การจัดการโครงข่ายประสิทธิภาพสูงสำหรับกลุ่มวิจัยแบบไร้สาย

THE EFFICIENCY NETWORK MANAGEMENT FOR WIRELESS COMMUNICATION RESEARCH GROUP

โดย

นางสาว พรพิมล จินุพันธุ์

นางสาว พิมพ์ลักษณ์ กลิ่นมาลี

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

THE EFFICIENCY NETWORK MANAGEMENT
FOR WIRELESS COMMUNICATION RESEARCH GROUP

BY

Miss. Pornpimol Jinupun

Miss. Pimluck Klinmalee

A THESIS SUBMITTED IN PARTAIL FULFILLMENT OF

THE REQUIREMENT FOR THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2005

หัวข้อปริญญานิพนธ์	การจัดการโครงข่ายประสิทธิภาพสูงสำหรับกลุ่มวิจัยแบบไร้สาย
ชื่อนักศึกษา	นางสาวพรพิมล จินุพันธุ์ รหัสนักศึกษา 45010504 นางสาวพิมพ์ลักษณ์ กลิ่นมาลี รหัสนักศึกษา 45010540
อาจารย์ที่ปรึกษา	อาจารย์สถาพร พรหมวงศ์ ผศ.พิชญ สุพรรณกูล
ระดับการศึกษา	ปริญญาตรี วิศวกรรมบัณฑิต สาขาวิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2548

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้รับปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต

.....

(อาจารย์สถาพร พรหมวงศ์)

อาจารย์ผู้ควบคุมปริญญานิพนธ์

.....

(ผศ.พิชญ สุพรรณกูล)

อาจารย์ผู้ควบคุมปริญญานิพนธ์

หัวข้อปริญญานิพนธ์	การจัดการโครงข่ายประสิทธิภาพสูงสำหรับกลุ่มวิจัยแบบไร้สาย
ชื่อนักศึกษา	นางสาวพรพิมล จินุพันธุ์ รหัสประจำตัว 45010504
	นางสาวพิมพ์ลักษณ์ กลิ่นมาลี รหัสประจำตัว 45010540
อาจารย์ที่ปรึกษา	อาจารย์สถาพร พรหมวงศ์
	ผศ. พิชญ สุพรรณกูล
ระดับการศึกษา	ปริญญาตรีวิศวกรรมศาสตรบัณฑิต
	สาขาวิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2548

บทคัดย่อ

โครงงานนี้ได้ทำการศึกษาและวางระบบโครงข่ายที่มีประสิทธิภาพสำหรับกลุ่มวิจัยแบบสื่อสารไร้สาย วัตถุประสงค์ของโครงงานนี้คือเพื่อออกแบบและจัดการโครงข่ายให้มีประสิทธิภาพเสถียรภาพ และความปลอดภัยต่อข้อมูลระดับสูง ได้มีการออกแบบและติดตั้งไฟร์วอล (Firewall) DNS เซิร์ฟเวอร์ (DNS Server) เว็บเซิร์ฟเวอร์ (Web Server) และเซิร์ฟเวอร์ข้อมูลกลาง (FileServer) ไฟร์วอลล์ ใช้สำหรับป้องกันผู้บุกรุก DNS เซิร์ฟเวอร์ ใช้สำหรับสร้างโดเมนลูก (Child Domain) รวมทั้งยังได้ออกแบบโฮมเพจ (Home Page) และเว็บบอร์ด (Web Board) สำหรับบริการและแลกเปลี่ยนข้อมูลไว้ในเว็บเซิร์ฟเวอร์ เซิร์ฟเวอร์ฐานข้อมูลกลางใช้สำหรับเก็บข้อมูลวิจัยในกลุ่มวิจัยการสื่อสารไร้สาย

Thesis Title	The Efficiency Network Management For Wireless Communication Research Group		
Student	Miss Pompimol	Jinupun	ID.45010504
	Miss Pimluck	Klinmalee	ID.45010540
Advisor	Mr. Sathaporn	Promwong	
	Asst.Prof. Pitchaya	Supanakoon	
Graduate Level	Bachelor Degree of Information Engineering		
Department	Information Engineering		
Academic Year	2005		

Abstract

In this project, the efficiency network management for wireless communication research group is studied and managed. The purpose is to design and manage the network which is high efficiency, stable and security. The firewall, DNS, web and central database servers are designed and setup. The firewall is used for protecting the hackers. The DNS server is used for making the child domain. The home page and web board are design for servicing and exchanging acknowledge in the web server. The central database server is used for storing the research database inside the wireless communication research group.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้มิอาจสำเร็จลุล่วงไปได้ หากขาดความช่วยเหลือจากบุคคลต่าง ๆ ดังนั้นทางคณะผู้จัดทำจึงใคร่ขอขอบพระคุณบุคคลต่าง ๆ ดังต่อไปนี้

ขอขอบพระคุณ อาจารย์สถาพร พรหมวงศ์, ผศ. พิชญ สุพรรณกูล, พี่นิวัฒน์ และ พี่ ๆ ที่ศูนย์วิจัยคอมพิวเตอร์ที่คอยให้ความช่วยเหลือ และให้คำแนะนำ ตั้งแต่เริ่มทำโครงงาน จนสำเร็จออกมาเป็นปริญญานิพนธ์ฉบับนี้

นางสาวพรพิมล จินุพันธุ์
นางสาวพิมพ์ลักษณ์ กลิ่นมาลี
คณะผู้จัดทำ

สารบัญ

สารบัญ (ต่อ)

สารบัญ (ต่อ)

สารบัญรูป

สารบัญรูป (ต่อ)

สารบัญรูป (ต่อ)

สารบัญตาราง

บทนำ

บทที่ 1

1.1 แนวความคิดและที่มา

เนื่องจากการจัดการระบบเน็ตเวิร์คภายในห้องกลุ่มวิจัยแบบไร้สายแบบเดิม ยังคงขาดความปลอดภัยและเสถียรภาพ ทำให้กลุ่มวิจัยแบบไร้สายมีปัญหาในการใช้ระบบเครือข่ายภายในและภายนอก รวมถึงเรื่องความปลอดภัย จึงได้ออกแบบการวางระบบเน็ตเวิร์คภายในห้องกลุ่มวิจัยแบบไร้สายใหม่ โดยเน้นในเรื่องความปลอดภัย ความมีเสถียรภาพ รวมทั้งเว็บแอพพลิเคชั่น (Web Application)

1.1.1 ความปลอดภัย

ในเรื่องความปลอดภัยแบบเดิมได้พบปัญหาหลายอย่าง เช่น ใช้ Public IP เพียงอย่างเดียว ซึ่งจำนวนหมายเลขเน็ตเวิร์คไม่เพียงพอต่อความต้องการของผู้ใช้ นอกจากนี้ยังทำให้ความปลอดภัยในระบบลดลงด้วย และเกิดปัญหาในเรื่องไอพีชนกันเนื่องจากผู้ใช้ได้ทำการคอนฟิกไอพีกันเอง รวมถึงการจัดสรรไอพีที่ไม่เป็นระบบเป็นต้นจึงได้คิดพัฒนาโดยการทำการออกแบบระบบเน็ตเวิร์คใหม่ให้มีความปลอดภัยมากขึ้น

1.1.2 เสถียรภาพ

ในเรื่องเสถียรภาพนั้นแบบเดิมใช้ Single Box เพียงตัวเดียวซึ่งทำหน้าที่เป็นเกตเวย์ไฟล์วอล เมื่อพังทำให้ผู้ใช้ภายในไม่สามารถต่อเชื่อมออกเน็ตเวิร์คภายนอกได้ จึงทำการออกแบบให้มีเสถียรภาพมากขึ้น

1.1.3 ความปลอดภัยต่อข้อมูลภายในเซิรฟเวอร์

แบบเดิมภายในห้องวิจัยแบบไร้สายนั้นไม่มีการจัดการใดๆเกี่ยวกับข้อมูลของอาจารย์และนักศึกษาซึ่งเป็นข้อมูลที่มีความสำคัญระดับสูง จึงได้คิดพัฒนาให้มีการจัดเก็บเป็นระบบมากขึ้นโดยทำเป็นฐานข้อมูล ให้มีความปลอดภัยระดับสูง

1.1.4 เว็บแอพพลิเคชั่น

แต่ก่อนนั้นภายในห้องวิจัยไม่มีระบบบริการภายในห้องทำให้เกิดความยุ่งยากต่างๆ เช่น การดาวน์โหลดงานวิจัย เป็นต้น จึงได้คิดสร้างเว็บไซต์ภายในห้องวิจัยแบบไร้สายขึ้นเพื่อให้มีแจ้งข่าวสาร, ความรู้ต่างๆ รวมถึงทำการสร้างเว็บบอร์ด (Webboard) เพื่อให้ผู้ใช้ภายในห้องวิจัยได้ติดต่อหรือตั้งกระทู้ถาม- ตอบเกี่ยวกับข่าวสาร หรือ วิชาการความรู้ภายในห้องวิจัย

1.2 จุดประสงค์

- เพื่อพัฒนาการจัดระบบเน็ตเวิร์คภายในห้องกลุ่มวิจัยแบบไร้สาย
- เพื่อป้องกันการเข้าถึงระบบเครือข่ายจากภายนอก
- เพื่อเก็บข้อมูลจากการทำ thesis ของนักศึกษา ให้เป็นระบบ
- เพื่อแลกเปลี่ยนความรู้ ข่าวสาร ภายในและภายนอกกลุ่มวิจัย

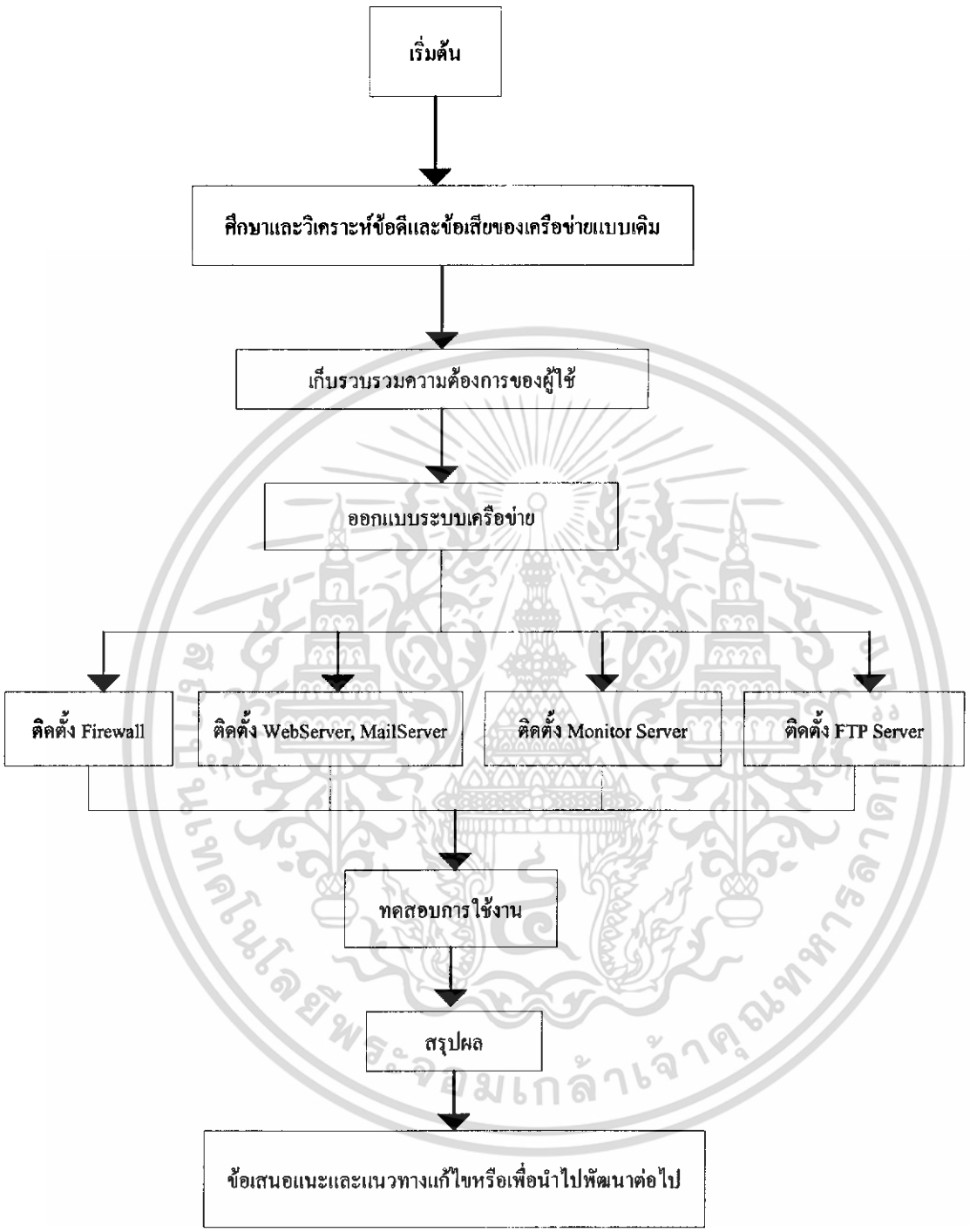
1.3 ขอบเขตโครงงาน

- ศึกษาระบบเครือข่ายเดิม ว่ามีส่วนดีและส่วนบกพร่องอย่างไร รวมถึงศึกษาความต้องการและการใช้งานในระบบเครือข่ายของผู้ใช้
- ทำการวางแผน ออกแบบ และวิเคราะห์ ระบบเน็ตเวิร์คให้มีประสิทธิภาพ โดยมีการติดตั้งไฟร์วอล(Firewall) ออกแบบโฮมเพจ (Home Page) และเว็บบอร์ด
- เพิ่มระบบความปลอดภัยของเซิร์ฟเวอร์ฐานข้อมูลกลาง (Central Database Server)
- ทำการคอนฟิก (configure) อุปกรณ์ต่าง ๆ ที่ใช้ในการวางระบบเน็ตเวิร์ค

1.4 ผลที่คาดว่าจะได้รับ

- ระบบเน็ตเวิร์คที่พัฒนาขึ้นในห้องกลุ่มวิจัยไร้สายมีประสิทธิภาพและเสถียรภาพมากขึ้น สามารถป้องกันการเข้าถึงระบบเน็ตเวิร์คภายในจากภายนอกได้
- ข้อมูลที่ได้จากการทำ thesis ของนักศึกษาถูกเก็บเป็นระบบมากขึ้น
- ผู้ใช้สามารถแลกเปลี่ยนความรู้ ข่าวสาร ภายในและภายนอกกลุ่มวิจัยได้

1.5 ขั้นตอนดำเนินการ



รูปที่ 1.1 ขั้นตอนการดำเนินการ

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 บทนำ

ในบทนี้ได้กล่าวถึงทฤษฎีเกี่ยวกับเครือข่าย ซึ่งประกอบไปด้วย อันดับแรกคือ ไฟล์วอล ซึ่งหน้าที่รักษาความปลอดภัยให้กับเครือข่าย อันดับที่สองคือ DHCP ทำหน้าที่กำหนดไอพีแอดเดรสแบบอัตโนมัติให้แก่เครื่องไคลเอ็นต์ในระบบ อันดับที่สามคือ DNS Server (Domain Name System) ทำหน้าที่แปลงหมายเลขแอดเดรสให้อยู่ในรูปแบบของโดเมนเนม (Domain Name) หรือ แปลงกลับจากโดเมนเนมไปเป็นไอพีแอดเดรส อันดับที่สี่คือ NAT (Network Address Translation) เป็นวิธีการหนึ่งในการแปลงและแปลไอพีแอดเดรสของเครือข่ายภายในให้เป็นไอพีแอดเดรส ซึ่งเป็นที่ยอมรับและสื่อสารบนอินเทอร์เนต อันดับที่ห้าคือ IPtables เป็นโปรแกรมโอเพ่นซอร์ส สำหรับควบคุมกฎการเข้า-ออกแพ็กเก็ตข้อมูล ในที่นี้จะเป็นส่วนหนึ่งในไฟร์วอลเพื่อเพิ่มความปลอดภัยให้แก่ระบบเครือข่ายระบบเครือข่าย อันดับที่หกคือ Webserver(เว็บเซิร์ฟเวอร์) โดยมีการอัพโหลด (upload) โฮมเพจของห้องกลุ่มวิจัยแบบไร้สายไว้ เพื่อความสะดวกในการใช้งานของผู้ใช้เกี่ยวกับแอพพลิเคชั่นต่างๆ โดยโฮมเพจและแอพพลิเคชั่นต่างๆได้ใช้ PHP Hypertext Preprocessor (PHP) ในการสร้างขึ้นเพื่อติดต่อระหว่างฝั่งเซิร์ฟเวอร์และไคลเอ็นต์ (Client) อันดับที่เจ็ดคือ Mail Server (เมล์เซิร์ฟเวอร์) ซึ่งใช้ในการรับส่ง Email (อีเมล์)ในเครือข่ายอินเตอร์เน็ต และอันดับที่แปดคือ FTP Server (เอฟทีพี เซิร์ฟเวอร์) เป็นการถ่ายโอนแฟ้มข้อมูลระหว่างเครื่องคอมพิวเตอร์ 2 เครื่อง ซึ่งอยู่บนเครือข่ายอินเทอร์เน็ต

2.2 ไฟร์วอล

ไฟร์วอล เป็นเครื่องมือรักษาความปลอดภัยให้กับเครือข่ายภายใน โดยป้องกันผู้บุกรุก (Intrusion) ที่มาจากเครือข่ายภายนอก ถ้าผู้บุกรุกมาจากเครือข่ายภายใน ระบบนี้จะไม่สามารถป้องกันได้ สิ่งที่สามารถป้องกันได้ เช่น ไวรัสคอมพิวเตอร์ (Virus), หนอนคอมพิวเตอร์ (Worm), การโจมตีแบบ DoS (Denial of Service), ม้าโทรจัน (Trojan House), IP Spoofing ฯลฯ

โดยมีลักษณะการบุกรุก ดังนี้

- ไวรัสคอมพิวเตอร์ จะแย่งใช้หรือ ทำลายทรัพยากรของคอมพิวเตอร์ เช่น ไฟล์ข้อมูล, แรม
- หนอนคอมพิวเตอร์ จะแย่งใช้ทรัพยากรของคอมพิวเตอร์ เช่น เขียนไฟล์ขยะลงบนฮาร์ดดิสก์ จนทำให้ฮาร์ดดิสก์เต็ม

- DoS จะร้องขอบริการ (Service) ต่างๆ จากเซิร์ฟเวอร์ จนทำให้เซิร์ฟเวอร์ล่ม
- ม้าโทรจัน จะซ่อนอยู่ในเครื่องไคลเอนต์หรือเซิร์ฟเวอร์ เมื่อถึงเวลามันจะทำการเปิดพอร์ตของเครื่องนั้นให้กับผู้บุกรุก เช่น แฮกเกอร์สามารถรีโหมดเข้ามาควบคุมการทำงานของเครื่องนั้นได้
- IP Spoofing การปลอมหมายเลขไอพีต้นฉบับเพื่อลักลอบเข้ามาในเครือข่าย

ไฟร์วอลมีทั้งชนิดฮาร์ดแวร์ เช่น Router ที่ประกอบด้วยฟังก์ชั่น Screening Device, Layer 3 Switch และซอฟท์แวร์ ที่เป็น Ipchains, Iptables

วิธีการของไฟร์วอล คือ จำกัดให้มีการผ่านเข้าออกแพ็คเก็ตข้อมูลได้ที่จุดเดียว (Controlled point) และป้องกันผู้บุกรุกที่พยายามจะเข้ามาในเครือข่าย โดยทำการวิเคราะห์ว่าจะอนุญาตให้แพ็กเก็ตนั้นผ่านไปหรือไม่ ตามกฎที่กำหนดไว้

สิ่งที่ไฟร์วอลสามารถทำได้

- รักษาความปลอดภัย เนื่องจากเป็นจุดเดียวที่เครือข่ายภายในติดต่อกับเครือข่ายภายนอก
- สามารถตรวจสอบ และเก็บล็อกเหตุการณ์ต่าง ๆ ระหว่างการติดต่อของเครือข่ายภายในและ เครือข่ายภายนอก
- สามารถกำหนดกฎ ในการอนุญาต หรือไม่อนุญาตในการใช้บริการต่าง ๆ ภายในเครือข่าย

สิ่งที่ไฟร์วอลไม่สามารถทำได้

- ไฟร์วอลไม่สามารถป้องกันผู้บุกรุกที่อยู่ภายในเครือข่าย (Internal Network)
- ไฟร์วอลไม่สามารถป้องกันการโจมตีที่ไม่ได้ส่งผ่านไฟร์วอล เช่น ผู้ใช้ภายในเครือข่ายมีการเชื่อมต่อกับอินเทอร์เน็ตในทางอื่นซึ่งไม่ผ่านไฟร์วอล โดยที่ผู้ดูแลระบบไม่รับทราบ เช่น การ Dial-up ไปยังอินเทอร์เน็ตจากเครื่องคอมพิวเตอร์ส่วนบุคคลที่อยู่ภายในเครือข่าย

ไฟร์วอลแบ่งออกได้ 3 ประเภท

1. Packet Filtering เป็นไฟร์วอลระดับพื้นฐาน ใช้สำหรับตรวจสอบ ไอพีแอดเดรส และพอร์ทที่อยู่ต้นทางและปลายทาง รวมทั้งกรองแพ็กเกตข้อมูล ทั้ง TCP, UDP ได้

2. Circuit-Level Firewall เป็นไฟร์วอลประเภทพร็อกซีเซิร์ฟเวอร์ ที่เป็นตัวคั่นกลางระหว่างเครือข่ายภายในกับเครือข่ายภายนอก โดยใช้เทคนิค SPI (Stateful Packet Inspection)

ซึ่งมีหลักการทำงานเช่นเดียวกับ Packet Filtering และสามารถเพิ่มกฎในการเข้าถึง (Access Rules) เพื่อใช้ในการควบคุมทราฟฟิกได้

3. Application Level Firewall เป็นไฟร์วอลประเภท พร็อกซีเซริฟเวอร์ ที่ทำงานระดับแอพพลิเคชั่น มีหน้าที่ป้องกันเครือข่ายภายในกับเครือข่ายภายนอกไม่ให้ติดต่อกันโดยตรง การร้องขอ (Request) และการตอบกลับ (Response) ต้องกระทำผ่าน พร็อกซีเซิร์ฟเวอร์

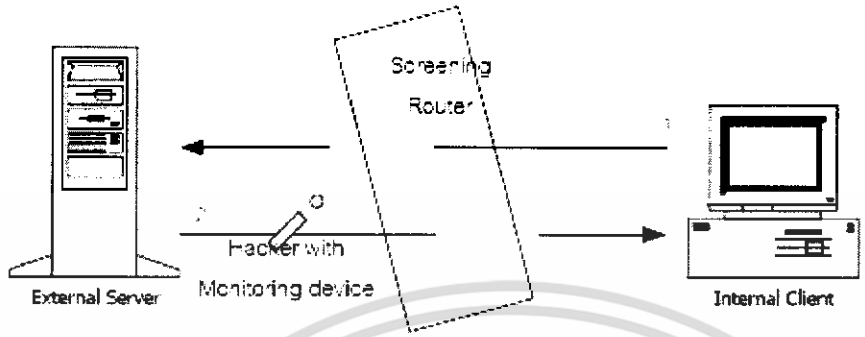
2.2.1 รูปแบบการทำงานของไฟร์วอล

สามารถแบ่งรูปแบบการทำงานของไฟร์วอลได้ 3 ประเภท ตามกลวิธีในการป้องกันเครือข่าย สำหรับการทำงานบนเราเตอร์ในเลเยอร์ระดับล่าง จะใช้วิธีการกรองแพ็คเก็ต โดยอาศัยข้อมูลในส่วนเฮดเดอร์ เรียกว่า "แพ็คเก็ตฟิลเตอร์ริ่ง" (Packet Filtering) ส่วนการทำงานในเลเยอร์ระดับสูง จะใช้พร็อกซีเซิร์ฟเวอร์ในการตรวจสอบเนื้อหาภายในแพ็กเก็ตและแสดงผลการตรวจสอบ เรียกว่า "พร็อกซีเซิร์ฟเวอร์เกตเวย์ " และประเภทสุดท้าย ใช้สำหรับเก็บสถานะการทำงานไว้เป็นลำดับขั้น เรียกว่า "สเตทฟูลอินสเป็คชัน"

2.2.1.1 แพ็กเก็ตฟิลเตอร์ริ่ง

เป็นวิธีที่ใช้กันอย่างแพร่หลาย หลักการทำงานคือ ไฟร์วอลจะทำการตรวจสอบแพ็คเก็ตโดยพิจารณาแพ็กเก็ตที่เข้าออกตามกฎที่ตั้งไว้ และวิเคราะห์แพ็คเก็ตเหล่านี้จากเฮดเดอร์ของแพ็คเก็ตนั้น ๆ เช่น แอดเดรสต้นทาง แอดเดรสปลายทาง พอร์ทหรือโพรโตคอล เมื่อแพ็คเก็ตแต่ละแพ็คเก็ตเข้ามาสู่ไฟร์วอล จะนำมาเทียบกับกฎที่ตั้งไว้ หากเข้ากับกฎใดกฎหนึ่ง ก็พิจารณาว่ากฎนั้นมีจุดมุ่งหมายอย่างไรเช่น อนุญาตให้แพ็กเก็ตผ่านไปได้หรือดร็อปแพ็คเก็ตนั้นทิ้งไป และหากไม่เข้ากับกฎใด จะพิจารณาค่าดีฟอลต์ ว่าอนุญาตให้แพ็กเก็ตผ่านไปได้หรือไม่อนุญาตให้ผ่านไป ซึ่งเป็นกระบวนการทำงานในชั้นเน็ตเวิร์ก (Network Layer) ซึ่งวิธีการนี้เป็นวิธีที่ง่ายและทำงานได้รวดเร็วที่สุด แต่จุดอ่อนคือ ความยากในการกำหนดกฎต่าง ๆ ให้รัดกุม และการติดต่อกันระหว่างโฮสต์ต้นทางและโฮสต์ปลายทางได้โดยตรง อาจส่งผลให้ข้อมูลต่าง ๆ ของโฮสต์ปลายทาง และโฮสต์อื่น ๆ

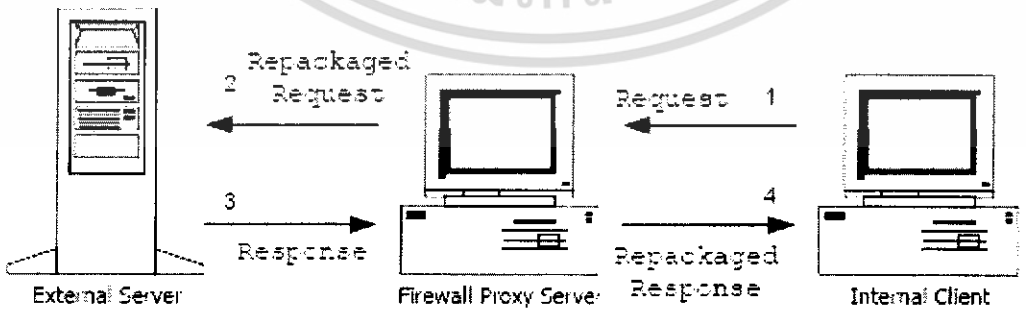
ที่ติดต่ออยู่กับโฮสต์ปลายทางถูกโจมตีได้ ไฟร์วอลชนิดนี้สามารถต่อต้านการโจมตีได้หลายชนิด เช่น IP Spoofing, Source Routing Attack, Tiny Fragmentation Attack



รูปที่ 2.1 รูปแบบการทำงานของแพ็คเก็ตฟิลเตอร์ริ่ง

2.2.1.2 พร็อกซีเซิร์ฟเวอร์เกตเวย์ (Proxy-Server Gateway)

พร็อกซีทำงานในชั้นเลเยอร์บน เป็นโปรแกรมแอปพลิเคชั่นที่ทำงานอยู่ระหว่างสองเครือข่าย โดยจะทำงานในลักษณะของการส่งข้อมูลต่อ เมื่อแพ็คเก็ตมาถึง พร็อกซีจะแยกข้อมูลส่วนที่เป็นเฮดเดอร์ของแพ็กเก็ตออก เหลือเพียงข้อมูลในของชั้นแอพพลิเคชั่น เช่น กรณีเป็นข้อมูลของเว็บ พร็อกซีจะแยกเฮดเดอร์ออกเหลือเพียงส่วนโพรโตคอล HTTP จากนั้นก็จะนำเฮดเดอร์มาพิจารณาเทียบกับกฎต่าง ๆ ที่กำหนดไว้ หากตรงตามกฎที่อนุญาตให้ส่งต่อ จะนำข้อมูล HTTP นั้นมาประกอบเป็นแพ็กเก็ตขึ้นมาใหม่ และส่งต่อไปยังเป้าหมาย การทำงานลักษณะเช่นนี้ ทำให้การติดต่อระหว่างผู้ใช้ภายในกับภายนอกไม่ต้องติดต่อกันโดยตรง แต่พร็อกซีต้องรับภาระการทำงานอย่างหนัก ทำให้ประสิทธิภาพในการติดต่อระหว่างเครือข่ายลดลง ทำงานช้า แต่มีความปลอดภัยมากกว่าเนื่องจากมีการพิจารณาข้อมูลถึงระดับแอปพลิเคชั่น แต่สามารถส่งต่อได้เฉพาะโพรโตคอลที่ไฟร์วอลรู้จักเท่านั้น



รูปที่ 2.2 รูปแบบการทำงานของพร็อกซีเซิร์ฟเวอร์เกตเวย์

พร็อกซีเซิร์ฟเวอร์เกตเวย์แบ่งออกได้เป็น 2 ประเภท

- เซอร์กิต-เลเวล เกตเวย์ (Circuit – Level Gateway)

เป็นพร็อกซีที่ทำหน้าที่ควบคุมการติดต่อระหว่างเครือข่ายภายในและภายนอก โดยปราศจากช่องว่าง กล่าวคือ จะมีวงจรเสมือน (Virtual Circuit) อยู่ระหว่างโฮสต์ภายในเครือข่ายกับพร็อกซีเซิร์ฟเวอร์ เมื่อมีการร้องขอการติดต่อ (Request) จากเครือข่ายภายใน แพ็คเก็ตจะถูกส่งให้วงจรเสมือนผ่านไปยังพร็อกซีเซิร์ฟเวอร์ ซึ่งจะส่งการร้องขอการติดต่อไปยังอินเทอร์เน็ต หลังจากทำการแปลงหมายเลขไอพีเรียบร้อยแล้ว ในทำนองเดียวกัน การตอบรับ(Response)จากอินเทอร์เน็ต จะถูกส่งมายังพร็อกซีเซิร์ฟเวอร์ผ่านวงจรเสมือน ก่อนส่งกลับให้โฮสต์ต้นทาง โดยในการติดต่อนี้ เครือข่ายภายนอกจะไม่สามารถมองเห็นโฮสต์ใดๆ ภายในเครือข่ายได้เลย การติดต่อแบบนี้จะใช้ในกรณีที่ผู้ใช้ภายในเครือข่ายกับอินเทอร์เน็ตไว้ใจได้เท่านั้น

- แอปพลิเคชั่น-เลเวล เกตเวย์ (Application – Level Gateway)

สำหรับแอพพลิเคชั่นเกตเวย์ นอกจากจะทำงานเช่นเดียวกับวงจรเสมือนแล้ว ยังเพิ่มความสามารถในการตรวจสอบแพ็กเก็ต ทั้งส่วนเฮดเดอร์และส่วนเนื้อหาของข้อมูลภายในแพ็คเก็ต เพื่อหยุดการส่งข้อมูลจากภายนอก และยังสนับสนุนการให้บริการ สำหรับโพรโตคอลแบบต่างๆ คือ Telnet, FTP, HTTP และ SMTP

2.2.1.3. สเตทฟูลอินสเป็คชั่นหรือไดนามิกแพ็คเก็ตฟิลเตอร์ริ่ง

เป็นรูปแบบการทำงานแบบใหม่ ซึ่งพัฒนาจากแพ็คเก็ตฟิลเตอร์ริ่ง ซึ่งเดิมการตัดสินใจ จะพิจารณาจากข้อมูลในส่วนเฮดเดอร์เท่านั้น และพิจารณาเฉพาะแพ็กเก็ตนั้น ๆ โดยไม่ได้พิจารณาความสัมพันธ์ของแพ็กเก็ตก่อนหน้า สเตทฟูลอินสเป็คชั่นพัฒนาโดยการพิจารณาทั้งในส่วนของข้อมูลที่ใช้ในการตัดสินใจ โดยพิจารณาข้อมูลในส่วนของ Payload และจะพิจารณาความสัมพันธ์ของแพ็คเก็ตที่อยู่ก่อนหน้านี้ประกอบการวิเคราะห์ โดยบันทึกสถานะของข้อมูล เช่น เมื่อมีการส่งข้อมูล ตารางสถานะจะบันทึกหมายเลขพอร์ทต้นทาง หมายเลขพอร์ทปลายทาง เรียกว่า “Saving the state” เมื่อมีแพ็คเก็ตตอบรับ จะนำแพ็กเก็ตที่รับมานั้น เปรียบเทียบกับ “Saving state” ที่บันทึกไว้ว่ามีข้อมูลตรงกันหรือไม่ การทำงานของ สเตทฟูลอินสเป็คชั่น จะมีการติดตามสถานะ (State) การทำงานของการเชื่อมต่อแบบ TCP ทำให้สามารถพิจารณารูปแบบการทำงานได้ทุกกระบวนการ ไม่ได้พิจารณาเพียงข้อมูลในแต่ละแพ็คเก็ต นอกจากนี้สเตทฟูลอินสเป็คชั่นยังมีความปลอดภัยมากกว่า เนื่องจากสามารถปิดพอร์ตที่มีหมายเลขมากกว่า 1,024 ได้ เช่น การเชื่อมต่อแบบ TCP โดยเว็บนั้น เริ่มแรกจะติดต่อกันโดยผ่านพอร์ต 80 แต่หลังจากที่ติดต่อสำเร็จ จะมีการใช้หมายเลข

พอร์ตแบบสุ่ม โดยมีหมายเลขมากกว่า 1,024 ซึ่งทำให้ไฟร์วอลแบบแพ็คเก็ตฟิลเตอร์ริ่งจำเป็นต้องเปิดพอร์ตที่มีหมายเลขมากกว่า 1,024 ตลอดเวลา แต่สเตทฟูลจะเปิดพอร์ตเฉพาะเวลาที่มีการเชื่อมต่อผ่านพอร์ตนั้น ๆ เท่านั้น หากแพ็กเก็ตถูกตรวจสอบแล้วว่าไม่เป็นไปตามกฎก็จะปิดพอร์ตนั้นทันที สเตทฟูลอินสเปคชั่น มีข้อจำกัด คือ ไม่รู้จักการทำงานในระดับแอพพลิเคชั่น และยังคงเป็นการติดต่อระหว่างผู้ใช้ภายในกับภายนอกโดยตรง

