การสุ่มตัวอย่างแพ็กเก็ตสำหรับตรวจจับหนอนแบบกราดตรวจ

นายเลิศพงษ์ เลิศไพศาลวงศ์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2551

PACKET SAMPLING FOR SCANNING WORM DETECTION

Mr. Lerdpong Lerdpaisarnwong

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science Program in Computer Science

Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University

Academic Year 2008

หัวข้อวิทยานิพนธ์	การสุ่มตัวอย่างแพ็กเก็ตสำหรับตรวจจับหนอนแบบกราดตรวจ
โดย	นายเลิศพงษ์ เลิศไพศาลวงศ์
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	อาจารย์ ดร.ยรรยง เต็งอำนวย

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์

(รองศาสตราจารย์ ดร.บุญสม เลิศหิรัญวงศ์)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ

(อาจารย์ จารุมาตร ปิ่นทอง)

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

(อาจารย์ ดร.ยรรยง เต็งอำนวย)

..... กรรมการ

(อาจารย์ ดร. ณัฐวุฒิ หนูไพโรจน์)

..... กรรมการ

(อาจารย์ ธงชัย โรจน์กังสดาล)

เลิศพงษ์ เลิศไพศาลวงศ์ : การสุ่มตัวอย่างแพ็กเก็ตสำหรับตรวจจับหนอนแบบกราดตรวจ. (PACKET SAMPLING FOR SCANNING WORM DETECTION) อ. ที่ปรึกษาวิทยานิพนธ์หลัก: อ.ดร.ยรรยง เต็งอำนวย, 80 หน้า.

ปัจจุบันข้อมูลบนเครือข่ายมีจำนวนมากขึ้นเรื่อยๆ ทำให้การเก็บข้อมูลสำหรับการวิเคราะห์ต้องใช้เนื้อที่มากและต้องมีการบริหารจัดการที่ดี วิธีการสุ่มตัวอย่างแพ็กเก็ตสามารถนำมาใช้กับสนิฟเฟอร์ ทำให้สามารถเก็บข้อมูลเป็นระยะเวลานานขึ้น แต่การสุ่มตัวอย่างแพ็กเก็ตทำให้ประสิทธิภาพในการตรวจจับหนอนลดน้อยลง เพราะอัตราการกราดตรวจของหนอนหลังจากถูกสุ่มตัวอย่างจะลดน้อยลงด้วย จึงต้องเลือกอัตราการสุ่มตัวอย่างให้เหมาะสม งานวิจัยนี้ได้เสนอวิธีการสุ่มตัวอย่างสำหรับลดข้อมูล พร้อมทั้งการหาขอบเขตขั้นต่ำของการสุ่มตัวอย่างและอัตราการกราดตรวจของหนอนที่สามารถตรวจถูกจับได้อย่างมีประสิทธิภาพ

ภาควิชา...วิศวกรรมคอมพิวเตอร์... ลายมือชื่อนิสิต...เลิศพงษ์...
สาขาวิชา...วิทยาศาสตร์คอมพิวเตอร์... ลายมือชื่อ อ. ที่ปรึกษาวิทยานิพนธ์หลัก...
ปีการศึกษา...2551...

4870444321 : MAJOR COMPUTER SCIENCE

KEYWORDS: SNIFFER / PACKET SAMPLING / INTRUSION DETECTION SYSTEM / SNORT / WORM

LERDPONG LERDPAISARNWONG: PACKET SAMPLING FOR SCANNING WORM DETECTION, ADVISOR: YUNYONG TENG-AMNUAY, Ph.D., 80 pp.

At present, data volume in the network is increasing dramatically. To keep traffic log for analysis, huge storage and extensive administration are needed. Packet sampling technique applied to sniffer is an interesting method for lengthening logging period. But packet sampling may cause some problems in worm detection performance, since some traffic log are lost and may not be adequate in capturing worm characteristics. Sampling rate needs to be chosen by considering worm scanning characteristic. This research proposes a packet sampling procedure for sniffer to increase duration of traffic logging, as well as establishing lower limit of sampling rate and minimum scanning rate for detecting scanning worm.

Department: Computer Engineering Student's Signature เลิศพงษ์

Field of Study: Computer Science Advisor's Signature

Academic Year: 2008

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สามารถเกิดขึ้นได้จากการอนุเคราะห์และขัดเกลาแนวคิดของอาจารย์ ดร.ยรรยง เต็งอำนวย ซึ่งเป็นอาจารย์ที่ปรึกษาวิทยานิพนธ์ จนทำให้มีแนวคิดที่ชัดเจนและสามารถทำได้จริง นอกจากท่านอาจารย์ที่ปรึกษาแล้วยังต้องขอขอบพระคุณศูนย์คอมพิวเตอร์ คณะวิศวกรรมศาสตร์ที่ได้อนุเคราะห์ให้ใช้สถานที่สำหรับติดตั้งเครื่องคอมพิวเตอร์สำหรับทดลอง และคุณ ชนินทร์ มหารักษ์ จากศูนย์คอมพิวเตอร์ที่ให้คำแนะนำและช่วยเหลือในการติดตั้งและแก้ปัญหาเครื่องคอมพิวเตอร์สำหรับทดลอง

สุดท้ายนี้อยากขอขอบคุณพี่ๆน้องๆห้องปฏิบัติการ ISEL ทุกคน อาจารย์ทุกท่าน เพื่อนๆที่อยู่ในรุ่นเดียวกัน และครอบครัวที่คอยให้กำลังใจเสมอตลอดเวลาที่ศึกษาอยู่ในจุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

หน้า

หน้า

หน้า

สารบัญตาราง

หน้า

หน้า

หน้า

สารบัญภาพ

หน้า

หน้า

หน้า

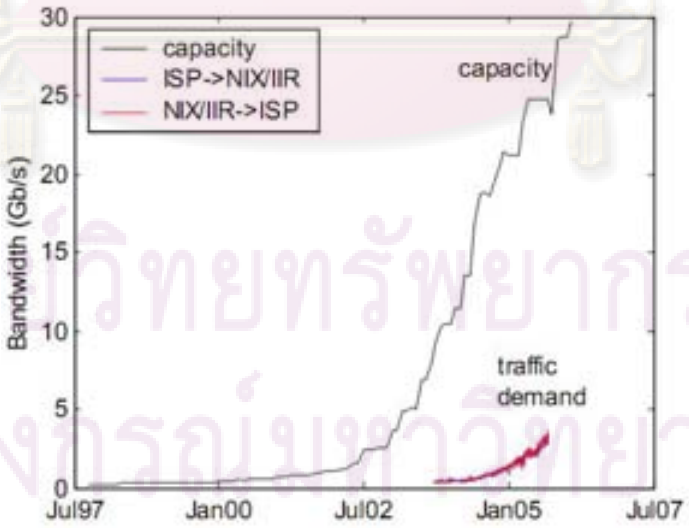
บทที่ 1

บทนำ

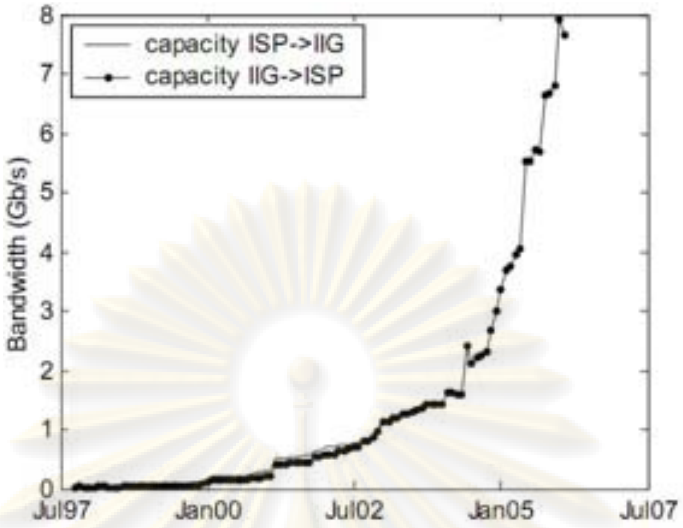
1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันข้อมูลจราจรมีจำนวนมากขึ้นเรื่อยๆ ทำให้การเก็บข้อมูลสำหรับการวิเคราะห์ต้องใช้เนื้อที่มากขึ้นและต้องมีการบริหารจัดการที่ดี วิธีการสุ่มตัวอย่างแพ็กเก็ตสามารถนำมาใช้กับโปรแกรมสนิฟเฟอร์ ทำให้เก็บข้อมูลเป็นระยะเวลานานขึ้น ในงานวิจัยของ Cabani [1] ได้กล่าวว่าขนาดเนื้อที่เก็บข้อมูลเพิ่มขึ้น 2 เท่าทุกๆ 12 เดือน ส่วนการใช้เครือข่ายเพิ่มขึ้น 2 เท่าทุกๆ 9 เดือน จะเห็นได้ว่าการใช้เครือข่ายมีแนวโน้มเพิ่มมากขึ้นกว่าเนื้อที่เก็บข้อมูล และในงานวิจัยของ Pongpaibool [2] ได้กล่าวถึงแนวโน้มข้อมูลจราจรตั้งแต่ปี 1992 ถึง 2007 ทั้งในประเทศและต่างประเทศซึ่งเพิ่มขึ้นอย่างต่อเนื่องดังแสดงในรูปที่ 1.1 และ 1.2

จากแนวโน้มนี้ทำให้เกิดผลกระทบต่อการเก็บข้อมูลเพื่อการวิเคราะห์ด้านต่างๆ เช่นทางนิติคอมพิวเตอร์ (Computer Forensics) หรือภัยทางเครือข่ายเช่น หนอน และการโจมตีแบบปฏิเสธการให้บริการ (Denial of Service) เพราะต้องเก็บข้อมูลจำนวนมากขึ้นทำให้เก็บข้อมูลได้ระยะเวลาน้อยลง วิธีการสุ่มตัวอย่างแพ็กเก็ตสามารถใช้กับสนิฟเฟอร์ ทำให้เก็บข้อมูลเป็นระยะเวลานานขึ้น แต่การสุ่มตัวอย่างแพ็กเก็ตทำให้ประสิทธิภาพในการตรวจจับหนอนลดน้อยลง จึงต้องศึกษาผลกระทบที่มีต่อการตรวจจับหนอนแบบกราดตรวจ



รูปที่ 1.1 ปริมาณข้อมูลจราจรภายในประเทศ



รูปที่ 1.2 ปริมาณข้อมูลจราจรที่ออกไปต่างประเทศ

1.2 วัตถุประสงค์ของการวิจัย

ในงานวิจัยนี้นี้มีวัตถุประสงค์เพื่อพัฒนาการสุ่มตัวอย่างสำหรับลดข้อมูลพร้อมทั้งศึกษาผลกระทบของการสุ่มตัวอย่างที่มีต่อการตรวจจับหนอนแบบกราดตรวจด้วยโปรแกรมสนอร์ต

1.3 ขอบเขตของการวิจัย

1. ใช้วิธีสุ่มตัวอย่างแบบมีชั้นภูมิและการสุ่มตัวอย่างการเชื่อมต่อ
2. ใช้เครื่องแม่ข่ายที่เป็น Intel based และใช้ระบบปฏิบัติการลินุกซ์
3. หนอนที่ตรวจจับเป็นหนอนแบบกราดตรวจ
4. ตรวจจับหนอนจากพฤติกรรมการกราดตรวจโดยใช้โปรแกรมสนอร์ต

1.4 ขั้นตอนของการวิจัย

1. ศึกษาทฤษฎีการสุ่มตัวอย่างจากงานวิจัยอื่นๆที่เกี่ยวข้องโดยเน้นไปที่การนำวิธีการสุ่มตัวอย่างไปใช้เพื่อลดการเก็บข้อมูลจราจรและตรวจจับการกราดตรวจของหนอน

2. ศึกษาผลกระทบของการสุ่มตัวอย่างที่มีต่อการตรวจจับการกราดตรวจด้วยโปรแกรมสนอร์ต

3. พัฒนาระบบทั้งการเก็บข้อมูลและการตรวจสอบการกราดตรวจของหนอนให้เป็นระบบเดียวกันเพื่อการใช้งานที่สะดวก

4. วิเคราะห์และสรุปผล พร้อมข้อเสนอแนะ

5. จัดทำรายงานวิทยานิพนธ์ฉบับสมบูรณ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ลดภาระค่าใช้จ่ายในการจัดหาเนื้อที่เก็บข้อมูลเพราะสามารถเก็บข้อมูลจราจรได้ระยะเวลามากขึ้นทำให้การวิเคราะห์ข้อมูลจราจรมีประสิทธิภาพมากขึ้น

2. สามารถนำโปรแกรมที่ถูกพัฒนาไปใช้ในองค์กรขนาดเล็กเพื่อสนับสนุนผู้บริหารระบบ เช่นองค์กรในต่างจังหวัด เพื่อลดภาระของผู้ดูแลระบบในการเก็บข้อมูลจราจรและตรวจจับหนอนแบบกราดตรวจ

3. มีค่าใช้จ่ายต่ำเพราะเป็นโปรแกรมแบบโอเพนซอร์ส (Open Source) และสามารถนำไปใช้งานกับคอมพิวเตอร์ส่วนบุคคลได้

4. ได้องค์ความรู้จากการศึกษาผลกระทบของการสุ่มตัวอย่างแพ็กเก็ตที่มีต่อการกราดตรวจของหนอนเพื่อผู้ที่สนใจสามารถนำไปพัฒนาต่อได้

1.6 ผลงานตีพิมพ์จากวิทยานิพนธ์

ส่วนหนึ่งของงานวิทยานิพนธ์นี้ได้รับการตีพิมพ์เป็นบทความวิชาการในหัวเรื่อง “การสุ่มตัวอย่างแพ็กเก็ตสำหรับตรวจจับหนอนแบบกราดตรวจ” โดยเลิศพงษ์ เลิศไพศาลวงศ์ และยรรยง เต็งอำนวย ในงานประชุมวิชาการ “The 12th National Computer Science and Engineering Conference (NCSEC 2008)” ซึ่งจัดขึ้น ณ โรงแรมลองบีช พัทยา ประเทศไทย ระหว่างวันที่ 20-21 พฤศจิกายน 2551 ดังภาคผนวก ค หน้า 71

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

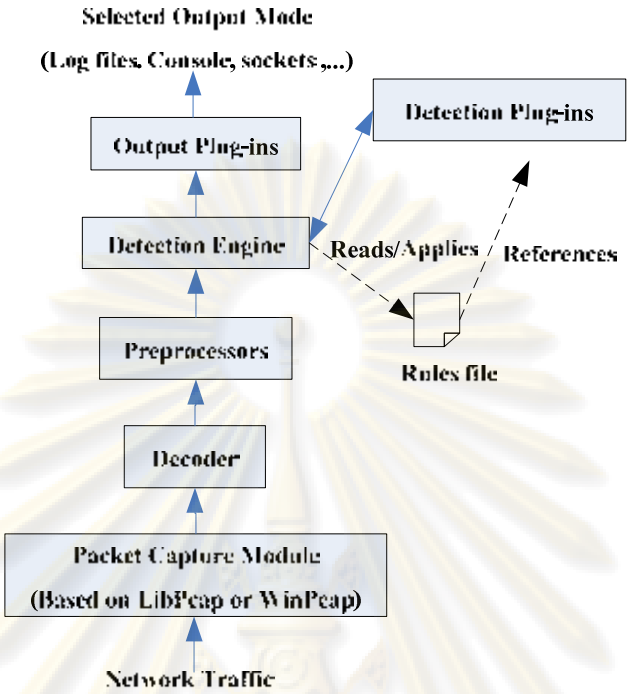
ในบทนี้กล่าวถึงทฤษฎีที่เกี่ยวข้องกับงานวิจัยนี้ โดยประกอบด้วยทฤษฎีหลัก ได้แก่ การสุ่มตัวอย่าง ซึ่งเป็นแนวคิดสำหรับการลดการเก็บแพ็กเก็ต การทำงานของสนอร์ต ใช้ในการตรวจจับการกราดตรวจของหนอน โปรแกรมเบส (BASE - Basic Analysis and Security Engine) สำหรับดูผลการตรวจจับผ่านหน้าเว็บ ลักษณะของหนอนแบบกราดตรวจ และกล่าวถึงงานวิจัยที่เกี่ยวข้อง

2.1 การสุ่มตัวอย่าง

การสุ่มตัวอย่างคือกระบวนการที่ใช้สุ่มจากประชากรที่ต้องการศึกษา โดยในงานวิจัยนี้ประชากรคือแพ็กเก็ตที่อยู่บนเครือข่าย การสุ่มตัวอย่างแบ่งเป็น 3 ประเภทหลัก [3] คือ การสุ่มตัวอย่างแบบมีชั้นภูมิ (stratified random sampling) การสุ่มตัวอย่างแบบมีระบบ (systematic sampling) และการสุ่มตัวอย่างแบบง่าย (simple random sampling) โดยการเลือกวิธีการสุ่มที่เหมาะสมจะกล่าวถึงในบทถัดไป

2.2 สนอร์ต

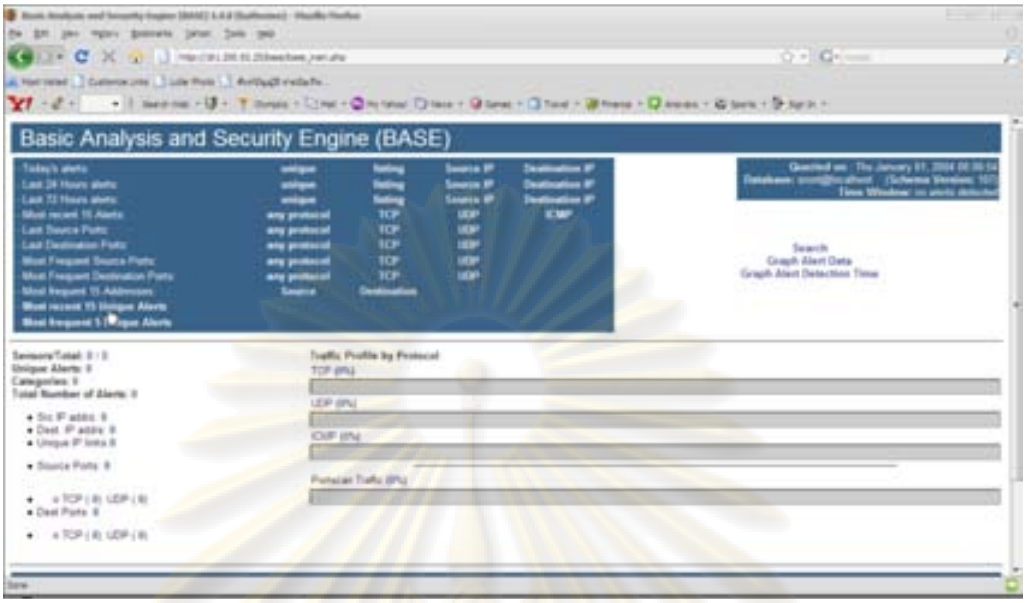
สนอร์ต (Snort) [4] เป็นโปรแกรมตรวจจับผู้บุกรุกทางเครือข่าย (network intrusion detection) โดยเป็นโปรแกรมโอเพนซอร์ส (open source) ซึ่งใช้กันโดยแพร่หลาย สนอร์ตจะทำงานโดยมีขั้นตอนซึ่งแสดงดังรูปที่ 2.1 [5] โปรแกรมสนอร์ตใช้คลังข้อมูล Libpcap (สำหรับลีนุกซ์) หรือ Winpcap (สำหรับวินโดวส์) ในการจับแพ็กเก็ตแล้วส่งให้ส่วน Decoder ถอดรหัสแพ็กเก็ตเป็นโปรโตคอลต่างๆ หลังจากนั้น พรีโพรเซสเซอร์จะตรวจจับพฤติกรรมผิดปกติต่างๆที่ได้กำหนดไว้ จากนั้นใช้ Detection Engine ในการตรวจสอบแพ็กเก็ตโดยใช้กฎ (Rule) ต่างๆ เมื่อเสร็จแล้วสนอร์ตจะแสดงผลลัพธ์ในรูปแบบต่างๆ ผ่านส่วน Output



รูปที่ 2.1 ขั้นตอนการทำงานของสนอร์ต

2.3 โปรแกรมเบส

โปรแกรมเบส (BASE - Basic Analysis and Security Engine) [6] เป็นหน้าเว็บสำหรับวิเคราะห์และแสดงผลการตรวจจับความผิดปกติต่างๆที่ได้จากสนอร์ต ผลการตรวจจับจะถูกเก็บเป็นรูปแบบฐานข้อมูลด้วยโปรแกรม mysql และเมื่อต้องการแสดงผล โปรแกรมเบสจะดึงผลจากฐานข้อมูลมาแสดงที่หน้าเว็บ โปรแกรมเบสเป็นโปรแกรมโอเพนซอร์สและใช้กันอย่างแพร่หลาย ตัวอย่างหน้าเว็บของโปรแกรมเบส แสดงได้ดังรูปที่ 2.2



รูปที่ 2.2 หน้าเว็บของโปรแกรมเบส

2.4 หนอนแบบกราดตรวจ

หนอนแบบกราดตรวจ (scanning worm) มีพฤติกรรมที่สำคัญคือ กราดตรวจหาโฮสต์เพื่อแพร่กระจายไปยังเครื่องที่มีช่องโหว่ให้มากที่สุด [7] โดยมีขั้นตอนการทำงาน 4 ขั้นตอน [8] คือ การค้นหาเป้าหมาย การตรวจสอบช่องโหว่ การส่งโปรแกรม และการเริ่มการทำงานของโปรแกรม ในงานวิจัยนี้จะตรวจจับหนอนในขั้นตอนค้นหาเป้าหมาย โดยในการค้นหาเป้าหมาย หนอนจะหาโฮสต์ที่ออนไลน์อยู่และมีช่องโหว่ โดยใช้วิธีการกราดตรวจแบบอัตโนมัติ ดังนั้นจึงเกิดแพ็กเก็ตการเชื่อมต่อ (connection) ไปยังหมายเลขไอพีต่างๆ เป็นจำนวนมาก

หนอนมีมากมายหลายชนิด แต่ละชนิดมีอัตราการกราดตรวจ (จำนวนการเชื่อมต่อที่ส่งไปยังโฮสต์เหยื่อ) แตกต่างกัน ในที่นี้จะยกตัวอย่างหนอนที่รู้จักโดยทั่วไปจากงานวิจัยต่างๆ [9-12] แสดงในตารางที่ 2.1 หนอนที่มีอัตราการกราดตรวจต่ำที่สุดคือหนอนบลาสเตอร์ มีอัตราการกราดตรวจ 10 แพ็กเก็ตต่อวินาที ซึ่งจะกำหนดเป็นอัตราการกราดตรวจที่ต่ำสุดในงานวิจัยนี้

ตารางที่ 2.1 อัตราการกราดตรวจของหนอนต่อโฮสต์

ชนิด	อัตราการกราดตรวจ (แพ็กเก็ตต่อวินาที)
Blaster	10-20
Welchia	70

ตารางที่ 2.1 (ต่อ) อัตราการกราดตรวจของหนอนต่อโฮสต์

ชนิด	อัตราการกราดตรวจ (แพ็กเก็ตต่อวินาที)
Nimda	200
Code Red	200-400
Slammer	2000

2.5 การเชื่อมต่อและโฟลว์

การเชื่อมต่อ (connection) [13] ในงานวิจัยนี้หมายถึง การติดต่อระหว่างโฮสต์หนึ่งไปยังโฮสต์หนึ่ง โดยแพ็กเก็ตที่มีการเชื่อมต่อเดียวกันจะมีหมายเลยไอพีต้นทางและหมายเลขไอพีปลายทางเหมือนกัน ส่วนโฟลว์ [14] หมายถึงการติดต่อกันระหว่างโฮสต์ในระดับชั้นขนส่ง (transport layer) แพ็กเก็ตที่มีโฟลว์เดียวกันจะมีหมายเลขไอพีต้นทาง หมายเลขไอพีปลายทาง พอร์ตต้นทาง พอร์ตปลายทาง และโพรโตคอลเหมือนกัน

2.6 งานวิจัยที่เกี่ยวข้อง

ผู้วิจัยได้ศึกษางานวิจัยของ Brauckhoff และคณะ [15] พบว่าเมื่อสุ่มตัวอย่างแพ็กเก็ตเพื่อหาความผิดปกติของเครือข่าย ตัววัดความผิดปกติอย่างเช่น จำนวนแพ็กเก็ตที่ได้จากการสุ่มสามารถประมาณค่าจำนวนแพ็กเก็ตเมื่อไม่ได้ถูกสุ่มได้อย่างแม่นยำ และค่าเอ็นโทรปี (คือตัววัดความหลากหลายของหมายเลขไอพีที่อยู่บนข้อมูลจราจร) ที่อัตราการสุ่มตัวอย่างต่างๆกลับไม่ลดลง งานวิจัยของ Kawahara และคณะ [16] ได้นำเทคนิคการสุ่มตัวอย่างมาใช้กับการตรวจหาความผิดปกติต่างๆ ของเครือข่ายเพื่อลดขนาดของข้อมูลที่จะเก็บ โดยพบว่า ความผิดปกติ เช่น การกราดตรวจเครือข่าย และ SYN Flood ก่อให้เกิดโฟลว์ (flow) ขนาดเล็กเป็นจำนวนมาก เมื่อสุ่มตัวอย่างแล้วโฟลว์เหล่านี้จะหายไปเนื่องจากมีโอกาสถูกเลือกน้อยเพราะมีเพียง 1-2 แพ็กเก็ตในโฟลว์เท่านั้น ทำให้ตรวจพบความผิดปกติได้ยาก ดังนั้นเขาจึงเสนอวิธีการแบ่งกลุ่มสำหรับการตรวจจับความผิดปกติโดยการจัดกลุ่มข้อมูล เช่น ถ้าต้องการตรวจจับการกราดตรวจ (scanning) ให้จัดกลุ่มข้อมูลตามหมายเลขไอพีต้นทาง ซึ่งวิธีการนี้ทำให้เพิ่มประสิทธิภาพในการตรวจจับความผิดปกติจากข้อมูลจราจรที่ถูกสุ่มได้ และงานวิจัยของ Sekar และคณะ [13] เสนอวิธีการตรวจจับหนอนที่มีอัตราการกราดตรวจต่ำ โดยกล่าวว่าพฤติกรรมของหนอนจะพยายามกราดตรวจหาโฮสต์ (host) เป้าหมายที่ไม่ซ้ำกันเสมอ แต่ในข้อมูลจราจรที่ปกติ โฮสต์จะติดต่อไปยัง

โฮสต์อื่นๆน้อย แล้วติดต่อซ้ำๆกัน เมื่อขยายช่วงเวลาในการตรวจจับการกราดตรวจของหนอนทำให้สามารถแยกแยะพฤติกรรมการกราดตรวจกับข้อมูลจราจรปกติได้อย่างถูกต้อง

ในงานนี้ผู้วิจัยใช้พรีโพรเซสเซอร์ Flow-portscan ของสนอร์ต ตรวจจับการกราดตรวจของหนอน ซึ่งมีวิธีการตรวจจับใกล้เคียงกับที่ Kawahara และคณะ เสนอไว้ และนำวิธีของ Sekar และคณะ เสนอมาประยุกต์ใช้เพื่อตรวจจับอัตราการกราดตรวจที่ลดลงเนื่องจากการสุ่มตัวอย่าง และศึกษาผลกระทบที่มีต่อการตรวจจับกราดตรวจของสนอร์ต

บทที่ 3

การดำเนินงานวิจัย

ในบทนี้จะกล่าวถึง สมมติฐาน การคำนวณ และทฤษฎีต่างๆที่จะนำไปทดลองในบทถัดไป อีกทั้งยังนำทฤษฎีต่างๆ มาประยุกต์ใช้การพัฒนาโปรแกรมสุ่มตัวอย่างสำหรับเพิ่มช่วงเวลาเก็บข้อมูล

3.1 การตรวจจับการกราดตรวจ

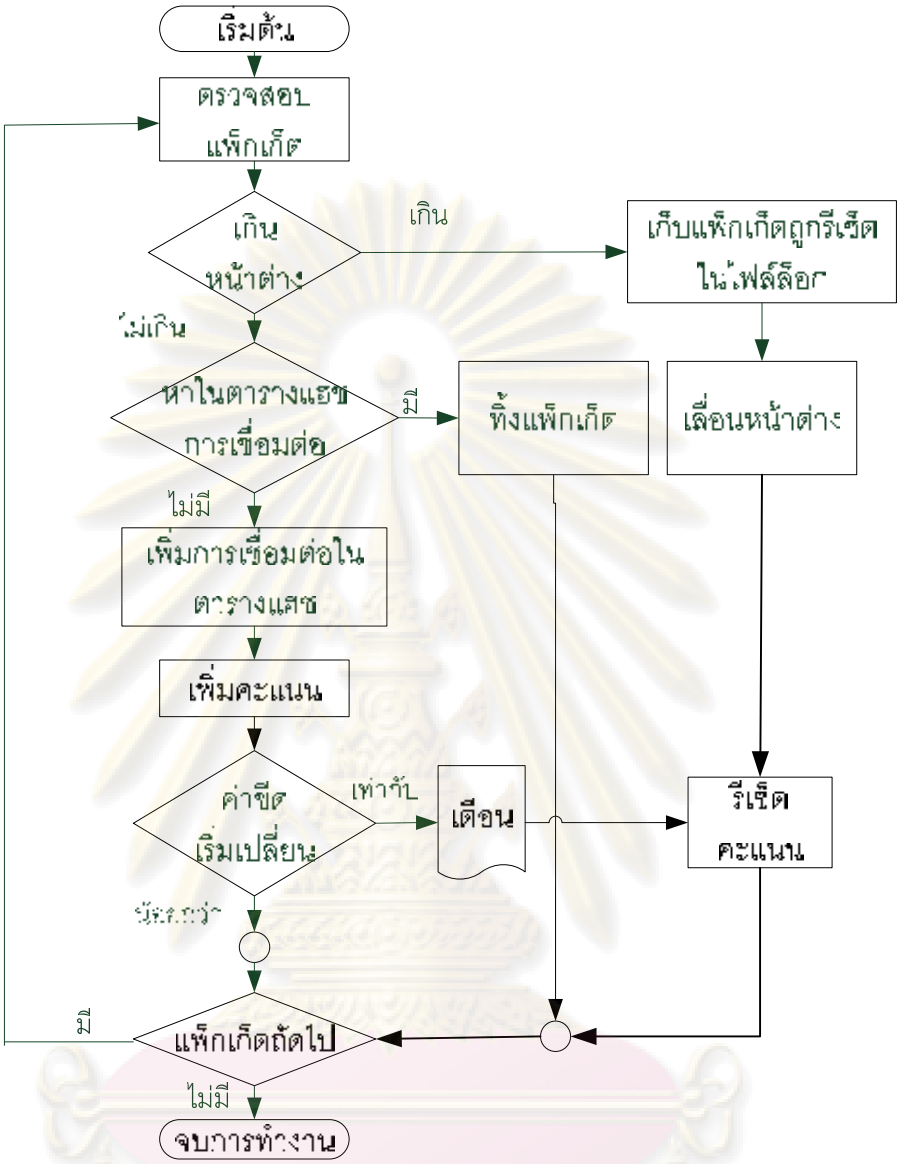
ในงานวิจัยนี้ใช้ Flow-portscan ซึ่งเป็นพรีโพรเซสเซอร์ในโปรแกรมสนอร์ต สำหรับตรวจจับการกราดตรวจของหนอน ตามรูปที่ 3.1 โดยการทำงานของพรีโพรเซสเซอร์ Flow-portscan มีดังนี้ เริ่มต้นจะแบ่งกลุ่มแพ็กเก็ตตามหมายเลขไอพีต้นทาง ถ้าหมายเลขไอพีต้นทางเดียวกันแต่มีหมายเลขไอพีปลายทาง หมายเลขพอร์ตปลายทาง หรือหมายเลขโปรโตคอลต่างกันให้นับเป็น 1 คะแนน แต่เนื่องจากงานวิจัยนี้มีขอบเขตเฉพาะการตรวจจับการกราดตรวจของหนอน ดังนั้นจะใช้เฉพาะหมายเลขไอพีปลายทางที่ต่างกันเท่านั้น [17] โดยเก็บคะแนนไว้ในหน่วยความจำ เมื่อคะแนนเกินค่าที่กำหนด สนอร์ตจะทำการเตือนว่ามีความผิดปกติเกิดขึ้น คะแนนถูกคิดเป็น 0 หรือ รีเซต (reset) และเริ่มนับใหม่ทุกครั้งเมื่อสิ้นสุดช่วงเวลาตรวจจับ โดยกระบวนการทำงานของพรีโพรเซสเซอร์สามารถแสดงได้ในรูปที่ 3.1

