ชื่อ นาย รวีโรจน์ ทองดี **รหัสนักศึกษา** 62010763

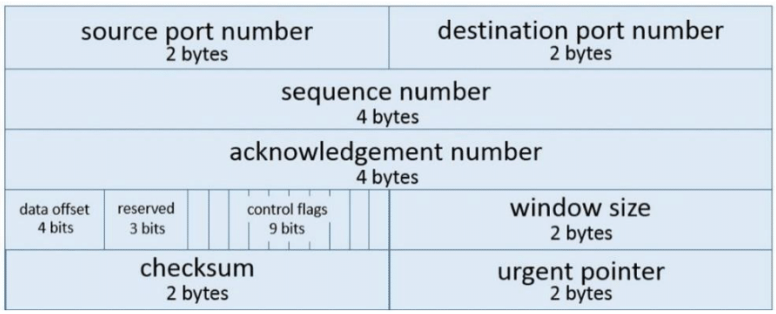
กิจกรรมที่ 6 : TCP Connection

กิจกรรมที่ 5 : TCP Connection

กิจกรรมครั้งนี้จะเป็นการทำความเข้าใจกับโปรโตคอล TCP (Transmission Control Protocol) ซึ่ง TCP มีคุณสมบัติในการทำงานอยู่ 5 ประการได้แก่

- Reliable, in-order delivery คือ การส่งไม่ผิดพลาดโดยข้อมูลมีการเรียงตามลำดับ
- Connection Oriented คือ ต้องมีการสร้างการเชื่อมต่อก่อน และมีการแลกเปลี่ยนข้อมูลควบคุม
- Flow Control ควบคุมการไหลของข้อมูลระหว่าง Process ทั้ง 2 ด้าน
- Congestion Control ควบคุมการไหลของข้อมูลผ่านอุปกรณ์เครือข่าย
- Full Duplex data สามารถส่งได้ทั้ง 2 ทาง ในการเชื่อมต่อเดียวกัน

Connection Setup

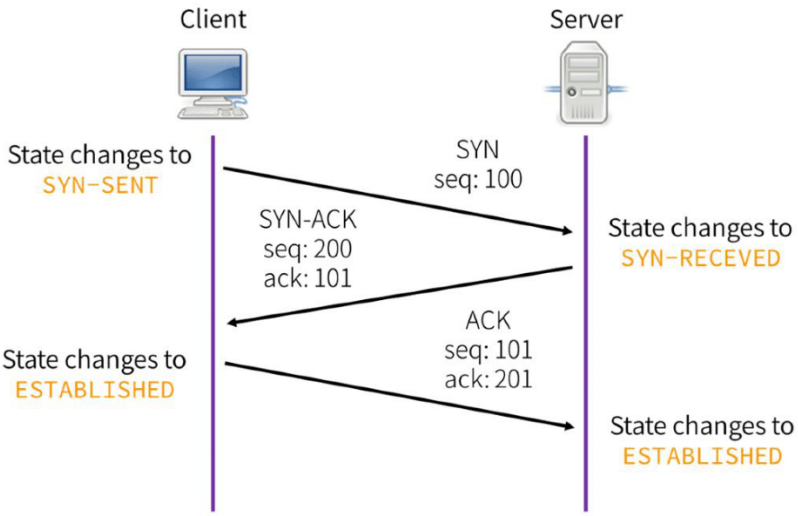


รูปแสดง TCP Header

ก่อนเริ่มการส่งข้อมูลทุกครั้งของ TCP จะต้องมีการสร้าง Connection ขึ้นมาก่อนโดย Client จะเริ่มสร้างการเชื่อมต่อไปที่ Server ซึ่งประกอบด้วย 3 ขั้นตอน

