การปรับแต่งประสิทธิภาพสำหรับชุดซอฟต์แวร์ตรวจวัดเครือข่ายแบบพาสซีฟโดยใช้เครื่องคอมพิวเตอร์ส่วนบุคคล

PERFORMANCE TUNING FOR PC-BASED PASSIVE NETWORK MEASUREMENT SOFTWARE

แสงเพ็ชร พระฉาย
SANGPETCH PRACHAI

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2548

ISBN 974-15-2069-7

สำนักหอสมุดกลาง

การปรับแต่งประสิทธิภาพสำหรับซอฟท์แวร์ตรวจวัดเครือข่าย แบบพาสซีฟโดยใช้เครื่องคอมพิวเตอร์ส่วนบุคคล

PERFORMANCE TUNING FOR PC-BASED PASSIVE NETWORK MEASUREMENT SOFTWARE

แสงเพ็ชร พระฉาย

SANGPETCH PRACHAI

เลขหมู่.....
เลขทะเบียน 61223
วัน,เดือน,ปี 17 ก.ค. 2549

.b.....
.i.....

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พ.ศ.2548

ISBN 974-15-2069-7

PERFORMANCE TUNING FOR PC-BASED PASSIVE NETWORK MEASUREMENT SOFTWARE

SANGPETCH PRACHAI

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF SCIENCE IN INFORMATION TECHNOLOGY
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2005

ISBN 974-15-2069-7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COPYRIGHT 2005
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การปรับแต่งประสิทธิภาพสำหรับซอฟท์แวร์ตรวจวัดเครือข่ายแบบพาสซีฟโดยใช้เครื่องคอมพิวเตอร์ส่วนบุคคล

PERFORMANCE TUNING FOR PC-BASED PASSIVE NETWORK MEASUREMENT SOFTWARE

ชื่อนักศึกษา นายแสงเพ็ชร พระฉาย

รหัสประจำตัว 43067174

ปริญญา วิทยาศาสตรมหาบัณฑิต

สาขาวิชา เทคโนโลยีสารสนเทศ

อาจารย์ผู้ควบคุมวิทยานิพนธ์ ผศ.อัครินทร์ คุณกิตติ

คณะกรรมการสอบวิทยานิพนธ์		ลายมือชื่อ
ผศ.อัครินทร์	คุณกิตติ	
รศ.ดร.รัตติกร	วรากูลศิริพันธุ์	
รศ.ดร.นพพร	โชติกกำธร	
ผศ.ดร.จันทร์บูรณ์	สถิตวิริยวงศ์	
ผศ.ดร.โชติพัชร์	ภรณวลัย	

วัน / เดือน/ ปี ที่สอบ 18 ตุลาคม 2548 เวลา 13.00 น. เป็นต้นไป

สถานที่สอบ ณ ห้อง M 04 (ชั้นลอย) คณะเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัยรับรองแล้ว

(ผศ.ดร.จารุวัตร เจริญสุข)

คณบดีบัณฑิตวิทยาลัย

วันที่ 15 **เดือน** ธันวาคม **พ.ศ.** 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การปรับแต่งประสิทธิภาพสำหรับซอฟท์แวร์ตรวจวัดเครือข่ายแบบพาสซีฟโดยใช้เครื่องคอมพิวเตอร์ส่วนบุคคล
นักศึกษา	นายแสงเพ็ชร พระฉาย
รหัสนักศึกษา	43067174
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
พ.ศ.	2548
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ผศ.อัครินทร์ คุณกิตติ

บทคัดย่อ

ความต้องการวิเคราะห์พฤติกรรมการสื่อสารข้อมูลเครือข่ายเป็นสิ่งจำเป็นและมีความสำคัญอย่างยิ่งกับทุกองค์กร วิทยานิพนธ์ฉบับนี้จึงเสนอวิธีการออกแบบการตรวจวัดคุณลักษณะข้อมูลเครือข่าย โดยใช้เทคนิคการตรวจวัดข้อมูลแบบพาสซีฟเพื่อแจกแจงคุณลักษณะข้อมูลสื่อสารที่มีชนิดเป็น IPV4 และมีโปรโตคอลเป็น TCP, UDP และ ICMP หลักการตรวจจับข้อมูลใช้ Libpcap Library ในการกรองข้อมูลที่ค่นอยู่ระหว่างตัวระบบ ที่มีการทำงานเป็นแบบรับและส่งต่อ วิธีการที่นำเสนอนี้ได้พัฒนาเครื่องมือขึ้นด้วยภาษา C++ เพื่อทดสอบกับระบบปฏิบัติการ FreeBSD 4.5 Release โดยออกแบบโครงสร้างโปรแกรมให้หน่วยวิเคราะห์ข้อมูลต่าง ๆ ทำงานแบบแบ่งเวลาประมวลผล ที่ประกอบด้วย การตรวจวัดความยาว เวลาระหว่างการมา เวลาการสื่อสารข้อมูลไปกลับ เวลาการสื่อสารข้อมูลผ่านตัวระบบ และความสูญเสียข้อมูลบนตัวระบบ

วิทยานิพนธ์ฉบับนี้ได้แสดงขั้นตอนการพัฒนา และการศึกษาประสิทธิภาพการทำงานของเครื่องมือ ด้วยการทดลองตรวจวัดความถูกต้องเปรียบเทียบกับเครื่องมือวัดชนิดอื่น และทดลองวิเคราะห์ผลกระทบที่เกิดจาก ค่ารอบเวลา(Hz Option) การแบ่งเวลาประมวลผล(Thread) และความเร็วของหน่วยประมวลผลกลาง(CPU Speed) ซึ่งผลการทดลองพบว่า ในด้านความถูกต้องเครื่องมือวิจัยสามารถตรวจวัดจำนวนและเวลาของข้อมูลได้ไม่แตกต่างจากเครื่องมือวัดอื่น ๆ ในด้านปัจจัยของการตั้งค่ารอบเวลาบนเครื่องมือวิจัยพบว่า ไม่ส่งผลกระทบต่อความถูกต้องในการวัดค่าเวลาข้อมูล แต่จะมีผลเมื่อตั้งค่ารอบเวลาบนตัวจำลองระบบ ในด้านปัจจัยของการแบ่งเวลาประมวลผลพบว่า หน่วยวิเคราะห์ข้อมูลที่โปรแกรมแบบหน่วยเดียว สามารถทำงานในอัตราที่เร็วกว่าการทำงานแบบหลายหน่วย ซึ่งเป็นผลจากความสูญเสียในการสลับเวลาทำงานของหน่วยประมวลผล ในด้านปัจจัยจากความเร็วของหน่วยประมวลผลกลางพบว่า ความเร็วของหน่วยประมวลกลางไม่ใช่ปัจจัยเดียวที่จะทำให้เครื่องมือวิจัยสามารถทำงานได้เร็วขึ้น แต่ต้องพิจารณาทั้งขนาดของ BUS, Cache และ RAM ด้วย ซึ่งการศึกษาค่าที่ประเมินได้จากงานวิจัยนี้ ได้ช่วยปรับแต่งและทำให้เครื่องมือวิจัยที่พัฒนา สามารถทำงานได้อย่างมีประสิทธิภาพเพิ่มขึ้น

Thesis Title	Performance Tuning for PC-Based Passive Network Measurement Software.
Student	Mr. Sangpetch Prachai
Student ID.	43067174
Degree	Master of Science
Programme	Information Technology
Year	2005
Thesis Advisor	Assist.Prof. Akharin Khunkitti

ABSTRACT

The need to analyze behavior and data network communication is essential and very important to every organization. Thus, this thesis would like to propose the methods, design, and traffic measurement of internet network characteristics by using the passive traffic measurement technique to distribute data network characteristics that contain with IPV4 and the TCP, UDP, ICMP protocol. The process for data capture is done by using the Libpcap Library to sort out the filter from the system under test that works by store and forward of network policy. The demonstration program is developed by using the C++ program and tested to the FreeBSD 4.5 Release Operating System. The structure of the program is designed for the sub-process of analysis unit to work as a multi-processing unit that will proceed with measuring, packet's length, inter-arrival times, response time, delay, and loss on System Under Test at the same time.

This thesis will elaborate the step of development, the study of performance, and the function of the tools by measuring the accuracy and compare it with other traffic measurement tools, and analyze the side effect that arises from a setting of the Hz Options, a program as Thread System and a CPU speed of PC-Based. The result of the test turned out to be that in accuracy of the propose tools, the research tools is working properly for the quantity and the packet's time stamp, and in setting of the Hz Option of propose tools, the research tools do not effect the accuracy of the measuring to packet's time stamp but they will have some effects on the setting on simulation of System Under Test in term of the program as Thread System of propose tools, it turned out that on the analysis unit that is programmed as a single thread can operate at a faster unit rate than one that is programmed as a multiple thread. This is because the lost from overhead of context switching in each process. In CPU speed of PC-Based, the result turned out that the CPU speed is not the only factor that will speed up the research tools but also the size of System's

BUS, Cache and Ram. Therefore, by the knowledge from these results, it has improved our developed tools to work more efficiently.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จได้อย่างดี ด้วยคำแนะนำ และคำปรึกษาจาก ผศ.อัครินทร์ คุณกิตติ ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์ ข้าพเจ้ารู้สึกทราบซึ้งในความอนุเคราะห์จากท่านอาจารย์และขอขอบพระคุณเป็นอย่างสูง

ขอกราบพระคุณคณาจารย์ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า

ขอขอบคุณเพื่อนๆ พี่ๆ น้อง ๆ ในคณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

ขอขอบคุณบัณฑิตศึกษาและบัณฑิตวิทยาลัย คณะเทคโนโลยีสารสนเทศที่ให้ความช่วยเหลือในเรื่องต่าง ๆ

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้าที่เป็นกำลังใจและให้การสนับสนุนในทุก ๆ เรื่อง ทำให้ข้าพเจ้าสามารถทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมาจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

แสงเพ็ชร พระฉาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ

สารบัญ (ต่อ)

หน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่ หน้า

สารบัญภาพ (ต่อ)

รูปที่ หน้า

สารบัญภาพ (ต่อ)

รูปที่ หน้า

สารบัญภาพ (ต่อ)

รูปที่ หน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ระบบเครือข่ายอินเตอร์เน็ตเป็นเครือข่ายการสื่อสารข้อมูลที่มีขนาดใหญ่ การขยายและเชื่อมต่อเครือข่ายย่อย ย่อมส่งผลทำให้เกิดการแลกเปลี่ยนข้อมูลที่เพิ่มขึ้นตามไปด้วย ดังนั้นจึงเป็นปัจจัยสำคัญของผู้บริหารเครือข่าย ที่ต้องคอยตรวจสอบหรือวิเคราะห์ประสิทธิภาพการสื่อสารข้อมูลให้เกิดความต่อเนื่อง และเพื่อปรับปรุงข้อบกพร่องของการสื่อสารข้อมูลให้อยู่ในสภาพที่เหมาะสมอยู่เสมอ ในสภาพของการจัดระบบเครือข่ายจริงภายในองค์กรหนึ่ง สามารถเปรียบเทียบได้กับเครือข่ายขนาดใหญ่ที่ถูกย่อส่วนให้เล็กลง แต่ละหน่วยงานจะได้รับหมายเลขไอพี(Internet Protocol Address)อย่างน้อยหนึ่งชุด สำหรับการแลกเปลี่ยนข้อมูลไปยังภายนอกหรือเครือข่ายอื่น โดยผู้ดูแลเครือข่ายขององค์กรจะเป็นผู้จัดสรรหมายเลขไอพีเป็นชุดย่อยแจกจ่ายให้กับหน่วยงานภายในตามความเหมาะสม ดังนั้นการสื่อสารข้อมูลภายในองค์กรจึงมีลักษณะเป็นลำดับชั้น(Hierarchy) โดยมีสมาชิกอย่างน้อยหนึ่งโนด(Node)ทำหน้าที่รวบรวมหรือรับส่งข้อมูลไปตามเส้นทางที่สมาชิกของโนดต้นทางกำหนด และขอนิยามกลุ่มของโนดเหล่านี้เรียกว่า ตัวระบบ(System Under Test) ซึ่งอาจอยู่ในรูปของอุปกรณ์อิเลคทรอนิกส์ที่ทำหน้าที่สื่อสารข้อมูลโดยทั่วไป เช่น คอมพิวเตอร์ เราท์เตอร์(Router) หรือสวิชชิ่ง(Switching)

ปัญหาจากการสื่อสารข้อมูลจะไม่เกิดขึ้น ถ้าทุกโนดสื่อสารมีประสิทธิภาพเพียงพอและไม่สร้างปัญหาต่อการให้บริการรับส่งข้อมูลไปยังโนดปลายทาง แต่โดยทั่วไปแล้วมักไม่เป็นเช่นนี้ เนื่องจากพฤติกรรมการสื่อสารข้อมูลของแต่โนดไม่เท่าเทียมกัน ปริมาณของข้อมูลที่แต่ละโนดส่งออกไป อาจทำให้เกิดความหนาแน่นขึ้นที่โนดใด ๆ บนเครือข่าย และปัญหานี้อาจส่งผลกระทบต่อเนื่องจากโนดหนึ่งไปยังอีกโนดหนึ่ง และอาจมีผลต่อเนื่องขยายเป็นวงกว้างไปทั้งระบบ

การออกแบบเครือข่ายที่ได้รับการวิเคราะห์อย่างเหมาะสม จะนำมาซึ่งค่าใช้จ่ายและประสิทธิภาพในการสื่อสารข้อมูลที่เหมาะสมตามไปด้วย ซึ่งในความเป็นจริงการวิเคราะห์การใช้งานทรัพยากร หรืออุปกรณ์เครือข่ายอาจได้รับการวางแผนอย่างเหมาะสมแล้ว แต่ก็อาจเกิดปัญหานี้ขึ้นได้อีก หากผู้ดูแลขาดการตรวจสอบและวิเคราะห์คุณลักษณะการสื่อสารข้อมูลภายในเครือข่ายอย่างต่อเนื่อง ซึ่งพฤติกรรมและความต้องการของผู้ใช้เครือข่ายนั้นสามารถเปลี่ยนแปลงได้ตลอดเวลา และผู้ดูแลระบบก็ไม่สามารถวิเคราะห์ได้จากการสอบถามผู้ใช้เพียงอย่างเดียว แต่ต้องสามารถวิเคราะห์ได้จากพฤติกรรมการสื่อสารข้อมูลจริงบนเครือข่าย เพื่อนำไปใช้วางแผนและออกแบบอุปกรณ์เครือข่ายที่เหมาะสม ดังนั้นการตรวจวัดข้อมูลสื่อสารจริงบนเครือข่ายจึง

เป็นความสำคัญอย่างยิ่งกับทุกองค์กร ที่ต้องการเครื่องมือที่มีประสิทธิภาพเพียงพอต่อการนำไปใช้วิเคราะห์ปรากฏการณ์หรือคุณลักษณะเครือข่ายสื่อสารได้อย่างถูกต้องและเป็นปัจจุบัน ข้อมูลที่วัดได้สามารถนำไปใช้ประเมินความต้องการของระบบได้อย่างแม่นยำ โดยทั่วไปเทคนิคที่นิยมนำไปใช้สร้างเครื่องมือวัดได้แบ่งออกเป็น 2 ลักษณะ คือ

1. การวัดข้อมูลแบบแอคทีฟ (Active Traffic Measurement) หมายถึง การส่งข้อมูลไปบนเครือข่ายสื่อสาร และรอผลจากการตอบสนองข้อมูล ตัวอย่างเครื่องมือเหล่านี้ ได้แก่ Ping, Bing, Traceroute, TPing, HTping, PChar

2. การวัดข้อมูลแบบพาสซีฟ (Passive Traffic Measurement) หมายถึง การดักจับข้อมูลที่ไหลผ่านบนเครือข่ายจริง ตัวอย่างเครื่องมือเหล่านี้ ได้แก่ Libpcap Library, Coral Library, CFlowd[1], NetTramet[2], FlowScan[3], CoralReef[4], Tcpdump, MRTG

ปัญหาหลักของการตรวจวัดด้วยวิธีการแบบแอคทีฟ คือไม่สามารถวัดได้เมื่อเครือข่ายเกิดความหนาแน่นจนไม่สามารถส่งข้อมูลแบบโต้ตอบได้ นอกจากนี้ยังสร้างภาระงานเพิ่มให้กับระบบ และเหมาะสมกับการศึกษาคุณลักษณะข้อมูลที่ได้จากการโต้ตอบเท่านั้น การวัดแบบพาสซีฟจึงเป็นเทคนิคที่ช่วยทำให้นักวิเคราะห์ สามารถตรวจวัดข้อมูลที่ไหลผ่านเครือข่ายจริงได้ โดยปราศจากผลกระทบใด ๆ ต่อระบบ และเหมาะกับการศึกษาพฤติกรรมข้อมูลที่สื่อสารอยู่บนเครือข่ายจริงได้อย่างต่อเนื่อง ดังนั้นงานวิจัยนี้จึงนำเสนอการออกแบบวิธีการตรวจวัดคุณลักษณะการสื่อสารข้อมูลที่จำเป็นต่อการนำไปใช้วางแผนและออกแบบเครือข่าย รวมถึงการศึกษาปัจจัยที่มีผลกระทบต่อประสิทธิภาพของเครื่องมือที่พัฒนาขึ้น เพื่อนำไปใช้ในการปรับแต่งเพิ่มความสามารถให้เพียงพอต่อการใช้ตรวจวัดจริงบนเครือข่ายได้

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

การศึกษาวิจัยนี้ได้เสนอวิธีการตรวจวัดคุณลักษณะการสื่อสารข้อมูลเครือข่าย โดยมีวัตถุประสงค์หลักในการศึกษาคือ

1. ออกแบบวิธีการตรวจวัด และวิเคราะห์การสื่อสารข้อมูลบนเครือข่ายอินเตอร์เน็ตด้วยวิธีการทางสถิติ เพื่อแจกแจงคุณลักษณะหรือพฤติกรรมการสื่อสารข้อมูล ประกอบด้วย การตรวจวัดความยาวแพ็กเก็ต(Packets Size) เวลาระหว่างการมา(Inter-Arrival Times) เวลาการสื่อสารข้อมูลไปและกลับ(Response Times) เวลาการสื่อสารข้อมูลผ่านตัวระบบ(Delay on System Under Test) และความสูญเสียข้อมูลบนตัวระบบ(Loss on System Under Test)

2. ศึกษาวิธีการพัฒนาเครื่องมือวัดตามวิธีการที่ออกแบบ

3. วิเคราะห์ความถูกต้อง เมื่อเทียบกับเครื่องมือวัดอื่น

เอกสารนี้มีลิขสิทธิ์ ห้ามมิให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ศึกษาปัจจัยที่มีผลกระทบต่อการตรวจวัด ทั้งในด้านความถูกต้อง และความเร็วในการวิเคราะห์ข้อมูล โดยพิจารณาจากค่ารอบเวลา(Clock Tick) และเทคนิคที่ใช้ในการวิเคราะห์ทั้งแบบหน่วยประมวลผลเดียว(Single Thread) และ หลายหน่วยประมวลผล(Multiple Thread)

1.3 สมมติฐานของการศึกษาวิจัย

ผลจากการศึกษาและวิจัยสามารถพัฒนาเครื่องมือเพื่อใช้ในการวิเคราะห์พฤติกรรมการสื่อสารข้อมูลเครือข่ายในเชิงสถิติได้ตามสมมติฐานการวิจัย คือ

1. ผลการออกแบบจะสามารถนำไปใช้สร้างเป็นเครื่องมือตรวจวัดการสื่อสารข้อมูลได้ถูกต้องไม่ต่างจากเครื่องมือวัดอื่น โดยพิจารณาตามกฎและทฤษฎีการสื่อสารข้อมูลทั้ง 3 ชนิด ได้แก่ TCP, UDP และ ICMP เปรียบเทียบกับวิธีการหรือเครื่องมือวัดอื่นๆ ผลลัพธ์ที่ได้อาจจะมีความแตกต่างกัน และช่วยสะท้อนถึงข้อบกพร่องในการออกแบบเครื่องมือวัดได้

2. เครื่องมือวิจัยจะสามารถตรวจวัดเวลาได้ถูกต้องโดยไม่ขึ้นกับค่ารอบเวลา(Hz Option) ที่กำหนดให้กับเครื่องมือ ซึ่งระบบปฏิบัติการ FreeBSD สามารถกำหนดค่ารอบเวลาให้กับหน่วยประมวลผลได้ ด้วยการตั้งค่า Hz Option ใน Kernel ซึ่งผลกระทบของการปรับค่ารอบเวลา อาจมีผลต่อหน่วยที่ทำหน้าที่กำกับเวลาข้อมูลให้คลาดเคลื่อนแตกต่างกัน

3. ค่ารอบเวลาที่กำหนดให้เครื่องมือวัดจะมีผลทำให้เครื่องมือมีอัตราเร็วในการวิเคราะห์ข้อมูลที่แตกต่างกัน การที่หน่วยประมวลผลได้รับทรัพยากรเวลาในการประมวลผลที่เวลาต่างกัน อาจมีผลทำให้อัตราการวิเคราะห์ข้อมูลโดยรวมมีความแตกต่างกัน ค่ารอบเวลาสั้นจะมีผลทำให้การตอบสนองจากหน่วยวิเคราะห์ต่าง ๆ ดีขึ้น แต่อาจทำให้ต้องสูญเสียเวลาในการวิเคราะห์ข้อมูลเพิ่มขึ้น และทำให้ผลรวมเวลาการวิเคราะห์ข้อมูลต่อแพ็กเก็ตต้องใช้รอบเวลามากขึ้นตามไปด้วย

4. การแยกหน่วยดักจับข้อมูลออกจากคอมพิวเตอร์เครื่องเดียวกันจะมีผลทำให้อัตราเร็วในการวิเคราะห์ข้อมูลแตกต่างกัน หน่วยดักจับข้อมูลเป็นส่วนเชื่อมต่อกับอีเทอร์เนตกราด (Ethernet Card)เพื่ออ่านข้อมูลจากเครือข่ายโดยตรง ภาระงานของหน่วยประมวลผลนี้จะขึ้นอยู่กับการไหลของข้อมูลที่มีอยู่จริงบนเครือข่าย ดังนั้นการแยกหน่วยดักจับข้อมูลออกไปทำงานบนคอมพิวเตอร์เครื่องอื่น ทรัพยากรเวลาจะถูกเฉลี่ยคืนให้กับหน่วยวิเคราะห์ และอาจมีผลทำให้ความเร็วในการวิเคราะห์ข้อมูลของเครื่องมือมีความแตกต่างกัน

5. เทคนิคการวิเคราะห์ข้อมูลแบบหน่วยประมวลผลเดียว(Single Thread)จะมีอัตราเร็วในการวิเคราะห์ข้อมูลดีกว่าแบบหลายหน่วยประมวลผล(Multiple Thread) เครื่องมือที่ออกแบบได้พัฒนาให้ทำงานโดยใช้เทคนิคการวิเคราะห์แบบแบ่งเวลาประมวลผล ดังนั้นการจัดสรรทรัพยากรเวลาให้กับหน่วยประมวลผลจำนวนมาก อาจมีผลทำให้ประสิทธิภาพในการวิเคราะห์

ข้อมูลต่ำลง เนื่องจากต้องสูญเสียเวลาไปกับการสลับหน่วยเวลาทำงานมากกว่าการวิเคราะห์ข้อมูลแบบหน่วยประมวลผลเดียว

1.4 ทฤษฎีและแนวคิดที่นำมาใช้กับวิทยานิพนธ์

วิธีการตรวจวัดพฤติกรรมการสื่อสารข้อมูลบนเครือข่าย ได้ออกแบบโดยใช้เทคนิคการวัดแบบพาสซีฟเพื่อแบ่งหน่วยประมวลผลในการตรวจจับ แยกชนิด และวิเคราะห์ข้อมูลเครือข่าย ได้แก่ TCP, UDP และ ICMP ซึ่งข้อมูลแต่ละชนิดจะมีพฤติกรรมทั้งจำนวนและเวลาการสื่อสารข้อมูลที่แตกต่างกัน วิทยานิพนธ์เล่มนี้ได้แสดงวิธีการพัฒนาเครื่องมือ โดยโปรแกรมการทำงานแบบแบ่งเวลาประมวลผลบนระบบปฏิบัติการ FreeBSD ให้ตรวจจับข้อมูลโดยเรียกใช้ Libpcap Library ใช้ระบบวิเคราะห์ข้อมูลจากหน่วยความร่วม(Share Memory) จัดการกับข้อมูลด้วยวิธีการแบบต้นไม้(Binary Tree) เปรียบเทียบการตรวจจับข้อมูลกับ Coral Library โดยลักษณะเด่นของเครื่องมือวัดคือ สามารถตรวจวัดข้อมูลเครือข่ายได้ โดยที่หน่วยทำงานต่าง ๆ เป็นอิสระจากกัน ทำให้สามารถเพิ่มขยายหรือพัฒนาการวิเคราะห์คุณลักษณะข้อมูลในรูปแบบต่าง ๆ ได้อย่างมีประสิทธิภาพ

1.5 ขอบเขตของวิทยานิพนธ์

วิทยานิพนธ์ฉบับนี้ได้เสนอผลงานวิจัยเพื่อออกแบบวิธีการตรวจวัดและศึกษาการพัฒนาเครื่องมือสำหรับวิเคราะห์คุณลักษณะการสื่อสารข้อมูลบนเครือข่ายจริง โดยใช้เทคนิคการวัดแบบพาสซีฟ เพื่อแจกแจงคุณลักษณะและปัญหาของการสื่อสารข้อมูลผ่านตัวระบบ เฉพาะข้อมูลที่มีชนิดเป็น IPV4 ด้วยการตรวจจับข้อมูลผ่านคู่อีเทอร์เนตกราด(Ethernet Card)ที่คั่นอยู่ระหว่างตัวระบบ โดยที่ตัวระบบมีลักษณะการทำงานเป็นแบบรับและส่งต่อ(Store and Forward) วิธีการตรวจจับข้อมูลใช้ Libpcap Library เพื่อกรองเฉพาะส่วนหัวของข้อมูล TCP, UDP และ ICMP ที่สำคัญไปใช้ในการวิเคราะห์ เช่น ใช้ไอพี(IP: Internet Address)ในการจำแนกทิศทางการสื่อสารผ่านเข้าและออกจากตัวระบบ นำเสนอการพัฒนาเครื่องมือด้วยภาษา C++ เพื่อทดลองกับระบบปฏิบัติการ FreeBSD 4.5 Release โดยใช้วิธีคำนวณและกำกับเวลา(Time Stamp)จากการติดตั้งหน่วยตรวจจับข้อมูลบนคอมพิวเตอร์เครื่องเดียวกัน และทำการศึกษาวิจัยประสิทธิภาพการทำงานของเครื่องมือที่พัฒนาขึ้น ทั้งในด้านความถูกต้องและความเร็วในการวิเคราะห์ข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6 ขั้นตอนการศึกษาวิจัย

วิทยานิพนธ์ฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

บทที่ 1 กล่าวถึงความเป็นมาของงานวิจัย ความมุ่งหมายและวัตถุประสงค์ สมมติฐาน ทฤษฎีที่ใช้ ขอบเขตของการวิจัย และขั้นตอนการศึกษา

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในการวิจัย พื้นฐานของการสื่อสารข้อมูลผ่านเครือข่ายอินเตอร์เน็ต การตรวจจับข้อมูลด้วย บีพีเอฟ(BPF: Berkeley Packet Filter) และการออกแบบเครื่องมือต่าง ๆ โดยใช้เทคนิคแบบพาสซีฟ

บทที่ 3 กล่าวถึงวิธีการออกแบบเครื่องมือตรวจวัดข้อมูลเครือข่าย ประกอบด้วย การตรวจวัดความยาว เวลาระหว่างการมา เวลาการสื่อสารข้อมูลไปกลับ เวลาการสื่อสารข้อมูลผ่านตัวระบบ และความสูญเสียข้อมูลบนตัวระบบ

บทที่ 4 กล่าวถึงการวิเคราะห์ประสิทธิภาพของเครื่องมือที่ออกแบบ ประกอบด้วย การวิเคราะห์ความถูกต้อง และการวิเคราะห์ความเร็วในการตรวจวัดข้อมูลเครือข่าย เพื่อแสดงให้เห็นว่าวิธีการที่นำเสนอนั้นสามารถใช้ในการตรวจวัดข้อมูลจริงบนเครือข่ายได้ และมีปัจจัยใดที่เป็นผลกระทบทำให้ประสิทธิภาพของการพัฒนาเครื่องมือตามวิธีการที่ออกแบบมีความแตกต่างกันในด้านการตรวจวัดข้อมูล

บทที่ 5 เป็นบทสรุปผลการวิจัยและข้อเสนอแนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีพื้นฐานการสื่อสารและการตรวจวัดข้อมูล

ในหัวข้อนี้จะกล่าวถึงทฤษฎีพื้นฐานต่างๆ ที่เกี่ยวข้องกับพฤติกรรมการสื่อสารข้อมูลตามชนิดต่าง ๆ ระบบและโครงสร้างข้อมูลที่ตรวจจับได้จาก Libpcap Library และการพัฒนาเครื่องมือโดยใช้เทคนิคแบบพาสซีฟ ซึ่งเนื้อหาในบทนี้จะกล่าวถึงเทคนิคการออกแบบเครื่องมือวัดข้อมูลโฟลว(Flows) และการตรวจจับข้อมูลโดยใช้ Corallib Library เพื่อแจกแจงคุณลักษณะการสื่อสารข้อมูล ซึ่งเนื้อหาทั้งหมดนี้จำเป็นสำหรับการศึกษา และประเมินประสิทธิภาพของระบบการออกแบบเครื่องมือโดยทั่วไป

2.1 บทนำ

ผู้ใช้ ผู้บริหาร และนักออกแบบระบบคอมพิวเตอร์ต่างให้ความสนใจในการตรวจวัดประสิทธิภาพ โดยมีเป้าหมายที่จะได้รับหรือสามารถจัดเตรียมระบบคอมพิวเตอร์ ที่มีประสิทธิภาพสูงแต่มีค่าใช้จ่ายต่ำด้วยกันทั้งสิ้น ผลการตรวจวัดต้องสามารถนำไปใช้ได้อย่างต่อเนื่องและทันต่อสถานภาพปัจจุบัน อีกทั้งมีผลสืบเนื่องให้สามารถขยายขีดความสามารถไปในอนาคตได้ ปัจจุบันคอมพิวเตอร์ส่วนบุคคล(Personal Computer)ได้มีการพัฒนาให้มีประสิทธิภาพที่สูงขึ้นและได้ทำการเชื่อมต่อให้เกิดการแลกเปลี่ยนการสื่อสารกันจนมีขนาดใหญ่เรียกว่า เครือข่ายอินเตอร์เน็ต การสื่อสารข้อมูลบนเครือข่ายใช้มาตรฐานเดียวกันเรียกว่า TCP/IP (Transmission Control Protocol / Internet Protocol) โดยมีแพ็กเก็ต(Packets)ทำหน้าที่ในการลำเลียงข้อมูลระหว่างคู่สื่อสารผ่านสายสัญญาณหลากหลายชนิด ปัจจุบันอัตราการเชื่อมต่อและแลกเปลี่ยนข่าวสารได้เพิ่มขึ้นอย่างต่อเนื่อง ปริมาณความต้องการของแต่ละหน่วยงานก็แตกต่างกัน ดังนั้นเทคโนโลยีการตรวจวัดเครือข่ายจึงเป็นปัจจัยหนึ่งที่ต้องนำมาใช้ในการพัฒนาประสิทธิภาพการสื่อสารข้อมูลของหน่วยงาน

การจัดการเครือข่ายอาจทำขึ้นเฉพาะภายในองค์กร(Campus Network)ที่เชื่อมต่ออยู่กับเครือข่ายอื่น ๆ ผ่านผู้ให้บริการเครือข่ายสาธารณะ(Public Internet Service Provider) อุปกรณ์เครือข่ายที่เชื่อมต่อกับเครือข่ายภายนอกจึงเป็นองค์ประกอบหลักที่สำคัญขององค์กร ที่จะต้องวิเคราะห์ความเหมาะสม เพื่อให้เกิดการสูญเสียค่าใช้จ่ายให้สอดคล้องกับประสิทธิภาพที่จะได้มาให้มากที่สุด ซึ่งปัจจัยที่สำคัญของการวิเคราะห์ประสิทธิภาพการสื่อสารข้อมูลที่สำคัญ คือ

1. ปริมาณการสื่อสาร(Traffic Quantity) หมายถึง จำนวนข้อมูลที่มีการแลกเปลี่ยนกันบนเครือข่ายคอมพิวเตอร์ ซึ่งจะมีผลกับหน่วยจัดเก็บข้อมูลบนอุปกรณ์เครือข่ายโดยตรง โดยรวมแล้วจำนวนและขนาดของข้อมูลจะมีความสัมพันธ์กัน สถานภาพเครือข่ายที่พบข้อมูลกระจาย

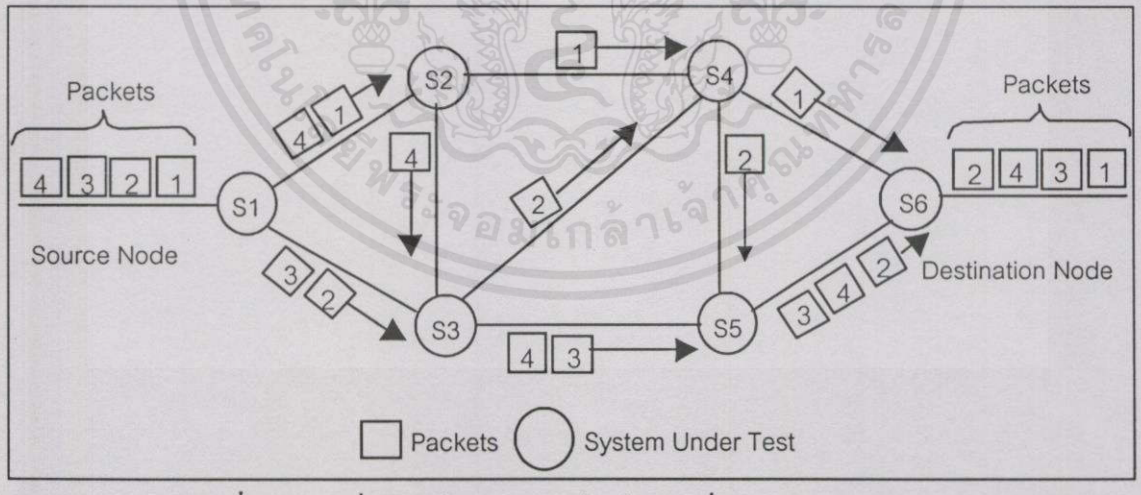
เป็นชิ้นส่วนเล็ก ๆ อาจสะท้อนให้เห็นถึงการสูญเสียเวลาในการรวมกลุ่มข้อมูลที่ปลายทางได้ หรือข้อมูลที่มีความยาวมากก็จะสะท้อนถึงอัตราการใช้พื้นที่จัดเก็บข้อมูลบนอุปกรณ์เครือข่ายด้วย ในกรณีที่กำหนดพฤติกรรมการรับและส่งข้อมูลเป็นแบบรับและส่งต่อ(Store and Forward)

2. เวลาที่ใช้(Time Used) หมายถึง ระยะเวลาที่ใช้ไปในการแลกเปลี่ยนข่าวสาร กล่าวได้ว่าการสื่อสารข้อมูลใด ๆ ย่อมต้องการความรวดเร็วและถูกต้องของข้อมูลให้มากที่สุด การตอบรับการสื่อสารข้อมูลที่ใช้เวลานาน ๆ จะสะท้อนให้เห็นถึงความบกพร่องในการจัดนโยบายการสื่อสารข้อมูลโดยภาพรวมได้ หรือถ้าพิจารณาข้อกำหนดของการใช้ TCP(Transmission Control Protocol) เวลาที่นานเกินไปอาจทำให้ผู้ส่งต้องส่งข้อมูลใหม่(Retransmission)และซ้ำซ้อนไปยังปลายทางเป็นจำนวนมาก และอาจทำให้เกิดความหนาแน่นและสูญเสียข้อมูลต่อไปได้

ปัจจัยที่กล่าวถึงเป็นผลนำมาซึ่งการวิจัย และการออกแบบเครื่องมือตรวจวัดพฤติกรรมการสื่อสารข้อมูลต่าง ๆ ที่จำเป็นต่อการนำไปใช้พยากรณ์ความสามารถของระบบ สามารถนำไปใช้ขจัดปัญหารวมถึงการออกแบบเครือข่ายได้อย่างเหมาะสม

2.2 เทคนิคการสื่อสารข้อมูล(Communication Technique)

เทคนิคการสื่อสารข้อมูลบนเครือข่ายอินเตอร์เน็ตใช้วิธีการเรียกว่า แพ็กเก็ตสวิชชิ่ง (Packet Switching)[5][6][7] ซึ่งเป็นวิธีการที่ยอมให้คอมพิวเตอร์ต่าง ๆ สามารถใช้เส้นทางหรือใช้บางส่วนของเส้นทางได้ในเวลาเดียวกันดังแสดงได้ในภาพที่ 2.1



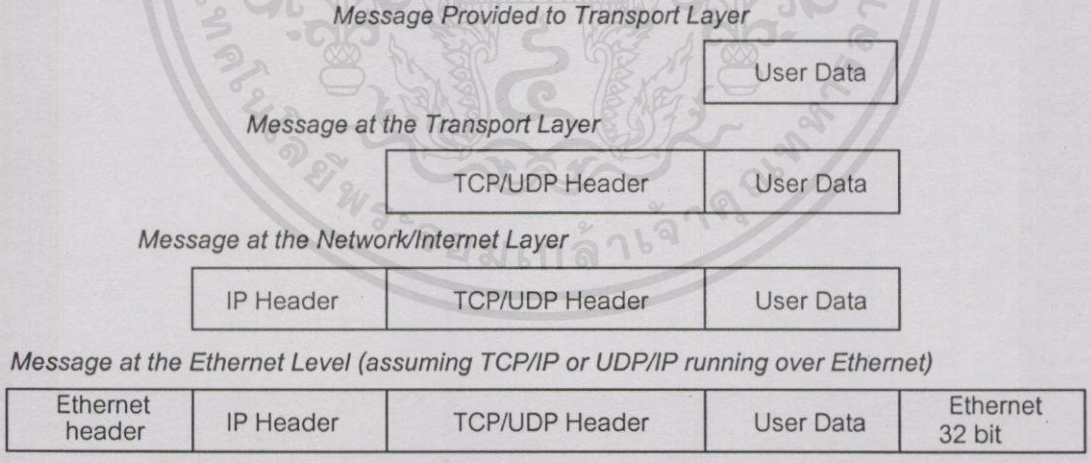
ภาพที่ 2.1 การสื่อสารข้อมูลแบบแพ็กเก็ตสวิชชิ่ง(Packet Switching)

แพ็กเก็ตจากโนดต้นทาง(Source Node) จะส่งผ่านไปบนสายสัญญาณเดียวกันเข้าสู่อุปกรณ์สวิชชิ่ง(Switching)หรือเราท์เตอร์(Router)ไปยังโนดปลายทาง(Destination Node) ซึ่งขอใช้นิยามเรียกหน่วยบริการนี้ว่า ตัวระบบ(System Under Test) แทนความหมายที่รวมถึงอุปกรณ์การสื่อสารข้อมูลทุกชนิดบนเครือข่ายอินเตอร์เน็ต แพ็กเก็ตหนึ่ง ๆ ที่ไปถึงตัวระบบต้องถูกจัดเก็บให้

สมบูรณ์ก่อนการตัดสินใจ เพื่อส่งต่อให้กับตัวระบบที่อยู่ถัดไป และอาจไม่ตรงตามลำดับเมื่อไปถึงโนดผู้รับปลายทาง เนื่องจากตัวระบบมีนโยบายการกำหนดเส้นทางที่แตกต่างกัน เช่น ระบบมาก่อนได้รับบริการก่อน(First Come First Serve) หรือ ระบบบริการแบบเท่าเทียม(Fair Queue) ดังนั้นตัวระบบจึงเป็นตัวแปรสำคัญที่ทำให้เกิดความความล่าช้าในการสื่อสารขึ้น โดยแปรผันตามความยาวของแพ็กเก็ต และนโยบายในการเลือกเส้นทางของตัวระบบเอง ซึ่งในความเป็นจริงความยาวของแพ็กเก็ตโดยมากมักมีค่าไม่เท่ากันเนื่องจากใช้กลไกควบคุมการสื่อสารข้อมูลด้วย TCP/IP(Transmition Control Protocol/Internet Protocol) นอกจากนี้การสื่อสารข้อมูลอาจขาดความสมบูรณ์ขึ้นได้เมื่อเกิดความสูญเสียขึ้นที่ตัวระบบเพราะพื้นที่จัดเก็บข้อมูลบนตัวระบบไม่เพียงพอต่ออัตราการมาของแพ็กเก็ต

2.3 คุณลักษณะของแพ็กเก็ต(Packets Characteristic)

ระบบการสื่อสารบนเครือข่ายคอมพิวเตอร์ได้มีการออกแบบมาตรฐาน เพื่อใช้อธิบายถึงการเชื่อมต่อ และใช้เป็นข้อตกลงในการออกแบบโปรโตคอล(Protocol)เพื่อการสื่อสารข้อมูล เรียกว่า OSI (Open System Interconnection Model) โดยแบ่งหน้าที่การทำงานออกเป็น 7 ชั้น (Layer) แต่ละชั้นทำหน้าที่ห่อหุ้มข้อมูลเรียกว่า แพ็กเก็ต เพื่อส่งต่อไปยังผู้รับปลายทาง ตามภาพที่ 2.2 ที่แสดงให้เห็นถึงการห่อหุ้มข้อมูลจนกระทั่งเป็นแพ็กเก็ต

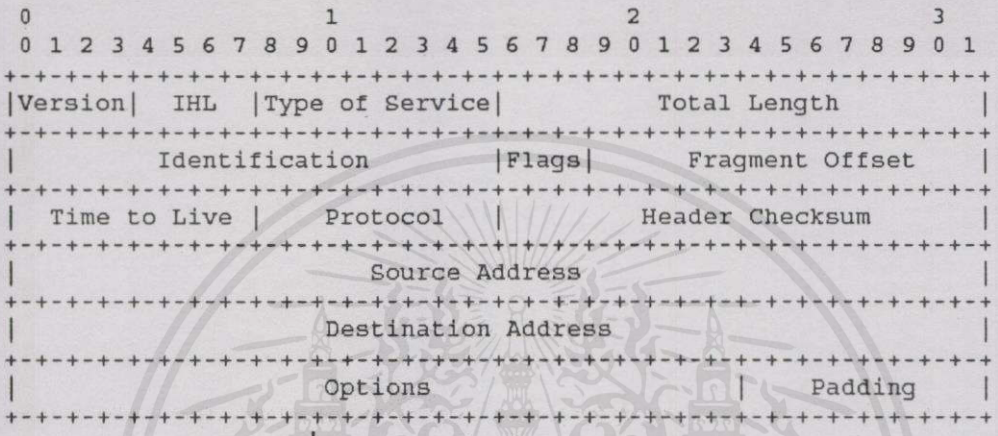


ภาพที่ 2.2 การห่อหุ้มแพ็กเก็ต(Packet Encapsulation)

เมื่อข้อมูล(User Data) ส่งจากชั้นบนสุดคือ Application Layer ลงไปยังชั้นที่ต่ำกว่า แต่ละชั้นก็จะทำการเพิ่มข้อมูลลงไปที่ส่วนด้านหน้าของข้อมูลจากชั้นก่อนหน้า เช่น TCP/UDP Header, IP Header และ Ethernet Header ถูกเติมลงในชั้น Transport, Network และ Data Link ตามลำดับ โดยมีรูปแบบของข้อมูลส่วนหัว(Header Format)ตามโครงสร้าง IPV4 Header ดังต่อไปนี้

2.3.1 ส่วนหัวของไอพี(Internet Protocol Header)

เป็นข้อมูลที่ถูกเพิ่มในส่วนของ Network Layer หรือเรียกส่วนนี้ว่า IP Datagram มีหน้าที่ในการเชื่อมต่อและรับประกันการส่งข้อมูลไปยังผู้รับปลายทางให้ถูกต้อง นอกจากนี้ยังมีหน้าที่ในการแยกชิ้นส่วน(Fragmentation) และรวมกลุ่มข้อมูล(Reassembly) ที่เกิดจากกรณีข้อมูลมีขนาดเกินกว่าค่าสูงสุดในการส่งข้อมูลหรือเรียกว่า MTU(Maximum Transmission Unit Size) ดังแสดงได้ในภาพที่ 2.3



