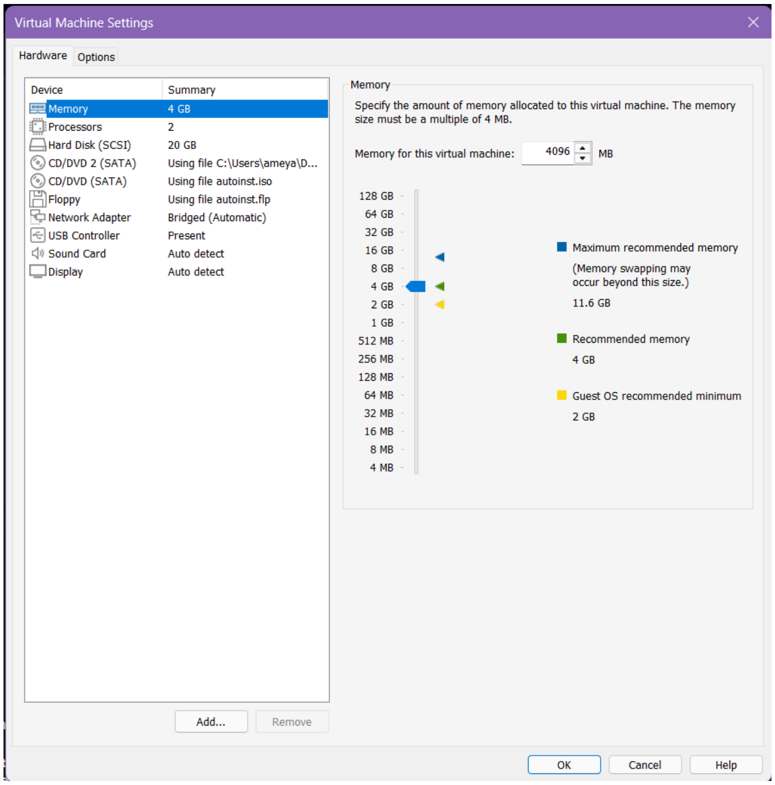
Step 1: Create 2 VMs in VMware

- Ubuntu/Kali Linux (4 GB RAM each)
- Enable bridged or NAT networking to allow communication



Device	Summary
Memory	4 GB
Processors	2
Hard Disk (SCSI)	20 GB
CD/DVD 2 (SATA)	Using file C:\Users\ameya\D...
CD/DVD (SATA)	Using file autoinst.iso
Floppy	Using file autoinst.flp
Network Adapter	Bridged (Automatic)
USB Controller	Present
Sound Card	Auto detect
Display	Auto detect

Memory

Specify the amount of memory allocated to this virtual machine. The memory size must be a multiple of 4 MB.

Memory for this virtual machine: 4096 MB

🟦 Maximum recommended memory
(Memory swapping may occur beyond this size.)
11.6 GB

🟩 Recommended memory
4 GB

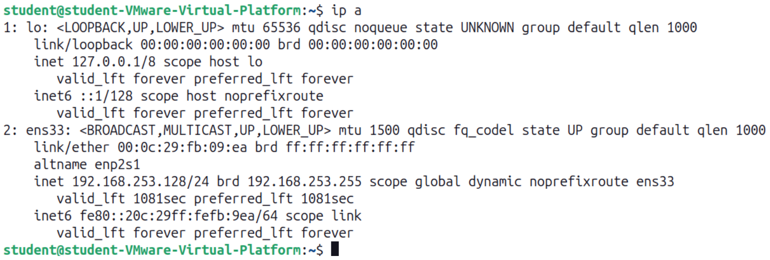
🟨 Guest OS recommended minimum
2 GB

Step 2: Start SSH Server (on one VM)

```
sudo apt install openssh-server -y
sudo systemctl enable ssh
sudo systemctl start ssh
```

