สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบจัดการปริมาณทราฟฟิกของเรียลไทม์แอพพลิเคชั่น

BANDWIDTH MANAGEMENT SYSTEM
FOR REAL TIME APPLICATION

โดย

ทศพร อภิญญาปัญจะ
พรทิพย์ เลยบุญ
ศศิวัณย์ จิตพึงธรรม

อาจารย์ที่ปรึกษา

อาจารย์ลภัส ประดิษฐ์ทัศนีย์

เลขหมู่..........073071
เลขทะเบียน..........
วัน,เดือน,ปี.......- 2 ก.ค. 2550

.b...11780095...
.i..........................

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ภาคเรียนที่2 ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับก... ญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BANDWIDTH MANAGEMENT SYSTEM

FOR REAL TIME APPLICATION

THOSSAPRON APINYAPANJA

PORNTHIP LOEIBOON

SASIWAN JITPHUNGTHAM

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION TECNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2/2006

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COPYRIGHT 2007

FACULTY ON INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบรับรองปริญญานิพนธ์ ประจำปีการศึกษา 2549

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบจัดการปริมาณทราฟฟิกของเรียลไทม์แอพพลิเคชั่น

Bandwidth Management System for Real Time Application

ผู้จัดทำ

1.ทศพร อภิญญาปัญจะ รหัสประจำตัว 46060013

2. นางสาวพรทิพย์ เลยบุญ รหัสประจำตัว 46060075

3. นางสาวศศิวัณย์ จิตพึ่งธรรม รหัสประจำตัว 46060080

....อาจารย์ที่ปรึกษา

(อาจารย์ลภัส ประดิษฐ์ทัศนีย์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ	ระบบจัดการปริมาณทราฟฟิกของเรียลไทม์แอพพลิเคชั่น
นักศึกษา	นายทศพร อภิญญาปัญจะ รหัสประจำตัว 46060013
	นางสาวพรทิพย์ เลยบุญ รหัสประจำตัว 46060075
	นางสาวศศิวัณย์ จิตพึ่งธรรม รหัสประจำตัว 46060080
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
ปีการศึกษา	2549
อาจารย์ที่ปรึกษา	อาจารย์ลภัส ประดิษฐ์ทัศนีย์

บทคัดย่อ

โครงงานนี้เป็นการพัฒนาแอพพลิเคชั่นเพื่อใช้จัดการแบนด์วิดท์ในเครือข่ายผ่านระบบเว็บไซต์ โดยมีการเน้นไปที่การให้บริการที่เกี่ยวข้องกับสื่อประสม (Multimedia) มากขึ้น เพื่อให้การสื่อสารของโปรแกรมประยุกต์ประเภทต่างๆให้ได้รับบริการอย่างดีที่สุด, เท่าเทียมกันและเกิดประโยชน์สูงสุด

วัตถุประสงค์หลักของโครงงานคือ สามารถจัดการการสลับอัลกอลิธึมเทคนิคการควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์ได้อย่างอัตโนมัติให้เหมาะสมกับสถานะการณ์ต่างๆ ระหว่างเทคนิค HTB (Hierarchical Token Bucket) ซึ่งสามารถรับประกันค่าแบนด์วิดท์ต่ำสุดและแบนด์วิดท์สูงสุดที่กำหนดและเทคนิค HFSC (Hierarchical Fair Service Curve) ซึ่งเป็นเทคนิคการควบคุมและจัดสรรปริมาณข้อมูลบนเครือข่ายเช่นเดียวกันกับ HTB แต่มีการระบุถึงการรับประกันค่าแบนด์วิดท์และค่าความหน่วงของทราฟฟิก เพราะว่าในปัจจุบันคุณภาพการให้บริการ (QoS) ของทราฟฟิกแบบเรียลไทม์เป็นปัจจัยสำคัญที่จำเป็นต้องนำมาพิจารณา

นอกจากนั้นโครงงานยังสนับสนุนการจัดการและควบคุมปริมาณทราฟฟิกบนเครือข่ายผ่านระบบเว็บไซต์ สามารถกำหนดการกรองทราฟฟิกโดยใช้พอร์ต ไอพีแอดเดรส โปรโตคอลระดับชั้นทรานสปอร์ต และ ระดับชั้นแอพพลิเคชั่น ซึ่งช่วยให้ผู้ใช้ที่ไม่มีความรู้เกี่ยวกับกลไกลการควบคุมปริมาณทราฟฟิกและการใช้งานชุดคำสั่ง TC สามารถเข้าใจและใช้งานได้ง่าย นอกจากนั้นยังสามารถแสดงข้อมูลการควบคุมปริมาณทราฟฟิกผ่านการแสดงผลทางกราฟที่เข้าใจได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Bandwidth Management System for Real Time Application
Student	Mr. Thossapron Apinyapanja Student ID. 46060013
	Ms. Pornthip Loeyboon Student ID. 46060075
	Ms. Sasiwan Jitphungtham Student ID. 46060080
Degree	Bachelor of Science
Program	Information Technology
Academic Year	2006
Advisor	Mr. Lapas Pradittasnee

ABSTRACT

This project is a network traffic shaper which provides a Web Interface for support of Bandwidth Management in network.

Development for automatic manage appropriate traffic situation that consider not only data traffic but also consider in Quality of Service (QoS) of Real-Time traffic by using Linux traffic control technique algorithm between HTB algorithm, which guarantee maximum and minimum bandwidth in each class and HFSC algorithm, which also consider in link-sharing and guarantee real time service with delay and bandwidth allocation.

Project also provides a web interface which let users define bandwidth filter based on IP, port, protocol, Layer7-filter and let users learn and use the traffic shaping mechanism. This should be possible for everyone who has no deeper knowledge of Linux and difficult syntax of the tc pcommand. It also draws statistic graph about the current bandwidth usage and traffic statistic.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานพัฒนาระบบงานนี้ ประสบความสำเร็จได้ด้วยความกรุณาจากอาจารย์ที่ปรึกษา อาจารย์ลภัส ประดิษฐ์ทัศนีย์ ที่ให้ความช่วยเหลือ ให้คำชี้แนะช่วยแก้ปัญหาตลอดจนให้ความรู้และประสบการณ์ที่ดีแก่ข้าพเจ้า

ขอขอบคุณ Martin A. Brown ผู้คิดค้นอัลกอลิทึม HTB และ Patrick McHardy และ Andy Furniss ผู้คิดค้นอัลกอลิทึม HFSC ในการให้ความรู้หลักการทำงานของอัลกอลิทึมผ่านระบบ Mailing List จาก lartc@mailman.ds9a.nl

ขอขอบคุณผู้สรรสร้างมิวสิควีดีโอ Crescendo "ดินแดนแห่งความรัก" ที่ทำให้ห้องโปรเจคเป็นดินแดนแห่งความรักจริงๆ และ "She will be loved" จาก Maroon5 สำหรับความเพลินเพลินที่ล้นเหลือตลอดการทำการทดลอง

ขอขอบคุณ "น้อง Shappo" ที่ทำให้ใจเต้นรัวทุกครั้งที่ทำการทดลอง "น้อง Perly Berry" ที่เต็มไปด้วยไวรัส และ Spyware "น้อง As-Jung" สำหรับ MMS Server ที่ควบคุมไม่ค่อยได้ "น้อง As-Kung" ที่อาศัยเป็น monitor ในครั้งคราว "น้อง Q-Kung" สำหรับแรม1 GHz น่าอิจฉาจิงๆ " เครื่อง Dell ทั้งหลาย" ที่เพื่อนๆ อุตสาห์ให้ยืมมาทำการทดลอง และ ที่ขาดไม่ได้ คือ "น้องเครื่องขาว CPU 1GHz" ที่สามารถสร้างทราฟฟิกได้มากมายอย่างไม่น่าเชื่อ สำหรับร่วมทุกข์ร่วมสุขกันตลอด 1 ปีเต็ม

สำหรับคุณงามความดีอันใดที่เกิดจากวิทยานิพนธ์ฉบับนี้ ข้าพเจ้ามอบให้กับบิดามารดา ซึ่งเป็นที่รักและเคารพยิ่ง ตลอดจนครูอาจารย์ที่เคารพทุกท่านที่ได้ประสิทประสาทวิชาความรู้และถ่ายทอดประสบการณ์ที่ดีให้แก่ข้าพเจ้า

มหาวิทยาลัยพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย..I
บทคัดย่อภาษาอังกฤษ..II
กิตติกรรมประกาศ...III
สารบัญ...IV
สารบัญตาราง...IX
สารบัญรูป..X
บทที่ 1 บทนำ...1
1.1 ความเป็นมาและความสำคัญของปัญหา...1
1.2 จุดมุ่งหมายและวัตถุประสงค์ของการศึกษา...2
1.3 ขอบเขตของโครงงาน...3
1.4 ขั้นตอนของการศึกษา...3
1.5 ประโยชน์ที่คาดว่าจะได้รับ..3
บทที่ 2 ทฤษฎีพื้นฐานที่ใช้ในการวิจัย และระบบควบคุมปริมาณทราฟฟิก.....................5
2.1 สาเหตุและแนวคิดการควบคุมปริมาณทราฟฟิก...5
2.2 การควบคุมปริมาณข้อมูล..6
2.3 ส่วนประกอบของการควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลีนุกซ์..............7
2.3.1 ข้อกำหนดของคิว...8
2.3.2 คลาส..9
2.3.3 ตัวกรองแพ็กเก็ต...11
2.3.3.1 ไอพีเทเบิ้ล...12
2.3.3.2 เน็ตฟิวเตอร์...13
2.3.3.3 เลเยอร์7-เน็ตฟิวเตอร์..13
2.4 การควบคุมปริมาณข้อมูลด้วยเทคนิคโทเค็น/บักเก็ตแบบลำดับชั้น.....................14
2.4.1 แนวคิดโทเค็น/บักเก็ต..14
2.4.2 แนวคิดการทำงานของเทคนิคโทเค็น/บักเก็ตแบบลำดับชั้น..............................16
2.4.3 คุณสมบัติการจัดลำดับความสำคัญของคลาสใน HTB....................................18
2.4.4 คุณสมบัติสนับสนุนการส่งข้อมูลแบบเบิร์ส์..18
2.4.5 การใช้งานชุดคำสั่งควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการ ลินุกซ์ด้วยเทคนิค HTB...19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

2.5 การควบคุมปริมาณข้อมูลด้วยเทคนิคการประกันบริการแบบลำดับชั้นด้วยเส้นโค้งประกันบริการ....25

2.5.1 แบบจำลองเส้นโค้งการรับประกันการให้บริการอย่างยุติธรรมของการเชื่อมต่อร่วมกัน (FSC Link-Sharing)....26

2.5.2 การให้บริการแบบลำดับชั้น....26

2.5.3 การรับประกันคุณภาพโดยใช้เส้นโค้งการรับประกันการให้บริการ....26

2.5.3.1 แนวคิดของเส้นโค้งการรับประกันการให้บริการ....26

2.5.3.2 เส้นโค้งการรับประกันการให้บริการและการรับประกันความสามารถ....27

2.5.3.3 เส้นโค้งการรับประกันการให้บริการและการจัดการบริการอย่างยุติธรรม....27

2.5.4 เทคนิคการรับประกันบริการแบบลำดับชั้นด้วยเส้นโค้งการรับประกันการให้บริการ (HFSC)....28

2.5.4.1 แนวคิดและหลักการทำงานพื้นฐานของ HFSC....29

2.5.5 การใช้งานชุดคำสั่งควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลีนุกซ์ด้วยเทคนิค HFSC....33

บทที่ 3 แนวคิดและหลักการจัดการปริมาณทราฟฟิกของเรียลไทม์แอพพลิเคชั่น....38

3.1 เปรียบเทียบประสิทธิภาพระหว่างอัลกอลิทึม HTB และ HFSC....38

3.3.1 จุดประสงค์การเปรียบเทียบประสิทธิภาพ....38

3.3.2 ออกแบบการทดลอง....38

3.3.3 ขั้นตอนการทดสอบเว็บแอพพลิเคชั่นตามกฎควบคุมปริมาณทราฟฟิก....39

3.3.4.1 ทดสอบความสามารถในการจัดการทราฟฟิกในสภาวะปกติ....39

3.3.4.2 ทดสอบความสามารถในการจัดการทราฟฟิกเฉพาะชนิดข้อมูลและเรียลไทม์....40

3.3.4.3 ทดสอบความสามารถในการจัดการทราฟฟิกในสภาวะเบิร์ส์ชนิดข้อมูล....41

3.3.4.4 ทดสอบความสามารถในการจัดการทราฟฟิกในสภาวะเบิร์ส์ชนิดเรียลไทม์....42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

3.3.4 ผลการทดสอบเปรียบเทียบประสิทธิภาพ
ระหว่างอัลกอลิทึม HTB และ HFSC....42
3.3.4.1 การจัดสรรแบนด์วิดท์....42
3.3.4.2 การจัดการค่าความหน่วง....45
3.3.5 สรุปผลการทดสอบเปรียบเทียบประสิทธิภาพ
ระหว่างอัลกอลิทึม HTB และ HFSC....47
3.2 แนวคิดการทำงานแบบสลับอัลกอลิทึมโดยอัตโนมัติ....47
3.2.1 หลักการสลับอัลกอลิทึมแบบอัตโนมัติ....48
3.3 หลักการจุดบ่งบอกการสลับอัลกอลิทึม (Threshold)....49
3.3.1 ออกแบบการทดลอง....49
3.3.2 การทดลองโดยการเปลี่ยนอัตราการส่งข้อมูลในเครือข่าย
ระหว่างโปรโตคอล UDP และ TCP....50
3.3.3 การทดลองโดยใช้เทคนิคการเปลี่ยนแปลงค่าแบนด์วิดท์ที่รับประกัน
และค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้ แทนอัตราการเพิ่มขึ้น
ของปริมาณข้อมูลในคลาสของทราฟฟิกเรียลไทม์....52
3.3.4 สรุปผลการทดลอง....58
บทที่ 4 การวิเคราะห์และการออกแบบระบบ....59
4.1 การทำงานระบบเดิม....59
4.2 ปัญหาระบบเดิม....60
4.3 การทำงานระบบใหม่....60
4.4 การทำงานโดยภาพรวมของระบบ....62
4.5 แผนภาพคลาส....64
4.6 แผนภาพอีอาร์....65
4.7 แผนภาพสถานะ....65
4.8 ไดอะแกรมแสดงขั้นตอนการทำงานหลักของแอพพลิเคชั่น....66
4.9 พจนานุกรมข้อมูล....75
4.10 การออกแบบด้านโครงสร้างการทำงาน....82
4.10.1 การออกแบบด้านฟังก์ชั่นการทำงาน....82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

4.10.2 การออกแบบด้านหน้าติดต่อผู้ใช้งาน..84
4.10.2.1 การล็อกอินเข้าใช้งานระบบ..84
4.10.2.2 การกำหนดค่าให้กับระบบ..85
1. ส่วนการจัดการชื่อบัญชีผู้ใช้งาน..85
2. ส่วนการกำหนดค่าให้กับเครื่อง..87
4.10.1.1 การกำหนดกฎนโยบายของการควบคุมปริมาณทราฟฟิก........88
1. ส่วนการสร้างกฎนโยบายการควบคุมปริมาณทราฟฟิก......88
2. ส่วนการแสดงและแก้ไขกฎนโยบาย
การควบคุมปริมาณทราฟฟิก..97
3. ส่วนการแสดงชุดคำสั่งกฎนโยบายของ
การควบคุมปริมาณทราฟฟิก..98
4.10.1.2 การรายงานสถิติการใช้งานทางเครือข่าย..99
1. รายงานสถิติการใช้บริการทางเครือข่ายตามคลาส............99
2. รายงานสถิติการใช้บริการทางเครือข่ายโดยภาพรวม.......100
4.10.1.3 แสดงบทคัดย่อของโครงงาน..101
บทที่ 5 การพัฒนาและทดสอบระบบ..102
5.1 เครื่องมือที่ใช้ในการพัฒนาระบบ..102
5.1.1 ฮาร์ดแวร์..102
5.1.2 ซอฟต์แวร์..102
5.2 การทดสอบการทำงานระบบ..104
5.2.1 จุดประสงค์ของการทดสอบการทำงานระบบ..104
5.2.2 เครื่องมือที่ใช้ในการทดสอบการทำงานระบบ..104
5.2.3 แผนภาพจำลองการทดสอบการทำงานระบบ..104
5.2.4 กฎนโยบายควบคุมปริมาณทราฟฟิกในการทดสอบ
การทำงานระบบ..105
5.2.5 ขั้นตอนการทดสอบเว็บแอพพลิเคชั่นตามกฎควบคุมปริมาณทราฟิก...107
5.2.5.1 การทดสอบการกรองและการควบคุมปริมาณทราฟฟิก
พร้อมกัน 2 อินเทอร์เฟส..107

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

5.2.5.2 การทดสอบการกรองและการควบคุมปริมาณทราฟฟิก ได้ด้วยอัลกอลิทึมแบบ HTB, HFSC และ Automatic Swapping Algorithm 107

5.2.6 การทดสอบการทำงานระบบ กรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน....110

5.2.7 การทดสอบเว็บแอพพลิเคชั่น กรณีควบคุมปริมาณทราฟฟิก 1 อินเทอร์เฟส....116

5.2.8 สรุปผลการทดสอบแอพพลิเคชั่น....120

บทที่ 6 บทสรุป....122

6.1 สรุปผลการพัฒนาระบบ....122

6.1.1 จุดเด่นของโครงงาน....123

6.1.2 จุดด้อยของโครงงาน....124

6.2 ประโยชน์ของระบบงานที่พัฒนาขึ้นใหม่....125

6.3 ข้อจำกัดของการพัฒนาระบบ....126

6.4 ปัญหาและอุปสรรคระหว่างการพัฒนาระบบ....126

6.5 ข้อเสนอแนะ....127

บรรณานุกรม....130

ภาคผนวก ก การติดตั้งระบบ....132

ภาคผนวก ข การใช้งานระบบ....140

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ หน้า

2.1 แสดงสถานะของคลาสและการยืมแบนด์วิดท์ของ HTB..24

3.1 รูปแบบการทดลองอัตราส่วนระหว่างทราฟฟิกแบบ TCP และ UDP 51

3.2 รูปแบบการทดลองการเปลี่ยนแปลงอัตราระหว่างค่าแบนด์วิดท์ที่รับประกัน
และค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้..53

3.3 ตารางแสดงการสูญหายของแพ็คเก็ตประเภทเรียลไทม์
เมื่อภายในเครือข่ายมีเพียงทราฟฟิกสื่อประสม...54

4.1 พจนานุกรมข้อมูลของระบบควบคุมปริมาณทราฟฟิก..75

4.2 ข้อมูลของการปรับแต่งทราฟฟิก..75

4.3 ข้อมูลคลาสตามกฎนโยบาย...76

4.4 ข้อมูลการกรองทราฟฟิกโดยใช้พอร์ต..77

4.5 ข้อมูลการกรองทราฟฟิกโดยใช้ไอพีแอดเดรส..77

4.6 ข้อมูลการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นขนส่งข้อมูล.................................78

4.7 ข้อมูลการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นแอพพลิเคชั่น...............................78

4.8 ข้อมูลกฎนโยบายบริการ...78

4.9 ข้อมูลสถิติการใช้งานทราฟฟิกขาเข้า 79

4.10 ข้อมูลสถิติการใช้งานทราฟฟิกขาออก 80

4.11 ข้อมูลผู้ใช้งาน...80

4.12 ข้อมูลการติดตั้งแอพพลิเคชั่น...81

5.1 แสดงปริมาณทราฟฟิกที่ใช้ในการทดสอบการกรองและ
การควบคุมปริมาณทราฟฟิกพร้อมกัน 2 อินเทอร์เฟส..107

5.2 ตารางแสดงอัตราการใช้งานแบนด์วิดท์ในการทดสอบแอพพลิเคชั่น..........................109

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ หน้า

2.1 แสดงโครงสร้างเครือข่ายของลีนุกซ์และการอ้างอิงคู่กับโมเดล OSI..........................6

2.2 แสดงโครงสร้างเครือข่ายที่เกี่ยวข้องกับการควบคุมทราฟฟิก..........................7

2.3 แสดงตัวอย่างของข้อกำหนดของคิวที่มีคลาสและการกรองแพ็คเก็ตอยู่ภายใน..............8

2.4 แสดงการทำงานของ Classless Qdisc ประเภท pfifo_fast..........................10

2.5 แสดงการทำงานของ Classless Qdisc ประเภท SFQ..........................10

2.6 แสดงการทำงานของ Classless Qdisc ประเภท TBF..........................11

2.7 แสดงแนวคิดของโทเค็น/บัคเก็ต..........................15

2.8 แสดงแนวคิดการทำงานของโทเค็น/บัคเก็ต..........................16

2.9 แสดงตัวอย่างการควบคุมปริมาณทราฟฟิกโดยเทคนิค HTB..........................17

2.10 แสดงกราฟอัตราความเร็วในการส่งข้อมูล ซึ่งมีอัตราแตกต่างกัน ในช่วงเวลา Burst Time..........................19

2.11 ชุดคำสั่งควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลีนุกซ์ด้วย HTB..........................19

2.12 แสดงตัวอย่างคำสั่งสร้าง Root Qdisc ของ HTB..........................20

2.13 แสดงตัวอย่างคำสั่งสร้าง Root Class ของ HTB..........................21

2.14 แสดงตัวอย่างคำสั่งสร้างตัวกรองทราฟฟิก (Filter) ที่คลาสราก..........................21

2.15 แสดงตัวอย่างคำสั่งสร้างคลาสภายในของ HTB แบบให้ยืมแบนด์วิดท์ ทั้งคลาสภายในได้..........................21

2.16 แสดงตัวอย่างคำสั่งสร้างคลาสภายในของ HTB แบบกำหนดไม่ให้ยืมแบนด์วิดท์ข้ามคลาสภายใน..........................22

2.17 แสดงตัวอย่างคำสั่งสร้างตัวกรองทราฟฟิกที่คลาสภายใน..........................22

2.18 แสดงตัวอย่างคำสั่งสร้างคลาสภายในของ HTB..........................22

2.19 แสดงตัวอย่างคำสั่งสร้างคลาสสุดท้ายให้แต่ละบริการ..........................23

2.20 แสดงตัวอย่างคำสั่งสร้างตัวกรองทราฟฟิก..........................23

2.21 แสดงตัวอย่างคำสั่งแสดงรายละเอียดข้อมูลข้อกำหนดของคิว คลาสและตัวกรองที่อินเทอร์เฟซ eth0..........................23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่ หน้า

2.22 กราฟแสดงเส้นโค้งการประกันการให้บริการที่บ่งบอกถึง
ผลรวมของแบนด์วิดท์ที่ได้มีการประกันไว้..........27

2.23 แสดงถึงตัวอย่างโครงสร้างลำดับชั้นของแต่ละคลาสใน HFSC..........29

2.24 แสดงถึงตัวอย่างโครงสร้าง HFSC และความสัมพันธ์ระหว่าง 3 ตัวแปร
คือ Eligible Time, Deadline และ Virtual Time..........30

2.25 แสดงถึงค่า Virturl Time ของคลาส..........31

2.26 แสดงถึงค่า Eligible Timeและ Deadline Time ของคลาส..........31

2.27 แสดงคำสั่งควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลีนุกซ์ด้วย HFSC..........33

2.28 แสดงตัวอย่างคำสั่งการสร้าง Root Qdisc ของ HFSC ที่ขาอินเทอร์เฟซ Eth2..........35

2.29 แสดงตัวอย่างคำสั่งการสร้างคลาสแม่..........35

2.30 แสดงตัวอย่างคำสั่งการสร้างคลาสสุดท้ายของ HFSC..........36

2.31 แสดงตัวอย่างคำสั่งการสร้างข้อกำหนดของคิวให้กับคลาสสุดท้ายของ HSFC..........37

2.32 แสดงตัวอย่างคำสั่งการสร้างตัวกรองแพ็กเก็ต เพื่อทำการกรองแพ็คเก็ต UDP
ที่มีพอร์ตเท่ากับ4444..........38

2.33 แสดงตัวอย่างคำสั่งดูข้อมูลการทำงานของข้อกำหนดคิว..........37

2.34 แสดงตัวอย่างคำสั่งดูข้อมูลการทำงานของคลาส..........37

3.1 กฎนโยบายควบคุมปริมาณทราฟฟิกในการทดสอบเปรียบเทียบประสิทธิภาพ
ระหว่างอัลกอลิทึม HTB และ HFSC..........38

3.2 แผนภาพจำลองเครือข่ายของการทดสอบเปรียบเทียบประสิทธิภาพ
ระหว่างอัลกอลิทึม HTB และ HFSC..........39

3.3 แสดงขั้นตอนทดสอบความสามารถในการจัดการทราฟฟิกในสภาวะปกติ..........39

3.4 แสดงขั้นตอนทดสอบความสามารถในการจัดการทราฟฟิกเฉพาะชนิดข้อมูล..........40

3.5 แสดงขั้นตอนทดสอบความสามารถในการจัดการทราฟฟิกเฉพาะชนิดเรียลไทม์..........40

3.6 แสดงขั้นตอนทดสอบความสามารถในการจัดการทราฟฟิกในสภาวะเบิร์ส
ชนิดข้อมูลและชนิดเรียลไทม์..........41

3.7 แสดงความสามารถในการจัดสรรแบนด์วิดท์ในการทดลองที่ 3.1.2.1..........42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่ หน้า

3.8 แสดงความสามารถในการจัดสรรแบนด์วิดท์ในสภาวะที่เครือข่าย มีเฉพาะทราฟฟิกชนิดข้อมูลเท่านั้น......43

3.9 แสดงความสามารถในการจัดสรรแบนด์วิดท์ในสภาวะที่เครือข่าย มีเฉพาะทราฟฟิกชนิด เรียลไทม์เท่านั้น......43

3.10 แสดงความสามารถในการจัดสรรแบนด์วิดท์ในสภาวะเบิรส์ชนิดข้อมูล......44

3.11 แสดงความสามารถในการจัดสรรแบนด์วิดท์ในสภาวะเบิรส์เฉพาะชนิดเรียลไทม์......44

3.12 แสดงความสามารถในการจัดการค่าความหน่วงในสภาวะเครือข่ายปกติ......45

3.13 แสดงความสามารถในการจัดการค่าความหน่วงในสภาวะเบิรส์ชนิดข้อมูล......45

3.14 แสดงความสามารถในการจัดการค่าความหน่วงในสภาวะเบิรส์ชนิดเรียลไทม์......46

3.15 แสดงแนวคิดหลักการสลับอัลกอลิทึมอัตโนมัติ (Automatic Swapping Algorithm).....48

3.16 แสดงแผนผังนโยบายการควบคุมปริมาณทราฟฟิกที่ใช้ในการทดลอง......50

3.17 แสดงกราฟเปอร์เซ็นต์การสูญหายของแพ็คเก็ตของอัลกอลิทึม HTB และ HFSC ในการทดลองในกรณีที่สภาวะของเครือข่ายมีเพียงทราฟฟิการรับและส่งไฟล์ มัลติมีเดียเท่านั้น......55

3.18 แสดงกราฟเปอร์เซ็นต์การสูญหายของแพ็คเก็ตของอัลกอลิทึม HTB และ HFSC ในการทดลองกรณีที่สภาวะแวดล้อมที่เครือข่ายมีทราฟฟิกประเภทอื่นด้วย......56

3.19 แสดงกราฟเปอร์เซ็นต์การสูญหายของแพ็คเก็ตของอัลกอลิทึม HTB และ HFSC ในการทดลองกรณีที่สภาวะที่ทุกคลาสมีทราฟฟิกอย่างหนาแน่นเข้าสู่คลาส......57

4.1 ตัวอย่างชุดคำสั่งแบบบรรทัดการใช้งานระบบควบคุมปริมาณทราฟฟิก......59

4.2 การใช้งานระบบควบคุมปริมาณทราฟฟิกแบบเดิม......60

4.3 การใช้งานระบบควบคุมปริมาณทราฟฟิกแบบใหม่......62

4.4 ระบบควบคุมปริมาณทราฟฟิก......63

4.5 การไหลของแพ็คเก็ตของระบบควบคุมปริมาณทราฟฟิก......63

4.6 แผนภาพคลาสของการพัฒนาระบบ......64

4.7 แผนภาพอีอาร์ของการพัฒนาระบบ......65

4.8 แผนภาพสถานะของการทำงานของระบบ......65

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่ หน้า

4.9 ไดอะแกรมแสดงขั้นตอนการทำงานของไฟล์ make_default.php......66

4.10 ไดอะแกรมแสดงขั้นตอนการทำงานของไฟล์ make_iptables.php......68

4.11 ไดอะแกรมแสดงขั้นตอนการทำงานของไฟล์ make_tc.php......69

4.12 ไดอะแกรมแสดงขั้นตอนการทำงานของการสร้างสคริปต์อัลกอลทึม HTB......71

4.13 ไดอะแกรมแสดงขั้นตอนการทำงานของการสร้างสคริปต์อัลกอลทึม HFSC......72

4.14 ไดอะแกรมแสดงขั้นตอนการทำงานของไฟล์ shaper_collector.php......73

4.15 โครงสร้างของเว็บแอพพลิเคชั่นโดยรวม......82

4.16 โครงสร้างของเว็บแอพพลิเคชั่นของการกำหนดค่าให้กับระบบ......83

4.17 โครงสร้างของเว็บแอพพลิเคชั่นของการจัดการชื่อบัญชีผู้ใช้งาน......83

4.18 โครงสร้างของเว็บแอพพลิเคชั่นของการกำหนดค่าให้กับเครื่อง......83

4.19 โครงสร้างของเว็บแอพพลิเคชั่นของการกำหนดนโยบายของการควบคุมปริมาณทราฟฟิก......84

4.20 โครงสร้างของเว็บแอพพลิเคชั่นของรายงานสถิติการใช้งานทางเครือข่าย......84

4.21 หน้าต่างล็อกอินเข้าใช้งานระบบ......85

4.22 หน้าต่างแสดงข้อมูลของบัญชีผู้ใช้งาน......85

4.23 หน้าต่างแสดงการแก้ไขข้อมูลบัญชีผู้ใช้งาน......86

4.24 หน้าต่างแสดงการสร้างบัญชีผู้ใช้งาน......86

4.25 หน้าต่างแสดงการลบบัญชีผู้ใช้งาน......87

4.26 หน้าต่างแสดงการดูค่าต่างๆ ที่ได้กำหนดไว้ในเครื่อง......87

4.27 หน้าต่างแสดงการแก้ไขค่าต่างๆให้กับเครื่อง......88

4.28 หน้าต่างแสดงการกำหนดค่าให้กับเครื่องเริ่มต้น......89

4.29 หน้าต่างแสดงการกำหนดค่าข้อมูลของการปรับแต่งทราฟฟิก......90

4.30 หน้าต่างแสดงการกำหนดค่าของคลาสระดับที่ 1......91

4.31 หน้าต่างแสดงการกำหนดค่าของคลาสระดับที่ 2......92

4.32 หน้าต่างแสดงการกำหนดค่าของคลาสระดับที่ 3......93

4.33 หน้าต่างแสดงข้อมูลคลาสสุดท้ายทั้งหมด......94

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่ หน้า

4.34 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้ไอพีแอดเดรส....95

4.35 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้พอร์ต....95

4.36 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นขนส่งข้อมูล....96

4.37 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นแอพพลิเคชั่น....96

4.38 หน้าต่างแสดงสรุปกฎนโยบายของการควบคุมปริมาณทราฟฟิก....97

4.39 หน้าต่างแสดงและแก้ไขกฎนโยบายของการควบคุมปริมาณทราฟฟิก....97

4.40 หน้าต่างแก้ไขข้อมูลคลาสที่กำหนดการกรองทราฟฟิก....98

4.41 หน้าต่างแสดงชุดคำสั่งกฎนโยบายของการควบคุมปริมาณทราฟฟิก....98

4.42 กราฟเส้นแสดงปริมาณไบต์ข้อมูลในเครือข่ายตามคลาส....99

4.43 กราฟวงกลมแสดงปริมาณไบต์ข้อมูลในเครือข่ายตามคลาส....99

4.44 กราฟเส้นแสดงปริมาณไบต์ข้อมูลในเครือข่ายโดยภายรวม....100

4.45 กราฟวงกลมแสดงปริมาณไบต์ข้อมูลในเครือข่ายโดยภายรวม....100

4.46 แสดงรายงานบทคัดย่อโครงงาน....101

5.1 แผนภาพจำลองเครือข่ายของการทดสอบการทำงานระบบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟส....104

5.2 แผนภาพจำลองเครือข่ายของการทดสอบการทำงานระบบกรณีควบคุมปริมาณทราฟฟิก 1 อินเทอร์เฟส....105

5.3 กฎนโยบายควบคุมปริมาณทราฟฟิกในการทดสอบการทำงานระบบ....105

5.4 ขั้นตอนการการทดสอบการทำงานระบบด้วยการใช้อัลกอลิทึมแบบ HTB....107

5.5 ขั้นตอนการการทดสอบการทำงานระบบด้วยการใช้อัลกอลิทึมแบบ HFSC....108

5.6 ขั้นตอนการการทดสอบการทำงานระบบด้วยการใช้อัลกอลิทึมแบบ Auto Swapping....108

5.7 แสดงกฎนโยบายควบคุมปริมาณทราฟฟิกกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน....110

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่ หน้า

5.8 กราฟเส้นแสดงปริมาณข้อมูลในเครือข่ายโดยภาพรวมที่ผ่านอินเทอร์เฟส eth0 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน....................112

5.9 กราฟเส้นแสดงปริมาณข้อมูลในเครือข่ายโดยภาพรวมที่ผ่านอินเทอร์เฟส eth2 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน....................112

5.10 กราฟวงกลมแสดงปริมาณข้อมูลในเครือข่ายโดยภาพรวมที่ผ่านอินเทอร์เฟส eth0 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน....................113

5.11 กราฟวงกลมแสดงปริมาณข้อมูลในเครือข่ายโดยภาพรวมที่ผ่านอินเทอร์เฟส eth2 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน....................113

5.12 กราฟเส้นแสดงปริมาณข้อมูลในเครือข่ายตามคลาสที่ผ่านอินเทอร์เฟส eth0 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน....................114

5.13 กราฟเส้นแสดงปริมาณข้อมูลในเครือข่ายตามคลาสที่ผ่านอินเทอร์เฟส eth2 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน....................114

5.14 กราฟวงกลมแสดงปริมาณข้อมูลในเครือข่ายตามคลาสที่ผ่านอินเทอร์เฟส eth0 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน....................115

5.15 กราฟวงกลมแสดงปริมาณข้อมูลในเครือข่ายตามคลาสที่ผ่านอินเทอร์เฟส eth2 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกันตาม..............115

5.16 แสดงกฎนโยบายควบคุมปริมาณทราฟฟิกที่กำหนดจากเว็บแอพพลิเคชั่น กรณีควบคุมปริมาณทราฟฟิก 1 อินเทอร์เฟส..117

5.17 กราฟแสดงผลการควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึม HFSC กรณีควบคุมปริมาณทราฟฟิก 1 อินเทอร์เฟส..117

5.18 กราฟแสดงผลการควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึม HTB กรณีควบคุมปริมาณทราฟฟิก 1 อินเทอร์เฟส..118

5.19 กราฟแสดงผลการควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึม Auto Swapping กรณีควบคุมปริมาณทราฟฟิก 1 อินเทอร์เฟส..118

5.20 กราฟแสดงการเปรียบเทียบแนวโน้มการดร็อปแพ็คเก็ต (Drop) จากการทดลองทั้ง 3 อัลกอลิทึม..119

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่ หน้า

5.21 กราฟแสดงการเปรียบเทียบแนวโน้มการส่งแพ็คเก็ตใหม่ (Retransmission) จากการทดลองทั้ง 3 อัลกอลิทึม....119

6.1 การส่งแพ็คเก็ตระหว่างเครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์หลัก (Active Linux Traffic Control) ให้กับเครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์สำรอง (Stand-By Linux Traffic Control)....127

6.2 การเกิดการล้มเหลวเกิดขึ้นแต่การเชื่อมต่อกับระบบยังไม่ขาด....128

ก.1 ระบบปฏิบัติการลินุกซ์อูบันตู5.10....132

ข.1 หน้าต่างล็อกอินเข้าสู่ระบบ....141

ข.2 หน้าต่างแสดงรายงานสถิติการใช้งานทางเครือข่าย....141

ข.3 หน้าต่างแสดงข้อมูลของบัญชีผู้ใช้งาน....142

ข.4 หน้าต่างแสดงการแก้ไขข้อมูลบัญชีผู้ใช้งาน....143

ข.5 หน้าต่างแสดงการสร้างบัญชีผู้ใช้งาน....143

ข.6 หน้าต่างแสดงการลบบัญชีผู้ใช้งาน....144

ข.7 หน้าต่างแสดงการดูค่าต่างๆที่ได้กำหนดไว้ในเครื่อง....145

ข.8 หน้าต่างแสดงการแก้ไขค่าต่างๆให้กับเครื่อง....145

ข.9 หน้าต่างแสดงข้อกำหนดในการสร้างกฎนโยบายควบคุมปริมาณทราฟฟิก....147

ข.10 หน้าต่างแสดงการกำหนดค่าให้กับเครื่องเริ่มต้น....147

ข.11 หน้าต่างแสดงการกำหนดค่าข้อมูลของการปรับแต่งทราฟฟิก....148

ข.12 หน้าต่างแสดงการกำหนดค่าของคลาสระดับที่ 1....149

ข.13 หน้าต่างแสดงการกำหนดค่าของคลาสระดับที่ 2....149

ข.14 หน้าต่างแสดงการกำหนดค่าของคลาสระดับที่ 3....150

ข.15 หน้าต่างแสดงข้อมูลคลาสสุดท้ายทั้งหมดในกฎนโยบายของการควบคุมปริมาณทราฟฟิกเพื่อกำหนดชนิดการกรองทราฟฟิก..150

ข.16 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้ไอพีแอดเดรส....151

ข.17 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้พอร์ต....152

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่ หน้า

ข.18 หน้าต่างแสดงการกำหนดการกรองทราฟฟิก
โดยใช้โปรโตคอลระดับชั้นขนส่งข้อมูล......152

ข.19 หน้าต่างแสดงการกำหนดการกรองทราฟฟิก
โดยใช้โปรโตคอลระดับชั้นแอพพลิเคชั่น......153

ข.20 หน้าต่างแสดงการสรุปกฎนโยบายของการควบคุมปริมาณทราฟฟิก......154

ข.21 หน้าต่างแสดงการยืนยันการสร้างกฎนโยบายควบคุมปริมาณทราฟฟิก......154

ข.22 หน้าต่างแสดงกฎนโยบายควบคุมปริมาณทราฟฟิก......155

ข.23 หน้าต่างแสดงการแก้ไขกฎนโยบายควบคุมปริมาณทราฟฟิก......156

ข.24 หน้าต่างแสดงการแก้ไขกฎนโยบายควบคุมปริมาณทราฟฟิก......156

ข.25 หน้าต่างแสดงรายงานสถิติการใช้งานทางเครือข่ายตามคลาส......157

ข.26 หน้าต่างแสดงรายงานสถิติการใช้งานเครือข่ายโดยภาพรวม......158

ข.27 หน้าต่างแสดงบทคัดย่อของโครงงาน......158

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากในปัจจุบันมีการใช้งานระบบเครือข่ายอินเตอร์เน็ตกันอย่างแพร่หลาย จึงเกิดการให้บริการหลายประเภทบนเครือข่าย โดยมีการเน้นไปที่การให้บริการที่เกี่ยวข้องกับสื่อประสม (Multimedia) มากขึ้น ทำให้แอพพลิเคชั่นประเภทที่เน้นการตอบสนองในทันที(Real-time Application) หรือประเภทที่ต้องการการรับประกันความต่อเนื่องในการรับส่งข้อมูล มีจำนวนการใช้งานเพิ่มขึ้นอย่างรวดเร็ว ทำให้ทรัพยากรที่มีอยู่อย่างจำกัด เช่น แบนด์วิดท์นั้นมีไม่เพียงพอต่อการใช้งานของโปรแกรมประยุกต์ต่างๆ ดังนั้นเพื่อให้เกิดประโยชน์สูงสุดในการใช้งาน จึงเกิดแนวความคิดที่ทำการควบคุมปริมาณการใช้งานแบนด์วิดท์ เพื่อให้การสื่อสารของโปรแกรมประยุกต์ประเภทต่างๆให้ได้รับบริการอย่างดีที่สุดเท่าเทียมกัน และ เกิดประโยชน์สูงสุด

เนื่องด้วยการใช้บริการระบบเครือข่ายในแต่ละประเภทนั้นมีความต้องการใช้แบนด์วิดท์ที่ไม่เท่ากัน เช่น การรับ-ส่งแฟ้มข้อมูลประเภทวีดิโอ นั้นจำเป็นที่จะต้องใช้แบนด์วิดท์อย่างต่อเนื่อง ไม่เช่นนั้นจะเกิดการทำงานที่ล้มเหลวขึ้น แต่ในการใช้บริการบางประเภทอาจไม่จำเป็นที่จะต้องใช้แบนด์วิดท์อย่างต่อเนื่อง เช่น การใช้บริการอีเมล์ (Electronic Mail) ดังนั้นหาก ณ เวลาหนึ่ง โปรแกรมประยุกต์หนึ่งโปรแกรมบนเครื่องคอมพิวเตอร์มีการใช้แบนด์วิดท์ในปริมาณที่มากที่สุดเท่าที่เครือข่ายจะสามารถให้บริการได้แล้ว อาจทำให้การขอใช้บริการที่มาหลังจากนั้นจะต้องทำการรอในคิว ถ้าหากมีการขอใช้บริการในแบบต้องการแบนด์วิดท์อย่างต่อเนื่องและแบบไม่ต้องการแบนด์วิดท์อย่างต่อเนื่องเกิดขึ้นพร้อมกัน การขอใช้บริการในแบบแรกจะได้อยู่ในลำดับที่ได้รับบริการก่อน เพื่อเป็นการป้องกันการล้มเหลวในการทำงาน เนื่องจากเป็นบริการที่ไม่สามารถจะรอได้นานและท้ายสุดแล้วบริการทั้งสองจะได้รับบริการอย่างสมบูรณ์ทั้งคู่ แต่หากให้บริการแบบแรกที่ไม่สามารถรอคอยได้ ถูกเลือกให้รอและให้บริการในภายหลังนั้น บริการดังกล่าวอาจเกิดความล้มเหลวได้ ทำให้คุณภาพในการให้บริการนั้นลดลง ดังนั้นปัญหาคือ จะมีวิธีการจัดการอย่างไร ให้บริการในแต่ละประเภทมีการใช้ทรัพยากรอย่างคุ้มค่าที่สุด ทรัพยากรที่มีอยู่ได้มีการกระจายไปให้บริการกับงานต่างได้อย่างทั่วถึงและยุติธรรม ดังนั้นเพื่อให้การจัดการทรัพยากรหรือแบนด์วิดท์ที่มีอยู่อย่างจำกัดให้สามารถถูกจัดสรรแก่งานในแต่ละประเภทได้อย่างเต็มที่ จึงควรมีการนำอุปกรณ์ หรือ โปรแกรมการทำงานที่สามารถบริหารจัดการและรับประกันคุณภาพแบนด์วิดท์มาใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในระบบปฏิบัติการลินุกซ์ มีการควบคุมทราฟฟิกบนระบบเครือข่าย โดยใช้การจัดลำดับการส่งแพ็คเก็ตที่เรียกว่า การควบคุมทราฟฟิก (Traffic Control) ซึ่งปัจจุบันได้กลายมาเป็นส่วนหนึ่งของมาตรฐานของเคอร์เนลในระบบปฏิบัติการลินุกซ์ โดย HTB (Hierarchical Token Bucket) และ HFSC (Hierarchical Fair Service Curve) เป็นเทคนิคการควบคุมปริมาณข้อมูลในเครือข่ายบนระบบปฏิบัติการลินุกซ์ ซึ่งผู้ดูแลระบบสามารถนำมาติดตั้งใช้งานได้ฟรี โดยเทคนิค HTB สามารถกรองแพ็คเก็ต จัดลำดับความสำคัญของแพ็คเก็ต จำกัดแบนด์วิดท์ได้ถึงระดับของผู้ใช้ หรือ กำหนดตามบริการได้สามารถจองแบนด์วิดท์ไว้สำหรับบางแอพพลิเคชั่น หรือ ให้ผู้ใช้บางคนได้ และ สามารถยืมแบนด์วิดท์จากคลาสอื่นมาใช้งานแบบชั่วคราวได้ ส่วนเทคนิค HFSC เป็นเทคนิคการควบคุมและจัดสรรปริมาณข้อมูลบนเครือข่ายเช่นเดียวกันกับ HTB แต่มีการประยุกต์โดยนำหลักการทำงานพื้นฐานของแบบจำลองของเส้นโค้งการให้บริการอย่างยุติธรรมของการใช้การเชื่อมต่อร่วมกัน (Fair Service Curve Link-Sharing Model) มาเป็นพื้นฐานในการทำงาน HFSC มีการทำงานบนพื้นฐานของเส้นโค้งการประกันการให้บริการ ซึ่งจะใช้ในการระบุถึงการรับประกันคุณภาพของทราฟฟิก หรือ บริการในช่วงเวลาหนึ่งๆ (Session) ในส่วนของแบนด์วิดท์และค่าความหน่วง โดยที่คลาสต่างๆจะมีการรับประกัน โดยอาศัยเส้นโค้งการประกันการให้บริการ

1.2 จุดมุ่งหมายและวัตถุประสงค์ของการศึกษา

จากการทดลองการเปรียบเทียบประสิทธิภาพระหว่างอัลกอลิทึม HTB และ HFSC แสดงให้เห็นถึงจุดเด่นของแต่ละอัลกอลิทึมในการจัดการควบคุมปริมาณทราฟฟิกในสถานการณ์ที่แตกต่างกัน ดังนั้นในโครงงานนี้จึงได้ดึงเอาจุดเด่นของทั้งสองอัลกอลิทึมมาปรับใช้ร่วมกันเพื่อใช้ในการจัดการปริมาณทราฟฟิกเรียลไทม์ในสถานการณ์ที่แตกต่างกัน โดยอาศัยการสลับอัลกอลิทึมโดยอัตโนมัติเพื่อเพิ่มประสิทธิภาพและปรับปรุงคุณภาพการให้บริการให้เหมาะสมกับสถานการณ์ต่างๆ และทำการวิเคราะห์หาข้อสรุปค่า Threshold ที่ใช้บ่งบอกถึงจุดที่การทำงานควรมีสลับอัลกอลิทึม แล้วนำมาพัฒนาเป็นแอพพลิเคชั่นที่ทำงานบนระบบปฏิบัติการลินุกซ์ เพื่อใช้ในการควบคุมปริมาณทราฟฟิกในเครือข่าย โดยมีเป้าหมาย คือ การจัดการปริมาณทราฟฟิกของเรียลไทม์ เพื่อให้ผู้ใช้สามารถที่จะทำการควบคุมปริมาณทราฟฟิกได้ง่าย สะดวกมากยิ่งขึ้น และสามารถใช้ทรัพยากรที่มีจำกัด เช่น แบนด์วิดท์ให้เกิดประสิทธิภาพสูงที่สุดรวมถึงช่วยในการรับประกันคุณภาพบริการต่างๆได้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงการ

ทำการสร้างแอพพลิเคชั่นที่ใช้ในการจัดการและควบคุมปริมาณทราฟฟิกในเครือข่ายให้เป็นไปตามกฎนโยบายที่ผู้ใช้กำหนด โดยพิจารณาถึงคุณภาพการให้บริการของทราฟฟิกเรียลไทม์ โดยนำเอาจุดเด่นระหว่างอัลกอลิทึม HTB และ HFSC มาปรับใช้งานร่วมกันในการจัดการเมื่อเครือข่ายมีสภาวะแวดล้อมที่แตกต่างกันผ่านการสลับอัลกอลิทึมโดยอัตโนมัติ โดยแอพพลิเคชั่นจะทำการวัดค่าปริมาณทราฟฟิกขาเข้านำมาพิจารณาเทียบกับค่า Threshold ถึงการปรับเปลี่ยนอัลกอลิทึมและเมื่อเกิดสถานการณ์ที่ตรงตามเงื่อนไขที่กำหนดจะมีการสลับให้อัลกอลิทึมอีกแบบที่มีความสามารถทำงานได้ดีกว่าในสภาพแวดล้อมขณะนั้นๆทำงานโดยอัตโนมัติ ซึ่งจะมีการแสดงผลการควบคุมปริมาณทราฟฟิกออกมาเป็นรูปแบบกราฟเส้นและกราฟวงกลมที่ชัดเจนและเข้าใจได้ง่าย

1.4 ขั้นตอนของการศึกษา

1. วิเคราะห์และออกแบบการทดลอง ทำการทดลอง และ สรุปผลการเปรียบเทียบประสิทธิภาพระหว่างอัลกอลิทึม HTB และ HFSC
2. พิจารณาหาปัจจัยที่จะนำมาเป็นตัววัดถึงการเปลี่ยนแปลงอัลกอลิทึม
3. วิเคราะห์และออกแบบการทดลอง ทำการทดลอง และ สรุปผลการทดลองหาค่า Threshold ที่ใช้บ่งบอกถึงจุดที่การทำงานต้องสลับอัลกอลิทึม
4. ออกแบบระบบในการพัฒนาแอพพลิเคชั่นและส่วนติดต่อกับผู้ใช้
5. สร้างแอพพลิเคชั่นตามระบบที่ได้ออกแบบไว้
6. ทำการทดลองเพื่อวัดประสิทธิภาพของแอพพลิเคชั่นที่สร้างขึ้น
7. วิเคราะห์ผลการทดลอง โดยทำการวิเคราะห์ถึงประสิทธิภาพการทำงานของแอพพลิเคชั่นที่ได้สร้างขึ้น
8. สรุปผลการวิเคราะห์การทดลอง เพื่อหาจุดบกพร่องและข้อแก้ไขต่อไป

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. เข้าใจวิธีการใช้งานและการกำหนดค่าต่างๆของระบบควบคุมปริมาณทราฟฟิกในเครือข่ายบนระบบปฏิบัติการลินุกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เข้าใจหลักการและการนำอัลกอลิทึม HTB และ HFSC ไปใช้ในการนำมาควบคุมปริมาณข้อมูลในเครือข่ายบนระบบปฏิบัติการลินุกซ์

3. สามารถวิเคราะห์ความสามารถของเทคนิคที่ใช้ควบคุมปริมาณทราฟฟิกภายในเครือข่ายบนระบบปฏิบัติการลินุกซ์และสามารถนำไปประยุกต์ใช้ในการควบคุมปริมาณทราฟฟิกระบบควบคุมปริมาณทราฟฟิกภายในเครือข่ายได้

4. สามารถนำความรู้จากการศึกษาในขั้นต้นมาสร้างเป็นแอพพลิเคชั่นที่สามารถใช้งานได้จริงได้

5. สร้างระบบการควบคุมทราฟฟิกผ่านทางเว็บไซต์โดยสามารถเข้าใจและใช้งานได้ง่าย นั่นคือผู้ใช้ที่ไม่มีความรู้พื้นฐานเกี่ยวกับกลไกการควบคุมปริมาณทราฟฟิกและการใช้งานชุดคำสั่ง Tc ก็สามารถเข้าใจและใช้งานได้ง่าย

6. การแสดงผลกฎนโยบายควบคุมปริมาณทราฟฟิกและแสดงข้อมูลการใช้งานแบนด์วิดท์ผ่านกราฟทำให้สามารถเข้าใจได้ง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีพื้นฐานที่ใช้ในการวิจัยและระบบควบคุมปริมาณทราฟฟิก

2.1 สาเหตุและแนวคิดการควบคุมปริมาณทราฟฟิก

โดยทั่วไปโครงสร้างของทีซีพี/ไอพี (TCP/IP) ในระบบปฏิบัติการลินุกซ์นั้น พยายามแบ่งแบนด์วิดท์ให้ทุกๆการติดต่ออย่างเท่าเทียมกัน ซึ่งทำให้เกิดความยุติธรรมในการใช้ทรัพยากร แต่ถ้าในเหตุการณ์ที่มี 100 การเชื่อมต่อ ใช้ทรัพยากรเครือข่าย(ในที่นี้คือ แบนด์วิดท์) พร้อมกันแล้ว แต่ละการเชื่อมต่อจะได้ใช้แบนด์วิดท์เพียงแค่ 1 เปอร์เซ็นต์เท่านั้น ดังนั้นอาจจะมีผลต่อทราฟฟิกบางประเภทได้ นอกจากนี้ในช่วงหลายปีที่ผ่านมาผู้ให้บริการอินเตอร์เน็ต (ISP) มีการควบคุมปริมาณทราฟฟิกของโปรแกรมประเภทบิตทอร์เรนต์ (BitTorrent) และระบบการแชร์ไฟล์อื่นๆที่อิงกับการสื่อสารของโปรแกรมประเภทเพียร์ทูเพียร์ (P2P) เนื่องจากทราฟฟิกเหล่านี้มีการใช้แบนด์วิดท์มากกว่าที่คาดการณ์ไว้ ทำให้ผู้ให้บริการต้องทำการควบคุมเพื่อลดการเกิดความแปรปรวนของค่าความหน่วง (Delay Jitter) และลดการสูญหายของแพ็คเก็ต (Packet Loss) ในเครือข่าย

การควบคุมปริมาณทราฟฟิกเป็นกลไกที่ควบคุมทราฟฟิกที่ถูกส่งในเครือข่ายและทำการจัดลำดับทราฟฟิกที่ถูกส่งออก เพื่อที่ให้เครือข่ายทำงานได้มีประสิทธิภาพดีที่สุด ซึ่งในการควบคุมปริมาณทราฟฟิกนั้นจะต้องทำการคัดแยกประเภทของทราฟฟิก จัดระบบแถวคอย กำหนดโยบายคุณภาพของการให้บริการ และ ความเป็นธรรมในการจัดการทราฟฟิก โดยยอมให้ผู้ดูแลระบบสามารถเข้ามาจัดการ หรือ ปรับปรุงการทำงานของระบบแถวคอยของแพ็คเก็ตได้ตามความเหมาะสม เช่น อาจทำให้ทราฟฟิกแต่ละประเภทนั้นมีระดับความสำคัญที่ต่างกัน ซึ่งหากแบนด์วิดท์มีปริมาณไม่เพียงพอ การเชื่อมต่อที่มีความสำคัญมากกว่าจะมีสิทธิ์ได้ใช้ก่อน โดยจุดที่ควรทำการควบคุมปริมาณแบนด์วิดท์นั้นควรเป็นจุดที่มีการเกิดคอขวดเกิดขึ้น เช่น ที่ไฟล์วอลล์ เราเตอร์ หรือ จุดที่มีการแย่งกันใช้แบนด์วิดท์ โดยผลที่จะได้จากการควบคุมปริมาณทราฟฟิก มีดังนี้

- ลดการเกิดความแปรปรวนของค่าความหน่วง
- ลดการสูญหายของแพ็คเก็ต
- ลดเวลาค่าความหน่วงในการส่งข้อมูล

ในกระบวนการควบคุมปริมาณทราฟฟิกนั้นเป็นการทำงานที่รวมถึงการตัดสินใจว่าที่อินเทอร์เฟสขาเข้าจะมีการยอมรับแพ็คเก็ตของทราฟฟิกแบบใดบ้าง ด้วยอัตราเท่าไร และ พิจารณาว่าที่อินเทอร์เฟสขาออก มีการยอมให้แพ็คเก็ตของทราฟฟิกแบบใดที่ส่งออกไปบ้าง ด้วยลำดับอย่างไร และด้วยอัตรา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เท่าไร ดังนั้น จึงมักจะมีการนำการควบคุมปริมาณทราฟฟิกไปทำงานร่วมกับระบบการควบคุมคุณภาพการให้บริการ

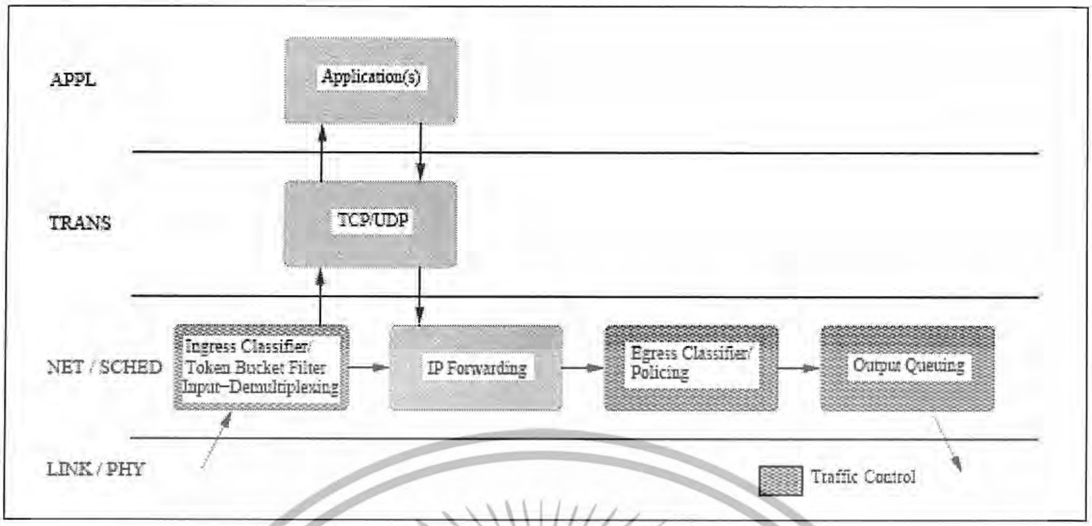
2.2 การควบคุมปริมาณข้อมูล

ในระบบปฏิบัติการลินุกซ์ มีการใช้แบนด์วิดท์ร่วมกันและมีการจัดลำดับการส่งแพ็คเก็ตที่เรียกว่า การควบคุมทราฟฟิก (TC:Traffic control) ซึ่งมาพร้อมกับระบบปฏิบัติการลินุกซ์ ตั้งแต่เคอร์เนล 2.1.90 และตั้งแต่เคอร์เนล 2.3 การควบคุมทราฟฟิกได้กลายเป็นส่วนหนึ่งของมาตรฐานของเคอร์เนล แม้ว่าการควบคุมทราฟฟิกจะสนใจที่ขนาดของทราฟฟิกในการส่งออก แต่โดยพื้นฐานแล้วลินุกซ์สนับสนุนการทำงานนโยบายของทราฟฟิกขาเข้าด้วย

ภาพที่ 2.1 แสดงโครงสร้างเครือข่ายของลินุกซ์และการอ้างอิงคู่กับ โมเดล OSI

จากภาพที่ 2.1 ลำดับชั้นดาต้าลิงก์ (Data link layer), ลำดับชั้นเน็ตเวิร์ก (Network layer) และ ลำดับชั้นทรานสปอร์ต (Transport layer) เป็นส่วนของการทำงานที่อยู่ในส่วนของเคอร์เนล แต่โปรโตคอลในระดับแอพพลิเคชั่น (Application layer) และแอพพลิเคชั่นต่างๆจะทำงานอยู่ในส่วนของผู้ใช้ ซึ่งการจัดลำดับการส่งแพ็คเก็ตและการควบคุมทราฟฟิกถูกจัดการให้ทำงานอยู่ใต้ระดับลำดับชั้นเน็ตเวิร์ก แต่จะอยู่เหนือไดร์เวอร์ของอุปกรณ์ต่างๆ โดยโปรโตคอลต่างๆในลำดับชั้นเน็ตเวิร์ก เช่น IP, IPX และ Appletalk สามารถทำการจัดลำดับแพ็คเก็ตใหม่ก่อนที่จะทำการส่งแพ็คเก็ตเหล่านั้นลงไปให้กับเน็ตเวิร์กการ์ด (Network interface card) ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.2 แสดงโครงสร้างเครือข่ายที่เกี่ยวข้องกับการควบคุมทราฟฟิก

ในภาพที่ 2.2 แสดงถึงเส้นทางการส่งผ่านแพ็คเก็ตภายในโครงสร้างเครือข่ายของลินุกซ์ เมื่อแพ็คเก็ตมาถึงเน็ตเวิร์กการ์ดและถูกส่งต่อไปยังลำดับชั้นเน็ตเวิร์กจะทำการแยกจัดกลุ่มและทำการตรวจสอบกับนโยบาย ซึ่งอาจมีการลบแพ็คเก็ตทิ้ง หากได้รับแพ็คเก็ตมากเกินกว่าอัตราที่กำหนดไว้ โดยถือเป็นการควบคุมปริมาณข้อมูลของการควบคุมทราฟฟิกด่านแรกที่ขาเข้า จากนั้นจะมีการตรวจสอบว่าเป็นแพ็คเก็ตที่ปลายทางเป็นของเครื่องตัวเองหรือไม่ โดยจะทำการส่งแพ็คเก็ตให้กับ Input Demultiplexing ซึ่งหากเป็นแพ็คเก็ตที่ส่งมาให้ตัวลินุกซ์เกตเวย์เอง แพ็คเก็ตก็จะถูกส่งต่อไปให้ยังโปรโตคอลบนลำดับชั้นที่สูงกว่า เช่น TCP, UDP, ICMP เพื่อทำงานต่อไป แต่หากไม่ใช่ หรือ เมื่อโปรโตคอลในลำดับชั้นด้านบนทำงานเสร็จแล้ว ต้องการส่งแพ็คเก็ตตอบกลับก็จะส่งไปให้ในส่วนของฟอร์เวิร์ดดิ้งเพื่อที่จะตัดสินใจว่าจะต้องส่งแพ็คเก็ตนั้นๆ ต่อให้กับโหนดใดในเครือข่ายและเมื่อแพ็คเก็ตมาถึงที่ขาออกจะถูกจัดกลุ่ม แล้วจึงจัดลำดับในการส่งออก ซึ่งในการจัดลำดับอาจมีการลบแพ็คเก็ตทิ้ง มีการหน่วงการส่ง หรือ อาจถูกส่งออกทันทีเลยก็ได้ ถือเป็นการควบคุมปริมาณทราฟฟิกข้อมูลที่ฝั่งขาออก

2.3 ส่วนประกอบของการควบคุมทราฟฟิกบนระบบปฏิบัติการลินุกซ์

ส่วนประกอบพื้นฐานของโครงสร้างการควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์ ได้แก่ ข้อกำหนดของแถวคอย (Queuing Disciplines), คลาส (Classes) และ การกรองแพ็คเก็ต (Filter) ซึ่งในการจัดลำดับแพ็คเก็ตนั้นจะใช้ส่วนประกอบ 3 อย่างนี้ เพื่อให้ได้พฤติกรรมการทำงานตามที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

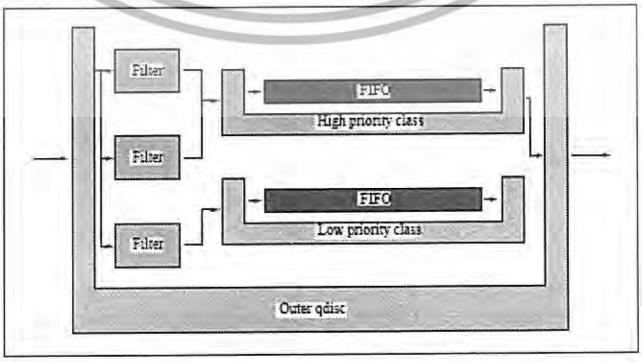
2.3.1 ข้อกำหนดของแถวคอย (Queuing Disciplines)

ข้อกำหนดของแถวคอย หรือ Qdiscs เป็นส่วนที่บริหารจัดการแพ็คเก็ตที่เข้ามาในแถวคอยและต้องส่งออกจากแถวคอย โดยจะมีการจัดลำดับในการเก็บแพ็คเก็ตตามอัลกอลิทึม หรือ เทคนิคที่เราเลือกใช้ ปกติแล้วการนำแพ็คเก็ตเข้าแถวคอยนั้นจะทำงานเมื่อระบบปฏิบัติการต้องการที่จะส่งต่อแพ็คเก็ตและการดึงแพ็คเก็ตออกจากแถวคอยนั้นจะเกิดขึ้นได้ก็ต่อเมื่ออุปกรณ์นั้นสามารถที่จะส่งแพ็คเก็ตต่อไปได้ ตัวอย่างเช่น แถวคอยที่มีการจัดการแบบลำดับก่อน-หลัง (First In First Out) ที่มีความสำคัญเท่าเทียมกันหมดจะมีการรับแพ็คเก็ตและการดึงแพ็คเก็ตออกจากแถวคอยด้วยลำดับเดียวกัน โดยข้อกำหนดแถวคอยสามารถที่จะแบ่งได้เป็น 2 อย่าง คือ

- Classfull Qdiscs คือ ข้อกำหนดของแถวคอยที่มีลักษณะเป็นคอนเทนเนอร์ คือ มีทั้งคลาสและตัวกรองแพ็คเก็ตอยู่ภายใน ซึ่งตัวกรองแพ็คเก็ตเป็นโมดูลที่ช่วยในการจัดลำดับและแยกแพ็คเก็ตเข้าสู่คลาสต่างๆ ส่วนคลาสจะเป็นการรวมกลุ่มของแพ็คเก็ตที่มีคุณสมบัติใกล้เคียงกัน
- Classless Qdiscs คือ ข้อกำหนดของแถวคอยที่ไม่มีคลาสลูก แต่ยังสามารถทำการกรองแพ็คเก็ตได้