ภาพที่ 2.3 โครงสร้างข้อมูลใน IP Datagram

โดยที่	Version	ใช้แสดงรูปแบบของ Internet Header
	IHL	ใช้บอกความยาวของ Internet Header
	Type of Service	ใช้บ่งบอกคุณสมบัติของการให้บริการ
	Total length	ใช้บอกความยาวของ Datagram ที่รวมถึง Internet Header
	Identification	ใช้บอกลำดับของข้อมูลที่ส่งไปยังปลายทาง
	Flags	ใช้บอกสถานะของ Datagram เมื่อมีการแยกชิ้นส่วน
	Time to Live	ใช้บอกเวลาสูงสุดที่ Datagram อยู่ได้ในระบบเครือข่าย
	Protocol	ใช้บอกประเภทของโปรโตคอลใน Transport Layer
	Header Checksum	ใช้ตรวจสอบความถูกต้องของ Internet Header
	Source Address	ใช้บอกที่อยู่ต้นทางของผู้ส่ง
	Destination Address	ใช้บอกที่อยู่ปลายทางของผู้ส่ง
	Options	ใช้ประโยชน์โดยปรับเปลี่ยนได้ตามความต้องการ

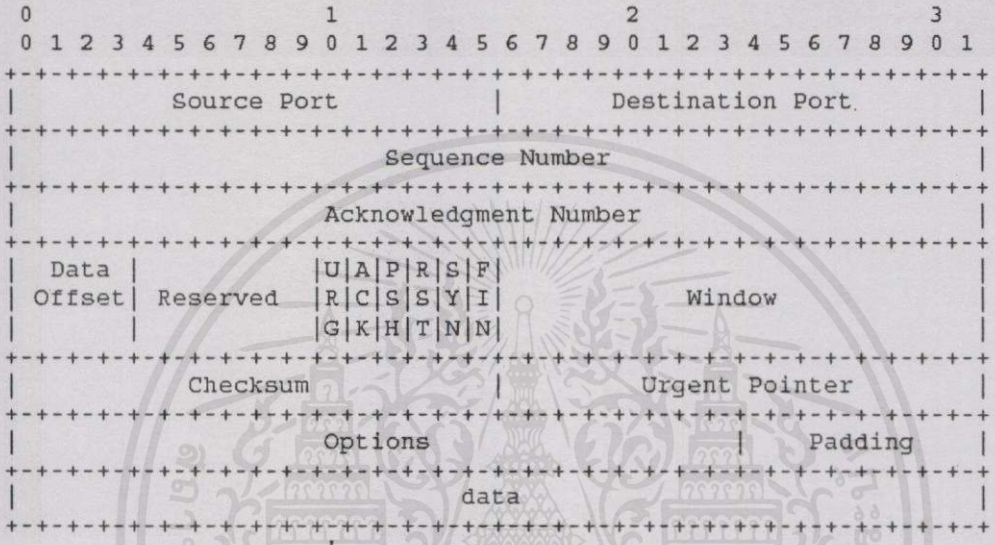
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 ส่วนหัวในชั้น Transport Layer

เป็นข้อมูลที่เพิ่มเติมต่อจาก Internet Header เพื่อทำหน้าที่รับประกันความน่าเชื่อถือของการรับและส่งข้อมูล โดยแบ่งรูปแบบข้อมูลเป็น 2 โปรโตคอล(Protocol) คือ TCP(Transmission Control Protocol) และ UDP(User Datagram Protocol) นอกจากนี้ยังมีอีกหนึ่งโปรโตคอล

(Protocol)คือ ICMP(Internet Control Message Protocol) ซึ่งโดยทั่วไปนิยมใช้นำเสนอให้อยู่ในชั้น Network โดยโครงสร้างข้อมูลมีลักษณะที่ประกอบด้วยส่วนที่สำคัญคือ

1. โครงสร้างข้อมูล TCP Header การสื่อสารข้อมูลด้วย TCP ผู้ส่งจะได้รับการรับประกันการสื่อสารข้อมูลคือ ข้อมูลครบถ้วนและถูกต้อง มีส่วนในการควบคุมความคับคั่งเพื่อลดความเสี่ยงจากการสูญเสียข้อมูลที่ส่งไปยังปลายทาง โดยใช้โครงสร้างข้อมูลที่แสดงได้ในภาพที่ 2.4



ภาพที่ 2.4 โครงสร้างข้อมูล TCP Header

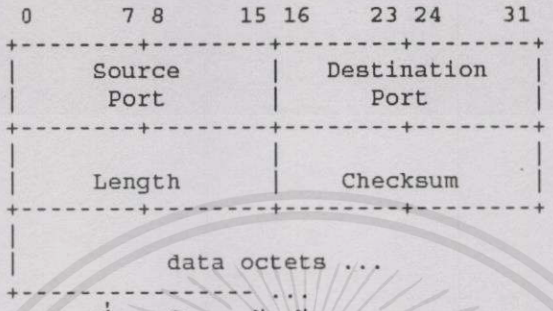
โดยที่	Source Port	ใช้บอกหมายพอร์ต(Port)ที่ใช้ส่งข้อมูลจากต้นทาง
	Destination Port	ใช้บอกหมายเลขพอร์ตที่ต้องใช้รับปลายทาง
	Sequence Number	ใช้บอกเลขลำดับของข้อมูลในแต่ละส่วน(Segment)
	Acknowledgment Number	ใช้ในการคาดคะเนการส่งกลับมาของผู้รับ
	Data Offset	ใช้บอกจุดเริ่มต้นของข้อมูล
	Reserved	สงวนไว้ใช้ในอนาคต
	URG	ใช้บอกสถานะความสำคัญของข้อมูล
	ACK	ใช้บอกการตอบรับจากผู้ส่ง
	PSH	ใช้บอกสถานะในการส่งข้อมูล
	RST	ใช้บอกสถานะในการยกเลิกการติดต่อ
	SYN	ใช้บอกสถานะในเชื่อมต่อ
	FIN	ใช้บอกสถานะในการปิดการเชื่อมต่อ
	Window	ใช้ควบคุมการรับชิ้นส่วนข้อมูลจากผู้ส่ง
	Checksum	ใช้ตรวจสอบความถูกต้องของ TCP Header
	Urgent Pointer	ใช้ลำดับความสำคัญในการรับส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Options ใช้ประโยชน์โดยปรับเปลี่ยนได้ตามความต้องการ

Data ใช้บรรจุข้อมูลที่อยู่เหนือกว่า Transport Layer

2. โครงสร้างข้อมูล UDP Header การสื่อสารข้อมูลด้วย UDP นิยมใช้กับข้อมูลที่ต้องการส่งไปให้ถึงผู้รับด้วยความรวดเร็ว โดยไม่คำนึงถึงความสมบูรณ์และความถูกต้องของข้อมูลที่จะไปถึงฝั่งผู้รับ โดยมีโครงสร้างที่แสดงได้ในภาพที่ 2.5



ภาพที่ 2.5 โครงสร้างข้อมูล UDP Header

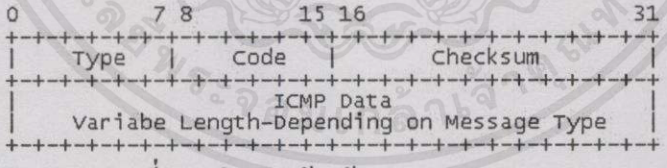
โดยที่ Source Port ใช้บอกหมายพอร์ตที่ใช้ส่งข้อมูลจากต้นทาง

Destination Port ใช้บอกหมายเลขพอร์ตที่ต้องใช้รับปลายทาง

Length ใช้บอกความยาวของข้อมูลรวมถึง UDP Header

Checksum ใช้ตรวจสอบความถูกต้องของ UDP Header

3. โครงสร้างข้อมูล ICMP Header การสื่อสารข้อมูลด้วย ICMP นิยมใช้เพื่อรายงานความผิดพลาดหรือสถานภาพของการสื่อสารบนเครือข่าย โดยมีโครงสร้างที่แสดงได้ดังภาพที่ 2.6



ภาพที่ 2.6 โครงสร้างข้อมูล ICMP Header

โดยที่ Type ใช้บอกรูปแบบของการรายงานสถานภาพเครือข่าย

code ใช้เป็นหมายเลขบอกสถานภาพเครือข่าย

Checksum ใช้ตรวจสอบความถูกต้องของ ICMP Header

ICMP Data เพิ่มเติมได้เพื่อให้เกิดประโยชน์กับการรายงาน เช่นในกรณีของการตรวจสอบการไปถึงของข้อมูล อาจใช้สัญญาณตอบกลับ(Echo) ที่ประกอบด้วยสัญญาณการเรียก (Request) และตอบ(Reply) ซึ่งอาจส่งได้หลายครั้งจากผู้ส่งเดียวกัน ดังนั้นจึงต้องมี ICMP Identifier เพื่อระบุคู่สัญญาณให้ถูกต้องในแต่ละครั้ง นอกจากนี้ในแต่ละครั้งสัญญาณอาจมีการส่งได้หลายครั้ง เช่น การใช้ Ping Command บน Operating System ที่สามารถระบุจำนวนของ

สัญญาณเรียกได้ ดังนั้นการร้องขอแต่ละครั้งจึงต้องมี ICMP Sequence Number เพื่อบอกลำดับของการตรวจสอบภายใต้ ICMP Identifier หนึ่ง ๆ ด้วย

การสื่อสารข้อมูลบนเครือข่ายอินเตอร์เน็ตในปัจจุบันได้ใช้เทคนิคของการส่งแพ็กเก็ตไปบนเครือข่าย โดยที่แอพพลิเคชั่น(Application)แต่ละชนิดต้องเลือกใช้เทคนิคการสื่อสารข้อมูลตามความเหมาะสม ซึ่งในแต่ละเทคนิคจะมีขบวนการรับส่งข้อมูลที่แตกต่างกัน

2.4 เทคนิคการตรวจจับข้อมูล(Capturing Technique)

ปัจจุบันการรวบรวมข้อมูลเครือข่ายได้กลายมาเป็นความสำคัญที่ต้องแสวงหาเครื่องมือ ที่จะนำมาวิเคราะห์ประสิทธิภาพการทำงานของเครือข่าย สิ่งที่สำคัญของการรวบรวมข้อมูล คือ การตรวจจับต้องไม่ทำให้เกิดความสูญเสียหรือมีผลกระทบต่อการสื่อสารข้อมูล ปัจจุบันการตรวจจับข้อมูลมี 2 วิธีที่ใช้โดยทั่วไป คือ ใช้เครื่องมือตรวจจับเป็นฮาร์ดแวร์(Hardware)โดยตรงหรือการพัฒนาซอฟท์แวร์(Software)ขึ้นโดยอาศัยคอมพิวเตอร์ส่วนบุคคลทั่วไป(PC Workstation) ให้ทำหน้าที่ตรวจจับแทน ซึ่งมีประสิทธิภาพต่ำกว่าวิธีการแรก แต่ความสามารถในการปรับปรุงให้ตรงกับความต้องการของผู้ใช้ทำได้ดีกว่า และสามารถพัฒนาขึ้นใช้เองได้

2.4.1 BPF บนระบบปฏิบัติการยูนิกซ์

จากที่นักวิจัย S.McCanne และ V.Jacobson [8] ได้ทำการพัฒนา BPF(Berkeley Packet Filter) ขึ้นในมหาวิทยาลัยแคลิฟอเนีย(University of California) ทำให้นำไปสู่การสร้างไดร์เวอร์(Driver)ตรวจจับข้อมูล(Capture Driver) ที่มีประสิทธิภาพขึ้นบนระบบปฏิบัติการยูนิกซ์(Unix) เรียกว่า Libpcap Library[9] และถูกนำไปสร้างคำสั่ง Tcpdump บนระบบปฏิบัติการยูนิกซ์ที่แพร่หลายและใช้งานกันโดยทั่วไป BPF Driver เป็นเครื่องมือที่ถูกเรียกใช้งานโดยโปรแกรมประยุกต์บนระบบปฏิบัติการยูนิกซ์เพื่อให้อ่านแพ็กเก็ตผ่านอีเทอร์เนตกราด(Ethernet Card) ที่เชื่อมต่ออยู่กับเครือข่าย(Network Adapter) ซึ่งต่างจากไดร์เวอร์ทั่วไปที่ไม่ได้ถูกควบคุมโดยตรงจากอุปกรณ์เชื่อมต่อ โครงสร้างของไดร์เวอร์ประกอบด้วยส่วนสำคัญ 2 หน่วยคือ

1. หน่วยเชื่อมต่อกับเครือข่าย(Network Tap) เป็นฟังก์ชั่นที่เขียนขึ้นเพื่อทำหน้าที่ร้องขอให้ตรวจจับแพ็กเก็ตจากเครือข่าย โดยหลีกเลี่ยงจากการใช้งานกับ BPF โดยตรง ส่วนนี้จะถูกเรียกใช้โดยแผงวงจรเชื่อมต่อเครือข่าย(Network Adapter) เพื่อตรวจจับการมาของ แพ็กเก็ต และทำการสำเนาข้อมูลให้กับหน่วยจัดเก็บข้อมูลของโปรแกรมประยุกต์(Listening Application) ถ้าแพ็กเก็ตผ่านการตรวจสอบโดยเงื่อนไขของหน่วยกรองข้อมูล

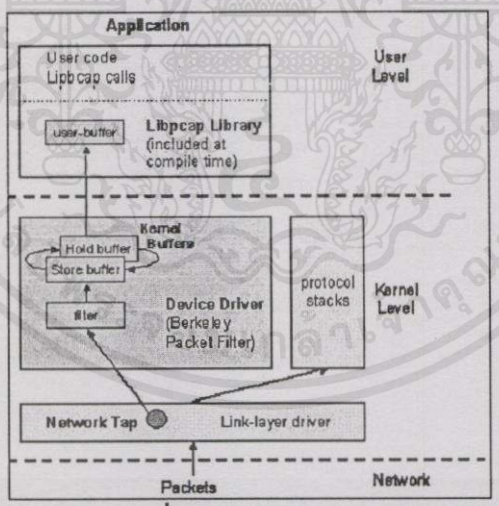
2. หน่วยกรองแพ็กเก็ต(Packet Filter) เป็นหน่วยที่ทำหน้าที่แยกชนิดแพ็กเก็ตโปรแกรมประยุกต์ส่วนใหญ่ได้ใช้ BPF ในการตรวจสอบและละทิ้ง(Reject) แพ็กเก็ตที่ไม่ได้รับความสนใจออกไปเพื่อเพิ่มประสิทธิภาพและความรวดเร็วในการวิเคราะห์ข้อมูล หน่วยแยกชนิดข้อมูลจะมีลักษณะเป็นฟังก์ชั่นที่ทำงานแบบตรรก(Boolean) นอกจากนี้ยังมีประโยชน์กับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยสำรองข้อมูล(Buffer)ด้วย เนื่องจากในบางครั้งบางโปรแกรมประยุกต์อาจให้ความสนใจเฉพาะส่วนหัวของแพ็กเก็ต โดยไม่สนใจในส่วนอื่นของข้อมูล ดังนั้นหน่วยแยกชนิดข้อมูลสามารถตัดเอาเฉพาะข้อมูลที่สำคัญจากส่วนหัว คัดลอกไปจัดเก็บบนหน่วยสำรองข้อมูล เพื่อรอการส่งต่อให้กับหน่วยจัดเก็บข้อมูลของโปรแกรมประยุกต์ต่อไปได้ ซึ่งจะทำให้การบริหารพื้นที่จัดเก็บข้อมูลมีประสิทธิภาพ และสามารถลดความสูญเสียหรือละทิ้งบางแพ็กเก็ตที่ไม่สามารถให้บริการการทำสำเนาข้อมูลได้ทันต่อการมาของแพ็กเก็ตลงได้

2.4.2 ภาพรวมการทำงานของ BPF

BPF มีส่วนสำรองข้อมูล(Buffer) 2 ส่วนที่มีความสำคัญต่อกระบวนการตรวจจับข้อมูลของทุกหน่วย การตรวจจับข้อมูลจะถูกเรียกโดยโปรแกรมประยุกต์ผ่าน BPF ผ่านไปยัง IOCTL หน่วยสำรองข้อมูลจะถูกเตรียมขึ้นโดย BPF ที่มีขนาด 4 KB ซึ่งหน่วยสำรองข้อมูลชุดแรกถูกเรียกว่า Store Buffer ซึ่งทำหน้าที่จัดเก็บข้อมูลที่ตรวจพบจากอุปกรณ์เชื่อมต่อเครือข่าย(Network Adapter) หน่วยที่สองเรียกว่า Hold Buffer ทำหน้าที่คัดลอกแพ็กเก็ตเพื่อส่งต่อให้กับโปรแกรมประยุกต์ โดยปกติแล้ว Hold Buffer จะคอยการสลับที่หรือคัดลอกข้อมูลจาก Store Buffer ในกรณีที่ Store Buffer บรรจุข้อมูลเต็มแล้ว ซึ่งกระบวนการทั้งหมดนี้จะไม่อยู่รวมกับส่วนของ Adapter Device Driver ดังแสดงได้ในภาพที่ 2.7



ภาพที่ 2.7 โครงสร้าง BPF

เมื่อแพ็กเก็ตมาถึงเครือข่ายในระดับ Link Layer แพ็กเก็ตจะถูกส่งต่อไปยังแต่ละชั้นของโปรโตคอล(Protocol Stack) BPF จะทำการเรียกใช้ฟังก์ชั่นในส่วนของ Network Tap เพื่อส่งต่อไปยังหน่วยกรองข้อมูลที่ได้กำหนดเงื่อนไขและจำนวน(Packet Bytes)ที่ต้องการจัดเก็บข้อมูลจากระดับผู้ใช้ไว้แล้ว(สามารถกำหนดเงื่อนไขจากการเรียกฟังก์ชั่นใน Libpcap Library) โดยในขณะที่ตรวจสอบนั้นข้อมูลจะยังไม่ถูกคัดลอกไปในส่วนของ Kernel Level เนื่องจากจะทำให้พื้นที่หน่วยความจำต้องเสียไปในกรณีที่แพ็กเก็ตนั้นถูกปฏิเสธจากเงื่อนไขในการยอมรับในระดับ

ของผู้ใช้ ในกรณีที่ข้อมูลผ่านเงื่อนไขในการกรองแล้วก็จะถูกคัดลอกเพื่อจัดเก็บไปไว้ในส่วนของStore Buffer และสลับไปไว้ที่ Hold Buffer เมื่อพื้นที่จัดเก็บเต็ม ในขณะเดียวกันก็จะมีอีกกระบวนการหนึ่งที่คอยอ่านข้อมูลจาก Hold Buffer เพื่อส่งต่อให้กับระดับของผู้ใช้ การรับข้อมูลจาก BPF ในระดับผู้ใช้สามารถอ่านข้อมูลได้มากกว่า 1 แพ็กเก็ตในแต่ละครั้ง แต่ละครั้ง BPF จะทำการห่อหุ้มข้อมูลเพิ่มเติมลงไป(Encapsulate) โดยมีส่วนข้อมูลที่สำคัญได้แก่ เวลาที่มาถึง(Time Stamp) ความยาว และลำดับของข้อมูล(Offset of Data Alignment) ตามลำดับ

ปัจจุบัน Libpcap Library ได้ถูกนำไปใช้โดยแพร่หลายในการสร้างเครื่องมือวัดซึ่งโดยทั่วไป ใช้กับการตรวจวัดด้วยวิธีดักจับจากข้อมูลสื่อสารจริง(Passive Traffic Measurement) ตัวอย่างเช่น การวัดข้อมูลโฟลว(Data Flow) ด้วย NetTraMet, NetFlow, Cflowd, FlowScan และ CoralReef ซึ่งเครื่องมือทั้ง 4 ชนิดแรกได้ถูกพัฒนาขึ้นจากการใช้ Libpcap Library ส่วน CoralReef ได้พัฒนาจาก Corallib Library ซึ่งพัฒนาต่อจาก Libpcap Library ในภายหลัง

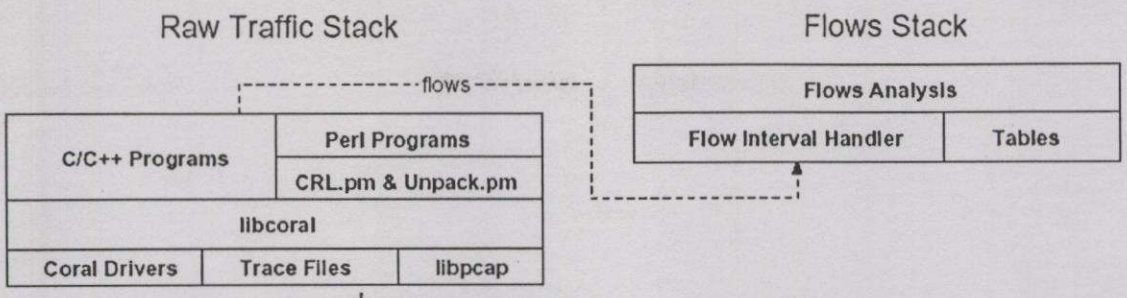
2.5 CoralReef ซอฟท์แวร์ตรวจวัดการสื่อสารบนเครือข่ายอินเตอร์เน็ต

CoralReef เป็นซอฟท์แวร์(Software)ที่พัฒนาขึ้น[10] โดยออกแบบให้มีโครงสร้างที่ประกอบด้วย 2 หน่วยงานที่สำคัญ คือ

1. หน่วยดักจับข้อมูลดิบ(Raw Traffic) เป็นส่วนที่ใช้ติดต่อกับ Protocol Stack (Packets หรือ Cells) มีหน้าที่ในการดักจับข้อมูลจากเครือข่าย โครงสร้างภายในประกอบด้วย LibCoral C library เป็นโปรแกรมที่สามารถเรียกใช้และเขียนเป็นแอพพลิเคชั่นได้ด้วยภาษา C, C++ หรือ Perl ภายใน LibCoral ประกอบด้วยส่วน API(Application Program Interface) สำหรับดักจับข้อมูลได้จากทั้ง ATM และ POS cards บางส่วนของ API สามารถอ่านแพ็กเก็ตจากแฟ้มข้อมูลได้หลากหลายรูปแบบ เช่น Coral Format ,NLANR Format, PCAP, DAG ATM และ POS Format นอกจากนี้ยังมีส่วนช่วยสำหรับตรวจสอบโครงสร้างข้อมูลเครือข่าย ได้แก่ โมดูล(Module)สำหรับตรวจสอบเพื่อกำหนดเส้นทางของ IP Address โดยใช้ BGP(Border Gateway Protocol) ภายใต้ AS(Autonomous System) หนึ่ง ๆ

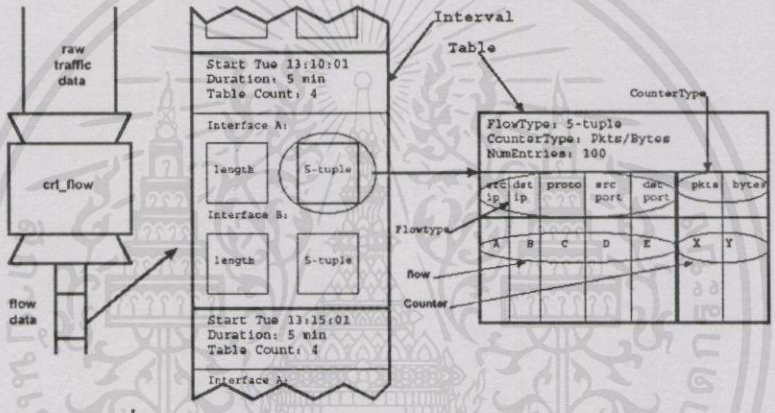
2. หน่วยจัดเก็บข้อมูลโฟลว(Flows Stack) เป็นส่วนที่ทำหน้าที่จัดการรวบรวมข้อมูลดิบให้เป็นหน่วยข้อมูลที่เรียกว่า "Flow Intervals" รูปแบบจัดเก็บเป็นตารางแจกแจงความถี่ข้อมูล มีหน่วยวัดข้อมูลอยู่ในรูปของ ผลรวมความยาวมีหน่วยเป็นไบต์(Bytes) จำนวนแพ็กเก็ต(Packets) ข้อมูลโฟลวของแต่ละสถานี(Host) ชนิดเภทของ IP Protocol และพอร์ตสื่อสาร(Port) ดังแสดงได้ในภาพที่ 2.8

เอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.8 โครงสร้างของ CoralReef Software

หน่วยวิเคราะห์ข้อมูลโฟลว(Flows Analysis) จะทำหน้ารวบรวมข้อมูลให้อยู่ในรูปที่ สามารถนำไปวิเคราะห์ต่อได้ โดยมีขั้นตอนจัดเก็บที่ประกอบด้วยข้อมูลที่สำคัญได้แก่ Source IP, Destination IP, Protocol, Source Port และ Destination Port ดังแสดงได้ในภาพที่ 2.9



ภาพที่ 2.9 การจัดเก็บข้อมูลโฟลวด้วย CoralReef

ข้อมูลที่ตรวจจับได้(Raw Traffic Data)จาก Interface จะถูกนำไปรวบรวมไว้ในหน่วย Flow Stack ซึ่งอยู่ในรูปตารางข้อมูลโฟลว(Flow Type) นอกจากนี้ขบวนการจัดเก็บยังมีหน่วยนับ(Counter Type)ที่สามารถสร้างตารางประมวลผลเพิ่มเติมได้ เช่น วิเคราะห์จำนวน ความยาว และ เวลาการมาถึงของข้อมูล ที่สามารถนำไปพัฒนาเพื่อสร้างรายงานข้อมูลได้ตามภาพที่ 2.10 และ 2.11

```
# 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
```

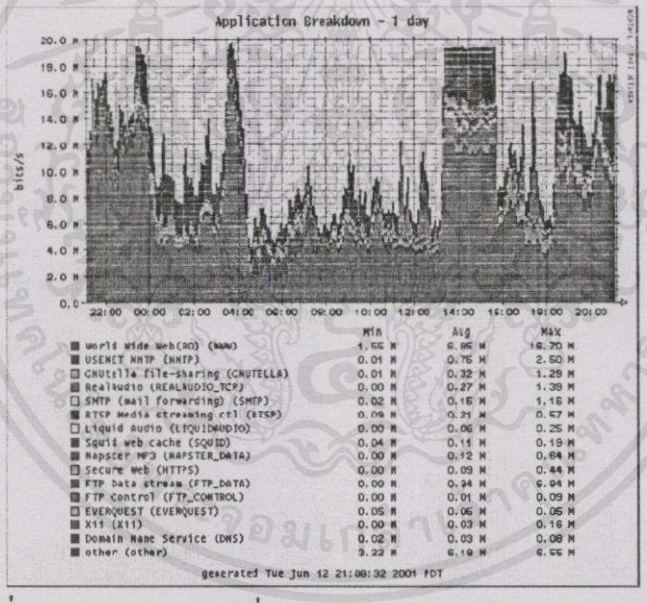
ภาพที่ 2.10 ตัวอย่างรายงานข้อมูลโฟลว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

