ระบบตรวจจับผู้บุกรุกชาญฉลาด

INTELLIGENT INTRUSION DETECTION SYSTEM

ภูริชญ์ ลิขิตวทัญญู
สัจพงศ์ กาญจนรังษี

รายงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2550

ปริญญานิพนธ์ปีการศึกษา 2550

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบตรวจจับผู้บุกรุกชาญฉลาด

Intelligent Intrusion Detection System

ผู้จัดทำ

1. นายภูริชญ์ ลิขิตวทัญญู รหัสนักศึกษา 47010577
2. นายสัจพงศ์ กาญจนรังษี รหัสนักศึกษา 47010823

อาจารย์ที่ปรึกษา

(อาจารย์ธนัญชัย ตรีภาค)

อาจารย์ที่ปรึกษาร่วม

(อาจารย์อำนาจ ขาวเน)

ระบบตรวจจับผู้บุกรุกเครือข่ายชาญฉลาด

นายภูริชญ์ ลิขิตวทัญญู รหัสประจำตัว 47010577
นายสัจพงศ์ กาญจนรังษี รหัสประจำตัว 47010823
อาจารย์ธนัญชัย ตรีภาค อาจารย์ที่ปรึกษา
อาจารย์อำนาจ ขาวเน อาจารย์ที่ปรึกษาร่วม
ปีการศึกษา 2550

บทคัดย่อ

ความปลอดภัยนับเป็นปัจจัยสำคัญอย่างหนึ่งในการใช้งานคอมพิวเตอร์ที่มีการเชื่อมต่อเป็นเครือข่ายอย่างในปัจจุบัน เนื่องจากมีผู้ต้องการบุกรุกเครื่องคอมพิวเตอร์ผ่านทางเครือข่ายมากขึ้น ระบบตรวจจับผู้บุกรุกชาญฉลาดที่พัฒนาขึ้นเป็นระบบตรวจจับผู้บุกรุกที่มุ่งเน้นความสามารถในการตรวจจับโจมตีเพื่อให้ปิดบริการ โดยใช้ความสามารถของระบบปัญญาประดิษฐ์ โครงข่ายใยประสาทเทียมเพื่อวิเคราะห์ข้อมูลในเครือข่าย เพื่อตรวจสอบว่าข้อมูลนั้นเข้าข่ายการโจมตีระบบเครือข่ายหรือไม่ โดยในขั้นตอนการออกแบบโครงสร้างของระบบทั้งหมด ได้มีการศึกษาถึงรูปแบบการโจมตีที่มีผลต่อการทำงานของชั้นทีซีพี/ไอพี ผลที่ได้จากการศึกษานำไปใช้เป็นข้อมูลการโจมตีเพื่อให้โครงข่ายใยประสาทเทียมเรียนรู้ และการศึกษาเกี่ยวกับระบบโครงข่ายใยประสาทเทียม ผลที่ได้จากการศึกษานำไปใช้ในการวิเคราห์ข้อมูลว่าเป็นการโจมตีจริงหรือไม่ ซึ่งระบบที่กล่าวมาแล้วข้างต้นทำให้ผู้ดูแลระบบสามารถตรวจสอบการโจมตีที่เกิดขึ้นในระบบที่ตนดูแลอยู่ได้

Intelligent Intrusion Detection System

Mr. Puritch	Likhitwatanyoo	47010577
Mr. Sajapong	Karnjanarungsee	47010823
Mr. Thanunchai	Threepak	Advisor
Mr. Amnach	Kawne	Co-Advisor

ABSTACT

Security is one of the most important topics when using computers on network because there are many attacks via computer network. Intelligence Intrusion Detection System is developed for detecting Denial of Service (DoS) attack by using artificial intelligent such as Artificial Neural Networks (ANN). This project's design has been started by study of attacks that effects to TCP/IP layer, anomaly data that used in ANN, and for analysis network's data. Thus, this project provided an Intelligent Intrusion Detection System in order to monitor and analyze these attacks.

กิตติกรรมประกาศ

โครงงานฉบับนี้สำเร็จได้ด้วยดี เนื่องจากได้รับการแนะนำ สนับสนุน และให้คำปรึกษาเป็นอย่างดีจาก อาจารย์ธนัญชัย ตรีภาค และอาจารย์อำนาจ ขาวเน อาจารย์ที่ปรึกษาโครงงาน ซึ่งต้องขอขอบพระคุณเป็นอย่างสูงที่ท่านช่วยแนะนำ ให้คำปรึกษา อบรม กระทั่งสุดท้ายสามารถสำเร็จลุล่วงไปได้ด้วยดี

รวมทั้งอาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกท่าน ที่ให้การอบรมสั่งสอนวิชาความรู้แก่คณะผู้จัดทำมาโดยตลอด

ขอขอบพระคุณ บิดา มารดา ผู้เป็นที่เคารพรักยิ่งของคณะผู้จัดทำเป็นอย่างสูงที่ทำให้คณะผู้จัดทำมีวันนี้ ซึ่งท่านให้การอบรมสั่งสอน เลี้ยงดู และให้โอกาสในการศึกษา จึงขอกราบขอบพระคุณมา ณ ที่นี้

ขอขอบคุณ ณัชชา นิยมธรรมกิจ ที่ช่วยแนะนำ ร่วมกันคิด ร่วมให้คำปรึกษา ในส่วนของการดักจับข้อมูล วาทิตย์ วงวาทิน และ วรชัย กึกก้อง ที่ช่วยในเรื่องของคำปรึกษาแปลกๆใหม่ๆ คอยช่วยเหลือในยามจำเป็น นอกจากนี้ขอขอบคุณเพื่อนๆ ทุกท่านทั้งในห้องเน็ทเวอร์กและห้องอื่นที่เอ่ยนามมาแล้ว และไม่ได้เอ่ยนาม ในความช่วยเหลือในด้านการผ่อนคลาย อาหารการกิน ข้อคิดเห็นและเป็นกำลังใจตลอดมา ขอขอบคุณห้องวิจัยเน็ทเวอร์กที่ให้สถานที่ในการทำงานโปรเจค และใช้ในงานอื่นๆ

สุดท้ายนี้ขอขอบพระคุณผู้ดูแลระบบคอมพิวเตอร์ภาควิชาวิศวกรรมศาสตร์ และสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่อำนวยความสะดวกในการใช้งานเครือข่าย

นายภูริชญ์ ลิขิตวทัญญู
นายสัจพงศ์ กาญจนรังษี

สารบัญ

สารบัญ (ต่อ)

สารบัญ (ต่อ)

สารบัญตาราง

สารบัญรูป

สารบัญรูป (ต่อ)

บทที่ 1
บทนำ

1.1 ความสำคัญและที่มา

การนำคอมพิวเตอร์มาให้บริการเพื่อความสะดวกในชีวิตประจำวัน อาจมีผู้ไม่ประสงค์ดีต้องการที่จะบุกรุกทำลายการให้บริการการให้บริการดังกล่าว

รูปแบบการโจมตีเครือข่ายมีวิธีหลากหลายวิธีการ การโจมตีรูปแบบหนึ่งที่ไม่ได้สร้างความเสียหายให้กับระบบแต่เป็นการค้นหาช่องทางในการโจมตีหรือเจาะเข้าระบบคือการสแกนพอร์ต (Port Scan) นอกจากนี้การโจมตีอีกวิธีหนึ่งที่ทำให้เกิดการผิดปรกติกับระบบเครือข่ายโดยไม่ต้องคำนึงถึงระบบปริบัติการที่ใช้ สามารถทำให้ระบบไม่สามารถให้บริการได้ นั่นก็คือการโจมตีเพื่อให้ปิดบริการ(Denial of Service หรือ DoS)

โดยทั่วไปการเชื่อมต่อจะกระทำผ่านพอร์ตทีซีพี/ไอพี และพอร์ตทีซีพี/ไอพีก็จะเชื่อมต่อกับบริการ(Service) การสร้างการเชื่อมต่อแต่ละครั้งมีการเปิดช่องทางและต้องใช้ทรัพยากรของระบบ เช่น หน่วยความจำที่ทำหน้าที่เป็นบัฟเฟอร์ หรือความสามารถของซีพียู รวมทั้งแบนด์วิดของเครือข่าย โดยในสภาวะปกติแล้วระบบสามารถจัดหาทรัพยากรเหล่านี้ได้อย่างพอเพียง แต่หากอยู่ในสภาวะที่ไม่ปกติ เช่น มีการโจมตีบริการจะส่งผลให้ทรัพยากรที่มีอยู่ถูกใช้ไปอย่างรวดเร็ว และส่งผลระบบไม่สามารถให้บริการต่อไปได้เนื่องจากผลของการโจมตีดังกล่าวมีสูงมาก นอกจากนี้ช่องทางที่เปิดอยู่อาจทำให้เกิดการโจมตีได้ง่าย ทำให้จำเป็นต้องมีการตรวจสอบข้อมูลในเครือข่ายอยู่ตลอดเวลาว่ามีการโจมตีหรือไม่ ทำให้เกิดแนวคิดการทำระบบตรวจจับผู้บุกรุกทางเครือข่ายคอมพิวเตอร์(Network Intrusion Detection System : NIDS) ขึ้นมา เพื่อตรวจสอบการบุกรุกทางเครือข่ายและทำการแจ้งเตือนไปยังผู้ดูแลระบบ และเก็บข้อมูลการโจมตีดังกล่าวไว้ในล็อกไฟล์ เพื่อใช้ตรวจสอบได้ในภายหลัง

และเนื่องจากที่ผ่านมา ระบบตรวจจับผู้บุกรุกจะใช้กฎซึ่งระบุการโจมตีเอาไว้ว่าข้อมูลใดเป็นการโจมตีและข้อมูลใดเป็นข้อมูลปกติ ซึ่งระบบดังกล่าวไม่สามารถพลิกแพลงได้เนื่องจากต้องอ้างอิงจากกฎอย่างตายตัว ดังนั้นจึงทำให้เกิดแนวความคิดที่จะใช้ระบบปัญญาประดิษฐ์(Artificial Intelligent) มาใช้วิเคราะห์ข้อมูลว่าข้อมูลนั้นเป็นการโจมตีหรือไม่ หรือเกิดการเปลี่ยนแปลงจากสภาวะปกติซึ่งอาจเป็นการโจมตีได้

โครงงานนี้มุ่งเน้นการศึกษาประเภทต่างๆของการโจมตีเพื่อให้ปิดบริการโดยการจัดแบ่งประเภทของการโจมตี รวมทั้งศึกษาแนวทางการตรวจสอบการโจมตีในแต่ละประเภท เพื่อพัฒนาระบบตรวจจับบนระบบปฏิบัติการวินโดว์ให้สามารถตรวจจับการโจมตีดังกล่าวได้อย่างมีประสิทธิภาพ

1.2 วัตถุประสงค์ของโครงงาน

โครงงานที่จัดทำขึ้นนี้ จัดทำภายใต้วัตถุประสงค์หลัก 4 ประการ ได้แก่

(1) เพื่อศึกษารายละเอียดและการทำงานของการบุกรุกทางเครือข่ายคอมพิวเตอร์

(2) เพื่อศึกษาแนวทางการตรวจสอบการบุกรุกทางเครือข่ายคอมพิวเตอร์

(3) เพื่อศึกษารายละเอียดและการทำงานของโครงข่ายใยประสาทเทียม

(4) เพื่อสร้างระบบตรวจจับผู้บุกรุกทางเครือข่ายคอมพิวเตอร์โดยใช้โครงข่ายใยประสาทเทียมในการตรวจสอบ

1.3 ขอบเขตของโครงงาน

ขอบเขตการทำงานของโครงงานนี้ ได้แก่

(1) ตรวจดูความผิดปรกติในส่วนเฮดเดอร์ของ ไอพี, ทีซีพี, ยูดีพี และ ไอซีเอ็มพี และมีการเก็บข้อมูลแพ็กเก็ตเชิงปริมาณเพื่อนำมาใช้ในการวิเคราะห์การโจมตี

(2) ใช้ โครงข่ายใยประสาทเทียมในการวิเคราะห์การโจมตีเพื่อปิดบริการและการสแกนพอร์ต

(3) พัฒนาระบบตรวจจับการบุกรุกทางเครือข่ายบนระบบปฏิบัติการวินโดวส์

(4) ระบบที่สร้างขึ้นต้องสามารถตรวจจับ และเก็บข้อมูลที่เกิดการบุกรุกตามเวลาจริงได้

1.4 ขั้นตอนการดำเนินงาน

(1) ศึกษารายละเอียดเกี่ยวกับทีซีพี/ไอพีเบื้องต้น

(2) ศึกษาเกี่ยวกับระบบตรวจจับผู้บุกรุกทางเครือข่ายคอมพิวเตอร์

(3) ศึกษาเกี่ยวกับระบบโครงข่ายใยประสาทเทียม

(4) ศึกษารายละเอียดและการทำงานของการการสแกนพอร์ตและโจมตีเพื่อให้ปิดบริการสำหรับสแต็กทีซีพี/ไอพี

(5) หาข้อมูลที่สามารถเป็นอินพุตให้กับโครงข่ายใยประสาทเทียมแล้วสามารถบอกถึงการโจมตีได้

(6) ออกแบบสถาปัตยกรรมของโปรแกรมโดยรวม

(7) ออกแบบโครงสร้างของโครงข่ายใยประสาทเทียม

(8) ออกแบบโครงสร้างและขั้นตอนการทำงานของระบบตรวจจับผู้บุกรุกทางเครือข่ายคอมพิวเตอร์

(9) เขียนโปรแกรมตรวจจับผู้บุกรุกทางเครือข่ายพร้อมทดสอบการทำงาน

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

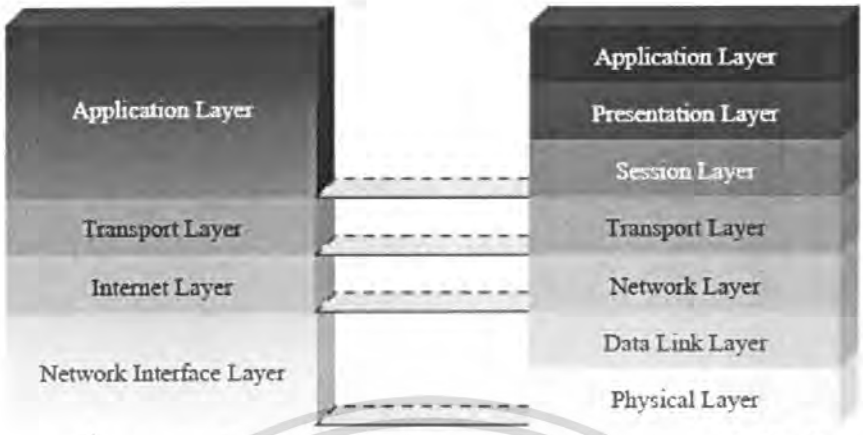
2.1 โพรโตคอลทีซีพี/ไอพี

2.1.1 ความเป็นมาของโพรโตคอลทีซีพี/ไอพี

ทีซีพี/ไอพี เป็นมาตรฐานการรับส่งข้อมูลระหว่างคอมพิวเตอร์สองระบบ ที่มีขึ้นเมื่อกระทรวงกลาโหมสหรัฐฯ หรือ Department of Defense (DOD) ทำการทดลองในปี ค.ศ.1969 เชื่อมโยงคอมพิวเตอร์ทางทหารของแต่ละหน่วย ซึ่งเป็นคอมพิวเตอร์ต่างชนิดกันให้สามารถติดต่อรับส่งข้อมูลกันได้ โครงการนี้มีชื่อว่า Advanced Research Projects Agency Network หรือ ARPANET ซอฟต์แวร์ที่ใช้ควบคุมการรับส่งข้อมูลของ ARPANET ประกอบด้วยส่วนหลักๆ 2 ส่วน คือ ทีซีพี (Transmission Control Protocol หรือ TCP) และ ไอพี (Internet Protocol หรือ IP) ซึ่ง ทีซีพี มีหน้าที่ตรวจสอบการรับส่งข้อมูลระหว่างคอมพิวเตอร์ผู้รับและผู้ส่ง ให้ได้รับข้อมูลถูกต้องครบถ้วน ส่วนไอพีจะมีหน้าที่เลือกเส้นทางที่ใช้รับส่งข้อมูลผ่านระบบเครือข่าย และตรวจสอบที่แอดเดรสของผู้รับ เรียกว่าไอพีแอดเดรส (IP Address) ต่อมาในปี ค.ศ.1983 ทีซีพี/ไอพีถูกกำหนดให้เป็นมาตรฐานการรับส่งข้อมูลของกระทรวงกลาโหมสหรัฐฯ และได้รวมเป็นส่วนหนึ่งของระบบปฏิบัติการยูนิกซ์ ส่งผลให้มีการใช้งานกันอย่างกว้างขวาง ในปัจจุบันใช้งานอยู่ในแทบทุกเครือข่าย ไม่ว่าจะเป็นเครือข่ายเฉพาะที่หรือเครือข่ายในบริเวณกว้าง ทีซีพี/ไอพีเชื่อมกลุ่มเครือข่ายย่อยเข้าด้วยกันเป็นเครือข่ายขนาดใหญ่ หรือ อินเทอร์เน็ต

2.1.2 การเชื่อมต่อของโพรโตคอลทีซีพี/ไอพี (TCP/IP Linking)

ทีซีพี/ไอพี (TCP/IP หรือ Transmission Control Protocol/Internet Protocol) เป็นโพรโตคอลในการสื่อสารในระบบอินเทอร์เน็ตและอินทราเน็ต การทำงานของทีซีพี/ไอพีสามารถเปรียบเทียบกับโมเดลอ้างอิงโอเอสไอ (Open System Interconnection Reference Model: OSI) ตามมาตรฐานไอเอสโอ (International Organization for Standardization: ISO) ได้ดังรูปที่ 2.1



รูปที่ 2.1 การเปรียบเทียบเลเยอร์ของโอเอสไอกับเลเยอร์ของทีซีพี/ไอพี

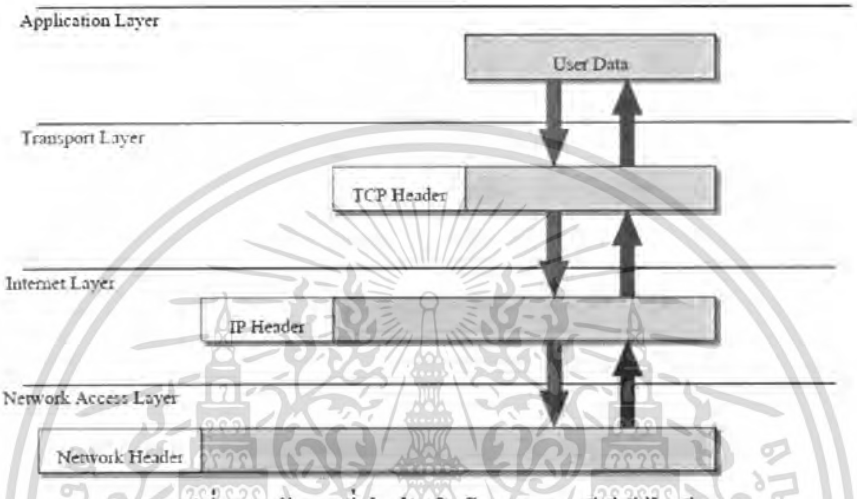
ในแต่ละระดับชั้นของทีซีพี/ไอพีมีการทำงานที่แตกต่างกัน ตั้งแต่การติดต่อกับแอพพลิเคชันจนกระทั่งแปลงเป็นสัญญาณส่งไปตามสายสัญญาณ ซึ่งการทำงานในแต่ละระดับชั้นของทีซีพี/ไอพี มีดังตารางที่ 2.1

ตารางที่ 2.1 การทำงานของแต่ละระดับชั้นของทีซีพี/ไอพี

ชื่อระดับชั้น	หน้าที่
1. ชั้นแอพพลิเคชัน (Application Layer)	ชั้นนี้รองรับการทำงานของแอพพลิเคชันต่างๆ ที่ทำงานเป็นโพรเซสอยู่ในเครื่องต้นทางและปลายทาง โดยจัดการเชื่อมต่อระหว่างโพรเซส หรือแอพพลิเคชันที่อยู่ต่างเครื่องกัน โดยการทำงานของแอพพลิเคชันต่างๆมีการติดต่อกันตามแต่ละโพรโตคอลเฉพาะแล้วแต่แอพพลิเคชันที่ใช้งาน ซึ่งจะขอบริการจากชั้นทรานสปอร์ตอีกทีหนึ่ง
2. ชั้นทรานสปอร์ต (Transport Layer)	มีการสร้างการเชื่อมต่อกันระหว่างแอพพลิเคชันแบบ end-to-end โดยจุดที่เชื่อมต่อกันเพื่อรับส่งข้อมูลนี้เรียกว่า พอร์ต (port) หรือซ็อกเก็ต (Socket)ในชั้นนี้มีบริการหลักอยู่ 2 แบบ คือ Connection Oriented โดยเรียกผ่านโพรโตคอลทีซีพี (TCP: Transmission Control Protocol) และ Connectionless ซึ่งเรียกผ่านโพรโตคอลยูดีพี (UDP: User Datagram Protocol) ซึ่งกล่าวถึงในหัวข้อถัดไป
3. ชั้นอินเทอร์เน็ต (Internet Layer)	ชั้นนี้มีหน้านี้ส่งผ่านข้อมูลระหว่างเครือข่าย โดยมีโพรโตคอลที่ทำงานเป็นกลไกสำคัญในการส่งผ่านข้อมูลไปยังเครือข่ายใดๆ ในอินเทอร์เน็ต คือ ไอพี (Internet Protocol: IP) ซึ่งกล่าวถึงในหัวข้อถัดไป นอกจากนี้ในชั้นนี้ยังมีโพรโตคอลทำงานอยู่ด้วยอีก 2 ชนิด คือ ไอซีเอ็มพี (Internet Control Message Protocol: ICMP) และเออาร์พี (Address Resolution Protocol: ARP)

