**Lab 12: FTP Configuration and Implementation using Packet Tracer**

**Theory**

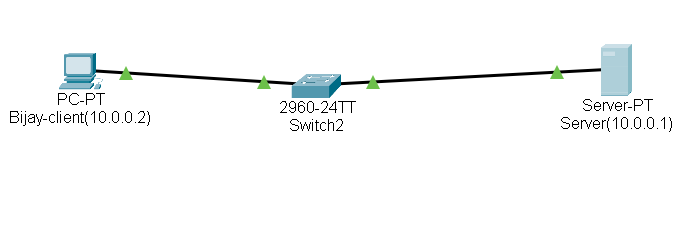
**FTP (File Transfer Protocol)** is a standard network protocol for transferring files between a client and a server over a TCP/IP network. It uses port 21 for control commands and port 20 for data transfer in active mode. Clients can access servers either anonymously or with authentication. FTP operates in active and passive modes, depending on who initiates the data connection. For secure transfers, FTPS or SFTP can be used to encrypt data. FTP supports file operations like uploading, downloading, and managing files with commands such as RETR, STOR, and DELE. In ASCII mode, it converts line endings between different operating systems, while in binary mode, it preserves the exact byte sequence of files. Despite its functionality, FTP is less secure compared to modern protocols due to its lack of built-in encryption, making FTPS and SFTP preferable for sensitive data transfers.

**Key Concepts of FTP**

1. **Client-Server Model:** FTP follows a client-server architecture where the client initiates requests for file operations, and the server responds. Clients can request files, upload files, or perform other file management tasks.
2. **Ports:** FTP uses port 21 for control commands and port 20 for data transfer in active mode. In passive mode, the server assigns a dynamic port for data transfer.
3. **Active and Passive Modes:** In active mode, the client opens a port for data transfer, and the server connects to it. In passive mode, the server opens a port, and the client connects to it, aiding in firewall and NAT traversal.
4. **Authentication:** FTP can operate in anonymous mode (no password required) or authenticated mode (username and password required), helping manage access and security.
5. **FTP Commands:** Common commands include LIST (view files), RETR (download files), STOR (upload files), DELE (delete files), and MKD (create directories).
6. **Data Types:** FTP supports ASCII mode for text files (converting line endings between systems) and binary mode for non-text files (preserving exact byte sequences).

**FTP Security:** FTPS (FTP over SSL/TLS) and SFTP (SSH File Transfer Protocol) are used to secure FTP connections, protecting data from interception and unauthorized access.

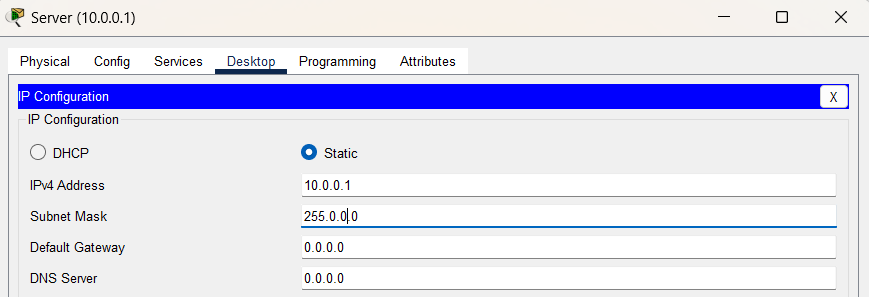
**Network Diagram**

*Fig:Network diagram*

**Configuring FTP Server and FTP Client**

**FTP Server Configuration**

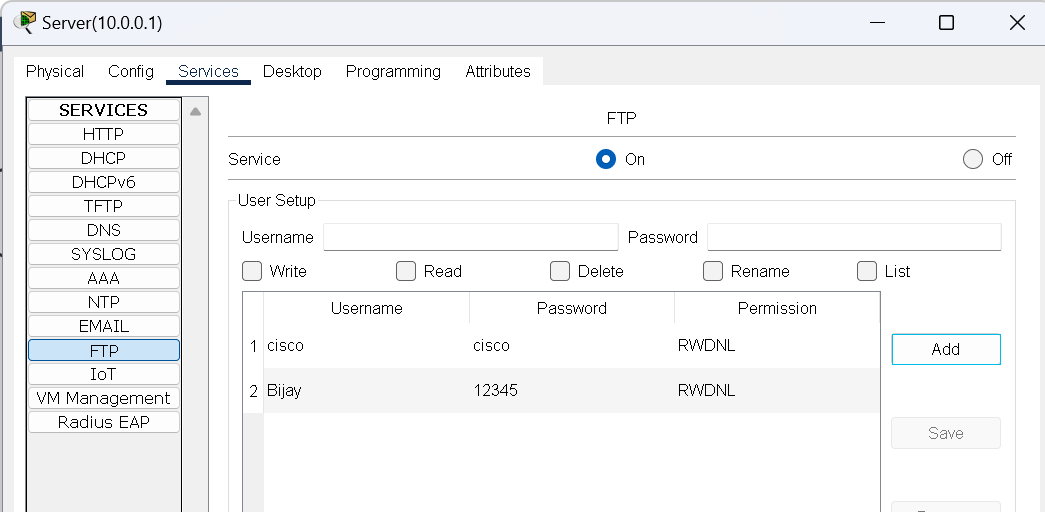
**Step1:**Click on server and go to ip configuration and set ip address and subnet mask.



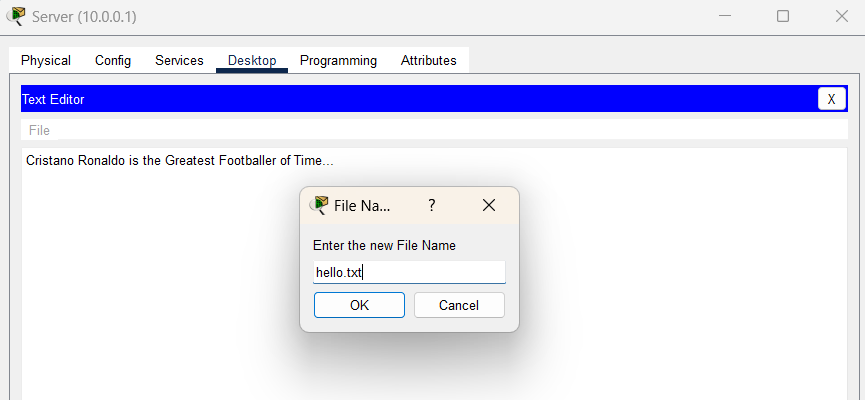
*Fig: IP configuration on Sever*

**Step 2:**Click on server, go to service, click ftp and click on ON button.

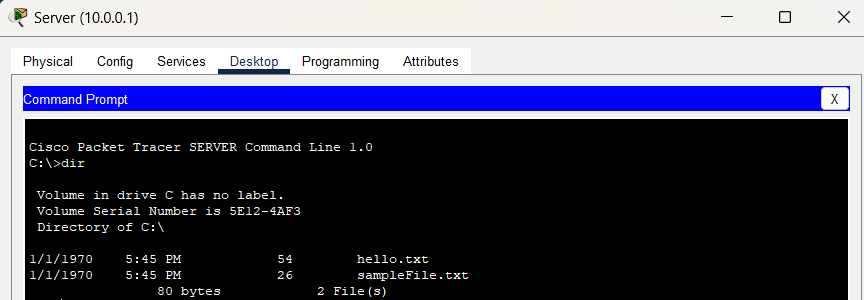
**Step 3:**Set username, password ,and tick write, read, delete, rename and list. Click add.

*Fig: Server Configuration*

**Step 4:** Go to desktop, click text editor write something and save the file as hello.txt.

*Fig: Creating a file name hello.txt*

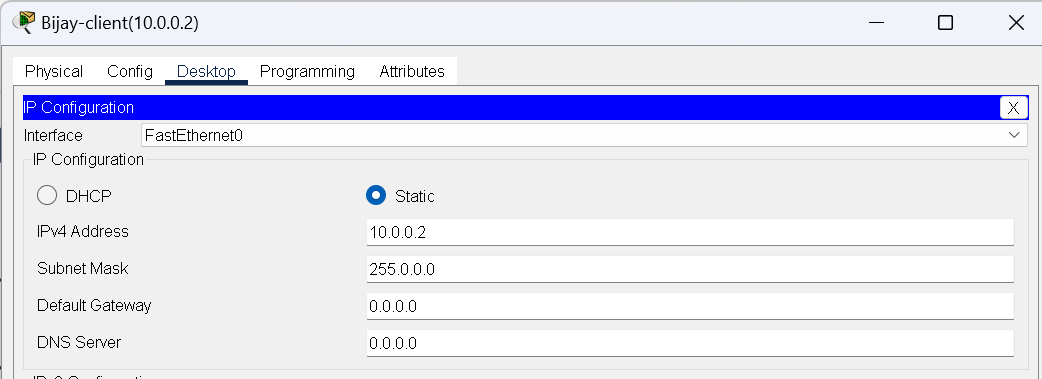
**Step 5:**In desktop, open command prompt and type dir command, we can see the file.

