

Assignment No. 04

Aim: Introduction to server administration (server administration commands and their applications) and configuration of

a. FTP

b. Web Server

Theory:

Introduction:

A server administrator, or admin has the overall control of a server. This can be in the context of a business organization, where often a server administrator oversees the performance and condition of multiple servers in the business, or it can be in the context of a single person running a game server. The admin for a server typically represents the owners and financiers of the server. Alternatively, an owner can grant administrator rights to a regular player (or clan member) on the server.

The Server Administrator's role is to design, install, administer, and optimize company servers and related components to achieve high performance of the various business applications supported by tuning the servers as necessary. This includes ensuring the availability of client/server applications, configuring all new implementations, and developing processes and procedures for ongoing management of the server environment. Where applicable, the Server Administrator will assist in overseeing the physical security, integrity and safety of the data center/server farm.

a) FTP:

The File Transfer Protocol (FTP) is a standard network protocol used to transfer computer files from one host to another host over a TCP-based network, such as the Internet. FTP is built on client-server architecture and uses separate control and data connections between the client and the server. FTP is built on client-server architecture and uses separate control and data connections between the client and the server. FTP users may authenticate themselves using a clear-text sign-in protocol, normally in the form of a username and password, but can connect anonymously if the server is configured to allow it. For secure transmission that protects the username and password, and encrypts the content, FTP is often secured with SSL/TLS (FTPS). SSH File Transfer Protocol (SFTP) is sometimes also used instead, but is technologically different. The first

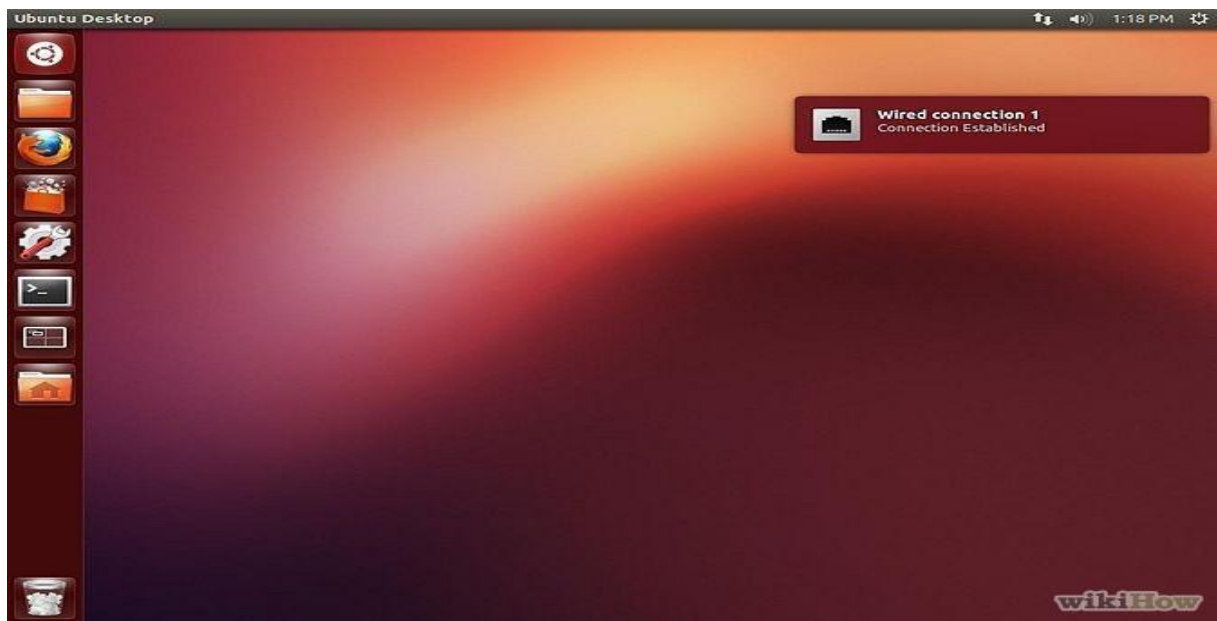
FTP client applications were command-line applications developed before operating systems had graphical user interfaces, and are still shipped with most Windows, Unix, and Linux operating systems. Many FTP clients and automation utilities have since been developed for desktops, servers, mobile devices, and hardware, and FTP has been incorporated into productivity applications, such as Web page editors.

History of FTP server: The original specification for the File Transfer Protocol was written by Abhay Bhushan and published as RFC 114 on 16 April 1971. Until 1980, FTP ran on NCP, the predecessor of TCP/IP. The protocol was later replaced by a TCP/IP version, RFC 765 (June 1980) and RFC 959 (October 1985), the current specification. Several proposed standards amend RFC 959, for example RFC 2228 (June 1997) proposes security extensions and RFC 2428 (September 1998) adds support for IPv6 and defines a new type of passive mode.

