

Nattapat Yuvasuta	59070501028
Nuttawut Kuadplod	59070501029
Niti Buesamae	59070501043
Panumas Wongsang	59070501056
Supadet Jomthepmala	59070501069

## **Part II : Transmitting Data with TCP**

Include relevant parts of the Wireshark data to support each of your answers.

2.1 How many TCP segments (both control and data) are exchanged between PC1 and PC2 ?

**Ans.** 30

2.2 What are the sizes of the data segments in Question 2.1? Are they all the same ? If not, explain why they have different sizes.

**Ans.** 1024,1448(6 blocks),528(3 blocks). They aren't same because it can't send data  
 $10 \times 1024 = 10240$  in 10 time then the amount of data  $1024 + 6 \times 1448 + 3 \times 528 = 10240$   
 (byte stream service)

2.3 How many segments are TCP control segments? What are they (look at the Flag field) ?

**Ans.** 10 segments there are Reserved, nonce, congestion, ECN-echo, Urgent , ACK, Push, Reset, Syn, Fin

2.4 Compare the total number of bytes transmitted, in both directions, including Ethernet, IP, and TCP headers, to the amount of application data transmitted.

**Ans.** 66

Nattapat Yuvasuta	59070501028
Nuttawut Kuadplod	59070501029
Niti Buesamae	59070501043
Panumas Wongsang	59070501056
Supadet Jomthepmala	59070501069

### Part III: File Transfers Using TCP and UDP

3.1 From the timestamps recorded by Wireshark, obtain the times it took to transfer the large file with FTP and with TFTP. Which one, FTP or TFTP, transfers the file faster? Look at the Wireshark data and use your knowledge of FTP, TFTP, TCP, and UDP to explain the outcome.

#### FTP

116	72.229351	10.0.3.2	10.0.3.1	FTP	86 Request: RETR largefile.txt
214...	74.002978	10.0.3.1	10.0.3.2	FTP	90 Response: 226 Transfer complete.

#### TFTP

15	20.052914	10.0.3.2	10.0.3.1	TFTP	67 Read Request, File: largefile.txt, Transfer type: netascii
790...	35.363549	10.0.3.2	10.0.3.1	TFTP	60 Acknowledgement, Block: 40313

**Ans.** FTP เร็วกว่า เนื่องจากว่า FTP ใช้โปรโตคอล TCP ที่มีการสร้างช่องทางการรับส่งข้อมูล ทำให้การันตีว่าข้อมูลจะถูกส่งไปยังผู้รับอย่างแน่นอน และลำดับข้อมูลอีกด้วย แต่ TFTP ใช้ โปรโตคอล UDP ซึ่งไม่มีการสร้างช่องทางการรับส่งข้อมูล ทำให้ไม่มีการการันตีว่าข้อมูลที่ส่งไปนั้นจะไปถึงยังผู้รับ อีกทั้งยังไม่มีการลำดับข้อมูลด้วย เมื่อข้อมูลสูญหายระหว่างทางทำให้ TFTP ต้องส่งข้อมูลใหม่ทั้งหมด เนื่องจากว่าไม่รู้ว่ามีข้อมูลส่วนไหนสูญหาย จึงทำให้ TFTP โอนถ่ายข้อมูลช้ากว่า FTP

3.2 How many parallel connections were created in the FTP session? What are those connections for? What are the source/destination port numbers of those connections?

214...	74.002978	10.0.3.1	10.0.3.2	FTP	90 Response: 226 Transfer complete.
214...	74.003213	10.0.3.2	10.0.3.1	TCP	66 36724 → 21 [ACK] Seq=201 Ack=644 Win=29312 Len=0 TSval=418645 TSecr=418654
214...	74.015134	10.0.3.2	10.0.3.1	TCP	66 52112 → 20 [FIN, ACK] Seq=1 Ack=20480002 Win=755200 Len=0 TSval=418646 TSecr=418654
214...	74.015234	10.0.3.1	10.0.3.2	TCP	66 20 → 52112 [ACK] Seq=20480002 Ack=2 Win=29312 Len=0 TSval=418655 TSecr=418646
214...	74.198863	Cisco_dc:al:81	Spanning-tree-(for...	STP	60 Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
214...	76.202981	Cisco_dc:al:81	Spanning-tree-(for...	STP	60 Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
214...	76.597318	Cisco_dc:al:81	Cisco_dc:al:81	LOOP	60 Reply
214...	78.207896	Cisco_dc:al:81	Spanning-tree-(for...	STP	60 Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
214...	80.217158	Cisco_dc:al:81	Spanning-tree-(for...	STP	60 Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
214...	82.217665	Cisco_dc:al:81	Spanning-tree-(for...	STP	60 Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
214...	84.037532	10.0.3.2	10.0.3.1	FTP	72 Request: QUIT
214...	84.037769	10.0.3.1	10.0.3.2	FTP	80 Response: 221 Goodbye.
214...	84.037851	10.0.3.1	10.0.3.2	TCP	66 21 → 36724 [FIN, ACK] Seq=658 Ack=207 Win=29056 Len=0 TSval=419657 TSecr=419649
214...	84.038517	10.0.3.2	10.0.3.1	TCP	66 36724 → 21 [ACK] Seq=207 Ack=658 Win=29312 Len=0 TSval=419649 TSecr=419657
214...	84.039536	10.0.3.2	10.0.3.1	TCP	66 36724 → 21 [FIN, ACK] Seq=207 Ack=659 Win=29312 Len=0 TSval=419649 TSecr=419657
214...	84.039590	10.0.3.1	10.0.3.2	TCP	66 21 → 36724 [ACK] Seq=659 Ack=208 Win=29056 Len=0 TSval=419658 TSecr=419649
214...	84.225272	Cisco_dc:al:81	Spanning-tree-(for...	STP	60 Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
214...	86.227511	Cisco_dc:al:81	Spanning-tree-(for...	STP	60 Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
214...	86.608871	Cisco_dc:al:81	Cisco_dc:al:81	LOOP	60 Reply
214...	88.232461	Cisco_dc:al:81	Spanning-tree-(for...	STP	60 Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
214...	90.236869	Cisco_dc:al:81	Spanning-tree-(for...	STP	60 Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
214...	92.242210	Cisco_dc:al:81	Spanning-tree-(for...	STP	60 Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001

▶ Frame 21450: 90 bytes on wire (720 bits), 90 bytes captured (720 bits) on interface 0  
 ▼ Ethernet II, Src: Raspberr\_6a:01:7a (b8:27:eb:6a:01:7a), Dst: Raspberr\_bf:0b:d7 (b8:27:eb:bf:0b:d7)  
 ▶ Destination: Raspberr\_bf:0b:d7 (b8:27:eb:bf:0b:d7)  
 ▶ Source: Raspberr\_6a:01:7a (b8:27:eb:6a:01:7a)  
 Type: IPv4 (0x0800)  
 ▶ Internet Protocol Version 4, Src: 10.0.3.1, Dst: 10.0.3.2  
 ▶ Transmission Control Protocol, Src Port: 21, Dst Port: 36724, Seq: 620, Ack: 201, Len: 24  
 ▼ File Transfer Protocol (FTP)  
 ▶ 226 Transfer complete.\r\n

**Ans.** จำนวน 2 session คือ Control connection(Passive mode) สำหรับสร้างช่องทางการรับส่งข้อมูล และ Data connection(Active mode) สำหรับการรับส่งข้อมูล โดย source ใช้พอร์ต 21 และ destination ใช้พอร์ต 36724

Nattapat Yuvasuta	59070501028
Nuttawut Kuadplod	59070501029
Niti Buesamae	59070501043
Panumas Wongsang	59070501056
Supadet Jomthepmala	59070501069

#### Part IV: TCP Connection Management

Use the saved Wireshark output to answer the following questions. Include relevant parts of the Wireshark data to support each of your answers.

4.1 Identify the segments of the three-way handshake. Which flags are set in the TCP headers? Explain how these flags are interpreted by the receiving TCP server or TCP client.

**Ans.** SYN and Ack are set in the TCP headers  
 PC1 **sends** a TCP **SYN**chronize packet to PC2  
 PC2 receives PC1's **SYN**  
 PC1 receives PC2's **SYN-ACK**  
 PC1 sends **ACK**nowledge  
 PC2 receives **ACK**.

4.2 During the connection setup, the TCP client and TCP server tell each other the first sequence number they will use for data transmission. What are the initial sequence numbers of the TCP client and the TCP server? In Wireshark, you need to go to Preferences → Protocols → TCP and uncheck the box that says “Relative Sequence Number” before answering this question.

18	22.119100	10.0.3.1	10.0.3.2	TCP	74	40158 → 23	[SYN] Seq=2177393940 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=472011 TSecr=0 WS=128
19	22.119458	10.0.3.2	10.0.3.1	TCP	74	23 → 40158	[SYN, ACK] Seq=2918594200 Ack=2177393941 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=472011 TSecr=472002
20	22.119516	10.0.3.1	10.0.3.2	TCP	66	40158 → 23	[ACK] Seq=2177393941 Ack=2918594201 Win=29312 Len=0 TSval=472011 TSecr=472002

**Ans.** Sequence number of client is 2177393940 and sequence number of server is 2918594200

4.3 Identify the first segment that contains application data. What is the sequence number used in the first byte of application data sent from the TCP client to the TCP server?

18	22.119100	10.0.3.1	10.0.3.2	TCP	74	40158 → 23	[SYN] Seq=2177393940 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=472011 TSecr=0 WS=128
19	22.119458	10.0.3.2	10.0.3.1	TCP	74	23 → 40158	[SYN, ACK] Seq=2918594200 Ack=2177393941 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=472011 TSecr=472002
20	22.119516	10.0.3.1	10.0.3.2	TCP	66	40158 → 23	[ACK] Seq=2177393941 Ack=2918594201 Win=29312 Len=0 TSval=472011 TSecr=472002
21	22.119757	10.0.3.1	10.0.3.2	TELNET	93	Telnet Data ...	

**Ans.** Sequence number is 2177393941

4.4 The TCP client and TCP server exchange the initial window sizes to get the maximum amount of data that the other side can send at any time. Determine the values of the initial window sizes for the TCP client and the TCP server.

18	22.119100	10.0.3.1	10.0.3.2	TCP	74	40158 → 23	[SYN] Seq=2177393940 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=472011 TSecr=0 WS=128
19	22.119458	10.0.3.2	10.0.3.1	TCP	74	23 → 40158	[SYN, ACK] Seq=2918594200 Ack=2177393941 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=472011 TSecr=472002
20	22.119516	10.0.3.1	10.0.3.2	TCP	66	40158 → 23	[ACK] Seq=2177393941 Ack=2918594201 Win=29312 Len=0 TSval=472011 TSecr=472002

**Ans.** Window size of client is 29200 and Window size of client is 28960

4.5 What is the MSS value that is negotiated between the TCP client and the TCP server?

18	22.119100	10.0.3.1	10.0.3.2	TCP	74	40158 → 23	[SYN] Seq=2177393940 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=472011 TSecr=0 WS=128
19	22.119458	10.0.3.2	10.0.3.1	TCP	74	23 → 40158	[SYN, ACK] Seq=2918594200 Ack=2177393941 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=472011 TSecr=472002
20	22.119516	10.0.3.1	10.0.3.2	TCP	66	40158 → 23	[ACK] Seq=2177393941 Ack=2918594201 Win=29312 Len=0 TSval=472011 TSecr=472002

**Ans.** MSS value is 1460

Nattapat Yuvasuta	59070501028
Nuttawut Kuadplod	59070501029
Niti Buesamae	59070501043
Panumas Wongsang	59070501056
Supadet Jomthepmala	59070501069

#### 4.6 How long does it take to open the TCP connection?

18	22.119100	10.0.3.1	10.0.3.2	TCP	74	40158 → 23 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=472011
19	22.119458	10.0.3.2	10.0.3.1	TCP	74	23 → 40158 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 T
20	22.119516	10.0.3.1	10.0.3.2	TCP	66	40158 → 23 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=472011 TSecr=472002

**Ans. 0.000416 sec.**

4.7 In Step 3, identify the packets that are involved in closing the TCP connection. Which flags are set in these packets? Explain how these flags are interpreted by the receiving TCP server or TCP client.

128	71.656273	10.0.3.1	10.0.3.2	TCP	66	40158 → 23 [FIN, ACK] Seq=2177394136 Ack=2918594770 Win=30336 Len=0 TSval=479695 TSecr=473921
129	71.657262	10.0.3.2	10.0.3.1	TCP	66	23 → 40158 [FIN, ACK] Seq=2918594770 Ack=2177394137 Win=30080 Len=0 TSval=476956 TSecr=476965
130	71.657358	10.0.3.1	10.0.3.2	TCP	66	40158 → 23 [ACK] Seq=2177394137 Ack=2918594771 Win=30336 Len=0 TSval=476965 TSecr=476956

**Ans.** ใช้ FIN และ ACK เริ่มต้นจาก client ทำการส่ง FIN ไปยัง server เพื่อขอร้องให้มีการหยุดรับส่งข้อมูล จากนั้น server จะทำการส่ง FIN+ACK มายัง client เพื่อ confirm ว่าจะหยุดรับส่งข้อมูล และเมื่อ client ต้องการยืนยัน ก็ส่ง ACK ตอบกลับไป เป็นอันเสร็จสิ้นการรับส่งข้อมูล

4.8 Describe how the closing of the connection in Step 4 is different from Step 3. How long does the Telnet server wait until it closes the TCP connection?

Include relevant parts of the Wireshark output to support each of your answers.

152	98.497673	10.0.3.1	10.0.3.2	TCP	74	40160 → 23 [SYN] Seq=2143145214 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=479649 TSecr=0 WS=128
153	98.498026	10.0.3.2	10.0.3.1	TCP	74	23 → 40160 [SYN, ACK] Seq=3162997102 Ack=2143145215 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=479
154	98.498085	10.0.3.1	10.0.3.2	TCP	66	40160 → 23 [ACK] Seq=2143145215 Ack=3162997103 Win=29312 Len=0 TSval=479649 TSecr=479640
155	98.498435	10.0.3.1	10.0.3.2	TELNET	93	Telnet Data ...
156	98.498656	10.0.3.2	10.0.3.1	TCP	66	23 → 40160 [ACK] Seq=3162997103 Ack=2143145242 Win=29056 Len=0 TSval=479640 TSecr=479649
157	98.632169	10.0.3.2	10.0.3.1	TELNET	78	Telnet Data ...
158	98.632225	10.0.3.1	10.0.3.2	TCP	66	40160 → 23 [ACK] Seq=2143145242 Ack=3162997115 Win=29312 Len=0 TSval=479662 TSecr=479653
159	98.632434	10.0.3.2	10.0.3.1	TELNET	105	Telnet Data ...
160	98.632445	10.0.3.1	10.0.3.2	TCP	66	40160 → 23 [ACK] Seq=2143145242 Ack=3162997154 Win=29312 Len=0 TSval=479662 TSecr=479653
161	98.632607	10.0.3.1	10.0.3.2	TELNET	210	Telnet Data ...
162	98.632801	10.0.3.2	10.0.3.1	TCP	66	23 → 40160 [ACK] Seq=3162997154 Ack=2143145386 Win=30080 Len=0 TSval=479653 TSecr=479662
163	98.633308	10.0.3.2	10.0.3.1	TELNET	69	Telnet Data ...
164	98.633343	10.0.3.1	10.0.3.2	TELNET	69	Telnet Data ...
165	98.633761	10.0.3.2	10.0.3.1	TELNET	69	Telnet Data ...
166	98.633806	10.0.3.1	10.0.3.2	TELNET	69	Telnet Data ...
167	98.634019	10.0.3.2	10.0.3.1	TELNET	88	Telnet Data ...
168	98.664127	10.0.3.1	10.0.3.2	TCP	66	40160 → 23 [ACK] Seq=2143145392 Ack=3162997182 Win=29312 Len=0 TSval=479666 TSecr=479653
169	98.679592	10.0.3.2	10.0.3.1	TELNET	85	Telnet Data ...
170	98.679648	10.0.3.1	10.0.3.2	TCP	66	40160 → 23 [ACK] Seq=2143145392 Ack=3162997201 Win=29312 Len=0 TSval=479667 TSecr=479658

  

203	146.022593	Cisco_dc:al:81	CDP/VTP/DTP/PAGP/U...	CDP	438	Device ID: Switch Port ID: FastEthernet0/1
204	146.366050	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
205	148.375737	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
206	150.380579	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
207	151.516852	Cisco_dc:al:81	Cisco_dc:al:81	LOOP	60	Reply
208	152.381386	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
209	154.385674	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
210	156.390905	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
211	158.395477	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
212	158.639166	10.0.3.2	10.0.3.1	TELNET	68	Telnet Data ...
213	158.639259	10.0.3.1	10.0.3.2	TCP	66	40160 → 23 [ACK] Seq=2143145392 Ack=3162997203 Win=29312 Len=0 TSval=485663 TSecr=485654
214	158.639546	10.0.3.2	10.0.3.1	TELNET	101	Telnet Data ...
215	158.639575	10.0.3.1	10.0.3.2	TCP	66	40160 → 23 [ACK] Seq=2143145392 Ack=3162997238 Win=29312 Len=0 TSval=485663 TSecr=485654
216	158.647976	10.0.3.2	10.0.3.1	TCP	66	23 → 40160 [FIN, ACK] Seq=3162997238 Ack=2143145392 Win=30080 Len=0 TSval=485655 TSecr=485663
217	158.648097	10.0.3.1	10.0.3.2	TCP	66	40160 → 23 [FIN, ACK] Seq=2143145392 Ack=3162997239 Win=29312 Len=0 TSval=485664 TSecr=485655
218	158.648352	10.0.3.2	10.0.3.1	TCP	66	23 → 40160 [ACK] Seq=3162997239 Ack=2143145393 Win=30080 Len=0 TSval=485655 TSecr=485664
219	160.400707	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
220	161.528677	Cisco_dc:al:81	Cisco_dc:al:81	LOOP	60	Reply
221	162.405644	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
222	163.644045	Raspberr_6a:01:7a	Raspberr_bf:0b:d7	ARP	42	Who has 10.0.3.2? Tell 10.0.3.1
223	163.644346	Raspberr_bf:0b:d7	Raspberr_6a:01:7a	ARP	60	10.0.3.2 is at b8:27:eb:b7:0b:d7

**Ans.** ใน Step 3 นั้น เราทำการสั่งปิด telnet จาก PC1 นั่นคือ PC1 เป็นคนส่ง FIN+ACK มายัง PC2 แต่ใน Step 4 นี้เราทำการสั่งปิด telnet จาก PC1 เช่นกัน ต่างกันที่เราไม่พิมพ์อะไรเลยเมื่อมีหน้าต่างแจ้งเตือนมาให้กรอกข้อมูล ทำให้ PC2 สั่งปิด session จาก PC1 นั่นคือ PC2 เป็นคนส่ง FIN+ACK มายัง PC1 ซึ่งเวลาที่ใช้ในการรอระหว่าง PC2 สั่งปิด session จาก PC1 ประมาณ 1~2 นาที



Nattapat Yuvasuta	59070501028
Nuttawut Kuadplod	59070501029
Niti Buesamae	59070501043
Panumas Wongsang	59070501056
Supadet Jomthepmala	59070501069

4.9 How often does the TCP client try to establish a connection? How much time elapses between the repeated attempts to open a connection?

25	37.941971	10.0.3.1	10.0.3.100	TCP	74	40656 → 23 [SYN] Seq=2178847912 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=494846 TSecr=0 WS=128
26	38.037021	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
27	38.936148	10.0.3.1	10.0.3.100	TCP	74	[TCP Retransmission] 40656 → 23 [SYN] Seq=2178847912 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=49
28	39.044975	Cisco_dc:al:81	Cisco_dc:al:81	LOOP	60	Reply
29	48.898844	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
30	40.936146	10.0.3.1	10.0.3.100	TCP	74	[TCP Retransmission] 40656 → 23 [SYN] Seq=2178847912 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=49
31	42.103034	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
32	44.107558	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
33	44.946142	10.0.3.1	10.0.3.100	TCP	74	[TCP Retransmission] 40656 → 23 [SYN] Seq=2178847912 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=49
34	48.112001	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
35	48.117356	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
36	49.053052	Cisco_dc:al:81	Cisco_dc:al:81	LOOP	60	Reply
37	49.518307	Cisco_dc:al:81	CDP/VTP/DTP/PagP/U...	DTP	90	Dynamic Trunk Protocol
38	49.518381	Cisco_dc:al:81	CDP/VTP/DTP/PagP/U...	DTP	90	Dynamic Trunk Protocol
39	50.122629	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
40	52.137201	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
41	52.956158	10.0.3.1	10.0.3.100	TCP	74	[TCP Retransmission] 40656 → 23 [SYN] Seq=2178847912 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=49
42	53.503275	Cisco_dc:al:81	CDP/VTP/DTP/PagP/U...	CDP	438	Device ID: Switch Port ID: FastEthernet0/1
43	54.136796	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
44	56.136964	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
45	58.142578	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
46	59.060825	Cisco_dc:al:81	Cisco_dc:al:81	LOOP	60	Reply
47	60.146835	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
48	62.152810	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
49	64.157852	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
50	66.161825	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
51	68.166327	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
52	68.996154	10.0.3.1	10.0.3.100	TCP	74	[TCP Retransmission] 40656 → 23 [SYN] Seq=2178847912 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=49
53	69.068035	Cisco_dc:al:81	Cisco_dc:al:81	LOOP	60	Reply
54	70.176854	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
55	72.180713	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
56	74.181518	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
57	76.185934	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
58	78.191267	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
59	79.064551	Cisco_dc:al:81	Cisco_dc:al:81	LOOP	60	Reply
60	79.524831	Cisco_dc:al:81	CDP/VTP/DTP/PagP/U...	DTP	90	Dynamic Trunk Protocol
61	79.524837	Cisco_dc:al:81	CDP/VTP/DTP/PagP/U...	DTP	90	Dynamic Trunk Protocol
62	80.196286	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
63	82.201154	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
64	84.206084	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
65	86.215226	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
66	88.220105	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
67	89.071143	Cisco_dc:al:81	Cisco_dc:al:81	LOOP	60	Reply
68	90.220346	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
69	92.225486	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
70	94.230439	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
71	96.235356	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
72	98.239934	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
73	99.074306	Cisco_dc:al:81	Cisco_dc:al:81	LOOP	60	Reply
74	100.245099	Cisco_dc:al:81	Spanning-tree-(for...	STP	60	Conf. Root = 32768/1/08:cc:a7:dc:al:80 Cost = 0 Port = 0x8001
75	101.076151	10.0.3.1	10.0.3.100	TCP	74	[TCP Retransmission] 40656 → 23 [SYN] Seq=2178847912 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=50

**Ans.** The TCP client try to establish 6 times. At 1st took 1 sec, 2rd spent 2 sec, 3rd spent 4 sec, 4th spent 8 sec, 5th spent 16 sec and 6th spent 32 sec

4.10 Does the TCP client send out any control segments when it gives up on establishing a connection? Why or Why not?

**Ans.** No, it doesn't. It just tell that can't connect

4.12 What kind of segment is returned by TCP at PC2 to close this connection? How long does the process of ending the connection take?

23	31.976603	10.0.3.1	10.0.3.2	TCP	74	56534 → 80 [SYN] Seq=2921716318 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=515902 TSecr=0 WS=128
24	31.976946	10.0.3.2	10.0.3.1	TCP	60	80 → 56534 (RST, ACK) Seq=0 Ack=2921716319 Win=0 Len=0

**Ans.** RST+ACK to close this connection. It takes 0.000343 sec to close this connection.

4.13 Suppose you want to know if a certain network process is running at a given host, how would you exploit the knowledge of TCP connection establish to do it?

**Ans.** Send SYN ,If server return SYN,ACK it's mean the port is opened but if it return RST,ACK it's mean the port is closed.

Nattapat Yuvasuta	59070501028
Nuttawut Kuadplod	59070501029
Niti Buesamae	59070501043
Panumas Wongsang	59070501056
Supadet Jomthepmala	59070501069

## Part V: TCP Bulk Data Transfer

Include relevant parts of the Wireshark data to support each of your answers.

5.1 How frequent does the receiver send ACKs? Determine and explain the rule used by TCP to send ACKs.

**Ans.** They are some rule and delay of ACK segment send. Data from PC1 send very fast, so the ACK is later after the data has already been received.

ACK segment will ask for the next Sequence number that is not yet received.  
For example. If PC1 send sequence 1 with size 500, PC2 will send back ACK 501.

5.2 How many bytes of data does the receiver acknowledge in a single ACK? What is the largest amount of data acknowledged in a single ACK?

**Ans.** Typical ACK acknowledge 944,1024, 1448 bytes data (and much more).

Biggest TCP payload of 1514

ACK for sequence 28537 from seq. 48809

116	69.665866	10.0.3.1	10.0.3.2	TCP	1514 34654 → 5001 [ACK] Seq=48809 Ack=1 Win=29312 Len=1448 TSval=526890 TSecr=526881
117	69.665973	10.0.3.2	10.0.3.1	TCP	66 5001 → 34654 [ACK] Seq=1 Ack=28537 Win=86144 Len=0 TSval=526881 TSecr=526890

5.3 What is the maximum and minimum window size advertised by the receiver? How does the window size vary during the lifetime of the TCP connection?

**Ans.** Window size increase when successful sending data without error. And decrease when congestion occur.

Minimum of 28960 (when SYN ACK).

Maximum of 132480 (when receive the last segment).

61	69.663030	10.0.3.2	10.0.3.1	TCP	74 5001 → 34654 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1460 SACK_PERM=1 TSval=526880 TSecr=526890 WS=128
137	69.669046	10.0.3.2	10.0.3.1	TCP	66 5001 → 34654 [FIN, ACK] Seq=1 Ack=51202 Win=132480 Len=0 TSval=526881 TSecr=526891

5.4 Select an arbitrary ACK segment sent by PC2 to PC1 and relate it to a segment sent by PC1. How long did it take from the transmission of the segment until the ACK arrives at PC1?

**Ans.** The delay is approximately 0.00042 second.

64	69.663183	10.0.3.1	10.0.3.2	TCP	1514 34654 → 5001 [ACK] Seq=1025 Ack=1 Win=29312 Len=1448 TSval=526890 TSecr=526880
65	69.663202	10.0.3.1	10.0.3.2	TCP	1514 34654 → 5001 [ACK] Seq=2473 Ack=1 Win=29312 Len=1448 TSval=526890 TSecr=526880
66	69.663226	10.0.3.1	10.0.3.2	TCP	1514 34654 → 5001 [ACK] Seq=3921 Ack=1 Win=29312 Len=1448 TSval=526890 TSecr=526880
67	69.663244	10.0.3.1	10.0.3.2	TCP	1514 34654 → 5001 [ACK] Seq=5369 Ack=1 Win=29312 Len=1448 TSval=526890 TSecr=526880
68	69.663266	10.0.3.1	10.0.3.2	TCP	1514 34654 → 5001 [ACK] Seq=6817 Ack=1 Win=29312 Len=1448 TSval=526890 TSecr=526880
69	69.663286	10.0.3.1	10.0.3.2	TCP	1514 34654 → 5001 [ACK] Seq=8265 Ack=1 Win=29312 Len=1448 TSval=526890 TSecr=526880
70	69.663303	10.0.3.1	10.0.3.2	TCP	1514 34654 → 5001 [ACK] Seq=9713 Ack=1 Win=29312 Len=1448 TSval=526890 TSecr=526880
71	69.663324	10.0.3.1	10.0.3.2	TCP	1514 34654 → 5001 [ACK] Seq=11161 Ack=1 Win=29312 Len=1448 TSval=526890 TSecr=526880
72	69.663343	10.0.3.1	10.0.3.2	TCP	1514 34654 → 5001 [ACK] Seq=12609 Ack=1 Win=29312 Len=1448 TSval=526890 TSecr=526880
73	69.663603	10.0.3.2	10.0.3.1	TCP	66 5001 → 34654 [ACK] Seq=1 Ack=1025 Win=31104 Len=0 TSval=526880 TSecr=526890

5.5 Does the TCP sender generally transmit the maximum amount of data allowed by the advertised window size? Explain.

**Ans.** No, because it is limited by the Maximum Segment Size.

Nattapat Yuvasuta	59070501028
Nuttawut Kuadplod	59070501029
Niti Buesamae	59070501043
Panumas Wongsang	59070501056
Supadet Jomthepmala	59070501069

5.6 After the TCP sender has sent all its data, what segment does it send?

**Ans.** After all ACKs received segments, receiver send the statistic of transmission rate with PSH ACK flag back to PC1. Then Close the connection by sending FIN ACK segment.