ctc
Control Technology Corp.

CONTROL TECHNOLOGY CORPORATION

Model 5200 Logging & FTP Client Applications Guide

Model 5200 Logging & FTP Client Applications Guide

WARNING: Use of CTC Controllers and software is to be done only by experienced and qualified personnel who are responsible for the application and use of control equipment like the CTC controllers. These individuals must satisfy themselves that all necessary steps have been taken to assure that each application and use meets all performance and safety requirements, including any applicable laws, regulations, codes and/or standards. The information in this document is given as a general guide and all examples are for illustrative purposes only and are not intended for use in the actual application of CTC product. CTC products are not designed, sold, or marketed for use in any particular application or installation; this responsibility resides solely with the user. CTC does not assume any responsibility or liability, intellectual or otherwise for the use of CTC products.

The information in this document is current as of the following Hardware and Firmware revision levels. Some features may not be supported in earlier revisions. See www.ctc-control.com for the availability of firmware updates or contact CTC Technical Support.

Model Number	Hardware Revision	Firmware Revision
5200	All Revisions	>= 5.00.36

TABLE OF CONTENTS

Blank

CHAPTER 1

Introduction

This document provides details about Data Logging and FTP Client file transfer capabilities resident within the 5222 series controllers. The 5222 is an enhanced version of the 5220, consisting of greater memory capacity, internal web server, ftp client, and logging, as detailed below:

Controller Feature	**5220**	**5222**			
FTP Server	X	X			
FTP Client		X			
Web Server	X	X			
Data Logging		X			
CT webHMI		X			

It is suggested that a thorough understanding of the following guides be acquired prior to using the features within this document:

- Model 5200 Remote Administration Guide (Document #951-520001)
- Model 5200 Script Language Guide (Document #951-520003)
- Model 5200 Communications Protocol Guide (Document #951-520002)

Blank

CHAPTER

2

Data Logging

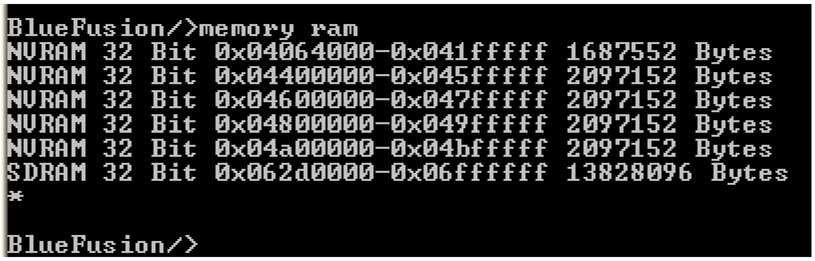
Data Logging, on a 5200 controller, consists of the process of periodically recording collected information in a pre-determined file format. Data may consist of any combination of register contents, a data table cell, time/date information, and/or constant string text. Data is logged in a record format derived from a definition file called ‘log.ini’. This definition file lists all the logging record formats required, each individually selectable at run-time.

Logging Controller Setup

Since Data Logging causes a file to be re-opened, during each write operation, and records appended, a FLASH disk is not supported, only NVRAM. Once a NVRAM disk is created it is referenced as a virtual sub-directory called ‘data’, residing within the /_system/Messages FLASH directory.