ตารางที่ 2.1 เปรียบเทียบรูปแบบการทำงานของไฟร์วอลทั้ง 3 ประเภท

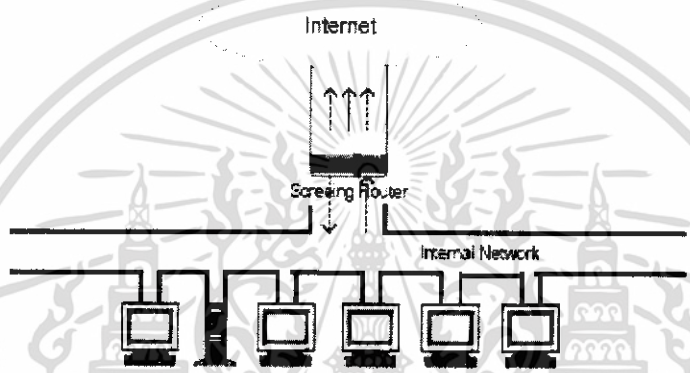
	Packet Filter	Stateful Inspection	พร็อกซีเซิร์ฟเวอร์ Gateways
ข้อดี	• ประสิทธิภาพดี • ง่ายในการ implement • ไม่ขึ้นกับแอปพลิเคชั่น (Application Independent)	• ประสิทธิภาพดี • เปิดพอร์ตเฉพาะเมื่อมีการติดต่อ • สนับสนุนเกือบทุกบริการ	• ไม่เปิดเผยหมายเลขไอพีภายใน • พิจารณาเนื้อหาข้อมูลด้วย • มี User Authentication • เก็บรายละเอียด log ได้มาก
ข้อเสีย	• เปิดเผยหมายเลขไอพีภายใน • มีการเปิดช่องว่างทิ้งไว้ถาวร • No User Authentication • ใช้การเชื่อมต่อโดยตรงกับภายนอก	• No User Authentication • ใช้การเชื่อมต่อโดยตรงกับภายนอก • เปิดเผยหมายเลขไอพีภายใน	• ประสิทธิภาพต่ำกว่า • ต้องมีพร็อกซี สำหรับทุกๆแอปพลิเคชั่นที่ใช้ • ไม่มีการป้องกันในระดับชั้นที่ต่ำกว่าชั้นแอปพลิเคชั่น • เปิดเผยระบบปฏิบัติการ

2.2.2 สถาปัตยกรรมของไฟร์วอล

2.2.2.1 ซิงเกิลบ็อกซ์ (Single Box)

- สกรีนนิ่ง เราเตอร์ (Screening Router)

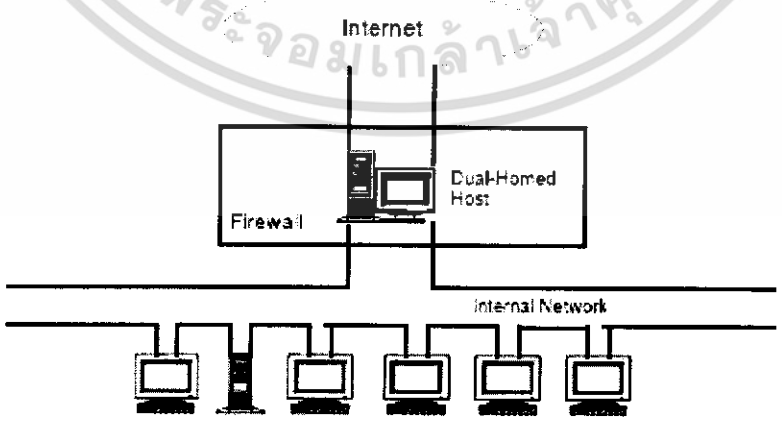
สามารถกรองแพ็กเก็ตตามกฎที่ตั้งไว้ โดยพิจารณาจากข้อมูลในส่วนเฮดเดอร์ของไอพีเท่านั้น เช่น อนุญาตหรือไม่อนุญาตพอร์ตหรือไอพีแอดเดรสใด ไฟร์วอลลักษณะนี้เป็นการเชื่อมต่อระหว่างเครือข่ายภายในและเครือข่ายภายนอก ดังนั้นหากมีการโจมตีที่ไฟร์วอล ซึ่งเป็นตัวหน้าด่านเพียงตัวเดียว หากการโจมตีสามารถผ่านไฟร์วอลได้จะสามารถเข้ามาในเครือข่ายภายในได้



รูปที่ 2.3 สกรีนนิ่ง เราเตอร์

- ดูอัล – โฮม โฮสต์ (Dual-Home Host)

เป็นคอมพิวเตอร์ที่เชื่อมต่ออยู่กับเครือข่ายอย่างน้อย 2 เครือข่าย สามารถทำหน้าที่เป็นเราเตอร์ได้ แต่ไม่อนุญาตให้มีการติดต่อผ่านกันโดยตรงโดยไม่ผ่านดูอัล-โฮมโฮสต์ ทำให้คอมพิวเตอร์ต้องรับภาระหนัก จึงเหมาะกับเครือข่ายที่มีข้อมูลเข้าออกน้อย ๆ เนื่องจากสามารถรักษาความปลอดภัยได้มากกว่าสกรีนนิ่งเราเตอร์

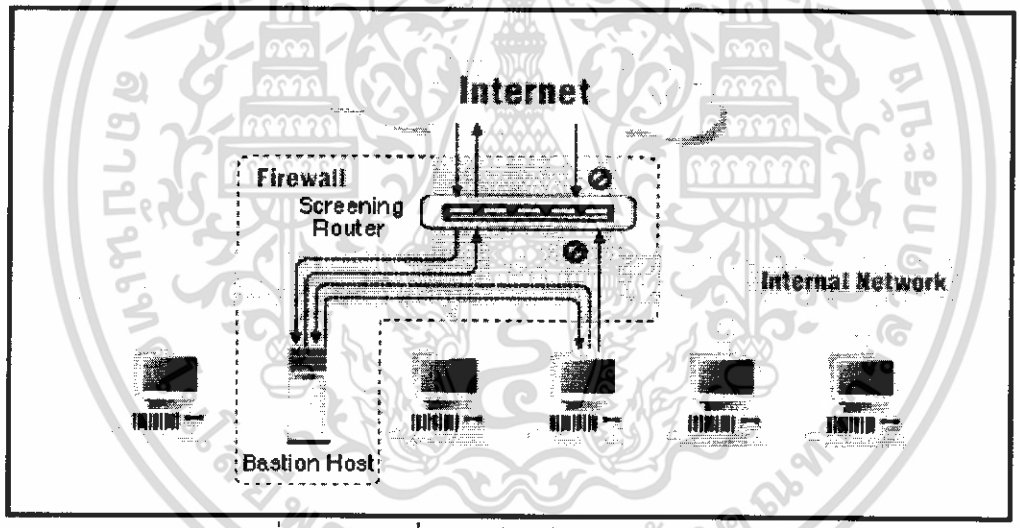


รูปที่ 2.4 สกรีนนิ่งโฮสต์ อาร์คิเทคเจอร์

2.2.2.2 มัลติเพิล-เพอโพส บ็อกซ์

- สกรีนนิ่งโฮสต์ อาร์คิเทคเจอร์ (Screen Host Architecture)

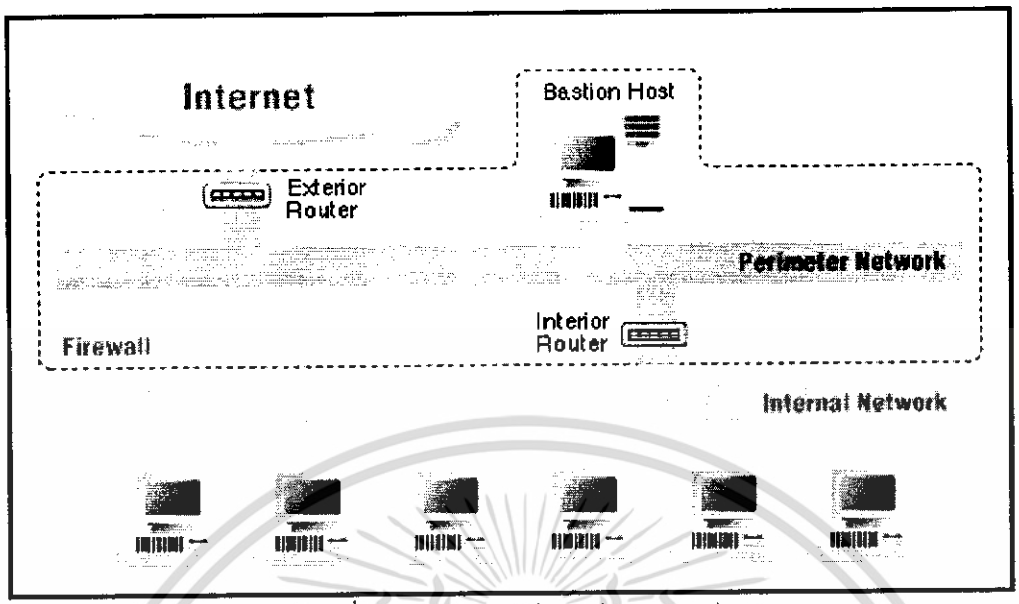
ประกอบไปด้วยไฟร์วอลประเภทแพ็กเก็ตฟิลเตอร์ริ่งและเบสชันโฮสต์(BastionHost) โดยเบสชันโฮสต์เป็นคอมพิวเตอร์พิเศษ ที่อนุญาตให้มีการเชื่อมต่อกับเครือข่ายภายนอก ซึ่งจะทำหน้าที่เป็นพร็อกซีสำหรับส่งแพ็กเก็ตต่อไปยังเครื่องอื่น ๆ ภายในเครือข่าย ดังนั้นหากมีการบริการใด ๆในระบบเบสชันโฮสต์จะต้องรู้จักโพรโตคอลนั้น และเบสชันโฮสต์จะต้องมีการพิสูจน์สิทธิ์ (Authentication) ที่เหมาะสม การเริ่มใช้งานนั้น แพ็กเก็ตจะถูกกรองโดยแพ็คเก็ตฟิลเตอร์ริ่งเราเตอร์ก่อนตามกฎที่กำหนดว่าควรจะอนุญาตหรือไม่อนุญาต จากนั้นทำการร้องขอพิสูจน์สิทธิ์กับเบสชันโฮสต์ และให้เบสชันโฮสต์เป็นตัวกลางจัดการเชื่อมต่ออีกครั้งหนึ่ง ข้อเสียของโครงสร้างแบบนี้ขึ้นอยู่กับการทำงานของฟิลเตอร์ริ่งเราเตอร์ หากมีการทำงานผิดพลาด เบสชันโฮสต์จะให้บริการทุกการเชื่อมต่อ ทำให้เครือข่ายภายในถูกโจมตีได้ จึงไม่ควรให้บริการที่มีความเสี่ยงสูง เช่น เว็บเซิร์ฟเวอร์



รูปที่ 2.5 สกรีนนิ่งโฮสต์ อาร์คิเทคเจอร์

- สกรีนสับเน็ตอาร์คิเทคเจอร์ (Screened Subnet Architecture)

ต่างจากสถาปัตยกรรมสกรีนนิ่งโฮสต์ ตรงที่ส่วนของเบสชันโฮสต์จะมีการแบ่งเครือข่ายภายในออกจากเครือข่ายที่ให้บริการซึ่งให้ความปลอดภัยมากกว่า ในกรณีนี้จะมีไฟร์วอลถึงสองตัว คือ เราเตอร์ภายนอกและเราเตอร์ภายใน เราเตอร์ภายในจะทำหน้าที่ป้องกันไม่ให้เครือข่ายภายในถูกโจมตี และป้องกันไม่ให้มีการโจมตีจากเครือข่ายภายในได้ โดยจะกรองทั้งแพ็กเก็ตที่มาจากอินเทอร์เน็ตและเบสชันโฮสต์



รูปที่ 2.6 สกรีนสับเน็ตอาร์คิเทคเจอร์

นอกจากนี้ยังประกอบด้วยเครือข่าย DMZ (De-militarize zone) ซึ่งเป็นการวางโฮสต์ที่ไม่น่าเชื่อถือหรือมีความปลอดภัยต่ำ (Untrusted Host) เช่น FTP Server และเว็บไซต์สาธารณะ ไว้ภายในไฟร์วอล ให้แพ็กเก็ตผ่านไปได้ แต่อยู่นอกเครือข่ายภายใน ซึ่งการทำเช่นนี้ไฟร์วอลจะทำการเชื่อมต่อกับสามเครือข่าย และการทำเช่นนี้เป็นการเพิ่มประสิทธิภาพ ความปลอดภัย ความน่าเชื่อถือ แต่ไม่ได้เพิ่มระดับของความปลอดภัยในการเข้าถึงจากโฮสต์ที่อยู่ในเครือข่ายภายใน

2.3 DHCP (Dynamic Host Configuration Protocol)

การกำหนดค่าไอพีแอดเดรสให้กับคอมพิวเตอร์แต่ละเครื่องที่เชื่อมต่อภายในเครือข่าย ทำได้ 2 วิธี คือ กำหนดไอพีแอดเดรสแบบตายตัว (Static Addressing Assignment) ซึ่งเครื่องคอมพิวเตอร์ที่ได้รับการกำหนดค่าดังกล่าวจะใช้เลขหมายไอพีแอดเดรสดังกล่าว ในการอ้างถึงตนเองกับการสื่อสารในทุกรูปแบบตลอดเวลา ในกรณีนี้ผู้ดูแลระบบเครือข่ายอาจประสบปัญหาว่าไอพีแอดเดรสที่ได้รับการอนุญาตให้ใช้งานนั้น มีไม่เพียงพอกับจำนวนเครื่องคอมพิวเตอร์ที่เชื่อมต่อกับเครือข่ายภายใน หรืออาจมีเครื่องคอมพิวเตอร์กลุ่มทำการติดต่อสื่อสารไม่บ่อยนัก สามารถทำการจัดสรรเลขหมายไอพีแอดเดรสแบบเป็นครั้งคราว หรือที่นิยมเรียกกันเป็นทางการว่า จัดสรรแบบพลวัต (Dynamic Addressing Assignment)

DHCP เป็นการกำหนดไอพีแอดเดรสแบบอัตโนมัติให้แก่เครื่องไคลเอ็นต์ในระบบ ที่ติดตั้งโปรโตคอล TCP/IP เป็นการลดความซ้ำซ้อนของหมายเลขไอพีแอดเดรส เนื่องจาก DHCP

เซิร์ฟเวอร์จะเป็นตัวแจกจ่ายไอพีแอดเดรสที่ไม่ซ้ำให้แก่เครื่องไคลเอ็นต์ DHCP เซิร์ฟเวอร์จะขอบเขตในการจ่ายไอพีแอดเดรส โดยผู้ดูแลระบบกำหนดว่าจะเริ่มที่หมายเลขไอพีใด เช่น 192.168.1.10 – 192.168.1.220 หมายถึงมีขอบเขตในการแจกไอพีแอดเดรสอยู่จำนวน 210 เครื่อง เมื่อเครื่องไคลเอ็นต์เริ่มบูตจะทำการขอหมายเลขไอพีแอดเดรส (Subnet Mask, Default Gateway และค่าอื่น ๆ) จากเครื่อง DHCP เซิร์ฟเวอร์ เครื่อง DHCP เซิร์ฟเวอร์จะส่งไอพีแอดเดรสกลับไปยังไคลเอ็นต์

โดยสามารถจะแบ่งเป็นขั้นตอนการทำงานได้ดังนี้

รูปที่ 2.7 การร้องขอไอพีแอดเดรส

- เครื่องไคลเอ็นต์ ทำการค้นหาตำแหน่งที่อยู่ของ DHCP เซิร์ฟเวอร์บนระบบเครือข่ายโดยการส่งแมสเซส DHCP Discover ออกไปบนเครือข่ายเพื่อร้องขอไอพีแอดเดรส
- เครื่อง DHCP เซิร์ฟเวอร์ทำการค้นหาหมายเลขไอพีแอดเดรสจากฐานข้อมูลในเครื่องเพื่อไม่ให้ซ้ำกัน และส่งแมสเซส DHCP Offer กลับไปให้เครื่องไคลเอ็นต์ที่ร้องขอ
- เมื่อเครื่องไคลเอ็นต์ได้รับหมายเลขไอพีแอดเดรสเรียบร้อยแล้ว ไคลเอ็นต์จะส่งสัญญาณตอบกลับ DHCP Request มาให้ทราบ
- เครื่องเซิร์ฟเวอร์ DHCP จะส่งสัญญาณ DHCP Ack กลับไปยังเครื่องไคลเอ็นต์เพื่อให้เริ่มใช้งาน (และเซิร์ฟเวอร์ DHCP จะเก็บหมายเลขไอพีแอดเดรส นั้นเอาไว้ไม่ให้เครื่องอื่นใช้ซ้ำ)

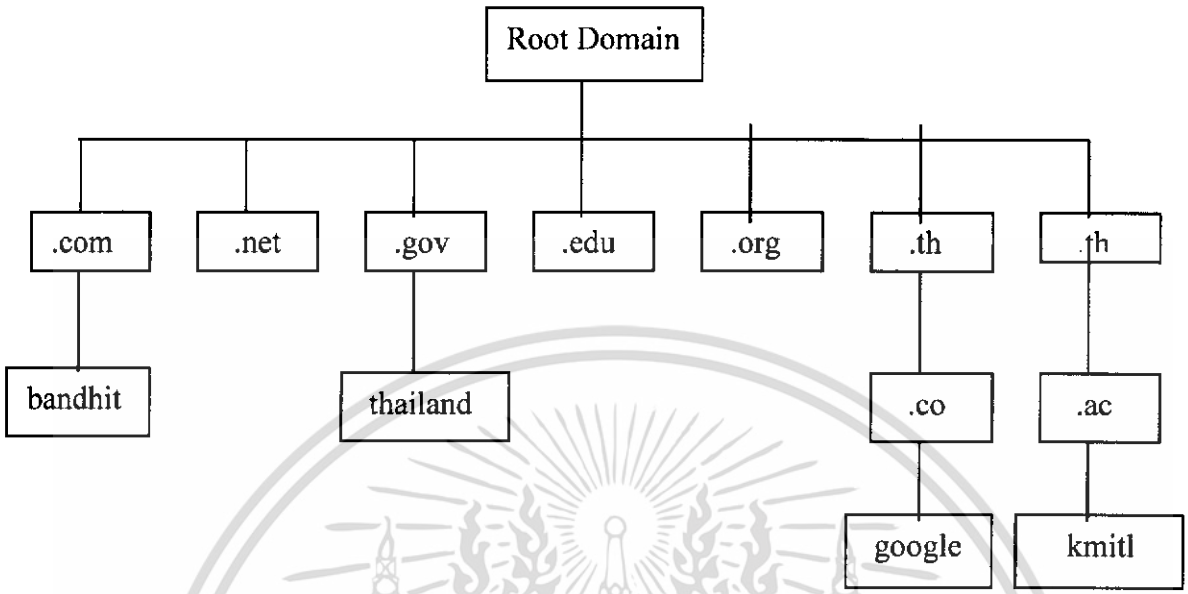
2.4 DNS Server (Domain Name System)

บนเครือข่ายอินเตอร์เน็ตที่มีการเชื่อมโยงผู้คนมากมายเข้าด้วยกัน จะมีหน่วยงาน InterNIC (Internet Network Information Center) เป็นผู้ดูแลเกี่ยวกับการจดทะเบียนเน็ตเวิร์ก หรือโดเมนที่มีการเชื่อมต่อกับระบบอินเตอร์เน็ต โดยจะดูแลรายชื่อและหมายเลขไอพีแอดเดรสทั้งหมด ทุกเครื่องจะต้องเชื่อมต่อผ่านโปรโตคอล TCP/IP ซึ่งต้องใช้ไอพีแอดเดรสในการอ้างถึง เช่น การเข้าสู่เว็บเซิร์ฟเวอร์ ถ้าเว็บเซิร์ฟเวอร์มีไอพีแอดเดรส เป็น 203.247.62.173 จะต้องใช้หมายเลขนี้ในการติดต่อ แต่เนื่องจากหมายเลขไอพีแอดเดรส ของเซิร์ฟเวอร์นั้นมีอยู่หลายหมายเลข เป็นเรื่องยากในการจดจำตัวเลขว่าเป็นของเซิร์ฟเวอร์หรือเว็บไซต์ใด จึงมีการกำหนดมาตรฐานระบบชื่อโดเมน หรือ DNS (Domain Name System) มาใช้

DNS เป็นระบบแปลงหมายเลขไอพีแอดเดรส ให้อยู่ในรูปแบบของโดเมนเนม หรือ แปลงกลับจากโดเมนเนม เป็นไอพีแอดเดรสได้ หลังจากที่ได้ลงทะเบียนชื่อโดเมนผ่าน ISP จะใช้เวลาประมาณ 2 วันในการกระจายชื่อโดเมนนี้ไปทั่วโลก ในการตั้งชื่อโดเมนส่วนมากจะใช้ชื่อหน่วยงาน, บริษัท หรือสถานศึกษา เพื่อให้ง่ายต่อการจดจำ มาตรฐานระบบ DNS ประกอบไปด้วย ชื่อเครือข่าย ชื่อสับโดเมน และชื่อโดเมน เช่น ไอพีแอดเดรส เป็น 203.147.62.173 สามารถเปลี่ยนเป็นตัวหนังสือ หรือชื่อโดเมนได้ดังนี้

รูปที่ 2.8 โดเมน

จะเห็นว่าชื่อสับโดเมนและโดเมนเป็นอักษรย่อ รูปแบบของโดเมนมีอยู่ 2 แบบด้วยกันคือ โดเมน 2 ระดับ และโดเมน 3 ระดับ

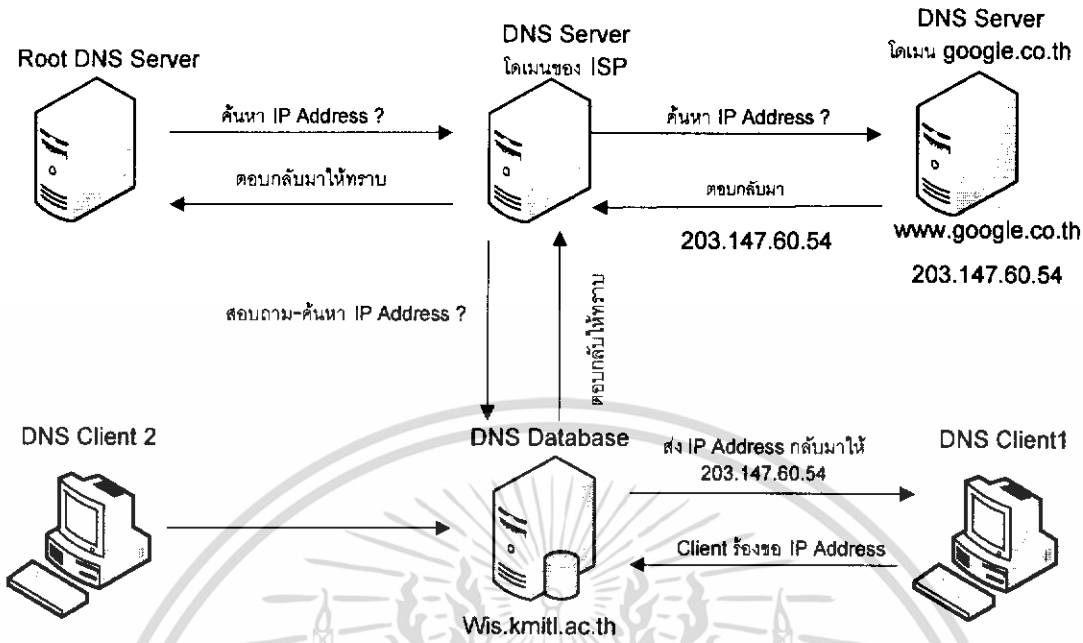


รูปที่2.9 โครงสร้างลำดับชั้นของ DNS

DNS มีโครงสร้างแบบต้นไม้ (tree) เป็นลำดับชั้น (hierarchical) ดังรูป ส่วนบนสุดคือ root domain ส่วนรองลงมาจะเป็น โดเมนเนม กี่ระดับก็แล้วแต่การใช้งาน

หลักการทำงานของ DNS

DNS ถูกออกแบบให้เป็นฐานข้อมูลแบบกระจาย มีกลไกในการทำงานแบบไคลเอ็นต์-เซิร์ฟเวอร์ โดย DNS Server จะเก็บหมายเลขไอพีแอดเดรส คู่กับชื่อเครื่องแบบโดเมนเนม เรียกว่า Name Resolution ไว้ เพื่อค้นหาชื่อตามการร้องขอของเครื่อง DNS ไคลเอ็นต์ ขั้นตอนการทำงานของ DNS มีดังนี้



รูปที่2.10 ขั้นตอนการทำงานของ DNS

- เครื่อง DNS Client ต้องการเชื่อมต่อกับ google.co.th จะส่งการร้องขอหมายเลขไอพีแอดเดรสของเว็บไซต์ www.google.co.th ผ่านกระบวนการ resolver ไปยัง DNS เซิร์ฟเวอร์ บนโดเมนของตน (ในที่นี้คือ Kmitl.ac.th) และถ้าเครื่องDNS เซิร์ฟเวอร์ ไม่มีข้อมูลเกี่ยวกับเว็บไซต์นี้ จะส่งการร้องขอไปยัง DNS เซิร์ฟเวอร์ ของ ISP ผู้ให้บริการอินเตอร์เน็ต
- DNS เซิร์ฟเวอร์ ของ ISP ทำการค้นหาหมายเลขไอพีแอดเดรสนั้น (ถ้าไม่พบจะส่งไปเพื่อค้นหาใน Root DNS Server อีกต่อหนึ่ง) เมื่อพบแล้วจะส่งกลับมายังเครื่อง DNS Server บนโดเมน kmitl.ac.th
- เครื่อง DNS เซิร์ฟเวอร์ของ kmitl.ac.th จะบันทึกหมายเลขไอพีแอดเดรส ของเว็บไซต์ www.google.co.th ไว้ใน Cache Memory (ถ้ามีการร้องขอไอพีแอดเดรสนี้อีกครั้ง ส่งไปได้ทันที โดยไม่ต้องร้องขออีกครั้ง) จากนั้นจะส่งไอพีแอดเดรสของเว็บไซต์นี้กลับไปให้เครื่อง DNS ไคลเอ็นต์เพื่อทำการเชื่อมต่อเข้าสู่เว็บไซต์ตามต้องการ

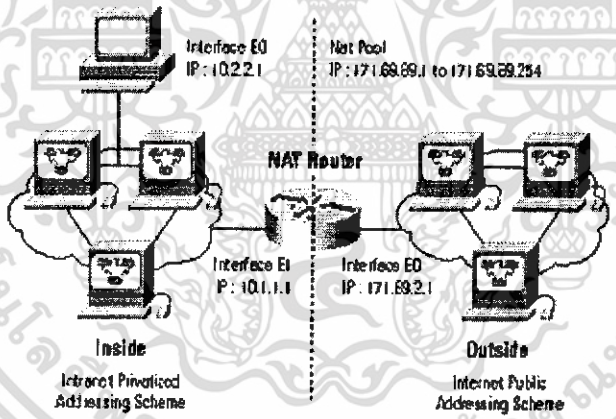
2.5 NAT (Network Address Translation)

Network Address Translation (NAT) เป็นวิธีการหนึ่งในการแปลงและแปลไอพีแอดเดรสของเครือข่ายภายในเป็นไอพีแอดเดรสที่สามารถสื่อสารบนอินเทอร์เนตได้

NAT สามารถใช้ไอพีแอดเดรสที่เป็น private IP (ซึ่งเป็น ไอพีแอดเดรส ที่ไม่ต้องจดทะเบียนบนอินเทอร์เนต) และสามารถซ่อนไอพีแอดเดรสในเครือข่ายภายใน (private IP) ได้ ทำให้เครือข่ายภายในมีความปลอดภัยเพิ่มมากขึ้น

- หลักการทำงานของ NAT

NAT เป็นระบบการอินเตอร์เฟซ (Interface) กับอินเทอร์เนต ที่ไม่ขึ้นอยู่กับโพรโตคอล (Protocol) และ แอพพลิเคชั่นรวมทั้ง อุปกรณ์ฮาร์ดแวร์ ใด ๆ ซึ่งหมายความว่า NAT สามารถนำมาประยุกต์ใช้งานกับ เราท์เตอร์ (Router) หรือคอมพิวเตอร์ที่ทำหน้าที่เป็นเราท์เตอร์ โดยมีอินเตอร์เฟสหนึ่งสำหรับติดต่อกับเครือข่ายภายใน และอีกอินเตอร์เฟสหนึ่งสำหรับการติดต่อกับเครือข่ายภายนอก ตัวอย่างการเชื่อมต่อ เช่น การติดตั้ง NAT ที่ Border เราท์เตอร์ ซึ่งเป็น เราท์เตอร์สำหรับเชื่อมต่อเครือข่ายย่อยภายในองค์กรกับเครือข่ายภายนอก



รูปที่ 2.11 ลักษณะการเชื่อมต่อของ NAT เราท์เตอร์

NAT สามารถเชื่อมต่อสื่อสารทั้งในแบบ Inbound และ Outbound คือ สามารถจัดการกับไอพีแอดเดรสที่ผ่านเข้ามา หรือ ไอพีแอดเดรสที่วิ่งออกไปโดยสามารถจัดการกับไอพีแอดเดรสต้นทางและปลายทางได้เป็นอย่างดี

NAT สามารถทำงานในสถานการณ์ 3 ประการ ดังนี้

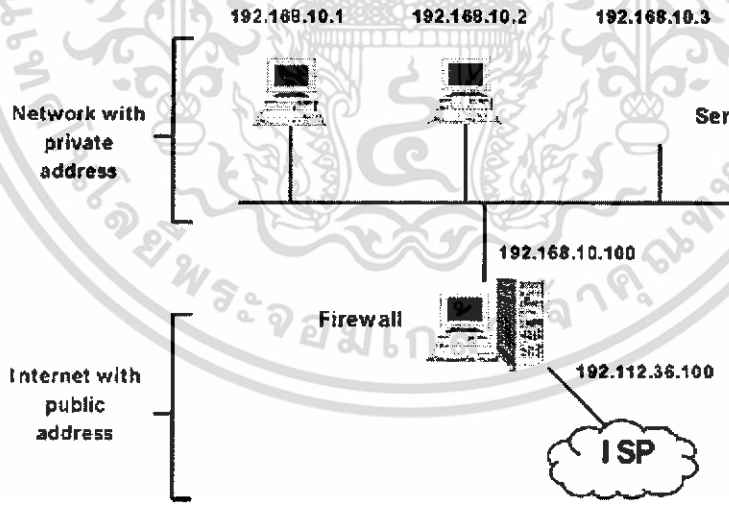
- ทำหน้าที่แปลงและแปลไอพีแอดเดรสต้นทางที่มาจากเครือข่ายภายใน
- ทำหน้าที่แปลงและแปลไอพีแอดเดรสต้นทางที่มาจากเครือข่ายภายนอกเช่น อินเทอร์เนต
- ทำหน้าที่แปลงและแปล ไอพีแอดเดรสปลายทางภายในเครือข่าย

แม้ว่า NAT สามารถใช้ กับ ไอพีแอดเดรสภายนอก แต่โดยทั่วไป NAT มีไว้เพื่อการแปลไอพีแอดเดรสภายในเครือข่าย ซึ่งมีจุดประสงค์เพื่อซ่อนไอพีแอดเดรสภายในเครือข่าย และแปลงไอพีแอดเดรสที่ไม่ต้องจดทะเบียน (private IP) ไปใช้เป็นไอพีแอดเดรสที่จดทะเบียนถูกต้อง ทำให้สามารถวิ่งไปตามเส้นทางบนอินเทอร์เนตได้

ชนิดของ NAT

NAT มีอยู่ 2 ชนิด คือ

- Static NAT เป็นการแปลงไอพีแอดเดรสชนิดกำหนดค่าแอดเดรสตายตัวจากเครือข่ายภายในไปยังเครือข่ายภายนอก ส่วนแอดเดรสภายนอกจะไม่มีการเปลี่ยนแปลง ดังนั้นความสัมพันธ์ระหว่างไอพีแอดเดรสของเครือข่ายภายนอกและภายในจะเป็นแบบแน่นอนตายตัว

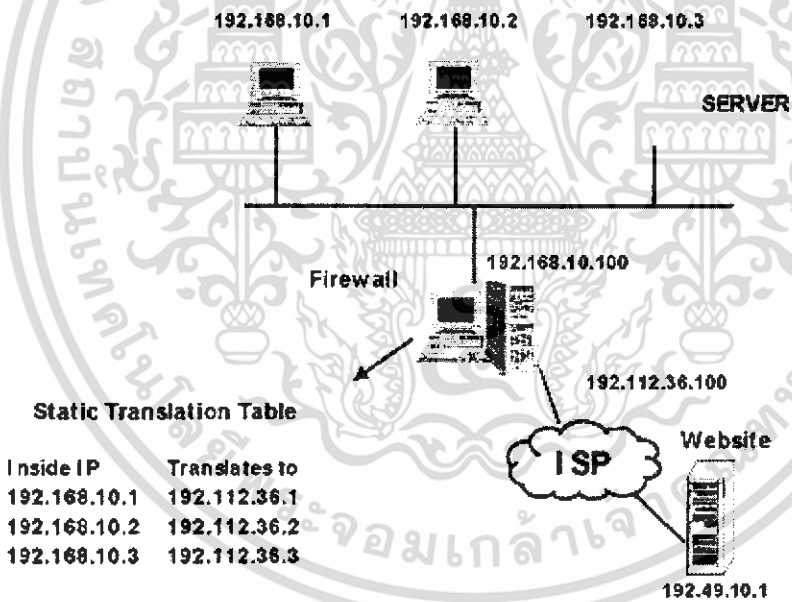


รูปที่ 2.12 ไอพีแอดเดรส ของเครือข่ายภายในกับเครือข่ายภายนอก

จากรูป2.12 เครื่องคอมพิวเตอร์ส่วนบุคคล และเซิร์ฟเวอร์ภายในเครือข่ายกำหนดใช้แอดเดรสส่วนตัว (Private IP) เมื่อต้องการติดต่อกับเครือข่ายภายนอก ต้องผ่านการแปลงไอพีแอดเดรสก่อนที่จะออกจากเครือข่ายเสมอ

