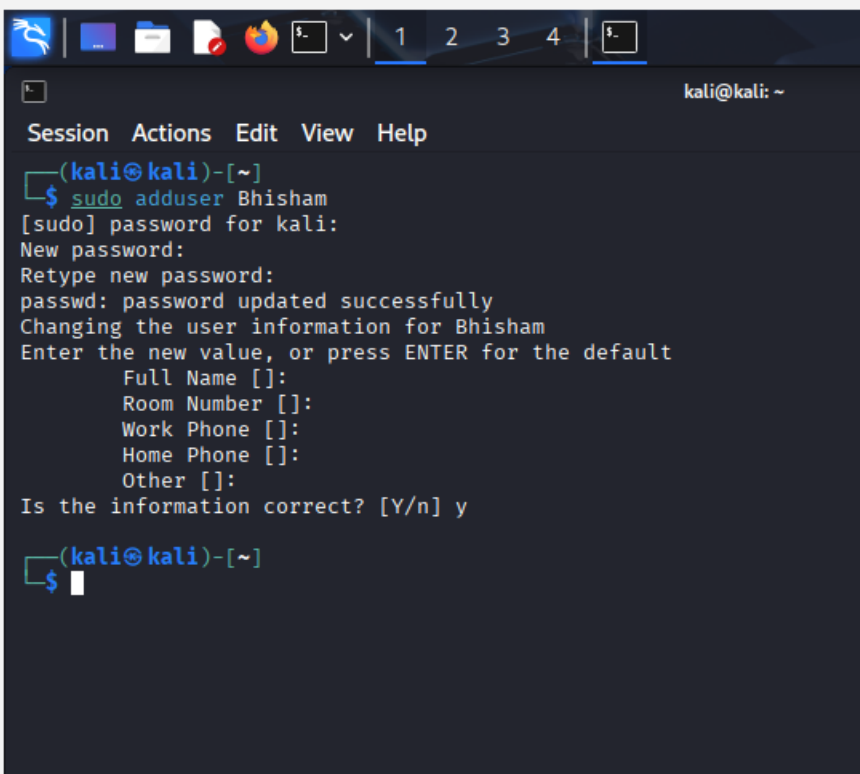


# Project-2: Comparative Analysis of Telnet and SSH Using Wireshark –

*By: Bhisham Sahu*

## 1. Kali Linux Hostname / User Configuration:

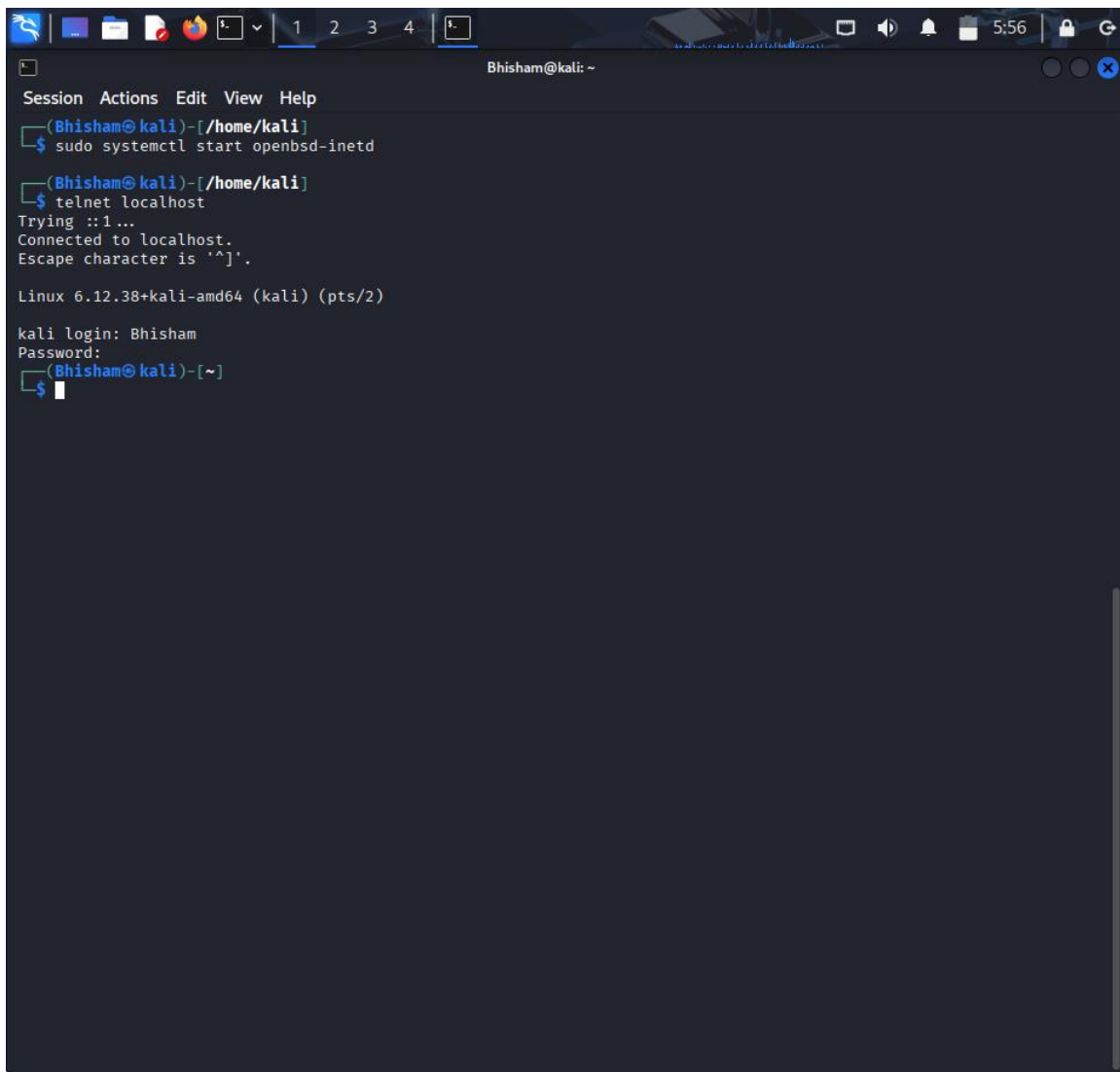
A non-root user (Bhisham) was created on Kali Linux and user switching was successfully performed to simulate a real-world multi-user operating environment.



```
kali@kali: ~  
Session Actions Edit View Help  
(kali@kali)-[~]  
$ sudo adduser Bhisham  
[sudo] password for kali:  
New password:  
Retype new password:  
passwd: password updated successfully  
Changing the user information for Bhisham  
Enter the new value, or press ENTER for the default  
Full Name []:  
Room Number []:  
Work Phone []:  
Home Phone []:  
Other []:  
Is the information correct? [Y/n] y  
(kali@kali)-[~]  
$
```

## 2. Telnet Service Configuration and Activation:

The Telnet service was enabled using `openbsd-inetd`, allowing plaintext communication for demonstrating protocol-level security weaknesses.

A terminal window titled 'Bhisham@kali: ~' with a menu bar (Session, Actions, Edit, View, Help) and a tab bar (1, 2, 3, 4, 5). The terminal shows the following commands and output:

```
(Bhisham@kali)-[/home/kali]
$ sudo systemctl start openbsd-inetd

(Bhisham@kali)-[/home/kali]
$ telnet localhost
Trying ::1...
Connected to localhost.
Escape character is '^['.

Linux 6.12.38+kali-amd64 (kali) (pts/2)

kali login: Bhisham
Password:
(Bhisham@kali)-[~]
$
```

### 3. Wireshark Capturing Telnet Traffic:

Wireshark was used to capture Telnet network traffic on TCP port 23, enabling packet-level inspection of unencrypted data.

The screenshot displays the Wireshark network protocol analyzer interface. The top menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, and Help. The main window is divided into three panes: the packet list, packet details, and packet bytes.

The packet list pane shows a list of captured packets. The filter is set to 'telnet'. The list includes columns for No., Time, Source, Destination, Protocol, Length, and Info. The selected packet (No. 135) is highlighted in blue.