Check IP:

```
ip a
```



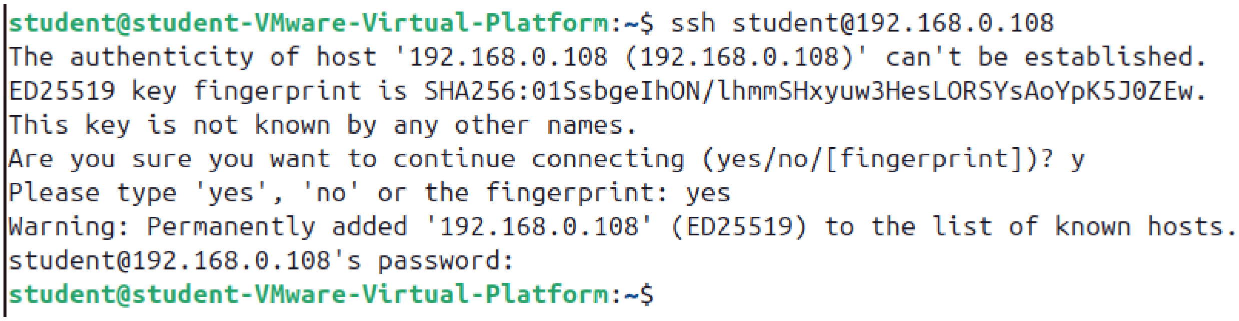
```
student@student-VMware-Virtual-Platform:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:fb:09:ea brd ff:ff:ff:ff:ff:ff
    altname enp2s1
    inet 192.168.253.128/24 brd 192.168.253.255 scope global dynamic noprefixroute ens33
       valid_lft 1081sec preferred_lft 1081sec
    inet6 fe80::20c:29ff:fefb:9ea/64 scope link
       valid_lft forever preferred_lft forever
student@student-VMware-Virtual-Platform:~$
```

Step 3: Connect via SSH (from other VM)

ssh username@<target_ip>

Note: Here the username & target ip is the VM name and its ip address. (Check the IP using ifconfig)

Accept the key and enter password. This simulates secure communication.

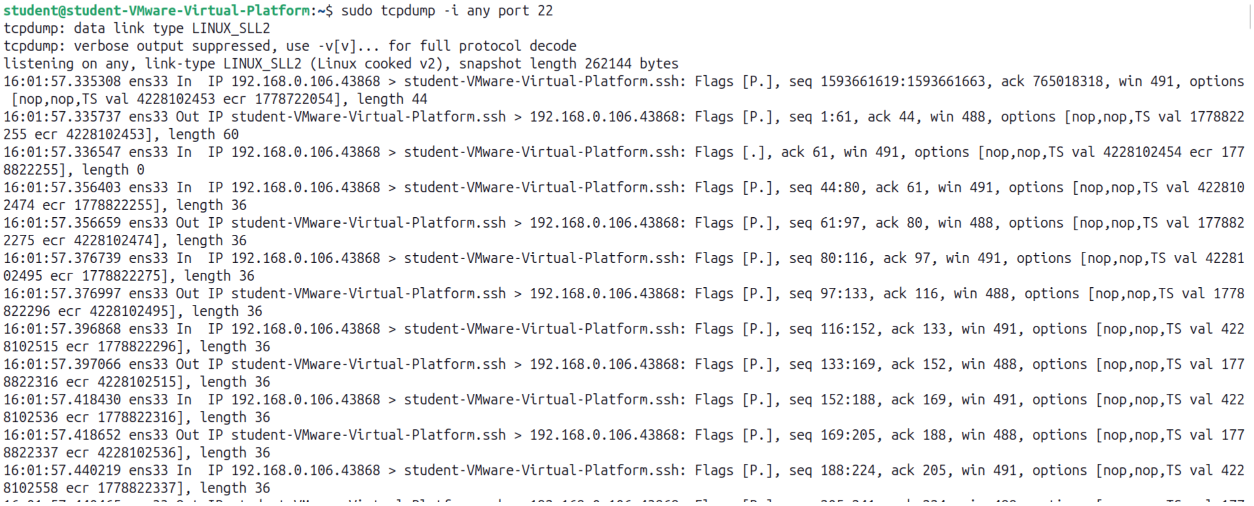


```
student@student-VMware-Virtual-Platform:~$ ssh student@192.168.0.108
The authenticity of host '192.168.0.108 (192.168.0.108)' can't be established.
ED25519 key fingerprint is SHA256:01SsbgeIhON/lhmmSHxyuw3HesLORSYsAoYpK5J0ZEw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '192.168.0.108' (ED25519) to the list of known hosts.
student@192.168.0.108's password:
student@student-VMware-Virtual-Platform:~$
```

