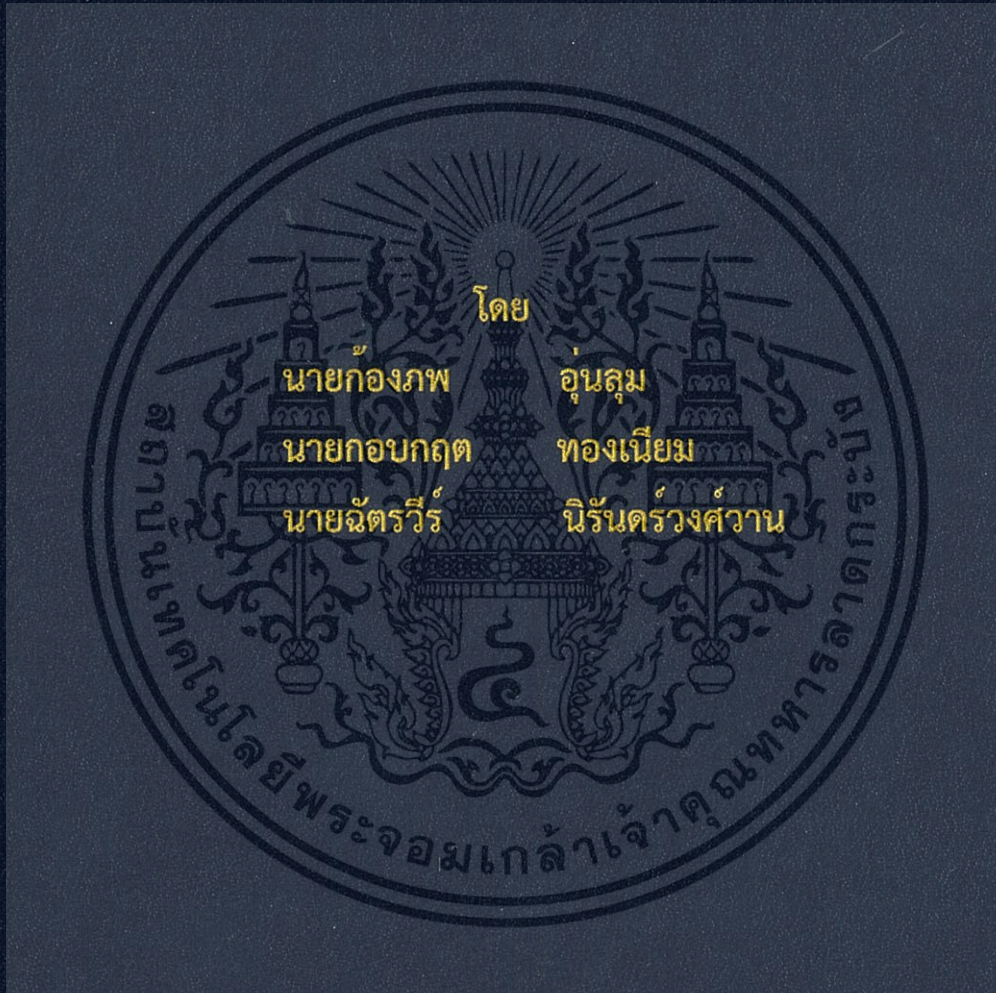


อุปกรณ์จัดการจราจรฟิสิกส์โดยโปรโตคอล

Traffic shaper by protocol



ปฏิญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2558

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

อุปกรณ์จัดการทราฟฟิกโดยโปรโตคอล

Traffic shaper by protocol



T144416

โดย

นายก้องภพ

อุ่นลุ่ม

นายกอบกฤต

ทองเนียม

นายฉัตรวีร์

นิรันดร์วงค์วาน

สงขหนุ

เลขทะเบียน 144416

รับเดือนปี 24 พ.ย. 2559

อาจารย์ที่ปรึกษา

รศ.ดร.สุวิพล ลิทธิชีวะภาค



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2558

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์จัดการทราฟฟิกโดยโปรโตคอล

Traffic shaper by protocol



โดย

นายก้องภพ อุ่นลุ่ม 55010063

นายกอบกฤต ทองเนียม 55010064

นายฉัตรวีร์ นีรันตร์วงศ์วาน 55010210

อาจารย์ที่ปรึกษา

รศ.ดร.สุวิพล ลิทธิชีวะภาค

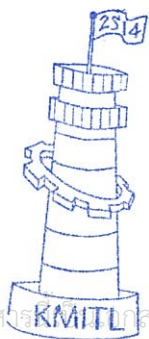
ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

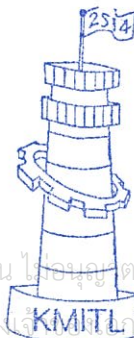
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2558




ผ่านการตรวจรูปเล่มแล้ว


.....
อาจารย์ที่ปรึกษา
12 / 10 / 59



ผ่านการตรวจชิ้นงานแล้ว


.....
กรรมการผู้ตรวจชิ้นงาน
12 / 10 / 59

ปริญญานิพนธ์ปีการศึกษา 2558

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง อุปกรณ์จัดการทราฟฟิกโดยโปรโตคอล

TRAFFIC SHAPER BY PROTOCOL

ผู้จัดทำ

1. นายก้องภพ อุ่นลุม 55010063
2. นายกอบกฤต ทองเนียม 55010064
3. นายฉัตรวีร์ นรินทร์วงศ์วาน 55010210

(รศ.ดร.สุวิพล ลิทธิชีวภาค)

อาจารย์ที่ปรึกษา

กิตติกรรมประกาศ

ปริญญานิพนธ์เล่มนี้สำเร็จลุล่วงไปได้ด้วยดี ด้วยคำแนะนำและคำปรึกษาจากอาจารย์ที่ปรึกษา รศ.ดร.สุวิพล ลิทธิชีวภาคและคุณภุชฌ์ธนิกร ศรีธนสาร ที่ได้ให้คำแนะนำในการทำปริญญานิพนธ์นี้ตลอดจนให้ความรู้และคำแนะนำเกี่ยวกับทางด้านโครงข่าย รวมไปถึงส่วนต่างๆที่เกี่ยวข้องพร้อมทั้งว่ากล่าวตักเตือนเพื่อให้ปริญญานิพนธ์นี้สำเร็จลุล่วงไปได้ด้วยดี

สุดท้ายนี้ผู้จัดทำขอขอบพระคุณ บิดา มารดา ตลอดจนครอบครัวที่ได้ให้การสนับสนุนช่วยเหลือผู้จัดทำและคอยให้กำลังใจมาอย่างสม่ำเสมออย่างต่อเนื่อง ผู้จัดทำหวังว่าปริญญานิพนธ์เรื่องนี้มีประโยชน์และสามารถนำไปประยุกต์ใช้สืบต่อไปได้



นายกิ่งภพ อุ่นลุม
 นายกอบกฤต ทองเนียม
 นายฉัตรวีร์ นรินทร์วงศ์วาน
 ผู้จัดทำ

อุปกรณ์จัดการจราจรไฟฟ้โดยโพรโตคอล
TRAFFIC SHAPER BY PROTOCOL

โดย นายก้องภพ อุ่นลุม 55010063
นายกอบกฤต ทองเนียม 55010064
นายฉัตรวีร์ นรินทร์รวงศ์วาน 55010210

อาจารย์ที่ปรึกษา รศ.ดร.สุวิพล ลิทธิชีวะภาค

บทคัดย่อ

ปฏิญานีพนนั้เป็นการพัฒนาแอปพลิเคชันเพื่อออกแบบการใช้บริการของ IPTABLE และ TC เพื่อควบคุมจราจรไฟฟ้ภายในโครงข่ายให้เพียงพอต่อความต้องการของผู้ใช้งาน โดยใช้การจำกัดแบนด์วิดท์ของจราจรไฟฟ้ประเภทอื่นๆลง เพื่อให้ประเภทของจราจรไฟฟ้ที่กำหนดไว้มีความเหมาะสมต่อการใช้งานและยังได้มีการรับประกันความเร็วขั้นต่ำของประเภทของจราจรไฟฟ้ที่สนใจอีกด้วยโดยกระบวนการทำงานของระบบถูกพัฒนาใช้ภาษาC++ และภาษา PHP ร่วมกันภายในกระบวนการ อีกทั้งยังได้มีการแสดงผลของปริมาณจราจรไฟฟ้ของระบบผ่านหน้าเว็บไซต์และสามารถที่จะเปลี่ยนค่าตัวแปรผ่านทางหน้าเว็บ เพื่อที่จะกำหนดจราจรไฟฟ้ในภายในโครงข่ายตามต้องการได้นอกยั้งนี้ยังได้มีการใช้ GSM MODULE เพื่อแจ้งเตือนไปยังผู้ดูแลระบบในกรณีที่เกิดปัญหาขึ้นภายในระบบอีกด้วย

ABSTRACT

This thesis is a development of application for designing services of IPTABLE and TC. The traffic in network is controlled by using bandwidth limitation of other applications to make the bandwidth of desired application suitably for using in network. Furthermore, there is a guarantee of minimum speed in system for making sure about the lowest possible speed. The operation of this system uses C++ and PHP to program together. Moreover, the traffic controller is shown in website for controlling and monitoring the traffic in system easily and GSM module is used for notifying to user in case of occurring problems in the system.



สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
สารบัญ	IV
สารบัญรูป	V
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตของปริญญานิพนธ์	2
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง	3
2.1 การจัดการทราฟฟิก	3
2.2 การแสดงผลผ่านหน้าเว็บ	15
บทที่ 3 การออกแบบและการจัดทำปริญญานิพนธ์	24
3.1 การออกแบบ	24
3.2 เครื่องมือที่ใช้ในการทดลอง	26
3.3 การจัดเก็บผลการทดลอง	26

สารบัญ (ต่อ)

	หน้า
บทที่ 4 ผลการทดลอง	35
4.1.1 ทดลองการดาวน์โหลดข้อมูลโดยใช้โปรโตคอล HTTP ทั้งแบบจำกัดและไม่จำกัดกราฟฟิก	35
4.1.2 ทดลองการดาวน์โหลดข้อมูลโดยใช้โปรโตคอล HTTP ทั้งแบบจำกัดและไม่จำกัดกราฟฟิก	36
4.1.3 ทดลองการแจ้งเตือนผ่าน SMS เมื่อขาดการเชื่อมต่อกับอินเทอร์เน็ต	37
บทที่ 5 สรุปผลและข้อเสนอแนะ	38
5.1 สรุปผล	38
5.2 ข้อเสนอแนะ	38
บรรณานุกรม	39
ภาคผนวก	40
ภาคผนวก ก SOURCE CODE ส่วนแสดงผลผ่านหน้าเว็บไซต์	
ภาคผนวก ข SOURCE CODE ส่วนประมวลผลของระบบ	

สารบัญรูป

รูปที่	หน้า	
2.1	QUEUING DISCIPLINE (QDISC)	11
2.2	หลักการทํางานของ HTB, HIERARCHICAL TOKEN BUCKET	12
2.3	ET-GSM SIM900B MODULE	14
2.4	สัญลักษณ์ของ HTML 5	16
2.5	ตัวอย่างของเว็บไซต์ที่พัฒนาด้วย BOOTSTRAPS	18
3.1	ภาพรวมระบบ	24
3.2	การทํางานของ IPTABLE	24
3.3	การทํางานของส่วนควบคุมแบนวิดท์	25
3.4	การใช้คำสั่งตั้งค่าพอร์ต ETH0 และ ETH1 ของเซิร์ฟเวอร์	26
3.5	กำหนดค่า IP ADDRESS ให้กับพอร์ต ETH0 และ ETH1 ของเซิร์ฟเวอร์	27
3.6	แสดงผลของการตั้งค่าพอร์ต ETH0	27
3.7	แสดงผลของการตั้งค่าพอร์ต ETH1	27
3.8	การทดสอบการทํางานของพอร์ต ETH0	28
3.9	การทดสอบการทํางานของพอร์ต ETH1	28
3.10	การใช้คำสั่งตั้งค่าระบบ DHCP ของเซิร์ฟเวอร์	28
3.11	กำหนดค่าช่วง IP ADDRESS ให้ระบบ DHCP ของเซิร์ฟเวอร์	29
3.12	PACKAGE DHCP OFFER ของ DHCP เซิร์ฟเวอร์ ที่ตรวจพบโดยโปรแกรม WIRESHARK	29
3.13	หมายเลข IP ADDRESS ที่เครื่องลูกข่ายได้รับจาก DHCP เซิร์ฟเวอร์	30
3.14	การใช้คำสั่งตั้งค่าระบบ NAT ของเซิร์ฟเวอร์	30
3.15	การตั้งค่าไฟล์ SYSCTL.CONF ของเซิร์ฟเวอร์	31
3.16	การทดสอบการทํางานของระบบ NAT	31
3.17	TRAFFIC CONTROLLER	32
3.18	การดาวน์โหลดไฟล์โดยไม่มีกำหนดกราฟฟิคของ HTTP	32
3.19	การดาวน์โหลดไฟล์โดยมีกำหนดกราฟฟิคของ HTTP	33

รูปที่		หน้า
3.20	การดาวน์โหลดไฟล์โดยมีการรับประกันความเร็วต่ำสุด	33
3.21	การอัปโหลดไฟล์โดยมีการรับประกันความเร็วต่ำสุด	34
4.1	กราฟแสดงปริมาณทราฟฟิกเมื่อทำการดาวน์โหลดไฟล์ของ HTTP	35
4.2	กราฟแสดงปริมาณทราฟฟิกเมื่อทำการดาวน์โหลดไฟล์ของ HTTPS	36
4.3	SMS ที่ได้รับจาก GSM MODULE	37



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากในปัจจุบันนี้มีแอปพลิเคชันที่ให้บริการบนโครงข่ายอินเทอร์เน็ตอยู่หลากหลายชนิด ซึ่งมีทั้งส่วนที่จำเป็นและส่วนที่ไม่จำเป็น โดยอาจก่อให้เกิดอันตรายหรือการรบกวนต่อระบบโครงข่ายภายในองค์กรได้ ซึ่งทำให้เกิดปัญหาในการควบคุมดูแลและจัดการระบบภายในองค์กร อีกทั้งในบางครั้งยังมีการใช้งานแบนด์วิดธ์จำนวนมากและอาจก่อให้เกิดปริมาณทราฟฟิกภายในโครงข่ายเพิ่มมากขึ้น ซึ่งถ้าไม่มีการควบคุมดูแลก็จะทำให้เกิดปัญหาการใช้งานโครงข่ายอินเทอร์เน็ตภายในองค์กรได้ กระบวนการจัดการทราฟฟิกบนโครงข่ายเป็นการกำหนดวิธีการในการเข้าถึงโครงข่ายของแต่ละแพ็คเกจ โดยจะดูว่าแต่ละแพ็คเกจที่เข้ามามีความต้องการที่จะใช้บริการอะไรเข้ามาทางอินเทอร์เน็ตเพลสโตน ใช้เส้นทางอย่างไรเป็นต้น เพื่อที่จะดูความเหมาะสมในการออกแบบระบบการควบคุมทราฟฟิกแล้วจะได้แก้ไขปัญหาก็ถูกจุดอีกทั้งยังช่วยให้ผู้ดูแลโครงข่ายสามารถที่จะควบคุมการเข้าใช้งานแบนด์วิดธ์ได้ เพื่อช่วยให้องค์กรสามารถประหยัดค่าใช้จ่ายที่ต้องจ่ายสำหรับค่าเช่าแบนด์วิดธ์ได้อีกด้วย

อุปกรณ์ที่ใช้สำหรับควบคุมปริมาณทราฟฟิกเหล่านั้นนั้น จะต้องมีประสิทธิภาพและมีความเหมาะสมกับพฤติกรรมของผู้ใช้งานภายในโครงข่ายเพื่อให้เหมาะสมแก่ลักษณะการใช้งาน โดยวิธีการที่ใช้ในการควบคุมปริมาณทราฟฟิกนั้นมีอยู่หลายวิธีการ เช่น ควบคุมปริมาณแบนด์วิดธ์ ควบคุมเวลาที่เข้าใช้งาน ควบคุมจำนวนผู้ใช้งาน แต่วิธีการที่เราใช้นั้นเป็นการควบคุมการเข้าใช้บริการของแอปพลิเคชันต่างๆ โดยที่จะไปควบคุมปริมาณโปรโตคอลที่เข้ามาใช้บริการ เพื่อที่จะทำให้ไม่เกิดการให้บริการอย่างใดอย่างหนึ่งมากเกินไปนั่นเอง

เพราะฉะนั้นจึงมีแนวคิดที่จะสร้างอุปกรณ์ ที่มีความสามารถในการจัดการกับทราฟฟิกที่อยู่ภายในโครงข่าย โดยใช้วิธีการจำกัดการเรียกใช้บริการของแอปพลิเคชันต่างๆ ซึ่งจะไปกำหนดที่ตัวโปรโตคอลเป็นหลักเพื่อให้ผู้ดูแลโครงข่ายสามารถที่จะควบคุมปริมาณการเข้าใช้บริการต่างๆได้อีกทั้งยังได้มีการจัดทำหน้าเว็บสำหรับไว้คอยดูทราฟฟิกที่เกิดขึ้นภายในโครงข่ายและมีการแจ้งเตือนผ่านโทรศัพท์มือถือในกรณีที่อุปกรณ์มีปัญหาการเชื่อมต่ออินเทอร์เน็ตอีกด้วย

1.2 วัตถุประสงค์

- 1) เพื่อจำกัดปริมาณ Traffic ในระบบโครงข่าย
- 2) เพื่อศึกษาการใช้งานระบบปฏิบัติการลินุกซ์

1.3 ขอบเขตของปริณญาณินพนธ์

- 1) สามารถที่จะจำแนก Traffic ตามโปรโตคอล หรือ source ip
- 2) ติดตั้งระบบปฏิบัติการลินุกซ์ที่เครื่อง PC ทั้งหมด 1 เครื่อง



บทที่ 2

ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 การจัดการทราฟฟิก

การจัดการทราฟฟิกในโครงข่ายด้วยโปรโตคอลนั้น จะต้องมีการทำการติดตั้งระบบที่มีประสิทธิภาพและมีความเหมาะสมเพื่อที่จะวัดค่าของปริมาณของแพ็คเก็ตที่วิ่งอยู่ในโครงข่ายที่สนใจ และนำค่าที่ได้ไปวิเคราะห์ผล ซึ่งระบบดังกล่าวนี้เราได้มีการติดตั้งและตั้งค่าต่างๆดังต่อไปนี้

