

ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล.

การพัฒนาระบบควบคุม

ปริมาณจราจรในเครือข่าย

Development of Traffic Control System

โดย

อดิษฐ์ ศิลารมย์

รหัสประจำตัว 46066224



H002313

อาจารย์ที่ปรึกษา

ผศ. ดร. โชติพัชร ภรณ์วลัย

วัน เดือน ปี.....	19 ก.พ. 2550
เลขทะเบียน.....	02313
เลขเรียกหนังสือ.....	๑๗ ๑131ก 2547
"ห้องสมุดคณะเทคโนโลยีสารสนเทศ สจล."	

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

ภาคเรียนที่ 2 ปีการศึกษา 2547

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ การพัฒนาระบบควบคุมปริมาณกราฟฟิกในเครือข่าย
นักศึกษา นาย อติศัย ศีลารมย์
อาจารย์ที่ปรึกษา ผศ.ดร. โชติพัชร ภรณ์วลัย
ระดับการศึกษา วิทยาศาสตร์มหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
แขนงวิชา วิทยาการสารสนเทศ
ปีการศึกษา 2547

บทคัดย่อ

ระบบการควบคุมปริมาณกราฟฟิกนั้น เป็นชุดระบบการทำงานตัวที่มีอยู่ในตัวเคอร์เนลระบบปฏิบัติการลินุกซ์ ซึ่งมีไว้สำหรับจัดการควบคุมการใช้ปริมาณกราฟฟิกให้ตรงกับความต้องการในแต่ละเครือข่ายที่แตกต่างกันไปได้อย่างเหมาะสม แต่ด้วยภาษาที่ใช้ในการคอนฟิกค่าของตัวระบบควบคุมปริมาณกราฟฟิกนั้นมีความยากและซับซ้อนในการใช้งาน

จึงได้ทำการพัฒนาระบบกราฟฟิกยูสเซอร์อินเทอร์เฟซ ในรูปแบบเว็บเบสแอปพลิเคชัน โดยใช้ภาษาพีเอชพีและฐานข้อมูลมายเอสคิวเอล เพื่อใช้ทำการควบคุมการทำงานของระบบเดิมเพื่อที่จะลดความซับซ้อนของตัวระบบลงและทำให้ผู้ใช้งานสามารถใช้งานได้ง่ายขึ้น และทำการเสริมความสามารถของระบบเดิม ที่ใช้ไอพีเทเบิลในการกรองแพ็กเก็ตที่ขึ้นเน็ตเวิร์กเลขอร์ให้สามารถกรองแพ็กเก็ตได้ในระดับแอปพลิเคชันเลเยอร์ ซึ่งจะช่วยให้ระบบจัดการควบคุมกราฟฟิกในประเภทต่าง ๆ ให้มีประสิทธิภาพมากขึ้นอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title Development of Traffic Control System
Advisor Asst. Prof. Dr. Chotipat Pornavalai
Level of Study Master of Science in Information Technology
Major Information Science
Academic Year 2004

ABSTRACT

Traffic Control System is set of operating system that is in kernel of Linux operating system which is for controlling amount of traffic to support requirement each of different network but languages that use for configure values are difficult and complicate to use.

Therefore, I have developed graphic user interface in web base application by using PHP Language and My SQL database to control operating of old system to eliminate complicate of system and easy to use and increase efficiency of old system. By using IP Table to filter packages at Network Layer to filter packages in level of Application Layer which helps the Control Traffic System more efficiency.

กิตติกรรมประกาศ

โครงการพัฒนาระบบงานนี้ สำเร็จได้ด้วยความช่วยเหลือและการสนับสนุนจากหลายฝ่าย ดังต่อไปนี้

1. ผศ.ดร. โชติพัชร ภรณวลัย ซึ่งเป็นอาจารย์ที่ปรึกษาโครงการและอาจารย์ที่ปรึกษาสัมมนา ที่กรุณาให้คำปรึกษาในข้อปัญหาต่าง ๆ ที่เกิดขึ้นระหว่างพัฒนาระบบงาน
2. คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เอื้อเพื่อหนังสือในการค้นคว้า เพื่อพัฒนาระบบงาน
3. พี่จ๋อน ที่กรุณาให้คำปรึกษาเกี่ยวกับระบบปฏิบัติการลินุกซ์และคำแนะนำในการพัฒนาระบบงาน
4. เพื่อน ๆ ทุกคนที่คอยให้คำแนะนำ และให้ความช่วยเหลือ และเป็นกำลังใจมาโดยตลอด

จึงใคร่ขอขอบคุณบุคคลดังกล่าวข้างต้นมา ณ โอกาสนี้

นายอดิษฐ์ สิลารมย์

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII
บทที่	
1. บทนำ.....	1
1.1. ความเป็นมาและความสำคัญ.....	1
1.2. วัตถุประสงค์ของการพัฒนาระบบ.....	2
1.3. ขอบเขตของการพัฒนาระบบ.....	2
1.4. ขั้นตอนการพัฒนาระบบ.....	3
1.5. ประโยชน์ที่คาดว่าจะได้รับ.....	3
1.6. เครื่องมือที่ใช้ในการพัฒนาระบบ.....	3
2. ทฤษฎีที่เกี่ยวข้อง.....	5
2.1. เราเตอร์ (Router).....	5
2.2. เว็บเซิร์ฟเวอร์ (Web Server).....	6
2.2.1 การติดตั้งเว็บเซิร์ฟเวอร์บนระบบปฏิบัติการลินุกซ์.....	7
2.3. HTML.....	9
2.4. PHP.....	10
2.5.1 หลักการทำงานของ PHP.....	11
2.5.2 การติดตั้ง PHP บนระบบปฏิบัติการลินุกซ์.....	11
2.5. My SQL.....	11
2.6. Traffic Control System.....	12

สารบัญ (ต่อ)

หน้า

2.6.1	องค์ประกอบที่สำคัญของ Traffic Control.....	14
2.6.2	องค์ประกอบต่าง ๆ ของ Traffic Control ใน Linux.....	15
2.6.3	Classless Queuing Disciplines.....	17
2.6.4	Hierarchical Token Bucket (HTB).....	20
2.7.	L7 Netfilter	23
2.8.	Tc Language.....	28
2.8.1	Filter with Iptables.....	29
3.	การวิเคราะห์และออกแบบระบบ.....	31
3.1.	การทำงานของระบบเดิม.....	31
3.2.	ปัญหาที่พบในระบบปัจจุบัน.....	32
3.3.	แนวความคิดในการออกแบบระบบ.....	32
3.4.	ขั้นตอนในการทำงานของระบบใหม่.....	33
3.5.	การออกแบบฐานข้อมูล.....	41
3.6.	Data Dictionary	43
4.	การพัฒนาและทดสอบระบบ.....	48
4.1	เครื่องมือและซอฟต์แวร์ที่ใช้ในการพัฒนาระบบ.....	48
4.2	ผลการพัฒนาระบบ.....	49
4.3	การทดสอบการทำงานของระบบ.....	50
5.	บทสรุป.....	53
5.1	สรุปผลการออกแบบและพัฒนาระบบ.....	53
5.2	ปัญหาและข้อจำกัดที่พบ.....	53
5.3	ประโยชน์ที่ได้รับจากการออกแบบและพัฒนาระบบ.....	54

สารบัญ (ต่อ)

	หน้า
ภาคผนวก ก.....	55
ภาคผนวก ข.....	63
บรรณานุกรม.....	73
ประวัติผู้เขียน.....	74



สารบัญตาราง

ตารางที่	หน้า
3.1 รายละเอียดตาราง Admin.....	43
3.2 รายละเอียดตาราง Interface.....	43
3.3 รายละเอียดตาราง Class.....	43
3.4 รายละเอียดตาราง Filter.....	44
3.5 รายละเอียดตาราง Host.....	45
3.6 รายละเอียดตาราง Host_detail.....	45
3.7 รายละเอียดตาราง Grouphost.....	46
3.8 รายละเอียดตาราง Service.....	46
3.9 รายละเอียดตาราง Service_detail.....	46
3.10 รายละเอียดตาราง Groupservice.....	47

สารบัญรูป

หน้า

รูปที่

2.1 หลังจากที่ใช้คำสั่ง setup แล้วคลิกเลือก System services.....	8
2.2 หลังจากทีคลิกเลือก System services แล้วกาเครื่องหมาย * ข้างหน้า httpd.....	8
2.3 เรียกเว็บไซต์ Default ที่คิดมากับ httpd.....	9
2.4 การทำงานของ PHP.....	10
2.5 Network Packet Processing using Linux Traffic Control.....	13
2.6 การทำงานของ FIFO (First-in First-out).....	17
2.7 การทำงานของ pffifo_fast.....	18
2.8 การทำงานของ Token Bucket Filter.....	19
2.9 ลักษณะ โครงสร้างของ Link Sharing.....	21
2.10 ลักษณะ โครงสร้างของ Sharing hierarchy.....	22
2.11 ตาราง ascii table ใช้เปรียบเทียบค่าในการอ่าน Regular Expression.....	26
2.12 รูปจำลองจากการคอนฟิกค่าใน tc utility.....	29
3.1 รูปแบบการทำงานในระบบเดิม.....	31
3.2 รูปแบบการคอนฟิกค่าในระบบเดิม.....	31
3.3 ภาพของรูปแบบในการพัฒนาระบบใหม่.....	33
3.4 Window Navigation Diagram ของส่วนเมนูหลัก.....	34
3.5 Window Navigation Diagram ของส่วนเมนูการคอนฟิกค่าในอินเทอร์เน็ตเฟส.....	35
3.6 Window Navigation Diagram ของส่วนเมนูการกำหนดรายละเอียดตัวกรอง.....	36
3.7 Flow Chart แสดงถึงการกำหนดแบนวิดธ์ในแต่ละอินเทอร์เน็ตเฟส.....	37
3.8 Flow Chart แสดงถึงการสร้างคลาสในแต่ละอินเทอร์เน็ตเฟส.....	38
3.9 Flow Chart แสดงถึงการกำหนดค่ารายละเอียดของตัวกรอง.....	39
3.10 Flow Chart แสดงถึงการกำหนดเงื่อนไขในแต่ละคลาส.....	40
3.11 แสดง E-R Diagram.....	42

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.1 หน้าจอการเรียกใช้งานระบบจากโปรแกรมบราวเซอร์.....	49
4.2 สภาพแวดล้อมเครือข่ายที่ใช้ในการทดสอบระบบ.....	50
4.3 การคอนฟิกค่าของระบบในการทดสอบ.....	51
4.4 การทดสอบจับอัตราความเร็วในการดาวน์โหลดในแต่ละคลาส.....	52



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญ

ในปัจจุบันเครือข่ายอินเทอร์เน็ตมีแนวโน้มในการเจริญเติบโตที่รวดเร็วมาก รูปแบบการทำธุรกรรมและบริการต่าง ๆ เกิดขึ้นมากมายบนโลกอินเทอร์เน็ต ไม่ว่าจะเป็นการซื้อขายสินค้า การค้นหาข้อมูลต่างๆ รวมถึงบริการอื่น ๆ ที่มีประโยชน์อีกมากมาย ทำให้เกิดความต้องการที่จะเชื่อมต่อเข้าสู่โลกอินเทอร์เน็ตมากขึ้น แต่การจะเชื่อมต่อเข้าสู่โลกอินเทอร์เน็ตนั้นก็ต้องแลกกับค่าใช้จ่ายที่ต้องเสียในการเชื่อมต่อ ซึ่งราคาดังกล่าวจะสูงหรือจะต่ำนั้นก็ขึ้นอยู่กับคุณภาพของการเชื่อมต่อหรืออีกนัยหนึ่งก็คือความเร็วและความราบรื่นในการเข้าสู่โลกอินเทอร์เน็ตนั่นเอง โดยคุณภาพที่ได้นั้นก็ขึ้นกับการเชื่อมต่อที่มีการใช้งานเพียงคนเดียวหรือผู้ใช้งานตามบ้านนั้นก็อาจไม่มีปัญหาในการใช้งานเท่าใดนัก เนื่องจากผู้ใช้งานสามารถเลือกที่จะจัดสรรความแบนวิดธ์ที่ได้มาตามต้องการได้ เช่น ถ้าต้องการที่จะเล่นเว็บให้เร็วก็อาจจะต้องหยุดการทำงานอย่างอื่นก่อน เช่น การโหลดข้อมูลขนาดใหญ่ แต่ถ้าการเชื่อมต่อนั้นมีผู้ใช้มากกว่าหนึ่งคนหรือมีการใช้งานมากกว่าหนึ่งจุดนั้นก็อาจจะทำให้เกิดปัญหาในการจัดสรรแบนวิดธ์ที่ได้มา เพราะอาจจะมีบางจุดที่มีการใช้งานที่ต้องการแบนวิดธ์มากทำให้อีกจุดหนึ่งไม่สามารถใช้งานได้ เนื่องจากการกลไกการทำงานในการใช้งานในระบบเครือข่ายนั้นเป็นแบบการแข่งขันเพื่อใช้งาน คือพยายามที่จะใช้งานให้ได้มากที่สุดเท่าที่จะทำได้ จึงอาจทำให้ผู้ใช้บางคนหรือบริการบางจุดไม่สามารถใช้งานได้โดยมีประสิทธิภาพเท่าใดนัก

ด้วยเหตุนี้จึงได้มีการพัฒนาตัวควบคุมปริมาณทราฟฟิกในเครือข่ายขึ้นมาทั้งในรูปแบบของอุปกรณ์ และซอฟต์แวร์ทั้งนี้ขึ้นอยู่กับทางเลือกใช้งานให้เหมาะสมกับเครือข่ายนั้น ซึ่งในตัวซอฟต์แวร์ที่ได้ถูกพัฒนาออกมาให้สามารถนำไปใช้ได้ฟรีก็จะมีอยู่ในระบบปฏิบัติการของลินุกซ์ซึ่งคิดมากับตัวเคอร์เนลของตัวระบบปฏิบัติการเลย แต่ถึงจะซอฟต์แวร์ที่ได้มีการพัฒนามาให้มีประสิทธิภาพที่ดีและสามารถนำไปใช้ฟรีได้นั้น แต่ผู้ใช้อาจจะประสบกับปัญหาในการใช้งานอีกเนื่องจากในการใช้งานนั้นยังต้องการผู้ใช้ที่มีความรู้ความเข้าใจในตัวระบบและด้วยภาษาในการคอนฟิกที่มีความซับซ้อนและยากในการใช้งาน

โครงการนี้จึงได้ทำการพัฒนาระบบ ที่จะสามารถให้ผู้ใช้ระบบควบคุมปริมาณทราฟฟิกในเครือข่ายได้ง่ายขึ้นโดยใช้รูปแบบของ Graphic User Interface (GUI) จึงทำให้ผู้ใช้ไม่ต้อง

มีความเชี่ยวชาญในการคอนฟิกค่าต่าง ๆ มากนัก โดยการพัฒนาจะใช้เทคโนโลยีของ Web-Base Application ซึ่งผู้ใช้สามารถที่จะทำการควบคุมระบบจากที่ใดก็ได้เพียงแค่บนเครื่องของผู้ใช้มีโปรแกรมบราวเซอร์เท่านั้นเอง รวมทั้งเพิ่มความสามารถในการตรวจสอบแพ็กเกจในระดับชั้นแอปพลิเคชันเลเยอร์โดยใช้ความสามารถของ L7 Netfilter (patch ใน iptables) เพื่อที่จะตรวจสอบโปรแกรมหรือบริการบางประเภทที่สามารถทำการเปลี่ยนพอร์ตบริการในการใช้งานได้ เช่น โปรแกรมประเภท Peer to Peer ที่สามารถปรับเปลี่ยนพอร์ตในการใช้งานได้โดยไม่มีรูปแบบ ซึ่งจะช่วยให้ตัวโปรแกรมมีประสิทธิภาพในการควบคุมรูปแบบของการใช้งานเครือข่ายได้มากขึ้นนั่นเอง

1.2 วัตถุประสงค์ของพัฒนาระบบ

1. เพื่อที่จะได้ทำการศึกษาการทำงานของระบบควบคุมปริมาณกราฟฟิกในระบบปฏิบัติการลินุกซ์เพื่อที่จะระบุถึงขอบเขตและความสามารถในการทำงานของตัวระบบ
2. ทำการพัฒนาระบบ GUI (Graphic User Interface) เพื่อที่จะทำการควบคุมระบบควบคุมปริมาณกราฟฟิกเดิมในระบบปฏิบัติการลินุกซ์เพื่อให้ใช้งานได้ง่ายขึ้น
3. ทำการเพิ่มความสามารถของตัวกรองแพ็กเกจ ในไอพีเทเบิลซึ่งเราจะใช้ในการตรวจสอบประเภทของกราฟฟิกให้มีความสามารถในการกรองแพ็กเกจได้ในระดับชั้นแอปพลิเคชันเลเยอร์จากเดิมที่กรองได้แค่ในระดับเน็ตเวิร์กเลเยอร์
4. เพื่อเป็นแนวทางในการออกแบบ และพัฒนาระบบควบคุมปริมาณกราฟฟิกในตัวระบบปฏิบัติการลินุกซ์ในรูปแบบใหม่ให้มีประสิทธิภาพมากยิ่งขึ้น

1.3 ขอบเขตของการพัฒนาระบบ

1. ทำการพัฒนา Graphic User Interface เพื่อทำการควบคุมระบบควบคุมปริมาณกราฟฟิกในเครือข่ายในตัวระบบปฏิบัติการลินุกซ์ในการทำงานพื้นฐานหลัก ๆ บางส่วน
2. ทำการพัฒนาเพิ่มความสามารถของการคัดแยกประเภทของกราฟฟิกซึ่งใช้ไอพีเทเบิลในการคัดแยกให้มีความสามารถกรองได้ในระดับแอปพลิเคชันเลเยอร์
3. ทำการพัฒนาให้เหมาะสมกับการใช้งานในเครือข่ายที่มีการควบคุมปริมาณกราฟฟิกที่ไม่มีความซับซ้อนในการใช้งานมากนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมีให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขั้นตอนการพัฒนาระบบ

1. ศึกษาระบบการทำงานของระบบควบคุมปริมาณกราฟฟิคในเครือข่ายในตัวระบบปฏิบัติการลินุกซ์เพื่อให้เข้าถึงกลไกการทำงานของตัวระบบ
2. ศึกษาถึงโปรแกรมหรือบริการที่มีความยืดหยุ่นบางตัว เพื่อที่จะนำไปสร้างกฎเกณฑ์ในการควบคุมในการใช้งานของตัวโปรแกรมและบริการเหล่านั้น
3. ศึกษาถึงเทคโนโลยีที่เกี่ยวข้องในการพัฒนาระบบเพื่อจะได้ทำการเลือกใช้อย่างเหมาะสมและมีประสิทธิภาพมากที่สุด

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ศึกษาถึงความสามารถในการทำงานเพื่อควบคุมปริมาณกราฟฟิคในเครือข่ายและ ความสามารถด้านเน็ตเวิร์คในตัวระบบปฏิบัติการลินุกซ์แล้วนำมาประยุกต์ใช้งานได้อย่างมีประสิทธิภาพ
2. เพื่อเพิ่มประสิทธิภาพการใช้งานของตัวระบบให้มีการใช้งานที่ง่ายขึ้น
3. สามารถนำระบบไปประยุกต์ใช้ในองค์กรได้อย่างมีประสิทธิภาพ

1.6 เครื่องมือที่ใช้ในการพัฒนาระบบ

1. ซาร์ตแวร์ (คอมพิวเตอร์จำนวน 2 เครื่อง) สำหรับทำเป็นเราเตอร์หรือเครื่อง Gateway 1 เครื่อง และเป็นเครื่องไคลเอ็นท์ 1 เครื่อง
 - CPU AMD Atlon(TM) MP 1600+
 - RAM 512
 - Hard disk 40 GB
2. ซอฟต์แวร์
 - OS Linux Redhat 9.0 (เครื่องที่ทำหน้าที่เป็นเราเตอร์)
 - OS Window XP Professional (เครื่องที่ทำหน้าที่เป็นไคลเอ็นท์)
 - Apache_1.3.33 (Web Server)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Php-4.3.10 (ภาษาในการเขียนเว็บแอปพลิเคชัน)
- MySQL-server-4.1.10-0 (ฐานข้อมูล)
- MySQL-client-4.1.10-0 (ฐานข้อมูล)
- Edit plus 2.10c (เครื่องมือในการเขียน โปรแกรม)
- Iproute2 (โปรแกรมควบคุมปริมาณทราฟฟิก)
- Iptable-1.2.9 (โปรแกรมเพื่อทำการคัดแยกประเภทของทราฟฟิก)
- kernel-2.4-layer7-0.9.1.patch (patch เพื่อให้เคอร์เนลสนับสนุนให้ไอพีเทเบิลสามารถกรองแพ็กเก็ตในระดับชั้นแอปพลิเคชันเลเยอร์ได้)
- iptables-layer7-1.0.patch (patch เพื่อเพิ่มความสามารถของไอพีเทเบิลในการกรองแพ็กเก็ตในระดับแอปพลิเคชันเลเยอร์ได้)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ระบบควบคุมปริมาณกราฟิก

2.1 เราเตอร์ (Router)

เราเตอร์เป็นอุปกรณ์ที่ใช้เป็นตัวเชื่อมระหว่างเน็ตเวิร์กสองเน็ตเวิร์ก หรือหลาย ๆ เน็ตเวิร์ก เพื่อให้สามารถรับส่งข้อมูลกันได้ เราเตอร์มีหน้าที่ในการเลือกเส้นทางในการที่จะส่งแพ็กเก็ต (ซึ่งเป็นข้อมูลในระดับ Layer 3) เพื่อให้แพ็กเก็ตดังกล่าววิ่งไปยังปลายทางที่ถูกต้องและเหมาะสมที่สุด โดยมันจะอ่านค่าแอดเดรสปลายทางจากเฮดเดอร์ของแพ็กเก็ต จากนั้นมันก็จะประมวลผลเพื่อตัดสินใจว่าจะทำการส่งผ่านไปบนเส้นทางใด (วิ่งออกทางอินเตอร์เฟซใด) เราเตอร์เป็นอุปกรณ์ที่ทำงานในระดับ Layer 3 (Network Layer) ของ OSI Reference Model เนื่องจากมันมีการตัดสินใจเลือกเส้นทาง โดยพิจารณาจากค่าแอดเดรสที่อยู่ในแพ็กเก็ต

2.2.1 การใช้ระบบปฏิบัติการลินุกซ์ทำงานเป็นเราเตอร์

ระบบปฏิบัติการลินุกซ์นั้นมีความสามารถหลายอย่าง รวมถึงความสามารถที่จะทำตัวเองเป็นเราเตอร์แต่เราก็ต้องคำนึงด้วยว่ามันเป็นเพียงการจำลองการทำงานของเราเตอร์เท่านั้น ดังนั้นความสามารถในการทำงานของมันนั้นจึงไม่อาจเทียบเท่าอุปกรณ์เราเตอร์จริง ซึ่งความสามารถของตัวมันนั้นก็ขึ้นอยู่กับฮาร์ดแวร์ (เครื่องคอมพิวเตอร์) ที่นำมาใช้ เพราะฉะนั้นจึงอาจไม่ค่อยเหมาะสมที่จะนำมาใช้กับเครือข่ายที่มีข้อมูลหรือกราฟิกวิ่งผ่านจำนวนมาก ๆ แต่ก็มีความสามารถพอเพียงที่จะนำมาใช้เครือข่ายขนาดเล็กและขนาดกลางที่มีจำนวนข้อมูลหรือกราฟิกวิ่งผ่านจำนวนไม่มากนัก

การทำให้ระบบปฏิบัติการลินุกซ์ทำงานเป็นเราเตอร์นั้น จะต้องใช้เครื่องคอมพิวเตอร์ที่มีการติดตั้งเน็ตเวิร์กการ์ด (Network Card) มากกว่า 1 การ์ด โดยเราจะต้องคอนฟิกค่าไอพีแอดเดรส สับเน็ต และค่าอื่น ๆ ตามแต่ความเหมาะสมกับเครือข่ายนั้น ๆ ให้กับเน็ตเวิร์กการ์ด

เมื่อทำการคอนฟิกค่าให้กับเน็ตเวิร์กการ์ดแล้ว จะต้องทำการแก้ไขค่าบางตัวเพื่อที่จะให้ระบบทำการฟอเวิร์ดแพ็กเก็ตได้ ซึ่งค่าดีฟอลท์ในระบบปฏิบัติการลินุกซ์นั้นถึงเราจะทำการติดตั้งเน็ตเวิร์กการ์ดมากกว่าหนึ่งการ์ดแล้ว ระบบก็ยังไม่อนุญาตให้เราทำการฟอเวิร์ดแพ็กเก็ตไป

ยังเน็ตเวิร์กการ์ดอื่น ซึ่งอยู่ในไฟล์ชื่อ `/proc/sys/net/ipv4/ip_forward` ซึ่งค่าโดยคิฟอลท์ของระบบนั้นจะมีค่าเป็น “0” (ไม่พอเวอร์คแพ็กเก็ต) ซึ่งเราสามารถดูค่าของไฟล์นี้โดยใช้คำสั่งดังนี้

```
[root@router root]# cat /proc/sys/net/ipv4/ip_forward
```

ซึ่งถ้าเราทำการเปลี่ยนค่าในไฟล์นี้ให้เป็น “1” (พอเวอร์คแพ็กเก็ต) จะต้องใช้คำสั่งดังนี้

```
[root@router root]# echo "1" /proc/sys/net/ipv4/ip_forward
```

แต่ในการคอนฟิกค่าแบบนี้เมื่อเราทำการบูตเครื่องขึ้นมาใหม่จะทำให้ค่านั้นกลับมาเป็น 0 เหมือนค่าของคิฟอลท์ในระบบเหมือนเดิม ดังนั้นเราจะต้องทำการกำหนดค่านี้ในไฟล์ `/etc/sysctl.conf` เพื่อที่จะทำให้ค่าในไฟล์นี้เป็น “1” โดยถาวร โดยแก้ไขที่บรรทัด `net.ipv4.ip_forward = 1` ดังนี้

```
[root@router root]# cat /etc/sysctl.conf
#Kernel sysctl configuration file for Red Hat Linux
#For binary values, 0 is disabled, 1 is enabled. See sysctl(8) and sysctl.conf(5) for more details
#Controls IP packet forwarding
net.ipv4.ip_forward = 1
..
[various show message]
..
```

2.2 เว็บเซิร์ฟเวอร์ (Web Server)

การที่เราเข้าเยี่ยมชมเว็บไซต์ต่าง ๆ และทำการเรียกหน้าเว็บเพจของเว็บไซต์นั้นมาอ่านได้ ก็คือการอ่านไฟล์ HTML ที่อยู่บนเครื่องเว็บเซิร์ฟเวอร์นั่นเอง ดังนั้นเว็บเซิร์ฟเวอร์จะทำหน้าที่รอการติดต่อเข้ามาของบราวเซอร์เพื่อดูว่าบราวเซอร์นั้นต้องการเว็บเพจใด จะได้ส่งเว็บเพจดังกล่าวไปให้บราวเซอร์เพื่อแสดงผลให้ผู้ใช้งานเห็น นอกจากนี้เว็บเซิร์ฟเวอร์จะให้บริการอินเทอร์เน็ตแล้ว ยังสามารถนำมาประยุกต์ใช้ในเครือข่ายภายในองค์กรได้อีกด้วย