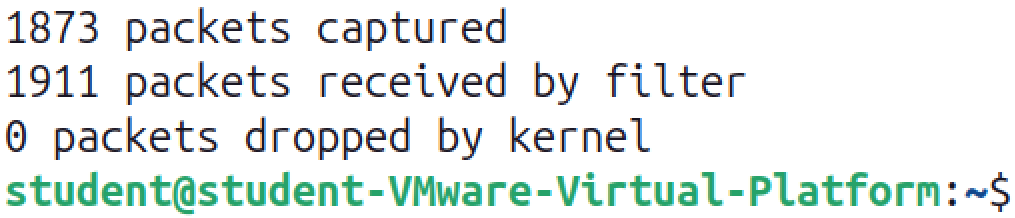
Step 4: Monitor Network using tcpdump

sudo tcpdump -i any port 22

Now try the SSH connection again and observe the output.



```
student@student-VMware-Virtual-Platform:~$ sudo tcpdump -i any port 22
tcpdump: data link type LINUX_SLL2
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on any, link-type LINUX_SLL2 (Linux cooked v2), snapshot length 262144 bytes
16:01:57.335308 ens33 In  IP 192.168.0.106.43868 > student-VMware-Virtual-Platform.ssh: Flags [P.], seq 1593661619:1593661663, ack 765018318, win 491, options
 [nop,nop,TS val 4228102453 ecr 1778722054], length 44
16:01:57.335737 ens33 Out IP student-VMware-Virtual-Platform.ssh > 192.168.0.106.43868: Flags [P.], seq 1:61, ack 44, win 488, options [nop,nop,TS val 1778822
255 ecr 4228102453], length 60
16:01:57.336547 ens33 In  IP 192.168.0.106.43868 > student-VMware-Virtual-Platform.ssh: Flags [.], ack 61, win 491, options [nop,nop,TS val 4228102454 ecr 177
8822255], length 0
16:01:57.356403 ens33 In  IP 192.168.0.106.43868 > student-VMware-Virtual-Platform.ssh: Flags [P.], seq 44:80, ack 61, win 491, options [nop,nop,TS val 422810
2474 ecr 1778822255], length 36
16:01:57.356659 ens33 Out IP student-VMware-Virtual-Platform.ssh > 192.168.0.106.43868: Flags [P.], seq 61:97, ack 80, win 488, options [nop,nop,TS val 177882
2275 ecr 4228102474], length 36
16:01:57.376739 ens33 In  IP 192.168.0.106.43868 > student-VMware-Virtual-Platform.ssh: Flags [P.], seq 80:116, ack 97, win 491, options [nop,nop,TS val 42281
02495 ecr 1778822275], length 36
16:01:57.376997 ens33 Out IP student-VMware-Virtual-Platform.ssh > 192.168.0.106.43868: Flags [P.], seq 97:133, ack 116, win 488, options [nop,nop,TS val 1778
822296 ecr 4228102495], length 36
16:01:57.396868 ens33 In  IP 192.168.0.106.43868 > student-VMware-Virtual-Platform.ssh: Flags [P.], seq 116:152, ack 133, win 491, options [nop,nop,TS val 422
8102515 ecr 1778822296], length 36
16:01:57.397066 ens33 Out IP student-VMware-Virtual-Platform.ssh > 192.168.0.106.43868: Flags [P.], seq 133:169, ack 152, win 488, options [nop,nop,TS val 177
8822316 ecr 4228102515], length 36
16:01:57.418430 ens33 In  IP 192.168.0.106.43868 > student-VMware-Virtual-Platform.ssh: Flags [P.], seq 152:188, ack 169, win 491, options [nop,nop,TS val 422
8102536 ecr 1778822316], length 36
16:01:57.418652 ens33 Out IP student-VMware-Virtual-Platform.ssh > 192.168.0.106.43868: Flags [P.], seq 169:205, ack 188, win 488, options [nop,nop,TS val 177
8822337 ecr 4228102536], length 36
16:01:57.440219 ens33 In  IP 192.168.0.106.43868 > student-VMware-Virtual-Platform.ssh: Flags [P.], seq 188:224, ack 205, win 491, options [nop,nop,TS val 422
8102558 ecr 1778822337], length 36
```



