FTP (FILE TRANSFER PROTOCOL) VS SFTP (SECURE FILE TRANSFER PROTOCOL) SECURITY

1. **OBJECTIVE:** The aim of this project is to demonstrate the security differences between FTP and SFTP by configuring, analyzing, and comparing their file transfer methods in a simulated environment. It aims to show how FTP exposes data due to lack of encryption, while SFTP secures data using encryption during both authentication and transfer.

2. FTP SETUP AND ANALYSIS

a) FTP configuration

To simulate an insecure file transfer environment, the vsftpd service was installed and configured on Kali Linux. The configuration allowed anonymous access and a test file named credentials.txt was placed in the directory /srv/ftp/files/ and made accessible with proper permissions

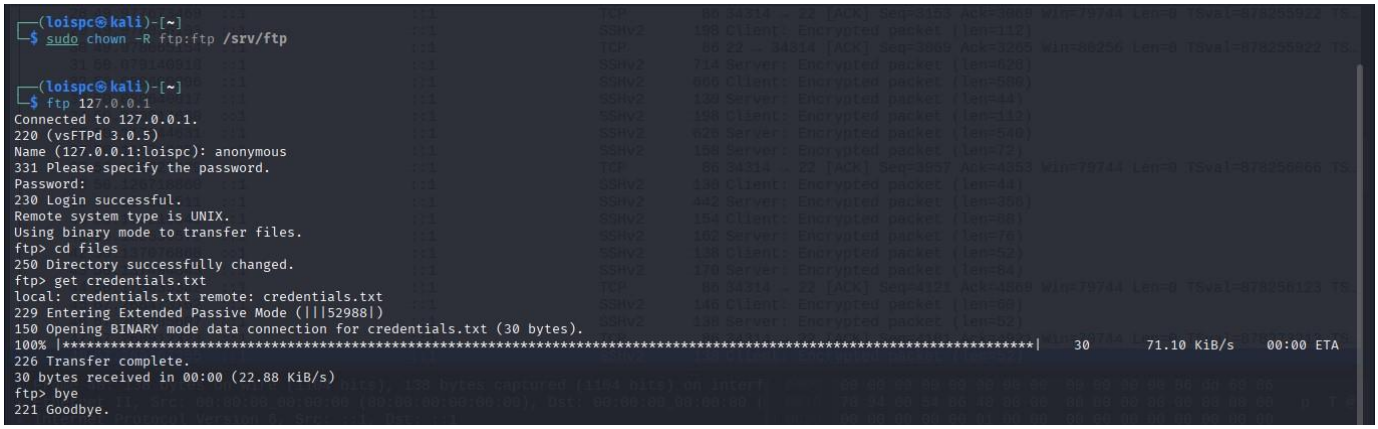
b) FTP file transfer procedure

The FTP client connected using anonymous login and downloaded the file using the get command after navigating to the appropriate directory

