

SPRING 2019

CS479 Project Report

Introduction to Cyber Security

Sara Jammal

Selin Özdaş

Berke Soysal

Cevat Barış Yılmaz



Project Type: DDoS Attack Tool

User Interface

The image displays three separate Tkinter window screenshots, each titled 'tk', representing different parts of a DDoS Attack Tool's user interface.

Window 1: DDoS Attacker

This window serves as the main menu, featuring a vertical stack of buttons for selecting different attack types:

- UDP Flood
- ICMP Flood
- Syn Flood
- Ping of Death
- HTTP Flood
- RST/FIN attack
- DNS Flood
- Synonymous IP Attack

Window 2: HTTP Flooder

This window is for configuring an HTTP Flood attack. It includes the following fields and buttons:

- Destination IP:** A text input field containing the value '192.168.1.23'.
- Port:** A text input field containing the value '80'.
- Number of packages:** A text input field containing the value '50'.
- Attack:** A button at the bottom to initiate the attack.

Window 3: Udp Flooder

This window is for configuring a UDP Flood attack. It includes the following fields and buttons:

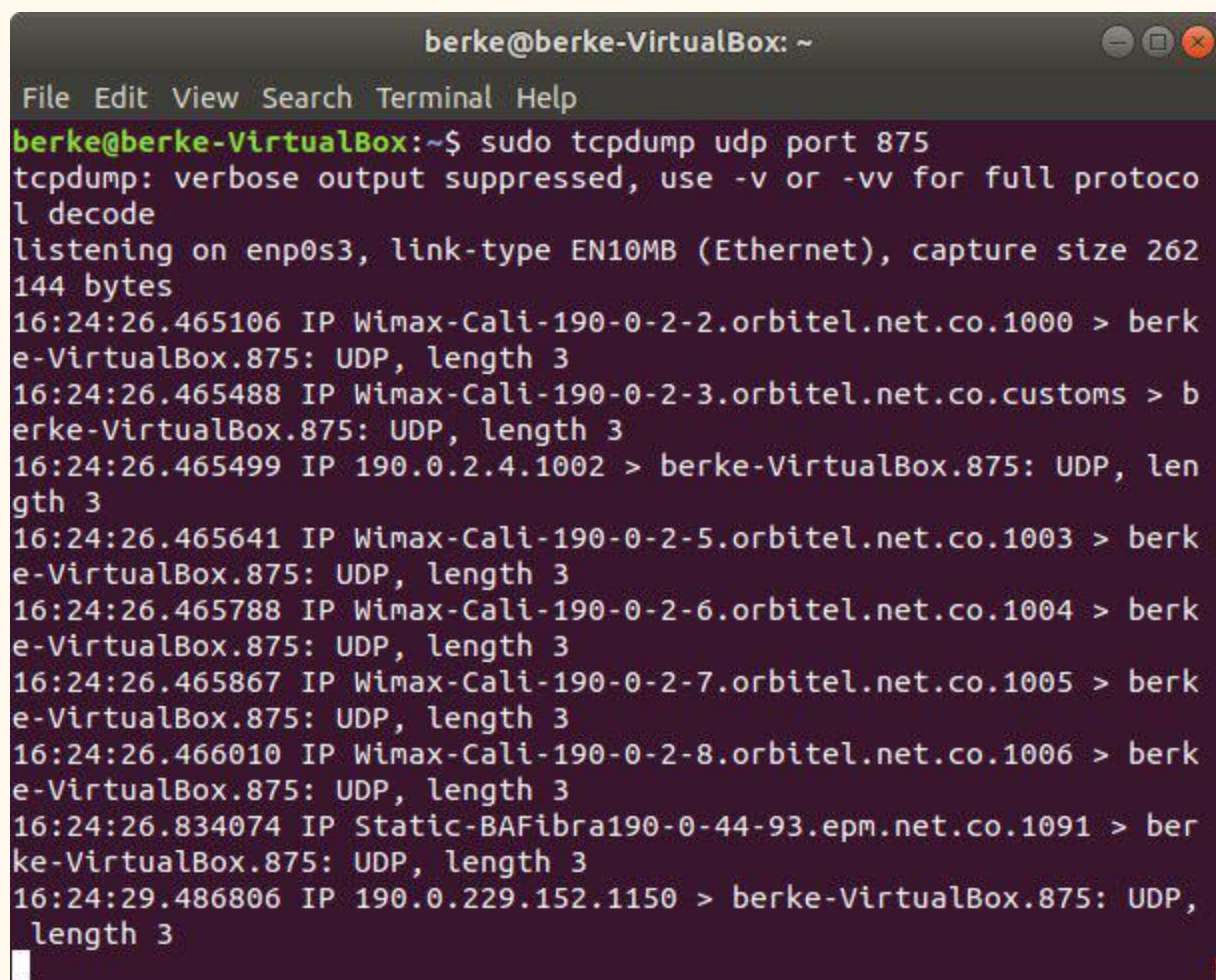
- Destination IP:** An empty text input field.
- Port:** An empty text input field.
- Attack:** A button to initiate the attack.
- Back to Home:** A button to return to the main menu.