2.1.1 IPTABLES

2.1.1.1 ความหมายของ IPTABLES

IPTABLES คือ อุปกรณ์ที่ใช้เพิ่มระดับความปลอดภัยให้กับโครงข่ายโดยเป็นตัวป้องกันระหว่างโครงข่ายภายในกับโครงข่ายภายนอกและยังคอยช่วยป้องกันแพ็คเก็ตอันตรายไม่ให้วิ่งเข้ามายังโครงข่ายภายในได้ เราสามารถใช้ Linux ทำเป็น Firewall IPTABLES ได้โดยการติดตั้ง IPTABLES เป็นคำสั่งบนลินุกซ์ที่ใช้ปรับแต่งให้กับระบบเพื่อให้สามารถกรองแพ็คเก็ตและสามารถทำการแปลงแอดเดรสได้อีกด้วย โดยใช้คำสั่งที่มีมาพร้อมกับลินุกซ์เคอร์เนลเวอร์ชัน 2.4 ขึ้นไปและสามารถใช้งานได้โดยไม่ต้องคอมไพล์เคอร์เนล

2.1.1.2 รูปแบบการใช้งาน IPTABLES

จะมีรูปแบบการใช้งานโดยกฎ (rule) ที่เขียนขึ้นจะเป็นตัวบอกเคอร์เนลว่าให้กระทำการอะไรในกรณีที่พบแพ็คเก็ตตรงตามที่ระบุไว้หมายถึงตารางที่ต้องการระบุมี 3 ตารางคือ ฟิวเตอร์เทเบิล(Filter table) แนตเทเบิล(Nat table) แมงเกิลเทเบิล(Mangle table) เช่น

1) iptables -t nat หมายถึงให้ทำงานกับแนตเทเบิลในกรณีที่ไม่ได้ระบุตารางไอพีเทเบิล จะถือว่าคำสั่งดังกล่าวระบุถึงฟิวเตอร์เทเบิล

2) <command> จะเป็นตัวสั่งให้ไอพีเทเบิลทำในสิ่งที่ต้องการ เช่น iptables -A INPUT ซึ่งหมายถึงให้สร้างกฎต่อท้ายอินพุตเชน(INPUT chain)ในฟิวเตอร์เทเบิล

3) <match> เป็นส่วนที่ใช้ตรวจสอบว่าแพ็คเก็ตมีข้อมูลตรง (match) กับที่ระบุไว้หรือไม่ เช่น มีไอพีแอดเดรสต้นทาง (source ip address) เป็น 1.2.3.4

4) <target/jump> เป็นตัวระบุว่าจะเมื่อเจอแพ็คเก็ตที่ตรง (match) ก็จะทำตามที่เราระบุไว้ เช่น ถ้าแพ็คเก็ตใดมีไอพีแอดเดรสต้นทาง(source ip address) เป็น 1.2.3.4 ให้ทิ้งแพ็คเก็ตนั้น (DROP packet)

2.1.1.3 รูปแบบการ match

ต้องมีการตั้งเงื่อนไขของการ match ซึ่งจะต้องอาศัยความเข้าใจในเรื่อง IP, TCP, UDP, และ ICMP มาบ้างพอสมควรจึงจะสามารถตั้งเงื่อนไขที่เหมาะสมและตรงตามความต้องการได้ ซึ่งมีรายละเอียดดังนี้

1) การระบุไอพีแอดเดรสของต้นทางและปลายทางสามารถที่จะระบุไอพีแอดเดรสต้นทาง(source ip address)ของแพ็คเก็ตได้ โดยใช้ -s หรือ --source หรือ --src สำหรับไอพีแอดเดรสปลายทาง (destination ip address) ก็ใช้ -d หรือ --destination หรือใช้ -dst

2) การทำ Inversion ในบางกรณีนั้นหากต้องการระบุให้เป็นอินเวิร์ส (inverse) เช่น อนุญาตให้ทุกไอพียกเว้นไอพีที่ระบุไว้ ซึ่งการใช้คำสั่งดังกล่าวสามารถทำได้โดยใช้เครื่องหมาย ! นำหน้า argument ที่ต้องการ (เครื่องหมาย ! หมายถึง NOT) เช่น -p ! TCP ซึ่งจะตรงกับโปรโตคอลทุกๆ ตัวที่ไม่ใช่ ทีซีพี หรือ -s ! local host ซึ่งหมายถึงแพ็คเก็ตที่มีไอพีแอดเดรสต้นทางอื่นๆ ยกเว้น local host (127.0.0.1)

3) การระบุโปรโตคอลสามารถระบุโปรโตคอลที่ต้องการได้ดังต่อไปนี้คือ ทีซีพี (TCP), ยูดีพี (UDP), ไอซีเอ็มพี (ICMP) หรือสามารถใช้ตัวเลขแทนได้(สำหรับ *NIX อ้างอิงได้จาก /etc/protocols)และยังสามารถใช้ได้ทั้งตัวอักษรพิมพ์เล็กหรือพิมพ์ใหญ่ (ใช้ได้ทั้ง tcp และ TCP) เช่น -p TCP หรือ -p ! tcp

4) การระบุอินเตอร์เฟซทำได้โดยใช้พิมพ์ -i หรือ --in-interface ตามด้วยชื่ออินเตอร์เฟซ (interface) ใช้เพื่อระบุ incoming interface ซึ่งหมายถึงว่า แพ็คเก็ต ที่จะตรงกับกฎนี้ต้องเข้ามาจากอินเตอร์เฟซ ที่กำหนด เช่น -i eth0 หมายความว่าทุกแพ็คเก็ตที่เข้ามาทาง eth0 จะตรงกับกฎนี้ ทั้งนี้ชื่ออินเตอร์เฟซที่สามารถใช้นั้นสามารถตรวจสอบได้โดยใช้คำสั่ง ifconfig และ -o หรือ --outinterface ตามด้วยชื่อของอินเตอร์เฟซ ใช้เพื่อระบุ outgoing interface ซึ่งหมายถึงว่า แพ็คเก็ตที่จะตรงกับกฎนี้กำลังจะเดินทางผ่านอินเตอร์เฟซที่ระบุไว้เช่น -o eth1 หรือ -o ! eth1

2.1.1.4 การทำ Fragment Packet

ในการส่งข้อมูลใน ip network นั้นเป็นเรื่องปกติที่จะเกิดการแฟรกเมนต์ของแพ็คเก็ต เนื่องจากขนาดของแพ็คเก็ตมีขนาดใหญ่เกินไปที่จะส่งไปในครั้งเดียวจำเป็นต้องมีการแบ่งแพ็คเก็ตออกเป็นหลายๆชิ้นย่อยทยอยส่งไป ซึ่งเรียกกระบวนการนี้ว่าการทำแฟรกเมนต์โดยเครื่องปลายทางจะทำหน้าที่ประกอบแฟรกเมนต์แพ็คเก็ต (fragment packet) รวมกันเป็นแพ็คเก็ตที่สมบูรณ์ดั้งเดิม ข้อมูลที่เป็นแฟรกเมนต์แพ็คเก็ตนั้นจะมีเฮดเดอร์(header) ที่สมบูรณ์แค่แพ็คเก็ตแรก

เท่านั้น ส่วนแพ็คเก็ตที่ตามมาจะมีแค่เฮดเดอร์บางส่วน คือ ไอพีแอดเดรสเท่านั้นไม่มีข้อมูลของโปรโตคอลแนบมาด้วย ดังนั้นการตรวจสอบข้อมูลเฮดเดอร์ของทีซีพี ยูดีพี ไอซีเอ็มพี จึงไม่สามารถทำได้ในแพ็คเก็ตที่สองเป็นต้นมา หากมีการแนบ (NAT) แพ็คเก็ตประเภทแฟรกเมนต์แพ็คเก็ตจะถูกประกอบเข้าด้วยกันจนสมบูรณ์ก่อนที่แพ็คเก็ตจะเข้าไปถึงแพ็คเก็ตฟิลเตอร์ริง(packet filtering) ดังนั้นจึงไม่มีความจำเป็นที่จะต้องกังวลเกี่ยวกับแฟรกเมนต์แพ็คเก็ต ในกรณีที่ไม่ได้ใช้เน็ตแพ็คเก็ตประเภทแฟรกเมนต์แพ็คเก็ตก็จะไม่ถูกประกอบเข้าด้วยกัน ไอพีเทเบิลจะมีกระบวนการในการทำงานกับแฟรกเมนต์แพ็คเก็ตดังนี้ หลังจากที่แฟรกเมนต์แพ็คเก็ตแรกผ่านเข้ามาแล้วไอพีเทเบิลสามารถตรวจสอบได้ว่าจะอนุญาตให้ผ่านหรือไม่ให้ผ่าน ในขณะที่แฟรกเมนต์แพ็คเก็ตที่สองและหลังจากนั้นที่ตามมาจะไม่สามารถนำไปตรวจสอบกับกฎใดๆเลย เช่น -p TCP --sport www หรือแม้แต่ว่า -p TCP --sport ! www แต่ยังสามารถเขียนกฎให้ตรวจสอบทั้งแฟรกเมนต์แพ็คเก็ตตัวที่สองและที่ตามมาหลังจากนั้นได้ด้วยการใช้ -f หรือ -- fragment ทั้งนี้อาจจะเขียนในทางตรงข้ามคือไม่ต้องตรวจสอบแฟรกเมนต์แพ็คเก็ตที่สองและหลังจากนั้นโดยใช้ ! -f ก็ได้โดยปกติแล้วมักจะปล่อยให้แฟรกเมนต์แพ็คเก็ตผ่านไปเนื่องจากถ้าสามารถทั้งแฟรกเมนต์แพ็คเก็ตแรกได้แล้วแพ็คเก็ตทั้งหมดก็จะไม่สามารถถูกประกอบที่เครื่องปลายทางได้ แต่ทั้งนี้แฟรกเมนต์แพ็คเก็ตที่ถูกปล่อยให้ดังกล่าวอาจจะทำให้เครื่องที่รับนั้นเกิดการชะงักหรือเกิดความเสียหายได้หรืออาจจะเกิดการโจมตีแบบดีไนอัลออฟเซอร์วิส(Denial of Service) โดยใช้แฟรกเมนต์แพ็คเก็ตได้

2.1.2 Network Address Translation (NAT)

2.1.2.1 Network Address Translation (NAT)

NAT เป็นวิธีการหนึ่งในการแปลงและแปลไอพีแอดเดรสของเครือข่ายภายในให้เป็นไอพีแอดเดรสซึ่งเป็นที่ยอมรับและใช้ในการติดต่อสื่อสารบนโครงข่ายอินเทอร์เน็ต ซึ่ง NAT ให้ประโยชน์มากมายหลายอย่าง เช่น สามารถใช้ไอพีแอดเดรสที่ตั้งขึ้นมาเองได้ (ซึ่งเป็น ไอพีแอดเดรสที่ไม่ต้องจดทะเบียนบนอินเทอร์เน็ต) เพียงแต่ใช้ไอพีแอดเดรสที่ให้บริการอินเทอร์เน็ตให้มากพออีกทั้งยังสามารถซ่อนไอพีแอดเดรสที่อยู่ในเครือข่าย(ไอพีแอดเดรสที่กำหนดขึ้นเอง) ได้และยังไม่จำเป็นที่จะต้องไปอ้างแอดเดรสเลขหมายซ้ำๆ เมื่อต้องการติดต่อกับอินเทอร์เน็ตหรือเครือข่ายขององค์กร

2.1.2.2 หลักการทำงานของ NAT

NAT เสมือนเป็นระบบการอินเทอร์เน็ตเฟสกับอินเทอร์เน็ตที่ไม่ขึ้นอยู่กับ Protocol และ Application รวมทั้งอุปกรณ์ Hardware ใดๆ ซึ่งหมายความว่า NAT สามารถถูกนำมาประยุกต์ใช้งานกับเราเตอร์หรือคอมพิวเตอร์ที่ทำหน้าที่เป็นเราเตอร์ใดๆ ที่มีลักษณะการเชื่อมต่อโดยมีด้านหนึ่งสำหรับเครือข่ายภายในและอีกด้านหนึ่งกับเครือข่ายภายนอก ดังเช่น

อินเทอร์เน็ต ตัวอย่างการเชื่อมต่อเช่น การติดตั้ง NAT ที่Borderเราเตอร์ซึ่งเป็นเราเตอร์ที่เชื่อมต่อเครือข่ายย่อยๆ ต่างๆ ภายในองค์กรกับเครือข่ายภายนอก

2.1.2.3 จุดประสงค์ของการใช้ NAT

1) หากมีเครือข่ายภายในที่มีการจัดตั้งConfigured ให้ใช้ไอพีแอดเดรสที่ไม่ได้จดทะเบียนให้ใช้งานบนอินเทอร์เน็ตหรือจากISP การใช้NATจะช่วยให้เกิดการเชื่อมต่อเครือข่ายภายในองค์กรกับอินเทอร์เน็ตซึ่งถือว่าเป็นเครือข่ายสาธารณะที่มีความปลอดภัยเนื่องจาก NATจะทำการแปลงไอพีแอดเดรสของเครือข่ายในองค์กรให้ใช้ไอพีแอดเดรสที่จดทะเบียนถูกต้องบนอินเทอร์เน็ตเสียก่อนที่จะส่งข้อมูลข่าวสารออกไปที่อินเทอร์เน็ต

2) คอมพิวเตอร์ภายในองค์กรที่ได้รับการจัดตั้งไอพีแอดเดรสแบบ Static NAT(เช่น Mail Servers เป็นต้น)สามารถที่จะถูกแปลแอดเดรสเมื่อติดต่อกับภายนอกโดยไม่ต้องมีการ Update ข้อมูลบันทึกบน NAT

3) NATสามารถเก็บรักษาไอพีแอดเดรสที่ได้จดทะเบียนถูกต้องจำนวนหนึ่งไว้เพื่อให้เครือข่ายภายในใช้เพื่อติดต่อกับอินเทอร์เน็ต

4) การออกแบบเครือข่ายภายใต้NATสามารถทำได้ง่ายเนื่องจากมีแบบแผนการอ้างแอดเดรสอย่างรัดกุมจำกัด (ไม่มีข้อจำกัดในการเติบโตของเครือข่ายซึ่งอาจนำไปสู่ไอพีแอดเดรสที่ไม่พอเพียง)

2.1.3 iftop

2.1.3.1 หลักการของ iftop

iftop เป็นเครื่องมือที่ช่วยในการดูและระบบที่อยู่ใน command line ที่ใช้ตรวจสอบการเชื่อมต่อในโครงข่ายโดยที่จะตรวจสอบการเชื่อมต่อและจะแสดงปริมาณแบนด์วิดท์ที่ใช้อยู่ในปัจจุบัน ซึ่งจะมีคำสั่งต่างๆ ดังต่อไปนี้

- 1) iftop -i ethX ใช้ดู traffic ในวงขา Lan
- 2) iftop -i pppX ใช้ดู traffic ในวงขา WAN
- 3) h แสดงข้อความนี้
- 4) n ไม่ให้ค้นหา hostname
- 5) N ไม่ให้เปลี่ยนเลขพอร์ตที่ให้บริการ
- 6) b ไม่ให้แสดงกราฟรูปแท่งของกราฟฟิค
- 7) B แสดงแบนด์วิดท์ในหน่วย bytes

2.1.4 Wireshark

2.1.4.1 หน้าที่

Wireshark มีลักษณะคล้าย Tcpdump แต่ Wireshark มีลักษณะที่ชัดเจนกว่าและมีระบบการจัดแยกคัดกรองข้อมูลมากกว่า Wiresharkอนุญาตให้ผู้ใช้สามารถเน้นระบบต่างๆผ่านไปยังเครือข่ายโดยใส่ network interface ไปยัง promiscuous mode

2.1.4.2 ลักษณะพิเศษ

1)Wireshark เป็นซอฟต์แวร์มีลักษณะเฉพาะใช้ Pcap เชื่อมโยงแพ็คเก็ต ดังนั้นมันสามารถใช้กับแพ็คเก็ตเฉพาะรุ่นที่มีเครือข่าย Pcap

2)สามารถเชื่อมต่อข้อมูลผ่านสายไฟจากเครือข่ายภายในที่พังกอาศัยหรือจากแฟ้มข้อมูลที่บันทึกไว้

3)สามารถอ่านข้อมูลจากจำนวนเครือข่ายประเภทต่างรวมทั้ง Ethernet ,IEEE 802.11 , PPP และ loopback

4)Captured network data สามารถ browsed via a GUI

5)สามารถแก้ไขโปรแกรมได้

6)ใช้ Ping - ins สำหรับ New protocols

7)การเชื่อมโยงUSB สามารถยึดเข้ากับWiresharkซึ่ง

ลักษณะเฉพาะนี้รองรับเฉพาะภายใต้ระบบ Linuxเท่านั้น รูปโปรแกรมtcpdumpที่เป็นต้นกำเนิดของระบบ Wireshark เป็นรูปแบบ libpcap จึงสามารถรองรับด้วยlibpcap and WinPcap ดังนั้นมันจึงสามารถอ่านไฟล์จากแอปพลิเคชันอื่นที่ใช้ format เดียวกัน เช่น tcpdump และ CA NetMaster และยังสามารถอ่านข้อมูลจากเครือข่ายระบบอื่นๆเช่น snoop, Network General's Sniffer, and Ms Network Monitor

2.1.4.3 ความปลอดภัย

การตรวจจับการเชื่อมโยงเครือข่ายจากศูนย์กลางระบบต้องใช้สิทธิพิเศษสำหรับข้อมูลบางส่วนเพราะเหตุผลนี้เวอร์ชันเก่าของ Ethereal / Wireshark และ tethereal / tshark มักจะถูกใช้โดยผู้มีเอกลิทธิเท่านั้น ในระบบไร้สายมีความเป็นไปได้ที่จะใช้เครื่องมือรักษาความปลอดภัยโดยระบบ Aircrack เพื่อตรวจจับ IEEE 802.11 และอ่านผลข้อมูลด้วย Wireshark

2.1.5 routing table

2.1.5.1 ค่าใน routing table

ในการที่จะได้มาซึ่งค่าใน routing table นั้นจะมีวิธีการได้มาซึ่งข้อมูลอยู่ด้วยกัน 2 แบบ ดังต่อไปนี้

1) Static Route คือ การเพิ่มเส้นทางใน Routing Table ด้วยผู้ดูแลเน็ตเวิร์คเพื่อบอกให้เราท์เตอร์ทราบว่าถ้าต้องการจะส่งข้อมูลไปที่ Subnet Address ใดจะต้องส่งผ่านเราท์เตอร์ตัวไหนค่าที่ถูกป้อนเข้าไปในตารางเลือกเส้นทางนี้มีค่าที่ตายตัว ดังนั้นการเปลี่ยนแปลงที่เกิดขึ้นใดๆบนเครือข่ายจะต้องให้ผู้ดูแลเน็ตเวิร์คเข้ามาจัดการทั้งหมดซึ่งเหมาะสำหรับเครือข่ายที่มีขนาดเล็กเนื่องจากสามารถแน่ใจว่าข้อมูลข่าวสารจะต้องวิ่งไปบนเส้นทางที่กำหนดไว้ให้ตายตัว ไม่ต้องใช้ Software เลือกเส้นทางใดๆทั้งสิ้นและช่วยประหยัดการใช้แบนด์วิดธ์ของเครือข่ายลงได้มาก

