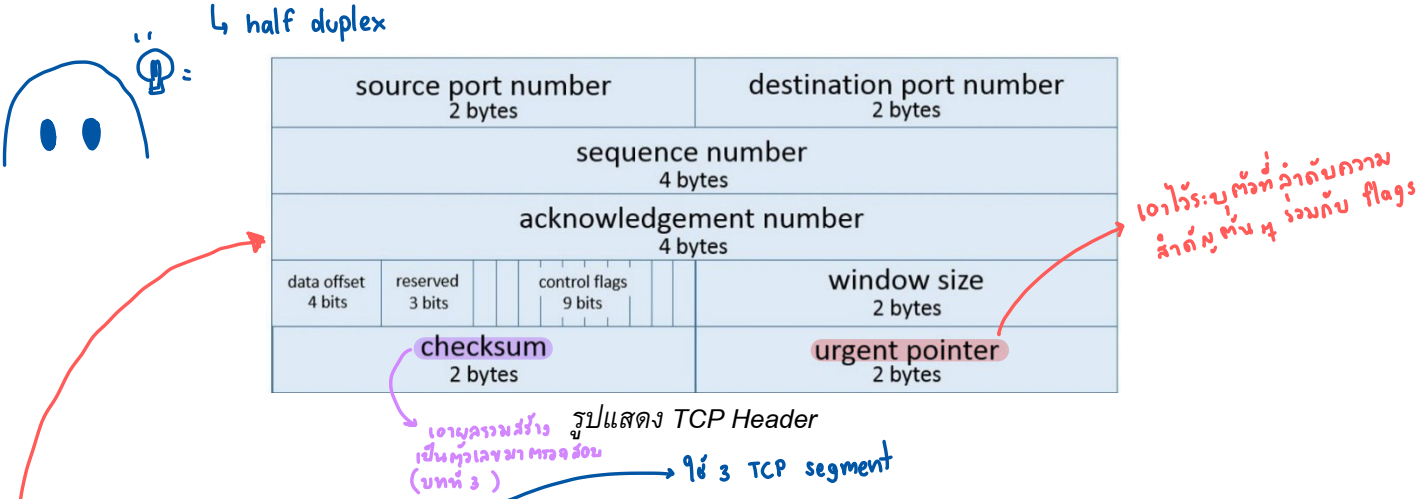
กิจกรรมที่ 6 : TCP Connection

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

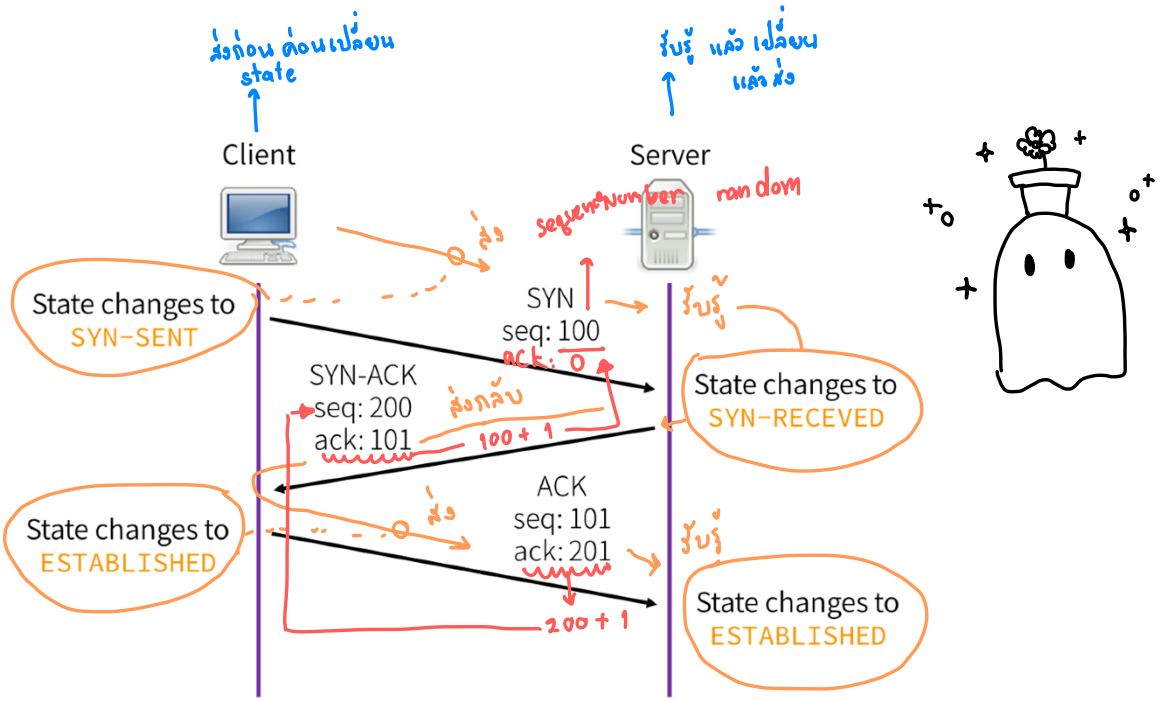


รูปแสดง TCP Header

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 ก็จะตอบกลับด้วย 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 ให้บอกว่ามี Flag ใดที่ Set บ้าง

