

กิจกรรมที่ 6 : TCP Connection

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

- Reliable, in-order delivery คือ การส่งไม่ผิดพลาดโดยข้อมูลมีการเรียงตามลำดับ
- Connection Oriented คือ ต้องมีการสร้างการเชื่อมต่อ ก่อน และมีการแยกเปลี่ยนข้อมูลควบคุม
- Flow Control ควบคุมการให้ผลของข้อมูลระหว่าง Process ทั้ง 2 ด้าน
- Congestion Control ควบคุมการให้ผลของข้อมูลผ่านอุปกรณ์เครือข่าย
- Full Duplex data สามารถส่งได้ทั้ง 2 ทาง ใน การเชื่อมต่อเดียว กัน

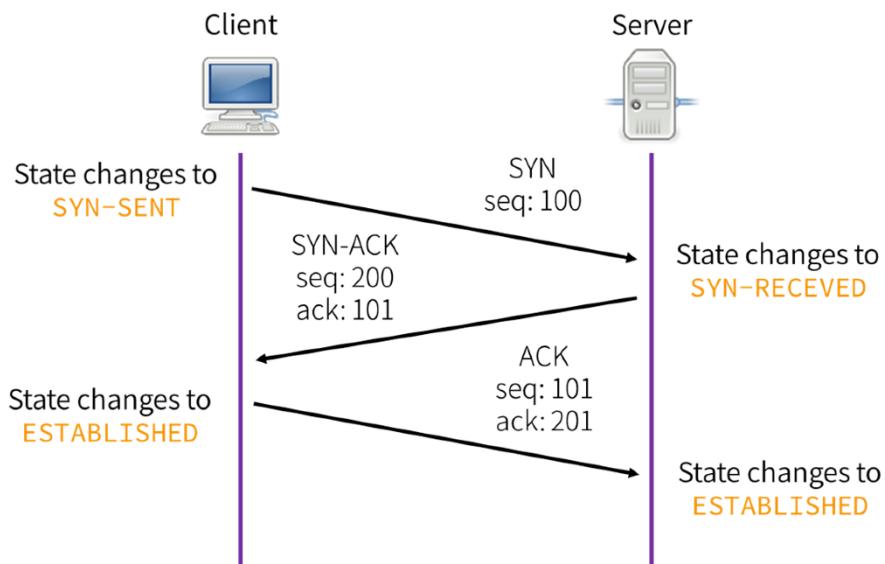
Connection Setup

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		

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# และส่ง
- เมื่อ Server ได้รับ packet SYN จะตอบกลับโดย packet SYN-ACK โดย Server จะมีการสร้างหมายเลข ISN ของตนเองขึ้นมา เช่นกัน โดยใส่ใน SEQ# และนำหมายเลข SN:Client+1 และใส่ใน ACK# และส่ง
- เมื่อ Client ได้รับ packet SYN-ACK ก็จะตอบกลับโดย packet ACK สุดท้าย โดย Client จะนำ SN:Client+1 ใส่ใน SEQ# และนำ SN:Server+1 ใส่ใน ACK# และส่ง เมื่อถึงตรงนี้จะถือว่าฝั่ง Client สร้างการเชื่อมต่อสำเร็จแล้ว ซึ่ง Client สามารถจะเริ่มส่งข้อมูลได้
- เมื่อ Server ได้รับ packet ACK สุดท้าย จะถือว่าฝั่ง Server สร้างการเชื่อมต่อสำเร็จแล้ว เช่นกัน



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

SYN

Src Port : 61598	Dest Port : 80
Seq # : 610997682	
Ack # : 0	
Flags : 0x002	8192

SYN-ACK

Src Port : 80	Dest Port : 61598
Seq # : 4134094401	
Ack # : 610997683	
Flags : 0x012	14300

ACK

Src Port : 61598	Dest Port : 80
Seq # : 610997683	
Ack # : 4134094402	
Flags : 0x010	65780

- ค่าความยาวข้อมูลของ packet ทั้ง 3 เท่ากับเท่าไรบ้าง **66, 66, 54 bytes** ตามลำดับ
- ใน packet SYN มีข้อมูลอื่นๆ ส่งมาด้วยหรือไม่ อะไรมาก (ดูในคอลัมน์ info) และข้อมูลต่างๆ เหล่านั้นมีความหมายอะไรหรือไม่ไปใช้อะไร (ให้ค้นหาข้อมูลเพิ่มเติมจากหนังสือ)

ข้อมูล	ความหมาย
$Win = 8192$	The window size from TCP header
$Len = 0$	TCP Segment Length
$MSS = 1460$	Maximum segment size
$WS = 4$	Window scale