No.	Time	Source	Destination	Protocol	Length	Info
93	74.673097861	10.75.192.40	10.75.192.195	TELNET	143	89 bytes data
99	79.838974178	10.75.192.195	10.75.192.40	TELNET	60	1 byte data
100	79.840635469	10.75.192.40	10.75.192.195	TELNET	55	1 byte data
102	80.316593539	10.75.192.195	10.75.192.40	TELNET	60	1 byte data
103	80.318010824	10.75.192.40	10.75.192.195	TELNET	55	1 byte data
105	81.138164300	10.75.192.195	10.75.192.40	TELNET	60	1 byte data
106	81.140125281	10.75.192.40	10.75.192.195	TELNET	60	6 bytes data
108	81.277057844	10.75.192.195	10.75.192.40	TELNET	60	1 byte data
109	81.279174817	10.75.192.40	10.75.192.195	TELNET	55	1 byte data
111	82.023657481	10.75.192.195	10.75.192.40	TELNET	60	2 bytes data
112	82.025126267	10.75.192.40	10.75.192.195	TELNET	66	12 bytes data
114	82.067823103	10.75.192.40	10.75.192.195	TELNET	264	210 bytes data
116	83.987824415	10.75.192.195	10.75.192.40	TELNET	60	1 byte data
117	83.988489026	10.75.192.40	10.75.192.195	TELNET	55	1 byte data
119	84.247536178	10.75.192.195	10.75.192.40	TELNET	60	1 byte data
120	84.247931286	10.75.192.40	10.75.192.195	TELNET	55	1 byte data
122	85.199262159	10.75.192.195	10.75.192.40	TELNET	60	1 byte data
123	85.199721548	10.75.192.40	10.75.192.195	TELNET	55	1 byte data
125	86.727686185	10.75.192.195	10.75.192.40	TELNET	60	1 byte data
126	86.728032974	10.75.192.40	10.75.192.195	TELNET	55	1 byte data
128	86.938179533	10.75.192.195	10.75.192.40	TELNET	60	1 byte data
129	86.939350236	10.75.192.40	10.75.192.195	TELNET	55	1 byte data
131	87.896101011	10.75.192.195	10.75.192.40	TELNET	60	2 bytes data
132	87.897491230	10.75.192.40	10.75.192.195	TELNET	284	230 bytes data
134	91.591628290	10.75.192.195	10.75.192.40	TELNET	60	1 byte data
135	91.592354164	10.75.192.40	10.75.192.195	TELNET	55	1 byte data
137	91.927304641	10.75.192.195	10.75.192.40	TELNET	60	1 byte data
138	91.928211019	10.75.192.40	10.75.192.195	TELNET	55	1 byte data
140	92.573852908	10.75.192.195	10.75.192.40	TELNET	60	2 bytes data
141	92.575003093	10.75.192.40	10.75.192.195	TELNET	66	12 bytes data
143	92.616981960	10.75.192.40	10.75.192.195	TELNET	315	261 bytes data

The packet details pane for the selected packet (No. 135) shows the following structure:

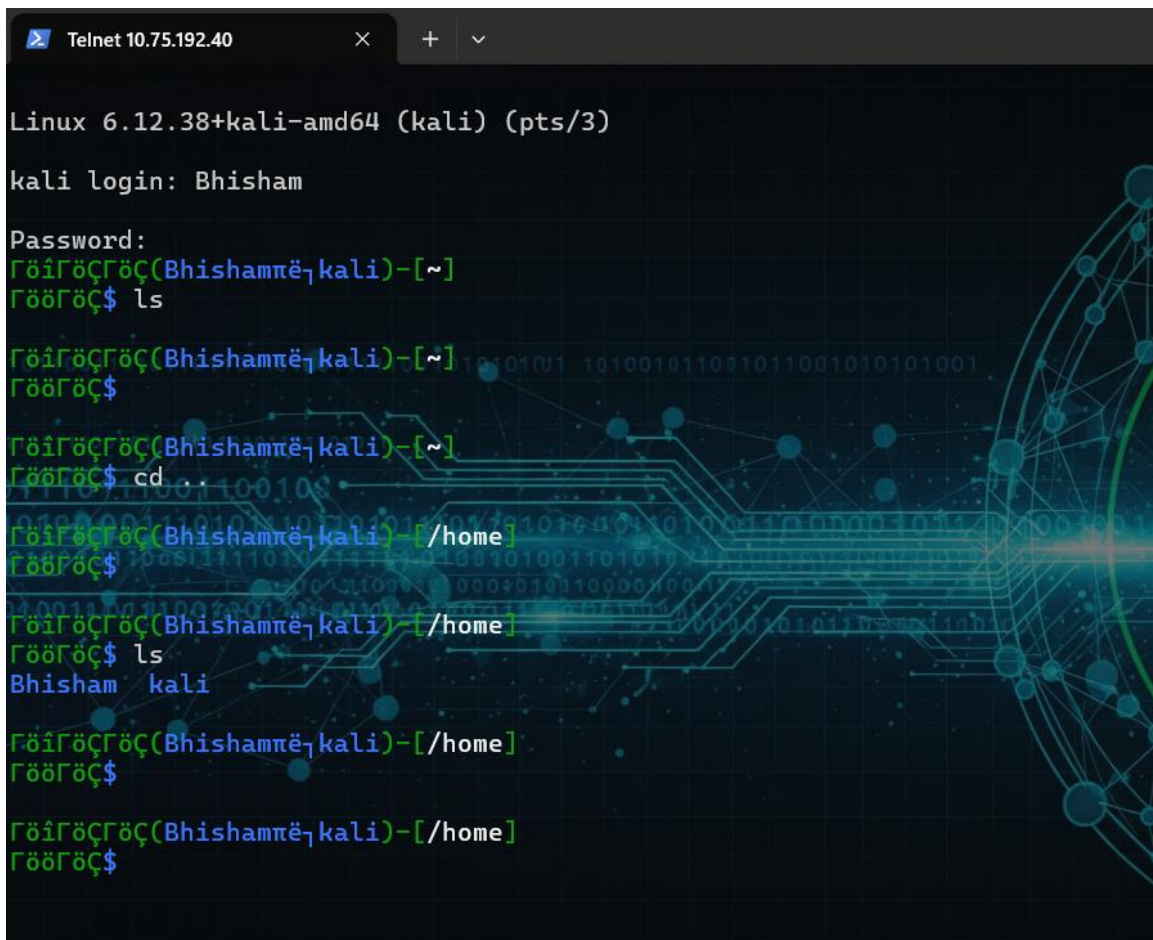
- Frame 135: 55 bytes on wire (440 bits), 55 bytes captured (440 bits) on interface eth0
- Ethernet II, Src: PCSSystemtec\_1f:b7:23 (08:00:27:1f:b7:23), Dst: 10.75.192.195 (08:00:27:1f:b7:23)
- Internet Protocol Version 4, Src: 10.75.192.40, Dst: 10.75.192.195
- Transmission Control Protocol, Src Port: 23, Dst Port: 23
- Telnet