ตัวอย่างของแถวคอยที่มีความสำคัญหลายระดับ เมื่อแพ็คเก็ตเข้ามายังข้อกำหนดของแถวคอยก็จะถูกส่งไปยังกลุ่มของตัวกรองแพ็คเก็ต ซึ่งก็จะทำการแยกแล้วส่งแพ็คเก็ตให้คลาสต่างๆต่อไป โดยเราจะสามารถกำหนดระดับความสำคัญให้แก่แพ็คเก็ตได้ โดยแต่ละคลาสควรมีการเก็บแพ็คเก็ตเข้าแถวคอยแบบลำดับก่อน-หลังที่มีความสำคัญเพียงระดับเดียว

หรืออาจมีการแบ่งข้อกำหนดของแถวคอยตามตามทิศทางของทราฟฟิกก็ได้โดยอาจกำหนดเป็นทราฟฟิกขาออกจากอินเทอร์เฟส (Outbound Traffic หรือ Egress Traffic) และทราฟฟิกขาเข้ามายังอินเทอร์เฟส (Inbound Traffic หรือ Ingress Traffic) โดยในแต่ละอินเทอร์เฟสของการ์ดแลนจะมีทั้งข้อกำหนดของแถวคอยขาออก (Egress Qdisc) และ ข้อกำหนดของแถวคอยขาเข้า (Ingress Qdisc)



ภาพที่ 2.3 แสดงตัวอย่างของข้อกำหนดของแถวคอยที่มีคลาสและการกรองแพ็คเก็ตอยู่ภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2 คลาส(Classes)

คลาสเป็นส่วนที่รวบรวมกฎเพื่อใช้ในการจัดการกับแพ็คเก็ต เช่น การกำหนดขนาดแบนด์วิดท์สูงสุดและต่ำสุด โดยปกติแล้วคลาสจะถูกควบคุมการทำงานโดยข้อกำหนดของแถวคอยโดยจะต้องมีหนึ่งคลาสที่เป็นคลาสราก (Root Class) เสมอ ซึ่งคลาสสามารถมีคลาสลูก (Child Class) ได้หลายคลาสหรือมีคลาสลูกเพียงคลาสเดียวก็ได้ ในแต่ละคลาสจะมีตัวกรองแพ็คเก็ตเพื่อทำการกรองแพ็คเก็ตเข้ามาทำงานหรือใช้เพื่อแยกแพ็คเก็ต รวมถึงตัดแพ็คเก็ตที่วิ่งเข้ามายังคลาสทิ้งได้ ซึ่งในแต่ละคลาสจะมีหมายเลขคลาส (Class Id) ที่กำหนดขึ้นโดยผู้ใช้ ซึ่งจะประกอบด้วย 2 ส่วน ได้แก่

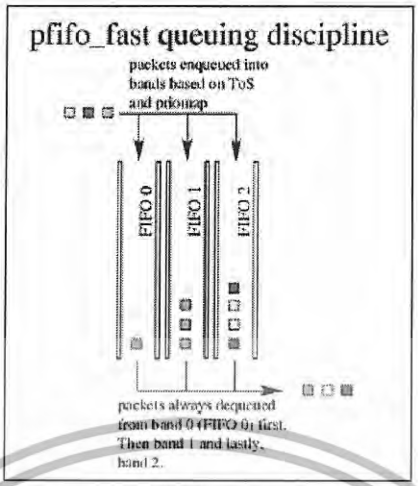
- หมายเลขหลัก ใช้เป็นตัวบ่งชี้ว่ามีคลาสแม่ (Parent) เป็นคลาสใด ดังนั้นทุกคลาสที่แตกออกจากคลาสเดียวกันจะต้องมีหมายเลขหลักเหมือนกันและหากมีหมายเลขหลักเป็น 1 แสดงว่ามีคลาสแม่เป็นข้อกำหนดของแถวคอยที่เป็นรากและตัวเลขระบุตัวตน (Handle) ffff:0 จะถูกสงวนไว้สำหรับข้อกำหนดของแถวคอยขาเข้า
- หมายเลขรอง หากมีค่าเป็น 0 จะเป็นการระบุว่าเป็นข้อกำหนดของแถวคอย แต่หากมีค่าเป็นอย่างอื่นจะหมายถึงคลาสในแต่ละคลาสที่มีคลาสแม่เดียวกันก็จะมีหมายเลขรองที่ไม่ซ้ำกัน

โดยเมื่อคลาสถูกสร้างขึ้นมาจะมีการกำหนดข้อบังคับของแถวคอยให้เป็นลำดับแถวคอยแบบเข้าก่อนออกก่อนเพื่อใช้ในการเก็บแพ็คเก็ต แต่เราสามารถเปลี่ยนแปลงได้ ส่วนใหญ่ในการเซ็ตค่าให้กับคลาสนั้น ทำได้สองอย่าง คือ ปรับเปลี่ยนคุณสมบัติต่างๆของคลาสและการผูก/ปลดตัวกรองแพ็คเก็ตหรือข้อบังคับของแถวคอยออก โดย Leaf Class คือ คลาสสุดท้ายของข้อกำหนดของแถวคอยซึ่งจะเป็นข้อกำหนดของแถวคอยประเภท Classless Qdisc ซึ่งโดยปกติของระบบปฏิบัติการลินุกซ์ คลาสสุดท้ายจะมีกลไกการทำงานที่จะส่งแพ็คเก็ตออกไปทันทีที่ได้รับมา (Packet FIFO) และจะไม่มีคลาสลูกอีก

ตัวอย่างประเภทของ Classless Qdisc ได้แก่

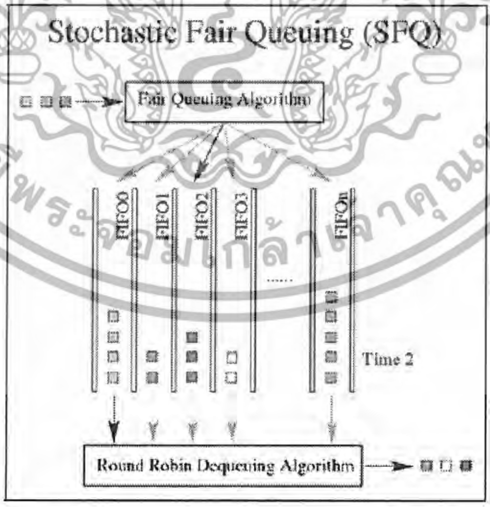
- pfifo_fast เป็นแถวคอยแบบ FIFO ที่มีความสำคัญอยู่ 3 ระดับ คือ 0 ,1 และ 2 โดยระดับความสำคัญสูงสุดคือ 0 โดยแพ็คเก็ตที่มีลำดับความสำคัญสูงจะได้เข้ารับบริการก่อน ซึ่งในแต่ละระดับความสำคัญก็จะมีการจัดการแบบ FIFO แถวคอยในลักษณะนี้เป็นแถวคอยที่ถูกเลือกให้แบบอัตโนมัติและสามารถที่จะสร้างคลาสใหม่ๆเพิ่มเติมเข้าไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.4 แสดงการทำงานของ Classless Qdisc ประเภท pfifo_fast

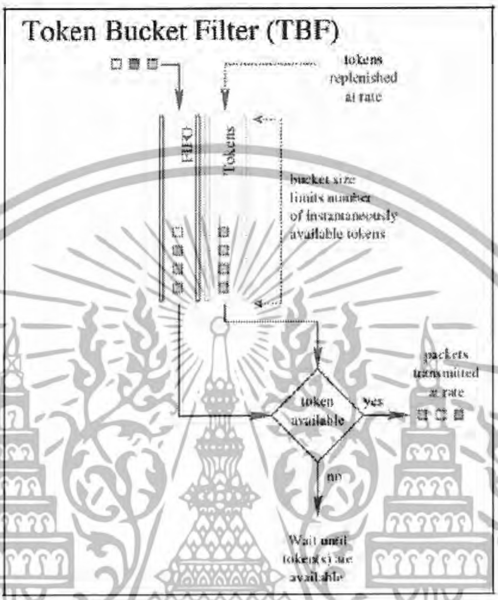
- SFQ (Stochastic Fair Queuing) เป็นการกำหนดแพ็คเก็ตขาเข้า เข้าไปในแถวคอยย่อยๆ ด้วยฟังก์ชั่นแฮช (Hashing Function) ซึ่งอนุญาตให้ผู้ดูแลระบบสามารถกำหนดเวลาในการเปลี่ยนอัลกอลิทึมการแฮชได้ เพื่อป้องกันการไหลของข้อมูลไปยังแถวคอยย่อยหนึ่งๆ มากเกินไปและส่งต่อแพ็คเก็ตออกด้วยวิธีวนรอบ (Round Robin)



ภาพที่ 2.5 แสดงการทำงานของ Classless Qdisc ประเภท SFQ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- TBF (Token Bucket Filter) ใช้วิธีแนวคิดของโทเค็น/บัคเก็ต โดยแพ็คเก็ตที่จะถูกส่งออกจากแถวคอยแบบ FIFO ได้จะต้องจับคู่กับโทเค็นที่เหลืออยู่ มิฉะนั้นแพ็คเก็ตดังกล่าวจะถูกหน่วงเวลา การทำงานลักษณะนี้เป็นการกำหนดอัตราความเร็วของการส่งต่อแพ็คเก็ตออกไปด้วยอัตราความเร็วตามที่ได้กำหนดไว้



ภาพที่ 2.6 แสดงการทำงานของ Classless Qdisc ประเภท TBF

2.3.3 ตัวกรองแพ็คเก็ต (Filter)

ข้อกำหนดของแถวคอยและคลาสใช้ตัวกรองแพ็คเก็ตเพื่อช่วยในการแยกและจัดกลุ่มแพ็คเก็ตที่เข้ามาในแต่ละคลาส ซึ่งในแต่ละคลาสจะมีตัวกรองแพ็คเก็ตมาทำงานร่วมกันกี่ตัวก็ได้ ไม่ว่าจะเป็น Classfull Qdisc หรือ Classless Qdisc ก็ได้ โดยปกติเวลาแพ็คเก็ตวิ่งเข้ามาในแถวคอยจะเข้ามาทาง Root Qdisc ก่อนเป็นลำดับแรก แล้วหลังจากนั้นจะพยายามจับคู่ระหว่างแพ็คเก็ตข้อมูลเข้ากับตัวกรองแพ็คเก็ตที่ได้กำหนดไว้เพื่อจะได้แยกแพ็คเก็ตเข้าสู่คลาสต่างๆ กระบวนการจับคู่นี้จะเริ่มพิจารณาจากตัวกรองแพ็คเก็ตที่มีระดับความสำคัญสูงสุดก่อน โดยตัวกรองแพ็คเก็ตจะมีการทำงาน 2 ส่วน คือ กลไกการแยกแยะแพ็คเก็ตข้อมูล (Classifier) และกลไกการจำกัดปริมาณการไหลของทราฟฟิก (Policer)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการกรองแพ็คเก็ตนั้นจะทำการตรวจสอบในส่วนของหัวของแพ็คเก็ต เพื่อที่จะนำมาพิจารณาว่าจะให้ผ่านไม่ให้ผ่าน หรือ อาจจะพิจารณาบางอย่างที่ซับซ้อนกว่านี้ก็ได้ ซึ่งจริงๆ แล้วการทำการกรองแพ็คเก็ตนั้น สามารถช่วยได้ในหลายๆแง่ อย่างเช่น

- ช่วยในเรื่องของการควบคุม คือ สามารถที่จะควบคุมได้ว่าจะให้แพ็คเก็ตที่มีรูปแบบใดบ้างที่จะสามารถผ่านเข้ามายังเครือข่ายของเรา เช่น ไม่ต้องการให้แพ็คเก็ตที่เป็นโปรโตคอล ICMP ผ่าน ไม่ต้องการให้แพ็คเก็ตที่มาจากไอพีแอดเดรสใดผ่านก็สามารถที่จะกำหนดได้

- ช่วยในเรื่องของความปลอดภัย เช่น หากต้องการที่จะระวังเรื่องของภัยคุกคามที่เรียกว่า 'Ping of Death' ก็สามารถที่จะทำการป้องกันได้ว่าอัตราการ Ping เท่าไรที่สามารถรับได้ ซึ่งหากมากกว่านั้นเมื่อไรให้ทำการละทิ้ง

- ช่วยในเรื่องของการเฝ้าระวัง เช่น สามารถช่วยตรวจสอบได้ว่า หากมีเครื่องใดภายในเครือข่ายของเราทำการส่งแพ็คเก็ตแปลกๆออกไป อาจมีการเซ็ตให้ตัวกรองแพ็คเก็ตแจ้งได้ หากเกิดเหตุการณ์ที่ไม่ปกติขึ้น เป็นต้น

ตัวอย่างของตัวกรองแพ็คเก็ตได้แก่

2.3.3.1 ไอพีเทเบิ้ล (IPTABLES)

IPTABLES เป็นชุดคำสั่งที่ทำงานในระบบปฏิบัติการลินุกซ์ ที่มีรูปแบบคำสั่งในการทำงานเป็นดังนี้

iptables [table] <command> <match> <target/jump> (2.1)

โดยกฎที่กำหนดขึ้นนั้นจะเป็นเป็นตัวบอกเคอร์เนลว่าให้กระทำอย่างไร ในกรณีที่พบแพ็คเก็ตตรงตามที่ระบุไว้

- **[table]** หมายถึง ตารางหรือ Table ที่ต้องการระบุ เช่น iptables -t nat หมายถึงให้ทำงานกับ NAT Table ซึ่ง IPTABLES มีการทำงานร่วมกับตารางหลัก 3 ตาราง ได้แก่ Filter, NAT, Mangle แต่หากไม่ได้ระบุตาราง IPTABLES จะถือว่าเป็นการเรียกใช้ตาราง Filter โดยอัตโนมัติ

- **<command>** จะเป็นตัวสั่งให้ IPTABLES ทำในสิ่งที่ต้องการ เช่น iptables -A INPUT ซึ่งหมายถึงให้สร้างกฎต่อท้าย INPUT Chain ในตาราง Filter

- **<match>** เป็นส่วนที่ใช้ตรวจสอบว่าแพ็คเก็ตมีข้อมูลตรงกับกฎที่ระบุไว้หรือไม่ โดยส่วนที่สามารถนำมาใช้พิจารณาได้นั้นมีอยู่หลายส่วน เช่น ไอพีแอดเดรส โปรโตคอล และ พอร์ต เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **<target/jump>** เป็นตัวระบุว่าเมื่อเจอแพ็คเก็ตที่มีข้อมูลตรงกับที่ระบุไว้แล้วก็จะกระทำตามที่ระบุไว้ เช่น ถ้าแพ็คเก็ตใดมีไอพีแอดเดรสต้นทางตรงกับ 1.2.3.4 ให้ทำการดร็อปแพ็คเก็ตนั้นทิ้งไป

โดยในการตรวจสอบกฎเมื่อทำการกรองแพ็คเก็ตนั้นจะทำการตรวจสอบเรียงลำดับตามกฎที่ถูกกำหนดไว้ ดังนั้นหากเราทำการสลับลำดับของกฎนั้นๆอาจได้ผลที่ต่างกัน เนื่องจากแพ็คเก็ตหนึ่งๆสามารถที่จะตรงกับกฎหลายๆข้อได้

2.3.3.2 เน็ตฟิลเตอร์ (NetFilter)

NetFilter เป็นกลุ่มของเครื่องมือที่อยู่ในเคอร์เนลของลินุกซ์ที่ช่วยในการดักจับ ช่วยทำการกรอง เปลี่ยนแปลงข้อมูลในส่วนหัวของแพ็คเก็ต (Mangle) และทำการแปลงรูปแพ็คเก็ต (Transform) ซึ่งสามารถทำงานร่วมกับ IPTABLES ได้ โดยสิ่งที่ NetFilter สามารถทำได้ ได้แก่

- เลือกแพ็คเก็ตจากพารามิเตอร์ตามที่เราเซ็ต เช่น ไอพีแอดเดรสของต้นทางและปลายทาง สถานะของแพ็คเก็ต
- ทำการ DROP, ACCEPT และ Reject แพ็คเก็ต
- ทำการเปลี่ยนแปลงค่าใน flag ของแพ็คเก็ต (Mangle)
- ทำการเปลี่ยนแปลงค่าไอพีแอดเดรส (NAT)
- ทำการคัดแยกและจัดกลุ่มแพ็คเก็ต

โครงสร้างคำสั่งที่ใช้ คือ

```
tc filter [ add | change | replace ] dev DEV [ parent qdiscid | root ]        (2.2)
      protocol protocol prio priority filter
      type [ filtertype specific parameters ] flowid flowid
```

2.3.3.3 เลเยอร์7-เน็ตฟิวเตอร์ (L7-Filter NetFilter)

L7-Filter NetFilter เป็นส่วนของการทำงานที่มีความสามารถที่เพิ่มเติมจาก Netfilter ซึ่งสิ่งที่น่าสนใจของ L7-Filter NetFilter ได้แก่ สามารถทำการบล็อกได้ตามโปรโตคอล ควบคุมการใช้แบนด์วิดท์ และ สามารถทำบันทึกการใช้งานที่สามารถกรองแพ็คเก็ตได้ถึงระดับชั้นแอพพลิเคชั่น โดยทำการพิจารณาที่เนื้อหาในแพ็คเก็ตว่าตรงกับรูปแบบที่กำหนดไว้หรือไม่ วิธีนี้เหมาะสำหรับการตรวจจับโปรแกรมประเภทเพียร์ทูเพียร์ (P2P) ที่มีการสุ่มหมายเลขพอร์ตทำงานไปเรื่อยๆ ทำให้ผู้ดูแลระบบตรวจจับแพ็คเก็ตเหล่านี้ได้ยาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างคำสั่งที่ใช้

iptables [specify table & chain] -m layer7 --l7proto [name of protocol] -j [action] (2.3)

2.4 การควบคุมปริมาณข้อมูลด้วยเทคนิคโทเค็น/บัคเก็ตแบบลำดับชั้น (Hierarchical Token Bucket : HTB)

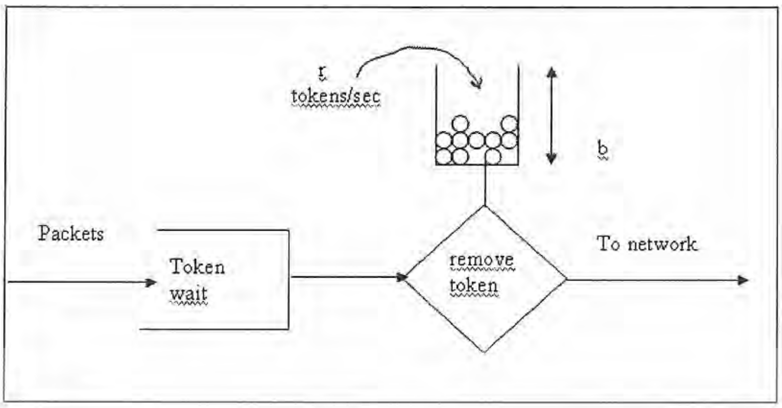
การควบคุมปริมาณข้อมูลด้วยเทคนิคโทเค็น/บัคเก็ตแบบลำดับชั้น (Hierarchical Token Bucket) เป็นการใช้แนวคิดของโทเค็น/บัคเก็ต (Tokens/Buckets) และตัวกรองแพ็คเก็ตในการควบคุมปริมาณข้อมูลด้วยแบบจำลองการยืม (Borrowing Model) ที่ซับซ้อน ทำให้สามารถนำไปประยุกต์ใช้ได้หลายวิธี เช่น การจำกัดแบนด์วิดท์ (Bandwidth Shaping) และการรับประกันแบนด์วิดท์ (Bandwidth Guarantee) เป็นต้น

HTB เป็น Classful Qdisc ประเภทหนึ่งที่มาพร้อมกับเคอร์เนลของระบบปฏิบัติการลินุกซ์ตั้งแต่รุ่น 2.4.20 เป็นต้นมา ซึ่งสามารถจำกัดแบนด์วิดท์ได้ถึงระดับของผู้ใช้ หรือ กำหนดตามบริการได้ สามารถจองแบนด์วิดท์ไว้สำหรับบางแอพพลิเคชั่นหรือผู้ใช้บางคนได้และสามารถยืมแบนด์วิดท์จากคลาสอื่นมาใช้งานแบบชั่วคราวได้

2.4.1 แนวคิด โทเค็น/บัคเก็ต (Tokens/Buckets)

โทเค็น/บัคเก็ตเป็นกลไกการควบคุมการจัดเรียงและอัตราความเร็วในการส่งข้อมูล หรือ นำแพ็คเก็ตออกมาจากแถวคอย โดยโทเค็น/บัคเก็ตจะประกอบไปด้วยบัฟเฟอร์ หรือ บัคเก็ต ซึ่งภายในจะเก็บหน่วยข้อมูลหนึ่งๆ เรียกว่า “โทเค็น” (Token) ที่มีการระบุความเร็วไว้แล้ว (Token Rate) โทเค็นจะถูกจับคู่กับแพ็คเก็ตจากแถวคอยและถ้าโทเค็นมีจำนวนมากพอก็จะสามารถทำการส่งข้อมูลได้ แต่ถ้าหากว่าไม่มีโทเค็นเหลืออยู่ในบัคเก็ตก็จะไม่สามารถส่งข้อมูลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



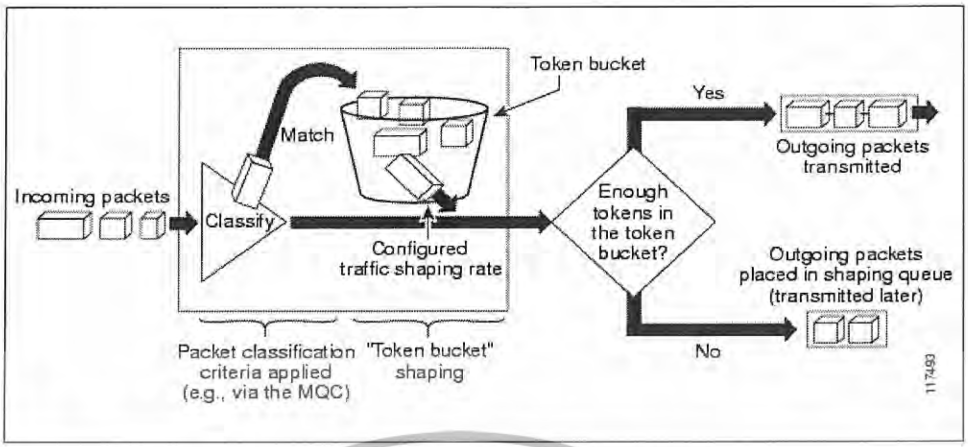
ภาพที่ 2.7 แสดงแนวคิดของโทเค็น/บัคเก็ต

จากภาพที่ 2.7 แสดงถึงแนวคิดของโทเค็น/บัคเก็ต โดยโทเค็นใหม่ที่ถูกสร้างขึ้นจะถูกเพิ่มในบัคเก็ตที่ความเร็ว r โทเค็นต่อ 1 วินาที (Token/Sec) โดยจำนวนโทเค็นสูงสุดที่ถูกสะสมในบัคเก็ตได้จะมีขนาดเท่ากับ b ไบต์ แต่ถ้าหากว่าบัคเก็ตเต็มโทเค็นใหม่ที่เข้ามาจะถูกโยนทิ้ง โดยการทำงานโทเค็น/บัคเก็ตจะประกอบด้วยค่าพารามิเตอร์ 3 ตัวคือ

1. Average Rate คือ ความเร็วเฉลี่ยในการส่งแพ็คเก็ต
2. Peak Rate คือ ค่าความเร็วสูงสุดที่สามารถส่งแพ็คเก็ตได้ในช่วงระยะสั้นๆ
3. Burst Size คือ จำนวนแพ็คเก็ตสูงสุดซึ่งสามารถทำการส่งได้ในช่วงระยะเวลาสั้นๆ

นั่นคือ โทเค็นจะถูกเก็บสะสมมากน้อยขึ้นกับขนาดของบัคเก็ต ซึ่งโทเค็นที่ไม่ได้ใช้จะสามารถสะสมไว้ใช้ในการส่งข้อมูลที่ความเร็วเกินที่กำหนดมาตรฐานของโทเค็น ในกรณีที่มีการส่งข้อมูลแบบเบิร์ส (Burst) เฉพาะระยะสั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.8 แสดงแนวคิดการทำงานของโทเค็น/บัคเก็ต

จากภาพที่ 2.8 ถ้าข้อมูลที่เข้ามามีความเร็วเท่ากับความเร็วของโทเค็น ในกรณีนี้แพ็คเก็ตข้อมูลที่เข้ามาจะถูกจับเข้ากับโทเค็นและถูกส่งออกจากแถวคอยโดยไม่เกิดการหน่วง แต่ถ้าข้อมูลเข้ามาที่ความเร็วมากกว่าความเร็วของโทเค็น นั่นหมายถึงเป็นช่วงที่บัคเก็ตที่เหลืออยู่มีไม่เพียงพอต่อความต้องการทำให้เกิดช่วงเวลาคอขวด ณ เวลาขณะนั้น โดยถ้าหากมีแพ็คเก็ตวิ่งเข้ามาก็จะถูกส่งต่อเข้าไปในแถวคอย

2.4.2 แนวคิดการทำงานของเทคนิคโทเค็น/บัคเก็ตแบบลำดับชั้น (Hierarchical Token Bucket : HTB)

แนวคิดพื้นฐานของการทำงานคือ พิจารณาจากแถวคอยที่ทำการจัดการทราฟฟิกขาออก (Outbound Traffic) ในทุกๆอุปกรณ์ คือ Root Qdisc โดยกำหนดชนิดของแถวคอยสามารถกำหนดให้มีการใช้เทคนิคการจัดการทราฟฟิกในแถวคอยได้เพียงเทคนิคเดียวเท่านั้น เช่น โดยปกติระบบปฏิบัติการลินุกซ์ใช้การจัดการแถวคอยแบบเข้าก่อน-ออกก่อน ซึ่งภายในข้อกำหนดของแถวคอยจะประกอบด้วยคลาสมากกว่า หรือ เท่ากับหนึ่งคลาส โดยคลาสที่เป็น HTB จะถูกเซ็ตด้วยค่าพารามิเตอร์ 2 ชนิด คือ

- ค่า Rate หมายถึง ค่าที่รับประกันแบนด์วิดท์ขั้นต่ำที่คลาสนั้นๆจะต้องมีไว้ใช้งาน
- ค่า Ceil หมายถึง ค่าแบนด์วิดท์สูงสุดของคลาสนั้นๆ ที่อนุญาตให้ใช้งานได้

คลาสลูก (Child node) สามารถยืมใช้แบนด์วิดท์บางส่วนจากคลาสแม่ (Parent Node) ได้ โดยค่าที่สามารถยืมได้จะต้องทำให้แบนด์วิดท์ของคลาสลูกมีค่าไม่เกินค่า Ceil

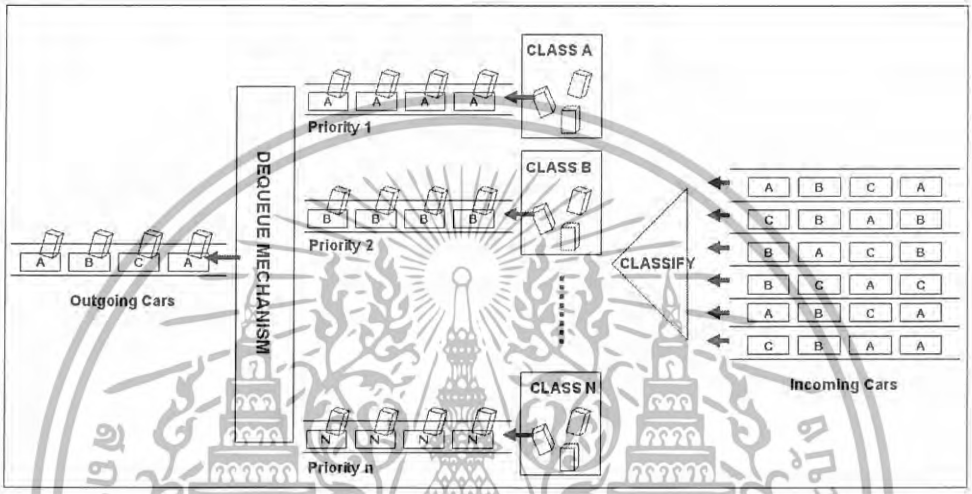
HTB จัดเป็นกลไกการทำงานประเภท Classful Queuing ในระบบปฏิบัติการลินุกซ์ในการควบคุมทราฟฟิกในระบบ โดยมี Rate และ Ceil ที่อนุญาตให้ผู้ใช้สามารถควบคุมแบนด์วิดท์ได้เป็นอย่างดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

สำหรับแต่ละคลาสที่ใช้จัดการทราฟฟิกชนิดต่างๆ ทำให้สามารถนำมาใช้เพื่อจัดการปัญหาคอขวด ระหว่างเครือข่ายเน็ตเวิร์คและอินเตอร์เน็ต

ในโครงสร้างลำดับชั้นของ HTB นั้น แพ็คเก็ตจะถูกจับคู่ได้ก็ต่อเมื่อแพ็คเก็ตนั้นถึงคลาสสุดท้ายแล้ว นอกจากนั้นในโครงสร้างลำดับชั้นสามารถสร้างคลาสได้ทั้งหมดสูงสุด 10,000 คลาส และความลึกของโครงสร้างลำดับชั้นสามารถสร้างความลึกได้ถึงระดับ 8



ภาพที่ 2.9 แสดงตัวอย่างการควบคุมปริมาณทราฟฟิกโดยเทคนิค HTB

จากภาพที่ 2.9 แสดงตัวอย่างการควบคุมปริมาณรถยนต์บนทางด่วน เปรียบได้กับเทคนิคการควบคุมปริมาณทราฟฟิกของ HTB นั่นคือ เนื่องจากปริมาณรถยนต์ที่ต้องการใช้บริการทางด่วนมีปริมาณมาก แต่ทางด่วนไม่สามารถให้บริการรถยนต์ทุกคันวิ่งบนทางด่วนได้ทั้งหมด ดังนั้นจึงจำเป็นต้องมีการควบคุมปริมาณรถยนต์ที่วิ่งบนทางด่วน

โดยรถยนต์จะถูกทำการคัดแยกชนิดของรถยนต์ เพื่อส่งต่อไปด่านเก็บเงินที่แตกต่างกัน ยกตัวอย่างรถยนต์ A เป็นรถยนต์สิบล้อจะถูกส่งไปยังด่านเก็บเงินเฉพาะรถสิบล้อ เป็นต้น เปรียบเสมือนการกรองทราฟฟิกเพื่อส่งไปทำงานต่อยังคลาสต่อไป

ด่านเก็บเงิน มีหน้าที่ในการสร้างตั๋วเก็บไว้เพื่อควบคุมปริมาณรถยนต์ออก เปรียบเสมือนบัคเก็ตที่ทำการเก็บโทเค็นเพื่อใช้ในการควบคุมปริมาณทราฟฟิกต่อไป โดยจะมีขนาดเท่ากับเบิร์สเพื่อรองรับการส่งข้อมูลแบบเบิร์สที่ HTB ได้ทำการสนับสนุนการทำงาน

ตั๋วมีไว้เพื่อควบคุมปริมาณรถออก เปรียบเสมือนโทเค็นที่ใช้ในการควบคุมปริมาณทราฟฟิก โดยในช่วงวินาทีหนึ่งอัตราการสร้างของโทเค็นจะมีจำนวนไม่เกินค่าเบิร์สและไม่ต่ำกว่าค่า Rate ที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
073071

รับประกันแบนด์วิดท์ขั้นต่ำไว้ โดยผลรวมของโทเค็นในแต่ละคลาสจะเปรียบเสมือนแบนด์วิดท์ที่แต่ละคลาสนั้นได้รับ ซึ่งแสดงถึงปริมาณข้อมูลที่สามารถส่งได้ในช่วงระยะเวลาหนึ่ง

รถยนต์ที่ได้รับตั๋วจะได้รับสิทธิ์ในการส่งข้อมูลออกจากด่านเก็บเงิน โดยด่านเก็บเงินจะมีกลไกในการควบคุมปริมาณรถยนต์สูงสุดที่สามารถส่งออกในช่วงระยะเวลาหนึ่ง นั่นคือ Ceil ที่ได้กำหนดไว้

เมื่อรถยนต์ต้องลงจากทางด่วนจะถูกบีบให้เหลือแค่เลนเดียวเท่านั้นที่สามารถออกจากทางด่วนได้ โดย HTB จะเลือกรถยนต์ที่ออกจากด่านเก็บเงินที่มีค่าความสำคัญมากที่สุดออกก่อน เปรียบเสมือนแพ็คเก็ตที่มีค่าลำดับความสำคัญต่ำกว่าออกก่อน

เนื่องจากต้องมีการจำกัดปริมาณรถยนต์ที่ใช้งานทางด่วนในช่วงระยะเวลาหนึ่ง ซึ่งปริมาณรถยนต์จะวิ่งได้มากน้อยเพียงใดขึ้นอยู่กับตั๋วหรือโทเค็น ซึ่งได้รับการจัดสรรไปในแต่ละด่านเก็บเงินในอัตราที่ได้กำหนดไว้ ถ้าหากว่าด่านเก็บเงินไหนมีตั๋วเหลือใช้ เพื่อประสิทธิภาพที่ดีขึ้นของภาพรวม สามารถให้ด่านเก็บเงินอื่นยืมตั๋วได้ เพื่อจะได้ให้บริการรถยนต์คันอื่นที่รอใช้บริการทางด่วน นั่นคือ สนับสนุนคุณสมบัติการยืม/คืนแบนด์วิดท์

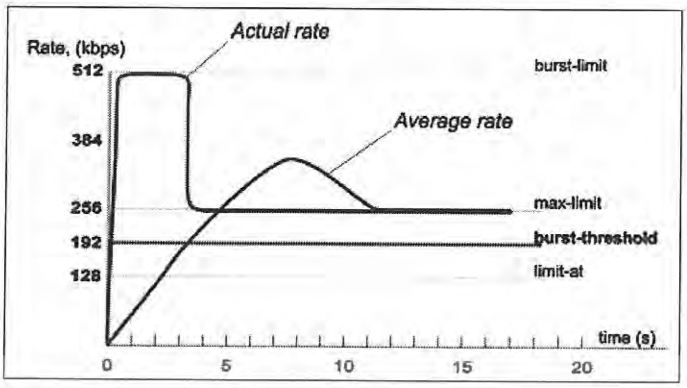
2.4.3 คุณสมบัติการจัดลำดับความสำคัญของคลาสใน HTB (HTB Priority)

ค่าความสำคัญ (Priority) ในแต่ละคลาสจะมีผลสำคัญต่อการจัดลำดับแพ็คเก็ตใน HTB นั่นคือ เมื่อไหร่ที่ leaf class ต้องการจะทำการส่งต่อทราฟฟิก (โดยที่มีแค่บางคลาสที่จะเก็บแพ็คเก็ตไว้ในแถวคอย) ซึ่ง HTB จะให้บริการคลาสที่อยู่ในระดับที่ต่ำที่สุดซึ่งมีค่าความสำคัญสูงที่สุดก่อน หลังจากนั้นจะให้บริการคลาสลำดับชั้นถัดไป จนถึงลำดับชั้นสูงสุด

2.4.4 คุณสมบัติสนับสนุนการส่งข้อมูบแบบเบิร์ส (Burst traffic)

Burst คือ ช่วงเวลาที่อนุญาตให้สามารถส่งข้อมูลในอัตราความเร็วสูงสำหรับในช่วงเวลาสั้นๆ โดยในช่วงเวลาที่เกิดการส่งข้อมูลแบบเบิร์สนั้น HTB จะมีการคำนวณอัตราความเร็วเฉลี่ย (Average Data Rate) ของแต่ละคลาส โดยถ้าค่าเฉลี่ยดังกล่าวน้อยกว่าค่าที่กำหนดไว้ (Burst-Threshold) คลาสก็ยังสามารถใช้ความเร็วในการรับส่งข้อมูลแบบเบิร์สได้อยู่และอัตราความเร็วในการส่งข้อมูล (Actual Data Rate) จริงจะได้ค่าไม่เกิน Burst-Limit ที่ตั้งไว้ แต่ถ้าค่าความเร็วเฉลี่ยของแต่ละคลาสเกินกว่าค่า Burst-Threshold เมื่อไหร่ ค่าอัตราความเร็วในการส่งข้อมูลก็จะตกลงมาอยู่ในช่วง Max-limit และ Limit-at ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.10 แสดงกราฟอัตราความเร็วในการส่งข้อมูล ซึ่งมีอัตราแตกต่างกันในช่วงเวลา Burst Time

2.4.5 การใช้งานชุดคำสั่งควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์ด้วย เทคนิค HTB

การประกาศข้อกำหนดของแถวคอย :

tc qdisc [add | change | replace | link] dev DEV [parent qdisc-id | root] [handle qdisc-id] htb [default N] [r2q N]

การประกาศคลาส :

tc class [add | change | replace] dev DEV parent qdisc-id [classid class-id] qdisc [qdisc specific parameters] [rate R1] [burst B1] [prio P] [ceil R2] [quantum Q]

การประกาศตัวกรองทราฟฟิก :

tc filter [add | change | replace] dev DEV [parent qdisc-id | root] [protocol protocol] [prio priority] filtertype [filtertype specific param-eters] [flowid flow-id]

การดูข้อมูลการทำงานของข้อกำหนดของแถวคอย :

tc [-s | -d] qdisc show [dev DEV]

tc [-s | -d] class show [dev DEV]

การดูข้อมูลการทำงานของข้อกำหนดของแถวคอย :

tc filter show [dev DEV]

ภาพที่ 2.11 ชุดคำสั่งควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์ด้วย HTB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความหมายแต่ละตัวแปรของคลาสใน HTB มีดังนี้

default

เป็นตัวแปรทางเลือก โดยค่าดีฟอล์ตคือ 0 หมายถึงทราฟฟิกใดๆที่ไม่สามารถแบ่งกลุ่มได้จะถูกส่งไปทำงานต่อยังคลาสตามหมายเลขที่กำหนดไว้

rate

ใช้สำหรับกำหนดค่าต่ำสุดของความเร็วที่ต้องการจำกัดทราฟฟิก ต้องพิจารณาการรับประกันแบนด์วิดท์ที่คลาสสุดท้ายด้วย

ceil

ใช้สำหรับกำหนดค่าสูงสุดของความเร็วที่ต้องการจำกัดทราฟฟิกที่สามารถเกินค่าตัวแปร Rate ไปได้ชั่วขณะ

burst

ใช้สำหรับกำหนดขนาดของบัคเก็ต Rate โดย HTB จะนำแพ็คเก็ตออกจากแถวคอยเมื่อมีจำนวนโทเค็นเพียงพอ ขณะที่เกิดทราฟฟิกแบบเบิร์สเข้ามา

quantum

เป็นตัวแปรที่ HTB ใช้ในการควบคุมการยืมแบนด์วิดท์ ซึ่งค่านี้ HTB จะคำนวณให้เองอัตโนมัติ

r2q

เป็นค่าที่คำนวณขึ้นมาจากผู้ใช้งาน ซึ่งจะช่วยกำหนดค่าควอนตัมด้วย

prio

ค่าลำดับความสำคัญของคลาสสุดท้าย ตัวเลขน้อยจะเลือกไปทำงานก่อน

ตัวอย่างการใช้งาน HTB ผ่านชุดคำสั่ง TC Command

1. คำสั่งสร้าง Root Qdisc ของ HTB

```
tc qdisc add dev eth0 root handle 1:0 htb r2q 5 default 4
```

ภาพที่ 2.12 แสดงตัวอย่างคำสั่งสร้าง Root Qdisc ของ HTB

จากคำสั่งเป็นการสร้าง Root Qdisc ของ HTB ที่อินเทอร์เฟส eth0, หมายเลขระบุตัวตนคือ 1:0 หรืออาจจะเขียนเป็น 1: ก็ได้ กำหนดค่า r2q เป็น 5 ซึ่งใช้คำนวณค่าควอนตัมเพื่อใช้ในการควบคุมการยืม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบนด์วิดท์ ดีฟอล์ตคลาสเป็นหมายเลข 4 หมายถึงถ้าหากมีแพ็คเก็ตเข้ามาแล้วทำการไม่สามารถแยกแยะได้ว่าตรงกับคลาสใด ก็จะถูกส่งไปที่คลาสที่มีหมายเลขระบุตัวตน เป็น 1:4 แทน

2. คำสั่งสร้างคลาสราก (Root Class) ของ HTB

```
tc class add dev eth0 parent 1:0 classid 1:1 htb rate 480kbit
```

ภาพที่ 2.13 แสดงตัวอย่างคำสั่งสร้าง Root Class ของ HTB

จากคำสั่งเป็นการสร้างคลาสรากซึ่งอาจจะมีมากกว่าหนึ่งคลาสรากก็ได้ โดยมีคลาสแม่คือ 1:0 มีค่าหมายเลขระบุตัวตน เป็น 1:1 และมีค่าแบนด์วิดท์สูงสุดที่ยอมให้คลาสรากนี้ใช้งานได้คือ 480 กิโลบิตต่อวินาที

3. คำสั่งสร้าง Filter เพื่อกรองทราฟฟิกที่เข้ามาครั้งแรกผ่านคลาสราก

```
tc filter add dev eth0 parent 1:0 protocol all u32 match u32  classid 1:1
```

ภาพที่ 2.14 แสดงตัวอย่างคำสั่งสร้างตัวกรองทราฟฟิก (Filter) ที่คลาสราก

จากคำสั่งเป็นการสร้างตัวกรองทราฟฟิก ทำการกรองทราฟฟิกที่เข้ามาครั้งแรกผ่านคลาสรากโดยอนุญาตให้ทราฟฟิกทั้งหมดสามารถผ่านเข้ามาได้ ผ่านตัวกรองแพ็คเก็ตชนิด u32

4. คำสั่งสร้างคลาสภายใน ของ HTB แบบให้ยืมแบนด์วิดท์ได้ทั้ง คลาสภายใน

```
tc class add dev eth0 parent 1:1 classid 1:2 htb rate 192kbit ceil 480kbit prio 2
tc class add dev eth0 parent 1:1 classid 1:3 htb rate 96kbit ceil 480kbit prio 6
tc class add dev eth0 parent 1:1 classid 1:4 htb rate 192kbit ceil 480kbit prio 8
```

ภาพที่ 2.15 แสดงตัวอย่างคำสั่งสร้างคลาสภายในของ HTB แบบให้ยืมแบนด์วิดท์ทั้งคลาสภายในได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากคำสั่งเป็นการสร้างคลาสลูกภายในคลาสรากซึ่งมีหมายเลขคลาสและลำดับความสำคัญที่แตกต่างกัน รวมถึงค่า Rate ซึ่งเป็นค่าแบนด์วิดท์ต่ำสุดของคลาสที่จะถูกรับประกัน นอกจากนี้ยังจำกัดกำหนดค่า Ceil ให้แต่ละคลาสสามารถยืมแบนด์วิดท์มาใช้งานได้ไม่เกิน 480 กิโลบิตต่อวินาที

5. คำสั่งสร้างคลาสภายในของ HTB แบบไม่ให้ยืมแบนด์วิดท์ข้ามคลาสภายใน

```
tc class add dev eth0 parent 1:1 classid 1:2 htb rate 192kbit ceil 192kbit prio2
tc class add dev eth0 parent 1:1 classid 1:3 htb rate 96kbit ceil 96kbit prio 6
tc class add dev eth0 parent 1:1 classid 1:4 htb rate 192kbit ceil 192kbit prio 8
```

ภาพที่ 2.16 แสดงตัวอย่างคำสั่งสร้างคลาสภายในของ HTB
แบบกำหนดไม่ให้ยืมแบนด์วิดท์ข้ามคลาสภายใน

6. คำสั่งสร้างตัวกรองแพ็คเก็ตเพื่อกรองทราฟฟิกที่เข้ามาในแต่ละคลาสภายใน

```
tc filter add dev eth0 parent 1:1 protocol all prio 2 u32 match u32 0 0 classid 1:2
tc filter add dev eth0 parent 1:1 protocol all prio 2 u32 match u32 0 0 classid 1:3
tc filter add dev eth0 parent 1:1 protocol all prio 2 u32 match u32 0 0 classid 1:4
tc filter add dev eth0 parent 1:1 protocol all prio 2 u32 match u32 0 0 classid 1:5
```

ภาพที่ 2.17 แสดงตัวอย่างคำสั่งสร้างตัวกรองทราฟฟิกที่คลาสภายใน

7. คำสั่งสร้างคลาสภายในของ HTB ให้กับแต่ละบริการของเครือข่าย

```
tc class add dev eth0 parent 1:2 classid 1:21 htb rate 24kbit ceil 480kbit
tc class add dev eth0 parent 1:2 classid 1:22 htb rate 48kbit ceil 480kbit
tc class add dev eth0 parent 1:2 classid 1:23 htb rate 24kbit ceil 480kbit
```

ภาพที่ 2-18 แสดงตัวอย่างคำสั่งสร้างคลาสภายในของ HTB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. คำสั่งกำหนดข้อกำหนดของแถวคอยสำหรับเป็นคลาสสุดท้ายให้กับแต่ละบริการ

```
tc qdisc add dev eth0 parent 1:21 handle 210: fifo
tc qdisc add dev eth0 parent 1:22 handle 220: fifo
tc qdisc add dev eth0 parent 1:23 handle 230: fifo
```

ภาพที่ 2.19 แสดงตัวอย่างคำสั่งสร้างคลาสสุดท้ายให้แต่ละบริการ

9. คำสั่งกำหนดตัวกรองทราฟฟิกให้กับคลาสต่างๆ ที่ได้สร้างไว้แล้วข้างต้นเพื่อทำการกรองหรือจำแนกแพ็คเก็ตให้ไหลเข้าไปทำงานแต่ละคลาส

```
tc filter add dev eth0 parent 1:0 protocol ip prio 1 u32 match ip dst 192.168.1.0/26 \
match ip sport 22 0xffff flowid 1:21
```

ภาพที่ 2.20 แสดงตัวอย่างคำสั่งสร้างตัวกรองทราฟฟิก

จากคำสั่งเป็นการกรองแพ็คเก็ตที่เครือข่าย 192.168.1.0 ซึ่งค่า flowid จะเป็นตัวกำหนดว่า แพ็คเก็ตที่ตรงกับเงื่อนไขใดก็จะถูกไหลเข้าไปทำงานต่อที่ flowid นั้น

8. คำสั่งดูรายละเอียดของข้อกำหนดของแถวคอยคลาสและตัวกรองแพ็คเก็ตที่ได้กำหนดไปข้างต้น

```
tc [qdisc|class|filter] show eth0
```

ภาพที่ 2.21 แสดงตัวอย่างคำสั่งแสดงรายละเอียดข้อมูลข้อกำหนดของแถวคอยคลาส
และตัวกรองที่อินเทอร์เฟส eth0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.6 คุณสมบัติการยืมและคืนแบนด์วิดท์ (Borrowing)

การยืมแบนด์วิดท์เป็นกลไกพื้นฐานส่วนหนึ่งของ HTB ซึ่งคลาสลูกจะสามารถยืมโทเค็นจากคลาสแม่ได้ถ้ามีการใช้งานแบนด์วิดท์เกินกว่าโทเค็นที่มีอยู่ โดยคลาสลูกจะพยายามยืมแบนด์วิดท์จากคลาสแม่จนกระทั่งค่าแบนด์วิดท์ทั้งหมดที่มีถึงค่า Ceil ที่มีกำหนดไว้ โดยจะมีการทำงานตามกลไกการยืม ดังตารางต่อไปนี้

ตารางที่ 2.1 สถานะของคลาสและการยืมแบนด์วิดท์ของ HTB

ชนิดคลาส	คลาส	สถานะ	การทำงาน
Leaf	*< rate*	*HTB_CAN_SEND*	- คลาสสุดท้ายสามารถให้คลาสอื่นภายใต้คลาสแม่เดียวกันยืมโทเค็นไปใช้งานได้ - คลาสสุดท้ายจะทำการส่งแพ็คเก็ตออกจากแถวคอยได้ขึ้นกับจำนวนโทเค็นที่มีอยู่ ในกรณีที่ไม่มีการส่งข้อมูลแบบหนาแน่น
Leaf	*> rate, < ceil*	*HTB_MAY_BORROW*	- คลาสสุดท้ายสามารถยืมโทเค็นจากคลาสแม่ได้ ถ้าหากคลาสแม่ยังมีโทเค็นเหลืออยู่ โดยยืมได้ตามจำนวนค่าควอนตัมที่กำหนดไว้และยืมได้มากที่สุดไม่เกินค่าที่กำหนดไว้
Leaf	*> ceil*	*HTB_CANT_SEND*	- ไม่มีแพ็คเก็ตที่จะเข้ามาในแถวคอยได้ ทำให้แพ็คเก็ตถูกหน่วงได้
inner, root	*< rate*	*HTB_CAN_SEND*	คลาสภายในสามารถให้คลาสลูกของตัวเองยืมโทเค็นได้
inner, root	*> rate, < ceil*	*HTB_MAY_BORROW*	คลาสภายในสามารถยืมโทเค็นจากคลาสแม่ของตนได้และสามารถให้คลาสลูกของตัวเองยืมได้ตามจำนวนควอนตัมที่กำหนดไว้
inner, root	*> ceil*	*HTB_CANT_SEND*	คลาสภายในไม่สามารถยืมโทเค็นจากคลาสแม่ได้และไม่สามารถให้คลาสลูกของตัวเองยืมได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการยืมและคืนแบนด์วิดท์ระหว่างคลาสลูกและคลาสแม่นั้น ปริมาณแบนด์วิดท์ที่แต่ละคลาสจะได้รับจะขึ้นกับแนวคิดพื้นฐานของการทำงานควอนตัม คือ ในความเป็นจริงเมื่อหลายๆ คลาสต้องการที่จะยืมแบนด์วิดท์จากคลาสแม่ จำเป็นต้องใช้ตัวเลขไบต์ค่าหนึ่งในการขอใช้แบนด์วิดท์ก่อนที่จะได้รับแบนด์วิดท์จากคลาสแม่ โดยคลาสจะได้รับจำนวนแบนด์วิดท์ที่ยืมมาจากคลาสแม่ในอัตราส่วนเดียวกันกับควอนตัม

ในการทำงาน ค่าควอนตัมควรที่จะมีค่าน้อยที่สุดเท่าที่จะทำได้ แต่ควรที่มีขนาดใหญ่กว่าค่า MTU เพื่อที่ว่าจะได้สามารถส่งแพ็คเก็ตได้มากที่สุดในหนึ่งครั้ง แต่ควรจะมีค่าน้อยกว่า 60,000 เพื่อป้องกันปัญหาการรอคอยนิรันด์ (Starvation) และการหน่วงที่นานเกินไป

โดยปกติไม่จำเป็นต้องทำการระบุค่าควอนตัมเองเนื่องจาก HTB จะทำการคำนวณให้เองโดยอัตโนมัติ โดยจะทำการคำนวณค่าควอนตัมในแต่ละคลาส โดยการนำ Rate ของแต่ละคลาสมาทำการหารด้วยค่าพารามิเตอร์ “r2q” แต่ถึงอย่างไรความเร็วในการส่งข้อมูลเมื่อคลาสนั้นได้รับอนุญาตให้ทำการส่งก็ยังคงขึ้นอยู่กับค่าพารามิเตอร์ Rate และ Ceil แม้ว่าในคลาสนั้น จะมีค่าควอนตัมขนาดใหญ่ก็ตาม

2.5 การควบคุมปริมาณข้อมูลด้วยเทคนิคการประกันบริการแบบลำดับชั้นด้วยเส้นโค้งการรับประกันการให้บริการ (Hierarchical Fair Service Curve : HFSC)

การเพิ่มขึ้นของเครือข่ายที่ให้บริการในปัจจุบัน ต้องมีการสนับสนุนการทำงานของโปรแกรมในปัจจุบันที่มีการทำงานซับซ้อนและมีการสนับสนุนคุณลักษณะของทราฟฟิกที่มีความหลากหลายได้โดยมุ่งความสนใจไปที่การประกันคุณภาพของการให้บริการบนเครือข่ายรวม (QoS) โดยเฉพาะทราฟฟิกประเภทเรียลไทม์

HFSC จะมีเป้าหมายในการทำงาน 3 ด้าน คือ

1. มีการรับประกันคุณภาพบริการคลาสสุดท้ายโดยใช้เทคนิคเส้นโค้งการประกันการให้บริการ
2. ในคลาสกลาง (Interior Class) จะพยายามให้ผลต่างของการใช้แบนด์วิดท์ของคลาสลูก (Child Node) น้อยที่สุด เพื่อความยุติธรรมในการกระจายการใช้ทรัพยากร
3. มีการจัดสรรแบนด์วิดท์ ที่เป็นส่วนที่ไม่จำเป็นในการใช้ในขณะนั้น ให้กับคลาสอื่นที่มีคลาสแม่เป็นคลาสเดียวกัน (Sibling Class) ได้อย่างยุติธรรม

การรับประกันการให้บริการจะอยู่ที่คลาสสุดท้าย โดยการทำงานของคลาสสุดท้ายจะแบ่งตามแอพพลิเคชั่น ซึ่งจะอ้างถึงว่าเป็น 1 งานที่เราต้องให้บริการ ดังนั้นสามารถเตรียมการรับประกันการให้บริการแบบเรียลไทม์ในแต่ละงานโดยการใช้การรับประกันคลาสสุดท้ายที่อยู่ในโครงสร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1 แบบจำลองเส้นโค้งการรับประกันการให้บริการอย่างยุติธรรมของการใช้การเชื่อมต่อร่วมกัน (Fair Service Curve link-sharing Model: FSC Link-Sharing)

FSC Link-Sharing มีรูปแบบการทำงานที่นำเอาหลักการทำงานพื้นฐานที่สำคัญ คือ การให้บริการแบบลำดับชั้น การจัดการบริการที่ต้องการการตอบสนองทันที การประกันคุณภาพโดยใช้เส้นโค้งการประกันการให้บริการ การจัดการบริการอย่างยุติธรรม และ การให้บริการแบบลำดับความสำคัญ เพื่อนำมาจัดการกับค่าความหน่วงการให้บริการและใช้ในการจัดสรรแบนด์วิดท์ ซึ่งสามารถควบคุมจัดการและรับประกันคุณภาพของการให้บริการประเภทต่างๆบนเครือข่ายได้

2.5.2 การให้บริการแบบลำดับชั้น (Hierarchical Link – Sharing)

จุดประสงค์ที่สำคัญของ Hierarchical Link-Sharing คือ

- สามารถรับประกันได้ว่าแต่ละคลาสจะต้องได้รับทรัพยากรอย่างน้อยที่สุดเพียงพอต่อความต้องการ
- สามารถรับได้ประกันว่าจะมีเกณฑ์ในการกระจายแบนด์วิดท์ที่เป็นส่วนที่ไม่จำเป็นในการใช้ในขณะนั้น ให้กับคลาสอื่นที่มีคลาสแม่เป็นคลาสเดียวกันได้อย่างยุติธรรม

2.5.3 การประกันคุณภาพโดยใช้เส้นโค้งการรับประกันการให้บริการ

ข้อดีของการรับประกันคุณภาพของการให้บริการ โดยใช้เส้นโค้งการประกันการให้บริการ คือ มีความยืดหยุ่นในการจัดสรรทรัพยากร (ในที่นี้คือแบนด์วิดท์) ให้แก่บริการที่มีค่าความหน่วงและความต้องการค่าทรูพุตที่หลากหลายๆได้

2.5.3.1 แนวคิดของเส้นโค้งการรับประกันการให้บริการ

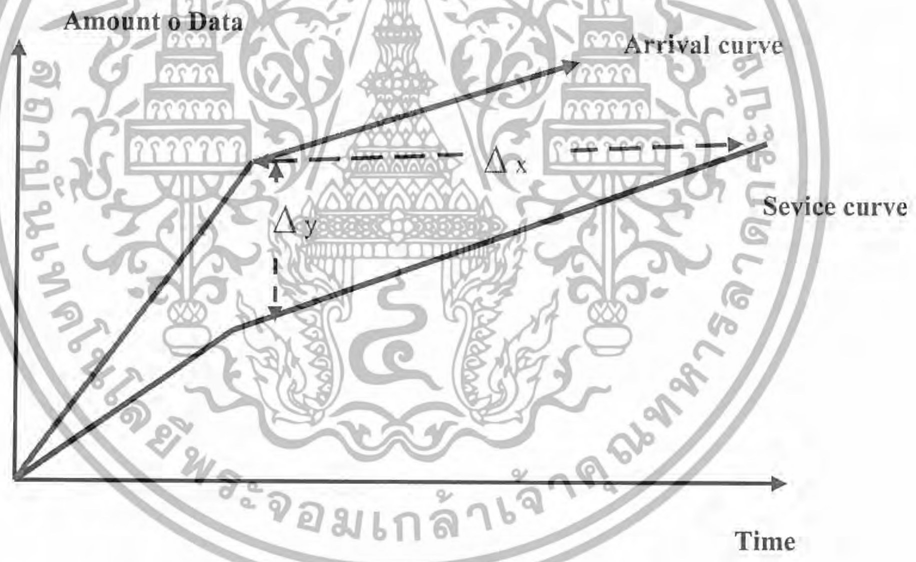
แนวคิดอย่างง่ายของเส้นโค้งการประกันการให้บริการ คือ แต่ละคลาสในลำดับชั้นจะมีการกำหนดผลรวมของการให้บริการ (มีหน่วยเป็น Bits) ที่คลาสนั้นๆ ให้บริการแก่บริการ หรือ การเชื่อมต่อที่เข้ามา ซึ่งอัตราของการให้บริการขั้นต่ำของคลาสที่รองรับการเชื่อมต่อ i ในช่วงเวลา $[0,t)$ จะสามารถเขียนแทนด้วยเส้นโค้งการประกันการให้บริการ $S_i(t)$ ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.3.2 เส้นโค้งการรับประกันการให้บริการและ การรับประกันความสามารถ

จากภาพที่ 2.22 ถ้าเส้นโค้งการประกันการให้บริการ เป็นการอ้างถึงอัตราการให้บริการที่มีการรับประกันไว้ นั่นคืออัตราที่บริการหรืองานที่เข้ามายังคลาสนั้นๆ จะถูกให้บริการหรือส่งออกจากคลาสซึ่งจะต้องไม่ต่ำกว่าอัตราที่กำหนดไว้ในเส้นโค้งการให้บริการ เส้นโค้งการให้บริการจริง (Arrive Curve) เป็นอัตราที่คลาสนั้นได้ให้บริการหรือทำการส่งข้อมูลออกไปจริงๆในชั่วระยะเวลาหนึ่งๆ ดังนั้น การให้บริการแก่งานที่เข้ามานั้น คลาสจะสามารถหน่วงอัตราการให้บริการได้ โดยมีขอบเขตที่จะสามารถเกิดการหน่วงการให้บริการโดยไม่ทำให้อัตราการให้บริการจริงต่ำกว่าที่รับประกันไว้

ในเส้นโค้งการประกันการให้บริการ คือ ระยะทางห่างในแกนนอนระหว่างเส้นโค้งการให้บริการจริงกับเส้นโค้งการประกันการให้บริการ และระยะห่างในแกนตั้งระหว่างเส้นโค้งการให้บริการจริงกับเส้นโค้งการประกันการให้บริการจะบอกถึงความสามารถของคลาสว่า ณ เวลาหนึ่งๆ คลาสสามารถให้บริการได้มากกว่าที่มีการรับประกันไว้เท่าใด



ภาพที่ 2.22 กราฟแสดงเส้นโค้งการประกันการให้บริการที่บ่งบอกถึงผลรวมของแบนด์วิดท์ที่ได้มีการประกันไว้

2.5.3.3 เส้นโค้งการประกันบริการและการจัดการบริการอย่างยุติธรรม

ในขณะที่เส้นโค้งการการประกันการให้บริการ โดยทั่วไปจะมุ่งไปที่การรับประกันผลรวมที่น้อยที่สุดของการให้บริการที่ให้แก่งานหรือบริการที่เข้ามายังคลาส ซึ่งไม่ได้กล่าวถึงการจัดการในส่วนของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บริการ (ในที่นี้คือ แบนด์วิดท์) ที่เกินหรือเหลือใช้ที่เป็นส่วนที่เกินกว่าที่คลาสต้องการใช้รับประกันการให้บริการให้แก่คลาสที่มีความต้องการใช้ทรัพยากรในการเพิ่มประสิทธิภาพการทำงาน

การจัดการการกระจายแบนด์วิดท์อย่างยุติธรรมมีความสำคัญ คือ ในการกระตุ้นให้บริการแบบการแชร์การเชื่อมต่อมีการแบ่งปันทรัพยากรแบบไดนามิก (Dynamic Sharing) ให้แก่คลาสที่มีบรรพบุรุษ (Parent Class) เดียวกันนั้น การแบ่งปันทรัพยากรแบบไดนามิกจะมีความสำคัญ คือ ถ้าบางแอพพลิเคชั่นในคลาสมีการทำงานแบบปรับตัวได้ (Adaptive) นั่นคือ ในช่วงระยะเวลาหนึ่งสามารถทำการส่งข้อมูลเกินกว่าค่าแบนด์วิดท์ที่น้อยที่สุดที่ได้รับประกันไว้ได้ การจัดการการกระจาย (Fair Scheduler) จะทำให้เกิดการเพิ่มประสิทธิภาพการทำงานของการให้บริการของคลาสและทำให้อัตราการดรอปแพ็คเก็ตที่เข้ามายังคลาสต่ำลง ซึ่งจะเป็นการสนับสนุนความยืดหยุ่นของความสามารถของแอพพลิเคชั่น ตัวอย่างเช่น แอพพลิเคชั่นประเภทวีดีโอ (Video Application) นอกจากจะมีการจองแบนด์วิดท์ขั้นต่ำที่ใช้ในการส่งแล้ว ก็ยังสามารถใช้แบนด์วิดท์ที่เป็นส่วนเกินจากคลาสอื่นได้เพื่อเพิ่มคุณภาพของการให้บริการได้

2.5.4 เทคนิคการประกันบริการแบบลำดับชั้นด้วยเส้นโค้งการรับประกันการให้บริการ (HFSC)

HFSC นำหลักการทำงานพื้นฐานของแบบจำลองเส้นโค้งประกันบริการอย่างยุติธรรมของการใช้การเชื่อมต่อร่วมกัน (Fair Service Curve Link-Sharing Model) มาเป็นพื้นฐานในการทำงาน โดยหลักการการจัดการ (Schedule) จะแบ่งออกเป็น 2 การทำงานหลักๆ คือ

- การทำงานโดยใช้หลักการของเรียลไทม์ (Real-Time Criterion) มีเป้าหมายของการทำงาน คือ เพื่อให้สามารถมั่นใจได้ว่าเส้นโค้งการประกันการให้บริการของคลาสสุดท้ายทั้งหมดจะสามารถรับประกันได้ว่าคลาสได้รับแบนด์วิดท์ตามที่รับประกันไว้จริง โดยเน้นการทำงานไปที่การจัดการค่าความหน่วงของแพ็คเก็ต
- การทำงานโดยใช้หลักการของการแชร์การเชื่อมต่อ (Link-Sharing Criterion) มีเป้าหมายให้เส้นโค้งการประกันการให้บริการของคลาสภายในจะได้รับแบนด์วิดท์ในระดับที่เพียงพอและมีการกระจายแบนด์วิดท์ที่เหลือใช้จากคลาสลูกใดๆ ไปยังคลาสลูกอื่นๆ อย่างยุติธรรม ซึ่งจะพยายามให้การใช้งานแบนด์วิดท์ของคลาสลูกมีความเท่าเทียมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

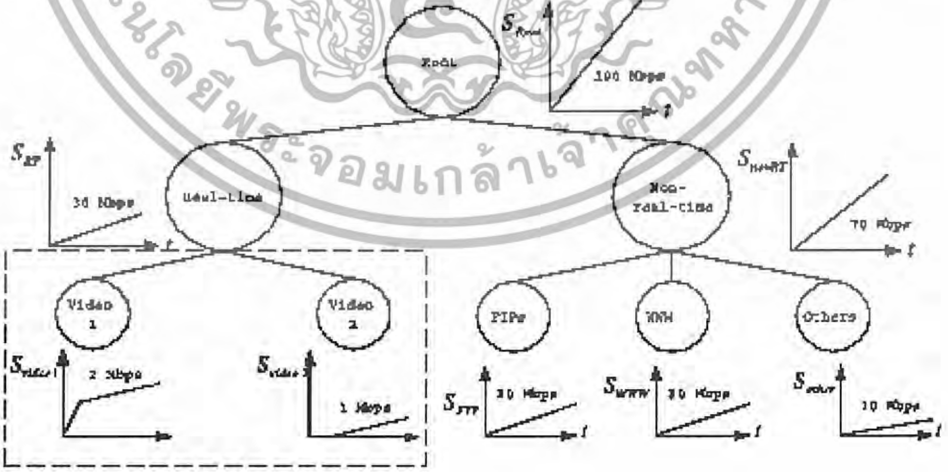
2.5.4.1 แนวคิดและหลักการทำงานพื้นฐานของ HFSC