2.2.1 การติดตั้งเว็บเซิร์ฟเวอร์บนระบบปฏิบัติการลินุกซ์

บนระบบปฏิบัติการลินุกซ์นั้นมีโปรแกรมเว็บเซิร์ฟเวอร์ติดตั้งมาด้วย ซึ่งก็คือ Apache ซึ่งบริการของเว็บเซิร์ฟเวอร์บนระบบปฏิบัติการลินุกซ์นั้นจะเรียกว่า httpd ซึ่งมาจากคำว่า HTTP Daemon ซึ่งการติดตั้งนั้นเราจะต้องดาวน์โหลด Apache เวอร์ชันที่ไม่เกิน 1.3.xx เพราะจะต้องทำการตั้งค่าให้ Apache มีสิทธิเช่นเดียวกับ root ถ้าใช้เวอร์ชันใหม่ ๆ ต้องโหลด Patch มาเพิ่มเพื่อที่มีสิทธิเช่นเดียวกับ root ได้ซึ่งค่อนข้างยุ่งยาก โดยในโครงการนี้ใช้ "apache_1.3.33.tar.gz" ในการติดตั้งซึ่งมีวิธีการติดตั้งดังนี้

```
[root@router root]# tar xvfz apache_1.3.33.tar.gz
[root@router root]# cd apache_1.3.33
[root@router root]# env CFLAGS="-DBIG_SECURITY_HOLE" ./configure --prefix=/www
--enable-module=so (ทำการ compile และ config ให้รันเป็น user root ได้)
[root@router root]# make
[root@router root]# make install
```

ในโครงการนี้เราทำการติดตั้ง Apache ไว้ในไดเรกทอรี /www เมื่อติดตั้งเสร็จเรียบร้อยแล้วให้แก้ไขค่าคอนฟิกใน /www/bin/httpd.conf ดังนี้เพื่อที่จะให้ Apache รันเป็น root

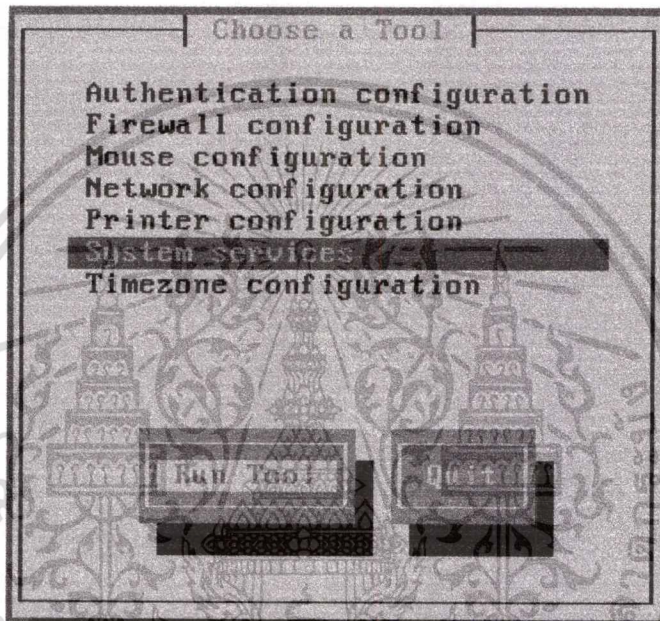
```
User root
Group root
```

เมื่อติดตั้งเสร็จเรียบร้อยแล้ว ให้ทำการรันเซอร์วิสก่อนเว็บเซิร์ฟเวอร์จึงจะทำงานได้

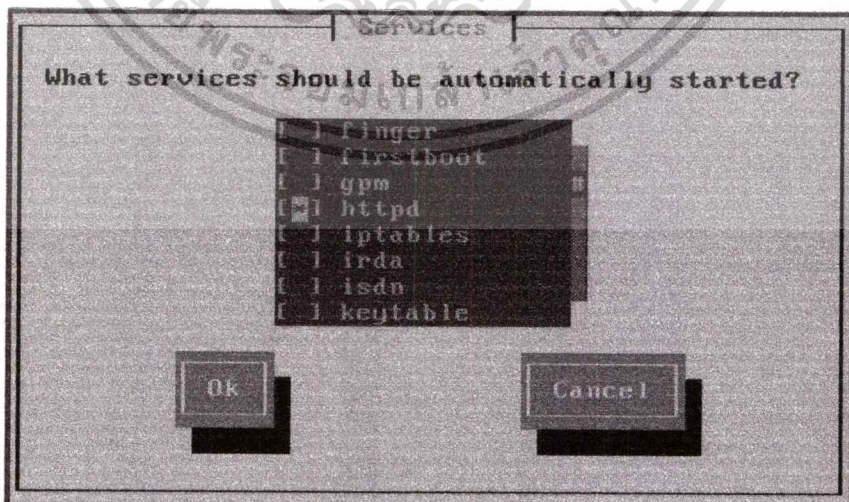
```
[root@router root]# /www/bin/apachectl start
Starting Apache: [OK]
```

แต่เมื่อทำการบู๊ตเครื่องอีกรอบเราจะต้องมาทำการรันเซอร์วิสอีกครั้งหนึ่ง ดังนั้นเราสามารถที่จะกำหนดให้เซอร์วิสทำงานทุกครั้งทีบู๊ตเครื่องได้ โดยการเรียกใช้คำสั่ง setup แล้วเลือก System service แล้วกาเครื่องหมาย * ข้างหน้า httpd

```
[root@router root]# setup
```



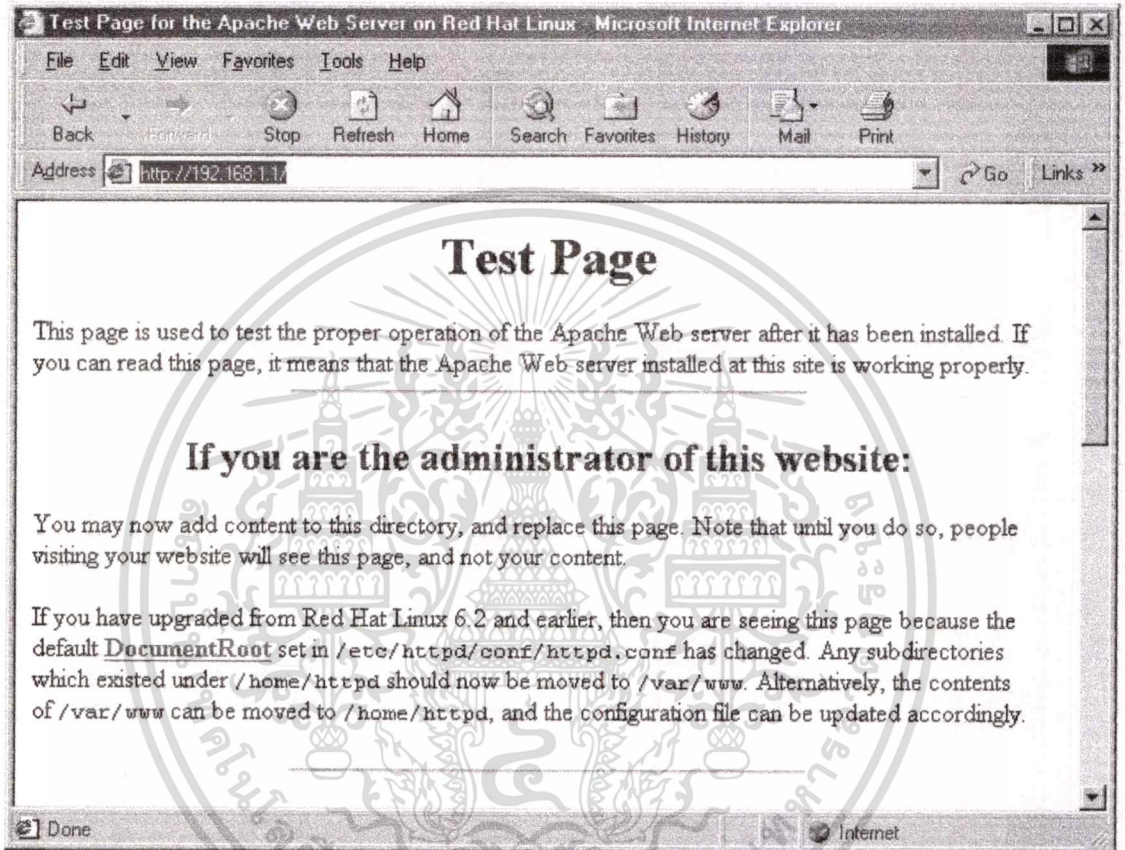
รูปที่ 2.1 หลังจากที่ใช้คำสั่ง setup แล้วคลิกเลือก System services



รูปที่ 2.2 หลังจากทีคลิกเลือก System services แล้วกาเครื่องหมาย * ข้างหน้า httpd

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นทำการทดลองดูว่าเว็บเซิร์ฟเวอร์ทำงานหรือยัง โดยการป้อน URL บนเครื่องไคลเอนต์ที่อยู่ในแลนเดียวกัน โดยอ้างไปที่ IP Address ของเครื่องเว็บเซิร์ฟเวอร์ อย่างเช่นในตัวอย่างพิมพ์ <http://192.168.1.1> จะปรากฏภาพที่บราวเซอร์ดังรูปที่ 2.3



รูปที่ 2.3 เรียกเว็บไซต์ Default ที่ติดมากับ httpd

2.3 HTML

HTML คือ เว็บเพจ เพื่อให้โปรแกรมเว็บเบราว์เซอร์ต่าง ๆ (ไม่ว่าจะเป็นโปรแกรม Internet Explorer, Netscape, Opera หรือเบราว์เซอร์ตัวอื่น ๆ) สามารถแปลงคำสั่ง และ แสดงผลเป็นรูปภาพ เสียง หรือข้อมูลได้ ภาษา HTML เป็นข้อความ (Text) ที่เป็นรหัสแอสกี (ASCII) ธรรมดา ๆ กับ รหัสที่อยู่ในเครื่องหมาย < > และมีนามสกุลเป็น *.html โดยเมื่อเราเปิดโปรแกรมเว็บเบราว์เซอร์ เราจะไม่สามารถพบรหัสเหล่านี้ได้เลยบนจอภาพ แต่รหัสเหล่านี้จะเป็นคำสั่งที่บอกโปรแกรมเว็บเบราว์เซอร์ ของเราว่า รูปแบบของข้อความเป็นอย่างไร ตัวหนา/เอียง หรือหัวข้อต่างๆ จะต้องมีการ โหลดรูปภาพกราฟฟิกหรือไม่ รวมไปถึงการสร้างจุดเชื่อมโยงหรือลิงค์ (Link) ที่เชื่อมโยงไปยังเว็บเพจอื่น ๆ

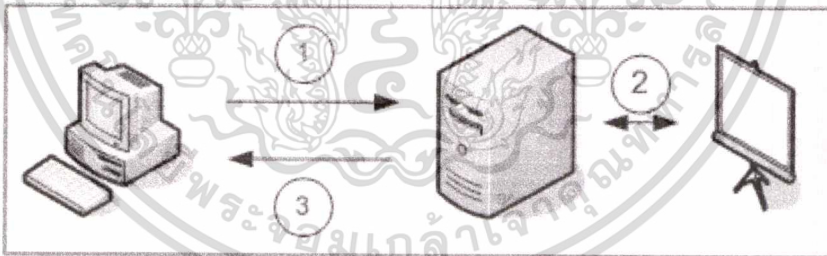
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 PHP

PHP ย่อมาจากคำว่า “Personal Home Page Tool” แต่ต่อมาได้เปลี่ยนชื่อให้ตรงตามกฎเกณฑ์ของ GNU ชื่อในปัจจุบันของ PHP นั้นย่อมาจาก “Hypertext Preprocessor” เป็นการเขียนคำสั่งหรือโค้ดโปรแกรมบนฝั่งเซิร์ฟเวอร์ (Server-Side Script) คือ มีการทำงานที่ฝั่งของเครื่องเซิร์ฟเวอร์ แล้วส่งการแสดงผลมายังเบราว์เซอร์ของตัวไคลเอนต์และนอกจากนี้ มันยังเป็นสคริปต์ที่ฝังบนตัวภาษา HTML อีกด้วย

PHP เป็นภาษาจำพวก Script Language คำสั่งต่าง ๆ จะเก็บอยู่ในไฟล์ที่เรียกว่า สคริปต์ (Script) และเวลาใช้งานต้องอาศัยตัวแปลชุดคำสั่ง ตัวอย่างของภาษาสคริปต์เช่น JavaScript, Perl เป็นต้น ลักษณะของ PHP ที่แตกต่างจากภาษาสคริปต์แบบอื่น ๆ คือ PHP ได้รับการพัฒนาและออกแบบมา เพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาโดยอัตโนมัติ ดังนั้นจึงกล่าวว่า PHP เป็นภาษาที่เรียกว่า Server-Side Script หรือ HTML - Embedded Scripting language เป็นเครื่องมือที่สำคัญชนิดหนึ่งที่ช่วยให้เราสามารถสร้างเอกสารแบบ Dynamic HTML ได้อย่างมีประสิทธิภาพมากขึ้น

2.4.1 หลักการทำงานของ PHP



รูปที่ 2.4 การทำงานของ PHP

- ขั้นตอนที่ 1 ฝั่งไคลเอนต์จะทำการร้องขอหรือเรียกใช้งานไฟล์ PHP ที่เก็บไว้ในเครื่องเซิร์ฟเวอร์
- ขั้นตอนที่ 2 ฝั่งเซิร์ฟเวอร์จะทำการค้นหาไฟล์ PHP แล้วทำการประมวลผลไฟล์ PHP ตามที่ไคลเอนต์ทำการร้องขอมา และทำการประมวลผลไฟล์ PHP
- ขั้นตอนที่ 3 ส่งผลลัพธ์จากการประมวลผลไปให้กับเครื่องไคลเอนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 การติดตั้ง PHP บนระบบปฏิบัติการลินุกซ์

การจะติดตั้ง PHP ให้ทำงานได้นั้นบนระบบปฏิบัติการลินุกซ์ติดตั้งบริการเว็บเซิร์ฟเวอร์ก่อนจึงจะสามารถทำการติดตั้ง PHP ลงไปได้ ซึ่งการติดตั้งนั้นสามารถดาวน์โหลดไฟล์จากอินเทอร์เน็ตได้ ซึ่งสามารถติดตั้งโดยใช้คำสั่งดังนี้

```
[root@router root]# tar xvfz php-4.3.10.tar.gz
[root@router root]# cd php-4.3.10/
[root@router root]# ./configure --with-mysql --with-apxs=/www/bin/apxs
[root@router root]# make
[root@router root]# make install
[root@router root]# cp php.ini-dist /usr/local/lib/php.ini
```

ซึ่งเมื่อทำการติดตั้งเสร็จจะต้องเพิ่มค่าในไฟล์ httpd.conf ด้วย

```
LoadModule php4_module libexec/libphp4.so
Addtype application/x-httpd-php .php .html
Addtype application/x-httpd-php-source .phps
```

เมื่อทำการติดตั้งเสร็จเรียบร้อยแล้วสามารถตรวจสอบการทำงานได้ โดยนำเว็บเพจที่เขียนด้วยภาษา PHP นำไปไว้ในไดเรกทอรี /www/htdocs แล้วลองทำการเรียกจากบราวเซอร์ดูว่าหน้าเว็บเพจนั้นสามารถประมวลผลได้หรือไม่

2.5 MySQL

MySQL เป็นฐานข้อมูลแบบ open source ที่ได้รับความนิยมในการใช้งานสูงสุดโปรแกรมหนึ่งบนเครื่องให้บริการ มีควมสามารถในการจัดการกับฐานข้อมูลด้วยภาษา SQL (Structures Query Language) อย่างมีประสิทธิภาพ มีความรวดเร็วในการทำงาน รองรับการทำงานจากผู้ใช้หลายๆ คนและหลายๆ งานได้ในขณะเดียวกัน ซึ่งสามารถดาวน์โหลดนำมาใช้งานได้ฟรี ซึ่งเราจะต้องดาวน์โหลดไฟล์มา 2 ตัวด้วยกันคือ MySQL-client-4.1.10-0.i386.rpm และ MySQL-server-4.1.10-0.i386.rpm

ซึ่งในการติดตั้งสามารถทำได้ดังนี้

```
[root@router root]# rpm -i MySQL-client-4.1.10-0.i386.rpm
[root@router root]# rpm -i MySQL-server-4.1.10-0.i386.rpm
```

โดยเมื่อทำการติดตั้งเสร็จแล้วก็จะสามารถใช้งานได้เลย โดยไม่ต้องคอนฟิกค่าใด ๆ อีก

2.6 Traffic Control Concept

Traffic Control เป็นชื่อที่ถูกตั้งให้กับระบบแถวคอย (Queuing System) หรือ กลไกการทำงานซึ่งทำการรับแพ็กเก็ตและทำการจัดการส่งแพ็กเก็ตนั้นออกไปบนอุปกรณ์เราเตอร์ และในกลไกการทำงานนี้จะประกอบไปด้วย การกำหนดอัตราความเร็วในการรับแพ็กเก็ตเข้ามา และอัตราความเร็วในการส่งแพ็กเก็ตนั้นออกไปด้วย

ในกลไกการทำงานส่วนใหญ่ของ Traffic Control จะประกอบไปด้วยแถวคอยหนึ่งแถวซึ่งจะคอยรับแพ็กเก็ตเข้าสู่แถวคอยและนำแพ็กเก็ตนั้นออกจากแถวคอยอย่างรวดเร็วที่สุด ซึ่งความเร็วในการนำแพ็กเก็ตออกจากขึ้นอยู่กับความสามารถของฮาร์ดแวร์นั้น ซึ่งประเภทของแถวคอยที่ใช้กันส่วนมานั้นจะใช้ FIFO (First in, First out) แพ็กเก็ตที่เข้ามาก่อนจะออกจากแถวคอยก่อน

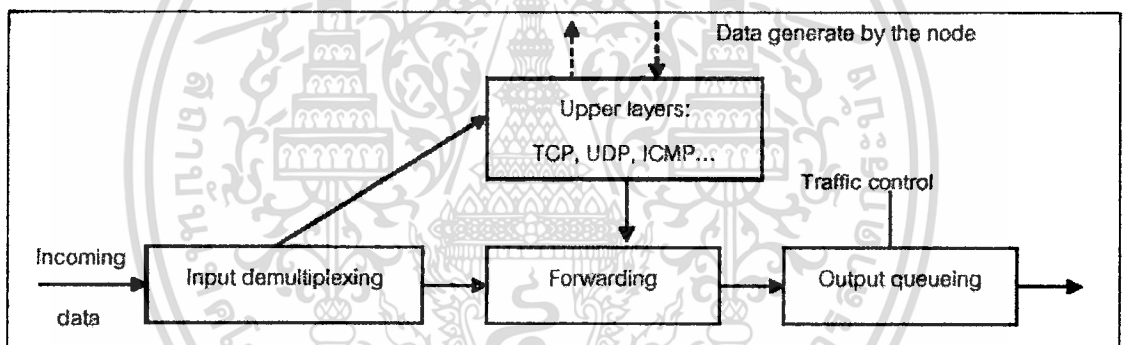
ซึ่งการใช้แถวคอยเป็นแนวทางที่จะช่วยในการจัดการเก็บงานหรือข้อมูล ที่ยังไม่มี การตัดสินใจจัดการอะไรไว้ก่อน เพราะข้อมูลที่ส่งเข้ามาอาจจะทำการประมวลผลไม่ทันจึงต้องทำการเก็บไว้ในแถวคอยก่อนแล้วค่อย ๆ นำมาจัดการเพื่อที่จัดส่งข้อมูลเหล่านั้นตามออกไป

อย่างเช่นในกรณีที่เครื่องเดสก์ทอปและเครื่องเว็บเซิร์ฟเวอร์นั้นใช้ช่องทางเดียวกันเพื่อที่จะออกไปสู่โลกอินเทอร์เน็ต ก็อาจจะมี การแข่งขันเพื่อแย่งชิงแบนด์วิธกันเกิดขึ้น เครื่องเว็บเซิร์ฟเวอร์อาจจะส่งข้อมูลออกจากเครื่อง (ภายนอกหรือขอข้อมูลเว็บเพจเข้ามา) มากกว่าแบนด์วิธที่ช่องทางสื่อสารขาออกมีอยู่ ในจุดนี้เองเราเตอร์จะเริ่มทำการทิ้งแพ็กเก็ตที่รับเข้ามา (ถ้าบัฟเฟอร์ในเครื่องเต็ม) ซึ่งในขณะนั้นเครื่องเดสก์ทอปซึ่งอาจจะมีการสื่อสารกับภายนอกอยู่ก็อาจจะต้องพบกับ การสูญหายของแพ็กเก็ตและเวลาแฝงในอัตราที่สูงมาก ดังนั้นเราจะทำการแบ่งแถวคอยภายในออกตามประเภทของบริการ (แอปพลิเคชัน) ทั้งสองชนิดออกจากกัน (เว็บแอปพลิเคชันและเครื่องเดสก์ทอป) เพื่อที่จะเป็นทางออกที่ดีสำหรับการแบ่งปันทรัพยากรของเครือข่ายให้กับแอปพลิเคชันทั้งสองชนิดนั้น

ในกรทำงานส่งข้อมูลของระบบเครือข่ายนั้นจะเป็นในลักษณะของ “Best effort” คือพยายามส่งให้ดีที่สุด ดังนั้นเครื่องหรืออุปกรณ์บนเครือข่ายถ้ามีการส่งข้อมูลเข้าไปในเครือข่ายจะทำการส่งข้อมูลให้มากที่สุดเท่าที่จะทำได้ ดังนั้นในระบบเครือข่ายจะเกิดการแข่งขันเพื่อที่จะใช้ทรัพยากรในเครือข่ายหรือแบนด์วิธนั่นเอง โดยถ้าเราไม่มีการจัดการในการควบคุม

Traffic Control ก็คือชุดการทำงานซึ่งจะอนุญาตให้ผู้ใช้ทำการแบ่งการควบคุมของแถวคอยออกเป็นส่วน ๆ หรือประเภทของการใช้งานบนอุปกรณ์เครือข่าย ซึ่งสามารถจัดการการไหลของทราฟฟิกและแพ็กเก็ตในเครือข่ายได้เป็นอย่างดี ซึ่งในบางครั้งคำว่า Quality of Service (QoS) ก็มักถูกใช้ในความหมายเดียวกับ traffic Control ด้วย

ซึ่งในระบบปฏิบัติการลินุกซ์นั้น จะมีชุดของกลไกการทำงานที่ใช้ในการจัดการและควบคุมทราฟฟิกในเครือข่ายติดมาด้วยกับระบบปฏิบัติการ ซึ่งเป็นบริการอย่างหนึ่งที่รู้จักกันดีของระบบปฏิบัติการตัวนี้ ซึ่งติดมากับเคอร์เนลในเวอร์ชัน 2.2 และ 2.4



รูปที่ 2.5 Network Packet Processing using Linux Traffic Control

จากรูปที่ 3.1 จะมีการแสดงให้เห็นว่าเคอร์เนลจะจัดการกับแพ็กเก็ตที่เข้ามาอย่างไร และมันจะทำการสร้างแพ็กเก็ตออกไปสู่ระบบเครือข่ายได้อย่างไร โดย Input De-Multiplexer จะทำการพิจารณา แพ็กเก็ตที่เข้ามาว่าอยู่ในขอบเขตของ Local node หรือไม่ ถ้าใช่มันจะส่งไปยังระดับชั้น Layer ที่อยู่สูงกว่า เพื่อทำ process ต่อไป ถ้าไม่มันจะทำการส่งแพ็กเก็ตไปที่ FORWARDING BLOCK ซึ่ง FORWARDING BLOCK อาจจะได้รับการสร้างแพ็กเก็ตจาก Layer ที่อยู่สูงกว่าจากนั้นแพ็กเก็ตเหล่านี้จะถูกส่งไปที่ Output Interface ซึ่ง ณ นี้จะมีการทำ Traffic Control โดยจะใช้ Queuing Discipline ในการควบคุม Packet เพื่อที่จะทำการส่งออกไปสู่ระบบเครือข่าย

Queuing Discipline นั้นเป็นชื่อเรียกของวิธีการที่ใช้จัดระเบียบในแถวคอย (queue) หรือนั่นก็คือ การทำ Traffic Control นั่นเอง บางครั้งเราก็ใช้ตัวย่อว่า “qdisc”

2.6.1 องค์ประกอบที่สำคัญของ Traffic Control

- Shaping (การจัดรูปแบบ)

เป็นกระบวนการอย่างหนึ่งซึ่งจะทำการหน่วงเวลาแพ็กเก็ต ก่อนที่จะทำการจัดส่งเข้าไปที่ แลวดคอกออกเพื่อที่จะให้ได้อัตราการส่งแพ็กเก็ตออกตามที่ต้องการ โดยมันจะพยายามที่จะจำกัด หรือแบ่งสรรทรัพยากรฟิกตามอัตราความเร็วที่ได้กำหนดไว้ (โดยปกติจะประมาณโดยวิธีคำนวณแบบ packets per second หรือ bits/bytes per second)

- Scheduling (การกำหนดรูปแบบของแพ็กเก็ต)

เป็นกระบวนการอย่างหนึ่งซึ่งแพ็กเก็ตจะทำการกำหนดรูปแบบระหว่างช่องทางเข้า (input) และช่องทางออก (output) ของแต่ละแลวดคอก ซึ่งโดยส่วนมากการกำหนดแผนงานหรือรูปแบบของ แพ็กเก็ตจะเป็นแบบ FIFO (first-in first-out) ซึ่งก็มีรูปแบบอื่น ๆ ขึ้นอยู่ว่าเราจะนำกระบวนการใด มากำหนดรูปแบบของแพ็กเก็ต ซึ่งในมุมมองกว้าง ๆ ชุดของกระบวนการควบคุมกราฟฟิกบน แลวดคอกนั้นจะถูกพิจารณาว่าเป็น “Scheduler”

- Classifying (การจัดแบ่งประเภทของแพ็กเก็ต)

เป็นกระบวนการอย่างหนึ่งซึ่งแพ็กเก็ตนั้นจะถูกทำการแยกเพื่อการดูแลที่แตกต่างกันไป อาจจะเป็นไปได้ในแลวดคอกที่แตกต่างกัน ซึ่งอุปกรณ์เครือข่ายสามารถที่จะจัดแบ่งประเภทของ แพ็กเก็ตตามเส้นทางที่แตกต่างกันได้ ซึ่งในการรูปแบบการทำงานของระบบปฏิบัติการลินุกซ์นั้น จะอนุญาตให้แพ็กเก็ตที่มีลำดับการทำงานที่ต่อเนื่องกันเป็นแพ็กเก็ตในประเภทเดียวกัน

- Policing (การจำกัดกราฟฟิกในแต่ละแลวดคอก)

เป็นกระบวนการอย่างหนึ่งซึ่งจะทำการจำกัดกราฟฟิกในแต่ละแลวดคอกเพื่อให้แน่ใจว่าจะ ไม่มีจุดใดบนเครือข่ายใช้แบนวิธ้มากเกินกว่าที่ได้กำหนดไว้ให้ ซึ่งจะทำการจัดการกับจุดนั้นที่ทำการใช้แบนวิธ้มากเกินไปโดยอาจจะทำการทิ้งแพ็กเก็ตนั้นไปเลย

2.6.2 องค์ประกอบต่าง ๆ ของ Traffic Control ใน Linux

- qdisc (queuing discipline)

บางทีอาจจะเรียกว่า Scheduler (ตัวกำหนดรูปแบบของแพ็กเก็ตในแฉวคอย) ซึ่งทุก ๆ อินเทอร์เฟซนั้นจะต้องมีตัว scheduler อยู่ซึ่งขึ้นอยู่กับว่าจะใช้ตัว Scheduler แบบใด ซึ่งโดยปกติค่าดีฟอลท์ของ Scheduler ในระบบปฏิบัติการลินุกซ์นั้นจะเป็นแบบ FIFO

classful qdisc จะประกอบไปด้วย class หรือก็คือ การจัดแบ่งประเภทของแพ็กเก็ตนั่นเอง และการจัดแบ่งประเภทของแพ็กเก็ตนั้นจะถูกดูแลด้วย filter

Classless qdisc นั้นจะไม่มี การจัดแบ่งประเภทของแพ็กเก็ต ซึ่งก็คือ ไม่มีการทำ classifying นั่นก็หมายความว่า จะไม่สามารถทำการเพิ่ม filters เข้าไปได้นั่นเอง

สำหรับคำศัพท์ที่อาจจะเกิดความสับสนที่ใช้กันใน queuing discipline คือ “root” และ “ingress” ซึ่งจริง ๆ แล้วมันไม่ใช่ queuing discipline แต่จะให้ความหมายว่า egress คือ ทราฟฟิกขาออก (outbound traffic) ส่วน ingress นั้นหมายถึงทราฟฟิกขาเข้า (inbound traffic)