สมมติว่า มีคอมพิวเตอร์ส่วนบุคคลเครื่องหนึ่งซึ่งมี ไอพีแอดเดรส ภายในเป็น 192.168.10.1 ทำการส่งข่าวสารไปที่ อินเทอร์เนต โดยอ้าง แอดเดรส ที่ 140.49.10.1 ซึ่งแอดเดรสนี้ เป็น แอดเดรส บนอินเทอร์เนต ลักษณะนี้แพ็คเก็ตที่วิ่งออกจากคอมพิวเตอร์เครื่องนั้น จะมี แอดเดรส ต้นทางเป็น 192.168.10.1 ในกรณีนี้ เมื่อแพ็กเก็ต มาถึง NAT Router จะถูกแปลงเป็น 192.112.36.1 ซึ่งเป็น ไอพีแอดเดรส ที่ผู้จัดการเครือข่ายได้กำหนดขึ้น ลักษณะนี้ ทำให้แอดเดรส ภายในสอดคล้องกับแอดเดรสภายนอกอย่างแน่นอนตายตัว และทุกครั้งที่ติดต่อไปยังภายนอก ต้องใช้ แอดเดรสเดิมเสมอ

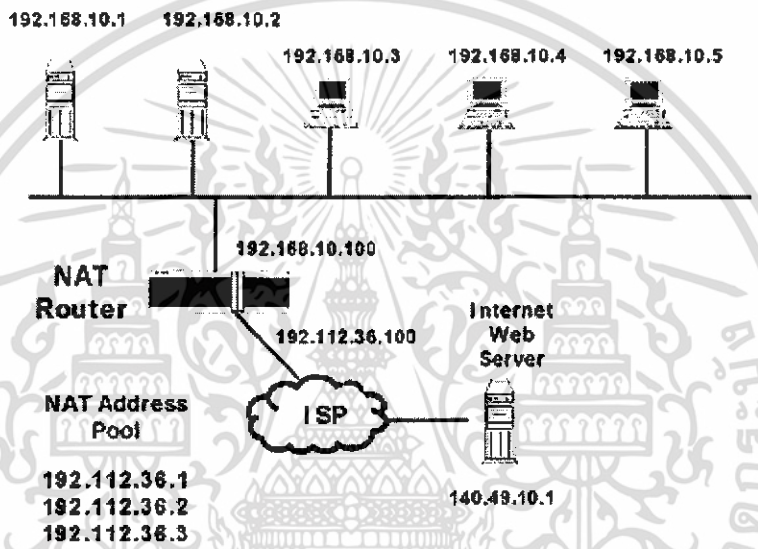
และเมื่อมีการตอบกลับมาจากเว็บไซต์ที่อยู่บนอินเทอร์เนต ตัว NAT แพ็กเก็ตเราท์เตอร์จะใช้กระบวนการย้อนกลับ โดยเราท์เตอร์ทำการพิจารณาค่าแอดเดรสปลายทางบนแพ็กเก็ต ที่ส่งตรงมาจากเว็บไซต์ (140.49.10.1) จากนั้นทำการพิสูจน์เครื่องคอมพิวเตอร์ภายในเครือข่ายที่เว็บไซท์นี้ต้องการติดต่อด้วย และกำหนดไอพีแอดเดรสเพื่อติดต่อกับเครื่องคอมพิวเตอร์ต่อไป ดังรูปที่2.13



รูปที่ 2.13 ลักษณะการอ้าง แอดเดรส และ MAP แอดเดรส แบบ Static

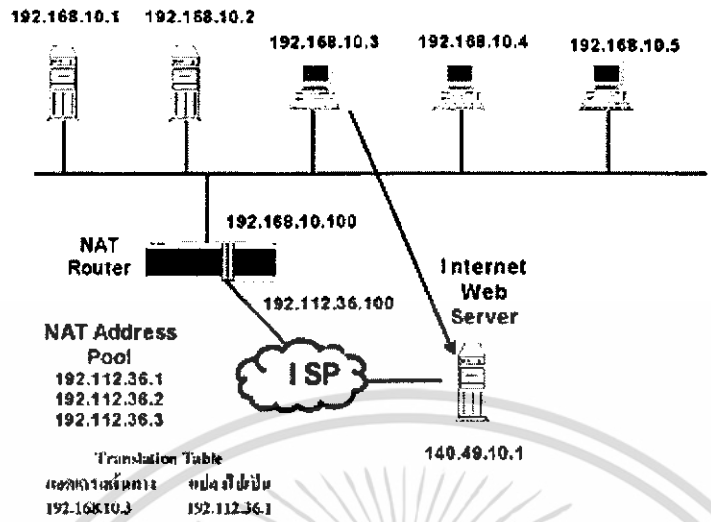
- Dynamic NAT เป็นการนำไอพีแอดเดรสจากกลุ่มของไอพีแอดเดรสที่ร่วมใช้งานกัน หรือที่เรียกว่า แอดเดรส Pool (พลู) มาทำการแปลง จาก แอดเดรส Pool ภายใน ให้เป็น Address Pool สำหรับเครือข่ายภายนอก หรือในทางกลับกัน รูปแบบนี้จะต้องได้รับการจัด Configure โดยผู้ดูแลระบบเครือข่าย แต่หลังจากที่จัดการ Configure เรียบร้อยแล้ว เราท์เตอร์ที่สนับสนุน NAT จะเป็นผู้จ่ายไอพีแอดเดรสให้กับคอมพิวเตอร์อย่างเหมาะสม และเพื่อเพิ่มความรวดเร็ว

ในการทำงาน ผู้บริหารจัดการเครือข่ายจะต้องทำการ Map (แมพ) ระยะของ ไอพีแอดเดรส หากเป็นไปได้ (ลักษณะนี้ คล้ายกับการทำงานของ DHCP เซิร์ฟเวอร์ ที่ไม่ได้กำหนดเครื่อง PC (พีซี) แต่ละเครื่องให้มีไอพีแอดเดรสที่ตายตัว โดยผู้จัดการเครือข่ายจะกำหนดแอดเดรสขึ้นมาจำนวนหนึ่ง เป็นระยะหรือช่วงของแอดเดรส เช่น 192.80.20.15 - 192.80.20.50 โดยเครื่องคอมพิวเตอร์จะไม่ได้รับหมายเลขไอพีซ้ำกัน ข้อแตกต่างกันระหว่าง NAT กับ DHCP Server คือ ไอพีแอดเดรส ของ NAT ที่แจกให้กับเครื่องคอมพิวเตอร์ที่เข้า-ออกบนเครือข่าย ดังรูป 2.14



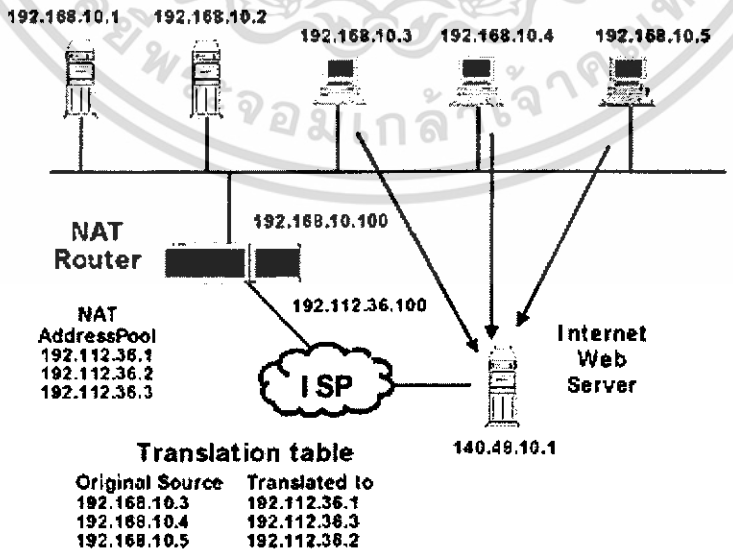
รูปที่ 2.14 ตัวอย่างการทำงานของ Dynamic NAT

Dynamic NAT มีการกำหนดแอดเดรสที่ใช้ติดต่อกับเครือข่ายภายนอกแบบพลวัตร คือ ระบบที่กำหนดแอดเดรส จะมีการเปลี่ยนแปลงทุกครั้งที่คอมพิวเตอร์ภายในเครือข่ายนั้นมีการสถาปนาการเชื่อมต่อกับคอมพิวเตอร์ภายนอกเครือข่าย หรืออาจมีการเปลี่ยนเป็นระยะเวลา ดังรูปที่ 2.14



รูปที่ 2.15 การ Map แอดเดรส ของ PC เครื่องต่อไป ด้วย Dynamic NAT

สมมติว่า เครื่องคอมพิวเตอร์ส่วนบุคคลพร้อมด้วยไอพีแอดเดรสภายในเป็น 192.168.10.3 ใช้ HTTP Traffic (เอชทีทีพี ทราฟฟิก) ไปที่ 140.49.10.1 เมื่อ NAT เราท์เตอร์ได้รับไอพีดาต้าแกรมแต่ละแพ็คเก็ต จะทำการดึงแอดเดรสภายนอก (ที่ผ่านการจดทะเบียนแล้ว) จากแอดเดรส Pool (ในที่นี้ คือ 192.112.36.1, 192.112.36.2 และ 192.112.36.3) และทำการเปลี่ยนแอดเดรสต้นทางด้วยแอดเดรสภายนอก ในขณะเดียวกัน NAT เราท์เตอร์จะสร้างตารางที่ประกอบด้วยแอดเดรส ที่ใช้เพื่อแปลง เช่นเดียวกับที่ Static NAT ใช้เพื่อแปลงแอดเดรสแบบตายตัวหน้าตาของตาราง แอดเดรส เป็นไปตามตัวอย่าง รูปที่ 2.16



รูปที่ 2.16 การใช้ NAT Router ของเครื่องพีซีต่างๆ จากเครือข่ายภายใน

ในเวลาเดียวกันไอพีดาต้าแกรมที่มาจากคอมพิวเตอร์เครื่องที่สองบนเครือข่ายภายในที่ส่งข่าวสารไปยังเครือข่ายภายนอก จะได้รับการปฏิบัติในทำนองเดียวกันโดย N A T เราท์เตอร์

ตัวอย่าง เช่น NAT เราท์เตอร์ อาจแปลง ไอพีแอดเดรสต้นทาง ที่อยู่ใน IP Header ที่ติดต่อออกไปเพื่อต้องการใช้งาน FTP แอพพลิเคชั่น (FTP Application) จาก PC ที่ใช้ แอดเดรส 192.168.10.4 ไปที่ แอดเดรส ต่อไปใน NAT Address Pool ซึ่งประกอบด้วยแอดเดรสจำนวนหนึ่ง (เช่น 192.112.36.1 - 192.112.36.3) และ NAT เราท์เตอร์ จะแปลงแอดเดรส 192.168.10.4 เป็น 192.112.36.3 ส่วนคอมพิวเตอร์อีกเครื่องหนึ่งคือ 192.168.10.5 จะถูกแปลงเป็น 192.112.36.2 หากเครื่องพีซีทั้งสองต้องการติดต่อกับเครือข่ายภายนอก ดังรูปที่ 2-14

เมื่อคอมพิวเตอร์แต่ละเครื่องสิ้นสุดการสื่อสารข้อมูลกับเครือข่ายภายนอกตัว NAT เราท์เตอร์ จะเรียกแอดเดรสภายนอกคืนกลับไปยัง Address Pool เพื่อให้ผู้ใช้รายต่อไป

2.6 IPTABLES

รูปแบบการใช้งาน IPtables เบื้องต้น จะมีรูปแบบการใช้งานดังนี้คือ

iptables [table] <command> <match> <target/jump>

โดยกฎที่กำหนดขึ้นจะเป็นเป็นตัวบอกเคอร์เนลว่าให้กระทำอย่างไร ในกรณีที่พบแพ็กเก็ตตรงตามที่ระบุไว้

- [table] หมายถึง ตารางที่ต้องการระบุ เช่น iptables -t nat หมายถึงให้ทำงานกับ nat table ในกรณีที่ไม่ได้ระบุตาราง iptables จะถือว่าคำสั่งดังกล่าวระบุถึง filter table โดยอัตโนมัติ
- <command> จะเป็นตัวสั่งให้ iptables ทำในสิ่งที่ต้องการ เช่น iptables -A INPUT ซึ่งหมายถึงให้สร้าง rule ต่อท้าย INPUT chain ใน filter table
- <match> เป็นส่วนที่ใช้ตรวจสอบว่าแพ็คเก็ตมีข้อมูลตรง (match) กับที่ระบุไว้หรือไม่ เช่น มีไอพีแอดเดรสต้นทางเป็น 1.2.3.4
- <target/jump> เป็นตัวระบุว่าเมื่อเจอแพ็คเก็ตที่ match ก็จะกระทำ (action) ตามที่ระบุไว้ เช่น ถ้าแพ็คเก็ตใดมีไอพีแอดเดรสต้นทางเป็น 1.2.3.4 ให้ DROP แพ็กเก็ตนั้นทิ้งไป

iptables สามารถทำงานได้กับตาราง (table) 3 ตารางหลัก สามารถระบุตารางได้โดยใช้ออปชัน -t ตามด้วยชื่อ table คือ

1. Filter table ใช้สำหรับกรอง packet มี 3 built-in chain คือ INPUT, OUTPUT, FORWARD
2. NAT table ใช้สำหรับการแปลงแอดเดรส (Network Address Translation) มี 3 built-in chain คือ PREROUTING, POSTROUTING, OUTPUT
3. Mangle table เป็นตารางที่ใช้เปลี่ยนแปลงหรือแก้ไข packet เช่น เปลี่ยนค่า TTL , MARK ซึ่งปกติจะใช้ในการทำ routing ที่มีความซับซ้อนสูง มี 2 built-in chain คือ PREROUTING chain (ใช้แก้ไข packet ก่อนที่จะเข้าสู่ไฟร์วอลล์และก่อนเข้าสู่ routing decision) และ OUTPUT chain (ใช้แก้ไข packet ที่ถูกสร้างโดยไฟร์วอลล์ก่อนที่มันจะถูกส่งไปยัง routing decision) แต่ไม่สามารถทำ network address translation หรือ masquerading ที่ table นี้ได้

การใช้คำสั่ง iptables

- -A เพิ่มกฎใหม่ต่อท้าย chain (Append rule) เช่น

  # iptables -A INPUT -p ALL -i eth0 -j ACCEPT

- -D ลบกฎ (Delete rule) เช่น

  # iptables -D INPUT --dport 80 -j DROP

- -I เพิ่มกฎใหม่ ใน chain (Insert rule) เช่น

  # iptables -I OUTPUT -p ALL -s 127.0.0.1/32 -j ACCEPT

- -R แทนที่กฎเดิม ด้วย rule ใหม่ (Replace rule)

- -L แสดงกฎทั้งหมดใน chain (ถ้าไม่ระบุ chain จะแสดงกฎทั้งหมดใน filter table ทั้งสาม built-in chain) เช่น

  # iptables -L

  # iptables -L -t nat

  # iptables -L INPUT

- -F ลบกฎทั้งหมดใน chain ทิ้ง เช่น

  # iptables -F INPUT

  # iptables -F mychain

- -Z ใช้ reset byte counter สำหรับทุกกฎใน chain ที่กำหนด เช่น

  # iptables -Z INPUT

- -N ใช้สร้าง chain ใหม่ เช่น

  # iptables -N mychain

- -X ลบ chain ที่ไม่มีกฎซึ่งสามารถลบ user-defined chain ที่ไม่มีกฎได้ แต่ไม่สามารถลบ built-in chain ได้ เช่น

  # iptables -X emptychain

- -P เปลี่ยน default policy ของ chain ค่าที่ใช้ได้คือ ACCEPT, DROP ทั้งนี้ค่านี้มีความสำคัญอย่างมากเพราะหากแพ็คเก็ตถูกส่งเข้ามาในchainแล้วและไม่เข้ากับกฎใด ๆ เลย แพ็กเก็ตนั้นต้องถูกตัดสินใจโดย default policy ของ chain นั้นๆ เช่น

  # iptables -P FORWARD DROP

  ซึ่งหากแพ็กเก็ตถูกส่งเข้ามายัง FORWARD chain และไม่ตรงกับกฎใด ๆ ใน FORWARD chain จะถูก DROP ทันที

- -E ใช้เปลี่ยนชื่อ chain ใหม่ เช่น

  # iptables -E myoldchain mynewchain

การตั้งค่า Match

การตั้งเงื่อนไขของการ match นั้นจะต้องอาศัยความเข้าใจในเรื่อง IP, TCP, UDP และ ICMP มาบ้างพอสมควร จึงจะสามารถตั้งเงื่อนไขที่เหมาะสมและตรงตามความต้องการได้ ซึ่งมีรายละเอียดดังนี้

- การระบุไอพีแอดเดรสต้นทางหรือปลายทาง สามารถระบุไอพีแอดเดรสต้นทางของแพ็กเก็ต โดยใช้ -s หรือ --source หรือ --src และใช้ -d หรือ --destination หรือ --dst สำหรับ destination ไอพีแอดเดรส การระบุไอพีแอดเดรสนั้นสามารถทำได้ 4 แบบ คือ
  1. ใช้ชื่อเต็มแทน เช่น localhost หรือ www.kmitl.ac.th
  2. ระบุไอพีแอดเดรสโดยตรง เช่น 127.0.0.1 หรือ 202.44.204.33
  3. ระบุเป็น group ของไอพีแอดเดรส เช่น 202.44.204.0/24 ซึ่งหมายถึงไอพีแอดเดรสตั้งแต่ 202.44.204.0 - 202.44.204.255
  4. หรืออาจจะใช้ 202.44.204.0/255.255.255.0 แทน 202.44.204.0/24 ได้

- การทำInversion

  ในบางกรณีนั้นหากต้องการระบุเป็น inverse เช่น อนุญาตให้ทุกไอพียกเว้นไอพีที่ระบุไว้ ซึ่งการใช้คำสั่งดังกล่าวสามารถทำได้ โดยใช้เครื่องหมาย ! นำหน้า argument ที่ต้องการ (เครื่องหมาย ! หมายถึง NOT) เช่น -p ! TCP ซึ่งจะ match กับโพรโตคอลทุกตัวที่ไม่ใช่ TCP หรือ -s ! localhost ซึ่งหมายถึงแพ็กเก็ตที่มีไอพีแอดเดรสต้นทางยกเว้น localhost (127.0.0.1)

- การระบุโพรโตคอล

  สามารถระบุโพรโตคอลที่ต้องการได้ดังนี้ คือ TCP, UDP, ICMP หรือสามารถใช้ตัวเลขแทนได้ (สำหรับ *NIX อ้างอิงได้จาก /etc/protocols) สามารถใช้ได้ทั้งตัวอักษรเล็กหรือใหญ่ (ใช้ได้ทั้ง tcp และ TCP) เช่น -p TCP หรือ -p ! tcp

- การระบุinterface

  -i หรือ --in-interface ตามด้วยชื่อ interface ใช้เพื่อระบุ incoming interface แพ็คเก็ตที่จะ match กับกฎนี้ต้องเข้ามาจาก interface ที่กำหนด เช่น -i Eth0 หมายความว่า ทุกแพ็คเก็ตที่เข้ามาทาง Eth0 จะ match กับกฎนี้ ทั้งนี้ชื่อ interface ที่สามารถใช้ได้นั้น สามารถตรวจสอบได้โดยใช้คำสั่ง ifconfig และ -o หรือ --out-interface ตามด้วยชื่อของ interface ใช้เพื่อระบุ outgoing interface ซึ่งหมายถึแพ็คเก็ตที่จะ match กับกฎนี้กำลังจะเดินทางผ่าน interface ที่ระบุไว้ เช่น -o Eth1 หรือ -o ! Eth1

Match Extension

เป็น netfilter packet ที่อยู่ในช่วงทดลองใช้ รูปแบบการใช้งานให้ใช้ –m แล้วตามด้วย match ที่ต้องการ เช่น -m mac ทั้งนี้มีออปชันให้เลือกใช้งานดังต่อไปนี้

- mac

  รูปแบบการใช้งาน: -m mac หรือ --match mac ใช้ตรวจสอบ source MAC address ว่าตรงกับค่าที่ระบุไว้หรือไม่ มีประโยชน์สำหรับ PREROUTING, INPUT chain โดยมีออปชันให้ใช้งานคือ

  --mac-source เช่น --mac-source 00:55:81:CC:42:FF

- limit

  รูปแบบการใช้งาน: -m limit หรือ --match limit

  ใช้เพื่อจำกัดจำนวนของการ match ที่อาจจะมากเกินไปเป็นประโยชน์สำหรับกฎที่วางไว้

ตอนท้ายสุดของ chain (ใช้ร่วมกันกับ DROP policy) ซึ่งส่วนใหญ่เป็น rule ที่ใช้เก็บข้อมูลลงล็อกไฟล์ ซึ่งถ้าผู้บุกรุกส่ง packet ที่ไม่เข้าข่ายกฎใดๆ ใน chain จนกระทั่งมาถึงกฎที่ทำหน้าที่เก็บล็อกนี้ ถ้าแพ็คเก็ตที่เข้ามามีจำนวนมากก็อาจจะทำให้ฮาร์ดดิสก์เต็มได้ ดังนั้นจึงต้องใช้จำกัดจำนวนในการเก็บข้อมูลลงล็อก ซึ่งมีออปชันให้ใช้งานดังนี้คือ

--limit ตามด้วยตัวเลข ซึ่งบอกถึงจำนวนครั้งสูงสุดของการ match กับกฎที่ยินยอมต่อ 1 วินาที เช่น --limit 5/s ทั้งนี้ยังสามารถใช้หน่วยเวลาอื่นได้ เช่น /second /minute /hour /day เช่น -m limit --limit 3/minute

The State Match

รูปแบบการใช้งาน: -m state หรือ --match state เป็นโมดูลที่ใช้ประโยชน์ได้เป็นอย่างดี มีออปชันให้ใช้งานดังนี้

- NEW

  รูปแบบการใช้งาน: -m state --state new หรือ --match state --state new

  หมายถึงแพ็กเก็ตที่เป็นตัวสร้างการเชื่อมต่อใหม่

- ESTABLISHED

  รูปแบบการใช้งาน: -m state --state established หรือ --match state --state established

  หมายถึง แพ็กเก็ตที่เกี่ยวข้องกันกับการเชื่อมต่อที่สร้างไว้แล้ว เช่น echo-reply packet หรือแพ็คเก็ตที่ส่งข้อมูลออกไปจากเว็บเซิร์ฟเวอร์เมื่อมี request web service เข้ามา

- RELATED

  รูปแบบการใช้งาน: -m state --state related หรือ --match state --state related เป็นแพ็กเก็ตที่เกี่ยวข้องกับการเชื่อมต่อที่สร้างไว้แล้ว แต่ไม่ใช่ส่วนหนึ่งส่วนใดของการเชื่อมต่อนั้น เช่น FTP data packet (port 20) ที่เกิดขึ้นจากการใช้คำสั่งใน FTP command (port 21)

- INVALID

  รูปแบบการใช้งาน: -m state --state invalid หรือ --match state --state invalid เป็นแพ็กเก็ตที่ไม่เกี่ยวข้องกับส่วนอื่น เช่น ICMP echo-reply ที่เกิดขึ้น โดยที่ไม่มีเครื่องได้การส่ง echo-request (กรณีเช่นนี้เกิดขึ้นได้เนื่องจากอาจจะโดนโจมตีแบบ Smurf attack)

$IPTABLES -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT

เป็นคำสั่งที่ทำให้ Iptables สามารถทำงานได้ในรูปแบบของ Stateful Inspection ที่แท้จริง โดยกฎนี้จะตรวจสอบแพ็กเก็ต ว่ามีเป็นส่วนหนึ่งของการเชื่อมต่อ ที่สร้างไว้แล้วหรือไม่ (ESTABLISHED) ถ้าใช่จะปล่อยให้ผ่านไป (ACCEPT) และในกรณีที่เครื่องภายในเครือข่ายเรียกใช้ FTP ไปยังเครื่องอื่นในอินเทอร์เน็ตนั้น คำสั่งที่ส่งไปจะใช้ พอร์ทปลายทางเป็น 21 แต่ data พอร์ทที่ใช้สำหรับรับส่งข้อมูลใน FTP นั้นเป็นพอร์ท 20 ซึ่งพอร์ทที่เกิดขึ้นนี้ถือว่ามีความสัมพันธ์(related) กับพอร์ท 21 ดังนั้นจึงสามารถรับส่งไฟล์ผ่านพอร์ท 20 ได้โดยไม่จำเป็นต้องสร้างกฎ เพิ่มเติม

- NAT Table

เป็นตารางที่ใช้สำหรับทำ Network Address Translation เช่น เปลี่ยนค่าไอพีแอดเดรสต้นทาง , ไอพีแอดเดรสปลายทาง จุดสำคัญ คือ มีเพียงแพ็กเก็ตแรกเท่านั้นที่เข้ามาที่ chain นี้ ส่วนแพ็คเก็ตถัดไปจะถูกกระทำเหมือนแพ็กเก็ตแรกได้รับ ดังนั้นจึงไม่ควรทำแพ็กเก็ต filtering ที่ chain เหล่านี้

การใช้งานตาราง Nat นั้น ใช้ออปชัน -t nat และ target ที่สามารถใช้งานได้ คือ SNAT, DNAT, Masquerade, Redirect ซึ่งมีรายละเอียดดังนี้

- SNAT

การทำ NAT ต้นทาง จะทำที่ POSTROUTING chain โดย คือทำการเปลี่ยนแอดเดรสต้นทาง ก่อนที่จะส่งแพ็กเก็ตนั้นออกไป ซึ่งสามารถใช้ออปชัน -o (outgoing interface) ร่วมด้วยได้ นอกจากนี้ยังใช้ -j SNAT และ --to--source หรือ --to เพื่อเปลี่ยนไอพีแอดเดรสหรือพอร์ทไปตามต้องการได้ เช่น

```
##เปลี่ยนไอพีแอดเดรสต้นทางเป็น 1.2.3.4
# iptables -t nat -A POSTROUTING -o eth0 -j SNAT --to 1.2.3.4

##เปลี่ยนไอพีแอดเดรสต้นทางเป็น 1.2.3.4, 1.2.3.5 หรือ 1.2.3.6
# iptables -t nat -A POSTROUING -o eth0 -j SNAT --to 1.2.3.4-1.2.3.6

##เปลี่ยนไอพีแอดเดรสต้นทางเป็น 1.2.3.4 port 1-1023
# iptables -t nat -A POSTROUTING -p tcp -o eth0 -j SNAT --to 1.2.3.4:1-1023
```

- DNAT

  การทำ NAT ปลายทาง จะทำภายใต้ PREROUTING chain คือการเปลี่ยนค่า แอดเดรส ปลายทาง หรือพอร์ท ก่อนที่จะส่งแพ็คเก็ตไปยัง routing decision

  โดยปกติการใช้งานจะระบุ -j DNAT และใช้ --to-destination หรือ --to และยังสามารถใช้ -i (incoming interface) ร่วมด้วย เช่น

  ##เปลี่ยนแอดเดรสปลายทางเป็น 192.168.1.20

  # iptables -t nat -A PREROUTING -i eth0 -j DNAT --to 192.168.1.20

  ##เปลี่ยนแอดเดรสปลายทางเป็น 192.168.1.20, 192.168.1.21 หรือ 192.168.1.22

  # iptables -t nat -A PREROUTING -i eth0 -j DNAT --to 192.168.1.20-192.168.1.22

  ##เปลี่ยนแอดเดรสปลายทางของ web traffic เป็น 192.168.1.50 port 8080

  # iptables -t nat -A PREROUING -p tcp --dport 80 -ieth0 -j DNAT --to 192.168.1.50:80

2.7 Webserver (เว็บเซิร์ฟเวอร์)

เว็บเซิร์ฟเวอร์ คือ แอพพลิเคชั่นที่ทำหน้าที่รับ และประมวลผลข้อมูลที่ร้องขอจากผู้ใช้บริการอินเทอร์เนตโดยผ่านทางเว็บบราวเซอร์ (Webbrowser) หลังจากเว็บบราวเซอร์รับคำร้องและประมวลผลแล้ว ผลลัพธ์จะถูกส่งกลับไปยังผู้ใช้ โดยแสดงผลในเว็บบราวเซอร์นั่นเอง นอกจากเว็บบราวเซอร์จะให้บริการในอินเทอร์เน็ตแล้ว ยังสามารถนำมาปนะยุกต์ใช้ในเครือข่ายภายในองค์กรหรืออินทราเน็ตได้อีกด้วย

เว็บเซิร์ฟเวอร์นั้นเป็นกลไกสำคัญต่อองค์กรต่าง ๆ และช่วยสนับสนุนแอปพลิเคชั่นใหม่ ๆ ในลักษณะของ Web based Application ซึ่ง linux ได้รวมเอา Apache Web Server อันเป็นเว็บเซิร์ฟเวอร์ที่มีประสิทธิภาพสูง และนิยมใช้มากที่สุดไว้แล้ว และนอกจากตัว Apache เองแล้ว linux ยังมีส่วนประกอบอื่น ๆ ที่เกี่ยวข้องกับการพัฒนาบริการต่าง ๆ บนเว็บให้มาพร้อมกันด้วย เช่น PHP เป็นต้น

Apache

ในความเป็นจริงแล้ว สถานะของApacheในปัจจุบันถูกแบ่งออกในเชิงการประยุกต์ใช้งานได้ 2 ทาง คือ การใช้งานทางตรง หรือการใช้งานโดยเน้นหนักไปในฐานะของ HTTP Server ซึ่งถูกนำไปใช้งานเป็นเว็บเซิร์ฟเวอร์โดยตรง ในส่วนนี้ยังสามารถแยกลักษณะการใช้งานออกไปได้อีกหลายทิศทางขึ้นอยู่กับลักษณะของงานและคุณลักษณะพิเศษต่าง ๆ ที่เสริมเข้าไปอีกด้วย ได้แก่

- การใช้งานเป็น Mirror Site ด้วยความสามารถจากโมดูลในกลุ่ม mod_proxy.c ทำให้เราสามารถประยุกต์ใช้Apacheเป็นเว็บไซต์ Mirror ได้ โดยสามารถสำเนาเนื้อหาจากเว็บไซต์ที่ได้รับการอนุญาติแล้วมาให้บริการในเซิร์ฟเวอร์ของเราได้
- ทำหน้าที่เป็น Web Redirector หรือทำหน้าที่เป็นตัวช่วยเปลี่ยนทิศทางของผู้ชมที่มาจากแหล่งต้นทางที่แตกต่างกันให้ไปสู่ URL หรือเซิร์ฟเวอร์ที่กำหนดขึ้นใหม่ได้ ซึ่งมาจากความสามารถของโมดูล mod_rewrite.c
- การสร้างเว็บไซต์ส่วนบุคคล หรือ Personal Home Page การใช้งานแบบนี้เป็นที่นิยมมากในสถานศึกษา มหาวิทยาลัย โดยอาศัยการทำงานของโมดูล mod_userdir.c จะช่วยให้ยูสเซอร์ทุกคนในเว็บเซิร์ฟเวอร์มีเว็บไซต์ส่วนตัวได้โดยอัตโนมัติ โดยมี URL เป็นชื่อเว็บไซต์นั้นตามด้วยเครื่องหมาย ~ และชื่อของยูสเซอร์นั้น ๆ เช่น ยูสเซอร์ gump ในเซิร์ฟเวอร์ www.tepleela.ac.th จะมี URL เป็น http://www.tepleela.ac.th/~gump/ เป็นต้น ซึ่งทำให้สมาชิก นักเรียน นักศึกษา มีเว็บไซต์เป็นของตนเองที่จะใช้ฝึกหัดสร้างเว็บไซต์ และเผยแพร่ข้อมูลสู่สาธารณะได้ตามต้องการ
- การเป็น Virtual Host ลักษณะนี้เป็นที่นิยมกันมากทีเดียว คือ การสร้างเว็บไซต์มากกว่า 1 เว็บไซต์โดยใช้เครื่องเซิร์ฟเวอร์เพียงเครื่องเดียว และใช้หมายเลขไอพีแอดเดรสเพียงหมายเลขเดียวในการอ้างถึงเว็บไซต์หลายชื่อ หรือที่เรียกว่า Name Based Virtual Host ซึ่งช่วยให้ลดค่าใช้จ่ายไปได้มาก
- การเป็นเว็บเซิรฟเวอร์ที่สนับสนุนเทคโนโลยีเว็บอื่น ๆ Apache 1.3 และ 2.0 เป็นเพียงหนึ่งในโปรเจคของ The Apache Software Foundation เท่านั้น ยังมีโปรเจคอื่น ๆ ที่เป็นโปรเจคต่อเนื่องจากอาปาเช่อีกมากมาย เช่น Jakarta เป็นโปรเจคเสริมเพื่อทำให้อาปาเช่สนับสนุน Java Platform โดยหนึ่งในจำนวนโปรแกรมที่เป็นที่รู้จักกันเป็นอย่างดีก็คือ Tomcat 5 ซึ่งเสริมการสนับสนุน Java Servlet 2.4 และ Java Server Pages 2.0

ในทางอ้อม การประยุกต์ใช้อาปาเช่เว็บเซิร์ฟเวอร์ยังถูกนำมาใช้เพื่อเป็นส่วนประกอบในงานด้านอื่น ๆ อีก โดยอยู่ในฐานะช่องทางติดต่อระหว่างผู้ใช้กับแอปพลิเคชั่นต่าง ๆ ในลักษณะของ Web based User Interface ซึ่งผู้ใช้อินเตอร์เน็ตทั่วไปมีความคุ้นเคยดีอยู่แล้ว อีกทั้งยังลดการบำรุงรักษาและคอนฟิกในฝั่งเครื่องไคลเอ้นไปได้มากอีกด้วย การใช้งานในทางอ้อมที่ว่านี้ ได้แก่

- เป็นยูสเซอร์อินเทอร์เฟสเข้าสู่ยูทิลิตี้ Apacheถูกนำไปพัฒนาร่วมกับซอฟต์แวร์(Soft ware) ต่าง ๆ มากมายทั้งซอฟต์แวร์เชิงพานิชย์ และฟรีซอฟต์แวร์ เพื่อใช้เป็นอินเทอร์เฟสที่สะดวกต่อการใช้งานยิ่งขึ้น เช่น ซอฟต์แวร์บริหารจัดการโปรแกรมตรวจสอบและกำจัดไวรัส ซอฟต์แวร์ช่วยการคอนฟิกและใช้งานลีนุกซ์เซิร์ฟเวอร์