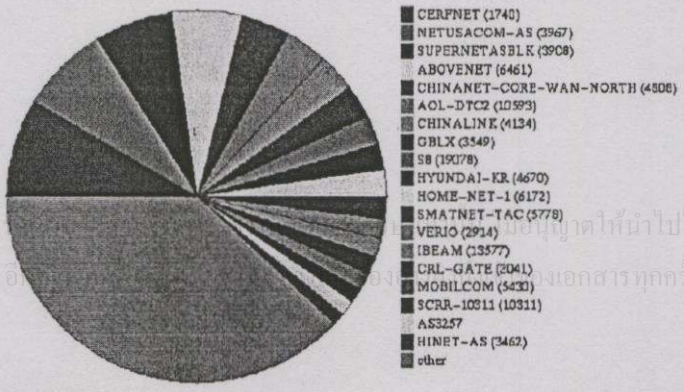
```
# time 1001975450.054545 (10.000000), packets lost: 0
# if[subif] v4pkts     v4bytes v6pkts v6bytes non_ip v4pkts/s v4bits/s v6pkts/s v6bits/s
0[135]          1          40      0       0      0     0.10    32.00     0.00     0.00
0[110]       2269     2536140      0       0      0    22.69     2.03M     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 2.11 ตัวอย่างรายงานจำนวนและอัตราการมาถึงของแพ็กเก็ต

จากการออกแบบ CoralReef ได้นำไปสู่การพัฒนาเครื่องมือวัดที่หลากหลายขึ้น โดยนำส่วนโครงสร้างในการจัดเก็บข้อมูลไปใช้สร้างกราฟ เพื่อแสดงและชี้ถึงคุณลักษณะการสื่อสารข้อมูลได้หลากหลายชนิด ยกตัวอย่างเช่น งานวิจัยของ David Moore ในเรื่อง "The CoralReef software suite as a tool for system and network administrators" ได้นำไปใช้จำแนกชนิดของข้อมูลโฟลวที่แสดงได้ในภาพที่ 2.12 และ 2.13



ภาพที่ 2.12 ตัวอย่างกราฟที่จำแนกคุณลักษณะข้อมูลในแต่ละชนิด

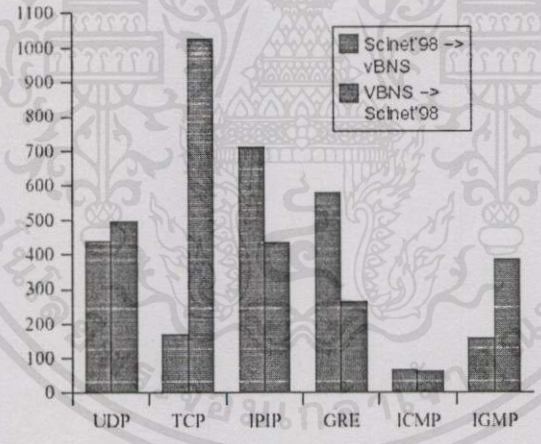


เอกสารนี้เป็นเอกสาร... ...ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อ... ...เอกสารทุกครั้งที่มีการนำไปใช้

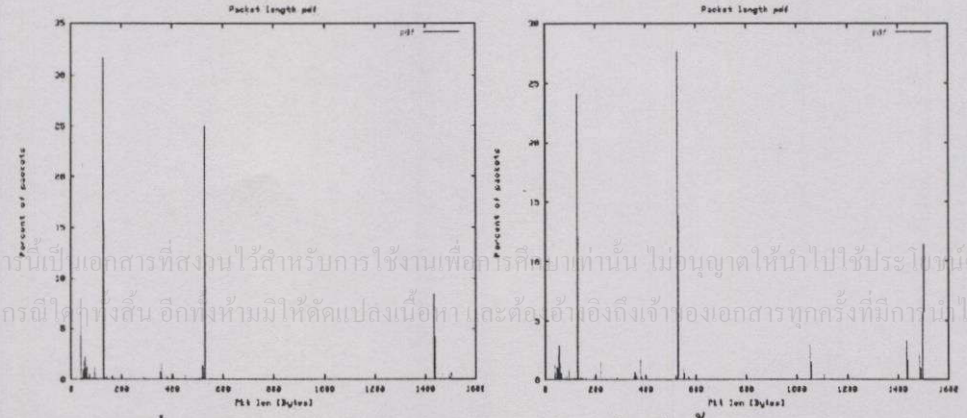
ภาพที่ 2.13 ตัวอย่างกราฟที่จำแนกอัตราส่วนข้อมูลโฟลวในแต่ละชนิด

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

Coral Library ได้พัฒนาจนเป็นเครื่องมือดักจับข้อมูลที่มีประสิทธิภาพ สามารถนำไปใช้พัฒนาเครื่องมือตรวจวัดอื่น ๆ ได้โดยแพร่หลาย ซึ่งนอกจากข้อมูลโฟลวแล้วยังมีงานวิจัยอื่นที่ใช้ Coral Library ในการช่วยสนับสนุนการวิเคราะห์และสะท้อนปัญหาที่เกิดขึ้นกับเครือข่ายการสื่อสารได้ เช่น งานวิจัยของ Brynjar Age Viken[11] เรื่อง "Passive monitor of Internet Traffic at Supercomputer'98" ได้นำไปใช้ตรวจจับคุณลักษณะ ความยาว และเวลาระหว่างการมาของแพ็กเก็ต โดยใช้ร่วมกับอุปกรณ์ Optical Splitter คือ OC3 ทำงานกับหน่วยตรวจจับข้อมูลที่เป็นระบบปฏิบัติการ FreeBSD บนเครื่องคอมพิวเตอร์ที่มี CPU เป็น Intel Pentium II ความเร็ว 400 MHz มีหน่วยความจำสำรองขนาด 128 MB ตรวจจับข้อมูลจากคู่ Interface ลงบน SCSI Hard Disk โดยพัฒนาโปรแกรมด้วยภาษา C++ ให้ทดลองตรวจจับข้อมูลจริงที่มีการสื่อสารข้อมูลระหว่างเครือข่าย SCiNet'98 และ vBNS ในรัฐฟอริดา สหรัฐอเมริกา ผลการใช้งานสามารถดักจับข้อมูลบนเครือข่ายจริงได้ โดยนำข้อมูลไปจัดเก็บบน Hard Disk และสร้างรายงานการสื่อสารข้อมูลแต่ละชนิด โดยสามารถจำแนกการสื่อสารข้อมูลได้เป็น 2 ทิศทางคือ การสื่อสารขาไปจาก SCiNet'98 ไปยัง vBNS และ ขากลับจาก vBNS ไปยัง SCiNet'98 ดังแสดงได้ในภาพที่ 2.14, 2.15 และ 2.16



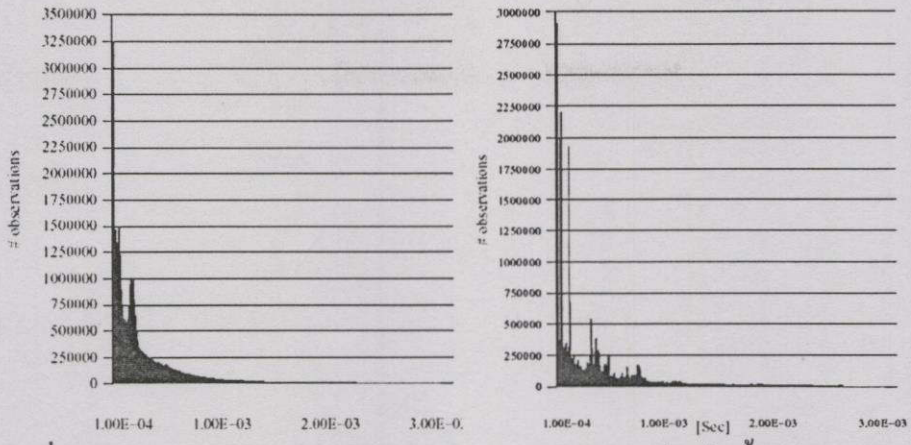
ภาพที่ 2.14 ตัวอย่างกราฟวิเคราะห์ปริมาณของแพ็กเก็ตแต่ละชนิดทั้งขาไปและกลับ



ภาพที่ 2.15 ตัวอย่างกราฟแจกแจงความยาวแพ็กเก็ตทั้งขาไปและกลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

61223



ภาพที่ 2.16 ตัวอย่างกราฟแจกแจงเวลาระหว่างการมาของแพ็กเก็ตทั้งขาไปและกลับ

จากการศึกษาวิธีการออกแบบเครื่องมือโดยใช้วิธีการดักจับข้อมูลจากเครือข่ายจริง ด้วย CoralReef พบว่า วิธีการพัฒนาเครื่องมือได้ใช้เทคนิคการแบ่งหน่วยประมวลผลไปดักจับข้อมูล ทำให้การวิเคราะห์ข้อมูลเป็นอิสระ สามารถสร้างรูปแบบรายงานได้หลากหลายโดยใช้ฐานข้อมูลจากหน่วยเดียวกันเรียกว่า Flow Stack ดังนั้นด้วยเทคนิคนี้จึงมีแนวคิดที่จะนำไปใช้กับการพัฒนาเครื่องมือตรวจวัดพฤติกรรมการสื่อสารข้อมูลอื่น ๆ ที่มีความจำเป็นต่อการนำไปวิเคราะห์ความบกพร่องหรือปัญหาที่เกิดจากการสื่อสารข้อมูล ได้แก่ ความยาว เวลาระหว่างการมา เวลาการสื่อสารข้อมูลไปกลับ เวลาการสื่อสารข้อมูลผ่านตัวระบบ และความสูญเสียข้อมูลบนตัวระบบ ซึ่งเครื่องมือที่พัฒนาขึ้นนี้จะสามารถทำงานได้อย่างต่อเนื่อง และหลีกเลี่ยงการจัดเก็บข้อมูลบน Hard Disk ซึ่งเป็นข้อจำกัดที่เกิดขึ้นจากงานวิจัยของ Brynjar Age Viken

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

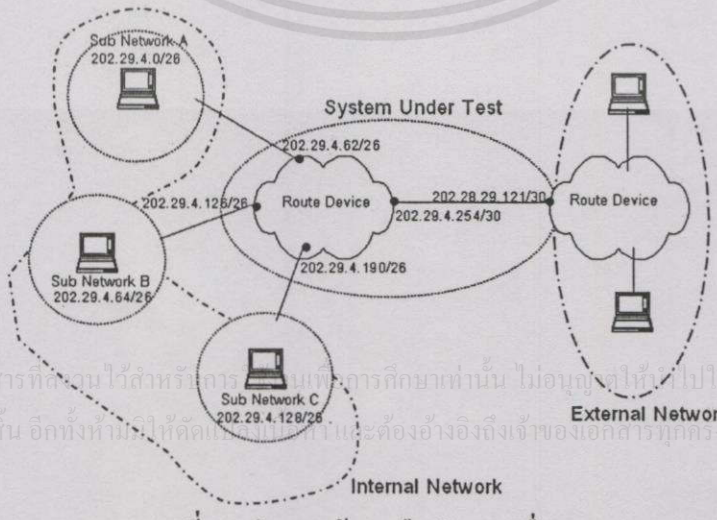
บทที่ 3
การออกแบบวิธีตรวจวัดข้อมูลเครือข่าย

ข้อด้วยของการพัฒนาเครื่องมือตรวจวัดแบบ CoralReef คือ โครงสร้างข้อมูลที่ใช้ในการวิเคราะห์หรือที่เรียกว่า Flows Stack นั้นเอื้อต่อการนำไปใช้ตรวจวัดข้อมูลโฟลวเท่านั้น แต่ก็มีลักษณะเด่นคือหน่วยที่ทำหน้าที่ตรวจจับ และวิเคราะห์ข้อมูลเป็นอิสระจากกัน ทำให้สามารถเพิ่มหน่วยประมวลผลในด้านการวิเคราะห์ความยาว และเวลาระหว่างการมาของข้อมูลเข้าไปได้โดยง่าย ส่วน Corallib Library ที่ใช้ตรวจวัดคุณลักษณะข้อมูล ความยาว และเวลาระหว่างการมาก็มีข้อด้อยที่ต้องจัดเก็บข้อมูลบน Hard Disk ก่อน ทำให้ผลการวิเคราะห์ขาดความต่อเนื่อง แต่ก็มีลักษณะเด่นคือ สามารถวิเคราะห์พฤติกรรมได้ตามทิศทางการสื่อสารข้อมูล

ในบทที่ 3 นี้จะกล่าวถึงการออกแบบวิธีการตรวจวัด ความยาว เวลาระหว่างการมา เวลาการสื่อสารข้อมูลไปกลับ เวลาการสื่อสารข้อมูลผ่านตัวระบบ และการสูญเสียข้อมูลบนตัวระบบ ด้วยเทคนิคแบบพาสซีฟโดยใช้ Libpcap Library ในการตรวจจับข้อมูล มีลักษณะเด่นคือ ทำการแบ่งหน่วยประมวลผลเพื่อตรวจจับ แยกชนิด และวิเคราะห์ข้อมูลให้เป็นอิสระจากกัน เพิ่มขบวนการจัดโครงสร้างข้อมูลให้เหมาะกับการวิเคราะห์ข้อมูลที่หลากหลายขึ้น รวมทั้งหลีกเลี่ยงการจัดเก็บข้อมูลบน Hard Disk และใช้การจัดเก็บข้อมูลบนหน่วยความจำเป็นหลักเพื่อการวิเคราะห์ข้อมูลและสร้างผลลัพธ์การแจกคุณลักษณะข้อมูล

3.1 หลักการตรวจวัดข้อมูล

โดยทั่วไปการเชื่อมต่อการสื่อสารข้อมูลบนเครือข่าย สามารถแสดงได้ดังภาพที่ 3.1



ภาพที่ 3.1 โครงสร้างเครือข่ายการสื่อสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 3.1 เป็นโครงสร้างการสื่อสารข้อมูลทั่วไป ที่มีการเชื่อมต่อการสื่อสารข้อมูลระหว่างเครือข่ายภายใน(Internal Network) และเครือข่ายภายนอก(External Network) เมื่อมีการสื่อสารข้อมูลจากโนดที่อยู่ภายในเครือข่ายย่อย (Sub Network)ไปยังโนดอื่น ๆ ข้อมูลจะถูกส่งไปยังอุปกรณ์สืบค้นเส้นทาง เพื่อตรวจสอบหมายเลขไอพีที่ติดมากับแพ็กเก็ตและส่งต่อไปตามนโยบายที่กำหนดขึ้นโดยผู้ดูแลเครือข่าย ซึ่งแพ็กเก็ตอาจถูกส่งไปยังโนดปลายทางที่มีอยู่ทั้งภายในหรือภายนอกเครือข่าย

อุปกรณ์สืบค้นเส้นทางจะมีลักษณะเช่นเดียวกับโนดสื่อสารอื่น ๆ คือ มีหมายเลขไอพีและอาจมีได้มากกว่าหนึ่งหมายเลขเพื่อแทนค่าช่องทางการสื่อสารข้อมูล และต้องมีอย่างน้อยหนึ่งหมายเลขที่ใช้สำหรับเชื่อมต่อกับเครือข่ายภายใน และอีกหนึ่งหมายเลขสำหรับเชื่อมต่อข้อมูลไปยังฝั่งเครือข่ายภายนอก ดังนั้นด้วยลักษณะการจัดการเครือข่ายนี้ สามารถจัดกลุ่มของสมาชิกโนดต่าง ๆ ได้เป็น 3 กลุ่มคือ

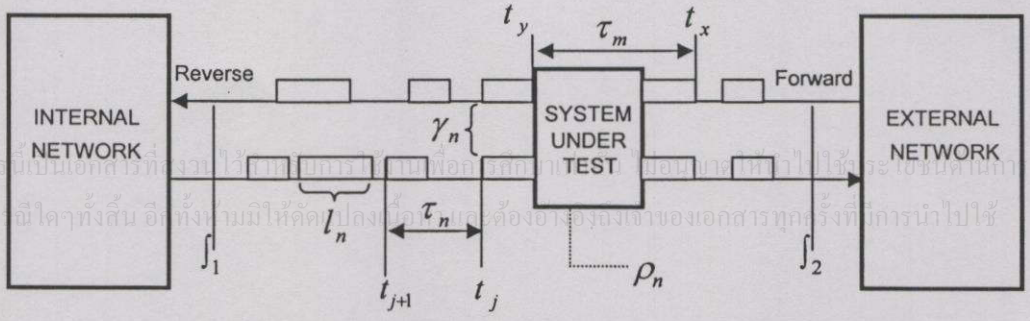
1. กลุ่มสมาชิกเครือข่ายภายใน(Internal Network Node) หมายถึง กลุ่มของโนดผู้ดูแลเครือข่าย โดยมีหมายเลขไอพีอ้างอิงไปถึงอุปกรณ์สืบค้นเส้นทางเดียวกัน ได้แก่ กลุ่มเครือข่ายย่อย 202.29.4.0/26, 202.29.4.64/26 และ 202.29.4.128/26

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. กลุ่มสมาชิกเครือข่ายภายนอก(External Network Node) หมายถึง กลุ่มของโนดที่ใช้หมายเลขไอพีแตกต่างจาก 2 กลุ่มแรก

3.1.1 คุณลักษณะที่ต้องการตรวจวัดและทิศทางการสื่อสารข้อมูล

พิจารณารูปแบบโครงสร้างเครือข่าย ที่มีการสื่อสารข้อมูลด้วยแพ็กเก็ตจากโนดต้นทางที่อยู่ฝั่งเครือข่ายภายใน(Internal Network)ส่งผ่านตัวระบบ(System Under Test) ไปยังโนดปลายทางที่อยู่ฝั่งเครือข่ายภายนอก(External Network) เมื่อทำการยุบรวมเครือข่ายย่อยแล้วจะสามารถนำไปพิจารณาพฤติกรรมที่เกี่ยวข้องกับการตรวจวัดข้อมูลในการวิจัยได้ ดังภาพที่ 3.2



ภาพที่ 3.2 พารามิเตอร์ที่เกี่ยวข้องกับการตรวจวัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ l_n หมายถึง ความยาวของแพ็กเก็ตใด ๆ ไม่รวมส่วนหัวของ Ethernet Header มีหน่วยในการวิเคราะห์เป็นไบต์(Bytes)

τ_n หมายถึง ผลต่างของเวลาระหว่างแพ็กเก็ตแรกกับแพ็กเก็ตถัดไป($|t_j, t_{j+1}|$) ที่เดินทางถึงตัวระบบ หรือนิยามเรียกว่า เวลาระหว่างการมา(Inter-Arrival Times) มีหน่วยในการวิเคราะห์เป็นวินาที

γ_n หมายถึง ผลต่างของเวลาระหว่างแพ็กเก็ตเรียก(Request) และตอบ(Response) ($|t_j, t_y|$) หรือนิยามเรียกว่า เวลาการสื่อสารข้อมูลไปกลับ(Response Times) มีหน่วยในการวิเคราะห์เป็นวินาที

τ_m หมายถึง ผลต่างของเวลาที่แพ็กเก็ตเดียวกันผ่านเข้าและออกจากตัวระบบ ($|t_x, t_y|$) มีหน่วยการวิเคราะห์เป็นวินาที

ρ_n หมายถึง แพ็กเก็ตเดียวกันที่ฝั่งขาเข้าและไม่พบในฝั่งขาออกจากตัวระบบ มีหน่วยวิเคราะห์เป็นจำนวนแพ็กเก็ต

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

นอกจากนี้ในงานวิจัยได้กำหนดให้พารามิเตอร์ในแต่ละชนิด สามารถแสดงผลลัพธ์โดยจำแนกทิศทางการสื่อสารข้อมูลได้ 2 ลักษณะ คือ ทิศทางขาไปและ ขากลับ

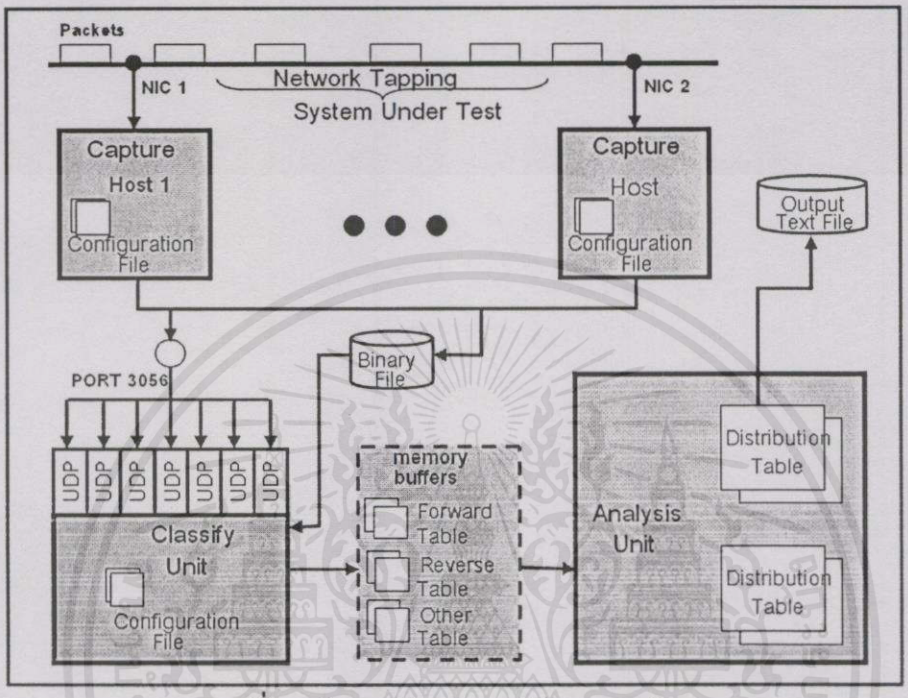
3.1.2 การออกแบบโครงสร้างวิธีการตรวจวัดข้อมูล

เพื่อให้การประมวลผลเป็นอิสระจากกัน ดังนั้นจึงออกแบบให้แยกหน่วยทำงานที่สำคัญออกจากกัน แต่ละหน่วยจะสามารถทำงานได้ด้วยการโปรแกรมแบบแบ่งเวลาประมวลผล ซึ่งเป็นคุณสมบัติเด่นที่สามารถพัฒนาเครื่องมือได้โดยสอดคล้องกับการทำงานบนระบบปฏิบัติการ FreeBSD โดยแบ่งโครงสร้างการทำงานเป็น 3 หน่วยประมวลผล คือ

1. หน่วยตรวจจับข้อมูล(Capture Unit) มีหน้าที่ตรวจจับและรวบรวมแพ็กเก็ตบนเครือข่าย เพื่อส่งต่อให้กับหน่วยแยกชนิดข้อมูล(Classify Unit) สามารถทำได้ 2 ลักษณะ คือ ส่งข้อมูลผ่านเครือข่ายโดยตรง โดยใช้แพ็กเก็ตชนิด UDP(User Datagram Protocol) หรือจัดเก็บข้อมูลตามโครงสร้างที่ออกแบบไว้บน Hard Disk เพื่อรอการประมวลผลจากหน่วยแยกชนิดในภายหลัง

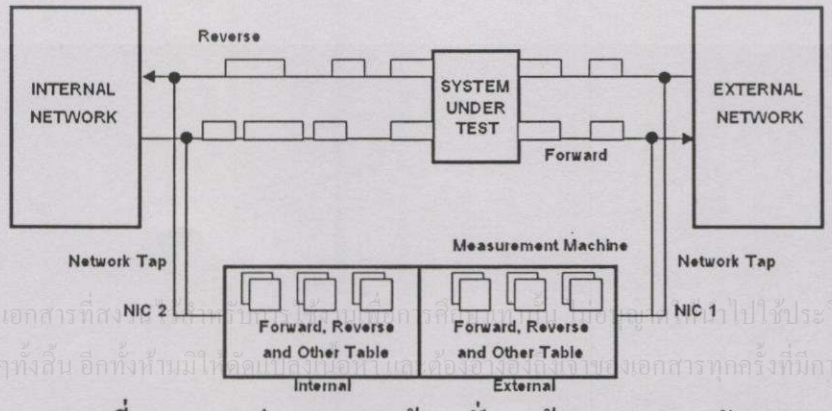
2. หน่วยแยกชนิดข้อมูล(Classifying Unit) ทำหน้าที่คำนวณเวลา และแยกชนิดข้อมูลที่ได้จากหน่วยตรวจจับตามทิศทางการสื่อสารข้อมูล โดยจัดเก็บในรูปตารางฐานข้อมูลบนหน่วยความจำที่สามารถใช้ร่วมกันได้(Shared memory buffers)จากหน่วยประมวลผลอื่น ๆ ซึ่งตารางข้อมูลแบ่งออกเป็น 3 ลักษณะตามจำนวนอีเทอร์เนตกราดที่ใช้ตรวจจับข้อมูลบนเครือข่าย ได้แก่ ตารางจัดเก็บข้อมูลตามทิศทางขาไป(Forward Table) ใช้จัดเก็บข้อมูลที่ทำหน้าส่งจาก

เครือข่ายภายใน(Internal Network)ผ่านตัวระบบไปยังเครือข่ายภายนอก(External Network) ตารางจัดเก็บข้อมูลในทิศทางขากลับ(Reverse Table) ใช้จัดเก็บข้อมูลที่ทำหน้าที่ส่งจากเครือข่ายภายนอกผ่านตัวระบบ ไปยัง เครือข่ายภายใน และตารางจัดเก็บข้อมูลอื่น ๆ (Other Table) ใช้จัดเก็บข้อมูลที่ต่างจากไปจากสองตารางแรก ดังแสดงในภาพที่ 3.3



ภาพที่ 3.3 โครงสร้างหน่วยประมวลผลข้อมูล

จำนวนตารางข้อมูลจะขึ้นอยู่กับจำนวนจุดเชื่อมต่อของหน่วยตรวจจับข้อมูล เช่น ถ้ามีจุดเชื่อมต่อ 2 จุด ทำหน้าที่ตรวจจับข้อมูลบนฝั่งเครือข่ายขาเข้า(Internal)และเครือข่ายฝั่งขาออก(External)จากตัวระบบ ข้อมูลแต่ละฝั่งจะถูกแยกชนิดเป็นตารางข้อมูลที่เหมือนกัน จากนั้นหน่วยวิเคราะห์ก็จะนำข้อมูลจากตารางไปประมวลผลร่วมกัน ดังแสดงได้ในภาพที่ 3.4



ภาพที่ 3.4 การแบ่งตารางฐานข้อมูลฝั่งขาเข้าและออกจากตัวระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 3.4 หน่วยตรวจจับและแยกชนิดข้อมูลสามารถติดตั้งอยู่บนคอมพิวเตอร์เครื่องเดียวกันได้(Measurement Machine) การตรวจจับข้อมูลจะขึ้นอยู่กับจำนวนของ NIC(Network Interface Card)ที่ติดตั้งบนเครื่องคอมพิวเตอร์ แต่ละ NIC จะทำการตรวจจับข้อมูลและนำไปแบ่งแยกเพื่อจัดเก็บเป็นตารางข้อมูลในแต่ละชนิด ได้แก่ ตารางจัดเก็บข้อมูลขาไป(Forward Table) ตารางจัดเก็บข้อมูลขากลับ(Reverse Table) และตารางจัดเก็บข้อมูลอื่น ๆ(Other Table)

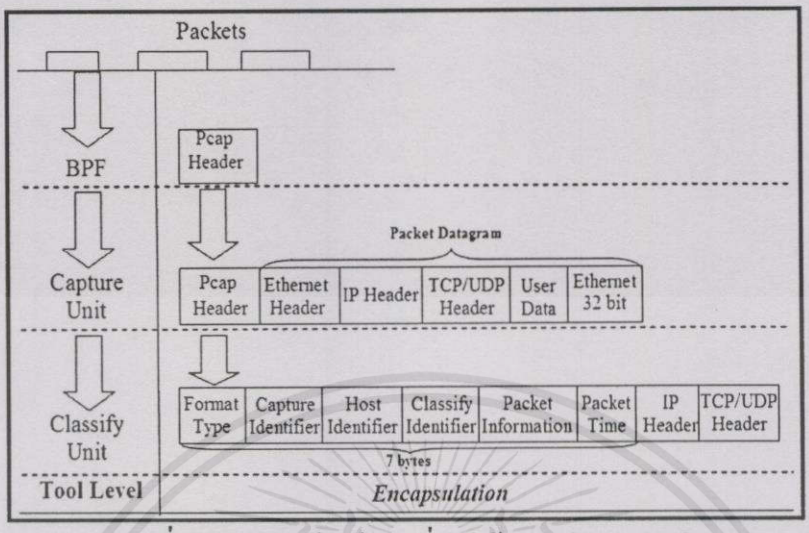
3. หน่วยวิเคราะห์ข้อมูล(Analysis Unit) มีหน้าที่นำข้อมูลจากตารางฐานข้อมูลที่ถูกแยกชนิดแล้วไปวิเคราะห์พฤติกรรมการสื่อสารตามคุณลักษณะต่าง ๆ และจัดเก็บผลลัพธ์ในรูปของตารางแจกแจงความถี่(Distribution Table)บนหน่วยความจำ เมื่อผู้ใช้หยุดการตรวจวัดข้อมูลตารางแจกแจงความถี่ทั้งหมดจะถูกนำไปบันทึกเก็บไว้เป็นแฟ้มข้อมูล เพื่อนำไปใช้วิเคราะห์ในเชิงสถิติต่อไป ซึ่งขบวนการภายในของหน่วยวิเคราะห์ยังมีการแบ่งแยกเป็นหน่วยประมวลผลย่อย(Child Process) ให้ทำหน้าที่วิเคราะห์คุณลักษณะในแต่ละชนิดไปพร้อมกัน ได้แก่ หน่วยวิเคราะห์ความยาว เวลาระหว่างการมา เวลาการสื่อสารข้อมูลผ่านตัวระบบ เวลาการสื่อสารข้อมูลไปกลับ และความสูญเสียข้อมูลบนตัวระบบ เพื่อให้เครื่องมือนี้สามารถเลือกตรวจวัดเฉพาะคุณลักษณะใดก็ได้โดยเป็นอิสระจากกัน

3.2 การตรวจจับข้อมูลบนเครือข่าย

หน่วยรวบรวมข้อมูลได้ถูกออกแบบให้มีได้หลายหน่วย สามารถตรวจจับข้อมูลได้เท่ากับจำนวนของ NIC(Network Interface Card)ที่มีอยู่บนตัวระบบ ดังนั้นก่อนการวิเคราะห์จึงต้องสร้างข้อตกลงร่วมกันระหว่างหน่วยประมวลผลต่าง ๆ ขึ้นก่อน และการที่หน่วยตรวจจับข้อมูลอยู่บนคอมพิวเตอร์ต่างเครื่องกันจึงทำให้มีผลกระทบต่อการวัดค่าเวลาที่แตกต่างกัน ในงานวิจัยนี้จึงออกแบบให้ใช้วิธีที่หน่วยตรวจจับข้อมูลทำงานอยู่บนคอมพิวเตอร์เครื่องเดียวกันไปก่อน เพื่อรับประกันความถูกต้องจากการกำกับเวลาข้อมูล(Time Stamp)

ข้อมูลที่ตรวจจับได้จากเครือข่ายอาจมีความยาวที่ไม่เท่ากัน ขึ้นอยู่กับแพ็กเก็ตเหล่านั้นว่าทำหน้าที่ใดไปยังปลายทาง แพ็กเก็ตที่ทำหน้าที่ลำเลียงข้อมูลอาจมีความยาวที่สูงกว่าแพ็กเก็ตชนิดอื่น ดังนั้นเพื่อให้เกิดความรวดเร็วในการประมวลผลก่อนที่จะส่งต่อไปยังหน่วยแยกชนิดจึงต้องออกแบบให้มีเฉพาะส่วนที่จำเป็นต่อการนำไปวิเคราะห์เท่านั้น คือ นำไปใช้เฉพาะส่วนหัวของแพ็กเก็ต (Packet Header) นอกจากนี้เพื่อให้หน่วยแยกชนิดสามารถจำแนกที่มาของข้อมูลว่ามาจาก NIC บนหน่วยตรวจจับข้อมูลใด จึงต้องผนวกส่วนหัวที่ประกอบด้วยรายละเอียดของหน่วยตรวจจับข้อมูล(Capture Header)ไปด้วย โดยแพ็กเก็ตที่ตรวจจับได้จากเครือข่ายด้วยขบวนการของ BPF/Libpcap ที่ประกอบด้วย Ethernet Header, IP Header, TCP หรือ UDP Header, User Data และ Ethernet 32 bit แต่ละส่วนจะถูกนำไปสร้างเป็นโครงสร้างข้อมูลใหม่โดยนำไปผนวกเฉพาะส่วน IP Header และ TCP/UDP Header เท่านั้น ซึ่งส่วนหัวข้อมูลที่ใช้

ห่อหุ้มเพิ่มเติมลงไปได้ออกแบบให้ใช้เนื้อที่ขนาด 7 ไบต์ เพื่อบรรจุรายละเอียดของหน่วยตรวจจับข้อมูล ดังแสดงได้ในภาพที่ 3.5



ภาพที่ 3.5 โครงสร้างข้อมูลที่ส่งให้กับหน่วยแยกชนิด

การห่อหุ้มประกอบด้วยข้อมูลที่สำคัญ ตามลำดับดังนี้

1. Format Type มีขนาด 8 bits ใช้บอกรูปแบบของข้อมูลที่ต้องการจัดส่งให้กับหน่วยแยกชนิด กำหนดไว้เพียงรูปแบบเดียวคือ เป็น Binary โดยตั้งค่าไว้เป็น 0

2. Capture Identifier มีขนาด 8 bits ใช้บอกตำแหน่งของหน่วยตรวจจับข้อมูล ซึ่งจะถูกนำไปใช้ในหน่วยแยกชนิด เพื่อระบุว่าที่มาของข้อมูลที่ได้จากฝั่งเครือข่ายภายใน(Internal Network) หรือฝั่งเครือข่ายภายนอก(External Network) หรือใช้แทน ลำดับที่ ของ Interface Card ที่เชื่อมต่อกับเครือข่าย

3. Host Identifier มีขนาด 8 bits ใช้บอกที่มาของข้อมูลว่ามาจากคอมพิวเตอร์เครื่องใด

4. Classifying Identifier มีขนาด 8 bits ใช้บอกสถานะ การแยกชนิดข้อมูล ตั้งค่าไว้ 3 ชนิด คือ 0 หมายถึง ยังไม่มีการแยกชนิด, 1 หมายถึง เป็นข้อมูลที่มีทิศทางการสื่อสารขาออก (Forward Direction), 2 หมายถึง เป็นข้อมูลที่มีทิศทางการสื่อสารขาเข้า(Reverse Direction) และ 255 หมายถึง เป็นข้อมูลอื่น ๆ ที่แตกต่างจากสถานภาพ 1 และ 2

5. Packet Information มีขนาด 8 bits ใช้แทนค่ารูปแบบการส่งข้อมูลไปยังหน่วยแยกชนิด ตั้งค่าได้ตามสถานภาพต่าง ๆ คือ 0 หมายถึง ส่งเฉพาะ Capture Information Header, 1

6. หมายถึง ผนวกส่วน IP Header ไปด้วย และ 3 หมายถึง ผนวกส่วน TCP/UDP Header ไปด้วย

7. Packet Time Stamp มีขนาด 16 bytes ใช้บอกเวลาที่แพ็กเก็ตถูกตรวจจับได้จากขบวนการของ Libpcap ที่มีลักษณะข้อมูลคือ

เอกสารนี้มีลิขสิทธิ์สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
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 เป็นตัวแปรแทนค่า เวลาที่กำกับข้อมูล

caplen เป็นตัวแปรแทนค่า ความยาวข้อมูลรวมที่ pcap ตรวจจับได้

length เป็นตัวแปรแทนค่า ความยาวของแพ็กเก็ตที่ตรวจจับได้

ตัวแปร ts จะถูกนำไปแทนค่าให้กับ Packet Time Stamp ในส่วนหัวของข้อมูลที่ประกอบขึ้นตามโครงสร้างของ timeval ที่มีลักษณะข้อมูลคือ

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

โดยที่ tv_sec เป็นตัวแปรที่ใช้เก็บหน่วยเวลาในรูปของวินาที(รวม ชั่วโมง นาที และวินาที) ณ เวลาปัจจุบัน

tv_usec เป็นส่วนที่ใช้เก็บหน่วยเวลาที่ต่ำกว่า วินาที คือเป็น ไมโครวินาที (Microseconds) ณ เวลาปัจจุบัน

3.3 การแยกชนิดข้อมูลเครือข่าย

3.3.1 การออกแบบหน่วยแยกชนิดและโครงสร้างจัดเก็บข้อมูล

หน่วยแยกชนิดข้อมูล(Classifying Unit)เป็นหน่วยที่ทำมีหน้าที่แยกประเภทของข้อมูลสื่อสารไว้บนตารางตามทิศทางการสื่อสารข้อมูล ได้แก่ ตารางข้อมูลขาไป(Forward Table) ตารางข้อมูลขากลับ(Reverse Table) และตารางข้อมูลอื่น ๆ (Other Table)

ตารางข้อมูลได้ออกแบบให้จัดเก็บไว้บนหน่วยความจำของเครื่องคอมพิวเตอร์ โดยรูปแบบการจัดเก็บข้อมูลของทุกตารางมีลักษณะที่เหมือนกัน คือรวบรวมเฉพาะส่วนที่มีความสำคัญต่อการนำไปใช้ในการวิเคราะห์เท่านั้น ซึ่งส่วนใหญ่ได้มาจากส่วนหัวของแพ็กเก็ตที่ใช้บอกคุณลักษณะข้อมูลที่อยู่ในชั้น Network และ Transport Layer เป็นหลัก สำหรับข้อมูลในทุกระเบียนเมื่อถูกแยกชนิดแล้วจะถูกจัดเก็บไว้บนบนตารางข้อมูลแต่ละชนิด โดยใช้วิธีบริหารจัดการแบบหน่วยความจำร่วม(Share Memory) เพื่อรอการเรียกใช้จากหน่วยวิเคราะห์ข้อมูลอื่น ๆ ซึ่งแต่ละระเบียนมีโครงสร้างที่ประกอบด้วยลักษณะที่แสดงได้ในตารางที่ 3.1

ตารางที่ 3.1 โครงสร้างตารางจัดเก็บข้อมูล

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

ผลรวมขนาดของแต่ละระเบียนในทุกตารางข้อมูลมีขนาดเท่ากันคือ 63 ไบต์(Bytes) แต่ละตารางจองพื้นที่จัดเก็บข้อมูลไว้บนหน่วยความจำเป็นจำนวนคงที่ คือ 10,000 ระเบียน ดังนั้นผลรวมของการใช้หน่วยความจำสำหรับจัดเก็บข้อมูลแต่ละชนิดจึงมีขนาดเท่ากับ 3.7 MB ในทุกตาราง ซึ่งทำให้การวิเคราะห์ข้อมูลด้วยเครื่องมือวิจัย สามารถทำงานได้บนคอมพิวเตอร์ส่วนบุคคลที่มีหน่วยความจำต่ำสุดเพียง 16 MB

3.3.2 ขั้นตอนวิธีการแยกชนิดข้อมูล

ข้อมูลที่ตรวจจับได้จะถูกกรองเฉพาะที่มีชนิดเป็นไอพีแพ็กเก็ตเท่านั้น โดยนำไอพีต้นทาง(Source IP)และปลายทาง(Destination IP)ของแพ็กเก็ต ไปทำการทดสอบเงื่อนไขเพื่อแยกจัดเก็บข้อมูลตามตารางต่าง ๆ โดยมีเงื่อนไขที่สำคัญดังนี้

เงื่อนไขที่ 1 กรณีที่ไอพีปลายทางของแพ็กเก็ตไม่ใช่ไอพีที่ใช้อยู่บนตัวระบบ หรือนิยามสัญลักษณ์ด้วย SUT(System Under Test) และไอพีต้นปลายทางไม่ใช่ไอพีที่ใช้อยู่บนเครือข่ายภายใน หรือนิยามสัญลักษณ์ด้วย INT(Internal Network) ให้ถือว่าเป็นแพ็กเก็ตที่ทำหน้าที่ส่งข้อมูลออกจากเครือข่ายภายใน ดังนั้นต้องจัดเก็บข้อมูลไว้บนตารางข้อมูลในทิศทางขาไป

เงื่อนไขที่ 2 กรณีที่ไอพีต้นทางของแพ็กเก็ตไม่ใช่ไอพีที่ใช้อยู่บนตัวระบบและเครือข่ายภายใน รวมถึงไอพีปลายทางของแพ็กเก็ตไม่ใช่ไอพีที่ใช้อยู่บนเครือข่ายภายใน ให้ถือเป็นแพ็กเก็ตที่ทำหน้าที่ส่งข้อมูลมาจากเครือข่ายภายนอก ดังนั้นต้องจัดเก็บข้อมูลไว้บนตารางข้อมูลในทิศทางขากลับ

นอกเหนือไปจากเงื่อนไขก่อนหน้า ให้ถือว่าเป็นไอพีแพ็กเก็ตอื่น ๆ ที่ยังไม่ได้รับความสนใจ เช่น ไอพีแพ็กเก็ตที่รับส่งข้อมูลกันเฉพาะภายในเครือข่าย หรืออาจเป็นไอพีแพ็กเก็ตที่สื่อสารกันระหว่างเครือข่ายภายในกับตัวระบบ ข้อมูลเหล่านี้ให้เก็บไว้ในตารางข้อมูลอื่น ๆ เพื่อรอการวิเคราะห์ต่อไป ดังแสดงได้จากขั้นตอนวิธีในภาพที่ 3.6

```
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 */
```

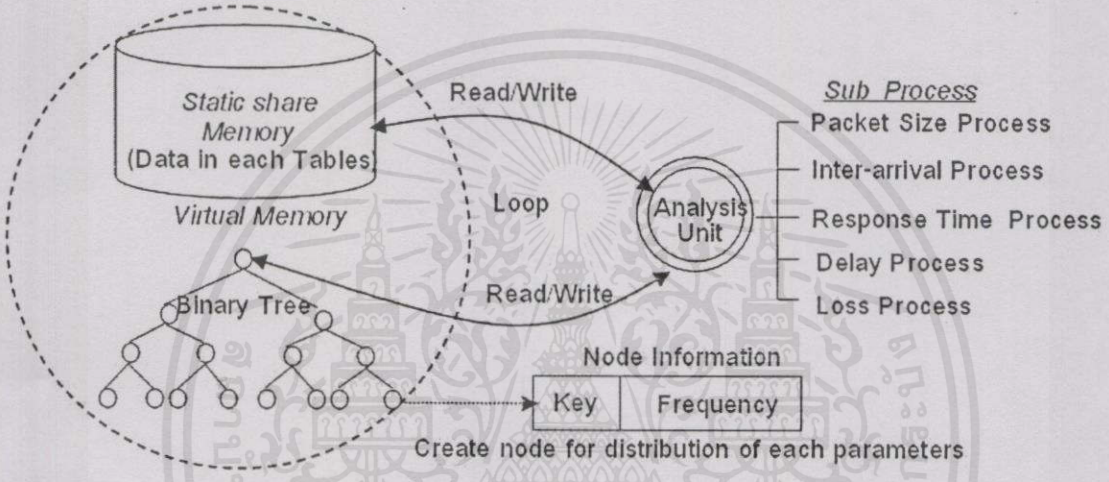
ภาพที่ 3.6 ขั้นตอนวิธีการแยกชนิดข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การวิเคราะห์ข้อมูลเครือข่าย

3.4.1 การออกแบบหน่วยวิเคราะห์ข้อมูล

หน่วยวิเคราะห์ข้อมูลได้ออกแบบให้แบ่งหน่วยประมวลผล เพื่อวิเคราะห์คุณลักษณะข้อมูลตามชนิดต่าง ๆ จากฐานข้อมูลบนหน่วยความจำที่เดียวกัน ตารางผลลัพธ์จะถูกสร้างขึ้นตามจำนวนหน่วยประมวลผลโดยใช้หลักการบริหารและจัดการหน่วยความจำแบบต้นไม้(Binary Tree) โดยทุกหน่วยประมวลผลจะทำงานไปพร้อมกันจนกว่าจะได้รับการสั่งหยุด(ผู้ใช้กดแป้น Ctrl+C) จึงจะเขียนผลลัพธ์ทั้งหมดในรูปแบบแฟ้มข้อมูลแจกแจงความถี่ ดังภาพที่ 3.7



ภาพที่ 3.7 ขบวนการวิเคราะห์ข้อมูล

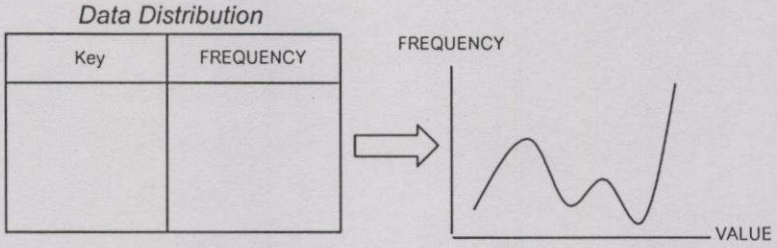
หน่วยวิเคราะห์อ่านข้อมูลจากหน่วยความจำที่เดียวกัน และสร้างตารางผลลัพธ์ที่มีคุณสมบัติตามขบวนการแบบต้นไม้ โดยใช้โครงสร้างข้อมูลของโนดตามภาพที่ 3.8

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

ภาพที่ 3.8 โครงสร้างโนด

โครงสร้างข้อมูลออกแบบให้สอดคล้องกับตารางแจกแจงความถี่ข้อมูล(Data Distribution Table) ซึ่งประกอบด้วย ตัวแปร Key ใช้แทนค่าข้อมูล และตัวแปร Frequency ใช้แทนจำนวนนับเหตุการณ์ที่เกิดค่าข้อมูลขึ้น ตารางแจกแจงความถี่สามารถนำไปใช้ประโยชน์ได้หลายรูปแบบในเชิงสถิติ เช่น นำไปสร้างแผนภูมิ (Histogram) เพื่อศึกษาคุณลักษณะของการเกิดเหตุการณ์ หรือวิเคราะห์ค่าเป็นตัวเลขเพื่อวิเคราะห์ความหมายต่าง ๆ ได้แก่ ค่าต่ำสุด(Minimum) ค่าสูงสุด

(Maximum) ค่าเฉลี่ย(Average) ค่าความแปรปรวน(Variance) และค่าเบี่ยงเบนมาตรฐาน(Standard Division) เป็นต้น ดังแสดงได้ในภาพที่ 3.9



ภาพที่ 3.9 ตัวอย่างการแจงแจงความถี่และกราฟ

3.4.2 ขั้นตอนวิธีตรวจวัดความยาวและเวลาระหว่างการมา

ขั้นตอนการวิเคราะห์ความยาว เวลาระหว่างการมาของแพ็กเก็ต เริ่มจากการอ่านข้อมูลบนตารางแยกชนิดเฉพาะแพ็กเก็ตที่มีชนิดเป็น TCP UDP และ UDP นำค่าความยาวข้อมูลจากตาราง ไปจัดเก็บบนตารางแจกแจงความถี่ตามชนิดข้อมูลที่วิเคราะห์ ในกรณีข้อมูลที่วิเคราะห์ไม่ใช่แพ็กเก็ตแรก ให้นำค่าเวลาของข้อมูล(Packet Time Stamp) ไปคำนวณผลต่างของเวลาข้อมูลระหว่างแพ็กเก็ตก่อนหน้าและปัจจุบัน จากนั้นจึงนำค่าเวลาไปจัดเก็บบนตารางแจกแจงความถี่ตามชนิดข้อมูลนั้น ดังแสดงได้ตามภาพที่ 3.10

```
For i = 1 to Number of both Packet in Forward and Reverse Table {
    if (Considered Packets ∈ Protocol(TCP,UDP,ICMP) {
        Save pkt_i.ipLen to Protocol(TCP,UDP,ICMP) of packet size Table
        if (Pkt_i is not first in Table) {
            t_current = TimeStamp of current packet
            t_previous = TimeStamp of Previous Packet
            Compute τ_n ; τ_n = (t_current - t_previous)
            Save τ_n to Protocol(TCP,UDP,ICMP) of inter-arrival Table
        }
    }
}
```

ภาพที่ 3.10 ขั้นตอนวิธีตรวจวัดความยาวและเวลาระหว่างการมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.3 ขั้นตอนวิธีตรวจวัดเวลาการสื่อสารข้อมูลไปกลับ

ทำการวิเคราะห์เฉพาะแพ็กเก็ตที่โปรโตคอลมีชนิดเป็น TCP และ ICMP(Echo)เท่านั้น โดยขอนิยามข้อมูลที่ได้รับการพิจารณาเหล่านี้เรียกว่า Considered Packet แต่ละแพ็กเก็ตจะถูก

นำไปเปรียบเทียบหาข้อมูลที่เป็นคู่สื่อสารกัน จากตารางข้อมูลที่มีทิศทางการสื่อสารตรงข้ามและอยู่ในฝั่งเครือข่ายเดียวกัน ซึ่งขอนิยามแพ็กเก็ตที่เป็นคู่สื่อสารกันเรียกว่า Corresponding Packet ดังแสดงได้ตามภาพที่ 3.11

```
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 3.11 ขั้นตอนวิธีตรวจวัดเวลาการสื่อสารข้อมูลไปกลับ

โดยที่ Res_i หมายถึง แพ็กเก็ตฝ่ายตอบ

Req_i หมายถึง แพ็กเก็ตฝ่ายเรียก

Response Table หมายถึง ตารางข้อมูลที่มีทิศทางตรงข้ามกับ Request Table หรือกรณีที่วัดเวลาไปกลับจากเครือข่ายภายนอก จะหมายถึงตารางที่มีทิศทางเป็นขากลับ(Reverse Table)

Request Table หมายถึง ตารางข้อมูลที่มีทิศทางตรงข้ามกับ Response Table หรือกรณีที่วัดเวลาไปกลับจากเครือข่ายภายนอก จะหมายถึงตารางที่มีทิศทางเป็นขาไป(Forward Table)

ขั้นตอนการวิเคราะห์ใช้วิธีเปรียบเทียบ โดยนำแพ็กเก็ตฝ่ายตอบ(Res_i)ไปค้นหาแพ็กเก็ตฝ่ายเรียก(Req_i) ในตารางข้อมูลที่มีทิศทางการสื่อสารตรงข้ามกัน เมื่อพบแพ็กเก็ตที่เป็นคู่สื่อสารกันให้คำนวณหาเวลาการสื่อสารข้อมูลไปกลับได้จากสมการที่ (3.1)

$$\gamma_n = t_y - t_j \qquad (3.1)$$

โดยที่ t_y หมายถึง เวลาของแพ็กเก็ตฝ่ายตอบ

t_j หมายถึง เวลาของแพ็กเก็ตฝ่ายเรียก

การตรวจวัดเวลาไปกลับของคู่แพ็กเก็ต ต้องแยกพิจารณาเปรียบเทียบกลุ่มการสื่อสารของข้อมูลที่ต่างชนิดกันระหว่าง ICMP และ TCP ดังนี้คือ

ไม่ว่าการใช้งานนั้นจะเป็นการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. การเปรียบเทียบคู่สื่อสารข้อมูลชนิด ICMP ข้อมูลชนิด ICMP โดยทั่วไปถูกใช้เพื่อรายงานสถานภาพของเครือข่าย ซึ่งแบ่งออกเป็น 2 ลักษณะคือ แจ้งความผิดพลาดที่เกิดขึ้นจากการสื่อสารข้อมูลบนเครือข่าย เช่น ข้อมูลไม่ถึงปลายทาง(Destination Unreachable) ต้นทางระงับการทำงาน(Source Quench) การรอเกินขอบเขตของเวลา(Time Exceeded) และ สอบถามสถานะ(Query Message) เช่น การตอบรับ(Echo Request and Reply) ความต้องการด้านข้อมูล(Information Request and Replay) และ การกำกับเวลา(Timestamp Request and Replay) สำหรับงานวิจัยนี้จะพิจารณาเฉพาะ ICMP แพ็กเก็ตที่ทำหน้าที่ในส่วนของการตอบรับข้อมูลเท่านั้น(Echo Request and Reply) ซึ่งตามทฤษฎีมีลำดับขั้นตอนในการสื่อสารข้อมูล คือ คอมพิวเตอร์ต้นทางจะส่ง ICMP แพ็กเก็ตเรียกว่า ฝ่ายเรียก(Echo Request) โดยมีโครงสร้างที่สำคัญ คือ ชนิด(Type) เป็น 0x00 และสุ่มหมายเลขลำดับ(Identifier)ไปยังคอมพิวเตอร์ปลายทาง เมื่อคอมพิวเตอร์ปลายทางได้รับข้อมูล จะตอบกลับด้วย ICMP แพ็กเก็ตที่มีชนิด เป็น 0x08 และมีลำดับเป็นหมายเลขเดียวกับตอนขามา ดังนั้นด้วยลักษณะการทำงานของ ICMP โปรโตคอลชนิดนี้ จึงกำหนดเงื่อนไขในการเปรียบเทียบเพื่อหาคู่แพ็กเก็ตสื่อสารดังแสดงได้ในภาพที่ 3.12

```
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 &&
                    Req_j.icmp_seq == Res_i.icmp_seq)
```

ภาพที่ 3.12 การเปรียบเทียบคู่แพ็กเก็ตสื่อสารข้อมูลชนิด ICMP

2. การเปรียบเทียบคู่สื่อสารข้อมูลชนิด TCP การสื่อสารข้อมูลด้วยโปรโตคอล TCP เป็นวิธีการรับส่งข้อมูลที่มีความน่าเชื่อถือในการแลกเปลี่ยนข้อมูล ขั้นตอนการสื่อสารข้อมูลจึงมีขบวนการตรวจสอบความถูกต้องระหว่างผู้รับและส่งข้อมูล โดยแบ่งเป็น 3 ขั้นตอนหลัก คือ สร้างการเชื่อมต่อเพื่อรับส่งข้อมูล(Connection) รับส่งข้อมูล(Data Transfer) และปิดการเชื่อมต่อ(Closed Connection) ซึ่งแต่ละขั้นตอนใช้วิธีการที่แตกต่างกันดังนี้

ขั้นตอนที่ 1 การเชื่อมต่อข้อมูล(Connection) หมายถึง การที่แอพพลิเคชั่น(Application)หรือ Client Host ฝั่งที่ต้องการข้อมูลสร้างการเชื่อมต่อไปยังอีกฝั่งหรือนิยามเรียกว่า Server Host โดยมี 3 ขบวนการที่สำคัญ คือ

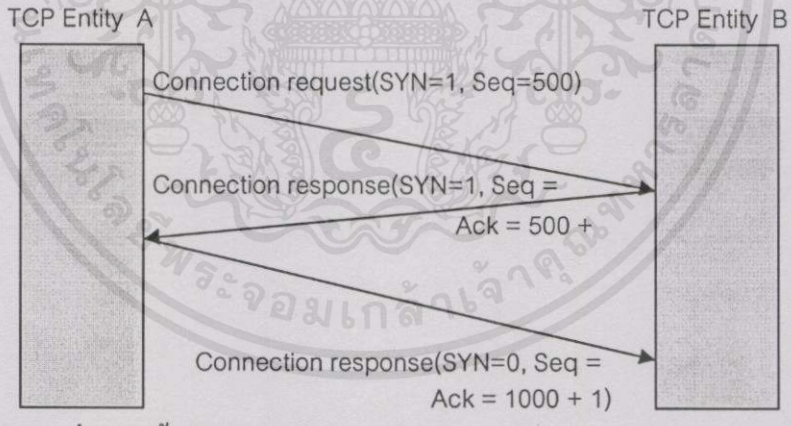
เอกสารนี้เป็นเอกสาร... ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขบวนการที่ 1 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

ขบวนการที่ 2 เมื่อ Server Host ได้รับ SYN segment ก็จะส่ง SYNACK segment กลับไปให้ Client Host โดยที่ SYNACK segment นี้ก็ยังไม่มีข้อมูลใด ๆ แต่จะมีลักษณะข้อมูลสำคัญ 3 อย่าง คือ SYN bit กำหนดค่าเป็น 1 ค่า TCP acknowledgment number field ใน TCP Header จะกำหนดโดยหมายเลขลำดับเดิมจากฝั่ง Client Host เพิ่มขึ้นอีก 1 และตั้งค่าหมายเลขลำดับในฝั่ง Server Host ขึ้นใหม่และกำกับลงใน sequence number field

ขบวนการที่ 3 หลังจาก Client Host ได้รับ SYNACK segment แล้วก็จะส่ง segment สุดท้ายไปให้ Server Host อีก โดยกำหนดค่าต่าง ๆ ได้แก่ กำหนด SYN bit เป็น 0 กำหนดค่า sequence number field ด้วยหมายเลขลำดับเดิมที่เพิ่มขึ้นอีก 1 โดยฝั่ง Server Host และกำหนด TCP acknowledgement number field เป็นหมายเลขลำดับของฝั่ง Server Host เพิ่มอีก 1

ขบวนการนี้เรียกว่า Three-Way Handshake เมื่อสิ้นสุดแล้วจะหมายถึง การสร้าง TCP Connection เป็นผลสำเร็จ ดังแสดงได้ในภาพที่ 3.13

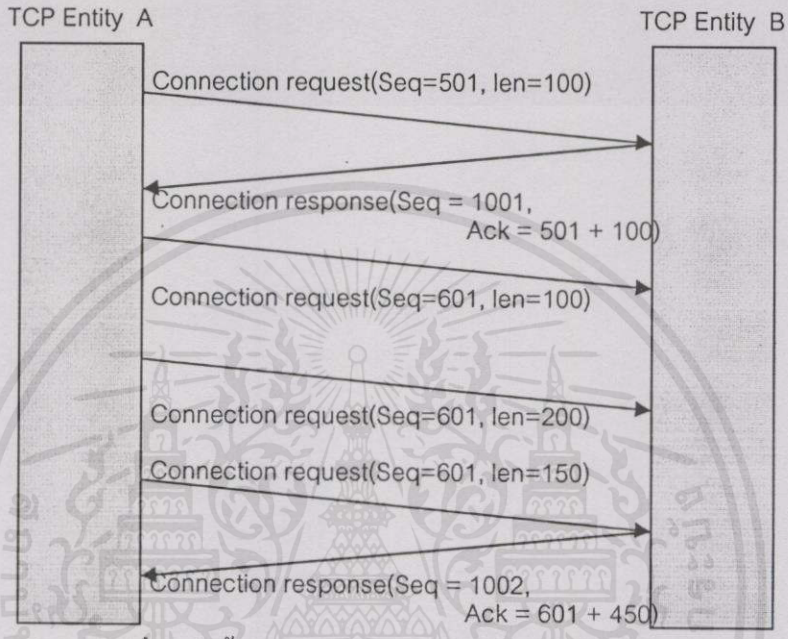


ภาพที่ 3.13 ขั้นตอนการเชื่อมต่อเพื่อรับส่งข้อมูลชนิด TCP

ขั้นตอนที่ 2 การรับส่งข้อมูล(Data Transfer) หลังจากที่ขบวนเชื่อมต่อเป็นผลสำเร็จ โปรโตคอล TCP จะเริ่มทำการรับส่งข้อมูลโดยมีขบวนการที่สำคัญคือ ฝั่ง Client Host ทำการส่งข้อมูลโดยกำหนด SYN bit เป็น 0 และกำกับ TCP sequence number ต่อจากขบวนการเชื่อมต่อไปยัง Server Host ขบวนการนี้อาจเกิดขึ้นได้มากกว่า 1 segment หรืออาจทำจนครบจำนวนข้อมูลที่ต้องการส่ง ขึ้นอยู่กับการกำหนด window size และเมื่อฝั่ง Server Host ได้รับข้อมูลก็จะตอบกลับโดยกำหนด SYN bit เป็น 0 และกำหนด TCP sequence number เดิมต่อจากขบวนการเชื่อมต่อ ส่วน TCP acknowledgment number จะถูกกำหนดจาก TCP sequence

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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 3.14

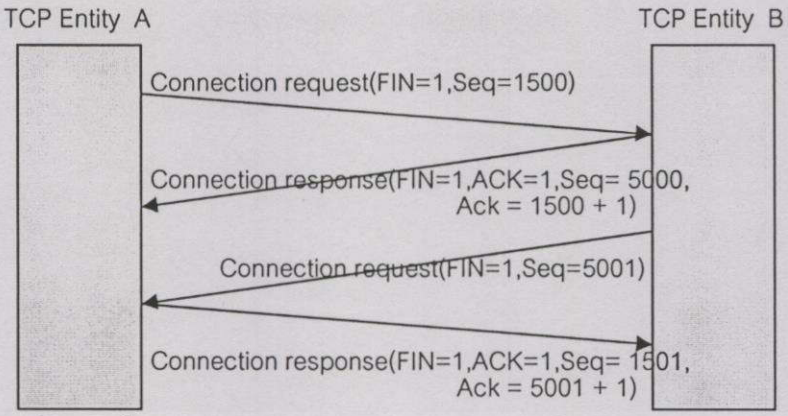


ภาพที่ 3.14 ขั้นตอนการรับส่งข้อมูลชนิด TCP

ดังนั้นด้วยลักษณะการทำงานแบบนี้ การเปรียบเทียบหาคู่แพ็กเก็ตสื่อสารข้อมูลอาจกระทำได้แต่เทคนิคของเวลาที่ใช้ไปกับการรอแพ็กเก็ตที่เป็นคู่สื่อสารข้อมูลกันนั้น จำเป็นต้องเสียทรัพยากรไปกับการจัดเก็บข้อมูลเป็นปริมาณมาก ในงานวิจัยนี้จึงขอละเว้นจากการตรวจสอบคู่แพ็กเก็ตนี้ไปก่อน

ขั้นตอนที่ 3 ปิดการเชื่อมต่อ(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 3.15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.15 ขั้นตอนปิดการเชื่อมต่อในการรับส่งข้อมูลชนิด TCP

จากขบวนการสื่อสารข้อมูลด้วย TCP โปรโตคอล จึงกำหนดให้มีการวิเคราะห์เวลาการสื่อสารไปกลับเฉพาะในช่วงเวลาของการเปิดและปิดการเชื่อมต่อเท่านั้น โดยมีเงื่อนไขของการเปรียบเทียบเพื่อหาคู่แพ็กเก็ตสื่อสารได้ดังภาพที่ 3.16