HFSC จะใช้โครงสร้างลำดับชั้นที่ประกอบไปด้วยคลาส ซึ่งแสดงถึงทราฟฟิกของข้อมูล โดยแต่ละคลาสจะถูกสร้างขึ้นมาพิเศษเพื่อให้บริการการแชร์การเชื่อมต่อลำดับชั้น แต่ยกเว้นสำหรับกรณีแอพพลิเคชั่นหรือบริการระดับผู้ใช้จะมีความสัมพันธ์กับคลาสสุดท้ายนั้น HFSC จะทำงานบนพื้นฐานของเส้นโค้งการประกันการให้บริการซึ่งจะใช้ในการระบุถึงการของการรับประกันคุณภาพของบริการหรืองานในส่วนของแบนด์วิดท์ที่ได้รับและขอบเขตของการเกิดการหน่วงอัตราการให้บริการ โดยงาน (Session) ที่ i นั้นจะถูกรับประกันโดยเส้นโค้งการประกันการให้บริการ $S_i(t)$ ตามสมการข้างล่าง ดังนี้

$$S_i(t_2 - t_1) \leq W_i\ (t_2, t_1)\ ;\ t_1\ < t_1 \qquad (2.1)$$

โดยจากสมการที่ 2.1 ค่า t_2 คือช่วงเวลาเวลาที่ทำการส่งบิตสุดท้ายของแพ็คเก็ตของงาน (Session) i, t_1 คือ เวลาเริ่มต้นที่คลาสงาน (Session i) เข้ามา และ W_i คือ ผลรวมทั้งหมดของแบนด์วิดท์ที่งาน (Session i) ได้รับในช่วงระยะเวลา $(t_2\ ,t_1]$ ซึ่งโดยปกติงานประเภทเรียลไทม์ (Real Time Session) จะต้องการค่าความหน่วงที่ต่ำและอัตราการส่งข้อมูลปานกลาง แต่ว่างานที่ไม่ใช่ประเภทเรียลไทม์ (Non-Real Time Session) จะทำการรับประกันอัตราการส่งข้อมูลและความหน่วงที่สูงกว่า

แอพพลิเคชั่นประเภทเรียลไทม์นั้น แต่ละประเภทของแอพพลิเคชั่นจะถูกแยกตามคลาสสุดท้าย ซึ่งบริการหรือแบนด์วิดท์ที่แต่ละคลาสได้รับนั้นสามารถมีการแบ่งปันการใช้กับแอพพลิเคชั่นที่ไม่ใช่ประเภทเรียลไทม์ได้ด้วย โดยแบนด์วิดท์ที่แต่ละคลาสจะได้รับพิจารณาจากการจัดสรรแบนด์วิดท์

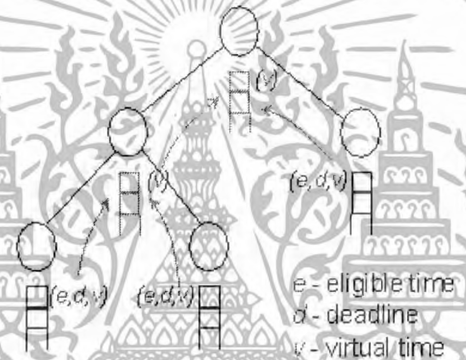


ภาพที่ 2.23 แสดงถึงตัวอย่างโครงสร้างลำดับชั้นของแต่ละคลาสใน HFSC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 2.23 จะแสดงตัวอย่างของโครงสร้างลำดับชั้น จากตัวอย่าง คลาสรากจะแสดงเส้นโค้งการประกันการให้บริการเป็นเส้นตรงเฉียงที่มีค่าความชัน เท่ากับ 100 เมกะบิตต่อวินาที โดยแต่ละคลาสในระดับถัดมาจะได้รับการรับประกันแบนด์วิดท์ให้เป็นไปตามที่กำหนดไว้ซึ่งผลรวมของบริการที่รับประกันไว้ให้คลาสลูกจะต้องไม่เกินกว่าผลรวมการประกันบริการของคลาสแม่ โดยคลาสสุดท้ายอาจมีการรับประกันแบนด์วิดท์ขั้นต่ำที่เหมือนกัน แต่มีอัตราการกระจายแบนด์วิดท์ที่เหลือใช้ต่างกัน โดยคุณสมบัติดังกล่าวทำให้เกิดความยืดหยุ่นในการให้บริการของคลาสต่างๆ ในโครงสร้าง

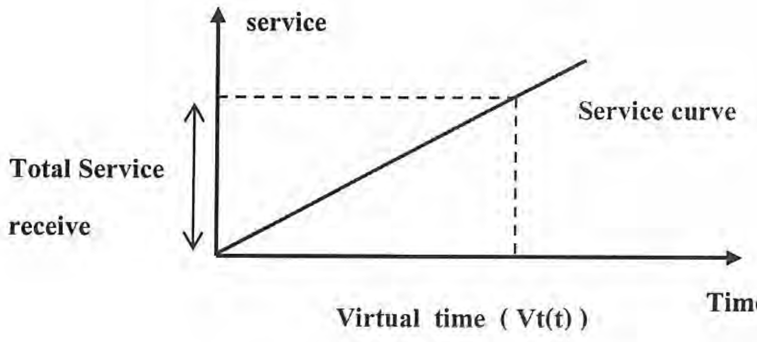
HFSC จะมีการทำงานโดยแยกเป็น 2 หลักการทำงานซึ่งทำงานร่วมกันในโครงสร้างเดียวกัน โดยในแต่ละคลาสจะมีตัวแปรที่เกี่ยวข้องอยู่ 3 ตัวแปรคือ ค่า Eligible Time, Deadline Time และ Virtual Time



ภาพที่ 2.24 แสดงถึงตัวอย่างโครงสร้าง HFSC และความสัมพันธ์ระหว่าง 3 ตัวแปร คือ Eligible Time, Deadline และ Virtual Time

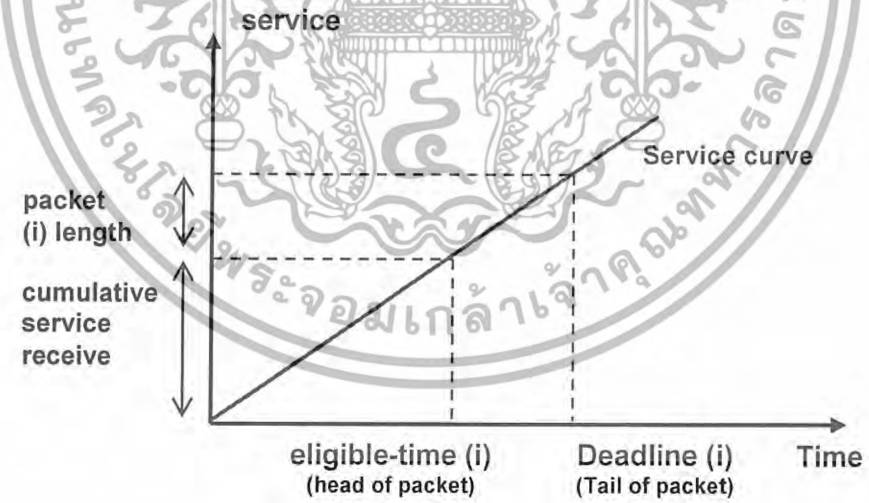
จากภาพที่ 2.24 จะแสดงให้เห็นถึงความสัมพันธ์ของทั้งสามตัวแปรกับคลาสในลำดับชั้น ซึ่งคลาสภายใน จะเกี่ยวข้องกับตัวแปร Virtual time เพียงตัวแปรเดียว แต่ในคลาสสุดท้ายจะเกี่ยวข้องกับตัวแปรทั้งสาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 2.25 แสดงถึงค่า Virtual Time ของคลาส

ในแต่ละคลาสของโครงสร้างของ HFSC จะมีการเก็บผลรวมของบริการที่คลาสนั้นได้ให้บริการออกไปไว้ ซึ่งจากภาพที่ 2.25 ถ้าลากจุดตัดในแนวแกนตั้งที่เป็นผลรวมของการให้บริการของคลาสนั้นไปยังเส้นโค้งการรับประกันการให้บริการแล้วลากจากจุดตัดดังกล่าวไปที่แนวแกนนอน ก็จะได้ค่า Virtual Time ซึ่งนำมาใช้ในการพิจารณาเลือกคลาสที่จะสามารถส่งแพ็คเก็ตได้ ในการทำงานแบบใช้หลักการแชร์การเชื่อมต่อ



ภาพที่ 2.26 แสดงถึงค่า Eligible Time และ Deadline Time ของคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 2.26 เนื่องจากแต่ละคลาสจะมีการเก็บค่าผลรวมของบริการที่คลาสนั้นได้ให้บริการไว้ ซึ่งถ้าลากไบต์สุดท้ายของผลรวมการให้บริการมายังเส้นโค้งการรับประกันการให้บริการและลากจากจุดตัดดังกล่าวมาที่แนวแกนนอนแล้ว จะได้ค่าของ Eligible Time ของแพ็คเก็ตที่อยู่ในคลาสนั้นๆ ซึ่งจากภาพจะเห็นว่าค่า Deadline จะเป็นค่าของเวลาที่ส่งแพ็คเก็ตที่มีความยาว i ออกจากคลาส ซึ่งจากการทำงานของ HFSC ข้างต้นนั้น จะมีการทำงาน 2 แบบ ซึ่งในแต่ละช่วงเวลาจะมีการเลือกทำงานเพียงแบบเดียว

โดยปัจจัยในการเลือกแบบการทำงานของ HFSC นั้นจะพิจารณาที่ ค่า Eligible ในแต่ละคลาสซึ่งถ้า ณ เวลาใดๆ คลาสใดๆ ก็ตาม มีค่า Eligible Time ที่กำหนดให้แพ็คเก็ตน้อยกว่าค่าเวลาในปัจจุบันแล้ว จะกล่าวได้ว่าแพ็คเก็ตของคลาสนั้นเป็นแพ็คเก็ตที่ Eligible และถ้าคลาสสุดท้าย คลาสใดคลาสหนึ่งในลำดับชั้นมีแพ็คเก็ตแบบ Eligible แล้ว HFSC จะมีการทำงานแบบใช้หลักการของเรียลไทม์ (Real-Time Criterion) ซึ่งจะมีกลไกเลือกคลาสที่จะสามารถส่งแพ็คเก็ตออกได้โดยเลือกคลาสที่มีค่า Deadline Time น้อยที่สุดจากกลุ่มของคลาสที่มีแพ็คเก็ตแบบ Eligible

แต่ถ้า ณ เวลาช่วงหนึ่งไม่มีคลาสใดๆ มีแพ็คเก็ตแบบ Eligible แล้ว HFSC จะเปลี่ยนไปใช้การทำงานแบบการแชร์การเชื่อมต่อ (Link-Sharing Criterion) ซึ่งจะเริ่มทำงานจากคลาสรากลงไปยังลำดับชั้นต่างๆ ในโครงสร้าง ซึ่งมีกลไกการเลือกคลาสที่จะสามารถนำแพ็คเก็ตออกได้โดยมีการพิจารณาคลาสที่มีค่า Virtual Time น้อยที่สุด เป็นคลาสที่สามารถส่งแพ็คเก็ตออกได้ ในช่วงเวลานั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.5 การใช้งานชุดคำสั่งควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์ด้วย HFSC

การประกาศข้อกำหนดของแถวคอย :

```
tc qdisc [ add | change | replace | link ] dev DEV [ parent qdisc-id | root ]
[ handle qdisc-id] htb [default N] [r2q N]
```

การประกาศคลาส :

```
tc class [ add | change | replace ] dev DEV parent qdisc-id [ classid class-id ]
qdisc [ qdisc specific parameters ]
sc [m1 rate] [d ms] [m2 rate]
rc [m1 rate] [d ms] [m2 rate]
ls [m1 rate] [d ms] [m2 rate]
ul [m1 rate] [d ms] [m2 rate]
```

การประกาศตัวกรองแพ็คเก็ต :

```
tc filter [ add | change | replace ] dev DEV [ parent qdisc-id | root ] [protocol protocol]
[prio priority] filtertype [ filtertype specific param-eters ] [flowid flow-id]
```

การดูข้อมูลการทำงานของข้อกำหนดของแถวคอย :

```
tc [-s | -d ] qdisc show [ dev DEV ]
tc [-s | -d ] class ls [dev DEV]
```

การดูข้อมูลการทำงานของ Filter :

```
tc filter show [dev DEV]
```

ภาพที่ 2.27 แสดงคำสั่งควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์ด้วย HFSC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความหมายแต่ละตัวแปรของคลาสใน HFSC มีดังนี้

default

เป็นตัวแปรที่ระบุว่าคลาสนั้นเป็นดีฟอล์ตคลาส หมายถึง ทราฟฟิกใดๆที่ไม่สามารถแบ่งกลุ่มได้ก็จะถูกส่งออกจากแถวคอยนี้

bandwidth

เป็นตัวแปรแสดงถึงจำนวนแบนด์วิดท์ทั้งหมดของอินเตอร์เฟสนั้นๆ

m1

เป็นค่าความชันของเส้นโค้งการประกันการให้บริการในส่วนแรก ในกรณีที่เส้นโค้งการประกันการให้บริการมีลักษณะเป็นเส้นโค้งการประกันการให้บริการอย่างต่อเนื่องแบบที่มีความชันสองค่า (Two-Piece Linear Service Curve)

โดยค่าความชันที่คำนวณได้จากเส้นโค้งการประกันการให้บริการจะแสดงถึงอัตราเร็วในการส่งข้อมูลแบบเบิร์ส โดยถ้าหากเซ็ตให้ค่า m1 มีค่าน้อยกว่าค่า m2 แล้ว HFSC จะคิดว่าค่า m1 เป็นศูนย์

d

เป็นค่า x ณ จุดเชื่อมต่อระหว่าง 2 ค่าความชันในเส้นโค้งการประกันการให้บริการอย่างต่อเนื่องแบบที่มีความชันสองค่า

m2

เป็นค่าความชันของเส้นโค้งการประกันการให้บริการในส่วนที่สอง ในกรณีที่เส้นโค้งการประกันการให้บริการมีลักษณะเป็นเส้นโค้งการประกันการให้บริการอย่างต่อเนื่องแบบที่มีความชันสองค่า

โดยค่าความชันที่คำนวณได้จากเส้นโค้งบริการจะแสดงถึงอัตราเร็วในการส่งข้อมูลขั้นต่ำที่ได้รับประกันไว้

sc

เป็นการประกาศเส้นโค้งการประกันการให้บริการให้คลาสๆ นั้นทำงานด้วยอัลกอลิทึมหลักการของเรียลไทม์ (Real-Time Criterion) และหลักการของการแชร์การเชื่อมต่อ (Link-Sharing Criterion)

rt

เป็นการประกาศเส้นโค้งการประกันการให้บริการให้คลาสๆ นั้นทำงานด้วยอัลกอลิทึมหลักการของเรียลไทม์

ls

เป็นการประกาศเส้นโค้งการประกันการให้บริการให้คลาสๆ นั้นทำงานด้วยอัลกอลิทึมหลักการของการแชร์การเชื่อมต่อ (Link-Sharing Criterion)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ul

เป็นการประกาศเส้นโค้งการประกันการให้บริการให้คลาสๆ นั้นโดยค่าความชันที่คำนวณได้จากเส้นโค้งบริการจะแสดงถึงอัตราเร็วในการส่งข้อมูลสูงสุดของคลาสๆนั้น

ตัวอย่าง การใช้งาน HFSC ผ่านชุดคำสั่ง TC Command

1. คำสั่งสร้าง Root Qdisc ของ HFSC

```
tc qdisc add dev eth2 handle 1: root hfsc default 16
```

ภาพที่ 2.28 แสดงตัวอย่างคำสั่งการสร้าง Root Qdisc ของ HFSC ที่ขาอินเทอร์เฟส Eth2

จากคำสั่งเป็นการสร้าง Root Qdisc ของ HFSC ที่อินเทอร์เฟส Eth0 และมีหมายเลขระบุตัวตน (handle) คือ 1:0 หรืออาจจะเขียนเป็น 1: ก็ได้ โดยกำหนดให้คลาสดีฟอล์ตเป็นหมายเลข 16 หมายถึงถ้าหากมีแพ็คเก็ตเข้ามาแล้วทำการไม่สามารถแยกแยะได้ว่าตรงกับคลาสไหน ก็จะให้ไปทำงานที่คลาสที่มีหมายเลขระบุตัวตนเป็น 1:16 แทน

2. คำสั่งสร้างคลาสภายในของ HFSC ต่อจากระดับแรกสำหรับผู้ใช้แต่ละคนหรือแยกให้บริการเป็นประเภทเรียลไทม์กับงานที่ไม่ใช่ประเภทเรียลไทม์ก็ได้

```
tc class add dev eth2 parent 1: classid 1:1 hfsc sc m1 0kbit d 0us m2 1Mbit ul m1 0kbit d 0us m2
1Mbit
```

ภาพที่ 2.29 แสดงตัวอย่างคำสั่งการสร้างคลาสแม่

จากคำสั่งเป็นการสร้างคลาสโดยมีคลาสแม่ คือ 1:0 และมีค่าหมายเลขระบุตัวตนเป็น 1:10 โดยกำหนดให้มีแบนด์วิดท์เท่ากับ 1 เมกะบิตต่อวินาทีและจำนวนแบนด์วิดท์สูงสุดที่ได้รับเท่ากับ 1 เมกะบิตต่อวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. คำสั่งสร้างคลาสสุดท้ายของ HFSC เพื่อระบุถึงชนิดทราฟฟิกที่แตกต่างกัน

```
tc class add dev eth2 parent 1:1 classid 1:11 hfsc rt m1 12kbit d 100ms m2 200kbit ls m2 200kbit ul m2 350kbit
tc class add dev eth2 parent 1:1 classid 1:12 hfsc rt m1 12kbit d 400ms m2 100kbit ls m2 100kbit ul m2 350kbit
tc class add dev eth2 parent 1:1 classid 1:13 hfsc rt m1 12kbit d 100ms m2 35kbit ls m2 35kbit ul m2 350kbit
tc class add dev eth2 parent 1:1 classid 1:14 hfsc rt m1 12kbit d 400ms m2 250kbit ls m2 250kbit ul m2 550kbit
tc class add dev eth2 parent 1:1 classid 1:16 hfsc ls m2 100kbit ul m2 300kbit
```

ภาพที่ 2.30 แสดงตัวอย่างคำสั่งการสร้างคลาสสุดท้ายของ HFSC

จากคำสั่งเป็นการสร้างคลาสสุดท้ายของ HFSC เพื่อระบุถึงชนิดทราฟฟิกที่แตกต่างกัน โดยแต่ละหมายเลขคลาสจะกำหนดเส้นโค้งการประกันการให้บริการให้ทำงานด้วยอัลกอลิทึมหลักการของเรียลไทม์ (rt) สำหรับทำการรับประกันการให้บริการ นั่นคือรับประกันแบนด์วิดท์ขั้นต่ำและค่าความหน่วงในคลาสฯ นั้น และ ทำงานด้วยอัลกอลิทึมหลักการของการเชื่อมต่อ (ls) เพื่อสนับสนุนการกระจายแบนด์วิดท์ระหว่างคลาสอย่างยุติธรรม

นอกจากนี้ยังประกาศเส้นโค้งการประกันบริการ (ul) ให้คลาสฯ นั้น เพื่อแสดงถึงอัตราเร็วในการส่งข้อมูลสูงสุดของคลาสฯ นั้นที่จะได้รับ

4. คำสั่งกำหนดข้อกำหนดของแถวคอยสำหรับเป็นคลาสสุดท้ายให้กับคลาสของแต่ละบริการ

```
tc qdisc add dev eth2 handle 11: parent 1:11 fifo
tc qdisc add dev eth2 handle 12: parent 1:12 fifo
tc qdisc add dev eth2 handle 13: parent 1:13 fifo
tc qdisc add dev eth2 handle 14: parent 1:14 fifo
tc qdisc add dev eth2 handle 15: parent 1:15 fifo
tc qdisc add dev eth2 handle 16: parent 1:16 fifo
```

ภาพที่ 2.31 แสดงตัวอย่างคำสั่งการสร้างข้อกำหนดของแถวคอยให้กับคลาสสุดท้ายของ HSFC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. คำสั่งทำการสร้างตัวกรองแพ็คเก็ตให้เหมาะกับในแต่ละคลาสสุดท้าย

```
iptables -t mangle -A POSTROUTING -p udp --sport 4444 -j CLASSIFY --set-class 1:11
```

ภาพที่ 2.32 แสดงตัวอย่างคำสั่งการสร้างตัวกรองแพ็คเก็ต เพื่อทำการกรองแพ็คเก็ต UDP ที่มีพอร์ตเท่ากับ 4444

จากคำสั่งสร้างตัวกรองแพ็คเก็ต เพื่อทำการกรองแพ็คเก็ต UDP ที่มีพอร์ตเท่ากับ 4444 ซึ่งค่า Flowid จะเป็นตัวกำหนดว่าแพ็คเก็ตที่ตรงกับเงื่อนไขใด ก็จะถูกไหลเข้าไปทำงานต่อที่ Flowid ของ Classid 1:11

6. คำสั่งดูข้อมูลการทำงานของข้อกำหนดแถวคอย

```
tc –s –d qdisc show
```

ภาพที่ 2.33 แสดงตัวอย่างคำสั่งดูข้อมูลการทำงานของข้อกำหนดแถวคอย

7. คำสั่งดูข้อมูลการทำงานของคลาส

```
tc -s -d class ls dev eth0
```

ภาพที่ 2.34 แสดงตัวอย่างคำสั่งดูข้อมูลการทำงานของคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

แนวคิดและหลักการจัดการปริมาณทราฟฟิก ของเรียลไทม์แอพพลิเคชั่น

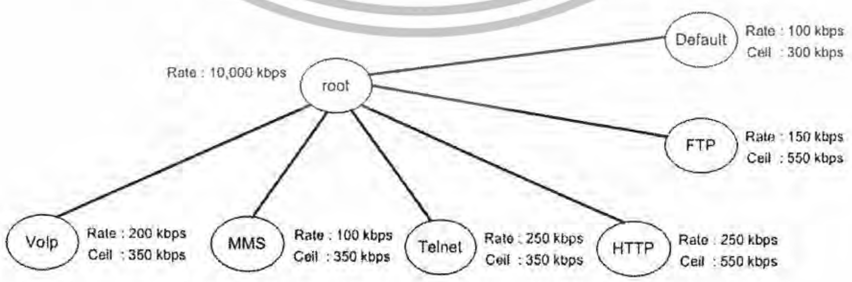
ปัจจุบันระบบเครือข่ายอินเตอร์เน็ตมีการเน้นการให้บริการที่เกี่ยวข้องกับสื่อประสมมากขึ้น ทำให้แอพพลิเคชั่นบางประเภทที่เน้นการตอบสนองในทันที หรือ ประเภทที่ต้องการการรับประกันความต่อเนื่องในการรับส่งข้อมูลมีจำนวนเพิ่มขึ้นอย่างรวดเร็ว จึงจำเป็นต้องมีแนวคิดและหลักในการจัดการปริมาณทราฟฟิกของเรียลไทม์แอพพลิเคชั่นเพื่อไม่ให้งานในลักษณะที่มีความสำคัญมาก หรือ งานที่ต้องการการตอบสนองในทันที เช่น เอชทีทีพี (HTTP) หรือเครือข่ายโทรศัพท์อินเตอร์เน็ต (VoIP) ถูกแอพพลิเคชั่นอย่างเพียร์ทูเพียร์มาเบียดบังการใช้แบนด์วิดท์

3.1 เปรียบเทียบประสิทธิภาพระหว่างอัลกอลิทึม HTB และ HFSC

3.1.1 จุดประสงค์การเปรียบเทียบประสิทธิภาพ

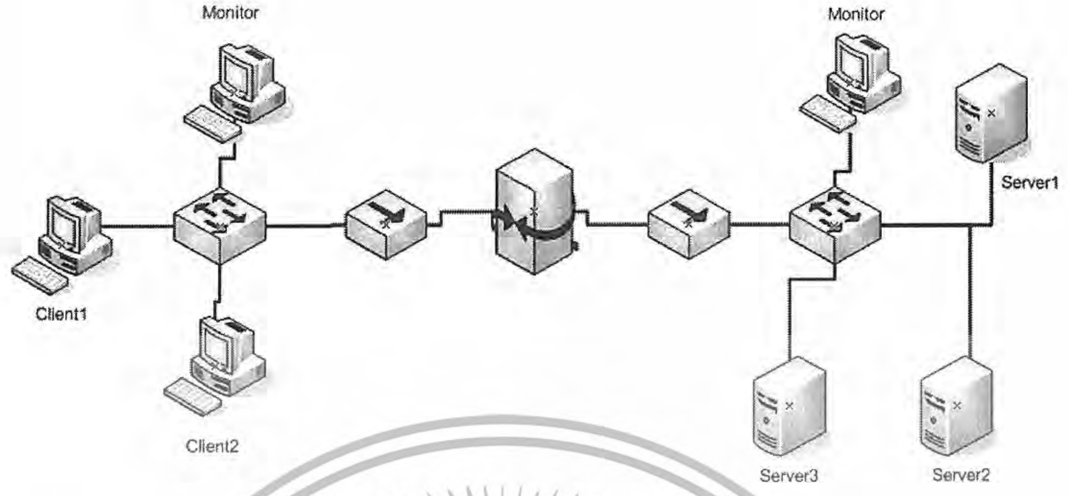
- เปรียบเทียบความสามารถในการจัดสรรแบนด์วิดท์ในแต่ละอัลกอลิทึม
- เปรียบเทียบความสามารถในการจัดการค่าความหน่วงในแต่ละอัลกอลิทึม
- เปรียบเทียบความสามารถในการจัดการทราฟฟิกแบบเบิร์สทั้งชนิดข้อมูลและเรียลไทม์

3.1.2 ออกแบบการทดลอง



ภาพที่ 3.1 กฎนโยบายควบคุมปริมาณทราฟฟิกในการทดสอบเปรียบเทียบประสิทธิภาพระหว่างอัลกอลิทึม HTB และ HFSC

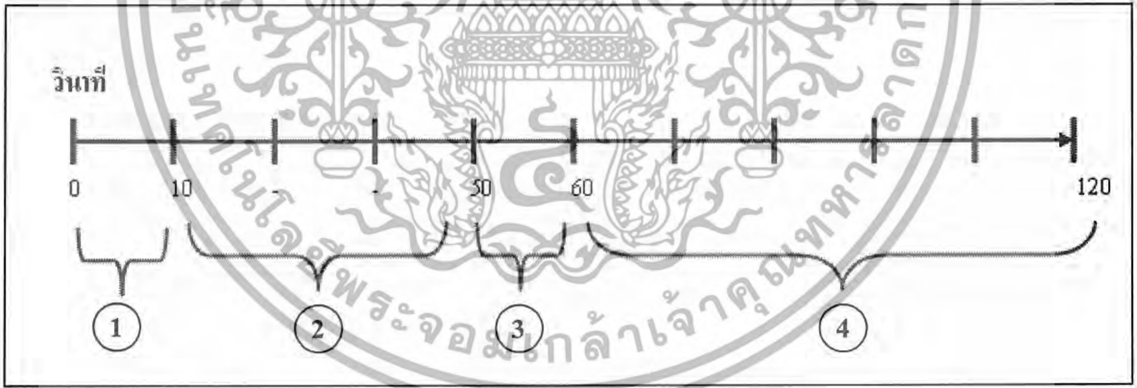
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.2 แผนภาพจำลองเครือข่ายของการทดสอบเปรียบเทียบประสิทธิภาพระหว่างอัลกอลิทึม HTB และ HFSC

3.1.3 ขั้นตอนการทดสอบเว็บแอพพลิเคชั่นตามกฎควบคุมปริมาณทราฟฟิก

3.1.3.1 ทดสอบความสามารถในการจัดการทราฟฟิกในสภาวะปกติ



ภาพที่ 3.3 แสดงขั้นตอนทดสอบความสามารถในการจัดการทราฟฟิกในสภาวะปกติ

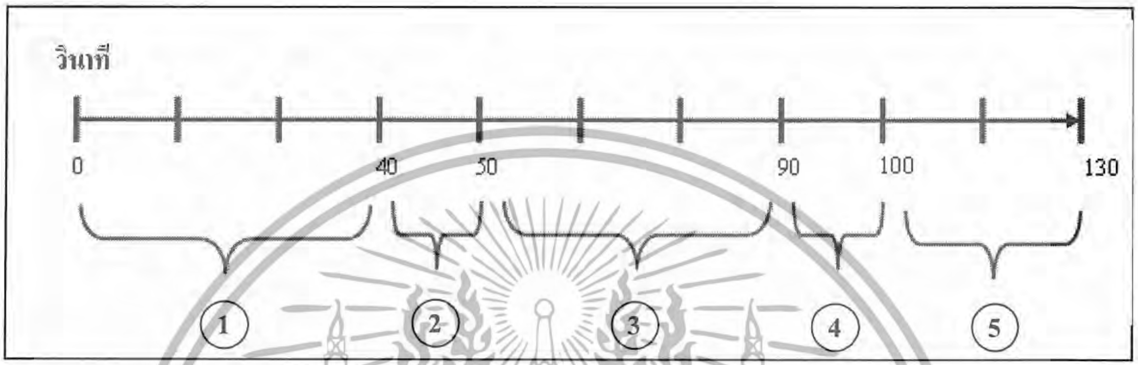
จากภาพ 3.3 แสดงขั้นตอนการทดสอบดังนี้

1. (วินาทีที่ 0-10) ทำการเปิดแอพพลิเคชั่นต่างในฝั่งไคลเอนท์ โดยที่เครื่องลินุกซ์เกตเวย์ที่ทำหน้าที่ควบคุมปริมาณทราฟฟิกไม่มีการใช้ชุดคำสั่งควบคุมปริมาณทราฟฟิก
2. (วินาทีที่ 11-50) เมื่อค่าทรูพุตของทุกแอพพลิเคชั่นที่ทั้งฝั่งเครื่องเซิร์ฟเวอร์และเครื่องไคลเอนท์ มีค่าคงที่ ให้สังเกตค่าทรูพุตของแต่ละแอพพลิเคชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. (วินาทีที่ 51-60) ทำการใช้ชุดคำสั่งควบคุมปริมาณทราฟฟิกที่เครื่องลินุกซ์เกตเวย์
4. (วินาทีที่ 61 – วินาทีที่ 120) สังเกตุการเปลี่ยนแปลงของค่าทรูพุตของแต่ละแอพพลิเคชั่นที่ฝั่งเครื่องไคลเอนท์ว่ามีการเปลี่ยนแปลงอย่างไรและเป็นไปตามที่รับประกันไว้หรือไม่

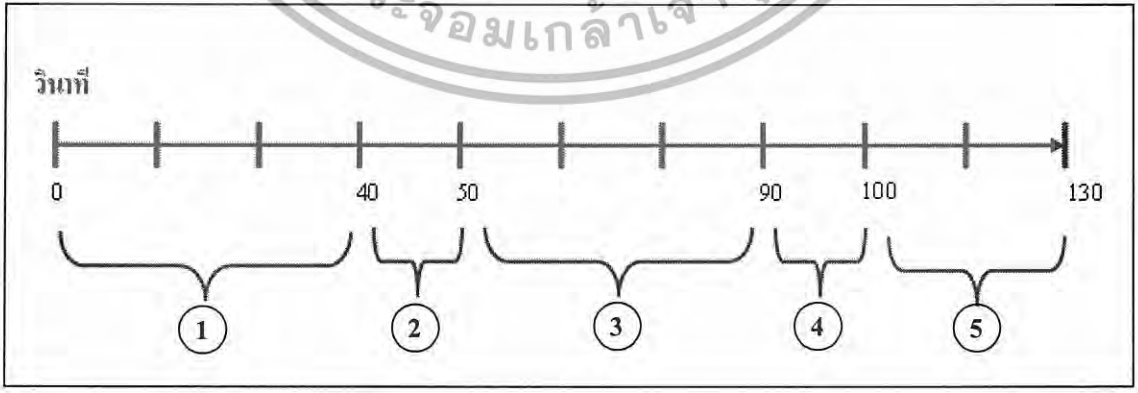
3.1.3.2 ทดสอบความสามารถในการจัดการทราฟฟิกเฉพาะชนิดข้อมูลและเรียลไทม์



ภาพที่ 3.4 แสดงขั้นตอนทดสอบความสามารถในการจัดการทราฟฟิกเฉพาะชนิดข้อมูล

จากภาพ 3.4 แสดงขั้นตอนการทดสอบดังนี้

1. (วินาทีที่ 0-30) สร้างบริการประเภท HTTP ไปยังฝั่งเครื่องไคลเอนท์
2. (วินาทีที่ 31-40) เปิดบริการแอพพลิเคชั่น FTP เพื่อดึงการใช้แบนด์วิดท์จากบริการ HTTP
3. (วินาทีที่ 41-70) สังเกตค่าทรูพุตที่เปลี่ยนไปของบริการ HTTP
4. (วินาทีที่ 71-80) ปิดบริการแอพพลิเคชั่น FTP
5. (วินาทีที่ 81-100) สังเกตค่าทรูพุตที่เปลี่ยนไปของบริการ HTTP หลังจากปิดบริการแอพพลิเคชั่น FTP

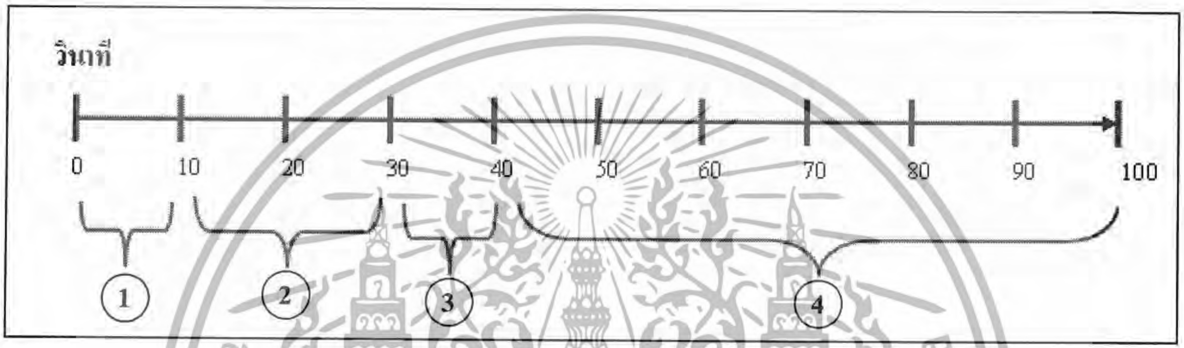


ภาพที่ 3.5 แสดงขั้นตอนทดสอบความสามารถในการจัดการทราฟฟิกเฉพาะชนิดเรียลไทม์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพ 3.5 แสดงขั้นตอนการทดสอบดังนี้

1. (วินาทีที่ 0-30) สร้างบริการประเภท VoIP เข้าไปยังฝั่งเครื่องไคลเอนท์
2. (วินาทีที่ 31-40) เปิดบริการแอพพลิเคชั่น MMS เพื่อดึงการใช้แบนด์วิดท์บริการ VoIP
3. (วินาทีที่ 41-70) สังเกตค่าทรูพุตที่เปลี่ยนไปของบริการ VoIP
4. (วินาทีที่ 71-80) ปิดบริการแอพพลิเคชั่น MMS
5. (วินาทีที่ 81-100) สังเกตค่าทรูพุตที่เปลี่ยนไปของบริการ VoIP หลังจากปิดบริการแอพพลิเคชั่น MMS



ภาพที่ 3.6 แสดงขั้นตอนทดสอบความสามารถในการจัดการทราฟฟิกในสภาวะเบิร์ส ชนิดข้อมูลและชนิดเรียลไทม์

3.1.3.3 ทดสอบความสามารถในการจัดการทราฟฟิกในสภาวะเบิร์สชนิดข้อมูล

จากภาพ 3.6 แสดงขั้นตอนการทดสอบดังนี้

1. (วินาทีที่ 0-10) ทำการเปิดแอพพลิเคชั่นต่างๆในฝั่งเครื่องไคลเอนท์
2. (วินาทีที่ 11-30) เมื่อค่าทรูพุตของแต่ละแอพพลิเคชั่นมีค่าคงที่ สังเกตว่าค่าทรูพุตเป็นไปตามที่รับประกันไว้หรือไม่
3. (วินาทีที่ 31-40) ใช้โปรแกรมสร้างทราฟฟิกสร้างบริการประเภท HTTP ที่มีความหนาแน่นส่งไปยังฝั่งไคลเอนท์ทุกเครื่อง
4. (วินาทีที่ 41-100) สังเกตการเปลี่ยนแปลงของค่าทรูพุตของแต่ละแอพพลิเคชั่นที่ฝั่งเครื่องไคลเอนท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3.4 ทดสอบความสามารถในการจัดการทราฟฟิกในสภาวะเบิร์สชนิดเรียลไทม์

จากภาพ 3.6 แสดงขั้นตอนการทดสอบดังนี้

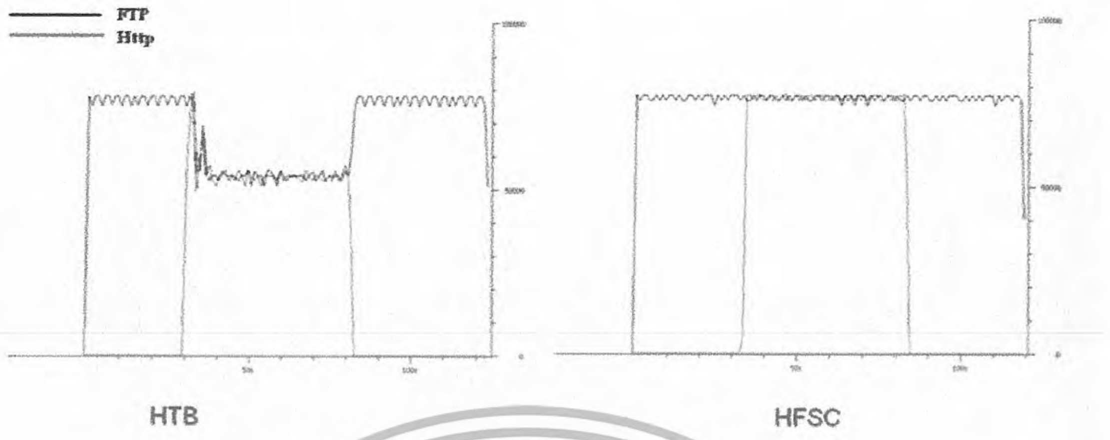
1. (วินาทีที่ 0-10) ทำการเปิดแอพพลิเคชั่นต่างๆในฝั่งเครื่องไคลเอนท์
2. (วินาทีที่ 11-30) เมื่อค่าทรูพุตของแต่ละแอพพลิเคชั่นมีค่าคงที่ สังเกตว่าค่าทรูพุตเป็นไปตามที่รับประกันไว้หรือไม่
3. (วินาทีที่ 31-40) ใช้โปรแกรมสร้างทราฟฟิก สร้างบริการประเภท MMS ที่มีความหนาแน่นส่งไปยังฝั่งเครื่องไคลเอนท์ทุกเครื่อง
4. (วินาทีที่ 41-100) สังเกตการเปลี่ยนแปลงของค่าทรูพุตของแต่ละแอพพลิเคชั่นที่ฝั่งเครื่องไคลเอนท์

3.1.4 ผลการทดสอบเปรียบเทียบประสิทธิภาพระหว่างอัลกอลิทึม HTB และ HFSC

3.1.4.1 การจัดสรรแบนด์วิดท์

ภาพที่ 3.7 แสดงความสามารถในการจัดสรรแบนด์วิดท์ในสภาวะเครือข่ายปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

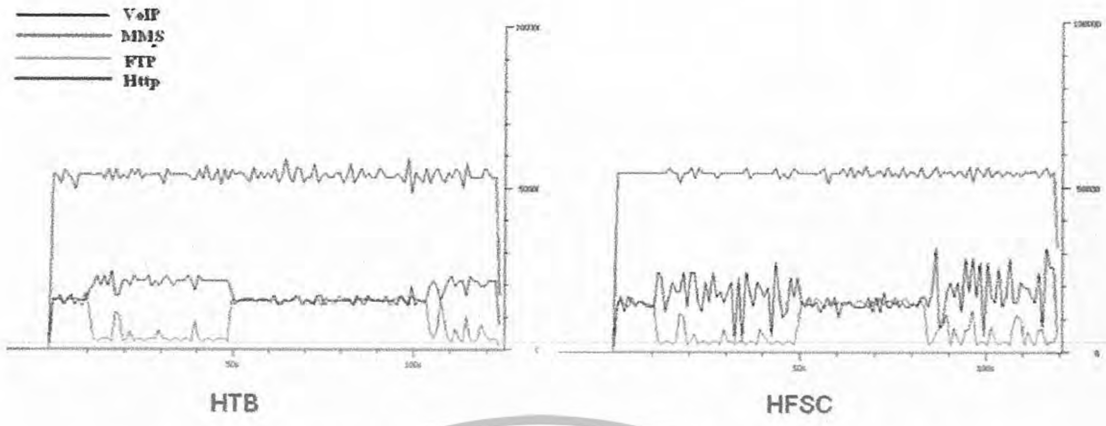


ภาพที่ 3.8 แสดงความสามารถในการจัดสรรแบนด์วิดท์ในสภาวะที่เครือข่ายมีเฉพาะทราฟฟิกชนิดข้อมูลเท่านั้น

ภาพที่ 3.9 แสดงความสามารถในการจัดสรรแบนด์วิดท์ในสภาวะที่เครือข่ายมีเฉพาะทราฟฟิกชนิดเรียลไทม์เท่านั้น

จากภาพที่ 3.7 ถึง 3.9 แสดงการเปรียบเทียบความสามารถในการจัดสรรแบนด์วิดท์ระหว่างอัลกอลิทึม HTB และ HFSC โดยจากภาพที่ 3.7 ถึง 3.10 แสดงให้เห็นว่า HFSC สามารถจัดสรรแบนด์วิดท์ได้ดีกว่า HTB ทั้งในสภาวะทราฟฟิกปกติ สภาวะที่เครือข่ายมีเฉพาะทราฟฟิกชนิดข้อมูล และสภาวะที่เครือข่ายมีเฉพาะทราฟฟิกชนิดข้อมูลเรียลไทม์เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.10 แสดงความสามารถในการจัดสรรแบนด์วิดท์ในสภาวะเบิร์สชนิดข้อมูล

ภาพที่ 3.11 แสดงความสามารถในการจัดสรรแบนด์วิดท์ในสภาวะเบิร์สเฉพาะชนิดเรียลไทม์

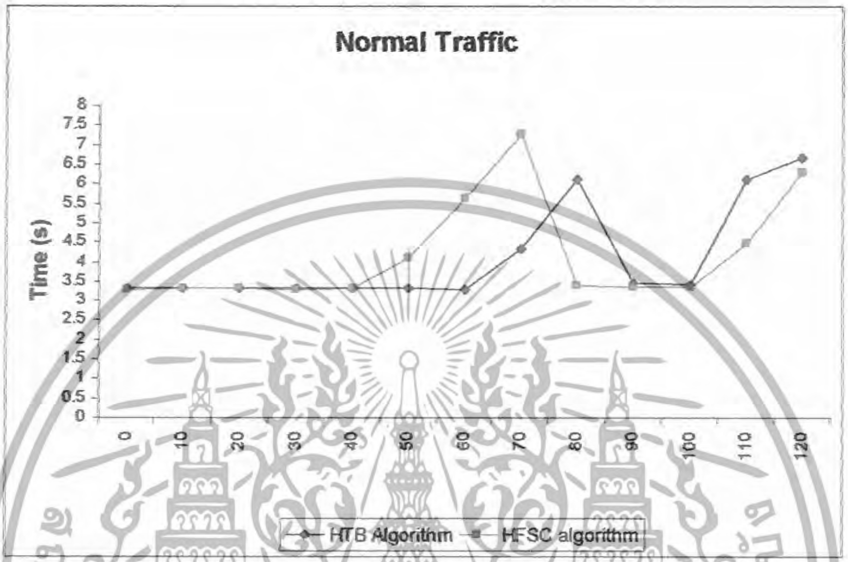
จากภาพที่ 3.10 แสดงความสามารถในการจัดสรรแบนด์วิดท์ในสภาวะเบิร์สชนิดข้อมูล ซึ่งเนื่องจากสภาวะที่มีทราฟฟิกชนิดข้อมูลอย่างหนาแน่น มีผลต่อการทำงานของคลาสบริการเรียลไทม์น้อยมาก เนื่องจากมีการจัดสรรให้การควบคุมแบนด์วิดท์ของคลาสบริการข้อมูลและบริการเรียลไทม์อยู่คนละคลาสกัน ดังนั้นทราฟฟิกชนิดข้อมูลจึงส่งผลกระทบต่อการทำงานของคลาสบริการเรียลไทม์น้อยมาก ทำให้ HFSC สามารถจัดการทราฟฟิกสภาวะเบิร์สชนิดข้อมูลได้ดีกว่า HTB

จากภาพที่ 3.11 แสดงการจัดการทราฟฟิกสภาวะเบิร์สชนิดเรียลไทม์ เห็นได้ชัดว่า เมื่อทราฟฟิกมีความหนาแน่นมากขึ้น HFSC ไม่สามารถจัดการทราฟฟิกได้ดี เมื่อเทียบกับ HTB พิจารณาได้จากค่าทรูพุตจากกราฟของ HFSC คือ เมื่อมีการสร้างทราฟฟิกแบบเบิร์สชนิดเรียลไทม์ หรือ MMS จะส่งผลโดยตรงต่อการจัดสรรแบนด์วิดท์ของคลาสบริการ MMS ในแต่ละอัลกอลิทึม โดยจากกราฟ

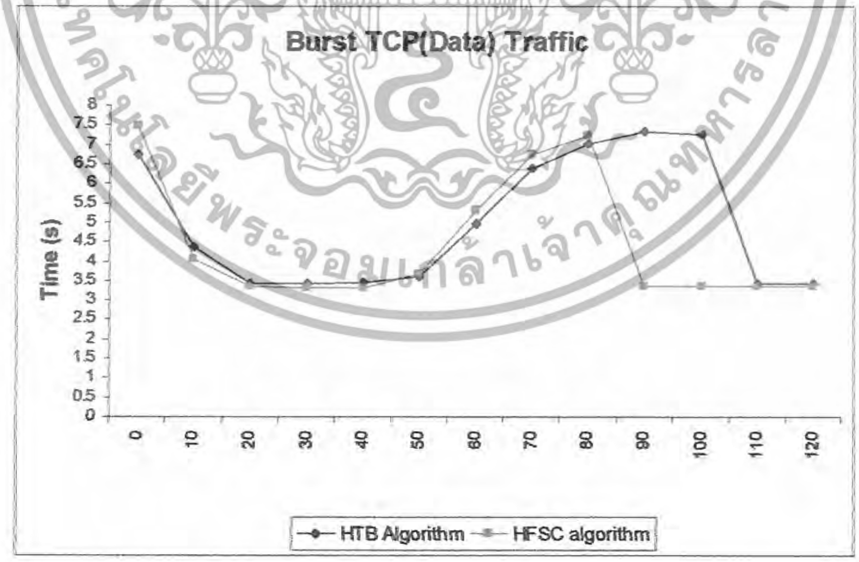
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงให้เห็นว่า HTB สามารถจัดการได้ดีกว่าและยังคงเป็นไปตามค่าแบนด์วิดท์ที่ได้รับประกันไว้เมื่อเทียบกับ HFSC ซึ่งการจัดสรรแบนด์วิดท์ลดต่ำลงมากเมื่อเทียบกับ HTB

3.1.4.2 การจัดการค่าความหน่วง (Delay)

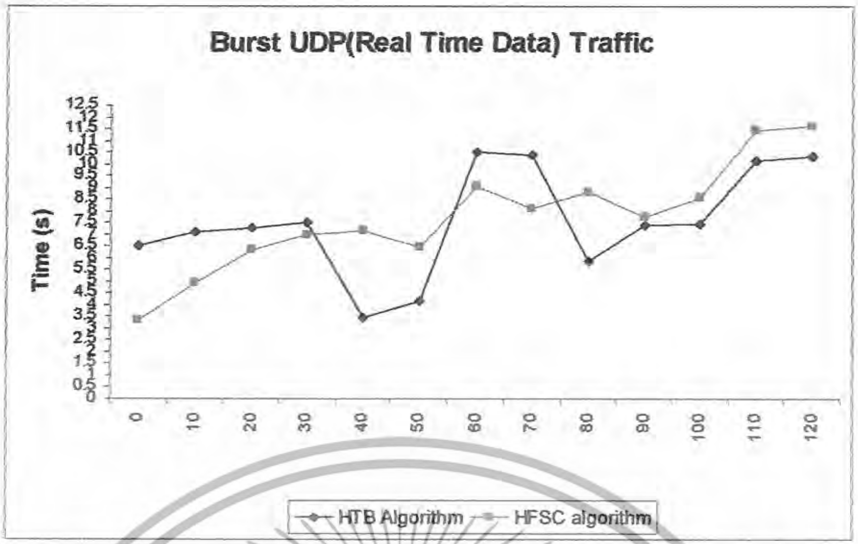


ภาพที่ 3.12 แสดงความสามารถในการจัดการค่าความหน่วงในสภาวะเครือข่ายปกติ



ภาพที่ 3.13 แสดงความสามารถในการจัดการค่าความหน่วงในสภาวะเบิร์สชนิดข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.14 แสดงความสามารถในการจัดการค่าความหน่วงในสภาวะเบิร์สชนิดเรียลไทม์

จากภาพที่ 3.12 แสดงความสามารถในการจัดการค่าความหน่วงของคลาสบริการเรียลไทม์ในสภาวะปกติสังเกตได้ว่า HFSC จะมีค่าความหน่วงสูงกว่า HTB เนื่องมาจากโครงสร้างการทำงานที่ซับซ้อนมากกว่าเมื่อเทียบกับ HTB แต่เมื่อพิจารณาโดยรวมแล้วจะเห็นได้ว่า HFSC มีการจัดการค่าความหน่วงให้มีค่าที่คงที่ได้เร็วกว่า HTB และแนวโน้มค่าความหน่วงที่ต่ำกว่า HTB

จากภาพที่ 3.13 แสดงความสามารถในการจัดการค่าความหน่วงของคลาสบริการเรียลไทม์ในสภาวะทราฟฟิกหนาแน่นชนิดข้อมูล สังเกตได้ว่า HFSC จะใช้เวลาในการจัดการค่าความหน่วงให้คงที่ได้เร็วกว่า HTB เนื่องจากสภาวะเบิร์สชนิดข้อมูลจะมีผลกระทบต่อการทำงานของคลาสบริการเรียลไทม์น้อยมาก ตามหลักการ จัดสรรแบนด์วิดท์ให้คลาสของทราฟฟิกเรียลไทม์และข้อมูลจะอยู่คนละคลาสแยกกันชัดเจนและการส่งของทราฟฟิกข้อมูลจะอยู่ในรูปแบบ TCP ซึ่งจะมีกลไกการควบคุมอัตราการส่งข้อมูล (Sliding Window) ทำให้ต่อให้ปริมาณทราฟฟิกแบบเบิร์สมากแค่ไหนจะสามารถส่งได้มากที่สุดเท่ากับค่าที่ได้รับประกันแบนด์วิดท์ให้คลาสฯ นั้นเท่านั้น

จากภาพที่ 3.14 แสดงความสามารถในการจัดการค่าความหน่วงของคลาสบริการเรียลไทม์ในสภาวะเบิร์สชนิดเรียลไทม์ สังเกตได้ว่าในช่วงแรก HFSC สามารถจัดการค่าความหน่วงได้ดีกว่า HTB แต่หลังจากที่มีการสร้างทราฟฟิกแบบเบิร์สชนิดเรียลไทม์ เห็นได้ชัดว่า HFSC จะจัดการค่าความหน่วงได้ดีแค่ช่วงเวลาสั้นๆ ในระยะแรกและต่อมาจะจัดการค่าความหน่วงได้ลดลงเมื่อเทียบกับ HTB ที่สามารถจัดการได้ดีกว่า เนื่องจากสภาวะเบิร์สชนิดเรียลไทม์จะมีผลกระทบต่อการทำงานของคลาสบริการเรียลไทม์มาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.5 สรุปผลทดสอบเปรียบเทียบประสิทธิภาพระหว่างอัลกอลิทึม HTB และ HFSC

- HFSC สามารถจัดการทราฟฟิกในสภาวะปกติได้ดีกว่า HTB ทั้งในด้านการจัดสรรแบนด์วิดท์และค่าความหน่วง
- HFSC สามารถจัดการทราฟฟิกแบบเบิร์สชนิดข้อมูลได้ดีกว่า HTB ทั้งในด้านการจัดสรรแบนด์วิดท์และค่าความหน่วง
- HTB สามารถจัดการทราฟฟิกแบบเบิร์สชนิดเรียลไทม์ได้ดีกว่า HFSC ทั้งในด้านการจัดสรรแบนด์วิดท์และค่าความหน่วง
- HFSC จะจัดการได้ดีในสถานการณ์ปกติ แต่เมื่อไหร่ที่ทราฟฟิกมีความหนาแน่นมากขึ้น HTB จะจัดการได้ดีกว่า เนื่องจากโครงสร้างการทำงานที่ซับซ้อนน้อยกว่าเมื่อเทียบกับ HFSC

3.2 แนวคิดการทำงานสลับอัลกอลิทึมโดยอัตโนมัติ (Automatic Swapping Algorithm)

จากการศึกษาและเปรียบเทียบประสิทธิภาพระหว่างอัลกอลิทึม HTB และ HFSC ทำให้เห็นถึงจุดเด่นของแต่ละอัลกอลิทึมในการจัดการควบคุมปริมาณทราฟฟิกในสถานการณ์ที่แตกต่างกัน ซึ่งวัตถุประสงค์หลักของโครงงาน คือ การควบคุมปริมาณทราฟฟิกของเรียลไทม์แอพพลิเคชั่น ดังนั้นจะอาศัยการสลับอัลกอลิทึมโดยอัตโนมัติในการควบคุมคุณภาพการให้บริการและปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์ให้เหมาะสมกับสถานการณ์ต่างๆ

อัลกอลิทึมที่นำมาใช้ควบคุม คือ อัลกอลิทึม HTB (Hierarchical Token Bucket) ซึ่งสามารถรับประกันค่าแบนด์วิดท์ต่ำสุดและแบนด์วิดท์สูงสุดตามที่กำหนด อีกทั้งมีความสามารถในการจัดการทราฟฟิกที่มีความหนาแน่นมากๆ หรือทราฟฟิกแบบเบิร์ส และอัลกอลิทึม HFSC (Hierarchical Fair Service Curve) ซึ่งเป็นเทคนิคการควบคุมและจัดสรรปริมาณทราฟฟิกข้อมูลบนเครือข่ายเช่นเดียวกับ HTB แต่สามารถรับประกันค่าแบนด์วิดท์และค่าความหน่วงของทราฟฟิกได้ ซึ่งสองตัวแปรนี้เป็นตัวแปรสำคัญในการควบคุมคุณภาพการให้บริการ (QoS) ของทราฟฟิกแบบเรียลไทม์ สามารถจัดการทราฟฟิกในสภาวะปกติ สภาวะทราฟฟิกที่มีเฉพาะข้อมูล เฉพาะเรียลไทม์ และสภาวะเบิร์สชนิดข้อมูลได้

โดยการที่จะสลับอัลกอลิทึมโดยอัตโนมัติจำเป็นต้องมีค่า Threshold บ่งบอกถึงจุดที่การทำงานต้องสลับอัลกอลิทึมเพื่อประสิทธิภาพการทำงานที่เพิ่มขึ้นและปรับปรุงคุณภาพการให้บริการต่อไป

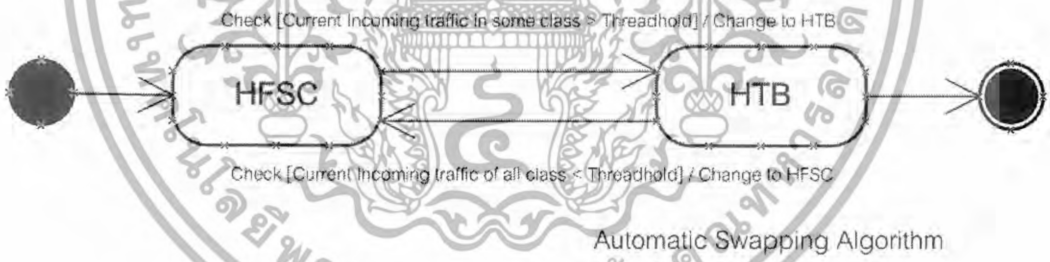
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 หลักการสลับอัลกอลิทึมแบบอัตโนมัติ

จากผลสรุปการเปรียบเทียบประสิทธิภาพระหว่าง HTB และ HFSC สามารถสรุปได้ว่า HFSC จะจัดการได้ดีในสถานการณ์ปกติ แต่เมื่อไหร่ที่ทราฟฟิกมีความหนาแน่นมากขึ้น HTB จะจัดการได้ดีกว่าเนื่องจากโครงสร้างการทำงานที่ซับซ้อนน้อยกว่าเมื่อเทียบกับ HFSC

หลักการสลับอัลกอลิทึมแบบอัตโนมัติจะมีการทำงาน คือ โดยในสภาวะปกติจะควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึม HFSC และเมื่อไหร่ที่ค่าทราฟฟิกขาเข้าคลาสใดคลาสหนึ่งในกฎนโยบายควบคุมปริมาณทราฟฟิกมีค่ามากกว่าหรือเท่ากับค่า Threshold นั่นหมายถึง ทราฟฟิกมีความหนาแน่นมากขึ้นในปริมาณที่ HFSC ไม่สามารถจัดการได้ดีเท่าเดิมจะทำการสลับอัลกอลิทึมไปเป็น HTB ซึ่งสามารถจัดการทราฟฟิกที่มีความหนาแน่นมากๆ ได้ดีกว่า

กรณีสลับอัลกอลิทึมกลับมาเป็น HFSC จะทำก็ต่อเมื่อทุกคลาสในกฎนโยบายควบคุมปริมาณทราฟฟิกมีค่าทราฟฟิกขาเข้าน้อยกว่าค่า Threshold เพื่อป้องกันปัญหากรณีที่คลาสใดคลาสหนึ่งเพิ่งเริ่มสลับอัลกอลิทึมเป็น HTB เพื่อจัดการทราฟฟิกที่หนาแน่น แต่ต้องสลับกลับมาเป็น HFSC เพราะคลาสอื่นมีค่าทราฟฟิกขาเข้าไม่เกินค่า Threshold เป็นผลให้ไม่สามารถแก้ไขปัญหาการจัดการควบคุมปริมาณทราฟฟิกได้



ภาพที่ 3.15 แสดงแนวคิดหลักการสลับอัลกอลิทึมอัตโนมัติ (Automatic Swapping Algorithm)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 หลักการจุดบ่งบอกการสลับอัลกอลิทึม (Threshold)

หลักการทำงานแนวคิดการสลับอัลกอลิทึมโดยอัตโนมัติจำเป็นต้องมีค่า Threshold บ่งบอกถึงจุดที่การทำงานต้องสลับอัลกอลิทึมที่มีความถูกต้องและเชื่อถือได้ เพื่อประสิทธิภาพการทำงานที่เพิ่มขึ้นและปรับปรุงคุณภาพการให้บริการ

3.3.1 ออกแบบการทดลอง

ในการทดลองเกี่ยวกับค่า Threshold นั้นมีการอ้างอิงผลที่ได้จากโครงงานที่ผ่านมาเกี่ยวกับประสิทธิภาพในการทำงานในสถานการณ์ต่างๆของทั้งสองอัลกอลิทึม เพื่อนำมาช่วยเป็นเกณฑ์พิจารณาว่าค่า Threshold ที่ใช้ในการสลับอัลกอลิทึมควรพิจารณาจากองค์ประกอบใด

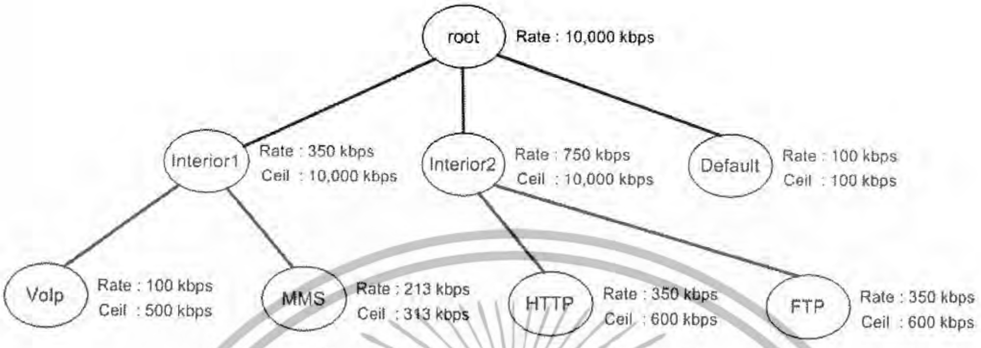
ในการทดลองจะพิจารณาถึงคุณสมบัติของทราฟฟิกประเภทเรียลไทม์เป็นหลัก ซึ่งทราฟฟิกประเภทดังกล่าวสามารถทำการรับและส่งข้อมูลได้ทั้งบนโปรโตคอล TCP และ UDP แต่ส่วนใหญ่นั้นทราฟฟิกเรียลไทม์มักจะใช้โปรโตคอล UDP ในการรับส่งข้อมูล เนื่องจากโปรโตคอล UDP มีการทำงานที่มีจุดเด่น คือ มีความเร็วในการรับส่งข้อมูล ข้อมูลที่ส่งมีขนาดเล็ก ไม่มีการทำงานเกี่ยวกับการส่งข้อมูลซ้ำ และไม่มีการคำนวณอัตราเร็วของการส่งข้อมูล ซึ่งหากข้อมูลมีการสูญหายบางส่วนหรือข้อมูลที่ส่งเกิดการหน่วง (Delay) จะถูกละความสนใจไป แต่หากใช้โปรโตคอล TCP จะมีเป้าหมายในการทำงานของโปรโตคอล คือ การเพิ่มระดับการส่งข้อมูลให้อยู่ในระดับสูงสุด โดยที่ยังมีความถูกต้องในข้อมูลนั้นๆ ซึ่งใช้เทคนิคที่เรียกว่า Slow Start คือ TCP จะทำการส่งข้อมูลจำนวนน้อยๆ และค่อยๆ เพิ่มปริมาณจนปลายทางแจ้งมาว่าแพ็คเก็ตมีการสูญหาย TCP จะถือว่าปริมาณการส่งข้อมูลที่สมบูรณ์ก่อนการส่งแพ็คเก็ตสูญหาย คือ ค่าการส่งข้อมูลสูงสุดและจะใช้ค่านั้นในการส่งข้อมูลต่อไป

การรับส่งข้อมูลที่ใช้ในการทดลองนี้ ใช้เทคนิคการรับ-ส่งข้อมูลจากเครื่องเซิร์ฟเวอร์บริการสารสื่อประสม (Streaming Media Server) ซึ่งเมื่อไฟล์มัลติมีเดียถูกเรียกใช้งานผ่านทางเว็บบราวเซอร์โดยผ่านโปรโตคอลที่ใช้ติดต่อระหว่างเครื่องเซิร์ฟเวอร์บริการสารสื่อประสมและเครื่องไคลเอนท์ เช่น โปรโตคอล Microsoft Media Server (MMS) เว็บบราวเซอร์จะมีการส่งไฟล์ขนาดเล็กที่เรียกว่า เมตะไฟล์ (Meta file) ไปยังโปรแกรมเล่นไฟล์มัลติมีเดียที่เครื่องไคลเอนท์ ซึ่งในเมตะไฟล์นี้ จะระบุปลายทางไปยังเครื่องเซิร์ฟเวอร์บริการสารสื่อประสม หลังจากนั้นโปรแกรมเล่นไฟล์มัลติมีเดียจะติดต่อกับเครื่องเซิร์ฟเวอร์บริการสารสื่อประสมโดยตรงโดยไม่ผ่านเว็บบราวเซอร์อีก ทำให้ผู้ฟังสามารถรับชมหรือฟังสื่อดังกล่าวได้ทันทีโดยมีบัฟเฟอร์เป็นตัวช่วย

จากข้อสรุปของผลการทดสอบเปรียบเทียบประสิทธิภาพระหว่างอัลกอลิทึม HTB และ HFSC ว่ามีปัจจัยใดที่มีผลต่อประสิทธิการทำงานในการควบคุมปริมาณทราฟฟิก แสดงให้เห็นว่าปริมาณทราฟฟิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในเครือข่ายมีผลต่อความสามารถในการทำงานของทั้งสองอัลกอลิทึม จึงนำมาเป็นตัวแปรที่ใช้ในการออกแบบการทดลองโดยมีการกำหนดนโยบายการควบคุมปริมาณทราฟฟิก ดังภาพที่ 3.16



ภาพที่ 3.16 แสดงแผนผังนโยบายการควบคุมปริมาณทราฟฟิกที่ใช้ในการทดลอง

3.3.2 การทดลองโดยการเปลี่ยนอัตราการส่งข้อมูลในเครือข่ายระหว่างโปรโตคอล UDP และ TCP