- เป็นช่องทางแสดงผลข้อมูลระบบและเครือข่าย เนื่องจากApacheถูกผนวกเอาไว้กับลีนุกซ์เซิร์ฟเวอร์ทุกดิสทริบิวชั่น หรือ ถ้าเป็นโอเอสอื่น (Windows, Mac OS X) ก็สามารถติดตั้งใช้งานได้ฟรี และสามารถแสดงผลได้ทั้งตัวอักษร รูปภาพ รูปกราฟ ได้โดยตรง จึงมีการนำอาปาเช่มาใช้งานด้านการแสดงผลข้อมูลระบบ และกราฟสถิติต่าง ๆ มากมาย เช่น MRTG ใช้แสดงข้อมูลกราฟที่ได้ข้อมูลจาก Router หรือ SNMP Server โปรแกรม SARG ใช้แสดงตารางสถิติการเข้าชมเว็บไซต์ของผู้ใช้งาน Squid Proxy Server โปรแกรมประเภท Log Analyzer เป็นต้น

- ใช้เป็น Web Mail ข้อดีของการใช้งานอีเมล์ผ่านทางเว็บบราวเซอร์เป็นสิ่งที่เราต่างทราบกันเป็นอย่างดี Apacheในฐานะที่เป็น Front-End ของระบบอีเมล์จึงเป็นงานอีกลักษณะหนึ่งที่เรานิยมนำมาใช้งานร่วมกับระบบ Mail Server

- เป็นอินเทอร์เฟสของแอปพลิเคชั่นเฉพาะทาง มีซอฟต์แวร์เป็นจำนวนมากที่พัฒนาโดยทำงานภายใต้สภาพแวดล้อมที่เรียกว่า Web based Applications ทั้งที่เป็นการพัฒนาขึ้นเพื่อใช้งานในองค์กรโดยเฉพาะ และทั้งที่เป็นซอฟต์แวร์สำเร็จรูป เช่น โปรแกรมประเภท Groupware หรือ Web based collaboration ต่าง ๆ ระบบสนับสนุนสารสนเทศภายในองค์กร เป็นต้น

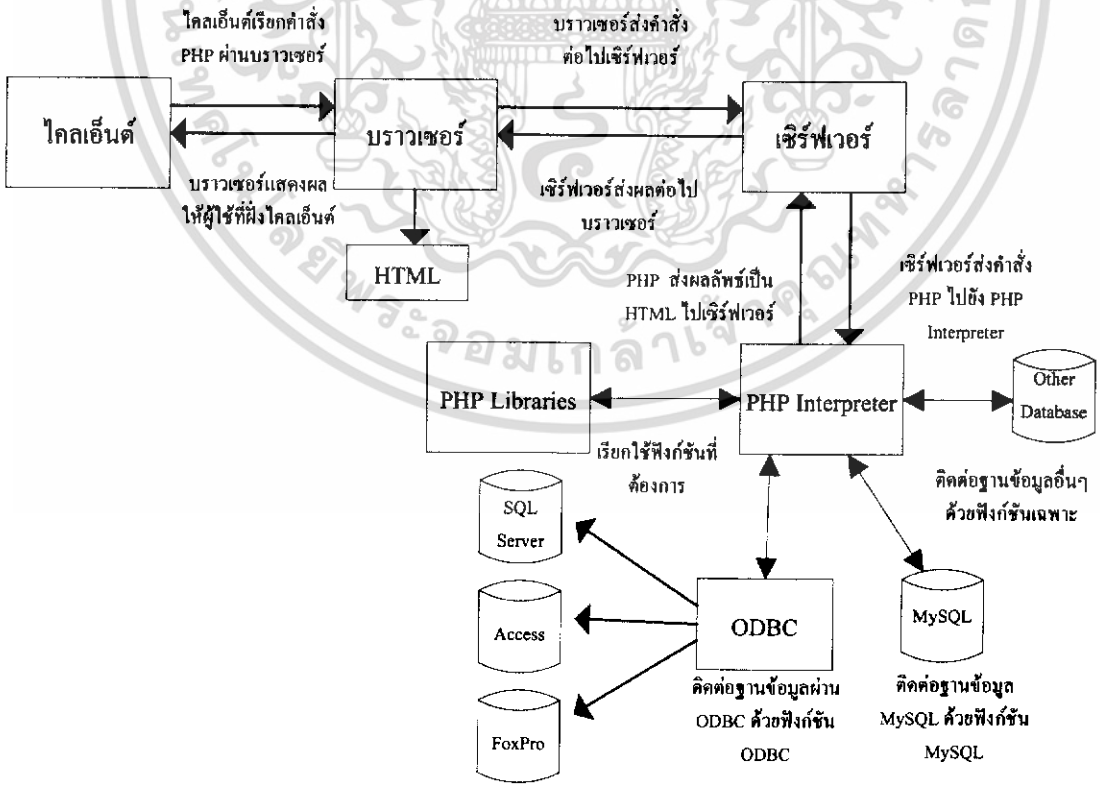
PHP

PHP หมายถึง PHP Hypertext Preprocessor ซึ่งเป็นภาษาจำพวก scripting language โดยคำสั่งต่างๆจะเก็บอยู่ในไฟล์ที่เรียกว่า สคริปต์ (script) และเวลาใช้งานต้องอาศัยตัวแปลชุดคำสั่ง ตัวอย่างของภาษาสคริปก็เช่น JavaScript, Perl เป็นต้น ลักษณะของ PHP ที่แตกต่างจากภาษาสคริปต์แบบอื่นๆ คือ PHP ได้รับการพัฒนาและออกแบบมา เพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ ดังนั้นจึงกล่าวว่า PHP เป็นภาษาที่เรียกว่า

server-side หรือ HTML-embedded scripting language ที่ประมวลผลฝั่งเซิร์ฟเวอร์แล้วส่งผลลัพธ์ไปฝั่งไคลเอ็นต์ผ่านเว็บบราวเซอร์ซึ่งเป็นเครื่องมือที่สำคัญชนิดหนึ่งที่ช่วยให้เราสามารถสร้างเอกสารแบบ Dynamic HTML ได้อย่างมีประสิทธิภาพและมีลูกเล่นมากขึ้น

หลักการทำงานของ PHP

เนื่องจาก PHP จะทำงานโดยมีตัวแปรและเอ็กซิคิวต์ที่ฝั่งเซิร์ฟเวอร์ อาจจะเรียกการทำงานว่าเป็น Server Side ส่วนการทำงานของบราวเซอร์ของผู้ใช้เรียกว่า Client Side โดยการทำงานจะเริ่มต้นที่ผู้ใช้ส่งความต้องการผ่านเว็บบราวเซอร์ทาง HTTP ซึ่งอาจเป็นการกรอกแบบฟอร์ม หรือใส่ข้อมูลที่ต้องการ ข้อมูลเหล่านั้นจะเป็นเอกสาร PHP (เอกสารเหล่านี้จะมีส่วนขยายเป็น PHP หรือ PHP3 แล้วแต่ผู้ใช้กำหนด เช่น search.php เป็นต้น) เมื่อเอกสาร PHP เข้ามาถึงเว็บเซิร์ฟเวอร์ก็จะถูกส่งไปให้ PHP เพื่อทำหน้าที่แปลงคำสั่งแล้วเอ็กซิคิวต์คำสั่งนั้น หลังจากนั้น PHPจะสร้างผลลัพธ์ในรูปแบบ เอกสารHTML ส่งกลับไปให้เว็บเซิร์ฟเวอร์เพื่อส่งต่อไปให้บราวเซอร์แสดงผลทางฝั่งผู้ใช้ต่อไป (HTTP Response) ซึ่งลักษณะการทำงานแบบนี้ จะคล้ายกับการทำงานของ CGI (Common Gateway Interface) หรืออาจจะกล่าวได้ว่า PHP ก็คือโปรแกรม CGI ประเภทหนึ่งก็ได้ ลักษณะการทำงานจะเป็นดังรูปข้างล่างนี้



รูปที่ 2.17 หลักการทำงานของ PHP

2.8 Mail Server

Mail Server มีโปรโตคอลหลักที่ใช้ในการรับส่ง Email ในเครือข่ายอินเตอร์เน็ต คือ SMTP (Simple Mail Transport Protocol) ทำหน้าที่ รับและส่ง Email ระหว่าง Mail Server มี POP (Post Office Protocol) และ IMAP (Internet Messages Access Protocol) ช่วยอำนวยความสะดวก ในการที่ไม่ต้องใช้ telnet logon เข้ามายัง Mail Server เพื่ออ่านจดหมาย แต่ใช้ โปรแกรม Email Client เช่น Outlook Express, Eudora ในการอ่านจดหมายซึ่งสะดวกกว่า

การทำงานทั่ว ๆ ไปของ E-mail โดยสรุปมีเพียง 2 ประเภท คือ การส่ง E-mail และการรับ E-mail โดยโปรโตคอล SMTP (Simple Mail Transfer Protocol) จะใช้ขณะที่ User agent (ยูสเซอร์เอเจนต์) ส่ง E-mail มาที่ MTA (Mail Transfer Agent) (เฉพาะแบบ offline) และใช้ในขณะรับและส่ง E-mail ระหว่าง MTA ด้วยกัน สำหรับการใช้ mail แบบ offline คือเครื่องที่ผู้ใช้ใช้อ่าน mail ไม่ได้ติดต่อกับเครื่องที่มี mailbox ตลอดเวลา อาจเลือกดาวน์โหลด mail มาเก็บไว้ที่เครื่องของตัวเอง จะมีโปรโตคอลสำหรับรับ E-mail ที่เกี่ยวข้อง ที่ใช้งานกันแพร่หลาย มีอยู่ 2 แบบ คือ โปรโตคอล POP (Post Office Protocol) และ IMAP (Internet Message Access Protocol) ซึ่งจะทำหน้าที่ดาวน์โหลด หรืออัพโหลด (upload) จากเครื่องของผู้ใช้ไปยังเครื่องที่มี MTA อยู่

2.8.1 POP3

POP3 เป็นโปรโตคอลที่ทำหน้าที่โหลด E-mail มาจาก MTA ไปยัง User Agent ซึ่งในปัจจุบันได้พัฒนามาจนถึง version 3 แล้ว หรือเรียกย่อ ๆ ว่า POP3 (Post Office Protocol version 3) โปรโตคอลนี้เป็นตัวแรกที่ถูกออกแบบมาเพื่อใช้รับ E-mail และเพื่อให้สนับสนุนการทำงานแบบ offline ซึ่งกลไกของ POP3 นี้จะทำงานในแบบ Offline โดยติดต่อเข้าไปยัง mail server แล้ว ดาวน์โหลด E-mail ทั้งหมดมาไว้ที่ User Agent จากนั้นจะลบ E-mail ที่ server นั้น ทิ้งไป เพื่อป้องกันการ download ซ้ำ แต่ผู้ใช้จะทำงานแบบ online กับ server ไม่ได้ เนื่องจากการอ่าน E-mail จะดึง E-mail ที่เก็บไว้ใน User Agent ขึ้นมาให้อ่านหลังจากที่ ดาวน์โหลด มาเก็บไว้ ซึ่งในขณะนั้น อาจไม่ได้ online อยู่กับ network ก็ได้

โปรโตคอลของ POP3 นี้จะทำงานในแบบของไคลเอนต์เซิร์ฟเวอร์ คือ มีโปรแกรม POP Server ใน mail server และ POP Client ในเครื่องของผู้รับ ซึ่งปกติจะฝังอยู่ในโปรแกรมที่เป็น User Agent เลย โปรแกรมทั้ง2 จะติดต่อกันโดยใช้คำสั่งที่เป็นรหัส ASCII คือเมื่อด้านที่รับทำคำสั่งก็จะทำงานตามคำสั่งนั้น แล้วตอบกลับมามีค่าเป็น (+OK) หมายถึง ทำงานได้เรียบร้อย หรือ (-ERR) หมายถึง เกิดปัญหาขึ้นทำงานไม่ได้ ซึ่งในคำสั่งที่ต้องมีการตอบกลับและส่งข้อมูลกลับมา โดยประกอบด้วยข้อมูลหลาย ๆ บรรทัดนั้น POP3 จะให้บรรทัดสุดท้ายเป็นเครื่องหมาย (.) ตามด้วย

Carriage Return และ Line Feed หมายถึงการสิ้นสุดชุดข้อมูล แต่ในกรณีที่ข้อมูลบรรทัดสุดท้าย มีข้อมูลที่เป็นจุดด้วย จะใช้เทคนิคที่เรียกว่า Character Stuffing เพื่อแก้ปัญหา โดยจะเติมจุดลงไปอีก 1 ตัว เพื่อเป็นตัวบ่งชี้ ว่าข้อมูลนั้นเป็นจุด ซึ่งจะแตกต่างจากสัญลักษณ์แสดงการสิ้นสุดของข้อมูล

การทำงานของ POP3 จะทำงานร่วมกับโปรโตคอล TCP (Transmission Control Protocol) โดยทั่วไป จะใช้ port 110 ในการติดต่อ ขั้นตอนการทำงานของ POP3 จะประกอบด้วย 3 สถานะ คือ สถานะขออนุมัติ,สถานะรับส่งรายการ และสถานะปรับปรุงข้อมูล ซึ่งในแต่ละสถานะจะรับรู้คำสั่งต่าง ๆ ของโปรโตคอลที่แตกต่างกัน ดังนี้

1.สถานะขออนุมัติ (Authorization State) เมื่อเริ่มต้นติดต่อกับเซิร์ฟเวอร์จะเป็นการ เข้าสู่สถานะการขออนุมัติ โดยไคลเอนต์จะต้องแจ้งชื่อผู้ใช้ และ รหัสผ่าน (password) เพื่อขออนุมัติจากเซิร์ฟเวอร์ก่อน โดยไคลเอนต์จะใช้คำสั่ง USER เพื่อระบุชื่อผู้ใช้ หรือคำสั่ง PASS เพื่อกำหนด Password แต่ในกรณีที่ชื่อ Password ถูกเข้ารหัสไว้ และไม่ได้เป็นค่า ASCII ทั่วไป ไคลเอนต์จะใช้คำสั่งAPOPทำงานแทนคำสั่ง USER และ PASS

2. สถานะรับส่งรายการ (Transaction State) หลังจากที่ได้รับอนุมัติจากเซิร์ฟเวอร์แล้ว ก็จะเข้าสู่สถานะที่ใช้คำสั่งในการทำงานต่าง ๆ

3.สถานะปรับปรุงข้อมูล (Update State) เมื่อ User Agent เลิกใช้งานด้วยคำสั่ง QUIT ของ POP3 เซิร์ฟเวอร์ก็จะเข้าสู่สถานะปรับปรุงข้อมูล เพื่อลบอีเมล์ที่ดาวน์โหลดเรียบร้อยแล้วออกไป จากนั้นก็จะเข้าสู่สถานะขออนุมัติใหม่โดยอัตโนมัติ เพื่อรอรับการทำงานครั้งต่อไป

สถานะขออนุมัติ (Authorization State)

เมื่อ POP3 Client ติดต่อกับ POP3 Server ก็จะแสดงบรรทัดติดต่อขึ้นมาบรรทัดหนึ่ง และบอกจุดสิ้นสุดด้วย CRLF (Carriage Return Line Feed) ตัวอย่างเช่น

s :+OK POP3 server read

เป็นการตอบรับของ POP3 ซึ่ง POP3 server จะแสดงเครื่องหมาย + บอกการตอบรับว่าในขณะนั้นสามารถให้บริการแก่ Client ตามที่ร้องขอ

เมื่อ POP3 อยู่ในฐานะ Authorization State แล้วก็จะทำการยืนยันแก่ POP3 server โดยมีวิธีการยืนยันอยู่สองวิธี คือ

- คำสั่ง USER รวมกับคำสั่ง PASS
- คำสั่ง APOP

การใช้คำสั่ง USER และคำสั่ง PASS ในขั้นแรก Client ต้องใช้คำสั่ง USER ก่อนถ้า POP3 Server ตอบมาด้วยสถานะบ่งชี้ว่าเป็นเครื่องหมาย + ("+OK") เครื่อง client ก็จะใส่คำสั่ง PASS เข้าไปในการทำงานหรือคำสั่ง QUIT เพื่อบอกสถานะว่าหยุดการทำงานถ้าหากสถานะบ่งชี้เป็นเครื่องหมาย - ("ERR") เครื่อง Client ต้องส่งคำสั่งไปใหม่หรือยกเลิกโดยใช้คำสั่ง Quit ไปเลยก็ได้ เมื่อเครื่อง Client ส่งคำสั่ง Pass แล้ว POP3 Server จะใช้ทั้งคำสั่ง USER และ PASS เพื่อพิจารณาว่าเครื่องClientใดสามารถเข้าไปใช้งานภายใน Maildropได้

POP3 Server ได้มีการจำกัดการเข้าถึงใน Maildrop เพื่อป้องกันไม่ให้ผู้ที่ไม่มีสิทธิ์เข้าไปทำการเปลี่ยนแปลงหรือลบข้อมูลใน Maildrop ก่อนจะเข้าสู่ช่วง Update State ถ้าการ lock สำเร็จ POP3 Server ก็จะตอบสนองด้วยสถานะการบ่งชี้ เป็น + ขณะนี้ POP3 ก็จะเข้าสู่ช่วง Transaction State ซึ่งไม่มี Message ที่ถูกทำเครื่องหมาย Delete ถ้าไม่สามารถเปิด Maildrop เนื่องจากเหตุผลบางประการ เช่น lock ไม่ได้ , Client ปฏิเสธการเข้าถึง Maildrop ที่เหมาะสม หรือ Maildrop ไม่สามารถกระจายข้อมูลได้, Mail Server จะแสดงสถานะบ่งชี้เป็นเครื่องหมาย - ถ้ามีการ lock แต่ POP3 Server ยังแสดงสถานะบ่งชี้เป็นเครื่องหมายลบอยู่ จะต้องดูที่สำดับการ LOCK ในการปฏิเสธคำสั่ง หลังจากได้รับตัวบ่งชี้สถานะเป็นเครื่องหมายลบ Server ก็จะปิดการติดต่อถ้า Server ยังไม่ปิดการติดต่อเครื่อง Client ก็จะส่งคำสั่งมาอีก หรือไม่ก็ใช้คำสั่ง Quit ออกไปเลยเมื่อ POP3 Server ได้เปิด Maildrop ก็จะส่งหมายเลข Message ไปยังแต่ละ Message ซึ่งขนาดของแต่ละ Messagee จะอยู่ในรูปของเลขฐาน 8 ข้อความแรกใน Maildrop จะได้รับหมายเลข Message เป็น 1ลำดับที่สอง ก็เป็น 2 ตามลำดับไปเรื่อยๆ คำสั่ง POP3 และหมายเลขจะเป็นเลขฐาน 10

ตารางที่ 2.2 สรุปรายละเอียดของคำสั่งต่างๆใน POP3

คำสั่ง	พารามิเตอร์	สถานะ	รายละเอียด
USER	ชื่อผู้ใช้งาน	ขออนุมัติ	แจ้งชื่อผู้ใช้และระบุ Mailbox ที่จะใช้
PASS	Password	ขออนุมัติ	เป็นคำสั่งที่ใช้ระบุ Password โดยจะใช้ต่อจากคำสั่ง USER
APOP	ชื่อ, Password	ขออนุมัติ	ทำหน้าที่เหมือนคำสั่ง USER และ PASS รวมกันแต่ข้อมูลจะถูกเข้ารหัสก่อนส่งไป
STAT	ไม่ระบุ	รับส่งรายการ	เป็นคำสั่งตรวจสอบ

			สภาพเซิร์ฟเวอร์,ขนาดของอีเมล์ที่จะดาวน์โหลด
UIDL	หมายเลขข้อความ	รับส่งรายการ	ใช้ตรวจสอบหมายเลขประจำของอีเมล์
LIST	หมายเลขข้อความ	รับส่งรายการ	ใช้ตรวจสอบหมายเลขของอีเมล์ และขนาดของอีเมล์
RETR	ข้อความ	รับส่งรายการ	เป็นคำสั่งที่ใช้ส่งข้อมูลของอีเมล์
DELE	ข้อความ	รับส่งรายการ	เป็นการระบุเครื่องหมายการลบลงในอีเมล์ที่จะลบ และอีเมล์เหล่านั้นจะถูกลบออกจากเมล์บ็อกซ์เมื่อใช้คำสั่ง QUIT เมื่อสิ้นสุดการทำงาน
RSET	ไม่ระบุ	รับส่งรายการ	คำสั่งนี้จะยกเลิกเครื่องหมายการลบอีเมล์ที่เคยกำหนดไว้ด้วยคำสั่ง DELE ออกไปทุก ๆ อีเมล์
TOP	หมายเลขข้อความ,จำนวนบรรทัด	รับส่งรายการ	เซิร์ฟเวอร์จะส่งข้อมูลย้อนกลับไปเท่ากับจำนวนบรรทัดที่ระบุ
NOOP	ไม่ระบุ	รับส่งรายการ	เป็นคำสั่ง No Operation
QUIT	ไม่ระบุ	รับส่งรายการและขออนุมัติ	ใช้เมื่อจบการทำงาน หากมีอีเมล์ซึ่งทำเครื่องหมายว่าจะลบไว้ อีเมล์เหล่านั้นจะถูกลบจากเมล์บ็อกซ์ในขั้นตอนนี้

2.8.2 SMTP

SMTP เป็นโปรโตคอลที่ใช้ส่งอีเมล์จาก User agent ของผู้ส่งไปยัง MTA ของผู้ส่ง และส่งต่อไปยัง MTA เครื่องอื่น ๆ ที่เป็นจุดผ่านในการเชื่อมต่อไปยังเครื่องของผู้รับ โปรโตคอล SMTP จะทำงานร่วมกับโปรโตคอล TCP โดยใช้พอร์ต 25 ซึ่งคำสั่งต่าง ๆ ของ SMTP จะเป็นลักษณะเดียวกับ POP3 คือเป็น ASCII ลงท้ายด้วย Carrige Return และ Line Feed ส่วนข้อความที่ตอบกลับมาจะนำหน้าด้วยเลข 3 หลัก เป็นสัญลักษณ์แสดงลักษณะการทำงานของคำสั่งที่ได้รับ

เมื่อเริ่มต้นการติดต่อ SMTP จะกำหนดให้ User Agent ของผู้ส่งต้องส่งคำสั่ง HELLO พร้อมกับรายละเอียดด้านผู้ส่งออกไป จากนั้นจะส่งคำสั่ง MAIL เพื่อแจ้งให้เซิร์ฟเวอร์เตรียมรับอีเมล์ ในส่วนของเซิร์ฟเวอร์เมื่อพร้อมที่จะรับอีเมล์ก็ตอบรับอีเมล์ก็รับอีเมล์ก็ตอบรับกลับมาด้วยคำสั่ง OK จากนั้นที่ด้านส่งก็จะเริ่มส่งโดยใช้คำสั่ง RCPT เพื่อกำหนดอีเมล์แต่ละฉบับไปที่ส่งไป ซึ่งการส่งข้อมูลของอีเมล์ก็จะระบุด้วยคำสั่งDATA

การส่งอีเมล์ของโปรโตคอล SMTP ได้จัดเตรียมคำสั่งอื่น ๆ ไว้เพื่ออำนวยความสะดวกและคล่องตัวในการทำงาน ซึ่งประกอบด้วย คำสั่ง VRFY เพื่อให้ด้านที่ส่งตรวจสอบรายชื่อว่าผู้ใช้รายนี้มีสิทธิ์ใช้งาน E-mailbox นั้น ๆ หรือไม่ , คำสั่ง EXPN ใช้จัดการและตรวจสอบรายชื่อจากลิสต์รายชื่อ และคำสั่ง TURN ใช้สลับให้ Client ของผู้ส่งทำหน้าที่รับข้อมูลจากเซิร์ฟเวอร์แทน

เมื่อได้รับคำสั่งต่างๆ ของผู้ส่งแล้ว เซิร์ฟเวอร์จะมีหน้าที่ตรวจสอบความถูกต้องของคำสั่ง จากนั้น จึงทำงานตามคำสั่ง แล้วส่งผลตอบกลับมา ส่วนลักษณะของข้อมูลที่ตอบกลับ (Reply Message) นั้นจะเป็นข้อมูลที่อยู่ในรูปของ Text ที่เป็น ASCII โดยจะประกอบด้วยตัวเลข นำหน้าข้อความสามหลัก ทำหน้าที่แสดงสถานะการทำงานของเซิร์ฟเวอร์ และเปลี่ยนสถานะการทำงานของโปรโตคอล SMTP ด้วย ถัดจากตัวเลขจะคั่นด้วยช่องว่างแล้วตามด้วยเครื่องหมาย Carriage Return และ Linc Feed ตัวอย่างเช่น 500 Syntax error, command unreconized หมายถึง คำสั่งที่ส่งไปไม่ถูกต้อง หรือ 503 Bad sequence of commands หมายถึง ลำดับการส่งคำสั่งไม่ถูกต้องเหล่านี้เป็นต้น

ในการส่ง (E-mail) ของโปรโตคอล SMTP นั้น จะใช้วิธีอ้างถึงเซิร์ฟเวอร์อื่น ๆ ตามแบบ DNS (Domain Name System) เช่นเดียวกับระบบอื่น ๆ ในอินเทอร์เนต และยังสามารถส่งอีเมล์ๆไปยังผู้รับคนเดียวหรือหลาย ๆ คนพร้อมกันได้ด้วย

ตารางที่ 2.3 แสดงรายละเอียดในคำสั่งต่าง ๆ ของ SMTP ที่ใช้งานอยู่

คำสั่ง	รายละเอียด
DATA	จะเป็นคำสั่งที่ใช้ต่อจาก RCPT เพื่อส่งข้อมูลของอีเมล์
SEND	ทำหน้าที่เหมือนคำสั่ง DATA แต่ไม่ค่อยมีผู้ใช้งาน
SOML	ทำหน้าที่เหมือนคำสั่ง DATA แต่ไม่ค่อยมีผู้ใช้งาน
SAML	ทำหน้าที่เหมือนคำสั่ง DATA แต่ไม่ค่อยมีผู้ใช้งาน
VRFY	เป็นคำสั่งที่ใช้เพื่อตรวจสอบความถูกต้องของชื่อและเมล์บ็อกซ์
EXPN	เป็นคำสั่งเพื่อตรวจสอบรายละเอียดของลิสต์รายชื่อ
HELP	ใช้ตรวจสอบคำสั่งที่สามารถใช้งานได้กับเซิร์ฟเวอร์
NOOP	เป็นคำสั่ง No Operation เมื่อเซิร์ฟเวอร์ได้รับคำสั่งนี้จะตอบ OK กลับมา
QUIT	สิ้นสุดการติดต่อ
RSET	ยกเลิกการส่งข้อมูลในขณะนี้
TURN	เป็นคำสั่งที่สลับหน้าที่ของผู้ส่งข้อมูลมาทำหน้าที่รับข้อมูลแทน
RCPT	เป็นคำสั่งเพื่อระบุอีเมล์ที่จะส่งทีละฉบับ โดยเป็นคำสั่งที่ใช้ต่อจาก MAIL

SMTP Reply Codes (ตัวเลขแสดงสถานะของคำสั่ง)

211 System status, or system help reply

214 Help message

220 Service ready

221 Service closing transmission channel

250 Requested mail action okay ,completed

251 User not local ; will forward to

354 Start mail input ; end with .

421 Service not available, closing transmission channel

450 Request mail action not taken : mailbox unavailable

451 Request action aborted : local error in processing

452 Requested action not taken : insufficient system storage

500 Syntax error in parameters or arguments

501 Syntax error in parameters or arguments

502 Command not implemented

503 Bad sequence of commands

504 Command parameter not implemented

550 Requested action not taken : mailbox unavailable

551 User not local ; please try

552 Requested mail action aborted : exceeded storage allocation

553 Requested action not taken : mailbox name not allowed

554 Transaction failed

2.8.3 IMAP

IMAP ก็คือ ผู้ใช้สามารถเลือกดาวน์โหลดเฉพาะ (E-Mail) ที่ต้องการได้ โดยไม่จำเป็นต้องโหลดมาทั้งหมดเหมือนโปรโตคอล POP3 นอกจากนี้ IMAP ยังสามารถรองรับการทำงานได้ทั้งแบบ Offline, Online และแบบ Disconnected อีกด้วย ดังนั้นประโยชน์ที่ได้จาก IMAP ก็คือ หากผู้ใช้มีอีเมล์แอดเดรสเพียงชื่อเดียว แต่มีเครื่องที่ใช้งานอยู่หลายเครื่องก็จะสามารถเลือกดาวน์โหลดเฉพาะอีเมล์ที่ต้องการมาเก็บไว้ที่เครื่องใดก็ได้ แต่ถ้าเป็น POP3 การดาวน์โหลดจะต้องทำพร้อมกันหมดทุกอีเมล์ ดังนั้น IMAP จึงเป็นโปรโตคอลที่สามารถทำงานกับสายสื่อสารที่มีความเร็วต่ำได้เป็นอย่างดี

การทำงานของ IMAP นี้เหมือนกับโปรโตคอลอื่น ๆ โดยทำงานร่วมกับ TCP ใช้พอร์ตหมายเลข 143 และแบ่งเป็นสถานะต่าง ๆ ออกเป็น 4 สถานะ โดยในแต่ละสถานะมีวัตถุประสงค์และคำสั่งที่ใช้งานแตกต่างกัน มีรายละเอียดต่าง ๆ ดังนี้

1. สถานะก่อนอนุมัติ (Non-authenticated State) เป็นสถานะที่กำลังรอให้ไคลเอนต์ติดต่อเข้ามาเพื่อขออนุมัติใช้ ดังนั้นในด้านไคล์เอนต์จะต้องแจ้งชื่อล็อกอิน (Login) ของ Mail Server นั้นและ password (พาสเวิรด์) ด้วยคำสั่ง LOGIN หรือ AUTHENTICATE ก่อนจึงจะเริ่มใช้งานได้ จากนั้นจึงเปลี่ยนไปเป็นสถานะได้รับการอนุมัติ

2. สถานะได้รับการอนุมัติ (Authenticated State) เป็นสถานะที่สามารถใช้คำสั่งต่าง ๆ ที่เกี่ยวกับการเลือกและใช้งานเมล์บอกซ์ เช่น คำสั่ง SELECT เพื่อเลือกเมล์บอกซ์ หรือคำสั่ง CREATE เพื่อสร้างเมล์บอกซ์ เป็นต้น ในการเลือกเมล์บอกซ์ด้วยคำสั่ง SELECT หรือ EXAMINE นี้จะเปลี่ยนไปเป็นสถานะการเลือกเมล์บอกซ์

3. สถานะการเลือกเมล์บอกซ์ (Selected State) เป็นสถานะที่จะเข้าใช้งานอีเมล์ในแต่ละเมล์บอกซ์หลังจากที่เลือกเมล์บอกซ์ไว้แล้วในสถานะก่อนหน้านี้

4. สถานะเลิกใช้งาน (Logout State) เมื่อต้องการเลิกใช้งาน หรือสิ้นสุดการทำงานของ IMAP จะเข้าสู่สถานะเลิกใช้งานโดยใช้คำสั่ง LOGOUT

จากสถานะทั้ง 4 นี้ไม่จำเป็นต้องทำงานเรียงต่อกันเสมอไปบางครั้งอาจจะมีการทำงาน ข้ามจากสถานะหนึ่งไปยังอีกสถานะหนึ่งได้ ตัวอย่างเช่น เมื่อเข้าสู่สถานะที่ได้รับการอนุมัติ (Authenticate State) และลบอีเมล์ที่ไม่ต้องการใช้งานทิ้งไปด้วยคำสั่ง DELETE แล้วและไม่ต้องการทำงานอื่น ๆ ต่อก็สามารถใช้คำสั่ง LOGOUT เพื่อเปลี่ยนสถานะเป็นเลิกการใช้งาน (Logout State) ได้โดยไม่จำเป็นต้องเข้าสู่สถานะการเลือกเมล์บอกซ์ (Selected State) ก่อน ซึ่งจะเรียกว่า Untagged response หรือ Asterisk tag (*) ส่วนคำตอบของเซิร์ฟเวอร์ที่เป็นผลการทำงานตามคำสั่งต่าง ๆ ของไคลเอนต์นั้น จะประกอบด้วยคำสั่งทั้งหมด 5 แบบ คือ

- OK แสดงว่าผลการทำงานของคำสั่งนั้น ๆสำเร็จและเรียบร้อยแล้ว
- NO แสดงว่าการทำงานตามคำสั่งนั้นไม่สำเร็จ
- BAD แสดงว่าคำสั่งที่ส่งไปไม่ถูกต้อง หรือมีพารามิเตอร์ไม่ถูกต้อง
- PREAUTH แสดงว่าไม่จำเป็นต้องใช้คำสั่ง LOGIN เพราะเคยมีการสถานะที่ได้รับการอนุมัติ (Authenticated State) แล้ว
- BYE แสดงว่าเซิร์ฟเวอร์ได้จบการทำงานไปแล้ว

2.9 FTP Server

FTP เป็นคำย่อมาจากภาษาอังกฤษว่า Flie Transfer Protocol เป็นการถ่ายโอนแฟ้มข้อมูลระหว่างเครื่องคอมพิวเตอร์ 2 เครื่อง ซึ่งอยู่บนเครือข่ายอินเทอร์เน็ต ในระบบเครือข่ายอินเตอร์เน็ตมีเครื่องคอมพิวเตอร์ที่ให้บริการการถ่ายโอนแฟ้มข้อมูลเป็นสาธารณะ มีอยู่เป็นจำนวนมาก เรียกเครื่องคอมพิวเตอร์ที่ให้บริการถ่ายโอนแฟ้มข้อมูลนี้ว่า FTP Server

คำสั่งต่างๆที่ใช้ใน FTP

- คำสั่ง dir (สำหรับโปรแกรม FTP ที่เรียกใช้จากดอส หรือ วินโดว์)
  dir เป็นคำสั่งสำหรับดูชื่อแฟ้มข้อมูลและชื่อไดเรคทอรีใน เครื่องคอมพิวเตอร์ปลายทาง

```
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
total 5
drwx--x--x  3 root     daemon        512 Jan 31 17:33 bin
drwx--x--x  2 root     daemon        512 Jan 31 17:30 dev
drwx--x--x  2 root     daemon        512 Jan 31 17:34 etc
drwxr-xr-x 10 root     daemon        512 Feb  6 11:10 pub
drwx--x--x  3 root     daemon        512 Jan 31 17:37 usr
226 Transfer complete.
804 bytes received in 0.066 seconds (4.5 Kbytes/s)
ftp>
```

- คำสั่ง cd [ไดเรคทอรี]

  cd เป็นคำสั่งที่ใช้เปลี่ยนไปยังไดเรคทอรีย่อยของเครื่องคอมพิวเตอร์ปลายทาง ที่ต้องการ

```
ftp> cd pub
250 CWD command successful.
ftp> cd pics/bmp
250 CWD command successful.
```

