

กิจกรรมที่ 6 : TCP Connection

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

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

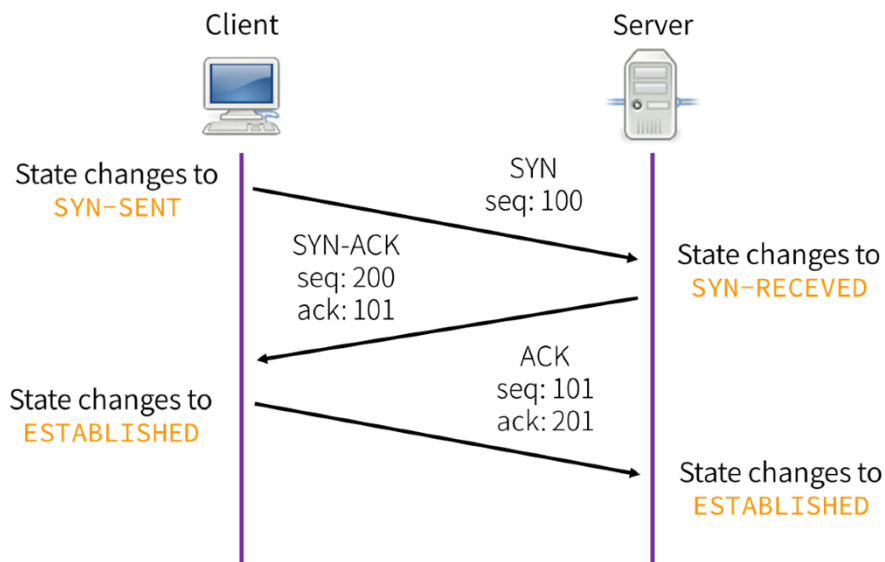
source port number 2 bytes				destination port number 2 bytes			
sequence number 4 bytes							
acknowledgement number 4 bytes							
data offset 4 bits		reserved 3 bits		control flags 9 bits		window size 2 bytes	
checksum 2 bytes				urgent pointer 2 bytes			

รูป/แสดง TCP Header

TCP Connection Setup (TCP 3-way Handshake)

ก่อนเริ่มการส่งข้อมูลทุกครั้งของ TCP จะต้องมีการสร้าง Connection ขึ้นมาก่อนโดย Client จะเริ่มสร้างการเชื่อมต่อไปที่ Server ซึ่งประกอบด้วยการรับส่ง TCP segment ระหว่าง Client-Server จำนวน 3 TCP segments

- Client ส่ง TCP segment ที่เซต SYN flag ไปที่ Server โดย Client จะสร้างหมายเลข Sequence Number เรียกว่า Initial Sequence Number (ISN) ขึ้นมา (ในรูปสมมติว่า 100) ใส่ใน SEQ# แล้วส่ง
- เมื่อ Server ได้รับ TCP segment ที่เซต SYN flag แล้วจะตอบกลับไปด้วย TCP segment ที่เซต SYN-ACK flags โดย Server จะมีการสร้างหมายเลข ISN ของตนเองขึ้นมาเช่นกัน โดยใส่ใน SEQ# และนำหมายเลข SN:Client+1 แล้วใส่ใน ACK# แล้วส่ง
- เมื่อ Client ได้รับ TCP segment ที่เซต SYN-ACK flags ก็จะต้องตอบกลับด้วย TCP segment ที่เซต ACK flag ซึ่งถือเป็น TCP segment สุดท้ายในการสร้าง TCP Connection โดย Client จะนำ SN:Client+1 ใส่ใน SEQ# และนำ SN:Server+1 ใส่ใน ACK# แล้วส่ง เมื่อส่ง TCP segment ดังกล่าวออกไปแล้ว จะถือว่าฝั่ง Client สร้างการเชื่อมต่อสำเร็จแล้ว ซึ่ง Client สามารถจะเริ่มส่งข้อมูลได้
- เมื่อ Server ได้รับ TCP segment สุดท้ายในการสร้าง TCP Connection ซึ่งมี ACK flag เซตเอาไว้ จะถือว่าฝั่ง Server สร้างการเชื่อมต่อสำเร็จแล้วเช่นกัน



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

not udp and not ssl and not arp and not snmp and not icmp

SYN

Src Port : 61598	Dest Port : 80
Seq # : 610997682	
Ack # : 0	
Flags : 0x002	Window Size : SYN

SYN-ACK

Src Port : 80	Dest Port : 61598
Seq # : 4134094401	
Ack # : 610997683	
Flags : 0x012	Window Size : SYN, ACK

