Scenario

Consider a network with a Maximum Transmission Unit (MTU) of 80 bytes. A sender needs to transmit data of size 300 bytes using the UDP protocol. The UDP header is 8 bytes, and the IP header is 20 bytes.

- Each IP packet includes the IP header (20 bytes).
- Each UDP packet includes the UDP header (8 bytes).
- The payload size of each fragment must be adjusted according to the given MTU.

Questions

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

จาก MTU 80 bytes ต้องการส่ง Data 300 bytes ต้องแบ่งส่งทั้งหมด 4 fragments

Fragment 1: 8 (UDP) + 72 (Data)

Fragment 2 - 3: 80 (Data) each

Fragment 4: 68 (Data)

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.

Code:

```
#!/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" * 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" * 67 + "\n"
pkt4 = ip/payload

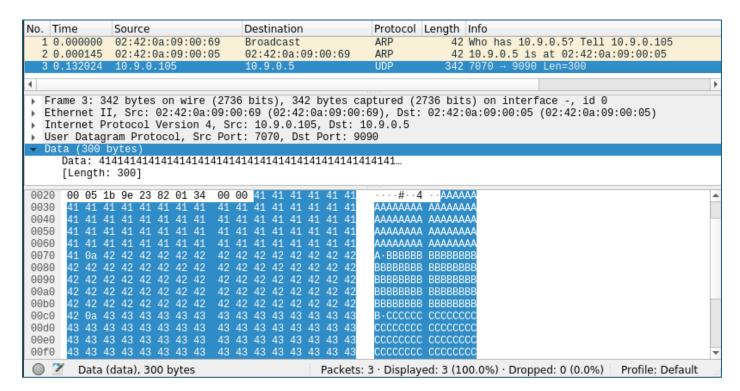
# Sending fragments
send (pkt2)
send (pkt2)
send (pkt3)
send (pkt4)
```

Attacker:

```
root@21ea68d7f732:/volumes# ./fragment_spoof2.py
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
.
Sent 1 packets.
.
root@21ea68d7f732:/volumes#
```

Victim:

Wireshark:



Data 300 bytes ถูกตัดแบ่งส่งเป็น 4 fragments โดย

- fragment ที่ 1 ส่ง UDP Header 8 bytes กับ data 72 bytes (A 71 ตัว + \n) รวมเป็น 80 bytes
- fragment ที่ 2 และ 3 ส่ง data 80 bytes (fragment 2: B 79 ตัว + \n และ fragment 3: C 79 ตัว + \n)
- fragment ที่ 4 ส่ง data ส่วนที่เหลือ 68 bytes (D 67 ตัว + \n)

ทุก fragment มี IP header ระบุปลายทาง ระบุ ID เป็น 1000 และเลขลำดับ fragment offset ที่ fragment นั้นๆ เริ่มต้น เนื่องจาก unit ของ offset มีค่าเป็น 8 bytes การระบุตำแหน่งจึงต้องหารค่าด้วย 8 คือ

- fragment ที่ 1 เริ่มที่ 0
- fragment ที่ 2 เริ่มที่ 80 bytes ค่า offset คือ 80/8 = 10
- fragment ที่ 2 เริ่มที่ 160 bytes ค่า offset คือ 160/8 = 20
- fragment ที่ 2 เริ่มที่ 240 bytes ค่า offset คือ 240/8 = 30

ปลายทาง: 10.9.0.5 จะรอรับทุก fragment ให้ครบก่อน จึงจะแสดงผลบน terminal