ตารางที่ 2.1 (ต่อ)

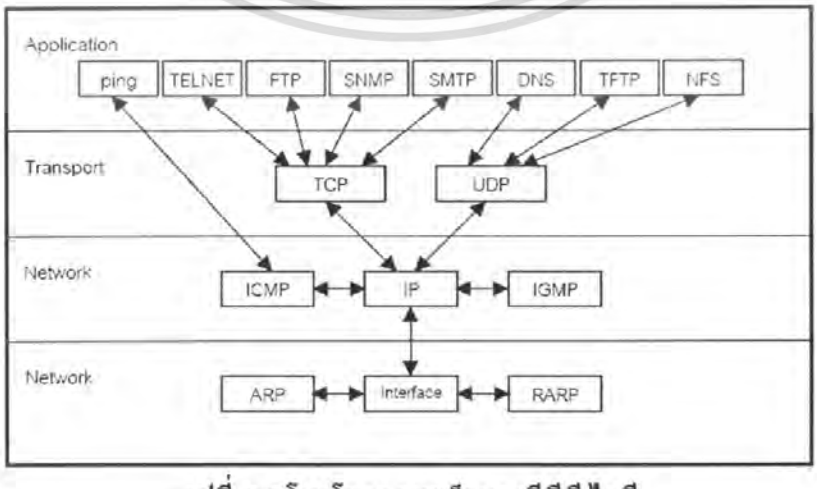
4. ชั้นเน็ตเวิร์กอินเตอร์เฟส (Network Interface Layer)	ทำหน้าที่ในการแปลงข้อมูลให้อยู่ในรูปที่เหมาะสมกับเครือข่ายแต่ละแบบ ซึ่งแตกต่างกันออกไป และแปลงเป็นสัญญาณไฟฟ้าส่งไปยังเครือข่าย



รูปที่ 2.2 ข้อมูลที่ส่งผ่านในโมเดลของทีซีพี/ไอพี

2.1.3 โพรโตคอลสแต็ก

การทำงานตามโปรแกรมประยุกต์หนึ่งๆไม่ได้ใช้โพรโตคอลพร้อมกันทั้งชุด หากแต่ใช้เพียงโพรโตคอลที่สัมพันธ์กันไปในแต่ละระดับชั้นของแบบอ้างอิง ตัวอย่างเช่นการใช้งานเทลเน็ต (Telnet) จะอาศัยทีซีพีและไอพี ตามลำดับ การซ้อนทับของโพรโตคอลจากระดับชั้นบนไปชั้นล่างเรียกว่าโพรโตคอลแสตค (Protocol Stack) ดังรูปที่ 2.3



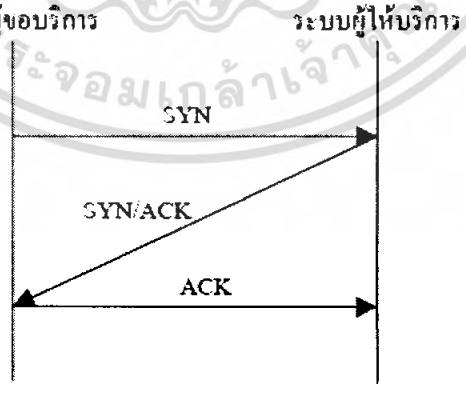
รูปที่ 2.3 โพรโตคอลสสต็กของทีซีพี/ไอพี

ไอพีซึ่งอยู่ในระดับชั้นเน็ตเวอร์คตามรูป เป็นแกนสำคัญของ โพรโตคอลแสตค เนื่องจากทั้ง ทีซีพี และ ยูดีพี ต้องใช้ไอพีเพื่อเลือกเส้นทางส่งแพ็กเก็ต ในระดับชั้นเน็ตเวอร์คยังมีไอซีเอ็มพีสนับสนุนการทำงานของไอพีเพื่อรายงานข้อผิดปกติที่เกิดขึ้นเนื่องจากการส่งแพ็กเก็ต และมีไอจีเอ็มพีดูแลการจัดกลุ่มโฮสต์ในเครือข่ายมัลติคาสต์ ระดับชั้นทรานสปอร์ตมี 2 โพรโตคอล ที่สำคัญ คือ ทีซีพีและยูดีพี แอพลิเคชันจะเลือกใช้ทีซีพีหรือยูดีพีตามลักษณะงาน โพรโตคอลระดับล่างถัดจากไอพีได้แก่ โพรโตคอล ระดับเน็ตเวอร์คอินเตอร์เฟสซึ่งกำหนดการทำงานตามเทคโนโลยีเครือข่ายที่ใช้งาน ในระดับชั้นนี้มี โพรโตคอล

ในชุดของ ทีซีพี/ไอพี ทำหน้าที่สนับสนุนการทำงานอยู่สอง โพรโตคอล คือ เออาร์พี และอาร์เออาร์พี ทั้งสองโพรโตคอลทำหน้าที่แปลงค่าระหว่างแอดเดรสไอพี กับ ฮาร์ดแวร์แอดเดรสในชุดโพรโตคอลทีซีพี/ไอพีนี้ มีโพรโตคอลหลักที่ขอกล่าวถึง 5 โพรโตคอล ได้แก่ โพรโตคอลทีซีพี โพรโตคอลยูดีพี ซึ่งทำงานในชั้นทรานสปอร์ต และโพรโตคอลไอพี โพรโตคอลเออาร์พี โพรโตคอลไอซีเอ็มพี ซึ่งทำงานในชั้นอินเทอร์เน็ต โดยมีรายละเอียดดังต่อไปนี้

2.1.3.1 โพรโตคอลทีซีพี (TCP: Transmission Control Protocol)

การทำงานที่สำคัญอย่างหนึ่งของโพรโตคอลทีซีพี คือ การทำ 3-way Handshake ซึ่งเป็นกระบวนการเริ่มต้นในการสร้างการเชื่อมต่อในชั้นทรานสปอร์ต กล่าวคือ ในการติดต่อกันระหว่างระบบในเครือข่ายต้องมีการสร้างการเชื่อมต่อไปยังระบบที่ให้บริการก่อน โดยผู้ขอบริการส่งสัญญาณ SYN เพื่อขอบริการ จากนั้นผู้ให้บริการจะส่งสัญญาณ ACK เพื่อตอบรับการเชื่อมต่อที่ร้องขอมาจึงสามารถรับส่งข้อมูลกันได้ ดังรูปที่ 2.4



รูปที่ 2.4 การทำ 3-way Handshake

การเชื่อมต่อแบบ 3-way handshake นี้ เป็นการตรวจสอบความพร้อมของทั้งฝ่ายส่งและฝ่ายรับ และการกำหนดค่าเริ่มต้นของพารามิเตอร์ต่างๆ ของทั้งสองฝ่ายให้ตรงกัน หลังจากกระบวนการทำ 3-way handshake สิ้นสุด ทั้งสองฝ่ายจึงสามารถรับและส่งข้อมูลซึ่งกันและกันได้

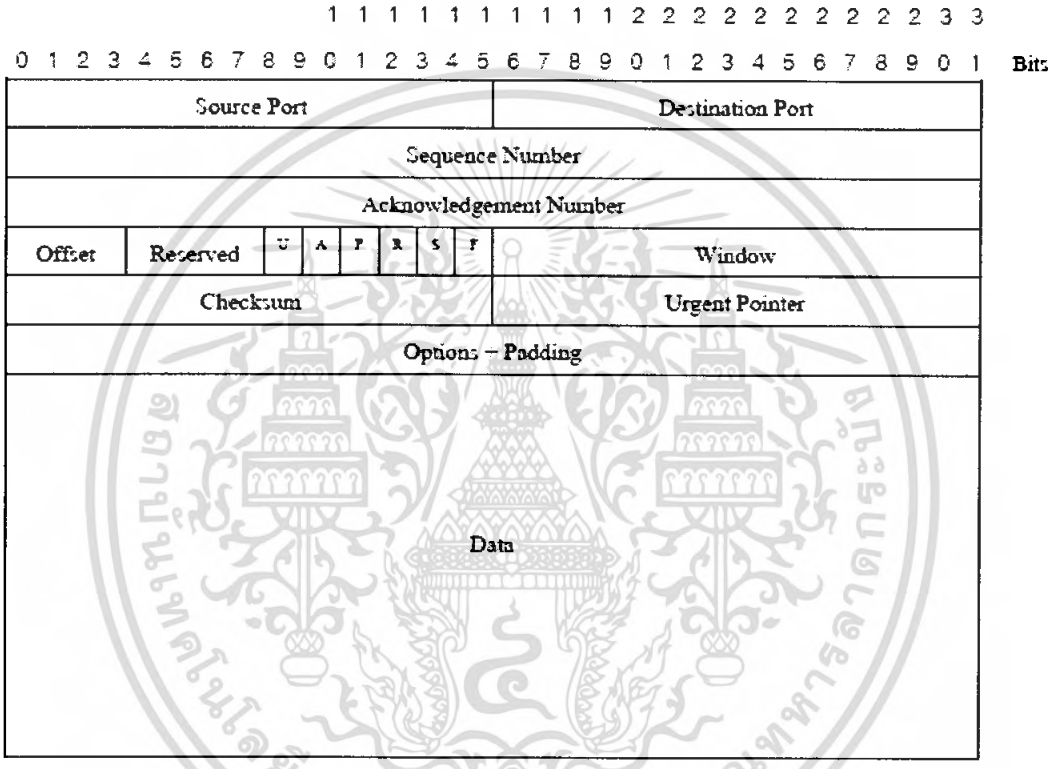
ดังนั้นโพรโตคอลทีซีพีจึงเป็นโพรโตคอลที่มีการรับส่งข้อมูลแบบ "Connection Oriented" ทำให้การทำงานของทีซีพีมีความน่าเชื่อถือมากขึ้น หน้าที่การทำงานของทีซีพีในการรับส่งข้อมูลมีหน้าที่หลัก 6 ข้อคือ

1. ควบคุมการรับส่งข้อมูล (Basic Data Transfer)
2. ความน่าเชื่อถือในการรับส่งข้อมูล (Reliability)
3. ควบคุมการไหลของข้อมูล (Flow Control)
4. การทำมัลติเพล็ก (Multiplexing)
5. ควบคุมการเชื่อมต่อ (Connection)
6. ความปลอดภัยในการรับส่งข้อมูล (Security)

<u>ส่วนประกอบของทีซีพีเฮดเดอร์</u>

1. Source Port : เป็นหมายเลขพอร์ตของบริการที่เครื่องต้นทาง
2. Destination Port : เป็นหมายเลขพอร์ตของบริการเครื่องปลายทาง
3. Sequence Number : เป็นหมายเลขที่บอกลำดับของการรับส่งข้อมูลของเครื่องที่ต้องการขอส่งข้อมูล
4. Acknowledgement Number : เป็นหมายเลขที่บอกลำดับของการรับส่งข้อมูลที่ฝั่งรับข้อมูลปกติ ค่าของ Acknowledgement Number มีค่าเท่ากับ Sequence Number (ของอีกฝั่งหนึ่ง) + 1 เสมอ
5. Data Offset : เป็นตัวบอกค่าออฟเซตของข้อมูล เพราะทีซีพีนั้นไม่มีการกำหนดความยาวที่แน่นอนของข้อมูล จึงต้องมีออฟเซตเป็นตัวบอก
6. Flag : เป็นบิตที่บอกชนิดของข้อมูล ได้แก่
   - URG: Urgent Pointer Field Significant - แสดง Urgent Pointer
   - ACK: Acknowledgement Field Significant – แสดงการ Acknowledgement
   - PSH: Push Function
   - RST: Reset The Connection - แสดงเมื่อรีเซ็ทการเชื่อมต่อ
   - SYN: Synchronize Sequence Number - หมายเลขแพ็กเก็ตที่ส่งแบบซิงโครนัส
   - FIN: No more data from sender – แสดงว่าไม่มีข้อมูลที่ส่งจากผู้ส่งแล้ว
7. Window : เป็นเลขบอกจำนวนของอ็อกเต็ต (octet) ของข้อมูล จัดการในส่วนของ end-to-end flow control

8. Checksum : เป็นส่วนที่ตรวจสอบความถูกต้องของข้อมูล
9. Urgent Pointer : เป็นตัวชี้ตำแหน่งของ Urgent Data
10. Option and Padding : เป็นตัวบอกออปชันของโพรเซสที่ใช้ทีซีพี
11. Data : เนื้อข้อมูลที่ต้องการสื่อสาร มีขนาดได้ไม่ต่ำกว่า 5 32-บิตเวิร์ด (6 บิตแรกสงวนไว้และกำหนดให้เป็นศูนย์)



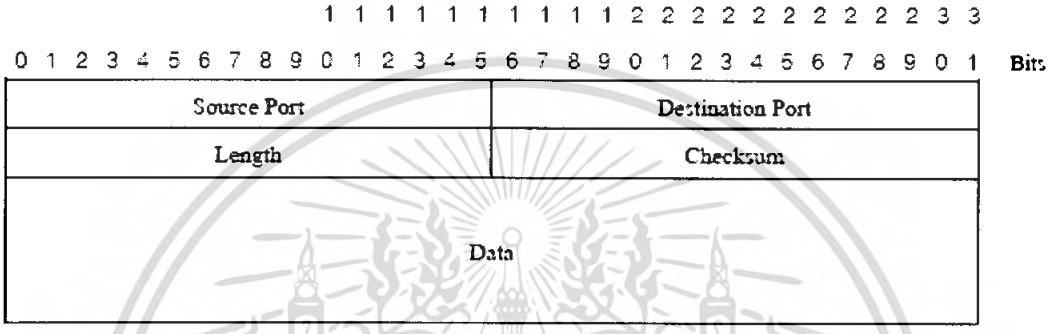
รูปที่ 2.5 แพ็กเก็ตทีซีพี

2.1.3.2 โพรโตคอลยูดีพี (UDP: User Datagram Protocol)

โพรโตคอลยูดีพีเป็นโพรโตคอลในการติดต่อสื่อสารในชั้นทรานสปอร์ต (Transport Layer) การทำงานคล้ายกับทีซีพีมาก คือ จัดการเกี่ยวกับการสื่อสารระหว่างเครื่อง แต่เป็นแบบ Connectionless คือทั้งฝ่ายส่งและฝ่ายรับไม่จำเป็นต้องอาศัยการสร้างช่องทางเชื่อมต่อกัน โดยไม่ต้องมีการแจ้งให้ฝ่ายรับข้อมูลเตรียมรับข้อมูลเหมือนโพรโตคอลทีซีพี และไม่มีการส่งสัญญาณตรวจสอบว่าข้อมูลถึงเครื่องปลายทางอย่างถูกต้องครบถ้วนในการส่งข้อมูลแต่ละครั้ง จึงไม่มีการส่งข้อมูลใหม่อีกในกรณีที่เกิดความผิดพลาดของการส่งข้อมูล

ส่วนประกอบของ UDP Frame

1. Source Port: เป็นค่าตัวเลข 16 บิต บอกพอร์ตของบริการที่เครื่องต้นทาง
2. Destination Port: เป็นค่าตัวเลข 16 บิต บอกพอร์ตของบริการที่เครื่องปลายทาง
3. Length: เป็นค่าตัวเลข 16 บิต บอกความยาวของข้อมูล
4. Checksum: เป็นค่าตัวเลข 16 บิต ตรวจสอบความถูกต้องของข้อมูลที่ส่ง



รูปที่ 2.6 แพ็กเก็ตยูดีพี

2.1.3.3 โพรโตคอลไอพี (IP: Internet Protocol)

โพรโตคอลไอพีเป็นโพรโตคอลที่จัดการเกี่ยวกับแอดเดรสของแต่ละแพ็กเก็ต เพื่อให้ส่งแพ็กเก็ตต่างๆ ไปยังเป้าหมายได้ถูกต้อง การทำงานของไอพีเป็นเพียงการส่งข้อมูลไปยังเครื่องเป้าหมายเท่านั้นไม่มีการส่งสัญญาณขอบริการ หรือสัญญาณให้บริการระหว่างกันเหมือนทีซีพี เรียกว่าการเชื่อมต่อแบบ Connectionless ซึ่งระบบทั้งสองตั้งสมมติฐานว่าการเชื่อมต่อระหว่างกันไม่มีความผิดพลาดเกิดขึ้นแน่

เนื่องจากมาตรฐานในเครือข่ายมีหลากหลาย ขนาดของแพ็กเก็ตในแต่ละมาตรฐานจึงมีความแตกต่างกันออกไป ทำให้การส่งข้อมูลระหว่างอุปกรณ์ในเครือข่ายนั้นอาจมีการแบ่งข้อมูลออกเป็นแพ็กเก็ตย่อยๆ ในระหว่างการส่ง เรียกว่า การทำแฟร็กเมนเตชัน (Fragmentation) เช่น แพ็กเก็ตของ FDDI มีขนาด 4,500 ไบต์ หากเครื่องปลายอยู่ในเครือข่าย Ethernet ซึ่งมีขนาดของแพ็กเก็ตสูงสุดเพียง 1,500 ไบต์ดังนั้นการส่งแพ็กเก็ตไปยังเครื่องปลายทางจึงต้องมีการแบ่งเป็นแพ็กเก็ตย่อย และเมื่อแพ็กเก็ตย่อยมาถึงเครื่องเป้าหมายก็จะมารวมกันเป็นแพ็กเก็ตเดิมที่มีขนาด 4,500 ไบต์อีกครั้ง เรียกการรวมกันนี้ว่า การรีแอสเซม-เบิ้ล (Reassemble) ซึ่งทำให้ได้ข้อมูลเหมือนที่ส่งมาจากเครื่องต้นทาง

ส่วนประกอบของแพ็กเก็ตไอพี

1. version : เป็นค่าตัวเลข 4 บิต บอกเวอร์ชันของมาตรฐานไอพีที่ใช้ โดยปกติมีค่าเป็น 4 ซึ่งหมายถึง IPv4
2. Internet Header Length (IHL) : เป็นตัวบอกความยาวเฮดเดอร์ของไอพี
3. Type of Service : เป็นส่วนที่บอกการทำงานของแพ็กเก็ตที่ส่งว่าทำหน้าที่อะไร มีทั้งหมด 8 บิต โดย

   Bit 0-2: บอกรายละเอียดการทำงานของแพ็กเก็ตนั้นๆ

   111 - Network Control

   110 - Internetwork Control

   101 - CRITIC / ECP

   100 - Flash Override

   011 – Flash

   010 – Immediate

   001 – Priority

   000 – Routine

   Bit 3: บอกถึงลักษณะของดีเลย์

   0 = Normal Delay – มีดีเลย์ปกติ

   1 = Low Delay – มีดีเลย์ต่ำ

   Bit 4: บอกถึงประเภทของทรูพุต

   0 = Normal Throughput – มีทรูพุตปกติ

   1= High Throughput – มีทรูพุตสูง

   Bit 5: บอกถึงประเภทของความน่าเชื่อถือ

   0 = Normal Reliability – มีความน่าเชื่อถือพอประมาณ

   1 = High Reliability – มีความน่าเชื่อถือสูง

   Bit 6-7: กันไว้ใช้ในอนาคต
4. Total Length : มีขนาด 16 บิต บอกถึงความยาวในดาต้าแกรมของไอพี
5. Identification field : เป็นตัวเลข 16 บิต เป็นค่าประจำตัวของไอพีนั้น โดยโฮสต์ที่ส่งเป็นผู้กำหนด และเพิ่มค่าขึ้นหนึ่งเมื่อมีการส่งดาต้าแกรมของไอพีใหม่ ซึ่งใช้ในการประกอบกลับ
6. Flag : เป็นตัวเลข 3 bit บอกลักษณะของแพ็กเก็ตว่ามีการแฟร็กเมนต์หรือไม่

