ระบบตรวจจับการบุกรุกทางเครือข่าย

กรณีศึกษา การตรวจจับการโจมตีแบบ Ping Flood

Network Intrusion Detection System (NIDS)

Case study : Ping Flood Detection

นายณพงศ์ กิ่งเกล้า รหัส 44362556

นายธวัชชัย สิงงาม รหัส 44362614

นางสาวไพลิน สนานคุณ รหัส 44362903

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2547

ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน : ระบบตรวจจับการบุกรุกทางเครือข่าย
กรณีศึกษา การตรวจจับการโจมตีแบบ Ping Flood

ผู้ดำเนินโครงงาน : นายณพงศ์ กิ่งเกล้า รหัส 44362556
นายธวัชชัย สิงงาม รหัส 44362614
นางสาวไพลิน สนานคุณ รหัส 44362903

อาจารย์ที่ปรึกษา : อาจารย์พงศ์พันธ์ กิจสนาโยธิน

สาขาวิชา : วิศวกรรมคอมพิวเตอร์

ภาควิชา : วิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปีการศึกษา : 2547

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะกรรมการสอบโครงงานวิศวกรรม

P. Kijsanthin .....ประธานกรรมการ
(อ. พงศ์พันธ์ กิจสนาโยธิน)

.....กรรมการ
(ดร. สุรเชษฐ์ กานต์ประชา)

.....กรรมการ
(ดร. พนมขวัญ ริยะมงคล)

หัวข้อโครงงาน : ระบบตรวจจับการบุกรุกทางเครือข่าย
กรณีศึกษา การตรวจจับการ โจมตีแบบ Ping Flood

ผู้ดำเนินโครงงาน : นายณพงศ์ กิ่งเกล้า รหัส 44362556
นายธวัชชัย สิงงาม รหัส 44362614
นางสาวไพลิน สนานคุณ รหัส 44362903

อาจารย์ที่ปรึกษา : อาจารย์พงศ์พันธ์ กิจสนาโยธิน

สาขาวิชา : วิศวกรรมคอมพิวเตอร์

ภาควิชา : วิศวกรรมไฟฟ้าและคอมพิวเตอร์

ปีการศึกษา : 2547

.....

บทคัดย่อ

โครงงานนี้เป็นการศึกษาเกี่ยวกับการทำงาน ของระบบตรวจจับการบุกรุกทางเครือข่าย โดยวิเคราะห์ ออกแบบโปรแกรม และพัฒนาโปรแกรมระบบตรวจจับการบุกรุกทางเครือข่ายบนระบบปฏิบัติการตระกูลยูนิกส์

จากการทดสอบระบบตรวจจับการบุกรุกทางเครือข่ายด้วยโปรแกรมที่พัฒนาขึ้น พบว่ามีประสิทธิภาพและสามารถใช้งานได้จริง และเป็นกรณีศึกษาในการตรวจจับแบบ Ping Flood เพื่อพัฒนา และทำให้เกิดประสิทธิภาพในการรักษาความปลอดภัยบนระบบเครือข่ายคอมพิวเตอร์มากยิ่งขึ้น

Project Title : Network Instrusion Detection System (NIDS)
Case study : Ping Flood Detection

Name : Mr.Napong Kingklao ID 44362556
: Mr.Thawatchai Singngam ID 44362614
: Miss.Pailin Sanankun ID 44362903

Project Adviser : Mr.Phongphun Kijsanayothin

Major : Computer Engineering

Department : Electrical and Computer Engineering

Academic year : 2004

Abstract

The purpose of this project is expending the knowledge which concern to study about working process of network intrusion detection system by analysis , design and develop network intrusion detection system.

This project can be applied in network security system. The experiment result shows that the developed program works on computer network system . The intrusion detection program alerts when Ping Flood Intrusion attacks the network system .

กิตติกรรมประกาศ

โครงงานชิ้นนี้ ต้องอาศัยความรู้ในทุกๆด้านทั้งทางด้านฮาร์ดแวร์และซอฟแวร์และต้องศึกษาความรู้เพิ่มเติมต่างๆมากมาย ทางคณะผู้จัดทำโครงงานจึงขอขอบพระคุณอาจารย์พงษ์พันธ์ กิจสนาโยธิน ที่ได้เอื้อเฟื้อห้องทำโปรเจคที่มีความพร้อมสูงทางด้านฮาร์ดแวร์และคำแนะนำดีๆ สำหรับแนวทางในการทำโครงงาน ขอบคุณ www.google.co.th ที่ได้ตอบคำถามทุกคำถาม ทางด้านเทคนิค และความรู้ในการเขียนโปรแกรม ขอบคุณ คุณเรืองไกร รังสิพล ที่ได้แต่งหนังสือที่สุดยอดและเป็นคู่มือของคณะผู้จัดทำตลอดมา ขอบคุณหนังสือเกี่ยวกับแฮกเกอร์ทั้งหลายที่ช่วยสอนคณะผู้จัดทำว่าระบบอินเตอร์เน็ตช่างกว้างใหญ่ น่าค้นหา น่าเรียนรู้ และไม่ปลอดภัยจริงๆ

และที่ขาดไม่ได้จริงๆคือขอขอบคุณเพื่อนร่วมโครงงานทุกท่านที่คอยให้การสนับสนุนและให้กำลังใจกันและกันตลอดมา

นายณพงศ์ กิ่งเกล้า
นายธวัชชัย สิงงาม
นางสาวไพลิน สนานคุณ

สารบัญ

สารบัญ(ต่อ)

หน้า

สารบัญ(ต่อ)

หน้า

สารบัญ(ต่อ)

หน้า

สารบัญ(ต่อ)

สารบัญ(ต่อ)

หน้า

สารบัญรูป

สารบัญรูป(ต่อ)

สารบัญรูป(ต่อ)

สารบัญตาราง

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มา

ในปัจจุบัน คอมพิวเตอร์ได้เข้ามามีบทบาทอย่างมาก ต่อการดำเนินชีวิตของคนเราไม่ว่าจะเป็นการรับส่งไปรษณีย์อิเล็กทรอนิกส์ การซื้อขายสินค้าผ่านอินเทอร์เน็ต การศึกษาค้นคว้าหาความรู้ การโอนไฟล์มาจากเครื่องบริการ หรือการย้ายไฟล์ไปเก็บไว้บนเครื่องบริการ การแลกเปลี่ยนไฟล์ และการทำธุรกิจต่างๆ ดังนั้นเมื่อคอมพิวเตอร์เชื่อมโยงกัน เป็นโครงข่ายขนาดใหญ่ และมีความสำคัญมากขึ้นเรื่อยๆ มีแนวโน้มที่การใช้งานที่แพร่หลาย และเติบโตอย่างรวดเร็ว ในทางกลับกันก็เกิดผู้ที่ไม่หวังดีมากขึ้นเรื่อยๆ เช่นกัน ไม่ว่าจะเป็น การบุกรุกเพื่อยึดครองเครื่องบริการที่เป็นเป้าหมายการขโมยข้อมูลลูกค้า การขโมยข้อมูลบัตรเครดิต การขโมยความลับคู่แข่งทางธุรกิจ การทำให้ระบบหยุดให้บริการ ตรวจจนมีการค้นพบช่องโหว่มากขึ้น ทำให้การบุกรุกของผู้ไม่หวังดี ทำได้ง่ายขึ้นในปัจจุบัน บางครั้งการบุกรุกระบบหนึ่ง อาจใช้ระยะเวลาเป็นเดือน แต่บางครั้งก็อาจใช้เวลาแค่หนึ่งชั่วโมงจากการหาความรู้จากอินเทอร์เน็ต ขึ้นเพื่อป้องกันจากการบุกรุกก็ได้มีการพัฒนาระบบต่างๆมาเพื่อใช้ในการตรวจสอบ และป้องกันการกระทำที่ไม่เหมาะสม

ระบบตรวจจับการบุกรุกทางเครือข่าย (NIDS: Network Intrusion Detection System) ก็เป็นอีกระบบหนึ่งที่สร้างขึ้นมา เพื่อตรวจสอบการบุกรุกทางเครือข่าย เมื่อมีการบุกรุกก็สามารถแจ้งเตือนผู้ดูแลระบบ ให้หาวิธีป้องกันระบบของตนเองให้ปลอดภัยได้ นอกจากนี้ระบบตรวจจับผู้บุกรุกยังสามารถที่จะเก็บข้อมูลไว้เพื่อใช้ในการวิเคราะห์ในโอกาสหน้าได้อีกด้วย

โครงงานนี้ มุ่งเน้นการศึกษาทางด้านการหาวิธีการ การตรวจจับการโจมตีต่างๆ เพื่อใช้ในการพัฒนาระบบตรวจจับการบุกรุกทางเครือข่ายคอมพิวเตอร์ได้ และเป็นระบบที่พัฒนาอยู่บนระบบปฏิบัติการลีนุกซ์หรือยูนิกส์

1.2 วัตถุประสงค์ของโครงงาน

1) ศึกษารายละเอียดในการโจมตีในรูปแบบต่างๆ

2) ศึกษาเกี่ยวกับโพรโตคอลต่างๆ ทั้งวิธีการการทำงาน รวมทั้งข้อบกพร่องของโพรโตคอลแต่ละประเภท

3) ศึกษาการสำรวจระบบ และแนวทางตรวจสอบการสำรวจระบบ

4) ศึกษาการโจมตีระบบในรูปแบบต่างๆ รวมถึงวิธีการในการตรวจสอบการถูกโจมตี

5) ศึกษาวิธีการในการเจาะเข้าสู่ระบบเครือข่ายคอมพิวเตอร์และค้นหาแนวทางในการป้องกันหรือตรวจสอบวิธีการเข้าสู่ระบบเครือข่ายโดยวิธีดังกล่าว

6) พัฒนาระบบตรวจจับการบุกรุกระบบเครือข่ายทางคอมพิวเตอร์ได้

7) ศึกษาและจัดแบ่งประเภทของการโจมตีต่างๆ โดยเฉพาะบนระดับชั้นเครือข่ายชั้นทรานสปอร์ต (Transport) ชั้นแอพพลิเคชั่น และศึกษาหาแนวทางในการตรวจสอบการโจมตีเหล่านั้น

1.3 ขอบข่ายของโครงงาน

1) ออกแบบและพัฒนาระบบตรวจจับการบุกรุกทางเครือข่ายคอมพิวเตอร์

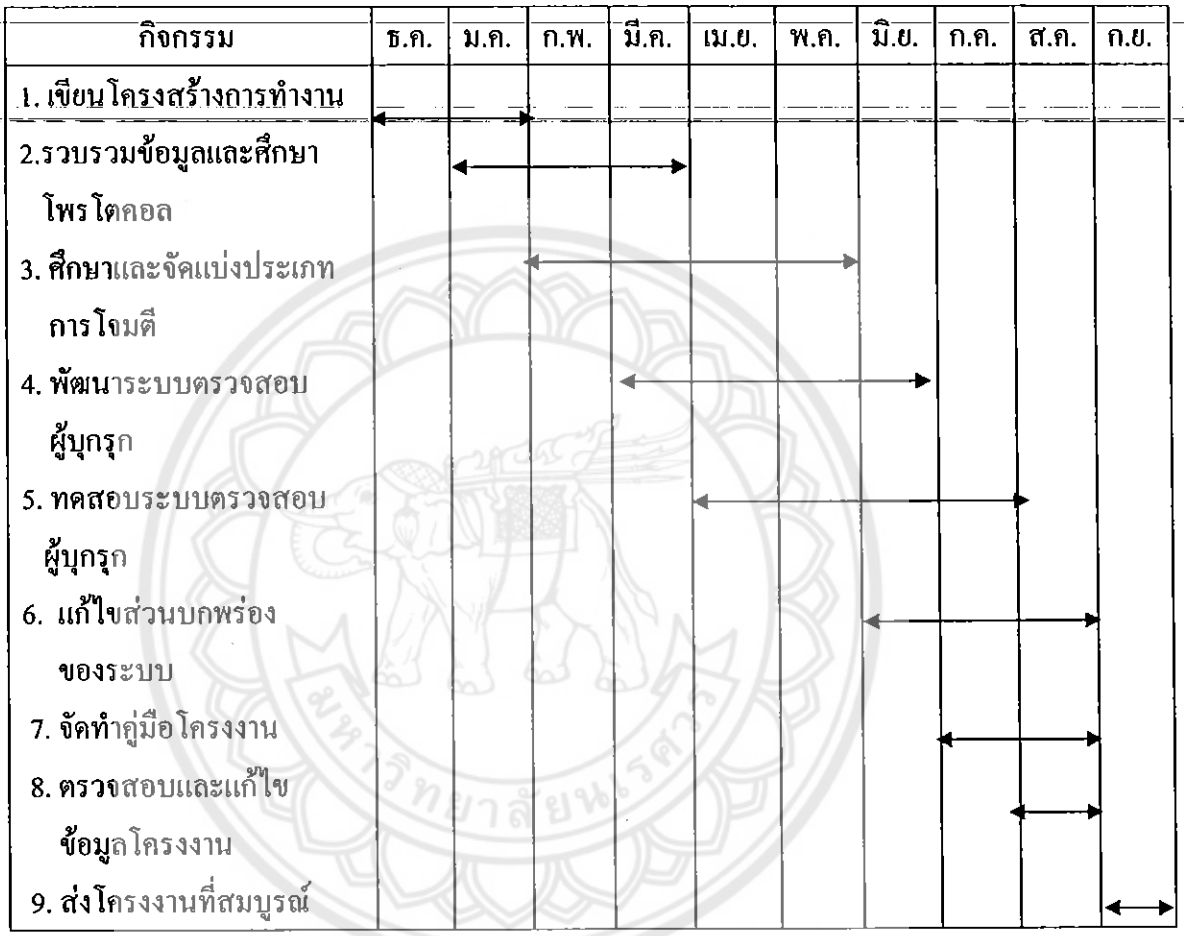
2) พัฒนาระบบที่สามารถตรวจจับ แจ้งเตือน และเก็บข้อมูลไว้วิเคราะห์เกี่ยวกับการเจาะระบบได้อย่างถูกต้องหรือใกล้เคียงที่สุด

3) ระบบที่พัฒนาขึ้นใช้ระบบปฏิบัติการตระกูลยูนิกส์ หรือลีนุกซ์

4) ระบบที่ใช้งานต้องสามารถทำงานข้ามระบบปฏิบัติการได้

1.4 ขั้นตอนการดำเนินงาน

การดำเนินงานของโครงงานระบบตรวจจับผู้บุกรุก มีระยะเวลาการทำงาน ดังนี้



1.5 ผลที่คาดว่าจะได้รับ

1. สามารถสร้างระบบตรวจจับการบุกรุกทางเครือข่ายที่มีประสิทธิภาพและมีเสถียรภาพที่ดี
2. มีการติดตั้งระบบตรวจจับการบุกรุกเพื่อเพิ่มความปลอดภัยให้แก่เครือข่ายคอมพิวเตอร์

1.6 งบประมาณที่ต้องใช้

1. ค่าหนังสือ	1,000 บาท
2. ค่าจ้างถ่ายเอกสาร และจัดรูปเล่มรายงาน	1,200 บาท
3. ค่าวัสดุคอมพิวเตอร์	800 บาท
รวมค่าใช้จ่าย	3,000 บาท (สามพันบาทถ้วน)

บทที่ 2
ทฤษฎีและหลักการระบบเครือข่าย

2.1 ระบบเครือข่าย (Network)

2.1.1 แบ่งปันข้อมูลร่วมกัน

ความสามารถในการใช้ข้อมูลรวมกันอย่างรวดเร็วและราคาถูก ได้รับการพิสูจน์ให้เห็นว่าเป็นหนึ่งในเทคโนโลยีที่ทันสมัยของการใช้ระบบเครือข่าย ซึ่งได้รับการรายงานว่าการใช้จดหมายอิเล็กทรอนิกส์เป็นกิจกรรมอันดับหนึ่งสำหรับผู้ใช้อินเตอร์เน็ต ธุรกิจหลายอย่างได้ลงทุนในการใช้ระบบเครือข่าย โดยเฉพาะอย่างยิ่งเพื่อให้ได้ความได้เปรียบในการใช้จดหมายอิเล็กทรอนิกส์

ระบบเครือข่ายสามารถลดความต้องการการใช้กระดาษสำหรับติดต่อสื่อสารเพิ่มเพื่อความมีประสิทธิภาพและทำให้มีข้อมูลทุกประเภทอย่างต่อเนื่องสำหรับผู้ใช้ทุกคนที่ต้องการ โดยการทำให้ข่าวสารข้อมูลมีพร้อมสำหรับการแบ่งปันการใช้ร่วมกัน ผู้บริหารจะสามารถใช้ความสามารถเหล่านี้ในการติดต่อสื่อสารกับคนจำนวนมากได้อย่างรวดเร็ว มีประสิทธิภาพ และจัดการตารางเวลาการประชุมกับตัวแทนของพนักงานทั้งบริษัท หรือการทำธุรกิจจากระยะไกลก็มีความเป็นไปได้สูงและทำได้ง่ายกว่าแต่ก่อน

2.1.2 การแบ่งปันการใช้ Hardware และ Software ร่วมกัน

ก่อนที่จะมีระบบเครือข่าย ผู้ใช้เครื่องคอมพิวเตอร์จะต้องมีอุปกรณ์ต่าง ๆ ที่จะใช้ เป็นของตนเองเท่านั้นจึงจะสามารถทำงานได้ตามที่ต้องการเช่น พิมพ์งาน

ระบบเครือข่ายทำให้ผู้ใช้หลายๆคนสามารถใช้ข้อมูล และใช้อุปกรณ์เสริมร่วมกันในเวลาเดียวกันได้ ถ้ามีหลายคนต้องการใช้เครื่องพิมพ์เขาเหล่านั้นสามารถใช้เครื่องพิมพ์ที่มีอยู่บนระบบเครือข่ายได้ ระบบเครือข่ายจะสามารถใช้การแบ่งบันการใช้โปรแกรมประยุกต์ร่วมกันเพื่อให้เป็นมาตรฐานเดียวกันได้ เช่น โปรแกรมการจัดการเอกสาร (Word Processing) และอื่นๆ เพื่อให้มีความมั่นใจว่าทุกคนในระบบเครือข่ายใช้โปรแกรมประยุกต์รูปแบบเดียวกัน ซึ่งจะทำให้เอกสารต่างๆสามารถถูกใช้ร่วมกันได้อย่างมีประสิทธิภาพ และจะเป็นการง่ายอย่างยิ่งสำหรับผู้ใช้ในการเรียนรู้การใช้โปรแกรมการจัดการเอกสารเพียงอย่างเดียว แทนที่จะต้องเรียนรู้การใช้โปรแกรมการจัดการเอกสารที่มีอยู่ทั้งหมด 4-5 แบบ

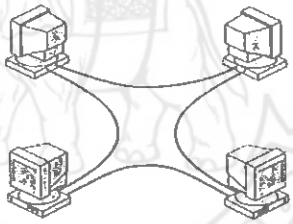
2.1.3 การบริหารจัดการและการสนับสนุนแบบรวมศูนย์กลาง

ทำให้สามารถบริหารจัดการและให้การสนับสนุนระบบเครือข่ายคอมพิวเตอร์ได้ง่ายขึ้น หากใช้ระบบปฏิบัติการเครือข่าย หรือโปรแกรมประยุกต์อันใดอันหนึ่ง และติดตั้งให้เครื่องคอมพิวเตอร์มีลักษณะเหมือนกัน ช่างเทคนิคจะสามารถให้ความช่วยเหลือได้อย่างมีประสิทธิภาพมากกว่าที่จะต้องให้การสนับสนุนระบบที่มีความหลากหลายและเป็นลักษณะเฉพาะ

2.2 ประเภทของระบบเครือข่าย

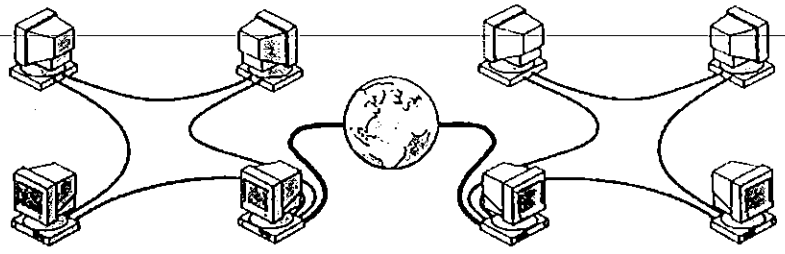
2.2.1 แบ่งตามขนาดและการทำงาน LAN กับ WAN

LAN (Local area Network) เป็นพื้นฐานของการสร้างระบบเครือข่ายคอมพิวเตอร์ ระบบเครือข่ายท้องถิ่นที่มีตั้งแต่อย่างง่าย (คอมพิวเตอร์ 2 เครื่องเชื่อมต่อกันด้วยสายเคเบิล) จนถึงอย่างซับซ้อน (คอมพิวเตอร์และอุปกรณ์เสริมต่างๆ นับร้อยเชื่อมโยงกันผ่านบริษัทใหญ่ๆ) การจำแนกความแตกต่างของระบบ LAN โดยการจำกัดของเขตในทางภูมิศาสตร์



รูปที่ 2.1 ระบบเครือข่ายท้องถิ่น (Local area Network)

WAN (Wide Area Network) ไม่มีการจำกัดขอบเขตทางภูมิศาสตร์ สามารถทำการเชื่อมต่อเครื่องคอมพิวเตอร์และอุปกรณ์อื่นๆ ในซีกโลกตรงข้ามได้ ระบบเครือข่ายอย่างกว้างขวางสร้างขึ้นมาจากการเชื่อมต่อระบบเครือข่ายท้องถิ่นหลายๆ เครือข่ายเข้าด้วยกัน บางทีระบบเครือข่ายอย่างกว้างที่สุดก็คือระบบอินเตอร์เน็ตนั้นเอง

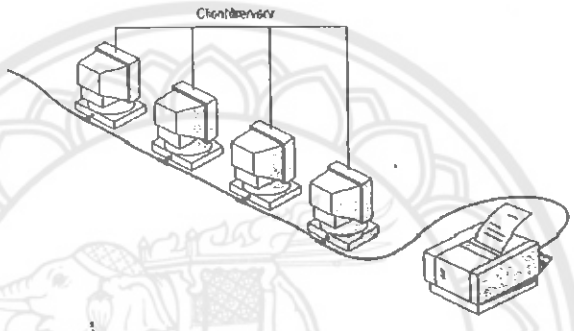


รูปที่ 2.2 ระบบเครือข่ายอย่างกว้าง (Wide Area Network)

2.2.2 แบ่งตามหน้าที่ของเครื่อง คอมพิวเตอร์

ระบบเครือข่ายแบบ Peer – to – Peer

ในระบบเครือข่ายแบบ Peer – to – Peer จะไม่มีเครื่องที่อุทิศให้เป็นเครื่อง Server โดยเฉพาะ ไม่มีการจัดลำดับชั้นของเครื่องคอมพิวเตอร์ทั้งหลาย คอมพิวเตอร์ทั้งหมดจะเท่าเทียมกัน และรู้จักกันในชื่อว่า Peer ฟังก์ชันการทำงานของคอมพิวเตอร์แต่ละเครื่องจะเหมือน เป็นทั้งเครื่อง Server และ client และจะไม่มีผู้บริหารระบบมารับผิดชอบการจัดการระบบเครือข่ายทั้งหมด ผู้ใช้ของแต่ละเครื่องในระบบจะเป็นผู้พิจารณาว่า ข้อมูลใดในเครื่องคอมพิวเตอร์ของตนเอง ที่จะแบ่งปันให้ใช้ร่วมกันในระบบ



รูปที่ 2.3 ระบบเครือข่ายแบบ Peer-to-Peer

ขนาด (SIZE)

ระบบเครือข่ายแบบ Peer – to – Peer เรียกอีกอย่างหนึ่งว่าเป็นแบบ Workgroups บอกเป็นในว่าเป็นกลุ่มคนขนาดเล็ก ซึ่งโดยทั่วไปจะมีเครื่องคอมพิวเตอร์ไม่เกิน 10 เครื่อง ในระบบเครือข่ายแบบ Peer – to – Peer

ราคา (COST)

ระบบเครือข่ายแบบ Peer – to – Peer เป็นระบบอย่างง่าย เพราะว่าคอมพิวเตอร์แต่ละเครื่องมีฟังก์ชั่นการทำงานเหมือนเป็นทั้งเครื่อง Server และ Client จึงไม่มีความต้องการเครื่อง Server ที่มีสมรรถนะสูงเป็นศูนย์กลาง หรือต้องการอุปกรณ์ อื่นในการทำให้เป็นระบบเครือความจุสูง ดังนั้น ระบบเครือข่ายแบบ Peer – to – Peer จึงมีราคาถูกกว่าระบบเครือข่ายแบบ Server – base

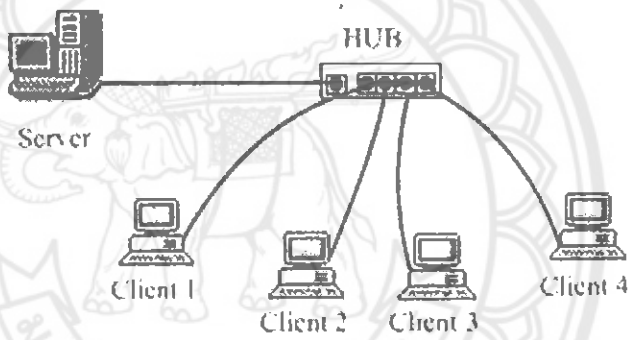
ระบบปฏิบัติการ (OPERATING SYSTEM)

ในระบบเครือข่ายแบบ Peer – to – Peer ไม่มีความต้องการโปรแกรมจัดการระบบเครือข่ายที่เป็นมาตรฐานเดียวกันในการทำงาน และการจัดการระบบรักษาความปลอดภัย เช่นเดียวกับโปรแกรมจัดการระบบเครือข่ายที่ออกแบบมาสำหรับเครื่องที่อุทิศให้เป็น Server จะมีฟังก์ชันการ

ทำงานเป็นเครื่องให้บริการเพียงอย่างเดียว และไม่สามารถใช้เป็นเครื่องลูกข่ายหรือเครื่อง Workstation ได้

ระบบเครือข่ายแบบ Server – based

ในสภาวะแวดล้อมที่มีผู้ใช้มากกว่า 10คนระบบเครือข่ายแบบ Peer–to–Peer ที่คอมพิวเตอร์แต่ละเครื่องทำหน้าที่เป็นทั้ง Server และ Client อาจจะไม่เพียงพอ ดังนั้นระบบเครือข่ายส่วนใหญ่จะจัดให้มีเครื่องคอมพิวเตอร์ที่อุทิศให้เป็นเครื่อง Server คือทำหน้าที่เป็นผู้ให้บริการหรือเป็นแม่ข่ายเพียงอย่างเดียว ไม่สามารถเป็นเครื่องลูกข่ายหรือเครื่อง Workstation ได้ การที่เครื่อง Server ถือว่าเป็นเครื่องที่ต้องอุทิศให้กับระบบ เนื่องจากจะไม่สามารถทำงานเป็นเครื่องลูกข่ายได้ และจากการที่เครื่อง Server เหล่านั้นจะให้บริการตามคำร้องขอของเครื่องลูกข่ายในระบบเครือข่ายให้ได้ประโยชน์สูงสุดอย่างรวดเร็วและทำให้มีความมั่นใจในระบบการรักษาความปลอดภัยของ File และ Directory ระบบเครือข่ายแบบ Server –Based



รูปที่ 2.4 ระบบเครือข่ายแบบ Server-based

การที่ระบบเครือข่ายใหญ่ขึ้น (จำนวนของการเชื่อมต่อเครื่องคอมพิวเตอร์และระยะทางกายภาพกับการรับส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์เหล่านั้น เติบโตขึ้น) ทำให้มีความต้องการเครื่อง Server มากกว่า 1 เครื่อง การกระจายการปฏิบัติงานในระบบเครือข่ายไปยังเครื่อง Server หลายเครื่อง ทำให้มั่นใจได้ว่าการปฏิบัติงานแต่ละอย่างจะทำได้อย่างมีประสิทธิภาพมากเท่าที่จะเป็นไปได้

เครื่อง Server

การที่เครื่อง Server ต้องปฏิบัติงานอย่างหลากหลายและซับซ้อน สำหรับระบบเครือข่ายขนาดใหญ่เครื่อง Server เริ่มที่จะมีความพิเศษเพื่อรองรับการขยายความต้องการของผู้ใช้

ตัวอย่างของเครื่อง Server

File and Print Server จะจัดการผู้ใช้ในการ access และใช้ทรัพยากรไฟล์และเครื่องพิมพ์ ตัวอย่างเช่น เมื่อใช้โปรแกรมการจัดการเอกสาร (Word Processing) โปรแกรมจะ Run บนเครื่อง เอกสารที่ถูกเก็บอยู่ในเครื่อง File and Print Server จะถูก Load เข้ามาในหน่วยความจำหลักในเครื่องเพื่อที่จะสามารถทำการแก้ไขหรือเรียบเรียงได้ หรืออาจกล่าวว่า เครื่อง File and Print Server ถูกใช้เป็นคลังสำหรับจัดเก็บ File และข้อมูล

Application Server จะทำงานทางด้าน Server ให้กับโปรแกรมประยุกต์ที่มีลักษณะ Client/Server เช่นเดียวกับการทำให้มีข้อมูลในเครื่อง Client ตัวอย่างเช่น เครื่อง Server จะจัดข้อมูลจำนวนมาก และทำการจัดการให้ง่ายต่อการเรียกกับมาใช้งาน ดังนั้นเครื่อง Application Server จึงต่างจากเครื่อง File and Print Server ข้อมูลหรือไฟล์จะถูก Download มายังเครื่องคอมพิวเตอร์ที่ทำการร้องขอ แต่ด้วยการทำงานของเครื่อง Application Server แล้วฐานข้อมูลทั้งหมดยังคงอยู่บนเครื่อง Server จะมีเพียงผลของการร้องขอเท่านั้นที่จะถูก Download มายังเครื่องที่ใช้งาน

Mail Server จะมีการทำงานเหมือนเครื่อง Application Server ในลักษณะของการที่มี Server Application และ Client Application จะมีเพียงข้อมูลที่ถูกเลือกเท่านั้นที่จะ Download จากเครื่อง Server ไปยังเครื่อง Client

Fax Server จะจัดการจราจรของสัญญาณ Fax ที่ผ่านเข้าและออกจากระบบเครือข่าย โดยการแบ่งบันการใช้ Fax Modem boards ร่วมกัน

Communication Server จะจัดการไหลของข้อมูลระหว่างข่าวสารทางจดหมายอิเล็กทรอนิกส์ ระหว่างเครื่อง Server ของระบบเครือข่ายของเรากับระบบเครือข่ายอื่น เช่น เครื่องคอมพิวเตอร์ Mainframe หรือผู้ใช้ทางไกลที่หมุนโทรศัพท์เข้ามายังเครื่อง Server ผ่านทาง Modem

ตารางที่2.1 เปรียบเทียบเครือข่าย Peer -to -Peer กับ Server Based

ข้อพิจารณา	ระบบเครือข่ายแบบ Peer -to -Peer	ระบบเครือข่ายแบบ Server Based
ขนาด	ดีสำหรับเครื่องคอมพิวเตอร์สำหรับ 10 เครื่อง	มีข้อจำกัดโดยเครื่อง Server และ Hardware ระบบเครือข่าย
การรักษาความปลอดภัย	จัดตั้งโดยผู้ใช้เครื่องคอมพิวเตอร์แต่ละคน	มีการรักษาความปลอดภัยอย่างกว้างขวางในทรัพยากรเดียวกันและการรักษาความปลอดภัยของผู้ใช้
การบริการการจัดการ	เป็นความรับผิดชอบของผู้ใช้แต่ละคน สำหรับการบริหารจัดการของตนเองไม่มีความจำเป็นต้องใช้ผู้บริหารระบบเต็มเวลา	การควบคุมระบบเครือข่ายถูกรวมศูนย์กลาง โดยต้องการผู้บริการระบบที่มีความรู้อย่างน้อย 1 คน

2.3 Topology ระบบเครือข่าย

ความหมายของTopology

คำว่า Topology หรือคำว่า Network Topology จะกล่าวถึงการจัดรูปแบบการเชื่อมต่อหรือโครงสร้างทางกายภาพของเครื่องคอมพิวเตอร์ สายเคเบิลและอุปกรณ์อื่นๆ บนระบบเครือข่าย คำว่า " Topology " เป็นคำมาตรฐานที่ผู้เชี่ยวชาญระบบเครือข่ายส่วนมากใช้อ้างอิงในการออกแบบระบบเครือข่ายพื้นฐาน การออกแบบระบบเครือขายทั้งหมดล้วนเกิดมาจากระบบเครือข่ายมาตรฐานทั้งหมด4 แบบ

2.3.1 แบบ Bus

รูปที่ 2.5 ระบบเครือข่ายที่ใช้ Topology แบบ Bus

Topology แบบบัส จะถูกอ้างอิงในลักษณะ Liner bus เพราะว่าเครื่องคอมพิวเตอร์ทั้งหมดถูกต่อเชื่อมในแนวเส้นตรง เป็นวิธีที่ง่ายและธรรมดาที่สุดของการต่อเชื่อมระบบเครือข่ายคอมพิวเตอร์ รูปที่ 2.5 แสดง Topology แบบ Bus โดยทั่วไป ซึ่งประกอบด้วยเคเบิลเส้นเดี่ยวเรียกว่า Trunk (หรือเรียกว่า backboneหรือ segment) ที่คอมพิวเตอร์ทั้งหมดในระบบเครือข่ายถูกต่อเข้ากับสายเคเบิลนี้เพียงเส้นเดียว

การติดต่อสื่อสารบน Bus

คอมพิวเตอร์ระบบเครือข่ายที่ใช้ Topology แบบ Bus จะติดต่อสื่อสารกันโดยกำหนดการจ่าหน้าให้ข้อมูลถูกส่งไปยังเครื่องคอมพิวเตอร์ที่เฉพาะเจาะจง และและส่งข้อมูลนั้นไปบนสายเคเบิลในรูปแบบของสัญญาณทางไฟฟ้า การที่จะเข้าในการติดต่อสื่อสารบน Bus คุณจะต้องทำความคุ้นเคยกับแนวคอก 3 ประการ คือ

- การส่งสัญญาณ (Sending of Signal)
- การสะท้อนกลับของสัญญาณ (Signal Bounce)
- ตัวสิ้นสุดปลายทาง (Terminator)

การส่งสัญญาณ ข้อมูลในระบบเครือขายในรูปแบบของสัญญาณทางไฟฟ้า จะถูกส่งไปยังคอมพิวเตอร์ทุกเครื่องบนระบบเครือข่าย แต่จะมีคอมพิวเตอร์เพียงเครื่องเดียวที่มีที่อยู่ตรงกับการจ่าหน้าเท่านั้นที่ได้รับข่าวสารนั้น คอมพิวเตอร์อื่นนอกเหนือจากนี้ทั้งหมดจะปฏิเสธการรับข้อมูล เพราะว่ามีเพียงคอมพิวเตอร์เครื่องเดียว ในขณะหนึ่งที่สามารถส่งข้อมูลไปบนระบบเครือข่ายแบบ Bus ได้ จำนวนเครื่องคอมพิวเตอร์ที่เชื่อมต่อในระบบ จึงมีผลกระทบต่อประสิทธิภาพการทำงานของเครือข่าย ยิ่งมีเครื่องคอมพิวเตอร์มากเท่าใดบนระบบเครือข่าย ก็จะมีคอมพิวเตอร์จำนวนมากที่รอจะส่งข้อมูลไปบน Bus จึงเป็นผลทบให้การทำงานของระบบเครือข่ายช้าลง

ยังไม่มีมาตรฐานในการวัดว่าผลกระทบที่เกิด ว่าจำนวนเครื่องคอมพิวเตอร์เท่าใดจะทำให้ความเร็วในการทำงานของระบบเครือข่ายเป็นอย่างไร ผลกระทบต่อประสิทธิภาพการทำงานของระบบเครือข่ายไม่เป็นเอกเทศต่อจำนวนเครื่องคอมพิวเตอร์ ทั้งนี้มีปัจจัยอื่นนอกเหนือจากจำนวนคอมพิวเตอร์ในระบบเครือข่ายที่มีผลกระทบต่อประสิทธิภาพการทำงานของระบบเครือข่าย

- ความสามารถของ Hardware เครื่องคอมพิวเตอร์บนระบบเครือข่าย
- จำนวนรอบของชุดคำสั่งที่รอปฏิบัติ
- ประเภทของโปรแกรมประยุกต์ (ตัวอย่างเช่น Client-Server หรือระบบการแบ่งปันการใช้ไฟล์ร่วมกัน) ที่ทำงานอยู่บนระบบเครือข่าย
- ประเภทของสายเคเบิลที่ใช้ในระบบเครือข่าย
- ระยะห่างระหว่างเครื่องคอมพิวเตอร์บนระบบเครือข่าย

เครื่องคอมพิวเตอร์บน Bus สามารถส่งข้อมูลไปยังเครื่องคอมพิวเตอร์เครื่องอื่นบนระบบเครือข่ายหรือเฝ้ารอรับข้อมูลจากคอมพิวเตอร์เครื่องอื่นบนระบบเครือข่าย แต่ไม่มีหน้าที่รับผิดชอบในการเคลื่อนย้ายข้อมูลจากคอมพิวเตอร์จากตัวหนึ่งไปยังอีกตัวหนึ่งดังนั้นหากมีเครื่องใดเครื่องหนึ่งในระบบเสียก็จะไม่ส่งผลกระทบต่อเครื่องอื่นๆ

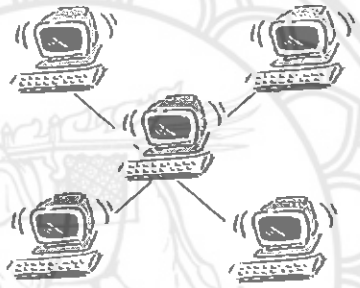
การสะท้อนกลับของสัญญาณเนื่องจากข้อมูลหรือสัญญาณทางไฟฟ้าจะถูกส่งไปทั่วทั้งระบบของเครือข่ายจึงมีการเดินทางจากปลายด้านหนึ่งไปยังปลายอีกด้านหนึ่ง ถ้าสัญญาณนั้นยังคงไม่ถูกขัดขวาง สัญญาณก็คงสะท้อนไปตามสายเคเบิลและจะป้องกันไม่ให้เครื่องคอมพิวเตอร์เครื่องอื่นส่งสัญญาณออกมาได้ ดังนั้นสัญญาณนั้นจึงต้องทำให้ถูกหยุดลงหลังจากที่มีโอกาสที่จะไปถึงที่อยู่ปลายทาง

ตัวสิ้นสุดปลายทาง (Terminator) การที่จะหยุดการสะท้อนไปมาของสัญญาณ จะมีอุปกรณ์ที่เรียกว่า ตัวสิ้นสุดปลายทาง (Terminator) อยู่ที่ปลายทางของสายเคเบิลแต่ละด้านเพื่อดูดซึมสัญญาณอิสระการดูดซึมสัญญาณนี้จะทำให้สายเคเบิลว่าเพื่อให้คอมพิวเตอร์เครื่องอื่นสามารถส่งสัญญาณได้

ในกรณีอื่นๆเช่น ต้องการต่อสายเคเบิล ก็จะมีอุปกรณ์เชื่อมต่อ มีลักษณะรูปทรงกระบอก (Barrel Connector) นำมาใช้ในการเชื่อมต่อสายเคเบิล 2 ส่วนเขาด้วยกัน อย่างไรก็ตามตัวต่อนี้จะทำให้สัญญาณอ่อนลงจึงควรนำมาใช้อย่างมัธยัสถ์ เพราะว่าสายเคเบิลเส้นเดียวย่อมดีกว่าสาย 2 เส้นมาต่อกันพูดถึงในกรณีที่สัญญาณอ่อนลงก็มีอุปกรณ์ที่เรียกว่า Repeater (ตัวทวนสัญญาณซ้ำ) สามารถนำมาใช้ในการต่อสายเคเบิล 2 เส้น ที่จริงแล้ว Repeater จะเพิ่มสัญญาณก่อนที่จะส่งออกไปตามวิถีทางของมัน

2.3.2 แบบ Star

ใน Topology แบบ Star ส่วนของคอมพิวเตอร์แต่ละเครื่องจะต่อกับอุปกรณ์ศูนย์กลางที่เรียกว่า Hub รูปที่ 2.6 แสดงการต่อแบบ Star โดยใช้ Hub ไปยังเครื่องคอมพิวเตอร์ทุกเครื่องบนระบบเครือข่าย Topology แบบ Star

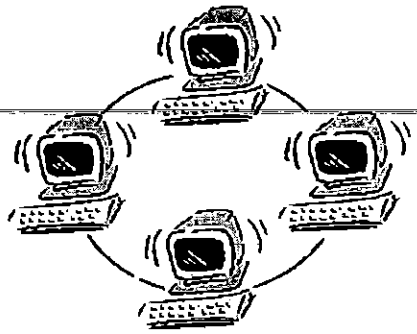


รูปที่ 2.6 ระบบเครือข่ายแบบ Star

ระบบเครือข่ายแบบ Star ให้มีข้อได้เปรียบของการรวมทรัพยากรและการบริหารจัดการเข้าสู่ศูนย์กลาง อย่างไรก็ตาม เนื่องจากคอมพิวเตอร์แต่ละเครื่องถูกต่อเข้ากับจุดศูนย์กลาง Topology แบบนี้จึงใช้สายเคเบิลจำนวนมากอีกทั้งถ้าจุดศูนย์กลางล้มเหลวระบบก็พังไปด้วย

2.3.3 แบบ Ring

Topology แบบ Ring จะเชื่อมต่อคอมพิวเตอร์ในรูปวงกลมวงเดียว จึงไม่เหมือนกับ Topology แบบ Bus ที่ไม่มีปลายทางที่เป็นจุดสิ้นสุด สัญญาณจะเดินทางไปรอบๆวงในทิศทางเดียวกันผ่านเครื่องคอมพิวเตอร์ซึ่งทำหน้าที่เสมือน Repeater ที่เพิ่มขนาดสัญญาณก่อนที่จะส่งออกไปยังเครื่องคอมพิวเตอร์ถัดไป รูปที่ 2.7 แสดงการต่อแบบ Star ที่มี Server 1 เครื่อง และ Workstation 4 เครื่อง เนื่องจากต่อแบบ Star จึงมีผลทำให้ถ้าเครื่องใดเครื่องหนึ่งใช้ไม่ได้ก็จะเสียทั้งระบบ



รูปที่ 2.7 ระบบเครือข่ายแบบ Ring

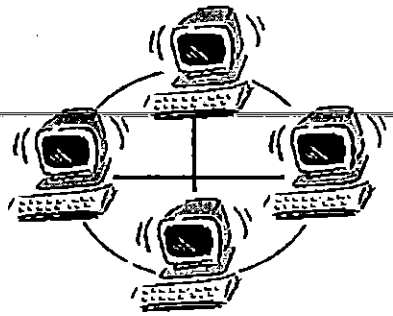
Token Passing

วิธีการหนึ่งในการส่งข้อมูลไปรอบๆ วงแหวน เรียกว่า Token Passing (Token เป็นลำดับพิเศษของ bit ที่เดินทางไปรอบๆระบบเครือข่ายแบบ Token – Ring ซึ่งแต่ละระบบเครือข่ายจะมีเพียง 1 Token) จะถูกส่งผ่านจากคอมพิวเตอร์เครื่อง 1 ไปยังอีกเครื่อง 1 จนกระทั่งถึงเครื่องคอมพิวเตอร์ที่ข้อมูลถูกส่งไปถึง รูปที่ 2.8 แสดง Topology แบบToken – Ring เครื่องคอมพิวเตอร์ที่ทำการส่งจะปรับปรุง Token โดยการใส่การจ่าหน้าแบบอิเล็กทรอนิกส์ลงบนข้อมูลแล้วส่งไปรอบๆ วงแหวนข้อมูลจะถูกส่งโดยเครื่องคอมพิวเตอร์แต่ละเครื่องจนกระทั่งพบเครื่องที่มีที่อยู่ตรงกับที่จ่าหน้าไว้

เครื่องคอมพิวเตอร์ที่รับข่าวสารแล้ว ก็จะทำการส่งข่าวสารกลับไปยังเครื่องที่ส่งข้อมูลมาให้ ให้ทราบว่าได้รับข้อมูลเรียบร้อยแล้วหลังจากการตรวจสอบความเป็นจริงแล้วเครื่องคอมที่จะทำการส่งข้อมูลก็จะสร้าง Token ใหม่ แล้วปล่อยสู่ระบบเครือข่าย Token จะไหลวนจนกระทั่งเครื่อง Workstation ต้องการใช้ในการส่งข้อมูล

2.3.4 แบบ Mesh

ระบบเครือข่ายที่ใช้ Topology แบบ Mesh แสนอให้มีการทำซ้ำกันมากๆและมีความอ่อนตัวสูงใน Topology แบบ Mesh คอมพิวเตอร์จะถูกเชื่อมต่อกับเครื่องอื่นๆ ทั้งหมดสายเคเบิลที่แยกต่างหากโครงสร้างแบบนี้จัดให้มีทางเดินของเส้นทางซ้ำซ้อนกันทั่วทั้งระบบเครือข่ายหากสายเคเบิลเส้นใดเสียที่เหลือก็จะทำหน้าที่แทน อีกทั้งมีความง่ายต่อการแก้ปัญหาและเพิ่มความคล่องตัวจึงมีความหมายในทางบวก ระบบเครือค่ายมักมีราคาแพงในการจัดตั้ง เนื่องจากต้องใช้สายเคเบิลจำนวนมาก บ่อยครั้งที่มีการใช้ Topology แบบ Mesh ร่วมกับ Topology แบบอื่น ในการจัดสร้าง Topology แบบ Hybrid



รูปที่ 2.8 ในTopology แบบ Mesh เครื่องคอมพิวเตอร์จะถูกต่อเชื่อมเข้ากับเครื่องอื่นๆทั้งหมดโดยใช้ สายเคเบิลที่แยกต่างหาก

การเลือก Topology มีปัจจัยที่ต้องนำมาพิจารณาในการตกลงว่า Topology แบบใดที่มีความเหมาะสมกับความต้องการขององค์กร

ตารางที่ 2.2 ข้อได้เปรียบและเสียเปรียบของ Topology แบบต่างๆ

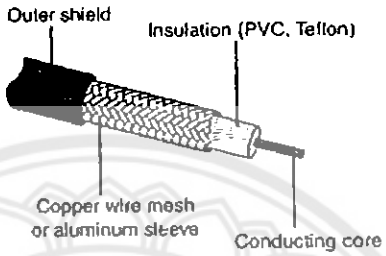
Topology	ข้อได้เปรียบ	ข้อเสียเปรียบ
Bus	ใช้สายเคเบิลอย่างประหยัด สื่อมีราคาไม่แพงและง่ายต่อการวางสาย เป็นระบบอย่างง่ายและเชื่อถือได้ง่ายต่อการขยาย Bus	การทำงานของระบบเครือข่ายช้าหากมีการส่งข้อมูลหนาแน่น ยากที่จะแยกแยะปัญหาที่จะเกิดขึ้น การแตกแยกของสายเคเบิลจะมีผลต่อผู้ใช้หลายคน
Ring	ระบบจัดให้มีมีการเข้าถึงอย่างเท่าเทียมกันโดยคอมพิวเตอร์ทุกเครื่อง การดำเนินการของระบบไม่คำนึงว่ามีผู้ใช้เท่าใด	ความเสียหายของคอมพิวเตอร์เครื่องหนึ่งจะมีผลกระทบต่อส่วนที่เหลือ ยากที่จะแยกแยะปัญหาที่เกิดขึ้น
Star	ง่ายต่อการปรับปรุงและเพิ่มเครื่องคอมพิวเตอร์ในระบบ สามารถตรวจสอบและบริหารจัดการระบบโดยศูนย์กลางได้ ความเสียการของเครื่องคอมพิวเตอร์เครื่องหนึ่ง ไม่มีผลกระทบต่อส่วนที่เหลือของระบบ	การปรับโครงสร้างระบบเครือข่ายจะรบกวนการปฏิบัติงาน การปรับโครงสร้างระบบเครือข่ายจะรบกวนการปฏิบัติงาน ถ้าจุดศูนย์รวมของระบบล้มเหลว ระบบจะล้มทั้งหมด
Mesh	จัดให้มีการทำซ้ำเพิ่มมากขึ้น และมีความเชื่อถือได้ เช่นเดียวกับการง่ายต่อการแก้ไข	การจัดตั้งระบบมีราคาเพราะว่าต้องใช้เคเบิลจำนวนมาก

2.4 อุปกรณ์ในระบบเครือข่าย

2.4.1 สายเคเบิลระบบเครือข่าย

- สายโคแอกเชียล (Coaxial Cable)

สายโคแอกเชียลเคยเป็นสายที่ใช้กันอย่างกว้างขวางในระบบเครือข่ายเนื่องจากราคาถูก น้ำหนักเบามีความอ่อนตัวง่ายในการเดินสาย



รูปที่ 2.9 สายโคแอกเชียลที่แสดงให้เห็นชั้นใน

Shield จะปกป้องสัญญาณที่ทำการส่งโดยการดูดซับสัญญาณอิเล็กทรอนิกส์อื่นๆ ที่เรียกว่า Noise เพื่อไม่ให้เข้าไปยังแกนกลาง ของสายเคเบิลเพราะจะทำลายข้อมูลได้ สายเคเบิลที่มีชั้นของฉนวนกระดาษตะกั่ว 1 ชั้น ของสายถักโลหะ 1 ชั้น เรียกว่าเป็นสายชีลด์ 2 ชั้น สำหรับสภาวะแวดล้อมที่มีการรบกวนสูงมากขึ้น ก็จะมีชีลด์ 4 ชั้น ซึ่งประกอบด้วยฉนวนกระดาษตะกั่วหุ้ม 2 ชั้น และสายถักโลหะหุ้ม 2 ชั้นแกนกลางของสายโคแอกเชียลทำหน้าที่ส่งสัญญาณอิเล็กทรอนิกส์ที่เป็นข้อมูล สายที่เป็นแกนกลางสามารถเป็นได้ทั้งแบบแข็งหรือแบบเกลียว ถ้าแกนกลางเป็นแบบแข็งส่วนใหญ่จะเป็นทองแดง

โดยรอบแกนกลางเป็นชั้นของฉนวนที่ไม่นำไฟฟ้า ซึ่งแยกแกนกลางออกจากตาข่ายถักตาข่ายถักนี้จะทำหน้าที่เหมือนสายดิน และปกป้องแกนกลางจากสัญญาณรบกวนและ Crosstalk (Crosstalk เป็นสัญญาณส่วนเกินที่เกิดจากการที่สายไปอยู่ใกล้กัน)

สายโคแอกเชียล มี 2 ประเภท

1. สายแบบบาง (Thinnet)

เป็นสายโคแอกเชียลที่มีความอ่อนตัว มีความหนาประมาณ 0.64 เซนติเมตร (0.25นิ้ว) เนื่องจากสายโคแอกเชียลประเภทนี้ สามารถโค้งงอได้และง่ายต่อการเดินสาย จึงสามารถใช้ได้ในระบบการติดตั้งแบบทุกประเภท

2. สายแบบหนา (Thicknet)

เป็นสายโคแอกเชียลแบบแข็งมีเส้นผ่านศูนย์กลางประมาณ 1.27 เซนติเมตร แกนกลางของแบบหนาจะหนากว่าทำให้ส่งสัญญาณได้ไกลขึ้นจึงหมายความว่าสายแบบหนา จึงส่งสัญญาณได้ไกลกว่าแบบบางสายแบบหนาส่งสัญญาณได้ประมาณ 500 เมตร สายเคเบิลแบบบางและสายเคเบิลแบบหนา โดยทั้งไปสายเคเบิลที่มีความหนามากจะเดินสายได้ยากกว่าสายเคเบิลแบบบางที่มีความอ่อนตัวมาก สายเคเบิลแบบหนาไม่สามารถโค้งงอได้โดยง่าย นี้คือสิ่งที่ต้องนำมาพิจารณาในการเดินสายที่ต้องดึงสายเคเบิลผ่านพื้นที่ที่ยากลำบาก เช่น ท่อหุ้มสายใต้ดินและรางน้ำ สายแบบหนามีราคาสูงกว่าแบบบางแต่ก็สามารถนำสัญญาณไปได้ไกลกว่า

รูปที่ 2.10 สายแบบหนาที่มีแกนกลางหนากว่าสายแบบบาง

- สาย UTP (Unshield Twisted – Pair)

UTP ใช้คุณลักษณะเฉพาะ 10 Base T เป็นประเภทของสายคู่พันเกลียวที่ได้รับ ความนิยมสูงสุดที่ใช้ในการเดินสายระบบเครือข่ายท้องถิ่น (LAN) ความยาวสูงสุดของส่วนของสายเคเบิล คือ 100 เมตร หรือ 328 ฟุต

ประเภทของสาย UTP

- ประเภทที่ 1 เป็นสาย UTP ที่ใช้เป็นสายเคเบิลโทรศัพท์แบบดั้งเดิมที่ซึ่งจะสามารถนำสัญญาณเสียงได้แต่ไม่รวมถึงการส่งข้อมูลสายโทรศัพท์เกือบทั้งหมด

- ประเภทที่ 2 สนับสนุนการส่งข้อมูลของสาย UTP ถึง 4 Mbps ซึ่งประกอบด้วยสายทองแดงพันเกลียว 4 คู่

- ประเภทที่ 3 สนับสนุนการส่งข้อมูลของสาย UTP ถึง 16 Mbps ซึ่งประกอบด้วยสายทองแดงพันเกลียว 4 คู่ โดยทำเลียว 3 รอบต่อ1 ฟุต

- ประเภทที่ 4 สนับสนุนการส่งข้อมูลของสาย UTP ถึง 20 Mbps ซึ่งประกอบด้วยสายทองแดงพันเกลียว 4 คู่ โดยพันเป็นเกลียวสายทองแดง 4 รอบ เข้ากับสายเกลียวที่เหลือ 3 คู่ต่อฟุต

- ประเภทที่ 5 สนับสนุนการส่งข้อมูลของสาย UTP ถึง 100 Mbps ซึ่งประกอบด้วยสายทองแดงพันเกลียว 4 คู่ โดยพันเป็นเกลียวสายทองแดง 4 รอบ

2.4.2 การ์ดระบบเครือข่าย (NIC - Network Interface Card)

การ์ดระบบเครือข่ายจัดเตรียมให้มีการต่อเชื่อมระหว่างสายเคเบิลกับเครื่องคอมพิวเตอร์ ทำหน้าที่เป็นตัวเชื่อมทางการภาพระหว่างเครื่องคอมพิวเตอร์กับสายเคเบิลของระบบเครือข่าย จากรูปที่ 2.14 แสดง NIC ที่มีหัวเชื่อมต่อสำหรับสายโคแอกเชียล การ์ดได้รับการติดตั้งใน Expansion slot ของเครื่องคอมพิวเตอร์แต่ละเครื่องและเครื่อง Server ในระบบเครือข่าย หลังจากติดตั้ง NIC แล้ว สายเคเบิลของระบบเครือข่ายจะถูกต่อเข้ากับ Port ของการ์ดนั้นเพื่อสร้างการเชื่อมต่อทางการภาพระหว่างเครื่องคอมพิวเตอร์กับส่วนของระบบเครือข่าย

2.4.3 อุปกรณ์เชื่อมต่อในระบบเครือข่าย

10 - 100 Base T Hub

เป็นอุปกรณ์รวมสายตามมาตรฐาน 802.3 เพื่อเชื่อมโยงระบบ 802.3 แบบ Star ลักษณะการเชื่อมโยงทำให้สายแบบ UTP โดยแต่ลพเส้นมีความยาว 100 เมตร การขยายพอร์ตทำได้จำนวนมากไม่จำกัด เหมือน 10 Base 2 และถ้า Workstation มีปัญหาก็จะไม่ทำให้ระบบล้มเหลว

Printer Server

เป็นอุปกรณ์เชื่อมต่อกับเครือข่ายเพื่อทำให้การต่อเครื่องพิมพ์เข้ากับเครือข่ายได้หลายเครื่องในการใช้งาน ผู้ที่ใช้อยู่บนเครือข่ายสามารถเลือกใช้เครื่องพิมพ์ใดก็ได้โดยการส่งแฟ้มออกมาพิมพ์ พรินเตอร์เซอร์ฟเวอร์มีบัฟเฟอร์เพื่อจัดคิวได้

CD-ROM Server

เป็นอุปกรณ์อ่าน CD-ROM เพื่อเป็นฐานข้อมูลกลาง เพื่อใช้เครือข่ายเชื่อมกับตัวอ่าน CD-ROM ผู้ใช้ในเครือข่ายสามารถเรียกค้นข้อมูลจากฐานข้อมูล CD-ROM ได้ ปกติ CD-ROM Server จะประกอบด้วยตัวอ่าน CD-ROM ได้หลายแผ่น เพื่อสร้างเป็นฐานข้อมูลขนาดใหญ่

Repeater

เป็นอุปกรณ์เพื่อใช้ในการเปลี่ยนตัวกลางนำสัญญาณจากตัวกลางหนึ่ง เช่น จากไฟเบอร์ ออปติก มายังโคแอกเชียล หรือการเชื่อมต่อระหว่างตัวกลางเดียวกันก็ได้การใช้รีพีตเตอร์จะทำให้เครือข่ายทั้ง 2 ข้างเสมือนเชื่อมกัน โดยสัญญาณจะวิ่งทะลุถึงกันได้หมด รีพีตเตอร์จึงไม่มีการกันข้อมูล แต่จะมีประโยชน์ในการเชื่อมต่อให้ได้ความยาวมากขึ้น

