ABSTRACT

File Transfer Protocol is used for transferring files to clients over networks using client-server architecture. FTP provides two mechanisms for file transfer; one is anonymous method and another one is password authentication mechanism. All the communication between client and server is without encryption means data is transferred in clear text whether it is password or ftp commands. There are some requirements which are considered important while file transfers and these are Authentication, Integrity and Confidentiality. For implementing security in file transfer protocol we use FTPS rather than FTP as it is more secure. As FTPS uses some encryption mechanisms, it adds some extra process which effects the performance. FTP and FTPS are configured according to the security requirements for file transfer. Adding some extra process overhead in FTPS like encryption, it affects the performance. This research paper compares both FTPS and FTP on Linux and Windows server.

Chapter 1

INTRODUCTION

A server is a system that responds to requirements across a computer network to make available, or assist to provide, a network examination. Servers also run on the same hardware as a computer does, the only difference is the services and the programs it provides to the client. In many situations, computers offer many services and it has series of servers in sequence.

Servers generally work as client-server architecture. Servers provide the necessary program in order to work with new users. As a result the server performs many tasks in place of users. Generally the users communicate with the server by using the network but there is one possibility that the server run on the same system. Server is a program which also acts as a socket listener.

Servers often present required services over a network, both to private users within a huge organization or else to public users. Usual computing servers are application server, file server, print server, web server, database server, mail server, gaming server, or various other types of servers.

Several systems employ the client / server network model as well as email services and Web sites. Another model, peer-to-peer networking enables the entire computers to perform as both server and client as required.

1.1 Types of Servers

1.1.1 Application Servers

At times referred to as a kind of application servers take up a huge amount of computing area among the end user and database servers. Middleware is software which connects two or else divides application. There are various middleware goods that connect a database system to Web server. It allow for data from the database which user

demands using forms and it also allows the server to return vibrant Web pages based on top of the user's needs and profile.

The word middleware is used to explain different products to facilitate as the bond among two applications. Therefore, it is different from import and export features with the purpose to build it into one of the applications. Middleware is at times known as plumbing for the reason that it joins two sides of an application and transfers data trapped between them. General Middleware category includes:

- Message Passing
- TP monitors
- Object Request Brokers (ORBs)
- DCE environments
- RPC systems
- Database access systems

1.1.2 Audio/Video Servers

Audio/Video servers convey multimedia capabilities by enabling broadcast streaming multimedia content in Web sites. Streaming is a method for sending data in such a way that it can be processed as continuous steady stream. Streaming technologies are very important with the expansion of the Internet as most users do not have speedy access to download big multimedia files speedily. Through streaming, the client browser starts showing the data prior to the whole file has been transmitted.

For using streaming technology, the client getting the data should be capable to gather the information and transmit it as a stable stream and this steady stream is then used by application where it is then processed the data and further changes it to pictures or sound. It concludes that if client receives more data speedily than required, then there is a need to save the extra data and we save it in a buffer. Moreover, the data doesn't arrive fast enough then the appearance of the data to the user will not be even or we can say smooth.

There are many streaming technologies rising at this time. To send audio data, there is one i.e., de facto standard is best for Progressive Network's RealAudio.

1.1.3 Chat Servers

Chat servers facilitate a huge amount of users to exchange data in the surroundings comparable to Internet newsgroups to facilitate real-time conversation capabilities. Here real time means happening instantly. The term real time is used to illustrate various computer features. For example, the real-time operating systems are those systems which react to the input instantly. These systems are useful for tasks such as navigation, and for this the computer is required to respond to a continuous flow of fresh data without any break. Many daily useable computers are not real time as they take few minutes to respond to the client.

Real time means that the time to complete an event is same in real as the time taken by the computer. One of the example is graphics animation in this a real-time program displays objects at the similar rate as that they would move in real.

1.1.4 Fax Servers

Fax servers are useful where we want to send original documents to other place and also want to reduce the telephonic conversations.

1.1.5 Groupware Servers

These servers are software’s that is planned to allow users to work together, despite of place, with help of the Internet or a company Intranet and also to work as one in a virtual environment.

1.1.6 IRC Servers

These are used when we want real-time capabilities, and this server is made up of different networks at different locations and it allows users to hook up with one other using an Internet Relay Chat server network.

1.1.7 List Servers

List servers present a method to better handle mailing lists; either they are one-way lists with the purpose to convey announcements, newsletters or interactive debate open for all.

1.1.8 Mail Servers

Mail server's is used to store and save mails above company networks using LANs and WANs and also on the Internet.

1.1.9 News Servers

News servers are servers that perform as a distribution and delivery resource designed for many public news groups presently available above the USENET information system. USENET is a wide-reaching report panel method which can be used all the way through the Internet or by various other online services. The USENET is having near about 14,000 forums called newsgroups and they cover all possible concern group. It is accessed by thousands of users throughout the globe.

1.1.10 Proxy Servers

Proxy servers are used between an external server (typically another server on the Web) and a user system or program usually a Web browser to check requests, improve performance, and share connections.

1.1.11 Telnet Servers

A Telnet server provides a facility to users to log on to a remote computer and carry out jobs as if they work on the remote system itself.

1.1.12 Web Servers

Web server serves static and dynamic content to a Web browser. It loads file from disk and allocate it throughout the network to user's Web browser. They use HTTP mediate to exchange data between browser and server.

1.1.13 FTP Server

File Transfer Protocol is a archetypal network protocol worn to transfer files as of single host to another host in intemperance of a TCP-predestined network, like Internet. FTP is assemble on client-server propose as well as utilize separate categorize plus data connections amongst the consumer with the server. FTP patrons may authenticate themselves through a clear-text sign-in method, habitually in the manifestation of a username with password, additional can bond secretly if the server is configured to sanction it. For confined communication to encrypt the username and password, as well as encrypts the content, FTP is regularly cosseted through SSL/TLS ("FTPS"). SSH File Transfer Protocol ("SFTP") is from time to time as well used in its place, but is technically dissimilar.

As a consumer, we can use FTP with an easy command line edge (for case, as of the Windows MS- DOS punctual window) or by a profitable agenda that present a graphical consumer line. Our net browser can also construct FTP requirements to download agenda we choose as of a Web page. Using FTP, we can as well inform (delete, rename, move, and copy) records at a server. We require retrieving an FTP server. Though, openly accessible files are simply admission by means of unidentified FTP.

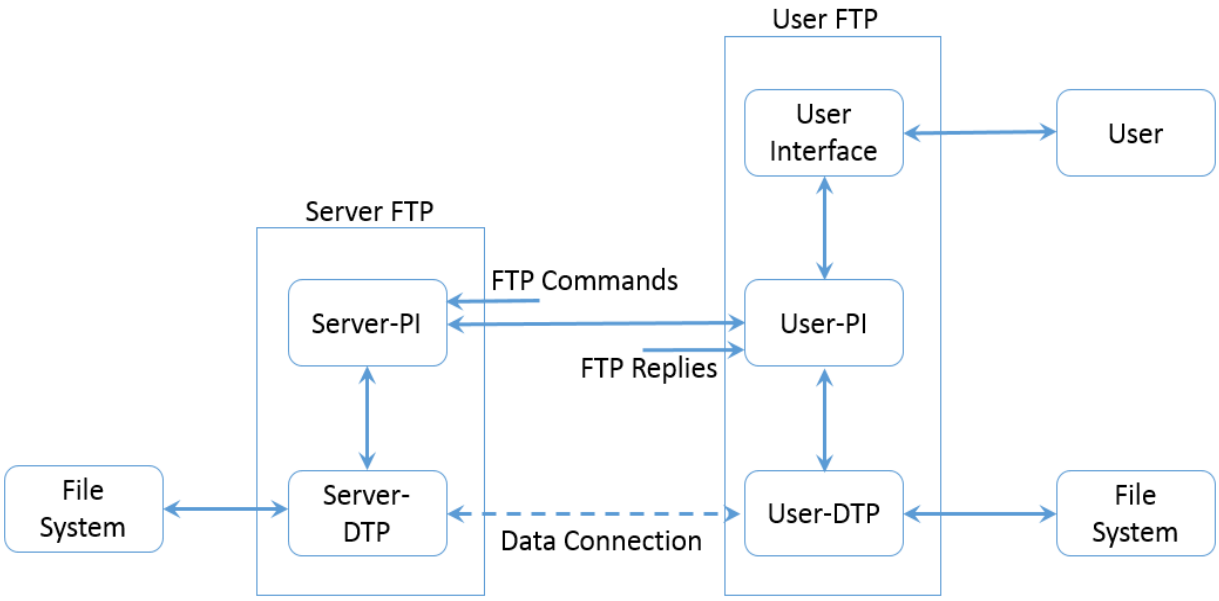


Fig: 1.1 working of FTP server

1.1.13.1 *Connection Methods*

FTP sprint above the Transmission Control Protocol (TCP). Typically FTP servers pay attention on the renowned port number 21 (IANA-kept) for inward links from customers. A link to this seaport as of the FTP client forms the manage watercourse on which instructions are approved to the FTP server and reply are composed. FTP make use of out-of-band organize; it unlock devoted data links on additional port information. The strictures for the statistics streams depend on the particularly demanded transportation mode. Data links typically use port numeral 20.

Active mode:

FTP consumer unfasten a active port, throws the FTP server the active port amount through which it got connected and wait for a link from the FTP server. Once the FTP server connects to the data link layer of the FTP client it connects to port 20 on the FTP server.

Passive mode:

The FTP server release a dynamic port, compel the FTP client using the server's IP address to connect to the port using which we can pay attention more than the organize brook and stay for a link from the FTP consumer. In this pencil case, the FTP clients attach the foundation port of the link to a active port. To utilize passive form, the customer uses the direct link to throw a PASV control to the server plus then accept a server IP address plus server port figure as of the server, which the consumer after that uses to release a data association from an random consumer port to the server IP address in addition to server port numeral received. Equally methods were modernized in September 1998 to prop up IPv6.

Login:

FTP login exploits a standard username and password method for yielding access. The username is throws to the server with the USER authority, plus the password is drives with the PASS control. If the data offer by the customer is established by the server, server will transmit a salutation to the customer and the meeting will begin. If

server supports it, client may log in with no provided that login qualifications, except the similar server could allow only inadequate contact for such sessions.

Anonymous FTP:

Host that present an FTP examination may offer unknown FTP contact. Users usually log into the examiner with an unknown account whilst provoked for user name. Even though users are normally requested to throw their email address as an alternative of a code word, no confirmation is in reality executed on the complete data. Lots of FTP hosts whose reason is to offer software updates will permit unknown logins.

1.1.13.2 *Role of FTP protocol*

FTP protocol tells us about how the data can be transferred on to network.
The main points of FTP protocol are:

- It allows user to use remote computers using programs
- To protect a user from duplications while file is stored among hosts,
- To encourage file sharing
- To send data consistently and efficiently. FTP can be used openly by a user using a terminal but it is designed mainly for use by programs.

1.1.13.3 *The FTP model*

FTP protocol uses a client-server model it means client sends instructions and the server awaits instruction to perform desired actions.

When we open an FTP link, there are two transmission channels which are in use:

- A control channel for commands
- And a channel for data transfer

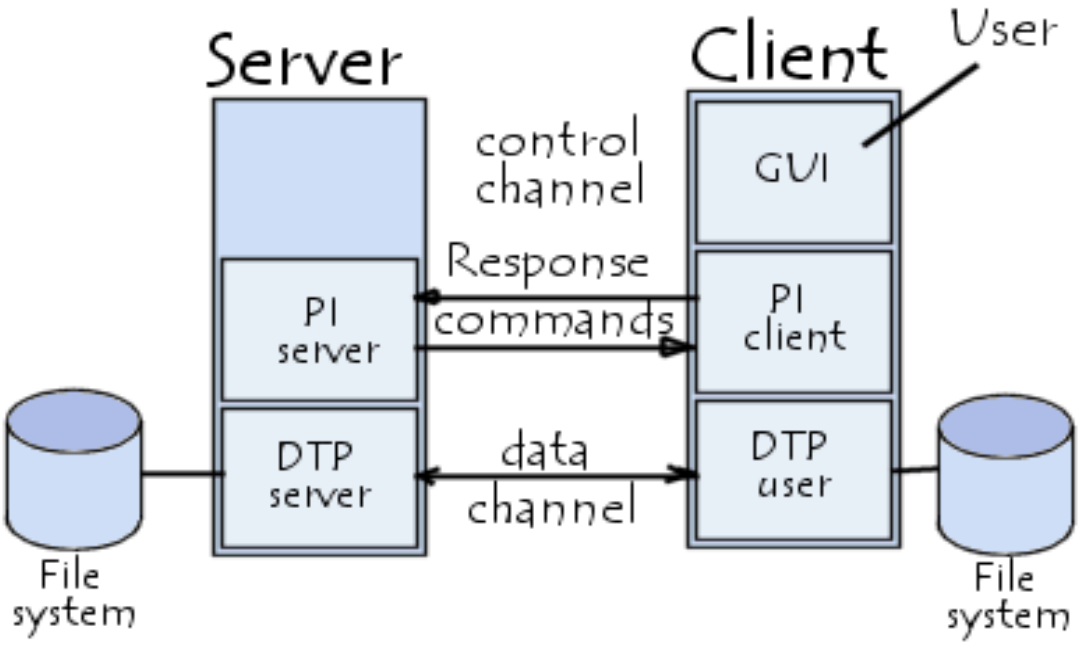


Fig: 1.2 FTP Model

There are two processes running on both the client and server to manage the information sent over the channel:

DTP (Data Transfer Process):

The data transfer process is mainly used for the establishment and link organization over the channel. The DTP used at client side is called USER-DTP and DTP used at server side is called SERVER-DTP.

PI (Protocol Interpreter):

The Protocol Interpreter is different for client and server. The main work is to check for instructions which are over the link by the control panel used for the data transfer process control.

The instructions received from USER-PI over the control channel are analyzed by the server-PI. The control channel is the main part for proper communication between the server and the client. USER-PI gives instruction over the channel and SERVER-PI has to forward those instructions over the channel. For connection establishment with the ftp

server USER-PI is in charge. It also performs several other functions like controlling USER-DTP, transmitting FTP commands and getting replies from the SERVER-PI.

The USER-PI is the first part which communicates with the ftp server. When connection is established the USER-PI starts communication over the channel using ftp commands. When the client sends FTP commands to the server; then the server reacts accordingly. The server after getting request runs its DTP and then sends a reply to the client. Now, for sending the response the server opens a port and server-PI made a connection with client. The client DTP then starts listening on the port opened by the server for any incoming data.

There are two channels for data transfer using ftp, the control and data channels. These both work on different ports. For data transfer the control channel is used first and then the data is transferred using the data channel on the specified ports.

During data transfer the control channel in this type of configuration will remain open. And if the control channel is broken through transmission then the server can stop the transmission.

1.2 Data Representation and Storage

Transmission of data between sender and receiver happens for a storage space and sometimes it becomes important to make some alterations information generated from data to remove the remedies from the data transferred in the process. This ramification process makes information more suitable for sender and receiver. NVT-ASCII is a good example of diversified storage organization.

When transmission is done for different sort of word length in between sender and receiver in form of binary many difficulties occurs. Sending process is not unique all time so sender has to take care of it. To send a 32 bytes information in form of 36 bytes information, system is required to have right justification of 32 bit words in 36 bit word and the transmission should be of quality with efficiency. Data representation and

transmission needs to be clarified and user must be aware of it and what modifications are required in functions. FTP provides very limited representation of data. User required doing this alteration on their own if needs to make other transformations different from the limited capacity of the FTP.

1.2.1 Data Types

Data types which are accessed in FTP by a client tell us about a type of the data transferred. These types may be defined as ASCII or EBCDIC as well as in Local byte as a byte size and these are also known as "logical byte size." Here, one point is noticed that there is nothing related with size of byte which is transmitted over the link, known as “transfer byte size” and also these two are not complex. The logical byte size of ASCII is of 8 bits. If we are talking about the Local byte then in this the logical byte size is specified in second parameter of the data command send over the link.

1.2.1.1 *ASCII Type*

This is used mainly for transferring the text files in only one case that they don’t transfer data using EBCDIC technique. It is the most common and default kind and is used by all FTP implementations.

The data transferred by the sending server converts it from internal data technique to the all known and useful NVT-ASCII form and it is of 8 bit. And after the data is received by the receiver, it is converted from the NVT-ASCII to the form it used for its internal use. To use with the regular NVT method, we use a sequence <CRLF> which is helpful to indicate the end of a line in a text file. If we are using NVT-ASCII technique signifies that the transferred data between the devices is of 8 bytes. The data types used in ASCII and EBCDIC are briefly discussed.

1.2.1.2 *EBCDIC Type*

It is mainly used for internal representation of data while the data is transferred between the two clients. The data transmitted over the channel is shown as 8-bit EBCDIC representation. The only variations in EBCDIC and ASCII types are the code used.

1.2.1.3 *IMAGE Type*

In this the data transferred are in continuous bits and these are grouped into 8 bit long bytes. The data received by the client must be stored as adjoining bits. The storage organization must consist of padding for each file and of each record at some level like word or byte. The padding used is either of adding of all zeros or all ones and are used at the end of the file and if we are considering of records then at end of records. If file is restored then we must know about padding to get back the original file if we don't know the padding added to file then we never retrieve the original file. When the data is transferred over a channel then we must have the padding information to process the file. Image type is mainly used to transfer binary data or to store the binary data. Image type is recognized and is used by the entire ftp using protocols and techniques.

1.2.1.4 *LOCAL Type*

When we transfer any data, then it is grouped into logical bytes and these bytes are of the size which is specified by the Byte size parameter. There is no default value set for the byte size parameter but the value of it must be a decimal integer. Nowhere is it defined that the transfer byte size must be same as logical byte size. If there is any difference between the byte sizes of these two, then the logical bytes should be grouped in contiguous packets. It does not care about the transfer byte limits. It also adds some padding at the end of a file if necessary. Once the data is transferred then the specific host transforms the data according to the logical byte size. We can get the same file by using this process if we use the same parameters for the same file and it will be used by all the techniques.

1.2.2 Format Control

ASCII and EBCDIC types will accept an optional second parameter and it is to explain if we are using any kind of vertical format control which is linked with a file. There are many types of data representation techniques.

When we transfer any file to a client then we have three things in our mind or we have three reasons for it and these are:

1. Either for printing purpose

2. Or for storing data
3. And last is for processing or for retrieval of file.

First we talk about printing, when a file is transmitted to a client then the client must know about the vertical format representation of the data. Now if we talk about storage, then there are chances that we can get exactly the same file in same form once we transferred it to a client. And at last if we are moving a file from any one client to another then it is possible that we can process the file without any errors. Any single format techniques do not fulfill all the conditions of the user.

1.2.3 Data Structures

The structure of a file tells us about many things. It will also influence the transfer mode of a file as well as the understanding and storage space of the file. The structure of a file is default if we are not considering the STRU command. The file structure must use both file and record structures for all the transmissions occurred over a channel by all ftp protocols.

There are various representation methods; there are three file structures implemented in FTP and it also defines the structure of a file.

Page-structure:

It is the place where the file is constructed and the file consists of the self-sufficient indexed pages.

Record-structure:

In this section the file is a collection of sequential records or ordered records.

File-structure:

In this the file contains sequence of data in continuous form and it has no internal structure.

1.3 Data connections in FTP Server

First of all you have to make a suitable connection among the ports and select the appropriate parameter for that. Then you will be able to transmit the data. Consumer as

well as the server-DTPs has their unique defined ports. Both the user process and control connection ports are identical; apart from the user the server has adjoining to control connection port.

The size of the transmitted data is 8-bit bytes. Importance of the size of the data in sonly for the pure transmission of the data; it doesn't comportment on demonstration of the data inside a host's file system.

The first transmit request command is usually listen by the same port as pervious in inactive data transfer method. The route of the data transmit is verified by the FTP request control. The data connection will be start to port by the server which gets the earlier request. After establishing the connection the user-PI gets the authentication reply because of the data transfer begin among DTP and Server-PI. Only the user-PI can work on non default port but other must have their default port to work.

The PORT command gives the client information of alternating data port. The user-PI establishes data connection between both server-PI. The other server is commanded to listen the work of the parallel server. Demonstrating the other server port user-PI send the PORT command to Server A the last both send the suitable commands to respective port. The progression report is define in segment on FTP respond of it totally depends on the server whether it want to seal or begin the connection.

In general, it is the server's dependability to keep the data connection to begin it and to seal it. If the block data try to transmit by the user-DTP it shows an error. The data transmission should be blocked in the following circumstances.

1. The server has done its work.
2. The server accepts an ABORT authority from the client.
3. The port requirement is misused by a authority from the client.
4. The control connection is blocked officially or otherwise.
5. A disregarded fault condition takes place.

Or else the shut is a server choice, the use of which the server should point out to the user-process by any a 250 or 226 replies simply.

1.4 Data Connections Managements

Defined ports of the Data Connection:

FTP should be acknowledged how to use avoided data association ports, the User-PI defines how to utilize the non-default ports.

Defined ports of Non-Default Data:

The non -default data port control is identified by the user-PI using PORT control. To categorize a non-default server side data the user PI would ask data port through the PASV control. Since an association is definite by the couple of addresses, moreover of these events is sufficient to get a dissimilar data association, at rest it is allowed to do both guidelines to utilize new ports.

Reuse of the Data Connection:

To transfer data over a channel, the stream of data broadcasted on it the end fraction of the file has to be displayed by concluding the alliance. We can face problems if we want to transfer many files at a time. At many times the TCP connection will not be able to respond at one time. These are also helpful to transfer large data files easily over a channel.

There are two ways to describe this thing. First is to use a port other than default and another one is to use a different method to transfer the data.

There are various methods to transfer the data between client and server. In these one is Stream transfer mode, the main advantage is that it is random in nature and it does not establish connection if the channel got congested. There are various modes by which we can transfer data in sort form like Block and compressed modes. These modes also provide faster data transfer rates. These are also having such encoding in the protocol that

client will know the end of file and for this it will use some kind of escape characters. Using, these methods the data transfer provides secure way to transfer the data.

1.5 Transmission Modes

Next deliberation in transmitting data is choosing the suitable communication method At hand three methods: one which designs the data as well as permits for restart events, solitary which also condenses the data for resourceful transmit; and solitary which surpasses the data with small or no dispensation. In this last casing the style interrelates with the organization characteristic to conclude the type of dispensation. In the compacted method, the demonstration type verifies the stuffing byte.

The different modes of transmissions and is as follows:

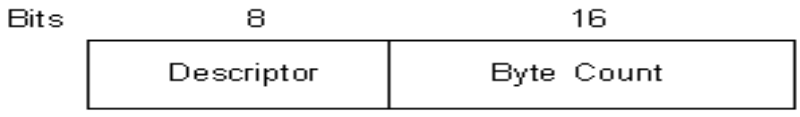
1.5.1 Stream Mode

The byte form the data is allows in it. There is no limitation on illustration in present; only the proof structured are allowed. EOR and EOF are précised by two-byte organizes code. The First byte should be one, second may contain zero. If the data is departing in the sequence of all ones, it must be repetitive in the subsequent bytes.

In this mode the end of file is indicated by a sequence character and is responsible for better transmission of data over the channel.

1.5.2 Block Mode

This is the method in which data is transferred in blocks. The header of each block is shown below:



FTP Block Header

There are some codes used by the ftp protocols:

- 16 – This part is used as Restart Marker

- 32 – This part is used for errors in the block
- 64 – This part indicates EOF
- 128 – this block indicates EOR

The Byte Count Field shows the number of bytes in the data block.
The descriptor code divides in two parts, one is end portion in the file (EOF) and another one is end part of the record (EOR).

1.5.3 Compressed Mode

In this mode the information is transmitted as regular data, transmitted in a byte string and as compressed data.

Compressed mode is used to compress data for better transmission rates and for better quality.

1.6 FTP Commands

FTP commands are of three types:

1. Access control commands
2. Transfer parameter commands
3. FTP service commands

Table: 1.1 Access Control Commands

Access control commands	
Command	Description
USER	Character string allowing the user to be identified. User identification is necessary to establish communication over the data channel.
PASS	Character string specifying the user's password. This command must immediately precede the USER command. It falls to the client to hide the display of this command for security reasons.
ACCT	Character string representing the user's account. The command is generally not necessary. During the response accepting the password, if the response is 230 this stage is not necessary, if the response is 332, it is.
CWD	Change Working Directory: this command enables the current directory to be changed. This command requires the directory's access path to be fulfilled as an argument.
CDUP	Change to Parent Directory: this command allows you to go back to the parent directory. It was introduced to solve problems of naming the parent directory according to the system (generally "..").
SMNT	Structure Mount:
REIN	Reinitialize:
QUIT	Command enabling the current session to be terminated. The server waits to finish the transfer in progress if the need arises, then supplies a response before closing the connection.