2) Dynamic Route เป็นการใช้ซอฟต์แวร์ที่ติดตั้งมากับเราท์เตอร์เพื่อทำหน้าที่แลกเปลี่ยนข้อมูลข่าวสารที่เกี่ยวกับการเลือกเส้นทางระหว่างเราท์เตอร์ หลักการทำงานคือ เราท์เตอร์จะส่ง routing table ที่สมบูรณ์ของตัวเองให้กับเราท์เตอร์เพื่อนบ้านเรียกว่ามี Routing Protocol ที่ใช้ในการแลกเปลี่ยน routing table โดยที่ผู้ดูแลเครือข่ายไม่ต้องแก้ไขข้อมูล routing table ในเราท์เตอร์เลย เหมาะสำหรับเครือข่ายขนาดใหญ่เพราะเราท์เตอร์สามารถจัดการหาเส้นทางเองหากมีการเปลี่ยนแปลงของเครือข่ายเกิดขึ้น โดย Routing Protocol จะมี Distance Vector และ Link State ซึ่ง routing protocol ทั้งสองประเภทจะมีจุดประสงค์ที่เหมือนกันคือการทำให้เราท์เตอร์ปัจจุบันมีตาราง routing table ที่ประกอบด้วยเส้นทางที่ดีที่สุดที่สามารถส่งข้อมูลไปถึงซบเน็ตแอดเดรสปลายทางทั้งหมดได้แต่สิ่งที่แตกต่างกันก็คือวิธีการที่จะทำให้จุดประสงค์ข้างต้นลุล่วงไปได้

2.1.5.2 เรคคอร์ดของ routing table

จะประกอบด้วยเรคคอร์ดหลายๆ เรคคอร์ดที่มีฟิลด์สำคัญต่างๆ เรคคอร์ดนี้จะคล้ายกับเรคคอร์ดในฐานข้อมูลแต่ในเชิงเน็ตเวิร์คจะเรียกเรคคอร์ดนี้ว่าเร้าติ้งเอ็นทรี (Routing entry) ภายในตารางเร้าติ้งเทเบิลจะประกอบด้วยเร้าติ้งเอ็นทรีอยู่หลายบรรทัดเรียงต่อกัน ซึ่งมีฟิลด์สำคัญต่างๆดังนี้

1) แหล่งที่มาของเร้าติ้งเอ็นทรีว่าเรียนรู้มาจากเร้าติ้งโปรโตคอลใดหรือมาจากการเพิ่มเร้าติ้งโดยผู้ดูแลระบบ

2) เป้าหมายปลายทางซึ่งเป็นได้ทั้งเน็ตเวิร์คแอดเดรสในคลาสหลักและในคลาสร้อย เช่น เน็ตเวิร์คแอดเดรส 10.0.0.0 ซึ่งเป็นเน็ตเวิร์คแบบเต็มคลาส A และ 10.10.1.0/24 ซึ่งเป็นซับเน็ตแอดเดรสที่ขอย่อยมาจากเน็ตเวิร์คคลาส A

3) Administrative Distance (AD) เป็นตัวเลขบอกถึงลำดับความสำคัญหรือความน่าเชื่อถือของเราตั้งเอ็นทรีนี้ (trustworthiness)

4) ค่าเมตริกหรือเรียกอีกอย่างว่าค่า Cost ของเส้นทางนี้ ซึ่งแต่ละเราต์โปรโตคอลจะมีหลักการกำหนดค่า Cost แตกต่างกันไป

5) แอดเดรสของเราท์เตอร์ตัวถัดไป (Next Hop เราท์เตอร์) เพื่อส่งแพ็คเกตไปยังซัพเนตแอดเดรสปลายทาง

6) อินเตอร์เฟซของเราท์เตอร์ที่ใช้เป็นทางออกไปยังเราท์เตอร์ตัวถัดไปหรือไปยังซัพเนตแอดเดรสปลายทาง เรียกว่า Outgoing Interface

2.1.6 DHCP(Dynamic Host Configuration Protocol)

2.1.6.1 หลักการเกี่ยวกับ DHCP

DHCP คือโปรโตคอลที่ใช้ในการจัดสรรการตั้งค่าการเข้าใช้งานเครือข่าย เช่น IP Address, Subnet Mask, Gateway, DNS รวมถึงพารามิเตอร์อื่นๆที่เกี่ยวข้องกับการตั้งค่าให้กับเครื่องไคลเอนต์(Client) แบบอัตโนมัติซึ่งโปรโตคอล DHCP นิยมใช้กับองค์กรหรือสำนักงานที่มีเครื่องไคลเอนต์เป็นจำนวนมากทำให้ผู้ดูแลระบบไม่ต้องเสียเวลาในการกำหนดค่าต่างๆให้กับเครื่องไคลเอนต์เอง โดยส่วนมากโปรโตคอล DHCP มีการใช้หมายเลขเครื่องในการพิสูจน์ตัวจริงแต่อย่างไรก็ตามหมายเลขเครื่องสามารถถูกปลอมแปลงได้ซึ่งจะเป็นช่องโหว่เพื่อให้ผู้บุกรุกสามารถลักลอบเข้ามาสร้างความเสียหายภายในระบบได้ โปรโตคอล DHCPทำงานแบบเซิร์ฟเวอร์-ไคลเอนต์ถูกพัฒนามาจากโปรโตคอลเก่า คือ BOOTP ที่มีมาในปี 1985 โดยนาย Gilmore ของบริษัท Sun Microsystem และทุนสนับสนุนของมหาวิทยาลัย Stanford ซึ่งส่วนมากอยู่ภายใต้มาตรฐาน RFC 2131 ถูกประกาศในเดือนมีนาคมปี ค.ศ. 1997 โปรโตคอลDHCP ยังถูกพัฒนาการทำงานอยู่เรื่อยๆ รุ่นล่าสุดของ DHCP คือ DHCPv6 ซึ่งใช้กับโปรโตคอลIPv6 ที่ได้รับการอนุมัติเมื่อเดือนกรกฎาคม ปี ค.ศ. 2003

2.1.6.2 ลักษณะการทำงานของโปรโตคอล DHCP

1) DHCP Discover Message เมื่อเครื่องไคลเอนต์ถูกเปิดขึ้นมา จะทำการค้นหาเครื่องเซิร์ฟเวอร์ โดยการบรอดคาสต์ข้อความ DHCP Discover ไปยังทุกๆ เครื่องในเครือข่ายและยังส่งหมายเลขเครื่องและชื่อของเครื่องไคลเอนต์ไปด้วย

2) DHCP Offer Message เมื่อเครื่องเซิร์ฟเวอร์ได้รับข้อความที่บรอดคาสต์มาจากเครื่องไคลเอนต์ เครื่องเซิร์ฟเวอร์จะแสดงตัวว่าเป็นผู้ที่สามารถให้บริการได้โดย

การส่งข้อความ DHCP Offer ไปยังไคลเอนต์ ซึ่งข้อความจากเครื่องเซิร์ฟเวอร์ได้ส่งไปยังเครื่องไคลเอนต์ก่อนเครื่องนั้นก็จะถูกเลือกใช้ (First-Come/First-Served)

3) DHCP Request Message เครื่องไคลเอนต์จะตอบรับ โดยการบรอดคาสต์ข้อความ DHCP Request กลับไปที่เซิร์ฟเวอร์

4) DHCP Acknowledge Message เซิร์ฟเวอร์จะส่ง IP Address, Subnet mark, DNS และค่าพารามิเตอร์ต่างๆ ไปยังไคลเอนต์โดยข้อความ DHCP Acknowledge

2.1.7 ระบบปฏิบัติการ Debian Linux

2.1.7.1 ระบบปฏิบัติการ Debian Linux

ระบบปฏิบัติการ Debian Linux เป็นชุดของซอฟต์แวร์เสรีที่พัฒนาโดยอาสาสมัครภายใต้โครงการเดเบียน ภายใต้โครงการนี้มีเดเบียนลินุกซ์ (Debian GNU/Linux) ที่ใช้ลินุกซ์เป็นเคอร์เนลและใช้เครื่องมือต่างๆ ในโครงการ GNU ประกอบกันเป็นระบบปฏิบัติการได้โดยเดเบียนมีชื่อเสียงในฐานะที่เป็นลินุกซ์ดิสทริบิวชันแรกๆ ที่ประสบความสำเร็จในการพัฒนาด้วยชุมชนล้วนๆ โดยไม่มีเอกชนอยู่เบื้องหลังมีการสร้างสัญญาประชาคมบทนิยามซอฟต์แวร์เสรีและแนวนโยบายที่ชัดเจนทั้งทางเทคนิคและการบริหารงานกลายเป็นต้นแบบให้กับชุมชนอื่นๆ ต่อมา รวมถึงปริมาณแพ็คเกจในโครงการมากกว่า 37,000 แพ็คเกจและรองรับสถาปัตยกรรมคอมพิวเตอร์มากกว่า 11 ชนิด ซึ่งครอบคลุมตั้งแต่คอมพิวเตอร์ฝังตัวไปจนถึงเมนเฟรมมีลินุกซ์ดิสทริบิวชันจำนวนมากที่นำเดเบียนไปพัฒนาต่อ อย่างเช่น อูบุนตุ หรือ Knoppix เป็นต้นเดเบียนเป็นดิสโทรที่พัฒนาสิ่งแปลกใหม่ให้เกิดขึ้นในวงการลินุกซ์เป็นอย่างมาก เช่น มีระบบจัดการแพ็คเกจ ระบบติดตั้งจากคลั่ง และยูทิลิตี้ที่สำคัญเป็นเทคโนโลยีของตนเองจนกลายเป็นรูปแบบการใช้งานที่เป็นอีกหนึ่งบรรทัดฐานของการใช้งานลินุกซ์เช่นเดียวกับที่กลุ่ม Red Hat ได้สร้างไว้

2.1.8 traffic control

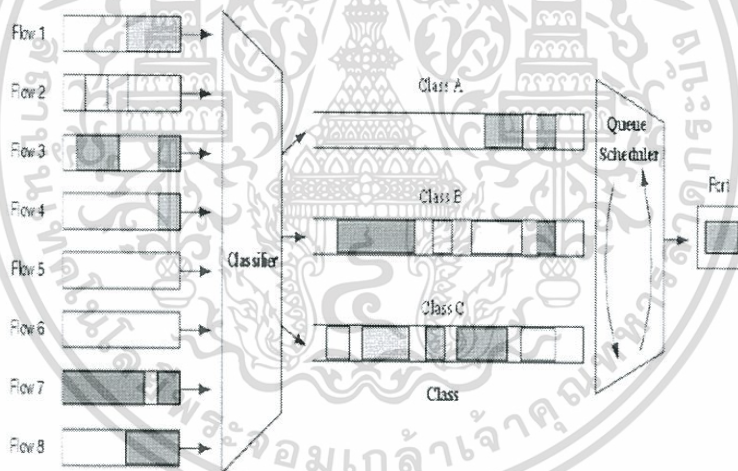
2.1.8.1 หลักการเกี่ยวกับ traffic control

Traffic control เป็นการควบคุมการใช้งานแบนด์วิดธ์ของระบบเครือข่ายให้มีประสิทธิภาพและเป็นไปตามที่ต้องการในการติดต่อสื่อสาร Traffic control จะทำการแบ่งชนิดการสื่อสารออกเป็นการจัดหมวดหมู่ (Classification) การจัดลำดับ (Queuing) และกำหนดขนาดของแบนด์วิดธ์เพื่อให้เกิดการใช้งานอย่างคุ้มค่าและมีประสิทธิภาพมากที่สุด (QoS) การทำงานของ Traffic control จะทำงานที่ปลายทางของเครือข่ายเพื่อทำการควบคุมการทำงานของสายสื่อสารทั้ง

ขาเข้าและขาออก Traffic control มีหลักการการทำงานเหมือนกับ ไฟล์วอลโดยจะเลือกแพ็คเก็ตที่จะทำการควบคุมเข้าไปในโครงข่ายเพื่อที่จะสามารถทำให้กำหนดอัตราการไหลของข้อมูลได้

2.1.8.2 Queuing discipline (qdisc)

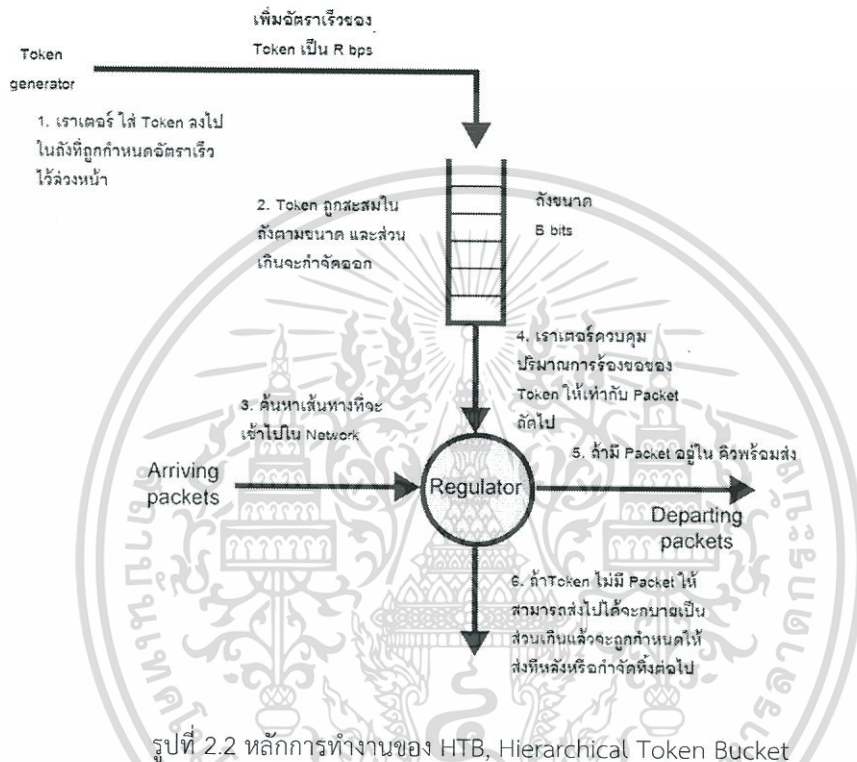
ทำหน้าที่ในการจัดการกับแพ็คเก็ตที่เข้าออกของระบบเครือข่ายตามที่ได้กำหนดไว้ด้วยวิธีการต่างๆ ชุดคำสั่งของ Queuing discipline ในระบบปฏิบัติการลินุกซ์ซึ่งมีอยู่ 2 ประเภทคือ Classless qdisc และ Classful qdisc โดยส่วนใหญ่ที่มีการใช้งานคือ Classful qdisc เพราะสามารถออกแบบคลาสของการทำงานได้หลายๆ คลาสตามที่ต้องการและในส่วนของ Classless qdisc จะมีคลาสการทำงานเพียงแค่คลาสเดียว การทำงานของ Classful qdisc ประกอบด้วย Classifier ทำหน้าที่ในการจัดการคัดแยกแพ็คเก็ตแล้วส่งเข้าสู่คลาสที่ได้สร้างไว้ Class เป็นคลาสการทำงานที่สร้างขึ้นโดยแต่ละคลาสสามารถกำหนดขนาดของแบนด์วิดท์ที่แตกต่างกันให้กับแต่ละคลาสได้และส่วนสุดท้ายคือ Queue Scheduler ทำหน้าที่ในการกำหนดคิวของแพ็คเก็ตที่จะออกสู่อินเทอร์เน็ต



รูปที่ 2.1 Queuing discipline (qdisc)

2.1.8.3 HTB, Hierarchical Token Bucket

ใช้แนวความคิดของ tokens/buckets และใช้ filter ในการควบคุมปริมาณ ข้อมูลจึงมีความสามารถในการจำกัดปริมาณแบนวิดท์และการมีรับประกันแบนวิดท์ด้วยโดยมีหลักการ ทำงานดังภาพดังนี้



รูปที่ 2.2 หลักการทำงานของ HTB, Hierarchical Token Bucket

2.8.1.4 Filter หรือ Classifier

ในส่วนนี้ทำหน้าที่จำแนกแพ็คเก็ตให้ส่งไปทำงานตาม class ต่างๆตามที่ qdisc กำหนดอยู่ชุดคำสั่งของ tc นั้น สามารถรองรับ Filterโดยส่วนใหญ่่นั้นได้ถูกติดตั้งมากับ iproute โดย Filterที่ใช้ใน QoS Generator คือ u32 classifierซึ่งจะทำการกรองแพ็คเก็ตจาก fields ของ IP header ของแต่ละแพ็คเก็ตเช่นหมายเลข port ต้นทาง (sport) , ปลายทาง(dport), และ ชนิดของบริการ (Type of service) เป็นต้น

2.8.1.5 Class

คำสั่ง Class จะทำงานตามคำสั่ง qdisc กำหนดว่ามีคลาสประเภทไหนมีจำนวนเท่าไร ซึ่งคำสั่งของclassนั้นประกอบไปด้วยกฎต่างๆที่ใช้ควบคุมแพ็คเก็ตที่ออกจากการ์ดแลน เช่นข้อมูลของแต่ละคลาสว่ามีการกำหนดค่าในการจำกัดแบนด์วิดธ์อย่างไร

2.1.9 หลักการของภาษา C++

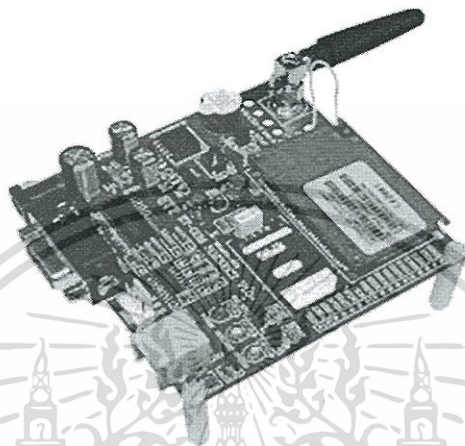
C++ คือภาษา C programming language รุ่นใหม่เป็นภาษาในการเขียนโปรแกรมถูกพัฒนาโดย Dr.Bjarne Stroustrup ซึ่งเป็นนักวิจัยอยู่ที่ห้องปฏิบัติการ Bell Labs ประเทศสหรัฐอเมริกาในระหว่างปีพ.ศ. 2525-2528 ภาษาC++เกิดจากแนวคิดในการเพิ่มประสิทธิภาพภาษาC โดยได้นำความสามารถของภาษา C มาพัฒนาให้เป็นโปรแกรมภาษาที่มีความเป็น Object Oriented Programming (โปรแกรมเชิงวัตถุ) และนี่เองคือที่มาของภาษา C++ จากการพัฒนาทำให้ทุกสิ่งทีภาษา C ทำได้ภาษาC++ ก็จะสามารถทำได้

ภาษา C++ ถูกออกแบบมาสำหรับการทำงานภายใต้สิ่งแวดล้อมระบบปฏิบัติการUNIXด้วยภาษา C++ ผู้เขียนโปรแกรมสามารถเขียนโปรแกรมได้อย่างมีประสิทธิภาพมากขึ้นนอกจากนี้การเขียนโปรแกรมเพื่อให้สามารถนำกลับมาใช้ได้ใหม่ (reusability) ก็ยังสามารถทำได้ง่ายขึ้น