- คำสั่ง cd .. หรือ cdup

  cd .. หรือ cdup เป็นคำสั่งที่ใช้เปลี่ยนไดเรคทอรีขึ้นไปอีกหนึ่งระดับของ เครื่องคอมพิวเตอร์ปลายทาง

```
ftp> cd..
250 CWD command successful.
```

- คำสั่ง get [ชื่อแฟ้มต้นทาง] [ชื่อแฟ้มปลายทาง]

  get เป็นคำสั่งที่ใช้คัดลอกแฟ้มจากคอมพิวเตอร์ปลายทาง ไปยังคอมพิวเตอร์ต้นทาง ในกรณีที่ไม่พิมพ์ชื่อแฟ้มปลายทาง เครื่องจะตั้งชื่อแฟ้มปลายทางเหมือนกับชื่อแฟ้มต้นทาง

```
ftp> get door.bmp door.bmp
200 PORT command successful.
150 Opening ASCII mode data connection
226 Transfer complete.
local: door.bmp remote: door.bmp
163958 bytes received in 0.84 seconds (
ftp>
```

- **คำสั่ง mget [ชื่อแฟ้ม] [ชื่อแฟ้ม]**

  mget เป็นคำสั่งคัดลอก แฟ้มหลาย ๆ แฟ้มจาก คอมพิวเตอร์ปลายทางตาม รูปแบบที่กำหนดมาที่คอมพิวเตอร์ต้นทาง โดยเครื่องจะถามความต้องการที่ละแฟ้มข้อมูล

ถ้าไม่ต้องการให้เครื่องถามทีละไฟล์ให้ใช้คำสั่ง prompt เข้าช่วย

```
ftp> prompt
Interactive mode off.
ftp> mget *.bmp
```

- **คำสั่ง quit หรือ bye**

  quit หรือ bye เป็นคำสั่งที่ใช้ออกจาก ftp

ตารางที่ 2.4 คำสั่งต่างที่ใช้ใน FTP

? [คำสั่ง] / help [คำสั่ง]	**แสดงข้อความช่วยเหลือ อธิบายคำสั่งใน ftp**
asscii	คัดลอกแฟ้มข้อมูลแบบแอสกี
binary	คัดลอกแฟ้มข้อมูลแบบไบนารี
bell	ให้ส่งเสียงเมื่อคัดลอกแฟ้มข้อมูลเสร็จ
bye	จบการทำงานและออกจาก ftp
cd [ไดเรกทอรี]	เปลี่ยนไดเรคทอรี ของคอมพิวเตอร์ปลายทาง
cd .. หรือ cdup	เปลี่ยนไดเรกทอรีของคอมพิวเตอร์ปลายทางขึ้นไปหนึ่งระดับ
lcd [ไดเรคทอรี]	เปลี่ยนไดเรคทอรีของคอมพิวเตอร์ปลายทาง
close หรือ disconnect	จบการเชื่อต่อกับคอมพิวเตอร์ปลายทางแต่ยังไม่ออกจาก ftp
dir [ชื่อแฟ้ม]	แสดงรายชื่อแฟ้มของคอมพิวเตอร์ปลายทาง
get [ชื่อแฟ้ม] [ชื่อแฟ้ม]	คัดลอกแฟ้มจากคอมพิวเตอร์ปลายทางมาที่คอมพิวเตอร์ต้นทาง
mget [ชื่อแฟ้ม] [ชื่อแฟ้ม]	คัดลอกแฟ้มจากคอมพิวเตอร์ปลายทางมาที่คอมพิวเตอร์ต้นทางแบบหลายแฟ้ม
put [ชื่อแฟ้ม] [ชื่อแฟ้ม]	คัดลอกแฟ้มจากคอมพิวเตอร์ต้นทางไปไว้ที่คอมพิวเตอร์ปลายทาง
mput [ชื่อแฟ้ม] [ชื่อแฟ้ม]	คัดลอกแฟ้มจากคอมพิวเตอร์ต้นทางไปไว้ที่ คอมพิวเตอร์ปลายทางแบบหลายแฟ้ม
prompt [on] [off]	กำหนดให้มีการโต้ตอบกับผู้ใช้เพื่อเลือกแฟ้มเมื่อใช้ mget ,mput
pwd	แสดงไดเรคทอรีของรีโมตโฮสต์

2.9.1 VSFTP

FTP มีหลากหลายชนิด ไม่ว่าจะเป็น WU-FTP, ProFTP, หรือ BSD-FTP ล้วนมีช่องโหว่ที่ถูกค้นพบมากมาย จึงทำให้มีความเสี่ยงต่อการถูกผู้ไม่ประสงค์ดีโจมตีได้ อย่างไรก็ตาม มีอีกโปรแกรมหนึ่งชื่อ VSFTP ซึ่งมีออปชันด้านความปลอดภัยสำหรับเซิร์ฟเวอร์ FTP ที่เป็น Linux/UNIX ดังนั้น VSFTP จึงเป็นอีกทางเลือกหนึ่งที่จำเป็นต้องติดตั้งเซิร์ฟเวอร์ FTP และต้องการความปลอดภัยยิ่งขึ้น

คุณสมบัติของVSFTP

VSFTP เป็นเดมอน (daemon) ของเซิร์ฟเวอร์ FTP ที่รันบนระบบปฏิบัติการ Linux/UNIX ซึ่งมีคุณสมบัติหลักอยู่ 3 อย่าง ได้แก่

- มีความปลอดภัย
- มีประสิทธิภาพดี
- มีความมั่นคง

ในแง่ของความปลอดภัยนั้น VSFTP ได้ถูกออกแบบมาเพื่อแก้ไขจุดบกพร่องที่พบในการติดตั้ง wu-ftpd, proftpd, และแม้แต่ bsd-ftpd โดยการ ไม่ใช้บัญชีชื่อ root ซึ่งมีความเสี่ยง และใช้คุณสมบัติด้านความปลอดภัยที่มีประสิทธิภาพอย่าง chroot นอกจากนี้ยังมีการใช้เทคนิคการพัฒนาโปรแกรมอย่างปลอดภัยเพื่อแก้ปัญหาหน่วยความจำล้น (buffer overflows) ด้วย และนอกจากนี้ยังมีการกล่าวถึง vsftpd ในแง่ของความปลอดภัยมากมาย ดังต่อไปนี้

- vsftpd ไม่มีจุดบกพร่องชื่อ "globbing" ซึ่งมีผลกระทบต่อเซิร์ฟเวอร์ FTP หลายแห่ง
- vsftpd ไม่มีจุดบกพร่องชื่อ "globbling" ที่เกี่ยวข้องกับการโจมตีแบบ denial of service

วิธีเริ่มต้นการทำงานของ VSFTP

VSFTP สามารถรันได้ 2 โหมด ได้แก่

1. โหมด stand-alone
2. โหมด inetd/xinetd

การรันผ่านเดมอน inetd หรือ xinetd สามารถควบคุมการทำงานของโปรแกรมได้ง่ายกว่า และเป็นวิธีที่ควรใช้มากกว่าแบบ stand-alone สิ่งสำคัญที่ต้องคำนึงถึงคือ ในการตั้งค่าของ VSFTP นั้น มันจะรับการเชื่อมต่อแบบ anonymous เท่านั้น (สมมติว่าได้สร้างบัญชีชื่อ "ftp" ไปก่อนหน้านี้แล้ว) ดังนั้นถ้าต้องการอนุญาตให้บัญชีผู้ใช้ในเครื่องสามารถเชื่อมต่อเข้ามาใช้งาน ftp ได้ จะต้องทำการตั้งค่า Pluggable Authentication Modules (PAM) ซึ่งจะกล่าวถึงต่อไปด้วย

1.การทำงานแบบ stand-alone

สิ่งที่ต้องทำสำหรับการรัน VSFTP ในโหมด stand-alone ได้แก่

- เพิ่มประโยค "listen = YES" ไว้ในบรรทัดสุดท้ายของไฟล์ /etc/vsftpd.conf
- รันคำสั่ง # /usr/local/sbin/vsftpd &

2.การใช้ xinetd

เนื่องจากในที่นี้ได้ทำการทดสอบและใช้งานบนระบบ Red Hat 9 (ซึ่งใช้ xinetd) ดังนั้นจะเน้นที่การดำเนินการติดตั้งในโหมด xinetd

- กรณีรัน inetd
  - เพิ่มบรรทัดต่อไปนี้ในไฟล์ /etc/inetd.d

    ftp stream tcp nowait root /usr/local/sbin/vsftpd
  - รีสตาร์ทเดมอน inetd ด้วยคำสั่ง

    # kill -SIGHUP <pid of inetd>
- กรณีรัน xinetd
  - ไฟล์คอนฟิกูเรชันสำหรับการเริ่มต้นทำงานของ VSFTP ในเครื่องที่รัน xinetd นั้น อยู่ภายใต้ไดเรกทอรี /etc/xinetd.d โดยมีชื่อว่า vsftpd หากไม่พบไฟล์ดังกล่าว ให้ทำการสำเนาจากตัวอย่างของ vsftpd ที่มากับ VSFTP distribution (/tmp/vsftpd-1.2.0/xinetd.d/vsftpd) มาไว้ในไดเรกทอรีนี้ ในไฟล์ vsftpd นี้ มีพารามิเตอร์จำนวนหนึ่งภายใต้ "service ftp" ซึ่งระบุให้ทราบวิธีการทำงานของเซิร์ฟเวอร์ VSFTP ดังต่อไปนี้

ตารางที่ 2.5 แสดงพารามิเตอร์ภายใต้ service ftp

พารามิเตอร์	ค่าที่กำหนด	ความหมาย
socket_type	stream	เป็นชนิดของ TCP socket ที่ใช้สำหรับโพรโตคอลนี้ นั่นคือ FTP เป็น TCP stream
wait	no	เกี่ยวกับความสามารถสำหรับ socket ที่จะรับข้อความ
user	root	ผู้ใช้ที่จะเป็นผู้เปิดให้บริการ ftp คือใคร (ในที่นี้คือ root) หมายเหตุ : VSFTP จะลดสิทธิลงทันทีที่เริ่มทำงาน
server	/usr/local/sbin/vsftpd	เป็นตำแหน่งของโปรแกรมเซิร์ฟเวอร์ที่เกี่ยวข้องกับไฟล์คอนฟิกูเรชันนี้ ถ้ามีการตั้ง vsftpd ในไดเรกทอรีที่แตกต่างออกไปก็ต้องแก้ไขเปลี่ยนแปลงค่านี้
nice	10	ออปชันนี้ใช้แก้ไขลำดับตารางเวลาดีฟอลต์สำหรับโพรเซส โดย

		10 คือค่าดีฟอลต์ ส่วนค่าสูงสุดคือ 20
disable	no	เซอร์วิสนี้ไม่ถูกปิด หรืออีกนัยหนึ่งคือ มันควรเริ่มทำงานทันทีเมื่อ xinetd เริ่มต้นทำงาน
per_source	5	เป็นการระบุจำนวนของการเชื่อมต่อเข้ามาใช้งานในเวลาเดียวกันจากหมายเลข IP เดียวกัน โดยเพื่อความปลอดภัย ควรตั้งค่าไว้เป็น 5
instances	200	กำหนดจำนวนสูงสุดของการเชื่อมต่อ FTP ไปยังเซิร์ฟเวอร์ในเวลาเดียวกัน ซึ่งมีประโยชน์ต่อการกำหนดโหลดการทำงานของเซิร์ฟเวอร์ โดยเพื่อความปลอดภัยควรตั้งค่าไว้เป็น 200
no_access	No default	เป็นการระบุรายการ IP address ที่ไม่อนุญาตให้เข้าถึงเซอร์วิสนี้ เช่น 192.168.1.4

-เมื่อปรับแก้ไฟล์ /etc/xinetd.d/vsftpd เรียบร้อยแล้วให้ทำการรีสตาร์ท xinetd ด้วยคำสั่ง

```
# /etc/init.d/xinetd restart
หรือ
# service xinetd restart
```

- และเพื่อให้ xinetd เริ่มทำงานทุกครั้งที่เครื่องรีสตาร์ท ให้ใช้คำสั่งต่อไปนี้

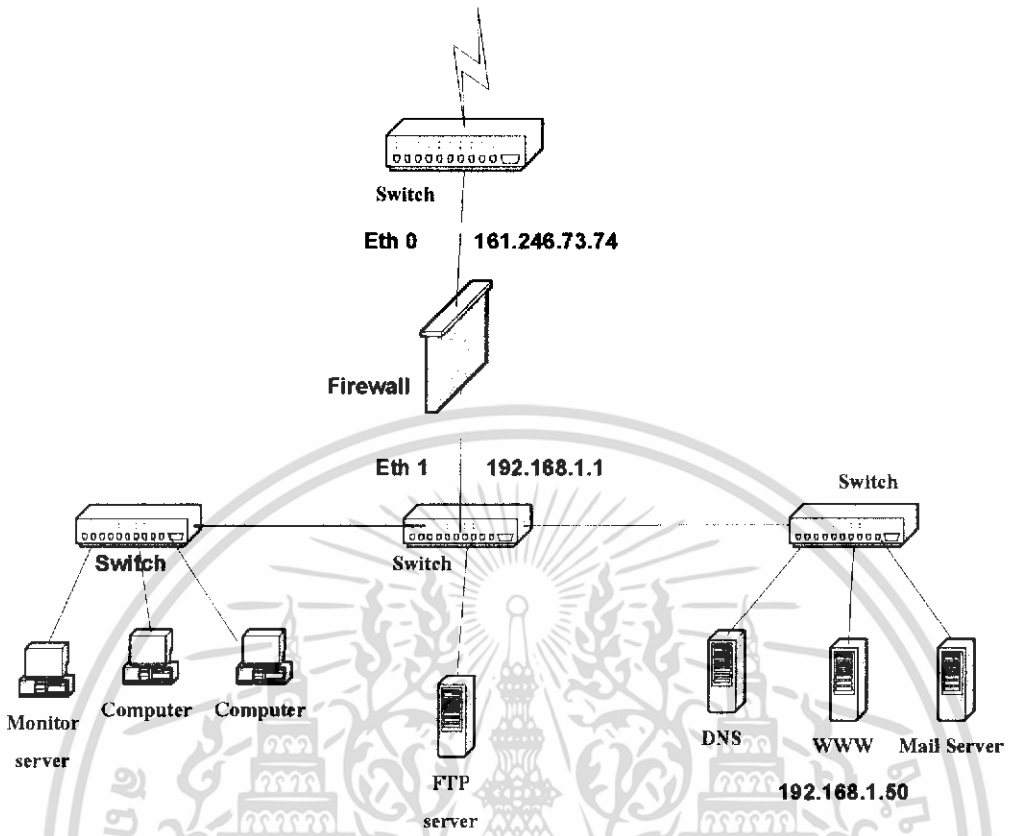
```
# chkconfig --add xinetd
# chkconfig --level 345 xinetd on
```

บทที่ 3

การออกแบบระบบเครือข่าย

3.1 ขั้นตอนการออกแบบระบบเครือข่าย

1. วิเคราะห์ความต้องการของผู้ใช้ในระบบเครือข่ายห้องวิจัย ความต้องการของผู้ใช้ ได้แก่
    - ต้องการระบบเครือข่ายภายในที่เป็นสัดส่วน แยกจากเครือข่ายภายนอกอย่างชัดเจน
    - ต้องการความปลอดภัยจากการบุกรุกจากบุคคลภายนอกที่ไม่หวังดี
    - ต้องการความปลอดภัยจากการโจมตีจากเครือข่ายภายนอก เช่น ไวรัส โทรจัน
    - ต้องการความสะดวกสบายในการใช้งานเครือข่าย ไม่พบปัญหาในการใช้งาน เช่น การชนกันของไอพี
    - มีการดูแลเครื่อง client ภายในระบบเครือข่าย เช่น การแจ้งเตือนไวรัส
    - ต้องการความสะดวกสบายในการใช้งาน มีบริการต่าง ๆ ในระบบเครือข่าย
    - ต้องการความปลอดภัยระดับสูงแก่เซิร์ฟเวอร์ฐานข้อมูล
2. ศึกษาข้อบกพร่องของระบบเครือข่ายเดิมเพื่อทำการพัฒนาให้ดีขึ้น
3. ออกแบบระบบเครือข่ายเพื่อใช้งานในห้องวิจัย โดยคำนึงถึง
    - ประสิทธิภาพ (Efficiency)
    - ความเป็นประโยชน์ (Available)
    - ความน่าเชื่อถือ (Reliable)
    - ระดับความปลอดภัย
    - งบประมาณ
    - ความยากง่ายในการจัดการและตั้งค่า



รูปที่ 3.1 ระบบเน็ตเวิร์คที่ออกแบบมาสำหรับห้องวิจัย

จากรูปจะเห็นว่าเป็นระบบเครือข่ายภายใน ที่มีการแยกออกจากเครือข่ายภายนอก โดยใช้ไอพีแอดเดรสที่เป็น Private IP ซึ่งเป็นไอพีที่ไม่สามารถเชื่อมต่อกับเครือข่ายภายนอกได้โดยตรง ดังนั้นจึงต้องทำ NAT (Network Address Translation) เพื่อแปลง Private IP เป็น Public IP และแปลง Public IP เป็น Private IP เพื่อติดต่อกับเครือข่ายภายนอก

การออกแบบเครือข่ายประกอบด้วย

3.2 การออกแบบไฟร์วอล

ในการออกแบบได้ใช้ไฟร์วอล เพื่อเพิ่มความปลอดภัยและเสถียรภาพ ของระบบเครือข่ายภายใน

ไฟร์วอลจะเชื่อมต่อกับเครือข่ายภายนอก ซึ่งมีหน้าที่ คือ เป็นเราต์เทอร์ในการหาเส้นทาง และกำหนดกฎการเข้า-ออก ของแพ็กเก็ตของข้อมูล และทำการแปลงแอดเดรสระหว่าง Private IP และ Public IP ซึ่งมีกฎดังนี้

สำหรับแพ็กเก็ตขาเข้า (Inbound Packet)

กำหนดให้กฎดีฟอล์ท ซึ่งเป็นกฎสุดท้ายเมื่อเงื่อนไขแพ็กเก็ตที่ผ่านเข้ามาไม่ตรงกับเงื่อนไขใด ๆ เลย ให้เป็น ยกเลิก (DROP) คือไม่ยอมให้แพ็กเก็ตนั้นผ่านเข้ามายังตัวไฟร์วอลล์ได้

INPUT Chain (แพ็คเก็ตที่อนุญาตให้ผ่านเข้ามาได้)

ตารางที่ 3.1 บอกถึงแพ็คเก็ตที่อนุญาตให้ผ่านเข้ามาได้

Input Interface	Output Interface	Protocol	Match	Source	Destination	Port	Target
Eth0	-	TCP	-	Any	161.246.73.74	22	ACCEPT
Eth1	-	TCP	-	Any	192.168.1.1	22	ACCEPT
Eth0,Eth1	-	ICMP type 8 (request)	Limit 10/minute	Any	Any		ACCEPT
Eth0,Eth1	-	ICMP type 0 (reply)	Limit 10/minute	Any	Any		ACCEPT
Eth0,Eth1	-	TCP(syn)	Limit 1/s	Any	Any		ACCEPT
Eth0,Eth1	-	-	State ESTABLISHED,RELATED	-	-		ACCEPT
Eth0	-	-	state NEW	-	-		DROP

แพ็คเก็ตที่อนุญาตให้ผ่านเข้ามาได้ แสดงดังตารางที่ 3.1 ซึ่งมีการทำงานดังนี้

- บรรทัดที่หนึ่งเป็นการอนุญาตให้ข้อมูลแพ็กเก็ตเป็นโพรโตคอล TCP พอร์ท 22 (shell) ที่มาจากอินเตอร์เฟสภายนอก (Eth0) และปลายทางเป็น 161.246.73.74 ผ่านเข้ามายังตัวไฟร์วอลได้
- บรรทัดที่สองเป็นการอนุญาตให้ข้อมูลแพ็กเก็ตเป็นโพรโตคอล TCP พอร์ท 22 (shell) ที่มาจากอินเตอร์เฟสภายใน (Eth1) และปลายทางเป็น 192.168.1.1 ผ่านเข้ามายังตัวไฟร์วอลได้
- บรรทัดที่สามและบรรทัดที่สี่ เป็นการอนุญาตให้ข้อมูลแพ็กเก็ตที่เป็นโพรโตคอล ICMP Type 8,0 ซึ่งเป็น Echo Request และ Echo Reply สามารถผ่านเข้ามายังตัวไฟร์วอลล์ได้ โดยจำกัดจำนวนครั้ง 10 ครั้ง/นาที เพื่อป้องกันการโจมตีแบบ PING FLOOD

- บรรทัดที่ห้าเป็นอนุญาตให้ข้อมูลแพ็กเก็ตเป็นโพรโตคอล TCP ซึ่งติดต่อโดยวิธี Three Ways Handshake สามาถส่งแพ็กเก็ต SYN ติดต่อกันได้สูงสุดไม่เกิน 1 ครั้ง/วินาที เพื่อป้องกันการโจมตีแบบ SYN FLOOD
- บรรทัดที่หกเป็นอนุญาตให้ข้อมูลแพ็กเก็ตต่าง ๆ ที่ไม่ได้เกิดจากการเริ่มต้นการติดต่อจากเครือข่ายภายนอก สามารถผ่านเข้ามายังไฟร์วอลได้
- บรรทัดที่เจ็ดเป็นการตรวจสอบแพ็กเก็ตที่มาจากภายนอกว่าเป็นแพ็คเก็ตที่เป็นตัวสร้าง การติดต่อใหม่หรือไม่ ถ้าใช่จะไม่ให้ผ่านไปยังไฟร์วอล (DROP)

สำหรับแพ็กเก็ตขาออก (Outbound Packet)

กำหนดให้กฎดีฟอล์ทซึ่งเป็นกฎสุดท้ายเมื่อเงื่อนไขแพ็กเก็ตที่ผ่านเข้ามาไม่ตรงกับเงื่อนไขใดๆ เลย ให้เป็นยอมรับ (accept) คือ ยอมให้แพ็กเก็ตนั้นผ่านเข้ามายังตัวไฟร์วอลได้

OUTPUT Chain (แพ็คเก็ตที่อนุญาตให้ผ่านออกไปได้)

ตารางที่ 3.2 บอกถึงแพ็คเก็ตที่อนุญาตให้ผ่านออกไปได้

Input Interface	Output Interface	Protocol	Match	Source	Destination	Port	Target
-	Any	Any	-	Any	Any	Any	ACCEPT

แพ็คเก็ตที่อนุญาตให้ผ่านออกไปได้ แสดงดังตารางที่ 3.2

สำหรับแพ็กเก็ตส่งต่อ (Forward Packet)

กำหนดให้กฎดีฟอล์ท ซึ่งเป็นกฎสุดท้ายเมื่อเงื่อนไขแพ็กเก็ตที่ผ่านเข้ามาไม่ตรงกับเงื่อนไขใด ๆ เลย ให้เป็นยอมรับ คือ ยอมให้แพ็กเก็ตนั้นผ่านเข้ามายังตัวไฟร์วอลได้ เพื่อให้ง่ายต่อการกำหนดกฎ จึงแบ่งการกฎการส่งต่อ เป็น 2 ชนิด คือ ชนิดที่ส่งจากเครือข่ายภายนอกไปยังเครือข่ายภายใน ซึ่งในที่นี้จะตั้งชื่อกฎนี้ว่า EXTRANET-KMITL และชนิดที่ส่งจากเครือข่ายภายในไปยังเครือข่ายภายนอก ซึ่งในที่นี้จะตั้งชื่อกฎนี้ว่า EXTRANET-KMITL

FORWARD Chain (บอกถึงแพ็กเก็ตส่งต่อ)

ตารางที่ 3.3 บอกถึงแพ็กเก็ตส่งต่อ

Input Interface	Output Interface	Protocol	Match	Source	Destination	Port	Target

Eth0	Eth1	Any	-	Any	Any	Any	EXTRANET-KMITL
Eth1	Eth0	Any	-	Any	Any	Any	KMITL-EXTRANET

แพ็คเก็ตที่บอกถึงแพ็กเก็ตส่งต่อ แสดงดังรูป 3.3 ซึ่งมีการทำงานดังนี้

- บรรทัดที่หนึ่งเป็นการให้แพ็กเก็ตข้อมูลใด ๆ ที่มาจากอินเตอร์เฟสภายนอก (Output Interface) ที่เป็น Eth0 และจะส่งต่อไปยังอินเตอร์เฟสภายใน (Input Interface) ที่เป็น Eth1 กระโดดไปยัง chain ที่ชื่อว่า EXTRANET-KMITL
- บรรทัดที่สองเป็นการให้แพ็กเก็ตข้อมูลใด ๆ ที่มาจากอินเตอร์เฟสภายใน (Input Interface) ที่เป็น Eth1 และจะส่งต่อไปยัง อินเตอร์เฟสภายนอก (Output Interface) ที่เป็น Eth0 กระโดดไปยัง chain ที่ชื่อว่า KMITL-EXTRANET

EXTRANET-KMITL

ตารางที่ 3.4 บอกถึงกฎของการติดต่อจาก EXTERNET ไป KMITL

Input Interface	Output Interface	Protocol	Match	Source	Destination	Port	Target
-	-	-	State ESTABLISHED,RELATED	-	-	-	ACCEPT
-	-	TCP	-	Any	161.246.73.75	80	ACCEPT
-	-	TCP	-	Any	192.168.1.50	80	ACCEPT
-	-	Any	-	Any	Any		DROP

แพ็กเก็ตที่บอกถึงกฎของการติดต่อจาก EXTERNET ไป KMITL แสดงดังรูป 3.4 ซึ่งมีการทำงานดังนี้

- บรรทัดที่หนึ่งเป็นการอนุญาตให้ข้อมูลแพ็กเก็ตต่าง ๆ ที่ไม่ได้เกิดจากการเริ่มต้นการติดต่อจากเครือข่ายภายนอก สามารถผ่านเข้ามายังเครือข่ายภายในได้
- บรรทัดที่สองและบรรทัดที่สามเป็นการอนุญาตให้ข้อมูลแพ็กเก็ตใด ๆ ที่มีปลายทางเป็น 161.246.73.75 พอร์ท 80 สามารถผ่านเข้ามายังเครือข่ายภายใน เพื่อใช้งานเวบซิร์ฟเวอร์ได้ และอนุญาตให้ข้อมูลแพ็กเก็ตใด ๆ ที่มีปลายทางเป็น 192.168.1.50 พอร์ท 80 สามารถผ่านเข้ามายังเครือข่ายภายใน เพื่อใช้งานเว็บเซิร์ฟเวอร์ได้

- บรรทัดที่สี่ถ้าในกรณีที่แพ็กเก็ตที่กำลังจะเข้ามาไม่ตรงกับเงื่อนไขใด ๆ เลย ก็จะยกเลิกไม่ให้ข้อมูลนั้น ผ่านเข้ามายังเครือข่ายภายในได้ (DROP)

KMITL-EXTRANET

ตารางที่ 3.5 บอกถึงกฎของการติดต่อจาก KMITL ไป EXTRANET

Input Interface	Output Interface	Protocol	Match	Source	Destination	Port	Target
-	-	TCP	-	192.168.1.50	Any	Any	ACCEPT
-	-	UDP	-	192.168.1.50	Any	Any	ACCEPT
-	-	-	State ESTABLISHED,RELATED	Any	Any	Any	ACCEPT
-	-	Any	-	Any	Any	Any	DROP

แพ็คเก็ตที่บอกถึงกฎของการติดต่อจาก KMITL ไป EXTRANET แสดงดังรูป 3.5 ซึ่งมีการทำงานดังนี้

- บรรทัดที่หนึ่งและบรรทัดที่สอง เป็นการอนุญาตให้ข้อมูลแพ็กเก็ตต่างๆ ทั้งโพรโตคอล TCP และ UDP ที่มาจากแอดเดรสต้นทางคือ 192.168.1.50 สามารถติดต่อกับเครือข่ายภายนอกได้
- บรรทัดที่สามเป็นการอนุญาตให้ข้อมูลแพ็กเก็ตต่าง ๆ ที่ไม่ได้เกิดจากการเริ่มต้นการติดต่อจากเครือข่ายภายนอก สามารถผ่านเข้ามายังเครือข่ายภายในได้
- บรรทัดที่สี่ถ้าในกรณีที่แพ็กเก็ตที่กำลังจะเข้ามา ไม่ตรงกับเงื่อนไขใด ๆ เลย ก็จะยกเลิกไม่ให้ข้อมูลนั้น ผ่านเข้ามายังเครือข่ายภายนอกได้ (DROP)

ตาราง NAT

ตารางที่ 3.6 เป็นการสร้างกฎต่างๆของการทำ NAT

CHAIN	Input Interface	Output Interface	Protocol	Source	Destination	Port	To Address	Target
PREROUTING	Eth0	Eth1	-	161.246.73.75	-	80	192.168.1.50	-
PREROUTING	Eth0	-	Any	10.0.0.0/8	Any	Any	-	DROP
PREROUTING	Eth0	-	Any	127.0.0.0/8	Any	Any	-	DROP
PREROUTING	Eth0	-	Any	172.16.0.0/12	Any	Any	-	DROP
PREROUTING	Eth0	-	Any	192.168.1.0/16	Any	Any	-	DROP
POSTROUTING	Eth1	Eth0	Any	192.168.1.0/24	Any	Any	161.246.73.77-161.246.73.80	-

แพ็กเก็ตที่บอกถึงการสร้างกฎต่างๆของการทำNAT แสดงดังรูป 3.6 ซึ่งมีการทำงานดังนี้

- ในตาราง NAT ประกอบด้วย PREROUTING CHAIN และ POSTROUTING CHAIN
- กำหนดให้ PREROUTING CHAIN เป็นการส่งแพ็กเก็ตข้อมูล จากอินเตอร์เฟสภายนอกที่เป็น Eth0 และจะส่งต่อไปยังอินเตอร์เฟสภายในที่เป็น Eth1
- และ POSTROUTING CHAIN เป็นการส่งแพ็กเก็ตข้อมูล จาก อินเตอร์เฟสภายในที่เป็น Eth1 ส่งต่อไปยัง อินเตอร์เฟสภายนอกที่เป็น Eth0
- มีการทำ static NAT จากแอดเดรสต้นทาง 161.246.73.75 แปลงเป็นแอดเดรส 192.168.1.50 พอร์ท 80 เพื่อใช้งานเว็บเซิรฟ์เวอร์
- และมีการป้องกันการปลอม IP (IP spoofing) ใน PREROUTING CHAIN คือ ถ้ามีแพ็กเก็ตข้อมูลมาจากอินพุตอินเตอร์เฟสและมีแอดเดรสเป็น 10.0.0.0/8 127.0.0.0/8 172.16.0.0/12 192.168.1.0/16 ซึ่งเป็น private IP ให้ทำการยกเลิกข้อมูลนั้น

ในโครงงานนี้ได้ใช้คอมพิวเตอร์ส่วนบุคคลทำหน้าที่เป็นไฟร์วอลภายนอก โดยมีสเปคเครื่องคอมพิวเตอร์ดังนี้

- ซีพียู : Pentium4 3.0GHz
- แรม : 512 MHz
- ฮาร์ดดิสก์ : 40GHz

เทคโนโลยีที่เลือกใช้

- ใช้ระบบปฏิบัติการลินุกซ์เนื่องจาก เป็นระบบปฏิบัติการ โอเพ่นซอร์ส (Opensource) ที่มีเสถียรภาพสูง มีฟังก์ชันการทำงาน ที่ทำหน้าที่เป็นไฟร์วอลด้วย
- ใช้โปรแกรม IPtables ที่เป็นโอเพ่นซอร์ส ในการกำหนดนโยบายต่าง ๆ (Policy)
- InterScan Viruswall (ISVW) ทำหน้าที่ป้องกันไวรัสโดยตรวจสอบที่ HTTP, SMTP และ FTP Service
- โปรแกรม MRTG (Multi Router Traffic Grapher) เพื่อช่วยงานในส่วนของการมอนิเตอร์และรวบรวมข้อมูลจากระบบเครือข่าย

โปรแกรม MRTG (Multi Router Traffic Grapher) ซึ่งสามารถมอนิเตอร์ระบบเครือข่าย นำข้อมูลมารวบรวมไว้ และนำเสนอเป็นรูปกราฟผ่าน เว็บเพจ โดยผู้จัดการระบบสามารถมอนิเตอร์ข้อมูลผ่านโปรแกรมเว็บบราวเซอร์ได้ทันที แต่คุณสมบัติเหล่านี้ไม่ได้เกิดจากตัวโปรแกรมMRTG

แต่เพียงลำพัง ยังต้องอาศัยโปรแกรมอื่น ๆ เข้ามาสนับสนุนอีกจึงจะสามารถทำงานที่กล่าวไว้ได้ครบถ้วนสมบูรณ์ ซึ่งมีลำดับการทำงานดังนี้

อันดับแรก การที่จะรวบรวมข้อมูลต่าง ๆ จากระบบเครือข่ายมาได้ จำเป็นต้องอาศัยเครื่องมืออะไรก็ได้ที่ทำหน้าที่เป็นตัวแทนของเรา หรือที่ เรียกว่า Agent เฝ้าจับตาดูความเปลี่ยนแปลงของระบบเครือข่าย และส่งข้อมูลออกมาให้ทราบ ซึ่งโดยปกติจะอาศัยโปรโตคอล SNMP (Simple Network Management Protocol) ซึ่งเป็นคุณสมบัติหนึ่งภายในอุปกรณ์ Router หรือ Switches ทำหน้าที่เป็น Network Management Server

ต่อมาตัวโปรแกรม MRTG จะอ่านข้อมูลผ่าน SNMP Agent ตามระยะเวลาการสุ่มข้อมูลที่กำหนดไว้ แล้วพล็อตกราฟเป็นไฟล์รูปภาพ เก็บไว้ที่ไดเร็คทอรี่ที่กำหนดไว้

สุดท้ายโปรแกรมที่ทำหน้าที่เป็นเว็บเซิร์ฟเวอร์จะนำเสนอข้อมูลกราฟที่สร้างจาก MRTG ผ่านทางโปรโตคอล HTTP ทำให้สามารถดูกราฟแสดงรายงานการใช้แบนด์วิธได้จากโปรแกรมเว็บบราวเซอร์

3.3 การออกแบบเว็บเซิร์ฟเวอร์, DNS เซิร์ฟเวอร์ และ เมล์เซิร์ฟเวอร์

เว็บเซิร์ฟเวอร์, DNS เซิร์ฟเวอร์ และ เมลเซิร์ฟเวอร์ อยู่ในส่วนของเครือข่ายภายในที่มีการทำ Static NAT ผ่านทางไฟร์วอลภายนอก