Table: 1.2 Transfer Parameter Commands

Transfer parameter commands	
Command	Description
PORT	Character string allowing the port number used to be specified.
PASV	Command making it possible to indicate to the DTP server to stand by for a connection on a specific port chosen randomly from among the available ports. The response to this command is the IP address of the machine and port.
TYPE	This command enables the type of format in which the data will be sent to be specified.
STRU	Telnet character specifying the file structure (F for File, R for Record, P for Page).
MODE	Telnet character specifying data transfer method (S for Stream, B for Block, C for Compressed).

Table: 1.3 FTP Service Commands

FTP service commands	
Command	Description
RETR	This command (RETRIEVE) asks the server DTP for a copy of the file whose access path is given in the parameters.
STOR	This command (store) asks the server DTP to accept the data sent over the data channel and store them in a file bearing the name given in the parameters. If the file does not exist, the server creates it, if not it overwrites it.
STOU	This command is identical to the previous one, only it asks the sever to create a file where the name is unique. The name of the file is returned in the response.
APPE	Thanks to this command (append) the data sent is concatenated into the file bearing the name given in the parameter if it already exists, if not, it is created.
ALLO	This command (allocate) asks the server to plan a storage space big enough to hold the file whose name is given in the argument.
REST	This command (restart) enables a transfer to be restarted from where it stopped. To do so, the command sends the marker representing the position in the file where the transfer had been interrupted in the parameter. This command must immediately follow a transfer command.
RNFR	This command (rename from) enables a file to be renamed. In the parameters it indicates the name of the file to be renamed and must be immediately followed by the RNTO command.
RNTO	This command (rename to) enables a file to be renamed. In the parameters it indicates the name of the file to be renamed and must be immediately followed by the RNFR command.
ABOR	This command (abort) tells the server DTP to abandon all transfers associated with the previous command. If no data connection is open,

DELE	This command (delete) allows a file to be deleted, the name of which is given in the parameters. This command is irreversible; confirmation can only be given at client level.
RMD	This command (remove directory) enables a directory to be deleted. The name of the directory to be deleted is indicated in the parameters.
MKD	This command (make directory) causes a directory to be created. The name of the directory to be created is indicated in the parameters.
PWD	This command (print working directory) makes it possible to resend the complete current directory path.
LIST	This command allows the list of files and directories present in the current directory to be resent. This is sent over the passive DTP. It is possible to place a directory name in the parameter of this command, the server DTP will send the list of files in the directory placed in the parameter.
NLST	This command (name list) enables the list of files and directories present in the current directory to be sent.
SITE	This command (site parameters) causes the server to offer specific services not defined in the FTP protocol.
SYST	This command (system) allows information on the remote server to be sent.
STAT	This command (status) makes it possible to transmit the status of the server, for example to know the progress of a current transfer. This command accepts an access path in the argument, it then returns the same information as LIST but over the control channel.
HELP	This command gives all the commands understood by the server. The information is returned on the control channel.

FTP responses

The FTP response is the way by which synchronization is ensured among the FTP server and client. Every instruction sent by the user is analyzed by the server and it will perform an action and send back a reply.

The response consists of a 3 number code shows the means in which the instruction is being processed by the client. The 3 numbered response code is difficult for humans to read so each code is described with a text.

The responses from the client consist of 3 digits and the meaning is as follows:
The first digit shows whether the connection is successful or not.
The second digit shows about what the response from client refers to.
The third digit provides additional information compared to second number.

Table: 1.4 First Number Code of FTP Response

First number		
Digit	Meaning	Description
1yz	Preliminary positive response	The action requested is in progress, a second response must be obtained before sending a second command
2yz	Positive fulfillment response	The action requested has been fulfilled, a new command can be sent
3yz	Intermediary positive response	The action request is temporarily suspended. Additional information is awaited from the client
4yz	Negative fulfillment response	The action requested has not taken place because the command has temporarily not been accepted. The client is requested to try again later
5yz	Permanent negative response	The action requested has not taken place because the command has not been accepted. The client is requested to formulate a different request

Table 1.5 Second Number Code of FTP Response

Second number		
Digit	Meaning	Description
x0z	Syntax	The action has a syntax error, or is a command not understood by the server
x1z	Information	This is a response sending back information (for example a response to a STAT command)
x2z	Connections	The response relates to the data channel
x3z	Authentication and accounts	The response relates to the (USER/PASS) login or the request to change the account (CPT)
x4z	Not used by the FTP protocol	
x5z	File system	The response relates to the remote file system

1.7 Secure Socket Layer and Transport Layer Security

Secure file transfer over the Internet revealed SSH (Secure Shell, also known as Secure Socket Shell) and SSL (Secure Socket Layer) as the two current primary options used for secure file transfer communications. As of this writing equally viable products exist for enhanced secure file transfer using either SSH or SSL. Both SSH and SSL can accomplish session traffic encryption and connection authentication using industry standard encryption algorithms such as RSA key exchange and Triple DES.

SSH:

SSH was originally designed as a replacement for unsecured applications such as telnet, rlogin, rsh and ftp where usernames and passwords are sent in clear text across a network. It can also be used to securely “tunnel” other applications. The standard TCP/IP

port used for SSH is 22. SSH and its associated components are applications that can perform a variety of tasks.

SSL:

SSL was originally designed by Netscape Corporation, as an Internet browser add-on (as opposed to an "application" in the case of SSH) for secure web communications. SSL is a universally accepted standard for secure web based transactions such as credit card purchases and other ecommerce. It typically uses TCP/IP port 443.

TLS (Transport Layer Security) Protocol is used to provide privacy and data integrity between two communicating applications. SSL protocol provides a way to communicate client and server so that attacker can't tamper the message.

SSL itself is not an application. Operating at the Transport layer of the OSI (Open Systems Interconnection) model it provides services to other higher layer application protocols, functioning as an application independent method for confidential, authenticated, integrity based communication between applications. Figure below depicts SSL/TLS in logical relation to other applications using the seven layer OSI model.

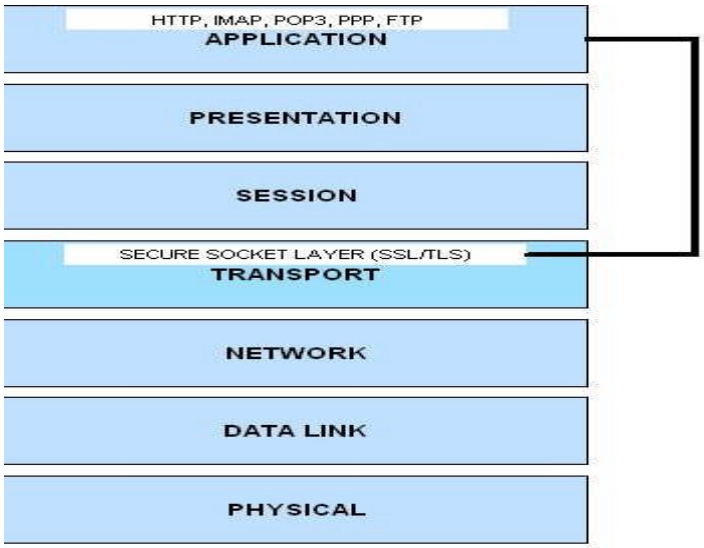


Fig1.3 SSL/TLS OSI Model Relationship

FTPS (File Transfer Protocol Secure)

FTPS protocol adds an extra layer of protection to the clients by implementing encryption and using some secure protocols like TLS protocol to transfer data over the TCP/IP network. FTPS is the upgraded version of FTP which removes the limitations of the FTP server. FTPS provides secure protocol support as provided by some other services like SMTP (Simple Mail Transfer Protocol Service Extension for Secure SMTP over TLS) and HTTPS (supports Transport Layer Security protocol for secure connection). FTPS overcomes the limitations of FTP server like eavesdrop ping, tampering and message forgery across the network. It supports full functionality for Secure Socket Layer (SSL) cryptographic protocol and Transport Layer Security protocol. It also implements the use of client-side certificates and server-side public key authentication mechanisms. It also supports well-suited ciphers for transferring data over the network including AES, Triple DES, DES and also some of hash functions such as SHA and MD5.

Typically one of two possible modes is used for FTP over SSL:

Explicit SSL/TLS –AUTH SSL, AUTH TLS: connection starts on standard FTP port 21, switches to SSL or TLS based on FTP client requesting SSL encryption via AUTH SSL or AUTH TLS command respectively. In Explicit Mode the clients have complete power on which areas of the link are to be encrypted.

Implicit SSL/TLS –FTP connection starts on a designated port (usually 990), SSL is started at the beginning of the connection. Explicit SSL should be used where standards compliance is mandated. In Implicit Mode, the entire FTP session is encrypted. FTPS was used in explicit mode in this research.

SFTP:

The SSH file transfer protocol or secure FTP (SFTP) is used to transfers files between client and server in a secure way. SFTP uses the Secure Shell protocol (SSH) to transfer files. FTP does not encrypt any data but SFTP encrypts the data transmitting between client and the server, as a result the sensitive information like password can’t be forged or tempered between transmissions. It is good alternative to FTP.

Chapter 2

LITERATURE REVIEW

Anand Srivastava Linux in excess of the past duo of years has developed to the position where it has been acknowledged as a practicable proposal for server applications. This renovation is due to its steadiness and shore up offered by a few corporations. At present it is being worn by Internet Service Providers. Linux will be established for additional grim applications merely if it can hold serious loads. This research examines the performance of Linux in one such request, the FTP server. A number of experimentations were demeanor to resolve the performance below diverse environments. Termination pedestals on these experiments are drained and prearranged in this research. Research prove that Linux execute relatively well below serious loads.[1].

T. Kiran This research explain the propose and accomplishment of a system that permits the news to contact files on remote anonymous FTP sites clearly. By transparency we represent that each and every files on each and every remote FTP sites in the planet emerge to be the part of the local file system tree and can be admittance by means of any of the recognizable Unix programs exclusive of need to change or still recompile these programs, This is accomplished by applying a new sort of file system, known as FTP file system, for Linux. The FTP file system include the file transfer protocol (FTP) within the kernel and build files on remote accomplish sites emerge as local files. It utilizes a disk cache to cache freshly accessed file. A user level method, known as cache daemon, occasionally removes cached files that gratify the system administrator individual criteria, in order to present some resemblance of cache rationality [2].

Roy Gregory Franks Client-server systems are developing more and more familiar in the world these days as clients shift to networks of distributed, interacting computers. This structure of job demands new recital models as the communications in client-server systems are additional complex than the categories maintained by classic queuing network solvers such as Mean Value Analysis. Layered Queuing Network is one

of these replicas; it uses hierarchical putrefaction and replacement setbacks to answer the model [3].

This research illustrates a fresh analytic modeling tool called LQNS (Layered Queuing Network Solver) which expands previous methods worn to model distributed client-server systems. The assistance of the theory are as trails. First, the form now supports forwarding. Forwarding is a method where a respond to a client is postponed to a lesser level server in a multi-level system, humanizing routine by reducing communication traffic. Forwarding can also be worn to renovate open models to closed models. Second, systems that use before time respond can be modeled. Early answers are worn to decrease the response time by responding to a client earlier than all of its work at a server is finished. Previous methods have been extensive to multi servers and to allow multiple clients. Third, actions have been set up. Activities characterize the minimum unit of modeling detail and can have subjective precedence associations. Finally, the solver has been unmitigated to hold models with both standardized and assorted threads inside a task. Standardized threads are worn to model multi servers. Assorted threads are worn to model fork-join communications such as asynchronous remote process calls and in RAID storage devices. The solver also integrates exactness enhancements for models with premature replies and for models with several layers.

The solver has been worn to investigate abundant systems found in continuation today including a tele-operator arrangement and a business processing system. In conclusion, a widespread presentation model of the Linux 2.0 Network File System (NFS) is obtainable [4].

M. Horowitz and S. Lunt In this paper we discuss the FILE TRANSFER PROTOCOL (FTP) and its security considerations. Many new authentication schemes are introduced in this paper. These extension mechanisms provide integrity and confidentiality while transferring the data using FTP server. These mechanisms are applicable on both channels i.e. the control and data channels.

New optional commands which were introduced in this paper are AUTH, ADAT, PROT, CCC, CONF and MIC. These commands increase the security part wile transferring the data over the network.

Dag Henning Liodden Sørbø The normal technique of transmitting files from a FTP server to clients is through TCP connections on the Internet. The whole file is transferred separately to each and every user using a uni cast link. The most common thing which happens is that clients download the same file at the same time within certain time interval. The whole time server sends several copies of the same data. This results in needless data sending within the system. This research focus on Cache Cast file server, which removes needless data sending by using CacheCast method. This method also helps in removing redundancy.

Cache Cast method is used when the same data section must be transferred to various users in a very short time frame. In a live streaming design, all clients overwhelming the similar video or voice stream are getting the same records synchronously. Thus, live streaming schemes can really profit from Cache Cast. During transfer in file server, the clients are not harmonized per se. Cache Cast sustain in a file server requires a unique system idea. The main idea in the Cache Cast server is to reorganize the file blocks prior to transmission, such that the identical file block is transport to numerous clients. Cache Cast is then capable to eliminate the unneeded data transfers.

This research consists of the design, implementation and appraisal of the Cache Cast file server. The method is executed in the ns-3 network simulator, in sort to execute experimentation in a network with dozens of customers. Three chief aspects of the scheme are appraise, specifically the belongings on the bandwidth expenditure in the system, the collision on the download time practiced by the clients, and the equality among alongside connected customers. The presentation of the Cache Cast file server is balanced against the presentation of an FTP server.

The assessment has exposed that the Cache Cast file server executes appreciably superior than an FTP server, which transport the files using TCP. It distributes the files quicker to the receivers, and trim down the total bandwidth expenditure in the network. In our research, the download time is concentrated by a cause of 10 and the bandwidth enthusiastic is 89 % less then when using an FTP server. These presentation gains are accredited to the Cache Cast sustain in the file server. The appraisal also shows that the Cache Cast files server guarantee fairness amongst challenging customers.

M. Allman and S. Ostermann The File Transfer Protocol (FTP) has many mechanisms which can make this protocol vulnerable to many network security issues. The FTP protocol is used to transfer files by the server to the client. The proxy FTP creates many security problems. The FTP protocol also allows the user to enter n number of attempts for entering the credentials such as username and password. Due to this issue an attacker can perform a password guessing attack commonly known as brute force attack. The research paper suggests the solutions for the security problems contained in FTP which also helps the system administrators to strengthen the FTP managed by them.

Chapter 3

IMPLEMENTATION

3.1 VMware

VMware is a corporation that was accepted in 1998 who offers diverse software and applications for virtualization. In these days, it has become one of the key contributors of virtualization software in the invention. VMware's software's be capable to classify in two levels: desktop applications along with server applications.

VMware is a virtualization as well as cloud computing software provider in support of x86-compatible systems. VMware Inc. has its head office in Palo Alto of California. VMware virtualization is base lying on the ESX/ESXi bare metal hypervisor, supporting virtual technology. The word "VMware" is time and again used in suggestion to particular VMware Inc. products for eg. VMware Horizon Application Manager, VMware v Cloud Director, VMware Workstation and many more.

Virtual machines are used widely in IBM architecture computers as they make possible to run an additional operating system and it runs in such a way that it appears as it is running on a different set of hardware. It helps people to install and analyze more than one operating system on a single hardware or on a single computer. VMware was founded by five special IT experts. The company formally launched its earliest product, in 1999 that was VMware Workstation, and after that VMware GSX Server in 2001. The company has started many advance applications since that occasion.
VMware's desktop software is friendly with all OSs, like Linux, Windows, and Mac OS X. VMware offer three miscellaneous type of desktop application:

- **VMware Workstation:** This creation is used to install and run a variety of copies or cases of the similar operating systems or besides diverse operating systems on a exacting physical PC.
- **VMware Fusion:** This was designed for Mac users plus offers extra compatibility between all extra VMware products.

- **VMware Player:** This was launched as freeware by VMware, intended for consumers who do not have accredited VMware software. This is software intended only for personal use.

VMware's product hypervisors planned for servers that can run openly on the server hardware with no need of an additional primary OS. VMware's procession of server software includes:

- **VMware ESX Server:** It is an enterprise-level illumination, which is constructing to show improved functionality in estimation to the freeware VMware Server substantial as of a small scheme overhead. VMware ESX is integrated with VMware vCenter to facilitate added results to get improved the manageability and consistency of the server achievement.
- **VMware ESXi Server:** This is similar to the ESX Server apart from that the examine console is change with BusyBox system as well as it necessitate terribly small disk space to control.
- **VMware Server:** It's a freeware application that can be worn over obtainable operating systems similar to Linux or Microsoft Windows.

3.2 Virtualization

Virtualization enables today's X86 computers on the way to run numerous operating systems and applications, building your communications simpler as well as more efficient. Applications catch deployed quicker, performance with availability swell and operations turn out to be automated, ensuing in that's easier to relate and not as a lot of expensive to hold plus handle.

3.2.1 Server Virtualization

The architectural blueprint of today's x86 server allows running one OS at a time. It also helps in twice the better resource utilization and also make possible in extra cost cutting as it provides same hardware to one or more operating system. It provides us the technique by which we can run many operating systems on a single piece of hardware or system. Virtualization technology is a key technology which helps us in developing

future technology. It also overcomes many other limitations and provides path for future development.

Server virtualization allows us to choose any server beyond its hardware limitations. Most of the servers in market dealing with low hardware quality which gradually decreases its performance.

VMware vSphere provide us the virtualization which helps in:

- 80 % better utilization of resources
- Up to 50 % declination in expenses
- 10:1 or higher server consolidation gain

3.2.2 Network Virtualization

Network Virtualization is a term which can be defined as software which can provide exact duplication of hardware for operating system. Network virtualized software's are helpful in initial stages of learning. They also provide an easy and cost effective way to build a network. These are also hardware independent and they provide large operational freedom to the client. These are stable networks and easy to handle as well as for debugging.

Network virtualization current sensible networking mediation plus services dependable ports, switches, firewalls, load balancers, routers, VPNs as well as more to connected workloads.

We can build a remarkably scalable network structure with the aim to provide enhanced levels ordered capability moreover nimbleness, QoS, previous provisioning and replica, through observing, error bugging, all this with protection by VMware network virtualization application.

VMware NSX will be the world's best system and safety virtualization strategy given that a full-service, programmatic in addition mobile virtual network worn for virtual equipment, organized on zenith of any ordinary purpose IP network hardware.

VMware NSX phase group all together the most excellent of Nicira NVP plus VMware vCloud Networking and Security devoted on one phase. VMware NSX represents a entire group of abridge logical networking rudiments plus services together by means of routers, firewalls, logical switches, VPN, monitoring bonus security.

3.2.3 Desktop Virtualization

Arranging desktops as a supervised renovate offer you the juncture to get exploit faster to unreliable requirements and occasion. You can reduce expenses and supplement overhaul by promptly and simply deliver virtualized desktops in addition to applications to separation workplaces, outsourced in addition to offshore workers in addition movable workers on iPad as well Android tablets.

VMware desktop explanations are scalable, firm, entirely protected and exceptionally accessible to assurance maximum uptime and efficiency. Diminish consumption and organization by transporting desktops as renovate. In attendance protected remote access to teleworkers in addition to temporary staff with no sacrifice recital.

3.2.4 Application Virtualization

Associations are steadily additional virtualizing of their Tier 1 operation important product submissions. This includes records, CRM, email, ERP, Java middleware, business intelligence, teamwork and lots of others.

In order to defend the needed stages of SLA and QoS for these Tier 1 business products in virtual surroundings, IT professionals must interest similarly on the virtualization tools of the arrangement additionally on the strong association and monitoring of virtualized business applications, and also on preserve profitable policy for commerce steadiness and tragedy recuperation.

These virtualized software's essentially run enhanced and give high convenience, tragedy recuperation, impetus and swiftness as well as cloud-readiness. During the

VMware Tier 1 request Virtualization answer construct on VMware vCloud Suite, you can enlarge the attribute of IT services distributed, while make simpler your transportation, exploit efficiency and get rid of it.

3.2.5 Storage Virtualization

Storage virtualization is division of the storage space on basis of software-defined level with the purpose to show improvement in the space effectiveness without any additional hardware for storage.

It have to permit quick stipulation so that high-performance, space-efficient storage space can be spun up as instantaneous as a Virtual Machine these days. It is necessary to offer a VM-centric space administration duplication that is intuitive for virtual administrator, and they are fulfilling on additional space organization jobs in virtual environment. And it has to bond equally with the hypervisor place to compel occupant, outstanding workflows.

Storage virtualization of VMware is grouping of ability that present a simplification layer for physical storage belongings to be administered, addressed and optimized in virtualization expenditure.

Storage virtualization skill offers a fundamentally improved way to manage storage possessions for your virtual transportation, giving your industry the potential to:

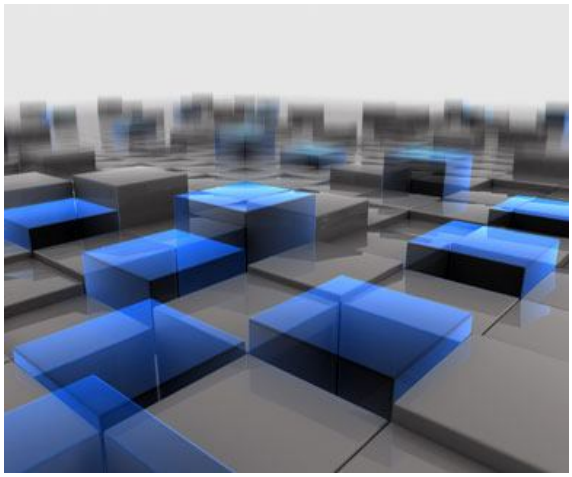
- expansively pick up storage supply operation along with suppleness
- abridge OS patching plus driver requirements, in nastiness of storage topology
- enlarge application uptime in addition abridge day-to-day function
- control and correspond your accessible storage setups

3.3 VMware Virtualization

Formulate effects easier for your IT communications throughout confirmed virtualization elucidations assemble on VMware vSphere in the route of Operations administration, the industry's important virtualization plus cloud management stage.

Virtualization help you to reduce assets cost during server consolidation and neat down operating cost all the way during computerization, while decrease missing revenue by dropping both considered and unexpected downtime. Diminish capital along with operational costs by growing energy efficiency along with requiring less hardware in the company of server consolidation.

Improve business continuity as well as disaster recovery capability for your virtualized infrastructure during improved and basic disaster recovery solutions with vCenter Site Recovery Manager.



Virtualized Tier 1 industry decisive endeavor applications, including databases, business applications, in addition to deliver the maximum SLAs and top performance. Achieve policy-based automation plus make sure compliance and performance by a zero-touch infrastructure with VMware v Center Operations Management Suite used for virtualization management.

Discover why the software-defined data center is the most excellent and most capable cloud infrastructure elucidation.

Reason for Choosing VMware for Virtualization

VMware virtualization solution are build on VMware vSphere, our confirmed, vigorous and consistent virtualization platform—and the pick of additional than 500,000 clients, together with the complete luck overall 100. It's modernization and brilliance have been known by strategic do research firms like Gartner, who position it in the leaders side of the Gartner, pleasant Quadrant is used for Server Virtualization interactions.

Paping:

Paping is a computer program used to check whether the host is alive or reachable over the network on a specified port. It is a network administration utility and helpful for

the network administrators to check the host for further reference. Its name is on another network utility tool ping.

The Internet Control Message Protocol (ICMP) is used to detect the operating system of a remote host over the network. This protocol has a limitation that the network administrators block it for security reasons. The ping utility can't be used when ICMP is blocked and the service cannot be determined.

Paping can be used on both windows and Linux systems. Once the paping is initiated it measures the time taken to establish the connection and also keep record of any failures in the connection. The final result includes connection time and it is in much summarized form. The services available on the network must keep their relevant TCP or UDP ports open to perform their task. The paping utility attempts to connect to the port we specified to determine if the service is available or not.