Attack Types

For our DDoS attack tool, we have implemented the following attacks. You can see the indicator of compromise screenshots of the attacks after the description of the attack:

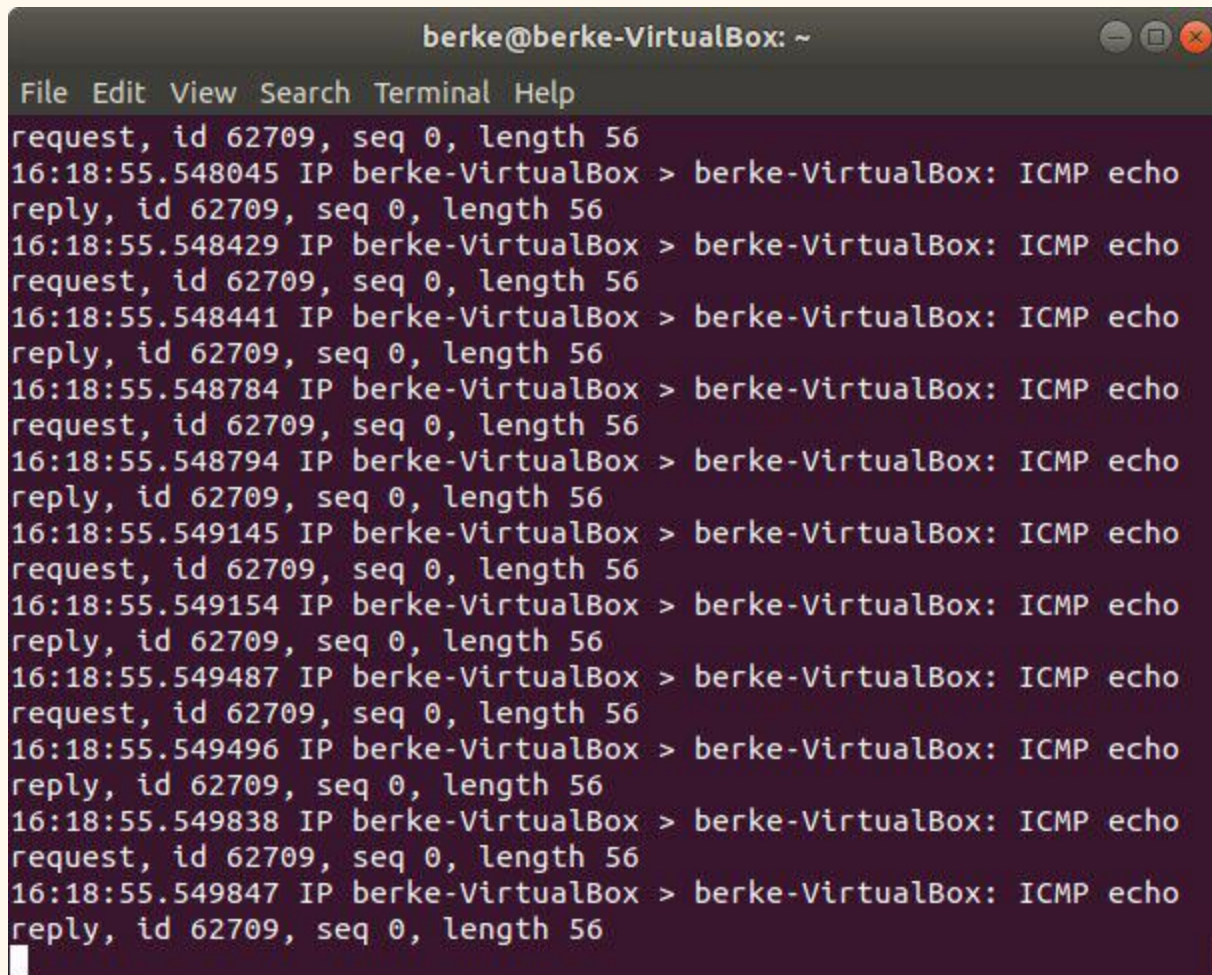
UDP Flood:

This attacks are created with sending a server a flood of User Datagram Protocol(UDP) packets.