The packet bytes pane shows the raw data for the selected packet, displayed in hexadecimal and ASCII. The data is: 0000 64 32 a8 fb e5 a5 08 00 27 1f b7 23 08 00 45 00 10 00 29 64 dd 40 00 40 06 40 70 0a 4b c0 28 0a 00 20 c0 c3 00 17 d3 94 be c2 3f a9 27 35 f1 61 50 00 30 01 f6 95 9d 00 00 6c

The status bar at the bottom indicates: Packets: 145 · Displayed: 71 (49.0%) · Dropped: 0 (0.0%) · Profile: Default

## 4. Telnet Login and Command Execution:

A successful Telnet login was performed from a Windows client, and commands were executed on the Kali Linux server.

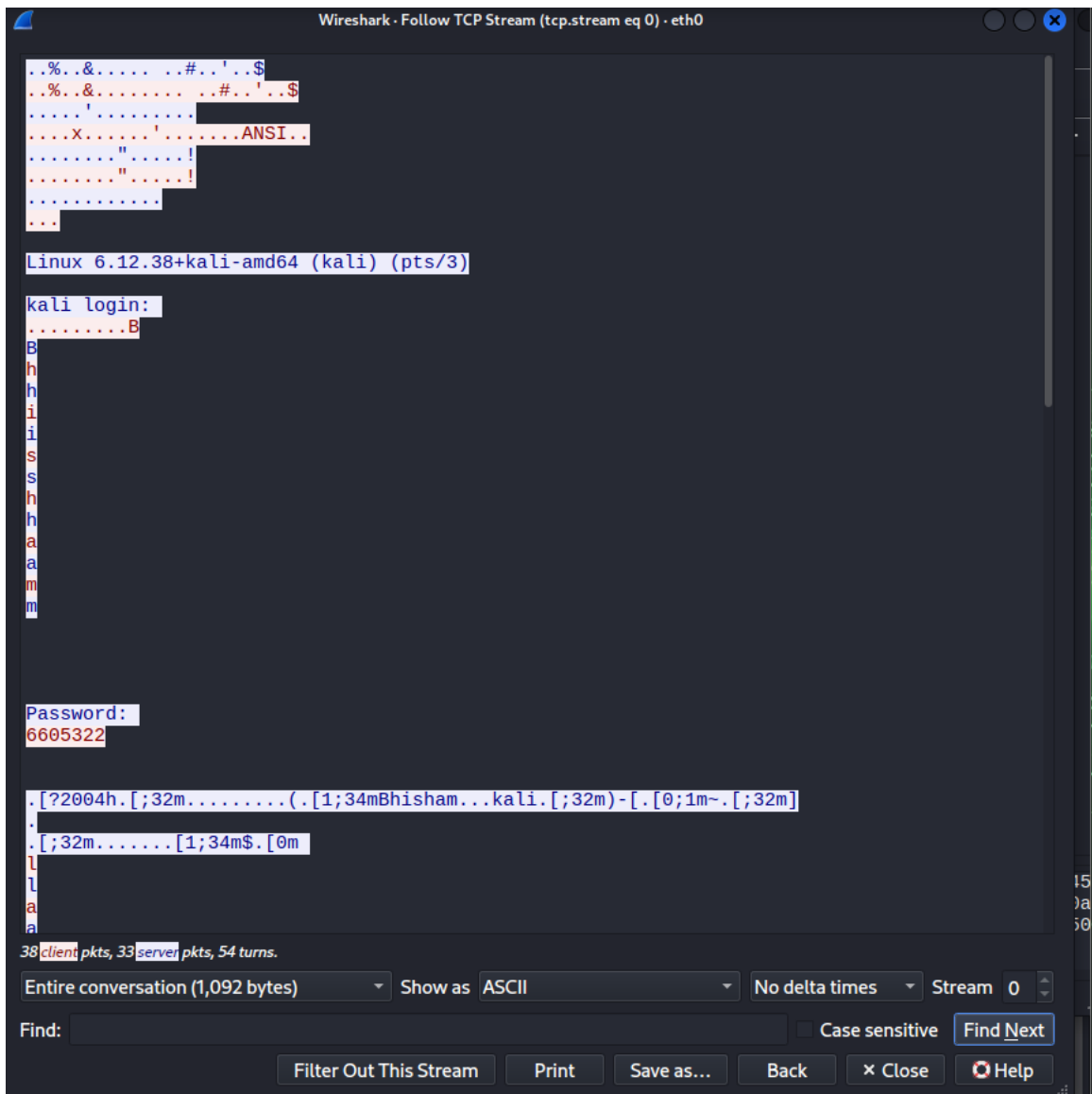