- Client การส่ง packet SYN ไปที่ Server โดย Client จะมีการสร้างหมายเลข Sequence Number เรียกว่า ISN : Initial Sequence Number ขึ้นมา (ในรูปสมมติว่า 100) ใส่ใน SEQ# แล้วส่ง
- เมื่อ Server ได้รับ packet SYN จะตอบกลับโดย packet SYN-ACK โดย Server จะมีการสร้างหมายเลข ISN ของตนเองขึ้นมาเช่นกัน โดยใส่ใน SEQ# และนำหมายเลข SN:Client+1 แล้วใส่ใน ACK# แล้วส่ง
- เมื่อ Client ได้รับ packet SYN-ACK ก็จะตอบกลับโดย packet ACK สุดท้าย โดย Client จะนำ SN:Client+1 ใส่ใน SEQ# และนำ SN:Server+1 ใส่ใน ACK# แล้วส่ง เมื่อถึงตรงนี้จะถือว่าฝั่ง Client สร้างการเชื่อมต่อสำเร็จแล้ว ซึ่ง Client สามารถจะเริ่มส่งข้อมูลได้
- เมื่อ Server ได้รับ packet ACK สุดท้าย จะถือว่าฝั่ง Server สร้างการเชื่อมต่อสำเร็จแล้วเช่นกัน

ชื่อ นาย รวีโรจน์ ทองดี **รหัสนักศึกษา** 62010763



1. ให้เปิดไฟล์ http-browse101d.pcapng ค้นหา 3 way handshake แรกในไฟล์แล้ว บันทึกข้อมูลลงในตารางด้านล่าง (ทั้ง Seq# และ Ack# ให้ใช้แบบ raw ในช่อง Flag ให้บอกว่ามี Flag ใดที่ Set บ้าง

SYN

Src Port : 61598	Dest Port : 80
Seq # : 610997682 (raw)	
Ack # : 0 (raw)	
Flags : 0x002 (SYN)	[TCP Flags: ....1. = Syn : Set]

SYN-ACK

Src Port : 80	Dest Port : 61598
Seq # : 4134094401 (raw)	
Ack # : 610997683 (raw)	
Flags : 0x012 (SYN , ACK)	[TCP Flags: ....1 ..1. = Acknowledgment : Set , Syn : Set]

ACK

Src Port : 61598	Dest Port : 80
Seq # : 610997683 (raw)	
Ack # : 4134094402 (raw)	
Flags : 0x012 (SYN , ACK)	[TCP Flags: ....1 .... = Acknowledgment : Set]

- ค่าความยาวข้อมูลของ packet ทั้ง 3 เท่ากับเท่าไรบ้าง 66 , 66 , 54
- ใน packet SYN มีข้อมูลอื่นๆ ส่งมาด้วยหรือไม่ อะไรบ้าง (ดูในคอลัมน์ info) และข้อมูลต่างๆ เหล่านั้นมีความหมายอะไรหรือนำไปใช้อะไร (ให้ค้นหาข้อมูลเพิ่มเติมจากหนังสือ)

ชื่อ นาย รวีโรจน์ ทองดี **รหัสนักศึกษา** 62010763

ข้อมูล	ความหมาย
Len = 0	TCP Segment Len : 0 (ความยาวของ segment)
MSS = 1460	Maximum segment size: 1460 bytes
WS = 4	Window scale: 2 (multiply by 4)
SACK_PERM = 1	Selective acknowledgment permitted

- ใน packet SYN-ACK มีข้อมูลอื่นๆ ส่งมาด้วยหรือไม่ อะไรบ้าง (ดูในคอลัมน์ info) และข้อมูลต่างๆ เหล่านั้นมีความหมายอะไรหรือนำไปใช้อะไร

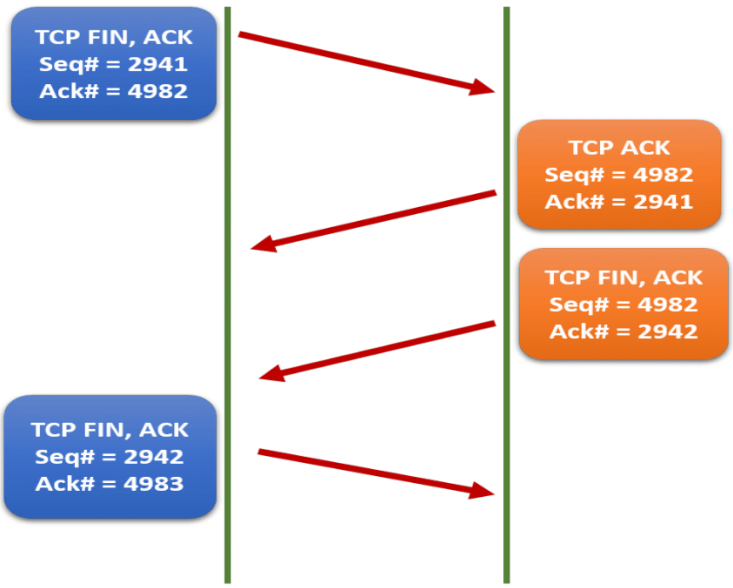
ข้อมูล	ความหมาย
Len = 0	TCP Segment Len : 0 (ความยาวของ segment)
MSS = 1430	Maximum segment size: 1430 bytes
WS = 64	Window scale: 6 (multiply by 64)
SACK_PERM = 1	Selective acknowledgment permitted