Bridge

มีลักษณะคล้ายรีพีตเตอร์ แต่จะกันสัญญาณระหว่างอุปกรณ์ในแต่ละ เซกเมนต์ออกจากกัน บริดจ์จึงทำให้การเชื่อมต่อระหว่างเครือข่ายมีประสิทธิภาพมากขึ้น ลดการชนกันของข้อมูลลงไป บริดจ์จึงเป็นสะพานสำหรับข้อมูล 2 เครือข่าย

2.5 ROUTER

Router เป็นอุปกรณ์ใน Interconnects network ทำหน้าที่หาเส้นทางที่ดีที่สุด Router ใช้ Logical addressและRouter จะกำหนดเส้นทางของข้อมูลโดยที่ Router จะเก็บตารางของเส้นทาง นอกจากนี้ router ยังเป็นตัวเชื่อมกับภายนอกและต่อลง Switching hub อีกที่หนึ่ง ซึ่ง Router กับ Switching จะทำงานร่วมกัน

2.5.1 หน้าที่การทำงานของ Router

Routing Techniques และ Protocol

เป็นการถ่ายทอดแบบ Connectionless หมายความว่า Router สามารถให้บริการการจัดการรับส่งข้อมูลระหว่างคอมพิวเตอร์หรือเครือข่ายโดยไม่ต้องมีการสร้างการเชื่อมต่อระหว่างเครื่องคอมพิวเตอร์หรือเครือข่ายเนื่องจากการสร้างการเชื่อมต่อนั้นกระทำที่โปรโตคอลระหว่างคอมพิวเตอร์ทั้ง 2 ฝ่ายแล้ว

End System Protocol

ถูกนำมาใช้เพื่อการถ่ายเทข้อมูลภายใต้โปรโตคอลในระดับ Layer 3 และระดับ Layer 4 โดยที่โพรโตคอลในแต่ละแบบจะทำหน้าที่ในการดูแลขั้นตอนการแลกเปลี่ยนข้อมูลระหว่างคอมพิวเตอร์โดยที่ Router จะทำหน้าที่รับและส่งข้อมูลอย่างเดียว

Intermediate System Protocol

ทำหน้าที่สนับสนุนการจัดการการเลือกเส้นทางสำหรับอุปกรณ์ที่ดูแลการการจัดเลือกเส้นทาง โปรโตคอลที่ใช้ในการเลือกเส้นทางระหว่าง Router สามารถแบ่งออกเป็น 2 ระดับคือ

- Distance Vector
- Link state

การใช้ Router เพื่อดูแลเกี่ยวกับการรับส่งข้อมูล

- Data Filtering
- Data Forwarding

2.5.2 ชนิดของ Router

แบ่ง Router ตามลักษณะการเชื่อมต่อได้ 3 แบบคือ

Interior Router

เป็น Router ที่มีไว้เพื่อการเชื่อมต่อระหว่างเครือข่ายภายในองค์กรที่อยู่ติดกัน นอกจากนี้ Interior Router อาจเป็นไปในรูปแบบ Server Based คือการนำเอาเครื่อง Server มาติดตั้ง Lan Card หลายๆ ตัวจากนั้นทำการกำหนด Subnet เพื่อให้เชื่อมต่อกัน

Exterior Router

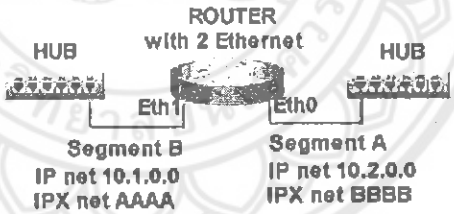
เป็น Router ที่มีไว้เพื่อการเชื่อมต่อกับเครือข่ายหลัก ได้แก่การเชื่อมต่อผ่าน WAN และมีการใช้ Address ของเครือข่ายที่ต่างกัน โดยที่ Router จะทำการเชื่อมต่อกันได้นั้นจะต้องใช้ Static Routing หรือ Dynamic Routing

Border Router

เป็น Router ที่ทำหน้าที่เป็นตัวแทนของ Router ของ Router ต่างๆ ที่เชื่อมต่อกันเป็นเครือข่าย Router ที่มีขนาดใหญ่และมีการแบ่งเป็น Domain ซึ่งเมื่อมีการแบ่งเป็น Domain ต้องมี Main Router เพื่อเป็นตัวแทนการเชื่อมต่อระหว่าง Domain

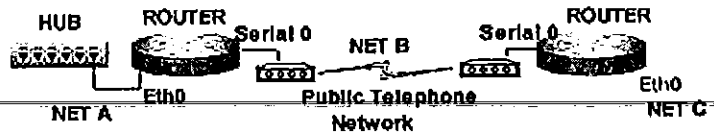
2.5.3 ลักษณะการนำเราเตอร์ไปใช้งานจะมีอยู่ 2 ลักษณะ

แบบที่หนึ่งใช้สำหรับเชื่อมระหว่างแลน 2 เซกเมนต์ดังรูปที่ 1 จุดประสงค์ของการใช้งานในลักษณะนี้มีหลายอย่าง เช่น ลด Traffic ในเครือข่ายขนาดใหญ่ให้ลดลงด้วยการแบ่งเครือข่ายออกเป็น 2 เซกเมนต์หรือมากกว่า โดยใช้เราเตอร์เป็นตัวคั่นระหว่างเซกเมนต์ ซึ่งจะช่วยให้การส่งแพ็กเกจแบบ Broadcast ถูกจำกัดอยู่ภายในเซกเมนต์เท่านั้น และช่วยกั้นไม่ให้แพ็กเกจที่ต้องการคุยกันภายในเซกเมนต์ไม่ให้ข้ามไปรบกวนเซกเมนต์อื่น หรือกรณีเมื่อขอ IP Address จาก ISP (Internet Service Provider) เพื่อใช้ติดต่อกับเครือข่ายอินเทอร์เน็ตอยู่หนึ่งคลาสแต่ต้องการแบ่งให้หน่วยงานต่าง ๆ เป็นเครือข่ายย่อย (Sub Network) ต้องใช้เราเตอร์เป็นตัวคั่นระหว่างเครือข่ายย่อย



รูปที่ 2.11 Router ที่เชื่อม LAN 2 SEGMENT

แบบที่สองใช้สำหรับเชื่อม 2 เครือข่ายที่อยู่ห่างกันเกินความสามารถของมาตรฐานในสาย 10Base5 (500 เมตร), Wireless Lan (ใช้คลื่นวิทยุ) หรือสายเส้นใยนำแสง โดยจะใช้สายเคเบิลโทรศัพท์ในการเชื่อม 2 เครือข่าย ดังรูปที่ 2



รูปที่ 2.12 Router เชื่อม 2 Network เข้าด้วยกัน

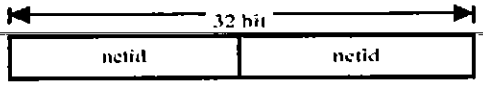
เราเตอร์ประกอบด้วยส่วนสำคัญ 2 ส่วน คือ ฮาร์ดแวร์และซอฟต์แวร์ระบบปฏิบัติการ ด้านหลังของเราเตอร์ประกอบด้วยพอร์ต Ethernet ซึ่งนิยมใช้เป็น RJ45 สำหรับต่อสาย UTP ไปเชื่อมต่อกับฮับ หรือสวิสชิง ในเราเตอร์รุ่นใหม่ ๆ จะใช้ตัวเชื่อมต่อเป็นแบบ FastEthernet ซึ่งสามารถเลือกความเร็วได้ว่าจะใช้ความเร็วของ Ethernet เป็น 10 MB หรือ 100 MB เพื่อให้เหมาะสมกับเครือข่าย

2.6 ไอพีแอดเดรส

อุปกรณ์ที่เชื่อมเข้าเครือข่าย และสามารถทำงานตามข้อกำหนดของทีซีพี/ไอพีคือจะต้องมีแอดเดรสประจำอุปกรณ์นั้น อุปกรณ์ดังกล่าวอาจเป็นโฮสต์ เราเตอร์ เครื่องพิมพ์ หรือแม้กระทั่งอุปกรณ์สำนักงาน เช่น โทรศัพท์ หรือเครื่องถ่ายเอกสาร ไอพีรุ่นสี่กำหนดให้ใช้ไอพีแอดเดรสขนาด 32 บิต อุปกรณ์ที่เชื่อมกับอินเทอร์เน็ตจะมีไอพีแอดเดรส 32 บิต ประจำอินเทอร์เฟสที่ไม่ซ้ำกัน อุปกรณ์อย่างเราเตอร์จะมีหลายอินเทอร์เฟส ซึ่งแต่ละอินเทอร์เฟสจะมีไอพีแอดเดรสหลายค่าตามจำนวนอินเทอร์เฟสโดยไม่ซ้ำค่ากัน แต่ถ้าเป็นเครื่องคอมพิวเตอร์หรือโฮสต์ปกติจะมีเพียงแค่อินเทอร์เฟสเดียว จึงมักเรียกว่าไอพีแอดเดรสเป็นแอดเดรสประจำโฮสต์

แอดเดรสขนาด 32 บิตมีจำนวนแอดเดรสรวมเท่ากับ 2^{32} (4,294,967,296) แต่เมื่อนำมาจัดสรรแล้วไม่สามารถใช้งานได้ครบทั้งหมด ไอพีแอดเดรสนิยมเขียนในรูปเลขฐานสิบ โดยแบ่งเลข 32 บิตเป็น 4 ไบต์ แต่ละไบต์แทนด้วยตัวเลขฐานสิบหนึ่งตัวคั่นแต่ละไบต์ใช้ด้วยเครื่องหมายจุด เช่น แอดเดรส 10011100 00110100 00000010 00000001 จะเขียนได้เป็น 161.246.2.1

แอดเดรสขนาด 32 บิต ประกอบขึ้นจากหมายเลขสองส่วนคือ เลขเครือข่าย (Network Number หรือ Network Identifier หรือ NetID) และเลขโฮสต์ (Host Number หรือ Host Identifier หรือ HostID) เลขเครือข่ายใช้สำหรับจัดคลาสเครือข่าย ส่วนเลขโฮสต์นั้นใช้ระบุหมายเลขโฮสต์ (หรืออีกนัยหนึ่งคืออินเทอร์เฟสของโฮสต์) ในเครือข่าย ไอพีแอดเดรสจึงแบ่งได้เป็นสองส่วน



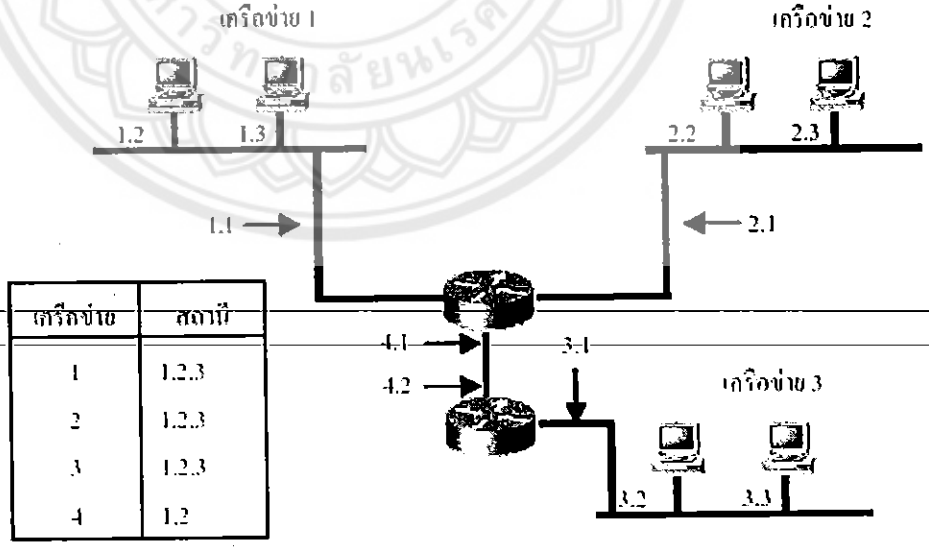
รูปที่ 2.13 จำนวนบิตที่ใช้ สำหรับเลขเครือข่ายและเลขโฮสต์ขึ้นอยู่กับคลาสที่สังกัด

ในปัจจุบันฟิลด์กำหนดเลขเครือข่ายนิยมเรียกว่า พรีฟิกซ์เครือข่าย (Network-Prefix) เพราะทุกโฮสต์ในเครือข่ายจะต้องมีพรีฟิกซ์หรือบิตนำหน้าเหมือนกัน ตัวอย่างเช่นหากมีเลขเครือข่ายจำนวน 16 บิต ก็จะเรียกว่า พรีฟิกซ์ 16 เป็นต้น

2.6.1 ความสำคัญของเลขเครือข่ายและเลขโฮสต์

การจัดแบ่งไอพีแอดเดรสออกเป็นสองส่วนที่ประกอบด้วยเลขเครือข่ายและเลขโฮสต์ก็เพื่อประโยชน์ในการดูแลระบบ เราเตอร์จะอาศัยเลขเครือข่ายเพื่อเลือกเส้นทางส่งแพคเก็จด้วยหลักการต่อไปนี้

- โฮสต์ที่มีเครือข่ายชุดเดียวกันย่อมอยู่ภายในเครือข่ายเดียวกัน และสามารถสื่อสารถึงกันด้วยเฟรมคาต้าลิงค์โดยไม่ต้องพึ่งเราเตอร์
- โฮสต์ที่มีเลขเครือข่ายต่างกันจะอยู่ต่างเครือข่ายกัน การสื่อสารระหว่างโฮสต์จะอาศัยเราเตอร์ที่เชื่อมต่อเครือข่ายเป็นผู้นำส่งแพ็กเก็ต เราเตอร์อาจเชื่อมเครือข่ายที่อยู่ติดกันหรือส่งแพ็กเก็ตเราเตอร์อื่นไปยังปลายทางดังรูป

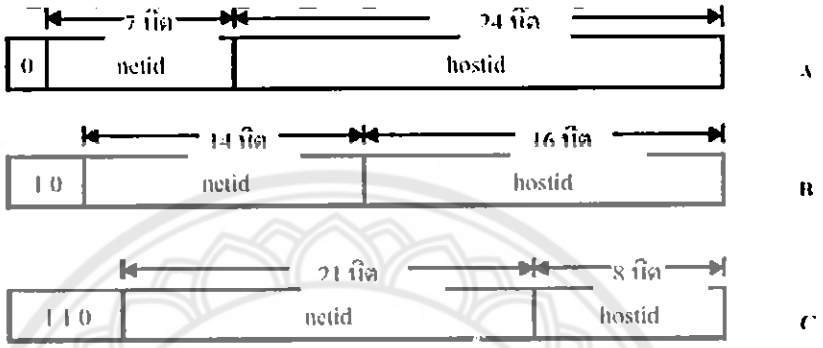


เครือข่าย	สถานี
1	1,2,3
2	1,2,3
3	1,2,3
4	1,2

รูปที่ 2.14 เราเตอร์เชื่อมโยงเครือข่ายที่มีเลขเครือข่ายต่างกัน

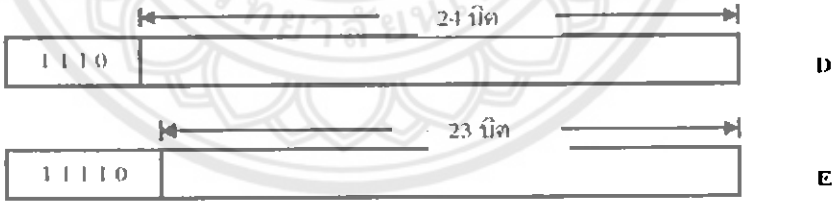
2.6.2 การจัดคลาสเครือข่าย

ไอพีแอดเดรสมีการจัดแบ่งออกเป็นกลุ่มหรือคลาส เครือข่ายที่ใช้งานในปัจจุบันมักสังกัดอยู่ในคลาสใดคลาสหนึ่ง คือ คลาส A, B หรือ C การแบ่งคลาสอาศัยจำนวนพรีฟิกซ์เครือข่ายที่แตกต่างกันตามรูปที่ 2.15 แต่ละคลาสจึงมีจำนวนเครือข่ายในสังกัดและจำนวนโฮสต์ต่อเครือข่ายไม่เท่ากัน



รูปที่ 2.15 การแบ่งคลาสเครือข่าย

การจัดคลาสตามรูปที่ 2.15 เป็นการจัดแบ่งตามการใช้งานเครือข่ายทั่วไป ในขณะที่ยังมีอีก 2 คลาสซึ่งใช้เพื่อจุดประสงค์เฉพาะได้แก่ คลาส D และ E ดังรูปที่ 4-4 เครือข่ายคลาส D เป็นเครือข่ายแบบมัลติคลาสซึ่งจะกล่าวในบทที่ 12 ส่วนคลาส E สงวนไว้ใช้งานหากมีความจำเป็นอื่นใดในอนาคต ทั้งสองคลาสนี้ไม่ได้แบ่งเลขโฮสต์จึงไม่มีกำหนดจำนวนโฮสต์ไว้



รูปที่ 2.16 การแบ่งคลาส D และ E

การจัดคลาสโดยใช้พรีฟิกซ์เป็นการผนวกข้อมูลเพื่อใช้ในการเลือกเส้นทาง เช่น หากตรวจพบว่าพรีฟิกซ์ 2 บิตแรก มีค่าเป็น 10 แสดงว่าเป็นแอดเดรสในคลาส B ซึ่งมีค่า 16 บิตแรก กำหนดกลุ่มเครือข่ายและ 16 บิตถัดมาเป็นเลขโฮสต์

2.6.3 ลักษณะสำคัญของแต่ละคลาส

จำนวนเครือข่ายในแต่ละคลาสและจำนวนโฮสต์สูงสุดที่มีได้ สามารถคำนวณได้จากจำนวนบิตที่ใช้งานตามสูตร 2^n เมื่อ n คือจำนวนบิต ตัวอย่างเช่นในคลาส B มีเลขโฮสต์จำนวน 16 บิต จึงมีโฮสต์ได้ไม่เกิน 2^{16} ซึ่งเท่ากับ 65,536 แต่เลขโฮสต์ที่ทุกบิตเป็น "0" และ เป็น "1" จะสงวนไว้ใช้งานกรณีเฉพาะ จำนวนโฮสต์จึงมีลดลงไป 2 โฮสต์ทุกเครือข่าย หรือนั่นคือมีโฮสต์ไม่เกิน $2^{16} - 2 = 65,534$ สูตร $2^n - 2$ จะใช้กับการคำนวณจำนวนเครือข่ายในคลาสและจำนวนโอสต์ ทั้งคลาส A, B และ C ดังนี้

คลาส A

เครือข่ายในคลาส A มีบิตซ้ายสุดเป็น 0 และใช้ 7 บิตถัดมากำหนดเครือข่าย ส่วนอีก 24 เป็นเลขโฮสต์ คลาส A จึงมีเลขเครือข่ายได้ 2^7 หรือ 128 ค่า แต่เครือข่าย 0.0.0.0 และ 127.0.0.0 สงวนไว้ เป็นแอดเดรสเฉพาะงาน คือ 0.0.0.0 ซึ่งจะเป็นแอดเดรสที่ไว้กำหนดเส้นทางโดยปริยาย (Default Route) ส่วน 127.0.0.0 นั้นเป็นแอดเดรสลูปแบ็ค คือเป็นแอดเดรสที่ใช้เพื่อที่จะเชื่อมเข้าสู่อินเตอร์ลูปแบ็ค ดังนั้นจำนวนเครือข่ายในคลาส A จึงมีได้ 126 เครือข่ายคือเลขที่ขึ้นต้นด้วย 1.0.0.0 ถึง 126.0.0.0

แต่ละเครือข่ายในคลาส A มีแอดเดรสได้ 2^{24}—2 หรือเท่ากับ 16,777,214 คือตั้งแต่ 0.01 ถึง 255.255.254 เครือข่ายในคลาส A ใช้กับหน่วยงานขนาดใหญ่ที่ต้องการแอดเดรสเป็นจำนวนมาก เครือข่ายคลาสนี้จัดสรรให้กับหน่วยงานในยุคแรกเริ่มของอินเทอร์เน็ต แอดเดรสเครือข่ายที่เหลืออยู่ส่วนใหญ่จะสงวนไว้

สังเกตว่าในคลาส A นี้เมื่อกล่าวถึงเฉพาะเลขเครือข่ายก็จะเขียนเฉพาะค่าที่แสดงเลขเครือข่ายที่ขนาด 8 บิต เท่านั้นเช่น 2 หรือ 26 ในทำนองเดียวกันเมื่อกล่าวเฉพาะเลขโฮสต์ก็จะเขียนเฉพาะหมายเลขเครือข่ายโดยให้เลขโฮสต์เป็น "0" เช่น 2.0.0.0 รูปแบบการเขียนเช่นนี้ใช้กับคลาส B และ C เช่นกัน

คลาส B

เครือข่ายในคลาส B มีบิตแรกเริ่มเป็น 10 และใช้ 14 บิตถัดมากำหนดเลขเครือข่ายจำนวนบิตที่กำหนดเลขโฮสต์มีขนาด 16 บิต คลาส B จึงมีสมาชิกเครือข่ายได้ $2^{14} - 2$ หรือ 16,382 คือตั้งแต่ 128.1.0.0 ถึง 192.254.0.0 แต่ละเครือข่ายมีเลขโฮสต์ได้ $2^{16} - 2$ หรือเท่ากับ 65,534 แอดเดรส หรือตั้งแต่ 0.1 ถึง 255.254

เครือข่ายในคลาส B มักจัดสรรให้กับหน่วยงานขนาดกลาง ในปัจจุบันมีเครือข่ายในคลาส B เหลือไม่มากนัก และมักไม่จัดสรรเครือข่ายในคลาสนี้ให้กับผู้ขอจดทะเบียนรายใหม่ หากไม่มีความจำเป็นอย่างแท้จริง

คลาส C

เครือข่ายในคลาส C มีพรีฟิกซ์ 110 และใช้ 21 บิตถัดมาเป็นเลขเครือข่าย จำนวนบิตที่เป็นเลขโอสต์มีเพียง 8 บิต คลาส C จึงมีเลขเครือข่ายได้ตั้งแต่ 192.0.1.0 ถึง 223.255.254.0 รวม 2,097,150 เครือข่ายแต่ละเครือข่ายมีเลขโฮสต์ได้ตั้งแต่ 1 ถึง 254

จำนวนแอดเดรสได้จำกัดเพียง 254 แอดเดรสทำให้เครือข่ายเหมาะสำหรับหน่วยงานขนาดเล็ก หากจำเป็นต้องใช้โฮสต์มากกว่านี้ต้องขอใช้เครือข่ายคลาส C หลายเครือข่าย

คลาส D และ E

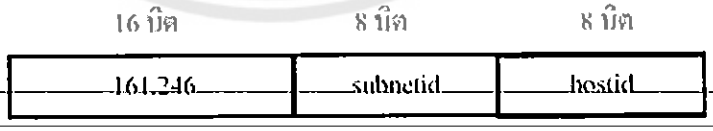
เครือข่ายในคลาส C และ D ไม่มีการจัดแบ่งเลขเครือข่ายและเลขโฮสต์คลาส D มี 3 บิตแรกเป็น 111 จึงมีแอดเดรสตั้งแต่ 224.0.0.0 ถึง 239.255.255.255 แอดเดรสในคลาสนี้เรียกว่า มัลติคาสต์แอดเดรส (Multicast Address)

สำหรับคลาส E มีแอดเดรสจาก 240.0.0.0 ถึง 254.255.255.255 ซึ่งสำรองไว้เพื่อความจำเป็นเฉพาะงานในอนาคต

2.6.4 การแบ่งเครือข่ายย่อย

เครือข่ายที่สังกัดในคลาส A และ B เป็นเครือข่ายที่มีจำนวนโฮสต์ได้เป็นจำนวนมากกล่าวคือ 16,777,214 และ 65,534 ตามลำดับ ในทางปฏิบัติแล้วเราไม่สามารถต่อเชื่อมโฮสต์ทั้งหมดในเครือข่ายเดี่ยวๆ ได้ เพราะข้อจำกัดทางฮาร์ดแวร์ ผู้วางระบบจึงต้องจัดแบ่งเครือข่ายขนาดใหญ่ให้เล็กลงเป็นเครือข่ายขนาดย่อย หรือซับเน็ต (Subnet) การแบ่งซับเน็ต นอกจากจะจัดจำนวนโฮสต์ให้เหมาะสมกับฮาร์ดแวร์ของเครือข่ายแล้วยังช่วยอำนวยความสะดวกในการบริหารเครือข่าย

การจัดซับเน็ตใช้วิธีแบ่งบางส่วนของเลขโฮสต์มาใช้เป็นเลขซับเน็ต (SubnetID) เพื่อกำหนดว่าเป็นเครือข่ายย่อยที่เท่าใด ตัวอย่างเช่นเครือข่าย 161.246.0.0 ซึ่งอยู่ในคลาส B อาจใช้ 8 บิตแรกของเลขโฮสต์เป็นเลขซับเน็ต และ 8 บิตที่เหลือใช้สำหรับเลขโฮสต์ดังรูปที่ 2.17



รูปที่ 2.17 ตัวอย่างการแบ่งเครือข่ายย่อยของ 161.246

จำนวนบิตของเลขซับเน็ตเป็นตัวกำหนดจากจำนวนเครือข่ายย่อย ซับเน็ตขนาด 8 บิต สำหรับเครือข่าย 161.246.0.0 จะมี 254 ซับเน็ต ($2^{subnetid}-2$) แต่ละซับเน็ตมี 254 โฮสต์ ($2^{hostid}-2$) ดังตารางที่ 2.5 เลขซับเน็ตที่ทุกบิตเป็น "1" และ "0" จะสงวนไว้ใช้เฉพาะ ดังนั้นซับเน็ต 161.246.0.0 และ 161.246.555.0 จึงนำมาใช้ไม่ได้

ตาราง 2.3 การจัดแบ่งเครือข่าย 161.246 ด้วยซับเน็ต 8 บิต

ซับเน็ตที่	เครือข่ายย่อย	แอดเดรสเริ่มต้น	แอดเดรสสุดท้าย
1	161.246.1.0	161.246.1.1	161.246.1.254
2	161.246.2.0	161.246.2.1	161.246.2.254
3	161.246.3.0	161.246.3.1	161.246.3.254
...	...	...	...
...	...	...	...
252	161.246.252.0	161.246.252.1	161.246.252.254
253	161.246.253.0	161.246.253.1	161.246.253.254
254	161.246.254.0	161.246.254.1	161.246.254.254

2.6.5 ซับเน็ตมาสก์

เมื่อผู้วางระบบเลือกขนาดซับเน็ตแล้ว จะกำหนดพารามิเตอร์ เพื่อไว้ใช้บอกให้โฮสต์และเราเตอร์ทราบว่าซับเน็ตที่ใช้งานมีขนาดกี่บิต ค่านี้เรียกว่า ซับเน็ตมาสก์ (Subnet Mask)

ซับเน็ตมาสก์เป็นตัวเลข 32 บิต ซึ่งเขียนอยู่ในรูป Dotted – Decimal เช่นเดียวกับการเขียนไอพีแอดเดรส ซับเน็ตมาสก์จะมีบิตที่ตรงกับเลขเครือข่ายและเลขซับเน็ตเท่ากับ "1" ส่วนบิตที่ตรงกับเลขโฮสต์มีค่าเท่ากับ "0" การเลือกซับเน็ตมาสก์ควรใช้ค่าที่มีบิต "1" อยู่ติดกันจากทางซ้ายมือไปทางขวามือเสมอ

ตัวอย่างเครือข่าย 161.246.0.0 ซึ่งแบ่งให้มีเลขซับเน็ตและเลขโฮสต์อย่างละ 8 บิต จะมีค่าซับเน็ตมาสก์เท่ากับ 255.255.255.0 ค่านี้คำนวณได้จากการเขียนไอพีแอดเดรสทั้ง 4 หลัก และใส่เลขฐานสองค่า "1" ให้ครบทุกบิตที่เป็นเลขเครือข่ายและเลขซับเน็ต จากนั้นให้ใส่ค่า "0" สำหรับเลขโฮสต์ แล้วจึงแปลงเลขฐานสองที่

	8 บิต	8 บิต	8 บิต	8 บิต
1. นำค่าไอพีแอดเดรส	161	246	SubnetID	HostId
2. กำหนดบิต "1" และ "0"	11111111	11111111	11111111	00000000
3. แปลงเป็นเลขฐานสิบ	255	255	255	0

เครือข่าย 161.246.0.0 ซึ่งใช้ซับเน็ตมาสก์เท่ากับ 255.255.255.0 เรียกว่า ซับเน็ตมาสก์ 24 บิต เนื่องจากมีบิตที่มีค่า "1" จำนวน 24 บิต หรือเขียนตามรูปแบบที่นิยมใช้ในปัจจุบันคือ 161.246.0.0/24 โดยเรียกว่าเครือข่าย 161.246.0.0 มีพรีฟิกซ์ 24 บิต

สังเกตว่า 161.246.0.0/24 ใช้เลขซับเน็ตจำนวน 8 บิต ดังนั้นนอกจากจะเรียกว่ามีพรีฟิกซ์ 24 บิต แล้ว ยังเรียกได้อีกว่าใช้ซับเน็ตจำนวน 8 บิต

บทที่ 3
ทฤษฎีและหลักการโปรโตคอลทีซีพี/ไอพี

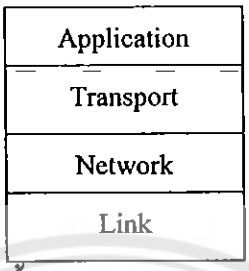
3.1 ความเป็นมาของโพรโตคอลทีซีพี/ไอพี

โพรโตคอล ทีซีพี/ไอพี เป็นชุดของโพรโตคอล ที่มีการพัฒนามาตั้งแต่ ปี 1960 โดยมีวัตถุประสงค์ให้สามารถสื่อสารจากต้นทางข้ามเครือข่ายไปยังปลายทางได้ และสามารถที่จะหาเส้นทางเองได้อัตโนมัติ ถึงแม้ว่าในระหว่างทางผ่านเครือข่ายจะมีปัญหา แต่ก็สามารถหาเส้นทางส่งข้อมูลไปถึงปลายทางได้ จนในปี ค.ศ. 1969 กระทรวงกลาโหมสหรัฐฯ มีการทำการทดลองเชื่อมโยงคอมพิวเตอร์ ระหว่างสองเครือข่ายซึ่งเป็นระบบที่แตกต่างกันให้สามารถติดต่อรับส่งข้อมูลกันได้ โครงการนี้มีชื่อเรียกว่า Advanced Research Project Network (ARPANET) ซอฟต์แวร์ที่ใช้ควบคุมการรับส่งข้อมูลของ ARPANET ประกอบไปด้วย 2 ส่วนหลัก คือ โพรโตคอลทีซีพี (TCP: Transmission Control Protocol) และ โพรโตคอลไอพี (IP: Internet Protocol) โดยที่ ทีซีพี มีหน้าที่ในการตรวจสอบการรับส่งข้อมูลระหว่างคอมพิวเตอร์ผู้รับและผู้ส่งให้ได้รับข้อมูลถูกต้อง ส่วนไอพี มีหน้าที่ในการเลือกเส้นทางที่ใช้รับส่งข้อมูล ผ่านระบบเครือข่ายและตรวจสอบที่แอดเดรสของผู้รับ ซึ่งเรียกว่า ไอพีแอดเดรส ต่อมาในปี ค.ศ. 1983 ทีซีพี/ไอพี ถูกกำหนดให้เป็นมาตรฐานการรับส่งข้อมูล ของกระทรวงกลาโหมของสหรัฐอเมริกา จึงถือว่าทีซีพี/ไอพี มีต้นกำเนิดมาจากโครงการ ARPANET และต่อมาได้ถูกรวมเป็นส่วนหนึ่งของระบบปฏิบัติการยูนิกส์ และได้ถูกใช้กันอย่างแพร่หลาย ซึ่งในปัจจุบันมีการใช้งานอยู่ในแทบทุกเครือข่าย จนเป็นเครือข่ายขนาดใหญ่ กลายเป็นอินเทอร์เน็ตอย่างในปัจจุบัน

โพรโตคอล ทีซีพี/ไอพี ได้รับการออกแบบให้เป็นอิสระ จากชนิดของคอมพิวเตอร์ฮาร์ดแวร์และระบบปฏิบัติการ การทำงานของโพรโตคอลมีความเชื่อถือได้สูงและทำงานได้แม้ในบางสภาวะที่มีการสื่อสารมีความผิดปกติ รวมทั้งความสามารถในการเลือกเส้นทางในการส่งข้อมูลได้ โพรโตคอล ทีซีพี/ไอพี ไม่ได้มีเพียงสองโพรโตคอลดังที่ได้กล่าวมา แต่ ทีซีพี/ไอพี เป็นกลุ่มของโพรโตคอลที่นำมาจัดรวมกันไว้ เรียกว่าเป็นชุดโพรโตคอลทีซีพี/ไอพี

3.1.1 การแบ่งชั้น (Layering)

ทีซีพี/ไอพี (TCP/ IP) เป็นชุดของโพรโตคอล ที่ประกอบด้วยโพรโตคอลย่อยหลายตัว แต่ละตัวก็ทำหน้าที่ ในแต่ละชั้น ซึ่งรับผิดชอบและแปลความหมายของข้อมูล ในแต่ละระดับของการสื่อสาร ซึ่งโดยภาพรวมแล้วทีซีพี/ไอพี แบ่งออกเป็น 4 ชั้น ดังนี้



รูปที่ 3.1 ชั้นทีซีพี/ไอพี (TCP/ IP layer)

หน้าที่และความรับผิดชอบของแต่ละชั้นมี ดังนี้

1. ชั้นเชื่อมโยง (Link Layer) ในชั้นนี้จะเป็นโปรแกรมขับอุปกรณ์ ที่ทำงานอยู่บนระบบปฏิบัติการแต่ละระบบ โดยจะทำหน้าที่รับผิดชอบในการรับส่งข้อมูลตั้งแต่ระดับกายภาพสัญญาณไฟฟ้า จนกระทั่งถึงการแปลความหมายจากระดับแรงดันสัญญาณไฟฟ้ากลายเป็นข้อมูลของคอมพิวเตอร์ โพรโตคอลระดับนี้ เช่น อีเทอร์เน็ต (Ethernet) และเอสแอลไอพี(Serial Line Internet Protocol)

2. ชั้นเครือข่าย (Network Layer) รับผิดชอบในการรับ-ส่งข้อมูล ในเครือข่ายส่งต่อข้อมูลไปยังจุดหมายปลายทาง โดยโพรโตคอลระดับนี้ได้แก่ โพรโตคอลไอพี ,โพรโตคอลไอซีเอ็มพี ,โพรโตคอลไอจีเอ็มพี

a. ชั้นรับส่งข้อมูล (Transport Layer) รับผิดชอบในการรับส่งข้อมูลระหว่างเครื่องหนึ่งไปยังอีกเครื่องหนึ่ง และจะทำการส่งข้อมูลไปให้ชั้นของการประยุกต์ นำไปใช้งานต่อ โพรโตคอลระดับนี้ได้แก่ โพรโตคอลทีซีพี (TCP), โพรโตคอล (UDP)

b. ชั้นประยุกต์ (Application Layer) เป็นชั้นที่แอพพลิเคชั่นเรียกโพรโตคอลระดับล่างๆลงไป เพื่อวัตถุประสงค์แตกต่างกัน เช่น

โพรโตคอลเอฟทีพี (FTP :File Transfer Protocol)

ใช้สำหรับรับส่งข้อมูลระหว่างคอมพิวเตอร์

โพรโตคอลเอสเอ็มทีพี (SMTP :Simple Mail Transfer Protocol)

ใช้สำหรับรับส่งจดหมายอิเล็คทรอนิคส์

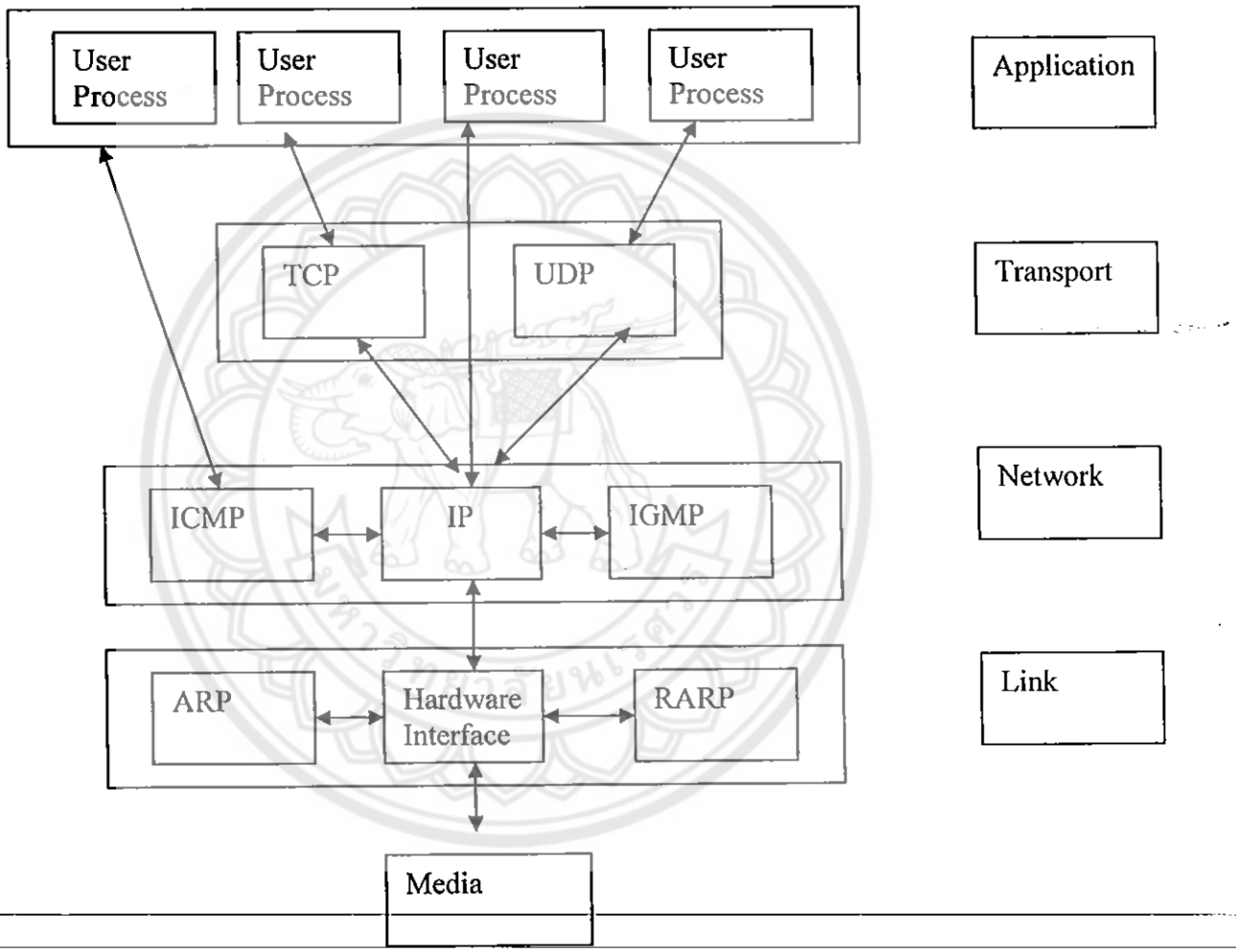
โพรโตคอลเทลเน็ต (Telnet)

ใช้สำหรับควบคุมเครื่องระยะไกล

โพรโตคอลเฮชทีทีพี (HTTP :Hypertext Transfer Protocol)

เป็นโพรโตคอลที่ใช้รับและส่งข้อมูลเว็บเพจ

ระดับชั้นทีซีพี (TCP Layer)



รูปที่ 3.2 ชั้นของโพรโตคอลต่างๆในชุดของ ทีซีพี/ไอพี

โพรโตคอลทีซีพี (TCP) : อยู่ในชั้นรับส่งข้อมูลทำหน้าที่จัดการและควบคุมการรับส่งข้อมูลให้มีเสถียรภาพและเชื่อถือได้

โพรโตคอลยูดีพี (UDP) : อยู่ในชั้นรับส่งข้อมูล ทำหน้าที่จัดการและควบคุมการรับส่งข้อมูล เช่นเดียวกัน แต่ ไม่มีกลไกการรับส่งที่มีเสถียรภาพและเชื่อถือได้ โดยปล่อยหน้าที่นี้ให้กันกับแอพพลิเคชั่นชั้นเป็นผู้ทำหน้าที่นี้แทน

โพรโตคอลไอพี (IP) : อยู่ในชั้นเครือข่าย เป็นโพรโตคอลหลัก ในการสื่อสารข้อมูลซึ่งกลไกสำคัญที่ทำให้ข้อมูลสามารถเคลื่อนที่ไปยังปลายทางได้ ก็คือโพรโตคอลไอพีนี้เอง

โพรโตคอลไอซีเอ็มพี (ICMP: Internet Control Message Protocol) : อยู่ในชั้นเครือข่าย โดยทำหน้าที่เสริมการทำงานของไอพีให้สมบูรณ์ โดยจะเป็นโพรโตคอลที่คอยส่งข่าวสาร และแจ้งความผิดพลาดให้แก่ไอพี

โพรโตคอลไอจีเอ็มพี (IGMP: Inter Group Management Protocol) : อยู่ในชั้นเครือข่าย ทำหน้าที่ในการส่ง UDP Datagram ไปยังกลุ่มของคอมพิวเตอร์ในเครือข่ายหลายๆตัวพร้อมกัน

โพรโตคอลเออาร์พี (ARP: Address Reservationo Protocol) : อยู่ในชั้นเชื่อมโยง ทำหน้าที่เปลี่ยนไอพี แอดแดรสที่ใช้โดยให้เป็นแอดแดรสของตัวเชื่อมเครือข่าย (Network Interface)

โพรโตคอลอาร์เออาร์พี (RARP) : อยู่ในชั้นเชื่อมโยง ทำหน้าที่กลับกันกับเออาร์พี คือ เปลี่ยนระหว่าง แอดแดรสของตัวเชื่อมเครือข่ายให้เป็นแอดแดรสที่ใช้โดยอินเตอร์เน็ตแอดเดรส

3.1.2 อินเตอร์เน็ต แอดเดรส (Internet Address)

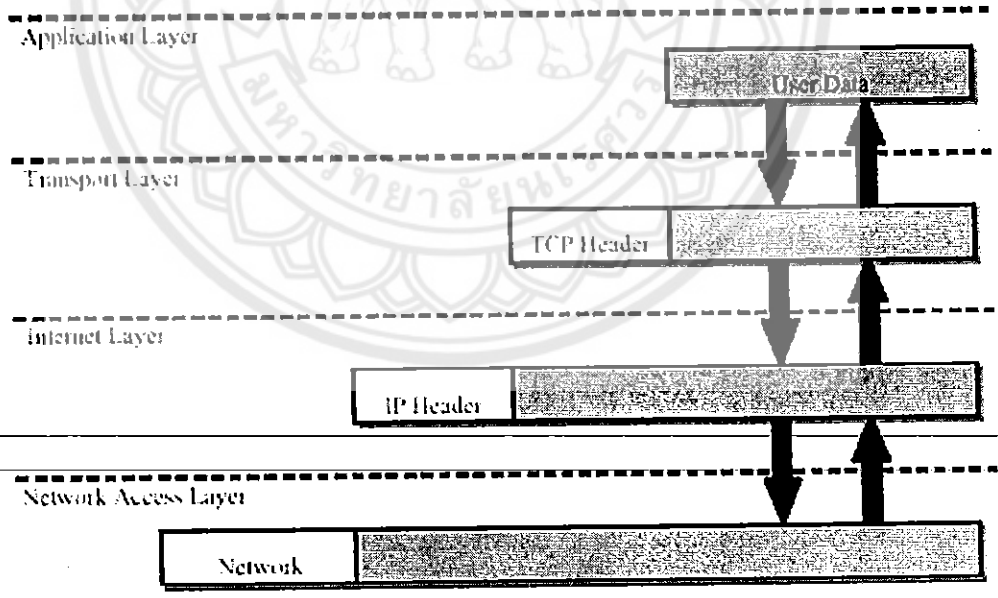
ทุกการเชื่อมต่อที่อยู่บนอินเตอร์เน็ต จะต้องมีหมายเลขประจำตัวเพื่อใช้ในการสื่อสารข้อมูล เรียกว่าอินเตอร์เน็ต แอดเดรส หรือเรียกย่อๆว่า ไอพี แอดเดรส โดยค่าไอพีแอดเดรสนี้จะเป็นหมายเลขจำนวน 32 บิตแต่แทนที่จะกำหนดให้เลขทั้ง 32 บิตนั้น ถูกนับต่อเนื่องกันไป วิธีการแบ่งหมายเลขดังกล่าวออกเป็นกลุ่มของเลขขนาด 8 บิต จำนวน 4 ชุด และคั่นแต่ละชุดด้วยจุด เช่น192.168.68.1

นอกจากนี้ ในไอพี แอดเดรส ยังถูกแบ่งออกเป็น 2 ส่วนคือ ส่วนที่เป็นแอดเดรสของเครือข่าย (Network ID) และส่วนที่เป็นแอดเดรสของแม่ข่าย (Host ID) ซึ่งข้อมูลดังกล่าวจะถูกใช้สำหรับค้นหาเส้นทางของไอพีในการส่งข้อมูลจากต้นทางไปจนถึงปลายทางได้อย่างถูกต้อง

Class	Range
A	0.0.0.0-127.255.255.255
B	128.0.0.0-191.255.255.255
C	192.0.0.0-223.255.255.255
D	224.0.0.0-239.255.255.255
E	240.0.0.0-255.255.255.255

ตารางที่ 3.1 แสดงช่วงของ ไอพี แอดเดรสของแต่ละคลาส

3.1.3 การเก็บข้อมูล และการส่งข้อมูล



รูปที่ 3.3 แสดงการส่งข้อมูลในโมเดลของทีซีพี/ไอพี

ในการรับส่งข้อมูลนั้น ข้อมูลที่รับส่งกันจริงๆบนเครือข่ายนั้นจะประกอบด้วย 2 ส่วนคือ ข้อมูลจริงกับข้อมูลของโพรโตคอล ข้อมูลของโพรโตคอลเรียกว่าเฮดเดอร์ เรียกส่วนที่มีข้อมูลจริงกับข้อมูลของโพรโตคอลว่าโปรแกรมสำเร็จ เปรียบเสมือนการส่งจดหมายซึ่งจะต้องประกอบไปด้วยเนื้อความของจดหมายและซองจดหมายที่เขียนชื่อที่อยู่ ติดแสตมป์ จะเปรียบเหมือนข้อมูลที่ใช้ในการรับส่งข้อมูลของโพรโตคอลนั้น 1 โพรโตคอลก็จะใส่ 1 ซอง ถ้าข้อมูลต้องส่งผ่านหลายชั้น จำนวนซองก็จะถูกใส่เพิ่มหลายชั้นตามลำดับ ดังนั้นถ้าส่งข้อมูลผ่านโพรโตคอลทีซีพีข้อมูลก็จะมีการทำงานตามลำดับดังนี้

ลำดับที่ 1 ซองของทีซีพี (TCP)

ลำดับที่ 2 ซองไอพี (IP)

ลำดับที่ 3 ซองอีเทอร์เน็ต (Ethernet)

และฝ่ายที่รับข้อมูลก็จะต้องแกะซองออกตามลำดับ โดยจะต้องแกะซองของอีเทอร์เน็ตก่อน แล้วจึงจะเจอซองของไอพี เมื่อแกะซองของไอพี แล้วก็เจอซองของทีซีพี และในลำดับสุดท้ายก็เจอข้อมูลที่ต้องการตามลำดับ

3.1.4 หมายเลขประจำตัวของโพรโตคอล **(Port Number)**

ในโพรโตคอลทีซีพี/ไอพีจะมีการกำหนดพอร์ตอยู่ในหัวเรื่องของโปรแกรมสำเร็จ เพื่อระบุว่าข้อมูลเซกเมนต์นี้เป็นของโปรแกรมประยุกต์อะไร ดังเช่น พอร์ต 20, 21 เป็นของ เอฟทีพี, 23 เป็นของ เทลเน็ต, พอร์ต 80 เป็นของเฮชทีทีพี เป็นต้น

ในชุดโพรโตคอลทีซีพี/ไอพี มีโพรโตคอลหลักที่ขอกล่าวถึงหลักๆ 5 โพรโตคอล ได้แก่ โพรโตคอลทีซีพี โพรโตคอลยูดีพี โพรโตคอลไอซีเอ็มพี โพรโตคอลไอพี และโพรโตคอลเออาร์พี ซึ่งการทำงานของแต่ละโพรโตคอลมีรายละเอียด ดังนี้

3.2 โพรโตคอลทีซีพี (TCP: Transmission Control Protocol)

โพรโตคอลทีซีพี มีการส่งข้อมูลมีความสัมพันธ์ต่อเนื่องกัน มีกลไกในการตรวจสอบทั้งด้านส่งและด้านรับเพื่อให้แน่ใจว่าทั้งสองฝั่งมีความพร้อมและสามารถสื่อสารกันได้จริงจึงจะมีการรับส่งข้อมูลเกิดขึ้น จนมีการรับ-ส่งข้อมูลแล้วก็จะมีการยืนยันความถูกต้องทุกครั้งของการสื่อสารเพื่อรับประกันว่าข้อมูลที่รับ-ส่งนั้นถูกต้องตรงกันทั้ง 2 ฝ่าย ด้วยลักษณะเช่นนี้การสื่อสารด้วยโพรโตคอลทีซีพี จึงเสมือนว่าทั้งสองฝ่ายคือฝ่ายรับและฝ่ายส่งได้ทำการต่อสายเครือข่ายถึงกัน (Connected) ตลอดเวลาที่การรับส่งข้อมูลจนกระทั่งการสื่อสารทั้งหมดเสร็จสิ้นจึงจะทำการยกเลิกการเชื่อมต่อนั้น

3.2.1 บริการของทีซีพี (TCP Services)

จุดเด่นประการสำคัญของทีซีพี ที่กล่าวถึงอยู่เสมอคือ ความมีเสถียรภาพและความถูกต้องของการสื่อสารซึ่งมีความเชื่อถือได้สูง ดังนี้

1. ข้อมูลที่จะส่งผ่านโพรโตคอลทีซีพี นั้นจะต้องถูกนำมาแตกย่อยออกเป็นส่วนๆให้มีขนาดเหมาะสมสำหรับการส่ง โดยโพรโตคอลทีซีพีจะเป็นตัวพิจารณาว่า ควรมีขนาดเท่าใดจะทำให้การรับ-ส่งนั้นมีประสิทธิภาพมากที่สุด

2. ในการส่งข้อมูลแต่ละครั้งของโพรโตคอลทีซีพี จะมีการจับเวลาเพื่อรอให้อีกฝ่ายหนึ่งตอบกลับมาว่าส่งถึงแล้ว ถ้าถึงกำหนดเวลาแต่ยังไม่มีการตอบกลับมา โพรโตคอลทีซีพีถือว่าข้อมูลยังไม่ถึงปลายทางก็จะจัดส่งไปใหม่

3. ทุกๆครั้งที่โพรโตคอลทีซีพีได้รับข้อมูลก็จะมีการตอบรับยืนยันกลับไปยังผู้ส่งว่าได้รับข้อมูลเรียบร้อยแล้ว

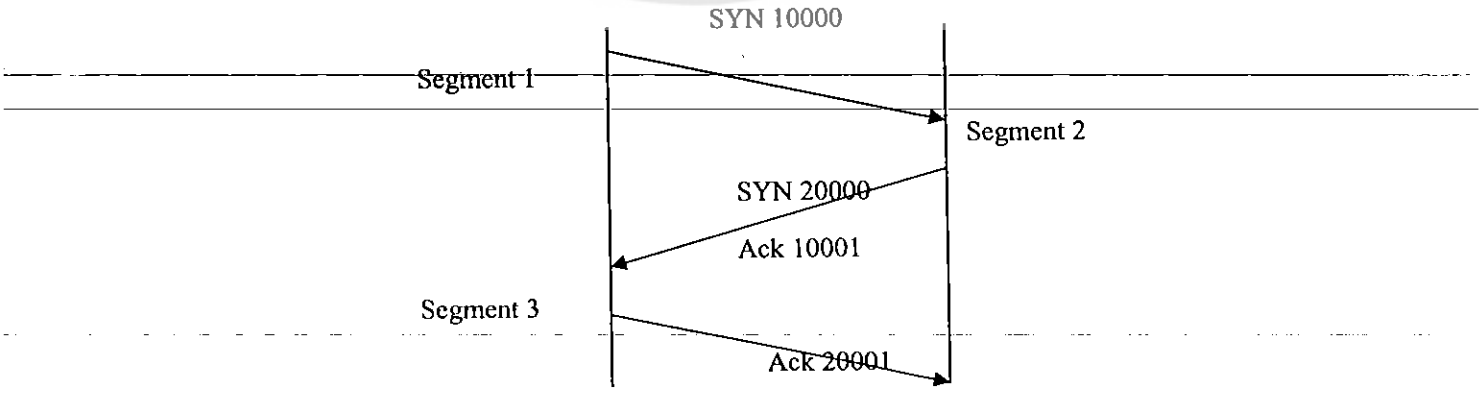
4. โพรโตคอลทีซีพี มีการตรวจสอบความถูกต้อง ซึ่งจะคลอบคลุมทั้งส่วนส่วนหัวของโพรโตคอลทีซีพี (TCP Header) และในส่วนของข้อมูล (TCP Data)

5. เมื่อมีการรับข้อมูลที่ถูกแยกออกเป็นส่วนย่อย (Flagment) โพรโตคอลทีซีพีจะต้องเรียงข้อมูลที่ได้รับมาให้ถูกต้อง

6. การรับ-ส่งด้วยโพรโตคอลไอพี อาจจะมีข้อมูลซ้ำได้ ดังนั้นโพรโตคอลทีซีพี จะต้องทราบได้ว่าข้อมูลนี้ซ้ำกับของเดิม

7. โพรโตคอลทีซีพีมีกลไกในการควบคุมการรับ-ส่งข้อมูลที่เหมาะสมระหว่างผู้รับกับผู้ส่ง คือการส่งข้อมูลต้องส่งไปให้ผู้รับเท่าที่ผู้รับมีที่เก็บข้อมูลเพียงพอ

3.2.2 การสร้างการเชื่อมต่อ (Connection Esablishment)



รูปที่ 3.4 การสร้างการเชื่อมต่อ (Connection Establishment)

ก่อนที่โพรโตคอลทีซีจะสามารถรับส่งข้อมูลได้จะต้องมีการสถาปนา หรือการสร้างให้การเชื่อมต่อเกิดขึ้นก่อน เปรียบเทียบการต่อสายของทั้งสองฝั่งให้เชื่อมถึงกัน ซึ่งโพรโตคอลทีซีพี ได้กำหนดขั้นตอนในการเริ่มต้นสร้างการเชื่อมต่อดังนี้

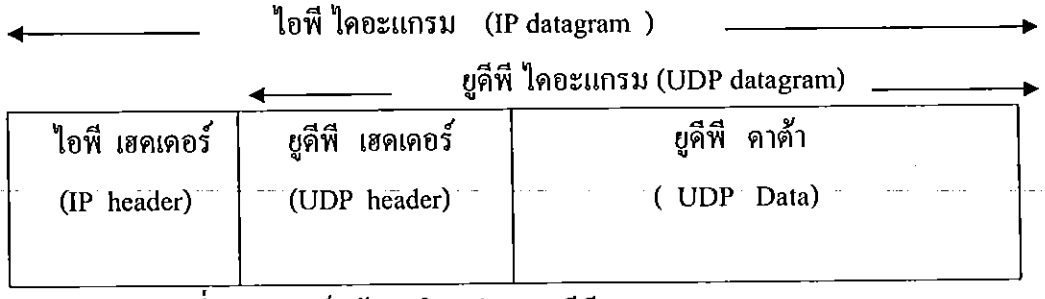
1. เครื่องลูกข่ายจะทำการส่งเซกเมนต์ โดยเปิดสัญญาณการเชื่อมต่อ (SYN Flag) ระบุหมายเลขพอร์ตที่ต้องการติดต่อบนเครื่องบริการและระบุหมายเลขลำดับของการส่งข้อมูลออกมา ไอเอสเอน (ISN: Initial Sequence Number)
2. เครื่องบริการ เมื่อได้รับข้อมูลเซกเมนต์จากข้อหนึ่งแล้ว ก็จะตอบกลับด้วยการเพิ่มค่าของไอเอสเอน ขึ้นอีกหนึ่งพร้อมทั้งระบุหมายเลขลำดับของตนเอง และเปิดสัญญาณการเชื่อมต่อกับสัญญาณยอมรับการเชื่อมต่อ
3. เครื่องลูกข่ายเมื่อได้รับการตอบกลับจากเครื่องบริการตามข้อสอง ก็ทำการตอบกลับเพิ่มค่าของไอเอสเอน ขึ้นอีกหนึ่ง และเปิดสัญญาณการเชื่อมต่อ (ACK Flag) ด้วยเช่นกัน

เมื่อผ่านการสร้างการเชื่อมต่อทั้งสามขั้นตอนแล้ว ดังนั้นตอนนี้ทั้งลูกข่ายและเครื่องบริการเปรียบเสมือนมีการเชื่อมต่อถึงกันแล้ว จึงสามารถรับส่งข้อมูลกันได้ตลอดเวลาจนกว่าจะมีการยุติการเชื่อมต่อนั้น ขั้นตอนทั้งสามนั้นเรียกว่า "Threee-ways handshakes"

SYN	เป็นข้อมูลระดับบิตที่ใช้ในการเริ่มต้นขอการติดต่อกับปลายทาง
ACK	เป็นข้อมูลระดับบิตที่ใช้แสดงว่า แอคโนวเลจ นัมเบอร์ (Acknowledge Number) พร้อมใช้งาน
ISN	เป็นหมายเลขลำดับการเชื่อมต่อแต่ละครั้ง
Acknowledge Number	คล้ายกันกับไอเอสเอ็นแต่ไว้สำหรับการตอบกลับ