   Bit 0: สงวนไว้ ปกติเป็น 0

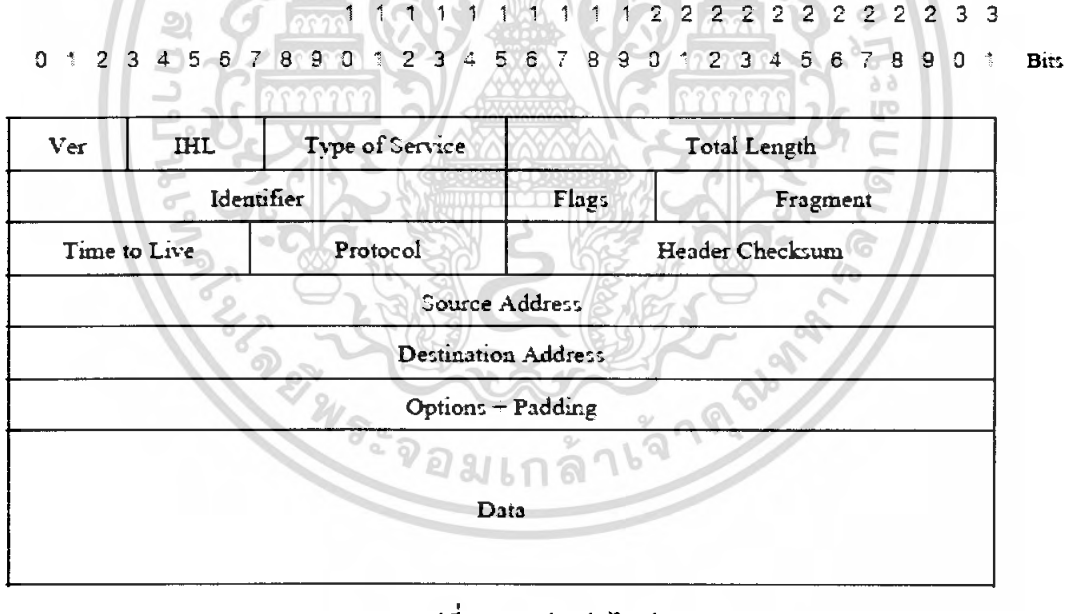
   Bit 1: 0 = บอกว่าแพ็กเก็ตมีการแตกแพ็กเก็ตย่อย

1 = บอกว่าแพ็กเก็ตไม่มีการแตกแพ็กเก็ตย่อย

Bit 2: 0 = บอกว่าแพ็กเก็ตนั้นเป็นแพ็กเก็ตสุดท้ายที่ได้จากการแตกแพ็กเก็ตย่อย

1 = บอกว่าแพ็กเก็ตนั้นยังไม่ใช่แพ็กเก็ตสุดท้ายที่ได้จากการแตกแพ็กเก็ตย่อย

7. Fragment Offset : เป็นค่าตัวเลข 13 บิต บอกออปเซตของแฟร็กเมนต์เมื่อเทียบในดาต้าแกรม
8. Time To Live (TTL) : เป็นตัวเลข 8 บิต บอกช่วงเวลาของแพ็กเก็ตที่ยังอยู่ในเครือข่ายได้ โดยกำหนดค่าเป็นจำนวนเราท์เตอร์สูงสุดที่ดาต้าแกรมผ่านได้ ซึ่งโดยทั่วไปทีค่าระหว่าง 32 ถึง 64 และลดค่าลงเรื่อยๆ เมื่อผ่านเราเตอร์ เพื่อเป็นการป้องกันแพ็กเก็ตล้นเครือข่าย
9. Protocol : เป็นตัวเลข 8 bit บอกถึงโพรโตคอลที่อยู่เหนือขึ้นไป ว่าเป็นโพรโตคอลระดับสูงกว่าประเภทใด
10. Header Checksum : เป็นค่าตัวเลข 32 บิต ใช้ตรวจสอบความถูกต้องของเฮดเดอร์
11. Source Address : เป็นค่าตัวเลข 32 บิต บอกถึงไอพีแอดเดรสของเครื่องต้นทาง
12. Destination Address : เป็นค่าตัวเลข 32 บิต บอกถึงไอพีแอดเดรสของเครื่องปลายทาง



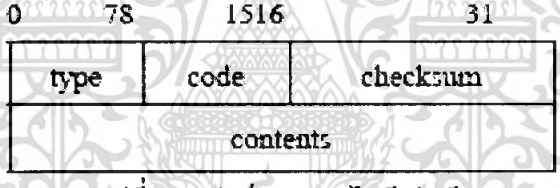
รูปที่ 2.7 แพ็กเก็ตไอพี

2.1.3.4 โพรโตคอล ไอซีเอ็มพี (ICMP: Internet Control Message Protocol)

หน้าที่หลักของ โพรโตคอล ไอซีเอ็มพี คือการแจ้งหรือแสดงข้อความจากระบบ เพื่อบอกให้ผู้ใช้ทราบว่าเกิดอะไรขึ้นในการส่งผ่านข้อมูลนั้น ซึ่งปัญหาส่วนมากที่พบ คือส่งไปไม่ได้ หรือปลายทางรับข้อมูลไม่ได้ เป็นต้น นอกจากนี้ โพรโตคอล ไอซีเอ็มพี ยังถูกเรียกใช้งานจากเครื่องเซิร์ฟเวอร์ และ เราท์เตอร์ อีกด้วย เพื่อแลกเปลี่ยนข้อมูลที่ใช้ควบคุม ส่วนรูปแบบการทำงานของ

โพรโตคอลไอซีเอ็มพีนั้นจะทำงานควบคู่กับโพรโตคอลไอพีในระดับเดียวกัน และข้อความต่างๆที่แจ้งให้ทราบจะถูกผนึกอยู่ภายในข้อมูลของไอพี(ไอพีดาทาแกรม) อีกทีหนึ่ง ข้อความที่โพรโตคอลไอซีเอ็มพีส่งนั้น แบ่งออกได้ 2 แบบคือ ICMP error message หรือข้อความแจ้งข้อผิดพลาด และ ICMP query หรือข้อความเรียกขอข้อมูลเพิ่มตัวอย่างกลไกการทำงานของโพรโตคอล ไอซีเอ็มพี เช่น เมื่อมีการส่งผ่านข้อมูลจากผู้ใช้ไปยังปลายทางที่ไม่ถูกต้อง หรือขณะนั้นเครื่องปลายทางเกิดปัญหาจนไม่สามารถรับข้อมูลได้ ที่เราท์เตอร์จะส่งข้อความแจ้งเป็นข้อความไอซีเอ็มพี(ICMP message) ที่ชื่อ destination unreachable ให้กับผู้ส่งข้อมูล นอกจากนี้ตัวข้อมูลที่แจ้งข้อความก็จะมีส่วนของข้อมูลไอพีดาทาแกรมที่เกิดปัญหาด้วย ดังนั้นเมื่อผู้ส่งข้อมูลได้รับข้อความแจ้งแล้วก็จะได้ทราบว่าจุดที่เกิดปัญหานั้นอยู่ที่ใด

ดังนั้นโพรโตคอล ไอซีเอ็มพี จึงกลายมาเป็นเครื่องมืออย่างหนึ่งในการช่วยทดสอบเครือข่าย เช่น คำสั่ง ping ที่เรามักใช้ทดสอบว่าเครื่องเซิร์ฟเวอร์ที่ให้บริการหรืออุปกรณ์ที่ต่ออยู่ในเครือข่ายอินเทอร์เน็ตนั้นยังทำงานเป็นปกติหรือไม่ แล้วคำสั่ง ping มีการเรียกใช้งานโพรโตคอลไอซีเอ็มพี แจ้งเป็นข้อความให้ทราบอีกต่อหนึ่ง



รูปที่ 2.8 ฟอร์แมตของไอซีเอ็มพี

1. Type ขนาด 8 บิต : กำหนดค่าความผิดพลาดและการรายงานสถานะ การใช้งานในปัจจุบันมีทั้งหมด 15 ประเภท
2. code ขนาด 8 บิต : รหัสความผิดพลาดย่อย
3. Checksum ขนาด 16 บิต : ค่าผลรวมตรวจสอบแบบ 1's complement สำหรับใช้ตรวจสอบความผิดพลาด โดยคำนวณผลรวมของ type, code และ contents
4. Contents ขนาดไม่คงที่ : ฟิลด์นี้ใช้บรรจุข้อมูลข่าวสารเพิ่มเติมเพื่อแจ้งกลับซึ่งจะขึ้นอยู่กับค่า type และ code

2.2 การโจมตีเพื่อให้ปิดบริการ

การโจมตีเพื่อให้ปิดบริการ (Denial of Service: DoS) หมายถึงการกระทำใดๆ ที่ทำให้ระบบเป้าหมายไม่สามารถให้บริการบางอย่างได้ หรือไม่สามารถให้บริการ ต่อไปได้อีก โดยทั่วไปโจมตีที่พอร์ตของทีซีพี/ไอพี ซึ่งเชื่อมต่อกับบริการ (Service) ที่รองรับพอร์ตนั้นๆ ดังนั้นการโจมตีพอร์ตจึงเท่ากับการโจมตีบริการของระบบนั่นเอง และอาจมีผลทำให้ระบบนั้นไม่สามารถให้บริการบางอย่างได้ หรือไม่สามารถให้บริการใดๆได้เลย

ในที่นี้ประเภทของการโจมตีเฉพาะบนชั้นทรานสปอร์ตและชั้นอินทเอร์เน็ต สามารถแบ่งออกเป็น 2 แบบหลักๆ ได้แก่

2.2.1 การส่งแพ็กเก็ตจำนวนมาก (Amount of Packets Sending)

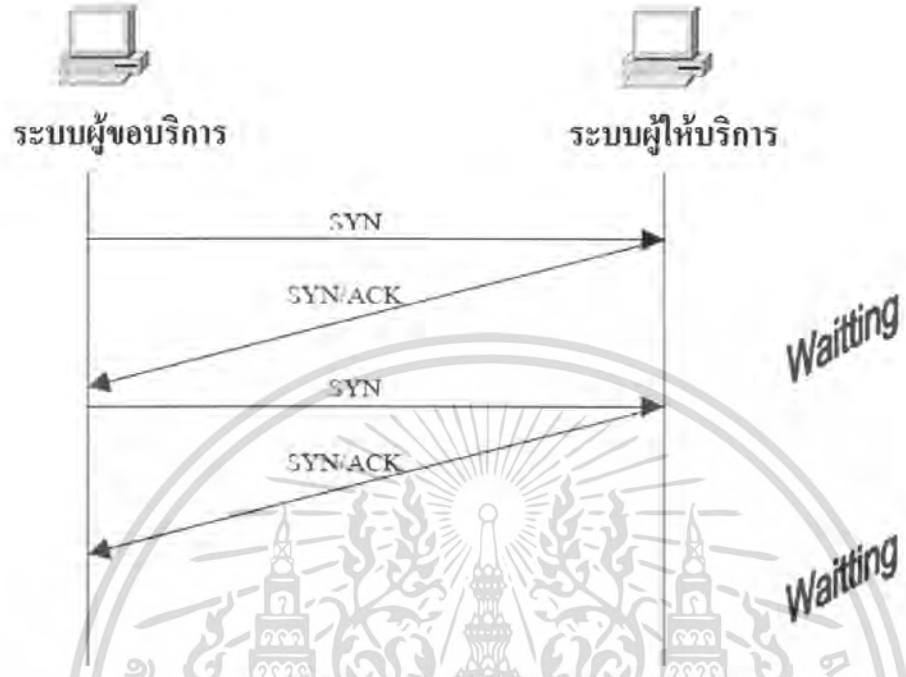
การโจมตีแบบนี้เป็นการส่งแพ็กเก็ตปริมาณมากเข้าไปยังระบบเป้าหมาย อาจทำให้ระบบเป้าหมายไม่สามารถให้บริการบางอย่างได้ หรือไม่สามารถทำงานต่อไปได้ ซึ่งแพ็กเก็ตที่ส่งออกไปนี้สามารถแบ่งออกได้เป็น

(1) แพ็กเก็ตข้อมูล (Data Packets)

การโจมตีแบบนี้เป็นการส่งแพ็กเก็ตข้อมูลปริมาณมาก เมื่อข้อมูลเข้ามาสู่เครื่องเป้าหมายก็เก็บไว้ในบัฟเฟอร์ก่อนที่จะนำมาประมวลผลอีกครั้ง ดังนั้นหากส่งแพ็กเก็ตเข้ามาเป็นปริมาณมากอาจทำให้บัฟเฟอร์ของเครื่องเป้าหมาย ไม่เพียงพอที่จะสามารถรองรับแพ็กเก็ตเหล่านั้นได้ทั้งหมด ซึ่งอาจทำให้เครื่องเป้าหมายให้บริการได้ช้าลง หรือต้องหยุดการให้บริการไปเลย

(2) แพ็กเก็ตสำหรับการควบคุม (Control Packets)

ตัวอย่างของการโจมตีแบบนี้ ได้แก่ การทำ SYN Flooding ปกติการเชื่อมต่อแบบ 3-way handshake เป็นไปตามลักษณะที่ได้อธิบายในหัวข้อ 2.3 แต่ในการโจมตีลักษณะนี้ใช้วิธีทำให้การทำ 3-way handshake ไม่สมบูรณ์ กล่าวคือ เครื่องที่ขอบริการส่งสัญญาณ SYN ไป แต่เมื่อได้รับสัญญาณ ACK จากเครื่องที่ให้บริการแล้วไม่ส่งสัญญาณ SYN ตอบกลับไป ทำให้เครื่องที่ให้บริการต้องเปิดการเชื่อมต่อรอการตอบกลับดังรูปที่ 2.12 ซึ่งการเปิดการเชื่อมต่อรอเอาไว้นี้ต้องใช้ทรัพยากรของระบบส่วนหนึ่ง และหากมีการส่งสัญญาณในลักษณะนี้มากๆ และทรัพยากรของระบบมีไม่เพียงพอ อาจทำให้ระบบไม่สามารถให้บริการอย่างอื่น หรือให้บริการกับผู้ร้องขอรายอื่นได้



รูปที่ 2.9 การส่งแพ็กเก็ตแบบ SYN Flood

2.3 การสแกนพอร์ต

เป็นการตรวจสอบว่าเครื่องที่เป้าหมายนั้นเปิดให้บริการอะไรบ้าง เนื่องจากบริการต่างๆ จะรอรับอยู่ที่พอร์ต โดยปกติจะมีค่าเป็นมาตรฐาน เช่น บริการรับส่งไฟล์ (เอฟทีพี) ให้บริการที่พอร์ต 21 หรือบริการเทอร์มินอล (เทลเน็ต) จะให้บริการที่พอร์ต 23 เป็นต้น ดังนั้นหากทราบหมายเลขพอร์ตที่เปิดรอการเชื่อมต่ออยู่ ก็จะทราบชื่อบริการที่มีอยู่ในเครื่องนั้นๆ ได้ โดยวิธีการสแกนพอร์ตสามารถแบ่งได้ 2 วิธีคือ

2.3.1 สแกนแบบพาสซีฟ (Passive Scan)

เป็นการสแกนโดยผู้สแกนไม่จำเป็นต้องติดต่อกับเครื่องเป้าหมายโดยตรง เช่นการแอบมองแพ็กเก็ตที่ส่งออกมาจากเครื่องเป้าหมาย หรืออาจทำได้โดยใช้เทคนิคการปลอมหมายเลขไอพี (IP Spoofing)ร่วมกับการหาความสัมพันธ์ของซีเควนซ์นัมเบอร์ เป็นต้น สำหรับการสแกนประเภทนี้จะไม่ขออธิบายในรายละเอียด เนื่องจากได้รับความนิยมน้อยมาก และไม่อยู่ในขอบเขตของโครงงานสำหรับให้ทำการตรวจสอบการสแกนพอร์ตจากวิธีการแบบพาสซีฟ

2.3.2 สแกนแบบแอคทีฟ (Active Scan)

เป็นการสแกนโดยผู้สแกนติดต่อกับเครื่องเป้าหมายโดยตรงผ่านโพรโตรคอลไอพี สามารถแบ่งย่อยได้อีกหลายเทคนิค ดังนี้

- **TCP SYN SCAN**

วิธีนี้ผู้สแกนจะทำการส่ง SYN แพ็กเก็ต (เซตค่าแฟล็ก SYN ไว้เป็น 1) เพื่อทำการติ ดต่อโดยตรงกับเป้าหมายโดยไม่ผ่านระบบปฏิ บัติ การ และรอผลการตอบรั บของเป้าหมายกลับมา ซึ่งหากเป้าหมายทำงานอยู่ก็จะตอบกลับมาด้วย SYN ACK (เป็นแพ็กเก็ตที่เซตค่าแฟล็ก SYN และ ACK ไว้เป็น1) หรือหากไม่มีแอพพลิเคชันทำงานอยู่จะตอบกลับมาด้วย RST การสแกนแบบนี้หากตรวจสอบบนโฮสต์เป้าหมายจะพบว่ามีการขอเชื่อมต่อเข้ามา แต่ไม่สามารถเปิดการติ ดต่อได้สำเร็จ เทคนิคนี้ บางครั้ งถูกเรียกว่า half-open scanning คือไม่สามารถทำ 3-way handshake ได้ จึงไม่มีการเชื่อมต่อใดๆเกิดขึ้นระหว่างเครื่องผู้สแกน กับเครื่องที่ถูกสแกน

- **FIN SCAN**

เป็นการส่ง FIN แพ็กเก็ตไปยังเป้าหมาย โดยที่เครื่องเป้าหมายก็จะยังตอบแพ็กเก็ตนั้นกลับไป แม้จะไม่มีการสื่อสารใดๆมาก่อนก็ตาม ซึ่งโดยปกติแล้วแพ็กเก็ตที่เซตค่าFIN เป็น 1 จะเป็นแพ็กเก็ตที่ใช้ในการตอบกลับ และการตอบกลับของเครื่องเป้าหมายสำหรับพอร์ตที่เปิดไว้ และพอร์ตที่ไม่ได้เปิดให้บริการก็ไม่เหมือนกัน หากเป็นพอร์ตที่เปิดอยู่ก็จะตอบด้วย FIN ACK กลับไป และหากเป็นพอร์ตที่ไม่ได้เปิดก็จะตอบด้วย RST ACK

- **SYN/FIN SCAN**

วิธีนี้จะใช้ TCP Flag ทั้ง SYN และ FIN พร้อมกัน ซึ่งปกติเป็นแฟล็กที่ไม่มีกำหนดไว้ในโปรโตคอล และจะไม่พบแฟล็กเช่ นนี้ในการสื่ อสารตามปกติ เป็ นอันขาดเพราะโดยปกติแล้ว SYN Flag จะใช้เมื่อเริ่มการติดต่อ ส่วน FIN จะใช้เมื่อต้องการยุติการติดต่อ การตอบรับของโฮสต์แต่ละประเภทในกรณีที่ ทำงานอยู่นั้นอาจจะแตกต่างกันไป เช่นเป็น SYN ACK หรือ FIN ACK อย่างใดอย่างหนึ่ง ส่วนการตอบรับในกรณีที่พอร์ตปิดจะตอบเหมือนกันคือ RST

- **NULL SCAN**

วิธีนี้จะไม่ใช้แฟล็กใดๆในการสแกนเลย โดยส่งแพ็กเก็ตที่ไม่มีแฟล็กใดที่ถูกเซตไว้เลยไปยังเป้าหมาย เป็นการเซ็ตแฟล็กทุกค่าให้เป็น 0 หมด ซึ่งแพ็กเก็ตลักษณะนี้จะไม่มีอยู่ในโปรโตคอล โดยทั่วไปการตอบสนองแพ็กเก็ตที่ไม่ได้อยู่ในโปรโตคอล จะมีการตอบรับที่ต่างกันออกไปตามแต่ประเภทของระบบปฏิบัติการ ดังนั้นนอกจากการใช้แพ็กเก็ตเหล่านี้เพื่อการสแกนพอร์ตแล้วยังสามารถนำแพ็กเก็ตเหล่านี้ไปใช้ในการตรวจสอบระบบปฏิ บัติการของเป้า

หมายได้อีกด้วยโดยการส่งแพ็กเก็ตที่มีแฟล็กซึ่งไม่อยู่ในข้อกำหนด การส่งแพ็กเก็ตลักษณะนี้ หากพอร์ตของเครื่องเป้าหมายปิดอยู่ การตอบรับจะเป็นการส่ง RST กลับไป

- **X'mas SCAN**

จะเป็นการส่งแพ็กเก็ต TCP ที่เซ็ตแฟล็ก FIN, Push, URGENTไปยังพอร์ตเป้าหมายที่เครื่องปลายทาง ซึ่งมักไม่เป็นที่ สนใจในการตรวจสอบเท่ากับ SYN-ACK-RSTเครื่องปลายทางจะส่งแพ็กเก็ต TCP RST ของพอร์ตที่ปิดอยู่กลับมาให้

