเตอร์ปลายทาง และเมื่อคอมพิวเตอร์ปลายทางได้รับจะต้องตอบกลับด้วย ICMP แพ็กเก็ตที่มี ชนิด เป็น 0x08(ฐาน 16) และมีลำดับเป็นหมายเลขเดียวกับแพ็กเก็ตฝ่ายเรียกไปยังคอมพิวเตอร์ต้นทาง

ด้วยลักษณะการทำงานของ ICMP โปรโตคอลชนิดนี้ ในงานวิจัยจึงกำหนดวิธีการเปรียบเทียบเพื่อหาคู่แพ็กเก็ตสื่อสารและเวลาไปกลับของแพ็กเก็ตชนิดนี้ด้วยขบวนการดังแสดงได้ในภาพที่ 30

Considered Packet = IP Packet(Res_i.ip_proto==ICMP(0x01) &&
Res_i.icmp_type==Echo Reply(0x00))
Corresponding Packet = (Req_j.ip_proto == Res_i.ip_proto &&
Req_j.ip_dst == Res_i.ip_src &&
Req_j.ip_src == Res_i.ip_dst &&
Req_j.icmp_type == Echo(0x08) &&
Req_j.icmp_id == Res_i.icmp_id &&

ภาพที่ 30 การเปรียบเทียบคู่แพ็กเก็ตของการสื่อสารข้อมูลชนิด ICMP

3.4.3.2 การตรวจวัดเวลาการสื่อสารไปกลับของข้อมูล TCP

โปรโตคอล TCP เป็นวิธีการรับส่งข้อมูลที่มีความน่าเชื่อถือในการแลกเปลี่ยนข้อมูล ขั้นตอนของการสื่อสารข้อมูลจะมีขบวนการตรวจสอบความถูกต้องของข้อมูลที่ได้รับจากฝั่งส่ง โดยทั่วไปการสื่อสารข้อมูลชนิดนี้สามารถแบ่งได้เป็น 3 ขั้นตอนหลัก คือ สร้างการเชื่อมต่อเพื่อรับส่งข้อมูล(Connection) รับและส่งข้อมูล(Data Transfer) และ ปิดการเชื่อมต่อ(Closed Connection) แต่ละขั้นตอนจะมีวิธีการสื่อสารข้อมูลที่แตกต่างกันดังนี้

ขั้นตอนเชื่อมต่อข้อมูล(Connection) หมายถึง แอพพลิเคชั่น(Application)ฝั่งที่ต้องการสร้างการเชื่อมต่อนิยามเรียกว่า Client Host ไปยังอีกฝั่งเรียกว่า Server Host โดยมี 3 ขบวนการที่สำคัญดังนี้

- Client host ส่ง TCP segment ไปยัง server host โดยที่ segment นี้จะยังไม่มีข้อมูลใด ๆ แต่จะทำการกำหนด SYN bit ให้ TCP Flags field ใน TCP Header ของ segment ให้มีค่าเป็น 1 พร้อมทั้งตั้งค่าหมายเลขลำดับการรับส่งข้อมูล(Initial Sequence Number) ขึ้นมา และกำกับไปใน

segment ด้วย แล้วจึงส่ง segment นี้ ไปยัง Server Host และ เนื่องจาก segment นี้มีการกำหนด SYN bit เป็น 1 จึงนิยามเรียก segment นี้ว่า SYN segment

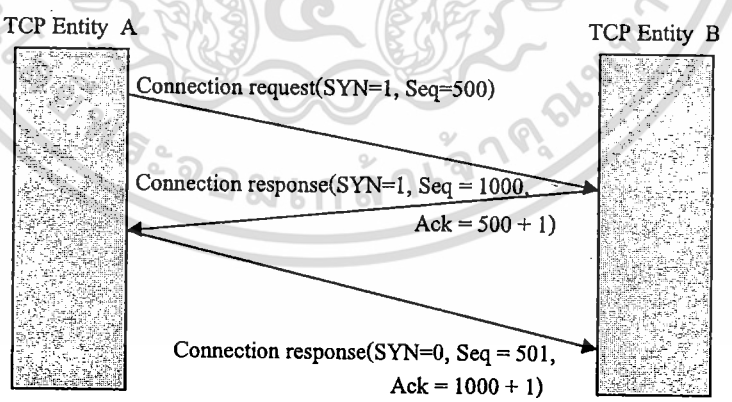
- เมื่อ Server Host ได้รับ SYN segment ก็จะส่ง SYNACK segment กลับไปให้ Client Host โดยที่ SYNACK segment นี้ก็ยังไม่มีข้อมูลใด ๆ แต่จะมีลักษณะสำคัญ 3 อย่าง คือ

1. SYN bit กำหนดค่าเป็น 1
2. ค่า TCP acknowledgment number field ใน TCP Header จะกำหนดโดยหมายเลขลำดับเดิมจากฝั่ง Client Host เพิ่มขึ้นอีก 1
3. ตั้งค่าหมายเลขลำดับในฝั่ง Server Host ขึ้นใหม่และกำกับลงใน sequence number field

- หลังจาก Client Host ได้รับ SYNACK segment แล้วก็จะส่ง segment สุดท้ายไปให้ Server Host อีก โดยกำหนดค่าต่าง ๆ ดังนี้

1. กำหนด SYN bit เป็น 0
2. กำหนดค่า sequence number field ด้วยหมายเลขลำดับเดิมที่เพิ่มขึ้นอีก 1 โดยฝั่ง Server Host
3. กำหนด TCP acknowledgement number field เป็นหมายเลขลำดับของฝั่ง Server Host เพิ่มขึ้นอีก 1

หลังจาก 3 ขั้นตอนหรือนิยามเรียก ขบวนการนี้ว่า Three-Way Handshake เสร็จแล้วก็จะหมายถึง การสร้าง TCP Connection เป็นผลสำเร็จ ดังแสดงได้ในภาพที่ 31

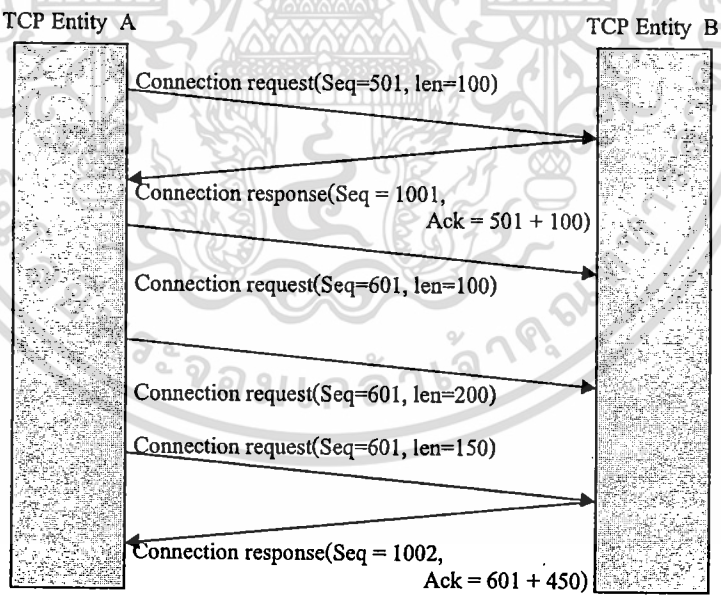


ภาพที่ 31 ขั้นตอนการเชื่อมต่อเพื่อรับส่งข้อมูลชนิด TCP

ขั้นตอนการรับส่งข้อมูล(Data Transfer) หลังจากที่ขบวนเชื่อมต่อเป็นผลสำเร็จ โปรโตคอล TCP ก็จะเริ่มทำการรับส่งข้อมูลโดยมีขั้นตอนที่สำคัญคือ

- ฝั่ง Client Host ทำการส่งข้อมูลโดยกำหนด SYN bit เป็น 0 และกำกับ TCP sequence number ต่อจากขบวนการเชื่อมต่อไปยัง Server Host ขบวนการนี้อาจเกิดขึ้นได้มากกว่า 1 segment หรืออาจทำจนครบจำนวนข้อมูลที่ต้องการส่ง ขึ้นอยู่กับการกำหนด window size
- ฝั่ง Server Host เมื่อได้รับข้อมูลก็จะตอบกลับโดยกำหนด SYN bit เป็น 0 และกำหนด TCP sequence number เดิมต่อจากขบวนการเชื่อมต่อ ส่วน TCP acknowledgment number จะถูกกำหนดจาก TCP sequence number ของฝั่ง Client Host บวกเพิ่มด้วยความยาวของข้อมูลที่ได้รับทั้งหมดฝั่ง Client Host

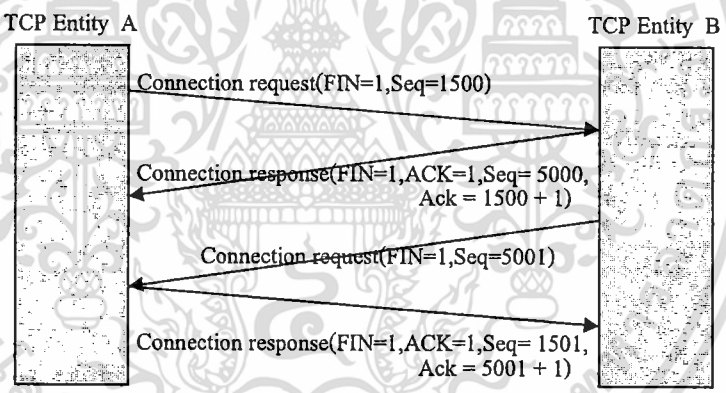
ดังนั้นด้วยขบวนการรับส่งข้อมูลแบบนี้จึงไม่สามารถคาดการณ์ได้ว่าข้อมูลชุดหนึ่ง ๆ จากฝั่ง Client Host ที่ส่งให้กับ Server Host จะเกิดการตอบกลับจากฝั่ง Server Host กี่ครั้ง เพราะขึ้นอยู่กับ Window Size ของฝั่ง Server Host ที่คอยกำกับการได้รับข้อมูลจากฝั่ง Client Host ดังแสดงได้ในภาพที่ 32



ภาพที่ 32 ขั้นตอนการรับส่งข้อมูลชนิด TCP

ดังนั้นด้วยลักษณะการทำงานแบบนี้ การเปรียบเทียบหาคู่แพ็กเก็ตสื่อสารข้อมูลอาจกระทำได้แต่เทคนิคของเวลาที่ใช้ไปกับการรอแพ็กเก็ตที่เป็นคู่สื่อสารข้อมูลกันนั้นอาจทำให้ต้องเสียทรัพยากรไปกับการจัดเก็บข้อมูลเป็นปริมาณมาก ในงานวิจัยนี้จึงขอละเว้นจากการตรวจสอบคู่แพ็กเก็ตนี้ไปก่อน

ขั้นตอนการปิดการเชื่อมต่อ(Closed Connection) เมื่อการรับส่งข้อมูลเสร็จสิ้นแล้ว Client Host ก็จะทำการปิดการเชื่อมต่อด้วยการส่ง segment ที่กำหนด FIN bit เป็น 1 ไปให้กับทาง Server Host จากนั้น Server Host ก็จะส่ง ACK bit เป็น 1 ไปให้กับทาง Client Host และจะรอระยะเวลาหนึ่ง เพื่อให้แน่ใจว่าทาง Client Host ได้รับ acknowledgement segment แล้ว เมื่อ Server Host แน่ใจว่าไม่มีการส่งใหม่(Retransmission) ก็จะส่ง FIN segment กลับไปให้ทาง Client Host เมื่อทาง Client Host ได้รับ FIN segment แล้ว ก็จะทำการส่ง acknowledgement segment ไปให้ทาง Server Host แล้วจะรอระยะเวลาหนึ่งเพื่อให้แน่ใจว่าทาง Server Host ได้รับแล้ว เมื่อครบเวลาที่รอ Client Host ก็จะปลดปล่อยทรัพยากรต่าง ๆ แล้วปิด Connection อย่างสมบูรณ์ ส่วนทางด้าน Server Host เมื่อได้รับ acknowledgement segment ก็จะทำการปิด Connection เช่นเดียวกันดังแสดงได้ในภาพที่ 33



ภาพที่ 33 ขั้นตอนปิดการเชื่อมต่อในการรับส่งข้อมูลชนิด TCP

จากขบวนการสื่อสารข้อมูลด้วย TCP โปรโตคอล ในงานวิจัยนี้จึงกำหนดให้มีการวิเคราะห์เวลาการสื่อสารไปกลับเฉพาะในช่วงเวลาของการเปิด และปิดการเชื่อมต่อเท่านั้น โดยมีเงื่อนไขของการวิเคราะห์เวลาระหว่างเปิดการเชื่อมต่อดังภาพที่ 34

```
Considered Packet = (Res_i.ip_proto==0x06 &&
                     Res_i.th_flags==0x12)
Corresponding Packet = (Req_j.ip_p == Res_i.ip_p &&
                        Req_j.ip_dst == Res_i.ip_src &&
                        Req_j.ip_src == Res_i.ip_dst &&
                        Req_j.tcp_sport == Res_i.tcp_dport &&
                        Req_j.tcp_dport == Res_i.tcp_sport &&
                        Req_j.tcp_flags == 0x02 &&
                        Req_j.tcp_seq == Res_i.tcp_seq - 1)
```

ภาพที่ 34 การเปรียบเทียบคู่แพ็กเก็ตชนิด TCP ระหว่างสร้างการเชื่อมต่อ

กลุ่มข้อมูลที่พิจารณาคือแพ็กเก็ตของฝ่ายตอบ(Res_i) ที่มี Protocol filed ใน TCP Header เท่ากับ 0x06 และส่งสัญญาณข้อมูลชนิด SYN/ACK(TCP flags field มีค่าเป็น 0x12)ให้กับฝ่ายเรียก(Req_j)ที่ส่งด้วยสัญญาณชนิด SYN(TCP flags field มีค่าเป็น 0x02) ในการเปรียบเทียบคู่ของแพ็กเก็ตจะพิจารณาจาก Protocol, IP Source, IP Destination, Destination Port, Source Port field ที่เท่ากันและมี Sequence Number field ของแพ็กเก็ตในฝ่ายเรียกมีค่าเท่ากับ Sequence Number field ลบด้วย 1 ของแพ็กเก็ตฝ่ายตอบ ส่วนเงื่อนไขของการวิเคราะห์เวลาระหว่างปิดการเชื่อมต่อสามารถแสดงดังภาพที่ 35

```
Considered Packet = (Res_i.ip_proto==0x06 &&
                     Res_i.th_flags==0x10)
Corresponding Packet = (Req_j.ip_p == Res_i.ip_p &&
                        Req_j.ip_dst == Res_i.ip_src &&
                        Req_j.ip_src == Res_i.ip_dst &&
                        Req_j.tcp_sport == Res_i.tcp_dport &&
                        Req_j.tcp_dport == Res_i.tcp_sport &&
                        Req_j.tcp_flags == 0x11 &&
                        Req_j.tcp_seq == Res_i.tcp_seq - 1)
```

ภาพที่ 35 การเปรียบเทียบคู่แพ็กเก็ตชนิด TCP ระหว่างปิดการเชื่อมต่อ

กลุ่มข้อมูลที่พิจารณาคือแพ็กเก็ตของฝ่ายตอบ(Res_i) ที่มี Protocol filed ใน TCP Header เท่ากับ 0x06 และส่งสัญญาณข้อมูลชนิด ACK(TCP flags field มีค่าเป็น 0x10)ให้กับฝ่ายเรียก(Req_j)ที่ส่งด้วย

สัญญาณชนิด FIN/ACK(TCP flags field มีค่าเป็น 0x11) ในการเปรียบเทียบคู่ของแพ็กเก็ตจะพิจารณาจาก Protocol, IP Source, IP Destination, Destination Port, Source Port field ที่เท่ากันและมี Sequence Number field ของแพ็กเก็ตในฝ่ายเรียกมีค่าเท่ากับ Sequence Number field ลบด้วย 1 ของแพ็กเก็ตฝ่ายตอบเช่นเดียวกับกรณีเปิดการเชื่อมต่อ

3.4.4 ขั้นตอนการตรวจวัดความล่าช้าและความสูญเสีย

การวิเคราะห์ความล่าช้าในงานวิจัยกำหนดให้วิเคราะห์เฉพาะข้อมูลชนิด TCP, UDP และ ICMP โดยขอนิยามข้อมูลที่ได้รับการพิจารณาเหล่านี้ว่า Considered Packet แต่ละแพ็กเก็ตจะถูกนำไปเปรียบเทียบหาข้อมูลเดียวกันที่ผ่านเข้าและออกจากตัวระบบในตารางข้อมูลที่มีทิศทางการสื่อสารทางเดียวกันแต่อยู่ในฝั่งเครือข่ายตรงข้ามกัน และขอนิยามเรียกแพ็กเก็ตที่เป็นข้อมูลเดียวกันนี้ว่า Ownership Packet โดยมีขั้นตอนดังแสดงได้ในภาพที่ 36

```
Window = 100;
/* Record Range for Searching Ownership */
j = 1;
/* Record Number of  Packet in Exit Table */
Max = Maximum Number of Packet in Exit Table
For i = 1 to Number of Packet in Entrance Table {
  if (Ent_i is Considered Packet) {
    t_i = TimeStamp of Ent_i;
    if (j - Window / 2 ≥ 0)
      j = j – Window / 2;
    else j = MAX + (j – Window / 2);
      old_position_of_j = j
      match = false;
      While (true) {
        t_j = TimeStamp of Ext_j
```

ต่อ

```
        if (Ext_j Ownership to Ent_i) {
          match = true;
          break;
        }
        if (++j > MAX) j = 1;
        if (Loop Count > Window) break;
      } /* end of while */
      if (match = true)
        Compute τ_m ; τ_m = (t_ext - t_ent) = t_x – t_y
      else {
        Save Packet Information to Loss Table;
        j = old_position_of_j;
      }
    }
  } /* end of for */
```

ภาพที่ 36 ขั้นตอนการวิเคราะห์ความล่าช้าและความสูญเสีย

จากภาพการตรวจวัดจะใช้วิธีเปรียบเทียบโดยนำแพ็กเก็ตฝั่งขาเข้า(Entrance Table)ไปค้นหาแพ็กเก็ตเดียวกันในฝั่งขาออก(Exit Table) บนตารางข้อมูลที่มีทิศทางการสื่อสารทางเดียวกัน เช่น แพ็ก