A screenshot of a terminal window titled 'berke@berke-VirtualBox: ~'. The terminal shows the command 'sudo tcpdump udp port 875' being executed. The output indicates that verbose output is suppressed and that the tool is listening on the network interface 'enp0s3' with a capture size of 262144 bytes. Subsequently, several lines of network traffic are displayed, each representing a UDP packet from various IP addresses to the destination 'berke-VirtualBox.875'. The packets have a length of 3 bytes. The source IP addresses include 'Wimax-Cali-190-0-2-2.orbitel.net.co.1000', 'Wimax-Cali-190-0-2-3.orbitel.net.co.customs', '190.0.2.4.1002', 'Wimax-Cali-190-0-2-5.orbitel.net.co.1003', 'Wimax-Cali-190-0-2-6.orbitel.net.co.1004', 'Wimax-Cali-190-0-2-7.orbitel.net.co.1005', 'Wimax-Cali-190-0-2-8.orbitel.net.co.1006', 'Static-BAFibra190-0-44-93.epm.net.co.1091', and '190.0.229.152.1150'. The terminal window has a standard Linux-style title bar with minimize, maximize, and close buttons.

ICMP(ping) Flood:

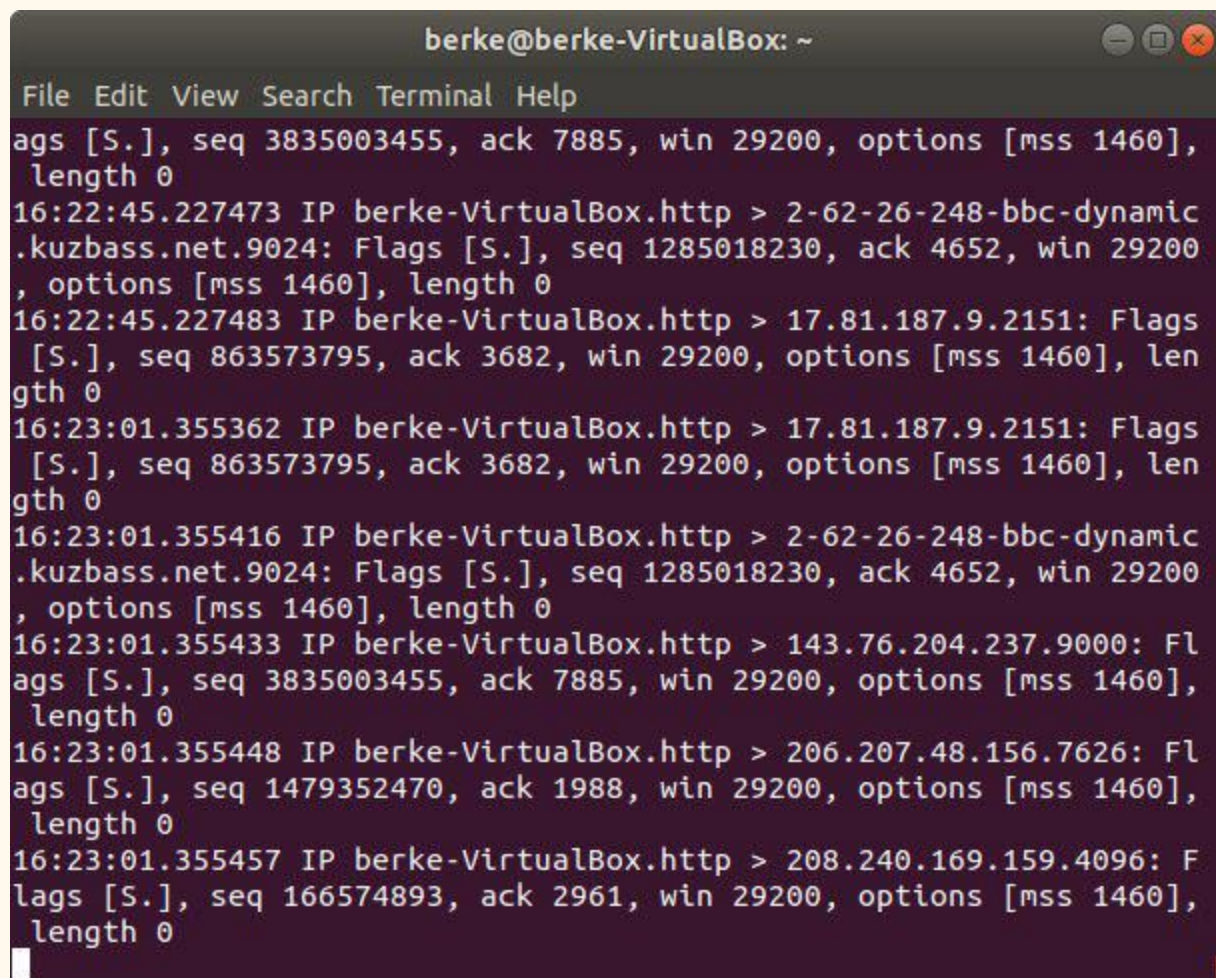
This attack is similar to above one, an ICMP attack is executed by sending floods of ICMP Echo Request(ping) packets without waiting for reply, that results in a slow response for other users of the server.

