ใบรับรองวิทยานิพนธ์

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

วิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์)

ปริญญา

วิศวกรรมคอมพิวเตอร์	วิศวกรรมคอมพิวเตอร์
สาขา	**ภาควิชา**

เรื่อง การตรวจการโจมตีแบบฟลัดดิ้งที่แหล่งต้นทางโดยวิธีเชิงเวฟเล็ต

Wavelet-Based Detection of Source-Network Packet Flooding

นามผู้วิจัย นายเกรียงไกร ลิ่มทอง

ได้พิจารณาเห็นชอบโดย

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก .....

(ผู้ช่วยศาสตราจารย์พีรวัฒน์ วัฒนพงศ์, Ph.D.)

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม .....

(รองศาสตราจารย์พันธุ์ปิติ เปี่ยมสง่า, D.Sc.)

หัวหน้าภาควิชา .....

(ผู้ช่วยศาสตราจารย์เขมะฑัต วิภาตะวนิช, Ph.D.)

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์รับรองแล้ว

.....

(รองศาสตราจารย์กัญจนา ธีระกุล, D.Agr.)

คณบดีบัณฑิตวิทยาลัย

วันที่ ..... **เดือน** ..... **พ.ศ.** .....

วิทยานิพนธ์

เรื่อง

การตรวจการโจมตีแบบฟลัดดิ้งที่แหล่งต้นทางโดยวิธีเชิงเวฟเล็ต

Wavelet-Based Detection of Source-Network Packet Flooding

โดย

นายเกรียงไกร ลิ่มทอง

เสนอ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

เพื่อความสมบูรณ์แห่งปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์)

พ.ศ. 2552

เกรียงไกร ลิ่มทอง 2552: การตรวจการโจมตีแบบฟลัดดิ้งที่แหล่งต้นทางโดยวิธีเชิงเวฟเล็ต ปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมคอมพิวเตอร์) สาขาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก: ผู้ช่วยศาสตราจารย์พีรวัฒน์ วัฒนพงศ์, Ph.D. 94 หน้า

วิทยานิพนธ์ฉบับนี้ได้นำเสนอวิธีตรวจจับการโจมตีแบบฟลัดดิ้งที่แหล่งต้นทางโดยวิธีเชิงเวฟเล็ต โดยการเปลี่ยนข้อมูลที่ได้รับจากระบบเครือข่ายคอมพิวเตอร์ให้อยู่ในรูปสัญญาณที่มีความสัมพันธ์ระหว่างจำนวนชุดข้อมูลที่ส่งกับช่วงเวลา จากนั้นจึงแยกส่วนประกอบความถี่จากสัญญาณดังกล่าวเป็นส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำแต่ละระดับ และวิเคราะห์ความผิดปกติในระบบเครือข่ายคอมพิวเตอร์จากส่วนประกอบความถี่ในแต่ระดับ ถ้าไม่เกิดความผิดปกติในระบบเครือข่ายคอมพิวเตอร์ก็ให้นำรูปสัญญาณดังกล่าวไปคำนวณทางสถิติเพื่อสร้างค่าฐานสำหรับนำมาใช้ในการตรวจจับความผิดปกติต่อไป

งานวิจัยนี้ได้ทำการศึกษาโปรโตคอลมาตรฐานที่ใช้ในระบบเครือข่ายคอมพิวเตอร์ ได้แก่ ICMP, TCP SYN, TCP SYN/ACK และ UDP โดยกำหนดช่วงเวลาในการจัดเก็บข้อมูลเท่ากับ 1 วินาที และใช้เวฟเล็ตแม่แบบ Haar ในการแยกส่วนประกอบความถี่ของสัญญาณที่ได้รับจากระบบเครือข่ายคอมพิวเตอร์ ข้อมูลที่นำมาทดลองจัดเก็บจากห้องปฏิบัติการคอมพิวเตอร์และอินเตอร์เน็ต มหาวิทยาลัยเกษตรศาสตร์ บางเขน ระหว่างเดือน มิถุนายน ถึง สิงหาคม 2551

ผลจากงานวิจัยนี้แสดงให้เห็นว่าเราสามารถใช้การแยกส่วนประกอบความถี่ด้วยเวฟเล็ตและการวิเคราะห์ส่วนประกอบความถี่ดังกล่าวเพื่อตรวจจับความผิดปกติที่เกิดจากการโจมตีแบบฟลัดดิ้งที่แหล่งต้นทางได้ โดยมีปัจจัยหลักที่มีผลต่อความเที่ยงตรงและแม่นยำในการตรวจจับการส่งชุดข้อมูลจำนวนมากที่เครือข่ายต้นทางคือระดับของส่วนประกอบความถี่ที่นำมาใช้ในการตรวจจับ

_____ / ___ / ___

ลายมือชื่อนิสิต ลายมือชื่ออาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

Kriangkrai Limthong 2009: Wavelet-Based Detection of Source-Network Packet Flooding. Master of Engineering (Computer Engineering), Major Field: Computer Engineering, Department of Computer Engineering. Thesis Advisor: Associate Professor Pirawat Watanapongse, Ph.D. 94 pages.

In this thesis, we proposed wavelet-based flooding detection method at source of a computer network. We changed the data which received from the computer network to signal that have a relationship between number of packet and time interval. After that, we decomposed the original signal into high frequency part and low frequency part for each level. Moreover, we analyze each level of the composite parts in order to detect anomaly behavior in computer network. If it is not have an abnormality in computer network, we will combine the original signal with normal signal which store in database so as to use for next detection.

In this work, we studied on the standard protocols that using in computer network such as, ICMP, TCP SYN, TCP SYN/ACK and UDP. The interval time in this work is 1 second and we use Haar mother of wavelet to decomposed original signal. The data set, we collected from Kasetsart IT Square laboratory rooms between June and August 2008.

The results from this work indicated that we can use wavelet-based decomposition and analysis of high frequency part and low frequency part in order to detect anomaly at source of computer network. The main of parameters for accurate and precisely detection of source network packet flooding is level of decomposition that use for this detection method.

_____ _____ ___ / ___ / ___

Student's signature Thesis Advisor's signature

กิตติกรรมประกาศ

กราบขอบพระคุณ ผศ.ดร.พีรวัฒน์ วัฒนพงศ์ ประธานกรรมการที่ปรึกษาวิทยานิพนธ์หลัก รศ.ดร.พันธุ์ปิติ เปี่ยมสง่า กรรมการที่ปรึกษาวิทยานิพนธ์ร่วม และ ดร.ศิวรักษ์ ศิวโมกษธรรม หัวหน้าหน่วยปฏิบัติการวิจัยเทคโนโลยีและนวัตกรรมเพื่อความมั่นคงของประเทศ ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ ที่ให้คำปรึกษาในการเรียน การค้นคว้าวิจัย ตลอดจนการตรวจแก้ไขวิทยานิพนธ์จนกระทั่งเสร็จสมบูรณ์

วิทยานิพนธ์นี้ได้รับทุนสนับสนุนจากสถาบันบัณฑิตวิทยาศาสตร์และเทคโนโลยีไทย (TGIST) สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยีแห่งชาติ ประจำปี 2551 สัญญาทุนเลขที่ TGIST 01-51-092

ขอขอบคุณเพื่อน ๆ ทุกคนที่ช่วยเหลือและให้คำแนะนำต่าง ๆ โดยเฉพาะเพื่อน MCPE 11 ที่คอยให้คำแนะนำช่วยเหลือแก่ข้าพเจ้า อีกทั้งยังเป็นกำลังใจแก่ข้าพเจ้า และที่ลืมไม่ได้คือเธอผู้นั้น ผู้ซึ่งเป็นแรงและกำลังใจของข้าพเจ้าตลอดมา

สุดท้าย กราบขอบพระคุณ คุณพ่อ คุณแม่ พี่ชาย และพี่สาว ผู้ซึ่งสร้างครอบครัวที่อบอุ่น ผู้เป็นความรักและกำลังใจที่ยิ่งใหญ่ให้แก่ข้าพเจ้า อีกทั้งยังเป็นแรงบันดาลใจและให้การสนับสนุนข้าพเจ้าเสมอมา

เกรียงไกร ลิ่มทอง

เมษายน 2552

สารบัญ

สารบัญตาราง

สารบัญภาพ

สารบัญภาพ (ต่อ)

ภาพที่ **หน้า**

ภาพผนวกที่

สารบัญภาพ (ต่อ)

ภาพผนวกที่ **หน้า**

สารบัญภาพ (ต่อ)

ภาพผนวกที่ **หน้า**

สารบัญภาพ (ต่อ)

ภาพผนวกที่ **หน้า**

คำอธิบายสัญลักษณ์และคำย่อ

CERT	=	Computer Emergency Response Team
CWT	=	Continuous Wavelet Transform
DoS	=	Denial of Service
DDoS	=	Distributed Denial of Service
DWT	=	Discrete Wavelet Transform
FT	=	Fourier Transform
FFT	=	Fast Fourier Transform
GT	=	Gabor Transform
HPF	=	High-Pass Filter
ICMP	=	Internet Message Control Protocol
KITS	=	Kasetsart IT Square
LPF	=	Low-Pass Filter
MRA	=	Multi-Resolution Analysis
NS	=	The Network Simulator
NS-2	=	The Network Simulator 2
QMF	=	Quadrature Mirror Filter
SNMP	=	Simple Network Management Protocol
SSE	=	Error Sum of Squares
STFT	=	Short-Time Fourier Transform
TCP SYN	=	Transmission Control Protocol with SYN flag
TCP SYN/ACK	=	Transmission Control Protocol with SYN and ACK flag
UDP	=	User Datagram Protocol
WT	=	Wavelet Transform

การตรวจการโจมตีแบบฟลัดดิ้งที่แหล่งต้นทางโดยวิธีเชิงเวฟเล็ต

Wavelet-Based Detection of Source-Network Packet Flooding

คำนำ

เครือข่ายอินเตอร์เน็ตมีการเจริญเติบโตอย่างรวดเร็วและมีผู้ใช้งานเพิ่มขึ้นเป็นจำนวนมาก ในช่วงระยะเวลาไม่กี่ปีที่ผ่านมา สิ่งที่ควรตระหนักเป็นอย่างมากในการใช้งานคือความมั่นคงและความปลอดภัยบนอินเตอร์เน็ต Computer Emergency Response Team (CERT) ซึ่งทำหน้าที่ตรวจสอบเหตุการณ์ที่เกี่ยวข้องกับความมั่นคงและความปลอดภัยบนอินเตอร์เน็ตรายงานถึงจำนวนเหตุการณ์ผิดปกติที่เกิดขึ้นบนอินเตอร์เน็ตได้เพิ่มจาก 6 ครั้งในปี 1988 เป็น 137,529 ครั้งในปี 2003 เหตุการณ์ดังกล่าวทำให้เกิดความเสียหายทั้งทางด้านเศรษฐกิจและสังคมแก่ผู้ใช้งานอินเตอร์เน็ตเป็นจำนวนมาก โดยเหตุการณ์ผิดปกติดังกล่าวเกิดมาจากสาเหตุหลายประการ เช่น เกิดจากไวรัสคอมพิวเตอร์, เกิดจากการโจรกรรมข้อมูลธุรกรรมด้านการเงิน, เกิดจากการโจมตีเพื่อหยุดการให้บริการ เป็นต้น

การโจมตีเพื่อหยุดการให้บริการ (Denial of Service : DoS) และการโจมตีแบบกระจายเพื่อหยุดการให้บริการ (Distributed Denial of Service : DDoS) เป็นสาเหตุหนึ่งของเหตุการณ์ผิดปกติที่เกิดขึ้นเป็นจำนวนมากในระบบอินเตอร์เน็ต การโจมตีดังกล่าวสามารถแบ่งวิธีการโจมตีได้เป็น 2 วิธี วิธีที่หนึ่งเป็นการโจมตีไปยังช่องโหว่ในโปรแกรมระบบของอุปกรณ์หรือเครื่องคอมพิวเตอร์เพื่อทำให้อุปกรณ์หรือเครื่องคอมพิวเตอร์หยุดทำงาน วิธีที่สองเป็นวิธีที่ใช้กันอย่างแพร่หลายคือการโจมตีแบบฟลัดดิ้งไปยังอุปกรณ์หรือเครื่องคอมพิวเตอร์ ทำให้อุปกรณ์หรือเครื่องคอมพิวเตอร์ดังกล่าวถูกใช้ทรัพยากร (ได้แก่ หน่วยประมวลผล, หน่วยความจำ, อุปกรณ์จัดเก็บข้อมูลและช่องทางรับส่งข้อมูล) เป็นจำนวนมากจนไม่สามารถให้บริการได้ตามปกติ ข้อแตกต่างระหว่างการโจมตีแบบ DoS และการโจมตีแบบ DDoS คือ การโจมตีแบบ DoS มีเส้นทางในการโจมตีเพียงเส้นทางเดียว ส่วนการโจมตีแบบ DDoS มีเส้นทางในการโจมตีมาจากหลายเส้นทาง

การตรวจจับการโจมตีแบบ DoS และ DDoS สามารถแบ่งวิธีการตรวจจับตามลักษณะการทำงานได้ 2 วิธี วิธีแรกเป็นวิธีตรวจจับการใช้งานที่ผิด (Misuse Detection) วิธีนี้ต้องทราบถึงสัญลักษณ์ (Signature) ของการโจมตี ถ้าตรวจพบสัญลักษณ์ดังกล่าวในระบบเครือข่ายคอมพิวเตอร์

ก็ส่งสัญญาณเตือนว่ามีการโจมตีเกิดขึ้น วิธีตรวจจับแบบนี้มีข้อจำกัดตรงที่ตรวจจับได้เฉพาะการโจมตีที่รู้จักสัญลักษณ์และจำเป็นต้องมีการปรับปรุงสัญลักษณ์เมื่อมีการโจมตีแบบใหม่เกิดขึ้น วิธีที่สองคือวิธีตรวจจับการใช้งานที่ผิดปกติ (Anomaly Detection) วิธีนี้ต้องเรียนรู้และจำลองพฤติกรรมการใช้งานที่ปกติก่อนและจัดเก็บไว้เป็นข้อมูลอ้างอิง เมื่อตรวจพบการใช้งานที่แตกต่างจากแบบจำลองการใช้งานที่ปกติก็ส่งสัญญาณเตือนว่ามีการโจมตีเกิดขึ้น วิธีการตรวจจับแบบนี้มีข้อดีกว่าการตรวจจับแบบแรกตรงที่สามารถตรวจจับการโจมตีแบบใหม่ได้โดยไม่จำเป็นต้องรู้ถึงสัญลักษณ์ของการโจมตี

มีงานวิจัยจำนวนมากได้นำเสนอการตรวจจับการโจมตีแบบ DoS และ DDoS ในหลาย ๆ วิธี อาทิเช่น การตรวจจับการโจมตีโดยใช้เทคนิค Data Mining, การตรวจจับการโจมตีโดยใช้เทคนิคทางสถิติเป็นต้น แต่วิธีการตรวจจับการโจมตีดังกล่าวเป็นการตรวจจับที่ระบบเครือข่ายของเครื่องเป้าหมาย ซึ่งมีจำนวนชุดข้อมูลจำนวนมากพอจะเห็นความแตกต่างกับการใช้งานแบบปกติอย่างเห็นได้ชัด ดังนั้นความท้าทายของการตรวจจับที่ระบบเครือข่ายต้นทางของการโจมตีจึงอยู่ที่การแยกความแตกต่างระหว่างชุดข้อมูลที่ใช้งานปกติกับชุดข้อมูลที่เกิดจากการโจมตี โดยชุดข้อมูลดังกล่าวจะมีความใกล้เคียงกันมาก งานวิจัยนี้ได้นำเสนอวิธีตรวจจับการโจมตีแบบ DoS หรือ DDoS ที่ใช้วิธีส่งชุดข้อมูลจำนวนมากไปยังเครื่องเป้าหมาย โดยเสนอการตรวจจับที่ขาออกของระบบเครือข่าย ซึ่งการตรวจจับดังกล่าวช่วยให้ทราบว่ามีเครื่องคอมพิวเตอร์ลูกข่ายภายในระบบเครือข่ายกำลังโจมตีแบบ DoS หรือ DDoS ไปยังเครื่องคอมพิวเตอร์ภายนอกหรือไม่ วิธีนี้จะช่วยให้เราสามารถตรวจจับการโจมตีได้ตั้งแต่ต้นทางของเส้นทางการโจมตี และสามารถตรวจจับการโจมตีได้ใกล้กับผู้โจมตีได้มากที่สุด

วัตถุประสงค์

1. ศึกษาและพัฒนาวิธีตรวจจับการโจมตีแบบฟลัดดิ้งที่เครือข่ายต้นทาง เพื่อพิจารณาผลการตรวจจับในแง่ของ

1.1. จำนวนชุดข้อมูลต่อเวลาที่สามารถตรวจจับได้

1.2. ช่วงระยะเวลาการโจมตีแบบฟลัดดิ้งที่สามารถตรวจจับได้

1.3. ช่วงระยะเวลาหน่วงระหว่างเวลาที่เกิดการโจมตีแบบฟลัดดิ้งกับเวลาที่ระบบสามารถตรวจจับได้

2. ศึกษา วิเคราะห์และประยุกต์การแปลงเวฟเล็ตเพื่อนำมาใช้ในการตรวจจับการโจมตีแบบฟลัดดิ้งที่เครือข่ายต้นทาง

การตรวจเอกสาร

มีงานวิจัยจำนวนมากเสนอแนวทางในการตรวจจับความผิดปกติภายในระบบเครือข่ายคอมพิวเตอร์ (Peng *et al.*, 2007) โดยแนวทางที่เสนอเป็นการตรวจจับการใช้งานที่ผิดปกติ (Anomaly Detection) เนื่องจากสามารถตรวจจับความผิดปกติรูปแบบใหม่ ๆ ได้ ต่างจากการตรวจจับการใช้งานที่ผิด (Misuse Detection) ที่สามารถตรวจจับได้เฉพาะความผิดปกติที่เคยเกิดขึ้นมาก่อนแล้วเท่านั้น งานวิจัยดังกล่าวได้เสนอวิธีการหลากหลายในการตรวจจับ เช่น การใช้เหมืองข้อมูล (Data Mining), การใช้สถิติ (Statistical) หรือการประมวลผลสัญญาณ (Signal Processing) เป็นต้น การตรวจเอกสารจึงตรวจเฉพาะงานวิจัยที่มีการเสนอวิธีที่ใกล้เคียงกับวิธีที่เสนอในงานวิจัยนี้ ซึ่งได้แก่การใช้สถิติและการประมวลผลสัญญาณ

ในปี 2001 ได้มีงานวิจัยเสนอวิธีตรวจจับความผิดปกติภายในระบบเครือข่ายคอมพิวเตอร์ที่เกิดจากการโจมตีแบบ Denial-of-Service (Blazek *et al.*, 2001) โดยใช้วิธีทางสถิติมาวิเคราะห์ข้อมูลโปรโตคอลแบบหลายระดับในระบบเครือข่าย เพื่อตรวจจับความเปลี่ยนแปลงจำนวนชุดข้อมูลต่อช่วงเวลาภายในระบบเครือข่ายที่เกิดจากการโจมตี ซึ่งใช้วิธีที่เรียกว่า Adaptive Sequential และ Batch-Sequential ในการตรวจจับ ข้อมูลที่ใช้ในการทดลองได้มาจากโปรแกรมจำลองชื่อ Network Simulator (NS) แต่เนื่องจากการตรวจจับด้วยวิธีดังกล่าวยังมีช่วงเวลาหน่วง (Delay) ระหว่างเวลาที่เริ่มโจมตีและเวลาที่ตรวจจับได้ ดังนั้นวิธีนี้จึงอาจไม่เหมาะกับระบบเครือข่ายที่ต้องการความถูกต้องแม่นยำในการตรวจจับระหว่างเวลาที่เริ่มโจมตีกับเวลาที่ตรวจจับได้

ต่อมาในปี 2002 ได้มีงานวิจัยเสนอวิธีตรวจจับความผิดปกติภายในระบบเครือข่าย (Barford *et al.*, 2002) ด้วยการใช้วิธีประมวลผลสัญญาณ (Signal Processing) โดยทำการเก็บข้อมูลการไหลของโปรโตคอลชั้นอินเตอร์เน็ต (IP Flow) และข้อมูลที่ได้จากโปรโตคอลที่ใช้ในการจัดการระบบเครือข่าย (SNMP) หลังจากนั้นจึงนำข้อมูลดังกล่าวมาผ่านกระบวนการแยกส่วนประกอบสัญญาณด้วยวิธีเวฟเล็ตให้อยู่ในรูปส่วนประกอบสัญญาณ 3 ความถี่ ได้แก่ ความถี่สูง, ความถี่กลางและความถี่ต่ำ และทำการวิเคราะห์ส่วนประกอบสัญญาณทั้ง 3 ความถี่เพื่อตรวจจับหาความผิดปกติที่เกิดขึ้นจากความผิดพลาดของอุปกรณ์ (Outages), การใช้งานที่มากเกินปกติ (Flash Crowd) และการโจมตี (Attack)

ต่อมาได้มีงานวิจัยเสนอวิธีตรวจจับการโจมตี Denial-of-Service ในปี 2006 (Carl *et al.*, 2006) ซึ่งเป็นการปรับปรุงกระบวนการตรวจจับความเปลี่ยนแปลงจำนวนชุดข้อมูลต่อช่วงเวลา

(Blazek *et al.*, 2001) โดยนำผลลัพธ์ที่ได้จากการวิเคราะห์ข้อมูลโปรโตคอลแบบหลายระดับในระบบเครือข่ายมาแยกส่วนประกอบสัญญาณด้วยวิธีเวฟเล็ต เพื่อตรวจจับความเปลี่ยนแปลงในระบบเครือข่ายที่เกิดจากการโจมตี งานวิจัยนี้ได้ทำการเก็บข้อมูลที่ใช้ในการทดลองได้มาจากโปรแกรมจำลองชื่อ Network Simulator 2 (NS-2) และจากข้อมูลการใช้งานภายในระบบเครือข่ายจริง ผลของงานวิจัยดังกล่าวแสดงให้เห็นว่าวิธีที่นำเสนอสามารถนำไปใช้ได้ในระบบเครือข่ายจริง และสามารถลดข้อผิดพลาดที่เกิดการโจมตีแล้วตรวจจับไม่พบ (False Negative) ลงได้

ในปี 2008 ได้มีงานวิจัยที่ทำการศึกษาเกี่ยวกับวิธีการตรวจจับความผิดปกติภายในเครือข่ายที่เกิดจากการโจมตีแบบ Denail-of-Service ที่มีอัตราการส่งข้อมูลต่ำ (Thatte *et al.*, 2008) โดยข้อมูลที่ได้มาจากจำนวนชุดข้อมูลต่อช่วงเวลาภายในระบบเครือข่าย จากนั้นจึงนำข้อมูลดังกล่าวไปคำนวณค่า Spectrum และ Entropy ของข้อมูลในแต่ละช่วงเวลาเพื่อหาการเปลี่ยนแปลงในระบบเครือข่ายที่เกิดจากการโจมตีแบบ Denail-of-Service ในงานวิจัยนี้ได้ทำการทดลองจากข้อมูลจำลองและจากข้อมูลที่เก็บมาจากระบบเครือข่ายจริง

และในปี 2008 ได้มีงานวิจัยทำการศึกษาเกี่ยวกับเวฟเล็ตแม่แบบต่าง ๆ ว่ามีผลกระทบกับการตรวจจับความผิดปกติภายในระบบเครือข่ายอย่างไร (Lu *et al.*, 2008) และได้เสนอวิธีในการตรวจจับความผิดปกติภายในระบบเครือข่าย โดยการใช้วิธีเชิงเวฟเล็ตและการประมาณค่าแบบถดถอยในการตรวจจับความผิดปกติ ในงานวิจัยนี้ได้ใช้ข้อมูลทดสอบจาก DARPA ปี 1999 มาใช้ในการทดลอง ในการทดลองได้ทำการทดสอบเวฟเล็ตแม่แบบต่าง ๆ ทั้งหมด 4 แบบ ได้แก่ Daubechies1, Coiflets1, Symlets2 และ Discrete Meyer จากการทดลองพบว่าเวฟเล็ตแม่แบบ Daubechies1 มีประสิทธิภาพในการตรวจจับความผิดปกติมากกว่าเวฟเล็ตแม่แบบอื่นอีก 3 แบบ

เห็นได้ว่าจากงานวิจัยที่ผ่านมาได้มีการเสนอวิธีในการตรวจจับความผิดปกติภายในระบบเครือข่าย ซึ่งส่วนใหญ่สามารถตรวจจับความผิดปกติในเครือข่ายที่เกิดการเปลี่ยนแปลงแบบทันทีทันใด ในงานวิจัยนี้จึงเสนอแนวทางในการตรวจจับความผิดปกติในเครือข่ายที่เกิดการเปลี่ยนแปลงแบบทันทีทันใด (Abrupt Change) และที่เกิดจากการเปลี่ยนแปลงแบบค่อยเป็นค่อยไป (Long Time Change) ซึ่งสามารถเปรียบเทียบกับวิธีอื่น ๆ ที่กล่าวมาแล้วข้างต้นได้ตามตารางที่ 1

จากตารางเปรียบเทียบการตรวจจับความผิดปกติภายในระบบเครือข่ายข้างต้น แม้ว่าจะใช้หลักการของวิธีเชิงเวฟเล็ตเหมือนกัน แต่การนำไปใช้ของวิธีที่เสนอยังมีความแตกต่างจากงานวิจัย

ข้างต้น ดังนั้นเพื่อให้เข้าใจถึงหลักการทำงานของวิธีที่เสนอในงานวิจัยนี้ จึงขอกล่าวถึงหลักการทำงานพื้นฐานที่สำคัญ 2 ประการที่ใช้ในการวิจัยนี้ คือ การแปลงเวฟเล็ตและการหาค่าสหสัมพันธ์

ตารางที่ 1 เปรียบเทียบวิธีการตรวจจับความผิดปกติ

Name	Data Source	Approach	Detect	Remark
Blazek'01	N of Packet	Statistics	Abrupt Change	Synthetic Trace
Barford'02	IP Flow, SMTP	Wavelet	Outages, Flash Crowed, Attack	DWT 3 Level
Carl'06	CUSUM	Wavelet	Abrupt Change	Synthetic and Live Trace
Thatte'08	N of Packet	Spectrum, Entropy	Abrupt Change	Live Trace
Lu'08	N of Flow	Wavelet	Abrupt Change	Compare Mother of Wavelet
Proposed	N of Packet	Wavelet	Abrupt Change, Longtime	Live Trace and DWT 10 Level

1. ทฤษฎีพื้นฐานของการแปลงเวฟเล็ต

เวฟเล็ต (Wavelet) เป็นฟังก์ชันทางคณิตศาสตร์ใช้ในการเปลี่ยนรูปสัญญาณที่อยู่ในรูปของฟังก์ชันเวลาให้เป็นส่วนประกอบของสัญญาณที่มีความถี่ที่แตกต่างกัน และทำการศึกษาแต่ละส่วนประกอบด้วยความละเอียดที่ตรงกันในแต่ละสเกล การแปลงเวฟเล็ตถูกนำมาใช้มากในการประมวลผลสัญญาณ (Signal Processing) ซึ่งวิธีการดังกล่าวได้มีพัฒนามาจากการประมวลผลสัญญาณพื้นฐานที่มีอยู่เดิม เช่น การแปลงฟูเรียร์ การแปลงกาบอร์ เป็นต้น และปรับปรุงให้มีรูปแบบที่เหมาะสมกับการใช้งานเฉพาะทางมากขึ้น ในหัวข้อนี้เป็นการอธิบายทฤษฎีพื้นฐานและความแตกต่างระหว่างวิธีการแปลงเวฟเล็ตกับวิธีการแปลงสัญญาณแบบอื่น ๆ เพื่อเป็นความรู้พื้นฐานของงานวิจัยในวิทยานิพนธ์ฉบับนี้

1.1 การแปลงฟูเรียร์ (Fourier Transform : FT)

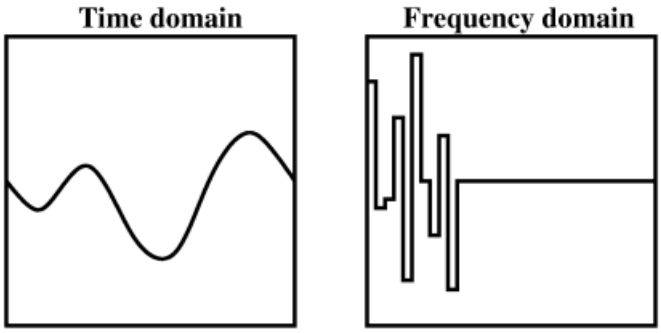
เป็นรูปแบบการประมวลผลสัญญาณที่นิยมกันอย่างแพร่หลาย โดยเป็นการแปลงสัญญาณจากเขตของเวลา (Time Domain) ไปเป็นเขตของความถี่ (Frequency Domain) เป็นการวิเคราะห์สัญญาณโดยอาศัยฟังก์ชันพื้นฐาน (Basic Function) ที่อยู่ในรูปของฟังก์ชันเลขชี้กำลัง (Exponential) การแปลงฟูเรียร์จะแบ่งส่วนประกอบของสัญญาณใด ๆ ให้อยู่ในรูปของฟังก์ชันโคไซน์ (Cosine) และฟังก์ชันไซน์ (Sine) ที่มีขนาดและความถี่ที่แตกต่างกันตลอดย่านความถี่ $-\infty$ ถึง ∞ ดังแสดงในสมการ (1)

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}dt \qquad (1)$$

และมีสมการของการวิเคราะห์แบบสุ่ม (Sampling Data) ที่เรียกว่า “การแปลงฟูเรียร์แบบเร็ว” (Fast Fourier Transform : FFT) ดังแสดงในสมการ (2)

$$F(n) = \sum_{k=0}^{N-1} f(k)e^{-\left(\frac{j2\pi kn}{N}\right)} \qquad (2)$$

เมื่อ $n = 1,2,\cdots N$



ภาพที่ 1 การแปลงสัญญาณด้วยฟูเรียร์จากเขตของเวลา (ซ้าย) ไปเป็นเขตของความถี่ (ขวา)

ที่มา: วิกิพีเดีย (2552)