3.2 การศึกษาวิธีสุ่มตัวอย่าง

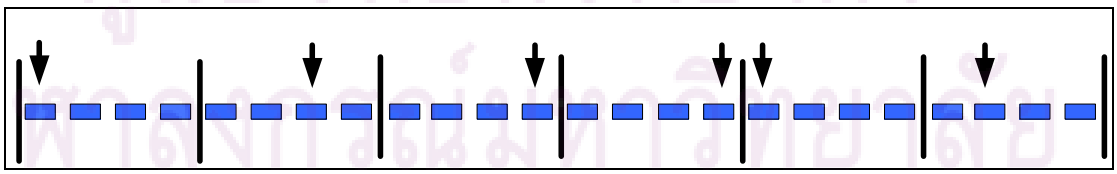
เพื่อลดจำนวนข้อมูลจราจรให้สามารถเก็บข้อมูลได้เป็นเวลานานขึ้น ผู้วิจัยได้ศึกษาการสุ่มตัวอย่างแบบต่างๆ [3] ทั้งการสุ่มตัวอย่างแบบมีชั้นภูมิ (stratified random sampling) การสุ่มตัวอย่างแบบมีระบบ (systematic sampling) และการสุ่มตัวอย่างแบบง่าย (simple random sampling) โดยมีรายละเอียดดังต่อไปนี้

3.2.1 การสุ่มตัวอย่างแบบมีชั้นภูมิ

การสุ่มตัวอย่างแบบมีชั้นภูมิถูกใช้ในมาตรฐาน RFC 3176 [18] ของเอสโฟลว์ การสุ่มตัวอย่างแบบมีชั้นภูมิจะแบ่งข้อมูลออกเป็นกลุ่มย่อยที่มีขนาดเท่ากันแล้วสุ่มข้อมูลมาหนึ่งครั้งจากกลุ่มย่อยนั้นๆ ทำให้สามารถตรวจจับการโจมตีที่มีรูปแบบได้ดี และโอกาสที่ข้อมูลจะถูกสุ่มออกมาจะมีเท่าๆกันเนื่องจากมีการกระจายการสุ่มไปทั่วทั้งข้อมูล ดังแสดงในรูปที่ 3.2



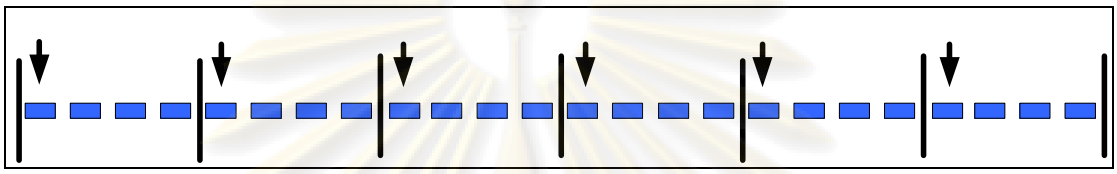
รูปที่ 3.1 กระบวนการทำงานของ Flow-portscan



รูปที่ 3.2 การสุ่มตัวอย่างแบบมีชั้นภูมิ

3.2.2 การสุ่มตัวอย่างแบบมีระบบ

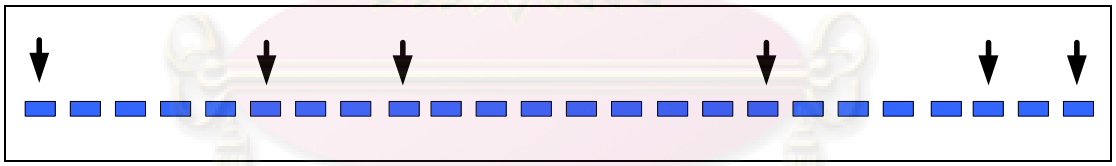
การสุ่มตัวอย่างแบบมีระบบจะเลือกข้อมูลที่ k ของกลุ่มย่อยแต่ละกลุ่ม ยกตัวอย่างเช่น กำหนดให้ k เป็น 1 ก็จะเลือกแพ็กเก็ตแรกของทุกๆกลุ่มข้อมูลย่อย ดังแสดงในรูปที่ 3.3 การสุ่มตัวอย่างแบบนี้ทำให้ยากต่อการตรวจจับการกราดตรวจที่มีรูปแบบได้ เช่น เมื่อการกราดตรวจของหนอนเกิดขึ้นที่แพ็กเก็ตที่ 4 ของกลุ่มข้อมูลย่อยก็จะไม่สามารถตรวจพบได้



รูปที่ 3.3 การสุ่มตัวอย่างแบบมีระบบ

3.2.3 การสุ่มตัวอย่างแบบง่าย

การสุ่มตัวอย่างแบบง่ายจะทำการสุ่มตัวอย่างจากข้อมูลทั้งหมดโดยไม่มีการแบ่งข้อมูลออกเป็นกลุ่มข้อมูลย่อยดังที่แสดงในรูปที่ 3.4 ซึ่งจะมีข้อเสียคือ การสุ่มตัวอย่างมีโอกาสที่จะกระจายไม่ทั่วทุกช่วงของข้อมูล เช่นเมื่อการกราดตรวจของหนอนอยู่ช่วงท้ายของข้อมูลแต่ปรากฏว่าช่วงท้ายของข้อมูลไม่มีการสุ่มตัวอย่าง ทำให้ไม่สามารถตรวจพบการกราดตรวจได้



รูปที่ 3.4 การสุ่มตัวอย่างแบบง่าย

3.2.4 ผลการศึกษาวิธีสุ่มตัวอย่างที่เหมาะสม

ผู้วิจัยได้เลือกวิธีการสุ่มตัวอย่างแบบมีชั้นภูมิสำหรับงานวิจัยนี้เพราะสามารถสุ่มได้ทั่วทั้งข้อมูลซึ่งดีกว่าการสุ่มแบบง่ายและได้ผลไม่เอนเอียง เนื่องจากแต่ละครั้งที่หยิบแพ็กเก็ตจะสุ่มทุกครั้งซึ่งดีกว่าการสุ่มแบบมีระบบ และการสุ่มตัวอย่างชนิดนี้ยังได้มีการนำไปประยุกต์ใช้ในเทคโนโลยีของ sFlow [18] ซึ่งเป็นเทคโนโลยีการสุ่มที่เป็นที่ยอมรับโดยทั่วไป อีกทั้งยังสะดวกและง่ายต่อการเขียนโปรแกรมเพราะใช้อัลกอริทึมที่ไม่ซับซ้อน

3.3 ผลกระทบของการสุ่มตัวอย่าง

เมื่อสุ่มตัวอย่างแพ็กเก็ต จำนวนแพ็กเก็ตที่ได้จากการสุ่มจะมีค่าไม่แน่นอน ในหัวข้อนี้จะแสดงให้เห็นถึงพฤติกรรมของแพ็กเก็ตเมื่อถูกสุ่มตัวอย่าง

3.3.1 **ความผิดพลาดจากการสุ่มตัวอย่าง**

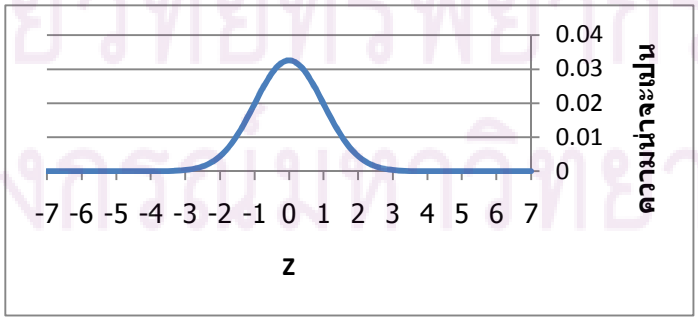
จำนวนแพ็กเก็ตที่ได้จากการสุ่มตัวอย่างจะน้อยกว่าจำนวนแพ็กเก็ตจริง การประมาณจำนวนแพ็กเก็ตการเชื่อมต่อของหนอนสามารถทำได้โดย กำหนดให้

N	แทนจำนวนการเชื่อมต่อทั้งหมด
n	แทนจำนวนการเชื่อมต่อจากการสุ่มตัวอย่าง
C	แทนจำนวนการเชื่อมต่อเฉลี่ยของหนอนเมื่อไม่ได้สุ่มตัวอย่าง
c	แทนจำนวนการเชื่อมต่อเฉลี่ยของหนอนจากการสุ่มตัวอย่าง
p	แทนความน่าจะเป็นที่การเชื่อมต่อของหนอนถูกสุ่มตัวอย่าง
$\overline{A}$	แทนจำนวนการเตือนเฉลี่ย
c_{base}	แทนค่าขีดเริ่มเปลี่ยน (threshold) ของจำนวนการเชื่อมต่อที่สนอร์ตตรวจจับได้

จะได้จำนวนการเชื่อมต่อของหนอนดังนี้ [19]

$$C = \frac{c}{n}N = pN \qquad (3.1)$$

และเนื่องจากสมการที่ 3.1 เป็นการคิดจำนวนการเชื่อมต่อจากการสุ่มตัวอย่าง ซึ่งเป็นค่าประมาณ โดยจากงานวิจัย [19] เมื่อสุ่มตัวอย่าง ค่าที่ได้จากการสุ่มตัวอย่างจะไม่แน่นอนซึ่งเขียนตารางการแจกแจงปกติได้ดังรูปที่ 3.5



รูปที่ 3.5 ตารางแจกแจงความถี่ปกติมาตรฐาน

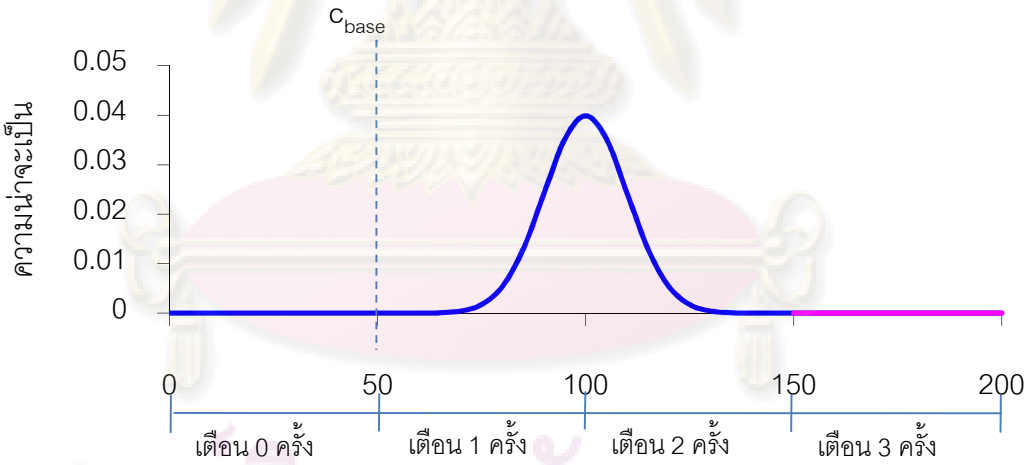
โดย Z คือ ตัวแปรสุ่มแบบปกติมาตรฐาน เราสามารถหา Z ได้จากสมการดังนี้

$$Z = \frac{x-c}{\sigma} \tag{3.2}$$

ส่วนเบี่ยงเบนมาตรฐาน สามารถหาได้จากสมการที่ 3.3 เมื่อ n มีค่ามากกว่า c มาก

$$\sigma = \sqrt{np(1-p)} \approx \sqrt{c} \tag{3.3}$$

สมมติให้สนอร์ตสามารถตรวจจับหนอนที่อัตราการกราดตรวจต่ำสุด 10 การเชื่อมต่อต่อวินาที ใช้หน้าต่างเวลาสำหรับตรวจจับ 10 วินาที ดังนั้นจะมี 100 การเชื่อมต่อ เมื่อสุ่มตัวอย่างด้วยอัตรา 50% จำนวนการเชื่อมต่อจะลดลงเหลือ 50 การเชื่อมต่อ ดังนั้นค่าขีดเริ่มเปลี่ยน (c_{base}) คือ 50 การเชื่อมต่อ ถ้าหนอนมีอัตราการกราดตรวจ 20 การเชื่อมต่อต่อวินาที ที่หน้าต่าง 10 วินาที หนอนจะมี 200 การเชื่อมต่อ แต่เมื่อสุ่มตัวอย่างด้วยอัตรา 50% จำนวนการเชื่อมต่อเฉลี่ยจะลดลงเหลือ 100 การเชื่อมต่อ (c) สามารถเขียนกราฟแจกแจงความน่าจะเป็นได้ดังรูปที่ 3.6



รูปที่ 3.6 จำนวนแพ็กเก็ตการเชื่อมต่อเทียบกับความผิดพลาดในการสุ่มตัวอย่าง

จากรูปที่ 3.6 ความน่าจะเป็นที่สนอร์ตจะไม่เตือน แพ็กเก็ตอยู่ในช่วง [0,50) ความน่าจะเป็นที่สนอร์ตจะเตือน 1 ครั้ง แพ็กเก็ตอยู่ในช่วง [50,100) ความน่าจะเป็นที่สนอร์ตจะเตือน 2 ครั้ง แพ็กเก็ตอยู่ในช่วง [100,150) และ ความน่าจะเป็นที่สนอร์ตจะเตือน 3 ครั้ง แพ็กเก็ตอยู่ในช่วง [150,200) ดังนั้น สามารถเขียนเป็นสมการหาจำนวนการเตือนเฉลี่ย ($\overline{A}$) ได้ดังนี้

$$\bar{A} = \sum_{i=1}^{\infty} iP_i \qquad (3.4)$$

โดยที่ i เป็นจำนวนการเตือนใดๆ และ P_i คือความน่าจะเป็นที่สนอร์ตจะเตือน i ครั้ง ความน่าจะเป็นของจำนวนการเตือน หาได้จากหาความน่าจะเป็นที่แพ็กเก็ตจะตกอยู่ในช่วงการเตือนนั้นดังสมการที่ 3.5 โดยเราสามารถหาได้โดยแปลง $c_{i,min}$ และ $c_{i,max}$ เป็น Z โดยใช้สมการที่ 3.2 และนำไปหาความน่าจะเป็นจากตารางการแจกแจงปกติ [20]

$$P_i = P[c_{i,\min} \le c_i < c_{i,\max}] \qquad (3.5)$$

ดังนั้นอัตราส่วนจำนวนการเตือนเฉลี่ยเมื่อสุ่มตัวอย่างกับจำนวนการเตือนเมื่อไม่ได้สุ่มตัวอย่าง (R) สามารถเขียนได้ดังสมการที่ 3.6

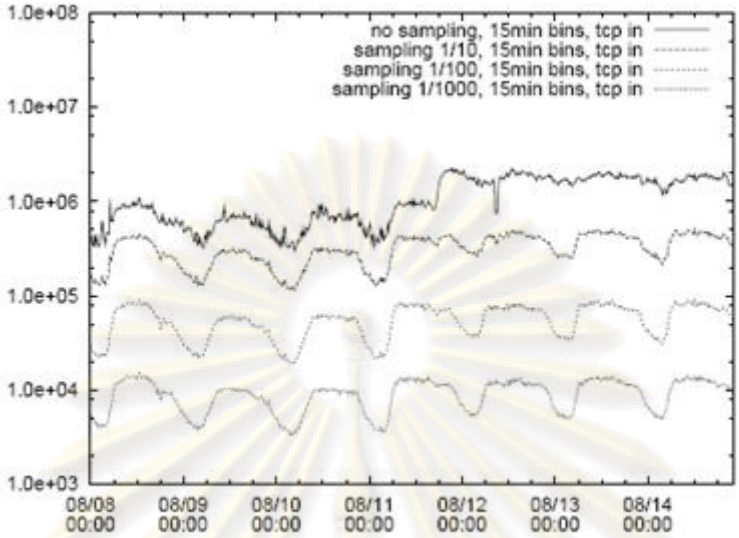
$$R = \frac{\bar{A}}{A_{100\%}} \qquad (3.6)$$

อัตราส่วนความผิดพลาดของการเตือนเมื่อสุ่มตัวอย่างแสดงดังสมการที่ 3.7 โดยถ้า E มีค่าลบจะเป็นผลลบลวง แต่ถ้ามีค่าบวกจะเป็นผลบวกลวง

$$E = \frac{\bar{A}}{A_{100\%}} - 1 \qquad (3.7)$$

3.3.2 จำนวนการเชื่อมต่อ

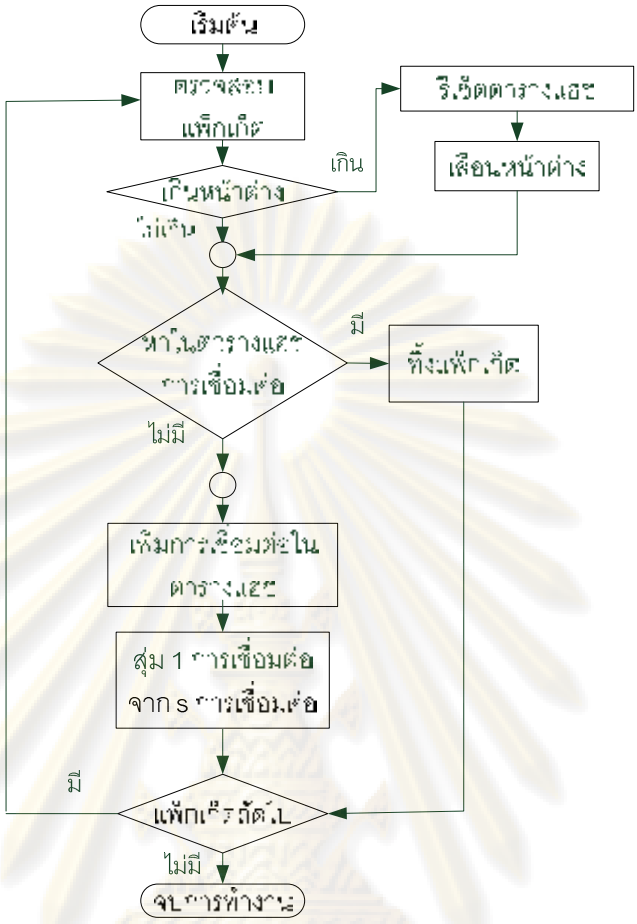
งานวิจัยของ Brauckhoff และคณะ [15] ศึกษาผลกระทบของโฟลว์จากการสุ่มตัวอย่าง พบว่าจำนวนโฟลว์ไม่ลดลงตามอัตราการสุ่มตัวอย่าง เนื่องจากความน่าจะเป็นที่โฟลว์จะถูกสุ่มตัวอย่างขึ้นอยู่กับจำนวนแพ็กเก็ตในโฟลว์นั้น จากรูปที่ 3.7 แกนตั้งเป็นจำนวนโฟลว์ แกนนอนเป็นเวลา เมื่อใช้อัตราการสุ่มตัวอย่าง 10% (1/10) จำนวนโฟลว์ควรจะลดลง 10 เท่า แต่จากผลการทดลอง จำนวนโฟลว์ลดลงจากเดิม 800,000 แพ็กเก็ต เหลือ 500,000 แพ็กเก็ต เท่านั้น ผู้วิจัยจึงมีสมมติฐานว่าพฤติกรรมของการเชื่อมต่อจะมีลักษณะเดียวกันเมื่อถูกสุ่มตัวอย่างแพ็กเก็ต เนื่องจากโฟลว์เป็นเซตย่อย (subset) ของการเชื่อมต่อ ถ้าสมมติฐานเป็นจริง จะเกิดผลบวกลวง เนื่องจาก เราลดค่าขีดเริ่มเปลี่ยนที่ตั้งไว้ในสนอร์ต ตามอัตราการสุ่มตัวอย่างเพื่อให้ได้จำนวนการเตือนของสนอร์ตเท่าเดิม แต่การเชื่อมต่อลดลงน้อยกว่าที่ควรจะเป็น



รูปที่ 3.7 จำนวนโฟลว์จากอัตราการสุ่มตัวอย่าง

3.4 การสุ่มตัวอย่างการเชื่อมต่อ

จาก 3.3.2 การสุ่มตัวอย่างแพ็กเก็ตแบบมีชั้นภูมิทำให้เกิดผลบวกลวงมากยิ่งขึ้น เพราะการติดต่อของโฮสต์โดยปกติแล้วมีจำนวนมากกว่า 1 แพ็กเก็ต และมีจำนวนแพ็กเก็ตไม่เท่ากัน ดังนั้นผู้วิจัยจึงได้คิดวิธีสุ่มตัวอย่างการเชื่อมต่อ ทำให้แต่ละการเชื่อมต่อมีความน่าจะเป็นที่ถูกสุ่มตัวอย่างเท่ากันโดยสร้างโปรแกรมที่ชื่อเอสดีพี (SDP - Scanning Detection Program) สำหรับสุ่มตัวอย่างการเชื่อมต่อและเก็บแพ็กเก็ต กระบวนการทำงานของการสุ่มตัวอย่างการเชื่อมต่อสามารถแสดงได้ดังรูปที่ 3.8 เริ่มต้นถ้าไม่มีแพ็กเก็ตเข้ามาเกินระยะเวลาที่กำหนด (มีค่ามากกว่าค่าหน้าต่างตรวจจับ) จะรีเซตตารางแฮช (ลบข้อมูลในตารางแฮช) และเลื่อนหน้าต่าง แต่ถ้าอยู่ในระยะเวลาที่กำหนด ข้อมูลการเชื่อมต่อของแพ็กเก็ตจะถูกตรวจสอบข้อมูลในตารางแฮช (โดยตรวจสอบหมายเลขไอพีต้นทางและหมายเลขไอพีปลายทาง) ถ้ามีการเชื่อมต่ออยู่ในตารางแฮชอยู่แล้ว จะทิ้งแพ็กเก็ตนั้น แต่ถ้าไม่มีการเชื่อมต่อในตารางแฮช จะเพิ่มข้อมูลการเชื่อมต่อใหม่ในตารางแฮช และแพ็กเก็ตนั้นจะถูกสุ่มตัวอย่าง โดยมีโอกาสถูกสุ่ม 1/s (สุ่ม 1 การเชื่อมต่อ ทุกๆ s การเชื่อมต่อ)



รูปที่ 3.8 กระบวนการสุ่มตัวอย่างการเชื่อมต่อ

3.5 การเพิ่มช่วงเวลาในการเก็บข้อมูล

เมื่อสุ่มตัวอย่างข้อมูล จะสามารถทำนายช่วงเวลาการเก็บข้อมูลจราจรที่เพิ่มขึ้นจากการสุ่มได้ดังนี้ ให้ K เป็นขนาดเนื้อที่ s เป็นอัตราการสุ่มตัวอย่าง และ B เป็นอัตราการใช้ข้อมูล การทำนายช่วงเวลาสำหรับเก็บข้อมูล (I) สามารถคำนวนได้ดังสมการที่ 3.8

$$I = \frac{K}{B \times s} \qquad (3.8)$$

ในบทนี้ได้กล่าวถึงแนวคิดการนำการสุ่มตัวอย่างมาใช้เพื่อเพิ่มระยะเวลาในการเก็บข้อมูล โดยศึกษาวิธีการสุ่มตัวอย่างที่เหมาะสมพบว่าการสุ่มตัวอย่างแบบมีชั้นภูมิเหมาะที่จะนำมาเก็บข้อมูลจราจร และทดสอบเก็บข้อมูลโดยการสุ่มตัวอย่างพบว่าสามารถเพิ่มระยะเวลาในการเก็บข้อมูลได้จริง ตัวอย่างเช่น การคำนวณเพื่อทำนายช่วงเวลาเก็บข้อมูลจราจรที่เพิ่มขึ้นจากการสุ่มตัวอย่างของคณะใหญ่ในมหาวิทยาลัยขนาด 40,000 คน มีวิธีการดังนี้ จากข้อมูลการจราจรที่

ผ่านเกตเวย์ของคณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ตั้งแต่เดือน มิถุนายน พ.ศ. 2550 ถึงเดือนมิถุนายน พ.ศ. 2551 พบว่ามีค่าเฉลี่ยการใช้แบนด์วิดท์ 78.03 เมกะบิตต่อวินาที (9.75 เมกะไบต์ต่อวินาที) โดยเป็นการรับข้อมูล 65.39 เมกะบิตต่อวินาที และส่งข้อมูล 12.64 เมกะบิตต่อวินาที แต่ในงานนี้ผู้วิจัยเก็บเฉพาะส่วนหัวของแพ็กเก็ตเพื่อนำมาตรวจจับพฤติกรรมการกราดตรวจของหนอน จึงเก็บเฉพาะ 68 ไบต์แรกเท่านั้น [21] โดยที่ขนาดของแพ็กเก็ตเฉลี่ยคือ 680 ไบต์ ซึ่งคิดเป็น 10 % นั่นหมายความว่าถ้าเก็บเฉพาะส่วนหัวของแพ็กเก็ต ข้อมูลจะถูกเก็บด้วยอัตรา 0.975 เมกะไบต์ต่อวินาที ด้วยขนาดเนื้อที่ 125 กิกะไบต์ จะสามารถเก็บข้อมูลได้ประมาณ 131,282 วินาที หรือประมาณ 36 ชั่วโมงเท่านั้น แต่ถ้าใช้การสุ่มตัวอย่างที่อัตราสุ่มต่ำลงก็จะสามารถเก็บข้อมูลได้ในช่วงเวลาที่มากขึ้น ดังนั้นจะเห็นว่าอัตราการสุ่มตัวอย่างจะแปรผกผันผันกับช่วงเวลาสำหรับเก็บข้อมูล โดยการทำนายช่วงเวลาสำหรับเก็บข้อมูลในตารางที่ 3.1 สามารถทำได้โดยอาศัยสมการที่ 3.8 นอกจากนี้ยังวิเคราะห์ผลกระทบที่เกิดขึ้นต่อการตรวจจับการกราดตรวจของสนอร์ต ทั้งพฤติกรรมการเชื่อมต่อ และอัตราส่วนความผิดพลาด โดยจะทดลองเพื่อทดสอบสมมติฐานและการวิเคราะห์ที่กล่าวไว้ในบทถัดไป

ตารางที่ 3.1 การทำนายช่วงเวลาที่สามารถเก็บข้อมูลได้เมื่อสุ่มตัวอย่าง

อัตราสุ่มตัวอย่าง (%)	n เลือก 1	ช่วงเวลาสำหรับเก็บข้อมูลโดยประมาณ (ชั่วโมง / วัน)
100	1	36 / 2
50	2	73 / 3
33.33	3	109 / 5
25	4	146 / 6
20	5	182 / 7
16.67	6	218 / 9
14.29	7	255 / 11
12.5	8	292 / 12
11.11	9	328 / 14
10	10	365 / 15

บทที่ 4

การทดลองและผลการทดลอง

ในบทนี้จะทดสอบความถูกต้องของสมมติฐาน ทั้งผลกระทบของการสุ่มตัวอย่างต่อพฤติกรรมการเชื่อมต่อ อัตราส่วนความผิดพลาด โดยจะกล่าวถึงวัตถุประสงค์ของการทดลอง เครื่องมือที่ใช้ในการทดลอง ข้อมูลสำหรับการทดลองขั้นตอนการทดลอง ผลการทดลอง และสรุปผลการทดลอง โดยมีรายละเอียดดังนี้

4.1 วัตถุประสงค์การทดลอง

เพื่อตรวจสอบสมมติฐานที่ตั้งไว้ว่าเป็นไปตามที่คาดไว้ในบทที่ 3 หรือไม่ โดยมีประเด็นหลักดังนี้

1. ทดลองสุ่มตัวอย่างเพื่อลดขนาดข้อมูล
2. ทดลองสุ่มตัวอย่างเพื่อวิเคราะห์ผลกระทบต่อพฤติกรรมการเชื่อมต่อ
3. ทดลองสุ่มตัวอย่างเพื่อวิเคราะห์ผลกระทบที่มีต่อความแม่นยำในการตรวจจับของสนอร์ต

4.2 เครื่องมือที่ใช้ในการทดลอง

เครื่องมือสำหรับการทดลองประกอบด้วย เครื่องคอมพิวเตอร์สำหรับเก็บและสุ่มตัวอย่างแพ็กเก็ต มีคุณสมบัติดังนี้ Pentium IV 2.66 กิกะเฮิร์ท หน่วยความจำ 512 เมกะไบต์ ติดตั้งระบบปฏิบัติการลีนุกซ์ 2.6 Fedora Core 4 เนื้อที่สำหรับเก็บข้อมูล 300 กิกะไบต์

4.3 ข้อมูลสำหรับทดลอง

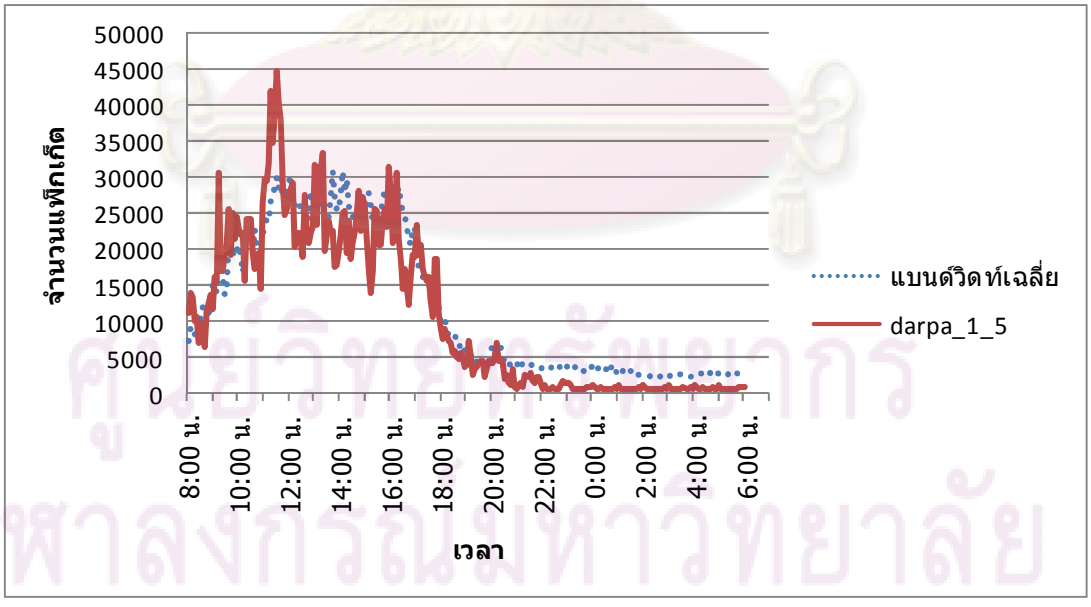
ในงานวิจัยนี้จำเป็นต้องมีการสร้างข้อมูลสำหรับทดลองซึ่งประกอบด้วย ข้อมูลควบคุม คือข้อมูลที่ผู้วิจัยสามารถกำหนดอัตราการกราดตรวจของหนอนได้ และข้อมูลจริงที่เก็บมาจากเครือข่ายโดยไม่มีการปรับเปลี่ยนใดๆ

4.3.1 ข้อมูลควบคุม

ข้อมูลควบคุมเป็นข้อมูลที่สร้างจากสภาพแวดล้อมควบคุมประกอบด้วย ข้อมูลสะอาด (ไม่มีการโจมตีใดๆ) และข้อมูลหนอนผนวกกัน โดยข้อมูลสะอาดใช้ข้อมูลมาตรฐานที่ใช้สำหรับ