A screenshot of a terminal window titled 'berke@berke-VirtualBox: ~'. The terminal shows a series of ICMP Echo Request and Reply packets. The requests are sent from 'berke-VirtualBox' to 'berke-VirtualBox' with ID 62709 and sequence 0. The replies are received from 'berke-VirtualBox' with the same ID and sequence. The timestamps for each packet are listed on the left side of the terminal output.

```
berke@berke-VirtualBox: ~
File Edit View Search Terminal Help
request, id 62709, seq 0, length 56
16:18:55.548045 IP berke-VirtualBox > berke-VirtualBox: ICMP echo
reply, id 62709, seq 0, length 56
16:18:55.548429 IP berke-VirtualBox > berke-VirtualBox: ICMP echo
request, id 62709, seq 0, length 56
16:18:55.548441 IP berke-VirtualBox > berke-VirtualBox: ICMP echo
reply, id 62709, seq 0, length 56
16:18:55.548784 IP berke-VirtualBox > berke-VirtualBox: ICMP echo
request, id 62709, seq 0, length 56
16:18:55.548794 IP berke-VirtualBox > berke-VirtualBox: ICMP echo
reply, id 62709, seq 0, length 56
16:18:55.549145 IP berke-VirtualBox > berke-VirtualBox: ICMP echo
request, id 62709, seq 0, length 56
16:18:55.549154 IP berke-VirtualBox > berke-VirtualBox: ICMP echo
reply, id 62709, seq 0, length 56
16:18:55.549487 IP berke-VirtualBox > berke-VirtualBox: ICMP echo
request, id 62709, seq 0, length 56
16:18:55.549496 IP berke-VirtualBox > berke-VirtualBox: ICMP echo
reply, id 62709, seq 0, length 56
16:18:55.549838 IP berke-VirtualBox > berke-VirtualBox: ICMP echo
request, id 62709, seq 0, length 56
16:18:55.549847 IP berke-VirtualBox > berke-VirtualBox: ICMP echo
reply, id 62709, seq 0, length 56
```


SYN Flood:

The tool is able to make SYN Flood attacks. It will exploit the weakness of TCP connection sequence ("three-way-handshake"). It sends numerous SYN packets to server, but it does not respond to server's SYN-ACK packets. As you can see there are SYN packets coming to the server with spoofed IP's.

A screenshot of a terminal window titled 'berke@berke-VirtualBox: ~'. The terminal displays a series of network traffic logs for a SYN flood attack. The logs show incoming SYN packets from various spoofed IP addresses to the local host (17.81.187.9). The packets have sequence numbers starting from 863573795 and increasing. The window has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'.

```
berke@berke-VirtualBox: ~
File Edit View Search Terminal Help
ags [S.], seq 3835003455, ack 7885, win 29200, options [mss 1460],
length 0
16:22:45.227473 IP berke-VirtualBox.http > 2-62-26-248-bbc-dynamic
.kuzbass.net.9024: Flags [S.], seq 1285018230, ack 4652, win 29200
, options [mss 1460], length 0
16:22:45.227483 IP berke-VirtualBox.http > 17.81.187.9.2151: Flags
[S.], seq 863573795, ack 3682, win 29200, options [mss 1460], len
gth 0
16:23:01.355362 IP berke-VirtualBox.http > 17.81.187.9.2151: Flags
[S.], seq 863573795, ack 3682, win 29200, options [mss 1460], len
gth 0
16:23:01.355416 IP berke-VirtualBox.http > 2-62-26-248-bbc-dynamic
.kuzbass.net.9024: Flags [S.], seq 1285018230, ack 4652, win 29200
, options [mss 1460], length 0
16:23:01.355433 IP berke-VirtualBox.http > 143.76.204.237.9000: Fl
ags [S.], seq 3835003455, ack 7885, win 29200, options [mss 1460],
length 0
16:23:01.355448 IP berke-VirtualBox.http > 206.207.48.156.7626: Fl
ags [S.], seq 1479352470, ack 1988, win 29200, options [mss 1460],
length 0
16:23:01.355457 IP berke-VirtualBox.http > 208.240.169.159.4096: F
lags [S.], seq 166574893, ack 2961, win 29200, options [mss 1460],
length 0
```