3.3 โพรโตคอลยูดีพี (UDP: User Datagram Protocal)

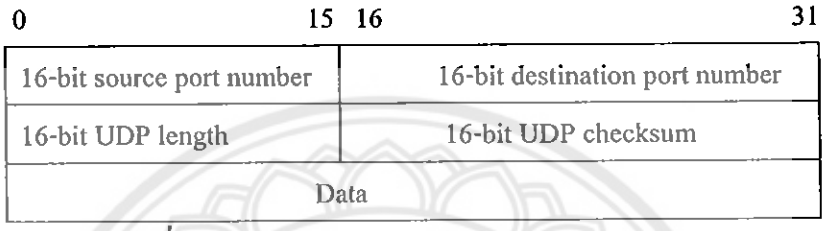
ยูดีพี เป็นโพรโตคอลพื้นฐานที่อาศัย โพรโตคอลไอพีเป็นพาหนะในการส่งข้อมูล โดยตัวยูดีพีนั้นจัดอยู่ในชั้นรับส่ง



รูปที่ 3.5 การเก็บข้อมูลในรูปแบบ ยูดีพี (UDP Encapsulate)

ยูดีพีดาต้าแกรมจะถูกจัดเก็บข้อมูล (Encapsulate) ลงใน ไอพี ดาต้าแกรม ดังแสดงในภาพ 2.3.1 โดยเมื่อจัดเก็บข้อมูล แล้ว 20 ไบต์แรก จะเป็นของส่วนระบุโพรโตคอลไอพี และในไบต์ที่ 9 ของส่วนระบุโพรโตคอลไอพี ต้องมีค่าเท่ากับ 17 ด้วย และคุณสมบัติที่สำคัญของยูดีพี คือ จัดรูปแบบข้อมูลอย่างง่ายให้อยู่ในรูปของยูดีพี ดาต้าแกรม และรับส่งข้อมูลชุดนี้ให้ถึงปลายทาง เท่านั้น ไม่มีกลไกใดๆในการยืนยันในการตรวจสอบยืนยันการรับส่งในตัวของยูดีพีเอง

ส่วนข้อมูลระบุโพรโตคอลยูดีพี (UDP header)



รูปที่ 3.6 ส่วนข้อมูลระบุโพรโตคอลยูดีพี (UDP header)

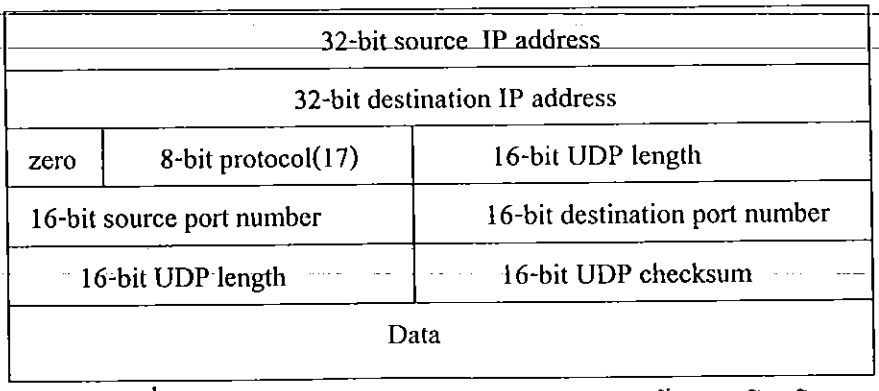
ไบต์ 0-3 หมายเลข ของช่องทางที่ส่งข้อมูลดาต้าแกรม

ไบต์ 2-3 หมายเลข ช่องทางที่รับข้อมูลดาต้าแกรมไปใช้งาน

ไบต์ 4-5 เป็นส่วนระบุความยาวของโพรโตคอลยูดีพี (UDP length) เป็นเขตข้อมูลที่ระบุความยาวของ ยูดีพี ดาต้าแกรมคือ ส่วนระบุโพรโตคอลยูดีพี+ ส่วนข้อมูลในโพรโตรคอลยูดีพี โดยขนาดต่ำสุดของส่วนระบุความยาวของโพรโตคอลยูดีพีมีค่า 8

ไบต์ 6-7 เป็นส่วนตรวจสอบความถูกต้องของโพรโตคอล (UDP Checksum) ซึ่งจะทำหน้าที่ตรวจสอบความถูกต้องของ ยูดีพี ดาต้าแกรมทั้งหมด

ส่วนตรวจสอบความถูกต้องของโพรโตคอลยูดีพี (UDP Checksum)



รูป 3.7 เขตข้อมูลที่ใช้ในการคำนวณหาส่วนตรวจสอบความถูกต้องของโพรโตคอลยูดีพี

กลไกในการหาค่า ความถูกต้อง (checksum) เพื่อใช้ในการตรวจสอบความถูกต้องของยูดีพี จะคล้ายกับการหาความถูกต้องของโพรโตคอลไอพี คือค่าความถูกต้องที่ได้จะเป็นค่าผลรวมของข้อมูลขนาด 16 บิต ทั้งหมด และแปลงเป็น one 's complement แต่ว่าจะมีจุดแตกต่างจากการหาความถูกต้องของไอพี อยู่ 2 ประการคือ

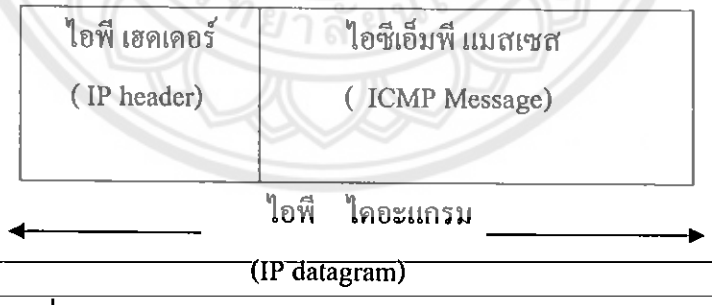
1. ขนาดของ ยูดีพี คาต้าแกรมจะไม่คงที่ เนื่องจากขนาดของส่วนที่เป็นข้อมูลซึ่งอาจจะเปลี่ยนแปลงได้ตามขนาดข้อมูลจริง

2. ถึงแม้ว่าจริงๆแล้ว ส่วนระบุโพรโตคอลยูดีพี จะมีขนาด 8 ไบต์ แต่ว่าในการหาความถูกต้องนั้นจะนำบางค่าใน ไอพี แอดเดรส มารวมกัน เป็นส่วนหนึ่งของส่วนระบุโพรโตคอลยูดีพี จากนั้นจึงหาค่าความถูกต้องทั้งหมดอีกทีหนึ่ง

3.4 โพรโตคอลไอซีเอ็มพี (ICMP: Internet Control Message Protocal)

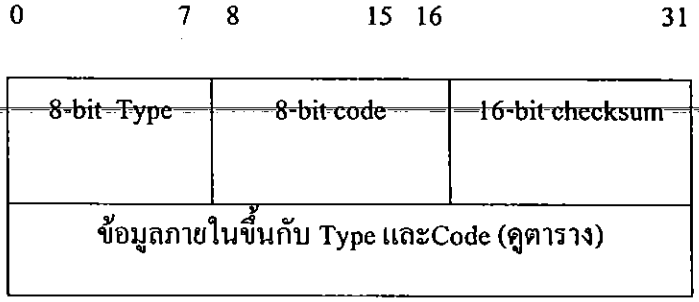
ไอซีเอ็มพี เป็นโปรโตคอลหนึ่งที่อยู่ในชุดของ ทีซีพี/ไอพี (TCP/IP Suite) มีหน้าที่ส่งข่าวสารและคำสั่ง ควบคุมของไอพี โดยเฉพาะการรับส่ง ข้อความผิดพลาด (Error Message) ด้วยลักษณะของ ไอซีเอ็มพี อยู่ทั้งในชั้นเดียวกับไอพีหรือชั้นที่สูงกว่าไอพี คือเทียบเท่าทีซีพีและยูดีพีก็ได้ ขึ้นอยู่กับลักษณะของ แมสเซส (Message) ที่ไอซีเอ็มพีทำการสื่อสาร ทั้งนี้ ไอซีเอ็มพีใช้ไอพีเป็นตัวส่งข้อมูลเช่นเดียวกับ ทีซีพีและยูดีพี

การจัดเก็บข้อมูล (Encapsulate) แบบไอซีเอ็มพี



รูปที่ 3.8 การจัดเก็บข้อมูลแบบไอซีเอ็มพี (ICMP Encapsulate)

จะเห็นว่าในส่วนของไอพีเฮดเดอร์ ก็ยังคงใช้ เฮดเดอร์ตามปกติของไอพี ทำให้กลไกในการรับส่งข้อมูลจากต้นทางไปยังปลายทางสามารถใช้กลไกปกติของ ไอพีได้ทันที สำหรับกลไกในการแสดงข้อความของไอซีเอ็มพี จะเก็บข้อมูลอะไร มีความหมายว่าอย่างไรบ้างจะแสดงในรูป 3.9



รูปที่ 3.9 ไอซีเอ็มพี แมสเซส (ICMP Massage)

ความหมายของข้อมูลใน ไอซีเอ็มพี แมสเซส (ICMP Massage)

บิตที่ 0-7 ชนิดของไอซีเอ็มพี เขตข้อมูลขนาด 8 บิตบอกถึง ประเภทของ**ไอซีเอ็มพี ที่กำลัง** สื่อสารอยู่

บิตที่ 8-15 รหัสของไอซีเอ็มพี เป็นเขตข้อมูลขนาด 8 บิต ที่เก็บข้อมูลร**หัสของ ไอซีเอ็มพี** แมสเซส

บิตที่ 16-31 ค่าความถูกต้องใช้เป็นตัวตรวจสอบความถูกต้องของไอซีเอ็มพี**แมสเซส**

การใช้งานของไอซีเอ็มพี

โดยทั่วไป ไอซีเอ็มพี จะใช้งานเพื่อ 2 ลักษณะคือ

- Query ใช้สำหรับสอบถามสถานะระหว่างกัน
- Error ใช้สำหรับรายงานข้อผิดพลาดที่เกิดขึ้น

สำหรับการรายงานความผิดพลาดนั้น ไอซีเอ็มพี จะไม่รายงานความ**ผิดพลาดของการส่ง ข้อมู**ลในกรณีต่างๆ เหล่านี้ เพื่อป้องกันการรายงานข้อผิดพลาดไม่รู้จบ

1. เกิดความผิดพลาดในการส่งเสียเอง ถึงแม้ว่า ไอซีเอ็มพี จะเป็นตัว**ที่คอยรายงานความ ผิดพ**ลาดของ ไอพี แต่ตัว ไอซีเอ็มพี (ICMP) เองก็ยังต้องอาศัยไอพี เป็นตัว**นำมันกลับไปยังที่อยู่ ปลายทางอยู่ดี การที่ ไอซีเอ็มพี อาจจะเดินทางกลับไม่ถึงปลายทางด้วยเหตุใดก็แล้วแต่ ย่อมเป็นสิ่ง ที่มีโอกาสเกิดขึ้นได้**

2. ข้อมูลที่ไอพีแอดเดรส ปลายทาง ไม่ได้เฉพาะเจาะจงกับเครื่องแม่ข่ายเพียงตัวเดียว แต่เป็นปลายทางประเภทแพร่สัญญาณ (Broadcast) และมัลติคาสต์ (Multicast)

3. คาต้าแกรมที่ทำหน้าที่เหมือนการแพร่สัญญาณของระดับชั้นเชื่อมโยง

4. คาต้าแกรมที่ถูกแยกเป็นส่วนย่อยๆมา

5. คาต้าแกรมที่ต้นทางไม่ได้เฉพาะเจาะจงกับเครื่องแม่ข่าย ดังนั้นแอดเดรสในลักษณะนี้ เมื่อนำมาใช้ เป็นแอดเดรสต้นทางก็จะไม่ได้รับ ไอซีเอ็มพี แมสเซส

3.5 โพรโตคอลเออาร์พี (ARP: Address Resolution Protocol)

เออาร์พี เป็นกระบวนการเปลี่ยนค่าระหว่างไอพี ไปเป็นแอดแดรสอีเทอร์เน็ต หรือที่เรารู้จักกันในชื่อ MAC Address คือแอดแดรสทางฮาร์ดแวร์ของอุปกรณ์ เช่น การ์ดแลน (Card LAN)

รูปแบบโปรแกรมสำเร็จรูปของ เออาร์พี (ARP Packet Format)

Ethernet Des. addr.	Ethernet Source Addr.	Frame Type	Hard Type	Prot Type	Hard Size	Prot Size	Sender Eth. Addr.	Sender ไอพี Addr.	Target Eth. Addr.	Tar. ไอพี Addr

รูปที่ 3.10 รูปแบบโปรแกรมสำเร็จรูปของ เออาร์พี

ในแพ็กเกจของ เออาร์พี จะประกอบด้วย

ไบต์ 0-5 **Ethernet Destionation Address** สำหรับอีเทอร์เน็ตทั่วไปจะหมายถึงแอดแดรสของปลายทาง แต่ว่าสำหรับในกรณีที่เป็นของโพรโตคอลเออาร์พีเนื่องจาก เป็นการส่งข้อมูลถึงแม่ข่ายที่อยู่บนเครือข่าย หรือที่เรียกว่าการแพร่สัญญาณ ดังนั้นทุกบิตของเขตข้อมูลนี้จึงต้องเป็น "1" ทั้งหมดคือ FF FF FF FF FF FF

ไบต์ 6-11 **Ethernet Source Address** เป็นแอดแดรสของผู้ที่ได้ทำการส่งเออาร์พีรีเควส (ARP Request) เอง เพื่อให้แม่ข่ายที่ต้องการตอบกลับสามารถตอบกลับมาได้อย่างถูกต้อง

ไบต์ 12-13 **Ethernet Frame Type** ระบุถึงโพรโตคอล ที่เก็บข้อมูลอยู่ในกรอบของอีเทอร์เน็ตนี้ สำหรับโพรโตคอลเออาร์พีจะต้องเป็น 0x0806

ไบต์ 14-15 **Hard Type** ระบุประเภทของแอดแดรสของอุปกรณ์ที่โพรโตคอลเออาร์พีกำลังถามอยู่ในกรณีคือแอดแดรสอีเทอร์เน็ต (Ethernet Address) ค่าจะต้องเป็น 1

ไบต์ 16-17 **Prot Type** ระบุโพรโตคอลที่จะต้องการถาม หมายถึง ต้องการถามแอดแดรส ของอุปกรณ์ของโพรโตคอลอะไร กรณีนี้คือโพรโตคอลไอพี

ไบต์ 18 **Hard Size** ระบุขนาดของฮาร์ดแวร์แอดแดรสเท่ากับ 6 สำหรับแอดแดรสอีเทอร์เน็ต

ไบต์ 19	**Port Size** ระบุขนาดของแอดแดรสในโพรโตคอลที่ถามเท่ากับ4 สำหรับโพรโตคอลไอพี (IP)
ไบต์ 20-21	**OP Field** เป็นการระบุว่าเป็น โพรโตคอลเออาร์พีชนิดใด 1=ARP Request 2=ARP Reply 3=RARP Request 4=RARP Reply
ไบต์ 22-27	**Sender Ethernet Address** ค่าแอดแดรส อีเทอร์เน็ต ของผู้ส่ง ซึ่งจะมีค่าซ้ำกับในไบต์ 6-11
ไบต์ 28-31	**Sender IP Address** คือค่าไอพี แอดเดรสของผู้ส่ง
ไบต์ 32-37	**Target Ethernet Address** จะว่างไว้สำหรับ เออาร์พี รีเควส
ไบต์ 38-41	**Target IP Address** คือค่า ไอพี แอดเดรสที่กำลังต้องการหาค่าแอดแดรสอินเทอร์เน็ต

3.6 โพรโตคอลไอพี (IP: Internet Protocol)

ไอพี เป็นโพรโตคอลที่ทำหน้าที่รับภาระในการนำข้อมูลไปส่งยังจุดหมายปลายทางไม่ว่าที่ใดๆในอินเตอร์เน็ต โพรโตคอลต่างๆไม่ว่าจะเป็นชุดโพรโตคอลทีซีพี ทั้ง ทีซีพี,ยูดีพี,ไอซีเอ็มพี ต่างก็ต้องอาศัยระบบนี้ เนื่องจากตัวโพรโตคอลไอพี นี้มีกลไกที่ค่อนข้างฉลาดในการหาเส้นทางขนส่งข้อมูล ถึงแม้ว่า ไอพี จะเป็นโพรโตคอลที่เชี่ยวชาญในการขนส่งข้อมูลไปได้ไกลๆ แต่ก็มีจุดด้วยคือ ไอพี เป็นโพรโตคอลที่ขนส่งข้อมูลได้อย่างรวดเร็วแต่ไม่มีการรับประกันว่าข้อมูลถึงปลายทางหรือไม่

ส่วนหัวของโพรโตคอลไอพี (IP Header)

4-bit version	**4-bits herder len.**	**8-bit type of service**	**16 bit total length**
16 bit identification		**3-bin flag**	**13-bit fragment offset**
8 bit TTL	**8 bit Protocol**	**16 bit header checksum**	
32 bit source IP address			
32 bit destination IP address			
Option (if any)			
data			

รูปที่ 3.11 ส่วนหัวของโพรโตคอลไอพี (IP Header)

สำหรับข้อมูลแต่ละส่วนในส่วนระบุโพรโตคอลมีความหมายดังนี้

บิต 0-3	รุ่นของทีซีพี/ไอพี ปัจจุบันเป็นรุ่น 4
บิต 4-7	**Header Length** ความยาวของส่วนระบุโพรโตคอล โดยทั่วไปถ้าไม่มีค่าอยู่ในส่วนนี้ จะเป็น 5 หมายความว่า ความยาวข้อมูล ความยาวข้อมูลมีขนาด 5* 32 บิต หรือเท่ากับ 20 ไบต์ บิต 8-15 Type of Service (TOS) ปัจจุบันไม่ได้ใช้งานแล้ว
บิต 16-31	**Total length** เป็นเขตข้อมูลที่มีไว้สำหรับบอกจำนวนไบต์ทั้งหมดของไอพี ดาต้าแกรม (IP Datagram) ด้วยขนาด 16 บิต ของเขตข้อมูลนี้แสดงว่าขนาดความยาวข้อมูลของไอพี ดาต้าแกรม จะมีขนาดสูงสุด65535 ไบต์ เขตข้อมูลนี้ เป็นเขตข้อมูลที่จำเป็นที่จะต้องระบุไว้ในทุกดาต้าแกรมเพื่อทำให้สามารถอดข้อมูลออกมาได้
บิต 32-47	**Identification** เป็นหมายเลขของดาต้าแกรมที่ส่งในกรณีที่มีการกระจายของดาต้าแกรม และนำกลับมารวมกันใหม่ จะรู้ได้ว่า มาจากดาต้าแกรมเดียวกัน
บิต 48-50	**flag** ใช้ในกรณีที่มีการแบ่งเป็นส่วนย่อยของดาต้าแกรม
บิต 51-63	**Flagment offset** ใช้ในการกำหนดตำแหน่งของข้อมูลใน 1 ดาต้าแกรมที่ถูกแยกย่อยกลับมาเรียงต่อกันในตำแหน่งของข้อมูลที่ถูกต้อง
บิต 64-71	**Time To Live** เป็นตัวเลข 8 บิตบอกช่วงเวลาของโปรแกรมสำเร็จที่ยังอยู่ในเครือข่ายได้ โดยจะกำหนดค่าเป็นจำนวนเราเตอร์สูงสุดที่ผ่านได้ ซึ่งโดยทั่วไป มีค่าอยู่ระหว่าง 32 ถึง 64 และลดค่าลงเรื่อยๆเมื่อผ่านเราเตอร์เพื่อเป็นการป้องกันโปรแกรมสำเร็จล้นเครือข่าย
บิต 72-79	**Protocol** เป็นตัวระบุว่าข้อมูลที่ ไอพี กำลังส่งอยู่นี้เป็นของโพรโตคอลอะไรเช่น ทีซีพี, ยูดีพี, ไอซีเอ็มพี
บิต 80-95	**Header checker**เป็นส่วนตรวจสอบความถูกต้องของข้อมูลในส่วนระบุโปรโตคอลเพื่อป้องกันการผิดพลาดในการส่งข้อมูล
บิต 96-127	**Source IP address** คือไอพีแอดเดรส ของผู้ส่งข้อมูล ดาต้าแกรม
บิต 128-163	**Destination IP address** คือ ไอพีแอดเดรสของปลายทางผู้รับข้อมูลดาต้าแกรม

3.7 ข้อบกพร่องของโพรโตคอลทีซีพี/ไอพี (TCP/ IP)

3.7.1 ขาดกลไกทางด้านความปลอดภัย

สำหรับโพรโตคอลทีซีพี/ไอพี แล้วหากมีการกระตุ้นที่ถูกต้องตามโพรโตคอลแล้วจะต้องมีการตอบรับเสมอหรือถ้าให้เข้าใจได้ง่ายก็คือหากถูกถามจะต้องตอบเสมอ โพรโตคอลจะกำหนดไว้ชัดเจนว่าหากมีการกระตุ้นที่ถูกต้องตามโพรโตคอลเข้าใจจะต้องตอบรับ เครื่องแม่ข่ายผู้รับจะทำหน้าที่ตรวจสอบเพียงถูกต้องตามที่โพรโตคอลกำหนดไว้หรือไม่เท่านั้น ไม่ได้กำหนดในเรื่องอื่นไว้ เช่น ปริมาณมากเกินไปหรือไม่ มีหน้าที่เพียงแต่ "ถามอะไร" แล้วก็ตอบกลับไป

ไม่มีกลไกในการตรวจสอบผู้ถาม และการจัดการกับการถามที่ผิดปกติ เช่นในกรณีปกติหากแม่ข่ายของเราได้รับการสอบถามจากไอซีเอ็มพีมา แม่ข่ายของเราไม่สามารถตรวจสอบได้ว่าผู้ส่งมีสิทธิ์ได้รับคำตอบหรือไม่หรือผู้ส่งถามซ้ำๆกันมากเกินไปแล้ว สิ่งที่แม่ข่ายจะทำได้ก็คือจะต้องตอบกลับไปด้วยคำตอบของไอซีเอ็มพีเท่านั้น และจะต้องตอบกลับไปในทุกกรณี ทั้งนี้เพราะกลไกความปลอดภัยไม่ได้อยู่ในโพรโตคอลด้วย

ด้วยช่องว่างเช่นนี้เองที่ถูกนำมาใช้ประโยชน์โดยเฉพาะนักเจาะระบบ ไม่ว่าจะเป็นการสำรวจเป้าหมายด้วยการสำรวจช่องทางการติดต่อ สำรวจเครือข่าย การทำให้ระบบปฏิเสธการทำงานบางบริการ เป็นต้น

3.7.2 การตอบรับเป็นสิ่งที่สามารถคาดหมายได้

เป็นที่แน่นอนว่าการตอบรับจะต้องเป็นสิ่งที่สามารถคาดหมายได้ เพราะการตอบรับใดๆจะต้องเป็นตามข้อกำหนดในโพรโตคอลซึ่งเผยแพร่แก่สาธารณชนอยู่แล้ว หากการตอบสนองคาดหมายไม่ได้ การสื่อสารทั้งสองทางก็ไม่เกิดขึ้น นักเจาะระบบจึงอาศัยการวิเคราะห์การตอบรับของเป้าหมายในสถานการณ์ต่างๆมาใช้ประโยชน์

3.7.3 การตอบรับกำหนดไว้ไม่ครอบคลุมทุกเงื่อนไข

ด้วยเป้าหมายของโพรโตคอลในเบื้องต้นคือ ความถูกต้อง ความมีเสถียรภาพและประสิทธิภาพของการสื่อสาร ดังนั้นเงื่อนไขข้อกำหนดต่างๆ ที่ระบุไว้นั้นก็เป็นไปเพื่อรับใช้วัตถุประสงค์ดังกล่าว จุดสำคัญอยู่ที่หากมีการสื่อสารที่ถูกต้องตามโพรโตคอล คอมพิวเตอร์ทั้งสองฝั่งจะต้องอำนวยความสะดวกให้ทำการการสื่อสารด้วยความถูกต้อง มีเสถียรภาพและประสิทธิภาพสูงสุด

แต่โพรโตคอลกลับละเลยในส่วนนี้ หากมีการสื่อสารที่ไม่ถูกต้องตามโพรโตคอลแล้วจะจัดการต่อไปอย่างไร อาจจะมีกำหนดไว้ในจุดที่สำคัญ แต่ก็ยังคงเหลือเงื่อนไขอื่นๆ อีกมากมายที่โพรโตคอลไม่ได้ระบุ ตัวอย่างเช่น ทีซีพีกำหนดให้ส่ง สัญญาณการเชื่อมต่อในตอนสร้างการเชื่อมต่อและส่งสัญญาณที่หยุดการเชื่อมต่อ ในตอนยกเลิกการติดต่อ ถามว่าหากมีการส่ง

ทีซีพีที่มีทั้ง สัญญาณการเชื่อมต่อ และสัญญาณที่หยุดการเชื่อมต่อพร้อมกันจะเกิดอะไรขึ้น ผู้รับจะตอบรับกลับไปอย่างไร แน่นอนว่าการส่งสัญญาณหรือข้อมูลใดๆ ที่ผิดข้อกำหนดในโพรโตคอลนั้นมิสามารถกระทำได้ในการทำงานปกติ เช่น หากเราจะส่งข้อมูลผ่านทีซีพี เราอาจจะเรียกส่วนต่อประสานโปรแกรมประยุกต์ (API: application program interface) หรือซ็อกเก็ต (Socket) มาทำงานที่เหลือ เอพีไอเหล่านั้นก็ไปจัดการข้อมูลในระดับล่างเองให้ถูกต้องตามโพรโตคอล แต่เอพีไอหรือ ซ็อกเก็ต เหล่านั้นก็เป็นแค่โปรแกรมชนิดหนึ่งที่ทำหน้าที่จัดระเบียบข้อมูลแล้วส่งไปยังอุปกรณ์ในระดับล่างเท่านั้น หากนักเจาะระบบโปรแกรมที่ทำหน้าที่ดังกล่าว ก็สามารถส่งข้อมูลไปยังอุปกรณ์ต่างๆได้โดยตรงเช่นกัน ถึงแม้ว่าจะต้องใช้ความรู้ในเรื่องของภาษาระดับต่ำและฮาร์ดแวร์บ้างก็ตาม แต่ก็เป็นสิ่งที่ทำได้

และเมื่อสามารถส่งข้อมูลไปยังอุปกรณ์ (Device) ได้ข้อมูลที่ส่งก็ไม่จำเป็นต้องเป็นข้อมูลที่ถูกกำหนดไว้ในโพรโตคอล ทำให้มีข้อมูลที่แปลกประหลาดและ ไม่จัดการได้ด้วยโพรโตคอล และข้อมูลเหล่านั้นก็แปรสภาพกลายมาเป็นเครื่องมือในการโจมตีได้

3.8 การตรวจจับการบุกรุก (Intrusion Detection)

การตรวจจับการบุกรุก เป็นการตรวจจับกิจกรรมที่ไม่เหมาะสม ไม่ถูกต้องหรือไม่ปกติ ซึ่งระบบการตรวจจับการบุกรุก (IDS : Intrusion Detection Systems) ซึ่งทำงานอยู่บนเครื่องแม่ข่ายเพื่อทำการตรวจจับเหตุการณ์หรือกิจกรรมต่างๆที่แอบแฝงบนแม่ข่าย เราเรียกว่าระบบตรวจจับการบุกรุกบนเครื่องแม่ข่าย (host-based IDS) และระบบการตรวจจับซึ่งทำงานบนระบบเครือข่ายเมื่อทำหน้าที่ในการตรวจสอบข้อมูลซึ่งไหลเวียนในเครือข่าย เราเรียกว่าระบบตรวจจับการบุกรุกเครือข่าย (network-based IDS)

3.8.1 ระบบตรวจจับการบุกรุกบนเครื่องแม่ข่าย (Host-base IDS)

ระบบตรวจจับจะทำการตรวจจับข้อมูล ที่ไหลเข้าและออกคอมพิวเตอร์แต่ละเครื่อง นอกจากนั้นระบบก็ยังตรวจสอบ ความสมบูรณ์ของข้อมูลของระบบ (system files) และเฝ้าดูกระบวนการที่กำลังทำงาน (processes) ที่น่าสงสัย

ระบบตรวจจับการบุกรุกบนเครื่องแม่ข่ายมี 2 ประเภทใหญ่คือ

1. ส่วนบุคคล (personal firewall/host wrappers)
2. เชื่อมต่อกับผู้แทนขายซอฟต์แวร์ (agent-based software)

ซึ่งทั้งสองชนิดจะมีประสิทธิภาพในการตรวจจับการบุกรุกจากภายในได้ดีกว่า ระบบตรวจจับการบุกรุกทางเครือข่ายแต่ประสิทธิภาพการตรวจจับการโจมตีจากภายนอกนั้นจะทำได้ดีพอๆ กัน

Host Wrappers หรือ personal firewall

สามารถที่จะทำการปรับแต่งให้ IDS ชนิดนี้ทำการตรวจสอบทุกๆกลุ่มข้อมูล (packet) บนเครือข่าย การพยายามที่จะเชื่อมต่อเข้ามา หรือการที่จะพยายามเข้าสู่ระบบ (login) เข้ามาซึ่งรวมถึงการเชื่อมต่อแบบติดต่อเข้ามา (dial-in)

Agent-based Software

สามารถที่จะเฝ้าตรวจการใช้สิทธิ์ของการใช้งาน (access) และการเปลี่ยนแปลงของข้อมูลระบบ (system files) รวมทั้งการเปลี่ยนแปลงสิทธิการใช้งาน (privilege) ของผู้ใช้

3.8.2ระบบตรวจจับการบุกรุกทางเครือข่าย (Network based IDS)

ระบบการตรวจจับการบุกรุกแบบเครือข่าย (network-based) นั้นจะทำการเฝ้าดูข้อมูลบนเครือข่ายโดยที่ระบบดังกล่าวจะทำการรับข้อมูลทั้งหมด ที่อยู่บนส่วนของเครือข่ายที่รับผิดชอบนอกเหนือจากส่วนของเครือข่ายที่รับผิดชอบ และชนิดของการสื่อสารอื่นๆ แล้วระบบดังกล่าวก็ไม่สามารถทำการตรวจจับ packet ต่างๆ จะถูกตรวจจับโดยตัวตรวจจับ (sensor) ของระบบ IDS ซึ่ง sensor จะมองเห็นเฉพาะ packet ที่ผ่านส่วนของเครือข่ายที่ sensor นั้นติดอยู่ packet ต่างๆ จะเป็นที่สนใจของ sensor ก็ต่อเมื่อ packet นั้นเข้ากับลายเซ็น (signature) ที่กำหนดซึ่งปกติแล้ว signature จะมี3ประเภทคือ

1 ลายเซ็นข้อความ (string signatures)

2 ลายเซ็นของช่องทางการติดต่อ (port signatures)

3 ส่วนหัวของลายเซ็นที่ผิดพลาด (header condition signatures)

String signature

จะมองหา text string ซึ่งอาจบ่งบอกถึงการโจมตี ตัวอย่างเช่น "cat" + + "7% host" อาจทำให้ระบบยูนิกซ์เกิดช่องโหว่ต่อการโจมตีบนเครือข่าย

Port signatures

จะเฝ้าดูการพยายามติดต่อเข้ามาทาง port ที่รู้จักกันดี และมักจะถูกโจมตี เช่น telnet จะใช้ TCP port 23, FTP จะใช้ TCP port 21/20 SUNRPC ใช้ TCP/UDP port 111 และ IMAP จะใช้ TCP port 143 ซึ่งถ้า ระบบของเราไม่ได้เปิด port ดังกล่าว แต่มีการพยายามเชื่อมต่อเข้ามา แสดงว่า packets ดังกล่าว อาจจะมีประสงค์ร้ายก็ได้

Header signatures

พยายามมองหา combination ที่อันตรายและผิดกฎของ packet header ตัวอย่างที่เห็นได้ชัดของ header signature คือ TCP packet ซึ่งมีทั้ง SYN และ FIN Flags

3.8.3 การจัดการทางด้านความปลอดภัย

ระบบการรักษาความปลอดภัยให้กับคอมพิวเตอร์จะมีประสิทธิภาพมากที่สุด เมื่อองค์กรนั้นๆ นำระดับของการรักษาความปลอดภัยหลายๆ ชั้นมาใช้ มักจะมีการเข้าใจผิดว่า แค่ด่านกันบุกรุก (firewall) ก็เพียงพอแล้ว ท่านควรมีแบบจำลองทางด้านความปลอดภัย (security model) อื่นๆด้วยเพื่อเพิ่มประสิทธิภาพในการรักษาความปลอดภัยให้แก่คอมพิวเตอร์ขององค์ก อันได้แก่

1. นโยบายทางด้านความปลอดภัย
2. การรักษาความปลอดภัยของระบบแม่ข่าย (host system)
3. การตรวจสอบระบบ
4. การรักษาความปลอดภัยให้แก่อุปกรณ์จัดเส้นทาง (router)
5. การใช้ firewall
6. ระบบการป้องกันการบุกรุก
7. แบบแผนการตอบสนองต่อเหตุการณ์ต่างๆ

แต่ละ layer จะมีประสิทธิภาพภายในการป้องกันการบุกรุกในระดับหนึ่ง หากผู้บุกรุกสามารถผ่าน layer มาได้หนึ่ง นั่นไม่ได้หมายความว่า ระบบของท่านจะถูก compromise เลย แต่ละ layer จะมีความเกี่ยวข้องกันอยู่ ถึงแม้ว่าท่านอาจจะทำการแยกประยุกต์ (implement)ในแต่ละชั้น แต่จะประสิทธิภาพสูงสุด เมื่อทั้งหมดถูกทำการแยกประยุกต์ ด้วยกัน จะเห็นได้ว่า IDS เป็นเพียงแค่ส่วนประกอบหนึ่งของ security model ที่มีประสิทธิภาพ

3.8.4 กระบวนการตรวจจับการบุกรุก

กระบวนการตรวจจับการบุกรุกนั้นสามารถทำได้ 2 วิธีคือ

1. ฐานความรู้ (knowledge-based)
2. ฐานของพฤติกรรม (behavior-based)

Knowledge-based IDS

Knowledge-based IDS จะอาศัยข้อมูลที่เกี่ยวกับการโจมตีชนิดต่างๆ พร้อมทั้งช่องโหว่ของระบบ ในการตรวจจับการให้ใช้ช่องโหว่ต่างๆ เมื่อความพยายามในการเข้าใช้นั้นถูกจับได้ IDS ก็จะทำการแจ้งเตือน เพราะฉะนั้นจะเห็นได้ว่า ความสมบูรณ์ และประสิทธิภาพของ IDS ชนิดนี้จะต้องขึ้นอยู่กับความทันสมัยของข้อมูลเกี่ยวกับการโจมตีต่างๆ

ข้อดีของวิธีการแบบนี้คือ อัตราการเกิดการแจ้งเตือนผิดๆ นั้นจะต่ำ และข้อมูลที่ได้จากIDS นั้นจะมีรายละเอียดที่ดีทำให้ง่ายต่อผู้ใช้ในการป้องกันและแก้ไขการโจมตี

ข้อเสียของวิธีนี้คือ ความยากในการรวบรวมข้อมูล เกี่ยวกับรูปแบบการโจมตี และการปรับปรุงข้อมูลเกี่ยวกับช่องโหว่ต่างๆ ให้ทันสมัยอยู่เสมอ เนื่องจากข้อมูลต่างๆ นั้น

ขึ้นอยู่กับระบบปฏิบัติการ, รูปแบบ,แพลตฟอร์ม (platform) และโปรแกรมประยุกต์ (application)

นอกจากชั้นการตรวจจับการโจมตีจากภายใน นั้นทำได้ยากเนื่องจากการโจมตีจากภายในเกี่ยวกับการละเมิดสิทธิของ user ซึ่งไม่ได้เกี่ยวข้องกับช่องโหว่แต่อย่างใด

Behavior-basedIDS

กระบวนการของการตรวจจับการบุกรุกแบบนี้คือ จะมีการแจ้งเตือนเมื่อระบบมีการตรวจพบความเบี่ยงเบนและความผิดปกติของระบบหรือของผู้ใช้จากการใช้ระบบปกติ ซึ่งในรูปแบบของพฤติกรรมที่เป็นปกตินั้น จะถูกรวบรวมจากข้อมูลอ้างอิงต่างๆ หลังจากนั้น IDS จึงจะทำการเปรียบเทียบระหว่างพฤติกรรมในขณะนั้นกับรูปแบบอ้างอิง ดังนั้นจะเห็นได้ว่าการเตือนที่ผิดพลาด (false alarm) จะเกิดขึ้นได้บ่อยครั้ง

ข้อดีของการตรวจจับโดยใช้เทคนิคลักษณะนี้ คือสามารถที่จะตรวจจับแบบการบุกรุกแบบใหม่ๆ ที่ไม่เคยมีมาก่อน และความเกี่ยวโยงกับระบบปฏิบัติการค่อนข้างต่ำ รวมทั้งยังสามารถที่จะตรวจจับการบุกรุกที่ไม่ได้โจมตีช่องโหว่ เช่น การโจมตีจากภายใน ข้อเสียที่สำคัญที่สุดนั่นก็คือ false alarm จะค่อนข้างสูง ในช่วงของการศึกษาพฤติกรรมของระบบ และเนื่องจากพฤติกรรมจะเปลี่ยนแปลงอยู่ตลอดเวลา เพราะฉะนั้น IDS ก็ต้องใช้เวลาในการศึกษา และเป็นเหตุให้ IDS ขัดข้องหรืออาจทำให้เกิด false alarm มากขึ้น

3.8.5 ระบบตรวจจับการบุกรุกทางเครือข่าย(NIDS: Network Instrusion Detection System)

การบุกรุก (Intrusion) คือความพยายาม ที่จะเจาะเข้าสู่ระบบหรือการใช้ระบบในทางที่ผิด (misuse) ระบบการตรวจจับการบุกรุก (Intrusion detection system - IDS) คือ ระบบที่จะทำการตรวจจับการบุกรุกดังกล่าวข้างต้น ซึ่งIDS จะแบ่งออกเป็น 3 ประเภท คือ

1. Network Intrusion Detection System (NIDS) โดยจะทำการเฝ้าดู packet ที่วิ่งผ่านสายส่ง (wire) ในเครือข่ายและพยายามที่จะค้นหาว่า hacker หรือ cracker พยายามที่จะเจาะเข้าสู่ระบบ ซึ่งตัวอย่างที่เห็นได้ชัด คือระบบที่จะเฝ้าตรวจ TCP connection request หรือว่า SYN ที่พยายามจะเชื่อมต่อมายัง port ต่างๆ ของเครื่องเป้าหมาย ซึ่ง NIDS นั้นอาจจะถูกติดตั้งบนเครื่องเป้าหมายเอง และจะคอยตรวจทุกการสื่อสาร (traffic) ของตัวเอง หรืออาจจะถูกติดตั้งบนเครื่องที่แยกอยู่ต่างหากและจะคอยตรวจทุก packet ที่ผ่านมาในเครือข่าย

2. System Integrity Verifiers (SIV) จะคอยตรวจสอบ system files ว่ามีการเปลี่ยนแปลงเกิดขึ้นหรือไม่ ซึ่งขณะเดียวกัน SYN อาจจะคอยตรวจสอบองค์ประกอบ (components) อื่นๆ อย่างเช่น windows registry หรือการตั้งเวลาการทำงาน (cron configuration) หรืออาจจะตรวจจับเมื่อผู้ใช้ปกติพยายามที่จะใช้สิทธิ์ของผู้ดูและระบบ (root หรือ admin) ซึ่งผลิตภัณฑ์ส่วนใหญ่ที่เป็น SYN มักจะเป็นแค่เครื่องมือมากกว่าระบบที่สมบูรณ์แบบ อย่างเช่นในกรณีของโปรแกรม Tripwire

จะตรวจจับการเปลี่ยนแปลงของ system files ที่สำคัญ แต่จะไม่มีการแจ้งเตือนที่เป็นแบบตามเวลาจริง (real time)

3. Log File Monitors (LFM) จะทำการเฝ้าดูไฟล์บันทึก (log files) ต่างๆที่สร้างขึ้นมาโดยบริการในเครือข่าย ซึ่ง LFM จะค้นหารูปแบบของ log files ที่จะบ่งบอกถึงการบุกรุก ตัวอย่างเช่น parser ของ HTTP server log files

3.9 ผู้บุกรุกระบบ (Hacker and Cracker)

คำที่ใช้แทนผู้บุกรุกคือ "hacker" และ "cracker" ซึ่ง hacker คือบุคคลที่ชอบเจาะเข้าสู่สิ่งต่างๆ hacker ที่ดีคือ บุคคลที่พยายามที่จะเข้าสู่ระบบคอมพิวเตอร์ของตัวเองและพยายามที่จะเข้าใจการทำงานของมัน แต่ hacker ที่เป็นผู้ร้าย คือบุคคลที่พยายามเจาะเข้าสู่ระบบของผู้อื่น ซึ่ง hacker ฝ่ายดีพยายามที่จะให้สื่อต่างๆ ใช้คำว่า cracker แทนแต่ยังไงก็แล้ว สำหรับบุคคลที่ต่างๆ พยายามจะเจาะเข้าสู่ระบบของเรา เราจะเรียกว่า ผู้บุกรุก "intruder" ซึ่งผู้บุกรุกจะถูกแบ่งเป็น 2 ประเภท คือ

1. จากภายนอก (Outsides)

หมายถึงผู้บุกรุกจากภายนอกเครือข่ายของท่านและบุคคลที่อาจจะโจมตีมาจากภายนอกเช่น การเปลี่ยนแปลงหน้ากากของ web server ของท่านหรือการ forward mail ผ่านทาง e-mail server ซึ่งการบุกรุกจากภายนอกอาจมาจาก Internet, การ dial-up, การบุกเข้าไปหรือเครือข่ายของคู่ค้าที่ทำการเชื่อมต่อมายังเครือข่ายของท่าน

2. จากภายใน (Insider)

คือ ผู้บุกรุกที่มีสิทธิ์ในการใช้เครือข่ายภายใน รวมทั้งผู้ใช้ที่ใช้สิทธิ์ในทางที่ผิด หรือการทำลักลอบใช้สิทธิ์ของผู้ใช้คนอื่นๆ ที่มีสิทธิ์เหนือกว่า

3.9.1 วิธีการในการเจาะระบบ

มี 3 ทางหลักๆ ที่ผู้บุกรุกจะเจาะเข้าสู่ระบบ คือ

1. Physical Intrusion ถ้าหากว่าผู้บุกรุกมีการเชื่อมต่อ ทางกายภาพกับเครื่องหรือระบบเครือข่าย การเจาะเข้าสู่ระบบจะเกิดขึ้นได้

2. System Intrusion การบุกรุกในลักษณะนี้สมมติว่าผู้บุกรุกมี account เรียบร้อยแล้วแต่มีสิทธิ์ต่ำ ถ้าไม่ได้มีการ update patch ให้กับระบบผู้บุกรุกจะใช้ช่องโหว่ของระบบในการครอบครองสิทธิ์ของ ผู้ดูแลระบบ

3. Remote Intrusion ผู้บุกรุกพยายามที่จะเจาะเข้าสู่ระบบข้ามเครือข่าย ซึ่งผู้บุกรุกจะไม่มีสิทธิ์ใดๆ เลยบนเครือข่ายนั้น

3.9.2 สาเหตุที่ทำให้ผู้เจาะระบบสามารถเจาะระบบได้

Software bugs

จะปรากฏอยู่ใน server daemons, โปรแกรมต่างๆ ของ client ระบบปฏิบัติการ ซึ่งสามารถจะแบ่ง Software bugs ออกเป็นประเภทต่างๆได้ดังนี้

1. Buffer Overflows ช่องโหว่ของความปลอดภัยคอมพิวเตอร์จะเกิดจากปัญหานี้ ตัวอย่างของ Software bugs เช่น โปรแกรมเมอร์ได้ทำการตั้งค่าจำนวน characters ที่จะรับ login username เป็น 256 เนื่องจากเขาคิดว่าคงจะไม่มีใครที่จะใช้ username ที่ยาวกว่า 256 characters ตัวอย่างเช่น 300 characters จะเกิดอะไรขึ้นกับอีก 50 charcaters ที่เหลือ ซึ่งอาจประกอบด้วย code ที่จะถูก ทำงาน โดย server และทำให้ hacker เจาะเข้าสู่ระบบได้ hacker จะทำการค้น bugs เหล่านี้โดยวิธีต่างๆ เช่น

- อาจจะค้นหาได้จาก internet
- นักเจาะระบบอาจจะทำการศึกษา โปรแกรมนั้นๆ โดยตัวเอง
- นักเจาะระบบอาจจะตรวจสอบทุกๆ ส่วนของโปรแกรมที่มีส่วนรับข้อมูล และพยายามที่จะ overflow โดยใส่ random ข้อมูล เข้าไป ซึ่งถ้าโปรแกรม นั้นหยุดการทำงานอาจจะทำให้นักเจาะระบบสามารถเจาะเข้าสู่ระบบนั้นได้ ซึ่งปัญหาดังกล่าว จะเกิดขึ้นกับโปรแกรมซึ่งเขียนด้วย C/C++ แต่จะไม่เจอกับโปรแกรมซึ่งเขียนด้วยจาวา

2. Unexpectes Combinations โดยทั่วๆ ไปโปรแกรมจะประกอบ code หลายๆ ชั้น ซึ่งชั้นที่อยู่ต่ำที่สุดคือระบบปฏิบัติการ ซึ่งเมื่อผู้บุกรุกส่งส่วนรับข้อมูลเข้าสู่โปรแกรมส่วนรับข้อมูลนั้นอาจจะไม่มีความหมายอะไรเลยสำหรับชั้นหนึ่ง แต่อาจจะทำให้เกิดผลกับอีกชั้นหนึ่ง จะเห็นได้ว่าโดยส่วนใหญ่ ภาษา PERL คือภาษาที่จะทำการประมวลผลบนเว็บซึ่งปกติแล้ว PERL จะส่งข้อมูลให้อีกโปรแกรมหนึ่ง

3. ส่วนข้อมูลเข้า (Input) ที่ไม่ถูกการประมวลผล โดยส่วนใหญ่ผู้เขียนโปรแกรมจะเขียนโปรแกรม ที่จะจัดการกับข้อมูลที่ถูกต้อง แต่จะไม่พิจารณากรณีที่ข้อมูลเมื่อไม่ได้ตรงตามข้อกำหนด (specification)

4. RaceConditions เนื่องจากปัจจุบันระบบส่วนใหญ่จะเป็นแบบ "Multitasking หรือว่า Multithreaded" หมายถึงว่าโปรแกรมหลายๆโปรแกรมสามารถถูก ทำงาน พร้อมๆ กันได้ ซึ่งจะเป็นอันตรายอย่างเช่น ถ้า โปรแกรม นั้นๆ ใช้ ข้อมูล เดียวกัน ตัวอย่างเช่น โปรแกรม A และ B จำเป็นที่จะต้องแก้ไขแฟ้มข้อมูล (file) เดียวกัน ในการที่จะแก้ไข file นี้ โปรแกรมจะต้องอ่าน ไฟล์ ไปเก็บไว้ใน หน่วยความจำ แล้วทำการแก้ไข เนื้อความใน memeory แล้วคัดลอก หน่วยความจำ กลับเข้าสู่ไฟล์ race conditions จะเกิดขึ้นเมื่อ โปรแกรม A อ่านไฟล์เข้าสู่ หน่วยความจำ แล้วทำการแก้ไข แต่ก่อนที่ A จะเขียนกลับลงสู่ไฟล์ โปรแกรม B ได้ทำการอ่าน แก้ไข เขียน ลงสู่ไฟล์เรียนร้อยแล้ว หลังจากนั้น A ทำการเขียน กลับไปในไฟล์ จะเห็นได้ว่าการแก้ไขโปรแกรมทั้งหมดของ โปรแกรม B จะหายไป

แต่ยังไงก็แล้วแต่ race conditions ค่อนข้างจะเกิดขึ้นได้ยาก ผู้บุกรุกต้องทำการทดลองเป็นพันๆ ครั้ง จึงจะสามารถเจาะเข้าสู่ระบบได้

System Configuration

bugsที่เกิดจากSystemConfiguration สามารถจำแนกได้ดังต่อไปนี้

1. Default Configurations ระบบส่วนใหญ่จะถูกจัดส่งจากผู้ขายด้วย default configuration ซึ่งง่ายในการใช้แต่นั้นก็หมายถึงง่ายในการเจาะด้วย

2. ความเกียจคร้านของผู้ดูแลระบบ มีระบบอยู่ไม่น้อยทีเดียวที่ถูกปรับแต่งให้ไม่ต้องใส่รหัสผ่านของผู้ดูและระบบเนื่องจากผู้ดูแลระบบเกียจคร้านที่จะปรับแต่ง แต่อยากให้เครื่องทำงานได้เร็วที่สุด เนื่องจากว่าการแก้ไขรหัสผ่าน ทีหลังทำได้ไม่ง่ายเลย ซึ่งจุดนี้อาจทำให้ผู้บุกรุกเจาะเข้าสู่ระบบได้

3. Hole Creation บางครั้ง ผู้ดูแลระบบ จะเปิดช่องโหว่บนเครื่องไว้ ซึ่งคู่มือการดูแลระบบส่วนใหญ่จะแนะนำให้ทำการปิดทุกอย่างที่ไม่จำเป็นเพื่อป้องกันการเกิดช่องโหว่ขึ้นมาโดยไม่ตั้งใจ

รหัสผ่าน Cracking

1. การเลือกใช้ รหัสผ่าน ที่ค่อนข้างอ่อน คนส่วนใหญ่จะเลือกใช้ รหัสผ่าน ที่เป็นชื่อของตนเอง, ชื่อลูก, ชื่อสามี, ภรรยา, สัตว์เลี้ยง, รุ่นของรถ หรือบางคนอาจใช้เป็น รหัสผ่าน หรืออาจไม่ใส่เลยจะเห็นว่าผู้บุกรุกทำการเดาสุ่มรหัสผ่านเหล่านี้ได้ง่าย

2. Dictionary Attacks ขั้นตอนต่อไปนี้ ผู้บุกรุกอาจจะใช้ โปรแกรม ที่จะทำการ ถอดรหัสผ่าน โดยที่ โปรแกรม ดังกล่าวจะเลือกคำที่เป็นไปได้ในดิกชั่นนารี dictionary attacks แล้วเปรียบเทียบกับคำในดิกชั่นนารี(dictionary)ที่ถูกเข้ารหัส

3. Brute force attacks จะคล้ายกับ dictionary attacks ผู้บุกรุกจะพยายามนำ ตัวอักษรต่างๆ มาผสมกันเป็น รหัสผ่าน อย่างเช่น รหัสผ่าน ที่มีอักษร 4 ตัว และเป็น ตัวเล็กเพียงอย่างเดียว อาจจะถูกถอดรหัสภายในเวลาแค่ไม่กี่นาที หรือถ้าเป็น รหัสผ่าน ความยาว 7 ตัวอักษร ทั้งที่เป็นตัวใหญ่และ ตัวเล็กอาจต้องใช้เวลาหลายเดือนในการถอดรหัส

Sniffing Unsecured traffic

สามารถทำได้หลายวิธีด้วยกัน

1. ตัวกลางที่ใช้ร่วมกัน ตัวอย่างเช่น ถ้าเป็น Etemet รุ่นเก่าๆ เพียงแค่นำโปรแกรมดักจับข้อมูล (sniffer) ไปติดตั้งบน wire ก็สามารถมองเห็นการสื่อสาร (traffic) ทั้งหมดได้ แต่จะยากขึ้นสำหรับ Switched Ethernet

2. Server Sniffing การติดตั้ง Sniffer โปรแกรม บน server นั้นอาจได้ข้อมูลที่จะนำไปใช้ในการเจาะเข้าสู่ client machine ได้

ข้อบกพร่องของการออกแบบ

ถึงแม้ว่าการใช้ Software จะถูกต้องตามออกแบบ (design) แต่อาจจะมี bugs ในตัว designs ที่จะนำไปสู่การเจาะระบบได้ เช่น

1. TCP/IP protocol เนื่องจาก TCP/IP ได้ถูกออกมาก่อนที่วิวัฒนาการทางด้าน hacking จะเหมือนในปัจจุบัน เพราะฉะนั้นเป็นไปได้ที่อาจจะมีข้อผิดพลาดทางด้านการออกแบบที่จะนำไปสู่ปัญหาทางด้านความปลอดภัยได้ ตัวอย่างเช่น Smurt attack, IP spoofing หรือ SYN floods ได้มีการพัฒนา IPSec เพื่อที่จะแก้ปัญหาต่างๆ เหล่านี้ แต่ยังไม่มีการใช้อย่างแพร่หลาย

2.Unix มี flaws ในระบบปฏิบัติการ UNIX ซึ่งอาจทำให้ระบบถูกเจาะได้ซึ่งปัญหาที่สำคัญที่สุดคือ access control system

3.10 สถาปัตยกรรมของระบบตรวจจับการบุกรุก (IDS Architecture)

กลวิธีในการตรวจจับการบุกรุก

1. การตรวจจับความผิดปกติ

วิธีการตรวจจับการบุกรุกใช้โดยปกติทั่วไป คือ การตรวจความผิดปกติโดยการพิจารณาจากสถิติต่าง ๆ เช่น CPU Utilization,การใช้ disk, การ login ของ user, การใช้ file ประโยชน์ของการใช้วิธีนี้คือสามารถจับความผิดปกติต่าง ๆ โดยที่ไม่ต้องรู้ตัวสาเหตุของความผิดปกตินั้น

2. การตรวจสอบรูปแบบของการโจมตี

โปรแกรม IDS โดยส่วนใหญ่จะใช้วิธีการตรวจสอบหรือศึกษารูปแบบการโจมตีซึ่งเป็นที่รู้จักกันดี นั่นหมายความว่า เทคนิคต่าง ๆ ที่ hacker ใช้ก็จะถูก code เข้าสู่ระบบเพื่อทำการตรวจจับเทคนิคนั้น วิธีการโดยทั่วไปคือการตรวจดู content ใน packet ว่าประกอบด้วย pattern ที่อาจจะแสดงถึงการพยายาม access เข้าสู่ระบบ เช่น ถ้า packet ประกอบด้วย "cgi-bin/phf?" หมายถึงการพยายาม access CGI script ที่อยู่บน web server และ IDS บางระบบอาจจะสร้างขึ้นมาจากฐานข้อมูลของ string เหล่านี้จำนวนมาก

เทคนิควิธีการในการตรวจสอบการโจมตี

เนื่องจาก traffic ของ IP Datagrams จะไหลมาตาม wire หลังจากที่ NIDS ได้ทำการ capture Datagrams ดังกล่าว ก็จะทำการ reassemble IP Datagrams และ IP Streams และทำการตรวจสอบ stream โดยการใช้เทคนิคดังต่อไปนี้

1. การตรวจสอบ Protocol Stack

มีการบุกรุกหลายชนิดซึ่งจะใช้การ violation IP, TCP, UDD และ ICMP โพรโตคอล เพื่อที่จะทำการโจมตี

2. การตรวจสอบ Appication Protocol

การบุกรุกบางชนิดใช้ invalid protocol behavior เพื่อที่จะให้เกิดการตรวจจับ ที่มีผลดีและประสิทธิภาพ จำเป็นที่จะต้องมีการ re-implement appication-layer protocol เพื่อที่จะได้ทำการตรวจจับพฤติกรรมที่น่าสงสัย

3. การสร้างเหตุการณ์ที่สามารถจะตรวจจับได้ใหม่

ระบบป้องกันการบุกรุกของเครือข่ายอาจใช้เป็นส่วนเสริมในการตรวจสอบเครือข่าย โดยร่วมกับโปรแกรมจัดการเครือข่าย ตัวอย่างเช่น หลังจากที่ NIDS ได้ทำการ log application layer protocol ยังใช้บนเครื่อง ระบบ log ของ systems ก็จะนำข้อมูลดังกล่าวไปใช้ร่วมกับเหตุการณ์อื่นๆ บนเครือข่าย

การทำงานหลังจากพบการโจมตี

1.SNMP Trap

ส่ง SNMP Trap Datagram ไปยังโปรแกรมจัดการเครือข่าย เช่น HP OpenView, Tiuoli

2.NT event

ส่งเหตุการณ์ที่เกิดขึ้นไปยัง WinNT event log

3.ส่ง E-mail

ส่ง E-mail ไปยัง admin

4.ทำการบันทึกการโจมตี

ควรบันทึกข้อมูลการโจมตี เช่น เวลา, ip address ของผู้บุกรุก, IP address และ port ของเหยื่อ หรือข้อมูลของโปรโตคอล

5.รวบรวมหลักฐาน

ทำการรวบรวมข้อมูลของ packet เพื่อประโยชน์ในการวิเคราะห์ภายหลัง

6.เปิดโปรแกรมอื่น

ทำการเปิดโปรแกรมอื่นแยกต่างหากเพื่อจัดการกับเหตุการณ์ที่เกิดขึ้น

7.ตัดการติดต่อของ TCP FIN เพื่อเลิกการติดต่อ

สถานที่ที่ควรติดตั้งระบบป้องกันการบุกรุก

1.เครือข่ายแม่ข่าย ระบบตรวจจับการบุกรุกนั้นจะสามารถถูกติดตั้ง บนเครื่องแม่ข่าย ตัวอย่างเช่น switched network ซึ่งจะเห็นได้ว่าเครื่องที่รัน windows นั้นจะไม่สามารถที่จะปกป้องตัวเองได้เลย เนื่องจากไม่มีความสามารถในการบันทึกเหตุการณ์ (log) เพื่อที่จะป้อนเข้าสู่ระบบตรวจจับการบุกรุกแบบ host-based ด้วยเหตุนี้อาจมีคนรันโปรแกรม รหัสผ่าน cracker โดยที่ไม่มีใครทราบได้เลย แต่ NIDS จะเป็น software ซึ่งสามารถตรวจจับการบุกรุกดังกล่าวได้

