

# Fraggle Attack on CitySmart Digital Infrastructure

Ramyasree Bakka

**Source IP:** 192.168.100.5 (the Kali machine's IP).

**Source Port:** 48845 (the dynamic port from which the packet is sent).

**Destination IP:** 192.168.100.255 (the broadcast address, meaning all devices in the subnet will receive the packet).

**Destination Port:** 7 (the UDP Echo service port, which is used for testing in this attack scenario).

**Length:** 13 bytes (the size of the UDP packet sent).

## Step1 : Setup UDP Echo Service on the Ubuntu (Victim) Machine

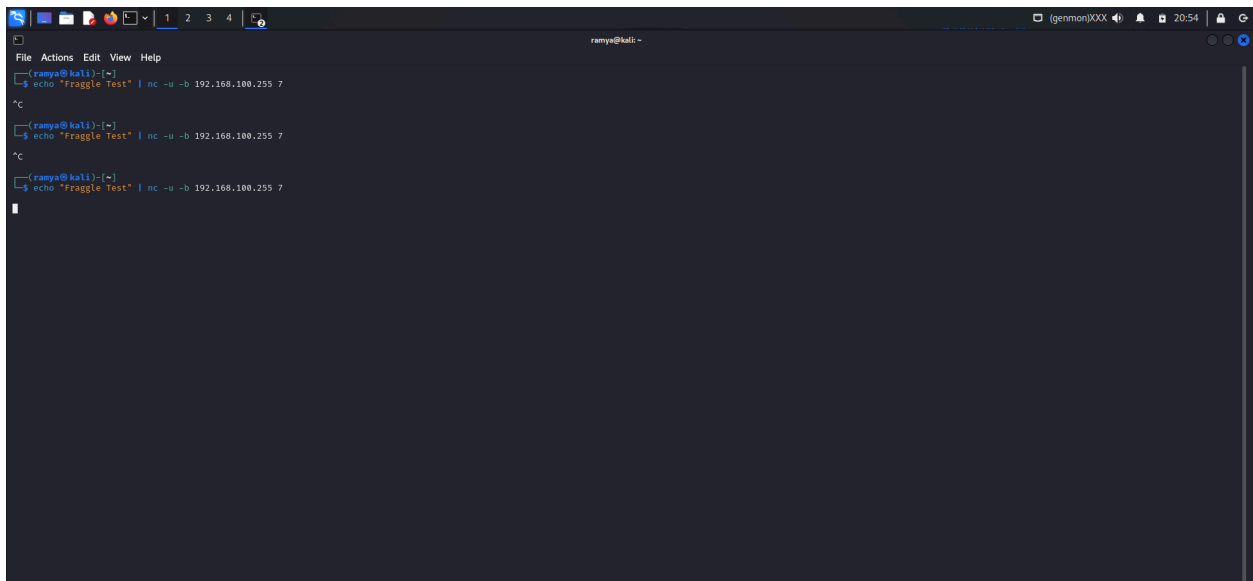
On the ubuntu terminal run the command to start UDP echo listener

**sudo nc -u -l -p 7** ( This listens for incoming UDP packets on port 7 (used for echo).

## Step2: Attack from kali machine

Send the spoofed UDP Echo request to the broadcast address of the network:

In kali terminal **echo "Fraggle Test" | nc -u -b 192.168.100.255 7**

A screenshot of a Kali Linux terminal window. The terminal shows the command 'echo "Fraggle Test" | nc -u -b 192.168.100.255 7' being executed. The output shows the command being run and the resulting network activity. The terminal window has a dark background and a light-colored text. The title bar of the window shows 'ramya@kali -'. The terminal output is as follows:

```
ramya@kali:~$ echo "Fraggle Test" | nc -u -b 192.168.100.255 7
^C
[ramya@kali:~]$ echo "Fraggle Test" | nc -u -b 192.168.100.255 7
^C
[ramya@kali:~]$ echo "Fraggle Test" | nc -u -b 192.168.100.255 7
```