- **UDP SCAN**

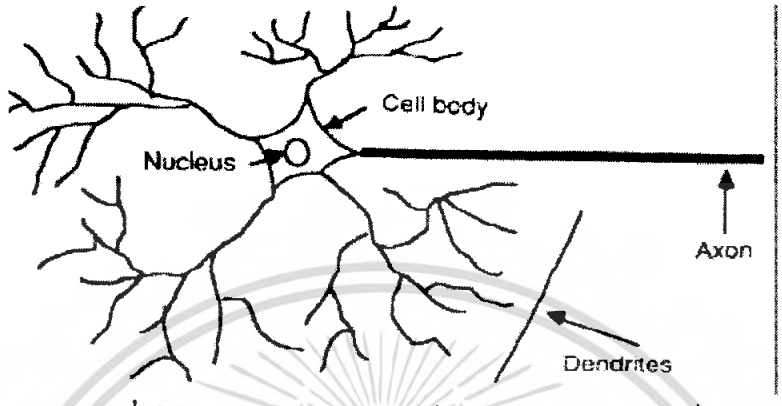
จะส่งแพ็กเก็ตของโปรโตคอล UDP ไปยังพอร์ตเป้าหมาย แต่เนื่องจาก UDPมีการจัดการที่แตกต่างจาก TCPโดยโปรโตคอล UDP เป็นโปรโตคอลแบบคอนเนกชันเลส(connectionless) ดังนั้นผลลัพธ์ของการสแกนเมื่อพอร์ตเปิดอยู่นั้นจะไม่สามารถคาดการณ์ได้ ขึ้นอยู่กับแต่ละแอพพลิเคชัน และไม่มีมาตรฐานที่เหมือนกันแต่อย่างใด ดังนั้นการสแกนUDP จึงต้องดูผลลัพธ์จาก ICMP เป็นหลัก หากพอร์ตไม่เปิดให้บริการ จะมี ICMP Messageว่า UDP Port Unreachable กลับมา และหากพอร์ตเปิดให้บริการ อาจมีการตอบรับหรือไม่และอย่างไร จะขึ้นอยู่กับการทำงานของแอพพลิเคชันที่เปิดพอร์ตนั้น แต่ที่แน่นอนคือจะไม่มี ICMP Message กลับมา

2.4 โครงข่ายใยประสาทเทียม

2.4.1 แนวคิดเกี่ยวกับโครงข่ายใยประสาทเทียม

โครงข่ายใยประสาทเทียม (Artificial neural network) คือ โมเดลทางคณิตศาสตร์ สำหรับประมวลผลสารสนเทศด้วยการคำนวณแบบ คอนเนคชันนิสต์ (Connectionist) เพื่อจำลองการทำงานของโครงข่ายประสาทในสมองมนุษย์ ด้วยวัตถุประสงค์ที่จะสร้างเครื่องมือซึ่งมีความสามารถในการเรียนรู้การจดจำแบบรูป (Pattern Recognition) และการอุปมานความรู้ (Knowledge Deduction) เช่นเดียวกับความสามารถที่มีในสมองมนุษย์ แนวคิดเริ่มต้นของเทคนิคนี้ได้มาจากการศึกษาข่ายงานไฟฟ้าชีวภาพ (Bioelectric Network) ในสมอง ซึ่งประกอบด้วย เซลล์ประสาท หรือ นิวรอน (Neurons) และ จุดประสานประสาท (Synapses) แต่ละเซลล์ประสาทประกอบด้วยปลายในการรับกระแสประสาท เรียกว่า เดนไดรท์ (Dendrite) ซึ่งเป็นค่าป้อนเข้า หรือ อินพุต (Input) และปลายในการส่งกระแสประสาทเรียกว่า แอคซอน (Axon) ซึ่งเป็นเหมือน ค่าผลที่ได้รับ หรือ เอาต์พุต (Output) ของเซลล์ เซลล์เหล่านี้ทำงานด้วยปฏิกิริยาไฟฟ้าเคมี เมื่อมีการกระตุ้นด้วยสิ่งเร้าภายนอกหรือกระตุ้นด้วยเซลล์ด้วยกัน กระแสประสาทจะวิ่งผ่านเดนไดรท์เข้าสู่นิวเคลียส ซึ่งจะเป็นตัวตัดสินว่าต้องกระตุ้นเซลล์อื่น ๆ ต่อหรือไม่ ถ้ากระแสประสาทแรงพอ นิวเคลียสก็จะกระตุ้นเซลล์อื่น ๆ ต่อไปผ่านทางแอคซอนของมัน

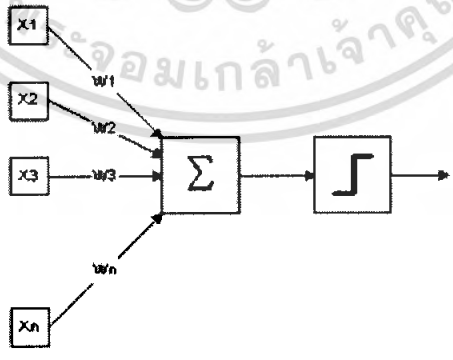
ตามแบบจำลองนี้ข่ายงานประสาทเกิดจากการเชื่อมต่อระหว่างเซลล์ประสาท จนเป็นโครงข่ายที่ทำงานร่วมกัน



รูปที่ 2.10 แบบจำลองของเซลล์ประสาทในสมองมนุษย์

2.4.2 โครงสร้างของโครงข่ายใยประสาทเทียม

นักวิจัยส่วนใหญ่ในปัจจุบันเห็นตรงกันว่าข่ายงานประสาทเทียมมีโครงสร้างแตกต่างจากข่ายงานในสมอง แต่ก็ยังเหมือนสมอง ในแง่ที่ว่าข่ายงานประสาทเทียม คือการรวมกลุ่มแบบขนานของหน่วยประมวลผลย่อยๆ และการเชื่อมต่อนี้เป็นส่วนสำคัญที่ทำให้เกิดสติปัญญาของข่ายงาน เมื่อพิจารณาขนาดแล้วสมองมีขนาดใหญ่กว่าข่ายงานประสาทเทียมอย่างมาก รวมทั้งเซลล์ประสาทยังมีความซับซ้อนกว่าหน่วยย่อยของข่ายงาน อย่างไรก็ดีหน้าที่สำคัญของสมอง เช่น การเรียนรู้ยังคงสามารถถูกจำลองขึ้นอย่างง่ายด้วยโครงข่ายประสาทนี้



รูปที่ 2.11 แบบจำลองของนิวรอนในคอมพิวเตอร์

2.4.3 หลักการของโครงข่ายใยประสาทเทียม

สำหรับในคอมพิวเตอร์นิวรอนประกอบด้วยอินพุตและเอาต์พุตเหมือนกัน โดยจำลองให้อินพุตแต่ละอันมีค่าน้ำหนัก (Weight) เป็นตัวกำหนดน้ำหนักของอินพุตโดยนิวรอนแต่ละหน่วยจะมีค่าขีดจำกัด (Threshold) เป็นตัวกำหนดว่าน้ำหนักรวมของอินพุตว่าต้องมากขนาดไหนจึงจะสามารถส่งเอาต์พุตไปยังนิวรอนตัวอื่นได้

2.4.4 การทำงานของโครงข่ายใยประสาทเทียม

เมื่อมีอินพุตเข้ามายังโคงข่ายแล้ว จะนำอินพุตมาคูณกับค่าน้ำหนักของแต่ละขา ผลที่ได้จากอินพุตทุกๆ ขาของนิวรอนจะนำมารวมกันแล้วเทียบกับค่าขีดจำกัดที่กำหนดไว้ หากผลรวมมีค่ามากกว่าค่าขีดจำกัดแล้วนิวรอนจะทำการส่งเอาต์พุตออกไป เอาต์พุตนี้จะถูกส่งต่อไปยังอินพุตของนิวรอนอื่นๆ ที่เชื่อมกันในโครงข่าย แต่หากว่าค่าที่ได้มีค่าน้อยกว่าค่าขีดจำกัดแล้วจะไม่ทำให้เกิดเอาต์พุต สามารถเขียนได้ดังนี้

if (sum(input * weight) > threshold) then output

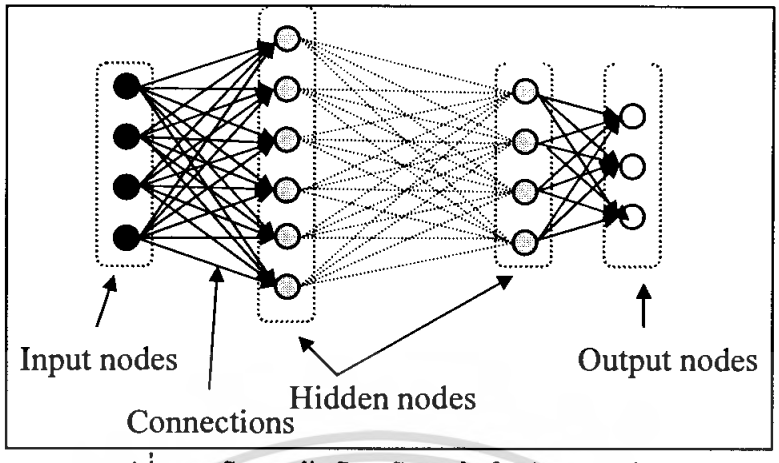
สิ่งสำคัญคือต้องทราบค่าน้ำหนักและค่าขีดจำกัดสำหรับสิ่งที่ต้องการเพื่อให้คอมพิวเตอร์เรียนรู้จดจำ ซึ่งเป็นค่าที่ไม่แน่นอน แต่สามารถกำหนดให้คอมพิวเตอร์ปรับค่าเหล่านั้นได้โดยการสอนรูปแบบ (Pattern) ของสิ่งที่เราต้องการให้รู้เรียนรู้ เรียกว่า กระบวนการถ่ายทอดแบบย้อนกลับหรือ แบคพรอเพเกชัน (Back Propagation) ซึ่งเป็นกระบวนการย้อนกลับของการรู้จำ ในการฝึกระบบโครงข่ายแบบป้อนไปข้างหน้า (Feed-forward Neural Networks) จะมีการใช้กระบวนการถ่ายทอดแบบย้อนกลับเพื่อใช้ในการปรับปรุงค่าน้ำหนักของโครงข่าย (network weight) หลังจากใส่รูปแบบข้อมูลสำหรับฝึกให้แก่โครงข่ายในแต่ละครั้งแล้ว ค่าที่ได้รับ (output) จากโครงข่ายจะถูกนำไปเปรียบเทียบกับผลที่คาดหวัง แล้วทำการคำนวณหาค่าความผิดพลาด ซึ่งค่าความผิดพลาดนี้จะถูกส่งกลับเข้าสู่โครงข่ายเพื่อใช้แก้ไขค่าน้ำหนักคะแนนต่อไป

อย่างเช่นจะเรียนรู้จดจำรูปสามเหลี่ยม กับรูปสี่เหลี่ยม เราอาจแบ่งค่าป้อนเข้าเป็น 9 ตัวคือเป็นตาราง 3x3 ถ้าวาดรูปสี่เหลี่ยมหรือสามเหลี่ยมให้เต็มกรอบ 3x3 พอดี สี่เหลี่ยมจะมีส่วนของขอบอยู่ในช่อง 1, 2, 3, 4, 6, 7, 8, 9 ดังนั้นจึงให้น้ำหนักตรงช่องเหล่านี้มีค่ามากๆ หากมีเส้นขีดผ่านก็เอามาคูณกับน้ำหนักแล้วก็เอามารวมกัน ตั้งค่าให้พอเหมาะก็จะสามารถแยกแยะระหว่างสี่เหลี่ยมกับสามเหลี่ยมได้ ซึ่งนี่คือหลักการของโครงข่ายใยประสาทเทียม

1	2	3
4	5	6
7	8	9

1	2	3
4	5	6
7	8	9

รูปที่ 2.12 การแยกแยะระหว่างสี่เหลี่ยมและสามเหลี่ยม



รูปที่ 2.13 โครงสร้างโครงโครงข่ายใยประสาทเทียม

ค่าผลที่ได้รับของแต่ละบัพ (Node)

$$y_i = f(w_i^1 x_1 + w_i^2 x_2 + w_i^3 x_3 + \cdots + w_i^m x_m) \quad (2.1)$$

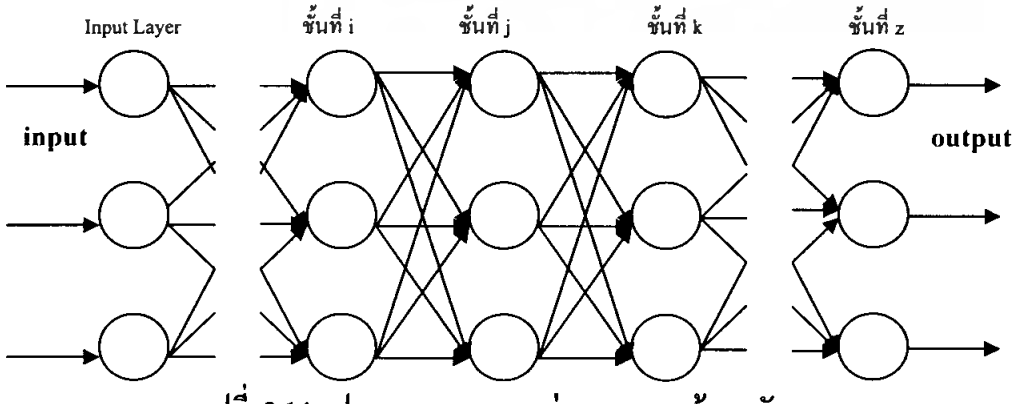
$$= f(\sum_j w_i^j x_j) \quad (2.2)$$

เมื่อ Xi = input จากบัพอื่นๆ

Wij = น้ำหนัก (weight) ของแต่ละแขน (connection)

2.4.5 กระบวนการถ่ายทอดแบบย้อนกลับ

กระบวนการถ่ายทอดแบบย้อนกลับ (Back-propagation) เป็นกระบวนการขั้นตอนวิธี (Algorithm) ที่ใช้ในการเรียนรู้ของโครงข่ายใยประสาทวิธีหนึ่งที่นิยมใช้ในโครงข่ายใยประสาทเพอร์เซปตรอนหลายชั้น (Multilayer Perceptron:MLP) เพื่อปรับค่าน้ำหนักในเส้นเชื่อมต่อระหว่างบัพให้เหมาะสม โดยการปรับค่านี้จะขึ้นกับความแตกต่างของค่าเอาต์พุตที่คำนวณได้กับค่าเอาต์พุตที่ต้องการ พิจารณารูปต่อไปนี้ประกอบ



รูปที่ 2.14 รูปแบบกระบวนการถ่ายทอดแบบย้อนกลับ

ขั้นตอนของกระบวนการถ่ายทอดแบบย้อนกลับ มีดังนี้

1.กำหนดค่าอัตราเร็วในการเรียนรู้ (Rate parameter: r)

2.สำหรับแต่ละตัวอย่างอินพุตให้ทำตามขั้นตอนต่อไปนี้จนกว่าได้ระดับผลการปฏิบัติ (Performance) ที่ต้องการ

- คำนวณหาค่าเอาต์พุตโดยใช้ค่าน้ำหนักเริ่มต้นซึ่งอาจได้จากการสุ่ม
- คำนวณหาค่า β : แทนประโยชน์ที่จะได้รับสำหรับการเปลี่ยนค่าเอาต์พุตของแต่ละบัพ
- ในชั้นเอาต์พุต (Output Layer)

$$\beta_z = d_z - o_z \qquad (2.3)$$

เมื่อ d_z แทนค่าเอาต์พุตที่ต้องการ

o_z แทนค่าเอาต์พุตที่คำนวณได้

- ในชั้นซ่อน (Hidden Layer)

$$\beta_j = \sum_k w_{j\ k}\, o_k (1 - o_k)\, \beta_k \qquad (2.4)$$

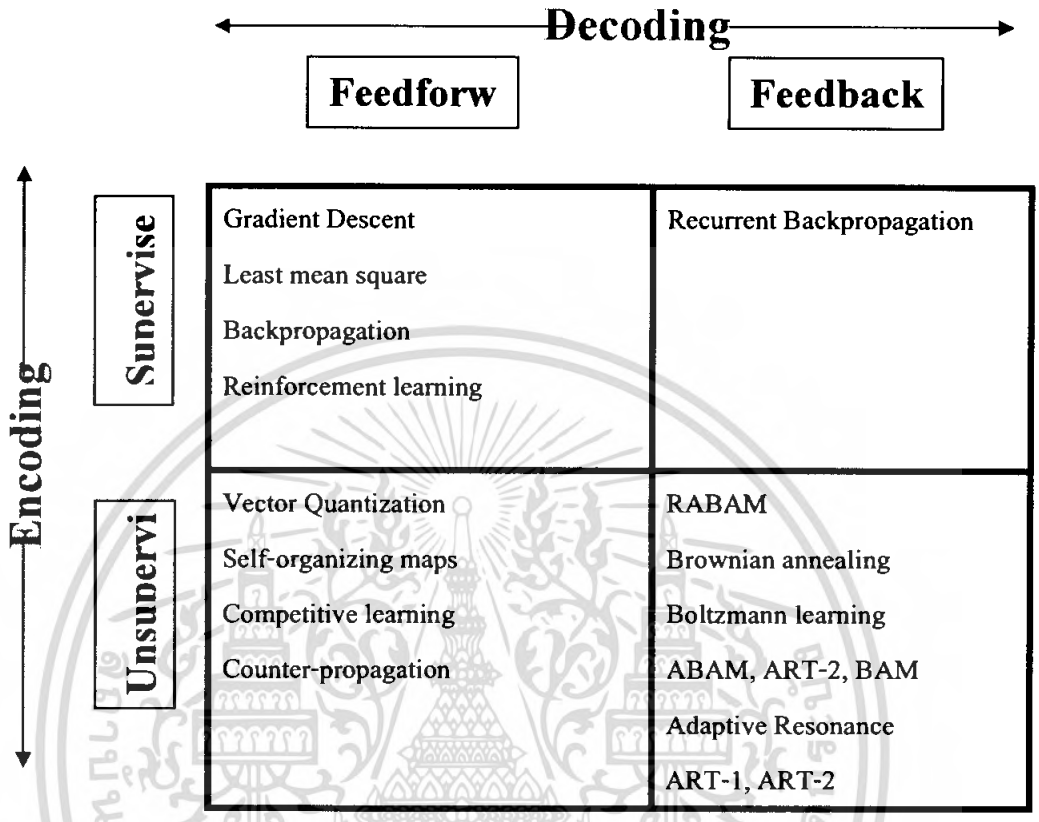
เมื่อ $w_{j\ k}$ แทนน้ำหนักของเส้นเชื่อมระหว่างชั้นที่ j กับ k

- คำนวณค่าน้ำหนักที่เปลี่ยนแปลงไปสำหรับในทุกน้ำหนัก ด้วยสมการต่อไปนี้

$$\Delta w_{i\ j} = r\, o_i\, o_j (1 - o_j)\, \beta_j \qquad (2.5)$$

- เพิ่มค่าน้ำหนักที่เปลี่ยนแปลง สำหรับตัวอย่างอินพุตทั้งหมด และเปลี่ยนค่าน้ำหนัก

2.4.6 อนุกรมวิธานโครงข่ายใยประสาทเทียม

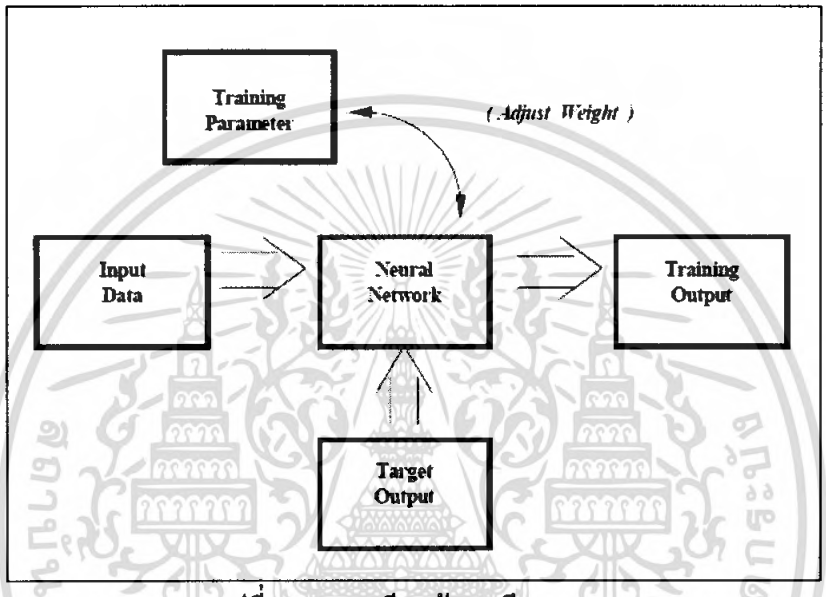


รูปที่ 2.15 อนุกรมวิธานโครงข่ายใยประสาทเทียม

2.4.7 การเรียนรู้สำหรับโครงข่ายใยประสาทเทียม

2.4.7.1 การเรียนรู้แบบมีการสอน (Supervised Learning)