2.รอบนอกของเครือข่ายIDS จะทำงานมีประสิทธิภาพสูงสุดเมื่อติดตั้งไว้ที่ network perimeter ตัวอย่างเช่น ทั้งสองด้านของ firewall หรือใกล้ ๆ กับ dialup server หรือบน links ที่จะเชื่อมต่อไปยัง partner networks

3. WAN backbone เป็นอีกตำแหน่งหน้าที่ IDS จะมีประสิทธิภาพการทำงานสูง เนื่องจากบ่อยครั้งที่มีการบุกรุกภายนอกสู่เครือข่ายของหน่วยงาน

4. Server Farms โดยปกติแล้ว server จะถูกติดตั้งไว้กับ network ของตัวเอง แต่ปัญหาที่เกิดขึ้นคือ IDS ไม่สามารถรองรับขนาดของ traffic ได้ สำหรับ server ที่มีความสำคัญมากๆ ท่านอาจจะติดตั้ง dedicated IDS สำหรับ server นั้น และเนื่องจาก IDS ควรน่าจะใช้กับ application server มากกว่า

5.LAN Backbones IDS ไม่นิยมนำมาใช้กับ LAN backbones เนื่องจากว่า LAN backbones มีขนาดของ traffic ที่ค่อนข้างสูง แต่บาง venders บางรายที่ใช้ IDS กับ switch

การทำงานร่วมกันระหว่าง IDS กับระบบความปลอดภัยอื่นๆ

1. ติดตั้ง firewall ระหว่าง network ที่มีความต้องการด้าน security ที่ต่างกัน
2. ใช้โปรแกรมตรวจหาช่องโหว่ของ network
3. ใช้โปรแกรมตรวจ host policy เพื่อให้แน่ใจว่าการเปลี่ยนแปลงที่เกิดขึ้นกับ host นั้นไม่มีความผิดพลาดใด ๆ เกิดขึ้น
4. ใช้ระบบตรวจจับการบุกรุกเครือข่าย (NIDS) และโปรแกรม packet sniffing
5. ใช้ host-based intrusion detection system และ virus scanner เพื่อแจ้งเตือนการบุกรุก
6. เขียนนโยบายในการปฏิบัติต่อการบุกรุกให้ง่ายและปฏิบัติตามได้ง่าย

ข้อควรคำนึงถึงในข้อควรคำนึงถึงในการ implement ระบบการตรวจจับการบุกรุก

1 . ระบบปฏิบัติการ WinNT และ UNIX จะมาพร้อมกับความสามารถในการ logging/auditing ซึ่งใช้ในการเฝ้าตรวจทรัพยากรซึ่งมีความเสี่ยงสูง ซึ่งในหัวข้อต่อ ๆ ไปจะกล่าวถึงวิธีการ configure Windows และ UNIX ในการตรวจจับการบุกรุก

2 . Services เช่น Web server , e-mail server และ database

3 .ระบบตรวจจับการบุกรุกเครือข่าย ที่จะเฝ้าตรวจ traffic ในเครือข่ายเพื่อค้นหาการบุกรุก

4 . Firewall ซึ่งโดยปกติจะมีความสามารถในการตรวจจับการบุกรุกได้ด้วย เนื่องจากจุดประสงค์หลักของ firewall คือการ block การบุกรุก ดังนั้นควรจะทำการตรวจจับการบุกรุกได้ด้วย

5. ระบบการจัดการเครือข่าย เช่น OpenView ซึ่งเป็นเครื่องมือสำหรับ network manager ในการแจ้งเตือน เมื่อมีพฤติกรรมที่น่าสงสัยเกิดขึ้น

บทที่ 4
รูปแบบการบุกรุก

4.1 ดักอ่านข้อมูลด้วย Packet Sniffer

Packet Sniffer เป็นเครื่องมือสำหรับการดักอ่านข้อมูลที่สื่อสารอยู่บนเน็ตเวิร์คเพื่อให้ได้มาซึ่งข้อมูลของผู้อื่นที่ไม่ใช่ของตน มีลักษณะการนำไปใช้งานใกล้เคียงกับการดักฟังทางโทรศัพท์ การนำ Packet Sniffer มาใช้อย่างได้ผลได้ทำให้ความเชื่อถือในความปลอดภัยของการสื่อสารในรูปแบบอิเล็กทรอนิกส์ลดต่ำลงไปมาก หลังจากที่เคยเชื่อว่าการสื่อสารระหว่างโฮสต์จะเป็นที่รู้กันระหว่างโฮสต์สื่อสารเท่านั้น กลับเป็นว่าใครก็ตามที่ใช้เน็ตเวิร์คร่วมกันกับโฮสต์ของเราสามารถที่จะแอบอ่านข้อมูลทุกๆอย่างที่เราสื่อสารกันได้โดยง่าย ซึ่งสามารถจะกระทำได้อย่างไร้ร่องรอยและยากแก่การป้องกัน

4.1.1 องค์ประกอบของสนิฟเฟอร์

สนิฟเฟอร์ เป็นเครื่องหมายทางการค้าซึ่งจดทะเบียนโดยบริษัท Network Associates Inc ของสหรัฐอเมริกา เพื่อใช้ในผลิตภัณฑ์ของตนชื่อ Sniffer Network Analyzer ซึ่งเป็นโปรแกรมวิเคราะห์การใช้งานเน็ตเวิร์คออกไปตามโปรโตคอลที่ใช้งานอยู่ เพื่อช่วยในการวางแผน ตรวจสอบและแก้ไขข้อบกพร่องที่อาจมีขึ้นในเน็ตเวิร์ค

สนิฟเฟอร์ที่สามารถทำงานได้นั้นจะต้องมีองค์ประกอบพื้นฐาน 4 ส่วนคือ

1. Hardware หมายถึงอุปกรณ์อิเล็กทรอนิกส์ต่างๆ ที่จะสามารถดักการอ่านสัญญาณจากเน็ตเวิร์คเข้ามาได้และสามารถนำสัญญาณที่ได้ทำการส่งต่อไป เพื่อประมวลผลออกมาเป็นข้อมูลทางคอมพิวเตอร์ได้มีหน้าที่หลักคือการจัดการกับการรับข้อมูลในระดับฟิสิคัล เช่นสัญญาณรบกวนการแก้ไขข้อผิดพลาดของสัญญาณ ซึ่งอุปกรณ์ทั่วไปคือเน็ตเวิร์คอะแดปเตอร์นั่นเอง

2. Driver เป็นโปรแกรมระดับล่างที่ควบคุมการดักข้อมูลของฮาร์ดแวร์

3. Buffer เป็นหน่วยความจำที่ใช้พักข้อมูลโดยจากการดักมาได้ของ Driver โดยจะทำการจัดเก็บเพียงชั่วคราวและทำการหมุนเวียนข้อมูลใหม่เข้ามาเสมอ เมื่อมีข้อมูลใดนั้นได้ปรากฏขึ้นบนเน็ตเวิร์ค กลไกการนำข้อมูลจากไดร์เวอร์มาเก็บยังบัฟเฟอร์นี้จะเป็นตัวบ่งบอกสมรรถนะของการดักข้อมูลของสนิฟเฟอร์นั้นว่าจะสามารถดักข้อมูลได้ความเร็วสูงสุดเท่าใด ถ้าหากกระบวนการนำข้อมูลไปเก็บเป็นไปอย่างล่าช้า ก็ย่อมทำให้สนิฟเฟอร์ไม่สามารถดักข้อมูลที่อยู่บนเน็ตเวิร์คได้ทันและต้องปล่อยข้อมูลนั้นทิ้งไป

4. Software เพื่อทำหน้าที่จัดการข้อมูลที่ได้รับเข้ามาโดยการประมวลผลตามวัตถุประสงค์ของการดักอ่านข้อมูล เนื่องจากข้อมูลที่ดักมาได้นั้นจะเป็นข้อมูลระดับต่ำ คือ Data Link Layer ซึ่งจะมีข้อมูลที่ยังไม่ได้ผ่านการดีมัลติเพล็กซ์ และจัดรูปแบบให้เข้าใจได้สิ่งที่จะได้จะเป็นข้อมูลเลขฐาน สอง 0 กับ 1 จำนวนมหาศาลที่ต้องมาแปลความหมายกันอีก ซึ่งข้อมูลจากการสื่อสารทุกๆ โฮสต์ที่ใช้เน็ตเวิร์คร่วมกันอยู่ ผสมกันอย่างไร้ระเบียบและไม่มีการแยกแยะกันว่าเป็นการสื่อสารเรื่องอะไร ระหว่างโฮสต์ใดกับโฮสต์ใด การที่จะแปลความหมายของข้อมูลเหล่านี้ได้ก็จำเป็นที่จะต้องมีโปรแกรมสำหรับทำหน้าที่จัดการกับกองข้อมูลขนาดใหญ่นี้ ให้อยู่ในรูปแบบที่เข้าใจกันมากขึ้น

สำหรับองค์ประกอบทั้งหมดของสนิฟเฟอร์นั้น สามารถดัดแปลงได้จากเครื่องคอมพิวเตอร์ที่ใช้งานทั่วไปเพราะฮาร์ดแวร์ที่ใช้กับคอมพิวเตอร์ เพื่อทำการสื่อสารข้อมูลในเน็ตเวิร์คนั้นก็ได้ถูกออกแบบให้มีฟังก์ชันการสื่อสารข้อมูลที่สลับซับซ้อน และมีฟังก์ชั่นบางส่วนที่สามารถดัดแปลงเป็นสนิฟเฟอร์ได้ไม่ยาก จึงมีผู้ใช้เครื่องคอมพิวเตอร์ทั่วไปทำหน้าที่เป็นสนิฟเฟอร์ รวมทั้งแฮกเกอร์ก็มักดัดแปลงคอมพิวเตอร์ของเหยื่อเป็นสนิฟเฟอร์เพื่อแอบอ่านข้อมูลของผู้อื่นในเน็ตเวิร์คอยู่เสมอ

4.1.2 การทำงานของสนิฟเฟอร์

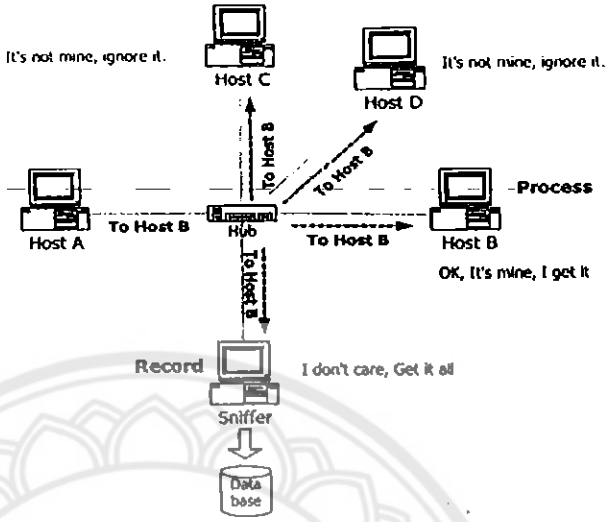
การที่สนิฟเฟอร์สามารถดักข้อมูลที่อยู่บนเน็ตเวิร์คได้นั้นมีสาเหตุที่สำคัญด้วยกันคือ ด้วยลักษณะโปรโตรคอลอีเธอร์เนตที่ใช้หลักการกระจายของข้อมูลไปยังทุกโฮสต์ที่อยู่ในเน็ตเวิร์ค และอาศัยโฮสต์แต่ละตัวทำหน้าที่จำแนกการสื่อสารของตัวเอง นั่นหมายความว่าทุกแพ็คเกจที่ใช้สื่อสารกันได้นั้นได้ถูกส่งไปยังทุกโฮสต์ ซึ่งจะได้รับพร้อมกันและเหมือนกันเพียงแต่การที่จะสื่อสารกันได้อย่างถูกต้องนั้น โฮสต์แต่ละตัวต้องมีกระบวนการที่ทำสามารถรู้ได้ว่าข้อมูลแพ็กเกจใดเป็นของตนเองและข้อมูลแพ็กเกจใดไม่ใช่ของตน ทุกๆๆแพ็กเกจที่กระจายลงบนเน็ตเวิร์คนั้นจะมีหมายเลขระบุชัดเจนคือ MAC Address หรือ Ethernet Address ซึ่งจะบอกว่าแพ็กเกจมาจากฮาร์ดแวร์ใดในเน็ตเวิร์ค ให้ระบุได้ว่าแพ็กเกจนั้นส่งมาจากโฮสต์ใด และต้องการส่งให้โฮสต์ใด

MAC Address จะเป็นหมายเลขเฉพาะตามฮาร์ดแวร์ทุกชนิดที่ใช้ในการสื่อสารโปรโตคอลอีเธอร์เนต และในทางทฤษฎีด้วยแล้วฮาร์ดแวร์ทุกชนิดจะไม่มี MAC Address ที่ซ้ำกันเลย โดยทั่วไป MAC Address จะถูกกำหนดตายตัวอยู่ในROM ของฮาร์ดแวร์และไม่สามารถเปลี่ยนแปลงได้โดยซอฟแวร์ โดยการใช้งานของฮาร์ดแวร์จะต้องควบคู่ไปกับไดร์เวอร์ของฮาร์ดแวร์นั้นๆ โดยปกติแล้วไดร์เวอร์จะถูกกำหนดให้ปฏิบัติตามโปรโตคอลอย่างเข้มงวดคือ

- ให้รับข้อมูลที่มี MAC Address เป็นของตนเองเท่านั้น (ห้ามอ่านข้อมูลคนอื่น)
- ให้ส่งข้อมูลที่มี MAC Address เป็นของตนเองเท่านั้น (ห้ามปลอมข้อมูลคนอื่น)

ปัจจัยอีกประการคือสิ่งที่ทำให้สนิฟเฟอร์สามารถดักการอ่านข้อมูลของผู้อื่นได้คือการใช้สื่อกลางของเนตเวิร์กร่วมกัน ไม่ว่าโทโปโลยี 10BaseT, 10Base5, 100BaseTx ล้วนแต่ใช้สื่อกลาง

ในการรับส่งข้อมูลร่วมกันเมื่อประกอบเข้ากับลักษณะของโปรโตคอลแล้วทำให้สนิฟเฟอร์ได้ใช้ข้อบกพร่องในส่วนนี้ดักข้อมูลของโฮสต์ที่ต่ออยู่บนสื่อกลางเดียวกันได้ในทันที การที่จะดักอ่านข้อมูลได้นั้นต้องเกิดขึ้นจากโฮสต์ที่ใช้สื่อกลางอย่างใดอย่างหนึ่ง เช่น สายเคเบิล หรือ ฮับเดียวกัน



รูปที่ 4.1 จำลองการทำงานของสนิฟเฟอร์

ในรูปที่ 4.1 ในเน็ตเวิร์คมีโฮสต์อยู่ 4 ตัวต่อรวมกันเป็นเนตเวิร์กโดยใช้โฮสต์ร่วมกัน และมีโฮสต์อีกตัวหนึ่ง ซึ่งรัน โปรแกรมที่ทำหน้าที่เป็นสนิฟเฟอร์เพื่อดักการอ่านข้อมูลเสียบต่อร่วมอยู่ในฮับนั้นเช่นกัน จากภาพจะแสดงให้เห็นการกระจายของข้อมูลที่ Host A ต้องการส่งให้ Host B ซึ่งแท้จริงเมื่อ Host A ส่งข้อมูลให้ Host B นั้นข้อมูลได้ถูกกระจายไปทุกๆโฮสต์ที่ต่ออยู่บนฮับรวมถึงสนิฟเฟอร์ด้วย โดยเมื่อมีข้อมูลไปปรากฏที่โฮสต์ต่างๆ จะมีปฏิกิริยาดังนี้

- Host B เมื่อได้รับข้อมูลก็จะตรวจสอบ MAC Address ถ้าพบว่าเป็นข้อมูลที่ส่งมาหาตนเองก็จะทำการอ่านและส่งต่อให้ระบบปฏิบัติการประมวลผล

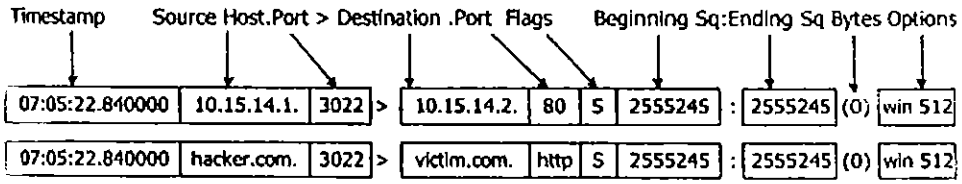
- Host C และ Host D โฮสต์ทั้งสองซึ่งมีเน็ตเวิร์คอะแดปเตอร์อยู่ในโหมดปกติ เมื่อได้ทำการตรวจสอบแล้วเห็นว่าไม่ใช่ข้อมูลของตนเองก็จะไม่สนใจที่จะนำข้อมูลเหล่านั้นไปประมวลผล

- Sniffer สำหรับสนิฟเฟอร์นั้นมีเน็ตเวิร์คอะแดปเตอร์ที่ทำงานอยู่ในโพรมิสคูอัสโหมดนั่นคือจะไม่สนใจว่าข้อมูลที่เข้ามานั้นเป็นของใคร สนิฟเฟอร์จะรับข้อมูลทั้งหมดแล้วนำไปประมวลผลเพื่อหาข้อมูลที่มีประโยชน์ต่อไป

จากเน็ตเวิร์คในภาพนั้น หากสนิฟเฟอร์ถูกติดตั้งในตำแหน่งดังกล่าวแล้ว ก็จะทำให้ข้อมูลทั้งหมดที่โฮสต์ทั้ง 4 สื่อสารกันโดยผ่านฮับที่ใช้ร่วมกัน ไม่ว่าข้อมูลใดจะปรากฏที่ฮับก็สามารถถูกดักอ่านได้โดยสนิฟเฟอร์ทันที โดยที่โฮสต์ทั้ง 4 ไม่รู้ระแคะระคายใดๆเลยว่าข้อมูลที่ส่งไปจะถูกดักอ่าน

4.2 วิธีอ่านแพ็กเกจ

4.2.1 โพรโตคอลทีซีพี/ไอพี



ข้อมูลที่ปรากฏจะประกอบด้วยฟิลด์ดังนี้

Timestamp : แสดงเวลาที่ได้รับเป็นหน่วยชั่วโมง : นาที : วินาที.เศษของวินาที

Source Host : แสดง IP Address ต้นทางสามารถแสดงได้ 2 แบบ คือ IP Address โดยตรง เช่นในตัวอย่างคือ 10.15.14.1 หรือ หากสามารถแปลง IP เป็นชื่อโฮสต์ได้ก็จะแสดงชื่อโฮสต์ เช่น hacker.com เป็นต้น

Port : แสดงหมายเลขของ TCP พอร์ตต้น ทางว่าเป็นพอร์ตหมายเลขใด สามารถแสดงได้ 2 แบบ คือ หมายเลขพอร์ตที่ไม่ใช้บริการของมาตรฐาน และชื่อของบริการหากหมายเลขพอร์ตนั้นเป็นพอร์ตบริการมาตรฐาน เช่น HTTP, SMTP, ECHO, CHARGEN

Destination Host Port เช่นเดียวกับ Source Host , Port เปลี่ยนจากต้นทางเป็นปลายทาง

Flags : แสดง TCP Flag ที่มาพร้อมกับแพ็กเกจนี้ ซึ่งสามารถแสดงค่าได้ตาม TCP Flag ที่มีอยู่ S = SYN , F = FIN P = Push U = Urgent R = Reset

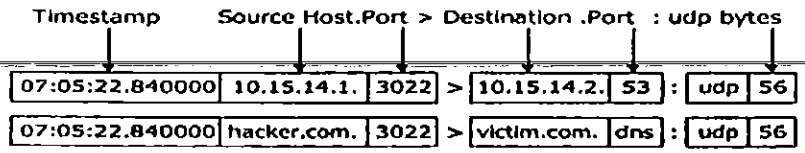
Begining Sequence Number : หมายเลข ISN ซึ่ง TCPo นั้นใช้ในการควบคุมการสื่อสารทั้งในระหว่าง 3 –ways Handshake และเมื่อสามารถเชื่อมต่อได้แล้วและเริ่มส่งข้อมูล

Ending Sequence Number : หมายเลข ISN บวกกับขนาดของข้อมูลที่ส่ง เพื่อเป็นการบอกได้ว่าแพ็กเกจที่จะ ACK กลับมาจะต้องใช้หมายเลขนี้

Bytes : ขนาดของข้อมูลที่ส่งมาพร้อมแพ็กเกจนี้ ในระหว่าง 3 –ways Handshake ขนาดของข้อมูลจะเป็น 0 เสมอจนกว่าจะเริ่มการส่ง

Option : เป็นค่า TCP Option ที่โฮสต์ต้นทางต้องการบอกโฮสต์ปลายทาง จากตัวอย่างจะเป็นการแสดงเมื่อ TCP Option ได้กำหนดค่าของ windows size เท่ากับ 512 ไบต์

4.2.2 โพรโตคอล ยูดีพี



ข้อมูลที่ปรากฏจะประกอบด้วยฟิลด์ดังนี้

Timestamp : แสดงเวลาที่ได้รับเป็นหน่วยชั่วโมง : นาที : วินาที.เศษของวินาที

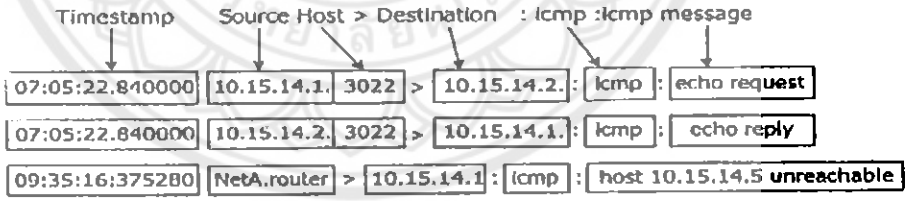
Source Host : แสดง IP Address ต้นทางสามารถแสดงได้ 2 แบบ คือ IP Address โดยตรง เช่น ในตัวอย่างคือ 10.15.14.1 หรือ หากสามารถแปลงIP เป็นชื่อโฮสต์ได้ ก็จะแสดงชื่อโฮสต์เช่น hacker.com เป็นต้น

Port : แสดงหมายเลขของ TCP พอร์ตต้นทางว่าเป็นพอร์ตหมายเลขใดจะสามารถแสดงได้ 2 แบบ คือหมายเลขพอร์ตที่ไม่ใช้บริการของมาตรฐาน และชื่อของบริการหากหมายเลขพอร์ตนั้นเป็นพอร์ตบริการมาตรฐานตัวอย่าง เช่น HTTP,SMTP,ECHO,CHARGEN

Destination Host Port : เช่นเดียวกับ Source Host ,Port เปลี่ยนจากต้นทางเป็นปลายทาง

Bytes : ขนาดของข้อมูล

4.2.3 โพรโตคอล ไอซีเอมพี



ข้อมูลที่ปรากฏจะประกอบด้วยฟิลด์ดังนี้

Timestamp : แสดงเวลาที่ได้รับเป็นหน่วยชั่วโมง : นาที : วินาที.เศษของวินาที

Source Host : แสดง IP Address ต้นทางสามารถแสดงได้ 2 แบบ คือ IP Address โดยตรง เช่นในตัวอย่างคือ 10.15.14.1 หรือ หากสามารถแปลงIP เป็นชื่อโฮสต์ได้ก็จะแสดงชื่อโฮสต์ เช่น hacker.com เป็นต้น แต่สำหรับ ICMP ในบางกรณีแพ็กเกจอาจจะถูกกำเนิดมาจากเราเตอร์ได้ ทำให้ข้อมูลในฟิลด์นี้ปรากฏว่าเป็นเราเตอร์แทน

Destination Host Port : เช่นเดียวกับ Source Host , Port เปลี่ยนจากต้นทางเป็นปลายทาง

Icmp message : หมายถึง message ที่ส่งมากับ icmp แพ็กเกจนี้ซึ่งโปรแกรม TCPDUMP ได้ทำการแปล มาจากตารางของ icmp โดยอัตโนมัติโดยจะอยู่ในรูปรหัส และนำมาเทียบกับตารางก่อนจึงจะได้ Message

4.3 Stimulus & Response

การกระตุ้นและการตอบรับเป็นกลไกส่วนหนึ่งที่สำคัญ ในการเจาะเข้าไปยังระบบต่างๆ โดยทั่วไปแล้วผู้ใช้จะไม่ทราบเลยว่า ตลอดเวลาที่คอมพิวเตอร์ของเราต่ออยู่กับเน็ตเวิร์คนั้นเครื่องคอมพิวเตอร์ของเราได้ทำหน้าที่ทั้งกระตุ้นและรับอยู่ตลอดเวลา ซึ่งการกระตุ้นและตอบรับโดยตัวมันเองนั้นมิได้ก่อให้เกิดความเสียหายแต่อย่างใด ในทางกลับกันการกระทำทั้ง 2 อย่างมีความสำคัญอย่างยิ่งในการควบคุมการรับส่งข้อมูลระหว่างกัน เช่นกรณีที่เราจะทำการส่งผ่านข้อมูลผ่าน TCP ในขั้นตอนแรกสุดสิ่งที่เราจะกระทำคือ Connection Establishment เสียก่อน ตามที่กำหนดอยู่ในโปรโตคอลของ TCP ซึ่งขั้นตอนโดยละเอียดดังนี้

- เครื่องคอมพิวเตอร์ A ส่ง SYN, ISN ไปยังเครื่องคอมพิวเตอร์ B เพื่อเป็นการกระตุ้นให้รู้ว่าจะทำการขอสื่อสารด้วย

- เครื่องคอมพิวเตอร์ B เมื่อได้รับสัญญาณ SYN จากเครื่องคอมพิวเตอร์ A ก็จะตอบรับตามข้อตกลงในโปรโตคอล TCP ก็คือส่ง SYN + ACK พร้อมระบุ Sequence ตามข้อกำหนด

จะเห็นได้ว่าการกระตุ้นและการตอบรับเป็นเรื่องปกติทั่วๆไป ที่การสื่อสารข้อมูลจำเป็นต้องมีเพื่อให้การรับ-ส่งข้อมูลระหว่าง 2 ฝั่ง เป็นไปอย่างถูกต้องพร้อมเพียงและสอดคล้องกันรวมถึงวิธีการก็จะถูกกำหนดไปตามแต่ละโปรโตคอลไป โดยส่วนใหญ่จะเป็นข้อกำหนดในโปรโตคอลระดับกลางขึ้นไป ซึ่งอาศัยการตอบโต้ของผู้รับและผู้ส่งโปรโตคอล

4.3.1 ข้อบกพร่องของทีซีพี/ไอพี

ข้อบกพร่องหลักที่แฮกเกอร์หาประโยชน์จากการกระตุ้นและการตอบรับ 3 ข้อคือ

ขาดกลไกด้านความปลอดภัย

สำหรับ TCP/IP แล้วหากมีการกระตุ้นที่ถูกต้องตามโปรโตคอลแล้วจะต้องตอบรับเสมอ โดนโปรโตคอลจะกำหนดไว้ชัดเจนว่า หากมีการกระตุ้นที่ถูกต้องตามโปรโตคอลจะต้องยอมรับ โฮสต์ผู้รับจะทำหน้าที่ตรวจสอบเพียงถูกต้องตามโปรโตคอลหรือไม่เท่านั้น ไม่ได้กำหนดเรื่องอื่น เช่น ปริมาณมากเกินไปหรือไม่ , ต่อเนื่องหรือไม่, ไม่มีกลไกการตรวจสอบผู้ถาม และการจัดการกับคำถามที่ผิดปกติ เช่นในกรณีปกติ หากโฮสต์ของเราได้รับ ICMP Echo request โฮสต์ของเราไม่สามารถตรวจสอบได้ว่าผู้ส่งมีสิทธิ์ได้รับคำตอบหรือไม่หรือผู้ส่งถามซ้ำๆกันมากเกินไปแล้ว สิ่งที่โฮสต์จะทำคือจะต้องตอบกลับไปด้วย ICMP Echo reply เท่านั้น และจะต้องตอบกลับทุกกรณี ทั้งนี้

เพราะกลไกการรักษาความปลอดภัยมิได้อยู่ในโปรโตคอลด้วย ซึ่งโดยช่องว่างนี้เองที่ถูกนำมาใช้ประโยชน์โดยการเข้ามาขโมยข้อมูล ไม่ว่าจะเป็นการสแกนพอร์ตสแกนเวิร์ก การ DoS ด้วย SYN Flood, Ping Flood เป็นต้น ซึ่งสามารถกระทำได้โดยที่โฮสต์เทบไม่สามารถป้องกันตัวเองได้เลย

การตอบรับเป็นสิ่งที่คาดหมายได้

การตอบรับจะต้องเป็นสิ่งที่คาดหมายได้ เพราะว่าการตอบรับใดๆนั้นจะต้องเป็นไปตามข้อกำหนดในโปรโตคอลซึ่งเผยแพร่แก่สาธารณชนอยู่แล้ว หากการตอบสนองคาดหมายไม่ได้ การสื่อสารสองทางจะไม่เกิดขึ้นเพราะสื่อสารกันไม่รู้เรื่อง แฮกเกอร์จึงอาศัยการวิเคราะห์การตอบรับของเป้าหมายในสถานการณ์ต่างๆมาใช้ให้เป็นประโยชน์

ในตารางที่ 4.1 จะแสดงตัวอย่างการวิเคราะห์จากการกระตุ้นและคาดการณ์ผลตอบรับในกรณีต่างๆที่จะได้จากเป้าหมาย ซึ่งสามารถนำมาขยายผลให้เกิดประโยชน์มากขึ้นได้ จากตัวอย่างจะเห็นว่าการส่ง TCP ไปยังเป้าหมายเพียงแพ็กเกจเดียวเท่านั้นก็สามารถทราบข้อมูลของเป้าหมายได้พอสมควร โดยที่เป้าหมายสามารถจะหลีกเลี่ยงการให้ข้อมูลไปได้อย่างใด

ตารางที่ 4.1 ตัวอย่างการวิเคราะห์เป้าหมาย

วิธีการกระตุ้น	การตอบรับที่คาดว่าจะได้รับ	ผลการณ์วิเคราะห์
ส่ง TCP SYN พอร์ต 80	TCP SYN,ACK	1.โฮสต์เป้าหมายยังทำงานอยู่ 2.โฮสต์เป้าหมายมีแอพพลิเคชั่นให้บริการโดยใช้พอร์ต 80 เนื่องจากมีการตอบรับจากพอร์ต 80 3.สันนิษฐานว่าโฮสต์เป้าหมายทำงานเป็นเว็บเซิร์ฟเวอร์ (เนื่องจากเว็บเซิร์ฟเวอร์ใช้พอร์ต 80 ในการให้บริการ)
	RST	1.โฮสต์เป้าหมายทำงานอยู่ 2.โฮสต์เป้าหมายไม่มีแอพพลิเคชั่นใดทำงานอยู่ที่พอร์ต 80
	ICMP Host Unreachable	1.โฮสต์เป้าหมายไม่ทำงาน ปิดเครื่องอยู่

การตอบรับกำหนดไว้ไม่ครอบคลุมทุกเงื่อนไข

ด้วยเป้าหมายของโปรโตคอลในเบื้องต้นคือความถูกต้องความมีเสถียรภาพ และความมีประสิทธิภาพของการสื่อสาร ดังนั้นเงื่อนไขข้อกำหนดต่างๆที่ระบุไว้นั้นก็เพื่อให้เป็นไปเพื่อรับใช้วัตถุประสงค์ดังกล่าวจุดสำคัญอยู่ที่หากมีการสื่อสารที่ถูกต้องตามโปรโตคอล คอมพิวเตอร์ทั้ง 2 ฝั่งจะต้องอำนวยความสะดวกให้ทำการสื่อสารด้วยความถูกต้อง มีเสถียรภาพและประสิทธิภาพสูงสุด แต่โปรโตคอลกลับละเลยในส่วนนี้ ซึ่งแน่นอนว่าการส่งสัญญาณหรือข้อมูลใดๆที่ผิดข้อกำหนดโปรโตคอลนั้นมิสามารถกระทำได้ในการทำงานปกติ เช่น หากจะส่งข้อมูลผ่าน TCP อาจจะเรียก API (Application Program Interface) หรือ Socket มาทำงานที่เหลือ API เหล่านี้ก็ไปจัดการข้อมูลในระดับล่างเองให้ถูกต้องตามโปรโตคอล แต่อย่าลืมว่า API เหล่านั้นก็เป็นโปรแกรมชนิดหนึ่งที่ทำหน้าที่จัดระเบียบข้อมูลแล้วส่งไปยังอุปกรณ์ในเลเยอร์ล่างเท่านั้น หากแฮกเกอร์มีโปรแกรมที่ทำหน้าที่ดังกล่าว ก็สามารถส่งข้อมูลไปยังอุปกรณ์ต่างๆได้โดยตรงเช่นกัน

4.3.2 การนำคุณสมบัติของ Stimulus & Response ไปใช้งาน

การสำรวจเป้าหมาย

การสำรวจเป้าหมายหรือที่เรียกว่าการสแกน เช่นพอร์ตสแกน เน็ตเวิร์คสแกน หรือ การส่งสัญญาณไปกระตุ้นเป้าหมายในลักษณะต่างๆเช่น หากเป็นการพอร์ตสแกนก็ทำโดยการส่งสัญญาณ TCP SYN ไปยังทุกพอร์ตของเป้าหมาย แล้วการตอบรับมาของแต่ละพอร์ตจะทำให้สันนิษฐานได้ว่าโฮสต์เป้าหมาย เปิดให้บริการอะไรไว้บ้าง หรือสำหรับเน็ตเวิร์คสแกนก็โดยการส่ง ICMP Echo Request ไปยังทุก IP ในเน็ตเวิร์คนั้นและคอยการตอบ ICMP Echo Reply กลับมาจากแต่ละ IP จะทำให้ทราบได้ว่าโฮสต์ใดที่เปิดใช้งานอยู่ในเน็ตเวิร์คบ้าง และการขาดกลไกด้านความปลอดภัยที่ดีในระดับโปรโตคอล เป็นการเปิดโอกาสให้ผู้อื่นได้ทำการสำรวจเป้าหมายต่างๆอย่างเสรี ทำให้ละเมิดสิทธิของผู้อื่นในเน็ตเวิร์คด้วยการสแกนโดยไม่ได้รับอนุญาตเกิดขึ้นในหลายระดับ สิ่งที่ควรตระหนักก็คือ การที่แฮกเกอร์สามารถเจาะเข้าระบบเข้าไปยังระบบของเหยื่อได้นั้นจะต้องอาศัยทักษะความชำนาญและเครื่องมือพอสมควรเพื่อค้นหาจุดอ่อนรูรั่วของเป้าหมาย และที่สำคัญที่สุดคือจะต้องมีข้อมูลของเหยื่อมากพอสมควรการเจาะระบบนั้นๆจึงดำเนินการได้สำเร็จแฮกเกอร์น้อยรายนักที่จะสามารถเจาะผ่านระบบรักษาความปลอดภัยของเหยื่อได้ โดยไม่มีข้อมูลของเหยื่อเลย ดังนั้นกระบวนการให้ได้มาซึ่งข้อมูลจะเป็นกระบวนการต้นๆของความพยายามในการเจาะระบบก็คือการสำรวจเป้าหมายด้วยวิธีต่างๆนั่นเอง เป็นการป้องกันมิให้โฮสต์ของเราถูกสแกนได้โดยง่าย ถึงแม้ว่าไม่ใช่เป็นการป้องกันการเจาะระบบเข้าโดยตรง แต่ก็ถือว่าเป็นการป้องกันที่มีประสิทธิภาพซึ่งจะช่วยให้ระดับความยากในการเจาะระบบสูงขึ้น และลดความเสี่ยงให้น้อยลงได้

การก่อกวนการทำงานของเป้าหมาย

การก่อกวนการทำงานของเป้าหมาย อาจเป็นการก่อกวนโฮสต์หรืออุปกรณ์ในเน็ตเวิร์คโดยส่วนใหญ่จะอาศัยความบกพร่องของกระบวนการนี้ การขาดกลไกด้านความปลอดภัยเป็นปัจจัยที่สำคัญที่นอกจากจะทำให้โฮสต์ต้องตอบรับการกระตุ้นแบบต่างๆ และเปิดเผยข้อมูลให้กับแฮกเกอร์เมื่อถูกสแกนแล้ว ที่แย่กว่านั้นคือโฮสต์ไม่มีทางเลือกที่จะไม่ตอบรับการตอบรับของโฮสต์ใดๆเป็นสิ่งที่คาดหมายได้ โดยตราบใดที่โฮสต์ที่โฮสต์ยังคงทำงานได้ก็ต้องตอบรับทุกครั้งที่มีการกระตุ้นไม่ว่าการกระตุ้นนั้นจะเพื่อวัตถุประสงค์ใด และมาจากไหน เนื่องจากการตอบรับเป็นกระบวนการหนึ่งของการสื่อสารข้อมูล ดังนั้นเมื่อโฮสต์ได้รับการกระตุ้นไม่ว่าจะเป็นแบบใด การส่งสัญญาณตอบรับกลับไปได้นั้นในเบื้องหลังแล้วจะต้องใช้ทรัพยากรที่เกี่ยวข้องในระบบนี้พอสมควร ถึงแม้ว่าอาจจะใช้ทรัพยากรไปในจำนวนไม่มากเมื่อเทียบกับทรัพยากรที่มีอยู่ และด้วยปริมาณทรัพยากรที่ถูกใช้ไปเพียงเล็กน้อยทำให้ในสภาวะปกติแล้วผู้ใช้ไม่ทราบ และไม่ได้รับผลกระทบจากกระบวนการนี้เลย แต่หากต้องทำการตอบรับการกระตุ้นในปริมาณมหาศาลแล้วทรัพยากรที่มีอยู่ก็อาจจะใช้ให้หมดไปได้เช่นกัน เทคนิคการการก่อกวนโฮสต์เป้าหมายโดยการส่งแพ็กเกจปริมาณมากๆไปยังเป้าหมายนั้นเรียกว่า Flooding คือจะคล้ายกับการทำให้เน็ตเวิร์คนั้นท่วมท้นไปด้วยแพ็กเกจที่ไม่มีประโยชน์โดยทั่วไปจะส่งผลกระทบต่อเนตเวิร์กอันดับแรก แต่ก็มีไม่น้อยที่ส่งผลกระทบต่อโฮสต์ด้วยเช่นกันที่รู้จักกันโดยทั่วไปคือ

SYN Flooding คือการส่ง TCP SYN แพ็กเก็ตจำนวนมากไปยังเป้าหมาย

Ping Flooding คือการส่ง ICMP Echo Request จำนวนมากไปยังเป้าหมาย

เนื่องจากการกระตุ้น และการตอบรับเป็นส่วนหนึ่งของการสื่อสารข้อมูลจึงเป็นเรื่องยากที่จะป้องกันตนเองมิให้ถูกโจมตี เพราะว่าการโจมตีและการสื่อสารปกติบางครั้งดูผิวเผินมิแตกต่างกัน ดังนั้นสิ่งที่ดีที่สุดที่จะทำให้เราป้องกันตนเองได้คือ จะต้องมีความรู้และเข้าใจในการตอบรับการกระตุ้นตามปกติเสียก่อน อาจจะทำให้เราทราบได้ว่าส่วนใดบ้างที่สำคัญ ส่วนใดบ้างที่จำเป็นต้องใช้ จึงค่อยป้องกันตามความเหมาะสม เพราะการป้องกันที่มากเกินไปนั้นอาจจะทำให้ระบบไม่สามารถสื่อสารหรือให้บริการได้ตามปกติ

4.3.3 Stimulus & Response ของแต่ละโพรโตคอล

TCP stimulus – Responce

การกระตุ้นสำหรับ TCP นั้นมีข้อมูลสำคัญที่จะต้องระบุในการกระตุ้นคือ IP Address และหมายเลขพอร์ตปลายทาง ในสภาวะปกติเงื่อนไขในการตอบรับในแต่ละสถานการณ์จะแตกต่างกันออกไป การกระตุ้นสามารถทำได้โดยง่ายเพียงแต่ใช้คำสั่ง Telnet ไปยัง IPAdress กับหมายเลขพอร์ตที่ต้องการและสังเกตผลการตอบรับตามแต่ละเงื่อนไขดังนี้

กรณีที่1 โฮสต์เป้าหมายเปิดให้บริการบนพอร์ตที่ระบุ

ในกรณีนี้เราสามารถคาดหมายการตอบรับจากเซิร์ฟเวอร์ได้ เพราะกระบวนในการสร้าง Connection ของ TCP จะเสร็จสิ้นได้อย่างสมบูรณ์ โดยข้อมูลการสื่อสารทั้งสองฝั่งจะเป็นดังนี้

Stimuli	client.com : 25001> server.com.telnet S 2774653900 : 2774653900 (0) win 8760 <MSS1460> (DF)
Response	server.com.telnet > client.com.25001 S 2500700000: 2500700000 (0) ack 2774653901 win 1024 <MSS1460>

กรณีที่ 2 โฮสต์เป้าหมายไม่เปิดให้บริการบนพอร์ตที่ระบุ

กรณีนี้คือ Host ปลายทางทำงานอยู่แต่ไม่เปิดให้บริการบนพอร์ตที่ถูกสอบถามมาปฏิกิริยาต่อการSYN เข้ามายังพอร์ตที่ไม่เปิดให้บริการคือ Host จะทำการตอบกลับไปยังผู้ที่ SYN มา ด้วยการส่ง Reset Flag กลับเพื่อไปยุติการติดต่อทันที เพื่อปฏิเสธผู้ที่ติดต่อเข้ามามิให้มีการติดต่อกับพอร์ตนี้

Stimuli	client.com : 25001> server.com.telnet : S 2759957870 : 2759957870 (0) win 8760 <MSS1460> (DF)
Response	server.com.telnet > client.com.25001 R 0:0 (0) Ack 2759957871 WIN 0

กรณีที่ 3 โฮสต์ปลายทางไม่มีอยู่บนเน็ตเวิร์ค

กรณีนี้คือ Host ปลายทางไม่มีอยู่บนเน็ตเวิร์คอาจจะเป็นเพราะปิดเครื่องอยู่ไม่ได้ต่อเข้ากับเน็ตเวิร์ค โฮสต์ไม่สามารถสื่อสารกับเน็ตเวิร์คได้ด้วย TCP โดยสัญญาณที่ได้รับตอบกลับนี้จะไม่ใช่ TCP แต่จะเป็น ICMP Unreadable ซึ่งเป็นกลไกการควบคุมการส่งที่อยู่ในเลเยอร์ของ IP ที่ต่ำลงไป และผู้ส่ง ICMP กลับมาก็เป็นเราเตอร์ซึ่งทำหน้าที่ตามที่กำหนดไว้ใน IP และเมื่อ routerไม่สามารถส่ง IP Datagram ไปยัง server.com ที่ต่อกลับตนเองได้

Stimuli	client.com : 25001> server.com.telnet S 2759957870 : 2759957870 (0) win 8760 <MSS 1460>
Response	router.com > client.com : ICMP : host server.com: uncreachable

กรณีที่ 4 โฮสต์ปลายทางถูกบล็อกไว้โดยเราเตอร์

กรณีนี้คือโฮสต์มีอยู่จริงและต่ออยู่กับเน็ตเวิร์ค แต่เราเตอร์อาจจะกำหนดAccess control List ว่าให้โฮสต์ใดสามารถให้บริการพอร์ตใดได้บ้าง รวมทั้งการควบคุมให้มีการสื่อสารภายนอกสามารถทำการผ่านเราเตอร์ไปยังปลายทางผ่านพอร์ตใดได้บ้าง เพื่อป้องกันการติดต่อผ่านพอร์ตอื่นที่ System Admin ไม่ทราบและไม่ต้องการ

ICMP Stimulus – Response

ICMP เป็นโปรโตคอลที่ไม่ใช้หมายเลขพอร์ตในการสื่อสารจะอาศัยเพียง IP Address ของต้นทางและปลายทางเท่านั้น เนื่องจากการบริการที่ใช้ ICMP นั้นมีบริการจำกัด วัตถุประสงค์หลักคือนำข่าวสารไปยังปลายทางเท่านั้น อย่างไรก็ตามมี code บางอย่างใน ICMP ที่ใช้เพื่อการสอบถามข้อมูลผู้รับ ICMP code นั้นก็มีหน้าที่ต้องตอบกลับมาและให้ข้อมูลตามที่ได้รับการร้องขอ ซึ่งการสอบถามข้อมูลเหล่านี้ก็สามารถใช้ในการกระตุ้นและนำผลที่ได้รับมาใช้ให้เกิดประโยชน์ได้

กรณีที่ 1 ICMP echo request-reply

Stimuli client.com > server.com : icmp : echo request

Response server.com > client.cm : icmp : echo reply

ที่ปรากฏข้างต้นเป็นการกระตุ้นและการตอบรับตามปกติของ ICMP echo request ส่วนใหญ่การใช้ประโยชน์จาก ICMP echo request – reply คือการใช้เพื่อตรวจสอบสถานะของโฮสต์ปลายทางว่าต่ออยู่หรือไม่ หรือที่รู้จักในคำสั่ง ping โดยการต่อกับเน็ตเวิร์คนั้นตามปกติเราสามารถใช้ ping เป็นตัวยืนยันการคงอยู่ของโฮสต์ปลายทางได้ แต่ในกรณีที่มีอุปกรณ์อื่นทำงานอยู่ด้วยเช่น ไฟร์วอล หรือเราเตอร์ที่มีการควบคุมการเข้าออกของICMP

กรณีที่ 2 ICMP Time Exceed

ICMP Time Exceed จะถูกส่งกลับมายังปลายทางเมื่อค่าของ TTL ใน IP Header นั้นลดลงเป็น 0 และเราเตอร์จะไม่ทำการส่งแพ็กเกจนั้นอีกต่อไป และจะแจ้งกลับไปยังผู้ส่งให้ทราบว่าเวลาหมดแล้วและไม่ถูกส่งต่อ มีการนำคุณสมบัติในข้อนี้ของICMPไปใช้งานเพื่อการตรวจสอบเส้นทางเดินของข้อมูลจากต้นทางไปยังปลายทางเพื่อหาว่าจะต้องผ่านเราเตอร์ที่จุดใดบ้าง

4.4 ความสำคัญของพอร์ต

พอร์ตเป็นช่องทางการสื่อสารของ TCP/IP กับแอพพลิเคชั่น ไม่มีพอร์ตก็ไม่มีช่องทางในการสื่อสารกับผู้อื่น เนื่องจาก TCP/IP เป็นโปรโตคอลที่ทำงานอยู่ในเลเยอร์ที่ค่อนข้างสูง ดังนั้นพอร์ตของ TCP/IP จึงเป็นลักษณะลอจิคัลคือไม่ได้อาศัยองค์ประกอบทางกายภาพใดๆเป็นเพียงข้อมูลขนาด 16 บิต ซึ่งอยู่ในไบต์ที่ 0-4 ของ TCP Header และ UDP Header เท่านั้น ต่างจากพอร์ตทั่วไปซึ่งเป็นพอร์ตในระดับกายภาพที่เรานั้นรู้จัก เช่น พอร์ตอนุกรม, พอร์ตขนาน, พอร์ตอีเธอร์เนต ซึ่งพอร์ตประเภทนี้จะต้องอาศัยองค์ประกอบทางกายภาพ ในการเพิ่มหรือลดจะต้องมีการเพิ่มหรือลดลงจริงๆจึงกระทำได้ และแน่นอนหากว่าเป็นพอร์ตที่ต้องอาศัยองค์ประกอบทางกายภาพจริงๆแล้วก็ย่อมจะมีขีดจำกัดทางกายภาพด้วยเช่นกัน

4.4.1 พอร์ตของทีซีพี/ไอพี

พอร์ตของ TCP และ UDP จะมีได้ทั้งสิ้นอย่างละ 65534 พอร์ต ดังนั้นหากเครื่องของเราใช้โปรโตคอลนี้ก็จะมีช่องทางการสื่อสารข้อมูลได้ถึง 131,068 พอร์ต ซึ่งโดยทั่วไปพอร์ตจะมีสถานะคือเปิด กับ ปิด พอร์ตเปิดหมายถึงการมีแอพลิเคชั่นใดๆใช้งานพอร์ตนั้นอยู่และเปิดรับการสื่อสารที่พอร์ตดังกล่าว หากมีการพยายามติดต่อมายังที่พอร์ตที่เปิดไว้ก็จะมีการตอบรับและดำเนินการสื่อสารกันต่อไป พอร์ตที่ปิดหมายถึงไม่มีแอพพลิเคชั่นใดใช้งานอยู่ หากมีความพยายามติดต่อไปยังพอร์ตที่ปิดอยู่ก็จะถูกปฏิเสธทันทีตามที่กำหนดในโปรโตคอล ไม่ว่าจะเป็น TCP หรือ UDP การที่จะเริ่มการสื่อสารใดๆนั้น จะต้องมีฝ่ายใดฝ่ายหนึ่งเปิดพอร์ตรอไว้ก่อน ที่เรียกว่าเป็นเซิร์ฟเวอร์ (server) และพอร์ตที่เปิดเรียกว่าเซิร์ฟเวอร์พอร์ต (server port) และอีกฝ่ายจะต้องส่งสัญญาณติดต่อด้วยเรียกว่าไคล์เอนต์ (client) พร้อมทั้งเปิดพอร์ตของตนเองไว้เพื่อรอการติดต่อกลับจากเซิร์ฟเวอร์เรียกว่าไคล์เอนต์พอร์ต (client port) หากไม่มีการเปิดเซิร์ฟเวอร์พอร์ตแล้วไว้การสื่อสารใดๆของ TCP/IP ก็ไม่สามารถเริ่มต้นได้

4.4.2 การเปิดพอร์ต

เราสามารถกำหนดได้เพียงส่วนที่อยู่ในระดับไอพี คือ หมายเลขไอพี สับเน็ตมาสก์ และเกตเวย์เท่านั้น จะไม่สามารถกำหนดได้ว่าจะเปิดปิดพอร์ตใดบ้าง การที่พอร์ตใดจะเปิดให้บริการเป็นเซิร์ฟเวอร์พอร์ตนั้นจะต้องมีแอปพลิเคชั่นทำงานอยู่บนพอร์ตนั้นเสมอ คือมีโปรแกรมที่จะรับหน้าที่โต้ตอบและจัดการการสื่อสารที่มายังพอร์ตนั้น จึงอาจเปรียบได้ว่าพอร์ตก็คือแอปพลิเคชั่นการที่มีพอร์ตเปิดอยู่ก็หมายถึงการมีแอปพลิเคชั่นทำงานอยู่

นอกจากแอปพลิเคชั่นจะเปิดพอร์ตเพื่อใช้งานแล้ว ระบบปฏิบัติการที่อาศัย ทีซีพี/ไอพี ก็จะต้องเปิดพอร์ตเพื่อใช้ในกิจการของระบบปฏิบัติการด้วย โดยที่ผู้ใช้ไม่รู้ตัว เพราะเป็นการใช้งานภายในของระบบปฏิบัติการและผู้ผลิตคิดว่าผู้ใช้ไม่จำเป็นต้องรู้ จึงทำให้ผู้ใช้อาจถูกบุกรุกจากพอร์ตเหล่านี้ด้วย ซึ่งเมื่อเริ่มใช้แอปพลิเคชั่นมาก เครื่องคอมพิวเตอร์ของเราก็จะเริ่มเปิดพอร์ตมากขึ้น ซึ่งเป็นการเปิดช่องทางให้ผู้อื่นติดต่อเข้ามาได้มากขึ้นตามไปด้วย

4.4.3 การปิดพอร์ต

การปิดพอร์ต คือ การไม่ยอมรับการติดต่อเข้ามายังพอร์ตนั้นๆ เช่นเดียวกับการเปิดพอร์ต เราไม่สามารถปิดพอร์ตนั้นโดยตรงได้ด้วยโฮสต์ทั่วไป หากจะปิดพอร์ต จะต้องหยุดการทำงานของแอปพลิเคชั่นก่อนแล้วพอร์ตจะถูกปิดไปเอง หรือสามารถทำได้โดยผ่านไฟร์วอล เราเตอร์ หรืออุปกรณ์ Layer 4 Switch การปิดพอร์ตไม่ใช่เรื่องยาก ปัญหาที่เกิดขึ้นเนื่องมาจากพอร์ตไม่ได้ปิดด้วยสาเหตุต่างๆ

พอร์ตที่เปิดไว้โดยไม่ได้ตั้งใจ

การเปิดพอร์ตเป็นการเปิดแบบลอจิคัลและมองไม่เห็น ดังนั้นหากเราไม่ทำการตรวจสอบโฮสต์ของเราให้ดี จะไม่ทราบว่ามีพอร์ตใดเปิดอยู่ ส่วนใหญ่เกิดจากแอปพลิเคชั่นอื่นๆ มาเปิดพอร์ตบนโฮสต์เราโดยที่เราไม่เคยติดตั้งเข้าไปด้วยเลย โดยแอปพลิเคชั่นเหล่านี้จะได้มาตั้งแต่ขั้นตอนการติดตั้งระบบปฏิบัติการซึ่งผู้ผลิตคาดว่าผู้ใช้ต้องการใช้แอปพลิเคชั่นเหล่านั้น ผู้ใช้สามารถตรวจสอบว่ามีแอปพลิเคชั่นใดบ้างที่ทำงานอยู่ โดยที่เรานั้นไม่ต้องการให้หยุดการทำงานของแอปพลิเคชั่นเหล่านั้น พอร์ตก็จะถูกปิดไปเอง

พอร์ตของระบบปฏิบัติการ

เป็นพอร์ตที่จำเป็นสำหรับระบบปฏิบัติการนั้นๆ หากไม่เปิดพอร์ตเปล่านี้ ระบบปฏิบัติการก็จะไม่สามารถทำงานได้อย่างสมบูรณ์ เช่น ไมโครซอฟท์ วินโดวส์ เอ็นที จะต้องใช้พอร์ต 135-139 ของ ทีซีพี ในการทำงาน พอร์ตประเภทนี้จะไม่สามารถปิดลงได้ เนื่องจากแอปพลิเคชั่นที่ใช้งานพอร์ตนั้นเป็นส่วนหนึ่งของระบบปฏิบัติการ ข้อเสียอย่างมากก็คือ นอกจากผู้ใช้จะไม่สามารถปิดพอร์ตเหล่านี้ได้ ยังเป็นการบอกผู้บุกรุกอีกด้วยว่าใช้ระบบปฏิบัติการอะไร และทำให้ผู้บุกรุกสามารถโจมตีได้ง่ายขึ้น

พอร์ตที่เปิดแบบสุ่ม

เกิดจากแอปพลิเคชั่นบางประเภทที่มีการใช้งานพอร์ตมากกว่า 1 พอร์ต โดยมีหมายเลขพอร์ตที่คงที่ไว้เป็นหลัก 1 พอร์ต ส่วนพอร์ตที่จะเปิดเป็นการชั่วคราวนี้ ไคลเอนต์และเซิร์ฟเวอร์ จะมีการตกลงกันเพื่อเปลี่ยนไปสื่อสารกันที่พอร์ตนั้นๆ ซึ่งการเปิดพอร์ตประเภทนี้มีปัญหาคือ

- พอร์ตจะปิดลงเมื่อการใช้งานเสร็จสิ้น แต่หากแอปพลิเคชั่นทำงานผิดพลาดหรือหยุดลงกลางคัน พอร์ตก็อาจจะถูกเปิดค้างทิ้งไว้
- การไม่มีหมายเลขพอร์ตแน่นอน ทำให้ควบคุมและตรวจสอบได้ยาก หากพอร์ตที่ใช้บังเอิญตรงกับพอร์ตที่อันตรายซึ่งใช้โดยโปรแกรมประเภทโทรจัน
- หา TD0.0 มีการนำไฟร์วอลมาใช้งาน การกำหนดกฎสำหรับไฟร์วอลจะทำได้ยากเพราะกฎของไฟร์วอลจะตั้งอยู่บนพื้นฐานของการใช้พอร์ตเป็นหลัก

4.4.4 พอร์ตอันตราย

ปัจจุบันมีโปรแกรมจำนวนมากที่ถูกเขียนขึ้นด้วยวัตถุประสงค์มุ่งร้าย โดยใช้เป็นเครื่องมือประกอบการบุกรุกไปยังโฮสต์ต่างๆ โปรแกรมประเภทนี้ได้แก่

ม้าโทรจัน

เป็นโปรแกรมสำเร็จรูปในการเข้ามาแอบล้วงความลับกัน ใช้หลักการที่เรียกว่า Plugin หรือ Attachment รวมเข้ากับโปรแกรมใช้งานจริง ๆ ใส่รวมไปแล้ว คนที่แกล้งก็ทำเป็นส่งไฟล์ให้ คนอื่นหรืออาจส่งไฟล์ไปกับเมล์ เมื่อผู้ได้รับไฟล์นั้นไปสั่ง Run ใช้งานเข้า โปรแกรมก็จะแอบเปิด Port

ให้เครื่องของเราทำงานคล้ายเป็น Server คนที่ต้องการแกล้งหรือล้วงความลับก็สามารถที่จะ เข้ามาใช้งานเครื่องเราได้ทุกอย่าง เช่น ดูไฟล์ความลับเกี่ยวกับ Password ต่าง ๆ ที่เรามีใช้กับ บริการต่าง ๆ

แบ็คออริฟิช

แบ็คออริฟิชผู้ที่ต้องการแกล้งจะนำไปรวมกับไฟล์หรือโปรแกรมอื่น เช่นเดียวกับตัว ม้าโทรจันเมื่อเครื่องได้รับโปรแกรมนี้ แบ็คออริฟิช จะแอบทำงาน เปิดเครื่องคอมพิวเตอร์ให้ติดต่อกันได้ ทางอินเตอร์เน็ต โดยเครื่องที่ติด แบ็คออริฟิช ก็จะคล้าย Server ให้ผู้ที่แกล้งเป็น Client เข้ามาควบคุม เครื่อง Server ซึ่งเป็นเครื่องที่ติด แบ็คออริฟิช อยู่ล้วงความลับต่าง ๆ ได้ ใช้คำสั่งต่าง ๆ บนเครื่องคอมพิวเตอร์ของผู้ติด แบ็คออริฟิช ได้ แบ็คออริฟิชทำงานได้บน Windows 95 & 98 ไม่สามารถทำงานบน Windows NT