SYN

Src Port : 61598	Dest Port : 80
Seq # : 610997682	
Ack # : 0	
Flags : SYN	Window Size : 8192

↳ บอกความสามารถในการรับ

SYN-ACK

Src Port : 80	Dest Port : 61598
Seq # : 4134094401	
Ack # : 1	
Flags : SYN, ACK	Window Size : 14300

ACK

Src Port : 61598	Dest Port : 80
Seq # : 610997682	
Ack # : 4134094401	
Flags : ACK	Window Size : 65780

- ค่าความยาวข้อมูลของ packet ทั้ง 3 เท่ากับเท่าไรบ้าง 66 , 66 , 54
- ใน packet ที่เซต SYN flag มีข้อมูลอื่นๆ ส่งมาด้วยหรือไม่ อะไรบ้าง (ดูในคอลัมน์ info) และข้อมูลต่างๆ เหล่านั้นมีความหมายอะไรหรือนำไปใช้อะไร (ให้ค้นหาข้อมูลเพิ่มเติมจากหนังสือ)
  - MSS - maximum segment site
  - WS - window scale
  - SACK_PERM - SACK permitted

ข้อมูล		ความหมาย
MSS	Maximum segment size	ค่า parameter ระบุขนาดข้อมูลสูงสุดที่ server ทั้ง 2 รับรู้ เพื่อป้องกันไม่ให้ packet มีขนาดเกินป้องกันการที่ packet drop หายไป
WS	window scale	ค่าที่ควบคุมขนาดข้อมูล window ก่อนจะมีการรับทราบ
SACK_PERM	SACK permitted	ตัวเลือกอนุญาตผู้ส่ง TCP ขยักได้ Selective Acknowledgment (SACK) ในระหว่างการส่งข้อมูล

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

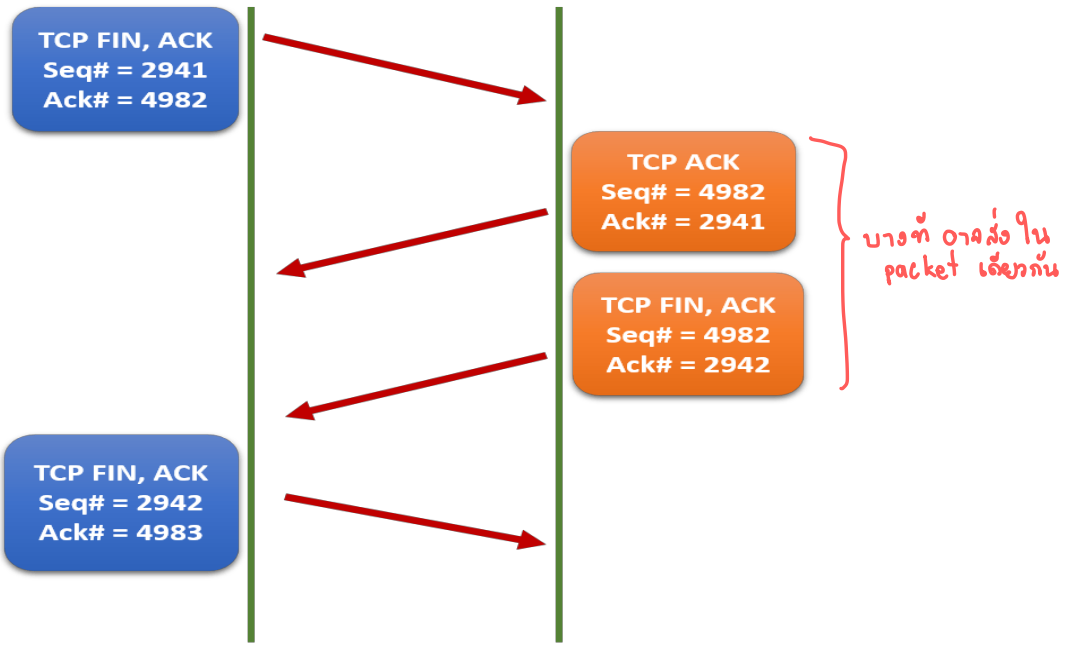
ข้อมูล		ความหมาย
Ack = 1		ยืนยันกับผู้ส่งว่าผู้รับได้ packet แล้ว และพร้อมรับ packet ถัดไป
MSS	Maximum segment size	ค่า parameter ระบุขนาดข้อมูลสูงสุดที่ server ทั้ง 2 รับรู้ เพื่อป้องกันไม่ให้ packet มีขนาดเกินป้องกันการที่ packet drop หายไป
WS	window scale	ค่าที่ควบคุมขนาดข้อมูล window ก่อนจะมีการรับทราบ
SACK_PERM	SACK permitted	ตัวเลือกอนุญาตผู้ส่ง TCP ขยักได้ Selective Acknowledgment (SACK) ในระหว่างการส่งข้อมูล