เป็นการทดลองที่ทำเพื่อหาว่าอัตราส่วนของทราฟฟิกประเภท UDP และ TCP ในเครือข่ายมีผลต่อความสามารถในการควบคุมปริมาณทราฟฟิกหรือไม่ เมื่อทราบแล้วว่าปริมาณทราฟฟิกในเครือข่ายนั้นมีผลต่อความสามารถในการจัดการคุณภาพการให้บริการบนเครือข่าย จึงได้ทำการทดลองเพื่อหาว่าอัตราส่วนระหว่างทราฟฟิกประเภท TCP และ UDP มีผลทำให้ความสามารถในการควบคุมปริมาณทราฟฟิกของแต่ละอัลกอลิทึมแตกต่างกันหรือไม่ เนื่องจากคุณสมบัติของทราฟฟิกแต่ละประเภทนั้นแตกต่างกัน เช่น โปรโตคอล TCP จะมีการส่งแพ็คเก็ตใหม่เมื่อเกิดการสูญหาย มีการตรวจสอบว่าแพ็คเก็ตที่ส่งออกไปนั้นปลายทางได้รับหรือไม่ แต่คุณสมบัติเหล่านี้ไม่ปรากฏในทราฟฟิกประเภท UDP ด้วยคุณสมบัติที่ต่างกันเหล่านี้ทำให้อัตราส่วนระหว่างทราฟฟิกทั้งสองชนิดน่าจะมีผลต่อความสามารถในการจัดการคุณภาพการให้บริการของแต่ละอัลกอลิทึมในการทดลอง จึงได้ทำการสร้างทราฟฟิกขนาดเท่ากับแบนด์วิดท์ คือ 10 เมกะบิตต่อวินาที โดยทราฟฟิกแบบ UDP จะถูกกรองเข้าคลาสเดียวกับ MMS ส่วนทราฟฟิกแบบ TCP จะถูกกรองเข้าคลาสที่เป็น HTTP ตามแผนผังนโยบายที่ผ่านมา ซึ่งจะมีการแบ่งการทดลองตามอัตรา TCP ต่อ UDP ดังตารางที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3-1 รูปแบบการทดลองอัตราส่วนระหว่างทราฟฟิกแบบ TCP และ UDP

	อัตรา TCP ของทราฟฟิกทั้งหมด (เปอร์เซ็นต์)	**อัตรา UDP ของทราฟฟิกทั้งหมด (เปอร์เซ็นต์)**
การทดลองที่ 1.1	100	0
การทดลองที่ 1.2	90	10
การทดลองที่ 1.3	80	20
การทดลองที่ 1.4	70	30
การทดลองที่ 1.5	60	40
การทดลองที่ 1.6	50	50
การทดลองที่ 1.7	40	60
การทดลองที่ 1.8	30	70
การทดลองที่ 1.9	20	80
การทดลองที่ 1.10	10	90
การทดลองที่ 1.11	0	100

จากการทดลองในตารางที่ 3-1 เมื่อทำการสังเกตการแสดงผลภาพและเสียงที่เครื่องปลายทางของทราฟฟิกประเภทเรียลไทม์ที่มีปริมาณทราฟฟิกเป็นดังค่าอัตราส่วนต่างๆตามตารางข้างต้นสามารถวิเคราะห์การทดลองได้ดังนี้

1. จากการสังเกตผลในการเล่นภาพและเสียงของโปรแกรมทางฝั่งไคลเอนท์พบว่าการแสดงผลของไฟล์มัลติมีเดียในการทดลองต่างๆ ไม่มีการเปลี่ยนแปลงมากนัก แม้ว่าจะมีการเปลี่ยนแปลงอัตราส่วนของปริมาณทราฟฟิกทั้งสองชนิด ซึ่งเกิดจากการที่อัตราของทราฟฟิกประเภท UDP และ TCP นั้นไม่มีผลต่อทราฟฟิกประเภทเรียลไทม์อย่างเด่นชัดและปริมาณข้อมูลนั้นมากเกินกว่าที่จะเห็นข้อแตกต่างในการควบคุมปริมาณทราฟฟิกได้ระหว่างการทดลอง

2. จากการสังเกตผลของกราฟทรูพุตโดยโปรแกรมอีเทอร์เรียล (Ethereal) ที่ฝั่งเครื่องเซิร์ฟเวอร์สังเกตได้ว่าไม่ว่าจะมีการสร้างทราฟฟิกจากต้นทางเพื่อส่งไปยังปลายทางมากเพียงใด ค่าทรูพุตที่ส่งได้จริงของโปรโตคอล TCP จะมีค่าไม่เกินค่าคงที่ค่าหนึ่งเสมอนั่นอาจเป็นเพราะกลไกการควบคุมปริมาณการส่งข้อมูล (Sliding Window) ของโปรโตคอล TCP ที่จะมีการเข้าจังหวะการส่ง(Synchronize) ระหว่างต้นทางและปลายทาง ทำให้ต้นทางและปลายทางส่งข้อมูลด้วยอัตราสูงสุดที่จะไม่ทำให้ข้อมูล

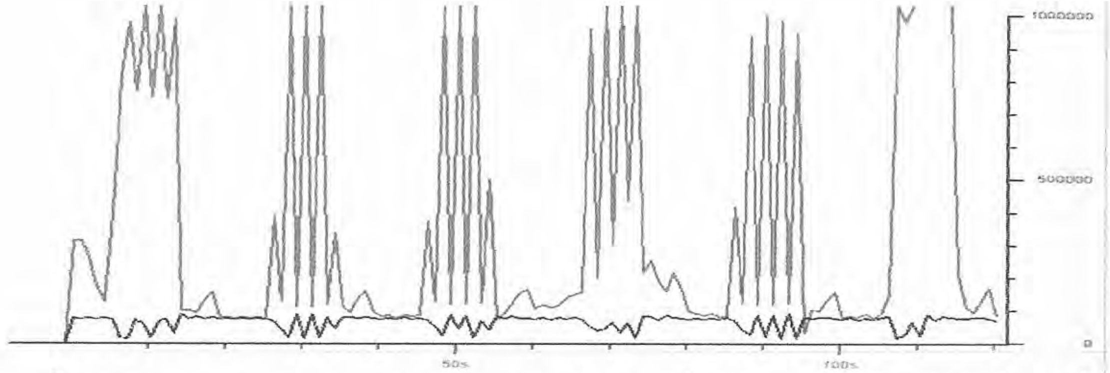
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เกิดการสูญหาย ซึ่งแตกต่างจากโปรโตคอล UDP ซึ่งไม่มีคุณสมบัติดังกล่าวทำให้ทรูพุตในการส่งข้อมูลของโปรโตคอล UDP เพิ่มมากขึ้นเมื่อต้นทางส่งข้อมูลด้วยอัตราที่เพิ่มขึ้น

ภาพที่ 3.17 กราฟแสดงผลทรูพุตจากโปรแกรมอีเทอร์เรียลทางฝั่งเซิร์ฟเวอร์ในการทดลองที่ 1.2 ที่มีการกำหนดอัตราส่วนของ TCP และ UDP เป็น 90 เปอร์เซ็นต์และ 10 เปอร์เซ็นต์ตามลำดับ

ภาพที่ 3.18 กราฟแสดงผลทรูพุตจากโปรแกรมอีเทอร์เรียลทางฝั่งเซิร์ฟเวอร์ในการทดลองที่ 1.7 ที่มีการกำหนดอัตราส่วนของ TCP และ UDP เป็น 40 เปอร์เซ็นต์และ 60 เปอร์เซ็นต์ตามลำดับ

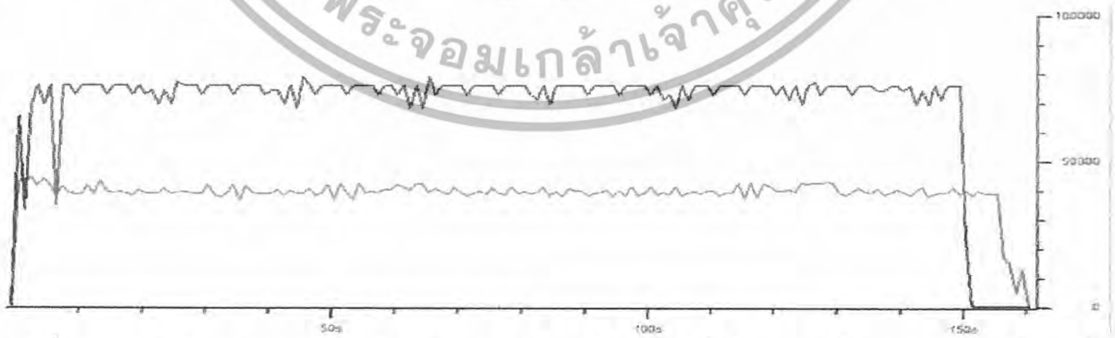
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.19 กราฟแสดงผลทรูพุตจากโปรแกรมอีเทอร์เรียลทางฝั่งเซิร์ฟเวอร์ในการทดลองที่ 1.10 ที่มีการกำหนดอัตราส่วนของ TCP และ UDP เป็น 10 เปอร์เซ็นต์และ 90 เปอร์เซ็นต์ตามลำดับ

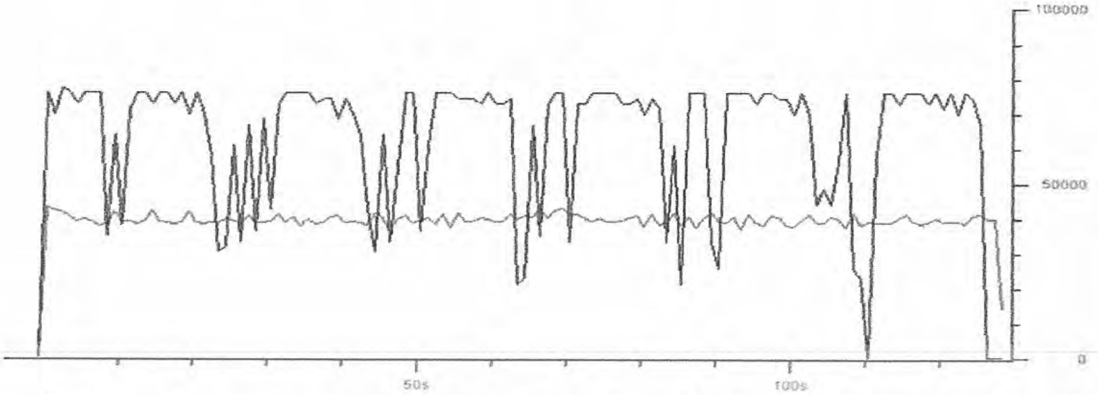
จากรูปที่ 3.18 ถึง 3.19 เป็นกราฟตัวอย่าง ของการทดลองที่ 1.2, 1.7 และ 1.10 ตามลำดับ ซึ่งแสดงถึงทรูพุตของ TCP และ UDP ทางฝั่งเซิร์ฟเวอร์ โดยกราฟเส้นสีแดง คือ ทรูพุตของโปรคตคอล UDP และ กราฟเส้นสีดำ คือ ทรูพุตของโปรคตคอล TCP สังเกตได้ว่าแม้จะมีการสร้างทราฟฟิกจากต้นทางเพื่อส่งไปยังปลายทางมากเพียงใดค่าทรูพุตที่ส่งได้จริงของโปรโตคอล TCP จะมีค่าประมาณ 80,000 ไบต์ต่อวินาที หรือ 625 กิโลบิตต่อวินาทีเสมอ ต่างจากโปรโตคอล UDP ที่มีทรูพุตเพิ่มขึ้นเมื่อ มีการสร้างทราฟฟิกที่ต้นทางเพิ่มขึ้น

3. จากการสังเกตผลของกราฟทรูพุตโดยโปรแกรมอีเทอร์เรียลที่ฝั่งเครื่องไคลเอนท์ สังเกตได้ว่าแม้จะมีการสร้างทราฟฟิกจากต้นทางเพื่อส่งไปยังปลายทางมากเพียงใด ค่าทรูพุตของโปรโตคอล TCP และ UDP ที่ฝั่งไคลเอนท์จะมีค่าไม่เกินค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้ ตามที่ได้กำหนดไว้ให้ทั้งสองโปรโตคอลแล้วข้างต้น

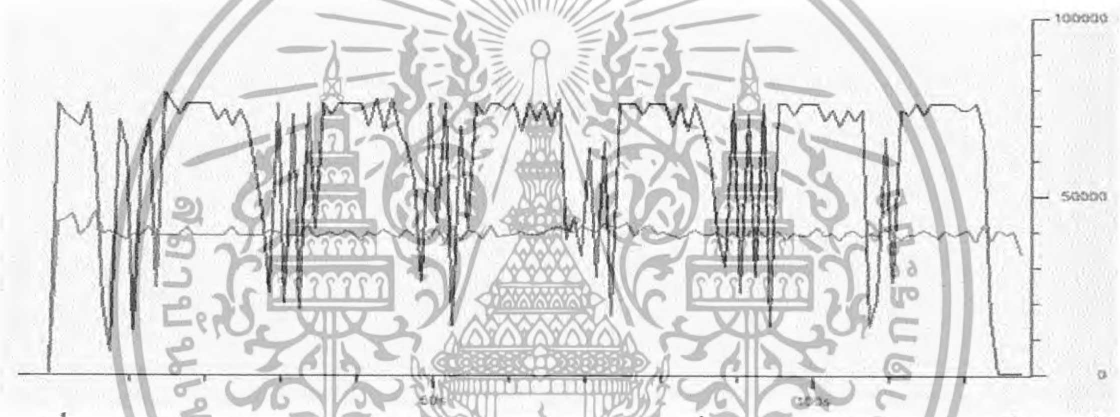


ภาพที่ 3.20 กราฟแสดงผลทรูพุตจากโปรแกรมอีเทอร์เรียลทางฝั่งไคลเอนท์ในการทดลองที่ 1.2 ที่มีการกำหนดอัตราส่วนของ TCP และ UDP เป็น 90 เปอร์เซ็นต์และ 10 เปอร์เซ็นต์ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.21 กราฟแสดงผลทรูพุตจากโปรแกรมอีเทอร์เรียลทางฝั่งไคลเอนท์ในการทดลองที่ 1.7 ที่มีการกำหนดอัตราส่วนของ TCP และ UDP เป็น 40 เปอร์เซ็นต์และ 60 เปอร์เซ็นต์ตามลำดับ



ภาพที่ 3.22 กราฟแสดงผลทรูพุตจากโปรแกรมอีเทอร์เรียลทางฝั่งไคลเอนท์ในการทดลองที่ 1.10 ที่มีการกำหนดอัตราส่วนของ TCP และ UDP เป็น 10 เปอร์เซ็นต์และ 90 เปอร์เซ็นต์ตามลำดับ

จากรูปที่ 3.20 ถึง 3.22 เป็นกราฟตัวอย่าง ของการทดลองที่ 1.2 ,1.7 และ 1.10 ตามลำดับ ซึ่งแสดงถึงทรูพุตของบริการ MMS และ HTTP ทางฝั่งไคลเอนท์ โดยกราฟเส้นสีแดงคือ ทรูพุตของคลาส MMS และ กราฟเส้นสีดำคือ ทรูพุตของบริการ HTTP โดยจะสังเกตได้ว่าแม้จะมีการสร้างทราฟฟิกจากต้นทางเพื่อส่งไปยังปลายทางมากเพียงใด ค่าทรูพุตที่ส่งได้จริงของโปรโตคอล TCP และ UDP จะมีค่าไม่เกินค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้ที่ได้กำหนดไว้ในข้างต้น ดังนั้นจะสังกตได้ว่าทรูพุตของ MMS จะมีค่าไม่เกิน 40,00 ไบต์ต่อวินาที หรือ 313 กิโลบิตต่อวินาที และ ทรูพุตของ HTTP จะมีค่าไม่เกิน 76,500 ไบต์ต่อวินาที หรือ 600 กิโลบิตต่อวินาที

จากการวิเคราะห์การทดลองดังข้างต้นสรุปได้ว่า อัตราส่วนของโปรโตคอล TCP และ UDP ในเครือข่ายมีผลกระทบต่อประสิทธิภาพของการเล่นสื่อมัลติมีเดียไม่มากนัก ซึ่งการเพิ่มขึ้นของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการวิเคราะห์การทดลองดังข้างต้นสรุปได้ว่า อัตราส่วนของโปรโตคอล TCP และ UDP ในเครือข่ายมีผลกระทบต่อประสิทธิภาพของการเล่นสื่อมัลติมีเดียไม่มากนัก ซึ่งการเพิ่มขึ้นของค่าทรูพุตของโปรโตคอล UDP มีผลกระทบต่อการเล่นไฟล์มัลติมีเดียมากกว่าการเพิ่มขึ้นของค่าทรูพุตของโปรโตคอล TCP เนื่องจากทราฟฟิกประเภท UDP เมื่อผ่านการควบคุมปริมาณทราฟฟิกจะเข้าคลาสเดียวกับทราฟฟิกของไฟล์มัลติมีเดียที่ใช้ในการทดลอง ดังนั้นการเพิ่มขึ้นของค่าทรูพุตของโปรโตคอล UDP มีผลกระทบต่อการเล่นไฟล์มัลติมีเดียมากกว่าการเพิ่มขึ้นของค่าทรูพุตทราฟฟิกประเภทโปรโตคอล TCP ที่อยู่ต่างคลาสกัน

3.3.3 การทดลองโดยใช้เทคนิคการเปลี่ยนแปลงค่าแบนด์วิดท์ที่รับประกันและค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้ แทนอัตราการเพิ่มขึ้นของปริมาณข้อมูลในคลาสของทราฟฟิกเรียลไทม์

จากการทดลองในข้างต้นทำให้ทราบว่าการเพิ่มขึ้นของทราฟฟิกที่เข้ามายังคลาสเดียวกับคลาสของไฟล์มัลติมีเดียน่าจะมีผลต่อการเล่นของไฟล์มัลติมีเดีย ดังนั้นในการทดลองนี้จึงใช้เทคนิคการเปลี่ยนแปลงค่าแบนด์วิดท์ที่รับประกันและค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้แทนอัตราการเพิ่มขึ้นของปริมาณข้อมูลในคลาสของทราฟฟิกเรียลไทม์ โดยในขั้นต้นกำหนดให้ค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้มีค่าเท่ากับ 313 กิโลบิตต่อวินาที ซึ่งเป็นค่าแบนด์วิดท์ปกติที่เครื่องเซิร์ฟเวอร์และไคลเอนท์ ใช้ในการรับและส่งไฟล์มัลติมีเดียและกำหนดให้ค่าแบนด์วิดท์ที่คลาสทำการยืมได้ในมีค่าเท่ากับ 100 กิโลบิตต่อวินาที ดังนั้นค่าแบนด์วิดท์ที่รับประกันจะมีค่าเท่ากับ 213 กิโลบิตต่อวินาที และทำการลดค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้ในการทดลองถัดไปให้มีค่าต่ำกว่าค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้ของการทดลองก่อนหน้า 10 เปอร์เซ็นต์ ส่วนทราฟฟิกทั้งหมดในการทดลองนี้จะมีขนาดเท่ากับ 313 กิโลบิตต่อวินาที ที่เกิดจากการเปิดโปรแกรมทางฝั่งไคลเอนท์เพียงอย่างเดียว อัตราในการกำหนดขนาดแบนด์วิดท์ที่รับประกันและค่าแบนด์วิดท์ที่มากที่สุดที่คลาส MMS สามารถใช้งานได้เป็นดังตารางที่ 3-2

จากการทดลองในตารางที่ 3-2 เมื่อทำการสังเกตการแสดงผลภาพและเสียงที่เครื่องปลายทางของทราฟฟิกประเภทเรียลไทม์ที่มีการควบคุมทราฟฟิกที่ลินุกซ์เกตเวย์และมีค่าการเปลี่ยนแปลงดังในตารางข้างต้นสามารถวิเคราะห์การทดลองได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3-2 รูปแบบการทดลองการเปลี่ยนแปลงอัตราระหว่างค่าแบนด์วิดท์ที่รับประกันและค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้

	ค่าแบนด์วิดท์ที่รับประกัน	**ค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้**
การทดลองที่ 2.1	313	213
การทดลองที่ 2.2	282	182
การทดลองที่ 2.3	254	154
การทดลองที่ 2.4	229	129
การทดลองที่ 2.5	205	105
การทดลองที่ 2.6	185	85
การทดลองที่ 2.7	166	66
การทดลองที่ 2.8	150	50
การทดลองที่ 2.9	135	35
การทดลองที่ 2.10	121	21
การทดลองที่ 2.11	110	10

ในการทดลองกรณีที่สภาวะของเครือข่ายมีเพียงทราฟฟิกของการรับและส่งไฟล์มัลติมีเดียเท่านั้น ในการทดลองที่ 2.7 ซึ่งมีการกำหนดให้ค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้ เท่ากับ 166 กิโลบิตต่อวินาที จากการสังเกตการเล่นภาพและเสียงของการเล่นไฟล์มัลติมีเดีย ปรากฏว่ามีความแตกต่างกับการทดลองก่อนหน้าอย่างชัดเจน คือ มีการค้างและกระตุก ทำให้เกิดการเล่นไฟล์ที่ไม่ติดต่อกันอย่างเห็นได้ชัด จากอาการดังกล่าวแสดงว่าน่าจะมีการสูญหายของแพ็คเก็ตเกิดขึ้น จึงได้มีการพิจารณาหาจำนวนแพ็คเก็ตที่สูญหายไปในแต่ละการทดลอง แล้วนำผลการเล่นไฟล์มัลติมีเดียที่ได้ไปพิจารณาควบคู่กับจำนวนการสูญหายของแพ็คเก็ตดังที่แสดงในตารางที่ 3-3 และภาพที่ 3.23 ที่แสดงกราฟเปอร์เซ็นต์การสูญหายของแพ็คเก็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3-3 ตารางแสดงการสูญหายของแพ็คเก็ตประเภทเรียลไทม์ เมื่อภายในเครือข่ายมีเพียงทราฟฟิกสื่อประสม

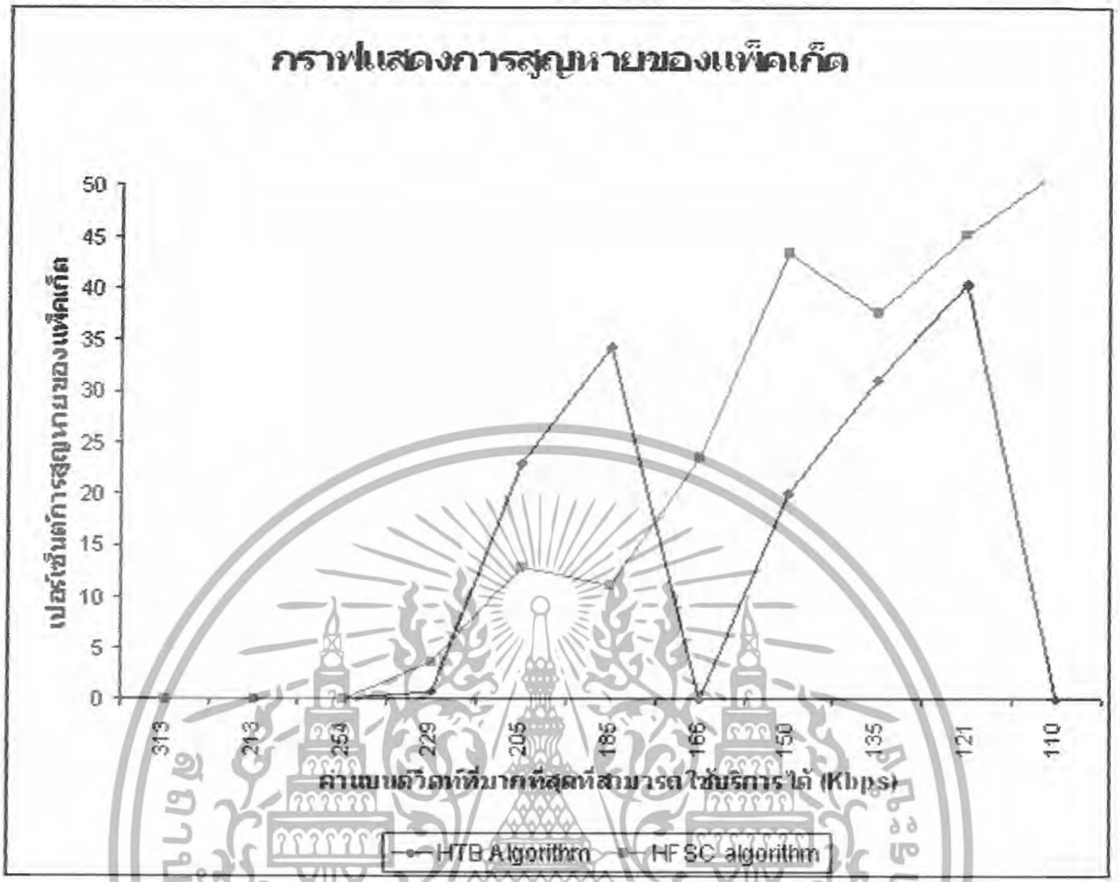
	ค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้	ค่าแบนด์วิดท์ที่รับประกัน	ประเภทของอัลกอลิทึม	จำนวนแพ็คเก็ตทั้งหมดที่มีการส่ง	จำนวนแพ็คเก็ตที่ส่งได้	จำนวนแพ็คเก็ตที่สูญหาย
การทดลองที่ 2.1	313	213	HTB	2132	2132	0
			HFSC	1275	2175	0
การทดลองที่ 2.2	282	182	HTB	2154	2154	0
			HFSC	2142	2142	0
การทดลองที่ 2.3	254	154	HTB	2200	2200	0
			HFSC	2198	2198	0
การทดลองที่ 2.4	229	129	HTB	2173	2173	0
			HFSC	2220	2220	0
การทดลองที่ 2.5	205	105	HTB	2303	2303	0
			HFSC	2194	2194	0
การทดลองที่ 2.6	185	85	HTB	1786	1753	33
			HFSC	2307	2307	0
การทดลองที่ 2.7	166	66	HTB	2306	2306	0
			HFSC	2133	1886	237
การทดลองที่ 2.8	150	50	HTB	2260	257	38
			HFSC	2219	331	82
การทดลองที่ 2.9	135	35	HTB	1712	1700	12
			HFSC	1955	1766	189
การทดลองที่ 2.10	121	21	HTB	1769	1558	211
			HFSC	1645	1474	171
การทดลองที่ 2.11	110	10	HTB	2626	1623	1003
			HFSC	2610	1630	980

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 3.23 แสดงกราฟเปอร์เซ็นต์การสูญหายของแพ็คเก็ตของอัลกอลิทึม HTB และ HFSC ในการทดลองในกรณีที่สภาวะของเครือข่ายมีเพียงทราฟฟิการรับและส่งไฟล์มัลติมีเดียเท่านั้น

จากภาพที่ 3.17 จะสังเกตได้ว่าช่วงที่ค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้มีค่าเท่ากับ 166 กิโลบิตต่อวินาทีนั้นเปอร์เซ็นต์การสูญหายของแพ็คเก็ตของ HFSC สูงกว่า HTB 15.52 เปอร์เซ็นต์ และในการทดลองต่อๆ ไปเปอร์เซ็นต์การสูญหายของแพ็คเก็ตของ HFSC มีค่ามากกว่า HTB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.24 แสดงกราฟเปอร์เซ็นต์การสูญหายของแพ็คเก็ตของอัลกอลิทึม HTB และ HFSC ในการทดลองกรณีที่สภาวะแวดล้อมที่เครือข่ายมีทราฟฟิคประเภทอื่นด้วย

ในการทดลองนี้ทำในสภาวะที่เครือข่ายมีทราฟฟิคประเภทอื่นๆด้วย ซึ่งเมื่อนำกราฟที่แสดงเปอร์เซ็นต์การสูญหายของแพ็คเก็ตของการทดลองนี้มาพิจารณาดังภาพที่ 3.24 พบว่าในช่วงที่ค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้มีค่าเท่ากับ 166 กิโลบิตต่อวินาทีนั้น เปอร์เซ็นต์การสูญหายของแพ็คเก็ตของอัลกอลิทึม HFSC มีค่าสูงกว่า HTB ที่ไม่มีการสูญหายถึง 22 เปอร์เซ็นต์ และในการทดลองต่อๆ ไปเปอร์เซ็นต์การสูญหายของแพ็คเก็ตของอัลกอลิทึม HFSC มีค่ามากกว่า HTB ทั้งที่การทดลอง 2.6 และ 2.5 นั้น เปอร์เซ็นต์การสูญหายของแพ็คเก็ตของอัลกอลิทึม HTB มีค่ามากกว่า HFSC มากอย่างเห็นได้ชัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 3.25 แสดงกราฟเปอร์เซ็นต์การสูญหายของแพ็คเก็ตของอัลกอลิทึม HTB และ HFSC ในการทดลองกรณีที่สภาวะที่ทุกคลาสมีทราฟฟิกอย่างหนาแน่นเข้าสู่คลาส

ในการทดลองนี้ คือ การทดลองในสภาวะที่เครือข่ายมีการสร้างทราฟฟิกประเภทอื่นๆ รวมทั้งทราฟฟิกชนิดเดียวกับไฟล์มัลติมีเดียเพิ่ม เพื่อจำลองสภาวะที่มีทราฟฟิกเข้าไปยังคลาสที่มีการกำหนดไว้ในข้างต้นต่างๆอย่างหนาแน่นและเมื่อนำผลของกราฟเปอร์เซ็นต์การสูญหายของแพ็คเก็ตมาพิจารณาดังภาพที่ 3.25 พบว่าตั้งแต่การทดลองที่ 2.7 ที่มีกำหนดค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้มีค่าเท่ากับ 166 กิโลบิตต่อวินาที จะเห็นว่าเปอร์เซ็นต์การสูญหายของแพ็คเก็ตของ HFSC สูงกว่า HTB อย่างเห็นได้ชัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.4 สรุปผลการทดลอง

ซึ่งเมื่อพิจารณาการทดลองข้างต้นที่ทำการทดสอบการควบคุมทราฟฟิกของทั้งสองอัลกอลิทึม โดยพิจารณาทั้งประสิทธิผลของการเล่นไฟล์มัลติมีเดียและตรวจสอบจำนวนการสูญหายของแพ็คเก็ตของทุกการทดลองในสภาวะต่างๆของเครือข่ายแล้ว ผลการทดลองชี้ให้เห็นว่า ในการทดลองที่กำหนดค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้มีค่าเท่ากับ 166 กิโลบิตต่อวินาทีของทุกสภาวะเครือข่าย เมื่อใช้เทคนิค HFSC นั้น ผลของการเล่นไฟล์มัลติมีเดียที่เครื่องไคลเอนท์ให้ผลการเล่นไฟล์ที่มีการเปลี่ยนแปลงจากการทดลองก่อนหน้าค่อนข้างเด่นชัด คือ ภาพมีการค้างและขาดหายไปเป็นจำนวนมากพอที่จะทำให้เกิดการไม่ประติดประต่อกันของเรื่องราวของไฟล์มัลติมีเดีย

จากการนำข้อมูลการสูญหายของแพ็คเก็ตในแต่ละการทดลองมาคิดเป็นเปอร์เซ็นต์และสร้างเป็นกราฟดังข้างต้น เมื่อนำมาเปรียบเทียบในทุกสภาวะของเครือข่ายที่ได้ทำการทดลองและพิจารณาที่การทดลองที่ 2.7 เป็นต้นไป พบว่าเมื่อใช้อัลกอลิทึม HFSC เปอร์เซ็นต์การสูญหายของแพ็คเก็ตมีค่าสูงกว่าการใช้อัลกอลิทึม HTB อย่างเห็นได้ชัด โดยจากข้างต้นที่กล่าวไว้ว่า ใช้เทคนิคการทดลองโดยการลดค่าแบนด์วิดท์ที่รับประกันและค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้นี้แทนอัตราการเพิ่มขึ้นของปริมาณข้อมูลในคลาสของทราฟฟิกเรียลไทม์

ดังนั้นเมื่อพิจารณาการกำหนดช่วงค่าแบนด์วิดท์ที่มากที่สุดที่สามารถใช้บริการได้มีค่าเท่ากับ 166 กิโลบิตต่อวินาที ในการทดลองที่ 2.7 นั้นสามารถหาค่าอัตราการเพิ่มขึ้นของทราฟฟิกได้เป็น 180 เปอร์เซ็นต์ของทราฟฟิกขั้นต่ำที่ต้องการใช้ในการเล่นไฟล์มัลติมีเดีย ดังนั้นในการกำหนดค่า Threshold เพื่อบ่งบอกถึงจุดที่การทำงานต้องสลับอัลกอลิทึมจาก HFSC ไปเป็น HTB เพื่อเพิ่มประสิทธิภาพการควบคุมปริมาณทราฟฟิกและปรับปรุงคุณภาพการให้บริการในระยะยาวนั้นควรจะทำการสลับเมื่อทราฟฟิกของคลาสที่พิจารณามีค่าเกินกว่าค่าที่ได้กำหนดไว้ 180 % ของค่าแบนด์วิดท์ขั้นต่ำที่ได้มีการรับประกันไว้ในคลาสที่ต้องการพิจารณา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การวิเคราะห์และออกแบบระบบ

4.1 การทำงานระบบเดิม

การควบคุมปริมาณทราฟฟิกในเครือข่ายบนเครื่องระบบปฏิบัติการลินุกซ์ การทำงานเดิมจะสั่งงานเป็นลักษณะชุดคำสั่งแบบบรรทัด (Command Line) ด้วยชุดคำสั่ง TC ที่มีอยู่ในแพ็คเก็ต Iproute2 ซึ่งยังไม่มีเครื่องมือในการช่วยบริหารจัดการ หรือ เครื่องมือช่วยสร้างชุดคำสั่งควบคุมปริมาณทราฟฟิกที่จะคอยอำนวยความสะดวกต่อการใช้งาน ทำให้ผู้ที่ใช้งานจะต้องมีความรู้ความสามารถทางเทคนิคสูงมาก ตัวอย่างการใช้งานระบบควบคุมปริมาณทราฟฟิกแบบชุดคำสั่งแบบบรรทัด เป็นดังนี้

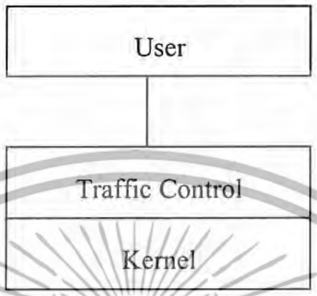
```
tc qdisc add dev eth2 handle 1: root HTB default 16
tc class add dev eth2 parent 1: classid 1:1 HTB rate 100000kbit
tc class add dev eth2 parent 1:1 classid 1:11  HTB rate 200Kbit ceil 350Kbit (Burst) 12kbit
tc class add dev eth2 parent 1:1 classid 1:12 HTB rate 100Kbit ceil 350Kbit (Burst) 12kbit
tc class add dev eth2 parent 1:1 classid 1:13 HTB rate 35Kbit ceil 350Kbit (Burst) 12kbit
tc qdisc add dev eth2 handle 11: parent 1:11 fifo
tc qdisc add dev eth2 handle 12: parent 1:12 fifo
tc qdisc add dev eth2 handle 13: parent 1:13 fifo
iptables -t mangle -N all
iptables -t mangle -A POSTROUTING -j all
iptables -t mangle -A POSTROUTING -p udp --sport 4444 -j CLASSIFY --set-class 1:11
iptables -t mangle -A POSTROUTING -m layer7 --l7proto mms -j CLASSIFY --set-class 1:12
iptables -t mangle -A POSTROUTING -m layer7 --l7proto telnet -j CLASSIFY --set-class 1:13
iptables -t mangle -A POSTROUTING  -m layer7 --l7proto ftp ftp-data -j CLASSIFY --set-class 1:14
iptables -t mangle -A POSTROUTING  -m layer7 --l7proto http -j CLASSIFY --set-class 1:15
iptables -t mangle -A all -j CLASSIFY --set-class 1:16
iptables -t mangle -A all -j RETURN
```

ภาพที่ 4.1 ตัวอย่างชุดคำสั่งแบบบรรทัดการใช้งานระบบควบคุมปริมาณทราฟฟิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากชุดคำสั่งดังภาพที่ 4.1 เป็นตัวอย่างการใช้งานชุดคำสั่งแบบบรรทัด ซึ่งหากการใช้งานมีความหลากหลายมากขึ้น ความซับซ้อนของชุดคำสั่งก็จะมีมากขึ้นตามไปด้วย

แผนภาพการทำงานชุดคำสั่งแบบบรรทัดการใช้งานระบบควบคุมปริมาณทราฟฟิก แสดงได้ดังภาพที่ 4.2



ภาพที่ 4.2 การใช้งานระบบควบคุมปริมาณทราฟฟิกแบบเดิม

4.2 ปัญหาของระบบเดิม

ปัญหาของระบบการใช้งานควบคุมปริมาณทราฟฟิกแบบชุดคำสั่งแบบบรรทัดนั้น มีดังต่อไปนี้

1. ต้องอาศัยผู้ใช้งานที่มีความรู้ความสามารถทางเทคนิคสูง
2. ขาดความยืดหยุ่นในการปรับปรุงแก้ไขค่า
4. ภาษา TC ที่ใช้ในการเขียนชุดคำสั่งมีความซับซ้อนและใช้งานค่อนข้างยาก
4. การรายงานแสดงผลทำความเข้าใจยากและมีรูปแบบที่จำกัด
5. ในช่วงระยะเวลาหนึ่งสามารถทำงานได้อัลกอลิทึมเดียวเท่านั้น ทำให้การสลับอัลกอลิทึมให้ทำงานตามสถานการณ์เฉพาะเจาะจงมีความลำบากและยุ่งยาก

4.3 การทำงานของระบบใหม่

แนวคิดในการออกแบบระบบใหม่ ให้มีความสามารถดังนี้

1. ใช้เทคโนโลยีเว็บแอพพลิเคชั่นด้วยภาษา PHP ในการพัฒนาบนระบบปฏิบัติการลินุกซ์เพื่อนำมาใช้งานแทนการกำหนดค่าแบบบรรทัด
2. อ้างอิงการทำงานของระบบควบคุมปริมาณทราฟฟิกด้วยเทคนิคโทเค็น/บัคเก็ตแบบลำดับชั้น (Hierarchical Token Bucket HTB) และเทคนิคการประกันบริการแบบลำดับชั้นด้วยเส้นโค้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรับประกันการให้บริการ (Hierarchical Fair Service Curve HFSC) โดยจะสนับสนุนการทำงาน คือ การการรับประกันแบนด์วิดท์ (Guarantee Rate) การจำกัดแบนด์วิดท์ (Maximum Rate) และการจัดลำดับความสำคัญ (Priority)

4. สนับสนุนการทำงานตามไอพีแอดเดรส พอร์ต โปรโตคอลระดับชั้นขนส่งข้อมูล (Transport Layer) ค่าลำดับความสำคัญ และ เพิ่มความสามารถการทำงานที่ระดับแอพพลิเคชั่นของโมเดลโอเอสไอ (OSI Model)

4. กำหนดโครงสร้างของคลาสไว้ลึก 4 ระดับ จากการทำงานสูงสุด 8 ระดับ

5. การกรองแพ็คเก็ต แบ่งเป็น 3 ชนิด โดยใช้ไอพีเทเบิลในการกรองทราฟฟิก คือ

5.1 กรองระดับชั้นแอพพลิเคชั่น (Application Layer)

5.2 กรองระดับชั้นขนส่งข้อมูล (Transport Layer)

5.3 กรองระดับชั้นเครือข่าย (Network Layer)

6. แบ่งการควบคุมทราฟฟิกทั้งขาเข้า (Inbound Traffic) และขาออก (Outbound Traffic)

7. สามารถควบคุมปริมาณทราฟฟิกได้หลายๆ อินเทอร์เฟสพร้อมกัน

8. เก็บข้อมูลต่างๆ ไว้บนฐานข้อมูล MySQL

9. การออกแบบการพัฒนาระบบ การกำหนดค่าระบบควบคุมปริมาณทราฟฟิกในลักษณะอ็อบเจ็ค (Object)

10. สร้างค่าอ็อบเจ็คบริการเริ่มต้น (Pre-Define Object) สำหรับบริการที่เป็นที่รู้จัก (Well Known Services) และใช้งานบ่อยๆ

11. เก็บสถิติการใช้งานบริการต่างๆในเครือข่ายลงฐานข้อมูลและนำไปแสดงผลกราฟฟิกผ่านกราฟ

12. การออกแบบอยู่บนสมมติฐานของไฟล์วอลล์ที่เครื่องลินุกซ์เกตเวย์ มีนโยบายดังนี้

12.1 เปิดให้เครื่องใดๆ เข้ามาใช้บริการ HTTP (TCP/80)

12.2 ทำการส่งต่อแพ็คเก็ต (Forward Packets) ออกไปหมดทุกกรณี

12.3 แพ็คเก็ตขาออก (Out put) ออกจากเครื่องได้ทั้งหมด

13. การออกแบบการสร้างกฎนโยบายการควบคุมปริมาณทราฟฟิกถูกออกแบบให้มีการทำงานเป็นวิซาร์ด (Wizard) เพื่อง่ายในการทำงานและเข้าใจง่ายของผู้ใช้งานทั่วไป

14. สนับสนุนเทคนิคการทำงานแบบสลับอัลกอลิทึมอัตโนมัติ (Automatic Swapping Algorithm) นั่นคือ โดยปกติจะทำงานด้วยอัลกอลิทึม HFSC แต่ถ้าหากในขณะที่ทำงานค่าความหนาแน่นของปริมาณทราฟฟิกขาเข้าในเน็ตเวิร์ค (Incoming Network Load) มีค่ามากกว่าค่า Threshold จะทำการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สลับอัลกอลิทึมเป็น HTB และถ้าเมื่อไหร่ค่าความหนาแน่นของทราฟฟิกขาเข้าในเน็ตเวิร์คมีค่าน้อยกว่าค่า Threshold จะทำการสลับอัลกอลิทึมกลับเป็น HFSC อีกครั้ง

15. เพิ่มความสามารถในความปลอดภัยของข้อมูล (Security) โดยการกำหนดให้มีการยืนยันตัวผู้ใช้ก่อนที่จะสามารถเข้าใช้งานระบบได้ ซึ่งทำสามารถที่จะทำการเพิ่ม ลบ และแก้ไขข้อมูลของผู้ใช้ได้

จากภาพที่ 4.3 เป็นการทำงานชุดคำสั่งใช้งานระบบควบคุมปริมาณทราฟฟิกแบบใหม่ ซึ่งมีเว็บแอพพลิเคชั่นคอยช่วยอำนวยความสะดวกให้ผู้ใช้ได้ใช้งานได้สะดวกสบายยิ่งขึ้น

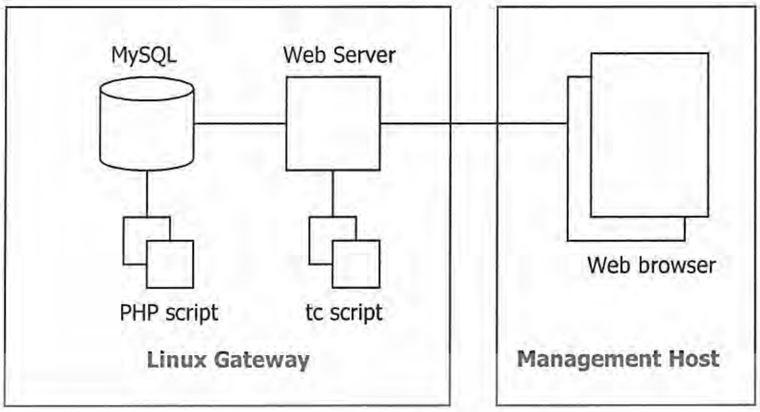
ภาพที่ 4.3 การใช้งานระบบควบคุมปริมาณทราฟฟิกแบบใหม่

4.4 การทำงานโดยภาพรวมของระบบ

การทำงานจะให้ผู้ใช้ ใช้งานผ่านเว็บบราวเซอร์ติดต่อเข้ามายังเครื่องแม่ข่ายเว็บ (Web Server) ผ่านทางโปรโตคอล HTTP ซึ่งข้อมูลเริ่มต้นจะถูกกำหนดขึ้นผ่านหน้าต่างวิซาร์ดเพื่อความสะดวกและเข้าใจง่ายของผู้ใช้ เช่น ข้อมูลการกำหนดค่าของเครื่อง ค่าข้อมูลของกฎนโยบายของการควบคุมปริมาณทราฟฟิกจะถูกจัดเก็บลงฐานข้อมูล เป็นต้น

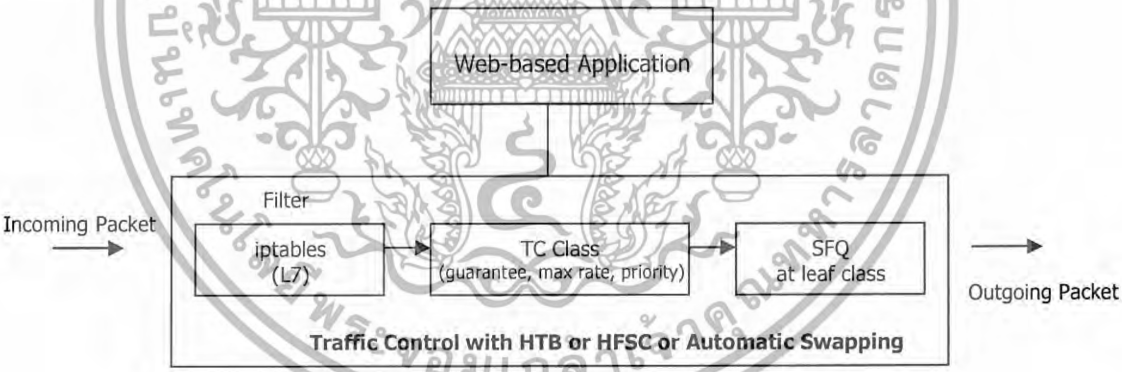
เมื่อทำการติดตั้งกฎนโยบายแล้วจะได้เชลล์สคริปต์ (TC Script) ที่ประกอบด้วยกฎนโยบายของการควบคุมปริมาณทราฟฟิกแบบบรรทัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.4 ระบบควบคุมปริมาณทราฟฟิก

จากภาพที่ 4.4 ในส่วนของเครื่องลินุกซ์เกตเวย์ ทำงานเป็น 2 ส่วน คือ ส่วนของการสร้างกฎนโยบายของการควบคุมปริมาณทราฟฟิก แล้วนำกฎนโยบายดังกล่าวสร้างออกมาเป็นเชลล์สคริปต์ (TC Script) และส่วนการทำงานของสคริปต์ (PHP Script) ที่ทำการอ่านค่าสถิติทางเครือข่ายของ TC เข้าไปเก็บไว้ที่ฐานข้อมูล เพื่อนำไปแสดงผลเป็นรายงานทางเว็บต่อไป



ภาพที่ 4.5 การไหลของแพ็คเก็ตของระบบควบคุมปริมาณทราฟฟิก

จากภาพที่ 4.5 แสดงให้เห็นถึงการไหลของแพ็คเก็ตตั้งแต่ที่แพ็คเก็ตวิ่งเข้ามาในระบบและได้วิ่งออกไปจากระบบ การทำงานเป็นดังนี้ เมื่อเว็บแอพพลิเคชั่นทำการติดตั้งกฎนโยบาย แพ็คเก็ตที่วิ่งเข้ามายังเครื่องลินุกซ์เกตเวย์จะถูกกรองด้วยไอพีเทเบิลสำหรับควบคุมปริมาณทราฟฟิกในระดับแอพพลิเคชั่น โดยการควบคุมปริมาณทราฟฟิกจะทำงานด้วยอัลกอลิทึมใดขึ้นกับการกำหนดของผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้ใช้สามารถเลือกกำหนดอัลกอลิทึมในการควบคุมปริมาณทราฟฟิกได้ 3 ชนิด คือ ด้วยเทคนิคโทเค็น/บัคเก็ตแบบลำดับชั้น (HTB) เทคนิคการประกันบริการแบบลำดับชั้นด้วยเส้นโค้งการรับประกันการให้บริการ (HFSC) หรือเทคนิคการทำงานแบบสลับอัลกอลิทึมอัตโนมัติ (Automatic Swapping Algorithm)

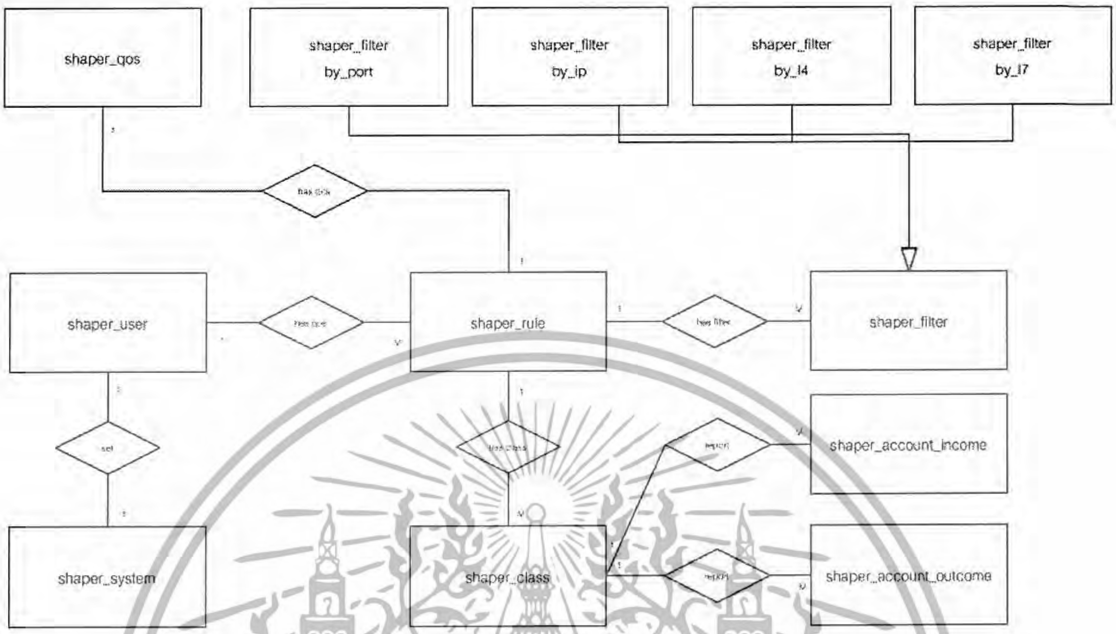
หลังจากนั้นแพ็คเก็ตก็จะถูกปรับแต่งตามการกำหนดแบนด์วิดท์ต่ำสุดที่รับประกัน แบนด์วิดท์สูงสุดที่กำหนด และ การจัดลำดับความสำคัญที่ได้กำหนดไว้ จะปล่อยแพ็คเก็ตออกไปยังอินเทอร์เฟสด้วยอัลกอลิทึมแบบ SFQ ต่อไป

4.5 แผนภาพคลาส (Class-Diagram)

ภาพที่ 4.6 แผนภาพคลาสของการพัฒนาระบบ

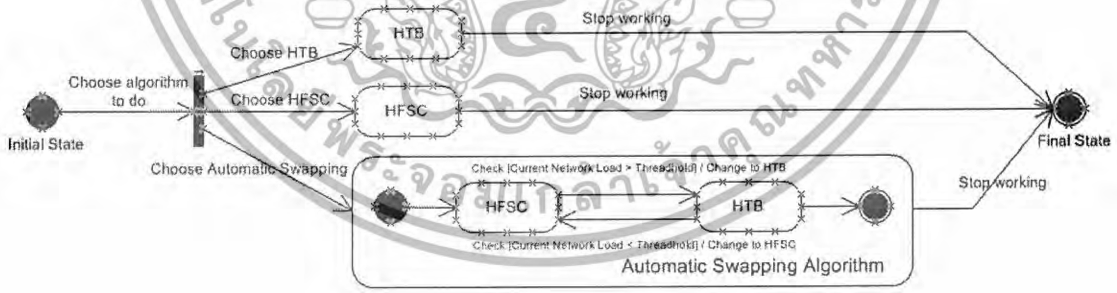
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 แผนภาพอีอาร์ (ER-Diagram)



ภาพที่ 4.7 แผนภาพอีอาร์ของการพัฒนาระบบ

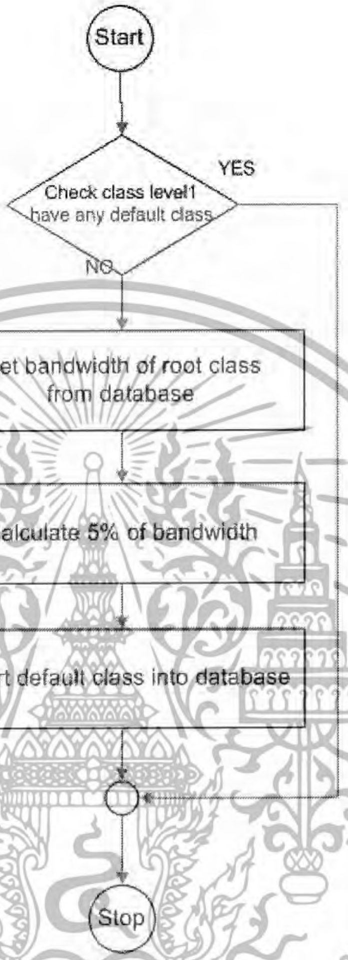
4.7 แผนภาพสถานะ (State Chart-Diagram)



ภาพที่ 4.8 แผนภาพสถานะของการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.8 ไดอะแกรมแสดงขั้นตอนการทำงานหลักของแอพพลิเคชั่น (Flow Chart)



ภาพที่ 4.9 ไดอะแกรมแสดงขั้นตอนการทำงานของไฟล์ make_default.php

จากภาพที่ 4.9 แสดงขั้นตอนการทำงานของไฟล์ make_default.php โดยจะเริ่มการทำงานตั้งแต่ผู้ใช้กำหนดนโยบายในการทำงานเสร็จ ซึ่งในการทำงานนั้นจะไปเริ่มที่การสร้างคลาสดีฟอลต์ (Default) ก่อน โดยจะเริ่มจากการตรวจสอบที่คลาสในระดับชั้นที่ 1 ของอินเทอร์เฟสนั้นๆ ว่ามีคลาสที่ชื่อว่าดีฟอลต์หรือไม่ หากยังไม่มีจะทำการดึงขนาดแบนด์วิดท์ของคลาสรากของอินเทอร์เฟสนั้นๆ แล้วนำมาคำนวณหาค่า 5 เปอร์เซ็นต์ แล้วเพิ่มข้อมูลเกี่ยวกับคลาสดีฟอลต์ลงในฐานข้อมูล เพื่อนำไปใช้ในการทำงานต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ต้องทำการสร้างคลาสดีฟอลต์ในระดับชั้นที่ 1 เนื่องจากว่าหากทราฟฟิกที่เข้ามาไม่ตรงตามการกรองเข้าคลาสใดๆในระดับชั้นที่ 1 เลย แล้วหากเราใช้อัลกอลิทึม HTB ทราฟฟิกนั้นๆจะผ่านไปได้โดยไม่โดนทำการควบคุมปริมาณทราฟฟิกเลย ซึ่งหากมีทราฟฟิกในกรณีนี้มากๆอาจก่อให้เกิดการทำงานของระบบแบบที่ไม่ตรงตามที่ต้องการ เช่น การแย่งใช้แบนด์วิดท์จนแต่ละคลาสอาจจะไม่ได้ใช้ตามที่สมควร แต่หากเราใช้อัลกอลิทึม HFSC ทราฟฟิกนั้นๆจะไม่สามารถผ่านการควบคุมปริมาณทราฟฟิกไปได้เลย ซึ่งหากทราฟฟิกที่เรากล่าวถึงเป็นทราฟฟิกที่ใช้ในการสร้างการเชื่อมต่อการติดต่อนั้นๆ ก็จะไม่สามารถเชื่อมต่อได้ ทำให้การส่งข้อมูลนั้นไม่เกิดขึ้น

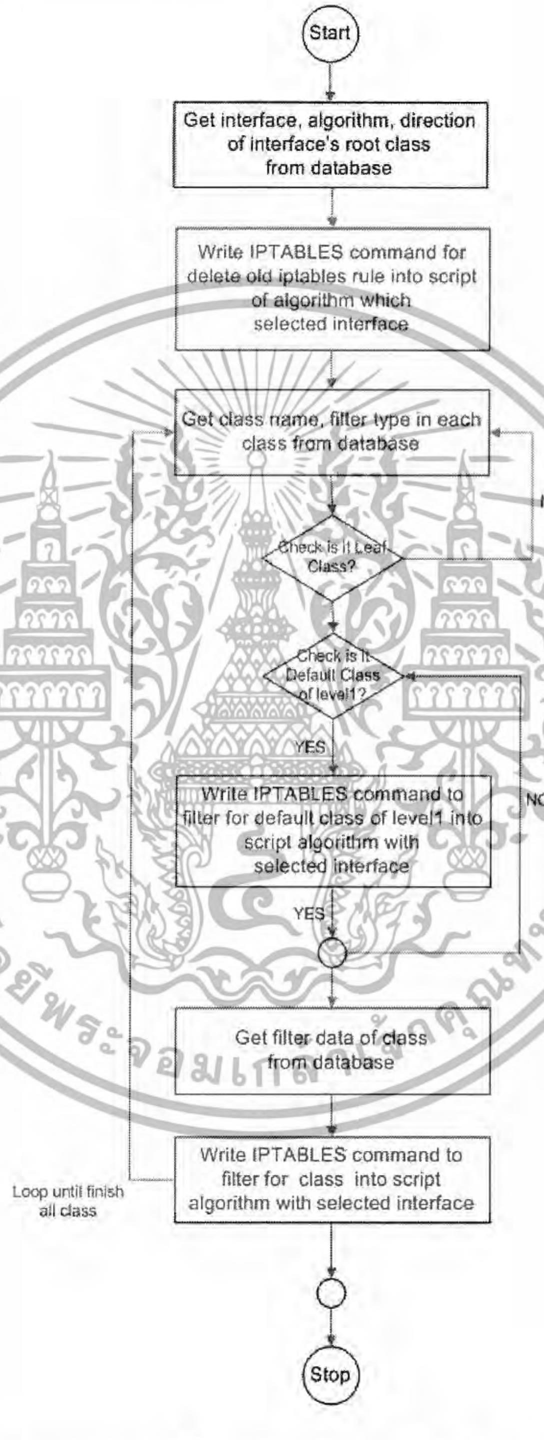
ดังนั้นจึงถือว่าคลาสดีฟอลต์ในระดับชั้นที่ 1 นั้นมีผลต่อการทำงานของระบบมาก จึงต้องมีการสร้างคลาสนี้ขึ้นมาเพื่อให้การทำงานเป็นไปตามที่เราต้องการ แต่หากผู้ใช้ได้ทำการสร้างไว้อยู่แล้วระบบจะไม่ทำการสร้างซ้ำอีก โดยการที่ระบบสร้างคลาสดีฟอลต์นี้ขึ้นมาให้นั้นจะมีการกำหนดแบนด์วิดท์ที่รับประกันและแบนด์วิดท์สูงสุดให้มีขนาดเป็น 5 เปอร์เซ็นต์ของขนาดที่ผู้ใช้กำหนดแบนด์วิดท์ให้กับคลาสรากของอินเทอร์เฟสนั้นๆ จากการที่กำหนดให้ขนาดเป็นห้าเปอร์เซ็นต์นั้นเนื่องจากไม่ต้องการให้ทราฟฟิกทั้งหมดที่ถูกกรองเข้าคลาสนี้ไปกระทบการทำงานหลักของระบบมากนัก จึงทำการแบ่งออกมาให้คลาสดีฟอลต์เพียง 5 เปอร์เซ็นต์

จากภาพที่ 4.10 แสดงขั้นตอนการทำงานของไฟล์ make_iptables.php นั่นคือ หลังจากที่ทำการตรวจสอบและสร้างคลาสดีฟอลต์ให้กับอินเทอร์เฟสแล้ว ต่อมาจะทำการสร้างชุดคำสั่งไอพีเทเบิลเพื่อใช้ในการกรองทราฟฟิกเข้าในแต่ละคลาส ซึ่งในการกรองทราฟฟิกนั้นจะสนใจที่คลาสสุดท้าย (Leaf Class) เท่านั้น โดยในการกรองจะมีให้เลือกทั้งหมดสี่แบบ ได้แก่ การกรองด้วยไอพีแอดเดรส พอร์ต โปรโตคอลระดับชั้นทรานสปอร์ต และระดับชั้นแอพพลิเคชั่น ซึ่งผู้ใช้จะสามารถเลือกกรองได้ด้วยการกรองแบบใดแบบหนึ่งเท่านั้น แต่จะกำหนดได้หลายข้อในการกรองแบบเดียวกัน

เริ่มการทำงานในส่วนของการสร้างชุดคำสั่งไอพีเทเบิลด้วย การดึงข้อมูลของคลาสรากของอินเทอร์เฟสนั้นๆ เพื่อนำมาพิจารณาว่าชุดคำสั่งไอพีเทเบิลนั้นควรเขียนลงในสคริปต์ที่ชื่อว่าอะไร ต่อมาทำการเขียนชุดคำสั่งที่จะใช้ในการลบคำสั่งไอพีเทเบิลเดิมที่เคยกำหนดไว้ออก แล้วทำการดึงข้อมูลต่างๆของคลาสจากฐานข้อมูล แล้วทำการตรวจสอบว่าเป็นคลาสปลายหรือไม่ หากใช่ก็ทำการตรวจสอบต่อว่าเป็นคลาสดีฟอลต์ในระดับคลาสที่ 1 หรือไม่ ที่ต้องตรวจสอบว่าใช่คลาสดีฟอลต์หรือไม่นั้นก็เนื่องจากชุดคำสั่งไอพีเทเบิลสำหรับคลาสดีฟอลต์นั้นจะมีรูปแบบที่แตกต่างจากชุดคำสั่งที่ใช้ในการกรองเข้าคลาสอื่นๆ ซึ่งหากเป็นคลาสดีฟอลต์จริงก็จะทำการเขียนชุดคำสั่งไอพีเทเบิลของคลาสดีฟอลต์นั้นๆลงในสคริปต์ แต่หากไม่ใช่คลาสดีฟอลต์ก็จะทำการดึงข้อมูลเกี่ยวกับการกรองของคลาสปลายนั้นๆ เพื่อนำมาสร้างชุดคำสั่งที่ใช้ในการกรองข้อมูลเข้าคลาสนั้นๆต่อ โดยจะทำการวน

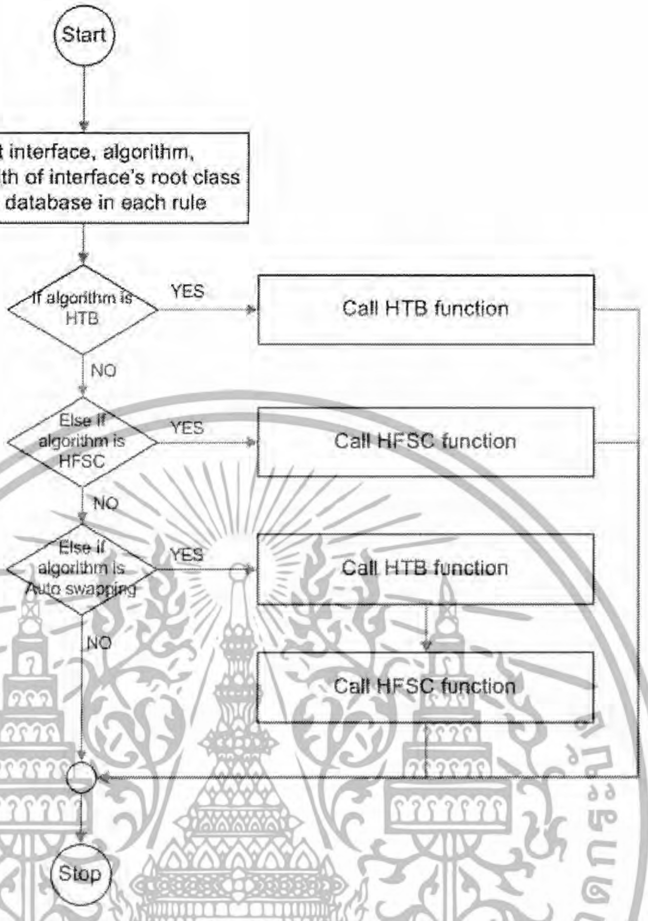
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรวจสอบแล้วสร้างชุดคำสั่งเช่นนี้จนครบทุกคลาสที่อยู่ภายใต้คลาสรากของอินเทอร์เฟสนั้นๆจึงเป็นอันเสร็จกระบวนการสร้างชุดคำสั่งไอพีเทเบิล



ภาพที่ 4.10 ไดอะแกรมแสดงขั้นตอนการทำงานของไฟล์ make_iptables.php

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.11 ไดอะแกรมแสดงขั้นตอนการทำงานของไฟล์ make_tc.php

จากภาพที่ 4.11 แสดงขั้นตอนการทำงานของไฟล์ make_tc.php นั่นคือ หลังจากทำการตรวจสอบคลาสดีฟอลต์และสร้างกฎการกรองทราฟฟิกด้วยชุดคำสั่งไอพีเทเบิลแล้ว จากนั้นจะเริ่มการทำงานในการสร้างชุดคำสั่งที่เกี่ยวกับการควบคุมปริมาณทราฟฟิกโดยตรง คือ การสร้างชุดคำสั่ง TC เริ่มด้วยการดึงข้อมูลเกี่ยวกับอัลกอลิทึมที่ผู้ใช้เลือก ขนาดแบนด์วิดท์ของคลาสราก และ อินเทอร์เฟสที่ต้องการทำการควบคุมปริมาณทราฟฟิก โดยในเวลาหนึ่งจะมีการกำหนดกฎนโยบายการควบคุมปริมาณทราฟฟิกได้แค่ 1 ชุดเท่านั้นในแต่ละอินเทอร์เฟส ดังนั้นหากเมื่อไหร่ที่ต้องการเปลี่ยนแปลง อัลกอลิทึมในการควบคุมปริมาณทราฟฟิก ต้องทำการสั่งยกเลิกกฎชุดเก่าก่อนแล้วค่อยทำการสั่งให้ กฎชุดใหม่ทำงาน โดยในระบบที่พัฒนาจะทำการเก็บชุดคำสั่งของการกรองทราฟฟิกกับชุดคำสั่งของการควบคุมปริมาณทราฟฟิกไว้ด้วยกันแล้วเก็บเป็นสคริปต์ไว้ ซึ่งแต่ละอัลกอลิทึมจะเก็บเป็นสคริปต์ที่แยกกัน แล้วคนละอินเทอร์เฟสก็แยกเป็นคนละสคริปต์อีกเช่นกัน ตัวอย่างเช่น หากผู้ใช้ทำการเลือก อัลกอลิทึม