เก็ตใน Internal Forward Table เปรียบเทียบแพ็กเก็ตใน External Forward Table หรือแพ็กเก็ตใน Internal Reverse Table เปรียบเทียบกับแพ็กเก็ตใน External Reverse Table โดยจะนำแพ็กเก็ตที่ตรวจพบว่าเป็นแพ็กเก็ตเดียวกันจากฝั่งขาเข้าและออกไปคำนวณหาความล่าช้าของการสื่อสารข้อมูลผ่านตัวระบบจากสมการ

$$\tau_m = t_x - t_y \tag{2}$$

โดยที่ t_x หมายถึง เวลาของแพ็กเก็ตฝั่งขาเข้า

t_y หมายถึง เวลาของแพ็กเก็ตฝั่งขาออก

แพ็กเก็ตที่ตรวจไม่พบในฝั่งขาออกตามขอบเขตที่กำหนด(Record Range for Searching) จะถือเป็นความสูญเสียจากการสื่อสารข้อมูลผ่านตัวระบบ และจะนำไปจัดเก็บไว้ในตารางความสูญเสียข้อมูล (Loss Table) ที่มีอยู่ทั้งทิศทางขาออกและขาเข้า การตรวจวัดความล่าช้าและความสูญเสียมีเงื่อนไขที่ใช้พิจารณากลุ่มการสื่อสารข้อมูลทุกชนิดดังแสดงได้ในภาพที่ 37

```
Considered=IP Packet( Ent_i.ip_proto==0x01 ||
                      Ent_i.ip_proto==0x06 ||
                      Ent_i.ip_proto==0x11 )
Ownership=(Ext_j.ip_proto == Ent_i.ip_proto &&
           Ext_j.ip_id == Ent_i.ip_id &&
           Ext_j.ip_off == Ent_i.ip_off &&
           Ext_j.ip_dst == Ent_i.ip_dst &&
           Ext_j.ip_src == Ent_i.ip_src &&
           Ext_j.ip_len == Ent_i.ip_len &&
           Ext_j.proto_header == Ent_j.proto_header)
```

ภาพที่ 37 การเปรียบเทียบหาคู่แพ็กเก็ตในขั้นตอนวิเคราะห์ความล่าช้าและความสูญเสีย

จากภาพจะพิจารณาเฉพาะแพ็กเก็ตที่มี Protocol เป็น ICMP(protocol field มีค่าเป็น 0x01), TCP(protocol field มีค่าเป็น 0x06) หรือ UDP(protocol มีค่าเป็น 0x11) ไปเปรียบเทียบหาคู่ แพ็กเก็ตเดียวกันด้วยเงื่อนไขที่มี Protocol, Identifier, IP Offset, IP Destination, IP Source, IP Total Length และ

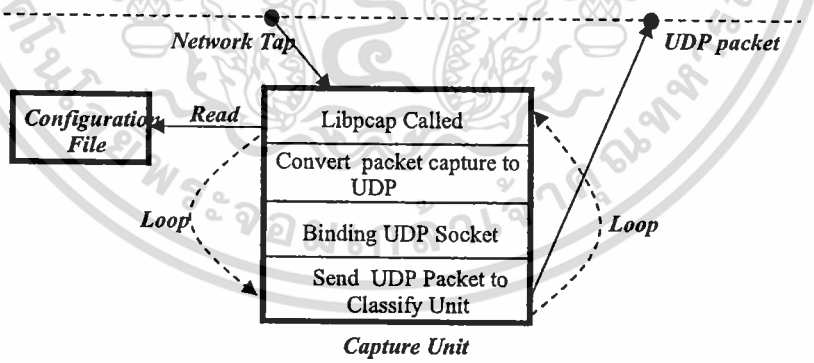
Protocol Header ในแต่ละ field ของข้อมูลแต่ละชนิดที่ตรงกัน(ICMP Header, TCP Header และ UDP Header)

3.5 โปรแกรมการวิเคราะห์ข้อมูลเครือข่าย

โปรแกรมการวิเคราะห์ข้อมูลได้ออกแบบให้มีลักษณะการทำงานเป็นแบบ Multi-Processing ดังนั้นหน่วยตรวจจับและหน่วยแยกชนิดข้อมูลจึงจำเป็นต้องมีข้อตกลงในการทำงานร่วมกัน นอกจากนี้ด้วยลักษณะการตรวจจับข้อมูลที่มีความต่อเนื่องและไม่มีข้อจำกัดด้านเวลาในการวิเคราะห์เครื่องมือวิจัยจึงต้องออกแบบให้บริหารจัดการหน่วยความจำอย่างมีประสิทธิภาพในการจัดเก็บข้อมูลและการนำข้อมูลไปใช้วิเคราะห์ผลลัพธ์ ซึ่งจะถูกเขียนขึ้นหลังจากที่โปรแกรมได้รับสัญญาณการหยุดทำงาน(Interrupt Signature)จากผู้ใช้งาน โดยโปรแกรมได้ใช้เทคนิคต่าง ๆ ในการทำงานที่มีลักษณะดังนี้

3.5.1 การออกแบบข้อกำหนดในการตรวจจับข้อมูล

หน่วยตรวจจับสามารถส่งข้อมูลไปยังหน่วยแยกชนิดได้ด้วยการส่งผ่านไปบนเครือข่าย ด้วยการจัดรูปแบบแพ็กเก็ตใหม่ให้มีชนิดเป็น UDP และทำการสร้างช่องทางการสื่อสารร่วมกันระหว่างสองหน่วยให้สามารถแลกเปลี่ยนข้อมูลกันได้ด้วยการเปิดพอร์ตสื่อสารช่องทางเดียวกัน(Binding UDP Socket) การตรวจจับข้อมูลจะถูกทำงานโดย Libpcap Library ซึ่งมีหน้าที่ตรวจจับและกำกับเวลาการสื่อสารข้อมูลของแพ็กเก็ต ณ ขณะที่ตรวจจับได้ ดังแสดงได้ในภาพที่ 38



ภาพที่ 38 ขบวนการตรวจจับข้อมูล

จากภาพที่ 38 ในส่วนของ แฟ้มข้อมูลติดตั้ง(Configuration File) จะเป็นส่วนที่สำคัญที่ทุกหน่วยตรวจจับข้อมูลใช้เป็นข้อตกลงร่วมกันในการรับส่งข้อมูลไปยังหน่วยแยกชนิด โดยออกแบบให้มีข้อกำหนดที่สำคัญดังนี้

DEVICE หมายถึง ชื่อที่ FreeBSD ใช้เชื่อมต่อกับอุปกรณ์ Network Card เช่น Fxp0, wb0 และอื่น ๆ

SENDTOHOST หมายถึง หมายเลขไอพีของเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นหน่วยแยกชนิดข้อมูล

PORT หมายถึง หมายเลขพอร์ตชนิด UDP ที่ใช้ในการรับส่งข้อมูลระหว่างคอมพิวเตอร์ที่ทำหน้าที่ตรวจจับและแยกชนิดข้อมูล

FORMATTYPE,CAPTUREID,HOSTID,CLASSFYID,PACKETINFO หมายถึง ค่าที่ใช้กำหนดคุณสมบัติไว้ตาม Capture Header

STOREDISK หมายถึง สถานะที่กำหนดให้หน่วยตรวจจับบันทึกข้อมูลไว้บนดิสก์

FILENAME หมายถึง ชื่อแฟ้มที่ต้องการจัดเก็บข้อมูล

ดังแสดงตัวอย่างได้ในภาพที่ 39

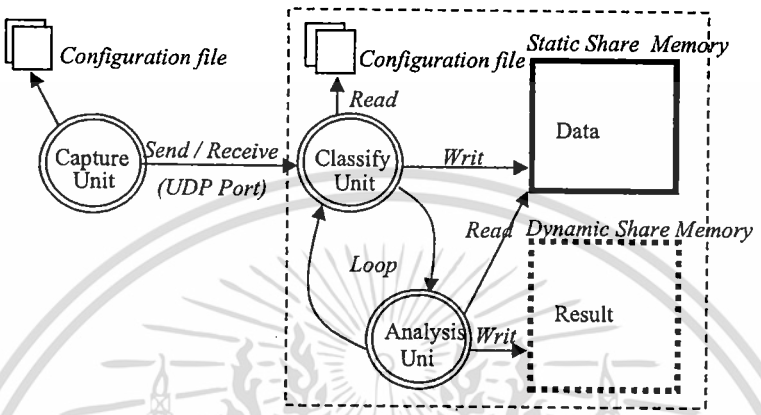
```
# ethernet device such as fpx0, wb0, etc...
DEVICE          wb0
# ip host(Classifying and Analysis Data) for sent packet
SENDTOHOST      127.0.0.1
# udp port communication, default is 3056
PORT            3056
# Format data type 0 - binary (Reserve other number for future) , default is 0
FORMATTYPE      0
# Identifier of internal or external network, input value between 0 - 9 , default is 0
CAPTUREID       0
# Identifier data of epch hosts
HOSTID          0
# packet classifying 0-Not 1-Classified(Reserve for future), default is 0
CLASSIFYID      0
# Packet information 0x0-None(Individual Time Stamp)
#                    0x1-Combine Network Layer
#                    0x3-Combine Transport Layer
#                    0xF-Reserve
#                    default is 3
PACKETINFO      3
# save packets to disk ,  enable-save , disable-none
STOREDISK       disable
# file name, default is !disk0.dat!
FILENAME        disk0.dat
```

ภาพที่ 39 แฟ้มติดตั้งการใช้งานหน่วยตรวจจับข้อมูล

3.5.2 การออกแบบข้อกำหนดในการแยกชนิดข้อมูล

หน่วยแยกชนิดเป็นหน่วยที่มีหน้าที่สำคัญคือรับแพ็กเก็ตจากหน่วยตรวจจับข้อมูลและทำการจัดเก็บแพ็กเก็ตลงตารางข้อมูลแต่ละชนิดให้ถูกต้อง โดยตารางข้อมูลจะถูกสร้างขึ้นในรูปของหน่วยความจำร่วมแบบคงที่(Static Share Memory)ซึ่งหน่วยวิเคราะห์ข้อมูลสามารถเปิดอ่านและนำไปใช้สร้างตารางผลลัพธ์(Result Table)ต่อไปได้ ดังนั้นการทำงานของหน่วยตรวจจับ ข้อมูล หน่วยแยกชนิด และหน่วยวิเคราะห์ข้อมูลจะต้องปฏิบัติงานไปพร้อม ๆ กันโดยอาศัยหลักการเขียน

โปรแกรมแบบ Multi-Processing ให้ทั้งสามหน่วยทำงานเป็นวัฏจักร(Loop)ที่ไม่รู้จบจนกว่าจะได้รับสัญญาณหยุดการทำงาน(Interrupt Signal)จากผู้ใช้ ลักษณะการทำงานสามารถแสดงได้ในภาพที่ 40



ภาพที่ 40 ขั้นตอนการทำงานของหน่วยแยกชนิด

เพื่อให้การทำงานของหน่วยแยกชนิดสามารถกำหนดรูปแบบข้อมูลให้กับหน่วยวิเคราะห์ข้อมูลได้อย่างถูกต้อง ดังนั้นหน่วยแยกชนิดจึงต้องออกแบบให้มีแฟ้มติดตั้งข้อกำหนดก่อนการใช้งาน โดยมีพารามิเตอร์ที่สำคัญดังนี้

PORTRCVE หมายถึง หมายเลขพอร์ตการสื่อสารข้อมูลที่ใช้รับส่งข้อมูลระหว่างหน่วยตรวจจับและแยกชนิดข้อมูล

MEDIAREAD หมายถึง อุปกรณ์ที่ต้องการให้หน่วยแยกชนิดรับส่งแพ็กเก็ตมาตรวจสอบ ซึ่งกำหนดให้มีได้ 2 อุปกรณ์ คือ ผ่านทางเครือข่าย หรือ อ่านจากแฟ้มข้อมูล

INDISK หมายถึง ชื่อแฟ้มจัดเก็บข้อมูลจากฝั่งเครือข่ายภายในที่ถูกบันทึกได้จากหน่วยตรวจจับข้อมูล จะถูกใช้ก็ต่อเมื่อไม่มีการกำหนดให้ MEDIAREAD เป็น Network

EXDISK หมายถึง ชื่อแฟ้มจัดเก็บข้อมูลจากฝั่งเครือข่ายภายนอกที่ถูกบันทึกได้จากหน่วยตรวจจับข้อมูล จะถูกใช้ก็ต่อเมื่อไม่มีการกำหนดให้ MEDIAREAD เป็น Network

INTERNALNET หมายถึง ไอพีที่ใช้อยู่บนฝั่งเครือข่ายภายใน มีรูปแบบการกำหนดค่าคือ IP Address/Mask ตัวอย่าง เช่น 192.168.10.0/25 หมายถึงกลุ่มของไอพีตั้งแต่ 192.168.10.0 ถึง 192.168.10.127

SUTIP หมายถึง ไอพีที่ใช้งานในกลุ่มของตัวระบบ

INTERNALCAPID หมายถึง หมายเลขของ Capture Identifier ที่อยู่ฝั่งเครือข่ายภายใน

EXTERNALCAPID หมายถึง หมายเลขของ Capture Identifier ที่อยู่ฝั่งเครือข่ายภายนอก

PACKETSIZE หมายถึง สถานะภาพการวิเคราะห์ความยาวแพ็กเก็ต

INTERARRIVAL หมายถึง สถานะภาพการวิเคราะห์เวลาระหว่างการมา

DELAY หมายถึง สถานะภาพการวิเคราะห์ความล่าช้าของตัวระบบ

ROUNDTRIPTIME หมายถึง สถานะภาพการวิเคราะห์เวลาการสื่อสารไปและกลับ

LOSS หมายถึง สถานะภาพการวิเคราะห์ความสูญเสียข้อมูล

FLOWED หมายถึง สถานะภาพการวิเคราะห์ข้อมูลโฟลว

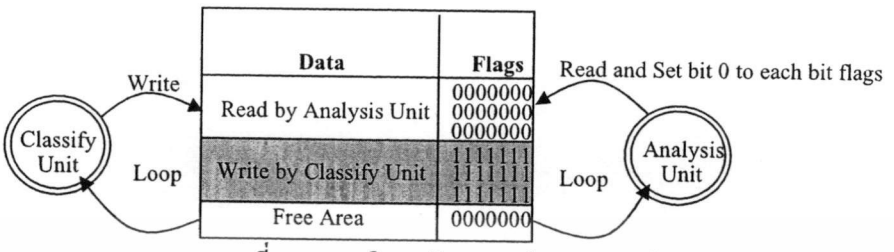
FLWEXP หมายถึง เวลาที่ใช้ตรวจสอบการหมดอายุของข้อมูลโฟลวบนเครือข่าย ปกติกำหนดไว้ที่ 64 วินาที ดังแสดงได้ในภาพที่ 41

```
#Port Receive from Capture Unit
PORTRCVE        3056
#Media for Read Packet Data
MEDIAREAD       network
#Internal disk file name, default is "disk0.dat"
INDISK          disk0.dat
#External disk file name, default is "disk1.dat"
EXDISK          disk1.dat
#Internal Network IP
INTERNALNET     192.168.10.0/25,192.168.10.128/25
#System Under Test IP address
SUTIP           192.168.10.254,192.168.20.254
#Capture ID that live in Internal Network
INTERNALCAPID 0
#Capture ID that live in External Network
EXTERNALCAPID 1
#enable or disable Analysis data of result distribution table
PACKETSIZE      disable
INTERARRIVAL    disable
DELAY           enable
ROUNDTRIPTIME   disable
LOSS            disable
#Enable Analysis data flowed
FLOWED          disable
FLWEXP          64
```

ภาพที่ 41 แฟ้มติดตั้งข้อกำหนดการแยกชนิดข้อมูล

3.5.3 การบริหารและจัดการหน่วยความจำร่วม

เพื่อให้การวิเคราะห์ข้อมูลใช้หน่วยความจำอย่างมีประสิทธิภาพ หน่วยแยกชนิดได้ออกแบบให้ข้อมูลที่ถูกบันทึกบนหน่วยความจำจนครบทั้งตารางแล้วสามารถวนกลับมาใช้ตำแหน่งของหน่วยความจำเริ่มต้นเดิมได้อีกครั้ง ถ้าระเบียนดังกล่าวได้ถูกนำไปใช้วิเคราะห์ข้อมูลจนครบตามความต้องการของหน่วยวิเคราะห์แล้ว ดังแสดงได้ในภาพที่ 42



ภาพที่ 42 การบริหารจัดการหน่วยความจำร่วม

การกลับไปใช้หน่วยความจำเริ่มต้นใหม่จะต้องมีความสัมพันธ์กับหน่วยวิเคราะห์ข้อมูล เนื่องจากช่วงเวลาการมาของข้อมูลอาจเร็วกว่ากระบวนการสร้างผลลัพธ์โดยหน่วยวิเคราะห์ ดังนั้น หน่วยแยกชนิดจึงต้องตรวจสอบก่อนการบันทึกข้อมูลอยู่เสมอว่า ข้อมูลที่จะเขียนทับนั้นถูกอ่านหรือนำไปใช้สร้างผลลัพธ์โดยหน่วยวิเคราะห์มูลแล้วหรือไม่ เพื่อแก้ปัญหานี้จึงต้องเพิ่ม Flags ที่มีขนาดข้อมูลเท่ากับ 8 bits(ตามที่แสดงไว้ในตารางที่ 1) มาใช้บอกสถานะของการนำไปใช้ใหม่ โดยแต่ละบิตสามารถกำหนดสถานะได้ 2 ค่า คือ 1 ใช้แทนค่า ระเบียนที่ถูกเขียนโดยหน่วยแยกชนิดหรือยังไม่มีการอ่านไปใช้จากหน่วยวิเคราะห์ และ 0 ใช้แทนความหมาย ระเบียนที่ถูกอ่านไปใช้แล้วโดยหน่วยวิเคราะห์ ในงานวิจัยได้กำหนดให้แต่ละ bits ของ Flags มีสถานภาพเพื่อบอกการนำไปใช้จากหน่วยวิเคราะห์ข้อมูลที่แตกต่างกัน โดยเรียงจากบิตซ้ายไปขวาดังนี้

บิตที่ 1 สถานะของการวิเคราะห์ความยาวแพ็กเก็ต
บิตที่ 2 สถานะของการวิเคราะห์เวลาระหว่างการมา
บิตที่ 3 สถานะของการวิเคราะห์เวลาการสื่อสารข้อมูลไปและกลับ
บิตที่ 4 สถานะของการวิเคราะห์ความล่าช้า
บิตที่ 5 สถานะของการวิเคราะห์ความสูญเสีย
บิตที่ 6 สถานะของการวิเคราะห์ข้อมูลโฟลว
บิตที่ 7 , 8 ยังไม่ถูกใช้งาน

ดังนั้นถ้าทุก bits ของ Flags มีค่าเป็น 0 ก็จะหมายถึง ข้อมูลที่กำลังวิเคราะห์ได้ถูกนำไปใช้วิเคราะห์โดยหน่วยวิเคราะห์ข้อมูลแล้วในทุกพารามิเตอร์ ดังแสดงได้ในภาพที่ 43