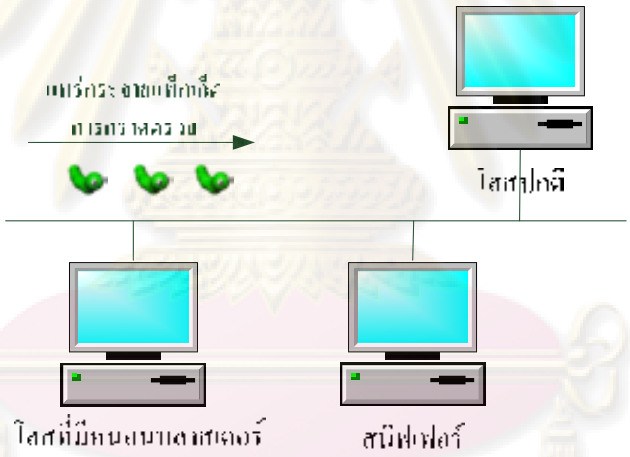
ทดสอบระบบตรวจจับผู้บุกรุก ส่วนข้อมูลหนอนเป็นข้อมูลที่ผู้วิจัยผลิตขึ้นเอง โดยมีรายละเอียดดังนี้

ข้อมูลสะอาดเป็นข้อมูลจากห้องปฏิบัติการลิงคอล์น [22] แห่งสถาบันเทคโนโลยีแมสซาซูเซตส์ เป็นข้อมูลที่ชื่อว่า 1999 DARPA Intrusion Detection Evaluation Data Set หรือ DARPA99 โดยสร้างจากสถิติการใช้งานของเครือข่ายของฐานทัพอากาศสหรัฐ เช่น การใช้อีเมล เว็บไซต์ FTP telnet และ web server ข้อมูลนี้เป็นข้อมูลมาตรฐานที่ใช้ในงานวิจัยต่างๆสำหรับตรวจสอบระบบตรวจจับผู้บุกรุก เช่น ในงานวิจัยของ Al-Hammadi [23] และ Song [24] ข้อมูลมาตรฐานมีทั้งหมด 5 สัปดาห์ (ไม่มีข้อมูลวันเสาร์และอาทิตย์) โดยข้อมูลสัปดาห์ที่หนึ่งและสามเป็นข้อมูลสะอาด (clean data) ที่ไม่มีการโจมตีหรือความผิดปกติ ส่วนข้อมูลสัปดาห์ที่สอง สี่ และห้า เป็นข้อมูลที่มีการโจมตี เนื่องจากข้อมูลสะอาดในสัปดาห์ที่หนึ่งและห้ามีพฤติกรรมการใช้งานที่เหมือนกันดังนั้นเพื่อความสะดวกผู้วิจัยจะเลือกข้อมูลในสองสัปดาห์นี้มา 1 วัน เพื่อเป็นตัวแทนของข้อมูลสะอาดทั้งหมด ผู้วิจัยเลือกข้อมูลสัปดาห์ที่หนึ่ง วันที่ห้ามาใช้เป็นข้อมูลสะอาดเนื่องจากเป็นข้อมูลที่มีการใช้แบนด์วิดท์ใกล้เคียงกับข้อมูลเฉลี่ยทั้งสองอาทิตย์มากที่สุดดังรูปที่ 4.1 ข้อมูล DARPA99 สัปดาห์ที่ 1 วันที่ 5 ครอบคลุมระยะเวลา 22 ชั่วโมง จำนวน 2,833,054 แพ็กเก็ต ขนาด 565.9 เมกะไบต์



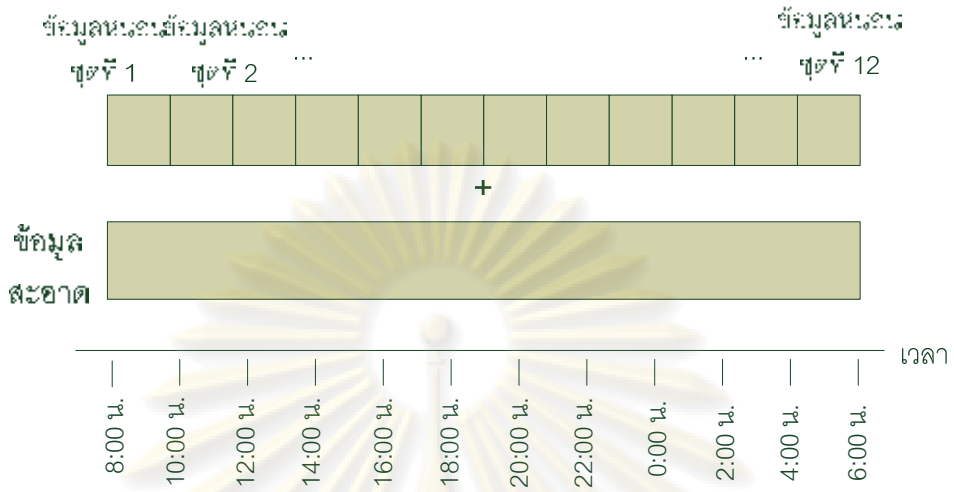
รูปที่ 4.1 ข้อมูลสะอาด

ข้อมูลหนอนเป็นข้อมูลที่ได้จากเครือข่ายจำลองในห้องปฏิบัติการ โดยการติดตั้งเครือข่ายและโฮสต์ดังรูปที่ 4.2 เมื่อเริ่มโปรแกรมหนอนบลาสเตอร์บนเครื่องคอมพิวเตอร์ หนอนจะส่งแพ็กเก็ตการเชื่อมต่อจำนวนมากเพื่อกราดตรวจหาโฮสต์ที่มีช่องโหว่ แพ็กเก็ตการเชื่อมต่อจะถูกเก็บในรูปแบบไฟล์ .cap ด้วยโปรแกรมสนิฟเฟอร์ สาเหตุที่เลือกหนอนบลาสเตอร์ในการทดลองนี้เพราะเป็นหนอนที่มีอัตราการกราดตรวจต่ำ (ที่ 10 การเชื่อมต่อต่อวินาที) ถ้าตรวจจับได้ก็สามารถตรวจจับหนอนที่มีอัตราการกราดตรวจสูงชนิดอื่นได้ ผลการสร้างข้อมูลหนอน ได้แพ็กเก็ตจากหนอนบลาสเตอร์จำนวน 66,879 แพ็กเก็ต มีอัตราการกราดตรวจเฉลี่ย 11 การเชื่อมต่อต่อวินาที ผู้วิจัยได้สำเนาและแก้ไขอัตราการกราดตรวจของหนอนให้เป็น 10 20 30 40 50 60 70 80 90 และ 100 การเชื่อมต่อต่อวินาทีตามลำดับ ดังนั้นข้อมูลหนอนประกอบด้วยหนอนที่มีอัตราการกราดตรวจตั้งแต่ 10 ถึง 100 การเชื่อต่อต่อวินาที ครอบคลุมระยะเวลา 2 ชั่วโมง มีจำนวนแพ็กเก็ต 667,080 แพ็กเก็ต ขนาด 50 เมกะไบต์



รูปที่ 4.2 โครงสร้างเครือข่ายสำหรับเก็บข้อมูลหนอน

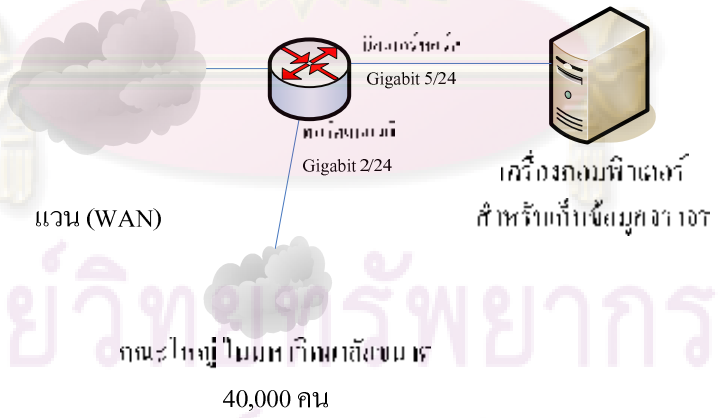
ในการรวมข้อมูลสะอาดและข้อมูลหนอนเข้าด้วยกัน ผู้วิจัยได้ปรับเปลี่ยนช่วงเวลาของข้อมูลหนอนให้อยู่ในช่วงเวลาของข้อมูลสะอาด โดยช่วงเวลาระหว่างแพ็กเก็ตของข้อมูลหนอนยังคงเท่าเดิม แล้วนำมาผนวกกันโดยใช้เวลาเป็นตัวอ้างอิง วิธีการรวมข้อมูลเช่นนี้ทำให้รูปแบบการกราดตรวจของหนอนไม่เปลี่ยนแปลง เนื่องจากข้อมูลสะอาดมีระยะเวลา 22 ชั่วโมง แต่ระยะเวลาของข้อมูลหนอนมีประมาณ 2 ชั่วโมง ดังนั้นผู้วิจัยจึงคัดลอกและแก้ไขช่วงเวลาของข้อมูลหนอนจำนวน 11 ไฟล์ เพื่อให้ได้ข้อมูลหนอนที่ต่อเนื่องกัน 22 ชั่วโมง และรวมกับข้อมูลสะอาดดังรูปที่ 4.3 โดยวิธีการสร้างข้อมูลควบคุมโดยละเอียดสามารถดูได้ที่ภาคผนวก ข



รูปที่ 4.3 การผนวกข้อมูลสะอาดกับข้อมูลหนอน

4.3.2 ข้อมูลจริง

ข้อมูลจริงเป็นข้อมูลของคณะใหญ่ในมหาวิทยาลัยขนาด 40,000 คน มีโครงสร้างเครือข่ายดังรูปที่ 4.4 โดยมีรายละเอียดคือ ระยะเวลา 20 ชั่วโมง จำนวน 153,064,243 แพ็กเก็ต ขนาด 125 กิกะไบต์



รูปที่ 4.4 โครงสร้างเครือข่ายสำหรับเก็บข้อมูลคณะใหญ่แห่งหนึ่ง

4.4 ขั้นตอนการทดลอง

ขั้นตอนการทดลองต่อไปนี้ทดสอบว่า การสุ่มตัวอย่างมีผลต่อขนาดข้อมูล พฤติกรรมจากเชื่อมต่อของหนอน และ ความแม่นยำในการตรวจจับของสนอร์ตอย่างไร

4.4.1 ผลการสุ่มตัวอย่างเพื่อลดขนาดข้อมูล

ในหัวข้อ 3.5 ได้ทำนายการเก็บข้อมูลไว้ ดังนั้นในหัวข้อนี้จะทดลองสุ่มตัวอย่างเพื่อหาการลดขนาดข้อมูล ข้อมูลที่ทดลองใช้ข้อมูลสะอาด มีขนาดข้อมูล 593,379,612 ไบต์ โดยสุ่มตัวอย่างแบบมีชั้นภูมิและสุ่มตัวอย่างการเชื่อมต่อ มีอัตราการสุ่มตัวอย่างตั้งแต่ 100% ไปจนถึง 10% ผลการทดลองแสดงได้ดังตารางที่ 4.1

ตารางที่ 4.1 ขนาดข้อมูลหลังจากสุ่มตัวอย่างเฉพาะหัวแพ็กเก็ต

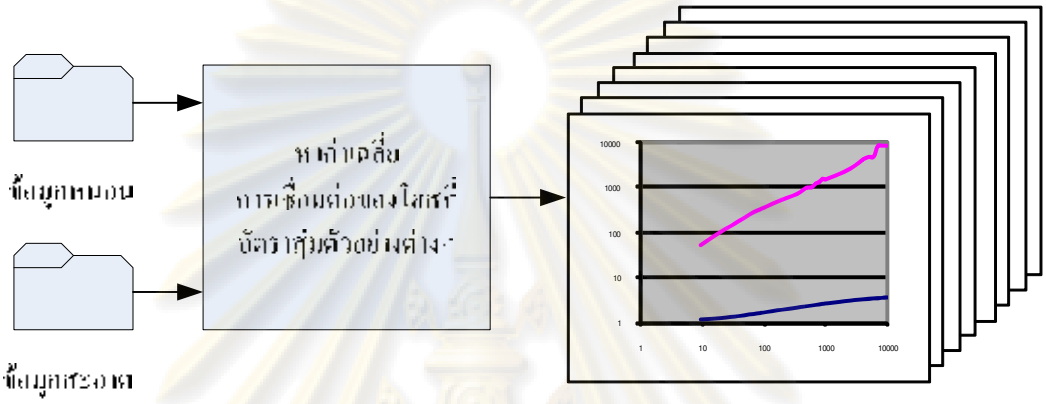
อัตราสุ่มตัวอย่าง (%) เฉพาะหัวแพ็กเก็ต	n เลือก 1	ขนาดข้อมูล (ไบต์)	
		สุ่มตัวอย่างแบบมีชั้นภูมิ	สุ่มตัวอย่างการเชื่อมต่อ
100	1	220,617,459	1,021,346
50	2	110,309,725	510,508
33.33	3	110,309,725	340,510
25	4	73,539,868	255,615
20	5	55,152,805	204,357
16.67	6	44,125,505	170,343
14.29	7	36,773,259	145,859
12.5	8	31,517,232	127,674
11.11	9	27,577,297	113,504
10	10	22,062,134	102,189

ผลการทดลองแสดงให้เห็นว่าเมื่อสุ่มตัวอย่างทำให้ข้อมูลลดลงจริง โดยเมื่อใช้วิธีสุ่มตัวอย่างแบบมีชั้นภูมิขนาดข้อมูลจะลดลงตามสมการที่ 3.8 แต่ใช้วิธีสุ่มตัวอย่างการเชื่อมต่อขนาดข้อมูลจะลดลงต่ำกว่าเสมอเนื่องจากเก็บเฉพาะการเชื่อมต่อเท่านั้น โดยขนาดของข้อมูลที่ลดลงไม่แน่นอน ขึ้นอยู่กับจำนวนการเชื่อมต่อของข้อมูล

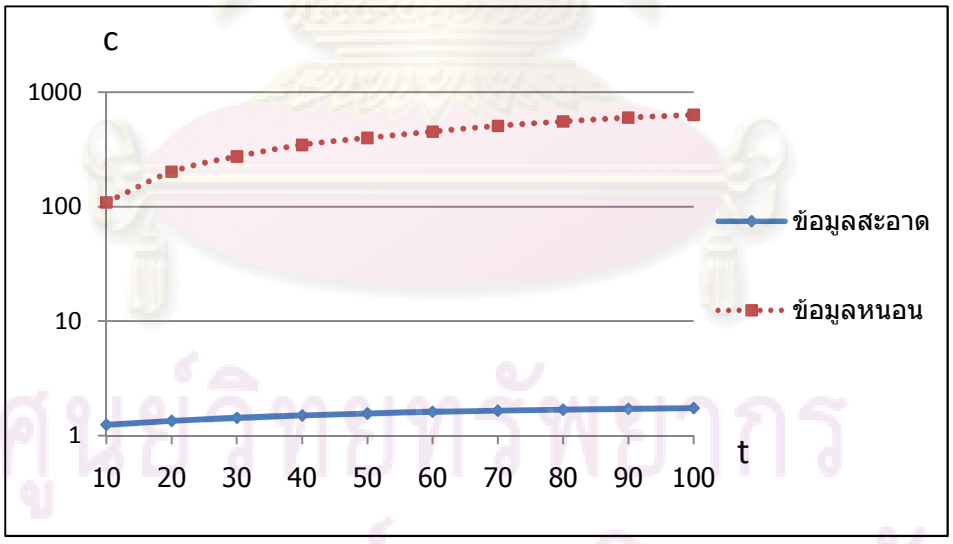
4.4.2 ผลกระทบที่มีต่อจำนวนการเชื่อมต่อ

ผู้วิจัยหาค่าเฉลี่ยการเชื่อมต่อของโฮสต์ปกติจากข้อมูลสะอาด และโฮสต์ที่ติดหนอนจากข้อมูลหนอน โดยมีกระบวนการทดลองดังรูปที่ 4.5 มีรายละเอียดดังนี้ ข้อมูลหนอนกับข้อมูล

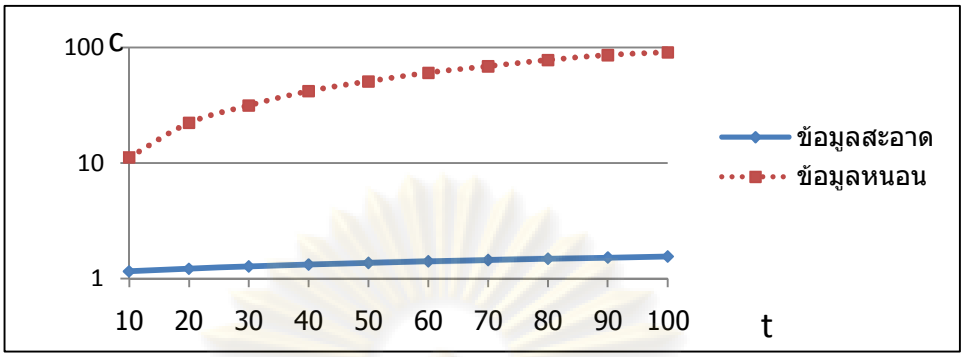
สะอาดจะถูกนำมาหาค่าเฉลี่ยการเชื่อมต่อของโฮสต์ ที่หน้าต่างเวลา 10, 30, 50, 70, 100 วินาที แล้วนำมาสร้างกราฟเพื่อเปรียบเทียบจำนวนการเชื่อมต่อที่อัตราสุ่มตัวอย่าง 100 % 10 % และ 1 % ตามลำดับ โดยกราฟแต่ละอันแทนอัตราสุ่มตัวอย่าง มีแกนนอนเป็นหน้าต่างและแกนตั้งเป็นจำนวนการเชื่อมต่อเฉลี่ย ผลการทดลองแสดงได้ดังรูปที่ 4.6 ถึง 4.8



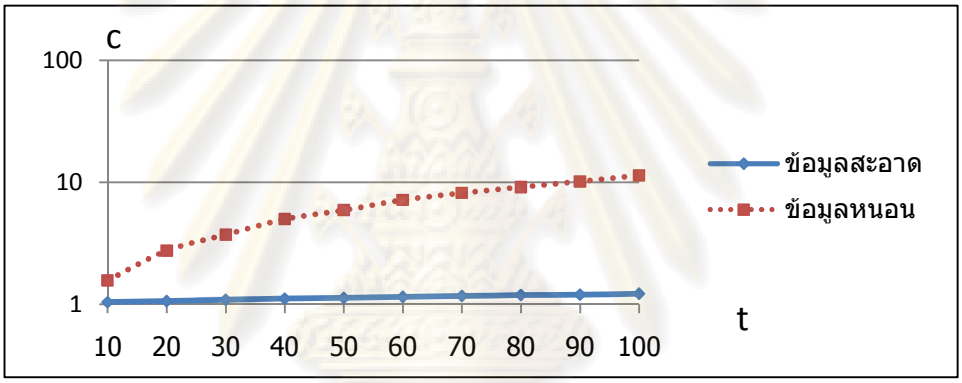
รูปที่ 4.5 การหาจำนวนแพ็กเก็ตเฉลี่ยในแต่ละอัตราการสุ่มตัวอย่าง



รูปที่ 4.6 จำนวนการเชื่อมต่อเฉลี่ยของโฮสต์ที่อัตราการสุ่มตัวอย่าง 100 %



รูปที่ 4.7 จำนวนการเชื่อมต่อเฉลี่ยของโฮสต์ที่อัตราการสุ่มตัวอย่าง 10%



รูปที่ 4.8 จำนวนการเชื่อมต่อเฉลี่ยของโฮสต์ที่อัตราการสุ่มตัวอย่าง 1%

จากผลการทดลองพบว่าเมื่อใช้อัตราการสุ่มตัวอย่างน้อยลง จำนวนการเชื่อมต่อของหนอนจะลดลงจนใกล้เคียงกับจำนวนการเชื่อมต่อของโฮสต์ปกติที่ระดับหน้าต่างแคบ จึงทำให้สนอร์ตไม่สามารถแยกแยะโฮสต์ที่ติดหนอนและโฮสต์ปกติได้

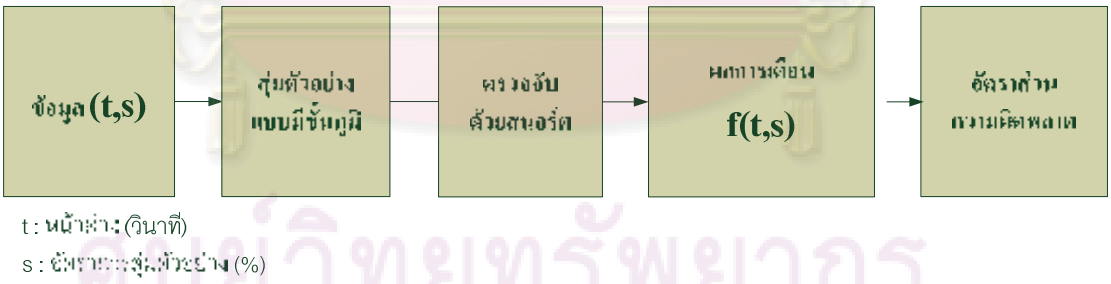
4.4.3 ผลกระทบที่มีต่อความแม่นยำในการตรวจจับ

ในส่วนนี้เป็นการทดลองสุ่มตัวอย่างข้อมูลควบคุมเพื่อวิเคราะห์ว่าหลังจากสุ่มตัวอย่างแล้วจำนวนการเตือนของหนอนที่อัตราการกราดตรวจต่างๆ จะมีความแม่นยำมากน้อยเพียงใด จากนั้นจึงทดลองสุ่มตัวอย่างกับข้อมูลจริง เพื่อดูว่ามีความแตกต่างกับข้อมูลทดลองอย่างไร โดยความแม่นยำในการทดลองขึ้นอยู่กับอัตราส่วนความผิดพลาดของการเตือน โดยจะทดลองกับการสุ่มตัวอย่าง 2 ชนิดคือ การสุ่มตัวอย่างแบบมีชั้นภูมิ และการสุ่มตัวอย่างการเชื่อมต่อ เพื่อให้เข้าใจง่ายและไม่สับสน ในการทดลองในบทนี้จะแสดงผลการทดลองที่หน้าต่าง 10 วินาทีเท่านั้น ส่วนผล

การทดลองที่หน้าต่าง 30 50 70 และ 100 วินาที ซึ่งมีผลการทดลองที่เป็นไปในแนวทางเดียวกัน สามารถดูได้ในภาคผนวก ก

4.3.3.1 การทดลองสำหรับการสุ่มตัวอย่างแบบมีชั้นภูมิ

การทดลองนี้มีรายละเอียดดังรูปที่ 4.9 มีรายละเอียดการทำงานดังนี้ ข้อมูลควบคุมและข้อมูลจริงจะถูกสุ่มตัวอย่างแบบมีชั้นภูมิ โดยมีอัตราการสุ่มตัวอย่างและหน้าต่างเวลาต่างๆ สมมติให้ตัวแปร t เป็นหน้าต่างเวลา และตัวแปร s เป็นอัตราการสุ่มตัวอย่าง หน้าต่างเวลามีค่าตั้งแต่ 10 30 50 70 และ 100 วินาที และ อัตราการสุ่มตัวอย่างมีค่าตั้งแต่ 100%, 50%, 25%, 20%, 10%, 2%, 1% โดยในแต่ละหน้าต่างเวลา ผู้วิจัยใช้อัตราการสุ่มตัวอย่างที่ตรวจจับข้อมูลควบคุมด้วยสนอร์ตแล้วไม่พบโฮสต์ที่เป็นข้อมูลปกติ แต่พบเฉพาะการกราดตรวจของหนอนเท่านั้น เซตของ t, s ทั้งหมดที่ตรวจจับแล้วไม่พบโฮสต์ปกติคือ {10,100}, {10,50}, {10,25}, {10,20}, {10,10}, {10,5}, {30,100}, {30,50}, {30,25}, {30,20}, {30,10}, {30,5}, {30,2}, {50,100}, {50,50}, {50,25}, {50,20}, {50,10}, {50,5}, {50,2}, {50,1}, {70,100}, {70,50}, {70,25}, {70,20}, {70,10}, {70,5}, {70,2}, {70,1}, {100,100}, {100,50}, {100,25}, {100,20}, {100,10}, {100,5}, {100,2}, {100,1} จากนั้นนำข้อมูลที่ได้จากการสุ่มตัวอย่างไปตรวจจับหนอนด้วยโปรแกรมสนอร์ต ผลลัพธ์คือจำนวนการเตือนของสนอร์ต จากนั้นนำมาวิเคราะห์อัตราส่วนความผิดพลาดเพื่อวิเคราะห์ความแม่นยำในการตรวจจับหนอน

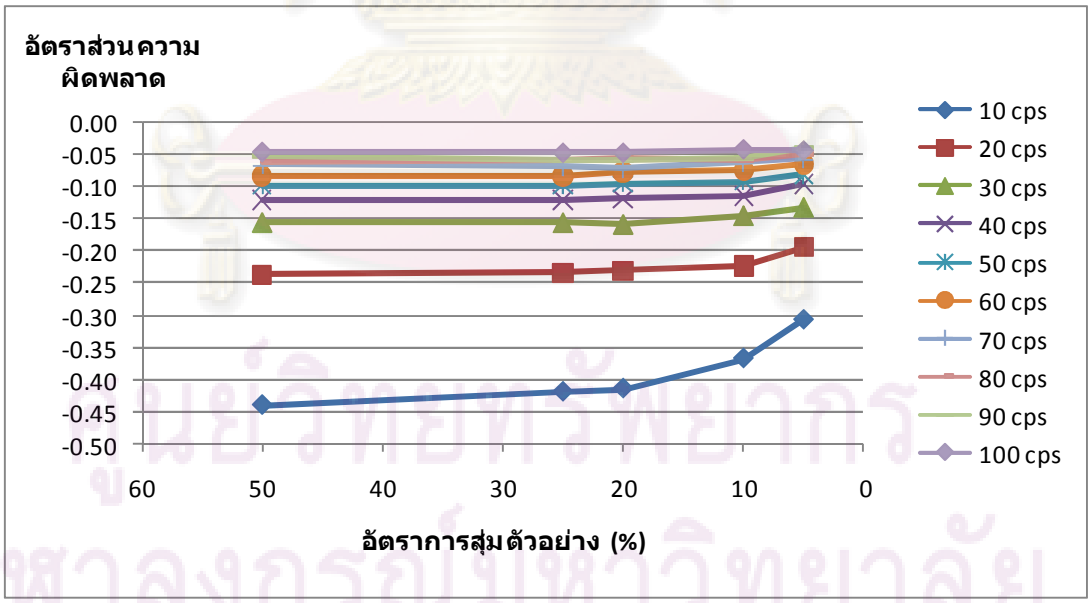


รูปที่ 4.9 การทดสอบความแม่นยำโดยการสุ่มตัวอย่างแบบมีชั้นภูมิ

เมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลควบคุมพบว่าได้อัตราส่วนความผิดพลาดดังตารางที่ 4.2 และรูปที่ 4.10

ตารางที่ 4.2 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิ กับข้อมูลควบคุมที่หน้าต่าง 10 วินาที

อัตราการกราตรวจ(cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)
10	-0.44	-0.42	-0.41	-0.37	-0.31
20	-0.24	-0.23	-0.23	-0.22	-0.19
30	-0.16	-0.16	-0.16	-0.15	-0.13
40	-0.12	-0.12	-0.12	-0.11	-0.10
50	-0.10	-0.10	-0.10	-0.09	-0.08
60	-0.08	-0.08	-0.08	-0.07	-0.07
70	-0.07	-0.07	-0.07	-0.06	-0.06
80	-0.06	-0.06	-0.05	-0.06	-0.05
90	-0.05	-0.06	-0.06	-0.05	-0.04
100	-0.05	-0.05	-0.05	-0.04	-0.04



รูปที่ 4.10 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิ กับข้อมูลควบคุมที่หน้าต่าง 10 วินาที

จากผลการทดลอง เมื่อสุ่มตัวอย่างจากข้อมูลควบคุมพบว่า ยิ่งหนอนที่มีอัตราการกราดตรวจสูง (มีจำนวนการเชื่อมต่อมากกว่าค่าต่ำสุดที่สนอร์ตตรวจจับได้) และอัตราการสุ่มตัวอย่างน้อยลง อัตราส่วนความผิดพลาดก็จะน้อยลง เป็นไปตามที่ได้คำนวณไว้ในหัวข้อ 3.3.1 โดยผลการคำนวณแสดงในตารางที่ 4.3

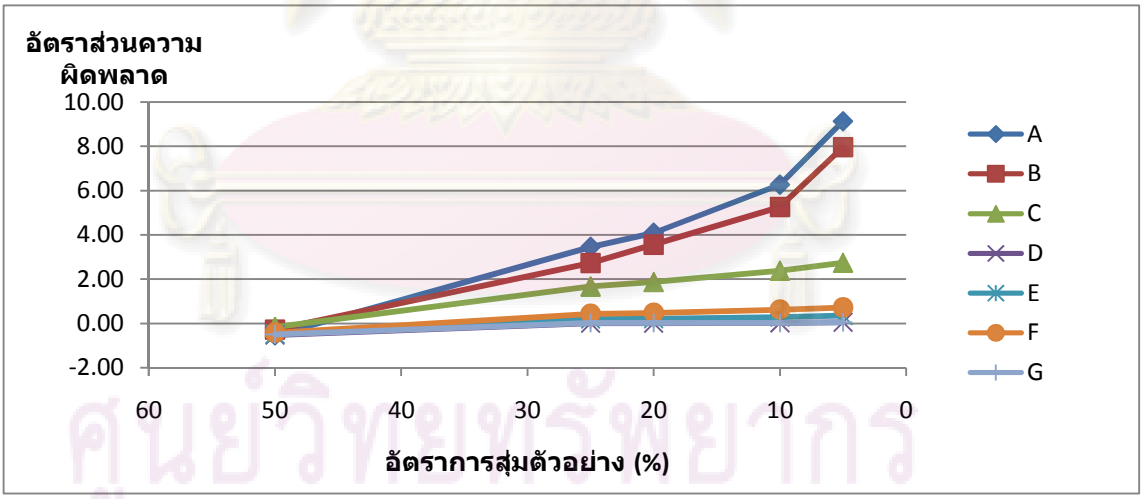
ตารางที่ 4.3 ผลการคำนวณอัตราส่วนความผิดพลาด
เมื่อสุ่มตัวอย่างกับข้อมูลควบคุมที่หน้าต่าง 10 วินาที

อัตราการกราดตรวจ(cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)
10	-0.44	-0.42	-0.41	-0.37	-0.29
20	-0.23	-0.22	-0.22	-0.20	-0.15
30	-0.16	-0.15	-0.15	-0.13	-0.10
40	-0.12	-0.11	-0.11	-0.10	-0.07
50	-0.09	-0.09	-0.09	-0.08	-0.06
60	-0.08	-0.08	-0.07	-0.07	-0.05
70	-0.07	-0.07	-0.06	-0.06	-0.04
80	-0.06	-0.06	-0.06	-0.05	-0.04
90	-0.05	-0.05	-0.05	-0.04	-0.03
100	-0.05	-0.05	-0.04	-0.04	-0.03

เมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลจริง ถ้าให้ความผิดพลาดในการเตือน 1 ครั้ง ก่อให้เกิดอัตราส่วนความผิดพลาดไม่เกิน 5 % (เป็นค่าที่ใช้ในงานวิจัยโดยทั่วไป เช่น ในงานของ [25] และผู้วิจัยเห็นว่าเป็นค่าความผิดพลาดที่ไม่ก่อให้เกิดนัยสำคัญ) ดังนั้นการเสียการเตือนหรือมีการเตือนเพิ่มขึ้น 1 ครั้ง ต้องทำให้มีอัตราส่วนความผิดพลาดไม่เกิน 5 % ดังนั้นสนอร์ตต้องเตือนอย่างน้อย 20 ครั้ง หมายเลขไอพีที่มีการเตือนมากกว่า 20 ครั้ง มีอัตราส่วนความผิดพลาด ดังตารางที่ 4.4 และรูปที่ 4.11

ตารางที่ 4.4 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลจริงที่หน้าต่าง 10 วินาที

หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)
A	27.62	-0.55	3.45	4.09	6.27	9.14
B	29.75	-0.28	2.72	3.56	5.26	7.95
C	41.19	-0.17	1.66	1.87	2.38	2.74
D	84.98	-0.53	0.00	0.01	0.02	0.05
E	87.73	-0.49	0.19	0.21	0.28	0.35
F	88.90	-0.42	0.44	0.48	0.62	0.72
G	407.30	-0.50	0.02	0.02	0.02	0.05



รูปที่ 4.11 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลจริงที่หน้าต่าง 10 วินาที

จากผลการทดลอง หมายเลขไอพี A, B, C, D, E และ F มีผลบวกลวงจำนวนมากและมีค่าไม่ตรงกับผลการคำนวณในตารางที่ 4.5 เนื่องจากในแต่ละการเชื่อมต่อมีแพ็กเก็ตมากกว่า 1 แพ็กเก็ต โดยปกติแล้วการกราดตรวจของหนอนจะมีการส่งแพ็กเก็ตสำหรับการเชื่อมต่อเพียง 1

แพ็กเก็ตเท่านั้น แต่การเชื่อมต่อที่ใช้มากกว่า 1 แพ็กเก็ตและมีพฤติกรรมการกราดตรวจ อาจเกิดจากการใช้โปรแกรมกราดตรวจเพื่อตรวจสอบเครือข่าย หรือโปรแกรมเพียร์ทูเพียร์ ซึ่งมีพฤติกรรมใกล้เคียงกับการกราดตรวจของหนอน [26] โดยจำนวนการเตือนเหล่านี้เพิ่มขึ้นมากเมื่อสุ่มตัวอย่างซึ่งเป็นไปตามสมมติฐานที่ได้กล่าวไว้ในหัวข้อที่ 3.3.2

ตารางที่ 4.5 ผลการคำนวณอัตราส่วนความผิดพลาด
เมื่อสุ่มตัวอย่างกับข้อมูลจริงที่หน้าต่าง 10 วินาที

หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)
A	27.62	-0.21	-0.18	-0.17	-0.15	-0.11
B	29.75	-0.16	-0.15	-0.15	-0.13	-0.10
C	41.19	-0.10	-0.11	-0.11	-0.10	-0.07
D	84.98	-0.06	-0.05	-0.05	-0.05	-0.04
E	87.73	-0.06	-0.05	-0.05	-0.05	-0.03
F	88.90	-0.05	-0.05	-0.05	-0.04	-0.03
G	407.30	-0.01	-0.01	-0.01	-0.01	-0.01

นอกจากนี้ยังพบโฮสต์ที่ถูกเตือนเพิ่มขึ้นมาเป็นจำนวนมากเมื่อใช้อัตราการสุ่มตัวอย่างน้อยลง ดังแสดงในตารางที่ 4.6 ซึ่งถือเป็นผลบวกลวงเนื่องจากเมื่อไม่ได้สุ่มตัวอย่างสนอร์ตไม่ตรวจพบโฮสต์เหล่านี้

ตารางที่ 4.6 จำนวนโฮสต์ที่เป็นผลบวกลวงเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิ

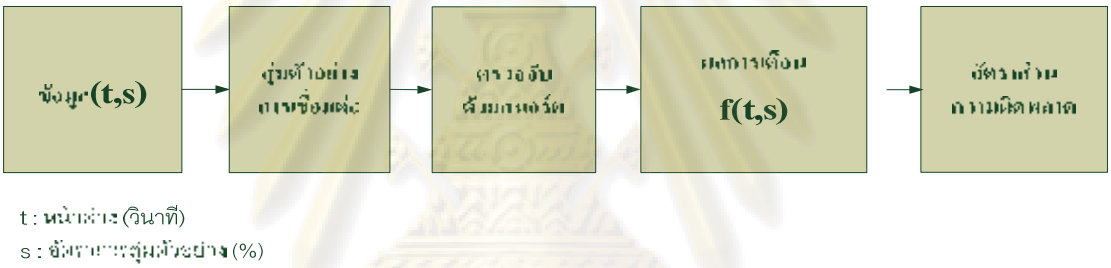
หน้าต่าง	สุ่ม 100 % (1 เอา 1)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
10	0	0	110	235	721	1121	-	-
30	0	4	13	19	107	435	950	1336
50	0	0	7	10	28	190	607	973
70	0	0	3	5	15	83	427	759

ตารางที่ 4.6 (ต่อ) จำนวนโฮสต์ที่เป็นผลบวกลวงเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิ

หน้าต่าง	สุ่ม 100 % (1 เอา 1)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
100	0	2	2	2	4	11	24	93

4.4.3.2 การทดลองสำหรับการสุ่มตัวอย่างการเชื่อมต่อ

การทดลองนี้เหมือนกับการสุ่มตัวอย่างแบบมีชั้นภูมิ แต่เปลี่ยนวิธีการสุ่มตัวอย่างเป็นการสุ่มตัวอย่างการเชื่อมต่อเท่านั้น โดยมีรายละเอียดดังรูปที่ 4.12



รูปที่ 4.12 การทดสอบความแม่นยำโดยการสุ่มตัวอย่างการเชื่อมต่อ

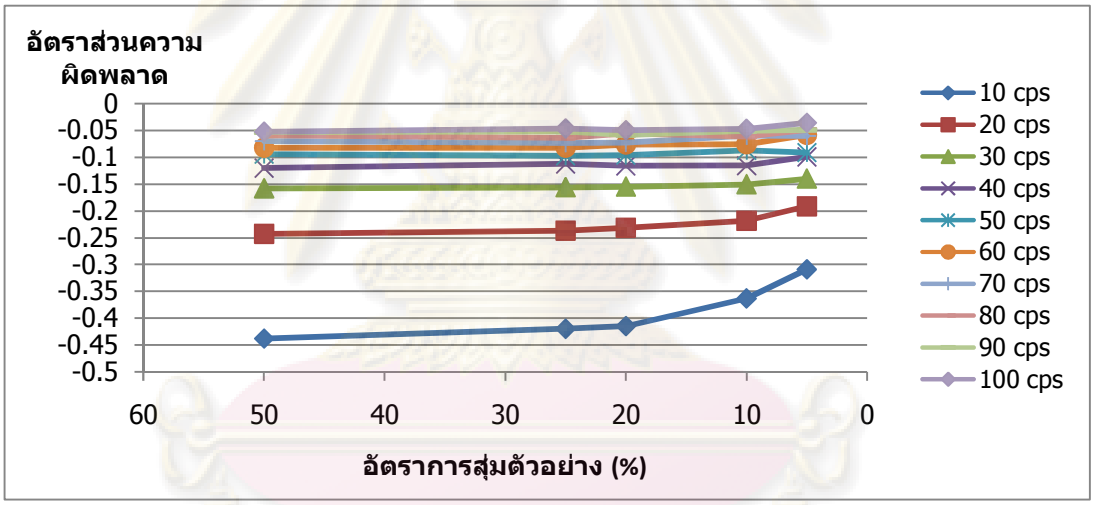
เมื่อสุ่มตัวอย่างข้อมูลควบคุมด้วยวิธีสุ่มตัวอย่างการเชื่อมต่อพบว่าได้ผลการทดลองดังตารางที่ 4.7 และรูปที่ 4.13 โดยพบว่าผลการทดลองมีค่าใกล้เคียงกับการสุ่มตัวอย่างแบบมีชั้นภูมิ

ตารางที่ 4.7 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลควบคุมที่หน้าต่าง 10 วินาที

อัตราการกราดตรวจ(cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)
10	-0.44	-0.42	-0.41	-0.36	-0.31
20	-0.24	-0.24	-0.23	-0.22	-0.19
30	-0.16	-0.16	-0.15	-0.15	-0.14
40	-0.12	-0.11	-0.12	-0.12	-0.10
50	-0.09	-0.10	-0.10	-0.09	-0.09

ตารางที่ 4.7 (ต่อ) อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลควบคุมที่หน้าต่าง 10 วินาที

อัตราการกราตรวจ(cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)
60	-0.08	-0.08	-0.08	-0.08	-0.06
70	-0.07	-0.07	-0.07	-0.06	-0.06
80	-0.06	-0.06	-0.06	-0.06	-0.05
90	-0.05	-0.05	-0.06	-0.05	-0.05
100	-0.05	-0.05	-0.05	-0.05	-0.04

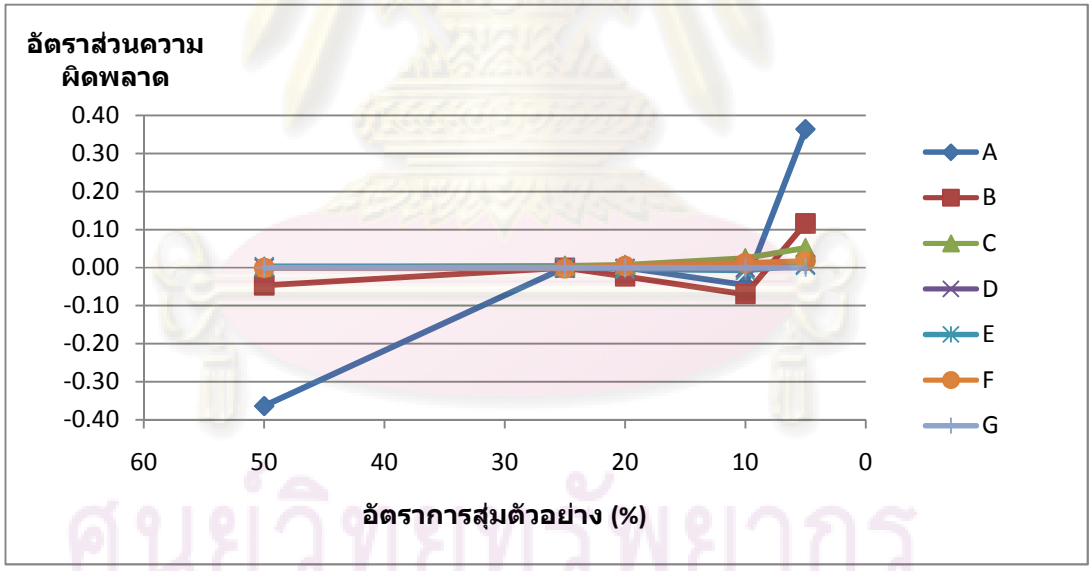


รูปที่ 4.13 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลควบคุมที่หน้าต่าง 10 วินาที

เมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลจริงพบว่ามีอัตราส่วนความผิดพลาดดังตารางที่ 4.8 และรูปที่ 4.14 พบว่ามีอัตราส่วนความผิดพลาดน้อยมาก

ตารางที่ 4.8 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลจริงที่หน้าต่าง 10 วินาที

หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)
A	27.62	-0.36	0.00	0.00	-0.05	0.36
B	29.75	-0.05	0.00	-0.02	-0.07	0.12
C	41.19	0.00	0.00	0.01	0.02	0.05
D	84.98	0.00	0.00	0.00	0.00	0.01
E	87.73	0.00	0.00	0.00	-0.01	0.01
F	88.90	0.00	0.00	0.00	0.01	0.02
G	407.30	0.00	0.00	0.00	0.00	0.00



รูปที่ 4.14 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลจริงที่หน้าต่าง 10 วินาที

นอกจากนี้ยังจำนวนโฮสต์ที่เป็นผลบวกลวงจะมีจำนวนน้อยมากเมื่อเทียบกับการสุ่มตัวอย่างแบบมีชั้นภูมิในตารางที่ 4.5 โดยจำนวนโอสต์ที่เป็นผลบวกลวงนี้แสดงดังในตารางที่ 4.9

ตารางที่ 4.9 จำนวนโฮสต์ที่เป็นผลบวกลวงเมื่อสุ่มตัวอย่างการเชื่อมต่อ

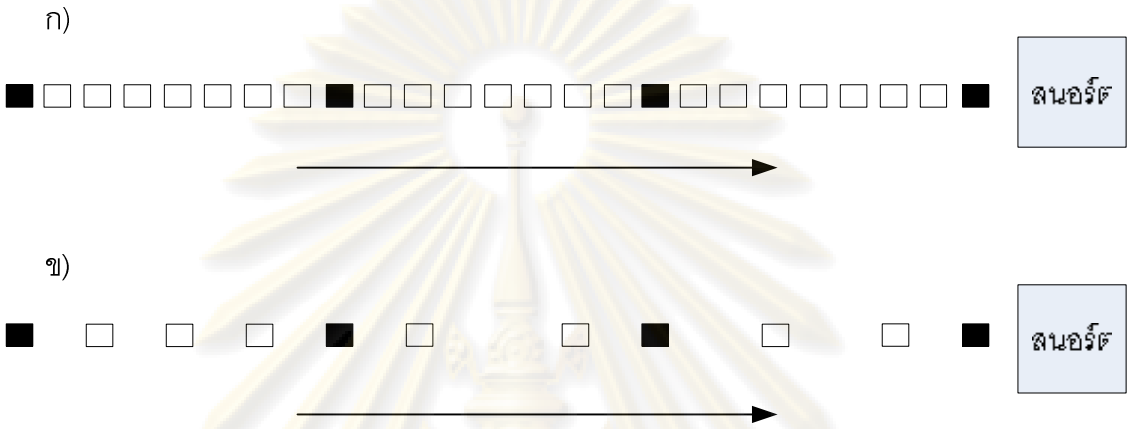
หน้าต่าง	สุ่ม 100 % (1 เอา 1)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
10	0	0	0	1	1	2	-	-
30	0	0	1	0	0	0	2	5
50	0	0	1	0	0	0	2	5
70	0	0	0	0	0	0	0	5
100	0	0	1	1	1	0	1	2

ผู้วิจัยกำหนดอัตราส่วนความผิดพลาดที่เกิดขึ้นมีค่าไม่เกิน 5 % (0.05) (ดังที่กล่าวใน 4.3.3.1) ดังนั้นจากตารางที่ 4.5 หนอนที่ถูกตรวจจับได้ต้องมีอัตราการกราดตรวจที่มากกว่า 90 การเชื่อมต่อต่อวินาที ผลของการตรวจจับข้อมูลจริงของสนอร์ตในตารางที่ 4.8 พบการเตือนของหมายเลขไอพี G มีอัตราการกราดตรวจ 407 การเชื่อมต่อต่อวินาที ซึ่งผลการทดลองที่ได้อาจมีค่าต่างจากที่คำนวณเนื่องจากหนอนในเครือข่ายจริงมีอัตราการกราดตรวจไม่คงที่

จากการทดลองที่ผ่านมาในบทนี้สรุปได้ว่าเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลจริง จะเกิดผลบวกลวงจำนวนมากซึ่งผลการทดลองเป็นไปตามสมมติฐานใน 3.3.2 ดังนั้นจึงแก้ไขโดยเปลี่ยนไปใช้การสุ่มตัวอย่างการเชื่อมต่อ เราสามารถประมาณค่าอัตราส่วนความผิดพลาดได้จากการคำนวณในหัวข้อ 3.3.1 ซึ่งผลการทดลองกับข้อมูลควบคุมพบว่าใกล้เคียงกับที่คำนวณ ในงานวิจัยนี้กำหนดให้อัตราส่วนความผิดพลาดมีค่าไม่เกิน 0.05 ดังนั้นหนอนที่ตรวจจับได้ต้องมีการเตือนมากกว่า 20 ครั้ง และมีอัตราการกราดตรวจตั้งแต่ 90 การเชื่อมต่อต่อวินาทีขึ้นไป ตัวอย่างหนอนที่สามารถตรวจจับได้ เช่น Nimda, Code Red, และ Slammer จากทดลองตรวจจับหนอนในเครือข่ายคณะใหญ่ ในสถาบันอุดมศึกษาขนาด 40,000 คน แห่งหนึ่งสามารถตรวจจับหมายเลขไอพี G ซึ่งมีอัตราการกราดตรวจ 407 การเชื่อมต่อต่อวินาที

ในงานวิจัยนี้สนใจเฉพาะอัตราการกราดตรวจของหนอนที่ตรวจจับได้ด้วยสนอร์ต โดยการตรวจจับนี้ไม่ขึ้นอยู่กับความเร็วหรือปริมาณข้อมูลที่ส่งผ่านเครือข่าย ยกตัวอย่างเช่น จากรูปที่ 4.15 ก) และ 4.15 ข) ให้กล่องสีดำคือหนอน และกล่องสีขาวคือแพ็กเก็ตปกติ ถ้าเครือข่ายมีความเร็วสูงขึ้นจำนวนแพ็กเก็ตการกราดตรวจของหนอนเจือจางลงอย่างมากแต่หนอนที่ตรวจจับที่สนอร์ตยังคงมีอัตราการกราดตรวจเท่าเดิม (4 การเชื่อมต่อ) จะสามารถตรวจจับหนอนโดยใช้ ค่าขีดเริ่มเปลี่ยน และหน้าต่างเวลาเดิมได้ แต่ถ้าเครือข่ายมีความเร็วสูงนี้ขึ้นทำให้หนอนกราดตรวจ

สูงขึ้น ต้องวิเคราะห์และปรับเปลี่ยนตัวแปรที่กำหนดในสนอร์ตใหม่คือ ค่าขีดเริ่มเปลี่ยนและหน้าต่างเวลาใหม่ ซึ่งในงานวิจัยนี้เริ่มใช้หนอนในเครือข่ายที่มีความเร็ว 1Gbps



รูปที่ 4.15 จำนวนแพ็กเก็ตการเชื่อมต่อของหนอนต่อหน่วยเวลาที่เครือข่ายความเร็วต่างกัน

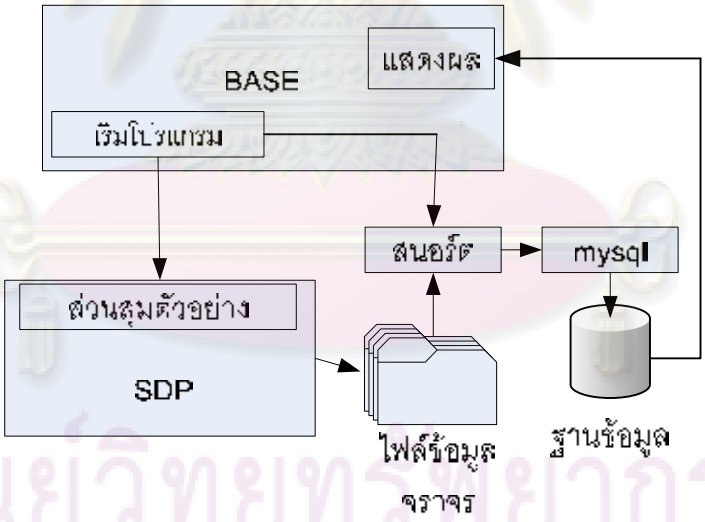
บทที่ 5

การประยุกต์ใช้งาน

จากการทดลองที่ผ่านมา แสดงให้เห็นว่าเมื่อใช้การสุ่มตัวอย่าง สนอร์ตยังคงตรวจพบการกราดตรวจของหนอนได้ แต่การเก็บข้อมูลด้วยเอสดีพี (SDP – Scanning Detection Program) และการตรวจจับการกราดตรวจของหนอนยังเป็นโปรแกรมที่แยกกันและการเริ่มโปรแกรมทั้งเอสดีพี และ สนอร์ต ต้องทำผ่านคอมมานไลน์ (command line) ทำให้ยากต่อการใช้งาน ดังนั้นในบทนี้จะกล่าวถึงการนำโปรแกรมต่างๆ มารวมเป็นระบบเดียวกัน เพื่อให้ผู้ดูแลระบบสามารถสั่งงานได้จากหน้าเว็บที่เดียว รายละเอียดในการพัฒนาส่วนต่างๆ และตัวอย่างการใช้งานระบบมีดังต่อไปนี้

5.1 องค์ประกอบของระบบจัดเก็บและตรวจสอบข้อมูล

องค์ประกอบของระบบระบบจัดเก็บและตรวจสอบข้อมูลจราจรแสดงได้ดังรูปที่ 5.1



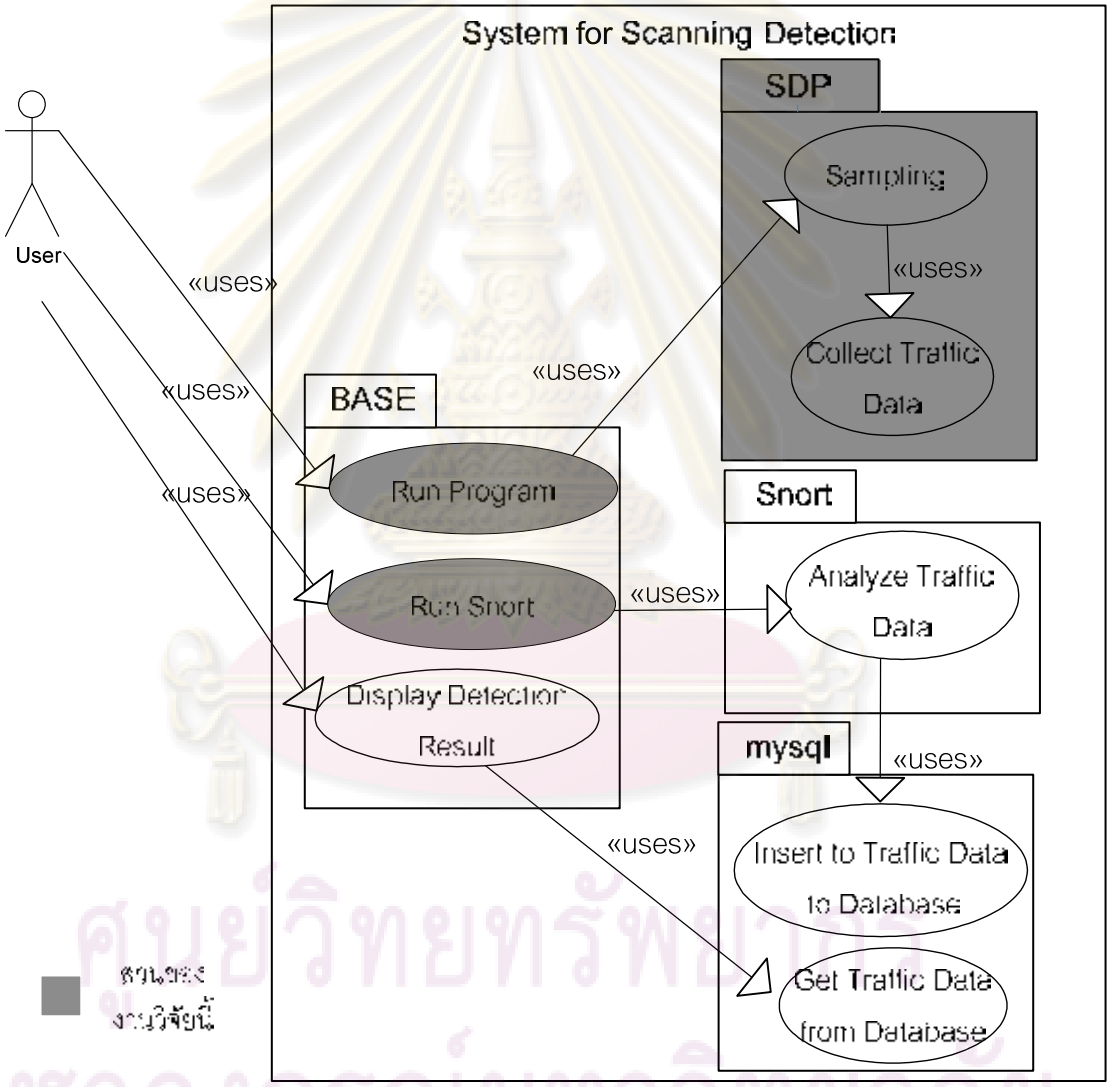
รูปที่ 5.1 องค์ประกอบของระบบจัดเก็บและตรวจสอบข้อมูลจราจร

จากรูปที่ 5.1 ระบบประกอบด้วยส่วนประกอบหลัก 4 ส่วน ได้แก่ หน้าเว็บ โปรแกรมสนอร์ต โปรแกรม SDP และ โปรแกรม mysql โดยแก้ไขรหัสโปรแกรม SDP ให้สามารถสุ่มตัวอย่างได้ด้วยค่าที่เรากำหนด และเขียนสคริปต์เพิ่มเติมสำหรับเก็บข้อมูลจราจรโดยวิธีวนรอบ (Round Robin) นอกจากนี้ผู้วิจัยได้เพิ่มส่วนการเริ่มโปรแกรมและหยุดโปรแกรมจากหน้าเว็บหลัก ทั้ง

โปรแกรม SDP สำหรับสุ่มตัวอย่างและเก็บข้อมูลจราจร และโปรแกรมสนอร์ตสำหรับตรวจจับการกราดตรวจของหนอน) สาเหตุที่ใช้เบส เป็นเว็บสำหรับแสดงผลการตรวจจับของสนอร์ตเพราะเป็นซอร์ฟแวร์แบบโอเพนซอร์ส

5.2 การทำงานของระบบโดยรวม

การทำงานในภาพรวมของระบบแสดงได้ดังรูปที่ 5.2



รูปที่ 5.2 การทำงานของระบบโดยรวม

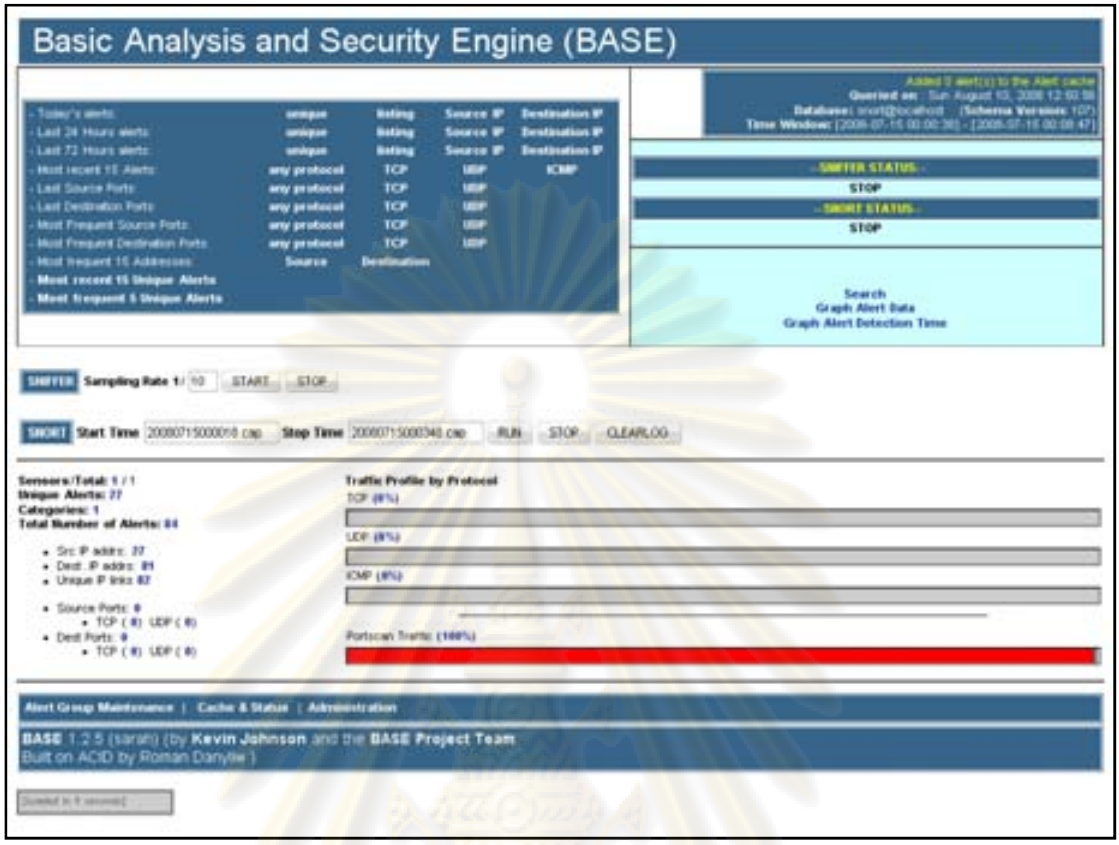
จากรูปที่ 5.2 เมื่อผู้ใช้ติดต่อระบบผ่านหน้าเว็บและต้องการเก็บข้อมูลจราจร ผู้ใช้สามารถเริ่มโปรแกรมจากหน้าเว็บ โดยโปรแกรมจะสุ่มตัวอย่างตามค่าที่ได้เลือกไว้จากหน้าเว็บเพื่อเก็บข้อมูลจราจร ซึ่งสามารถปรับเปลี่ยนอัตราการสุ่มตัวอย่างได้ตั้งแต่ 100% ถึง 1% ระบบจะ

เก็บข้อมูลจราจรโดยวิธีวนรอบ (Round Robin) เมื่อเนื้อที่เก็บข้อมูลเต็มโปรแกรมจะเก็บข้อมูลทับข้อมูลที่ถูกเก็บไว้นานที่สุด ส่วนสนอร์ตทำหน้าที่วิเคราะห์ข้อมูลแล้วเก็บผลการวิเคราะห์ลงในฐานข้อมูลด้วยโปรแกรม Mysql เมื่อผู้ใช้ต้องการดูผลการวิเคราะห์ ให้ใช้โปรแกรมเบสในการดึงข้อมูลจากฐานข้อมูลใน Mysql มาแสดงผ่านหน้าเว็บ

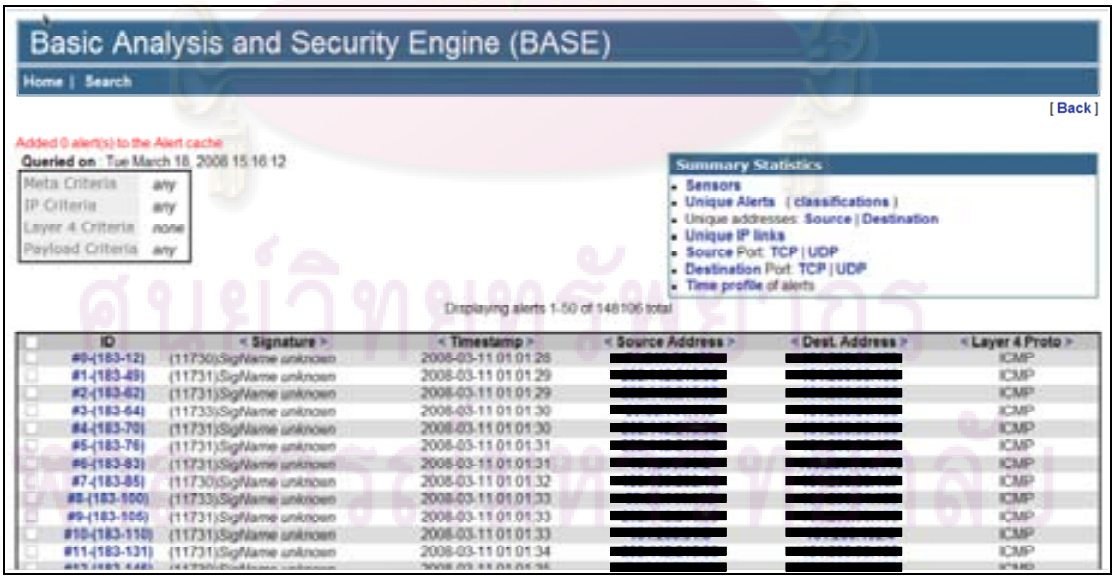
5.3 เว็บปรับแต่งค่าและแสดงผล

เริ่มต้นจะเป็นหน้าเว็บไซต์สำหรับเข้าระบบโดยจะให้ใส่ชื่อผู้ใช้และรหัสผ่าน ดังรูปที่ 5.3 หลังจากใส่ชื่อผู้ใช้และรหัสผ่านก็จะเข้าหน้าหลักแสดงผลสรุปของการตรวจจับความผิดปกติต่างๆ และแสดงส่วนเพิ่มเติมที่ผู้วิจัยได้พัฒนาขึ้นคือ การเริ่มโปรแกรม SDP และสนอร์ตสำหรับเก็บข้อมูลจราจร ซึ่งสามารถกำหนดอัตราการสุ่มตัวอย่างได้ และการตรวจจับการกราดตรวจของหนอนจากข้อมูลจราจรที่เก็บได้ด้วยโปรแกรมสนอร์ต นอกจากนี้ยังสามารถแสดงสถานะได้ว่าโปรแกรม SDP และสนอร์ตทำงานอยู่หรือไม่ โดยแสดงดังรูปที่ 5.4 เมื่อเข้าไปดูรายละเอียดของผลการตรวจจับ เบสจะแสดงรายละเอียดคือหมายเลขแสดงชนิดของการตรวจจับ ชื่อของการตรวจจับ เวลา หมายเลขไอพีต้นทาง หมายเลขไอพีปลายทาง และ โปรโตคอล ดังรูปที่ 5.5