3.4 Windows Implementation

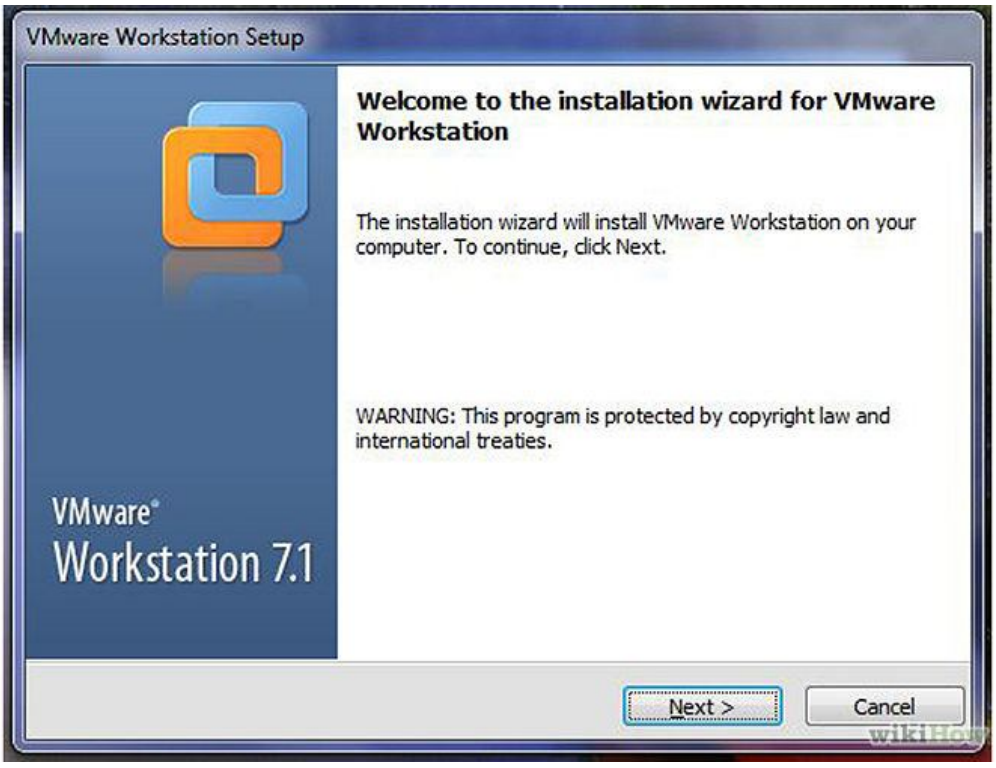
This is regarding how to set up a virtual machine on a PC. Virtual machine (VM) is a application completion of a mechanism (a computer) that implement software like a physical machine. Mainly people contain single computer. If you desire to set up a LAN or a mechanism to be small-scale testing, it's not sufficient. You as well desire to include both windows and Linux OS. Purchase a computer is not value it. Providentially, Virtual machine product can virtual a set of guests in a user computer.

Steps :

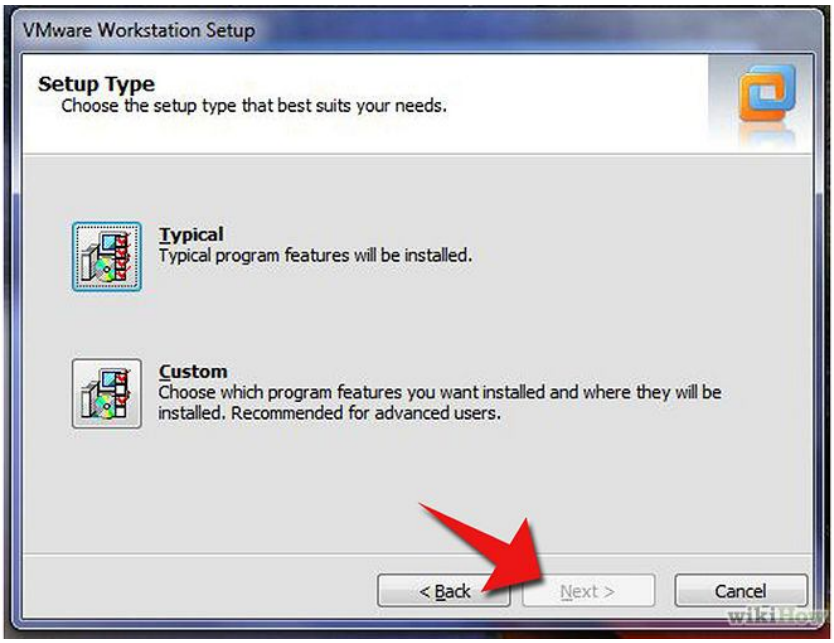
1. Installation of the VMware Workstation on Windows user computers.

Note: For installing the Workstation on a Windows 7 user computer, you need administrator privileges.

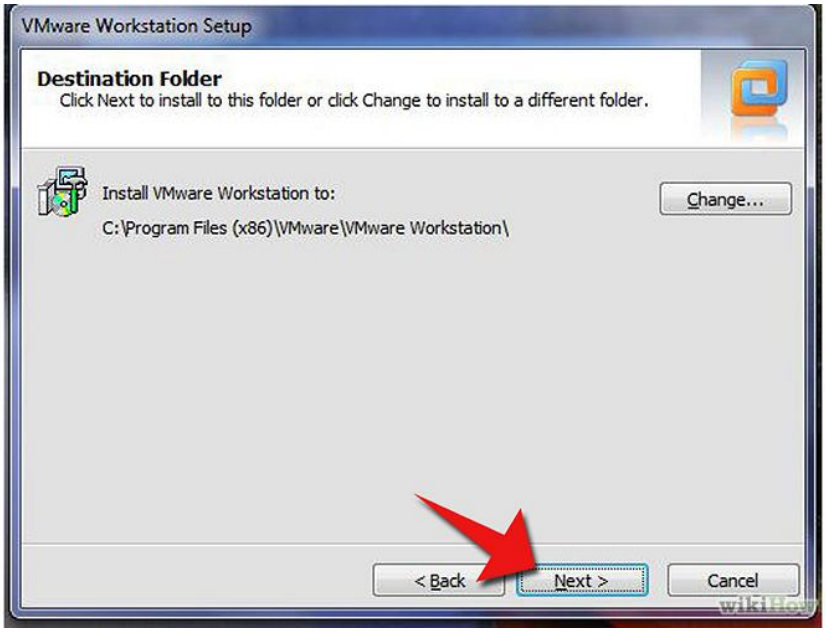
Click on the package of VMware Workstation package that you owned in a CD or in a package form.



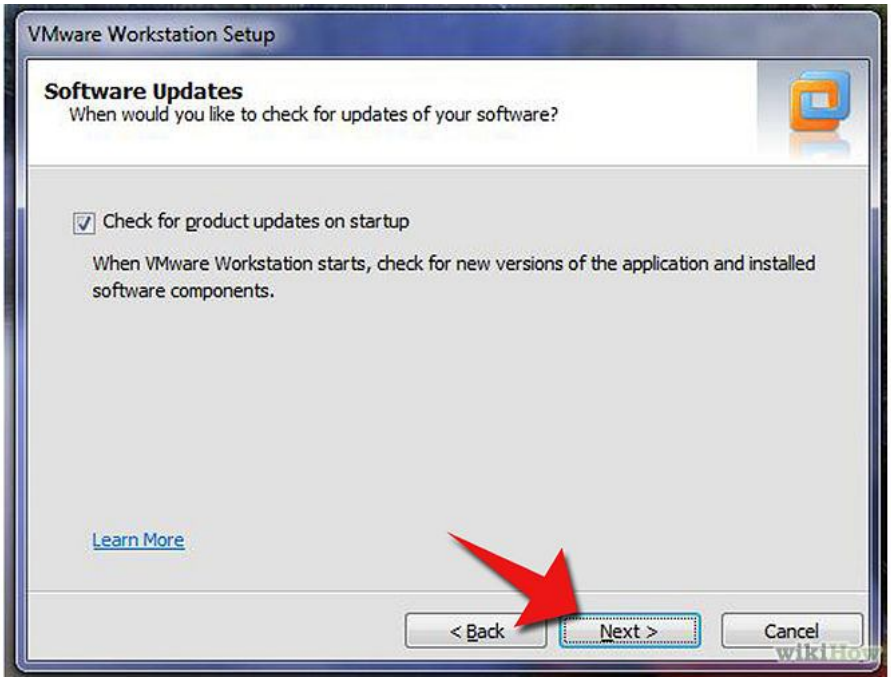
2. Click Next.



3. Select the set up type.

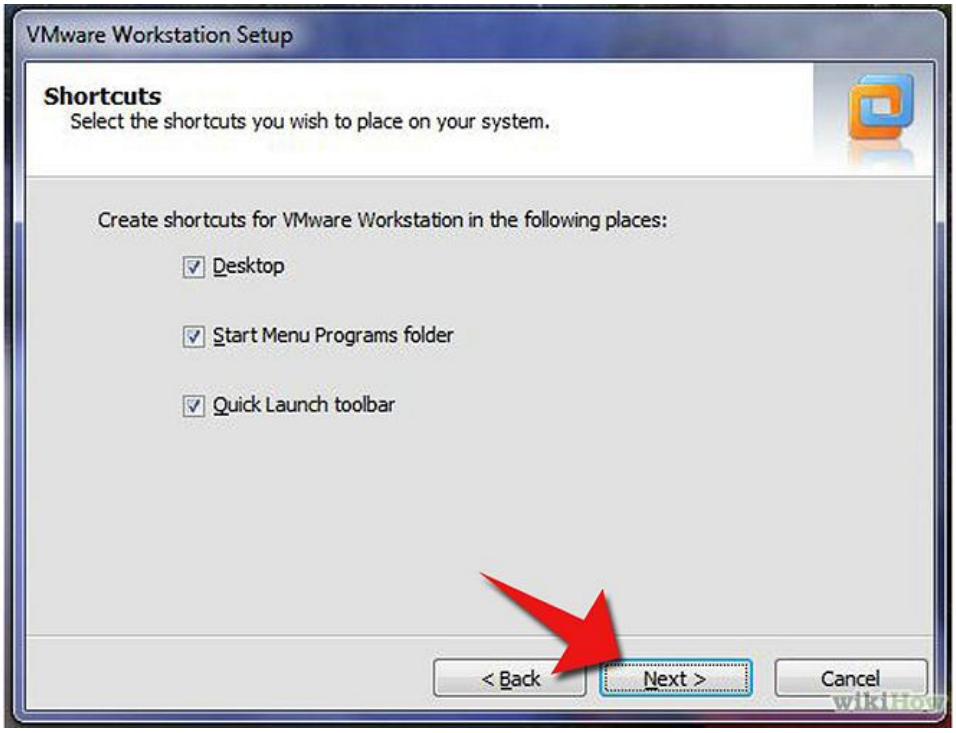


4. Select the directory in which you want to install VMware Workstation.

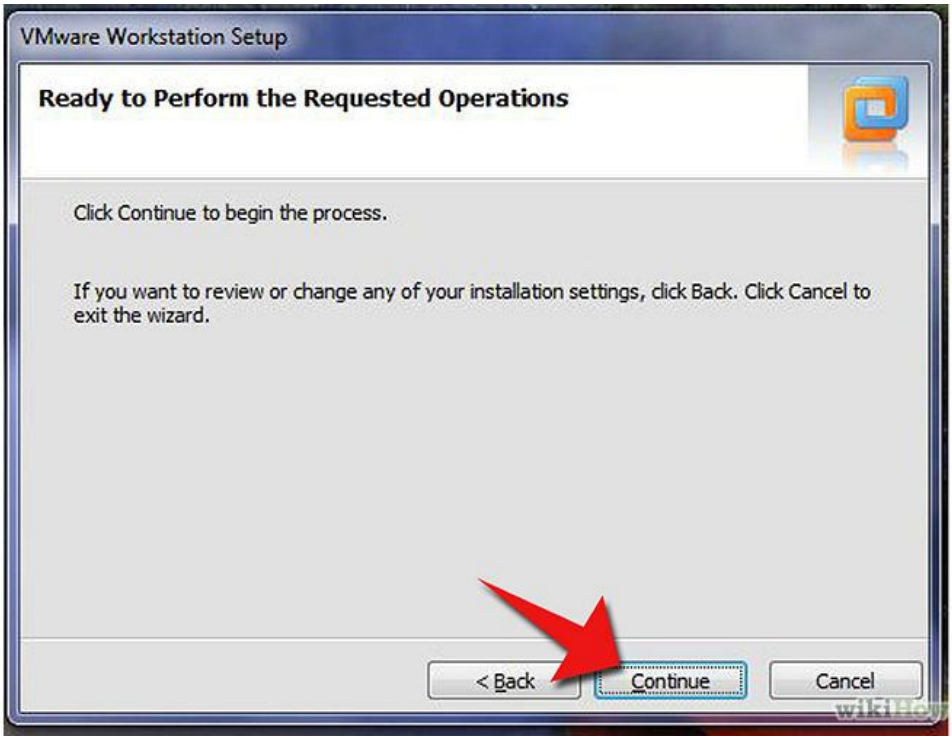


5. If we want to check for updates on startup then select this box. If not then uncheck this box.

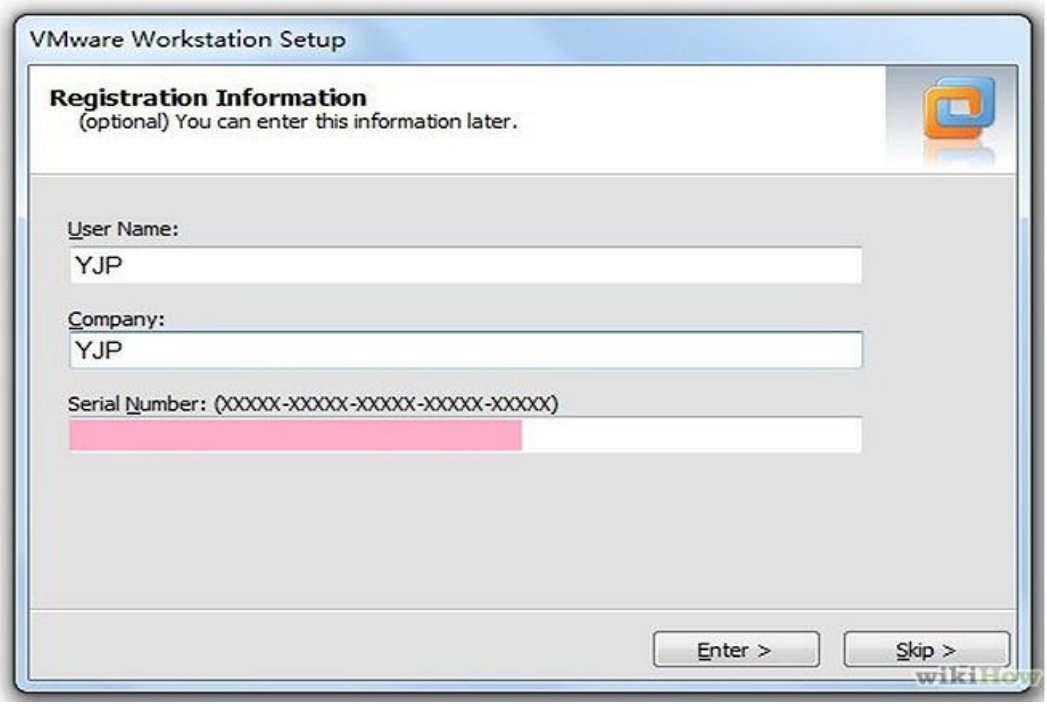
6. If you like to send feedback to VMware then select this box. Uncheck this if we do not want to send feedback to it. Then, Click on next button.



7. Select the boxes for which you want the installer to create shortcuts. The choices are Quick Launch toolbar, Desktop and Start menu program folder. Uncheck the shortcut if you do not want to create during installation.



8. Now, installer had gathered all the needed information and it is all set to start installing the software. If there are any changes or modifications you want to change it is the time to make those adjustments and for that Click Back button. If there are no changes to make then, proceed and click on Continue button. After this the installer starts installing files to PC.



9. When this window appears enter the serial number and click on Enter button.

10. After completion restart your computer.

11. Agree to the terms and conditions in the agreement option, and then click on Next button and at last click on Finish button.

3.4.1 Windows Server 2008 in VMware Workstation

For installing Windows Server 2008 we need a CD/DVD or an ISO file having server 2008 files. Before starting installation check the hardware architecture whether it is 64 bit or 32 bit. Then start installing the server according the CPU architecture.

1. Firstly, open VMware Workstation. Then click on New Virtual Machine.

2. There choose the default action and in next part check the radio button for typical virtual machine installation.

3. In next step choose to continue with default installation of Windows and then pick the Windows Server 2008 version.

4. Provide a name to the virtual machine for installation I named it "Win Server".

5. Allocate the disk to the Windows Server from the available memory.

6. The next step is to boot the Server.

7. Go through some of the settings like amount of RAM used.

8. Click the Power button to start the windows server installation.

9. After all this follow the standard procedure of Windows installation till it completes.

3.4.2 Cent OS in VMware Workstation

1. First step is to insert the operating system CD or else use ISO file.

2. Secondly click on Power button to power on the machine.

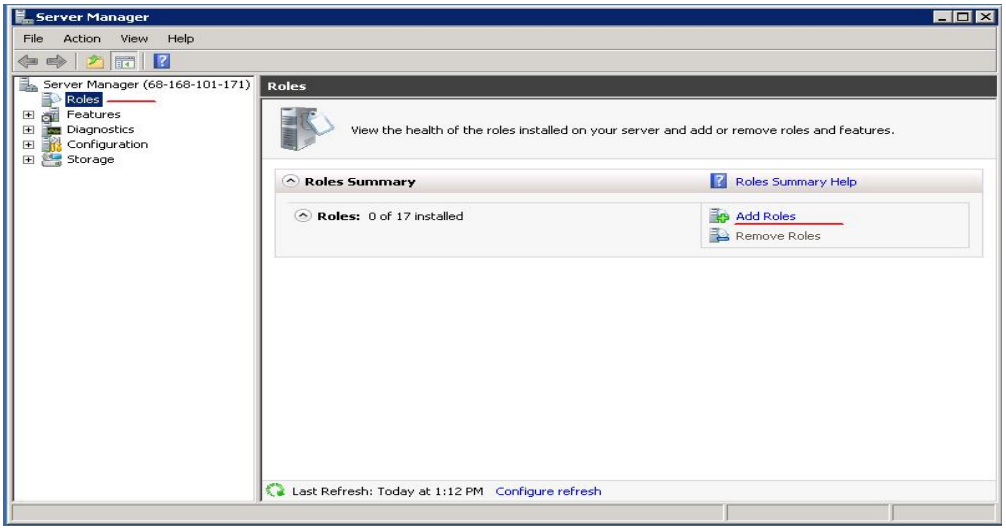
3. Follow the instructions to complete the installation process.

4. Click on Yes button when it asks for partitioning the drive.

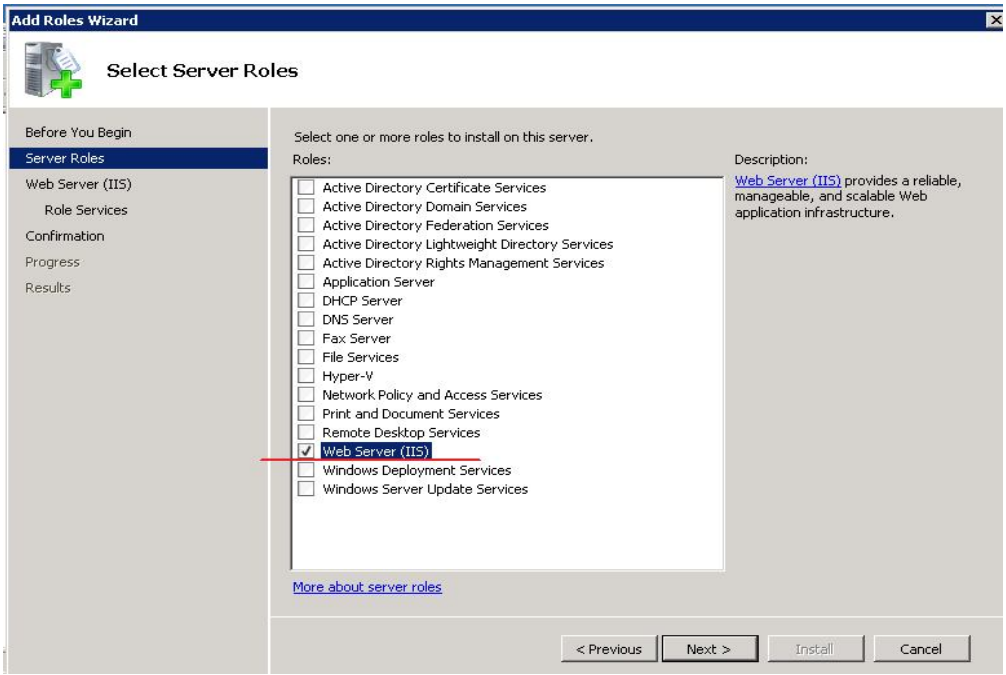
5. At last click on Next button and then Finish to complete the installation.

3.5 FTP in Windows Server 2008

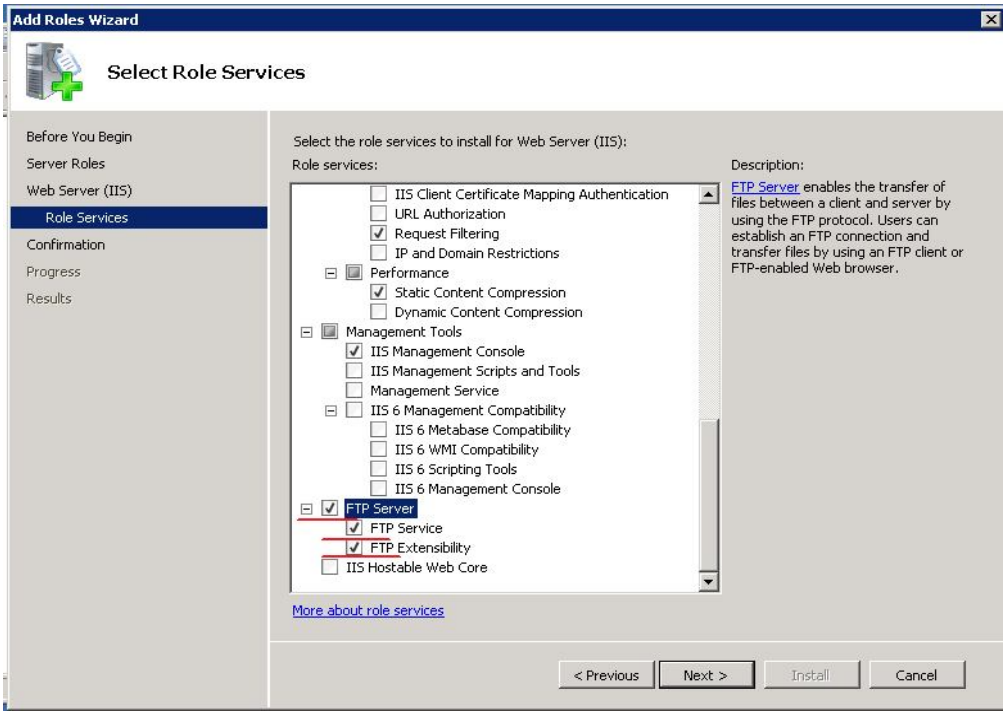
- Open Server Manager then click on Roles after that add some roles by clicking on "Add Roles" button.



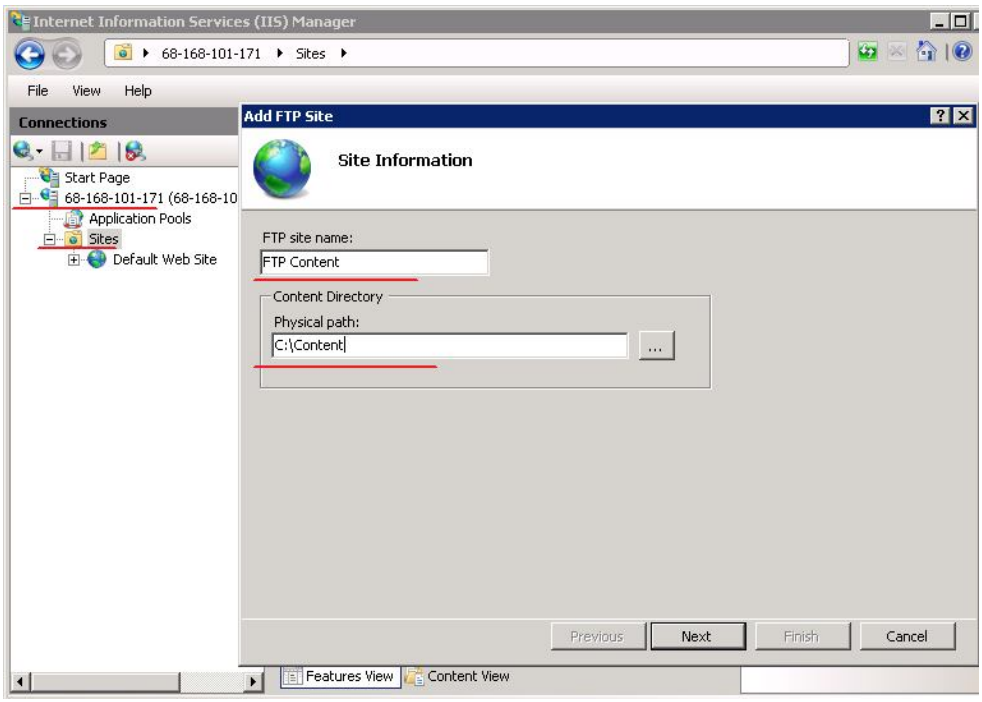
- We get another window and in this check the box Web Server (IIS) to install.