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

เลือกตามฝั่งที่ส่งข้อมูล

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
- ฝั่ง B จะตอบด้วย packet ที่มี ACK flag โดยใช้ SEQ# เท่ากับ ACK# ของ FIN/ACK ก่อนหน้า และใช้ ACK# เท่ากับของ ~~SYN#~~ SEQ# 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)

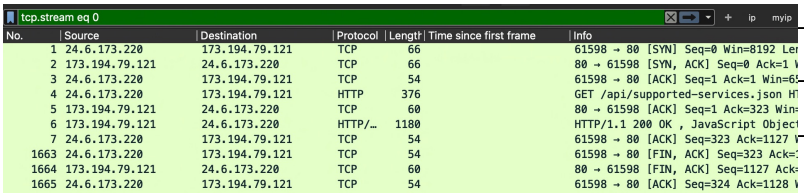
Packet# 1663	
Src Port : 61598	Dest Port : 80
Seq # : 323	
Ack # : 1127	
Flags : 0x011 (FIN, ACK)	Window Size : 16163

Packet# 1664	
Src Port : 80	Dest Port : 61598
Seq # : 1127	
Ack # : 324	
Flags : 0x011 (FIN, ACK)	Window Size : 291

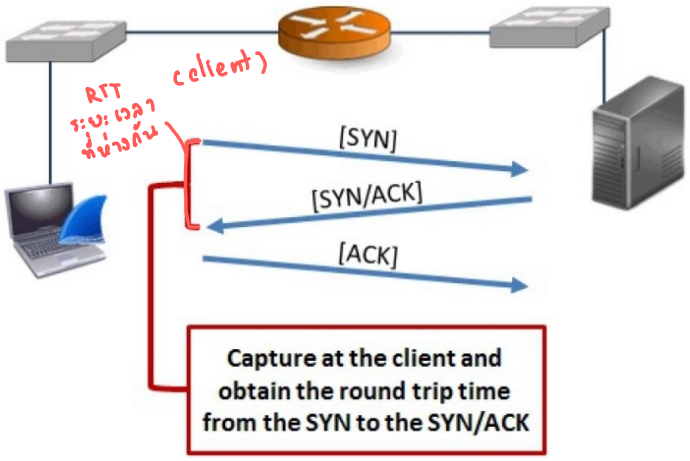
Packet# 1665	
Src Port : 61598	Dest Port : 80
Seq # : 324	
Ack # : 1128	
Flags : 0x010 (ACK)	Window Size : 16163