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 3.16 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 3.17

```
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 3.17 การเปรียบเทียบคู่แพ็กเก็ตชนิด TCP ระหว่างปิดการเชื่อมต่อ

กลุ่มข้อมูลที่พิจารณาคือแพ็กเก็ตของฝ่ายตอบ(Res_i) ที่มี Protocol filed ใน TCP Header เท่ากับ 0x06 และส่งสัญญาณข้อมูลชนิด ACK(TCP flags field มีค่าเป็น 0x10)ให้กับฝ่ายเรียก(Req_j)ที่ส่งด้วยสัญญาณชนิด FIN/ACK(TCP flags field มีค่าเป็น 0x11) ในการเปรียบเทียบคู่ของแพ็กเก็ตจะพิจารณาจาก Protocol, IP Source, IP Destination, Destination Port, Source Port field ที่เท่ากัน และมี Sequence Number field ของแพ็กเก็ตในฝ่ายเรียกมีค่าเท่ากับ Sequence Number field ลบด้วย 1 ของแพ็กเก็ตฝ่ายตอบเช่นเดียวกับกรณีเปิดการเชื่อมต่อ

3.4.4 ขั้นตอนวิธีการตรวจวัดเวลาการสื่อสารและความสูญเสียข้อมูลผ่านตัวระบบ

การวิเคราะห์เวลาการสื่อสารข้อมูลผ่านตัวระบบ ในงานวิจัยกำหนดให้วิเคราะห์เฉพาะข้อมูลชนิด TCP, UDP และ ICMP โดยขอนิยามข้อมูลที่ได้รับการพิจารณาเหล่านี้เรียกว่า Considered Packet แต่ละแพ็กเก็ตจะถูกนำไปเปรียบเทียบหาข้อมูลหน่วยเดียวกัน ที่ผ่านเข้าและออกจากตัวระบบ จากตารางที่มีทิศทางการสื่อสารข้อมูลทางเดียวกันแต่อยู่ฝั่งเครือข่ายตรงข้ามกัน ซึ่งนิยามเรียกแพ็กเก็ตที่เป็นหน่วยข้อมูลเดียวกันนี้เรียกว่า Ownership Packet ตัวอย่างเช่น นำแพ็กเก็ตใน Internal Forward Table เปรียบเทียบแพ็กเก็ตใน External Forward Table หรือ นำแพ็กเก็ตใน Internal Reverse Table เปรียบเทียบกับแพ็กเก็ตใน External Reverse Table โดยจะนำแพ็กเก็ตที่ตรวจพบว่าเป็นแพ็กเก็ตหน่วยเดียวกัน จากฝั่งขาเข้าและออกไปคำนวณหาเวลาการสื่อสารข้อมูลผ่านตัวระบบได้จากสมการที่ (3.2)

$$\tau_m = t_x - t_y \qquad (3.2)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ t_x หมายถึง เวลาของแพ็กเก็ตฝั่งขาเข้า

t_y หมายถึง เวลาของแพ็กเก็ตฝั่งขาออก

ดังแสดงได้ในภาพที่ 3.18

```
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;
```

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 3.18 ขั้นตอนวิธีการวิเคราะห์เวลาการสื่อสารและความสูญเสียข้อมูลผ่านตัวระบบ

แพ็กเก็ตที่ตรวจไม่พบในฝั่งขาออกตามพื้นที่ที่กำหนด(Record Range for Searching) จะถือเป็นความสูญเสีย ซึ่งถูกจะนำไปจัดเก็บไว้ในตารางความสูญเสียข้อมูล(Loss Table)ที่จัดเตรียมพื้นที่ไว้ทั้งทิศทางขาไปและขากลับ โดยใช้หลักการเปรียบเทียบด้วยเงื่อนไขที่แสดงได้ในภาพที่ 3.19

```
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_.proto_header == Ent_.proto_header)
```

ภาพที่ 3.19 การเปรียบเทียบหาคู่แพ็กเก็ตในขั้นตอนวิเคราะห์เวลาการสื่อสารและความสูญเสียข้อมูลผ่านตัวระบบ

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

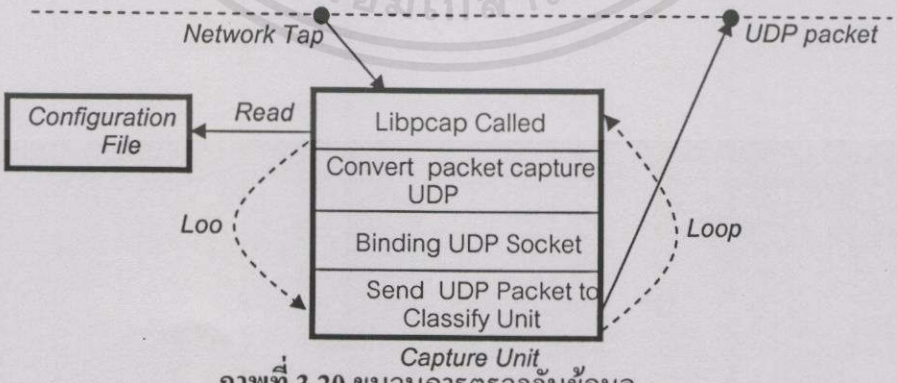
จากภาพที่ 3.19 พิจารณาเฉพาะแพ็กเก็ตที่มี Protocol เป็น ICMP(protocol field มีค่าเป็น 0x01), TCP(protocol field มีค่าเป็น 0x06) หรือ UDP(protocol มีค่าเป็น 0x11) ไปเปรียบเทียบหาคู่แพ็กเก็ตเดียวกัน ด้วยเงื่อนไขที่มี Protocol, Identifier, IP Offset, IP Destination, IP Source, IP Total Length และ Protocol Header ในแต่ละ field ของข้อมูลแต่ละชนิดที่ตรงกัน(ICMP Header, TCP Header และ UDP Header)

3.5 การโปรแกรมวิเคราะห์ข้อมูลเครือข่าย

โปรแกรมการวิเคราะห์ข้อมูลได้ออกแบบและพัฒนาด้วยโปรแกรมภาษา C++ [12]ให้มีลักษณะการทำงานเป็นแบบแบ่งเวลาประมวลผล(Multiple Thread) ดังนั้นหน่วยตรวจจับ หน่วยแยกชนิด และหน่วยวิเคราะห์ข้อมูลจึงจำเป็นต้องมีข้อตกลงในการทำงานร่วมกัน นอกจากนี้ด้วยลักษณะการตรวจจับข้อมูลที่มีความต่อเนื่อง และไม่มีข้อจำกัดด้านเวลาในการวิเคราะห์ข้อมูล เครื่องมือวิจัยจึงต้องออกแบบให้มีการบริหารจัดการหน่วยความจำอย่างมีประสิทธิภาพ ทั้งในด้านการจัดเก็บข้อมูลและการนำข้อมูลไปใช้วิเคราะห์ผลลัพธ์ ที่จะถูกเขียนขึ้นหลังจากที่โปรแกรมได้รับการสั่งหยุดจากผู้ใช้งาน ดังนั้นในการโปรแกรมคอมพิวเตอร์จึงต้องใช้เทคนิคต่าง ๆ ในการทำงานร่วมกัน โดยมีลักษณะดังนี้

3.5.1 การออกแบบข้อกำหนดในการตรวจจับข้อมูล

หน่วยตรวจจับสามารถส่งข้อมูลไปยังหน่วยแยกชนิดได้ด้วยการสร้างช่องทางการสื่อสารร่วมกัน โดยใช้พอร์ตสื่อสารชนิด UDP(Binding UDP Socket) ดังนั้นทั้ง 2 หน่วยจึงจำเป็นต้องมีการออกแบบข้อตกลงในการแลกเปลี่ยนข้อมูลหรือนิยามเรียกว่า Configuration File เพื่อใช้กำหนดโครงสร้างข้อมูลใหม่จากการตรวจจับได้โดย Libpcap Library ดังแสดงได้ในภาพที่ 3.20



ภาพที่ 3.20 ขบวนการตรวจจับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Configuration File จึงเป็นส่วนที่สำคัญที่ทุกหน่วยตรวจจับข้อมูลใช้เป็นข้อตกลงร่วมกันในการส่งข้อมูลไปยังหน่วยแยกชนิด โดยออกแบบให้มีข้อกำหนดที่สำคัญดังนี้

1. DEVICE หมายถึง ชื่อที่ FreeBSD ใช้เชื่อมต่อกับอุปกรณ์ Ethernet Card เช่น Fxp0, wb0 หรืออื่น ๆ

2. SENDTOHOST หมายถึง หมายเลขไอพีของเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นหน่วยแยกชนิดข้อมูล

3. PORT หมายถึง หมายเลขพอร์ตชนิด UDP ที่ใช้ในการรับส่งข้อมูลระหว่างคอมพิวเตอร์ที่ทำหน้าที่ตรวจจับและแยกชนิดข้อมูล

4. FORMATTYPE,CAPTUREID,HOSTID,CLASSFYID,PACKETINF หมายถึง ค่าที่ใช้กำหนดคุณสมบัติไว้ตาม Capture Header

5. STOREDISK หมายถึง สถานะที่กำหนดให้หน่วยตรวจจับบันทึกข้อมูลไว้บน Hard Disk

6. FILENAME หมายถึง ชื่อแฟ้มที่ต้องการจัดเก็บฐานข้อมูล

ดังแสดงรูปแบบการติดตั้งได้ในภาพที่ 3.21

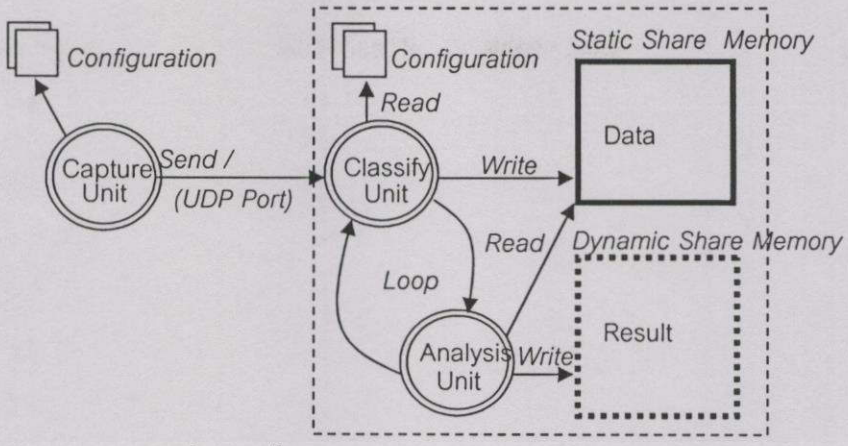
```
# ethernet device such as fpx0, wb0, etc...
DEVICE        wb0
# ip host(Classifying and Analysis Data) for sent packet
SENDTOHOST    127.0.0.1
# udp port communication, default is 3056
PORT          3056
# Format data type 0 - binary (Reserve other number for future) , default is 0
FORMATTYPE    0
# Identifier of internal or external network, input value between 0 - 9 , default is 0
CAPTUREID     0
# Identifier data of each 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 packet's to disk ,  enable-save , disable-none
STOREDISK     disable
# file name, default is 'disk0.dat'
FILENAME      disk0.dat
```

ภาพที่ 3.21 แฟ้มติดตั้งการใช้งานหน่วยตรวจจับข้อมูล

3.5.2 ออกแบบข้อกำหนดในการแยกชนิดข้อมูล

หน่วยแยกชนิดเป็นหน่วยที่มีหน้าที่สำคัญคือรับแพ็กเก็ตจากหน่วยตรวจจับข้อมูลและทำการจัดเก็บแพ็กเก็ตลงตารางข้อมูลแต่ละชนิดให้ถูกต้อง โดยตารางข้อมูลจะถูกสร้างขึ้นในรูปของหน่วยความจำร่วมแบบคงที่(Static Share Memory) ซึ่งหน่วยวิเคราะห์ข้อมูลสามารถเปิดอ่านและนำไปใช้สร้างตารางแจกแจงความถี่(Distribution Table)ต่อไปได้ ดังนั้นการทำงานของหน่วยตรวจจับข้อมูล หน่วยแยกชนิด และหน่วยวิเคราะห์ข้อมูลจะต้องปฏิบัติงานไปพร้อม ๆ กันโดยอาศัยหลักการเขียนโปรแกรมแบบ Multiple Thread ให้ทั้งสามหน่วยทำงานเป็นวัฏจักร(Loop)ที่ไม่รู้จบจนกว่าจะได้รับการสั่งหยุดจากผู้ใช้ ดังแสดงได้ในภาพที่ 3.22

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.22 ขั้นตอนการทำงานของหน่วยแยกชนิด

ดังนั้นเพื่อให้การทำงานของหน่วยแยกชนิดสามารถกำหนดรูปแบบข้อมูลให้กับหน่วยวิเคราะห์ข้อมูลได้อย่างถูกต้อง หน่วยแยกชนิดจึงต้องออกแบบให้มีแฟ้มติดตั้งข้อกำหนดก่อนการใช้งาน โดยมีพารามิเตอร์ที่สำคัญดังนี้

1. PORTRCVE หมายถึง หมายเลขพอร์ตการสื่อสารข้อมูลที่ใช้รับส่งข้อมูลระหว่างหน่วยตรวจจับและแยกชนิดข้อมูล

2. MEDIAREAD หมายถึง อุปกรณ์ที่ต้องการให้หน่วยแยกชนิดรับส่งแพ็กเก็ตมาตรวจสอบ ซึ่งกำหนดให้มีได้ 2 อุปกรณ์ คือ ผ่านทางเครือข่าย หรือ อ่านจากแฟ้มข้อมูล

3. INDISK หมายถึง ชื่อแฟ้มจัดเก็บข้อมูลจากฝั่งเครือข่ายภายในที่ถูกบันทึกได้จากหน่วยตรวจจับข้อมูล จะถูกใช้ก็ต่อเมื่อไม่มีการกำหนดให้ MEDIAREAD เป็น Network

4. EXDISK หมายถึง ชื่อแฟ้มจัดเก็บข้อมูลจากฝั่งเครือข่ายภายนอกที่ถูกบันทึกได้จากหน่วยตรวจจับข้อมูล จะถูกใช้ก็ต่อเมื่อไม่มีการกำหนดให้ MEDIAREAD เป็น Network

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

6. SUTIP หมายถึง ไอพีที่ใช้งานในกลุ่มของตัวระบบ

7. INTERNALCAPID หมายถึง หมายเลขของ Capture Identifier ที่อยู่ฝั่งเครือข่ายภายใน

8. EXTERNALCAPID หมายถึง หมายเลขของ Capture Identifier ที่อยู่ฝั่งเครือข่ายภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. PACKETSIZE หมายถึง สถานะการวิเคราะห์ความยาวแพ็กเก็ต

10. INTERARRIVAL หมายถึง สถานะการวิเคราะห์เวลาระหว่างการมา

11. DELAY หมายถึง สถานะการวิเคราะห์เวลาการสื่อสารข้อมูลผ่านตัวระบบ

12. ROUNDTRIPTIME หมายถึง สถานะการวิเคราะห์เวลาการสื่อสารข้อมูลไปและกลับ

13. LOSS หมายถึง สถานะการวิเคราะห์ความสูญเสียข้อมูลบนตัวระบบ

14. FLOWED หมายถึง สถานะการวิเคราะห์ข้อมูลโฟลว

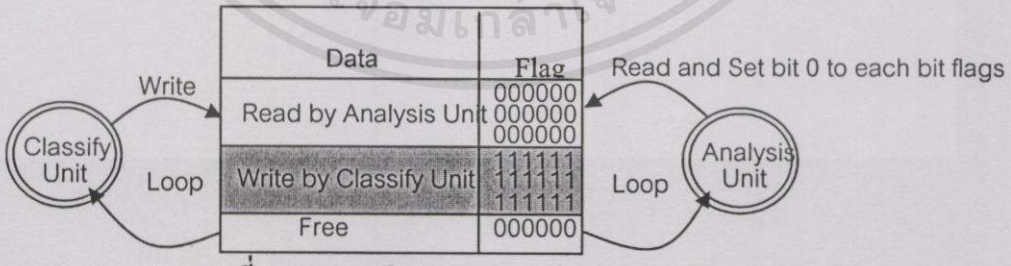
15. FLWEXP หมายถึง เวลาที่ใช้ตรวจสอบการหมดอายุของข้อมูลโฟลวบนเครือข่าย ซึ่งค่าปกติกำหนดไว้ที่ 64 วินาที ดังแสดงได้ในภาพที่ 3.23

```
#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 3.23 แฟ้มติดตั้งข้อกำหนดการแยกชนิดข้อมูล

3.5.3 การบริหารและจัดการหน่วยความจำร่วม

เพื่อให้การวิเคราะห์ข้อมูลใช้หน่วยความจำอย่างประหยัดและมีประสิทธิภาพ หน่วยแยกชนิดจึงออกแบบให้ข้อมูลที่ถูกบันทึกบนหน่วยความจำจนครบทั้งตาราง สามารถวนกลับมาใช้ตำแหน่งของหน่วยความจำเริ่มต้นเดิมได้อีกครั้ง ถ้าระเบียนดังกล่าวได้ถูกนำไปใช้วิเคราะห์ข้อมูลจนครบตามความต้องการของหน่วยวิเคราะห์แล้ว ดังแสดงได้ในภาพที่ 3.24



ภาพที่ 3.24 การบริหารจัดการหน่วยความจำร่วม

การกลับไปใช้หน่วยความจำเริ่มต้นใหม่จะต้องมีความสัมพันธ์กับหน่วยวิเคราะห์ข้อมูล เนื่องจากช่วงเวลาการมาของข้อมูลอาจเร็วกว่ากระบวนการสร้างผลลัพธ์โดยหน่วยวิเคราะห์ ดังนั้นหน่วยแยกชนิดจึงต้องตรวจสอบก่อนการบันทึกข้อมูลอยู่เสมอว่า ข้อมูลที่จะเขียนทับนั้นถูกอ่านหรือนำไปใช้สร้างผลลัพธ์โดยหน่วยวิเคราะห์มูลแล้วหรือไม่ เพื่อแก้ปัญหานี้จึงต้องเพิ่ม Flags ที่มีขนาดข้อมูลเท่ากับ 8 bits(ตามที่แสดงไว้ในตารางที่ 3.1) มาใช้บอกสถานะของการนำไปใช้ใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

และถ้าทุก bits ของ Flags มีค่าเป็น 0 หมายถึง ข้อมูลที่กำลังวิเคราะห์ได้ถูกนำไปใช้วิเคราะห์แล้วในทุกพารามิเตอร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

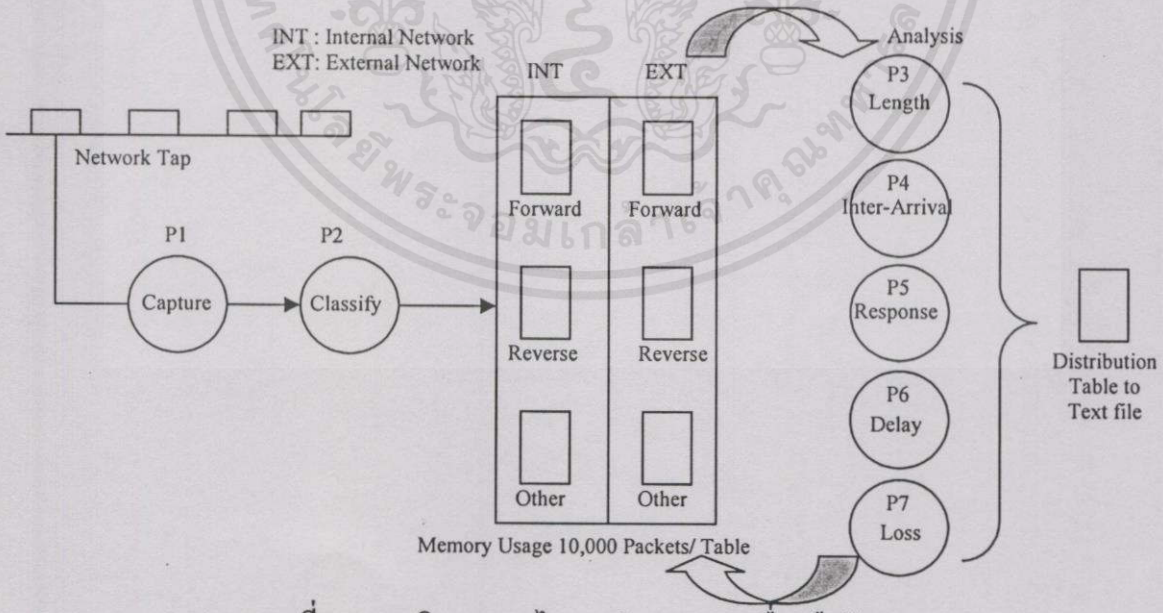
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและวิเคราะห์ประสิทธิภาพ

ในบทนี้จะกล่าวถึงการพัฒนาเครื่องมือวัดตามวิธีการที่ออกแบบ ทำการทดลองกับระบบจำลองที่ใช้อุปกรณ์การสื่อสารข้อมูลจริงบนเครือข่าย ผลที่ได้จะแสดงให้เห็นถึงประสิทธิภาพในการตรวจวัดข้อมูลของเครื่องมือ ทั้งในด้านความถูกต้อง ความเร็วในการวิเคราะห์ข้อมูล และปัจจัยที่ส่งผลกระทบต่อการออกแบบเครื่องมือวัด

4.1 เทคนิคและกลไกการทำงานของเครื่องมือวัด

เทคนิคการตรวจวัดได้ออกแบบให้ทำงานเป็นแบบแบ่งหน่วยเวลาประมวลผล 3 หน่วย หลักคือ หน่วยตรวจจับ(Capture) หน่วยแยกชนิด(Classify) และ หน่วยวิเคราะห์ข้อมูล(Analysis) โดยที่หน่วยวิเคราะห์ข้อมูลจะประกอบด้วยหน่วยวิเคราะห์ย่อยอีก 5 หน่วย คือ หน่วยวิเคราะห์ความยาว เวลาระหว่างการมา เวลาการสื่อสารข้อมูลไปกลับ เวลาการสื่อสารข้อมูลผ่านตัวระบบ และ ความสูญเสียข้อมูล ซึ่งหน่วยวิเคราะห์ต่าง ๆ จะอ่านข้อมูลจากตารางข้อมูลที่ถูกแยกชนิดตามทิศทางการสื่อสารข้อมูลไว้บนหน่วยความจำ เพื่อนำไปสร้างเป็นผลลัพธ์ในรูปของตารางแจกแจงพฤติกรรมการสื่อสารข้อมูล ดังภาพที่ 4.1



ภาพที่ 4.1 เทคนิคและกลไกการทำงานของเครื่องมือวัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

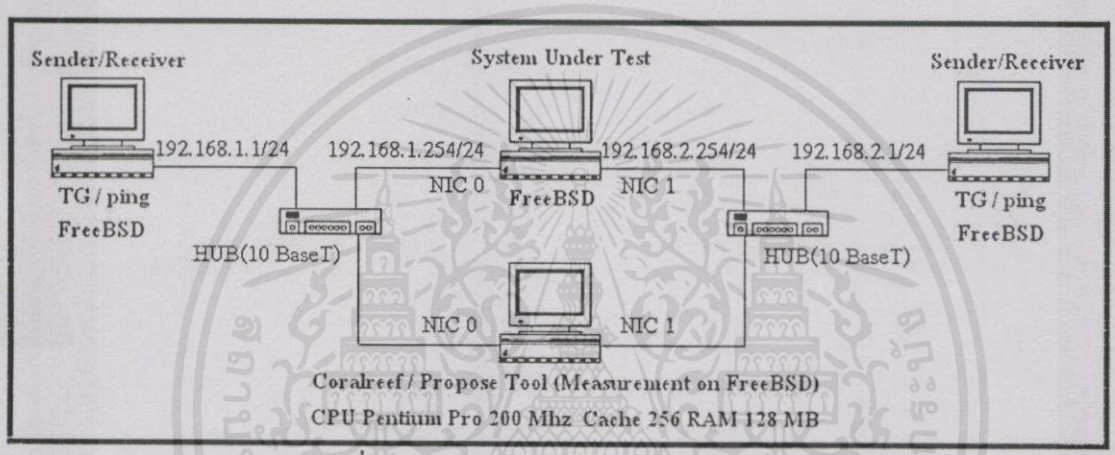
ดังนั้นเพื่อให้ผลการตรวจวัดด้วยเครื่องมือวิจัยที่ออกแบบมีประสิทธิภาพ และสามารถนำไปใช้ตรวจวัดข้อมูลสื่อสารจริงได้ ในงานวิจัยนี้จึงได้ทำการศึกษาปัจจัยต่าง ๆ ที่เกี่ยวข้องกับเทคนิคและวิธีการตรวจวัดเมื่อนำไปใช้ตรวจวัดข้อมูลจริง ดังนี้คือ

1. ศึกษาความถูกต้องในการตรวจวัด โดยพิจารณาผลการตรวจวัดเปรียบเทียบกับเครื่องมืออื่น ๆ

2. ศึกษาประสิทธิภาพของปัจจัยที่มีผลกระทบต่อการตรวจวัด โดยพิจารณาจาก ค่ารอบเวลา การประมวลผลแบบ Single หรือ Multiple Thread และความเร็วของหน่วยประมวลผลกลาง

4.2 แบบจำลองการทดลอง

ออกแบบโดยใช้อุปกรณ์จริงบนเครือข่าย เพื่อสร้างภาระงานการสื่อสารข้อมูลจริง และทำการตรวจวัดด้วยเครื่องมือวิจัย โดยติดตั้งเครื่องมือที่มีองค์ประกอบ ดังภาพที่ 4.2



ภาพที่ 4.2 แบบจำลองการทดลองตรวจวัดความถูกต้อง

แบบจำลองประกอบด้วยคอมพิวเตอร์จำนวน 1 ชุด ติดตั้งเครื่องมือวัดที่มีคุณสมบัติของหน่วยประมวลผลกลางเป็น Pentium Pro ความเร็ว 200 MHz หน่วยความจำขนาด 128 MB ติดตั้งระบบปฏิบัติการ FreeBSD version 4.5[13] และใช้ Hz Option ของ Kernel เท่ากับ 5000

คอมพิวเตอร์ที่ทำหน้าที่เป็นฝ่ายรับ(Sender)และส่ง(Receiver)ข้อมูล ใช้แผงวงจรเครือข่าย(Network Interface Card)ความเร็ว 100 Mbit/Sec จำนวน 1 ชุด ส่วนคอมพิวเตอร์ที่ทำหน้าที่เป็นตัวระบบ(System Under Test)และเครื่องมือวิจัย(Propose Tool)ใช้แผงวงจรเครือข่ายจำนวน 2 ชุด โดยคอมพิวเตอร์แต่ละชุดเชื่อมต่อกับอุปกรณ์ HUB ความเร็ว 10 Mbit/Sec เพื่อจำลองการสื่อสารระหว่าง 2 เครือข่ายคือ 192.168.1.0/24 เป็นฝั่งเครือข่ายภายใน(Internal Network) และ 192.168.2.0/24 เป็นฝั่งเครือข่ายภายนอก(External Network)

4.3 การวิเคราะห์ความถูกต้องเมื่อเทียบกับเครื่องมือวัดอื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องมือวัดได้ออกแบบขั้นตอนวิธี ให้ตรวจวัดพฤติกรรมการสื่อสารข้อมูล ซึ่งปัจจุบันคุณลักษณะข้อมูลบางชนิดสามารถตรวจวัดพฤติกรรมได้จากเครื่องมือที่ใกล้เคียงกัน เช่น Coral

Library Tool เป็นเครื่องมือที่ใช้ตรวจจับข้อมูลบนเครือข่าย หรือคำสั่ง Ping ใช้ตรวจวัดเวลาไปกลับโดยใช้โปรโตคอลชนิด ICMP ดังแสดงคุณลักษณะการตรวจวัดข้อมูลของเครื่องมือชนิดต่าง ๆ ได้จากตารางที่ 4.1

ตารางที่ 4.1 เปรียบเทียบความสามารถของเครื่องมือตรวจวัด

เครื่องมือวัด	ประเภทการวัด	ความสามารถในการวัด	
		โปรโตคอล	ค่าที่วัด
Propose Tool (เครื่องมือวิจัย)	Passive	TCP,UDP และ ICMP	Packets Count, Length, Inter-Arrival Rate, Response Time, Delay, Loss of Distribution
Coral Library Tool	Passive	TCP,UDP และ ICMP	Package Count, Packets Capture
Ping Command	Active	ICMP	Package Count, Response Time

ดังนั้นในการวิเคราะห์ความถูกต้องของเครื่องมือวิจัย สามารถพิจารณาได้จากการเปรียบเทียบผลลัพธ์การวิเคราะห์ข้อมูลที่ใกล้เคียงกันด้วยเครื่องมือเหล่านี้ได้ โดยตั้งสมมติฐานการวิจัยไว้ว่าผลการออกแบบจะสามารถนำไปใช้สร้างเป็นเครื่องมือตรวจวัดการสื่อสารข้อมูลได้ถูกต้องไม่ต่างจากเครื่องมือวัดอื่น

4.3.1 การทดลองตรวจวัดความยาวและเวลาระหว่างการมาของข้อมูล

ทดลองตรวจวัดความยาวและเวลาระหว่างการมาของข้อมูล โดยเปรียบเทียบระหว่างผลการวัดด้วยเครื่องมือวิจัยและ Coral Library Tool จากการสุ่มข้อมูลชนิด TCP และ UDP ด้วย TG(Traffic Generator) ที่มีความยาวแตกต่างกันจำนวน 500 แพ็กเก็ตจากฝั่งเครือข่ายภายในไปยังฝั่งเครือข่ายภายนอก เปรียบเทียบผลการตรวจวัดด้วยค่าสถิติต่าง ๆ ได้แก่ ค่าเฉลี่ย ค่าสัมประสิทธิ์สหสัมพันธ์ และค่าเปอร์เซ็นต์ความผิดพลาด(Percent Error)จากสมการที่ (4.1)

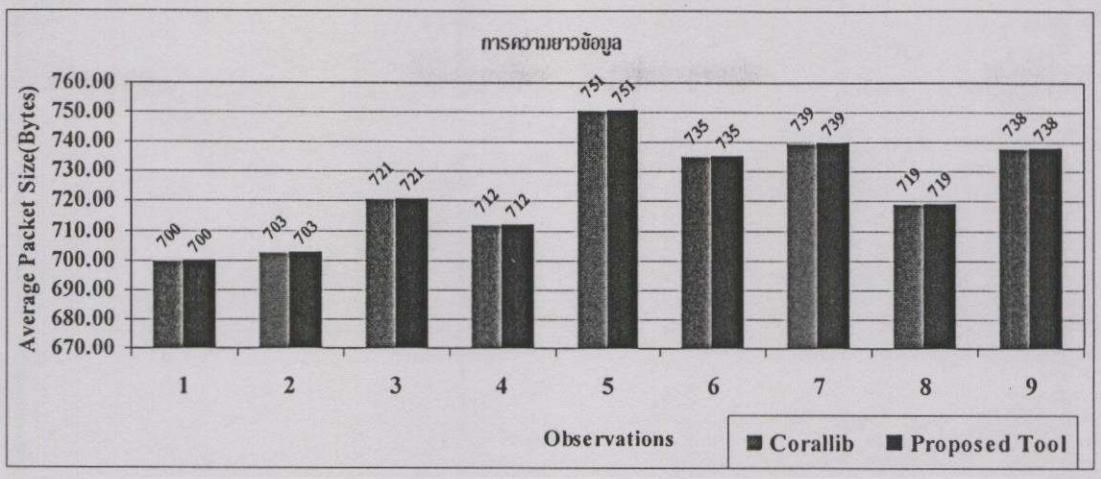
$$percent\ error = \left| \frac{measure\ value - true\ value}{true\ value} \right| * 100 \qquad (4.1)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

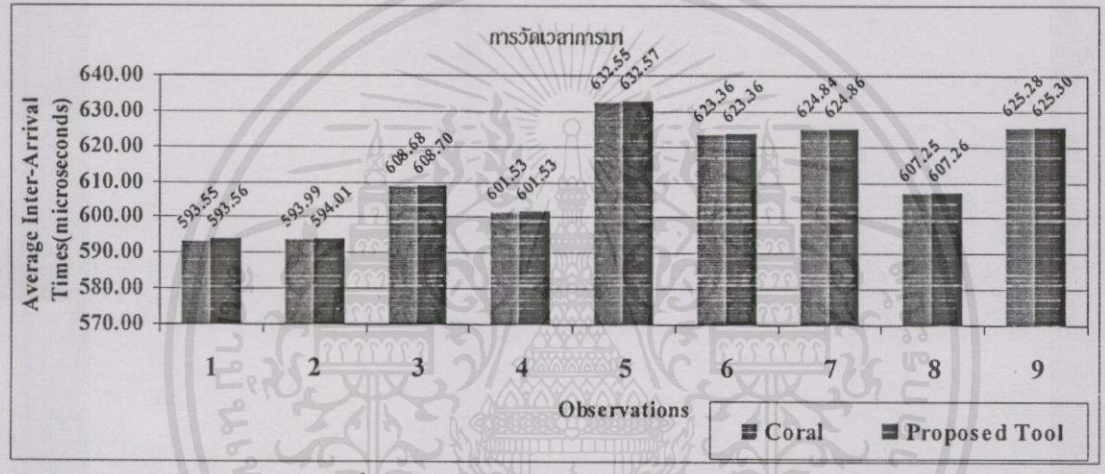
โดยที่ *percent error* หมายถึง เปอร์เซ็นต์ความผิดพลาด

measure value หมายถึง ค่าที่วัดได้จากเครื่องมือ

measure value หมายถึง ค่าที่วัดได้จากเครื่องมือที่จะนำมาใช้เปรียบเทียบ



ภาพที่ 4.3 ผลการวิเคราะห์ความยาวข้อมูล



ภาพที่ 4.4 ผลการวิเคราะห์เวลาระหว่างการมา

ในภาพที่ 4.3 แสดงลักษณะการตรวจวัดความยาวข้อมูลสรุปได้ว่า ผลการเปรียบเทียบไม่มีความแตกต่างกัน ทั้ง 2 เครื่องมือตรวจวัดจำนวนแพ็กเก็ตได้ครบตามจำนวน คือ 500 แพ็กเก็ต ค่าเฉลี่ยความยาววัดได้เท่ากัน ค่าเปอร์เซ็นต์ความผิดพลาดเมื่อเทียบกับ Coral Library Tool วัดได้ 0.00% ค่าสัมประสิทธิ์สหสัมพันธ์วัดได้ 1.00 ส่วนในรูปที่ 4.4 แสดงลักษณะการตรวจวัดเวลาระหว่างการมา ผลการเปรียบเทียบทั้ง 2 เครื่องมือตรวจวัดค่าเวลาระหว่างการมาได้ใกล้เคียงกัน โดยเครื่องมือวิจัยและ Coral Library Tool ตรวจวัดค่าเฉลี่ยได้ 612.34 และ 612.35 μs ค่าเปอร์เซ็นต์ความผิดพลาดวัดได้ 0.0021% ค่าสัมประสิทธิ์สหสัมพันธ์วัดได้ 0.99 สรุปได้ว่าผลการวัดจากทั้ง 2 เครื่องมือตรวจวัดเวลาระหว่างมาได้ใกล้เคียงกันสูงมาก

4.3.2 การทดลองตรวจวัดเวลาการสื่อสารข้อมูลไปกลับ

ทดลองตรวจวัดเวลาการสื่อสารข้อมูลไปกลับ โดยแยกพิจารณาตามชนิดข้อมูลสื่อสารต่อไปนี้

1. ทดลองตรวจวัดข้อมูลชนิด ICMP โดยส่งข้อมูลชนิด ICMP ด้วยคำสั่ง Ping จำนวน 1,000 แพ็กเก็ต จากฝั่งเครือข่ายภายในไปยังฝั่งเครือข่ายภายนอก ทำซ้ำการทดลองโดยปรับค่าเวลาคอยก่อนการส่งข้อมูลออกจากตัวจำลองระบบระหว่าง 0 ถึง 1000 มิลลิวินาที หรือนิยามเรียกค่าเวลานี้ว่า Emulated Delay โดยใช้คำสั่ง ipfw บน FeeBSD ด้วยรูปแบบคือ

ipfw add pipe 1 ip from any to any in

ipfw pipe 1 config delay [n] ms

โดยที่ [n] หมายถึง ค่าเวลาคอยก่อนการส่งข้อมูลออกจากตัวจำลองระบบ มีหน่วยเวลาเป็นมิลลิวินาที

เปรียบเทียบผลการวัดค่าที่ได้จากคำสั่ง Ping โดยใช้ค่าสถิติต่าง ๆ ได้แก่ ค่าเฉลี่ย เปอร์เซ็นต์ความผิดพลาด และค่าสัมประสิทธิ์สหสัมพันธ์ของผลต่างจาก Emulated Delay โดยสมการที่ (4.2)

$$diff_t = |tm_t - td_t| \tag{4.2}$$

โดยที่ $diff_t$ หมายถึง ผลต่างจากเวลาหน่วง ที่ตรวจวัดได้ในลำดับที่ t

tm_t หมายถึง ค่าเวลาไปกลับที่ตรวจวัดได้จากเครื่องมือ

td_t หมายถึง ค่า Emulated Delay ที่กำหนดให้กับตัวจำลองระบบ มีค่าอยู่ระหว่าง 0 ถึง 1000 มิลลิวินาที

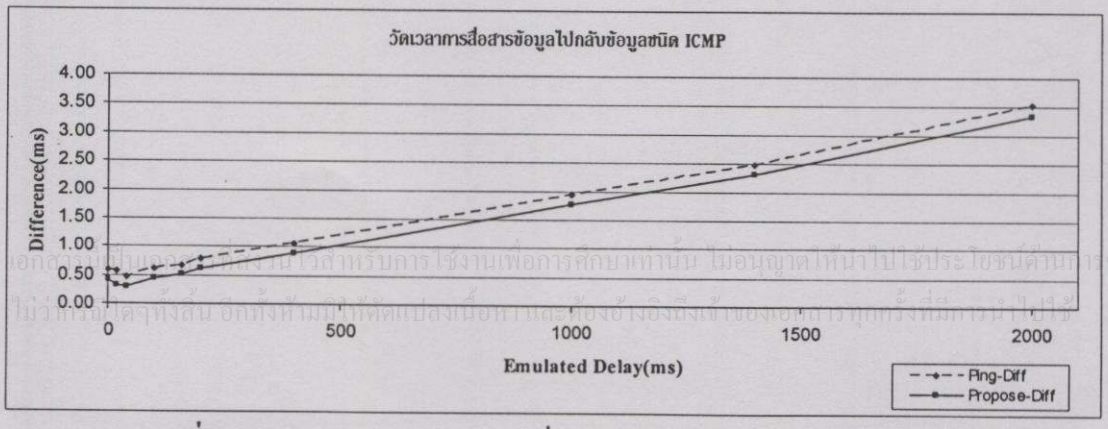
และ ค่าทดสอบอัลกอริทึมด้วยค่าสมบูรณ์ของความผิดพลาด โดยสมการที่ (4.3)

$$absolute\ error = \left| \frac{measured\ value - emulated\ delay}{emulated\ delay} \right| \tag{4.3}$$

โดยที่ *absolute error* หมายถึง ค่าสมบูรณ์ของความผิดพลาด

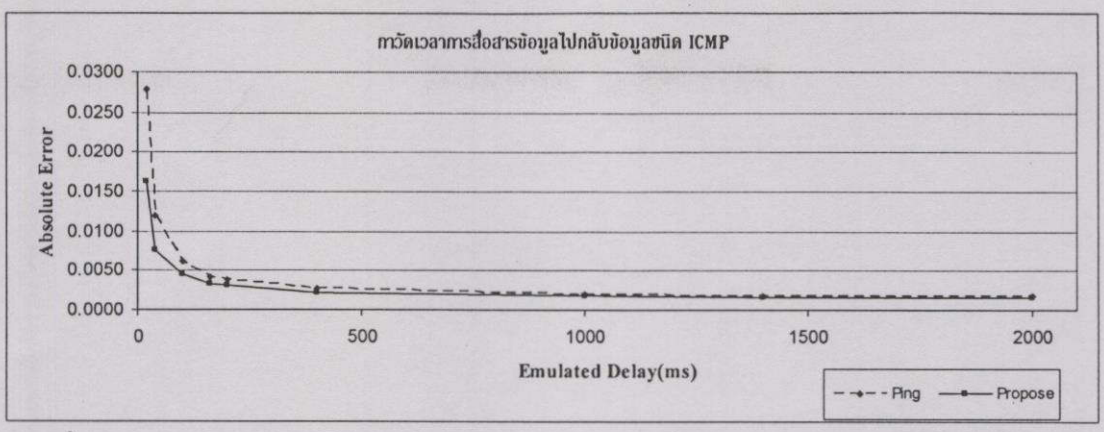
measured value หมายถึง ค่าเวลาที่ตรวจวัดได้จากเครื่องมือ

emulated delay หมายถึง ค่าเวลาการคอยก่อนการส่งข้อมูลออกจากตัวจำลองระบบ



ภาพที่ 4.5 ผลการวิเคราะห์เวลาการสื่อสารข้อมูลไปกลับข้อมูลชนิด ICMP

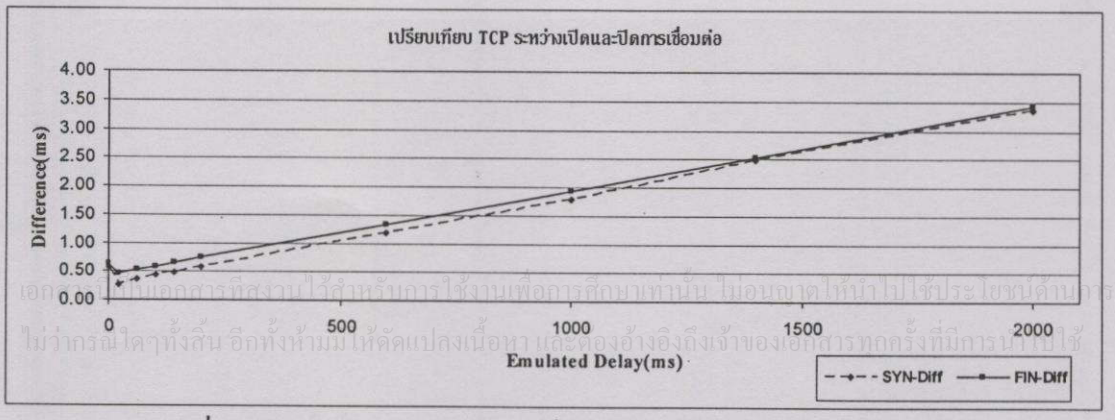
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.6 การทดสอบอัลกอริทึมในการวิเคราะห์เวลาการสื่อสารข้อมูลไปกลับข้อมูลชนิด ICMP

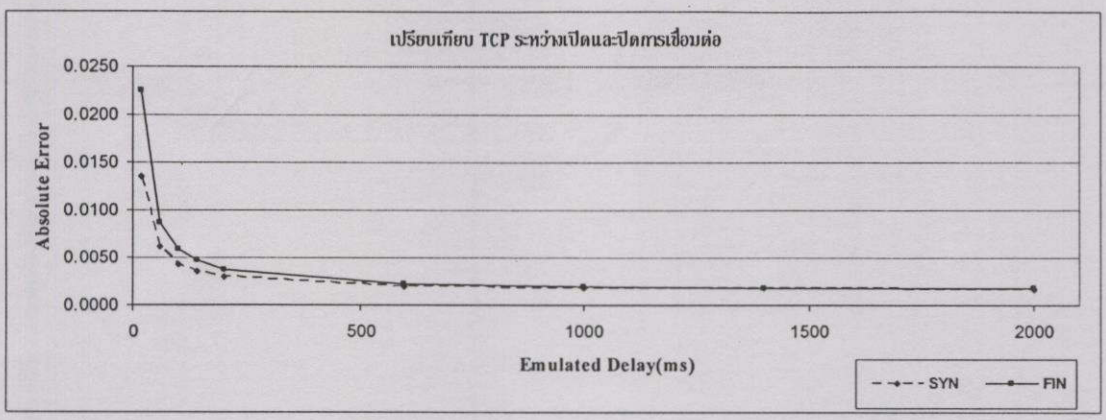
ในภาพที่ 4.5 แสดงลักษณะการตรวจวัดเวลาการสื่อสารข้อมูลไปกลับข้อมูลชนิด ICMP สามารถวัดได้ค่าเฉลี่ยของผลต่างจาก Emulated Delay ได้ใกล้เคียงกัน โดยเครื่องมือวิจัยและคำสั่ง Ping ตรวจวัดค่าเฉลี่ยได้ 1.090328 และ 1.276191 มิลลิวินาที ค่าเปอร์เซ็นต์ความผิดพลาดเมื่อเทียบกับคำสั่ง Ping วัดได้ 0.013% ค่าสัมประสิทธิ์สหสัมพันธ์วัดได้ 0.999820852 ส่วนในภาพที่ 4.6 แสดงลักษณะการวิเคราะห์จากค่าทดสอบอัลกอริทึมคือ เมื่อเพิ่มค่า Emulated Delay สูงขึ้นค่าความแตกต่างในการตรวจวัดค่าเวลาจะลดน้อยลงโดยลำดับจนกระทั่งไม่มีความแตกต่างกัน สรุปได้ว่าผลการวัดของทั้ง 2 เครื่องมือมีความใกล้เคียงกันสูงมาก โดยมีข้อสังเกตเพิ่มเติมคือค่าความแตกต่างของเวลาจะเพิ่มขึ้นตาม Emulated Delay เนื่องมาจากการกำหนดค่ารอบเวลาของ FreeBSD ซึ่งอ้างอิงจาก [14]

