

ภาคผนวก ก. รหัสคำสั่งสำหรับถ่ายโอนข้อมูลด้วย FTP

กระบวนการเก็บข้อมูลการใช้งานเครือข่ายได้ใช้รหัสคำสั่ง shell script บนเครื่องที่ใช้บันทึกปริมาณข้อมูลที่เกิดขึ้นบนเราเตอร์มหาวิทยาลัยยล (gemini) เพื่อถ่ายโอนเพิ่มข้อมูลชื่อ gemini.psu.ac.th_202.12.74.2.log ที่เป็นข้อมูลของปริมาณข้อมูลที่ผ่านเข้าออกเราเตอร์ดังกล่าวมายังเครื่องคอมพิวเตอร์ปลายทางของผู้วิจัยโดยใช้วิธีการตั้งค่าเวลาให้มีการทำงานอัตโนมัติด้วย crontab เพื่อให้เรียกใช้รหัสคำสั่งนี้เป็นประจำทุกวัน โดยรหัสคำสั่งเป็นดังนี้

```
#set file name for copy
day = `date '+ %Y%m%d%H%M'`
sname = "(path name)/gemini.psu.ac.th_202.12.74.2.log"
dname = "(path name)/gemini202log$day.txt"

#copy file: gemini.psu.ac.th_202.12.74.2.log
cp $sname $dname

#ftp to remote host (Destination IP address)
HOST = '(destination IP address)'
USER = '(user name)'
PASSWD = '(password)'

ftp -n $HOST << SCRIPT
    quote USER $USER
    quote PASS $PASSWD
    put $dname
    quit
SCRIPT

rm -f $dname

exit 0
```

- เมื่อ
- Path name – เป็นตำแหน่งที่เก็บเพิ่มข้อมูลที่ต้องการ
 - User name – เป็นชื่อบัญชีผู้ใช้ที่เครื่องคอมพิวเตอร์ปลายทาง
 - Password – เป็นรหัสผ่านของบัญชีผู้ใช้
 - Destination IP address – หมายเลขไอพีเครื่องปลายทาง

ภาคผนวก ข. Traffic Tools (ttools)

Traffic Tools เป็นเครื่องมือที่ได้มีการใช้งานบ่อยครั้งในงานวิจัยนี้ มีหน้าที่การทำงานหลากหลายทั้งการเก็บบันทึกข้อมูลแพ็กเก็ตในเครือข่าย ตรวจสอบปริมาณข้อมูลแยกตามระดับความสำคัญ แปลงแพ็กเก็ตข้อมูลที่ได้บันทึกไว้ในรูปแบบไบนารีมาเป็นตัวอักษร และจำลองส่งแพ็กเก็ตตามข้อมูลที่ได้บันทึกไว้ โดยแยกอธิบายแต่ละโหมดดังนี้

1. Traffic Capture [ttools: Capture mode]

โหมดมีหน้าที่เก็บ และบันทึกแพ็กเก็ตข้อมูลในเครือข่าย โดยการบันทึกได้เลือกเฉพาะข้อมูลใน 86 ไบต์แรกของ IP datagram เพื่อให้ครอบคลุมข้อมูลส่วนหัวของแพ็กเก็ตที่จำเป็นต่อการใช้งานเท่านั้น และประหยัดพื้นที่ในการจัดเก็บข้อมูล โดยข้อมูล 86 ไบต์ประกอบด้วย

- ส่วนหัวของเฟรมอีเทอร์เน็ต 14 ไบต์ ประกอบด้วย
 - Destination MAC address
 - Source MAC address
 - Type ; เมื่อมีค่าเท่ากับ 0x0800 แสดงว่าเป็น IP datagram
- ส่วนหัวของ IP datagram 20 – 60 ไบต์ ตามข้อมูลใน Header length ประกอบด้วย

● Version	● Time of live
● Header length (5-15)	● Protocol
● Type of service	● Header Checksum
● Total length	● Source IP address
● Identification	● Destination IP address
● Flag	● Option (ตาม Header length)
● Fragment offset	
- ส่วนหัวของ TCP หรือ UDP 12 ไบต์
 - TCP/UDP source port number
 - TCP/UDP destination port number
 - Sequence number (เฉพาะ TCP เท่านั้น)
 - Acknowledgement number (เฉพาะ TCP เท่านั้น)

คำสั่งเรียกใช้งาน

```
# ttools c interface filename.raw [file size]
```

เมื่อ

c – ชื่อโหนดการทำงาน

interface – ชื่อในการอ้างอิงการ์ดเครือข่ายที่ต้องการเก็บข้อมูลแพ็กเก็ต เช่น eth0 หรือ eth1 ในระบบปฏิบัติการ Linux Redhat

filename.raw – ชื่อเพิ่มข้อมูลสำหรับจัดเก็บข้อมูลแพ็กเก็ต

file size – เป็นขนาดของเพิ่มข้อมูลสูงสุดที่ใช้ในการเก็บข้อมูลในหน่วย MB หากไม่กำหนดจะมีค่าเป็น 600