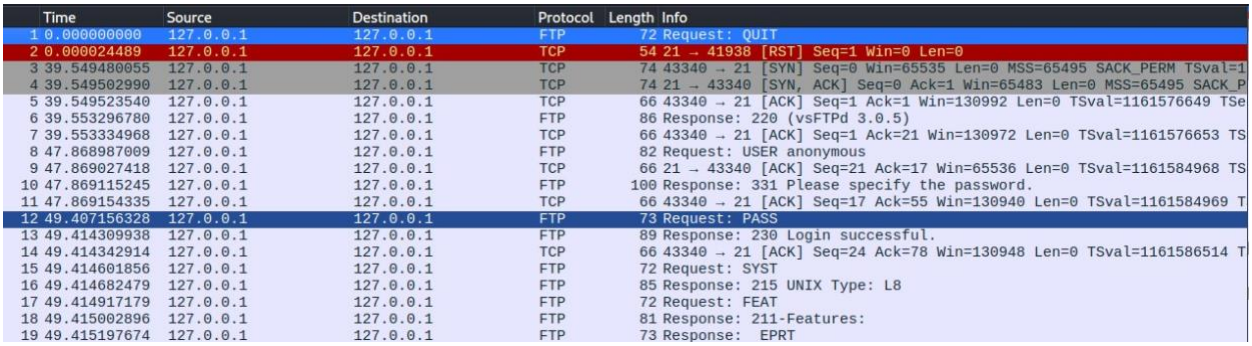
c) FTP traffic capture

Wireshark was used to capture FTP packets on the loopback interface with the appropriate filter. The FTP session showed all traffic in plaintext, including login credentials and file contents.