RAM Disk Creation

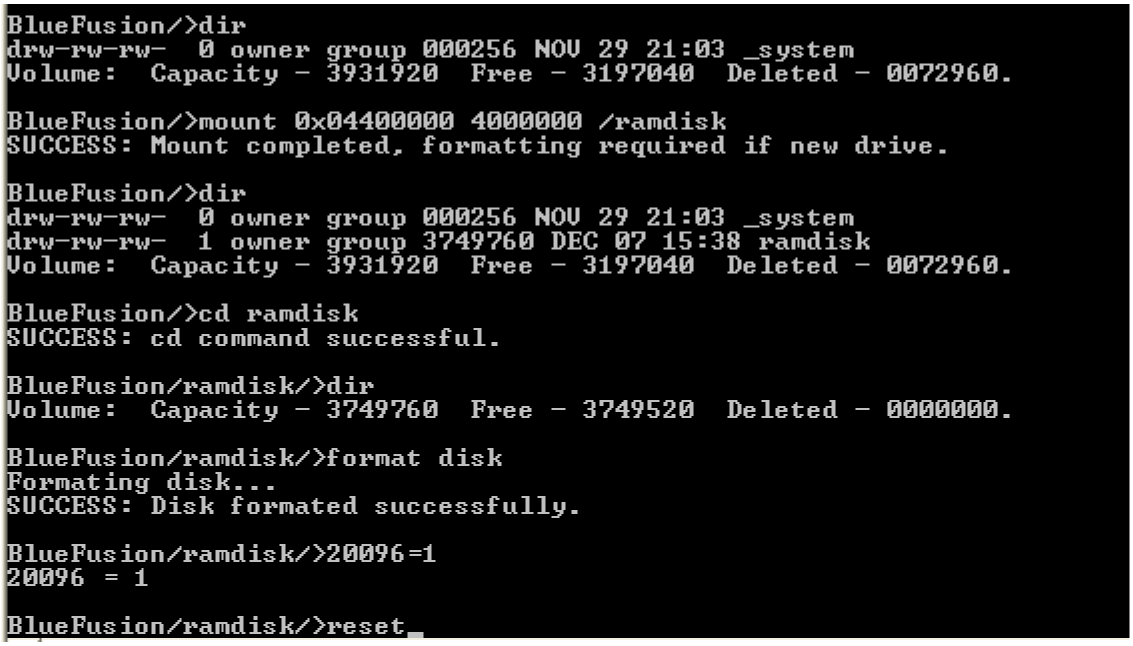
Any non-volatile NVRAM may be used as the NVRAM disk. NVRAM disks are accessed exactly as FLASH disks, with the additional feature of reopening and appending. To view available memory, issue the ‘memory ram’ command via Telnet session:



```
BlueFusion/>memory ram
NVRAM 32 Bit 0x04064000-0x041fffff 1687552 Bytes
NVRAM 32 Bit 0x04400000-0x045fffff 2097152 Bytes
NVRAM 32 Bit 0x04600000-0x047fffff 2097152 Bytes
NVRAM 32 Bit 0x04800000-0x049fffff 2097152 Bytes
NVRAM 32 Bit 0x04a00000-0x04bfffff 2097152 Bytes
SDRAM 32 Bit 0x062d0000-0x06ffffff 13828096 Bytes
*

BlueFusion/>
```

Depending on the configuration of the controller a number of regions can be seen. Any consecutive memory region of the same width may be used. For example 0x04064000 – 0x041fffff is available or a larger consecutive block, 0x04400000 – 0x04bfffff.

The ‘mount’ command, detailed in the *Model 5200 Remote Administration Guide (Document #951-520001)* is used to create the disk as part of the root directory. For example, to summarize the commands necessary to create a 4M NVRAM disk called ‘ramdisk’ starting at 0x04400000:

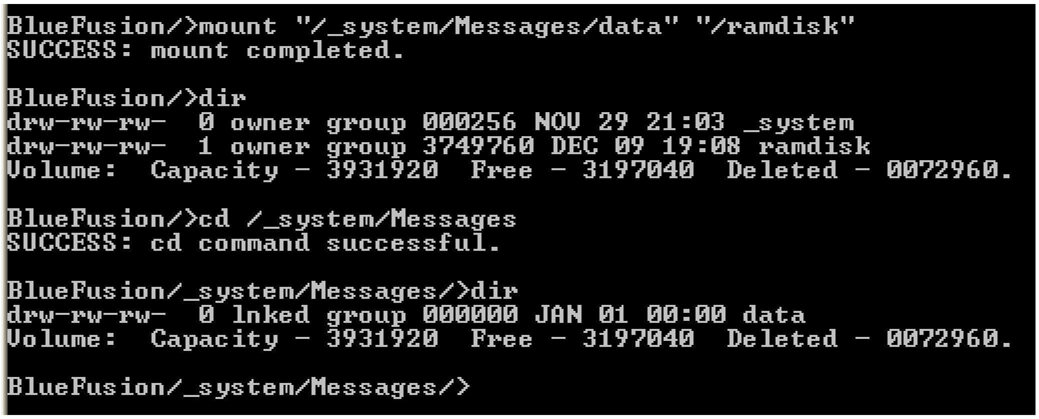


```
BlueFusion/>dir
drw-rw-rw-  0 owner group 000256 NOV 29 21:03 _system
Volume:  Capacity - 3931920  Free - 3197040  Deleted - 0072960.

BlueFusion/>mount 0x04400000 4000000 /ramdisk
SUCCESS: Mount completed, formatting required if new drive.

BlueFusion/>dir
drw-rw-rw-  0 owner group 000256 NOV 29 21:03 _system
drw-rw-rw-  1 owner group 3749760 DEC 07 15:38 ramdisk
Volume:  Capacity - 3931920  Free - 3197040  Deleted - 0072960.

BlueFusion/>cd ramdisk
SUCCESS: cd command successful.

BlueFusion/ramdisk/>dir
Volume:  Capacity - 3749760  Free - 3749520  Deleted - 0000000.

BlueFusion/ramdisk/>format disk
Formating disk...
SUCCESS: Disk formated successfully.

BlueFusion/ramdisk/>20096=1
20096 = 1

BlueFusion/ramdisk/>reset_
```

The controller needs to be rebooted after storing a 1 to register 20096, making the mount permanent. The ‘reset’ command is shown being used above, for rebooting. Never turn the controller off during a disk write operation as this may result in unstable operation.

Virtual Directory Creation

Log files are expected to exist in a virtual directory linked to the /_system/Messages FLASH directory, called ‘data’. This allows you to reference the main flash disk file structure but in actuality be redirected to a NVRAM disk of your choice and size. The ‘mount’ command is used to create a virtual directory. Reference the *Model 5200 Remote Administration Guide*. In summary, the following example is used:



```
BlueFusion/>mount "/_system/Messages/data" "/ramdisk"
SUCCESS: mount completed.

BlueFusion/>dir
drw-rw-rw-  0 owner group 000256 NOV 29 21:03 _system
drw-rw-rw-  1 owner group 3749760 DEC 09 19:08 ramdisk
Volume:  Capacity - 3931920  Free - 3197040  Deleted - 0072960.

BlueFusion/>cd /_system/Messages
SUCCESS: cd command successful.

BlueFusion/_system/Messages/>dir
drw-rw-rw-  0 lnked group 000000 JAN 01 00:00 data
Volume:  Capacity - 3931920  Free - 3197040  Deleted - 0072960.

BlueFusion/_system/Messages/>
```

All log commands will operate on the 'data' sub-directory, with the full path being /_system/Messages/data. The following section will describe this process in more detail.

Virtual directories are volatile and must be re-created upon every power cycle, typically by the use of a script file (_startup.ini).

Logging Record Format and Operation

A predefined format file must reside within the /_system/Messages directory called 'log.ini'. This file contains individual records defining the desired contents to be written to a log file during a logging sequence.

The contents of the 'log.ini' file follows the same format as that of the 'message.ini' file, detailed in the *Model 5200 Communications Protocol Guide (Document #951-520002).* This string-formatted message is similar to the format supported by the 'C' function 'sprintf'. Each message may consist of either text and/or embedded references to any number of registers, where the values will be substituted just prior to writing to the log file. A maximum completed record size (each line in a log file) of 512 bytes is supported. Message format definitions are stored as records in a file called `log.ini` which is located in the `/_system/Messages` subdirectory of the flash disk. Each line of 'log.ini' is considered a record, from 1 to a maximum of 50 pre-defined formats.