Protocol overview:

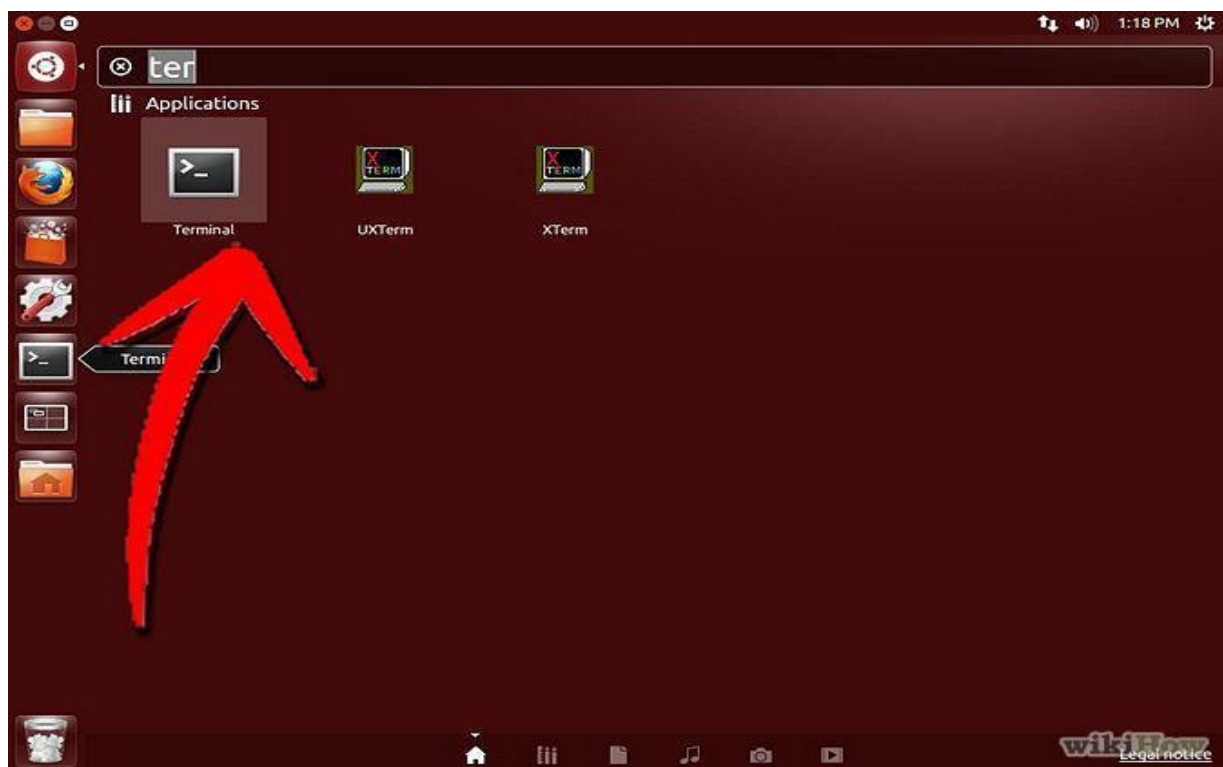
- i) **Communication and data transfer:** FTP may run in active or passive mode, which determines how the data connection is established. In both cases, the client creates a TCP control connection from a random unprivileged port N to the FTP server command port 21. In active modes, the client starts listening for incoming data connections on port N+1 from the server (the client sends the FTP command PORT N+1 to inform the server on which port it is listening).
- ii) **Login:** FTP login utilizes a normal username and password scheme for granting access. The username is sent to the server using the USER command, and the password is sent using the PASS command. If the information provided by the client is accepted by the server, the server will send a greeting to the client and the session will commence. If the server supports it, users may log in without providing login credentials, but the same server may authorize only limited access for such sessions.
- iii) **Anonymous FTP:** A host that provides an FTP service may provide anonymous FTP access. Users typically log into the service with an 'anonymous' (lower-case and case-sensitive in some FTP servers) account when prompted for user name. Although users are commonly asked to send their email address instead of a password, no verification is actually performed on the supplied data. Many FTP hosts whose purpose is to provide software updates will allow anonymous logins.

b) FTP Installation:



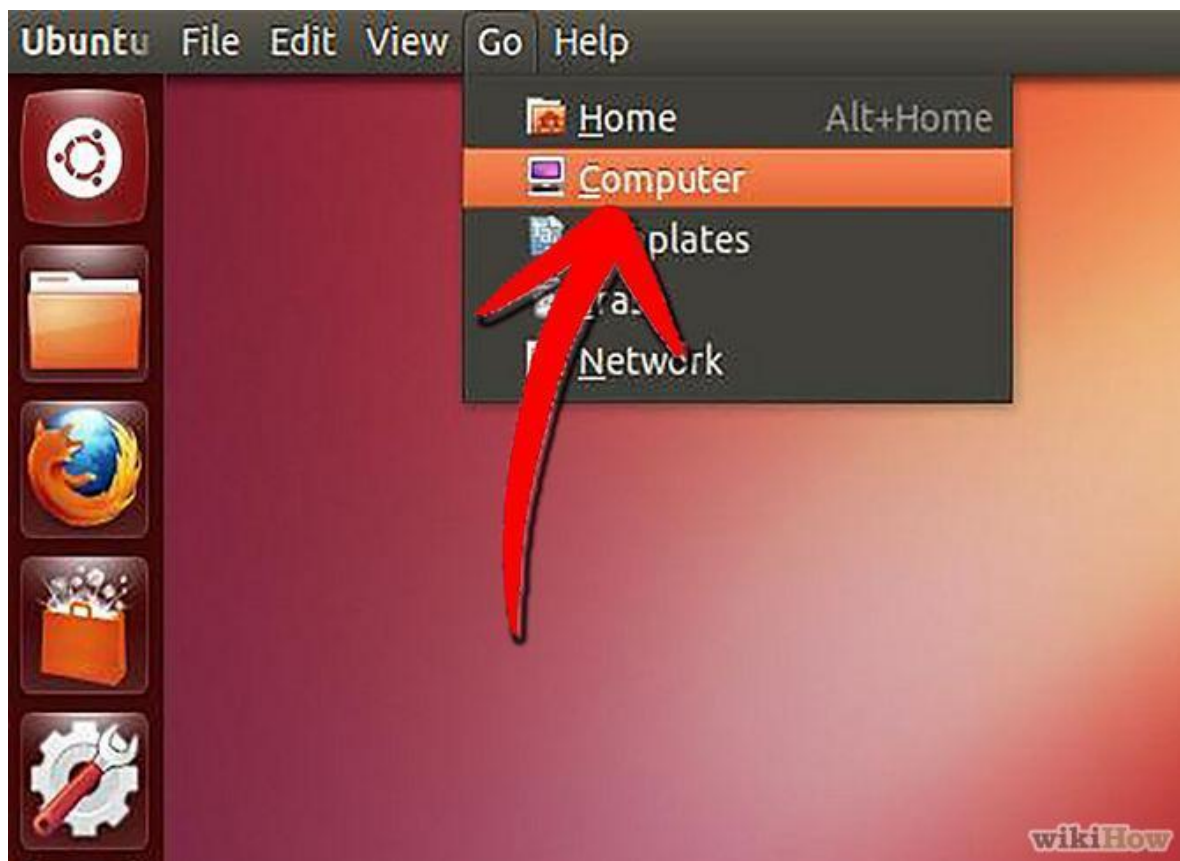
1

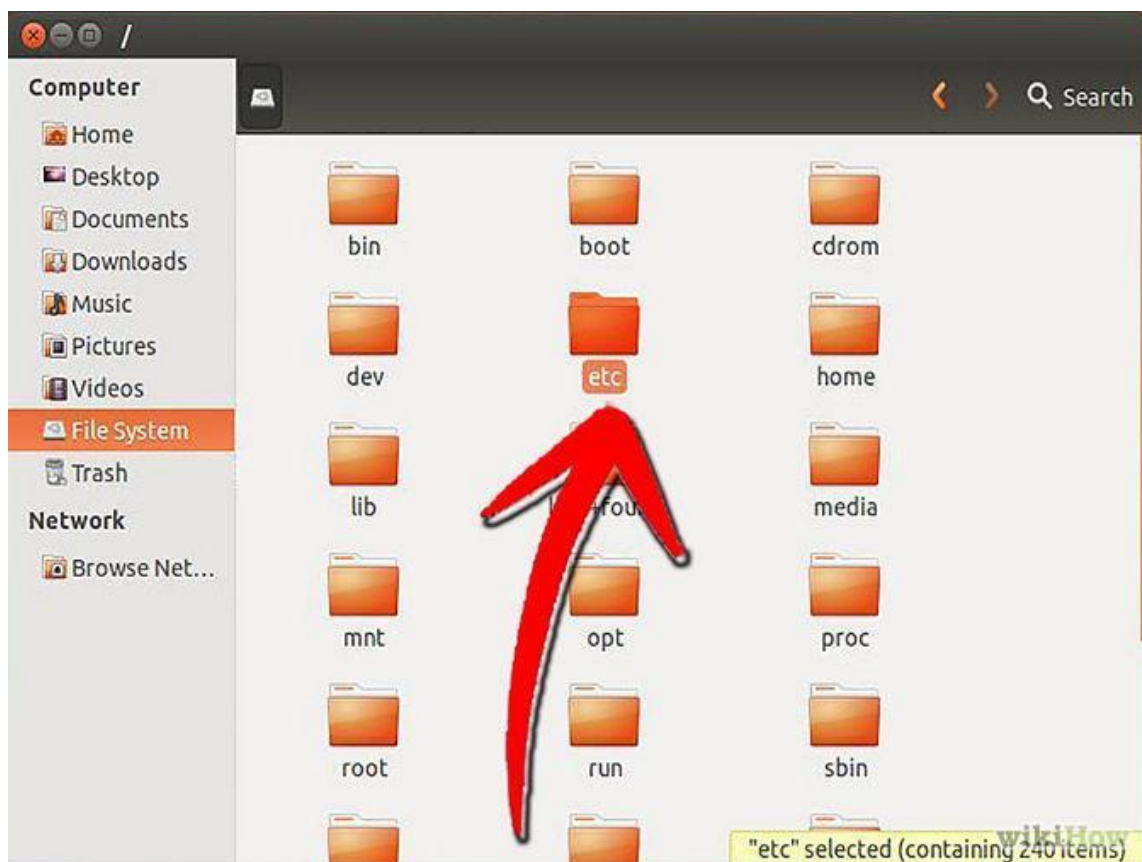