Ping of Death:

A ping of death attack is carried out by sending malicious pings to a computer. It will send a computer a packet that bigger than the maximum packet length of IP packet, 65,535 bytes. This will result in memory overflow for victim computer.

```
berke@berke-VirtualBox: ~  
File Edit View Search Terminal Help  
quest, id 0, seq 0, length 1480  
16:28:34.198071 IP 178.214.98.253 > berke-VirtualBox: icmp  
16:28:34.199122 IP 178.214.98.253 > berke-VirtualBox: icmp  
16:28:34.200113 IP 178.214.98.253 > berke-VirtualBox: icmp  
16:28:34.201137 IP 178.214.98.253 > berke-VirtualBox: icmp  
16:28:34.202097 IP 178.214.98.253 > berke-VirtualBox: icmp  
16:28:34.203045 IP 178.214.98.253 > berke-VirtualBox: icmp  
16:28:34.312913 IP 111.60.67.111 > berke-VirtualBox: ICMP echo req  
uest, id 0, seq 0, length 1480  
16:28:34.313338 IP 111.60.67.111 > berke-VirtualBox: icmp  
16:28:34.314781 IP 111.60.67.111 > berke-VirtualBox: icmp  
16:28:34.316124 IP 111.60.67.111 > berke-VirtualBox: icmp  
16:28:34.317437 IP 111.60.67.111 > berke-VirtualBox: icmp  
16:28:34.318804 IP 111.60.67.111 > berke-VirtualBox: icmp  
16:28:34.321645 IP 111.60.67.111 > berke-VirtualBox: icmp  
16:28:34.815324 IP 155.40.149.214 > berke-VirtualBox: ICMP echo re  
quest, id 0, seq 0, length 1480  
16:28:34.816858 IP 155.40.149.214 > berke-VirtualBox: icmp  
16:28:34.817961 IP 155.40.149.214 > berke-VirtualBox: icmp  
16:28:34.819432 IP 155.40.149.214 > berke-VirtualBox: icmp  
16:28:34.820709 IP 155.40.149.214 > berke-VirtualBox: icmp  
16:28:34.821698 IP 155.40.149.214 > berke-VirtualBox: icmp  
16:28:34.822747 IP 155.40.149.214 > berke-VirtualBox: icmp
```


HTTP Flood:

Attacker sends legitimate HTTP Get/Post Request, and it will force the server to allocate maximum resources possible with every request it sends.