The logging of data is controlled by two registers, the *Log Selection Register* (12325) and the *Log String Transfer Register* (12326). The *Log Selection Register* determines the name of the log file to be written, 'Log###.log', where ### is the register contents at the time of an operation. For example writing a 1 to the register defines the name of the file accessed to be 'Log001.log', a 2 would be Log002.log, etc. Up to 999 different sequences may be used.

Writing to the *Log String Transfer Register* (12326) causes the actual write operation to occur, selecting which record in the 'log.ini' file to dynamically format and write to the NVRAM disk file. A read on the *Log String Result Register* (12327) returns the status of the write, as defined below:

- 0 – Success
- 43 – File error, either 'log.ini' is missing or 'Log###.ini' create error.
- 44 – No such record in the 'log.ini' file
- 53 – Write error, check available disk space
- -1 – Default value after setting the *Log Selection Register*

Log.ini Format

As described previously, the 'log.ini' file format is similar in structure to that of the 'C' sprintf function, with additional enhancements. References to registers, data table cells and time/date stamp formats are supported using this extended format:

Register (decimal) - %0#dr<register> or %dr<register>

Example: %05dr13002 (fix size with leading 0's to at least 5 places, reg 13002)
Register (hexidecimal) - %0#xr<register> or %Xr<register>
Example: %05xr13002 (fix size with leading 0's to at least 5 places, reg 13002)

Register (ascii) - %cr<register> or %cr<register>,<length> or %cr<register>,r<register>
Example: %cr12001,r12302 (convert the serial port buffer to ASCII characters)
Example: %cr12001,3 (convert the first 3 serial port buffer registers to ASCII)

Data Table Cell - %0#dD<row>,<col> or %dD<row>,<col>
Example: %05dD1,2 (fix size with leading 0's to at least 5 places, row 1, column 2, from the data table).

Time/Date Stamp - %T!<time/date format>
Example: %T!HH:mm:ss!
%T!MM/DD/YYYY!
Where each below are optional:
HH = hour (24 hour format)
mm = minute
ss = seconds
MM – month
DD – day
YY – year in 2 decimal format, no century.
YYYY – year in 4 decimal format, including century.
E – Day in week, text – Mon, Tue, Wed, Thu, Fri, Sat, or Sun
Z – Time zone information in 5 digit format - <sign>HH:mm from GMT
Note:

- All other characters are treated as filler text, except ending '!'.
- Maximum 48 character Time/Date Stamp string.

Typically a log record consists of text with a 'sprintf" formatted specification, intermixed, as required, with the format detailed above. Therefore, to read register 8501 to be exactly 5 characters with preceding 0's, `%05dr8501` would be inserted in the text string. Note the `%05d` is the same as a 'printf'/'sprintf' and actually uses the exact same function, only enhanced. This means `%05Xr8501` would cause hexadecimal values to be generated. Sample strings using the previous example could be entered in the `log.ini` file as:

```
Analog Value = %05dr8500\r\n
Analog Value = %05dr8501\r\n
```

If the above are the only two entries in the 'log.ini' file, then writing a 2 to the ***Log String Transfer Register*** would cause the second line to be processed and the following to be written to the disk if a 583 were in register 8501:

```
Analog Value = 00583<CR><LF>
```

Log Format Example

Assume a record format of the following is desired:

- Comma delimited format
- Field 1 – MM/DD/YYYY
- Field 2 – HH:mm
- Field 3 – Time tic register 13002
- Field 4 – Analog Input 1
- Field 5 – Analog Input 5
- Field 6 – New line separator <CR> <LF>

The format for this in a 'log.ini' file would be a record inserted with the following format:

```
%T!MM/DD/YYYY!, %T!HH:mm!, %dR13002, %dR8501, %dR8505\r\n
```

If desired constant text could be added or merged around the above data although in this example it was not needed. Additional records could be added to the log.ini file to represent other formats to be logged. In this example only one is required thus it will be referenced as the first record in the 'log.ini' file.

