



**Federal TVET Institute**  
Department of Information Communication Technology  
Master of ICT Teachers Education



# Learning Guide 3: Web Server Management and Services

(IIS, FTP, DHCP and DNS)

## Information Sheet 3.1: Implementing IIS Web Server and FTP Server

**Dr. Patrick D. Cerna (Associate Professor)**

### **3.1 What is Web Server?**

Web servers are computers that deliver (*serves up*) Web pages. Every Web server has an IP address and possibly a domain name. For example, if you enter the URL (<http://www.tveti.edu.et/index.php>) in your browser, this sends a request to the Web server whose domain name is **www.tveti.edu.et**. The server then fetches the page named `index.php` and sends it to your browser.

Any computer can be turned into a Web server by installing server software and connecting the machine to the Internet. There are many Web server software applications, including public domain software from NCSA and Apache, and commercial packages from Microsoft, Netscape and others.

It's a server used to communicate with Web Browsers as its clients and the communication protocol used in this case is HTTP (HyperText Transfer Protocol). This is why a Web Server is also called an HTTP Server. A **web server** is a computer system that processes requests via HTTP, the basic network protocol used to distribute information on the World Wide Web. The term can refer either to the entire system, or specifically to the software that accepts and supervises the HTTP requests. The most common use of web servers is to host websites, but there are other uses such as gaming, data storage, running enterprise applications, handling email, FTP, or other web uses.

#### **3.1.2 How Web Server Works?**

The primary function of a web server is to store, process and deliver web pages to clients. The communication between client and server takes place using the Hypertext Transfer Protocol (HTTP). Pages delivered are most frequently HTML documents, which may include images, style sheets and scripts in addition to text content.

A user agent, commonly a web browser or web crawler, initiates communication by making a request for a specific resource using HTTP and the server responds with the content of that

resource or an error message if unable to do so. The resource is typically a real file on the server's secondary storage, but this is not necessarily the case and depends on how the web server is implemented.

While the primary function is to serve content, a full implementation of HTTP also includes ways of receiving content from clients. This feature is used for submitting web forms, including uploading of files.

Many generic web servers also support server-side scripting using Active Server Pages (ASP), PHP, or other scripting languages. This means that the behaviour of the web server can be scripted in separate files, while the actual server software remains unchanged. Usually, this function is used to create HTML documents dynamically ("on-the-fly") as opposed to returning static documents. The former is primarily used for retrieving and/or modifying information from databases. The latter is typically much faster and more easily cached but cannot deliver dynamic content.

Web servers are not always used for serving the World Wide Web. They can also be found embedded in devices such as printers, routers, webcams and serving only a local network. The web server may then be used as a part of a system for monitoring and/or administering the device in question. This usually means that no additional software has to be installed on the client computer, since only a web browser is required (which now is included with most operating systems).

### **3.1.2 HTTP - HyperText Transfer Protocol**

It's a stateless request-response based communication protocol. It's used to send and receive data on the Web i.e., over the Internet. This protocol uses reliable TCP connections either for the transfer of data to and from clients which are Web Browsers in this case. HTTP is a stateless protocol means the HTTP Server doesn't maintain the contextual information about the clients communicating with it and hence we need to maintain sessions in case we need that feature for

our Web-applications. Read more about this concept in the articles - Need for Session Tracking and Session Tracking Implementation in Servlets.