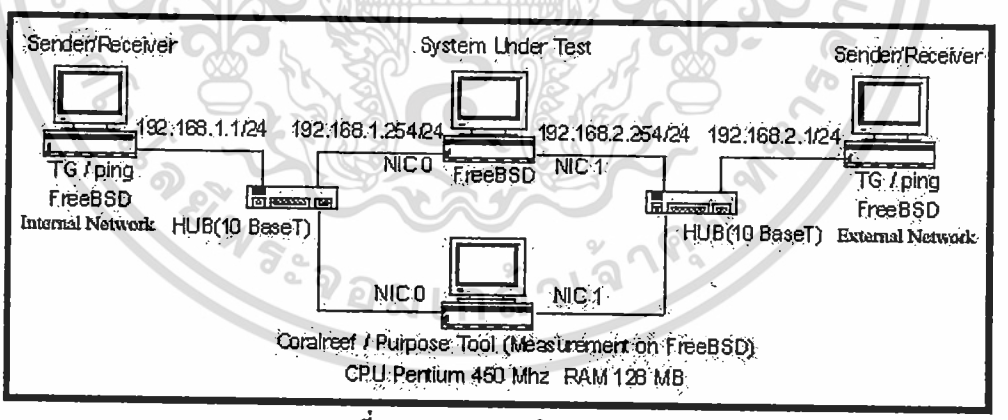
```
MAX_REC = 10000;
While (true) {
        If (pkt_i.flags ∈ zero)
                Replace packet information in each Table on memory;
        Else
                Break and report result to end user;
        If (++i > MAX_REC)
            i = 1
}
```

ภาพที่ 43 ขั้นตอนการจัดเก็บข้อมูลแต่ละตารางบนหน่วยความจำ

บทที่ 4
การทดสอบความถูกต้อง

4.1 บทนำ

เพื่อให้เกิดความน่าเชื่อถือในการออกแบบวิธีการตรวจวัดการสื่อสารข้อมูล ในงานวิจัยได้ทำการทดลองพัฒนาเครื่องมือตรวจวัดด้วยโปรแกรมภาษา C++ บนระบบปฏิบัติการฟรีบีเอสดี(FreeBSD 4.5-Release) ตัวภาษาจะเรียกใช้ Libpcap Library ในการตรวจจับข้อมูล และนำเครื่องมือไปทำการทดลองวัดการสื่อสารข้อมูลที่จำลองสถานการณ์คอมพิวเตอร์ที่อยู่ระหว่างสองเครือข่ายย่อยที่ใช้ไอพีหมายเลข 192.168.1.1/24 และ 192.168.2.1/24 ตามลำดับ โดยใช้ซอฟท์แวร์สร้างการสื่อสารข้อมูล (Traffic Generator)ด้วย TG และ ping คอมพิวเตอร์ที่ทำหน้าที่เป็นตัวระบบ(System Under Test)ติดตั้งระบบปฏิบัติการฟรีบีเอสดีและ Routed Protocol มี Network Interface Cards จำนวน 2 หน่วย แต่ละหน่วยใช้ไอพีหมายเลข 192.168.1.254 และ 192.168.2.254 ทำหน้าที่เป็น Gateway เชื่อมต่อกับ HUB ที่อยู่ฝั่งเครือข่ายภายในและภายนอกตามลำดับ คอมพิวเตอร์ที่ใช้ตรวจวัดมีคุณสมบัติของ CPU เป็น Pentium 2 ความเร็ว 450 Mhz หน่วยความจำขนาด 128 MB ตรวจจับแพ็กเก็ตผ่าน 2 NIC ที่ความเร็ว 10 Mb/sec โดยกำหนดแฟ้มติดตั้งให้เป็นไปตามภาพที่ 44 (ดูในภาคผนวก)



ภาพที่ 44 แบบจำลองในการทดลอง

แบบจำลองสถานการณ์ที่กำหนดจะถูกนำไปใช้เพื่อพิสูจน์ความน่าเชื่อถือในการตรวจวัดพารามิเตอร์ต่าง ๆ ในงานวิจัย ซึ่งจะพิจารณาและนำเสนอวิธีการทดสอบความถูกต้องของการตรวจวัดจากค่าทางสถิติพื้นฐานในแต่ละพารามิเตอร์ด้วยวิธีการต่าง ๆ ดังนี้

4.1.1 การวิเคราะห์ความยาวและเวลาระหว่างการมา ทดสอบด้วยการรับส่งข้อมูลระหว่างคอมพิวเตอร์ตามแบบจำลองที่อยู่ฝั่งเครือข่ายภายในไปยังเครือข่ายภายนอก ให้ทำการรับส่งข้อมูล 9 ชุด แต่ละชุดจะสุ่มความยาวแพ็กเก็ตที่ไม่เท่ากันจำนวน 500 แพ็กเก็ตโดยใช้โปรแกรมชื่อ TG(Traffic Generator) จากนั้นตรวจจับข้อมูลแต่ละชุดด้วยเครื่องมือวัดในงานวิจัย ที่นิยามเรียกว่า Purpose Tool และเครื่องมือวัดที่พัฒนาจาก Corallib Library เปรียบเทียบกันด้วยค่าทางสถิติต่าง ๆ ได้แก่ ค่าเฉลี่ย, ค่าความแตกต่าง และค่าความเชื่อมั่น โดยมีหน่วยวัดความยาวแพ็กเก็ตเป็นไบต์(bytes) และเวลาระหว่างการมาเป็น ไมโครวินาที(Microseconds) ด้วยสมการดังนี้

$$avg_t(p) = \frac{\sum_{i=1}^{n} p_i}{n} \tag{3}$$

โดยที่ $avg_t(p)$ ใช้แทน ค่าเฉลี่ยของข้อมูล p ชุดที่ t

p ใช้แทน ความหมายของข้อมูลที่ต้องการทดสอบ

p_i ใช้แทน ค่าของข้อมูลที่ต้องการทดสอบ เช่น ค่าความยาวหรือเวลาระหว่างการมาที่ตรวจจับได้ในลำดับที่ i

n ใช้แทน จำนวนข้อมูลทั้งหมดที่ตรวจจับได้

$$diff_t = \left| avg_t(p) - avg_t(c) \right| \tag{4}$$

โดยที่ $diff_t$ ใช้แทน ค่าความแตกต่างในชุดที่ t

$avg_t(p)$ ใช้แทน ค่าเฉลี่ยของข้อมูลที่ได้รับความสนใจ เช่น ค่าเฉลี่ยความยาวหรือเวลาระหว่างการมาที่ตรวจวัดได้จากเครื่องมือในงานวิจัย

$avg_t(c)$ ใช้แทน ค่าเฉลี่ยของข้อมูลที่นำมาเปรียบเทียบ เช่น ค่าความยาวหรือเวลาระหว่างการมาที่ตรวจวัดได้จากเครื่องมือชนิดอื่น

$$cf(c_1, c_2) = (\bar{x} - t_{[1-\alpha/2;n-1]} s/\sqrt{n}, \bar{x} + t_{[1-\alpha/2;n-1]} s/\sqrt{n}) \tag{5}$$

โดยที่ $cf(c_1, c_2)$ ใช้แทน ค่าความความเชื่อมั่น(Confidence Interval)

$\bar{x}$ ใช้แทน ค่าเฉลี่ยความยาวหรือเวลาระหว่างการมาของแพ็กเก็ต

α ใช้แทน ความสำคัญในที่นี้กำหนดเป็น 95%

n ใช้แทน จำนวนครั้งในการทดสอบ

$t_{[1-\alpha/2,n-1]}$ ใช้แทน ค่าความแปรผันของคะแนน t ที่สัมประสิทธิ์ (1 – 95 / 2) หรือ ทดสอบที่ความเชื่อมั่น 95% ณ ระดับ n – 1

s ใช้แทน ค่าเบี่ยงเบนมาตรฐานของความยาวหรือเวลาระหว่างการมาของแพ็กเก็ต

4.1.2 การวิเคราะห์เวลาการสื่อสารข้อมูลไปกลับ แบ่งการทดลองออกเป็น 2 ครั้ง ได้แก่

ครั้งที่ 1 ทดลองกับข้อมูลชนิด ICMP ทำการทดลองด้วยการรับส่งข้อมูลระหว่างคอมพิวเตอร์ตามแบบจำลองให้ส่งข้อมูลจากเครื่องที่อยู่ฝั่งเครือข่ายภายในไปยังเครื่องที่อยู่ฝั่งเครือข่ายภายนอกด้วย คำสั่ง ping เป็นจำนวน 10 ชุด ชุดละ 1,000 แพ็กเก็ต แต่ละชุดของการทดลองได้กำหนดความล่าช้าที่แตกต่างกันบนเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นตัวระบบ คือ 0, 10, 20, 50, 80, 100, 200, 500, 700 และ 1000 ms ตามลำดับ และขอนิยามเรียกความล่าช้าเสมือนเหล่านี้ว่า Emulated Delay โดยใช้ความสามารถของคำสั่ง ipfw ซึ่งเป็นคำสั่งสำเร็จรูปบนระบบปฏิบัติการ FreeBSD หน่วงเวลาการรับส่งข้อมูลแต่ละแพ็กเก็ตในฝั่งขาเข้าตามรูปแบบดังนี้

```
ipfw add pipe 1 ip from any to any in
ipfw pipe 1 config delay <n> ms
```

โดยที่ <n> หมายถึง ตัวเลขจำนวนเต็ม ที่มีหน่วยเวลาเป็น มิลลิวินาที

การทดลองในแต่ละชุดจะทำการตรวจจับด้วยเครื่องมือในงานวิจัยเปรียบเทียบกับเวลาที่ตรวจวัดได้จากคำสั่ง ping โดยมีหน่วยวัดเวลาเป็นมิลลิวินาที ทุกแพ็กเก็ตจากทั้ง 2 เครื่องมือจะถูกนำไปหาผลต่างของค่าฉลี่ยจาก Emulated Delay และ ทำบรรทัดฐาน(Normalize)ค่าเวลา ด้วยสมการดังนี้

$$diff_t = \left| avg_t(tr) - ted_t \right| \quad \text{where} \quad ted_t = td_t \times 2 \qquad (6)$$

โดยที่ $avg_t(tr)$ ใช้แทน ค่าเฉลี่ยเวลาไปกลับที่ตรวจวัดได้จากเครื่องมือในชุดที่ t

ted_t ใช้แทน ค่า Emulated Delay ชุดที่ t

td_t ใช้แทน เวลาหน่วงที่ตัวระบบ(Time Delay) ตามงานวิจัยคือ 0, 10, 20, 50, 80, 100, 200, 500, 700 และ 1000 ms ตามลำดับ

$$Nml_t = \frac{diff_t}{ted_t} \quad (7)$$

โดยที่ Nml_t หมายถึง บรรทัดฐาน(Normalize)ผลต่างของค่าเฉลี่ยชุดที่ t

ครั้งที่ 2 ทดลองกับข้อมูลชนิด TCP ทำการทดลองด้วยการรับส่งข้อมูลระหว่างคอมพิวเตอร์ตามแบบจำลองให้ส่งข้อมูลความยาวขนาด 500 ไบต์ จากเครื่องที่อยู่ฝั่งเครือข่ายภายในไปยังเครื่องที่อยู่ฝั่งเครือข่ายภายนอกด้วยโปรแกรม ttcp ซึ่งเป็นโปรแกรมสำเร็จรูปในการรับส่งข้อมูลชนิด TCP บนระบบปฏิบัติการ FreeBSD เป็นจำนวน 10 ชุด ชุดละ 1,000 ครั้ง แต่ละชุดของการทดลองได้กำหนดความล่าช้าที่แตกต่างกันบนเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นตัวระบบ คือ 0, 10, 20, 50, 80, 100, 200, 500, 700 และ 1000 ms ตามลำดับ เช่นเดียวกับการทดลองด้วยข้อมูลชนิด ICMP ในการทดลองนี้เนื่องจากปัจจุบันยังไม่มีเครื่องมือวัดใดออกแบบเป็นลักษณะใกล้เคียงกัน ในงานวิจัยจึงกำหนดการทดลองให้เปรียบเทียบกันระหว่างช่วงเวลาเชื่อมต่อ(Connection) และปิดการเชื่อมต่อ(Close)ที่ตรวจวัดได้จากเครื่องมือวิจัย และใช้สมการวัดความถูกต้องแบบเดียวกับการวัดข้อมูลชนิด ICMP คือ หาผลต่างของค่าฉลี่ยจาก Emulated Delay และการทำบรรทัดฐานค่าเวลาเปรียบเทียบกันในแต่ละชุด

4.1.3 การวิเคราะห์ความล่าช้า ทดลองด้วยการรับส่งข้อมูลระหว่างคอมพิวเตอร์ตามแบบจำลองให้ส่งข้อมูลความยาวขนาด 1,500 ไบต์ จากเครื่องที่อยู่ฝั่งเครือข่ายภายในไปยังเครื่องที่อยู่ฝั่งเครือข่ายภายนอกเป็นจำนวน 10 ชุด ชุดละ 1,000 แพ็กเก็ต แต่ละชุดของการทดลองได้กำหนดความล่าช้าที่แตกต่างกันบนเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นตัวระบบ คือ 0, 10, 30, 50, 70, 100, 300, 500, 700 และ 1000 ms ตามลำดับ เช่นเดียวกับการทดลองวิเคราะห์เวลาการสื่อสารข้อมูลไปกลับ ในการทดลองนี้ได้ทำซ้ำการทดลองแบบเดียวกันถึง 3 ครั้ง แต่ละครั้งเป็นการทดลองกับข้อมูลต่างชนิดกันคือ ICMP, TCP และ UDP ตามลำดับ ผลการทดลองในแต่ชุดจะถูกนำไปเปรียบเทียบกัน โดยวัดความถูกต้องจากการสมการหาผลต่างของค่าฉลี่ยจาก Emulated Delay และการทำบรรทัดฐานค่าเวลาเปรียบเทียบกันในแต่ละชุด ดังสมการต่อไปนี้

$$diff_t = \left| avg_t(d) - td_t \right| \quad (8)$$

โดยที่ $avg_t(d)$ ใช้แทน ค่าเฉลี่ยเวลาความล่าช้าที่ตรวจวัดได้จากเครื่องมือในชุดที่ t

td_t ใช้แทน เวลาหน่วงที่ตัวระบบ(Time Delay) ตามงานวิจัยคือ 0, 10, 20, 50, 80, 100, 200, 500, 700 และ 1000 ms ตามลำดับ

$$Nml_t = \frac{diff_t}{ta_t} \tag{9}$$

4.1.4 การวิเคราะห์ความสูญเสียจากการสื่อสารข้อมูล ทดลองด้วยการรับส่งข้อมูลระหว่างคอมพิวเตอร์ตามแบบจำลองให้ส่งข้อมูลจากเครื่องที่อยู่ฝั่งเครือข่ายภายในไปยังเครื่องที่อยู่ฝั่งเครือข่ายภายนอก โดยแบ่งการทดลองออกเป็น 2 ครั้ง ดังนี้

ครั้งที่ 1 ทดลองส่งข้อมูลด้วย คำสั่ง ping เป็นจำนวน 8 ชุด ชุดละ 100 แพ็กเก็ต แต่ละชุดของการทดลองได้กำหนดความสูญเสียที่แตกต่างกันบนเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นตัวระบบ คือ 0, 7, 22, 42, 54, 74, 86 และ 99 เปอร์เซนต์ตามลำดับ และขอนิยามเรียกความสูญเสียเสมือนเหล่านี้ว่า Emulated Loss Rate โดยจะอาศัยความสามารถของคำสั่ง ipfw ซึ่งเป็นคำสั่งสำเร็จรูปบนระบบปฏิบัติการ FreeBSD ตามรูปแบบดังนี้

```
ipfw add pipe 1 ip from any to any in
ipfw pipe 1 config plr <n>
```

โดยที่ <n> หมายถึง ตัวเลขจำนวนเต็ม ที่มีค่าอยู่ระหว่าง 0.00 ถึง 1.00

ในแต่ละการทดลองจะทำการตรวจจับด้วยเครื่องมือในงานวิจัยเปรียบเทียบค่าที่กำหนดความสูญเสียบนตัวระบบและการตรวจวัดที่ได้จากคำสั่ง ping ผลลัพธ์ความสูญเสียในทุกแพ็กเก็ตจากทั้ง 2 เครื่องมือจะถูกนำไปหาผลต่างของค่าฉลี่ยจากสมการ

$$loss_t = \frac{N\,x100}{L} \tag{10}$$

โดยที่ $loss_t$ หมายถึง เปอร์เซนต์ความสูญเสียในชุดที่ t

N หมายถึง จำนวนแพ็กเก็ตที่ทดลองส่ง

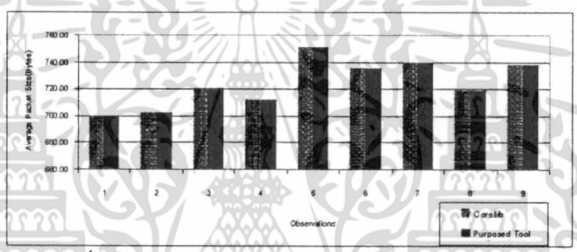
L หมายถึง จำนวนแพ็กเก็ตที่เกิดความสูญเสีย

ครั้งที่ 2 ทดลองส่งข้อมูลชนิด TCP, UDP และ ICMP เป็นจำนวน 10 ชุด ชุดละ 100 แพ็กเก็ต โดย ICMP จัดส่งข้อมูลที่มีความยาวแต่ละแพ็กเก็ตขนาด 84 ไบต์ ข้อมูล TCP ส่งครั้งเดียวด้วยความยาวรวม 145,500 ไบต์ และ UDP แต่ละแพ็กเก็ตส่งด้วยความยาวขนาด 80 ไบต์ แต่ละชุดของ

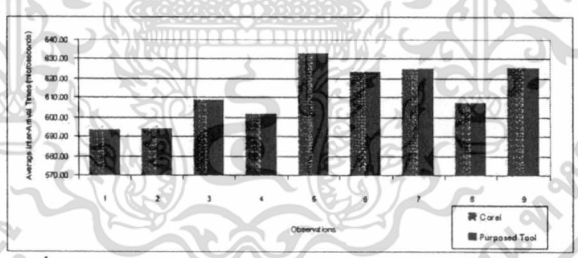
การทดลองได้กำหนดความสูญเสียที่แตกต่างกันบนเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นตัวระบบ คือ 0, 5, 10, 20, 30, 50, 70 และ 90 เปอร์เซนต์ตามลำดับ เพื่อเปรียบเทียบความสูญเสียที่แตกต่างกันของข้อมูลในแต่ละชนิด โดยใช้สมการหาอัตราความสูญเสียแบบเดียวกับการทดลองในครั้งที่ 1

4.2 ผลการวิเคราะห์ความยาวและเวลาระหว่างการมา

ทดลองสุ่มข้อมูลที่มีความยาวไม่เท่ากันเป็นจำนวน 500 แพ็กเก็ตไปในฝั่งตรงกันข้าม แล้วทำการตรวจวัด เพื่อเปรียบเทียบผลลัพธ์การทดลองระหว่างเครื่องมือวัดในงานวิจัย(Purposed Tool) และเครื่องมือที่พัฒนาจาก Corallib Library เป็นจำนวน 9 ครั้ง ผลการทดลองของทั้ง 2 เครื่องมือได้คุณลักษณะของข้อมูล ดังภาพที่ 45 และ 46