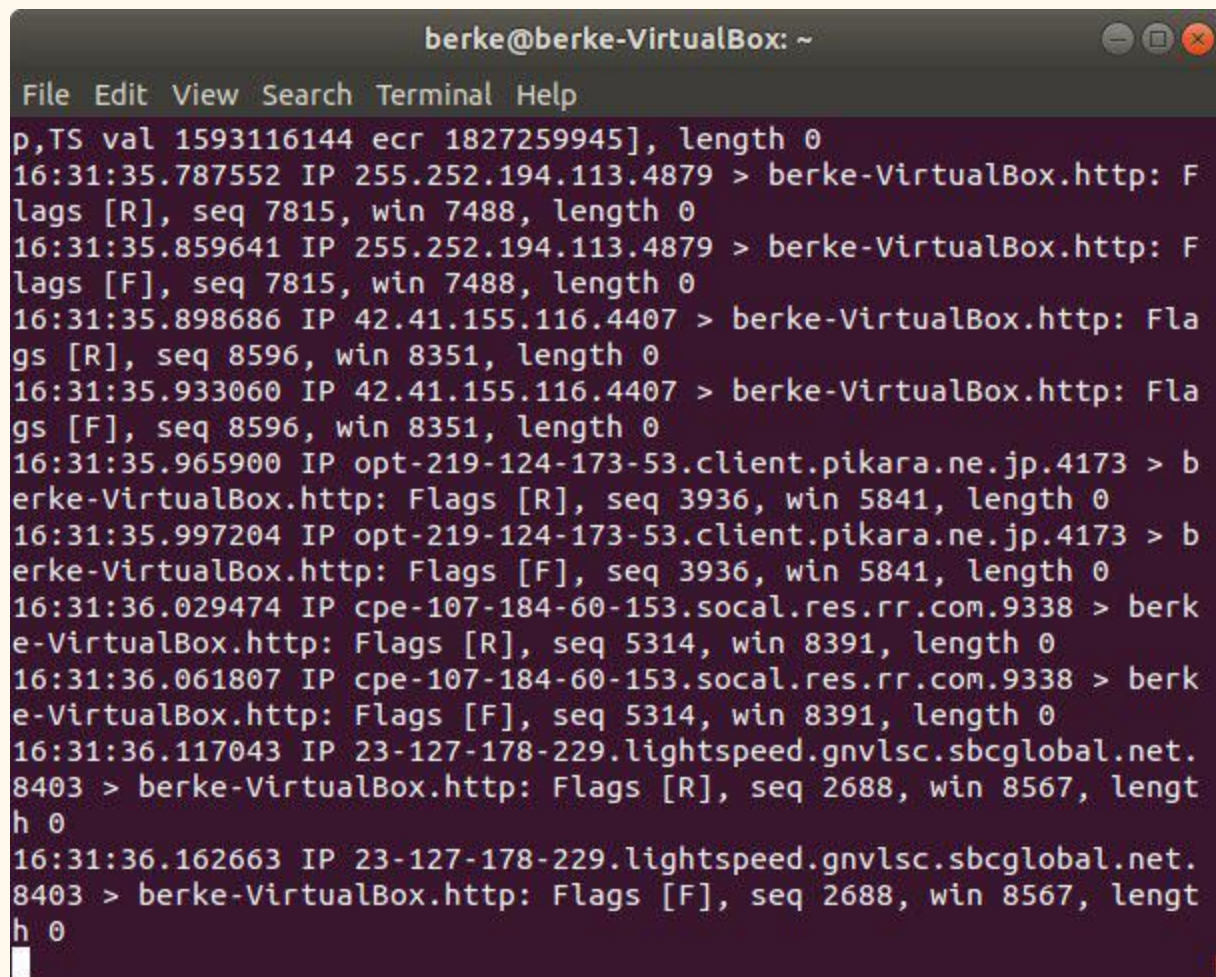
```

berke@berke-VirtualBox: ~
File Edit View Search Terminal Help
.1\n" 400 0 "-" "-"
192.168.43.225 - - [15/May/2019:16:27:11 +0300] "GET /YLH[\ HTTP/
1.1\n" 400 0 "-" "-"
192.168.43.225 - - [15/May/2019:16:27:11 +0300] "GET /+1,U9 HTTP/1
.1\n" 400 0 "-" "-"
192.168.43.225 - - [15/May/2019:16:27:11 +0300] "GET /m*D.s HTTP/1
.1\n" 400 0 "-" "-"
192.168.43.225 - - [15/May/2019:16:27:11 +0300] "GET /PnHq1 HTTP/1
.1\n" 400 0 "-" "-"
192.168.43.225 - - [15/May/2019:16:27:11 +0300] "GET /J:'~\ HTTP/
1.1\n" 400 0 "-" "-"
192.168.43.225 - - [15/May/2019:16:27:11 +0300] "GET /G)&Z, HTTP/1
.1\n" 400 0 "-" "-"
192.168.43.225 - - [15/May/2019:16:27:11 +0300] "GET /yGj/w HTTP/1
.1\n" 400 0 "-" "-"
192.168.43.225 - - [15/May/2019:16:27:11 +0300] "GET /4Rbix HTTP/1
.1\n" 400 0 "-" "-"
192.168.43.225 - - [15/May/2019:16:27:11 +0300] "GET /=ZNG[ HTTP/1
.1\n" 400 0 "-" "-"
192.168.43.225 - - [15/May/2019:16:27:11 +0300] "GET /V\\6Y. HTTP/
1.1\n" 400 0 "-" "-"
192.168.43.225 - - [15/May/2019:16:27:11 +0300] "GET /=d(g7 HTTP/1
.1\n" 400 0 "-" "-"
berke@berke-VirtualBox:~$

```

RST/FIN Flood:

In an RST/FIN Flood attack, attacker sends a large number of spoofed RST/FIN packets that do not belong to any session on the target server.