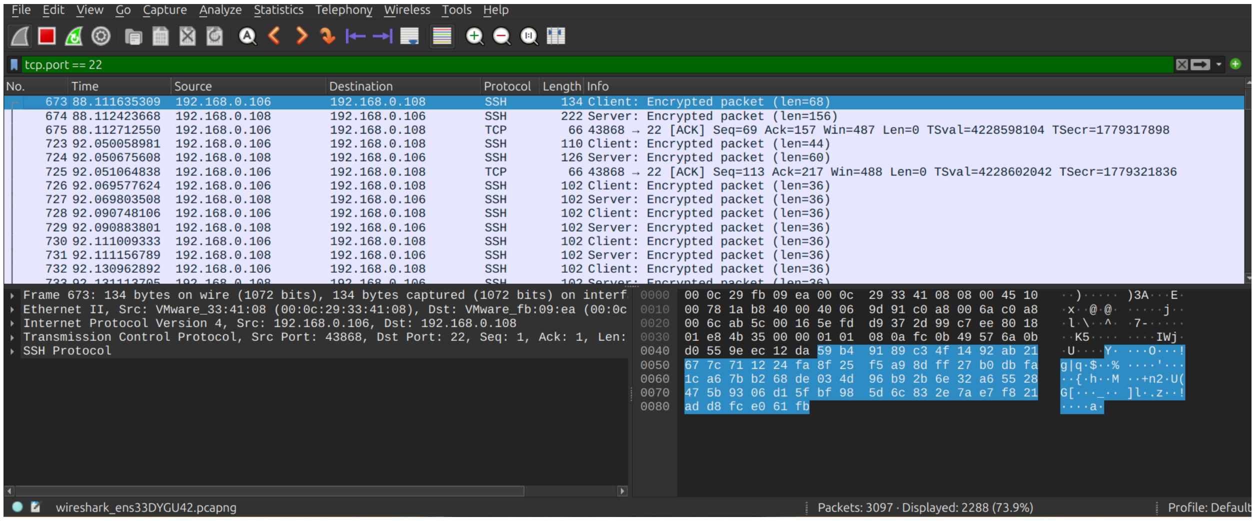
```
1873 packets captured
1911 packets received by filter
0 packets dropped by kernel
student@student-VMware-Virtual-Platform:~$
```

Step 5: Use Wireshark for Packet Capture

sudo wireshark &

Capture packets on eth0 or ens33

Apply filter: tcp.port == 22


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
tcp.port == 22
No.  Time          Source         Destination    Protocol Length Info
 673 88.111635309  192.168.0.106  192.168.0.108  SSH      134 Client: Encrypted packet (len=68)
 674 88.112423668  192.168.0.108  192.168.0.106  SSH      222 Server: Encrypted packet (len=156)
 675 88.112712550  192.168.0.106  192.168.0.108  TCP       66 43868 → 22 [ACK] Seq=69 Ack=157 Win=487 Len=0 TSval=4228598104 TSecr=1779317898
 723 92.050058981  192.168.0.106  192.168.0.108  SSH      110 Client: Encrypted packet (len=44)
 724 92.050675608  192.168.0.108  192.168.0.106  SSH      126 Server: Encrypted packet (len=60)
 725 92.051064838  192.168.0.106  192.168.0.108  TCP       66 43868 → 22 [ACK] Seq=113 Ack=217 Win=488 Len=0 TSval=4228602042 TSecr=1779321836
 726 92.069577624  192.168.0.106  192.168.0.108  SSH      102 Client: Encrypted packet (len=36)
 727 92.069803508  192.168.0.108  192.168.0.106  SSH      102 Server: Encrypted packet (len=36)
 728 92.090748106  192.168.0.106  192.168.0.108  SSH      102 Client: Encrypted packet (len=36)
 729 92.090883801  192.168.0.108  192.168.0.106  SSH      102 Server: Encrypted packet (len=36)
 730 92.111009333  192.168.0.106  192.168.0.108  SSH      102 Client: Encrypted packet (len=36)
 731 92.111156789  192.168.0.108  192.168.0.106  SSH      102 Server: Encrypted packet (len=36)
 732 92.130962892  192.168.0.106  192.168.0.108  SSH      102 Client: Encrypted packet (len=36)
 733 92.131113705  192.168.0.108  192.168.0.106  SSH      102 Server: Encrypted packet (len=36)
```