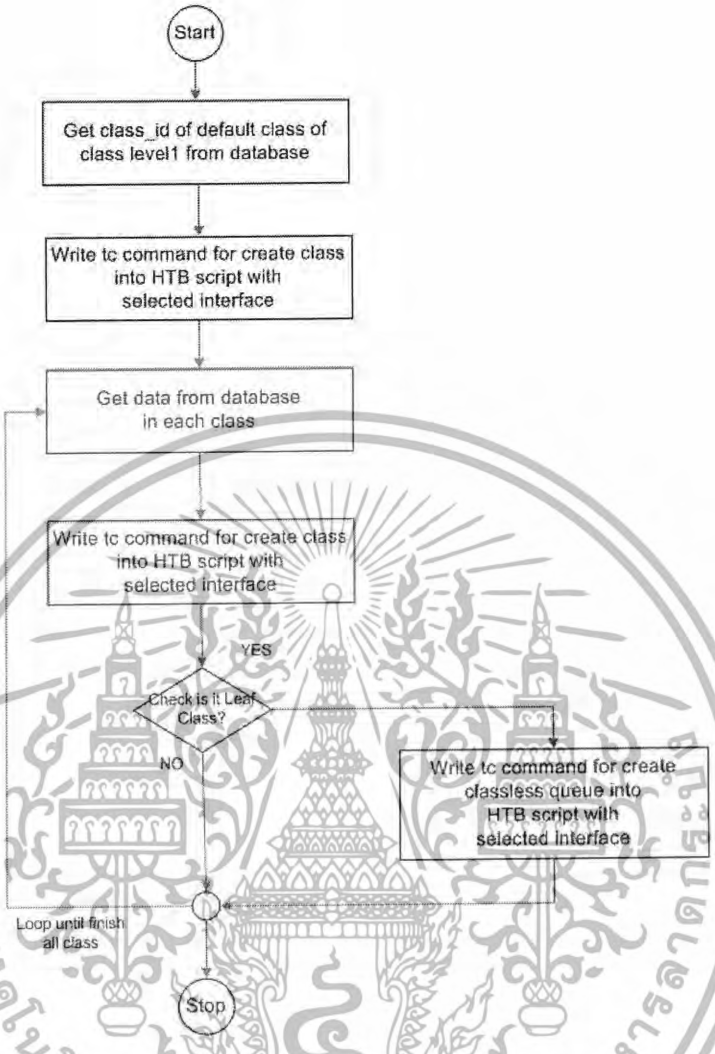
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบ HTB ของอินเทอร์เฟส eth0 ก็จะมีสคริปต์ที่ชื่อว่า HTB_eth0.sh เก็บไว้ที่ path เดียวกับโปรแกรม แต่หากผู้ใช้ทำการเลือกแนวคิดการสลับอัลกอลิทึมโดยอัตโนมัติ (Automatic Swapping Algorithm) ที่อินเทอร์เฟส eth2 ด้วย ก็จะมีการสร้างชุดคำสั่งลงในสคริปต์เพิ่มอีก 2 สคริปต์ คือ ชุดคำสั่งของการควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึมแบบ HTB และ HFSC ของอินเทอร์เฟส eth2 ที่มีชื่อว่า HTB_eth2.sh และ HFSC_eth2.sh โดยในทั้ง 2 สคริปต์นี้จะเก็บกฎการกรองทราฟฟิกที่เป็นคำสั่งชุดเดียวกัน มีการสร้างคลาสจากคำสั่ง TC เหมือนกัน จะต่างกันที่อัลกอลิทึมที่ใช้ในการควบคุมปริมาณทราฟฟิกและการใช้งานพารามิเตอร์เท่านั้น

ดังนั้นเนื่องจากระบบทำการรองรับการควบคุมปริมาณทราฟฟิกที่สองอินเทอร์เฟสพร้อมกัน ซึ่งหากแต่ละอินเทอร์เฟสเลือกใช้อัลกอลิทึมแบบการสลับอัลกอลิทึมโดยอัตโนมัติแล้ว ก็จะมีจำนวนสคริปต์ในระบบทั้งหมด 4 สคริปต์ ซึ่งจะมีการสร้างสคริปต์ขึ้นใหม่ในการกำหนดกฎครั้งแรกเท่านั้น แต่หากมีการกำหนดกฎใหม่ก็จะเป็นการเขียนสคริปต์นั้นทับเฉยๆ ซึ่งหากแต่เดิมใช้อัลกอลิทึมแบบการสลับอัลกอลิทึมโดยอัตโนมัติ ก็จะมี 2 สคริปต์ของอินเทอร์เฟสนั้นๆ แต่หากพอกำหนดใหม่เป็นการเลือกใช้อัลกอลิทึมแบบ HTB ในระบบก็จะยังคงมี 2 สคริปต์อยู่ คือ มีทั้ง HTB และ HFSC แต่สคริปต์ที่ถูกเขียนทับจะมีแค่สคริปต์ของ HTB เท่านั้นและจะเป็นสคริปต์เดียวที่ถูกรัน ส่วนสคริปต์ HFSC เดิม จะไม่ถูกรัน เนื่องจากในการเลือกสคริปต์มารันนั้นจะเลือกตามอัลกอลิทึมที่ระบุไว้ในฐานข้อมูล หากผู้ใช้เลือกที่จะทำการควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึมแบบ HTB แล้วถึงแม้ว่าในระบบจะมี 2 สคริปต์ก็ตามแต่ว่าจะเลือกแค่สคริปต์ของ HTB เท่านั้นมาทำงาน

ดังนั้นหากดึงข้อมูลของอัลกอลิทึมที่ผู้ใช้เลือกในการควบคุมปริมาณทราฟฟิกขึ้นมาแล้ว เป็น HTB ก็จะไปเรียกใช้ฟังก์ชั่น HTB เพื่อสร้างสคริปต์การทำงานของ HTB ของอินเทอร์เฟสนั้นๆ แต่หากเป็น HFSC ก็จะไปเรียกใช้ฟังก์ชั่น HFSC เพื่อสร้างสคริปต์การทำงานของ HFSC เช่นกัน แต่หากเป็น อัลกอลิทึมการสลับอัลกอลิทึมโดยอัตโนมัติก็จะไปเรียกทั้งสองฟังก์ชั่นเพื่อสร้างทั้งสองสคริปต์เลย

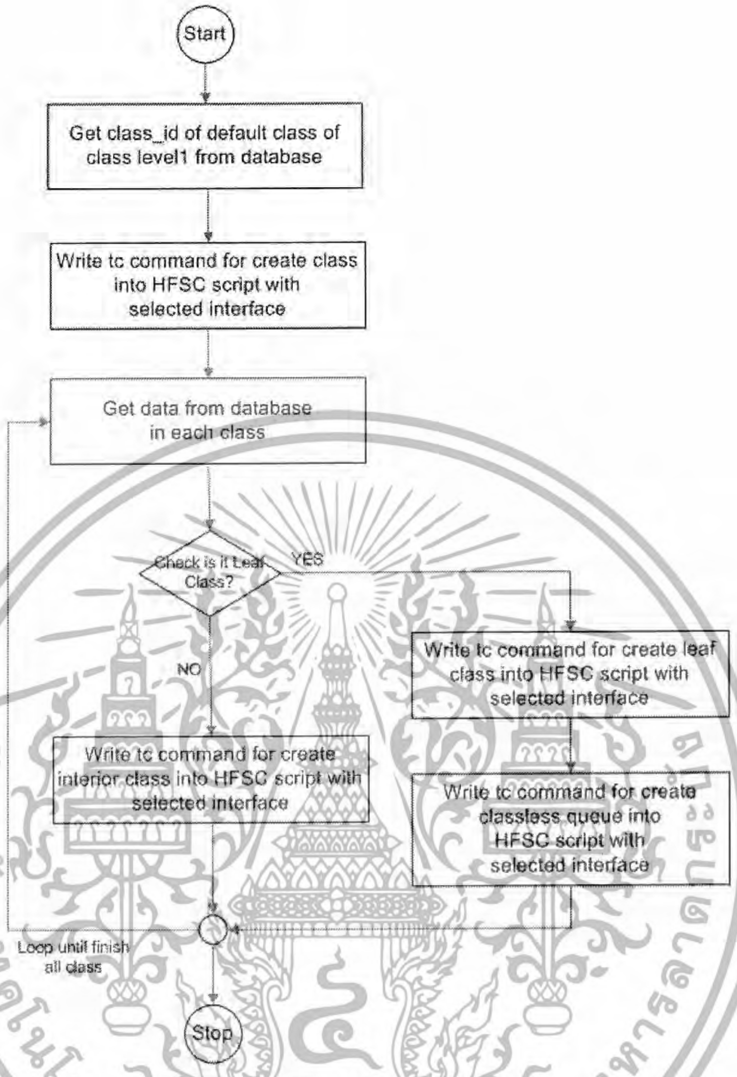
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.12 ไดอะแกรมแสดงขั้นตอนการทำงานของการสร้างสคริปต์อัลกอลทึม HTB

จากภาพที่ 4.12 แสดงขั้นตอนการทำงานของการสร้างสคริปต์อัลกอลทึม HTB โดยจะเริ่มด้วยการดึงคลาสไอดี (Class ID) ของคลาสดีฟอลต์ในระดับที่ 1 มาก่อน เพื่อนำมาใช้ในการสร้างคลาสรากของการทำงานแบบ HTB แล้วนำไปเขียนลงในสคริปต์ของอัลกอลิทึม HTB ของอินเทอร์เฟสนั้นๆ จากนั้นทำการดึงข้อมูลของแต่ละคลาสเพื่อทำการเขียนสคริปต์ที่ในการสร้างคลาส การแจกแจงปริมาณแบนด์วิดท์ที่รับประกัน และปริมาณแบนด์วิดท์ที่มากที่สุดที่คลาสนั้นๆใช้งานได้ แล้วจะทำการตรวจสอบว่าคลาสนั้นๆเป็นคลาสปลายหรือไม่ ซึ่งหากว่าใช่จะมีการสร้างข้อกำหนดของแถวคอยแบบที่ไม่มีคลาสลูก (Classless Qdiscs) ซึ่งในระบบเราเลือกใช้แถวคอยแบบ SFQ (Stochastic Fair Queuing) แล้วทำการวนเช็คเช่นนี้จนครบทุกคลาสตามที่ผู้ใช้สร้าง

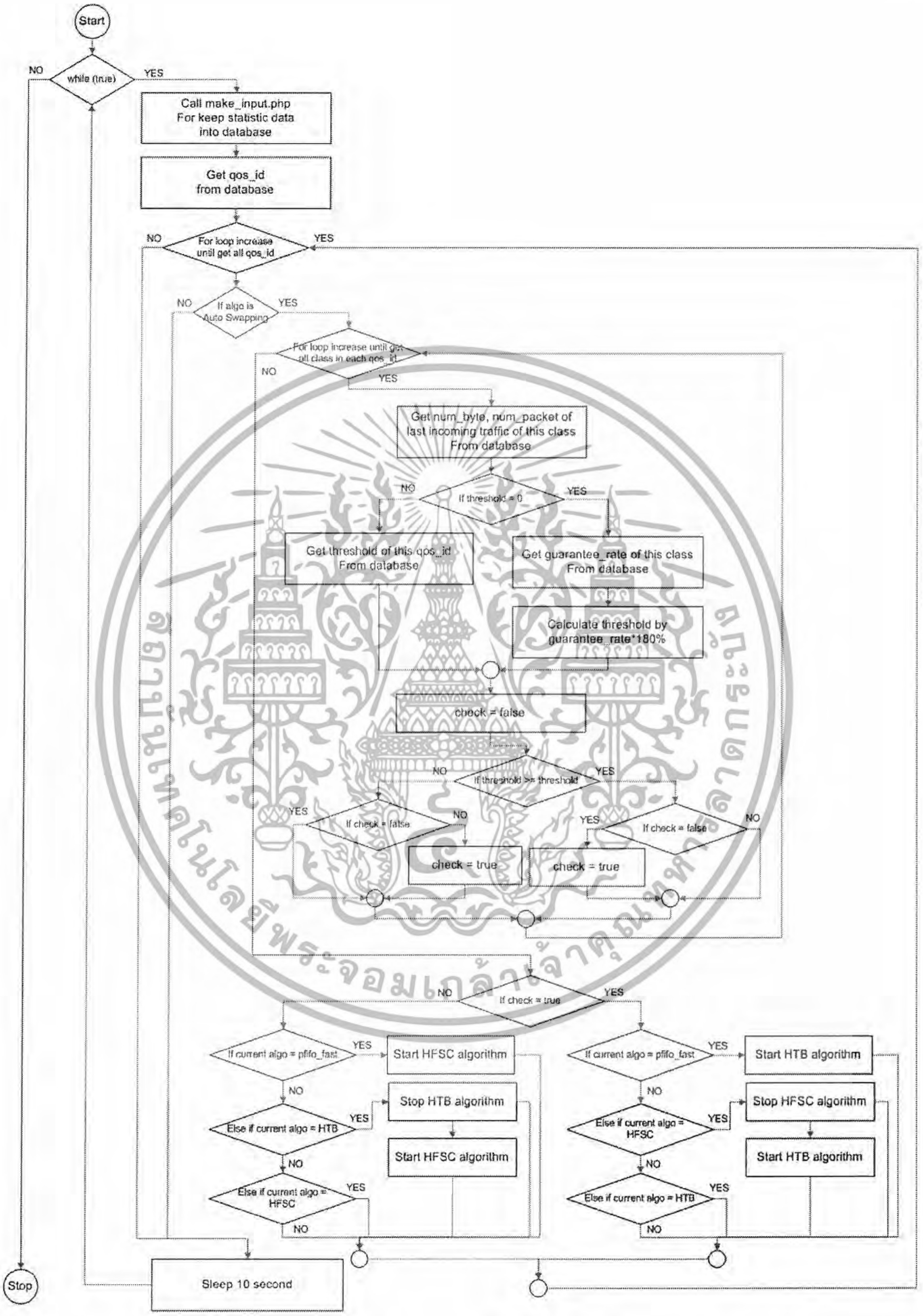
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.13 ไดอะแกรมแสดงขั้นตอนการทำงานของการสร้างสคริปต์อัลกอลทึม HFSC

จากภาพที่ 4.13 แสดงขั้นตอนการทำงานของการสร้างสคริปต์อัลกอลทึม HFSC โดยจะมีขั้นตอนการทำงานที่คล้ายกับฟังก์ชั่น HTB แต่จะต่างกันที่ HFSC มีชุดคำสั่งที่ใช้สร้างคลาสระดับกลาง (Interior Class) และระดับปลายที่ต่างกัน (Leaf Class) ดังนั้นหลังจากที่ทำการสร้างคลาสรากเสร็จแล้วจะทำการเช็คก่อนว่าคลาสนั้นๆที่ทำการดึงข้อมูลมาเป็นคลาสระดับปลายหรือไม่ หากไม่ใช่จะทำการสร้างคลาสด้วยชุดคำสั่งของคลาสระดับกลาง แต่หากว่าเป็นคลาสระดับปลายจริงจะทำการสร้างคลาสด้วยชุดคำสั่งของคลาสระดับปลายพร้อมทั้งจะทำการสร้างข้อกำหนดของแถวคอยแบบที่ไม่มีคลาสลูก (Classless Qdiscs) ด้วย ซึ่งใช้เป็นแบบ SFQ เช่นเดียวกันกับใน HTB แล้วทำการวนสร้างคลาสเช่นนี้จนครบทุกคลาสตามที่ผู้ใช้สร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.14 ไดอะแกรมแสดงขั้นตอนการทำงานของไฟล์ shaper_collector.php

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 4.14 แสดงขั้นตอนการทำงานของไฟล์ shaper_collector.php ทำหน้าที่ในการเก็บข้อมูลการใช้งานทราฟฟิกลงฐานข้อมูลทุกๆ 10 วินาทีและเชคเงื่อนไขการสลับอัลกอลิทึม หากมีการกำหนดให้ทำงานด้วยแนวคิดสลับอัลกอลิทึมโดยอัตโนมัติ โดยจะเป็นการวนลูปการทำงานแบบไม่รู้จบเพื่อเก็บข้อมูลและเชคเงื่อนไขการสลับอัลกอลิทึมทุกๆ 10 วินาที โดยทุกครั้งที่เชคเงื่อนไข while ลูปซึ่งมีค่าเป็นจริงตลอดการทำงานจะทำการเก็บข้อมูลการใช้งานทราฟฟิกผ่านการเรียกไฟล์ make_input.php เพื่อเก็บข้อมูลลงฐานข้อมูล แล้วทำการเชคจำนวนกฎนโยบายควบคุมปริมาณทราฟฟิกในฐานข้อมูลและทำการวนลูปเพื่อดึงไอดีของกฎนโยบายควบคุมปริมาณทราฟฟิก (qos_id) มาทำงานต่อไป โดยทำการเชคอัลกอลิทึมที่กำหนดในกฎนโยบายควบคุมปริมาณทราฟฟิกว่าเป็นแนวคิดการสลับอัลกอลิทึมโดยอัตโนมัติ (Automatic Swapping algorithm) หรือไม่ ถ้าใช่จะผ่านกระบวนการเชคเงื่อนไขในการสลับอัลกอลิทึมต่อไป แต่ถ้าไม่ใช่จะทำการเรียกฟังก์ชั่น sleep เพื่อหน่วงเวลา 10 วินาที เพื่อเริ่มการทำงานใหม่ต่อไป

กระบวนการเชคเงื่อนไขในการสลับอัลกอลิทึม เริ่มต้นทำงานโดยการวนลูปดึงข้อมูลแต่ละคลาสในกฎนโยบายควบคุมปริมาณทราฟฟิกนั้นๆ ขึ้น นั่นคือ จำนวนไบต์ข้อมูลทราฟฟิกขาเข้าของคลาสๆ นั้นจากฐานข้อมูลและค่า Threshold โดยทำการเชคค่า Threshold ถ้าหากมีค่าเป็นศูนย์หมายถึงผู้ใช้กำหนดให้แอพพลิเคชั่นคำนวณหาค่า Threshold ที่เหมาะสม โดยแอพพลิเคชั่นจะนำค่าแบนด์วิดท์ต่ำสุดที่รับประกันแก่คลาสๆ นั้น (guarantee rate) มาคูณกับค่า 180% ซึ่งได้จากการทดลองหาค่า Threshold ที่เหมาะสม แต่ถ้าหากค่า Threshold มีค่าไม่เท่ากับศูนย์นั้นหมายถึงผู้ใช้กำหนดค่า Threshold ในการควบคุมปริมาณทราฟฟิกเอง

หลักการสลับอัลกอลิทึมจะมีตัวแปรชนิดบูล (Boolean) ชื่อ check ซึ่งมีค่าเริ่มต้นทุกครั้งที่ทำงานเป็น false โดยเมื่อไหร่ที่ค่าจำนวนไบต์ข้อมูลทราฟฟิกขาเข้าล่าสุดของแต่ละคลาสมีค่ามากกว่าหรือเท่ากับค่า Threshold จะทำการเปลี่ยนค่าตัวแปร check เป็น true กรณีที่มีค่าเป็น false แต่ถ้ามีค่าเป็น true อยู่แล้วจะคงค่าไว้ แต่ถ้าหากค่าข้อมูลมีค่าน้อยกว่าค่า Threshold จะไม่ทำการเปลี่ยนแปลงค่าใน ตัวแปร check เมื่อทำการวนลูปจนครบทุกคลาสในกฎนโยบายควบคุมปริมาณทราฟฟิกแล้ว จะทำการตรวจสอบว่าสุดท้ายตัวแปร check มีค่าเป็นอะไร ถ้าหากมีค่าเป็น true หมายถึงว่ามีคลาสมากกว่าหนึ่งคลาสที่มีค่าจำนวนไบต์ข้อมูลทราฟฟิกขาเข้าล่าสุดมากกว่าหรือเท่ากับค่า Threshold ดังนั้นจะทำการสลับอัลกอลิทึมเป็น HTB เพื่อจัดการทราฟฟิกที่มีความหนาแน่นมาก แต่ถ้ามีค่าเป็น false หมายถึงว่าทุกคลาสในกฎนโยบายควบคุมปริมาณทราฟฟิกจะมีค่าจำนวนไบต์ข้อมูลทราฟฟิกขาเข้าล่าสุดต่ำกว่าค่า Threshold ดังนั้นจะทำการสลับอัลกอลิทึมกลับเป็น HFSC เพื่อจัดการทราฟฟิกในสถานะปกติดังเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.9 พจนานุกรมข้อมูล (Data Dictionary)

ตารางฐานข้อมูลที่ได้ออกแบบไว้ ดังตารางที่ 3-1 ประกอบด้วย 10 ตาราง ใช้ในการข้อมูลของผู้ใช้งาน ข้อมูลกำหนดค่าของเครื่อง ข้อมูลการสร้างกฎและข้อมูลสถิติต่างๆ

ตารางที่ 4-1 พจนานุกรมข้อมูลของระบบควบคุมปริมาณทราฟฟิก

Table Name	Description
shaper_qos	ข้อมูลของการปรับแต่งทราฟฟิก
shaper_class	ข้อมูลคลาสตามกฎนโยบาย
shaper_filter_by_port	ข้อมูลการกรองโดยใช้พอร์ต
shaper_filter_by_ip	ข้อมูลการกรองทราฟฟิกโดยใช้ไอพีแอดเดรส
shaper_filter_by_l4	ข้อมูลการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นขนส่งข้อมูล
shaper_filter_by_l7	ข้อมูลการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นแอพพลิเคชั่น
shaper_rule	ข้อมูลกฎนโยบายบริการ
shaper_accounting_income	ข้อมูลสถิติการใช้งานทราฟฟิกขาเข้า
shaper_accounting_outcome	ข้อมูลสถิติการใช้งานทราฟฟิกขาออก
shaper_user	ข้อมูลผู้ใช้งาน
shaper_setting	ข้อมูลการติดตั้งแอพพลิเคชั่น

ตารางที่ 4-2 ข้อมูลของการปรับแต่งทราฟฟิก

System Name: Web Based Application for Bandwidth Management for Real-Time Application					
Table Name: shaper_qos					
Table Description: ข้อมูลของการปรับแต่งทราฟฟิก					
Create Date: 5/12/2006					
Modify Date: 5/12/2006					
Field Name	Data Type	Key	NULL	Description	Ref.Table
qos_id	Integer	PK		หมายเลขการปรับแต่ง	
interface	Varchar(45)			ชื่ออินเทอร์เฟส	
direction	Varchar(45)			ทิศทางของแพ็คเก็ต in = ขาเข้า, out = ขาออก	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4-2 (ต่อ) ข้อมูลของการปรับแต่งทราฟฟิก

Field Name	Data Type	Key	NULL	Description	Ref.Table
bandwidth	Double			ขนาดแบนด์วิดท์	
algo	Varchar(45)			ชนิดอัลกอลิทึม	
threshold	Double			ค่าขอบเขตการสลับอัลกอลิทึม	

ตารางที่ 4-3 ข้อมูลคลาสตามกฎนโยบาย

System Name: Web Based Application for Bandwidth Management for Real-Time Application					
Table Name: shaper_class					
Table Description: ข้อมูลคลาสตามกฎนโยบาย					
Create Date: 5/12/2006					
Modify Date: 5/12/2006					
Field Name	Data Type	Key	NULL	Description	Ref.Table
class_idx	Integer	PK		หมายเลขคลาส	
class_name	Varchar(45)			ชื่อคลาส	
parent_id	Varchar(45)			หมายเลข handle หลักของคลาสแม่	
class_id	Double			หมายเลข handle ของคลาส	
guarantee_rate	Varchar(45)			แบนด์วิดท์ต่ำสุดที่รับประกัน	
max_rate	Double			แบนด์วิดท์สูงสุดที่กำหนด	
priority	Integer			ค่าลำดับความสำคัญ	
filter_type	Varchar(45)			ชนิดการกรองทราฟฟิก	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4-4 ข้อมูลการกรองทราฟฟิกโดยใช้พอร์ต

System Name: Web Based Application for Bandwidth Management for Real-Time Application					
Table Name: shaper_filter_by_port					
Table Description: ข้อมูลการกรองทราฟฟิกโดยใช้พอร์ต					
Create Date: 5/12/2006					
Modify Date: 5/12/2006					
Field Name	Data Type	Key	NULL	Description	Ref.Table
filter_id	Integer	PK		หมายเลขข้อมูลการกรอง	
direction	Varchar(45)			ทิศทาง src = จากต้นทางไปปลายทาง dst = จากปลายทางไปต้นทาง	
protocol_port	Varchar(45)			ชนิดโปรโตคอล (udp, tcp)	
port_number	Varchar(45)			หมายเลยพอร์ต	

ตารางที่ 4-5 ข้อมูลการกรองทราฟฟิกโดยใช้ไอพีแอดเดรส

System Name: Web Based Application for Bandwidth Management for Real-Time Application					
Table Name: shaper_filter_by_ip					
Table Description: ข้อมูลการกรองทราฟฟิกโดยใช้ไอพีแอดเดรส					
Create Date: 5/12/2006					
Modify Date: 5/12/2006					
Field Name	Data Type	Key	NULL	Description	Ref.Table
filter_id	Integer	PK		หมายเลขข้อมูลการกรอง	
direction	Varchar(45)			ทิศทาง src = จากต้นทางไปปลายทาง dst = จากปลายทางไปต้นทาง	
ipaddress	Varchar			หมายเลขไอพีแอดเดรส	
subnet_mask	Varchar(45)			หมายเลขสับเน็ต	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4-6 ข้อมูลการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นขนส่งข้อมูล

System Name: Web Based Application for Bandwidth Management for Real-Time Application					
Table Name: shaper_filter_by_l4					
Table Description: ข้อมูลการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นขนส่งข้อมูล					
Create Date: 5/12/2006					
Modify Date: 5/12/2006					
Field Name	Data Type	Key	NULL	Description	Ref.Table
filter_id	Integer	PK		หมายเลขข้อมูลการกรอง	
protocol	Varchar(45)			ชื่อโปรโตคอล	

ตารางที่ 4-7 ข้อมูลการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นแอพพลิเคชั่น

System Name: Web Based Application for Bandwidth Management for Real-Time Application					
Table Name: shaper_filter_by_l7					
Table Description: ข้อมูลการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นแอพพลิเคชั่น					
Create Date: 5/12/2006					
Modify Date: 5/12/2006					
Field Name	Data Type	Key	NULL	Description	Ref.Table
filter_id	Integer	PK		หมายเลขข้อมูลการกรอง	
protocol	Varchar(45)			ชื่อโปรโตคอล	

ตารางที่ 4-8 ข้อมูลกฎนโยบายบริการ

System Name: Web Based Application for Bandwidth Management for Real-Time Application					
Table Name: shaper_rule					
Table Description: ข้อมูลกฎนโยบายบริการ					
Create Date: 5/12/2006					
Modify Date: 5/12/2006					
Field Name	Data Type	Key	NULL	Description	Ref.Table
rule_id	Integer	PK		หมายเลขกฎนโยบาย	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4-8 (ต่อ) ข้อมูลกฎนโยบายบริการ

Field Name	Data Type	Key	NULL	Description	Ref.Table
qos_id	Integer	FK		หมายเลขการปรับแต่ง	shaper_qos
class_idx	Integer	FK		หมายเลขคลาส	shaper_class
filter_id_port	Integer	FK		หมายเลขการกรองโดยพอร์ต	shaper_filter_by_port
filter_id_ip	Integer	FK		หมายเลขการกรองโดยไอพีแอดเดรส	shaper_filter_by_ip
filter_id_l4	Integer	FK		หมายเลขการกรองโดยโปรโตคอลระดับชั้นขนส่งข้อมูล	shaper_filter_by_l4
filter_id_l7	Integer	FK		หมายเลขการกรองโดยโปรโตคอลระดับชั้นแอพพลิเคชั่น	shaper_filter_by_l7
user_id	Integer	FK		ชื่อผู้แก้ไข	shaper_user

ตารางที่ 4-9 ข้อมูลสถิติการใช้งานทราฟฟิกขาเข้า

System Name: Web Based Application for Bandwidth Management for Real-Time Application					
Table Name: shaper_accounting_income					
Table Description: ข้อมูลสถิติการใช้งานทราฟฟิกขาเข้า					
Create Date: 5/12/2006					
Modify Date: 5/12/2006					
Field Name	Data Type	Key	NULL	Description	Ref.Table
account_id	Integer	PK		หมายเลขสถิติการใช้งาน	
class_idx	Integer	FK		หมายเลขคลาส	shaper_class
num_packet	Integer			จำนวนแพ็คเก็ต	
num_byte	Integer			จำนวนไบต์	
total_packet	Integer			จำนวนแพ็คเก็ตทั้งหมด	
total_byte	Integer			จำนวนไบต์ทั้งหมด	
date	Date			วันเดือนปี	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4-9 (ต่อ) ข้อมูลสถิติการใช้งานทราฟฟิกขาเข้า

Field Name	Data Type	Key	NULL	Description	Ref.Table
time	Time			เวลา	

ตารางที่ 4-10 ข้อมูลสถิติการใช้งานทราฟฟิกขาออก

System Name: Web Based Application for Bandwidth Management for Real-Time Application					
Table Name: shaper_accounting_outcome					
Table Description: ข้อมูลสถิติการใช้งานทราฟฟิกขาออก					
Create Date: 5/12/2006					
Modify Date: 5/12/2006					
Field Name	Data Type	Key	NULL	Description	Ref.Table
account_id	Integer	PK		หมายเลขสถิติการใช้งาน	
class_idx	Integer	FK		หมายเลขคลาส	shaper_class
num_packet	Integer			จำนวนแพ็คเก็ต	
num_byte	Integer			จำนวนไบต์	
total_packet	Integer			จำนวนแพ็คเก็ตทั้งหมด	
total_byte	Integer			จำนวนไบต์ทั้งหมด	
date	Date			วันเดือนปี	
time	Time			เวลา	

ตารางที่ 4-11 ข้อมูลผู้ใช้งาน

System Name: Web Based Application for Bandwidth Management for Real-Time Application					
Table Name: shaper_user					
Table Description: ข้อมูลผู้ใช้งาน					
Create Date: 5/12/2006					
Modify Date: 5/12/2006					
Field Name	Data Type	Key	NULL	Description	Ref.Table
user_id	Integer	PK		หมายเลขผู้ใช้งาน	
user_name	Varchar(45)			ชื่อผู้ใช้งาน	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4-11 (ต่อ) ข้อมูลผู้ใช้งาน

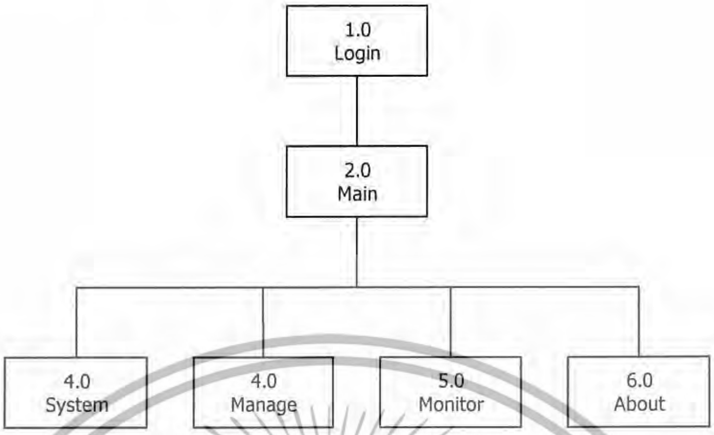
Field Name	Data Type	Key	NULL	Description	Ref.Table
password	Varchar(45)			รหัสผู้ใช้งาน	
name	Varchar(45)			ชื่อเต็มผู้ใช้งาน	
surname	Varchar(45)			นามสกุลผู้ใช้งาน	
role	Varchar(45)			บทบาทหน้าที่ผู้ใช้งาน	

ตารางที่ 4-12 ข้อมูลการติดตั้งแอพพลิเคชั่น

System Name: Web Based Application for Bandwidth Management for Real-Time Application					
Table Name: shaper_setting					
Table Description: ข้อมูลการติดตั้งแอพพลิเคชั่น					
Create Date: 5/12/2006					
Modify Date: 5/12/2006					
Field Name	Data Type	Key	NULL	Description	Ref.Table
setting_id	Integer	PK		หมายเลขข้อมูลการติดตั้ง	
mysql_host	Varchar(45)			ชื่อโฮสของระบบฐานข้อมูล	
mysql_db	Varchar(45)			ชื่อระบบฐานข้อมูล	
mysql_user	Varchar(45)			ชื่อผู้ใช้งานฐานข้อมูล	
mysql_pass	Varchar(45)			รหัสผู้ใช้งานฐานข้อมูล	
tc_path	Varchar(45)			ที่ตั้งของคำสั่ง tc	
iptable_path	Varchar(45)			ที่ตั้งของคำสั่ง iptables	
user_id	Integer	FK		ชื่อผู้แก้ไข	shaper_user

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.10 การออกแบบด้านโครงสร้างการทำงาน



ภาพที่ 4.15 โครงสร้างของเว็บแอพพลิเคชั่นโดยรวม

4.10.1 การออกแบบด้านฟังก์ชั่นการทำงาน

จากภาพที่ 4.15 แบ่งโครงสร้างการทำงานของระบบเว็บแอพพลิเคชั่น ออกเป็นดังนี้

1. Login การพิสูจน์ตัวตนก่อนเข้าใช้งานระบบเว็บแอพพลิเคชั่น
2. Main หน้าหลักของระบบเว็บแอพพลิเคชั่น ประกอบด้วย

2.1 System การกำหนดค่าให้กับระบบ ดังภาพที่ 4.16 ประกอบด้วย

2.1.1 User การจัดการชื่อบัญชีผู้ใช้งาน ดังภาพที่ 4.17 ประกอบด้วย

- การสร้างบัญชีผู้ใช้งาน
- การลบบัญชีผู้ใช้งาน
- การแก้ไขข้อมูลและเปลี่ยนรหัสบัญชีผู้ใช้งาน

2.1.2 Setting การกำหนดค่าให้กับเครื่อง ดังภาพที่ 4. 18 ประกอบด้วย

- การกำหนดค่าต่างๆ ให้กับเครื่อง
- การดูค่าต่างๆ ที่ได้กำหนดไว้ในเครื่อง

2.2 Manage การกำหนดนโยบายของการควบคุมปริมาณทราฟฟิก ดังภาพที่ 4.19 ประกอบด้วย

- การสร้างกฎนโยบายของการควบคุมปริมาณทราฟฟิก
- การแสดง ลบ และฆแก้ไขข้อมูลกฎนโยบายของการควบคุมปริมาณทราฟฟิก
- การแสดงกฎนโยบายของการควบคุมปริมาณทราฟฟิก

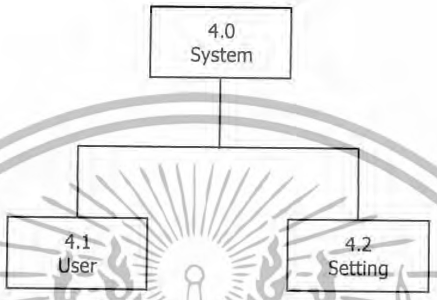
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 Monitor รายงานสถิติการใช้งานทางเครือข่าย ดังภาพที่ 4.20 ประกอบด้วย

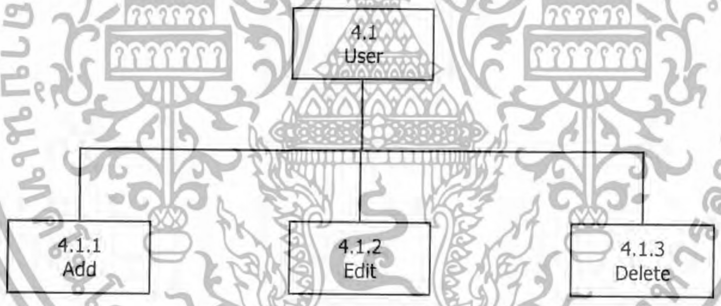
- การดูรายงานสถิติการใช้บริการทางเครือข่ายตามคลาส
- การดูรายงานสถิติการใช้บริการทางเครือข่ายโดยภายรวมของอินเทอร์เฟส

2.4 About

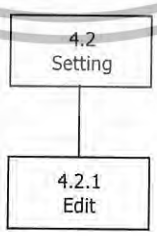
- แสดงรายงานบทคัดย่อโครงงาน



ภาพที่ 4.16 โครงสร้างของเว็บแอพพลิเคชั่นของการกำหนดค่าให้กับระบบ

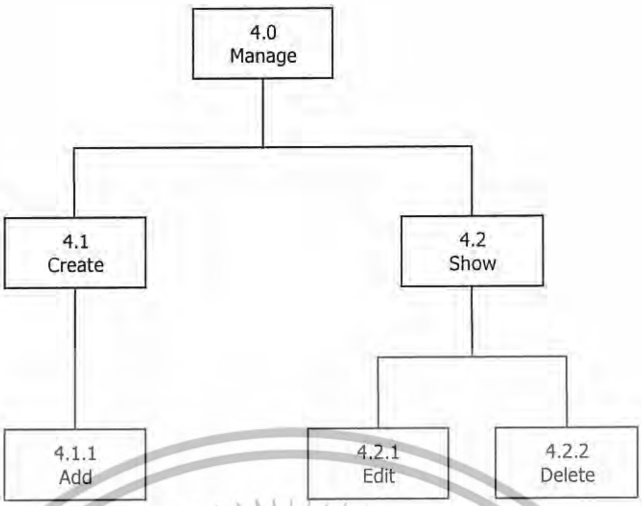


ภาพที่ 4.17 โครงสร้างของเว็บแอพพลิเคชั่นของการจัดการชื่อบัญชีผู้ใช้งาน

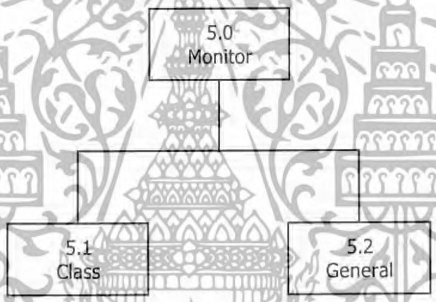


ภาพที่ 4.18 โครงสร้างของเว็บแอพพลิเคชั่นของการกำหนดค่าให้กับเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.19 โครงสร้างของเว็บแอพพลิเคชั่นของการกำหนดนโยบายของการควบคุมปริมาณทราฟฟิก



ภาพที่ 4.20 โครงสร้างของเว็บแอพพลิเคชั่นของรายงานสถิติการใช้งานทางเครือข่าย

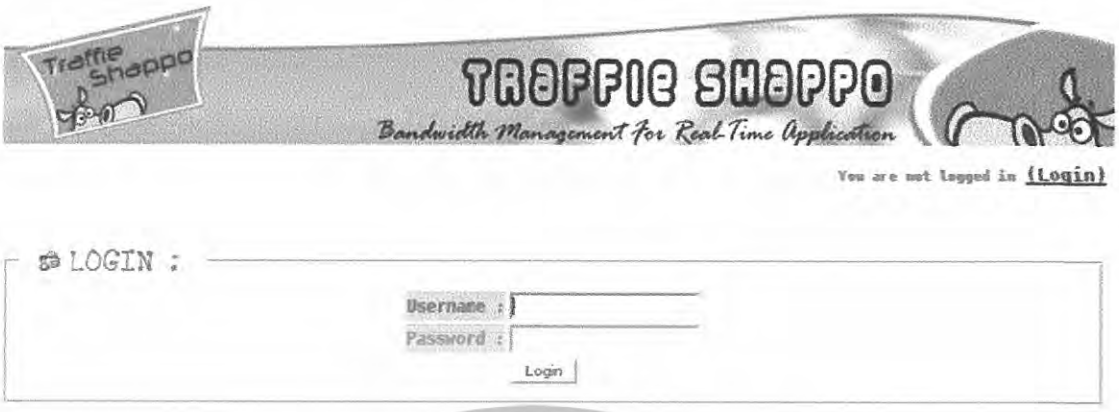
4.10.2 การออกแบบด้านหน้าต่างติดต่อผู้ใช้งาน (User Interface)

แบ่งส่วนการทำงานของเว็บแอพพลิเคชั่น ออกได้เป็นดังนี้

4.10.2.1 การล็อกอินเข้าใช้งานระบบ (Login)

หน้าต่างล็อกอินเข้าใช้งานระบบเพื่อควบคุมสิทธิ์ในการใช้งานระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



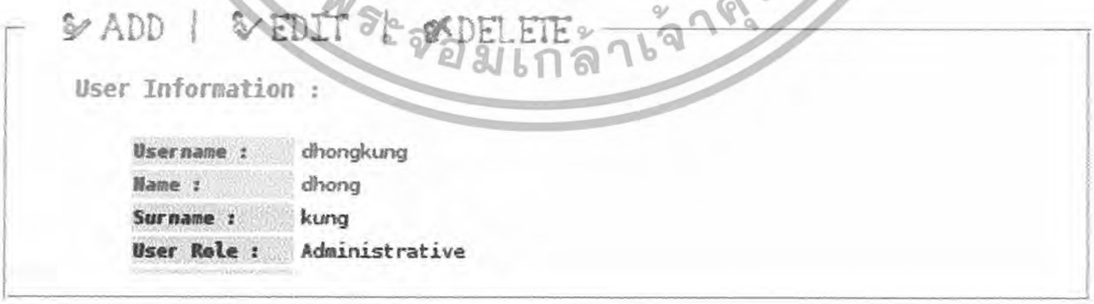
ภาพที่ 4.21 หน้าต่างล็อกอินเข้าใช้งานระบบ

4.10.2.2 การกำหนดค่าให้กับระบบ (Setting)

ส่วนหน้าต่างเหล่านี้จะเป็นตัวกำหนดค่าให้กับระบบในการทำงานของเว็บแอพพลิเคชั่น โดยทำการแบ่งออกเป็น 2 การทำงานหลัก ดังนี้

1. **ส่วนการจัดการชื่อบัญชีผู้ใช้งาน (User)**

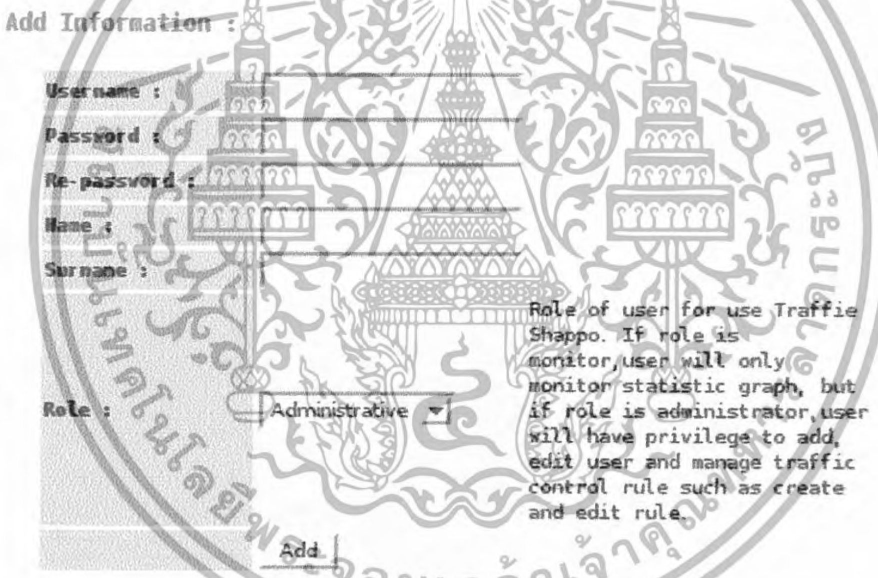
   ส่วนการจัดการผู้ใช้งาน แบ่งออกเป็น การแสดงข้อมูลของบัญชีผู้ใช้งาน การแก้ไขข้อมูลบัญชีผู้ใช้งาน การเปลี่ยนรหัสบัญชีผู้ใช้งาน การสร้างบัญชีผู้ใช้งาน การลบบัญชีผู้ใช้งาน และการเปิดดูรายการทั้งหมดของบัญชีผู้ใช้งาน



ภาพที่ 4.22 หน้าต่างแสดงข้อมูลของบัญชีผู้ใช้งาน

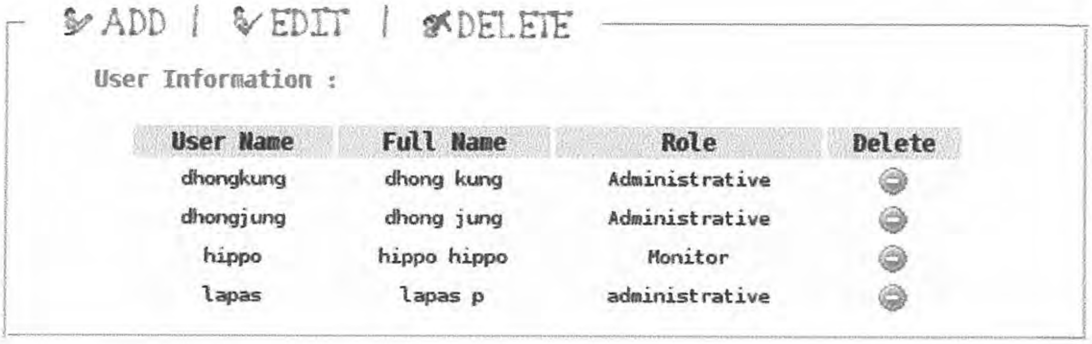
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 4.23 หน้าต่างแสดงการแก้ไขข้อมูลบัญชีผู้ใช้งาน



ภาพที่ 4.24 หน้าต่างแสดงการสร้างบัญชีผู้ใช้งาน

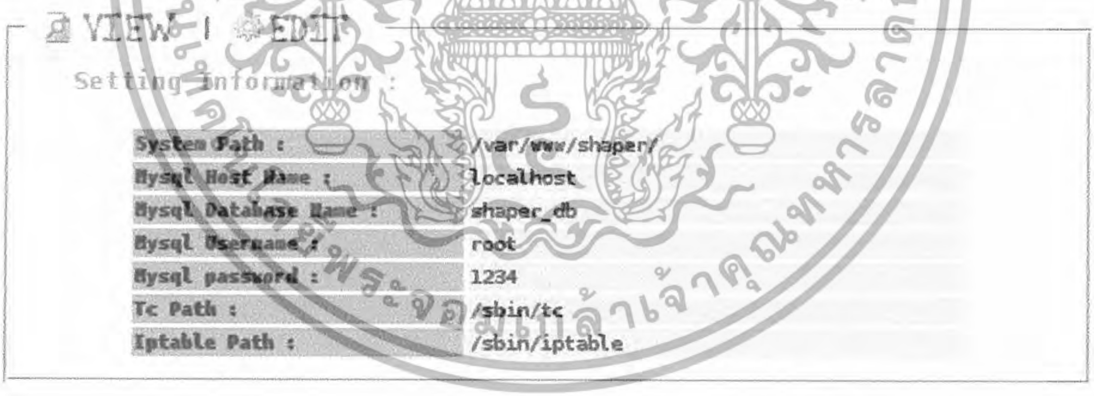
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.25 หน้าต่างแสดงการลบบัญชีผู้ใช้งาน

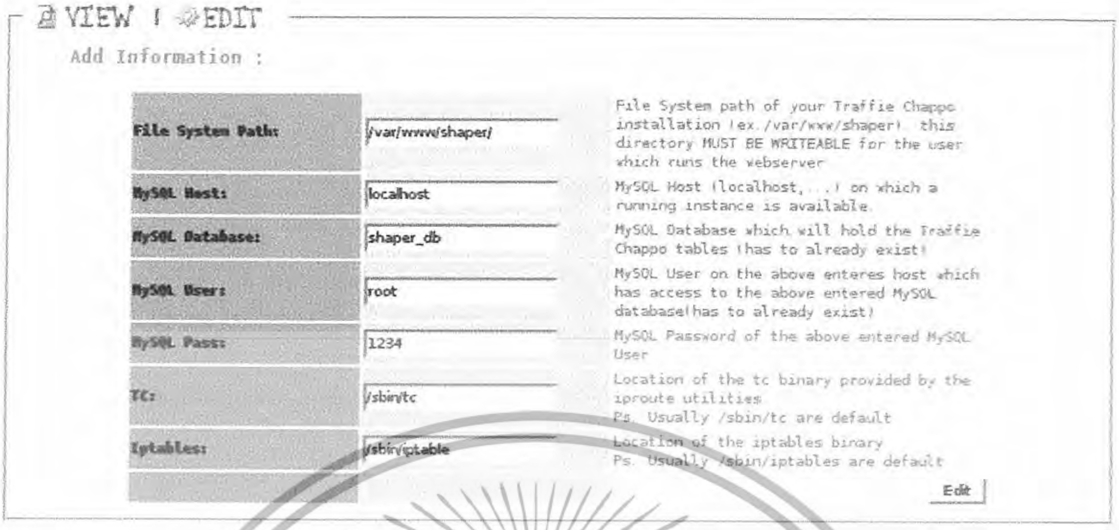
2. **ส่วนการกำหนดค่าให้กับเครื่อง (Setting)**

ส่วนการกำหนดค่าให้กับเครื่อง เป็นการกำหนดข้อมูลพื้นฐานในการทำงานของเว็บแอพพลิเคชั่นและการติดตั้งของเว็บแอพพลิเคชั่น เช่น ที่ตั้งของแอพพลิเคชั่น ชื่อโฮสของระบบฐานข้อมูล ชื่อระบบฐานข้อมูล ชื่อผู้ใช้งานฐานข้อมูล รหัสผู้ใช้งานฐานข้อมูล ที่ตั้งของคำสั่ง Tc และ ไอพีเทเบิล รวมถึงการดูค่าข้อมูลต่างๆ ที่ได้กำหนดไว้ในฐานข้อมูล



ภาพที่ 4.26 หน้าต่างแสดงการดูค่าต่างๆ ที่ได้กำหนดไว้ในเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.27 หน้าต่างแสดงการแก้ไขค่าต่างๆให้กับเครื่อง

4.10.2.3 การกำหนดกฎนโยบายของการควบคุมปริมาณทราฟฟิก (Manage)

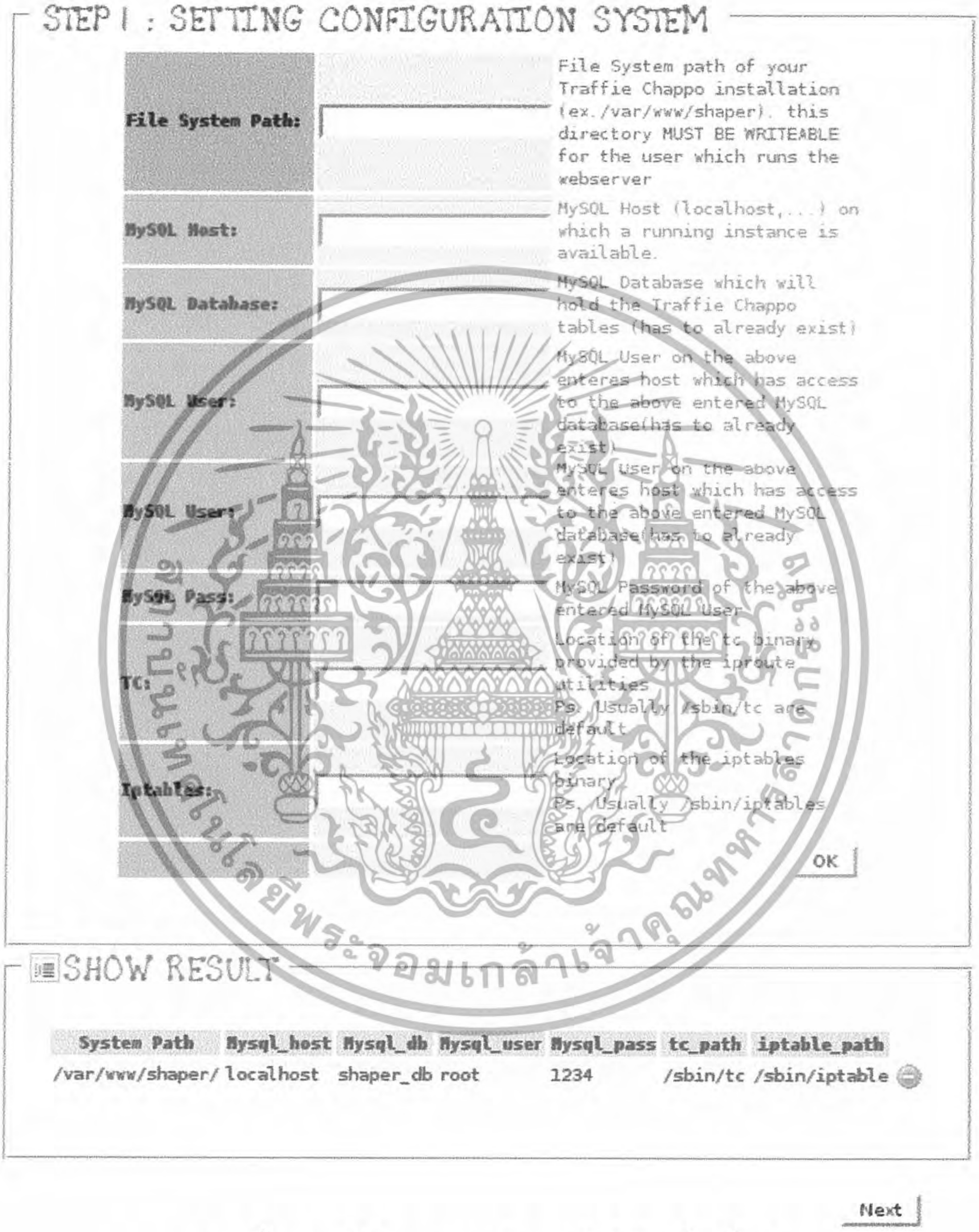
ส่วนหน้าต่างเหล่านี้จะเป็นตัวจัดการนโยบายของการควบคุมปริมาณทราฟฟิก (Policy) ในการทำงานของเว็บแอพพลิเคชั่น โดยทำการแบ่งออกเป็น 2 การทำงานหลัก ดังนี้

1. **ส่วนการสร้างกฎนโยบายของการควบคุมปริมาณทราฟฟิก (Create)**

หน้าต่างสร้างกฎนโยบายของการควบคุมปริมาณทราฟฟิกถูกออกแบบให้มีการทำงานแบบวิซาร์ดเพื่อกำหนดค่าเริ่มต้นในการควบคุมปริมาณทราฟฟิก โดยทำการแบ่งออกเป็น 7 การทำงานย่อย ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1 กำหนดค่าให้กับเครื่องเริ่มต้น (Setting)

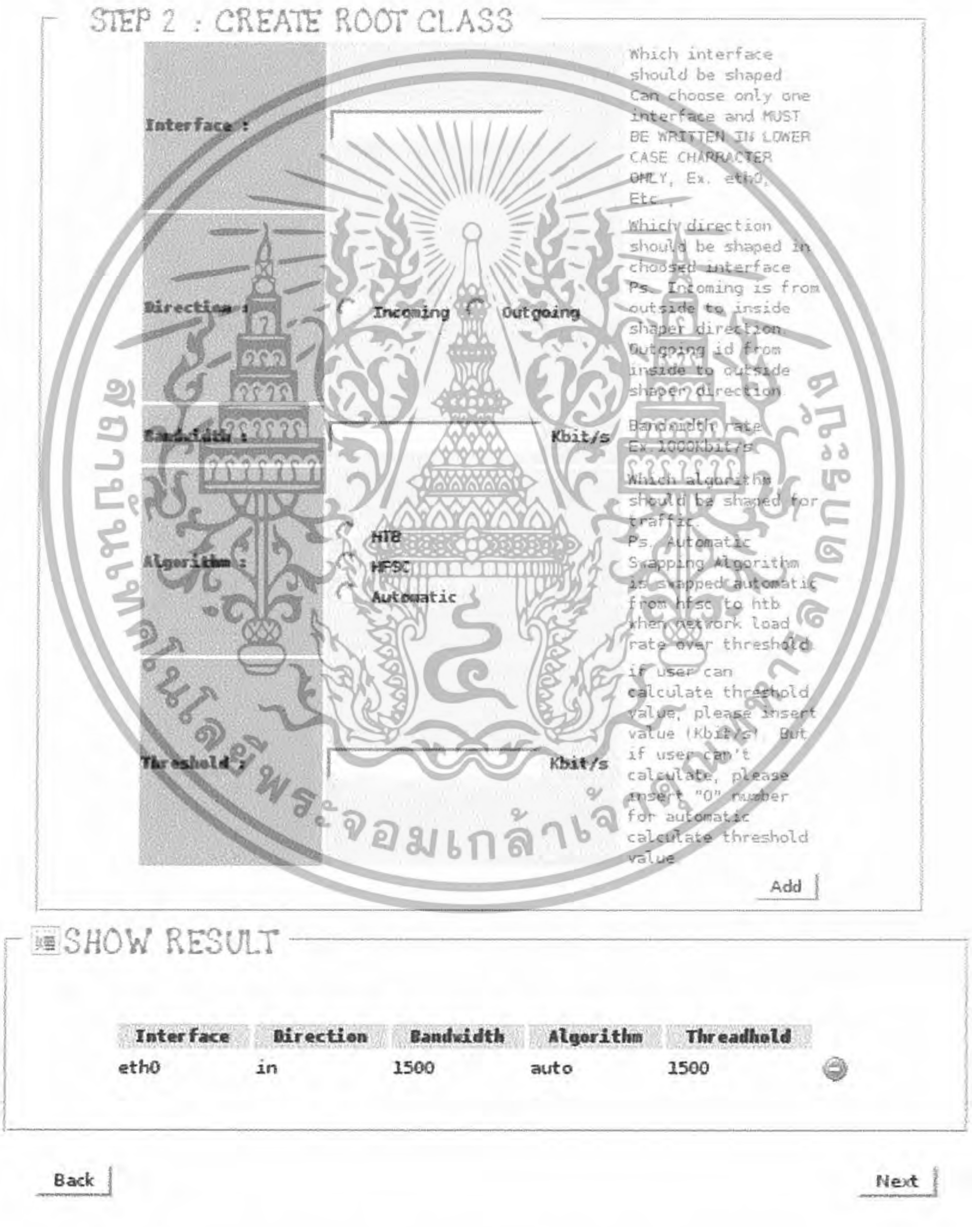


ภาพที่ 4.28 หน้าต่างแสดงการกำหนดค่าให้กับเครื่องเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 กำหนดค่าข้อมูลของการปรับแต่งทราฟฟิก (Quality of Service)

หน้าต่างสำหรับกำหนดค่าหลักของกฎนโยบายการควบคุมปริมาณทราฟฟิก ได้แก่ชื่ออินเทอร์เฟส ทิศทางของการควบคุมปริมาณทราฟฟิก ขนาดแบนด์วิดท์ ซึ่งค่าของแบนด์วิดท์นี้จะใช้เป็นค่าแบนด์วิดท์สูงสุดที่ใช้ในการกำหนดค่ากฎควบคุมปริมาณทราฟฟิก อัลกอลิทึมที่ใช้ในการควบคุมปริมาณทราฟฟิก และ ค่าขอบเขตการสลับอัลกอลิทึม

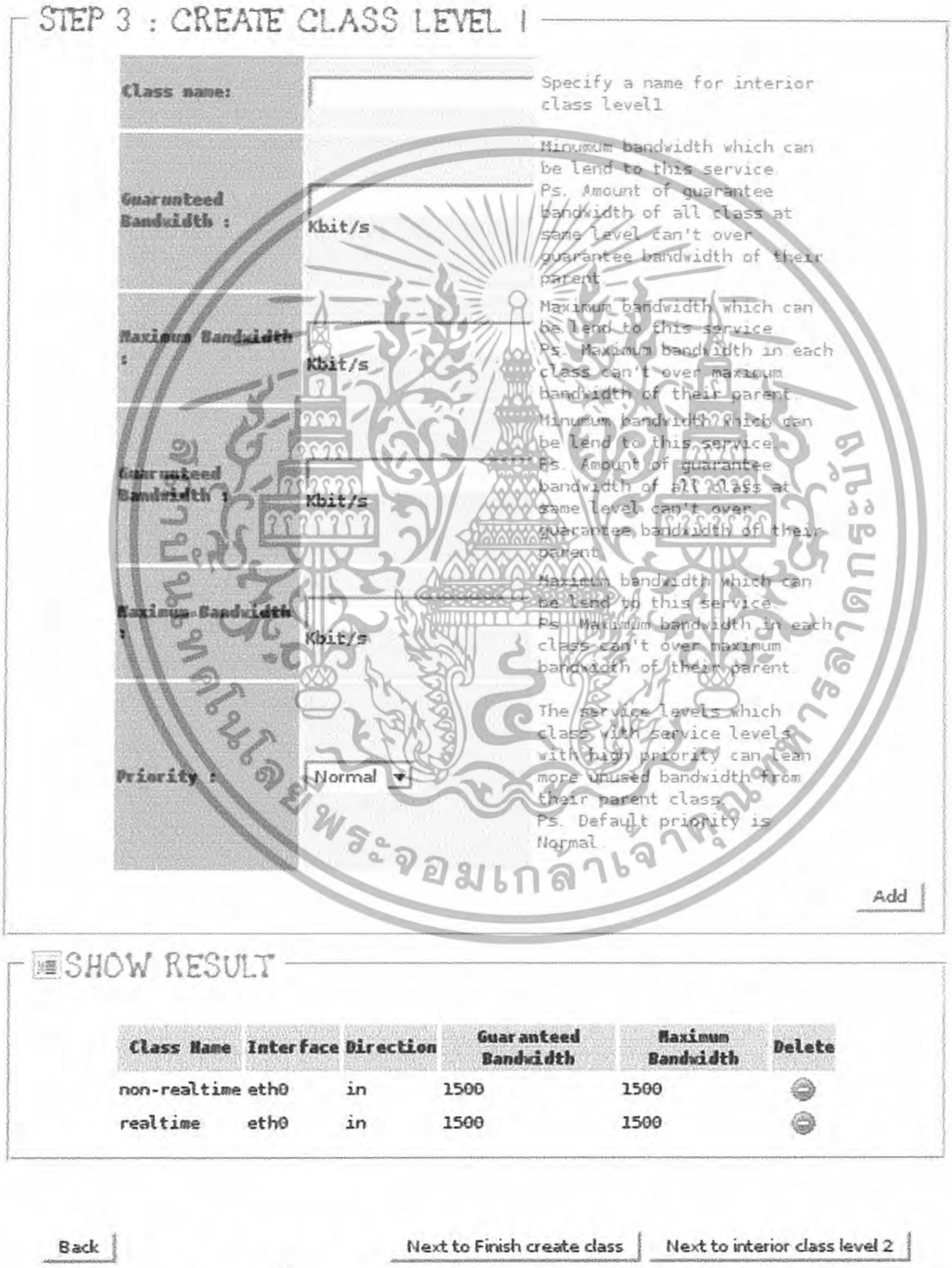


ภาพที่ 4.29 หน้าต่างแสดงการกำหนดค่าข้อมูลของการปรับแต่งทราฟฟิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

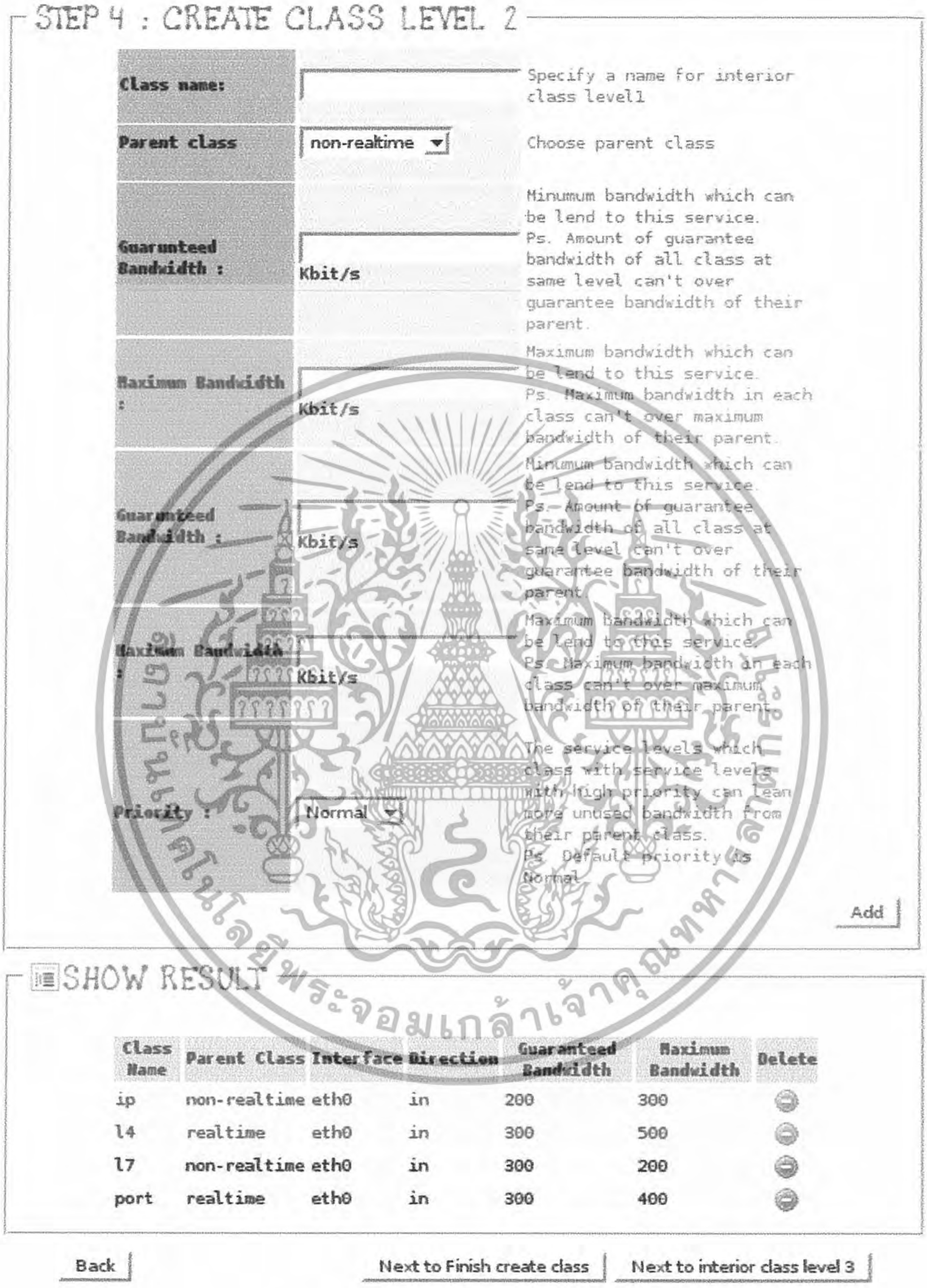
1.3 กำหนดค่าของคลาสในระดับต่างๆ (Hierarchical Class Setting)