2. ทดลองตรวจวัดข้อมูลชนิด TCP โดยส่งข้อมูลชนิด TCP ด้วย TG(Traffic Generator) และ ICMP ด้วยคำสั่ง Ping จำนวน 1,000 แพ็กเก็ต จากฝั่งเครือข่ายภายในไปยังฝั่งเครือข่ายภายนอก ปรับค่า Emulated Delay บนตัวจำลองระบบระหว่าง 0 ถึง 1000 มิลลิวินาที และเปรียบเทียบผลการวัดกับค่าเวลาที่วัดได้จากคำสั่ง Ping โดยใช้ค่าสถิติต่าง ๆ คือ ค่าเฉลี่ย เปอร์เซ็นต์ความผิดพลาด และค่าสัมประสิทธิ์สหสัมพันธ์

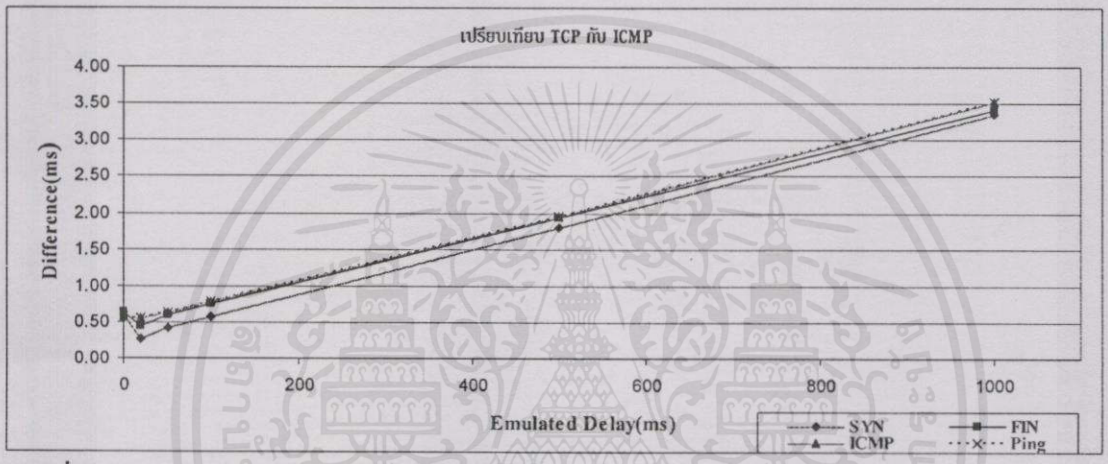


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 4.7 ผลการวิเคราะห์เวลาการสื่อสารข้อมูลไปกลับข้อมูลชนิด TCP



ภาพที่ 4.8 การทดสอบอัลกอริทึมในการวิเคราะห์เวลาการสื่อสารข้อมูลไปกลับข้อมูลชนิด TCP



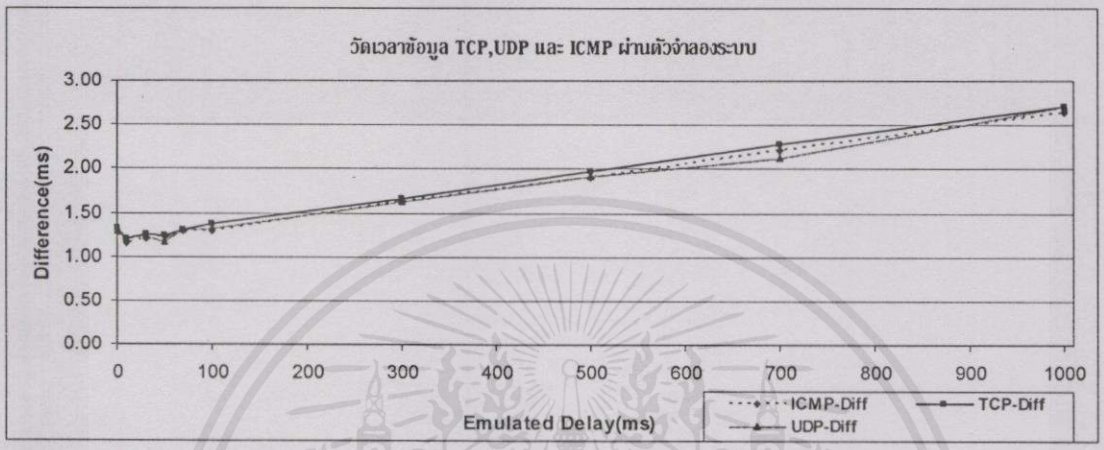
ภาพที่ 4.9 ผลการวิเคราะห์เวลาการสื่อสารข้อมูลไปกลับเปรียบเทียบระหว่างข้อมูลชนิด TCP และ ICMP

ในภาพที่ 4.7 แสดงลักษณะการเปรียบเทียบผลการตรวจวัดข้อมูลชนิด TCP ในช่วงเปิดและปิดการเชื่อมต่อ สามารถวัดค่าเฉลี่ยของผลต่างจาก Emulated Delay ได้ใกล้เคียงกัน โดยในช่วงเปิดและปิดการเชื่อมต่อข้อมูลชนิด TCP ตรวจวัดค่าเฉลี่ยได้ 1.167693 และ 1.289948 มิลลิวินาที ตามลำดับ ในภาพที่ 4.8 แสดงลักษณะค่าทดสอบอัลกอริทึมได้ว่า เมื่อค่า Emulated Delay สูงขึ้น ค่าความแตกต่างในการตรวจวัดเวลาจะลดน้อยลงจนมีนัยสำคัญที่ไม่มีความแตกต่างกัน ส่วนในภาพที่ 4.9 เมื่อเปรียบเทียบค่าเปอร์เซ็นต์ความผิดพลาดในการวัดค่าเวลาช่วงเปิดการเชื่อมต่อ(Sync) กับคำสั่ง Ping วัดได้ 0.030302% และในช่วงปิด(Fin)วัดได้ 0.008252% ค่าสัมประสิทธิ์สหสัมพันธ์ในช่วงเปิดการเชื่อมต่อเทียบกับคำสั่ง Ping วัดได้ 0.996926 และในช่วงปิดวัดได้ 0.999048 สรุปได้ว่าผลการวัดค่าเวลาการสื่อสารข้อมูลไปกลับด้วยเครื่องมือทั้ง 2 ชนิดนั้นมีความใกล้เคียงกันสูงมาก

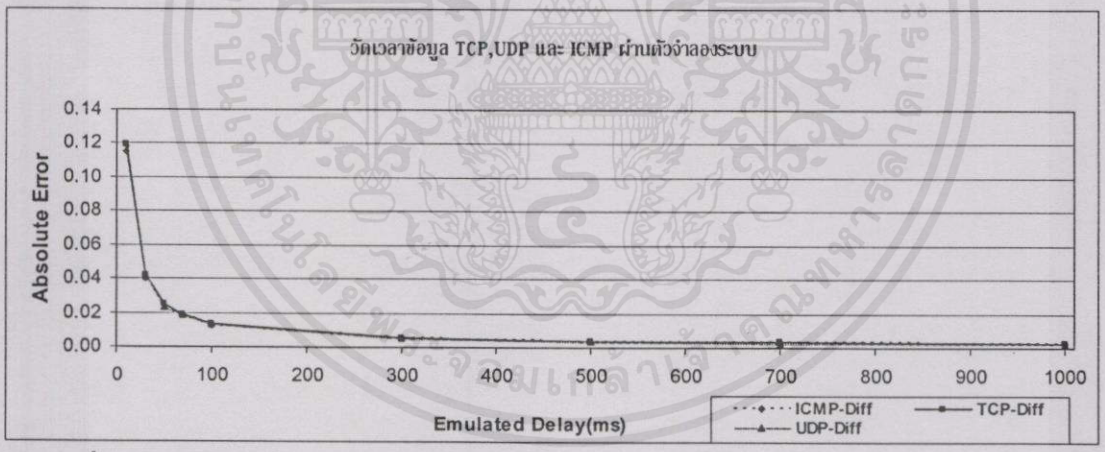
ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.3 การทดลองตรวจวัดเวลาการสื่อสารข้อมูลผ่านตัวจำลองระบบ

ทดลองตรวจวัดโดยส่งข้อมูลชนิด ICMP, TCP และ UDP ความยาวขนาด 1,500 ไบต์ จำนวน 1,000 แพ็กเก็ตจากฝั่งเครือข่ายภายในไปยังฝั่งเครือข่ายภายนอก ปรับค่า Emulated Delay ระหว่าง 0 ถึง 1000 มิลลิวินาที และเปรียบเทียบผลการวัดกับครึ่งหนึ่งของค่าเวลาที่วัดได้จากคำสั่ง Ping



ภาพที่ 4.10 ผลการวิเคราะห์เวลาการสื่อสารข้อมูลผ่านตัวจำลองระบบ



ภาพที่ 4.11 การทดสอบอัลกอริทึมในการวิเคราะห์เวลาการสื่อสารข้อมูลผ่านตัวจำลองระบบ

ในภาพที่ 4.10 และ 4.11 แสดงลักษณะการเปรียบเทียบผลการวัดข้อมูล ICMP กับคำสั่ง Ping สามารถวัดค่าเฉลี่ยของผลต่างจาก Emulated Delay ได้ใกล้เคียงกัน โดยเครื่องมือวิจัยตรวจวัดค่าเวลาข้อมูลชนิด TCP และ UDP ได้ 1.630671 และ 1.585756 มิลลิวินาที ส่วนคำสั่ง Ping วัดได้คือ 1.582579 มิลลิวินาที ค่าเปอร์เซ็นต์ความผิดพลาดเมื่อเทียบกับคำสั่ง Ping ข้อมูลชนิด TCP วัดได้ 0.017325% และข้อมูลชนิด UDP วัดได้ 0.001144524% ค่าสัมประสิทธิ์สหสัมพันธ์เมื่อเทียบกับการวัดข้อมูลชนิด ICMP ด้วยคำสั่ง Ping ข้อมูลชนิด TCP วัดได้ 1.00 และข้อมูลชนิด UDP วัดได้

0.999999992 สรุปได้ว่าผลจากการทดลองตรวจวัดเวลาการสื่อสารข้อมูลผ่านตัวจำลองระบบสามารถวัดได้ค่าที่ใกล้เคียงกันสูงมาก

4.3.4 การทดลองตรวจวัดความสูญเสียข้อมูลบนตัวจำลองระบบ

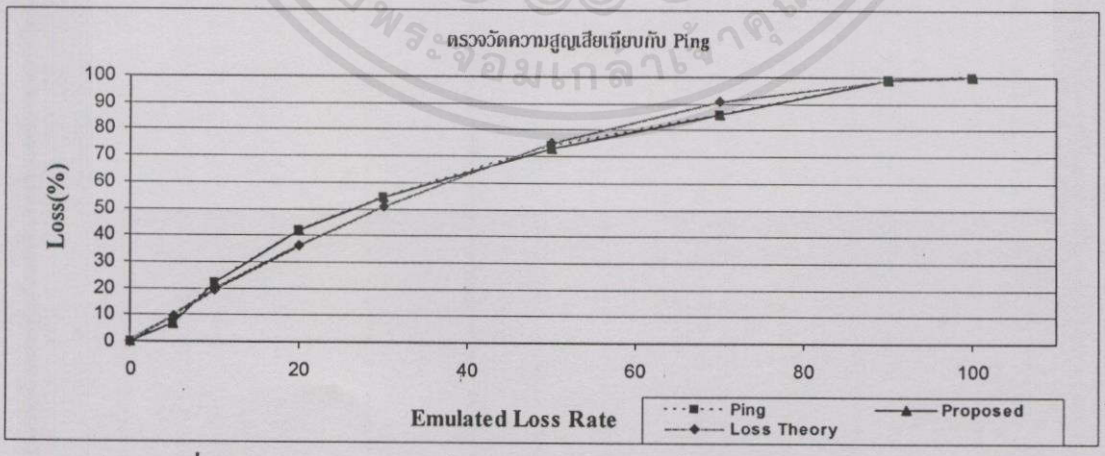
ทดลองตรวจวัดความสูญเสียข้อมูลสื่อสารระหว่างเครื่องมือวิจัยและ คำสั่ง Ping โดยส่งข้อมูลชนิด ICMP ด้วยคำสั่ง Ping จำนวน 100 แพ็กเก็ต จากฝั่งเครือข่ายภายในไปยังฝั่งเครือข่ายภายนอก ทำซ้ำการทดลองโดยปรับค่าความสูญเสียข้อมูลให้กับตัวจำลองระบบระหว่าง 0 ถึง 100 เปอร์เซ็นต์ หรือนิยามเรียกค่านี้ว่า Emulated Loss โดยใช้คำสั่ง ipfw บน FreeBSD ด้วยรูปแบบดังนี้

```
ipfw add pipe 1 ip from any to any in
ipfw pipe 1 config plr [n]
```

โดยที่ [n] หมายถึง ตัวเลขจำนวนเต็ม แทนความน่าจะเป็นของอัตราการสร้างความสูญเสียให้กับข้อมูลที่สื่อสารผ่านตัวจำลองระบบ เปรียบเทียบผลการวัดกับค่าความสูญเสียตามทฤษฎีโดยสมการที่ (4.4)

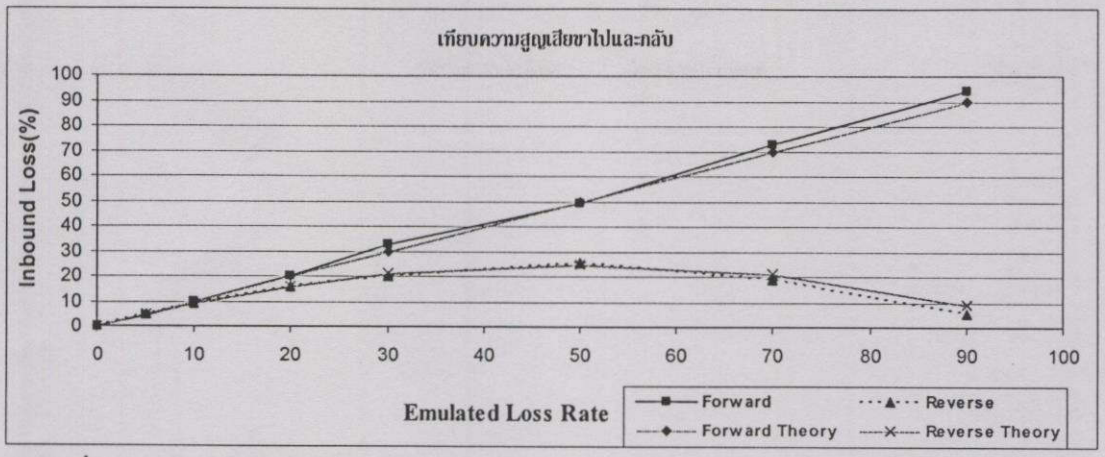
$$theoretical\ loss = loss\ rate + \left[\frac{pkts\ reverse * loss\ rate}{100} \right] \qquad (4.4)$$

โดยที่ *theoretical loss* หมายถึง ค่าความสูญเสียตามทฤษฎี
pkts reverse หมายถึง ข้อมูลที่เหลือจากความสูญเสีย
loss rate หมายถึง อัตราความสูญเสียข้อมูล

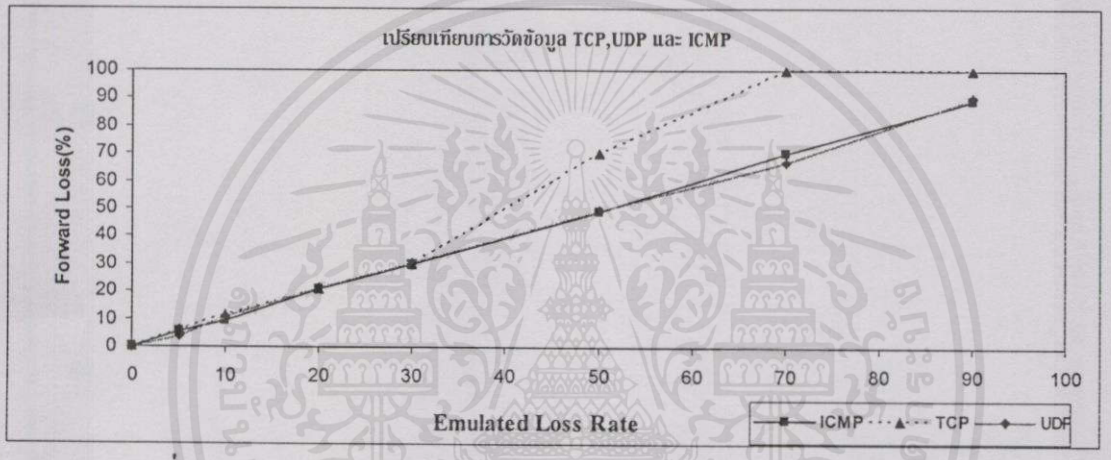


ภาพที่ 4.12 ผลการวิเคราะห์ความสูญเสียข้อมูลสื่อสารผ่านตัวจำลองระบบ

เอกสารนี้เป็นลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย ห้ามนำไปใช้เพื่อประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.13 ผลการวิเคราะห์ความสูญเสียข้อมูลผ่านตัวจำลองระบบแยกตามทิศทางการสื่อสาร

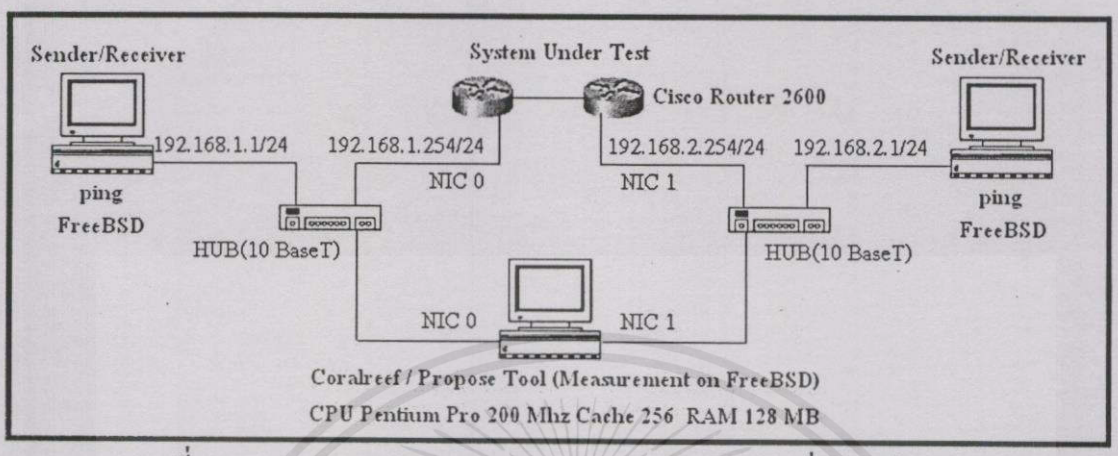


ภาพที่ 4.14 ผลการวิเคราะห์ความสูญเสียข้อมูลชนิดต่าง ๆ ผ่านตัวจำลองระบบ

ในภาพที่ 4.12 แสดงลักษณะการเปรียบเทียบผลการวัดความสูญเสียข้อมูลผ่านตัวจำลองระบบทั้ง 2 เครื่องมือสามารถวัดได้ไม่แตกต่างกัน และเมื่อเทียบกับค่าความสูญเสียตามทฤษฎีลักษณะที่ได้มีความใกล้เคียงกันมาก ในภาพที่ 4.13 เครื่องมือวิจัยสามารถพิจารณาความสูญเสียได้ทั้งขาไปและกลับ ค่าที่วัดได้เมื่อเทียบกับความสูญเสียตามทฤษฎีแล้วมีความใกล้เคียงกัน ส่วนภาพที่ 4.14 แสดงลักษณะการวัดข้อมูลชนิด UDP และ ICMP สามารถวัดได้ใกล้เคียงกัน ส่วนข้อมูลชนิด TCP จะแตกต่างออกไปเมื่อตัวจำลองระบบมีอัตราความสูญเสียมากกว่า 30% ขึ้นไป ค่าเปอร์เซ็นต์ความแตกต่างเมื่อเทียบกับอัตราความสูญเสียที่กำหนดให้กับตัวจำลองระบบ ข้อมูลชนิด ICMP วัดได้ 3.126531% ข้อมูลชนิด TCP วัดได้ 18.72279% และข้อมูลชนิด UDP วัดได้ 6.678974% ค่าสัมประสิทธิ์สหสัมพันธ์เมื่อเทียบกับอัตราความสูญเสียที่กำหนดให้กับตัวจำลองระบบ ข้อมูลชนิด ICMP วัดได้ 0.9998 ข้อมูลชนิด TCP วัดได้ 0.9816 และ ข้อมูลชนิด UDP วัดได้ 0.9992 สรุปได้ว่าผลการตรวจวัดความสูญเสียข้อมูลสามารถวัดได้ไม่แตกต่าง และมีค่าใกล้เคียงกันสูงมากเมื่อเทียบจากทั้งความสูญเสียที่คำนวณตามทฤษฎีและเครื่องมือวัดอื่น

4.3.5 การทดลองตรวจวัดด้วยสัญญาณการสื่อสารข้อมูลจริง

ทดลองเปลี่ยนตัวจำลองระบบในการวิเคราะห์ข้อมูล โดยใช้อุปกรณ์ที่จำลองจากคอมพิวเตอร์ส่วนบุคคลไปเป็นอุปกรณ์ Cisco Router รุ่น 2600 ดังแสดงได้ในภาพที่ 4.15



ภาพที่ 4.15 แบบจำลองการทดลองตรวจวัดบนสัญญาณการสื่อสารข้อมูลจริง

ทำการทดลองโดยตรวจวัดเวลาการสื่อสารข้อมูลผ่านตัวจำลองระบบ โดยส่งข้อมูลชนิด ICMP ด้วยคำสั่ง Ping จำนวน 200 แพ็กเก็ตต่อการทดลอง การทดลองแต่ละครั้งทำการปรับค่าความยาวแพ็กเก็ตระหว่าง 64 ถึง 1500 bytes และปรับค่าอัตราการรับส่งข้อมูลระหว่าง 64 Kbit/Sec ถึง 1 Mbit/Sec โดยใช้รูปแบบคำสั่ง คือ

clock rate <transmission rate>

โดยที่ transmission rate หมายถึง อัตราการรับส่งข้อมูลในหน่วย บิตต่อวินาที

ทำการทดลองเปรียบเทียบด้วยวิธีการแบบเดียวกันกับตัวจำลองระบบที่เป็นคอมพิวเตอร์ส่วนบุคคล โดยแตกต่างกันเฉพาะการกำหนดอัตราการรับส่งข้อมูลด้วย ipfw คือ

ipfw add pipe 1 ip from any to any in

ipfw pipe 1 config bw <n> Kbit/Sec

โดยที่ <n> หมายถึง อัตราการรับส่งข้อมูลในหน่วย กิโลบิตต่อวินาที

เปรียบเทียบผลการวัดค่าเวลาระหว่างเครื่องมือวิจัย และคำสั่ง Ping โดยวาดกราฟผลต่างจากเวลาในการรับส่งข้อมูลต่อหน่วยที่คำนวณได้จากสมการที่ (4.5)

$$T\_delay_i = \frac{(length_i + NHrd)\, x\, 8\, x 1000}{Bw_i} \qquad (4.5)$$

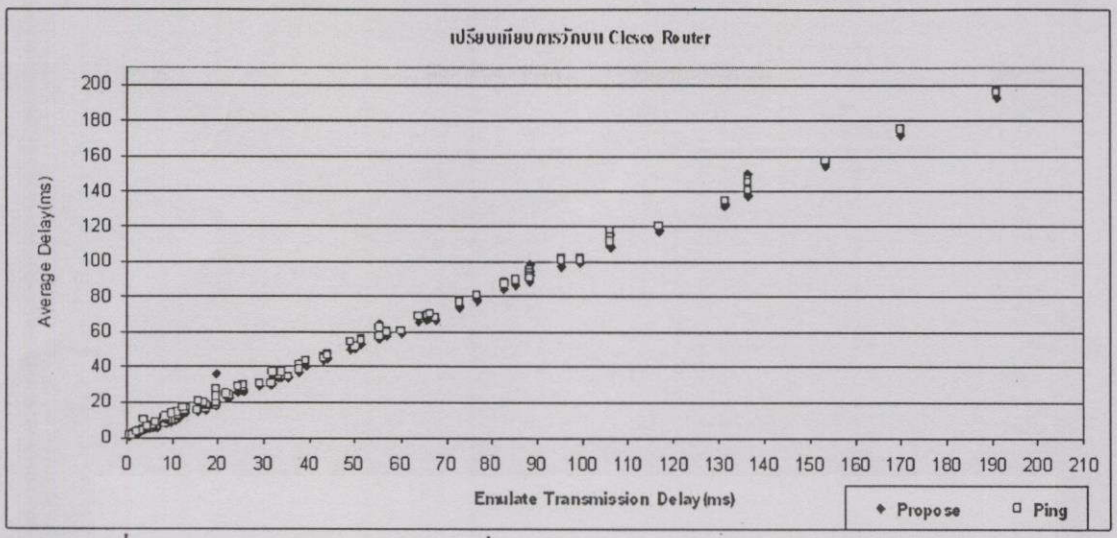
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ T_delay_i หมายถึง เวลาที่ใช้ในการส่งข้อมูลต่อหน่วย

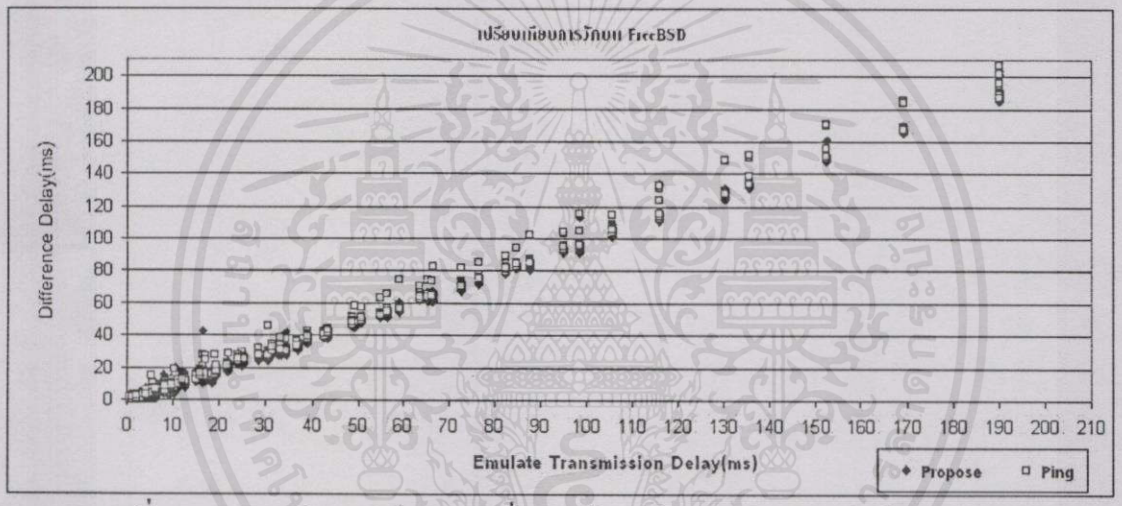
$length_i$ หมายถึง ความยาวข้อมูล

$NHrd$ หมายถึง ความยาวข้อมูลส่วนหัวของแพ็กเก็ต

Bw_i หมายถึง อัตราการส่งข้อมูลต่อหน่วยเวลา



ภาพที่ 4.16 ผลการวิเคราะห์เวลาการสื่อสารข้อมูลผ่านตัวจำลองระบบด้วย Cisco Router



ภาพที่ 4.17 ผลการวิเคราะห์เวลาการสื่อสารข้อมูลผ่านตัวจำลองระบบด้วย FreeBSD

ในภาพที่ 4.16 และ 4.17 แสดงลักษณะการวิเคราะห์เวลาการสื่อสารข้อมูลผ่านตัวจำลองระบบโดยใช้กราฟแบบ XY กระจาย สรุปได้ว่าการวัดค่าเวลาบนตัวจำลองทั้ง 2 ชนิดได้ลักษณะการกระจายที่เหมือนกัน ค่าเวลาที่วัดได้มีการกระจายเพิ่มขึ้นตามค่าเวลาคำนวณที่ใช้ในการส่งข้อมูลต่อหน่วย สรุปได้ว่าผลการวัดค่าเวลาการสื่อสารข้อมูลด้วยสัญญาณการสื่อสารจริง เครื่องมือวิจัยสามารถวัดได้จริง โดยคุณลักษณะค่าเวลาที่ตรวจวัดได้เป็นไปตามหลักการสื่อสารเมื่อเทียบค่าเวลาที่ใช้ในการส่งข้อมูลต่อความยาวแพ็กเก็ตหนึ่ง ๆ

4.3.6 สรุปผลการวิเคราะห์ความถูกต้อง

ผลการทดลองและการวิเคราะห์ความถูกต้องเมื่อเปรียบเทียบกับเครื่องมือวัดชนิดอื่นสามารถสรุปข้อมูลในเชิงสถิติได้ ดังตารางที่ 4.2 และ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีนำไปจำหน่าย แจกจ่าย หรือนำไปใช้ในรูปแบบอื่นใดที่คล้ายคลึงกัน

ตารางที่ 4.2 สรุปการวิเคราะห์ความถูกต้องเมื่อเทียบกับเครื่องมือวัดชนิดอื่น

ลำดับที่	รายการที่วัด	ชนิดข้อมูลที่วัด	ค่าความแตกต่าง(%)		สัมประสิทธิ์สหสัมพันธ์	
			ค่า	เปรียบเทียบกับ	ค่า	เปรียบเทียบกับ
1	Length	ทุกชนิด	0.000000	Coral	1.000000	Coral
2	Inter-Arrival Times	ทุกชนิด	0.002100	Coral	0.990000	Coral
3	Response Times	ICMP	0.013000	Ping	0.999821	Ping
4	Response Times	TCP(Sync)	0.030302	Ping	0.996926	Ping
5	Response Times	TCP(Fin)	0.008252	Ping	0.999048	Ping
6	Delay on SUT	TCP	0.017325	Ping / 2	1.000000	Ping / 2
7	Delay on SUT	UDP	0.001145	Ping / 2	0.999999	Ping / 2
8	Loss on SUT	ICMP	0.000000	Ping	1.000000	Ping
9	Delay on FreeBSD	ICMP	0.087146	Ping / 2	0.999780	Ping
10	Delay on Cisco Router	ICMP	0.185527	Ping / 2	0.999156	Ping

ตารางที่ 4.3 สรุปการวิเคราะห์ความถูกต้องเมื่อเทียบกับค่า Emulated

ลำดับที่	รายการที่วัด/เครื่องมือที่วัด	ชนิดข้อมูลที่วัด	ค่าความผิดพลาด(%)	
			ค่า	เปรียบเทียบกับ
1	Response Times / Propose	ICMP	0.465500	Emulated-delay
2	Response Times / Propose	TCP(Syn)	0.375400	Emulated -delay
3	Response Times / Propose	TCP(Fin)	0.530100	Emulated -delay
4	Response Times / Ping	ICMP	0.605100	Emulated -delay
5	Delay on SUT / Propose	TCP	0.488180	Emulated -delay
6	Delay on SUT / Propose	UDP	0.448660	Emulated -delay
7	Delay on SUT / Ping	ICMP	0.405940	Emulated -delay
8	Delay on SUT / Ping	ICMP	3.126531	Emulated -loss
9	Loss on SUT / Propose	TCP	18.722790	Emulated -loss
10	Loss on SUT / Propose	UDP	6.678974	Emulated -loss
11	Delay on Cisco / Propose	ICMP	7.513400	Transmission-delay
12	Delay on Cisco / Ping	ICMP	18.049600	Transmission-delay
13	Delay on FreeBSD / Propose	ICMP	12.062900	Transmission-delay
14	Delay FreeBSD / Ping	ICMP	21.372200	Transmission-delay

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการวิเคราะห์เครื่องมือวิจัยเทียบกับเครื่องมือชนิดอื่น โดยรวมแล้วเครื่องมือวิจัยสามารถตรวจวัดได้อย่างถูกต้องเมื่อพิจารณาตามค่าสถิติที่ได้จากข้อมูลชุดเดียวกัน โดยเฉพาะค่า

สัมประสิทธิ์สหสัมพันธ์ที่ตรวจวัดได้จากทั้ง 2 เครื่องมือนั้นมีค่าที่ใกล้เคียงกับ 1.00 ในเกือบทุกการทดลอง ซึ่งแปรความหมายจากค่าสถิติได้ว่าทั้ง 2 เครื่องมือนั้น สามารถตรวจวัดข้อมูลได้ใกล้เคียงกันมาก ส่วนค่าสถิติของผลต่างหรือเปอร์เซ็นต์ความผิดพลาดเมื่อเทียบกับเครื่องมือชนิดอื่นส่วนใหญ่มีค่าที่แตกต่างกันน้อยมาก ซึ่งโดยทั่วไปค่าที่สามารถยอมรับได้ในทางสถิตินั้นต้องอยู่ในระดับไม่เกิน 0.05 เปอร์เซ็นต์ และจากการทดลองตรวจวัดข้อมูลทั้ง 5 คุณลักษณะ(ในตารางที่ 4.2 ลำดับที่ 1-8)ผลต่างที่วิเคราะห์ได้ก็อยู่ในเกณฑ์ที่ไม่เกินกำหนด

ในตารางที่ 4.3 แสดงการวิเคราะห์เครื่องมือวิจัยเมื่อเทียบกับค่า Emulated ค่าผลต่างในการวัดเวลาการสื่อสารข้อมูลไปกลับและเวลาการสื่อสารข้อมูลผ่านตัวจำลองระบบ เมื่อเทียบระหว่างเครื่องมือวิจัยและเครื่องมือชนิดอื่นลักษณะที่ได้มีความใกล้เคียงกัน โดยค่าที่ตรวจวัดได้จากเครื่องมือวิจัยจะมีค่าที่ใกล้เคียงกับค่า Emulated มากกว่าเครื่องมือวัดชนิดอื่น ส่วนความสูญเสียที่วัดได้ค่าผลต่างอยู่ในระดับสูง(ในตารางที่ 4.3 ลำดับที่ 8-10)นั้น เป็นผลมาจากการเปรียบเทียบกับความน่าจะเป็นที่กำหนดค่าความสูญเสียให้กับตัวจำลองระบบ ซึ่งค่าที่กำหนดเป็นค่าสุ่มและไม่คงที่ และไม่สามารถควบคุมให้เกิดขึ้นตามจำนวนที่กำหนดได้ ดังนั้นความแตกต่างจึงมีค่ามากขึ้น และแปรปรวนตามเหตุการณ์ความสูญเสียที่จะเกิดขึ้นจริง

4.4 การศึกษาประสิทธิภาพของเครื่องมือวิจัย

นอกจากความถูกต้องในการตรวจวัดข้อมูลแล้ว เครื่องมือที่ออกแบบเป็นพาสซีฟควรจะมีความสามารถในการตรวจวัดได้อย่างต่อเนื่อง หรือรองรับปริมาณการสื่อสารข้อมูลบนเครือข่ายอินเตอร์เน็ตทั่วไปได้ ดังนั้นการวิเคราะห์ความสามารถโดยพื้นฐานของเครื่องมือวัดจึงเป็นอีกปัจจัยหนึ่ง ที่จะทำให้รู้ขีดความสามารถที่จำกัดของเครื่องมือ และนำไปสู่การศึกษาเพื่อวิเคราะห์ปัจจัยด้านต่าง ๆ ที่ส่งผลกระทบทำให้ความสามารถของเครื่องมือนั้นต่ำลง นอกจากนี้ข้อมูลที่ได้จากศึกษาอาจนำไปใช้ปรับแต่งให้เครื่องมือมีประสิทธิภาพเพิ่มขึ้นและสามารถนำไปใช้งานได้จริง โดยมีเป้าหมายที่จะศึกษาปัจจัยที่สำคัญดังนี้

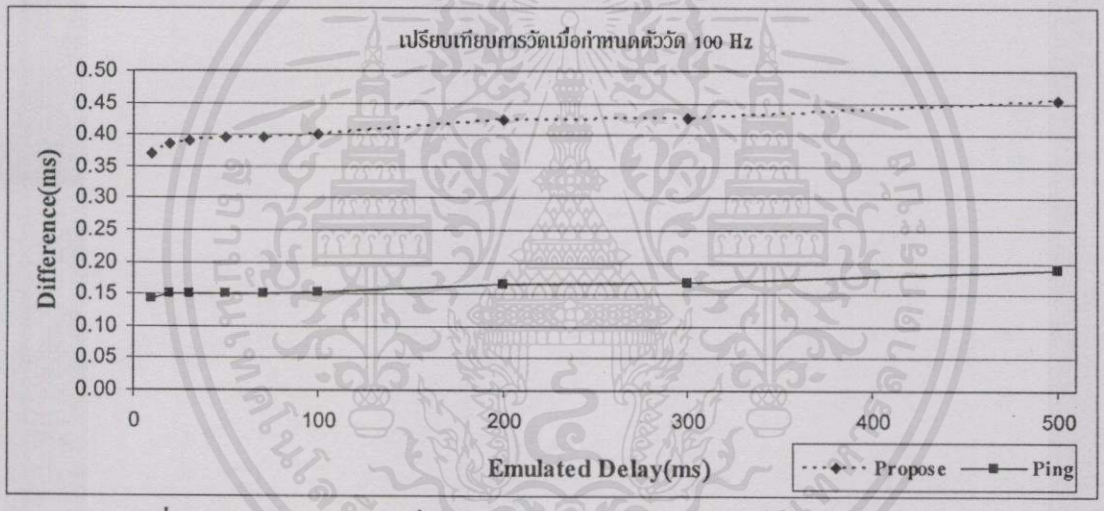
4.4.1 การศึกษาผลกระทบของ Hz Option กับการตรวจวัดค่าเวลา

จากกรณีศึกษาเรื่อง "การใช้งานระบบปฏิบัติการ FreeBSD เพื่อใช้เป็นหน่วยประมวลผลในการรับส่งข้อมูลผ่านเครือข่าย" ระบบ Kernel ของ FreeBSD สามารถกำหนดให้ใช้พารามิเตอร์ (Parameters)ที่สำคัญในการสลับรอบเวลาของหน่วยประมวลผลต่าง ๆ ได้ โดยเรียกพารามิเตอร์นี้ว่า Hz Option ซึ่งระบบปฏิบัติการ FreeBSD ได้เสนอแนะให้ใช้ค่าที่ดีที่สุดคือ 1000 Hz หรือ 1 มิลลิวินาทีต่อรอบประมวลผล ซึ่งจะช่วยทำให้ผลการตอบสนองต่อการปฏิบัติงานของหน่วยประมวลผลต่าง ๆ นั้นดีขึ้น ดังนั้นจากการที่เครื่องมือวิจัยถูกออกแบบให้แยกหน่วยประมวลผลออกเป็น 3 กลุ่มหลัก คือ หน่วยตรวจจับ หน่วยแยกชนิด และหน่วยวิเคราะห์ข้อมูล ค่ารอบเวลาอาจมีผลทำให้หน่วยตรวจจับข้อมูลได้รับทรัพยากรเวลาที่ดีขึ้น และอาจส่งผลทำให้ความ

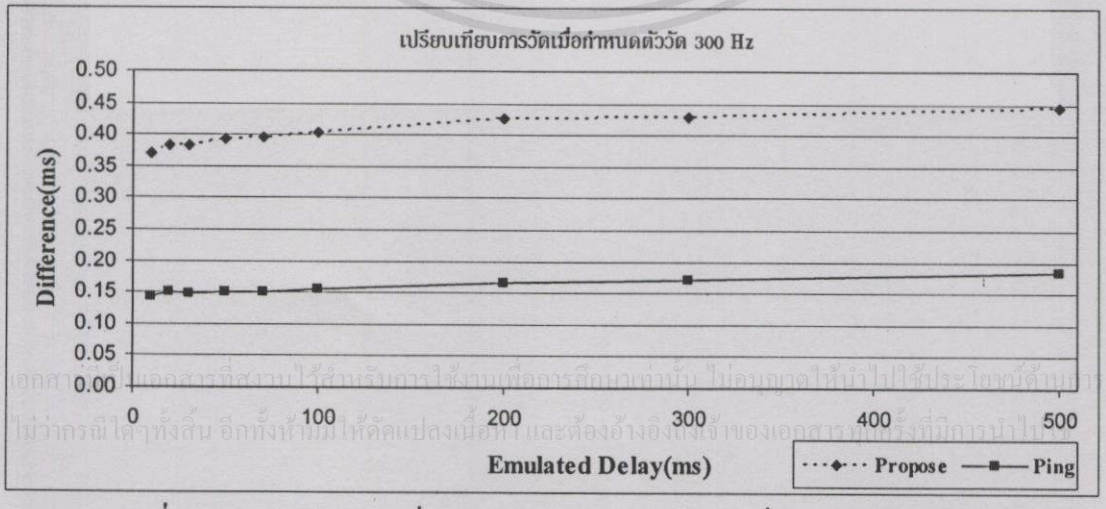
ลิขสิทธิ์เป็นของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น และห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แม่นยำในการกำกับเวลาข้อมูลต่อหน่วยนั้นมีความแตกต่างกัน หรือคาดการณ์สมมติฐานได้ว่าค่า Hz Option น่าจะมีผลกระทบต่อการตรวจวัดข้อมูลของเครื่องมือวิจัยแตกต่างกัน ซึ่งการทดลองอาจนำมาซึ่งค่า Hz Option ที่เหมาะสมที่จะนำไปใช้กับเครื่องมือวิจัย และอาจช่วยสะท้อนถึงระดับความแม่นยำของการตรวจวัดค่าเวลา หรือสะท้อนถึงความเร็วในการวิเคราะห์ข้อมูลของเครื่องมือด้วย ดังนั้นเพื่อศึกษาปัจจัยนี้จึงได้ทำการทดลองโดยปรับค่า Hz Option ทั้งบนตัวจำลองระบบและตัววัดเพื่อแสดงพฤติกรรมค่าเวลาที่ตรวจวัดได้ดังนี้

1. การทดลองปรับค่า Hz Option บนเครื่องมือวัด ทดลองตรวจวัดเวลาการสื่อสารข้อมูลผ่านตัวจำลองระบบ โดยส่งข้อมูลชนิด ICMP ด้วยคำสั่ง Ping จำนวน 1,000 แพ็กเก็ตต่อการทดลอง การทดลองแต่ละครั้งทำการปรับค่า Emulated Delay บนตัวจำลองระบบระหว่าง 10 ถึง 500 มิลลิวินาที กำหนดค่า Hz Option บนเครื่องมือวัดระหว่าง 300 ถึง 5,000 Hz ส่วนตัวจำลองระบบมีค่าคงที่ คือ 1,000 Hz และวาดกราฟผลต่างเวลาจาก Emulated Delay