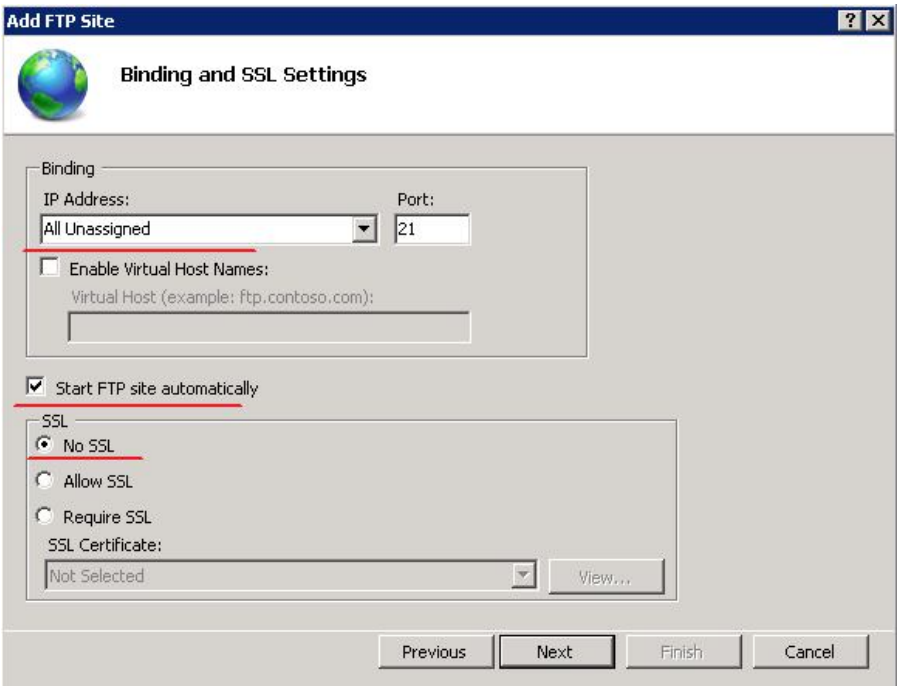
- Now click on the Next button. There are many services showing on this page, click on FTP Server and its sub parts then Click Next button to finish the role installation.



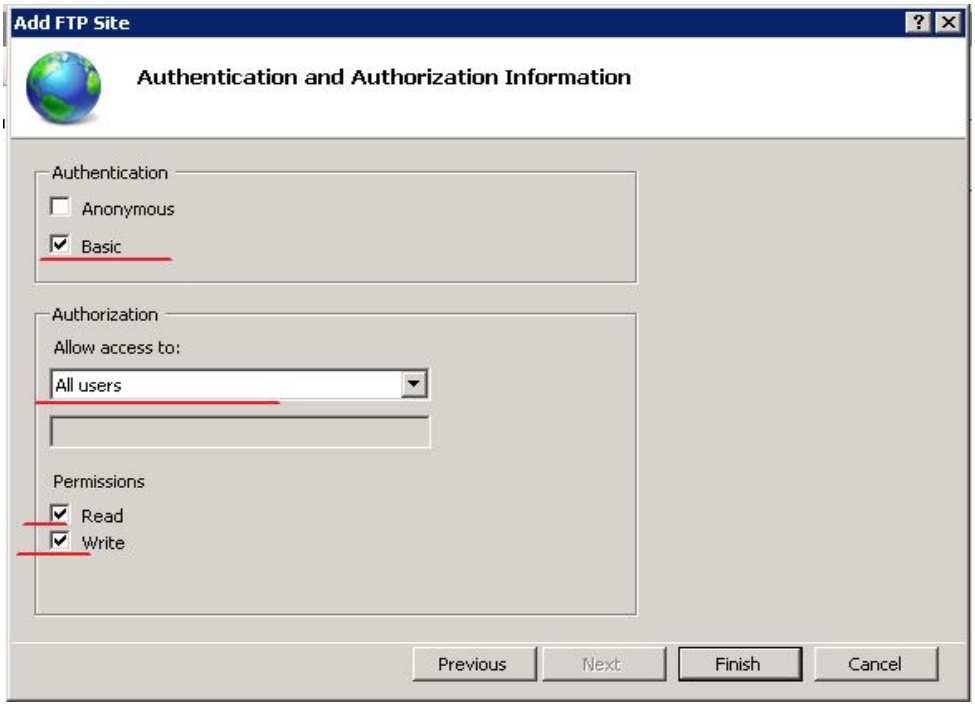
- Now open IIS Manager then enlarge the server, right click on the Sites. After then click on Add FTP Site provide it a site name and the path where we want to save it.



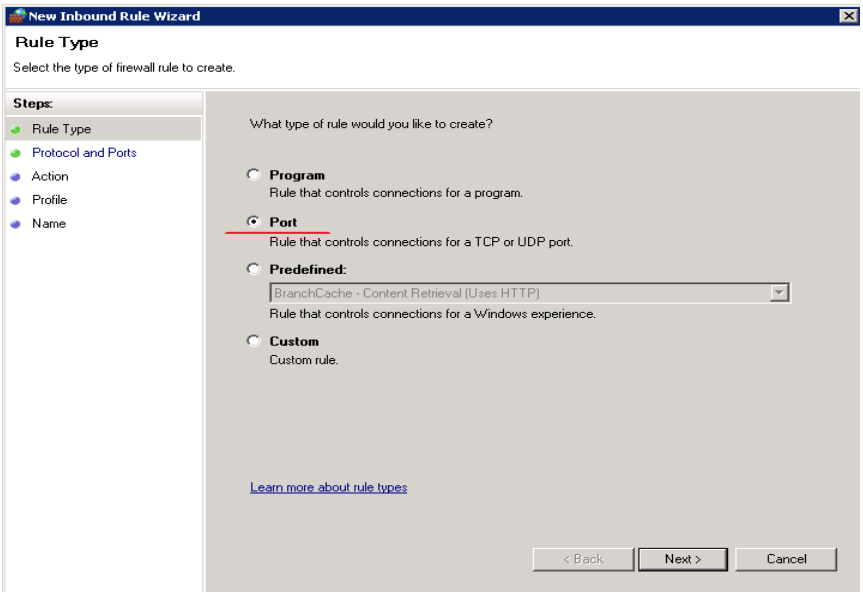
- Next step is to configure binding and Secure Socket Layer.



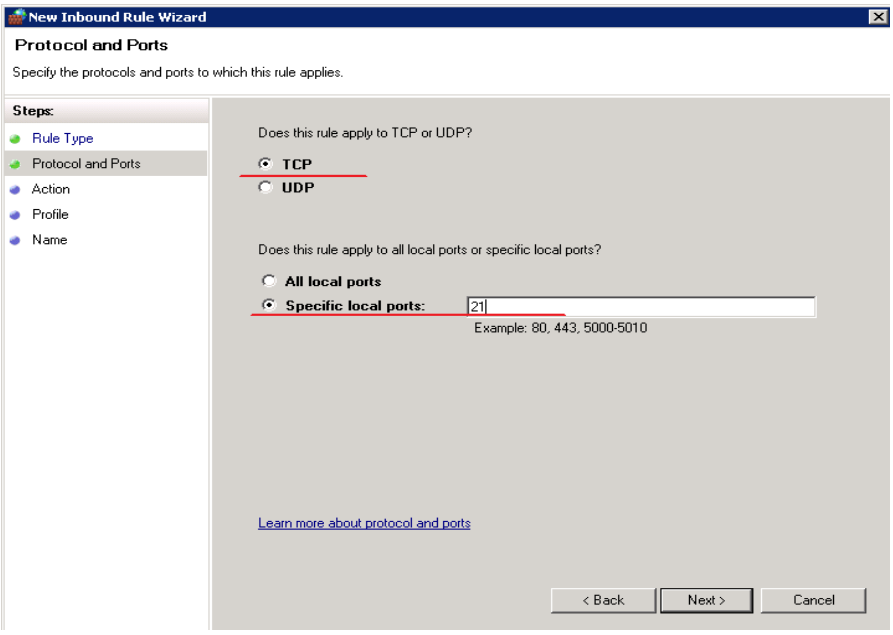
- Next is to configure the authorization for security aspects.



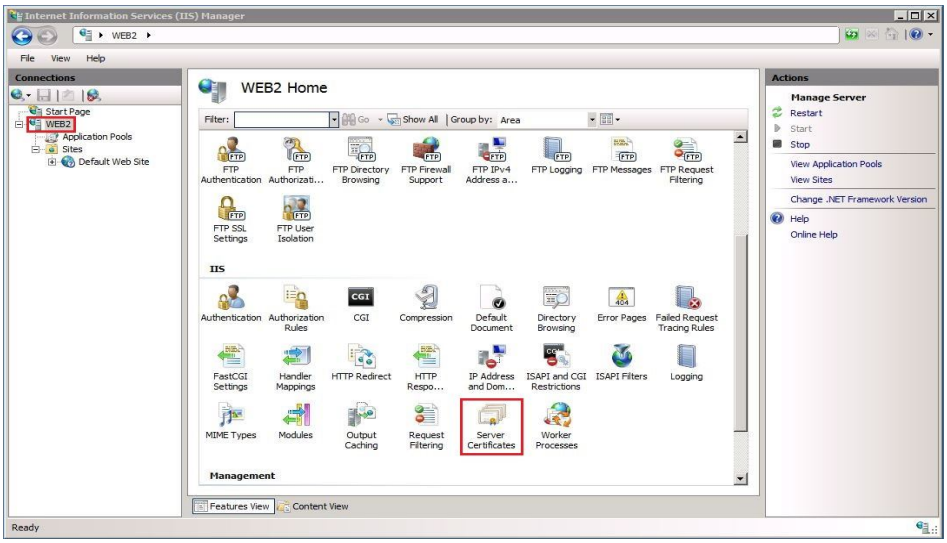
- Now click on finish button.
- For security aspects configure the Windows Firewall and add some Inbound Rules. For creating a rule clicks on New Rule and select Port.



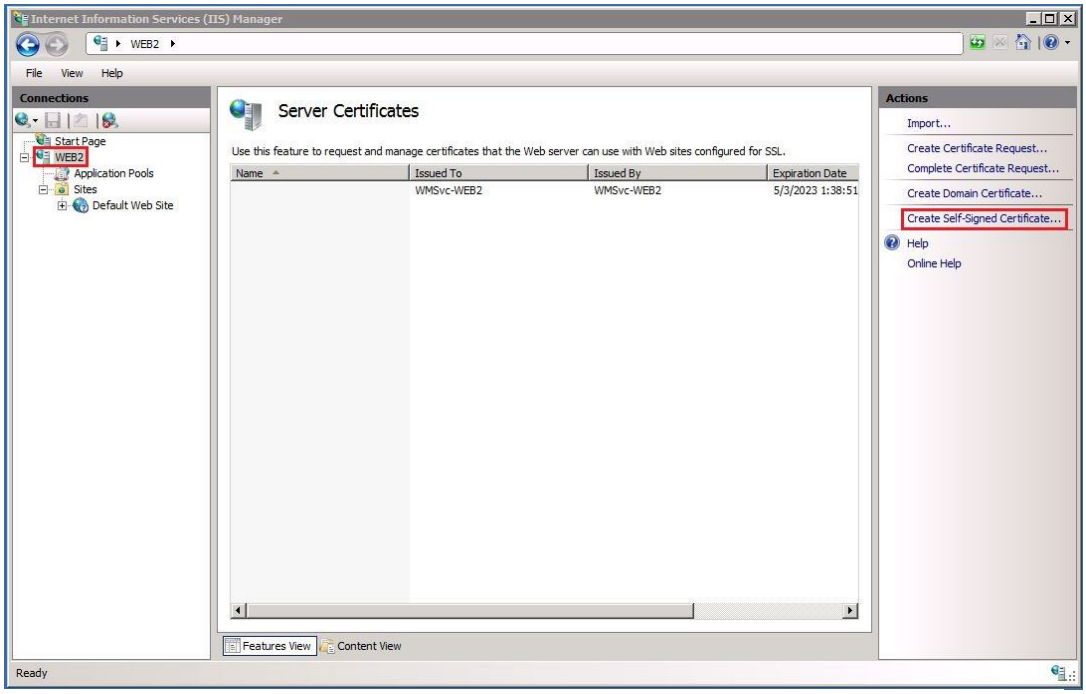
- The above rules are applied to TCP port 21 then click on Next button.



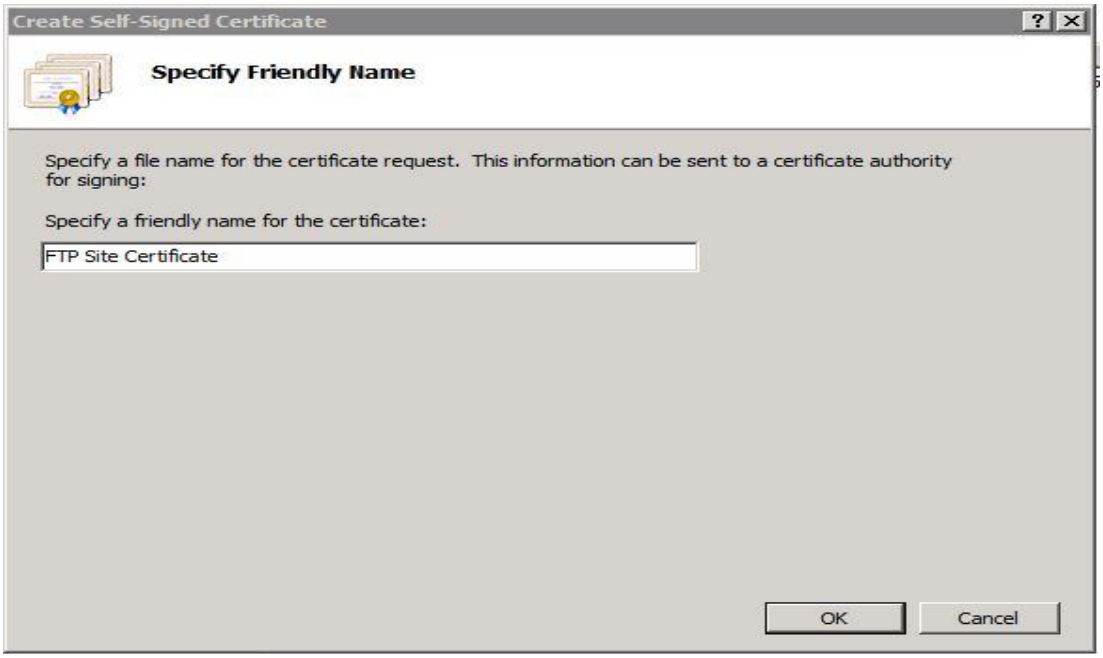
- Open **IIS Manager** and click on the server object. After this click on **Server Certificates**.



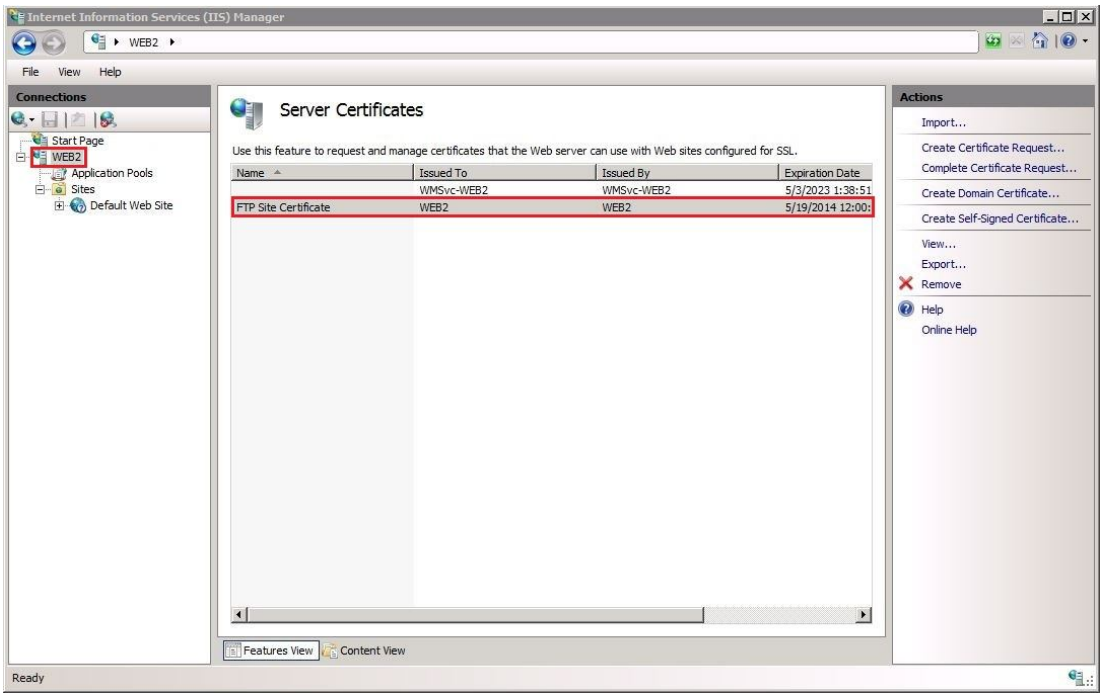
- Under Actions pane on right side click **Create Self-Signed Certificate.**



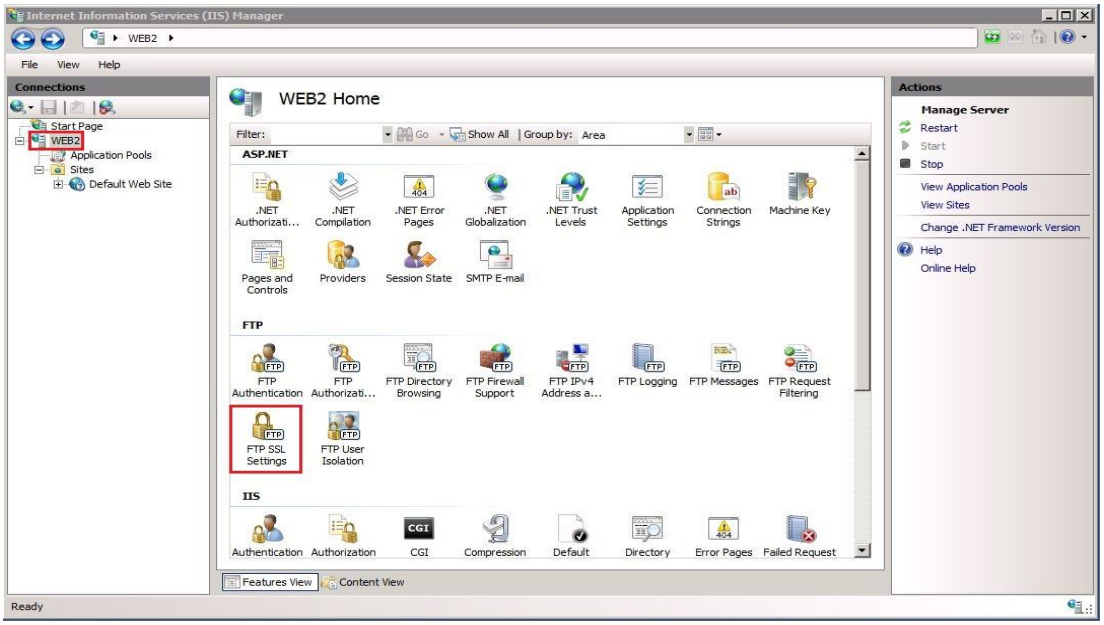
- Use any name to create the certificate. I used **FTPuser** as a name for my implementation and then click on **OK** button:



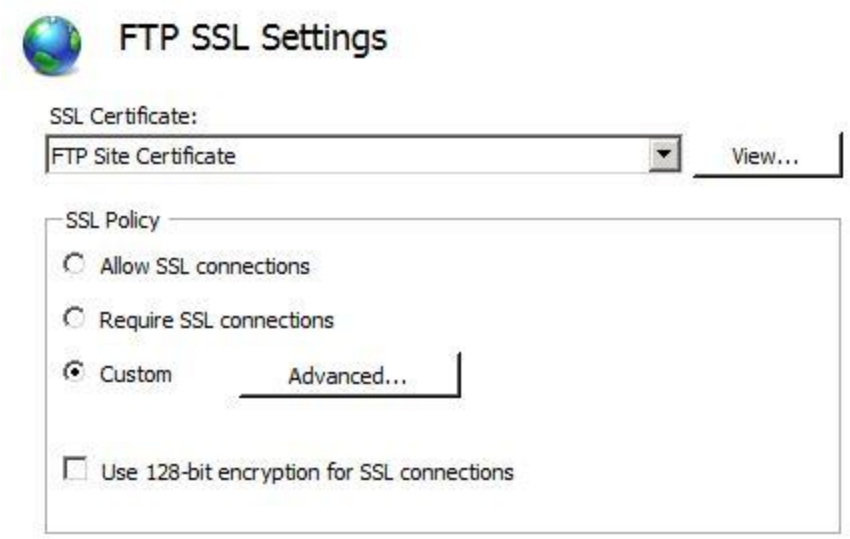
- We will see the certificate created in the list.



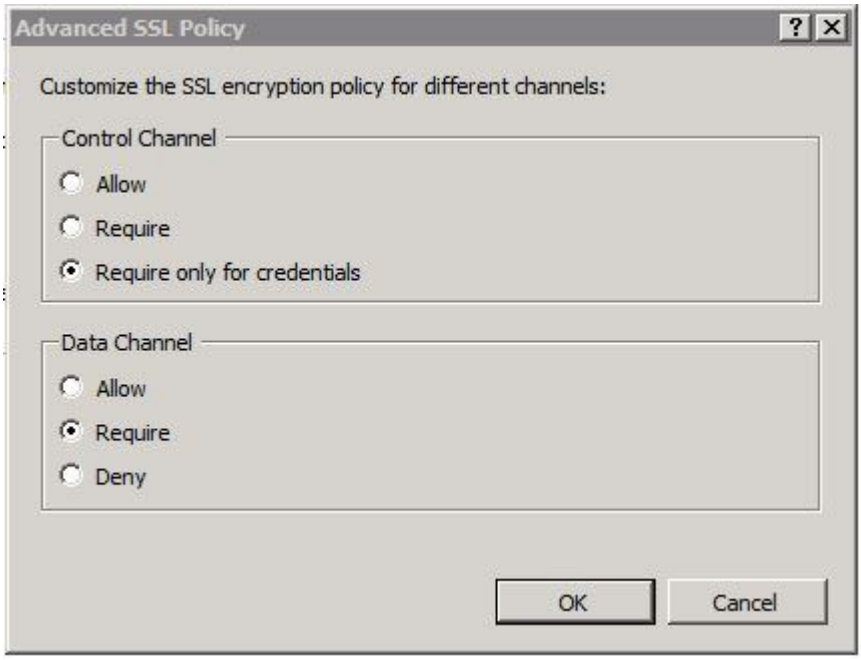
- Again open the **FTP SSL Settings**.