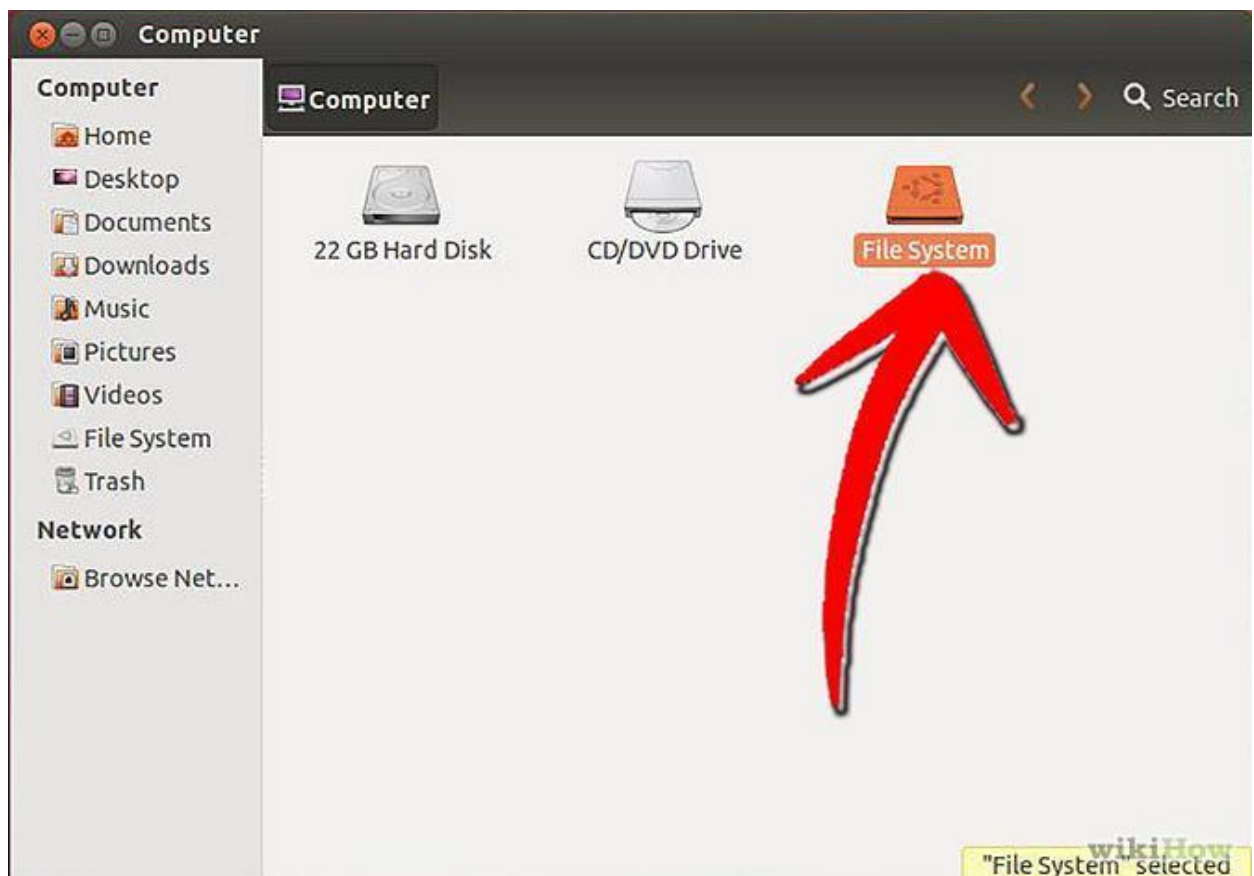
Boot up Ubuntu Linux.