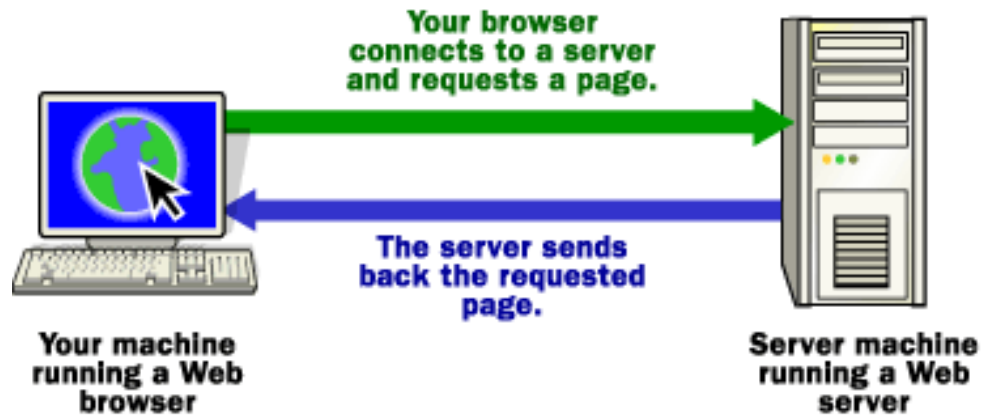
As we just saw that HTTP is a request-response based protocol. That means the client will initiate the communication by sending a request (normally called an HTTP Request) and the HTTP Server (or Web Server) will respond back by sending a response (usually called an HTTP Response). Everytime a client needs to send the request, it first establishes a TCP reliable connection with the Web Server and then transfer the request via this connection. The same happens in case a Web Server needs to send back an HTTP Response to a client. Any of the two parties - the client or the server can prematurely stop the transfer by terminating the TCP connection.

### 3.1.3 How Web Server Works?

#### The Basic Process

**Step 1.** Let's say that you are sitting at your computer, surfing the Web, and you get a call from a friend who says, "I just read a great article! Type in this URL and check it out. It's at [http://www.tveti.edu.et /index.htm](http://www.tveti.edu.et/index.htm)." So you type that URL into your browser and press return. And magically, no matter where in the world that URL lives, the page pops up on your screen. At the most basic level possible, the following diagram shows the steps that brought that page to your screen:

Your browser formed a connection to a Web server, requested a page and received it.



**Figure 3.1 – Architecture of Web Server**

**Step 2.** The browser communicated with a name server to translate the server name "http://www.tveti.edu.et" into an **IP Address**, which it uses to connect to the server machine. The browser then formed a connection to the server at that IP address on port 80. (We'll discuss ports later in this article.)

**Step 3.** Following the HTTP protocol, the browser sent a GET request to the server, asking for the file "http://www.howstuffworks.com/web-server.htm." (Note that **cookies** may be sent from browser to server with the GET request -- see How Internet Cookies Work for details.)

**Step 4.** Finally, the server then sent the HTML text for the Web page to the browser. (Cookies may also be sent from server to browser in the header for the page.) The browser read the HTML tags and formatted the page onto your screen.

Port number	Process name	Protocol used	Description
20	FTP-DATA	TCP	File transfer—data
21	FTP	TCP	File transfer—control
22	SSH	TCP	Secure Shell
23	TELNET	TCP	Telnet
25	SMTP	TCP	Simple Mail Transfer Protocol
53	DNS	TCP and UDP	Domain Name System
67 (client to server) and 68 (server to client)	DHCPv4	UDP	Dynamic Host Configuration Protocol version 4
69	TFTP	UDP	Trivial File Transfer Protocol
80	HTTP	TCP and UDP	Hypertext Transfer Protocol
110	POP3	TCP	Post Office Protocol 3
123	NTP	TCP	Network Time Protocol
143	IMAP	TCP	Internet Message Access Protocol
443	HTTPS	TCP	Secure implementation of HTTP
546 (client to server) and 547 (server to client)	DHCPv6	UDP	Dynamic Host Configuration Protocol version 6
3389	RDP	TCP	Remote Desktop Protocol

**Table 3.1 - Typical Network Services and Protocol Details**

## 3.2 What is an FTP Server?

### 3.2.1 Overview of FTP Server

**FTP** allows you to transfer files between two computers on the Internet. FTP is a simple network protocol based on Internet Protocol and also a term used when referring to the process of copying files when using FTP technology.