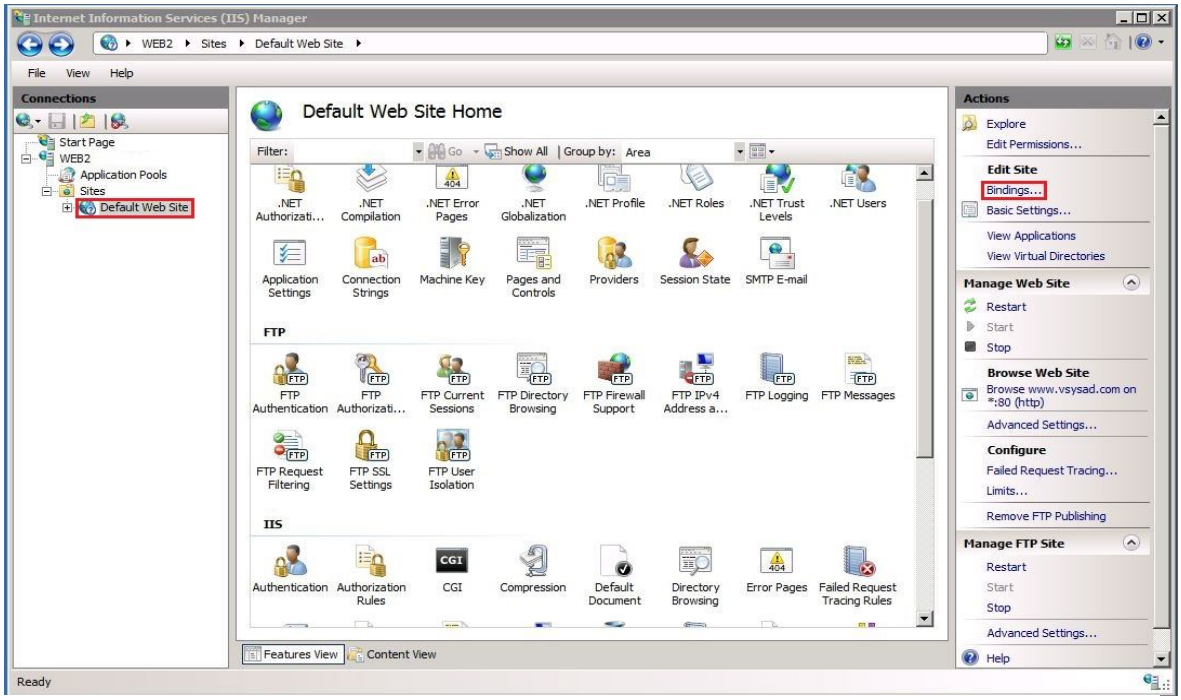
- Under **SSL Certificate** select the certificate we created earlier. Under **SSL Policy** select **Custom** and then click on the **Advanced** button:



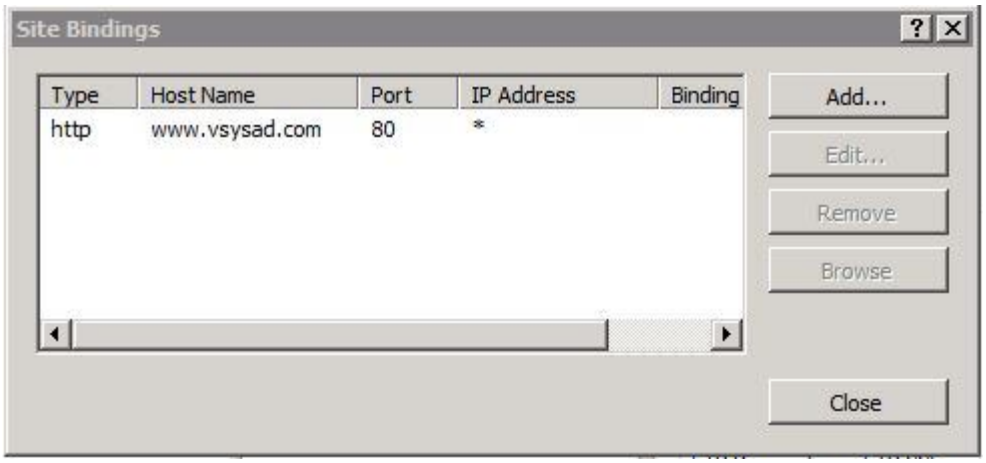
- Under **Control Channel** select **Require only for credentials** and under **Data Channel** select **Require** and then click on **OK**:



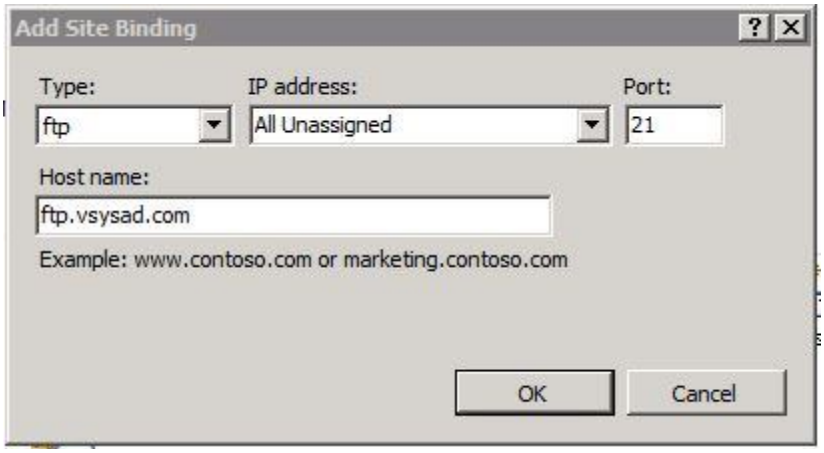
- **Click on the** Default Web Site **and then click on** Bindings **in the** Actions **pane:**



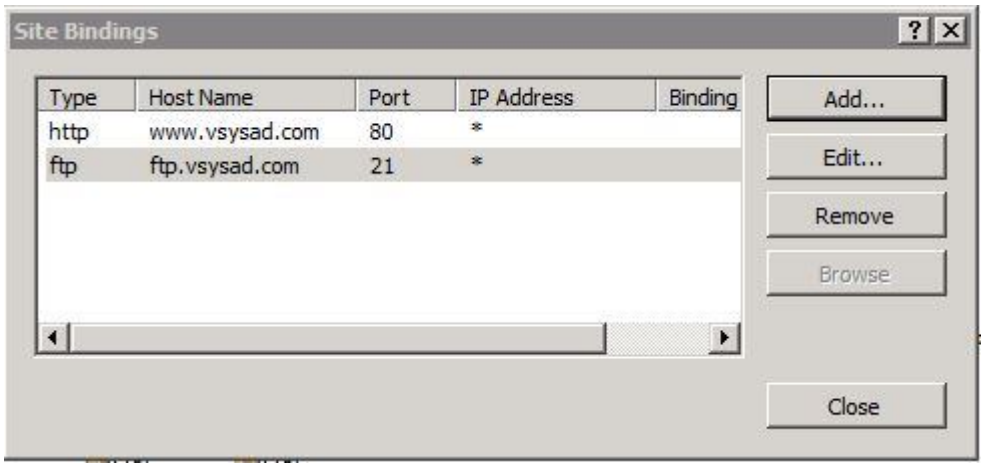
- **In the** Site Bindings **section click on the** Add **Button:**



- **In the** Add Site Binding **section select the** Type **as** ftp**, leave the** IP Address **box as** All Unassigned. **Enter the hostname and then click on** OK**:**



- **After above step confirm the added ftp site binding details and then click** Close**:**



- After then keep the default configuration to allow the connection and apply it to all profiles and finish the wizard.
- Now it's time to check the FTP server, it must be running.

3.6 FTP in Linux

1. Firstly, we install FTP Server and for this we install vsftpd (Very Secure FTP Daemon).
[root] # yum install -y vsftpd

2. Second step is to configure FTP Server. To configure it we have to edit the file '/etc/vsftpd/vsftpd.conf".

```
[root@itbox4vn ~]# grep -v ^# /etc/vsftpd/vsftpd.conf
anonymous_enable=YES
local_enable=YES
write_enable=YES
local_umask=022
dirmessage_enable=YES
xferlog_enable=YES
connect_from_port_20=YES
xferlog_std_format=YES
listen=YES

pam_service_name=vsftpd
userlist_enable=YES
tcp_wrappers=YES
```

3. Now we must configure the SELinux and the iptables

[root] # setsebool -P ftpd_disable_trans=1

[root] # iptables -I INPUT -m tcp -p tcp --dport 20 -j ACCEPT

[root] # iptables -I INPUT -m tcp -p tcp --dport 21 -j ACCEPT

4. Next step is to connect to the FTP Server

```
hanthuy@evil:~$ ftp 192.168.1.103
Connected to 192.168.1.103.
220 (vsFTPd 2.0.5)
Name (192.168.1.103:hanthuy): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> pwd
257 "/"
ftp> cd pub
250 Directory successfully changed.
ftp> ls
200 PORT command successful. Consider using PASV.
150 Here comes the directory listing.
-rw-r--r--    1 0        0               0 Aug 03 17:29 itbox4vn.com
-rw-r--r--    1 0        0               0 Aug 03 17:29 testftp
226 Directory send OK.
ftp> lcd /home/hanthuy/Documents/
Local directory now /home/hanthuy/Documents
ftp> mget testftp
mget testftp? y
200 PORT command successful. Consider using PASV.
150 Opening BINARY mode data connection for testftp (0 bytes).
226 File send OK.
ftp>
```

There are many applications through which we can connect to the ftp server likewise Gnome Commander (Linux).

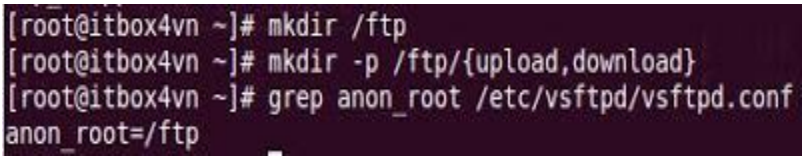
5. Last step is to secure our FTP Server

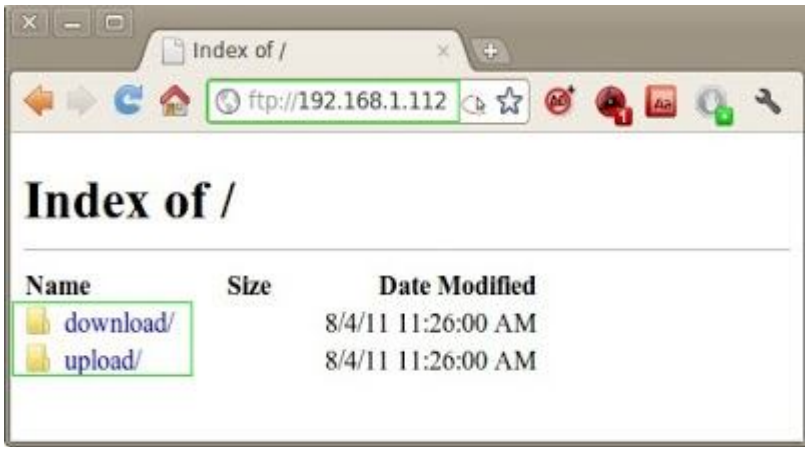
The FTP protocol supports two modes for file transfer one is active mode and another is passive mode. Active mode uses port 20 for connection with client. In passive mode, it uses a custom-defined range of ports above 1024. Vsftpd uses active mode by default.

Step 1. Disable the anonymous_enable option. It prevents the non-authorized users from accessing the FTP Server.
anonymous_enable=NO

Step 2. Restrict the users by using userlist_enable option from accessing the FTP Server.

Step 3. Must change the default folder





Step 4. Next step is to create Multi FTP Site

For example, I want to create one IP for local user and one IP address for anonymous user to log in.

```
[root@itbox4vn ~]# cd /etc/sysconfig/network-scripts/
[root@itbox4vn network-scripts]# cp ifcfg-eth0 ifcfg-eth0:0
[root@itbox4vn network-scripts]# grep IPADDR ifcfg-eth0
IPADDR=192.168.1.112
[root@itbox4vn network-scripts]# grep IPADDR ifcfg-eth0:0
IPADDR=192.168.1.113
```

Now one more config file for the new one.

```
[root@itbox4vn /]# cd /etc/vsftpd/
[root@itbox4vn vsftpd]# cp vsftpd.conf vsftpd1.conf
[root@itbox4vn vsftpd]# grep rule1 vsftpd.conf
anonymous_enable=YES  # rule1
listen_address=192.168.1.112  # rule1
[root@itbox4vn vsftpd]# grep rule2 vsftpd1.conf
anonymous_enable=NO # rule2
listen_address=192.168.1.113 # rule2
```

Step 5. Create SSL Certificate on Server

To create a self-signed certificate, we use a tool name openssl and then enter the following commands. If we don’t want to use a self signed certificate then use a certificate issued by 3rd party.

```
# cd /etc/vsftpd/

# /usr/bin/openssl req -x509 -nodes -days 365 -newkey rsa:1024 -keyout vsftpd.pem -out
vsftpd.pem
```

```
rsa_cert_file=/etc/vsftpd/vsftpd.pem
#rsa_private_key_file=/etc/vsftpd/vsftpd.key

ssl_enable=YES
allow_anon_ssl=YES
force_local_data_ssl=NO
force_local_logins_ssl=YES
ssl_tlsv1=YES
ssl_sslv2=NO
ssl_sslv3=NO
ssl_ciphers=HIGH
require_ssl_reuse=NO

pasv_max_port=65535
pasv_min_port=64000

## bandwidth allocation per anonymous session is set to roughly 30 KB/s ##
anon_max_rate=30000

## each local user is granted roughly 30 KB/s bandwidth ##
local_max_rate=30000

## client session is terminated after being idle for 300 seconds ##
idle_session_timeout=300

## maximum number of connections per source IP, which can help secure against DoS and DDoS attacks ##
max_per_ip=50
```

These Security configuration make the FTP server little bit more secure than the normal FTP server.

Chapter 4

PERFORMANCE MEASUREMENT

4.1 Windows Performance Measurement

In this we are measuring performance of the server having different aspects likewise RTT, Throughput, Latency, Jitter and bandwidth. After comparing all these aspects of both the servers (Linux and Windows), we are measuring the time to upload and download files of certain sizes.

4.1.1 Windows FTP Server

1. Round Trip Time

Table: 4.1 Round Trip Time of Window System

Min	Avg	Max
0.68	0.91	1.38
0.83	0.89	1.03
0.76	0.93	1.98
0.79	0.87	1.03
0.76	0.94	0.94

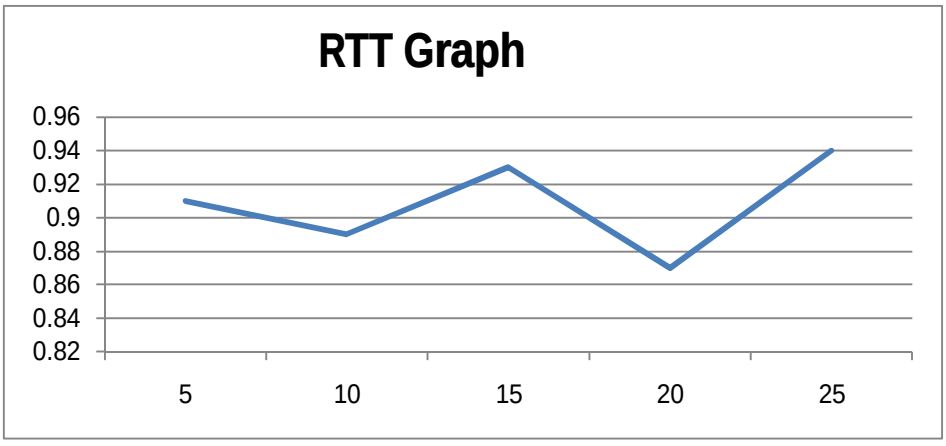


Fig 4.1: Round Trip Time Graph of Window System

2. Throughput

(MSS = 1500Byte, Loss = 1e-06%)

Table: 4.2 Throughput of Window System

RTT	Througput(Mbit/sec)
0.91	**125759.25**
0.89	**128585.30**
0.93	**123054.75**
0.87	**131541.29**
0.94	**121745.66**

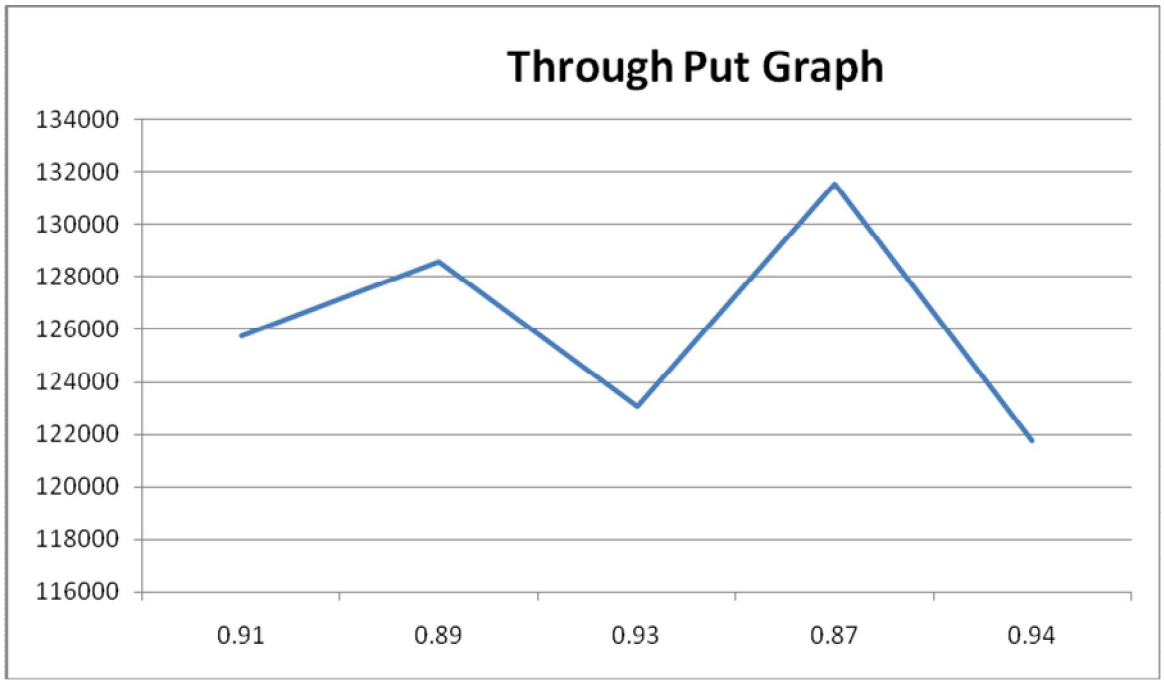


Fig: 4.2 Throughput Graph of Windows System

3. Latency

Table: 4.3 Latency of Window System

0.36
0.26
0.29
0.18
0.19

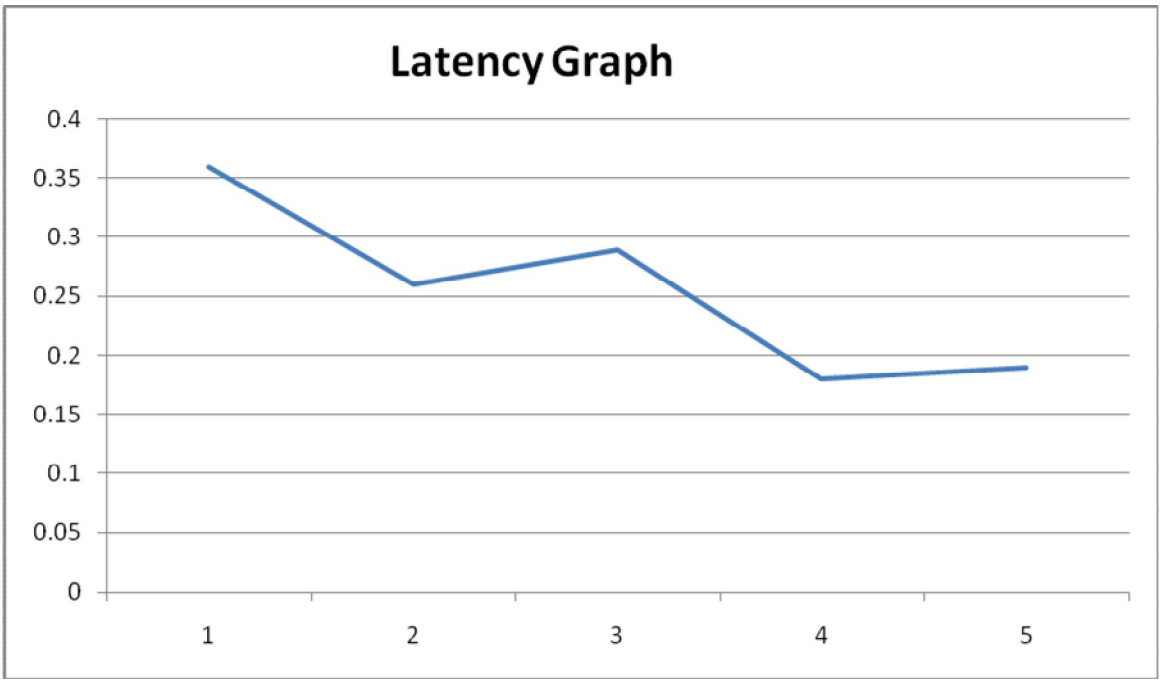


Fig: 4.3 Latency Graph of Window System

4. Jitter

Table 4.4 Jitter of Window system

0.29
0.14
0.16
0.35
0.16

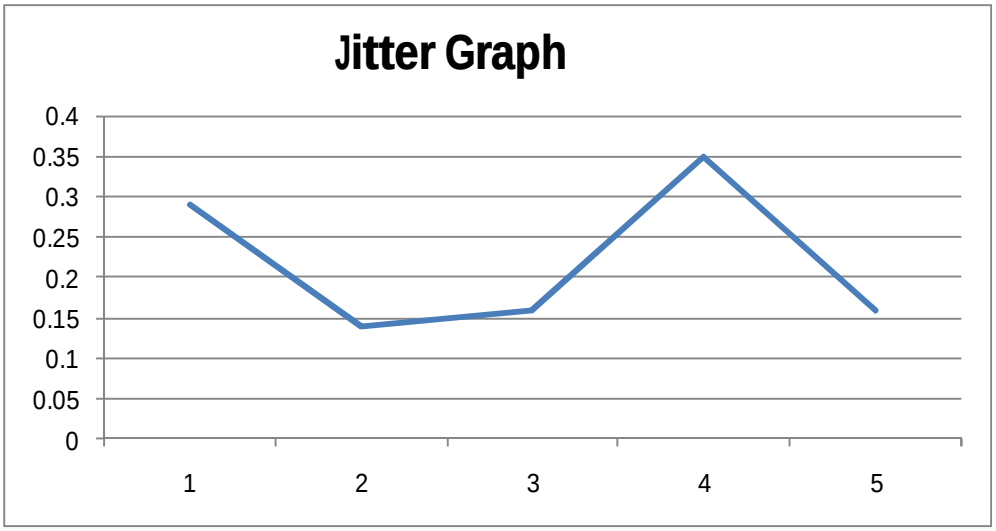


Fig: 4.4 Jitter Graph of Window System

5. Bandwidth

Table: 4.5 Bandwidth of Window System

Min	Avg	Max
0.90	0.92	0.97
0.79	1.00	1.23
0.77	1.02	1.90
0.87	0.87	1.59
0.74	1.08	1.02

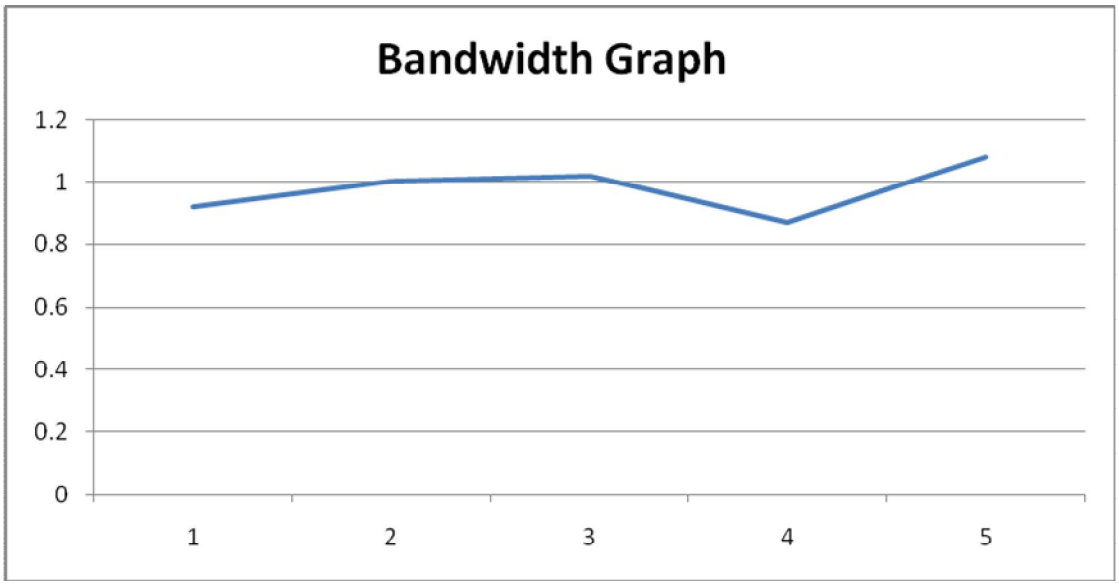


Fig: 4.5 Bandwidth Graph of Window System

4.1.2 Windows FTPS Server:

1. Round Trip Time

Table: 4.6 Round Trip Time Table of Window System with FTPS

Min	Avg	Max
0.86	1.07	2.10
0.87	1.15	1.91
0.79	0.88	1.13
0.84	1.17	3.19
0.60	1.15	3.93

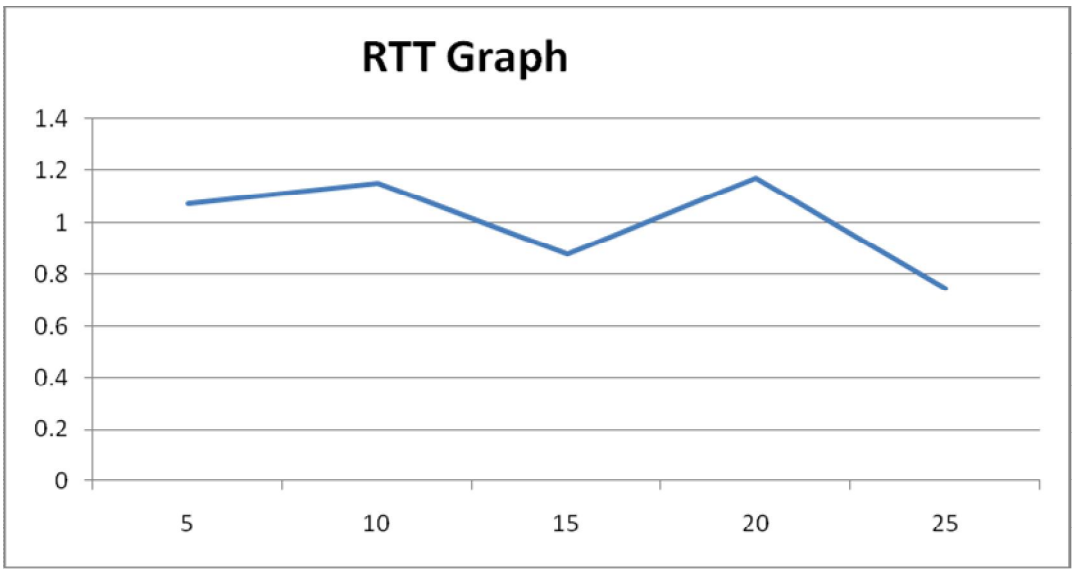


Fig: 4.6 Round Trip Time Graph of Window System with FTPS

2. Throughput
(MSS = 1500Byte, Loss = 1e-06%)

Table: 4.7 Throughput Table of Window System with FTPS

RTT	Througput(Mbit/sec)
1.07	**106954.13**
1.15	**99513.84**
0.88	**130046.50**
1.17	**97812.75**
1.15	**99513.84**

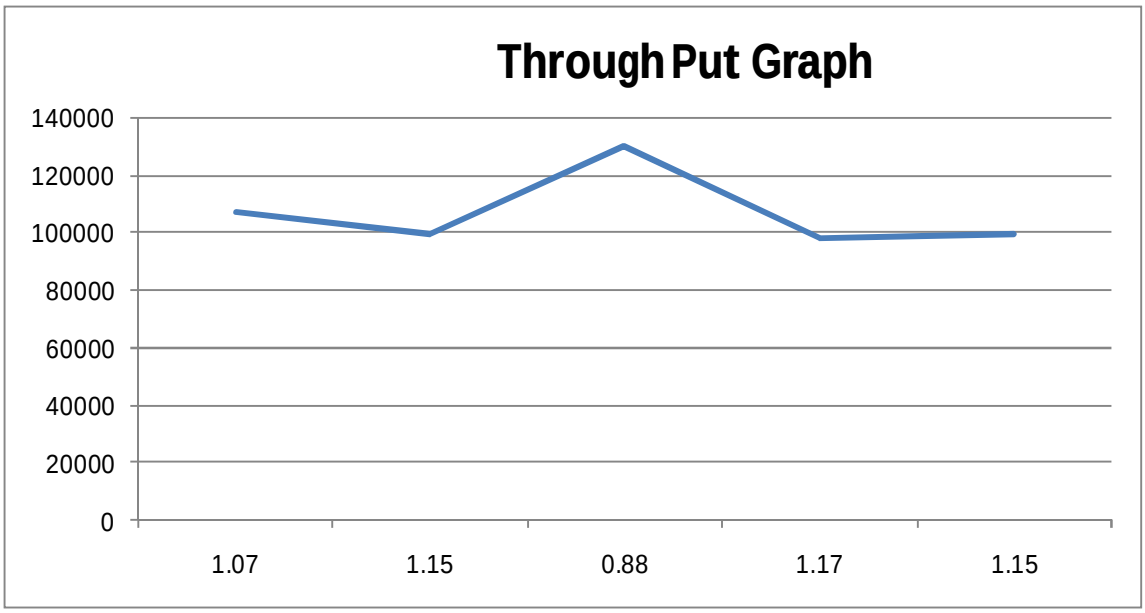


Fig: 4.7 Throughput Graph of Window System with FTPS

3. Latency

Table: 4.8 Latency Table of Window System with FTPS

0.44
0.29
0.30
0.25
0.22

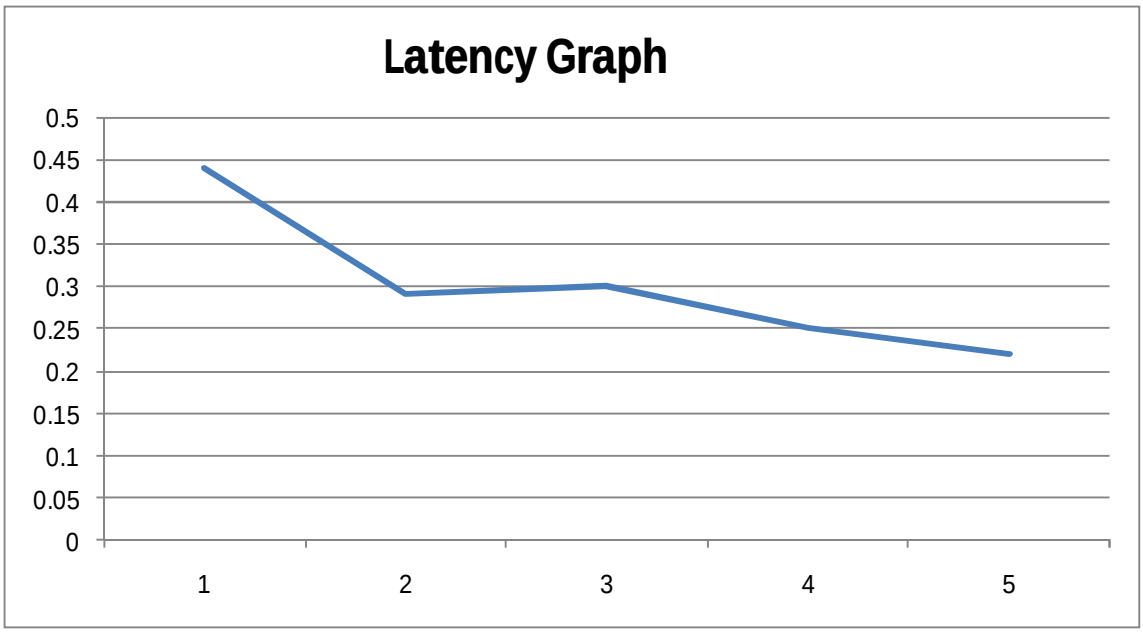


Fig: 4.8 Latency Graph of Window System with FTPS

4. Jitter

Table:4.9 Jitter of Window System with FTPS

0.39
0.22
0.20
0.23
0.18

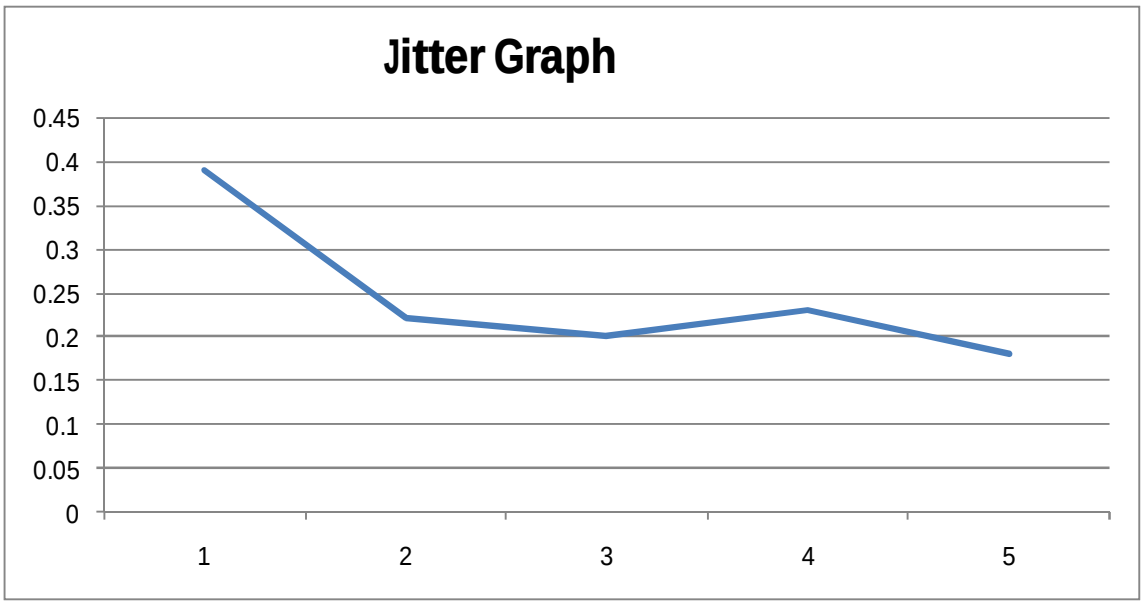


Fig: 4.9 Jitter Graph of Window System with FTPS

5. Bandwidth

Table: 4.10 Bandwidth table of Window System with FTPS

Min	Avg	Max
0.69	0.87	1.21
0.76	0.83	0.83
0.78	0.87	0.87
0.75	0.81	0.89
0.69	0.80	0.81

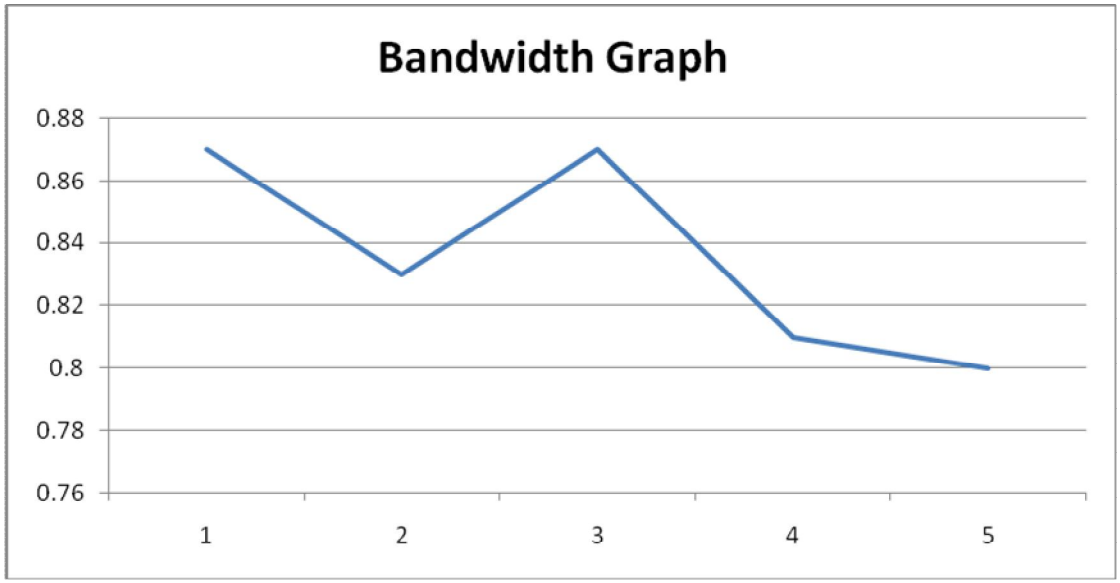


Fig: 4.10 Jitter Graph of Window System with FTPS

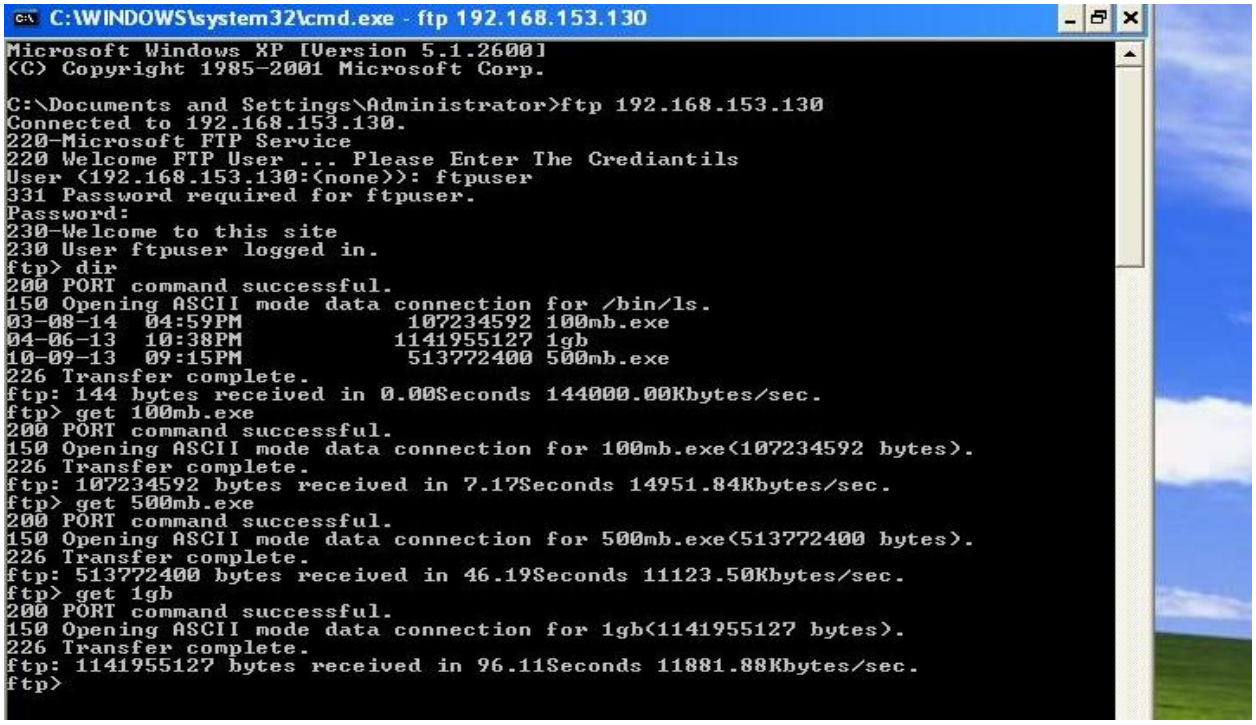
Downloading files from FTP server:

In this section we will discuss our experiments used to determine the transfer rate as a function of file sizes. We transferred files of 100MB, 500MB and 1GB. The results are given in the following table.

From Windows system:

Table 4.11: Downloading files from FTP Server

File Size	**100 MB**	**500 MB**	**1 GB**
Time (sec)	**7.17 (14951 Kbytes/sec)**	**46.19 (11123.50 Kbytes/sec)**	**96.11 (11881.88 Kbytes/sec)**

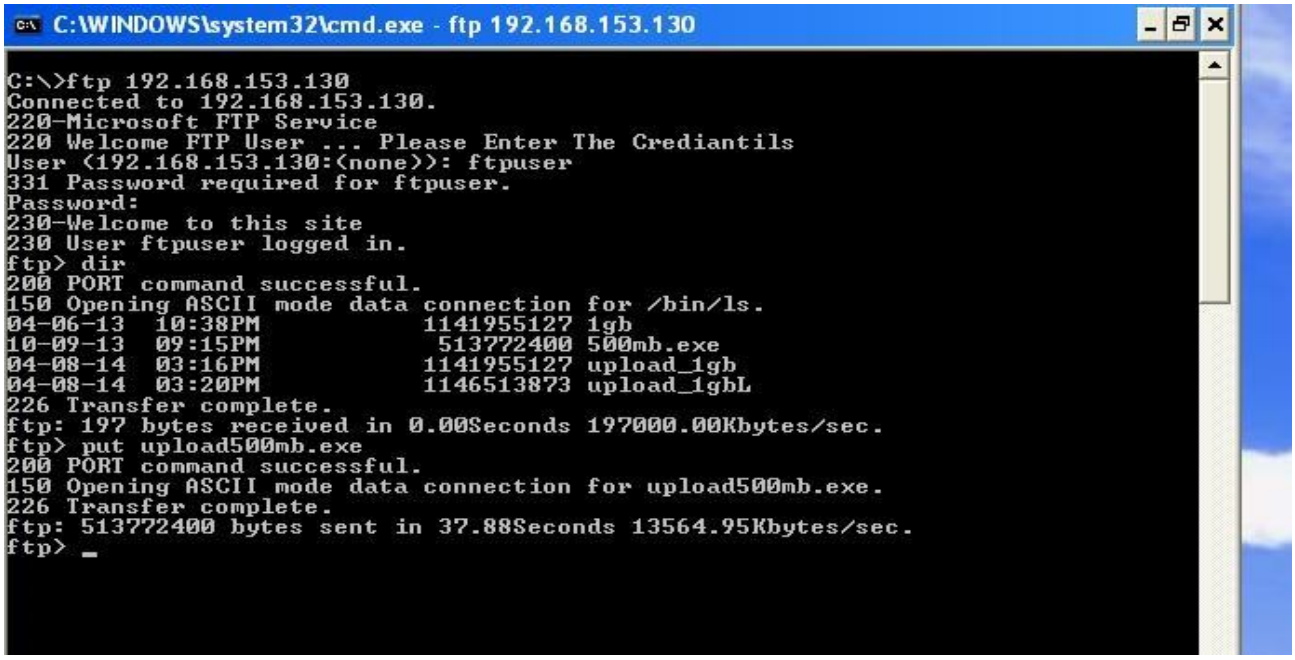


Uploading Files to Windows FTP server (from Windows System):

Table: 4.12 uploading file to Windows FTP Server (from windows system)

File Size	100 MB	500 MB	1GB
Time (sec)	4.02 (26474.21 Kbytes/sec)	37.88 (13564 Kbytes/sec)	112.72 (10131.08 Kbytes/sec)


C:\WINDOWS\system32\cmd.exe - ftp 192.168.153.141
C:\>ftp 192.168.153.141
Connected to 192.168.153.141.
220-Microsoft FTP Service
220 Welcome FTP User ... Please Enter The Crediantils
User (192.168.153.141:(none)): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230-Welcome to this site
230 Anonymous user logged in.
ftp> put upload_100mb
200 PORT command successful.
150 Opening ASCII mode data connection for upload_100mb.
226 Transfer complete.
ftp: 106320416 bytes sent in 4.02Seconds 26474.21Kbytes/sec.
ftp>


C:\WINDOWS\system32\cmd.exe - ftp 192.168.153.130
C:\>ftp 192.168.153.130
Connected to 192.168.153.130.
220-Microsoft FTP Service
220 Welcome FTP User ... Please Enter The Crediantils
User (192.168.153.130:(none)): ftpuser
331 Password required for ftpuser.
Password:
230-Welcome to this site
230 User ftpuser logged in.
ftp> dir
200 PORT command successful.
150 Opening ASCII mode data connection for /bin/ls.
04-06-13 10:38PM 1141955127 1gb
10-09-13 09:15PM 513772400 500mb.exe
04-08-14 03:16PM 1141955127 upload_1gb
04-08-14 03:20PM 1146513873 upload_1gbL
226 Transfer complete.
ftp: 197 bytes received in 0.00Seconds 197000.00Kbytes/sec.
ftp> put upload500mb.exe
200 PORT command successful.
150 Opening ASCII mode data connection for upload500mb.exe.
226 Transfer complete.
ftp: 513772400 bytes sent in 37.88Seconds 13564.95Kbytes/sec.
ftp> _

Downloading Files using Linux system:

Table: 4.13 Downloading Files using Linux system

File Size	100 MB	500 MB	1 GB
Time (sec)	15.1 (7088.02 Kbytes/sec)	77.50 (6626.80 Kbytes/sec)	180 (6331.74 Kbytes/sec)

```
ftp> get 100mb.exe
local: 100mb.exe remote: 100mb.exe
227 Entering Passive Mode (192,168,153,130,192,50).
125 Data connection already open; Transfer starting.
WARNING! 383987 bare linefeeds received in ASCII mode
File may not have transferred correctly.
226 Transfer complete.
107234592 bytes received in 15.1 secs (7088.02 Kbytes/sec)
ftp>
```

```
ftp> get 500mb.exe
local: 500mb.exe remote: 500mb.exe
227 Entering Passive Mode (192,168,153,130,192,54).
125 Data connection already open; Transfer starting.
WARNING! 1960021 bare linefeeds received in ASCII mode
File may not have transferred correctly.
226 Transfer complete.
513772400 bytes received in 77.5 secs (6626.80 Kbytes/sec)
ftp>
```

```
ftp> get 1gb
local: 1gb remote: 1gb
227 Entering Passive Mode (192,168,153,130,192,56).
125 Data connection already open; Transfer starting.
WARNING! 4584535 bare linefeeds received in ASCII mode
File may not have transferred correctly.
226 Transfer complete.
1141955127 bytes received in 180 secs (6331.74 Kbytes/sec)
ftp>
```

Uploading Files to Windows FTP server (from Linux System):

Table: 4.14 Uploading Files to Windows FTP server (from Linux System)

File Size	100 MB	500 MB	1GB
Time (sec)	9.71 (10995.08 Kbytes/sec)	29.9 (17228.81 Kbytes/sec)	180 (6331.74 Kbytes/sec)

```
230-Welcome to this site
230 Anonymous user logged in.
Remote system type is Windows_NT.
ftp> put upload_100mb
local: upload_100mb remote: upload_100mb
227 Entering Passive Mode (192,168,153,141,192,81).
125 Data connection already open; Transfer starting.
226 Transfer complete.
106746115 bytes sent in 9.71 secs (10995.08 Kbytes/sec)
ftp>
```

```
ftp> put uploadlin500mb.exe
local: uploadlin500mb.exe remote: uploadlin500mb.exe
227 Entering Passive Mode (192,168,153,130,192,70).
125 Data connection already open; Transfer starting.
226 Transfer complete.
515699006 bytes sent in 29.9 secs (17228.81 Kbytes/sec)
ftp>
```

```
ftp> put upload_1gbL
local: upload_1gbL remote: upload_1gbL
227 Entering Passive Mode (192,168,153,130,192,66).
125 Data connection already open; Transfer starting.
226 Transfer complete.
1146513873 bytes sent in 123 secs (9297.63 Kbytes/sec)
ftp>
```

4.2 Linux Performance Measurements

4.2.1 Linux FTP Server

1. **Round Trip Time**

Table: 4.15 Round trip Time Table of Linux System

Min	Avg	Max
0.62	0.75	1.01
0.59	0.72	1.03
0.64	0.73	1.20
0.57	0.70	0.81
0.62	0.68	0.85

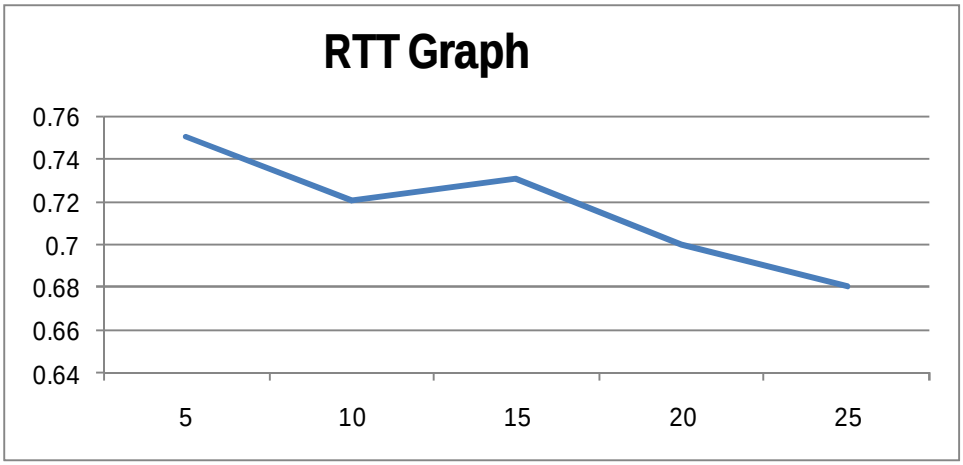


Fig: 4.11 Round trip Time Graph of Linux System

2. Throughput

(MSS = 1500Byte, Loss = 1e-06%)

Table: 4.16 Throughput Table of Linux System

RTT	Throughput (Mbit/sec)
0.75	**152587.89**
0.72	**158945.72**
0.73	**156768.38**
0.70	**163487.03**
0.68	**168295.47**

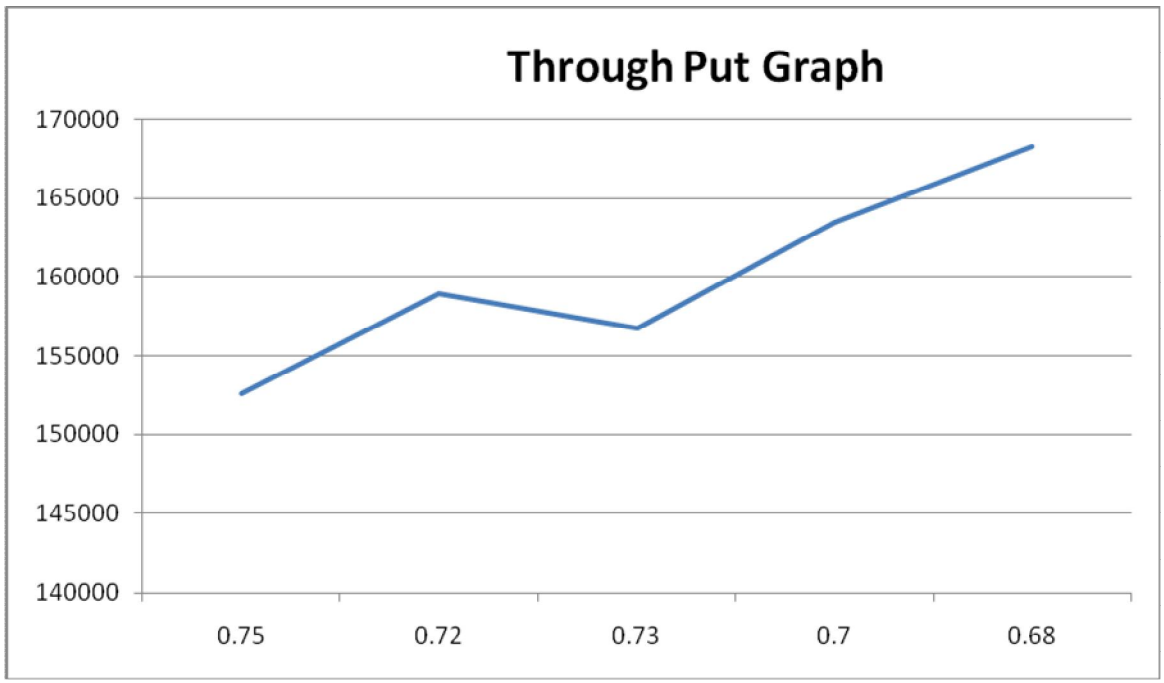


Fig: 4.12 Throughput Graph of Linux System

4. Latency

Table: 4.17 Latency Table of Linux System

0.08
0.06
0.02
0.09
0.03

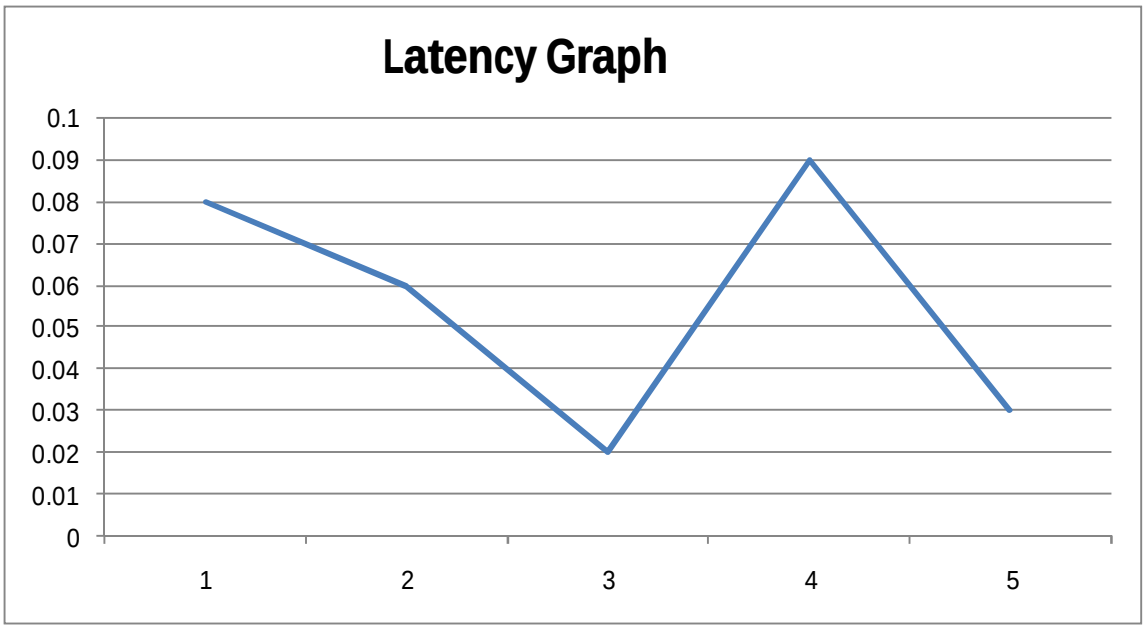


Fig: 4.13 Latency Graph of Linux System

5. Jitter

Table: 4.18 Jitter Table of Linux System

0.19
0.02
0.10
0.03
0.08

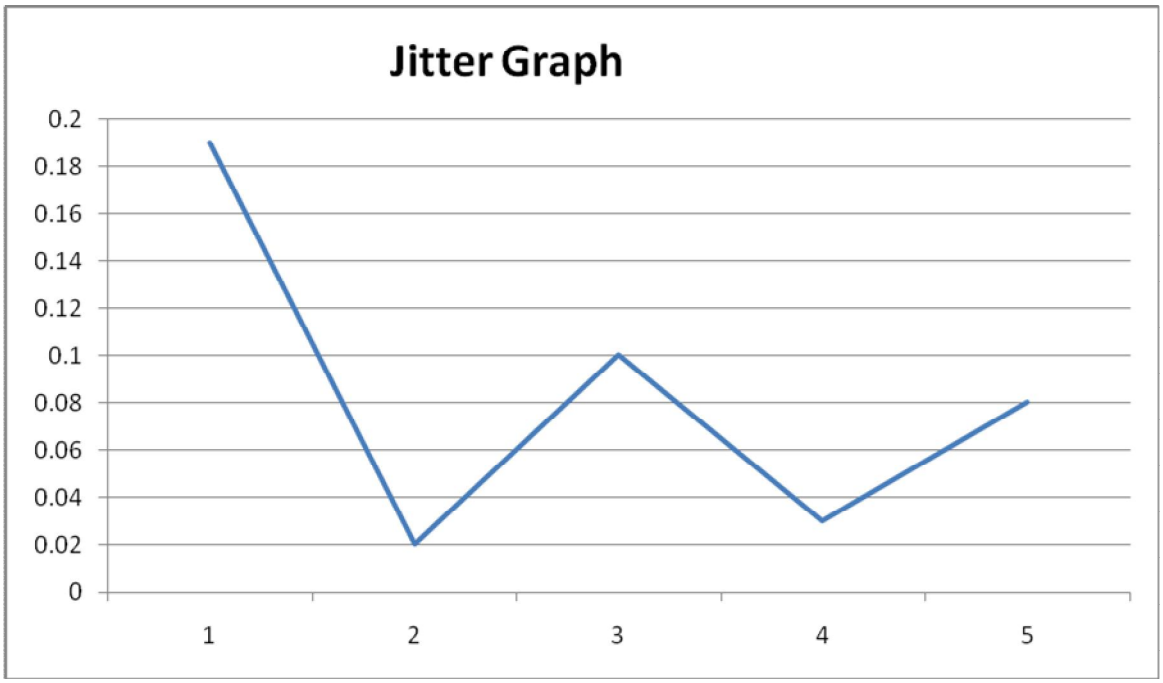


Fig: 4.14 Jitter Graph of Linux System

6. Bandwidth

Table: 4.19 Bandwidth Table of Linux System

Min	Avg	Max
0.62	0.72	0.95
0.63	0.69	0.85
0.64	0.67	0.69
0.55	0.63	0.71
0.61	0.69	0.73

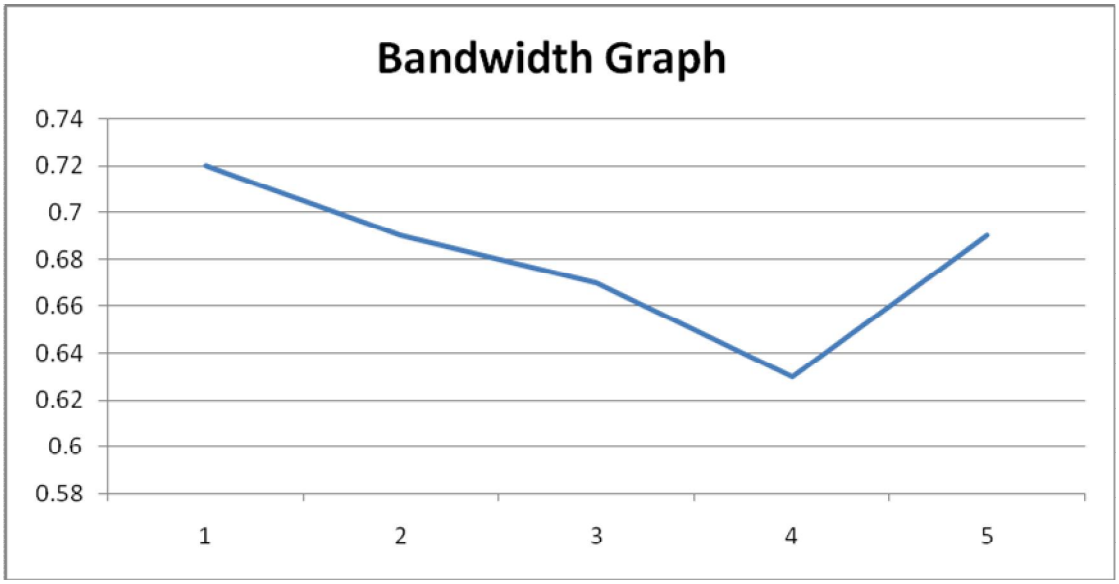


Fig: 4.15 Bandwidth Graph of Linux System

4.2.2 Linux FTPS Server

1. Round Trip Time

Table: 4.20 Round Trip Time Table of Linux System with FTPS

Min	Avg	Max
0.65	0.92	2.56
0.47	0.85	1.45
0.68	0.78	0.90
0.72	0.80	0.98
0.65	0.82	1.01

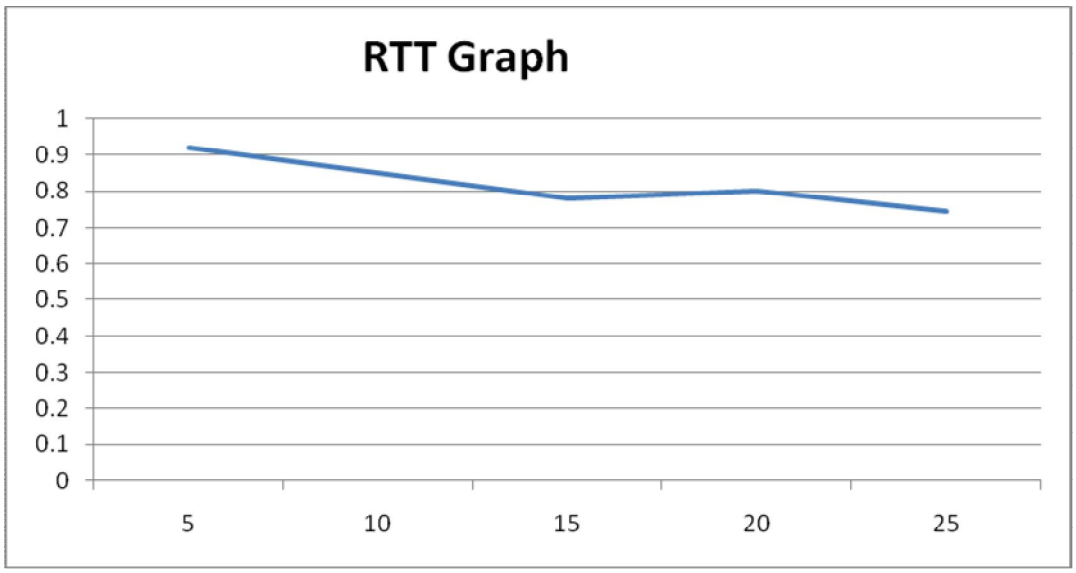


Fig: 4.16 Round Trip Time Graph of Linux System with FTPS

2. Throughput

(MSS = 1500Byte, Loss = 1e-06%)

Table: 4.21 Throughput Table of Linux System with FTPS

RTT	Througput(Mbit/sec)
0.92	**124392.30**
0.85	**134636.37**
0.78	**146719.13**
0.80	**143051.15**
0.82	**139562.10**

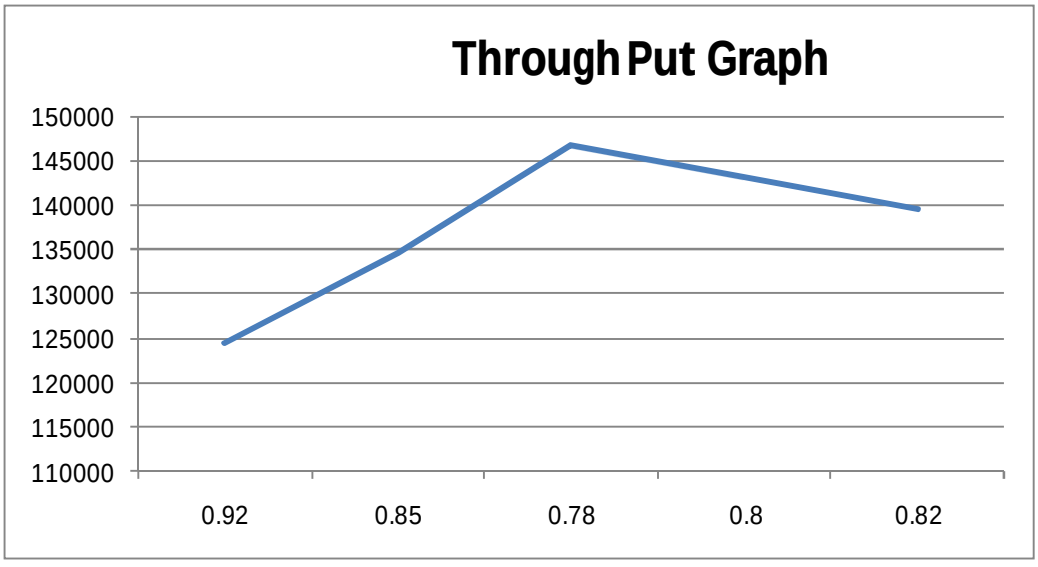


Fig: 4.17 Throughput Graph of Linux System with FTPS

4. Latency

Table: 4.22 Latency Table of Linux System with FTPS

0.17
0.09
0.11
0.08
0.13

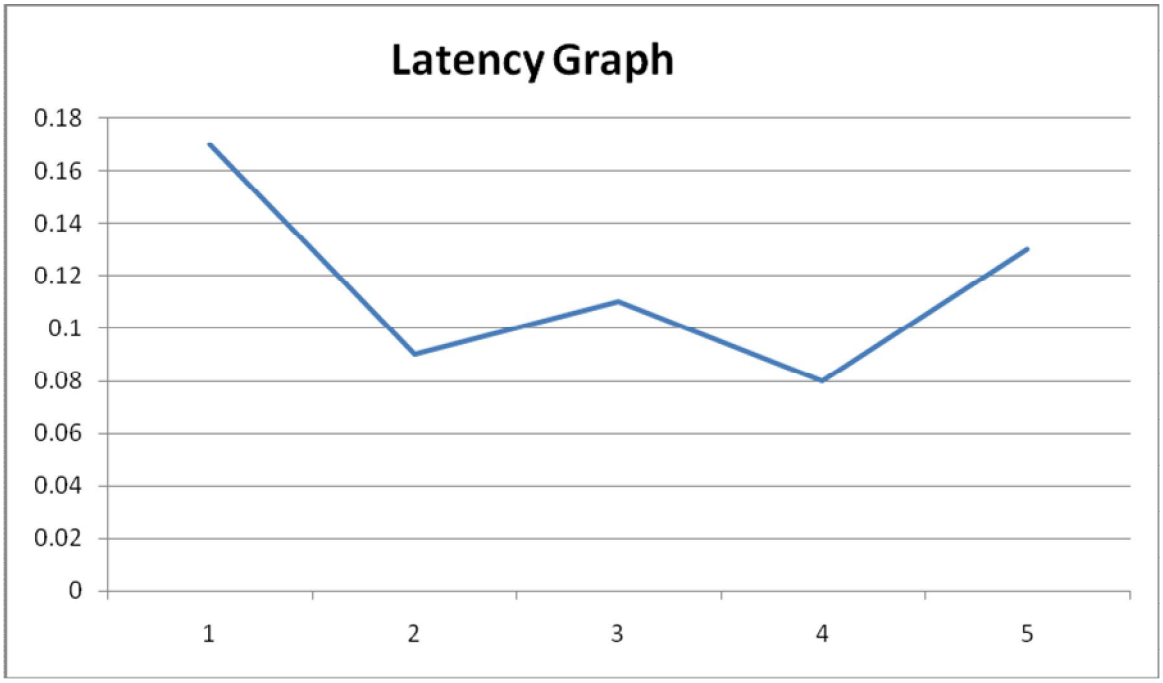


Fig: 4.18 Latency Graph of Linux System with FTPS

5. Jitter

Table: 4.23 Jitter Table of Linux System with FTPS

0.33
0.13
0.09
0.17
0.11

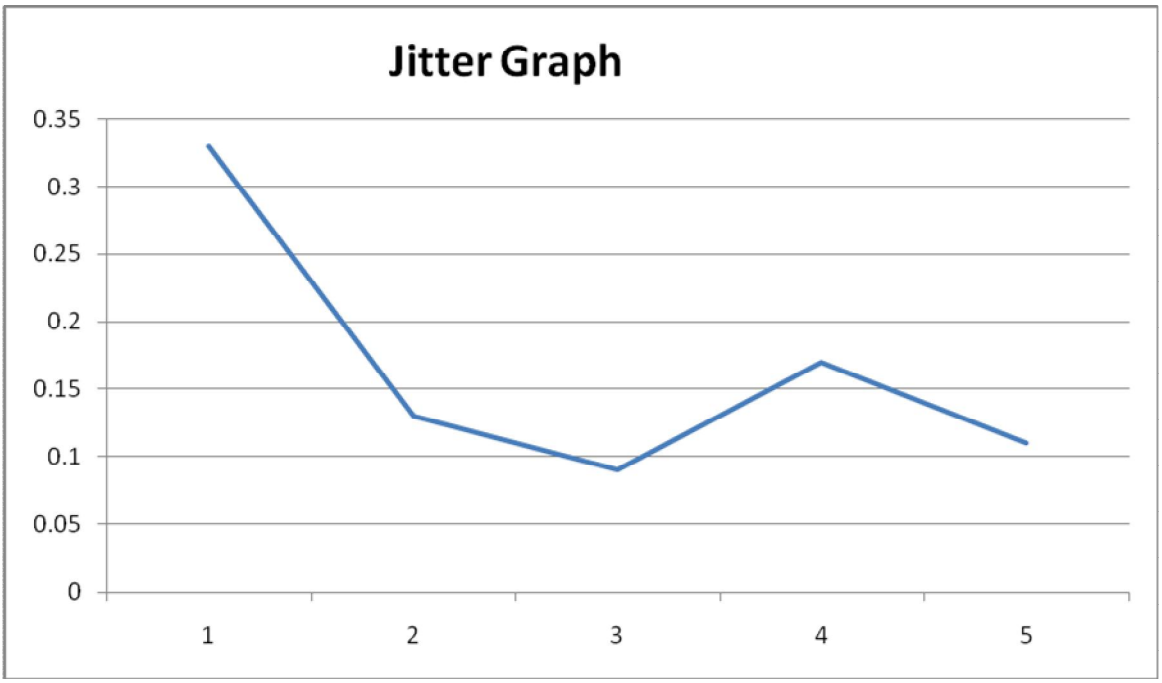


Fig: 4.19 Jitter Graph of Linux System with FTPS

6. Bandwidth

Table: 4.24 Bandwidth Table of Linux System with FTPS

Min	Avg	Max
0.64	0.77	0.92
0.58	0.75	1.01
0.56	0.72	1.05
0.71	0.83	1.01
0.69	0.73	0.79

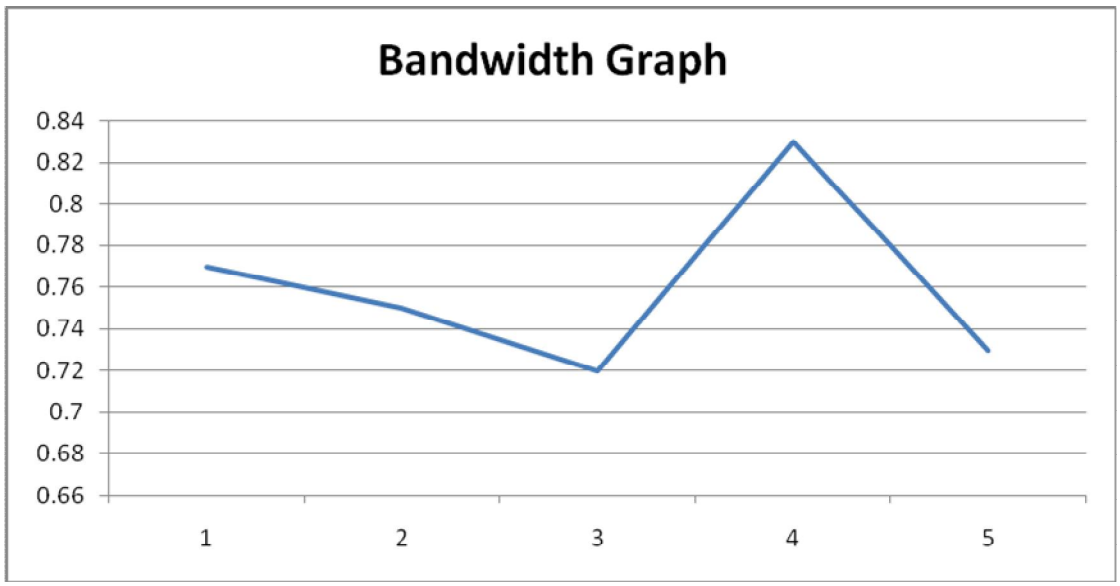


Fig: 4.20 Bandwidth Graph of Linux System with FTPS

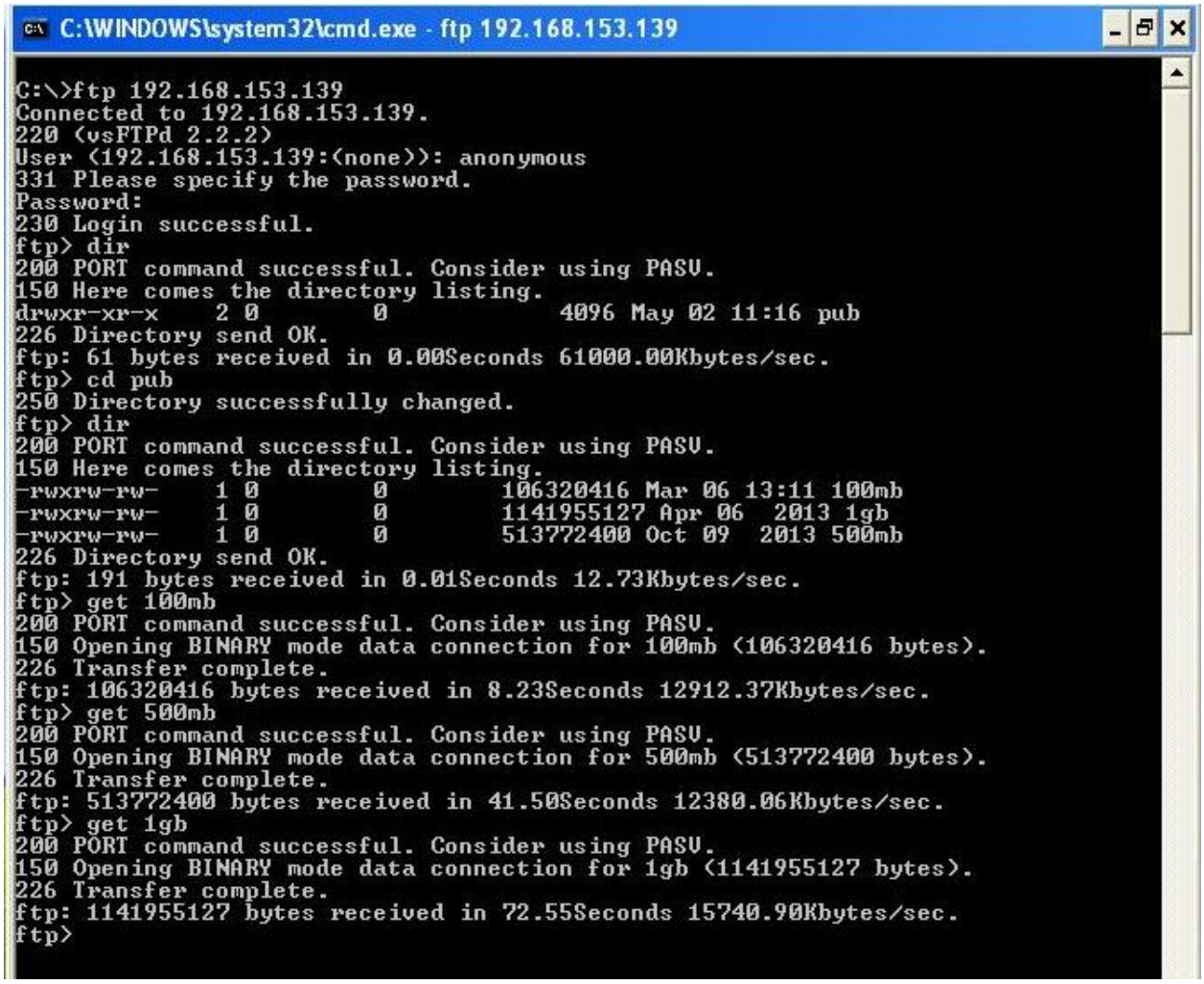
Downloading files from FTP server:

In this section we will discuss our experiments used to determine the transfer rate as a function of file sizes. We transferred files of 100MB, 500MB and 1GB. The results are given in the following table.