ซึ่งแต่ละอินเทอร์เฟซจะประกอบไปด้วยทั้งสองตัวนี้ ซึ่งตัวหลัก ๆ ก็คือ “egress qdisc” (ตัวจัดระเบียบแฉวคอยทราฟฟิกขาออก) หรือจะรู้จักกันโดยทั่วไปว่า “root” มันสามารถที่จะประกอบไปด้วย qdisc ที่มี class และโครงสร้าง class อะไรก็ได้ โดยทราฟฟิกที่ส่งผ่านออกไปทางอินเทอร์เฟซนั้นก็คือ “egress” หรือ “root” นั่นเอง

สำหรับทราฟฟิกที่จะถูกยอมรับบนอินเทอร์เฟซนั้นจะต้องผ่าน ingress qdisc ไปก่อน ด้วยขอบเขตของความสามารถในการทำงานที่จำกัดกว่า egress qdisc มันจะไม่อนุญาตให้มีการสร้าง class ถูกขึ้น และจะมีเพียง filter ติดเข้ามาด้วยเท่านั้น ซึ่งสำหรับจุดประสงค์ในการใช้งานนั้น ingress qdisc จะมีความสะดวกในการใช้งานซึ่งจะติด policer เพื่อที่จะจำกัดทราฟฟิกในแต่ละแฉวคอยที่จะให้ได้ในเน็ตเวิร์กการ์ดนั้น

ซึ่งในการทำงานนั้น egress qdisc นั้นจะมีประสิทธิภาพมากในการควบคุมทราฟฟิก ซึ่งจะมีการทำงานที่ดีกว่า ingress qdisc ซึ่งมีความสามารถเพียงแค่ policer เท่านั้น

- Class (การจัดแบ่งประเภทของแพ็กเก็ต)

class นั้นจะมีอยู่เพียงใน classful qdisc เท่านั้น class นั้นจะมีความยืดหยุ่นมากคือ สามารถที่จะประกอบไปด้วย class ถูกจำนวนมาก ๆ หรืออาจจะจะมีเพียง class ถูกเดียวก็ได้ และในแต่ละ class นั้นก็จะไม่มีกฎเกณฑ์ที่จะติด filter เข้าไปด้วย

leaf class คือ class สุดท้ายใน qdisc มันจะประกอบไปด้วย qdisc (ค่าเริ่มต้นคือ FIFO) และจะไม่ประกอบไปด้วย class ลูกอีกแล้ว

- filter

เป็นส่วนประกอบที่มีความซับซ้อนในระบบควบคุมทราฟฟิกในลินุกซ์ ซึ่ง filter จะจัดเตรียมกระบวนการที่จะทำการประสานองค์ประกอบสำคัญต่าง ๆ ของการควบคุมทราฟฟิก สิ่งที่ได้เห็นได้ง่าย ๆ และชัดเจนคือการ classify แพ็กเก็ต Linux filter จะอนุญาตให้ผู้ใช้ทำการ classify แพ็กเก็ตเกิดในแถวคอยออกด้วย filter หลาย ๆ ตัวหรืออาจจะเพียง filter เดียวก็ได้

- filter นั้นจะต้องประกอบไปด้วย classifier
- filter นั้นอาจจะประกอบไปด้วย policer
- filter สามารถที่จะติดเข้าไปได้ที่ root qdisc ซึ่งเป็นที่แรกที่แพ็กเก็ตจะวิ่งเข้าไปที่แถวคอย หรืออาจจะติดเข้าไปที่ classes ของ root qdisc นั้นอีกทีก็ได้

- Classifier

เป็นเครื่องมือที่ถูกใช้เป็นส่วนหนึ่งของ filter เพื่อที่จะระบุถึงคุณลักษณะของแพ็กเก็ตหรือความหมายของแพ็กเก็ตนั้น และที่รู้จักกันโดยทั่วไปก็คือ u32 Classifier จะอนุญาตให้ผู้ใช้ทำการเลือกแพ็กเก็ตบนพื้นฐานของแอททริบิวต์ต่าง ๆ ของแพ็กเก็ต

- Policer

เป็นองค์ประกอบการทำงานส่วนหนึ่งในระบบควบคุมทราฟฟิกในลินุกซ์ซึ่งเป็นส่วนหนึ่งของ filter policer มักจะถูกใช้ในเหตุการณ์ที่มากกว่าหรือน้อยกว่าอัตราความเร็วที่กำหนด ซึ่ง policer จะไม่ทำให้เกิดการหน่วงเวลาของทราฟฟิก มันสามารถที่จะจัดการเพียงสถานการณ์พื้นฐานที่ได้กำหนดกฎเกณฑ์เอาไว้

- Drop

เป็นกระบวนการทำงานพื้นฐานของการควบคุมทราฟฟิกในระบบควบคุมทราฟฟิกในลินุกซ์ซึ่งก็คือส่วนหนึ่งของ policer ซึ่ง policer ที่ติดไปกับทุก ๆ filter จะต้องมีการบวนการ “drop” (ทิ้งแพ็กเก็ต) ในกรณีที่บัฟเฟอร์ที่มีอยู่ถูกใช้ไปจนหมดระบบจะทำการ drop แพ็กเก็ตทิ้งไป หรือในกรณีรูปแบบของแพ็กเก็ตนี้ไม่ตรงกับกฎเกณฑ์ที่ตั้งไว้ให้ผ่าน

- Handle

ในทุก ๆ class และ classful qdisc จะต้องการเลขที่ระบุเฉพาะภายใน โครงสร้างของ Traffic Control เลขที่ระบุเฉพาะ คือรู้ว่าเป็นการควบคุมดูแล (handle) และมีสมาชิกที่เป็นส่วนประกอบสองส่วน คือ เลขหลัก (major number) และ เลขรอง (minor number) ซึ่งเลขนี้สามารถกำหนดได้ โดยต้องไม่มีกฎเกณฑ์ในการกำหนด (มีค่าเป็นเท่าไรก็ได้) แต่ผู้ใช้จะต้องกำหนดให้มีความสอดคล้องกันด้วย

Major ตัวแปรนี้จะให้ความหมายถึงเคอร์เนล ซึ่งผู้ใช้สามารถกำหนดขึ้นเองก็ได้ แต่อย่างไรก็ตามส่วนที่ที่เกิดขึ้น (object) ในโครงสร้างของ Traffic Control ที่มีพ่อแม่ (parent) เดียวกันนั้น จะต้องใช้เลขหลัก (major) ตัวเดียวกัน สำหรับ object ก็ตามที่ติดเข้าไปโดยตรงที่ root โดยตรงจะมีเลขหลักเริ่มต้นที่เลข 1

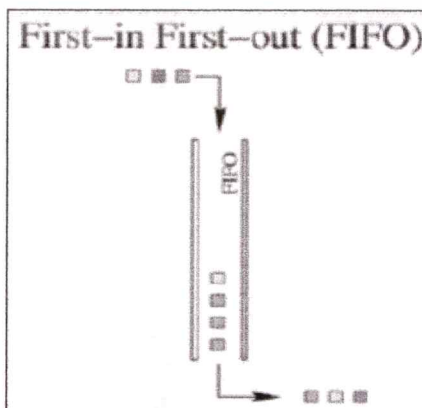
Minor ตัวแปรที่ระบุชัดเจนว่า object นั้นเป็น qdisc ถ้า minor นั้นเป็น 0 ถ้า minor เป็นค่าอื่น ๆ object นั้นจะเป็น class ทุก ๆ class ที่มีพ่อแม่เดียวกัน (parent) จะต้องมีเลข minor เฉพาะของมันเอง

2.6.3 Classless Queuing Disciplines

ในแต่ละ Classless qdisc นี้สามารถนำไปใช้เป็น qdisc หลักบนอินเตอร์เฟซได้ หรือสามารถนำไปใช้กับ leaf class ของ Classfull qdisc ได้

1. FIFO, First-in First-Out (pfifo and bfifo)

รูปแบบ FIFO อัลกอริทึม เป็นพื้นฐานสำหรับ default qdisc บน Linux Network Interface (pfifo_fast) มันจำไม่มีการจัดการ Shaping หรือ rearrange แพ็กเก็ต มันจะเป็นการจัดส่งแพ็กเก็ตแบบง่าย ๆ โดยการจัดส่งออกทันทีหลังจากที่รับแพ็กเก็ตเข้าไปในแถวคอยของมัน



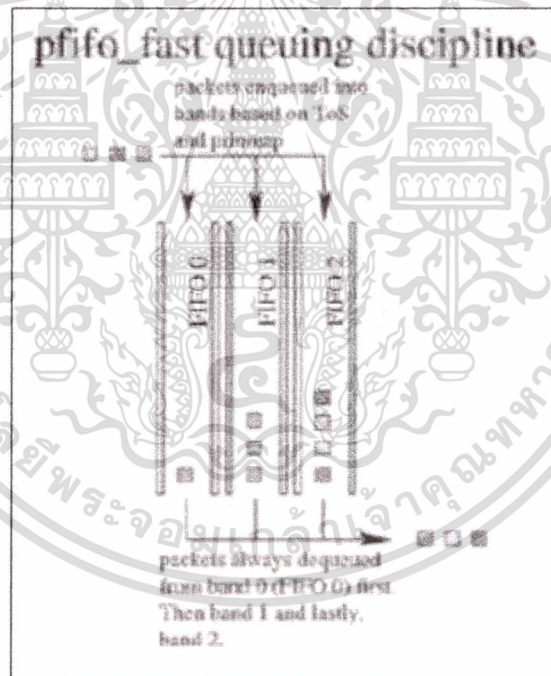
รูปที่ 2.6 การทำงานของ FIFO (First-in First-out)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่อย่างไรก็ตามในแต่ละแถวคอยแบบ FIFO qdisc นั้นก็จะมีขนาดที่จำกัด (ขนาดของบัฟเฟอร์) ซึ่งจะทำการป้องกันไม่ให้เกิดการล้นของแพ็กเก็ตในกรณีที่มันไม่สามารถนำแพ็กเก็ตออกจากแถวคอยได้เร็วพอในอัตราที่มันรับแพ็กเก็ตนั้นเข้ามา

2. pfifo_fast, the default Linux qdisc

pfifo_fast เป็น default qdisc ในทุก ๆ อินเทอร์เน็ตภายใต้ระบบปฏิบัติการลินุกซ์ ซึ่งมีพื้นฐานการทำงานมาจาก FIFO qdisc ซึ่งตัว pfifo_fast จะใช้การแบ่งแถวคอยออกเป็น 3 แถว (แถว 0, 1 และ 2) โดยแบ่งตามลำดับความสำคัญ (prioritization) ของแพ็กเก็ต ซึ่งในแต่ละแถวจะเรียกว่า “band” ซึ่งค่าลำดับความสำคัญนั้นจะดูจากฟิลด์ Tos (Type of Service) ในแต่ละแพ็กเก็ต ซึ่งถ้าแพ็กเก็ตใดมีค่าลำดับความสำคัญสูงก็จะถูกจัดให้อยู่ในแถวที่ 0 ซึ่งแพ็กเก็ตในแถวที่ 0 นี้จะถูกนำออกจากแถวคอยก่อนแถวคอยที่ 1 และ 2 ตามลำดับ



รูปที่ 2.7 การทำงานของ pfifo_fast

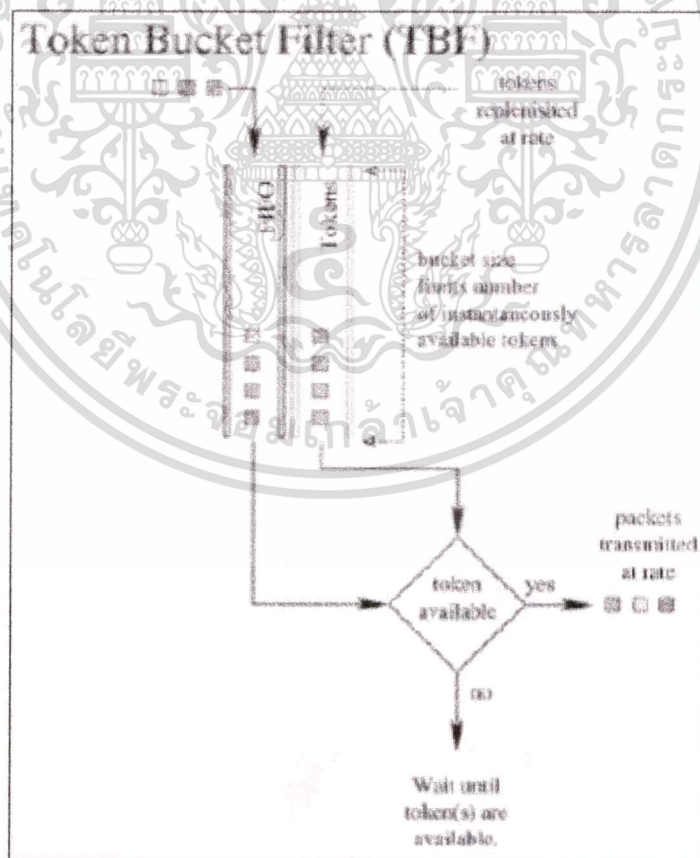
3. TBF, Token Bucket Filter

เป็นอัลกอริทึมที่นำมาใช้เพื่อปรับปรุงให้มีความอ่อนตัวมากขึ้น กลายเป็นวิธีที่เรียกว่า อัลกอริทึม Token bucket ซึ่งได้เพิ่มข้อกำหนดให้บัคเก็ตเก็บรักษาโทเก้นที่สร้างขึ้นตามจังหวะสัญญาณพิก้า ถ้าจำนวนแพ็กเก็ตมีจำนวนมากกว่าจำนวนโทเก้น แพ็กเก็ตที่เหลือจะต้องรอจนกว่าโทเก้นตัวใหม่จะเกิดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัลกอริทึมโทเคนบัคเก็ตมีวิธีจัดการนำส่งแพ็กเก็ตข้อมูลที่เข้ามาแบบ Burst ต่างจาก Leaky Bucket Algorithm ซึ่งจะไม่ยอมให้โฮสต์มีการเก็บสะสมเวลาที่ไม่ได้ใช้งาน (Idle time) เพื่อใช้สำหรับการส่งแพ็กเก็ตจำนวนมากในภายหลัง แต่โทเคนบัคเก็ตยอมให้มีการสะสมโทเคนได้ถึงขีดจำกัดอันหนึ่งที่กำหนดไว้ล่วงหน้า หมายความว่าโทเคนบัคเก็ตสามารถที่จะส่งแพ็กเก็ตติดต่อกันในทันทีทันใดได้เท่ากับจำนวนสูงสุดของโทเคนที่มีอยู่ในบัคเก็ต ทำให้รองรับการส่งข้อมูลได้สูงมากได้เป็นบางครั้งซึ่งจะมีผลตอบสนองโดยตรงต่อข้อมูลที่เข้ามาในแบบ Burst ได้เป็นอย่างดี ข้อแตกต่างอีกประการหนึ่ง คือ โทเคนบัคเก็ตไม่มีขีดจำกัดของแถวคอยเข้ามาเกี่ยวข้อง จึงไม่มีการลบแพ็กเก็ตข้อมูลที่ทิ้ง

Token Bucket Algorithm สามารถนำมาใช้ในการช่วยการจราจรระหว่างเราเตอร์สองตัวให้ราบรื่นขึ้น และยังช่วยจัดการส่งข้อมูลของโฮสต์ได้ อย่างไรก็ตามโทเคนบัคเก็ตสามารถบังคับให้โฮสต์หยุดส่งข้อมูลทั้งที่ต้องส่งหรือบังคับให้เราเตอร์หยุดส่งข้อมูลในขณะที่ข้อมูลจากแหล่งกำเนิดยังคงถูกส่งเข้ามาอย่างไม่ขาดสาย อันจะทำให้ข้อมูลสูญหายได้



รูปที่ 2.8 การทำงานของ Token Bucket Filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การคำนวณหาระยะเวลา Burst-out ที่เราเตอร์สามารถส่งข้อมูลด้วยอัตราความเร็วสูงสุด เป็นเรื่องที่ยากขึ้นซับซ้อน เนื่องจากในขณะที่มีการส่งข้อมูลอยู่นั้น โทกเก้นใหม่ก็จะถูกสร้างขึ้นมา ตลอดเวลา

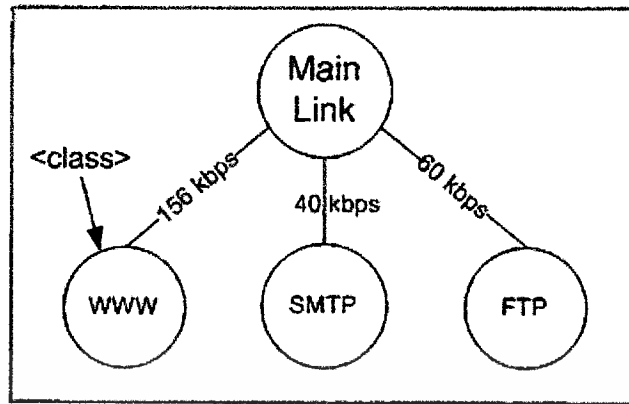
ปัญหาหลักของวิธีการ โทกเก้นบัคเก็ต ก็คือ การที่วิธีการนี้สนับสนุนการส่งข้อมูลแบบ Burst แม้ว่าจะสามารถควบคุมปริมาณและความถี่ได้ด้วยการกำหนดค่าอัตราการสร้างโทกเก้น และความเร็วสูงสุดในการส่งข้อมูลแล้วก็ตาม ระบบทั่วไปก็พยายามจะจำกัดจำนวนครั้งและความถี่ของการส่งข้อมูลที่มีความเร็วสูงสุด ในขณะที่ไม่ต้องการส่งข้อมูลที่มีความเร็วต่ำเสมอไป

2.6.4 Hierarchical Token Bucket (HTB)

เป็นรูปแบบหนึ่งของ queuing discipline ที่มีการทำงานในลักษณะที่เป็นโครงสร้างแบบ ต้นไม้และมีการจัดแบ่งประเภทในแต่ละโหนด ซึ่งรูปแบบการทำงานนั้นได้ถูกพัฒนามาจาก class base queuing discipline (CBQ) ซึ่งความสามารถในการทำงานนั้นค่อนข้างที่จะคล้ายคลึงกัน แต่เนื่องจาก CBQ มีความซับซ้อนและยากในการใช้งาน จึงได้มีการพัฒนา HTB ออกมา ซึ่ง HTB นั้น จะติดมากับแพ็คเกจ iproute2 ใน Linux Redhat ตั้งแต่เวอร์ชัน 9 ซึ่งเพียงแต่เราติดตั้งแพ็คเกจนี้โดยที่ไม่ต้องคอมไพล์เคอร์เนลใหม่ (ลินุกซ์เวอร์ชันเก่ากว่านี้ต้องคอมไพล์เคอร์เนลใหม่) ก็สามารถใช้งานความสามารถของ HTB ได้เลย โดยคอนฟิกผ่านทาง utility ที่ชื่อว่า "tc"

Link Sharing

HTB จะมีกระบวนการทำงานในลักษณะที่เป็นโครงสร้างคล้ายต้นไม้ โดยโหนดบนจะเหมือนกับจุดเชื่อมต่อสู่ภายนอก และกระจายแบนวิดท์ที่โหนดตัวเองมีอยู่ให้กับ class (ใน HTB จะใช้คำว่า class ในโหนดที่ไม่ใช่โหนดบนสุด และเนื่องจากในแต่ละโหนดก็จะเป็นบริการหรือการใช้งานแต่ละประเภทกันด้วย) ที่อยู่ต่ำกว่า โดยสามารถที่จะกำหนดจำนวนแบนวิดท์ให้กับแต่ละ class ในอัตราที่ต่างกันได้



รูปที่ 2.9 ลักษณะ โครงสร้างของ Link Sharing

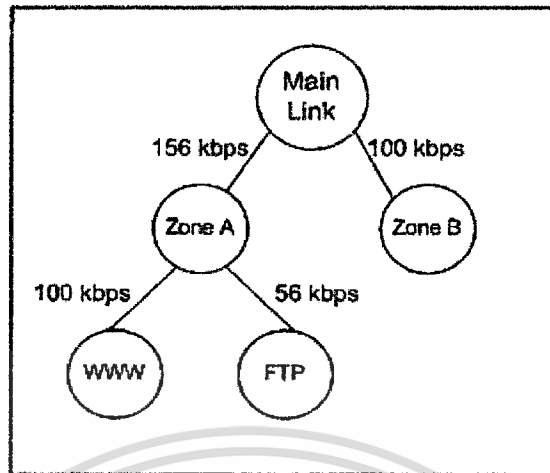
ในช่วงเวลาใดก็ตามที่ class ใดร้องขออัตราของแบนวิดธ์น้อยกว่าที่ได้จัดสรรไว้ให้ แบนวิดธ์ที่เหลืออยู่ใน class นั้นก็จะถูกกระจายให้กับ class อื่น ๆ ที่ทำการร้องขอมา โดยที่ class ที่ทำการร้องขอแบนวิดธ์ส่วนเกินนี้ จะต้องมีการกำหนดให้สามารถใช้แบนวิดธ์ส่วนเกินนี้ด้วยไม่อย่างนั้นก็ไม่สามารถใช้แบนวิดธ์ส่วนเกินนี้ได้ เรียกวิธีการนี้ว่า “borrowing” (การขอยืม)

เราจะสามารถกำหนดคลาสได้มากที่สุดเท่าไรนั้น ไม่มีข้อกำหนดที่แน่นอนว่าจะได้เท่าไร แต่จากการทดลองพบว่าเมื่อทำการเพิ่มคลาสจะได้มากที่สุดเพียง 9999 คลาส โดยการใช้ tc command นั้นจะฟ้อง error เมื่อทำการเพิ่มคลาส ที่ classid เกิน 19,999 แต่เมื่อมาดูค่าที่แสดงนั้น จะมี classid 9999 ที่แสดงอยู่เท่านั้น ดังนั้นจึงสรุปได้ว่า ถ้าทำการเพิ่มในช่วง classid 10,000-19999 จะไม่เกิด error แต่ค่า classid จะตัดออกให้เหลือแค่ 4 หลักหลังเท่านั้น เช่น 10000 จะเป็น 0000 หรือ 0 นั่นเอง ส่วนถ้าเป็น 12000 ก็จะเป็น 2000 นั่นเอง

เมื่อทำการสร้าง class ขึ้นมานั้นก็ต้องกำหนด qdisc ในแต่ละ class ซึ่งค่าดีฟอลท์ของ qdisc ในแต่ละ class นั้นจะเป็นแบบ pffifo

Sharing hierarchy

HTB ในรูปแบบนี้จะมีลักษณะที่เป็นโครงสร้างแบบต้นไม้ที่ไม่มีลำดับชั้นซึ่งจะแตกต่างกับ Link Sharing ซึ่งมีลักษณะเป็นแบบ flat คือมีแค่ลำดับชั้นเดียว ซึ่งรูปแบบการทำงานในแบบนี้จะช่วยให้เราสามารถสร้างรูปแบบการทำงานที่ซับซ้อนได้ง่ายขึ้น โดยในการกำหนดระดับชั้นนั้นสามารถกำหนดได้มากที่สุด 8 ระดับ



รูปที่ 2.10 ลักษณะ โครงสร้างของ Sharing hierarchy

Rate ceiling

ceil คือค่าที่ใช้กำหนดค่าในแต่ละ class ว่าสามารถใช้แบนวิดท์ได้มากที่สุดเท่าไร ซึ่งแบนวิดท์ที่ใช้ได้นั้นก็จะมาจากการยืมมา “borrowing” หรืออาจจะกำหนดให้ class นั้นใช้ได้เท่ากับแบนวิดท์ที่กำหนดให้กับ class นั้นตอนแรก

อย่างเช่นในกรณีที่เรากำหนดให้ class (Zone A) มีแบนวิดท์ที่ 156 kbps และมี ceil 256 kbps class (Zone A) จะสามารถใช้แบนวิดท์ได้มากที่สุดที่ 256 kbps แต่จะใช้ได้ในอัตรานี้ก็ต่อเมื่อ class (Zone B) ไม่ได้ใช้แบนวิดท์ 100 kbps นี้ยู่เท่านั้น แต่ถ้าเรากำหนดให้ ceil มีค่า 156 kbps class (Zone A) ก็จะสามารถใช้แบนวิดท์ได้สูงสุดแค่ 156 kbps ถึงแม้ class (Zone B) จะไม่ได้ใช้แบนวิดท์ของตัวเองอยู่ก็ตาม

Priorizing bandwidth share

การกำหนดลำดับความสำคัญให้กับ class ที่สร้างขึ้น โดยที่ class ใดต้องการขอขืมแบนวิดท์จาก class ที่ไม่ได้ใช้ class ที่มีค่าลำดับความสำคัญสูงกว่า จะสามารถนำแบนวิดท์นั้นไปใช้ก่อน class อื่น ๆ โดยการกำหนดลำดับในการขืม (priority) นั้นสามารถกำหนดระดับได้มากที่สุด 8 ระดับ มีค่า 0 ถึง 7 โดยค่า 0 นั้นเป็นค่าที่ลำดับความสำคัญมากที่สุด

2.7 L7 Netfilter

เป็นตัวจัดแบ่งประเภท (Classifier) ใน Linux kernel's Netfilter subsystem เพื่อที่จะระบุประเภทของบริการของแพ็กเก็ตในระดับ Application Layer (OSI Layer 7) นั่นก็คือสามารถกรอง (filter) แพ็กเก็ต เช่น HTTP , FTP , Kazaa ถึงแม้ว่าบริการนั้นจะทำการเปลี่ยนพอร์ตบริการก็ตาม ซึ่งความสามารถนี้ในตัวกรอง (filter) ใน Traffic control ในลินุกซ์ไม่สามารถทำได้ โดยตัว L7 Netfilter นั้นเป็น Patch เสริมของ kernel และ iptable(Firewall) ในตัวระบบปฏิบัติการลินุกซ์

ความสามารถโดยรวม

- สนับสนุนเคอร์เนลลินุกซ์ 2.4 และ 2.6
- สนับสนุนโปรโตคอล TCP,UDP และ ICMP บน IPv4
- ความยืดหยุ่นหรือสามารถเขียน Patch เพื่อความสามารถในการกรองแพ็กเก็ตได้

โดยเทคนิคที่ใช้ในการกรองแพ็กเก็ตของ L7 Netfilter คือ Regular Expression ซึ่งจะเป็นการกำหนดรูปแบบเพื่อการค้นหาข้อความหรือตัวอักษรว่า มีอยู่ในข้อความที่กำหนดหรือไม่ เช่น เราอยากทราบว่า ข้อความที่มีคนกรอกแบบฟอร์มเข้ามาบนเว็บของเรา มีคำหยาบหรือไม่ เราก็จะใช้ Regular expression เป็นตัวตรวจสอบ ถ้าไปถึงโปรแกรมพิมพ์เอกสาร ที่มีฟังก์ชันให้เราสามารถค้นและแก้ไขคำที่พิมพ์โดยการสั่ง replace ก็พอจะเข้าใจ concept ของ Regular expression แล้ว เพียงแต่ว่า Regular expression ทำได้มากกว่าเยอะ นักเขียนโปรแกรมบนเว็บส่วนมากใช้ Regular expression เพื่อตรวจสอบ email เบื้องต้น ว่า ผู้ใช้พิมพ์ email เข้ามาถูกรูปแบบหรือไม่ บางครั้งก็ใช้ในการตรวจสอบรูปแบบ เช่น หมายเลขประจำตัวนักศึกษา ซึ่งมีรูปแบบแน่นอน ถ้าพิมพ์มาไม่ถูกรูปแบบก็แสดงว่าพิมพ์ผิด เป็นต้น