รูปที่ 45 ผลการวิเคราะห์ความยาวของแพ็กเก็ต



ภาพที่ 46 ผลการวิเคราะห์เวลาระหว่างการมาของแพ็กเก็ต

ผลการตรวจวัดและเปรียบเทียบความยาวของทุกแพ็กเก็ตจากทั้ง 2 เครื่องมือมีค่าเท่ากันในทุกแพ็กเก็ต เมื่อวัดค่าเฉลี่ยในทุกแพ็กเก็ตได้ค่าเป็น 724.18 ไบต์ ค่าเปอร์เซนต์เฉลี่ยของความแตกต่างระหว่างเครื่องมือเป็น 0.00 ไบต์ เมื่อเปรียบเทียบด้วย Confidence Interval(95%) ได้ค่าเฉลี่ยของการตรวจวัดอยู่ในช่วงเดียวกันคือ (713.25,735.12) วิเคราะห์ได้ว่าทั้งสองเครื่องมือ ตรวจวัดความยาวของแพ็กเก็ตได้ไม่แตกต่างกัน

ส่วนผลการตรวจวัดเวลาระหว่างการมามีความแตกต่างกันน้อยมาก การตรวจวัดด้วย Corallib และเครื่องมือวิจัยให้ความแตกต่างกันในหน่วยเวลาเป็น Microseconds ค่าเฉลี่ยเวลาของ Corallib และ

เครื่องมือวิจัยคือ 612.34 และ 612.35 Microseconds ตามลำดับ ค่าเปอร์เซนต์เฉลี่ยของความแตกต่างระหว่างเครื่องมือคือ 0.000021 Microseconds เมื่อเปรียบเทียบด้วย Confidence Interval(95%) ได้ค่าเฉลี่ยของการตรวจวัดอยู่ในช่วง (603.30, 621.38) Microseconds และ (603.31,621.39) Microseconds วิเคราะห์ได้ว่าทั้งสองเครื่องมือตรวจวัดเวลาระหว่างการมาได้ไม่แตกต่างกัน

4.3 ผลการวิเคราะห์เวลาการสื่อสารข้อมูลไปกลับ

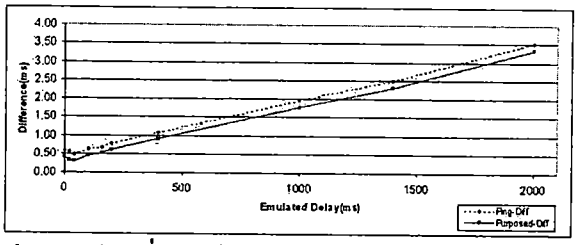
การวิเคราะห์เวลาการสื่อสารข้อมูลไปกลับ จะวิเคราะห์เฉพาะข้อมูลชนิด ICMP และ TCP เท่านั้น ส่วนการวิเคราะห์ข้อมูลชนิด UDP สามารถทำได้แต่อาจต้องถอดส่วนข้อมูลของแพ็กเก็ตไปจนถึงชั้นแอพพลิเคชั่น(Applicaltion) ดังนั้น อาจทำให้ขบวนการวิเคราะห์เกิดความล่าช้า เนื่องจากต้องใช้หน่วยความจำในการจัดเก็บข้อมูล เพื่อรอการตอบกลับของคู่สนทนาเป็นจำนวนมากทำให้เวลาในการค้นหานานกว่าข้อมูลในสองชนิดแรก และอาจทำให้จำนวนข้อมูลสะสมจนเกินความสามารถของหน่วยความจำที่มีอยู่บนเครื่องคอมพิวเตอร์ที่ทำหน้าวิเคราะห์ระบบ

4.3.1 ผลการวิเคราะห์ข้อมูลชนิด ICMP

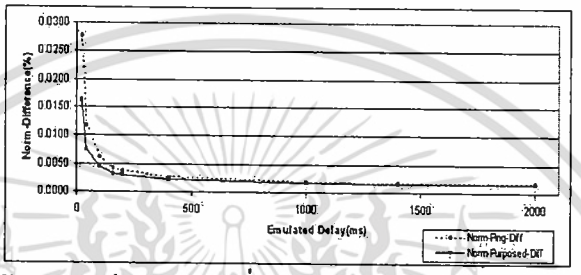
ทดลองส่งข้อมูลด้วย ICMP จำนวน 1,000 แพ็กเก็ตเป็นจำนวน 10 ชุดระหว่างเครือข่าย เพื่อเปรียบเทียบผลการตรวจวัดที่ได้จากเครื่องมือวิจัยและคำสั่ง ping ในระหว่างการรับส่งข้อมูลทำการหน่วงเวลาบนคอมพิวเตอร์ที่ทำหน้าเป็นตัวระบบในฝั่งขาเข้า(In Bound)ด้วยอัตราเวลา 0, 10, 20, 50, 80, 100, 200, 500, 700 และ 1000 ms ในแต่ละชุดตามลำดับ โดยใช้ความคำสั่ง ipfw พร้อมทั้งกำหนดค่า HZ OPTION ของ Kernel บนระบบปฏิบัติการ FreeBSD เป็น 5000 HZ ผลการทดลองในแต่ละชุดจะถูกนำไปหา ผลต่างของค่าเฉลี่ยเวลาไปกลับจาก Emulated Delay และ ทำบรรทัดฐานผลต่างของค่าเฉลี่ยเวลาไปกลับ แล้วจึงนำไปเขียนกราฟเพื่อเปรียบเทียบกัน ผลคือได้คุณลักษณะของข้อมูลดังภาพที่ 47 และ 48 และตารางข้อมูลที่ 2

ตารางที่ 2 ผลทดลองตรวจวัดเวลาไปกลับของข้อมูลชนิด ICMP

เวลาหน่วง	เวลาหน่วงไปกลับ	เวลาที่ตรวจวัดได้จากคำสั่ง Ping	เวลาที่ตรวจวัดได้จาก เครื่องมือวิจัย	ผลต่างของค่าเฉลี่ยจาก Emulated Delay ด้วย Ping Command	ผลต่างของค่าเฉลี่ยจาก Emulated Delay ด้วยเครื่องมือวิจัย	บรรทัดฐานผลต่างของค่าเฉลี่ย จาก Ping Command	บรรทัดฐานผลต่างของค่าเฉลี่ย จากเครื่องมือวิจัย
0	0	0.581830	0.415600	0.581830	0.415600		
10	20	20.554980	20.326190	0.554980	0.326190	0.027749	0.016310
20	40	40.473630	40.300410	0.473630	0.300410	0.011841	0.007510
50	100	100.623350	100.451570	0.623350	0.451570	0.006234	0.004516
80	160	160.674350	160.514480	0.674350	0.514480	0.004215	0.003216
100	200	200.775380	200.608320	0.775380	0.608320	0.003877	0.003042
200	400	401.066020	400.895450	1.066020	0.895450	0.002665	0.002239
500	1000	1001.948030	1001.755430	1.948030	1.755430	0.001948	0.001755
700	1400	1402.488600	1402.302570	2.488600	2.302570	0.001778	0.001645
1000	2000	2003.530740	2003.333260	3.530740	3.333260	0.001765	0.001667
ค่าเฉลี่ยของเวลา		533.271691000	533.090328	1.271691000	1.090328	0.006896762	0.004655325



ภาพที่ 47 ผลต่างของค่าเฉลี่ยเวลาไปกลับจาก Emulated Delay ของข้อมูล ICMP



ภาพที่ 48 ผลการบรรทัดฐานผลต่างของค่าเฉลี่ยเวลาไปกลับจาก Emulated Delay ของข้อมูล ICMP

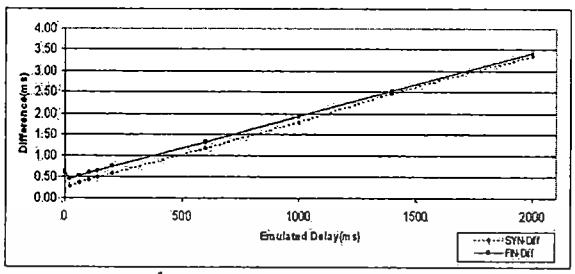
จากตารางที่ 2 ผลต่างของการหน่วงเวลากับเวลาที่ตรวจวัดได้จากทั้งสองเครื่องมือมีความแตกต่างกันในทุกชุดของการทดลอง โดยผลต่างของค่าเฉลี่ยเวลาไปกลับจาก Emulated Delay ที่ตรวจวัดได้จากคำสั่ง ping คือ 1.276191 ms ส่วนเครื่องมือวิจัยให้ความแตกต่างที่ 1.090328 ms ซึ่งน้อยกว่าการตรวจวัดด้วยคำสั่ง ping ความแตกต่างนี้เป็นผลมาจากทั้งสองเครื่องมือมีการกำกับเวลาของข้อมูลเพื่อนำไปใช้ในการคำนวณบนชั้นเครือข่ายที่แตกต่างกัน และเมื่อทำบรรทัดฐาน (Normalize)ผลต่างของทั้งสองเครื่องมือจะพบว่ามีเปอร์เซนต์ความแตกต่างที่ตรวจวัดได้จากคำสั่ง ping คือ 0.006897 ms ส่วนเครื่องมือวิจัยตรวจวัดได้ 0.004655 ms และเมื่อพิจารณาจากภาพที่ 47 และ 48 จะพบว่าการทดลองในแต่ละครั้งโดยใช้เวลาหน่วงที่มากขึ้นคุณลักษณะของทั้งสองเครื่องมือจะลดความแตกต่างกันลงไป ซึ่งจะเห็นได้จากกราฟของทั้งสองเริ่มมีค่าใกล้เคียงกันมากขึ้น

4.3.2 ผลการวิเคราะห์ข้อมูลชนิด TCP

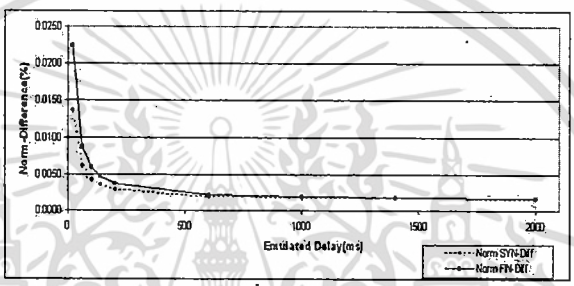
ทดลองส่งข้อมูลความยาวรวมขนาด 500 ไบต์ ชนิด TCP เป็นจำนวน 10 ชุด ชุดละ 1,000 ครั้ง ระหว่างเครือข่ายภายในและภายนอก ในระหว่างการรับส่งข้อมูลได้ทำการหน่วงเวลาของการทดลองบนคอมพิวเตอร์ที่ทำหน้าที่เป็นตัวระบบฝั่งขาเข้า(In Bound)ด้วยอัตรา 0, 10, 30, 50, 70, 100, 300, 500, 700 และ 1000 ms ในแต่ละชุดตามลำดับ ผลการทดลองในแต่ละชุดจะตรวจวัดค่าเวลาจากเครื่องมือวิจัยเท่านั้น โดยตรวจวัดเฉพาะช่วงเวลาเชื่อมต่อและปิดการเชื่อมต่อ จากนั้นจึงนำค่าเวลาในแต่ละชุดไปหา ผลต่างของค่าเฉลี่ยจาก Emulated Delay และ ทำบรรทัดฐาน(Normalize)ผลต่างของค่าเฉลี่ย เพื่อเขียนกราฟเปรียบเทียบกัน ผลคือได้คุณลักษณะของข้อมูลดังภาพที่ 49 และ 50 และตารางข้อมูลที่ 3

ตารางที่ 3 ผลทดลองตรวจวัดเวลาไปกลับของข้อมูลชนิด TCP ในช่วงการเชื่อมต่อ

เวลาหน่วง	เวลาหน่วงไปกลับ	เวลาช่วงการเชื่อมต่อ	เวลาช่วงปิดการเชื่อมต่อ	ผลต่างของค่าเฉลี่ยจาก Emulated Delay ในช่วงการเชื่อมต่อ	ผลต่างของค่าเฉลี่ยจาก Emulated Delay ในช่วงปิดการเชื่อมต่อ	บรรทัดฐานผลต่างของค่าเฉลี่ย ในช่วงการเชื่อมต่อ	บรรทัดฐานผลต่างของค่าเฉลี่ยในช่วงปิดการเชื่อมต่อ
0	0	0.583630	0.624610	0.583630	0.624610	0.000000	0.000000
10	20	20.271820	20.448650	0.271820	0.448650	0.013591	0.022433
30	60	60.367340	60.522120	0.367340	0.522120	0.006122	0.008702
50	100	100.421940	100.589390	0.421940	0.589390	0.004219	0.005894
70	140	140.492190	140.648600	0.492190	0.648600	0.003516	0.004633
100	200	200.575840	200.739100	0.575840	0.739100	0.002879	0.003696
300	600	601.184090	601.333460	1.184090	1.333460	0.001973	0.002222
500	1000	1001.795030	1001.924860	1.795030	1.924860	0.001795	0.001925
700	1400	1402.472340	1402.522810	2.472340	2.522810	0.001766	0.001802
1000	2000	2003.357900	2003.413080	3.357900	3.413080	0.001679	0.001707
ค่าเฉลี่ยของเวลา		553.152212	553.276668	1.152212	1.276668	0.003754	0.005301



ภาพที่ 49 ผลต่างของค่าเฉลี่ยเวลาไปกลับจาก Emulated Delay ของข้อมูล TCP



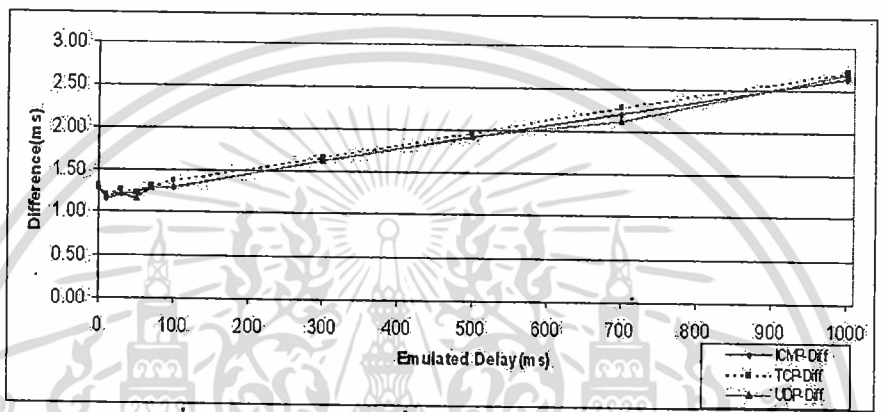
ภาพที่ 50 ผลการบรรทัดฐานผลต่างของค่าเฉลี่ยเวลาไปกลับจาก Emulated Delay ของข้อมูล TCP

จากภาพที่ 49, 50 และตารางที่ 3 ผลต่างระหว่างเวลาหน่วงกับเวลาที่ตรวจวัดได้จากทั้งสองเครื่องมือมีความแตกต่างกันในทุกการทดลองของแต่ละชุด และพบว่าเมื่อทำการทดลองด้วยการหน่วงเวลาที่มากขึ้น ช่วงเวลาของการเชื่อมต่อและปิดการเชื่อมต่อจะมีผลของความแตกต่างที่ใกล้เคียงกัน โดยมีผลต่างของค่าเฉลี่ยเวลาไปกลับจาก Emulated Delay ที่ตรวจวัดได้ในช่วงการเชื่อมต่อคือ 1.152212 ms ส่วนช่วงเวลาปิดการเชื่อมต่อให้ความแตกต่างที่ 1.276668 ms ซึ่งมีค่ามากกว่าในช่วงเวลาการเชื่อมต่อ ความแตกต่างนี้เป็นผลมาจากช่วงเวลาปิดการเชื่อมต่อของฝ่ายรับและส่งอาจต้องสูญเสียเวลาบางส่วนไปสำหรับเรียบเรียงข้อมูลก่อนปิดการเชื่อมต่อ เมื่อทำการบรรทัดฐาน(Normalize)ผลต่างของทั้งสองช่วงเวลา ผลคือเปอร์เซนต์ความแตกต่างที่ตรวจวัดได้ในช่วงการเชื่อมต่อ คือ 0.004171 ms ส่วนช่วงเวลาปิดการเชื่อมต่อคือ 0.005890 ms ผลการวิเคราะห์ในแต่ละครั้งเมื่อมีการทดลองโดยใช้การหน่วงเวลาที่มากขึ้น พบว่าทั้งสองช่วงเวลาจะเริ่มลดความแตกต่างกันลงไปซึ่งจะเห็นได้จากกราฟของทั้งสองเริ่มมีค่าใกล้เคียงกันมากขึ้น

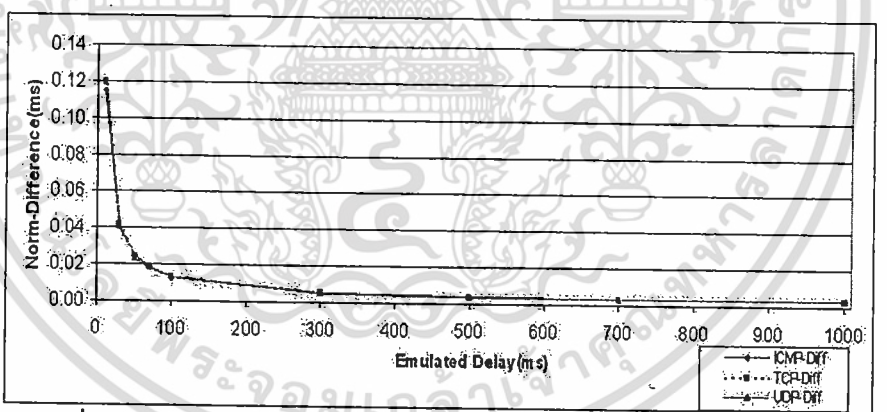
4.4 ผลการวิเคราะห์ความล่าช้าผ่านตัวระบบ

ทดลองส่งข้อมูลทั้ง 3ชนิดคือ ICMP, TCP และ UDP ที่มีขนาด 1,500 ไบต์ จำนวน 10 ชุด ชุดละ 1,000 แพ็กเก็ตระหว่างเครือข่ายภายในและภายนอก ระหว่างการรับส่งข้อมูลทำการหน่วงเวลาการ

ทดลองบนเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นตัวระบบฝั่งขาเข้า(In Bound)ด้วยอัตราเวลา 0, 10, 30, 50, 70, 100, 300, 500, 700 และ 1000 ms ในแต่ละชุดตามลำดับ ผลการทดลองในแต่ละชุดจะถูกนำไปหาผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay และ ทำบรรทัดฐาน(Normalize)ผลต่างของค่าเฉลี่ยความล่าช้า แล้วจึงนำผลการตรวจวัดด้วยเครื่องมือวิจัยไปเขียนกราฟเพื่อเปรียบเทียบกัน ระหว่างข้อมูลทั้ง 3 ชนิด ผลคือได้คุณลักษณะของข้อมูลดังภาพที่ 51 และ 52 และตารางข้อมูลที่ 4