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

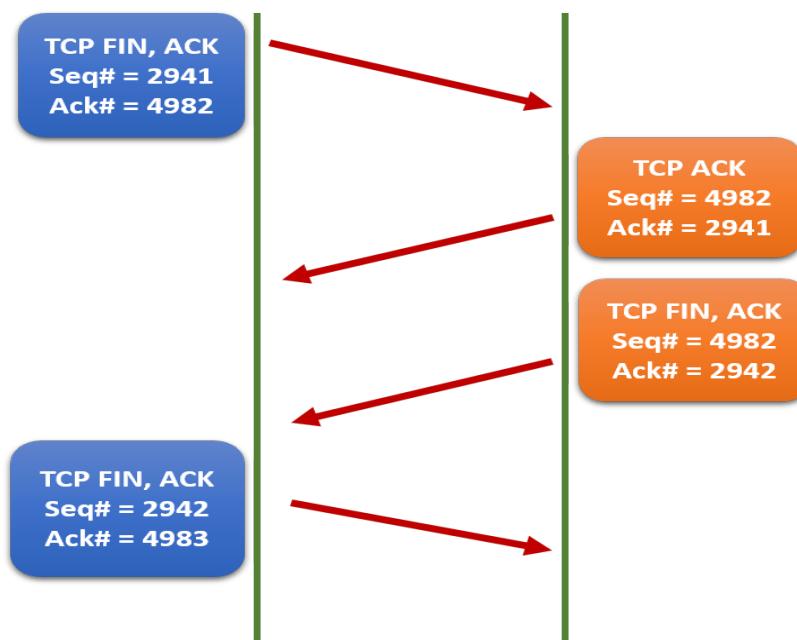
ข้อมูล	ความหมาย
$Win : 14300$	The window size value from the TCP header
$Len : 0$	TCP Segment Length
$WS : 64$	Window Scale
$SACK_PERM : 1$	Selective Acknowledgment

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

Client ที่ GET ข้อมูลที่ต้องการไปยัง Server ที่ Server สำหรับลูกค้าที่ต้องการให้
 แล้ว Server จะมายัง ACK ของ GET ที่เป็นห้องข้อมูลกับ ms หากถูก Client
 ได้รับข้อมูลที่ต้องการ คือ ACK กลับไป Server ทันที

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 เมื่อถึงจุดนี้จะถือว่าเป็น การสิ้นสุด 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	16163

Packet# 1664	
Src Port : 80	Dest Port : 61598
Seq # : 1127	
Ack # : 324	
Flags : 0x011	241

Packet# 1665	
Src Port : 61598	Dest Port : 80
Seq # : 324	
Ack # : 1128	
Flags : 0x010	16163

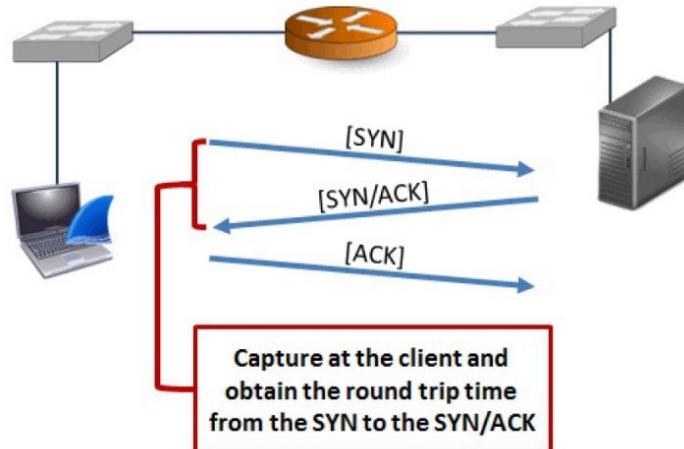
วิธีค้นหา

((ip.dst == 173.194.79.121) && (ip.src == 24.6.133.220)) or

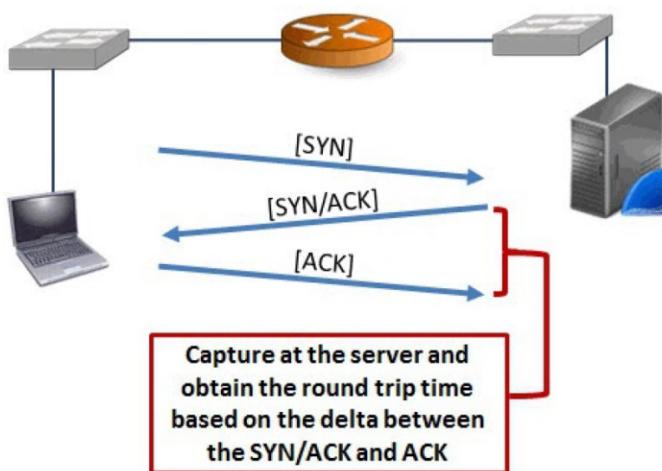
((ip.dst == 24.6.173.220) && (ip.src == 173.194.79.121))