ข้อดีของภาษา C++

- 1) ภาษา C++ จะมีการทำงานที่ค่อนข้างเร็วมากเมื่อเทียบกับภาษาอื่นและยังสามารถดำเนินการกับ Hardware ได้โดยที่โปรแกรมภาษาบางโปรแกรมอาจจะไม่สนับสนุนคุณลักษณะนี้
- 2) ภาษา C++ มีความเป็น Object Oriented Programming และยังเป็น Structure Programming ซึ่งเหมาะที่จะใช้ศึกษาเกี่ยวกับการเขียนโปรแกรมสำหรับผู้เริ่มต้นและนอกจากนั้นถ้าหากเราจะเรียนเรื่อง Data Structure หรือทางด้านอัลกอริทึมในต่างประเทศจะนิยมใช้ C++ ในการสอนรวมถึงการเรียนรู้อิงระบบการทำงานของระบบปฏิบัติการ ตำราส่วนใหญ่ก็จะใช้ C++ ในการสอนซึ่งถ้าเราสามารถอ่าน Source code C++ รู้เรื่องก็จะทำให้เราเรียนรู้เกี่ยวกับการเป็นโปรแกรมเมอร์ได้ง่ายขึ้น

2.1.10 ชุดพัฒนา GSM/GPRS รุ่น ET-GSM SIM900B



รูปที่ 2.3 ET-GSM SIM900B module

ET-GSM SIM900B เป็นชุดเรียนรู้และพัฒนาการระบบการสื่อสารไร้สายโดยใช้โมดูล GSM/GPRS รุ่น SIM900B ของ “SIMCom Ltd.” เป็นอุปกรณ์หลัก ซึ่ง SIM900B เป็นโมดูลสื่อสารระบบ GSM/GPRS ขนาดเล็กรองรับระบบสื่อสาร GSM ความถี่ 850/900/1800/1900 MHz โดยสั่งงานผ่านทางพอร์ตสื่อสารอนุกรม RS232 ด้วยชุดคำสั่ง AT Command สามารถประยุกต์ใช้งานได้มากมายหลายรูปแบบ ไม่ว่าจะเป็นการรับส่งสัญญาณแบบ Voice, SMS, Data, FAX และยังรวมถึงการสื่อสารด้วย Protocol TCP/IP อีกด้วย

ซึ่งตามปกติแล้วถึงแม้ว่าโมดูล SIM900B จะมีวงจรและ Firmware บรรจุไว้ภายในตัวเป็นที่เรียบร้อยแล้ว แต่ก็ยังไม่สามารถนำไปใช้งานได้โดยตรงทันทีเนื่องจากการใช้งานจริงๆ นั้น ผู้ใช้งานเองจำเป็นต้องออกแบบวงจรรอบนอกที่จำเป็นมาเชื่อมต่อกับขาสัญญาณของตัวโมดูลอีกในบางส่วน ไม่ว่าจะเป็นวงจรภาค Power Supply, วงจรเชื่อมต่อกับ SIM Card รวมไปถึงวงจร Line

Driver ของ RS232 เป็นต้น และยังมีบอร์ดสำหรับเป็นตัวกลางในการเชื่อมต่อระหว่างโมดูล SIM900B กับอุปกรณ์ภายนอกเพื่อให้ผู้ใช้งานสามารถนำโมดูล GSM ของ SIM900B

2.2 การแสดงผลผ่านหน้าเว็บ

2.2.1 ความเป็นมาและโครงสร้างของภาษา HTML

ในปี ค.ศ. 1990 Tim Berners-Lee นักวิทยาศาสตร์ชาวอังกฤษได้นำเสนอการจัดรูปแบบเอกสารที่เรียกว่า " HTML Tag " โดยใช้พื้นฐานมาจากภาษา SGML (Standard Generalized Markup Language) ซึ่งเป็นภาษาในการจัดโครงสร้างของเอกสารที่มีอยู่แล้วในขณะนั้นทั้งรูปแบบภาษา HTML ที่ Tim Berners-Lee นำเสนอนี้ได้ถูกนำไปใช้เป็นข้อกำหนดในการสร้างเว็บไซต์ตัวแรกคือ NCSA Mosaic ในเวลาต่อมา Tim Berners-Lee ได้จัดตั้งหน่วยงานแรกชื่อ World Wide Web Consortium (W3C) ขึ้นที่สถาบัน MIT เพื่อเข้ามาควบคุมดูแลและกำหนดมาตรฐานต่างๆของภาษา HTML โดยเฉพาะ แต่อย่างไรก็ตามเนื่องจากหลักการต่างๆของภาษา HTML ที่สร้างขึ้นมามีความยังไม่ชัดเจนและสมบูรณ์มากนัก ดังนั้นจึงมีการปรับปรุงแก้ไขกันเรื่อยมาจนถึงในปี 1994 ได้พัฒนาเป็น HTML 2.0 ซึ่งถือเป็นการเปลี่ยนแปลงครั้งสำคัญที่ทำให้ ภาษา HTML ถูกนำไปใช้กันอย่างแพร่หลายเนื่องจากมีการเพิ่มเติมลักษณะใหม่ที่น่าสนใจหลายอย่าง เช่น การแทรกรูปภาพทำฟอร์มในการรับข้อมูล, การใช้ตารางในการจัดรูปแบบ หลังจากนั้นภาษา HTML ก็ได้รับการพัฒนาโดยลำดับเป็นเวอร์ชัน 3.0, 3.2, 4.0, 4.01 และ XHTML

ในปี ค.ศ. 2004 กลุ่มบริษัทผู้สร้างเว็บเบราว์เซอร์ (Web Browse) ได้แก่ Apple, Mozilla และ Opera ได้รวมตัวกันก่อตั้งหน่วยงานที่มีชื่อว่า Web Hypertext Application Technology Working Group (WHATWG) เพื่อสร้างมาตรฐานใหม่และขยายขีดความสามารถของภาษา HTML ให้มากขึ้นกว่าเดิม ในขณะที่หน่วยงาน W3C นั้นเดิมมีโครงการที่จะพัฒนา XHTML 2.0 แต่ต่อมาได้ยกเลิกโครงการดังกล่าวแล้วหันมาสนับสนุนกลุ่ม WHATWG แทน จนกระทั่งปี ค.ศ. 2007 จึงได้มีการเผยแพร่ข้อกำหนดใหม่ของภาษา HTML เป็นครั้งแรก ซึ่งถือว่าการเปลี่ยนแปลงที่สำคัญอีกครั้งหนึ่งของ HTML โดยในเวอร์ชันนี้กำหนดให้มีชื่อว่า HTML5



รูปที่ 2.4 สัญลักษณ์ของ HTML 5

2.2.1.1 โครงสร้าง ภาษา HTML

- 1) ส่วนประกาศ เป็นส่วนที่กำหนดให้เบราว์เซอร์ทราบว่านี่คือภาษา HTML และจะต้องทำการแปรผลอย่างไร มีคำสั่งคู่เดียวคือ `<html>` และ `</html>` ปรากฏที่หัวและท้ายไฟล์
- 2) ส่วนหัวเรื่อง (head) เป็นส่วนที่แสดงผลข้อความบนไตเติ้ลบาร์ของเบราว์เซอร์และอาจมีคำสั่งสำหรับกำหนดรายละเอียดด้านเทคนิคอื่นๆอีก แทรกอยู่ระหว่างคำสั่ง `<head>` และ `</head>`
- 3) ส่วนเนื้อหา (body) เป็นส่วนที่มีความซับซ้อนมากที่สุดและสามารถใส่เทคนิคลูกเล่นเพื่อดึงดูดความสนใจจากผู้ชมได้มาก ความแตกต่างระหว่างเว็บไซต์ต่างๆแสดงควมมีฝีมือของผู้จัดทำศิลปะในการออกแบบจะอยู่ในส่วนนี้ทั้งหมดซึ่งจะแทรกอยู่ระหว่างคำสั่ง `<body>` และ `</body>`

2.2.1.2 Tag ต่างๆใน ภาษา HTML

โครงสร้างพื้นฐานของภาษา Computer เป็นส่วนที่สำคัญที่สุดของการเขียนภาษา Computer โดยทั่วไปแล้วมันจะต้องถูกเขียนขึ้นทุกครั้ง ภาษา HTML ก็เหมือนกับภาษา Computer ทั่วไปที่ไม่มีโครงสร้างพื้นฐานเฉพาะของมัน คำสั่งของ HTML ส่วนมากจะถูกกำหนดให้อยู่ในเครื่องหมาย `<` และ `>` ซึ่งถูกเรียกว่า Tag สำหรับในส่วนของคำสั่ง Tag ภายใน คำสั่งซึ่งสำหรับในส่วนของคำสั่ง Tag ภายในคำสั่ง โครงสร้างพื้นฐานพอที่จะอธิบายคร่าวๆได้ดังนี้

1) Title(ชื่อหัวเรื่อง)

จะถูกกำหนด อยู่ใน Tag คำสั่ง

<HEAD>

<TITLE> ชื่อหัวเรื่อง </TITLE>

</HEAD>

2) ข้อมูลที่ต้องการแสดงผล

จะเป็นส่วนที่แสดงให้เราเห็นไม่ว่าจะเป็น ตัวอักษร,รูปภาพ,ตาราง ฯลฯ (คำสั่งที่ต้องการแสดงผลจะอยู่ระหว่าง tag BODY ทั้งหมด) ซึ่งถูกกำหนดอยู่ระหว่างคำสั่ง <BODY> จนถึงคำสั่ง </BODY>

3) คำสั่งComment Tag เป็นคำสั่งที่ใช้ในการอธิบาย ซึ่งอยู่ใน HTML จะไม่มีการแสดงผลออกมาที่ Browser จะมีประโยชน์สำหรับผู้ที่ทำการแก้ไขโปรแกรมในภายหลัง <!--ใส่ข้อความใดๆก็ได้ เพื่อใช้ในการ อธิบาย-->

4) คำสั่งขึ้นบรรทัดใหม่

 เป็นคำสั่งที่ใช้กำหนดให้ข้อความที่เราพิมพ์ลงไปเอกสารขึ้นบรรทัดใหม่ได้ตามที่เราต้องการเพราะถ้าเราไม่ใช้คำสั่งสั่งให้เอกสารแสดงผลขึ้นบรรทัดใหม่ การแสดงผลของข้อความจะแสดงต่อกันไปเรื่อยๆแม้ว่าเราจะพิมพ์ข้อความขึ้นบรรทัดใหม่ก็ตาม

5) คำสั่งการย่อหน้าใหม่ รูปแบบคำสั่ง

<P>..... </P> หรือ <P> มีลักษณะคล้ายกับคำสั่ง < BR > แต่คำสั่งนี้ จะมีการเว้นบรรทัดว่างให้หนึ่งบรรทัดเพราะบางครั้งเราต้องการเว้นบรรทัดว่างหนึ่งบรรทัดแต่โปรแกรม Web Browser จะไม่เข้าใจการพิมพ์บรรทัดเปล่า

6) เส้นคั่นบรรทัด

เป็นคำสั่งที่ใช้แบ่งข้อความของจอภาพให้เป็นส่วนรูปแบบคำสั่ง <HR>

2.2.2 หลักการทำงานของ Bootstrap

Bootstrap คือ CSS Framework หรือ Front-end Framework ตัวหนึ่ง คำว่า front-end หมายถึงส่วนที่แสดงผลให้ Users ทั่วไปเห็น ส่วนframeworkนั้นจะหมายถึงสิ่งที่เข้ามาช่วยกำหนดกรอบของการทำงานให้เป็นไปในทางเดียวกัน ช่วยให้การพัฒนาเว็บไซต์ของเราเร็วขึ้นง่ายขึ้นและเป็นระบบมากขึ้น เปรียบเสมือนกับโปรแกรมเสริม(Plug-in)หรือโปรแกรมส่วนเสริม(Extension)ทำให้ไม่ต้องเสียเวลาในการเขียนโค้ดคำสั่งที่มีอยู่แล้ว โดยผู้ใช้งานสามารถเรียกติดตั้งและใช้งาน Bootstrap ที่ต้องการในเว็บไซต์ของผู้ใช้งานเองได้

Bootstrap คือโปรแกรมฟรีสำหรับพัฒนาเว็บไซต์แบบง่ายๆจากค่าย Twitter ซึ่งเครื่องมือนี้ช่วยให้เราไม่ต้องไปเขียนโค้ดเองทั้งหมด เพียงนำไปปรับใช้กับงานตนเอง คุณสมบัติที่นำมา CSS, JavaScript plugin โดยที่ CSS ที่ให้มาเป็น CSS3 ซึ่งมี Grids มาพร้อมด้วยช่วยให้การจัดวาง Layout ได้ง่ายขึ้น

2.2.2.1 ข้อดีของ Bootstrap

- 1) สามารถทำงานได้ดีแม้จะทำงานร่วมกับ Web browser รุ่นเก่าที่ค่อนข้างช้า นอกจากนี้ยังสามารถใช้งานได้ผ่านทางแท็บเล็ตและสมาร์ตโฟน
- 2) มีตัวช่วยที่ทำให้เราสามารถออกแบบเว็บไซต์ได้ง่าย และสวยงาม เช่น Grid ที่ช่วยในการจัด Layout ให้กับเว็บไซต์ นอกจากนี้แล้วเส้น Grid ยังช่วยให้การแบ่งแถบบนหน้าเว็บไซต์ได้ง่ายและมีความอิสระมากขึ้น



รูปที่ 2.5 ตัวอย่างของเว็บไซต์ที่พัฒนาด้วย Bootstrap

2.2.3 หลักการของ PHP

PHP Hypertext Preprocessor หรือชื่อเดิม Personal Home Page การแสดงผลของPHPจะปรากฏในลักษณะ HTML ซึ่งจะไม่ต้องแสดงคำสั่งที่ผู้ใช้เขียนเป็นลักษณะเด่นที่PHPแตกต่างจากภาษาในลักษณะ โคเลอเนต-ไซด์ สคริปต์ เช่น ภาษาจาวาสคริปต์ที่ผู้ชมเว็บไซต์สามารถอ่าน ดู และคัดลอก คำสั่งไปใช้เองได้ นอกจากนี้PHPยังเป็นภาษาที่เรียนรู้และเริ่มต้นได้ไม่ยากโดยมีเครื่องมือที่ช่วยเหลือและคู่มือที่สามารถหาอ่านได้ฟรีบนอินเทอร์เน็ต ความสามารถประมวลผลหลักของ PHP ได้แก่ การสร้างเนื้อหาอัตโนมัติจัดการคำสั่ง การอ่านข้อมูลจากผู้ใช้และประมวลผล การอ่านข้อมูลจากดาต้าเบส ความสามารถจัดการกับคุกกี้ ซึ่งทำงานเช่นเดียวกับโปรแกรมในลักษณะ CGI คุณสมบัติอื่น เช่น การประมวลผลตามบรรทัดคำสั่ง (command line scripting) ที่ทำให้ผู้เขียนโปรแกรมสร้างสคริปต์PHPซึ่งสามารถทำงานผ่านPHP parser โดยไม่ต้องผ่านเซิร์ฟเวอร์หรือเบราว์เซอร์ ซึ่งมีลักษณะเหมือนกับ Cron(ในยูนิกซ์หรือลินุกซ์) หรือ Task Scheduler (ในวินโดวส์) สคริปต์เหล่านี้สามารถนำไปใช้ในแบบ Simple text processing tasks ได้

การแสดงผลของPHPถึงแม้ว่าจุดประสงค์หลักใช้ในการแสดงผล HTML แต่ยังสามารถสร้าง XHTML หรือ XMLได้นอกจากนี้สามารถทำงานร่วมกับคำสั่งเสริมต่างๆซึ่งสามารถแสดงผลข้อมูลหลัก PDF แพลช(โดยใช้ libswf และ Ming) PHPมีความสามารถอย่างมากในการทำงานเป็นตัวประมวลผลข้อความจากPOSIX ExtendedหรือรูปแบบPerlทั่วไปเพื่อแปลงเป็นเอกสาร XML ในการแปลงและเข้าสู่เอกสาร XML เรารองรับมาตรฐาน SAXและDOMสามารถใช้รูปแบบXSLTของซึ่งเราสามารถทำเพื่อเปลี่ยนแปลงเอกสาร XML

2.2.3.1 ลักษณะความโดดเด่นของภาษาPHP

1) ความรวดเร็วในการพัฒนาโปรแกรม เพราะว่าPHP เป็นสคริปต์แบบEmbedded คือสามารถแทรกพร้อมกับ HTML Tag ได้อย่างอิสระและหากเราพัฒนาโค้ดไว้ในรูปแบบของ Class ที่เขียนขึ้นเพียงครั้งเดียวแล้วเรียกใช้งานได้ตลอดทำให้สะดวกรวดเร็วต่อการพัฒนาโปรแกรม

2) PHPเป็นโค้ดแบบเปิดเผย (Open Source)เนื่องจากมีผู้ใช้งานอยู่เป็นจำนวนมากทั่วโลกและมีเว็บไซต์อยู่เป็นจำนวนมากที่เป็นแหล่งรวบรวมซอสโค้ดโปรแกรมหรือจะเป็นบทความต่างๆทำให้ผู้ใช้มือใหม่ ๆ หรือผู้ที่ต้องการศึกษาสามารถค้นหาซอสโค้ดมาเป็นแนวทางในการพัฒนาโปรแกรมได้ง่ายขึ้น

3) การบริการหน่วยความจำ (Memory Usage)มีการใช้หน่วยความจำที่ดีขึ้น คือPHP4จะไม่เรียกใช้หน่วยความจำตลอดเวลาการทำงาน

2.2.3.2 หลักการทำงานของPHP

- 1) ฝั่งไคลเอนต์ (Client) จะทำการร้องขอหรือเรียกใช้งานไฟล์ที่เก็บในเครื่องเซิร์ฟเวอร์(Server)
- 2) ฝั่งเซิร์ฟเวอร์จะทำการค้นหาไฟล์ PHPแล้วทำการประมวลผลไฟล์PHP ตามที่ไคลเอนต์ทำการร้องขอมา
- 3) เป็นการติดต่อกับฐานข้อมูลและนำข้อมูลในฐานข้อมูลมาใช้ร่วมกับการประมวลผลส่งผลลัพธ์จากการประมวลผลไปให้เครื่องไคลเอนต์

2.2.3.3 แนวโน้มของการพัฒนาในอนาคตของภาษา PHP

PHP ได้รับการพัฒนาความสามารถขึ้นมาเรื่อยๆอย่างต่อเนื่อง ทั้งนี้เป็นเพราะการเปิดเผยซอร์สโค้ดของPHPสู่สาธารณะในลักษณะของopen source โปรแกรมที่มีการ Open sourceจะมีการพัฒนาได้อย่างรวดเร็วเนื่องจากทำให้มีหน่วยงานและองค์กรต่างๆ หรือบุคคลที่สนใจได้เข้ามาคัดลอก source program ไปแก้ไขและพัฒนาจากต้นแบบหรือต้นฉบับเดิมได้อย่างกว้างขวางจึงทำให้เกิดการพัฒนาตัวโปรแกรมได้อย่างที่รวดเร็วเพราะไม่ต้องเสียเวลาในการเขียนโปรแกรมใหม่ตั้งแต่การเริ่มต้นแต่เป็นเพียงการพัฒนาเพื่อให้ โปรแกรมมีประสิทธิภาพมากและเหมาะสมกับการใช้งานในแต่ละงานให้มากที่สุด