ภาพที่ 51 ผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay



ภาพที่ 52 ผลการบรรทัดฐานผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay

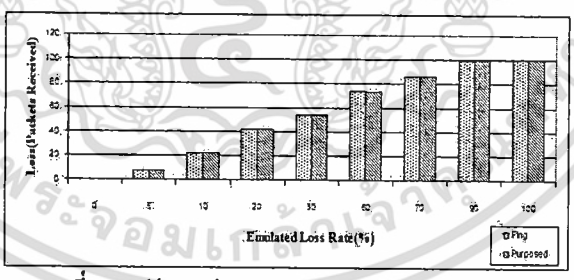
ตารางที่ 4 ผลทดลองตรวจวัดความล่าช้าของข้อมูลผ่านตัวระบบ

เวลาหน่วง	ตรวจวัดเวลาข้อมูลชนิด ICMP	ตรวจวัดเวลาข้อมูลชนิด TCP	ตรวจวัดเวลาข้อมูลชนิด UDP	ผลต่างของค่าเฉลี่ย จาก Emulated Delay ด้วยข้อมูลชนิด ICMP	ผลต่างของค่าเฉลี่ย จาก Emulated Delay ด้วยข้อมูลชนิด TCP	ผลต่างของค่าเฉลี่ย จาก Emulated Delay ด้วยข้อมูลชนิด UDP	บรรทัดฐานผลต่างของค่าเฉลี่ย ด้วยข้อมูลชนิด ICMP	บรรทัดฐานผลต่างของค่าเฉลี่ย ด้วยข้อมูลชนิด TCP	บรรทัดฐานผลต่างของค่าเฉลี่ย ด้วยข้อมูลชนิด UDP
0	1.29518	1.31395	1.28953	1.295180	1.313950	1.289530	0.000000	0.0000000	0.0000000
10	11.14945	11.19793	11.19438	1.149450	1.197930	1.194380	0.1149450	0.1197930	0.1194380
30	31.20823	31.25626	31.22897	1.208230	1.256260	1.228970	0.0402743	0.0418753	0.0409657
50	51.22351	51.23388	51.16018	1.223510	1.233880	1.160180	0.0244702	0.0246776	0.0232036
70	71.28950	71.31102	71.29564	1.289500	1.311020	1.295640	0.0184214	0.0187289	0.0185091
100	101.28520	101.3672	101.31170	1.285220	1.367210	1.311650	0.0128522	0.0136721	0.0131165
300	301.62170	301.6628	301.62910	1.621700	1.662790	1.629110	0.0054057	0.0055426	0.0054304
500	501.90750	501.9683	501.92590	1.907450	1.968300	1.925870	0.0038149	0.0039366	0.0038517
700	702.20660	702.2805	702.12180	2.206550	2.280490	2.121820	0.0031522	0.0032578	0.0030312
1000	1002.63900	1002.715	1002.70000	2.639000	2.714880	2.700410	0.0026390	0.0027149	0.0027004
เวลาเฉลี่ย	277.58260	277.6307	277.58580	1.582579	1.630671	1.585756	0.0225975	0.0234199	0.0230247

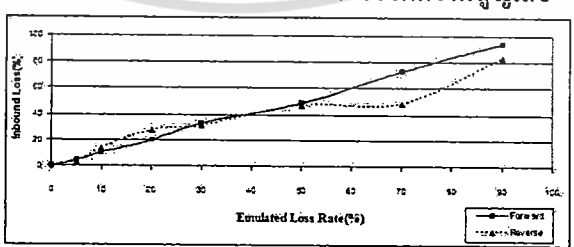
จากตารางที่ 4 ผลต่างของการหน่วงเวลากับความล่าช้าที่ ตรวจวัดได้จากข้อมูลทั้ง 3 ชนิดมีความแตกต่างกันในทุกครั้งของการทดลอง โดยที่ผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay ที่ตรวจวัดได้จากข้อมูลชนิด ICMP, TCP และ UDP คือ 1.582579, 1.630671 และ 1.585756 ms ตามลำดับ ความแตกต่างเหล่านี้เป็นผลมาจากเวลาที่ต้องสูญเสียไปกับการสื่อสารผ่านเข้าและออกจากตัวระบบ ซึ่งแพ็กเก็ตชนิด TCP มีความยาวของส่วนหัวมากกว่า ICMP และ UDP ดังนั้นค่าของเวลาจึงมีค่ามากกว่าเพียงเล็กน้อย เมื่อทำการบรรทัดฐาน(Normalize)ผลต่างของค่าเฉลี่ยความล่าช้ากับข้อมูลทั้ง 3 ชนิดดังภาพที่ 52 พบว่า เปอร์เซนต์ความแตกต่างที่ตรวจวัดได้จากข้อมูลชนิด ICMP, TCP และ UDP คือ 0.0251083, 0.0260221 และ 0.025583 ms ตามลำดับ โดยผลการทดลองในแต่ละชุดที่ใช้เวลาหน่วงมากขึ้น คุณลักษณะของข้อมูลแต่ละชนิดจะลดความแตกต่างกันลงไป สังเกตได้จากกราฟของข้อมูลทั้ง 3 ชนิดเริ่มมีค่าใกล้เคียงกันมากขึ้น

4.5 ผลการวิเคราะห์ความสูญเสียผ่านตัวระบบ

ทดลองส่งข้อมูล ICMP ด้วยคำสั่ง ping จำนวน 8 ชุด ชุดละ 100 แพ็กเก็ตระหว่างเครือข่ายภายในและภายนอก ในระหว่างการรับส่งข้อมูลทำการกำหนดความน่าจะเป็นให้เกิดความสูญเสียข้อมูลบนเครื่องคอมพิวเตอร์ที่เป็นตัวระบบในฝั่งขาเข้าด้วยอัตรา 0, 5, 10, 20, 30, 50, 70 และ 90 เปอร์เซ็นต์ในแต่ละชุดตามลำดับ โดยใช้คำสั่ง ipfw ผลการทดลองเมื่อเปรียบเทียบกับคำสั่ง ping สามารถตรวจวัดคุณลักษณะของข้อมูลได้ดังภาพที่ 53, 54 และตารางที่ 5



ภาพที่ 53 เปรียบเทียบผลการตรวจวัดความสูญเสีย



ภาพที่ 54 เปรียบเทียบความสูญเสียข้อมูลขาเข้าและขาออก

ตารางที่ 5 ผลทดลองตรวจวัดความสูญเสียของข้อมูลผ่านตัวระบบ

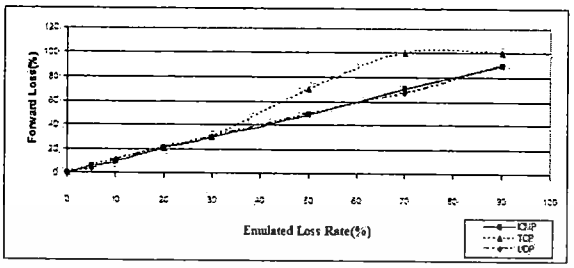
ทดลองครั้งที่	ตรวจวัดด้วยคำสั่ง Ping			ตรวจวัดด้วย เครื่องมือวิจัย								
	Emulated Loss Rate	Ping Received (packets)	Loss Rate(%)	Packets Forward (packets)	Forward Loss (packets)	Forward Loss Rate(%)	Packets Reverse (packets)	Reverse Loss (packets)	Reverse Loss Rate(%)	Forward & Reverse Loss Rate(%)	Received (packets)	Loss Rate(%)
1	0	100	0	100	0	0.0000	100	0	0.0000	0.0000	100	0
2	5	93	7	100	4	4.0000	96	3	3.1250	7.1250	93	7
3	10	78	22	100	10	10.0000	90	12	13.3333	23.3333	78	22
4	20	58	42	100	20	20.0000	80	22	27.5000	47.5000	58	42
5	30	46	54	100	33	33.0000	67	21	31.3433	64.3433	46	54
6	50	26	74	100	49	49.0000	51	24	47.0588	96.0588	27	73
7	70	14	86	100	73	73.0000	27	13	48.1481	121.1481	14	86
8	90	1	99	100	94	94.0000	6	5	83.3333	177.3333	1	99

จากตารางที่ 5 ผลการตรวจวัดความสูญเสียด้วยคำสั่ง ping และ เครื่องมือวิจัย พบว่าทุกครั้งของการทดลองเครื่องมือทั้งสองชนิดตรวจวัดได้ไม่แตกต่างกัน โดยที่แต่ละครั้งสามารถตรวจวัดอัตราความสูญเสีย(Loss Rate)ได้เป็น 0, 7, 22, 42, 54, 74, 86 และ 99 เปอร์เซนต์ตามลำดับ ดังแสดงได้ตามที่ภาพที่ 53 ส่วนภาพที่ 54 ในการทดลองครั้งเดียวกันเครื่องมือวิจัยมีความสามารถในการตรวจวัดอัตราความสูญเสียตามทิศทางขาออก(Forward Loss Rate)และตามทิศทางขาเข้า(Reverse Loss Rate)ได้ เมื่อนำไปวิเคราะห์ด้วยการวาดกราฟทิศทางขาออกจะมีลักษณะของอัตราความสูญเสียที่เพิ่มขึ้นตามลำดับ ซึ่งสอดคล้องหรือมีค่าแตกต่างกันเพียงเล็กน้อยกับอัตราความสูญเสียที่กำหนดให้กับตัวระบบ ส่วนทิศทางขาเข้าอัตราความสูญเสียจะมีค่าต่ำกว่าทิศทางขาออกเมื่อกำหนดอัตราความสูญเสียตั้งแต่ 50% ขึ้นไป ความแปรปรวนนี้เป็นผลมาจากแพ็กเก็ตฝ่ายตอบ(Echo Reply)มีจำนวนน้อยลงเนื่องมาจากความสูญเสียในฝั่งขาออก ดังนั้นขาเข้าจากภายนอกจึงมีอัตราส่วนของความสูญเสียต่ำกว่าทิศทางขาออกเมื่อคำนวณจากจำนวนทั้งหมดของแพ็กเก็ตฝ่ายตอบ

ทดลองอีกครั้งเพื่อเปรียบเทียบผลการตรวจวัดอัตราความสูญเสียของข้อมูลชนิด ICMP, TCP และ UDP ด้วยการส่งข้อมูลจำนวน 10 ชุด แต่ละชุดส่งด้วยข้อมูลชนิด ICMP ขนาด 84 ไบต์ จำนวน 1,000 แพ็กเก็ต ข้อมูลชนิด TCP ส่งเป็นความยาวรวม 145,500 ไบต์ และข้อมูลชนิด UDP ส่งเป็นแพ็กเก็ตขนาด 80 ไบต์ จำนวน 1,000 แพ็กเก็ตไปในฝั่งตรงกันข้าม(โดยเฉลี่ยแล้วข้อมูลแต่ละชนิดจะถูกส่งไปฝั่งตรงข้ามจำนวน 1,000 แพ็กเก็ตด้วย MTU ขนาด 1,500 ไบต์) แต่ละชุดของการทดลองจะมีการกำหนดอัตราความสูญเสียให้กับตัวระบบเป็น 0, 5, 10, 20, 30, 50, 70, 90 เปอร์เซนต์ตามลำดับ ผลการตรวจวัดได้คุณลักษณะของข้อมูลดังภาพที่ 55

ตารางที่ 6 ผลการทดลองตรวจวัดอัตราความสูญเสียกับข้อมูลต่างชนิด

Emulated Loss(%)	ICMP Loss Rate(%)	UDP Loss Rate(%)	TCP Loss Rate(%)
0	0.00	0.00	0.68
5	5.40	3.87	5.97
10	9.50	11.03	11.45
20	20.80	21.37	20.59
30	29.70	30.32	30.02
50	48.70	49.30	70.00
70	70.20	66.80	100.00
90	89.10	90.06	100.00



ภาพที่ 55 ผลการวิเคราะห์ความล่าช้าเปรียบเทียบกันระหว่างข้อมูลชนิด ICMP, TCP และ UDP

จากภาพที่ 55 ลักษณะของอัตราความสูญเสียข้อมูลชนิด ICMP, และ UDP มีความใกล้เคียงกับการกำหนดอัตราความสูญเสียที่ตัวระบบ(Emulated Loss Rate) ส่วน TCP จะมีความใกล้เคียงกับการกำหนดอัตราความสูญเสียในช่วงต่ำ ๆ เมื่อกำหนดไปถึงช่วงที่มีค่าสูงขึ้นข้อมูล TCP จะยกเลิกการติดต่อสื่อสารก่อนครบจำนวน เนื่องจากความสูญเสียที่เกิดขึ้นขาดความน่าเชื่อถือเกินกว่าการยอมรับได้ของการสื่อสารด้วยข้อมูลชนิด TCP ซึ่งสังเกตได้จากในช่วงอัตราความสูญเสียตั้งแต่ 70% ขึ้นไป

4.6 การทดลองตรวจวัดความล่าช้า และความสูญเสียพร้อมกัน

ตัวโปรแกรมในการวิเคราะห์ข้อมูลสามารถตรวจวัดแต่ละพารามิเตอร์แยกจากกัน หรือตรวจวัดไปพร้อมกันก็ได้ โดยเฉพาะการตรวจวัดความล่าช้าและความสูญเสียนั้นได้ใช้ขั้นตอนการวิเคราะห์ร่วมกัน เพื่อทดลองความสามารถของเครื่องมือวิจัยในการทำงานร่วมกันของขั้นตอนนี้ จึงออกแบบการทดลองเพิ่ม โดยกำหนดเวลาหน่วงและความสูญเสีย ดังแสดงได้ในตารางที่ 7

ตารางที่ 7 การกำหนด HZ OPTION ในแต่ละชุดการทดลอง

ชุดการทดลองที่	เวลาหน่วงบนตัวระบบ	ความสูญเสีย บนตัวระบบ
1	30	5
2	70	10
3	100	15
4	500	30
5	700	40

แต่ละชุดการทดลองจะใช้ความสามารถของ ipfw ซึ่งสามารถกำหนดเวลาหน่วงและความสูญเสียพร้อมกันได้บนท่อการสื่อสาร(pipe)เดียวกัน ด้วยรูปแบบคำสั่งดังนี้

ipfw add pipe 1 ip from any to any in

ipfw pipe 1 config delay <n>ms plr<n>

ทำการทดลองโดยส่งข้อมูลชนิด ICMP ด้วยคำสั่ง Ping ชุดละ 500 แพ็กเก็ต เพื่อเปรียบเทียบผลการวัดระหว่างคำสั่ง Ping และเครื่องมือวิจัย ผลการทดลองได้คุณลักษณะที่สามารถแสดงได้ในตารางที่ 8 ดังนี้

ตารางที่ 8 ผลการวัดความล่าช้าและความสูญเสียพร้อมกัน

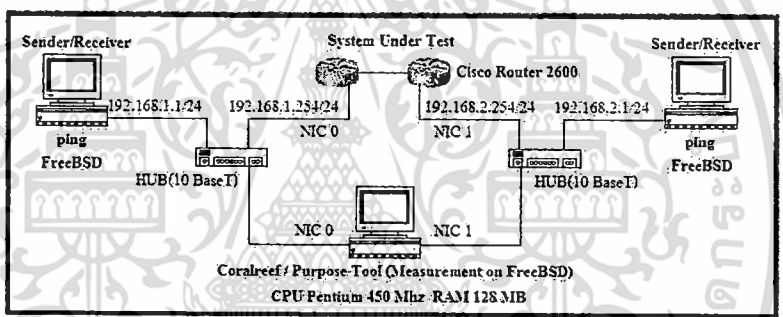
การทดลอง		ความล่าช้า		ความสูญเสีย			
Emulated Delay	Emulated Loss	Purpose (Diff)	Ping (Diff)	Packets Received		Purpose Loss(%)	Ping (%)
				Forward (packets)	Reverse (packets)		
30	5	0.145016	2.349908	473	449	10.20	10
50	10	0.143041	2.297498	450	412	17.60	17
100	15	0.142332	2.283647	425	370	26.00	26
500	30	0.131296	2.393837	336	230	54.00	54
700	40	0.127853	2.275691	302	170	66.00	66

ผลการวัดด้วยเครื่องมือวิจัยจะได้ผลต่างของค่าเฉลี่ยความล่าช้าที่ใกล้เคียงกับ Emulated Delay มากกว่า คำสั่ง Ping โดยวัดค่าเฉลี่ยของผลต่างจากเครื่องมือวิจัยได้ 0.137908 ms ส่วนคำสั่ง Ping วัดได้ 2.320116 ms และเมื่อเปรียบเทียบค่าความแม่นยำเครื่องมือวิจัยวัดได้ 0.001913% ส่วนคำสั่ง Ping วัดได้ 0.031031% เมื่อเปรียบเทียบกับค่าความล่าช้าที่กำหนดให้กับตัวระบบ ส่วนผลการวิเคราะห์ความสูญเสีย ทั้งสองเครื่องมือสามารถวัดได้ไม่แตกต่างกัน โดยเครื่องมือวิจัยสามารถวัดความสูญเสียที่เกิดขึ้นได้ทั้งขาไปและกลับจากจำนวนข้อมูลที่สูญเสียหายในแต่ละทิศทาง และหากปัดทศนิยมตามค่าที่วัดได้จากคำสั่ง Ping ก็จะได้ค่าที่ตรงกันคือ 10, 17, 26, 54 และ 66 เปอร์เซนต์ตามลำดับ

4.7 ผลการทดลองเพื่อวิเคราะห์ความถูกต้องของการตรวจจับเวลากับอุปกรณ์ชนิดอื่น

จากทดลองโดยใช้เครื่องมือวิจัยตรวจวัดเวลาตามพารามิเตอร์ต่าง ๆ เช่น ตรวจวัดเวลาเวลาไปกลับ หรือ ความล่าช้า ผลการทดลองพบว่าเมื่อนำไปคำนวณและวาดกราฟด้วยผลต่างจากค่าเฉลี่ยแล้วลักษณะของกราฟแต่ละเส้นที่ตรวจวัดได้จากทุกเครื่องมือเปรียบเทียบกันแล้วได้ค่าใกล้เคียงกัน แต่มีข้อสังเกตอีกปัจจัยหนึ่งคือ ผลต่างจากค่าเฉลี่ยไม่ได้มีลักษณะเป็นแบบสุ่ม กล่าวคือผลต่างจากค่าเฉลี่ยในทุกชุดของการทดลองน่าจะมีค่าใกล้เคียงกัน ดังนั้นเพื่อหาข้อสรุปในส่วนนี้ จึงได้ออกแบบการทดลองเพิ่มขึ้นเพื่อเปรียบเทียบผลต่างจากค่าเฉลี่ยที่ได้จากกรณีอื่น ดังนี้

4.7.1 ออกแบบการทดลองด้วยการทดสอบเปลี่ยนตัวระบบใหม่เป็นอุปกรณ์ Cisco Router 2600 จำนวน 2 ตัว ต่อเชื่อมกัน เปรียบเทียบกับการรับส่งข้อมูลตามแบบจำลองเดิม ดังแสดงได้ในภาพที่ 56