To transfer files with FTP, you use a program often called the *client*. An FTP client program initiates a connection to a remote computer running *FTP server* software. After the connection is established, the client can choose to send and/or receive copies of files, singly or in groups. To connect to an FTP server, a client requires a username and password as set by the administrator of the server. Many public FTP archives follow a special convention for that accepts a username of "anonymous." All the IP address tools you need on our free website.

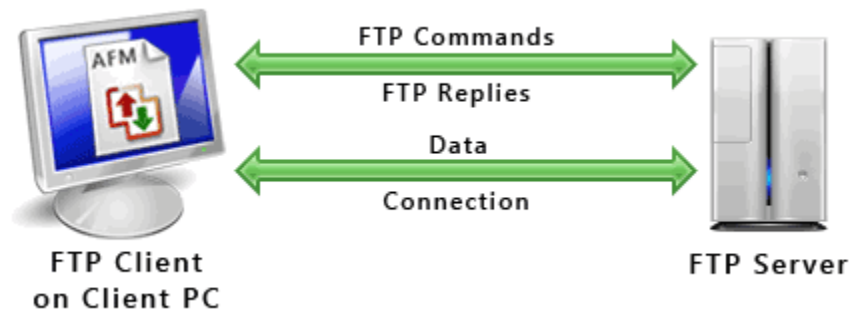
Simple FTP clients are included with most network operating systems, but most of these clients (such as FTP.EXE on Windows) support a relatively unfriendly command-line interface. Many alternative third-party FTP clients (e.g. FileZilla) have been developed that support graphic user interfaces (GUIs) and additional convenience features. In any FTP interface, clients identify the FTP server either by its IP address (such as 192.168.0.1) or by its host name (such as ftp.about.com).

FTP supports two modes of data transfer: plain text (ASCII), and binary. You set the mode in the FTP client. A common error when using FTP is attempting to transfer a binary file (such as a program or music file) while in text mode, causing the transferred file to be unusable.

### 3.2.2 How FTP Server Works?

Data exchange has been important from the early days of computing. A popular means of data exchange is connecting computers to one another.

The File Transfer Protocol (FTP) is used to transfer files between two computers over a network and Internet. In this article we will look at how to work with a FTP client



**Figure 3.2 – Architecture of FTP Server**

### Requirements for using FTP

1. An FTP client like Auto FTP Manager installed on your computer

2. Certain information about the FTP server you want to connect to:
  - a. The **FTP server address**. This looks a lot like the addresses you type to browse web sites.  
Example : Server address is "ftp.videodesk.net".  
Sometimes the server address will be given as a numeric address, like "64.185.225.87".
  - b. A user name and password. Some FTP servers let you connect to them anonymously.  
For anonymous connections, you do not need a user name and password.

### 3.2.3 FTP and Internet Connections

FTP uses one connection for commands and the other for sending and receiving data. FTP has a standard port number on which the FTP server "listens" for connections. A port is a "logical connection point" for communicating using the Internet Protocol (IP). The standard port number used by FTP servers is 21 and is used only for sending commands. Since port 21 is used exclusively for sending commands, this port is referred to as a **command port**. For example, to get a list of folders and files present on the FTP server, the FTP Client issues a "LIST" command. The FTP server then sends a list of all folders and files back to the FTP Client. So what about the internet connection used to send and receive data? The port that is used for transferring data is referred to as a **data port**. The number of the data port will vary depending on the "mode" of the connection.