หลังจากเรียกใช้คำสั่งจะพบการทำงานของโปรแกรมดังรูปที่ ข.1

ผลลัพธ์

- filename.raw → เก็บข้อมูลของแพ็กเก็ตในรูปแบบของไบนารี
- filename.raw.dly → เก็บเวลาระหว่างแพ็กเก็ตในหน่วยมิลลิวินาที (ms)
- filename.raw.log → เก็บผลสรุปการทำงานของโปรแกรม ดังนี้
 - จำนวนแพ็กเก็ตที่เป็น IP datagram
 - อัตราการไหลเวียนของ IP datagram ในหน่วยแพ็กเก็ตต่อวินาที (packet/second)
 - ปริมาณข้อมูลของ IP datagram ที่พบในหน่วย Mbps
 - จำนวนแพ็กเก็ตที่ไม่ใช่ IP datagram
 - อัตราการไหลเวียนของแพ็กเก็ตที่ไม่ใช่ IP datagram
 - ปริมาณข้อมูลของแพ็กเก็ตที่ไม่ใช่ IP datagram
 - เวลาที่เริ่มบันทึกข้อมูล
 - เวลาที่หยุดบันทึกข้อมูล
 - เวลาที่ใช้ในการบันทึกข้อมูลในหน่วยวินาที

```

2:172.28.80.211 - default - SSH Secure Shell
File Edit View Window Help
Linux Network Traffic Capturing By: Nontapon Rattanapittayapor
Start capture: Wed Nov 3 10:12:18 2004
Running time: 0 day(s) 00 hour(s) 00 min(s) and 55 sec(s) [00055]
Capture from: eth0 Refresh: 5 second(s)
Capture file size = 8/600 MB

+-----+-----+-----+
|                               | IP Datagram | Other Frame |
+-----+-----+-----+
| Counters :                    | 178666     | 0           | packet(s)  |
| Rate      :                    | 3309       | 0           | packet(s)/s|
| Bytes    :                    | 134.207993 | 0.000000   | MB         |
| Throughput:                   | 19.882666  | 0.000000   | MBps      |
+-----+-----+-----+

Output files
- Raw traffic (86 bytes): /tap/test.raw
- Time between packet : /tap/test.raw.dly
- Summary : /tap/test.raw.log

Exit program, pressed 'q'

การรายงานผลเมื่อสิ้นสุดการทำงานของโปรแกรม
Capture time: 0 day(s) 00 hour(s) 00 min(s) and 54 sec(s) 501 ms (54.501 s)
+-----+-----+-----+
|                               | IP-Datagram | Other-Frame |
+-----+-----+-----+
| Counters                    | 180017     | 0           | packet(s)  |
| Rate                        | 3273       | 0           | packet(s)/s|
| MBytes                      | 135.233566 | 0.000000   | MB         |
| Throughput                   | 19.670338  | 0.000000   | MBps      |
+-----+-----+-----+

[root@TMG traffic_tools]#
Connected to 172.28.80.211 SSH2 - aes128-cbc - hmac-md5 - none 78x35

```

รูปที่ ข.1 แสดงหน้าต่างโปรแกรม ttools ในการเก็บข้อมูลเครือข่าย

2. Traffic Monitoring [ttools: Monitor mode]

โหมดการทำงานนี้ผู้ดูแลระบบสามารถดูแพ็กเก็ตที่เข้าออกเครือข่าย โดยแยกตามกลุ่มหรือระดับความสำคัญของแพ็กเก็ตตามเงื่อนไขที่ได้กำหนดไว้ และบันทึกสถิติเกี่ยวกับปริมาณของข้อมูลในแต่ละกลุ่มทุกช่วงเวลาที่กำหนด ซึ่งการจัดกลุ่มแพ็กเก็ตใช้รหัสโปรแกรมเดียวกับเครื่องมือจัดลำดับนำส่งข้อมูล (Traffic Management, TMg) ที่จะเสนอในภาคผนวก ง.

คำสั่งเรียกใช้งาน

```
# ttools m interface policy [refresh]
```

เมื่อ

m – ชื่อ โหมดการทำงาน

interface – ชื่อในการอ้างอิงการ์ดเครือข่ายที่ต้องการติดตาม

policy – ชื่อเพิ่มข้อมูลที่มีข้อมูลของกฎที่ใช้ในการจัดกลุ่มของแพ็กเก็ต

refresh – ช่วงเวลาที่โปรแกรมใช้ในการติดตาม ก่อนที่จะแสดงข้อมูลทางจอภาพ และบันทึกเป็นค่าสถิติในแฟ้มข้อมูล มีหน่วยเป็นวินาที ในกรณีที่ไม่วัดค่าตัวแปรจะมีค่าเท่ากับ 5 วินาที