รูปที่ 56 การทดลองกับแบบจำลองที่ตัวระบบเป็นอุปกรณ์ Cisco Router

แบบจำลองใหม่จะเหมือนกับแบบจำลองแรก แตกต่างกันเฉพาะในส่วนของตัวระบบที่ออกแบบให้ใช้กับอุปกรณ์ Cisco Router จำนวน 2 ตัวต่อเชื่อมกันผ่านช่องสัญญาณ Serial ทำการทดลองวัดความล่าช้า ด้วยการรับส่งข้อมูลระหว่างคอมพิวเตอร์ที่อยู่ฝั่งเครือข่ายภายในไปยังเครื่องที่อยู่ฝั่งเครือข่ายภายนอก ด้วยข้อมูลชุดเดียวกันกับทั้งสองแบบจำลอง โดยแบ่งการทดลองออกเป็น 8 ชุด แต่ละชุดรับส่งด้วยข้อมูลชนิด ICMP ที่มีความยาวขนาด 64, 128, 256, 512, 768, 1024, 1200 และ 1500 ไบต์ ตามลำดับ นอกจากนี้ในแต่ละชุดได้แบ่งการทดลองออกเป็น 11 ชุดย่อย โดยปรับเปลี่ยนอัตราการรับส่งข้อมูลที่ตัวระบบให้ส่งข้อมูลชุดย่อยจำนวน 300 แพ็กเก็ต ด้วยความเร็ว 64000, 72000, 115200, 128000, 148000, 192000, 250000, 384000, 500000, 768000 และ 1000000 bit/sec ตามลำดับ โดยมีรูปแบบคำสั่งปฏิบัติการที่ตัวระบบเป็นอุปกรณ์ Router หรือ FreeBSD ดังนี้

- อุปกรณ์ Router กำหนดอัตราการรับส่งที่ช่องสัญญาณ Serial ด้วยรูปแบบ

clock rate <transmission rate>

โดยที่ <transmission rate> หมายถึง อัตราการรับส่งข้อมูล มีหน่วยเป็น บิตต่อวินาที

- เครื่องคอมพิวเตอร์ที่ติดตั้งระบบปฏิบัติการ FreeBSD กำหนดด้วยรูปแบบ

ipfw add pipe 1 ip from any to any in

ipfw pipe 1 config bw <n>Kbit/sec

โดยที่ <n> หมายถึง อัตราการรับส่งข้อมูลมีหน่วยเป็น กิโลบิตต่อวินาที

ดังนั้นการทดลองในครั้งนี้มีทั้งสิ้น 88 ชุดการทดลองย่อยโดยสร้างความแปรผันระหว่าง ขนาดของแพ็กเก็ต และอัตราในการรับส่งข้อมูลระหว่างทั้งสองแบบจำลอง ดังแสดงได้ในตารางที่ 7

ตารางที่ 9 รูปแบบการทดลองเพื่อเปรียบเทียบกับตัวระบบชนิดอื่น

ลำดับ	แบบจำลองที่ตัวระบบเป็น FreeBSD		แบบจำลองที่ตัวระบบเป็นอุปกรณ์ Router	
	ความยาวแพ็กเก็ต (bytes)	อัตราการรับส่ง (/sec)	ความยาวแพ็กเก็ต (bytes)	อัตราการรับส่ง (bit/sec)
1	64	64000	64	64000
...	64	72000	64	72000
	...	...	...	...
11	64	1000000	64	1000000
12	128	64000	128	64000
...	128	72000	128	72000
	...	...	...	...
88	1500	1000000	1500	1000000

ผลการตรวจวัดความล่าช้าในแต่ละชุดจะถูกนำไปวาดกราฟ x, y 2 แบบ คือ

แบบที่ 1 แสดงความสัมพันธ์ของความล่าช้าที่ตรวจวัดได้จากการแปรผันตามความยาวแพ็กเก็ตและอัตราการรับส่งข้อมูล โดยให้แกน y คือ ความล่าช้าที่ตรวจวัดได้จากเครื่องมือวัดในงานวิจัย ส่วนแกน x คือ อัตราความล่าช้าเสมือน หรือนิยามเรียกว่า Emulated Transmission Delay ที่คำนวณได้จากสมการ

$$T\_delay_i = \frac{(length_i + Network\ Header)\ x\ 8\ x 1000}{bandwidth_i} \qquad (9)$$

โดยที่ t_delay_i หมายถึง อัตราความล่าช้าเสมือนที่ i มีหน่วยเป็นมิลลิวินาที

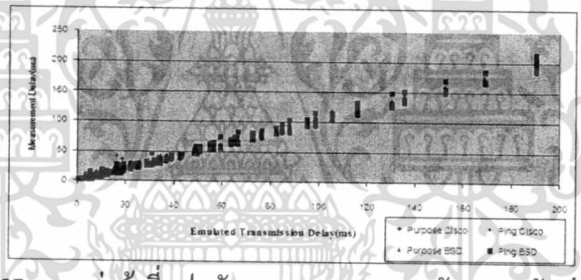
$length_i$ หมายถึง ความยาวแพ็กเก็ต

$length_i$ หมายถึง ความยาวส่วนหัวของแพ็กเก็ต กรณีที่เป็น ICMP จะมีค่าเท่ากับ 20 ไบต์

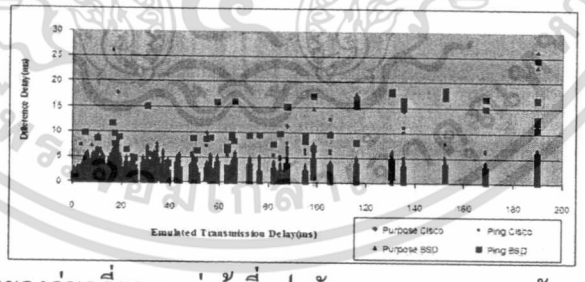
$length_i$ หมายถึง อัตราการรับส่งข้อมูลที่กำหนดให้ตัวระบบ มีหน่วยเป็นบิตต่อวินาที

แบบที่ 2 แสดงความสัมพันธ์ผลต่างของค่าเฉลี่ยจาก Emulated Transmission Delay ที่แปรผันตามความยาวแพ็กเก็ตและอัตราการรับส่งข้อมูล โดยให้แกน y คือ ผลต่างของค่าเฉลี่ยที่ตรวจวัดได้จากเครื่องมือวัดในงานวิจัย ส่วนแกน x คือ อัตราความล่าช้าเสมือน

เมื่อทดลองตรวจวัด โดยกำหนด HZ OPTION ที่ Kernel บนระบบปฏิบัติการ FreeBSD ที่ติดตั้งเครื่องมือวิจัยเป็น 5000 Hz ผลการทดลองสามารถแสดงได้ในภาพที่ 57 และ 58



ภาพที่ 57 ความล่าช้าที่แปรผันตามความยาวและอัตราการรับส่งข้อมูล



ภาพที่ 58 ผลต่างของค่าเฉลี่ยความล่าช้าที่แปรผันตามความยาวและอัตราการรับส่งข้อมูล

จากภาพที่ 57 ผลการทดลองพบว่าการตรวจวัดข้อมูลที่ได้จากคำสั่ง ping และ เครื่องมือวิจัย ทั้งแบบจำลองเดิมที่ตัวระบบเป็น FreeBSD และตามแบบจำลองใหม่ที่ตัวระบบเป็นอุปกรณ์ Router ให้ผลลัพธ์การตรวจวัดความล่าช้าได้ไม่แตกต่างกัน โดยมีลักษณะของกราฟที่เป็นไปตามทฤษฎีคือ เมื่อความยาวของแพ็กเก็ตเพิ่มขึ้น ย่อมส่งผลให้ความล่าช้าในการรับส่งข้อมูล ใช้เวลามากขึ้นตามไปด้วย

และจากภาพที่ 58 ผลต่างของค่าเฉลี่ยจาก Emulated Transmission Delay ที่ตรวจวัดได้จากทั้งสองแบบจำลองก็ไม่แตกต่างเช่นเดียวกัน โดยผลลัพธ์ที่ได้จากการทดลองที่ตัวระบบเป็นอุปกรณ์ Router ผลต่างของค่าเฉลี่ยยังคงแปรผันโดยเพิ่มตามความล่าช้าในการรับส่งข้อมูล ทั้งที่ควรจะมีลักษณะเป็นแบบสุ่ม ดังนั้นจึงสรุปได้ว่าอุปกรณ์อื่น ๆ ที่เป็นตัวระบบยังคงให้ผลลัพธ์เช่นเดียวกับเครื่องคอมพิวเตอร์ที่จำลองเป็นตัวระบบ

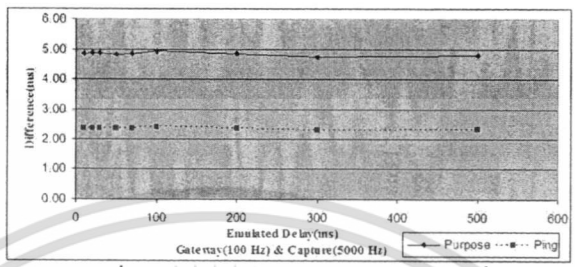
4.7.2 ออกแบบการทดลองโดยใช้แบบจำลองเดิมเพื่อเปรียบเทียบการรับส่งข้อมูลผ่านตัวระบบที่แปรผันตาม HZ OPTION ใน Kernel ของระบบปฏิบัติการ FreeBSD โดยแบ่งการทดลองออกเป็น 12 ชุด แต่ละชุดรับส่งด้วยข้อมูลชนิด ICMP ที่มีความยาวขนาด 84 ไบต์ เท่ากัน การทดลองในแต่ละชุดจะกำหนดความแปรผัน HZ OPTION บนตัวระบบและคอมพิวเตอร์ที่ติดตั้งเครื่องมือวิจัยให้มีค่าเป็นไปตามลำดับ ดังแสดงได้ในตารางที่ 8

ตารางที่ 10 การกำหนด HZ OPTION ในแต่ละชุดการทดลอง

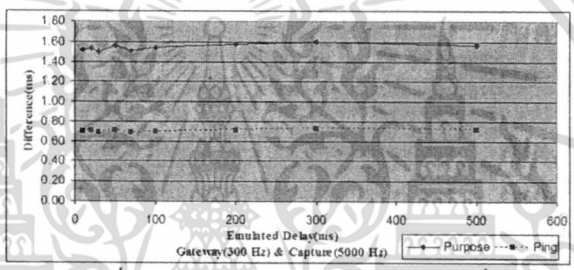
ชุดการทดลองที่	ตัวระบบ(Hz)	เครื่องที่ติดตั้งเครื่องมือวิจัย(Hz)	แปรผันตาม
1	100	5000	ตัวระบบ
2	300	5000	ตัวระบบ
3	500	5000	ตัวระบบ
4	1000	5000	ตัวระบบ
5	3000	5000	ตัวระบบ
6	5000	5000	ตัวระบบ
7	1000	100	เครื่องมือวิจัย
8	1000	300	เครื่องมือวิจัย
9	1000	500	เครื่องมือวิจัย
10	1000	1000	เครื่องมือวิจัย
11	1000	3000	เครื่องมือวิจัย
12	1000	5000	เครื่องมือวิจัย

ในการทดลองแต่ละชุดได้แบ่งการทดลองออกเป็น 9 ชุดย่อย โดยการปรับเปลี่ยนเวลาหน่วงในการรับส่งข้อมูลที่ตัวระบบให้ส่งข้อมูลจำนวน 500 แพ็กเก็ต ด้วยเวลาหน่วงในแต่ละชุดเท่ากับ 10,

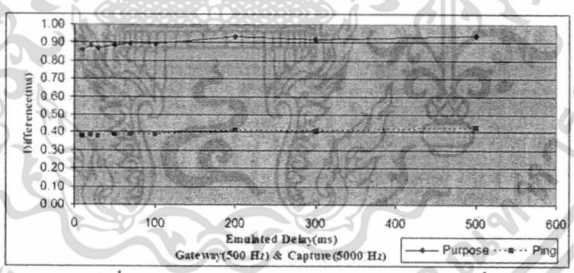
20, 30, 50, 70, 100, 200, 300 และ 500 ms ตามลำดับ ผลการทดลองสามารถแสดงได้ในภาพที่ 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70 และ ตารางที่ 9, 10, 11, 12 ดังนี้



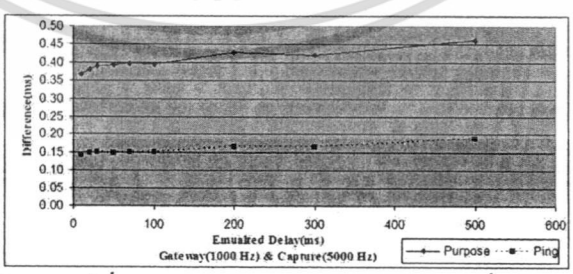
ภาพที่ 59 ผลการวิเคราะห์เมื่อกำหนดตัวระบบ 100 Hz และ เครื่องมือวิเคราะห์ 5000 Hz



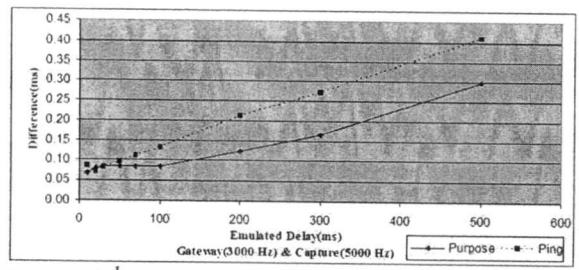
ภาพที่ 60 ผลการวิเคราะห์เมื่อกำหนดตัวระบบ 300 Hz และ เครื่องมือวิเคราะห์ 5000 Hz



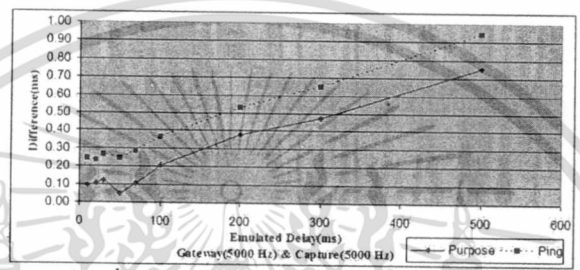
ภาพที่ 61 ผลการวิเคราะห์เมื่อกำหนดตัวระบบ 500 Hz และ เครื่องมือวิเคราะห์ 5000 Hz



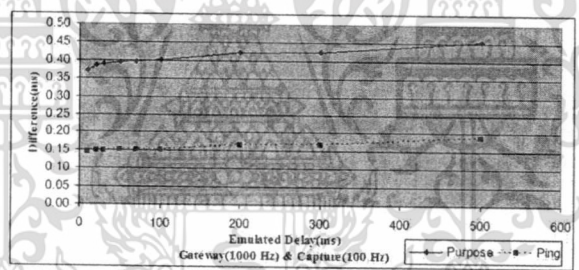
ภาพที่ 62 ผลการวิเคราะห์เมื่อกำหนดตัวระบบ 1000 Hz และ เครื่องมือวิเคราะห์ 5000 Hz



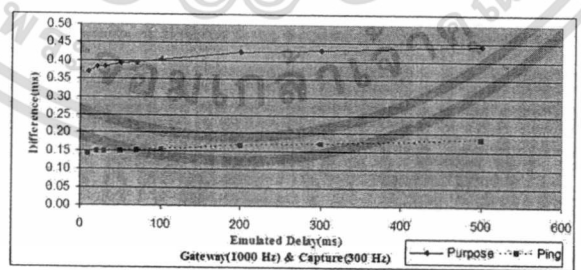
ภาพที่ 63 ผลการวิเคราะห์เมื่อกำหนดตัวระบบ 3000 Hz และ เครื่องมือวิเคราะห์ 5000 Hz



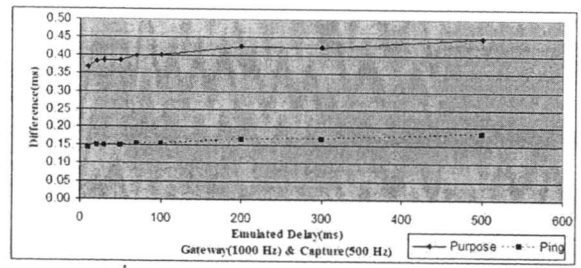
ภาพที่ 64 ผลการวิเคราะห์เมื่อกำหนดตัวระบบ 5000 Hz และ เครื่องมือวิเคราะห์ 5000 Hz



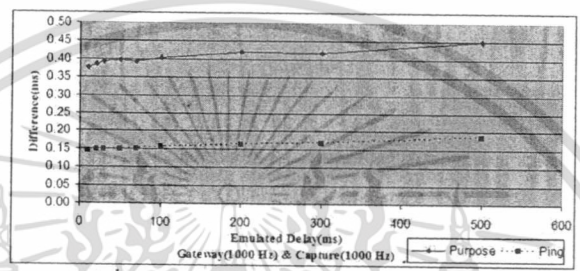
ภาพที่ 65 ผลการวิเคราะห์เมื่อกำหนดตัวระบบ 1000 Hz และ เครื่องมือวิเคราะห์ 100 Hz



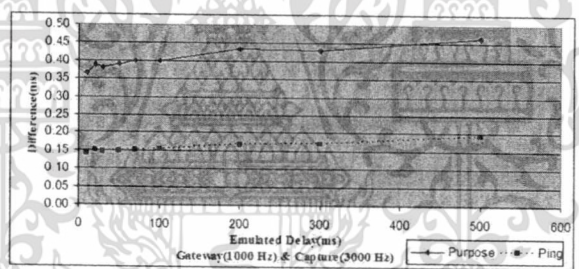
ภาพที่ 66 ผลการวิเคราะห์เมื่อกำหนดตัวระบบ 1000 Hz และ เครื่องมือวิเคราะห์ 300 Hz



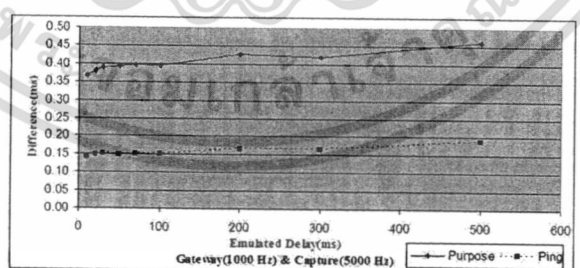
ภาพที่ 67 ผลการวิเคราะห์เมื่อกำหนดตัวระบบ 1000 Hz และ เครื่องมือวิเคราะห์ 500 Hz



ภาพที่ 68 ผลการวิเคราะห์เมื่อกำหนดตัวระบบ 1000 Hz และ เครื่องมือวิเคราะห์ 1000 Hz



ภาพที่ 69 ผลการวิเคราะห์เมื่อกำหนดตัวระบบ 1000 Hz และ เครื่องมือวิเคราะห์ 3000 Hz



ภาพที่ 70 ผลการวิเคราะห์เมื่อกำหนดตัวระบบ 1000 Hz และ เครื่องมือวิเคราะห์ 5000 Hz