i. Terminal showing FTP session and get command

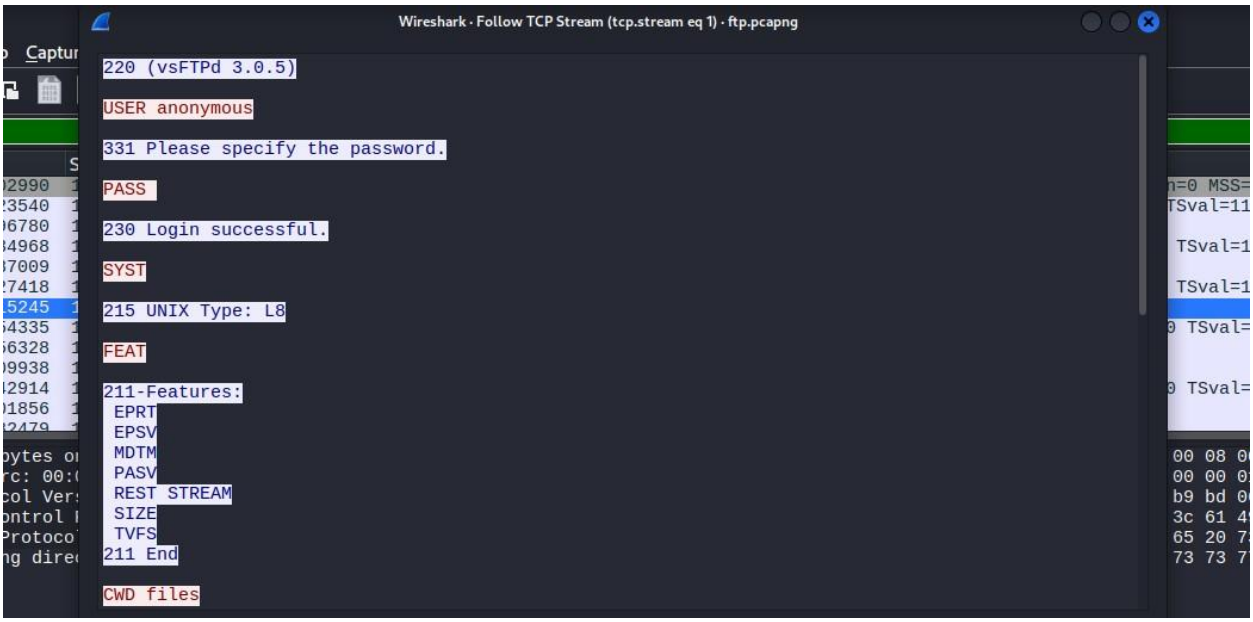


ii. Wireshark showing USER, PASS, and RETR commands



	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	127.0.0.1	127.0.0.1	FTP	72	Request: QUIT
2	0.000024489	127.0.0.1	127.0.0.1	TCP	54	21 → 41938 [RST] Seq=1 Win=0 Len=0
3	39.549480055	127.0.0.1	127.0.0.1	TCP	74	43340 → 21 [SYN] Seq=0 Win=65535 Len=0 MSS=65495 SACK_PERM TSval=1
4	39.549502990	127.0.0.1	127.0.0.1	TCP	74	21 → 43340 [SYN, ACK] Seq=0 Ack=1 Win=65483 Len=0 MSS=65495 SACK_P
5	39.549523540	127.0.0.1	127.0.0.1	TCP	66	43340 → 21 [ACK] Seq=1 Ack=1 Win=130992 Len=0 TSval=1161576649 TSe
6	39.553296780	127.0.0.1	127.0.0.1	FTP	86	Response: 220 (vsFTPd 3.0.5)
7	39.553334968	127.0.0.1	127.0.0.1	TCP	66	43340 → 21 [ACK] Seq=1 Ack=21 Win=130972 Len=0 TSval=1161576653 TS
8	47.868987009	127.0.0.1	127.0.0.1	FTP	82	Request: USER anonymous
9	47.869027418	127.0.0.1	127.0.0.1	TCP	66	21 → 43340 [ACK] Seq=21 Ack=17 Win=65536 Len=0 TSval=1161584968 TS
10	47.869115245	127.0.0.1	127.0.0.1	FTP	100	Response: 331 Please specify the password.
11	47.869154335	127.0.0.1	127.0.0.1	TCP	66	43340 → 21 [ACK] Seq=17 Ack=55 Win=130940 Len=0 TSval=1161584969 T
12	49.407156328	127.0.0.1	127.0.0.1	FTP	73	Request: PASS
13	49.414309938	127.0.0.1	127.0.0.1	FTP	89	Response: 230 Login successful.
14	49.414342914	127.0.0.1	127.0.0.1	TCP	66	43340 → 21 [ACK] Seq=24 Ack=78 Win=130948 Len=0 TSval=1161586514 T
15	49.414601856	127.0.0.1	127.0.0.1	FTP	72	Request: SYST
16	49.414682479	127.0.0.1	127.0.0.1	FTP	85	Response: 215 UNIX Type: L8
17	49.414917179	127.0.0.1	127.0.0.1	FTP	72	Request: FEAT
18	49.415002896	127.0.0.1	127.0.0.1	FTP	81	Response: 211-Features:
19	49.415197674	127.0.0.1	127.0.0.1	FTP	73	Response: EPRT

iii. TCP stream showing credential.txt in plaintext



3. SFTP SETUP AND ANALYSIS

a) SFTP configuration

SFTP was tested using the built-in OpenSSH server. The existing user account 'loispc' was used to transfer a test file named credential.txt located in the /srv/ftp/files/ directory.

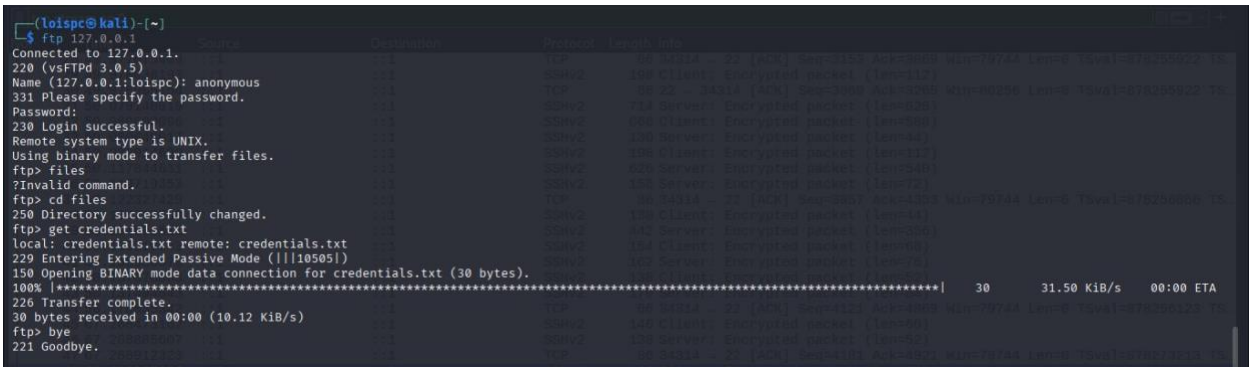
b) SFTP file transfer procedure

An SFTP session was established and the file was downloaded securely using the get command. The SSH fingerprint was verified upon first connection.