วิธีค้นหา

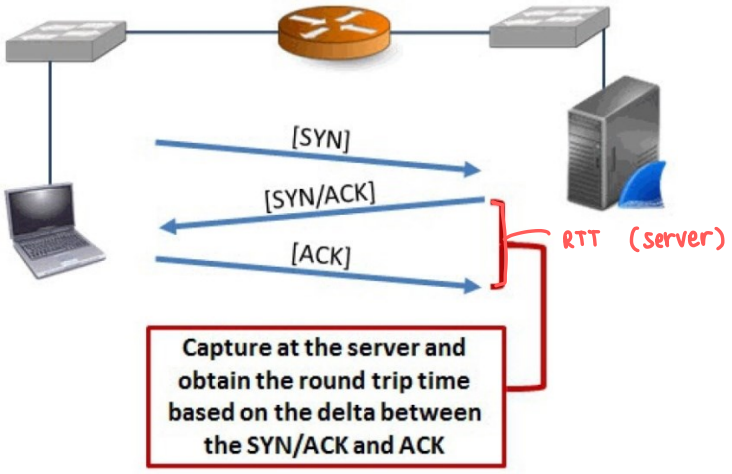
tcp.stream eq 0



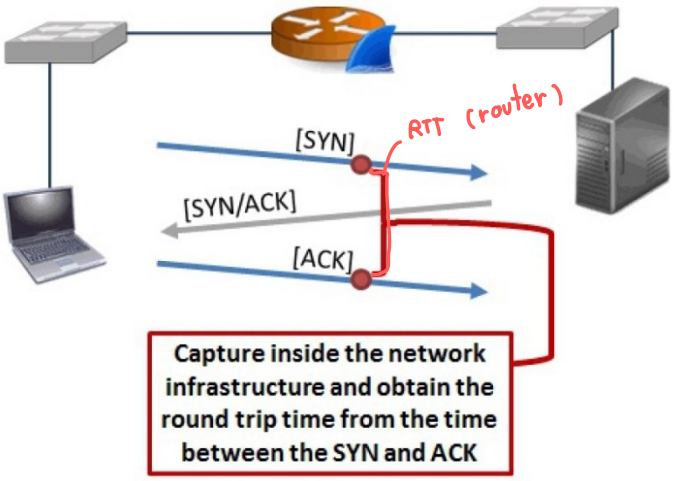
3. ใน Wireshark เราสามารถจะหา packet ที่มีคุณลักษณะของ flags เฉพาะได้ โดยใช้ display filter tcp.flags เช่น **tcp.flags.syn==1** หรือ **tcp.flags.ack==1** ซึ่งเราสามารถใช้หา RTT ของ TCP handshake ได้ โดยการหา RTT ของ TCP handshake มี 3 แบบ คือ วัดจากฝั่ง Client จะใช้เวลาระหว่าง SYN และ SYN-ACK



และวัดจากฝั่ง Server จะใช้เวลาระหว่าง SYN/ACK กับ ACK



แต่ในกรณีที่วัดจากอุปกรณ์ ควรใช้ระหว่าง SYN และ ACK ตามรูป (ไม่แม่นนัก)



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)

   1 ((tcp.flags.syn==1 and tcp.flags.ack==0)||(tcp.flags.syn==1 and tcp.flags.ack==1))and tcp.stream eq 0

   2 (tcp.ack == 1 && tcp.nxtseq ==1) && tcp.stream eq 0

   3 ((tcp.flags.syn == 1 && tcp.flags.ack==0)||(tcp.ack==1&&tcp.seq==1)) && tcp.stream eq 0 && !http o

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
①	http://compro.ce.kmitl.ac.th	0.009968000
②	https://www.reg.kmitl.ac.th	0.007457000
③	https://www.ce.kmitl.ac.th	0.010479000

① ((tcp.flags.syn == 1 && tcp.flags.ack==0)||(tcp.ack==1&&tcp.seq==1)) && tcp.stream eq 0 && !http

No.	Source	Protocol	Length	Time since first fra	Info
1	192.168.14.80	TCP	78	0.000000000	57406 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=3016450451 TSecr=0 SACK_PERM
3	192.168.14.80	TCP	66	0.009968000	57406 → 80 [ACK] Seq=1 Ack=1 Win=131328 Len=0 TSval=3016450461 TSecr=3157792974

② ((tcp.flags.syn == 1 && tcp.flags.ack==0)||(tcp.ack==1&&tcp.seq==1)) && tcp.stream eq 0

No.	Source	Protocol	Length	Time since first fra	Info
1	192.168.14.80	TCP	78	0.000000000	57452 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=4152756587 TSecr=0 SACK_PERM
3	192.168.14.80	TCP	66	0.007457000	57452 → 443 [ACK] Seq=1 Ack=1 Win=131328 Len=0 TSval=4152756594 TSecr=1192764183

③ ((tcp.flags.syn == 1 && tcp.flags.ack==0)||(tcp.ack==1&&tcp.seq==1)) && tcp.stream eq 0 && !http

No.	Source	Protocol	Length	Time since first	Info
1	192.168.14.80	TCP	78	0.000000000	57581 → 443 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=64 TSval=471267358 TSecr=0 SACK_PERM
3	192.168.14.80	TCP	66	0.010479000	57581 → 443 [ACK] Seq=1 Ack=1 Win=131328 Len=0 TSval=471267369 TSecr=3790222651

- ให้ตอบว่าระหว่าง RTT ที่วัดในครั้งนี้ กับ HTTP RTT ที่วัดในครั้งก่อนหน้านี้ บอกถึงอะไร และแตกต่างกันอย่างไร

ครั้งนี้ คือ เวลา ในการส่ง packet → server และส่งกลับมา
ครั้งก่อน คือ เวลา ในการ ส่ง data ระหว่าง web browser กับ web server

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