ภาพที่ 1 เป็นตัวอย่างการแปลงสัญญาณจากเขตของเวลาไปเป็นเขตของความถี่ด้วยการแปลงฟูเรียร์ ซึ่งการวิเคราะห์สัญญาณด้วยการแปลงฟูเรียร์จะมีความแม่นยำด้านความถี่และมีความเหมาะสมในการวิเคราะห์สัญญาณที่มีลักษณะเป็นคาบเวลาที่แน่นอน (Stationary Signal) แต่เมื่อนำไปวิเคราะห์สัญญาณที่มีการเปลี่ยนแปลงเป็นคาบเวลาที่ไม่แน่นอน (Non-Stationary Signal) ทำให้ขาดข้อมูลด้านเวลาและทำให้เกิดข้อผิดพลาดของสัญญาณเมื่อทำการแปลงสัญญาณกลับ (Invert Transform) จากเขตของความถี่กลับมาเป็นเขตของเวลา ซึ่งข้อมูลด้านเวลาเป็นข้อมูลที่มีความสำคัญมากในการวิเคราะห์สัญญาณที่มีลักษณะเป็นคาบเวลาที่ไม่แน่นอน ตัวอย่างเช่น สัญญาณเสียง สัญญาณคลื่นในสายอากาศ เป็นต้น งานวิจัยในวิทยานิพนธ์ฉบับนี้สัญญาณที่ได้จากระบบเครือข่ายเป็นสัญญาณที่มีคาบเวลาไม่แน่นอนดังนั้นจึงไม่สามารถใช้การแปลงฟูเรียร์ได้

1.2 การแปลงกาบอร์ (Gabor Transform : GT)

เป็นการแปลงสัญญาณที่ได้พัฒนาขึ้นมาเพื่อแก้ปัญหาของการแปลงฟูเรียร์ โดยก่อนทำการแปลงฟูเรียร์ของสัญญาณใด ๆ ให้นำสัญญาณดังกล่าวมาคูณด้วยฟังก์ชันหน้าต่าง (Window Function) ที่มีลักษณะเป็นฟังก์ชันแบบเกาส์เซียน ผลของสัญญาณที่ได้จากการแปลงกาบอร์สามารถระบุถึงความถี่และเฟสของสัญญาณได้ การแปลงกาบอร์สามารถเขียนเป็นสมการคณิตศาสตร์ได้ดังแสดงในสมการ (3)

$$G_x(t,f) = \int_{-\infty}^{\infty} e^{-\pi(\tau-t)^2} e^{-j2\pi f\tau} x(\tau)d\tau \qquad (3)$$

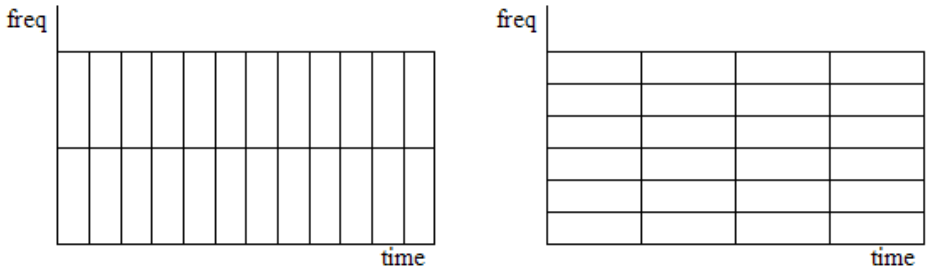
จากสมการการแปลงกาบอร์เห็นได้ว่าการแปลงสัญญาณรูปแบบนี้สามารถเลือกตำแหน่งของเวลาที่ทำการวิเคราะห์ได้โดยการกำหนดพารามิเตอร์ t และสามารถเลือกความถี่ที่ต้องการวิเคราะห์ด้วยพารามิเตอร์ f แต่เนื่องจากการแปลงกาบอร์มีฟังก์ชันหน้าต่างแบบเดียวจึงไม่เหมาะสมกับการประยุกต์ใช้งานในการวิเคราะห์สัญญาณในหลาย ๆ รูปแบบได้

1.3 การแปลงฟูเรียร์ช่วงเวลาสั้น (Short-Time Fourier Transform : STFT)

จากข้อจำกัดของการแปลงฟูเรียร์และการแปลงกาบอร์ดังกล่าวจึงได้มีการพัฒนารูปแบบการวิเคราะห์สัญญาณมาเป็นการแปลงฟูเรียร์ช่วงเวลาสั้น ซึ่งเป็นการวิเคราะห์ที่ใช้ฟังก์ชันหน้าต่างเหมือนการแปลงกาบอร์แต่สามารถเลือกฟังก์ชันหน้าต่างได้ ทำให้มีความยืดหยุ่นในการวิเคราะห์สัญญาณมากยิ่งขึ้น สมการการแปลงฟูเรียร์ช่วงเวลาสั้นสามารถเขียนเป็นสมการคณิตศาสตร์ได้ดังแสดงในสมการ (4)

$$STFT\{x(t)\} \equiv X(\tau,\omega) = \int_{-\infty}^{\infty} x(t)w(t-\tau)e^{-j\omega t}dt \qquad (4)$$

เมื่อ $w(t)$ เป็นฟังก์ชันหน้าต่างที่ใช้ในการวิเคราะห์ โดยที่ตำแหน่งเวลาที่ทำการวิเคราะห์ถูกกำหนดโดยค่าพารามิเตอร์ τ และความถี่ที่ต้องการวิเคราะห์ถูกกำหนดด้วยความถี่หรือความกว้างของฟังก์ชันหน้าต่างนั้น ๆ ผลที่ได้จากการวิเคราะห์จะอยู่ในรูปของการแบ่งสัญญาณในลักษณะการแปลงฟูเรียร์ในช่วงเวลาและความถี่ที่ทำการวิเคราะห์ดังแสดงในภาพที่ 2



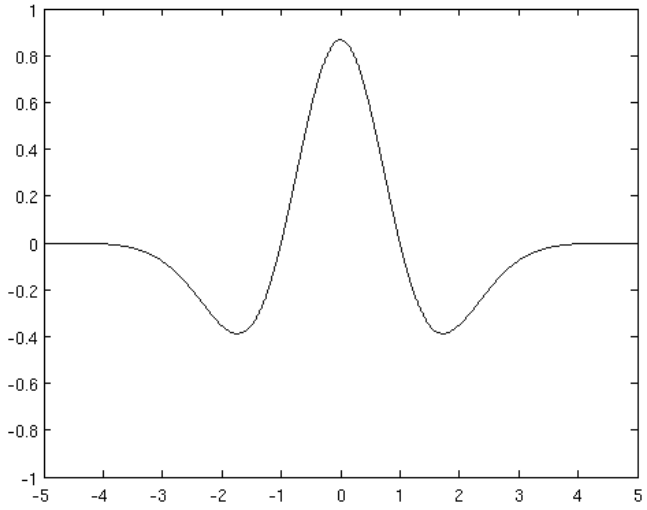
ภาพที่ 2 การแปลงฟูเรียร์ช่วงเวลาสั้นโดยพิจารณาความละเอียดด้านเวลา (ซ้าย) หรือพิจารณาความละเอียดด้านความถี่ (ขวา)

ที่มา: วิกิพีเดีย (2552)

การแปลงกาบอร์และการแปลงฟูเรียร์ช่วงเวลาสั้นมีลักษณะของช่วงเวลาการวิเคราะห์ที่คงที่ (Fixed Resolution Transform) ดังนั้นการใช้ช่วงเวลาการวิเคราะห์ที่คงที่ในการวิเคราะห์สัญญาณทุก ๆ ช่วงความถี่ จึงอาจไม่เหมาะสมในการวิเคราะห์สัญญาณในทางปฏิบัติ เนื่องจากสัญญาณที่มีความถี่สูงจะมีการเปลี่ยนแปลงที่รวดเร็วจึงควรใช้ช่วงเวลาที่แคบในการวิเคราะห์ ในขณะที่สัญญาณที่มีความถี่ต่ำจะมีการเปลี่ยนแปลงที่ช้าจึงควรใช้ช่วงเวลาที่กว้างในการวิเคราะห์ จากเหตุผลนี้จึงได้มีการพัฒนารูปแบบการวิเคราะห์สัญญาณที่มีการปรับระดับความละเอียดในการวิเคราะห์ที่เรียกว่า “การแปลงเวฟเล็ต”

2. การแปลงเวฟเล็ต (Wavelet Transform : WT)

ทฤษฎีเวฟเล็ต (Wavelet Theory) กล่าวถึงฟังก์ชันทางคณิตศาสตร์ที่ใช้อธิบายลักษณะของสัญญาณที่ประกอบด้วยกลุ่มของสัญญาณเฉพาะที่มีหลาย ๆ ขนาดมารวมกัน โดยสัญญาณเฉพาะดังกล่าวมีลักษณะเป็นคลื่นเล็ก ๆ ที่เรียกว่า “เวฟเล็ต” ซึ่งมีลักษณะเป็นคลื่นที่เพิ่มจากศูนย์อย่างรวดเร็ว มีการเปลี่ยนแปลงอย่างต่อเนื่อง (Oscillatory) และขนาดของคลื่นจะลดลงสู่ศูนย์อย่างรวดเร็ว ตัวอย่างของเวฟเล็ตดังแสดงในภาพที่ 3



ภาพที่ 3 เวฟเล็ตแบบ Mexican Hat

ที่มา: วิกิพีเดีย (2552)

2.1 การแปลงเวฟเล็ตแบบต่อเนื่อง (Continuous Wavelet Transform : CWT)

รูปแบบของการแปลงเวฟเล็ตแบบต่อเนื่องนั้นมีลักษณะการวิเคราะห์สัญญาณโดยอาศัยการปรับเปลี่ยนคุณสมบัติของเวฟเล็ตแม่ที่ใช้ในการวิเคราะห์ให้มีความต่อเนื่องกัน โดยการกำหนดสเกลของเวฟเล็ตแม่ให้มีค่าน้อยเพื่อใช้ในการวิเคราะห์ส่วนประกอบความถี่สูง และกำหนดสเกลของเวฟเล็ตแม่ให้มีค่ามากเพื่อใช้ในการวิเคราะห์ส่วนประกอบความถี่ต่ำ ซึ่งก็เป็นการปรับระดับความละเอียดในการวิเคราะห์อย่างต่อเนื่องให้เหมาะสมกับความถี่ที่ทำการวิเคราะห์ การแปลงเวฟเล็ตแบบต่อเนื่องสามารถเขียนเป็นสมการคณิตศาสตร์ได้ดังแสดงในสมการ (5)

$$CWT(a,b) \equiv X_w(a,b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(t)\psi\left(\frac{t-b}{a}\right) dt \qquad (5)$$

เมื่อ $x(t)$ สัญญาณที่ต้องการวิเคราะห์

$\psi(t)$ เวฟเล็ตแม่

a พารามิเตอร์สเกล

b พารามิเตอร์ตำแหน่งของเวลา

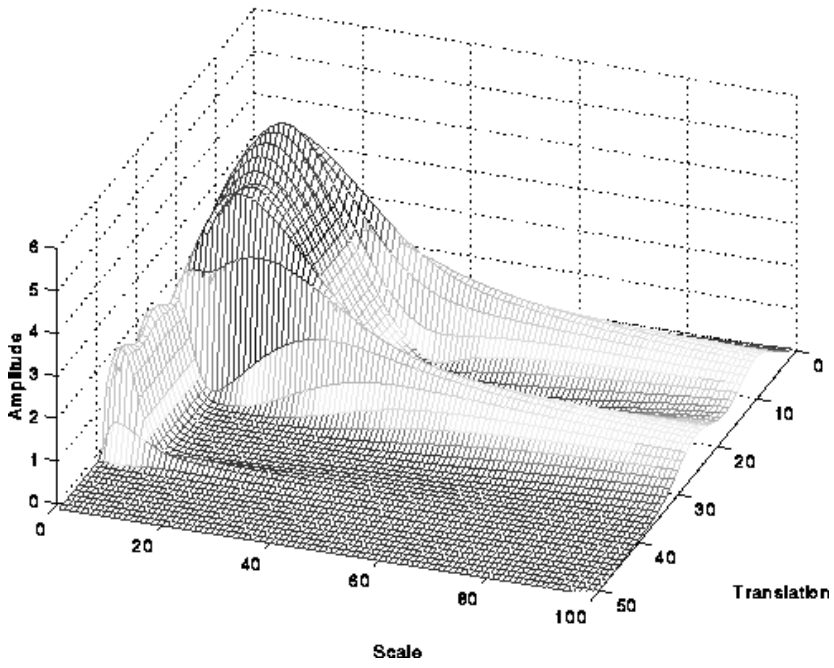
จากสมการในพจน์ของ $\psi(t)$ เป็นพจน์ของเวฟเล็ตแม่ที่เทียบได้กับพจน์ของฟังก์ชันหน้าต่างในการแปลงฟูเรียร์ช่วงเวลาสั้นนั่นเอง แต่เมื่อพิจารณาพจน์ของ $\psi(t)$ จะมีการเปลี่ยนคุณสมบัติของสเกลตามพารามิเตอร์ a และพารามิเตอร์ของการเลื่อนตำแหน่ง b ซึ่งมีขั้นตอนในการแปลงสัญญาณดังนี้

1. นำเวฟเล็ตแม่มาเปรียบเทียบกับสัญญาณที่ต้องการวิเคราะห์ที่จุดเริ่มต้นสัญญาณ
2. หาค่าสัมประสิทธิ์ c ซึ่งคำนวณจากความสัมพันธ์ (Correlation) ระหว่างเวฟเล็ตแม่กับสัญญาณที่ต้องการวิเคราะห์ ซึ่งค่าสัมประสิทธิ์ดังกล่าวขึ้นอยู่กับรูปร่างของเวฟเล็ตแม่ที่นำมาใช้ด้วย
3. เลื่อนตำแหน่งเวฟเล็ตแม่ไปทางขวาและทำซ้ำในขั้นตอนที่ 2 จนกระทั่งครอบคลุมสัญญาณทั้งหมด ซึ่งจากขั้นตอนที่ 1-3 นี้เป็นการแปลงเวฟเล็ตตลอดช่วงสัญญาณในสเกลแรก
4. เปลี่ยนสเกลของเวฟเล็ตแม่และทำตามขั้นตอนที่ 1-3 ใหม่
5. ทำตามขั้นตอนที่ 1-4 จนกระทั่งครบทุกสเกล

จากขั้นตอนทั้งหมดผลการวิเคราะห์จะอยู่ในรูปสัมประสิทธิ์ที่ได้จากการวิเคราะห์ของแต่ละสเกลในแต่ละส่วนของสัญญาณ และเนื่องจากการแปลงเวฟเล็ตเป็นไปในลักษณะที่มีการ

เปลี่ยนแปลงสเกลและการเลื่อนตำแหน่งในการวิเคราะห์อย่างต่อเนื่อง ผลลัพท์ที่ได้จึงมีลักษณะเป็นพื้นที่ที่มีความต่อเนื่องกัน ภาพที่ 4 เป็นตัวอย่างที่แสดงถึงผลการวิเคราะห์ทั้งหมดมาแสดงรูปแบบความสัมพันธ์ระหว่างสเกลและการเลื่อนตำแหน่ง

จากภาพที่ 4 แสดงให้เห็นว่าผลการแปลงเวฟเล็ตเมื่อนำมาเขียนเป็นกราฟจะแสดงในรูปของพื้นผิวที่ต่อเนื่องกัน ซึ่งการปรับเปลี่ยนสเกลในการวิเคราะห์ที่ต่อเนื่องจะทำให้ความแม่นยำทางด้านเวลาและความถี่ที่ดี แต่จะมีข้อเสียคือการนำไปใช้ในการวิเคราะห์สัญญาณจริงต้องการความรวดเร็วในการวิเคราะห์ การแปลงเวฟเล็ตแบบต่อเนื่องไม่เหมาะสมในการนำไปวิเคราะห์สัญญาณ เนื่องจากต้องใช้เวลาในการประมวลผลค่อนข้างมากและข้อมูลบางส่วนมีความซ้ำซ้อนเกินความจำเป็น



ภาพที่ 4 การแปลงเวฟเล็ตแบบต่อเนื่อง

ที่มา: Rowan University (2552)

2.2 การแปลงเวฟเล็ตแบบเต็มหน่วย (Discrete Wavelet Transform : DWT)

จากข้อจำกัดของการแปลงเวฟเล็ตแบบต่อเนื่องจึงมีการพัฒนารูปแบบการแปลงเวฟเล็ตมาเป็นการแปลงเวฟเล็ตแบบเต็มหน่วย ที่มีลักษณะการวิเคราะห์โดยการเปลี่ยนสเกลและการ

เลื่อนตำแหน่งในลักษณะเป็นช่วง ๆ ไม่ต่อเนื่องกัน แต่ก่อนที่จะกล่าวถึงรายละเอียดของการแปลงเวฟเล็ตแบบเต็มหน่วยจำเป็นต้องกล่าวถึงทฤษฎีพื้นฐานที่ใช้ในการอธิบายก่อน

2.2.1 การวิเคราะห์สัญญาณแบบหลายระดับความละเอียด (Multi-Resolution Analysis : MRA)

การวิเคราะห์สัญญาณแบบหลายระดับความละเอียดคือการวิเคราะห์สัญญาณที่สามารถเลือกระดับความละเอียดของสัญญาณได้ โดยการนำสัญญาณเล็ก ๆ ที่ระดับความละเอียด a ซึ่งมี b หลาย ๆ ตำแหน่งมารวมกันให้เกิดเป็นสัญญาณที่ระดับความละเอียดที่ต้องการ และเมื่อนำสัญญาณที่ทุกระดับความละเอียดมารวมกันก็จะได้สัญญาณเดิม ซึ่งหลักการของการวิเคราะห์สัญญาณหลายระดับความละเอียดนั้นสามารถอธิบายได้โดยอาศัยทฤษฎีของสเปซเวกเตอร์ (Space Vector)

2.2.2 ทฤษฎีของสเปซเวกเตอร์ (Space Vector Theory)

ความหมายของสเปซของเวกเตอร์ในการวิเคราะห์สัญญาณคือสเปซของสัญญาณที่เกิดจากการรวมกันของสัญญาณพื้นฐานย่อย ๆ ที่เรียกว่า “ฟังก์ชันพื้นฐาน” (Basic Function) โดยสมมติว่า V เป็นสเปซเวกเตอร์ที่มี j แสดงถึงระดับความละเอียดและจำนวนของฟังก์ชันพื้นฐานที่ประกอบขึ้นเป็นฟังก์ชันนั้น ซึ่งถ้า j มีค่าสูงขึ้นก็แสดงว่าที่ระดับความละเอียดสูงขึ้นและมีจำนวนฟังก์ชันพื้นฐานมากขึ้น ทำให้สัญญาณที่เกิดจากการประกอบกันจากฟังก์ชันมีความละเอียดมากขึ้นด้วย ดังนั้นอาจกล่าวได้ว่า j เป็นค่าแสดงถึงระดับความละเอียดของสัญญาณนั้นเอง โดยสามารถสรุปคุณสมบัติของการวิเคราะห์สัญญาณที่ระดับความละเอียดต่าง ๆ ได้ดังนี้

1. $V^{-\infty} \cdots \subset V^{-1} \subset V^{1} \cdots \subset V^{\infty}$
2. $Close_{L^2}\left(W_{j\in\mathbb{Z}}^{j}\right) = L^2(\mathbb{R}); \mathbb{R} =$ เซ็ตของจำนวนจริง
3. $\left(W_{j=\mathbb{Z}}^{j}\right) = \{0\}$
4. $V^{j} + W^{j} = V^{j+1}; j \in \mathbb{Z}; \mathbb{Z} =$ เซ็ตของจำนวนจริง
5. $f(x) \in V^{j} \Leftrightarrow f(2x) \in V^{j+1}; j \in \mathbb{Z}$

จากการที่ฟังก์ชันพื้นฐานประกอบกันเป็นสัญญาณที่ระดับความละเอียด j ภายในสเปซ V เรียกฟังก์ชันนี้ว่าเป็น “ฟังก์ชันสเกลลิ่ง” (Scaling Function : $\Phi(t)$) และฟังก์ชันสเกลลิ่งที่ระดับสูงจะมีความถี่สูงและระดับที่ต่ำลงมาจะมีความถี่ต่ำกว่าจากความสัมพันธ์ที่แสดงใน

คุณสมบัติข้อที่ 4 ซึ่งทำให้สามารถเขียนความสัมพันธ์ระหว่างฟังก์ชันสเกลในสเปซใด ๆ ได้ดังแสดงในสมการ (6)

$$\Phi_{j,k} = 2^{\frac{j}{2}}\Phi(2^j t - k); j, k \in \mathbb{Z} \quad (6)$$

จากสมการจะพบว่าถ้าระดับความละเอียดลดลงมาหนึ่งระดับแล้วฟังก์ชันพื้นฐานจะมีความถี่ลดลงมาครั้งละ 2 เท่า และอาศัยคุณสมบัติ MRA จะทำให้สามารถทำการประมาณสัญญาณ $f(t) \in L^2(IR)$ ไปอยู่ในสเปซที่ระดับความละเอียด j ใด ๆ ได้ดังแสดงในสมการ (7)

$$f_j(t) = \sum_k c_k^j \Phi_{j,k}(t) \quad (7)$$

โดย c_k^j ที่เป็นสัมประสิทธิ์หรือน้ำหนักที่คูณกับฟังก์ชันสเกลลิ่งที่ตำแหน่ง k ใด ๆ แล้วประกอบขึ้นเป็น $f(t)$ ที่ระดับความละเอียด j

จากคุณสมบัติข้อ 4 ของ MRA การวิเคราะห์สัญญาณที่ระดับความละเอียดต่ำลงมาจะทำให้พลังงานหรือสัญญาณบางส่วนหายไปอยู่อีกสเปซหนึ่งซึ่งเรียกว่า “สเปซของเวกเตอร์เวฟเล็ต” (Wavelet Vector Space : W^j) ซึ่งจะประกอบด้วยฟังก์ชันพื้นฐานที่เรียกว่า “ฟังก์ชันเวฟเล็ต” (Wavelet Function : $\psi(t)$) โดยสามารถเขียนฟังก์ชันเวฟเล็ตได้ดังแสดงในสมการ (8)

$$\psi_{j,k}(t) = 2^{\frac{j}{2}}\psi(2^j t - k) \quad (8)$$

ถ้ากำหนดให้ $g(t)$ เป็นสัญญาณที่เกิดจากฟังก์ชันพื้นฐาน $\psi_{j,k}(t)$ ภายในสเปซเดียวกันมารวมเป็นสัญญาณใด ๆ จะได้ดังแสดงในสมการ (9)

$$g(t) = \sum_k d_k^j \psi_{j,k}(t) \quad (9)$$

โดยที่ d_k^j เป็นค่าสัมประสิทธิ์หรือค่าน้ำหนักที่คูณกับฟังก์ชันเวฟเล็ตที่ตำแหน่งนั้น ๆ เพื่อเกิดเป็นสัญญาณ $g(t)$ ดังนั้นจากความสัมพันธ์ $V^j + W^j = V^{j+1}$ จะได้ดังแสดงในสมการ (10)

$$f_{j+1} = f_j + g_j \tag{10}$$

โดยสมมติให้ $f(t) \in V^{j+1}$ จะสามารถแตกกระจายให้ $f(t)$ ให้มีความละเอียดน้อยลงได้จากความสัมพันธ์ของ $V^j + W^j = V^{j+1}$ ในขณะเดียวกัน V^j สามารถแตกต่อไปได้อีกเรื่อย ๆ จนกระทั่ง $j = 0$ ดังนั้นเขียนความสัมพันธ์ได้ดังแสดงในสมการ (11)

$$V^{j+1} = V^0 + W^0 + W^1 + \cdots + W^j \tag{11}$$

เช่นเดียวกัน $f_{j+1} = f_j + g_j$ สามารถแตกกระจายเป็น f_j และ g_j ได้ ดังนั้นเราสามารถแสดงลักษณะของสัญญาณ $f(t)$ ในรูปของฟังก์ชันสเกลลิ่งและฟังก์ชันเวฟเล็ตได้ดังแสดงในสมการ (12)

$$f(t) = f_j + g_j + g_{j+1} + g_{j+2} + \cdots + g_\infty \tag{12}$$

รูปแบบการแตกกระจายสัญญาณ $f(t)$ ใด ๆ ในสเปซ V^0 ไปจนถึงระดับความละเอียดที่ j ในรูปของสัมประสิทธิ์ $c_j(m)$ และ $d_j(m)$ นี้เรียกว่า “การแปลงเวฟเล็ตแบบเต็มหน่วย” (Discrete Wavelet Transform) โดยสามารถเขียนได้ดังแสดงในสมการ (13)

$$DWT(m,n) = \frac{1}{\sqrt{a_0^m}} \int_{-\infty}^{\infty} f(t)\psi\left(\frac{t - nb_0a_0^m}{a_0^m}\right) dt \tag{13}$$

เมื่อ a_0^m คือการสเกล

$nb_0a_0^m$ คือการเลื่อนตำแหน่ง

m, n คือเลขจำนวนเต็มบวก

และการนำมาใช้งานจริงในทางปฏิบัติ สัญญาณที่เข้ามาจะอยู่ในรูปของการสุ่ม (Sample) ดังนั้นจากสมการข้างต้น จึงพัฒนามาดังแสดงในสมการ (14)

$$DWT(m,n) = \frac{1}{\sqrt{a_0^m}} \Sigma_k f(t)\psi\left(\frac{n - kb_0a_0^m}{a_0^m}\right) \tag{14}$$

เมื่อ m, n, k เป็นเลขจำนวนเต็มโดยที่

n คือจำนวนข้อมูล

m คือการสเกล

k คือการเลื่อนตำแหน่ง

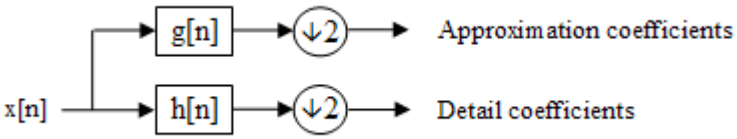
เมื่อพิจารณาในรูปแบบของการวิเคราะห์หลายระดับความละเอียดโดยมีการเปลี่ยนแปลงสเกลในการวิเคราะห์ให้ลดลงครั้งละ 2 เท่า $(a_0 = 2; b_0 = 1)$ แล้วจะได้รูปแบบการแปลงเวฟเล็ตแบบเต็มหน่วย ซึ่งมีชื่อเรียกว่า Dyadic Wavelet Transform ดังแสดงในสมการ (15)

$$DWT(m,n) = \frac{1}{\sqrt{2^m}} \sum_k f(t)\psi\left(\frac{n-k2^m}{2^m}\right) \quad (15)$$

ซึ่งเราสามารถอธิบายหลักการทำงานได้โดยอาศัยหลักการวิเคราะห์ด้วยตัวกรองสัญญาณ (Filter Bank Analysis)

2.2.3 การวิเคราะห์ด้วยตัวกรองสัญญาณ (Filter Bank Analysis)

ก่อนที่จะอธิบายวิธีสร้างการแปลงเวฟเล็ตด้วยตัวกรองสัญญาณนั้น ขออธิบายถึงหลักการพื้นฐานของตัวกรองสัญญาณด้วยตัวกรองสัญญาณแบบ 2 ช่องสัญญาณ (Two Channel Filter Bank) จะเป็นการแยกสัญญาณอินพุทออกเป็น 2 ส่วนคือ ส่วนของความถี่ต่ำและส่วนของความถี่สูง โดยจะถูกวิเคราะห์ด้วยคู่ชุดกรองซึ่งมีการลดความถี่ออกมาครึ่งหนึ่งดังรูป ซึ่งในพจน์ของ แสดงการลดอัตราการสุ่มลง (Down Sampling) ด้วย 2 หรือลดผลการวิเคราะห์จากตัวกรองออกครึ่งหนึ่งในแต่ละขั้นตอนของการวิเคราะห์ เมื่อนำสัญญาณอินพุทมาผ่านการแปลงเวฟเล็ตแบบเต็มหน่วย สัญญาณจะถูกแยกส่วนประกอบ (Decomposition) โดยตัวกรองแบบ 2 ช่อง ซึ่งจะแยกความถี่ในช่วงความถี่ที่ต้องการออกเป็น 2 ส่วนคือ

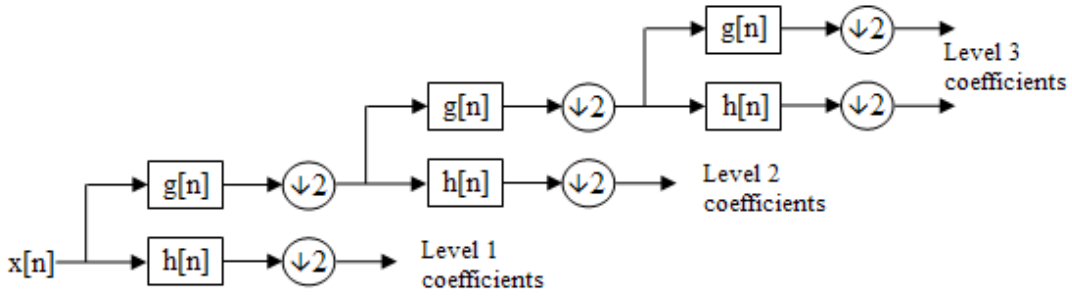


ภาพที่ 5 การแยกส่วนประกอบความถี่ด้วยตัวกรองแบบ 2 ช่อง

ที่มา: วิกิพีเดีย (2552)

1. ส่วนประกอบความถี่สูง ที่เรียกว่า “Detail (cD)” ซึ่งจะผ่านตัวกรองความถี่สูง (High-Pass Filter : HPF)
2. ส่วนประกอบความถี่ต่ำ ที่เรียกว่า “Approximation (cA)” ซึ่งจะผ่านตัวกรองความถี่ต่ำ (Low-Pass Filter : LPF)

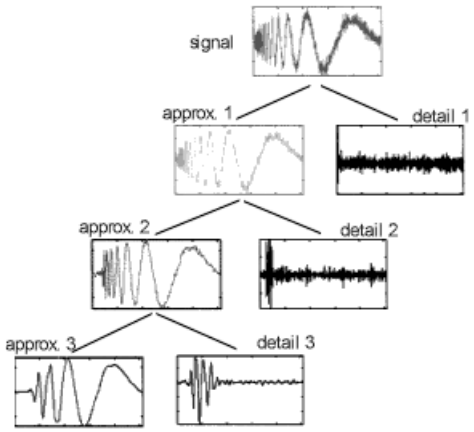
ภาพที่ 5 แสดงถึงการแยกส่วนประกอบของสัญญาณโดยตัวกรองแบบ 2 ช่อง และในทางกลับกันเราสามารถสังเคราะห์เพื่อสร้างกลับมาเป็นสัญญาณตั้งต้นได้โดยการนำส่วนประกอบความถี่ต่ำและส่วนประกอบความถี่สูงมารวมกัน การแยกส่วนประกอบของสัญญาณดังกล่าวจะมีการลดลงของอัตราการสุ่ม (Down Sampling) เป็น 2 เท่า



ภาพที่ 6 การแยกส่วนประกอบความถี่ด้วยตัวกรองแบบออคเทฟฟิลเตอร์แบงค์

ที่มา: วิกิพีเดีย (2552)