หลังจากเรียกใช้คำสั่งจะพบการทำงานของโปรแกรมดังที่ได้แสดงในรูปที่ ข.2

ผลลัพธ์

- bw.csv → เก็บปริมาณข้อมูลการใช้งานแบนด์วิดธ์ของแพ็กเก็ตแยกตามกลุ่ม หรือระดับความสำคัญทุก ๆ ช่วงเวลาวินาทีที่กำหนด
- log.csv → เก็บข้อมูลบางส่วนของแพ็กเก็ตในรูปแบบตัวอักษรที่สามารถอ่านได้ สำหรับข้อมูลที่เก็บมีโครงสร้างดังนี้
 - เวลาที่พบแพ็กเก็ตข้อมูลในรูปแบบของมิลลิวินาที
 - หมายเลขกลุ่ม หรือระดับความสำคัญของแพ็กเก็ต
 - ขนาดแพ็กเก็ต มีหน่วยเป็นไบต์
 - หมายเลขไอพีต้นทาง
 - หมายเลขไอพีปลายทาง
 - โพรโทคอลระดับชั้นนำส่งที่ใช้ (TCP/UDP)
 - หมายเลขพอร์ตต้นทาง
 - หมายเลขพอร์ตปลายทาง

```

2:172.28.80.211 - default - SSH Secure Shell
[root@TNG traffic_tools]# ./ttools m eth0 policy.cfg
Monitor Mode: eth0 policy.cfg
Number of policy: 8
Format: [Source IP] [Dest IP] [Source or Dest Port][Priority/Class]
policy# 0 - [ Any ] [ Any ] [ 0 ] [3]
policy# 1 - [ Any ] [ Any ] [ 0 ] [3]
policy# 2 - [ Any ] [ Any ] [ 80 ] [1]
policy# 3 - [ Any ] [ Any ] [ 8080 ] [1]
policy# 4 - [ Any ] [ Any ] [ 20 ] [2]
policy# 5 - [ Any ] [ Any ] [ 21 ] [2]
policy# 6 - [ Any ] [ Any ] [ 40 ] [4]
policy# 7 - [ Any ] [ Any ] [ 50 ] [5]

-----
time : 5
CLASS[1] - throughput: 3.716577 Mbps
CLASS[2] - throughput: 3.716849 Mbps
CLASS[3] - throughput: 3.717786 Mbps

time : 10
CLASS[1] - throughput: 3.999506 Mbps
CLASS[2] - throughput: 4.000322 Mbps
CLASS[3] - throughput: 4.000223 Mbps
  
```

รูปที่ ข.2 แสดงหน้าต่างโปรแกรม ttools ในการเฝ้ามองปริมาณข้อมูลบนเครือข่าย

3. Packet Converter [ttools: ConVert mode]

การบันทึกข้อมูลใน Capture mode โปรแกรมได้จับเก็บข้อมูลในรูปของไบนารี ทำให้ไม่สามารถอ่านข้อมูลได้โดยตรง โหมดนี้จึงมีหน้าที่ในการแปลงข้อมูลแพ็กเก็ตในแฟ้ม filename.raw ให้อยู่ในรูปแบบที่สามารถอ่านเข้าใจได้

คำสั่งเรียกใช้งาน

```
# ttools v filename.raw output-file.cvt
```

เมื่อ

v – ชื่อโหมดการทำงาน

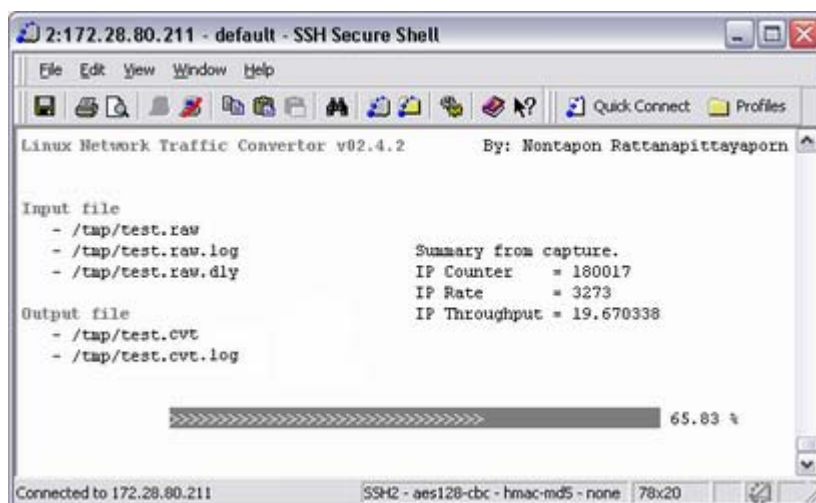
filename.raw – แฟ้มข้อมูลที่ได้จาก Capture mode