ACK

Src Port : 61598	Dest Port : 80
Seq # : 610997683	
Ack # : 4134094402	
Flags : 0x010	Window Size : ACK

SYN SYN-ACK ACK

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

win = 8192
MSS = 1460

WS = 4

ข้อมูล	ความหมาย
Window	เป็นการกำหนดขนาดระหว่าง Client and server ว่าต้องการรองรับปริมาณเท่าใดถึงจะตอบกลับ
MSS	ค่า parameter ระบุขนาดข้อมูลสูงสุดที่ server ทั้ง 2 จะรับรู้ เพื่อป้องกันไม่ให้ packet มีขนาดมากเกินไป
SACK_PERM	เลือกรับข้อมูลได้ว่า จะรับข้อมูลช่วงไหนถึงไหน ช่วงไหนที่ไม่อยู่ในรายการก็จะไม่ถึง
WS	เป็นการปรับขนาดหน้าต่าง

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

win = 14300
MSS = 1430

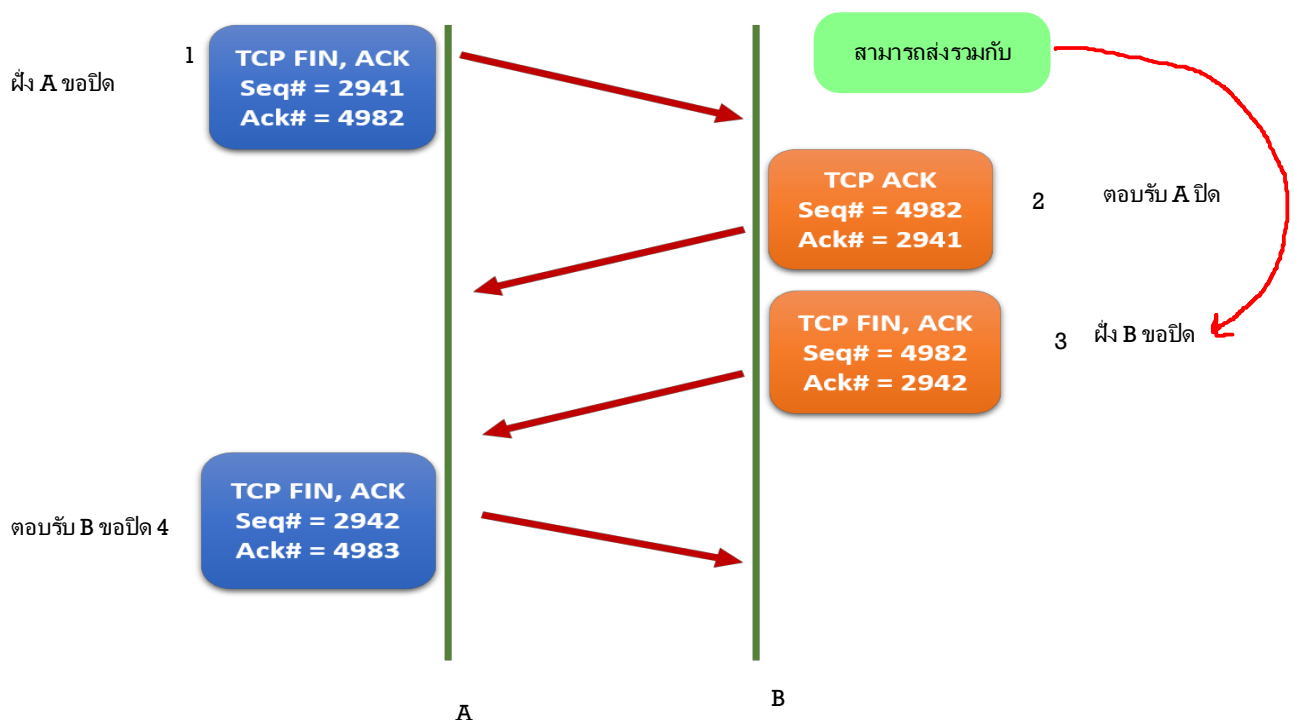
WS = 64

ข้อมูล	ความหมาย
Window	เป็นการกำหนดขนาดระหว่าง Client and server ว่าต้องการรองรับปริมาณเท่าใดถึงจะตอบกลับ
MSS	ค่า parameter ระบุขนาดข้อมูลสูงสุดที่ server ทั้ง 2 จะรับรู้ เพื่อป้องกันไม่ให้ packet มีขนาดมากเกินไป
SACK_PERM	เลือกรับข้อมูลได้ว่า จะรับข้อมูลช่วงไหนถึงไหน ช่วงไหนที่ไม่อยู่ในรายการก็จะไม่ถึง
WS	เป็นการปรับขนาดหน้าต่าง