**-b:** Sends the message to the broadcast address (192.168.100.255 in this case).

**192.168.100.255:** Broadcast address for your network.

**Port 7** is used for echo requests.

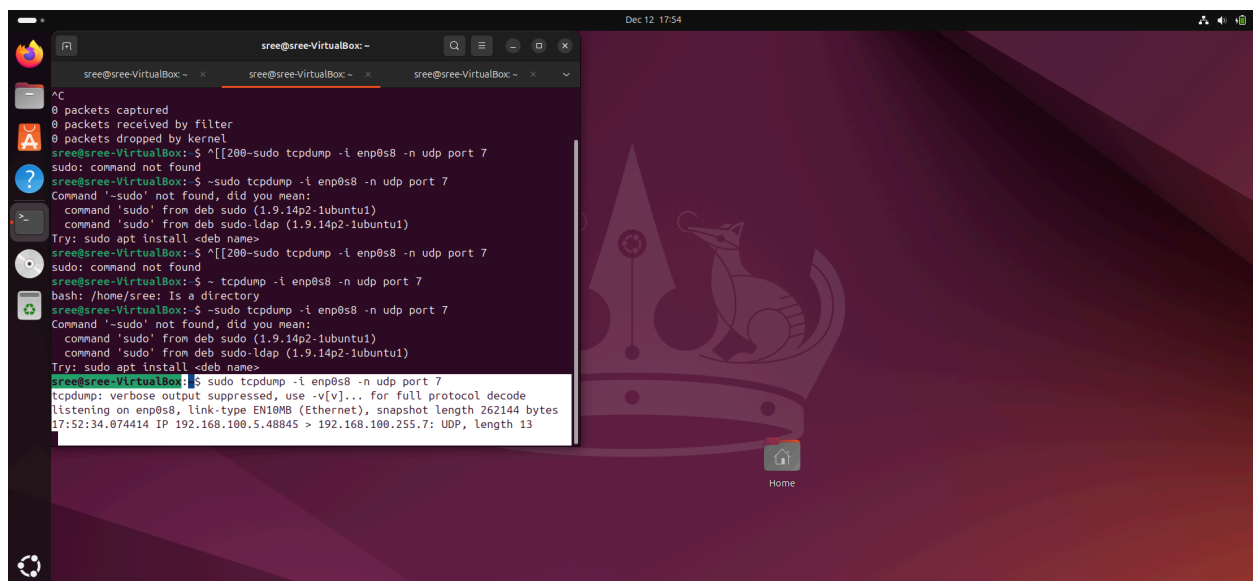
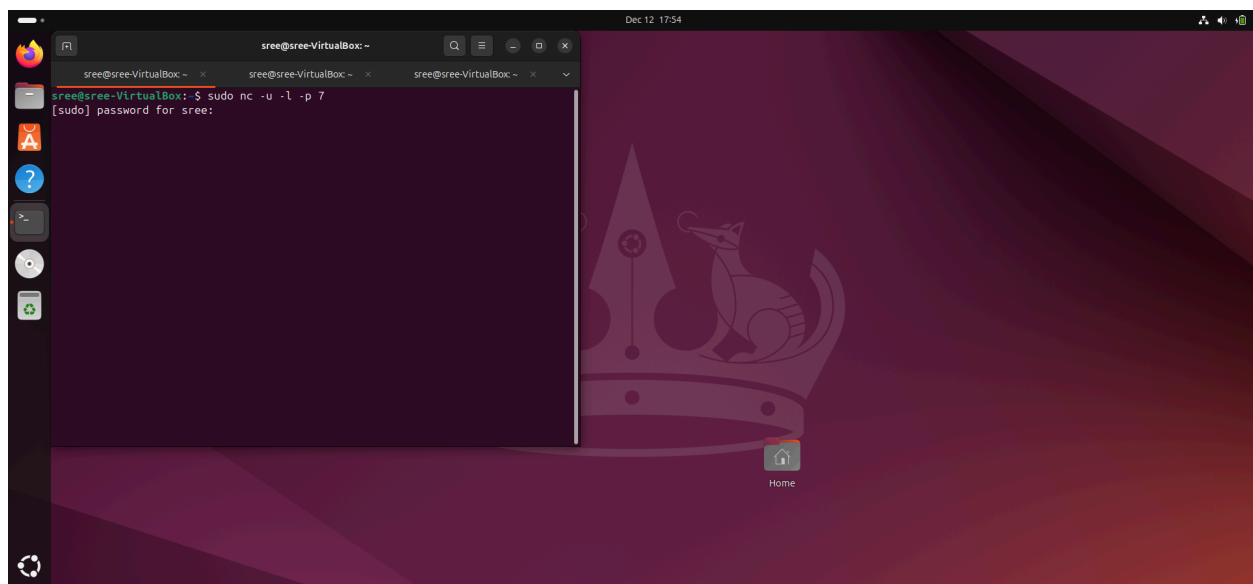
## Step3: monitor the attack in ubuntu open a new terminal to check tcpdump with cmd

```
sudo tcpdump -i enp0s8 -n udp port 7
```