1. เว็บเซิร์ฟเวอร์

เว็บเซิร์ฟเวอร์ สร้างขึ้นสำหรับบริการการใช้งานต่าง ๆ ให้กับอาจารย์และนักศึกษาภายในห้องวิจัยแบบไร้สาย เช่น โฮมเพจสร้างขึ้นเพื่อให้อาจารย์และนักศึกษาภายในห้องวิจัยสามารถค้นหาหรือดาวน์โหลดข้อมูลเกี่ยวกับ Ultra Wideband ได้ง่ายและสะดวกยิ่งขึ้น ทั้งนี้ยังได้จัดทำเว็บบอร์ดขึ้นมาเพื่อให้นักศึกษาที่มีปัญหาหรือคำถามที่เกี่ยวกับ Ultra Wideband ได้มาตั้งกระทู้คำถามไว้บนเว็บบอร์ดได้

2. DNS เซิร์ฟเวอร์

DNS เป็นระบบแปลงหมายเลขไอพีแอดเดรสให้อยู่ในรูปแบบของ โดเมนเนม หรือ แปลงกลับจากโดเมนเนมไปเป็นไอพีแอดเดรสได้ หลังจากที่ได้ลงทะเบียนชื่อโดเมนผ่านISPเรียบร้อยแล้ว ในที่นี้ได้ทำการประกาศสับโดเมน (Subdomain) ชื่อ wis.ite.kmitl.ac.th

3. เมล์เซิร์ฟเวอร์

เมล์เซิร์ฟเวอร์ มีหน้าที่ในการจัดการรับส่งอีเมล์ของผู้ใช้ภายในห้องวิจัย ทำให้การใช้งานอีเมล์มีความยืดหยุ่นขึ้น โดยมีแอดเดรสเป็น user@wis.ite.kmitl.ac.th

ในโครงงานนี้ได้ใช้คอมพิวเตอร์ส่วนบุคคล ทำหน้าที่เป็นเซิร์ฟเวอร์โดยมีสเปคเครื่องดังนี้

- ซีพียู : Pentium4 3.0GHz
- แรม : 512 MHz
- ฮาร์ดดิสก์ : 80GHz

เทคโนโลยีที่เลือกใช้

- ใช้ระบบปฏิบัติการลินุกซ์ เนื่องจากเป็นระบบปฏิบัติการ โอเพ่นซอร์สที่มีระบบปฏิบัติการลินุกซ์ เนื่องจากเป็นระบบปฏิบัติการโอเพ่นซอร์สที่มีเสถียรภาพสูง มีฟังก์ชันการทำงาน ที่ทำหน้าที่เป็นไฟร์วอลได้
- Apache Web Server เป็นเว็บเซิร์ฟเวอร์เพราะเป็นโอเพ่นซอร์สและใช้งานง่าย
- PHP เป็น Script สำหรับเขียน *.php
- MySQL สำหรับจัดการฐานข้อมูล
- PHPMyAdmin สำหรับจัดการระหว่าง PHP และ MySQL
- ใช้ Bind เป็น DNS เซิร์ฟเวอร์ เนื่องจากเป็น โอเพ่นซอร์ส
- ใช้ โปรแกรม Postfix ในการจัดการบริการ SMTP
- ใช้ โปรแกรม ClamAV เป็น Antivirus
- ใช้โปรแกรม Amavisd-New เป็น Antivirus Scanner สำหรับเป็นตัวกลางระหว่าง Postfix และ ClamAV
- ใช้โปรแกรม Squirrel Mail ในการบริหารจัดการเว็บเบสสำหรับติดต่อกับผู้ใช้เมลล์ผ่านเว็บ

3.4 การออกแบบมอนิเตอร์เซิร์ฟเวอร์

เป็นเซิร์ฟเวอร์ที่ทำหน้าที่ดูแลจัดการคอมพิวเตอร์ของผู้ใช้ภายในเครือข่าย ทำหน้าที่เป็น DHCP เซิร์ฟเวอร์ สำหรับ แจกไอพีให้แก่เครื่องไคลเอนต์แบบอัตโนมัติและมีโปรแกรมตรวจจับไวรัสในเครือข่ายภายใน สามารถแจ้งเตือนแก่ผู้ใช้ในระบบ มีการเก็บล็อกข้อมูลต่าง ๆ ของการใช้งานระบบเครือข่ายของบุคลากร เพื่อป้องกันและตรวจสอบข้อผิดพลาดการใช้งาน

ในโครงงานนี้ได้ใช้คอมพิวเตอร์ส่วนบุคคล ทำหน้าที่เป็นมอนิเตอร์เซิร์ฟเวอร์โดยมีสเปคเครื่องคอมพิวเตอร์ดังนี้

- ซีพียู : Pentium4 3.0GHz
- แรม : 512 MHz
- ฮาร์ดดิสก์ : 40GHz

เทคโนโลยีที่เลือกใช้

- ใช้ระบบปฏิบัติการ Windows Server 2003 เนื่องจาก เป็นระบบปฏิบัติการที่สามารถติดต่อกับผู้ใช้ได้สะดวก และมีโปรแกรมสนับสนุนมากมาย
- ใช้โปรแกรม Trend Micro ในการตรวจจับไวรัสในเครือข่าย

OfficeScan ทำหน้าที่ป้องกันไวรัสบนเครือข่าย โดยผู้บริหารสามารถจัดการเครื่องลูกข่ายต่าง ๆ เหล่านี้ได้จากจุดศูนย์กลาง เพื่อง่ายต่อการจัดการ

3.5 การออกแบบเซิร์ฟเวอร์ข้อมูล

เป็นเซิร์ฟเวอร์ที่ทำหน้าที่เก็บข้อมูลต่างๆหรือ Thesis สำหรับอาจารย์ในห้องวิจัย ซึ่งมีความสำคัญระดับสูง

ในโครงงานนี้ได้ใช้คอมพิวเตอร์ส่วนบุคคล ทำหน้าที่เป็นเซิร์ฟเวอร์ฐานข้อมูลโดยมีสเปคเครื่องคอมพิวเตอร์ดังนี้

- ซีพียู : Pentium4 3.0GHz
- แรม : 512 MHz
- ฮาร์ดดิสก์ : 40GHz

เทคโนโลยีที่เลือกใช้

- ระบบปฏิบัติการลินุกซ์เนื่องจากเป็นระบบปฏิบัติการโอเพ่นซอร์สที่มีเสถียรภาพสูง
- VSFTP (Very Secure FTP) สำหรับจัดการด้าน FTP ที่มีความปลอดภัยสูง

3.6 การออกแบบโฮมเพจ

เนื่องจากภายในห้องวิจัย Wireless Communication Research Group ต้องการโฮมเพจเพื่อให้บุคคลภายในและภายนอกสามารถมาหาความรู้เกี่ยวกับ Ultra Wide Band ได้อย่างสะดวก

ดังนั้นในโครงงานนี้จึงได้คิดสร้างโฮมเพจสำหรับห้องวิจัย Wireless Communication Research Group ขึ้นมา โดยในโฮมเพจประกอบด้วย

- Member (สมาชิก) จะบอกว่ามีบุคคลากรท่านไหนบ้างที่เป็นสมาชิกอยู่ในห้องวิจัยนี้
- Publiclist (ผลงานวิจัย) จะบอกถึงบทความวิจัยที่ตีพิมพ์ในวารสารในประเทศและต่างประเทศ
- Links (ลิ้งค์) ซึ่งภายในจะมี links ไปยังที่สำคัญต่างๆที่มีประโยชน์สำหรับห้องวิจัย
- Webmail (เว็บเมล์) ซึ่งจะ links ไปยัง Webmail ที่โครงงานนี้สร้างขึ้นมาเพื่อบริการเกี่ยวกับการรับส่งอีเมล์

นอกจากนี้ยังได้สร้างหน้าล็อกอินเพื่อให้ผู้ที่เป็นสมาชิกของห้องวิจัยเท่านั้นที่สามารถเข้าไปยังหน้าที่สามารถตั้งเว็บบอร์ดและดาวน์โหลดผลงานวิจัยต่างๆได้ โดยหน้าล็อกอินจะให้กรอกชื่อและ password ถ้าบุคคลใดยังไม่มีให้ทำการลงทะเบียนใหม่ (New Register) โดยทำการกรอกตามแบบฟอร์มที่สร้างขึ้นเพื่อกำหนด user name และ password แล้วค่อยนำมาใช้ในล็อกอิน ซึ่งเมื่อล็อกอินเสร็จแล้วสามารถตั้งกระทู้ต่างๆได้ โดยอยู่ในการดูแลของผู้ดูแลระบบ ถ้ามีข้อความไม่เหมาะสม ผู้ดูแลระบบเท่านั้นที่สามารถลบกระทู้นี้ได้ หรือจะล็อกอินเข้ามาเพื่อดาวน์โหลดผลงานวิจัยต่างๆมาศึกษาได้

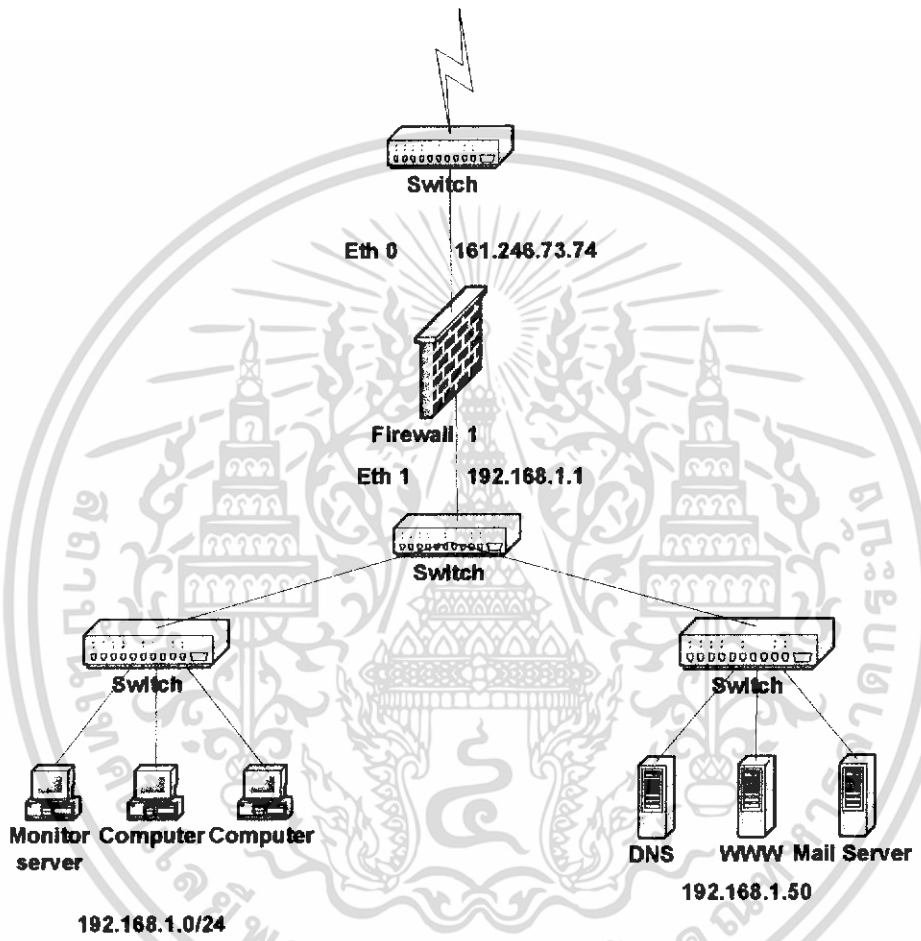
และเมื่อทำการสร้างโฮมเพจเสร็จเรียบร้อยแล้วจะนำไปลงที่ Web Server โดยใช้ Apache Web Server เนื่องจากเป็นเว็บเซิร์ฟเวอร์ที่มีประสิทธิภาพสูง และนิยมใช้มากที่สุด นอกจากนี้ยังใช้ Script language คือ PHP เพราะเป็นโอเพ่นซอร์สและสามารถติดต่อกับฐานข้อมูลที่ใช้ในโครงงานได้ (MySQL)

บทที่ 4

ผลการทดลอง

4.1 การจัดการระบบเครือข่ายภายใน

ได้ทำการสร้างเครือข่ายหลักเพียงบางส่วนของการออกแบบทั้งหมด ซึ่งได้แก่



รูปที่ 4.1 ระบบเน็ตเวิร์ค

จากรูปได้ทำการออกแบบระบบเน็ตเวิร์ค โดยมีรายละเอียดดังนี้

4.1.1 ไฟร์วอล

ประกอบด้วย 2 อินเทอร์เฟสสำหรับใช้ในการติดต่อ 2 อัน คือ Eth0 ซึ่งจะมีไอพีแอดเดรส เป็น 161.246.73.74 netmask เป็น 255.255.255.0 เพื่อติดต่อกับเครือข่ายภายนอก และ Eth1 จะมีไอพีแอดเดรส เป็น 192.168.1.1 netmaskเป็น 255.255.255.0 เพื่อติดต่อกับเครือข่ายภายใน

โดยมีหน้าที่ คือ

- เป็นเราต์เตอร์ในการหาเส้นทาง
- กำหนดกฎการเข้า-ออก ของแพ็กเก็ตของข้อมูล
- NAT สำหรับ แปลงไอพีแอดเดรสของเครือข่ายภายใน ให้เป็นไอพีแอดเดรส ซึ่งเป็นที่ยอมรับที่สามารถสื่อสารบนอินเทอร์เนตได้

4.1.2 เว็บเซิร์ฟเวอร์ , DNS เซิร์ฟเวอร์ และ เมล์เซิร์ฟเวอร์

มีไอพีแอดเดรส เป็น 192.168.1.50 netmask เป็น 255.255.255.0 จะทำการแปลงไอพีแอดเดรส (static NAT) จาก 161.246.73.75 เป็น 192.168.1.50 ก่อนที่เครือข่ายภายนอกจะเข้าสู่เว็บเซิร์ฟเวอร์ได้

โดยมีหน้าที่ คือ

- เว็บเซิร์ฟเวอร์ สำหรับโฮมเพจ และ เว็บแอพพลิเคชั่น เพื่อบริการผู้ใช้
- DNSเซิร์ฟเวอร์ สำหรับแปลงหมายเลขไอพีแอดเดรสให้อยู่ในรูปของโดเมนเนม หรือ แปลงกลับจากโดเมนเนมไปเป็นไอพีแอดเดรส และประกาศโดเมนลูกของห้องวิจัยเป็น “wis.ite.kmitl.ac.th”
- เมล์เซิร์ฟเวอร์ สำหรับจัดการรับส่งอีเมล์สำหรับผู้ใช้ในเครือข่าย

4.1.3 มอนิเตอร์เซิร์ฟเวอร์

มีไอพีแอดเดรสเป็น 192.168.1.51 netmask เป็น 255.255.255.0

โดยมีหน้าที่ คือ

- ดูแลจัดการคอมพิวเตอร์ของผู้ใช้ภายในเครือข่าย
- มีโปรแกรมตรวจจับไวรัสในเครือข่ายภายใน สามารถแจ้งเตือนแก่ผู้ใช้ในระบบ
- มีการเก็บล็อกข้อมูลต่าง ๆ ของการใช้งานระบบเครือข่ายของผู้ใช้
- DHCP เซิร์ฟเวอร์ สำหรับแจกไอพีให้แก่เครื่องไคลเอ็นต์แบบอัตโนมัติ

4.1.4 เซิร์ฟเวอร์ข้อมูล

มีไอพีแอดเดรสเป็น 192.168.1.52 netmask เป็น 255.255.255.0

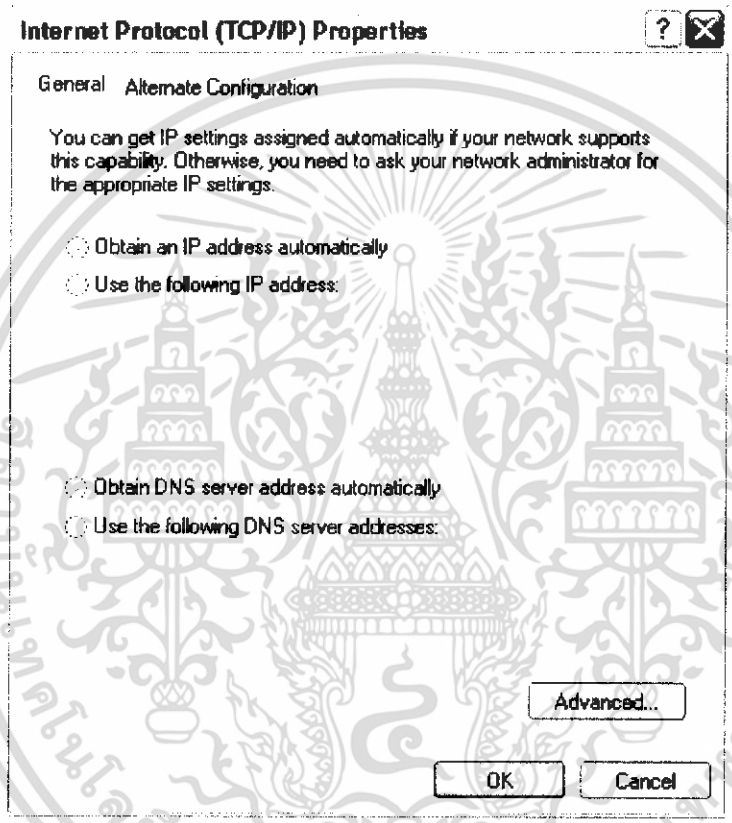
โดยมีหน้าที่ คือ

- เก็บข้อมูลของผู้ใช้ภายในเครือข่าย ที่มีความปลอดภัยสูง

4.2 ผลการทดลองระบบเครือข่าย

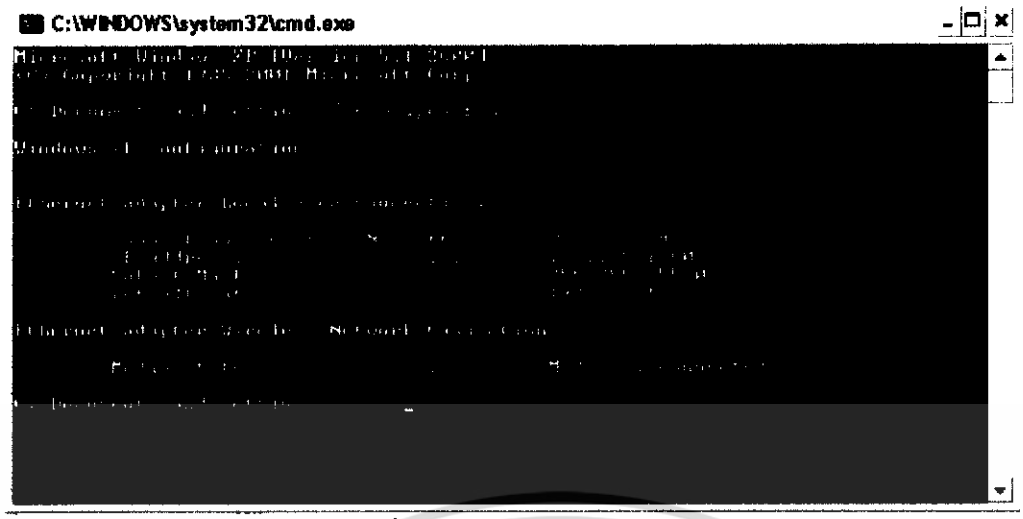
4.2.1 DHCP เซิร์ฟเวอร์

การใช้งาน DHCP



รูปที่ 4.2 ผู้ใช้รอรับไอพีแอดเดรสอัตโนมัติ

จากรูปที่ 4.2 เป็นการให้ผู้ใช้ที่เครื่องไคลเอ็นต์เข้าไปที่ Internet Protocol (TCP/IP) Properties และเลือก Obtain an IP Address automatically เพื่อรับไอพีแอดเดรสอัตโนมัติจาก DHCPเซิร์ฟเวอร์



รูปที่ 4.3 ผู้ใช้ได้รับไอพีแอดเดรส

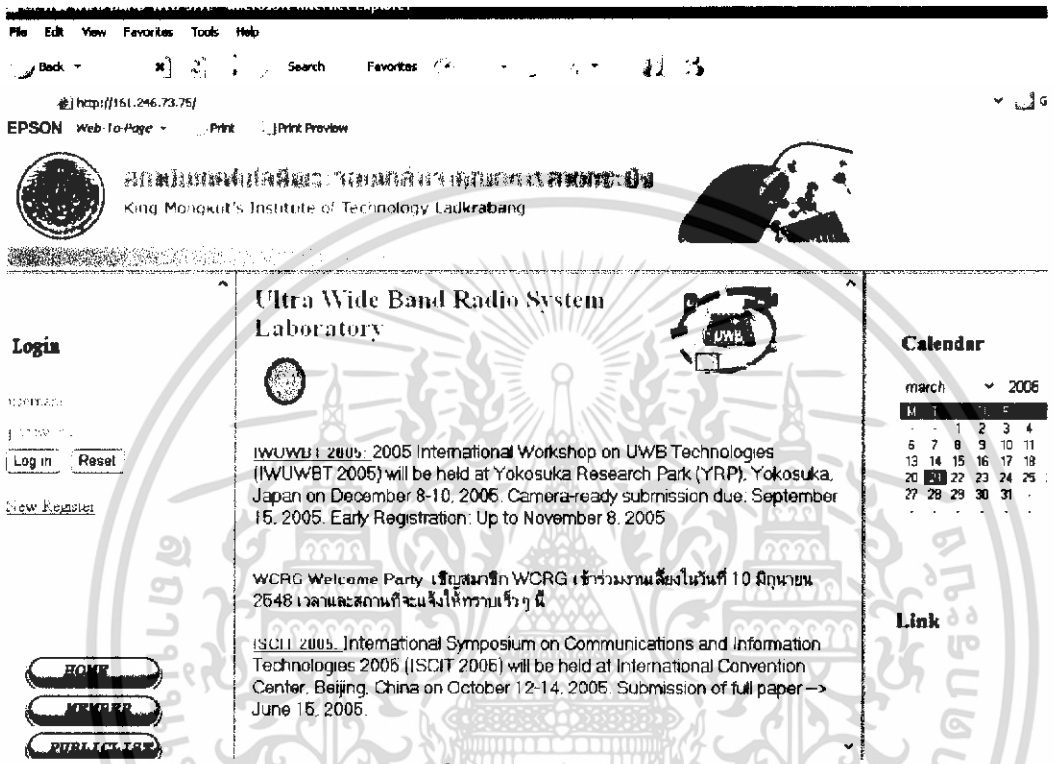
จากรูปที่ 4.3 เป็นการตรวจสอบหมายเลขไอพีแอดเดรสของเครื่องไคลเอ็นต์ที่ได้รับการแจกจาก DHCPเซิร์ฟเวอร์ เรียบร้อยแล้ว

4.2.2 NAT และ ไฟร์วอล

รูปที่ 4.4 การติดต่อกับเครือข่ายภายนอก โดยผ่าน NAT และ ไฟร์วอล

หลังจากได้ทำการสร้าง NAT และ ไฟร์วอล เพื่อให้ผู้ใช้ที่อยู่เครือข่ายภายใน สามารถติดต่อกับเครือข่ายภายนอกได้อย่างสะดวก ปลอดภัย และเครือข่ายภายนอกไม่สามารถมองเห็นเครือข่ายภายในได้ ซึ่งในกรณีนี้จะมีกฎการเข้าออกแพ็กเก็ตเปิดให้ติดต่อเฉพาะเว็บเซิร์ฟเวอร์เท่านั้น จากรูปที่4.4 ได้แสดงการติดต่อกับเครือข่ายภายนอกโดยผ่าน NAT และ ไฟร์วอล ที่สร้างขึ้น

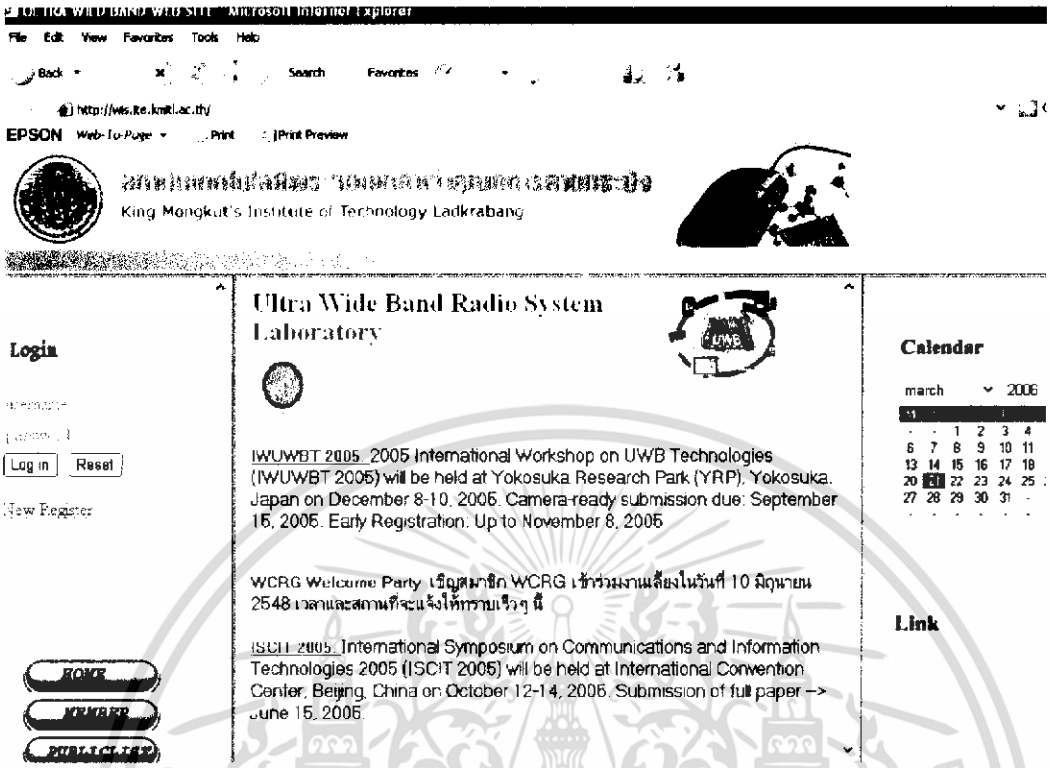
4.2.3 เว็บเซิร์ฟเวอร์



รูปที่ 4.5 เว็บเซิร์ฟเวอร์

จากรูป เป็นเว็บเซิร์ฟเวอร์ที่ภายในประกอบด้วยเนื้อหาเกี่ยวกับเรื่อง Ultra Wideband รวมถึงลิงค์ต่างๆที่เป็นประโยชน์เช่น เว็บไซต์ที่น่าสนใจสำหรับบุคคลที่สนใจเกี่ยวกับเรื่อง Ultra Wideband นี้ เป็นต้น

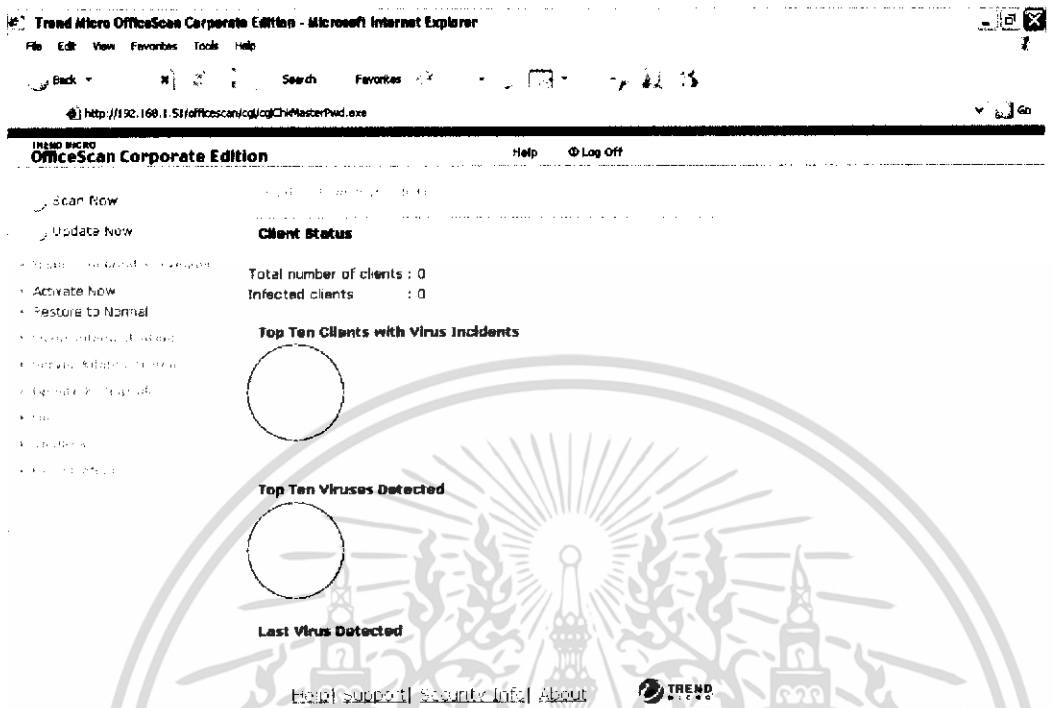
4.2.4 DNS เซิร์ฟเวอร์



รูปที่ 4.6 ประกาศโดเมนลูกเป็น wis.itc.kmitl.ac.th

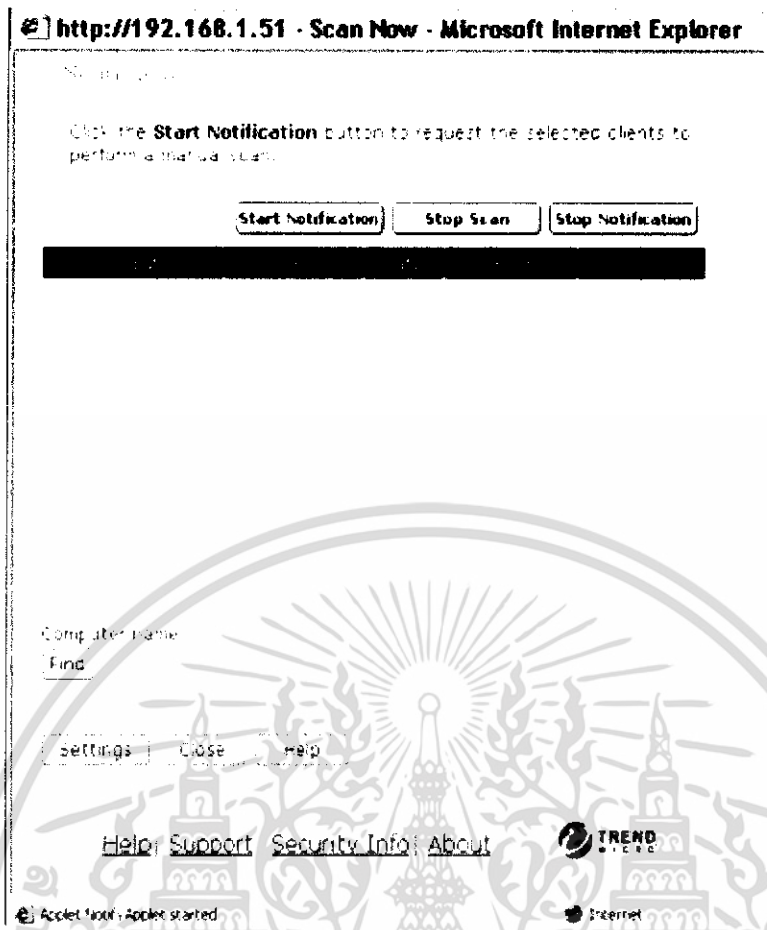
ในการทำ DNS เซิร์ฟเวอร์ ได้มีการประกาศโดเมนลูก ซึ่งจากรูปที่ 4.6 จะเห็นได้ว่าได้ประกาศโดเมนลูกให้เป็น wis.kmitl.ac.th ซึ่งทำขึ้นสำหรับห้องวิจัยแบบไร้สายโดยเฉพาะ โดยจะต้องไปทำการจดทะเบียนที่ศูนย์วิจัย

4.2.5 มอนิเตอร์เซิร์ฟเวอร์



รูปที่ 4.7 โปรแกรมในการตรวจจับไวรัสหรือความผิดปกติในเครือข่าย

จากรูปเป็นการจัดการดูแลเครื่องไคลเอ็นต์ ในการตรวจจับไวรัส หรือความผิดปกติภายในเครือข่าย โดยใช้โปรแกรม Trend Micro OfficeScan



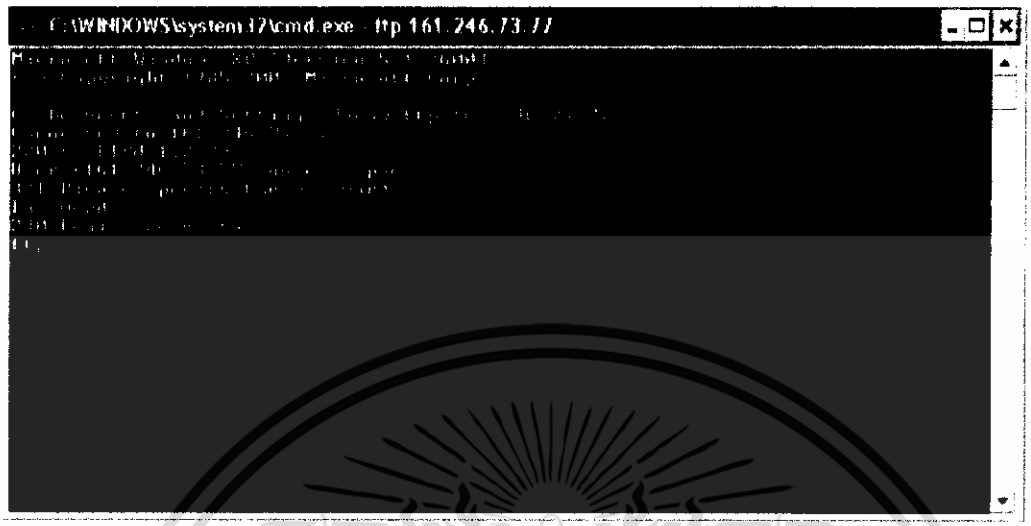
รูปที่ 4.8 แสดงการทำงานของโปรแกรม Trend Micro

จากรูปเป็นการแสดงการทำงานของโปรแกรม Trend Micro ตัวอย่างเช่น ในรูปนี้เป็นการสแกน (Scan) ตรวจจับไวรัสของเครื่องไคลเอ็นต์ที่มีไอพีแอดเดรส 192.168.1.51 ซึ่งอยู่ในเครือข่ายภายใน