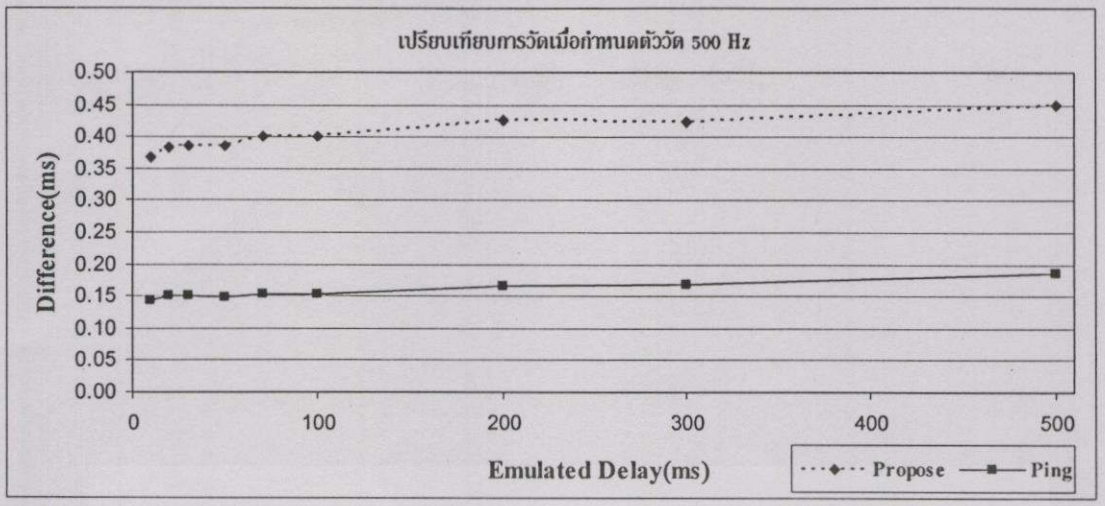


ภาพที่ 4.18 ผลการทดลองเมื่อกำหนดค่า Hz Option บนเครื่องมือวัดเป็น 100 Hz

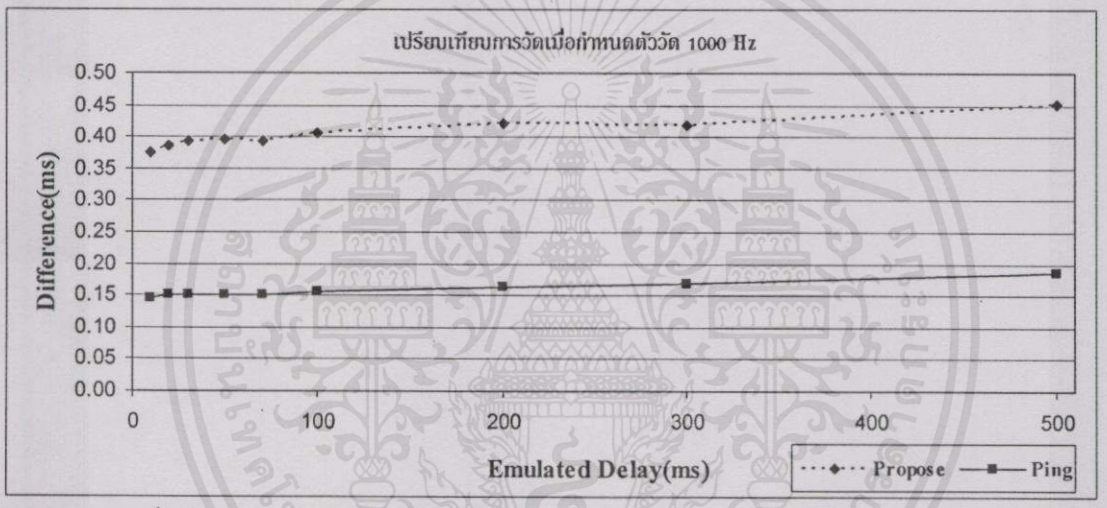


ภาพที่ 4.19 ผลการทดลองเมื่อกำหนดค่า Hz Option บนเครื่องมือวัดเป็น 300 Hz

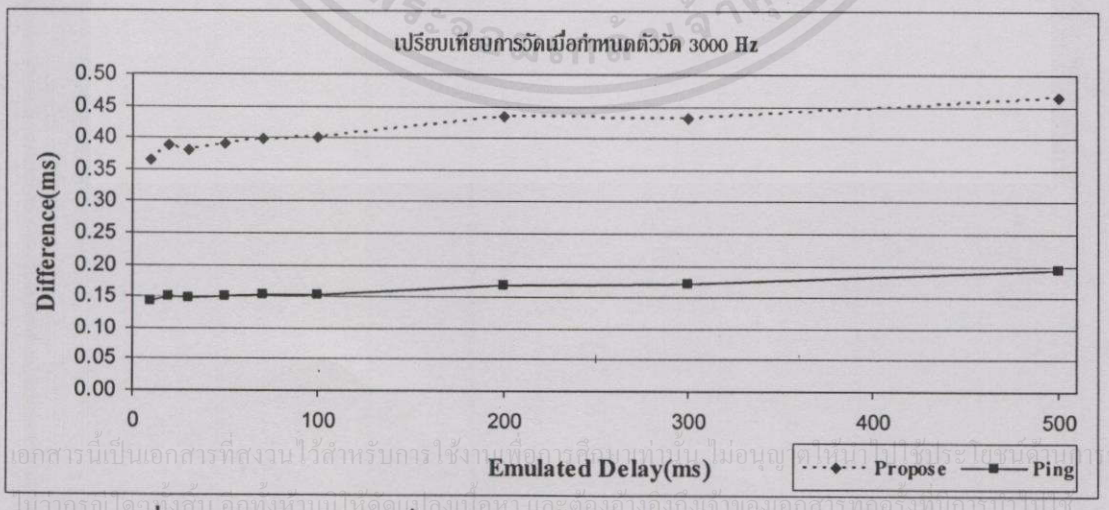
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.20 ผลการทดลองเมื่อกำหนดค่า Hz Option บนเครื่องมือวัดเป็น 500 Hz

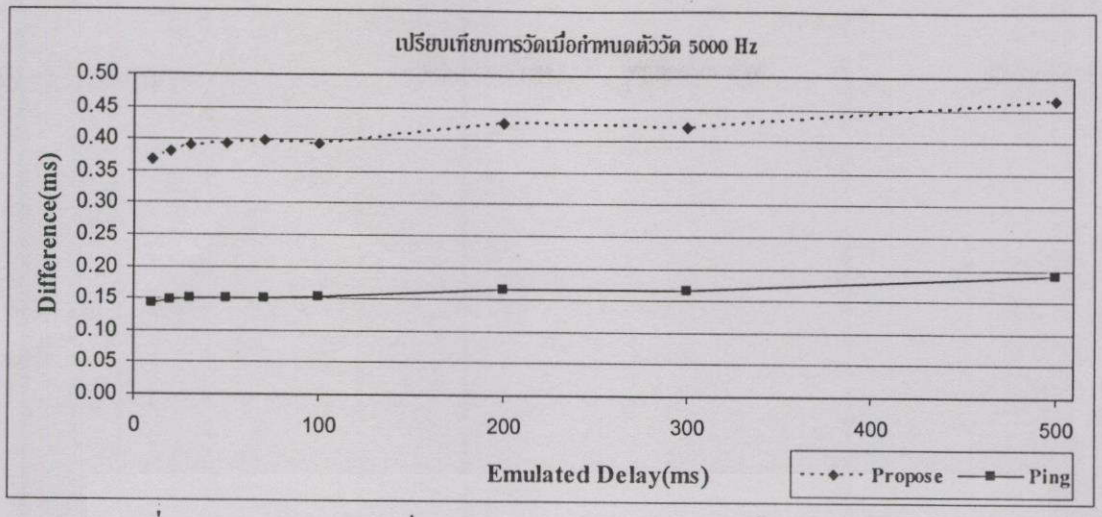


ภาพที่ 4.21 ผลการทดลองเมื่อกำหนดค่า Hz Option บนเครื่องมือวัดเป็น 1000 Hz

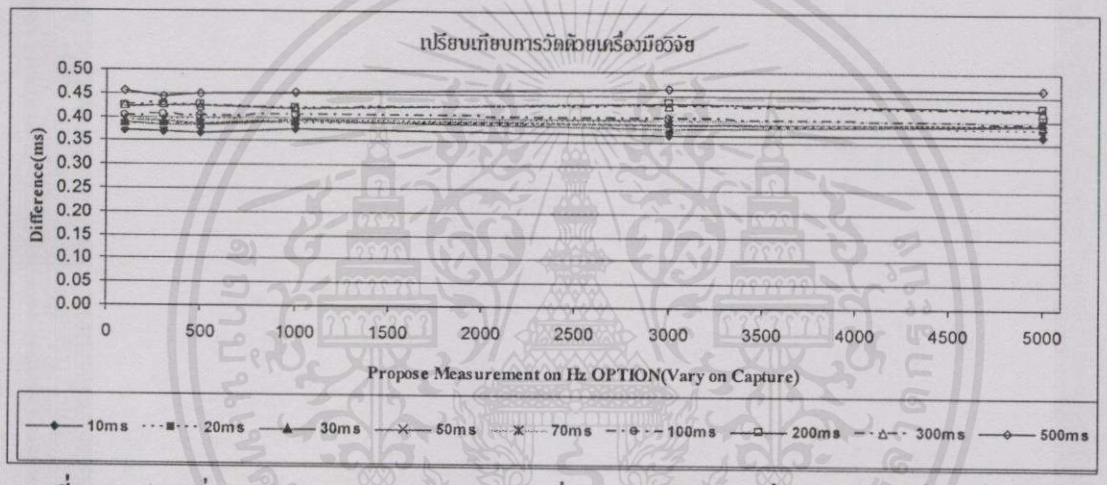


ภาพที่ 4.22 ผลการทดลองเมื่อกำหนดค่า Hz Option บนเครื่องมือวัดเป็น 3000 Hz

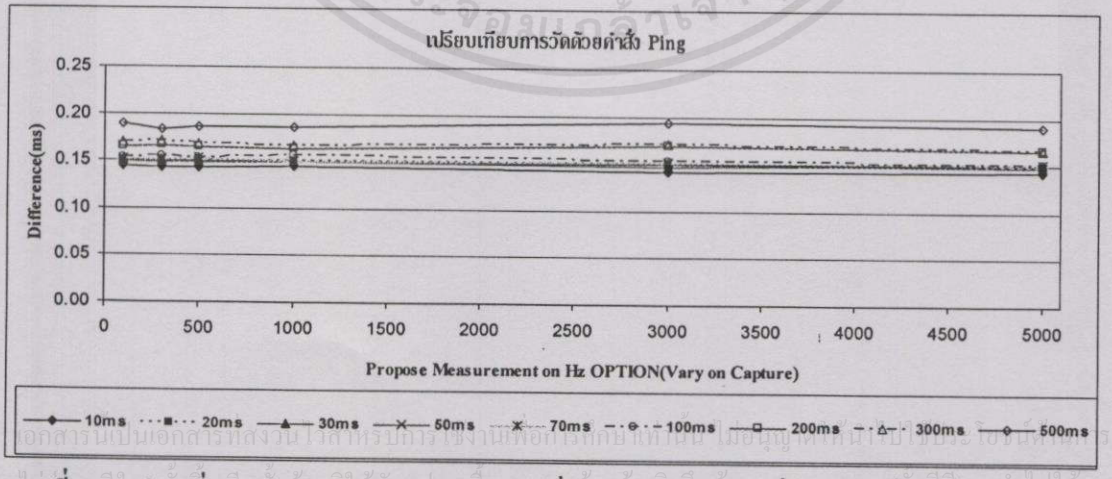
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.23 ผลการทดลองเมื่อกำหนดค่า Hz Option บนเครื่องมือวัดเป็น 5000 Hz



ภาพที่ 4.24 ค่าเฉลี่ยผลต่างจาก Emulated Delay ที่ตรวจวัดได้จากเครื่องมือวิจัย เมื่อเปลี่ยนค่า Hz Option บนเครื่องมือวัด

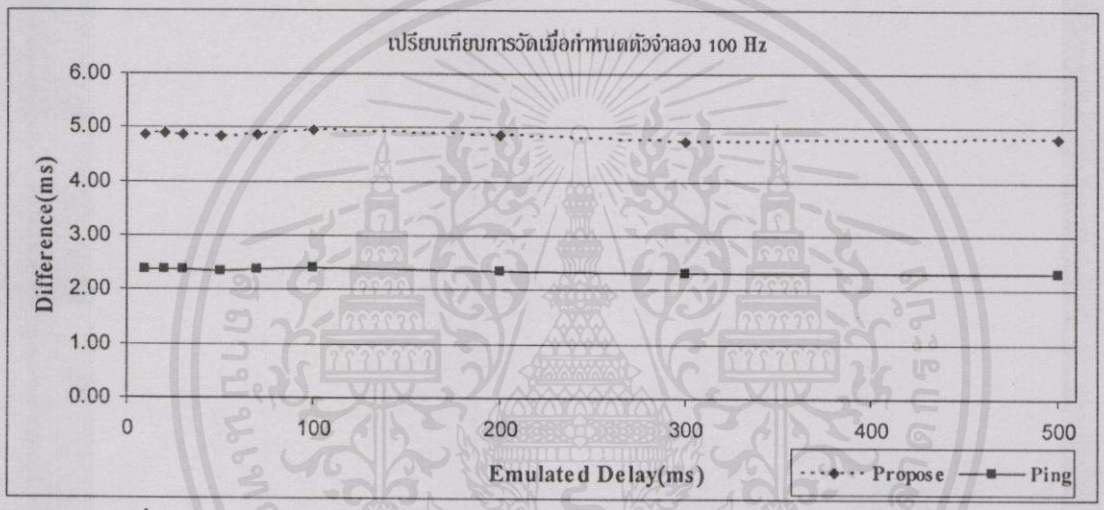


ภาพที่ 4.25 ค่าเฉลี่ยผลต่างจาก Emulated Delay ที่ตรวจวัดได้จากคำสั่ง Ping เมื่อเปลี่ยนค่า Hz Option บนเครื่องมือวัด

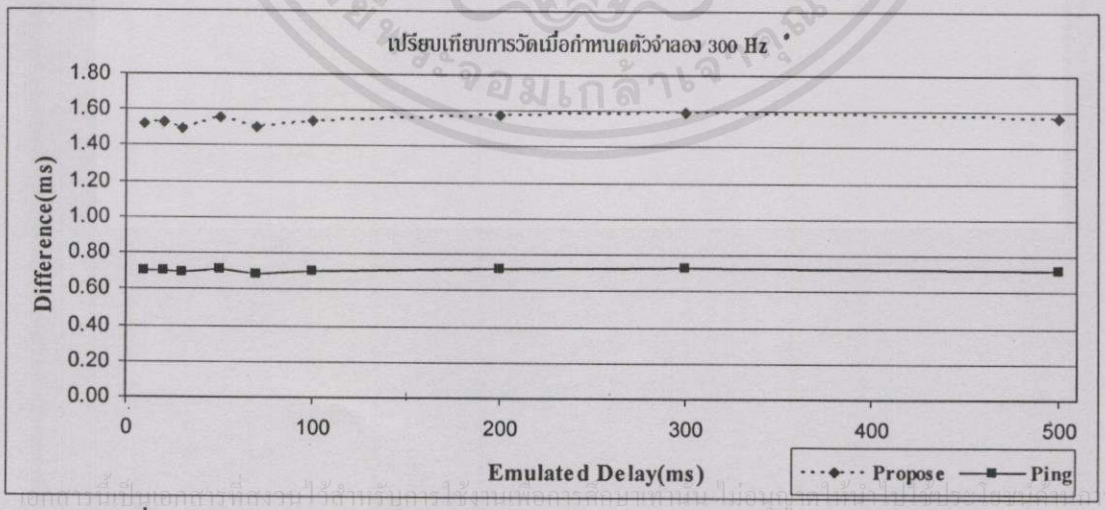
ลิขสิทธิ์มหาวิทยาลัยเชียงใหม่
Copyright© by Chiang Mai University
All rights reserved

ในภาพที่ 4.18, 4.19, 4.20, 4.21, 4.22, 4.23, 4.24 และ 4.25 พบว่าพฤติกรรมการวัดค่าเวลาการสื่อสารข้อมูลผ่านตัวจำลองระบบ ผลการปรับเปลี่ยนค่า Hz Option ในระดับต่าง ๆ คุณลักษณะที่ได้ไม่มีความแตกต่างกัน ค่า Hz Option ที่ตรวจวัดได้จากทั้งเครื่องมือวิจัย(Propose Tools) และคำสั่ง Ping ในแต่ละระดับมีความใกล้เคียงกัน

2. การทดลองปรับค่า Hz Option บนตัวจำลองระบบ ทดลองตรวจวัดเวลาการสื่อสารข้อมูลผ่านตัวจำลองระบบ โดยกำหนดค่า Hz Option ระหว่าง 300 ถึง 5000 Hz ส่งข้อมูลชนิด ICMP ด้วยคำสั่ง Ping จำนวน 1,000 แพ็กเก็ตต่อการทดลอง แต่ละครั้งทำการปรับค่า Emulated Delay บนตัวจำลองระบบระหว่าง 10 ถึง 500 มิลลิวินาที กำหนดค่า Hz Option ให้เครื่องมือวัดมีค่าคงที่ คือ 5,000 Hz และวาดกราฟผลต่างเวลาจาก Emulated Delay

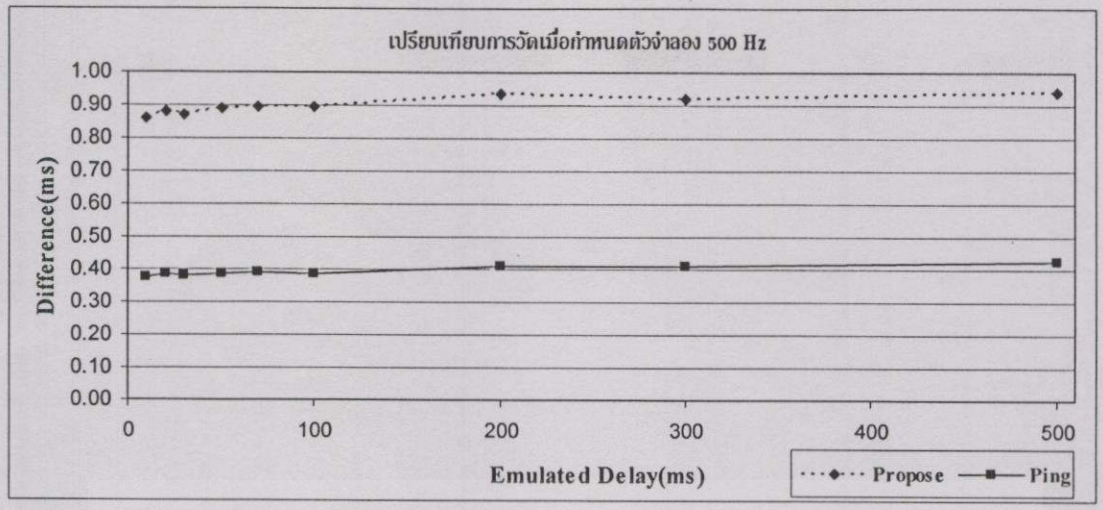


ภาพที่ 4.26 ผลการทดลองเมื่อกำหนดค่า Hz Option บนตัวจำลองระบบเป็น 100 Hz

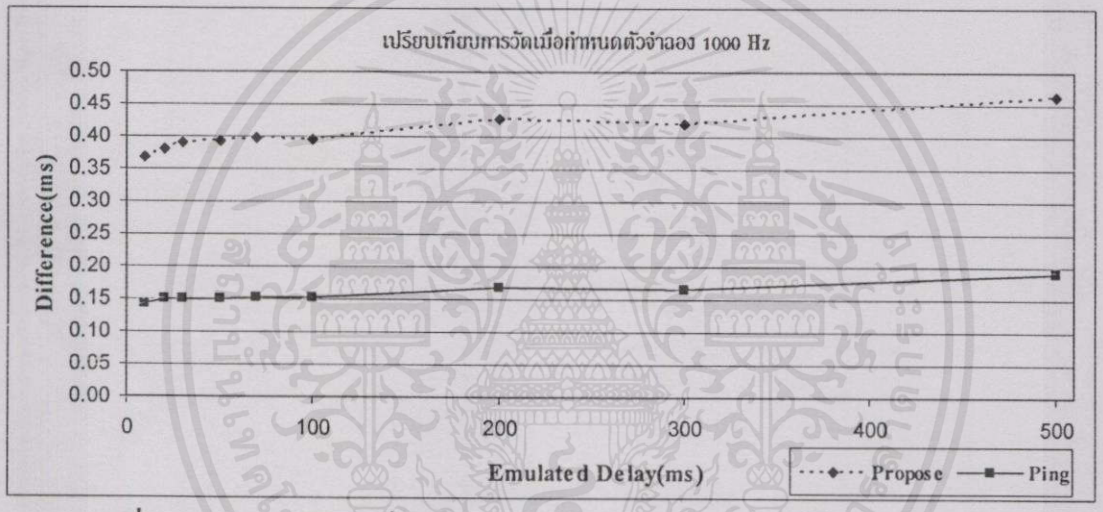


ภาพที่ 4.27 ผลการทดลองเมื่อกำหนดค่า Hz Option บนตัวจำลองระบบเป็น 300 Hz

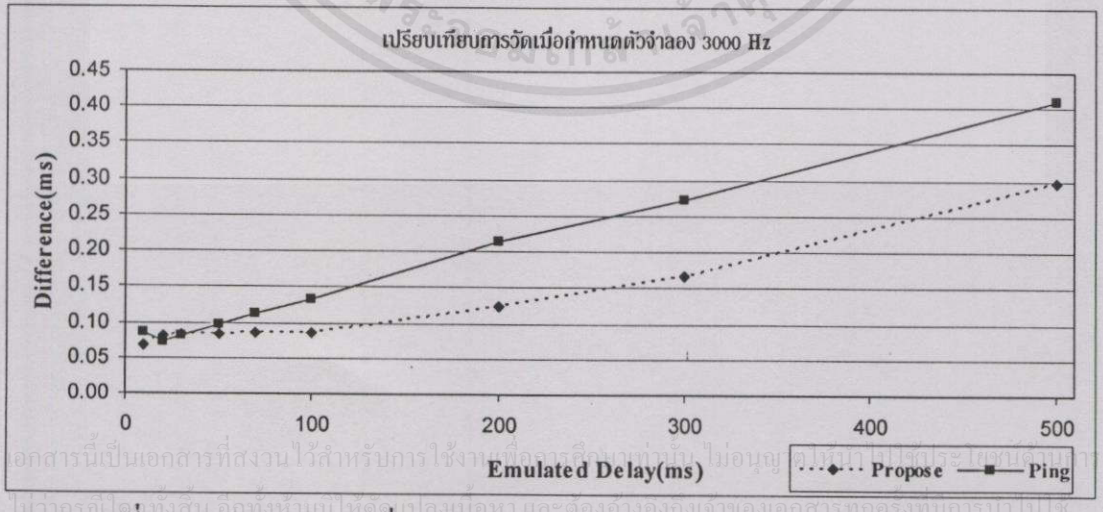
ไฟล์นี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.28 ผลการทดลองเมื่อกำหนดค่า Hz Option บนตัวจำลองระบบเป็น 500 Hz

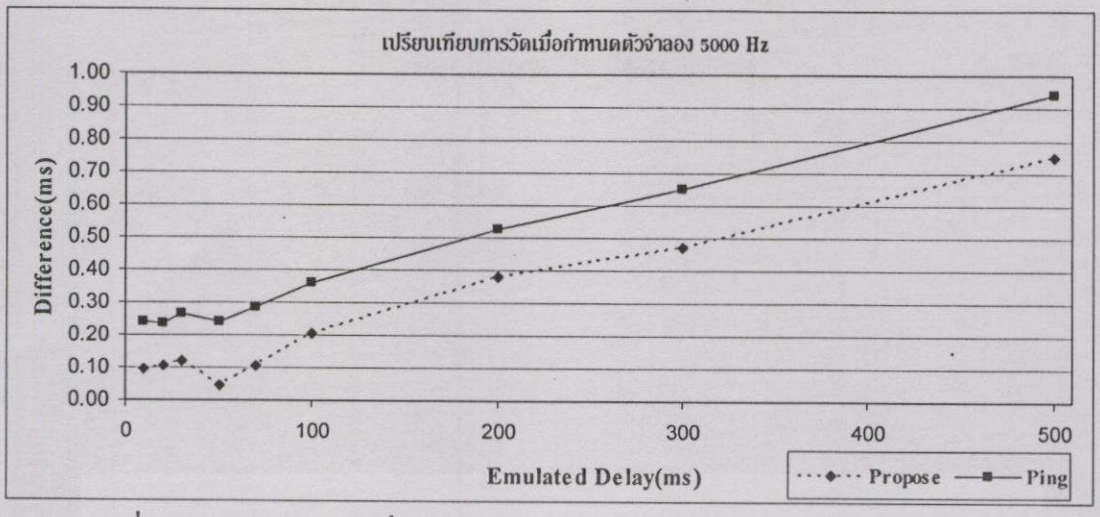


ภาพที่ 4.29 ผลการทดลองเมื่อกำหนดค่า Hz Option บนตัวจำลองระบบเป็น 1000 Hz

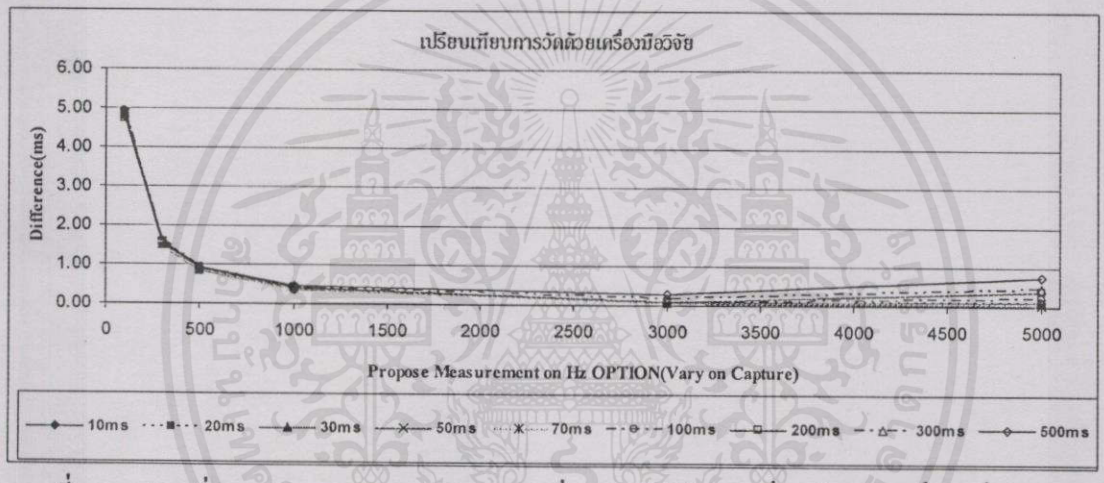


ภาพที่ 4.30 ผลการทดลองเมื่อกำหนดค่า Hz Option บนตัวจำลองระบบเป็น 3000 Hz

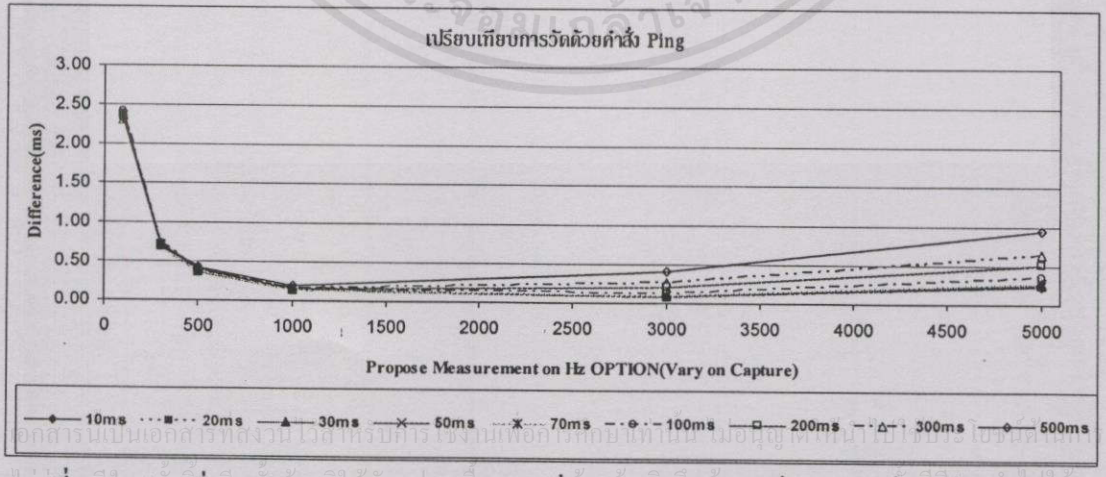
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำ ไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.31 ผลการทดลองเมื่อกำหนดค่า Hz Option บนตัวจำลองระบบเป็น 5000 Hz



ภาพที่ 4.32 ค่าเฉลี่ยผลต่างจาก Emulated Delay ที่ตรวจวัดได้จากเครื่องมือวิจัย เมื่อเปลี่ยนค่า Hz Option บนตัวจำลองระบบ



ภาพที่ 4.33 ค่าเฉลี่ยผลต่างจาก Emulated Delay ที่ตรวจวัดได้จากคำสั่ง Ping เมื่อเปลี่ยนค่า Hz Option บนตัวจำลองระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของลิขสิทธิ์ทุกครั้งที่มีการนำไปใช้

ในภาพที่ 4.26, 4.27, 4.28, 4.29, 4.30, 4.31, 4.32 และ 4.33 พบว่าพฤติกรรมการวัดค่าเวลาการสื่อสารข้อมูลผ่านตัวจำลองระบบ ผลการปรับเปลี่ยนค่า Hz Option ในระดับต่าง ๆ คุณลักษณะที่ได้คือ ผลต่างจะมีความคลาดเคลื่อนน้อยลงเมื่อมีการปรับค่า Hz Option บนตัวระบบเพิ่มขึ้น ซึ่งผลการวัดจากทั้งเครื่องมือวิจัย(Propose Tools) และคำสั่ง Ping สามารถวัดได้ไม่แตกต่างกัน

สรุปได้ว่าผลจากการทดลองปรับค่า Hz Option บนเครื่องมือวัด จะไม่ส่งผลทำให้การตรวจวัดค่าเวลาของหน่วยตรวจจับข้อมูลมีความแตกต่างกัน ดังแสดงได้ในตารางที่ 4.4 ที่ค่าเฉลี่ยความแตกต่างจาก Emulated Delay ในระดับ 10 มิลลิวินาทีตรวจวัดได้ใกล้เคียงกันมาก

ตารางที่ 4.4 ค่าเฉลี่ยของผลต่างจาก Emulated Delay ที่กำหนด Hz Option บนเครื่องมือวิจัย

เครื่องมือวัด	ค่าเฉลี่ยของผลต่างที่กำหนด Hz Option ในระดับต่าง ๆ(มิลลิวินาที)					
	100 Hz	300 Hz	500 Hz	1000 Hz	3000 Hz	5000 Hz
Propose	0.370642	0.369828	0.366798	0.375030	0.365254	0.367438
Ping	0.144052	0.142868	0.142881	0.144378	0.141908	0.142173

จากการค้นพบเพิ่มเติมเมื่อทดลองโดยปรับค่า Hz Option บนตัวจำลองระบบ พบว่าค่า Hz Option ที่สูงขึ้นจะมีผลทำให้ความคลาดเคลื่อนจาก Emulated Delay มีค่าน้อยลง หรือมีความแม่นยำเพิ่มขึ้น และจะต่ำสุดรวมไปถึงดีกว่าการตรวจวัดด้วยคำสั่ง Ping เมื่อกำหนดค่า Hz Option ในระดับ 3000 Hz ดังแสดงได้ในตารางที่ 4.5

ตารางที่ 4.5 ค่าเฉลี่ยของผลต่างจาก Emulated Delay ที่กำหนด Hz Option บนตัวจำลองระบบ

เครื่องมือวัด	ค่าเฉลี่ยของผลต่างที่กำหนด Hz Option ในระดับต่าง ๆ(มิลลิวินาที)					
	100 Hz	300 Hz	500 Hz	1000 Hz	3000 Hz	5000 Hz
Propose	4.862608	1.516744	0.861612	0.367438	0.067250	0.092764
Ping	2.364493	0.697754	0.376445	0.142173	0.086461	0.241043

สาเหตุเนื่องจากค่า Hz Option นั้น มีความสัมพันธ์กับระบบท่อรับส่งข้อมูล(Pipe) หรือเป็นกลไกการทำงานของระบบปฏิบัติการ FreeBSD ที่เรียกว่า DUMMYNET คือ เมื่อมีการกำหนดรอบความถี่สูงขึ้นจะทำให้เกิดผลตอบสนองต่อการรับส่งข้อมูลให้มีความราบรื่น(More Smooth)หรือมีความล่าช้าในการรับส่งข้อมูลน้อยลง ดังนั้นการสื่อสารข้อมูลจึงมีลักษณะที่มีความต่อเนื่องและลดเวลาสะสมที่เกิดจากการคอยในระบบลงไป[14] ดังนั้น Ping ซึ่งอ่านข้อมูลจาก Pipe จึงได้รับผลกระทบทำให้ค่าเวลาที่วัดได้มีการเปลี่ยนแปลงตามค่า Hz Option ด้วย ส่วนเครื่องมือวิจัยไม่มีผลกระทบเนื่องจากอ่านค่าเวลาข้อมูลโดยตรงจากอีเทอร์เนตกราดตามภาพที่ 4.30

4.4.2 การศึกษาผลกระทบของ Hz Option กับความเร็วในการวิเคราะห์ข้อมูล

หลักการตรวจวัดหน่วยแยกชนิดจะทำการจัดเก็บข้อมูลไว้บนหน่วยความจำที่สามารถใช้ร่วมกันได้(Share Memory) โดยจะกันพื้นที่ให้สามารถจัดเก็บข้อมูลก่อนการนำไปประมวลผลหรือนำไปวิเคราะห์ข้อมูลได้จำนวน 10,000 ระเบียนต่อตารางข้อมูล พื้นที่หน่วยความจำตามที่ออกแบบนี้ จะมีส่วนช่วยทำให้เครื่องมือวิจัยสามารถประมวลผลข้อมูลได้ครบถ้วน ในกรณีที่ข้อมูลสื่อสารมีอัตราการมาน้อยกว่าเวลาที่ หน่วยวิเคราะห์ข้อมูลต่าง ๆ สามารถประมวลผลหนึ่งหน่วยข้อมูลจนสมบูรณ์ ซึ่งนิยามพื้นที่ที่กำหนดนี้เรียกว่า "แถวคอย"[15] หรือนิยามการเกิดแถวคอยได้จากสมการที่ (4.6)

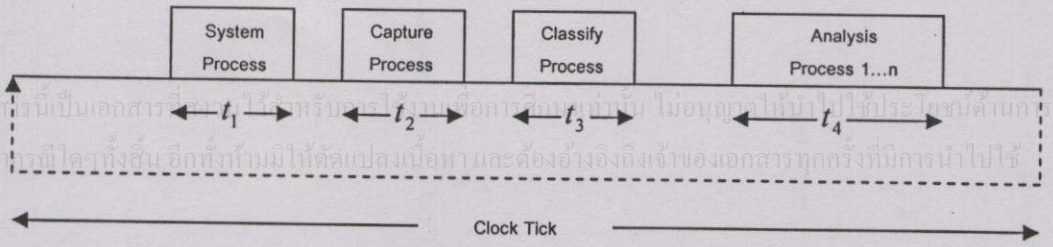
$$\text{Utilization} = \frac{\lambda}{\mu} \tag{4.6}$$

โดยที่ Utilization หมายถึง อัตราการให้บริการ กรณีคำนวณได้ไม่น้อยกว่า 1 จะเกิดแถวคอย

μ หมายถึง อัตราการให้บริการ(Service Rate) หรือเปรียบได้กับอัตราการวิเคราะห์ จำนวนข้อมูลได้ในหนึ่งหน่วยเวลา

λ หมายถึง อัตราการมา(Arrival Rate) หรือเปรียบได้กับอัตราการมาของข้อมูลต่อหนึ่งหน่วยเวลา

หลักการวิเคราะห์ข้อมูลด้วยรูปแบบนี้ ประสิทธิภาพของเครื่องมือจึงขึ้นอยู่กับคุณลักษณะการสื่อสารข้อมูลขององค์กรเป็นหลัก การสื่อสารที่มีปริมาณมากอาจส่งผลกระทบต่อความสามารถในการวิเคราะห์ข้อมูลของเครื่องมือ ที่มีความเร็วไม่เพียงพอต่อการมาของข้อมูลจริงบนเครือข่าย ซึ่งตามกลไกที่ออกแบบเครื่องมือวิจัยจะหยุดการทำงานลง เมื่อการวิเคราะห์ข้อมูลบนพื้นที่ที่กำหนดถูกนำไปใช้ จนกระทั่งไม่มีพื้นที่เหลือเพียงพอสำหรับเขียนทับ(Reuse Memory)ต่อไป และเมื่อพิจารณาขั้นตอนวิธีของเครื่องมือวิจัยจะพบว่าเครื่องมือวิจัยที่ออกแบบนั้น มีโครงสร้างที่เป็นไปตามกลไกของทฤษฎีแถวคอยเช่นเดียวกัน คือ มีหน่วยตรวจจับ(Capture Unit) และหน่วยแยกชนิดข้อมูล(Classify Unit)ทำหน้าที่จัดการกับแถวคอย และมีหน่วยวิเคราะห์ข้อมูล(Analysis Unit)ทำหน้าที่เป็นหน่วยบริการ ซึ่งแยกหน่วยประมวลผลย่อยตามจำนวนการตรวจวัดคุณลักษณะข้อมูลชนิดต่าง ๆ ซึ่งแต่ละหน่วยจะทำงานไปตามวัฏจักรต่อเนื่องกัน ดังภาพที่ 4.34



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 4.34 ขบวนการทำงานของเครื่องมือวิจัยบน ระบบปฏิบัติการ FreeBSD

โดยที่ Clock Tick หมายถึง รอบเวลาในการทำงานของทุกหน่วยประมวลผล ซึ่ง FreeBSD สามารถกำหนดได้ด้วย Hz Option

t_1 หมายถึง เวลาในการทำงานของหน่วยประมวลผลอื่น ๆ บนระบบปฏิบัติการ

t_2 หมายถึง เวลาในการทำงานของหน่วยตรวจจับข้อมูล

t_3 หมายถึง เวลาในการทำงานของหน่วยแยกชนิดข้อมูล

t_4 หมายถึง เวลาในการทำงานของหน่วยวิเคราะห์ข้อมูล ได้แก่ ความยาว เวลาระหว่างการมา เวลาการสื่อสารข้อมูลไปกลับ เวลาการสื่อสารข้อมูลผ่านตัวระบบและความสูญเสียข้อมูลบนตัวระบบ

จากภาพที่ 4.34 ทุกหน่วยประมวลผลจะทำงานโดยได้รับการจัดสรรทรัพยากรเวลาภายใต้การควบคุมของระบบปฏิบัติการ FreeBSD ซึ่งหากหน่วยบริการทำงานได้ช้ากว่าการมาของข้อมูล เนื้อที่ที่ถูกจัดเตรียมไว้ด้วยหน่วยแยกชนิดก็จะถูกนำไปใช้จัดเก็บข้อมูล และเกิดแถวคอยข้อมูลขึ้นเพื่อรอการประมวลผลจากหน่วยวิเคราะห์ข้อมูลต่อไป ดังนั้นประสิทธิภาพของเครื่องมือวัดจึงขึ้นอยู่กับความสามารถในการวิเคราะห์ข้อมูลเป็นหลัก ซึ่งสามารถพิจารณาหาความเร็วในการประมวลผลได้จากสมการที่ (4.7)

$$\mu = \frac{1}{t_4} \qquad (4.7)$$

โดยที่ t_4 หมายถึง เวลาที่หน่วยวิเคราะห์ประมวลผลข้อมูลต่อ 1 หน่วย

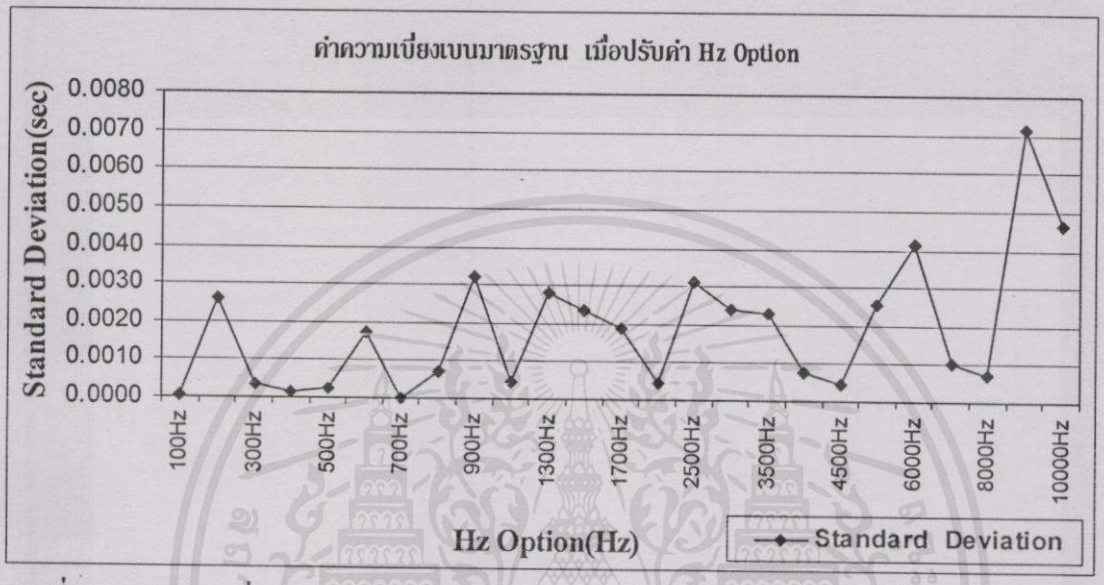
และด้วยกลไกสำคัญที่ค่าความถี่รอบเวลาเป็นตัวแปรสำคัญต่อการประมวลผลข้อมูลของเครื่องมือวิจัย ดังนั้นการกำหนดค่า Hz Option ให้กับเครื่องมือจึงอาจส่งผลทำให้ความเร็วในการวิเคราะห์ข้อมูลมีความแตกต่างกันไปด้วย

ค่า Hz Option ในระดับต่าง ๆ อาจส่งผลกระทบทำให้เกิดการปลดปล่อยทรัพยากรเวลาที่ไม่เหมาะสมคือ ปลดปล่อยทรัพยากรเวลาของหน่วยประมวลผลก่อนที่จะทำงานได้สำเร็จ ดังนั้นค่ารอบเวลาที่ถูกกำหนดนี้อาจกระทบไปถึงประสิทธิภาพในการวิเคราะห์ข้อมูลตามข้อสันนิษฐานคือ ค่ารอบเวลาอาจส่งผลให้ หน่วยประมวลผลหนึ่งมีเวลาไม่เพียงพอต่อการประมวลผลข้อมูล 1 หน่วย และต้องรอรอบการทำงานหรืออาจต้องทำงานหลายรอบเวลาเพื่อประมวลผลข้อมูล 1 หน่วยนั้นจนเสร็จสมบูรณ์ ดังนั้นการตั้งค่า Hz Option ในการตรวจวัดข้อมูลของเครื่องมือวิจัยจึงต้องตั้งค่าให้เหมาะสม และเพื่อที่จะทำให้เครื่องมือวิจัย สามารถประมวลผลข้อมูลได้อย่างรวดเร็วและต่อเนื่อง

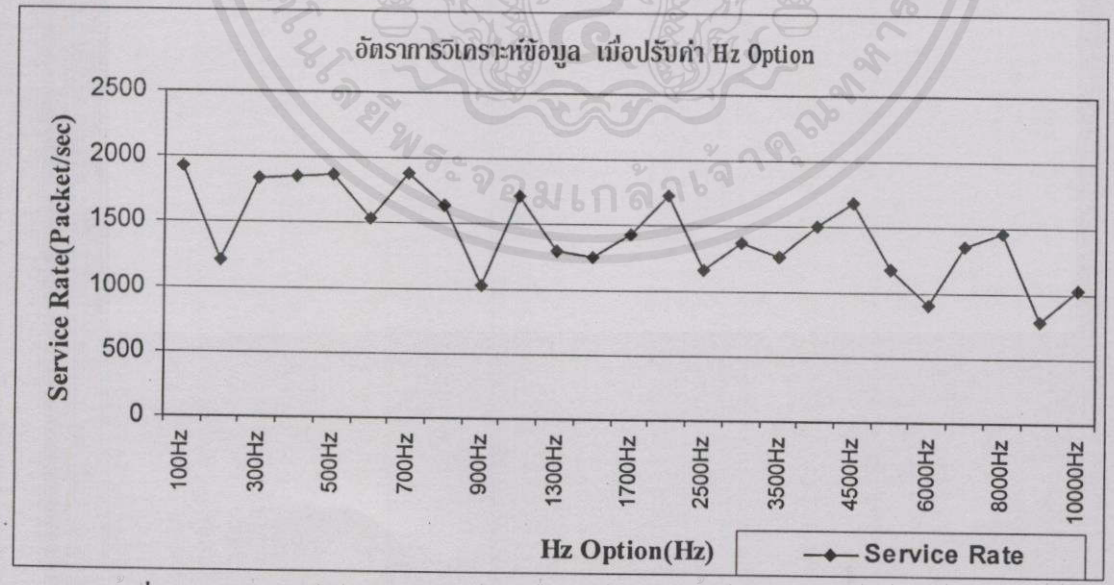
ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย
ห้ามมิให้คัดลอกหรือดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อแสดงพฤติกรรมการทำงานของเครื่องมือวัด จึงทำการทดลองโดยกำหนดค่า Hz Option บนเครื่องมือวิจัยอยู่ระหว่าง 100 ถึง 5000 Hz และกำหนดค่าHz Option บนตัวจำลอง

ระบบคงที่คือ 3000 Hz ทำการตรวจวัดเวลาที่ใช้ในการประมวลผลข้อมูลต่อ 1 หน่วย โดยส่งข้อมูลชนิด ICMP ด้วยคำสั่ง Ping จำนวน 1 แพ็กเก็ตต่อการทดลอง ทำซ้ำเป็นจำนวน 200 ครั้ง เพื่อหาค่าเฉลี่ยที่เป็นไปได้จาก การทดลองแต่ละครั้ง วิเคราะห์ผลการทดลองด้วยค่าสถิติต่าง ๆ ได้แก่ ค่าเฉลี่ย ค่าความแปรปรวน และค่าเบี่ยงเบนมาตรฐาน โดยวาดกราฟตามค่าเบี่ยงเบนมาตรฐาน และความเร็วในการวิเคราะห์