- ให้ดู packet ที่ส่งข้อมูล packet แรก (หรือ packet อื่นก็ได้) ให้ตอบว่าในข้อมูลที่ไมเท่ากันของ Client กับ Server ในการเลือกใช้ข้อมูลหนึ่ง (เนื่องจากทั้ง 2 ด้านต้องใช้พารามิเตอร์เดียวกันในการส่งข้อมูล) คิดว่ามีหลักในการเลือกอย่างไร
client และ server นิยมใช้โปรโตคอลในการสื่อสารระหว่าง client และ server ได้แก่ HTTP, FTP, SMTP, SSH และอื่นๆ
โปรโตคอลเหล่านี้มักจะมีพารามิเตอร์ที่ต้องการใช้ร่วมกันในการสื่อสารและแลกเปลี่ยนข้อมูล เช่น พารามิเตอร์ URL
ใน HTTP, พารามิเตอร์ username และ password ใน FTP และ SSH, หรือพารามิเตอร์ sender และ receiver
ใน SMTP ซึ่งเมื่อ client และ server ตกลงใช้พารามิเตอร์เดียวกันแล้วจะช่วยให้การสื่อสารและแลกเปลี่ยนข้อมูลระหว่างกันเป็นไปอย่างสมบูรณ์และราบรื่น โดยลดความผิดพลาดในการส่งข้อมูลและเพิ่มประสิทธิภาพในการสื่อสารระหว่าง client และ server ได้ด้วย

TCP Connection Termination (หรือ TCP Connection Teardown)

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
- ฝ่าย 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)
- ฝ่าย 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
- ฝ่าย 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 # : 610998005	
Ack # : 4134095528	
Flags : 0x011	Window Size : FIN,ACK

Packet# 1664	
Src Port : 80	Dest Port : 61598
Seq # : 4134095528	
Ack # : 610998006	
Flags : 0x011	Window Size : FIN,ACK

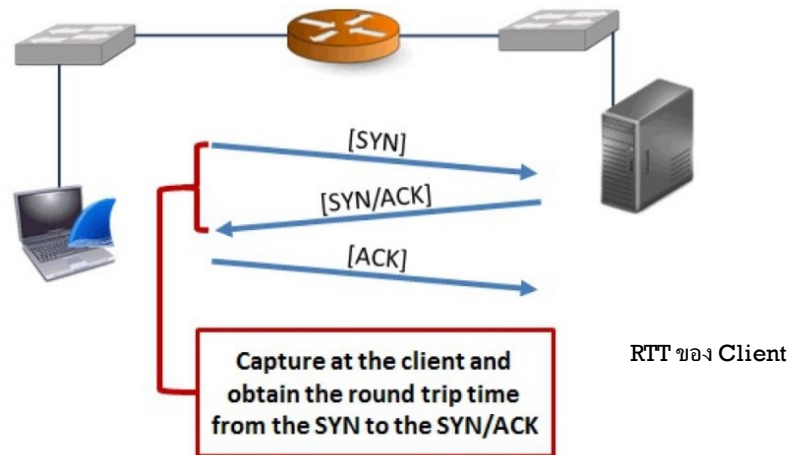
Packet# 1665	
Src Port : 61598	Dest Port : 80
Seq # : 610998006	
Ack # : 4134095529	
Flags : 0x010	Window Size : ACK

วิธีค้นหา

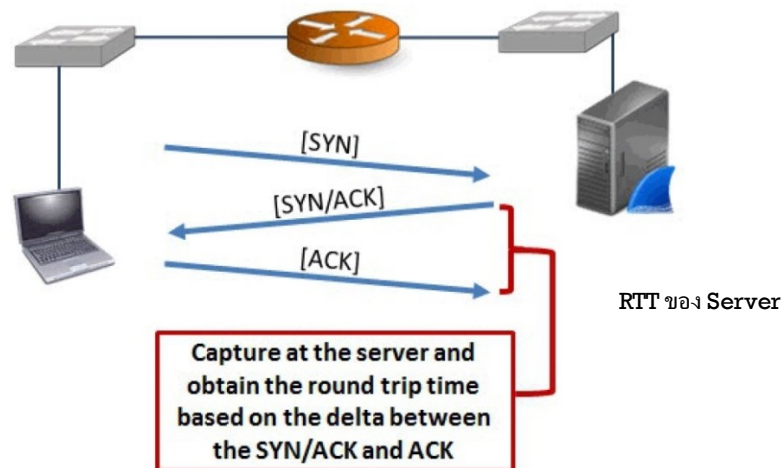
 (tcp.srcport == 61598 and tcp.dstport == 80) or (tcp.srcport == 80 and tcp.dstport == 61598)