รูปที่ 5.3 หน้าเว็บสำหรับลงบันทึกเข้า



รูปที่ 5.4 หน้าเว็บหลัก



รูปที่ 5.5 หน้าเว็บแสดงรายละเอียดผลการตรวจจับ

5.4 สรุปผลการประยุกต์ใช้งาน

ในบทนี้กล่าวถึงการพัฒนาระบบจัดเก็บและตรวจสอบข้อมูลจราจรให้สามารถใช้งานได้จริง โดยรวมส่วนต่างๆ ให้เป็นระบบเดียวกันและทำงานสัมพันธ์กัน เพื่อความสะดวกของผู้ใช้งาน โดยสามารถสั่งการผ่านหน้าเว็บได้ ทั้งการเก็บข้อมูลจราจร การสุ่มตัวอย่าง และการตรวจจับการกราดตรวจของหนอน

บทที่ 6

สรุปผลการวิจัยและข้อเสนอแนะ

6.1 สรุปผลการวิจัย

เนื่องจากความสามารถในการเก็บข้อมูลที่มีจำกัดและแนวโน้มการใช้เครือข่ายที่เพิ่มขึ้นเรื่อยๆ งานวิจัยนี้ใช้วิธีการสุ่มตัวอย่างทำให้เก็บข้อมูลสำหรับการวิเคราะห์เพื่อตรวจจับหนอนได้เป็นระยะเวลานานขึ้นเพราะจำนวนแพ็กเก็ตลดน้อยลง แต่การสุ่มตัวอย่างยังคงมีความผิดพลาดทั้งผลบวกลวงและผลลบลวง งานวิจัยนี้ได้แก้ปัญหาผลบวกลวงโดยเสนอการสุ่มตัวอย่างการเชื่อมต่อ และสามารถคำนวณอัตราการกราดตรวจของหนอนและอัตราการสุ่ม ที่ทำให้สามารถตรวจจับแพ็กเก็ตการกราดตรวจของหนอนได้อย่างมีประสิทธิภาพ (อัตราส่วนความผิดพลาดน้อย) โดยผลจากการทดลองในงานวิจัยนี้ พบว่าเมื่อให้จำนวนการเชื่อมต่อต่ำสุดที่สนอร์ตสามารถตรวจจับได้ (c_{base}) คือ 10 และอัตราส่วนความผิดพลาด (E) ไม่เกิน 0.05 ที่หน้าต่าง (t) 10 วินาที สามารถตรวจจับหนอนที่มีอัตราการกราดตรวจตั้งแต่ 90 การเชื่อมต่อต่อวินาที ได้ ด้วยอัตราการสุ่มตัวอย่าง 50 % ถึง 5 % ที่หน้าต่าง 30 วินาที สามารถตรวจจับหนอนที่มีอัตราการกราดตรวจตั้งแต่ 90 การเชื่อมต่อต่อวินาที ได้ด้วยอัตราการสุ่มตัวอย่าง 50 % ถึง 2 % และที่หน้าต่าง 50 70 และ 100 วินาที สามารถตรวจจับหนอนที่มีอัตราการกราดตรวจตั้งแต่ 90 การเชื่อมต่อต่อวินาที ได้ด้วยอัตราการสุ่มตัวอย่าง 50 % ถึง 1 %

การประยุกต์ใช้งานทำโดยเพิ่มการสุ่มตัวอย่างในโปรแกรม SDP ให้สามารถเก็บข้อมูลด้วยการสุ่มตัวอย่าง ส่วนการสร้างระบบจัดเก็บข้อมูลจราจรและวิเคราะห์ ผู้ใช้เพิ่มส่วนของการเริ่มทำงาน และหยุดทำงาน ของโปรแกรม SDP และโปรแกรมสนอร์ต บนหน้าเว็บ อีกทั้งยังสามารถปรับอัตราการสุ่มตัวอย่าง จากหน้าเว็บได้อีกด้วย

6.2 ปัญหาที่พบจากการวิจัย

ปัญหาหรือข้อจำกัดที่พบในงานวิจัยมีดังนี้

1. การจำลองข้อมูลหนอนมีความยากลำบากเนื่องจากเป็นข้อมูลอันตรายและอาจก่อให้เกิดผลเสียหายแก่เครือข่ายได้ทำให้ไม่สามารถนำหนอนอินเทอร์เน็ตไปทดลองปล่อยบนเครือข่ายจริงได้

2. รหัสต้นฉบับ (source code) ของสนอร์ตมีขนาดใหญ่และมีความซับซ้อนสูงมากทำให้ต้องใช้เวลานานในการศึกษาการทำงานของพรีโพรเซสเซอร์ flow-portscan เพื่อเพิ่มการสุ่มตัวอย่างลงไปในสนอร์ต

3. ใช้เวลานานในการทดลองเนื่องจากข้อมูลจราจรที่นำมาวิเคราะห์มีขนาดใหญ่ระดับกิกะไบต์ และทดลองหลายรอบ (ทดลองประมาณ 148 รอบ) โดยทดสอบกับข้อมูลทดลองใช้เวลาประมาณ 10 วัน และทดสอบกับข้อมูลจริง ใช้เวลาประมาณ 21 วัน

4. ต้องใช้เนื้อที่ในการเก็บข้อมูลจำนวนมากเพราะต้องเก็บทั้งข้อมูลจราจรและข้อมูลจราจรที่ถูกสุ่มเพื่อทดลองเปรียบเทียบผลการทดลอง โดยผู้วิจัยได้จัดหาจานบันทึกแบบแข็ง (hard disk) เพิ่มเติมสำหรับเก็บข้อมูล จำนวน 300 กิกะไบต์

5. สภาพแวดล้อมของข้อมูลควบคุมกับข้อมูลจริงมีความแตกต่างกัน ทั้งอัตราการกราดตรวจของหนอน จำนวนแพ็กเก็ต และตัวแปรอื่นๆที่ไม่สามารถคาดเดาได้ ทำให้ผลการทดลองของข้อมูลจริงมีความคลาดเคลื่อนพอสมควร

6.3 ข้อเสนอแนะ

สิ่งที่ควรพัฒนาต่อจากงานวิจัยชิ้นนี้

1. ศึกษาผลกระทบของการสุ่มตัวอย่างต่อการบุกรุกแบบอื่นๆ เช่น พฤติกรรมการสแกนพอร์ต หนอนที่แพร่กระจายผ่านโปรแกรมเพียร์ทูเพียร์ (Peer-to-Peer) หรือ การโจมตีแบบปฏิเสธการให้บริการ (Denial of Service)

2. ศึกษาอัตราการสุ่มตัวอย่างแบบอื่นทั้งการสุ่มแบบง่ายและการสุ่มแบบมีระบบ ที่มีต่อการกราดตรวจของหนอน

3. ศึกษาลักษณะการทำงานของเครือข่าย หรือลักษณะของโปรแกรมเครือข่ายของการสุ่มตัวอย่าง เพื่อเพิ่มประสิทธิภาพในการสุ่มที่ดีขึ้น เช่น สุ่มตามขนาดของโฟลว์ หรือปรับอัตราการสุ่มตามลักษณะเฉพาะของการใช้งานเครือข่ายนั้น

4. หาวิธีการการตรวจจับพฤติกรรมหนอนแบบอื่นนอกเหนือจากการตรวจจับการกราดตรวจ มาประยุกต์ใช้กับการสุ่มตัวอย่าง

5. พัฒนาการจำลองของข้อมูลจราจร [27] ที่มีการบุกรุกแบบต่างๆ เพื่อเป็นข้อมูลมาตรฐานสำหรับใช้ในงานวิจัยด้านความปลอดภัยเครือข่ายต่อไป

รายการอ้างอิง

[1] Cabani, A., Ramaswamy, S., Itmi, M., Al-Shukri, S., and Pécuchet, J. P. (2007). Distributed Computing Systems : P2P versus Grid Computing Alternatives. Innovation and Advanced Technique in Computer and Information Science and Engineering, pp. 47-52.

[2] Pongpaibool, P. (2006). Characteristics of Internet Traffic in Thailand [Online]. Avaliable from: http://internet.nectec.or.th/document/pdf/20060329ECTI2006_panita.pdf[2009, February 15]

[3] Claffy, K. C., Polyzos, G. C., and Braun, H. (1993). Application of Sampling Methodologies to Network Traffic Characterization. Proceedings of ACM SIGCOMM Computer Communication Review, pp. 194-203.

[4] Snort. (2008). Snort™ Users Manual [Online]. Available from: http://www.snort.org/docs/snort_htmanuals/htmanual_2832/[2009, February 15]

[5] Arboleda, A. F., and Bedón, Ch. E. (2005). Snort diagrams for developers [Online]. Available from: http://afrodita.unicauca.edu.co/~cbedon/snort/snortdevdiagrams.html[2009, February 15]

[6] BASE. (2008). Basic Analysis and Security Engine (BASE) project [Online]. Available from: http://base.secureideas.net/about.php[2009, February 15]

[7] Buchholz, F., et al. (2002). Digging for Worms, Fishing for Answers. Proceedings of the 18th Annual Computer Security Applications Conference, pp. 219-226.

[8] Lee, M., et al. (2007). An Approach for Classifying Internet Worms Based on Temporal Behaviors and Packet Flows. Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues, pp. 646-655.

[9] Williamson, M. M. (2002). Throttling Viruses: Restricting Propagation to Defeat Malicious Mobile Code. Proceedings of the 18th Computer Security Applications Conference, pp. 61-68.

[10] Twycross, J., and Williamson, M. M. (2003). Implementing and Testing a Virus Throttle. Proceedings of the 12th USENIX Security Symposium, pp. 285–294.

[11] Wong, C., Bielski, S., Studer, A., and Wang, C. (2005). On the Effectiveness of Rate Limiting Mechanisms [Online]. Avaliable from: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.87.3089&rep=rep1&type=pdf[2009, February 15]

[12] Zou, C. C., Gong, W., and Towsley, D. (2003). Worm Propagation Modeling and Analysis Under Dynamic Quarantine Defense. Proceedings of the 2003 ACM Workshop on Rapid Malcode, pp. 51-60.

[13] Sekar, V., Xie Y., Reiter M. K., and Zhang, H. (2006). A Multi-Resolution Approach for Worm Detection Containment. Proceedings of the International Conference on Dependable Systems and Networks, pp. 189-198.

[14] Clegg, R. G., Landa, R., Haddadi, H., Rio, M., and Moore, A. W. (2008). Techniques for flow inversion on sampled data. Computer Communications Workshops, 2008. INFOCOM, pp. 1-6.

[15] Brauckhoff, D., Tellenbach, B., Wagner, A., May, M., and Lakhina, A. (2006). Impact of packet sampling on anomaly detection metrics. Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement, pp. 159–164.

[16] Kawahara, R., Mori, T., Kamiyama, N., Harada, S., and Asano, S. (2007). A Study on Detecting Network Anomalies Using Sampled Flow Statistics. Applications and the Internet Workshops, 2007. SAINT Workshops 2007, pp. 81.

[17] Cardenas, A. A., Baras J. S., and Ramezani, V. (2004). Distributed Change Detection for Worms, DDoS and other Network Attacks. Proceedings of the 2004 American Control Conference, pp. 1008-1013.

[18] Phaal, P., Panchen, S., and McKee, N. (2008). InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks [Online]. Available from: http://www.isi.edu/in-notes/rfc3176.txt[2009, February 15]

[19] Phaal, P., and Panchen, S. (2002). Packet Sampling Basics [Online]. Available from: http://www.sflow.org/packetSamplingBasics/ index.htm[2009, February 15]

[20] Six Sigma. (2008). Table of the Standard Normal (z) Distribution [Online]. Available from: http://www.isixsigma.com/library/content/zdistribution.asp[2009, February 15]

[21] Tcpdump. (2008). Tcpdump Manual Page [Online]. Available from: http://linux.die.net/man/8/tcpdump[2009, February 15]

[22] Lincoln Laboratory. (2008). 1999 DARPA Intrusion Detection Evaluation Data Set [Online]. Available from: http://www.ll.mit.edu/mission/communications/ist/corpora/ideval/data/1999data.html[2009, February 15]

[23] Al-Hammadi, Y., and Leckie, C. (2005). Anomaly Detection for Internet Worms. Proceedings of the 9th IFIP/IEEE International Symposium on Integrated Network Management, pp. 126-133.

[24] Song, G., Sun, Z., and Li, X. (2007). The Research of Association Rules Mining and Application in Intrusion Alerts Analysis. Proceedings of the Second International Conference on Innovative Computing, Informatio and Control, pp. 567-567.

[25] Mori. T., Uchida, M., Kawahara, R., Pan, J., and Goto, S. (2004). Identifying Elephant Flows through Periodically Sampled Packets. Proceedings of The 4th ACM SIGCOMM Conference on Internet Measurement, pp. 115-120.

[26] Venkataraman, S., Song, D., Gibbons, P. B., and Blum, A. (2005). New Streaming Algorithm for Fast Detection of Superspreaders. Proceedings of the 12th ISOC Symposium on Network and Distributed Systems Security (SNDSS), pp. 149-166.

[27] Wireshark. (2009). SampleCaptures [Online]. Available from: http://wiki.wireshark.org/SampleCaptures[2009, February 15]

[28] Lamping, U., Sharp, R., and Warnicke, E. (2006). Wireshark 1.0.6 [Online]. Available from: http://www.wireshark.org/download.html[2009, February 15]

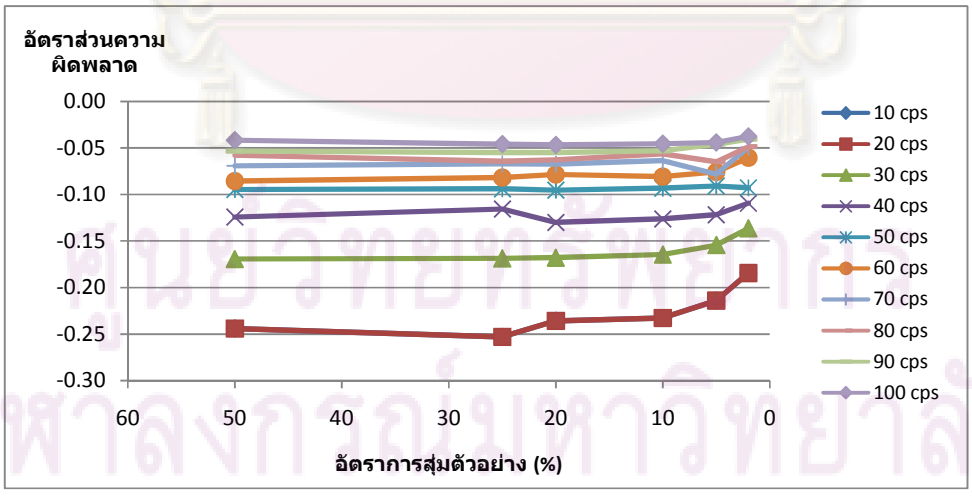
ภาคผนวก

ภาคผนวก ก

ผลการทดลองเพิ่มเติม

ตารางที่ ก.1 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลควบคุมที่หน้าต่าง 30 วินาที

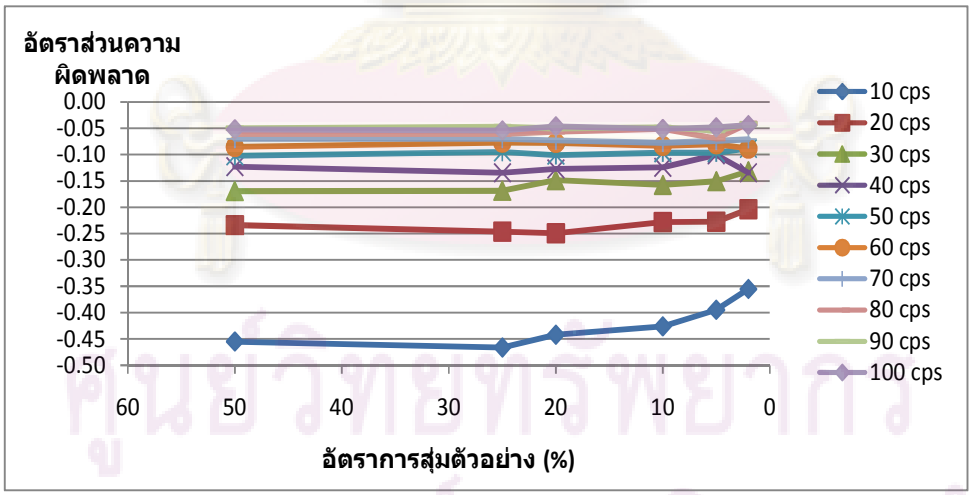
อัตราการกราตรวจ(cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)
10	-0.47	-0.43	-0.45	-0.41	-0.38	-0.31
20	-0.24	-0.25	-0.24	-0.23	-0.21	-0.18
30	-0.17	-0.17	-0.17	-0.16	-0.15	-0.14
40	-0.12	-0.12	-0.13	-0.13	-0.12	-0.11
50	-0.09	-0.09	-0.10	-0.09	-0.09	-0.09
60	-0.09	-0.08	-0.08	-0.08	-0.08	-0.06
70	-0.07	-0.07	-0.07	-0.06	-0.08	-0.05
80	-0.06	-0.06	-0.06	-0.06	-0.06	-0.05
90	-0.05	-0.05	-0.05	-0.05	-0.05	-0.04
100	-0.04	-0.05	-0.05	-0.05	-0.04	-0.04



รูปที่ ก.1 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลควบคุมที่หน้าต่าง 30 วินาที

ตารางที่ ก.2 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิ กับข้อมูลควบคุมที่หน้าต่าง 50 วินาที

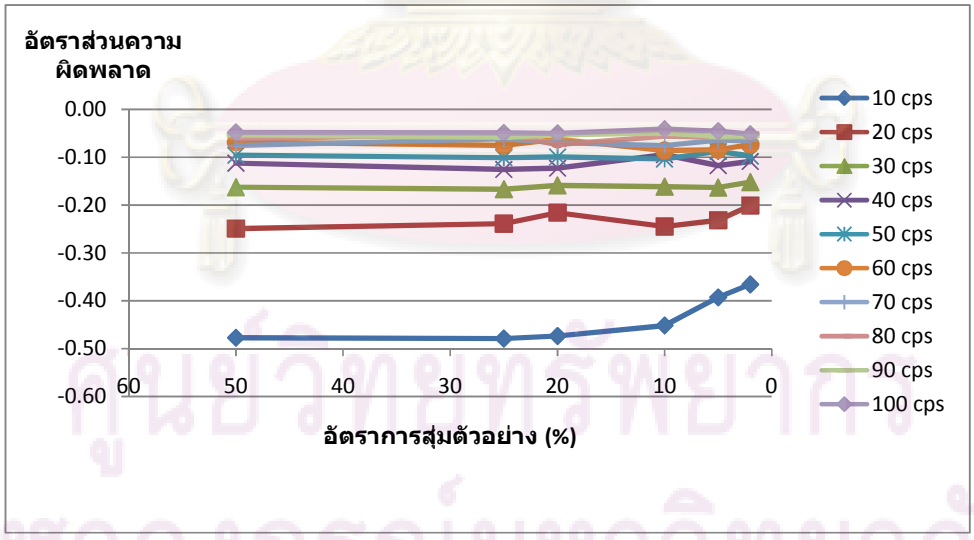
อัตราการกราตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
10	-0.45	-0.47	-0.44	-0.43	-0.39	-0.36	-0.30
20	-0.23	-0.25	-0.25	-0.23	-0.23	-0.20	-0.16
30	-0.17	-0.17	-0.15	-0.16	-0.15	-0.13	-0.13
40	-0.12	-0.13	-0.13	-0.12	-0.10	-0.13	-0.08
50	-0.10	-0.09	-0.10	-0.10	-0.10	-0.09	-0.09
60	-0.09	-0.08	-0.08	-0.08	-0.08	-0.09	-0.04
70	-0.07	-0.07	-0.07	-0.08	-0.07	-0.07	-0.05
80	-0.06	-0.06	-0.06	-0.05	-0.07	-0.04	-0.06
90	-0.05	-0.05	-0.05	-0.05	-0.05	-0.04	-0.05
100	-0.05	-0.05	-0.05	-0.05	-0.05	-0.04	-0.05



รูปที่ ก.2 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิ กับข้อมูลควบคุมที่หน้าต่าง 50 วินาที

ตารางที่ ก.3 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิ
กับข้อมูลควบคุมที่หน้าต่าง 70 วินาที

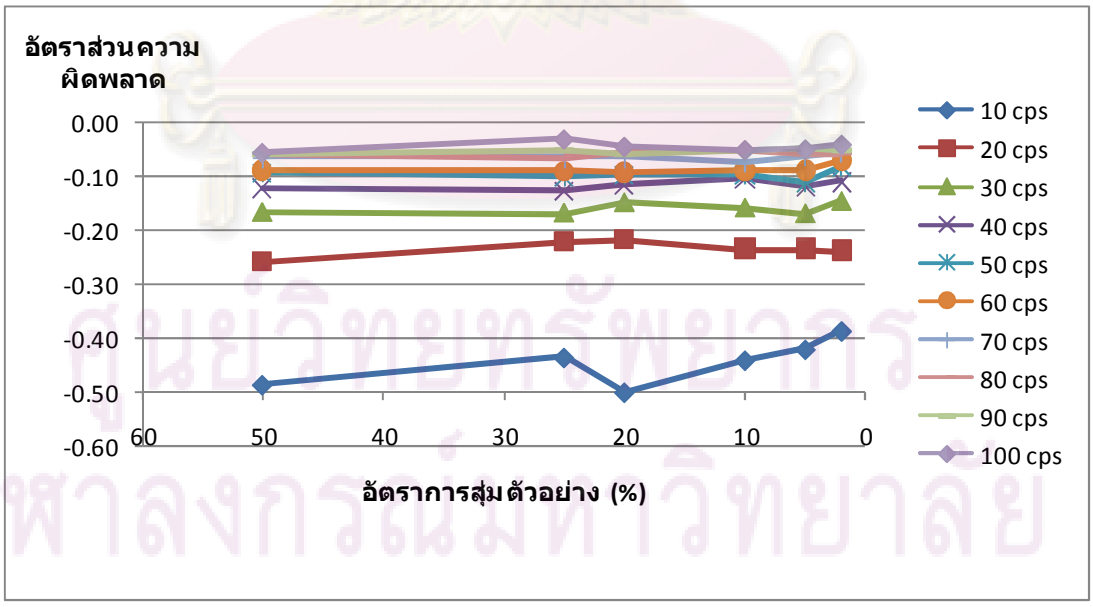
อัตราการกราตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
10	-0.48	-0.48	-0.47	-0.45	-0.39	-0.37	-0.31
20	-0.25	-0.24	-0.22	-0.24	-0.23	-0.20	-0.21
30	-0.16	-0.17	-0.16	-0.16	-0.16	-0.15	-0.12
40	-0.11	-0.13	-0.12	-0.09	-0.12	-0.11	-0.10
50	-0.10	-0.10	-0.10	-0.10	-0.09	-0.10	-0.08
60	-0.07	-0.08	-0.06	-0.09	-0.08	-0.07	-0.06
70	-0.08	-0.06	-0.07	-0.08	-0.06	-0.07	-0.07
80	-0.06	-0.05	-0.07	-0.06	-0.06	-0.05	-0.05
90	-0.05	-0.06	-0.05	-0.05	-0.06	-0.06	-0.05
100	-0.05	-0.05	-0.05	-0.04	-0.05	-0.05	-0.03



รูปที่ ก.3 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิ
กับข้อมูลควบคุมที่หน้าต่าง 70 วินาที

ตารางที่ ก.4 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิ
กับข้อมูลควบคุมที่หน้าต่าง 100 วินาที

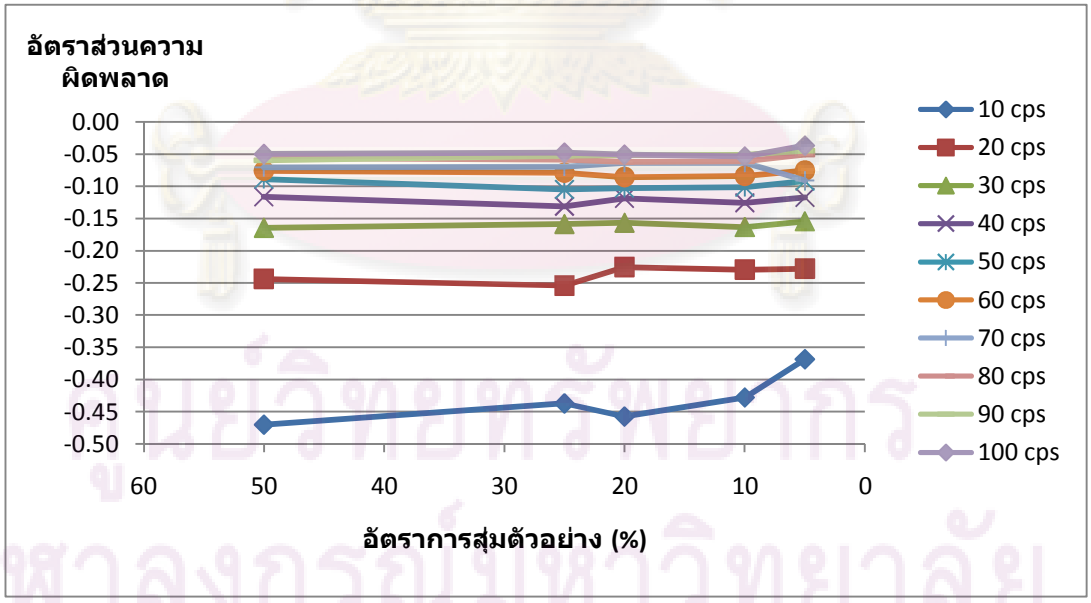
อัตราการกราตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
10	-0.49	-0.44	-0.50	-0.44	-0.42	-0.39	-0.34
20	-0.26	-0.22	-0.22	-0.23	-0.23	-0.24	-0.20
30	-0.17	-0.17	-0.15	-0.16	-0.17	-0.15	-0.17
40	-0.12	-0.13	-0.11	-0.10	-0.12	-0.11	-0.13
50	-0.09	-0.10	-0.10	-0.10	-0.11	-0.08	-0.09
60	-0.09	-0.09	-0.09	-0.09	-0.09	-0.07	-0.06
70	-0.06	-0.06	-0.06	-0.07	-0.06	-0.06	-0.03
80	-0.06	-0.06	-0.05	-0.05	-0.06	-0.06	-0.05
90	-0.06	-0.05	-0.06	-0.05	-0.05	-0.05	-0.08
100	-0.06	-0.03	-0.04	-0.05	-0.05	-0.04	-0.05



รูปที่ ก.4 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิ
กับข้อมูลควบคุมที่หน้าต่าง 100 วินาที

ตารางที่ ก.5 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลควบคุมที่หน้าต่าง 30 วินาที

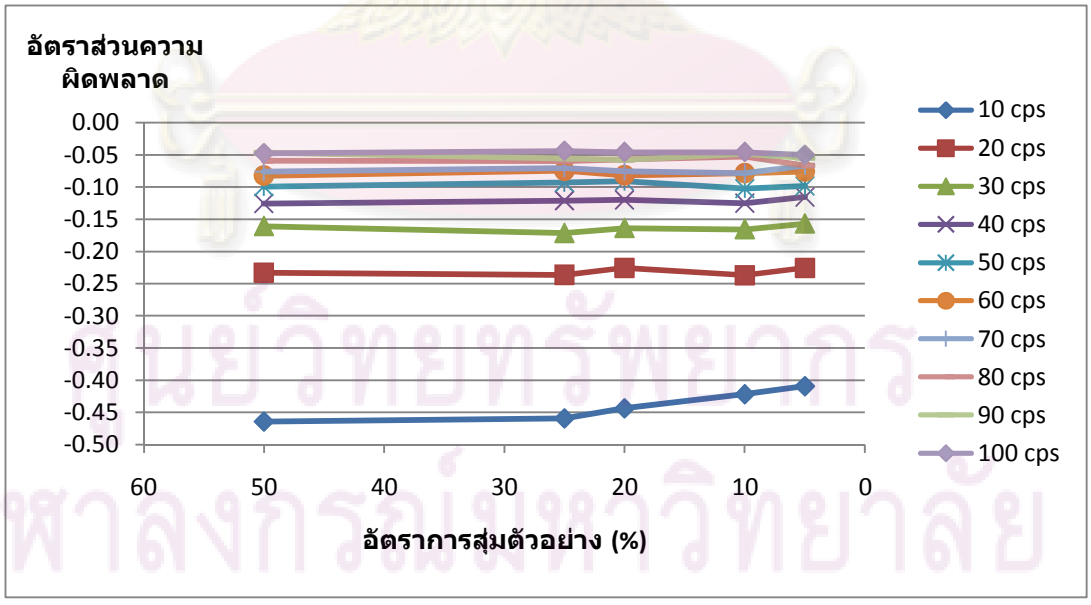
อัตราการกราตรวจ(cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)
10	-0.47	-0.44	-0.46	-0.43	-0.37	-0.32
20	-0.24	-0.25	-0.23	-0.23	-0.23	-0.19
30	-0.16	-0.16	-0.16	-0.16	-0.15	-0.14
40	-0.12	-0.13	-0.12	-0.13	-0.12	-0.10
50	-0.09	-0.10	-0.10	-0.10	-0.09	-0.08
60	-0.08	-0.08	-0.09	-0.08	-0.08	-0.07
70	-0.07	-0.07	-0.06	-0.06	-0.09	-0.06
80	-0.06	-0.06	-0.06	-0.06	-0.05	-0.05
90	-0.06	-0.05	-0.05	-0.05	-0.04	-0.05
100	-0.05	-0.05	-0.05	-0.05	-0.04	-0.05



รูปที่ ก.5 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลควบคุมที่หน้าต่าง 30 วินาที

ตารางที่ ก.6 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อม
ต่อกับข้อมูลควบคุมที่หน้าต่าง 50 วินาที