output-file.cvt – ชื่อไฟล์ผลลัพธ์ที่ได้หลังจากการแปลงข้อมูล

หลังจากเรียกใช้คำสั่งจะพบการทำงานของโปรแกรมดังที่ได้แสดงในรูปที่ ข.3

ผลลัพธ์

- output-file.out → เก็บข้อมูลบางส่วนของไอพีแพ็กเก็ต ประกอบด้วย
 - ลำดับของแพ็กเก็ต
 - วินาทีที่แพ็กเก็ตมาถึง
 - โปรโตคอลในชั้นนำส่งที่ใช้
 - ขนาดส่วนหัวของ IP datagram
 - ขนาดของแพ็กเก็ต
 - หมายเลขประจำตัวแพ็กเก็ต
 - หมายเลขไอพีต้นทาง
 - หมายเลขไอพีปลายทาง
 - หมายเลขพอร์ตต้นทาง
 - หมายเลขพอร์ตปลายทาง
 - นอกจากนี้หากโปรโตคอลชั้นนำส่งเป็น TCP จะบันทึกหมายเลขลำดับ และหมายเลขตอบกลับด้วย
- output-file.cvt.log → ข้อมูลสรุปที่ได้จากการแปล
 - จำนวน IP datagram ทั้งหมด แบ่งเป็น TCP UDP และอื่น ๆ
 - ปริมาณไบนารีของ IP datagram แบ่งเป็น TCP UDP และอื่น ๆ



รูปที่ ข.3 แสดงหน้าต่างโปรแกรม ttools ในขณะที่แปลงข้อมูลแพ็กเก็ต

4. เป็น Traffic Generator [ttools: Genrate mode]

โหมดการทำงานนี้มีหน้าที่จำลองแพ็กเก็ตเข้าสู่เครือข่ายตามพฤติกรรมที่ได้บันทึกไว้ โดยใช้ข้อมูลจากไฟล์ filename.raw filename.raw.dly และ filename.raw.log เครื่องมือนี้สามารถหนดวงเวลานำส่งแพ็กเก็ตตามระยะห่างของเวลาที่แพ็กเก็ตมาถึงได้เมื่อมี filename.raw.dly อยู่ในตำแหน่งเดียวกับ filename.raw แต่หากไม่มีไฟล์ดังกล่าวเครื่องมือนี้จะส่งแพ็กเก็ตออกทันที โดยไม่หนดวงเวลาไว้

คำสั่งเรียกใช้งาน

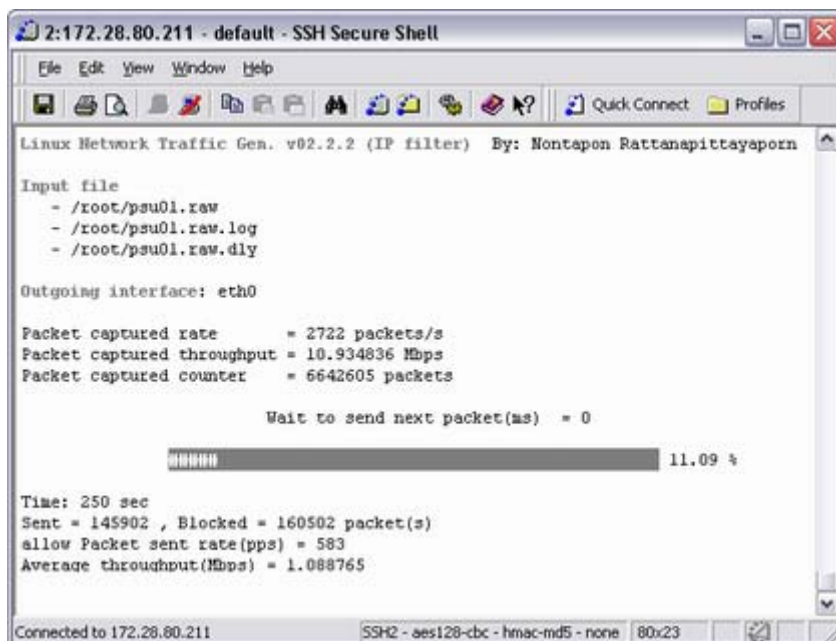
```
# ttools g interface filename.raw
เมื่อ
g - ชื่อโหมดการทำงาน
interface - ชื่อในการอ้างอิงการ์ดเครือข่ายที่ใช้ในการจำลอง
filename.raw - เพิ่มข้อมูลที่ได้จาก Capture mode
```

หลังจากเรียกใช้คำสั่งจะพบการทำงานของโปรแกรมดังที่ได้แสดงในรูปที่ ข.4

หมายเหตุ

การทำงานในโหมดจำลองส่งแพ็กเก็ตจากไฟล์ข้อมูลที่บันทึกไว้ได้เพิ่มคำสั่งให้มีกรองแพ็กเก็ตด้วย เนื่องจากต้องการจำลองส่งเฉพาะแพ็กเก็ตที่ออกจากเครือข่ายมหาวิทยาลัย-สงขลานครินทร์เท่านั้น จึงต้องกรองแพ็กเก็ตที่มาจากภายนอกทิ้งไป ดังนั้นข้อมูลของไฟล์ที่จะนำไปจำลองส่ง