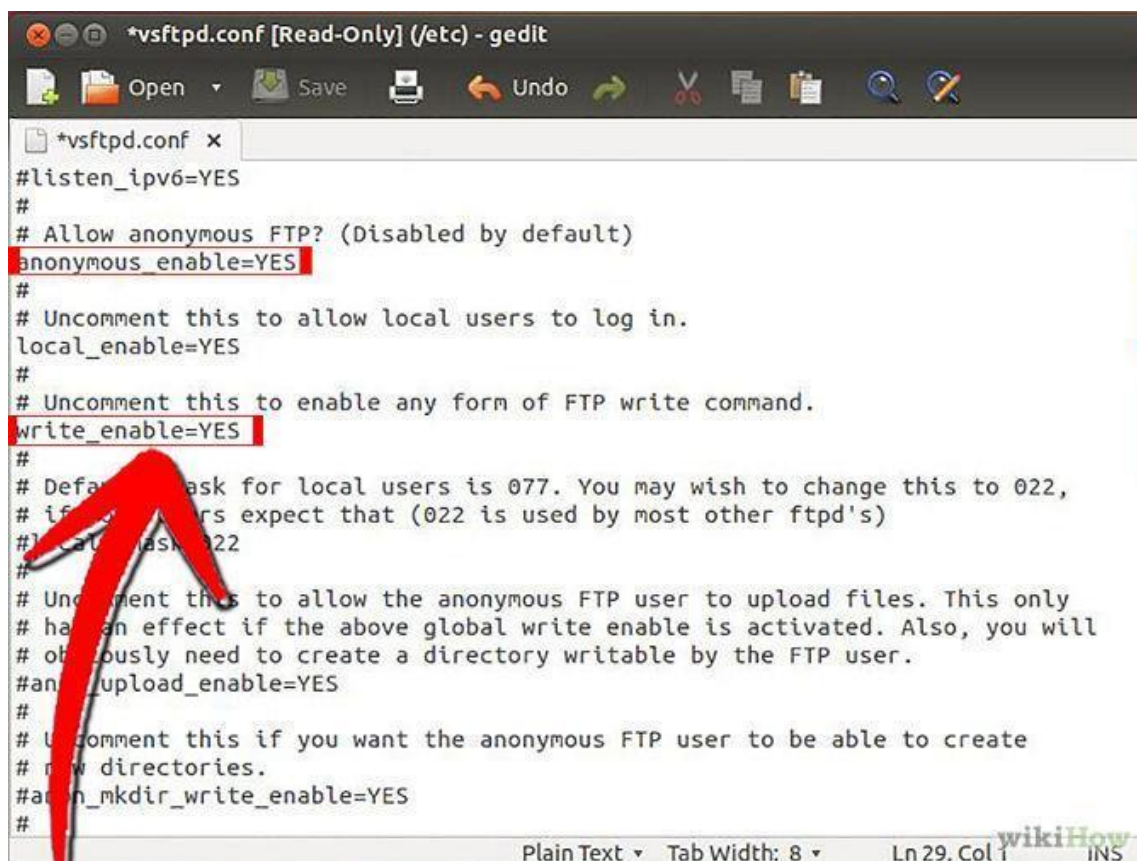
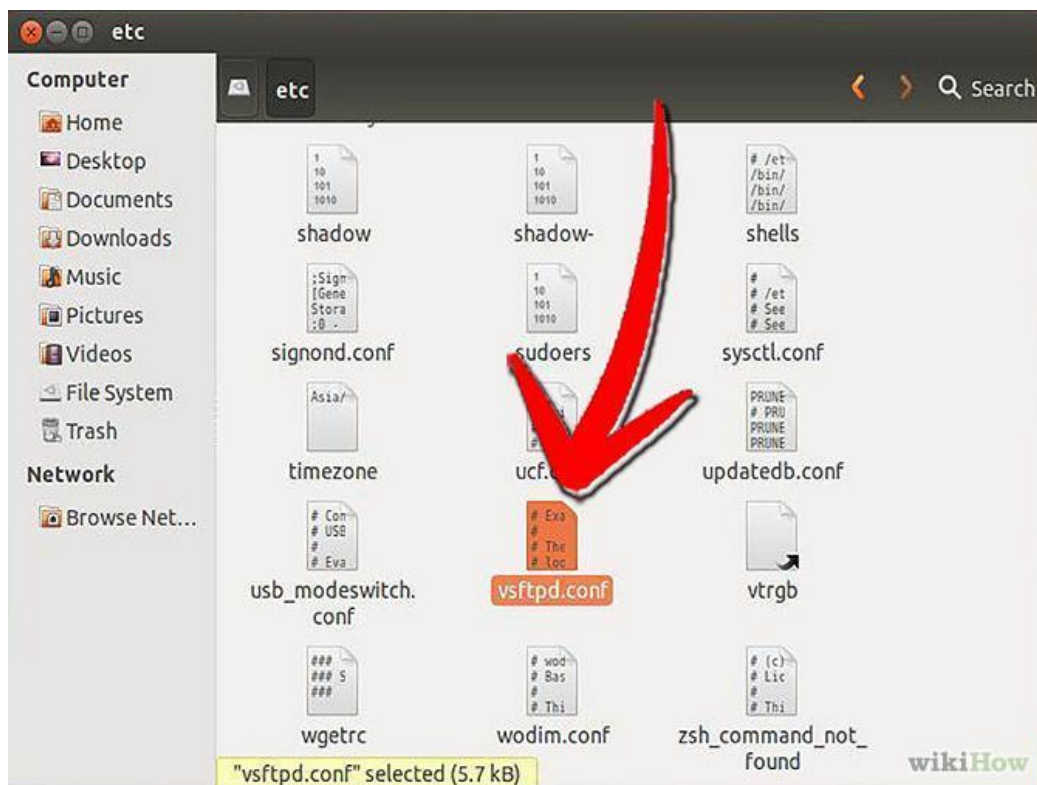