โดยจะยกตัวอย่างสัญลักษณ์ที่ใช้ในการตรวจสอบได้ดังนี้

สัญลักษณ์บางส่วนของ regular expression

^	คำ/อักษรที่อยู่หน้าเครื่องหมายนี้ ต้องเป็นคำขึ้นต้นของข้อความที่นำมาตรวจสอบเช่น “การ” เป็นการกำหนดว่า คำที่นำมาตรวจสอบต้องขึ้นต้นด้วยคำว่า การ เช่น “การทำดี” “การบ้าน” เป็นต้น คำพวกนี้จะผ่านการทดสอบ
\$	คำ/อักษรที่อยู่หน้าเครื่องหมายนี้ ต้องอยู่ตอนท้ายของข้อความที่นำมาตรวจสอบ เช่น “มา\$” จะถือว่าคำต่อไปนี้ถูกตามเงื่อนไข “ตามมา” “ขอขมา” หรือแม้แต่คำว่า “หมา” แต่คำว่า “ทำดี” จะไม่ผ่าน เพราะไม่ได้ลงท้ายด้วยคำว่า “มา” ตามเงื่อนไขนั่นเอง

+	คำ/อักษรที่อยู่หน้าเครื่องหมายนี้ ต้องมีปรากฏในคำที่นำมาตรวจสอบ อย่างน้อย 1 ตัว เช่น “ท+” จะถือว่าคำต่อไปนี้ผ่านการตรวจสอบ เช่น “ทองจุด” “วันทนา” “ถนนหนทางทุกแห่ง”
?	คำ/อักษรที่อยู่หน้าเครื่องหมายนี้ อาจจะมีปรากฏในคำที่นำมาตรวจสอบ หรือไม่ก็ได้ ถ้ามีจะมีกี่ตัวก็ได้ “ก?ข+S” หมายถึง อาจจะมีด้วยตัว ก และอักษรตัวสุดท้ายต้องมีตัว ข อย่างน้อย 1 ตัว (เครื่องหมาย + แสดงว่ามีอย่างน้อย 1 และ เครื่องหมาย \$ แสดงว่าเป็นตัวสุดท้าย)
*	เหมือนกับ ?
\S	ช่องว่าง หรือ whitespace
.	ใช้แทนตัวอักษรอะไรก็ได้ “ก.[0-9]” หมายถึง ตัว ก ตามด้วยตัวอักษรอะไรก็ได้ และต่อด้วยเลขอารบิก เลข 0-9 “^.{3}\$” หมายถึง ต้องมีตัวอักษรเพียง 3 ตัวเท่านั้น เป็นตัวเลข ตัวอักษร ภาษาไทย ภาษาอังกฤษ ได้ทั้งนั้น
[]	ใช้ระบุตำแหน่งในคำว่า ในตำแหน่งนี้จะมีตัวอักษรอะไรได้บ้าง เช่น “[นร]” เป็นการกำหนดว่า คำที่นำมาตรวจสอบ ต้องเป็นตัว น หรือ ตัว ร เท่านั้นจึงจะผ่าน มีความหมายเช่นเดียวกับ “น ร” “[ก-ค]” เป็นการบอกว่า คำที่นำมาจะต้องเป็น ตัว ก ข ค เท่านั้น เช่น ในกรณีเลขประจำตัวที่ขึ้นต้นด้วย ก ข หรือ ค เท่านั้น ถ้าพิมพ์ตัวแรกเป็นตัวอักษรตัวอื่นก็แสดงว่าพิมพ์ผิด เราจะเขียนได้ดังนี้ ^[ก-ค] “^[a-zA-Z]” เป็นการบอกว่า คำที่นำมาตรวจสอบต้องขึ้นต้นด้วยตัวอักษร จะเป็นตัวเล็กคือ a ถึง z หรือ ตัวใหญ่ คือ A ถึง Z ก็ได้ “[0-9๐-๕]” เป็นการบอกว่า ให้มีตัวเลข 1 ตัว เลขอะไรก็ได้ เลข 0 ถึง เลข 9 เป็นได้ทั้งเลขไทยและอารบิก ต่อด้วยเครื่องหมาย % [ก-๕] ตัว ก ถึง ฮ รวมทั้งสระทุกตัว และ ตัวเลขไทย ๐ ถึง ๕ [0-9๐-๕] เลข 0-9 ทั้งเลขไทยและฝรั่ง ^[0-9๐-๕]+\$ ให้มีเฉพาะตัวเลข 0-9 เลขไทยหรือเลขฝรั่งก็ได้ แต่ห้ามมีตัวอักษรใด ๆ “^[กข]{3}-[0-9]” ขึ้นต้นด้วยตัว ก หรือ ข จำนวน 3 ตัว ต่อด้วยเครื่องหมาย - และจบด้วยตัวเลขอารบิก เลข 0-9 เช่น “กขก-5” “กขค-3” เป็นต้น สิ่งต่อไปนี้จะไม่ผ่านหรือเป็นเท็จ เช่น “กขกขข” เพราะ ตัวที่ 4 ไม่ใช่เครื่องหมาย - และตัวสุดท้ายไม่ใช่ตัวเลข

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของกรมส่งเสริมการค้าระหว่างประเทศ กระทรวงพาณิชย์ หากมีข้อผิดพลาดประการใดขออภัยเป็นอย่างสูง

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	<p>“xx-” ตัวเลขสุดท้ายเป็นเลขไทย ไม่ว่าตัวอักษร หรือสัญลักษณ์ใด ๆ ที่อยู่ภายในเครื่องหมาย [] จะกลายเป็นสัญลักษณ์ธรรมดา เช่น + กลายเป็นเครื่องหมายบวก แทนที่จะหมายถึงว่า ต้องมีตัวอักษรอย่างน้อย 1 ตัว</p>
{}	<p>แสดงจำนวนครั้งที่ซ้ำกัน เช่น “กx{2}” หมายถึงให้มีตัว ก จำนวน 2 ตัว เช่น “กขข” “กx{2,}” หมายถึงให้มีตัว ก อย่างน้อย 2 ตัว เช่น “กขขข” “กx{3,5}” หมายถึงให้มีตัว ก จำนวน 3-5 ตัวเท่านั้น คือ “กขขข” “กขขขข” และ “กขขขขข”</p>
()	<p>ใช้รวมกลุ่มเข้าด้วยกันเป็นส่วนเดียวกัน เช่น “ก(ขค)*” หมายถึง ตัว ก และอาจจะตามด้วยตัว ขค หรือ ไม่มีตัว ขค ก็ได้ เครื่องหมาย * แสดงว่าจะมีหรือไม่มีก็ได้ “ก(ขค){1,5}” หมายถึง ตัว ก แล้วจะตามด้วย ขค จำนวน 1-5 ชุด เช่น “กขคขคขค” หรือ “กขคขค” ก็ได้</p>
	<p>เสนอทางเลือกอย่างใดอย่างหนึ่ง เช่น “การ ความ” เป็นการบอกว่า จะใช้คำว่า การ หรือ ความ ก็ได้ “(ก ขค)ง” เช่น กงจ หรือ ขคจ ก็ได้</p>

ตัวอย่างการตรวจสอบ pattern ของ FTP ซึ่งสามารถเปลี่ยนพอร์ตในส่วนของ data-connection เป็นพอร์ตอื่นได้ ในรูปแบบของ passive mode

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040	 	Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	&	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	((72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051))	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[[123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174	|	
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135]]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177		DEL

Source: www.LookupTables.com

รูปที่ 2.11 ตาราง ascii table ใช้เปรียบเทียบค่าในการอ่าน Regular Expression

Pattern FTP: ^220[x09-\x0d ~]*ftp

- ต้องขึ้นต้นด้วยค่า 220 เท่านั้น(^220)
- ส่วน x09 เป็นเลขฐาน 16 ที่มีความหมายว่า TAB หรืออาจจะเรียกพวกนี้ ตัวอักษรที่ไม่สามารถพิมพ์ได้ (not printed) ซึ่งค่าที่อยู่ตั้งแต่ 09 - 0d นั้นเป็นพวกค่าที่เรียกว่า "whitespace" นั่นเอง
- [~] หมายถึง ตัวอักษรที่สามารถพิมพ์ได้
- [~] หมายถึง ตัวอักษรที่สามารถพิมพ์ได้รวมทั้งช่องว่างด้วย
- [x09-\x0d ~]* หมายถึง ทุกตัวอักษรที่สามารถพิมพ์ได้รวมทั้ง 'whitespace'
- ftp หมายถึง ว่าต้องมีคำว่า ftp
- ดังนั้นจะสรุปได้ว่า คำที่นำมาตรวจสอบนั้นจะต้องขึ้นต้นด้วย 220 เท่านั้น และเมื่อนั้นจะต้องตามด้วยตัวเลขตัวใดก็ได้ที่สามารถพิมพ์ได้ และหรือ 'whitespace' ตัวใดก็ได้ และจะต้องจบด้วย คำว่า ftp

Pattern bittorrent: `^\x13bittorrent protocol` (bittorrent เป็นโปรแกรม P2P คล้ายกับ Kazaa) อาจจะมี `(\)` และต้องต้องขึ้นต้นด้วย `x13 (^)` และตามด้วยคำว่า `bittorrent protocol`
`0x13` is the length of "bittorrent protocol"

ซึ่งในการติดตั้งจะต้องทำการโหลด Layer 7 patches และ Protocol Definition มาโดยตัว Layer 7 patches จะเป็นตัวที่เพิ่มความสามารถทำให้ iptables (firewall) มีความสามารถในการตรวจสอบแพ็กเก็ตเกิดในระดับ Application Layer (OSI Layer7) ซึ่งจะต้องทำการคอมไพล์เคอร์เนลใหม่ด้วย ก่อนจะทำการคอมไพล์เคอร์เนลจะต้องทำการแก้ไขค่าบางส่วนก่อนทำการคอมไพล์เคอร์เนล ดังนี้

เลือก EXPERIMENTAL (Code maturity level options → Prompt for development and/or incomplete code/drivers).

เลือก Netfilter (Device Drivers → Networking support → Networking Options → Network packet filtering).

หลังจากนั้นเลือก connection tracking (Network packet filtering → IP: Netfilter Configuration → Connection tracking).

ที่หน้าจอเดิมเลือก "Connection tracking flow accounting" and "IP tables support".

และเลือก "Layer 7 match support".

เมื่อทำการคอนฟิกค่าต่าง ๆ เรียบร้อยแล้วจึงค่อยทำการคอมไพล์เคอร์เนลและทำการรีสตาร์ทเครื่อง

หลังจากคอมไพล์เคอร์เนลแล้วจะต้องลง Protocol Definition เพื่อที่จะใช้รูปแบบ (pattern) ในนี้สำหรับตรวจสอบค่าในแพ็กเก็ตที่ผ่านเข้ามา ซึ่งรูปแบบนั้นก็คือ Regular Expression นั่นเอง โดยตัวที่ทำการตรวจรูปแบบ ก็คือ ตัว iptables นั่นเอง ซึ่งถ้าพูดง่าย ๆ ก็คือ iptables จะเป็นตัว filter ตัวหนึ่งที่ใช้ในการกรองแพ็กเก็ตนั่นเอง ซึ่งจะมีความสามารถในการกรองมากกว่าในตัว Traffic Control มีอยู่

ตัว iptables จะทำงานกับ 3 ตารางหลัก ๆ คือ (Filter, NAT, Mangle) ซึ่งในการทำงาน L7 Netfilter นั้น จะใช้ตาราง Mangle ซึ่งเป็นตารางสำหรับแก้ไขข้อมูล TOS, TTL, MARK ของแพ็กเก็ต ซึ่งในส่วนนี้จะทำการแก้ไขข้อมูล MARK เพื่อที่จะให้ตัว Iproute2 ทำการตัดสินใจในการเลือกเส้นทางโดยใช้ฟิลด์ MARK นี้เป็นตัวกำหนด

2.8 Tc Language

ชุดคำสั่งใน tc นั้นจะเป็นคำสั่งในรูปแบบ command line ซึ่งจะสามารถอธิบายคำสั่งหลัก ซึ่งเกี่ยวกับการสร้าง qdisc แบบ HTB ได้ดังนี้

เริ่มต้นต้องทำการกำหนด Queue Discipline ที่อินเทอร์เฟซก่อนโดยเลข 1:0 นั้น ถ้าจะเปรียบเทียบกับก็คือ x:y ซึ่ง x คือเลขหลักที่ระบุถึงตัว qdisc นั้นเอง ซึ่งค่า x จะต้องมีค่าเป็น 1 เท่านั้น

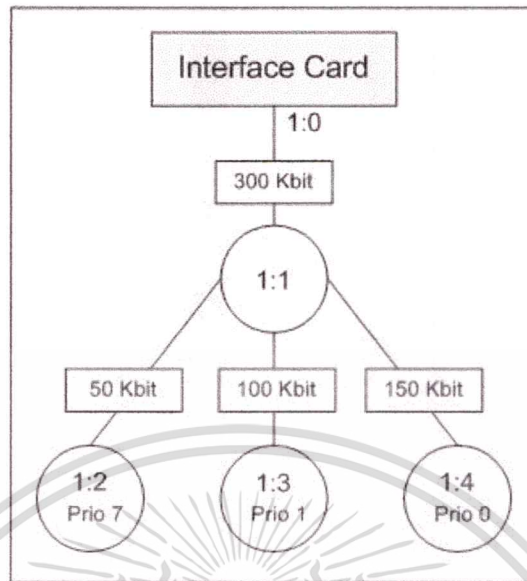
ส่วนค่า y จะหมายความถึงค่าที่บ่งบอกถึงความเป็นสมาชิกใน qdisc นั้น ๆ ซึ่งค่า y นั้นจะไม่ซ้ำกัน ซึ่งเมื่อเราสร้าง class ถัดลงไปก็จะเพิ่มเลขขึ้นเรื่อย ๆ ซึ่งคำสั่งในการสร้าง qdisc ที่ Interface จะอยู่ที่บรรทัดแรก โดยกำหนดให้เป็นแบบ HTB ส่วน default 2 หมายถึง ค่าคลาสดีฟอลท์ จะเป็นคลาสที่สร้างขึ้นเพื่อให้ทราฟฟิกที่ไม่ match กับคลาสใด ๆ เลขที่สร้างขึ้นให้หลังที่คลาสนี้

ข้อสังเกต : ใน tc นั้น 1kbps จะเท่ากับ 8kbit ดังนั้นในการคอนฟิกควรใช้หน่วยเป็น kbit หรือ Mbit

tc qdisc add dev eth0 root handle 1:0 htb default 2	(1)
tc class add dev eth0 parent 1:0 classid 1:1 htb rate 300kbit ceil 300kbit	(2)
tc class add dev eth0 parent 1:1 classid 1:2 htb rate 50kbit ceil 50kbps prio 7	(3)
tc class add dev eth0 parent 1:1 classid 1:3 htb rate 150kbps ceil 150kbps prio 0	(4)
tc class add dev eth0 parent 1:1 classid 1:4 htb rate 100kbps ceil 100kbps prio 1	(5)

ในบรรทัดที่สอง นั้นจะเป็นการสร้างคลาสแรกของ qdisc นั้นแต่ในที่นี้เราจะจำลองให้เหมือนกับว่าเป็น root class โดยกำหนดให้มีคลาสนี้มีอัตราที่ 300 kbit และสามารถเร็วสุดได้ที่ 300 kbit โดย parent จะเป็นการบอกถึงพ่อแม่ในลำดับชั้นซึ่งก็คือ 1:0 นั้นเอง โดยค่า classid จะมีค่าเพิ่มขึ้นเป็น 1:1 ซึ่งค่า 1 ข้างหลังก็คือค่า y นั้นเองซึ่งจะเพิ่มขึ้นเรื่อย ๆ

ส่วนบรรทัดที่ 3 – 5 นั้นจะเป็นคลาสลูกของ classid 1:1 อีกที่หนึ่งโดยสามารถกำหนดอัตรา rate และ ceil ได้แล้วแต่ต้องการแต่ทั้งสองค่านั้นจะไม่สามารถเกิน ค่า ceil ของ class 1:1 ได้



รูปที่ 2.12 รูปจำลองจากการคอนฟิกค่าใน tc utility

โดยค่า Prio นั้นจะเป็นค่าถึงการลำดับสิทธิ์ในการขอยืมแบนวิดท์ ซึ่งจะยกตัวอย่างว่าถ้า class 1:3 และ class 1:4 ใช้งานอยู่นั้นจะมีแบนวิดท์ ของ class 1:2 เหลืออยู่ซึ่ง class 1:4 ที่มี prio 0 (ค่า prio ต่ำมีสิทธิ์สูงที่สุด) จะได้แบนวิดท์นั้นไปใช้ก่อน

2.8.1 filter with Iptables

เมื่อเราทำการสร้าง class ต่าง ๆ เรียบร้อยแล้วเราก็ต้องมีตัวจัดการแบ่งประเภททราฟฟิก หรือการสร้างเงื่อนไขของประเภทของทราฟฟิกในแต่ละคลาสนั้นเอง ซึ่งในที่นี้เราจะเรียกว่าตัว filter นั้นเอง ซึ่งจะเป็นตัวที่สร้างเงื่อนไขในการกำหนดว่าทราฟฟิกใดจะอยู่ในคลาสนั้นเอง เพื่อที่จะกำหนดอัตราความเร็วของทราฟฟิกประเภทนั้นนั่นเอง

โดยตัวฟิลเตอร์นั้นเราจะเลือกใช้ iptable เข้ามาช่วยเสริมการทำงาน และด้วยในเหตุที่เรานำการฟิลเตอร์แพ็กเก็ตในระดับ Layer7 มาใช้งานด้วยซึ่งต้องใช้ผ่าน iptable จึงทำให้เราหลีกเลี่ยงการใช้ iptable ไม่ได้เลย ซึ่งการฟิลเตอร์จะมีคำสั่งดังนี้

```
Iptables -A POSTROUTING -t mangle -o eth1 -d 192.168.1.0/24 MARK --set-mark 1
```

```
Iptables -A POSTROUTING -t mangle -o eth1 -d 192.168.2.0/24 MARK --set-mark 2
```

```
tc filter add dev eth1 parent 1:0 handle 1 fw flowid 1:3
```

```
tc filter add dev eth1 parent 1:0 handle 1 fw flowid 1:4
```

โดยตัว iptables นั้นจะใช้ตาราง mangle ซึ่งจะเป็นตารางที่ใช้สำหรับแก้ไขข้อมูล TOS, TTL, MARK ของแพ็กเก็ต ซึ่งในส่วนของการทำงานของเราจะใช้ในส่วนของ MARK ซึ่งจริง ๆ การ MARK นั้นจะไม่ใช้การแก้ไขข้อมูลในแพ็กเก็ต เพียงแต่จะเป็นการสร้าง ตารางที่ทำการ Map กับกฎที่เรากำหนดใน iptable กับ id ที่มันสร้างขึ้น ซึ่งตรงนี้จะใช้เนื้อที่ใน kernel ในการสร้างตาราง map นั้นเอง โดยค่า id นั้นจะมีค่าได้มากที่สุด 2^{32} หรือก็คือสามารถตั้งกฎได้ถึง 2^{32} กฎนั่นเอง

เมื่อ iptable ทำการ mark แพ็กเก็ตในตารางแล้วตัว tc filter ก็จะตรวจตารางที่ได้ทำการ map ไว้เมื่อตรงกับ ค่า handle ใน tc filter ก็จะทำการส่งแพ็กเก็ตนั้นไปยัง class ที่กำหนดไว้นั่นเอง

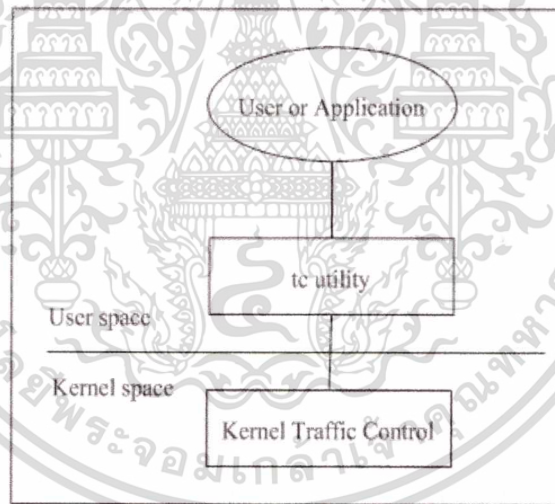


บทที่ 3

การวิเคราะห์และออกแบบระบบ

3.1 การทำงานของระบบเดิม

ในการทำงานของระบบควบคุมปริมาณทราฟฟิกในเครือข่าย ของระบบปฏิบัติการลินุกซ์ นั้น จะเป็นชุดของกลไกการทำงานที่ถูกใช้ในการกำหนดเงื่อนไขของทราฟฟิกในเครือข่าย ซึ่งสามารถที่กำหนดลำดับความสำคัญ (priority) จำกัดอัตราแบนด์วิดท์ของทราฟฟิกที่จะทำการส่งได้ ซึ่งชุดการทำงานนี้จะเรียกว่า “iproute2” ซึ่งการคอนฟิกค่าของตัวชุดการทำงานจะผ่านทาง utility ที่เรียกว่า “tc” ซึ่งใช้การคอนฟิกแบบ Command Line โดยใช้ภาษาที่เรียกว่า “tc language”



รูปที่ 3.1 รูปแบบการทำงานในระบบเดิม

```
[root@Gateway root]# tc qdisc add dev eth1 root handle 1:0 htb
[root@Gateway root]# tc class add dev eth1 parent 1:0 classid 1:1 htb rate 64kbp
s
[root@Gateway root]# tc class show dev eth1
class htb 1:1 root prio 0 rate 512Kbit ceil 512Kbit burst-2254b cburst 2254b
[root@Gateway root]# _
```

รูปที่ 3.2 รูปแบบการคอนฟิกค่าในระบบเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ปัญหาที่พบในระบบปัจจุบัน

- ความยากและซับซ้อนในการใช้งาน
- ภาษาที่ใช้ในการคอนฟิกมีความซับซ้อนและใช้งานยาก
- การแสดงค่าหลังจากที่คอนฟิกค่าไปแล้วใช้ภาษาในการแสดงผลที่ดูเข้าใจยาก
- ผู้ใช้ที่มีความเข้าใจเกี่ยวกับเครือข่ายมากพอสมควร
- ทำการปรับเปลี่ยนค่าคอนฟิกต่าง ๆ ได้ยาก

3.3 แนวความคิดในการออกแบบระบบ

เนื่องจากระบบเก่าจะใช้วิธีการคอนฟิกค่าแบบ Command line และการที่จะต้องคอนฟิกค่านั้นสามารถที่จะคอนฟิกที่หน้าเครื่องนั้นเลยหรือสามารถ Telnet เข้าไปเพื่อทำการคอนฟิกก็ได้ซึ่งเราจะต้องการให้ระบบใหม่สามารถคอนฟิกได้ทั้งที่อยู่หน้าเครื่องหรืออยู่บนเครื่องใด ๆ ก็ได้ ซึ่งการทำงานแบบเว็บเบสนั้นก็สามารถทำงานได้ในลักษณะนั้นเช่นกัน ซึ่งเพียงแค่เครื่องที่จะทำการรันระบบจะต้องโปรแกรมบราวเซอร์อยู่ด้วยก็สามารถที่จะใช้งานระบบได้แล้ว ซึ่งภาษา PHP นั้นมีความสามารถที่จะรัน System Command ของระบบปฏิบัติการลินุกซ์ได้ จึงได้เลือกนำการพัฒนาในระบบในรูปแบบของเว็บแอปพลิเคชันนั่นเอง

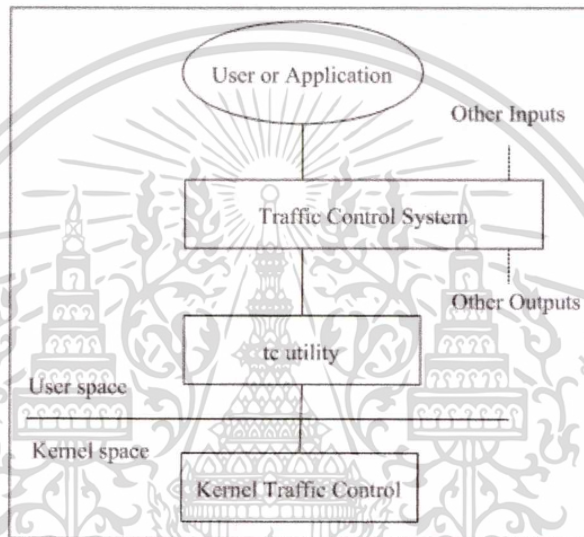
โดยในการพัฒนาระบบในรูปแบบของ Graphic User Interface (GUI) โดยระบบจะทำงานในรูปแบบของ Web-base Application นั้นจะให้ระบบมีความสามารถในการทำงานได้ดังต่อไปนี้

- ตัว Web-base จะใช้ภาษา PHP ในการพัฒนาซึ่งความสามารถของตัวโปรแกรมจะสนับสนุนการสร้าง system Command ในระบบปฏิบัติการลินุกซ์
- การทำงานของเว็บเบสจะทำให้ผู้ใช้สามารถคอนฟิกค่าต่าง ๆ ได้ง่ายขึ้น จากเดิมที่ใช้วิธีคอนฟิกแบบ Command Line และยังมีความสะดวกเนื่องจากผู้ใช้สามารถคอนฟิกผ่านทางเครื่องใดก็ได้
- ซึ่งในตัวระบบที่พัฒนานั้นจะใช้หลักการทำงาน Traffic Control แบบ Classful qdisc (HTB) เพราะมีความยืดหยุ่นในการใช้งานมากกว่าประเภทอื่นทั้งหมด ซึ่งเราจะใช้ความสามารถในการทำงานพื้นฐานของมันมาใช้ในบางส่วนเท่านั้น โดยจะกำหนดให้มีการสร้างคลาสแค่ 2 ระดับเท่านั้น จาก 8 ระดับ
- ในการกรองแพ็กเก็ตนั้นจะใช้ iptable ในการกรองแพ็กเก็ตทั้งหมด ซึ่งในส่วนนี้เราจะทำการติดตั้งความสามารถเพิ่มของ iptable ให้กรองแพ็กเก็ตในระดับ Layer 7 ซึ่งจะทำ User Interface เพื่อสามารถรวมกฎต่าง ๆ ที่สร้างให้เป็นกลุ่มได้เพื่อสะดวกในการกำหนดกฎ แต่ไม่สามารถที่จะจัดการเรื่องความซับซ้อนในการทับซ้อนของกฎใน iptable ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ในส่วนฐานข้อมูลจะใช้ Mysql Server ซึ่งในส่วนตัวโปรแกรมนั้นเหมาะกับระบบที่มีการใช้ข้อมูลในระดับที่ไม่ซับซ้อนและตัวระบบเองก็มีขนาดไม่ใหญ่มาก ซึ่งการเก็บข้อมูลบางส่วนไว้ในฐานข้อมูลนั้นก็เพื่อช่วยในการเก็บค่าคอนฟิก และการแสดงข้อมูลบางของระบบที่ได้ทำการคอนฟิกไปแล้วนั่นเอง

ซึ่งภาพรวมของระบบใหม่นั้นจะอยู่ในรูปของการเขียนคลุมทั้ระบบเก่า หรือ การเขียน GUI ให้กับระบบเดิมเพื่อให้ใช้งานได้ง่ายขึ้นนั่นเอง