ลักษณะของตัวกรองแบบ 2 ช่องสัญญาณในลักษณะนี้เรียกว่า “Quadrature Mirror Filter: QMF” ในลักษณะของ Dyadic Wavelet Transform จะเป็นการนำตัวกรองสัญญาณแบบ 2 ช่องมาเรียงต่อกันในลักษณะโครงสร้างแบบต้นไม้ โดยใช้สัญญาณเอาท์พุทในส่วนความถี่ต่ำมาทำการแยกความถี่ออกอีกครั้งหนึ่ง ซึ่งเป็นลักษณะของการวิเคราะห์แบบออคเทฟฟิลเตอร์แบงค์ (Octave Analysis Filter Bank) โดยที่แต่ละขั้นตอนจะมีการเปลี่ยนแปลงความถี่ในอัตราครั้งละ 2 เท่า ซึ่งสามารถอธิบายการทำงานได้ดังแสดงในภาพที่ 6



ภาพที่ 7 การแยกส่วนประกอบความถี่ในระดับ 1 – 3

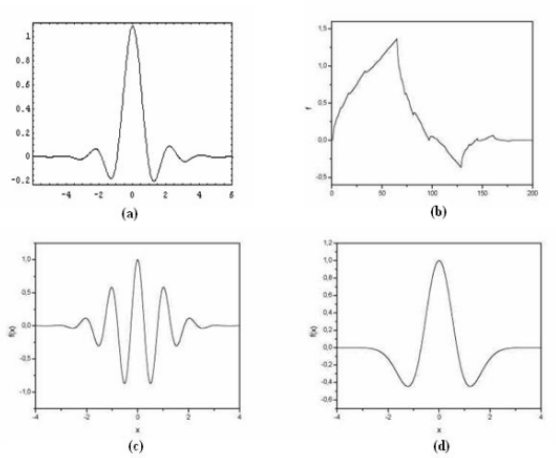
ที่มา: University of Ljubljana (2552)

จากรูปเป็นลักษณะของการแปลงเวฟเล็ตแบบเต็มหน่วยในลักษณะของ Dyadic Tree Structure โดยมีสัญญาณอินพุทเป็นสัญญาณที่ถูกสุ่มด้วยความถี่การสุ่ม f_s (Sampling Rate) จะถูกวิเคราะห์ด้วยตัวกรองสัญญาณ 2 ช่องในสเกลที่ 1 (2^1) ซึ่งสัญญาณจะถูกแยกออกเป็น 2 ช่วงความถี่คือส่วนความถี่สูงหรือ Detail 1 มีความถี่อยู่ในช่วง $\frac{f_s}{2} - \frac{f_s}{4}$ Hz และส่วนความถี่ต่ำหรือ Approximation 1 มีความถี่อยู่ในช่วง $\frac{f_s}{2} - 0$ Hz และในการวิเคราะห์ในสเกลที่ 2 (2^2) ก็ทำได้โดยการใช้ชุดกรองสัญญาณคู่เดิมมาวิเคราะห์ต่อจาก Approximation 1 ซึ่งผลการวิเคราะห์จะออกมาเป็นส่วนความถี่สูงสเกลที่ 2 หรือ Detail 2 ซึ่งมีความถี่ในช่วง $\frac{f_s}{4} - \frac{f_s}{8}$ Hz และส่วนความถี่ต่ำสเกลที่ 2 หรือ Approximation 2 มีความถี่อยู่ในช่วง $\frac{f_s}{8} - 0$ Hz ถ้าทำการวิเคราะห์ต่อในสเกลที่ 3 ก็สามารถทำซ้ำในลักษณะเดิม ในทางกลับกันก็สามารถที่จะรวมสัญญาณที่ทำการกระจายในหลายช่วงความถี่ให้กลับมาเป็นสัญญาณเดิมได้ ซึ่งผลการแปลงเวฟเล็ตแบบเต็มหน่วยได้ดังแสดงในภาพที่ 7

3. เวฟเล็ตแม่ (Mother Wavelet)

ในการวิเคราะห์สัญญาณใด ๆ ก็ตามนอกจากการเลือกรูปแบบในการวิเคราะห์ที่เหมาะสมแล้วยังมีความจำเป็นที่ต้องเลือกลักษณะของตัวกรองสัญญาณให้เหมาะสมด้วย สำหรับการแปลงเวฟเล็ตนั้นตัวกรองสัญญาณที่ใช้คือเวฟเล็ตแม่นั่นเอง ซึ่งเวฟเล็ตแม่มีอยู่ด้วยกันหลายแบบแต่ละแบบยังมีชนิดย่อย ๆ ลงไปอีก และเนื่องจากรูปแบบของเวฟเล็ตแม่ที่หลากหลายนี้เองจึงทำให้การแปลงเวฟเล็ตมีความยืดหยุ่นและสามารถนำไปประยุกต์ใช้งานได้หลากหลาย โดยเลือกเวฟเล็ตแม่

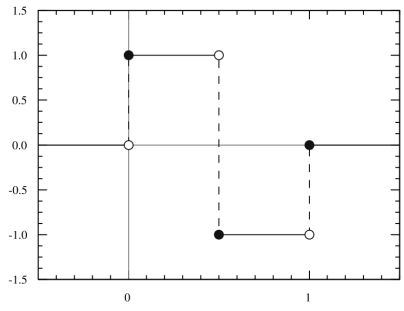
ให้เหมาะสมกับสัญญาณที่ต้องการวิเคราะห์ ภาพที่ 8 แสดงถึงตัวอย่างของเวฟเล็ตแม่ที่นิยมนำมาใช้วิเคราะห์สัญญาณ ส่วนภาพที่ 9 แสดงถึงเวฟเล็ตแม่แบบ Haar ที่ใช้ในงานวิจัยนี้



ภาพที่ 8 เวฟเล็ตแม่แบบ (a) Meyer (b) Daubechies (c) Morlet (d) Mexican Hat

ที่มา: Mathworks (2552)

ในวิทยานิพนธ์ฉบับนี้ทำการเปลี่ยนสัญญาณที่ได้รับจากระบบเครือข่ายให้เป็นสัญญาณที่อยู่ในรูปของจำนวนชุดข้อมูลต่อช่วงเวลา จากนั้นนำการแปลงเวฟเล็ตแบบเต็มหน่วยมาใช้ในการแยกส่วนประกอบของสัญญาณดังกล่าวแล้วนำไปวิเคราะห์ในแต่ละช่วงความถี่ เพื่อตรวจจับความผิดปกติที่เกิดจากการโจมตีแบบฟลัดดิ้งที่ระบบเครือข่ายต้นทาง



ภาพที่ 9 เวฟเล็ตแม่แบบ Haar

ที่มา: วิกิพีเดีย (2552)

4. สหสัมพันธ์ (Correlation)

สหสัมพันธ์คือตัววัดความสัมพันธ์ระหว่างตัวแปรตั้งแต่ 2 ตัวขึ้นไปว่ามีความสัมพันธ์กันมากน้อยเพียงใดหรือไม่ เช่น ความสูงกับน้ำหนัก, คะแนนสอบวิชาที่ 1 กับคะแนนสอบวิชาที่ 2 เป็นต้น ในการหาความสัมพันธ์ระหว่างตัวแปรสุ่มสองตัวขึ้นไป ถ้ามีตัวแปรสุ่มเพียง 2 ตัวจะเรียกความสัมพันธ์ระหว่างตัวแปรทั้งสองว่า “สหสัมพันธ์เชิงเดียว” (Simple Correlation) แต่ถ้ามีตัวแปรสุ่มมากกว่าสองตัวขึ้นไปเรียกว่า “สหสัมพันธ์พหุคูณ” (Multiple Correlation) งานวิจัยในวิทยานิพนธ์ฉบับนี้นำการหาค่าสหสัมพันธ์มาเปรียบเทียบความคล้ายคลึงของสัญญาณโดยใช้การวิเคราะห์สหสัมพันธ์เชิงเดียว

4.1 สหสัมพันธ์เชิงเดียว (Simple Correlation)

ในการแสดงความสัมพันธ์ระหว่างข้อมูล 2 ชุดอาจหาได้ในรูปสมการกำลังหนึ่งซึ่งแสดงถึงความสัมพันธ์เป็นเส้นตรง หรือสมการกำลังสองซึ่งแสดงถึงความสัมพันธ์เป็นพาราโบลา หรือสมการตั้งแต่กำลังสามเป็นต้นไป ในที่นี้จะพิจารณาเฉพาะการหาสหสัมพันธ์เชิงเส้น (Linear Correlation)

ถึงแม้จะทราบเส้นถดถอยที่แสดงถึงความสัมพันธ์ของข้อมูล 2 ชุดแล้วก็ตาม เรายังไม่ทราบว่าข้อมูล 2 ชุดนั้น ๆ มีความสัมพันธ์กันมากน้อยเพียงใด จึงต้องสร้างดัชนีขึ้นตัวหนึ่งเพื่อใช้วัดความมากน้อยของสหสัมพันธ์เชิงเส้นนี้ เรามักจะสมมติว่าการแจกแจงแบบมีเงื่อนไข $f(y|x)$ ของ Y เมื่อกำหนดค่า x เป็นการแจกแจงปกติที่มีค่าเฉลี่ย $\mu_{Y|x} = \alpha + \beta x$ ความแปรปรวน $\sigma^2_{Y|x} = \sigma^2$ และ X มีการแจกแจงปกติที่มีค่าเฉลี่ย μ_X ความแปรปรวน σ_X^2 ดังนั้นการแจกแจงความน่าจะเป็นร่วมกันของ X และ Y คือ

$$f(x,y) = N(y|x;\alpha+\beta x,\sigma^2)N(x;\mu_X,\sigma_X^2) \tag{16}$$

$$= \frac{1}{2\pi\sigma_X\sigma} e^{-\left(\frac{1}{2}\right)\left[\left\{\frac{y-(\alpha+\beta x)}{\sigma}\right\}^2+\left\{\frac{x-\mu_X}{\sigma_X}\right\}^2\right]} \tag{17}$$

เมื่อ $-\infty < x < \infty$ และ $-\infty < y < \infty$

ให้ $Y = \alpha + \beta x + E$

เมื่อ X เป็นตัวแปรสุ่มอิสระซึ่งไม่ขึ้นกับความคลาดเคลื่อน E จะได้

$$\mu_Y = \alpha + \beta\mu_X$$
$$\sigma_Y^2 = \sigma^2 + \beta^2\sigma_X^2$$

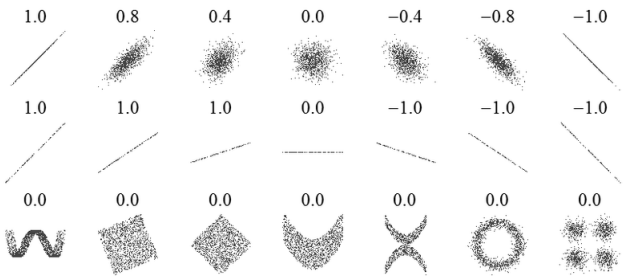
แทน σ_Y และ σ_Y^2 ในสมการแรกจะได้ฟังก์ชันการแจกแจงปกติสองตัวแปร (Bivariate Normal Distribution) คือ

$$f(x,y) = \frac{1}{2\pi\sigma_X\sigma_Y\sqrt{1-\rho^2}} e^{-\frac{1}{2(1-\rho^2)}\left[\left(\frac{x-\mu_X}{\rho X}\right)^2 - 2\rho\frac{(x-\mu_X)(y-\mu_Y)}{\sigma_X \sigma_Y} + \left(\frac{y-\mu_Y}{\sigma_Y}\right)^2\right]} \quad (18)$$

เมื่อ $-\infty < x < \infty$ และ $-\infty < y < \infty$
เมื่อ $\rho^2 = 1 - \frac{\sigma^2}{\sigma_Y^2} = \beta^2\frac{\sigma_X^2}{\sigma_Y^2}$

เรียกตัวคงค่า ρ ว่า “สัมประสิทธิ์สหสัมพันธ์” (Correlation Coefficient) ค่า ρ จะเป็น 0 เมื่อ $\beta = 0$ ซึ่งแสดงว่าไม่มีการถดถอยเชิงเส้น เนื่องจาก $\sigma_Y^2 \geq \sigma^2$ ฉะนั้น $-1 \leq \rho \leq 1$ ถ้า $\rho = \pm 1$ ก็ต่อเมื่อ $\rho^2 = 0$ ซึ่งแสดงว่า X และ Y มีความสัมพันธ์แบบเชิงเส้นอย่างสมบูรณ์ เมื่อ $\rho = +1$ เส้นตรงจะมีความชันเป็นบวก และเมื่อ $\rho = -1$ เส้นตรงจะมีความชันเป็นลบ จะกล่าวได้ว่าถ้า ρ เข้าใกล้ ± 1 จะได้ X และ Y มีสหสัมพันธ์หรือมีความสัมพันธ์เชิงเส้นอย่างดี แต่ถ้า ρ เข้าใกล้ 0 ก็แสดงว่ามีสหสัมพันธ์เพียงเล็กน้อยหรือไม่มีเลย

ในกรณีที่ค่าของตัวแปรสุ่มทั้งสองมีการเปลี่ยนแปลงไปในทางเดียวกันสัมประสิทธิ์สหสัมพันธ์จะมีเครื่องหมายบวก และในกรณีที่มีการเปลี่ยนแปลงกลับกันสัมประสิทธิ์สหสัมพันธ์จะมีเครื่องหมายเป็นลบ และถ้าไม่มีสหสัมพันธ์ค่าของสัมประสิทธิ์สหสัมพันธ์จะมีค่าเป็น 0 ดังแสดงในภาพที่ 9



ภาพที่ 10 ค่าสัมประสิทธิ์สหสัมพันธ์ของกลุ่มข้อมูลของจุด (x,y)

ที่มา: วิกิพีเดีย (2552)

4.2 สัมประสิทธิ์สหสัมพันธ์

ค่าประมาณของสัมประสิทธิ์สหสัมพันธ์ ρ คือ สัมประสิทธิ์สหสัมพันธ์ตัวอย่าง (Sample Correlation) r

$$r = \frac{S_{xy}}{\sqrt{S_{xx}S_{yy}}} = b\sqrt{\frac{S_{xx}}{S_{yy}}} \tag{19}$$

เพื่อแสดงว่า $-1 \leq r \leq 1$ จะเห็นได้จากค่า Error Sum of Squares (SSE) ดังนี้

$$SSE = S_{yy} - bS_{xy} \tag{20}$$

หารตลอดด้วย S_{yy} และแทนค่า b ด้วย S_{xy}/S_{xx}

$$r^2 = 1 - \frac{SSE}{S_{yy}} \tag{21}$$

เนื่องจาก $S_{yy} \geq SSE$ จึงสรุปได้ว่า $0 \leq r^2 \leq 1$ ดังนั้น $-1 \leq r \leq 1$ และ r มีค่า ± 1 เมื่อ $SSE = 0$ แสดงว่าเส้นถดถอยผ่านทุก ๆ จุด หรือไม่มีความคลาดเคลื่อนเลย ฉะนั้นค่าสูงสุดของ r คือ 1 และเนื่องจาก r มีเครื่องหมายได้ทั้ง $+$ และ $-$ ฉะนั้นค่าต่ำสุดของ r คือ -1 เครื่องหมายของ r จะแสดงทิศทางว่าข้อมูลแปรตามกันหรือกลับกัน และ r จะมีเครื่องหมายอย่างเดียวกับ b

ดังนั้นถ้าค่าสมบูรณ์ของ r ใกล้ 1 มาก แสดงว่ามีความสัมพันธ์เชิงเส้นอย่างดี แต่ถ้าค่าสมบูรณ์ของ r ไม่มากนัก (ประมาณ 0.6) แสดงว่ามีความสัมพันธ์เชิงเส้นดีพอสมควร ถ้าค่าสมบูรณ์ของ r ต่ำกว่านี้แสดงว่ามีความสัมพันธ์เชิงเส้นน้อย และถ้าค่าสมบูรณ์ของ r ใกล้ 0 (ประมาณ 0.1) แสดงว่าไม่มีความสัมพันธ์เชิงเส้นเลย

สำหรับค่า r ที่อยู่ระหว่าง -1 และ +1 เช่น $r = 0.3$ และ $r = 0.6$ เราทราบว่ามีสหสัมพันธ์ที่เป็นบวกทั้งคู่ แต่จะเป็นการไม่ถูกต้องถ้าจะสรุปว่าเมื่อ $r = 0.6$ แสดงถึงความสัมพันธ์เชิงเส้นที่เป็น 2 เท่าของค่า $r = 0.3$ อีกประการหนึ่งถ้าพิจารณาค่าของ r^2 จะได้ว่า $100r^2\%$ ของ

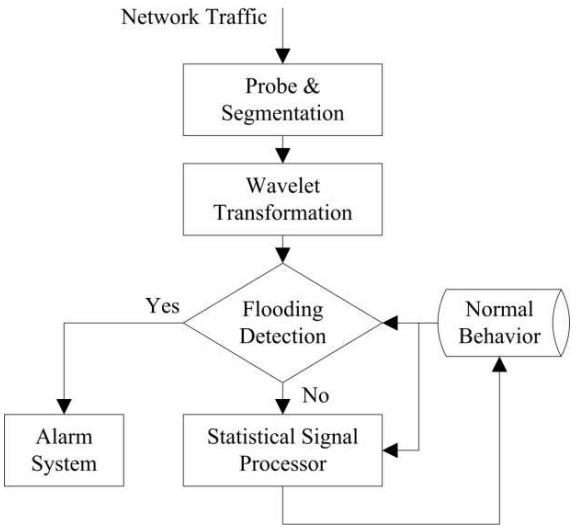
การแปรผันในค่าของ Y เป็นผลเนื่องมาจากการมีความสัมพันธ์เชิงเส้นกับตัวแปร X ดังนั้น สหสัมพันธ์ที่มีค่า 0.6 หมายความว่า 36% ของการแปรผันของตัวแปรสุ่ม Y เป็นผลเนื่องมาจากการมีความสัมพันธ์เชิงเส้นกับตัวแปรสุ่ม X

อุปกรณ์และวิธีการ

อุปกรณ์

1. ฮาร์ดแวร์ระบบ
   1.1. เครื่องคอมพิวเตอร์
   1.2. สื่อบันทึกข้อมูลภายนอก (External Harddisk)
2. ซอฟต์แวร์ระบบ
   2.1. ระบบปฏิบัติการ Linux
   2.2. คอมไพเลอร์ GCC
   2.3. โปรแกรมประยุกต์ GSL Library
   2.4. โปรแกรมประยุกต์ Gnu Plot
   2.5. โปรแกรมประยุกต์ Wireshark

วิธีการ



ภาพที่ 11 แผนผังการทำงานของวิธีตรวจจับการโจมตีแบบฟลัดดิ้ง

วิธีตรวจจับสามารถแบ่งการทำงานออกได้เป็น 5 ส่วนดังแสดงในภาพที่ 11 โดยแต่ละส่วนมีหน้าที่การทำงานดังต่อไปนี้

Probe and Segmentation

เป็นส่วนในการรับข้อมูลจากระบบเครือข่ายและนำมาคัดแยกเป็นกลุ่มที่เราสนใจ เช่น คัดแยกตามโปรโตคอล, คัดแยกตามไอพีแอดเดรส, คัดแยกตามพอร์ต เป็นต้น และเปลี่ยนข้อมูลดังกล่าวให้อยู่ในรูปสัญญาณของความสัมพันธ์ระหว่างจำนวนชุดข้อมูลต่อหน่วยเวลา ในงานวิจัยนี้ทำการคัดแยกข้อมูลจากระบบเครือข่ายตามโปรโตคอล ได้แก่ ICMP, TCP SYN, TCP SYN/ACK และ UDP

Wavelet Transformation

เป็นส่วนแยกส่วนประกอบความถี่ของสัญญาณที่ได้รับมาจากส่วน Probe and Segmentation ด้วยการแปลงเวฟเล็ตแบบเต็มหน่วยให้เป็นส่วนประกอบความถี่ในแต่ละระดับ เพื่อนำส่วนประกอบความถี่ในแต่ละระดับที่ได้ไปวิเคราะห์เพื่อหาความผิดปกติภายในระบบเครือข่ายคอมพิวเตอร์ ในการแปลงเวฟเล็ตแบบเต็มหน่วยของงานวิจัยนี้ใช้เวฟเล็ตแม่แบบต่าง ๆ ในการแยกส่วนประกอบความถี่ให้อยู่ในรูปของส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำ จากนั้นจึงนำส่วนประกอบความถี่ต่ำมาแยกส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำอีกครั้งในระดับต่อไป

Flooding Detection

เป็นส่วนในการตัดสินใจว่ามีการโจมตีแบบฟลัดดิ้งหรือไม่ โดยการนำส่วนประกอบความถี่ของสัญญาณที่ต้องการตรวจสอบมาเปรียบเทียบกับสัญญาณระบบเครือข่ายแบบปกติ (Normal Behavior) ที่เก็บอยู่ในฐานข้อมูล โดยเปรียบเทียบว่ามีความแตกต่างเกินค่าที่กำหนดไว้ (Baseline) หรือไม่ ถ้าแตกต่างกันเกินค่าที่กำหนดไว้ระบบจะตัดสินใจว่ามีการโจมตีแบบฟลัดดิ้งเกิดขึ้น และส่งสัญญาณเตือนไปยังระบบเตือนภัย (Alarm System) ถ้าแตกต่างกันไม่เกินค่าที่กำหนดไว้ ระบบจะตัดสินใจว่าสัญญาณดังกล่าวมีลักษณะเป็นปกติ ก็จะส่งต่อสัญญาณดังกล่าวไปประมวลผลที่ Statistics Signal Processor เพื่อคำนวณหาเส้นค่าเฉลี่ยก่อนที่เก็บลงในฐานข้อมูลต่อไป

Statistics Signal Processor

เป็นส่วนที่นำสัญญาณระบบเครือข่ายที่ได้จาก Flooding Decision และข้อมูลจากฐานข้อมูลที่จัดเก็บสัญญาณเครือข่ายแบบปกติมาประมวลผลด้วยวิธีหาเส้นค่าเฉลี่ยและความแปรปรวนของแต่ละช่วงเวลา จากนั้นจึงนำสัญญาณที่ได้จากการประมวลผลดังกล่าวไปจัดเก็บที่ฐานข้อมูลเพื่อใช้ในการอ้างอิงต่อไป

Normal Behavior Database

เป็นฐานข้อมูลที่จัดเก็บส่วนประกอบความถี่ในระดับที่ต้องการวิเคราะห์และมีการประมวลผลเรียบร้อยแล้ว เก็บในลักษณะแฟ้มข้อมูลที่ประกอบด้วยค่าเฉลี่ยและความแปรปรวนของแต่ละช่วงเวลา ฐานข้อมูลนี้เก็บสัญญาณของเครือข่ายระหว่างเวลา 0.00.00 – 23.59.59 นาฬิกา

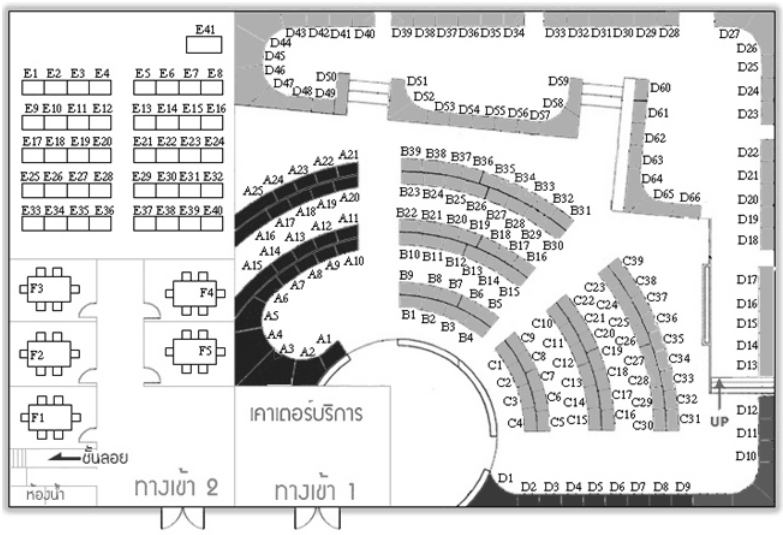
สถานที่จัดเก็บชุดข้อมูล

ข้อมูลที่นำมาใช้ในการทดลองได้จัดเก็บจากห้องปฏิบัติการคอมพิวเตอร์และอินเตอร์เน็ตพร้อมศูนย์การเรียนรู้ตามอัธยาศัยและศูนย์กิจกรรมไอทีของมหาวิทยาลัยเกษตรศาสตร์ ที่มีชื่อเป็นภาษาอังกฤษว่า Kasetsart Information Technology Square หรือที่เรียกกันสั้น ๆ ว่า KITS โดยทำการจัดเก็บทั้งสิ้นเป็นระยะเวลา 3 เดือน ระหว่างเดือนมิถุนายนถึงเดือนสิงหาคม พ.ศ. 2551

KITS เป็นศูนย์กิจกรรมไอทีที่เน้นการเรียนรู้ตามอัธยาศัย มีบรรยากาศการเรียนรู้แบบปัญญาภิรมย์ (Edutainment) เพื่อให้นิสิตได้ใช้เป็นศูนย์กิจกรรมไอที โดยสามารถดำเนินกิจกรรมร่วมกันระหว่างนิสิตกับนิสิต พี่กับน้อง เพื่อนสอนเพื่อน ทั้งในโลกแห่งความจริงเสมือน (Cyber) และโลกแห่งความเป็นจริง โดยมีกิจกรรมอย่างหลากหลาย เช่น การฝึกอบรม การสนทนากลุ่ม การนำเสนอ โดยมหาวิทยาลัยมีโครงการออกใบรับรองความรู้การเป็น e-Student ให้กับนิสิต

e-Zone ของมหาวิทยาลัยเป็นสถานที่ที่จะเอื้อประโยชน์ต่อการเรียนรู้ของนิสิต ซึ่งเน้นให้ตอบสนองต่อวิถีชีวิตใหม่แบบดิจิตัลของนิสิต ไอทีไร้สาย และ e-Edutainment เป็นต้น สำหรับ KITS เป็นสถานที่หนึ่งซึ่งเป็นแหล่งเรียนรู้ด้วยตนเองของนิสิต เพื่อเน้นให้เกิดสิ่งแวดล้อมในมหาวิทยาลัย ที่เอื้อต่อการศึกษา แสวงหาและเรียนรู้อย่างมีความสุข

จากนโยบายและความต้องการของผู้บริหารมหาวิทยาลัย นำโดย รศ.ดร.วิโรจ อิ่มพิทักษ์ อธิการบดีมหาวิทยาลัยเกษตรศาสตร์ในขณะนั้น ได้เล็งเห็นความสำคัญของการสร้างกิจกรรมการเรียนรู้ของนิสิต เพื่อให้นิสิตมีศูนย์เรียนรู้ตามอัธยาศัยและเรียนรู้ร่วมกัน จึงได้อนุมัติพื้นที่เพื่อสร้างศูนย์บริการคอมพิวเตอร์สำหรับนิสิตในชื่อ “Kasetsart IT Square” หรือ KITS โดยพื้นที่บริเวณนี้อยู่ชั้นล่างของอาคารกิจกรรมนิสิตและชั้นบนเป็นที่ทำการของศูนย์การศึกษานานาชาติและหอพักนิสิตต่างชาติ



ภาพที่ 12 แผนผังพื้นที่การให้บริการคอมพิวเตอร์ Kasetsart IT Square

ที่มา: มหาวิทยาลัยเกษตรศาสตร์ (2552)

จุดเริ่มต้นของ KITS เริ่มต้นจากนโยบายหลักของมหาวิทยาลัยเกษตรศาสตร์เน้นการใช้เทคโนโลยีสารสนเทศและการสื่อสาร เพื่อพัฒนาการเรียนรู้ของนิสิต เพื่อเพิ่มศักยภาพการใช้สารสนเทศในเรื่องความเข้มแข็งทางวิชาการ และก่อประโยชน์สูงสุดทั้งในด้านบริหาร การเรียนการสอน การวิจัย เป้าหมายที่สำคัญคือการพัฒนามหาวิทยาลัยก้าวสู่การเป็น e-University

ภายใต้นโยบายเทคโนโลยีสารสนเทศ มหาวิทยาลัยมีโครงการพัฒนานิสิตให้ก้าวสู่การเป็น e-Student จึงได้วางโครงสร้างพื้นฐานหลายอย่างเพื่อเอื้อต่อการสนับสนุน เช่น การวางโครงสร้าง KUWiN เครือข่ายบริการไร้สายครอบคลุมทั่วมหาวิทยาลัย การบริการการเรียนการสอนแบบ e-Learning การให้บริการห้องสมุดด้วยดิจิตอลไลบรารี ซึ่งเป็นกิจกรรมนิสิตทั้งที่เป็นแบบไซเบอร์สเปซและแบบสถานที่จริง

พื้นที่ให้บริการรวม 1,064 ตารางเมตร

1. พื้นที่บริการคอมพิวเตอร์ 562 ตารางเมตร
2. พื้นที่ห้องฝึกอบรม 130 ตารางเมตร
3. พื้นที่ห้องสนทนากลุ่มย่อย 54 ตารางเมตร
4. พื้นที่ห้องนิทรรศการ 164 ตารางเมตร
5. KUWiN Hot Spot พร้อมปลั๊กไฟ 150 ตารางเมตร

ภายใน KITS มีคอมพิวเตอร์ที่ให้บริการทั่วไป 169 เครื่อง ห้องฝึกอบรม 40 เครื่อง ห้องสนทนากลุ่มย่อย 5 เครื่อง และคอมพิวเตอร์บริการระบบจองใช้ 3 เครื่อง บริการใช้คอมพิวเตอร์ไม่มีค่าใช้จ่ายใด ๆ อีกทั้งมีบริการเครื่องพิมพ์เลเซอร์ขาวดำ, เครื่องพิมพ์เลเซอร์สีและเครื่องพิมพ์สีแบบพ่นหมึก คิดค่าบริการพิมพ์แบบตัดค่าใช้จ่ายอัตโนมัติผ่านบัญชีอิเล็กทรอนิกส์ (e-Wallet) ภายในอาคารสามารถใช้เครือข่ายไร้สายได้ทุกตารางนิ้ว ผู้ใช้บริการสามารถจองใช้บริการล่วงหน้าผ่านอินเตอร์เน็ตด้วยรหัสบัญชีเครือข่ายนนทรีของมหาวิทยาลัยได้ที่เว็บไซต์ http://kits.ku.ac.th/booking โดยกำหนดวันเวลาที่ต้องการได้ การผ่านเข้าออกจะใช้ระบบตรวจสอบด้วยคอมพิวเตอร์และประตูอัตโนมัติที่ใช้หัวอ่าน RFID

เปิดให้บริการทุกวันเวลาราชการ ระหว่างเวลา 08.30-24.00 น. และวันเสาร์ระหว่างเวลา 08.30-16.30 น. แก่ นิสิต บุคลากร อาจารย์ และบุคคลที่มีรหัสบัญชีเครือข่ายนนทรีโดยไม่เสียค่าธรรมเนียมการใช้บริการ