### 3.2.4 Active and Passive Connection Mode

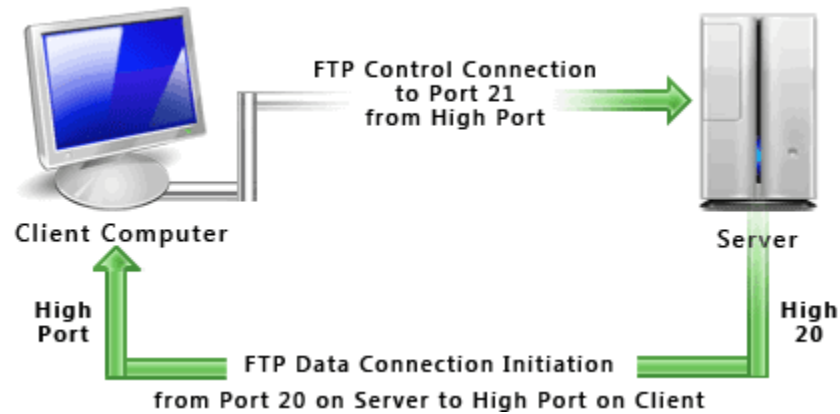
The FTP server may support **Active** or **Passive** connections, or both. In an Active FTP connection, the client opens a port and listens and the server actively connects to it. In a Passive FTP connection, the server opens a port and listens (passively) and the client connects to it. You must grant Auto FTP Manager access to the Internet and to choose the right type of FTP Connection Mode.

Most FTP client programs select passive connection mode by default because server administrators prefer it as a safety measure. Firewalls generally block connections that are "initiated" from the outside. Using passive mode, the FTP client (like FileZilla) is "reaching out" to the server to make the connection. The firewall will allow these outgoing connections, meaning that no special adjustments to firewall settings are required.

If you are connecting to the FTP server using **Active mode** of connection you must set your firewall to

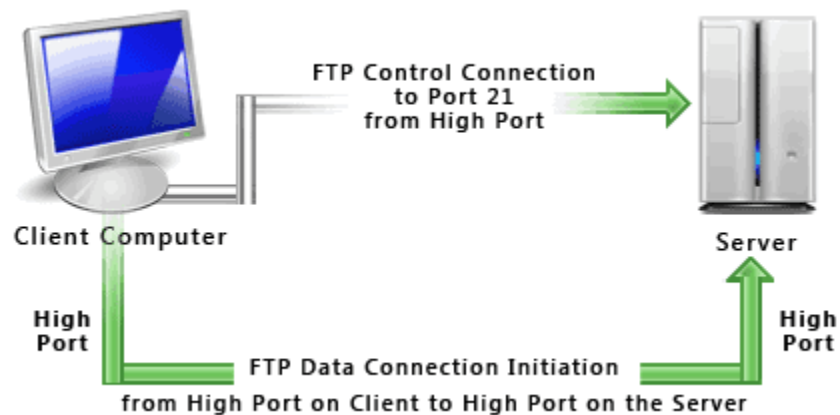


accept connections to the port that your FTP client will open. However, many Internet service providers block incoming connections to all ports above 1024. Active FTP servers generally use port 20 as their data port.



**Figure 3.3 FTP Active Mode**

It's a good idea to use **Passive mode** to connect to an FTP server. Most FTP servers support the Passive mode. For Passive FTP connection to succeed, the FTP server administrator must set his / her firewall to accept all connections to any ports that the FTP server may open. However, this is the server administrator's problem (and standard practice for servers). You can go ahead, make and use FTP connections.



**Figure 3.4 FTP Passive Mode**

Once the FTP Client manages to open the internet connections, one for command and one for data, it starts communicating with the FTP server. You are now ready to transfer your files and folders between the two connected computers with FileZilla.

=====END=====

### References:

1. “*Principles of Network and System Administration*” (2nd Edition), John Wiley and Sons Ltd, Mark Burgess, 2004.
2. “*Essential System Administration*”, 3rd Edition, O’Reilly and Associates Inc., Len Frisch, 2003.
3. “*Running Linux*”, (5th Edition), O’Reilly and Associates Inc., Matthias Kalle Dalheimer and Matt Welsh, 2007.
4. [http://www.webopedia.com/TERM/F/FTP\\_Server.html](http://www.webopedia.com/TERM/F/FTP_Server.html)
5. [http://www.webopedia.com/TERM/W/Web\\_server.html](http://www.webopedia.com/TERM/W/Web_server.html)