4.2.6 FTP

FTP สามารถเข้าได้ 3 แบบคือ

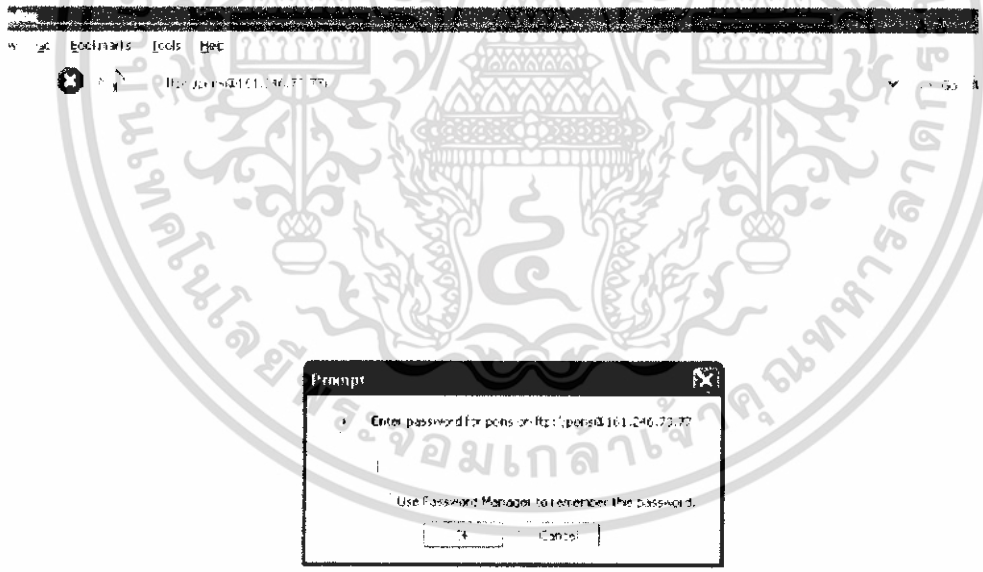
1. เข้าโดยผ่านทาง Dos



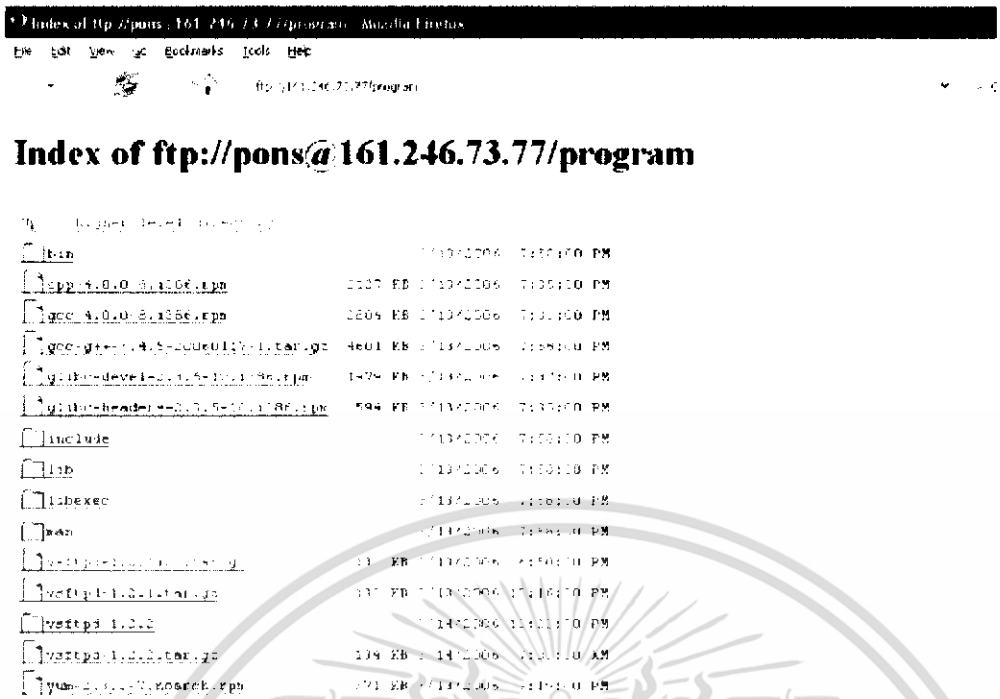
รูปที่ 4.9 การเข้า FTP ผ่านทาง Dos

จากรูปเป็นการแสดงการเข้า FTP ผ่านทาง Dos โดย พิมพ์ ftp ตามด้วยหมายเลขไอพี

2. เข้าโดยผ่านทางเว็บบราวเซอร์



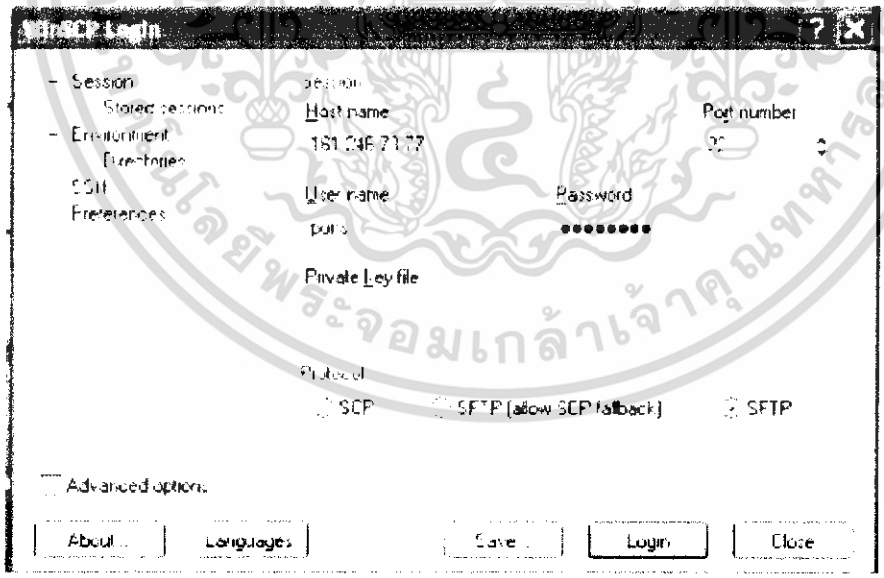
รูปที่ 4.10 การเข้าFTP ผ่านทางเว็บบราวเซอร์



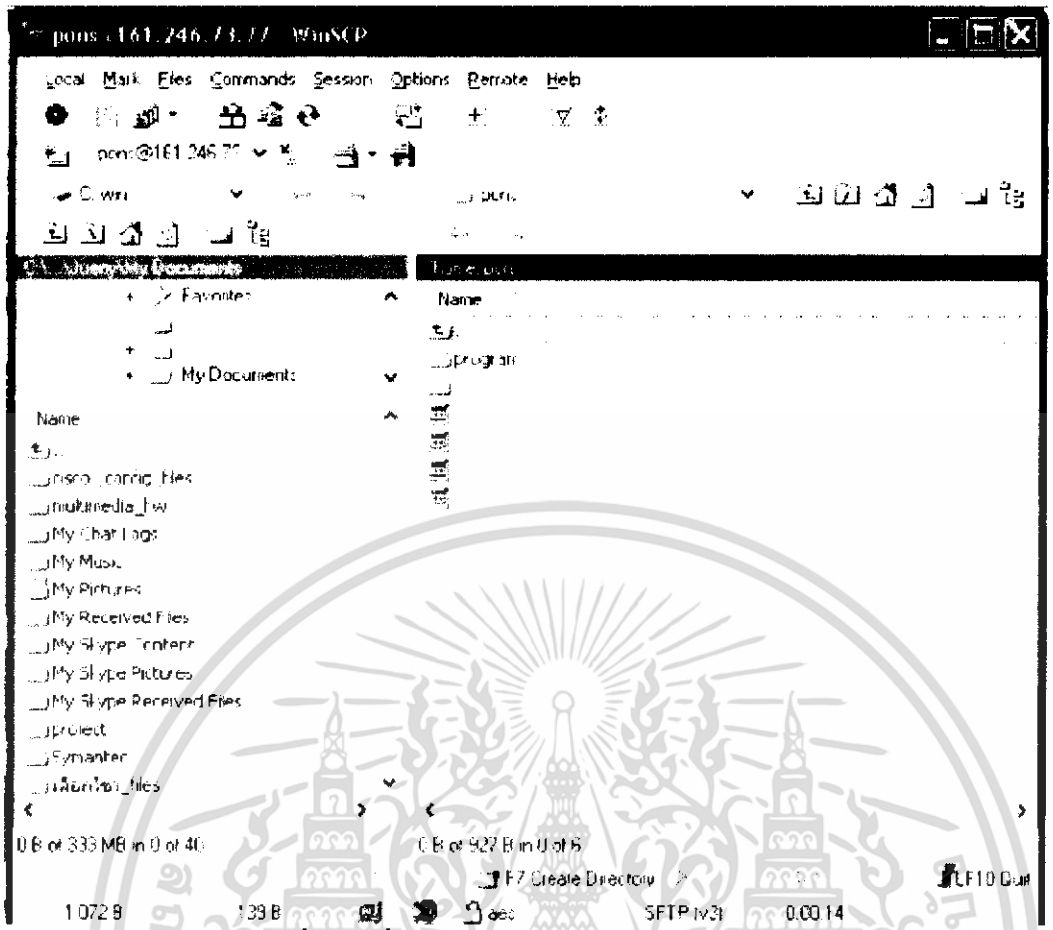
รูปที่ 4.11 เมื่อทำการใส่password ให้ถูกต้องตามที่กำหนด

จากรูปที่ 4.10 และ 4.11 เป็นการเข้า FTP โดยผ่านทางเว็บบราวเซอร์ ซึ่งเมื่อใส่ password ถูกตามที่กำหนดก็สามรถเข้ามาใช้งานได้

3. เข้าโดยผ่านทาง WinSCP



รูปที่ 4.12 การเข้าFTPผ่านทาง WinSCP

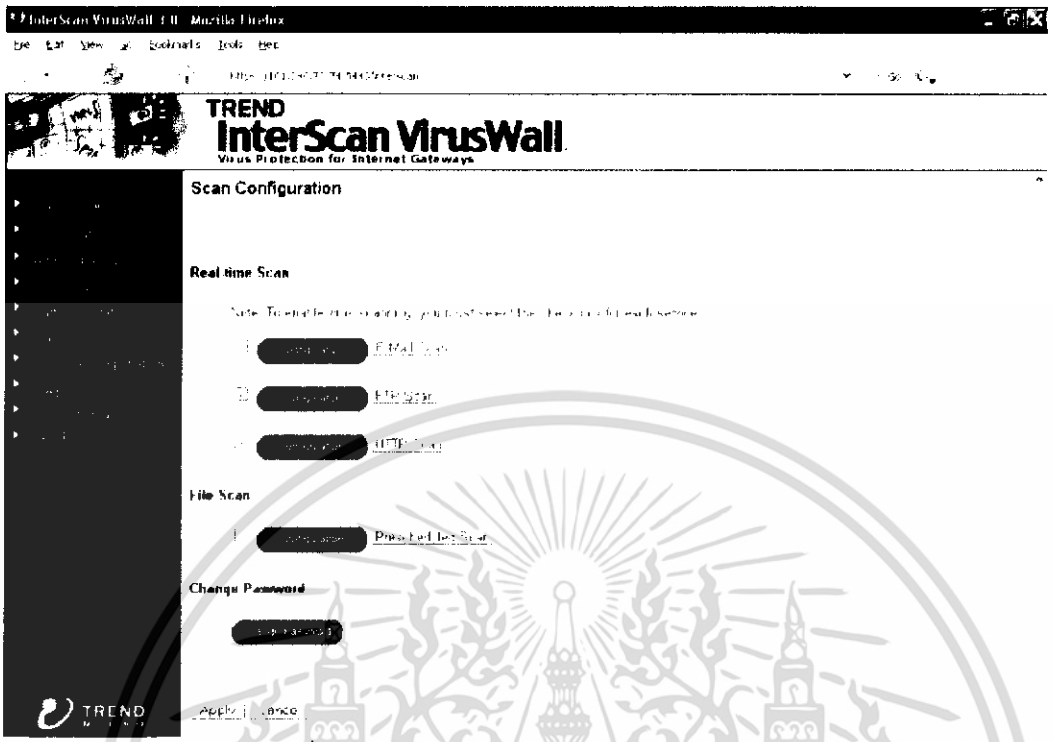


รูปที่ 4.13 เมื่อใส่ username และ password ถูกต้อง

จากรูป 4.12 และ 4.13 เป็นการใช้ FTP ผ่านทาง WinSCP ซึ่งจะให้ใส่ Host name, username และ password เมื่อใส่ค่าทั้งหมดถูกต้องก็สามารถใช้งานได้

4.2.7 Antivirus Gateway

เป็นตัวจับไวรัสโดยวางไว้ที่ไฟร์วอล

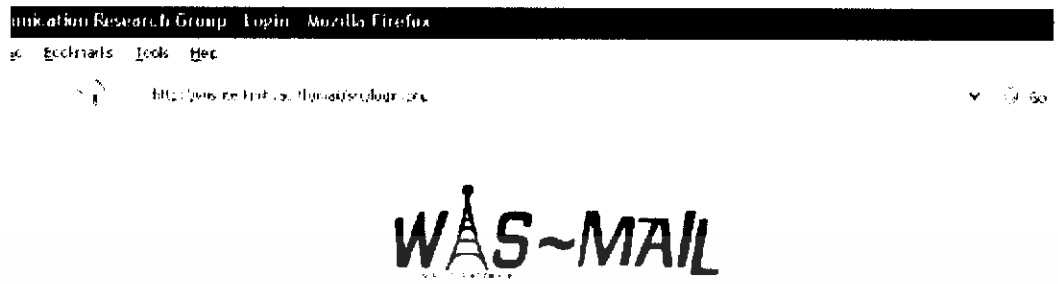


รูปที่ 4.14 โปรแกรม Trend InterScan VirusWall

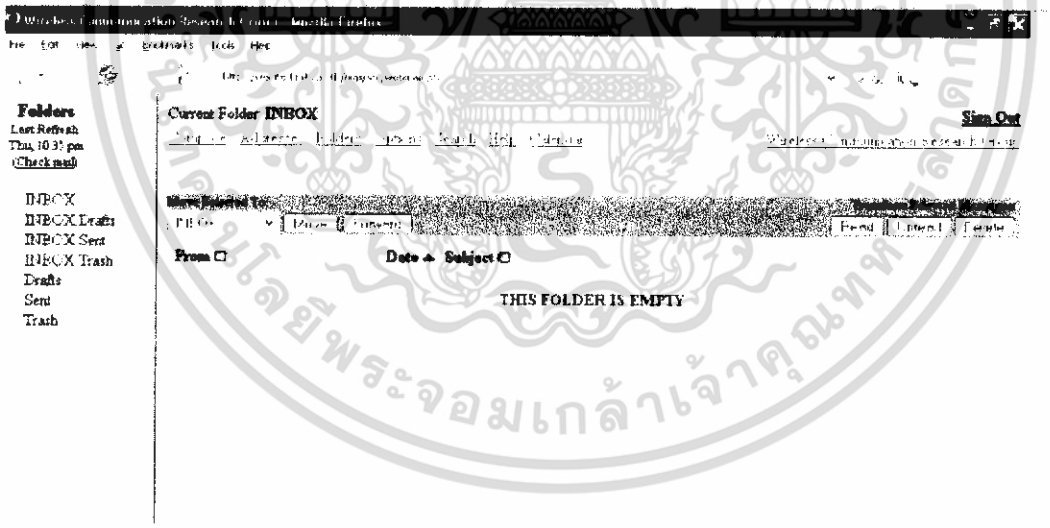
รูปที่ 4.15 เป็นการเก็บ log (ล็อก) ต่างๆของโปรแกรม

จากรูป 4.14 และ 4.15 เป็น Antivirus Gateway ที่วางอยู่บนไฟร์วอลโดยทำหน้าที่คอยดูเกี่ยวกับไวรัสและเก็บ log ต่างๆ เอาไว้

4.2.8 Mail Server

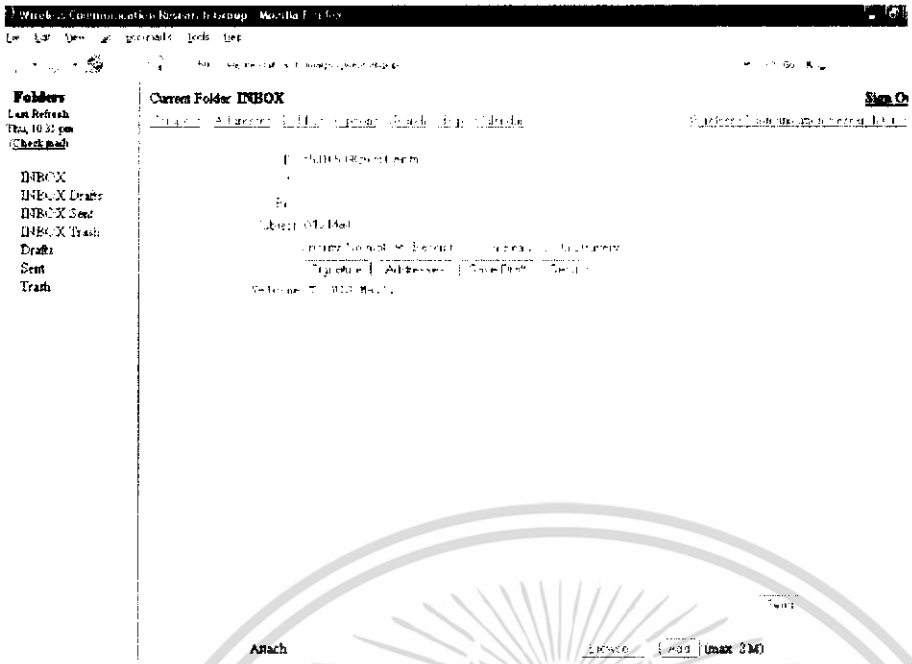


รูปที่ 4.16 เป็นการล็อกอินเข้าเมล์

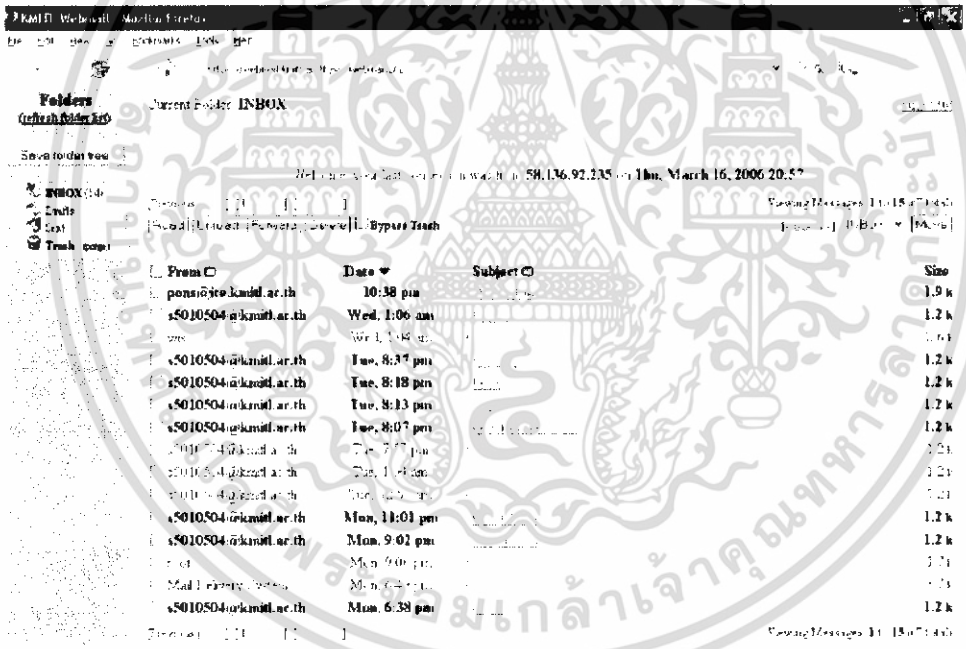


รูปที่ 4.17 เมื่อใส่ล็อกอินอย่างถูกต้อง

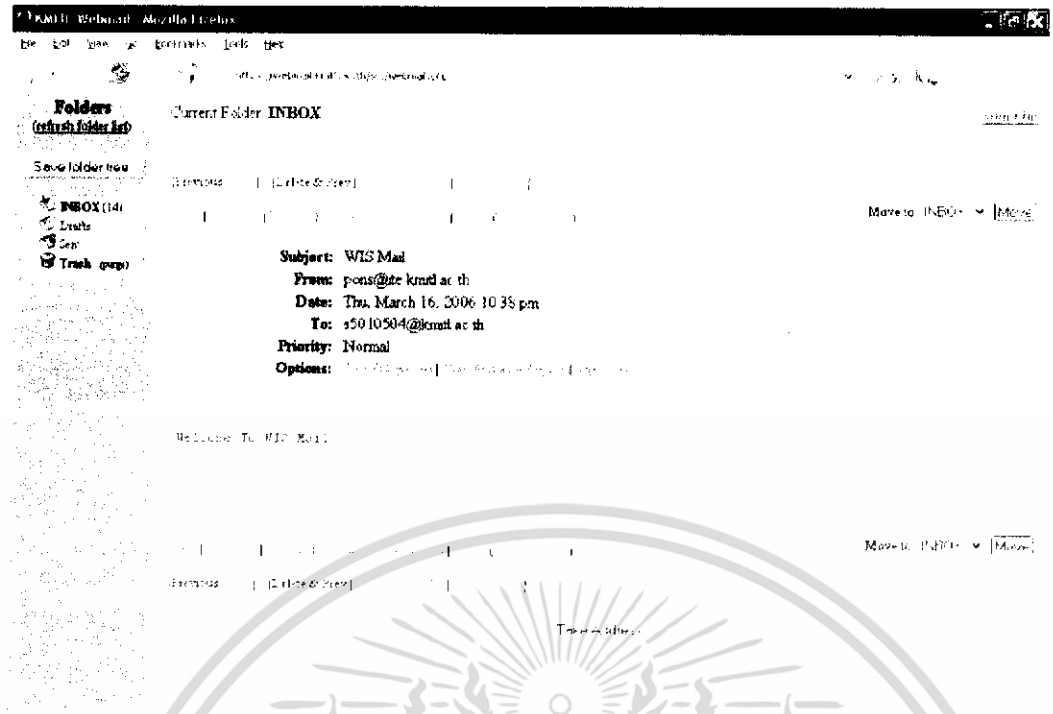
จากรูป 4.16 และ 4.17 เป็นการเข้ามลโดยเมื่อใส่รหัสอย่างถูกต้องก็สามารถเข้าใช้เมล์ของห้องวิจัยได้ โดยแต่ละคนจะมี username และ password



รูปที่ 4.18 เป็นการทดลองส่งเมล์ออกจาก wis.ite.kmitl.ac.th

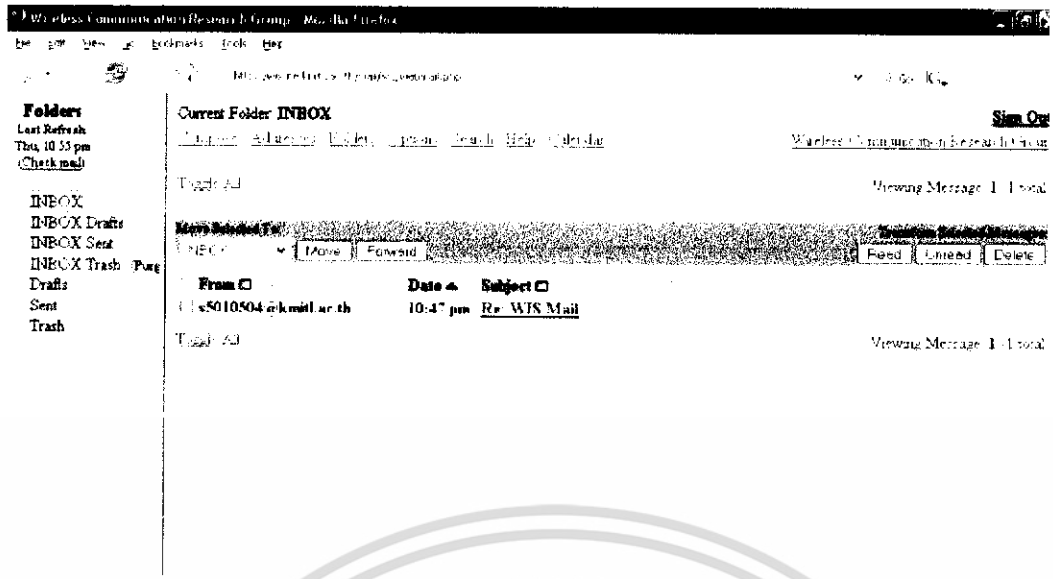


รูปที่ 4.19 ตรวจสอบการส่งเมล์

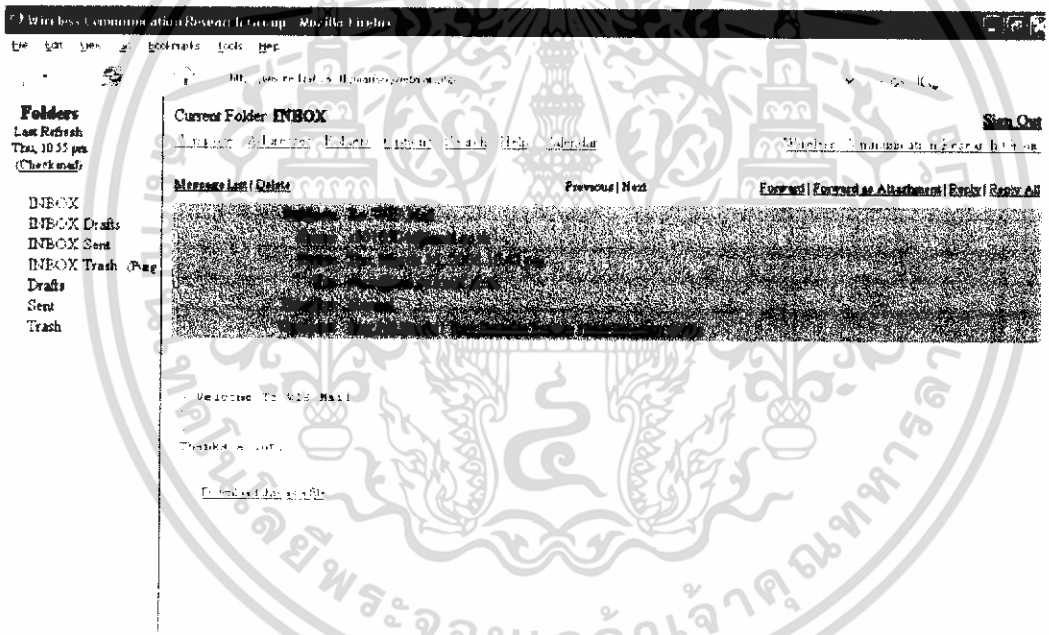


รูปที่ 4.20 ตรวจสอบการส่งเมล์

รูปที่ 4.21 ทดสอบโดย reply กับไปยังเมลล์เซิร์ฟเวอร์



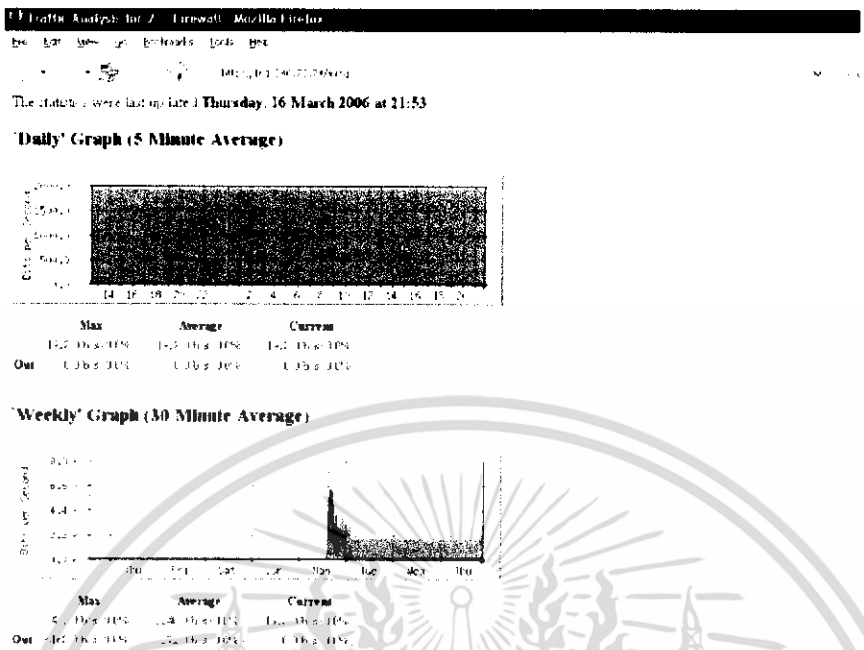
รูปที่ 4.22 ตรวจสอบการรับเมล์



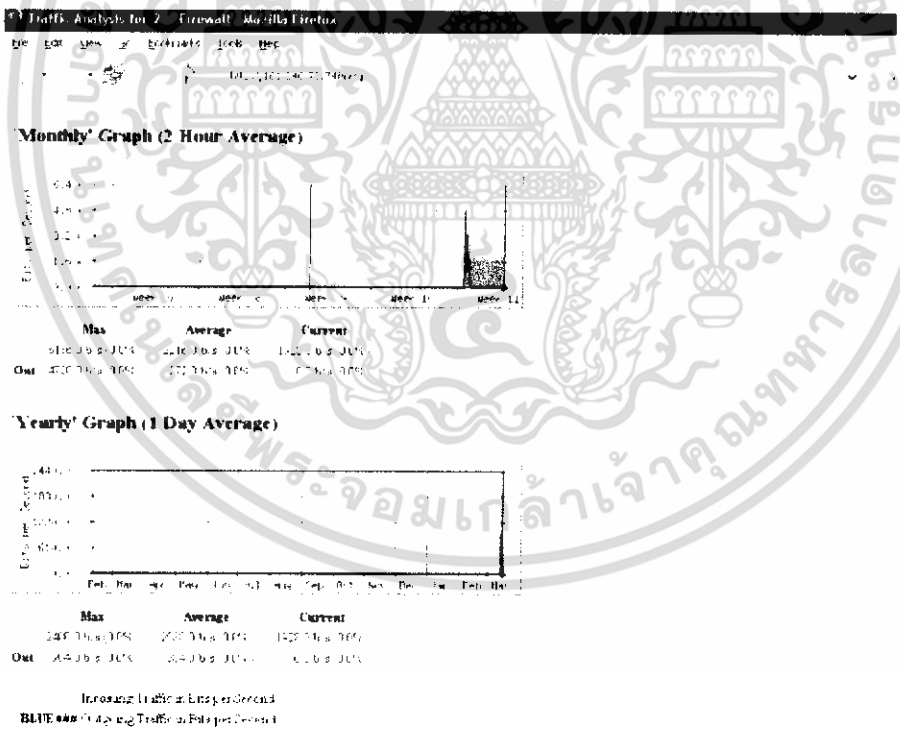
รูปที่ 4.23 ตรวจสอบการรับเมล์

4.2.9 โปรแกรมตรวจจับทราฟฟิก (MRTG)

MRTG เป็นโปรแกรมตรวจจับทราฟฟิกภายในเครือข่าย



รูปที่ 4.24 เป็นการดูทราฟฟิกภายในเครือข่ายแบบเป็นวันและอาทิตย์

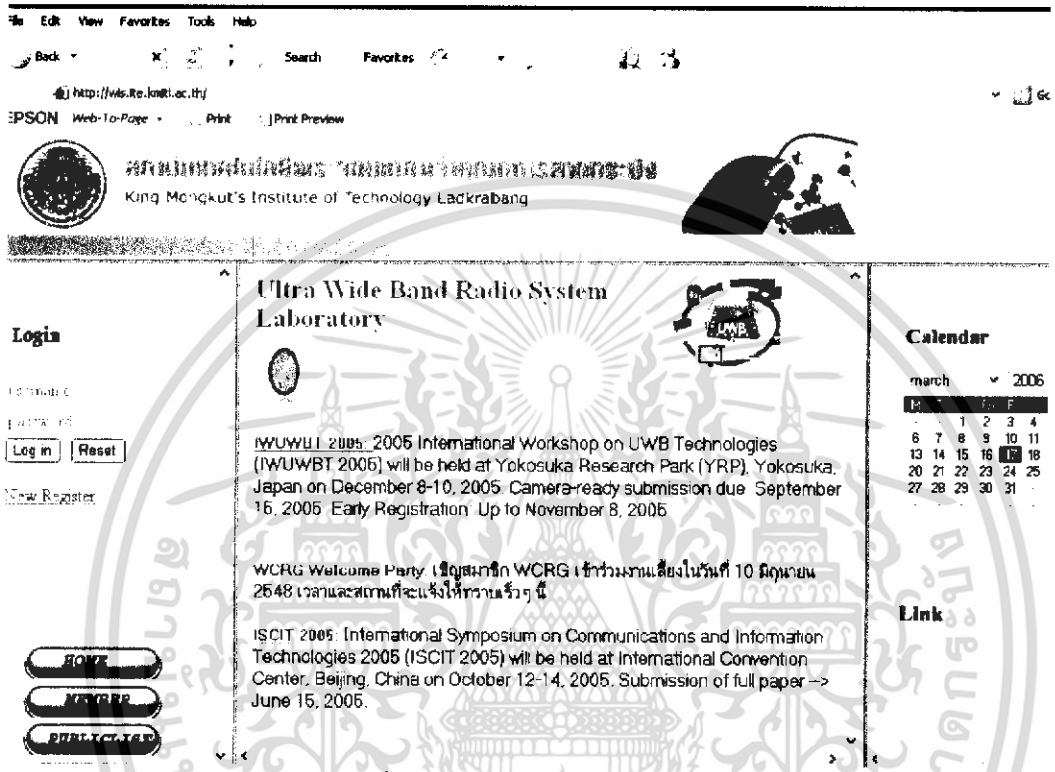


รูปที่ 4.25 เป็นการดูทราฟฟิกภายในเครือข่ายแบบเป็นเดือนและปี

รูปที่4.24 และ 4.25 เป็นการใช้ซอร์ฟแวร์ MRTG วางไว้ที่ไฟร์วอลเพื่อดูทราฟฟิกภายในระบบว่าเป็นอย่างไรซึ่งสามรถดูได้แบบทั้งเป็นวัน, อาทิตย์, เดือน, ปี ได้