หน้าต่างสำหรับกำหนดค่าของคลาสในระดับต่างๆ โดยสามารถกำหนดสูงสุดได้ 3 ระดับ โดยทำการกำหนดค่า ดังนี้ ชื่อคลาส แบนด์วิดท์ต่ำสุดที่รับประกัน แบนด์วิดท์สูงสุดที่กำหนด ค่าลำดับความสำคัญ และ ชนิดการกรองทราฟฟิกของคลาสนั้นๆ



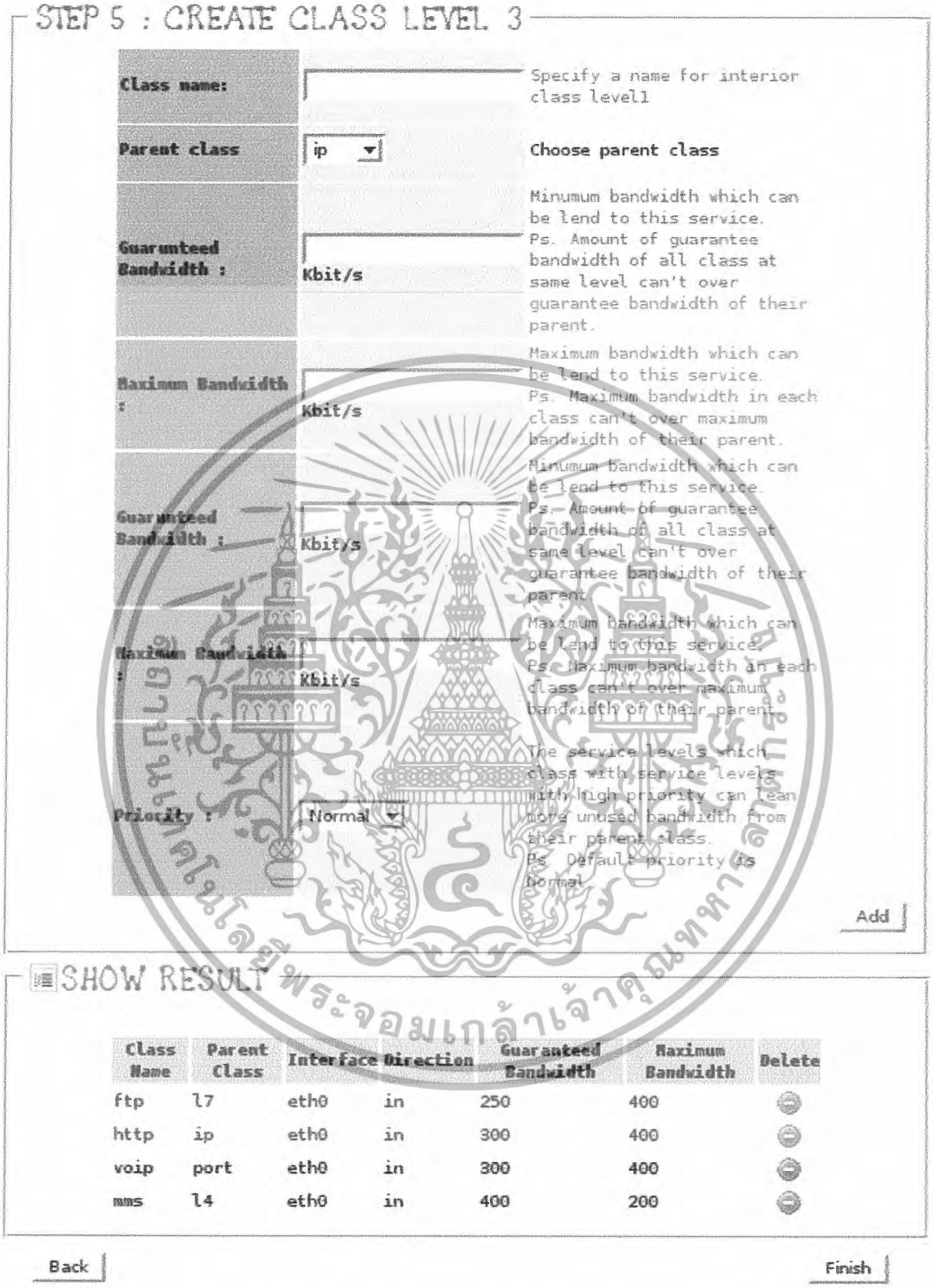
ภาพที่ 4.30 หน้าต่างแสดงการกำหนดค่าของคลาสระดับที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.31 หน้าต่างแสดงการกำหนดค่าของคลาสระดับที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.32 หน้าต่างแสดงการกำหนดค่าของคลาสระดับที่ 3

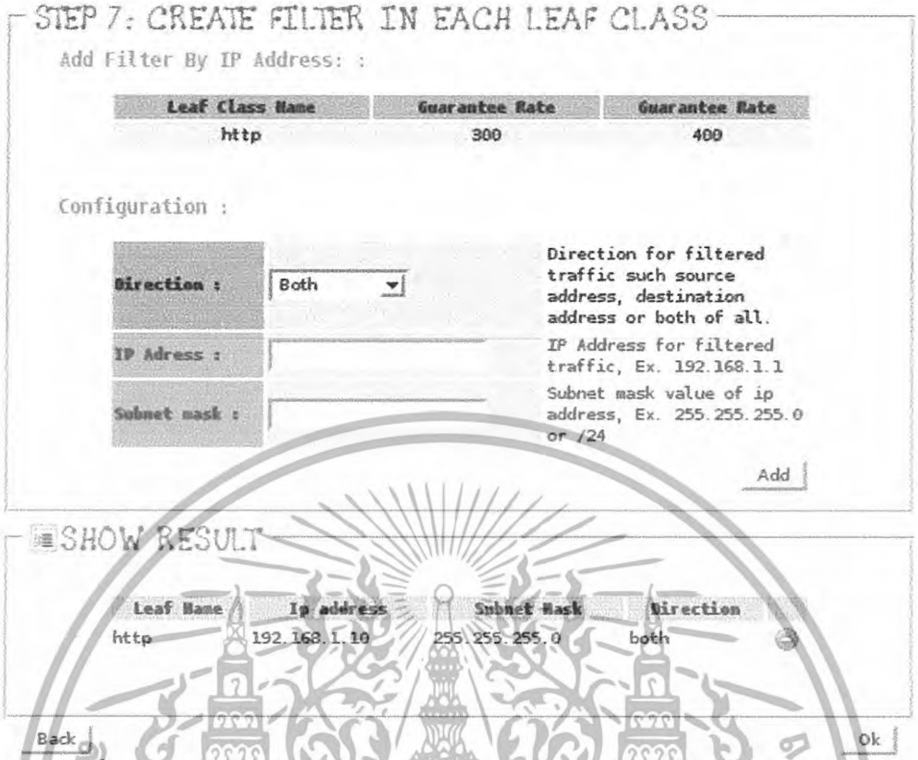
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 กำหนดชนิดการกรองทราฟฟิกของคลาส (Filter)

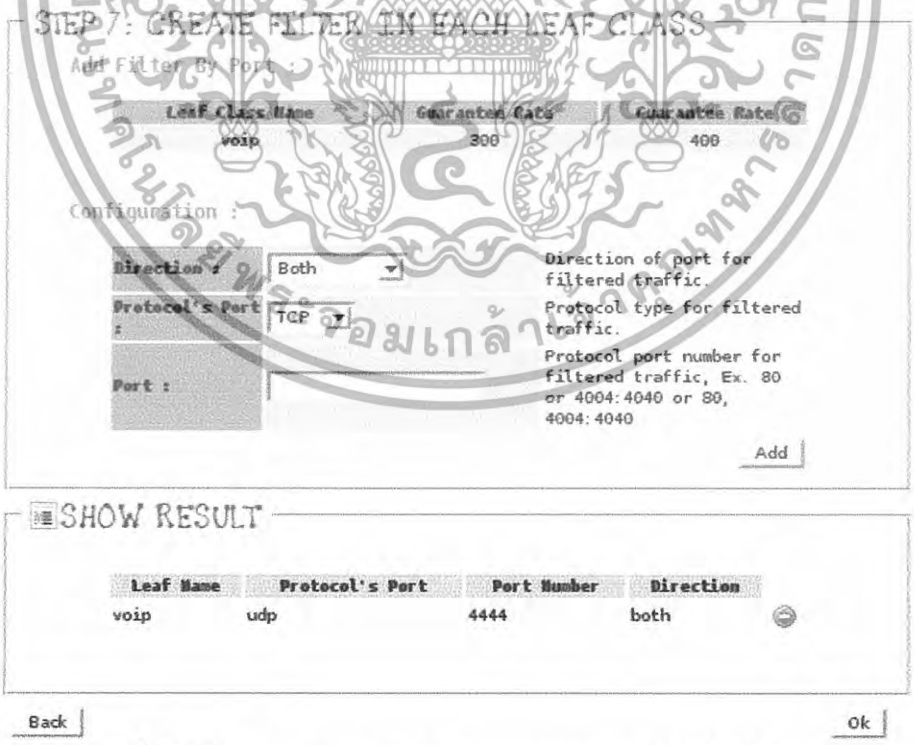
หน้าต่างสำหรับกำหนดชนิดการกรองทราฟฟิกของคลาส โดยทำการแสดงข้อมูลของคลาสสุดท้าย (Leaf Class) ทั้งหมดในกฎนโยบายของการควบคุมปริมาณทราฟฟิก โดยคลาสสามารถกำหนดชนิดการกรองทราฟฟิกได้เพียงชนิดเดียวจาก 4 ชนิด คือ การกรองทราฟฟิกโดยใช้พอร์ต การกรองทราฟฟิกโดยใช้ไอพีแอดเดรส การกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นขนส่งข้อมูล และ การกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นแอพพลิเคชั่น

ภาพที่ 4.33 หน้าต่างแสดงข้อมูลคลาสสุดท้ายทั้งหมด
ในกฎนโยบายของการควบคุมปริมาณทราฟฟิกเพื่อกำหนดชนิดการกรองทราฟฟิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

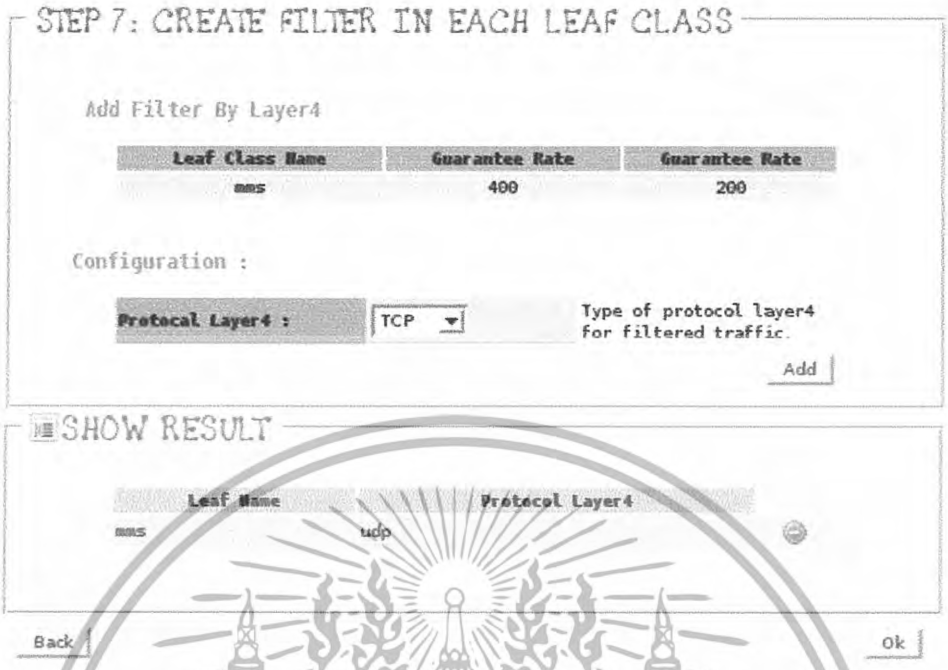


ภาพที่ 4.34 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้ไอพีแอดเดรส

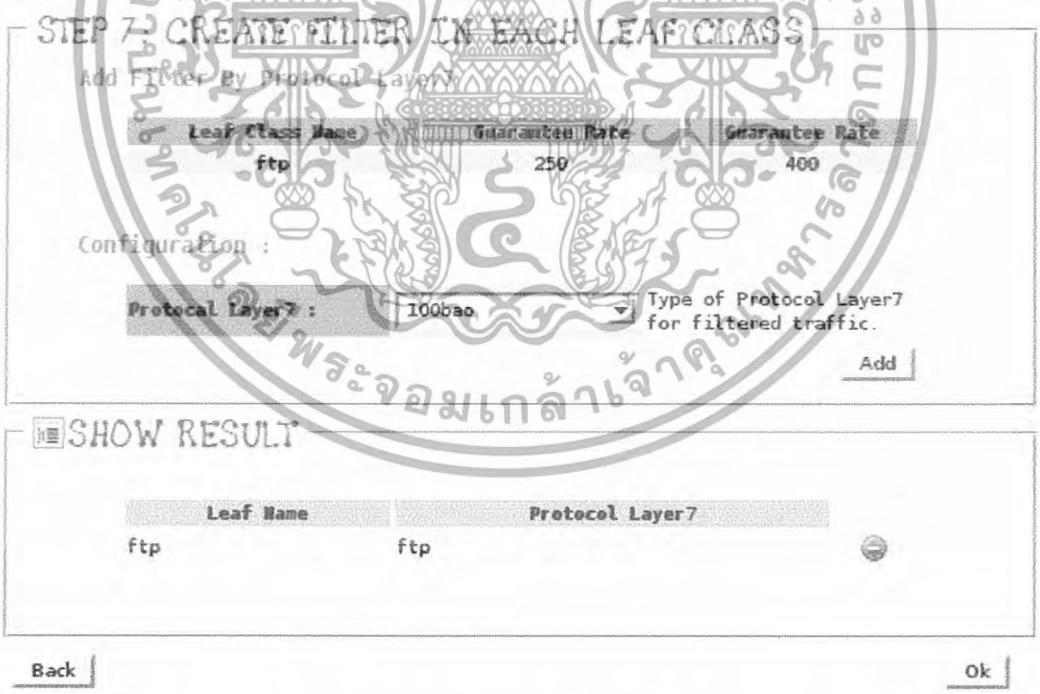


ภาพที่ 4.35 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้พอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.36 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นขนส่งข้อมูล



ภาพที่ 4.37 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นแอพพลิเคชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

STEP 8: CONCLUDE CONFIGURATION CLASS

Root Qdisc :

Interface	Direction	Algorithm	Bandwidth	Threadhold
eth0	in	auto	1500	1500

Class :

Class Name	Guarantee Rate	Max Rate	Filter Type	Filter Description
non-realtime	1500	1500		
l7	300	200		
ftp	250	400	l7	filtered by ftp
ip	200	300		
http	300	400	ip	both direction, ip address 192.168.1.10/255.255.255.0
realtime	1500	1500		
port	300	400		
voip	300	400	port	both direction, protocol udp(port 4444)
l4	300	500		
mms	400	200	l4	filtered by udp

Back | Finish | Cancle

ภาพที่ 4.38 หน้าต่างแสดงสรุปกฎนโยบายของการควบคุมปริมาณทราฟฟิก

2. **ส่วนการแสดงและแก้ไขกฎนโยบายของการควบคุมปริมาณทราฟฟิก (Edit Policy)**

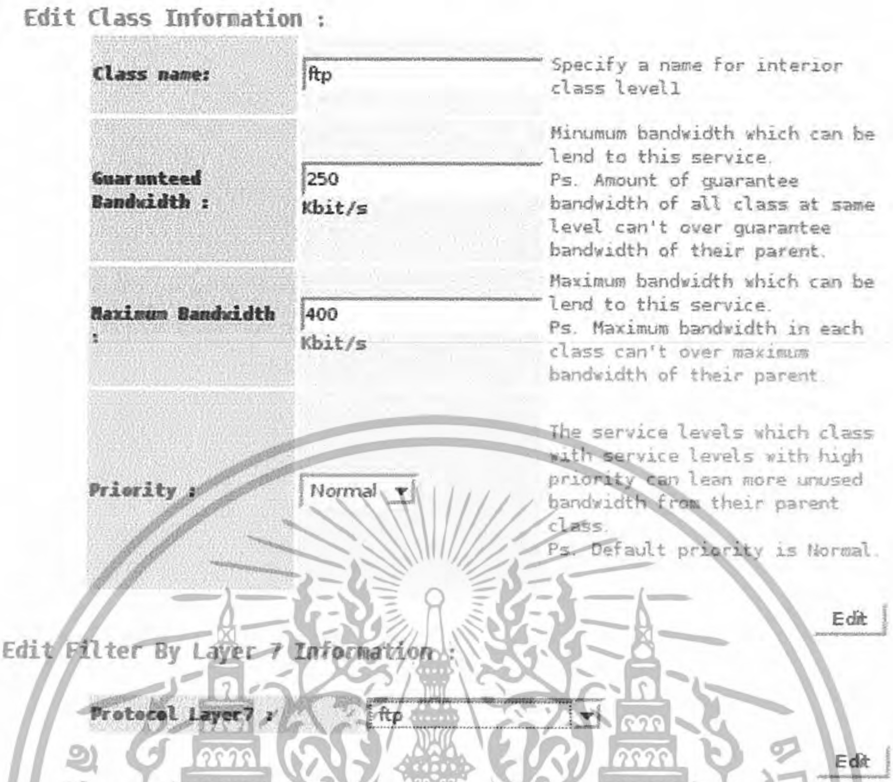
eth0 (Inbound) Information

Interface	Direction	Algorithm	Bandwidth	Threadhold	Edit	Delete
eth0	Inbound	auto	1500	1500		

Class Name	Guarantee Rate	Max Rate	Filter Type	Filter Description	Edit
non-realtime	1500	1500			
l7	300	200			
ftp	250	400	l7	filtered by ftp	
ip	200	300			
http	300	400	ip	both direction, ip address 192.168.1.10/255.255.255.0	
realtime	1500	1500			
port	300	400			
voip	300	400	port	both direction, protocol udp(port 4444)	
l4	300	500			
mms	400	200	l4	filtered by udp	
default	75	75			

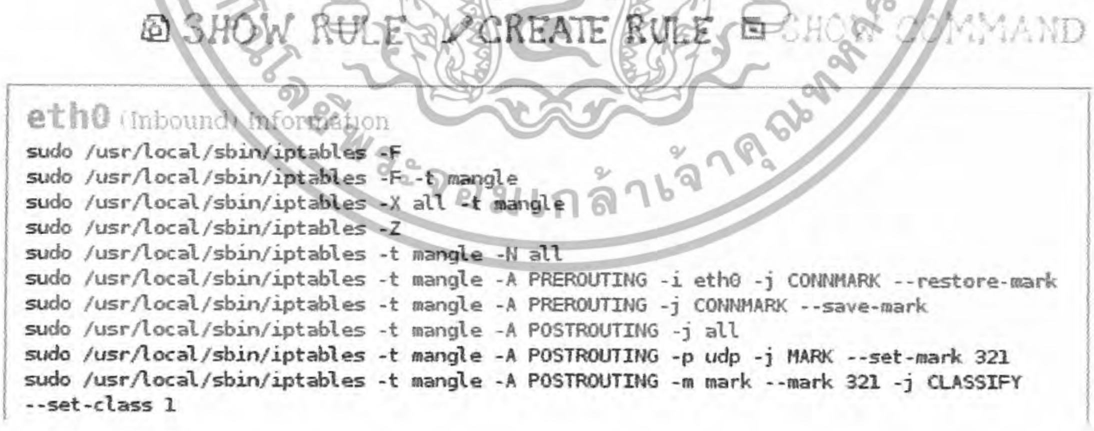
ภาพที่ 4.39 หน้าต่างแสดงและแก้ไขกฎนโยบายของการควบคุมปริมาณทราฟฟิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 4.40 หน้าต่างแก้ไขข้อมูลคลาสที่กำหนดการกรองทราฟฟิก

3. ส่วนการแสดงชุดคำสั่งกฎนโยบายของการควบคุมปริมาณทราฟฟิก




```
sudo /usr/local/sbin/iptables -F
sudo /usr/local/sbin/iptables -F -t mangle
sudo /usr/local/sbin/iptables -X all -t mangle
sudo /usr/local/sbin/iptables -Z
sudo /usr/local/sbin/iptables -t mangle -N all
sudo /usr/local/sbin/iptables -t mangle -A PREROUTING -i eth0 -j CONNMARK --restore-mark
sudo /usr/local/sbin/iptables -t mangle -A PREROUTING -j CONNMARK --save-mark
sudo /usr/local/sbin/iptables -t mangle -A POSTROUTING -j all
sudo /usr/local/sbin/iptables -t mangle -A POSTROUTING -p udp -j MARK --set-mark 321
sudo /usr/local/sbin/iptables -t mangle -A POSTROUTING -m mark --mark 321 -j CLASSIFY
--set-class 1
```

ภาพที่ 4.41 หน้าต่างแสดงชุดคำสั่งกฎนโยบายของการควบคุมปริมาณทราฟฟิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

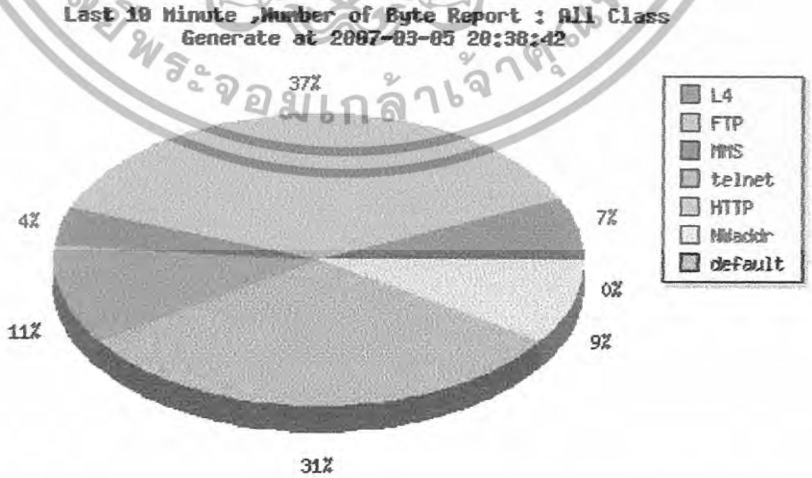
4.10.2.4 การรายงานสถิติการใช้งานทางเครือข่าย (Monitor)

ส่วนหน้าต่างเหล่านี้จะเป็นตัวรายงานสถิติการใช้งานทางเครือข่าย โดยประกอบข้อมูล 2 ส่วนหลักๆ คือ การดูรายงานสถิติการใช้บริการทางเครือข่ายตามคลาสที่ได้จากการทำงานชุดคำสั่ง TC และการดูรายงานสถิติการใช้บริการทางเครือข่ายโดยภายรวมของแต่ละอินเทอร์เฟส

การรายงานสถิติการใช้งานทางเครือข่ายสามารถเลือกแสดงเป็นช่วงเวลาได้ คือ รายสิบนาทีล่าสุด รายหนึ่งชั่วโมงล่าสุด รายสี่ชั่วโมงล่าสุด รายวัน รายอาทิตย์ รายเดือน และ รายปี

1. **รายงานสถิติการใช้บริการทางเครือข่ายตามคลาส**

ภาพที่ 4.42 กราฟเส้นแสดงปริมาณไบต์ข้อมูลในเครือข่ายตามคลาส

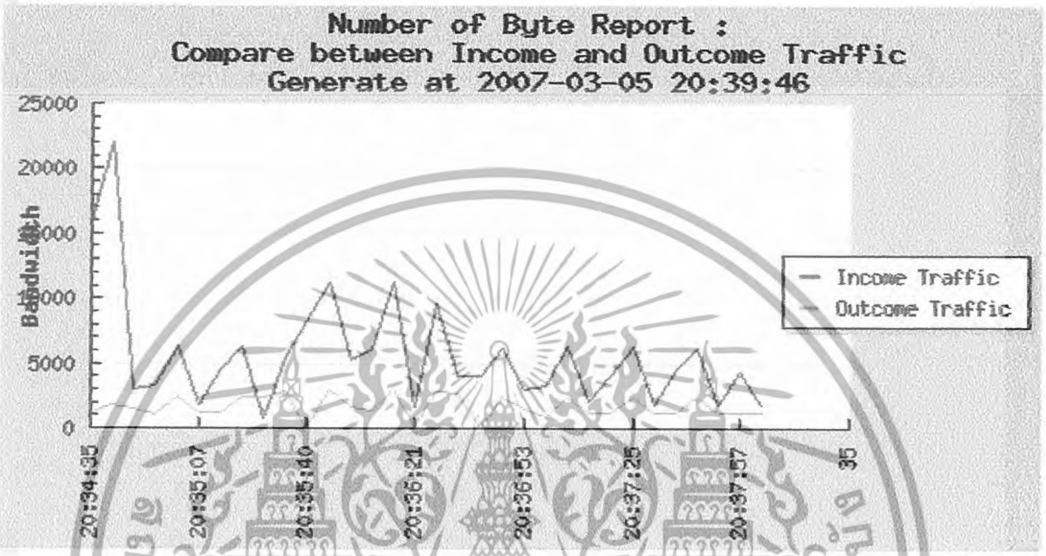


ภาพที่ 4.43 กราฟวงกลมแสดงปริมาณไบต์ข้อมูลในเครือข่ายตามคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. รายงานสถิติการใช้บริการทางเครือข่ายโดยภาพรวม

ส่วนรายงานสถิติเหล่านี้จะแสดงการเปรียบเทียบผลรวมการใช้งานเครือข่ายโดยภาพรวมระหว่างปริมาณทราฟฟิกขาเข้าและปริมาณทราฟฟิกขาออก

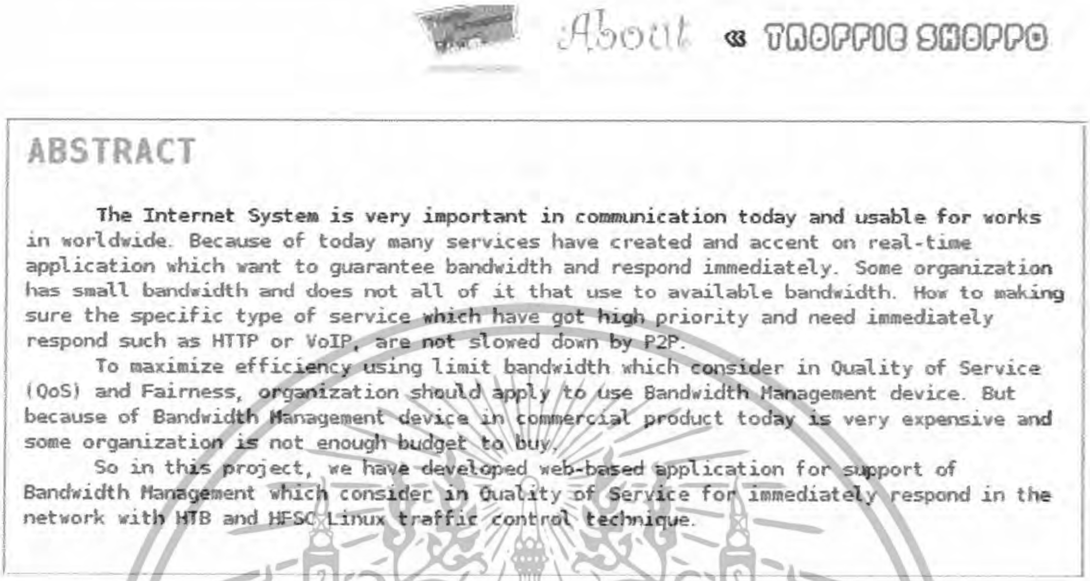


ภาพที่ 4.44 กราฟเส้นแสดงปริมาณไบต์ข้อมูลในเครือข่ายโดยภาพรวม

ภาพที่ 4.45 กราฟวงกลมแสดงปริมาณไบต์ข้อมูลในเครือข่ายโดยภาพรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.10.2.5 แสดงบทคัดย่อของโครงงาน (About)



ABSTRACT

The Internet System is very important in communication today and usable for works in worldwide. Because of today many services have created and accent on real-time application which want to guarantee bandwidth and respond immediately. Some organization has small bandwidth and does not all of it that use to available bandwidth. How to making sure the specific type of service which have got high priority and need immediately respond such as HTTP or VoIP, are not slowed down by P2P.

To maximize efficiency using limit bandwidth which consider in Quality of Service (QoS) and Fairness, organization should apply to use Bandwidth Management device. But because of Bandwidth Management device in commercial product today is very expensive and some organization is not enough budget to buy.

So in this project, we have developed web-based application for support of Bandwidth Management which consider in Quality of Service for immediately respond in the network with HTB and HFSC Linux traffic control technique.

ภาพที่ 4.46 แสดงรายงานบทคัดย่อ โครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การพัฒนาและทดสอบระบบ

5.1 เครื่องมือที่ใช้ในการพัฒนาระบบ

เครื่องมือที่ใช้ในการพัฒนาระบบ แบ่งออกเป็น ดังนี้

5.1.1 ฮาร์ดแวร์

ได้แก่ เครื่องคอมพิวเตอร์จำนวน 8 เครื่อง เพื่อใช้เป็นเครื่องลินุกซ์เกตเวย์ จำนวน 1 เครื่อง และใช้เป็นเครื่องไคลเอนท์และเซิร์ฟเวอร์ จำนวน 7 เครื่อง โดยมีคุณสมบัติดังนี้

- ซีพียู เพนเที่ยม 4 ความเร็ว 1.60 กิกกะเฮิร์ต
- ฮาร์ดดิสก์ ความจุ 20 กิกกะไบต์
- หน่วยความจำ ขนาด 512 เมกกะไบต์

5.1.2 ซอฟต์แวร์

ซอร์ฟแวร์ที่ใช้พัฒนาระบบ ใช้ในการทำงานระบบ และ ใช้ในการทดลองระบบ มีรายชื่อดังต่อไปนี้

1. Linux Ubuntu 5.10 (Brezzy Bradger)

เครื่องลินุกซ์เกตเวย์ที่ทำหน้าที่เป็นเครื่องที่ติดตั้งระบบและทำการควบคุมปริมาณทราฟฟิก โดยทำการคอมไพล์เคอร์เนลใหม่เป็นรุ่น 2.6.15.4 เพื่อสนับสนุนการทำงานระดับชั้นแอพพลิเคชั่น

2. Microsoft Windows XP Professional

เครื่องไคลเอ็นท์ที่ทำหน้าที่จำลองสร้างทราฟฟิกจากเครือข่ายภายใน (Internal Network) ออกไปยังเครือข่ายภายนอก (External Network) และจำลองเครื่องเซิร์ฟเวอร์สำหรับให้บริการเอฟทีพี (FTP)

3. Microsoft Server 2003 Enterprise Edition

เครื่องเซิร์ฟเวอร์สำหรับให้บริการสารสื่อประสม (MMS)

4. Apache Web Server

โปรแกรมเครื่องแม่ข่าย (Web Server) ใช้สำหรับให้บริการเว็บ ได้แก่ กำหนดค่ากฎนโยบายควบคุมปริมาณทราฟฟิก เก็บข้อมูล และ แสดงผลข้อมูลสถิติการใช้งานเครือข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. PHP

ภาษาที่ใช้ในการพัฒนาระบบเว็บแอพพลิเคชั่น

6. iproute2

ชุดโปรแกรมคำสั่งการทำงานด้านเครือข่าย

7. L7 Filtering

ชุดคำสั่งการกรองทราฟฟิกที่รองรับการทำงานที่ระดับแอพพลิเคชั่น

8. L7-Protocols

โมดูลที่รวบรวมข้อมูลที่ใช้ในการกรองทราฟฟิกระดับแอพพลิเคชั่น

9. Kernel's Patch for L7 Filtering

โปรแกรมเพิ่มที่ให้เคอร์เนลรองรับการกรองทราฟฟิกระดับแอพพลิเคชั่น

10. iptables

ชุดคำสั่งไฟล์วอล

11. iptables's Patch for L7 Filtering

โปรแกรมแก้ไขที่ทำให้ไอพีเทเบิลรองรับการกรองทราฟฟิกระดับแอพพลิเคชั่น

12. JPGraph

เครื่องมือใช้ในการสร้างกราฟเส้นและกราฟวงกลมแสดงสถิติการใช้งานเครือข่าย

13. Ethereal หรือ Wireshark

โปรแกรมดักจับและวิเคราะห์ทราฟฟิก

14. MySql Server

โปรแกรมฐานข้อมูล ใช้เก็บข้อมูลต่างๆ ของระบบ

15. PhpMyAdmin

โปรแกรมจัดการฐานข้อมูลผ่านระบบเว็บไซต์

16. p2p VoIp, Beta 1.1

โปรแกรมวอยซ์โอเวอร์ไอพี

17. Lan Traffic V2 และ Ip Traffic

โปรแกรมสร้างทราฟฟิก

18. VMware Workstation

โปรแกรมจำลองระบบปฏิบัติการลินุกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 การทดสอบการทำงานระบบ

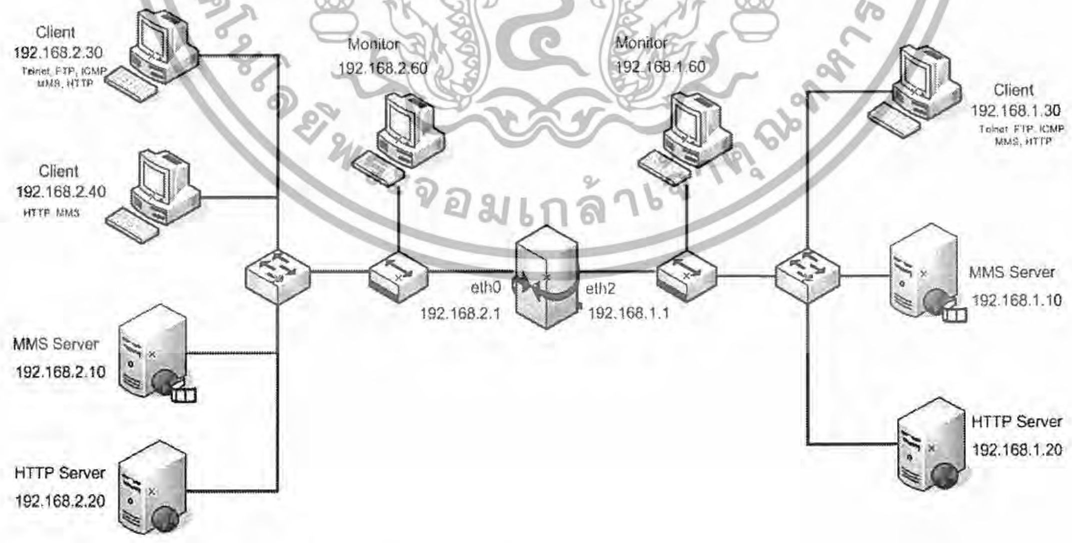
5.2.1 จุดประสงค์ของการทดสอบการทำงานระบบ

- แสดงให้เห็นว่าระบบสามารถทำการกรองทราฟฟิกโดยการกำหนดการกรองด้วย ไอพีแอดเดรส พอร์ต โปรโตคอลระดับชั้นขนส่งข้อมูล และ ระดับชั้นแอพพลิเคชั่น
- แสดงให้เห็นว่าระบบสามารถควบคุมปริมาณทราฟฟิกได้ทั้ง 2 อินเทอร์เฟสพร้อมกัน
- แสดงให้เห็นว่าระบบสามารถควบคุมปริมาณทราฟฟิกได้ด้วยอัลกอลิทึมแบบ HTB, HFSC และ Auto Swapping Algorithm

5.2.2 เครื่องมือที่ใช้ในการทดสอบการทำงานระบบ

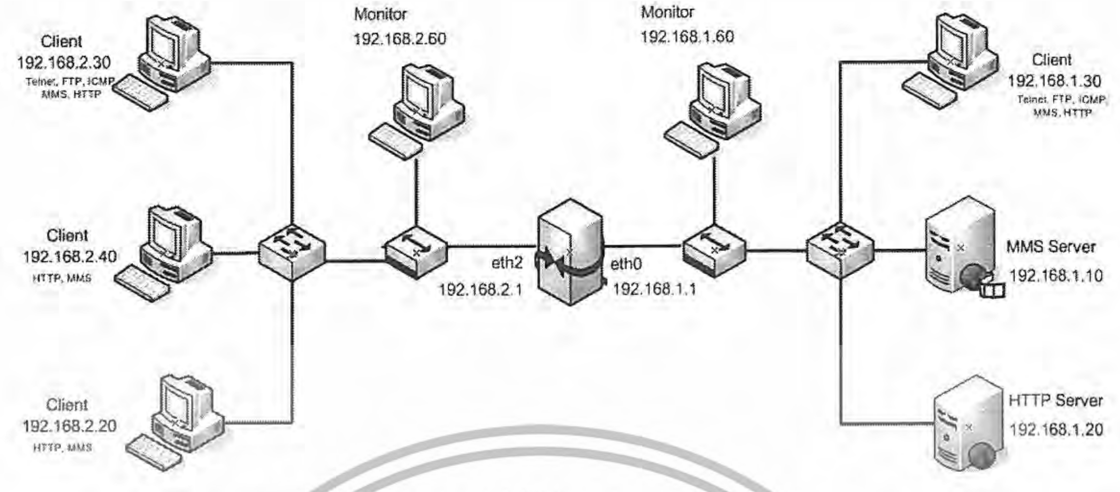
- เครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์ จำนวน 1 เครื่อง
- เครื่องคอมพิวเตอร์ จำนวน 8 เครื่อง
- เครื่องเซิร์ฟเวอร์ จำนวน 8 เครื่อง
- ฮับ (SMC EtherEZ hub 10BASE-T port) จำนวน 2 เครื่อง
- สวิตช์ (TP-link 10/100 Fast Ethernet Switch) จำนวน 2 เครื่อง

5.2.3 แผนภาพจำลองการทดสอบการทำงานระบบ



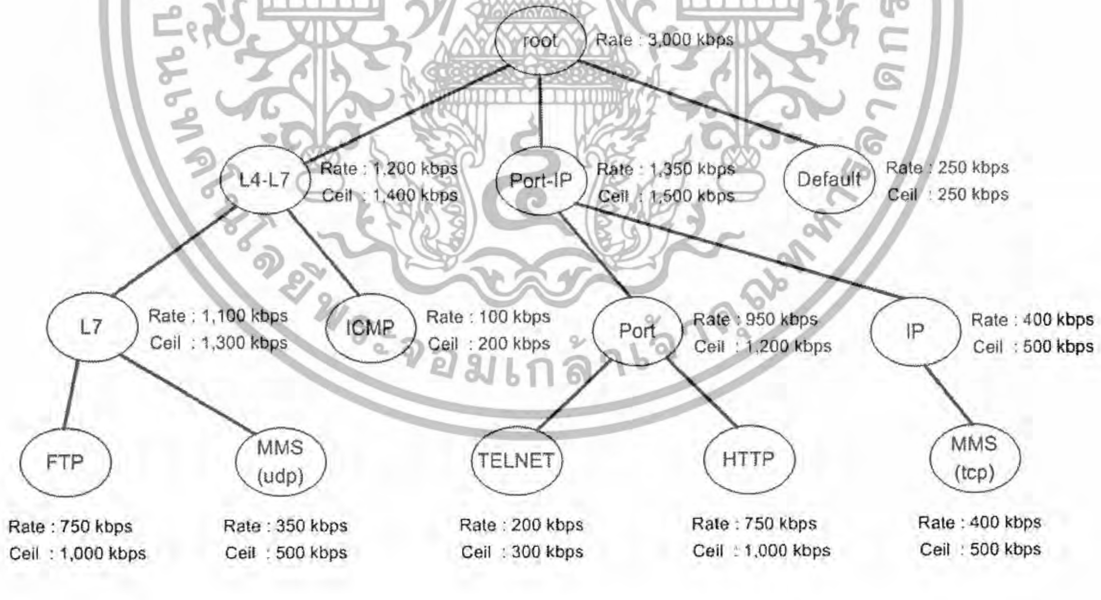
ภาพที่ 5.1 แผนภาพจำลองเครือข่ายของการทดสอบการทำงานระบบ กรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 5.2 แผนภาพจำลองเครือข่ายของการทดสอบการทำงานระบบ
กรณีควบคุมปริมาณทราฟฟิก 1 อินเทอร์เฟส

5.2.4 กฎนโยบายควบคุมปริมาณทราฟฟิกในการทดสอบการทำงานระบบ



ภาพที่ 5.3 กฎนโยบายควบคุมปริมาณทราฟฟิกในการทดสอบการทำงานระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทดสอบกำหนดกฎนโยบายควบคุมปริมาณทราฟฟิกดังนี้

- กำหนดให้บริการ FTP จากเครื่องใดๆ ทำการกรองทราฟฟิกด้วยโปรโตคอลระดับชั้นแอพพลิเคชั่น รับประกันแบนด์วิดท์ขั้นต่ำที่ 750 กิโลบิตต่อวินาทีและจำกัดแบนด์วิดท์ที่ 1,000 กิโลบิตต่อวินาที
- กำหนดให้บริการ MMS จากเครื่องใดๆ ทำการกรองทราฟฟิกด้วยพอร์ตที่มีการรับส่งข้อมูลด้วยโปรโตคอลแบบ UDP ทำงานที่หมายเลขพอร์ตช่วง 1755, 5005, 1024:4443, 4445:5500 รับประกันแบนด์วิดท์ขั้นต่ำที่ 350 กิโลบิตต่อวินาทีและจำกัดแบนด์วิดท์ที่ 500 กิโลบิตต่อวินาที
- กำหนดให้บริการ Telnet จากเครื่องใดๆ ทำการกรองทราฟฟิกด้วยพอร์ตที่มีการรับส่งข้อมูลด้วยโปรโตคอลแบบ TCP ทำงานที่หมายเลขพอร์ตช่วง 23 รับประกันแบนด์วิดท์ขั้นต่ำที่ 200 กิโลบิตต่อวินาทีและจำกัดแบนด์วิดท์ที่ 300 กิโลบิตต่อวินาที
- กำหนดให้บริการ ICMP จากเครื่องใดๆ ทำการกรองทราฟฟิกด้วยโปรโตคอลระดับชั้นขนส่งข้อมูล รับประกันแบนด์วิดท์ขั้นต่ำที่ 100 Kbps และ จำกัดแบนด์วิดท์ที่ 200 กิโลบิตต่อวินาที
- กำหนดให้บริการ HTTP จากเครื่องใดๆ ทำการกรองทราฟฟิกด้วยพอร์ตที่มีการรับส่งข้อมูลด้วยโปรโตคอลแบบ TCP ทำงานที่หมายเลขพอร์ต 80 รับประกันแบนด์วิดท์ขั้นต่ำที่ 750 กิโลบิตต่อวินาทีและจำกัดแบนด์วิดท์ที่ 1,000 กิโลบิตต่อวินาที
- กำหนดให้บริการ MMS จากเครื่องใดๆ ทำการกรองทราฟฟิกด้วยไอพีแอดเดรส 192.168.2.0/24 ซึ่งทำการกรองทราฟฟิกทั้งหมดที่เข้ามายังเน็ตเวิร์คหมายเลข 192.168.2.0 ที่ไม่ใช่ทราฟฟิกของคลาสบริการทั้ง 5 คลาสที่ได้กำหนดไว้ โดยจากการทดลองทราฟฟิกที่เข้ามายังคลาสบริการดังกล่าวเป็นทราฟฟิกบริการ MMS ที่มีการรับส่งข้อมูลด้วยโปรโตคอล TCP ทำงานด้วยพอร์ต 554 รับประกันแบนด์วิดท์ขั้นต่ำที่ 400 กิโลบิตต่อวินาทีและจำกัดแบนด์วิดท์ที่ 500 กิโลบิตต่อวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.5 ขั้นตอนการการทดสอบการทำงานระบบตามกฎควบคุมปริมาณทราฟฟิก

5.2.5.1 การทดสอบการกรองและการควบคุมปริมาณทราฟฟิกพร้อมกัน 2 อินเทอร์เฟส

เริ่มจากการกำหนดกฎนโยบายการควบคุมปริมาณทราฟฟิกให้กับทั้ง 2 อินเทอร์เฟสด้วยนโยบายเดียวกันตามภาพที่ 5.3 แล้วทำการทดลองด้วยการควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึมแบบ HFSC แล้วทำการสร้างทราฟฟิกดังนี้

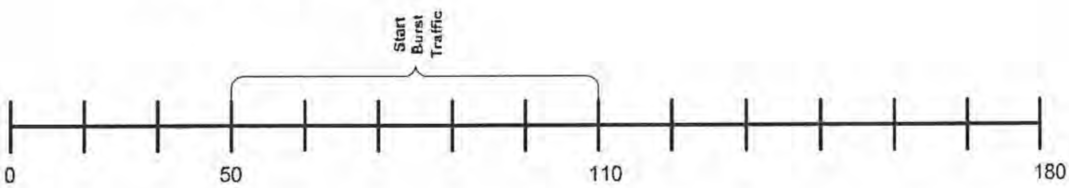
ตารางที่ 5.1 แสดงปริมาณทราฟฟิกที่ใช้ในการทดสอบการกรองและการควบคุมปริมาณทราฟฟิกพร้อมกัน 2 อินเทอร์เฟส

อัตราส่วน TCP:UDP		อัตราแบนด์วิดท์ของแต่ละคลาสตลอดการทดลอง						อัตราแบนด์วิดท์ที่เพิ่มช่วงวินาทีที่ 50 – 110	
TCP (Kbps)	UDP (Kbps)	FTP (Kbps)	NWaddr (Kbps)	ICMP (Kbps)	HTTP (Kbps)	MMS (Kbps)	TELNET (Kbps)	FTP (Mbps)	MMS (Mbps)
70	30	1000	-	100	750	350	200	5	2.45

5.2.5.2 การทดสอบการควบคุมปริมาณทราฟฟิกได้ด้วยอัลกอลิทึมแบบ HTB, HFSC และ Auto Swapping Algorithm

ในการการทดสอบการทำงานระบบจะแบ่งการทดสอบออกเป็น 3 การทดลองหลัก ซึ่งทั้ง 3 การทดลองนี้จะใช้กฎนโยบายในการควบคุมปริมาณทราฟฟิก มีขั้นตอนและทำการทดลองภายใต้สภาวะแวดล้อมเดียวกันทุกประการ โดยทั้ง 3 การทดลองนั้นแตกต่างกันดังนี้

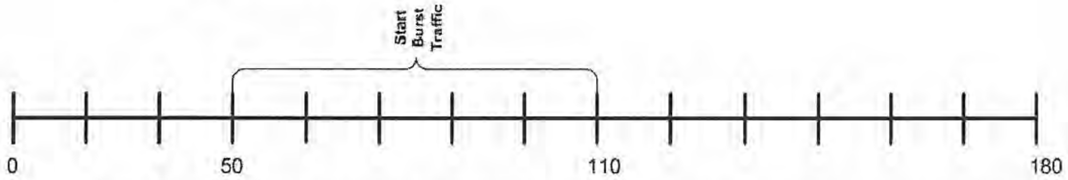
- ในการทดลองที่ 1 เราจะใช้การควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึมแบบ HTB ตลอดการทดลอง



ภาพที่ 5.4 ขั้นตอนการการทดสอบการทำงานระบบด้วยการใช้อัลกอลิทึมแบบ HTB

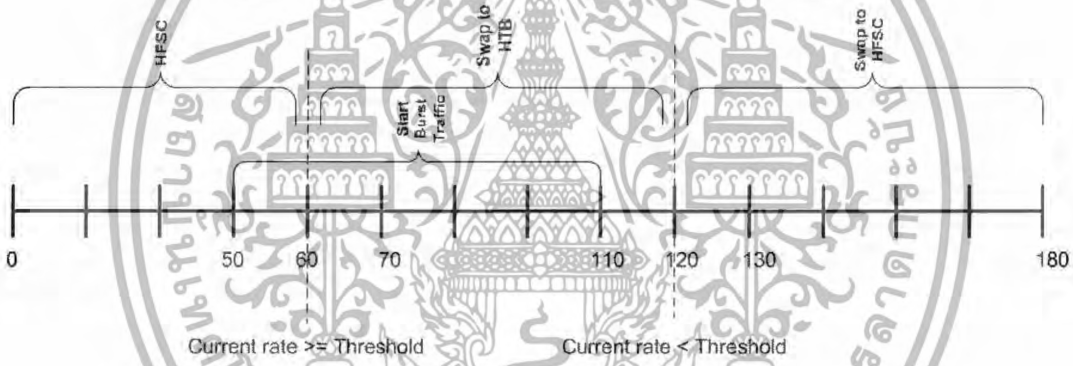
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ในการทดลองที่ 2 เราจะใช้การควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึมแบบ HFSC ตลอดการทดลอง



ภาพที่ 5.5 ขั้นตอนการการทดสอบการทำงานระบบด้วยการใช้อัลกอลิทึมแบบ HFSC

- ในการทดลองที่ 3 เราจะใช้การควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึมแบบ Automatic Swapping Algorithm ซึ่งในแต่ละช่วงของการทดลองนั้นจะมีการใช้อัลกอลิทึมแบบ HTB และ HFSC สลับกันไป



ภาพที่ 5.6 ขั้นตอนการการทดสอบการทำงานระบบด้วยการใช้อัลกอลิทึมแบบ Auto Swapping

โดยในแต่ละการทดลองหลักนั้นจะแบ่งออกเป็น 11 การทดลองย่อย ซึ่งมีการแบ่งการทดลองตามอัตราส่วนการใช้งานเครือข่ายด้วยโปรโตคอลระดับชั้นขนส่งข้อมูล คือ โปรโตคอล TCP และ UDP ดังตารางที่ 5.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.2 ตารางแสดงอัตราการใช้งานแบนด์วิดท์ในการทดสอบแอพพลิเคชั่น

อัตราส่วน TCP:UDP		**อัตราแบนด์วิดท์ของแต่ละคลาสตลอดการทดลอง**						**อัตราแบนด์วิดท์ที่เพิ่มช่วงวินาทีที่ 50 – 110**	
TCP (Kbps)	**UDP (Kbps)**	**FTP (Kbps)**	**NWaddr (Kbps)**	**ICMP (Kbps)**	**HTTP (Kbps)**	**MMS (Kbps)**	**TELNET (Kbps)**	**FTP (Mbps)**	**MMS (Mbps)**
0	100	-	-	-	-	350	200	-	9.45
10	90	1000	-	100	300	350	200	-	8.45
20	80	1000	-	100	750	350	200	-	7.45
30	70	1000	-	100	750	350	200	1	6.45
40	60	1000	-	100	750	350	200	2	5.45
50	50	1000	-	100	750	350	200	3	4.45
60	40	1000	-	100	750	350	200	4	3.45
70	30	1000	-	100	750	350	200	5	2.45
80	20	1000	-	100	750	350	200	6	1.45
90	10	1000	-	100	750	350	200	7	0.45
100	0	1000	-	100	750	-	-	8	-

ขั้นตอนในการทดสอบการทำงานระบบมีดังต่อไปนี้

1. ก่อนเริ่มทำการทดลองมีการกำหนดกฎนโยบายควบคุมปริมาณทราฟฟิกลงเครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์ ซึ่งจะเลือกใช้อัลกอลิทึมที่แตกต่างกันไปตามการทดลองหลัก
2. 0 - 10 วินาที เปิดทราฟฟิกพร้อมกันทุกคลาสบริการตามปริมาณในช่องที่แสดงอัตราแบนด์วิดท์ของแต่ละคลาสตลอดการทดลอง ดังตารางด้านบน
3. 20 - 50 วินาที ทราฟฟิกของทุกคลาสบริการอยู่ในระดับคงที่
4. 50 - 60 วินาที เปิดทราฟฟิกแบบเบิร์สเข้าคลาสบริการ FTP และ MMS ตามปริมาณในช่องที่แสดงอัตราแบนด์วิดท์ที่เพิ่มช่วงวินาทีที่ 50 – 110 ดังตารางด้านบน
5. 110 - 120 วินาที หยุดเปิดทราฟฟิกแบบเบิร์สเข้าคลาสบริการ FTP และ MMS
6. 120 - 130 วินาที ปริมาณทราฟฟิกกลับเข้าสู่สภาวะปกติ
7. 130 - 180 วินาที ทราฟฟิกของทุกคลาสบริการอยู่ในระดับคงที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยในการทดลองหลักที่ 3 ที่มีการใช้อัลกอลิทึมแบบ Auto Swapping Algorithm นั้น ในช่วงวินาทีที่ 60 -70 หลังจากที่มีการสร้างทราฟฟิกแบบเบิร์สเข้าไปนั้น จะมีการตรวจจับได้ว่าทราฟฟิกขาเข้านั้นมีปริมาณที่เกินค่า Threshold ของคลาส ที่เกินกว่าปริมาณที่รับประกันขั้นต่ำไว้ 180% ดังนั้นจากเดิมที่ใช้อัลกอลิทึมแบบ HFSC อยู่ จะทำการสลับอัลกอลิทึมไปใช้อัลกอลิทึมแบบ HTB แทน และ เมื่อวินาทีที่ 120 – 130 หลังจากที่เลิกทำการสร้างทราฟฟิกแบบเบิร์สแล้ว จะมีการตรวจจับได้ว่าปริมาณทราฟฟิกขาเข้านั้นมีปริมาณน้อยกว่าค่า Threshold แล้ว จึงทำการสลับอัลกอลิทึมอีกครั้งจาก HTB กลับมาเป็น HFSC ดังเดิม

แต่ในการทดลองหลักที่ 1 และ 2 นั้น จะไม่มีการคำนึงถึงปริมาณทราฟฟิกขาเข้าว่ามากหรือน้อยกว่าค่า Threshold อยู่แล้ว

5.2.6 การทดสอบการทำงานระบบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน

eth0 (Inbound) Information

Interface	Direction	Algorithm	Bandwidth	Threadhold	Edit	Delete
eth0	Inbound	auto	3000	3000		

Class Name	Guarantee Rate	Max Rate	Filter Type	Filter Description	Edit
l4-l7	1200	1400			
l7	1100	1300			
ftp	750	1000	l7	filtered by ftp	
mms(udp)	350	500	port	both direction, protocol udp(port 1755, 5005, 1024:4443, 4445:5500)	
icmp	100	200	l4	filtered by icmp	
port-ip	1350	1500			
port	950	1200			
telnet	200	300	port	both direction, protocol tcp(port 23)	
http	750	1000	port	both direction, protocol tcp(port 80)	
ip	400	500			
mms(tcp)	400	500	ip	both direction, ip address 192.169.2.0/255.255.255.0	
default	150	150			

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

eth2 (Inbound) Information

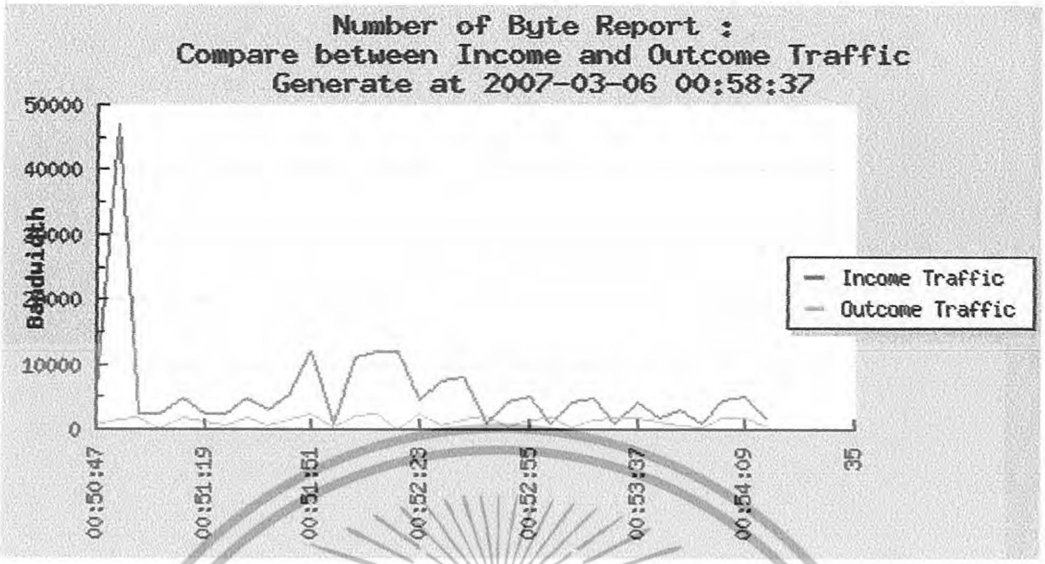
Interface	Direction	Algorithm	Bandwidth	Threadhold	Edit	Delete
eth2	Inbound	auto	3000	3000		

Class Name	Guarantee Rate	Max Rate	Filter Type	Filter Description	Edit
l4-l7	1200	1400			
l7	1100	1300			
ftp	750	1000	l7	filtered by ftp	
mms(udp)	350	500	port	both direction, protocol udp(port 1755,5005,1024:4443,4445:5500)	
icmp	100	200	l4	filtered by icmp	
port-ip	1350	1500			
port	950	1200			
telnet	200	300	port	both direction, protocol tcp(port 23)	
http	750	1000	port	both direction, protocol tcp(port 80)	
ip	400	500			
mms(tcp)	400	500	ip	both direction, ip address 192.169.2.0/255.255.255.0	
default	150	150			

ภาพที่ 5.7 แสดงกฎนโยบายควบคุมปริมาณทราฟฟิก
กรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน

จากภาพที่ 5.7 แสดงถึงการที่ระบบสามารถทำการกำหนดกฎนโยบายในการทำงานได้พร้อมกันทั้งสองอินเทอร์เฟสจริง

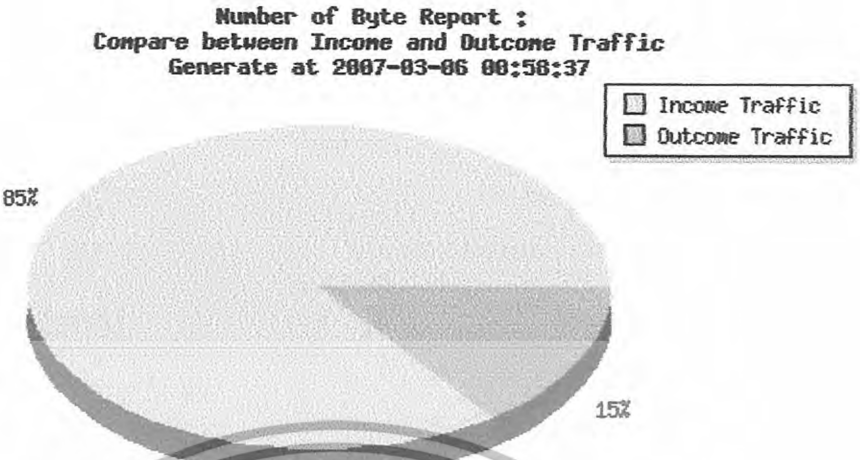
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



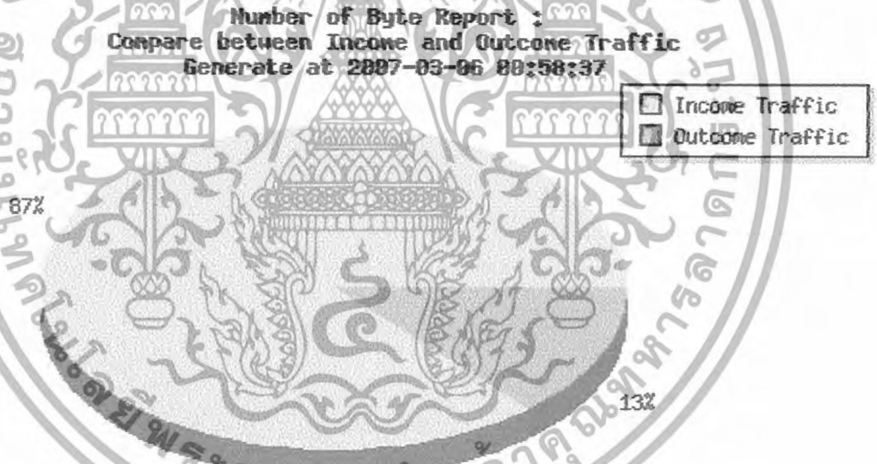
ภาพที่ 5.8 กราฟเส้นแสดงปริมาณข้อมูลในเครือข่ายโดยภาพรวมที่ผ่านอินเทอร์เฟส eth0 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน

ภาพที่ 5.9 กราฟเส้นแสดงปริมาณข้อมูลในเครือข่ายโดยภาพรวมที่ผ่านอินเทอร์เฟส eth2 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 5.10 กราฟวงกลมแสดงปริมาณข้อมูลในเครือข่าย โดยภาพรวมที่ผ่านอินเทอร์เฟส eth0 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน

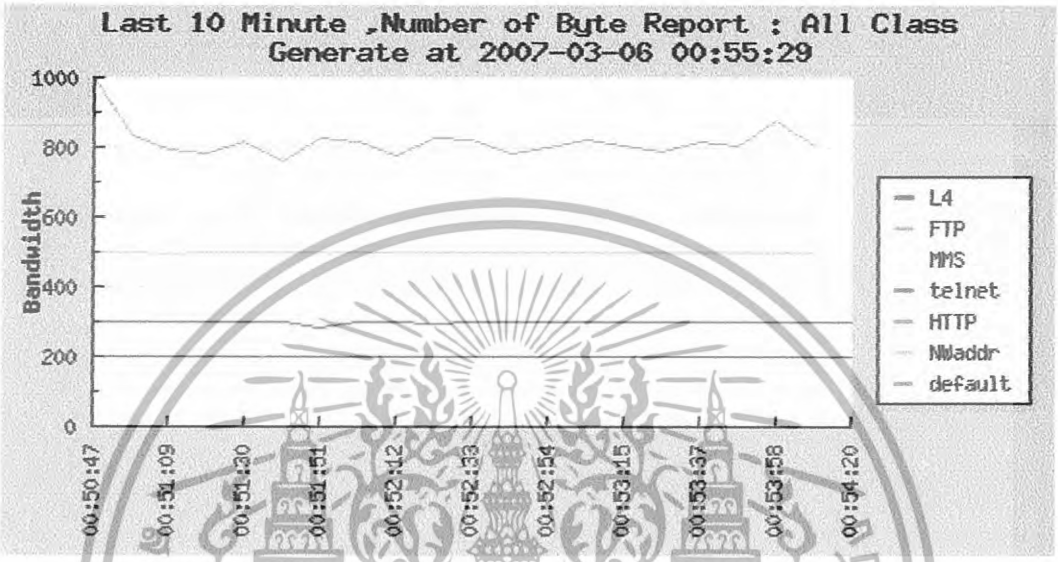


ภาพที่ 5.11 กราฟวงกลมแสดงปริมาณข้อมูลในเครือข่าย โดยภาพรวมที่ผ่านอินเทอร์เฟส eth2 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน

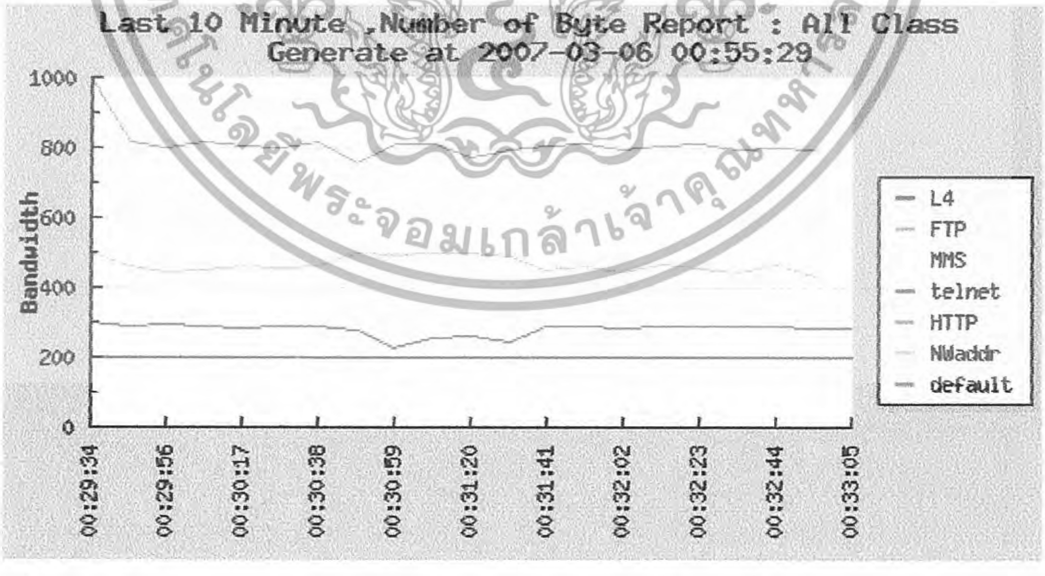
ภาพที่ 5.8 ถึง 5.11 แสดงให้เห็นว่าทราฟฟิกขาเข้าทั้งสองอินเทอร์เฟสถูกควบคุมปริมาณทราฟฟิกจริง ดังจะเห็นได้จากปริมาณทราฟฟิกขาเข้าและขาออกมีปริมาณที่แตกต่างกันมาก ซึ่งจากภาพที่ 5.8 และ 5.9 นั้น จะเห็นว่าปริมาณทราฟฟิกขาเข้าจากเส้นสีแดงนั้นจะมีอัตราเฉลี่ยของปริมาณทราฟฟิกอยู่ที่ 10 เมกกะบิตต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วินาที ตามปริมาณทราฟฟิกที่เราสร้างขึ้นจริงๆ ส่วนปริมาณทราฟฟิกขาออกนั้นก็มีปริมาณตามกฎนโยบายควบคุมปริมาณทราฟฟิกที่เรากำหนดไว้จริงๆว่าให้ทราฟฟิกที่ผ่านไปได้ทั้งหมดไม่เกิน 3 เมกกะบิตต่อวินาที ซึ่งดูได้จากคลาสรากจากภาพที่ 5.3

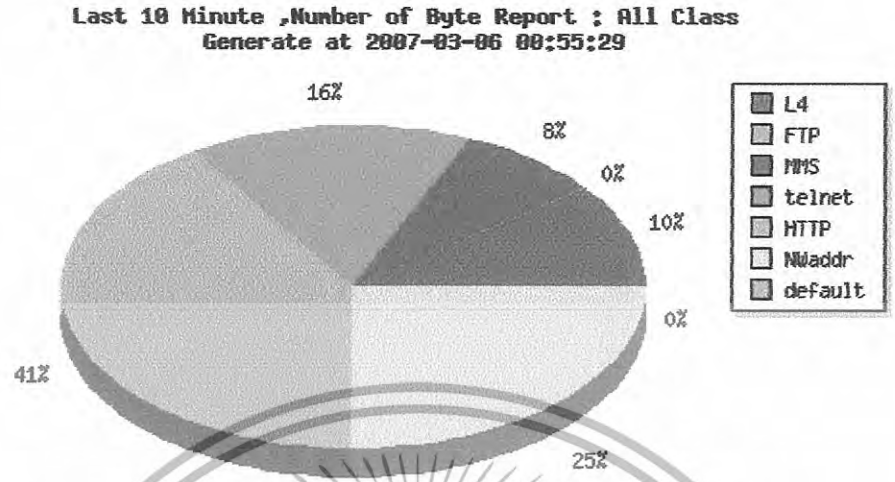


ภาพที่ 5.12 กราฟเส้นแสดงปริมาณข้อมูลในเครือข่ายตามคลาสที่ผ่านอินเทอร์เฟส eth0 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน



ภาพที่ 5.13 กราฟเส้นแสดงปริมาณข้อมูลในเครือข่ายตามคลาสที่ผ่านอินเทอร์เฟส eth2 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 5.14 กราฟวงกลมแสดงปริมาณข้อมูลในเครือข่ายตามคลาสที่ผ่านอินเทอร์เฟส eth0 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกัน

ภาพที่ 5.15 กราฟวงกลมแสดงปริมาณข้อมูลในเครือข่ายตามคลาสที่ผ่านอินเทอร์เฟส eth2 จากการทดสอบกรณีควบคุมปริมาณทราฟฟิก 2 อินเทอร์เฟสพร้อมกันตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 5.12 ถึง 5.15 แสดงให้เห็นว่าระบบสามารถทำการกรองทราฟฟิกโดยการกำหนดการกรองด้วยไอพีแอดเดรส, พอร์ต, โปรโตคอลระดับชั้นขนส่งข้อมูล และโปรโตคอลระดับชั้นแอพพลิเคชั่นได้จริง ดังนี้

- การกรองทราฟฟิกด้วยไอพีแอดเดรสของคลาสบริการ MMS ที่มีการรับส่งข้อมูลด้วยโปรโตคอล TCP ตามเส้นสีเทา
- การกรองทราฟฟิกด้วยพอร์ตของคลาสบริการ Telnet ตามเส้นสีชมพู
- การกรองทราฟฟิกด้วยโปรโตคอลระดับชั้นขนส่งข้อมูลของคลาสบริการ ICMP ตามเส้นสีแดง
- การกรองทราฟฟิกด้วยโปรโตคอลระดับชั้นแอพพลิเคชั่นของคลาสบริการ FTP ตามเส้นสีเขียว

แล้วยังแสดงให้เห็นว่าทราฟฟิกของบริการต่างๆ ถูกควบคุมปริมาณได้ตามกฎนโยบายควบคุมปริมาณทราฟฟิกจริง และในแต่ละคลาสนั้นได้รับการกระจายแบนด์วิดท์ตามที่ได้รับการรับประกันขั้นต่ำ และการจำกัดปริมาณแบนด์วิดท์สูงสุดจริง อย่างเช่นจากคลาส FTP ก็ได้รับการกระจายแบนด์วิดท์อยู่ที่ระหว่าง 750 กิโลบิตต่อวินาที ถึง 1000 กิโลบิตต่อวินาทีจริง

5.2.7 การทดสอบการทำงานระบบกรณีควบคุมปริมาณทราฟฟิก 1 อินเทอร์เฟส

ในการทดลองกรณีที่เป็นการควบคุมปริมาณทราฟฟิก 1 อินเทอร์เฟสนั้น จะทำการควบคุมที่อินเทอร์เฟส eth2 ที่เป็นทราฟฟิกขาเข้า ซึ่งได้ทำการกำหนดกฎการควบคุมปริมาณทราฟฟิกไว้ดังนี้

พระจอมเกล้าเจ้าคุณทหาร

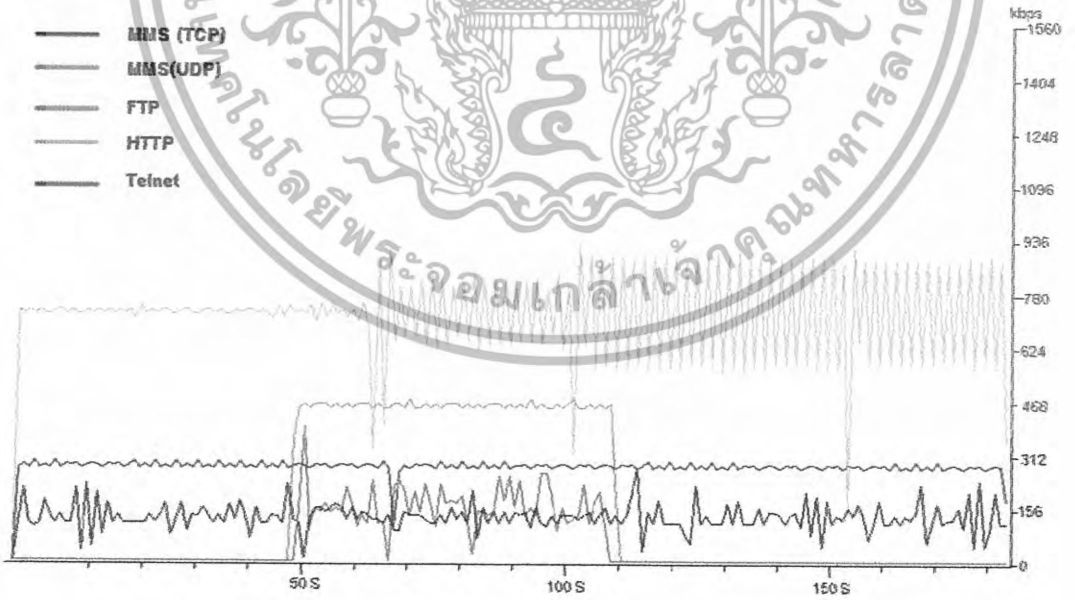
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

eth2 (Inbound) Information

Interface	Direction	Algorithm	Bandwidth	Threadhold	Edit	Delete
eth2	Inbound	auto	3000	3000		

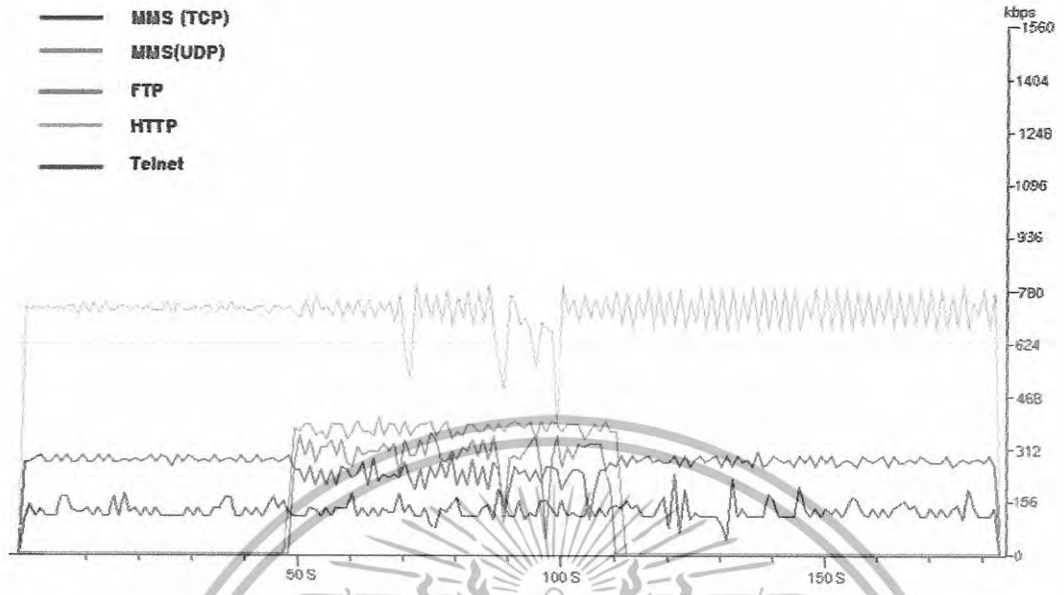
Class Name	Guarantee Rate	Max Rate	Filter Type	Filter Description	Edit
l4-l7	1200	1400			
l7	1100	1300			
ftp	750	1000	l7	filtered by ftp	
mms(udp)	350	500	port	both direction, protocol udp(port 1755, 5005, 1024:4443, 4445:5500)	
icmp	100	200	l4	filtered by icmp	
port-ip	1350	1500			
port	950	1200			
telnet	200	300	port	both direction, protocol tcp(port 23)	
http	750	1000	port	both direction, protocol tcp(port 80)	
ip	400	500			
mms(tcp)	400	500	ip	both direction, ip address 192.169.2.0/255.255.255.0	
default	150	150			

ภาพที่ 5.16 แสดงกฎนโยบายควบคุมปริมาณทราฟฟิกที่กำหนดจากเว็บแอพพลิเคชั่น กรณีควบคุมปริมาณทราฟฟิก 1 อินเทอร์เฟส

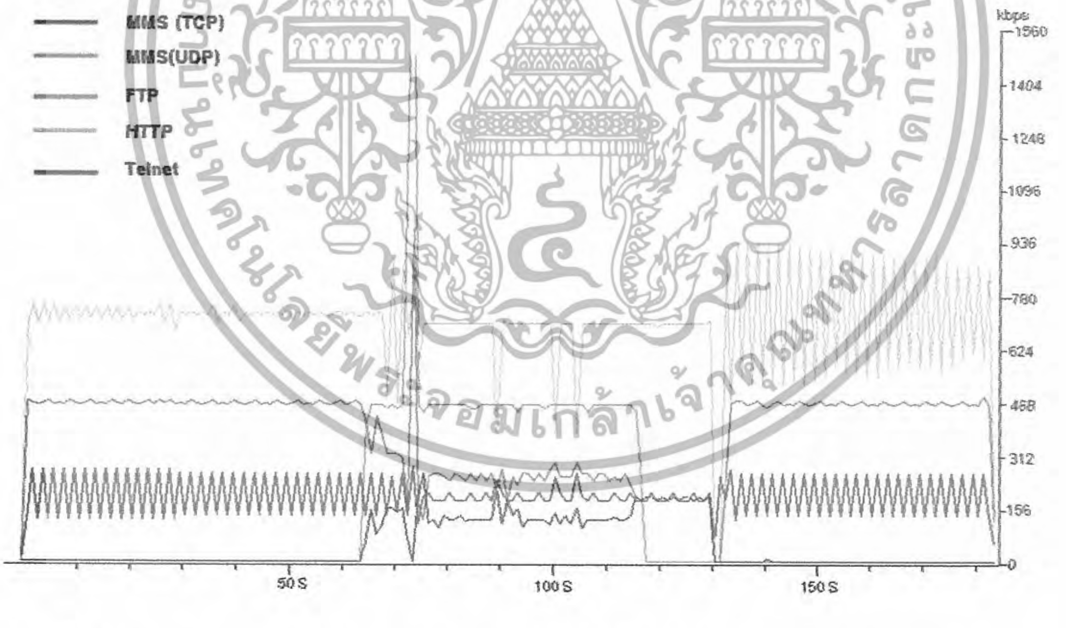


ภาพที่ 5.17 กราฟแสดงผลการควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึม HFSC กรณีควบคุมปริมาณทราฟฟิก 1 อินเทอร์เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

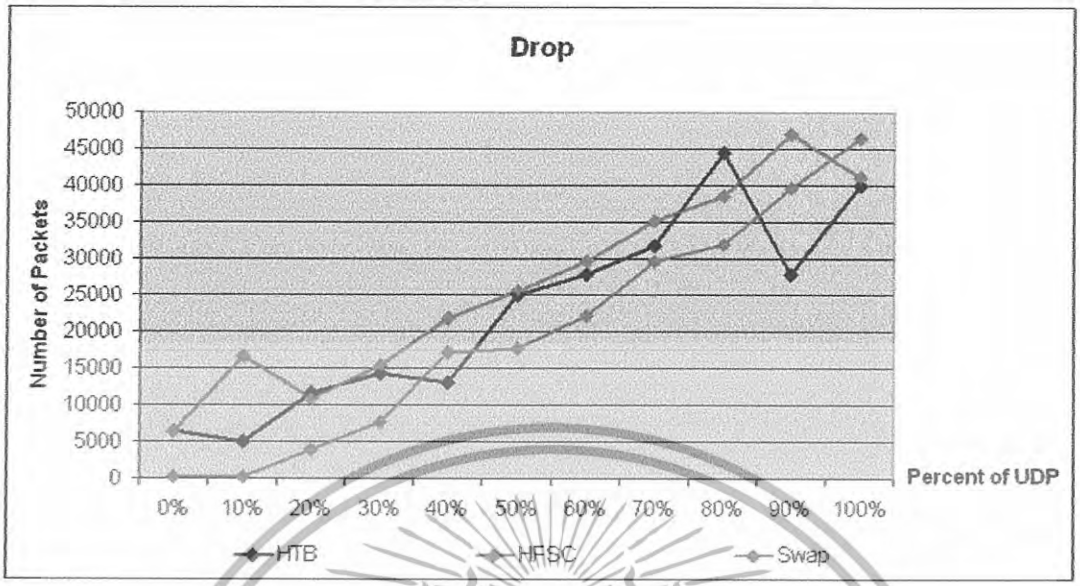


ภาพที่ 5.18 กราฟแสดงผลการควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึม HTB กรณีควบคุมปริมาณทราฟฟิก 1 อินเทอร์เฟส

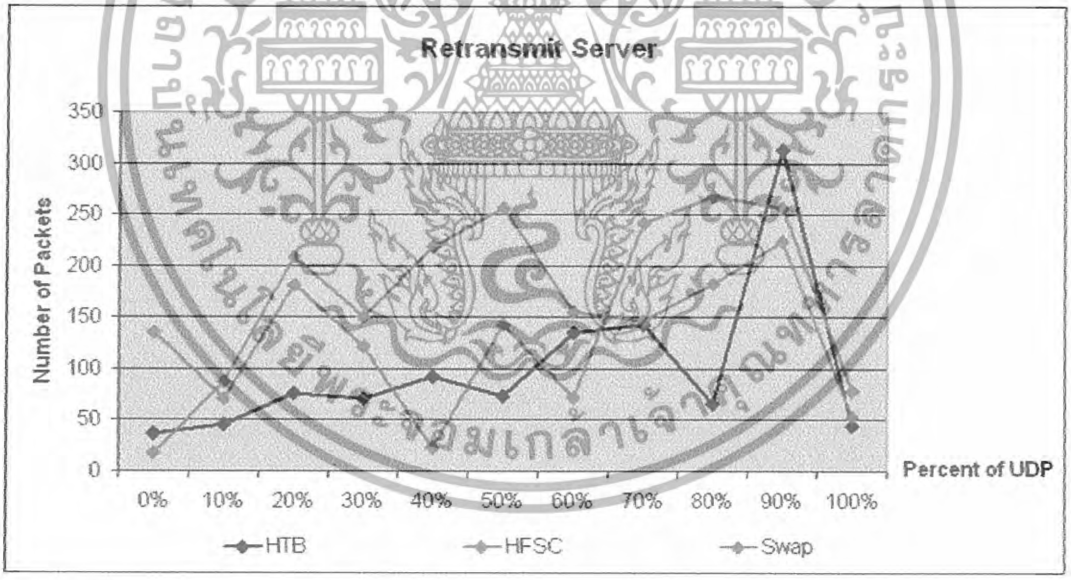


ภาพที่ 5.19 กราฟแสดงผลการควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึม Auto Swapping กรณีควบคุมปริมาณทราฟฟิก 1 อินเทอร์เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 5.20 กราฟแสดงการเปรียบเทียบแนวโน้มการดร็อปแพ็คเก็ต (Drop) จากการทดลองทั้ง 3 อัลกอลิทึม



ภาพที่ 5.21 กราฟแสดงการเปรียบเทียบแนวโน้มการส่งแพ็คเก็ตใหม่ (Retransmission) จากการทดลองทั้ง 3 อัลกอลิทึม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภาพที่ 5.17 ถึง 5.19 แสดงผลการทดลองที่ได้จากโปรแกรม Ethereal จากการทดลองที่อัตราส่วนปริมาณทราฟฟิกของ TCP ต่อ UDP เป็น 30 ต่อ 70 ซึ่งผ่านการควบคุมปริมาณทราฟฟิกจากทั้ง 3 อัลกอลิทึม จะเห็นว่ารูปแบบกราฟของการควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึม HTB และ HFSC นั้นจะค่อนข้างคล้ายคลึงกัน คือในช่วงวินาทีที่ 0 – 60 เส้นกราฟค่อนข้างที่จะคงที่ แล้วจะเริ่มไม่มีการเปลี่ยนแปลงมากตั้งแต่ช่วงวินาทีที่ 60 เป็นต้นไป ซึ่งในช่วงที่วินาทีที่ 50 – 110 นั้น จะมีการสร้างทราฟฟิกแบบเบิร์สเข้าไป แต่ถึงแม้ว่าจะเลิกสร้างทราฟฟิกแบบเบิร์สแล้วก็ยังคงเห็นว่าเส้นกราฟก็จะยังไม่นิ่งเหมือนในช่วงแรก เนื่องมาจากว่าปริมาณทราฟฟิกที่ยังคงอยู่ในแถวคอยนั้นยังมีปริมาณมากและสะสมอยู่ แต่อย่างไรก็ตามจะเห็นว่าพอทราฟฟิกมีปริมาณที่หนาแน่นมากขึ้นแล้ว เส้นกราฟของ HTB นั้นจะยังคงนิ่งกว่าของ HFSC จึงเป็นการสนับสนุนในการทดลองพารามิเตอร์ที่ว่าเมื่อมีทราฟฟิกปริมาณมากเข้ามาจะเห็นว่า HTB สามารถจัดการได้ดีกว่า HFSC

ในการทดลองที่ใช้อัลกอลิทึม Auto Swapping นั้น จากที่กล่าวไปแล้วว่าในช่วงวินาทีที่ 0 – 60 นั้น จะใช้อัลกอลิทึม HFSC ในการควบคุมปริมาณทราฟฟิกจากนั้นในช่วงวินาทีที่ 60-120 จะเปลี่ยนมาใช้อัลกอลิทึม HTB และหลังจากนั้นในช่วงวินาที่ 120-180 จะเปลี่ยนกลับมาใช้อัลกอลิทึม HFSC ดังเดิม แล้วจากกราฟแสดงผลการทดลองที่ใช้อัลกอลิทึม Auto Swapping นั้นจะเห็นได้ว่ากราฟในช่วงแรกและช่วงหลังนั้นมีรูปร่างที่คล้ายคลึงกับกราฟแสดงผลการทดลองที่ใช้อัลกอลิทึม HFSC ในช่วงเวลาเดียวกัน ส่วนในช่วงกลางของการใช้อัลกอลิทึม Auto Swapping นั้นจะมีรูปร่างที่คล้ายคลึงกับกราฟแสดงผลการทดลองที่ใช้อัลกอลิทึม HTB ในช่วงเวลาเดียวกัน

พระจอมเกล้าเจ้าคุณ

5.2.8 สรุปผลการทดสอบแอพพลิเคชั่น

จะเห็นว่าในการทดลองที่ใช้อัลกอลิทึม Auto Swapping ช่วงรอยต่อที่เปลี่ยนอัลกอลิทึมจาก HFSC มาเป็น HTB นั้น จะมีการส่งทราฟฟิกออกไปในปริมาณที่สูงมาก ซึ่งเกินกว่าค่าแบนด์วิดท์สูงสุดที่สามารถใช้งานได้ ที่เป็นเช่นนี้เนื่องจากว่าในช่วงรอยต่อนั้นเป็นช่วงที่ไม่มีการควบคุมปริมาณทราฟฟิกเลย เป็นช่วงสั้นๆที่ยกเลิกการใช้อัลกอลิทึม HFSC แล้วยังไม่ได้ทำการรันสคริปต์ของอัลกอลิทึม HTB แต่จะเห็นว่าคลาสที่มีปริมาณทราฟฟิกพุ่งขึ้นสูงนั้นจะมีแต่คลาส MMS ที่เป็นทราฟฟิกแบบ UDP เท่านั้น ส่วนคลาสอื่นๆที่เป็นทราฟฟิกแบบ TCP นั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะไม่มีการพุ่งขึ้นสูงอย่างนี้ เนื่องจาก TCP จะมีการตรวจสอบอยู่แล้วว่าปริมาณที่สามารถส่งได้มากที่สุดเป็นเท่าไหร่ จึงไม่มีการส่งเกินจากที่เป็นไปได้ ต่างจาก UDP ที่ไม่มีการตรวจสอบหาค่าที่มากที่สุด จึงทำการส่งตามปริมาณทราฟฟิกที่เราสร้างไว้จริงๆ

ดังนั้นเมื่อมีช่วงที่ไม่ได้ทำการควบคุมทราฟฟิกจึงมีแต่ UDP เท่านั้นที่จะส่งผ่านไปด้วยทราฟฟิกปริมาณที่มากกว่าค่าที่กำหนดตามกฎนโยบาย ซึ่งจากคุณสมบัตินี้ของ UDP ก่อให้เกิดผลอีกอย่างกับระบบ เนื่องจากในอัลกอลิทึมแบบ Auto Swapping นั้น จะใช้ปริมาณทราฟฟิกที่เข้ามายังระบบและค่า Threshold ในการวัดว่าควรจะสลับอัลกอลิทึมหรือไม่ ดังนั้นจึงมีแต่ทราฟฟิกแบบ UDP เท่านั้นที่จะมีปริมาณทราฟฟิกขาเข้าที่เกินกว่าค่า Threshold ได้ แต่หากเป็นทราฟฟิกแบบ TCP จะไม่มีเหตุการณ์ที่ปริมาณทราฟฟิกขาเข้ามีปริมาณที่มากกว่าค่า Threshold เนื่องจากว่าปริมาณที่ทราฟฟิกแบบ TCP จะส่งเข้ามานั้นจะไม่มากกว่าค่าขอบเขตที่มากที่สุดที่คลาสนั้นๆจะสามารถส่งทราฟฟิกออกได้อยู่แล้ว ดังนั้นหากใช้ Auto Swapping Algorithm ก็จะมีแต่ทราฟฟิกแบบ UDP เท่านั้นที่สามารถทำให้เกิดการสลับอัลกอลิทึมได้

จากภาพที่ 5.20 และ 5.21 เป็นกราฟที่สรุปผลการทดลองทั้ง 11 การทดลองย่อยของทั้ง 3 การทดลองหลัก ซึ่งเส้นกราฟแต่ละเส้นนั้นจะแสดงถึงแต่ละการทดลองหลัก โดยมีจุดที่เชื่อมกราฟแต่ละจุดเป็นค่าสรุปของแต่ละการทดลองย่อย โดยภาพที่ 5.20 แสดงถึงปริมาณการดร็อปแพ็คเก็ตจะเห็นว่าตามแนวโน้มแล้วการควบคุมปริมาณทราฟฟิกด้วยอัลกอลิทึม HFSC นั้นจะมีปริมาณที่มากที่สุด ที่รองมาก็จะเป็น HTB ส่วน Auto Swapping Algorithm นั้นจะมีแนวโน้มของการดร็อปแพ็คเก็ตน้อยที่สุด แล้วจากภาพที่ 5.21 ที่แสดงปริมาณการส่งแพ็คเก็ตใหม่ จะมีช่วงแรกที่ Auto Swapping Algorithm มีปริมาณการส่งแพ็คเก็ตใหม่มากที่สุด แต่ในช่วงหลังจะมีแนวโน้มที่น้อยกว่าอีกสองอัลกอลิทึมที่เหลือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

บทสรุป

การพัฒนาระบบเว็บแอพพลิเคชั่นควบคุมปริมาณทราฟฟิกในเครือข่ายที่ทำงานอยู่บนระบบปฏิบัติการลินุกซ์ มีวัตถุประสงค์เพื่อปรับปรุงการทำงานของระบบงานปัจจุบันที่เดิมเป็นแบบชุดคำสั่งแบบบรรทัดเป็นระบบเว็บแอพพลิเคชั่นที่สามารถนำไปใช้งานได้ง่ายและมีประสิทธิภาพมากยิ่งขึ้น อีกทั้งพิจารณาถึงคุณภาพการให้บริการ (QoS) ที่เกี่ยวข้องกับสื่อประสม ซึ่งทำให้สามารถควบคุมปริมาณทราฟฟิกของบริการต่างๆอย่างเช่น โปรแกรมประยุกต์ประเภทให้ได้รับบริการอย่างดี เท่าเทียมกัน และเกิดประโยชน์สูงสุด

6.1 สรุปผลการพัฒนาระบบ

โครงงานการพัฒนาระบบนี้มีวัตถุประสงค์เพื่อพัฒนาระบบเว็บแอพพลิเคชั่นที่ช่วยสร้างชุดคำสั่งควบคุมปริมาณทราฟฟิกในเครือข่ายบนระบบปฏิบัติการลินุกซ์ ให้สามารถใช้งานได้สะดวกสบาย และใช้งานได้รวดเร็วขึ้น ช่วยจัดการด้านรายงาน (Accounting) ของการใช้บริการต่างๆ ในเครือข่ายในรูปแบบกราฟฟิกให้เข้าใจง่าย

วัตถุประสงค์ของโครงงานนี้ คือ การควบคุมปริมาณทราฟฟิกของเรียลไทม์แอพพลิเคชั่น โดยอาศัยการสลับอัลกอลิทึมของการควบคุมปริมาณทราฟฟิกแบบอัตโนมัติ ซึ่งอัลกอลิทึมที่เราใช้ในการสลับนั้นมี 2 อัลกอลิทึม ได้แก่ HTB และ HFSC ซึ่งได้ทำการทดลองเพื่อหาความแตกต่างจากผลการทำงานของทั้งสองอัลกอลิทึม แล้วนำมาสรุปเพื่อหาว่าในสถานการณ์ที่ต่างๆกันนั้น อัลกอลิทึมใดสามารถที่จะจัดการแล้วทำให้การทำงานของเรียลไทม์แอพพลิเคชั่นมีประสิทธิภาพที่ดีกว่า ซึ่งจะใช้อัลกอลิทึม HFSC ในช่วงที่มีปริมาณทราฟฟิกไม่หนาแน่นมากนัก เนื่องจากโครงงานของเรามุ่งเน้นไปที่การให้บริการของแอพพลิเคชั่นแบบเรียลไทม์ ซึ่งในภาวะปกตินั้นอัลกอลิทึมแบบ HFSC จะสามารถควบคุมการให้บริการของทราฟฟิกแบบเรียลไทม์ได้เกิดประสิทธิภาพดีกว่า แต่เมื่อมีทราฟฟิกปริมาณมากอัลกอลิทึมแบบ HTB จะสามารถทำการควบคุมได้ดีกว่า ดังนั้นระบบจะเริ่มการทำงานด้วยอัลกอลิทึม HFSC แต่เมื่อมีทราฟฟิกปริมาณมากระบบจะทำการสลับอัลกอลิทึมในการทำงานมาเป็น HTB แทน และเมื่อปริมาณทราฟฟิกลดน้อยลงระบบก็จะทำการสลับอัลกอลิทึมในการทำงานกลับมาเป็น HFSC อีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งในการที่จะชี้วัดว่าสถานการณ์แบบใดควรที่จะสลับอัลกอลิทึมในการทำงานนั้น ได้ทำการทดลองเพื่อหาค่า Threshold ที่จะใช้เป็นการตัดสินว่า ณ ขณะนั้นมีปริมาณทราฟฟิกมากหรือน้อย และควรที่จะใช้อัลกอลิทึมแบบใดในการควบคุมปริมาณทราฟฟิก โดยค่า Threshold ที่เหมาะสมที่ทำให้ทั้ง 2 อัลกอลิทึมมีผลการทำงานที่ต่างกันนั้นมีค่าประมาณ 180 เปอร์เซ็นต์ของปริมาณแบนด์วิดท์ที่รับประกันขั้นต่ำไว้ของคลาสนั้นๆ ซึ่งในระบบมีการกำหนดว่าหากปริมาณทราฟฟิกของคลาสใดก็ตามมีปริมาณที่มากกว่าค่า Threshold ของคลาสนั้นๆแล้ว แสดงว่าทราฟฟิกมีปริมาณมาก การจัดการด้วยอัลกอลิทึมแบบ HTB จะเหมาะกว่า ดังนั้นจะมีการเปลี่ยนมาใช้อัลกอลิทึม HTB แต่หากมีไม่มีคลาสใดเลยที่มีปริมาณทราฟฟิกที่เกินกว่าค่า Threshold แสดงว่าทราฟฟิก ณ ขณะนั้นยังคงมีปริมาณที่ปกติอยู่ การจัดการด้วยอัลกอลิทึมแบบ HFSC จะเหมาะกว่า ดังนั้นจะมีการเปลี่ยนมาใช้อัลกอลิทึม HFSC

ปริมาณทราฟฟิกจะมีค่ามากกว่า 180 เปอร์เซ็นต์ของปริมาณแบนด์วิดท์ที่รับประกันในขั้นต่ำได้นั้นเกิดจากทราฟฟิกแบบ UDP ซึ่งไม่มีการปรับอัตราการส่งให้อยู่ในปริมาณที่เหมาะสมเหมือนกับโปรโตคอล TCP อีกทั้งโปรโตคอล UDP ส่วนใหญ่ให้บริการทราฟฟิกแบบเรียลไทม์อยู่แล้ว ดังนั้นจึงสรุปได้ว่า ทราฟฟิกที่จะทำให้เกิดการสลับอัลกอลิทึมส่วนใหญ่นั้นจะเป็นทราฟฟิกแบบเรียลไทม์

ระบบที่ทำการพัฒนานี้สามารถนำไปใช้งานควบคุมปริมาณทราฟฟิกบนเครือข่ายได้จริงและเหมาะสมในการทำงานระยะยาว เนื่องจากสามารถช่วยอำนวยความสะดวกในการสร้าง แก้ไข และ ลบกฎนโยบายควบคุมปริมาณทราฟฟิก มีรายงานสรุปสถิติการใช้งานบริการในเครือข่ายที่ทำความเข้าใจได้ง่าย และ สามารถทำการปรับเปลี่ยนอัลกอลิทึมในการควบคุมปริมาณทราฟฟิกได้เหมาะสมกับสถานการณ์ต่างๆ รวมถึงสามารถปรับปรุง และรับประกันคุณภาพการให้บริการสื่อประสมด้วย

6.1.1 จุดเด่นของโครงงาน

- เหมาะกับการใช้ควบคุมปริมาณทราฟฟิกในหลากหลายสถานการณ์ที่ผู้ใช้ไม่ต้องเข้ามาปรับเปลี่ยนอัลกอลิทึมควบคุมปริมาณทราฟฟิกบ่อยๆ สามารถใช้เทคนิคการทำงานแบบสลับอัลกอลิทึมอัตโนมัติจัดการในระยะยาวได้
- สามารถกำหนดให้ทำการควบคุมปริมาณทราฟฟิกได้จากสองอินเทอร์เฟสพร้อมกัน โดยในแต่ละอินเทอร์เฟสนั้นสามารถกำหนดกฎนโยบายการควบคุมปริมาณทราฟฟิกในการควบคุมแยกกันได้
- สามารถสร้างกฎนโยบายการควบคุมปริมาณทราฟฟิกได้อย่างง่ายดายผ่านระบบเว็บไซต์ซึ่งมีข้อแนะนำในการใช้งานทุกขั้นตอน
- สามารถแก้ไขกฎนโยบายการควบคุมปริมาณทราฟฟิกที่ได้ทำการกำหนดไปแล้วได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถแสดงกฎนโยบายการควบคุมปริมาณทราฟฟิกที่ผู้ใช้เป็นผู้ออกแบบได้ ซึ่งมีทั้งที่แสดงเป็นชุดคำสั่งและแสดงเป็นตารางให้เห็นถึงความสัมพันธ์ระหว่างคลาสบริการด้วยกัน และระหว่างคลาสบริการกับกฎการกรองทราฟฟิก

- สามารถแสดงผลการทำงานให้ออกมาอยู่ในรูปกราฟสถิติการใช้งานเครือข่ายได้ โดยมีกราฟที่แสดงทั้งปริมาณข้อมูลขาเข้าของแต่ละคลาส ปริมาณข้อมูลขาเข้าโดยรวมของอินเทอร์เฟส ปริมาณข้อมูลขาออกของแต่ละคลาส และปริมาณข้อมูลขาออกโดยรวมของอินเทอร์เฟส ซึ่งแสดงทั้งในรูปแบบของจำนวนไบต์ และจำนวนแพ็คเก็ตได้

- สามารถเลือกอัลกอลิทึมในการควบคุมปริมาณทราฟฟิกได้ถึง 3 แบบ ได้แก่ อัลกอลิทึมแบบ HTB, HFSC และ สามารถเลือกที่จะให้ทำงานได้ทั้งสองอัลกอลิทึมแบบที่เหมาะสมตามสถานการณ์ต่างๆ ได้ นั่นคือ เทคนิคการทำงานแบบสลับอัลกอลิทึมอัตโนมัติ

- หากเลือกที่จะใช้อัลกอลิทึมในการควบคุมปริมาณทราฟฟิกแบบที่ปรับระหว่างอัลกอลิทึม HTB และ HFSC ได้ตามความเหมาะสม โปรแกรมก็สามารถที่จะกำหนดค่า Threshold ได้แบบอัตโนมัติ ในกรณีที่ผู้ใช้ไม่ได้ทำการกำหนดค่าไว้ แต่หากผู้ใช้ต้องการที่จะกำหนดค่าเองก็มีข้อแนะนำให้ในตอนที่ทำการกำหนดกฎการควบคุมปริมาณทราฟฟิก

- หากผู้ใช้ไม่ได้ทำการสร้างคลาสดีฟอลต์ในคลาสระดับที่ 1 ไว้ ระบบจะทำการสร้างให้แบบอัตโนมัติ เพื่อให้การทำงานสามารถดำเนินได้ตามปกติ

6.1.2 จุดด้อยของโครงงาน

- ไม่สามารถสร้างกฎการกรองที่เป็นแบบที่ซ้อนกลุ่มบริการให้กับคลาสได้
- ยังไม่สามารถที่จะกรองโปรโตคอลระดับแอพพลิเคชั่นได้ครอบคลุมทุกโปรโตคอล เนื่องจากระบบทำการกรองโปรโตคอลระดับแอพพลิเคชั่นได้ด้วยการดึงโมดูลที่ชื่อว่า L7-Protocol มาใช้งาน ดังนั้นเมื่อในโมดูลไม่มีโปรโตคอลที่เกี่ยวกับเรียลไทม์แอพพลิเคชั่นที่ครบถ้วน ระบบจึงยังไม่สามารถที่จะกรองทราฟฟิกที่เป็นเรียลไทม์แอพพลิเคชั่นได้มีประสิทธิภาพมากพอ
- ฐานข้อมูลต้องเก็บข้อมูลเป็นจำนวนมาก เนื่องจากต้องมีการตรวจสอบและเก็บข้อมูลเกี่ยวกับปริมาณทราฟฟิกทุกๆ 10 วินาที
- ระบบต้องทำงานทำงานหนัก เนื่องจากโปรแกรมต้องรันการทำงานตลอดทุกๆ 10 วินาที ซึ่งหากผู้ใช้เลือก Automatic Swapping Algorithm แล้ว ในทุกๆ 10 วินาที นอกจากระบบจะทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตรวจสอบปริมาณทราฟฟิกทั้งขาเข้าและขาออกแล้ว ยังต้องทำการเปรียบเทียบกับค่า Threshold อีกด้วย อาจทำให้เกิดการทำงานที่รวนได้

- ประสิทธิภาพและคุณภาพจากการควบคุมปริมาณทราฟฟิกยังขึ้นกับตัวผู้ใช้เป็นสำคัญว่าผู้ใช้มีความรู้เกี่ยวกับกำหนดกฎการควบคุมปริมาณทราฟฟิกมากน้อยแค่ไหน

6.2 ประโยชน์ของระบบงานที่พัฒนาขึ้นใหม่

- ด้านผู้พัฒนาระบบ
    1. เข้าใจการทำงานของระบบควบคุมปริมาณทราฟฟิกบนเครื่องลินุกซ์ และสามารถนำมาประยุกต์ใช้งานได้
    2. ได้ศึกษาการเขียนโปรแกรมด้วยเทคนิคต่างๆ
- ด้านผู้ใช้งานระบบ
    1. ลดภาระในการสร้างกฎ เนื่องจากหากชุดคำสั่งที่ต้องการใช้มีความซับซ้อนมากๆ จะเกิดเป็นความไม่สะดวก
    2. มีระบบควบคุมปริมาณทราฟฟิกบนเครื่องลินุกซ์ที่ใช้งานได้ง่าย สะดวกสบาย อีกทั้งช่วยปรับปรุง และรับประกันคุณภาพการให้บริการสื่อประสม
    3. ผู้ใช้ที่ไม่มีความรู้พื้นฐานเกี่ยวกับกลไกการควบคุมปริมาณทราฟฟิกและการใช้งานชุดคำสั่ง TC สามารถเข้าใจและใช้งานได้ง่าย
    4. การแสดงผลนโยบายควบคุมปริมาณทราฟฟิกและแสดงข้อมูลการใช้งานแบนด์วิดท์ผ่านการแสดงผลทางกราฟสามารถเข้าใจได้ง่าย
- ด้านความปลอดภัยของข้อมูล
    1. มีการยืนยันตัวผู้ใช้ก่อนที่จะเข้าใช้งานระบบ โดยสามารถที่จะเพิ่ม ลบ และแก้ไขข้อมูลที่ใช้ในการยืนยันตัวได้
- ด้านความประหยัดค่าใช้จ่ายขององค์กร
    1. เหมาะกับองค์กรที่มีขนาดเล็กหรือขนาดกลาง ที่ไม่มีงบประมาณเพียงพอที่จะจัดซื้อระบบควบคุมปริมาณทราฟฟิกที่ขายในท้องตลาดที่มีราคาแพงมาใช้งานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 ข้อจำกัดของการพัฒนาระบบ

1. ในระหว่างการทดลองยังขาดแอพพลิเคชั่นที่มีความสามารถในการสร้างทราฟฟิกจำลอง (Traffic Generator) ในระดับชั้นแอพพลิเคชั่นได้ เพื่อให้เห็นถึงความสามารถในการควบคุมปริมาณทราฟฟิกได้ชัดเจนยิ่งขึ้น
2. ในการทดลองขาดตัวช่วยในการวิเคราะห์ทราฟฟิกข้อมูลในระดับชั้นแอพพลิเคชั่น เช่น ค่าความหน่วง และค่าความแปรปรวนของค่าความหน่วง เป็นต้น ทำให้ต้องใช้เวลามากในการทำผลการทดลองให้ออกมาในรูปผลของกราฟ และทำผลการทดลองประกอบการวิเคราะห์ต้องทำผ่านมือ ทำให้ผลที่ได้จากการทดลองอาจคลาดเคลื่อนไปบ้าง
3. จำนวนเครื่องคอมพิวเตอร์มีไม่เพียงพอในการนำมาใช้ในการทดลอง
4. เนื่องจากในการทดลองต้องมีการคำนวณหาค่าความหน่วง ซึ่งเป็นเรื่องยากในการกำหนดให้ทุกเครื่องทำงานที่เวลาเดียวกัน ทำให้เกิดปัญหาผลการทดลองของค่าความหน่วงที่อาจคลาดเคลื่อนไปได้

6.4 ปัญหาและอุปสรรคระหว่างการพัฒนาระบบ

เครื่องเซิร์ฟเวอร์บริการสารสื่อประสม (MMS Server) ทำการกระจายข้อมูลไม่คงที่ นั่นคือ บางครั้งทำการกระจายข้อมูลในรูปแบบแพ็คเก็ต TCP หรือ UDP ซึ่งไม่สามารถกำหนดให้กระจายข้อมูลในรูปแบบตายตัวได้ อีกทั้งการทดสอบระบบไม่สามารถกรองทราฟฟิกของโปรโตคอลระดับชั้นแอพพลิเคชั่นชนิดบริการสื่อประสมได้

เนื่องจากโดยพื้นฐานการทำงานของบริการสารสื่อประสม จะประกอบด้วย 2 โปรโตคอลหลักทำงานร่วมกัน คือ โปรโตคอล RTSP ที่ใช้ในการสร้างการเชื่อมต่อก่อนที่จะเริ่มส่งข้อมูลจริงและโปรโตคอล RTP ที่ใช้ในการส่งข้อมูล ซึ่งรูปแบบสำหรับกรองทราฟฟิกบริการสารสื่อประสมนั้นต้องทำการกรองทราฟฟิกที่เป็นโปรโตคอล RTP แต่โปรโตคอลนี้ไม่ได้มีมาในโมดูลที่ระบบใช้ในการกรองทราฟฟิกระดับชั้นแอพพลิเคชั่น (L7-Protocol) ทำให้ไม่สามารถกรองทราฟฟิกดังกล่าวได้ ดังนั้นในการทดลองจึงทำการกรองทราฟฟิกบริการสารสื่อประสมโดยใช้พอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

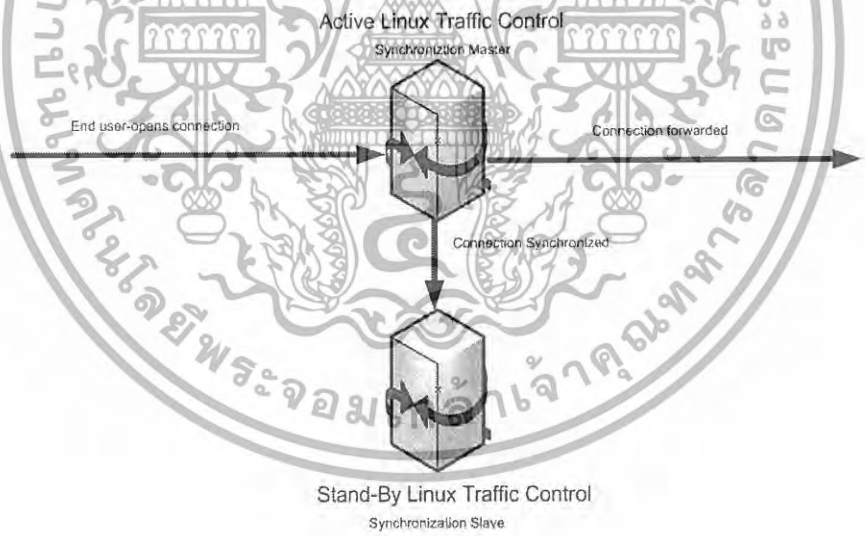
6.5 ข้อเสนอแนะ

ระบบที่ได้ทำการพัฒนาขึ้นมานี้ถือได้ว่าเสร็จสมบูรณ์ในระดับที่น่าพอใจ มีความสามารถที่รองรับการทำงานพื้นฐานได้และสามารถนำมาใช้งานได้จริง แต่ยังมีบางส่วนที่ยังทำงานได้ไม่ครบสมบูรณ์ และสามารถพัฒนาต่อเนื่องไปได้ เพื่อให้ระบบมีประสิทธิภาพมากขึ้น ได้แก่

1. Connection Synchronization

   ในขณะที่มีการควบคุมปริมาณทราฟฟิก หากเครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์ (Linux Traffic Control) เกิดการล้มเหลวเกิดขึ้น การเชื่อมต่อของทราฟฟิกการให้บริการอาจสูญหายได้ เนื่องจากแพ็คเก็ตที่กำลังส่งอยู่ระหว่างการให้บริการนั้นจะอยู่ที่เครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกกซ์ตัวที่เกิดการล้มเหลวขึ้น

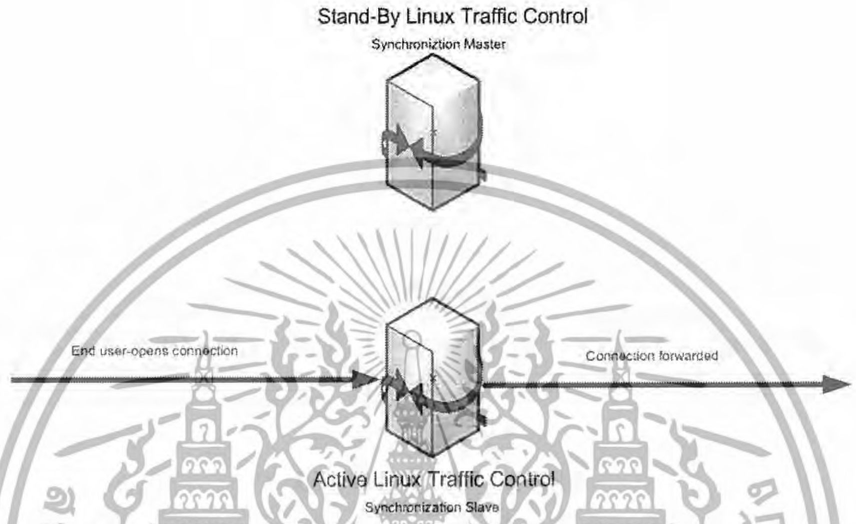
   สามารถแก้ไขปัญหาดังกล่าว ได้โดยแนวคิดของ Connection Synchronization ซึ่งมีแนวคิด คือ ให้มีการเลือกเครื่องหลักในการทำงาน (Master Node) ขึ้นระหว่างเครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกกซ์ ทั้งสองตัว



ภาพที่ 6.1 การส่งแพ็คเก็ตระหว่างเครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์หลัก (Active Linux Traffic Control) ให้กับเครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์สำรอง (Stand-By Linux Traffic Control)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ขอใช้บริการเริ่มสร้างการติดต่อกับระบบและเมื่อเครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์หลักที่ถือครองแอดเดรสเสมือน (Virtual IP) อยู่นั้นได้รับแพ็คเก็ตจากผู้ขอใช้บริการก็จะทำการส่งให้กับเครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์สำรองที่เป็นตัวที่รอทำงาน (Stand-By) อยู่ด้วย



ภาพที่ 6.2 การเกิดการล้มเหลวเกิดขึ้นแต่การเชื่อมต่อกับระบบยังไม่ขาด

เมื่อเกิดการล้มเหลวเกิดขึ้นเครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์สำรองก็จะทำหน้าที่หลักในการควบคุมปริมาณทราฟฟิก ถือครองแอดเดรสเสมือน (Virtual IP) และเปลี่ยนตัวเองไปอยู่ในลักษณะทำงานจริง (Active) แต่การเชื่อมต่อกับระบบจะยังไม่ขาดหายไป เนื่องมาจากการที่ตัวเครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์สำรองนั้นได้รับแพ็คเก็ตจากเครื่องหลักทุกครั้งที่มีแพ็คเก็ตเข้ามาในระบบ

2. การทำงานของระบบที่สามารถรองรับการอัพเดทออนไลน์ได้ เช่น ข้อกำหนดเพิ่มเติมของโปรโตคอลระดับแอพพลิเคชั่น หรือ การอัพเดทการทำงานของโปรแกรมและค่ากำหนดบางอย่างของระบบ เป็นต้น เพื่ออำนวยความสะดวกต่อการใช้งานของผู้ใช้ที่ไม่ต้องมาอัพเดทเอง
3. ระบบเว็บแอพพลิเคชั่นที่พัฒนาขึ้นมานี้สนับสนุนแค่การทำงานพื้นฐานของการทำงานในชุดคำสั่งแบบบรรทัด ยังไม่สามารถครอบคลุมการทำงานทั้งหมดที่ชุดคำสั่งแบบบรรทัดสามารถทำได้ ดังนั้นเพื่อความหลากหลายที่จะนำไปใช้งานบริหารจัดการควบคุมปริมาณทราฟฟิก ควรมีการพัฒนาต่อในเรื่องนี้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ควรนำหลักการปัญญาประดิษฐ์ (AI : Artificial Intelligence) มาใช้ในการสร้างกฎการควบคุมปริมาณทราฟฟิก เพื่อใช้ตัดสินใจในการสลับอัลกอลิทึมตามสถานการณ์ที่เหมาะสม
5. ทำการสร้างกฎการกรองที่ซ้อนกลุ่มบริการให้กับคลาสในการควบคุมปริมาณทราฟฟิกได้
6. พัฒนารายงานสถิติการควบคุมปริมาณทราฟฟิกรายวัน ในรูปแบบของไฟล์ข้อมูล PDF โดยแสดงข้อมูลทางสถิติ กราฟเส้นและกราฟวงกลมรายวันแสดงปริมาณทราฟฟิกในเครือข่าย
7. พัฒนารูปแบบการรายงานและข้อมูลสถิติการควบคุมปริมาณทราฟฟิกให้มีความหลากหลาย มีความยืดหยุ่น สามารถปรับปรุงใช้งานได้ตามความต้องการของผู้ใช้งานเองได้
8. ควรมีการสำรองข้อมูลสถิติการควบคุมปริมาณทราฟฟิกจากเครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์ ไปเก็บยังไฟล์เซิร์ฟเวอร์ (File Server) สำหรับสำรองข้อมูลและป้องกันข้อมูลสูญหาย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

กิตติ ภักดีวัฒนะกุล. 2548. **คัมภีร์ PHP.** พิมพ์ครั้งที่ 8. กรุงเทพฯ : เคทีพี คอมพ์ แอนด์ คอนซัลท์.

สัลยุทธ์ สว่างวรรณ. 2547. **ตำราคอมพิวเตอร์ระดับโลก เครือข่ายคอมพิวเตอร์.** พิมพ์ครั้งที่ 1. กรุงเทพฯ : บริษัท เพียร์สัน เอ็ดดูเคชั่น อินโดไชน่า จำกัด

ALTQ Tips. 2002. [Online].

Available : http://www.csl.sony.co.jp/~kjc/software/TIPS.txt

Bert Hubert. **Linux Advanced Routing & Traffic Control HOWTO.** [Online].

Available : http://lartc.org/howto/lartc.qdisc.classless.html

Chengzhi Li, Riccardo Bettati and Wei Zhao. **New Delay Analysis in High Speed Networks.**

Dan Singletary. 2003 **ADSL Bandwidth Management HOWTO.** [Online].

Available : http://www.tldp.org/HOWTO/ADSL-Bandwidth-Management-HOWTO/

Hanrijanto Sariowan, Rene L. Cruz and George C. Polyzos. 1995 **Scheduling for Quality of Service Guarantees via Service curve.**

Ion Stoica, Hui Zhang and T. S. Eugene Ng. 1996 **A Hierarchical Fair Service Curve Algorithm for Link Sharing, Real time and Priority Services.**

J.Babiarz and K.Chan. 2006. **Guidelines for DiffServ Service Class (RFC4594)**

Kihyun Pyun *, Junehwa Song and Heung-Kyu Lee. 2003 **A generalized hierarchical fair service curve algorithm for high network utilization and link-sharing.**

Klaus Rechert and Patric McHarhy **HFSC Scheduling with Linux** [Online].

Available : http://linux-ip.net/tc/hfsc.en/

lartc.org. **Linux Advanced Routing & Traffic Control (Quantum & r2q).** [Online].

Available : http://www.docum.org/docum.org/faq/cache/31.html

Lars Wischhof 2001 **Packet Scheduling for Bandwidth Sharing and Quality of Service Support in Wireless Local Area Networks.**

Martin A. Brown. 2003 **Traffic Control HOWTO.** [Online].

Available : http://linux-ip.net/articles/Traffic-Control-HOWTO/classful-qdiscs.html#qc-htb

Martin A. Brown. 2003. **Traffic Control using tcng and HTB HOWTO.** [Online].

Available : http://www.faqs.org/docs/Linux-HOWTO/Traffic-Control-tcng-HTB-HOWTO.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Martin Devera aka devik. 2005. **HTB Linux queuing discipline manual - user guide.** [Online].

Available : http://luxik.cdi.cz/~devik/qos/htb/manual/userg.htm. 2005.

Martin Devera aka devik. 2005 **Hierachical token bucket theory**. [Online].

Available : http://luxik.cdi.cz/~devik/qos/htb/manual/theory.htm

MikroTik. 2006. **Bandwidth Control.** [Online].

Available : http://www.mikrotik.com/docs/ros/2.9/root/queue

Patrick McHardy. **Hierarchical Fair Service Curve packet scheduler**.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

การติดตั้งระบบ

1. การติดตั้งระบบปฏิบัติการลินุกซ์

ระบบปฏิบัติการที่ใช้บนเครื่องเกตเวย์ คือ ระบบปฏิบัติการลินุกซ์อูบันตู (Linux Ubuntu 5.10) ดังภาพที่ ก.1 เนื่องจากเคอร์เนลเดิม คือ ระบบปฏิบัติการลินุกซ์ 2.6.14 ซึ่งไม่สนับสนุนการทำงานและการกรองทราฟฟิกระดับแอพพลิเคชั่น (L7 Filter) ดังนั้นเพื่อให้สนับสนุนการทำงานจำเป็นต้องคอมไพล์เคอร์เนลใหม่ด้วยคำสั่ง ดังนี้

ภาพที่ ก.1 ระบบปฏิบัติการลินุกซ์อูบันตู 5.10

สร้างรหัสผ่านให้แก่ root เพื่อให้สะดวกในการที่ไม่ต้องใช้ sudo นำหน้าคำสั่งที่ต้องใช้สิทธิ์ root เท่านั้นถึงจะสั่งทำงานได้ โดยใช้คำสั่ง

#sudo passwd root

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แก้ไฟล์โดยการคอมเมนต์ของ 2 บรรทัดแรกออก เพื่อให้สามารถไปดึงข้อมูลจากต้นทางที่ระบบปฏิบัติการลินุกซ์อูบันตู เตรียมไว้ให้ได้ โดยใช้คำสั่ง

#nano /etc/apt/sourcelist

จัดเตรียมโปรแกรมก่อนทำการติดตั้ง โดยรายชื่อโปรแกรมที่จะใช้ติดตั้งมีดังต่อไปนี้

#wget http://netfilter.org/projects/iptables/files/iptables-1.3.5.tar.bz2

#wget http://nchc.dl.sourceforge.net./sourceforge/l7-filter/netfilter-layer7-v2.2.tar.gz

#wget http://nchc.dl.sourceforge.net./sourceforge/l7-filter/l7-protocols-2006-05-11.tar.gz

#apt-get install fakeroot

#apt-get install libncurses5-dev

ทำการแตกแพ็คเก็จโปรแกรมไปไว้ในไดเร็กทอรี่ /usr/src/ ดังนี้

#tar –jxvf iptables-1.3.5.tar.bz

#tar –zxvf netfilter-layer7-v2.2.tar.gz

#tar –zxvf l7-protocols-2006-05-11.tar.gz

ทำการเปลี่ยนสิทธิ์ของโฟลเดอร์ข้อมูลทั้งหมดที่ทำการแตกแพ็คเก็จออกมา

#chrown root.root * -R

ทำการสร้างลิงค์ให้กับโฟลเดอร์ เพื่อให้การเรียกใช้นั้นสะดวกมากยิ่งขึ้น

#ln -s /usr/src/linux-2.6.15.4 linux

#ln -s /usr/src/iptables-1.3.5 iptables

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการลบค่าคอนฟิกเก่า ก่อนที่จะคอมไพล์เคอร์เนลและคัดลอกค่าคอนฟิกต้นฉบับเก็บไว้

```
#make mrproper
#cp /boot/config-2.6.11-1.1369_FC4 ./.config
```

ทำการติดตั้งไอพีเทเบิลใหม่ เพื่อให้สามารถทำงานร่วมกับ L7 Filter ได้ ดังคำสั่งต่อไปนี้

```
#cd /usr/src/linux
#patch -tp1 < /usr/src/netfilter-layer7-v2.1/kernel-2.6.13-2.6.15-layer7-2.1.patch
#cd /usr/src/iptables-1.3.5
#patch -tp1 < /usr/src/netfilter-layer7-v2.1/iptables-layer7-2.1.patch
#chmod +x /usr/src/iptables-1.3.5/extensions/.layer7-test
```

กำหนดการเลือกติดตั้งส่วนประกอบของระบบปฏิบัติการลินุกซ์ว่าจะทำการติดตั้งอะไรบ้าง โดยทำการเลือกใช้งาน L7 Filter ดังนี้

```
#make menuconfig
Networking --->
        Networking options --->
                Network packet filtering (replaces ipchains) --->
                        IP: Netfilter Configuration ---
                                <M> Layer 7 match support (EXPERIMENTAL)
                                [*] Layer 7 debugging output
Device Drivers --->
        Fusion MPT device support --->
                <*> Fusion MPT (base + ScsiHost) drivers
                OR
                <*> Fusion MPT ScsiHost drivers for SPI [kernel 2.6.13]
Device Drivers --->
        SCSI device support --->
                <*> SCSI device support
                <*> SCSI disk support
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCSI low-level drivers --->

<*> SYM53C8XX Version 2 SCSI support

เมื่อทำการติดตั้งเสร็จแล้วให้ทำการบันทึกการเปลี่ยนแปลงที่เราได้เลือกไว้

#make KERNEL_DIR=/usr/src/linux-2.6.14

#make install KERNEL_DIR=/usr/src/linux-2.6.14

ทำการติดตั้งรูปแบบของโปรโตคอลของ L7 Filter ใช้สำหรับตรวจสอบข้อมูลส่วนหัวของข้อมูลระดับชั้นแอพพลิเคชั่น ที่ไดเร็กทอรี่ /etc/l7-protocols

#cd /usr/src//l7-protocols-2006-01-22

#make install

ทำการคอมไพล์เคอร์เนลใหม่ โดยใช้คำสั่งดังนี้

#fakeroot make-kpkg --append-to-version=.layer7 kernel_image –initrd

2. การติดตั้งซอฟต์แวร์

2.1 ติดตั้งโปรแกรมเครื่องแม่ข่าย (Web Server) คือ Apache

#sudo apt-get install apache2

ทำการติดตั้ง php version 4 และทำการรีสตาร์ท Apache ใหม่

#sudo apt-get install php4

#sudo /etc/init.d/apache2 restart

ทำการทดสอบ Apache โดยการสร้างไฟล์ php ขึ้นมาทดสอบ โดยใส่รายละเอียดดังนี้

<?php phpinfo(); ?>

ทดสอบการทำงานโดยใช้ url คือ http://localhost/testphp.php และทำการแก้ไข /etc/apache2/apache2.conf ดังนี้จาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DirectoryIndex index.html index.htm index.shtml index.php index.php3 index.pl index.xhtml

ทำการแก้ไข /etc/mine.type และทำการ comment ข้อมูลดังนี้

```
#application/x-httpd-php                    phtml
#application/x-httpd-php-source             phps
#application/x-httpd-php3                   php3
#application/x-httpd-php3-preprocessed      php3p
#application/x-httpd-php4                   php4
```

ทำการแก้ไข /etc/apache2/mods-enabled/php4.conf และคอมเมนต์ข้อมูลดังนี้

```
<IfModule mod_php4.c>
#  AddType application/x-httpd-php .php .phtml .php3
#  AddType application/x-httpd-php-source .phps
</IfModule>
```

ทำการ enable module บางตัวของ Apache โดยใช้คำสั่งดังนี้

```
#a2enmod ssl
#a2enmod rewrite
#a2enmod suexec
#a2enmod include
```

ทำการ Reload Apache configuration โดยใช้คำสั่ง

```
#sudo /etc/init.d/apache2 force-reload
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ติดตั้งโปรแกรมฐานข้อมูล

ใช้โปรแกรมฐานข้อมูลชื่อ Mysql Server เนื่องจากสามารถใช้งานได้ฟรี โดยทำการติดตั้งด้วยคำสั่งดังนี้

#sudo apt-get install mysql-server libapache2-mod-auth-mysql php4-mysql

#sudo mysqladmin -u root password 'shaper'

#sudo /etc/init.d/mysql start

ทดสอบการติดตั้ง Mysql โดยเมื่อใช้คำสั่ง netstat –tap จะได้ผลดังนี้

tcp 0 0 localhost.localdo:mysql *:* LISTEN 2449/mysqld

ถ้าหากว่าไม่พบผลดังกล่าว ให้ทำการแก้ไขโดยใช้คำสั่ง /etc/mysql/my.cnf และคอมเมนต์ skip-networking: ออก ดังนี้

skip-networking

ในกรณีที่ทำการแก้ไขโดยใช้คำสั่ง /etc/mysql/my.cnf จำเป็นต้องทำการรีสตาร์ทMysql โดยใช้คำสั่งดังนี้

/etc/init.d/mysql restart

2.3 ติดตั้งโปรแกรมจัดการฐานข้อมูล (phpMyAdmin)

#sudo apt-get install phpmyadmin

ทำการเซ็ตค่าไฟล์คอนฟิกใน sudo gedit /etc/phpmyadmin/config.inc.php โดยทำการแก้ไขจาก $cfg['Servers'][$i]['password'] มาเป็น $cfg['Servers'][$i]['shaper']

ทำการแก้ไข้ข้อมูลใน sudo gedit /etc/apache2/apache2.conf ดังนี้

Alias /phpmyadmin "/usr/share/phpmyadmin"

<Directory "/usr/share/phpmyadmin">

AllowOverride None

Order allow,deny

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Allow from 127.0.0.1

</Directory>

ทำการรีสตาร์ท Apache ใหม่อีกครั้งดังนี้

#sudo /etc/init.d/apache2 restart

ทำการทดสอบการใช้งาน phpMyAdmin ผ่าน url คือ http://localhost/phpmyadmin

3. การติดตั้งแอพพลิเคชั่น

ทำการติดตั้งเว็ปแอพพลิเคชั่น ด้วยคำสั่งดังนี้

#cp /tmp

#tar xvfz TraffieShappo.tar.gz

#ln -s TraffieShappo shaper

#cd shaper

#cp –rf * var/www/

ทำการติดตั้งชุดคำสั่งสำหรับสร้างกราฟ โดยใช้คำสั่ง

#wget http://members.chello.se/jpgraph/jpgdownloads/jpgraph-1.20.3.tar.gz

#tar zxvf /var/www/shaper/jpgraph/jpgraph-1.20.5.tar.gz

#mv jpgraph-1.20.5/ jpgraph

ทำการสร้างฐานข้อมูล ตารางข้อมูล และกำหนดสิทธิ์ผู้เข้าใช้งานแอพพลิเคชั่น โดยใช้คำสั่งตามไฟล์ที่เก็บในโฟลเดอร์ /shaper/install/Database Setting Query.txt