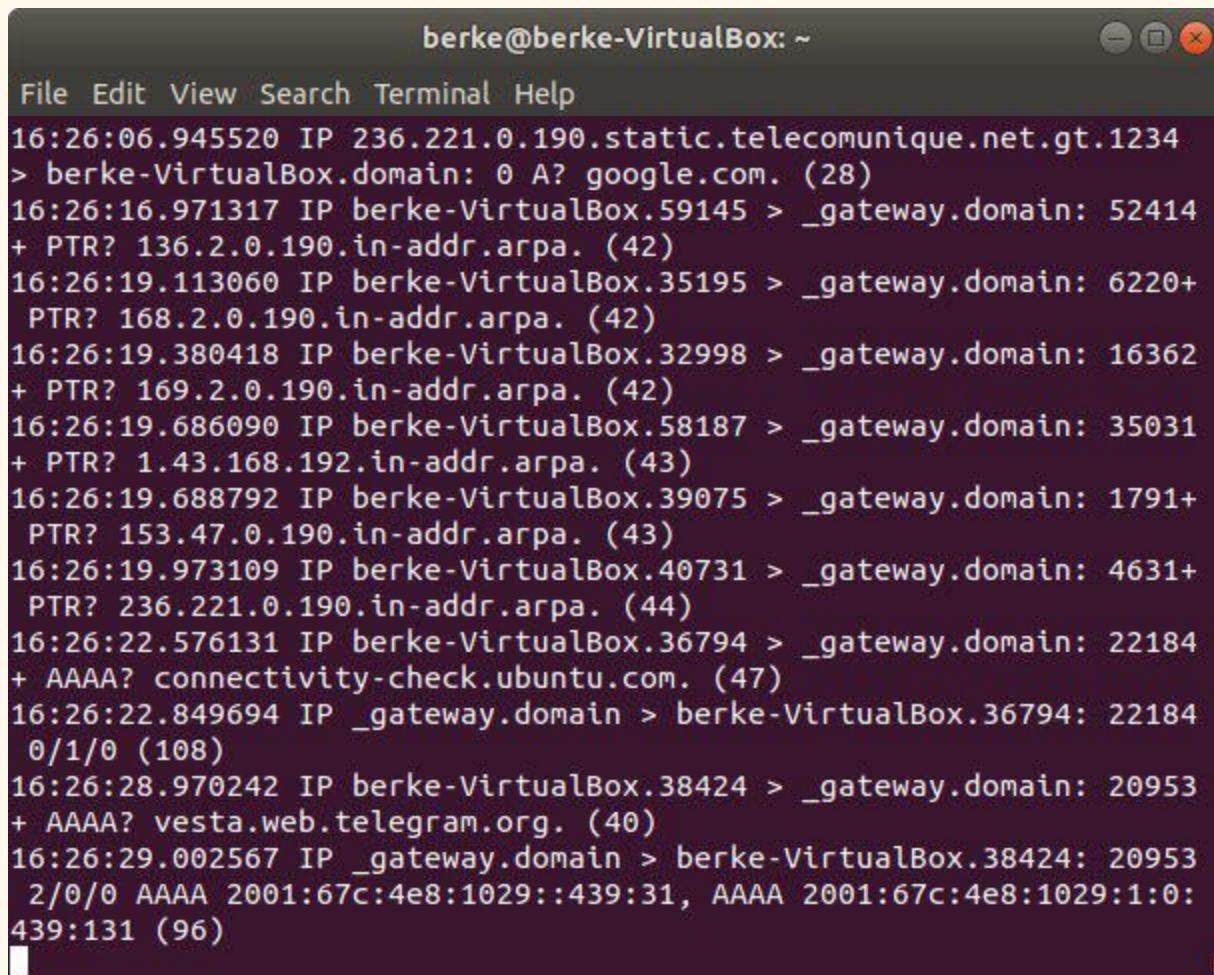
```

berke@berke-VirtualBox: ~
File Edit View Search Terminal Help
p,TS val 1593116144 ecr 1827259945], length 0
16:31:35.787552 IP 255.252.194.113.4879 > berke-VirtualBox.http: F
lags [R], seq 7815, win 7488, length 0
16:31:35.859641 IP 255.252.194.113.4879 > berke-VirtualBox.http: F
lags [F], seq 7815, win 7488, length 0
16:31:35.898686 IP 42.41.155.116.4407 > berke-VirtualBox.http: Fla
gs [R], seq 8596, win 8351, length 0
16:31:35.933060 IP 42.41.155.116.4407 > berke-VirtualBox.http: Fla
gs [F], seq 8596, win 8351, length 0
16:31:35.965900 IP opt-219-124-173-53.client.pikara.ne.jp.4173 > b
erke-VirtualBox.http: Flags [R], seq 3936, win 5841, length 0
16:31:35.997204 IP opt-219-124-173-53.client.pikara.ne.jp.4173 > b
erke-VirtualBox.http: Flags [F], seq 3936, win 5841, length 0
16:31:36.029474 IP cpe-107-184-60-153.socal.res.rr.com.9338 > berk
e-VirtualBox.http: Flags [R], seq 5314, win 8391, length 0
16:31:36.061807 IP cpe-107-184-60-153.socal.res.rr.com.9338 > berk
e-VirtualBox.http: Flags [F], seq 5314, win 8391, length 0
16:31:36.117043 IP 23-127-178-229.lightspeed.gnvlsc.sbcglobal.net.
8403 > berke-VirtualBox.http: Flags [R], seq 2688, win 8567, lengt
h 0
16:31:36.162663 IP 23-127-178-229.lightspeed.gnvlsc.sbcglobal.net.
8403 > berke-VirtualBox.http: Flags [F], seq 2688, win 8567, lengt
h 0

```


DNS Flood:

Flood a DNS server with DNS requests to prevent it from serving actual users properly.