```
Frame 673: 134 bytes on wire (1072 bits), 134 bytes captured (1072 bits) on interf
Ethernet II, Src: VMware_33:41:08 (00:0c:29:33:41:08), Dst: VMware_fb:09:ea (00:0c
Internet Protocol Version 4, Src: 192.168.0.106, Dst: 192.168.0.108
Transmission Control Protocol, Src Port: 43868, Dst Port: 22, Seq: 1, Ack: 1, Len:
SSH Protocol
```

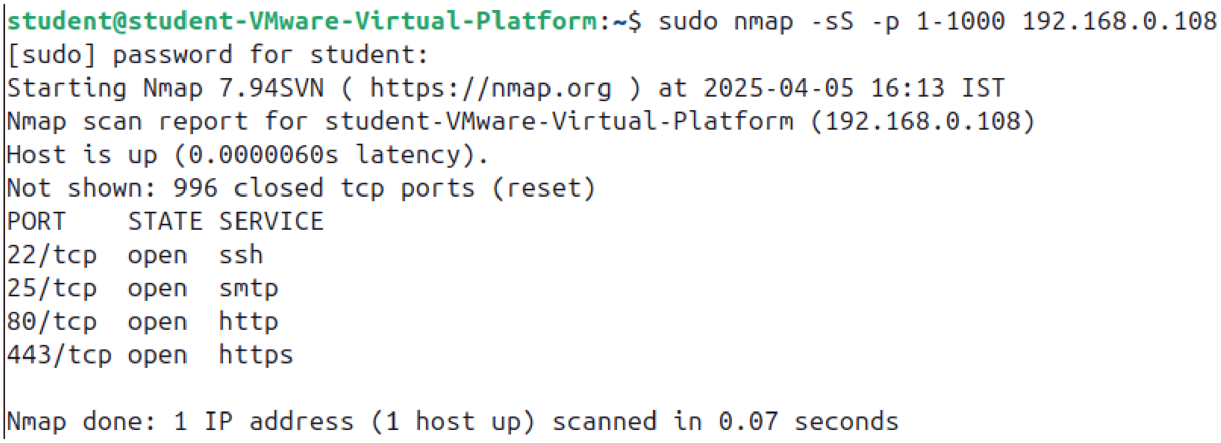
```
0000  00 0c 29 fb 09 ea 00 0c  29 33 41 08 08 00 45 10   ··)·····)3A···E·
0010  00 78 1a b8 40 00 40 06  9d 91 c0 a8 00 6a c0 a8   ·x··@·@·····j··
0020  00 6c ab 5c 00 16 5e fd  d9 37 2d 99 c7 ee 80 18   ·l·\··^··7-·····
0030  01 e8 4b 35 00 00 01 01  08 0a fc 0b 49 57 6a 0b   ··K5········IWj·
0040  d0 55 9e ec 12 da 59 b4  91 89 c3 4f 14 92 ab 21   ·U····Y····O···!
0050  67 7c 71 12 24 fa 8f 25  f5 a9 8d ff 27 b0 db fa   g|q·$··%····'···
0060  1c a6 7b b2 68 de 03 4d  96 b9 2b 6e 32 a6 55 28   ··{·h··M··+n2·U(
0070  47 5b 93 06 d1 5f bf 98  5d 6c 83 2e 7a e7 f8 21   G[···_··]l·.z··!
0080  ad d8 fc e0 61 fb                                  ····a·
```

wireshark_ens33DYGU42.pcapng Packets: 3097 · Displayed: 2288 (73.9%) Profile: Default

Step 6: Scan Ports with nmap

nmap -sS -p 1-1000 <target_ip>

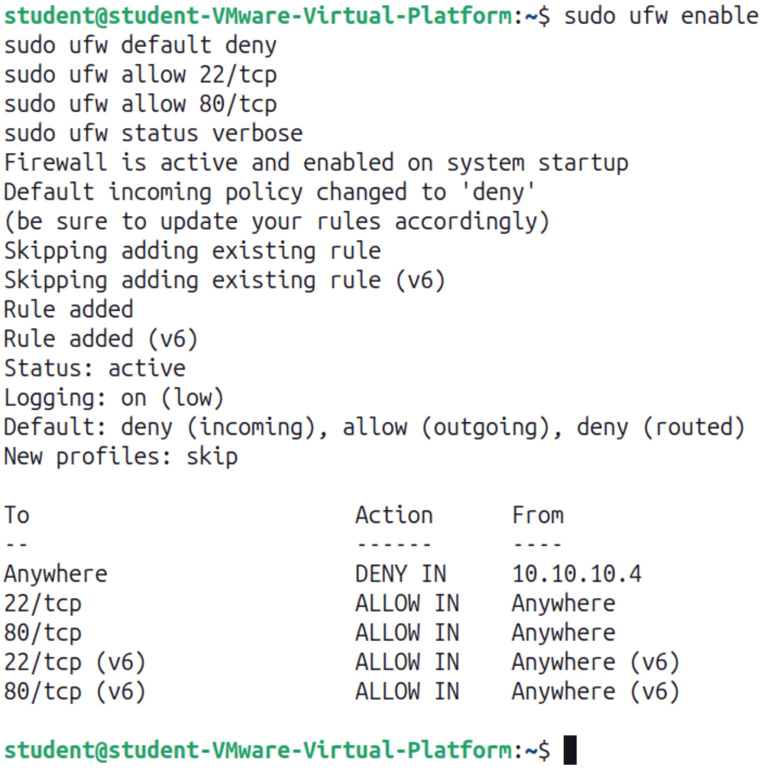
Observe open ports and services