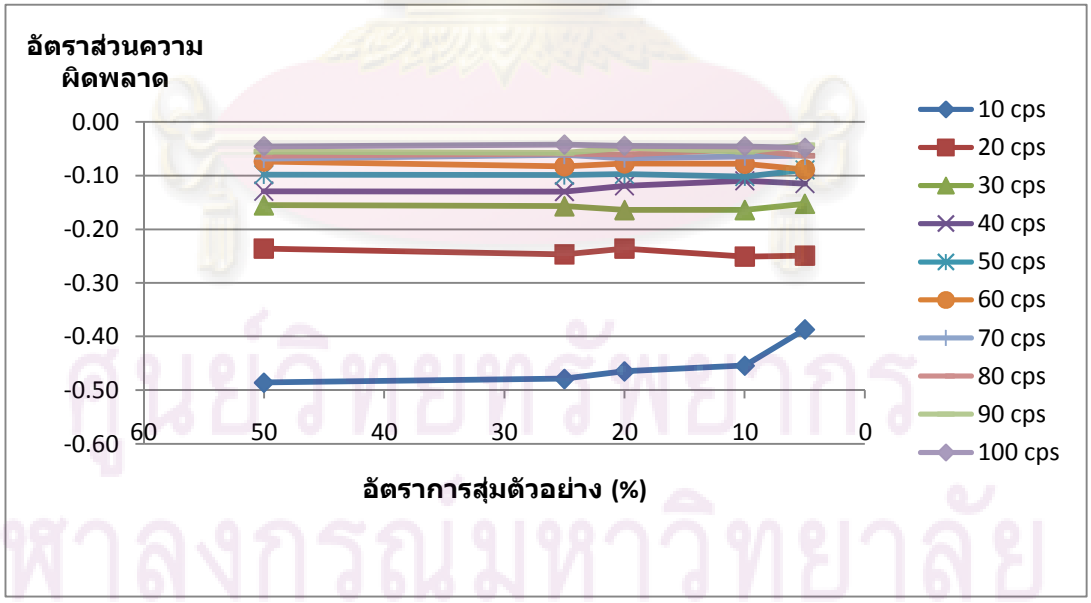
อัตราการกราตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
10	-0.46	-0.46	-0.44	-0.42	-0.41	-0.37	-0.31
20	-0.23	-0.24	-0.23	-0.24	-0.23	-0.22	-0.20
30	-0.16	-0.17	-0.16	-0.17	-0.16	-0.15	-0.15
40	-0.13	-0.12	-0.12	-0.12	-0.12	-0.11	-0.10
50	-0.10	-0.09	-0.09	-0.10	-0.10	-0.10	-0.07
60	-0.08	-0.07	-0.08	-0.08	-0.08	-0.07	-0.07
70	-0.08	-0.07	-0.08	-0.08	-0.07	-0.05	-0.05
80	-0.06	-0.06	-0.06	-0.05	-0.07	-0.06	-0.04
90	-0.05	-0.06	-0.06	-0.05	-0.05	-0.05	-0.04
100	-0.05	-0.04	-0.05	-0.05	-0.05	-0.05	-0.04



รูปที่ ก.6 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อ
กับข้อมูลควบคุมที่หน้าต่าง 50 วินาที

ตารางที่ ก.7 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลควบคุมที่หน้าต่าง 70 วินาที

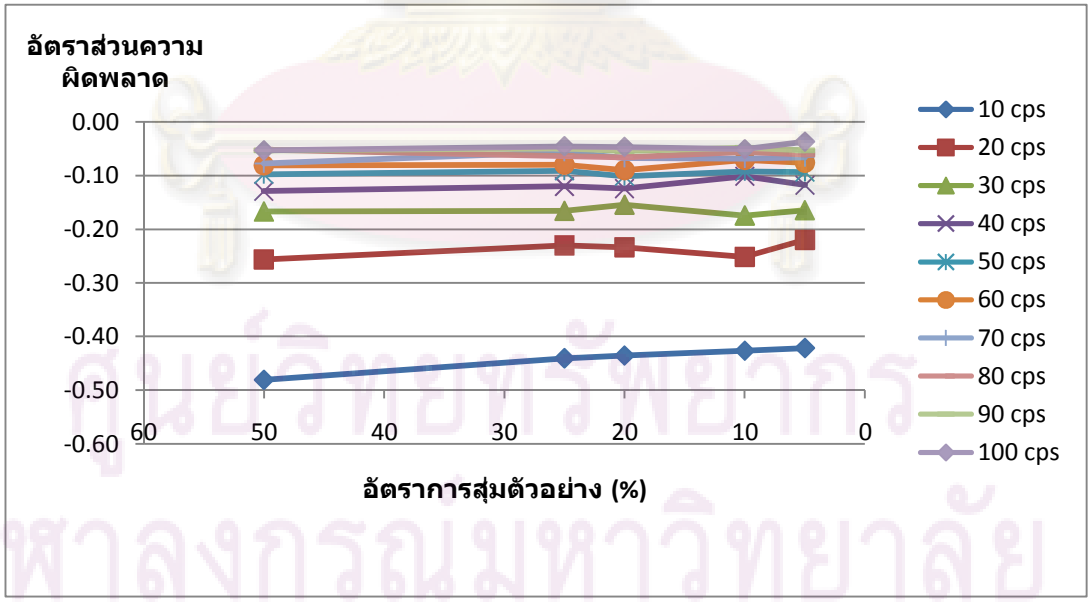
อัตราการกราตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
10	-0.49	-0.48	-0.46	-0.45	-0.39	-0.36	-0.31
20	-0.24	-0.25	-0.24	-0.25	-0.25	-0.22	-0.22
30	-0.16	-0.16	-0.16	-0.16	-0.15	-0.16	-0.12
40	-0.13	-0.13	-0.12	-0.11	-0.11	-0.12	-0.11
50	-0.10	-0.10	-0.10	-0.10	-0.09	-0.09	-0.08
60	-0.07	-0.08	-0.08	-0.08	-0.09	-0.08	-0.05
70	-0.07	-0.06	-0.07	-0.06	-0.06	-0.06	-0.06
80	-0.06	-0.06	-0.06	-0.05	-0.06	-0.04	-0.05
90	-0.06	-0.06	-0.05	-0.06	-0.04	-0.05	-0.06
100	-0.05	-0.04	-0.04	-0.05	-0.05	-0.05	-0.04



รูปที่ ก.7 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลควบคุมที่หน้าต่าง 70 วินาที

ตารางที่ ก.8 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลควบคุมที่หน้าต่าง 100 วินาที

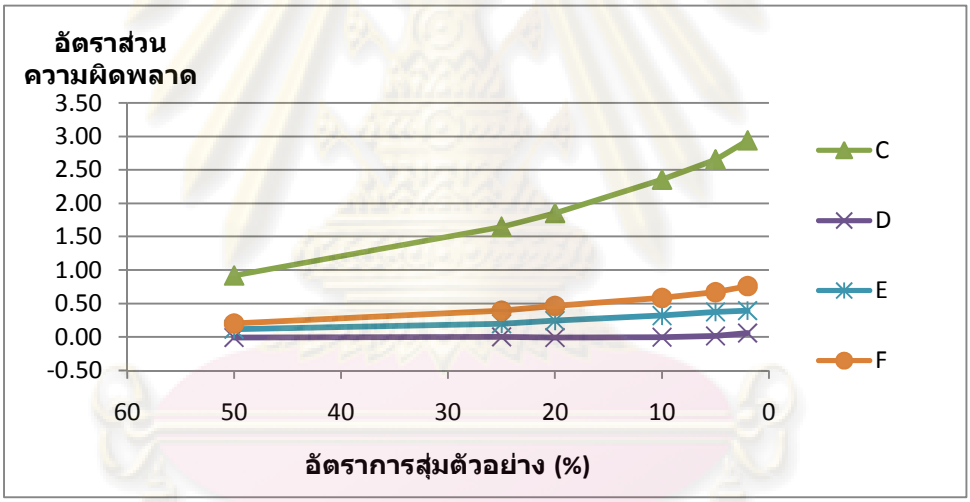
อัตราการกราตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
10	-0.48	-0.44	-0.44	-0.43	-0.42	-0.39	-0.35
20	-0.26	-0.23	-0.23	-0.25	-0.22	-0.25	-0.22
30	-0.17	-0.17	-0.15	-0.17	-0.16	-0.15	-0.14
40	-0.13	-0.12	-0.12	-0.10	-0.12	-0.11	-0.09
50	-0.10	-0.09	-0.10	-0.09	-0.09	-0.09	-0.07
60	-0.08	-0.08	-0.09	-0.07	-0.08	-0.09	-0.05
70	-0.08	-0.06	-0.07	-0.07	-0.07	-0.05	-0.06
80	-0.05	-0.06	-0.07	-0.06	-0.06	-0.05	-0.07
90	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	-0.06
100	-0.05	-0.05	-0.05	-0.05	-0.04	-0.03	-0.04



รูปที่ ก.8 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลควบคุมที่หน้าต่าง 100 วินาที

ตารางที่ ก.9 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลจริงที่หน้าต่าง 30 วินาที

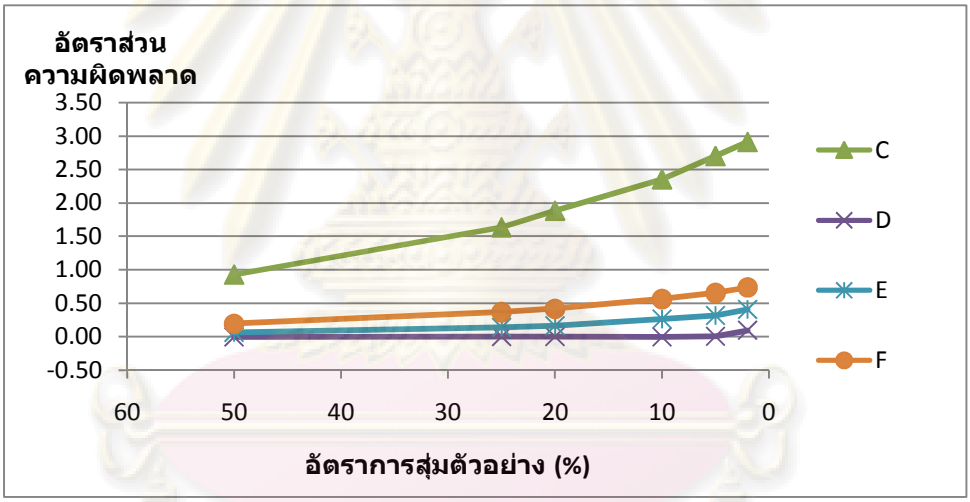
หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)
C	31.74	0.92	1.65	1.85	2.35	2.65	2.94
D	54.17	0.12	0.20	0.24	0.32	0.37	0.39
E	56.59	0.20	0.40	0.47	0.59	0.67	0.76
F	72.67	-0.01	0.00	-0.01	0.00	0.02	0.06



รูปที่ ก.9 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลจริงที่หน้าต่าง 30 วินาที

ตารางที่ ก.10 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลจริงที่หน้าต่าง 50 วินาที

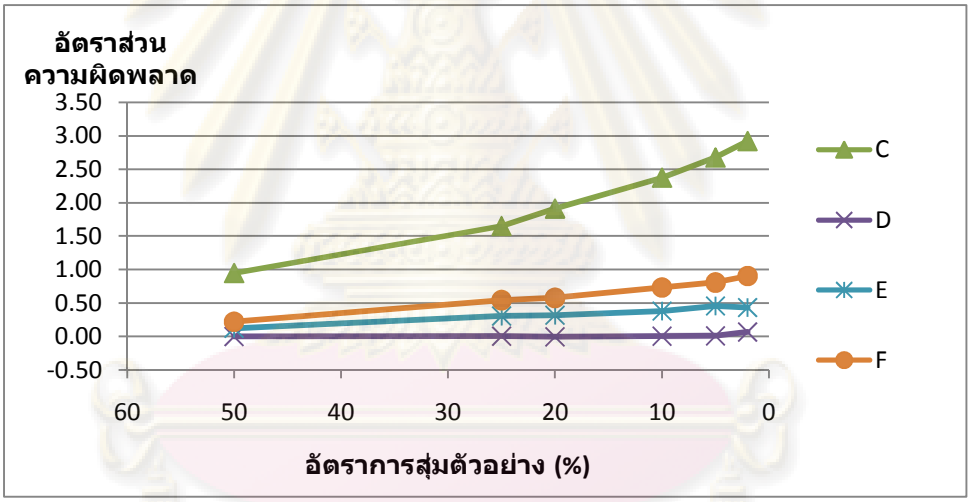
หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
C	29.62	0.93	1.64	1.88	2.35	2.70	2.91	3.03
D	61.81	0.00	0.00	0.00	0.00	0.00	0.09	0.21
E	74.90	0.06	0.14	0.17	0.26	0.32	0.41	0.40
F	72.94	0.19	0.37	0.42	0.56	0.65	0.74	0.77



รูปที่ ก.10 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลจริงที่หน้าต่าง 50 วินาที

ตารางที่ ก.11 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลจริงที่หน้าต่าง 70 วินาที

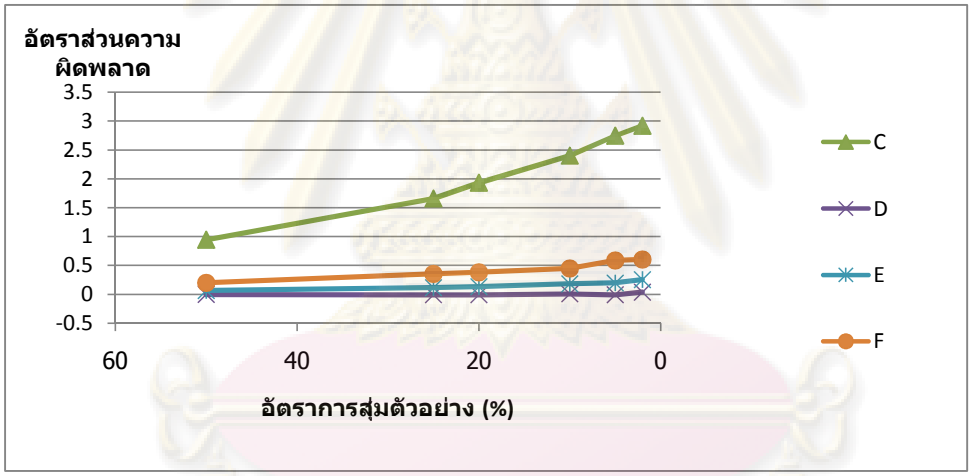
หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
C	28.52	0.95	1.65	1.91	2.38	2.68	2.92	3.06
D	59.47	0.00	0.01	-0.01	0.01	0.01	0.07	0.11
E	57.30	0.12	0.30	0.32	0.38	0.45	0.43	0.50
F	56.48	0.22	0.54	0.58	0.73	0.81	0.90	0.98



รูปที่ ก.11 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลจริงที่หน้าต่าง 70 วินาที

ตารางที่ ก.12 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลจริงที่หน้าต่าง 100 วินาที

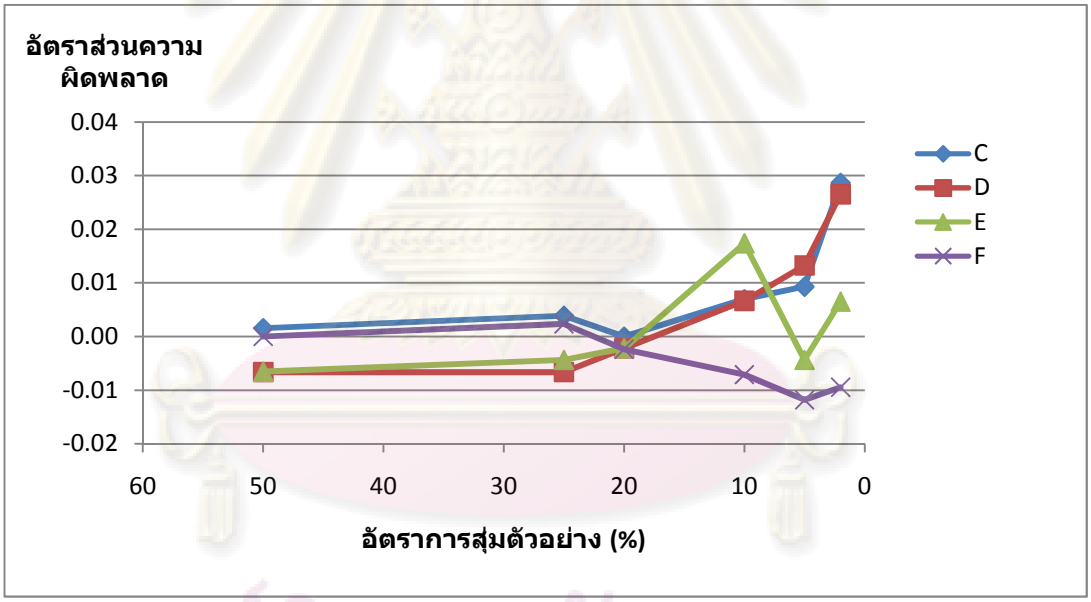
หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
C	27.78	0.94	1.66	1.93	2.40	2.74	2.92	3.06
D	52.40	0.00	-0.01	-0.01	0.01	-0.01	0.04	0.16
E	40.11	0.07	0.12	0.13	0.18	0.20	0.26	0.28
F	39.58	0.20	0.35	0.38	0.45	0.59	0.60	0.72



รูปที่ ก.12 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างแบบมีชั้นภูมิกับข้อมูลจริงที่หน้าต่าง 100 วินาที

ตารางที่ ก.13 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลจริงที่หน้าต่าง 30 วินาที

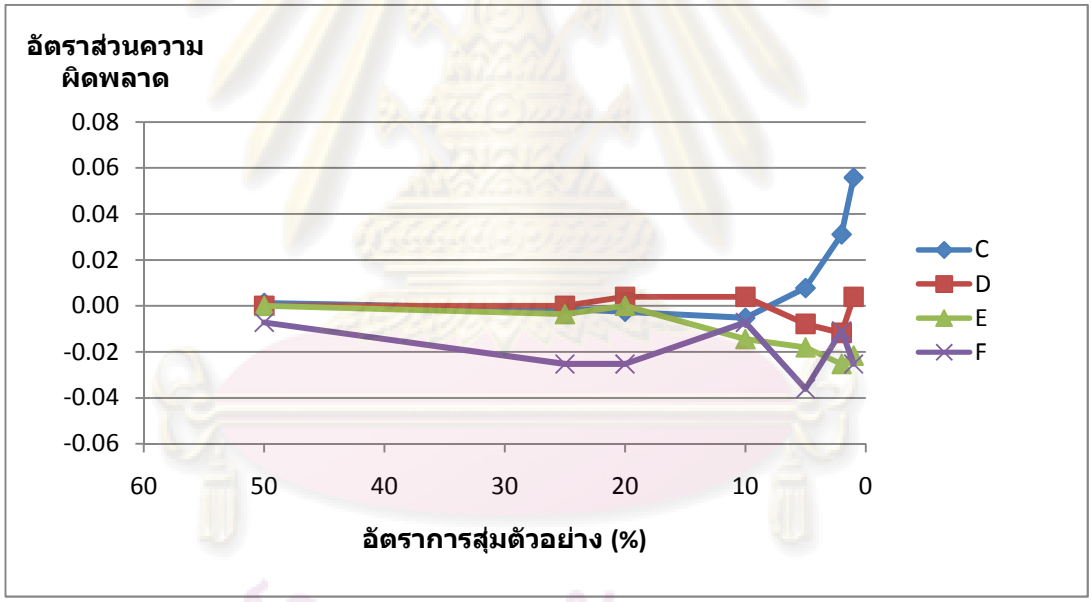
หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)
C	ตัวอย่0.00	0.00	0.00	0.00	0.01	0.01	0.03
D	72.67	-0.01	-0.01	0.00	0.01	0.01	0.03
E	54.17	-0.01	0.00	0.00	0.02	0.00	0.01
F	56.59	0.00	0.00	0.00	0.01	0.01	0.03



รูปที่ ก.13 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลจริงที่หน้าต่าง 30 วินาที

ตารางที่ ก.14 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลจริงที่หน้าต่าง 50 วินาที

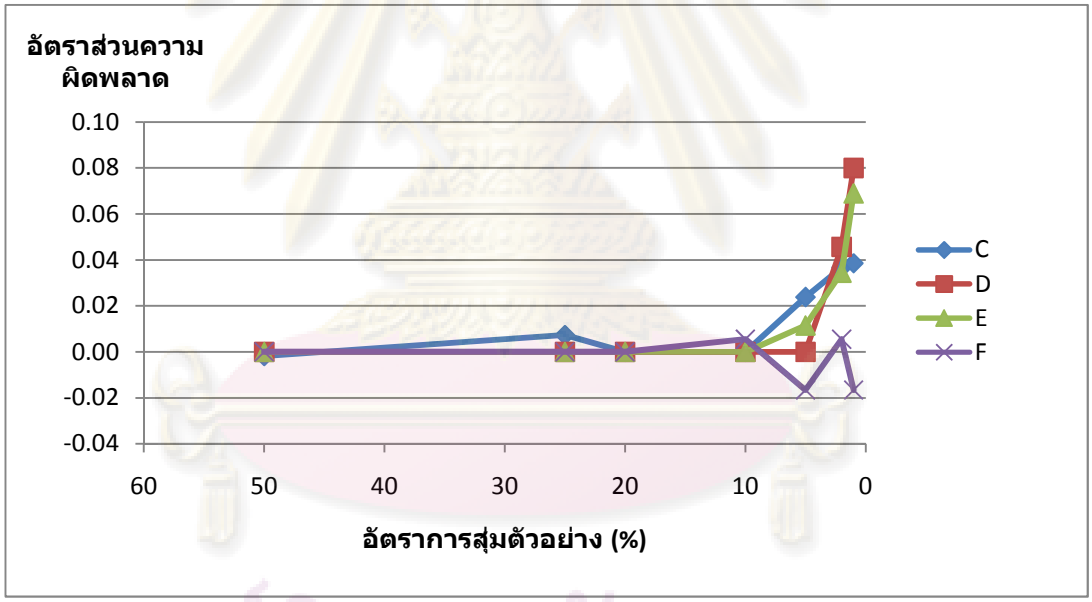
หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
C	29.62	0.00	0.00	0.00	-0.01	0.01	0.03	0.06
D	61.81	0.00	0.00	0.00	0.00	-0.01	-0.01	0.00
E	74.90	0.00	0.00	0.00	-0.01	-0.02	-0.03	-0.02
F	72.94	-0.01	-0.03	-0.03	-0.01	-0.04	-0.01	-0.03



รูปที่ ก.14 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลจริงที่หน้าต่าง 50 วินาที

ตารางที่ ก.15 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลจริงที่หน้าต่าง 70 วินาที

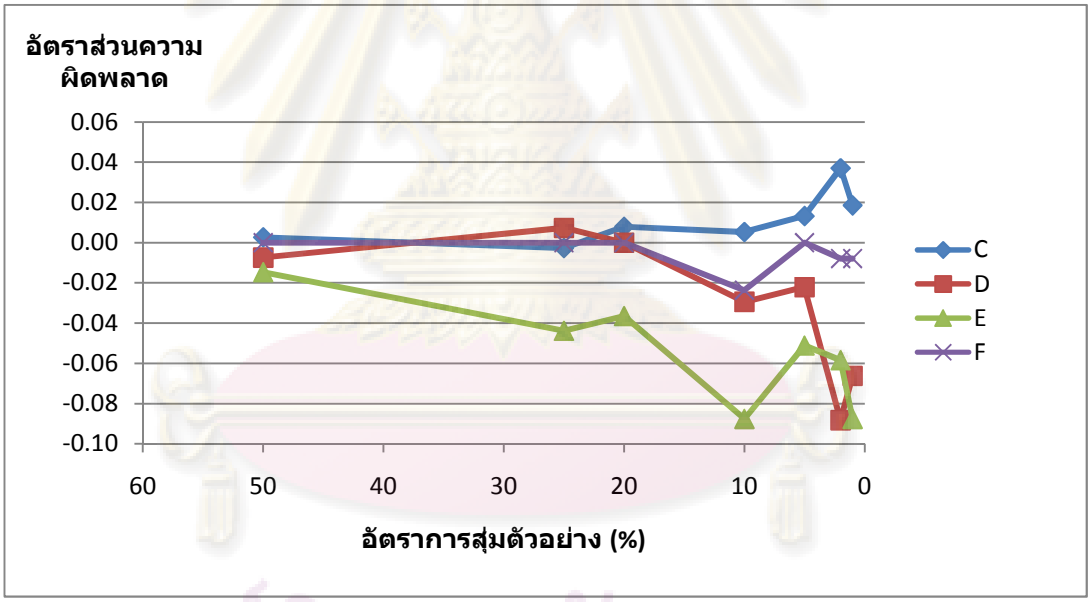
หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
C	28.52	0.00	0.01	0.00	0.00	0.02	0.04	0.04
D	59.47	0.00	0.00	0.00	0.00	0.00	0.05	0.08
E	57.30	0.00	0.00	0.00	0.00	0.01	0.03	0.07
F	56.48	0.00	0.00	0.00	0.01	-0.02	0.01	-0.02



รูปที่ ก.15 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลจริงที่หน้าต่าง 70 วินาที

ตารางที่ ก.16 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลจริงที่หน้าต่าง 100 วินาที

หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
C	27.78	0.00	0.00	0.00	0.01	0.01	0.01	0.04
D	52.40	0.00	-0.01	0.01	0.00	-0.03	-0.02	-0.09
E	40.11	0.00	-0.01	-0.04	-0.04	-0.09	-0.05	-0.06
F	39.58	0.00	0.00	0.00	0.00	-0.02	0.00	-0.01



รูปที่ ก.16 อัตราส่วนความผิดพลาดจากการทดลองเมื่อสุ่มตัวอย่างการเชื่อมต่อกับข้อมูลจริงที่หน้าต่าง 100 วินาที

ตารางที่ ก.17 ผลการคำนวณอัตราส่วนความผิดพลาด
เมื่อสุ่มตัวอย่างกับข้อมูลควบคุมที่หน้าต่าง 30 วินาที

อัตราการกราตรวจ(cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)
10	-0.47	-0.45	-0.45	-0.43	-0.40	-0.32
20	-0.24	-0.23	-0.23	-0.22	-0.21	-0.17
30	-0.16	-0.16	-0.16	-0.15	-0.14	-0.11
40	-0.12	-0.12	-0.12	-0.12	-0.11	-0.08
50	-0.10	-0.10	-0.10	-0.09	-0.09	-0.07
60	-0.08	-0.08	-0.08	-0.08	-0.07	-0.06
70	-0.07	-0.07	-0.07	-0.07	-0.06	-0.05
80	-0.06	-0.06	-0.06	-0.06	-0.05	-0.04
90	-0.05	-0.05	-0.05	-0.05	-0.05	-0.04
100	-0.05	-0.05	-0.05	-0.05	-0.04	-0.03

ตารางที่ ก.18 ผลการคำนวณอัตราส่วนความผิดพลาด
เมื่อสุ่มตัวอย่างกับข้อมูลควบคุมที่หน้าต่าง 50 วินาที

อัตราการกราตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
10	-0.47	-0.46	-0.46	-0.44	-0.42	-0.37	-0.29
20	-0.24	-0.24	-0.24	-0.23	-0.22	-0.20	-0.15
30	-0.16	-0.16	-0.16	-0.16	-0.15	-0.13	-0.10
40	-0.12	-0.12	-0.12	-0.12	-0.11	-0.10	-0.07
50	-0.10	-0.10	-0.10	-0.09	-0.09	-0.08	-0.06
60	-0.08	-0.08	-0.08	-0.08	-0.08	-0.07	-0.05
70	-0.07	-0.07	-0.07	-0.07	-0.07	-0.06	-0.04
80	-0.06	-0.06	-0.06	-0.06	-0.06	-0.05	-0.04
90	-0.05	-0.05	-0.05	-0.05	-0.05	-0.04	-0.03
100	-0.05	-0.05	-0.05	-0.05	-0.05	-0.04	-0.03

ตารางที่ ก.19 ผลการคำนวณอัตราส่วนความผิดพลาด
เมื่อสุ่มตัวอย่างกับข้อมูลควบคุมที่หน้าต่าง 70 วินาที

อัตราการกราตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
10	-0.48	-0.47	-0.47	-0.45	-0.43	-0.39	-0.34
20	-0.24	-0.24	-0.24	-0.23	-0.23	-0.21	-0.18
30	-0.16	-0.16	-0.16	-0.16	-0.15	-0.14	-0.12
40	-0.12	-0.12	-0.12	-0.12	-0.12	-0.11	-0.09
50	-0.10	-0.10	-0.10	-0.10	-0.09	-0.09	-0.07
60	-0.08	-0.08	-0.08	-0.08	-0.08	-0.07	-0.06
70	-0.07	-0.07	-0.07	-0.07	-0.07	-0.06	-0.05
80	-0.06	-0.06	-0.06	-0.06	-0.06	-0.05	-0.04
90	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	-0.04
100	-0.05	-0.05	-0.05	-0.05	-0.05	-0.04	-0.04

ตารางที่ ก.20 ผลการคำนวณอัตราส่วนความผิดพลาด
เมื่อสุ่มตัวอย่างกับข้อมูลควบคุมที่หน้าต่าง 100 วินาที

อัตราการกราตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
10	-0.48	-0.47	-0.47	-0.46	-0.44	-0.41	-0.37
20	-0.24	-0.24	-0.24	-0.24	-0.23	-0.22	-0.20
30	-0.16	-0.16	-0.16	-0.16	-0.16	-0.15	-0.13
40	-0.12	-0.12	-0.12	-0.12	-0.12	-0.11	-0.10
50	-0.10	-0.10	-0.10	-0.10	-0.09	-0.09	-0.08
60	-0.08	-0.08	-0.08	-0.08	-0.08	-0.07	-0.07
70	-0.07	-0.07	-0.07	-0.07	-0.07	-0.06	-0.06
80	-0.06	-0.06	-0.06	-0.06	-0.06	-0.06	-0.05
90	-0.05	-0.05	-0.05	-0.05	-0.05	-0.05	-0.04
100	-0.05	-0.05	-0.05	-0.05	-0.05	-0.04	-0.04

ตารางที่ ก.21 ผลการคำนวณอัตราส่วนความผิดพลาด
เมื่อสุ่มตัวอย่างกับข้อมูลจริงที่หน้าต่าง 30 วินาที

หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)
C	31.74	-0.09	-0.11	-0.12	-0.14	-0.13	-0.11
D	54.17	-0.08	-0.08	-0.09	-0.09	-0.08	-0.06
E	56.59	-0.11	-0.10	-0.09	-0.08	-0.08	-0.06
F	72.67	-0.05	-0.06	-0.06	-0.06	-0.06	-0.05

ตารางที่ ก.22 ผลการคำนวณอัตราส่วนความผิดพลาด
เมื่อสุ่มตัวอย่างกับข้อมูลจริงที่หน้าต่าง 50 วินาที

หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
C	29.62	-0.20	-0.18	-0.18	-0.17	-0.16	-0.13	-0.10
D	61.81	-0.05	-0.06	-0.06	-0.07	-0.07	-0.06	-0.05
E	74.90	-0.07	-0.07	-0.07	-0.06	-0.06	-0.05	-0.04
F	72.94	-0.05	-0.05	-0.06	-0.06	-0.06	-0.05	-0.04

ตารางที่ ก.23 ผลการคำนวณอัตราส่วนความผิดพลาด
เมื่อสุ่มตัวอย่างกับข้อมูลจริงที่หน้าต่าง 70 วินาที

หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
C	28.52	-0.28	-0.25	-0.24	-0.21	-0.18	-0.15	-0.13
D	59.47	-0.10	-0.09	-0.09	-0.08	-0.08	-0.07	-0.06
E	57.30	-0.12	-0.11	-0.11	-0.10	-0.08	-0.07	-0.06
F	56.48	-0.11	-0.11	-0.11	-0.10	-0.09	-0.08	-0.06

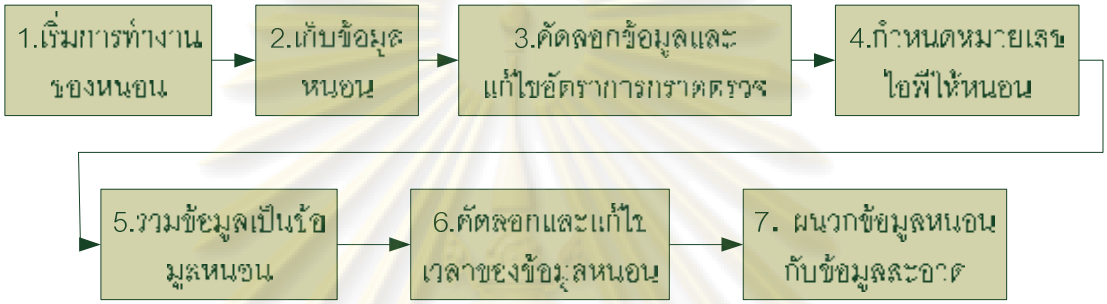
ตารางที่ ก.24 ผลการคำนวณอัตราส่วนความผิดพลาด
เมื่อสุ่มตัวอย่างกับข้อมูลจริงที่หน้าต่าง 100 วินาที