แพ็กเก็ตต้องมีหมายเลขไอพีต้นทางอยู่ในกลุ่มต่อไปนี้ 192.100.77.0/24 202.12.73.0/24 202.12.74.0/24 และ 202.28.96.0/22 หากต้องการแก้ไขข้อกำหนดเกี่ยวกับการกรองหมายเลขไอพีให้แก้ไขรหัสคำสั่งในไฟล์ allow_ip.h แล้วคอมไพล์โปรแกรมใหม่อีกครั้ง



```

2:172.28.80.211 - default - SSH Secure Shell
File Edit View Window Help
Linux Network Traffic Gen. v02.2.2 (IP filter) By: Montapon Rattanapittayaporn
Input file
- /root/psu01.raw
- /root/psu01.raw.log
- /root/psu01.raw.dly

Outgoing interface: eth0

Packet captured rate      = 2722 packets/s
Packet captured throughput = 10.934836 Mbps
Packet captured counter   = 6642605 packets

                    Wait to send next packet(ms) = 0
                    ██████████ 11.09 %

Time: 250 sec
Sent = 145902 , Blocked = 160502 packet(s)
allow Packet sent rate(pps) = 583
Average throughput(Mbps) = 1.088765

Connected to 172.28.80.211    SSH2 - aes128-cbc - hmac-md5 - none    80x23

```

รูปที่ ข.4 แสดงหน้าต่างโปรแกรม ttools ขณะจำลองส่งแพ็กเก็ตตามข้อมูลประวัติที่เก็บไว้

ภาคผนวก ก. Packet Generator (pgen)

Pgen เป็นเครื่องมือจำลองแพ็กเก็ตอีกตัวหนึ่งที่ได้พัฒนาขึ้น โดยมีเป้าหมายให้ผู้สร้างแพ็กเก็ตจำลองที่มีขนาดอยู่ในช่วง 60 - 1514 ไบต์ ซึ่งผู้ใช้สามารถกำหนดขนาดต่ำสุด และค่าสูงสุดของแพ็กเก็ตที่ต้องการจำลองได้ นอกจากนี้ผู้ใช้ยังสามารถกำหนดหมายเลขพอร์ตปลายทางที่ต้องการได้ และยังสามารถกำหนดความเร็วในการนำส่งแพ็กเก็ต (Mbps) ที่ต้องการได้ด้วย

คำสั่งเรียกใช้งาน

```
# pgen interface dest.port min.pkt.size max.pkt.size max.throughput
```

เมื่อ

interface – ชื่อในการอ้างอิงการ์ดเครือข่ายที่ใช้ในการจำลอง

dest.port – หมายเลขพอร์ตปลายทางที่ต้องการจำลอง

min.pkt.size – ขนาดเล็กสุดของแพ็กเก็ต ต้องมีค่ามากกว่าหรือเท่ากับ 60 ไบต์

max.pkt.size – ขนาดใหญ่สุดของแพ็กเก็ต ต้องมีค่าน้อยกว่าหรือเท่ากับ 1514 ไบต์

max.throughput – อัตราการนำส่งแพ็กเก็ตในหน่วย Mbps

หลังจากเรียกใช้คำสั่งจะพบการทำงานของโปรแกรมดังที่ได้แสดงในรูปที่ ก.1

```
Packet generator...

Interface : eth0
HWaddr >> 00:04:75:ED:86:1D
inet addr >> 10.28.80.10
Dest-Port# : 80 (17)
Packet size : 60 - 1514 bytes
Leaky bucket: 10.0000 Mbps

Dump 48 bytes of this packet
00000000 : ff ff ff ff ff ff 00 04 75 ed 86 1d 08 00 45 00 .....U.....E.
00000010 : 00 00 00 00 00 00 00 80 11 00 00 0a 1c 50 0a ff ff .....P...
00000020 : ff ff 00 00 00 50 00 00 00 00 00 00 00 00 00 .....P.....

Generate dest-port# 80, count: 8324 (10.001188 Mbps)
[ Last 5 sec. @ 1099462523 ]
```

รูปที่ ก.1 แสดงหน้าต่างโปรแกรม Packet Generator

ภาคผนวก ง. Traffic Manager (TMg)

เครื่องมือชิ้นนี้มีหน้าที่ในการจัดลำดับแพ็กเก็ตที่ผ่านเข้ามาก่อนนำส่งออกไปยังเครือข่ายอีกด้านหนึ่ง ประกอบด้วยวิธีการต่างกัน 3 แบบ คือ FIFO, PQ และ CBQ เมื่อเริ่มใช้งานเครื่องมือนี้จะอ่านข้อกำหนดการทำงานจากแฟ้มชื่อ `tm.cfg` ซึ่งมีตัวแปรควบคุมดังที่แสดงในตารางที่ ง.1 และอ่านนโยบายการจัดระดับความสำคัญ/กลุ่มจากไฟล์ `policy.cfg` ก่อน เพื่อใช้ในการทำงานของโปรแกรมต่อไป