ภาพที่ 4.35 ค่าความเบี่ยงเบนมาตรฐานของเวลาประมวลผลข้อมูลต่อ 1 หน่วยเมื่อปรับค่า Hz Option



ภาพที่ 4.36 ผลการวิเคราะห์ความเร็วในการประมวลผลข้อมูลเมื่อปรับค่า Hz Option

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

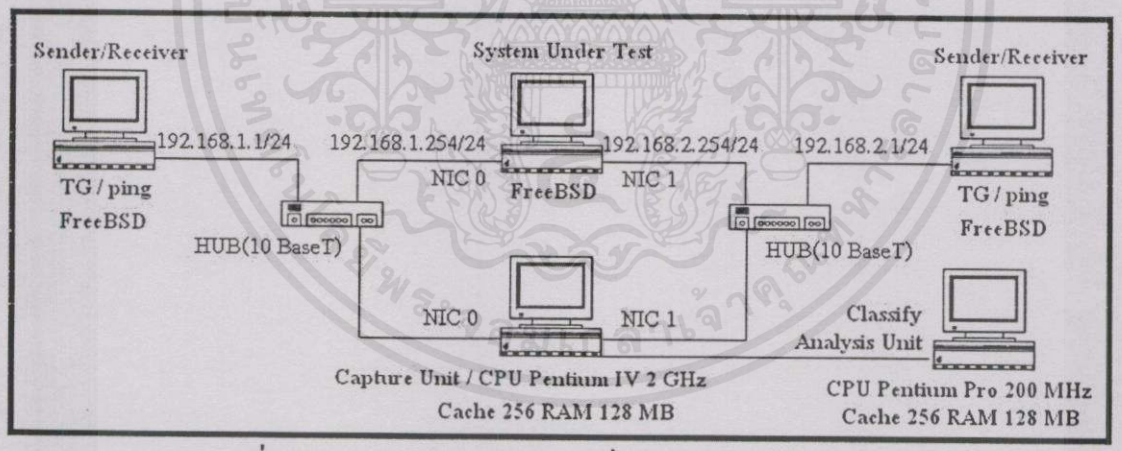
จากภาพที่ 4.35 ค่าความเบี่ยงเบนมาตรฐานของเวลาประมวลผล หรือความแปรปรวนจะมีค่าเพิ่มมากขึ้นตามค่า Hz Option ที่กำหนดให้กับเครื่องมือวิจัย ส่วนภาพที่ 4.36 ค่าเวลาเฉลี่ยในการ

ประมวลผลข้อมูลต่อ 1 หน่วยจะเพิ่มขึ้นตามค่า Hz Option ที่กำหนดให้กับเครื่องมือวิจัย และเมื่อคำนวณในรูปของความเร็วในการประมวลผล การกำหนดค่า Hz Option ในระดับสูง จะทำให้ประสิทธิภาพในการประมวลผลของหน่วยวิเคราะห์ข้อมูลต่ำลง โดยความเร็วสูงสุดอยู่ที่ 1,927 แพ็กเก็ตต่อวินาที เมื่อกำหนด Hz Option ในระดับ 100 Hz

สรุปได้ว่าในการเพิ่มค่า Hz Option ให้เครื่องมือวิจัยในระดับที่สูงขึ้น จะไม่ส่งผลทำให้การวิเคราะห์ข้อมูลเพิ่มขึ้นแต่อย่างใด เนื่องจากการกำหนดค่ารอบเวลาที่มีความถี่สูง จะทำให้เวลาในการวิเคราะห์ข้อมูลนั้นสั้นลง และทำให้ต้องปล่อยทรัพยากรเวลาไปก่อนที่จะทำงานได้เสร็จสมบูรณ์

4.4.3 การศึกษาความเร็วในการวิเคราะห์เมื่อแยกหน่วยตรวจจับข้อมูล

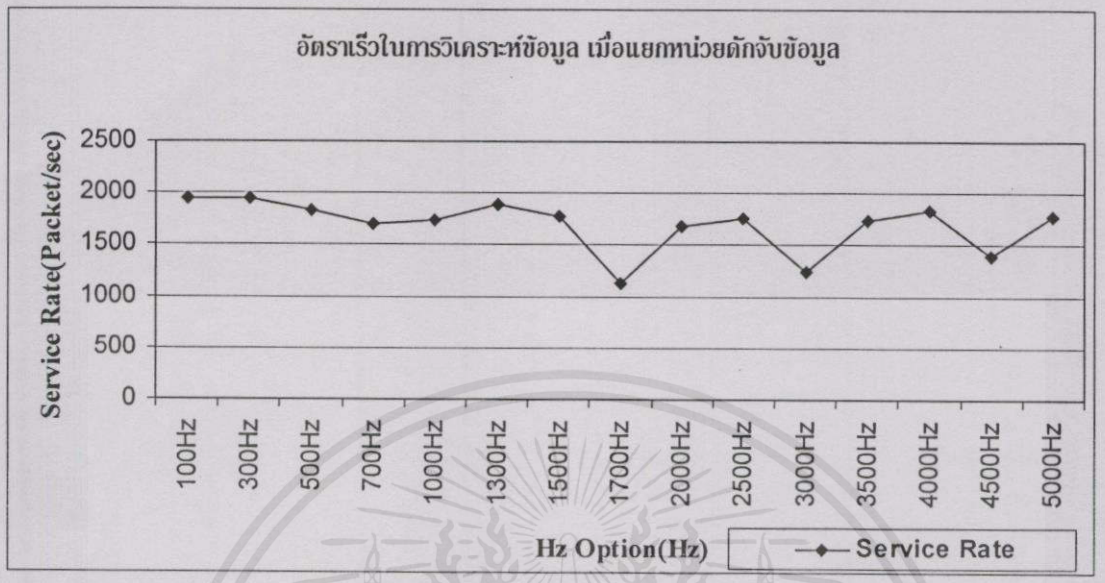
การทำงานแบบแบ่งเวลาประมวลผล ค่ารอบเวลาประมวลผลหนึ่ง ๆ อาจต้องใช้ไปกับการประมวลผลข้อมูลของแต่ละหน่วยเพิ่มขึ้น ดังนั้นการแยกหน่วยประมวลผลบางหน่วยออกอาจทำให้ความเร็วในการประมวลผลข้อมูลเพิ่มมากขึ้น เนื่องจากค่ารอบเวลาที่ต้องเสียไปกับหน่วยประมวลผลนั้นถูกกำจัดออกไป และจากการออกแบบเครื่องมือวิจัย การติดตั้งเครื่องมือเราสามารถแยกส่วนของหน่วยตรวจจับข้อมูลออกจากคอมพิวเตอร์เครื่องเดียวกันได้ โดยนำไปติดตั้งในคอมพิวเตอร์ให้ทำการส่งข้อมูลไปยังหน่วยแยกชนิดที่อยู่ต่างเครื่องกันได้ ดังภาพที่ 4.37



ภาพที่ 4.37 แบบจำลองการทดลองเมื่อแยกหน่วยตรวจจับข้อมูล

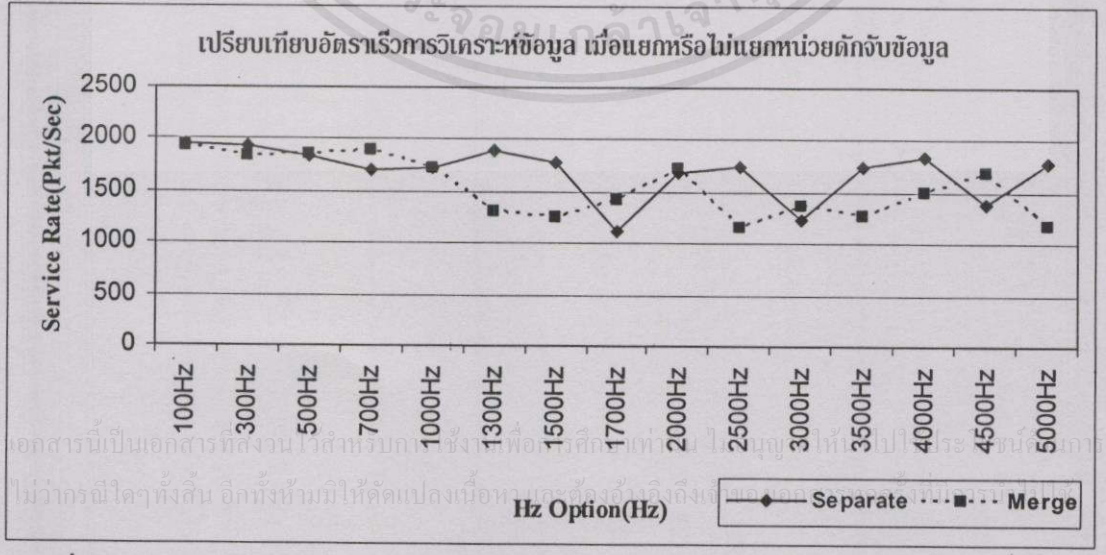
ด้วยลักษณะการทำงานตามวิธีการที่ออกแบบ จึงนิยามสมมติฐานไว้ดังนี้คือ การแยกหน่วยดักจับข้อมูลออกจากคอมพิวเตอร์เครื่องเดียวกัน จะมีผลทำให้อัตราเร็วในการวิเคราะห์ข้อมูลแตกต่างกัน เนื่องจากเวลาที่ใช้ไปกับการประมวลผลในส่วนของหน่วยตรวจจับข้อมูลลดลง ผลการตอบสนองในด้านเวลากับหน่วยประมวลผลข้อมูลที่เหลือ จึงอาจทำให้มีความแตกต่างไปจากวิธีการแบบที่ทำงานแบบครบทุกหน่วย ดังนั้นจึงทำการทดลองกำหนดค่า Hz Option บนเครื่องมือวิจัยอยู่ระหว่าง 100 ถึง 5000 Hz และกำหนดค่าHz Option บนตัวระบบ คงที่คือ 3000 Hz ทำการตรวจวัดเวลาที่ใช้ในการประมวลผลข้อมูลต่อ 1 หน่วย โดยส่งข้อมูลชนิด ICMP ด้วย

คำสั่ง Ping จำนวน 1 แพ็กเก็ตต่อการทดลอง ทำซ้ำเป็นจำนวน 200 ครั้ง เพื่อหาค่าเฉลี่ยที่เป็นไปได้จากการทดลองแต่ละครั้ง



ภาพที่ 4.38 ความเร็วในการวิเคราะห์ข้อมูลเมื่อแยกหน่วยตรวจจับข้อมูล

ในภาพที่ 4.38 ค่าเวลาเฉลี่ยในการประมวลผลข้อมูลต่อ 1 หน่วยยังคงเพิ่มขึ้นตามค่า Hz Option ที่กำหนดให้กับเครื่องมือวิจัย เมื่อคำนวณในรูปของความเร็วในการประมวลผล การกำหนดค่า Hz Option ในระดับสูง จะทำให้ประสิทธิภาพในการประมวลผลของหน่วยวิเคราะห์ข้อมูลยังคงต่ำลง เช่นเดียวกับการทดลองวัดในกรณีอยู่บนคอมพิวเตอร์เครื่องเดียวกัน ค่า Hz Option ที่ดีที่สุดที่จะกำหนดให้กับเครื่องมือวิจัยยังคงอยู่ในระดับ 100 Hz โดยวิเคราะห์ความเร็วเฉลี่ยสูงสุดได้ 1,952 แพ็กเก็ตต่อวินาที



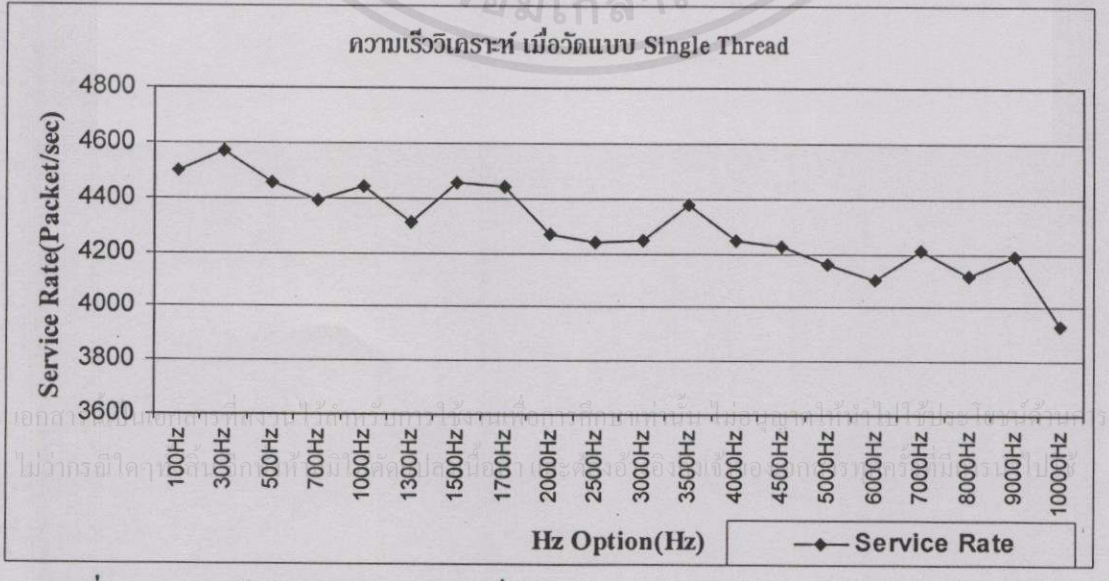
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 4.39 ความเร็วในวิเคราะห์ข้อมูลเปรียบเทียบระหว่างแยกและไม่แยกหน่วยตรวจจับข้อมูล

จากในภาพที่ 4.39 แสดงการเปรียบเทียบความเร็วในการวิเคราะห์ข้อมูลระหว่างการแยกและไม่แยกหน่วยตรวจจับข้อมูล สรุปได้ว่าการแยกหน่วยตรวจจับข้อมูลมีผลทำให้อัตราเร็วในการวิเคราะห์ข้อมูลดีขึ้นเล็กน้อย ซึ่งไม่มีนัยสำคัญเพียงพอที่จะสรุปได้ว่ามีความแตกต่างกัน ผลการเพิ่มค่า Hz Option ให้เครื่องมือวิจัยในระดับที่สูงขึ้น ยังคงมีลักษณะไม่แตกต่างจากการทดลองบนคอมพิวเตอร์เครื่องเดียวกัน คือค่า Hz Option ในระดับสูงไม่ส่งผลทำให้การวิเคราะห์ข้อมูลนั้นเพิ่มขึ้นแต่อย่างใด ค่า Hz Option ในระดับต่ำยังคงมีประสิทธิภาพในการวิเคราะห์ข้อมูลที่ดีกว่าในระดับสูง และมีค่าความเหมาะสมดีที่สุดอยู่ที่ระดับ 100 Hz

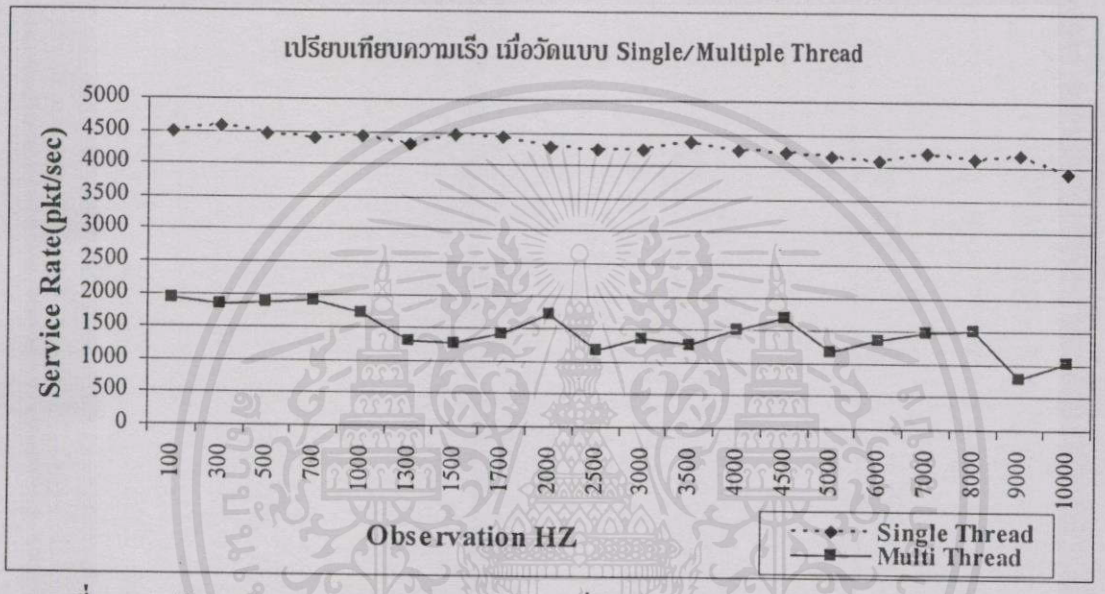
4.4.4 การศึกษาหน่วยวิเคราะห์ข้อมูลที่ทำงานในแบบ Single Thread

จากการทดลองเมื่อแยกหน่วยตรวจจับข้อมูลออกจากการตรวจวัดบนคอมพิวเตอร์เครื่องเดียวกัน ประสิทธิภาพในการวิเคราะห์ข้อมูลได้เพิ่มขึ้นเพียงเล็กน้อย ดังนั้นการลดหน่วยประมวลผลในวิเคราะห์ข้อมูลอาจทำให้ความเร็วในการวิเคราะห์ข้อมูลดีขึ้นได้ เนื่องจากความสูญเสียที่เกิดขึ้นจากขบวนการสลับเวลาประมวลผล(Overhead of Context Switching)[16]นั้นจะน้อยลง การนำหน่วยวิเคราะห์คุณลักษณะข้อมูลต่าง ๆ มารวมกันเพื่อลดความสูญเสียเวลาจากการสลับเวลา อาจทำให้อัตราการวิเคราะห์ข้อมูล ดีกว่าแบ่งหน่วยวิเคราะห์ข้อมูล(Multiple Thread) หรือทำให้เครื่องมือวิจัยสามารถประมวลผลข้อมูลต่อ 1 หน่วยได้รวดเร็วขึ้น โดยการทดลองปรับขั้นตอนวิธีในการวิเคราะห์ข้อมูลของเครื่องมือวิจัย ให้รวมหน่วยประมวลผลในการวิเคราะห์ข้อมูลเหลือเพียง 1 หน่วยประมวลผล กำหนดค่า Hz Option บนเครื่องมือวิจัยอยู่ระหว่าง 100 ถึง 5000 Hz และกำหนดค่า Hz Option บนตัวระบบ คงที่คือ 3000 Hz ทำการตรวจวัดเวลาที่ใช้ในการประมวลผลข้อมูลต่อ 1 หน่วย โดยส่งข้อมูลชนิด ICMP ด้วยคำสั่ง Ping จำนวน 1 แพ็กเก็ตต่อการทดลอง ทำซ้ำเป็นจำนวน 200 ครั้งเพื่อหาค่าเฉลี่ยที่เป็นไปได้จากการทดลองแต่ละครั้ง



ภาพที่ 4.40 ความเร็วในการประมวลผลเมื่อหน่วยวิเคราะห์ข้อมูลทำงานแบบ Single Thread

จากในภาพที่ 4.40 แสดงความเร็วในการประมวลผลเมื่อนำหน่วยวิเคราะห์ข้อมูลต่าง ๆ มารวมกันเพื่อให้ทำงานภายใต้หน่วยประมวลผลเดียว ผลการทดลองที่ได้คือค่าเวลาเฉลี่ยในการประมวลผลข้อมูลต่อ 1 หน่วยยังคงเพิ่มขึ้นตามค่า Hz Option ที่กำหนดให้กับเครื่องมือวิจัย เมื่อคำนวณในรูปของความเร็วในการประมวลผล การกำหนดค่า Hz Option ในระดับสูง จะทำให้ประสิทธิภาพในการประมวลผลของหน่วยวิเคราะห์ข้อมูลยังคงต่ำลง อัตราในการวิเคราะห์ข้อมูลสูงสุดที่เครื่องมือวิจัยทำได้คือ 4,568 แพ็กเก็ตต่อวินาที ซึ่งเมื่อเทียบกับการทดลองที่มีการแบ่งหน่วยวิเคราะห์ข้อมูลแล้วทำได้เพียง 1,927 แพ็กเก็ตต่อวินาที



ภาพที่ 4.41 เปรียบเทียบอัตราการวิเคราะห์ข้อมูลที่ทำงานในแบบ Single และ Multiple Thread

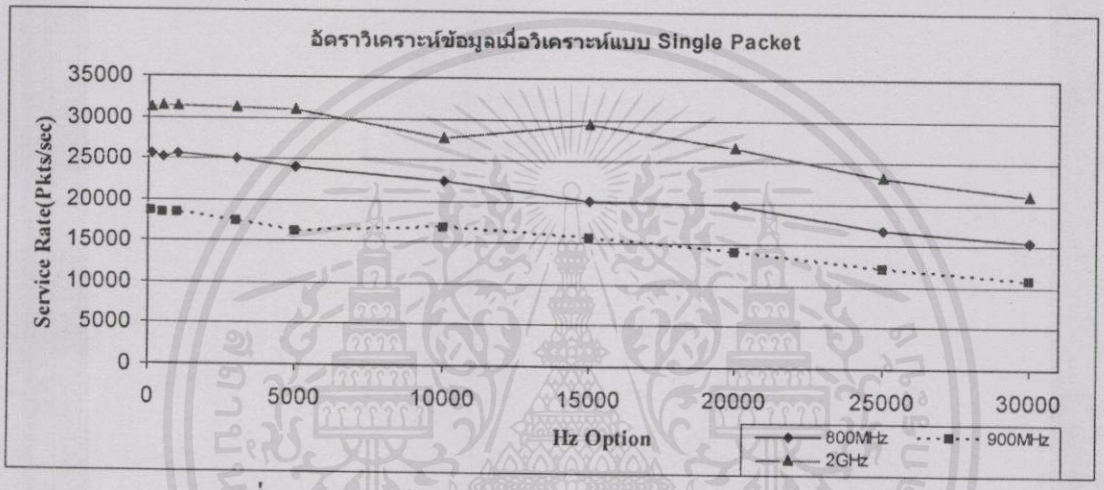
ในภาพที่ 4.41 ผลจากการทดลองสรุปได้ว่าการออกแบบขั้นตอนวิธีในการวิเคราะห์ข้อมูลแบบหน่วยเดียว จะช่วยทำให้อัตราการวิเคราะห์นั้นดีขึ้นกว่าเดิมอย่างเห็นได้ชัด ส่วนผลของการเพิ่มค่า Hz Option ให้กับเครื่องมือ ค่าอัตราในการวิเคราะห์ข้อมูลยังคงให้ผลลัพธ์ที่ไม่แตกต่างจากเดิม คือ การเพิ่มค่า Hz Option ไม่ส่งผลทำให้อัตราการวิเคราะห์ข้อมูลเพิ่มขึ้น เหตุผลยังคงเป็นลักษณะเดียวกัน คือ ค่ารอบเวลาที่สั้นลงความสูญเสียที่เกิดจากการสลับค่าเวลาจะเพิ่มขึ้น และมีลักษณะเป็นแบบสะสม(accumulative) รวมถึงในทุก 100 ms ที่ระบบปฏิบัติการต้องสลับค่าเวลาให้กับ kernel mode ด้วย ดังนั้นผลกระทบเมื่อคำนวณเป็นค่าเฉลี่ยเวลาจึงทำให้อัตราการวิเคราะห์ที่มีค่า Hz Option ในระดับสูงมีค่าต่ำกว่าอัตราการวิเคราะห์ที่มี Hz Option ในระดับต่ำ

4.4.5 การศึกษาความเร็วในการวิเคราะห์ข้อมูลด้วยหน่วยประมวลกลางชนิดต่าง ๆ

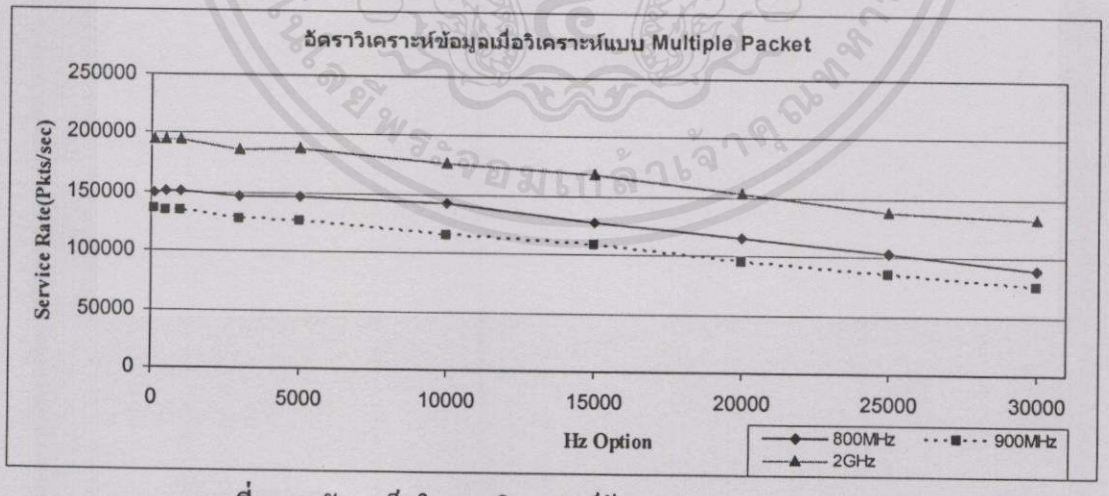
ปัจจุบันคอมพิวเตอร์ส่วนบุคคลได้พัฒนาให้หน่วยประมวลผลกลาง มีความเร็วเพิ่มขึ้น ส่งผลทำให้คอมพิวเตอร์สามารถประมวลผลงานต่าง ๆ ให้เกิดความรวดเร็วตามไปด้วย ดังนั้นการติดตั้งเครื่องมือวิจัยบนคอมพิวเตอร์ที่มีหน่วยประมวลผลกลางที่เร็วขึ้น อาจจะส่งผลทำให้

ความเร็วในการวิเคราะห์ข้อมูลดีขึ้นตามไปด้วย โดยการทดลองปรับค่า Hz Option บนเครื่องมือวิจัยอยู่ระหว่าง 100 ถึง 30000 Hz ทำการตรวจวัดเวลาที่ใช้ในการประมวลผลข้อมูลทั้งแบบ 1 การทดลองต่อ 1 แพ็กเก็ต(Single Pakcet) และแบบ 1 การทดลองต่อ 10000 แพ็กเก็ต(Multiple Packet) และวาดกราฟจากค่าเฉลี่ยในประมวลผลข้อมูลต่อ 1 แพ็กเก็ต โดยทำการทดลองกับคอมพิวเตอร์ส่วนบุคคลที่มีคุณสมบัติดังนี้

1. CPU Pentium III ความเร็ว 800 MHz Cache 256 KB และ RAM 256 MB
2. CPU Celeron 900MHz Cache 256 KB และ Ram 256 MB
3. CPU Pentium IV ความเร็ว 2 GHz Cache 1024 KB และ RAM 512 MB

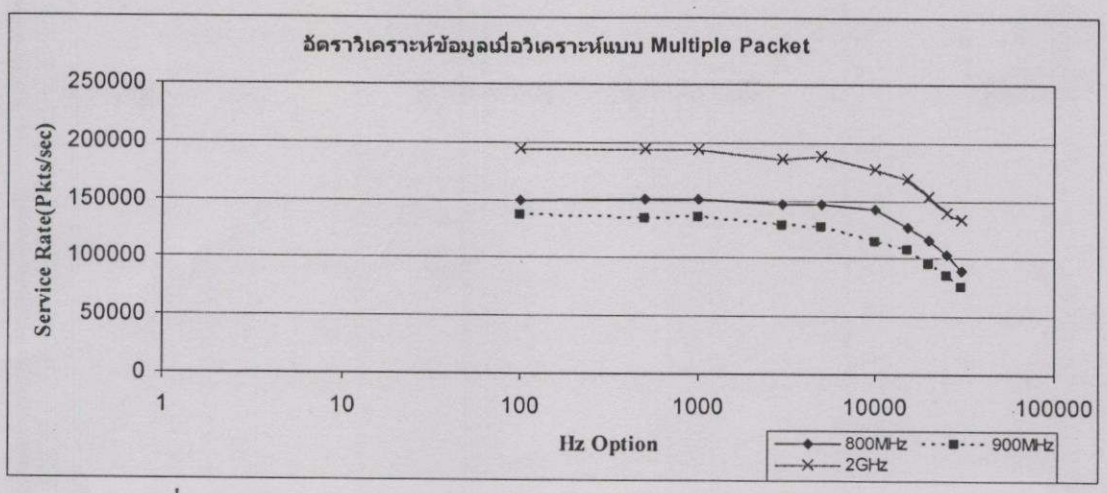


ภาพที่ 4.42 อัตราเร็วในการวิเคราะห์ข้อมูลแบบ Single Packets



ภาพที่ 4.43 อัตราเร็วในการวิเคราะห์ข้อมูลแบบ Multiple Packets

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.44 อัตราเร็วในการวิเคราะห์แบบ Multiple Packets (Logarithm Scale)

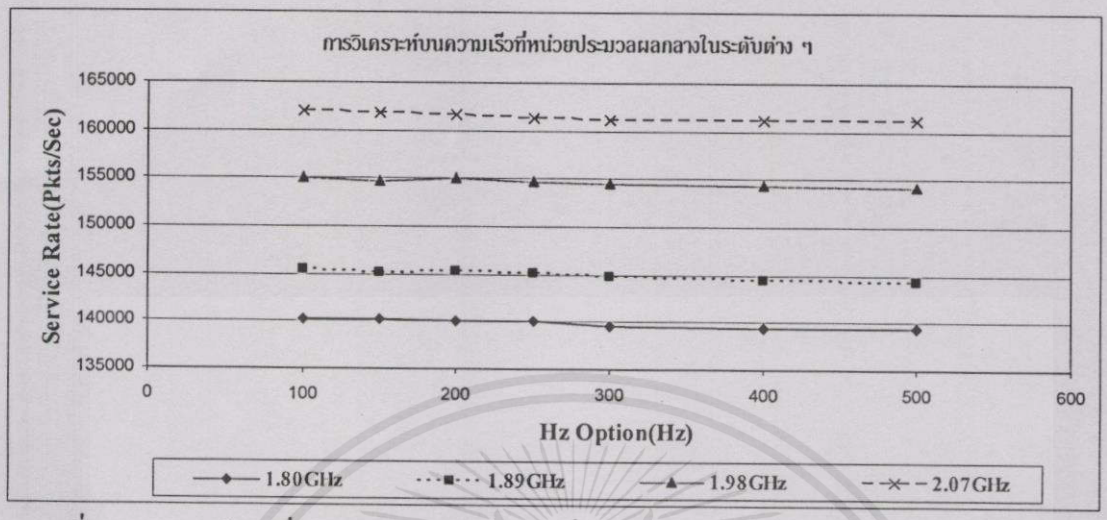
จากภาพที่ 4.42, 4.43, และ 4.44 ค่า Hz Option ที่ระดับ 100 Hz จะทำให้คอมพิวเตอร์ในทุกเครื่องสามารถวิเคราะห์ข้อมูลได้ในอัตราเร็วสูงสุด การเพิ่มค่า Hz Option จะมีผลทำให้ประสิทธิภาพในการวิเคราะห์ข้อมูลต่ำลง ซึ่งเป็นปัญหาจากขบวนการสลับเวลาของหน่วยประมวลผล การวิเคราะห์แบบ Multiple Packet พบว่า Threshold ของอัตราเร็วในการวิเคราะห์ข้อมูลมีจุดเปลี่ยนที่ใกล้เคียงกันคือ ที่ระดับ 10000 Hz

ดังนั้นจึงสรุปได้ว่า Hz Option ไม่ควรตั้งค่าเกิน 10000 Hz เนื่องจากเป็นจุดที่จะทำให้ประสิทธิภาพของเครื่องมือมีอัตราวิเคราะห์ต่ำลงอย่างรวดเร็ว และจากการทดลองความเร็วสูงสุดในการวิเคราะห์ข้อมูลโดยคอมพิวเตอร์ CPU ชนิด Celeron 900 MHz จะมีค่าต่ำกว่า Pentium III 800 MHz ดังนั้นค่า Hz ของ CPU ที่ต่างชนิดกันจะไม่สะท้อนถึงประสิทธิภาพของความเร็วในการวิเคราะห์ข้อมูลของเครื่องมือที่ออกแบบเพิ่มขึ้น เนื่องจาก CPU ต่างชนิดกันอาจมีระบบการเข้าถึงข้อมูลภายในที่แตกต่างกันด้วย เช่น ขนาดของระบบ BUS , Cache หรือ RAM ในขณะที่ผลการเปรียบเทียบบน CPU ชนิดเดียวกันคือ Pentium IV จะแสดงได้ว่าการเพิ่มความเร็ว CPU จะมีผลทำให้ความเร็วในการวิเคราะห์ข้อมูลเพิ่มขึ้น

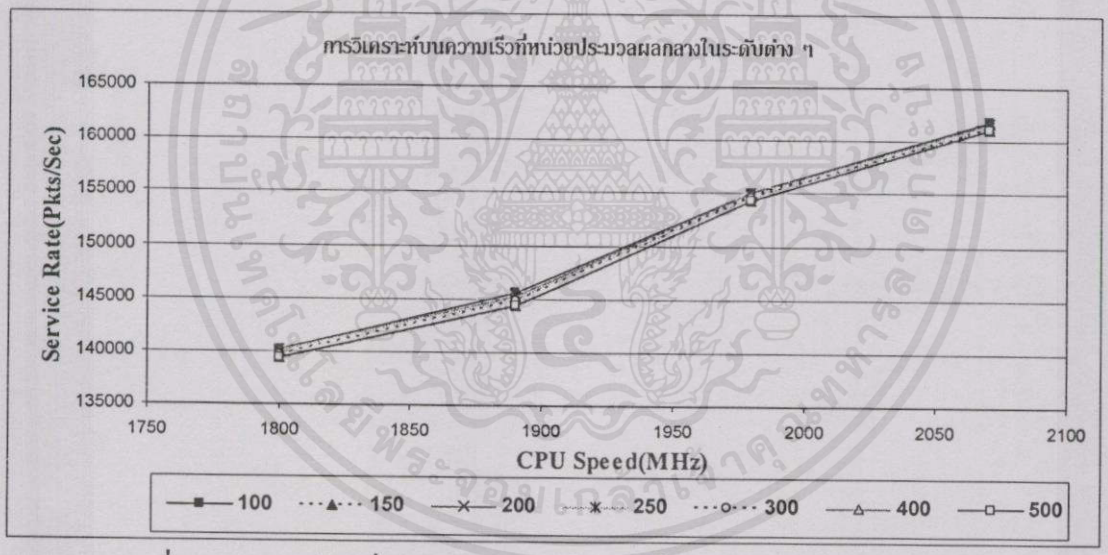
4.4.6 การศึกษาความเร็วในการวิเคราะห์ข้อมูลด้วยหน่วยประมวลกลางชนิดเดียวกัน

การตั้งค่า Hz Option ต่อเครื่องมือวัดในระดับต่าง ๆ จะมีผลทำให้ความเร็วในการวิเคราะห์ข้อมูลของเครื่องมือมีความแตกต่างกัน เนื่องจากความสูญเสียที่เกิดจากการสลับเวลาทำงานของหน่วยประมวลผลอื่น ๆ ภายในระบบ ดังนั้นเพื่อศึกษาการตั้งค่าที่เหมาะสมจึงออกแบบการทดลองโดยติดตั้งเครื่องมือบนคอมพิวเตอร์ส่วนบุคคลชนิด Pentium IV ความเร็ว CPU 1.8 GHz Cache 256 KB และ RAM 64 MB และทำการควบคุมตัวแปรอื่น ๆ ที่เกี่ยวข้อง ได้แก่ BUS, Cache, RAM และอุปกรณ์พ่วงต่ออื่น ๆ ให้คงที่ โดยตั้งค่าเฉพาะความเร็วของ CPU ให้แตกต่างกันเท่านั้นคือ 1.8 GHz, 1.89 GHz, 1.98GHz และ 2.07 GHz ทำการทดลองวิเคราะห์

ความเร็วเฉลี่ยต่อการประมวลผลข้อมูล 1 แพ็กเก็ตจากจำนวน 2,000 แพ็กเก็ตต่อการทดลอง โดยตั้งค่า Hz Option ระหว่าง 100 ถึง 500 Hz



ภาพที่ 4.45 อัตราเร็วเฉลี่ยในการวิเคราะห์ข้อมูลที่ Hz Option ในระดับต่าง ๆ บน CPU เดียวกัน



ภาพที่ 4.46 อัตราเร็วเฉลี่ยในการวิเคราะห์ข้อมูลที่ความเร็ว CPU ในระดับต่าง ๆ

จากภาพที่ 4.45 และ 4.46 พบว่าการตั้งค่า Hz Option ที่สูงขึ้นจะมีผลทำให้ค่าเฉลี่ยเวลาในการวิเคราะห์ข้อมูลเพิ่มขึ้นและทำให้อัตราเร็วเฉลี่ยในการวิเคราะห์ข้อมูลต่ำลง ซึ่งเป็นผลจากความสูญเสียในการสลับเวลาทำงานของหน่วยประมวลผลข้อมูลต่าง ๆ ภายในระบบ ดังนั้นการตั้งค่าที่เหมาะสมจึงอยู่ที่ระดับ 100 Hz หรือมีค่ารอบเวลาต่อหน่วยประมวลผลเป็น 10 มิลลิวินาทีต่อรอบทำงาน และนอกจากนี้การเพิ่มค่าความเร็วของ CPU จะมีผลทำให้อัตราเร็วเฉลี่ยในการวิเคราะห์ข้อมูลนั้นเพิ่มขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น นอกจากนี้ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการวิจัย และข้อเสนอแนะ

ความต้องการใช้เครื่องมือในการตรวจวัดคุณลักษณะข้อมูลเครือข่าย เป็นปัจจัยสำคัญสำหรับผู้ดูแลหรือออกแบบเครือข่าย เทคนิคในการวิเคราะห์ข้อมูลจึงเป็นความสำคัญที่จะช่วยทำให้ผลการวิเคราะห์ข้อมูลนั้น สามารถนำไปใช้ประเมินประสิทธิภาพของระบบได้อย่างเหมาะสม สามารถประหยัดค่าใช้จ่ายจากการออกแบบอุปกรณ์ หรือสัญญาณการสื่อสารข้อมูลที่เกินความจำเป็นของหน่วยงานได้ การพัฒนาเครื่องมือด้วยเทคนิคการตรวจวัดข้อมูลแบบพาสซีฟจะช่วยทำให้การตรวจวัดข้อมูลสามารถศึกษาพฤติกรรมการสื่อสารข้อมูลที่มีอยู่จริงบนเครือข่ายได้ ซึ่งในวิทยานิพนธ์นี้ ได้นำเสนอการออกแบบวิธีตรวจวัดคุณลักษณะการสื่อสารข้อมูลบนเครือข่ายอินเตอร์เน็ตด้วยคอมพิวเตอร์ส่วนบุคคล โดยใช้เทคนิคแบบพาสซีฟตรวจวัดค่าพฤติกรรมการสื่อสารข้อมูลต่าง ๆ ได้แก่ ความยาว เวลาระหว่างการมา เวลาการสื่อสารข้อมูลไปกลับ เวลาการสื่อสารข้อมูลผ่านตัวระบบ และความสูญเสียข้อมูลบนตัวระบบ ผลการวิจัยสามารถสรุปได้ดังนี้

1. การออกแบบขั้นตอนวิธีในการตรวจวัดข้อมูล สามารถนำไปใช้ในการพัฒนาเป็นเครื่องมือเพื่อตรวจวัดข้อมูลเครือข่ายโดยใช้คอมพิวเตอร์ส่วนบุคคลได้จริง

2. การวิเคราะห์ความถูกต้องเมื่อเทียบกับเครื่องมือวัดอื่น เครื่องมือวิจัยสามารถตรวจวัดได้ไม่แตกต่างกัน โดยวิเคราะห์ได้จากค่าเปอร์เซ็นต์ความแตกต่างที่มีค่าเฉลี่ยวัดได้ไม่เกิน 0.05 เปอร์เซ็นต์ ค่าสัมประสิทธิ์สหสัมพันธ์วัดได้ใกล้เคียงกับ 1.00 หรือสามารถตรวจวัดได้โดยมีค่าที่ใกล้เคียงกันมาก

3. การศึกษาประสิทธิภาพในการวิเคราะห์ข้อมูลพบว่า การปรับเปลี่ยนค่า Hz Option บนเครื่องมือวัดไม่มีผลต่อความถูกต้องในการตรวจวัดค่าเวลา แต่จากการค้นพบเพิ่มเติม Hz Option บนตัวจำลองระบบจะมีผลทำให้ค่าเวลาที่ตรวจวัดได้จากเครื่องมือวิจัย มีความคลาดเคลื่อนน้อยลง และดีที่สุดเมื่อกำหนด Hz Option ในระดับ 3000 Hz

4. การปรับแต่งประสิทธิภาพเครื่องมือในการวิเคราะห์ข้อมูล การออกแบบขั้นตอนวิธีในการวิเคราะห์ข้อมูลเป็นแบบหน่วยเดียว(Single Thread)จะช่วยทำให้อัตราเร็วการทำงานดีกว่าแบบแบ่งหน่วยวิเคราะห์ข้อมูลแบบหลายหน่วย(Multiple Thread) โดยสามารถวิเคราะห์ความเร็วสูงสุดในการติดตั้งเครื่องมือบนคอมพิวเตอร์ CPU ชนิด Pentium Pro 200 MHz Cache 256 KB และ RAM 128 MB ได้เฉลี่ย 4,568 แพ็กเก็ตต่อวินาที ที่ Hz Option ระดับ 100 Hz การปรับค่า Hz Option ที่สูงขึ้นจะมีผลทำให้เกิดการสูญเสียเวลาจากขบวนการสลับเวลาประมวลผล (Overhead of Context Switching)เพิ่มมากขึ้น และมีผลทำให้อัตราเร็วในการวิเคราะห์ข้อมูลนั้นต่ำลง เมื่อติดตั้งบนคอมพิวเตอร์ส่วนบุคคลที่มี CPU สูงขึ้นพบว่าการติดตั้งเครื่องมือในแต่ละ

เครื่องนั้น ต้องไม่กำหนดค่า Hz Option ให้สูงไปกว่า 10,000 Hz เนื่องจากเป็นจุดที่ทำให้อัตราเร็วในการวิเคราะห์ข้อมูลของเครื่องมือนั้นมีประสิทธิภาพต่ำลงอย่างรวดเร็ว และนอกจากนี้ความเร็วของหน่วยประมวลกลางก็ไม่ใช่ปัจจัยสำคัญเพียงประการเดียวที่จะทำให้อัตราเร็วในการวิเคราะห์ข้อมูลนั้นเพิ่มขึ้น ซึ่งต้องพิจารณาปัจจัยด้านอื่น ๆ รวมไปด้วย ได้แก่ ขนาดของ BUS, Cache และ RAM เป็นต้น

วิธีการที่นำเสนอในวิทยานิพนธ์เป็นเทคนิคหนึ่งเท่านั้นที่ช่วยในการตรวจวัดคุณลักษณะข้อมูลด้วยคอมพิวเตอร์ส่วนบุคคลโดยใช้เทคนิคแบบพาสซีฟได้ การศึกษาปัจจัยในด้านความถูกต้องของการตรวจวัดค่าเวลา และความเร็วในการวิเคราะห์ข้อมูลมีผลเพียงทำให้คอมพิวเตอร์ส่วนบุคคลนั้นสามารถทำงานเพื่อตรวจวัดคุณลักษณะข้อมูลได้ในระดับหนึ่ง แต่ก็ยังมีคุณลักษณะข้อมูลอื่นที่น่าสนใจ และสามารถทำให้เพิ่มประสิทธิภาพในทำงานเพิ่มขึ้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

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

[2] W. McRobb Daniel, **cflowd: Traffic flow analysis tool**, [Online] Available : http://www.caida.org/tools/ measurement/cflowd/.

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

[4] 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, RIPE NCC, [Online] Avalilable : http://www.caida.org/outreach/papers/pam2001/coralreef.xml, Apr, 2001.

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

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

[7] Philip Miller, **TCP/IP Explained**, Boston Oxford Jahannesburg, Melbourne, New Delhi, Singapore, 1997.

[8] S. McCanne and V. Jacobson, **The BSD Packet Filter: A New Architecture for User-level Packet Capture**. Proceedings of the 1993 Winter USENIX Technical Conference, San Diego, CA, Jan. 1993.

[9] Martin , **Packet Capture With libpcap and other Low Level Network Tricks**, Computer Enginearing, Northern Arizona University, [Online] Available: http://www.cet.nau.edu/~mc8/Socket/Tutorials/

[10] Ken Keys, David Moore, Ryan Koga, Edouard Lagache, Michael Tesch, and k cla_y, **The Architecture of CoralReef: An Internet Tra_c Monitoring Software Suite**, [Online] Available : http://www.caida.org/outreach/papers/itf.html, Nov, 1993.