2.2.3.4 การรองรับPHP

คำสั่งของPHPสามารถสร้างผ่านทางโปรแกรมแก้ไขข้อความทั่วไป เช่น โน้ตแพด หรือ vi ซึ่งทำให้การทำงานPHPสามารถทำงานได้ในระบบปฏิบัติการหลักเกือบทั้งหมดโดยเมื่อเขียน คำสั่ง แล้วนำมาประมวลผล Apache ,Microsoft Internet Information Server(IIS),Personal Web Server, Netscape และ iPlanet servers,Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd, และอื่นๆอีกมากมาย สำหรับส่วนหลักของ PHP ยังมี Module ในการรองรับCGIมาตรฐาน ซึ่งPHPสามารถทำงานเป็นตัวประมวลผล CGIด้วยPHP เราจะมีอิสรภาพในการเลือกระบบปฏิบัติการและเว็บเซิร์ฟเวอร์นอกจากนี้เรายังสามารถใช้สร้างโปรแกรมโครงสร้าง สร้างโปรแกรมเชิงวัตถุ (OOP) หรือสร้างโปรแกรมที่รวมทั้งสองอย่างเข้าด้วยกัน แม้ว่าความสามารถของคำสั่ง OOP มาตรฐานในเวอร์ชันนี้ยังไม่สมบูรณ์แต่ตัวไลบรารีทั้งหลายของโปรแกรมและตัวโปรแกรมประยุกต์ (รวมถึง PEAR library) ได้ถูกเขียนขึ้นโดยใช้รูปแบบการเขียนแบบ OOPเท่านั้น PHP สามารถทำงานร่วมกับฐานข้อมูลได้หลายชนิดซึ่งฐานข้อมูลที่รองรับ ได้แก่ Oracle dBase PostgreSQL IBM DB2 MySQL Informix ODBC โครงสร้างของฐานข้อมูลแบบ DBX ซึ่งทำให้PHPใช้กับฐานข้อมูลอะไรก็ได้PHPสามารถรองรับการสื่อสารกับการบริการในโปรโตคอลต่างๆ เช่น LDAP IMAP SNMP NNTP POP3 HTTP COM (บนวินโดวส์) และอื่นๆ อีกมากมายคุณสามารถเปิด Socket บนเครือข่ายโดยตรงและตอบโต้โดยใช้โปรโตคอลใดๆ ก็ได้ PHP

มีการรองรับสำหรับการแลกเปลี่ยนข้อมูลแบบ WDDX Complex กับ Web Programming อื่นๆ ทั่วไปได้พูดถึงในส่วน Interconnection, PHP มีการรองรับสำหรับ Java objects ให้เปลี่ยนมันเป็น PHP Object แล้วใช้งาน คุณยังสามารถใช้รูปแบบ CORBA เพื่อเข้าสู่ Remote Object ได้เช่นกัน คำสั่งของ HTML จะอยู่ระหว่างเครื่องหมายน้อยกว่า (<) และเครื่องหมายมากกว่า (>) ซึ่งเราเรียกว่า HTML tag ส่วนคำสั่งของ PHP นั้นก็จะอยู่ใน PHP tag PHP tag สามารถที่ใช้ในรูปแบบไหนก็ได้แต่การเขียนแบบ Short style นั้นจะเป็นที่นิยมในหมู่นักพัฒนา PHP มากกว่ารูปแบบอื่นๆ ทั้งนี้การที่จะใช้รูปแบบ Short style ได้นั้น ซึ่งจะต้องตั้งค่าให้ PHP ยอมรับรูปแบบนี้โดยการตั้งค่าใน php.ini หรือในระหว่างการ compile PHP ให้ enable short tag ด้วยการและการเขียนในรูปแบบ ASP style ก็ต้องมีการตั้งค่าให้สามารถใช้ในรูปแบบนี้ได้เช่นกัน การนำเอา PHP tag ไปแทรกใน HTML tag นั้น สามารถเอาไปแทรกที่ตำแหน่งไหนก็ได้ตั้งแต่ต้นไฟล์ไปจนถึงท้ายไฟล์แต่ในบางที่อาจจะกลับกัน ถ้าส่วนใหญ่ที่นั้นเป็น PHP ก็จะกลายเป็นว่า HTML แทรกอยู่ใน PHP

จะเห็นว่าเว็บเพจที่ใช้ PHP จะลงท้ายด้วย .php .php3 หรือ .html เช่น index.php เป็นต้นทั้งนี้ก็เพื่อเป็นการบอกให้ Web server รู้ว่าไฟล์ที่จะต้องส่งให้นี้เป็น PHP ซึ่งมันจะต้องทำการประมวลผลตามคำสั่งของ PHP เสียก่อนแล้วจึงค่อยส่งไฟล์ที่ได้หลังจากการประมวลผลสาเหตุที่ Web server รู้ว่าไฟล์ที่ลงท้ายด้วย .php นี้ เป็น PHP ก็เพราะคำสั่ง AddType application/x-httpd-php .php ที่เพิ่มเข้าไปในไฟล์ httpd.conf ในวิธีการติดตั้ง PHP

2.2.4 หลักการของ MySQL

เป็นฐานข้อมูลแบบ open source ที่ได้รับความนิยมในการใช้งานสูงสุดโปรแกรมหนึ่งบนเครื่องให้บริการมีความสามารถในการจัดการกับฐานข้อมูลด้วยภาษา SQL (Structures Query Language) อย่างมีประสิทธิภาพมีความรวดเร็วในการทำงานรองรับการทำงานจากผู้ใช้หลายๆ คนและหลายๆ งานได้

MySQL ถูกพัฒนาขึ้นโดย MySQL AB โดยมีลิขสิทธิ์การใช้งาน 2 แบบ นั่นคือ ผู้ดูแลระบบสามารถใช้งานซอฟต์แวร์ MySQL ได้โดยไม่มีค่าใช้จ่ายใด ๆ ภายใต้ลิขสิทธิ์ของ GNU General Public License (<http://www.gnu.org/licenses/>) หรืออาจเลือกใช้แบบที่มีลิขสิทธิ์ทางการค้าของ MySQL AB ซึ่งเป็นผู้ผลิตและพัฒนาซอฟต์แวร์โดยตรงก็ได้ หากไม่ต้องการเกี่ยวข้องกับข้อตกลงเรื่อง GPL รายละเอียดเพิ่มเติมเกี่ยวกับโปรแกรม MySQL สามารถหาข้อมูลเพิ่มเติมได้จาก <http://www.mysql.com> คำอธิบายเพิ่มเติมเกี่ยวกับหน้าที่ และความสามารถ และการทำงานของโปรแกรม MySQL มีดังต่อไปนี้

1) MySQL ถือเป็นระบบจัดการฐานข้อมูล (DataBase Management System : DBMS) ฐานข้อมูลมีลักษณะเป็นโครงสร้างของการเก็บรวบรวมข้อมูล การที่จะเพิ่มเติมเข้าถึงหรือประมวลผลข้อมูลที่เกิดขึ้นในฐานข้อมูลจำเป็นจะต้องอาศัยระบบจัดการฐานข้อมูลซึ่งจะทำหน้าที่เป็นตัวกลางในการจัดการกับข้อมูลในฐานข้อมูลทั้งสำหรับการใช้งานเฉพาะและรองรับการทำงานของแอปพลิเคชันอื่นๆ ที่ต้องการจะใช้งานข้อมูลในฐานข้อมูลเพื่อให้ได้รับความสะดวกในการจัดการกับข้อมูลจำนวนมาก MySQL ทำหน้าที่เป็นทั้งตัวฐานข้อมูลและระบบจัดการฐานข้อมูล

2) MySQL เป็นระบบจัดการฐานข้อมูลแบบrelationalฐานข้อมูลแบบ relational จะทำการเก็บข้อมูลทั้งหมดในรูปแบบของตารางแทนการเก็บข้อมูลทั้งหมดลงในไฟล์เพียงไฟล์เดียวทำให้ทำงานได้รวดเร็วและมีความยืดหยุ่นนอกจากนั้นแต่ละตารางที่เก็บข้อมูลสามารถเชื่อมโยงเข้าหากันทำให้สามารถรวมหรือจัดกลุ่มข้อมูลได้ตามต้องการโดยอาศัยภาษา SQL ที่เป็นส่วนหนึ่งของโปรแกรม MySQL ซึ่งเป็นภาษามาตรฐานในการเข้าถึงฐานข้อมูล

3) MySQL แจกจ่ายให้ใช้งานแบบ open source นั่นคือผู้ใช้งาน MySQL ทุกคนสามารถใช้งานและปรับแต่งการทำงานได้ตามต้องการสามารถดาวน์โหลดโปรแกรม MySQL ได้จากอินเทอร์เน็ตและนำมาใช้งานโดยไม่มีค่าใช้จ่ายใดๆ

2.2.5 หลักการของ CSS

CSS ย่อมาจาก Cascading Style Sheet มักเรียกโดยย่อว่า "สไตลชีต" คือภาษาที่ใช้เป็นส่วนของการจัดรูปแบบการแสดงผลเอกสาร HTML โดยที่ CSS กำหนดกฎเกณฑ์ในการระบุรูปแบบ (หรือ "Style") ของเนื้อหาในเอกสาร อันได้แก่ สีของข้อความ สีพื้นหลัง ประเภทตัวอักษร และการจัดวางข้อความ ซึ่งการกำหนดรูปแบบ หรือ Style นี้ใช้หลักการของการแยกเนื้อหาเอกสาร HTML ออกจากคำสั่งที่ใช้ในการจัดรูปแบบการแสดงผลกำหนดให้รูปแบบของการแสดงผลเอกสารไม่ขึ้นอยู่กับเนื้อหาของเอกสารเพื่อให้ง่ายต่อการจัดรูปแบบการแสดงผลลิ้งค์ของเอกสาร HTML โดยเฉพาะในกรณีที่มีการเปลี่ยนแปลงเนื้อหาเอกสารบ่อยครั้งหรือต้องการควบคุมให้รูปแบบการแสดงผลเอกสาร HTML มีลักษณะของความสม่ำเสมอทั่วกันทุกหน้าเอกสารภายในเว็บไซต์เดียวกัน โดยกฎเกณฑ์ในการกำหนดรูปแบบ (Style) เอกสาร HTML ถูกเพิ่มเข้ามาครั้งแรกใน HTML 4.0 เมื่อปีพ.ศ. 2539 ในรูปแบบของ CSS level 1 Recommendations ที่กำหนดโดย องค์กร World Wide Web Consortium หรือ W3C

ประโยชน์ของ CSS

1) CSS มีคุณสมบัติมากกว่า tag ของ html เช่น การกำหนดกรอบให้ข้อความรวมทั้งสี และรูปแบบของข้อความที่ได้กล่าวมาก่อนหน้านี้แล้ว

2) CSS นั้นกำหนดที่ต้นของไฟล์ html หรือตำแหน่งอื่น ๆ ก็ได้ และสามารถมีผลกับเอกสารทั้งหมดหมายถึงกำหนด ครั้งเดียวจุดเดียวก็มีผลกับการแสดงผลทั้งหมดทำให้เวลาแก้ไขหรือปรับปรุงทำได้สะดวกไม่ต้องไล่ตามแก้ tag ต่างๆ ทั้งทั้งเอกสาร

3) CSS สามารถกำหนดแยกไว้ต่างหากจาก ไฟล์เอกสาร html และสามารถนำมาใช้ร่วมกับเอกสารหลายไฟล์ได้ การแก้ไขก็แก้เพียงจุดเดียวก็มีผลกับเอกสารทั้งหมด

2.2.6 หลักการของ Javascript

JavaScript คือภาษาคอมพิวเตอร์สำหรับการเขียนโปรแกรมบนระบบอินเทอร์เน็ตที่กำลังได้รับความนิยมอย่างสูง Java JavaScript เป็นภาษาสคริปต์เชิงวัตถุ (ที่เรียกกันว่า "สคริปต์" (script) ซึ่งในการสร้างและพัฒนาเว็บไซต์ (ใช้ร่วมกับ HTML) เพื่อให้เว็บไซต์ของเราดูมีการเคลื่อนไหวสามารถตอบสนองผู้ใช้งานได้มากขึ้น ซึ่งมีวิธีการทำงานในลักษณะ "แปลความและดำเนินงานไปที่ละคำสั่ง" (interpret) หรือเรียกว่า อ็อบเจ็กต์โอเรียนเตด (Object Oriented Programming) ที่มีเป้าหมายในการออกแบบและพัฒนาโปรแกรมในระบบอินเทอร์เน็ตสำหรับผู้เขียนด้วยภาษา HTML สามารถทำงานข้ามแพลตฟอร์มได้โดยทำงานร่วมกับ ภาษา HTML และภาษา Java ได้ทั้งทางฝั่งไคลเอนต์ และทางฝั่งเซิร์ฟเวอร์

JavaScript ถูกพัฒนาขึ้นโดยเน็ตสเคปคอมมิวนิเคชันส์ (Netscape Communications Corporation) โดยใช้ชื่อว่า Live Script ออกมาพร้อมกับ Netscape Navigator 2.0 เพื่อใช้สร้างเว็บเพจโดยติดต่อกับเซิร์ฟเวอร์แบบ Live Wire ต่อมาเน็ตสเคปจึงได้ร่วมมือกับบริษัทซันไมโครซิสเต็มส์ปรับปรุงระบบของเบราวเซอร์เพื่อให้สามารถติดต่อกับภาษาจาวาได้และได้ปรับปรุง LiveScript ใหม่เมื่อ ปี 2538 แล้วตั้งชื่อใหม่ว่า JavaScript JavaScript สามารถทำให้การสร้างเว็บเพจมีลูกเล่นต่าง ๆ มากมายและยังสามารถโต้ตอบกับผู้ใช้ได้อย่างทันที เช่น การใช้เมาส์คลิก หรือ การกรอกข้อความในฟอร์ม เป็นต้น

เนื่องจาก JavaScript ช่วยให้ผู้พัฒนาสามารถสร้างเว็บเพจได้ตรงกับความต้องการและมีความน่าสนใจมากขึ้น ประกอบกับเป็นภาษาเปิดที่ใครก็สามารถนำไปใช้ได้ดังนั้นจึงได้รับความนิยมเป็นอย่างสูงมีการใช้งานอย่างกว้างขวางรวมทั้งได้ถูกกำหนดให้เป็นมาตรฐานโดย ECMA การทำงานของ JavaScript จะต้องมีการแปลความคำสั่งซึ่งขั้นตอนนี้จะถูกจัดการโดยเบราวเซอร์ (เรียกว่าเป็น client-side script) ดังนั้น JavaScript จึงสามารถทำงานได้เฉพาะบนเบราวเซอร์ที่สนับสนุนซึ่งปัจจุบันเบราวเซอร์เกือบทั้งหมดก็สนับสนุน JavaScript แล้ว

บทที่ 3

การออกแบบและการจัดทำปฏิญญาพันธัน์

3.1 การออกแบบ

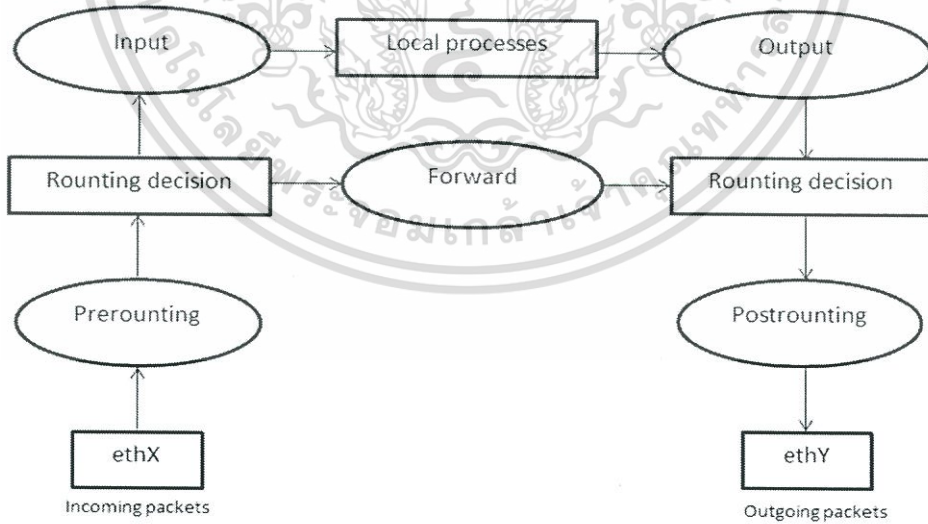
3.1.1 ภาพรวมระบบ



รูปที่ 3.1 ภาพรวมระบบ

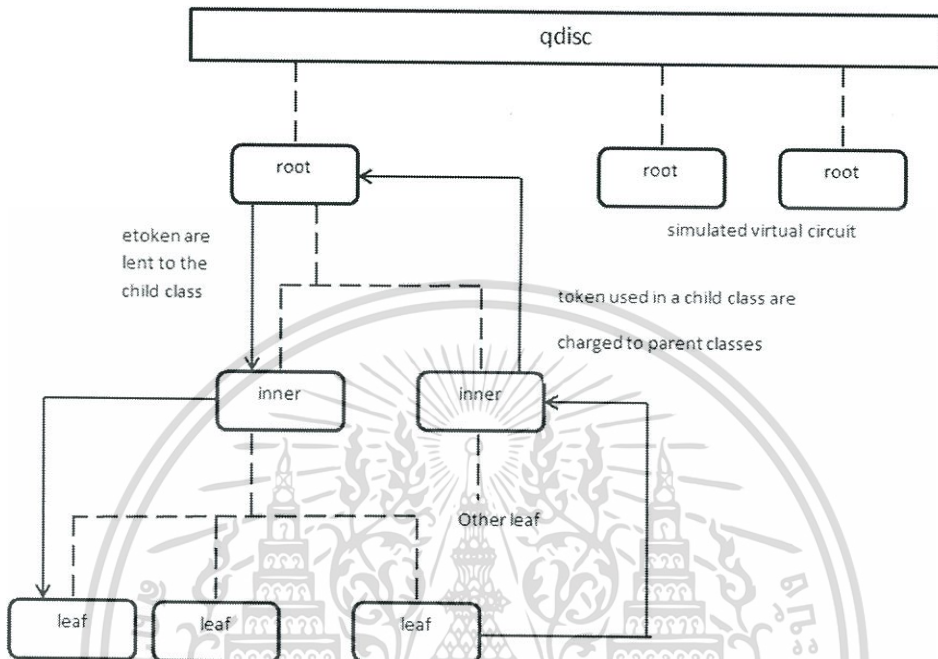
3.1.2 ส่วนควบคุมทราฟฟิก

3.1.2.1 IPTABLE



รูปที่ 3.2 การทำงานของ IPTABLE

3.1.2.2 ส่วนควบคุมแบนวิดธ์



รูปที่ 3.3 การทำงานของส่วนควบคุมแบนวิดธ์

3.1.2.3 DHCP

ทำการตั้งค่าระบบ DHCP โดยกำหนด subnet ที่จะให้ DHCP ทำงานเป็น 192.168.100.0/24 และช่วง IP ที่จะให้ระบบ DHCP ตั้งค่าอัตโนมัติเป็นช่วงเป็น 192.168.100.10 - 192.168.100.200