- ให้ดู packet ที่ส่งข้อมูล packet แรก (หรือ packet อื่นก็ได้) ให้ตอบว่าในข้อมูลที่ไม่เท่ากันของ Client กับ Server ในการเลือกใช้ข้อมูลหนึ่ง (เนื่องจากทั้ง 2 ด้านต้องใช้พารามิเตอร์เดียวกันในการส่งข้อมูล) คิดว่ามีหลักในการเลือกอย่างไร

ความสามารถในการรับส่งของฝั่งไม่เท่ากันจึงคิดว่าน่าจะเลือก parameter จากฝั่งที่มีข้อมูลน้อยกว่า เพื่อประสิทธิภาพและความเร็วในการสื่อสาร

Connection Terminated

เมื่อสิ้นสุดการส่งข้อมูลแล้ว ใน TCP จะมีการปิด Connection ซึ่งประกอบด้วย 4 ขั้นตอน



- ฝ่ายใดฝ่ายหนึ่งที่ต้องการปิด Connection (ต่อไปจะเรียก A และเรียกอีกฝั่งว่า B) จะส่ง packet ที่มี FIN/ACK flag มา โดยใช้ SEQ# และ ACK# เท่ากับ packet สุดท้ายก่อนจะปิด connection
- ฝั่ง B จะตอบด้วย packet ที่มี ACK flag โดยใช้ SEQ# เท่ากับ ACK# ของ FIN/ACK ก่อนหน้า และใช้ ACK# เท่ากับของ SYN# ของ packet ล่าสุด โดยเมื่อ A ได้รับ packet นี้ จะถือว่าเป็นการสิ้นสุด connection ของฝั่ง A (**หมายเหตุ** บางครั้งอาจไม่มีการส่ง packet นี้ โดยอาจรวมไปกับ packet ที่ 3
- ฝั่ง B จะเริ่มปิด Connection บ้าง โดยจะส่ง packet ที่มี FIN/ACK flag โดยใช้ SEQ# เท่ากับ ACK# ของ FIN/ACK ก่อนหน้า และใช้ ACK# เท่ากับของ SYN# ของ packet ล่าสุด +1
- ฝั่ง A จะตอบกลับการปิด Connection โดยจะส่ง packet ที่มี FIN/ACK flag โดยใช้ SEQ# เท่ากับ ACK# ของ FIN/ACK ก่อนหน้า และใช้ ACK# เท่ากับของ SYN# ของ packet ล่าสุด +1 เมื่อถึงจุดนี้จะถือว่าเป็นการสิ้นสุด Connection ของ B

2. ให้หา Packet ที่ปิด Connection ของ Connection ในข้อ 1 โดยให้บอกขั้นตอนการหาและป้อนรายละเอียดลงในตาราง (ข้อมูล Seq# และ Ack # ให้ใช้แบบ Relative)

Packet#	1663		
Src Port :	61598	Dest Port :	80
Seq # :	323		
Ack # :	1127		
Flags :	0x011 (FIN , ACK)	[TCP Flags: ....1 ..1. = Acknowledgment : Set , FIN : Set]	

Packet#	1664		
Src Port :	80	Dest Port :	61598
Seq # :	1127		
Ack # :	324		
Flags :	0x011 (FIN , ACK)	[TCP Flags: ....1 ..1. = Acknowledgment : Set , FIN : Set]	