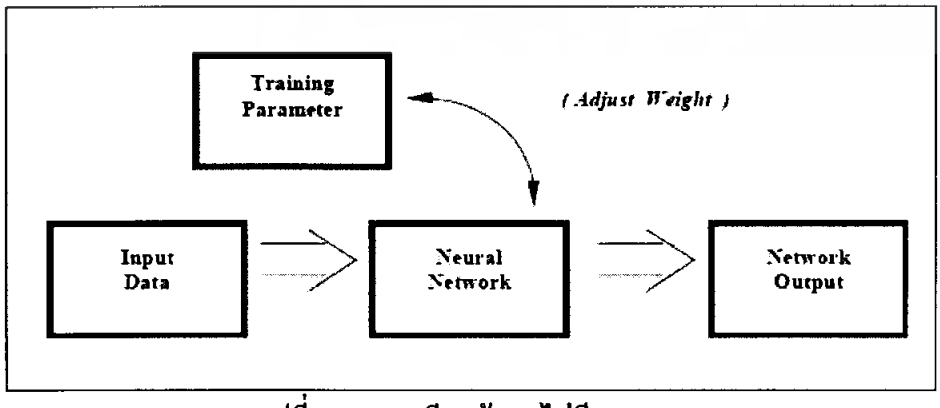
การเรียนแบบที่มีการตรวจคำตอบเพื่อให้โครงข่ายปรับตัว ชุดข้อมูลที่ใช้สอนโครงข่ายจะมีคำตอบไว้คอยตรวจดูว่าโครงข่ายให้คำตอบที่ถูกหรือไม่ ถ้าตอบไม่ถูก โครงข่ายก็จะปรับตัวเองเพื่อให้ได้คำตอบที่ดีขึ้น



รูปที่ 2.16 การเรียนรู้แบบมีการสอน

2.4.7.2 การเรียนรู้แบบไม่มีการสอน (Unsupervised Learning)

การเรียนแบบไม่มีผู้แนะนำ ไม่มีการตรวจคำตอบว่าถูกหรือผิด โครงข่ายจะจัดเรียงโครงสร้างด้วยตัวเองตามลักษณะของข้อมูล ผลลัพธ์ที่ได้ โครงข่ายจะสามารถจัดหมวดหมู่ของข้อมูลได้

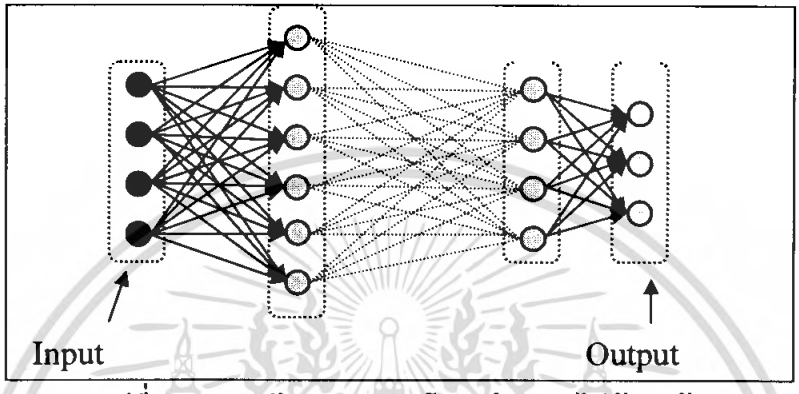


รูปที่ 2.17 การเรียนรู้แบบไม่มีการสอน

2.4.8 สถาปัตยกรรมโครงข่าย

2.4.8.1 โครงข่ายแบบป้อนไปข้างหน้า (Feedforward network)

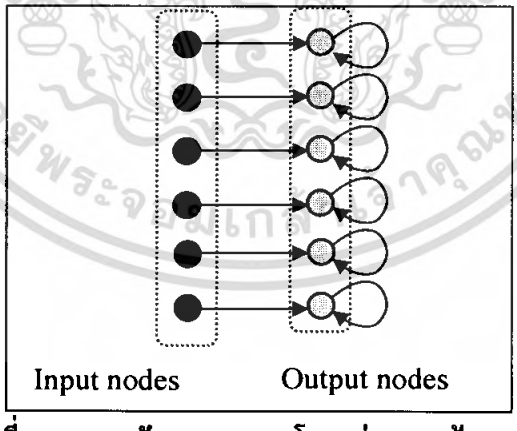
ข้อมูลที่ประมวลผลในโครงข่ายจะถูกส่งไปในทิศทางเดียวจากบัพที่รับค่า (Input nodes) ส่งต่อมาเรื่อยๆ จนถึงบัพผลลัพธ์ (output nodes) โดยไม่มีการย้อนกลับของข้อมูล หรือแม้แต่บัพในชั้นเดียวกันก็ไม่มีการเชื่อมต่อกัน



รูปที่ 2.18 สถาปัตยกรรมของโครงข่ายแบบไปข้างหน้า

2.4.8.2 โครงข่ายแบบป้อนกลับ (Feedback network, Recurrent network)

ข้อมูลที่ประมวลผลในโครงข่ายจะมีการป้อนกลับเข้าไปยังโครงข่ายหลายๆครั้ง กระทั่งได้คำตอบออกมา



รูปที่ 2.19 สถาปัตยกรรมของโครงข่ายแบบป้อนกลับ

2.4.8.3 ชั้นโครงข่าย (Network Layer)

พื้นฐานสามัญที่สำคัญของโครงข่ายใยประสาทเทียมประกอบไปด้วย 3 ส่วน หรือ 3 ชั้น (layer) ได้แก่ ชั้นของหน่วยรับค่า (Input Units) ที่ถูกเชื่อมต่อกับชั้นของหน่วยซ่อน (Hidden Units) ซึ่งเชื่อมต่อกับชั้นของหน่วยผลลัพธ์ (Output Units)

- การทำงานของหน่วยรับค่าจะทำหน้าที่ แทนส่วนของข้อมูลดิบ ที่จะถูกป้อนเข้าสู่โครงข่าย

- การทำงานของแต่ละหน่วยซ่อน จะถูกกำหนดโดยการทำงานของหน่วยรับค่า และค่าน้ำหนักบนความสัมพันธ์ระหว่างหน่วยรับค่าและหน่วยซ่อน

- พฤติกรรมการทำงานของหน่วยผลลัพธ์ จะขึ้นอยู่กับการทำงานของหน่วยซ่อนและค่าน้ำหนักระหว่างหน่วยซ่อนและหน่วยแสดงผล

ประเภทของโครงข่ายนี้สามารถกำหนดการแทนค่าให้แก่หน่วยรับค่าได้อย่างอิสระ ค่าน้ำหนักระหว่างหน่วยรับค่าและหน่วยซ่อนจะถูกกำหนดเมื่อหน่วยซ่อนกำลังทำงาน ดังนั้นเมื่อแก้ไขค่าน้ำหนักหน่วยซ่อนจะสามารถเลือกได้ว่าอะไรคือค่าที่ใช้แทนเข้ามา

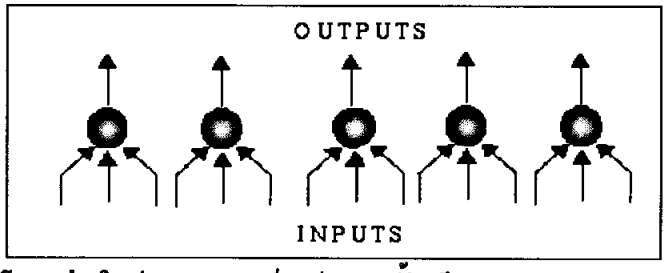
สถาปัตยกรรมของชั้น (Architecture of Layer) สามารถจำแนกออกเป็น 2 ประเภทคือ โครงข่ายใยประสาทแบบชั้นเดียว (Single-layer) และ แบบหลายชั้น (Multi-layer)

- โครงข่ายใยประสาทเพอร์เซปตรอนชั้นเดียว (Single-layer Perceptron) โครงข่ายใยประสาทที่ประกอบด้วยชั้นเพียงชั้นเดียว จำนวนบัพรับค่า (Input Nodes) ขึ้นอยู่กับจำนวนส่วนประกอบ (Components) ของค่าป้อนเข้า (Input Data) และฟังก์ชันกระตุ้น (Activation Function) ขึ้นอยู่กับลักษณะข้อมูลของผลลัพธ์เช่น ถ้าผลลัพธ์ที่ต้องการเป็น ใช่ หรือ ไม่ใช่ จะต้องใช้ฟังก์ชัยขีดจำกัด (Threshold Function)

$$f(x) = \begin{cases} 1 & \text{if } x \geq T \\ 0 & \text{if } x < T \end{cases} \qquad \text{T=Threshold level} \qquad (2.6)$$

หรือถ้าผลลัพธ์เป็นค่าตัวเลขที่ต่อเนื่อง เราต้องใช้ฟังก์ชันต่อเนื่อง (Continuous Function) เช่น ฟังก์ชันซิกมอยด์ (Sigmoid Function)

$$f(x) = \frac{1}{1+e^{x}} \qquad (2.7)$$



รูปที่ 2.20 โครงข่ายใยประสาทเพอร์เซปตรอนชั้นเดียว (Single-layer Perceptron)

- โครงข่ายใยประสาทเพอร์เซปตรอนหลายชั้น (Multi-layer Perceptron) โครงข่ายใยประสาทจะประกอบด้วยหลายชั้นโดยในแต่ละชั้นจะประกอบด้วยบัพ หรือเปรียบได้กับตัวเซลล์ประสาท (Neurons) ค่าน้ำหนักของเส้นที่เชื่อมต่อระหว่างบัพของแต่ละชั้น (เมทริก W), ค่าเวกเตอร์ถ่วง (Bias Vector: b) และค่าเวกเตอร์ลัพธ์ (Output Vector: a) โดย m เป็นตัวเลขบอกลำดับชั้นกำกับไว้ด้านบน เมื่อ p เป็นเวกเตอร์รับค่าเข้า (Input Vector) การคำนวณค่าผลลัพธ์สำหรับโครงข่ายใยประสาทที่มี M ชั้นจะเป็นดังสมการ

$$a^{m+1} = f^{m+1}(W^{m+1}a^{m} + b^{m+1}) \qquad (2.8)$$

เมื่อ $m = 0, 2, \ldots, M\text{-}I$

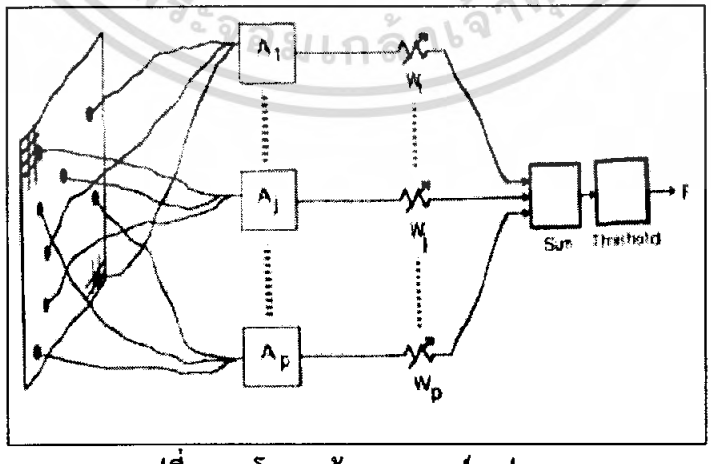
$a^{0} = p$

$a = a^{m}$

และ f เป็นฟังก์ชันโอน (Transfer Function)

2.4.8.4 เพอร์เซปตรอน (Perceptrons)

ในยุคทศวรรษที่ 1960 งานส่วนใหญ่ของข่ายงานได้รับการวิพากษ์วิจารย์ในหัวข้อเรื่องเพอร์เซปตรอนซึ่งค้นพบโดยแฟรงค์ โรเซนแบลทท์ (Frank Rosenblatt) โดยเพอร์เซปตรอนซึ่งกลายเป็นโมเดล MCP (neuron with weighted inputs) พร้อมกับส่วนต่อเติมจากรูปในส่วน A1, A2, Aj, Ap เรียกว่าหน่วยร่วม (Association Units) การทำงานเพื่อคัดเลือกสิ่งที่แตกต่างออกมาจากรูปภาพที่รับเข้าไป โดยเพอร์เซปตรอนสามารถคัดลอกความคิดพื้นฐานภายในของสัตว์เลี้ยงลูกด้วยนม หลักๆ แล้วจะใช้ในรูปแบบการจดจำแยกแยะ (recognition) และสามารถขยายให้มีความสามารถสูงกว่านี้



รูปที่ 2.21 โครงสร้างของเพอร์เซปตรอน

ในปี ค.ศ.1969 มินสกี (Minskey) และ พาเพิต (Papert) ได้ทำการตีพิมพ์หนังสือที่สามารถแสดงข้อมูลในเชิงคำนวนว่าเพอร์เซปตรอนชั้นเดียวไม่สามารถที่จะสร้างรูปแบบการจดจำพื้นฐาน (Basic Pattern Recognition Operation) ได้ เช่น การกำหนดความคล้ายคลึงของรูปร่าง หรือกำหนดว่ารูปร่างใดสัมพันธ์กันหรือไม่ แต่นักวิจัยไม่ทราบว่าการได้รับการฝึกฝนที่ถูกต้องซึ่งเพอร์เซปตรอนหลายชั้นสามารถแก้ไขสิ่งเหล่านี้ได้ ก่อนทศวรรษที่ 1980

2.4.9 การประยุกต์ใช้งานโครงข่ายใยประสาทเทียม

เนื่องจากความสามารถในการจำลองพฤติกรรมทางกายภาพของระบบที่มีความซับซ้อนจากข้อมูลที่ป้อนให้เรียนรู้ การประยุกต์ใช้โครงข่ายใยประสาทเทียมจึงเป็นทางเลือกใหม่ในการควบคุม ซึ่งมีผู้นำมาประยุกต์ใช้งานหลายประเภท ได้แก่

1. งานการจดจำรูปแบบที่มีความไม่แน่นอน เช่น ลายมือ ลายเซนต์ ตัวอักษร รูปหน้า
2. งานการประมาณค่าฟังก์ชันหรือการประมาณความสัมพันธ์ (มีอินพุตและเอาท์พุตแต่ไม่ทราบว่าอินพุตกับเอาท์พุตมีความสัมพันธ์กันอย่างไร)
3. งานที่สิ่งแวดล้อมเปลี่ยนแปลงอยู่เสมอ (โครงข่ายใยประสาทเทียมสามารถปรับตัวเองได้)
4. งานจัดหมวดหมู่และแยกแยะสิ่งของ
5. งานทำนาย เช่นพยากรณ์อากาศ พยากรณ์หุ้น
6. การประยุกต์ใช้ข่ายงานระบบประสาทควบคุมกระบวนการทางเคมีโดยวิธีพยากรณ์แบบจำลอง (Model Predictive Control)
7. การประยุกต์ใช้ข่ายงานระบบประสาทแบบแพร่กระจายกลับในการทำนายพลังงานความร้อนที่สะสมอยู่ในตัวอาคาร
8. การใช้โครงข่ายใยประสาทเทียมในการหาไซโครเมตริกชาร์ท (Psychrometric/Psycho metric Chart) การประยุกต์ใช้โครงข่ายใยประสาทเทียมควบคุมระบบปรับอากาศ (HVAC)

2.4.10 สรุปโครงข่ายประสาทเทียม (Artificial Neural Network)

โครงข่ายประสาทเทียม (Artificial Neural Network) คือ การสร้างคอมพิวเตอร์ที่จำลองเอาวิธีการทำงานของสมองมนุษย์ หรือทำให้คอมพิวเตอร์รู้จักคิดและจดจำในแนวเดียวกับโครงข่ายประสาทของมนุษย์ เพื่อช่วยให้คอมพิวเตอร์ฟังภาษามนุษย์ได้เข้าใจ อ่านออก และเรียนรู้จดจำได้ ซึ่งอาจเรียกได้ว่าเป็น สมองกล โครงสร้างของโครงข่ายประสาทเทียม ประกอบด้วย หน่วยรับค่า (Input Units) ,หน่วยผลลัพธ์ (Output Units) โดยมีการกำหนดค่าน้ำหนักให้แก่เส้นทางการนำเข้า

ของ ค่าป้อนเข้า (Input) แต่ละตัว ในการเรียนรู้ของโครงข่ายใยประสาท จะอาศัยกระบวนการถ่ายทอดแบบย้อนกลับ (Back-propagation Algorithm) ในการเขียน การสร้างการเรียนรู้สำหรับโครงข่ายประสาทเทียมเพื่อให้มีความคิดเสมือนมนุษย์ มีสองวิธี คือการเรียนรู้แบบมีการสอน (Supervised Learning) เปรียบเทียบกับคน เหมือนกับการสอนนักเรียนโดยมีครูผู้สอนคอยแนะนำ และ การเรียนรู้แบบไม่มีการสอน (Unsupervised Learning) เปรียบเทียบกับคน เช่น การที่เราสามารถแยกแยะพันธุ์พืช พันธุ์สัตว์ตามลักษณะรูปร่างของมันได้เองโดยไม่มีใครสอน

สถาปัตยกรรมโครงข่าย แบ่งเป็น 4 แบบ คือ โครงข่ายแบบป้อนไปข้างหน้า (Feedforward network), โครงข่ายแบบป้อนกลับ (Feedback network), ชั้นโครงข่าย (Network Layer) และ เพอร์เซปตรอน (Perceptrons)

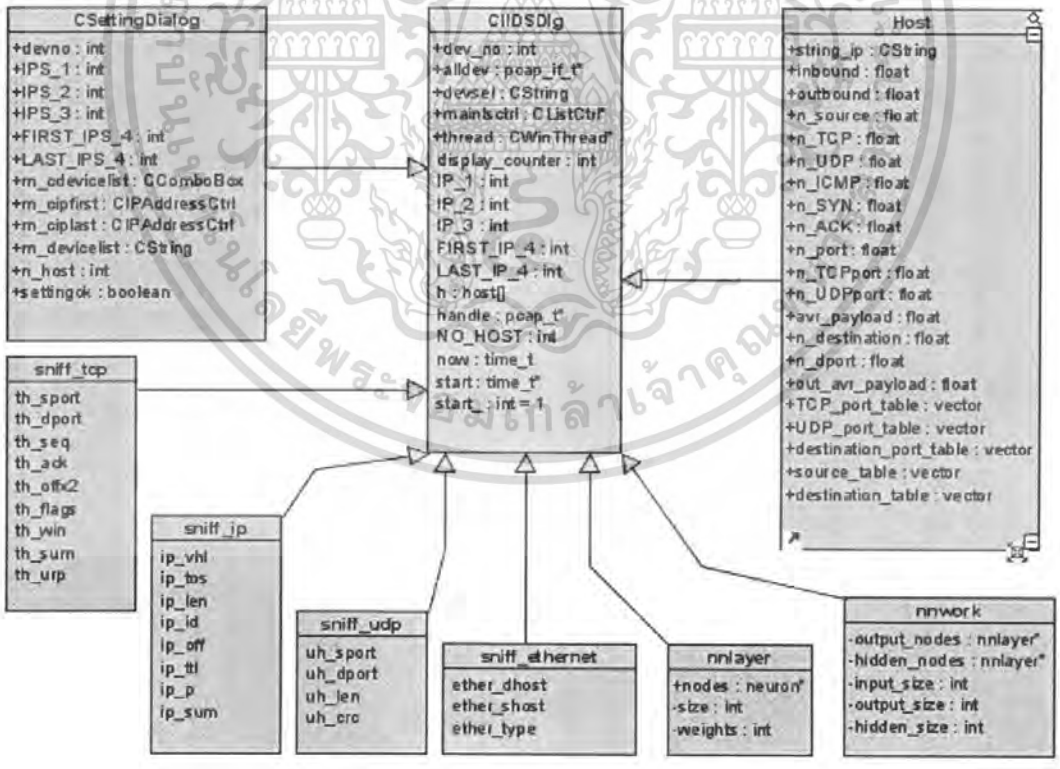
เนื่องจากความสามารถในการจำลองพฤติกรรมทางกายภาพของระบบที่มีความซับซ้อนจากข้อมูลที่ป้อนให้เพื่อการเรียนรู้ การประยุกต์ใช้โครงข่ายประสาทเทียมมีผู้นำมาประยุกต์ใช้งานหลายประเภท เช่น งานการจดจำรูปแบบที่มีความไม่แน่นอน งานการประมาณค่าฟังก์ชันหรือการประมาณความสัมพันธ์ งานที่สิ่งแวดล้อมเปลี่ยนแปลงอยู่เสมอเนื่องจากโครงข่ายประสาทเทียมสามารถปรับตัวเองได้ และนอกจากนี้ยังสามารถนำไปประยุกต์ในงานต่าง ๆ อีกหลายงาน ตัวอย่างของงานที่นำโครงข่ายประสาทเทียมไปประยุกต์ใช้งานเช่น ในการวิเคราะห์ และออกแบบระบบที่ช่วยในการแนะนำผู้ปฏิบัติงานในการควบคุมระบบปรับอากาศของอาคาร เพื่อให้ประหยัดพลังงานมากที่สุดในขณะที่ยังรักษาสมรรถะของระบบไว้สูงสุด ซึ่งเป็นการวิจัยที่เลือกอาคารสำนักงานใหญ่ธนาคารไทยพาณิชย์ (SCB Phase1) ซึ่งตั้งอยู่ที่ ใกล้สี่แยกรัชโยธิน บางเขน กรุงเทพฯ เป็นต้น