A screenshot of a Telnet session window titled 'Telnet 10.75.192.40'. The window shows a terminal interface with a dark background and green text. The terminal output is as follows:

```
Linux 6.12.38+kali-amd64 (kali) (pts/3)
kali login: Bhisham
Password:
ΓöîΓöÇΓöÇ(Bhisham@kali)-[~]
ΓööΓöÇ$ ls
ΓöîΓöÇΓöÇ(Bhisham@kali)-[~]
ΓööΓöÇ$
ΓöîΓöÇΓöÇ(Bhisham@kali)-[~]
ΓööΓöÇ$ cd ..
ΓöîΓöÇΓöÇ(Bhisham@kali)-[/home]
ΓööΓöÇ$
ΓöîΓöÇΓöÇ(Bhisham@kali)-[/home]
ΓööΓöÇ$ ls
Bhisham kali
ΓöîΓöÇΓöÇ(Bhisham@kali)-[/home]
ΓööΓöÇ$
ΓöîΓöÇΓöÇ(Bhisham@kali)-[/home]
ΓööΓöÇ$
```

The background of the terminal window features a faint, stylized graphic of a network or circuit with glowing blue and green lines and nodes.

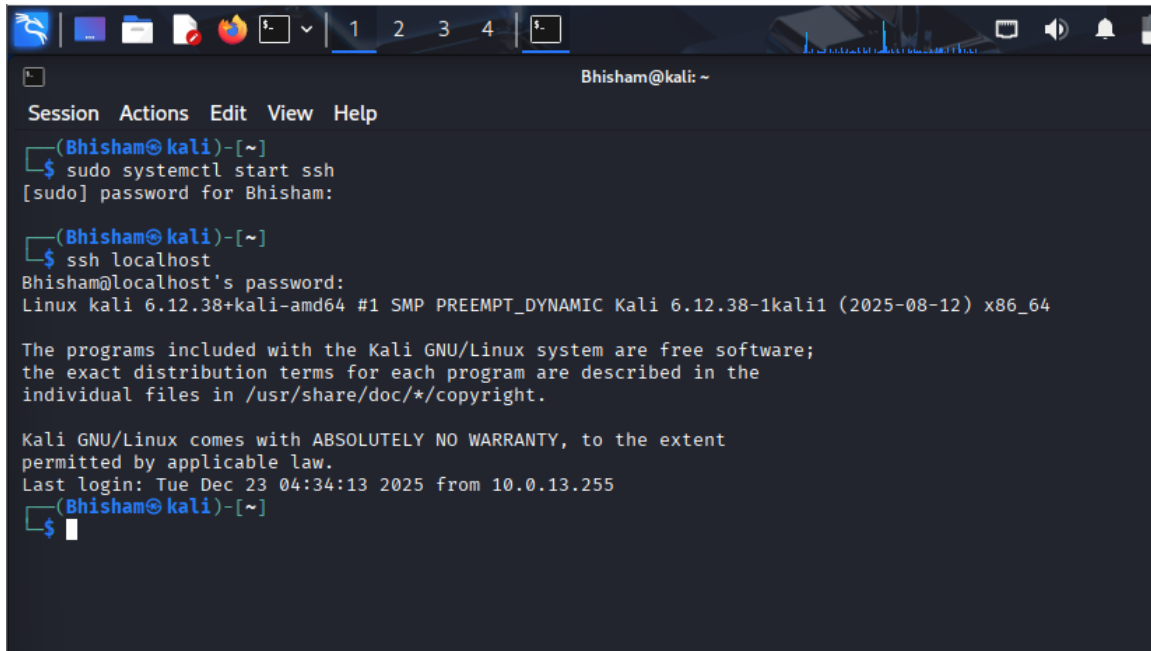
## 5. Plain-text Credentials Visible in Packets:

Username, password, and commands were observed in readable plaintext within the Telnet TCP stream, confirming the insecure nature of Telnet.



## 6. SSH Service Configuration and Activation:

The SSH service was verified to be active, enabling secure and encrypted remote access.

A terminal window titled "Bhisham@kali: ~" with a menu bar (Session, Actions, Edit, View, Help). The user runs "sudo systemctl start ssh" and provides the password. Then, they run "ssh localhost", which prompts for the password. The terminal displays system information: "Linux kali 6.12.38+kali-amd64 #1 SMP PREEMPT\_DYNAMIC Kali 6.12.38-1kali1 (2025-08-12) x86\_64". It also shows the Kali GNU/Linux warranty disclaimer and the last login time: "Last login: Tue Dec 23 04:34:13 2025 from 10.0.13.255". The prompt returns to the user, and a cursor is visible on the next line.

```
Bhisham@kali: ~  
Session Actions Edit View Help  
(Bhisham@kali)-[~]  
$ sudo systemctl start ssh  
[sudo] password for Bhisham:  
(Bhisham@kali)-[~]  
$ ssh localhost  
Bhisham@localhost's password:  
Linux kali 6.12.38+kali-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.38-1kali1 (2025-08-12) x86_64  
  
The programs included with the Kali GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Tue Dec 23 04:34:13 2025 from 10.0.13.255  
(Bhisham@kali)-[~]  
$
```

## 7. Wireshark Capturing SSH Traffic:

SSH network traffic was captured on TCP port 22, allowing analysis of encrypted communication.

No.	Time	Source	Destination	Protocol	Length	Info
69	55.967428093	10.75.192.195	10.75.192.40	SSHv2	87	Client: Protocol (SSH-)
71	55.995164594	10.75.192.40	10.75.192.195	SSHv2	87	Server: Protocol (SSH-)
72	55.997939847	10.75.192.195	10.75.192.40	SSHv2	1486	Client: Key Exchange I
73	56.024059814	10.75.192.40	10.75.192.195	SSHv2	1094	Server: Key Exchange I
74	56.027570849	10.75.192.195	10.75.192.40	SSHv2	102	Client: Elliptic Curve
75	56.038018363	10.75.192.40	10.75.192.195	SSHv2	546	Server: Elliptic Curve
79	61.610012624	10.75.192.195	10.75.192.40	SSHv2	70	Client: New Keys
81	61.654753468	10.75.192.195	10.75.192.40	SSHv2	98	Client: Encrypted pack
83	61.655408863	10.75.192.40	10.75.192.195	SSHv2	98	Server: Encrypted pack
84	61.656487827	10.75.192.195	10.75.192.40	SSHv2	122	Client: Encrypted pack
85	61.660807000	10.75.192.40	10.75.192.195	SSHv2	106	Server: Encrypted pack
87	73.229932126	10.75.192.195	10.75.192.40	SSHv2	202	Client: Encrypted pack
89	73.376290294	10.75.192.40	10.75.192.195	SSHv2	82	Server: Encrypted pack
90	73.396494740	10.75.192.195	10.75.192.40	SSHv2	166	Client: Encrypted pack
92	73.470656639	10.75.192.40	10.75.192.195	SSHv2	682	Server: Encrypted pack
94	73.513659940	10.75.192.195	10.75.192.195	SSHv2	98	Server: Encrypted pack
95	73.515081119	10.75.192.195	10.75.192.40	SSHv2	190	Client: Encrypted pack
97	73.547984166	10.75.192.40	10.75.192.195	SSHv2	162	Server: Encrypted pack
98	73.550371210	10.75.192.40	10.75.192.195	SSHv2	522	Server: Encrypted pack
100	73.821193587	10.75.192.40	10.75.192.195	SSHv2	98	Server: Encrypted pack
101	73.822570941	10.75.192.40	10.75.192.195	SSHv2	202	Server: Encrypted pack
104	77.580105844	10.75.192.195	10.75.192.40	SSHv2	90	Client: Encrypted pack
105	77.583182823	10.75.192.40	10.75.192.195	SSHv2	90	Server: Encrypted pack
106	77.599603115	10.75.192.195	10.75.192.40	SSHv2	90	Client: Encrypted pack
107	77.599935933	10.75.192.40	10.75.192.195	SSHv2	90	Server: Encrypted pack
108	77.623339372	10.75.192.195	10.75.192.40	SSHv2	90	Client: Encrypted pack
109	77.623744771	10.75.192.40	10.75.192.195	SSHv2	90	Server: Encrypted pack
110	77.655347223	10.75.192.195	10.75.192.40	SSHv2	90	Client: Encrypted pack
111	77.655633627	10.75.192.40	10.75.192.195	SSHv2	90	Server: Encrypted pack
112	77.686596703	10.75.192.195	10.75.192.40	SSHv2	90	Client: Encrypted pack
113	77.686869495	10.75.192.40	10.75.192.195	SSHv2	90	Server: Encrypted pack