[11] Brynjar Age Viken, **Passive Monitoring of Internet Traffic of SuperComputing'98** , Proceedings of EUNICE '99 ,Barcelona, Spain, Sept. 1999.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

[13] Randy Pratt, **FreeBSD Free Unix Operating System**, [Online] Available: http://www.treefort.org/~rpratt/freebsd, 2002.

[14] Alexandru Popa, **Using Dummynet for Traffic Shaping on FreeBSD**,[Online] Available: http://www.freebsd.org/, September 7,2003.

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

[16] George Neville-Neil, Marshall Kirk McKusick, **FreeBSD Process Management**, [Online] Available: http://www.awprofessional.com/ ,Addison-Wesley Professional, Boston, New York, Mar 4, 2005.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

โปรแกรม CoralLib ที่ใช้เปรียบเทียบในการตรวจวัดความยาว และเวลาระหว่างการมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#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;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
++(*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);
```

เอกสารนี้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
      }
    }
    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();
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

Configruration File ที่ใช้ในแบบจำลองการทดลองตรวจวัดข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แฟ้มติดตั้งข้อกำหนดในการตรวจวัดข้อมูลฝั่งขากลับ

```
# 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
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แฟ้มติดตั้งข้อกำหนดในการตรวจวัดข้อมูลฝั่งขาไป

```
# 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
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แฟ้มติดตั้งเครื่องมือตรวจวัด

```
BUFFER 10000
#Window Search
WINDOW 100
#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
```

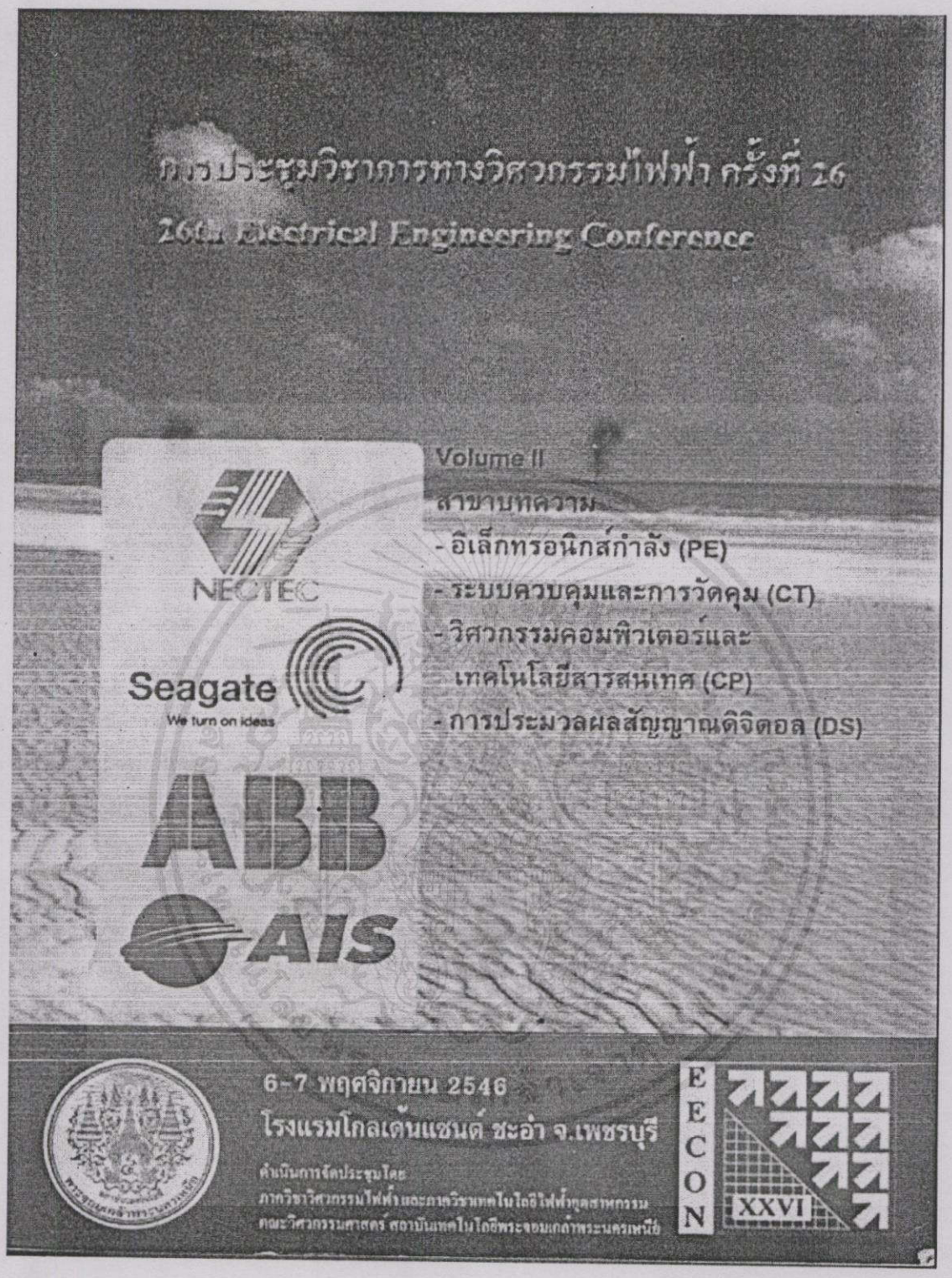
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

ผลงานวิจัยที่ได้รับการตีพิมพ์เผยแพร่

1. แสงเพ็ชร พระฉาย และอัครินทร์ คุณกิตติ, "การตรวจวัดคุณลักษณะการสื่อสารข้อมูลไปกลับบนเครือข่ายอินเทอร์เน็ต," การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 26, หน้า 958-963, เพชรบุรี, ประเทศไทย, 6-7 พฤศจิกายน, 2546.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตรวจวัดคุณลักษณะการสื่อสารข้อมูลไปกลับบนเครือข่ายอินเทอร์เน็ต

Measurement of Internet Packet Response Time Characteristics.

แสงเพ็ชร พระฉาย และอัศรินทร์ คุณกิตติ
คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ลาดกระบัง กรุงเทพฯ 10520
โทร (02)737-2551-4 โทรสาร (02)326-9074 E-Mail : mprachai@hotmail.com

บทคัดย่อ

บทความนี้นำเสนอกรอบและแนวคิดในการออกแบบเครื่องมือที่ใช้ตรวจวัดคุณลักษณะการสื่อสารข้อมูลไปกลับบนเครือข่ายอินเทอร์เน็ต ด้วยวิธีการที่ไม่ใช้การรับส่งข้อมูลระหว่างคู่ของการสื่อสาร ลักษณะการตรวจวัดใช้วิธีตรวจจับและกรองเฉพาะข้อมูลส่วนหัวของโพรโทคอลเวอร์ชั่นที่ 4 จากทั้งเครือข่ายภายในและภายนอก นำไปแบ่งแยกและจัดเก็บเป็นตารางข้อมูลที่เหมือนกัน 3 ชนิด คือ ตารางข้อมูลที่มีทิศทางการสื่อสารขาออก ตารางข้อมูลที่มีทิศทางการสื่อสารขาเข้า และตารางข้อมูลอื่น ๆ ตารางข้อมูลที่รวบรวมได้จะนำไปใช้กับการวิเคราะห์เวลาการสื่อสารไปกลับของข้อมูลชนิด ICMP และ TCP โดยพิจารณาคู่แพ็กเก็ตตอบสนองการสื่อสารกัน จากตารางข้อมูลที่มีทิศทางการสื่อสารตรงกันข้ามและอยู่ในฝั่งเครือข่ายเดียวกัน ทำการทดลองพัฒนาเครื่องมือตรวจวัดเวลาการสื่อสารข้อมูลไปกลับและนำไปวิเคราะห์ความถูกต้องด้วยการหน่วงเวลาระหว่างคู่การสื่อสาร ผลลัพธ์ที่ได้เปรียบเทียบกับการตรวจวัดข้อมูลชนิด ICMP ด้วย ping และเปรียบเทียบข้อมูลชนิด TCP ระหว่างเปิดและปิดการเชื่อมต่อ นอกจากนี้ได้นำเสนอความถูกต้องของการตรวจวัดความยาว และเวลาระหว่างการมาของแพ็กเก็ต เปรียบเทียบกับเครื่องมือที่พัฒนาจาก Corallib Library

คำสำคัญ : วิธีวัดที่ไม่ใช้การรับส่งข้อมูลระหว่างคู่สื่อสาร

Abstract

This paper presents the analysis framework and the design idea for a tool for the measurement of Internet response time characteristics. The measurement is designed for passive traffic measurement methods for the capture and filter of the packet headers of Internet Protocol Version 4 from the internal and external networks. The data collection has to be classified and stored to three types of tables in both the internal and external networks, ie: forward, reverse and other direction tables. The data in each table will be used to analyze the packet response time of ICMP and TCP protocols that consider the corresponding packets between opposite direction tables within the same networks. The experiment is to develop a measurement response time tool for accurate analysis with an assigned value of delay, then to compare the result to the result of ICMP response time with ping command and also to the result of TCP between open and closed connections. In addition, the paper will present an accurate analysis of packet length and inter-arrival time as compared to Corallib Library Tool.

Keywords : passive traffic measurement

1. บทนำ

ระบบเครือข่ายที่ได้รับการวิเคราะห์และออกแบบอย่างเหมาะสมจะนำมาซึ่งค่าใช้จ่ายและประสิทธิภาพของเครือข่ายที่เหมาะสมตามไปด้วย การตรวจวัดปริมาณการสื่อสารข้อมูล(Traffic Measurement) จึงเป็นปัจจัยสำคัญสำหรับผู้ดูแลระบบเพื่อขจัดปัญหาหรือปรับปรุงเครือข่ายต่างต่อเนื่อง วิธีการวัดพฤติกรรมการสื่อสารข้อมูลโดยทั่วไปแบ่งออกได้ 2 วิธี คือ การวัดที่ต้องใช้การรับส่งข้อมูลระหว่างคู่สื่อสาร (Active Traffic Measurement)และการวัดที่ไม่ใช้การรับส่งข้อมูลระหว่างคู่สื่อสาร(Passive Traffic Measurement) เครื่องมือวัดที่มีประสิทธิภาพจะต้องไม่ก่อให้เกิดผลกระทบหรือเพิ่มภาระงานให้กับระบบในขณะทำการวัด นอกจากนี้ผลลัพธ์ที่ได้ต้องสะท้อนถึงความเป็นจริงของการสื่อสารในสภาวะที่เป็นปัจจุบันและต่อเนื่อง ดังนั้นวิธีการวัดแบบที่ 2 จึงเป็นวิธีที่นิยมนำไปใช้ในการสร้างเครื่องมือวัดกันมาก ตัวอย่างเช่น การวัดข้อมูลไฟลว(Data Flow) ด้วย NetTraMet[1], NetFlow[2], Cflowd[3], FlowScan[4] และ CoralReef[5] นอกจากข้อมูลไฟลวแล้วยังมีข้อมูลอื่นอีกหลายชนิดที่จะช่วยสนับสนุนการวิเคราะห์และสะท้อนปัญหาที่เกิดขึ้นกับเครือข่ายการสื่อสารได้ เช่น ความยาวแพ็กเก็ต, เวลาระหว่างการมา, เวลาการสื่อสารไปกลับ ความล่าช้า และความสูญเสีย ซึ่งใน 2 ชนิดแรกสามารถตรวจวัดได้ด้วย NetTraMet หรือ CoralReef ที่พัฒนาด้วย Corallib Library หรือเวลาการสื่อสารไปกลับจะมีกับเฉพาะเครื่องมือที่ต้องใช้การรับส่งข้อมูลระหว่างคู่สื่อสาร เช่น ping ที่ใช้การรับส่งข้อมูลด้วย ICMP โปรโตคอล

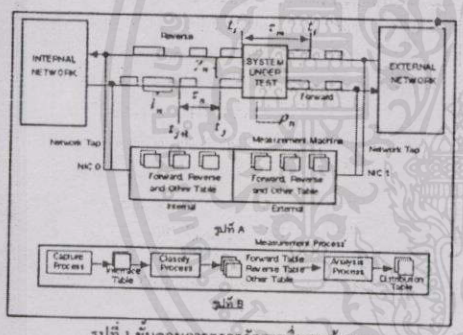
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นในบทความนี้จึงขอเสนอวิธีการตรวจวัดคุณลักษณะการสื่อสารข้อมูลที่มีเนื้อหาประกอบด้วย กรอบและแนวคิดในการรวบรวมข้อมูล การเลือกและนำข้อมูลไปใช้ในการวิเคราะห์ วิธีการวิเคราะห์เวลาการสื่อสารข้อมูลไปกลับ การทดลองวัดความยาวเฟรมเน็ตและเวลาระหว่างการมาเปรียบเทียบกับ Corallib Library และทดลองวัดเวลาการสื่อสารข้อมูลไปกลับ เปรียบเทียบกับ ping และที่หน่วงเวลาด้วย ipfw

2. ระบบการตรวจวัด

2.1 หลักการทำงาน

พิจารณาการสื่อสารแบบแพ็กเก็ตสวิช(Packet Switch)[6] ที่มีการส่งแพ็กเก็ตจากฝั่งเครือข่ายภายใน(Internal Network) ผ่านตัวระบบ(System Under Test)ไปในฝั่งเครือข่ายภายนอก(External Network) ซึ่งตัวระบบอาจอยู่ในรูปของอุปกรณ์เชื่อมต่อเครือข่าย เช่น Router Device หรือ Switch Device ในการสื่อสารข้อมูลระหว่างผู้รับและผู้ส่งสามารถจัดรูปแบบของการแลกเปลี่ยนข้อมูลออกได้ 2 ลักษณะ คือ มิทิศทางการสื่อสารเป็นขาออก(Forward Direction) และมีทิศทางการสื่อสารเป็นขาเข้า(Reverse Direction) ระบบการตรวจวัดใช้หลักการแบ่งหน่วยตรวจจับข้อมูล(Capture Process)ออกเป็น 2 ฝั่ง ข้อมูลที่รวบรวมได้ในครั้งแรกจัดเก็บในรูปของตารางข้อมูลเดียวบน Interface Table ของแต่ละหน่วย และจะถูกเรียกไปใช้โดยหน่วยแยกชนิด(Classify Process) ตามทิศทางการสื่อสารข้อมูลที่ประกอบด้วย ตารางการสื่อสารขาออก (Forward Table) ตารางการสื่อสารขาเข้า(Reverse Table) และตารางการสื่อสารอื่น ๆ (Other Table) ดังรูปที่ 1.A และ 1.B



รูปที่ 1 ขั้นตอนการตรวจวัดการสื่อสารข้อมูล

จากรูปที่ 1.B หน่วยวิเคราะห์(Analysis Process)จะทำหน้าที่อ่านข้อมูลที่ได้รับการแยกชนิดแล้วไปคำนวณและสรุปผลการวิเคราะห์ในรูปของตารางแจกแจงความถี่ข้อมูลเป็นขั้นตอนสุดท้าย และขอใช้สัญลักษณ์เพื่ออธิบายความหมายของคุณลักษณะข้อมูลต่าง ๆ ดังนี้

l_n หมายถึงความยาวแพ็กเก็ตรวบรวมได้จาก Network Header [7]

τ_n หมายถึงเวลาระหว่างการมาถึงตัวระบบของแพ็กเก็ต คำนวณได้จากผลต่างระหว่างเวลาของแพ็กเก็ตแรก(t_i)กับแพ็กเก็ตถัดไป(t_{i+1}) เดินทางถึงตัวระบบ

γ_n หมายถึงเวลาการสื่อสารข้อมูลไปและกลับของคู่แพ็กเก็ตที่มีการตอบรับการสื่อสารกัน คำนวณได้จากผลต่างระหว่างเวลาของแพ็กเก็ตฝ่ายตอบ(t_i)กับแพ็กเก็ตฝ่ายเรียก(t_j)จากคู่สื่อสารเดียวกัน

τ_m หมายถึงช่วงเวลาที่แพ็กเก็ตเดียวกันเดินทางเข้าและออกจากตัวระบบหรือเรียกช่วงเวลานี้ว่า ความล่าช้า คำนวณได้จากผลต่างระหว่างเวลาของแพ็กเก็ตเดียวกัน(t_i) ที่จัดเก็บในตารางการสื่อสารตามทิศทางเดียวกันแต่อยู่ฝั่งเครือข่ายตรงข้ามกัน

ρ_n หมายถึงแพ็กเก็ตที่เดินทางเข้าและไม่พบในฝั่งขาออกจากตัวระบบหรือเรียกว่า ความสูญเสีย วิเคราะห์ได้จากการเปรียบเทียบหาแพ็กเก็ตเดียวกัน(t_i)ในตารางการสื่อสารที่มีทิศทางเดียวกันแต่อยู่ฝั่งเครือข่ายตรงข้ามกัน

2.2 วิธีตรวจจับและแยกชนิดข้อมูล

ลักษณะของข้อมูลที่รวบรวมได้จาก Network Interface Cards ในครั้งแรกจะอยู่รวมกันทุกทิศทางทั้ง ขาออก ขาเข้า และการสื่อสารกันเองภายในเครือข่าย ข้อมูลในแต่ละ Interface Table จะถูกนำไปแยกชนิดตามทิศทางการสื่อสารข้อมูล โดยแบ่งออกเป็น 3 ชนิด คือ

ตารางข้อมูลขาออก(Forward Table) หมายถึง ตารางที่จัดเก็บข้อมูลไอพีแพ็กเก็ตที่มีทิศทางการสื่อสารจากเครือข่ายภายในไปยังเครือข่ายภายนอก

ตารางข้อมูลขาเข้า(Reverse Table) หมายถึง ตารางที่จัดเก็บข้อมูลไอพีแพ็กเก็ตที่มีทิศทางการสื่อสารจากเครือข่ายภายนอกเข้าสู่เครือข่ายภายใน

ตารางข้อมูลอื่น ๆ(Other Table) หมายถึง ตารางที่จัดเก็บข้อมูลทุกชนิดที่มีลักษณะแตกต่างไปจากที่กำหนด

โดยมีขั้นตอนการแบ่งแยกชนิดข้อมูลดังรูปที่ 2

```
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;
}
```

รูปที่ 2 การแบ่งแยกทิศทางการสื่อสารข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ SUT หมายถึง หมายเลขไอพีที่ใช้บนคัวระบบ

INT หมายถึง หมายเลขไอพีที่ใช้ในฝั่งเครือข่ายภายใน

การแบ่งแยกทิศทางการสื่อสารข้อมูลใช้วิธีเปรียบเทียบเฉพาะแพ็กเก็ตที่มีชนิดเป็นไอพีเวอร์ชั่นที่ 4 โดยมีเงื่อนไขหลักดังนี้

จัดเก็บบนตารางข้อมูลขาออกในกรณีที่ไอพีปลายทาง(ip_dst) ของแพ็กเก็ตไม่อยู่ในกลุ่มของไอพีที่ใช้บนคัวระบบ(SUT)และเครือข่ายภายใน(INT)

จัดเก็บบนตารางข้อมูลขาเข้าในกรณีที่ไอพีต้นทาง(ip_src) ของแพ็กเก็ตไม่อยู่ในกลุ่มไอพีที่ใช้บนคัวระบบและเครือข่ายภายใน และไอพีปลายทางไม่อยู่ในกลุ่มของไอพีที่ใช้บนฝั่งเครือข่ายภายใน

จัดเก็บบนตารางข้อมูลอื่น ๆ ทุกกรณีที่แพ็กเก็ตมีลักษณะแตกต่างไปจาก 2 กรณีแรก

หลังจากผ่านกระบวนการแบ่งแยกทิศทางการสื่อสารข้อมูลจากทั้ง 2 Interface Table แล้ว ในแต่ละ Interface Table จะถูกแบ่งแยกออกเป็นตารางข้อมูล 3 ลักษณะที่เหมือนกัน และจะถูกอ่านไปใช้โดยหน่วยวิเคราะห์ข้อมูล(Analysis Process)ต่อไป เช่น นำไปใช้วิเคราะห์ความยาวและเวลาระหว่างการมาของแพ็กเก็ต สามารถวิเคราะห์ได้จากทั้งตารางข้อมูลที่มีทิศทางการสื่อสารขาเข้าหรือออกจาก Interface Table ฝั่งเครือข่ายภายในที่ทำการแยกชนิดแล้ว โดยสามารถศึกษาขั้นตอนการวิเคราะห์ได้จากงานวิจัยที่พัฒนาเครื่องมือวัดด้วย Corallib Library[8].

3. วิธีการตรวจวัดเวลาการสื่อสารข้อมูลไปกลับ

ทำการวิเคราะห์เฉพาะแพ็กเก็ตที่โปรโตคอลมีชนิดเป็น TCP และ ICMP(Echo)เท่านั้น โดยขอนิยามข้อมูลที่ได้รับการพิจารณาเหล่านี้ว่า Considered Packet แต่ละแพ็กเก็ตจะถูกนำไปเปรียบเทียบหาข้อมูลที่เป็นคู่การสื่อสารกันจากตารางข้อมูลที่มีทิศทางการสื่อสารตรงข้ามกันและอยู่ในฝั่งเครือข่ายเดียวกัน และขอนิยามเรียกแพ็กเก็ตที่เป็นคู่การสื่อสารกันนี้ว่า Corresponding Packet โดยมีขั้นตอนดังรูปที่ 3

```
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_j
            if (Req_j Corresponding to Res_i)
                Compute γ_n : γ_n = (t_res - t_req) = t_i - t_j
        }
    }
}
```

รูปที่ 3 การตรวจวัดเวลาการสื่อสารข้อมูลไปและกลับ

การตรวจวัดใช้วิธีเปรียบเทียบโดยนำแพ็กเก็ตของฝ่ายตอบ (Response Table)ไปค้นหาแพ็กเก็ตของฝ่ายเรียก(Request Table) ในตารางข้อมูลที่มีทิศทางการสื่อสารตรงข้ามกัน(Forward or Reverse Table) ซึ่งแพ็กเก็ตของฝ่ายตอบอาจอยู่ในรูปของตารางข้อมูลขาออก หรือตารางข้อมูลขาเข้าก็ได้ ในที่นี้พิจารณาแพ็กเก็ตฝ่ายตอบจากตารางข้อมูลที่มีทิศทางการสื่อสารเป็นขาเข้าและนำไปพิจารณาแพ็กเก็ตฝ่ายเรียกในตารางที่มีทิศทางการสื่อสารเป็นขาออก โดยจะนำแพ็กเก็ตที่เป็นคู่สื่อสารกันไปคำนวณหาเวลาของการสื่อสารข้อมูลไปกลับจากสมการ

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

โดยที่ t_i หมายถึง เวลาของแพ็กเก็ตฝ่ายตอบ

t_j หมายถึง เวลาของแพ็กเก็ตฝ่ายเรียก

การตรวจวัดเวลาไปกลับของคู่แพ็กเก็ต ต้องแยกพิจารณากลุ่มการสื่อสารของข้อมูลที่ต่างชนิดกันดังนี้

3.1 การตรวจวัดเวลาการสื่อสารไปกลับของข้อมูล ICMP

พิจารณาเฉพาะแพ็กเก็ตฝ่ายตอบ(Res_i)ที่ใช้กับการสื่อสารแบบ Echo Reply หรือมี ICMP Type เท่ากับ 0x00 ไปเปรียบเทียบหาคู่ของการสื่อสารข้อมูลในฝ่ายเรียก(Req_j)ที่ใช้การสื่อสารแบบ Echo หรือมี ICMP Type เท่ากับ 0x08 โดยมี ICMP Identifier และ ICMP Sequence Number ที่ตรงกัน ดังรูปที่ 4

```
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 &&
                        Req_j.icmp_seq == Res_i.icmp_seq)
```

รูปที่ 4 การพิจารณาคู่แพ็กเก็ตของการสื่อสารข้อมูลชนิด ICMP

3.2 การตรวจวัดเวลาการสื่อสารไปกลับของข้อมูล TCP

โปรโตคอล TCP เป็นการสื่อสารข้อมูลที่มี Reliability การรับส่งข้อมูลมีความแตกต่างจากข้อมูลชนิดอื่น ๆ คือมีการแบ่งขั้นตอนของการสื่อสารออกเป็น 3 ขั้นตอน ได้แก่ สร้างการเชื่อมต่อ(Established) รับส่งข้อมูล(Syn-Sent) และ ปิดการเชื่อมต่อ(Closed) โดยที่แต่ละคู่ของการสื่อสารจะใช้คู่สัญญาณตอบรับที่แตกต่างด้วย SYN/ACK, PUSH/ACK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ FIN/ACK ตามลำดับ ในการวิเคราะห์ช่วงเวลาการสื่อสารไปกลับเราไม่สามารถหาคู่แพ็กเก็ตตอบรับในช่วงของการรับส่งข้อมูลได้ เนื่องจากไม่สามารถคาดเดาเหตุการณ์ที่จะเกิดความสูญเสียข้อมูลระหว่างการรับส่งข้อมูลต่อหนึ่งชุดได้ ดังนั้นการวิเคราะห์เวลาการสื่อสารไปกลับแบบ TCP จึงวิเคราะห์ได้เฉพาะในช่วงเวลาของการเปิด และปิดการเชื่อมต่อเท่านั้น โดยมีเงื่อนไขของการวิเคราะห์เวลาระหว่างเปิดการเชื่อมต่อดังรูปที่ 5

```
Considered Packet = (Res_ip_proto==TCP(0x06) &&
                     Res_th_flags==SYN|ACK(0x02|0x10))
Corresponding Packet = (Req_ip_p == Res_ip_p &&
                        Req_ip_dst == Res_ip_src &&
                        Req_ip_src == Res_ip_dst &&
                        Req_tcp_sport == Res_tcp_dport &&
                        Req_tcp_dport == Res_tcp_sport &&
                        Req_tcp_flags == SYN(0x02) &&
                        Req_tcp_seq == Res_tcp_seq - 1)
```

รูปที่ 5 เงื่อนไขระหว่าง TCP สร้างการเชื่อมต่อ

กลุ่มข้อมูลที่พิจารณาคือแพ็กเก็ตของฝั่งตอบ(Res)ที่ส่งสัญญาณข้อมูลชนิด SYN/ACK ให้กับฝั่งเรียก(Req)ที่ส่งด้วยสัญญาณชนิด SYN การเปรียบเทียบคู่ของแพ็กเก็ตพิจารณาจาก Sequence Number ของแพ็กเก็ตในฝั่งเรียกมีค่าเท่ากับ Sequence Number ตามด้วย 1 ของแพ็กเก็ตฝั่งตอบ และมีเงื่อนไขของการวิเคราะห์ช่วงเวลาระหว่างปิดการเชื่อมต่อดังรูปที่ 6

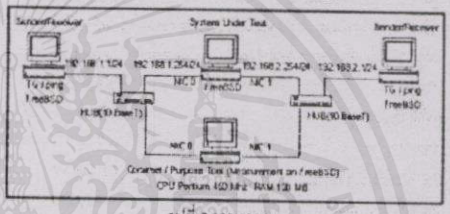
```
Considered Packet = (Res_ip_proto==TCP(0x06) &&
                     Res_th_flags==ACK(0x10))
Corresponding Packet = (Req_ip_p == Res_ip_p &&
                        Req_ip_dst == Res_ip_src &&
                        Req_ip_src == Res_ip_dst &&
                        Req_tcp_sport == Res_tcp_dport &&
                        Req_tcp_dport == Res_tcp_sport &&
                        Req_tcp_flags == FIN|ACK(0x01|0x10) &&
                        Req_tcp_seq == Res_tcp_seq - 1)
```

รูปที่ 6 กรณีที่ TCP ปิดการเชื่อมต่อ

กลุ่มของข้อมูลที่พิจารณาคือแพ็กเก็ตของฝั่งตอบ(Res)ที่ส่งสัญญาณข้อมูลชนิด ACK ให้กับฝั่งเรียก(Req)ที่ส่งด้วยสัญญาณชนิด FIN/ACK การเปรียบเทียบคู่ของแพ็กเก็ตจะพิจารณาจาก Sequence Number ของแพ็กเก็ตในฝั่งเรียกมีค่าเท่ากับ Sequence Number ตามด้วย 1 ของแพ็กเก็ตฝั่งตอบเช่นเดียวกับกรณีเปิดการเชื่อมต่อ

4. การทดลอง

พัฒนาเครื่องมือวัดด้วยโปรแกรมภาษา C++[9] บนระบบปฏิบัติการฟรีบีเอสดี(FreeBSD[10]) โดยเรียกใช้ Libpcap Library ในการตรวจจับข้อมูล ทำการทดลองวัดการสื่อสารข้อมูลระหว่างคอมพิวเตอร์ที่อยู่ภายใน 2 เครือข่ายย่อย ที่ใช้ไอพีหมายเลข 192.168.1.1/24 และ 192.168.2.1/24 ตามลำดับ โดยใช้ซอฟต์แวร์สร้างการสื่อสารข้อมูล (Traffic Generator)ด้วย TG และ ping คอมพิวเตอร์ที่ทำหน้าที่เป็นตัวระบบติดตั้งระบบปฏิบัติการฟรีบีเอสดี และเปิดโปรโตคอลค้นหาเส้นทางการสื่อสารด้วย Routing Information 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 II ความเร็ว 450 MHz หน่วยความจำขนาด 128 MB ตรวจจับแพ็กเก็ตผ่าน 2 NIC ที่ความเร็ว 10 Mb/sec ดังรูปที่ 7



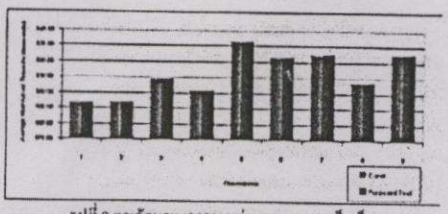
รูปที่ 7 การทดลอง

4.1 การวิเคราะห์ความยาวและเวลาระหว่างการมา

ทดลองสุ่มข้อมูลที่มีขนาดไม่เท่ากันเป็นจำนวน 500 แพ็กเก็ตส่งไปให้ฝั่งตรงกันข้าม ทำการทดสอบความแม่นยำด้วยการเปรียบเทียบผลการตรวจวัดระหว่างเครื่องมือวัดในงานวิจัย(Purposed Tool) และเครื่องมือที่พัฒนาจาก CoralLib Library เป็นจำนวน 9 ครั้ง ผลการทดลองของทั้ง 2 เครื่องมือ ตรวจวัดได้คุณลักษณะของข้อมูลดังรูปที่ 8 และ 9

รูปที่ 8 คุณลักษณะความยาวของแพ็กเก็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 9 คุณลักษณะเวลาระหว่างการมาของแพ็กเก็ต

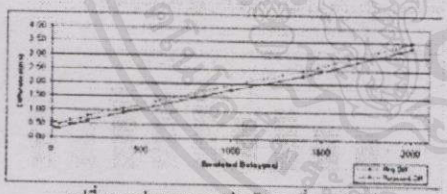
ผลการตรวจวัดและเปรียบเทียบความยาวของทุกแพ็กเก็ตจากทั้ง 2 เครื่องมือมีค่าเท่ากันในทุกแพ็กเก็ต ได้ค่าเฉลี่ยของทุกแพ็กเก็ตเป็น 724.18 Bytes เมื่อเปรียบเทียบด้วย Confidence Interval(95%)[11] ได้ค่าเฉลี่ยของการตรวจวัดอยู่ในช่วงเดียวกันคือ (713.25,735.12) วิเคราะห์ได้ว่าทั้งสองเครื่องมือ ตรวจวัดความยาวของแพ็กเก็ตได้ไม่แตกต่างกัน

ส่วนผลการตรวจวัดเวลาระหว่างการมามีความแตกต่างกันน้อยมาก การตรวจวัดด้วยเครื่องมือที่พัฒนาจาก Corallib Library และเครื่องมือวิจัยให้ความแตกต่างกันในหน่วยเวลาเป็น Microseconds ค่าเฉลี่ยเวลาของ Corallib Library และเครื่องมือวิจัยคือ 612.34 และ 612.35 Microseconds ตามลำดับ เปอร์เซนต์เฉลี่ยของความแตกต่างระหว่างเครื่องมือคือ 0.000021 เมื่อเปรียบเทียบด้วย Confidence Interval(95%) ให้ค่าเฉลี่ยของการตรวจวัดอยู่ในช่วง (603.30, 621.38) และ (603.31,621.39) วิเคราะห์ได้ว่าทั้งสองเครื่องมือตรวจวัดเวลาระหว่างการมาได้ไม่แตกต่างกัน

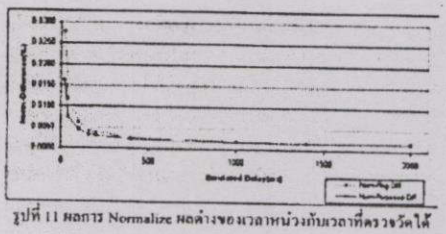
4.2 การวิเคราะห์เวลาการสื่อสารข้อมูลไปกลับ

4.2.1 ข้อมูลชนิด ICMP

ทดลองส่งข้อมูลด้วย ICMP เป็นจำนวน 100 แพ็กเก็ตระหว่างเครื่องข่าย เปรียบเทียบผลลัพธ์การตรวจวัดระหว่างเครื่องมือวิจัยและ ping ทำการหน่วงเวลาของการทดลองในแต่ละครั้งบนตัวระบบฝั่งขาเข้า(In Bound)ด้วยอัตราเวลา 0, 10, 20, 50, 80, 100, 200, 500, 700 และ 1000 ms ตามลำดับ โดยใช้ความสามารถของ ipfw บนระบบปฏิบัติการ BSD ผลการทดลองเปรียบเทียบได้คุณลักษณะของข้อมูลดังรูปที่ 10 และ 11



รูปที่ 10 ผลต่างของเวลาหน่วงกับเวลาที่ตรวจวัดได้

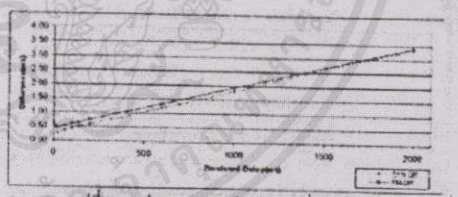


รูปที่ 11 ผลการ Normalize ผลต่างของเวลาหน่วงกับเวลาที่ตรวจวัดได้

จากรูปที่ 10 ผลต่างของการหน่วงเวลากับเวลาที่ตรวจวัดได้จากทั้งสองเครื่องมือมีความแตกต่างกันในทุกครั้งของการทดลอง โดยที่ค่าเฉลี่ยของความแตกต่างที่ตรวจวัดได้จาก ping คือ 1.276191 ms ส่วนเครื่องมือวิจัยให้ความแตกต่างที่ 1.090328 ms ซึ่งน้อยกว่าการตรวจวัดด้วย ping ความแตกต่างนี้เป็นผลมาจากทั้งสองเครื่องมือมีการคลาดเวลาของข้อมูลเพื่อนำไปใช้ในการคำนวณบนชั้นเครือข่ายที่แตกต่างกัน และเมื่อทำการ Normalize ผลต่างของทั้งสองเครื่องมือดังรูปที่ 11 จะพบว่ามีเปอร์เซนต์ความแตกต่างที่ตรวจวัดได้จาก ping คือ 0.006897 ส่วนเครื่องมือวิจัยตรวจวัดได้ 0.004655 การทดลองในแต่ละครั้งโดยใช้เวลาหน่วงที่มากขึ้นคุณลักษณะของทั้งสองเครื่องมือจะลดความแตกต่างกันลงไปซึ่งจะเห็นได้จากกราฟของทั้งสองเริ่มมีค่าใกล้เคียงกันมากตั้งแต่เวลาหน่วงที่ 100 ms หรือไปกลับที่ 200 ms ขึ้นไป

4.2.2 ข้อมูลชนิด TCP

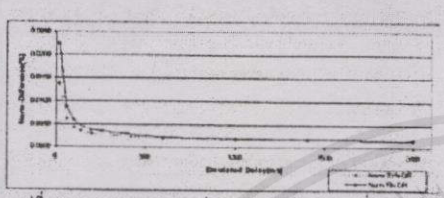
การทดลองข้อมูลชนิด TCP จะแตกต่างจากการทดลองอื่น ๆ เนื่องจากยังไม่พบเครื่องมือที่ใช้ตรวจวัดข้อมูลชนิดนี้ ดังนั้นจึงใช้วิธีการหน่วงเวลาของการทดลองในแต่ละครั้งบนตัวระบบฝั่งขาเข้า(In Bound) เป็น 0, 10, 30, 50, 70, 100, 300, 500, 700 และ 1000 ms ตามลำดับ ในแต่ละการทดลองทำการส่งด้วยข้อมูลชนิด TCP ความยาวขนาด 500 bytes จำนวน 100 ครั้ง ไปไปมาระหว่างกันข้าม ทำการเปรียบเทียบผลลัพธ์ของเวลาเฉลี่ยระหว่างขั้นตอนของการสื่อสารคือ เวลาระหว่างสร้างการเชื่อมต่อ และปิดการเชื่อมต่อ ผลการทดลองวิเคราะห์เวลาระหว่างสร้างการเชื่อมต่อได้คุณลักษณะข้อมูล ดังรูปที่ 12



รูปที่ 12 ผลต่างของเวลาหน่วงกับเวลาที่ตรวจวัดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลต่างระหว่างเวลาหน่วงกับเวลาที่ตรวจวัดได้จากทั้งสองเครื่องมือมีความแตกต่างกันในทุกการทดลองของการหน่วงเวลา และพบว่าเมื่อทำการทดลองด้วยการหน่วงเวลาที่มากขึ้น ช่วงเวลาของการเปิดและปิดการเชื่อมต่อจะมีผลของหวานแตกต่างที่ใกล้เคียงกัน โดยมีค่าเฉลี่ยของความแตกต่างที่ตรวจวัดได้ในช่วงเปิดการเชื่อมต่อคือ 1.152212 ms ส่วนช่วงเวลาปิดการเชื่อมต่อให้ความแตกต่างที่ 1.276668 ms ซึ่งมีค่ามากกว่าในช่วงเวลาเปิดการเชื่อมต่อ ความแตกต่างนี้เป็นผลมาจากช่วงเวลาปิดการเชื่อมต่อของฝ่ายรับและส่งอาจต้องสูญเสียเวลาบางส่วนไปสำหรับเรียกเรียงข้อมูลก่อนปิดการเชื่อมต่อ เมื่อทำการ Normalize ผลต่างของทั้งสองช่วงเวลา ผลลัพธ์ที่ได้มีลักษณะข้อมูลดังรูปที่ 13



รูปที่ 13 ผลการ Normalize ผลต่างของเวลาหน่วงกับเวลาที่ตรวจวัดได้

เปอร์เซนต์ความแตกต่างที่ตรวจวัดได้ในช่วงเปิดการเชื่อมต่อคือ 0.004171 ส่วนส่วนช่วงเวลาปิดการเชื่อมต่อคือ 0.005890 ผลการวิเคราะห์ในแต่ละครั้งเมื่อมีการทดลองโดยใช้การหน่วงเวลาที่มากขึ้น พบว่าทั้งสองช่วงเวลาเริ่มลดความแตกต่างกันลงไปซึ่งจะเห็นได้จากกราฟของทั้งสองเริ่มมีค่าแตกต่างกันเพียงเล็กน้อยตั้งแต่การหน่วงเวลาที่ 300 ms หรือไปกลับในเวลา 600 ms ขึ้นไป

5. บทสรุป

สำหรับบทความนี้ได้นำเสนอการออกแบบวิธีตรวจวัดคุณลักษณะการสื่อสารข้อมูลไปกลับบนเครือข่ายอินเทอร์เน็ต ทำการทดสอบความแม่นยำด้วยการตรวจวัดความยาว และเวลาระหว่างการมาเปรียบเทียบกันเครื่องมือที่พัฒนาจาก Coralib Library ผลการวิเคราะห์ด้วยเปอร์เซนต์ของความแตกต่าง และค่า Confidence Interval(95%) พบว่าทั้งสองวิธีการได้ผลที่ไม่แตกต่างกัน ส่วนการตรวจวัดเวลาการสื่อสารข้อมูลไปกลับด้วยข้อมูลชนิด ICMP กับ ping และข้อมูลชนิด TCP ระหว่างเปิดและปิดการเชื่อมต่อ ผลการเปรียบเทียบให้ความแตกต่างเกิดขึ้นในทุกการทดลอง ความแตกต่างที่เกิดขึ้น ในกรณีของ ICMP เป็นผลมาจากวิธีการคำนวณเวลาในชั้นเครือข่ายที่แตกต่างกัน ส่วน TCP เป็นผลมาจากการสูญเสียเวลาในการรวบรวมข้อมูลก่อนปิดการเชื่อมต่อ และเมื่อนำผลต่างจากการสื่อสารข้อมูลทั้ง 2 ชนิดไปทำ Normalize จะพบว่าช่วงเวลาไปกลับที่ตรวจวัดได้ให้ความแตกต่างกันน้อยลงเมื่อมีการทดลองด้วยการหน่วงเวลาที่มากขึ้น

การออกแบบวิธีตรวจวัดข้อมูลในลักษณะนี้มีความยืดหยุ่นต่อการวิเคราะห์ข้อมูลได้หลากหลายรูปแบบ ขึ้นกู่กับการเลือกใช้ตารางข้อมูลที่มีความเกี่ยวข้องกัน และสามารถนำวิธีการนี้ไปใช้สร้างเครื่องมือเพื่อตรวจวัดข้อมูลจริงบนเครือข่ายอินเทอร์เน็ตได้ เช่น เวลาการสื่อสารข้อมูลไปกลับ สามารถวิเคราะห์ได้จากตารางข้อมูลที่มีทิศทางการสื่อสารขาออกกับขาเข้าจากฝั่งเครือข่ายเดียวกัน และนอกจากนี้ยังสามารถนำไปใช้ในการวิเคราะห์หาความล่าช้า และความสูญเสียข้อมูลจากการสื่อสารผ่านตัวระบบ โดยพิจารณาจากตารางข้อมูลที่มีทิศทางการสื่อสารทางเดียวกันแต่อยู่ในฝั่งเครือข่ายที่แตกต่างกัน เช่น ความล่าช้าและความสูญเสียข้อมูลในทิศทางขาออก วิเคราะห์ได้จากตารางข้อมูลขาออกระหว่างฝั่งเครือข่ายภายในและภายนอก ซึ่งเป็นอีกแนวทางหนึ่งที่สามารถนำไปใช้ในการศึกษาและวิจัยต่อไป

เอกสารอ้างอิง

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

[2] "Cisco NetFlow," 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, USENIX:LISA.

[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", Prentic Hall, Inc. A Simon & Schuster Company Upper Saddle River, New Jersey, 1998.

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

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

นายแสงเพ็ชร พระฉาย เกิดมื่อวันที่ 30 พฤศจิกายน 2512 ที่จังหวัดนครราชสีมา สำเร็จการศึกษาระดับปริญญาตรี คอมพิวเตอร์ศึกษา จากวิทยาลัยครูนครราชสีมา ปีการศึกษา 2536 ปีพุทธศักราช 2535 เข้าทำงานในบริษัทเอกชน ชื่อ คอมเทคดาต้า ในตำแหน่ง Senior Programmer and Analyst มีผลงานในระหว่างปฏิบัติหน้าที่คือ พัฒนาระบบจัดเก็บสินค้าบริษัทบิวเตอร์เจมเซนต์เตอร์ พัฒนาไดร์เวอร์ภาษาไทยบนเครื่องดิจิตอล พัฒนาระบบบัญชีซื้อขายให้บริษัทคอมเทคด้าต้า พัฒนาระบบสหกรณ์ออมทรัพย์ด้วย RM Cobol พัฒนาไดร์เวอร์ควบคุมการพิมพ์ Digital and IBM Passbook Printer พัฒนาระบบสหกรณ์ออมทรัพย์ด้วย Progress Language พัฒนาระบบฝากถอนผ่าน ATM ธนาคารไทยพานิชย์ และพัฒนาระบบฝากถอนผ่าน ATM ธนาคารทหารไทย ในปีพุทธศักราช 2540 บรรจุเป็นข้าราชการ ตำแหน่ง อาจารย์ สถาบันราชภัฏนครราชสีมา มีหน้าที่รับผิดชอบนอกเหนือจากงานสอนคือ รองหัวฝ่ายทะเบียนประจำสำนักส่งเสริมวิชาการ หัวหน้าระบบเครือข่ายและเทคโนโลยีสารสนเทศ ประจำศูนย์คอมพิวเตอร์กลาง มีผลงานในระหว่างปฏิบัติหน้าที่คือ พัฒนาระบบสารสนเทศของสถาบันราชภัฏนครราชสีมา ร่วมพัฒนาระบบช่วยผู้ประสบภัยจากสึนามิกับกรมป้องกันและบรรเทาสาธารณภัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้