ข้อกำหนด	คำอธิบาย
<code>token_rate_mbps:</code>	อัตราการนำส่งข้อมูลสูงสุด
<code>bucket_size_bytes:</code>	ขนาดของ bucket สำหรับรองรับข้อมูลแบบ burst
<code>refresh:</code>	ระยะเวลาในการแสดง และ จัดเก็บผลลัพธ์ มีหน่วยเป็นวินาที
<code>start_qdisc:</code>	วิธีการจัดคิวที่สามารถใช้ได้ คือ <code>fifo pq</code> หรือ <code>cbq</code> แบบใดแบบหนึ่ง
<code>bwlog:</code>	ต้องการให้มีการเก็บข้อมูลการใช้แบนด์วิดท์หรือไม่ (1=ใช่)
<code>bwlog_file:</code>	ชื่อแฟ้มข้อมูลสำหรับเก็บข้อมูลการใช้แบนด์วิดท์
<code>pktlog:</code>	ต้องการให้มีการเก็บข้อมูลแพ็กเก็ตหรือไม่ (1=ใช่)
<code>pktlog_file:</code>	ชื่อแฟ้มข้อมูลสำหรับเก็บข้อมูลแพ็กเก็ต
<code>q1_size:</code>	ขนาดคิวที่ 1
<code>q2_size:</code>	ขนาดคิวที่ 2
<code>q3_size:</code>	ขนาดคิวที่ 3
<code>cbq_parent:</code>	จำนวนแพ็กเก็ตมากที่สุดที่จะมีการนำส่งใน 1 รอบบริการ
<code>q1_cbq_allot:</code>	จำนวนแพ็กเก็ตกลุ่ม 1 มากที่สุดที่อาจถูกนำส่งใน 1 รอบคำนวณ
<code>q2_cbq_allot:</code>	จำนวนแพ็กเก็ตกลุ่ม 2 มากที่สุดที่อาจถูกนำส่งใน 1 รอบคำนวณ
<code>q3_cbq_allot:</code>	จำนวนแพ็กเก็ตกลุ่ม 3 มากที่สุดที่อาจถูกนำส่งใน 1 รอบคำนวณ
<code>q1_cbq_borrow:</code>	คิวที่ 1 มีสิทธิ์ยืมโควต้าการนำส่งส่วนที่เหลือได้หรือไม่ (1=ยืมได้)
<code>q2_cbq_borrow:</code>	คิวที่ 2 มีสิทธิ์ยืมโควต้าการนำส่งส่วนที่เหลือได้หรือไม่ (1=ยืมได้)
<code>q3_cbq_borrow:</code>	คิวที่ 3 มีสิทธิ์ยืมโควต้าการนำส่งส่วนที่เหลือได้หรือไม่ (1=ยืมได้)

ตารางที่ ง.1 แสดงข้อกำหนดของโปรแกรม TMg (`tm.cfg`)

- สำหรับข้อกำหนดนโยบายใน policy.cfg ได้แบ่งออกเป็น 4 ส่วนหลักดังนี้
- ส่วนที่ 1: (บรรทัดที่ 1) ใช้สำหรับกำหนดลักษณะการตรวจสอบหมายเลขพอร์ต เมื่อ Port มีค่าเป็น 0 ให้ตรวจสอบกับเลขพอร์ตต้นทางของแพ็กเก็ตเท่านั้น
- 1 ให้ตรวจสอบทั้งหมายเลขพอร์ตต้นทาง และปลายทางของแพ็กเก็ต
- 2 ให้ตรวจสอบกับเลขพอร์ตปลายทางของแพ็กเก็ตเท่านั้น
- ส่วนที่ 2: (บรรทัดที่ 2) ใช้กำหนดค่าปริยายเกี่ยวกับระดับความสำคัญ หรือกลุ่มของแพ็กเก็ตที่ไม่ใช่ IP datagram
- ส่วนที่ 3: (บรรทัดที่ 3) ใช้กำหนดค่าปริยายเกี่ยวกับระดับความสำคัญ หรือกลุ่มของแพ็กเก็ตที่เป็น IP datagram
- ส่วนที่ 4: (บรรทัดที่เหลือ) ใช้กำหนดค่าระดับความสำคัญ หรือกลุ่มของแพ็กเก็ตที่มีรายละเอียดตรงกับเงื่อนไข

โดยแต่ละเงื่อนไขที่ใช้ในการตรวจสอบแบ่งออกได้ 4 ส่วนดังนี้ คือ

<หมายเลขไอพีต้นทาง> <หมายเลขไอพีปลายทาง> <หมายเลขพอร์ต> <ระดับความสำคัญ>

เมื่อระดับความสำคัญ/กลุ่ม มีเพียง 3 ค่าตามจำนวนของคิวในระบบ คือ 1 2 และ 3 ตามรายละเอียดด้านล่าง หากกำหนดค่านี้ให้เป็น 0 แพ็กเก็ตนั้นจะถูกทิ้งทันที