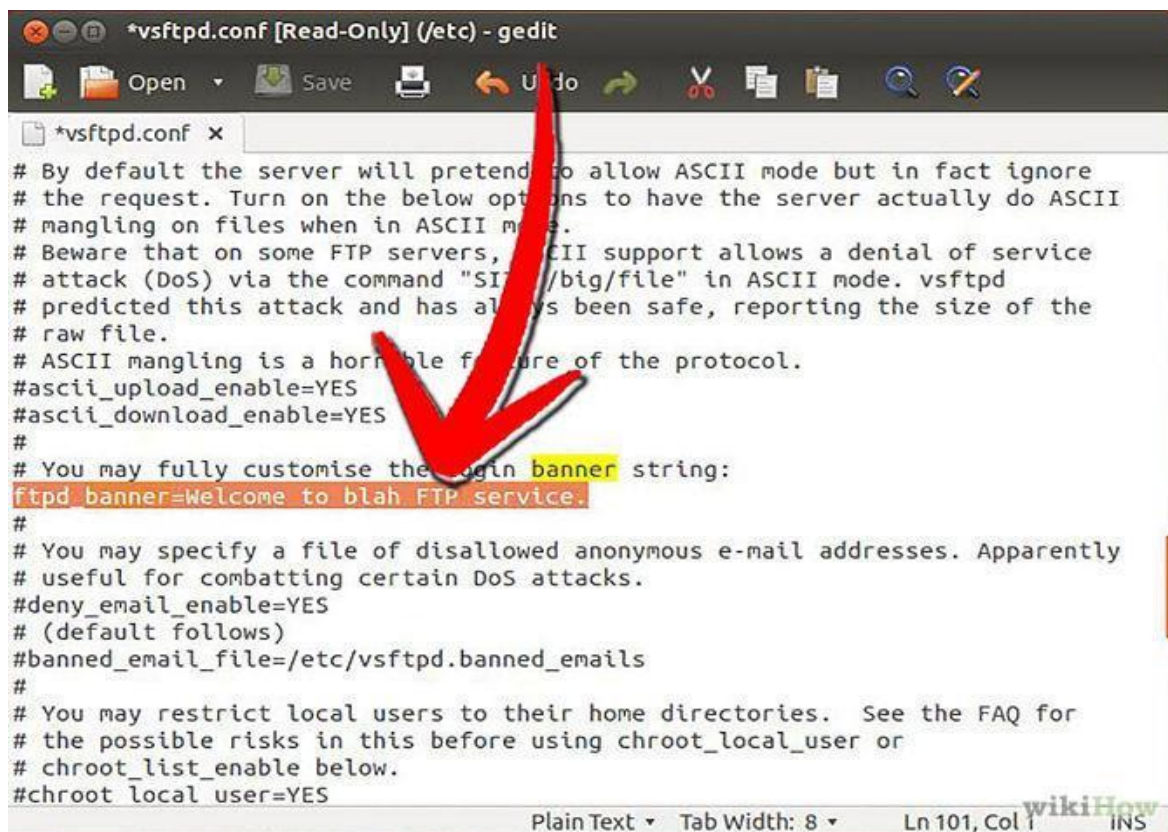
```
anuj@anuj: ~  
anuj@anuj:~$ sudo apt-get install vsftpd  
[sudo] password for anuj:  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following packages were automatically installed and are no longer required:  
  calligra-l10n-engb cdparanoia-k3b k3b-data k3b-i18n kde-l10n-engb  
  kde-l10n-zhcn kdevelop-l10n kdevelop-php-docs-l10n kdevelop-php-l10n  
  language-pack-kde-en libdv4 libdv4read4 libflac++6 libk3b6 libkcddb4  
  libmpcdec6  
Use 'apt-get autoremove' to remove them.  
The following NEW packages will be installed:  
  vsftpd  
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.  
Need to get 132 kB of archives.  
After this operation, 364 kB of additional disk space will be used.  
Get:1 http://in.archive.ubuntu.com/ubuntu/ quantal/main vsftpd i386 2.3.5-3ubuntu1 [132 kB]  
Fetched 132 kB in 12s (10.3 kB/s)  
Preconfiguring packages ...  
Selecting previously unselected package vsftpd.  
(Reading database ... 174406 files and directories currently installed.)  
Unpacking vsftpd (from .../vsftpd_2.3.5-3ubuntu1_i386.deb) ...  
Processing triggers for man-db ...  
Processing triggers for ureadahead ...  
ureadahead will be reprofiled on next reboot  
Setting up vsftpd (2.3.5-3ubuntu1) ...  
vsftpd start/running, process 2976  
Processing triggers for ureadahead ...  
anuj@anuj:~$
```