Frame 69: 87 bytes on wire (696 bits), 87 bytes captured on interface eth0	0000	08 00 27 1f b7 23 64 32	a8 fb e5 a5 08 00 45
Ethernet II, Src: Intel_fb:e5:a5 (64:32:a8:fb:e5:a5), Dst: 10.75.192.40	0010	00 49 1f 2b 40 00 80 06	46 02 0a 4b c0 c3 0a
Internet Protocol Version 4, Src: 10.75.192.195, Dst: 10.75.192.40	0020	c0 28 ff f1 00 16 c4 b4	06 d1 7a 06 dc d9 50
Transmission Control Protocol, Src Port: 65521, Dst Port: 22	0030	00 ff ae a0 00 00 53 53	48 2d 32 2e 30 2d 4f
SSH Protocol	0040	65 6e 53 53 48 5f 66 6f	72 5f 57 69 6e 64 6f

SSH Protocol: Protocol      Packets: 901 - Displayed: 789 (87.6%) - Dropped: 0 (0.0%)      Profile: Default

## 8. SSH Login and Command Execution:

Secure authentication and remote command execution were successfully performed using the SSH protocol.



```
Bhisham@kali: /home
PS C:\Users\Bhisham sahu> ssh Bhisham@10.75.192.40
The authenticity of host '10.75.192.40 (10.75.192.40)' can't be established.
ED25519 key fingerprint is SHA256:RR+fkKE3b05fN7JqN/Oi2hOCChrW0mrtP3ZtgGKkSUY.
This host key is known by the following other names/addresses:
C:\Users\Bhisham sahu/.ssh/known_hosts:4: 10.0.17.60
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.75.192.40' (ED25519) to the list of known hosts.
Bhisham@10.75.192.40's password:
Linux kali 6.12.38+kali-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.38-1kali1 (2025-08-12) x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Dec 23 06:07:25 2025 from ::1
(Bhisham@kali)-[~]
$ ls
(Bhisham@kali)-[~]
$ cd ..
(Bhisham@kali)-[/home]
$ ls
Bhisham kali
(Bhisham@kali)-[/home]
$ |
```

