

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

คำตอบ 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

```

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" * 79 + "\n"
pkt2 = ip/payload

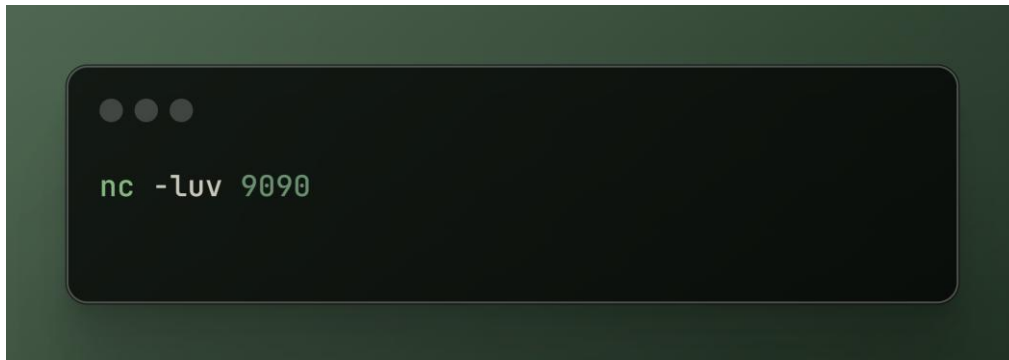
# Fragment No.3 (Fragment offset: (8 + 72 + 80)/8 = 20)
ip = IP(dst=dst_ip, id=ID, frag=20, flags=1)
ip.proto = 17
payload = "C" * 79 + "\n"
pkt3 = ip/payload

# 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(pkt1)
send(pkt2)
send(pkt3)
send(pkt4)

```

ฝั่งรับข้อมูลได้ script



Packet ที่ได้รับ

```
root@7a518blabe5b:/# nc -l -p 9090
Bound on 0.0.0.0 9090
Connection received on attacker-10.9.0.105-10.9.0.0 7070
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
DDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
```

ข้อมูลที่จับมาได้แสดงผลผ่าน wireshark

[illegible]

คำอธิบาย

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 จะมี UDP Header รวมอยู่ด้วย 8bytes

fragment 2 และ 3 จะมี 80bytes

fragment 4 ซึ่งเป็นตัวสุดท้ายจะเป็นจำนวน 68bytes

โดยจะเซต 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)

Fragment 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 fragment 1 และหาร 8 มาจาก 3bit ที่เป็น 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