เลือกดูข้อมูลที่ต้องการปิดการเชื่อมต่อ (FIN)

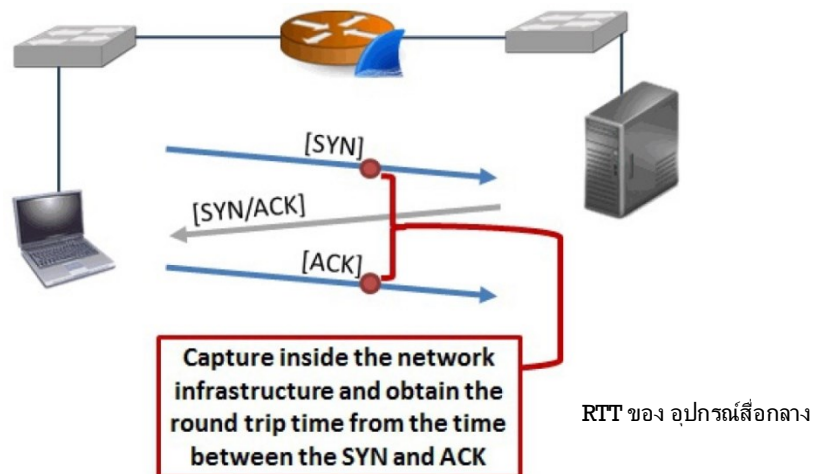
3. ใน Wireshark เราสามารถจะหา packet ที่มีคุณลักษณะของ flags เฉพาะได้ โดยใช้ display filter tcp.flags เช่น **tcp.flags.syn==1** หรือ **tcp.flags.ack==1** ซึ่งเราสามารถใช้เวลา RTT ของ TCP handshake ได้ โดยการหา RTT ของ TCP handshake มี 3 แบบ คือ วัดจากฝั่ง Client จะใช้เวลา ระหว่าง SYN และ SYN-ACK



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 ופן (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)

1

No.	Time	DNS Delta	Source	Destination	Protocol	Length	Info
1	0.000000		24.6.173.220	173.194.79.121	TCP	66	61598 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
2	0.035945		173.194.79.121	24.6.173.220	TCP	66	80 → 61598 [SYN, ACK] Seq=0 Ack=1 Win=14300 Len=0 MSS=1430 SACK_PERM WS=64

2

No.	Time	DNS Delta	Source	Destination	Protocol	Length	Info
2	0.035945		173.194.79.121	24.6.173.220	TCP	66	80 → 61598 [SYN, ACK] Seq=0 Ack=1 Win=14300 Len=0 MSS=1430 SACK_PERM WS=64
3	0.036067		24.6.173.220	173.194.79.121	TCP	54	61598 → 80 [ACK] Seq=1 Ack=1 Win=65780 Len=0

3

No.	Time	DNS Delta	Source	Destination	Protocol	Length	Info
1	0.000000		24.6.173.220	173.194.79.121	TCP	66	61598 → 80 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM
3	0.036067		24.6.173.220	173.194.79.121	TCP	54	61598 → 80 [ACK] Seq=1 Ack=1 Win=65780 Len=0

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 ของเว็บต่าง ๆ จำนวน 3 เว็บ แล้วนำค่ามาใส่ตาราง

URL	เวลา
www.ce.kmitl.ac.th	0.00087
www.datastruc.ce.kmitl.ac.th	0.000132
www.42bangkok.com	0.000082

- ให้ตอบว่าระหว่าง RTT ที่วัดในครั้งนี้ กับ HTTP RTT ที่วัดในครั้งก่อนหน้านี้ บอกถึงอะไร และแตกต่างกันอย่างไร

RTT ของ TCP เป็นแค่ช่วงเวลาในการ Handshake (การเชื่อมต่อ)

RTT ของ HTTP เป็นช่วงในการขอข้อมูลจาก Server จนส่งหน้า Website มาให้เรา

งานครั้งที่ 6

- การส่งงาน เขียนหรือพิมพ์ลงในเอกสารนี้ และส่งเป็นไฟล์ PDF เท่านั้น
- ตั้งชื่อไฟล์โดยใช้รหัสนักศึกษา ตามด้วย section และ _lab06 ตามตัวอย่างต่อไปนี้
64019999_sec20_lab06.pdf
- กำหนดส่ง ภายในวันที่ 24 กุมภาพันธ์ 2566 โดยให้ส่งใน Microsoft Teams ของรายวิชา