ทำการแก้ไขไฟล์ /etc/sudoers โดยเพิ่มข้อมูลดังนี้

www-data ALL=NOPASSWD:/sbin/tc

www-data ALL=NOPASSWD:/sbin/iptables

www-data ALL=NOPASSWD:/usr/local/sbin/iptables

www-data ALL=NOPASSWD:/usr/sbin/sudo

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

การใช้งานระบบ

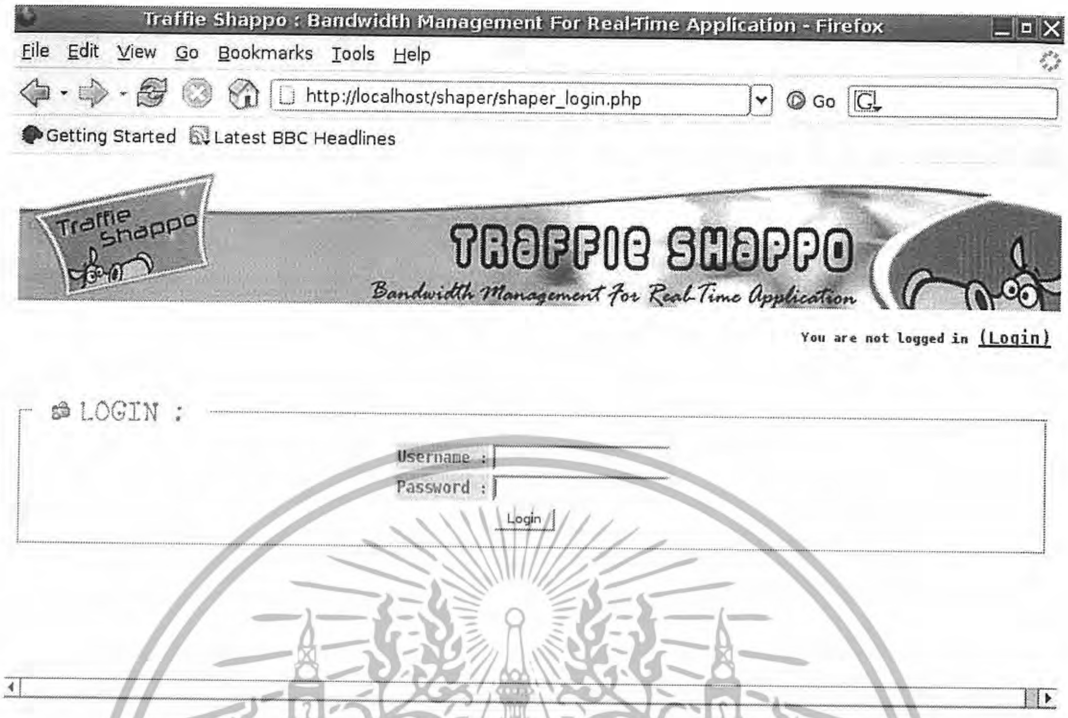
การเข้าใช้งานระบบเว็บแอพพลิเคชั่นควบคุมปริมาณทราฟฟิก สามารถเข้าใช้งานโดยผ่านทาง http://server_name/shaper_login.php หรือ http://server_name/shaper_login.php โดยรายละเอียดการใช้งาน มีดังนี้

1. การล็อกอินเข้าใช้งานระบบ (Login)

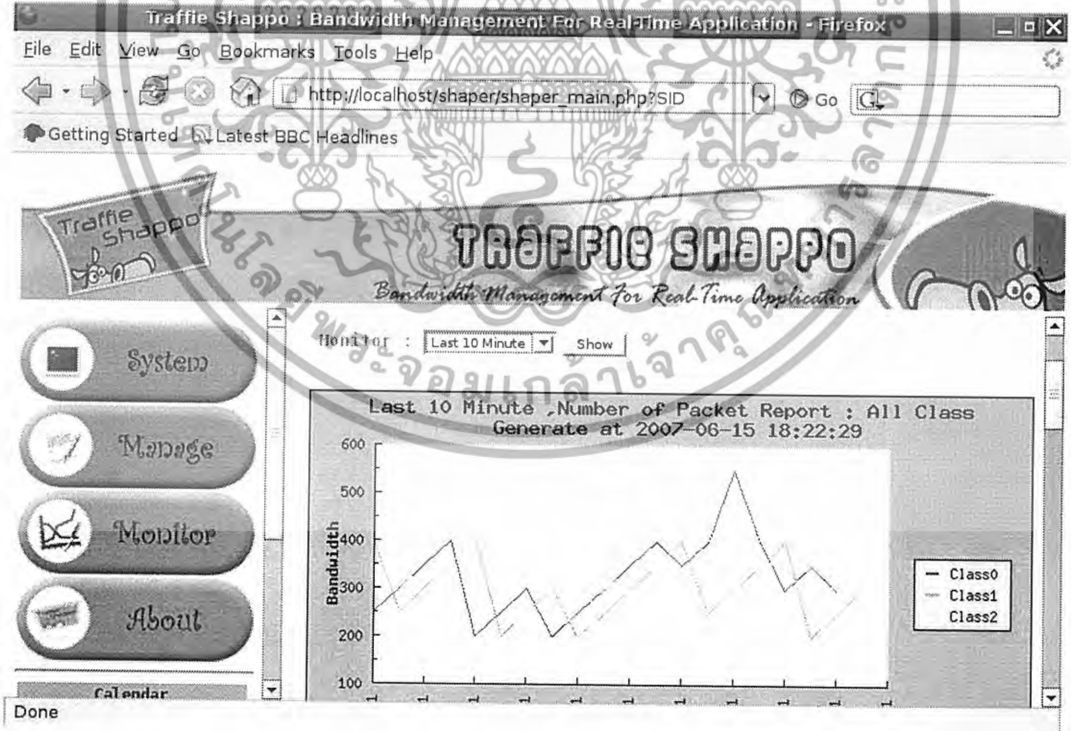
เมื่อทำการเปิดเข้าใช้งานระบบจะปรากฏหน้าล็อกอินเข้าใช้งาน ดังภาพที่ ข.1 เพื่อทำการพิสูจน์ตัวตนและสิทธิ์การเข้าใช้งานของแต่ละบัญชีรายชื่อ ซึ่งบัญชีรายชื่อแบ่งออกเป็น 2 ประเภท คือ ผู้ดูแลระบบควบคุมปริมาณทราฟฟิกบนเครื่องลินุกซ์ (Administrative) ซึ่งสามารถสร้างและลบข้อมูลต่างๆ ได้ เช่น ข้อมูลการกำหนดค่าของเครื่อง การกำหนดค่าของกฎนโยบาย การเพิ่มข้อมูล และ ลบข้อมูลผู้ดูแลระบบ เป็นต้น และผู้สังเกตการณ์ (Monitor) สามารถดูข้อมูลได้อย่างเดียว ไม่สามารถทำการสร้าง แก้ไข หรือ ลบข้อมูลใดๆ โดยบัญชีรายชื่อที่ติดตั้งมากับระบบมี 1 บัญชี คือ บัญชี “admin” และรหัสผ่าน “admin”

เมื่อทำการล็อกอินเข้ามาในระบบได้แล้ว จะพบกับหน้าต่างแสดงรายงานสถิติการใช้งานทางเครือข่าย โดยกำหนดให้เริ่มต้นแสดงกราฟเส้นและกราฟวงกลมรายงานสถิติการใช้บริการทางเครือข่ายตามคลาสในช่วงระยะเวลา 10 นาทีล่าสุด ดังภาพที่ ข.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ ข.1 หน้าต่างล็อกอินเข้าสู่ระบบ



ภาพที่ ข.2 หน้าต่างแสดงรายงานสถิติการใช้งานทางเครือข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

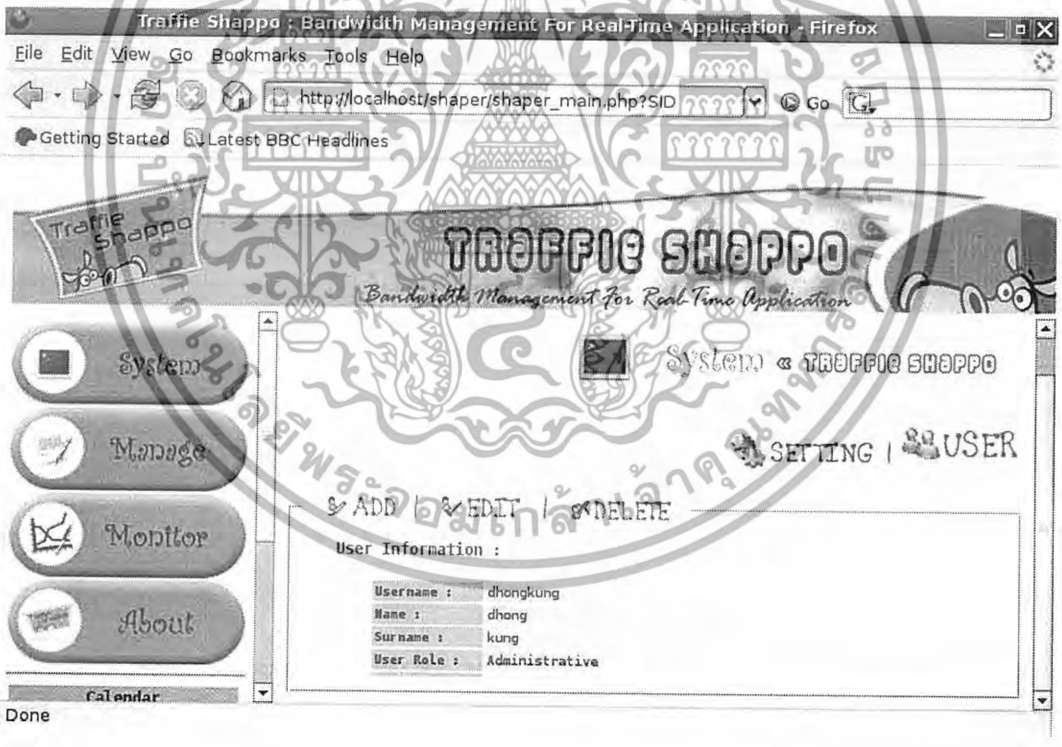
2. การจัดการชื่อบัญชีผู้ใช้งาน (User)

ภาพที่ ข.3 เป็นหน้าต่างแสดงข้อมูลของบัญชีผู้ใช้งาน โดยแสดงรายละเอียดข้อมูลของผู้ใช้งานที่สำคัญ รวมถึงแสดงถึงสิทธิ์การใช้งานที่ผู้ใช้ถูกกำหนดไว้ด้วย

ภาพที่ ข.4 เป็นหน้าต่างแสดงแก้ไขข้อมูลบัญชีผู้ใช้งาน โดยผู้ใช้ทุกคนมีสิทธิ์ในการแก้ไขข้อมูลส่วนตัวได้ แต่ไม่สามารถเปลี่ยนสิทธิ์การใช้งานที่ผู้ใช้ถูกกำหนดไว้ได้

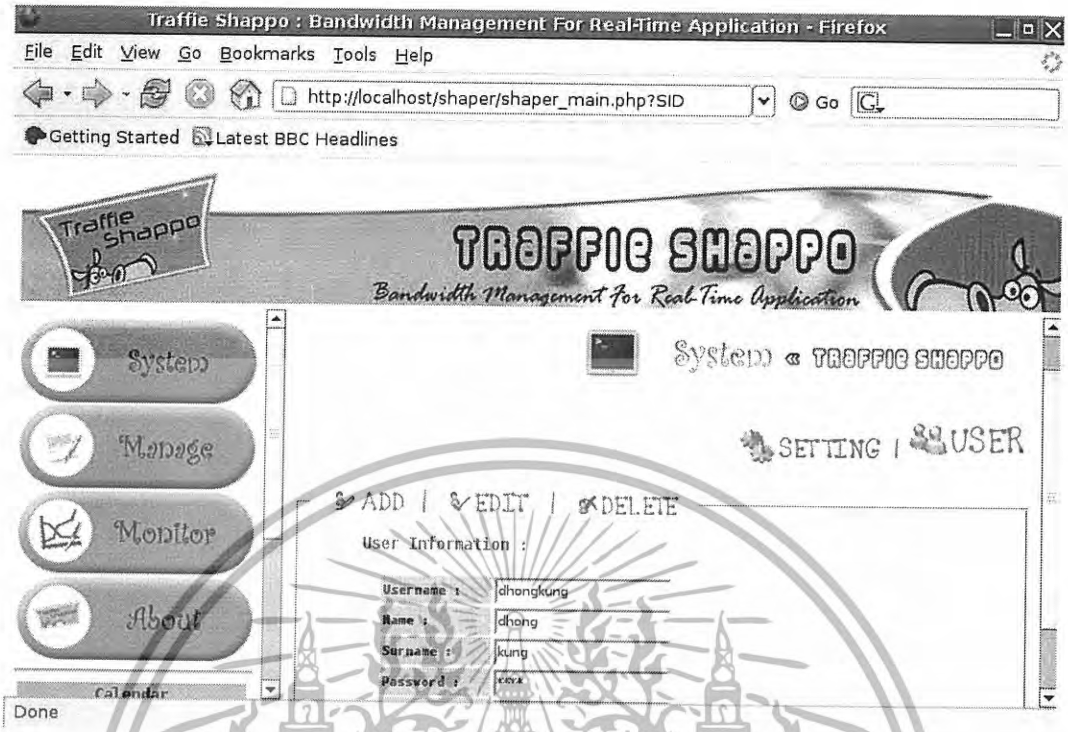
ภาพที่ ข.5 เป็นหน้าต่างแสดงการสร้างบัญชีผู้ใช้งาน โดยหน้าที่การสร้างบัญชีเป็นหน้าที่เฉพาะผู้ดูแลระบบควบคุมปริมาณทราฟฟิกบนเครื่องลินุกซ์เท่านั้น ผู้สังเกตการณ์ไม่ได้รับอนุญาตให้สร้างบัญชีผู้ใช้งานใหม่

ภาพที่ ข.6 เป็นหน้าต่างแสดงการลบบัญชีผู้ใช้งาน โดยหน้าที่การลบบัญชีเป็นหน้าที่เฉพาะผู้ดูแลระบบควบคุมปริมาณทราฟฟิกบนเครื่องลินุกซ์เท่านั้น ผู้สังเกตการณ์ไม่ได้รับอนุญาตให้ลบบัญชีผู้ใช้งานอื่น

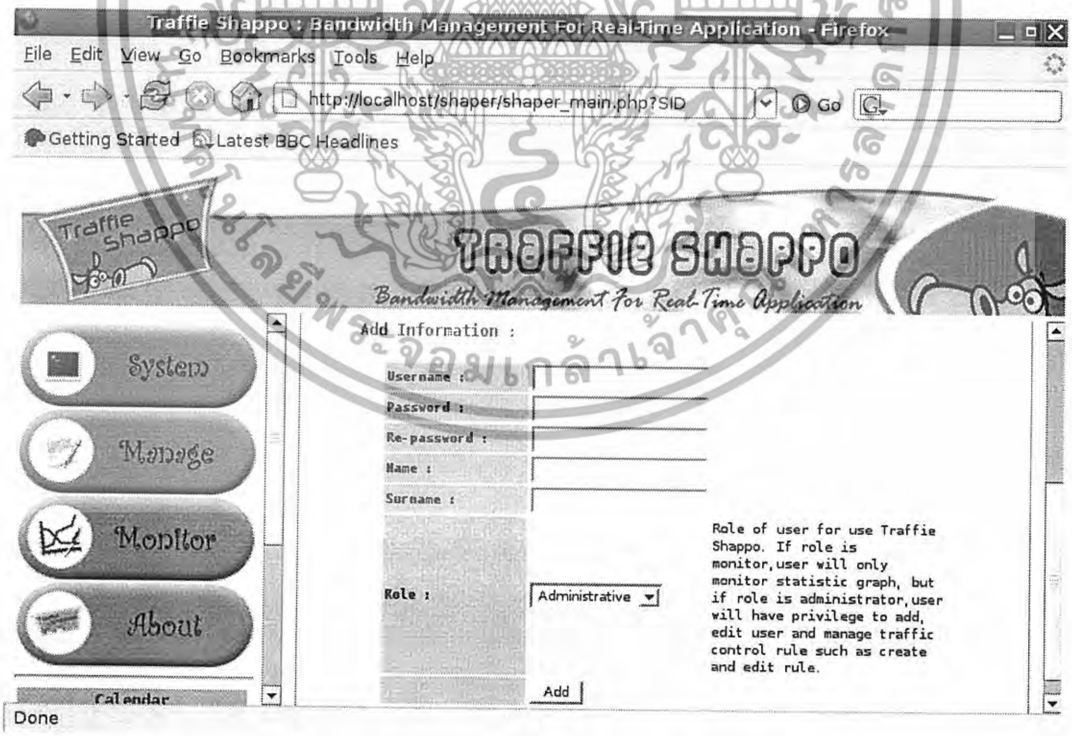


ภาพที่ ข.3 หน้าต่างแสดงข้อมูลของบัญชีผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ ข.4 หน้าต่างแสดงการแก้ไขข้อมูลบัญชีผู้ใช้งาน



ภาพที่ ข.5 หน้าต่างแสดงการสร้างบัญชีผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ ข.6 หน้าต่างแสดงการลบบัญชีผู้ใช้งาน

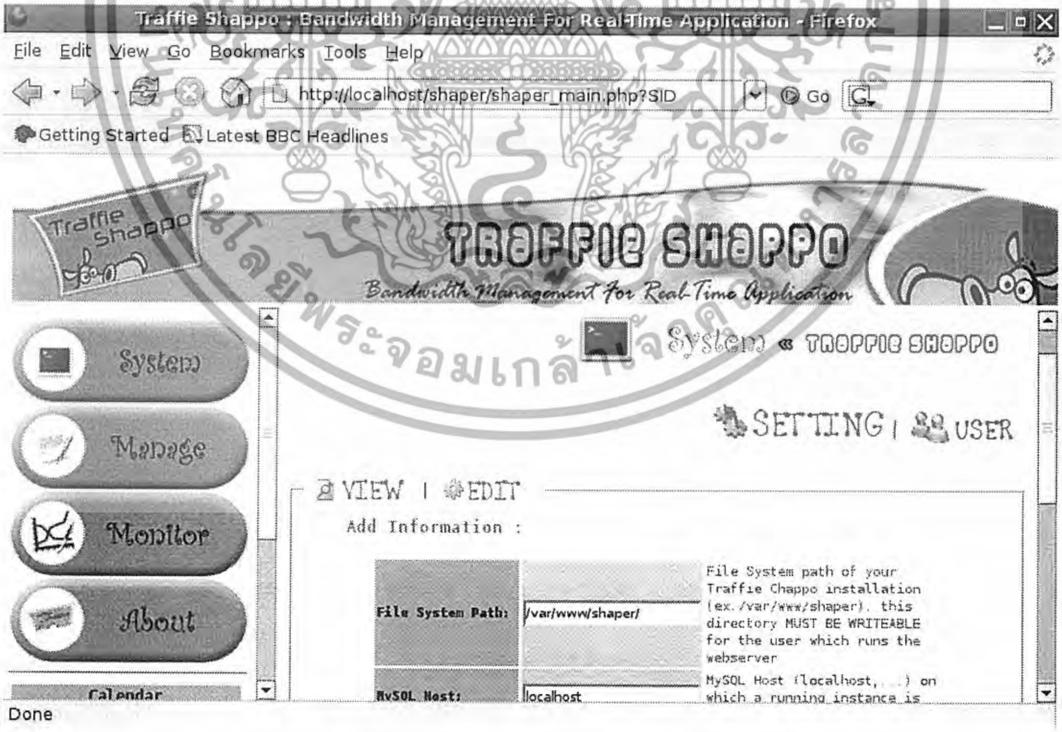
3. การกำหนดค่าให้กับเครื่อง (Setting)

ภาพที่ ข.7 เป็นหน้าต่างแสดงการดูค่าต่างๆที่ได้กำหนดไว้ในเครื่อง โดยแสดงรายละเอียดข้อมูลสำคัญในการทำงานของเว็บแอพพลิเคชั่น

ภาพที่ ข.8 เป็นหน้าต่างแสดงการกำหนดค่าต่างๆให้กับเครื่อง โดยหน้าที่การกำหนดค่าต่างๆให้กับเครื่องเป็นหน้าที่เฉพาะผู้ดูแลระบบควบคุมปริมาณทราฟฟิกบนเครื่องลินุกซ์เท่านั้น ผู้สังเกตการณ์ไม่ได้รับอนุญาตให้กำหนดค่าต่างๆให้กับเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ ข.7 หน้าต่างแสดงการดูค่าต่างๆที่ได้กำหนดไว้ในเครื่อง



ภาพที่ ข.8 หน้าต่างแสดงการแก้ไขค่าต่างๆให้กับเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. การจัดการกฎนโยบายควบคุมปริมาณทราฟฟิก (Manage)

4.1 วิซาร์ดการสร้างกฎนโยบายควบคุมปริมาณทราฟฟิก (Create Rule Wizard)

การสร้างกฎนโยบายควบคุมปริมาณทราฟฟิกเป็นหน้าที่เฉพาะผู้ดูแลระบบควบคุมปริมาณทราฟฟิกบนเครื่องลินุกซ์เท่านั้น ผู้สังเกตการณ์ไม่ได้รับอนุญาติให้กำหนดค่าต่างๆ ให้กับเครื่อง

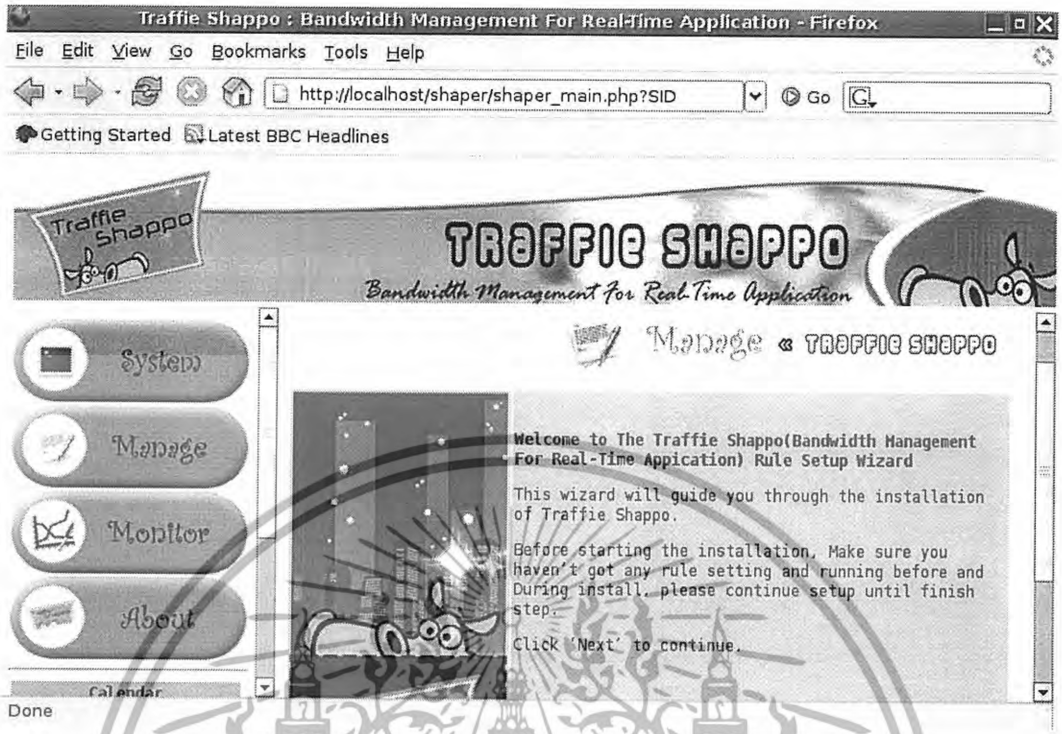
ภาพที่ ข.9 เป็นหน้าต่างแสดงข้อกำหนดในการสร้างกฎนโยบายควบคุมปริมาณทราฟฟิก โดยก่อนสร้างกฎควรตรวจสอบให้แน่ใจก่อนว่าอินเทอร์เฟสที่ต้องการใช้ควบคุมปริมาณทราฟฟิกในการสร้างกฎไม่ได้ถูกกำหนดไว้แล้วและถ้าหากเริ่มต้นสร้างกฎนโยบายควบคุมปริมาณทราฟฟิกตามวิซาร์ดดังกล่าวต้องทำให้เสร็จครบทุกขั้นตอน

ภาพที่ ข.10 เป็นหน้าต่างแสดงการกำหนดค่าให้กับเครื่องเริ่มต้น เพื่อใช้ในการทำงานของเว็บแอพพลิเคชั่นในการควบคุมปริมาณทราฟฟิกต่อไป

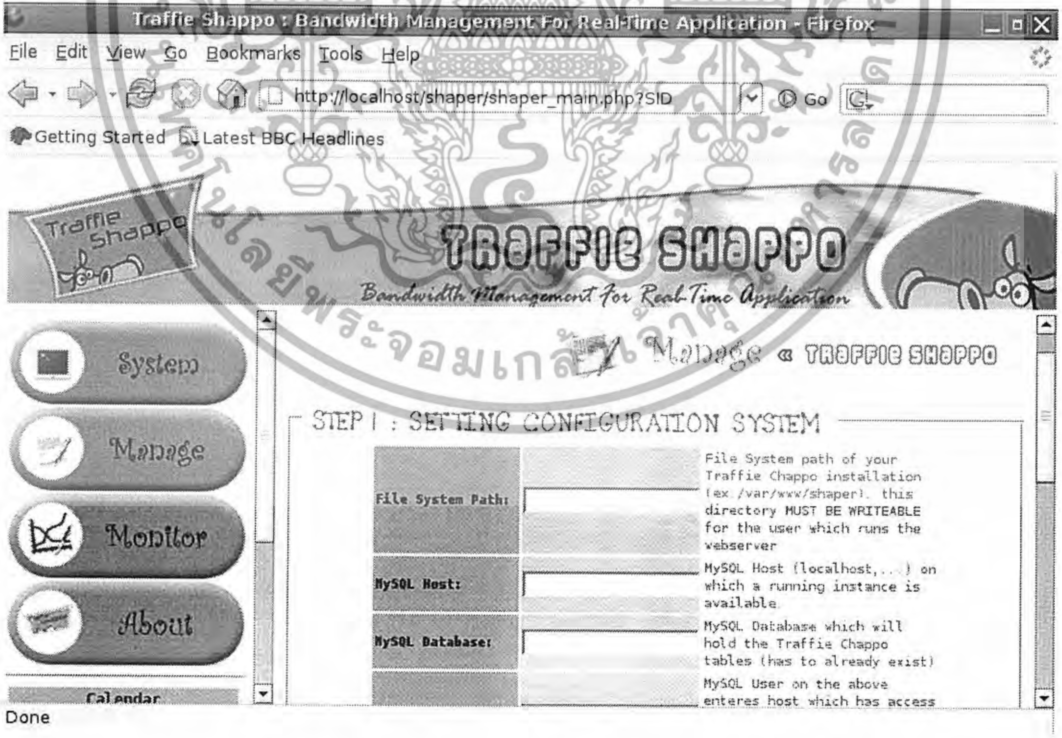
ภาพที่ ข.11 เป็นหน้าต่างแสดงการกำหนดค่าข้อมูลของการปรับแต่งทราฟฟิก โดยการกำหนดค่าอินเทอร์เฟสควรระมัดระวังในการกรอกค่าต้องเป็นตัวเล็กทั้งหมด นอกจากนั้นการกำหนดอัลกอลิทึมชนิด Swap Automatic หมายถึง แอพพลิเคชั่นจะทำการสลับอัลกอลิทึมระหว่าง HTB และ HFSC ตามความเหมาะสมของสถานการณ์ นั่นคือ โดยปกติจะทำงานด้วยอัลกอลิทึม HFSC แต่เมื่อไหร่ที่จำนวนไบต์ข้อมูลของคลาสใดคลาสหนึ่งเกินค่า Threshold ของคลาสนั้นๆ แอพพลิเคชั่นจะทำการสลับอัลกอลิทึมมาทำงาน HTB แทน และเมื่อไหร่ที่จำนวนไบต์ข้อมูลของทุกคลาสมีค่าต่ำกว่าค่า Threshold ของแต่ละคลาส แอพพลิเคชั่นจะสลับอัลกอลิทึมกลับมาเป็น HFSC เหมือนเดิม

การกำหนดค่า Threshold นั่น ถ้าหากผู้ใช้งานมีความรู้พื้นฐานเกี่ยวกับการควบคุมปริมาณทราฟฟิกและสามารถประมาณค่า Threshold ที่เหมาะสมกับกฎนโยบายควบคุมปริมาณทราฟฟิกก็สามารถที่จะกำหนดค่า Threshold ให้วิซาร์ดได้เลย แต่ถ้าหากผู้ใช้ไม่มีความรู้ดังกล่าวให้กำหนดค่าเป็น 0 เพื่อที่แอพพลิเคชั่นจะใช้ค่า Threshold ที่มาจากผลการทดลองของโครงงาน ถ้าหากว่าจำนวนไบต์ข้อมูลมีค่ามากกว่าจำนวนเปอร์เซ็นที่ได้จากการทดลองของค่ารับประกันขั้นต่ำของคลาสนั้นๆ แอพพลิเคชั่นจะทำการสลับอัลกอลิทึมเป็น HTB แทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ ข.9 หน้าต่างแสดงข้อกำหนดในการสร้างกฎนโยบายควบคุมปริมาณทราฟฟิก



ภาพที่ ข.10 หน้าต่างแสดงการกำหนดค่าให้กับเครื่องเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

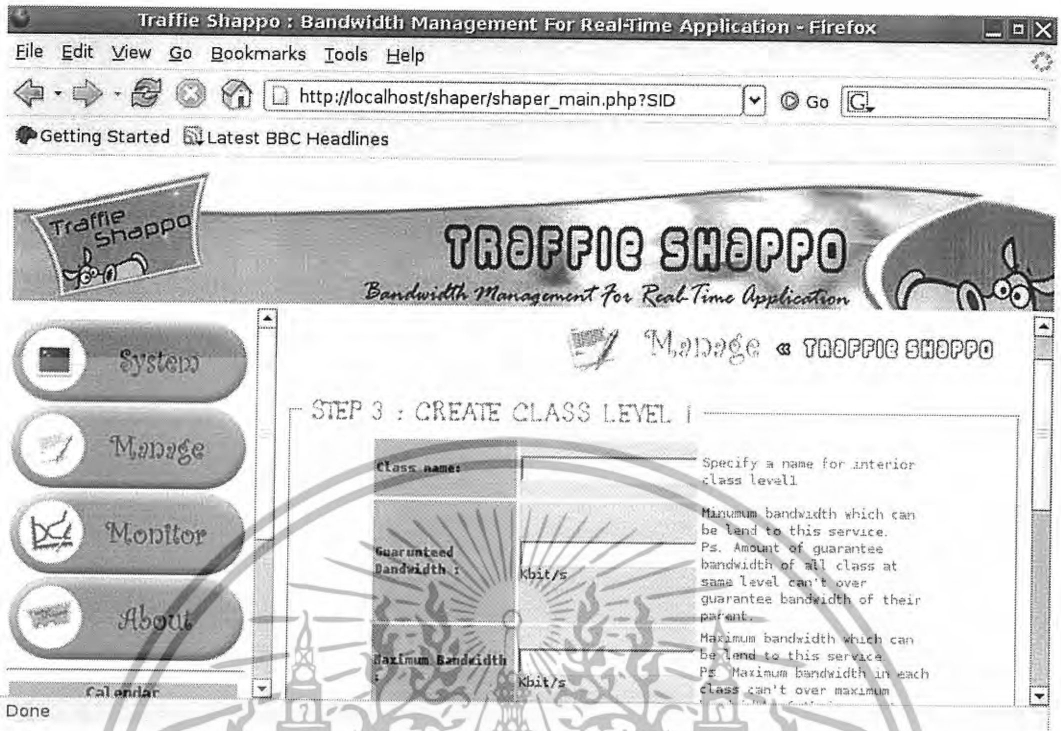
ภาพที่ ข.11 หน้าต่างแสดงการกำหนดค่าข้อมูลของการปรับแต่งทราฟฟิก

ภาพที่ ข.12 เป็นหน้าต่างแสดงการกำหนดค่าของคลาสระดับที่ 1 โดยถ้าหากผู้ใช้ต้องการสร้างกฎที่มีโครงสร้างคลาสลึก 1 ชั้น สามารถเลือก Next to Finish Create Class เพื่อกำหนดชนิดการกรองทราฟฟิกได้เลย แต่ถ้าหากต้องการสร้างกฎที่มีโครงสร้างลึกมากกว่า 1 ชั้นให้ทำการเลือก Next

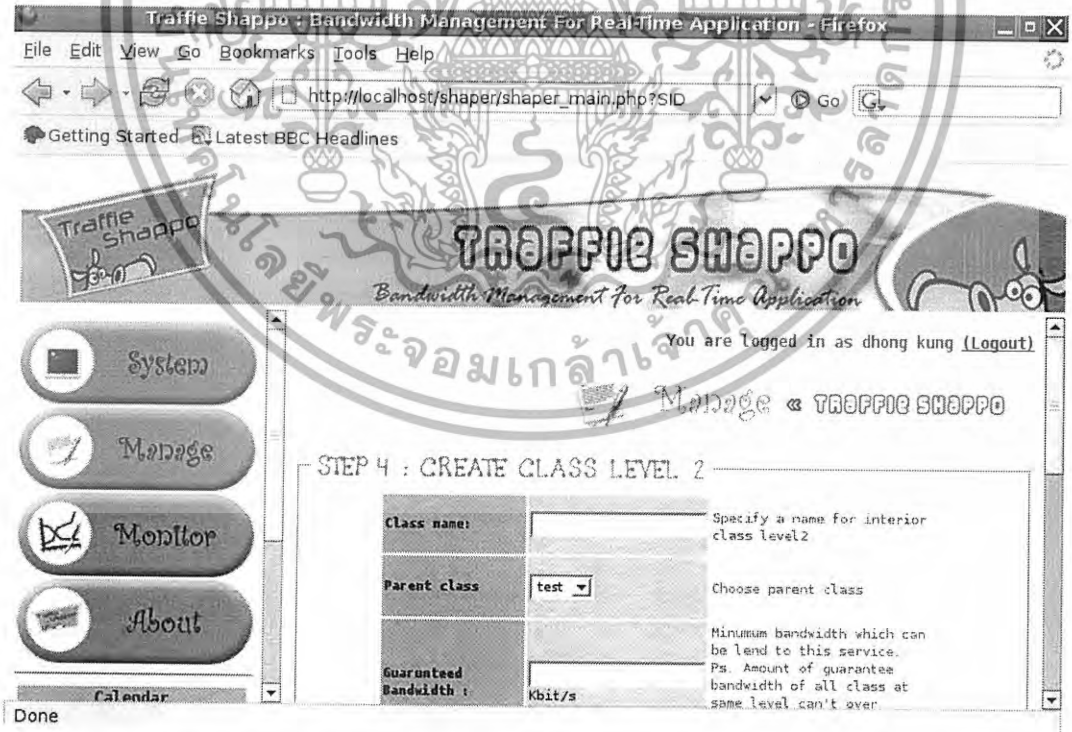
ภาพที่ ข.13 เป็นหน้าต่างแสดงแสดงการกำหนดค่าของคลาสระดับที่ 2 โดยในการสร้างคลาส ผู้ใช้ต้องเลือกคลาสแม่ของคลาสนั้นๆ ด้วย ซึ่งถ้าหากผู้ใช้ต้องการสร้างกฎที่มีโครงสร้างคลาสลึก 2 ชั้น สามารถเลือก Next to Finish Create Class เพื่อกำหนดชนิดการกรองทราฟฟิกได้เลย แต่ถ้าหากต้องการสร้างกฎที่มีโครงสร้างลึกมากกว่า 2 ชั้นให้ทำการเลือก Next

ภาพที่ ข.15 เป็นหน้าต่างสำหรับกำหนดชนิดการกรองทราฟฟิกของคลาส โดยทำการแสดงข้อมูลของคลาสสุดท้าย (Leaf Class) ทั้งหมดในกฎนโยบายของการควบคุมปริมาณทราฟฟิก โดยแต่ละคลาสสามารถเลือกกำหนดชนิดการกรองทราฟฟิกได้เพียงหนึ่งชนิดเท่านั้น แต่สามารถที่จะกำหนดได้หลายข้อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

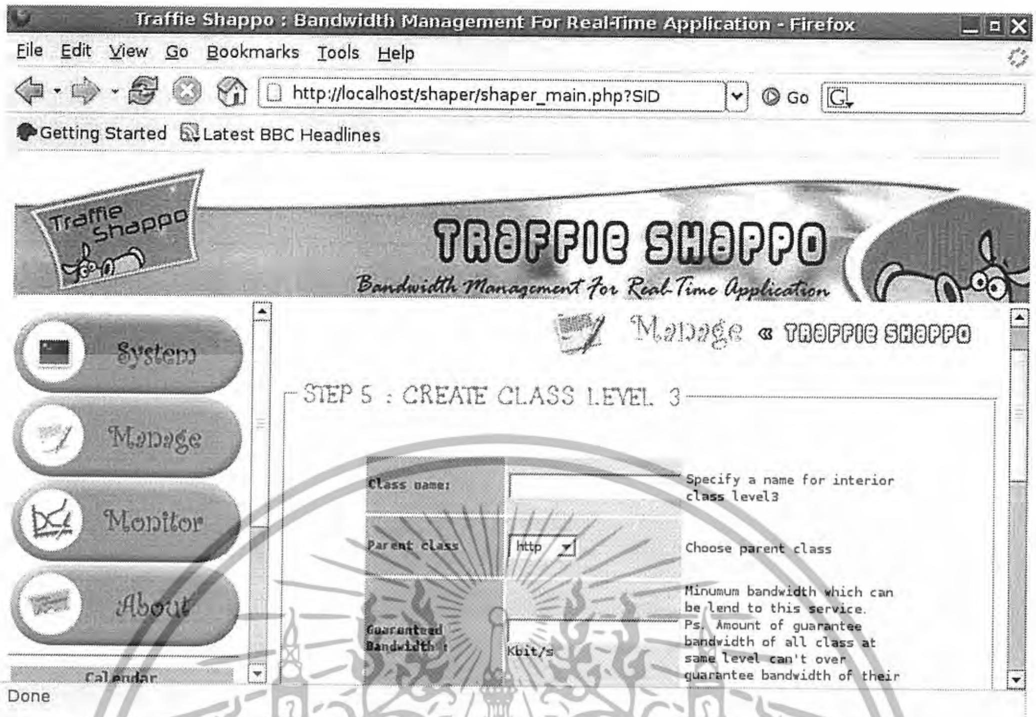


ภาพที่ ข.12 หน้าต่างแสดงการกำหนดค่าของคลาสระดับที่ 1

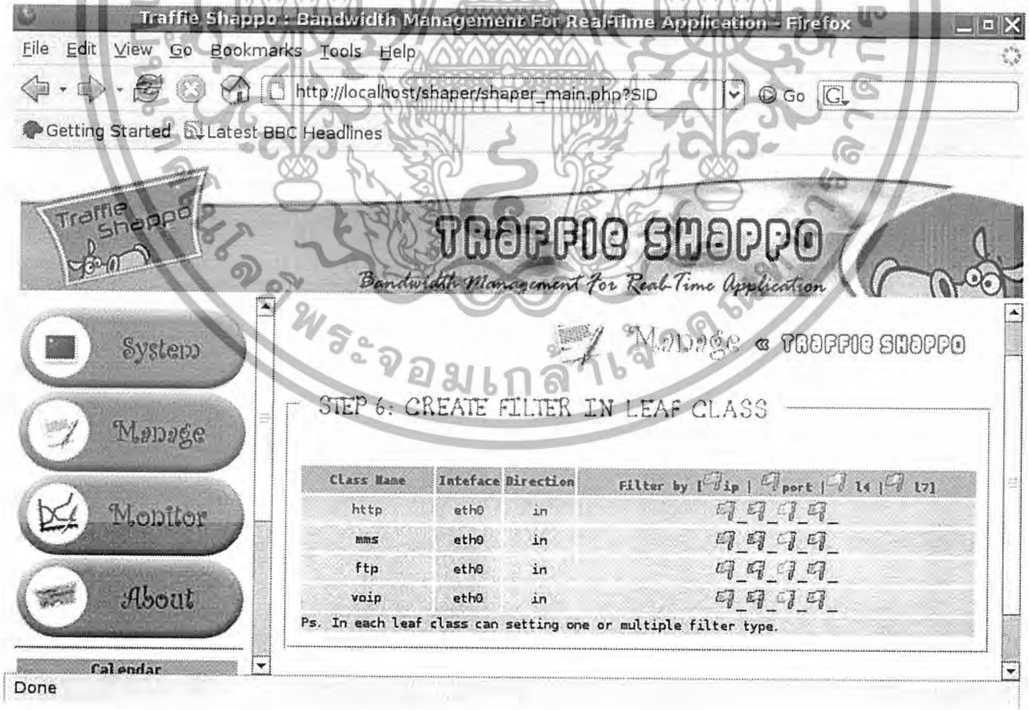


ภาพที่ ข.13 หน้าต่างแสดงการกำหนดค่าของคลาสระดับที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ ข.14 หน้าต่างแสดงการกำหนดค่าของคลาสระดับที่ 3



ภาพที่ ข.15 หน้าต่างแสดงข้อมูลคลาสสุดท้ายทั้งหมด
ในกฎนโยบายของการควบคุมปริมาณทราฟฟิกเพื่อกำหนดชนิดการกรองทราฟฟิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

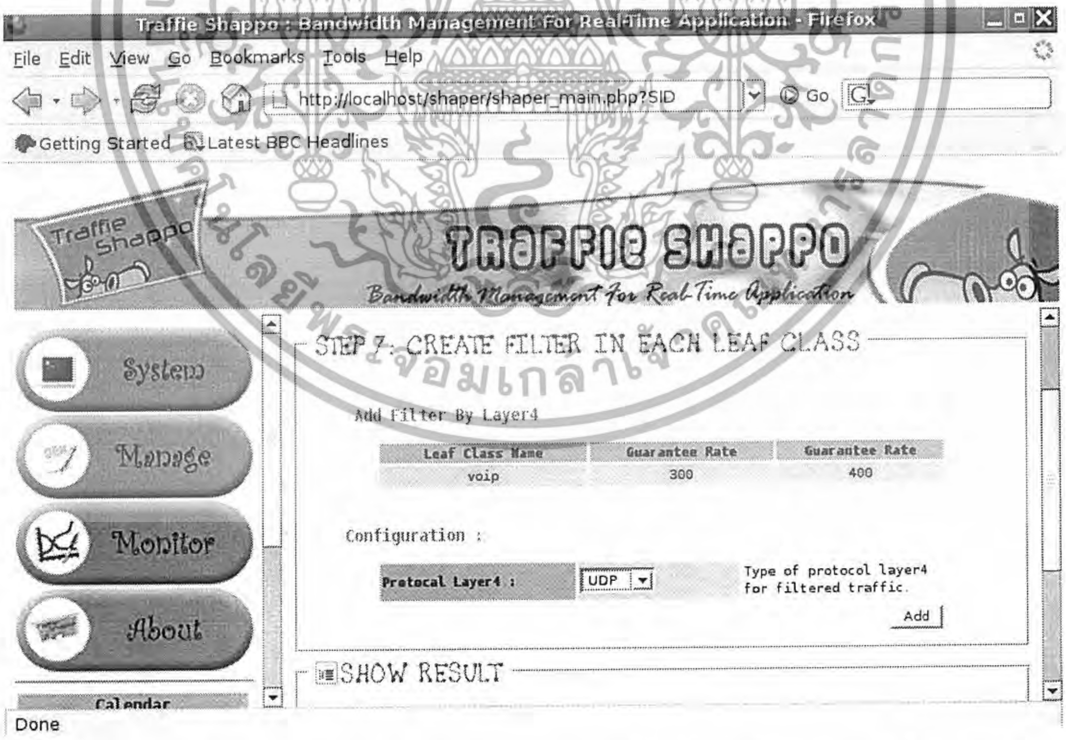
ภาพที่ ข.16 เป็นหน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้ไอพีแอดเดรส โดยผู้ใช้สามารถกรอกซับเน็ตมาสก์ (Subnet mask) ได้ 2 แบบ คือ เลขฐานสิบ เช่น 255.255.255.0 และเลขแสดงจำนวนบิต เช่น 24 เป็นต้น

ภาพที่ ข.17 เป็นหน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้พอร์ต โดยผู้ใช้สามารถกำหนดหมายเลขพอร์ต ได้ 3 แบบ คือ กรณีพอรต์เดียว เช่น 80 กรณีพอร์ตเป็นช่วง ทำการคั่นด้วยเครื่องหมายโคลอน เช่น 4004:4040 เป็นต้น และ กรณีใช้พอร์ตแบบผสมทั้งไม่เป็นช่วงและเป็นช่วง เช่น 80, 4004:4040 เป็นต้น

ภาพที่ ข.16 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้ไอพีแอดเดรส

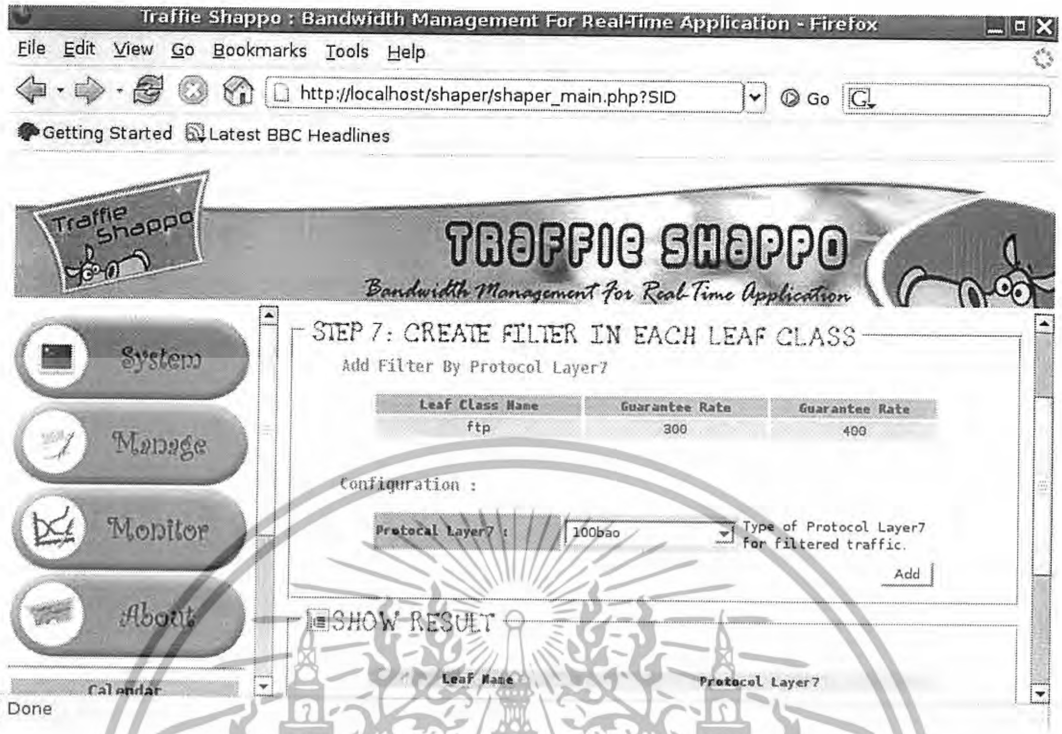
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ ข.17 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้พอร์ต



ภาพที่ ข.18 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นขนส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

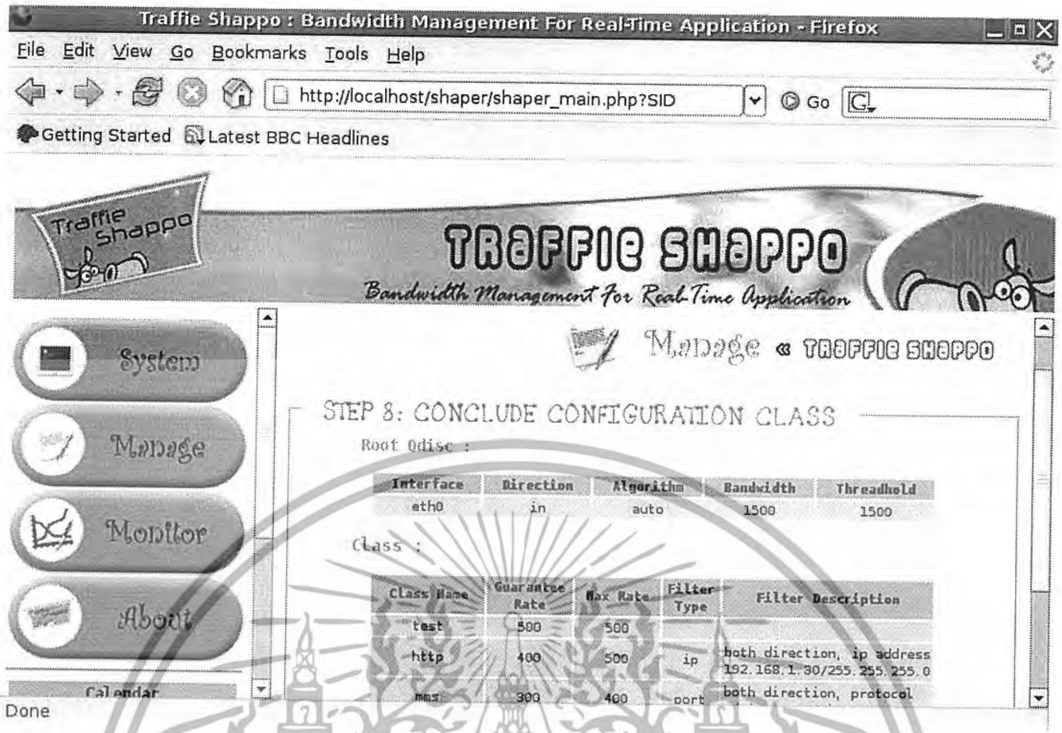


ภาพที่ ข.19 หน้าต่างแสดงการกำหนดการกรองทราฟฟิกโดยใช้โปรโตคอลระดับชั้นแอพพลิเคชั่น

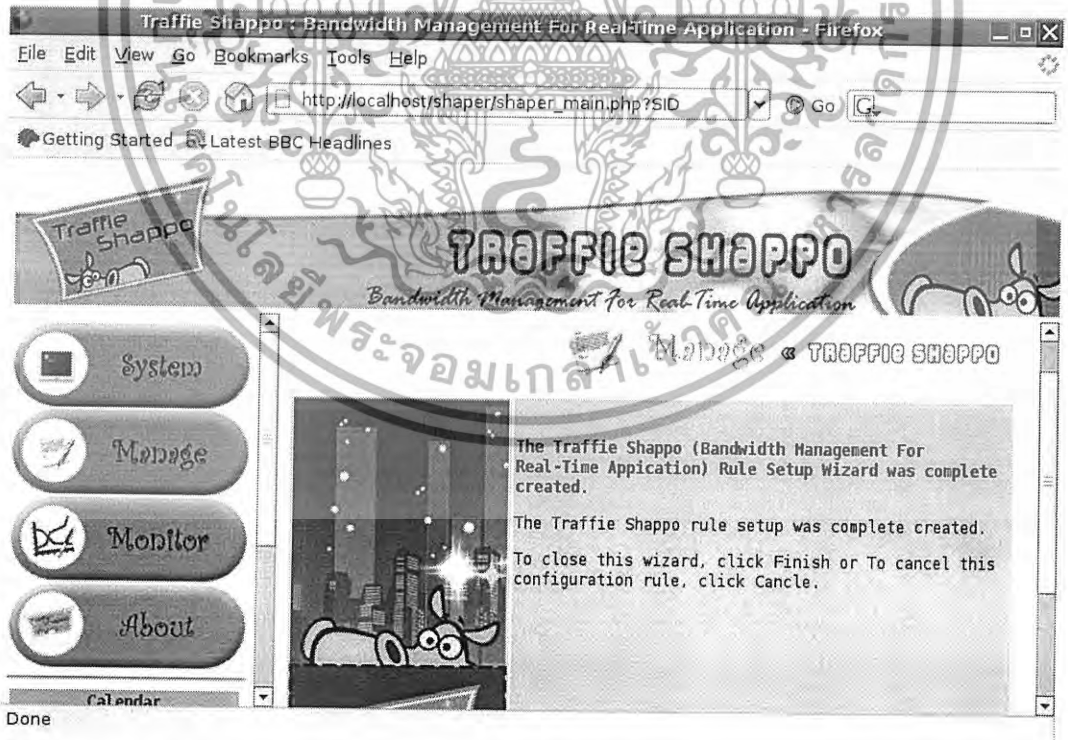
ภาพที่ ข.20 เป็นหน้าต่างแสดงการสรุปกฎนโยบายของการควบคุมปริมาณทราฟฟิกที่ผู้ใช้ได้กำหนดขึ้น โดยจะแสดงรายละเอียดข้อมูลทั้งหมดทั้งข้อมูลของการปรับแต่งทราฟฟิกและข้อมูลของคลาสบริการสำหรับตรวจเช็คความถูกต้อง ถ้าหากผู้ใช้ต้องการแก้ไขสามารถเลือก back เพื่อกลับไปแก้ไขข้อมูลที่ได้กำหนดไว้แล้วใหม่ได้

ภาพที่ ข.21 เป็นหน้าต่างแสดงการยืนยันการสร้างกฎนโยบายควบคุมปริมาณทราฟฟิกเพื่อทำการยืนยันการสร้างกฎเพื่อทำการสร้างสคริปต์กฎนโยบายควบคุมปริมาณทราฟฟิกต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ ข.20 หน้าต่างแสดงการสรุปกฎนโยบายของการควบคุมปริมาณทราฟฟิก



ภาพที่ ข.21 หน้าต่างแสดงการยืนยันการสร้างกฎนโยบายควบคุมปริมาณทราฟฟิก

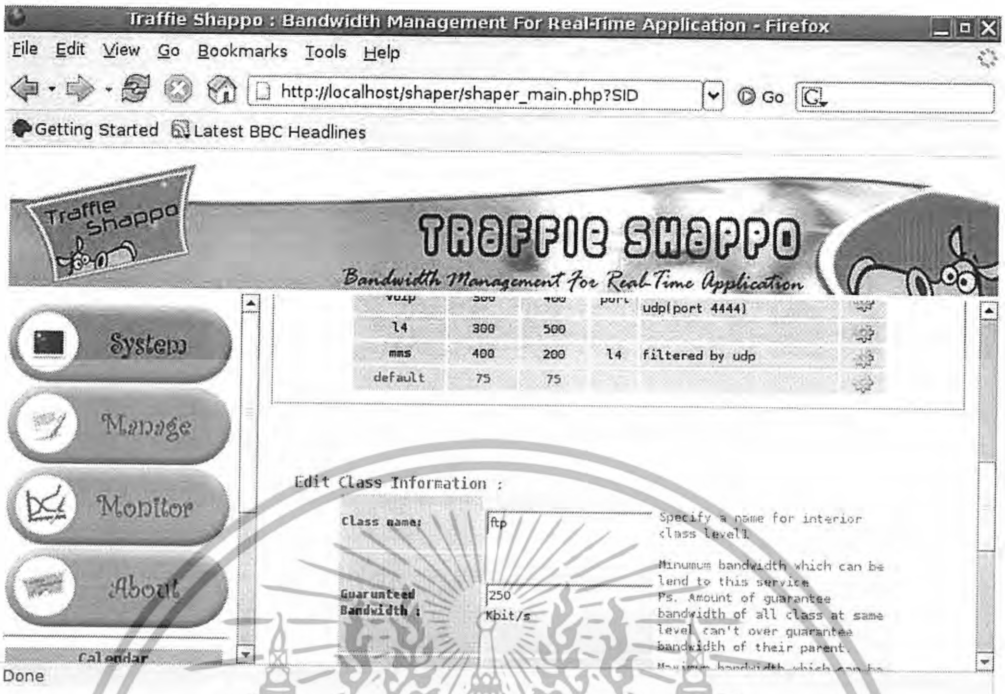
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การแสดงกฎนโยบายควบคุมปริมาณทราฟฟิก (Show Rule)

ภาพที่ ข.22 เป็นหน้าต่างแสดงกฎนโยบายควบคุมปริมาณทราฟฟิกทั้งหมดที่ได้กำหนดไว้ในเครื่องควบคุมปริมาณทราฟฟิกบนระบบปฏิบัติการลินุกซ์ โดยผู้ใช้สามารถแก้ไขค่าข้อมูลของการปรับแต่งทราฟฟิกและข้อมูลของคลาสบริการได้ โดยทำการเลือกที่สัญลักษณ์แก้ไขท้ายข้อมูลที่ผู้ใช้ต้องการแก้ไข ดังภาพที่ ข.23

ภาพที่ ข.22 หน้าต่างแสดงกฎนโยบายควบคุมปริมาณทราฟฟิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ ข.23 หน้าต่างแสดงการแก้ไขกฎนโยบายควบคุมปริมาณทราฟฟิก

4.3 การแสดงชุดคำสั่งกฎนโยบายควบคุมปริมาณทราฟฟิก (Show Command)

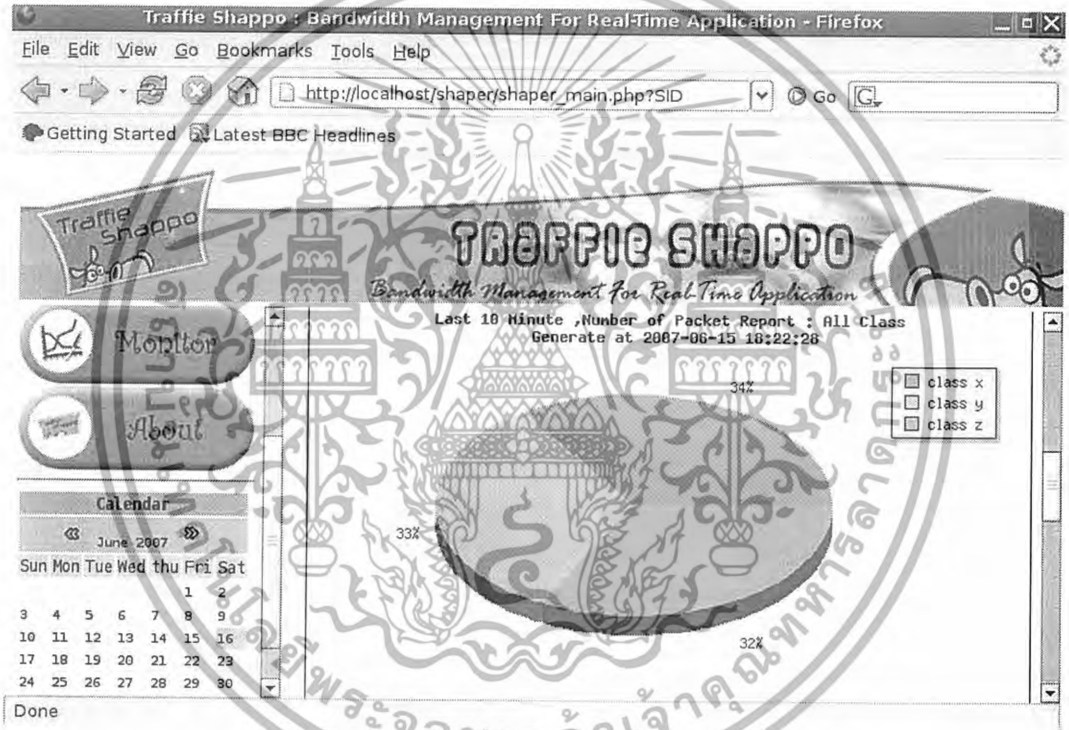
ภาพที่ ข.24 หน้าต่างแสดงการแก้ไขกฎนโยบายควบคุมปริมาณทราฟฟิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. การรายงานสถิติการใช้งานทางเครือข่าย (Monitor)

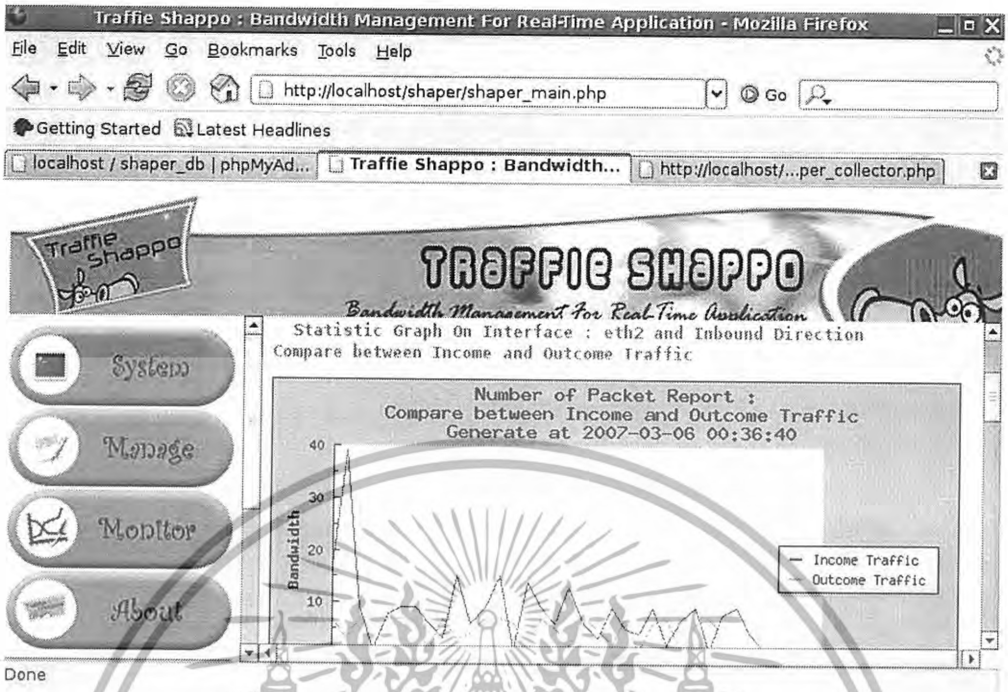
เป็นการแสดงผลสถิติการใช้งานบริการต่างๆในเครือข่าย แบ่งออกเป็น 2 ส่วนหลัก คือ รายงานสถิติการใช้งานทางเครือข่ายตามคลาส ดังภาพที่ ข.25 และรายงานสถิติการใช้งานเครือข่ายโดยภาพรวม ดังภาพที่ ข.26 โดยทั้งสองแบบจะทำการแสดงข้อมูลด้วยกราฟเส้นและกราฟวงกลมเหมือนกัน คือ แสดงปริมาณแพ็คเก็ตและแสดงปริมาณไบต์ข้อมูล

การรายงานสถิติการใช้งานทางเครือข่ายสามารถเลือกแสดงเป็นช่วงเวลาได้ คือ รายสิบนาทีล่าสุด รายหนึ่งชั่วโมงล่าสุด รายสี่ชั่วโมงล่าสุด รายวัน รายอาทิตย์ รายเดือน และ รายปี



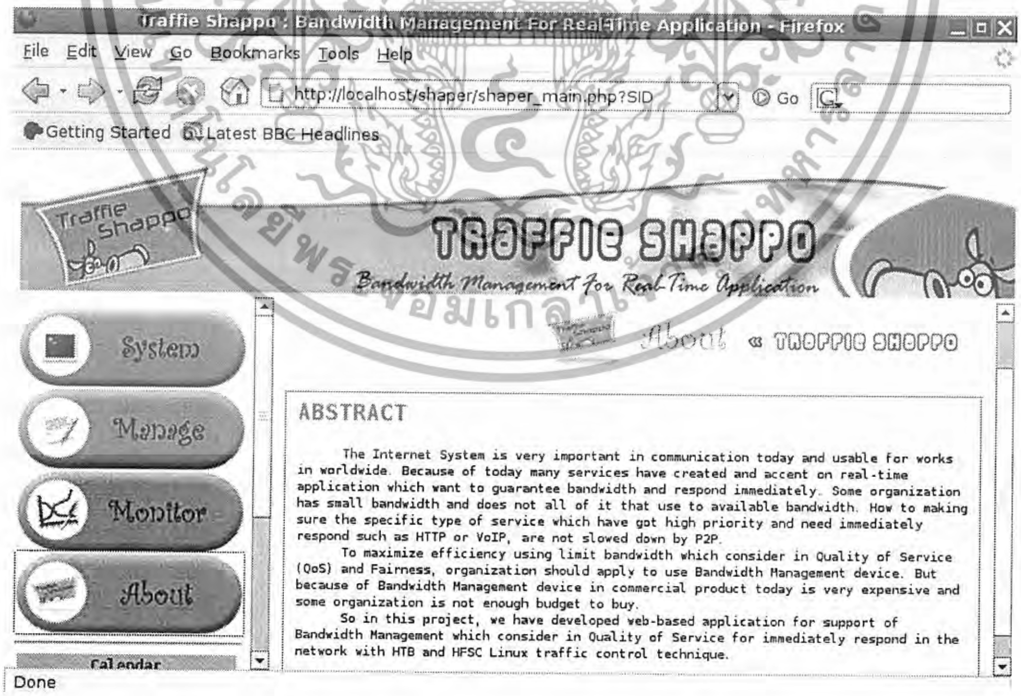
ภาพที่ ข.25 หน้าต่างแสดงรายงานสถิติการใช้งานทางเครือข่ายตามคลาส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ ข.26 หน้าต่างแสดงรายงานสถิติการใช้งานเครือข่ายโดยภาพรวม

6. การแสดงบทคัดย่อของโครงงาน (About)



ภาพที่ ข.27 หน้าต่างแสดงบทคัดย่อของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้