→ u1 Flag FIN, ACK

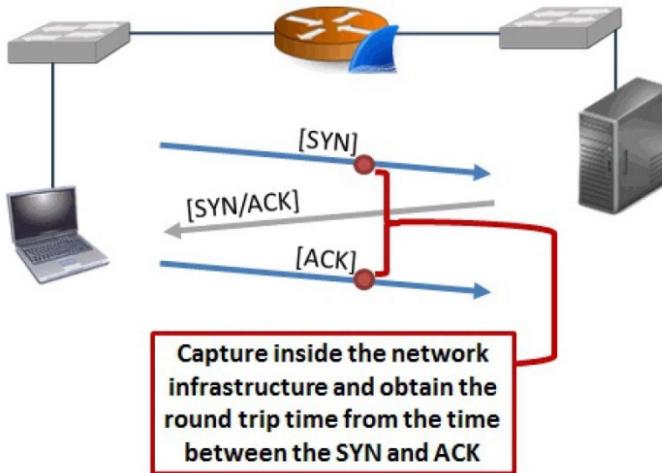
3. ใน Wireshark เราสามารถจะหา packet ที่มีคุณลักษณะของ flags เนพาะได้โดยใช้ display filter `tcp.flags` เช่น `tcp.flags.syn==1` หรือ `tcp.flags.ack==1` ซึ่งเราสามารถใช้หา RTT ของ TCP handshake ได้โดยการหา RTT ของ TCP handshake มี 3 แบบ คือ วัดจากฝั่ง Client จะใช้เวลาระหว่าง SYN และ SYN-ACK



และวัดจากฝั่ง Server จะใช้เวลาระหว่าง SYN/ACK กับ ACK



แต่ในกรณีที่วัดจากอุปกรณ์ ควรใช้ระหว่าง SYN และ ACK ตามรูป



4. จากไฟล์ http-browse101d.pcapng ให้สร้าง display filter ที่สามารถแสดงเฉพาะ packet ที่เป็น Open Connection (3 way handshake) คือที่กำหนดของทุกๆ TCP Stream โดยไม่มี packet อื่นๆ มาปน (นักศึกษาพยายามคิดด้วยตนเอง) ให้เขียนวิธีการหา และ display filter ของแต่ละอัน

- 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)

- (tcp.flags.syn == 1)

- (tcp.ack == 1) && !(tcp.flags.push == 1)

- (tcp.flags == 2) || (tcp.flags == 16 && tcp.ack == 1)

5. เราสามารถใช้ค่า RTT ของ TCP handshaking ตามข้อ 4 มาใช้วัดประสิทธิภาพของ Web Server ได้ เช่นกัน โดย Server ที่มีค่า RTT น้อย แสดงถึงการตอบสนองที่รวดเร็ว ดังนั้นให้ capture ข้อมูลจากเว็บ และใช้ display filter ตามข้อ 4 (ให้นักศึกษาเลือกใช้ตัวที่เหมาะสม) เพื่อหาค่า RTT ของเว็บต่างๆ จำนวน 3 เว็บ และนำค่ามาใส่ตาราง

URL	เวลา
www.facebook.com	1.0897 sec
www.coursera.org	0.005583 sec
www.youtube.com	1.891 sec

- ให้ต่อรองว่าระหว่าง RTT ที่วัดในครั้งนี้ กับ HTTP RTT ที่วัดในครั้งก่อนหน้านี้ บวกถึงอะไร และแตกต่างกันอย่างไร

RTT : ระยะเวลาที่ต้องการสำหรับ TCP handshake

HTTP RTT : ระยะเวลาที่ต้องการเมื่อ Browser ส่ง request ไปจนได้รับ response จาก Server

งานครั้งที่ 6

- การส่งงาน เขียนหรือพิมพ์ลงในเอกสารนี้ และส่งโดยเป็นไฟล์ PDF เท่านั้น
- ตั้งชื่อไฟล์โดยใช้รหัสนักศึกษา และ _Lab6 เช่น 63010789_Lab6.pdf
- กำหนดส่ง ภายในวันที่ 23 กุมภาพันธ์ 2565