ค่า 1 แทน มีความสำคัญระดับสูงสุด หรืออยู่กลุ่ม 1 จะได้รับบรรจุลงคิวที่ 1

ค่า 2 แทน มีความสำคัญระดับกลาง หรืออยู่กลุ่ม 2 จะได้รับบรรจุลงคิวที่ 2

ค่า 3 แทน มีความสำคัญระดับต่ำ หรืออยู่กลุ่ม 3 จะได้รับบรรจุลงคิวที่ 3

ตัวอย่างการกำหนดนโยบาย

Port 1	ให้ตรวจสอบทั้งหมายเลขพอร์ตต้นทาง และ พอร์ตปลายทาง
** 0 1	สำหรับแพ็กเก็ตที่ไม่ใช่ IP datagram กำหนดให้มีระดับความสำคัญสูงสุด
** 0 3	สำหรับไอพีแพ็กเก็ตที่ไม่ตรงตามเงื่อนไขด้านล่าง จะมีระดับความสำคัญต่ำ
** 80 2	แพ็กเก็ตที่มีหมายเลขพอร์ตเท่ากับ 80 มีความสำคัญระดับกลาง
172.28.80.1 * 0 1	แพ็กเก็ตที่มีหมายเลขไอพีต้นทาง 172.28.80.1 มีความสำคัญระดับสูงสุด
* 172.28.80.10 21 1	แพ็กเก็ตที่มีหมายเลขไอพีปลายทาง 172.28.80.10 และมีหมายเลขพอร์ต 21 ให้มีระดับความสำคัญสูงสุด

หมายเหตุ

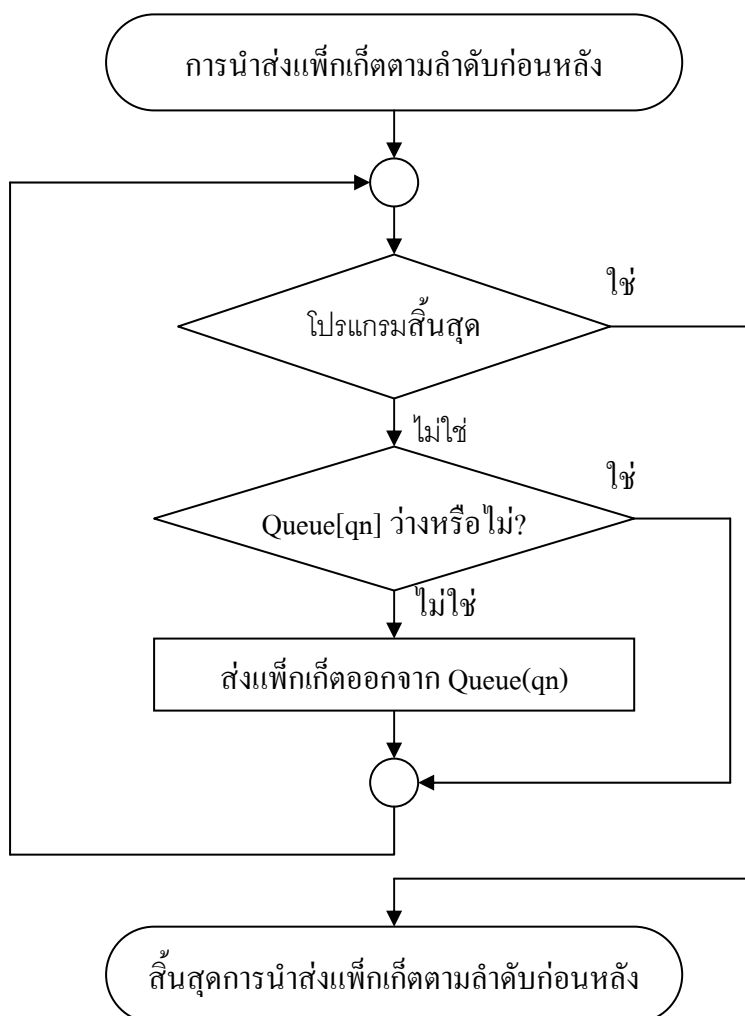
ในการตรวจสอบเงื่อนไขจะเป็นไปตามลำดับจากบนลงล่าง หากการตรวจสอบแพ็กเก็ตใดตรงกับเงื่อนไขที่พบแล้วจะไม่ตรวจสอบกับเงื่อนไขอื่นอีก

ส่วนต่อไปจะอธิบายถึงหลักการการทำงานของวิธีจัดคิวทั้ง 3 แบบที่ได้บรรจุอยู่ใน Traffic Manager (TMg) คือ FIFO, PQ และ CBQ โดยอาศัยแผนภาพ Flow-Chart ดังนี้