รูปที่ 3.3 ภาพของรูปแบบในการพัฒนาระบบใหม่

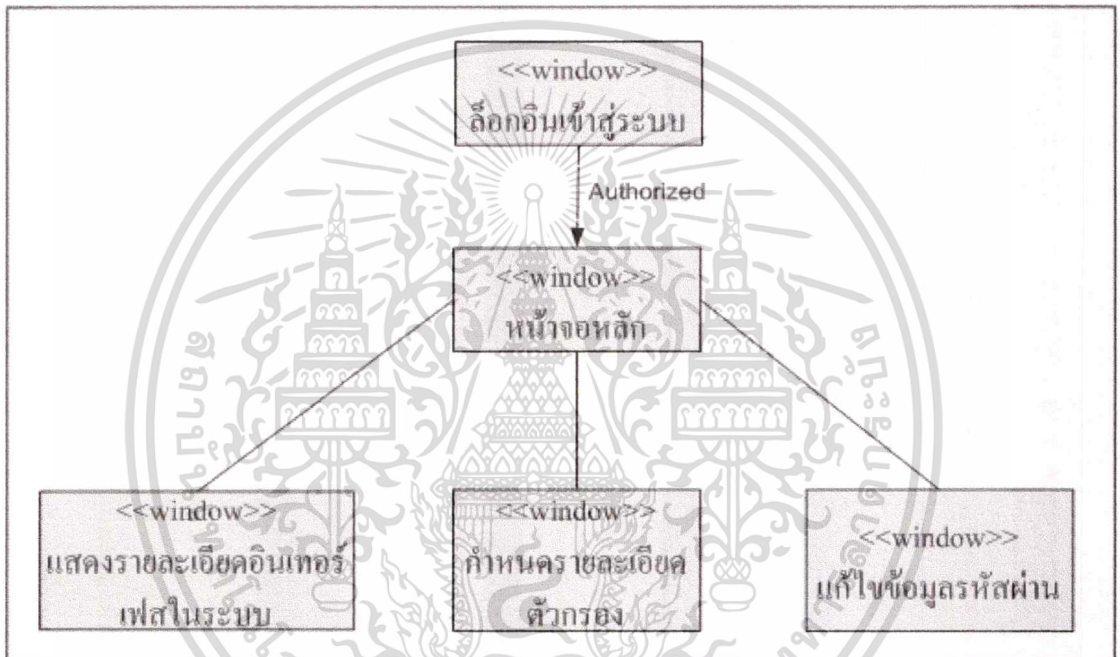
3.4 ขั้นตอนในการทำงานของระบบใหม่

ในการพัฒนาระบบเราจะทำการออกแบบโดยการสร้าง Interface Structure Design เพื่อที่จะสามารถมองเห็นถึงภาพรวมของระบบทั้งหมดได้โดยง่าย ซึ่งในการส่วนการทำงานส่วนใหญ่ของระบบจะมีผู้ใช้งานเพียงคนเดียวเท่านั้น ซึ่งจะสามารถอธิบายภาพรวมของระบบได้โดยการใช้ Window Navigation Diagram (WND) ดังนี้

ส่วนของการทำงานหลัก

เมื่อทำการล็อกอินเข้าสู่ระบบจะเข้าสู่เมนูหลักซึ่งสามารถจะเลือกการทำงานได้ดังนี้

1. ทำการเพิ่มข้อกำหนดเกี่ยวกับการกำหนดตัวกรอง
2. การกำหนดการควบคุมปริมาณกราฟิกในแต่ละอินเทอร์เฟซ
3. การเปลี่ยนแปลงรหัสผ่าน

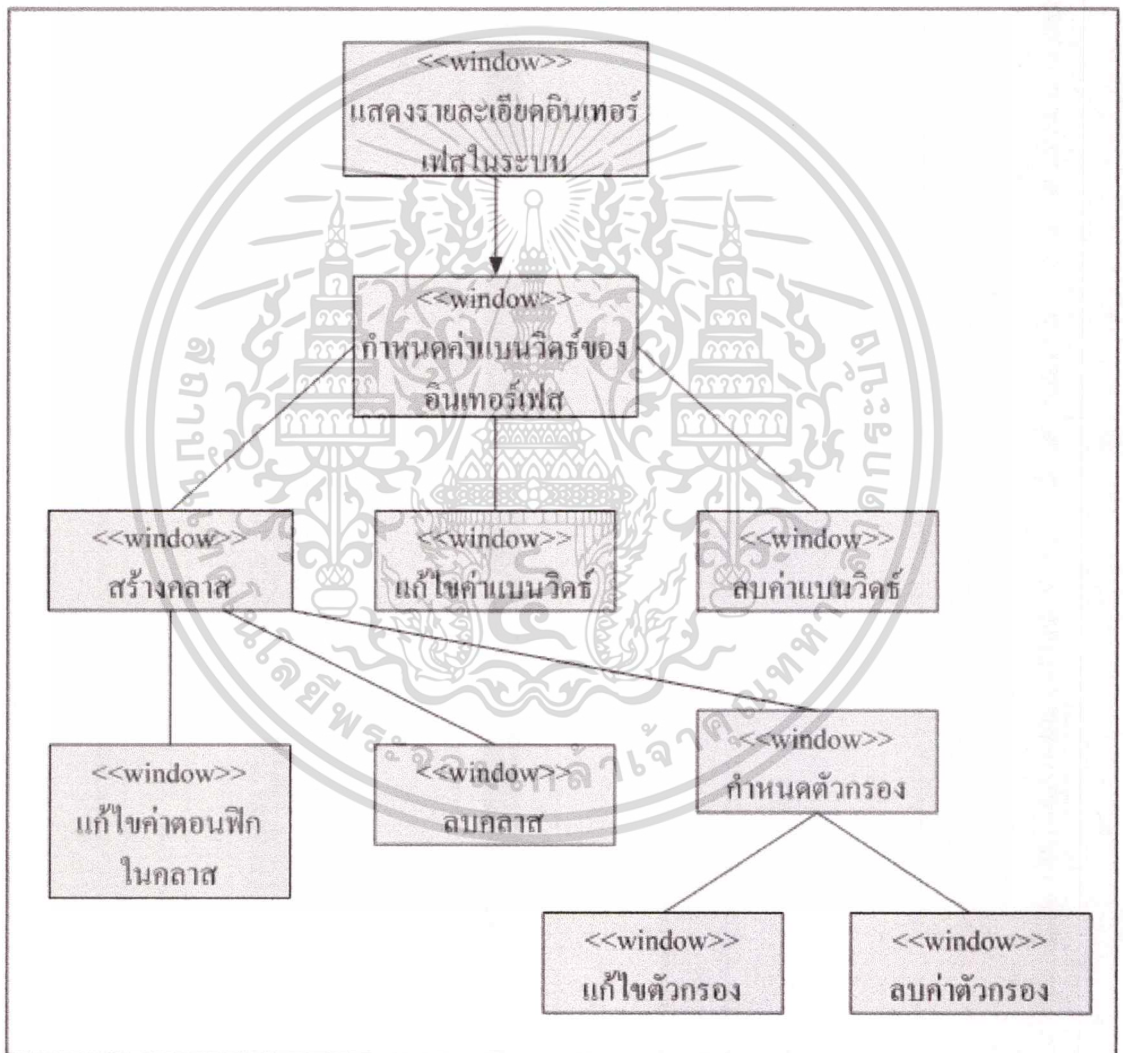


รูปที่ 3.4 Window Navigation Diagram ของส่วนเมนูหลัก

ส่วนของการทำงานเกี่ยวกับการคอนฟิกค่าในแต่ละอินเทอร์เน็ต

ส่วนของการคอนฟิกค่าในแต่ละอินเทอร์เน็ตนั้นสามารถเข้ามาในส่วนนี้ได้จากหน้าจอแสดงรายละเอียดอินเทอร์เน็ตในระบบซึ่งสามารถอธิบายได้ดังนี้

1. ทำการกำหนดค่าแบนวิดธ์ในอินเทอร์เน็ต แก้ไขค่าแบนวิดธ์ และทำการลบค่าที่คอนฟิกได้
2. สร้างคลาสในอินเทอร์เน็ต แก้ไขค่าคอนฟิกต่างๆ ของคลาสที่ทำการสร้างขึ้น
3. กำหนดตัวกรองในแต่ละคลาส ทำการแก้ไข และทำการลบได้



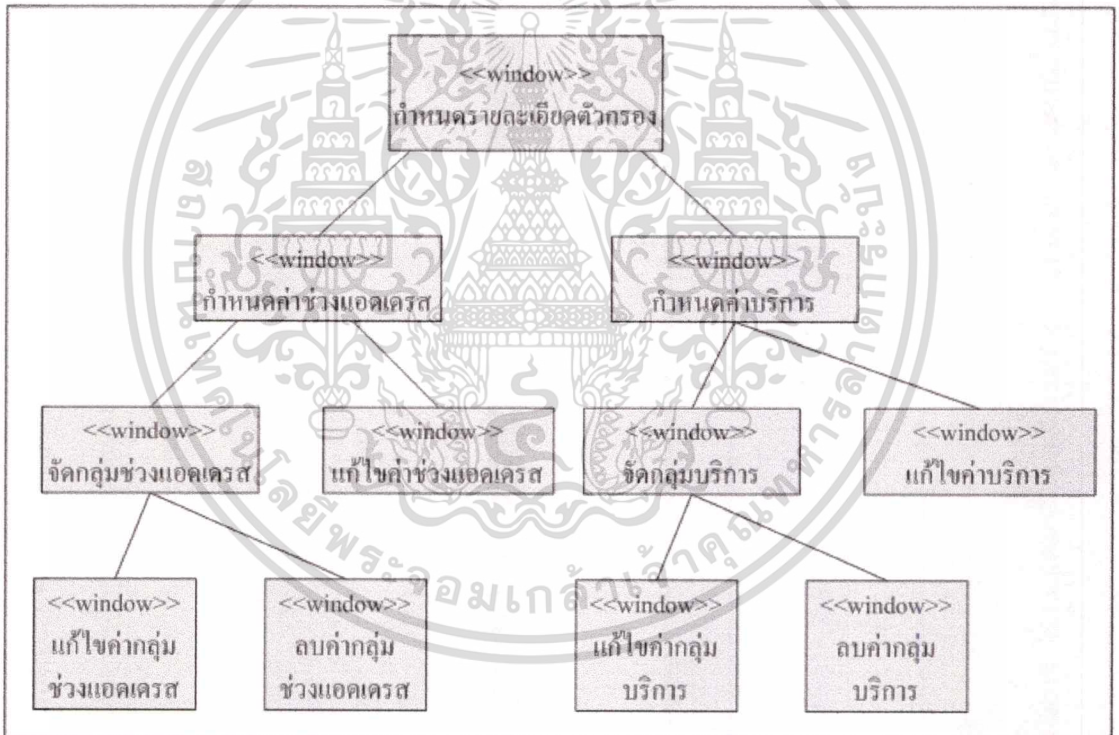
รูปที่ 3.5 Window Navigation Diagram ของส่วนเมนูการคอนฟิกค่าในอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของการทำงานเกี่ยวกับการกำหนดรายละเอียดตัวกรอง

ส่วนของการกำหนดค่าตัวกรองนั้นสามารถที่จะกำหนดค่าเป็นแบบกำหนดช่วงหรือบริการได้ ซึ่งสามารถที่จะทำการกำหนดเป็นแบบเดี่ยว ๆ หลังจากนั้นก็สามารถที่จะทำการจัดกลุ่มจากข้อกำหนดที่สร้างขึ้นได้ ซึ่งในส่วนของค่าที่ทำการกำหนดนี้สามารถที่จะนำไปใช้ในการกำหนดค่าตัวกรองในแต่ละคลาสอีกทีหนึ่ง ซึ่งสามารถอธิบายได้ดังนี้

1. กำหนดค่าตัวกรองช่วงแอดเดรส แก๊วค่า และทำการลบค่าที่คอนฟิกลงได้
2. ทำการจัดกลุ่มค่าตัวกรองช่วงแอดเดรส แก๊วค่า และทำการลบค่าที่คอนฟิกลงได้
3. กำหนดค่าตัวกรองบริการ แก๊วค่า และทำการลบค่าที่คอนฟิกลงได้
4. ทำการจัดกลุ่มค่าตัวกรองช่วงแอดเดรส แก๊วค่า และทำการลบค่าที่คอนฟิกลงได้



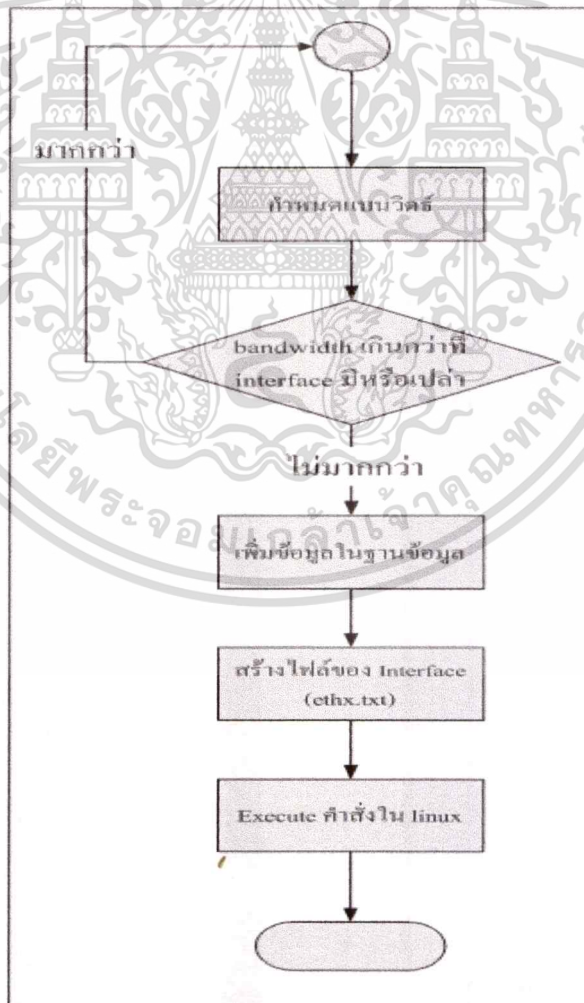
รูปที่ 3.6 Window Navigation Diagram ของส่วนเมนูการกำหนดรายละเอียดตัวกรอง

ในส่วนที่ผ่านมาเราได้ทำการอธิบายภาพรวมของระบบทั้งหมดจาก Window Navigation Diagram ซึ่งจะช่วยให้เรามองการทำงานของระบบได้ง่ายขึ้น และในส่วนนี้เราจะทำการแสดงการทำงานในส่วนหลัก ๆ ที่สำคัญของระบบซึ่งในส่วนต่าง ๆ นี้เราจะใช้ Flow Chart ในการอธิบายรายละเอียดของฟังก์ชันในการทำงานในแต่ละส่วน

ฟังก์ชันการทำงานการกำหนดแบนวิดธ์ในแต่ละอินเทอร์เฟซ

ระบบที่จะทำการพัฒนาจะต้องทำการแสดงรายละเอียดของอินเทอร์เฟซของระบบทั้งหมดออกมาเพื่อให้ผู้ใช้ได้สามารถที่จะทำการเลือกอินเทอร์เฟซนั้นและทำการกำหนดแบนวิดธ์ในแต่ละอินเทอร์เฟซเพื่อที่จะทำการกำหนดการควบคุมปริมาณทราฟฟิกได้ ซึ่งจะอธิบายได้ดังนี้

- กำหนดค่าแบนวิดธ์ในอินเทอร์เฟซ
- ทำการตรวจสอบว่าค่าแบนวิดธ์ที่กำหนดมีค่ามากกว่าที่อินเทอร์เฟซนั้นรับได้หรือเปล่า
- ทำการจัดเก็บข้อมูลลงในฐานข้อมูล
- ทำการสร้างไฟล์เพื่อเก็บไว้ในลักษณะของ Script
- ทำการ Execute คำสั่งในระบบปฏิบัติการเพื่อให้ระบบทำงาน



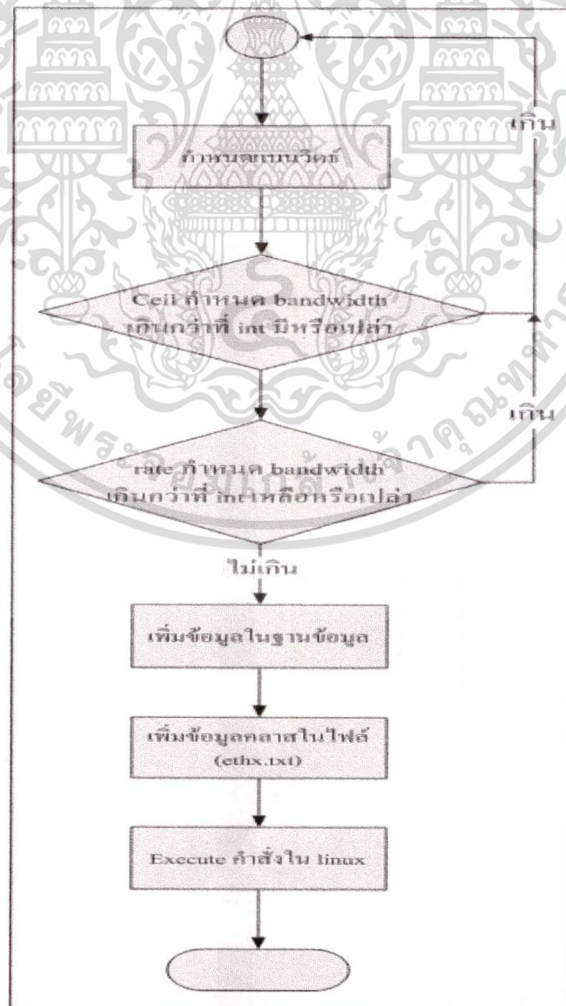
รูปที่ 3.7 Flow Chart แสดงถึงการกำหนดแบนวิดธ์ในแต่ละอินเทอร์เฟซ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชันการทำงานของการสร้างคลาส

หลังจากที่ทำการกำหนดแบนวิดธ์ในอินเทอร์เน็ตเฟสเรียบร้อยแล้ว ระบบจะต้องทำการสร้างคลาส ซึ่งการสร้างคลาสนั้นจะต้องกำหนดแบนวิดธ์ของ rate และ ceil ให้มีค่าได้ไม่มากกว่าแบนวิดธ์ที่กำหนดไว้ในอินเทอร์เน็ตเฟสนั้น ซึ่งจะอธิบายได้ดังนี้

- กำหนดค่าแบนวิดธ์ของ rate และ ceil ในคลาส
- ทำการตรวจสอบว่าค่าแบนวิดธ์ที่กำหนดมีค่ามากกว่าที่อินเทอร์เน็ตเฟสนั้นรับได้หรือเปล่าทั้งในส่วน of rate และในส่วน of ceil
- ทำการจัดเก็บข้อมูลลงในฐานข้อมูล
- ทำการสร้างไฟล์เพื่อเก็บไว้ในลักษณะของ Script
- ทำการ Execute คำสั่งในระบบปฏิบัติการเพื่อให้ระบบทำงาน



รูปที่ 3.8 Flow Chart แสดงถึงการสร้างคลาสในแต่ละอินเทอร์เน็ตเฟส

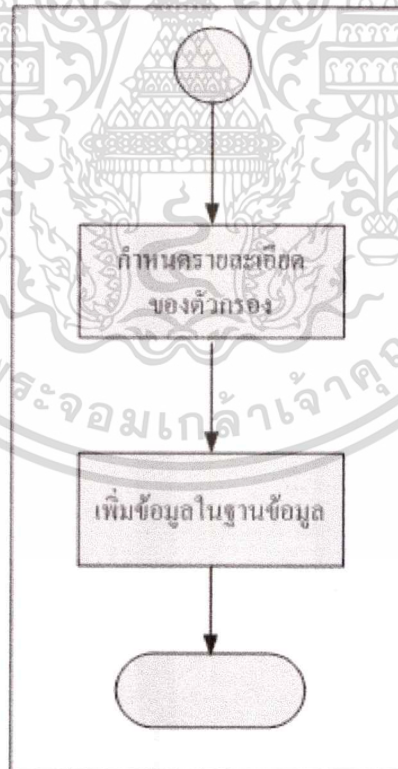
เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือการสงวนสิทธิ์ในเนื้อหา ไม่อนุญาตให้เผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชันการทำงานของกาหนดค่ารายละเอียดตัวกรอง

หลังจากที่ทำการสร้างคลาสเรียบร้อยแล้ว เราจะต้องมีการกำหนดตัวกรองในแต่ละคลาส เพื่อการกรองทราฟฟิกแต่ละประเภทให้อยู่ในคลาสที่เราต้องการ แต่ก่อนที่เราจะทำการกำหนดค่าตัวกรองได้นั้นเราจะต้องทำการกำหนดค่ารายละเอียดตัวกรองก่อน ซึ่งจะอธิบายได้ดังนี้

- กำหนดค่ารายละเอียดตัวกรอง ซึ่งค่านี้อาจจะมีค่าที่เป็น Host คือ IP Address, Network Address และกลุ่มของ IP และ Network Address ที่ทำการสร้างขึ้น และ Service คือ Protocol, Port Number และกลุ่มของ Service ที่ทำการสร้างขึ้นมา
- ทำการจัดเก็บข้อมูลลงในฐานข้อมูล

ซึ่งในส่วนการทำงานของฟังก์ชันนี้ จะแตกต่างจากการทำงานของฟังก์ชันอื่น คือ จะไม่มีการสร้างไฟล์ที่เป็น Script และสั่ง Execute System Command ในระบบปฏิบัติการเพราะจะเป็นเพียงการกำหนดรายละเอียดของตัวกรองเท่านั้น เพื่อที่จะทำให้การสร้างตัวกรองในหลายเงื่อนไขได้ง่ายขึ้นเท่านั้น ซึ่งในส่วนนี้ก็จะเป็นส่วนประกอบของตัวกรองอีกที่เท่านั้น

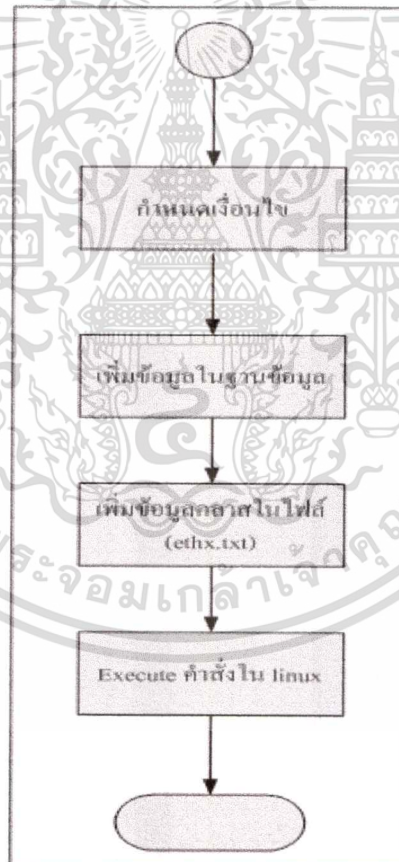


รูปที่ 3.9 Flow Chart แสดงถึงการกำหนดค่ารายละเอียดของตัวกรอง

ฟังก์ชันการทำงานของกาหนดตัวกรอง

หลังจากที่ทำการสร้างคลาสเรียบร้อยแล้ว เราจะต้องมีการกำหนดตัวกรองในแต่ละคลาส เพื่อการกรองกราฟฟิคแต่ละประเภทให้อยู่ในคลาสที่เราต้องการ และเมื่อเราทำการกำหนดค่ารายละเอียดตัวกรองแล้ว เราก็สามารถที่จะทำการกำหนดตัวกรองได้ ซึ่งอธิบายได้ดังนี้

- กำหนดค่าเงื่อนไขของตัวกรอง ซึ่งค่านี้จะเป็นค่าที่นำมาจากการกำหนดค่ารายละเอียดตัวกรอง ซึ่งในส่วนของระบบเราจะไม่มีการจัดการเกี่ยวกับความซับซ้อนของเงื่อนไขต่าง ๆ
- ทำการจัดเก็บข้อมูลลงในฐานข้อมูล
- ทำการสร้างไฟล์เพื่อเก็บไว้ในลักษณะของ Script
- ทำการ Execute คำสั่งในระบบปฏิบัติการเพื่อให้ระบบทำงาน



รูปที่ 3.10 Flow Chart แสดงถึงกาหนดเงื่อนไขในแต่ละคลาส

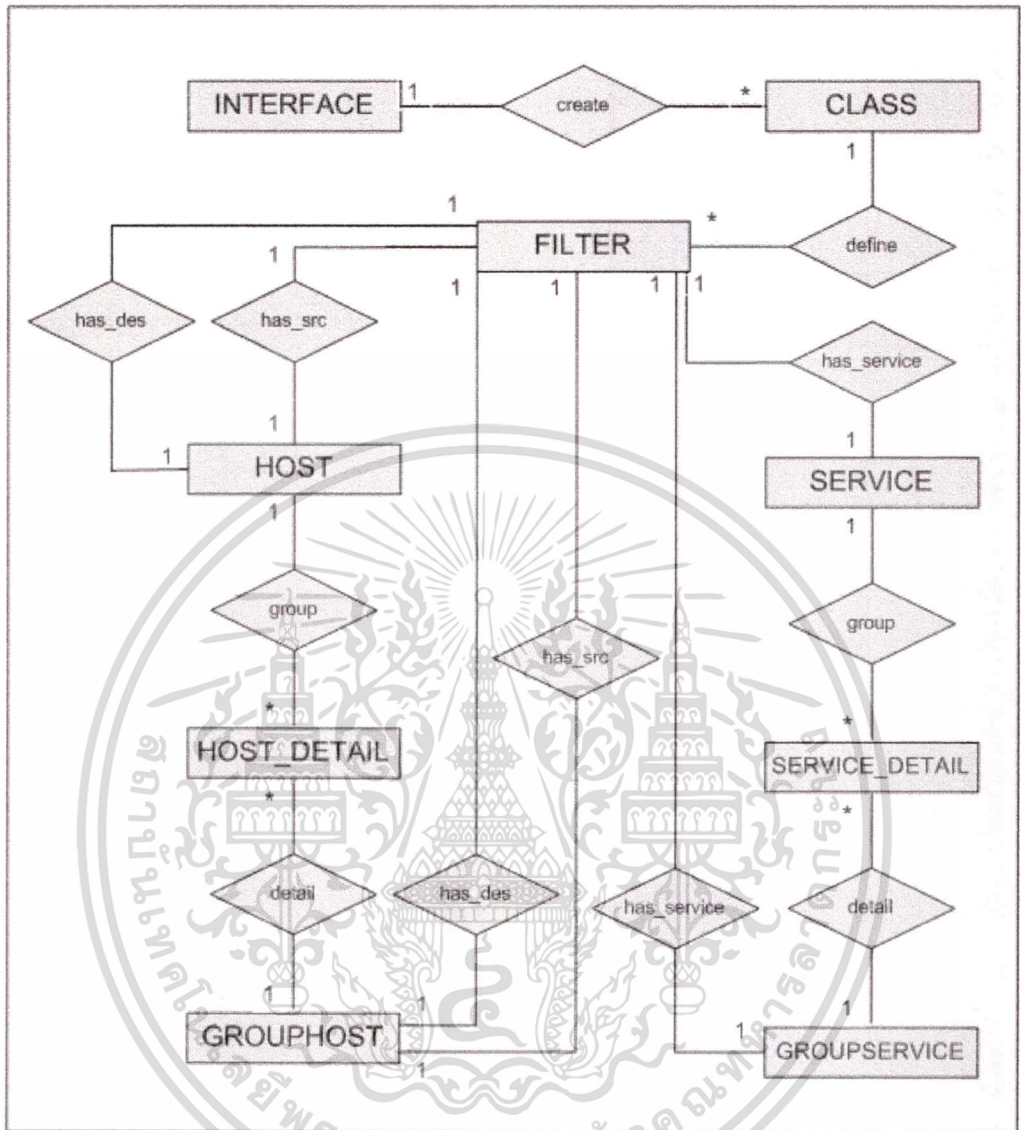
3.5 การออกแบบฐานข้อมูล

การออกแบบฐานข้อมูลในการพัฒนาระบบนี้จะเป็นฐานข้อมูลเชิงสัมพันธ์ ออกแบบนั้นได้ทำการอ้างอิงมาจาก Flow Chart ข้างต้น ประกอบด้วยทั้งหมด 9 ตาราง

ซึ่งการ

- ตาราง Admin เก็บข้อมูลของผู้ใช้ระบบ
- ตาราง Interface เก็บข้อมูลหลักของอินเทอร์เน็ต
- ตาราง Class เก็บข้อมูลหลักของคลาสที่ทำการสร้างขึ้น
- ตาราง Filter เก็บข้อมูลของเงื่อนไขหรือตัวกรองของแต่ละคลาส
- ตาราง Host เก็บข้อมูลการกำหนด Ip และ Network Address ที่ทำการสร้างขึ้น
- ตาราง Grouphost เก็บข้อมูลการจัดกลุ่มของตาราง Host
- ตาราง Host_detail เก็บรายละเอียดของกลุ่มของ Host
- ตาราง Service เก็บข้อมูลการกำหนดบริการที่ทำการสร้างขึ้น
- ตาราง Groupservice เก็บข้อมูลการจัดกลุ่มของบริการ
- ตาราง Service_detail เก็บรายละเอียดของกลุ่มบริการ