การบริหารจัดการของ KITS ดำเนินการภายใต้คณะกรรมการที่มาจาก 3 หน่วยงาน คือ กองยานพาหนะ อาคารและสถานที่ กองกิจการนิสิต และสำนักบริการคอมพิวเตอร์ โดยสำนักคอมพิวเตอร์ทำหน้าที่เป็นหน่วยงานปฏิบัติการให้บริการภายในอาคาร

กระบวนการทดลอง

การทดสอบตรวจจับความผิดปกติในระบบเครือข่ายคอมพิวเตอร์ในงานวิจัยนี้ได้จำลองการโจมตีแบบฟลัดดิ้งเข้าไปในระบบเครือข่ายคอมพิวเตอร์เริ่มที่เวลา 9.00 น. สิ้นสุดที่เวลา 23.00 น. โดยมีช่วงระยะเวลาห่างกัน 1 ชั่วโมง โดยมีรูปแบบจำลองทั้งหมด 3 รูปแบบ ได้แก่

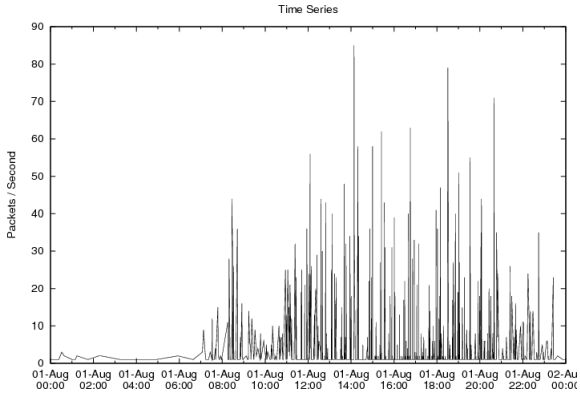
1. จำลองการโจมตีแบบฟลัดดิ้งเข้าไปในระบบเครือข่ายเป็นระยะเวลา 1 นาที โดยเริ่มจากเวลา 9.00-23.00 น. มีระยะห่างแต่ละช่วงการจำลองเป็นเวลา 1 ชั่วโมง โดยทำการโจมตีแบบฟลัดดิ้งระหว่าง 5-80 ชุดข้อมูลต่อวินาทีและเพิ่มขึ้นครั้งละ 5 ชุดข้อมูลต่อวินาที
2. จำลองการโจมตีแบบฟลัดดิ้งเข้าไปในระบบเครือข่ายเป็นระยะเวลาระหว่าง 1-45 นาที โดยเริ่มจากเวลา 9.00-23.00 น. มีระยะห่างแต่ละช่วงการจำลองเป็นเวลา 1 ชั่วโมง โดยทำการโจมตีแบบฟลัดดิ้งที่ 50 ชุดข้อมูลต่อวินาที
3. จำลองการโจมตีแบบฟลัดดิ้งโดยเริ่มจาก 1 ชุดข้อมูลต่อวินาทีเข้าไปในระบบเครือข่ายเป็นระยะเวลา 5 นาที จากนั้นจึงเพิ่มเป็น 2 ชุดข้อมูลต่อวินาทีเข้าไปในระบบเครือข่ายเป็นระยะเวลา 5 นาที และเพิ่มชุดข้อมูลเช่นนี้ไปเรื่อย ๆ จนถึง 50 ชุดข้อมูลต่อวินาที เริ่มจากเวลา 9.00-23.00 น. มีระยะห่างแต่ละช่วงการจำลองเป็นเวลา 1 ชั่วโมง

ผลและวิจารณ์

ผล

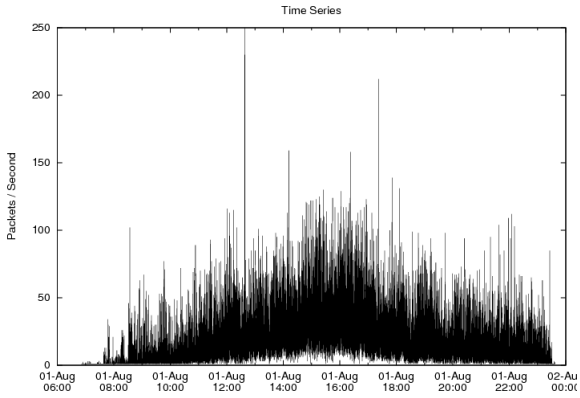
ผลการคัดแยกข้อมูลจราจรทางคอมพิวเตอร์

ผลการคัดแยกข้อมูลจราจรทางคอมพิวเตอร์ที่ส่งออกจากห้องปฏิบัติการคอมพิวเตอร์และอินเตอร์เน็ต มหาวิทยาลัยเกษตรศาสตร์ บางเขน โดยแยกตามประเภทของโปรโตคอล ได้แก่ ICMP, TCP SYN, TCP SYN/ACK และ UDP ระหว่างวันที่ 1 สิงหาคม 2551 ถึงวันที่ 29 สิงหาคม 2551 ในการทดลองนี้พิจารณาเฉพาะการใช้งานระหว่างวันจันทร์ถึงวันศุกร์ที่มีการใช้งานระหว่างเวลา 8.30 น.ถึง 24.00 น. ซึ่งมีการกำหนดค่าพารามิเตอร์ช่วงเวลา (Interval Time) ในการนับจำนวนชุดข้อมูลเท่ากับ 1 วินาที



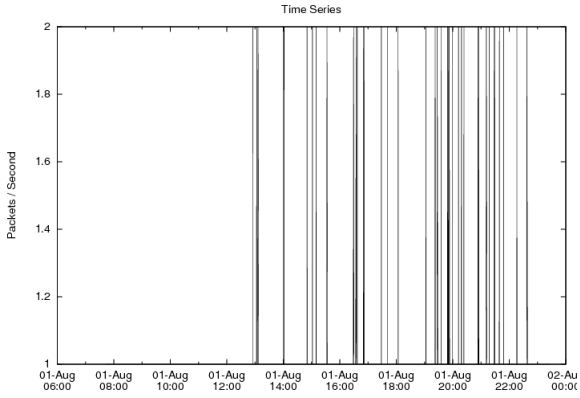
ภาพที่ 13 ตัวอย่างข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก

ภาพที่ 13 แสดงถึงตัวอย่างข้อมูลจราจรทางคอมพิวเตอร์ของโปรโตคอล ICMP ที่ส่งออกจากห้องปฏิบัติการคอมพิวเตอร์อินเตอร์เน็ตและคอมพิวเตอร์ต่อช่วงเวลา 1 วินาทีในวันที่ 1 สิงหาคม 2551 ซึ่งได้จากการทำงานในส่วนของ Probe and Segmentation จากภาพแสดงให้เห็นถึงปริมาณข้อมูลที่มีลักษณะแบบ Impulse เกิดเป็นช่วง ๆ เพราะการใช้งานภายในห้องปฏิบัติการมีการใช้งานโปรโตคอล ICMP ไม่มากนัก ส่วนใหญ่มีการส่งโปรโตคอลนี้ไม่เกิน 100 ชุดข้อมูลต่อวินาที



ภาพที่ 14 ตัวอย่างข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก

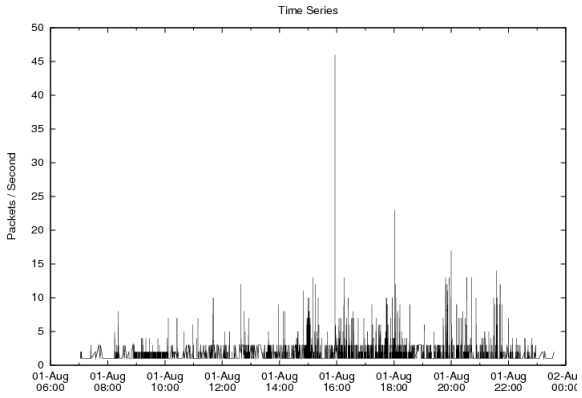
ภาพที่ 14 แสดงถึงตัวอย่างข้อมูลจราจรทางคอมพิวเตอร์ของโปรโตคอล TCP SYN ที่ส่งออกจากห้องปฏิบัติการคอมพิวเตอร์อินเตอร์เน็ตและคอมพิวเตอร์ต่อช่วงเวลา 1 วินาทีในวันที่ 1 สิงหาคม 2551 จากภาพแสดงให้เห็นถึงปริมาณข้อมูลที่เกิดขึ้นเป็นจำนวนมากและมีลักษณะคล้ายเสี้ยววงกลมคว่ำ เนื่องจากการใช้งานภายในห้องปฏิบัติการมีการใช้งานโปรโตคอล TCP SYN เป็นจำนวนมาก เช่น การร้องขอเพื่อดูข้อมูลในเว็บไซต์ต่างหรือการร้องขอเพื่อติดต่อการใช้งานต่าง ๆ เป็นต้น



ภาพที่ 15 ตัวอย่างข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก

ภาพที่ 15 แสดงถึงตัวอย่างข้อมูลจราจรทางคอมพิวเตอร์ของโปรโตคอล TCP SYN/ACK ที่ส่งออกจากห้องปฏิบัติการคอมพิวเตอร์อินเตอร์เน็ตและคอมพิวเตอร์ต่อช่วงเวลา 1 วินาทีในวันที่ 1 สิงหาคม 2551 จากภาพแสดงให้เห็นถึงปริมาณข้อมูลที่มีลักษณะแบบ Impulse เกิดเป็นช่วง ๆ เช่นเดียวกับโปรโตคอล ICMP และมีการส่งชุดข้อมูลต่อช่วงเวลาออกไปน้อยมากเมื่อเทียบกับ

โปรโตคอล ICMP ทั้งนี้เพราะโปรโตคอล TCP SYN/ACK เป็นโปรโตคอลที่ใช้ในการตอบการร้องขอเพื่อเริ่มต้นการติดต่อจากเครื่องคอมพิวเตอร์อื่น ๆ แต่เนื่องจากคอมพิวเตอร์ภายในห้องปฏิบัติการคอมพิวเตอร์ส่วนใหญ่เป็นเครื่องคอมพิวเตอร์ลูกข่าย ดังนั้นจึงไม่ค่อยมีการร้องขอเพื่อเริ่มต้นการติดต่อจากเครื่องคอมพิวเตอร์อื่น ๆ



ภาพที่ 16 ตัวอย่างข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก

ภาพที่ 16 แสดงถึงตัวอย่างข้อมูลจราจรทางคอมพิวเตอร์ของโปรโตคอล UDP ที่ส่งออกจากห้องปฏิบัติการคอมพิวเตอร์อินเตอร์เน็ตและคอมพิวเตอร์ต่อช่วงเวลา 1 วินาทีในวันที่ 1 สิงหาคม 2551 จากภาพแสดงให้เห็นถึงปริมาณข้อมูลที่มีลักษณะแบบ Impulse เกิดเป็นช่วง ๆ แต่มีความต่อเนื่องกันของสัญญาณมากกว่าโปรโตคอล ICMP เนื่องจากโปรโตคอล UDP มักเกิดจากการใช้งาน Streaming Data เช่น การดูโทรทัศน์ผ่านระบบเครือข่าย, การฟังวิทยุผ่านระบบเครือข่าย หรือการโทรศัพท์ผ่านระบบเครือข่าย เป็นต้น

ผลการเปรียบเทียบสหสัมพันธ์ของข้อมูลจราจรทางคอมพิวเตอร์

ดังนั้นเพื่อเป็นการวิเคราะห์ถึงลักษณะพฤติกรรมการส่งข้อมูลที่ได้จัดเก็บจากระบบเครือข่ายต้นทางว่าในแต่ละวันมีการส่งข้อมูลลักษณะคล้ายคลึงกันหรือไม่ เราจึงได้ทำการทดลองเปรียบเทียบค่าสหสัมพันธ์ของข้อมูลจราจรทางคอมพิวเตอร์ โดยทำการเปรียบเทียบข้อมูลจราจรทางคอมพิวเตอร์ในแต่ละวันกับข้อมูลจราจรทางคอมพิวเตอร์ในวันอื่น ๆ แยกตามโปรโตคอล

ตารางที่ 2 ค่าสัมประสิทธิ์สหสัมพันธ์ของโปรโตคอล ICMP

	1-Aug	4-Aug	5-Aug	6-Aug	7-Aug	8-Aug
1-Aug	1.000	0.001	0.000	-0.001	0.001	-0.002
4-Aug	0.001	1.000	0.006	0.001	0.000	-0.001
5-Aug	0.000	0.006	1.000	-0.001	0.000	0.000
6-Aug	-0.001	0.001	-0.001	1.000	0.001	0.003
7-Aug	0.001	0.000	0.000	0.001	1.000	-0.001
8-Aug	-0.002	-0.001	0.000	0.003	-0.001	1.000

ตารางที่ 3 ค่าสัมประสิทธิ์สหสัมพันธ์ของโปรโตคอล TCP SYN

	1-Aug	4-Aug	5-Aug	6-Aug	7-Aug	8-Aug
1-Aug	1.000	0.505	0.531	0.555	0.559	0.495
4-Aug	0.505	1.000	0.541	0.543	0.542	0.525
5-Aug	0.531	0.541	1.000	0.609	0.580	0.563
6-Aug	0.555	0.543	0.609	1.000	0.600	0.557
7-Aug	0.559	0.542	0.580	0.600	1.000	0.542
8-Aug	0.495	0.525	0.563	0.557	0.542	1.000

ตารางที่ 4 ค่าสัมประสิทธิ์สหสัมพันธ์ของโปรโตคอล TCP SYN/ACK

	1-Aug	4-Aug	5-Aug	6-Aug	7-Aug	8-Aug
1-Aug	1.000	0.015	0.027	0.017	0.037	0.040
4-Aug	0.015	1.000	0.020	0.009	0.001	0.000
5-Aug	0.027	0.020	1.000	0.016	0.020	0.055
6-Aug	0.017	0.009	0.016	1.000	0.026	0.016
7-Aug	0.037	0.001	0.020	0.026	1.000	0.022
8-Aug	0.040	0.000	0.055	0.016	0.022	1.000

ตารางที่ 5 ค่าสัมประสิทธิ์สหสัมพันธ์ของโปรโตคอล UDP

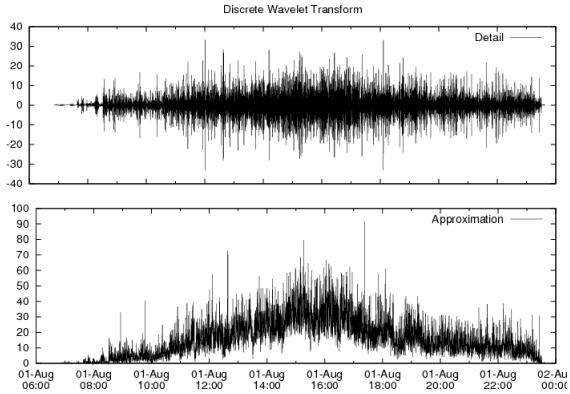
	1-Aug	4-Aug	5-Aug	6-Aug	7-Aug	8-Aug
1-Aug	1.000	0.048	-0.004	0.019	-0.011	0.015
4-Aug	0.048	1.000	-0.008	0.083	-0.053	0.019
5-Aug	-0.004	-0.008	1.000	-0.001	-0.012	0.151
6-Aug	0.019	0.083	-0.001	1.000	0.026	0.027
7-Aug	-0.011	-0.053	-0.012	0.026	1.000	-0.036
8-Aug	0.015	0.019	0.151	0.027	-0.036	1.000

ตารางที่ 2 ถึง 5 แสดงถึงค่าสัมประสิทธิ์สหสัมพันธ์ในแต่ละวันเปรียบเทียบกับวันอื่น ๆ แยกตามประเภทของโปรโตคอล ค่าสัมประสิทธิ์ดังกล่าวมีค่าอยู่ระหว่าง -1 ถึง 1 โดยค่าสัมประสิทธิ์ที่มีค่า -1 แสดงให้เห็นว่าข้อมูลจราจรทางคอมพิวเตอร์มีความสัมพันธ์ในทิศทางตรงกันข้าม ค่าสัมประสิทธิ์ที่มีค่า 1 แสดงให้เห็นว่าข้อมูลจราจรทางคอมพิวเตอร์มีความสัมพันธ์ในทิศทางเดียวกัน ส่วนค่าสัมประสิทธิ์ที่มีค่า 0 แสดงให้เห็นว่าข้อมูลจราจรทางคอมพิวเตอร์ไม่มีความสัมพันธ์กัน จากตารางดังกล่าวแสดงให้เห็นว่าข้อมูลจราจรทางคอมพิวเตอร์ของโปรโตคอล ICMP, TCP SYN/ACK และ UDP ในแต่ละวันมีความสัมพันธ์กับข้อมูลในวันอื่น ๆ น้อยมากเพราะส่วนใหญ่ค่าสัมประสิทธิ์ที่ได้มีค่าใกล้กับค่า 0 ส่วนข้อมูลจราจรทางคอมพิวเตอร์ของโปรโตคอล TCP SYN ในแต่ละวันมีความสัมพันธ์กับข้อมูลในวันอื่น ๆ มากกว่าโปรโตคอล ICMP, TCP SYN/ACK และ UDP เพราะค่าสัมประสิทธิ์ที่ได้มีค่าประมาณ 0.5-0.6 กล่าวอีกนัยหนึ่งก็คือข้อมูลจราจรทางคอมพิวเตอร์ของโปรโตคอล TCP SYN ในแต่ละวันมีรูปแบบของข้อมูลที่คล้ายคลึงกัน

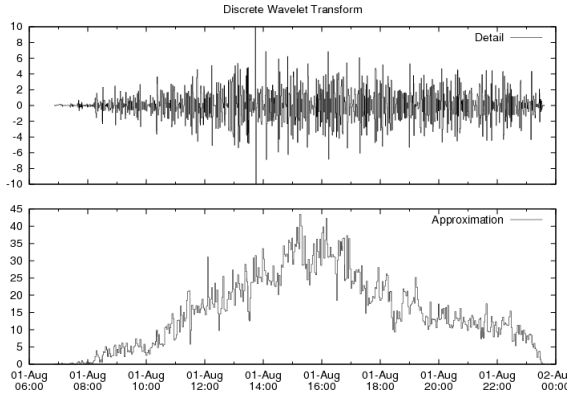
ผลการแยกส่วนประกอบความถี่ด้วยวิธีเวฟเล็ต

ผลการแยกส่วนประกอบความถี่สูงและความถี่ต่ำด้วยวิธีเวฟเล็ต ในการทดลองได้ทำการแยกส่วนประกอบความถี่แบบต่อเนื่องดังแสดงในภาพที่ 6 โดยใช้เวฟเล็ตแม่แบบ Haar ดังแสดงในภาพที่ 9 เนื่องจากสามารถคำนวนและแยกส่วนประกอบความถี่ได้รวดเร็วกว่าการใช้เวฟเล็ตแม่แบบอื่น และในการทดลองมีการแยกส่วนประกอบความถี่ทั้งหมด 11 ระดับ ในการแยกส่วนประกอบความถี่ที่มีระดับมากกว่า 11 ระดับนั้นส่วนประกอบความถี่ต่ำ (Approximation) จะมี

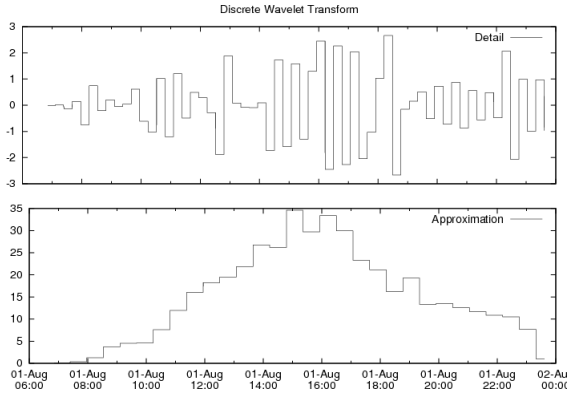
ลักษณะที่ค่อนข้างหยาบไม่เหมาะสำหรับการนำไปวิเคราะห์เพื่อหาความผิดปกติภายในเครือข่ายระบบคอมพิวเตอร์



ภาพที่ 17 ตัวอย่างส่วนประกอบความถี่สูง (บน) และส่วนประกอบความถี่ต่ำ (ล่าง) ระดับที่ 3



ภาพที่ 18 ตัวอย่างส่วนประกอบความถี่สูง (บน) และส่วนประกอบความถี่ต่ำ (ล่าง) ระดับที่ 7

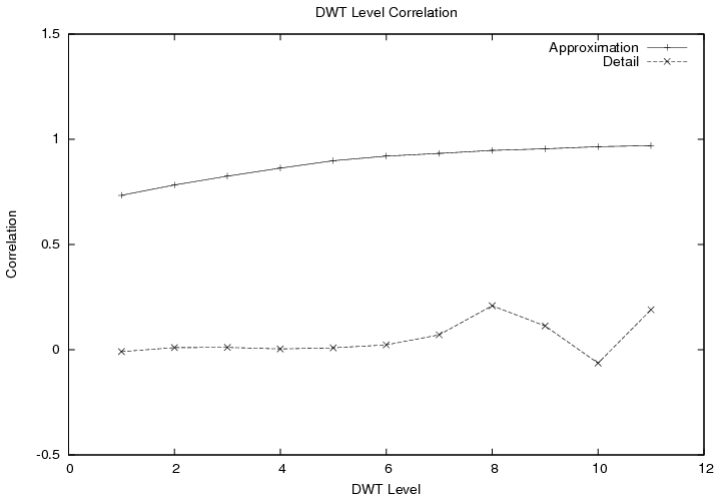


ภาพที่ 19 ตัวอย่างส่วนประกอบความถี่สูง (บน) และส่วนประกอบความถี่ต่ำ (ล่าง) ระดับที่ 11

ภาพที่ 17-19 แสดงถึงตัวอย่างส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำของข้อมูลจราจรทางคอมพิวเตอร์โปรโตคอล TCP SYN จากวันที่ 1 สิงหาคม 2551 ดังแสดงในภาพที่ 14 ซึ่งเป็นข้อมูลต้นแบบที่ต้องการแยกส่วนประกอบความถี่ด้วยวิธีเวฟเล็ต จากภาพที่ 17-19 แสดงถึงส่วนประกอบความถี่สูง (Detail) และส่วนประกอบความถี่ต่ำ (Approximation) ในระดับที่ 3, 7 และ 11 ตามลำดับ โดยการแยกส่วนประกอบความถี่ดังกล่าวทำงานในส่วนของ Wavelet Transformation

ผลการเปรียบเทียบสหสัมพันธ์ของส่วนประกอบความถี่

ดังนั้นเพื่อเป็นการวิเคราะห์ถึงลักษณะพฤติกรรมการส่งข้อมูลที่ได้จัดเก็บจากระบบเครือข่ายต้นทางว่าในแต่ละระดับของส่วนประกอบความถี่มีลักษณะของข้อมูลที่คล้ายคลึงกันหรือไม่ เราจึงได้ทำการทดลองเปรียบเทียบค่าสหสัมพันธ์ของส่วนประกอบความถี่ในแต่ละระดับ โดยทำการเปรียบเทียบส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำของข้อมูลจราจรทางคอมพิวเตอร์ในแต่ละวันกับข้อมูลจราจรทางคอมพิวเตอร์ในวันอื่น ๆ โดยแยกตามระดับของส่วนประกอบความถี่และแยกตามโปรโตคอล



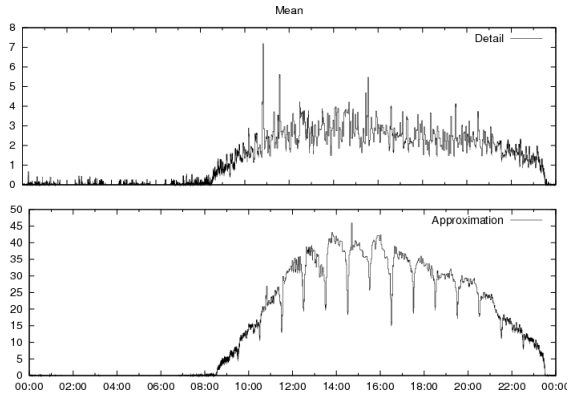
ภาพที่ 20 ตัวอย่างสหสัมพันธ์ของส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำ

ภาพที่ 20 แสดงให้เห็นถึงตัวอย่างของค่าสหสัมพันธ์ของส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำในแต่ละระดับของโปรโตคอล TCP SYN โดยเปรียบเทียบค่าสหสัมพันธ์ระหว่างวันที่ 1 สิงหาคม 2551 กับวันที่ 4 สิงหาคม 2551 เห็นได้ว่าเมื่อแยกส่วนประกอบความถี่ใน

ระดับที่สูงขึ้นค่าสหสัมพันธ์ในส่วนประกอบความถี่ต่ำก็จะเพิ่มขึ้นเข้าใกล้ค่า 1 แต่ส่วนประกอบความถี่สูงเมื่อแยกส่วนประกอบความถี่ในระดับที่สูงขึ้นค่าสหสัมพันธ์จะเพิ่มขึ้นแต่เพิ่มขึ้นในอัตราส่วนเพียงเล็กน้อยเมื่อเทียบกับส่วนประกอบความถี่ต่ำ

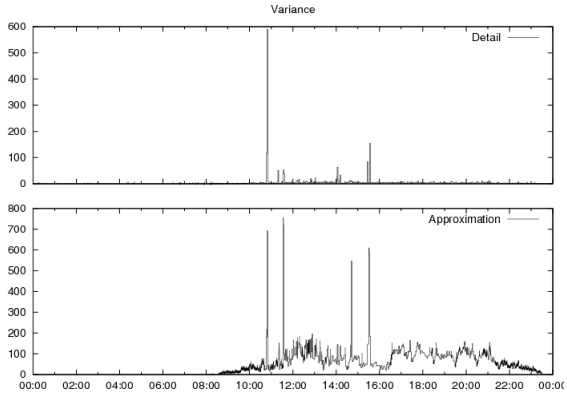
ผลการสร้างฐานข้อมูลเพื่อใช้ในการตรวจจับ

งานวิจัยนี้ได้ใช้ส่วนประกอบความถี่สูงและส่วนประกอบความที่ต่ำที่ระดับ 1-11 มาใช้ในการวิเคราะห์เพื่อเป็นแนวทางในการตรวจจับการโจมตีแบบฟลัดดิ้งภายในระบบเครือข่ายคอมพิวเตอร์ จึงมีข้อมูลผลการทดลองที่ได้เป็นจำนวนมากไม่สามารถแสดงได้ทั้งหมด ดังนั้นจึงแสดงตัวอย่างเฉพาะการทดลองของโปรโตคอล TCP SYN ในระดับที่ 7 เพียงระดับเดียว โดยการสร้างฐานข้อมูลของส่วนประกอบความถี่ในระดับที่ 7 ดังแสดงในภาพที่ 21 ถึง 23 นั้นได้มาจากการประมวลผลข้อมูลระหว่างวันที่ 1 ถึง 29 สิงหาคม 2551 โดยประมวลผลเฉพาะวันจันทร์ถึงวันศุกร์ที่มีการเปิดใช้งานปกติ

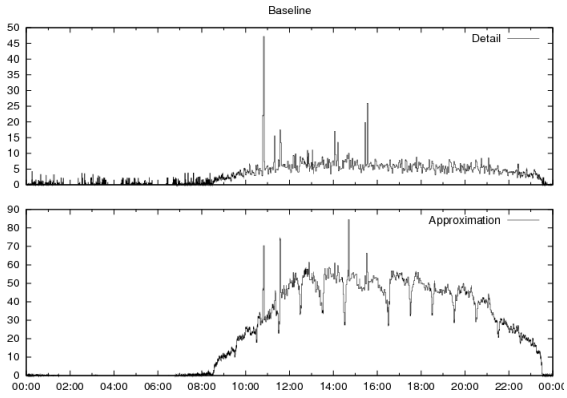


ภาพที่ 21 เส้นค่าเฉลี่ยของส่วนประกอบความถี่สูง (บน) และส่วนประกอบความถี่ต่ำ (ล่าง)

ภาพที่ 21 แสดงถึงเส้นที่ได้จากการคำนวณหาค่าเฉลี่ยของส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำของโปรโตคอล TCP SYN ในระดับที่ 7 ภาพที่ 22 แสดงถึงเส้นที่ได้จากการคำนวณหาค่าความแปรปรวนของส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำของโปรโตคอล TCP SYN ในระดับที่ 7 ระหว่างวันที่ 1 ถึง 29 สิงหาคม 2551 ภาพที่ 23 เป็นเส้นฐานของส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำที่ใช้สำหรับการตรวจจับ โดยเส้นฐานดังกล่าวจะถูกคำนวณมาจากค่าเฉลี่ยและค่าความแปรปรวน ในงานวิจัยนี้ได้คำนวณเส้นฐานเพื่อใช้ในการตรวจจับโดยใช้ค่าความเชื่อมั่นที่ 95%



ภาพที่ 22 เส้นค่าความแปรปรวนส่วนประกอบความถี่สูง (บน) และส่วนประกอบความถี่ต่ำ (ล่าง)

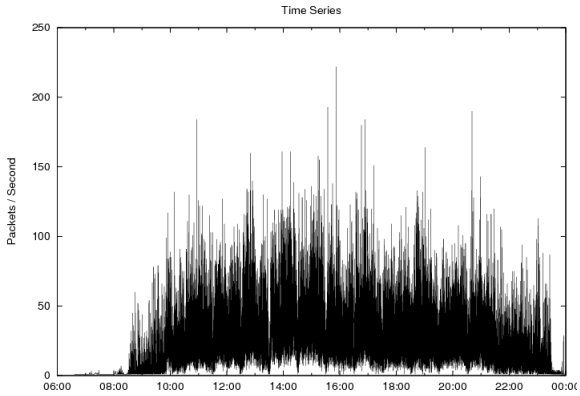


ภาพที่ 23 เส้นฐานส่วนประกอบความถี่สูง (บน) และส่วนประกอบความถี่ต่ำ (ล่าง)

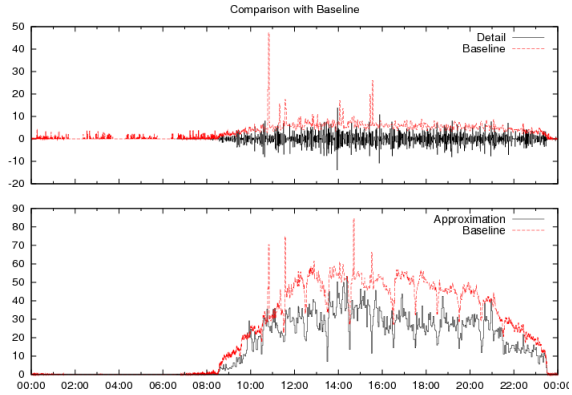
การทดลองในงานวิจัยนี้ได้ทำการคำนวณหาค่าเส้นเฉลี่ยของส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำในระดับต่าง ๆ เช่นเดียวกันกับการคำนวณเพื่อหาค่าต่าง ๆ ในส่วนประกอบความถี่ระดับที่ 7 ตามที่กล่าวมาข้างต้น