เนตบัส

NetBus เป็นโปรแกรมที่ยอมให้คนที่ต่อเชื่อมเข้ามายังเครื่องเรา สามารถเข้าถึงและ ควบคุมเครื่องของเราได้ NetBus มีความสามารถมากกว่า BO ตรงที่ NetBus ทำงานได้ทั้งบน ระบบปฏิบัติการ Windows NT, Windows 95 & Windows 98 ด้วย NetBus จึงถูกใช้ร่วมกับ Back Orifice แต่จะถูกใช้ไปในทางที่ผิดคือ ใช้ในการแกล้งกัน โดยติดต่อและควบคุมได้หลาย รูปแบบ เช่น เปิด-ปิดไดร์ฟซีดี ควบคุมเมาส์ไม่ได้ เช่น เปลี่ยนปุ่มการทำงานสลับปุ่มซ้ายไปขวา หรือจากขวาไปซ้าย

4.4.5 การใช้ข้อมูลของพอร์ตเพื่อการเจาะระบบ

สัญญาณเริ่มต้นของการบุกรุกคือการถูกสแกนพอร์ต การสแกนทุกครั้งนั้นผลลัพธ์ที่ได้คือรายละเอียดที่จะบอกได้ว่าโฮสต์ที่ถูกสแกนนั้นใช้ระบบปฏิบัติการอะไร มีแอพพลิเคชั่นใดทำงานอยู่บ้าง ซึ่งเป็นข้อมูลที่ประโยชน์มากสำหรับแฮกเกอร์ เปรียบเสมือนได้ทราบว่ามีช่องทางใดบ้างที่สามารถเจาะเข้าไปได้

นอกจากจะอาศัยโปรแกรมประเภทม้าโทรจัน หรือแบ็คออริฟิชซึ่งออกแบบมาเพื่อต้อนรับแฮกเกอร์โดยเฉพาะแล้วแอพพลิเคชั่นทั่วๆไปก็เป็นช่องทางที่สามารถใช้เพื่อเป็นทางผ่านได้เช่นกัน การที่แฮกเกอร์สามารถเจาะเข้าสู่ระบบได้นั้นก็โดยอาศัยความบกพร่องของแอพพลิเคชั่น ในการที่แอพพลิเคชั่นถูกเขียนขึ้นมาอย่างรัดคุมนั้นเพียงพอ และเปิดโอกาสให้ผู้ที่รู้ช่องทางเหล่านี้แอบใช้ประโยชน์ในการเล็ดรอดเข้าสู่ระบบ นอกจากข้อมูลว่าแอพพลิเคชั่นใดมีรูรั่วแล้ว วิธีที่จะอาศัยรูรั่วเหล่านั้นเข้ามาในระบบก็เป็นที่เผยแพร่กันทั่วไป บางประเภทอาจจะใช้เทคนิคเพียงเล็กน้อย บางประเภทอาจจะต้องใช้เทคนิคที่ซับซ้อนประกอบกับลำดับที่ถูกต้องจึงจะสามารถเข้าไปได้

ดังนั้นการเจาะระบบใดระบบหนึ่งนั้น บางครั้งเพียงแค่เป็นแฮกเกอร์สมัครเล่นและอาจไม่จำเป็นต้องมีความรู้ทางเทคนิคมากมายนัก อาศัยเพียงขั้นตอนที่เหมาะสมและเลือกเครื่องมือที่ถูกเขียนไว้แล้วเป็นโปรแกรมสำเร็จรูปสำหรับการเจาะระบบที่มีเผยแพร่อยู่ก็สามารถดำเนินการได้

สำเร็จการเจาะระบบโดยส่วนใหญ่ โอกาสสำเร็จนั้นมิได้ขึ้นอยู่กับว่าแฮกเกอร์เก่งสามารถเพียงใด แต่ขึ้นอยู่กับว่าเครื่องมือที่ใช้สัมฤทธิ์ผลหรือไม่

เนื่องจากโปรแกรมที่ใช้เจาะระบบเหล่านี้ เขียนมาจากความรู้เรื่องการอาศัยข้อบกพร่องของแอพพลิเคชั่นใดแอพพลิเคชั่นหนึ่งโดยเฉพาะเจาะจง ดังนั้นการนำโปรแกรมเพื่อเจาะสำหรับแอพพลิเคชั่นหนึ่งไปยังแอพพลิเคชั่นหนึ่งย่อมไม่ได้ผลอย่างแน่นอน เพราะข้อบกพร่องที่แตกต่างกันออกไปนั้นโปรแกรมแต่ละตัวไม่เหมือนกัน แม้กระทั่งแอพพลิเคชั่นตัวเดียวกันแต่ต่างกันคนละเวอร์ชั่นก็มีข้อบกพร่องที่ต่างกันออกไปโปรแกรมแต่ละตัวจึงจะระบุไว้ชัดเจนว่าสามารถเจาะระบบสำหรับแอพพลิเคชั่นใดได้บ้างและเวอร์ชันอะไร ตัวอย่างเช่น IIS (เว็บเซิร์ฟเวอร์ของไมโครซอฟต์) เวอร์ชั่น X.X.X Sendmail XX.XX ในทางกลับกันหากโปรแกรมเหล่านั้นถูกนำไปใช้กับแอพพลิเคชั่นที่ถูกต้องตรงตามที่ระบุ เป้าหมายที่ถูกเจาะเข้าไปอย่างไม่มีทางป้องกันได้

ความยากของการเจาะระบบด้วยเครื่องมือสำเร็จรูป จึงมิใช่การหาวิธีที่จะเจาะเข้าไปได้อย่างไรแต่อยู่ที่การค้นหาให้พบว่ามีใครที่จะเป็นเป้าหมายได้บ้าง แต่หากบังเอิญพบเป้าหมายที่ใช้แอพพลิเคชั่นเวอร์ชั่นตรงกับเครื่องมือที่มีอยู่พอดิบพอดี นั่นหมายถึงการเจาะระบบสำเร็จไปแล้ว

4.4.6 การวิเคราะห์แพ็กเกจโดยพิจารณาพอร์ตที่ใช้

ในการวิเคราะห์แพ็กเกจเพื่อตรวจสอบการใช้งานและหาร่องรอยการบุกรุกนั้น พอร์ตเป็นสิ่งสำคัญอันดับต้นๆที่จะใช้การพิจารณาว่าเป็นแพ็คเกจนั้นเป็นการสื่อสารของแอพพลิเคชั่นใด โดยจะต้องพิจารณาควบคู่ไปกับ TCP Flag เพื่อให้แน่ใจว่าหมายเลขพอร์ตที่พิจารณาอยู่นั้นทำหน้าที่เป็นไคลเอนต์พอร์ตหรือเซิร์ฟเวอร์พอร์ต ตัวอย่างเช่น

14:13:54:847401 10.15.14.20.2456 > 10.15.14.100.80 S

หมายความว่าโฮสต์ 10.15.14.20 ทำการส่งสัญญาณไปเพื่อขอสื่อสารกับ 10.15.14.100 และได้เปิดพอร์ตของตนเองไว้ที่หมายเลข 2456 เป็นไคล์เอนต์พอร์ตเพื่อรอการตอบกลับ และพอร์ตปลายทางที่ต้องการติดต่อด้วยคือพอร์ตหมายเลข 80 จากข้อมูลของแพ็กเกจเพียงเท่านี้ก็สามารถวิเคราะห์ได้ว่าโฮสต์หมายเลข 10.15.14.20 พยายามเริ่มต้นกระบวนการ 3-ways handshake ไปยัง 10.15.14.100 ที่มีแอพพลิเคชั่นให้บริการที่พอร์ต 80 ก็คือเว็บเซิร์ฟเวอร์นั้นเองข้อสันนิษฐานเบื้องต้นก็คือ 10.15.14.20 เป็นโฮสต์ที่ใช้โปรแกรมประเภทบราวเซอร์ และ 10.15.14.100 เป็นเว็บเซิร์ฟเวอร์ซึ่งอาจจะถูกหรือผิดก็ได้ขึ้นอยู่กับแพ็กเกจที่ตามมา

14:13:54.847401	10.15.14.20.2456	>	10.15.14.100.80 S
14:13.55.25401	10.15.14.100.80	>	10.15.14.20.2456 SA
14:13:55.66201	10.15.14.20.2546	>	10.15.14.100.80 A

จากข้อมูลข้างต้นแสดงว่ากระบวนการ 3 ways handshake ระหว่าง 10.15.14.20 กับ 10.15.14.100 ได้เกิดขึ้นอย่างสมบูรณ์ทำให้ข้อสมมติฐานดูใกล้เคียงมากขึ้นกล่าวคือแสดงว่าที่โฮสต์ 10.15.14.100 มีแอพพลิเคชั่นที่ทำงานอยู่บนพอร์ต 80 จริงและน่าจะเป็นเว็บเซิร์ฟเวอร์ ส่วนที่

10.15.14.20 ก็เป็นไคล์เอนต์ที่พร้อมสำหรับการสื่อสารกับเซิร์ฟเวอร์ที่พอร์ต 80 จึงมีแนวโน้มสูงมากที่จะเป็นเว็บบราวเซอร์

พอร์ตที่ปรากฏจะเป็นตัวระบุจุดมุ่งหมายของการสื่อสารนั้น เราสามารถวิเคราะห์การใช้งานเบื้องต้นได้โดยไม่จำเป็นต้องทำการตรวจสอบที่โฮสต์เลย เพียงอาศัยการตรวจจับแพ็คเกจที่ผ่านไปมาก็สามารถบอกได้ในระดับหนึ่ง การตรวจสอบการสื่อสารกันของโปรแกรม ตัวอย่างเช่น แบ็คออริฟิชจึงมักอาศัยการตรวจจับแพ็กเกจที่ใช้งานพอร์ตของแบ็คออริฟิชเป็นสำคัญ แต่อย่างไรก็ตามการวิเคราะห์แพ็กเกจนั้นจะต้องดูทิศทางหรือ TCP Flag ประกอบด้วยเสมอ เพราะมิฉะนั้นจะไม่ทราบว่าพอร์ตที่กำลังพิจารณาอยู่นั้นเป็นไคลเอนต์ พอร์ตหรือเป็นเซิร์ฟเวอร์พอร์ต

14:13:54.847401	10.15.14.20.2456	>	10.15.14.40.80 S
14:13:54.957422	10.15.14.20.2457	>	10.15.14.41.80 S
14:13:55.142341	10.15.14.20.2462	>	10.15.14.42.80 S
14:13:55.285414	10.15.14.20.2480	>	10.15.14.43.80 S
14:13:55.3465402	10.15.14.20.2493	>	10.15.14.44.80 S
14:13:55.881040	10.15.14.20.2502	>	10.15.1445.80 S

จากแพ็กเกจด้านบนนั้นจะเห็นว่าโฮสต์ 10.15.14.20 พยายามเริ่มต้นกระบวนการ 3 ways handshake กับพอร์ต 80 ของโฮสต์ 5 ตัวคือ 10.15.41.40 – 45 อย่างรวดเร็วโดยดูเหมือนว่าไม่สนใจจะรอคำตอบกลับของแต่ละโฮสต์ที่ SYN ไปเลยซึ่งไม่น่าจะเป็นพฤติกรรมปกติของเว็บบราวเซอร์หลายๆโปรแกรมพร้อมกันก็ไม่น่าจะเป็นไปได้เพราะเวลาใกล้ชิดกันเกินไป

4.5 Fragmentation

การแฟรกเมนต์ คือกระบวนการแบ่งแพ็กเกจที่มีขนาดใหญ่ออกเป็นแพ็กเกจที่มีขนาดเล็กหลายแพ็กเกจเพื่อให้เหมาะสมกับการส่งข้อมูลผ่านไปยังเน็ตเวิร์คต่างๆได้ แต่สิ่งที่ต้องคำนึงอยู่อยู่เสมอคือ IP เป็นโปรโตคอลที่เดินทางโดยอาศัยการเราต์ไปเป็นทอดๆผ่านทางเน็ตเวิร์คใด ผ่านอุปกรณ์ใด และจะต้องถูกส่งต่อไปอีกกี่ครั้ง

ด้วยการแพ็กเกจนั้นเดินทางออกไปโดยไม่สามารถทราบถึงเส้นทางข้างหน้าได้ และจะต้องผ่านหลายเน็ตเวิร์คกว่าที่จะถึงปลายทางนี่เอง จึงเป็นปัจจัยที่อยู่นอกเหนือการควบคุมของผู้บริหารระบบใดระบบหนึ่ง และการที่แพ็กเกจจะเดินทางไปได้ด้วยวิธีใดจึงจะต้องอาศัยความสามารถของโปรโตคอลเป็นหลัก ตัวโปรโตคอลเองเมื่อถูกออกแบบมาเพื่อให้มีความสามารถในลักษณะนี้จึงจำเป็นต้องมีความสามารถในการปรับตัวและยืดหยุ่นต่อความหลากหลายของเน็ตเวิร์กและอุปกรณ์ที่อยู่ในเน็ตเวิร์คได้เป็นอย่างดี

การศึกษาเรื่องการแฟรกเมนต์นอกจากจะเป็นไปเพื่อให้สามารถเข้าใจกลไกของ IP อย่างลึกซึ้งแล้ว การแฟรกเมนต์ก็ยังเป็นสิ่งจำเป็นในด้านความปลอดภัยเป็นอย่างมาก โดยมีการใช้การแฟรกเมนต์เป็นช่องทางในการสำรวจ เพื่ออำพรางตนเอง และการโจมตีเป้าหมายอย่างแพร่หลาย หากไม่เข้าใจพื้นฐานของแฟรกเมนต์ แล้วย่อมไม่สามารถปรับปรุงระบบรักษาความปลอดภัยของเน็ทเวิร์กให้รอดพ้นจากแฮกเกอร์ได้อย่างสมบูรณ์ โดยเฉพาะการโจมตี DoS การโจมตีที่ค่อนข้างมีประสิทธิภาพสูง มีผลให้เป้าหมายส่วนใหญ่มักจะไม่อยู่ในสภาพที่จะให้บริการได้เลยเมื่อโจมตี

4.5.1 MTU(Maximum Transmission Unit)

การที่เน็ตเวิร์กซึ่งต่อเชื่อมถึงกันล้วนแล้วแต่ใช้โปรโตคอล IP เหมือนกันนั้นมิได้เป็นการยืนยันได้ว่าเน็ทเวิร์กทั้งหมดเป็นระบบเดียวกันและเหมือนกันทุกประการ เพราะ IP ทำงานอยู่ในเลเยอร์ที่ 2 เป็นเลเยอร์ระบบกลางซึ่งต้องอาศัยเลเยอร์ที่ต่ำกว่าเป็นตัวส่งผ่านข้อมูลไปอีกชั้นหนึ่ง การที่เน็ตเวิร์กใช้ IP เหมือนกันจึงเป็นข้อตกลงในเลเยอร์ที่ 2 เท่านั้นส่วนในเลเยอร์ที่ 1 ย่อมสามารถผิดแผกแตกต่างกันไปตามความประสงค์ของเจ้าของเน็ตเวิร์กนั้นๆ

เลเยอร์ที่ต่ำกว่า IP คือดาต้าลิงค์เลเยอร์จะมีขนาดของการรับข้อมูลในแต่ละครั้งอยู่จำกัดค่าหนึ่งเรียกว่า MTU (Maxium Transmission Unit) หมายถึงในการให้ส่งข้อมูลในแต่ละครั้งของดาต้าลิงค์เลเยอร์จะสามารถมีข้อมูลมีข้อมูลได้สูงสุดเท่ากับขนาดของ MTU ไม่สามารถส่งข้อมูลได้มากกว่านี้ในครั้งเดียว ขนาดของ MTU นั้นจะถูกข้อจำกัดของโปรโตคอลแต่ละชนิดและกำหนดให้ไม่สามารถปรับเพิ่มรีลดลงด้วยคำสั่งใดๆของโปรโตคอลในเลเยอร์ที่สูงกว่าได้ จะทำหน้าที่บรรทุกข้อมูลที่เลเยอร์ที่สูงกว่าส่งลงมามีขนาดเล็กกว่า MTU ก็คงจะไม่มีปัญหาแต่อย่างใด แต่หากขนาดของข้อมูลที่ต้องการส่งมาขนาดใหญ่กว่า MTU ก็ต้องยกให้เป็นน่าที่ของเลเยอร์ที่สูงกว่าไปจัดการเองว่าจะทำอย่างไรกับกรณีนี้ดี เช่นลดขนาดของแพ็กเกจลง ทำการกระจายแพ็กเกจย่อย

ตารางที่ 4.2 ขนาดของMTU สำหรับลิงค์เลเยอร์แต่ละชนิด

Network	MTU(Bytes)
Hyper channel	65,535
16 Mbits/sec token-ring(IBM)	17,914
4 Mbits/sec token-ring (IEEE 802.5)	4,464
FDDI	4,352
Ethernet	1,500
IEEE 802.3/802.2	1,492
X.25	576
Point-to-Point	296

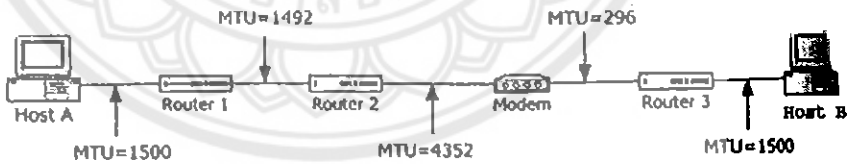
จากตารางที่ 4.2 แสดงให้เห็นขนาดของ MTU ของลิงค์เลเยอร์แต่ละชนิด ซึ่งขนาดของ MTU แต่ละชนิดส่วนใหญ่จะสัมพันธ์กับคุณสมบัติทางกายภาพของแต่ละโปรโตคอล จึงยากที่จะปรับเปลี่ยนขนาดของ MTU ตามความสามารถขนาดของ MTU จะสร้างปัญหาให้กับ IP ได้ใน 2 ลักษณะคือ

1. ขนาดของ IP ดาต้าแกรมไม่คงที่ สามารถเป็นไปได้ตั้งแต่ 30 ไบต์ไปจนถึง 65535 ไบต์และขนาดของ IP ดาต้าแกรมมีขนาดใหญ่กว่าขนาดของMTU ที่มีอยู่ในลิงค์เลเยอร์เกือบทั้งหมด ยกเว้น Hyper Channel กล่าวคือในเมื่อขนาดของ IP ดาต้าแกรมสูงสุดจะมีขนาดเท่ากับ 65535 ไบต์ นั่นหมายความว่าหากมีการส่งข้อมูลจาก IP ด้วยขนาดสูงสุดของ IP แล้วเป็นไปไม่ได้ที่ลิงค์เลเยอร์จะสามารถส่งข้อมูลได้ในครั้งเดียว

2. IP ออกแบบมาเพื่อให้สามารถใช้งานได้กับโครงสร้างของเน็ทเวิร์กที่หลากหลายและสามารถค้นหาเส้นทางการเดินทางได้ด้วยตนเอง โดยที่ระยะทางและจำนวนของเน็ตเวิร์คที่อยู่ระหว่างทางมิใช่ขีดจำกัด (นั่นทำให้การสื่อสารข้อมูลบนอินเตอร์เน็ทจึงต้องใช้ TCP/IP) แพ็กเกจจะต้องเดินทางลัดและผ่านเน็ตเวิร์คต่างๆ โดยเน็ทเวิร์กเหล่านั้นอาจจะใช้ลิงค์เลเยอร์ที่แตกต่างกันออกไป

4.5.2 Part MTU

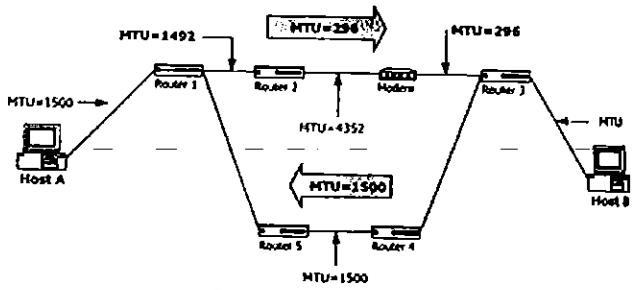
ในกรณีที่การสื่อสารเกิดขึ้นระหว่างสองโฮสต์โดยผ่านเน็ตเวิร์กหลากหลายชนิด เลเยอร์ที่ต่างกัน ทำให้ MTU ของแต่ละช่วงนั้นย่อมแตกต่างกันออกไป แต่อย่างไร MTU ระหว่างโฮสต์ทั้งคู่นั้นก็ยังคงมีจำกัดอยู่ที่ค่าใดค่าหนึ่ง



รูปที่ 4.2 เส้นทางข้อมูลมี MTU ที่แตกต่างกัน

ดังตัวอย่างในรูปที่ 4.2 เส้นทางของข้อมูลระหว่าง Host A กับ Host B ซึ่งมีขนาด MTU ที่อยู่ในเส้นทางเดินของข้อมูลที่แตกต่างกันตั้งแต่ 296 – 4352 ไบต์ขึ้นอยู่กับว่าจะพิจารณาในช่วงใดให้ขนาดของแพ็กเกจสามารถเดินทางผ่านทุกช่วงได้ในครั้งเดียว ค่า MTU จึงต้องเท่ากับค่าMTU ต่ำที่สุดที่อยู่ในเส้นทาง เนื่องจากหากมีขนาดของแพ็กเกจใหญ่กว่าค่า MTU ที่ต่ำสุด จะทำให้เมื่อแพ็กเกจเดินทางมาถึงช่วงที่ MTU ต่ำสุดดังกล่าวจะไม่สามารถถูกส่งผ่านไปได้ในครั้งเดียว ดังนั้น

เพื่อให้แพ็กเกจสามารถเดินทางไปได้จึงต้องมีขนาดเท่ากับMTUตรงที่ต่ำสุด และค่าMTU ที่ต่ำสุด จากตัวอย่างในรูปที่ 4.2 ก็คือ 296 ไบต์นั่นเอง



รูปที่ 4.3 Asymmetric Path MTU

สิ่งที่ต้องพิจารณาอีกอย่างนั่นคือ . นอกจากเส้นทางจะสามารถแปรเปลี่ยนได้ ทางของข้อมูลจะมีใน 2 ทิศทาง คือไปและกลับ ซึ่งไม่จำเป็นต้องใช้เส้นทางเดียวกันเสมอ แพ็กเกจสามารถใช้เส้นทางหนึ่งไปและกลับอีกเส้นทางหนึ่ง ดังนั้นการหาค่า Path MTU และกลับก็ไม่จำเป็นต้องเท่ากันเสมอไป Path MTU จึงสนใจเฉพาะเส้นทางระหว่างจุดหนึ่งไปยังอีกจุดหนึ่งเท่านั้นโดยไม่สนใจว่าโฮสต์ใดเป็นเซิร์ฟเวอร์ โฮสต์ใดเป็นไคลเอนต์ ดังภาพที่ 4.3 ที่ Path MTU จาก Host A – Host B เท่ากับ 296 ไบต์ แต่จาก Host B – Host A กลับเท่ากับ1500ไบต์ เนื่องจากใช่เส้นทางต่างกัน

4.5.3 Fragmentation

การแฟรกเมนต์คือการนำ IP ดาต้าแกรมมาแตกย่อยออกเป็นแพ็กเกจขนาดที่สามารถส่งผ่าน Path MTU ไปได้ การแฟรกเมนต์จะเกิดขึ้นเมื่อดาต้าแกรมมีขนาดใหญ่กว่า MTU แฟรก-เมนต์ อาจกระทำที่โฮสต์ต้นทางหรือที่เราเตอร์ระหว่างทางก็ได้ โดยส่วนย่อยแต่ละส่วนจะเรียกว่าแฟรกเมนต์ (fragment) ส่วนการรี แอสเซมเบิล (Reassemble เป็น การนำแฟรกเมนต์กลับมารวมเป็นดาต้าแกรมใหม่) จะเกิดขึ้นที่โฮสต์ปลายทาง

คุณสมบัติของแฟรกเมนต์

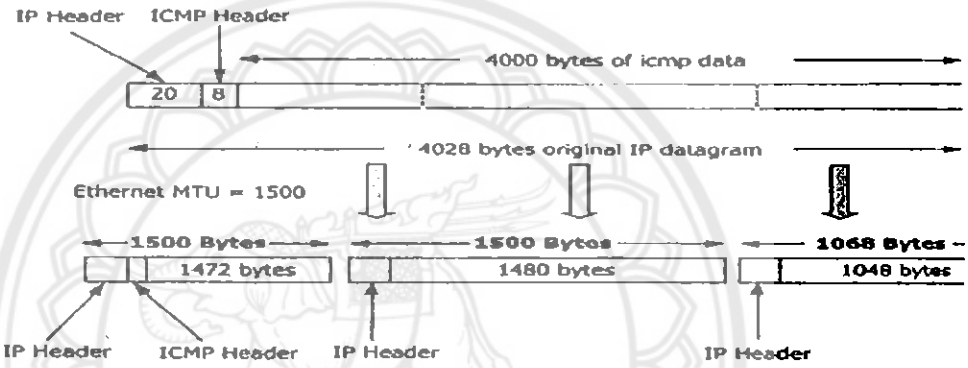
ทุกแฟรกเมนต์จะต้องมีคุณสมบัติดังนี้

ID : ทุกแฟรกเมนต์จะต้องมีหมายเลขอ้างอิง ของดาต้าแกรมอันเดียวกัน คือข้อมูลในฟิลด์ ID ของ IP Head นั่นเอง เพื่อให้เป็นข้อมูลที่บอกว่าแฟรกเมนต์ใดเป็นของดาต้าแกรมใด

Offset : แฟรกเมนต์จะต้องระบุตำแหน่งออฟเซ็ตเมื่อเทียบกับดาต้าแกรมต้นฉบับก่อนที่จาถูกทำการแฟรกต์เมนต์ นั่นคือต้องระบุว่า ส่วนย่อยนี้หากนำไปรีแอสเซมเบิลจะต้องถูกวางอยู่ตำแหน่งใด

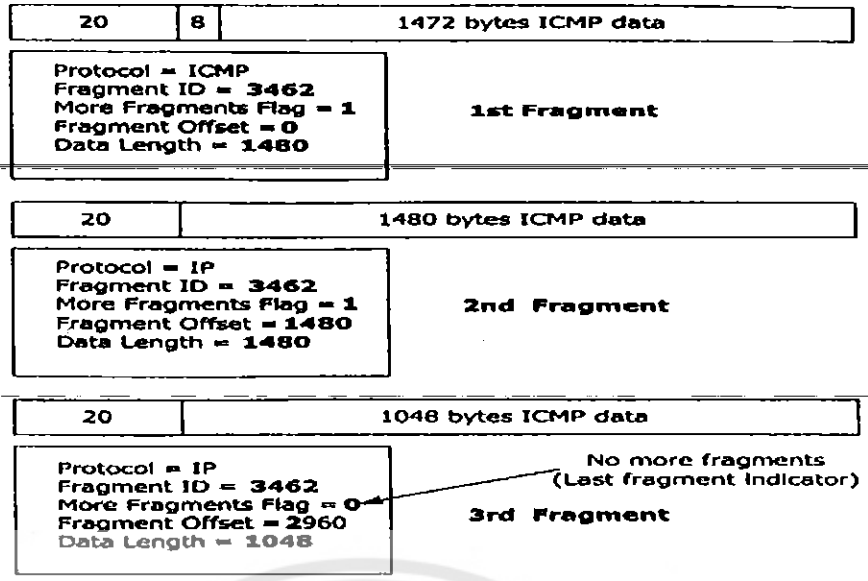
Payload : แฟรกต์เมนต์ต้องระบุขนาดของข้อมูลที่อยู่ในแฟรกเมนต์นั้นว่ามีข้อมูลขนาดเท่าใดที่ถูกแบ่งให้มาอยู่ในแกรมเมนต์นี้

More Fragment Flag : แฟรกเมนต์จะต้องระบุว่ามีแฟรกเมนต์อื่นตามมาอีกหรือไม่หากไม่มีแฟรกเมนต์อื่นตามมาหมายความว่าแฟรกเมนต์นี้เป็นแฟรกเมนต์สุดท้ายของดาต้าแกรม



รูปที่ 4.4 ตัวอย่างการแฟรกเมนต์ของ ICMP ขนาด 4000 ไบต์

จากตัวอย่างในรูปที่ 4.4 เป็นการแฟรกเมนต์ของ ICMP ขนาด 4000ไบต์ เมื่อนำมา encapsulate ด้วย IP ดาต้าแกรมขนาด 4028 ไบต์ เมื่อต้องการส่งผ่าน Ethernet ซึ่งขนาดของ MTU เท่ากับ 1500 จึงต้องนำมาทำการแฟรกเมนต์ให้มีขนาดไม่เกิน 1500 ไบต์เสียก่อน แต่เนื่องจากว่าทุกแฟรกเมนต์จะต้องมีเฮดเดอร์ตามปกติจึงต้องนำเฮดเดอร์ของ IP มาแนบไปกับด้วยเสมอ ดังนั้นทุกครั้งจึงต้องเสียเนื้อที่ 20 ไบต์ให้เท่ากับ IP Header สังเกตว่าในแฟรกเมนต์แรกจะต้องเสียเนื้อที่ส่วนหนึ่งให้กับ ICMP Header ด้วย จริงๆแล้วแฟรกเมนต์แรกมิได้มีอะไรแตกต่างจากแฟรกเมนต์อื่นๆ คือเสียเนื้อที่ให้กับ IP Header ไป 20 ไบต์เสมอ และเนื่องจากการแฟรกเมนต์นั้นทำให้ระดับ IP จึงไม่รู้จักว่าส่วนใดเป็น ICMP Header หรือ ICMP ดาต้าทั้งหมด IP จะถือว่าเป็นดาต้าของ IP ทั้งหมด และหลักการนี้เป็นเช่นเดียวกับการแฟรกเมนต์ของ TCP หรือ UDP



รูปที่ 4.5 รายละเอียดของแต่ละแฟรกเมนต์

โปรดสังเกตแฟรกเมนต์สุดท้ายซึ่งแฟรกของ More Frafment Flag (MF) จะถูกรีเซ็ตให้เป็น 0 เพื่อบอกให้ทราบว่าแฟรกเมนต์นี้เป็นแฟรกเมนต์สุดท้ายของ IP ดาต้าแกรมแล้วไม่มีแฟรกเมนต์อื่นใดอีก แต่ไม่ได้หมายความว่าโฮสต์ปลายทางจะได้รับแฟรกเมนต์นี้เป็นแพ็กเกจลำดับสุดท้าย การได้รับแพ็กเกจก่อนหลังอย่างไรนั้นขึ้นอยู่กับกระบวนการเราต์ติ้งของเราต์เตอร์ไม่เกี่ยวกับแฟรกนี้แต่อย่างใด

4.5.4 Fragmentation and Security

การแฟรกเมนต์มีประโยชน์มากสำหรับโปรโตคอล IP สามารถนำไปใช้งานได้กว้างขวาง และมีความยืดหยุ่นต่อโปรโตคอลในลิงค์เลเยอร์สูง ด้วยกลไกแฟรกเมนต์ที่มีอยู่ทำให้ IP สามารถปรับตัวเองให้ส่งผ่านไปยังลิงค์เลเยอร์ใดๆ ก็ได้โดยที่ยังคงความสามารถได้เช่นเดิม ปกติ การ แฟรก-เมนต์จะเกิดขึ้นเองโดยอัตโนมัติเมื่อจำเป็น นั่นคือเมื่อ IP นั่นคือเมื่อ IP เห็นว่าขนาดของดาต้าแกรมนั้นใหญ่เกินกว่าที่จะเดินทางไปได้ แน่นอนว่าหากการแฟรกเมนต์ นั้นเกิดจากกลไกของ IP เองก็ย่อมไม่มีปัญหาใดๆ แต่ในเมื่อ IP แพ็กเกจธรรมดาก็สามารถถูกสร้างปลอมขึ้นมาโดยไม่จำเป็นต้องถูกต้องตามโปรโตคอลได้แล้ว แฟรกเมนต์แพ็กเกจซึ่งก็เป็น IP แพ็กเกจอย่างหนึ่งก็ย่อมจะอยู่ในวิสัยที่สามารถที่สามารถสร้างปลอมขึ้นมาได้เช่นเดียวกัน โดยที่ไม่จำเป็นต้องถูกต้องตามขั้นตอนกระบวนการแฟรกเมนต์ที่กำหนดไว้ในโปรโตคอลแต่อย่างใด และแฟรกเมนต์ปลอมๆที่ถูกสร้างขึ้นมาเหล่านี้ก็เป็นปัญหาใหญ่ของความปลอดภัย

การปลอมแฟรกเมนต์แพ็กเกจก็เพื่อนทำให้ระบบรักษาความปลอดภัยมีความยุ่งยากมากขึ้นในการที่จะตรวจสอบรีป้องกัน เพราะจะต้องอาศัยการตรวจสอบที่ซับซ้อนมากขึ้นจึงจะสามารถพบ

ได้ โดยที่วัตถุประสงค์ของคาต้าแกรมนั้นก็ยังคงทำงานได้เช่นเดิม และไม่ได้รับผลกระทบจากการแฟรกเมนต์แต่อย่างใด ดังนั้นหากการแฟรกเมนต์ไม่ได้เกิดขึ้นจาก IP เอง ผู้ที่สร้างแฟรกเมนต์ก็สามารถเลือกลักษณะของแฟรกเมนต์ได้ตามใจชอบเช่น อาจจะแฟรกเมนต์ให้มีขนาดของแพ็กเกจเพียงแฟรกเมนต์ละ 20 ไบต์ก็สามารถทำได้ ข้อสังเกตเบื้องต้นของแฟรกเมนต์แพ็กเกจก็คือการแฟรกเมนต์ ตามปกติที่เกิดขึ้นโดย IP เองแล้วขนาดของแพ็กเกจจะต้องมีขนาดเท่ากับหรือใกล้เคียง Path MTU (ยกเว้นแฟรกเมนต์สุดท้ายซึ่งจะมีขนาดเท่ากับขนาดจริง) หากมีแฟรกเมนต์ใดที่มีขนาดแตกต่างจาก Path MTU มากก็ให้สันนิษฐานไว้ก่อนได้ว่าอาจจะมีสิ่งผิดปกติเกิดขึ้น

อีกประการหนึ่งที่ทำให้การกรองแพ็กเกจบนเราเตอร์ไม่สามารถใช้ได้ผลกับกับแฟรกเมนต์ก็คือการรีแอสเซมเบิลกลับมาเป็น IP คาต้าแกรมจะกระทำที่โฮสต์ปลายทางเท่านั้น นั่นทำให้เราไม่มีทางที่จะสามารถตรวจสอบเนื้อหาของแฟรกเมนต์ แพ็กเกจที่สมบูรณ์ได้

4.5.5 Stateful Inspection

นอกจากเราเตอร์แล้ว ไฟร์วอลล์ก็เป็นเครื่องมือที่ทำหน้าที่กรองแพ็กเกจเช่นกัน แต่ทว่าไฟร์วอลล์จะมีความสามารถมากกว่าในการตรวจสอบเลเยอร์ที่สูงขึ้นไป และจะยังสามารถทำการรีแอสแซมเบิลได้ด้วย ทำให้สามารถตรวจสอบเนื้อหาในคาต้าแกรมที่เข้ามาได้ จุดสำคัญของการตรวจสอบแฟรกเมนต์ก็คือ เนื่องจากการเดินทางเข้ามาของแฟรกเมนต์นั้นจะไม่เป็นไปตามลำดับก่อนหลังไฟร์วอลล์ที่จะสามารถตรวจสอบได้นั้นจะต้องมีความสามารถในการรีแอสแซมเบิล และจะต้องมีความสามารถวิเคราะห์สถานะของแฟรกเมนต์ได้ว่า มันมีความสัมพันธ์กับแฟรกเมนต์อื่นหรือไม่อย่างไรและจะต้องไม่พิจารณาแฟรกเมนต์เพียงแค่เป็น IP แพ็กเกจธรรมดาทั่วไป จะต้องนำแฟรกเมนต์นั้นไปทำการรีแอสแซมเบิลให้สมบูรณ์เสียก่อน จึงจะนำมาเปรียบเทียบกับกฎที่มีไว้ว่าคาต้าแกรมชนิดนี้รับอนุญาตให้ผ่านเข้าออกหรือไม่ หรือจะให้ไฟร์วอลล์ดำเนินการอย่างไร การนำแฟรกเมนต์ทั้งหมดมารวมกันแล้วตรวจสอบทั้งหมดนี้เองที่เรียกว่า Statefull Lnspection

การรีแอสแซมเบิลนั้นจะต้องใช้ทรัพยากรค่อนข้างมาก โดยเฉพาะหน่วยความจำ เพราะต้องจัดสรรสำหรับนำแฟรกเมนต์มาต่อเรียงกันในหน่วยความจำจนสมบูรณ์ทั้งคาต้าแกรมก่อนที่จะส่งต่อไปยังแอพพลิเคชั่นในขณะที่บางครั้งลำพังโฮสต์ที่รีแอสเซมเบิลเฉพาะคาต้าแกรมตนเองเพียงโฮสต์เดียวยังแทบจะยังไม่มีหน่อยความจำไม่พอ และทำงานได้ไม่สมบูรณ์ไฟร์วอลล์ที่ทำหน้าที่รีแอสเซมเบิลแทนโฮสต์หลายสิบหลายร้อยโฮสต์ย่อยจะต้องมีหน่อยความจำที่มากพอที่จะจัดการในเรื่องนี้ซึ่งก็ไม่ใช่เรื่องที่จะสามารถทำได้ง่ายนัก โดยเฉพาะอย่างยิ่งหากโดนโจมตีแบบ DoS ด้วยเทคนิคของแฟรกเมนต์แล้ว ไฟร์วอลล์ที่ออกแบบมาไม่ดีอาจหยุดทำงานก่อนโฮสต์เสียด้วยซ้ำ

นอกจากหน่วยความจำจะเป็นทรัพยากรหลักที่ใช้ในการรีแอสเซมเบิลแล้ว เวลาใช้ในการประมวลผลของ CPU ก็เป็นทรัพยากรที่สำคัญไม่ยิ่งหย่อนไปกว่ากัน ด้วยลักษณะความซับซ้อนใน

การรีแอสเซมเบิลของ IP ดังนั้น ประมวณผลที่ต้องเสียไปเพื่อนที่จัดการกับแฟรกเมนต์จึงมากกว่า แต่ก็เป็นการใช้เวลา CPU เพียงเล็กน้อยจนแทบจะไม่เห็นผลกระทบ แต่หากแบนด์วิดธ์ของโฮสต์มีขนาดใหญ่และมีปริมาณของแฟรกเมนต์มาก CPU ก็จะเริ่มทำงานมากขึ้นแปรผันตามปริมาณของแฟรกเมนต์ที่เข้ามา จึงมักจะพบเห็นโดยทั่วไปว่าการ DoS ที่ใช้แฟรกเมนต์เป็นอาวุธในการโจมตีจะส่งผลกระทบกับการทำงานของ CPU โดยตรงและใช้เวลาการประมวณผลของ CPU ทั้งหมดไปจัดการกับแฟรกเมนต์จนไม่สามารถให้บริการกับโปรแกรมอื่นๆได้

ดังนั้นไฟร์วอลล์ที่จะสามารถทำ Stateful Inspection ได้นั้นนอกจากจะต้องมีหน่อยความจำที่มากเพียงพอ และมี CPU ที่มีความเร็วสูงและจะต้องบริหารการใช้งานทรัพยากรทั้งหน่วยความจำและ CPU ได้มีประสิทธิภาพสูงมากจึงจะสามารถทำงานได้อย่างรวดเร็วและสามารถยืนหยัดต้านทานการ DoS ด้วยแฟรกเมนต์ได้

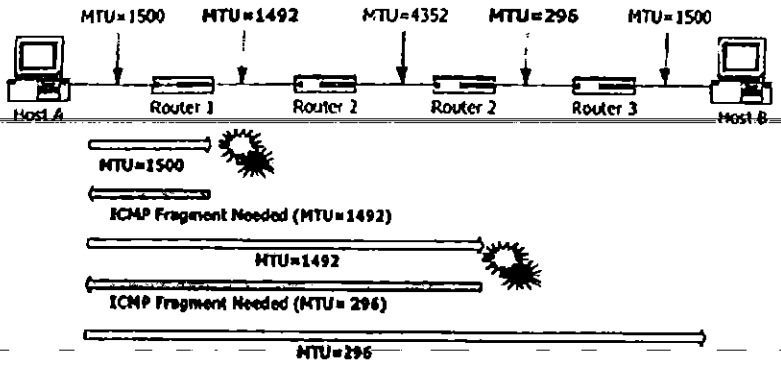
4.5.6 Don't Fragment

Don't Fragment Flag (DF) เป็นแฟรกที่ใช้สำหรับการกำกับดาต้าแกรมนั้น จะไม่ให้มีการแฟรกเมนต์โดยเด็ดขาด เนื่องจากในการทำงานทั่วไปนั้นการแฟรกเมนต์จะเกิดขึ้นเมื่อโปรโตคอล IP เห็นสมควร นั่นคือเมื่อ IP เห็นว่าดาต้าแกรมมีขนาดใหญ่กว่า Path MTU และจะเป็นไปโดยอัตโนมัติและอิสระใน IP เอง ทำให้การแฟรกเมนต์อาจจะเกิดขึ้นรึไม่เกิดขึ้นอยู่กับสภาพเส้นทางของการสื่อสารข้อมูลนั้นเอง ไม่เกี่ยวข้องกับปัจจัยอื่น

แต่เพื่อวัตถุประสงค์บางประการของผู้ใช้โปรโตคอลIP ก็มีช่องทางที่จะกำหนดให้แพ็กเกจนั้นถูกส่งไปยังปลายทางในสภาพเดิมเช่นเดียวกับต้นฉบับ จึงมีแฟล DF ให้กำหนดไปกับแพ็กเกจ ลักษณะนี้แพ็กเกจใดที่ถูกเซ็ตค่า DF เป็น 1 หมายความว่าดาต้าแกรมนั้นไม่อนุญาตให้ทำการแฟรกเมนต์ไม่ว่าในกรณีใดๆ ถึงแม้ว่าMTU ที่ดาต้าแกรมจะต้องเดินทางผ่านจะมีขนาดไม่เพียงพอก็ไม่อนุญาตให้ทำการแฟรกเมนต์

ดังนั้นเมื่อเกิดกรณีที่ลิงค์เลเยอร์ไม่สามารถจาส่งดาต้าแกรมต่อไปได้ เนื่องจาก MTU เล็กกว่าดาต้าแกรม IP ก็ไม่มีทางเลือกอื่นนอกจากการทำแพ็กเก็ตนี้ไป และแจ้งกลับไปยังผู้ส่งด้วย ICMP Fragment Needed พร้อมทั้งขนาดของ MTU ที่สามารถส่งได้

โดยส่วนใหญ่ DF จะถูกนำไปใช้ประโยชน์เพื่อการสำรวจ Path MTU ว่ามีขนาดเท่าไร โดยส่งดาต้าแกรมที่มีขนาดใหญ่ไปยังปลายทางพร้อมทั้งเซ็ตแฟรก DF ความผิดพลาดในเราเตอร์ช่วงใดช่วงหนึ่งจะได้รอรับ ICMP กลับมาพร้อมทั้งขนาด MTU จากเราเตอร์ช่วงที่มีปัญหา และโฮสต์ต้นทางจะทำการปรับขนาดของดาต้าแกรมให้มีขนาดที่จะผ่านMTU ช่วงนั้นไปได้และสามารถเดินทางไปไกลกว่าเดิม หากมีความผิดพลาด เนื่องจากขนาดของดาต้าแกรมในช่วงใดอีกนั้นโฮสต์ต้นทางก็ต้องปรับลดขนาดของดาต้าแกรมลงไปอีก จนกระทั้งแพ็กเกจนั้นสามารถผ่านไปจนถึงปลายทางได้โดยไม่มีข้อผิดพลาดใดๆ



รูปที่ 4.6 การใช้ DF เพื่อตรวจสอบ Path MTU

จากรูปที่ 4.6 จะเห็นข้อผิดพลาดที่เกิดจาก DF แฟรกมาประยุกต์ใช้เพื่อตรวจสอบ MTU โปรดสังเกตว่าหากดาต้าแกรมที่ส่วนออกไปไม่ได้เซต DF แฟรกไว้ จะไม่มีโอกาสได้เห็นความผิดพลาดได้เลยเพราะว่าไม่ว่าจะส่งแพ็กเกจของดาต้าแกรมขนาดเท่าใดไปแล้วเมื่อติดขัดไม่สามารถส่งผ่าน MTU ได้ก็จะถูก IP ทำการแฟรกเมนต์โดยอัตโนมัติ

4.6 การสำรวจเป้าหมายเบื้องต้น

ผู้ที่ทำการศึกษาพฤติกรรมของแฮกเกอร์ทุกคนจะทราบดีว่า ขั้นตอนเริ่มต้นก่อนการโจมตีเป้าหมายในแต่ละครั้งนั้นคือการหาข้อมูลของเป้าหมายมาวิเคราะห์ เพื่อหาจุดอ่อนที่สามารถโจมตีได้ง่ายที่สุด การสำรวจเป้าหมายนั้นเปรียบเสมือนการทำแผนที่เพื่อเป็นเส้นทางของการเดินทาง ยิ่งมีข้อมูลแม่นยำมากเท่าไร การกระทำต่อเป้าหมายก็จะสัมฤทธิ์ผลได้มากขึ้นเท่านั้น

การสำรวจเป้าหมายนั้นสามารถทำได้หลายระดับ เพื่อวัตถุประสงค์และเทคนิคที่แตกต่างกันหากเราสามารถล่วงรู้และตรวจผลการกระทำดังกล่าวไว้แต่เนิ่นๆ ย่อมทำให้สามารถเพิ่มความระมัดระวังและเตรียมพร้อมก่อนการถูกโจมตีจริงได้

4.6.1 การสำรวจเน็ตเวิร์ค

การสำรวจแบบนี้มีวัตถุประสงค์เพื่อจะตรวจสอบว่าในเน็ตเวิร์คเป้าหมายนั้นมีทรัพยากรอยู่มากน้อยเพียงใด ใช้อุปกรณ์อะไรอยู่บ้าง และมีการเชื่อมต่อกันอย่างไร ข้อมูลที่ได้นั้นจะเป็นภาพกว้างๆของทั้งหมด แม้ว่าเป็นข้อมูลเบื้องต้นแต่จะทำให้ทราบถึงขอบเขตของเป้าหมายได้เป็นอย่างดี ผลที่ได้จากการสำรวจนั้นจะสามารถบอกข้อมูลเหล่านี้ได้

จำนวนโฮสต์ที่มีอยู่ในเน็ตเวิร์ค

โดยทั่วไปในแต่ละเน็ตเวิร์ค จะสามารถคะเนถึงปริมาณโฮสต์ ที่มันจะสามารถมีได้จาก IP Address แต่จะไม่สามารถทราบได้จริงๆว่ามีโฮสต์ ที่เราใช้งานจริงอยู่มากน้อยแค่ไหนอย่างไร การสแกนแบบนี้จะเป็นการตรวจสอบจากสภาพใช้งานจริงของโฮสต์นั้น ผลลัพธ์ที่ได้จะถูกต้องแม่นยำ 100 เปอร์เซ็นต์ เพราะผลที่ได้เป็นการตอบรับในระดับเน็ตเวิร์คของทุกโฮสต์ที่สแกน การที่โอสต์ตอบรับการสแกนก็เป็นสิ่งยืนยันได้เป็นอย่างดีว่าโฮสต์ยังคงทำงานได้ตามปกติ และเน็ตเวิร์คในชั้นเลเยอร์ TCP/IP และเมื่อนำการตอบรับของทุกๆโฮสต์มาประมวลผลก็จะทราบได้ในเบื้องต้นว่าในเน็ตเวิร์คที่ถูกสแกนนั้นมีโฮสต์ใดทำงานอยู่บ้าง

ลักษณะการใช้งานของโฮสต์

เนื่องจากการตอบรับเมื่อถูกสแกนนั้นเป็นการแสดงให้เห็นถึงสภาวะการทำงานของโฮสต์ได้อีกทางหนึ่ง การสแกนนี้สามารถนำมาประยุกต์ใช้ในการตรวจสอบสถานการณ์ทำงานของโฮสต์ได้โดยการส่งสัญญาณไปสอบถามทุกๆโฮสต์ในเน็ตเวิร์คตามห้วงเวลาที่กำหนดไว้ ลักษณะการใช้งานของโฮสต์สามารถสันนิษฐานเบื้องต้นได้ จากเวลาในการเปิดปิดการใช้งานของโฮสต์ที่อยู่ในเน็ตเวิร์คนั่นเองลักษณะการสแกนเพื่อดูสภาวะของโฮสต์ จะทำให้แฮกเกอร์นั้นสามารถวิเคราะห์พฤติกรรมการใช้งานเครื่องคอมพิวเตอร์ต่างๆได้อีกด้วย

4.6.2 การสำรวจโฮสต์

การสำรวจในระดับเน็ตเวิร์คนั้นเป็นการสำรวจในระดับกว้างที่สุด เพื่อที่เราจะได้สามารถมองเห็นภาพรวมของเน็ตเวิร์คทั้งหมดและสามารถใช้ในการวางแผนเบื้องต้นได้ และการสำรวจในระดับต่อไปจะเพิ่มความละเอียดย่อยลงมาอีกระดับ คือการสำรวจในระดับโฮสต์ โดยที่แฮกเกอร์อาจจะเลือกโฮสต์ใดโฮสต์หนึ่งที่คาดว่าน่าจะสามารถเจาะเข้าไปได้ อย่างโดยง่ายโดยการวิเคราะห์พฤติกรรมการใช้งานเนื่องจากการสแกนเน็ตเวิร์คนั้นจะไม่มีข้อมูลที่ละเอียดมากพอ ที่จะชี้จุดอ่อนของระบบได้ส่วนใหญ่จะบอกในแง่ปริมาณมากกว่า แต่อย่างน้อยก็ระบุได้ว่า IP Address ใดที่มีโฮสต์เปิดใช้งานอยู่จริง และบอกว่าโฮสต์ตัวใดเปิดการใช้งานในเวลาใด ส่วนใหญ่โฮสต์ที่มักจะเป็นเป้าหมายแรกๆคือ โฮสต์ที่ไม่มีคนใช้งานประจำที่เครื่องนั้น คือเป็นเครื่องที่เปิดทิ้งไว้ นานๆจึงจะมีคนเข้ามา Log on ที่เครื่องสักครั้งเครื่องประเภทนี้มักจะไม่เป็นที่สนใจของผู้ที่ดูแลระบบ ไม่ค่อยมีคนมาตรวจสอบข้อผิดพลาด ทำให้แฮกเกอร์มีเวลาในการทดลองเจาะเข้าไปในระบบโดยไม่มีใครสังเกตได้

4.6.3 การสำรวจเพื่อหาแอพพลิเคชั่นเฉพาะ

จุดมุ่งหมายของการสำรวจในลักษณะนี้คือแฮกเกอร์อาจจะไม่มีเป้าหมายว่าจะเป็นโฮสต์ใดแน่นอนแต่จะเลือกเฉพาะเป้าหมายที่ใช้แอพพลิเคชั่นใดเป็นการเฉพาะ โดยส่วนใหญ่เราจะสามารถจำแนกจุดมุ่งหมายได้ดังนี้

สำรวจหาค่าแอพพลิเคชั่นเฉพาะเจาะจง

การเจาะเข้าไปในระบบโดยอาศัยจุดอ่อนของแอพพลิเคชั่นเป็นวิธีที่ง่ายและมีประสิทธิภาพที่สุดสำหรับแฮกเกอร์ ที่บังเอิญได้รับเครื่องมือสำหรับเจาะจงเลือกที่จะเข้าไปในแอพพลิเคชั่นใดแอพพลิเคชั่นหนึ่งเป็นการเฉพาะ เช่นทราบว่าดาต้าเบสเซิร์ฟเวอร์ ยี่ห้อหนึ่งมีปัญหาและสามารถโจมตีและบุกรุกเข้าไปได้โดยง่าย และแฮกเกอร์เองก็มีเครื่องมือพร้อมอยู่จึงทำการสำรวจดูว่ามีดาต้าเบสเซิร์ฟเวอร์ยี่ห้อดังกล่าวพร้อมหรือไม่ โดยการสแกนหาทางอินเตอร์เน็ตด้วยวิธีการเฉพาะ ซึ่งก็ดูได้จากการเปิดพอร์ตนั่นเอง ดาต้าเบสรุ่นที่ต้องการอาจจะใช้พอร์ตหมายเลข XXXX เพื่อใช้ในการติดต่อกับไคล์เอนต์ ดังนั้นการค้นหาดาต้าเบสนี้ที่เปิดให้บริการในอินเตอร์เน็ตก็จะกระทำได้โดยการสแกนหาเซิร์ฟเวอร์ใดๆก็ตามที่เปิดพอร์ต XXXX ได้โดยไม่สนใจพอร์ตอื่น

การสำรวจหาโปรแกรมแบ็คออริฟิซ

การสำรวจแบบนี้จะเป็นการสแกนหาโปรแกรม ที่ทำหน้าที่แอบเปิดพอร์ตในโฮสต์ทิ้งไว้แล้วเปิดโอกาสให้บุคคลอื่นสามารถเข้ามาดูข้อมูล หรือควบคุมโฮสต์นั้นได้โดยผ่านทางเน็ตเวิร์คโดยที่เจ้าของเครื่องอาจไม่รู้ตัว ซึ่งโปรแกรมประเภทนี้ เราจะสามารถควบคุมเครื่องที่เป็นระบบปฏิบัติการ Microsoft Windows ได้ทั้งหมด ไม่ว่าการแอบดูการกดแป้นพิมพ์ การดูหน้าจอ การเปิดแฟ้มข้อมูล การปิดเครื่อง เป็นต้น

การตรวจจับการสแกนประเภทนี้ทำได้ไม่ยากเพราะ โปรแกรมแบ็คออริฟิซแต่ละตัวมักจะใช้หมายเลขพอร์ตตายตัวสำหรับการสื่อสารระหว่างตัวมันกับเครื่องไคล์เอนต์อยู่แล้ว จากหมายเลขพอร์ตเหล่านี้ถือว่าเป็นพอร์ตอันตรายสำหรับ TCP/IP เลยที่เดียว ดังนั้นเพียงทำการจับตาดูแพ็กเกจที่พยายามสื่อสารโดยใช้หมายเลขพอร์ตก็สามารถตรวจพบได้

4.7 ทำแผนที่เป้าหมายโดยละเอียด

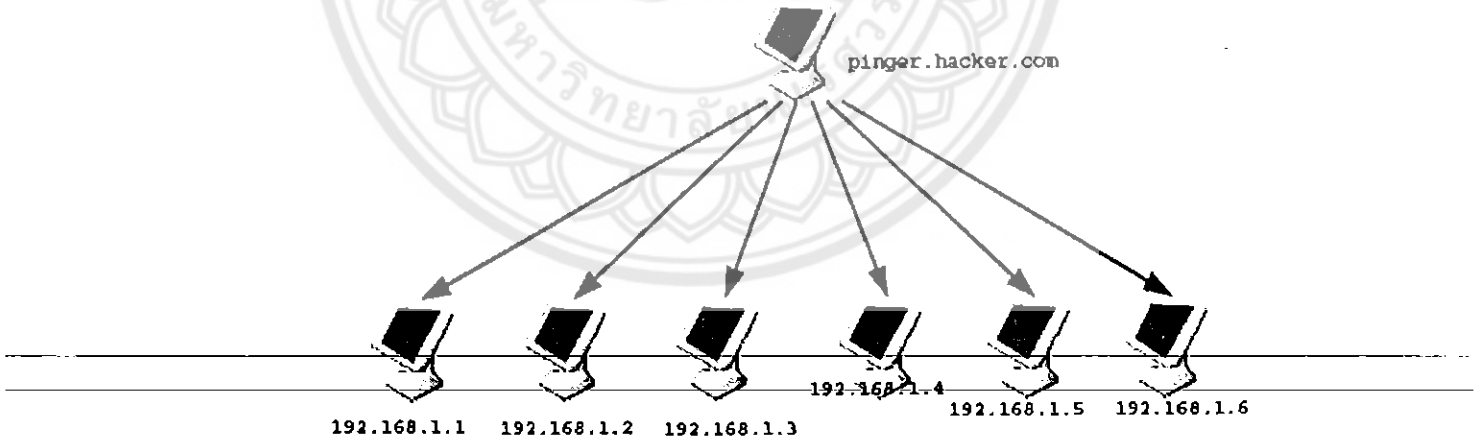
Network Mapping เป็นการสำรวจเน็ตเวิร์คโดยละเอียดเพื่อให้ได้มาซึ่งข้อมูลของทรัพยากรทุกๆอย่างที่มีอยู่บนเน็ตเวิร์ค ไม่ว่าจะเป็นจำนวนโฮสต์ ลักษณะการเชื่อมต่อของโฮสต์อุปกรณ์อื่นๆที่ไม่ใช่คอมพิวเตอร์ เส้นทางการเดินทางของแพ็กเกจ ช่องทางการสื่อสารกับเน็ตเวิร์คภายนอก การได้มาซึ่งข้อมูลรายละเอียดยิ่งมากก็ยิ่งเป็นประโยชน์ต่อแฮกเกอร์

4.7.1 Ping Sweep

หนึ่งในเทคนิคพื้นฐานที่นิยมกระทำกันก็คือ การ ping ไปยังเครื่องเป้าหมายจำนวนมากพร้อมๆกัน ในลักษณะคล้ายกับการกวาดหรือกราดยิง ซึ่งเราเรียกว่าการทำ ping sweep เพื่อตรวจสอบว่าเครื่องปลายทางใดบ้างที่ยังเปิดทำงานอยู่ โดยปกติ ถ้าคุณใช้คำสั่ง ping ธรรมดา, ping จะมีการส่ง แพ็กเกจ ICMP ECHO (Type 8) ออกไปยังเครื่องปลายทาง และจะทำการรอคอย ICMP