3.2 เครื่องมือที่ใช้ในการทดลอง

3.2.1 ซอฟต์แวร์

- 1) โปรแกรม IPTABLES
- 2) โปรแกรม iftop
- 3) โปรแกรม traffic control
- 4) โปรแกรม bootsrap
- 5) โปรแกรม HTML
- 6) โปรแกรม Apache
- 7) โปรแกรม Quick 'n easy FTP server
- 8) ระบบปฏิบัติการ Debian Linux
- 9) ระบบปฏิบัติการ windows 7

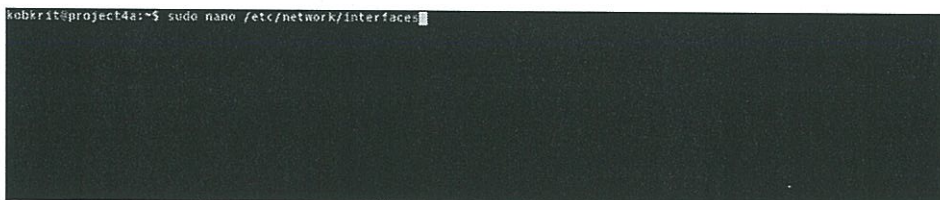
3.2.2 ฮาร์ดแวร์

- | | | |
|-------------------------------------|-------|-----------|
| 1) เครื่อง server HP ProLiant ML370 | จำนวน | 1 เครื่อง |
| 2) เครื่องคอมพิวเตอร์เวิร์กสเตชัน | จำนวน | 1 เครื่อง |
| 3) เครื่องคอมพิวเตอร์เว็บไคลเอนต์ | จำนวน | 1 เครื่อง |
| 4) switch layer 3 | จำนวน | 1 ตัว |
| 5) ET-GSM SIM900B module | จำนวน | 1 ตัว |

3.3 การจัดเก็บผลการทดลอง

3.3.1 การทดสอบการทำงานของระบบ NAT

ก่อนทำการทดลองเราจะตั้งค่าพอร์ต eth0 และ eth1 ของเซิร์ฟเวอร์ เพื่อทำการกำหนด IP Address ของเครื่องเซิร์ฟเวอร์ โดยป้อนคำสั่งผ่านหน้าต่าง Terminal ด้วยคำสั่ง “sudo nano /etc/network/interfaces”



```
kobkrit@project4a:~$ sudo nano /etc/network/interfaces
```

รูปที่ 3.4 การใช้คำสั่งตั้งค่าพอร์ต eth0 และ eth1 ของเซิร์ฟเวอร์

ทำการตั้งค่าพอร์ต eth0 และ eth1 ซึ่งเราจะกำหนด IP Address ของพอร์ต eth0 ซึ่งเป็น Public IP โดยใช้ DHCP ซึ่งเป็นการรับค่า IP Address ที่สุ่มมาและพอร์ต eth1 ซึ่งเป็น Private IP เราตั้ง IP Address เป็น 192.168.100.2/24 ดังรูป

```
GNU nano 2.2.6 File: /etc/network/interfaces
allow-hotplug eth0
iface eth0 inet dhcp

allow-hotplug eth1
iface eth1 inet static
    address    192.168.100.2
    netmask    255.255.255.0
    network    192.168.100.0
    broadcast  192.168.100.255
```

รูปที่ 3.5 กำหนดค่า IP Address ให้กับพอร์ต eth0 และ eth1 ของเซิร์ฟเวอร์

เมื่อทำการตั้งค่าเสร็จเราจะทำการตรวจสอบพอร์ต eth0 โดยใช้คำสั่ง sudo ipconfig eth0

```
kobkrit@project4a:~$ sudo ipconfig eth0
eth0      Link encap:Ethernet  HWaddr 00:0c:29:dce:13:1e
          inet addr:161.246.18.175  Bcast:161.246.18.255  Mask:255.255.255.0
          inet6 addr: fe80::20d:9dff:eece:131e/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3092607 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1671620 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3629547897 (3.3 GiB)  TX bytes:1176257150 (1.0 GiB)
          Interrupt:29

kobkrit@project4a:~$
```

รูปที่ 3.6 แสดงผลของการตั้งค่าพอร์ต eth0

และทำการตรวจสอบพอร์ต eth0 โดยใช้คำสั่ง sudo ipconfig eth0

```
kobkrit@project4a:~$ sudo ipconfig eth1
eth1      Link encap:Ethernet  HWaddr 00:11:0a:86:f6:4f
          inet addr:192.168.100.2  Bcast:192.168.100.255  Mask:255.255.255.0
          inet6 addr: fe80::211:aff:f68e:f64f64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1642477 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2883179 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1173542296 (1.0 GiB)  TX bytes:3598067775 (3.3 GiB)
          Interrupt:16

kobkrit@project4a:~$
```

รูปที่ 3.7 แสดงผลของการตั้งค่าพอร์ต eth1

จากรูปที่ 3.6 และรูปที่ 3.7 ได้แสดงให้เห็นว่าพอร์ต eth0 และพอร์ต eth1 ได้ถูกตั้งค่าดังนี้
 พอร์ต eth0 มี IP Address คือ 161.246.18.175
 พอร์ต eth1 มี IP Address คือ 192.168.100.2

หลังจากทำการตั้งค่าพอร์ต eth0 และ eth1 แล้ว เราจะทำการทดสอบการทำงานของพอร์ต eth0 โดยใช้คำสั่ง ping -c 4 161.246.18.175

```
kobkrit@kobkrit-A00255E:~$ ping -c 4 161.246.18.175
PING 161.246.18.175 (161.246.18.175) 56(84) bytes of data:
64 bytes from 161.246.18.175: icmp_seq=1 ttl=64 time=0.280 ms
64 bytes from 161.246.18.175: icmp_seq=2 ttl=64 time=0.279 ms
64 bytes from 161.246.18.175: icmp_seq=3 ttl=64 time=0.260 ms
64 bytes from 161.246.18.175: icmp_seq=4 ttl=64 time=0.210 ms

--- 161.246.18.175 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3001ms
rtt min/avg/max/mdev = 0.210/0.257/0.280/0.030 ms
kobkrit@kobkrit-A00255E:~$
```

รูปที่ 3.8 การทดสอบการทำงานของพอร์ต eth0

และทำการทดสอบการทำงานของพอร์ต eth1 โดยใช้คำสั่ง ping -c 4 192.168.100.2

```
kobkrit@kobkrit-A00255E:~$ ping -c 4 192.168.100.2
PING 192.168.100.2 (192.168.100.2) 56(84) bytes of data:
64 bytes from 192.168.100.2: icmp_seq=1 ttl=64 time=0.250 ms
64 bytes from 192.168.100.2: icmp_seq=2 ttl=64 time=0.258 ms
64 bytes from 192.168.100.2: icmp_seq=3 ttl=64 time=0.237 ms
64 bytes from 192.168.100.2: icmp_seq=4 ttl=64 time=0.276 ms

--- 192.168.100.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
rtt min/avg/max/mdev = 0.237/0.250/0.276/0.019 ms
kobkrit@kobkrit-A00255E:~$
```

รูปที่ 3.9 การทดสอบการทำงานของพอร์ต eth1

จากนั้นตั้งค่าของระบบ DHCP โดยใช้คำสั่ง sudo nano /etc/dhcp/dhcpd.conf

```
kobkrit@project4a:~$ sudo nano /etc/dhcp/dhcpd.conf
```

รูปที่ 3.10 การใช้คำสั่งตั้งค่าระบบ DHCP ของเซิร์ฟเวอร์

ทำการตั้งค่าระบบ DHCP โดยกำหนด subnet ที่จะให้ DHCP ทำงานเป็น 192.168.100.0/24 และช่วง IP ที่จะให้ระบบ DHCP ตั้งค่าอัตโนมัติเป็นช่วงเป็น 192.168.100.10 - 192.168.100.200 ดังรูป

```

GNU nano 2.2.6 File: /etc/dhcp/dhcpd.conf
# based on that. The example below shows a case where all clients
# in a certain class get addresses on the 10.17.224/24 subnet, and all
# other clients get addresses on the 10.0.29/24 subnet.

#class "foo" {
# match if substring (option vendor-class-identifier, 0, 4) = "SUHM";
#}

#shared-network 224-29 {
# subnet 10.17.224.0 netmask 255.255.255.0 {
# option routers rtr-224.example.org;
# }
# subnet 10.0.29.0 netmask 255.255.255.0 {
# option routers rtr-29.example.org;
# }
# pool {
# allow members of "foo";
# range 10.17.224.10 10.17.224.250;
# }
# pool {
# deny members of "foo";
# range 10.0.29.10 10.0.29.230;
# }
#}

subnet 192.168.100.0 netmask 255.255.255.0
{
    range 192.168.100.10 192.168.100.200;
    option subnet-mask 255.255.255.0;
    option broadcast-address 192.168.100.255;
    option routers 192.168.100.2;
    option domain-name-servers 161.246.52.21;
}

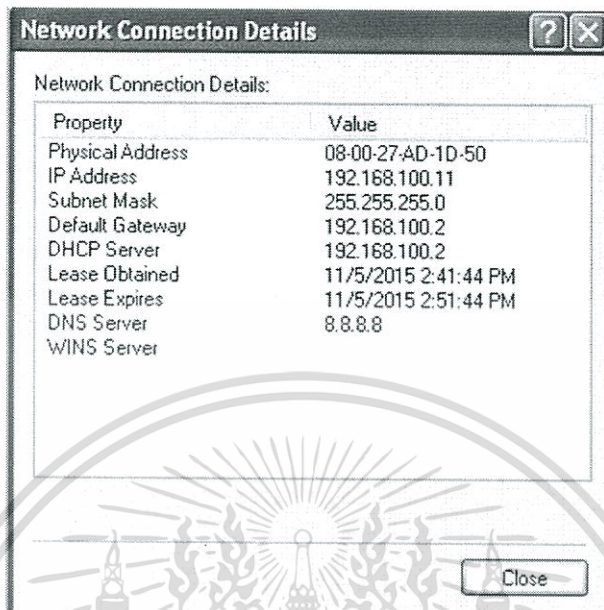
```

รูปที่ 3.11 กำหนดค่าช่วง IP Address ให้ระบบ DHCP ของเซิร์ฟเวอร์

หลังจากทำการตั้งค่า DHCP แล้ว เราจะทำการทดสอบการทำงานของ DHCP โดยการต่อสาย UTP จากเครื่องลูกข่ายเข้ากับพอร์ต eth1 ของเครื่องเซิร์ฟเวอร์ และใช้โปรแกรม Wireshark ตรวจสอบการทำงานของ DHCP

No.	Time	Source	Destination	Protocol	Length	Info
21	36.1334990	192.168.100.2	255.255.255.255	DHCP	342	DHCP offer - Transaction ID 0xfb2a98bb
<ul style="list-style-type: none"> ⊗ Frame 21: 342 bytes on wire (2736 bits), 342 bytes captured (2736 bits) on interface 0 ⊗ Ethernet II, Src: CadmusCo_1c:85:4c (08:00:27:1c:85:4c), Dst: Broadcast (ff:ff:ff:ff:ff:ff) ⊗ Internet Protocol version 4, Src: 192.168.100.2 (192.168.100.2), Dst: 255.255.255.255 (255.255.255.255) ⊗ User Datagram Protocol, Src Port: bootps (67), Dst Port: bootpc (68) ⊗ Bootstrap Protocol <ul style="list-style-type: none"> Message type: Boot Reply (2) Hardware type: Ethernet (0x01) Hardware address length: 6 Hops: 0 Transaction ID: 0xfb2a98bb Seconds elapsed: 0 ⊗ Bootp flags: 0x8000 (Broadcast) Client IP address: 0.0.0.0 (0.0.0.0) Your (client) IP address: 192.168.100.11 (192.168.100.11) Next server IP address: 0.0.0.0 (0.0.0.0) Relay agent IP address: 0.0.0.0 (0.0.0.0) Client MAC address: CadmusCo_ad:1d:50 (08:00:27:ad:1d:50) Client hardware address padding: 00000000000000000000 Server host name not given Boot file name not given Magic cookie: DHCP ⊗ Option: (53) DHCP Message Type <ul style="list-style-type: none"> Length: 1 						
0000	ff ff ff ff ff ff	08 00	27 1c 85 4c	08 00 45 10L..E.	
0010	01 48 00 00 00 00	00 80 11	14 eb c0 a8 64	02 ff ff	.H.....	
0020	ff ff 00 43 00 44	01 34 10 fe	02 01 06 00	fb 2a	...C.D.4.....	
0030	98 bb 00 00 00 00	00 00 00 00	00 00 c0 a8 64	0b 00 00	
0040	00 00 00 00 00 00	00 00 08 00	27 ad 1d 50	00 00 00P....	
0050	00 00 00 00 00 00	00 00 00 00	00 00 00 00	00 00 00	

รูปที่ 3.12 package DHCP Offer ของ DHCP เซิร์ฟเวอร์ ที่ตรวจพบโดยโปรแกรม Wireshark



รูปที่ 3.13 หมายเลข IP address ที่เครื่องลูกข่ายได้รับจาก DHCP เซิร์ฟเวอร์

จากนั้นตั้งค่าระบบ NAT โดยใช้คำสั่ง `sudo iptables -t nat -A POSTROUTING -s 192.168.100.0/24 -o eth0 -j MASQUERADE`

```
kobkrit@project40:~$ sudo iptables -t nat -A POSTROUTING -s 192.168.100.0/24 -o eth0 -j MASQUERADE
```

รูปที่ 3.14 การใช้คำสั่งตั้งค่าระบบ NAT ของเซิร์ฟเวอร์

ทำการตั้งค่า Linux ให้อนุญาตส่งแพ็คเก็ตข้ามอินเทอร์เน็ตเฟสโดยใช้คำสั่ง `sudo nano /etc/sysctl.conf`

```

GNU nano 2.2.6 File: /etc/sysctl.conf
#
# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.

#kernel.domainname = example.com

# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3

#####3
# Functions previously found in netbase

# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1

# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/277146/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1

# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1

# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1

```

รูปที่ 3.15 การตั้งค่าไฟล์ sysctl.conf ของเซิร์ฟเวอร์

ทดสอบการทำงานของระบบ NAT โดยใช้คำสั่ง ping -c 4 kmitl.ac.th
ที่เครื่องลูกข่าย

```

kobkrit@kobkrit-A0D255E:~$ ping -c 4 kmitl.ac.th
PING kmitl.ac.th (161.246.34.11) 56(84) bytes of data:
64 bytes from kmitl.ac.th (161.246.34.11): icmp_seq=1 ttl=62 time=0.594 ms
64 bytes from kmitl.ac.th (161.246.34.11): icmp_seq=2 ttl=62 time=0.573 ms
64 bytes from kmitl.ac.th (161.246.34.11): icmp_seq=3 ttl=62 time=0.599 ms
64 bytes from kmitl.ac.th (161.246.34.11): icmp_seq=4 ttl=62 time=0.534 ms

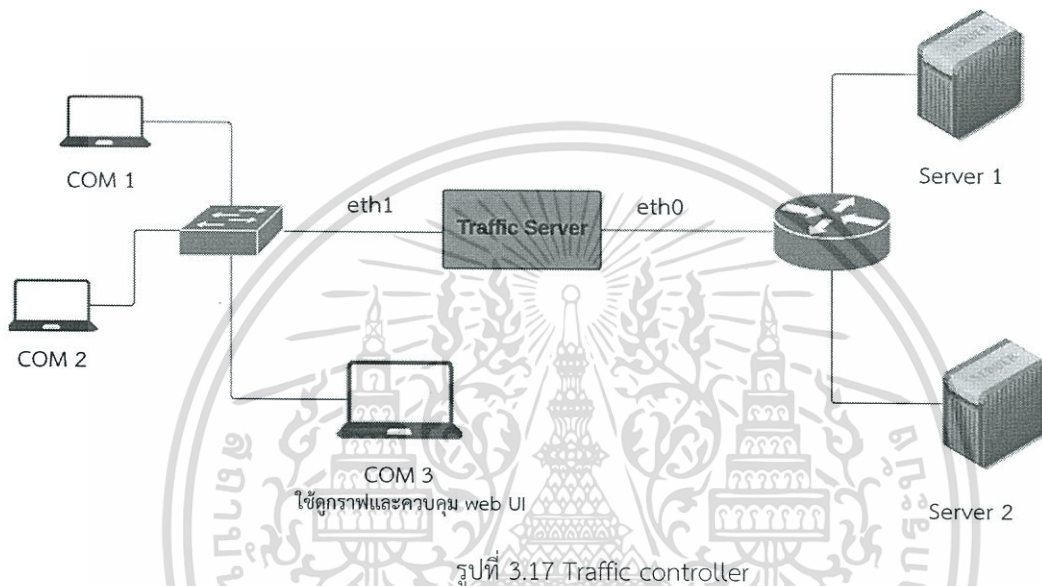
--- kmitl.ac.th ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 0.534/0.575/0.599/0.023 ms
kobkrit@kobkrit-A0D255E:~$

```

รูปที่ 3.16 การทดสอบการทำงานของระบบ NAT

3.3.2 การจัดเก็บผลการจัดทราฟฟิก

การจัดเก็บผลการจัดทราฟฟิก ในการทดลองได้ทำการดาวน์โหลดไฟล์เพื่อจำลองการให้บริการโปรโตคอล HTTP และโปรโตคอล HTTPS โดยได้ทำการทดลองทั้งสิ้น 4 การทดลองโดยลักษณะการเก็บผลเป็นไปตามไดอะแกรมดังต่อไปนี้



1. ความเร็วไฟล์จาก web server โดยไม่มีการจำกัดทราฟฟิก ทำได้โดยมีการตั้งค่าระบบ Traffic control server ผ่านทาง Web UI โดยใช้ตัวแปรดังต่อไปนี้

ประเภทของโปรโตคอลที่ควบคุม : HTTP
 ลาเบล : ไม่มี
 ลำดับที่ : ไม่มี
 ความเร็วที่ต้องการจำกัด : 0 KBps
 ความเร็วสูงสุด : 0 KBps