ผลการตรวจจับความผิดปกติ

ในการทดลองได้ทำการคำนวณเส้นค่าเฉลี่ย, เส้นค่าความแปรปรวนและเส้นฐานของส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำในแต่ละระดับ โดยทำการคำนวณจากข้อมูลระหว่างวันที่ 1 ถึง 29 สิงหาคม 2551 จากนั้นจึงนำข้อมูลจากเดือนอื่นมาทำการจำลองส่งชุดข้อมูลจำนวนมากออกจากระบบเครือข่าย เพื่อพิจารณาถึงจำนวนชุดข้อมูลต่อเวลาที่สามารถตรวจจับได้ และพิจารณาถึงช่วงระยะเวลาการโจมตีแบบฟลัดดิ้งที่สามารถตรวจจับได้ของส่วนประกอบความถี่ในแต่ละระดับ ดังนั้นจึงนำเสนอตัวอย่างการจำลองการโจมตีแบบฟลัดดิ้งและการตรวจจับการโจมตีแบบฟลัดดิ้งจากส่วนประกอบความถี่ในระดับที่ 7 เพียงตัวอย่างเดียว ในส่วนประกอบความถี่ระดับอื่น ๆ ก็ใช้แนวทางเดียวกันนี้ในการทดลอง

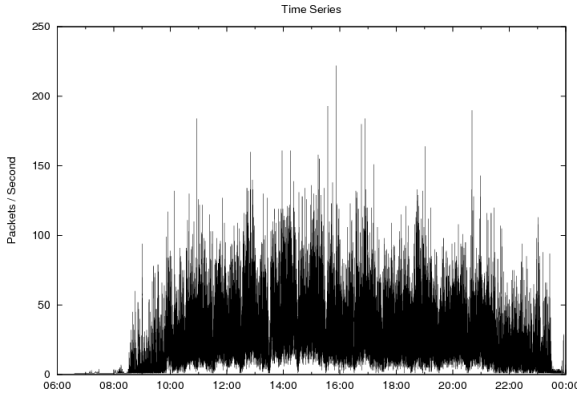


ภาพที่ 24 ข้อมูลจราจรทางคอมพิวเตอร์ที่ยังไม่จำลองการโจมตีแบบฟลัดดิ้ง

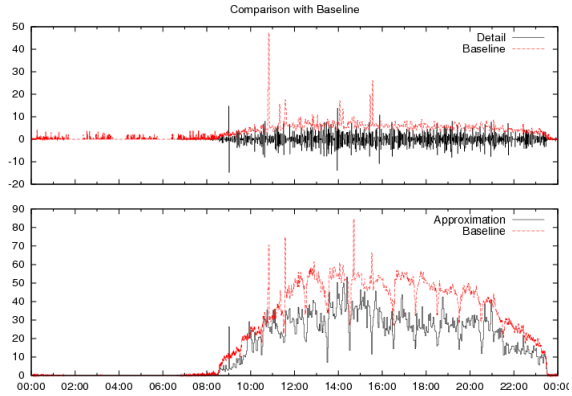


ภาพที่ 25 ส่วนประกอบความถี่สูง (บน) และส่วนประกอบความถี่ต่ำ (ล่าง) เปรียบเทียบกับเส้นฐาน

จากภาพที่ 24 แสดงให้เห็นถึงข้อมูลจราจรทางคอมพิวเตอร์ของโปรโตคอล TCP SYN ที่ยังไม่มีการแยกส่วนประกอบความถี่ เมื่อทำการแยกส่วนประกอบความถี่แล้วและนำมาเปรียบเทียบกับเส้นฐานที่มีการคำนวณเก็บเอาไว้ดังแสดงในภาพที่ 25 เห็นได้ว่าส่วนประกอบความถี่ต่ำของข้อมูลจราจรทางคอมพิวเตอร์ในระดับที่ 7 จะอยู่ใต้เส้นฐาน แสดงให้เห็นว่าในขณะที่ไม่มีการจำลองการโจมตีแบบฟลัดดิ้งระบบไม่สามารถตรวจพบความผิดปกติภายในระบบเครือข่าย



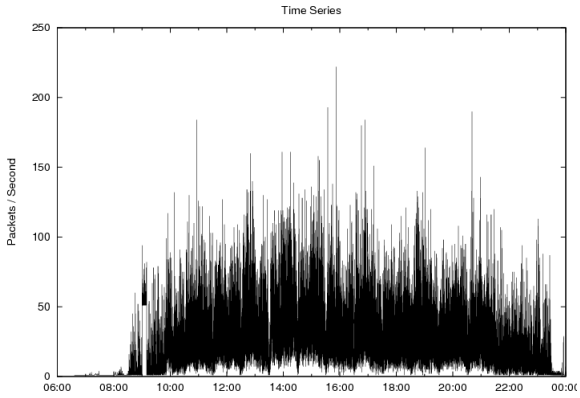
ภาพที่ 26 ข้อมูลจราจรทางคอมพิวเตอร์ที่จำลองการโจมตีแบบฟลัดดิ้งแบบที่ 1



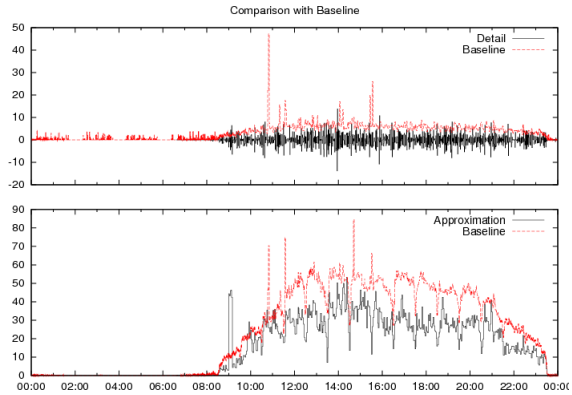
ภาพที่ 27 ส่วนประกอบความถี่ของการโจมตีแบบฟลัดดิ้งแบบที่ 1 เปรียบเทียบกับเส้นฐาน

ภาพที่ 26 แสดงถึงตัวอย่างของการจำลองการโจมตีแบบฟลัดดิ้งที่เวลา 9.00 น. เป็นระยะเวลา 1 นาที โดยมีการจำลองการโจมตีแบบฟลัดดิ้งขนาด 50 ชุดข้อมูลต่อวินาทีออกจากระบบเครือข่ายคอมพิวเตอร์ ภาพที่ 27 แสดงส่วนประกอบความถี่ในระดับที่ 7 ของการจำลองการ

โจมตีแบบฟลัดดิ้งดังกล่าว เห็นได้เราไม่สามารถแยกความแตกต่างระหว่างข้อมูลจราจรทางคอมพิวเตอร์ที่ปกติกับข้อมูลจราจรทางคอมพิวเตอร์ที่มีการจำลองการโจมตีแบบฟลัดดิ้งได้

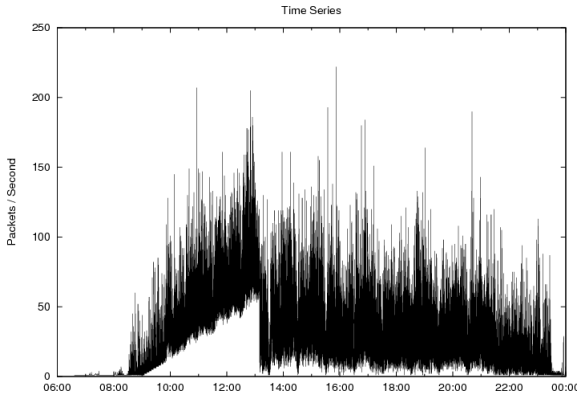


ภาพที่ 28 ข้อมูลจราจรทางคอมพิวเตอร์ที่จำลองการโจมตีแบบฟลัดดิ้งแบบที่ 2

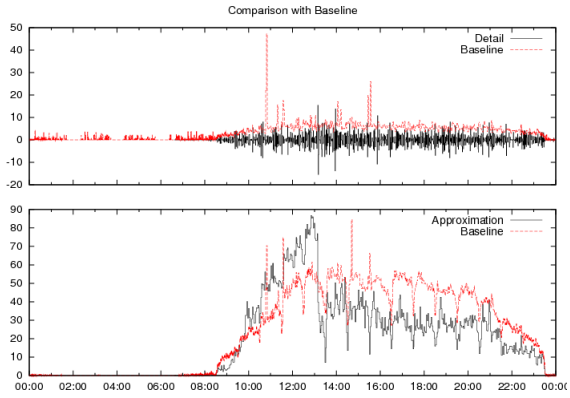


ภาพที่ 29 ส่วนประกอบความถี่ของการโจมตีแบบฟลัดดิ้งแบบที่ 2 เปรียบเทียบกับเส้นฐาน

ภาพที่ 28 แสดงถึงตัวอย่างของการจำลองการโจมตีแบบฟลัดดิ้งที่เวลา 9.00 น. เป็นระยะเวลา 10 นาที โดยมีการจำลองการโจมตีแบบฟลัดดิ้งขนาด 50 ชุดข้อมูลต่อวินาทีออกจากระบบเครือข่ายคอมพิวเตอร์ ภาพที่ 29 แสดงส่วนประกอบความถี่ในระดับที่ 7 ของการจำลองการโจมตีแบบฟลัดดิ้งดังกล่าว



ภาพที่ 30 ข้อมูลจราจรทางคอมพิวเตอร์ที่จำลองการโจมตีแบบฟลัดดิ้งแบบที่ 3



ภาพที่ 31 ส่วนประกอบความถี่ของการโจมตีแบบฟลัดดิ้งแบบที่ 3 เปรียบเทียบกับเส้นฐาน

ภาพที่ 30 แสดงถึงตัวอย่างของการจำลองการโจมตีแบบฟลัดดิ้งที่เวลา 9.00 น. เป็นจำลองการโจมตีแบบฟลัดดิ้งโดยเริ่มจาก 1 ชุดข้อมูลต่อวินาทีเข้าไปในระบบเครือข่ายเป็นระยะเวลา 5 นาที จากนั้นจึงเพิ่มเป็น 2 ชุดข้อมูลต่อวินาทีเข้าไปในระบบเครือข่ายเป็นระยะเวลา 5 นาที และเพิ่มชุดข้อมูลเช่นนี้ไปเรื่อย ๆ จนถึง 50 ชุดข้อมูลต่อวินาที ภาพที่ 31 แสดงส่วนประกอบความถี่ในระดับที่ 7 ของการจำลองการโจมตีแบบฟลัดดิ้งดังกล่าว

วิจารณ์

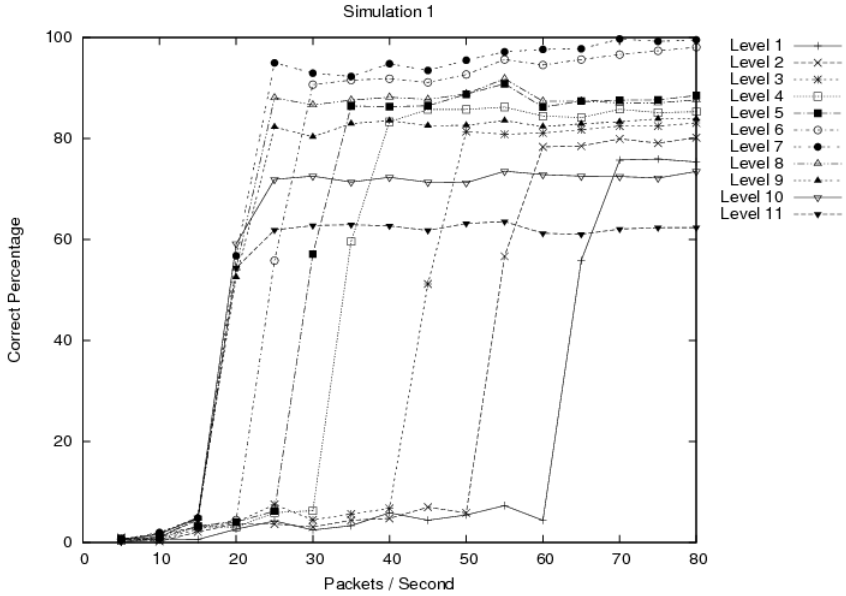
จากผลการวิเคราะห์เบื้องต้นพบว่าลักษณะการใช้งานจากห้องปฏิบัติการคอมพิวเตอร์และอินเตอร์เน็ตมีการใช้งานเพื่อค้นคว้าหาข้อมูลจากเว็บไซต์ต่าง ๆ เป็นจำนวนมาก จึงทำให้โปรโตคอลปรากฏเห็นการใช้งานได้อย่างชัดเจนคือ TCP SYN เพราะเป็นโปรโตคอลที่ใช้ในร้องขอเพื่อสร้างการเชื่อมต่อก่อนที่จะทำการรับส่งข้อมูล ซึ่งเป็นโปรโตคอลที่มีการใช้งานมากโปรโตคอลหนึ่งในระบบเครือข่ายอินเตอร์เน็ต ตัวอย่างโปรโตคอลที่มีการเรียกใช้โปรโตคอล TCP SYN ได้แก่ HTTP, FTP เป็นต้น โปรโตคอล TCP SYN/ACK เป็นโปรโตคอลที่ใช้ในการตอบการร้องขอเพื่อสร้างการเชื่อมต่อ โดยทั่วไปจะถูกส่งออกจากเครื่องคอมพิวเตอร์แม่ข่ายที่มีการถูกร้องขอใช้งาน แต่เครื่องคอมพิวเตอร์ที่ใช้งานในห้องปฏิบัติการคอมพิวเตอร์และอินเตอร์เน็ตส่วนใหญ่เป็นเครื่องคอมพิวเตอร์ลูกข่ายเพื่อให้บริการกับนักศึกษา จึงมีส่วนที่เป็นโปรโตคอล TCP SYN/ACK จำนวนน้อย ส่วนการใช้งานทางด้านอื่นมีการใช้งานน้อยทำให้โปรโตคอลอื่น ๆ มีใช้งานน้อยซึ่งได้แก่ ICMP และ UDP ดังนั้นงานวิจัยนี้จึงใช้โปรโตคอล TCP SYN จำลองการโจมตีแบบฟลัดดิ้ง

จากตารางที่ 2 ถึงตารางที่ 5 แสดงถึงค่าสัมประสิทธิ์สหสัมพันธ์ของโปรโตคอล ICMP, TCP SYN, TCP SYN/ACK และ UDP ตามลำดับ เห็นได้ว่าโปรโตคอล TCP SYN ที่เกิดขึ้นในแต่ละวันมีความสัมพันธ์กับวันอื่น ๆ เนื่องจากค่าสัมประสิทธิ์สหสัมพันธ์ของโปรโตคอลที่แสดงในตารางที่ 3 มีค่าประมาณ 0.3 ถึง 0.6 ส่วนค่าสัมประสิทธิ์สหสัมพันธ์ของโปรโตคอลอื่นมีค่าประมาณ 0 สามารถกล่าวได้ว่าการใช้งานระบบเครือข่ายคอมพิวเตอร์ในห้องปฏิบัติการอินเตอร์เน็ตและคอมพิวเตอร์มีการใช้งานโปรโตคอล TCP SYN ที่มีลักษณะรูปแบบคล้ายคลึงกันในแต่ละวัน ส่วนโปรโตคอลอื่นมีการใช้งานที่มีรูปแบบแตกต่างกันในแต่ละวัน ดังนั้นในงานวิจัยนี้จึงใช้โปรโตคอล TCP SYN ในการทดลองเพื่อทดสอบการตรวจจับการโจมตีแบบฟลัดดิ้ง

หลังจากนำสัญญาณที่ได้จากระบบเครือข่ายคอมพิวเตอร์มาแยกส่วนประกอบความถี่ด้วยวิธีเวฟเล็ตแล้ว ที่ระดับของการแยกส่วนประกอบความถี่มากขึ้นส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำของแต่ละวันจะมีรูปแบบที่คล้ายคลึงกันมากขึ้น สังเกตได้จากค่าสัมประสิทธิ์สหสัมพันธ์ของแต่ละระดับมีค่าเพิ่มขึ้นดังแสดงในภาพที่ 20 ซึ่งแสดงถึงความสัมพันธ์ระหว่างค่าสัมประสิทธิ์สหสัมพันธ์กับส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำในระดับ

ที่ 1 ถึงระดับที่ 11 ตามลำดับ เราจึงใช้ส่วนประกอบความถี่ต่ำในการตรวจจับการโจมตีแบบฟลัดดิ้ง

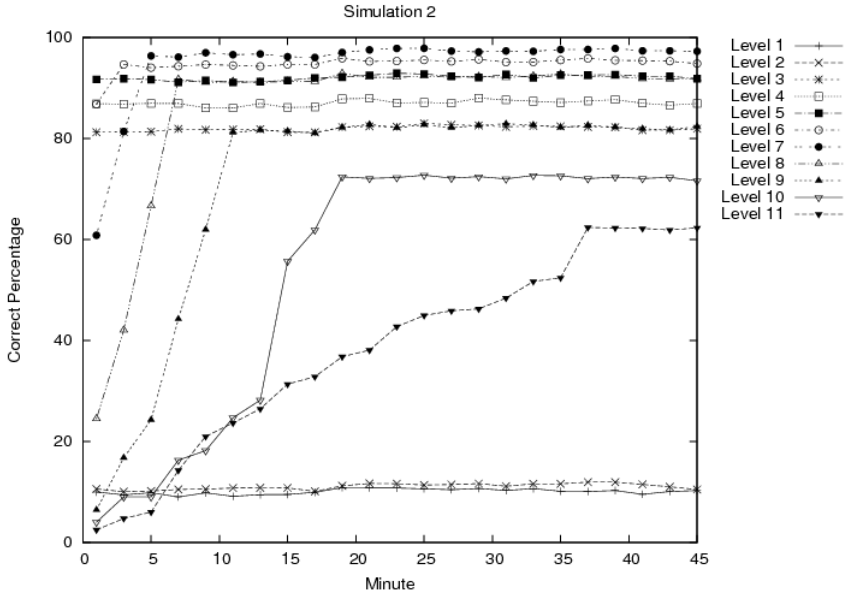
จากการจำลองการโจมตีแบบฟลัดดิ้งแบบที่ 1 โดยจำลองการโจมตีแบบฟลัดดิ้งเข้าไปในระบบเครือข่ายเป็นระยะเวลา 1 นาที โดยเริ่มจากเวลา 9.00-23.00 น. มีระยะห่างแต่ละช่วงการจำลองเป็นเวลา 1 ชั่วโมง โดยทำการโจมตีแบบฟลัดดิ้งระหว่าง 5-80 ชุดข้อมูลต่อวินาทีและเพิ่มขึ้นครั้งละ 5 ชุดข้อมูลต่อวินาที ได้ผลการจำลองดังแสดงในภาพที่ 32



ภาพที่ 32 ความสัมพันธ์ระหว่างเปอร์เซ็นต์ความถูกต้องกับจำนวนชุดข้อมูลต่อวินาที

ภาพที่ 32 แสดงให้เห็นถึงความสัมพันธ์ระหว่างเปอร์เซ็นต์ความถูกต้องเฉลี่ยที่ระบบสามารถตรวจจับการโจมตีแบบฟลัดดิ้งได้กับขนาดของชุดข้อมูลต่อวินาทีของส่วนประกอบความถี่ในแต่ละระดับ จากภาพแสดงให้เห็นว่าส่วนประกอบความถี่ในระดับสูงสามารถตรวจจับการโจมตีแบบฟลัดดิ้งได้โดยมีจำนวนชุดข้อมูลต่อช่วงเวลาที่น้อยกว่าเมื่อเทียบกับส่วนประกอบความถี่ต่ำ ในขณะที่ส่วนประกอบความถี่ในระดับที่ 7 มีเปอร์เซ็นต์ความถูกต้องเฉลี่ยสูงที่สุดเมื่อเทียบกับส่วนประกอบความถี่ในระดับอื่น ๆ

จากการจำลองการโจมตีแบบฟลัดดิ้งแบบที่ 2 โดยจำลองการโจมตีแบบฟลัดดิ้งเข้าไปในระบบเครือข่ายเป็นระยะเวลาระหว่าง 1-45 นาที โดยเริ่มจากเวลา 9.00-23.00 น. มีระยะห่างแต่ละช่วงการจำลองเป็นเวลา 1 ชั่วโมง โดยทำการโจมตีแบบฟลัดดิ้งที่ 50 ชุดข้อมูลต่อวินาที ได้ผลการจำลองดังแสดงในภาพที่ 33

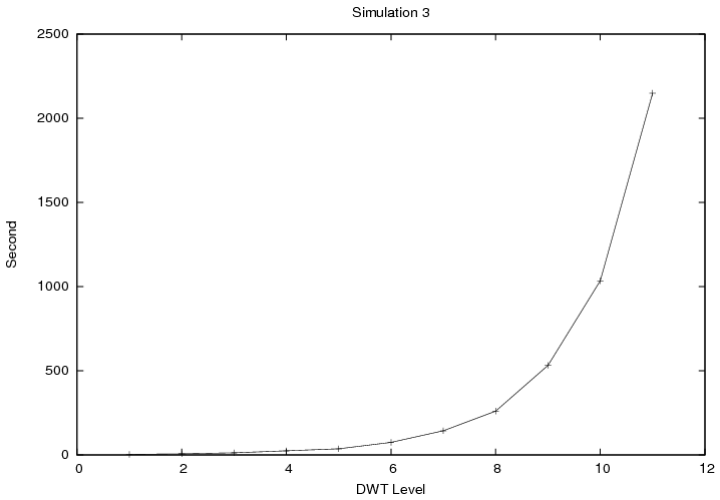


ภาพที่ 33 ความสัมพันธ์ระหว่างเปอร์เซ็นต์ความถูกต้องกับช่วงเวลาที่เกิดการโจมตีแบบฟลัดดิ้ง

ภาพที่ 33 แสดงให้เห็นถึงความสัมพันธ์ระหว่างเปอร์เซ็นต์ความถูกต้องเฉลี่ยที่ระบบสามารถตรวจจับการโจมตีแบบฟลัดดิ้งได้กับช่วงเวลาที่เกิดการโจมตีแบบฟลัดดิ้งของส่วนประกอบความถี่ในแต่ละระดับ จากภาพแสดงให้เห็นว่าส่วนประกอบความถี่ในระดับสูงสามารถตรวจจับการโจมตีแบบฟลัดดิ้งได้โดยมีเปอร์เซ็นต์ความถูกต้องเริ่มเข้าสู่จุดสูงสุดเมื่อมีช่วงเวลาที่เกิดการโจมตีแบบฟลัดดิ้งน้อยกว่าเมื่อเทียบกับส่วนประกอบความถี่ต่ำ ในขณะที่ส่วนประกอบความถี่ในระดับที่ 7 มีเปอร์เซ็นต์ความถูกต้องเฉลี่ยสูงที่สุดเมื่อเทียบกับส่วนประกอบความถี่ในระดับอื่น ๆ

จากการจำลองการโจมตีแบบฟลัดดิ้งทั้งสองการทดลองพบว่า เมื่อพิจารณาส่วนประกอบความถี่ที่มีระดับสูงขึ้นทำให้เราสามารถสร้างเส้นฐานจากส่วนประกอบความถี่ที่มีลักษณะคล้ายคลึงกัน นั่นหมายความว่าเราสามารถตรวจจับการโจมตีแบบฟลัดดิ้งในระบบเครือข่ายต้นทางได้ ถึงแม้ว่าการโจมตีแบบฟลัดดิ้งดังกล่าวมีจำนวนการส่งชุดข้อมูลเพียงเล็กน้อยต่อช่วงเวลา แต่เมื่อพิจารณาส่วนประกอบความถี่ที่มีระดับสูงขึ้นจะทำให้เราไม่สามารถตรวจจับการโจมตีแบบฟลัดดิ้งที่เกิดขึ้นในเวลาสั้นได้ จากการทดลองในงานวิจัยนี้ได้พิจารณาส่วนประกอบความถี่ที่ระดับที่ 7 พบว่าเป็นส่วนประกอบความถี่ที่เหมาะสมในการตรวจจับการโจมตีแบบฟลัดดิ้งที่เครือข่ายต้นทาง โดยสามารถตรวจจับจำนวนการส่งชุดข้อมูลที่มีขนาดมากกว่า 25 ชุดข้อมูลต่อวินาทีได้ และสามารถตรวจจับการส่งชุดข้อมูลที่เกิดขึ้นเป็นเวลามากกว่า 5 นาทีได้โดยทำให้มีเปอร์เซ็นต์ความถูกต้องในการตรวจจับมากที่สุด

จากการจำลองการโจมตีแบบฟลัดดิ้งแบบที่ 3 โดยจำลองการโจมตีแบบฟลัดดิ้งโดยเริ่มจาก 1 ชุดข้อมูลต่อวินาทีเข้าไปในระบบเครือข่ายเป็นระยะเวลา 5 นาที จากนั้นจึงเพิ่มเป็น 2 ชุดข้อมูลต่อวินาทีเข้าไปในระบบเครือข่ายเป็นระยะเวลา 5 นาที และเพิ่มชุดข้อมูลเช่นนี้ไปเรื่อย ๆ จนถึง 50 ชุดข้อมูลต่อวินาที เริ่มจากเวลา 9.00-23.00 น. มีระยะห่างแต่ละช่วงการจำลองเป็นเวลา 1 ชั่วโมง ได้ผลการจำลองดังแสดงในภาพที่ 34



ภาพที่ 34 ความสัมพันธ์ระหว่างช่วงเวลาหน่วงกับระดับของส่วนประกอบความถี่

ภาพที่ 34 แสดงให้เห็นถึงความสัมพันธ์ระหว่างช่วงเวลาหน่วงระหว่างเวลาที่เกิดการโจมตีแบบฟลัดดิ้งกับเวลาที่ระบบสามารถตรวจจับได้ในแต่ละระดับของส่วนประกอบความถี่ จากภาพแสดงให้เห็นว่าเมื่อใช้ระดับของส่วนประกอบความถี่ที่สูงขึ้นทำให้เกิดช่วงเวลาหน่วงระหว่างเวลาที่เกิดการส่งชุดข้อมูลกับเวลาที่ระบบสามารถตรวจจับได้เพิ่มขึ้นในลักษณะเอ็กโปเนนเชี่ยล หมายความว่าถ้าเราใช้ส่วนประกอบความถี่ที่สูงขึ้นจะทำให้ระบบตรวจจับการเกิดการโจมตีแบบฟลัดดิ้งได้ช้าลง

ประโยชน์ที่ได้จากงานวิจัยนี้คือแนวคิดในการตรวจจับความผิดปกติภายในระบบเครือข่ายคอมพิวเตอร์ ซึ่งสามารถตรวจจับความผิดปกติที่เกิดจากการโจมตีแบบฟลัดดิ้ง เช่น การโจมตีแบบ DoS หรือ DDoS เป็นต้น โดยแนวคิดในการตรวจจับดังกล่าวสามารถนำไปประยุกต์ใช้ตรวจจับความผิดปกติที่ขาออกของระบบเครือข่ายคอมพิวเตอร์ได้ อีกทั้งสามารถป้องกันไม่ให้ระบบเครือข่ายคอมพิวเตอร์เข้าไปมีส่วนร่วมในการโจมตีไปยังระบบเครือข่ายคอมพิวเตอร์อื่น และช่วยให้สามารถติดตามรอยของคนร้ายได้อย่างรวดเร็ว

สรุปและข้อเสนอแนะ

สรุป

การนำเสนอวิธีตรวจจับการโจมตีแบบฟลัดดิ้งที่เครือข่ายต้นทางในงานวิจัยนี้ จากผลการทดลองได้แสดงให้เห็น การใช้เวฟเล็ตเพื่อแยกส่วนประกอบความถี่สูงและส่วนประกอบความถี่ต่ำนำไปตรวจจับการโจมตีแบบฟลัดดิ้งที่ระบบเครือข่ายต้นทางได้ โดยในการตรวจจับด้วยวิธีดังกล่าวมีระดับของการแยกส่วนประกอบความถี่เป็นปัจจัยหลักที่มีผลต่อจำนวนชุดข้อมูลต่อเวลาและช่วงระยะเวลาการโจมตีแบบฟลัดดิ้งที่สามารถตรวจจับได้

ถ้าใช้ส่วนประกอบความถี่ในระดับต่ำ เช่น ส่วนประกอบความถี่ในระดับที่ 1-3 ระบบสามารถตรวจจับการโจมตีแบบฟลัดดิ้งที่เกิดเป็นระยะเวลาสั้นได้ แต่ไม่สามารถตรวจจับการส่งชุดข้อมูลที่มีการส่งชุดจำนวนข้อมูลเป็นจำนวนเพียงเล็กน้อยต่อเวลาได้ ในทางกลับกันถ้าใช้ส่วนประกอบความถี่ในระดับสูง เช่น ส่วนประกอบความถี่ในระดับที่ 9-11 ระบบสามารถตรวจจับการส่งชุดข้อมูลที่มีการส่งชุดข้อมูลเป็นจำนวนเพียงเล็กน้อยต่อช่วงเวลาได้ แต่ไม่สามารถตรวจจับการโจมตีแบบฟลัดดิ้งที่เกิดเป็นระยะเวลาสั้นได้

ผลจากการทดลองชี้ให้เห็นว่าส่วนประกอบความถี่ที่ระดับที่ 7 เป็นส่วนประกอบความถี่ที่เหมาะสมในการตรวจจับการโจมตีแบบฟลัดดิ้งที่เครือข่ายต้นทางมากที่สุด โดยสามารถตรวจจับจำนวนการส่งชุดข้อมูลที่มีขนาดมากกว่า 25 ชุดข้อมูลต่อวินาทีได้ และสามารถตรวจจับการส่งชุดข้อมูลที่เกิดขึ้นเป็นเวลามากกว่า 5 นาทีได้ โดยการตรวจจับที่ส่วนประกอบความถี่ดังกล่าวจะทำให้มีเปอร์เซ็นต์ความถูกต้องในการตรวจจับมากที่สุด

การนำวิธีการตรวจจับที่นำเสนอในงานวิจัยนี้ไปประยุกต์ใช้ จำเป็นต้องมีการเก็บข้อมูลจากระบบเครือข่ายคอมพิวเตอร์เป็นระยะเวลาหนึ่งก่อน เพื่อให้เกิดความมั่นใจได้ว่าข้อมูลที่ถูกจัดเก็บเพื่อที่นำไปใช้ในการตรวจจับการโจมตีแบบฟลัดดิ้งนั้นเพียงพอที่จะนำไปสร้างเป็นเส้นฐานได้

ข้อเสนอแนะ

วิธีตรวจจับการโจมตีแบบฟลัดดิ้งที่เครือข่ายต้นทางด้วยวิธีเวฟเล็ตในงานวิจัยนี้มีลักษณะการทำงานแบบประมวลผลเชิงกลุ่ม (Batch Processing) นั่นหมายความว่าต้องมีการจัดเก็บข้อมูลให้สิ้นสุดในแต่ละวันก่อน จากนั้นจึงนำข้อมูลที่ได้ไปประมวลผลเพื่อหาความผิดปกติในระบบเครือข่าย ดังนั้นสิ่งที่ควรพัฒนาต่อคือประยุกต์วิธีตรวจจับการโจมตีแบบฟลัดดิ้งดังกล่าวให้มีลักษณะการทำงานแบบประมวลผลแบบทันที (Real Time Processing) ซึ่งทำให้สามารถตรวจจับความผิดปกติและสามารถแจ้งเตือนความผิดปกติได้อย่างทันท่วงที

สำหรับระบบเครือข่ายคอมพิวเตอร์ที่มีการใช้งานเป็นรูปแบบที่ไม่คงที่ มีลักษณะการใช้งานในแต่ละวันแตกต่างกัน อาจต้องใช้ข้อมูลเฉพาะของแต่ละวันมาประมวลผลหาเส้นฐานของแต่ละวัน ตัวอย่างเช่น ใช้เฉพาะข้อมูลของวันจันทร์มาสร้างเส้นฐานวันจันทร์ เมื่อถึงวันจันทร์ที่ต้องการตรวจจับการโจมตีแบบฟลัดดิ้งก็จะนำเส้นฐานของวันจันทร์ที่สร้างไว้มาตรวจสอบ การประมวลผลในแต่ละวันเช่นนี้จะทำให้มีความแม่นยำในการตรวจจับมากยิ่งขึ้นสำหรับระบบเครือข่ายคอมพิวเตอร์ที่ลักษณะข้อมูลในแต่ละวันที่มีความแตกต่างกัน

เอกสารและสิ่งอ้างอิง

Barford, P., J. Kline, D. Plonka and A. Ron. 2002. A signal analysis of network traffic anomalies. **IMW '02: Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement.** ACM : 71-82.