วิธีการจัดคิวแบบ FIFO

ค่าคงที่ $qn = 0$

; ตัวชี้หมายเลขคิว ในที่นี้มีเพียงคิวเดียว คือคิวหมายเลข 0



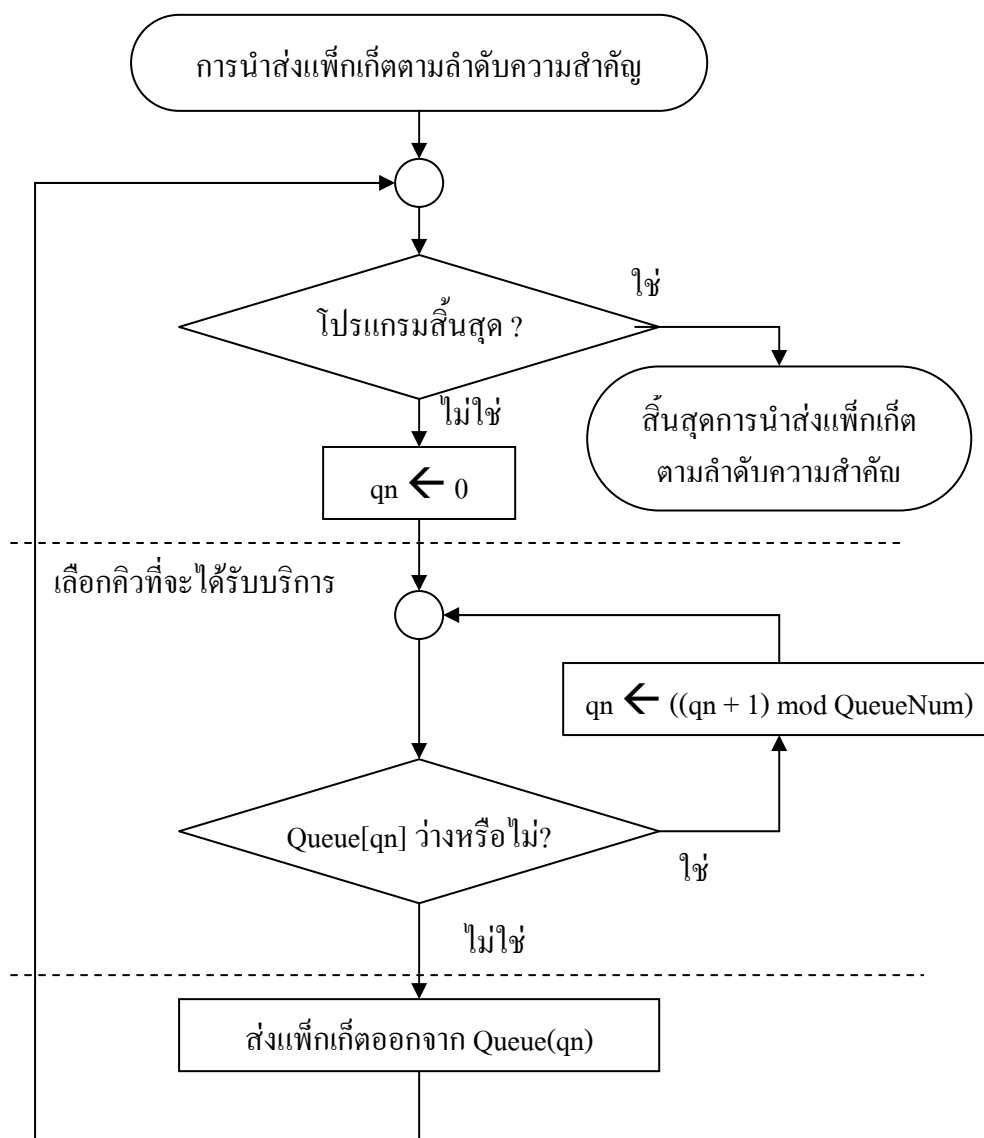
วิธีการจัดคิวแบบ PQ

ค่าคงที่ QueueNum = 3

; จำนวนคิวในระบบ

ตัวแปร qn

; ตัวชี้หมายเลขคิว มีค่าตั้งแต่ 0 ถึง 2



วิธีการจัดคิวแบบ CBO

ค่าคงที่	QueueNum = 3	; จำนวนคิว
ตัวแปร	premain	; จำนวนแฟ้มเกิดเหลือที่ยังสามารถให้บริการเพิ่มได้
	Parent	; จำนวนแฟ้มเกิดมากที่สุดที่จะให้บริการแต่ละรอบบริการ
	qcounter	; จำนวนแฟ้มเกิดที่อยู่ในคิวนั้น ๆ
	qremain	; จำนวนแฟ้มเกิดที่เหลืออยู่ในคิวนั้น ๆ
	qallot	; ค่ารับประกันการได้รับบริการของคิว
	qborrow	; สถานะการมีสิทธิ์ยืมหรือไม่
	qb	; สำหรับตรวจสอบว่าคิวนั้น ๆ จำเป็นต้องทำการยืมอีกหรือไม่
	bflag	; สถานะการตรวจสอบว่าเกิดการยืมหรือไม่
	service_q	; จำนวนแฟ้มเกิดที่คิวนั้น ๆ จะได้รับบริการในรอบบริการ
	bserv	; จำนวนแฟ้มเกิดที่คิวจะได้รับบริการจากการสอบถาม
	qn	; สำหรับเป็นจำนวนนับ
	sum	; จำนวนแฟ้มเกิดรวมทุกคิว
	first	; สำหรับตรวจสอบว่าเป็นการสอบถามครั้งแรกหรือไม่

