UDP Flood

Un **UDP flood** è un attacco di **Denial of Service (DoS)** che utilizza il protocollo UDP per sovraccaricare un sistema target.
L'attacco consiste nell'invio massivo di pacchetti UDP verso una o più porte della vittima.
Poiché UDP è un protocollo senza connessione, il sistema deve elaborare ogni pacchetto ricevuto.
Se la porta è chiusa, il target genera risposte ICMP, consumando ulteriori risorse.
Se la porta è aperta, il servizio tenta di gestire i pacchetti, aumentando il carico di lavoro.
UDP permette lo spoofing dell'indirizzo IP, rendendo difficile l'identificazione dell'attaccante.
Il traffico eccessivo può saturare banda, CPU e stack di rete.
Esistono varianti come UDP flood diretto e attacchi di amplificazione.
Le conseguenze includono rallentamenti o indisponibilità dei servizi.

Il programma simula un UDP Flood progettato per inviare pacchetti dati a una singola porta di una macchina target.
All'avvio, l'utente inserisce l'indirizzo IP di destinazione, la porta UDP target e la quantità di dati da inviare espressa in kilobyte.
Il codice utilizza il modulo `socket` per creare un socket UDP IPv4, impostando un timeout per evitare blocchi in attesa di risposta.
La dimensione di ciascun pacchetto è fissata a 1024 byte, equivalenti a 1 KB.
I dati inviati sono generati in modo casuale tramite il modulo `random`, simulando traffico binario realistico.
Per ogni kilobyte richiesto, il programma costruisce un payload casuale e lo invia alla porta specificata mediante la funzione `sendto()`.
Essendo UDP un protocollo senza connessione, l'invio non garantisce né consegna né risposta.
Terminato l'invio dei pacchetti, il programma tenta di ricevere una risposta dal target.
Se viene ricevuto un messaggio, la porta è considerata aperta e attiva.
In assenza di risposta entro il tempo limite, la porta viene classificata come "open o filtered".
Eventuali errori di rete o messaggi ICMP indicano una porta chiusa o non raggiungibile.
Il codice gestisce correttamente le eccezioni per evitare arresti anomali.
Al termine dell'esecuzione, il socket viene chiuso per liberare le risorse di sistema.