ตารางที่ 11 ค่าเฉลี่ยความล่าช้าที่ตรวจวัดได้จากการแปรผัน HZ OPTION บนตัวระบบ

เวลาหน่วง	ตัวระบบ 100 Hz และหน่วยตรวจจับ 5000 Hz		ตัวระบบ 300 Hz และหน่วยตรวจจับ 5000 Hz		ตัวระบบ 500 และ หน่วยตรวจจับ 5000Hz		ตัวระบบ 1000 และ หน่วยตรวจจับ 5000Hz		ตัวระบบ 3000 และ หน่วยตรวจจับ 5000Hz		ตัวระบบ 5000 และ หน่วยตรวจจับ 5000Hz	
	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง Ping	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง ping
10	5.139404	7.639737	8.486616	9.313160	9.146756	9.644047	9.645050	9.893395	9.995670	10.084209	10.092584	10.241043
20	15.123036	17.633155	18.475108	19.308382	19.123426	19.633406	19.633910	19.889822	19.963464	20.071233	20.079420	20.235085
30	25.136032	27.638278	28.507104	29.325100	29.139676	29.642608	29.622952	29.885976	29.978516	30.081947	30.120310	30.264316
50	45.165418	47.655849	48.450648	49.298256	49.118868	49.632076	49.620570	49.886687	49.992700	50.096601	50.046994	50.242377
70	65.139636	67.642711	68.503580	69.325855	69.108404	69.628158	69.616186	69.885266	70.007618	70.112250	70.104458	70.286998
100	95.054910	97.601487	98.467664	99.309034	99.113030	99.631604	99.620778	99.884880	100.025472	100.132237	100.204446	100.362324
200	195.150398	197.650639	198.430676	199.291854	199.067572	199.602962	199.579356	199.857533	200.115256	200.213297	200.377880	200.526473
300	295.255916	297.706788	298.408238	299.276034	299.081210	299.601840	299.586590	299.855652	300.166630	300.271787	300.470384	300.649426
500	495.198868	497.681393	498.434676	499.287515	499.063032	499.580221	499.538488	499.819774	500.297776	500.410076	500.750582	500.942364

ตารางที่ 12 ค่าเฉลี่ยความล่าช้าที่ตรวจวัดได้จากการแปรผัน HZ OPTION บนหน่วยตรวจจับข้อมูล

เวลาหน่วง	ตัวระบบ 1000 Hz และหน่วยตรวจจับ 100 Hz		ตัวระบบ 1000 Hz และหน่วยตรวจจับ 300 Hz		ตัวระบบ 1000 และ หน่วยตรวจจับ 500Hz		ตัวระบบ 1000 และ หน่วยตรวจจับ 1000Hz		ตัวระบบ 1000 และ หน่วยตรวจจับ 3000Hz		ตัวระบบ 1000 และ หน่วยตรวจจับ 5000Hz	
	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง Ping	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง ping
10	9.642846	9.893820	9.642040	9.893654	9.645286	9.894571	9.636054	9.890296	9.646546	9.894518	9.645050	9.893395
20	19.628538	19.888424	19.632592	19.889103	19.630016	19.888184	19.628376	19.887745	19.626344	19.885755	19.633910	19.889822
30	29.621768	29.886656	29.633266	29.891208	29.631102	29.889827	29.621666	29.886397	29.634480	29.891900	29.622952	29.885976
50	49.619794	49.886401	49.622022	49.887402	49.629178	49.890689	49.617440	49.885501	49.625248	49.888665	49.620570	49.886687
70	69.617726	69.884534	69.619566	69.885570	69.612500	69.881908	69.621320	69.887710	69.613600	69.882632	69.616186	69.885266
100	99.611228	99.880533	99.607180	99.877736	99.611382	99.879561	99.605998	99.878704	99.611380	99.880055	99.620778	99.884880
200	199.584272	199.861368	199.580554	199.858454	199.579992	199.859766	199.585436	199.861326	199.573396	199.856187	199.579356	199.857533
300	299.581110	299.850777	299.576010	299.849033	299.583032	299.852116	299.588384	299.855791	299.575866	299.848601	299.586590	299.855652
500	499.547318	499.822940	499.558452	499.828466	499.554512	499.825791	499.550782	499.825116	499.537190	499.817620	499.538488	499.819774

ตารางที่ 13 ผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay ที่ตรวจวัดได้จากการแปรผัน HZ OPTION บนตัวระบบ

เวลาหน่วง	ตัวระบบ 100 Hz และหน่วยตรวจจับ 5000 Hz		ตัวระบบ 300 Hz และหน่วยตรวจจับ 5000 Hz		ตัวระบบ 500 และ หน่วยตรวจจับ 5000Hz		ตัวระบบ 1000 และ หน่วยตรวจจับ 5000Hz		ตัวระบบ 3000 และ หน่วยตรวจจับ 5000Hz		ตัวระบบ 5000 และ หน่วยตรวจจับ 5000Hz	
	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง Ping	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง ping
10	4.862608	2.364493	1.516744	0.697754	0.861612	0.376445	0.367438	0.142173	0.067250	0.086461	0.092764	0.241043
20	4.876964	2.369085	1.529808	0.704334	0.882406	0.384714	0.380694	0.149032	0.080748	0.071623	0.104508	0.235085
30	4.866052	2.365992	1.497920	0.688794	0.869040	0.377758	0.390932	0.151128	0.083080	0.082059	0.120310	0.264316
50	4.836234	2.348319	1.554060	0.713124	0.887896	0.385936	0.393198	0.149543	0.084048	0.096601	0.047134	0.242377
70	4.861824	2.361735	1.502148	0.688467	0.896940	0.388176	0.397762	0.151720	0.084830	0.112250	0.104458	0.286998
100	4.945890	2.402037	1.536208	0.701938	0.894034	0.386194	0.393790	0.152080	0.084644	0.132237	0.204446	0.362324
200	4.851386	2.353373	1.572200	0.718812	0.935336	0.409044	0.426616	0.166353	0.124652	0.213297	0.377880	0.526473
300	4.745592	2.296830	1.594942	0.733264	0.922230	0.408760	0.420730	0.166004	0.166702	0.271787	0.470384	0.649426
500	4.803052	2.323581	1.568252	0.722209	0.938624	0.426533	0.463052	0.190122	0.297776	0.410076	0.750582	0.942364

ตารางที่ 14 ผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay ที่ตรวจวัดได้จากการแปรผัน HZ OPTION บนหน่วยตรวจจับข้อมูล

เวลาหน่วง	ตัวระบบ 1000 Hz และหน่วยตรวจจับ 100 Hz		ตัวระบบ 1000 Hz และหน่วยตรวจจับ 300 Hz		ตัวระบบ 1000 และ หน่วยตรวจจับ 500Hz		ตัวระบบ 1000 และ หน่วยตรวจจับ 1000Hz		ตัวระบบ 1000 และ หน่วยตรวจจับ 3000Hz		ตัวระบบ 1000 และ หน่วยตรวจจับ 5000Hz	
	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง Ping	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง ping	เครื่องมือวิจัย	คำสั่ง ping
10	0.370642	0.144052	0.369828	0.142868	0.366798	0.142881	0.375030	0.144378	0.365254	0.141908	0.367438	0.142173
20	0.385390	0.149564	0.382604	0.149781	0.383744	0.149984	0.385856	0.150591	0.387300	0.150999	0.380694	0.149032
30	0.391204	0.150156	0.383570	0.148848	0.385362	0.150287	0.392042	0.150567	0.381000	0.147610	0.390932	0.151128
50	0.395250	0.151315	0.392898	0.150532	0.385982	0.148375	0.396168	0.150467	0.389692	0.149707	0.393198	0.149543
70	0.396126	0.151500	0.394718	0.151622	0.400188	0.153476	0.393268	0.150754	0.398452	0.151826	0.397762	0.151720
100	0.400736	0.152949	0.404212	0.154660	0.401010	0.153463	0.405090	0.155782	0.400584	0.153967	0.393790	0.152080
200	0.423820	0.165202	0.426066	0.165954	0.426328	0.165024	0.420732	0.164250	0.434068	0.169249	0.426616	0.166353
300	0.425298	0.169261	0.429842	0.170395	0.423332	0.168122	0.419384	0.167219	0.430038	0.170907	0.420730	0.166004
500	0.455038	0.188372	0.443792	0.183474	0.448000	0.185493	0.451282	0.186040	0.464574	0.192860	0.463052	0.190122

จากภาพและตาราง ค่าที่ตรวจวัดได้จากการทดลองทุกชุดพบว่าการทดลองที่กำหนดความแปรผัน HZ OPTION ให้กับตัวระบบและเครื่องมือวิจัยมีผลต่อการตรวจวัดเวลาในทุกเครื่องมือ โดยผลการตรวจวัดระหว่างเครื่องมือวิจัยและคำสั่ง Ping ให้ผลลัพธ์ที่มีความใกล้เคียงกันทั้งค่าเฉลี่ยและผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay) เมื่อนำไปวาดกราฟจะพบว่าความแปรผันในแต่ละจุดจะเหมือนกัน โดยค่าเฉลี่ยที่ตรวจวัดได้จากเครื่องมือวิจัยจะมีค่าต่ำกว่าการตรวจวัดด้วยคำสั่ง Ping ในทุกจุด ในขณะที่ผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay ของคำสั่ง Ping จะมีค่าที่ต่ำกว่าในกรณีที่กำหนด HZ OPTION บนตัวระบบหรือเครื่องมือวิจัยต่ำกว่า 3000 Hz

เมื่อวิเคราะห์ตามการแปรผันบนเครื่องมือวิจัยจะพบว่าการปรับค่า HZ OPTION ผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay มีค่าใกล้เคียงตามกันในทุกจุด เช่น ในช่วงการหน่วงเวลา 500 มิลลิวินาที เครื่องมือวิจัยและคำสั่ง Ping สามารถตรวจวัดได้ตามลำดับดังนี้ ช่วง 100 Hz วัดได้ 0.455038 และ 0.188372 ช่วง 300 Hz วัดได้ 0.443792 และ 0.183474 ช่วง 500 Hz วัดได้ 0.448000 และ 0.185493 ช่วง 1000 Hz วัดได้ 0.451282 และ 0.186040 ช่วง 3000 Hz วัดได้ 0.464574 และ 0.192860 ช่วง 5000 Hz วัดได้ 0.463052 และ 0.190122 มิลลิวินาที

ส่วนการวิเคราะห์ตามการแปรผันบนตัวระบบจะพบว่าการปรับค่า HZ OPTION ผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay จะมีความแตกต่างกันในทุกจุด และมีผลเป็นตรงข้ามกันระหว่างการปรับ HZ OPTION กับผลต่างของค่าเฉลี่ยความล่าช้า คือ เมื่อปรับค่า HZ OPTION สูงขึ้นความแตกต่างของค่าเฉลี่ยก็จะมีลดลง เช่น ในช่วงการหน่วงเวลา 500 มิลลิวินาที เครื่องมือวิจัยและคำสั่ง Ping สามารถตรวจวัดได้ตามลำดับดังนี้ ช่วง 100 Hz วัดได้ 4.803052 และ 2.323581 ช่วง 300 Hz วัดได้ 1.568252 และ 0.722209 ช่วง 500 Hz วัดได้ 0.938624 และ 0.426533 ช่วง 1000 Hz วัดได้ 0.463052 และ 0.190122 ช่วง 3000 Hz วัดได้ 0.297776 และ 0.410076 ช่วง 5000 Hz วัดได้ 0.750582, 0.942364 มิลลิวินาที

ดังนั้นการปรับค่า HZ OPTION บนเครื่องมือวิจัยจะไม่มีผลต่อความเปลี่ยนแปลงในการตรวจวัดผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay แต่กลับมีผลต่อการปรับค่า HZ OPTION บนตัวระบบ และเมื่อพิจารณาจากกราฟจะพบว่าผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay จะมีลักษณะเป็นแบบสุ่มเมื่อกำหนดค่า HZ OPTION บนตัวระบบในช่วง 100 Hz และมีลักษณะเพิ่มความชันขึ้นเมื่อกำหนด HZ OPTION ในช่วงตั้ง 1000 Hz ขึ้นไป

บทที่ 5
สรุปผลการวิจัย

5.1 สรุปผลการทดลอง

งานวิจัยนี้ได้นำเสนอการออกแบบวิธีตรวจวัดคุณลักษณะการสื่อสารข้อมูลบนเครือข่ายอินเทอร์เน็ต โดยออกแบบให้ตรวจวัดค่าพารามิเตอร์ต่าง ๆ ได้แก่ ความยาวแพ็กเก็ต, เวลาระหว่างการมา, เวลาการสื่อสารข้อมูลไปกลับ, ความล่าช้าผ่านตัวระบบ และ ความสูญเสียข้อมูลบนตัวระบบ ผลการออกแบบและการตรวจวัดข้อมูลได้ทำการเปรียบเทียบกับเครื่องมือชนิดอื่น โดยได้ข้อสรุปจากการวัดพารามิเตอร์ต่าง ๆ ดังนี้

5.1.1 ผลการตรวจวัดความยาวและเวลาระหว่างการมาเปรียบเทียบกับเครื่องมือวิจัยที่พัฒนาจาก Corallib Library ตรวจวัดความยาวของทุกแพ็กเก็ตจากทั้ง 2 เครื่องมือได้ 724.18 ไบต์เท่ากัน ค่าเปอร์เซนต์เฉลี่ยของความแตกต่างระหว่างเครื่องมือเป็น 0.00 ไบต์ และเปรียบเทียบด้วย Confidence Interval(95%) ได้ค่าเฉลี่ยของการตรวจวัดอยู่ในช่วงเดียวกันคือ (713.25,735.12) ไบต์ ส่วนผลการตรวจวัดเวลาระหว่างการมา เมื่อตรวจวัดด้วย Corallib และเครื่องมือวิจัยได้ ค่าเฉลี่ยเวลาของ Corallib และ เครื่องมือวิจัยคือ 612.34 และ 612.35 Microseconds ตามลำดับ ค่าเปอร์เซนต์เฉลี่ยของความแตกต่างระหว่างเครื่องมือคือ 0.000021 Microseconds เมื่อเปรียบเทียบด้วย Confidence Interval(95%) ได้ค่าเฉลี่ยของการตรวจวัดอยู่ในช่วง (603.30, 621.38) Microseconds และ (603.31,621.39) Microseconds

5.1.2 ผลการตรวจวัดเวลาการสื่อสารข้อมูลไปกลับ เครื่องมือวิจัยได้ออกแบบให้ตรวจวัดกับข้อมูล 2 ชนิด คือ

5.1.2.1 ข้อมูล ICMP ตรวจวัดโดยเปรียบเทียบกับคำสั่ง Ping สามารถตรวจวัดผลต่างของค่าเฉลี่ยเวลาไปกลับจาก Emulated Delay ด้วยคำสั่ง ping ได้ 1.276191 ms ส่วนเครื่องมือวิจัยให้ค่าความแตกต่างที่ 1.090328 ms ซึ่งน้อยกว่าการตรวจวัดด้วยคำสั่ง ping ซึ่งความแตกต่างนี้เป็นผลมาจากทั้งสองเครื่องมือมีการกำกับเวลาของข้อมูลเพื่อนำไปใช้ในการคำนวณบนชั้นเครือข่ายที่แตกต่างกัน

5.1.2.2 ข้อมูล TCP ตรวจวัดโดยเปรียบเทียบระหว่างขั้นตอนการเชื่อมต่อ(Connection) และ ปิดการเชื่อมต่อ สามารถตรวจวัดผลต่างของค่าเฉลี่ยเวลาไปกลับจาก Emulated Delay ในช่วงการเชื่อมต่อคือ 1.152212 ms ส่วนช่วงเวลาปิดการเชื่อมต่อให้ความแตกต่างที่ 1.276668 ms ซึ่งมีค่ามากกว่าในช่วงเวลาการเชื่อมต่อ ซึ่งความแตกต่างเหล่านี้เป็นผลมาจากช่วงเวลาปิดการเชื่อมต่อของฝ่ายรับและส่งอาจต้องสูญเสียเวลาบางส่วนไปสำหรับเรียบเรียงข้อมูลก่อนปิดการเชื่อมต่อ

5.1.3 ผลการตรวจวัดความล่าช้า ออกแบบการทดลองโดยการวัดค่าเวลาข้อมูล 3 ชนิด คือ ICMP, TCP และ UDP เปรียบเทียบกับการหน่วงเวลาบนตัวระบบ ผลการตรวจวัดข้อมูลทั้ง 3 ชนิด ได้ค่าต่าง ๆ ดังนี้ ผลต่างของค่าเฉลี่ยจาก Emulated Delay ที่ตรวจวัดได้จากข้อมูลชนิด ICMP, TCP และ UDP คือ 1.582579, 1.630671 และ 1.585756 ms ตามลำดับ ซึ่งแพ็กเก็ตชนิด TCP จะมีค่าความแตกต่างของเวลามากกว่า 2 ชนิดแรกเนื่องจากมีความยาวของส่วนหัวมากกว่า ICMP และ UDP

5.1.4 ผลการตรวจวัดความสูญเสีย ออกแบบการทดลองโดยตรวจจับการรับส่งข้อมูลด้วยเครื่องมือวิจัย และคำสั่ง Ping เปรียบเทียบกับค่าความสูญเสียที่กำหนดให้ตัวระบบ สามารถตรวจวัดอัตราความสูญเสีย(Loss Rate)ได้เป็น 0, 7, 22, 42, 54, 74, 86 และ 99 เปอร์เซนต์ตามลำดับ ซึ่งเป็นค่าที่ใกล้เคียงกับความสูญเสียที่กำหนดให้กับตัวระบบ

5.1.5 การตรวจวัดความล่าช้าและความสูญเสียพร้อมกัน ออกแบบการทดลองโดยเปรียบเทียบผลการวัดระหว่างเครื่องมือวิจัยและคำสั่ง Ping โดยกำหนดชุดการทดลองที่แปรผันตัวระบบทั้งเวลาหน่วงและความสูญเสียไปพร้อมกัน ผลการวัดได้ค่าเฉลี่ยของผลต่างจากเครื่องมือวิจัยเป็น 0.137908 ms และได้จากคำสั่ง Ping เป็น 2.320116 ms ส่วนความสูญเสียสามารถวัดได้ตรงกันคือ 10, 17, 26, 54 และ 66 เปอร์เซนต์ตามลำดับ

5.1.6 การตรวจวัดค่าเวลากับตัวระบบที่เป็นอุปกรณ์อื่น ออกแบบการทดลองโดยตรวจวัดค่าความล่าช้าจากอุปกรณ์ Router เพื่อเปรียบเทียบกับผลที่ได้จากการตรวจวัดโดยแบบจำลองแรก ผลคือการทดลองกับตัวระบบทั้งสองชนิดให้ค่าที่มีลักษณะใกล้เคียงกันและพบว่าการปรับเปลี่ยน HZ OPTION จะมีผลกระทบต่อการวัดค่า ดังนั้นจึงออกแบบการทดลองอีกครั้งโดยจะแปรผันตาม HZ OPTION บนตัวระบบและเครื่องมือวิจัยเปรียบเทียบกัน ซึ่งผลการวัดได้ลักษณะข้อมูลดังนี้