Blazek, R.B., H. Kim, B. Rozovskii and A. Tartakovsky. 2001. A Novel Approach to Detection of 'Denial-of-Service' Attacks via Adaptive Sequential and Batch-Sequential Change-Point Detection Methods. **Proc. IEEE Workshop Information Assurance and Security.** IEEE CS Press : 220–226.

Carl, G., R.R. Brooksb and S. Rai. 2006. Wavelet based Denial-of-Service detection. **Computers & Security** 25 (8): 600-615.

Lu, W., M. Tavallaee and A.A. Ghorbani. 2008 Detecting Network Anomalies Using Different Wavelet Basis Functions. **IEEE Computer Society** (0): 149-156.

Peng, T., C. Leckie and K. Ramamohanarao. 2007. Survey of network-based defense mechanisms countering the DoS and DDoS problems. **ACM Computer Survey** 39 (1): 3-44.

Thatte, G., U. Mitra, and J. Heidemann. 2008 Detection of low-rate attacks in computer networks. **Computer Communications Workshops 2008.** INFOCOM. IEEE Conference: 1-6.

ภาคผนวก

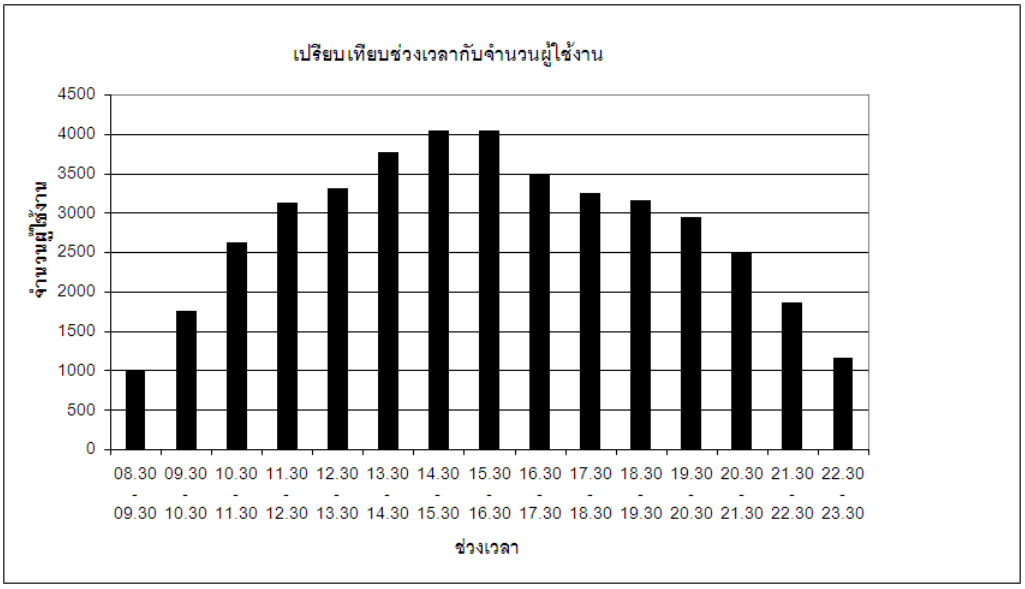
ภาคผนวก ก

สถิติการใช้งานห้องปฏิบัติการคอมพิวเตอร์และอินเตอร์เน็ต

สถิติการใช้งานเดือนมิถุนายน 2551

ตารางผนวกที่ ก1 จำนวนผู้ใช้งานแต่ละช่วงเวลาประจำเดือนมิถุนายน

วันที่	08.30-09.30	09.30-10.30	10.30-11.30	11.30-12.30	12.30-13.30	13.30-14.30	14.30-15.30	15.30-16.30	16.30-17.30	17.30-18.30	18.30-19.30	19.30-20.30	20.30-21.30	21.30-22.30	22.30-23.30
2	56	66	138	163	134	167	169	173	141	137	129	125	100	79	52
3	42	82	141	157	169	171	173	173	170	153	138	107	76	47	37
4	49	87	123	130	132	169	169	168	149	139	131	128	110	78	47
5	29	67	105	105	140	167	170	170	152	154	140	128	118	77	47
6	38	64	101	105	68	73	148	169	155	134	136	133	136	100	59
9	59	65	125	133	133	133	133	132	134	133	133	133	130	114	84
10	40	63	122	131	131	131	131	131	131	131	131	129	123	112	74
11	36	81	112	115	124	162	174	188	187	152	154	141	121	87	51
12	35	75	105	152	166	171	171	171	171	170	165	147	120	69	28
13	31	65	104	107	106	144	160	159	155	137	143	148	119	93	74
14	17	21	40	56	66	74	96	83	1	-	-	-	-	-	-
16	49	82	107	110	138	140	184	192	194	174	166	127	96	86	61
17	43	79	93	135	153	204	207	188	196	193	165	152	121	85	46
18	43	93	115	141	167	192	204	205	158	165	141	141	122	100	54
19	35	72	107	170	170	197	194	177	153	134	150	152	95	1	
20	48	92	157	169	167	181	205	196	190	147	154	136	109	97	58
21	6	30	63	66	104	105	105	101	-	1	-	-	-	-	-
23	59	101	142	167	172	170	170	201	203	200	186	183	164	142	77
24	51	84	86	104	156	170	204	205	195	180	184	167	134	106	64
25	49	84	103	139	158	170	170	174	168	169	170	155	127	96	72
26	48	82	105	154	169	168	169	169	135	127	137	138	119	77	49
27	43	68	104	103	104	153	169	167	163	147	130	112	109	96	59
28	20	33	44	64	18	98	93	82	-	-	-	-	-	-	-
30	31	70	104	155	169	164	168	164	167	169	168	166	155	119	72

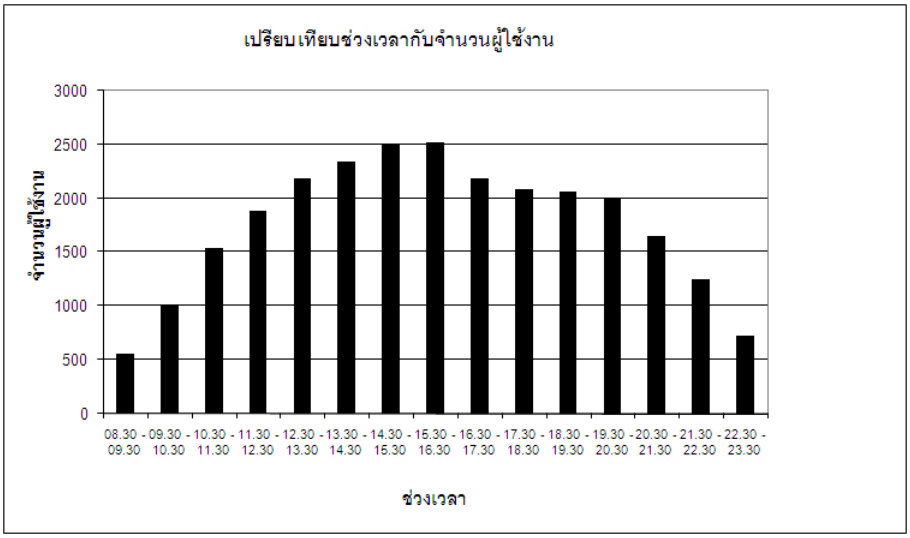


ภาพผนวกที่ ก1 จำนวนผู้ใช้งานรวมแต่ละช่วงเวลาเดือนมิถุนายน

สถิติการใช้งานเดือนกรกฎาคม 2551

ตารางผนวกที่ ก2 จำนวนผู้ใช้งานแต่ละช่วงเวลาประจำเดือนกรกฎาคม

วันที่	08.30-09.30	09.30-10.30	10.30-11.30	11.30-12.30	12.30-13.30	13.30-14.30	14.30-15.30	15.30-16.30	16.30-17.30	17.30-18.30	18.30-19.30	19.30-20.30	20.30-21.30	21.30-22.30	22.30-23.30
1	35	61	66	66	101	103	104	104	103	103	103	99	98	88	62
2	41	86	105	107	107	143	143	144	144	142	142	159	139	103	75
3	34	67	101	103	103	167	168	168	148	128	156	151	124	91	46
4	47	69	129	132	132	132	132	132	-	-	-	-	-	-	-
7	46	65	103	117	167	156	169	169	169	158	145	163	156	116	58
8	44	70	110	157	169	169	169	169	169	166	159	151	127	86	41
9	30	77	105	140	164	169	160	194	165	151	141	146	118	96	60
10	36	64	100	103	152	158	168	169	143	150	145	122	97	71	38
11	21	58	88	101	101	109	166	160	163	138	121	118	108	80	48
12	11	29	44	59	89	96	102	102	-	-	-	-	-	-	-
14	46	74	102	103	103	102	169	169	169	158	169	164	148	127	72
15	40	57	103	126	155	169	169	163	147	162	161	157	108	57	26
16	42	63	101	141	152	152	153	155	149	140	153	157	119	96	64
21	41	65	93	138	160	150	142	125	107	112	105	90	79	56	33
28	19	45	52	76	82	87	90	81	79	70	74	79	48	27	17
29	9	21	43	82	106	106	103	92	101	88	89	94	62	49	26
30	16	24	48	77	70	73	91	118	123	118	93	62	59	49	30
31	5	18	43	55	69	97	107	107	107	107	105	86	63	52	31

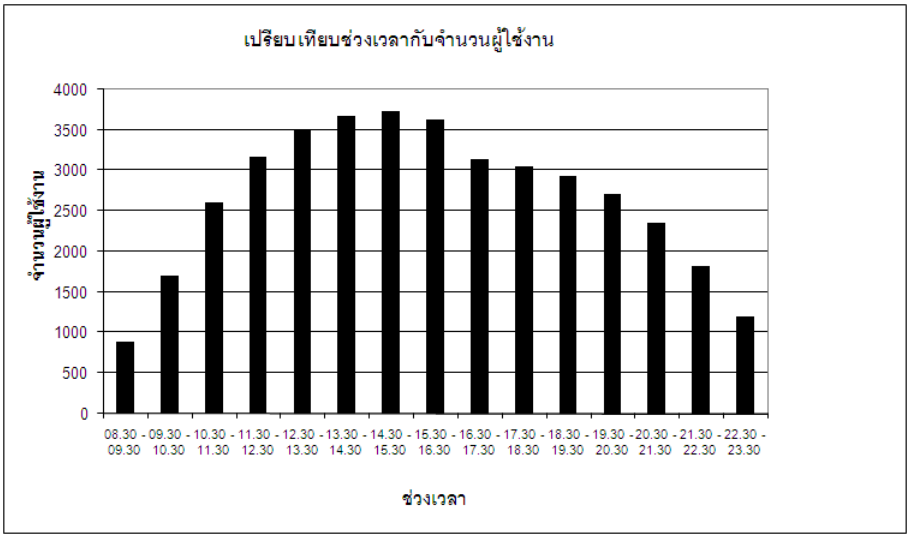


ภาพผนวกที่ ก2 จำนวนผู้ใช้งานรวมแต่ละช่วงเวลาเดือนกรกฎาคม

สถิติการใช้งานเดือนสิงหาคม 2551

ตารางผนวกที่ ก3 จำนวนผู้ใช้งานแต่ละช่วงเวลาประจำเดือนสิงหาคม

วันที่	08.30-09.30	09.30-10.30	10.30-11.30	11.30-12.30	12.30-13.30	13.30-14.30	14.30-15.30	15.30-16.30	16.30-17.30	17.30-18.30	18.30-19.30	19.30-20.30	20.30-21.30	21.30-22.30	22.30-23.30
1	20	34	63	85	107	127	138	149	114	95	86	70	64	64	41
2	6	23	43	51	65	85	101	76	-	-	-	-	-	-	-
4	44	70	94	150	163	151	173	173	173	173	168	159	142	97	68
5	35	77	131	173	171	173	172	173	171	173	166	142	105	61	48
6	40	75	115	149	172	170	171	172	172	158	156	135	113	94	67
7	27	60	104	106	152	172	172	172	168	146	147	129	106	66	35
8	27	68	106	108	108	110	108	108	144	134	105	90	75	62	56
11	47	83	119	144	162	169	174	172	148	138	126	110	98	85	67
13	46	100	143	174	174	174	174	174	174	172	172	170	129	101	81
14	35	65	102	108	144	174	174	174	174	174	173	157	139	104	66
15	38	73	102	106	109	107	122	142	142	145	128	149	143	83	54
16	18	29	39	70	103	101	83	-	-	-	-	-	-	-	-
18	52	95	136	158	163	174	174	174	174	174	174	170	158	138	86
19	54	98	120	135	135	174	174	174	173	171	166	159	118	106	71
20	52	85	135	157	174	174	173	174	173	172	171	163	157	127	74
21	46	81	139	149	174	174	174	174	174	174	169	142	120	101	63
22	46	67	101	142	152	174	174	174	-	-	-	-	-	-	-
25	39	106	173	174	174	173	174	174	173	174	174	171	165	136	73
26	61	102	157	174	174	174	174	174	174	173	174	171	154	120	72
27	43	103	145	174	174	174	174	174	171	174	170	167	144	104	65
28	44	80	129	172	174	174	174	174	174	174	174	169	150	111	71
29	42	86	135	168	174	174	174	174	173	157	140	93	80	65	42
30	12	20	45	65	94	107	107	105	-	-	-	-	-	-	-



ภาพผนวกที่ ก3 จำนวนผู้ใช้งานรวมแต่ละช่วงเวลาเดือนสิงหาคม

ภาคผนวก ข

โปรแกรมภาษา C

โปรแกรมจัดเก็บข้อมูลจากระบบเครือข่ายคอมพิวเตอร์

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include <string.h>

const int rowtable = 65536; //Max event in reference table.
int tpointer = 0;
#define N_ROW 65536

struct packetinfo {
	int index;
	char sadd[16];
	char dadd[16];
	int sport;
	int dport;
} packettable[N_ROW];

int checkevent(char[16], char[16], int, int);
void cleardata(int n);

int main(int argc, char *argv[]) {
	//Check argument.
	if(argc < 3) {
		printf("usage: %s <packetfile> <timeratio>\n", argv[0]);
		return -1;
	}//End check argument.
	char *datfile = argv[1];
	int timeratio = atoi(argv[2])/10000;

	cleardata(rowtable);

	double x, y;
	char line[90];
	char *sub_string;
	char *time;
	int isIndex = 1;
	int eventcount = 0;
	int nattack = 10;
	int attackcount[nattack];
	int c;
	for (c = 0; c < nattack; c++) attackcount[c] = 0;

	//Read input file to data1[i].
	FILE *tFile;
	char srcadd[16], dstadd[16];
	int srcport, dstport;
	isIndex = 1;
	int i = 0, j = 0;
	int k = 0, n = 0;
	if((tFile = fopen(datfile, "r")) != NULL) {
		while(fgets(line, 90, tFile) != NULL) {
			time = strtok(line, ",");
			while((sub_string = strtok(NULL, "\n\t ")) != NULL) {
```

```
                                        if(isIndex == 1) {
                                        i = atoi(sub_string)/timeratio;

                                                if(k != 0 && i != j) {
                                                        printf("%d ", n);
                                                        printf("\n");
                                                        cleardata(tpointer);
                                                        n = 0;
                                                        eventcount = 0;
                                                        for (c = 0; c < nattack; c++) attackcount[c] = 0;
                                                }
                                                if(k == 0 || i != j) {//First packet.
                                                        printf("%s, %d\t", time, i);
                                                        j = i; k++;
                                                }
                                                isIndex = 2;
                                        } else if(isIndex == 2) {
                                                strcpy(srcadd,sub_string);
                                                isIndex = 3;
                                        } else if(isIndex == 3) {
                                                strcpy(dstadd,sub_string);
                                                isIndex = 4;
                                        } else if(isIndex == 4) {
                                                srcport = atoi(sub_string);
                                                isIndex = 5;
                                        } else if(isIndex == 5) {
                                                dstport = atoi(sub_string);
                                                isIndex = 1;
                                                n++;
                                                attackcount[checkevent(srcadd, dstadd, srcport,
dstport)]++;
                                        }
                        }
                    }
            }
            fclose(tFile);

                printf("%d ", n);
                printf("\n");
                return 0;
}

int checkevent(char srcadd[16], char dstadd[16], int srcport, int dstport) {
                int i;

                for(i = 0; i < tpointer; i++) {
                        //Scan Port1.1                          src -, dst -, sport -, dport *
                        if(strcmp(srcadd,packettable[i].sadd) == 0 && strcmp(dstadd,packettable[i].dadd) ==
0 && srcport == packettable[i].sport) {return 1;}
                        //Scan Portm.1                          src -, dst *, sport -, dport -
                        else if(strcmp(srcadd,packettable[i].sadd) == 0 && srcport == packettable[i].sport
&& dstport == packettable[i].dport) {return 2;}
                        //DoS.1                                 src -, dst -, sport *, dport -
                        else if(strcmp(srcadd,packettable[i].sadd) == 0 &&
strcmp(dstadd,packettable[i].dadd) == 0 && dstport == packettable[i].dport) {return 3;}
                        //Scan Port1.2                          src -, dst -, sport *, dport *
```

```
                else if(strcmp(srcadd,packettable[i].sadd) == 0 &&
strcmp(dstadd,packettable[i].dadd) == 0) {return 4;}
                //Scan Portm.2                  src -, dst *, sport *, dport -
                else if(strcmp(srcadd,packettable[i].sadd) == 0 && dstport == packettable[i].dport)
{return 5;}
                //DoS.2                         src -, dst -, sport * dport *
                else if(strcmp(srcadd,packettable[i].sadd) == 0 &&
strcmp(dstadd,packettable[i].dadd) == 0) {return 6;}
                //Flash Crowd, DDoS.1           src *, dst -, sport *, dport -
                else if(strcmp(dstadd,packettable[i].dadd) == 0 && dstport == packettable[i].dport)
{return 7;}
                //DDoS.2                        src *, dst -, sport *, dport *
                else if(strcmp(dstadd,packettable[i].dadd) == 0) {return 8;}
                //Worm, Virus                   src *, dst *, sport *, dport -
                else if(dstport == packettable[i].dport) {return 9;}
        }

        strcpy(packettable[tpointer].sadd,srcadd);
        strcpy(packettable[tpointer].dadd,dstadd);
        packettable[tpointer].sport = srcport;
        packettable[tpointer].dport = dstport;
        tpointer++;
        return 0;
}

void cleardata(int n) {
        int i;
        for(i = 0; i < n; i++) {
                strcpy(packettable[i].sadd,"0.0.0.0");
                strcpy(packettable[i].dadd,"0.0.0.0");
                packettable[i].sport = 0;
                packettable[i].dport = 0;
        }
        tpointer = 0;
}
```

โปรแกรมแยกส่วนประกอบความถี่ด้วยเวฟเล็ต

```
#include <stdio.h>
#include <math.h>
#include <string.h>
#include <gsl/gsl_sort.h>
#include <gsl/gsl_wavelet.h>

int main(int argc, char *argv[]) {
	//Check argument.
	if(argc < 4) {
		printf("usage: %s <file> <length> <level>\n", argv[0]);
		return -1;
	}//End check argument.
	//Check level.
	int n = atoi(argv[2]);
	int level = atoi(argv[3]);
	if(level == 0) {
		printf("Level 0 : Can't decomposition.\n");
		return -1;
	} else if(level > (log(n)/log(2))) {
		printf("Max level is %g\n", log(n)/log(2));
		return -1;
	}//End check level.

	double *data = malloc(n * sizeof(double));
	double *cA = malloc(n * sizeof(double));
	double *cD = malloc(n * sizeof(double));
	double *A = malloc(n * sizeof(double));
	double *D = malloc(n * sizeof(double));

	//Initial data to zero.
	int i;
	for(i = 0; i < n; i++)
		data[i] = 0;
	char line[80];
	char *sub_string;
	//Read input file to data[i].
	FILE *f;
	int isIndex = 1;
	if((f = fopen(argv[1], "r")) != NULL) {
		while(fgets(line, 80, f) != NULL) {
			strtok(line, ",");
			while((sub_string = strtok(NULL, "\t\n")) != NULL) {
				if(isIndex == 1) {
					i = atoi(sub_string);
					isIndex = 0;
				} else {
					data[i] = atof(sub_string);
					isIndex = 1;
					//printf("data[%d]:%.10g\n", i, data[i]);
				}
			}
		}
```

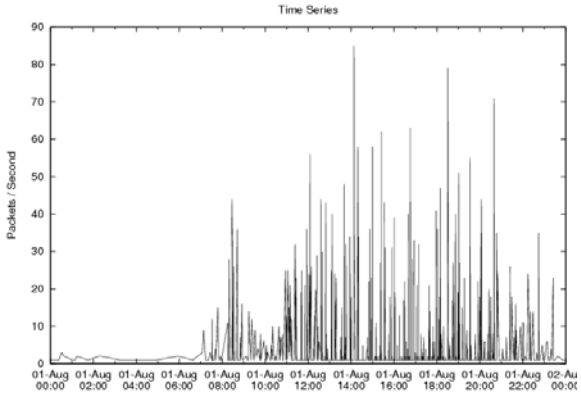
```
}
fclose(f);

//Initial wavelet variable.
int nd = n;
gsl_wavelet *w;
gsl_wavelet_workspace *work;
//w = gsl_wavelet_alloc(gsl_wavelet_daubechies, 4);
w = gsl_wavelet_alloc(gsl_wavelet_haar, 2);
work = gsl_wavelet_workspace_alloc(n);

int j;
//Start loop level.
for(j = 0; j < level; j++) {

	gsl_wavelet_transform_forward (w, data, 1, nd, work);

	for(i = 0; i < nd; i++) {
		if(i < nd/2) cA[i] = data[i];//for cA[i]
		else cA[i] = 0;
		if(i >= nd/2) cD[i] = data[i];//for cD[i]
		else cD[i] = 0;
	}

	//gsl_wavelet_transform_inverse(w, data, 1, nd, work);
	gsl_wavelet_transform_inverse(w, cA, 1, nd, work);
	gsl_wavelet_transform_inverse(w, cD, 1, nd, work);

	for(i = 0; i < nd; i++) {
		if(i < nd/2) data[i] = cA[2*i];
		else data[i] = 0;
	}
	nd = nd/2;
}//End loop level.

int step = pow(2,level-1);
for(i = 0; i < n/step; i++) {
	for(j = 0; j <= pow(2,level-1); j++) {
		A[(i*step)+j] = cA[i];
		D[(i*step)+j] = cD[i];
	}
}

//Print output data with timedata.
char *time;
if((f = fopen(argv[1], "r")) != NULL) {
	while(fgets(line, 80, f) != NULL) {
		time = strtok(line, ",");
		while((sub_string = strtok(NULL, "\t\n")) != NULL) {
			if(isIndex == 1) {
				i = atoi(sub_string);
				isIndex = 0;
			} else {
				printf("%s, %d\t%.10g %.10g\n", time, i, A[i], D[i]);
				isIndex = 1;
			}
		}
	}
```

