

ESERCIZIO.

Esercizio del Giorno (potete aiutarvi con ChatGPT)

Requisiti del Programma:

1. Input dell'IP Target:

- Il programma deve richiedere all'utente di inserire l'IP della macchina target.

2. Input della Porta Target:

- Il programma deve richiedere all'utente di inserire la porta UDP della macchina target.

3. Costruzione del Pacchetto:

- La grandezza dei pacchetti da inviare deve essere di 1 KB per pacchetto.
- Suggerimento: per costruire il pacchetto da 1 KB, potete utilizzare il modulo random per la generazione di byte casuali.

4. Numero di Pacchetti da Inviare:

- Il programma deve chiedere all'utente quanti pacchetti da 1 KB inviare.

Primo codice semplice.

```
C > EsercizioS2L3.py > main > target_port
1  import socket
2  import random
3
4  def main():
5      target_ip = input("Inserisci l'IP della macchina target: ")
6      target_port = int(input("Inserisci la porta UDP (per esempio 137)"))
7      num_packets = int(input("Quanti pacchetti da 1 KB vuoi inviare? "))
8      sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
9      packet_size = 1024
10     packet_data = bytearray(random.getrandbits(8) for _ in range(packet_size))
11     for i in range(num_packets):
12         try:
13             sock.sendto(packet_data, (target_ip, target_port))
14             print(f"Pacchetto {i+1} inviato.")
15         except Exception as e:
16             print(f"Errore durante l'invio del pacchetto {i+1}: {e}")
17     sock.close()
18
19 if __name__ == "__main__":
20     main()
```

Codice un po' complesso chiesto a ChatGPT.

Ho modificato il primo "if" in quanto ChatGPT aveva considerato tutte le 65535 porte ma io volevo solo la 137.

```
Esercizio.py > ...
import socket
import random
import sys

def main():
    target_ip = input('Inserisci IP della macchina target: ')
    while True:
        try:
            target_port = int(input('Inserisci la porta UDP (ad esempio 137): '))
            if 0 <= target_port == 137:
                break
            else:
                print('Per favore inserisci un numero di porta valido:')
        except ValueError:
            print("Per favore, inserisci un numero valido.")

    while True:
        try:
            num_packets = int(input("Quanti pacchetti da 1 KB vuoi inviare? "))
            if num_packets > 0:
                break
            else:
                print("Inserisci un numero positivo.")
        except ValueError:
            print("Per favore inserisci un numero valido.")

    packet_data = bytearray(random.getrandbits(8) for _ in range(1024))
    sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    try:
        for i in range(num_packets):
            try:
                sock.sendto(packet_data, (target_ip, target_port))
                print(f'Pacchetto {i+1} inviato.')
            except Exception as e:
                print(f'Errore durante l'invio del pacchetto {i+1}: {e}')
                sys.exit()
        print("Invio completato.")
    finally:
        sock.close()
    if __name__ == "__main__":
        main()
```

Ho cercato di modificare il codice aggiungendo le porte UDP che potevano essere inserite.

```
Esercizio.py > main
import socket
import random
import sys

def main():
    target_ip = input('Inserisci IP della macchina target: ')
    target_port_allow = [7, 8, 9, 19, 67, 68, 69, 123, 137, 138, 161, 162, 445, 514, 554, 631]
    while True:
        try:
            target_port = int(input('Inserisci la porta UDP (ad esempio 137): '))
            if target_port in target_port_allow:
                break
            else:
                print('Per favore inserisci un numero di porta valido:')
        except ValueError:
            print("Per favore, inserisci un numero valido.")

    while True:
        try:
            num_packets = int(input("Quanti pacchetti da 1 KB vuoi inviare? "))
            if num_packets > 0:
                break
            else:
                print("Inserisci un numero positivo.")
        except ValueError:
            print("Per favore inserisci un numero valido.")

    packet_data = bytearray(random.getrandbits(8) for _ in range(1024))
    sock = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
    try:
        for i in range(num_packets):
            try:
                sock.sendto(packet_data, (target_ip, target_port))
                print(f'Pacchetto {i+1} inviato.')
            except Exception as e:
                print(f'Errore durante l'invio del pacchetto {i+1}: {e}')
                sys.exit()
        print("Invio completato.")
    finally:
        sock.close()
    if __name__ == "__main__":
        main()
```