This will show the UDP traffic coming to port 7 from the broadcast address.

**Step4:** mitigating with iptables with **sudo iptables -A INPUT -p udp --dport 7 -j DROP**

And also block all broadcasting traffic **sudo iptables -A INPUT -m pkttype --pkt-type broadcast -j DROP**



```
sree@sree-VirtualBox: ~
After this operation, 350 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu noble/universe amd64 iperf amd64 2.1.9-1 [136 kB]
Fetched 136 kB in 1s (184 kB/s)
Selecting previously unselected package iperf.
(Reading database ... 210240 files and directories currently installed.)
Preparing to unpack .../iperf_2.1.9-1_amd64.deb ...
Unpacking iperf (2.1.9-1) ...
Setting up iperf (2.1.9-1) ...
Processing triggers for man-db (2.12.0-4build2) ...
sree@sree-VirtualBox: ~$ iperf -c 192.168.100.5 -u -b 100M -t 60
Client connecting to 192.168.100.5, UDP port 5001
Sending 1470 byte datagrams, IPG target: 112.15 us (kalman adjust)
UDP buffer size: 200 KByte (default)
[ 1] local 192.168.100.4 port 60555 connected with 192.168.100.5 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 1] 0.0000-60.0002 sec  709 MBytes  99.1 Mbits/sec
[ 1] Sent 505602 datagrams
read failed: Connection refused
read failed: Connection refused
[ 3] WARNING: did not receive ack of last datagram after 10 tries.
sree@sree-VirtualBox: ~$ sudo iptables -A INPUT -p udp --dport 7 -j DROP
sree@sree-VirtualBox: ~$
```

```
sree@sree-VirtualBox: ~
[ 1] Sent 505602 datagrams
read failed: Connection refused
read failed: Connection refused
[ 3] WARNING: did not receive ack of last datagram after 10 tries.
sree@sree-VirtualBox: ~$ sudo iptables -A INPUT -p udp --dport 7 -j DROP
sree@sree-VirtualBox: ~$ sudo iptables -A INPUT -n pkttype --pkt-type broadcast -j DROP
sree@sree-VirtualBox: ~$ sudo iptables -L -v
Chain INPUT (policy ACCEPT 14 packets, 2714 bytes)
pkts bytes target      prot opt in     out     source            destination
  3  123 DROP      udp    --    any    any    anywhere          anywhere
    udp dpt:echo
  0    0 DROP      all    --    any    any    anywhere          anywhere
    PKTTYPE = broadcast
Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target      prot opt in     out     source            destination
Chain OUTPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target      prot opt in     out     source            destination
sree@sree-VirtualBox: ~$ sudo kill -f "nc -u -l -p 7"
sree@sree-VirtualBox: ~$
```

**Step5** : stop the echo listener in ubuntu **sudo kill -f "nc -u -l -p 7"**

**General image how fraggle works**

# How Does a Fraggile Attack Work?