wikiHow





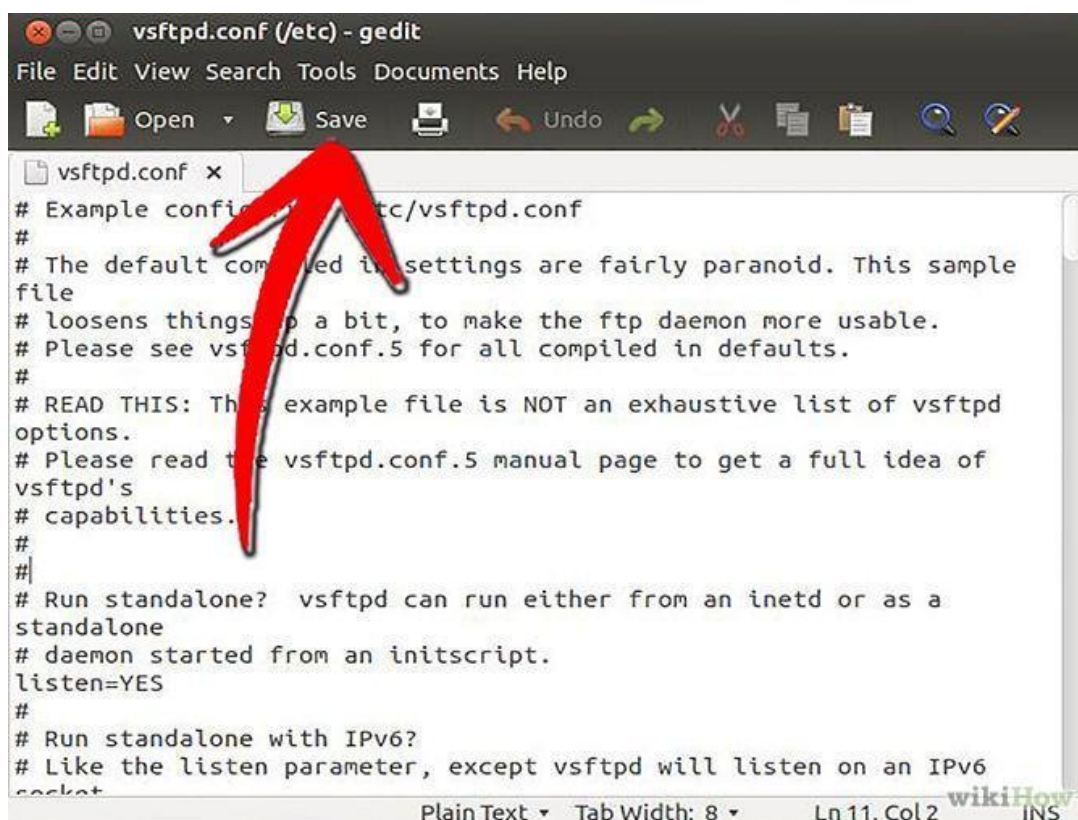




```
*vsftpd.conf [Read-Only] (/etc) - gedit

# By default the server will pretend to allow ASCII mode but in fact ignore
# the request. Turn on the below options to have the server actually do ASCII
# mangling on files when in ASCII mode.
# Beware that on some FTP servers, ASCII support allows a denial of service
# attack (DoS) via the command "SITE /big/file" in ASCII mode. vsftpd
# predicted this attack and has always been safe, reporting the size of the
# raw file.
# ASCII mangling is a horrible feature of the protocol.
#ascii_upload_enable=YES
#ascii_download_enable=YES
#
# You may fully customise the login banner string:
ftpd_banner=Welcome to blah FTP service.
#
# You may specify a file of disallowed anonymous e-mail addresses. Apparently
# useful for combatting certain DoS attacks.
#deny_email_enable=YES
# (default follows)
#banned_email_file=/etc/vsftpd.banned_emails
#
# You may restrict local users to their home directories. See the FAQ for
# the possible risks in this before using chroot_local_user or
# chroot_list_enable below.
#chroot local user=YES
```

