FTP (File Transfer Protocol) and SFTP (Secure File Transfer Protocol)

FTP (File Transfer Protocol) and SFTP (Secure File Transfer Protocol) are **both protocols used for transferring files between a client and a server over a network**. However, they differ significantly in terms of security and functionality.

FTP (File Transfer Protocol)

- **FTP is a standard network protocol used to transfer files between computers over a TCP-based network, such as the Internet**.
- **Port 21**: FTP traditionally operates on port 21, which is used to establish the connection between the client and the FTP server. Once connected, the file transfer process can begin.
- **Security**: FTP is **not secure by default**, as it transmits data, including login credentials, in plaintext. This makes it vulnerable to eavesdropping and attacks like packet sniffing.
- Active vs. Passive Modes:
    - In **active mode**, the **server initiates the connection** to the client for data transfer on port 20.
    - In **passive mode**, the **client initiates both the control connection and the data connection to the server**, which helps to navigate firewall restrictions more easily.

SFTP (Secure File Transfer Protocol)

- **SFTP** is a **secure version of FTP that operates over SSH (Secure Shell)** to provide encryption and security during file transfers.
- **Port 22**: SFTP runs on **port 22, the same port as SSH**, ensuring that all communications (including file transfers and login credentials) are encrypted.
- **Security**: Since SFTP is built on top of SSH, it **provides strong encryption, making it much safer than FTP for transferring sensitive data over a network.**
- **Functionality**: While FTP and SFTP both allow file transfer, SFTP also supports additional features such as file access, file modification, and directory listing commands, all securely over an encrypted channel.

Key Differences Between FTP and SFTP

1. 8Ports

- **FTP**: Uses **port 21** for control and can use port 20 for data in active mode.
- **SFTP**: Uses **port 22** because it runs over the SSH protocol.

2. Security:

- **FTP**: Transfers data in **plaintext and is vulnerable to interception**.
- **SFTP**: Encrypts all data, making it **much more secure**.

3. Authentication:

- **FTP**: Typically uses username and password in plaintext.
- SFTP: Uses SSH for authentication, which can involve username/password or public key authentication for added security.

Example Use Cases

- FTP: May be used in scenarios where **security is not a concern**, or inside private, controlled networks.
- SFTP: Preferred for **secure file transfer**, especially when sensitive data is involved or when operating over public networks.

Summary

- **FTP**:8 An older, less secure protocol used to transfer files over port 21.
- **SFTP**: A secure alternative built on SSH, operating over port 22, encrypting all data transmissions for security.