หมายเลขไอพี	อัตราการกราดตรวจ (cps)	สุ่ม 50 % (2 เอา 1)	สุ่ม 25 % (4 เอา 1)	สุ่ม 20 % (5 เอา 1)	สุ่ม 10 % (10 เอา 1)	สุ่ม 5 % (20 เอา 1)	สุ่ม 2% (50 เอา 1)	สุ่ม 1% (100 เอา 1)
C	27.78	-0.28	-0.27	-0.27	-0.24	-0.21	-0.17	-0.14
D	52.40	-0.05	-0.05	-0.06	-0.07	-0.08	-0.09	-0.08
E	40.11	-0.11	-0.12	-0.12	-0.12	-0.12	-0.11	-0.10
F	39.58	-0.16	-0.15	-0.14	-0.13	-0.12	-0.11	-0.10

ภาคผนวก ข

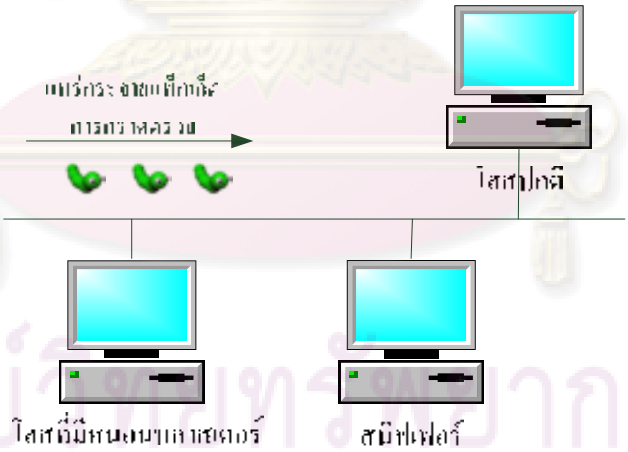
การสร้างข้อมูลควบคุม

วิธีการสร้างข้อมูลควบคุมมี 7 ขั้นตอนแสดงดังรูปที่ ข.1 โดยมีรายละเอียดของขั้นตอนการสร้างดังนี้



รูปที่ ข.1 วิธีการสร้างข้อมูลควบคุม

1. สร้างเครือข่ายสำหรับแพร่กระจายหนอน โดยเป็นเครือข่ายปิด (ไม่เชื่อมต่อกับเครือข่ายอื่นๆ) และเริ่มการทำงานของหนอน ดังรูปที่ ข.2



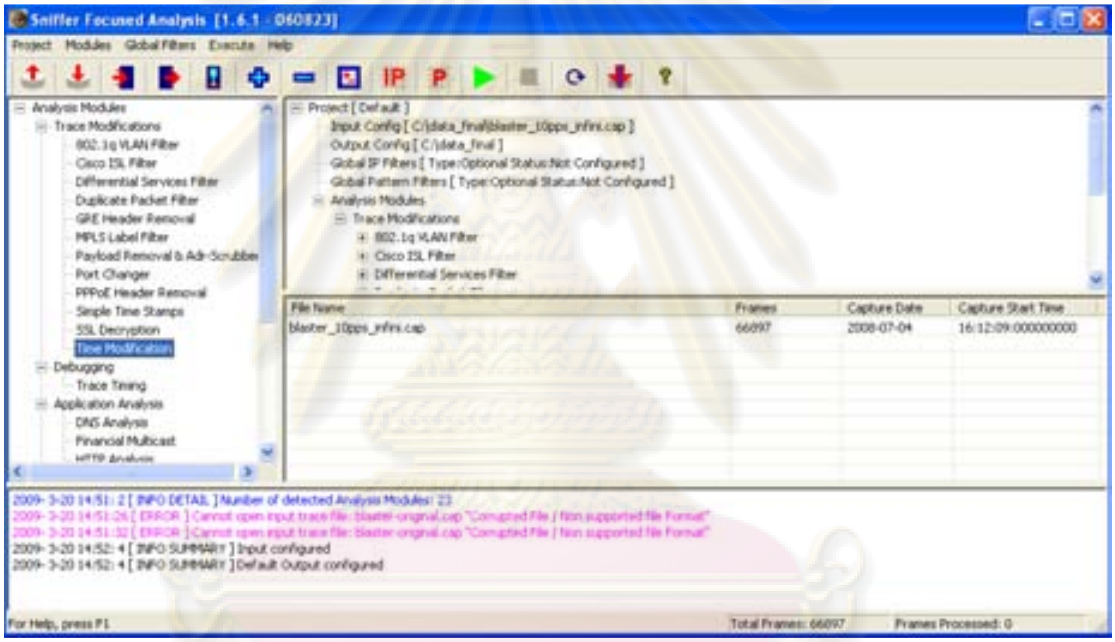
รูปที่ ข.2 การเริ่มการทำงานของหนอนบนเครือข่ายควบคุม

2. ข้อมูลหนอนถูกเก็บด้วยโปรแกรมสนิฟเฟอร์ที่ชื่อว่า วายชาร์ก (wireshark) [28] เนื่องจากเป็นฟรีแวร์ (freeware) และใช้กันโดยแพร่หลาย โดยข้อมูลหนอนที่ถูกเก็บจะอยู่ในรูปแบบ .cap โดยแสดงดังรูปที่ ข.3

1	0.000000	Vmware_2c:c3:de	Broadcast	ARP	Who has 161.200.92.146? Tell 161.200.92.30
2	0.022977	161.200.92.30	228.88.238.200	ICMP	Echo (ping) request
3	0.052561	161.200.92.30	238.139.111.107	ICMP	Echo (ping) request
4	0.052571	161.200.92.30	224.37.120.153	ICMP	Echo (ping) request
5	0.052574	161.200.92.30	231.148.148.216	ICMP	Echo (ping) request
6	0.052576	161.200.92.30	224.222.17.131	ICMP	Echo (ping) request
7	0.052578	161.200.92.30	227.202.237.207	ICMP	Echo (ping) request
8	0.052581	161.200.92.30	232.131.97.65	ICMP	Echo (ping) request
9	0.052583	161.200.92.30	238.246.180.14	ICMP	Echo (ping) request
10	0.052586	161.200.92.30	227.150.246.20	ICMP	Echo (ping) request

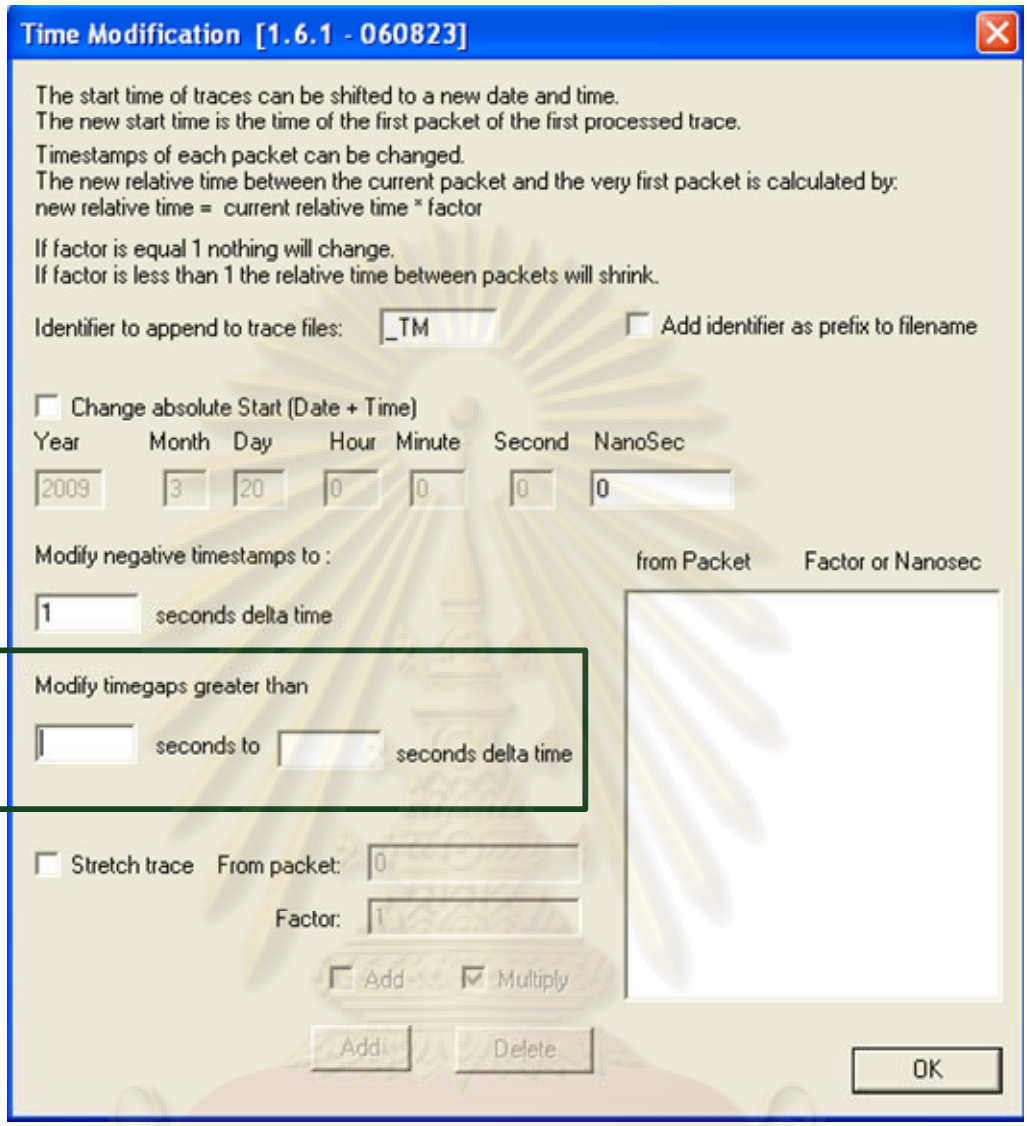
รูปที่ ข.3 รายละเอียดของไฟล์นามสกุล .cap

3. จากนั้นจะคัดลอกและปรับเปลี่ยนหนอนให้มีอัตราการกราดตรวจ 10 20 30 40 50 60 70 80 90 และ 100 การเชื่อมต่อต่อวินาที โดยใช้โปรแกรม Sniffer InfiniStream ในการปรับเปลี่ยนอัตราการกราดตรวจดังรูปที่ ข.4



รูปที่ ข.4 หน้าต่างหลักโปรแกรม Sniffer InfiniStream

ผู้วิจัยใช้ส่วน Time Modification -> Configuration ดังในรูปที่ ข.5 เพื่อปรับเปลี่ยนอัตราการกราดตรวจของหนอน และในหน้าต่าง Configuration จะมีกล่องข้อความให้ปรับเปลี่ยนระยะเวลาระหว่างแพ็กเก็ต ใช้ในการสร้างอัตราการกราดตรวจของหนอนต่างๆ เช่น ถ้าต้องการให้หนอนมีระยะเวลาการกราดตรวจ 10 การเชื่อมต่อต่อวินาที ระยะห่างระหว่างแพ็กเก็ตต้องมีค่า 0.1 วินาที เป็นต้น ในงานวิจัยนี้ใช้หนอนที่มีอัตราการกราดตรวจ 10 20 30 40 50 60 70 80 90 100 การเชื่อมต่อต่อวินาที



รูปที่ ข.5 หน้าต่าง Time Modification

4. หลังจากได้หนอนที่มีอัตราการกราดตรวจตามที่ต้องการ ผู้วิจัยกำหนดหมายเลขไอพีให้หนอนที่อัตราการกราดตรวจต่างๆ เนื่องจากเมื่อถูกตรวจจับด้วยสนอร์ตแล้ว จะสามารถระบุอัตราการกราดตรวจของหนอนได้ โดยใช้หมายเลขไอพีดังตารางที่ ข.1

ตารางที่ ข.1 หมายเลขไอพีของหนอนที่อัตราการกราดตรวจต่างๆ

หมายเลขไอพี	อัตราการกราดตรวจ
0.0.0.10	10
0.0.0.20	20
0.0.0.30	30

ตารางที่ ข.1 (ต่อ) หมายเลขไอพีของหนอนที่อัตราการกราดตรวจต่างๆ

หมายเลขไอพี	อัตราการกราดตรวจ
0.0.0.40	40
0.0.0.50	50
0.0.0.60	60
0.0.0.70	70
0.0.0.80	80
0.0.0.90	90
0.0.0.100	100

ผู้วิจัยใช้โปรแกรม tcprewrite ในการกำหนดหมายเลขไอพีให้ข้อมูลหนอน มีรูปแบบคำสั่งดังนี้

tcprewrite --pnat=*หมายเลขไอพีรับเข้า/หมายเลขสับเน็ตมาส:หมายเลขไอพีส่งออก* --infile=*ไฟล์รับเข้า* --outfile=*ไฟล์ส่งออก*

ตัวอย่างเช่น ถ้าต้องการเปลี่ยนหมายเลขไอพีจากเดิมคือ 192.168.1.1 เป็น 0.0.0.10 โดยไฟล์ต้รับเข้าคือ input.cap และไฟล์ส่งออกคือ output.cap คำสั่งจะเป็นดังนี้

tcprewrite --pnat=192.168.1.1/32:0.0.0.10 –infile=input.cap –outfile=worm10cps.cap

5. เมื่อกำหนดหมายเลขไอพีให้กับหนอนแล้วก็จะรวมข้อมูลหนอนเข้าด้วยกันโดยใช้โปรแกรม mergecap มีรูปแบบดังนี้

mergecap –w *ไฟล์ส่งออก ไฟล์รับเข้า1 ไฟล์รับเข้า2 … ไฟล์รับเข้าn*

คำสั่งที่ใช้มีดังนี้

mergecap –w wormdata.cap worm10cps.cap worm20cps.cap worm30cps.cap worm40cps.cap worm50cps.cap worm60cps.cap worm70cps.cap worm80cps.cap worm90cps.cap worm100cps

6. จากนั้นนำข้อมูลหนอนไปรวมกับข้อมูลสะอาดแต่เนื่องจากข้อมูลสะอาดมีระยะเวลา 22 ชั่วโมง และผู้วิจัยต้องการให้มีการกราดตรวจของหนอนเกิดขึ้นทั้ง 22 ชั่วโมง ดังนั้น จึงคัดลอกข้อมูลหนอนเป็นจำนวน 11 ชุด และปรับเปลี่ยนเวลาของข้อมูลแต่ละชุดให้เรียงต่อกันและอยู่ในช่วงเดียวกับข้อมูลสะอาดโดยใช้โปรแกรม editcap โดยมีรูปแบบคำสั่งดังนี้

editcap -t ±*เวลาที่ปรับเปลี่ยนจากเดิม ไฟล์รับเข้า ไฟล์ส่งออก*

ตัวอย่างคำสั่งที่ใช้มีดังนี้

editcap -t -294523928 wormdata.cap wormdata2.cap

7. จากนั้นจึงรวมข้อมูลด้วยโปรแกรม mergecap โดยคำสั่งที่ใช้มีดังนี้

mergecap controlleddata.cap darpa99.cap wormdata.cap wormdata2.cap wormdata3.cap wormdata4.cap wormdata5.cap wormdata6.cap wormdata7.cap wormdata8.cap wormdata9.cap wormdata10.cap

ภาคผนวก ค

ผลงานตีพิมพ์

ส่วนหนึ่งของงานวิทยานิพนธ์นี้ได้รับการตีพิมพ์เป็นบทความวิชาการในหัวเรื่อง “การสุ่มตัวอย่างแพ็กเก็ตแบบกราดตรวจ” โดยเลิศพงษ์ เลิศไพศาลวงศ์ และ ยรรยง เต็งอำนวย ในงานประชุมวิชาการ “The 12th National Computer Science and Engineering Conference (NCSEC 2008)” ซึ่งจัดขึ้น ณ โรงแรมลองบีช ชลบุรี ประเทศไทย ระหว่างวันที่ 20-21 พฤศจิกายน 2551

The 12th National Computer Science and Engineering Conference (NCSEC 2008)

การสุ่มตัวอย่างแพ็กเก็ตสำหรับตรวจจับหนอนแบบกราดตรวจ

Packet Sampling for Scanning Worm Detection

เอิสพงษ์ เอิสไพฑูรวงศ์ และ ยรรยงเตียอำนวย
ห้องปฏิบัติการวิศวกรรมระบบสารสนเทศ
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ถนนพญาไท เขตปทุมวัน กรุงเทพฯ10330 โทรศัพท์: 0-2218-6991
E-mail: g48tle@cp.eng.chula.ac.th และ Yunyong.T@Chula.ac.th

บทคัดย่อ

ปัจจุบันข้อมูลบนเครือข่ายมีจำนวนมากขึ้นเรื่อยๆ ทำให้การเก็บข้อมูลสำหรับการวิเคราะห์ต้องใช้เนื้อที่มากและต้องมีการบริหารจัดการที่ดี วิธีการสุ่มตัวอย่างแพ็กเก็ตสามารถนำมาใช้กับสนิฟเฟอร์ ทำให้สามารถเก็บข้อมูลเป็นระยะเวลานานขึ้น แต่การสุ่มตัวอย่างแพ็กเก็ตทำให้ประสิทธิภาพในการตรวจจับหนอนลดน้อยลง เพราะอัตราการกราดตรวจของหนอนหลังจากถูกสุ่มตัวอย่างจะลดน้อยลงด้วย จึงต้องเลือกอัตราการสุ่มให้เหมาะสม งานวิจัยนี้ได้เสนอวิธีการสุ่มตัวอย่างสำหรับลดข้อมูล พร้อมทั้งการหาขอบเขตขั้นต่ำของการสุ่มตัวอย่างที่สามารถตรวจจับหนอนแบบกราดตรวจได้

คำสำคัญ: การสุ่มตัวอย่างแพ็กเก็ต, หนอน, สนิฟเฟอร์

Abstract

At present, data volume in the network is increasing dramatically. To keep traffic log for analysis, a very huge storage and extensive administration are needed. Packet sampling applying with sniffer is an interesting method for lengthening logging period. But packet sampling may cause some problems in worm detection performance, since traffic log after sampling may be lost in capturing worm characteristic. Sampling rate needs to be chosen by considering worm scanning characteristic. This article proposes a packet sampling procedure for sniffer to increase interval of logging traffic, as well as lower limit of sampling rate for worm.

Keywords: packet sampling, worm, sniffer

1. คำนำ

ในงานวิจัยของ Pongpaibool [1] ได้กล่าวถึงแนวโน้มข้อมูลจราจรตั้งแต่ปี 1992 ถึง 2007 ทั้งในประเทศและต่างประเทศซึ่งเพิ่มขึ้นอย่างต่อเนื่อง จากแนวโน้มนี้ทำให้เกิดผลกระทบต่อการเก็บข้อมูลเพื่อการวิเคราะห์ด้านต่างๆ เช่น ทางนิติคอมพิวเตอร์ (Computer Forensics) หรือด้านความเครือข่าย เช่น หนอน และการโจมตีแบบปฏิเสธการให้บริการ (Denial of Service) เพราะต้องเก็บข้อมูลจำนวนมากขึ้นทำให้เก็บข้อมูลได้ระยะเวลาน้อยลง จึงได้มีการนำวิธีการสุ่มตัวอย่างมาใช้เพื่อลดปริมาณข้อมูล อีกแม้ว่าทำให้ได้ข้อมูลไม่ครบ แต่ผู้วิจัยจะแสดงให้เห็นในงานนี้ว่าการตรวจสอบพฤติกรรมการกราดตรวจของหนอนสามารถทำได้แม้ข้อมูลจะถูกสุ่มตัวอย่างในอัตราที่ต่ำที่สุด โดยงานวิจัยจะประกอบด้วยส่วนต่างๆ ตามลำดับต่อไปนี้คือ งานวิจัยที่เกี่ยวข้อง ทฤษฎีที่เกี่ยวข้อง การหาขอบเขตขั้นต่ำของการสุ่มตัวอย่าง การสร้างข้อมูลจราจรที่มีหนอน การทดลอง และสรุป

2. งานวิจัยที่เกี่ยวข้อง

ผู้วิจัยได้ศึกษางานวิจัยของ Brauckhoff และคณะ [2] พบว่าเมื่อสุ่มตัวอย่างแพ็กเก็ตเพื่อหาความผิดปกติของเครือข่าย ตัววัดความผิดปกติอย่างเช่น จำนวนแพ็กเก็ตที่ได้จากการสุ่ม สามารถประมาณค่าจำนวนแพ็กเก็ตเมื่อไม่ได้ถูกสุ่มได้อย่างแม่นยำ และค่าเอ็นโทรปีที่อัตราการสุ่มตัวอย่างต่างๆกลับไม่ลดลง งานวิจัยของ Kawahara และคณะ [3] ได้นำเทคนิคการสุ่มตัวอย่างมาใช้กับการตรวจหาความผิดปกติต่างๆของเครือข่ายเพื่อลดขนาดของข้อมูลที่จะเก็บ โดยพบว่า ความผิดปกติ เช่น การกราดตรวจเครือข่าย และ SYN Flood ก่อให้เกิดโฟลว์ (Flow) ขนาดเล็กเป็นจำนวนมาก เมื่อสุ่มตัวอย่างแล้วโฟลว์เหล่านี้จะหายไปเนื่องจากมีโอกาสถูกเลือกน้อยเพราะมีเพียง 1-2 แพ็กเก็ตในโฟลว์เท่านั้น ทำให้ตรวจพบความผิดปกติได้ยาก ดังนั้นเขาจึงเสนอวิธีการแบ่งกลุ่มสำหรับการตรวจจับความผิดปกติโดยการจัดกลุ่มข้อมูล เช่น ถ้าต้องการตรวจจับการกราดตรวจ (Scanning) ให้จัดกลุ่มข้อมูลตามหมายเลขไอพีต้นทาง ซึ่งวิธีการนี้ทำให้เพิ่มประสิทธิภาพในการตรวจจับความผิดปกติจากข้อมูลจราจรที่ถูกสุ่มได้ และงานวิจัยของ Sekar และคณะ [4] เสนอวิธีการตรวจจับหนอนที่มีอัตราการกราดตรวจต่ำ โดยกล่าวว่าพฤติกรรมของหนอนจะพยายามกราดตรวจหาโฮสต์ (Host) เป้าหมายที่ไม่ซ้ำกันเสมอ แต่ในข้อมูลจราจรที่ปกติ โฮสต์จะติดต่อไปยังโฮสต์อื่นๆน้อย แล้วติดต่อซ้ำๆกัน เมื่อขยายช่วงเวลาในการตรวจจับการกราดตรวจของหนอนทำให้สามารถแยกแยะพฤติกรรมการกราดตรวจกับข้อมูลจราจรปกติได้อย่างถูกต้อง

ในงานนี้ผู้วิจัยใช้พรีโพรเซสเซอร์ Flow-portscan ของสนอร์ต [5] ตรวจจับการกราดตรวจของหนอน ซึ่งมีวิธีการตรวจจับใกล้เคียงกับที่ Kawahara และคณะ [3] เสนอไว้ และนำวิธีของ Sekar และคณะ [4] เสนอมาประยุกต์ใช้เพื่อตรวจจับอัตราการกราดตรวจที่ลดลงเนื่องจากการสุ่มตัวอย่าง และคำนวนหาขอบเขตขั้นต่ำอัตราการสุ่มตัวอย่าง

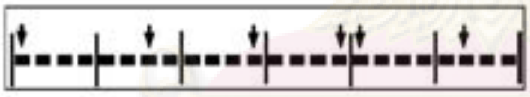
3. ทฤษฎีที่เกี่ยวข้อง

3.1 การสุ่มตัวอย่าง

เพื่อที่จะลดจำนวนข้อมูลจราจรผู้วิจัยได้ศึกษาการสุ่มตัวอย่างแบบต่างๆ [16] ทั้งการสุ่มตัวอย่างแบบมีชั้นภูมิ (stratified random sampling) การสุ่มตัวอย่างแบบมีระบบ (systematic sampling) และการสุ่มตัวอย่างแบบง่าย (simple random) โดยมีรายละเอียดดังต่อไปนี้

3.1.1 การสุ่มตัวอย่างแบบมีชั้นภูมิ

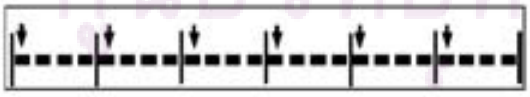
การสุ่มตัวอย่างแบบมีชั้นภูมิ [6] ถูกใช้ในมาตรฐาน RFC 3176 [7] ของเอสโฟล์ว (sFlow) [8] การสุ่มตัวอย่างแบบมีชั้นภูมิจะแบ่งข้อมูลออกเป็นกลุ่มย่อยที่มีขนาดเท่ากันแล้วสุ่มข้อมูลมาหนึ่งครั้งจากแต่ละกลุ่มย่อย ทำให้สามารถตรวจจับการกราดตรวจที่มีรูปแบบได้ดีและโอกาสที่ข้อมูลจะถูกสุ่มออกมามีเท่าๆกันเนื่องจากมีการกระจายการสุ่มไปทั่วทั้งข้อมูลดังแสดงตัวอย่างไว้รูปที่ 1



รูปที่ 1 การสุ่มตัวอย่างแบบมีชั้นภูมิ

3.1.2 การสุ่มตัวอย่างแบบมีระบบ

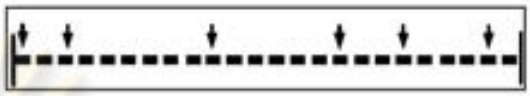
การสุ่มตัวอย่างแบบมีระบบจะเลือกข้อมูลที่ k ของกลุ่มย่อยแต่ละกลุ่ม ยกตัวอย่างเช่นกำหนดให้ k เป็น 1 ก็จะเลือกแพ็กเก็ตแรกของทุกๆกลุ่มข้อมูลย่อย ดังแสดงในรูปที่ 2



รูปที่ 2 การสุ่มตัวอย่างแบบมีระบบ

3.1.2 การสุ่มตัวอย่างแบบง่าย

การสุ่มตัวอย่างแบบง่ายจะทำการสุ่มตัวอย่างจากข้อมูลทั้งหมดโดยไม่มีการแบ่งข้อมูลออกเป็นกลุ่มข้อมูลย่อย โดยมีความน่าจะเป็นในการสุ่มแต่ละแพ็กเก็ตเท่ากัน ดังแสดงในรูปที่ 3



รูปที่ 3 การสุ่มตัวอย่างแบบง่าย

3.2 โปรแกรม Dumpcap

โปรแกรม Dumpcap [9] เป็นโปรแกรมอินเฟซเฟอร์ที่ใช้ในการเก็บกระแสข้อมูลในรูปแบบของ .cap ไฟล์ ซึ่งเป็นไฟล์มาตรฐานสำหรับการเก็บข้อมูลจราจร สามารถกำหนดขนาดและช่วงเวลาของไฟล์ที่จะเก็บได้ เก็บข้อมูลได้อย่างรวดเร็วและก่อให้เกิดการทิ้งแพ็กเก็ตน้อย

3.3 หนอนแบบกราดตรวจ

หนอนแบบกราดตรวจ (Scanning Worm) มีพฤติกรรมที่สำคัญคือ กราดตรวจทางไอพีแอดเดรสเพื่อแพร่กระจายไปยังเครื่องที่มีช่องโหว่ให้มากที่สุด [10] โดยมีขั้นตอนการทำงาน 4 ขั้นตอน [11] คือ การค้นหาเป้าหมาย การตรวจสอบช่องโหว่ การส่งโปรแกรม และการเริ่มการทำงานของโปรแกรม ในงานวิจัยนี้จะตรวจจับหนอนในขั้นตอนค้นหาเป้าหมาย

ตารางที่ 1 อัตราการกราดตรวจของหนอนต่อโฮสต์

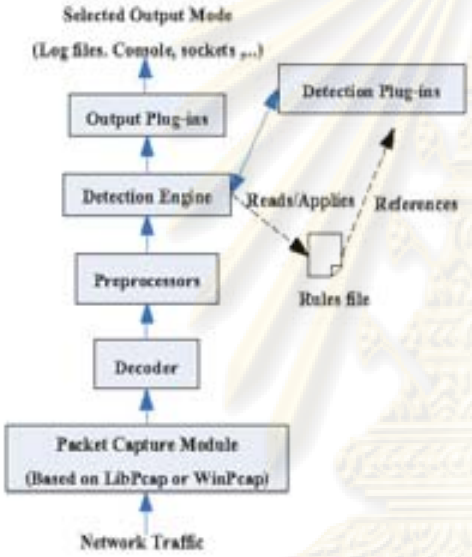
ชนิด	อัตราการกราดตรวจ(แพ็กเก็ตต่อวินาที)
Code Red	200-400
Nimda	200
Blaster	10-20
Welchia	70
Slammer	2000

ในการค้นหาเป้าหมาย หนอนจะหาโฮสต์ที่ออนไลน์อยู่และมีช่องโหว่ โดยใช้วิธีการกราดตรวจแบบอัตโนมัติ ดังนั้นจึงเกิดการพยายามเชื่อมต่อไปยังหมายเลขไอพีต่างๆ เป็นจำนวนมาก ผู้วิจัยได้ศึกษาถึงอัตราการกราดตรวจของหนอนสายพันธุ์ต่างๆ พบว่ามีหลากหลายชนิดและอัตราการกราดตรวจต่างกัน ในที่นี้จะขอยกตัวอย่างหนอนที่เป็นที่รู้จักและก่อให้เกิดความเสียหายเป็นจำนวนมาก ซึ่งแสดงได้ดังตารางที่ 1 [12-15]

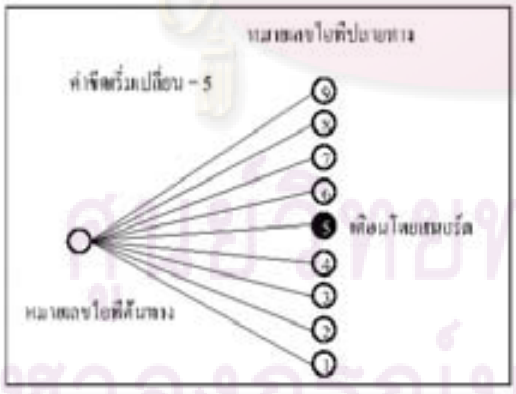
3.4 สนอร์ตและพรีโพรเซสเซอร์ Flow-portscan

สนอร์ต (Snort) [5] เป็นโปรแกรมตรวจจับผู้บุกรุกทางเครือข่าย (Network Intrusion Detection) โดยเป็นโปรแกรมโอเพนซอร์ส (Open Source) ซึ่งใช้กันโดยแพร่หลาย สนอร์ตจะทำงานโดยมีขั้นตอนดังแสดงรูปที่ 4 [16] โปรแกรมสนอร์ตใช้คลังข้อมูล Libpcap (สำหรับลีนุกซ์) หรือ Winpcap (สำหรับวินโดวส์) ในการจับแพ็กเก็ตแล้วส่งให้

ส่วน Decoder ถอดรหัสแพ็กเก็ตเป็นโปรโตคอลต่างๆ หลังจากนั้นพรีโพรเซสเซอร์จะตรวจจับพฤติกรรมผิดปกติต่างๆที่ได้กำหนดไว้ ในงานวิจัยนี้ ผู้วิจัยจะใช้พรีโพรเซสเซอร์ Flow-portscan ในการตรวจจับการกราดตรวจของหนอน จากนั้นใช้ Detection Engine ในการตรวจสอบแพ็กเก็ตโดยใช้กฎ (Rule) ต่างๆ เมื่อเสร็จแล้วสนอร์ตจะแสดงผลลัพธ์ในรูปแบบต่างๆ ผ่านส่วน Output