## 9. Encrypted SSH Packets (Unreadable Payload):

The SSH TCP stream payload appeared encrypted and unreadable, demonstrating effective cryptographic protection of data.



```
SSH-2.0-OpenSSH_for_Windows_9.5
SSH-2.0-OpenSSH_10.0p2 Debian-8
....
.....C...K.*,.Ag.....curve25519-sha256,curve25519-sha256@libssh.org,ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,diffie-hellman-group-exchange-sha256,diffie-hellman-group16-sha512,diffie-hellman-group18-sha512,diffie-hellman-group14-sha256,ext-info-c,kex-strict-c-v00@openssh.com,...ssh-ed25519-cert-v01@openssh.com,ecdsa-sha2-nistp256-cert-v01@openssh.com,ecdsa-sha2-nistp384-cert-v01@openssh.com,ecdsa-sha2-nistp521-cert-v01@openssh.com,sk-ssh-ed25519-cert-v01@openssh.com,sk-ecdsa-sha2-nistp256-cert-v01@openssh.com,rsa-sha2-512-cert-v01@openssh.com,rsa-sha2-256-cert-v01@openssh.com,ssh-ed25519,ecdsa-sha2-nistp256,ecdsa-sha2-nistp384,ecdsa-sha2-nistp521,sk-ssh-ed25519@openssh.com,sk-ecdsa-sha2-nistp256@openssh.com,rsa-sha2-512,rsa-sha2-256...lchacha20-poly1305@openssh.com,aes128-ctr,aes192-ctr,aes256-ctr,aes128-gcm@openssh.com,aes256-gcm@openssh.com...lchacha20-poly1305@openssh.com,aes128-ctr,aes192-ctr,aes256-ctr,aes128-gcm@openssh.com,aes256-gcm@openssh.com...umac-64-etm@openssh.com,umac-128-etm@openssh.com,hmac-sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.com,umac-64@openssh.com,umac-128@openssh.com,hmac-sha2-256,hmac-sha2-512...umac-64-etm@openssh.com,umac-128-etm@openssh.com,hmac-sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.com,umac-64@openssh.com,umac-128@openssh.com,hmac-sha2-256,hmac-sha2-512...none,zlib@openssh.com,zlib...none,zlib@openssh.com,zlib.....
....
.....V...[...r.S^.....mlkem768x25519-sha256,sntrup761x25519-sha512,sntrup761x25519-sha512@openssh.com,curve25519-sha256,curve25519-sha256@libssh.org,ecdh-sha2-nistp256,ecdh-sha2-nistp384,ecdh-sha2-nistp521,ext-info-s,kex-strict-s-v00@openssh.com...rsa-sha2-512,rsa-sha2-256,ecdsa-sha2-nistp256,ssh-ed25519...lchacha20-poly1305@openssh.com,aes128-gcm@openssh.com,aes256-gcm@openssh.com,aes128-ctr,aes192-ctr,aes256-ctr...lchacha20-poly1305@openssh.com,aes128-gcm@openssh.com,aes256-gcm@openssh.com,aes128-ctr,aes192-ctr,aes256-ctr...umac-64-etm@openssh.com,umac-128-etm@openssh.com,hmac-sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.com,hmac-sha1-etm@openssh.com,umac-64@openssh.com,umac-128@openssh.com,umac-128@openssh.com,hmac-sha2-256,hmac-sha2-512,hmac-sha1...umac-64-etm@openssh.com,umac-128-etm@openssh.com,hmac-sha2-256-etm@openssh.com,hmac-sha2-512-etm@openssh.com,hmac-sha1-etm@openssh.com,umac-64@openssh.com,umac-128@openssh.com,hmac-sha2-256,hmac-sha2-512,hmac-sha1...none,zlib@openssh.com...none,zlib@openssh.com...
....,.....Z.b..?VN...j9+.{.q+&..u.US+).....
....
.....3.....ssh-ed25519.....G.....3V.Df?..3.....W......d...N...Y(Q...~A:..w.+.
....P.X...S...ssh-ed25519...@...$......!...;y.o.-...mTI..#0D.jr..G5..Z6.0Xb.....n.K.
....{ }.....
....
.....1..Ww..0z.H.....d-v2....^.....~.X..`bC.)Z~B...!.
....gM...\\:..1.a...
....xrb.#r.D2>..r.b.q[.cj5.)..p.;.....{-w..].1)J@...d.....i7..E.<...l.....\`g+a...
....F'9o.5..S~s..m
387 client pkts, 402 server pkts, 771 turns.
Entire conversation (33 kB) Show as ASCII No delta times Stream 1
Find: Case sensitive Find Next
```

**10. Final Comparison Summary:**

This project presents a comparative security analysis of Telnet and SSH using real-time network traffic capture with Wireshark. Telnet was intentionally configured to demonstrate its insecure design, as it transmits usernames, passwords, and commands in plaintext, making it highly vulnerable to packet sniffing attacks. In

contrast, SSH was implemented to provide encrypted communication, ensuring confidentiality and integrity of data even when network traffic is intercepted. The captured packets clearly highlight the difference between unencrypted and encrypted protocols.

Based on the observed results, the project concludes that Telnet should not be used in modern networks, whereas SSH is the secure and industry-standard protocol for remote system administration.