บทที่ 3
ระบบตรวจจับผู้บุกรุกชาญฉลาด

3.1 เกี่ยวกับตัวโปรแกรม

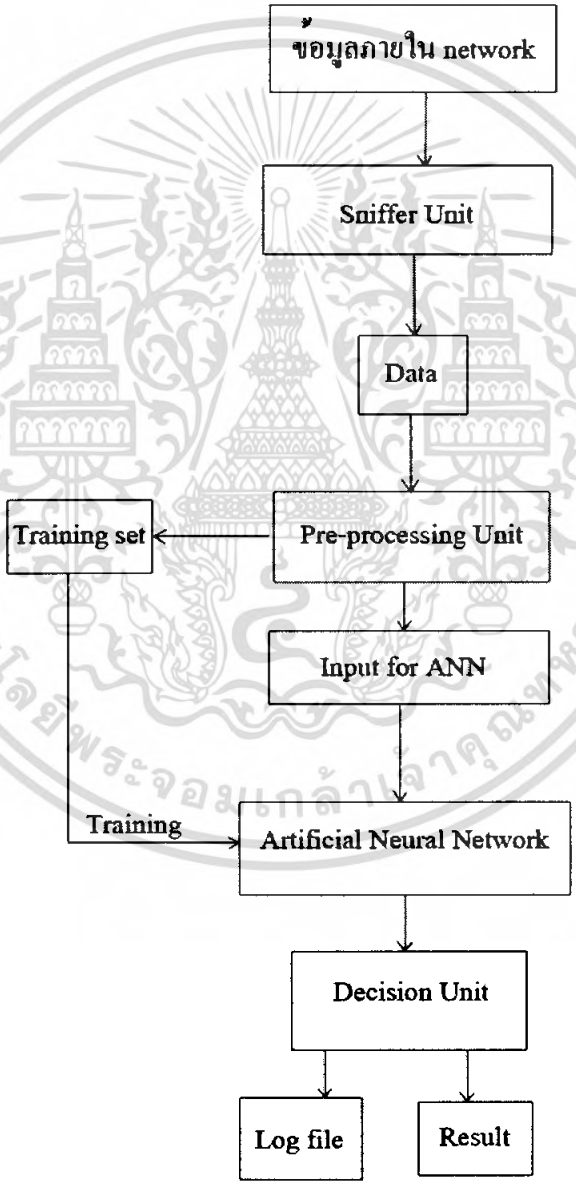
ระบบตรวจจับผู้บุกรุกชาญฉลาด เป็นระบบตรวจจับผู้บุกรุกทางเครือข่าย ทำงานบนระบบปฏิบัติการวินโดวส์ ทำการตรวจจับการโจมตีแบบปิดบริการและการสแกนพอร์ต ตรวจจับแยกตามไอพีแอสเดรส ทำให้สามารถบอกได้ว่าเครื่องไหนกำลังถูกโจมตีอยู่ ตัวโปรแกรมใช้การวิเคราะห์การโจมตีโดยอาศัยโครงข่ายใยประสาทเทียมที่มีการเรียนรู้จนสามารถแยกการโจมตีออกจากข้อมูลปกติได้แล้ว การวิเคราะห์ข้อมูลทำได้โดยดึงข้อมูลขึ้นมาจากเครือข่ายและทำการเตรียมข้อมูลให้เรียบร้อยก่อนจะส่งให้โครงข่ายใยประสาทเทียมประมวลผลเพื่อเอาค่าเอาท์พุตที่ได้มาตัดสินว่ามีการโจมตีเกิดขึ้นหรือไม่ ซึ่งถือเป็นการตรวจจับแบบอะนอมอลลี นั่นเอง

3.2 คลาสไดอะแกรมตัวโปรแกรม



3.3 สถาปัตยกรรมของโปรแกรม

โปรแกรมถูกออกแบบให้ เริ่มทำงานจากการดักจับข้อมูลในเครือข่าย แล้วจึงเข้าสู่ส่วนเตรียมการประมวล โดยจะทำหน้าที่เตรียมข้อมูลเพื่อนำไปใช้เป็นอินพุตของโครงข่ายใยประสาทเทียมเพื่อใช้ในการประมวลผลต่อไป หลังจากนั้นเมื่อโครงข่ายใยประสาทเทียมวิเคราะห์ผลแล้ว หากผลเข้าข่ายการโจมตี จะนำผลการวิเคราะห์เข้าสู่ส่วนการตัดสินใจเพื่อบอกถึงลักษณะการโจมตี ทั้งนี้จะทำการบันทึกการโจมตีไว้ด้วย ซึ่งสามารถเขียนได้ดังรูปที่ 3.1



รูปที่ 3.1 สถาปัตยกรรมของโปรแกรม

3.4 ชุดข้อมูลที่ดึงค่ามาจากเครือข่าย

การหาชุดข้อมูล (Dataset) ต้องทำการหาค่าต่างๆในสภาวะแวดล้อม (Environment) ที่เมื่อนำมาวิเคราะห์แล้วสามารถบอกได้ว่าอาจเกิดการโจมตีขึ้นภายในเครือข่าย จากตรงนี้จึงได้ทำการเลือกค่าจากสภาวะแวดล้อมออกมา 10 ค่าที่สามารถช่วยเป็นค่าบ่งชี้การโจมตีที่เกิดขึ้นในเครือข่ายได้ ได้แก่

- ปริมาณ TCP SYN
- ปริมาณ TCP ACK
- ปริมาณ TCP Packet
- ปริมาณ UDP Packet
- ปริมาณ ICMP Packet
- ปริมาณข้อมูล (Packet) ที่เข้าสู่เครื่อง
- ปริมาณข้อมูล (Packet) ที่ออกจากเครื่อง
- ปริมาณไอพีต้นทาง (Source IP) นอกขอบเขตที่สนใจ
- ปริมาณพอร์ต (Port) ปลายทาง
- ขนาดเฉลี่ยของขนาดข้อมูล (Payload)

ซึ่งทั้ง 10 ค่าที่เลือกมาล้วนแล้วแต่เป็นตัวที่มีความสำคัญที่สามารถบ่งชี้ว่าอาจจะเกิดการโจมตีขึ้นภายในเครือข่ายทั้งสิ้น ซึ่งค่าทั้งหมดจะถูกเก็บแยกไว้ตามแต่ละแอดเดรสเพื่อที่สามารถจะบอกได้ว่าการโจมตีนั้นเกิดขึ้นที่แอดเดรสใด โดยการเลือกแต่ละค่ามามีเหตุผลดังนี้

- ปริมาณ TCP SYN และ TCP ACK เพื่อดูการติดต่อระหว่างเครื่อง (Session) หรือพยายามทำการติดต่อเกิดขึ้นมากน้อยเพียงใดและมากเกินปกติหรือไม่
- ปริมาณข้อมูล (Packet) ที่เข้าสู่เครื่องและออกากเครื่อง เพื่อดูว่า ปริมาณข้อมูลปกติที่เครื่องมีการใช้งานปกติเป็นอย่างไร และหากเกิดการโจมตี จะเป็นอย่างไร จะเปลี่ยนแปลงอย่างไร
- ปริมาณ TCP UDP ICMP packet จะทำให้ทราบว่าข้อมูลแต่ละชนิดมีมากน้อยเพียงใดและมากเกินปกติหรือไม่
- ปริมาณไอพีต้นทาง (Source IP) นอกเครือข่ายเพื่อดูว่าการใช้งานปกติ จะมีการติดต่อกับเครื่องภายนอกมากน้อยเพียงใด หากเกิดการโจมตีขึ้น จะเปลี่ยนแปลงอย่างไร
- ปริมาณพอร์ต (Port) ปลายทาง เพื่อดูว่าปกติมีการเปิดพอร์ตในการติดต่อมาน้อยเพียงใด และหากเกิดการโจมตีขึ้น จะเปลี่ยนแปลงอย่างไร
- ขนาดเฉลี่ยของขนาดข้อมูล (Payload) เพื่อดูว่าข้อมูลที่เข้ามาโดยเฉลี่ยแล้ว มีขนาดข้อมูลประมาณเท่าไร และหากเกิดการโจมตีขึ้นจะมีขนาดเฉลี่ยเปลี่ยนไปอย่างไร

3.5 การเตรียมชุดข้อมูลที่เป็นอินพุตให้โครงข่ายใยประสาทเทียม

หลังจากได้ชุดข้อมูลขึ้นมาจากเครือข่ายแล้วยังถือว่าเป็นข้อมูลดิบอยู่ บางค่ายังไม่สามารถนำไปใช้เลยได้ เนื่องจากข้อมูลเหล่านี้บางค่าไม่สามารถบอกถึงการโจมตีบางชนิดได้ การใส่เข้าไป

ในโครงข่ายใยประสาทโดยไม่จำเป็นจะทำให้เกิดความผิดพลาดสูงได้ นอกจากนั้นข้อมูลดิบเหล่านี้บางตัวถ้านำมาใช้งานเลยจะมีความผิดพลาดค่อนข้างสูงเช่นกันในกรณีที่เปลี่ยนสภาพแวดล้อมไป

ดังนั้นการจะนำข้อมูลเหล่านี้มาใช้ในการตรวจจับการโจมตีด้วยโครงข่ายใยประสาทเทียมจำเป็นต้องเลือกเฉพาะค่าที่สามารถบ่งบอกถึงการโจมตีนั้นๆได้ นอกจากนั้นบางค่าต้องมีการนำไปเปลี่ยนเป็นค่าที่เหมาะสมถึงจะนำมาใช้ได้

ตารางที่ 3.1 แสดงอินพุตที่ใช้ป้อนให้กับโครงข่ายใยประสาทเทียม

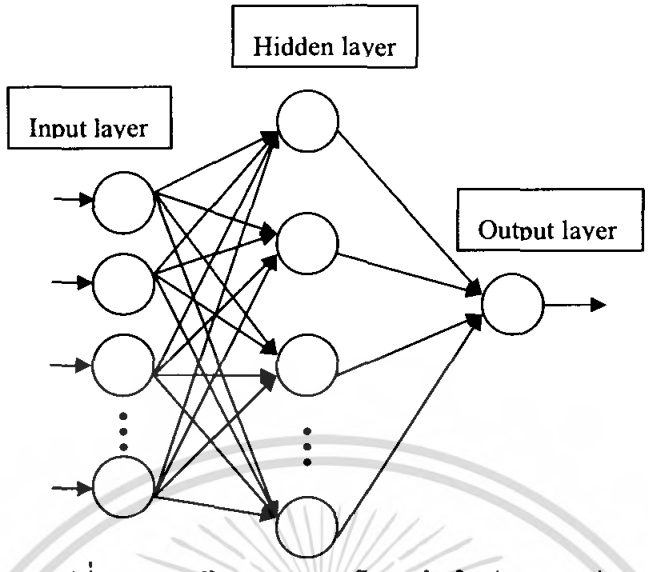
อินพุตของโครงข่ายส่วนการโจมตีเพื่อให้ปิดบริการ	อินพุตของโครงข่ายส่วนพอร์ตสแกน
ปริมาณ Source IP นอกขอบเขตที่สนใจ	ปริมาณพอร์ตปลายทาง
ขนาดเฉลี่ยของ Payload	ขนาดเฉลี่ยของ Payload
อัตราส่วนระหว่าง packet เข้าและออก	อัตราส่วนระหว่าง TCP SYN ต่อ ACK
อัตราส่วนระหว่าง TCP SYN ต่อ ACK	ปริมาณ TCP Packet
อัตราส่วนระหว่างปริมาณ UDP ต่อ TCP	ปริมาณ UDP Packet
อัตราส่วนระหว่างปริมาณ ICMP ต่อ TCP	

3.6 สถาปัตยกรรมของโครงข่ายใยประสาทเทียม

การออกแบบสถาปัตยกรรมของโครงข่ายใยประสาทเทียมเริ่มต้นจากการเลือกรูปแบบการเรียนรู้และลักษณะของโครงข่าย ซึ่งได้เลือกการเรียนรู้แบบมีการสอน ดังนั้นลักษณะของโครงข่ายที่เหมาะสมก็คือ เพอร์เซปตรอนแบบหลายชั้น ตรงนี้เลือกใช้ ชั้นซ่อน เพียง 1 ชั้น และใช้กระบวนการถ่ายทอดย้อนกลับในการปรับค่าภายในโครงข่ายด้วย

โครงข่ายใยประสาทเทียมของการโจมตีเพื่อให้ปิดบริการมีชั้น Input/Hidden/Output = 6/18/1

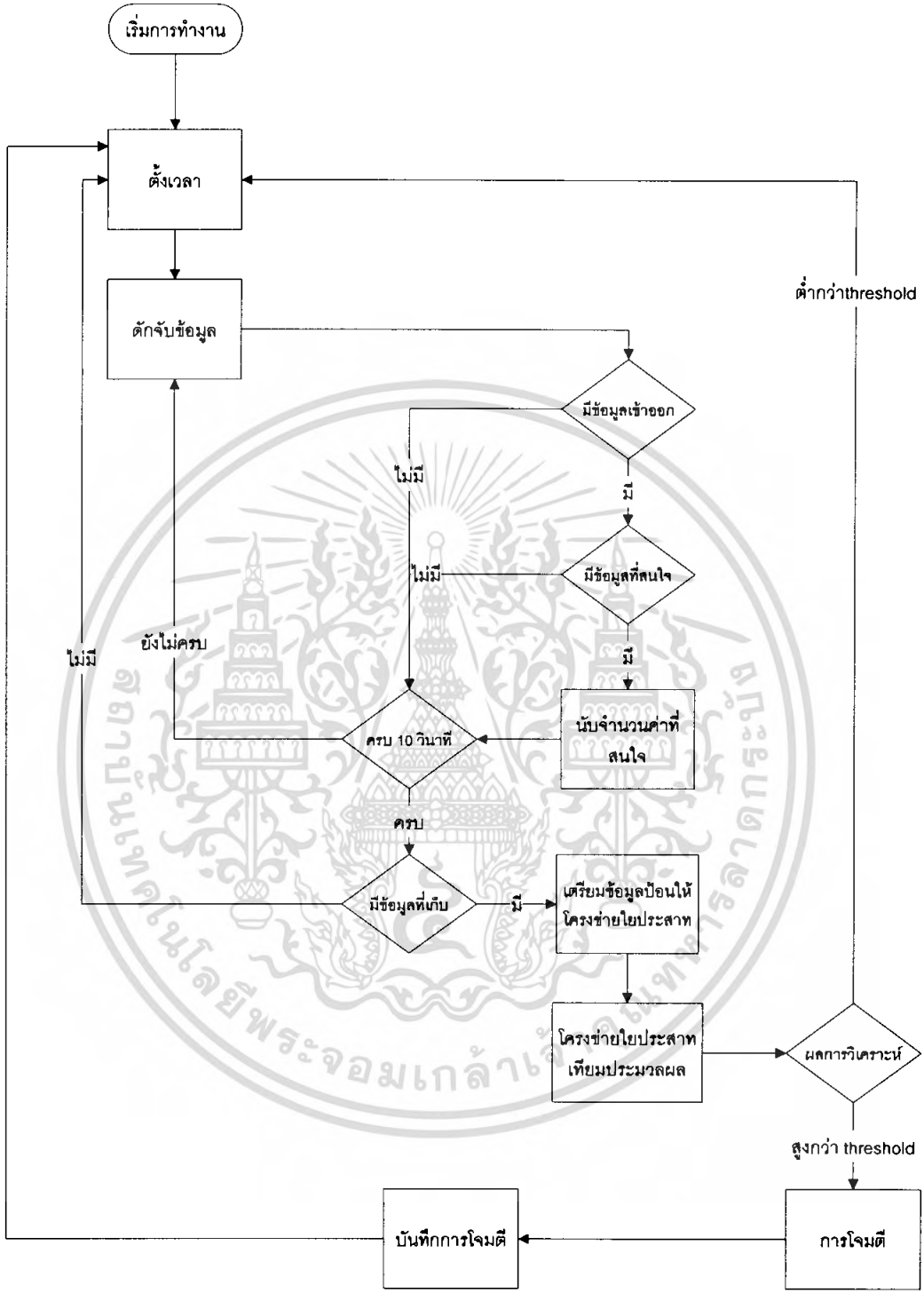
โครงข่ายใยประสาทเทียมของพอร์ตสแกน มีชั้น Input/Hidden/Output = 5/18/1



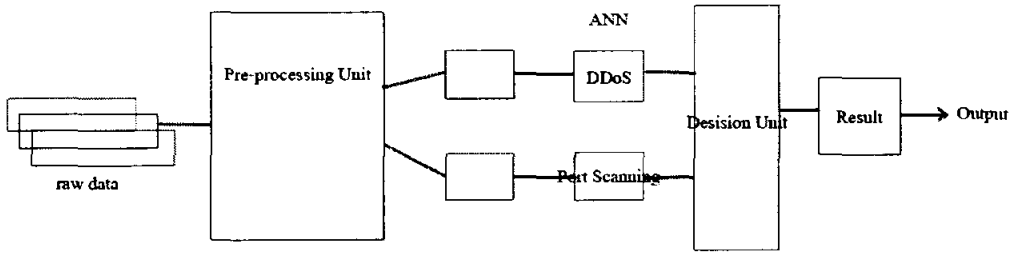
รูปที่ 3.2 สถาปัตยกรรมของโครงข่ายใยประสาทเทียม

3.7 ลักษณะการทำงานของโปรแกรม

โปรแกรมทำการตั้งเวลา 10 วินาทีไว้เพื่อดักจับข้อมูล ระหว่างช่วงเวลานี้โปรแกรมจะทำการดักจับข้อมูลภายในเครือข่ายและคัดเอาเฉพาะค่าที่สนใจเพื่อเก็บข้อมูลไว้ หลังจากนั้นโปรแกรมจะตรวจสอบว่ามีค่าพร้อมจะประมวลผล ก็จะนำเอาค่านั้นเข้าสู่กระบวนการเตรียมข้อมูลก่อนประมวลผล หลังจากเตรียมข้อมูลเสร็จเรียบร้อยแล้วก็จะป้อนค่าข้อมูลเหล่านั้นเข้าสู่โครงข่ายใยประสาทเทียมที่ได้ถูกสอนให้วิเคราะห์การโจมตีไว้เรียบร้อยแล้ว โดยโครงข่ายใยประสาทเทียมถูกสร้างขึ้นแยกไว้ตามการโจมตีแบบต่างๆเพื่อวิเคราะห์การโจมตีแยกๆกันไป เมื่อได้ค่าเอาท์พุตจากโครงข่ายใยประสาทเทียมส่วนตัดสินใจจะดูค่าเอาท์พุตนั้นว่ามีค่าเกินจากที่กำหนดหรือไม่ ถ้าเกินจากที่กำหนด แสดงว่ามีการโจมตีเกิดขึ้น ก็จะทำการแสดงผลพร้อมบันทึกข้อมูลการโจมตีนั้นไว้แล้วย้อมกลับไปทำกระบวนการแรกใหม่อีกครั้ง โดยกระบวนการทำงานทุกขั้นตอนสามารถดูได้จากผังกระบวนการทำงานในรูปที่ 3.3



รูปที่ 3.3 รูปแสดงกระบวนการทำงานของโปรแกรม



รูปที่ 3.4 รูปแสดงแผนผังการทำงานของโปรแกรม

3.8 การเตรียมการใช้งานโปรแกรม

เนื่องจากโปรแกรมนี้ถูกพัฒนาขึ้นด้วยภาษา C++ และใช้ไลบรารีของ WinPcap ในการดึงข้อมูลจากเครือข่ายผ่านทางอุปกรณ์เชื่อมต่อเครือข่าย ดังนั้นจำเป็นต้องทำการติดตั้ง WinPcap เข้ากับเครื่องที่จะใช้งานโปรแกรมก่อนเสมอ ยกเว้นเครื่องที่ได้เคยมีการลง WinPcap แล้ว สามารถใช้งานโปรแกรมได้ทันที