รูปที่ 4 แผนภาพการทำงานของสนอร์ต

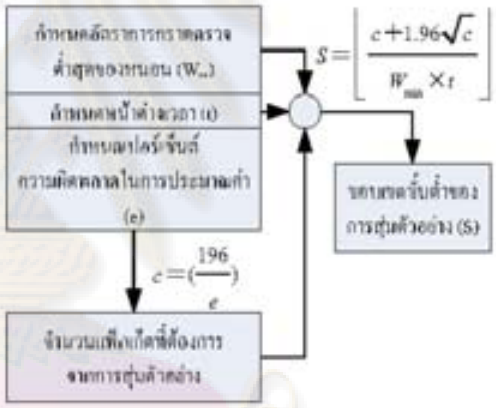


รูปที่ 5 การทำงานของพรีโพรเซสเซอร์ Flow-portscan

ส่วนพรีโพรเซสเซอร์ Flow-portscan จะแบ่งกลุ่มแพ็กเก็ตตามหมายเลขไอพีต้นทาง ถ้าหมายเลขไอพีต้นทางเดียวกันแต่มีหมายเลขไอพีปลายทาง หมายเลขพอร์ตปลายทาง หรือหมายเลขโปรโตคอลต่างกันให้นับเป็น 1 คะแนน แต่เนื่องจากงานวิจัยนี้มีขอบเขตแค่การตรวจจับการกราดตรวจของหนอน ดังนั้นจะใช้เฉพาะหมายเลขไอพีปลายทางที่ต่างกันเท่านั้น [17] โดยเก็บคะแนนไว้ในหน่วยความจำเมื่อคะแนนเกินค่าที่กำหนด สนอร์ตจะทำการแจ้งเตือนว่ามีความผิดปกติเกิดขึ้น การเก็บคะแนนถูกรีเซ็ตเป็น 0 และเริ่มนับใหม่ทุกครั้งเมื่อหมดช่วงเวลาตรวจจับ โดยการทำงานของพรีโพรเซสเซอร์สามารถแสดงได้ในรูปที่ 5

4. การหาขอบเขตขั้นต่ำของการสุ่มตัวอย่าง

เนื่องจากการสุ่มตัวอย่างทำให้จำนวนแพ็กเก็ตลดลง ดังนั้นอัตราการกราดตรวจของหนอนจึงลดลงจากความเป็นจริงด้วย ทำให้ไม่สามารถตรวจจับหนอนที่อัตราการกราดตรวจเดิมได้ด้วยค่าขีดเริ่มเปลี่ยน (Threshold) และช่วงเวลาตรวจจับเดิม ผู้วิจัยสามารถคำนวนหาขอบเขตขั้นต่ำของการสุ่มตัวอย่าง โดยมีขั้นตอนดังรูปที่ 6



รูปที่ 6 การหาขอบเขตขั้นต่ำของการสุ่มตัวอย่าง

เริ่มต้นคือ กำหนดตัวแปร 3 ตัว ได้แก่ อัตราการกราดตรวจต่ำสุดของหนอนที่สามารถตรวจจับได้ (W_{min}) มีหน่วยเป็นแพ็กเก็ตต่อวินาที ช่วงเวลาตรวจจับการกราดตรวจ (t) และเปอร์เซ็นต์ความผิดพลาดในการประมาณค่า (e)

กำหนดให้ c คือจำนวนแพ็กเก็ตจากการสุ่มตัวอย่าง C คือจำนวนแพ็กเก็ตทั้งหมด และ S คืออัตราการสุ่ม (เช่นตัวอย่างเช่น 1/3 หมายความว่าสุ่มตัวอย่างแพ็กเก็ต 1 แพ็กเก็ต ทุกๆ 3 แพ็กเก็ต) ข้อมูลที่ถูกสุ่มจะลดลงตามอัตราสุ่มดังสมการที่ 1 [18-19]

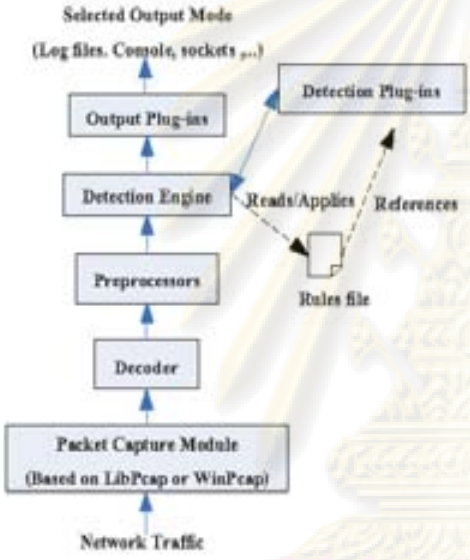
$$c = C \times S \tag{1}$$

โดยมีเปอร์เซ็นต์ความผิดพลาดในการประมาณค่า (e) ดังสมการที่ 2 [18-19]

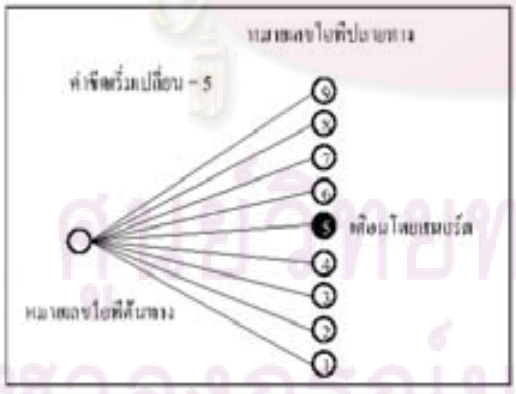
$$e \le 196 \times \sqrt{\frac{1}{c}} \tag{2}$$

นั่นหมายความว่าเมื่อค่า c มากกว่า e ก็จะน้อยเช่นถ้าสุ่มตัวอย่างจากข้อมูลจราจรได้ 10,000 แพ็กเก็ต ค่า e จะเป็น 1.96 % ค่า c ที่แปรผกผัน

ส่วน Decoder ถอดรหัสแพ็กเก็ตเป็นโปรโตคอลต่างๆ หลังจากนั้นพรีโพรเซสเซอร์จะตรวจจับพฤติกรรมผิดปกติต่างๆที่ได้กำหนดไว้ ในงานวิจัยนี้ ผู้วิจัยจะใช้พรีโพรเซสเซอร์ Flow-portscan ในการตรวจจับการกวาดตรวจของหนอน จากนั้นใช้ Detection Engine ในการตรวจสอบแพ็กเก็ตโดยใช้กฎ (Rule) ต่างๆ เมื่อเสร็จแล้วสนอร์ตจะแสดงผลลัพธ์ในรูปแบบต่างๆ ผ่านส่วน Output



รูปที่ 4 แผนภาพการทำงานของสนอร์ต

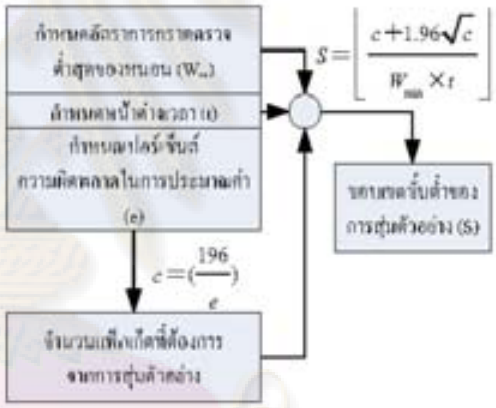


รูปที่ 5 การทำงานของพรีโพรเซสเซอร์ Flow-portscan

ส่วนพรีโพรเซสเซอร์ Flow-portscan จะแบ่งกลุ่มแพ็กเก็ตตามหมายเลขไอพีต้นทาง ถ้าหมายเลขไอพีต้นทางเดียวกันแต่มีหมายเลขไอพีปลายทาง หมายเลขพอร์ตปลายทาง หรือหมายเลขโปรโตคอลต่างกันให้นับเป็น 1 คะแนน แต่เนื่องจากงานวิจัยนี้มีขอบเขตแค่การตรวจจับการกวาดตรวจของหนอน ดังนั้นจะใช้เฉพาะหมายเลขไอพีปลายทางที่ต่างกันเท่านั้น [17] โดยเก็บคะแนนไว้ในหน่วยความจำ เมื่อคะแนนเกินค่าที่กำหนด สนอร์ตจะทำการแจ้งเตือนว่ามีความผิดปกติเกิดขึ้น การเก็บคะแนนถูกรีเซ็ตเป็น 0 และเริ่มนับใหม่ทุกครั้งเมื่อหมดช่วงเวลาตรวจจับ โดยการทำงานของพรีโพรเซสเซอร์สามารถแสดงได้ในรูปที่ 5

4. การหาขอบเขตขั้นต่ำของการสุ่มตัวอย่าง

เนื่องจากการสุ่มตัวอย่างทำให้จำนวนแพ็กเก็ตลดลง ดังนั้นอัตราการกวาดตรวจของหนอนจึงลดลงจากความเป็นจริงด้วย ทำให้ไม่สามารถตรวจจับหนอนที่อัตราการกวาดตรวจเดิมได้ด้วยค่าขีดเริ่มเปลี่ยน (Threshold) และช่วงเวลาตรวจจับเดิม ผู้วิจัยสามารถคำนวนหาขอบเขตขั้นต่ำของการสุ่มตัวอย่าง โดยมีขั้นตอนดังรูปที่ 6



รูปที่ 6 การหาขอบเขตขั้นต่ำของการสุ่มตัวอย่าง

เริ่มต้นคือ กำหนดตัวแปร 3 ตัว ได้แก่ อัตราการกวาดตรวจต่ำสุดของหนอนที่สามารถตรวจจับได้ (W_{min}) มีหน่วยเป็นแพ็กเก็ตต่อวินาที ช่วงเวลาตรวจจับการกวาดตรวจ (t) และเปอร์เซ็นต์ความผิดพลาดในการประมาณค่า (e)

กำหนดให้ c คือจำนวนแพ็กเก็ตจากการสุ่มตัวอย่าง C คือจำนวนแพ็กเก็ตทั้งหมด และ S คืออัตราการสุ่ม (เก็บตัวอย่างเช่น 1/3 หมายความว่าสุ่มตัวอย่างแพ็กเก็ต 1 แพ็กเก็ต ทุกๆ 3 แพ็กเก็ต) ข้อมูลที่ถูกสุ่มจะลดลงตามอัตราสุ่มดังสมการที่ 1 [18-19]

$$c = C \times S \tag{1}$$

โดยมีเปอร์เซ็นต์ความผิดพลาดในการประมาณค่า (e) ดังสมการที่ 2 [18-19]

$$e \leq 196 \times \sqrt{\frac{1}{c}} \tag{2}$$

นั่นหมายความว่าเมื่อค่า c มาก ค่า e ก็จะน้อยเช่นถ้าสุ่มตัวอย่างจากข้อมูลจราจรได้ 10,000 แพ็กเก็ต ค่า e จะเป็น 1.96 % ค่า c ที่แปรผกผัน

ในการทดลองนี้เพราะเป็นหนอนที่มีอัตราการกราดตรวจต่ำ ถ้าตรวจจับได้ก็สามารถตรวจจับหนอนที่มีอัตราการกราดตรวจสูงชนิดอื่นๆได้

จากการทดลองได้หนอนบลาสเตอร์จำนวน 66,897 แพ็กเก็ต ขนาด 5 เมกะไบต์ ช่วงเวลาของข้อมูลประมาณ 90 นาที มีอัตราการกราดตรวจเฉลี่ย 11 แพ็กเก็ตต่อวินาที ผู้วิจัยได้ทำสแนาไฟล์และแก้ไขอัตราการกราดตรวจของหนอนให้เป็น 5 ถึง 15 แพ็กเก็ตต่อวินาที ดังนั้นจึงได้ไฟล์หนอนจำนวน 11 ไฟล์ คงอัตราการกราดตรวจและผู้วิจัยใช้เฉพาะข้อมูล 12 นาทีแรกเท่านั้นเนื่องจากข้อมูลจราจรปกติมีระยะเวลา 12 นาที ขั้นตอนการสร้างข้อมูลหนอนสามารถแสดงได้ในรูปที่ 8

5.3 การรวมข้อมูลจราจร

ในการรวมข้อมูลตามขั้นตอนที่ 5.1 และ 5.2 เข้าด้วยกัน ผู้วิจัยได้ใช้วิธีการปรับเปลี่ยนช่วงเวลาของข้อมูลหนอนให้อยู่ในช่วงเวลาของข้อมูลจราจรปกติโดยช่วงเวลาระหว่างแพ็กเก็ตของข้อมูลหนอนยังคงเท่าเดิม ซึ่งแสดงดังตารางที่ 2 แล้วนำมารวมกันโดยใช้เวลาเป็นตัวอ้างอิง วิธีการรวมข้อมูลเช่นนี้ทำให้รูปแบบการกราดตรวจของหนอนไม่เปลี่ยนแปลง

ตารางที่ 2 ตัวอย่างการเปลี่ยนเวลาของหนอนหัวแพ็กเก็ตแรก

หมายเลขแพ็กเก็ต	เวลาเดิม	เวลาใหม่
1	07-08-11 17:36:30.000000	06-08-16 10:10:02.837700
2	07-08-11 17:36:30.022584	06-08-16 10:10:02.860284
3	07-08-11 17:36:30.109595	06-08-16 10:10:02.947295
4	07-08-11 17:36:30.110102	06-08-16 10:10:02.947802
5	07-08-11 17:36:30.175600	06-08-16 10:10:03.013300

6. การทดลอง

งานวิจัยนี้ได้แบ่งการทดลองออกเป็นสองส่วน ส่วนแรกผู้วิจัยจะแสดงให้เห็นว่าการสุ่มตัวอย่างทำให้สามารถเก็บช่วงเวลาของข้อมูลได้มากขึ้น ส่วนที่สองจะทดสอบว่าข้อมูลจากการสุ่มตัวอย่างสามารถตรวจจับอัตราการกราดตรวจต่ำสุดของหนอนที่กำหนดได้ด้วยด้วยพรีโพรเซสเซอร์ Flow-portscan

6.1 เครื่องมือสำหรับการทดลอง

เครื่องมือสำหรับการทดลองประกอบด้วย เครื่องคอมพิวเตอร์สำหรับเก็บและสุ่มตัวอย่างแพ็กเก็ต มีคุณสมบัติดังนี้ Pentium IV 2.66 กิกะเฮิร์ท หน่วยความจำ 512 เมกะไบต์ ติดตั้งระบบปฏิบัติการลีนุกซ์ 2.6 Fedora Core 4 เนื้อที่สำหรับเก็บข้อมูล 250 กิกะไบต์

6.2 การทำนายช่วงเวลาเก็บข้อมูลที่เพิ่มขึ้น

จากข้อมูลการจราจรที่ผ่านเกตเวย์ของคณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ตั้งแต่เดือน มิถุนายน พ.ศ. 2550 ถึงเดือน มิถุนายน พ.ศ. 2551 พบว่ามีค่าเฉลี่ยการใช้แบนด์วิดท์ 78.03 เมกะบิตต่อวินาที (9.75 เมกะไบต์ต่อวินาที) โดยเป็นการรับข้อมูล 65.39 เมกะบิตต่อวินาที และส่งข้อมูล 12.64 เมกะบิตต่อวินาที แต่ในงานนี้ผู้วิจัยจะเก็บเฉพาะส่วนหัวของแพ็กเก็ตเพื่อนำมาตรวจจับพฤติกรรมการกราดตรวจของหนอน จึงเก็บเฉพาะ 68 ไบต์แรกเท่านั้น [20] โดยที่ขนาดของแพ็กเก็ตเฉลี่ยคือ 680 ไบต์ ซึ่งคิดเป็น 10 % นั่นหมายความว่าถ้าเก็บเฉพาะส่วนหัวของแพ็กเก็ต ข้อมูลจะถูกเก็บด้วยอัตรา 0.975 เมกะไบต์ต่อวินาที ด้วยขนาดเนื้อที่ 250 กิกะไบต์ จะสามารถเก็บข้อมูลได้ประมาณ 315,077 วินาที หรือประมาณ 87 ชั่วโมงเท่านั้น แต่ถ้าใช้การสุ่มตัวอย่างที่อัตราสุ่มต่ำลง ก็จะสามารถเก็บข้อมูลได้ในช่วงเวลาที่มากขึ้น การคำนวนทำได้ดังนี้ ให้ K เป็นขนาดเนื้อที่ S เป็นอัตราการสุ่มตัวอย่าง และ B เป็นอัตราการใช้ข้อมูล การทำนายช่วงเวลาสำหรับเก็บข้อมูล (I) สามารถคำนวนได้ดังสมการที่ 8

$$I = \frac{K}{B \times S} \qquad (8)$$

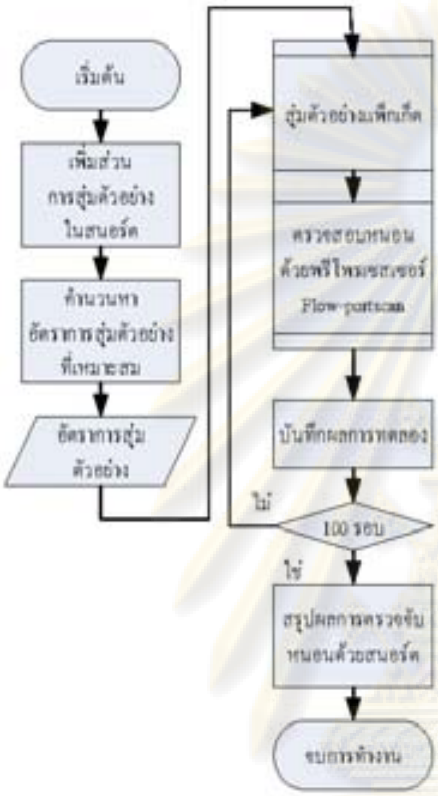
สมการที่ 8 ไม่คิดค่า ϵ เนื่องจากข้อมูลที่เก็บโดยทั่วไปจะมีจำนวนแพ็กเก็ตสูงมาก (เกิน 1,000,000 แพ็กเก็ต)

ตารางที่ 3 การทำนายช่วงเวลาที่สามารถเก็บข้อมูลได้เมื่อสุ่มตัวอย่าง

อัตราสุ่ม	ช่วงเวลาสำหรับเก็บข้อมูลโดยประมาณ (ชั่วโมง / วัน)
1	87 / 4
1/2	175 / 7
1/3	262 / 11
1/4	350 / 15
1/5	437 / 18
1/6	525 / 22
1/7	612 / 26
1/8	700 / 29
1/9	787 / 33
1/10	875 / 36

จะเห็นว่าอัตราการสุ่มตัวอย่างจะแปรผกผันกับช่วงเวลาสำหรับเก็บข้อมูล โดยการทำนายช่วงเวลาสำหรับเก็บข้อมูลในตารางที่ 3 สามารถทำได้โดยอ้างอิงสมการที่ 8

6.3 การตรวจจับหนอนจากข้อมูลที่ถูกสุ่มตัวอย่าง

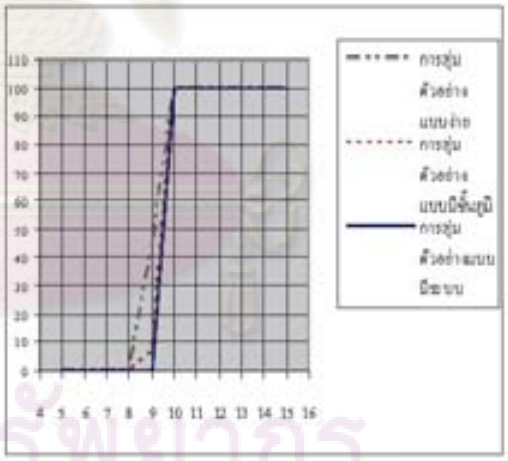


รูปที่ 9 ผังงานกระบวนการทดลอง

ในส่วนนี้จะเป็นการทดสอบสมมติฐานที่ตั้งไว้คือข้อมูลที่ถูกสุ่มแล้วยังคงสามารถตรวจพบการกราดตรวจของหนอนได้ โดยแก้ไขรหัสต้นแบบของสนอร์ต เพื่อเพิ่มความสามารถในการสุ่มตัวอย่างแพ็กเก็ต หลังจากนั้นจึงคำนวนหาขอบเขตขั้นต่ำของการสุ่มตัวอย่างสำหรับเครือข่าย สุ่มตัวอย่างข้อมูลที่ได้ด้วยโปรแกรมสนอร์ต จำนวน 100 ครั้ง แล้วสรุปผลการตรวจจับหนอน ดังรูปที่ 9 ข้อมูลสำหรับการทดลองคือข้อมูลจราจรที่มีหนอนบลาสเตอร์ จำนวน 11 ไฟล์ มีอัตราการกราดตรวจของหนอนตั้งแต่ 5 ถึง 15 แพ็กเก็ตต่อวินาทีตามลำดับ

ในขั้นตอนการหาขอบเขตขั้นต่ำของการสุ่มตัวอย่างในหัวข้อที่ 4 ต้องมีการกำหนดตัวแปร 3 ตัว ได้แก่ อัตราการกราดตรวจต่ำสุดของหนอน (W_{min}) ขนาดของช่วงเวลาตรวจจับการกราดตรวจ (t) และเปอร์เซ็นต์ความผิดพลาดในการประมาณค่า (e)

ในงานวิจัยนี้จะตรวจจับอัตราการกราดตรวจต่ำสุดคือ 10 แพ็กเก็ตต่อวินาที ซึ่งเป็นอัตราการกราดตรวจที่ต่ำที่สุดของหนอนบลาสเตอร์ ในขณะที่การใช้งานปกติมีอัตราการติดต่อไปยังโฮสต์ที่แตกต่างกันไม่เกิน 5 แพ็กเก็ตต่อวินาที [12] และถ้ากำหนดช่วงเวลาตรวจจับการกราดตรวจกว้างมากขึ้นจะยิ่งสามารถแยกระหว่างข้อมูลจราจรปกติและข้อมูลจราจรที่มีหนอนได้ถูกต้องมากยิ่งขึ้นดังที่กล่าวไว้ในหัวข้อที่ 2 [4] ตัวแปรตัวต่อมาคือช่วงเวลาตรวจจับการกราดตรวจ โดยถ้าช่วงเวลาในการตรวจจับมาก ก็จะสามารถตรวจพบแพ็กเก็ตที่ต้องการได้มาก (ค่า c) ทำให้เปอร์เซ็นต์ความผิดพลาดในการประมาณค่าน้อยลงดังในสมการที่ 2 ในงานวิจัยของ Sekar และคณะ [4] ได้กำหนดให้ช่วงเวลาตรวจจับการกราดตรวจมีตั้งแต่ 20 ถึง 500 วินาที แต่ในงานวิจัยนี้เป็นการวิเคราะห์ข้อมูลที่เก็บได้ ซึ่งเป็นข้อมูลที่ไม่เกิดขึ้นในเวลาจริง ดังนั้นเวลาในการเตือนที่มากขึ้น จึงไม่มีผลกระทบต่อความเสียหายกับเครือข่าย ดังนั้นจึงกำหนดค่าช่วงเวลาตรวจจับการกราดตรวจไว้ที่ 500 วินาที เพราะเป็นค่าช่วงเวลาสูงสุดที่ไม่เกินช่วงเวลาของไฟล์ข้อมูลที่นำมาทดลอง และตัวแปรตัวสุดท้ายคือ เปอร์เซ็นต์ความผิดพลาดในการประมาณค่า ซึ่งกำหนดให้มีค่าไม่เกิน 5 % เป็นค่าที่ยอมรับได้สำหรับการประมาณค่าในงานวิจัยนี้ อย่างไรก็ตามในเครือข่ายที่แตกต่างกันสามารถปรับค่าตัวแปรได้ตามความเหมาะสมของแต่ละเครือข่าย ในการทดลองนี้ได้หาขอบเขตขั้นต่ำของการสุ่มตัวอย่างตามขั้นตอนที่แสดงในรูปที่ 6 พบว่าขอบเขตขั้นต่ำของการสุ่มตัวอย่างที่คำนวนได้จากสมการที่ 7 คือ 1/3 นั่นหมายความว่าแพ็กเก็ตที่เป็นหนอนมีโอกาสถูกสุ่มตัวอย่างด้วยความน่าจะเป็น 1/3 ซึ่งจะสามารถตรวจจับการกราดตรวจของหนอนตั้งแต่ 9.68 แพ็กเก็ตต่อวินาที (จากสมการที่ 6) ได้อย่างถูกต้อง (~ 10 แพ็กเก็ตต่อวินาที)



รูปที่ 10 ผลการตรวจพบหนอนเมื่อใช้อัตราการสุ่ม 1/3

ผู้วิจัยได้ทดลองตามกระบวนการที่แสดงไว้ในรูปที่ 9 โดยใช้อัตราการสุ่ม 1/3 ด้วยวิธีการสุ่มตัวอย่างทั้งสามแบบ คือ การสุ่มตัวอย่างแบบมีชั้นภูมิ การสุ่มตัวอย่างแบบง่าย และการสุ่มตัวอย่างแบบมีระบบ จากรูปที่ 10 ที่อัตราการสแกนของหนอนอินเทอร์เน็ต ตั้งแต่ 5 แพ็กเก็ตต่อวินาที ถึง 8 แพ็กเก็ตต่อวินาที ไม่สามารถตรวจจับหนอนอินเทอร์เน็ตได้เลย ส่วนที่อัตราการสแกน 9 แพ็กเก็ตต่อวินาที จะตรวจจับพบการสแกนได้แต่ไม่ทุกครั้ง ในการสุ่มตัวอย่างแบบง่าย และการสุ่มตัวอย่างแบบมีชั้นภูมิ ที่อัตราการสแกน 10 แพ็กเก็ตต่อวินาที สามารถตรวจจับ

การสแกนได้ทุกครั้งที่ทดลอง ซึ่งแสดงให้เห็นว่าการคำนวนหาอัตราการสุ่มที่เหมาะสมสำหรับตรวจจับการสแกนของหนอนอินเทอร์เน็ตมีความถูกต้อง

7. สรุป

เนื่องจากความสามารถในการเก็บข้อมูลที่มีจำกัดและแนวโน้มการใช้เครือข่ายเพิ่มขึ้นเรื่อยๆ งานวิจัยนี้ใช้การสุ่มตัวอย่างที่ได้จากการคำนวน ทำให้เก็บข้อมูลสำหรับการวิเคราะห์เป็นระยะเวลานานขึ้นเพราะจำนวนแพ็กเก็ตลดน้อยลง และข้อมูลที่ถูกสุ่มตัวอย่างสามารถตรวจจับการกระทำตรวจของหนอนได้อย่างถูกต้องแม่นยำจากอัตราการสุ่ม 1/3 ที่คำนวนได้ในงานวิจัยนี้ทำให้เก็บข้อมูลเพิ่มขึ้นได้ 3 เท่า (จาก 87 ชั่วโมงเป็น 262 ชั่วโมง) และสามารถตรวจจับอัตราการกระทำตรวจของหนอนตั้งแต่ 10 แพ็กเก็ตต่อวินาทีได้อย่างถูกต้อง

8. กิตติกรรมประกาศ

ขอขอบคุณ กิตติศักดิ์ อิวรรณดูล ที่ให้โปรแกรมหนอน และคุณชนินทร์ มหารักษ์ ที่ช่วยเหลือเรื่องคอมพิวเตอร์สำหรับการทดลองมีปัญหา

เอกสารอ้างอิง

[1] P. Pongpaibool, *Characteristics of Internet Traffic in Thailand*, 3rd ECTI-CON 2006, Ubolrachatani, Thailand, May 2006.

[2] D. Brauckhoff, B. Tellenbach, A. Wagner, M. May, and A. Lakhina. "Impact of packet sampling on anomaly detection metrics," *In IMC '06: Proceedings of the 6th ACM SIGCOMM on Internet measurement*, pages 159–164, NewYork, NY, USA, 2006. ACM Press.

[3] R. Kawahara, T. Mori, N. Kamiyama, S. Harada, S. Asano, "A Study on Detecting Network Anomalies Using Sampled Flow Statistics," *Applications and the Internet Workshops, 2007. SAINT Workshops 2007. International Symposium on* , vol., no., pp.81-81, Jan. 2007.

[4] V. Sekar, Y. Xie, M. K. Reiter, and H. Zhang, "A multi-resolution approach for worm detection containment," in *Proc. International Conference on Dependable Systems and Networks*, 2006.

[5] M. Roesch and C. Green, "Snort User's Manual." [Online]. Available: http://www.snort.org [27 July 2008].

[6] K. Claffy, G. Polyzos, and H. Braun, "Application of Sampling Methodologies to Network Traffic Characterization", *Computer Communication Review*, Vol. 23, No. 4, pp. 194 - 203, 1993.

[7] P. Phaal, S. Panchen, N. McKee, "InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks", RFC 3176, Sep 2001.

[8] "sFlow." [Online]. Available: http://www.sflow.org [27 July 2008].

[9] "Dumpcap." [Online]. Available: http://www.wireshark.org/docs/man-pages/dumpcap.html [27 July 2008].

[10] F. Buchholz, T. Daniels, J. Early, "Digging for worms, fishing for answers," *Computer Security Applications Conference, 2002. Proceedings. 18th Annual*, vol., no., pp. 219-226, 2002.

[11] M. Lee, T. Shon, K. Cho, M. Chung, J. Seo, J. Moon, *An Approach for Classifying Internet Worms Based on Temporal Behaviors and Packet Flows*, Lecture Notes in Computer Science 2007.

[12] M.M. Williamson, "Throttling viruses: restricting propagation to defeat malicious mobile code," *Computer Security Applications Conference, 2002. Proceedings. 18th Annual*, vol., no., pp. 61-68, 2002.

[13] J. Twycross and M.M. Williamson, "Implementing and Testing a Virus Throttle," *Proc. 12th Usenix Security Symp.*, Usenix Assoc., 2003, pp. 285-294.

[14] C. Wong, S. Bielski, A. Studer, C. Wang, "On the Effectiveness of Rate Limiting Mechanisms," *8th International Symposium on Recent Advances in Intrusion Detection (RAID 2005)*, September 7-9, 2005, Seattle, Washington.

[15] C.C. Zou, W. Gong, D. Towsley, "Worm propagation modeling and analysis under dynamic quarantine defense," *Proceedings of the 2003 ACM workshop on Rapid malcode*, October 27-27, 2003, Washington, DC, USA.

[16] A. F. Arboleda and Ch. E. Bedón, "Snort diagrams for developers." [Online]. Available: http://afrodita.unicauca.edu.co/~cbedon/snort/snortdevdiagrams.html [27 July 2008].

[17] A.A. Cardenas, J.S. Baras and V. Ramezani, "Distributed Change Detection for Worms, DDoS and other Network Attacks," in *ACC '04, Boston*, MA, 2004.

[18] P. Phaal and S. Panchen, "Packet Sampling Basics." [Online]. Available: http://www.sflow.org/packetSamplingBasics/index.htm. [27 July 2008].

[19] J. Jedwab, P. Phaal, and B. Pinna. "Traffic estimation for the largest sources on a network, using packet sampling with limited storage," Technical Report HPL-92-35, HP Labs Technical Report, 1992.

[20] "Tcpdump." [Online]. Available: http://linux.die.net/man/8/tcpdump. [27 July 2008].

เอิสตงษ์ เอิสไตสาดวงศ์ สำเร็จการศึกษาระดับปริญญาตรี สาขาวิศวกรรมคอมพิวเตอร์ จากมหาวิทยาลัยอัสสัมชัญ พ.ศ. 2548 โดยปัจจุบันกำลังศึกษาระดับมหาบัณฑิต สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย งานวิจัยที่สนใจได้แก่ Network Security, Sniffer, Worm, และ Intrusion Detection System.

อรรอง เด็งอำนวย สำเร็จปริญญาบัณฑิต วิศวกรรมไฟฟ้า จุฬาลงกรณ์มหาวิทยาลัย เมื่อ พ.ศ. 2519 ปริญญามหาบัณฑิต วิทยาศาสตร์คอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย เมื่อ พ.ศ. 2522 และ Ph.D. Computer Science จาก Iowa State University, U.S.A. เมื่อ ค.ศ. 1984 ปัจจุบันเป็นอาจารย์ประจำภาควิชา วิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย งานวิจัยที่สนใจได้แก่ Internet Technology, Security, Enterprise Systems & Architecture, และ Distributed Systems.

ประวัติผู้เขียนวิทยานิพนธ์

นายเลิศพงษ์ เลิศไพศาลวงศ์ เกิดเมื่อวันที่ 22 มีนาคม พ.ศ. 2527 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาระดับปริญญาวิศวกรรมศาสตรบัณฑิต จากภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยอัสสัมชัญ ในปีการศึกษา 2547 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย ในปีการศึกษา 2548