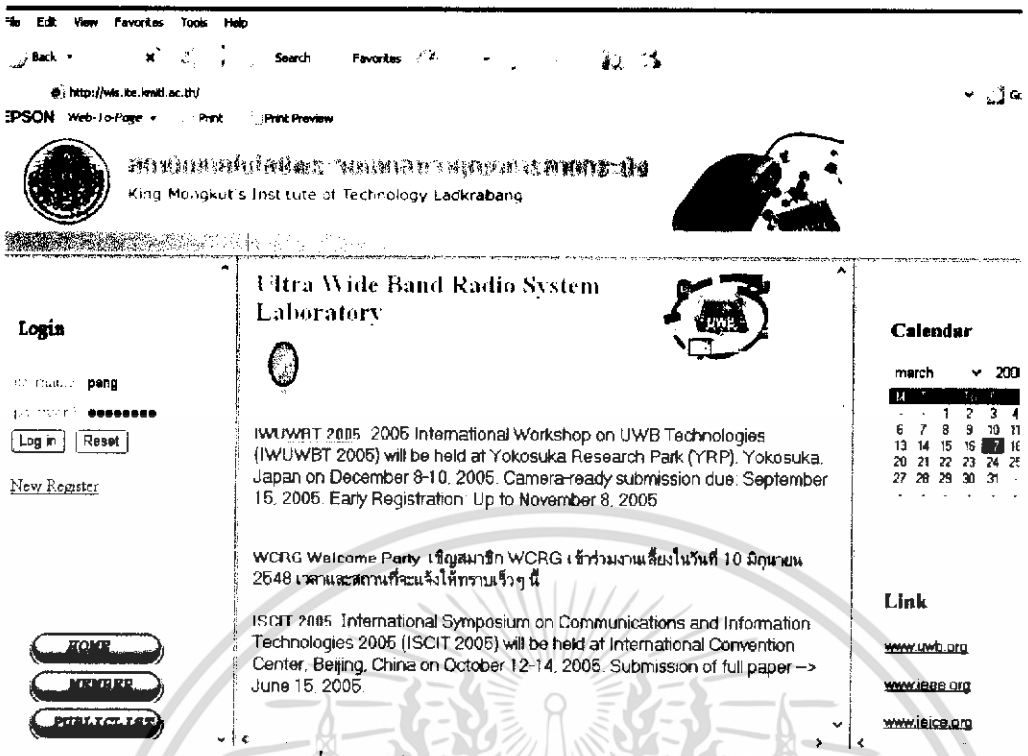
4.2.10 Homepage

ได้สร้าง Homepage ขึ้นมาเพื่อบริการบุคคลากรภายในห้องวิจัย

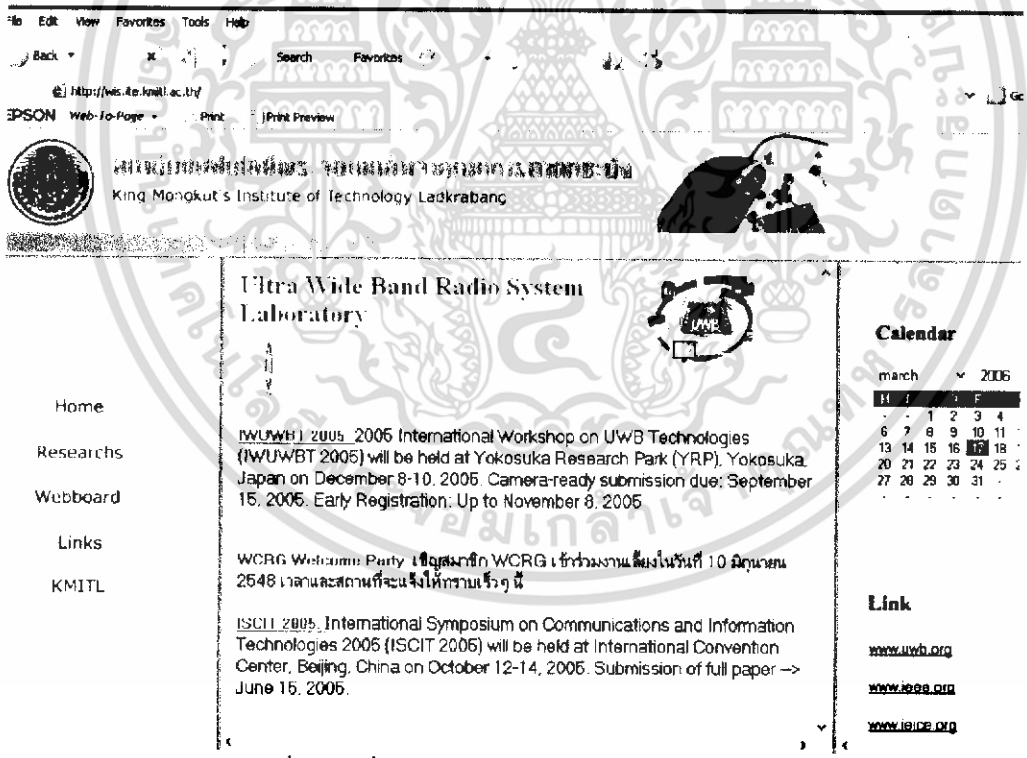


รูปที่ 4.26 เป็นรูปโฮมเพจหน้าแรก

รูปที่ 4.26 เป็นโฮมเพจหน้าแรกซึ่งมีลิ้งค์ต่างๆให้ทั้งบุคคลภายในและภายนอกสามารถเข้าชมได้แต่การตั้งกระทู้หรือดาวน์โหลดงานวิจัยต้องล็อกอินเพื่อตรวจสอบการเป็นสมาชิกก่อน

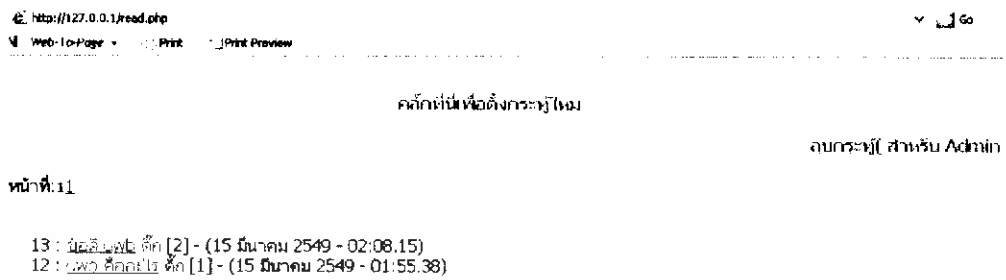


รูปที่ 4.27 เป็นการใส่ username และ password



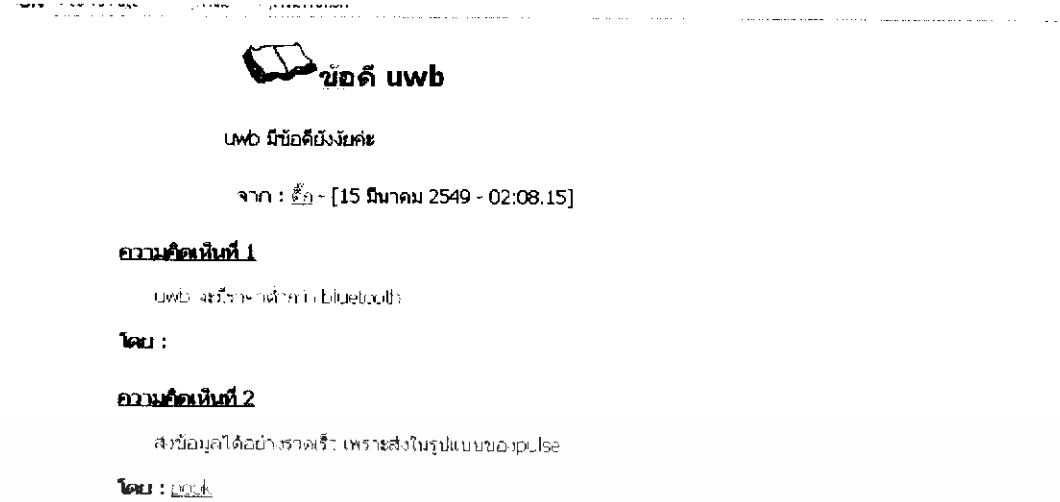
รูปที่ 4.28 เมื่อใส่ username และ password ถูกต้อง

จากรูปที่ 4.27 และ 4.28 เป็นการใส่ username และ password เพื่อให้เฉพาะสมาชิกในห้องวิจัยเท่านั้นที่สามารถเข้ามายังรูปที่ 4.28 ได้ เพื่อทำการตั้งกระทู้หรือตอบกระทู้ รวมถึงดาวน์โหลดงานวิจัยต่างๆ

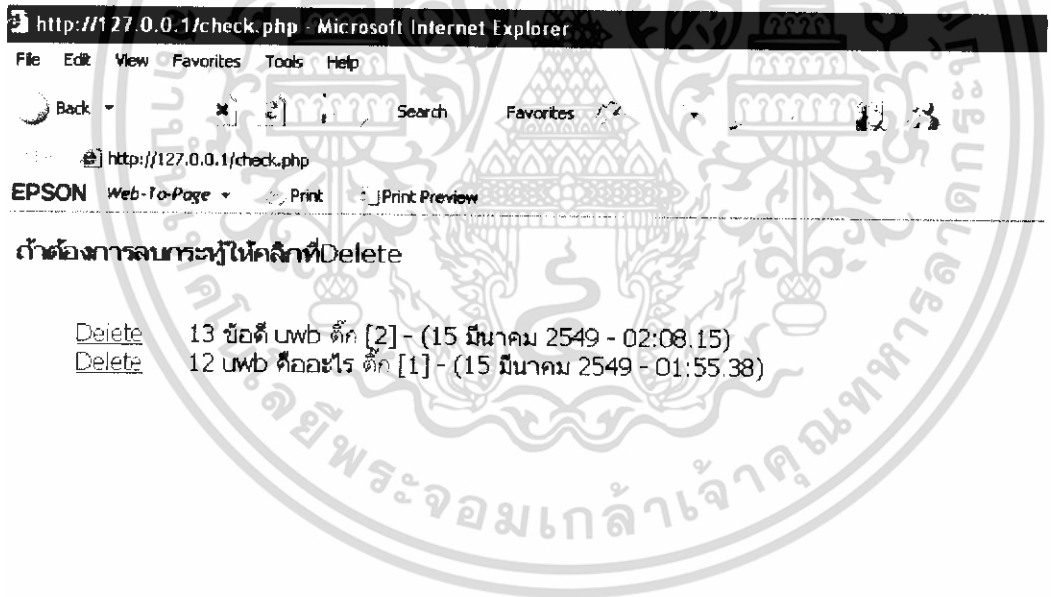


รูปที่ 4.29 ให้สมาชิกที่ลือกอินเข้ามาสามารถตั้งกระทู้ได้

รูปที่ 4.30 สมาชิกใส่รายละเอียดต่างๆเกี่ยวกับกระทู้

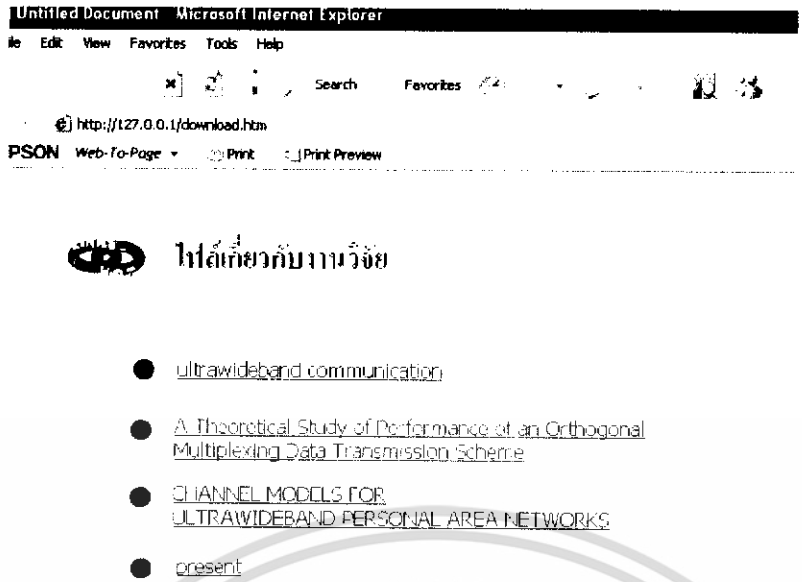


รูปที่ 4.31 ผู้ที่เป็นสมาชิกร่วมตอบกระทู้

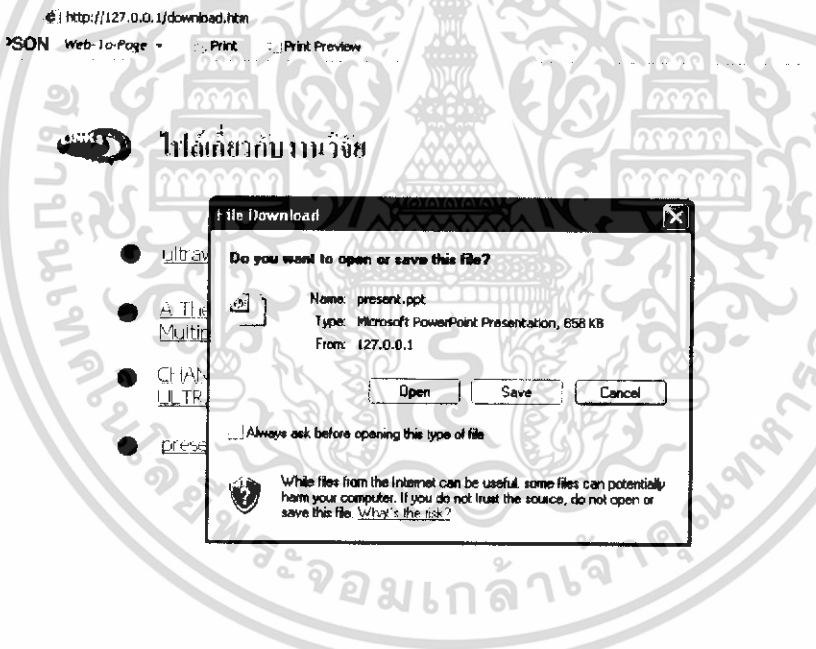


รูปที่ 4.32 ลบกระทู้ที่ไม่เหมาะสมได้

จากรูปที่ 4.29 – 4.32 สมาชิกที่ล็อกอินเข้ามาทำการตั้งกระทู้และตอบกระทู้ รวมถึงสามารถลบกระทู้ที่ไม่เหมาะสมออกได้ (สำหรับผู้ดูแลระบบเท่านั้น)



รูปที่ 4.33 สมาชิกสามารถเข้ามาอ่านหรือดาวน์โหลดงานวิจัยได้

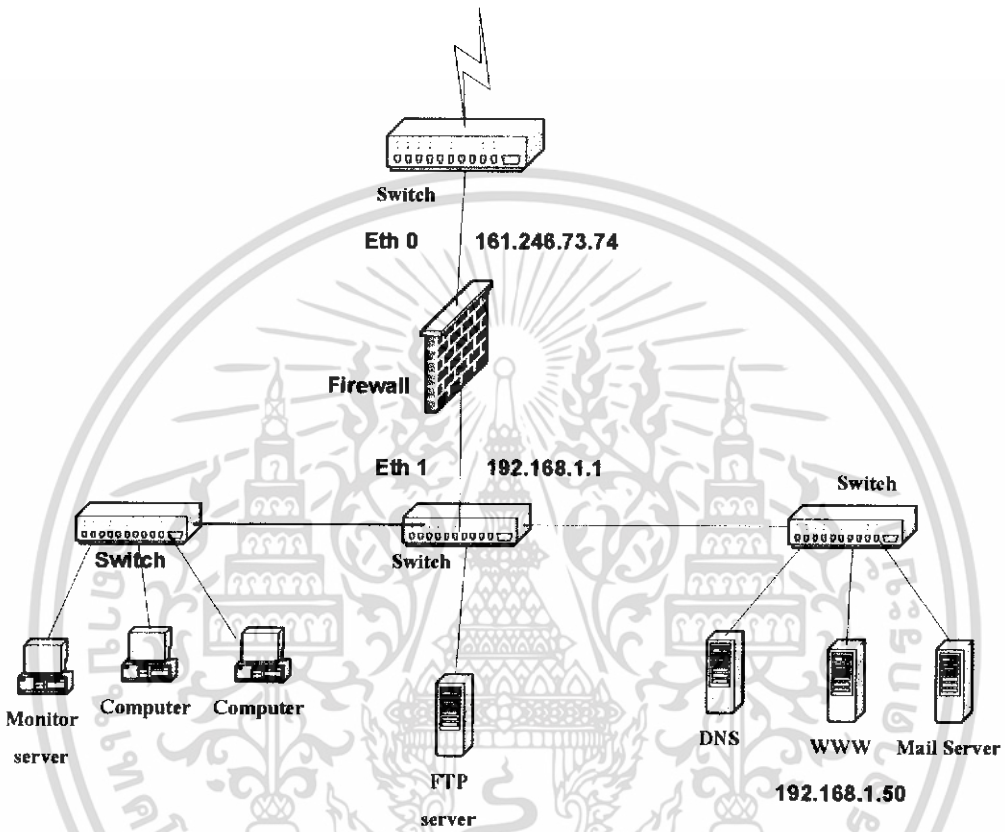


รูปที่ 4.34 สมาชิกสามารถดาวน์โหลดงานวิจัยได้

จากรูปที่ 4.33 – 4.34 สมาชิกที่ล็อกอินเข้ามานั้น สามารถเข้ามาอ่านงานวิจัยหรือดาวน์โหลดงานวิจัยของห้องวิจัยนี้ได้

บทที่ 5
บทสรุปและข้อเสนอแนะ

5.1 สรุปการออกแบบเน็ตเวิร์ก



รูปที่ 5.1 โครงสร้างเครือข่ายที่ทำการออกแบบแบบสมบูรณ์

- ไฟล์วอล สำหรับ ใช้เป็นเกตเวย์ติดต่อกับเครือข่ายภายนอก มีการกำหนดกฎการเข้า-ออก ของแพ็คเก็ตของข้อมูล และทำ NAT เพื่อแปลงแอดเดรสระหว่าง Private IP และ Public IP เพิ่มความปลอดภัยให้แก่เครือข่ายภายใน
- DNS Server สำหรับแปลงหมายเลขไอพีแอดเดรสให้อยู่ในรูปแบบของโดเมนเนม หรือ แปลงกลับจาก โดเมนเนมไปเป็นไอพีแอดเดรส ได้ หลังจากที่ได้ลงทะเบียนชื่อโดเมนผ่าน ISP เรียบร้อยแล้ว และประกาศโดเมนลูกเป็น “wis.kmitl.ac.th”
- เว็บเซิร์ฟเวอร์ สำหรับบริการการใช้งานต่าง ๆ ให้กับอาจารย์และนักศึกษาภายในห้องวิจัยแบบไร้สาย

- DHCPเซิร์ฟเวอร์ สำหรับ แจกไอพีให้แก่เครื่องไคลเอ็นต์แบบอัตโนมัติ
- มอนิเตอร์เซิร์ฟเวอร์ สำหรับดูแลจัดการคอมพิวเตอร์ของผู้ใช้ภายในเครือข่าย สามารถตรวจจับไวรัสในเครือข่ายอินทราเน็ต และแจ้งเตือนแก่ผู้ใช้ในระบบ
- ทำการติดตั้งเมล์เซริฟเวอร์ และเซิร์ฟเวอร์ข้อมูล ให้สมบูรณ์
- เพิ่มความปลอดภัยให้แก่ เซิร์ฟเวอร์ข้อมูล
- สร้างเว็บแอพพลิเคชั่นซึ่งมี การให้ user มาล็อกอิน เพื่อดาวน์โหลดและตั้งกระทู้เกี่ยวกับในห้องวิจัยได้ เพื่อบริการงานต่าง ๆ ให้แก่ผู้ใช้ภายในห้องวิจัย

5.2 ผลที่ได้รับ

- ระบบเครือข่ายภายในมีเสถียรภาพ และความปลอดภัยสูง ป้องกันการโจมตีจากเครือข่ายภายนอก ไปยังเครือข่ายภายในได้
- มีการใช้ Antivirus Gateway เพื่อกรอง packet ก่อนเข้ามายังเครือข่ายภายใน
- มีการตรวจจับ traffic ภายในเครือข่าย และวิเคราะห์เป็นกราฟ สำหรับดูแลและวิเคราะห์การใช้งานเครือข่าย
- สามารถใช้งานโดเมนลูก “wis.ite.kmitl.ac.th” ได้
- มีเว็บเซิร์ฟเวอร์ เพื่อประกาศข่าวสารต่าง ๆ ได้อย่างสะดวก
- มีการติดตั้ง เมล์เซิร์ฟเวอร์ เพื่อเพิ่มความยืดหยุ่นสำหรับการจัดใช้งานอีเมลล์
- DHCP สามารถทำงานได้อย่างมีประสิทธิภาพ ทำให้ผู้ใช้ไม่พบปัญหาเรื่องการเซตค่าไอพีแอดเดรสซ้ำซ้อน
- มีระบบแจ้งเตือนว่าพบไวรัส ในเครื่องหมายเลขไอพีใด ในระบบ

5.3 ปัญหา

5.3.1 งบประมาณ

- อุปกรณ์ต่าง ๆ เกี่ยวกับเน็ตเวิร์กมีราคาสูง ดังนั้นจึงเลือกใช้อุปกรณ์ที่สามารถใช้งานระดับปานกลาง
- ซอฟท์แวร์บางตัวที่จำเป็นจะต้องจ่ายค่าลิขสิทธิ์ จึงเลือกใช้ซอฟท์แวร์ที่เป็นโอเพ่นซอร์ส ซึ่งมีคุณสมบัติรองลงมา

5.4 แนวทางในการพัฒนาโครงงาน

- สามารถพัฒนาในส่วนเซิร์ฟเวอร์ข้อมูลโดยการทำระบบฐานข้อมูล (Database Server) เพื่อสะดวกในการใช้งานยิ่งขึ้น
- เพิ่มระบบความปลอดภัยและระบบตรวจจับผู้บุกรุก (IDS) ในระบบเครือข่าย

บรรณานุกรม

อ.บัณฑิต จามรภูติ, *คัมภีร์ Linux Redhat เล่ม 1*, สำนักพิมพ์ Bandhit, พ.ศ.2546

http://linux.thai.net/

http://linux.org

http://thaicert.nectec.or.th

ภาคผนวก ก.

Firewall

โปรแกรมที่ใช้สำหรับกำหนดนโยบายสำหรับเข้า – ออก ของข้อมูล

```
#!/bin/bash

FW_KMITL_EXT=161.246.73.74
FW_KMITL_INT=192.168.1.1
EXTIF=eth0
INTIF=eth1
NAT_KMITL=161.246.73.75

ip addr add 161.246.73.75 dev eth0
ip addr add 161.246.73.76 dev eth0
ip addr add 161.246.73.77 dev eth0
ip addr add 161.246.73.78 dev eth0
ip addr add 161.246.73.79 dev eth0
ip addr add 161.246.73.80 dev eth0
#enable forwarding
echo 1 > /proc/sys/net/ipv4/ip_forward

###block ip spoofing
for f in /proc/sys/net/ipv4/conf/*/rp_filter;
do
  echo 1 > $f
done

echo test firewall origin by pons

#clear policy
iptables -F
iptables -t nat -F
iptables -X
```

```
#define new policy
iptables -N KMITL-EXTRANET
iptables -N EXTRANET-KMITL

#default policy
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -P FORWARD DROP

########################input chain#######################

iptables -A INPUT -i $EXTIF -p tcp -d $FW_KMITL_EXT --dport 22 -j ACCEPT
iptables -A INPUT -i $EXTIF -p tcp -d $FW_KMITL_EXT --dport 25 -j ACCEPT  #ken
iptables -A INPUT -i $INTIF -p tcp -d $FW_KMITL_INT --dport 22 -j ACCEPT
iptables -A INPUT -p icmp --icmp-type echo-request -m limit --limit 10/minute -j ACCEPT
iptables -A INPUT -p icmp --icmp-type echo-reply -m limit --limit 10/minute -j ACCEPT
iptables -A INPUT -p tcp --syn -m limit --limit 1/s -j ACCEPT
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A INPUT -m state --state NEW -i eth0 -j DROP


########################output chain#######################

#iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
#iptables -A OUTPUT -p icmp -j ACCEPT


########################forward chain#######################

iptables -A FORWARD -i $EXTIF -o $INTIF -j EXTRANET-KMITL
```

```
iptables -A FORWARD -i $INTIF -o $EXTIF -j KMITL-EXTRANET

##########################define NAT#####################

iptables -t nat -A PREROUTING -d $NAT_KMITL -j DNAT --to-destination 192.168.1.50      #
NAT 161.246.73.75 --> 192.168.1.50 ##

#ipspoofing nat
iptables -t nat -A PREROUTING -i $EXTIF -s 10.0.0.0/8 -j DROP
iptables -t nat -A PREROUTING -i $EXTIF -s 127.0.0.0/8 -j DROP
iptables -t nat -A PREROUTING -i $EXTIF -s 172.16.0.0/12 -j DROP
iptables -t nat -A PREROUTING -i $EXTIF -s 192.168.1.0/16 -j DROP


iptables -t nat -A POSTROUTING -o $EXTIF -s 192.168.1.0/24 -d 0/0 -j SNAT --to
161.246.73.77-161.246.73.80
#iptables -t nat -A POSTROUTING -o $EXTIF -s 192.168.1.0/24 -d 0/0 -j SNAT --to
161.246.73.77

##########################KMITL-EXTRANET################
#iptables -A KMITL-EXTRANET -p tcp -s 192.168.1.50 -j ACCEPT
iptables -A KMITL-EXTRANET -p tcp -s 192.168.1.0/24 -j ACCEPT
#iptables -A KMITL-EXTRANET -p udp -s 192.168.1.50 -j ACCEPT
iptables -A KMITL-EXTRANET -p udp -s 192.168.1.0/24 -j ACCEPT
iptables -A KMITL-EXTRANET -p icmp --icmp-type echo-request -m limit --limit 10/minute -j
ACCEPT
iptables -A KMITL-EXTRANET -p icmp --icmp-type echo-reply -m limit --limit 10/minute -j
ACCEPT
#iptables -A KMITL-EXTRANET -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A KMITL-EXTRANET -j DROP

##########################EXTRANET-KMITL###################
```

```
iptables -A EXTRANET-KMITL -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A EXTRANET-KMITL -p tcp -d $NAT_KMITL --dport 80 -j ACCEPT
iptables -A EXTRANET-KMITL -p tcp -d 161.246.73.75 --dport 21 -j ACCEPT   #pons
iptables -A EXTRANET-KMITL -p tcp -d 161.246.73.75 --dport 25 -j ACCEPT   #ken
iptables -A EXTRANET-KMITL -p tcp -d 161.246.73.75 --dport 8080 -j ACCEPT   #ken
iptables -A EXTRANET-KMITL -p tcp -d 161.246.73.75 --dport 80 -j ACCEPT   #ken
iptables -A EXTRANET-KMITL -p tcp -d 192.168.1.50 --dport 80 -j ACCEPT
iptables -A EXTRANET-KMITL -j DROP
```

ภาคผนวก ข.

Mail server

1 ทำการติดตั้งโปรแกรม postfix, dovecot

2 ทำการติดตั้งโปรแกรม clamav, amavisd-new

3 ทำการติดตั้งโปรแกรม squirrelmail

วิธีการติดตั้ง postfix

1 ทำการ unpackage โดยใช้คำสั่ง tar xvfz postfix.tar.gz

2 ทำการ compile โปรแกรม โดยใช้คำสั่ง make , make install

3 ทำการ config ไฟล์ main.cf (/etc/postfix/main.cf)

โดยแก้ไข

mail_owner = postfix

myhostname = wis.ite.kmitl.ac.th

mydomain = ite.kmitl.ac.th

myorigin = $myhostname

inet_interfaces = all

mynetworks_style = class

mynetworks = 168.100.189.0/28, 127.0.0.0/8, 161.246.73.0/24

alias_maps = hash:/etc/aliases

setgid_group = postdrop

และเพิ่ม

content_filter=smtp-amavis:[127.0.0.1]:10024

maildrop_destination_recipient_limit=1

ต่อท้ายไฟล์ main.cf เพื่อให้รู้จักโปรแกรม amavis

4 ทำการ config ไฟล์ master.cf (/etc/postfix/master.cf)

ทำการเพิ่ม

smtp-amavis unix - - n - 2 lmtp

-o lmtp_data_done_timeout=1200

-o lmtp_send_xforward_command=yes

127.0.0.1:10025 inet n - n - - smtpd

-o content_filter=

-o local_recipient_maps=

-o relay_recipient_maps=

-o smtpd_restriction_classes=

-o smtpd_client_restrictions=

-o smtpd_helo_restrictions=

-o smtpd_sender_restrictions=

-o smtpd_recipient_restrictions=permit_mynetworks,reject

-o mynetworks=127.0.0.0/8

-o strict_rfc821_envelopes=yes

-o smtpd_error_sleep_time=0

-o smtpd_soft_error_limit=1001

-o smtpd_hard_error_limit=1000

ต่อท้ายไฟล์ master.cf

5 ใช้คำสั่ง postfix start สำหรับรันโปรแกรม

6 ใช้คำสั่ง postfix reload สำหรับการรีโหลดเมื่อมีการ config ค่าใหม่

7 ทำการตรวจสอบว่า สามารถใช้งานได้หรือไม่จาก telnet localhost 25

หมายเหตุ

ต้องทำการสร้าง user ก่อน ในที่นี้คือ postfix (useradd postfix)

และ group ในที่นี้คือ postdrop (groupadd postdrop)

วิธีการติดตั้ง amavisd-new

1 ทำการดาวน์โหลด module เบื้องต้น ซึ่งสามารถดาวน์โหลดได้จาก www.cpan.org

Archive::Tar (Archive-Tar-x.xx)

Archive::Zip (Archive-Zip-x.xx) (1.14 or later should be used!)

Compress::Zlib (Compress-Zlib-x.xx)

Convert::TNEF (Convert-TNEF-x.xx)

Convert::UUlib (Convert-UUlib-x.xxx) (1.05 or later, stick to new versions!)

MIME::Base64 (MIME-Base64-x.xx)

MIME::Parser (MIME-Tools-x.xxxx) (latest version from CPAN - currently 5.417)

Mail::Internet (MailTools-1.58 or later have workarounds for Perl 5.8.0 bugs)

Net::Server (Net-Server-x.xx) (version 0.88 finally does setuid right)

Net::SMTP (libnet-x.xx, ports/net/p5-Net) (>= libnet-1.16 for performance)

Digest::MD5 (Digest-MD5-x.xx) (2.22 or later)

IO::Stringy (IO-stringy-x.xxx)

Time::HiRes (Time-HiRes-x.xx) (use 1.49 or later, older can cause problems)

Unix::Syslog (Unix-Syslog-x.xxx)

BerkeleyDB with bdb library 3.2 or later (4.2 or later preferred)

การคอมไพล์ perl module ข้างต้น ให้ติดตั้ง perl compiler ก่อน แล้วทำการ compile โดยใช้คำสั่ง perl makefile.pl

จากนั้นทำการติดตั้ง perl module จากคำสั่ง make และ make install

2 ทำการ unpackage โดยใช้คำสั่ง tar amavisd-new.tar.gz

3 ทำการ compile โปรแกรม โดยใช้คำสั่ง make , make install

4 ทำการสร้าง user ก่อน ในที่นี้จะใช้ clamav (useradd clamav)

5 และ สร้าง group ในที่นี้จะใช้ clamav (groupadd clamav)

6 ทำการสร้าง home directory สำหรับ amavisd

7 mkdir /var/amavis

mkdir /var/amavis/tmp /var/amavis/var /var/amavis/db

chown -R amavis:amavis /var/amavis

chmod -R 750 /var/amavis

cp amavisd /usr/local/sbin/

```
chown root /usr/local/sbin/amavisd
chmod 755  /usr/local/sbin/amavisd
cp amavisd.conf /etc/
chown root /etc/amavisd.conf
chmod 644  /etc/amavisd.conf
mkdir /var/virusmails
chown amavis:amavis /var/virusmails
chmod 750 /var/virusmails
/usr/local/sbin/amavisd debug
```

8 ทดสอบการรันโปรแกรม /usr/local/sbin/amavisd

9 ทำการ config ไฟล์ amavisd.conf (/etc/amavisd.conf)

```
$daemon_user  = 'clamav';
$daemon_group = 'clamav'

 ['ClamAV-clamd',
   \&ask_daemon, ["CONTSCAN {}\n", "/var/run/clamav/clamd"],
   qr/\bOK$/, qr/\bFOUND$/,
   qr/^.*?: (?!Infected Archive)(.*) FOUND$/ ],

$inet_socket_port = 10024;
```

10 ทำการตรวจสอบว่า โปรแกรม amavisd สามารถใช้งานได้หรือไม่จาก

```
telnet localhost 10024
telnet localhost 10025
```

วิธีการติดตั้ง clamav

1 ทำการ unpackage โดยใช้คำสั่ง tar xvfz clamav.tar.gz

2 ทำการ compile โปรแกรม โดยใช้คำสั่ง make, make check และ make install

3 ทำการ config ไฟล์ clamd.conf

LocalSocket /var/run/clamav/clamd

User clamav

4 ทำการ config ไฟล์ freshclam.conf ให้ทำการ config ค่าตามความเหมาะสม หรือถ้าไม่ต้องการ set ค่า ก็ให้ ทำการคอมเมนต์ที่

#Example

5 ทำการทดสอบโปรแกรมทั้งสามโปรแกรมให้ใช้งานร่วมกันได้ โดยทำการทดสอบส่งเมล์และแนบไวรัส จากไฟล์ไดเรกทอรี่ test ของโปรแกรม clamav เพื่อทดสอบ และดูผลจาก log file (/var/log/mailllog) และถ้าระบบมีการแจ้งไวรัส และทำการเก็บไฟล์ใน /var/virusmails แสดงว่าระบบสามารถใช้งานได้

วิธีการติดตั้ง dovecot

1 ทำการ unpackage โดยใช้คำสั่ง tar xvfz dovecot.tar.gz

2 ทำการ compile โปรแกรม โดยใช้คำสั่ง make, make install

3 ทำการ config ไฟล์ dovecot.cont

โดยเพิ่มให้สามารถใช้งานโปรโตคอล pop secure และ imap secure ได้

protocols = imap imaps pop3 pop3s

4 ทำการรันโปรแกรม dovecot

5 ทำการทดสอบโปรแกรมว่าสามรถใช้งาน โดยการ telnet localhost 110 เพื่อทำการทดสอบการใช้งานโปรโตคอล pop และ telnet localhost 143 เพื่อทำการทดสอบการใช้งานโปรโตคอล imap