Plain Text ▾ Tab Width: 8 ▾ Ln 101, Col 1

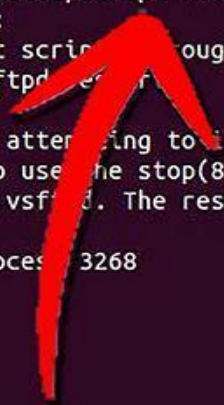


```
vsftpd.conf (/etc) - gedit
File Edit View Search Tools Documents Help

# Example configuration file for vsftpd. See /etc/vsftpd.conf
#
# The default compiled in settings are fairly paranoid. This sample
# file loosens things up a bit, to make the ftp daemon more usable.
# Please see vsftpd.conf.5 for all compiled in defaults.
#
# READ THIS: This example file is NOT an exhaustive list of vsftpd
# options. Please read the vsftpd.conf.5 manual page to get a full idea of
# vsftpd's capabilities.
#
# Run standalone? vsftpd can run either from an inetd or as a
# standalone daemon started from an initscript.
listen=YES
#
# Run standalone with IPv6?
# Like the listen parameter, except vsftpd will listen on an IPv6
# socket.
```


Plain Text ▾ Tab Width: 8 ▾ Ln 11, Col 2

```
anuj@anuj: ~  
anuj@anuj:~$ sudo /etc/init.d/vsftpd restart  
[sudo] password for anuj:  
Rather than invoking init scripts through /etc/init.d, use the service(8)  
utility, e.g. service vsftpd restart.  
  
Since the script you are attempting to invoke has been converted to an  
Upstart job, you may also use the stop(8) and then start(8) utilities,  
e.g. stop vsftpd ; start vsftpd. The restart(8) utility is also available.  
vsftpd stop/waiting  
vsftpd start/running, process 3268  
anuj@anuj:~$
```



wikiHow

```
root@anuj: /srv/ftp  
root@anuj:/srv# cd ftp  
root@anuj:/srv/ftp#
```



wikiHow

Server Commands:

Some useful networking commands you can enter at the command prompt include the following:

Command Name	Description
net	Used to start, stop, and view many networking operations
ipconfig	Displays the IP address and other TCP/IP configuration information for your workstation
hostname	Displays the Microsoft networking computer name; available only in Windows NT , 2000 , and XP
lpq	Displays the print queue status of an LPD printer; available only in Windows NT, 2000, and XP
ping	Verifies existence of remote host (connectivity)
nbtstat	NetBIOS over TCP/IP; gives statistics and technical NetBIOS information for the TCP/IP layer
netstat	Returns protocol statistics and current TCP/IP connections
ipxroute	Displays and modifies IPX routing tables
route	Manipulates TCP/IP routing information
tracert	Displays route taken by an ICMP to a remote host
finger	Displays information about the user; finger is turned off in IU's ADS Domain
arp	Displays or modifies information in the ARP (Address Resolution Protocol) cache
getmac	Lists the MAC (Media Access Control) Address on the computer network interfaces; available in Windows XP only

Linux Network Commands:

The network commands chapter explains various tools which can be useful when networking with other computers both within the network and across the internet, obtaining more information about other computers. This chapter also includes information on tools for network configuration, file transfer and working with remote machines.

Command Name	Description
Netstat	Displays contents of /proc/net files. It works with the LINUX Network Subsystem, it will tell you what the status of ports are i.e. Open, closed, waiting, masquerade connections. It will also display various other things. It has many different options.
Tcpdump	This is a sniffer, a program that captures packets off a network interface and interprets them for you. It understands all basic internet protocols, and can be used to save entire packets for later inspection.
Ping	The ping command (named after the sound of an active sonar system) sends echo requests to the host you specify on the command line, and lists the responses received their round trip time. You simply use ping as: ping ip_or_host_name
Hostname	Tells the user the host name of the computer they are logged into. Note: may be called <i>host</i> .
Traceroute	<i>traceroute</i> will show the route of a packet. It attempts to list the series of hosts through which your packets travel on their way to a given destination. Also have a look at <i>xtraceroute</i> (one of several graphical equivalents of this program).
Findsmb	<i>findsmb</i> is used to list info about machines that respond to SMB name queries (for example windows based machines sharing their hard disk's).
Ifconfig	This command is used to configure network interfaces, or to display their current configuration. In addition to activating and deactivating interfaces with the “up” and “down” settings, this command is necessary for setting an interface's address information if you don't have the <i>ifcfg</i> script.

Conclusion:
