1. Given the network conditions, how many fragments are required to transmit the 300-byte data?

<u>คำตอบ</u> 4 fragments

2. Implement manual fragmentation using Scapy and Netcat based on the given scenario. Capture the transmitted packets using Wireshark, analyze them, and explain how fragmentation was performed.

คำตอบ

ฝั่งส่งข้อมูล Code ที่ใช้

```
000
                      manual_fragmentation.py
#!/usr/bin/python3
from scapy.all import *
ID = 1000
dst_ip = "10.9.0.5"
# Fragment No.1 (Fragment offset: 0)
udp = UDP(sport=7070, dport=9090, chksum=0)
udp.len = 8 + 72 + 80 + 80 + 68
ip = IP(dst=dst_ip, id=ID, frag=0, flags=1)
payload = "A" * 71 + "\n"
pkt1 = ip/udp/payload
# Fragment No.2 (Fragment offset: (8 + 72)/8 = 10)
ip = IP(dst=dst_ip, id=ID, frag=10, flags=1)
ip.proto = 17
payload = "B" \star 79 + "\n"
pkt2 = ip/payload
# Fragment No.3 (Fragment offset: (8 + 72 + 80)/8 = 20)
ip = IP(dst=dst_ip, id=ID, frag=20, flags=1)
ip.proto = 17
payload = "C" * 79 + "\n"
pkt3 = ip/payload
# Fragment No.4 (Fragment offset: (8 + 72 + 80 + 80)/8 = 30)
ip = IP(dst=dst_ip, id=ID, frag=30, flags=0)
ip.proto = 17
payload = "D" \star 67 + "\n"
pkt4 = ip/payload
# Sending fragments
send(pkt1)
send(pkt2)
send(pkt3)
send(pkt4)
```

ฝั่งรับข้อมูลได้ script



Packet ที่ได้รับ

ข้อมูลที่จับมาได้แสดงผลผ่าน wireshark

No.	Time	Source	Destination	Protocol	Length Info							
	1 0.000000	10.9.0.5	10.0.2.3	DNS	80 Standard query 0x6701 PTR 0.0.0.0.in-addr.arpa							
	2 0.013879	10.0.2.3	10.9.0.5	DNS	80 Standard query response 0x6701 No such name PTR 0.0.0.0.in-ad							
	3 5.016651	02:42:e2:63:34:2e	02:42:0a:09:00:05	ARP	42 Who has 10.9.0.5? Tell 10.9.0.1							
	4 5.016711	02:42:0a:09:00:05	02:42:e2:63:34:2e	ARP	42 Who has 10.9.0.1? Tell 10.9.0.5							
	5 5.016731	02:42:e2:63:34:2e	02:42:0a:09:00:05	ARP	42 10.9.0.1 is at 02:42:e2:63:34:2e							
	6 5.016725	02:42:0a:09:00:05	02:42:e2:63:34:2e	ARP	42 10.9.0.5 is at 02:42:0a:09:00:05							
	7 5.584450	02:42:0a:09:00:69	Broadcast	ARP	42 Who has 10.9.0.5? Tell 10.9.0.105							
	8 5.584531	02:42:0a:09:00:05	02:42:0a:09:00:69	ARP	42 10.9.0.5 is at 02:42:0a:09:00:05							
	9 5.697032	10.9.0.105	10.9.0.5	UDP	342 7070 → 9090 Len=300							
) on interface -, id 0							
				: 02:42:0	a:09:00:05 (02:42:0a:09:00:05)							
▼ II			0.105, Dst: 10.9.0.5									
	0100 = Vers											
0101 = Header Length: 20 bytes (5)												
» Differentiated Services Field: 0x00 (DSCP: CSO, ECN: Not-ECT)												
	Total Length: 32											
Identification: 0x03e8 (1000)												
-	▶ Flags: 0x0000											
	Fragment offset: 0											
Time to live: 64												
Protocol: UDP (17) Header checksum: 0x613e [validation disabled] [Header checksum status: Unverified]												
								Source: 10.9.0.:	105			
								Destination: 10	9.0.5			
User Datagram Protocol, Src Port: 7070, Dst Port: 9090												
Source Port: 7070												
	Destination Port	: 9090										
	Length: 308											
	[Checksum: [miss	sing]]										
	Checksum Status	s: Not present]										
	Stream index: :	.1										
-	[Timestamps]	-										
	ta (300 bytes)											
→ D;												
₩ Da	Data: 414141414	41414141414141414141	414141414141414141									

<u>คำอธิบาย</u>

payload จะแยกออกมาเป็น 4 fragments โดยจะมีขนาดเท่ากับ:

8+72	80	80	68

โดย UDP Length คำนวณจาก payload 300bytes + UDP Header 8bytes = 308bytes แล้วเอา 308/80 = 3.85 เราปัดขึ้นเป็น 4 จึงได้ทั้งหมด 4 fragments

ใน fragment 1 จะ มี UDP Header รวมอยู่ด้วย 8bytes

fragment 2 และ 3 จะมี 80bytes

feagment 4 ซึ่งเป็นตัวสุดท้ายจะเป็นจำนวน 68bytes

โดยจะเซ็ท id ให้กับทุก fragments คือ 1000

และ ที่ fragment 2,3,4 จะมี ip protocol คือ 17 (UDP)

Fragment 1 จะมีค่า offset คือ 0 จากการเป็นตัวแรก และมีค่า flags คือ MF (1)

Feagment 2 มีค่า offset คือ 10 มีค่า flags คือ MF (1) วิธีการคำนวณ offset ของ fragment 2 คือ (8+72)/8=10 โดย 8 คือ UDP Header 72 คือ payload ของ flagment 1 และ หาร 8 มาจาก 3bit ที่ เป็น flags (2^3)

Fragment 3 มีค่า offset คือ 20 มีค่า flags คือ MF (1) วิธีการคำนวณ offset ของ fragment 3 คือ (8+72+80)/8=20

Fragment 4 มีค่า offset คือ 30 มีค่า flags คือ DF (0) วิธีการคำนวณ offset ของ fragment 4 คือ (8+72+80+80)/8=30 เนื่องจาก fragment 4 เป็นตัวสุดท้ายค่า flags จึงเป็น 0