ECHO_REPLY (Type 0) ที่ถูกส่งกลับมา ถึงแม้ ping จะมีประโยชน์สำหรับการทดสอบว่าเครื่องปลายทางเปิดอยู่หรือไม่ก็ตาม แต่ว่ามันจะเหมาะสำหรับเครื่องที่อยู่บนเน็ตเวิร์คขนาดเล็กถึงขนาดกลางเท่านั้น มันจะไม่มีประสิทธิภาพเพียงพอที่จะนำมาใช้ตรวจสอบเครื่องที่อยู่บนเน็ตเวิร์คขนาดใหญ่ได้ การตรวจสอบเครื่องที่อยู่ในเน็ตเวิร์คที่ใช้แอดเดรสในคลาส A อาจจะกินเวลานานหลายชั่วโมงกว่าจะทราบผล เทคนิคในการ ping sweep มีหลายเทคนิคแตกต่างกันไป เช่น ปกติแล้วการ ping จะรอคอยการตอบสนองจากเครื่องทีละเครื่อง ก่อนจะเปลี่ยนไปทดสอบเครื่องอื่นๆถัดไป มันจะใช้การส่ง แพ็กเกจ ICMP ออกไปพร้อมๆกันแบบขนานไปยังเครื่องปลายทางหลายๆเครื่อง ในลักษณะคล้าย "Round Robin" (คือส่งแพ็กเกจ ICMP ไปที่เครื่อง 1,2,3,....ถึงเครื่องสุดท้าย แล้ววนกลับมาส่งแพ็กเกจไปที่เครื่อง 1,2,3 ใหม่ไปเรื่อยๆ แล้ววนกลับมาอีก โดยไม่จะเป็นต้องหยุดรอจากตอบสนองจากเครื่องแรก) ดังนั้น จะทำงานได้รวดเร็วกว่าคำสั่ง ping ธรรมดามาก แต่อาจทำให้เกิดทราฟฟิกจำนวนมากที่เกิดขึ้นอาจเข้าไปรบกวนแบนด์วิธของ WAN Link ความเร็วอย่างต่ำเช่น 128K ISDN หรือ เฟรมรีเลย์ (Frame relay) แต่ในบางครั้งจะทำอย่างไรถ้าที่เน็ตเวิร์คเป้าหมายได้มีการบล็อกห้ามแพ็กเกจ ICMP ไว้ไม่ให้เข้าถึงเน็ตเวิร์คภายในได้

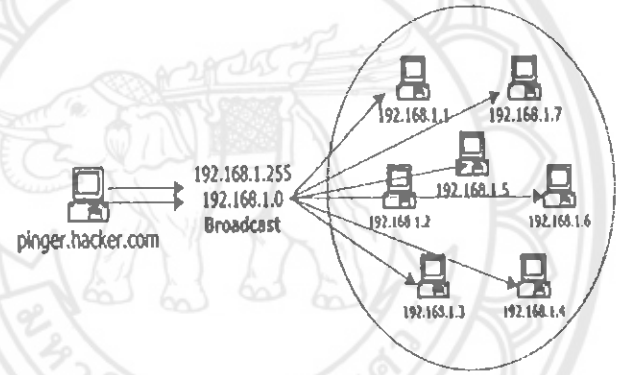
ถ้าแพ็กเกจ ICMP ถูกบล็อกเอาไว้ เรามีเทคนิคและเครื่องมืออื่นที่ใช้ตรวจสอบได้ว่าเครื่องปลายทางใดบ้างที่เข้าถึงได้และเปิดทำงานอยู่ แต่อย่างไรก็ตาม มันอาจไม่ถูกต้องและรวดเร็วเท่ากับตรวจสอบด้วย ping sweep



รูปที่4.7 แสดงการทำงานของ ping sweep

4.7.2 Broadcast Ping

วิธีนี้พัฒนาจากวิธี Individual Host ping ให้มีประสิทธิภาพมากขึ้น เนื่องจากวิธีที่แรกจะต้องทำการ ping โฮสต์ทุกตัวไล่ไปจนหมดเน็ตเวิร์ค หากจะทำการสำรวจเป้าหมายขนาดใหญ่จะต้องใช้เวลาค่อนข้างนาน เช่น หากว่าแฮกเกอร์ใช้แบนด์วิธต่ำ เช่น เลือกใช้โมเด็มในการต่อกับอินเตอร์เน็ตกว่าจะสแกนเสร็จแต่ละเน็ตเวิร์คต้องใช้เวลานานมาก วิธีที่ Broadcast Ping นี้จะแก้ไขปัญหาดังกล่าวโดยเป้าหมายเดิมยังคงไว้คือการพยายามส่ง ICMP Echo Request ไปยังทุกๆโฮสต์ โดยใช้แพ็กเกจน้อยลง โดยการ ping ไปที่ Broadcast Address ซึ่งจะทำให้ทุกโฮสต์ในเน็ตเวิร์คได้รับพร้อมกัน ซึ่งสามารถลดจำนวนแพ็กเกจที่ต้องส่งลงไปได้มากเท่าที่การบรอดคาสต์นั้นจะครอบคลุมโฮสต์จำนวนเท่าไร เช่น 255 โฮสต์ , 62255 โฮสต์ ก็เป็นได้โดยใช้เพียงการบรอดคาสต์เพียงแพ็กเกจเดียวจะทำให้ทุกเครื่องที่ได้รับ ICMP Echo Request ดังกล่าวทำการ Reply กลับมาตามปกติ และผู้สำรวจก็จะทราบทันทีว่าในเน็ตเวิร์คมีเครื่องใดเปิดใช้งานอยู่



รูปที่ 4.8 แสดงการ Broadcast Ping Packets

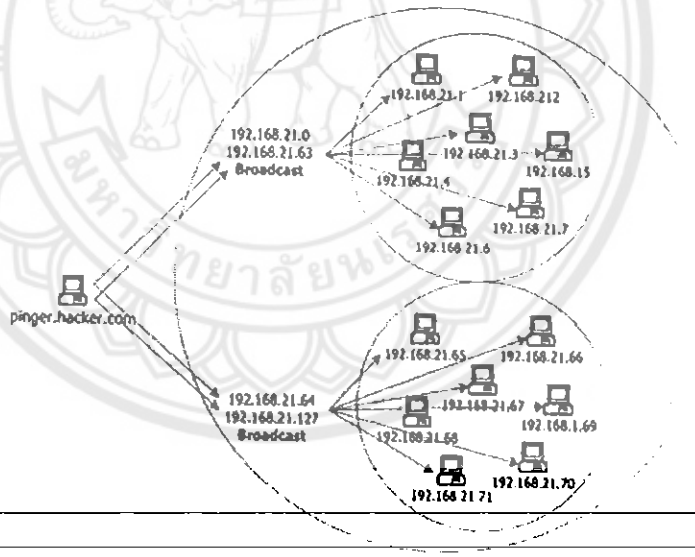
หากการ Ping นี้สัมฤทธิ์ผลหลังจากการ Ping ออกไปนั้นจะได้รับ ICMP Echo Reply จำนวนมากกลับมายังต้นทางเท่ากับจำนวนโฮสต์ที่เปิดใช้งานอยู่ขณะนั้น ซึ่งผู้ที่ส่ง Ping ออกไปก็จะต้องนำ ICMP Echo Reply เหล่านั้นไปประมวลผลให้ทันการ จากตัวอย่างจะแสดงให้เห็นการทำงาน Ping ไปครั้งละ Subnet ไปเรื่อยๆ และแต่ละครั้งจะหน่วงเวลาให้ห่างกันเล็กน้อย เพื่อเปิดโอกาสให้แฮกเกอร์มีเวลามากพอที่จะรับ ICMP Echo Reply จากเป้าหมายให้ครบถ้วนเสียก่อน

จากตัวอย่างในรูปที่ 4.8 จะพบว่าเป็นการส่งแบบ ICMP Echo Reply จากโฮสต์ที่มีชื่อว่า Hacker.Pinger.com โดยทำการ Ping ไล่ตั้งแต่บรอดคาสต์เน็ตเวิร์คในแต่ละเน็ตเวิร์คทั้งแบบ 255 และแบบ 0 ระยะห่างของการ Ping แต่ละครั้งจะเว้นไว้เพื่อรอรับ ICMP Echo Reply ที่จะตอบกลับมาจากเน็ตเวิร์คที่ถูกสแกน

ลักษณะการวิเคราะห์จุดมุ่งหมายของแพ็กเกจที่ใช้ในการสแกนแบบนี้กล่าว คือการ Ping โฮสต์นั้นอาจเป็นการ Ping ตามปกติที่ไม่มีจุดมุ่งหมายในทางร้ายก็ได้ ซึ่งจะต้องตรวจสอบข้อมูลเพิ่มเติมว่าเป็นการ Ping เพื่อมุ่งร้ายหรือไม่เพราะลักษณะแพ็กเกจเหมือนเดิม แต่หากการ Ping ที่ Broadcast Address ให้สันนิษฐานว่ามีเป้าหมายเพื่อมุ่งร้ายไม่ว่าจะเพื่อสำรวจเน็ตเวิร์คหรือ DoS เพราะการทำงานปกติของแอพพลิเคชั่นทั่วไปมีโอกาสน้อยมากที่จะทำการส่ง ICMP Echo แพ็กเกจออกไปโดยปลายทางที่บรอดคาสต์แอดเดรส

4.7.3 Subnet Broadcast Ping

ด้วยหลักการของการ Ping Broadcast นั้นยังคงมีขีดจำกัดบางประการคือค่า 255 ซึ่งอาจจะเป็นค่าบรอดคาสต์ที่ใช้ได้ผลสำหรับเน็ตเวิร์คในคลาสซีดั้งเดิม ที่ยังไม่มีการแบ่งเป็นเน็ตเวิร์คย่อย ซึ่งหากเน็ตเวิร์คถูกแบ่งแล้ว ค่าบรอดคาสต์แอดเดรสของแต่ละเน็ตเวิร์คย่อยก็มิใช่ค่า 255 ดังนั้นการ Ping โดยใช้แอดเดรสของ 255 และ 0 ย่อมไม่สัมฤทธิ์ผลจึงได้มีการปรับเทคนิคการ Ping โดยให้สอดคล้องกับลักษณะของเน็ตเวิร์คย่อยก็คือการเลือกPing ไปยังแอดเดรสที่น่าจะเป็นแอดเดรสของบรอดคาสต์ของทุกเน็ตเวิร์คย่อยด้วย



รูปที่ 4.9 แสดงSubnet Broadcast Ping

จากข้อมูลในแพ็กเกจในรูปที่ 4.9 เป้าหมายของการ Ping แต่ละแพ็กเกจตามลำดับนั้นก็คือการบรอดคาสต์สำหรับเน็ตเวิร์คย่อย ซึ่งสมมติฐานว่าเน็ตเวิร์คซีคลาส 192.168.21.0 น่าจะถูกแบ่งออกเป็น 4 เน็ตเวิร์คย่อยโดย Subnet Mask = 255.255.255.192

ตารางที่ 4.3 บรอดคาสต์แอดเดรสของเน็ตเวิร์คย่อย

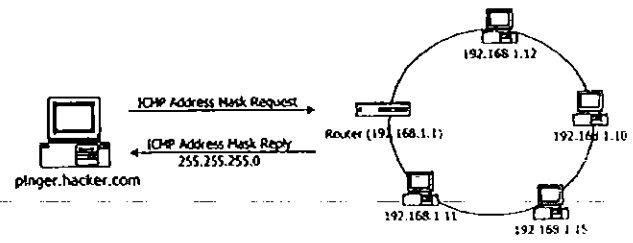
Net ID	Subnet ID	Broadcast(Normal)	Broadcast(BSD)
192.168.21.0	0	192.198.21.63	192.168.21.0
192.168.21.0	64	192.168.21.127	192.168.21.64
192.168.21.0	128	192.168.21.191	192.168.21.128
192.168.21.0	192	192.168.21.255	192.168.21.192

จะเห็นได้ว่าเพียงแค่ใช้ ICMP Echo Request จากโฮสต์ Scanner.net ไปยังบรอดคาสต์แอดเดรสของเน็ตเวิร์คย่อยทีละเน็ตเวิร์ค หากสมมติฐานถูกต้องโฮสต์ที่ตั้งไว้ทุกตัวในทุกเน็ตเวิร์คย่อยของเนตเวิร์ก 192.168.21.0 จะตอบ ICMP Echo Request มายัง Scanner.net

การใช้ ICMP Echo Request ในการสแกนเน็ตเวิร์คนั้นก็มีหลายเทคนิคเพื่อที่จะให้ได้มาซึ่งข้อมูลของเนตเวิร์กเป้าหมายโดยเร็วที่สุดและมีประสิทธิภาพที่สุด ในสภาพการใช้งานจริงๆนั้นอาจมีแพ็กเกจอื่นๆปะปนอยู่จำนวนมาก การวิเคราะห์ที่จะกระทำได้อย่างรวดเร็วคือการพยายามจำแนกเฉพาะออกมาเฉพาะ ICMP Echo อย่างเดียวก่อนเพื่อจะได้เห็นภาพรวมของการใช้แพ็กเกจเหล่านี้ในเน็ตเวิร์ค จากนั้นจึงนำมาหาความสัมพันธ์ของแต่ละแพ็กเกจและวิเคราะห์ลึกลงไปในระดับที่ซับซ้อนมากยิ่งขึ้น หากแพ็กเกจนั้นมีเป้าหมายในการสแกนเน็ตเวิร์คจริงๆ IP Address ที่ปรากฏจะเป็น IP Address จริงของผู้ส่งเสมอ

4.7.4 Address Mask Request

นอกจาก ICMP Echo Request ที่มักจะถูกใช้เพื่อทำการสำรวจเน็ตเวิร์คเสมอแล้วยังมี ICMP Type 17 (ICMP Address Mask Request) ที่ใช้สำหรับสอบถาม Address Mask จากเราเตอร์เพื่อที่จะได้ทราบว่าในเน็ตเวิร์คนั้นมีการแบ่งเน็ตเวิร์คย่อยอย่างไร จุดประสงค์ของการมีคำสั่งนี้ก็เพื่อกรณีที่นำโฮสต์ใหม่มาติดตั้งในเน็ตเวิร์คจะสามารถสอบถามค่า Subnet Mask ได้โดยไม่จำเป็นต้องไปดูโฮสต์จริง หรือกรณีที่โฮสต์ไม่มีดิสก์สามารถสอบถามค่า Subnet Mask จากเราเตอร์ทุกครั้งที่เปิดเครื่องขึ้นมาใช้งาน



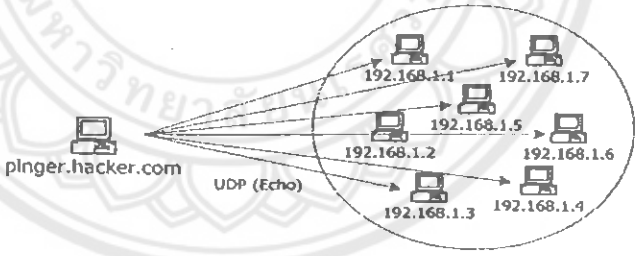
รูปที่ 4.10 Address Mask Request Scan

ในทางกลับกันคำสั่งนี้ก็ทำให้แฮกเกอร์ใช้การสอบถามข้อมูลนี้ ไปใช้เป็นส่วนหนึ่งของกระบวนการสำรวจเพื่อทำให้ได้มาซึ่งข้อมูลของเน็ตเวิร์คเป้าหมายมากที่สุด ดังนั้นจะเห็นว่าการสำรวจเป้าหมายด้วยวิธี Subnet Broadcast Ping นั้นหากจะให้มีประสิทธิภาพ ก็ควรที่เราจะทราบ Subnet Mask ด้วยเพราะมิฉะนั้น การส่งแพ็กเกจไปยังบอร์ดคาสต์แอดเดรสจะไม่ค่อยสัมฤทธิ์ผล

ด้วย ICMP Request นี้จะทำให้ผู้ส่งคำสั่งสามารถทราบได้ว่าเนตเวิร์กเป้าหมาย Subnet Mask เป็นอย่างไร ซึ่งจะเป็นประโยชน์ต่อการใช้สำรวจแบบ Subnet Broadcast Ping เป็นอย่างมาก โดยทั่วไป ICMP Request นี้ไม่มีการใช้งานมากนักเพราหากว่าโฮสต์ได้ทำการติดตั้งใช้งานอยู่ภายในเน็ตเวิร์คแล้วไม่มีความจำเป็นต้องสอบถาม Subnet Mask อีกดังนั้นไม่ควรที่จะเปิดบริการของ ICMP นี้บนเราเตอร์ให้ใช้งานเพราะจะเกิดโทษมากกว่าประโยชน์

4.7.5 UDP Echo Service

นอกจาก ICMP Echo ที่มีพฤติกรรมสะท้อนกลับแพ็กเกจแล้ว ยังมีบริการอีกชนิดหนึ่งซึ่งทำงานบน UDP เปิดให้บริการอยู่ที่พอร์ตหมายเลข 7 เรียกว่า "Echo Service" บริการนี้มีลักษณะการทำงานคล้าย ICMP Echo Request & Reply โดยแอพพลิเคชั่นที่ให้บริการนี้จะทำการสะท้อนกลับข้อมูลใดๆที่ส่งเข้ามายังพอร์ต UDP หมายเลข 7 ที่แอพพลิเคชั่นนี้ทำงานอยู่กลับไปยังผู้ส่งเสมอ ซึ่งการทำงานเช่นนี้สามารถนำไปประยุกต์ใช้กับการสำรวจเน็ตเวิร์คได้โดยง่าย และผลรับที่ได้เหมือนกับการใช้ ICMP Echo Request ทุกประการ



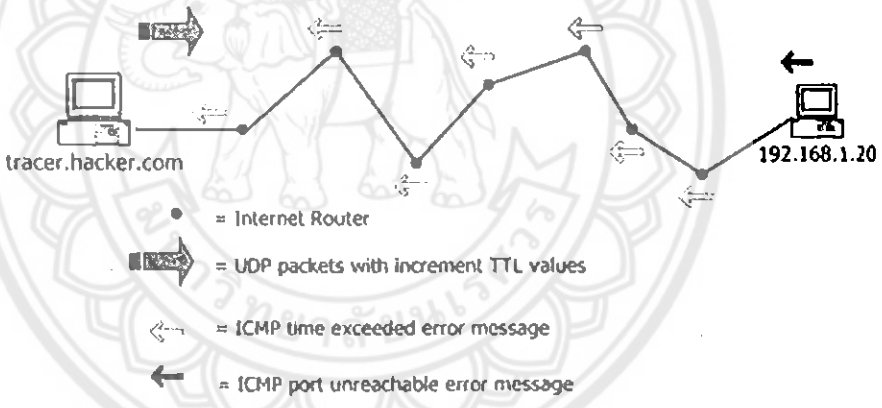
รูปที่ 4.11 UDP Echo Scan

การใช้ UDP ในการสแกนมีข้อดีคือจะไม่เป็นที่สังเกตได้ง่ายเพราะบริการ Echo ของ UDP ไม่เป็นที่รู้จักของผู้ดูแลระบบเท่าไรนัก ต่างจากการใช้ ICMP ซึ่งเป็นที่รู้จักโดยทั่วไปและสังเกตได้ง่าย ยิ่งกว่านั้น ICMP มักจะเป็นโปรโตคอลลำดับต้นๆที่ถูกไฟร์วอลล์บล็อกเอาไว้ ดังนั้นเมื่อใช้ ICMP เข้าไปไม่ได้ก็จะใช้ UDP เป็นเครื่องมืออันต่อไป หากเปรียบความง่ายแล้วการใช้ ICMP สแกนจะมีความสะดวกมากกว่า โดยส่วนใหญ่แฮกเกอร์น่าจะใช้ ICMP เป็นเครื่องมืออันดับแรก หากไม่สำเร็จจึงจะใช้เทคนิคในขั้นสูงต่อไป ขีดจำกัดของการสแกนขึ้นอยู่กับโฮสต์โดยจะสแกนได้

กับโฮสต์ที่เปิดให้บริการ Echo เท่านั้น ทำให้การสแกนแต่ละครั้งอาจได้ข้อมูลของโอสต์ไม่ได้ทั้งหมดที่อยู่ในเน็ตเวิร์คเพราะอาจมีโฮสต์บางตัวที่อาจไม่ได้เปิดให้บริการ Echo ไว้หรือโอสต์บางตัวอาจจะเป็นอุปกรณ์อื่นที่ทำงานอยู่บน TCP/IP แต่ไม่ใช่คอมพิวเตอร์ สำหรับคอมพิวเตอร์นั้นหากผู้บริหารระบบได้ทำการติดตั้งระบบปฏิบัติการได้อย่างถูกต้องแล้ว ถึงแม้ว่าถูกสแกนก็อาจจะไม่ตอบ แต่อย่างไรในสภาวะปกติไม่น่าจะมีแพ็กเกจที่ใช้บริการดังกล่าวในเน็ตเวิร์ค

4.7.6 Trace route

โปรแกรม Trace Route อาศัยคุณสมบัติของการหมดอายุลงของแพ็กเกจมาประยุกต์ใช้เป็นเครื่องมือในการตรวจสอบติดตามเส้นทางของแพ็กเกจ เพื่อให้ทราบว่าหากมีการส่งข้อมูลจากโฮสต์ต้นทาง ไปยังโฮสต์ปลายทางที่กำลังสนใจอยู่นั้น แพ็กเกจจะต้องเดินผ่านเราเตอร์จุดไหนบ้าง IP Address อะไร แต่ละช่วงของการเดินทางของแพ็กเกจใช้เวลาเดินทางมากน้อยแค่ไหน โปรแกรมจะทำงานโดยจงใจส่งแพ็กเกจให้ไปหมดอายุที่เราเตอร์ เพื่อให้เราเตอร์ทราบความผิดปกตินี้และรายงานกลับมา โดย ICMP แพ็กเกจที่เราเตอร์ส่งกลับมานั้นจะมี IP Address ของเราเตอร์ติดมาด้วย ทำให้ทราบ IP Address ของเราเตอร์ได้อัตโนมัติ



รูปที่ 4.12 Trace route

ขั้นตอนการทำงานของโปรแกรม Trace route

1. โปรแกรมจะเริ่มต้นด้วยการส่ง UDP แพ็กเกจไปยังปลายทางโดยให้ค่า TTL = 1เพื่อให้แพ็กเกจดังกล่าวหมดอายุที่ตัวแรกสุด

2. เพิ่มค่า TTL =2 เพื่อให้แพ็กเกจหมดอายุที่เราเตอร์ตัวที่สอง และเพื่อเพิ่มค่าในหมายเลขพอร์ตอีก 1

3.เพิ่มค่า TTL ขึ้นอีกทีละ 1 เช่นเดียวกับขั้นตอนที่ 2 แพ็กเกจก็จะหมดอายุที่เราเตอร์ตัวถัดไปอีก ทำเช่นนี้ไปเรื่อยๆจนกว่าจะไม่ได้รับ ICMP Time exceeded in transit จากเราเตอร์ใดๆ กลับมานั่นแสดงว่าแพ็กเกจถึงปลายทางแล้ว เนื่องจากแพ็กเกจไม่ต้องถูกเราต์อีก

4. รอรับ ICMP ที่ตอบกลับมาอีก 1 แพ็กเกจ ซึ่งอาจจะมีโอกาสเป็นได้ 2 กรณีก็คือ

- ICMP host unreachable คือแพ็กเกจได้เดินทางถึงเน็ตเวิร์คเป้าหมายแล้ว แต่ว่าโฮสต์ที่ปลายทางไม่ได้ทำงานอยู่จึง ติดต่อไม่ได้

- ICMP port unreachable คือ แพ็กเกจได้เดินทางถึงปลายทางโดยสมบูรณ์

4.8 สแกนพอร์ต TCP

เมื่อพอร์ตเป็นสิ่งสำคัญ การสแกนพอร์ตก็ถือเป็นขั้นตอนที่สำคัญเช่นเดียวกัน สิ่งที่ทำให้การสแกนพอร์ตเป็นการกระทำที่ถือได้ว่ามุ่งร้ายต่อระบบคือ ผลลัพธ์ของการสแกนพอร์ตจะทำให้แฮกเกอร์สามารถล่วงรู้ได้ว่ามีแอพพลิเคชันใดบ้างที่ทำงานอยู่บนโฮสต์ โดยปกติทั่วๆไปแล้วนั้นแอพพลิเคชันแต่ละชนิดที่เปิดให้บริการอยู่ จะได้หมายเลขพอร์ตที่ตายตัว และรู้จักกันโดยทั่วไป เมื่อผลการสแกนปรากฏว่ามีพอร์ตใดที่เปิดให้บริการอยู่ ก็สามารถนำข้อมูลที่ได้มาเทียบกับบริการมาตรฐาน ก็จะทราบได้ว่ามีแอพพลิเคชันใดที่ให้บริการอยู่ และข้อมูลเหล่านี้ก็จะเป็นประโยชน์ต่อการเลือกเทคนิคในการโจมตีต่อไปการสแกนพอร์ตนี้จะเป็นการสำรวจแต่ละโฮสต์ โดยผลลัพธ์ที่ได้จะทำให้ทราบว่าพอร์ตเป้าหมายเปิดให้บริการอยู่หรือไม่ แต่เนื่องจากการสแกนพอร์ตนี้เป็นการกระทำที่ส่อเจตนามุ่งร้ายอย่างชัดเจน เพราะในการปฏิบัติงานทั่วไปมีการสแกนพอร์ตน้อยมาก การสแกนพอร์ตจึงเข้าข่ายการบุกรุกเครือข่าย ดังนั้นเทคนิคในการสแกนจึงมีวิธีการที่ซับซ้อนขึ้นเรื่อยๆ เพื่อมิให้เป้าหมายรู้ตัว และไม่สามารถตรวจพบได้โดยง่ายทั้งนี้เราจะนำการสแกนพอร์ตมาใช้ในการตรวจสอบว่า ไฟร์วอลล์ที่ทำมาทดสอบนั้น สามารถป้องกันการสแกนพอร์ตได้หรือไม่

4.8.1 วิธี Connection Request

เทคนิคนี้เป็นการทำทีเสมือนว่าต้องการติดต่อไปยังแอพพลิเคชันที่ทำงานอยู่บนเซิร์ฟเวอร์นั้น โดยส่งสัญญาณไปขอเริ่มสื่อสารบนพอร์ตเป้าหมายบนเซิร์ฟเวอร์ จากนั้นก็รอผลตอบกลับจากพอร์ตนั้น ๆ ว่าจะตอบรับคำขอหรือไม่ หากมีแอพพลิเคชันอยู่ที่พอร์ตดังกล่าว ก็ย่อมจะต้องตอบรับและส่งสัญญาณมาเพื่อเริ่มการเชื่อมต่อในลำดับถัดไป

4.8.2 วิธี SYN Scan

การสแกนพอร์ตแบบนี้ หากพิจารณาดูแล้วจะใกล้เคียงกับวิธี TCP connect scanning แต่ที่ว่าสิ่งที่แตกต่างกันคือ วิธีนี้ผู้สแกนจะทำการส่ง SYN แพ็กเกจมาเพื่อทำการติดต่อเองโดยตรงกับเป้าหมายโดยไม่ผ่านระบบปฏิบัติการ และรอผลการตอบรับของเป้าหมายกลับมาด้วย SYN ACK หรือหากไม่มีแอพ-พลิเคชันทำงานอยู่ก็จะตอบกลับมาด้วย RST

4.8.3 วิธี FIN Scan

เมื่อ TCP SYN scanning นั้นสามารถทำได้โดยง่ายก็ย่อมสามารถถูกตรวจจับได้โดยง่ายเช่นกัน แต่อีกวิธีหนึ่งคือ TCP FIN scanning เป็นการสแกนที่สังเกตได้ค่อนข้างยาก โดยเฉพาะหากลำดับของพอร์ตของการสแกนเป็นแบบสุ่มและเว้นระยะพอสมควร ซึ่งจะทำให้แพ็กเกจที่ใช้สแกนสามารถเล็ดรอดเข้ามาได้โดยไร้ร่องรอย

โดยปกติ FIN แพ็กเกจจะเป็นแพ็กเกจของ TCP ที่จะส่งเมื่อยุติการติดต่อ นั่นหมายถึงจะต้องมีการสื่อสารกันมาก่อนแล้ว แต่ TCP FIN scanning จะเป็นการส่ง FIN แพ็กเกจไปยังเป้าหมายโดยไม่มีการสื่อสารใด ๆ มาก่อน และเป็นที่แน่นอนว่าเป้าหมายที่ได้รับจะต้องทราบอย่างแน่นอนว่า ไม่เคยได้รับการติดต่อจากไอพีแอดเดรสและพอร์ตต้นทางนั้นมาก่อนเลย แต่อย่างไรโฮสต์เป้าหมายก็ยังคงตอบ FIN แพ็กเกจนั้นกลับไปอยู่ดี

จุดสำคัญก็คือโดยปกติการตอบ FIN แพ็กเกจกลับไปของพอร์ตที่เปิดไว้ และพอร์ตที่ไม่ได้เปิดให้บริการก็ไม่เหมือนกัน หากเป็นพอร์ตที่ไม่ได้เปิดอยู่โฮสต์ก็จะตอบด้วย RST FLAG และหากเป็นพอร์ตที่เปิดให้บริการอยู่ก็จะต้องด้วย FIN ACK กลับไป

4.8.4 วิธี SYN/FIN Scan

วิธีการนี้จะใช้ TCP Flag ทั้ง SYN และ FIN พร้อมกัน ซึ่งการส่งทั้ง SYN และ FIN พร้อมกัน นั้นไม่มีกำหนดอยู่ในโปรโตคอล ดังนั้นการตอบรับของโฮสต์แต่ละประเภทในกรณีที่เปิดพอร์ตนั้นอยู่จะแตกต่างกันออกไป อาจจะตอบรับเป็น SYN ACK หรือ FIN ACK ส่วนการตอบรับในกรณีที่พอร์ตปิดจะตอบเหมือนกันคือ RST

4.9 สแกนพอร์ต UDP

4.9.1 Basic UDP Scanning

เทคนิคนี้จะส่งแพ็กเกจของโพรโตคอล UDP ไปยังพอร์ตเป้าหมาย ถ้าเครื่องปลายทางตอบกลับมาด้วยแพ็กเกจ ICMP type PORT UNREACHABLE นั่นหมายความว่าพอร์ตนั้นปิดอยู่ในทางตรงกันข้าม ถ้าเราไม่ได้รับแพ็กเกจ ICMP type ดังกล่าว เราสามารถสรุปได้ว่าพอร์ตนั้นเปิดอยู่ เนื่องจากโพรโตคอล UDP เป็นโพรโตคอลลักษณะconnectionless คือไม่รับรองว่าแพ็กเกจที่ส่งไป

จะถึงเครื่องปลายทางครบถ้วนหรือไม่ ดังนั้นความถูกต้องของผลลัพธ์ที่ได้จากเทคนิคนี้ก็อาจขึ้นกับปัจจัยอื่นๆ ด้วยเช่น ปริมาณทราฟฟิกในเน็ตเวิร์คและทรัพยากรบนเครื่องปลายทาง นอกจากนั้นมันยังเป็นเทคนิคที่ค่อนข้างช้าอีกด้วยถ้าคุณกำลังสแกนเน็ตเวิร์กที่ใช้งานไฟร์วอลล์ หรือเราท์เตอร์ที่มีการฟิลเตอร์กรองแพ็กเกจ จึงขอให้เตรียมใจไว้ด้วยกับผลลัพธ์ที่ไม่คาดคิดของ UDP scan

4.9.2 Trace Route Scan

โปรแกรม Trace Route ที่ทำงาน อยู่บน Unix นั้นใช้ในการส่ง UDP แพ็กเกจเพื่อค้นหาเส้นทาง โดยส่ง UDP ออกไปด้วยค่า TTL = 1 และเพิ่มไปเรื่อยๆจนกว่า UDP นั้นจะไปถึงปลายทางและปลายทางตอบกลับมาด้วย UDP Port Unreachable ลักษณะที่สำคัญของการแพ็กเกจที่ใช้เพื่อการ Trace route คือการที่ TTL จะเพิ่มขึ้นเรื่อยๆซึ่งการดัดแปลงคุณสมบัติของการ Trace Route เพียงเล็กน้อยก็จะสามารถนำไปใช้ในการสแกน UDP พอร์ตได้เช่นกัน

เนื่องจากแพ็กเกจของ Trace route เป็น UDP ดังนั้นการตอบสนองของโฮสต์ที่มีต่อ UDP ก็จะเหมือนกันไม่ว่าจะเป็นการ Trace route หรือการสแกนธรรมดา แต่โดยปกติแล้วการ Trace route จะเลือกส่ง UDP ไปยังพอร์ตที่ไม่น่าจะมีผู้ใช้งานปกติซึ่งจะอยู่ในช่วงของพอร์ตหมายเลข 33000-34999 แต่หากแฮกเกอร์ส่งแพ็กเกจสำหรับ Trace route ไปยังพอร์ตปกติก็จะสามารถ Trace route และสแกนพอร์ตได้ในตัว

4.10 Denial of Service Attack

Denial of Service Attack (DoS) เป็นรูปแบบการโจมตีที่มีจุดประสงค์เพื่อการทำให้เครือข่ายหรือโฮมเพจรวมทั้งเซิร์ฟเวอร์บนเครือข่ายปฏิเสธการให้บริการ หรือไม่สามารถที่จะดำเนินการต่อไปได้ ลักษณะการโจมตีแบบนี้เป็นการทำให้เครือข่ายเต็มไปด้วย Traffic ขนาดมหาศาลซึ่งคล้ายกับการที่มีบุคคลเป็นจำนวนมาก ต่างก็พร้อมใจกันได้โทรศัพท์ติดต่อเข้ามาที่โทรศัพท์หมายเลขเดียวกัน ทำให้สายโทรศัพท์ไม่ว่างตลอดเวลา จุดประสงค์ของการโจมตีแบบ DoS นี้อาจเกิดขึ้นจากความสนุก ความที่ต้องการลองวิชา หรือเจตนาในเชิงแข่งขันทางธุรกิจ รวมทั้งเจตนามุ่งร้ายอื่น ๆ การโจมตีในลักษณะนี้ไม่เพียงแต่ทำให้เครือข่ายติดขัดเนื่องจากปริมาณ Traffic ที่เพิ่มขึ้นเท่านั้น แต่ยังมีการส่งแพ็กเกจพิเศษที่ถูกจัดทำขึ้นเพื่อให้โปรโตคอลการทำงานของเครื่องคอมพิวเตอร์เป้าหมายเกิดความสับสน หรือทำให้แอพพลิเคชัน รวมทั้งการให้บริการต่าง ๆ บนเครื่องเป้าหมายหยุดทำงาน หรือไม่สามารถทำงานต่อไปได้

หลักการโจมตีที่นับว่าเป็นพื้นฐานและเก่าแก่ที่สุด ได้แก่ การโจมตีโดยอาศัยการส่งข่าวสารภายใต้คำสั่ง Ping หรือการใช้ซอฟต์แวร์ที่ทำให้เกิดการ Ping รวมทั้งส่ง SYN ไปที่เครื่องคอมพิวเตอร์ปลายทางในปริมาณมหาศาล ซึ่งปกติท่านสามารถพิสูจน์ทราบมันได้ ด้วยการใช้

ไฟร์วอลล์หรือซอฟต์แวร์ประเภทตรวจสอบการบุกรุก (Intruder Detection System - IDS) รายงานสภาวะการบุกรุกของการโจมตีดังกล่าว

4.10.1 Anomalous Packet

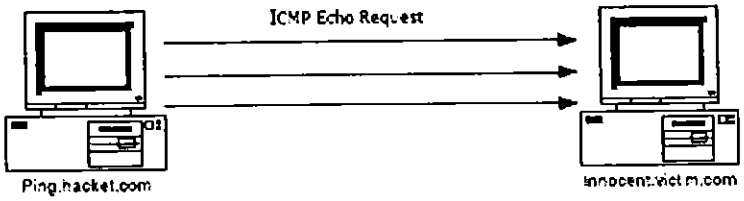
Anomalous Packet หมายถึงแพ็กเกจประหลาดที่ไม่มีโอกาสเกิดขึ้นในสภาวะปกติได้โดยอย่างสิ้นเชิง แพ็กเกจประเภทนี้เป็นการจงใจเปลี่ยนข้อมูลสำคัญที่ใช้ควบคุมการสื่อสารข้อมูลให้ผิดปกติ ผิดธรรมชาติของการสื่อสารข้อมูลธรรมดาไม่ว่า IP, UDP หรือ TCP ต่างก็มีข้อกำหนดอยู่ในโปรโตคอลของตนเองโดยในแต่ละโปรโตคอลย่อมมีค่าในเฮดเดอร์ที่จะใช้เป็นกลไกการควบคุมการสื่อสารข้อมูล โดยในสภาวะของการสื่อสารข้อมูลตามปกติแล้วค่าในเฮดเดอร์เหล่านี้จะมีค่าที่มีขอบเขตและคาดหมายได้ Anomalous Packet เหล่านี้จะเป็นแพ็กเกจ ที่ถูกดัดแปลงด้วยเทคนิคของการควบคุมเน็ตเวิร์คเลเยอร์โดยตรงแบบไม่ผ่านโปรโตคอล ซึ่งเป็นช่องทางให้แฮกเกอร์ทั้งหลายได้ใช้โจมตีเป้าหมายให้ทำงานผิดพลาดไปเพราะต้องจัดการกับแพ็กเกจที่ไม่ได้ระบุอยู่ในโปรโตคอลหรือมิเช่นนั้นก็ใช้เพื่ออำพรางตนเองในการสำรวจเป้าหมาย เป็นต้น สำหรับการดัดแปลงของแต่ละโปรโตคอลสามารถแสดงได้พอสังเขปดังนี้

- IP : IP Length, Fragment Offset, Fragment Flag, TCP Option
- UDP : UDP Length, Socket (Address & Port)
- TCP : TCP Flag, Socket (Address & Port)

Anomalous Packet ที่นิยมนำมาใช้งานกันมากที่สุดเห็นจะเป็น TCP โดยเฉพาะการปรับเปลี่ยนแฟลกไปต่างๆนานาสารพัดเงื่อนไขเท่าที่จะสามารถเปลี่ยนได้

4.10.2 Ping Flood Attack

Ping Flood เป็นการโจมตีที่ใช้กันในยุคแรกๆของ DoS เป็นการโจมตีที่มิได้อาศัยเทคนิคลึกลับซับซ้อนแต่อย่างใด อาศัยปริมาณแพ็กเกจมากๆเพียงอย่างเดียว แต่ถึงกระนั้นก็ตามจากการโจมตีวิธีนี้ก็สร้างความเสียหายได้ไม่น้อย



รูปที่ 4.13 แสดงการโจมตีแบบ Ping Flood Attack

หลักการโจมตีของ Ping Flood คือการส่ง ICMP Echo Request (แบบเดียวกับที่ได้จากคำสั่ง Ping) ปริมาณมากๆไปยังเป้าหมายอย่างรวดเร็ว ทำให้โฮสต์ที่ถูกโจมตีจะต้องคอยตอบ ICMP Echo Reply ตลอดเวลาจนแทบจะไม่มีเวลาทำงานอื่น ความรุนแรงของการโจมตีจะมากหรือน้อยแปรผันตามความเร็วของการส่ง ICMP แพ็กเกจเป็นหลัก

นอกจากการสร้างความเสียหายแก่เครื่องคอมพิวเตอร์เป้าหมายแล้ว Ping Flood ยังสร้างความเสียหายให้แก่เน็ตเวิร์คที่เครื่องคอมพิวเตอร์เป้าหมายนั้นอยู่ด้วยโดยความเสียหายจะมีขอบเขตมากน้อยเพียงใดขึ้นอยู่กับลักษณะการออกแบบของเน็ตเวิร์คนั้นๆถึงแม้ว่าแพ็กเกจที่ทำการส่งไปยังเซิร์ฟเวอร์เป้าหมายนั้นจะมีหมายเลข IP เป็นของเป้าหมายเครื่องเดียว แต่ในความเป็นจริงแล้วที่ชั้นเลเยอร์ต่ำลงไป เช่น Ethernet ต่างก็ใช้งานร่วมกันกับเครื่องโฮสต์อื่นๆในเน็ตเวิร์ค การที่ข้อมูลจะเดินทางจากที่ใดและจะไปที่ใดจะต้องผ่านเน็ตเวิร์คที่ใช้งานร่วมกันนี้ ดังนั้นเมื่อแพ็กเกจจำนวนมากถูกส่งมายังเครื่องเป้าหมาย นอกจากจะทำให้เครื่องเป้าหมายเสียหาย แล้วเน็ตเวิร์คอันเป็นทางผ่านก็จะเต็มไปด้วยแพ็กเกจนี้เช่นกัน ลักษณะเช่นนี้จะเกิดมากบนเน็ตเวิร์คที่มีการใช้การสื่อสารเลเยอร์ต่ำร่วมกันเช่น 10Base-5 หรือ10Base-T ที่ใช้ Hub เป็นตัวกระจายสัญญาณ

ข้อสังเกตสำหรับการโจมตีประเภทนี้คือ ปรากฏแพ็กเกจ ICMP Echo Request และICMP Echo Reply ปริมาณมหาศาลซึ่งมีการรับส่งกันระหว่างเครื่องเป้าหมายที่ถูกโจมตีกับเครื่องอื่นๆ ซึ่งอาจมีหรือไม่มีในอินเทอร์เน็ตก็ได้ เนื่องจากกระบวนการสำคัญอย่างหนึ่งของการโจมตีลักษณะนี้คือ ผู้โจมตีต้องปลอมหมายเลขไอพี (IP Spoofing) เสมอ เพื่อป้องกันไม่ให้แพ็กเกจICMP Echo Reply ถูกส่งกลับมายังเครื่องตัวเอง อันจะทำให้ผู้โจมตีได้รับผลจากการโจมตีด้วย และการปลอมไอพียังเป็นหลักประกันได้ว่าไม่สามารถติดตามได้ว่าผู้ใดเป็นผู้โจมตี รูปแบบแพ็กเกจที่เกิดขึ้นจากการโจมตีมีลักษณะดังนี้

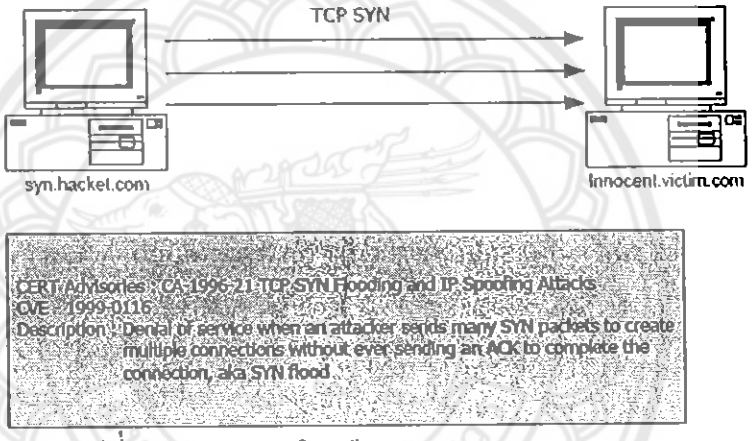
```
14:49:43.217137 62.51.12.23     > 10.1.1.10      : icmp: echo request
14:49:43.217175 10.1.1.10       > 62.51.12.23    : icmp: echo reply
14:49:43.217195 62.51.12.23     > 10.1.1.10      : icmp: echo request
14:49:43.217219 10.1.1.10       > 62.51.12.23    : icmp: echo reply
14:49:43.217245 96.141.106.124  > 10.1.1.10      : icmp: echo request
14:49:43.217279 10.1.1.10       > 96.141.10.124  : icmp: echo reply
14:49:43.219017 172.19.251.18   > 10.1.1.10      : icmp: net 162.75.127.79 unreachable
14:49:43.237136 75.126.62.65    > 10.1.1.10      : icmp: echo request
14:49:43.237169 10.1.1.10       > 75.126.62.65   : icmp: echo reply
14:49:43.237193 75.126.62.65    > 10.1.1.10      : icmp: echo request
14:49:43.237216 10.1.1.10       > 75.126.62.65   : icmp: echo reply
14:49:43.237240 218.155.179.58  > 10.1.1.10      : icmp: echo request
14:49:43.237272 10.1.1.10       > 218.155.17.58  : icmp: echo reply
```

รูปที่ 4.14 แพ็กเกจของ Ping Flood Attack

4.10.3 SYN Flood Attack

ถ้าจะนับว่าการโจมตีของ Ping Flood เป็นการทดลองกำลังกันในระดับ IP (ด้วย ICMP) SYN Flood ก็นับเป็นการลองกำลังในระดับ TCP สิ่งที่เป็นคุณสมบัติสำคัญของ TCP คือการเชื่อมต่อที่มีเสถียรภาพ โดยมีขั้นตอนการสถาปนาการเชื่อมต่อ และยุติการเชื่อมต่อ แต่ข้อดีเหล่านี้กลับถูกนำไปใช้เป็นอาวุธสำคัญในการโจมตี

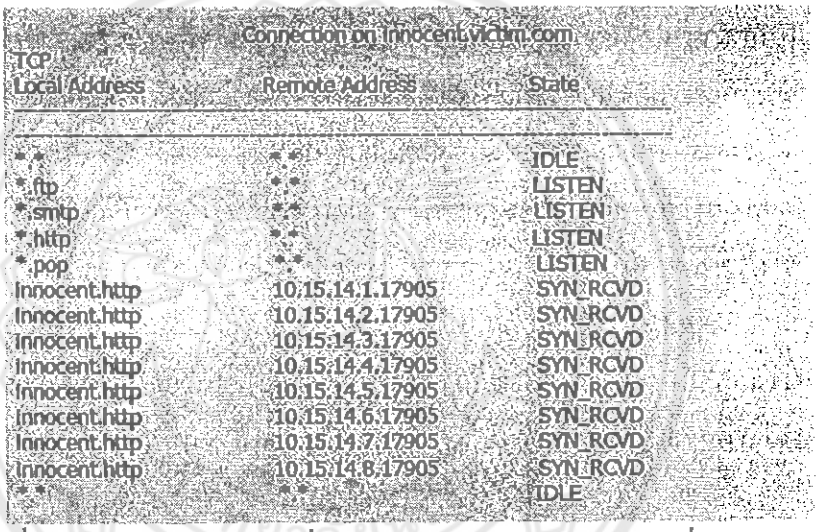
ด้วยลักษณะในการเริ่มต้นเชื่อมต่อของ TCP นั้นจะเป็นการตรวจสอบซึ่งกันและกันทั้ง 2 ฝ่ายที่เรียกว่า 3-ways handshake โดยเริ่มต้นจากเครื่องที่ต้องการติดต่อ ส่งสัญญาณ SYN มายังเซิร์ฟเวอร์ หลังจากนั้นการเริ่มต้นการเชื่อมต่อตามโปรโตคอลก็จะดำเนินต่อไป



รูปที่ 4.15 แสดงการโจมตีแบบ SYN Flood Attack

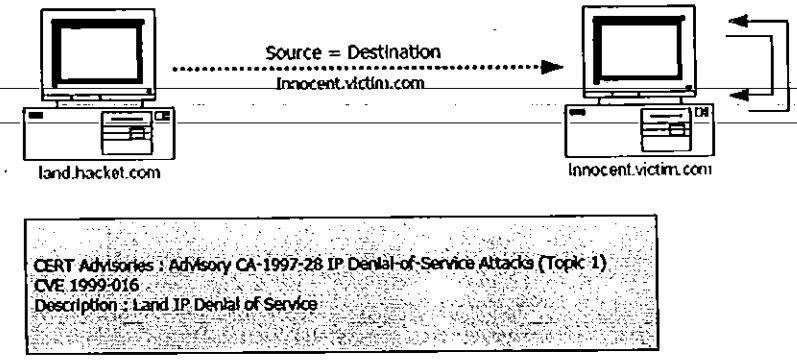
สิ่งที่ยากที่สุดในการทำงานของ TCP ก็คือการพยายามทำให้ทั้งสองฝ่ายสามารถสื่อสารกันได้อย่างถูกต้องและมีเสถียรภาพ สิ่งที่ต้องพิจารณาก็คือ แพ็กเกจของการเชื่อมต่อ จะเริ่มต้นด้วยการได้รับสัญญาณ SYN และจะต้องตอบ SYN ACK กลับไปให้แก่ผู้ที่ขอ จากนั้นต้องรอการตอบรับอีกครั้งหนึ่งของไคลเอนต์จึงจะจบกระบวนการ ดังนั้น เพื่อที่จะให้การเชื่อมต่อสามารถดำเนินไปได้อย่างต่อเนื่องโปรแกรมที่จัดการ 3-ways handshake ของเซิร์ฟเวอร์จะต้องจัดสรรหน่วยความจำจำนวนหนึ่งเพื่อรองรับการเชื่อมต่อแต่ละเซสชั่นจนกว่าการทำ 3-ways handshake จะสิ้นสุดลง โดยที่เซิร์ฟเวอร์เองก็ไม่มีทางรู้ได้เลยว่าไคลเอนต์จะ ACK จะวนกลับมาเพื่อจบการเชื่อมต่อนั้น คือของ 3-ways handshake เมื่อไร ซึ่งแน่นอนว่าในการบริหารหน่วยความจำและการสื่อสารข้อมูลจะต้องสร้างความสมดุลระหว่างเสถียรภาพของเซิร์ฟเวอร์ และประสิทธิภาพของการสื่อสาร เซิร์ฟเวอร์เองก็จะมีเวลาค่าหนึ่งที่จะรอให้ได้รับสัญญาณ ACK ตอบกลับมา หากถึงเวลาที่กำหนดแล้วไม่มีแพ็กเกจ ACK กลับมาเซิร์ฟเวอร์จะต้องยุติการรอนั้น และคืนหน่วยความจำให้แก่ระบบปฏิบัติการ

เทคนิคที่อาศัยการ SYN นี้จะส่งผลกระทบโดยตรงกับเซิร์ฟเวอร์ หากระบบปฏิบัติการของเซิร์ฟเวอร์เป้าหมายจัดการหน่วยความจำได้ไม่มีประสิทธิภาพพอ มันก็อาจจะหยุดทำงานได้ทันที จนถึงปัจจุบัน SYN Flood ก็ยังเป็นการโจมตีที่ได้ผลอยู่และหาทางป้องกันได้ยาก เนื่องจากยากที่จะจำแนกลักษณะของแพ็กเกจที่ใช้โจมตีกับแพ็กเกจที่ขอเริ่มต้นการเชื่อมต่อทั่วไป โดยเฉพาะสำหรับเซิร์ฟเวอร์ที่มีฮิตเรตสูงๆ เช่น เว็บเซิร์ฟเวอร์ และถึงแม้ว่าระบบปฏิบัติการรุ่นใหม่จะได้มีการปรับปรุงในด้านการจัดการหน่วยความจำให้u3604 ีขึ้นแต่ก็คงจะช่วยได้เพียงเซิร์ฟเวอร์ไม่ถึงกับหยุดทำงานไปเลยเมื่อโดนโจมตีเท่านั้นแต่ก็ยากที่จะทำให้เซิร์ฟเวอร์ไม่ถึงกับหยุดทำงานไปเลยเมื่อโดนโจมตีเท่านั้น แต่ก็ยากที่จะทำให้เซิร์ฟเวอร์ยังคงรักษาความสามารถในการให้บริการได้เช่นสภาวะปกติ



รูปที่ 4.16 แสดงสถานการณ์เชื่อมต่อบน innocent.victim.com เมื่อถูกโจมตี

4.10.4 Land Attack



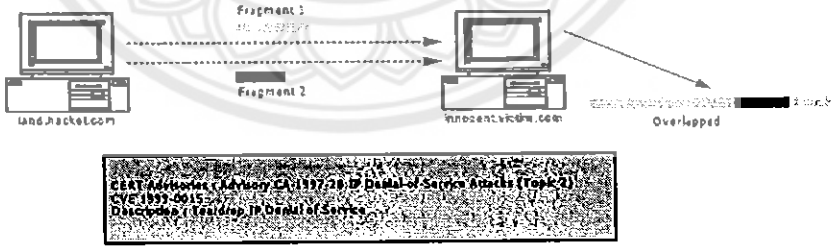
รูปที่ 4.17 แสดงการโจมตีแบบ Land Attack

ลักษณะการโจมตี

- หมายเลข IP ต้นทางเท่ากับ IP ปลายทาง
- หมายเลขพอร์ตต้นทางเท่ากับหมายเลขพอร์ตปลายทาง
- SYN Flag ถูก Set เสมือนขอเริ่มต้นการเชื่อมต่อ
- แพ็กเกจจะส่งไปยัง TCP พอร์ตที่เปิดอยู่

ปกติในข้อกำหนดของ TCP เมื่อมีการส่งสัญญาณ SYN มาเพื่อการเชื่อมต่อไปยังเป้าหมาย เป้าหมายก็จะตอบรับกลับไปยังผู้ส่งด้วย SYN ACK ตามหมายเลขพอร์ตและหมายเลข IP ต้นทาง แต่สำหรับการโจมตีแบบนี้หมายเลข IP ต้นทางและปลายทางอีกทั้งหมายเลขพอร์ตต้นทางและปลายทางจะถูกตั้งให้เป็นค่าเดียวกันดังนั้นการตอบกลับด้วย SYN ACK ก็จะตอบกลับไปที่ปลายทางเหมือนเดิม ซึ่งกรณีนี้ไม่มีกำหนดอยู่ในโปรโตคอลว่าควรจะทำอย่างไร โฮสต์จึงพยายามตอบสนองตามข้อกำหนดเท่าที่มีอยู่โดยการตอบกลับไปที่ IP Address และพอร์ตต้นทางที่ถูกบุกรุกมานั่นหมายถึงการตอบกลับเข้ามายังตัวเอง ซึ่งจะทำให้มีการตอบกลับไปมาของ TCP วนรอบอยู่ในตัวเองด้วยความเร็วสูง ทำให้คอมพิวเตอร์ต้องใช้ทรัพยากรที่มีอยู่ทั้งหมด ไม่ว่า CPU หน่วยความจำและอื่นๆเพื่อคอยจัดการกับ TCP ที่ตอบกลับไปมาดังกล่าวจนไม่อาจไปทำงานอื่นๆได้อีก จนดูเหมือนว่าเครื่องคอมพิวเตอร์หยุดทำงานและไม่ตอบสนองต่อการกระตุ้นใดๆแม้แต่คีย์บอร์ด ดังนั้นเหลือวิธีเดียวคือต้องรีเซ็ตเครื่องหรือปิดเครื่องจึงจะสามารถหยุดการวนรอบของ TCP ได้

4.10.5 Teardrop Attack



รูปที่ 4.18 แสดงการโจมตีแบบ Teardrop Attack

Teardrop เป็นการโจมตีโดยใช้ข้อบกพร่องของแฟรกเมนต์รีแอสเซมเบิล (การรวมแฟรกเมนต์แพ็กเกจหลายแพ็กเกจกลับมาเป็น IP ดาต้าแกรมเดียว) ของ IP เพื่อทำให้ระบบปฏิบัติการปลายทางของเป้าหมายทำงานผิดพลาด ไม่อยู่ในเงื่อนไขที่กำหนดไว้และหยุดทำงาน

ในการประกอบรวมแฟรกเมนต์แพ็กเกจกลับมาอยู่ใน 1 คาต้าแกรมนั้น IP จะ มีลักษณะการทำงานโดยพิจารณาจากข้อมูลของ 3 ฟิลด์คือ

1. Data length : ขนาดความยาวของข้อมูลในแพ็กเกจนั้น

2. Offset : ตำแหน่งของแพ็กเกจที่จะนำไปประกอบรวมกลับใน IP คาต้าแกรม

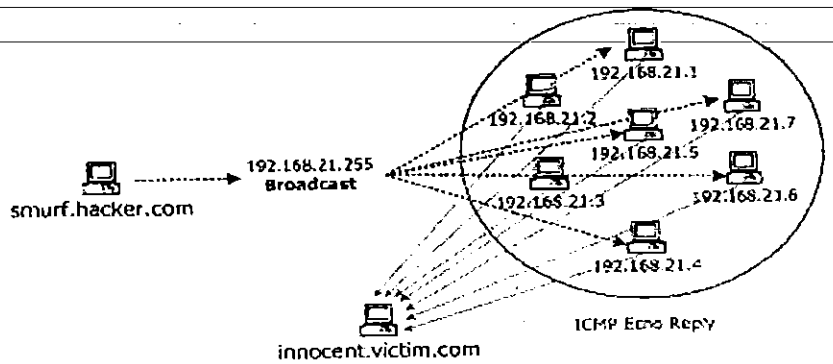
3. Flag : แฟลกซึ่งระบุว่าไม่มีแพ็กเกจต่อจากแพ็กเกจนี้อีก หมายถึงแพ็กเกจนี้เป็นส่วนที่อยู่สุดท้ายของคาต้าแกรม

โดยทั่วไปแล้ว IP แพ็กเกจที่ถูกแฟรกเมนต์มานั้นไม่จำเป็นจะต้องถึงปลายทางตามลำดับ ดังนั้นปลายทางผู้รับแพ็กเกจจึงต้องทำการรีแอสเซมเบิลโดยนำแพ็กเกจที่เข้ามาไปวางรอไว้เพื่อตรงตำแหน่งของ offset และรอจนกว่าทุกแพ็กเกจจะมาครบ จากนั้นจึงส่ง IP คาต้าแกรมที่สมบูรณ์นั้นไปยังเลเยอร์ถัดขึ้นไป ด้วยลักษณะของการทำงานเช่นนี้ มีช่องโหว่มากมายที่สามารถนำไปใช้ในการโจมตี