From Windows system:

Table: 4.25 Downloading files from Window

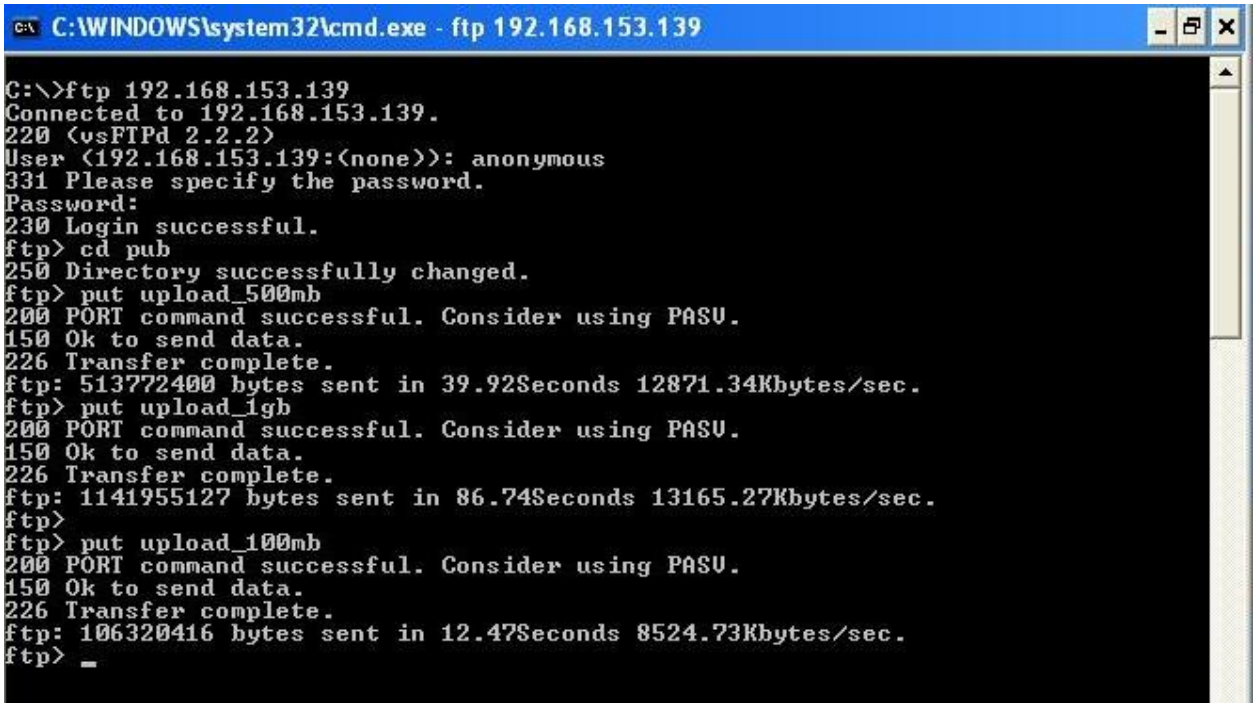
File Size	100 MB	500 MB	1 GB
Time (sec)	8.236 (12912 Kbytes/sec)	41.50 (12380.06 Kbytes/sec)	72.55 (15740.90 Kbytes/sec)



Uploading Files to Linux FTP server (from Windows System):

Table: 4.26 Uploading Files to Linux FTP server (from Windows System)

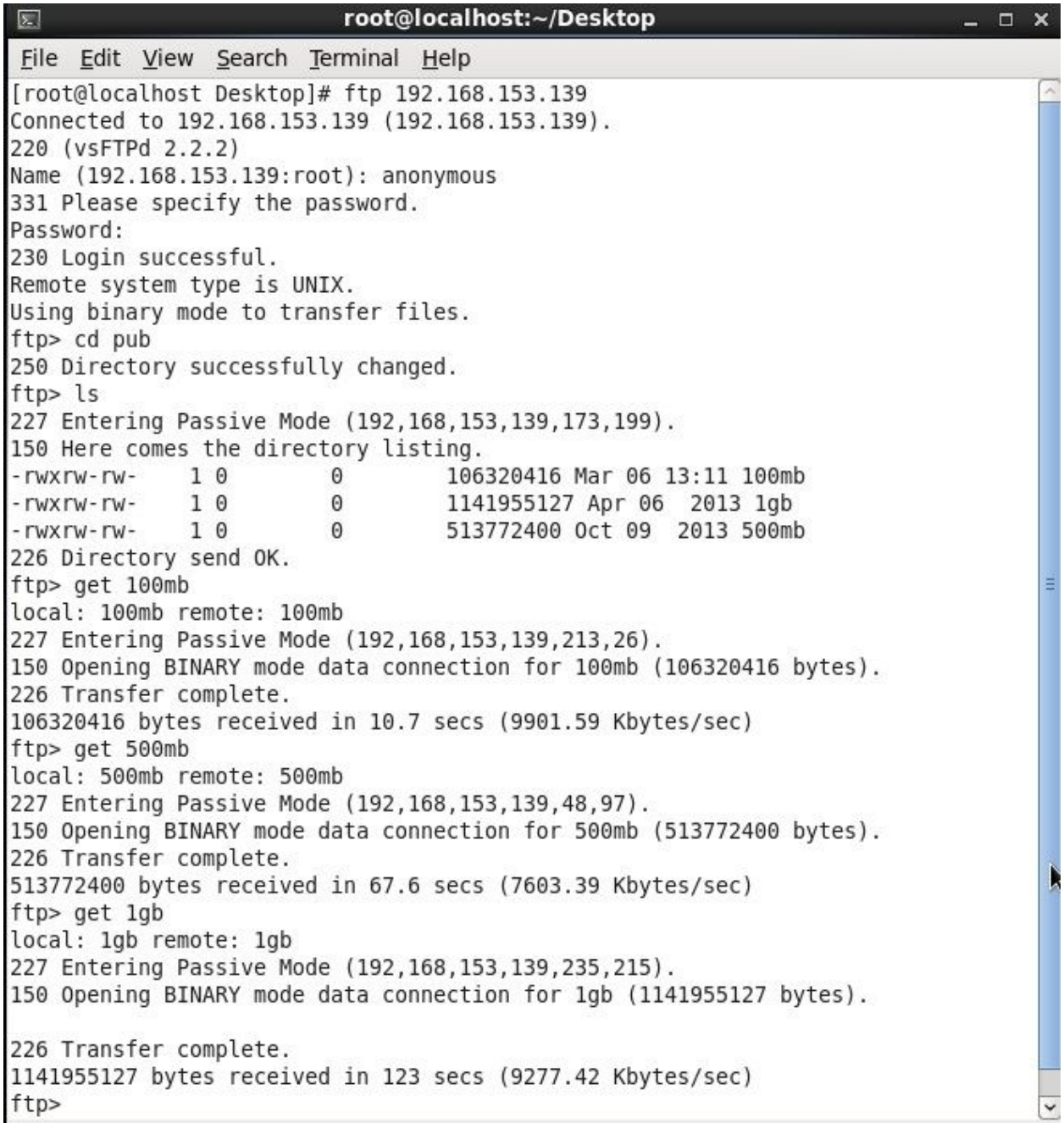
File Size	100 MB	500 MB	1GB
Time (sec)	12.47 (8524.73 Kbytes/sec)	39.92 (12871.34 Kbytes/sec)	86.74 (13165.27 Kbytes/sec)



Downloading Files using Linux system:

Table: 4.27 Downloading Files using Linux system

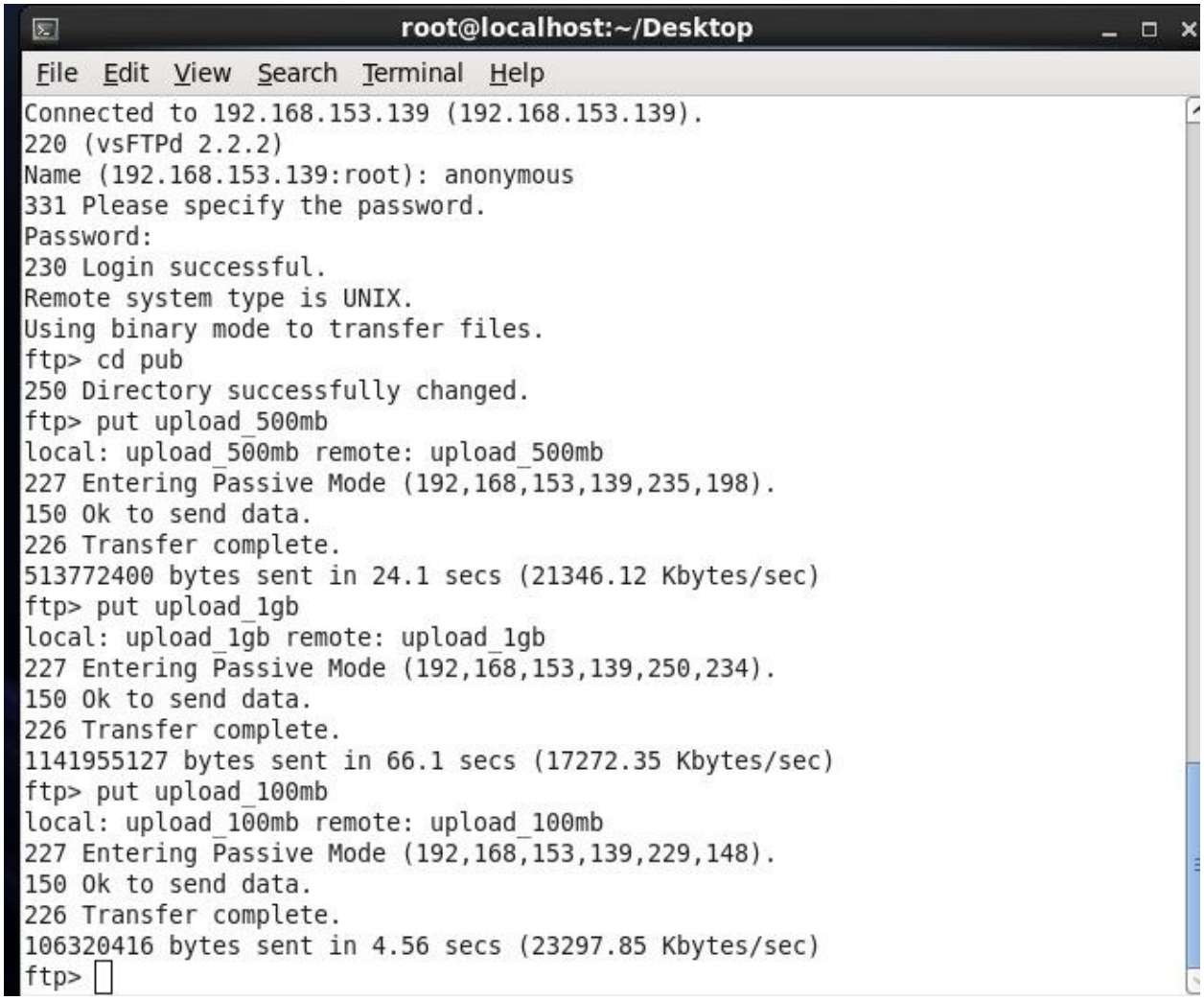
File Size	100 MB	500 MB	1 GB
Time (sec)	10.7 (9901.59 Kbytes/sec)	67.6 (7603.39 Kbytes/sec)	123 (9277.42 Kbytes/sec)



Uploading Files to Linux FTP server (from Linux System):

Table:4.28 Uploading Files to Linux FTP server (from Linux System)

File Size	100 MB	500 MB	1GB
Time (sec)	4.56 (23297.85 Kbytes/sec)	24.1 (21346.12 Kbytes/sec)	66.1 (17272.35 Kbytes/sec)


root@localhost:~/Desktop
File Edit View Search Terminal Help
Connected to 192.168.153.139 (192.168.153.139).
220 (vsFTPd 2.2.2)
Name (192.168.153.139:root): anonymous
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd pub
250 Directory successfully changed.
ftp> put upload_500mb
local: upload_500mb remote: upload_500mb
227 Entering Passive Mode (192,168,153,139,235,198).
150 Ok to send data.
226 Transfer complete.
513772400 bytes sent in 24.1 secs (21346.12 Kbytes/sec)
ftp> put upload_1gb
local: upload_1gb remote: upload_1gb
227 Entering Passive Mode (192,168,153,139,250,234).
150 Ok to send data.
226 Transfer complete.
1141955127 bytes sent in 66.1 secs (17272.35 Kbytes/sec)
ftp> put upload_100mb
local: upload_100mb remote: upload_100mb
227 Entering Passive Mode (192,168,153,139,229,148).
150 Ok to send data.
226 Transfer complete.
106320416 bytes sent in 4.56 secs (23297.85 Kbytes/sec)
ftp>

Chapter 5

CONCLUSION

The performance of windows and Linux have been examined under different circumstances like before and after security related modifications and also calculate the time taken by the ftp servers to download the files of various sizes. At the end of the thesis we found the following results.

The RTT, Bandwidth, latency and Jitter for windows is more than the Linux server. This means Linux is faster while doing any operation in ftp server. The Throughput of Linux ftp server is more than that of Windows as a result Linux works faster as it has more capacity to carry data in particular time.

The various parameters during our observation helps in concluding that Linux perform better than windows and can bear heavy load and traffic. With implementation of FTPS the performance of both operating systems depletes slightly. But after implementing FTPS on both operating systems Linux perform better than windows.

There is one more important aspect which is the ease in installing both the operating systems. This method provides a good idea about the knowledge required according to user's viewpoint. Linux server is slightly tough to install and use instead of Windows server.

RTT

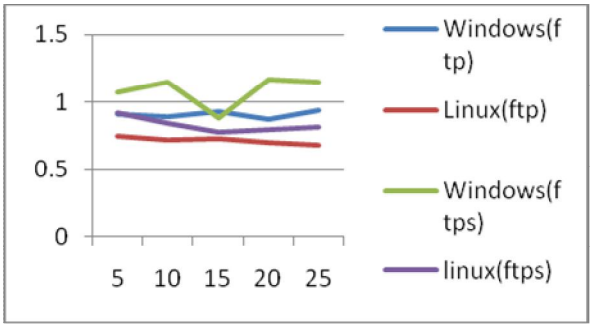


Fig: 5.1 RTT Comparison Graph

Throughputs

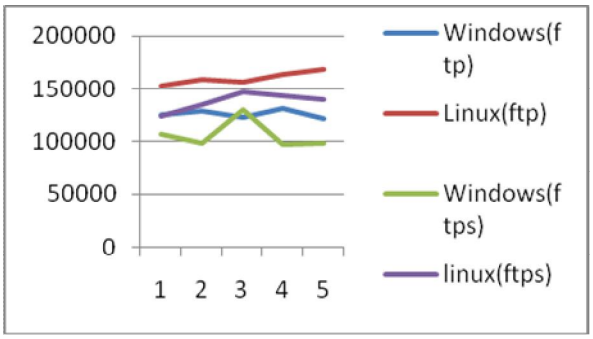


Fig: 5.2 Throughput Comparison Graph

Latency

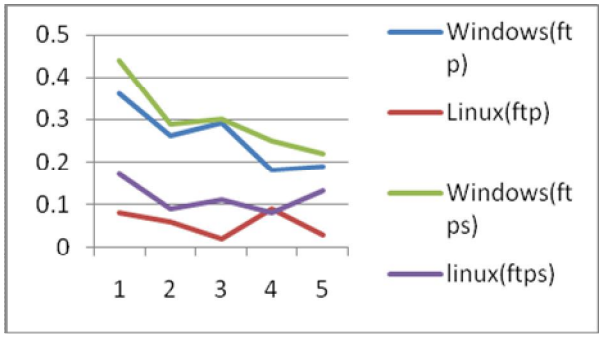


Fig: 5.3 Latency Comparison Graph

Jitter

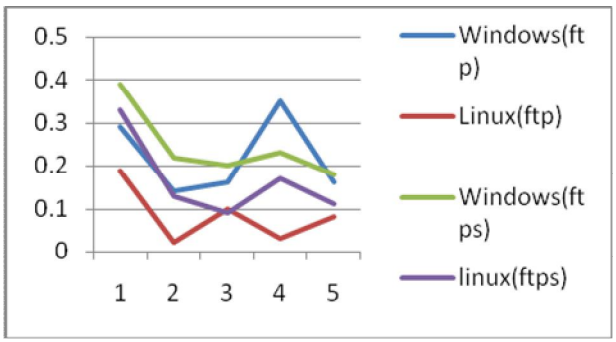


Fig 5.4: Jitter Comparison Graph

Bandwidth

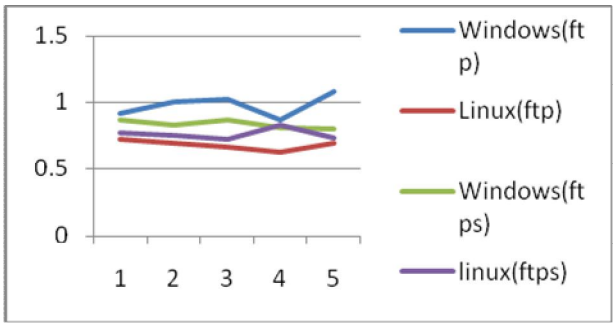


Fig 5.5 Bandwidth Comparison Graph

After analyzing these graphs it is clear that when the ftp and ftp services are used in Linux environment they give the enhanced result and from these graphs it is clear that RTT if the Linux is better than the windows and Latency, Bandwidth and jitter as well gives the better result when they are used in Linux based ftp server or Linux based ftp server.

REFERENCES

[1] Anand Srivastava, “Performance analysis of a Linux based FTP server” 1996

[2] Dag Henning Liodden Sørbø “Increasing the efficiency of a file server by removing redundant data transfers in popular downloads” 2013

[3] Roy Gregory Franks, “Performance Analysis of Distributed Server System” 1999

[4] T. Kiran, “Design and implementation of Transparent Anonymous FTP for Linux” 1998

[5] R. Braden. Requirements for Internet Hosts - Communication Layers. RFC 1122 (Standard), October 1989. Updated by RFCs 1349, 4379, 5884, 6093, 6298.

[6] Robert Braden, David Borman, and Craig Partridge. Computing the internet checksum. ACM SIGCOMM Computer Communication Review, 19(2):86–94, 1989.

[7] J.W. Byers, M. Luby, and M. Mitzenmacher. A digital fountain approach to asynchronous reliable multicast. Selected Areas in Communications, IEEE Journal on, 20(8):1528 – 1540, oct 2002.

[8] Maurice J. Bach. The Design of the UNIX Operating System. Prentice-Hall, Englewood Cli_s, NJ 07632, USA, 1986.

[9] Dah-Ming Chiu and Raj Jain. Analysis of the increase and decrease algorithms for congestion avoidance in computer networks. Computer Networks and ISDN systems, 17(1):1–14, 1989.

[10] Asit Dan, Dinkar Sitaram, and Perwez Shahabuddin. Scheduling policies for an on-demand video server with batching. In Proceedings of the second ACM international conference on Multimedia, pages 15–23. ACM, 1994.

[11] S. Floyd and E. Kohler. Profile for Datagram Congestion Control Protocol (DCCP) Congestion Control, March 2006.

[12] S. Floyd, E. Kohler, and J. Padhye. Profile for Datagram Congestion Control Protocol (DCCP) March 2006.

[13] Behrouz A Forouzan. TCP/IP protocol suite. McGraw-Hill, Inc., 2002.

[14] M. Horowitz and S. Lunt. FTP Security Extensions, Bellcore, October 1997.

[15] Jim Gemmell, Todd Montgomery, Tony Speakman, and Jon Crowcroft. The PGM reliable multicast protocol. Network, IEEE, 17(1):16–22, 2003.

[16] M. Handley, S. Floyd, J. Padhye, and J. Widmer. TCP Friendly Rate Control (TFRC): Protocol Specification. RFC 3448 (Proposed Standard), January 2003. Obsoleted by RFC 5348.

[17] Joao P Hespanha, Stephan Bohacek, Katia Obraczka, and Junsoo Lee. Hybrid modeling of TCP congestion control. In Hybrid Systems: Computation and Control, pages 291–304. Springer, 2001.

[18] M. Hosseini, D.T. Ahmed, S. Shirmohammadi, and N.D. Georganas. A Survey of Application-Layer Multicast Protocols. Communications Surveys Tutorials, IEEE, 9(3):58 –74, quarter 2007.

[19] M. Allman and S. Ostermann. FTP Security Considerations, Ohio University, May 1999.

[20] C. Diot, B.N. Levine, B. Lyles, H. Kassem, and D. Balensiefen. Deployment issues for the IP multicast service and architecture, feb 2000.

[21] J. Postel and J. Reynolds. File transfer protocol (ftp). Technical Report RFC-959, Network Working Group, 1985.

[22] Richard Stevens. Advanced Programming in the UNIX Environment. Ad-dison-Wesley, Reading, MA, USA, 1992.

[23] Theodore Ts_o, Remy Card, and Stephen Tweedie. Design and implemen-tation of the second extended _lesystem. In Proceedings of the First Dutch International Symposium on Linux.

[24] Larry Wall and Randal L. Schwartz. Programming Perl. Nutshell Hand-books. O'Reilly and Associates, Inc., 632 Petuluma Avenue, Sebastopol, CA 95472, _rst edition, January 1991.

[25] S.E. Deering. Host extensions for IP multicasting. RFC 1112 (Standard), August 1989. Updated by RFC 2236.

[26] Jim Gemmell, Jim Gray, and Eve Schooler. Fcast Multicast File Distribution. Network, IEEE, 14(1):58–68, 2000.