รูปที่ 3.11 แสดง E-R Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 Data Dictionary

ตารางที่ 3.1 รายละเอียดตาราง Admin

Name	Description	PK/FK	Characteristic	Domain	Null	Table Reference
<u>username</u>	ชื่อผู้ใช้งาน	PK	Type = varchar Length = 20		NO	
pwd	รหัสผ่าน		Type = varchar Length = 20		NO	

ตารางที่ 3.2 รายละเอียดตาราง Interface

Name	Description	PK/FK	Characteristic	Domain	Null	Table Reference
<u>inname</u>	ชื่ออินเทอร์เน็ตเฟส	PK	Type = varchar Length = 30		NO	
rate	อัตราแบนวิดท์		Type = int Length = 5		NO	
unit	หน่วยของแบนวิดท์		Type = varchar Length = 5		NO	

ตารางที่ 3.3 รายละเอียดตาราง Class

Name	Description	PK/FK	Characteristic	Domain	Null	Table Reference
<u>classid</u>	เลข id หลักของคลาส	PK	Type = int Length = 5		NO	
name	ชื่อคลาส		Type = int Length = 5		NO	
parent	เลข id หลักของคลาส พ่อแม่		Type = int Length = 5		NO	

rate	อัตราแบนวิดท์		Type = int Length = 5		NO	
unitrate	หน่วยของแบนวิดท์		Type = varchar Length = 5		NO	
ceil	อัตราแบนวิดท์ของ ceil		Type = int Length = 5			
unitceil	หน่วยของแบนวิดท์ ของ ceil		Type = varchar Length = 5			
prio	ค่าลำดับความสำคัญ		Type = int Length = 1		NO	
intname	ชื่ออินเทอร์เฟซ	FK	Type = varchar Length = 20			Interface

ตารางที่ 3.4 รายละเอียดตาราง Filter

Name	Description	PK/FK	Characteristic	Domain	Null	Table Reference
<u>idfilter</u>	รหัสตัวกรอง	PK	Type = int Length = 5		NO	
gorh_src	ค่าเช็คว่าเป็น Source IP เป็น Host หรือเป็น Group Host		Type = int Length = 1		NO	
ip_src	ช่วง Source IP	FK	Type = varchar Length = 30		NO	host, grouphost
gorh_dest	ค่าเช็คว่าเป็น Dest IP เป็น Host หรือเป็น Group Host		Type = int Length = 1		NO	
ip_dest	ช่วง Destination IP	FK	Type = varchar Length = 30		NO	host, grouphost

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

gors	ค่าเซ็คว่าเป็น Service หรือเป็น Group Service		Type=int Length = 1		NO	
service	ชื่อบริการ	FK	Type = varchar Length = 30		NO	service, groupservice

ตารางที่ 3.5 รายละเอียดตาราง Host

Name	Description	PK/FK	Characteristic	Domain	Null	Table Reference
host	ชื่อ โฮสต์	PK	Type = varchar Length = 30		NO	
ipaddr	ค่า IP Address		Type = varchar Length = 17		NO	
subnet	ค่า Subnet Mask		Type = int Length = 2		NO	
comment	รายละเอียดเพิ่มเติม		Type = varchar Length = 100		YES	

ตารางที่ 3.6 รายละเอียดตาราง Host_detail

Name	Description	PK/FK	Characteristic	Domain	Null	Table Reference
grouph	ชื่อกลุ่ม โฮสต์	PK,FK	Type = varchar Length = 30		NO	grouphost
host	ชื่อ โฮสต์	PK,FK	Type = varchar Length = 30		NO	host

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.7 รายละเอียดตาราง GroupHost

Name	Description	PK/FK	Characteristic	Domain	Null	Table Reference
<u>grouph</u>	ชื่อกลุ่มโฮสต์	PK	Type = varchar Length = 30		NO	
comment	รายละเอียดเพิ่มเติม		Type = varchar Length = 100		YES	

ตารางที่ 3.8 รายละเอียดตาราง Service

Name	Description	PK/FK	Characteristic	Domain	Null	Table Reference
<u>Service</u>	ชื่อบริการ	PK	Type = varchar Length = 30		NO	
proto	ชื่อโปรโตคอล		Type = varchar Length = 7		NO	
srcport	ค่า Source Port		Type = varchar Length = 11		NO	
destport	ค่า Destination Port		Type = varchar Length = 11		NO	
Comment	รายละเอียดเพิ่มเติม		Type = varchar Length = 100		YES	

ตารางที่ 3.9 รายละเอียดตาราง Service_detail

Name	Description	PK/FK	Characteristic	Domain	Null	Table Reference
<u>Groups</u>	ชื่อกลุ่มบริการ	PK,FK	Type = varchar Length = 30		NO	groupservice
<u>Service</u>	ชื่อบริการ	PK,FK	Type = varchar Length = 30		NO	service

ตารางที่ 3.10 รายละเอียดตาราง Groupservice

Name	Description	PK/FK	Characteristic	Domain	Null	Table Reference
<u>Group</u>	ชื่อกลุ่มบริการ	PK	Type = varchar Length = 30		NO	
comment	รายละเอียดเพิ่มเติม		Type = varchar Length = 100		YES	



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

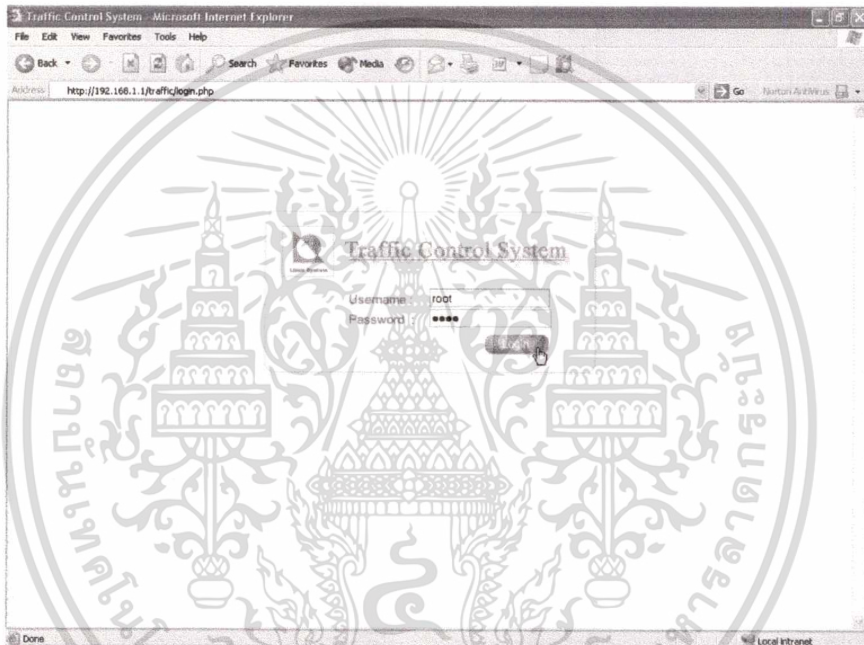
การพัฒนาและทดสอบระบบ

4.1 เครื่องมือและซอฟต์แวร์ที่ใช้ในการพัฒนาระบบ

1. ระบบปฏิบัติการ Linux Redhat 9.0 เนื่องจากตัวโปรแกรมควบคุมปริมาณกราฟฟิก เป็นโปรแกรมที่ทำงานบนระบบปฏิบัติการลินุกซ์ ซึ่งเป็นตัวโปรแกรมที่ติดกับเคอร์เนลตั้งแต่เวอร์ชัน 2.4.20 เป็นต้นไปซึ่งไม่ต้องทำการคอมไพล์เคอร์เนลใหม่ก็สามารถที่จะใช้งานความสามารถของตัวโปรแกรมได้ทันที
2. Iproute2 โปรแกรมควบคุมปริมาณกราฟฟิกในระบบปฏิบัติการลินุกซ์
3. Apache Web Server เป็นโปรแกรมเว็บเซิร์ฟเวอร์ที่จะทำการเก็บตัวโปรแกรม (PHP, HTML) ซึ่งทำหน้าที่คอยฟังการเรียกใช้โปรแกรมจากเครื่องอื่น ๆ ที่ใช้โปรแกรมเว็บเบราว์เซอร์อีกทีหนึ่งซึ่งตัวเว็บเซิร์ฟเวอร์นั้นก็ส่งตัวโปรแกรมนั้นให้อีกที
4. PHP เป็นภาษาที่ใช้ในการเขียน Script แทรกลงไป ในภาษา HTML เพื่อให้มีความสามารถในการติดต่อกับฐานข้อมูล และอื่น ๆ อีกมากมาย ซึ่งอีกความสามารถหนึ่งที่เรานำมาใช้คือการสั่ง Execute System Command ในระบบปฏิบัติการลินุกซ์นั่นเอง
5. MySQL Server เป็นโปรแกรมฐานข้อมูลที่สามารถทำงานได้ในระบบปฏิบัติการลินุกซ์ซึ่งมีความสามารถในการทำงานเหมาะสำหรับการสร้างฐานข้อมูลที่มีขนาดกลางและความซับซ้อนไม่มากนัก
6. Macromedia Dreamweaver MX เป็นโปรแกรมที่ช่วยในการเขียนภาษา HTML และ PHP ให้ง่ายขึ้น
7. Editplus เป็นโปรแกรมที่ช่วยในการเขียนภาษา HTML และ PHP ให้ง่ายขึ้น
8. Iptable เป็นตัวกรองกราฟฟิกหรือแบ่งประเภทของกราฟฟิกนั่นเองซึ่งในตัว tc นั้นจะมีตัวกรองของมันเองอยู่แล้ว แต่การใช้ iptable จะมีความสามารถในการกำหนดเงื่อนไขที่ยืดหยุ่นกว่าและสามารถกรองในระดับแอปพลิเคชันเลขอร์ได้อีกด้วย

4.2 ผลการพัฒนาระบบ

ในการพัฒนาระบบควบคุมปริมาณจราจรที่นั่นเราได้ทำการออกแบบให้อยู่ในรูปแบบของ Web Base Application ซึ่งการเรียกใช้งานระบบนั้นจะต้องมีโปรแกรมบราวเซอร์ในการแสดงผลและสั่งการทำงาน ซึ่งผู้ใช้งานสามารถเรียกใช้ระบบจากที่ใดก็ได้ เพียงแต่เครื่องนั้นจะต้องมีเครือข่ายที่เชื่อมต่อถึงเครื่องที่ทำการติดตั้งระบบไว้ และทำการพิมพ์ URL ให้ตรงกับ IP Address ของเครื่องระบบก็จะสามารถใช้งานระบบได้ทันที

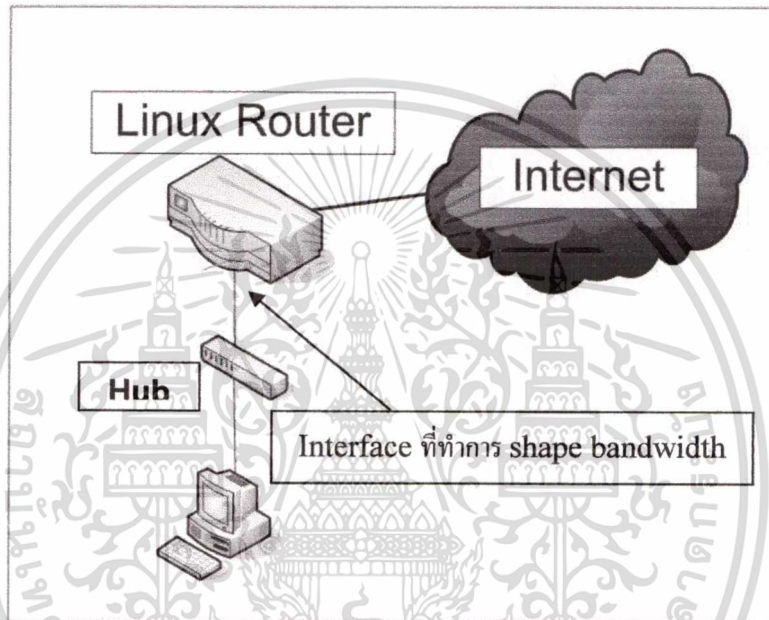


รูปที่ 4.1 หน้าจอการเรียกใช้งานระบบจากโปรแกรมบราวเซอร์

ซึ่งจากการลองใช้งานระบบนั้นสามารถที่จะทำการควบคุมการทำงานต่าง ๆ ที่เราต้องการได้เป็นอย่างดี สะดวกในการใช้งานมากขึ้น จากระบบเดิมที่ต้องทำการคอนฟิกค่าแบบ Command Line มาอยู่ในรูปแบบของ Graphic User Interface (GUI) ทำให้ผู้ใช้สามารถใช้งานระบบเข้าใจการใช้งานระบบได้ง่ายขึ้น ไม่ว่าจะป็นกำหนดค่าต่าง ๆ ของระบบ รวมถึงการแสดงผลการกำหนดค่าต่าง ๆ ของระบบที่สามารถดูเข้าใจได้ง่ายขึ้น

4.3 การทดสอบการทำงานของระบบ

จากการออกแบบและพัฒนาระบบควบคุมปริมาณกราฟฟิกในเครือข่ายนั้น ได้ผลออกมาตามที่คาดไว้ ซึ่งระบบสามารถทำการควบคุมปริมาณกราฟฟิกได้อย่างดี โดยบางเวลาก็อาจมีค่าที่คลาดเคลื่อนจากการที่ระบบกำหนดไว้บ้าง แต่ก็น้อยมาก โดยในการทดสอบระบบนั้นเราได้ทำการจำลองสภาพแวดล้อมของเครือข่าย โดยที่จะสามารถอธิบายได้ในรูปที่ 5.1



รูปที่ 4.2 สภาพแวดล้อมเครือข่ายที่ใช้ในการทดสอบระบบ

โดยในการทดลองควบคุมปริมาณกราฟฟิกนั้น เราจะทำการทดลองโดยการดาวน์โหลดข้อมูลจากอินเทอร์เน็ตจาก โดยจะดาวน์โหลดจาก Source IP ที่แตกต่างกัน 3 แห่ง เพื่อที่จะให้เห็นถึงความสามารถของระบบที่สามารถจัดการควบคุมอัตราการใช้แบนวิดท์ โดยระบบสามารถกำหนดได้จาก Source IP, Destination IP และ Port และ บริการในระดับแอปพลิเคชันเลเยอร์ ซึ่งในการทดสอบนี้จะทำให้เราเห็นถึงการควบคุมกราฟฟิกจากข้อกำหนดต่างที่ได้กล่าวไปข้างต้น

เริ่มต้นเราจะกำหนดกฎเกณฑ์ในการควบคุมโดยการกำหนดจาก Source IP ของ Web Server 3 แห่งด้วยกัน ที่แรกจะเป็นที่ Web Server ของลาดกระบัง ที่สองจะเป็นที่ Web Server ของ Thaiware และสุดท้ายจะเป็น Web Server ของ Nectec โดยเราจะทำการดาวน์โหลดข้อมูลจาก Web Server ทั้ง 3 แห่งนั้น พร้อม ๆ กัน โดยการกำหนดค่าของระบบจะเป็นดังนี้

ทดสอบการควบคุมแบนวิดธ์

Src: 203.185.132.73

Interface: < eth1 > แบนวิดธ์กำหนดไว้: 2048 kbps เหนือค่าสว. แก้ไข ลบ

ชื่อคลาส	แบนวิดธ์ที่กำหนด / มากที่สุด	จำนวนในการรับ		
A	256 kbps / 256 kbps	0	แก้ไข	ลบ
B	512 kbps / 512 kbps	0	แก้ไข	ลบ
C	1024 kbps / 1024 kbps	0	แก้ไข	ลบ
Default	128 kbps / 128 kbps	7	แก้ไข	ลบ

Src: 161.246.38.25

Src: 203.146.140.101

รูปที่ 4.3 การคอนฟิกค่าของระบบในการทดลอง

ในการคอนฟิกค่านั้นเราจะมีคลาสที่เรากำหนดเพื่อใช้ในการทดสอบด้วยกัน 3 คลาส ส่วนอีกหนึ่งคลาสเป็นคลาสหลักที่ต้องมีไว้สำหรับทราฟฟิกที่ไม่ตรงกับเงื่อนไขใด ๆ เลย ซึ่งคลาสที่เรากำหนดนั้น จะมีคลาส A (Nectec), B (ลาดกระบัง) และ C (Thaiware) โดยกำหนดค่า Source IP ของ Web Server นั้น ๆ ให้อยู่ในเงื่อนไขของแต่ละคลาส โดยให้คลาส A ให้สามารถใช้แบนวิดธ์ได้ในอัตรา 256 kbps หรือ 32 KB/s คลาส B 512 kbps หรือ 64 KB/s ส่วนคลาส C นั้นให้สามารถใช้แบนวิดธ์ได้ในอัตรา 1024 kbps หรือ 128 KB/s ซึ่งในการดาวน์โหลดข้อมูลจาก Web Server แต่ละที่นั้นเราจะใช้โปรแกรม Download Accelerator หรือเรียกสั้น ๆ ว่า DAP นั้นเอง ซึ่งโปรแกรมนี้สามารถที่จะแสดงอัตราความเร็วในการดาวน์โหลดเป็น KB/s ได้ ซึ่งจะทำให้เราทราบถึงอัตราความเร็วในการดาวน์โหลดของแต่ละคลาสที่เรากำหนดนั่นเอง

Class A : 256 kbps or 32 KB
(Filter) src : 203.185.132.73

Saving To: C:\Documents and Settings\admin\Desktop\LEXTRON_2.1.exe
Status: Receiving file... (5)
Resume: Resume Supported
URL File: http://lextron.net.ec.or.th/program/LEXTRON_2.1.exe
File Info:
Time Left: 3 Min(s), 48 Second(s)
Transfer Rate: 32.0 KB/Sec File Size: 13.56 MB of 20.70 MB

Class B : 512 kbps or 64 KB
(Filter) src : 161.246.38.25

Saving To: C:\Documents and Settings\Administrator\Windows\FKBB05935-SP2-ENU.exe
Status: Receiving file... (7)
Resume: Resume Supported
URL File: http://www.it.kmit.ac.th/wireless/Windows/FKBB05935-SP2-ENU.exe
File Info:
Time Left: 1 Hour(s), 2 Min(s), 32 Second(s)
Transfer Rate: 65.2 KB/Sec File Size: 27.00 MB of 265.01 MB

Class C : 1024 kbps or 128 KB
(Filter) src : 203.146.140.101

Saving To: C:\Documents and Settings\admin\Desktop\CEO_City_Updated.exe
Status: Receiving file... (5)
Resume: Resume Supported
URL File: http://download.thaiware.com/cecity/CEO_City_Updated.exe
File Info:
Time Left: 29 Min(s), 51 Second(s)
Transfer Rate: 1231 KB/Sec File Size: 43.06 MB of 259.68 MB

#	Size	Connection Status	Server location
1-	1.09 MB	Receiving file...	www.it.kmit.ac.th
2-	1.80 MB	Receiving file...	www.it.kmit.ac.th
3-	2.70 MB	Receiving file...	www.it.kmit.ac.th
4-	1.86 MB	Receiving file...	www.it.kmit.ac.th
5-	1.25 MB	Receiving file...	www.it.kmit.ac.th
6-	2.63 MB	Receiving file...	www.it.kmit.ac.th
7-	2.25 MB	Receiving file...	www.it.kmit.ac.th

#	Size	Connection Status	Server location
1-	7.76 MB	Receiving file...	download.thaiware.com
2-	6.37 MB	Receiving file...	download.thaiware.com
3-	3.64 MB	Receiving file...	download.thaiware.com
4-	6.55 MB	Receiving file...	download.thaiware.com
5-	2.72 MB	Receiving file...	download.thaiware.com
6-	1.14 MB	Receiving file...	download.thaiware.com

รูปที่ 4.4 การทดสอบจับอัตราการความเร็วในการดาวน์โหลดในแต่ละคลาส

จากการทดสอบการทำงานของระบบเป็นตามที่ได้ออกแบบไว้ ซึ่งระบบสามารถที่จะทำการจัดการควบคุมปริมาณกราฟฟิกได้เป็นอย่างดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมีให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทสรุป

5.1 สรุปผลการออกแบบและพัฒนาระบบงาน

จากการพัฒนาระบบควบคุมปริมาณทราฟฟิกบนเครือข่าย ผลที่ได้รับเป็นไปตามขอบเขตที่วางไว้ อันประกอบไปด้วยหน้าที่การทำงานดังต่อไปนี้

- ระบบสามารถที่จะกำหนดการควบคุมทราฟฟิกประเภทต่าง ๆ ได้เป็นอย่างดี เนื่องจากได้นำความสามารถของไอพีเทเบิลมาร่วมใช้งานด้วย จึงสามารถที่จะตั้งเงื่อนไขได้หลากหลายและมีความยืดหยุ่นสูง
- ระบบสามารถที่จะทำการกำหนดค่าคอนฟิกต่าง ๆ รวมถึงฟังก์ชันการทำงานของระบบได้อย่างครบถ้วนตามที่ออกแบบไว้ ซึ่งเพียงพอที่จะนำไปใช้งานในเครือข่ายขนาดกลางได้เป็นอย่างดี
- ระบบนั้นสามารถควบคุมปริมาณทราฟฟิกได้ตามที่กำหนดไว้เป็นอย่างดี ถึงแม้ว่าอาจจะมีค่าคลาดเคลื่อนไปบ้างเล็กน้อยแต่ก็มีไม่มากนัก

5.2 ปัญหาและข้อจำกัดที่พบ

1. สภาพแวดล้อมของเครือข่ายที่ได้ทำการทดสอบนั้นยังมีขนาดเล็กอยู่ ซึ่งไม่สามารถที่จะทดสอบระบบในกรณีที่มีปริมาณทราฟฟิกในเครือข่าย จำนวนมาก ๆ ได้
2. ระบบจะใช้ไอพีเทเบิลเป็นตัวแยกประเภทของทราฟฟิกซึ่งตัวไอพีเทเบิลนั้นจะใช้เทคนิคการ MARK แพ็กเก็ตซึ่งใช้ตาราง mangle ในการจับคู่แพ็กเก็ตซึ่งตัวตารางนี้ไม่สามารถที่จะทำการเรียงลำดับกฎได้
3. การคอนฟิกค่าของ tc นั้นมีการคอนฟิกค่าได้หลากหลายแบบมาก ซึ่งในระบบที่ทำการพัฒนานั้นได้นำความสามารถเพียงบางส่วนมาเท่านั้น ซึ่งก็เหมือนกับการลดความสามารถของระบบเดิมลงนั่นเอง เพียงแต่ทำให้ใช้งานได้ง่ายขึ้นเท่านั้นเอง
4. ระบบนี้เป็นเพียงระบบต้นแบบที่ทำการพัฒนาเป็นครั้งแรกซึ่งยังขาดคุณสมบัติด้านความปลอดภัยและคุณสมบัติอื่น ๆ อยู่บ้าง ซึ่งต้องให้ความสนใจในตัวระบบปฏิบัติการลินุกซ์ และแอปพลิเคชันต่าง ๆ ที่นำมาใช้งานร่วมด้วยให้ลึกซึ้งกว่านี้ ซึ่งจะทำให้ระบบสมบูรณ์แบบมากกว่านี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 ประโยชน์ที่ได้รับจากการออกแบบและพัฒนาระบบ

ประโยชน์ที่ได้รับจากการพัฒนาระบบในโครงการนี้สามารถสรุปได้ดังนี้

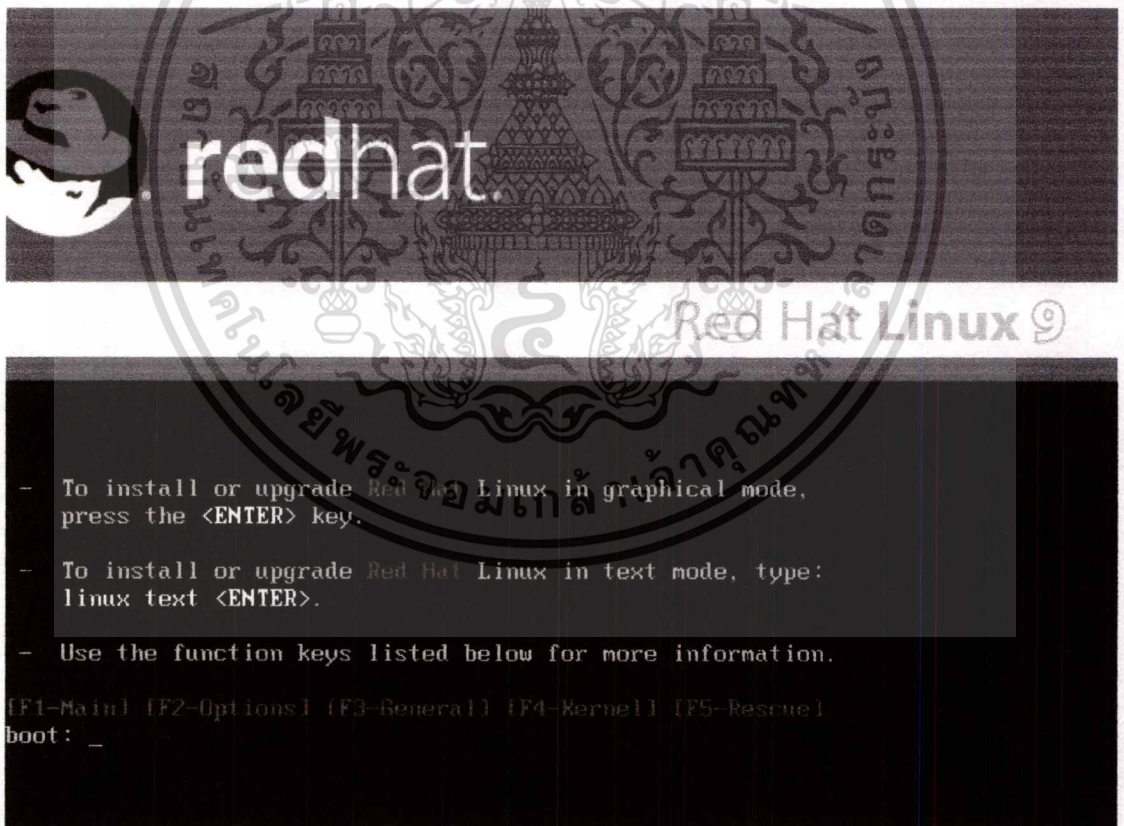
1. ได้เข้าใจในเทคโนโลยี PHP (Professional Home Page) และสามารถนำมาประยุกต์ใช้ในการพัฒนาระบบสารสนเทศได้
2. ได้ความรู้ และความเข้าใจในระบบปฏิบัติการลินุกซ์มากขึ้น สามารถนำความสามารถต่างๆ มาใช้งานได้อย่างมีประสิทธิภาพ
3. ได้ความรู้เกี่ยวกับการทำงานของระบบเครือข่ายมากขึ้น
4. สามารถนำระบบนี้ไปประยุกต์ใช้ให้เป็นประโยชน์ภายในองค์กรต่างๆ ได้
5. สามารถนำหลักการในการพัฒนาสารสนเทศนี้ ไปใช้ในการสร้างระบบสารสนเทศอื่นๆ ที่เป็นประโยชน์ต่อไปได้



ภาคผนวก ก. การติดตั้งระบบ

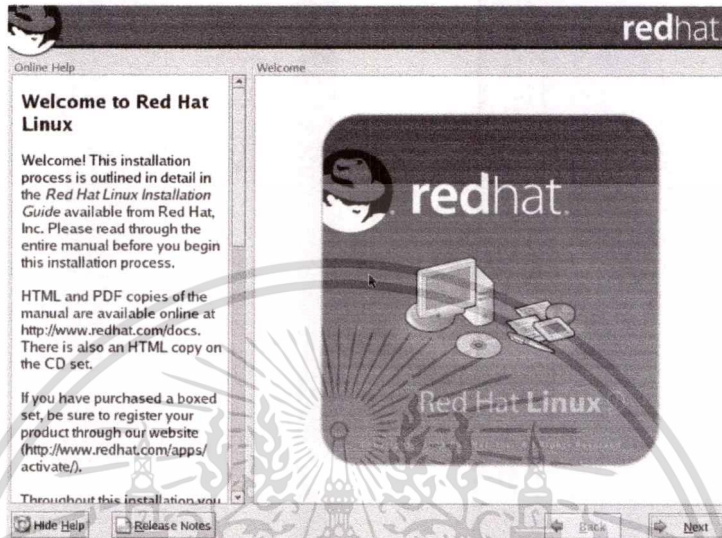
1. การติดตั้งระบบปฏิบัติการ Linux Redhat 9.0