The sequence of events to write a record to a log file with the name 'Log001.log' would be:

1. Set the *Log Execution Register (12325)* to a 1, this will set the *Log String Result Register* to a -1:   12325 = 1
2. Write a 1, for the first record of the 'log.ini' format file, to the *Log String Transfer Register (12326)* to actually do the write operation: 12326 = 1
3. Monitor *Log String Result Register (12327)* for a change from -1 to another value, 0 signifying success. Note that if background threads (Advanced Scripting, chapter 4) are not used the result and write operation occurs immediately upon writing to the Log String Transfer Register, control is returned to the task only after the write is totally complete. This means status is immediately complete and valid:

   0 – Success
   43 – Could not open the file, either 'log.ini' is missing or 'Log###.ini' create error.
   44 – No such record in the 'log.ini' file
   53 – Write error, check available disk space
4. Loop Back to step 2 to write the next record if the file name desired has not changed.

An example of three records of data would be:

```
12/06/2004, 13:41, 234567, 800, 1200
12/06/2004, 13:52, 246000, 801, 1198
12/06/2004, 13:58, 250007, 808, 1190
```

SNAPSHOT

Snapshot capabilities are available which renames a file while it is actively being appended to, allowing for real time uploads. For example if Log001.log is being created, and records appended, a single write to the *Snapshot Execution Register* (12329) renames Log001.log to Snap001.log. The host may then upload Snap001.log, which contains all of the logged data to that instant, while in the background records are still being written to a new Log001.log file. This allows for automatic synchronization of recorded data.

The *Snapshot Result Register* (12330) reflects the results of the operation:

SUCCESS = 0
ERROR_IOACCESS = 53 (log file does not exist or there is an existing Snap file of the desired name)

Log File Deletion

Both Log and Snap files may be deleted by writing the file number to a special deletion register:

```
LOG_DELETION_REGISTER - 12328
SNAP_DELETION_REGISTER – 12331
```

Upon deletion the respective status register will contain either a ‘0’ for success, or a 53 (ERROR_IOACCESS, no file existed).

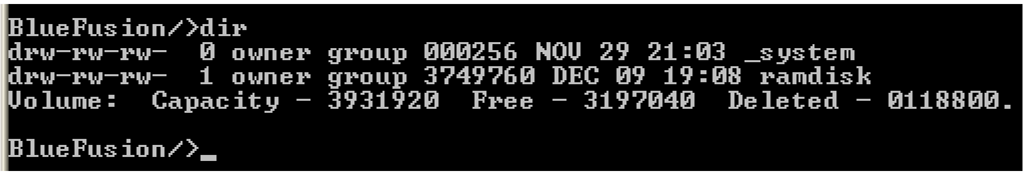
```
LOG_STRING_RESULT_REGISTER (12327)
SNAP_RESULT_REGISTER (12330)
```

Log Disk Maintenance

Periodically the NVRAM disk will reach maximum capacity. Unlike a disk on your PC, when you delete a log file, on a NVRAM disk, its space is still consumed on the drive itself. Only its directory reference is destroyed. This is necessary due to the real-time nature of the controller. NVRAM can be re-formatted quickly (< 300 milliseconds for a 4M disk), using the ‘format disk’ script command. Additionally, formatting must be done at a time when no other operations are occurring to the drive, otherwise a reset is required. Depending upon how much data is being logged and the size of your NVRAM this may be once a week, a month, or even a year. Once formatted all original disk space is returned. Background script routines typically are written to periodically wake and check space at a certain time, format the drive, and return to sleep.

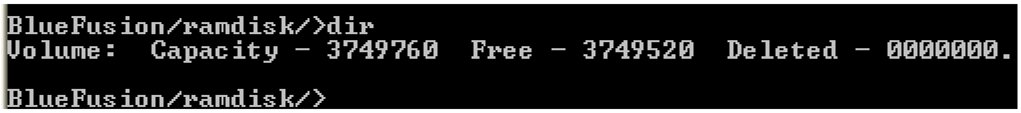
The FLASHDISK_SELECTION_REGISTER (register 12314) is used to select the active disk volume whose remaining space is to appear in the FLASHDISK_SPACEAVAIL_REGISTER (12315). By default it is 0, reflecting the root volume. As each volume is mounted it becomes a new volume and is assigned a number, incrementally. To determine a volume number, simply view a directory at the root level. The second column is the volume number; you will see a 0 in the _system row and a 1 in the mounted ramdisk row. Reading register 12315, will retrieve the actual bytes remaining.