**

*Fig: Using dir command to see file*

**FTP Client Configuration**

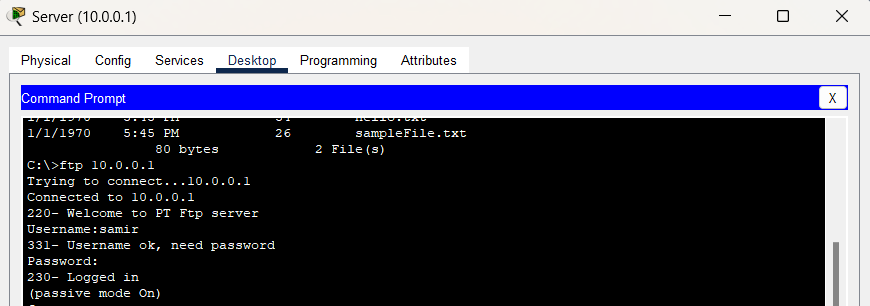
**Step 1:**Click on pc and go to ip configuration and set ip address and subnet mask.

**

*Fig: IP Configuration in Client*

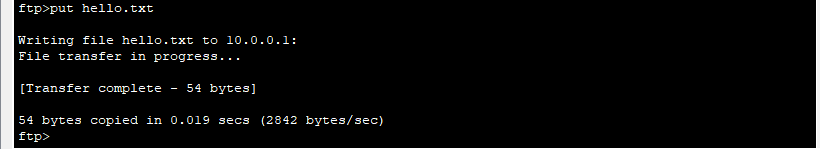
**FTP Server Connection**

In command prompt type command ‘ftp 10.0.0.1’then insert username and password, We will be connected to ftp server.

*Fig:FTP Server Connection*

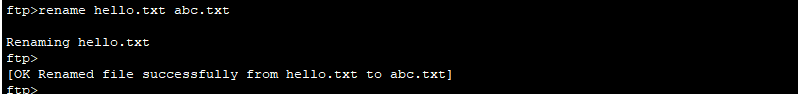
**TRANSFERRING FILE USING PUT COMMAND**

**Command:** Ftp > put hello.txt

*Fig: Transferring file using PUT comman*d

**RENAME FILE**

**Command:** Ftp > rename hello.txt abc.txt



*Fig:Renaming file*

**GET THE FILE AND SAVE THE COPY ON OUR MACHINE**

**Command:** Ftp > get abc.txt



*Fig: Saving copy of file in PC*

**GO TO PC**

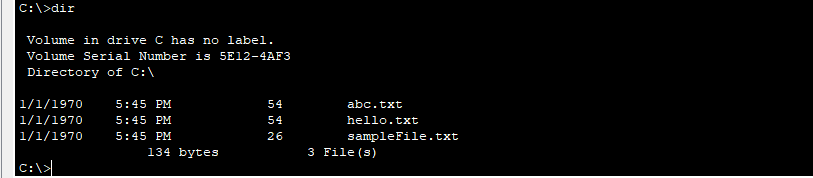
**Command:**Ftp> quit ftp



*Fig:Quiting FTP*

**DISPLAYING THE FILES**

**Command:** PC > dir

*Fig:Displaying the files*

**Conclusion**

In this lab, we successfully configured an FTP server and client using Packet Tracer. By following a step-by-step process, we learned how to set up IP addresses, enable the FTP service, and create user accounts with appropriate permissions. We also created a file, verified its existence through command-line operations, and demonstrated file management using FTP commands. This practical implementation enhances our understanding of the FTP client-server model, the role of IP addressing, and the importance of secure file transfer operations in network environments.