SPEED SETTING

APPLICATION	LABEL	PRIORITY	LIMITED SPEED	CEIL SPEED	UNITS
HTTP					Kbps

Save Clear All

รูปที่ 3.18 การดาวน์โหลดไฟล์โดยไม่มีกำหนดทราฟฟิกของ HTTP

2. ดาวน์โหลดไฟล์จาก web server โดยมีการจำกัดทราฟฟิกของโปรโตคอล HTTP
ทำได้โดยมีการตั้งค่าระบบ Traffic control server ผ่านทาง Web UI โดยใช้ตัวแปรดังต่อไปนี้

ประเภทของโปรโตคอลที่ควบคุม : HTTP
 ลำเบล : 1
 ลำดับที่ : 1
 ความเร็วที่ต้องการจำกัด : 200 KBps
 ความเร็วสูงสุด : 250 KBps

SPEED SETTING

APPLICATION	LABEL	PRIORITY	LIMITED SPEED	CEIL SPEED	UNITS
HTTP ▾	<input type="text" value="1"/>	<input type="text" value="1"/>	<input type="text" value="200"/>	<input type="text" value="250"/>	Kbps ▾
<input type="button" value="Save"/>			<input type="button" value="Clear All"/>		

รูปที่ 3.19 การดาวน์โหลดไฟล์โดยมีกำหนดทราฟฟิกของ HTTP

3. ดาวน์โหลดไฟล์จาก www.youtube.com ทำได้โดยมีการตั้งค่าระบบ Traffic control server ผ่านทาง Web UI โดยใช้ตัวแปรดังต่อไปนี้

ประเภทของโปรโตคอลที่ควบคุม : HTTPS
 ลำเบล : ไม่มี
 ลำดับที่ : ไม่มี
 ความเร็วที่ต้องการจำกัด : 0 KBps
 ความเร็วสูงสุด : 0 KBps

SPEED SETTING

APPLICATION	LABEL	PRIORITY	LIMITED SPEED	CEIL SPEED	UNITS
HTTPS ▾	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	Kbps ▾
<input type="button" value="Save"/>			<input type="button" value="Clear All"/>		

รูปที่ 3.20 การดาวน์โหลดไฟล์โดยไม่มีกำหนดทราฟฟิกของ HTTPS

4. ดาวน์โหลดไฟล์จาก www.youtube.com ทำได้โดยมีการตั้งค่าระบบ Traffic control server ผ่านทาง Web UI โดยใช้ตัวแปรดังต่อไปนี้

ประเภทของโปรโตคอลที่ควบคุม : HTTPS
 ลาเบล : 2
 ลำดับที่ : 2
 ความเร็วที่ต้องการจำกัด : 200 KBps
 ความเร็วสูงสุด : 250 KBps

SPEED SETTING

APPLICATION	LABEL	PRIORITY	LIMITED SPEED	CEIL SPEED	UNITS
HTTPS	2	2	200	250	Kbps

Save Clear All

รูปที่ 3.21 การดาวน์โหลดไฟล์โดยมีกำหนดกราฟฟิกของ HTTPS

3.3.3 การจัดเก็บผลการทดลองการแจ้งเตือนผ่าน GSM โมดูล ในกรณีที่ไม่มี
 เชื่อมต่ออินเทอร์เน็ต

- ทดลองดึงสายแลนออกจากเครื่อง Sever เพื่อจำลองสถานการณ์ขาดการ

เชื่อมต่อ

- รอรับผลการแจ้งเตือน SMS

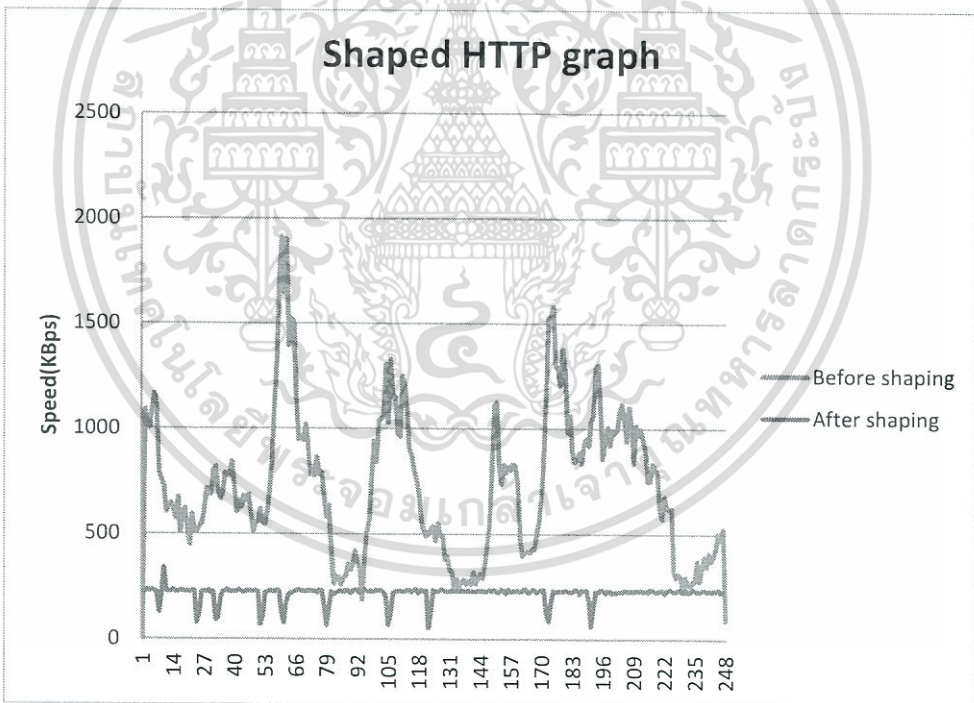
บทที่ 4

ผลการทดลอง

4.1 ผลการทดลอง

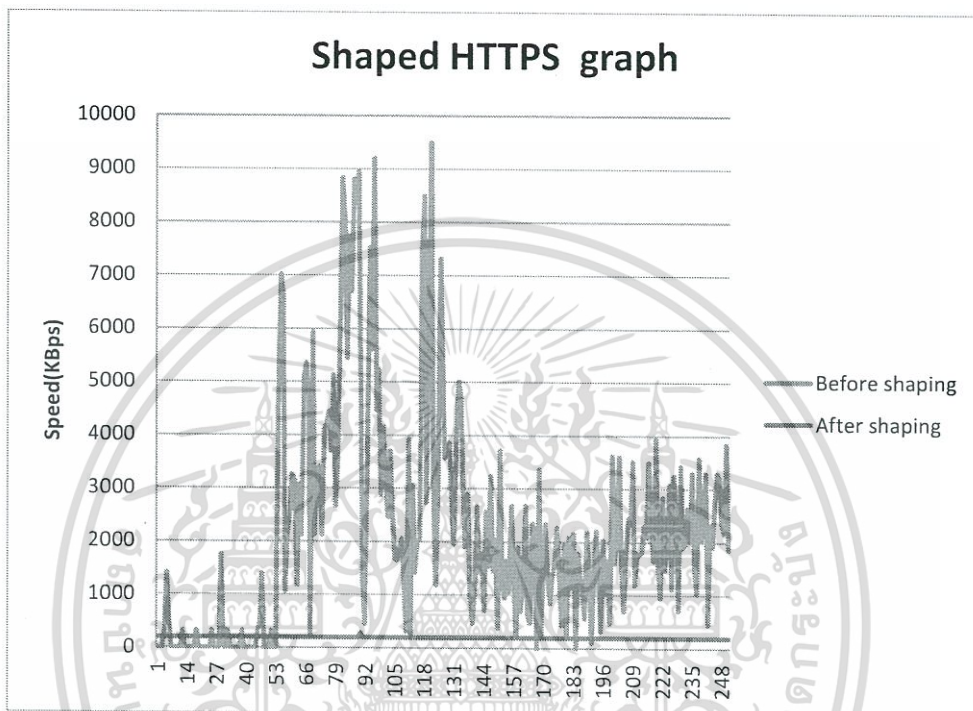
ในการทดลองผู้ทดลองได้ทำการทดลองทั้งหมด 5 การทดลอง โดยแบ่งเป็นการ 4 การทดลอง โดยแบ่งเป็นการทดลองทำกราฟฟิคโดยไม่มีการจำกัดกราฟฟิค 2 การทดลอง และการทดลองทำกราฟฟิคโดยมีการจำกัดกราฟฟิคอีก 1 การทดลอง และการทดลองการทำงานของ GSM โมดูลอีก 2 การทดลองโดยแบ่งเป็นการทดลองการทำงานแบบไม่เชื่อมต่ออินเทอร์เน็ต

4.1.1 ทดลองการดาวน์โหลดข้อมูลโดยใช้โปรโตคอล HTTP ทั้งแบบจำกัดและไม่จำกัดกราฟฟิค โดยผลการทดลองเป็นดังนี้



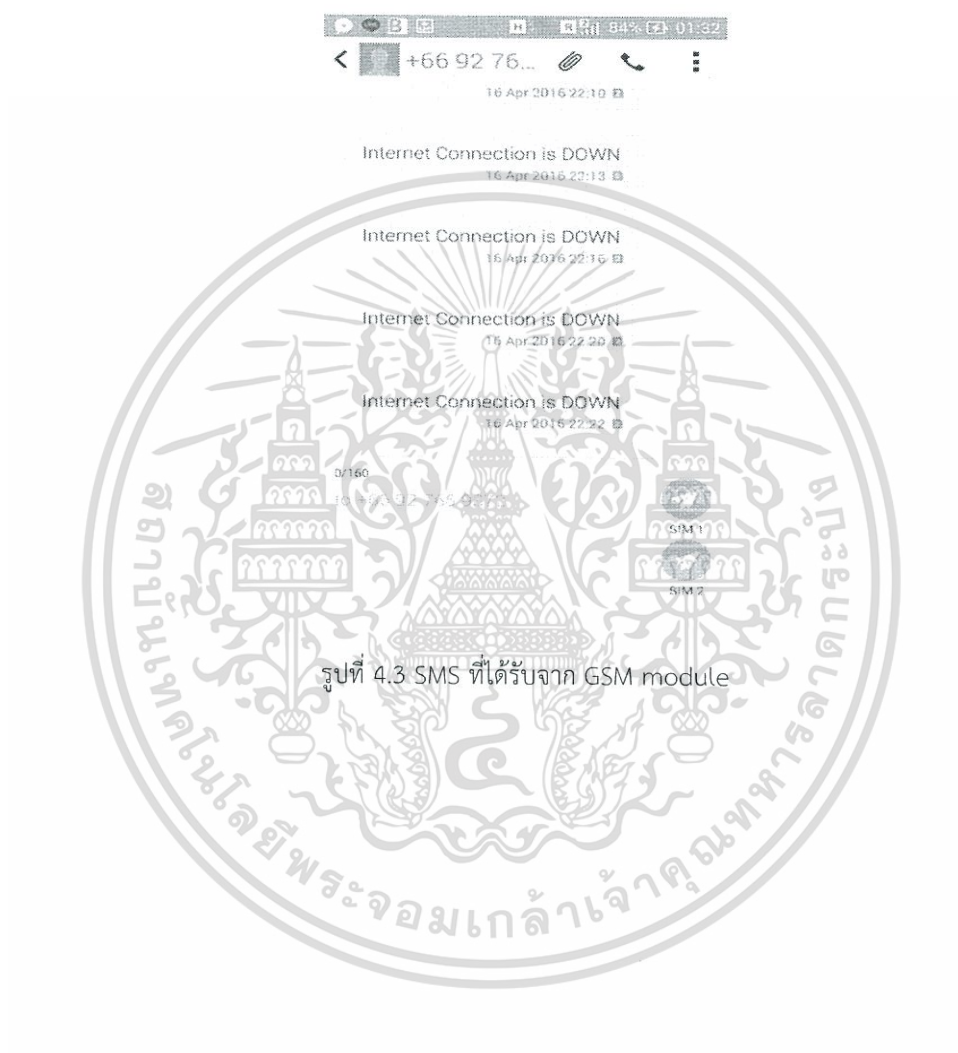
รูปที่ 4.1 กราฟแสดงปริมาณกราฟฟิคเมื่อทำการดาวน์โหลดไฟล์ของ HTTP

4.1.2 ทดลองการดาวน์โหลดข้อมูลด้วยโปรโตคอล HTTPS ทั้งแบบที่มีการจำกัดกราฟฟิกและไม่มีการจำกัดกราฟฟิกโดยผลการทดลองเป็นดังนี้



รูปที่ 4.2 กราฟแสดงปริมาณกราฟฟิกเมื่อทำการดาวน์โหลดไฟล์ของ HTTPS

4.1.3 ทดลองการแจ้งเตือน ผ่าน SMS เมื่อขาดการเชื่อมต่อกับอินเทอร์เน็ต มีผลการทดลองเป็นดังนี้



บทที่ 5

สรุปผลและข้อเสนอแนะ

จากทฤษฎีและหลักการต่างๆที่เกี่ยวข้องซึ่งได้นำมาสร้างชิ้นงานและเก็บผลการทดสอบ ทำให้ทางคณะผู้จัดทำได้ทราบว่าในทางทฤษฎีกับหลักการปฏิบัตินั้นมีความแตกต่างกันและมีปัญหาบางประการที่จำเป็นต้องใช้ความพยายามในการแก้ไขปัญหาที่อาจจะเกิดขึ้น

5.1 สรุปผล

สรุปผลการทดลองในส่วนของการออกแบบระบบในระบบปฏิบัติการลินุกซ์ซึ่งได้มีการทำการดาวน์โหลดไฟล์ ทั้งแบบที่มีการจำกัดกราฟฟิกและไม่มีการจำกัดกราฟฟิกของทั้ง HTTP และ HTTPS ผลที่ได้ออกมาเป็นที่น่าสนใจเพราะสามารถจัดการกราฟฟิกได้ตามที่กำหนด

สรุปผลการทดลองในส่วนของการทดสอบการทำงานบนหน้าเว็บไซต์สามารถที่จะรับค่าตัวแปรที่กำหนดและนำไปประมวลผลในระบบได้และยังมีการแสดงหน่วยความจำของอุปกรณ์ที่ใช้ติดตั้งระบบประมวลผลได้

สรุปผลการทดลองในส่วนของ GSM โมดูลสามารถที่จะแจ้งเตือนสถานะการเชื่อมต่ออินเทอร์เน็ตของอุปกรณ์ได้และจะแจ้งมายังผู้ใช้งานในกรณีที่ไม่มีการเชื่อมต่ออินเทอร์เน็ต

5.2 ข้อเสนอแนะ

ผู้สนใจควรมีความรู้พื้นฐานเกี่ยวกับหลักการการทำงานของระบบโครงข่ายและการใช้งานระบบปฏิบัติการลินุกซ์เป็นอย่างดีรวมถึงชุดคำสั่ง shell script ควรมีการพัฒนาให้มีการรองรับอัลกอริทึมได้หลากหลายแบบ

เนื่องจาก traffic control บางประเภทนั้นไม่มีพอร์ตที่แน่นอนทำให้การที่จะเข้าควบคุม traffic control ประเภทนี้ต้องมีเครื่อง server คอยตรวจจับว่าการเชื่อมต่อที่ได้มานั้นเป็นประเภทของกราฟฟิกประเภทใดโดยอ่านข้อมูลจากการเชื่อมต่อเหล่านั้นแต่ถ้ามีการเข้ารหัสข้อมูลมาเราจะไม่สามารถแบ่งการเชื่อมต่อได้เลยทำให้ควบคุมไม่ได้เช่น Tor , Bit Torrent เป็นต้น

บรรณานุกรม

- [1] เรืองไกร รังสิพล, “เจาะระบบ TCP/ IP จุดอ่อนของโปรโตคอลและวิธีป้องกัน”
บริษัท โปริวิชั่น จำกัด
- [2] John D. Sprins, Joseph L.Hammond and Krzysztof Pawlikowski
Telecommunication Protocols and design Addison Wesley, 1991





Total.php

```
<?php require_once 'connect.php';
$data = array();
if(isset($_GET['id'])){
    $sql = "SELECT * FROM testspeed WHERE id = ".$_GET['id'];
    $result = mysql_query($sql);
    $data = mysql_fetch_assoc($result);
}
?>
<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta http-equiv="Content-Type" content="text/html; charset=tis-620" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
    <title>Traffic Sharper</title>
    <link rel="stylesheet" type="text/css" href="/css/bootstrap.min.css">
    <script type="text/javascript" src="/js/jquery.js"></script>
    <script type="text/javascript" src="/js/bootstrap.min.js"></script>
    <script>
var reimg
var reimg1
window.onload=function () {
    reimg=document.getElementById('re')// reimg match re สำหรับกำหนด id
    setInterval(function () {
        d = new Date();
        imglink = 'configgraph.php?'+d.getTime(); //เพิ่มตัวแปรให้ php
        reimg.src=imglink; // เปลี่ยนตามด้านบน
    },1000)

    reimg1=document.getElementById('re1')// reimg match re สำหรับกำหนด id
```

```

        setInterval(function () {
            d = new Date();
            imglink = 'speed.php?' + d.getTime(); //เพิ่มตัวแปรให้ php
            reimg1.src=imglink; // เปลี่ยนตามด้านบน
        },1000)
    }
</script>
<script language="re">
    function refreshIt(element) {
        setTimeout(function() {
            element.src = element.src.split('?')[0] + '?' + new Date().getTime();
            refreshIt(element);
        }, 10); // refresh every 50ms
    }
</script>
<script language="javascript">
function Change()
{
if (document.form1.check1.checked == true)
{ document.form1.minspeed.disabled = true;
  document.form1.modesp.disabled = true; }
else
{ document.form1.minspeed.disabled = false;
  document.form1.modesp.disabled = false; }
}
function Change1()
{
if (document.form1.check2.checked == true)
{ document.form1.maxspeed.disabled = true;
  document.form1.modesp2.disabled = true; }
}

```

```

else
{ document.form1.maxspeed.disabled = false;
  document.form1.modesp2.disabled = false; }
}
</script>
</head>

<body>
  <div class="container" style="margin-top:10px">

    <!-- เริ่ม Grid System -->
    <div class="row">
      <div class="page-header">
        <center><h1>Traffic shaper
        <small>Tester</small></h1></center>
      </div>
    </div>

    <!-- ใส่อินโฟที่ -->
    <div class="container">
      <div class="row">
        <div class="col-md-12">
          <div class="panel panel-default">
            <div class="panel-heading">
              <h2>MODEL : HP</h2>
            </div>
            <div class="panel-body">
              <center></center>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```

```

</div>
</div>
</div>
<div class="row">
  <div class="col-md-12">
    <div class="panel panel-default">
      <div class="panel-heading">
        <h2>SPEED SETTING</h2>
      </div>
      <form name="form1" method="POST"
action="TC.php">
        <table border="0" align="center" class="table table-striped" style="width:95%">
          <br>
          <tr>
            <th bgcolor="#B4DBF7"><center>MODE</center></th>
            <th bgcolor="#B4DBF7"><center>APPLICATION</center></th>
            <th bgcolor="#B4DBF7"><center>MINIMUM
SPEED</center></th>
            <th bgcolor="#B4DBF7"><center>MAXIMUM
SPEED</center></th>
            <th bgcolor="#B4DBF7"><center>SUBMIT</center></th>
          </tr>
          <tr>
            <td><center><select name="ud">
              <option value="upload">UPLOAD</option>
              <option value="download">DOWNLOAD</option>
            </center>
          </td>
        </tr>
      </table>
    </div>
  </div>
</div>

```