หากเป็นการรับแฟรกเมนต์แพ็กเกจตามปกติแล้ว IP ก็จะนำแฟรกเมนต์แต่ละแพ็กเกจที่ได้รับมาวางรอไว้ในหน่วยความจำตามตำแหน่งที่ระบุในฟิลด์ offset ตามลำดับการมาถึงของแพ็กเกจ จะกระทั่งทุกแฟรกเมนต์ได้มาถึงครบหมดจึงรวมกลับมาเป็นคาต้าแกรมที่สมบูรณ์ หรือหากว่าเกินเวลาที่กำหนดแล้วแฟรกเมนต์ยังเดินทางมาไม่ครบ IP ก็จะแจ้งข้อผิดพลาดกลับไปพร้อมทั้งยกเลิกการรับคาต้าแกรมนั้น

การโจมตีของ Teardrop นั้นจะใช้การเหลื่อมกันของแพ็กเกจในระหว่างที่มีการรวมแฟรก-เมนต์แพ็กเกจเข้าด้วยกัน แต่ว่าเนื่องจากในการแฟรกเมนต์ตามปกติแล้วแพ็กเกจจะถูกแบ่งออกเป็นส่วนย่อย แต่สามารถนำมารวมกันใหม่ได้พอดี และตำแหน่งจะถูกต้องสอดคล้องกันเสมอ แพ็กเกจที่ใช้ในการโจมตีของ Teardrop โดยจะเป็นแพ็กเกจที่ถูกสร้างขึ้นมาโดยเฉพาะมิได้ผ่านกลไกการแฟรกเมนต์ ตามปกติของ IP โดยมีการระบุ offset ที่เหลื่อมเข้าไปในแพ็กเกจอื่นๆ ซึ่งจะไม่มีโอกาสเกิดขึ้นได้เลยในการทำงานปกติ ดังนั้นหากระบบปฏิบัติการไม่สามารถจัดการเงื่อนไขไม่ปกติเช่นนี้ได้ดีเพียงพอก็จะหยุดทำงานลงได้

4.10.6 Smurf Attack

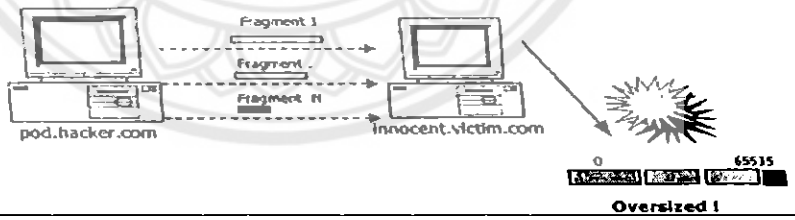


รูปที่ 4.19 แสดงการโจมตีแบบ Smurf Attack

Smurf เป็นการโจมตีที่มีรูปแบบที่ให้ผลกระทบที่ขยายตัวออกไปได้ในบริเวณวงกว้าง (amplication effect) เป็นการปรับปรุงเทคนิคการส่งแพ็กเกจจำนวนมากให้ฉลาดกว่าเดิม โดยการขยายการโจมตีทำให้แฮกเกอร์มีเครื่องทุ่นแรง และเปิดโอกาสให้ผู้ที่มีแบนด์วิดธ์ต่ำสามารถทำการ flood เป้าหมายได้รุนแรง จากคุณสมบัติของการบรอดคาสต์ การส่งแพ็กเกจใดไปยังแอดเดรสบรอดคาสต์จะทำให้ทุกโฮสต์ในเน็ตเวิร์คได้รับแพ็กเกจนั้นอย่างทั่วถึง ดังนั้นหากมีโฮสต์ใดที่ส่ง ICMP Echo Request มายัง บรอดคาสต์ ก็จะทำให้ทุกๆโฮสต์ทั้งหมดที่อยู่ในเน็ตเวิร์คนั้นได้รับ ICMP Echo พร้อมกัน และด้วยข้อกำหนดของ ICMP เมื่อโฮสต์ได้รับ ICMP Echo Request จะต้องตอบกลับด้วย ICMP Echo Reply กลับไปยังผู้ส่งเสมอ ซึ่งเป็นไปได้ว่าหากมีการส่ง ICMP Echo ไปยังบรอดคาสต์เพียงครั้งเดียวก็จะได้รับ ICMP Echo Reply ตอบกลับเท่ากับจำนวนเครื่องที่อยู่ในเน็ตเวิร์คนั้นเลย ปรากฏการณ์นี้เรียกว่าการขยายสัญญาณ (Amplification) โดยการบรอดคาสต์อัตราการขยายก็จะขึ้นอยู่กับปริมาณโฮสต์ที่อยู่ในเน็ตเวิร์คขณะนั้น โดยวิธีการโจมตีนี้จะทำได้ต้องอาศัยเทคนิคการปลอม IP Address ด้วย โดยให้ IP Address ต้นทางของ ICMP Echo Request ที่ส่งไปนั้นเป็น IP Address ของเป้าหมาย

การโจมตีชนิดนี้ เป้าหมายอาจไม่จำเป็นต้องเป็นโฮสต์ใดโฮสต์หนึ่ง โดยการดัดแปลงการโจมตีให้เกิดผลเสียหายต่อเน็ตเวิร์คได้โดยการทำให้เน็ตเวิร์คท่วมไปด้วยแพ็กเกจ เพียงแฮกเกอร์ส่งแพ็กเกจที่ใช้การโจมตีอย่างต่อเนื่อง และอัตราสูงก็จะทำให้เน็ตเวิร์คนั้นเต็มไปด้วยแพ็กเกจของ ICMP Echo Reply ซึ่งทำให้แพ็กเกจอื่น สำหรับใช้งานตามปกติไม่สามารถออกไปได้

4.10.7 Ping Of Death Attack



CERT Advisories : Advisory-CA-1996-26 Denial-Of-Service-Attack via-Ping
CVE: 1999-0128
Description : Oversized ICMP ping packets can result in a denial of service, aka *Ping o' Death*

รูปที่ 4.20 Ping Of Death Attack

การโจมตีแบบนี้เป็นอีกวิธีหนึ่งที่ใช้ข้อบกพร้องของแฟรกเมนต์มาเป็นช่องทางในการโจมตี ตามปกติแล้ว IP ดาต้าแกรมจะมีขนาดสูงสุดไม่เกิน 65535 ไบต์ การส่ง IP ดาต้าแกรม

ภายในแพ็กเกจเดียวทำอย่างไรก็ไม่เกินนี้ เพราะว่าจะถูกจำกัดด้วยขนาดของฟิลด์ซึ่งมีขนาดเพียง ไบต์เท่านั้น แต่การที่ขนาดของฟิลด์ใน IP Address ได้ถูกจำกัดไว้โดย IP ยังมีการแฟรกเมนต์ที่สามารถนำหลายๆแพ็กเกจมาต่อรวมกันเป็นดาต้าแกรมเดียว ซึ่งนับว่าเป็นช่องทางที่สามารถทลายข้อจำกัดของขนาดของดาต้าแกรมอันเนื่องมาจากขนาดของฟิลด์ได้ เพราะว่าเนื่องจากผลรวมขนาดของดาต้าแกรมนั้น จะต้องเป็นผลรวมของขนาดแฟรกเมนต์ทั้งหมดรวมกัน และขนาดของแต่ละแฟรกเมนต์ก็สามารถมีได้ถึง 64 K ดังนั้นเมื่อเรานำแฟรกเมนต์หลายๆ แฟรกเมนต์นั้นมารวมกันเป็นดาต้าแกรมเดียวก็มีโอกาสที่จะทำให้ขนาดของดาต้าแกรมทั้งหมดสูงกว่า 65535 ไบต์ได้

การโจมตีของ Ping Of Death เห็นข้อบกพร่องในส่วนนี้และพุ่งเป้าไปยังระบบปฏิบัติการที่ไม่ได้จัดการแฟรกเมนต์อย่างรัดกุมเพียงพอ ด้วยตรรกพื้นฐานว่าหากมีการจัดสรรหน่อยความจำไว้สูงสุด 64 K เพื่อรองรับดาต้าแกรม 1 ดาต้าแกรม และถ้าหากระบบปฏิบัติการไม่มีกลไกการตรวจสอบที่รอบคอบแล้ว การรีแอสเซมเบิลของแฟรกเมนต์นั่นเอง โดยหวังว่าเมื่อทุกแฟรกเมนต์ถูกนำไปใส่ในหน่อยความจำตามตำแหน่งที่ระบุในตัวแฟรกเมนต์นั่นเอง โดยที่เราหวังจะว่าเมื่อทุกแฟรกเมนต์ถูกนำไปใส่ในหน่อยความจำครบถ้วนแล้วจะได้ดาต้าแกรมที่สมบูรณ์ออกมาโดยปริยายอัตโนมัติ ซึ่งเป็นกลไกที่เรียบง่ายและมีประสิทธิภาพพอสมควรกับการสื่อสารตามปกติ ในกรณีของ Ping Of Death ก็คือการพยายามหลอกล่อกับกระบวนการนี้โดยการส่งแฟรกเมนต์ของ ICMP Echo Request ที่แฟรกเมนต์ไปยังเป้าหมายเสมือนการใช้คำสั่ง Ping แต่ตั้งใจทำให้ผลรวมของแฟรกเมนต์นั้นเกินกว่าขนาด 64 K ซึ่งเป็นขนาดของระบบปฏิบัติการจัดสรรหน่วยความจำเอาไว้ สิ่งที่จะเกิดขึ้นก็คือหากระบบปฏิบัติการทำการรีแอสเซมเบิลโดยไม่มีการตรวจสอบ และไม่เฉลียวใจว่าจะมีแฟรกเมนต์ใดที่ขนาดเกินกว่าหน่วยความจำแล้วนำข้อมูลในแฟรกเมนต์ไปใส่ในหน่วยความจำตามปกติ ข้อมูลก็จะล้นออกนอกหน่วยความจำที่จัดสรรไว้ (Overflow)

แน่นอนว่าด้วยกลไกการแฟรกเมนต์ตามปกติกรณีเช่นนี้ย่อยไม่สามารถเกิดขึ้นได้ แต่ทว่าแฮกเกอร์ก็สามารถสร้างแพ็กเกจปลอมขึ้นมาตบตาให้ดูเหมือนเป็นการแฟรกเมนต์ตามปกติ แต่ได้ซ่อนข้อบกพร่องนี้เอาไว้ หากระบบปฏิบัติการของเป้าหมายนั้นไม่มีกลไกในการควบคุมขนาดและความต่อเนื่องของแพ็กเกจที่รัดกุมเพียงพอก็จะทำให้แฮกเกอร์โจมตีได้สำเร็จโดยง่าย

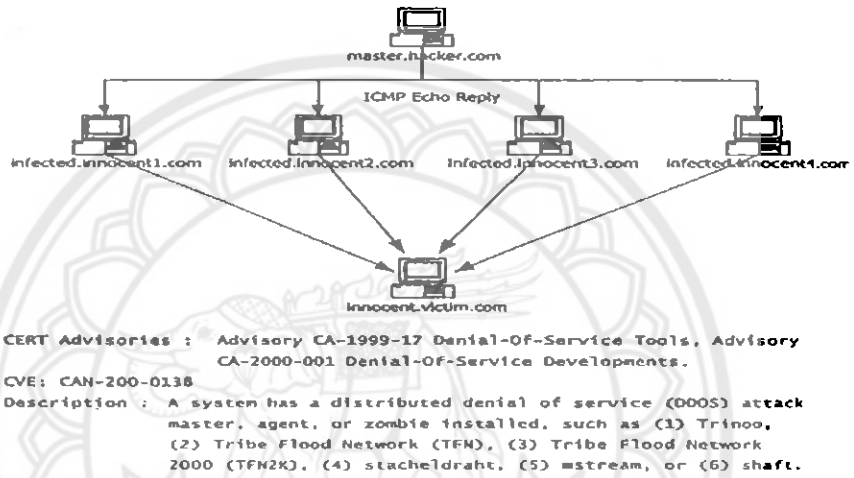
```
10.15.14.1 > innocent.victim.com: icmp: echo request (frag 56980:1480@0+)
10.15.14.1 > innocent.victim.com: (frag 56980:1480@1480+)
10.15.14.1 > innocent.victim.com: (frag 56980:1480@2960+)
10.15.14.1 > innocent.victim.com: (frag 56980:1480@4440+)
10.15.14.1 > innocent.victim.com: (frag 56980:1480@5920+)
10.15.14.1 > innocent.victim.com: (frag 56980:1480@59200+)
10.15.14.1 > innocent.victim.com: (frag 56980:1480@60680+)
10.15.14.1 > innocent.victim.com: (frag 56980:1480@62160+)
10.15.14.1 > innocent.victim.com: (frag 56980:1480@63640+)
10.15.14.1 > innocent.victim.com: (frag 56980:1480@65120)
```

รูปที่ 4.21 แพ็กเกจของการ Ping of Death Attack

แพ็กเกจที่ใช้ในการโจมตีจะทำการส่งแฟรกเมนต์ของ ICMP แพ็กเกจอย่างต่อเนื่องไปยังเป้าหมาย โดยแฟรกเมนต์ที่ส่งไปนั้นเมื่อนำไปประกอบรวมกันที่ปลายทางจะได้ IP คาต้าแกรมที่มีความยาวมากกว่า 65535 ไบต์ ซึ่งเกินกว่าข้อกำหนดของ IP ดังนั้นหากระบบปฏิบัติการของเครื่องเป้าหมายมีข้อบกพร่องอยู่และไม่ได้รับการแก้ไขอย่างถูกต้องก็จะหยุดทำงานลงทันที

4.10.8 Tribe Flood Network

Tribe Flood Network (TFN) เป็นการเปลี่ยนแนวทางการโจมตีใหม่ให้รุนแรงและซับซ้อนกว่าเดิม โดยการโจมตีเป้าหมายพร้อมๆกันด้วยหลายๆโฮสต์



รูปที่ 4.22 Tribe Flood Network

TFN เป็นวิธีการโจมตีที่ใช้ ICMP เป็นคำสั่งสำหรับสั่งงานโฮสต์หลายๆโฮสต์ ให้ทำการโจมตีเป้าหมายพร้อมกัน มีวิวัฒนาการที่น่าสนใจการใช้โฮสต์จำนวนมากร่วมมือกัน และก็เป็นรุ่นแรกๆของการโจมตีต่อต้านการทำงานแบบกระจาย DDoS (Distributed Denial of Service) ในเวลาต่อมา

การโจมตีแบบเบื้องต้นในยุคแรกๆที่อาศัยการส่งแพ็กเกจจำนวนมากไปยังเป้าหมายเพื่อให้เซิร์ฟเวอร์เป้าหมายทำงานช้าลงจนหยุดทำงาน หรือไม่ก็พยายามทำให้เน็ตเวิร์คของเป้าหมายท่วมไปด้วยแพ็กเกจที่เกิดจากการโจมตี จนกระทั่งข้อมูลที่สื่อสารตามปกติไม่สามารถสอดแทรกเข้าไปยังเซิร์ฟเวอร์ได้

อย่างไรก็ตามเทคนิคดังกล่าวก็ใช้ได้ผลเพียงระยะเริ่มแรกเท่านั้น ในระยะหลังการโจมตีดังกล่าวไม่ค่อยสร้างความเสียหายได้มากนัก

1. การจัดการ TCP Stack ของระบบปฏิบัติการต่างๆได้ถูกปรับปรุงให้มีประสิทธิภาพและเสถียรภาพมากยิ่งขึ้น

2. แบนด์วิดธ์ของเน็ตเวิร์คที่ไปยังเซิร์ฟเวอร์ เป้าหมาย ซึ่งมีขนาดใหญ่ขึ้นกว่าเดิมมากๆ เนื่องจากมีการปรับปรุงพัฒนาของฮาร์ดแวร์และอุปกรณ์เน็ตเวิร์ค

เพื่อแก้ปัญหาดังกล่าว ดังนั้นจึงมีผู้พัฒนาเทคนิคของ TFN ขึ้นมา เพื่อสำหรับเอาไว้ใช้ในการโจมตี โดยมีจุดประสงค์หลักคือ จะโจมตีโดยใช้เทคนิคของการทำเน็ตเวิร์คให้เต็ม (Flood) เช่นเดิม แต่เปลี่ยนจากการใช้เครื่องของแฮกเกอร์เพียงเครื่องเดียวเป็นผู้ส่งแพ็กเกจหลัก เป็นการใช้เครื่องจำนวนมากส่งมาโจมตี เป้าหมายเดียวกันพร้อมกัน ซึ่งจะทำให้ความรุนแรงของการโจมตีนั้นเพิ่มมากเป็นทวีคูณเท่ากับจำนวนเครื่องของผู้ร่วมโจมตีนั่นเอง

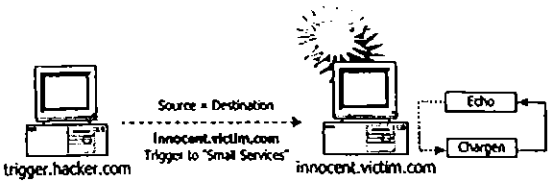
```
TFNmaster.hacker.com > infected.innocent1.com: icmp: echo reply
TFNmaster.hacker.com > infected.innocent1.com: icmp: echo reply
TFNmaster.hacker.com > infected.innocent1.com: icmp: echo reply
TFNmaster.hacker.com > infected.innocent2.com: icmp: echo reply
TFNmaster.hacker.com > infected.innocent2.com: icmp: echo reply
TFNmaster.hacker.com > infected.innocent2.com: icmp: echo reply
TFNmaster.hacker.com > infected.innocent3.com: icmp: echo reply
TFNmaster.hacker.com > infected.innocent3.com: icmp: echo reply
TFNmaster.hacker.com > infected.innocent3.com: icmp: echo reply
TFNmaster.hacker.com > infected.innocent4.com: icmp: echo reply
```

รูปที่ 4.23 แพ็กเกจของ Tribe Flood Network

การที่ TFN สร้างความเสียหายได้มากกว่าการโจมตีแบบ Flood ทั่วๆไปก็เพราะว่านอกจากการที่มีเครื่องจำนวนมากโจมตีพร้อมๆกันแล้ว การมี TFN Daemon เป็นการเปิดช่องทางให้แฮกเกอร์ที่เป็น TFN Master ซึ่งมีแบบวิดธ์ต่ำสามารถบังการ TFN Daemon ซึ่งจะอยู่ใน LAN Segment เดียวกับเป้าหมายและมีแบนด์วิดธ์สูงมากๆ ทำการโจมตีแทนตนเองได้ แบนด์วิดธ์จึงไม่เป็นอุปสรรคต่อการโจมตีแต่อย่างใด

จากตัวอย่างของแพ็กเกจในภาพที่ 4.22 จะปรากฏแพ็กเกจ ICMP Echo Reply ที่พบว่าทาง TFN Master ใช้สื่อสารกับ TFN Daemon เพื่อสั่งการให้ Daemon ทำการโจมตีเป้าหมายคำสั่งของการทำงานจะแฝงอยู่ในแพ็กเกจนั้นซึ่งหากโฮสต์อื่นที่มิใช่ TFN Daemon เรา ก็จะเห็นเป็นเพียงแค่ ICMP Echo Reply ธรรมดา แต่สำหรับ Daemon แล้วจะสามารถอ่านข้อมูลที่แฝงอยู่ใน ICMP นี้ได้

4.10.9 Diagnostic Port Attack



CERT Advisories : Advisory CA-1996-01-UDP Port Denial-of-Service Attack
CVE: CAN-1999-0103
Description : Echo and chargen, or other combinations of UDP services, can be used in tandem to flood the server, a.k.a. UDP bomb or UDP packet storm.

รูปที่ 4.24 Diagnostic Port Attack

ในตอนออกแบบ TCP/IP สมัยแรกนั้นได้มีการใส่คุณสมบัติและบริการในหลายอย่างใน TCP/IP เพื่อใช้ในการตรวจสอบสถานการณ์เชื่อมต่อ , บริการตรวจสอบเวลา , บริการเหล่านี้เรียกว่า " Small Service" ซึ่งแต่ละบริการมีรายละเอียดดังนี้

ตารางที่ 4.4 ชื่อบริการ หมายเลขพอร์ต และ การให้บริการ

ชื่อบริการ	หมายเลขพอร์ต	รายละเอียดการให้บริการ
Echo	7/UDP 7/TCP	เซิร์ฟเวอร์จะทำการตอบกลับด้วยข้อมูลเดียวกับที่ได้รับมาจากไคล์เอนต์
Discard	9/UDP 9/TCP	เซิร์ฟเวอร์จะไม่ตอบรับข้อมูลใดๆที่ส่งมาจากไคล์เอนต์
Daytime	13/UDP 13/TCP	เซิร์ฟเวอร์จะตอบเวลาและวันที่ กลับไปยังไคล์เอนต์ในรูปแบบที่สามารถอ่านได้เข้าใจ
Chargen	19/UDP	เซิร์ฟเวอร์จะตอบกลับไปยังไคล์เอนท์ด้วย ASCII Character อย่างต่อเนื่องจนกว่าไคล์เอนต์จะยุติการติดต่อ
Time	34/UDP 34/TCP	เซิร์ฟเวอร์จะตอบค่าเวลาในรูปแบบของ 32 บิต

บทที่ 5
ความรู้เกี่ยวกับระบบตรวจจับผู้บุกรุก

5.1 แนวความคิดพื้นฐานของระบบตรวจจับผู้บุกรุก

ข้อมูลที่เรามองเห็นได้โดยผ่านแอพพลิเคชั่นจะเป็นข้อมูลที่รับส่งกันตามปกติถูกต้องตามโพรโตคอลทุกประการ เพราะหากมีส่วนใดส่วนหนึ่งของข้อมูลนั้นเกิดผิดพลาด ไม่เป็นไปตามโพรโตคอล ไม่ว่าจะเป็นที่ชั้นใด ข้อมูลนั้นก็จะถูกดรอปทิ้งไป หรือข้อมูลบางอย่างที่ถึงแม้ว่าจะเป็นข้อมูลที่ถูกต้องตามปกติ แต่เป็นกลไกการรับส่งกันเองของโพรโตคอลเลเยอร์ล่างเพื่อให้การสื่อสารสมบูรณ์ ก็จะไม่ถูกส่งขึ้นมาให้ผู้ใช้ได้รับรู้ ซึ่งการทำงานลักษณะดังกล่าวในมุมมองของผู้ใช้งานแล้วน่าจะถูกต้องเพราะข้อมูลไม่เกี่ยวข้องก็ไม่น่าจะต้องส่งมาให้ผู้ใช้ได้พบเห็นแต่อย่างไร

ดังนั้นสิ่งที่ควรทำความเข้าใจเบื้องต้นก็คือ กิจกรรมต่างๆบนเน็ตเวิร์คที่ผู้ใช้เห็นและรับรู้นั้นเป็นเพียงส่วนที่ถูกกำหนดในแอพพลิเคชั่นว่าให้นำมาแสดงต่อผู้ใช้เท่านั้น สิ่งอื่นๆ ที่ผู้ใช้ไม่ได้เห็น มิได้หมายความว่าไม่มีกิจกรรมใดเกิดขึ้น ยังมีอะไรอีกมากมายที่เกิดขึ้นบนเน็ตเวิร์คหรือแม้กระทั่งบนเครื่องคอมพิวเตอร์ของเราโดยที่เราเองไม่รู้ตัวถึงแม้ว่าจะนั่งอยู่หน้าเครื่องตลอดเวลาก็ตาม และอีกประการหนึ่งคือการสื่อสารของคอมพิวเตอร์นั้นมีระบบรักษาความปลอดภัยต่ำมากโดยเฉพาะในระดับเน็ตเวิร์คเลเยอร์ หากใช้งานตามดีฟอลต์โดยไม่ได้มีการปรับแต่งเป็นพิเศษแล้วแทบจะไม่สามารถป้องกันตัวเองได้จากการติดต่อจากผู้อื่นเลย คือใครอยากจะส่งข้อมูลมาหาเครื่องเราก็สามารถทำได้ทันทีโดยที่เราหลีกเลี่ยงไม่ได้ สิ่งที่เราทำได้ก็คือเพียงแค่เลือกว่าจะนำข้อมูลนั้นไปใช้งานหรือไม่ หากไม่ใช่สิ่งที่ต้องการก็ดรอปทิ้งไป หากว่าใช่สิ่งที่ต้องการก็นำไปใช้งาน แต่อย่างน้อยที่สุดก็ต้องรับเข้ามาก่อนเสมอ แทนที่จะสามารถเลือกรับเฉพาะข้อมูลที่ต้องการเท่านั้น ซึ่งแค่จุดอ่อนนี้เพียงจุดเดียวก็สามารถนำไปใช้ในการโจมตีเพื่อให้ปิดบริการ เครื่องคอมพิวเตอร์ทั่วไปได้ทันที

หากเปรียบเทียบระบบหรือคอมพิวเตอร์ของเราเสมือนบ้าน ก็จะเป็นบ้านที่ไม่มีประตู ทุกคนสามารถเข้าออกได้อย่างเสรี ระบบปฏิบัติการและแอพพลิเคชั่นในเครื่องของเราเป็นเจ้าของบ้าน และข้อมูลที่สื่อสารกันไปมาบนเน็ตเวิร์คก็จะเป็นเหมือนคนเดินถนนทั่วไป เมื่อบ้านไม่มีประตูใครที่อยู่ข้างนอกอยากเข้ามาในบ้านก็เดินเข้ามาได้ตามปกติ เจ้าของบ้านนั้นมีหน้าที่เดินมาสอบถามทุกคนที่เข้ามาเพื่อให้ทราบว่าเป็นคนที่ต้องการติดต่อด้วยหรือไม่ หากไม่ใช่คนที่ติดต่อด้วยก็บอกให้เขากลับออกไป หากใช่ก็จะเชิญเข้ามาสนทนาด้วย เช่นหากแอพพลิเคชั่นของเรามีเฉพาะเมล์

เซิร์ฟเวอร์ เจ้าของบ้านก็จะยินดีต้อนรับเฉพาะบุรุษไปรษณีย์เท่านั้น หากใครที่ไม่ใช่ก็จะไม่สนทนาด้วย ไม่ว่าจะเป็นคนดีหรือไม่ดี หรือเป็นผู้ที่ทำหน้าที่อื่นที่อาจจะเข้ามาผิดบ้านก็ตาม อย่างไรก็ตามเจ้าของบ้านหลังนี้ไม่สามารถห้ามไม่ให้คนอื่นเดินเข้ามาได้หรือแม้กระทั่งไล่คนที่ไม่ต้องกาออกไปก็ทำไม่ได้ ไม่ว่าใครจะเข้ามาในบ้านเจ้าของบ้านก็ต้องคอยออกมาสอบถามทุกครั้งไป ถึงแม้ว่าจะเป็นคนเดิมๆ ที่พยายามจะเข้ามาซ้ำแล้วซ้ำอีกตลอดทั้งวัน

โดยส่วนใหญ่แอพพลิเคชั่นที่ให้บริการจะถูกออกแบบมา เพื่อให้บริการที่ดีที่สุดโดยไม่ได้ระวังอะไร เปรียบเสมือนคนที่มองโลกในแง่ดี ใครเข้ามาที่บ้านก็พยายามบริการอย่างดีที่สุด ดังนั้นหากใครจะกลั่นแกล้งเจ้าของบ้านก็จะทำได้ไม่ยากเย็น เช่นส่งคนเข้าไปในบ้านทีเดียวพร้อมกันหลายๆคนจนเจ้าของบ้านไม่มีเวลาไปทำงานอื่น (เทียบได้กับ Ping Flood) , ส่งคนเข้ามาในบ้านแต่พอเจ้าของบ้านถามอะไรก็ไม่ยอมตอบปล่อยให้เจ้าของบ้านรอ (SYN Flood) , ส่งคนหลายๆแบบมาหาเจ้าของบ้านเพื่อสืบว่าเจ้าของบ้านยินดีต้อนรับคนประเภทไหน (Port Scanning) เป็นต้น เมื่อแอพพลิเคชั่นไม่ได้ถูกออกแบบมาให้ระมัดระวังเรื่องความปลอดภัย แต่หากถูกก่อกวนมากๆจนไม่สามารถมือได้ก็จะหยุดทำงานในที่สุด

หากเปรียบระบบเป็นเสมือนบ้านแล้ว IDS ก็จะเป็นเสมือนยามรักษาการณ์ทำหน้าที่เป็นผู้ช่วยเจ้าของบ้าน เนื่องจากเจ้าของบ้านจะชำนาญเฉพาะเรื่องการบริการเท่านั้น กล่าวคือ ระบบตรวจจับผู้บุกรุก (Intrusion Detection System) จะช่วยเสริมจุดด้อยส่วนนี้ให้แข็งแรงมากยิ่งขึ้น โดยทำตัวเป็นยามที่ชำนาญในการวิเคราะห์คนผ่านเข้าออกโดยดูจากลักษณะของคนเหล่านั้น และรู้จักพฤติกรรมของอันธพาลหรือพวกก่อกวนเป็นอย่างดี หากใครมีพฤติกรรมต้องสงสัยก็จะรีบรายงานให้เจ้าของบ้านรู้ทันทีเมื่อได้รับรายงานแล้วจะดำเนินการอย่างไรต่อไปนั้นก็ต้องพิจารณาอีกที แต่อย่าง น้อยที่สุดก็เป็นการป้องกันภัย ล่วงหน้าสามารถรับรู้ถึงการพยายามบุกรุกหรือก่อกวนในทันทีที่เหตุการณ์เกิดขึ้น นับว่าระบบตรวจจับ ผู้ บุกรุกเป็นเครื่องมือสำคัญอย่างยิ่งที่จะรับมือกับการบุกรุกเข้ามา ของผู้ไม่หวังดี

5.2 ระบบตรวจจับผู้บุกรุก (IDS - Intrusion Detection System)

เป็นระบบจัดการความปลอดภัยสำหรับคอมพิวเตอร์และเครือข่ายที่พยายามจะตรวจหาและเตือนภัยเมื่อมีความพยายามในการบุกรุกเข้ามาในระบบ หรือ เครือข่าย เป็นอีกเครื่องมือหนึ่งที่ใช้กันอย่างมาก และมีความสำคัญอย่างยิ่งในปัจจุบัน ถึงแม้ว่าเครือข่ายอาจมีการป้องกันการบุกรุกอยู่แล้วโดยใช้ ไฟร์วอลล์ (Firewall) อย่างไรก็ตามไฟร์วอลล์ก็ยังไม่ใช่เครื่องมือที่จะป้องกันการบุกรุกได้โดยอัตโนมัติ จะต้องอาศัยผู้ที่บริหารกำหนดกฎให้เหมาะสมกับการใช้งาน และแม้จะมีกฎที่ดีแล้ว แต่ก็อาจไม่สามารถป้องกันการบุกรุกได้ การบริหารไฟร์วอลล์ที่ดีก็ควรจะมีการตรวจสอบย้อนหลัง และทดสอบการเจาะระบบเพื่อเป็นการทดสอบระบบอีกครั้ง ซึ่งตรงจุดนี้ ระบบตรวจจับผู้

บุกรุกจะช่วยได้มาก เพื่อตรวจสอบแพ็กเกจต่างๆที่ผ่านเข้ามา ถ้าตรวจพบการบุกรุก ผู้บริหารก็สามารถนำข้อมูลที่ได้ไปปรับปรุงกฎให้รัดกุมยิ่งขึ้น

ระบบตรวจจับผู้บุกรุก แบ่งเป็น 2 ประเภท คือ

1.ระบบตรวจจับผู้บุกรุกในโฮสต์ (Host-based Intrusion Detection System)

2.ระบบตรวจจับผู้บุกรุกเครือข่าย (Network Intrusion Detection System)

โดยระบบที่คณะผู้จัดทำได้ศึกษาในที่นี้คือ ระบบตรวจจับผู้บุกรุกเครือข่าย

5.3 ความหมายของการตรวจจับผู้บุกรุกทางเครือข่ายคอมพิวเตอร์

ระบบตรวจจับผู้บุกรุกเครือข่าย (Network Intrusion Detection System หรือ NIDS) เป็นแขนงหนึ่งของระบบตรวจจับผู้บุกรุก (Intrusion Detection System หรือ IDS) โดยเน้นไปทางการตรวจจับทางเครือข่ายคอมพิวเตอร์เป็นหลัก

โดยระบบนี้จะทำการตรวจสอบแพ็กเกจต่างๆ บนเครือข่ายเพื่อดูว่ามีข้อมูลที่ผิดปกติ หรือว่ามีพฤติกรรมที่น่าสงสัยหรือไม่ ซึ่งก็คือการพยายามค้นหาแฮ็กเกอร์ที่กำลังพยายามเข้ามาในระบบ หรือ ปิดการให้บริการของระบบ (a denial of service attack) โดยการนำแพ็กเกจต่างๆ ที่เข้ามาสู่ระบบ แล้วนำมาวิเคราะห์เปรียบเทียบกับกฎต่างๆ ที่ระบุไว้ว่าเป็นการโจมตีหรือการบุกรุก รวมถึงนโยบายขององค์กรก็นำมาพิจารณาด้วย เพื่อทำการตรวจสอบว่ามีสิ่งผิดปกติเกิดขึ้นกับระบบบ้างหรือไม่ ตัวอย่างที่มักจะเกิดขึ้นคือ การส่งแพ็กเกจที่เป็นการร้องขอการเชื่อมต่อ(TCP connection requests (SYN)) สู่พอร์ต(port) ต่างๆ บนเครื่อง เป้าหมาย หรือส่งแพ็กเกจจำนวนมากจนเครื่องเป้าหมายรับไม่ไหวทำให้ระบบต้องหยุดตัวลง โดยทั่วไปแล้วระบบตรวจจับผู้บุกรุกเครือข่ายนี้จะถูกติดตั้งบนเครื่องเดียวแต่ตรวจสอบและวิเคราะห์แพ็กเกจทั้งระบบเครือข่าย โดยจะต้องทำการติดตั้งบนระบบที่ใช้ฮับ (hub) ในการเชื่อมต่อ เพื่อที่จะสามารถรับข้อมูลทั้งหมดในช่องทางการสื่อสารได้ ถ้าเราติดตั้งระบบนี้บนสวิตช์เราจะรับข้อมูลได้เฉพาะของเครื่องเราเท่านั้น ซึ่งก็จะทำให้ไม่ได้มีผลในเชิงประสิทธิภาพ

5.4 หลักการทำงานพื้นฐานของระบบตรวจจับผู้บุกเครือข่าย

5.4.1 การดักจับแพ็กเกจจากเครือข่าย (Packet Sniffer)

ระบบตรวจจับผู้บุกรุกเครือข่าย ทำงานโดยการใช้แหล่งข้อมูลจากเครือข่าย เป็นการดักจับแพ็กเกจที่ผ่านมาในเครือข่ายที่อยู่ในแชร์โดเมน (share domain) เดียวกัน และนำข้อมูลแพ็กเกจมาวิเคราะห์ และเมื่อตรวจพบลักษณะที่ตรงกับข้อมูลที่จัดว่าเป็นการบุกรุกอยู่ก็จัดการตามที่ตั้งเอาไว้

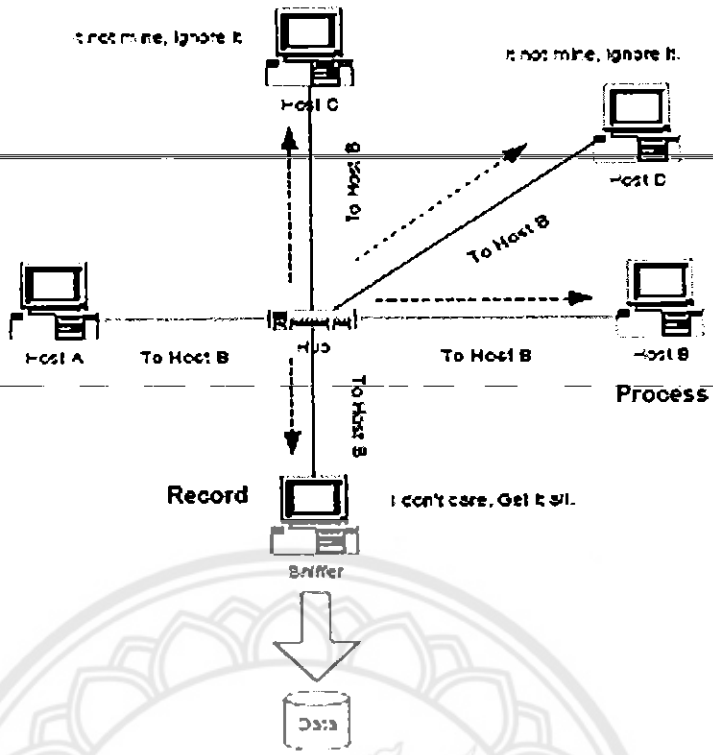
ต่อไปซึ่งอาจจะเป็นการเก็บข้อมูลลงล็อกไฟล์ (log file) หรือการแสดงข้อความเตือนผู้ดูแลระบบ ซึ่งในส่วนของการที่ได้ ข้อมูลมานั้นจะใช้หลักการของเครื่องมือที่ชื่อว่า แพ็กเกจสนิฟเฟอร์(Packet Sniffer หรือ Network Wire Tapping Device)

การที่สนิฟเฟอร์สามารถดักอ่านข้อมูลที่อยู่บนเน็ตเวิร์ค ได้นั้นมีสาเหตุที่สำคัญคือด้วยลักษณะของโพรโตคอลอีเทอร์เน็ตที่ใช้หลักการกระจายของข้อมูลไปยังทุกโฮสต์ ที่อยู่ในเน็ตเวิร์คและอาศัยโฮสต์แต่ละตัวทำหน้าที่จำแนกการสื่อสารของตนเอง นั่นหมายความว่าข้อมูลทุกแพ็กเกจที่ใช้สื่อสารกันนั้นได้ถูกส่งไปยังโฮสต์ทุกตัว ซึ่งจะได้รับพร้อมกันและเหมือนกัน เพียงแต่การที่จะสื่อสารกันได้อย่างถูกต้องนั้น โฮสต์แต่ละตัวจะต้องมีกระบวนการที่สามารถรู้ได้ว่าข้อมูลแพ็กเกจใดเป็นของตัวเอง และข้อมูล แพ็กเกจใดมิใช่ของตนเอง ทุกๆแพ็กเกจที่กระจายลงบนเน็ตเวิร์กนั้นจะมีหมายเลขระบุชัดเจนคือ MAC Address หรือ เรียกอีกอย่างหนึ่งว่าอีเทอร์เน็ตแอดเดรส(Ethernet Address) ซึ่งเป็นสิ่งที่บอกว่าแพ็กเกจมาจากฮาร์ดแวร์ใดในเน็ตเวิร์ค ทำให้สามารถระบุได้ว่าแพ็กเกจนั้นส่งมาจากโฮสต์ใด และต้องการส่งให้โฮสต์ใด

MAC Address จะเป็นหมายเลขเฉพาะ ตามฮาร์ดแวร์ทุกชนิด ที่ใช้การสื่อสารโดยโพรโตคอล อีเทอร์เน็ต และในทางทฤษฎีแล้วฮาร์ดแวร์ทุกชนิด จะไม่มี MAC Address ที่ซ้ำกัน โดยทั่วไป MAC Address จะถูกกำหนดตายตัวอยู่ใน Rom ของฮาร์ดแวร์และไม่สามารเปลี่ยนแปลงได้โดยซอฟต์แวร์ แต่เนื่องจาก การใช้งานของฮาร์ดแวร์นั้น ต้องควบคู่ไปกับไดรเวอร์ของฮาร์ดแวร์นั้นๆ ซึ่งโดยปกติแล้ว ไดรเวอร์จะถูกกำหนดให้ปฏิบัติตาม โพรโตคอลอย่างเข้มงวดคือ

- ให้รับข้อมูลที่มี MAC Address เป็นของตนเองเท่านั้น (ห้ามอ่านข้อมูลผู้อื่น)
- ให้ส่งข้อมูลโดยใช้ MAC Address ของตนเองเท่านั้น (ห้ามปลอมเป็นผู้อื่น)

หากใช้ไดรเวอร์ตามปกติที่มากับฮาร์ดแวร์แล้วเครื่องคอมพิวเตอร์ก็จะทำงานอยู่ในรูปแบบปกติ และไม่สามารถวุ่นวายกับข้อมูลอื่นได้ แต่อย่างไรก็ตามไดรเวอร์ก็เป็นเพียงโปรแกรมประเภทหนึ่งเท่านั้นที่ทำหน้าที่จัดการกับ การสื่อสารข้อมูลในระดับต่ำของฮาร์ดแวร์กับระบบปฏิบัติการ ดังนั้นถ้าเขียนไดรเวอร์ขึ้นมาใหม่ให้ไม่มีข้อจำกัดทางโพรโตคอล โดยละคุณสมบัติที่ให้รับข้อมูลเฉพาะที่มี MAC Address เป็นของตนเองเท่านั้นสนิฟเฟอร์ก็จะสามารถทำงานได้ โหมดการทำงานที่อนุญาตฮาร์ดแวร์รับข้อมูลของผู้อื่นเข้ามาได้ โดยไม่มีการปิดกั้นเรียกว่า โพรมิสคูอัสโหมด (Promiscuous Mode) เป็นโหมดที่ทำให้ฮาร์ดแวร์อ่านข้อมูลดิบทั้งหมดบนเน็ตเวิร์คเข้ามาในเครื่องคอมพิวเตอร์ของตนเองได้โดยไม่สนใจว่าจะเป็นของใคร ส่งให้ใคร และเป็นการละเมิดข้อบังคับของโพรโตคอลหรือไม่และเนื่องจาก ระบบตรวจจับผู้บุกรุกทางเครือข่ายทำงานโดยอาศัยหลักการนี้ ดังนั้นจึงทำให้เกิดข้อจำกัดในการใช้งาน คือ จะไม่สามารถตรวจจับนอกแชร์โดเมนของตัวเองได้ ดังนั้นถ้าเครือข่ายเป็นเครือข่ายที่ใช้สวิตช์ ระบบตรวจจับผู้บุกรุกจะไม่สามารถทำงานได้



รูป 5.1 รูปภาพจำลองการทำงานของสนิฟเฟอร์

5.4.2 ลักษณะของซิกเนเจอร์ที่ใช้ในการตรวจสอบ

การวิเคราะห์แพ็กเกจนั้น เราจะสนใจแพ็กเกจที่มีลักษณะซิกเนเจอร์ตรงกับที่มีข้อมูลอยู่ ซิกเนเจอร์แบ่งออกได้เป็น 3 ประเภท ดังนี้ คือ

1. สตริงซิกเนเจอร์ (string signatures)

จะสนใจส่วนของข้อมูลในแพ็กเกจ โดยหาส่วนของสตริงที่อาจบ่งถึงว่าเป็นการบุกรุกได้ ตัวอย่างของสตริงซิกเนเจอร์สำหรับระบบยูนิกซ์ เช่น "cat"++">/.rhosts" ซึ่งถ้าเจอในแพ็กเกจใด ก็อาจสรุปได้ว่าเป็นแพ็กเกจของการโจมตี

2. พอร์ตซิกเนเจอร์ (port signatures)

ตรวจดูการเชื่อมต่อที่ต่อไปยังพอร์ตที่นิยมใช้ในการโจมตี ตัวอย่างของพอร์ตเหล่านี้ ได้แก่ telnet (ทีซีพี พอร์ต 23) , FTP (ทีซีพี พอร์ต 21/20) , SUNRPC (ทีซีพี/ยูดีพี พอร์ต 111), และ IMAP (ทีซีพี พอร์ต 143) ซึ่งถ้าพอร์ตเหล่านี้ไม่ได้ถูกใช้โดยไซต์นั้นแล้วแพ็กเกจที่เข้ามายังพอร์ตเหล่านี้ก็จะจัดว่าเป็นที่น่าสงสัยว่าอาจเป็นการบุกรุก

3. เฮดเดอร์ซิกเนเจอร์ (header signature)

ตรวจสอบส่วนหัวของแพ็กเกจว่ามีส่วนประกอบที่ผิดปกติไม่สมเหตุสมผลหรือไม่ตัวอย่างที่รู้จักกันดีที่สุดก็คือ Winnuke หรืออีกตัวอย่างก็คือ แพ็กเกจที่มีทั้งแฟล็ก SYN และ FIN ตั้งไว้ซึ่งหมายความว่าผู้ส่งต้องการที่จะเริ่มและหยุดการติดต่อในเวลาเดียวกัน

5.5 ประโยชน์ของระบบตรวจจับผู้บุกรุกที่เป็นแบบทางเครือข่าย

ระบบตรวจจับผู้บุกรุกทางเครือข่ายมีหลายจุดแข็งซึ่งตัวตรวจจับผู้บุกรุก ที่เป็นแบบโฮสต์เบสไม่สามารถทำได้โดยลำพัง ได้แก่การที่สามารถดักจับแพ็กเกจแบบเรียลไทม์ และวิเคราะห์มันได้ โดยในหัวข้อต่อไปนี้จะเป็นจุดแข็งที่แสดงให้ถึงความจำเป็นของการมีระบบตรวจจับผู้บุกรุกเป็นส่วนหนึ่งระบบรักษาความปลอดภัย

1. ค่าของการดูแลจัดการ

ระบบรักษาความปลอดภัยทางเครือข่าย ให้การนำมาใช้กับระบบที่ต้องการเป็นไปได้ง่าย โดยสามารถติดตั้งเป็นจุดๆไปได้ และใช้ได้กับเครื่องเป้าหมายที่กว้างซอฟต์แวร์ที่ติดตั้งนั้นก็ไม่จำเป็นต้องติดในทุกเครื่องเหมือนกับแบบโฮสต์เบส การที่จุดติดตั้งการตรวจจับมีจำนวนน้อย ทำให้ค่าการดูแลและจัดการเป็นไปอย่างมีประสิทธิภาพมากขึ้น

2. การวิเคราะห์แพ็กเกจ

ระบบตรวจจับผู้บุกรุกทางเครือข่าย จะตรวจสอบในส่วนหัวของ แพ็กเกจเพื่อหาสัญญาณของการบุกรุก หรือการกระทำที่เป็นที่น่าสงสัย ซึ่งการโจมตีเพื่อปิดบริการในปัจจุบันหลายตัวจะถูกตรวจพบได้โดยการดูที่ส่วนหัวของมันเมื่อแพ็กเกจผ่านมาในเครือข่าย ตัวอย่างเช่น การโจมตีแบบ Land จะเป็นแพ็กเกจที่ถูกปลอมขึ้นมาให้มีไอพีแอดเดรสต้นทางและแอดเดรสปลายทางเหมือนกัน ซึ่งการโจมตีประเภทนี้จะถูกตรวจพบได้โดยง่าย เมื่อใช้ระบบตรวจจับผู้บุกรุกทางเครือข่ายทำงานแบบเรียลไทม์ ทั้งนี้การโจมตีที่ใช้แพ็กเกจแฟร็กเมนต์ เช่น Teardrop ก็สามารถถูกตรวจพบได้ในชั้นการวิเคราะห์แพ็กเกจเช่นกัน ซึ่งระบบตรวจจับผู้บุกรุกแบบโฮสต์เบสจะไม่สามารถตรวจพบการโจมตีประเภทเหล่านี้ได้ และนอกจากการตรวจสอบที่ส่วนหัวของแพ็กเกจแล้ว ระบบตรวจสอบผู้บุกรุกทางเครือข่ายยังสามารถตรวจสอบในส่วนเนื้อข้อมูลในแพ็กเกจเพื่อที่จะหาคำสั่งเฉพาะหรือรูปแบบโครงสร้างบางชนิดที่ใช้ในการโจมตี ซึ่งคำสั่งเหล่านี้จะเป็นตัวชี้บอกถึงว่าเป็นการโจมตี ไม่ว่าการโจมตีนั้นจะสำเร็จหรือไม่ก็ตาม ตัวอย่างเช่น ผู้บุกรุกทำการลองตรวจสอบการมีของโปรแกรม Back Orifice ในระบบที่ไม่ถูกบุกรุกโดย Back Orifice ซึ่งการกระทำนี้จะไม่มีผลกระทบต่อระบบนี้ แต่เราจะสามารถรู้ได้ถึงความพยายามที่จะบุกรุก ซึ่งถ้าเป็นแบบโฮสต์เบสจะไม่สามารถตรวจสอบแบบข้อมูลในแพ็กเกจได้

3. การลบร่องรอย

ระบบตรวจจับผู้บุกรุกคอมพิวเตอร์เครือข่ายใช้การตรวจจับแบบเรียลไทม์และเมื่อตรวจจับได้แล้ว ผู้บุกรุกจะไม่สามารถลบหลักฐานนี้ทิ้งได้ สิ่งที่ตรวจจับได้ไม่ใช่เพียงแค่การโจมตีเท่านั้น แต่ยังมีข้อมูลอื่น ที่อาจนำต่อไปได้ ถึงผู้บุกรุกด้วยปัญหาหนึ่งของระบบตรวจจับผู้บุกรุกทางคอมพิวเตอร์แบบโฮสต์เบสที่มักจะพบก็คือ ผู้บุกรุกเข้าใจและมีความรู้ในเรื่องล็อกไฟล์เป็นอย่างดี และมันก็มักจะเป็นที่แรกที่ผู้บุกรุกจะไปลบรอยของตัวเอง และเอาออก หรือทำลายข้อมูลส่วนนั้น

4. การตรวจจับและการตอบสนองแบบเรียลไทม์

ระบบตรวจสอบผู้บุกรุกทางเครือข่ายตรวจจับความผิดปกติ หรือการบุกรุกที่เกิดขึ้นอย่างทันต่อเหตุการณ์และรายงานในทันที ตัวอย่างเช่น ถ้าตรวจสอบพบว่าการโจมตีเพื่อให้ปิดบริการเกิดขึ้นโดยเกิดบนโพรโตคอลทีซีพี อาจจะมีการสั่งให้ระบบส่ง ทีซีพี รีเซตในทันทีเพื่อหยุดยั้งการโจมตีไว้ก่อนที่จะทำให้เกิดความเสียหายแก่ระบบมากขึ้น ในหลายๆ สถานการณ์ด้วยระบบแบบโฮสต์เบส การรับทราบถึงเหตุการณ์ที่เกิดจะเกิดขึ้นในภายหลังและบางทีอาจไม่มีการเตือนถึงเลย เนื่องมาจากที่ระบบได้เสียหายไปก่อนแล้ว ในการที่มีการเตือนแบบเรียลไทม์นั้นจะทำให้สามารถตอบสนองต่อเหตุการณ์ที่เกิดขึ้นได้อย่างทันท่วงที หรือถ้าไม่ต้องการก็สามารถรวบรวม ข้อมูลไว้เพื่อวิเคราะห์ต่อในภายหลังได้

5. การตรวจจับเจตนาที่มุ่งร้าย

ระบบจะมีประโยชน์มากในการต้องการตรวจจับถึงเจตนาของการกระทำ ถ้าหากนำระบบตรวจจับผู้บุกรุกทางเครือข่ายไปไว้ก่อนไฟร์วอลล์ เราจะสามารถรู้ได้ถึงความพยายามในการที่จะโจมตีของผู้บุกรุกได้ แม้ว่าแพ็กเกจที่ต้องการโจมตีนั้นจะถูกปฏิเสธโดยไฟร์วอลล์ก็ตาม ถ้าเป็นระบบแบบโฮสต์เบสจะไม่มีโอกาสตรวจพบความพยายามที่จะโจมตีที่โดนปฏิเสธออกไปแล้วนี้เลย เนื่องจากมันไม่ได้ถูกกระทำจริงบนโฮสต์ แต่มันก็ถือว่าเป็นสิ่งจำเป็นที่ต้องรู้ถึงความถี่และชนิดของการโจมตีที่กระทำบนเครือข่ายของเรา

6. การทำให้สมบูรณ์ยิ่งขึ้นและการช่วยตรวจสอบ

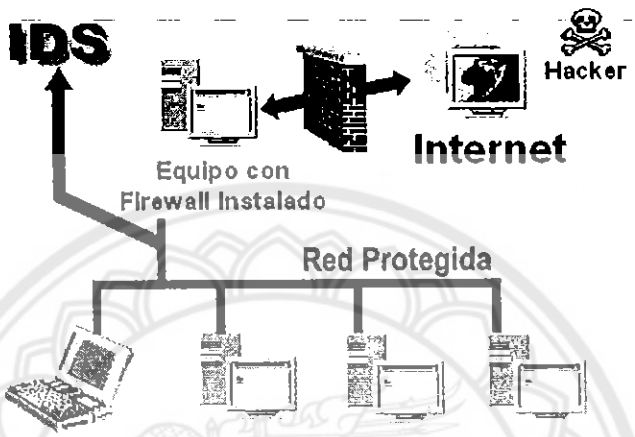
ระบบตรวจสอบผู้บุกรุกจะเป็นองค์ประกอบที่ทำให้ส่วนอื่นๆที่ใช้ในมาตรการรักษาความปลอดภัยอยู่แล้วสมบูรณ์ยิ่งขึ้น ตัวอย่างเช่นในการใช้การเข้ารหัสข้อมูล แม้ว่าระบบตรวจสอบผู้บุกรุกบนเครือข่ายจะไม่สามารถอ่านข้อมูลที่เข้ารหัสได้ แต่มันจะสามารถตรวจสอบได้ว่า ข้อมูลในเครือข่ายอันไหนที่ไม่ได้ถูกเข้ารหัสไว้ ส่วนในกรณีของไฟร์วอลล์ ระบบตรวจสอบผู้บุกรุกบนเครือข่ายจะช่วยในการตรวจสอบว่ามันได้ทำหน้าที่ในการป้องกันแพ็กเกจที่ควรจะปฏิเสธได้เต็มที่ถูกต้อง ครบถ้วนหรือยัง

7. การไม่ขึ้นอยู่กับระบบปฏิบัติการใดๆ

ระบบตรวจสอบผู้บุกรุกบนเครือข่ายไม่ขึ้นอยู่กับ ระบบปฏิบัติการของโฮสต์ที่เราต้องการตรวจสอบความผิดปกติ เหมือนกับวิธีของแบบโฮสต์เบสซึ่ง ข้อมูลในล็อกของระบบแบบโฮสต์เบสจะได้มาได้ต้องขึ้นกับการทำงานของระบบปฏิบัติงานที่ถูกต้อง

บทที่ 6
การทดลองและวิเคราะห์ผลการทดลอง

6.1 การติดตั้งโปรแกรม



รูปที่ 6.1. แสดงตำแหน่งการติดตั้งโปรแกรมตรวจจับการบุกรุกหลังไฟล์วอล

การติดตั้งโปรแกรมตรวจจับการบุกรุกที่ตำแหน่งหลังไฟล์วอลนั้นมีข้อดีตรงที่ให้ไฟล์วอลเป็นตัวกรอง packet ในครั้งแรกจากระบบอินเทอร์เน็ต หากมีแพ็กเก็จใดหลุดรอดออกมาโปแกรมตรวจจับการบุกรุกสามารถที่จะนำ packet หลังจากการกรองในชั้นแรกมาวิเคราะห์มาทำการวิเคราะห์และประมวลผล ซึ่งมันจะได้ผลดีกว่าตรงที่โปรแกรมตรวจจับการบุกรุกทางเครือข่ายไม่จำเป็นต้องวิเคราะห์ packet อื่นๆ ที่ไฟล์วอลไม่อนุญาตให้ผ่านเข้ามาในระบบเครือข่ายคอมพิวเตอร์ นอกจากนี้ ทำให้โปรแกรมตรวจจับการบุกรุกมีการทำงานที่รวดเร็ว เสถียร และมีระยะการใช้งานอย่างมีประสิทธิภาพได้ยาวนานยิ่งขึ้น

6.2 การทำงานของโปรแกรม

การทำงานของโปรแกรมตรวจจับการบุกรุก จะต้องสัมพันธ์กับฮาร์ดแวร์คอมพิวเตอร์ด้วย กล่าวคือ สวิตซ์ชิงต้องมีการเซ็ตค่าให้อยู่ในโหมดที่คอมพิวเตอร์ในระบบเครือข่ายแต่ละเครื่องสามารถที่จะรับ packet ของคอมพิวเตอร์เครื่องอื่นได้ การทำหน้าที่นี้ จะมีเพียงผู้ดูแลระบบเท่านั้นที่สามารถทำได้ ทำให้มีความปลอดภัยในระดับหนึ่ง การเปิดโหมดการทำงานของสวิตซ์ชิงที่อนุญาตให้เครื่องในเครือข่ายสามารถรับ packet อื่นนอกจากของเครื่องตัวเองนั้น มีวิธีการเซ็ตที่ง่ายเพียงกดปุ่มที่อนุญาตดักจับ packet ในเครือข่ายของสวิตซ์ชิงเท่านั้น จะสามารถศึกษาได้จากคู่มือ