5.1.6.1 เมื่อปรับค่า HZ OPTION บนเครื่องมือวิจัยจะให้ผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay มีค่าใกล้เคียงตามกันในทุกจุด เช่น ในช่วงการหน่วงเวลา 500 มิลลิวินาที เครื่องมือวิจัยและคำสั่ง Ping สามารถตรวจวัดได้ตามลำดับดังนี้ ช่วง 100 Hz วัดได้ 0.455038 และ 0.188372 ช่วง 300 Hz วัดได้ 0.443792 และ 0.183474 ช่วง 500 Hz วัดได้ 0.448000 และ 0.185493 ช่วง 1000 Hz วัดได้ 0.451282 และ 0.186040 ช่วง 3000 Hz วัดได้ 0.464574 และ 0.192860 ช่วง 5000 Hz วัดได้ 0.463052 และ 0.190122 มิลลิวินาที

5.1.6.2 เมื่อปรับค่า HZ OPTION บนตัวระบบผลต่างของค่าเฉลี่ยความล่าช้าจาก Emulated Delay จะมีความแตกต่างกันในทุกจุด และมีผลเป็นตรงข้ามกันระหว่างค่าที่ปรับ HZ OPTION กับผลต่างของค่าเฉลี่ยความล่าช้า คือ เมื่อปรับค่า HZ OPTION สูงขึ้นความแตกต่างของค่าเฉลี่ยก็จะลดลง เช่น ในช่วงการหน่วงเวลา 500 มิลลิวินาที เครื่องมือวิจัยและคำสั่ง Ping สามารถ

ตรวจวัดได้ตามลำดับดังนี้ ช่วง 100 Hz วัดได้ 4.803052 และ 2.323581 ช่วง 300 Hz วัดได้ 1.568252 และ 0.722209 ช่วง 500 Hz วัดได้ 0.938624 และ 0.426533 ช่วง 1000 Hz วัดได้ 0.463052 และ 0.190122 ช่วง 3000 Hz วัดได้ 0.297776 และ 0.410076 ช่วง 5000 Hz วัดได้ 0.750582, 0.942364 มิลลิวินาที

5.2 สรุปผลการวิจัย

เครื่องมือวิจัยที่ออกแบบเป็นเครื่องมือที่มีประสิทธิภาพในการวัดเทียบเท่ากับกับเครื่องมือวัดชนิดอื่น ๆ การตรวจวัดด้วยเครื่องมือวิจัยสามารถนำไปใช้จริงได้เมื่อนำไปทดลองกับการสื่อสารผ่านอุปกรณ์จริง เช่น ผลการวัดกับอุปกรณ์ Router ที่เป็นผลิตภัณฑ์ของ Cisco เป็นต้น ผลการวัดจะมีข้อแตกต่างจากเครื่องมือชนิดอื่นคือ สามารถแยกชนิดและทิศทางการสื่อสารข้อมูลได้อย่างแม่นยำ โดยสามารถจำแนกความแม่นยำตามพารามิเตอร์ต่างได้ดังนี้

5.2.1 ผลการวัดความยาว พบว่า การตรวจวัดระหว่างเครื่องมือวิจัยและเครื่องมือที่พัฒนาจาก Corallib ตรวจวัดได้ไม่แตกต่างกัน โดยให้ค่าความแตกต่างเป็น 0.00 ไบต์

5.2.2 ผลการวัดเวลาระหว่างการมา พบว่า การตรวจวัดระหว่างเครื่องมือวิจัยและเครื่องมือที่พัฒนาจาก Corallib มีความใกล้เคียงกันมาก ซึ่งสังเกตได้จาก ค่าความแตกต่างอยู่ที่ 0.000021 microseconds ซึ่งน้อยมาก และเมื่อเปรียบบนค่าความเชื่อมั่น 95% ผลจากทั้งสองเครื่องมือแปรความหมายได้ว่า สามารถตรวจวัดได้ไม่แตกต่างกัน

5.2.3 ผลการวัดเวลาการสื่อสารข้อมูลไปกลับ ให้ค่าความแม่นยำสูงมากและมีความใกล้เคียงกันระหว่างผลการตรวจวัดด้วยคำสั่ง Ping และ เครื่องมือวิจัย คือ

5.2.3.1 ผลการวัดข้อมูล ICMP ด้วยคำสั่ง Ping ให้ค่าความแม่นยำอยู่ที่ 0.006897% เมื่อเทียบกับเวลาหน่วงไปกลับบนตัวระบบ ส่วนเครื่องมือวิจัยให้ค่าความแทนยำอยู่ที่ 0.004655% เมื่อเทียบกับเวลาหน่วงไปกลับบนตัวระบบ และมีค่าสูงสุดในการตรวจวัดความแม่นยำระหว่างคำสั่ง Ping และเครื่องมือมือวิจัยอยู่ที่ 0.027749 และ 0.016310% ตามลำดับ

5.2.3.2 ผลการวัดข้อมูล TCP ในช่วงการเชื่อมต่อให้ค่าความแม่นยำอยู่ที่ 0.003754% เมื่อเทียบกับเวลาหน่วงไปกลับบนตัวระบบ ส่วนช่วงปิดการเชื่อมต่อให้ค่าความแทนยำอยู่ที่ 0.005301% เมื่อเทียบกับเวลาหน่วงไปกลับบนตัวระบบ และมีค่าสูงสุดในการตรวจวัดความแม่นยำระหว่างช่วงการเชื่อมต่อและปิดการเชื่อมต่ออยู่ที่ 0.013591 และ 0.022433% ตามลำดับ

5.2.4 ผลการวัดความล่าช้า ให้ค่าความแม่นยำสูงมากและมีความใกล้เคียงกันระหว่างผลการตรวจวัดข้อมูลชนิด ICMP, UDP และ TCP คือ ผลการวัดข้อมูล ICMP ให้ค่าความแม่นยำอยู่ที่

0.022597% เมื่อเทียบกับเวลาหน่วงบนตัวระบบ ผลการวัดข้อมูล UDP ให้ค่าความแม่นยำอยู่ที่ 0.023024% เมื่อเทียบกับเวลาหน่วงบนตัวระบบ และผลการวัดข้อมูล TCP ให้ค่าความแม่นยำอยู่ที่ 0.023419% และมีค่าสูงสุดในการตรวจวัดความแม่นยำระหว่างข้อมูลชนิด ICMP, UDP และ TCP อยู่ที่ 0.114945, 0.119438 และ 0.119793% ตามลำดับ

5.2.5 ผลการวัดความสูญเสีย ให้ค่าความแม่นยำที่เหมือนกันระหว่างผลการวัดด้วยคำสั่ง Ping และ เครื่องมือวิจัย คือ ผลการวัดจากทั้งสองเครื่องมืออยู่ที่ 0.61551% เมื่อเทียบกับค่าความน่าเป็นที่จะเกิดความสูญเสียซึ่งกำหนดให้กับตัวระบบ และมีค่าสูงสุดอยู่ที่ 1.2%

5.2.6 ผลการวัดความล่าช้าและความสูญเสียพร้อมกัน ผลคือให้ค่าความแม่นยำที่ใกล้เคียงกันระหว่างการวัดด้วยเครื่องมือวิจัย และคำสั่ง Ping โดยเครื่องมือวิจัยค่าความแม่นยำอยู่ที่ 0.001913% ส่วนคำสั่ง Ping วัดได้ 0.031031% เมื่อเทียบกับค่าความล่าช้าที่กำหนดให้กับตัวระบบ ส่วนผลการวัดค่าความสูญเสียในทุกชุดการทดลองได้ค่าที่ไม่แตกต่างกัน

นอกจากความแม่นยำในการตรวจวัดข้อมูลแล้ว เครื่องมือวิจัยสามารถนำผลการวิเคราะห์ไปจำแนกความผิดปกติตามทิศทางการสื่อสารข้อมูลได้ทั้งขาไปหรือขากลับ และมีความสามารถในการนำข้อมูลที่จัดเก็บไปวิเคราะห์ได้อย่างหลากหลาย เช่น สามารถจัดเก็บข้อมูลบน Storage Disk หรือจัดเก็บบนหน่วยความจำ หรือ จะรับส่งข้อมูลผ่านเครือข่ายไปยังหน่วยแยกชนิดโดยตรงก็ได้ จึงทำให้ผู้วิเคราะห์สามารถนำเอาข้อมูลที่ได้ไปทดลองซ้ำ หรือพัฒนาเครื่องมือวัดพารามิเตอร์อื่น ๆ ที่ต้องการเพิ่มเติมได้ตามความต้องการของผู้ดูแลระบบ

5.3 ข้อเสนอแนะในงานวิจัย

จากการออกแบบวิธีการตรวจวัดข้อมูลในงานวิจัยนี้ จะมีปัจจัยบางอย่างซึ่งเป็นอุปสรรคต่อการตรวจวัดในการงานวิจัย และขอสรุปเป็นข้อเสนอแนะที่สำคัญได้ดังนี้

5.3.1 การปรับ HZ OPTION บนเครื่องมือวิจัยหรือตัวระบบจะมีผลต่อการตรวจวัดเวลาการสื่อสารข้อมูลด้วย Libpcap ซึ่ง HZ OPTION บน FreeBSD จะมีค่าที่เหมาะสมเป็นไปตามประสิทธิภาพของ CPU บนเครื่องคอมพิวเตอร์ ซึ่งเป็นผลกระทบที่ตรวจพบได้ในขณะทำการวิจัยและอยู่นอกเหนือจากงานวิจัยที่จะวิเคราะห์ต่อไปว่า HZ OPTION ที่เหมาะสมต่อการตรวจวัดเวลาควรอยู่ที่เท่าใดต่อความสามารถของ CPU

5.3.2 การตรวจวัดด้วยเครื่องมือวิจัยมีข้อจำกัดที่ต้องใช้การกำกับเวลาอยู่บนเครื่องคอมพิวเตอร์เครื่องเดียวกัน หากเครื่องมือวิจัยได้รับการพัฒนาให้มีขบวนการกำกับเวลาที่สามารถแยกจากกันได้ก็จะทำให้เครื่องมือวิจัยมีความสมบูรณ์ยิ่งขึ้น

เอกสารอ้างอิง(References)

[1] N.Brownlee, "RFC 2123: Traffic flow measurement: Experiences with NeTraMet", Mar.1997, Status: INFORMATIONAL.

[2] "CiscoNetFlow," http://www.cisco.com/warp/public/732 /netflow/

[3] Daniel W.McRobb, "cflowd: Traffic flow analysis tool," http://www.caida.org/tools/ measurement/cflowd/.

[4] Dave Plonka, "FlowScan: A network traffic flow reporting and visualization tool," in LISA Winter 2000 Conference Proceedings. University of Wisconsin, Madison, Dec. 2000, USENIXLISA.

[5] Ken keys, David Moore, Ryan Koga, Edouard Lagache, Michael Tesch, and k claffy, "The architecture of CoralReef: an Internet traffic monitoring software suite," in PAM2001 – A workshop on Passive and Active Measurements. CAIDA, Apr.2001, RIPE NCC, http://www.caida.org/outreach/papers/pam2001 /coralreef.xml

[6] William Stallings,Data and Computer Communications, PRENTIC-HALL,Inc. Upper Saddle River,New Jersey,1997.

[7] James F.Kurose and Keith W.Ross,Computer Networking, Addison Wesley Longman,Inc. Boston San Francisco ,New York, 2001.

[8] Brynjar Age Viken "Passive Monitoring of Internet Traffic of SuperComputing'98" , 1998.

[9] W.Richard Stevens, "Unix Network Programming Volume 1",RENTIC-HALL, Inc. A Simon & Schuster Company Upper Saddle River, New Jersey, 1998.

[10] Randy Pratt, "FreeBSD Free Unix Operating System", http://www.treefort.org/~rpratt/freebsd, 2002.

[11] RAJ JAIN, "The Art of Computer Systems Performance Analysis", Digital Equipment Corporation Littleton, Massachusetts USA, 1991.

ภาคผนวก

โปรแกรมการเรียกใช้ Corallib Library ตัดมาบางส่วนเพื่อเปรียบเทียบการวัดแพ็กเก็ตและเวลาระหว่างการมา

```
#include <unistd.h>
#include <stdio.h>
#include <ctype.h>
#include <stdlib.h>
#include <string.h>
#include <sys/time.h>
#include <sys/param.h>
#include <sys/socket.h>
#include <sys/types.h>
#include <libcoral.h>
#include <netinet/in.h>
#include <netinet/in_systm.h>
#include <netinet/ip.h>
#include <netinet/tcp.h>
#include "gmt2.h"

#define FULLMASK 0xffffffff

long timeprevios=0,timeremain=0;
int countpkt=0;
uint32_t mymask;

static void count_and_print_pkt(coral_iface_t *iface,
   const coral_timestamp_t *timestamp, void *mydata,
```

```
    coral_pkt_buffer_t *buffer, coral_pkt_buffer_t *header,
    coral_pkt_buffer_t *trailer)

{
    time_t Time;
    struct tm *tm;
    struct timeval globaltime;
    long s,*countp = mydata;
    struct ip *ippt;
    coral_pkt_buffer_t buf[2], *src, *dst;
    uint32_t orgsrc_ipint, orgdst_ipint;
    struct in_addr orgsrc_ip, orgdst_ip;
    int protocolnum;

    ++(*countp);
    src = buffer;
    dst = &buf[0];
    coral_get_payload_by_layer(src,dst,3);
    if (dst->protocol == CORAL_NETPROTO_IP) {
      ippt = (struct ip *)(dst->buf);
      if (ippt->ip_v == 4) {

        orgsrc_ipint=(ippt->ip_src).s_addr&mymask;
        orgdst_ipint=(ippt->ip_dst).s_addr&mymask;
        protocolnum=ippt->ip_p;
        if (strcmp(inet_ntoa(orgsrc_ipint),"192.168.10.1")==0 &&
          strcmp(inet_ntoa(orgdst_ipint),"192.168.20.1")==0) {

          CORAL_TIMESTAMP_TO_TIMEVAL(iface,timestamp,&globaltime);
```

```
            s = (globaltime.tv_sec + thiszone) % 86400;
            Time = (globaltime.tv_sec + thiszone) - s;
            tm = gmtime(&Time);
            if (++countpkt == 1)
              timeprevios=globaltime.tv_sec*1000000+globaltime.tv_usec;
            timeremain=(globaltime.tv_sec*1000000+globaltime.tv_usec)
                    -timeprevios;
            timeprevios=globaltime.tv_sec*1000000+globaltime.tv_usec;
            fprintf(stdout,"%d,",countpkt);
            fprintf(stdout,"%d/",tm->tm_mday);
            fprintf(stdout,"%d/",tm->tm_mon+1);
            fprintf(stdout,"%d,",tm->tm_year+1900);
            fprintf(stdout,"%d:%d:%d.",s/3600,s%3600/60,s%60);
            fprintf(stdout,"%d,",globaltime.tv_usec);
            fprintf(stdout,"%d,",protocolnum);
            fprintf(stdout,"%s,",inet_ntoa(orgsrc_ipint));
            fprintf(stdout,"%s,",inet_ntoa(orgdst_ipint));
            fprintf(stdout,"%d,",ntohs(ippt->ip_len));
            fprintf(stdout,"%d\n",timeremain);
          }
        }
      }
      fflush(stdout);
}

int main()
{
   long count;
   coral_iface_t *iface;
```

```
int maskbits;
char countparams=2,*strparams[2]={"capture","-Csource=if:wb0"};

count = 0;
mymask = FULLMASK;
maskbits = 32;
mymask = mymask<<(32-maskbits);
mymask = htonl(mymask);
thiszone = gmt2local(0);
fflush(stdout);
if (coral_config_arguments(countparams,strparams) < 0) exit(-1);
if (coral_open_all() < 0) exit(-1);
if (coral_start_all() < 0) exit(-1);
coral_set_options(0,CORAL_OPT_PARTIAL_PKT);
iface = coral_next_interface(NULL);
//coral_read_pkts(iface, NULL,count_and_print_pkt, NULL, NULL, 0, &count);
coral_read_pkts(NULL,iface,count_and_print_pkt,NULL,NULL,NULL,&count);
printf("received %ld packets.\n", count);
coral_stop_all();
coral_close_all();
}
```

แฟ้มติดตั้งข้อกำหนดในการตรวจวัดข้อมูลฝั่งขาเข้า

```
# ethernet device such as fpx0, wb0, etc...
DEVICE      wb0
# ip host for sent packet
SENDTOHOST  127.0.0.1
# port token, default is 3056
PORT        3056
# format type 0 - normal(reserve in furture) , default is 0
FORMATTYPE  0
# ethernet identifier, input value between 0 - 9 , default is 0
CAPTUREID   0
# Identifier of each hosts
HOSTID      0
# packet classifying 0-Not 1-Classified, default is 0
CLASSIFYID  0
# packet information 0-None(individual timestamp) 1-Ip 3-Transport F-Reserve
# default is 3
PACKETINFO  3
# save packet's to disk ,  enable-save , disable-none
STOREDISK   disable
# file name, default is 'disk0.dat'
FILENAME    disk0.dat
```

แฟ้มติดตั้งข้อกำหนดในการตรวจวัดข้อมูลฝั่งขาออก

```
# ethernet device such as fpx0, wb0, etc...
DEVICE      fxp0
# ip host for sent packet
SENDTOHOST   127.0.0.1
# port token  default is 3056
PORT        3056
# format type 0 - normal(reserve in furture)
FORMATTYPE   0
# ethernet identifier, input value between 0 - 9, default is 0
CAPTUREID    1
# identifier of each hosts
HOSTID       0
# packet classifying 0-Not 1-Classified, default is 0
CLASSIFYID   0
# packet information 0-None(individual timestamp) 1-Ip 3-Transport F-Reserve
# default is 3
PACKETINFO   3
# save packet's to disk ,  enable-save , disable-none, default is disable
STOREDISK    disable
# file name, default name is 'disk1.dat'
FILENAME     disk1.dat
```

แฟ้มติดตั้งเครื่องมือตรวจวัด

```
#Port Receive from Capture Terminal
PORTRCVE     3056
#Media for Read Data
MEDIAREAD    network
#Internal disk file name, default is "disk0.dat"
INDISK       disk0.dat
#External disk file name, default is "disk1.dat"
EXDISK       disk1.dat
#Internal Network IP
INTERNALNET  192.168.10.0/25,192.168.10.128/25
#System Under Test IP address
SUTIP        192.168.10.254,192.168.20.254
#Capture ID that live in Internal Network
INTERNALCAPID 0
#Capture ID that live in External Network
EXTERNALCAPID 1
#enable or disable Analysis data of distribution table
PACKETSIZE   enable
INTERARRIVAL enable
DELAY        enable
ROUNDTRIPTIME enable
LOSS         disable
#Enable Analysis data flowed
FLOWED       enable
FLWEXP       64
```