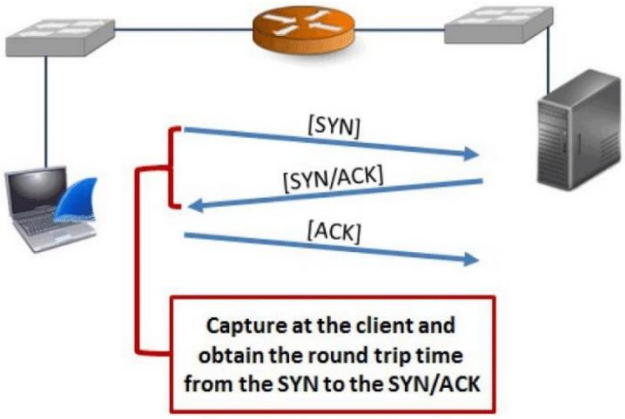
Packet#	1665		
Src Port :	61598	Dest Port :	80
Seq # :	324		
Ack # :	1128		
Flags :	0x010 (ACK)	[TCP Flags: ....1 .... = Acknowledgment : Set]	

วิธีค้นหา

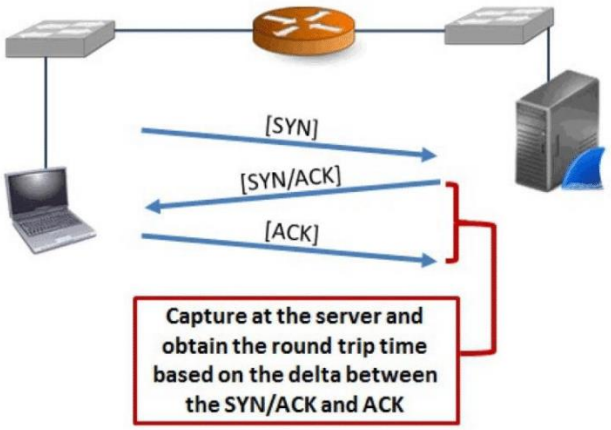
การค้นหา Packet ที่รูปแบบเป็น close connection หรือสังเกตจาก packet ที่มี FIN หรือ FIN/ACK และจากข้อที่ 1 เราทราบแล้วว่าเป็นการเชื่อมต่อที่ port เท่าใด จึงสามารถใช้งาน filter คำสั่ง tcp.src == 61595 ได้ ส่วน Packet ต่อๆ มาก็จะเป็น Packet ที่ติดๆ กัน

ชื่อ นาย รวีโรจน์ ทองดี **รหัสนักศึกษา** 62010763

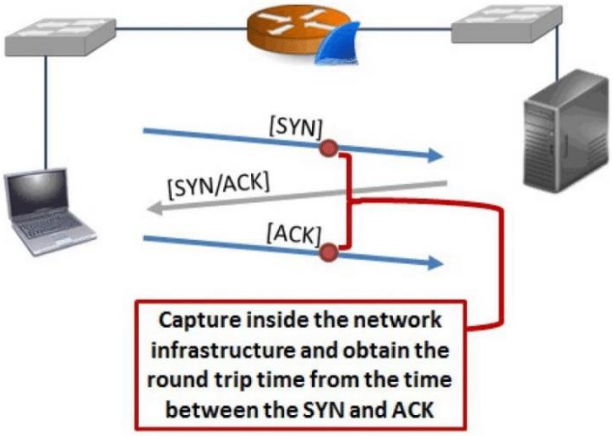
3. ใน Wireshark เราสามารถจะหา packet ที่มีคุณลักษณะของ flags เฉพาะได้ โดยใช้ display filter tcp.flags เช่น **tcp.flags.syn==1** หรือ **tcp.flags.ack==1** ซึ่งเราสามารถใช้หา RTT ของ TCP handshake ได้ โดยการหา RTT ของ TCP handshake มี 3 แบบ คือ วัดจากฝั่ง Client จะใช้เวลาระหว่าง SYN และ SYN-ACK