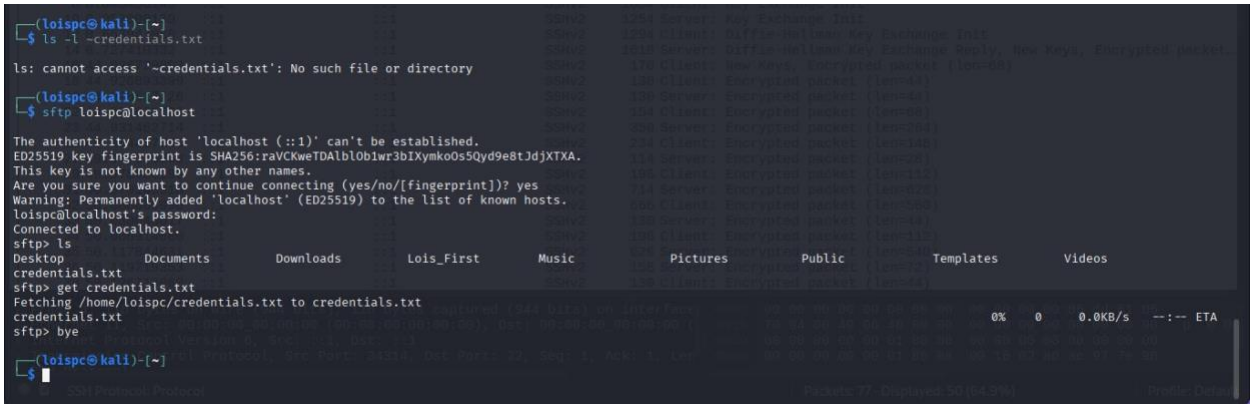
c) SFTP traffic capture

Wireshark was used to capture SSH traffic. The SFTP session was encrypted, and no credentials or file content were visible in the packet capture.