ในการติดตั้งนั้นเราต้องทำการเตรียมเครื่องคอมพิวเตอร์ ที่มีการติดตั้งอินเทอร์เฟซการ์ดมากกว่าหนึ่งอัน เพราะเราจะต้องจำลองให้เครื่องของเราทำงานเป็นเราเตอร์ซึ่งในตัวระบบปฏิบัติการลินุกซ์นั้น สามารถจำลองการทำงานของเราเตอร์ได้ โดยมีความสามารถในการ Forward Packet จากอินเทอร์เฟซการ์ดใบหนึ่งไปอีกใบหนึ่งได้ ซึ่งในการติดตั้งเราสามารถเลือกการติดตั้งอย่างง่าย ๆ โดยการบู๊ตเครื่องและนำ และให้เครื่องบู๊ตจากแผ่นติดตั้ง



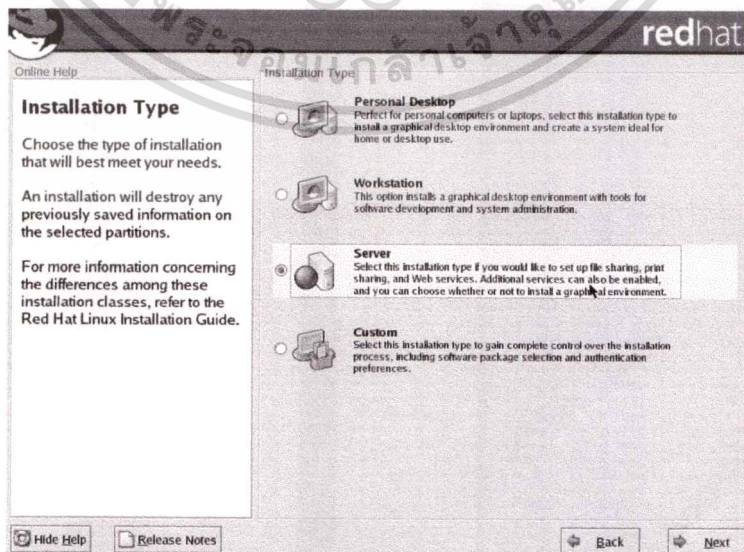
รูปที่ 1 หน้าจอแสดงหลังจากบู๊ตเครื่องจากแผ่นติดตั้ง

หลังจากนั้นให้กด Enter เพื่อที่จะเข้าสู่โหมดในการคอนฟิกค่าของระบบปฏิบัติการซึ่งในที่นี้ เราจะอธิบายในขั้นตอนที่สำคัญเกี่ยวกับระบบเท่านั้น โดยหลังจากนั้นจะปรากฏหน้าจอดังนี้



รูปที่ 2 หน้าจอก่อนเข้าสู่โหมดการคอนฟิกค่า

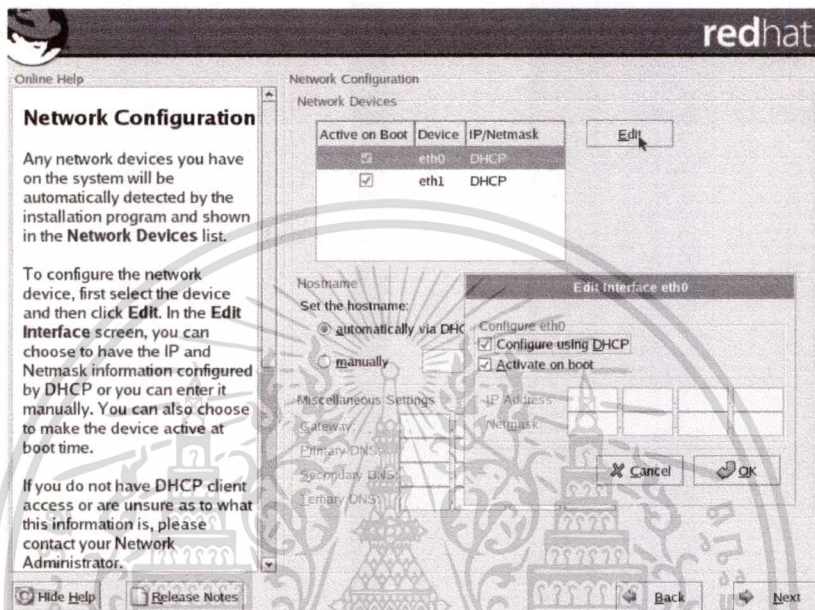
ในการติดตั้งนั้นจะมีการกำหนดค่าพื้นฐานต่าง ๆ เช่น ภาษาที่ใช้, อุปกรณ์ฮาร์ดแวร์, Timezone ซึ่งค่าหลัก ๆ ที่จำเป็นต้องคอนฟิกจะมีรูปแบบในการติดตั้ง (Installation Type) ซึ่งในที่นี้เราจะต้องเลือกให้มีการติดตั้งแบบ Server Type เพื่อให้เครื่องเหมาะกับการทำหน้าที่ให้บริการ เพราะระบบจะต้องมีการให้บริการ Web Server และ MySQL Server ด้วย



รูปที่ 3 การเลือกรูปแบบในการติดตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากนั้นเราจะต้องทำการคอนฟิกค่า IP Address ของอินเทอร์เฟซการ์ดแต่ละใบด้วยซึ่งในที่นี้เราสามารถที่จะคอนฟิกเลขหรือข้ามขั้นตอนไปก่อน แล้วค่อยไปคอนฟิกค่าอีกทีก็ได้หลังจากการติดตั้งเสร็จ

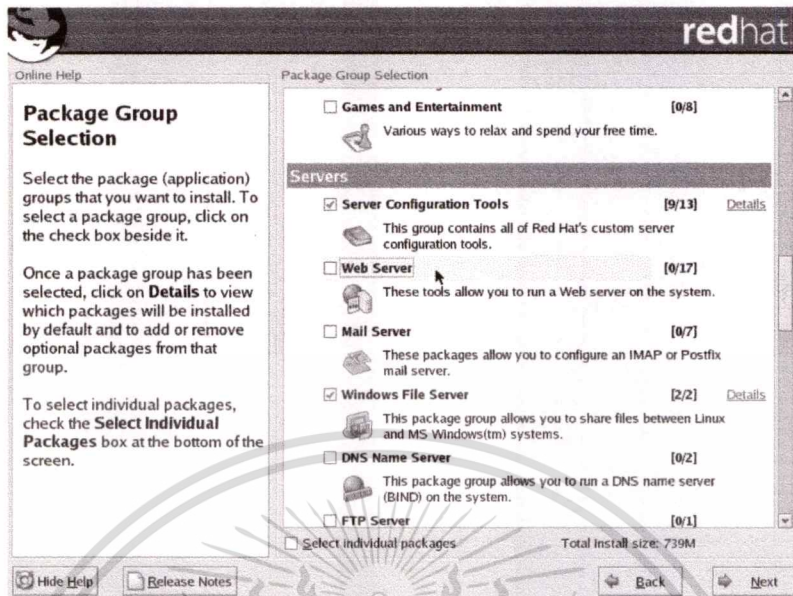


รูปที่ 4 การคอนฟิกค่าอินเทอร์เฟซการ์ด

หลังจากนั้นก็จะต้องทำการเลือก Package หรือตัวโปรแกรมที่จะต้องทำการติดตั้งซึ่งส่วนสำคัญที่จะต้องติดตั้งจะต้องมีดังนี้

- Development Tool ในส่วนนี้จะต้องทำการเพิ่มเพื่อใช้ในการคอมไพล์โปรแกรมบางตัว
- Kernel Development ใช้ในการคอมไพล์เคอร์เนลเพื่อให้ Iptable กรองแพ็กเก็ตในระดับแอปพลิเคชันเลขอร์ได้

และส่วนที่ต้องเลือกออกไปคือ Web Server และ MySQL Server ซึ่งเราจะต้องดาวน์โหลด source Code มาทำการคอมไพล์ติดตั้งใหม่ให้สนับสนุนการทำงานของโปรแกรมเราด้วย



รูปที่ 5 การเลือกติดตั้งแพ็คเกจ

หลังจากนั้นเครื่องจะทำการดำเนินการติดตั้งจนเสร็จเรียบร้อย เครื่องจะทำการบูตใหม่นั้นก็แสดงว่าได้ทำการติดตั้งเสร็จเรียบร้อยแล้ว

2. การติดตั้ง Apache Web Server

หลังจากที่เราทำการติดตั้งระบบปฏิบัติการเรียบร้อยแล้วจะต้องทำการติดตั้ง Web Server เพื่อทำหน้าที่เก็บโปรแกรมและให้เครื่องอื่น ๆ สามารถเรียกใช้โปรแกรมที่พัฒนาด้วย ซึ่งเราจะต้องนำ Source Code มาคอมไพล์ติดตั้งใหม่ด้วย เพราะเราจะต้องให้ Web Server มีสิทธิ์เป็น Root เพื่อให้ระบบทำการรัน System Command ในระบบปฏิบัติการด้วย ซึ่งจะติดตั้งดังนี้

```
[root@router root]# tar xvfz apache_1.3.33.tar.gz
[root@router root]# cd apache_1.3.33
[root@router root]# env CFLAGS="-DBIG_SECURITY_HOLE" ./configure --prefix=/www
--enable-module=so (ทำการ compile และ config ให้รันเป็น user root ได้)
[root@router root]# make
[root@router root]# make install
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราจะทำการติดตั้ง Apache ไว้ในไดเรกทอรี /www เมื่อติดตั้งเสร็จเรียบร้อยแล้วให้แก้ไขค่าคอนฟิกใน /www/bin/httpd.conf ดังนี้เพื่อที่จะให้ Apache รันเป็น root

```
User root
```

```
Group root
```

เมื่อติดตั้งเสร็จเรียบร้อยแล้ว ให้ทำการรันเซอร์วิสก่อนเว็บเซิร์ฟเวอร์จึงจะทำงานได้

```
[root@router root]# /www/bin/apachectl start
```

```
Starting Apache: [OK]
```

3. การติดตั้งภาษา PHP

การจะติดตั้ง PHP ให้ทำงานได้นั้นบนระบบปฏิบัติการลินุกซ์ติดตั้งบริการเว็บเซิร์ฟเวอร์ก่อนจึงจะสามารถทำการติดตั้ง PHP ลงไปได้ ซึ่งการติดตั้งนั้นสามารถดาวน์โหลดไฟล์จากอินเทอร์เน็ตได้ ซึ่งสามารถติดตั้งโดยใช้คำสั่งดังนี้

```
[root@router root]# tar xvfz php-4.3.10.tar.gz
```

```
[root@router root]# cd php-4.3.10/
```

```
[root@router root]# ./configure --with-mysql --with-apxs=/www/bin/apxs
```

```
[root@router root]# make
```

```
[root@router root]# make install
```

```
[root@router root]# cp php.ini-dist /usr/local/lib/php.ini
```

ซึ่งเมื่อทำการติดตั้งเสร็จจะต้องเพิ่มค่าในไฟล์ httpd.conf ด้วย

```
LoadModule php4_module libexec/libphp4.so
```

```
Addtype application/x-httpd-php .php .html
```

```
Addtype application/x-httpd-php-source .phps
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการติดตั้งเสร็จเรียบร้อยแล้วสามารถตรวจสอบการทำงานได้ โดยนำเว็บเพจที่เขียนด้วยภาษา PHP นำไปไว้ในไดเรกทอรี /www/htdocs แล้วลองทำการเรียกจากบราวเซอร์ดู

4. การติดตั้ง MySQL Server

การติดตั้ง MySQL Server นั้นจะต้องดาวน์โหลดไฟล์มา 2 ตัวด้วยกันคือ MySQL-client-4.1.10-0.i386.rpm และ MySQL-server-4.1.10-0.i386.rpm ซึ่งติดตั้งดังนี้

```
[root@router root]# rpm -i MySQL-client-4.1.10-0.i386.rpm
[root@router root]# rpm -i MySQL-server-4.1.10-0.i386.rpm
```

โดยเมื่อทำการติดตั้งเสร็จแล้วก็จะสามารถใช้งานได้เลย โดยไม่ต้องคอนฟิกค่าใด ๆ อีก

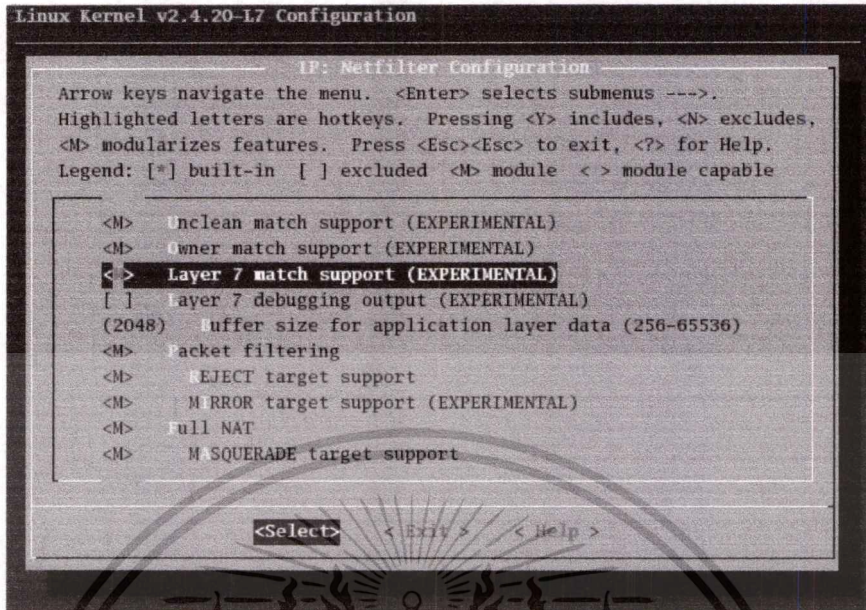
5. การคอนฟิกให้ไอพีเทเบิลรองรับการกรองแพ็กเก็ตในระดับชั้นแอปพลิเคชันเลเยอร์

การจะทำให้ไอพีเทเบิลมีความสามารถนี้จะต้องดาวน์โหลด Patch ต่าง ๆ ดังนี้

- Source Code Iptable (ติดตั้งไอพีเทเบิลใหม่)
- Patch Kernel (ให้เคอร์เนลรองรับการกรองในระดับแอปพลิเคชันเลเยอร์)
- Patch Iptable (ให้ไอพีเทเบิลรองรับการกรองในระดับแอปพลิเคชันเลเยอร์)
- Protocol Defination (ข้อมูลของ Protocol ต่าง ๆ ที่จะสนับสนุนในการกรอง)

ซึ่งเราจะต้องทำการคอมไพล์เคอร์เนลใหม่ซึ่งจะต้อง Install Patch ใหม่ลงใน Source Code ของเคอร์เนลก่อนแล้วทำการเพิ่มค่าคอนฟิกจึงค่อยทำการคอมไพล์ หลังจากนั้นจะต้องลบไอพีเทเบิลของเดิมออกก่อน และทำการ Install Patch ลงใน Source Code ของไอพีเทเบิลแล้วค่อยทำการติดตั้ง หลังจากนั้นจึง Install Protocol Defination ลงไปจึงจะสามารถใช้ความสามารถนี้ได้

```
[root@router root]# cd /usr/src/linux2.4.20/
[root@router linux2.4.20]# patch -p1 < layer7-kernel-v1.0.patch
[root@router linux2.4.20]# make menuconfig
ทำการเพิ่มคอนฟิกให้สนับสนุนการทำงานดังรูปที่ 6
```



รูปที่ 6 เลือก option Layer7 match Support

```
[root@router linux2.4.20-8]# make dep && clean && bzImage
[root@router linux2.4.20-8]# make modules && modules_install
[root@router linux2.4.20-8]# cp System.map /boot/System.map
[root@router linux2.4.20-8]# cp arch/i386/boot/bzImage /boot/vmlinuz-2.4.20-8
```

หลังจากนั้นทำการ reboot เครื่องใหม่

หลังจากนั้นจะต้องทำการติดตั้ง Iptable ใหม่ซึ่งจะต้องทำการลบ Iptable ตัวเก่าที่ทำการติดตั้งไว้ก่อนแล้วออกก่อนถึงจะสามารถทำการติดตั้งใหม่ได้

```
[root@router root]# rpm -e --nodeps iptables-1.2.7a
[root@router root]# tar -jvzf iptables-1.2.9.tar.bz
[root@router root]# cd iptables-1.2.9
[root@router iptable1.2.9]# patch -p1 < netfilter-layer7-v1.0.patch
[root@router iptable1.2.9]# make KERNEL_DIR=/path/to/patched/kernel
[root@router iptable1.2.9]# make install KERNEL_DIR=/path/to/patched/kernel
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

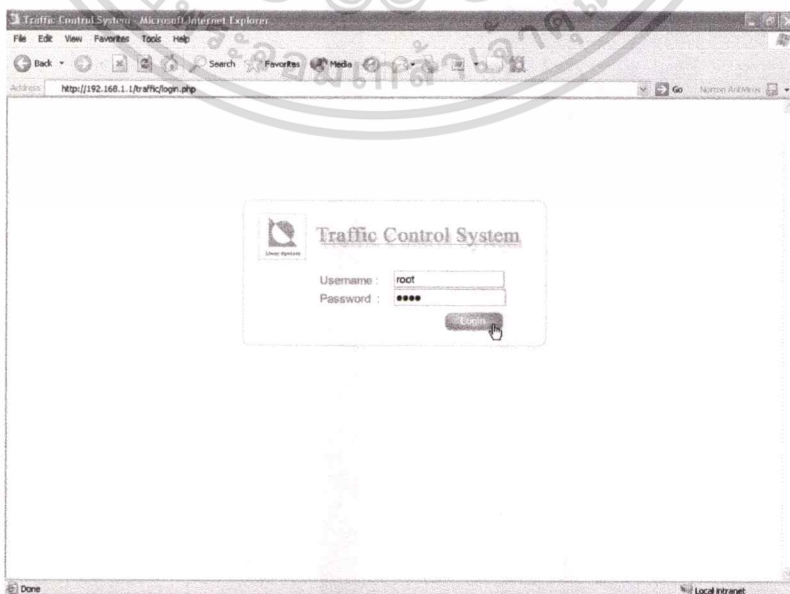
หลังจากนั้นจะต้องทำการติดตั้ง Protocol Defination ซึ่งเป็นไฟล์ที่จะทำให้ไอพีเทเบิลสามารถสนับสนุนการกรองโปรโตคอลต่าง ๆ ในไฟล์นี้ได้ซึ่งเมื่อทำการติดตั้งเสร็จแล้วเราสามารถดูข้อมูล โปรโตคอลต่าง ๆ ที่สนับสนุนได้อีกด้วย ซึ่งจะอยู่ใน Directory /etc/17-protocol/protocol

```
[root@router root]# tar xvfz 17-protocols-2005-02-06.tar.gz
[root@router root]# cd 17-protocols-2005-02-06
[root@router 17-protocols-2005-02-06]# make install
```

เมื่อทำการติดตั้งส่วนประกอบต่าง ๆ เสร็จเรียบร้อยแล้วให้นำไฟล์เตอร์โปรแกรมไปใส่ไว้ใน Directory /www/htdoc/ ส่วนการติดตั้งฐานข้อมูลให้ใช้คำสั่งดังนี้