และวัดจากฝั่ง Server จะใช้เวลาระหว่าง SYN/ACK กับ ACK

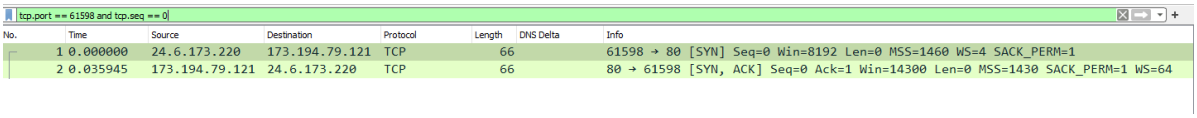


แต่ในกรณีที่วัดจากอุปกรณ์ ควรใช้ระหว่าง SYN และ ACK ตามรูป

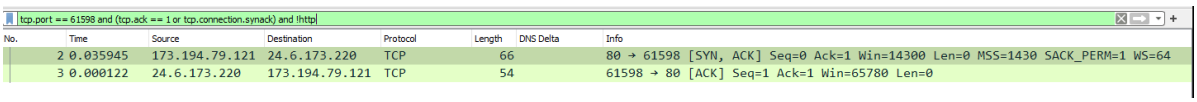


4. จากไฟล์ http-browse101d.pcapng ให้สร้าง display filter ที่สามารถแสดงเฉพาะ packet ต่อไปนี้ โดยไม่มี packet อื่นๆ มาปน (นักศึกษาพยายามคิดด้วยตนเอง)
   - packet SYN และ SYN/ACK ของ 3 way handshake (packet ที่ 1 และ 2)
   - packet SYN/ACK และ ACK ของ 3 way handshake (packet ที่ 2 และ 3)
   - packet SYN และ ACK 3 way handshake (packet ที่ 1 และ 3)

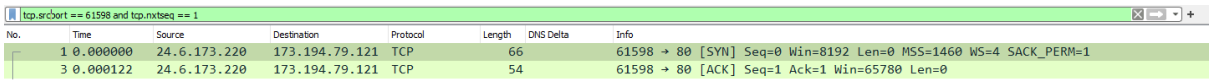
Packet SYN และ SYN/ACK ของ 3-way handshake



Packet SYN/ACK และ ACK ของ 3-way handshake



Packet SYN และ ACK ของ 3-way handshake



5. เราสามารถใช้ค่า RTT ของ TCP handshaking ตามข้อ 4 มาใช้วัดประสิทธิภาพของ Web Server ได้เช่นกัน โดย Server ที่มีค่า RTT น้อย แสดงถึงการตอบสนองที่รวดเร็ว ดังนั้นให้ capture ข้อมูลจากเว็บและใช้ display filter ตามข้อ 4 (ให้นักศึกษาเลือกใช้ตัวที่เหมาะสม) เพื่อหาค่า RTT ของเว็บต่างๆ จำนวน 3 เว็บ แล้วนำค่ามาใส่ตาราง

URL	เวลา
www.kmitl.ac.th	0.074040 ms
www.raweeroj.me	0.150283 ms
www.raweeroj.me	0.059157 ms

   - ให้ตอบว่าระหว่าง RTT ที่วัดในครั้งนี้ กับ HTTP RTT ที่วัดในครั้งก่อนหน้านี้ บอกถึงอะไร และแตกต่างกันอย่างไร

     RTT ที่เกิดขึ้นในชั้นของ Transport Layer นั้นจะจากการที่เครื่อง Client สร้าง Connection ไปยัง Server ปลายทาง มีการเชื่อมต่อแบบ Connection Oriented ซึ่งจะทำให้เห็นถึงการวัดประสิทธิภาพของเครือข่ายมากกว่า การวัด RTT HTTP ซึ่งเป็นการวัดประสิทธิภาพการตอบสนองของ http service (Web server) ซึ่งทำงานในชั้น Application Layer

งานครั้งที่ 6

- การส่งงาน ให้ส่งเป็นไฟล์ PDF จำนวน 1 ไฟล์ เท่านั้น
- ตั้งชื่อไฟล์โดยใช้รหัสนักศึกษา
- ส่วนบนของหน้าแรกให้มี รหัสนักศึกษา และ ชื่อนักศึกษา
- กำหนดส่ง ภายในวันที่ 21 กุมภาพันธ์ 2564