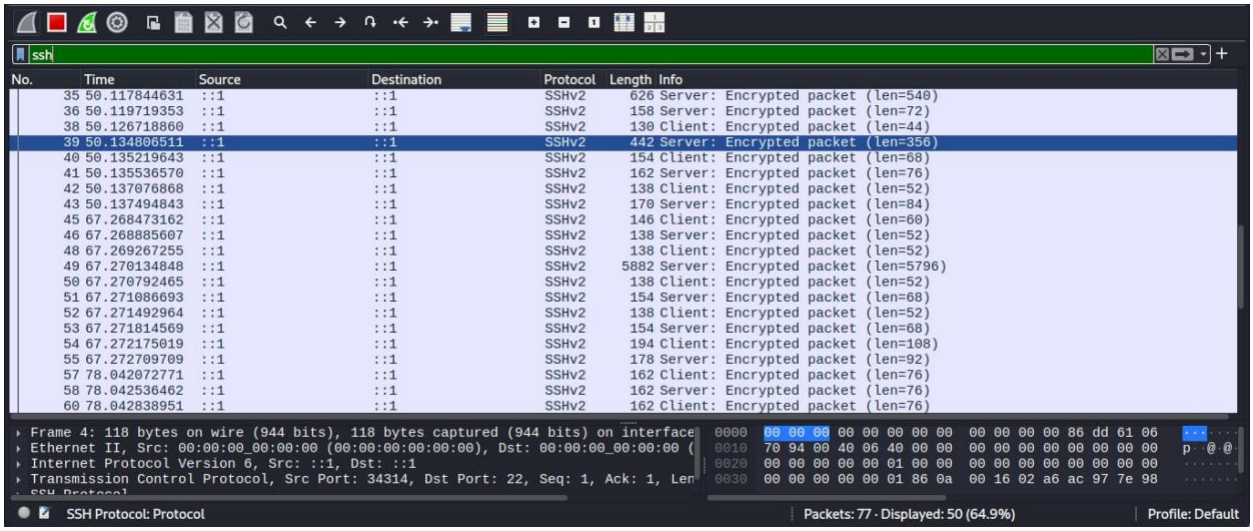
i. SFTP showing file transfer



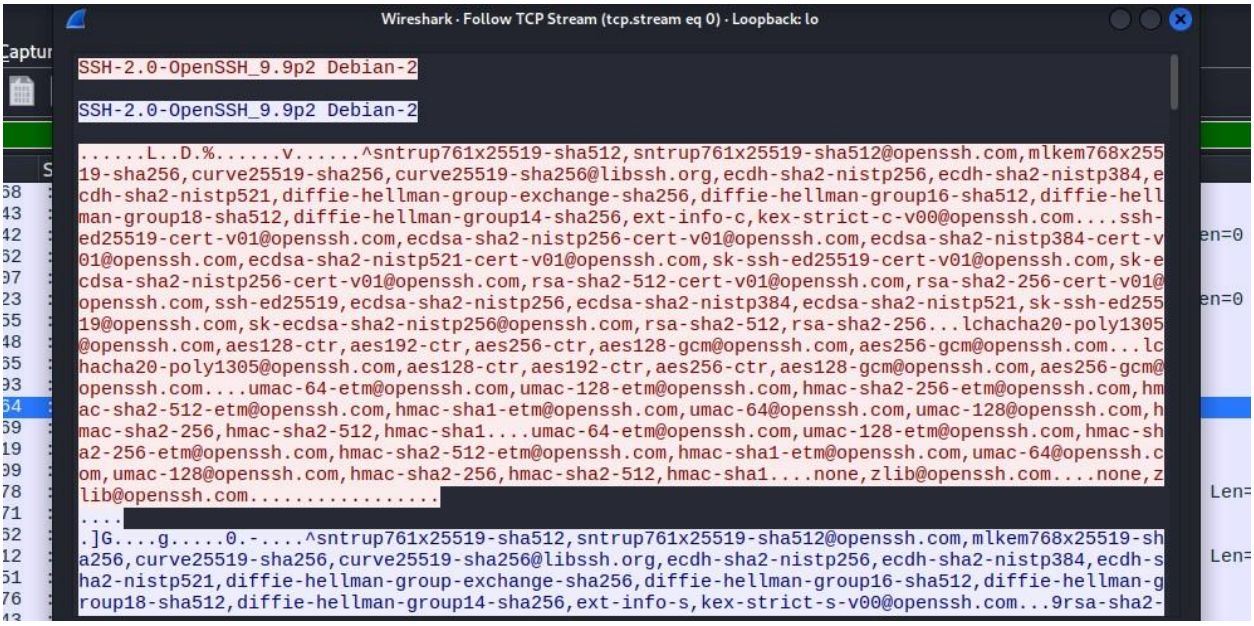
ii. SSH fingerprint prompt



iii. Wireshark showing SSH encrypted packages



iv. TCP stream showing encrypted content



4. REAL WORLD IMPLICATIONS OF FTP AND SFTP

FTP should not be used for transferring sensitive data across networks due to its lack of encryption. SFTP provides a secure alternative by encrypting both the authentication and data transfer process, making it suitable for use in compliance-regulated environments such as finance and healthcare.

5. REFLECTION AND LESSONS LEARNED

This project provided practical experience with network services and packet analysis. Challenges such as permissions errors, chroot restrictions, and interface selections in Wireshark reinforced the importance of understanding both system and network layers in cybersecurity.

6. CONCLUSION

This project successfully demonstrated the difference between insecure and secure file transfer methods. FTP was shown to expose sensitive data, while SFTP protected data through encryption. The hands-on approach provided a deeper understanding of protocol-level security.