```
[root@router root]# mysql
mysql > create database traffic;
mysql > exit;
[root@router root]# mysqldump traffic < traffic.sql
```

เมื่อติดตั้งเรียบร้อยแล้ว จะเรียกใช้งานระบบ ก็ไปที่โปรแกรมบราวเซอร์และพิมพ์ไปที่ url หมายเลข IP ของเครื่อง อย่างเช่น <http://192.168.1.1/traffic/login.php>



รูปที่ 7 แสดงหน้าการทำงานของระบบหลังจากติดตั้งเรียบร้อยแล้ว

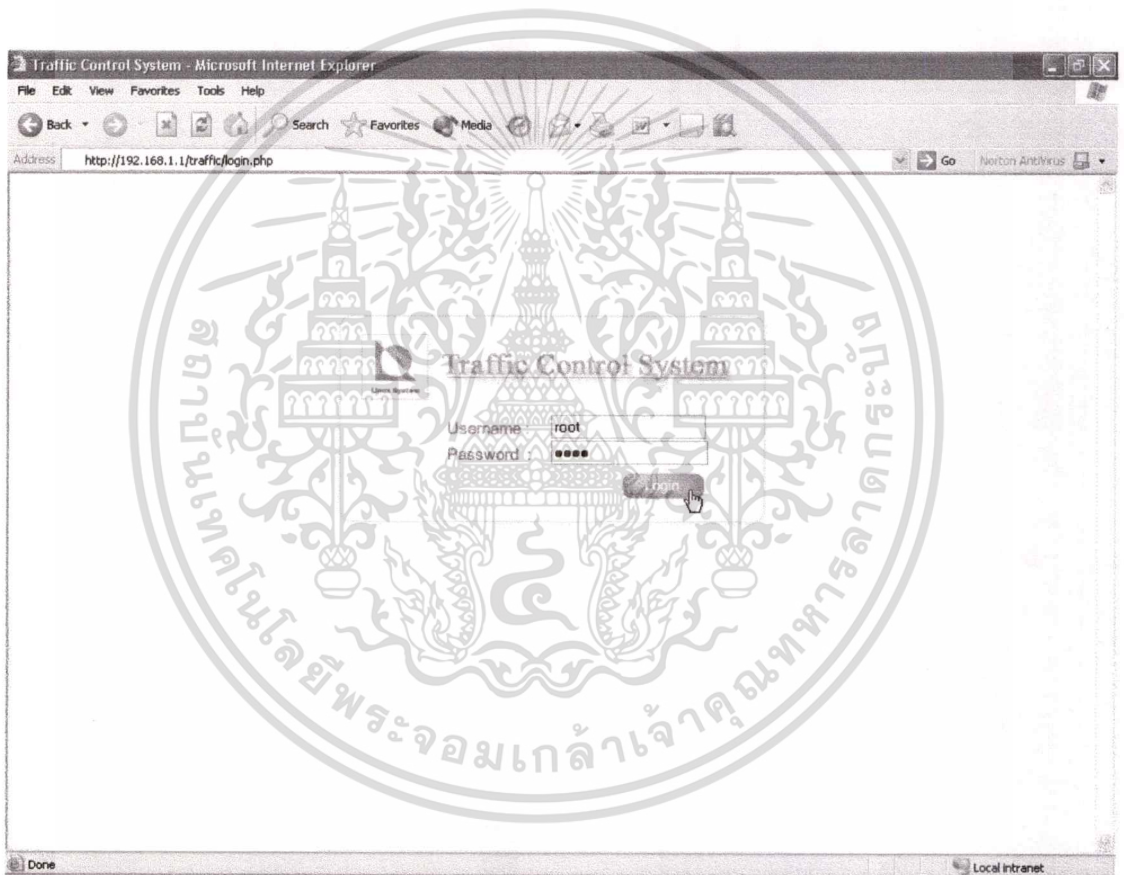
เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการค้าเท่านั้น เมื่อผู้ใดเห็นประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

การใช้งานระบบ

1. การเข้าใช้งานระบบ

ก่อนที่ผู้จะใช้จะเข้าใช้งานระบบ Traffic Control System ได้นั้น ผู้ใช้จะต้องทำการ login เข้าสู่ระบบก่อน โดยการใส่ Username และ Password ที่มีสิทธิ์ในการใช้งานระบบ

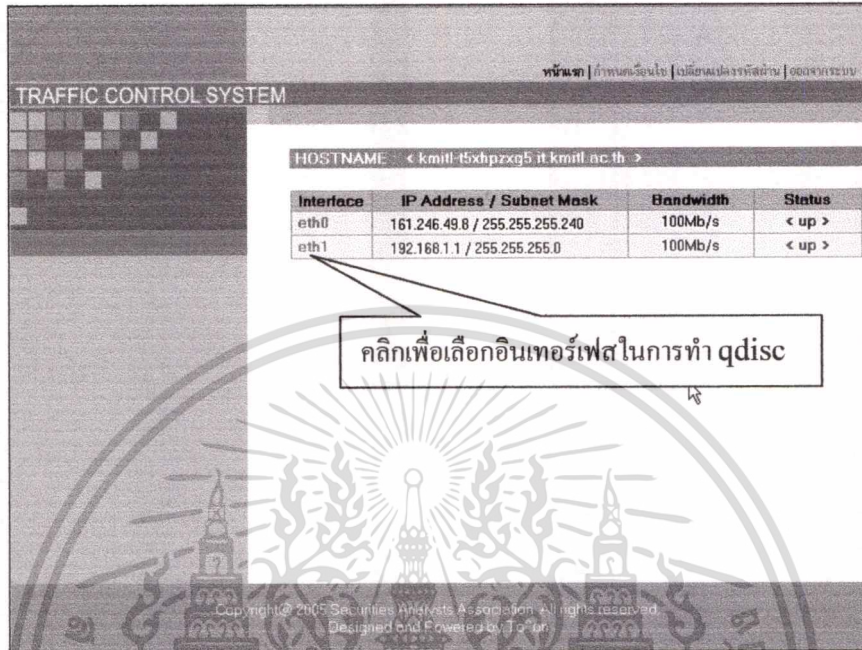


รูปที่ 1 หน้าจอ login ก่อนการเข้าใช้งานระบบ

ถ้าผู้ใช้ใส่ Username หรือ Password ผิด หรือผู้ใช้ไม่มีสิทธิ์ในการเข้าใช้งานระบบนั้น ระบบจะแสดงข้อผิดพลาดให้ผู้ใช้งานทราบ รวมทั้งให้ผู้ใช้งานทำการล็อกอินเพื่อเข้าระบบใหม่อีกครั้ง โดยถ้าผู้ใช้ล็อกอินผ่านแล้ว ระบบจะแสดงหน้าจอหลักของระบบ ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. หน้าแรกของระบบ



รูปที่ 2 หน้าจอหลักของระบบ

ในส่วนหน้าจอหลักจะประกอบด้วยเมนูหลัก ๆ ที่อยู่ด้านขวามือบน ซึ่งสามารถอธิบายความหมายหลัก ๆ ของเมนูดังกล่าวได้ดังต่อไปนี้

- หน้าแรก แสดงรายละเอียดของอินเทอร์เฟซที่อยู่ในระบบพร้อมทั้งข้อมูลในแต่ละอินเทอร์เฟซ เช่น IP Address, Subnet Mask, Bandwidth และ สถานะของอินเทอร์เฟซ และสามารถคลิกเลือกเพื่อที่จะเข้าไปกำหนด qdisc ในแต่ละอินเทอร์เฟซด้วย
- กำหนดเงื่อนไข เป็นการจัดกลุ่มของกฎ หรือ ตัวกรอง เพื่อทำการคัดแยกทราฟฟิกแต่ละประเภทออกไปตามคลาสที่เรากำหนด
- เปลี่ยนแปลงรหัสผ่าน เปลี่ยนรหัสผ่านในการเข้าใช้ระบบ
- ออกจากระบบ ทำการออกจากระบบ

3. การกำหนดเงื่อนไข

ในการกำหนดเงื่อนไขนั้นสามารถที่จะจัดกลุ่มของเงื่อนไขออกเป็นสองกลุ่มหลัก ๆ ด้วยกัน คือ ช่วงแอดเดรส (IP Address) และ บริการ (Port) โดยการจัดกลุ่มนี้จะช่วยให้ผู้ใช้ทำการเพิ่มเงื่อนไขจำนวนมาก ๆ ได้ง่ายขึ้น และยังช่วยให้ตรวจสอบรายละเอียดของเงื่อนไขที่เพิ่มเข้าไปได้อย่างสะดวกขึ้นอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 การกำหนดช่วงแอดเดรส

การกำหนดช่วงแอดเดรสนั้นเป็นส่วนหนึ่งในการกำหนดเงื่อนไขของแบ่งประเภททราฟฟิกออกไปในแต่ละคลาส ซึ่งในส่วนช่วงแอดเดรสนั้นจะสร้างไว้เพื่อเอาไว้เป็นเงื่อนไขในส่วนของ Source IP Address และ Destination IP Address ซึ่งในการกำหนดช่วงแอดเดรสนั้นจะกำหนดได้ 3 แบบ คือ

- **Host** ระบุถึง IP Address เพียงเครื่องเดียว คือมีสับเน็ตมาสังกัดเป็น 32 เท่านั้น
- **Network** ระบุ IP Address เป็นช่วง คือ สามารถที่จะกำหนด Subnet Mask ได้นั่นเอง
- **Group Network** สามารถทำการรวมกลุ่มของ Host และ Network ที่ทำการเพิ่มไว้ให้เป็นกลุ่ม ๆ ได้นั่นเอง

ชื่อ : Web Server

ชนิด : Host

แอดเดรส : 161 . 246 . 38 . 25

คำอธิบาย : IT Web Server

เพิ่ม ยกเลิก

รูปที่ 3 การเพิ่มเงื่อนไขแบบ Host

ชื่อ : Internal

ชนิด : Network

แอดเดรส : 192 . 168 . 1 . 0 / 24 (256 IPs)

คำอธิบาย : Lab666

เพิ่ม ยกเลิก

รูปที่ 4 การเพิ่มเงื่อนไขแบบ Network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อ : Group_Server

ชนิด : Group Network

กลุ่ม : Any, DNS Server, Internal, IT-Web, Nectec, Proxy

คำอธิบาย : Server Only

ปุ่ม: เพิ่ม, ยกเลิก

รูปที่ 5 การเพิ่มเงื่อนไขแบบ Group Network

เมื่อเราทำการเพิ่มช่วงแอดเดรสไปเรียบร้อยแล้ว เราสามารถที่จะข้อมูลที่เรทำการเพิ่มไว้แล้วได้ที่ เมนูรายละเอียดช่วงแอดเดรส ซึ่งในส่วนรายละเอียดที่แสดงเราสามารถที่จะการลบข้อมูลที่เรทำการเพิ่มไว้แล้วได้นั่นเอง

ชื่อ	ค่าที่กำหนด	อธิบาย	
Any	0.0.0.0/0	All IP	ลบ
DNS Server	192.168.1.12		ลบ
Internal	192.168.1.0/24	LAB 222	ลบ
IT-Web	161.246.38.25	...	ลบ
Nectec	203.185.132.73	...	ลบ
Proxy	192.168.1.13	proxy LAB	ลบ
Thaiware	203.146.140.101	ลบ
Toon	192.168.1.11	Test	ลบ
Web Server	192.168.1.11		ลบ
Group1	- Web Server - DNS Server - Proxy	Server Only	ลบ

รูปที่ 6 หน้าจอแสดงการจัดกลุ่ม Host, Network , Group Network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การกำหนดบริการ

การกำหนดบริการนั้นเป็นส่วนหนึ่งในการกำหนดเงื่อนไขของแบ่งประเภทกราฟฟิก ออกไปในแต่ละคลาส ซึ่งในส่วนการกำหนดบริการนั้นจะสร้างไว้เพื่อเอาไว้เป็นเงื่อนไขในส่วนของการกำหนด Service ของกราฟฟิก เช่น Port 80 เป็นบริการของ Web และ port 20,21 เป็นบริการของ FTP ซึ่งในการกำหนดบริการนั้นจะกำหนดได้ 2 แบบ คือ

- **Service** เราสามารถที่จะทำการกำหนดชื่อของบริการที่เราจะตั้งได้ เลือกโปรโตคอลของบริการนั้น ซึ่งจะมีเพียงโปรโตคอล TCP, UDP และ ICMP เท่านั้น ส่วน port นั้นเราสามารถทำการกำหนดได้ทั้ง Source Port และ Destination Port ซึ่งสามารถกำหนดเป็นเบอร์พอร์ตเพียงพอร์ตเดียว หรือสามารถที่จะกำหนดเบอร์พอร์ตเป็นช่วงก็ได้ ซึ่งช่วงนั้นสามารถอยู่ได้ในช่วง 1-65535 เท่านั้น

กำหนดช่วงแอดเดรส | รายละเอียดช่วงแอดเดรส | กำหนดบริการ | รายละเอียดบริการ

ชื่อ :	HTTP
ชนิด :	Service
โปรโตคอล :	TCP
Src/Dest Port :	80 : 1 / 1 : 65535
คำอธิบาย :	Web...
<input type="button" value="เพิ่ม"/> <input type="button" value="ยกเลิก"/>	

รูปที่ 7 หน้าจอแสดงการจัดกลุ่มบริการแบบพอร์ตเดียว

ชื่อ :	FTP
ชนิด :	Service
โปรโตคอล :	TCP
Src/Dest Port :	20 : 21 / 1 : 65535
คำอธิบาย :	ftp data & connection
<input type="button" value="เพิ่ม"/> <input type="button" value="ยกเลิก"/>	

รูปที่ 8 หน้าจอแสดงการจัดกลุ่มบริการแบบเป็นช่วงพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Group Service** สามารถทำการรวมกลุ่มของ Service ที่ทำการเพิ่มไว้ให้เป็นกลุ่ม ๆ ใด
นั่นเอง

ชื่อ : Group_service1

ชนิด : Service Group

กลุ่ม : All_TCP
DNS
FTP
HTTP
Ping

คำอธิบาย :

เพิ่ม ยกเลิก

รูปที่ 9 หน้าจอแสดงการจัดกลุ่มแบบ Service Group

เมื่อเราทำการเพิ่มบริการ ไปเรียบร้อยแล้ว เราสามารถที่จะข้อมูลที่เราทำการเพิ่มไว้แล้วได้ที่เมนูรายละเอียดบริการ ซึ่งในส่วนรายละเอียดที่แสดงเราสามารถที่จะการลบข้อมูลที่เราทำการเพิ่มไว้แล้วได้นั่นเอง

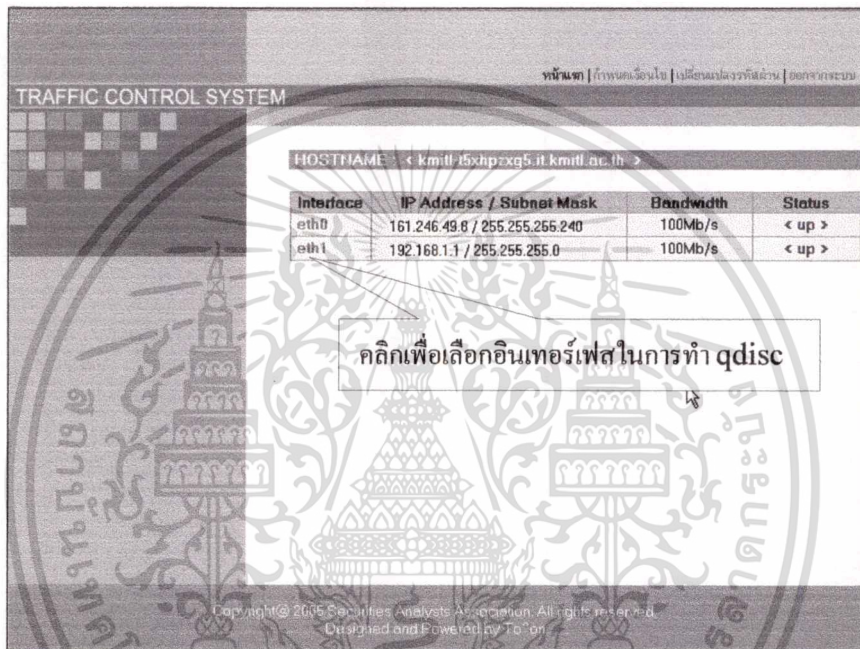
กำหนดช่วงแอดเดรส รายละเอียดช่วงแอดเดรส กำหนดบริการ รายละเอียดบริการ				
ชื่อ	โพรโทคอล	Src / Dest Port Group Details	อธิบาย	
All_TCP	TCP	1:65535 / 1:65535	All	ลบ
DNS	TCP/UDP	53 / 1:65535	ลบ
FTP	TCP	20:21 / 1:65535	ftp data & connection	ลบ
HTTP	TCP	80 / 1:65535	web only...	ลบ
Ping	ICMP	-	...	ลบ
Group1	-	- HTTP - FTP	ลบ

รูปที่ 10 หน้าจอแสดงการจัดกลุ่ม Service และ Service Group

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. การคอนฟิกค่าแต่ละอินเทอร์เฟซ

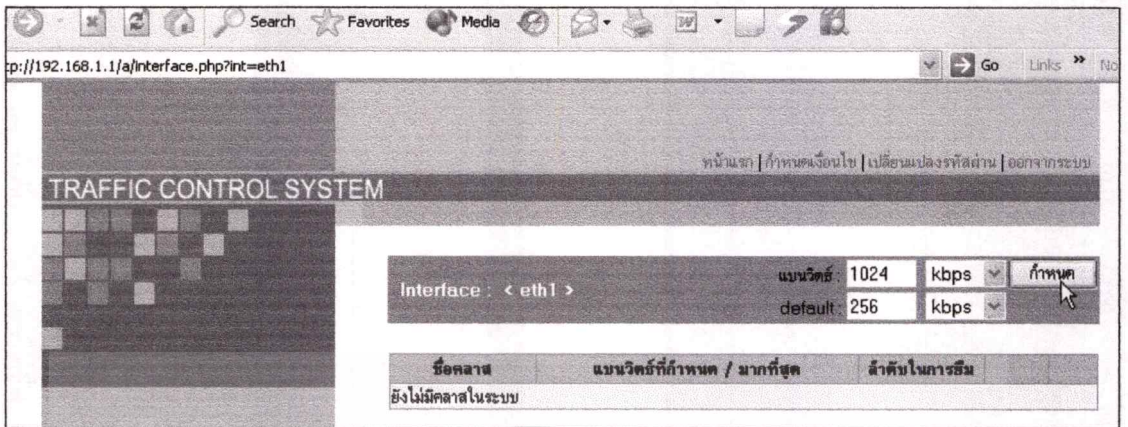
เมื่อทำการกำหนดเงื่อนไขเรียบร้อยแล้วเราก็สามารถที่จะนำเงื่อนไขที่เรากำหนดไว้ไปทำการสร้างกฎเพื่อแบ่งประเภทของทราฟฟิกได้แล้ว ดังนั้นเราจะต้องกลับมาที่หน้าแรกเพื่อที่จะทำการเลือกอินเทอร์เฟซที่เราต้องการที่จะกำหนดอัตราแบนวิดท์ โดยที่คลิกที่ชื่ออินเทอร์เฟซที่เราต้องการจะจัดการ



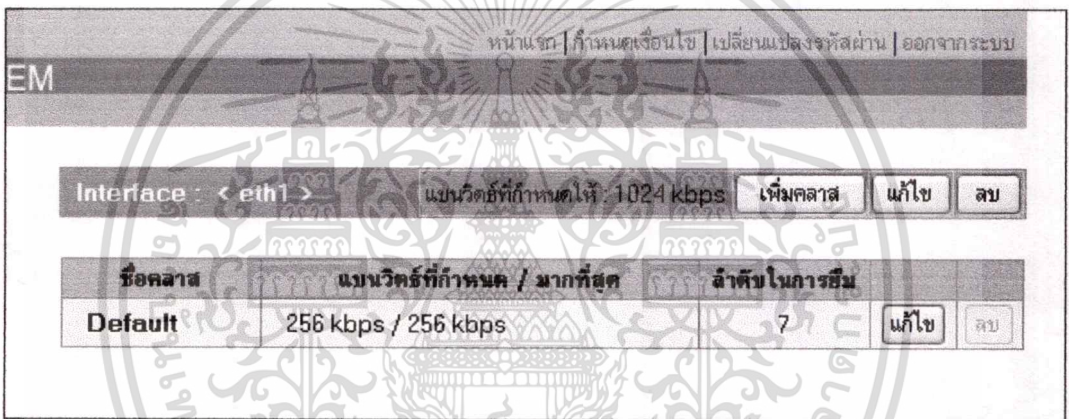
รูปที่ 11 คลิกที่ชื่อของอินเทอร์เฟซที่เราต้องการจะกำหนด

จากนั้นจะปรากฏหน้าจอในการคอนฟิกค่าของอินเทอร์เฟซนั้นขึ้นมา ซึ่งค่าเริ่มต้นที่เราจะต้องใส่เข้าไปในหน้านี้ ในกรณีที่เรายังไม่มีการกำหนดค่าใด ๆ เลย ในอินเทอร์เฟซนี้เลย คือจะต้องมีการใส่ค่าอัตราแบนวิดท์ของอินเทอร์เฟซนั้น ว่าเราต้องการที่จะให้อินเทอร์เฟซนั้นใช้อัตราแบนวิดท์ในช่วงเท่าใด ซึ่งสามารถที่จะกำหนดได้ในหน่วย kbps และ Mbps และอีกค่าหนึ่งก็คือ ค่าดีฟอลท์ซึ่งเป็นการสร้างคลาสหลักที่จะต้องมีในระบบ ซึ่งคลาสนี้จะทำหน้าที่เมื่อในระบบยังไม่มีการสร้างคลาสใด ๆ ขึ้นมาทราฟฟิกก็จะวิ่งออกที่คลาสดีฟอลท์นี้ หรือเมื่อมีการสร้างคลาสขึ้นมาแล้วทราฟฟิกใดที่ไม่ตกอยู่ในคลาสใดเลยก็จะตกอยู่ในคลาสดีฟอลท์นั่นเอง ซึ่งไม่สามารถที่จะกำหนดค่าได้เกินกว่าค่าแบนวิดท์ของอินเทอร์เฟซนั้น และเมื่อทำการสร้างคลาสดีฟอลท์ขึ้นมาแล้ว ก็ไม่สามารถที่จะทำการลบคลาสนี้ได้แต่สามารถที่จะทำการแก้ไขอัตราแบนวิดท์ได้ เมื่อเราใส่ค่าเรียบร้อยแล้วกดปุ่มกำหนด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

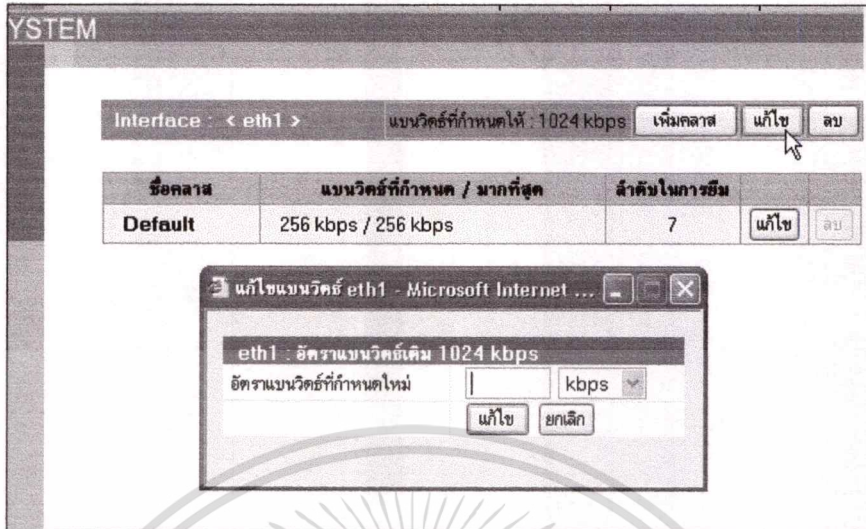


รูปที่ 12 หน้าจอแสดงการคอนฟิกในอินเทอร์เน็ต



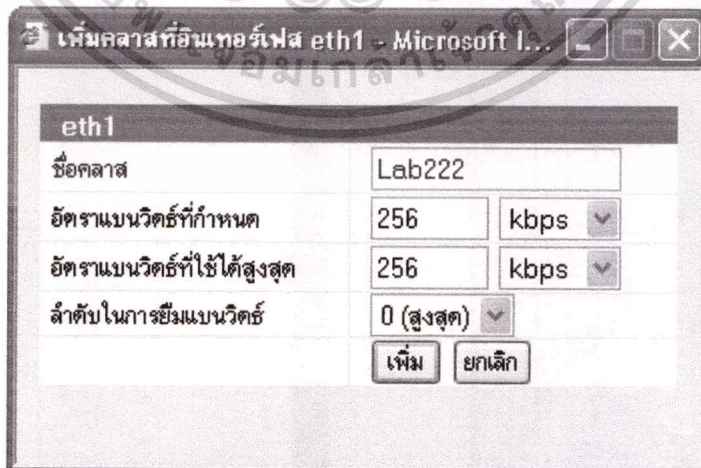
รูปที่ 13 แสดงคลาสดีฟอลท์หลังจากกำหนดที่อินเทอร์เน็ตนั้นแล้ว

เมื่อเราทำการกำหนดค่าแบนวิธที่อินเทอร์เน็ตนั้นแล้วเราสามารถที่จะทำการแก้ไขค่าแบนวิธได้โดยสามารถกดปุ่มแก้ไขก็จะปรากฏหน้าจอ Pop Up ขึ้นมาให้เรากำหนดค่าใหม่ และเราก็สามารถที่จะลบ ค่าที่เรากำหนดไว้แล้วในอินเทอร์เน็ตนั้นได้ โดยการกดปุ่มลบ ซึ่งในการกดปุ่มลบนี้ อาจจะต้องระวังในการกดด้วยเนื่องจากจะทำให้ค่าต่าง ๆ ที่เรากำหนดไว้ในอินเทอร์เน็ตนั้นถูกลบออกไปหมดด้วย ซึ่งเมื่อเรากดปุ่มลบนั้น จะมีข้อเตือนให้เราทราบก่อนจะการลบเพื่อยืนยันอีกทีหนึ่ง ซึ่งการเพิ่มหรือการแก้ไขแบนวิธของอินเทอร์เน็ตนั้นไม่สามารถที่จะกำหนดค่าของแบนวิธได้เกินค่าของ อัตราความเร็วที่อินเทอร์เน็ตสการ์คนั้นทำได้



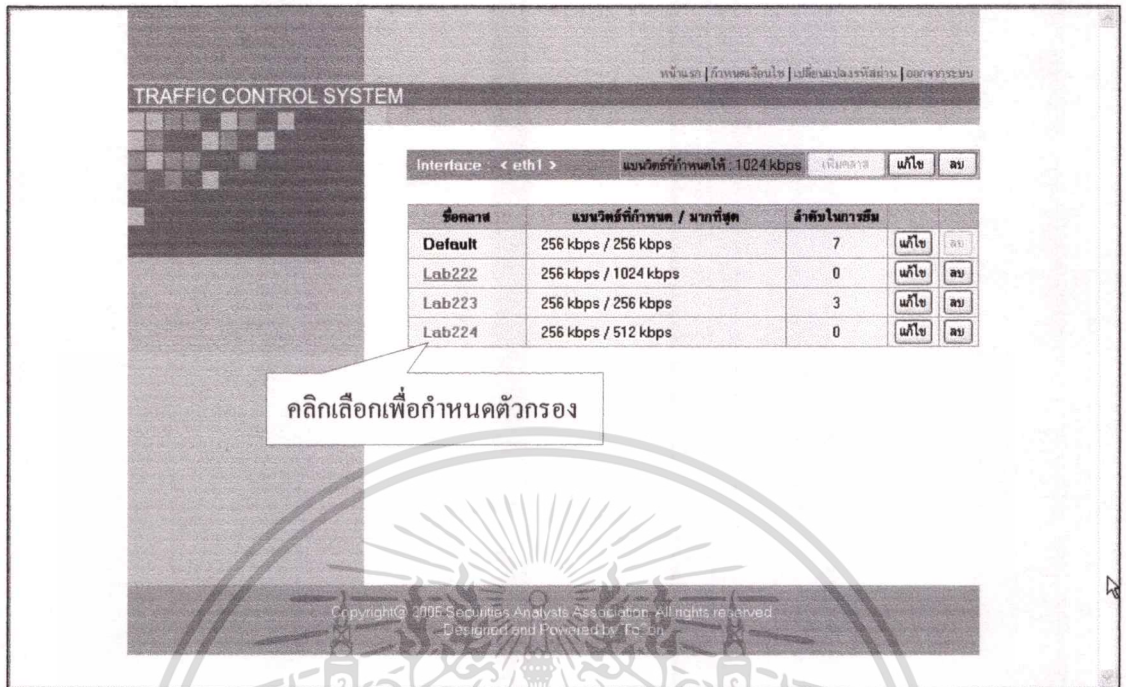
รูปที่ 14 แสดงการแก้ไขค่าแบนด์วิดท์ของอินเทอร์เน็ต

เมื่อเราต้องการที่จะทำการสร้างคลาสให้เราเลือกกดที่ปุ่มสร้างคลาสซึ่งเมื่อกดที่ปุ่มแล้วจะมีหน้าจอ Pop Up ขึ้นมาให้เราได้ใส่รายละเอียดของคลาสนั้น ๆ ซึ่งเราสามารถใส่ชื่อคลาส แบนด์วิดท์ของคลาสนั้น ส่วนอัตราแบนด์วิดท์ที่ใช้ได้สูงสุดนั้นเป็นค่าที่บอกว่าคลาสนั้นสามารถมีอัตราการใช้แบนด์วิดท์ได้มากที่สุดเท่าไร ซึ่งค่าในส่วนนี้ถ้ามากกว่าค่าอัตราแบนด์วิดท์ที่กำหนดให้คลาสนั้นแบนด์วิดท์ส่วนเกินที่จะนำมาใช้ก็จะยืมมาจากคลาสนั้น ๆ แทนซึ่งคลาสนั้นจะต้องไม่มีการใช้งานอยู่จึงจะสามารถนำมาใช้ได้ และกำหนดลำดับความสำคัญในการยืมแบนด์วิดท์ซึ่งสามารถกำหนดค่าได้ 7 ระดับ ซึ่งค่า 0 คือค่าที่มีลำดับความสำคัญมากที่สุด

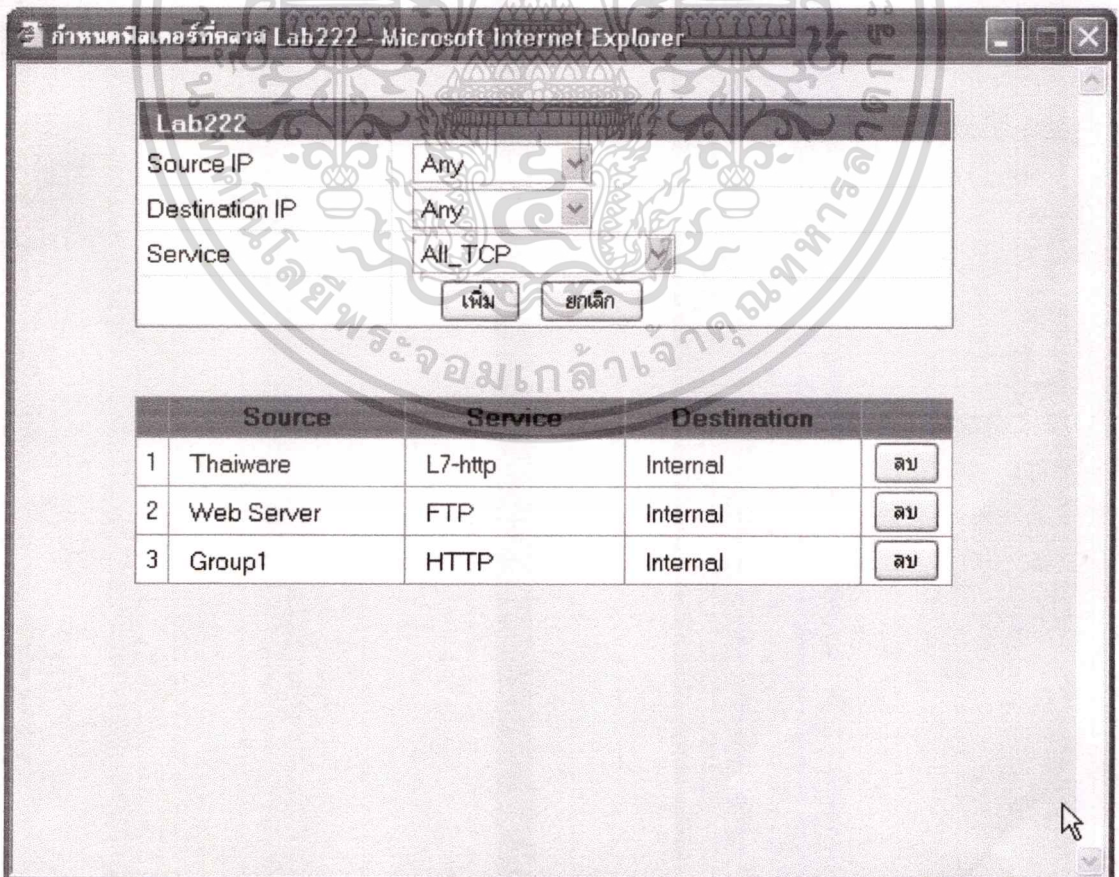


รูปที่ 15 หน้าจอแสดงการเพิ่มคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 16 หน้าจอแสดงคลาสที่มีทั้งหมดในอินเทอร์เน็ตเฟส



รูปที่ 17 หน้าจอแสดงการเพิ่มฟิลเตอร์และแสดงฟิลเตอร์ในแต่ละคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

- กิตติศักดิ์ เจริญ โภคานนท์. 2547. **กัมภีร์การสร้าง E-Commerce Application PHP4**.
 กรุงเทพฯ: ซีเอ็ดยูเคชั่น.
- ก่อกิจ วีระอาชากุล. 2545. **ติดตั้งและปรับแต่งเซิร์ฟเวอร์ Linux สำหรับ Admin Linux โดยเฉพาะ**.
 นนทบุรี: อิน โฟเพรส.
- ธวัชชัย ชมศิริ. 2547. **ติดตั้ง/ดูแล ระบบเครือข่ายคอมพิวเตอร์อย่างมืออาชีพ**.
 กรุงเทพฯ: ซีเอ็ดยูเคชั่น.
- บัณฑิต จามรภูติ. 2547. **กัมภีร์ REDHAT ENTERPRISE เล่ม 2**.
 กรุงเทพฯ: Bandhit press.
- Bert Hubert. 2004. **Linux Advanced Routing & Traffic Control HOWTO**. [Online]. Available:
<http://lartc.org/howto/>.
- Ethan Sommer. 2005. **Application Layer Packet Classifier for Linux**. [Online]. Available:
<http://17-filter.sourceforge.net>.
- Leonardo Balliache. 2003. **Differentiated Service on Linux HOWTO**. [Online]. Available:
<http://opalsoft.net/qos/>.
- Martin A.Brown. 2003. **Traffic Control HOWTO**. [Online]. Available:
<http://www.tldp.org/HOWTO/Traffic-Control-HOWTO/>.
- Martin A.Brown. 2004. **Traffic Control using tcng and HTB HOWTO**. [Online]. Available:
<http://www.tldp.org/HOWTO/Traffic-Control-tcng-HTB-HOWTO/>.
- Martin Devera. 2002. **HTB Linux queuing discipline manual – user guide**. [Online]. Available:
<http://luxik.cdi.cz/~devik/qos/htb/manual/userg.htm>.
- My php.net Group. 2005. **Installation PHP on UNIX Systems**. [Online]. Available:
<http://www.php.net/manual/en/install.unix.php>.
- Oskar Andreasson. 2003. **Iptables Tutorial 1.1.19**. [Online]. Available:
<http://iptables-tutorial.frozentux.net/iptables-tutorial.html>.

ประวัติผู้เขียน

ชื่อผู้เขียน	นายอดิษฐ์ ศิลารมย์
วันเกิด	11 กุมภาพันธ์ 2524
สถานที่เกิด	ฉะเชิงเทรา
วุฒิการศึกษาระดับปริญญาตรี	วท.บ. (วิทยาการคอมพิวเตอร์) คณะวิทยาศาสตร์ มหาวิทยาลัยกรุงเทพ ฯ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้