```
        }
        }
        fclose(f);
        return 0;
}
```

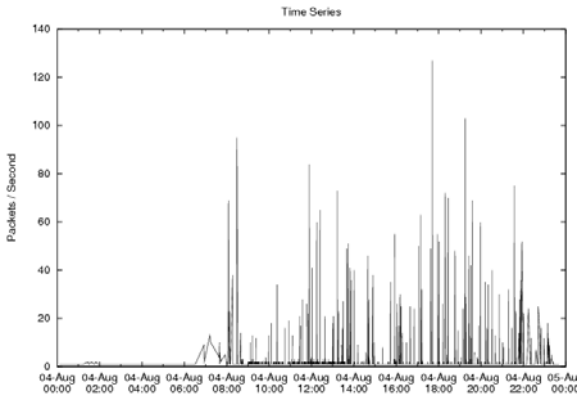
ภาคผนวก ค

ข้อมูลอนุกรมเวลาแยกตามโปรโตคอลประจำเดือนสิงหาคม 2551

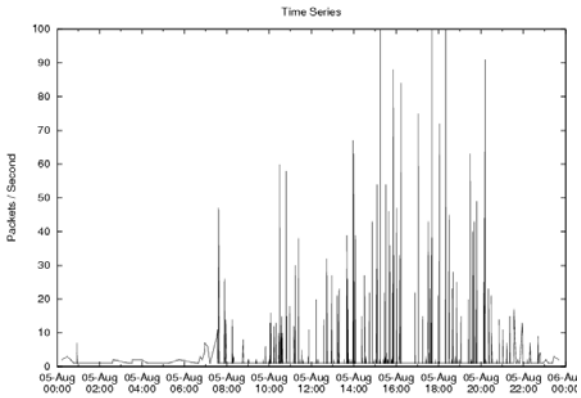
Internet Message Control Protocol (ICMP)



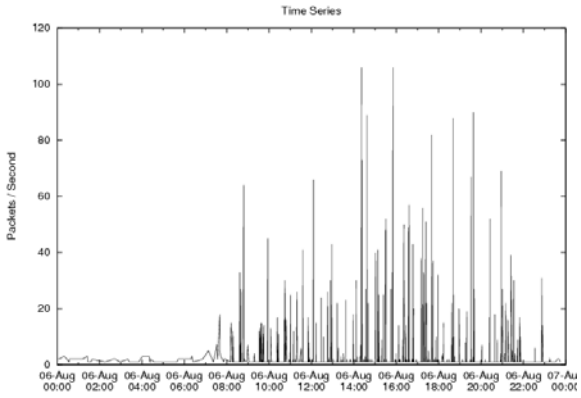
ภาพผนวกที่ ค1 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 1 สิงหาคม 2551



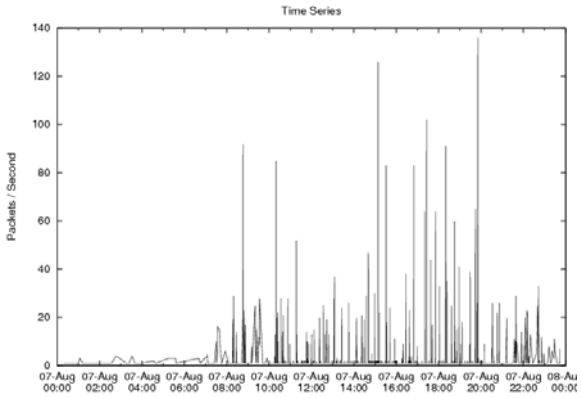
ภาพผนวกที่ ค2 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 4 สิงหาคม 2551



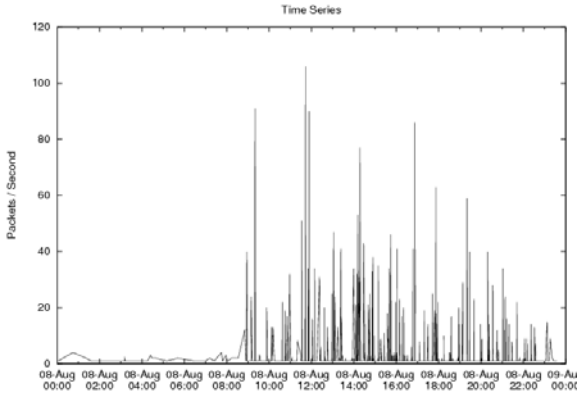
ภาพผนวกที่ ค3 จราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 5 สิงหาคม 2551



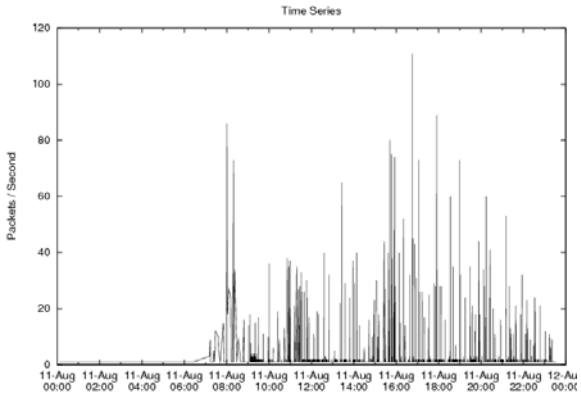
ภาพผนวกที่ ค4 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 6 สิงหาคม 2551



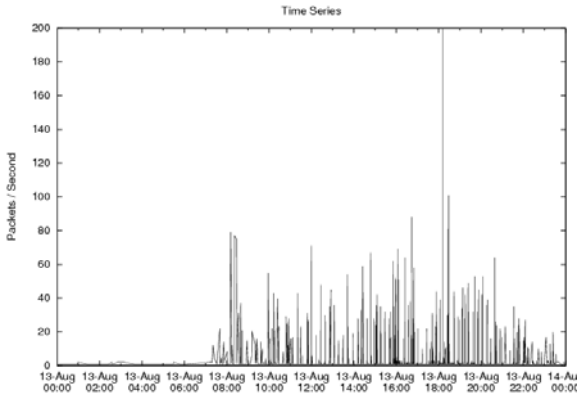
ภาพผนวกที่ ค5 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 7 สิงหาคม 2551



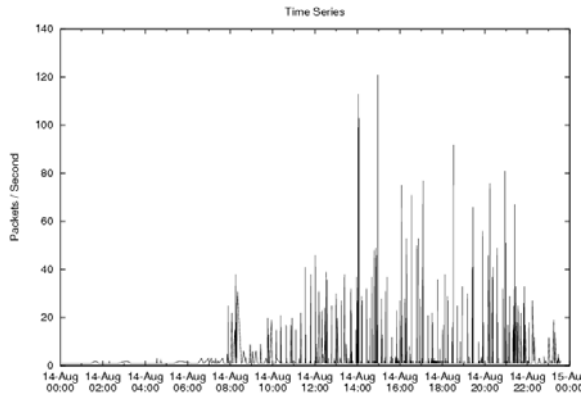
ภาพผนวกที่ ค6 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 8 สิงหาคม 2551



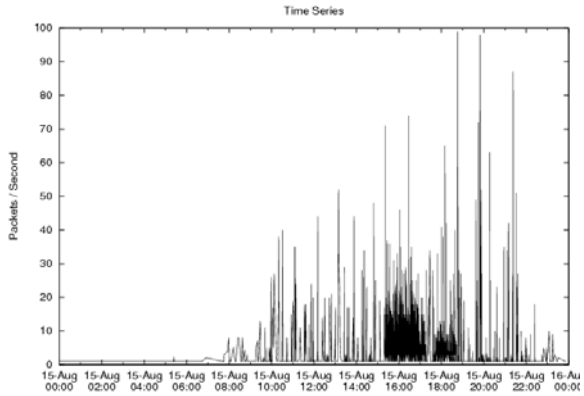
ภาพผนวกที่ ค7 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 11 สิงหาคม 2551



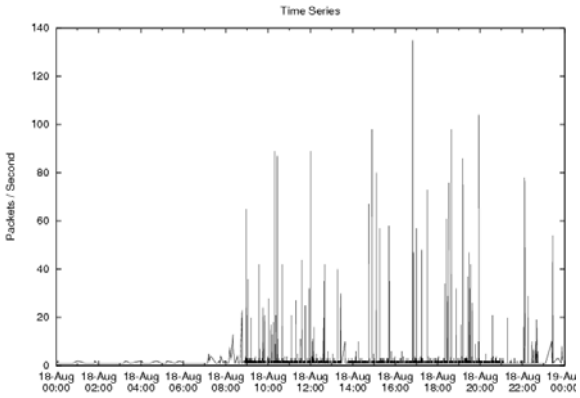
ภาพผนวกที่ ค8 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 13 สิงหาคม 2551



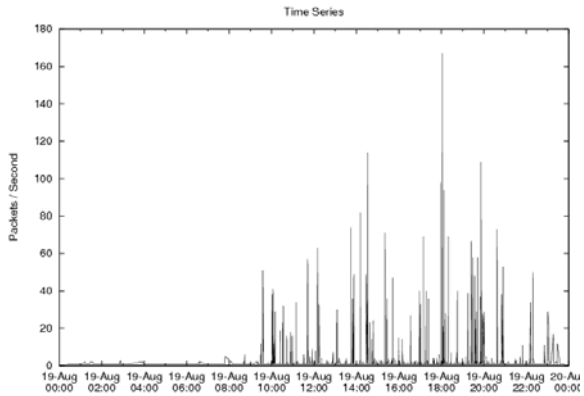
ภาพผนวกที่ ค9 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 14 สิงหาคม 2551



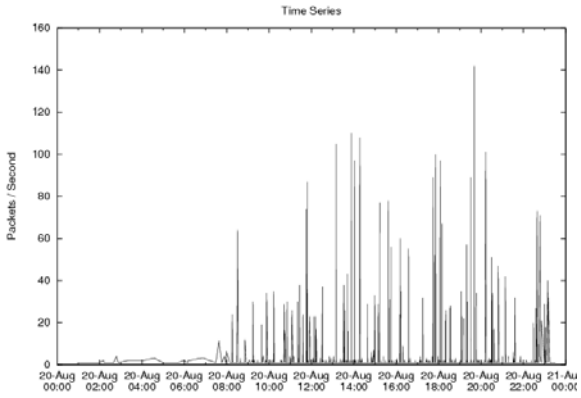
ภาพผนวกที่ ค10 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 15 สิงหาคม 2551



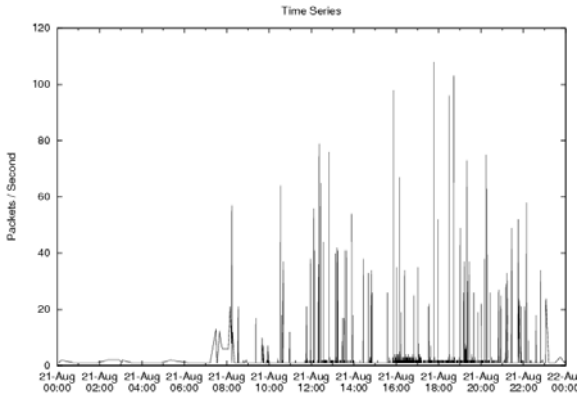
ภาพผนวกที่ ค11 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 18 สิงหาคม 2551



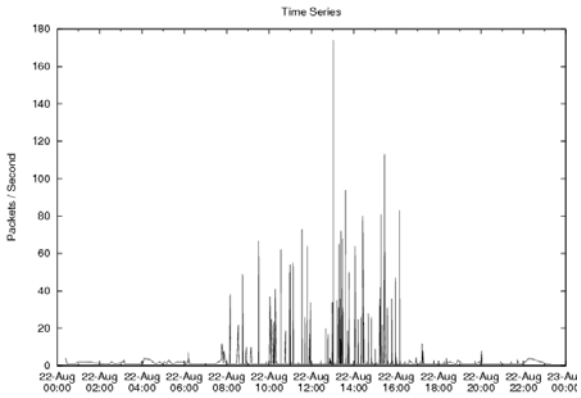
ภาพผนวกที่ ค12 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 19 สิงหาคม 2551



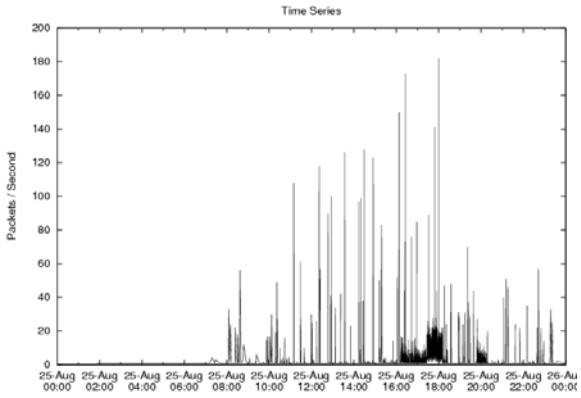
ภาพผนวกที่ ค13 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 20 สิงหาคม 2551



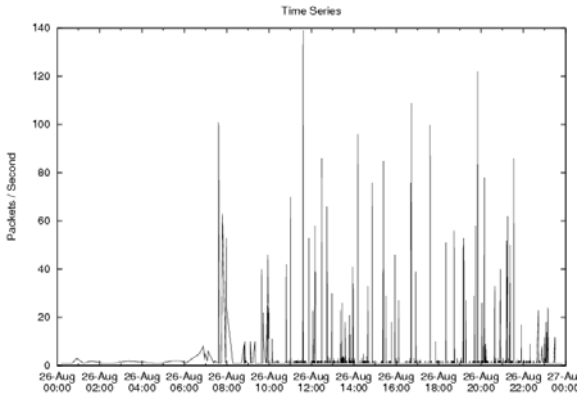
ภาพผนวกที่ ค14 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 21 สิงหาคม 2551



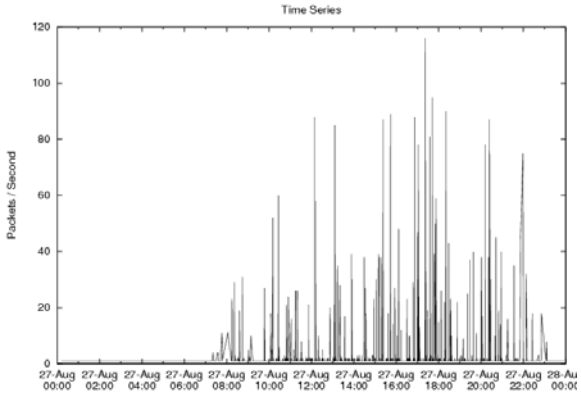
ภาพผนวกที่ ค15 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 22 สิงหาคม 2551



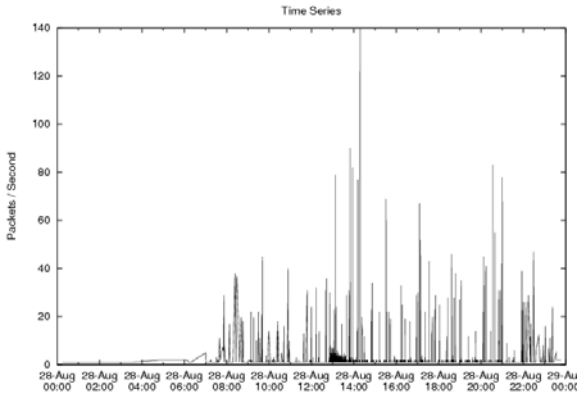
ภาพผนวกที่ ค16 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 25 สิงหาคม 2551



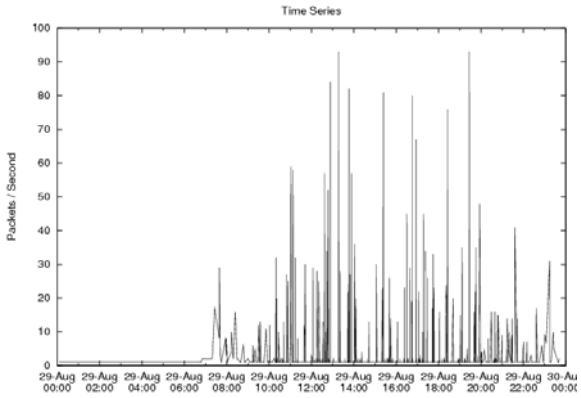
ภาพผนวกที่ ค17 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 26 สิงหาคม 2551



ภาพผนวกที่ ค18 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 27 สิงหาคม 2551

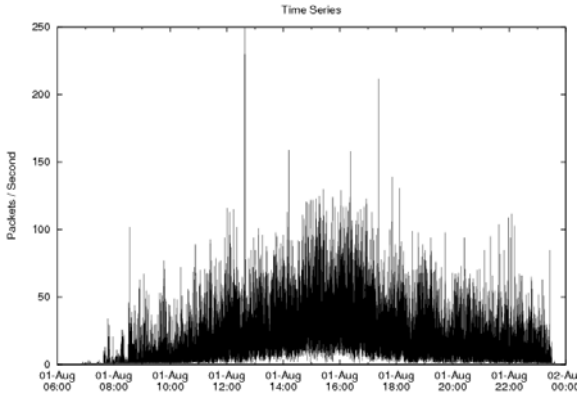


ภาพผนวกที่ ค19 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 28 สิงหาคม 2551

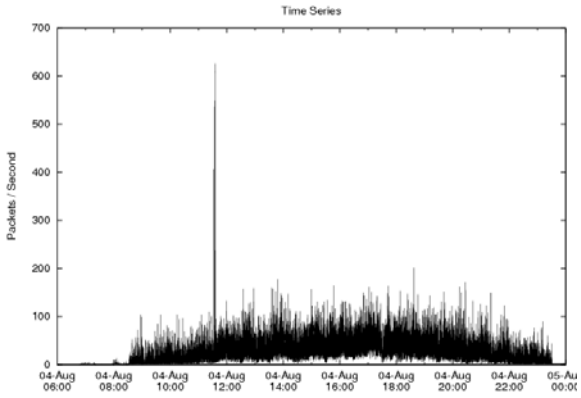


ภาพผนวกที่ ค20 ข้อมูลจราจรทางคอมพิวเตอร์ ICMP ขาออก วันที่ 29 สิงหาคม 2551

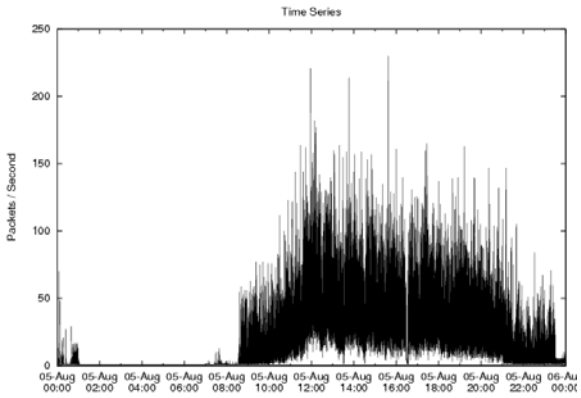
Transmission Control Protocol with SYN flag (TCP SYN)



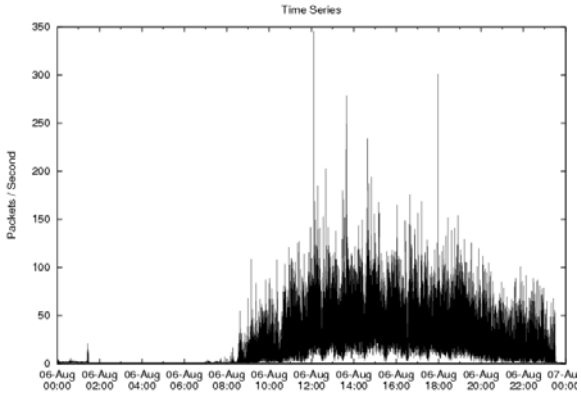
ภาพผนวกที่ ค21 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 1 สิงหาคม 2551



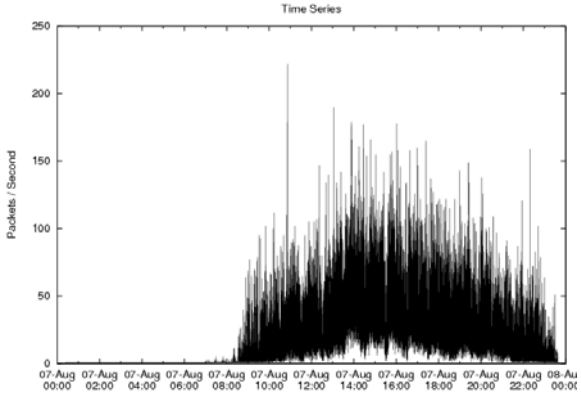
ภาพผนวกที่ ค22 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 4 สิงหาคม 2551



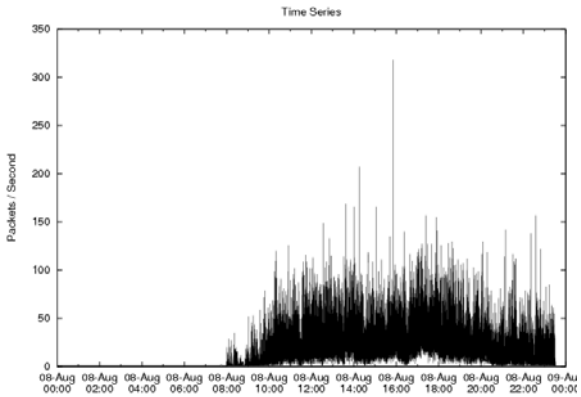
ภาพผนวกที่ ค23 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 5 สิงหาคม 2551



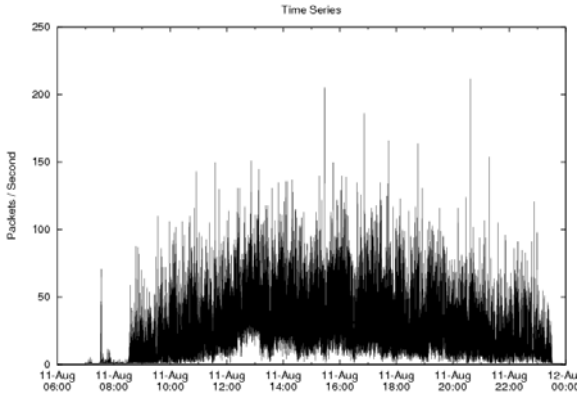
ภาพผนวกที่ ค24 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 6 สิงหาคม 2551



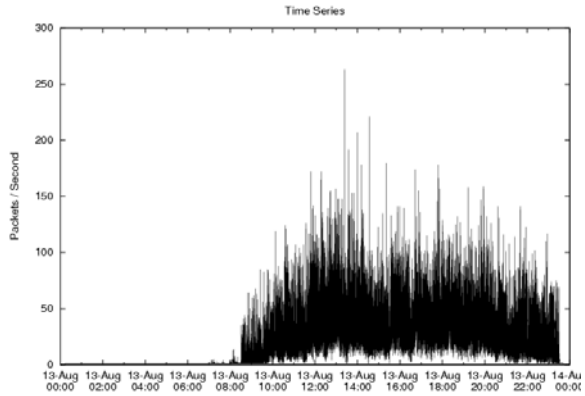
ภาพผนวกที่ ค25 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 7 สิงหาคม 2551



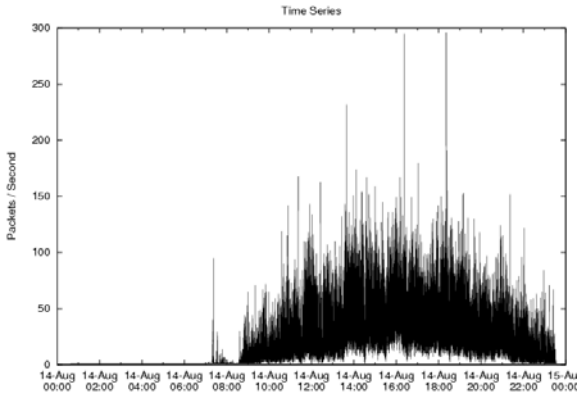
ภาพผนวกที่ ค26 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 8 สิงหาคม 2551



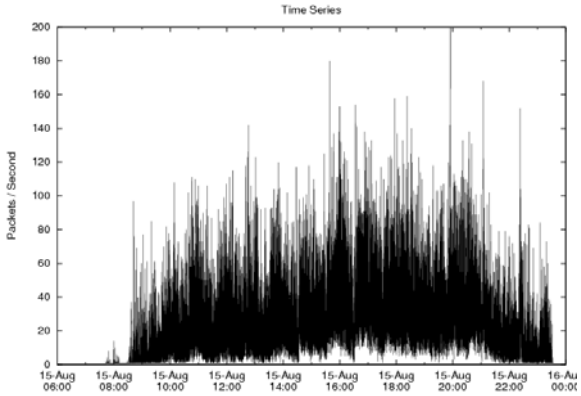
ภาพผนวกที่ ค27 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 11 สิงหาคม 2551



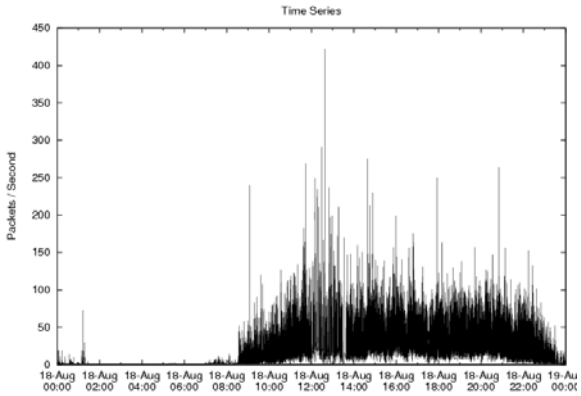
ภาพผนวกที่ ค28 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 13 สิงหาคม 2551



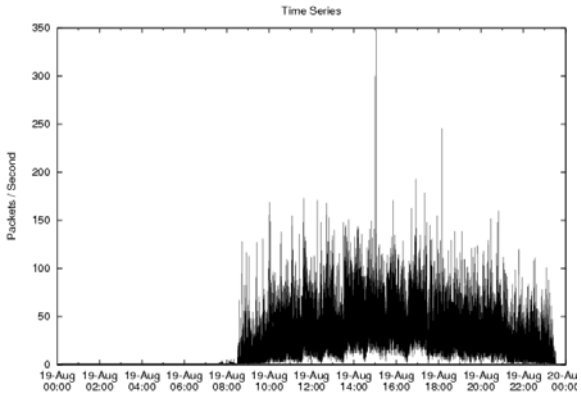
ภาพผนวกที่ ค29 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 14 สิงหาคม 2551



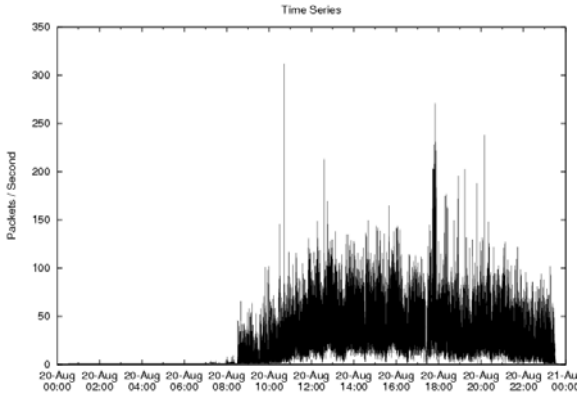
ภาพผนวกที่ ค30 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 15 สิงหาคม 2551



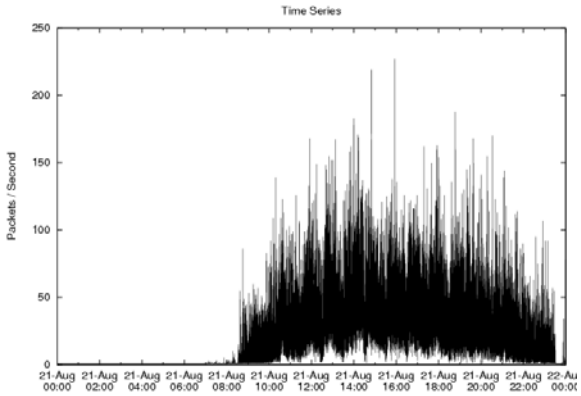
ภาพผนวกที่ ค31 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 18 สิงหาคม 2551



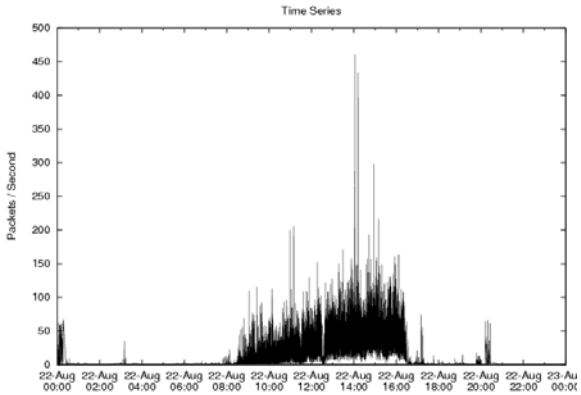
ภาพผนวกที่ ค32 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 19 สิงหาคม 2551



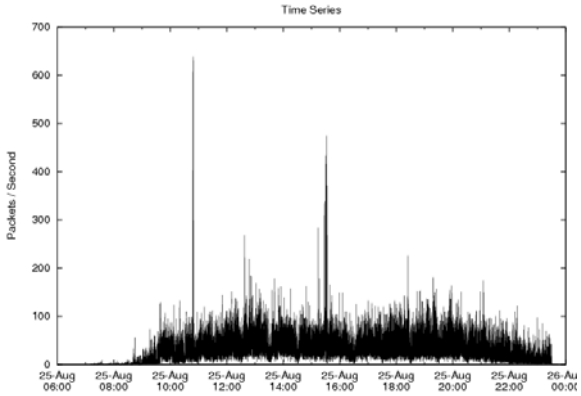
ภาพผนวกที่ ค33 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 20 สิงหาคม 2551



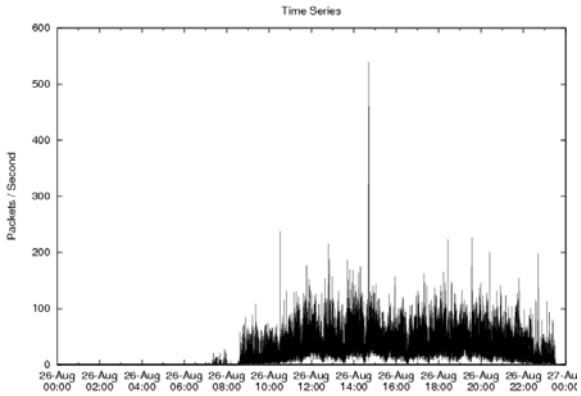
ภาพผนวกที่ ค34 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 21 สิงหาคม 2551



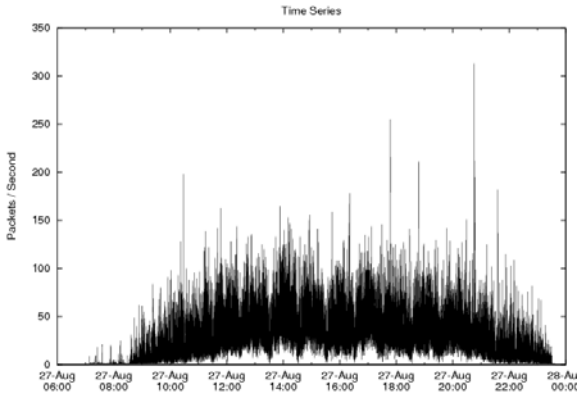
ภาพผนวกที่ ค35 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 22 สิงหาคม 2551



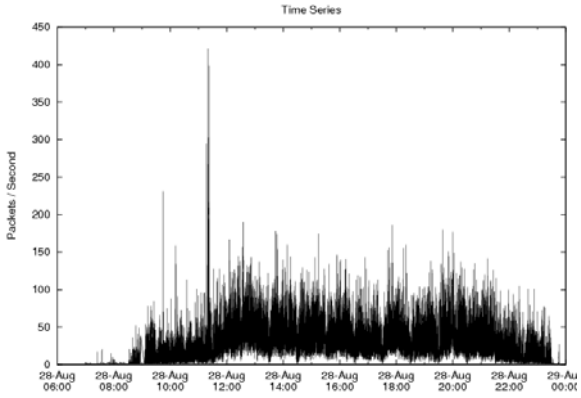
ภาพผนวกที่ ค36 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 25 สิงหาคม 2551



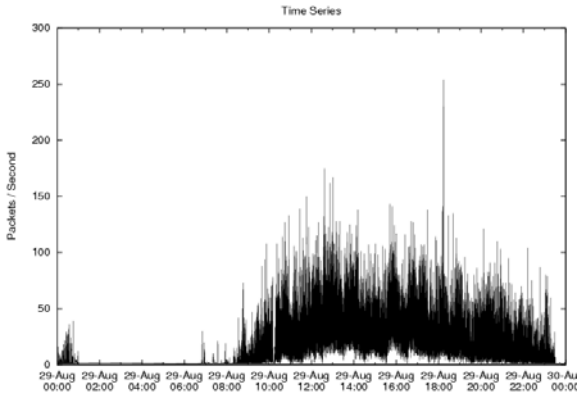
ภาพผนวกที่ ค37 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 26 สิงหาคม 2551



ภาพผนวกที่ ค38 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 27 สิงหาคม 2551

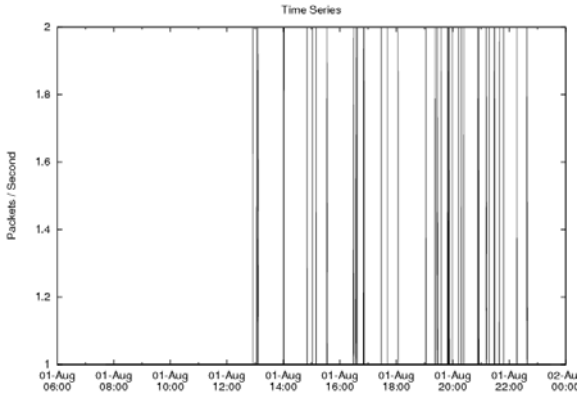


ภาพผนวกที่ ค39 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 28 สิงหาคม 2551

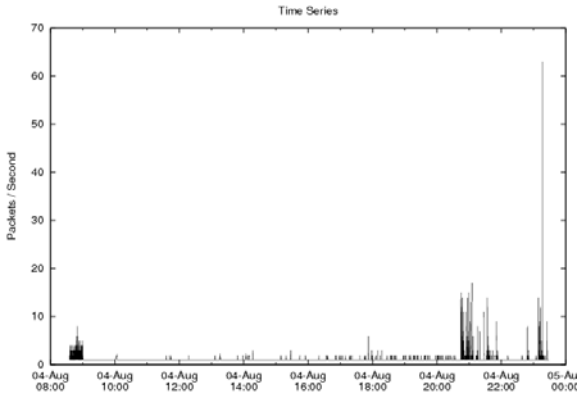


ภาพผนวกที่ ค40 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN ขาออก วันที่ 29 สิงหาคม 2551

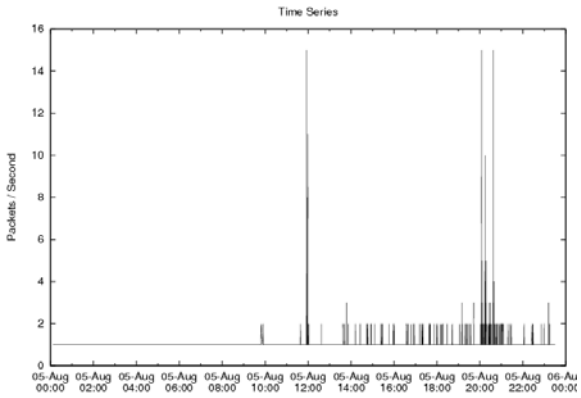
Transmission Control Protocol with SYN and ACK flag (TCP SYN/ACK)



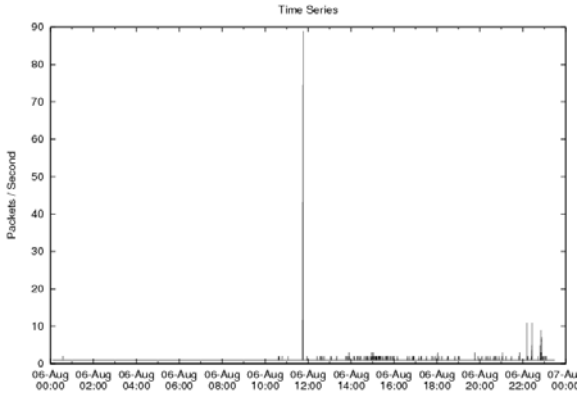
ภาพผนวกที่ ค41 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 1 สิงหาคม 2551