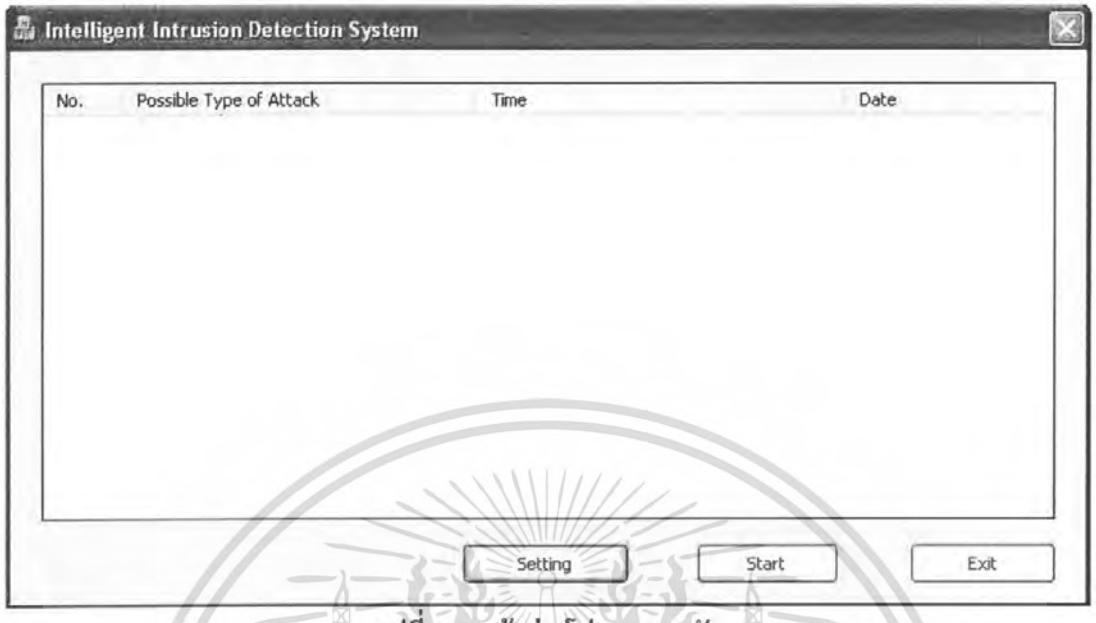
3.9 ส่วนประกอบของโปรแกรม

ชุดโปรแกรมประกอบไปด้วยไฟล์ต่างๆดังนี้

- IIDS.exe ใช้เริ่มทำงานโปรแกรม
- NN_DDOS.nnw เก็บค่าโครงข่ายใยประสาทส่วนวิเคราะห์การโจมตีเพื่อปิดบริการ
- NN_SCAN.nnw เก็บค่าโครงข่ายใยประสาทส่วนวิเคราะห์การสแกนพอร์ต

3.10 การใช้งานโปรแกรม

1. ทำการรันโปรแกรมขึ้นมาจะพบกับหน้าต่างไดอะลอกแรกดังรูปที่ 3.4



รูปที่ 3.5 หน้าต่างโปรแกรมหลัก

2. ก่อนเริ่มทำงานจริงจำเป็นต้องตั้งค่าการทำงานให้กับโปรแกรมก่อน โดยการกดที่ปุ่ม Setting จะแสดงหน้าต่างถัดไปดังรูปที่ 3.5

รูปที่ 3.6 หน้าต่างส่วนการตั้งค่าของโปรแกรม

3. หน้าต่างนี้ที่ช่อง Device ให้ทำการเลือกอุปกรณ์เครือข่ายที่เราต้องการ

4. ที่ช่อง IP Range ให้ใส่ช่วงของไอพีแอดเดรสที่เราสนใจจะตรวจจับการโจมตีเข้าไปเช่น 192.168.5.110 – 192.168.5.115 เป็นต้น ถ้าต้องการตรวจจับแค่ไอพี่แอดเดรสเดียวให้ใส่เหมือนกันทั้ง 2 ช่อง

5. กด OK เพื่อยืนยันการตั้งค่า จะกลับมายังหน้าต่างแรกอีกครั้ง

6. เริ่มทำงานโปรแกรมโดยกดปุ่ม Start โปรแกรมก็จำเริ่มการตรวจจับทันที

บทที่ 4

การทำงานและการทดลอง

4.1 ขั้นตอนการทำงาน

ขั้นตอนการทำงานประกอบไปด้วย

4.1.1 การเก็บข้อมูลปกติทางเครือข่าย

4.1.2 การเก็บข้อมูลการโจมตีทางเครือข่าย

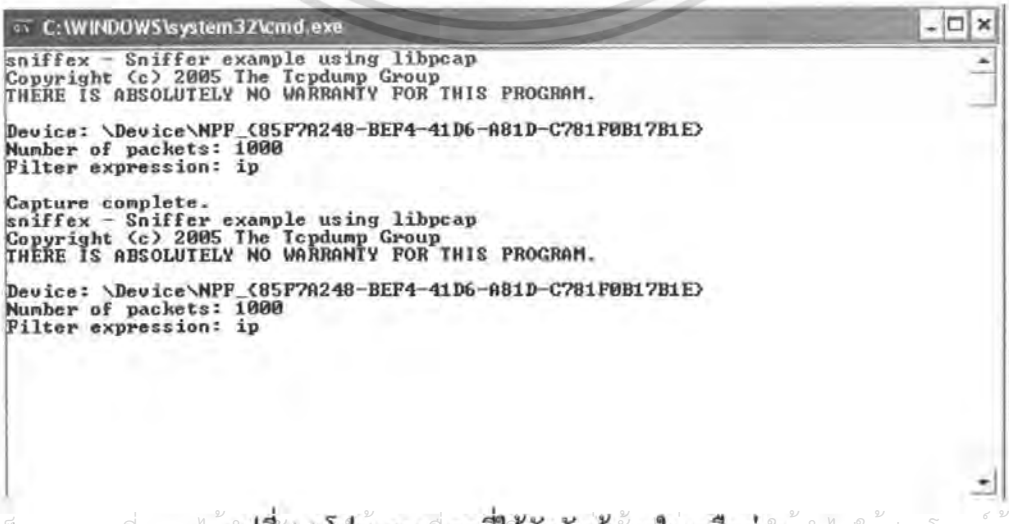
4.1.3 การเรียนรู้ของโครงข่ายใยประสาทเทียม

4.1.4 การตั้งค่าการทดสอบการโจมตีและทำส่วนติดต่อกับผู้ใช้

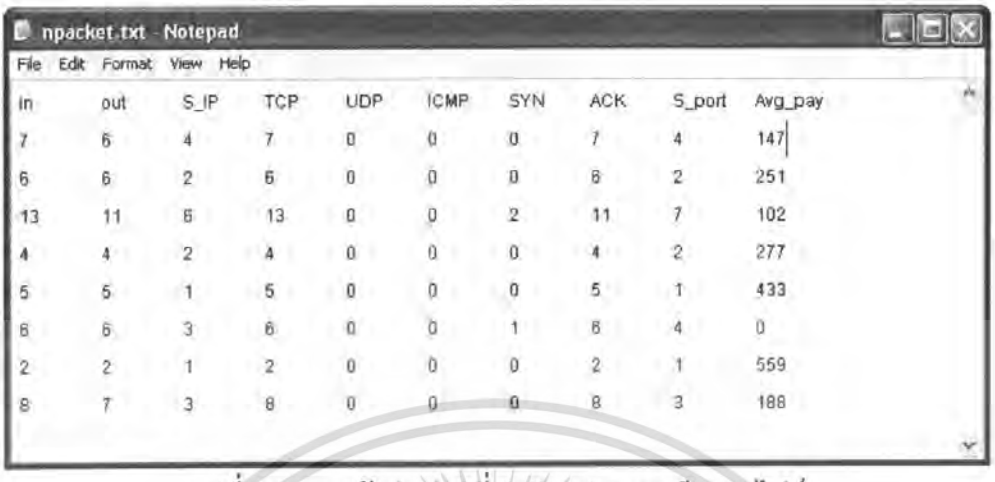
4.1.1 การเก็บข้อมูลปกติทางเครือข่าย

การเก็บข้อมูลปกติทางเครือข่าย คือข้อมูลที่ไม่มีการโจมตีเกิดขึ้นเลยภายในเครือข่าย เพื่อนำไปใช้ในการเรียนรู้ของโครงข่ายใยประสาทเทียม ทำได้โดย เก็บข้อมูลจากเครื่องตัวอย่าง ที่ระบุให้เป็นการใช้งานปกติ โดยในการทดลองนี้ เครื่องตัวอย่าง ได้ทำการติดตั้ง ระบบปฏิบัติการวินโดวส์ XP ทำการทดสอบโดยการใช้งานอินเตอร์เน็ท ผ่านเว็บเบราเซอร์ มีการใช้งานโปรแกรมรับส่งข้อความ(Instant Messenger) เช่น MSN Messenger มีการใช้งานการเล่นเกมออนไลน์ เช่น Ragnarok และ Warcraft 3 ผ่าน Battlenet

ขั้นตอนนี้ทำการเขียนโปรแกรมเพื่อดักจับข้อมูลในเครือข่ายที่ต้องการซึ่งได้แก่ข้อมูลทั้ง 10 ค่าที่ได้กล่าวมาแล้วในบทที่ 3 แล้วเก็บลงไฟล์เพื่อทดสอบการดักจับข้อมูลว่าสามารถทำได้จริง ได้โปรแกรมดังรูปที่ 4.1 จากนั้นก็นำโปรแกรมดังกล่าวไปทำงานที่เครื่องตัวอย่างที่กล่าวมาข้างต้น เพื่อทำการเก็บข้อมูลซึ่งถือว่าเป็นข้อมูลปกติทางเครือข่ายลงไฟล์เพื่อเตรียมนำไปใช้ในการสอนและเรียนรู้ของโครงข่ายใยประสาทเทียมเพื่อให้รู้ลักษณะของต่อไป



รูปที่ 4.1 โปรแกรมแรกที่ใช้ดักจับข้อมูลในเครือข่าย



npacket.txt - Notepad

File Edit Format View Help

in	out	S_IP	TCP	UDP	ICMP	SYN	ACK	S_port	Avg_pay
7	6	4	7	0	0	0	7	4	147
6	6	2	6	0	0	0	6	2	251
13	11	6	13	0	0	2	11	7	102
4	4	2	4	0	0	0	4	2	277
5	5	1	5	0	0	0	5	1	433
6	6	3	6	0	0	1	6	4	0
2	2	1	2	0	0	0	2	1	559
8	7	3	8	0	0	0	8	3	188

รูปที่ 4.2 แสดงข้อมูลปกติที่ดักจับมาและถูกเขียนลงไฟล์

4.1.2 การเก็บข้อมูลการโจมตีทางเครือข่าย

การเก็บข้อมูลการโจมตีทางเครือข่าย คือข้อมูลเมื่อเกิดความผิดปกติขึ้นภายในเครือข่าย เพื่อนำไปใช้ในการเรียนรู้ของโครงข่ายใยประสาทเทียม ทำได้โดย ใช้โปรแกรมที่ใช้ในการสร้างการโจมตีในเครือข่ายซึ่งหาได้ตามอินเทอร์เน็ตทั่วไปเพื่อทำการโจมตีไปยังเครื่องตัวอย่าง

ทำการเปิดโปรแกรมโจมตีเพื่อให้ปิดบริการ แล้วนำโปรแกรมดักจับข้อมูลส่วนแรกเช่นเดียวกันไปทำงานไว้ที่เครื่องตัวอย่างเพื่อดักจับข้อมูลที่เป็นการโจมตีเพื่อปิดบริการ แล้วเก็บไว้ในไฟล์ต่างหากจากไฟล์ข้อมูลปกติในเครือข่ายเพื่อนำไปใช้ในการเรียนรู้ของโครงข่ายใยประสาทเทียมเพื่อให้รู้ลักษณะการโจมตีเพื่อให้ปิดบริการ

ทำการเปิดโปรแกรมสแกนพอร์ต แล้วนำโปรแกรมดักจับข้อมูลส่วนแรกเช่นเดียวกันไปทำงานไว้ที่เครื่องตัวอย่างเพื่อดักจับข้อมูลที่เป็นการสแกนพอร์ต แล้วเก็บไว้ในไฟล์ต่างหากจากไฟล์ข้อมูลปกติในเครือข่ายและไฟล์การโจมตีเพื่อปิดบริการ เพื่อนำไปใช้ในการเรียนรู้ของโครงข่ายใยประสาทเทียมเพื่อให้รู้ลักษณะการสแกนพอร์ต

จากกระบวนการข้างต้นเราจะได้ข้อมูลทั้งหมด 3 ส่วนแยกอยู่คนละไฟล์ได้แก่

(1) ไฟล์ข้อมูลปกติในเครือข่าย

(2) ไฟล์ข้อมูลการโจมตีเพื่อให้ปิดบริการ

(3) ไฟล์ข้อมูลการสแกนพอร์ต

apacket.txt - Notepad

File Edit Format View Help

in	out	S_IP	TCP	UDP	ICMP	SYN	ACK	S_port	Avg_pay
169	169	5	169	0	0	158	11	63	10
178	173	5	178	0	0	160	18	61	38
172	171	5	172	0	0	165	7	65	5
169	168	5	169	0	0	156	14	60	50
177	177	5	177	0	0	166	12	65	9
173	173	4	173	0	0	166	8	65	6
181	179	4	181	0	0	168	13	66	45
189	186	5	189	0	0	165	24	66	53

รูปที่ 4.3 แสดงตัวอย่างการดักจับข้อมูลที่เป็นการสแกนพอร์ต

4.1.3 การเรียนรู้ของโครงข่ายใยประสาทเทียม

ทำการเขียนโปรแกรมส่วนโครงข่ายใยประสาทเทียมโดยแยกออกเป็น 2 โครงข่ายเพื่อใช้ตรวจจับการโจมตีที่แตกต่างกันทั้ง 2 แบบ

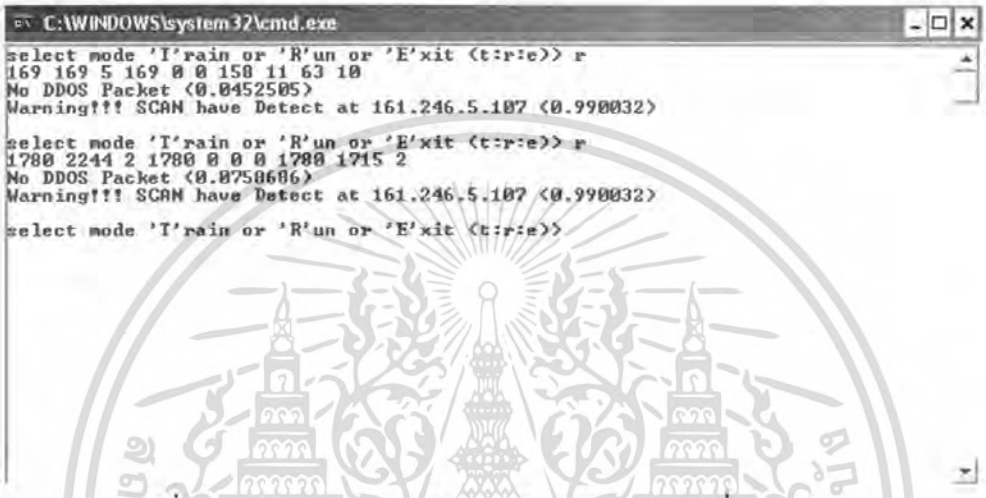
หลังจากนั้นก็นำข้อมูลทั้ง 3 ส่วนที่แยกไว้นำมาใช้ทำการเรียนรู้ให้กับโครงข่ายใยประสาทเทียม ว่าไฟล์ไหนคือเครือข่ายปกติไฟล์ไหนคือการโจมตีเพื่อให้ปิดบริการและไฟล์ไหนคือการสแกนพอร์ตโดยให้เรียนรู้ว่าข้อมูลปกติจะมีเอาท์พุตออกมาให้ใกล้ค่า "0" ส่วนข้อมูลการโจมตีต่างเอาท์พุตออกมาให้ใกล้ค่า "1" โครงข่ายใยประสาทเทียมจะทำการปรับค่าน้ำหนักภายในเพื่อให้ใกล้ค่าเอาท์พุตที่ต้องการมากที่สุด ดังรูปที่ 4.4

```
C:\WINDOWS\system32\cmd.exe
complete training Neural Network with Normal 289
complete training Neural Network with Normal 290
complete training Neural Network with Normal 291
complete training Neural Network with Normal 292
complete training Neural Network with Normal 293
complete training Neural Network with Normal 294
complete training Neural Network with Normal 295
complete training Neural Network with Normal 296
complete training Neural Network with Normal 297
complete training Neural Network with Normal 298
complete training Neural Network with Normal 299
complete training Neural Network with Normal 300
complete training Neural Network with Normal 301
complete training Neural Network with SCAN 1
complete training Neural Network with SCAN 2
complete training Neural Network with SCAN 3
complete training Neural Network with SCAN 4
complete training Neural Network with SCAN 5
complete training Neural Network with SCAN 6
complete training Neural Network with SCAN 7
complete training Neural Network with SCAN 8
complete training Neural Network with SCAN 9
complete training Neural Network with SCAN 10
complete training Neural Network with SCAN 11
complete training Neural Network with SCAN 12
```

รูปที่ 4.4 แสดงกระบวนการสอนให้โครงข่ายใยประสาทเรียนรู้

4.1.4 การตั้งค่าการทดสอบการโจมตีและทำส่วนติดต่อกับผู้ใช้

ทดสอบขั้นแรกของเรียนรู้ของโครงข่ายใยประสาทด้วยการป้อนค่าตัวเลขจากไฟล์ที่ดักจับเพื่อดูเอาท์พุตที่ได้หลังจากนั้นจึงทำการตั้งค่าเริ่มต้นความสำคัญ (Threshold) ให้เหมาะสมที่สุดเพื่อให้สามารถบอกได้ว่าค่าใดคือการโจมตี ตามรูปที่ 4.5



รูปที่ 4.5 แสดงการทดสอบด้วยการป้อนค่าการสแกนเพื่อดูเอาท์พุต

เมื่อปรับจนได้ค่าที่ดีที่สุดแล้ว ในขั้นต่อไปทำการรวมโปรแกรมส่วนที่ดักจับข้อมูลทางเครือข่ายกับส่วนโครงข่ายใยประสาทเทียมเข้าด้วยกันเพื่อทำงานตามเวลาจริงคือดักจับข้อมูล ทำการประมวลผล และแสดงผลออกมาทันที ตามรูปที่ 4.6

รูปที่ 4.6 แสดงการทดสอบการดักจับการโจมตีจริง

หลังจากได้โปรแกรมสมบูรณ์เรียบร้อยแล้ว ต่อไปจึงทำการเขียนส่วนหน้าจอติดต่อกับผู้ใช้ เพื่อนำโปรแกรมส่วนนี้ไปรวมเข้าด้วยกัน และเขียนส่วนรายละเอียดทั้งหมดเพิ่มเติมก็จะได้ระบบตรวจจับผู้บุกรุกทางเครือข่ายด้วยโครงข่ายใยประสาทเทียมที่เสร็จสมบูรณ์

4.2 การทดลองและผลการทดลอง

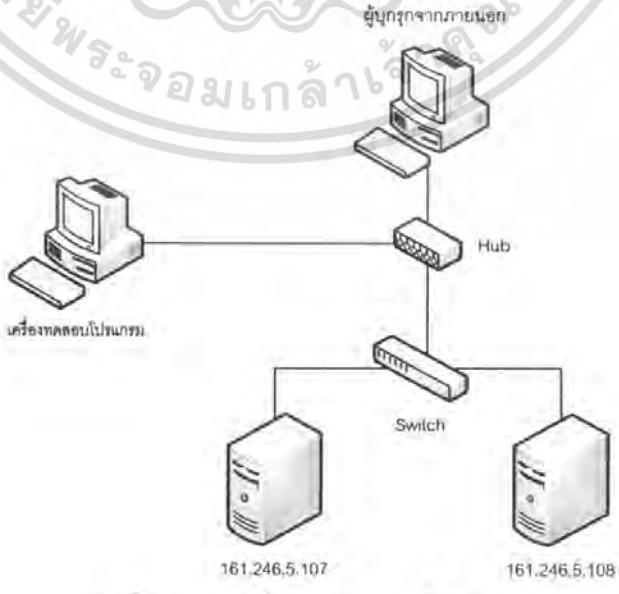
การทดลองส่วนต่อไปมีขึ้นเพื่อทดสอบการทำงานของโปรแกรมว่าสามารถตรวจจับการโจมตีเพื่อปิดบริการและการสแกนพอร์ตได้หรือไม่ นอกจากนี้ก็เป็นการทดสอบหาค่าการตรวจจับผิดพลาดที่เกิดขึ้นด้วย

4.2.1 การติดตั้งระบบทดสอบ

การทดสอบโปรแกรมทำโดยเตรียมอุปกรณ์ที่ประกอบไปด้วย

- คอมพิวเตอร์ที่ใช้ในการโจมตี 1 เครื่อง
- คอมพิวเตอร์ที่ใช้เป็นเหยื่อ 2 เครื่อง
- คอมพิวเตอร์ที่ใช้ทำงานโปรแกรม 1 เครื่อง
- ฮับ(Hub) 1 เครื่อง
- สวิทช์(Switch) 1 เครื่อง