A screenshot of a terminal window titled 'berke@berke-VirtualBox: ~'. The terminal shows a series of DNS requests and responses, indicating a flood attack. The requests are from various IP addresses to the 'berke-VirtualBox' domain, asking for 'A' records for 'google.com' and '_gateway.domain'. The responses show the domain's IP address (236.221.0.190) and the gateway's IP address (136.2.0.190). The terminal also shows a response for a PTR record for '136.2.0.190.in-addr.arpa'. The terminal window has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The terminal output is as follows:

```
berke@berke-VirtualBox: ~
File Edit View Search Terminal Help
16:26:06.945520 IP 236.221.0.190.static.telecomunique.net.gt.1234
> berke-VirtualBox.domain: 0 A? google.com. (28)
16:26:16.971317 IP berke-VirtualBox.59145 > _gateway.domain: 52414
+ PTR? 136.2.0.190.in-addr.arpa. (42)
16:26:19.113060 IP berke-VirtualBox.35195 > _gateway.domain: 6220+
PTR? 168.2.0.190.in-addr.arpa. (42)
16:26:19.380418 IP berke-VirtualBox.32998 > _gateway.domain: 16362
+ PTR? 169.2.0.190.in-addr.arpa. (42)
16:26:19.686090 IP berke-VirtualBox.58187 > _gateway.domain: 35031
+ PTR? 1.43.168.192.in-addr.arpa. (43)
16:26:19.688792 IP berke-VirtualBox.39075 > _gateway.domain: 1791+
PTR? 153.47.0.190.in-addr.arpa. (43)
16:26:19.973109 IP berke-VirtualBox.40731 > _gateway.domain: 4631+
PTR? 236.221.0.190.in-addr.arpa. (44)
16:26:22.576131 IP berke-VirtualBox.36794 > _gateway.domain: 22184
+ AAAA? connectivity-check.ubuntu.com. (47)
16:26:22.849694 IP _gateway.domain > berke-VirtualBox.36794: 22184
0/1/0 (108)
16:26:28.970242 IP berke-VirtualBox.38424 > _gateway.domain: 20953
+ AAAA? vesta.web.telegram.org. (40)
16:26:29.002567 IP _gateway.domain > berke-VirtualBox.38424: 20953
2/0/0 AAAA 2001:67c:4e8:1029::439:31, AAAA 2001:67c:4e8:1029:1:0:
439:131 (96)
```

Synonymous IP attack(LAND Attack):

A large number of TCP-SYN packets carrying the target server's Source IP and Destination IP are sent to the target server. The host server starts using additional system resources (RAM, CPU, etc.) to process each of the packets.

```
berke@berke-VirtualBox: ~  
File Edit View Search Terminal Help  
16:28:34.819432 IP 155.40.149.214 > berke-VirtualBox: icmp  
16:28:34.820709 IP 155.40.149.214 > berke-VirtualBox: icmp  
16:28:34.821698 IP 155.40.149.214 > berke-VirtualBox: icmp  
16:28:34.822747 IP 155.40.149.214 > berke-VirtualBox: icmp  
^C  
56 packets captured  
4100 packets received by filter  
4044 packets dropped by kernel  
berke@berke-VirtualBox:~$ sudo tcpdump port 80  
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode  
listening on enp0s3, link-type EN10MB (Ethernet), capture size 262144 bytes  
16:31:01.244651 IP berke-VirtualBox.5506 > berke-VirtualBox.http:  
Flags [S], seq 4317, win 2834, length 0  
16:31:01.288281 IP berke-VirtualBox.6993 > berke-VirtualBox.http:  
Flags [S], seq 4847, win 3374, length 0  
16:31:01.400389 IP berke-VirtualBox.8439 > berke-VirtualBox.http:  
Flags [S], seq 9713, win 3990, length 0  
16:31:01.585229 IP berke-VirtualBox.6934 > berke-VirtualBox.http:  
Flags [S], seq 7629, win 4948, length 0  
16:31:01.643464 IP berke-VirtualBox.5687 > berke-VirtualBox.http:  
Flags [S], seq 4521, win 3096, length 0
```