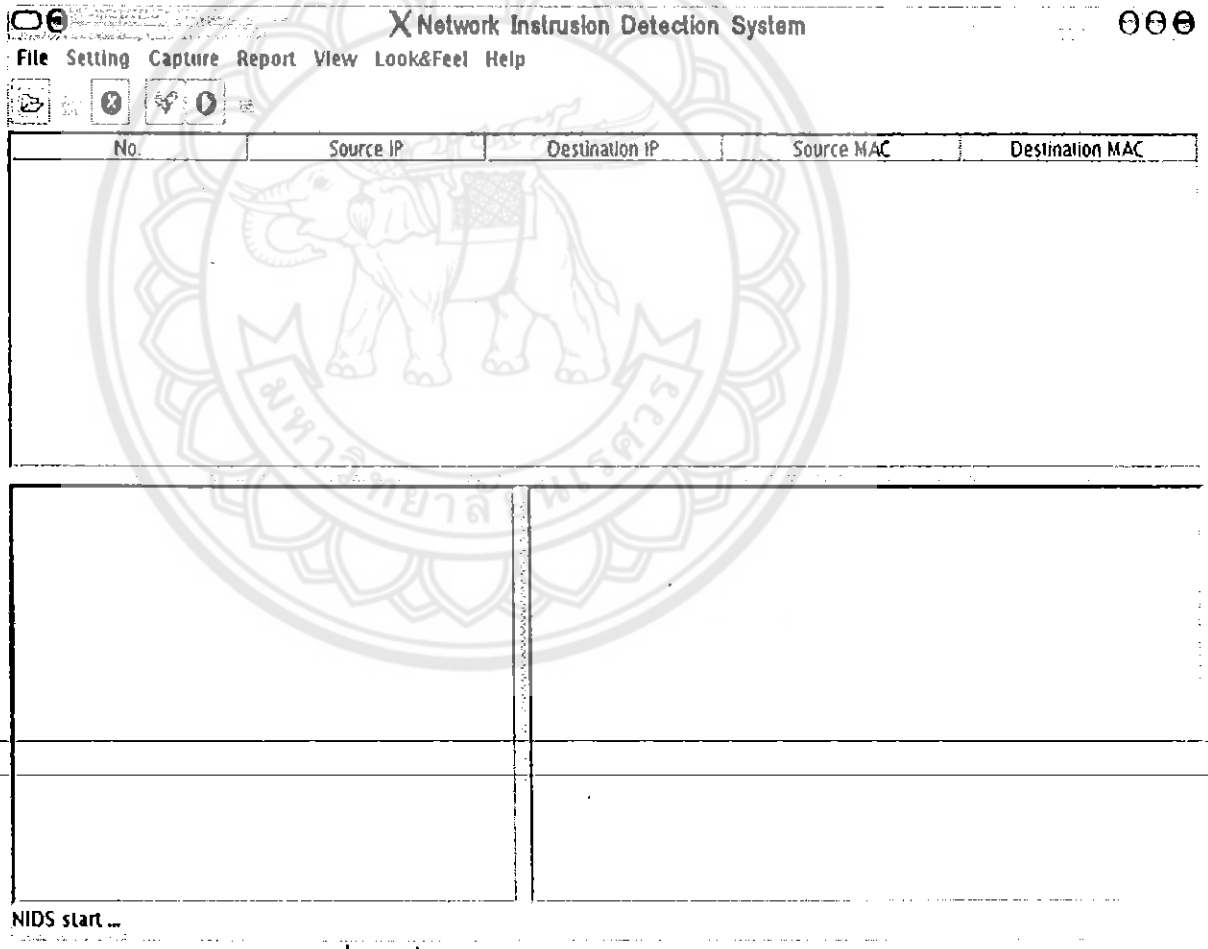
ของสวิตซ์ชิ่งแต่ละตัว ในการทดลองครั้งนี้ สวิตซ์ชิ่งที่ทำการทดลองเป็นของบริษัท 3COM จะมีปุ่มนี้อยู่ เพียงกดเปลี่ยนจาก โหมด DMIX ซึ่งเป็นโหมดที่ไม่อนุญาตให้เครื่องในระบบเครือข่ายคอมพิวเตอร์รับ packet ของเครื่องอื่นไหนเครือข่ายเดียวกันเป็น DMI เป็นโหมดที่อนุญาตให้เครื่องในระบบเครือข่ายคอมพิวเตอร์สามารถที่จะรับ packet ต่างๆได้ถึงแม้จะไม่ใช่ packet ของตัวเองก็ตาม

6.2.1 การเริ่มการทำงานของระบบตรวจจับการบุกรุก

การเริ่มการทำงานของระบบตรวจจับการบุกรุกนั้นมีวิธีการที่ง่าย โดยการเปิดเทอร์มินอลในระบบปฏิบัติการลีนุกซ์ขึ้นมา แล้วใช้คำสั่งดังต่อไปนี้

java -jar nids_ping_flood.jar

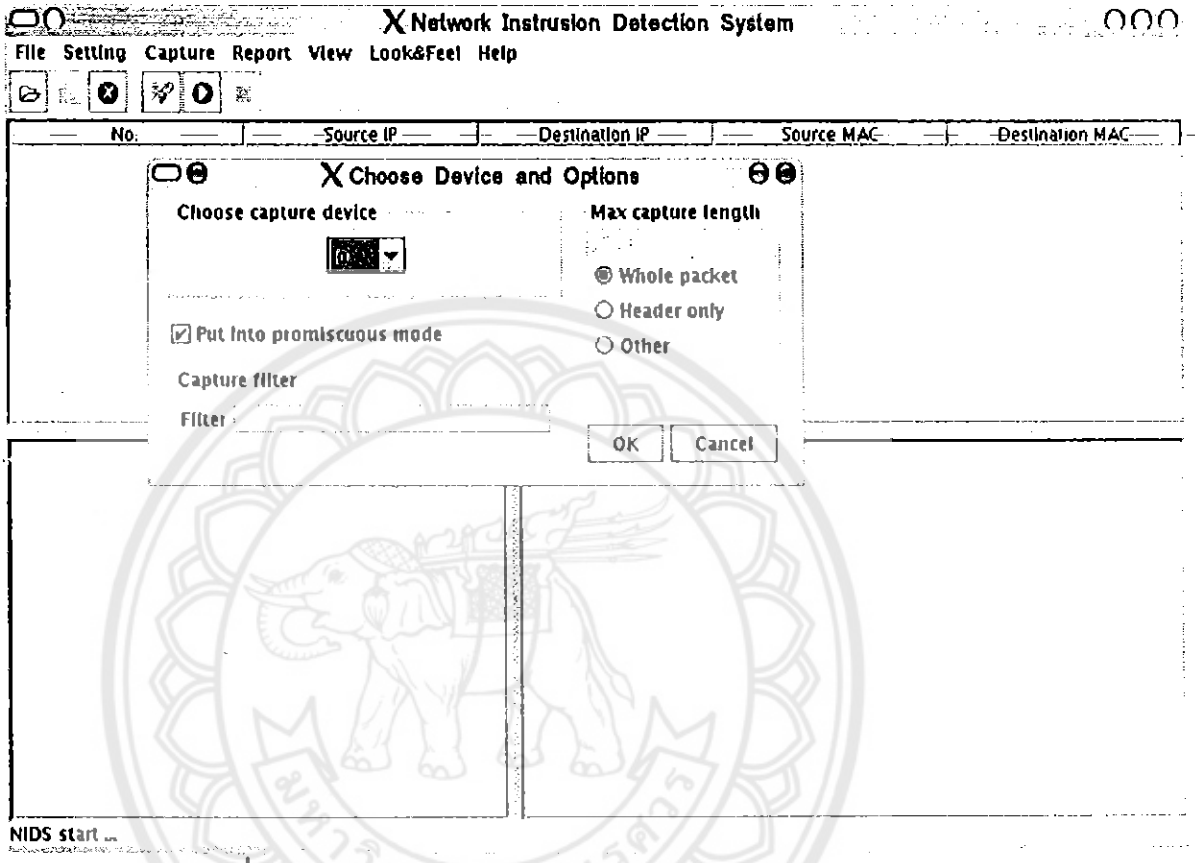
คำสั่งดังกล่าว เป็นการเริ่มการทำงานของระบบตรวจจับการบุกรุก ผลของการทำงาน ดังนี้



รูปที่ 6.2 เริ่มการทำงานของระบบตรวจจับการบุกรุก

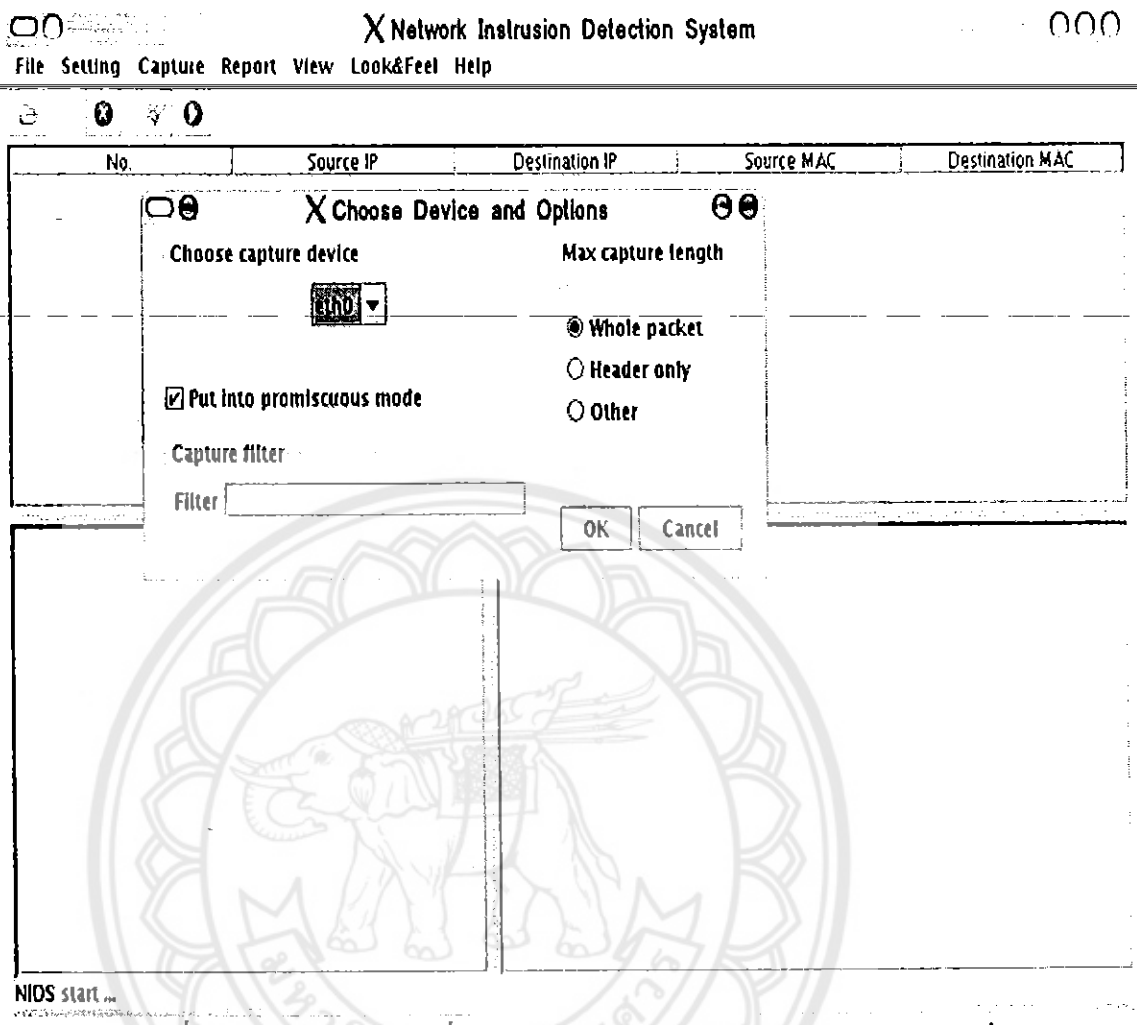
6.2.2 ทดสอบการทำงานและการวิเคราะห์ของระบบตรวจจับการบุกรุก

การทดสอบโปรแกรมให้คลิกไปที่ เมนู Capture-> Start ผลที่ได้จะมีหน้าต่างหน้าต่างสำหรับเลือกการ์ดแลนขึ้นมา



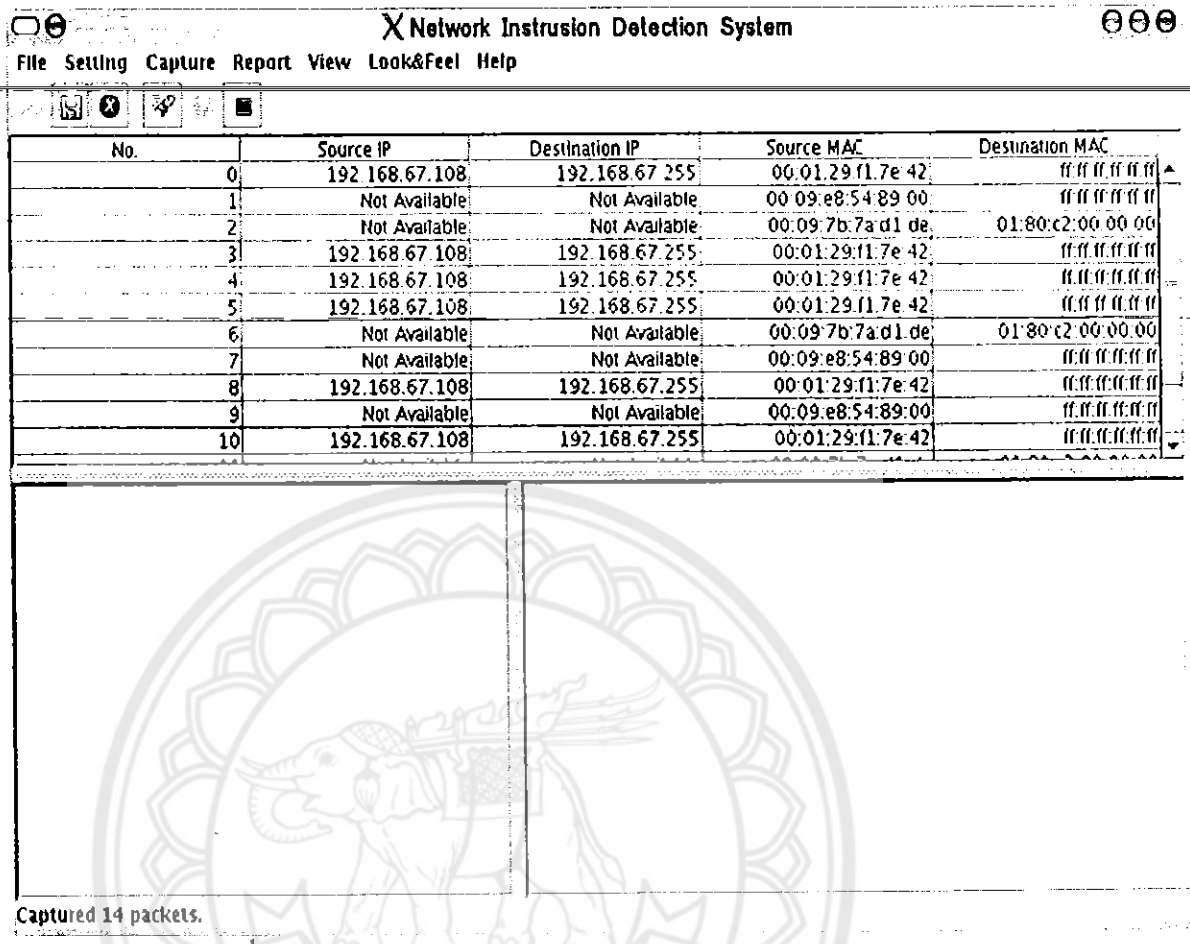
รูปที่ 6.3 หน้าต่างเลือกการ์ดแลนของระบบตรวจจับการบุกรุก

ในระบบปฏิบัติการลีนุกซ์การ์ดแลนแรกจะมีชื่อว่า eth0 สำหรับการ์ดแลนที่ 2 , 3 จะมีชื่อว่า eth1 , eth2 ตามลำดับ ดังในรูปที่ 6.3 จะเห็นได้ว่ามีการ์ดแลนชื่อ lo ความจริงแล้ว lo หมายถึงเครื่องลูปแบ็ก (loop back) หมายความว่าเป็นเครื่องที่ไม่ได้ติดต่อกับเครื่องอื่นในระบบเครือข่ายคอมพิวเตอร์ หรือเป็นเครื่องเดี่ยวๆนั้นเอง หากต้องการที่จะติดต่อกับเครื่องอื่นจำเป็นต้องเลือกการ์ดแลนที่มีชื่อว่า eth0 หรือ Ethernet card 0 เป็นการ์ดแลนที่ใช้เชื่อมกับระบบเครือข่ายคอมพิวเตอร์ ดังนี้



รูปที่ 6.4 เลือกการ์ด eth0 เพื่อใช้ในการติดต่อกับระบบเครือข่ายคอมพิวเตอร์

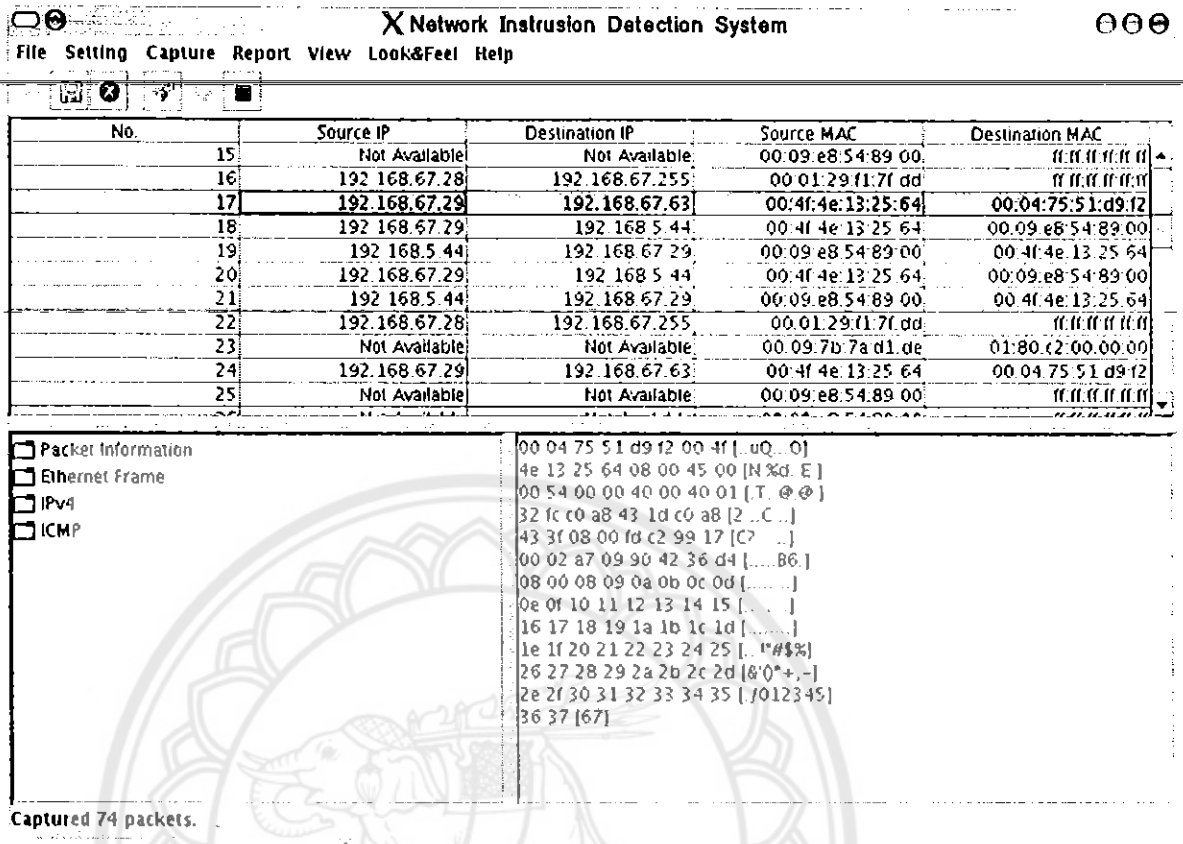
จากนั้น กดปุ่ม OK เพื่อเริ่มการตรวจจับ packet ที่ผ่านเข้าออกในระบบเครือข่ายคอมพิวเตอร์ พร้อมกันนั้นระบบตรวจจับการบุกรุกทางเครือข่ายยังนำ packet ที่ตรวจจับได้นำมาวิเคราะห์ว่าเป็น packet แปลกปลอม หรือเป็น packet ที่มีลักษณะเข้าข่ายของ packet ที่ใช้ในการโจมตีหรือไม่ ถ้าใช่ระบบตรวจจับการบุกรุกจะมีหน้าต่างแจ้งเตือนถึง packet ที่ต้องสงสัยและการบันทึกเวลาที่พบ packet ดังกล่าวพร้อมกับวิธีการโจมตี คำอธิบายถึงลักษณะของการโจมตีนั้นที่ระบบฐานข้อมูล MySQL พร้อมทั้งแจ้งเตือนไปยังผู้ดูแลระบบโดยการส่งไปรษณีย์อีเล็กทรอนิกส์



รูปที่ 6.5 ระบบตรวจจับการบุกรุกมีการตรวจจับทุก packet

ดังรูปที่ 6.5 จะเห็นได้ว่า ระบบตรวจจับการบุกรุกทางเครือข่าย มีการตรวจจับ packet ทุกชนิด ทั้งรู้จักและไม่รู้จัก packet ชนิดนั้น นำเข้าไปสู่ขบวนการวิเคราะห์ packet แยก packet ออกมาวิเคราะห์ แล้วแสดงส่วนของไอพีแอดเดรสและค่าแมค แอดเดรสของทั้งสองฝั่งทั้งทางที่เป็น Source IP กับ Destination IP มาแสดง เพื่อแสดงให้เห็นว่าฝ่ายใหนเป็นฝ่ายที่ติดต่อกับอีกเครื่องหนึ่งก่อน โดยฝ่าย Souce IP เป็นฝ่ายที่ทำการติดต่อขอใช้ทรัพยากร หรือ เป็นฝ่ายที่ใช้ในการโจมตี ส่วนฝ่าย Destination IP เป็นฝ่ายที่ถูกร้องขอเข้าใช้ทรัพยากรหรือในแง่หนึ่งถ้าเป็นการโจมตีก็คือเป็นเหยื่อของการโจมตีดังกล่าว

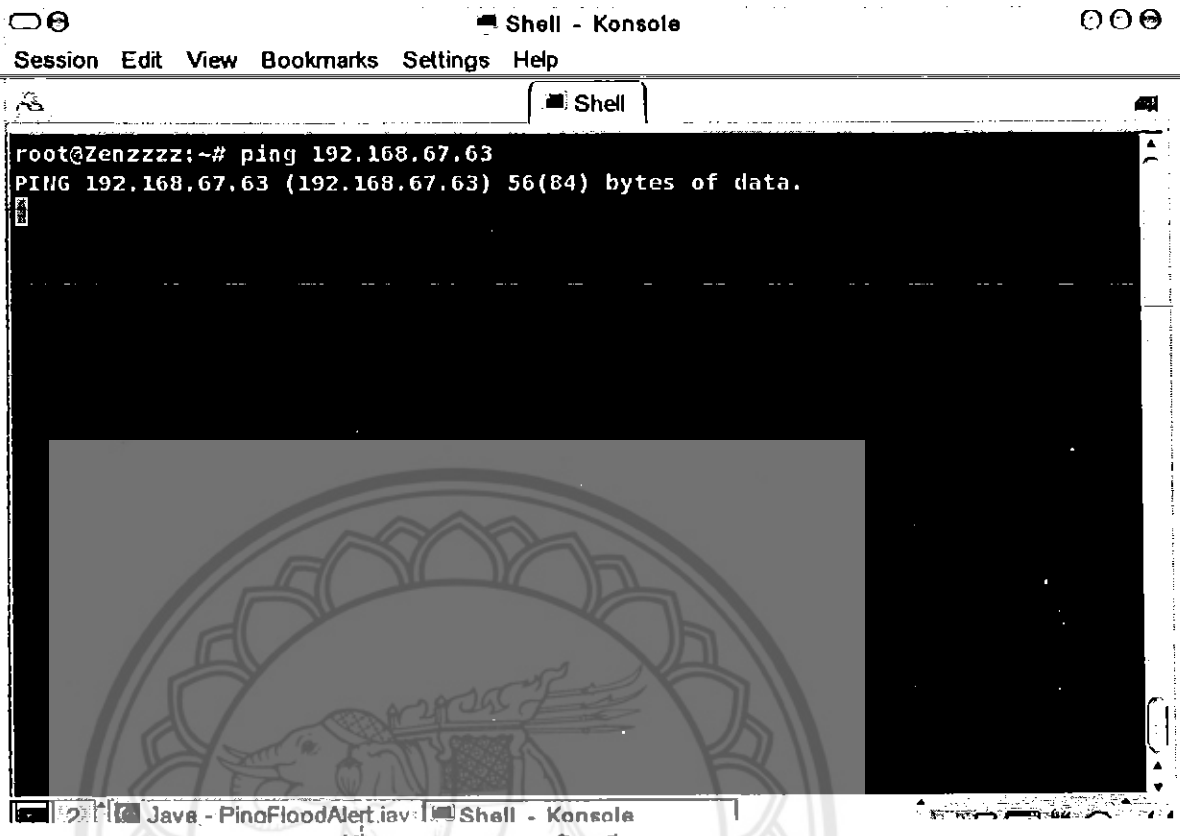
การวิเคราะห์ packet



รูปที่ 6.6 ส่วนของการวิเคราะห์การโจมตี

packet ที่ผ่านเข้ามาในกระบวนการวิเคราะห์การโจมตี หลังจากการที่ได้แตกย่อย packet โดยการนำเอาข้อมูลในแต่ละชั้นของการสื่อสารออกมาแล้ว จะนำมาวิเคราะห์ว่าเข้าข่ายการโจมตีด้วยวิธี Ping Flood หรือไม่ โดยการเช็คว่าถ้าเกิดมี packet ที่มีการส่ง ICMP Packet มา จำนวน 10 ครั้งในเวลา 10 วินาทีถือได้ว่า กลุ่ม packet ดังกล่าวเข้าข่ายวิธีการโจมตีแบบ Ping Flood ซึ่งโดยปรกติแล้ว การใช้ ICMP Packet จะใช้ในกรณีที่ต้องการทดสอบว่า เครื่องที่ต้องการทดสอบมีการทำงานอยู่หรือไม่ โดยการส่ง packet จำนวนเพียง 4 packet เท่านั้น แต่ในกรณีที่มี packet เกินดังกล่าว อาจเป็นไปได้สองแง่ คือ มีการโจมตีแบบ Ping Flood เพื่อให้ข้อมูลในระบบเครือข่ายล้นไปด้วยข้อมูลไม่เป็นประโยชน์ จนให้บริการอื่นๆไม่ได้ หรือมีการทดสอบโดยการส่ง packet มามากกว่า 4 packet ซึ่งในกรณีนี้มักจะมีน้อยหรือไม่ค่อยเกิดขึ้น ดังนั้น ระบบตรวจจับการบุกรุกทางเครือข่ายจึงสันนิษฐานไว้ก่อนว่า เกิดการโจมตีแบบ Ping Flood ไว้ก่อน

ทดสอบการโจมตี



รูปที่ 6.7 ทดสอบการโจมตีแบบ Ping Flood

การทดสอบการโจมตีแบบ Ping Flood จะมีการส่ง packet แบบ ICMP เป็นจำนวนมากไปหาเครื่องเป้าหมายที่ต้องการโจมตีให้หยุดบริการ ดังรูปที่ 6.7 เป็นการโจมตีเครื่องในระบบเครือข่ายคอมพิวเตอร์ที่มีไอพีแอดเดรส 192.168.67.63

การโจมตีแบบ Ping Flood เป็นวิธีการที่นิยมมาก แต่การโจมตีดังกล่าวอาจต้องใช้ระยะเวลานานกว่าที่เครื่องโจมตีหยุดให้บริการ ทั้งนี้ต้องคำนึงถึง ขนาดของแบนด์วิดของระบบเครือข่ายคอมพิวเตอร์ที่เครื่องเป้าหมายตั้งอยู่ ขนาดของ packet ที่ส่งไป แต่เป็นวิธีที่พัฒนาสู่การโจมตีอื่นๆ ที่มีผลการการโจมตีที่รุนแรงและรวดเร็วกว่า ดังนั้นการตรวจจับการโจมตีแบบ Ping Flood จึงมีความสำคัญเช่นเดียวกันกับการตรวจจับการโจมตีแบบอื่นๆ

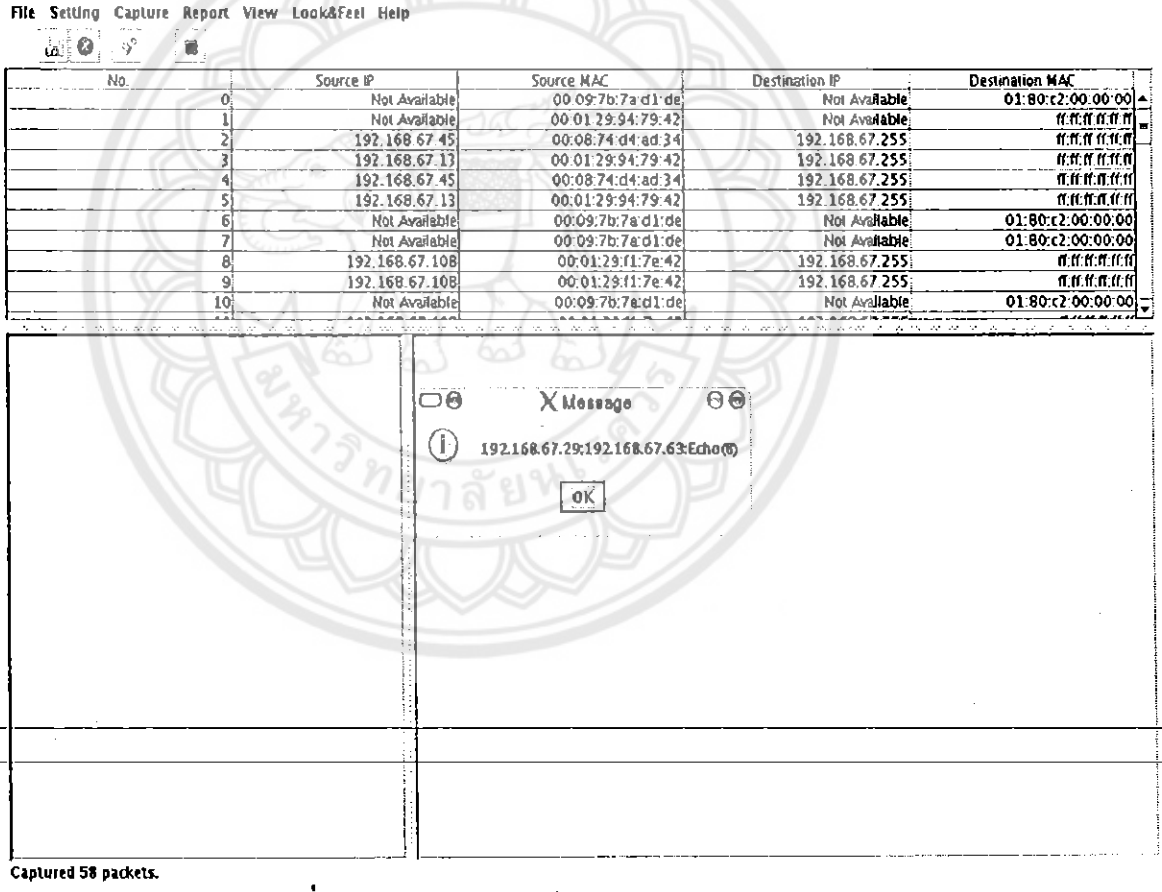
6.2.3 ผลการทำงานและการแจ้งเตือนของระบบตรวจจับการบุกรุก

หากส่วนวิเคราะห์การโจมตีแบบ Ping Flood ได้ตรวจจับ packet แล้วนำมาวิเคราะห์และเห็นว่ามีการโจมตีเกิดขึ้นในเครือข่ายระบบคอมพิวเตอร์ ระบบตรวจจับการบุกรุกก็มีการแจ้งเตือนอยู่สองวิธี คือ

1. การแจ้งเตือนแบบมีหน้าต่างแสดงถึงการโจมตี
2. การแจ้งเตือนโดยระบบตรวจจับการบุกรุกมีการส่งไปรษณีย์อิเลคทรอนิกส์

การแจ้งเตือนแบบนี้หน้าต่างแสดงถึงการโจมตี

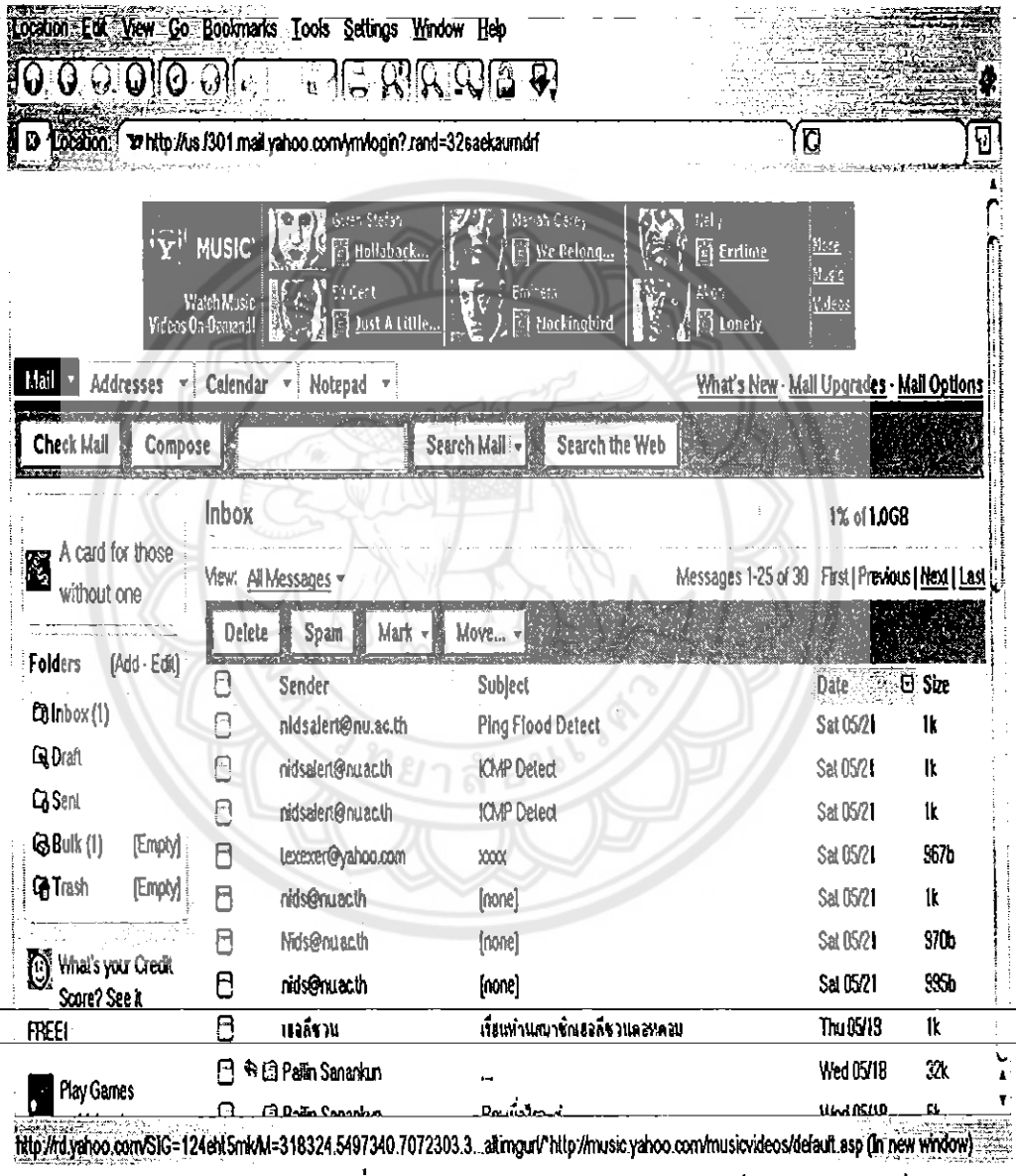
การแจ้งเตือนด้วยวิธีนี้ หากผู้ดูแลระบบนั่งอยู่หน้าเครื่องที่ได้ลงระบบตรวจจับการบุกรุกดังกล่าว จะสามารถทราบโดยทันทีว่า มีการโจมตีเกิดขึ้นแล้ว ทำให้ผู้ดูแลระบบสามารถหาทางป้องกันได้อย่างทันท่วงที



รูปที่ 6.8 แสดงหน้าต่างแจ้งการโจมตีแบบ Ping Flood

การแจ้งเตือนโดยระบบตรวจจับการบุกรุกมีการส่งไปรษณีย์อิเลคทรอนิกส์

การแจ้งเตือนโดยวิธีนี้มีความสะดวกมาก ในกรณีที่ผู้ดูแลระบบไม่ได้อยู่ในห้องควบคุม แต่สามารถที่จะทราบได้ว่า ในขณะนั้นระบบเครือข่ายคอมพิวเตอร์ถูกคุกคามหรือไม่โดยการเช็กจากไปรษณีย์อิเลคทรอนิกส์ หากเกิดการโจมตีแบบ Ping Flood เกิดขึ้นระบบตรวจจับการบุกรุกทางเครือข่ายจะส่งไปรษณีย์อิเลคทรอนิกส์มาแจ้งเตือนโดยอัตโนมัติ

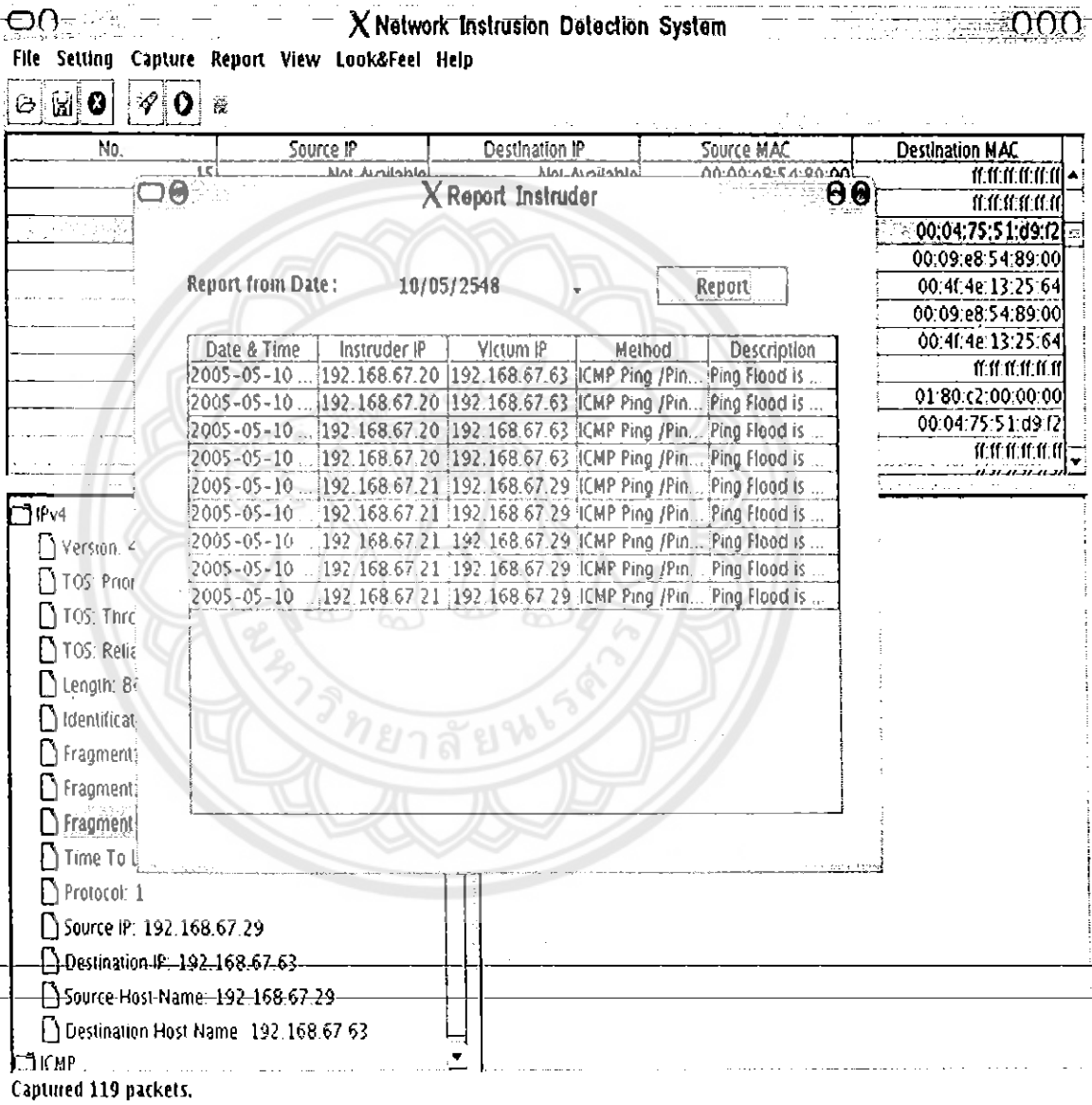


รูปที่ 6.9 แจ้งเตือนด้วยวิธีส่งไปรษณีย์อิเลคทรอนิกส์

6.2.4 รายงานผลการตรวจจับผู้บุกรุกทางเครือข่าย

ในส่วนการรายงานของระบบตรวจจับการบุกรุกทางเครือข่าย ว่ามีการดึงรายงานของการโจมตีแต่ละครั้งที่เก็บไว้ในฐานข้อมูล MySQL มาแสดงเพื่อเป็นเครื่องมือให้กับผู้ดูแลระบบไว้ทำการวิเคราะห์ว่าระบบที่ควบคุมอยู่มีความปลอดภัยเพียงใด

การดูรายงานคลิ้กที่เมนู **Report->Report Instruder** จะปรากฏหน้าต่างรายงาน ดังนี้



รูปที่ 6.10 แสดงรายงานที่เก็บไว้ในฐานข้อมูล MySQL

บทที่ 7
สรุปผลการทดลอง

7.1 สรุปผลการทดลอง

7.1.1 ส่วนของการติดตั้ง

ในการติดตั้งอาจมีการคอนฟิกที่ยุ่งยากบ้าง เพราะว่าใช้ภาษาจาวาในการพัฒนาโปรแกรม และใช้ตัวกลางที่พัฒนาจากภาษาซีเป็นติดต่อกับส่วนหลักของระบบปฏิบัติการ แต่ว่าก็มีข้อดีคือ สามารถที่จะติดตั้งโปรแกรมตรวจจับการบุกรุกได้ทุกระบบปฏิบัติการ เช่น ระบบปฏิบัติการวินโดว์ ระบบปฏิบัติการลีนุกซ์ หรือแม้กระทั้งบนระบบปฏิบัติการยูนิกส์

7.1.2 ส่วนในการทดสอบโปรแกรม

โปรแกรมที่ได้พัฒนาโดยภาษาจาวา อาจมีการทำงานที่ช้าบ้างในครั้งแรก เพราะเป็นภาษาที่มีการแปลทีละคำสั่งตอนขณะทำงานครั้งแรก แต่จะเก็บส่วนที่แปลแล้วไว้ในหน่วยความจำ ทำให้เมื่อมีการใช้งานดังกล่าวอีกครั้งจะมีความเร็วมากกว่าเดิมมาก โดยโปรแกรมตรวจจับการบุกรุกทางเครือข่ายดังกล่าว ทางผู้พัฒนาได้พัฒนาเพื่อตรวจจับการโจมตีแบบ ping flood เท่านั้นทำให้การโจมตีนอกเหนือจากนี้ไม่สามารถนำมาวิเคราะห์แล้วแจ้งเตือนได้

7.2 ข้อเสนอแนะ

- ควรมีการพัฒนาโปรแกรมเพื่อตรวจจับการโจมตีในรูปแบบต่างๆให้มากขึ้น
- ควรมีการติดตั้งโปรแกรมทางด้านระบบตรวจจับการบุกรุกทางเครือข่าย เพื่อความปลอดภัยในการรับส่งข้อมูล ของเครือข่ายคอมพิวเตอร์
- ควรมีการตรวจเช็คอุปกรณ์ทางเครือข่ายให้อยู่ในโหมดให้มีความปลอดภัย ไม่ให้อยู่ในโหมดที่ง่ายต่อการโจมตี

เอกสารอ้างอิง

[1] สุวัฒน์ ปุณณชัยยะ , ต้น ตัณฑ์สุทธิวงศ์ , สุพจน์ ปุณณชัยยะ . **เปิดโลกของ TCP/IP และโปรโตคอลของอินเตอร์เนต** . กรุงเทพมหานคร : โปรวิชั่น , 2543.

[2] อภิชน ไวท์ยางกรู, อังสนา วงศ์รัตนวิจิตร . **"ระบบตรวจจับผู้บุกรุก"** ปริญญานิพนธ์วิศวกรรมศาสตร์บัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 2544.

[3] เรืองไกร รังสิตพล. **เจาะระบบ TCP/IP จุดอ่อนของระบบและวิธีป้องกัน . กรุงเทพมหานคร :** โปรวิชั่น , 2544.

[4] ยรกิจ สังฆโสภณ, มานัส ขนาดนิด, สิทธิราช ศรีโพธิ์ทอง. "**รายงานการวิเคราะห์และติดตั้งระบบเครือข่ายคอมพิวเตอร์ ภายในคณะวิศวกรรมศาสตร์** "ปริญญานิพนธ์วิศวกรรมศาสตร บัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร, 2545.

ภาคผนวก

ภาคผนวก

การติดตั้งโปรแกรม ตรวจจับการบุกรุกทางเครือข่ายบนระบบปฏิบัติการลีนุกซ์

เนื่องจาก โปรแกรมตรวจจับการบุกรุกทางเครือข่ายพัฒนาจากภาษาจาวา ดังนั้นจะต้องมีตัวคอมไพเลอร์ภาษาจาวา หรืออย่างน้อยต้องมีจาวารันไทม์ จึงจะสามารถที่จะทำงานได้นอกจากนี้โปรแกรมตรวจจับการบุกรุกทางเครือข่ายยังมีส่วนติดต่อกับฐานข้อมูล MySQL และการส่งเมล์เพื่อแจ้งแก่ผู้ดูแลระบบ นอกจากนี้ยังมีไลบรารี่ที่เป็นตัวกลางในการติดต่อระหว่างโปรแกรมกับส่วนหลักของระบบปฏิบัติการด้วย วิธีการติดตั้งเครื่องมือต่างๆ มีดังต่อไปนี้

1. ติดตั้ง Java 2 Platform ,Standard Edition JRE หรือ SDK
2. ติดตั้ง MySQL Server และ MySQL Connector/ J
3. ติดตั้งไลบรารี่ libpcap
4. ติดตั้งไลบรารี่ jpcap
5. ติดตั้ง Java Mail กับ JAF

การติดตั้ง Java 2 Platform ,Standard Edition

1. ดาวน์โหลดได้ที่ http://java.sun.com
2. ดาวน์โหลด jdk-1.5.0_02-linux-i586.bin หรือเวอร์ชันอื่นๆที่ต้องการ
3. รันคำสั่งดังต่อไปนี้เพื่อลงโปรแกรม

   ./jdk-1.5.0_02-linux-i586.bin

4. เซ็ต JAVA_HOME ไปยังที่ที่ลง j2dk ไว้

   export JAVA_HOME=/usr/local/jdk-1.5.0_02

หรือสร้างอีกวิธีหนึ่ง ดังนี้คือ

ln –s /usr/local/jdk-1.5.0_02 /usr/lib/java

export JAVA_HOME=/usr/lib/java

ติดตั้ง MySQL Server

1. ดาวน์โหลดได้ที่ http://www.mysql.com
2. ดาวน์โหลดเวอร์ชันที่ต้องการ ในที่นี้ใช้ mysql-standard-4.0.24-pc-linux-gnu-i686.tar.gz
3. ลง MySQL ดังนี้

3.1 สร้างกลุ่มใหม่ตั้งชื่อกลุ่ม mysql เพื่อแยกออกจากกลุ่มอื่น

groupadd mysql

3.2 สร้างผู้ใช้ใหม่ชื่อ mysql และอยู่ในกลุ่ม mysql

useradd –g mysql mysql

3.3 สมมติว่าเก็บ MySQL ที่ดาวน์โหลดมาไว้ที่นี่

cd /usr/local

3.4 ขยาย souce code ที่ดาวน์โหลดออกไว้ที่พาท /usr/local

unzip < /usr/local/mysql-standard-4.0.24-pc-linux-gnu-i686.tar.gz | tar xvf –

3.5 สร้างลิ้งค์ย่อจาก /usr/local/mysql-standard-4.0.24-pc-linux-gnu-i686 เป็น mysql

ln –s /usr/local/mysql-standard-4.0.24-pc-linux-gnu-i686 mysql

3.6 เข้าไปในไดเรกทอรี่ mysql ผ่านลิ้งค์ที่สร้างไว้

cd mysql

3.7 รันสคิปที่สร้างฐานข้อมูลพื้นฐานประกอบด้วย mysql กับ test

script/mysql_install_db

3.8 เปลี่ยนสิทธิให้เป็นของผู้ดูแลระบบ

chown –R root

3.9 เปลี่ยนไดเรกทอรี่ data ให้สิทธิเป็นของผู้ใช้ชื่อ mysql

chown –R mysql data

3.10 แล้วเปลี่ยนกลุ่มเป็นกลุ่ม mysql

chgrp –R mysql

3.11 สั่งให้ MySQL Server ให้บริการอยู่ในโหมด deamon

bin/safe_mysqld –user=mysql &

ติดตั้ง MySQL Connector/J

MySQL Connector/J เป็นคลาสที่ใช้ในการเขียนโปรแกรมภาษาจาวาติดต่อกับฐานข้อมูล MySQL

1. ดาวน์โหลดได้ที่ http://www.mysql.com
2. ดาวน์โหลดเวอร์ชันที่ต้องการ ในที่นี้ใช้ mysql-connector-java-3.0.16-ga.tar.gz
3. ขยายไฟล์ที่ดาวน์โหลดมา

   tar xvfz mysql-connector-java-3.0.16-ga.tar.gz

4. copy **mysql-connector-java-3.0.16-ga-bin.jar** ที่ได้จากการขยายไฟล์ไปที่ $JAVA_HOME/jre/lib/ext

   cp mysql-connector-java-3.0.16-ga-bin.jar $JAVA_HOME/jre/lib/ext

ติดตั้งไลบรารี่ libpcap

Libpcapเป็นไลบรารี่ที่ต้องใช้ในการเชื่อมตัวระหว่างโปรแกรมที่พัฒนาขึ้น กับส่วนหลักของระบบปฏิบัติการ โปรแกรมทางด้าน สนิฟเฟอร์ กับระบบตรวจจับการบุกรุก นิยมใช้มากที่สุด และใช้กันร้อยละ 99 ของโปรแกรมที่พัฒนาทางด้านวิเคราะห์แพจเก็จ และทราฟฟิกของเครือข่าย

1. ดาวน์โหลดที่ http://www.tcpdump.com
2. ดาวน์โหลดเวอร์ชันที่ต้องการ ในที่นี้ใช้ libpcap-0.8.3.tar.gz
3. ลงโปรแกรมดังนี้

   3.1 คลายไฟล์ที่บีบอัดออกมา

   tar xvfz libpcap-0.8.3.tar.gz

   3.2 เข้าไปในไดเรกทอรีซอร์สโค้ด

   cd libpcap-0.8.3

   3.3 เช็คว่ามีสภาวะแวดล้อมพร้อมทำงานหรือไม่

   ./configure

   3.4 ลงโปรแกรม

   make install

ติดตั้งไลบรารี่ jpcap

ไลบรารี่ jpcap เป็นคลาสที่ใช้ติดต่อระหว่างการพัฒนาโปรแกรมด้วยภาษาจาวากับไลบารี่ lipcap อีกทีเพราะการพัฒนาด้วยภาษาซีนั้นถึงแม้จะมีการทำงานที่รวดเร็วแต่ยังยึดติดกับเพลตฟอร์มการทำงานอยู่ ซึ่งการพัฒนาโดยใช้ภาษาจาวาจะเป็นการลดข้อด้อยดังกล่าว และการทำงานของภาษาจาวาในปัจจุบันก็มีความเร็วไม่ด้วยกว่าภาษาซีเลย นอกจากนี้ยังง่ายต่อการนำโปรแกรมที่เคยพัฒนาแล้วมาพัฒนาต่อโดยง่าย

1. ดาวน์โหลดได้ที่ http://netresearch.ics.uci.edu/
2. ดาวน์โหลดเวอร์ชันที่ใหม่ที่สุด ในที่นี้ใช้ jpcap-0.4.tar.gz
3. การลงไลบรารี่ jpcap จะมีความยุ่งยากเล็กน้อย เพราะโค้ดที่พัฒนาเป็นโค้ดที่พัฒนามาบนระบบปฏิบัติการ Solaris ของ SUN ดังนั้นจึงต้องมีการแก้ไข souce code ก่อนที่จะคอมไพล์ ดังนี้

   3.1 คลายไฟล์ต้นฉบับที่ถูกบีบอัดไว้

   tar xvfz jpcap-0.4.tar.gz

   3.2 เข้าไปในไดเรกทอรีของไฟล์ต้นฉบับ

   cd Jpcap/src/c

   3.3 เปิดไฟล์ Jpcap_sub.h เพื่อมาแก้ไข

   vi Jpcap_sub.h

   3.4 คอมเมนต์บรรทัด #define HAVA_SA_LEN เซฟแล้วออกจากไฟล์นั้น

   /* #define HAVE_SA_LEN */

   3.5 เปิดไฟล์ JpcapSender.c ขึ้นมาแก้ไข

   vi JpcapSender.c

   3.6 แก้ซอร์สโค้ดจาก closesocket(soc_num) เป็น close(soc_num) เพราะในลีนุกส์ไม่มีฟังก์ชัน closesocket() แต่จะมีฟังก์ชัน close() ในการปิดการติดต่อ socket เซฟแล้วออกจากตัวแก้ไขไฟล์

   %s/closesocket(soc_num)/close(soc_num)/g

   3.7 แก้ไข Make file ใหม่

   vi Makefile

3.8 ใส่คอมเมนต์หน้าบรรทัดที่มีข้อความดังนี้

JNI_INCLUDE2=$(JAVA_DIR)/include/solaris

#JNI_INCLUDE2=$(JAVA_DIR)/include/solaris

3.9 เปิดคอมเมนต์หน้าบรรทัดที่มีข้อความ JNI_INCLUDE2=$(JAVA_DIR)/include/linux

JNI_INCLUDE2=$(JAVA_DIR)/include/linux

3.10ใส่คอมเมนต์หน้าบรรทัดที่มีข้อความ COMPILE_OPTION = -G

#COMPILE_OPTION = -G

3.11 เปิดคอมเมนต์หน้าบรรทัดที่มีข้อความ COMPILE_OPTION = -shared

COMPILE_OPTION = -shared

3.12 สั่ง Make โปรแกรมดังนี้

make

3.13 ผลลัพธ์ที่ได้จะได้ไลบรารี่ libjpcap.so ออกมา ให้นำไปไว้ที่ $JAVA_HOME/jre/lib/<arch>

cp libjpcap.so $JAVA_HOME/jre/lib/i386

3.14 จากนั้นคัดลอกไฟล์ jpcap.jar ในไดเรกทอรี lib ไปเก็บไว้ที่ $JAVA_HOME/jre/lib/ext

cp ../../lib/jpcap.jar $JAVA_HOME/jre/lib/ext/

ติดตั้ง Java Mail

1. ดาวน์โหลดได้ที่ http://java.sun.com
2. ดาวน์โหลดเวอร์ชันที่ต้องการ ในที่นี้ใช้ javamail-1_3_2.zip
3. คลายไฟล์ต้นฉบับที่บีบอัดออกมา

unzip javamail-1.3.2.zip

4. เข้าไปคัดลอกไฟล์ mail.jar ในไดเรกทอรีที่คลายได้ไปเก็บที่ $JAVA_HOME/jre/lib/ext/

cd javamail-1.3.2

cp mail.jar $JAVA_HOME/jre/lib/exe

ติดตั้ง JAF

1. ดาวน์โหลดได้ที่ http://java.sun.com
2. ดาวน์โหลดเวอร์ชันที่ต้องการ ในที่นี้ใช้ jaf-1_0_2-upd.zip
3. คลายไฟล์ต้นฉบับที่บีบอัดออกมา

   unzip jaf-1_0_2-upd.zip
4. เข้าไปคัดลอกไฟล์ activation.jar ในไดเรกทอรีที่คลายได้ ไปเก็บที่ $JAVA_HOME/jre/lib/ext/

   cd jaf-1_0_2-upd

   cp activation.jar $JAVA_HOME/jre/lib/ext/

ประวัติผู้เขียนโครงงาน

ชื่อ: ณพงศ์ กิ่งเกล้า

ภูมิลำเนา : 30 หมู่ 3 ต.ท่าสัก อ.พิชัย จ. อุตรดิตถ์ 53220

ประวัติการศึกษา: จบมัธยมศึกษาตอนต้นจากโรงเรียนอุตรดิตถ์

จบมัธยมศึกษาตอนปลายจากโรงเรียนอุตรดิตถ์

ปัจจุบันศึกษาอยู่ที่คณะวิศวกรรมศาสตร์สาขา วิศวกรรมคอมพิวเตอร์ ชั้นปีที่4

มหาวิทยาลัยนเรศวร

E-Mail: pongk_cpe@hotmail.com

ชื่อ: ธวัชชัย สิงงาม

ภูมิลำเนา: 24 หมู่ 7 ต.นาหนองไผ่ อ.ชุมพลบุรี จ.สุรินทร์ 32190

ประวัติการศึกษา: จบมัธยมศึกษาตอนต้นจากโรงเรียนบ้านคู่นาหนองไผ่

จบมัธยมศึกษาตอนปลายจากโรงเรียนท่าตูมประชาเสริมวิทย์

ปัจจุบันศึกษาอยู่ที่คณะวิศวกรรมศาสตร์สาขา วิศวกรรมคอมพิวเตอร์ ชั้นปีที่4

มหาวิทยาลัยนเรศวร

E-Mail: orcaxs@hotmail.com

ชื่อ: ไพลิน สนานคุณ

ภูมิลำเนา: 299/53 ถ.สวรรค์วิถี ต.ปากน้ำโพ อ.เมือง จ.นครสวรรค์ 60000

ประวัติการศึกษา: จบมัธยมศึกษาตอนต้นจากโรงเรียนสตรีนครสวรรค์

จบมัธยมศึกษาตอนปลายจากโรงเรียนสตรีนครสวรรค์

ปัจจุบันศึกษาอยู่ที่คณะวิศวกรรมศาสตร์สาขา วิศวกรรมคอมพิวเตอร์ ชั้นปีที่4

มหาวิทยาลัยนเรศวร

E-Mail: aomaom99naja@hotmail.com