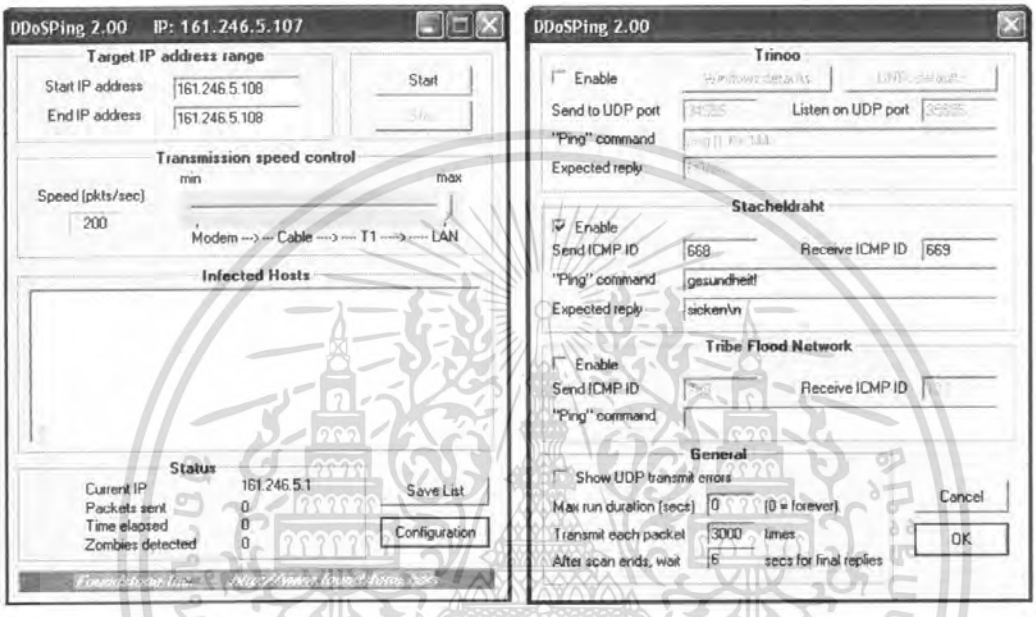
โดยคอมพิวเตอร์ทุกเครื่องจะต้องมีการ์ดเครือข่าย และทำการติดตั้งอุปกรณ์และตั้งค่าไอพีแอดเดรสตามความเหมาะสม ดังรูปที่ 6.7



รูปที่ 4.7 ระบบที่ทำการทดลองโปรแกรม

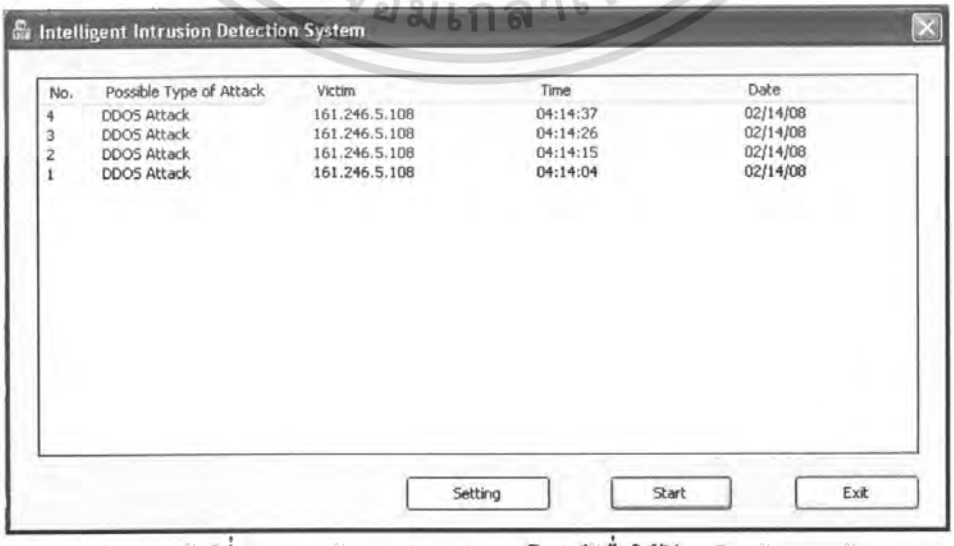
4.2.2 ทดสอบด้วยการโจมตีเพื่อให้ปิดบริการ (DDoS)

ทำการเริ่มทดสอบโปรแกรมโดยเริ่มจากการโจมตีเพื่อให้ปิดปริการ ซึ่งการทดสอบจะใช้โปรแกรม DDoSPing 2.0 ในการโจมตี ซึ่งการทดสอบจะทำการตั้งค่าเพื่อทำการโจมตีแบบ Stacheldrath จำนวน 6000 ครั้งโดยการตั้งค่าที่ Configuration ตามรูปที่ 6.8 แล้วกด Start



รูปที่ 4.8 แสดงตัวโปรแกรม DDoSPing 2.0 และการตั้งค่าเพื่อทดสอบ

จากการทดสอบโจมตีเพื่อให้ปิดบริการไปยังเครื่องหมายเลขไอพีแอดแดรส 161.246.5.108 ทำได้เสร็จในเวลา 30วินาที

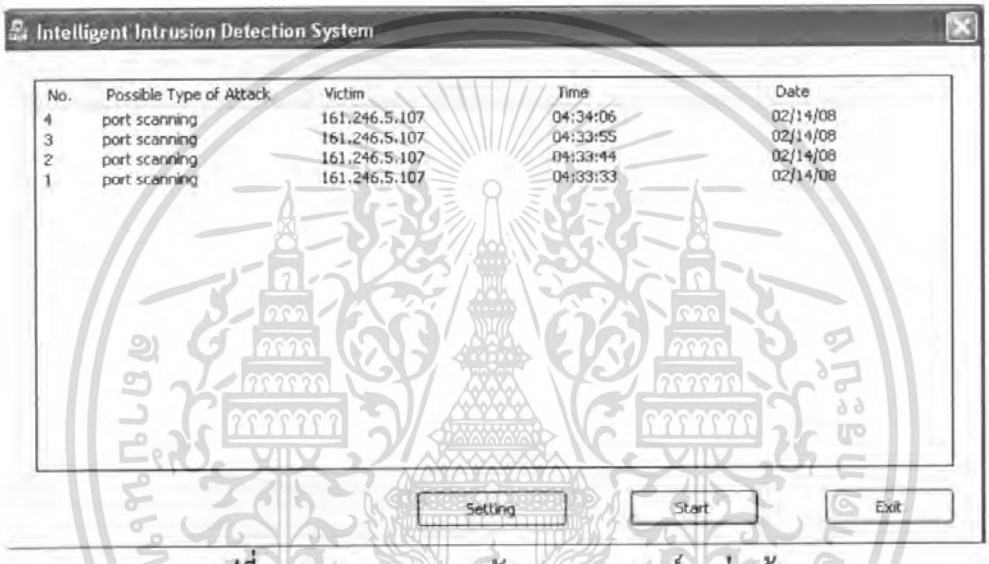


รูปที่ 4.9 แสดงการตรวจจับการโจมตีเพื่อให้ปิดบริการ

จากการทดลองพบว่าระบบสามารถตรวจจับการโจมตีเพื่อปิดบริการได้จริงซึ่งได้ผลการทดลองตามรูปที่ 4.9

4.2.3 ทดสอบการสแกนพอร์ตอย่างช้า (Polite Scan)

ทำการทดสอบโดยใช้โปรแกรม Nmap ทำการสแกนพอร์ตที่ไอพีแอดเดรส 161.246.5.107 โดยทำการสแกนพอร์ตอย่างช้าๆโดยใช้คำสั่ง nmap –sT –T2 161.246.5.107

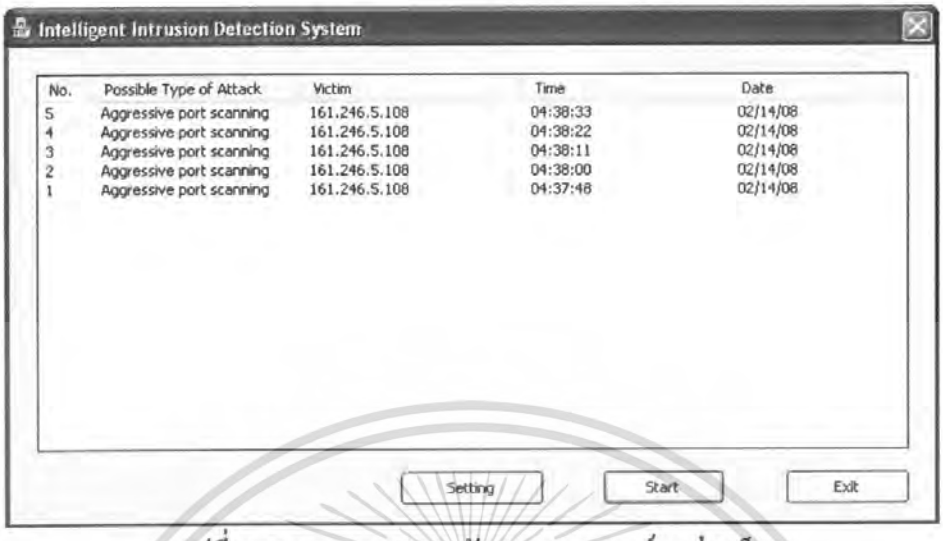


รูปที่ 4.10 แสดงการตรวจจับการสแกนพอร์ตอย่างช้า

จากการทดลองพบว่าระบบสามารถตรวจจับการสแกนพอร์ตอย่างช้าได้จริงซึ่งได้ผลการทดลองตามรูปที่ 4.10

4.2.4 ทดสอบการสแกนพอร์ตอย่างเร็ว (Aggressive Scan)

ทำการทดสอบด้วย Nmap ทำการสแกนพอร์ตไปยังเครื่องไอพีแอดเดรส 161.246.5.108 โดยการสแกนพอร์ตอย่างรวดเร็วโดยใช้คำสั่ง nmap –sT –T4 161.46.5.108

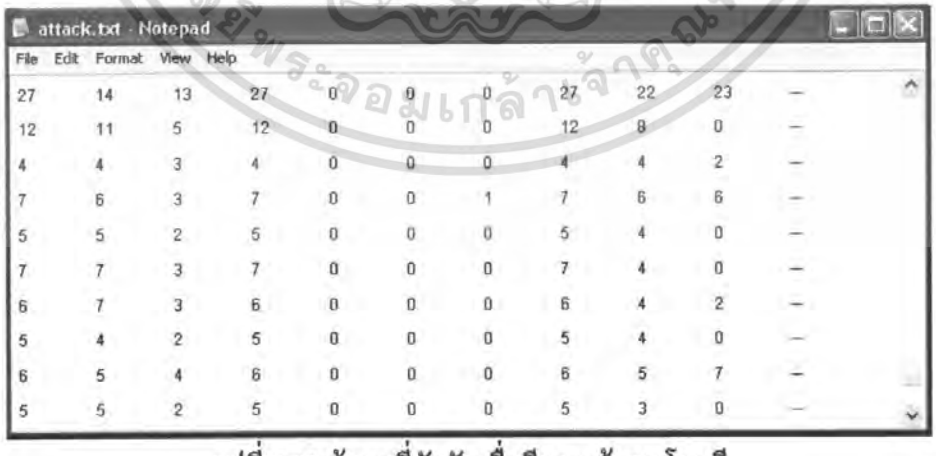


รูปที่ 4.11 แสดงการตรวจจับการสแกนพอร์ตอย่างเร็ว

จากการทดลองพบว่าระบบสามารถตรวจจับการสแกนพอร์ตอย่างเร็วได้จริงซึ่งได้ผลการทดลองตามรูปที่ 4.11

4.2.5 การวัดค่าการตรวจจับผิดพลาดที่เกิดขึ้น

ทำการทดลองด้วยการตรวจจับความผิดพลาด โดยการเพิ่มโปรแกรมให้ทุกครั้งที่มีการตรวจจับการโจมตีได้ ก็ให้โปรแกรมทำการเขียนค่าที่ดักจับได้จากเครือข่ายนั้นๆลงไฟล์ เพื่อนำมาตรวจสอบว่าเกิดการโจมตีขึ้นจริงหรือไม่ดังรูป



รูปที่ 4.12 ข้อมูลที่ดักจับเมื่อมีการแจ้งการโจมตี

เราทำการนำข้อมูลส่วนนี้ไปเปรียบเทียบกับข้อมูลที่เป็นข้อมูลเครือข่ายแบบปกติ และการโจมตีแยกเอาส่วนที่ไม่ใช่การโจมตีออกมาแล้วนับจำนวนเปรียบเทียบจากจำนวนทั้งหมด ที่ทำการตรวจจับหาการโจมตีตลอดช่วงเวลาจะได้ผลดังตารางที่ 4.1

ตารางที่ 4.1 แสดงผลการหาร้อยละของการตรวจจับผิดพลาด

เวลาที่ตรวจจับ	จำนวนครั้งทั้งหมดที่ตรวจจับ	จำนวนครั้งที่ตรวจพลาด	คิดเป็นร้อยละ
30 นาที	180 ครั้ง	15 ครั้ง	8.33 %
60 นาที	360 ครั้ง	25 ครั้ง	8.06 %
90 นาที	540 ครั้ง	46 ครั้ง	8.52 %
120 นาที	720 ครั้ง	59 ครั้ง	8.19 %
ค่าเฉลี่ยของการตรวจจับผิดพลาด			8.28 %

จากตารางจะเห็นว่าจำนวนครั้งของการตรวจจับผิดพลาด จะเพิ่มขึ้นตามเวลาที่มากขึ้นในการตั้งโปรแกรมเพื่อการตรวจจับ แต่เพิ่มในลักษณะคงที่และคำนวณค่าความผิดพลาดคิดเป็นร้อยละ เฉลี่ยแล้วอยู่ที่ 8.28 %

4.3 สรุปผลการทดลอง

ระบบสามารถตรวจจับและแสดงผลการโจมตีเพื่อปิดบริการได้ นอกจากนั้นยังสามารถตรวจจับการสแกนพอร์ตอย่างช้าและการสแกนพอร์ตอย่างเร็วได้ตามจุดประสงค์ที่ได้ตั้งไว้

การทำงานของระบบยังพบการตรวจจับที่ผิดพลาดเกิดขึ้นบ้าง ซึ่งการตรวจจับผิดพลาดที่เกิดขึ้นยังสามารถลดลงได้ด้วยการนำค่าที่มีการตรวจจับผิดพลาดนั้นๆส่งเข้าไปยังโครงข่ายใยประสาทเทียมให้เรียนรู้ว่าค่าดังกล่าวไม่ใช่การโจมตี ก็จะช่วยลดค่าน้ำหนักที่ข้อมูลเหล่านี้จะถูกตรวจจับออกมาอีกครั้งได้ หรือการศึกษาต่อเพื่อหาค่าใหม่ๆที่สามารถบ่งบอกถึงการโจมตีรูปแบบนั้นๆและนำมาเป็นอินพุต ให้กับโครงข่ายใยประสาทเพิ่มขึ้นจะช่วยให้แยกแยะกรณีเหล่านี้ออกไปจากการโจมตีได้ดียิ่งขึ้น

บทที่ 5
สรุปผลและวิจารณ์

5.1 บทสรุป

โครงงานนี้มุ่งพัฒนาระบบตรวจจับผู้บุกรุกทางระบบเครือข่ายแบบอะนอมอลลี บนระบบปฏิบัติการวินโดว์ เพื่อให้สามารถตรวจจับการโจมตีเพื่อให้ปิดบริการและการสแกนพอร์ต โดยใช้โครงข่ายใยประสาทเทียมในการวิเคราะห์ข้อมูลในเครือข่าย ผลที่ได้จากการทดลอง สามารถสร้างโปรแกรมที่สามารถตรวจจับผู้บุกรุกทางเครือข่าย โดยใช้โครงข่ายใยประสาทเทียม ในการวิเคราะห์จดจำรูปแบบปกติ และเมื่อเกิดการโจมตีได้ และเมื่อเกิดการโจมตีแล้ว จะทำการแจ้งให้ผู้ดูแลระบบทราบและมีการบันทึกลงล็อกไฟล์ไว้อ่านในภายหลังได้ตามจุดประสงค์ของโครงงาน แต่อย่างไรก็ตาม การใช้โครงข่ายใยประสาทเทียมมาวิเคราะห์ผลก็ยังมีปัญหาการตรวจจับผิดพลาดอยู่ได้ดังที่จะเสนอต่อไป

5.2 ปัญหาและอุปสรรค

ในการศึกษาการสร้างระบบตรวจจับผู้บุกรุกชาญฉลาดนี้ มีปัญหาและอุปสรรคในการพัฒนาหลายประการ ได้แก่

(1) เอกสารและตัวอย่างที่ถูกตีพิมพ์เกี่ยวกับ ระบบตรวจจับผู้บุกรุก ที่ใช้โครงข่ายใยประสาทเทียมนั้นมีอยู่น้อย

(2) เอกสารระบบตรวจจับผู้บุกรุกที่ใช้โครงข่ายใยประสาทเทียม ส่วนมากใช้โครงข่ายใยประสาทเทียมเพียงแต่ใช้แทนระบบอ้างอิงตามกฎในการวิเคราะห์เท่านั้น

(3) เอกสารที่สามารถหาได้ไม่มีข้อมูล ที่จะนำมาใช้เป็นข้อมูลการเรียนรู้ของโครงข่ายใยประสาทเทียม

(4) เนื่องจากใช้การตรวจจับแบบอะนอมอลลี จึงมีโอกาสเกิดการตรวจจับผิดพลาดขึ้นด้วย

5.3 แนวทางการพัฒนาต่อ

(1) เนื่องจากตัวระบบตรวจจับผู้บุกรุกนี้ใช้การตรวจจับแบบอะนอมอลลี ซึ่งมีโอกาสเกิดการตรวจจับผิดพลาดขึ้นด้วย โดยปัญหาตรงนี้สามารถลดลงได้อีก โดยการหาชุดข้อมูลที่บ่งบอกถึงการโจมตีเป็นอินพุตให้กับโครงข่ายใยประสาทเพิ่ม

(2) สามารถพัฒนาต่อเพื่อตรวจจับการโมตีรูปแบบอื่นๆ ที่เน้นการใช้ปริมาณได้ หรืออาจนำไปพัฒนาต่อเพื่อตรวจจับการใช้งานระบบเกินความจำเป็นได้

บรรณานุกรม

Zhimin Yang, Xiumei Wei, Luyan Bi, Dongping Shi, Hui Li. 2005. "An Intrusion Detection System Based on RBF Neural Network." Computer Supported Cooperative Work in Design, 2005. Proceeding of the Ninth International Conference on. 2(2). : 873-875

Andrew H. Sung, Srinivas Mukkamala. 2003. "Identifying Important Features for Intrusion Detection Using Support Vector Machines and Neural Networks." Application and the internet, 2003. Proceedings. 2003 Symposium on. : 209-216

Vladimir Golovko, Pavel Kochurko. 2005. "Intrusion Recognition Using Neural Networks." Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 2005. ISAACS 2005. IEEE. : 108-111

Jean-Philippe Planquart. 2003. "Application of Neural Networks to Intrusion Detection." GSEC Certification - version 1.2d. SANS Institute 2001

สุวัฒน์ ปุณณชัยยะ,ต้น ตัณฑ์สุทธิวงศ์ และสุพจน์ ปุณณชัยยะ . 2547. "เปิดโลก TCP/IP และโปรโตคอลของอินเตอร์เน็ต. Second Edition." กรุงเทพฯ : Provision.

นิรุธ อำนวยศิลป์. ม.ป.ป.. "Visual C++ and MFC Programming." กรุงเทพฯ : ดวงกมลสมัย.

Artificial Neural Network โครงข่ายประสาทเทียม. ม.ป.ป.. "Artificial Neural Network โครงข่ายประสาทเทียม." [ONLINE]. Available : http://202.28.94.55/web/320417/2548/work1/g26/Files/ Report_Neural%20Network.doc

WinPcap, The Packet Capture and Network Monitoring Library for Windows. [ONLINE]. Available : http://www.winpcap.org

MIT Lincoln Laboratory. "MIT Lincoln Laboratory - Intrusion Detection Attacks Database" - Intrusion Detection Attacks Database". [ONLINE]. Available : http://www.ll.mit.edu/IST/ideval/docs/1999/attackDB.html

Tutorials From FunctionX. "Visual C++ Tutorial - FunctionX". [ONLINE]. Available : http://www.functionx.com/visualc/index.htm

IET UOW Society. "Neural Network Library". [ONLINE]. Available : http://ieee.uow.edu.au/~daniel/software/libneural/

อมร วรรณพิณ, อังคาร ชูมงคล. 2547. "ระบบตรวจจับผู้บุกรุก Intrusion Detection System (IDS)." ปริญญานิพนธ์วิศวกรรมศาสตร์บัณฑิต ภาควิชาวิศวกรรมสารสนเทศ คณะวิศวกรรมศาสตร์,สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

ธนัญชัย ตรีภาค, วจี เทศวานิช. 2542. “ระบบตรวจจับผู้บุกรุกทางเครือข่ายคอมพิวเตอร์ Network Intrusion Detection System (NIDS).” ปริญญานิพนธ์วิศวกรรมศาสตร์บัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

ชาวดี บารมี, เมธี ชุมภูชา. 2544. “ระบบตรวจจับผู้บุกรุกเครือข่ายบน Win32 Network Intrusion Detection System for Win32.” ปริญญานิพนธ์วิศวกรรมศาสตร์บัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

ศราวุธ ฤทธิ์นันท์, ศักดิ์ชัย ฉายสุวรรณ. 2544. “ระบบตรวจจับผู้บุกรุกทางคอมพิวเตอร์ Intrusion Detection System.” ปริญญานิพนธ์วิศวกรรมศาสตร์บัณฑิต ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.