```
student@student-VMware-Virtual-Platform:~$ sudo nmap -sS -p 1-1000 192.168.0.108
[sudo] password for student:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-05 16:13 IST
Nmap scan report for student-VMware-Virtual-Platform (192.168.0.108)
Host is up (0.0000060s latency).
Not shown: 996 closed tcp ports (reset)
PORT    STATE SERVICE
22/tcp  open  ssh
25/tcp  open  smtp
80/tcp  open  http
443/tcp open  https

Nmap done: 1 IP address (1 host up) scanned in 0.07 seconds
```

Step 7: Secure Ports using UFW Firewall

```
    sudo ufw enable
    sudo ufw default deny
    sudo ufw allow 22/tcp
    sudo ufw allow 80/tcp
   sudo ufw status verbose
```



```
student@student-VMware-Virtual-Platform:~$ sudo ufw enable
sudo ufw default deny
sudo ufw allow 22/tcp
sudo ufw allow 80/tcp
sudo ufw status verbose
Firewall is active and enabled on system startup
Default incoming policy changed to 'deny'
(be sure to update your rules accordingly)
Skipping adding existing rule
Skipping adding existing rule (v6)
Rule added
Rule added (v6)
Status: active
Logging: on (low)
Default: deny (incoming), allow (outgoing), deny (routed)
New profiles: skip

To                         Action      From
--                         ------      ----
Anywhere                   DENY IN     10.10.10.4
22/tcp                     ALLOW IN    Anywhere
80/tcp                     ALLOW IN    Anywhere
22/tcp (v6)                ALLOW IN    Anywhere (v6)
80/tcp (v6)                ALLOW IN    Anywhere (v6)

student@student-VMware-Virtual-Platform:~$ 
```

Step 8: Buffer Overflow Simulation (C Program)

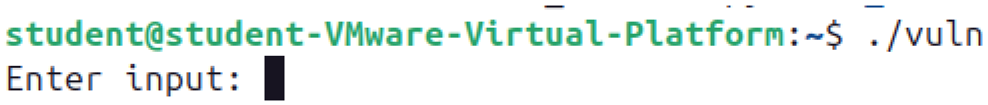
```
#include <stdio.h>
#include <string.h>

void vulnerable_function() {
        char buffer[10];
        printf("Enter input: ");
        gets(buffer);  // Unsafe
        printf("You entered: %s\n", buffer);
}