```

<td><center><select name="modepro">
<option value="web">Web</option>
<option value="video">Video</option>
<option value="ftp">FTP</option>
<option value="ssh">SSH</option>
<option value="dns">DNS</option>
</center>
</td>
<td><center><input type="checkbox" name="check1"
onClick="Change0"/> No Qos <input id="minspeed" name="minspeed" type="text"
placeholder="Number only">
<select id="modesp" name="modesp">
<option value="kpbs">Kbps</option>
<option value="Mbps">Mbps</option>
</center>
</td>
<td><center><input type="checkbox" name="check2"
onClick="Change1("> No Limit <input id="maxspeed" name="maxspeed" type="text"
placeholder="Number only">
<select id="modesp2" name="modesp2">
<option value="kpbs">Kbps</option>
<option value="Mbps">Mbps</option>
</center>
</td>
<td><center><input name="save" value="save"
type="submit"></center></td>

```

```

        </tr>
    </table>

    <?php
    if(isset($_GET['id'])){
        echo '<input type="hidden" name="id" value="' . $_GET['id'].
"/>';

    }
    ?>
</form>

</div>
</div>
</div>
<div class="container">
    <div class="row">
        <div class="col-md-12">
            <div class="panel panel-default">
                <div class="panel-heading">
                    <h2>APPLICATION SETTING</h2>
                </div>
                <div class="panel-body">
                    <form name="form" method="post"
action="save_project.php" enctype="multipart/form-data">
                        <table border="0" style="width:80%" align="center" >

```

```

<tr>

    <th bgcolor="#B4DBF7" ><center>APPLICATION</center></th>
    <th bgcolor="#B4DBF7" ><center>PORT </center></th>
    <th bgcolor="#B4DBF7" ><center>IP ADDRESS</center></th>
    <td align="center"><input type="submit" neme="save" value="SAVE"
style="width:100%" /> </td>

</tr>
<tr>
    <td><center><input type="text" name="app"
value="<?php echo (isset($data['app']))? $data['app']:'; ?>" ></center></td>
    <td><center><input type="text" name="port"
value="<?php echo (isset($data['port']))? $data['port']:'; ?>"></center></td>
    <td><center><input type="text" name="address"
value="<?php echo (isset($data['address']))? $data['address']:'; ?>"></center></td>
    <td align="center"><input type="button"
onClick="window.location='list_project.php';" value="LIST" style="width:100%" /> </td>

</tr>
</table>
<?php
if(isset($_GET['id'])){
echo '<input type="hidden" name="id" value=" '.$_GET['id'].' "/>';
}

?>
</div>
</div>
</div>

```

```
<div class="container">
<div class="row">
  <div class="col-md-12">
    <div class="panel panel-default">
      <div class="panel-heading">
        <h2>SPEED</h2>
      </div>
      <div class="panel-body">
        <center></center>
      </div>
    </div>
  </div>
</body>
</html>
```



TC.php

```
<html>
```

```
<header>
```

```
<title>Processor</title></header>
```

```
<script>
```

```
function popup() {
```

```
    alert("Invaild input");
```

```
    window.location = "total.php";
```

```
}
```

```
</script>
```

```
<script>
```

```
function popqos() {
```

```
    alert("QOS over BW");
```

```
    window.location = "total.php";
```

```
}
```

```
</script>
```

```
<script>
```

```
function popok() {
```

```
    alert("OK");
```

```
    window.location = "list_project_2.php";
```

```
}
```

```
</script>
```

```
<script>
```

```
function poperror() {
```

```
    alert("Error something wrong");
```

```
    window.location = "total.php";
```

```
}
```

```
</script>
```

```

<body>
<?php
$max0 = $_POST[maxspeed];
$min0 = $_POST[minspeed];
$getunit1 = $_POST[modesp];
$getunit2 = $_POST[modesp2];
$getmode = $_POST[modepro];
$check1 = $_POST[check1];
$check2 = $_POST[check2];
//echo '<h2>'.$check1.'::'.$check2.' </h2></br>';
if($check1!= "on")
{
    if( !is_numeric($min0) || $min0 == " " ){
        echo '<script type="text/javascript">popup();</script>';
    }
    else{
        $min=$min0;
    }
}
else
{
    $getunit1="";
    $min='noqos';
}
if($check2!= "on")
{

    if( !is_numeric($max0) || $max0 == " " ){
        echo '<script type="text/javascript">popup();</script>';
    }
}

```

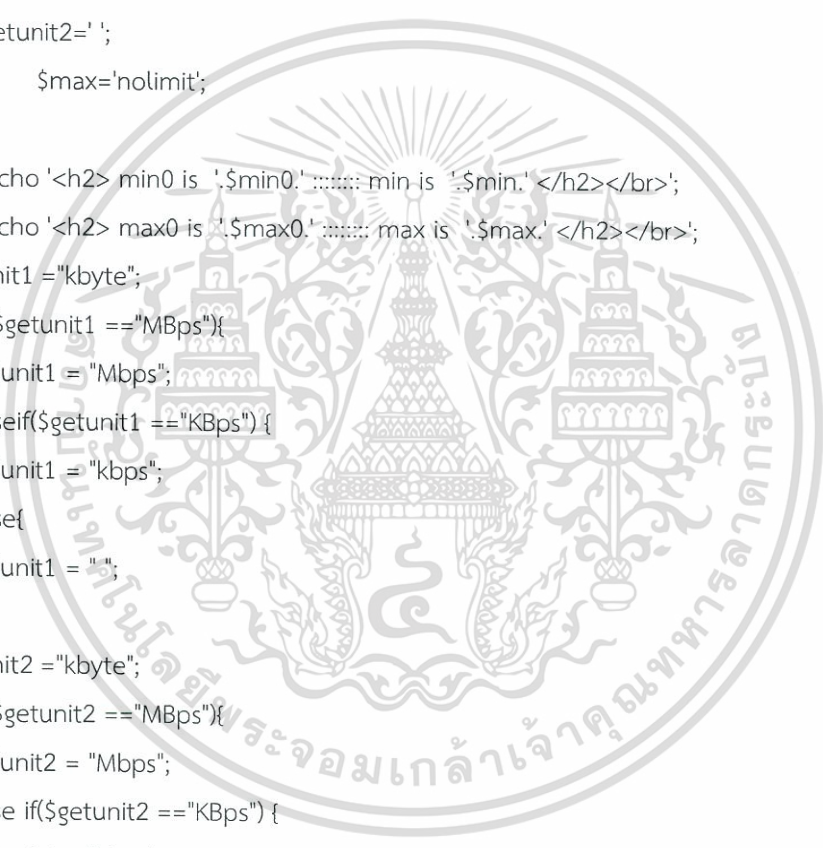
```

else{

        $max=$max0;

    }
}
else
{
    $getunit2=' ';
        $max='nolimit';
    }
//echo '<h2> min0 is '$min0.' : min is '$min.' </h2></br>';
//echo '<h2> max0 is '$max0.' : max is '$max.' </h2></br>';
$unit1="kbyte";
if($getunit1 == "MBps"){
    $unit1 = "Mbps";
}elseif($getunit1 == "KBps") {
    $unit1 = "kbps";
}else{
    $unit1 = " ";
}
$unit2="kbyte";
if($getunit2 == "MBps"){
    $unit2 = "Mbps";
}elseif($getunit2 == "KBps") {
    $unit2 = "kbps";
}else {
    $unit2 = " ";
}
}

```



```

//echo '<h2>Show'.'$getunit1.'.'$unit1.' and '$getunit2.'.'$unit2.' </h2></br>';
//echo '<h2>'.'$min.' '.'$unit1.'<----->'.'$max.' '.'$unit2.' </h2></br>';
$mod ="art";
if($getmode == "Web")
{$mod ="web";}
else if($getmode == "Video")
{$mod ="video";}
else if($getmode == "FTP")
{$mod ="ftp";}
else if($getmode == "SSH")
{$mod ="ssh";}
else if($getmode == "torrent")
{$mod ="torrent";}
else {$mod="dns";}

//echo '<h2>'.'$min.' '.'$unit1.'<----->'.'$max.' '.'$unit2.'<----->and mode is
'.'$mod.'</h2></br>';

$minspeed =$min.'"'.$unit1;
$maxspeed =$max.'"'.$unit2;

//Script POP
$result=(int)shell_exec("sudo service tcrule ',$mod.' ',$minspeed.' ',$maxspeed.'");

if($result == "0"){
$id =$_POST['id'];
//echo '<h2> ID is'.'$id.'</h2>';

```

```

require_once'connect.php';
$mode=$mod;
    if(isset($_POST['id'])) {
        $sql = "UPDATE testspeed SET mode = '$mode',min
        ='$minspeed',max='$maxspeed' WHERE id =$id " ;
    }else{
        $sql = "INSERT INTO testspeed (mode,min,max)"
        . "VALUES('$mode','$minspeed','$maxspeed)";
    }
//     echo 'shell_exec sudo service tcrule '.$mod.' '$minspeed.' '$maxspeed.' ;
mysql_query($sql) or die ('Invalid query: ' . mysql_error());
    echo '<script type="text/javascript">popok();</script>';
}else if($result == "1") {
    echo '<script type="text/javascript">popqos();</script>';
}else{
    echo '<script type="text/javascript">poperror();</script>';
}
////////////////////
//<form id="form2" name="form2" method="POST" action="save.php">
//<input type="checkbox" name="check3" checked="checked"
onChecked="this.form.submit()"> '.$mod.'
//<input type="checkbox" name="check4" checked="checked"
onChecked="this.form.submit()"> '.$minspeed.'
//<input type="checkbox" name="check5" checked="checked"
onChecked="this.form.submit()"> '.$maxspeed.'
//<noscript><input type="submit" value="submit"></noscript>
//</form>
////////////////////

```

```

/*
$id = $_POST['id'];
//echo '<h2> ID is '$id.'</h2>';
require_once 'connect.php';

$minspeed = $min."".$unit1;
$maxspeed = $max."".$unit2;

$mode=$mod;
if(isset($_POST['id'])) {
    $sql = "UPDATE testspeed SET mode = '$mode',min
=$minspeed,max='$maxspeed' WHERE id = $id " ;
}
else{
    $sql = "INSERT INTO testspeed (mode,min,max)"
    . "VALUES('$mode','$minspeed','$maxspeed)";
}
// echo 'shell_exec sudo service tcrule '$mod.' '$minspeed.' '$maxspeed.' ';

mysql_query($sql) or die ('Invalid query: ' . mysql_error());

//echo 'shell_exec sudo service tcrule '$mod.' '$minspeed.' '$maxspeed.' '

//shell_exec("sudo service tcrule '$mod.' '$min.'".$unit1 . ' '$max.'".$unit2.' ");

?>

```

```
<input type="button" onClick="window.location='total.php';" value="BACK"/>
<input type="button" onClick="window.location='list_project_2.php';" value=" LIST"/>

</body>
</html>
```

skip.php

```
<?php
$interface0="enp5s0f1";
$interface1=" wlp4s0";
$fontPath="/usr/share/fonts/truetype/liberation/";

# PHPlot Example: Bar chart, 3 data sets, shaded
require_once 'phplot.php';

$lan_tx_pre = (8*(float)shell_exec("/sbin/ifconfig $interface0|grep -E
'[R][X][[:space:]]*[b][y][t][e]*' | awk '{ print $6}'|cut -d':' -f2"))/1048576;
$lan_rx_pre = (8*(float)shell_exec("/sbin/ifconfig $interface0|grep -E
'[R][X][[:space:]]*[b][y][t][e]*' | awk '{ print $2}'|cut -d':' -f2"))/1048576;
$wan_tx_pre = (8*(float)shell_exec("/sbin/ifconfig $interface1|grep -E
'[R][X][[:space:]]*[b][y][t][e]*' | awk '{ print $6}'|cut -d':' -f2"))/1048576;
$wan_rx_pre = (8*(float)shell_exec("/sbin/ifconfig $interface1|grep -E
'[R][X][[:space:]]*[b][y][t][e]*' | awk '{ print $2}'|cut -d':' -f2"))/1048576;
sleep(1);
$lan_tx_pos = (8*(float)shell_exec("/sbin/ifconfig $interface0|grep -E
'[R][X][[:space:]]*[b][y][t][e]*' | awk '{ print $6}'|cut -d':' -f2"))/1048576;
$lan_rx_pos = (8*(float)shell_exec("/sbin/ifconfig $interface0|grep -E
'[R][X][[:space:]]*[b][y][t][e]*' | awk '{ print $2}'|cut -d':' -f2"))/1048576;
```

```
$wan_tx_pos = (8*(float)shell_exec("/sbin/ifconfig $interface1|grep -E  
'[R][X][[:space:]]*[b][y][t][e]*' | awk '{ print $6}'|cut -d'!' -f2"))/1048576;
```

```
$wan_rx_pos = (8*(float)shell_exec("/sbin/ifconfig $interface1|grep -E  
'[R][X][[:space:]]*[b][y][t][e]*' | awk '{ print $2}'|cut -d'!' -f2"))/1048576;
```

```
$wan_tx = $wan_tx_pos - $wan_tx_pre;
```

```
$wan_rx = $wan_rx_pos - $wan_rx_pre;
```

```
$lan_tx = $lan_tx_pos - $lan_tx_pre;
```

```
$lan_rx = $lan_rx_pos - $lan_rx_pre;
```

```
$data = array(  
  array("WLAN", $wan_rx, $wan_tx),  
  array("eLAN", $lan_rx, $lan_tx),  
);
```

```
$colorList2 = array('#2E2EFE', '#FE2EF7');
```

```
$plot2 = new PHPPlot(800, 600);
```

```
$plot2->SetFailureImage(False); // No error images
```

```
$plot2->SetPrintImage(False); // No automatic output
```

```
$plot2->SetFontTTF('x_label', "$fontPath/LiberationSans-Regular.ttf", 16);
```

```
$plot2->SetFontTTF('y_label', "$fontPath/LiberationSans-Regular.ttf", 12);
```

```
$plot2->SetFontTTF('legend', "$fontPath/LiberationSans-Regular.ttf", 10);
```

```
$plot2->SetImageBorderStyle('plain');
```

```
// $plot2->SetBackgroundColor('yellow');
```

```
$plot2->SetDataColors($colorList2);
```

```
$plot2->SetPlotType('bars');
```

```
$plot2->SetDataType('text-data');
```

```
$plot2->SetDataValues($data);
```

```
// $plot2->SetYDataLabelPos('plotin');
```

```
// $plot2->SetYTickIncrement(0.5); //10.5
```

```
# Main plot title:
$plot2->SetTitle('Networking Interfaces Mb/s');
# Make a legend for the 3 data sets plotted:
$plot2->SetLegend(array('RX', 'TX'));
# Turn off X tick labels and ticks because they don't apply here:
$plot2->SetXTickLabelPos('none');
$plot2->SetXTickPos('none');
$plot2->SetBackgroundColor('#F2F2F2');
$plot2->DrawGraph()
echo $plot2->PrintImage();
?>
```





ShaperByPort.sh

```
#!/bin/bash
```

```
phone="66865525351"
```

```
termal="ttyUSB0"
```

```
TestIP=$1
```

```
SECOND=$2
```

```
GAIN=$3
```

```
COMMAND=`ping -c 5 -i $SECOND $TestIP | grep packet | awk '{ print $6}'|cut -d'%' -f1`
```

```
if [ $COMMAND -eq 100 ]
```

```
then
```

```
    sudo stty 19200 cs8 -raw -parenb -cstopb crtscts -echo -F /dev/$termal
```

```
    sudo echo -e AT+CMGS="\n+$phone\n"\x0D' > /dev/$termal
```

```
    echo " Internet Connection is DOWN $COMMAND % "
```

```
    sleep 3
```

```
    sudo echo -e "Internet Connection is DOWN" "\x1A' > /dev/$termal
```

```
elif [ $COMMAND -gt $GAIN ]
```

```
then
```

```
    sudo stty 19200 cs8 -raw -parenb -cstopb crtscts -echo -F /dev/$termal
```

```
    sudo echo -e AT+CMGS="\n+$phone\n"\x0D' > /dev/$termal
```

```
    echo " Network Too Lost: $COMMAND % "
```

```
    sleep 3
```

```
    sudo echo -e "Internet TOO LOST $COMMAND %" "\x1A' > /dev/$termal
```

```
elif [ $COMMAND -eq 0 ]
```

```
then
```

```
    echo " Normal Service"
```

```

else
    sudo stty 19200 cs8 -raw -parenb -cstopb crtscts -echo -F /dev/$termal
    sudo echo -e AT+CMGS="\'+$phone\"'\x0D' > /dev/$termal
    echo " Can't find DNS, or Check Internet Connection !!"
    sleep 3
    sudo echo -e "Can not Find DNS, Please Check Internet Connection." '\x1A' >
/dev/$termal
fi

```

```

sendSMS.sh
#!/bin/bash
termal=$1
phone_=$2
text=$3
sudo stty 19200 cs8 -raw -parenb -cstopb crtscts -echo -F /dev/$termal
sudo echo -e AT+CMGS="\'+$phone\"'\x0D' > /dev/$termal
sleep 3
sudo echo -e "$text" '\x1A' > /dev/$termal

```

```

autoAlertsendSMS.sh
#!/bin/bash
phone="66865525351"
termal="ttyUSB0"
TestIP=$1
SECOND=$2
GAIN=$3
COMMAND=`ping -c 5 -i $SECOND $TestIP | grep packet | awk '{ print $6}'|cut -d'%' -f1`
if [ $COMMAND -eq 100 ]

```

```

then
    sudo stty 19200 cs8 -raw -parenb -cstopb crtscts -echo -F /dev/$terminal
    sudo echo -e AT+CMGS="\+$phone"\x0D' > /dev/$terminal
    echo " Internet Connection is DOWN $COMMAND % "
    sleep 3
    sudo echo -e "Internet Connection is DOWN" \x1A' > /dev/$terminal
elif [ $COMMAND -gt $GAIN ]
then
    sudo stty 19200 cs8 -raw -parenb -cstopb crtscts -echo -F /dev/$terminal
    sudo echo -e AT+CMGS="\+$phone"\x0D' > /dev/$terminal
    echo " Network Too Lost: $COMMAND % "
    sleep 3
    sudo echo -e "Internet TOO LOST $COMMAND %" \x1A' > /dev/$terminal
elif [ $COMMAND -eq 0 ]
then
    echo " Normal Service"
else
    sudo stty 19200 cs8 -raw -parenb -cstopb crtscts -echo -F /dev/$terminal
    sudo echo -e AT+CMGS="\+$phone"\x0D' > /dev/$terminal
    echo " Can't find DNS, or Check Internet Connection !!"
    sleep 3
    sudo echo -e "Can not Find DNS, Please Check Internet Connection." \x1A' >
/dev/$terminal
fi

```