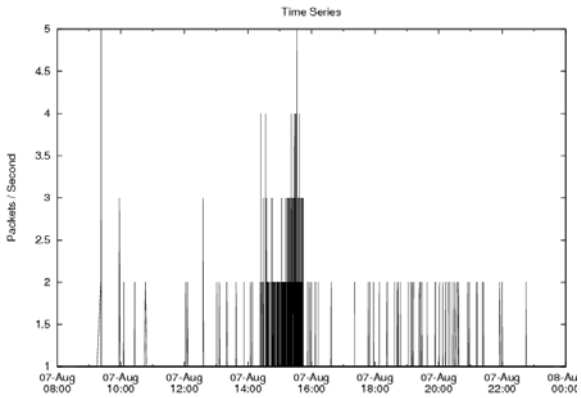
ภาพผนวกที่ ค42 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 4 สิงหาคม 2551



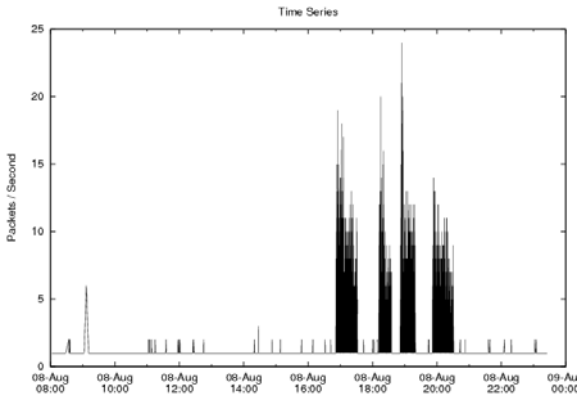
ภาพผนวกที่ ค43 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 5 สิงหาคม 2551



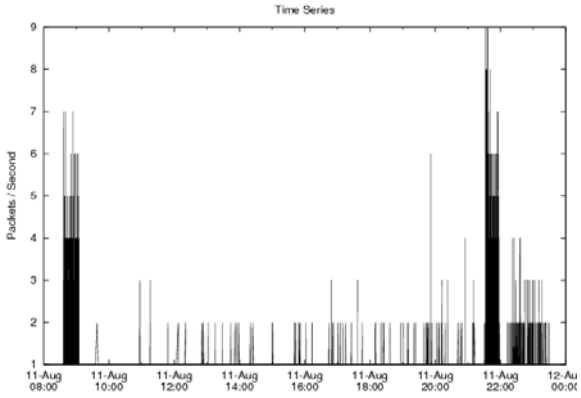
ภาพผนวกที่ ค44 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 6 สิงหาคม 2551



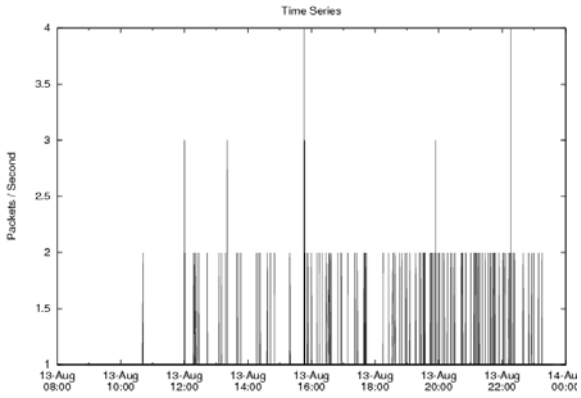
ภาพผนวกที่ ค45 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 7 สิงหาคม 2551



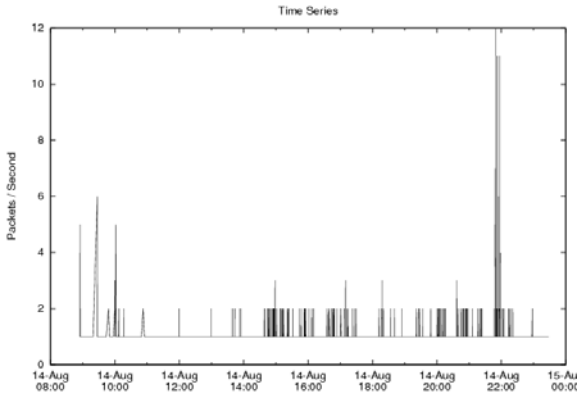
ภาพผนวกที่ ค46 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 8 สิงหาคม 2551



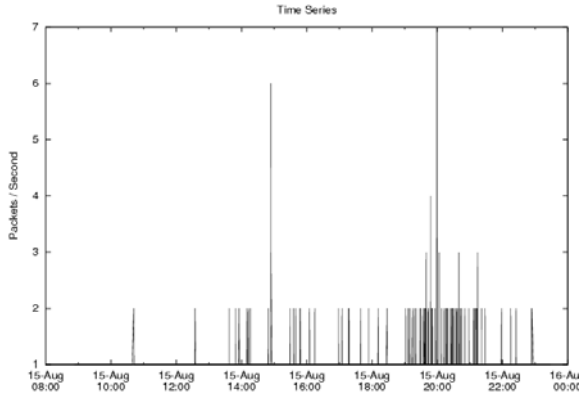
ภาพผนวกที่ ค47 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 11 สิงหาคม 2551



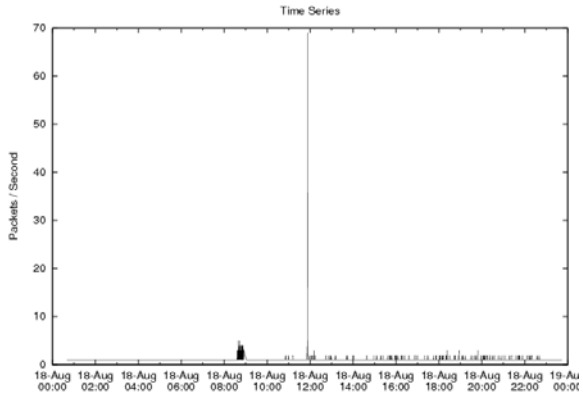
ภาพผนวกที่ ค48 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 13 สิงหาคม 2551



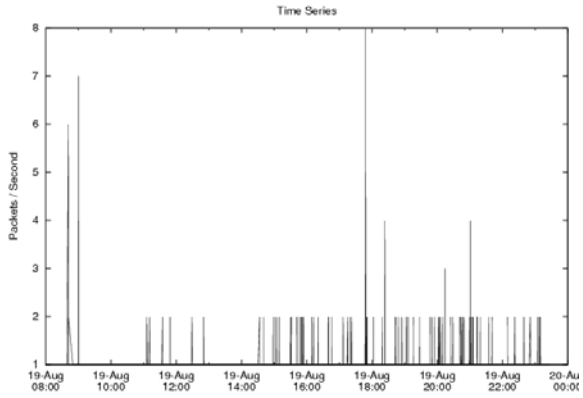
ภาพผนวกที่ ค49 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 14 สิงหาคม 2551



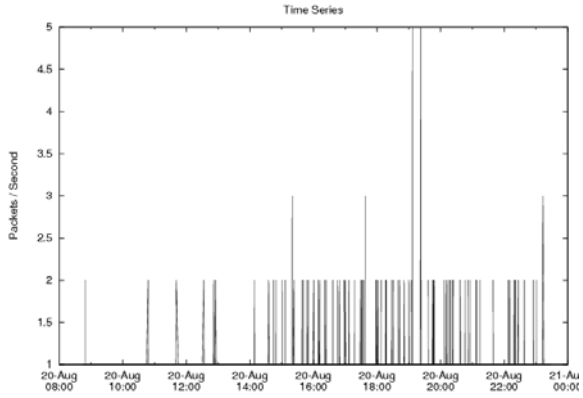
ภาพผนวกที่ ค50 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 15 สิงหาคม 2551



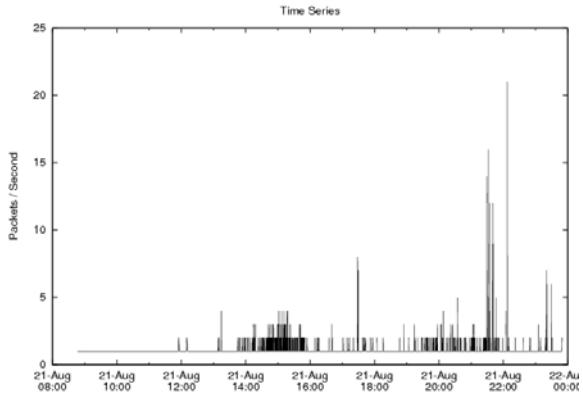
ภาพผนวกที่ ค51 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 18 สิงหาคม 2551



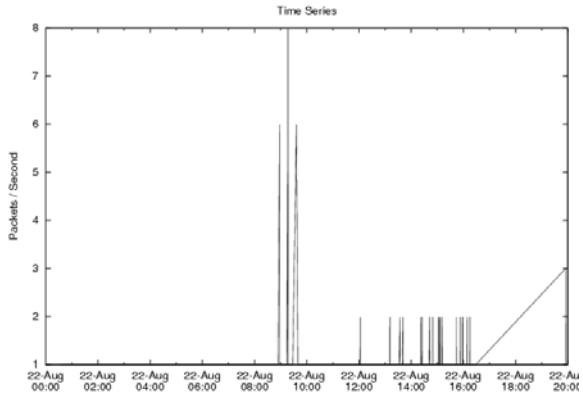
ภาพผนวกที่ ค52 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 19 สิงหาคม 2551



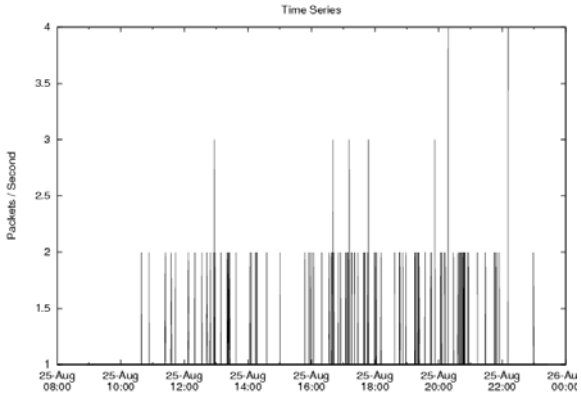
ภาพผนวกที่ ค53 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 20 สิงหาคม 2551



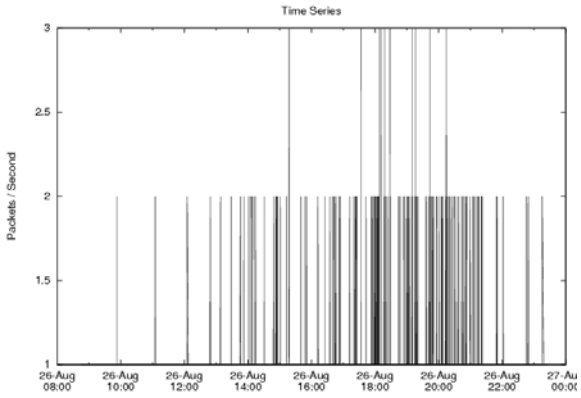
ภาพผนวกที่ ค54 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 21 สิงหาคม 2551



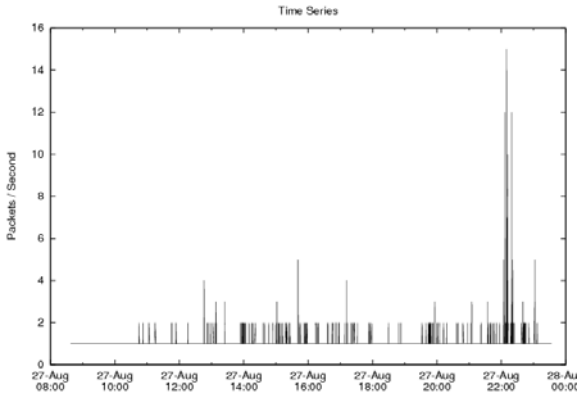
ภาพผนวกที่ ค55 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 22 สิงหาคม 2551



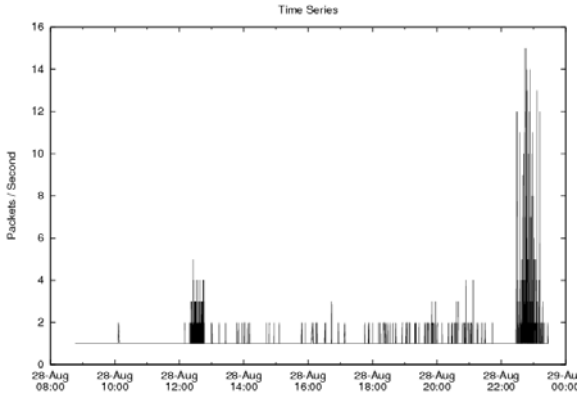
ภาพผนวกที่ ค56 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 25 สิงหาคม 2551



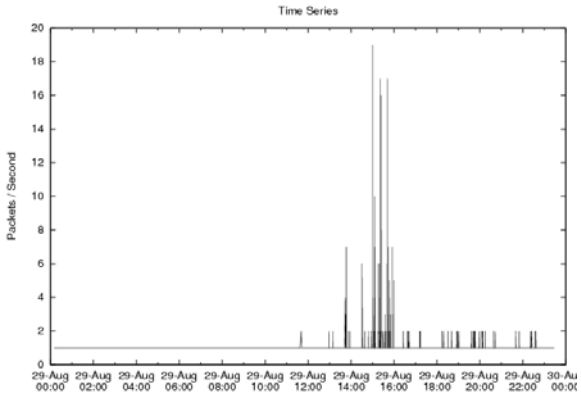
ภาพผนวกที่ ค57 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 26 สิงหาคม 2551



ภาพผนวกที่ ค58 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 27 สิงหาคม 2551

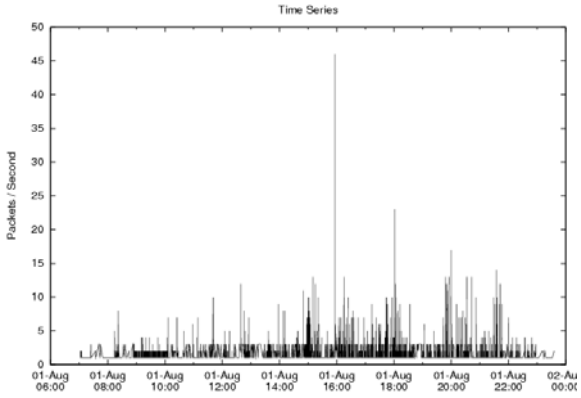


ภาพผนวกที่ ค59 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 28 สิงหาคม 2551

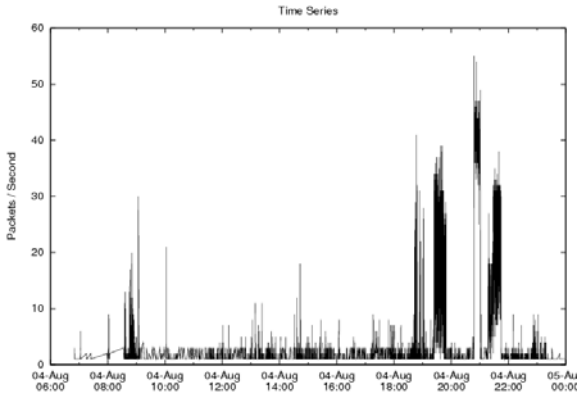


ภาพผนวกที่ ค60 ข้อมูลจราจรทางคอมพิวเตอร์ TCP SYN/ACK ขาออก วันที่ 29 สิงหาคม 2551

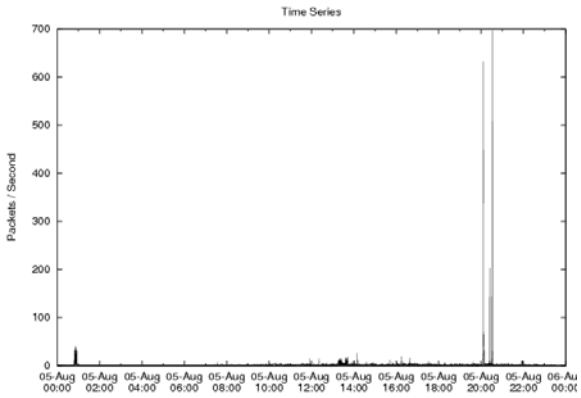
User Datagram Protocol (UDP)



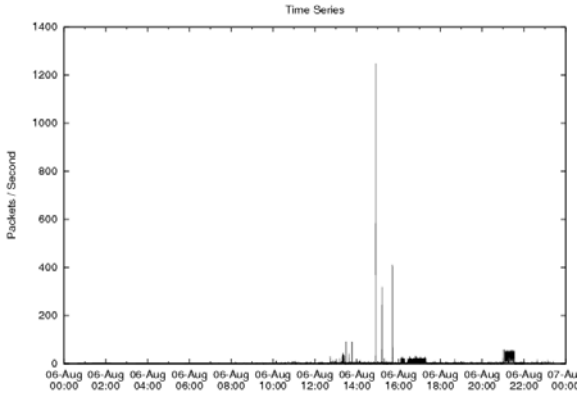
ภาพผนวกที่ ค61 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 1 สิงหาคม 2551



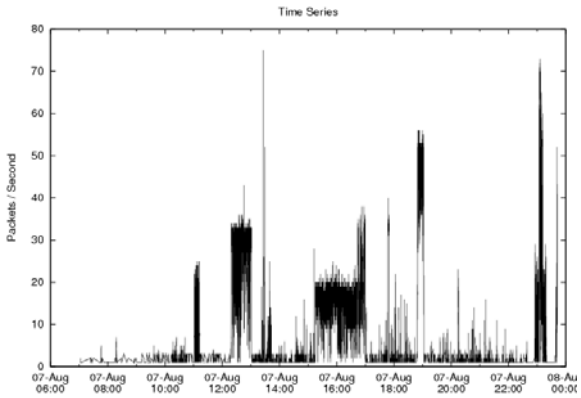
ภาพผนวกที่ ค62 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 4 สิงหาคม 2551



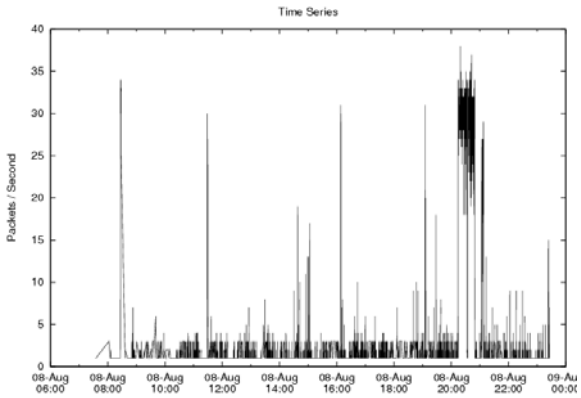
ภาพผนวกที่ ค63 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 5 สิงหาคม 2551



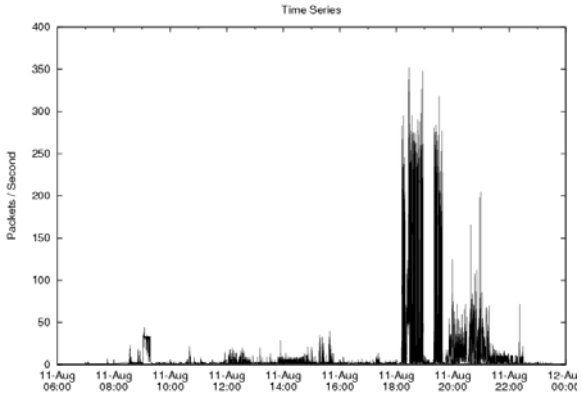
ภาพผนวกที่ ค64 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 6 สิงหาคม 2551



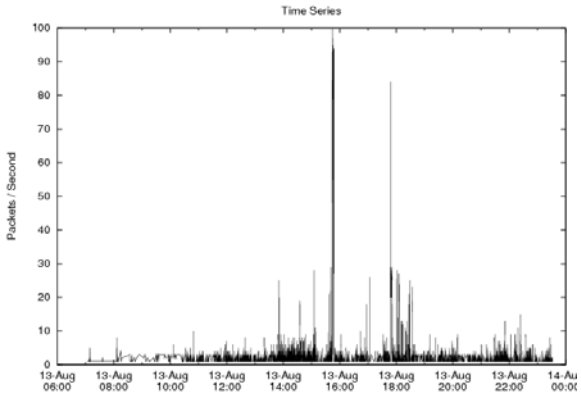
ภาพผนวกที่ ค65 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 7 สิงหาคม 2551



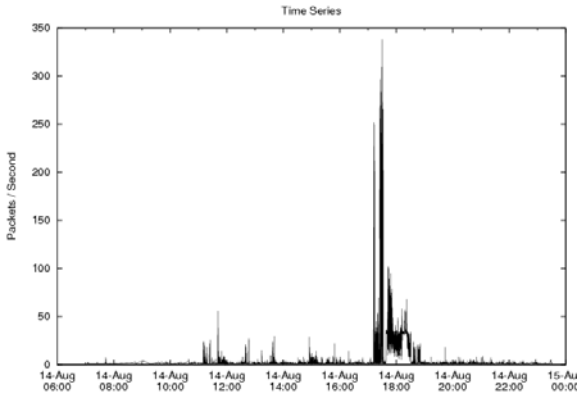
ภาพผนวกที่ ค66 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 8 สิงหาคม 2551



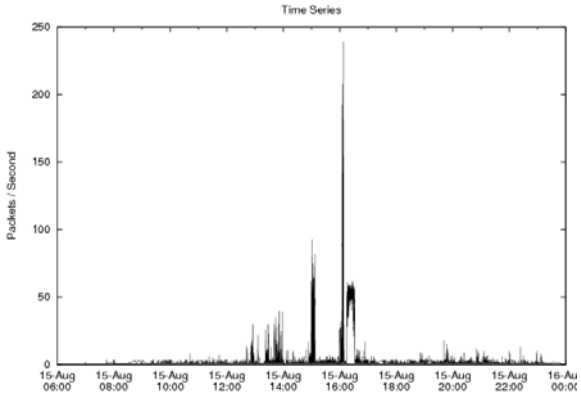
ภาพผนวกที่ ค67 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 11 สิงหาคม 2551



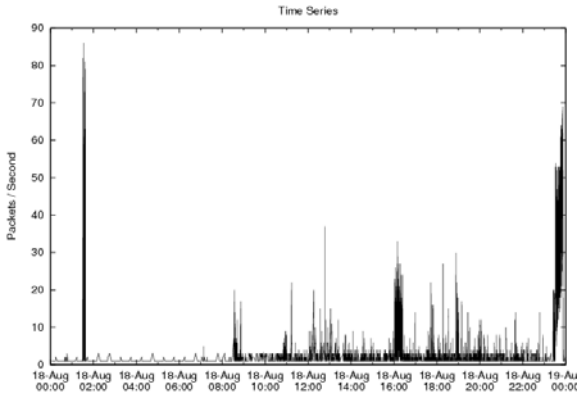
ภาพผนวกที่ ค68 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 13 สิงหาคม 2551



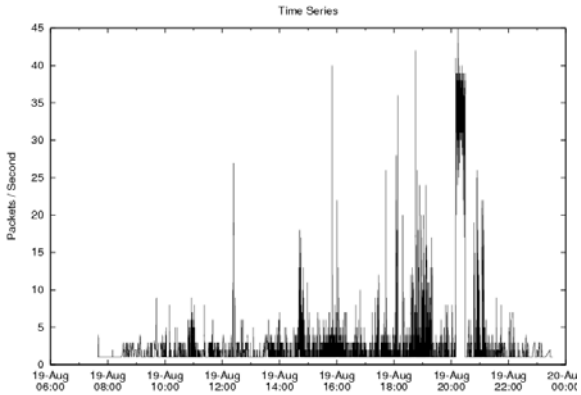
ภาพผนวกที่ ค69 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 14 สิงหาคม 2551



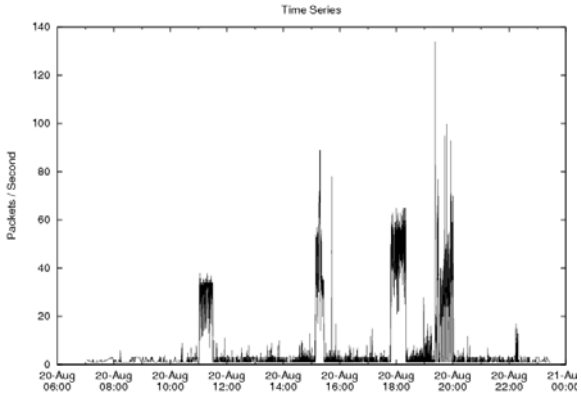
ภาพผนวกที่ ค70 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 15 สิงหาคม 2551



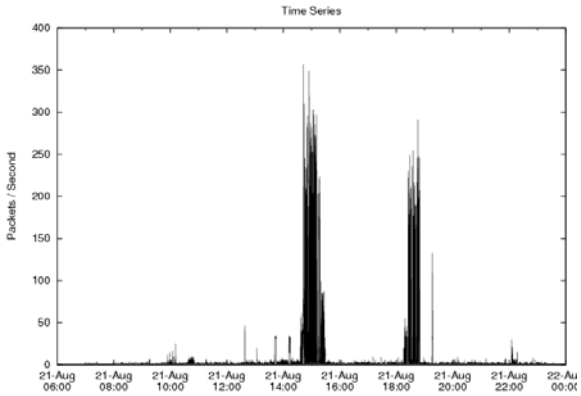
ภาพผนวกที่ ค71 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 18 สิงหาคม 2551



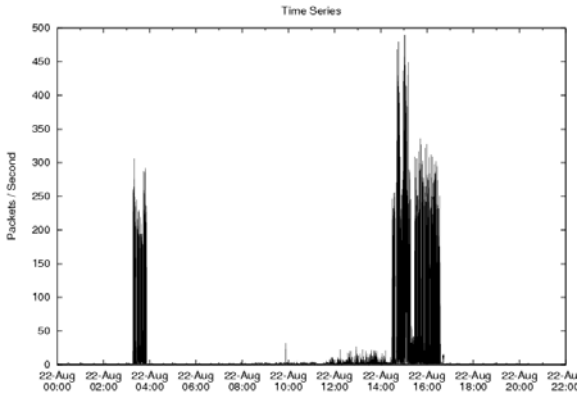
ภาพผนวกที่ ค72 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 19 สิงหาคม 2551



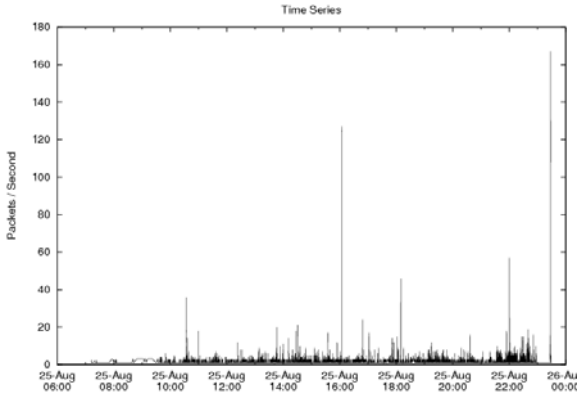
ภาพผนวกที่ ค73 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 20 สิงหาคม 2551



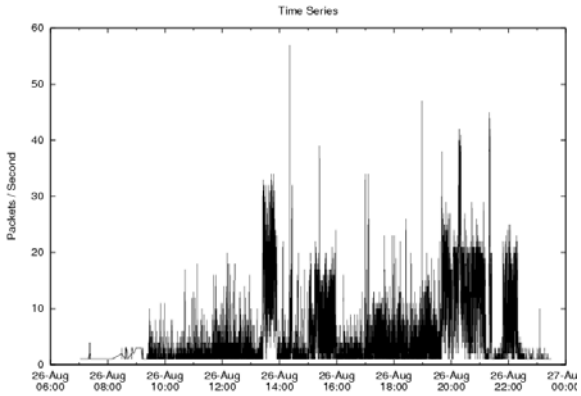
ภาพผนวกที่ ค74 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 21 สิงหาคม 2551



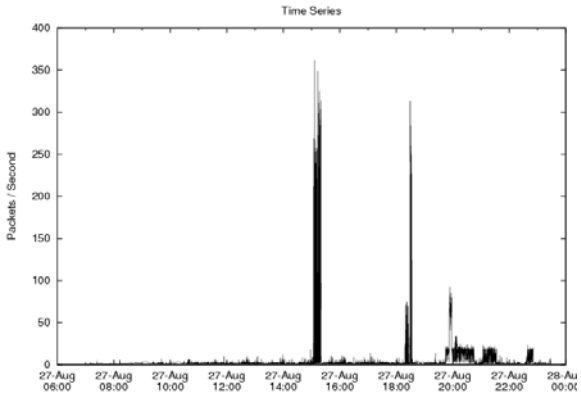
ภาพผนวกที่ ค75 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 22 สิงหาคม 2551



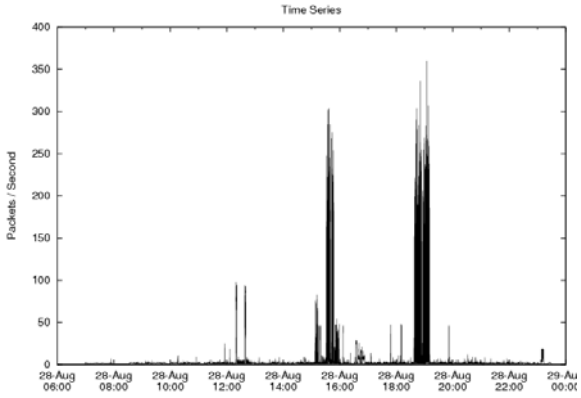
ภาพผนวกที่ ค76 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 25 สิงหาคม 2551



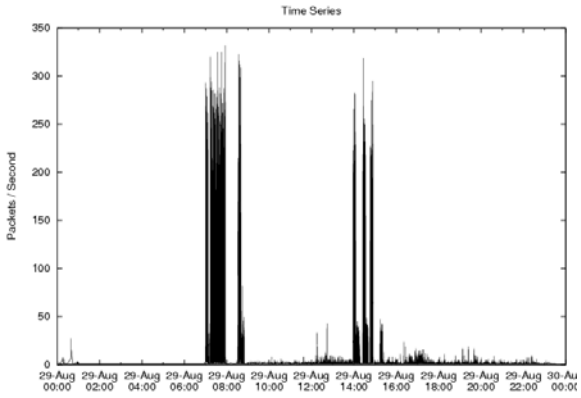
ภาพผนวกที่ ค77 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 26 สิงหาคม 2551



ภาพผนวกที่ ค78 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 27 สิงหาคม 2551



ภาพผนวกที่ ค79 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 28 สิงหาคม 2551



ภาพผนวกที่ ค80 ข้อมูลจราจรทางคอมพิวเตอร์ UDP ขาออก วันที่ 29 สิงหาคม 2551

ประวัติการศึกษา และการทำงาน

ชื่อ –นามสกุล	นายเกรียงไกร ลิ่มทอง
วัน เดือน ปี ที่เกิด	วันที่ 8 กรกฎาคม 2517
สถานที่เกิด	นครศรีธรรมราช
ประวัติการศึกษา	ระดับปริญญาตรี คณะสารสนเทศศาสตร์ สาขาวิศวกรรมคอมพิวเตอร์ เกียรตินิยมอันดับสอง มหาวิทยาลัยศรีปทุม
ตำแหน่งหน้าที่การงานปัจจุบัน	วิศวกร ระดับ 6
สถานที่ทำงานปัจจุบัน	ฝ่ายปฏิบัติการคอมพิวเตอร์และเครือข่าย บริษัท ไปรษณีย์ไทย จำกัด เลขที่ 111 ถ.แจ้งวัฒนะ หลักสี่ กรุงเทพฯ 10210
ผลงานดีเด่นและรางวัลทางวิชาการ	-
ทุนการศึกษาที่ได้รับ	ทุนสถาบันบัณฑิตวิทยาศาสตร์และเทคโนโลยีไทย (TGIST) สำนักงานพัฒนาวิทยาศาสตร์และเทคโนโลยี แห่งชาติ ปี 2551-2553 ทุนนักวิจัยแลกเปลี่ยน National Institute of Informatics (NII) Sokendai University กรุงโตเกียว ประเทศญี่ปุ่น ปี 2551