int main() {
        vulnerable_function();
return 0;
}
```

Save this as vuln.c (use command nano vuln.c)

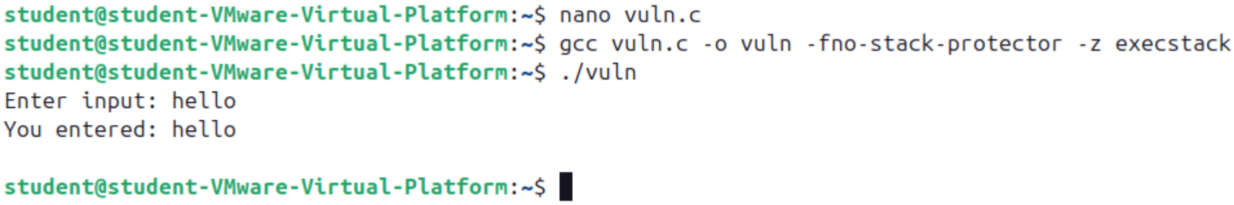
Compile and Run:

```
gcc vuln.c -o vuln -fno-stack-protector -z execstack
./vuln
```

Fix:
Replace gets(buffer) with fgets(buffer, sizeof(buffer), stdin);



```
student@student-VMware-Virtual-Platform:~$ ./vuln
Enter input: 
```

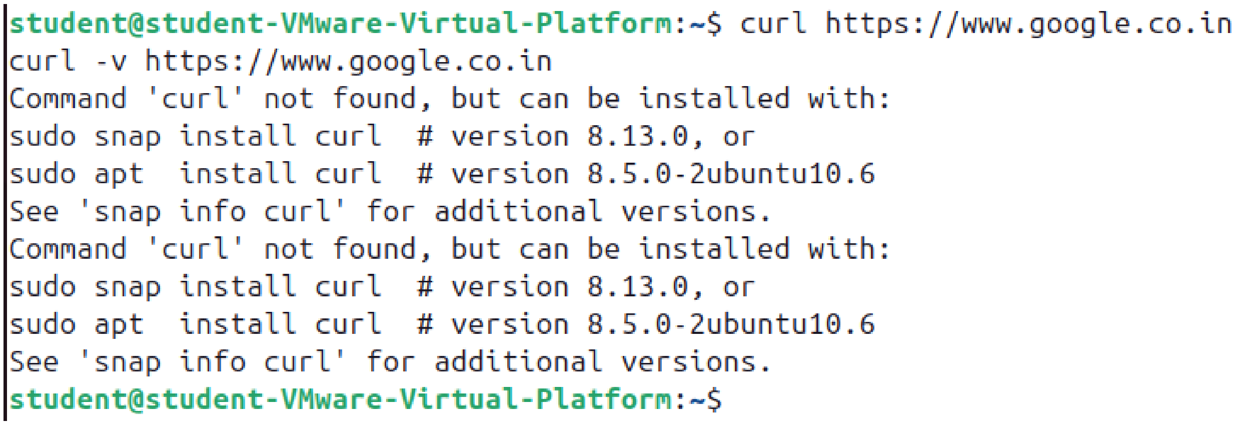


```
student@student-VMware-Virtual-Platform:~$ nano vuln.c
student@student-VMware-Virtual-Platform:~$ gcc vuln.c -o vuln -fno-stack-protector -z execstack
student@student-VMware-Virtual-Platform:~$ ./vuln
Enter input: hello
You entered: hello

student@student-VMware-Virtual-Platform:~$ 
```

Step 9: Simulate HTTPS Request

```
curl https://www.google.co.in

curl -v https://www.google.co.in
```

Observe headers, encryption (SSL/TLS)



```
student@student-VMware-Virtual-Platform:~$ curl https://www.google.co.in
curl -v https://www.google.co.in
Command 'curl' not found, but can be installed with:
sudo snap install curl  # version 8.13.0, or
sudo apt  install curl  # version 8.5.0-2ubuntu10.6
See 'snap info curl' for additional versions.
Command 'curl' not found, but can be installed with:
sudo snap install curl  # version 8.13.0, or
sudo apt  install curl  # version 8.5.0-2ubuntu10.6
See 'snap info curl' for additional versions.
student@student-VMware-Virtual-Platform:~$
```