Directory of the root drive gives the free space on that drive:



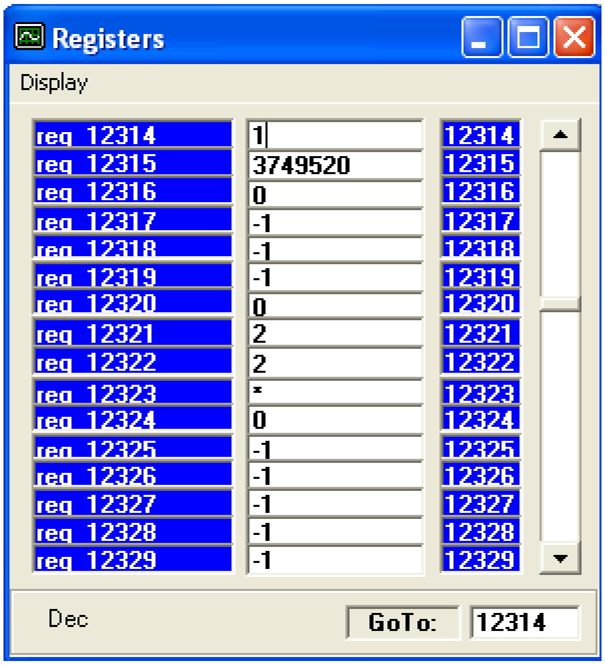
```
BlueFusion/>dir
drw-rw-rw-  0 owner group 000256 NOV 29 21:03 _system
drw-rw-rw-  1 owner group 3749760 DEC 09 19:08 ramdisk
Volume:  Capacity - 3931920  Free - 3197040  Deleted - 0118800.

BlueFusion/>_
```

Changing to the ramdisk drive shows the free space on that drive:



```
BlueFusion/ramdisk/>dir
Volume:  Capacity - 3749760  Free - 3749520  Deleted - 0000000.

BlueFusion/ramdisk/>
```

Below, using CTCMON, it is shown that volume 1 is being read, which is represented by the value of ‘1’ in register 12314. Note the value in register 12315, it refers to the size in bytes available in the NVRAM disk labeled ‘ramdisk’. It is the same as what was displayed in the Telnet session, above.



reg 12314	1	12314
reg 12315	3749520	12315
reg 12316	0	12316
reg 12317	-1	12317
reg 12318	-1	12318
reg 12319	-1	12319
reg 12320	0	12320
reg 12321	2	12321
reg 12322	2	12322
reg 12323	*	12323
reg 12324	0	12324
reg 12325	-1	12325
reg 12326	-1	12326
reg 12327	-1	12327
reg 12328	-1	12328
reg 12329	-1	12329

GoTo: 12314

Blank

CHAPTER 3

FTP Client

The 5200 Controller supports simultaneous FTP Server and/or Client sessions. FTP provides a standard means to transfer files to/from the controller from a host computer. When using the built-in FTP Server, the remote computer is the master, initiating file transfers. This section details the use of the 5200 FTP Client mode, where the controller is capable of initiating its own file transfers, to a remote host FTP Server. Simple files may be uploaded/downloaded, as well as firmware updates and new Quickstep application programs. Additional information with regards to the FTP Server capability may be found in the *Model 5200 Remote Administration Guide (Document #951-520001).*

Discussed in chapter 4, when using FTP Client commands within a script it must be executed as a background threaded operation. Background execution of a script would occur by adding 1000 to the base file script number. For example, a 1001 written to the 'Script Execution Register' (12311) results in Script001.ini executing as a background thread. Ftp Client operations should only be run from a command line within telnet or as a background thread, not as part of a Quickstep task (file numbers, 001 – 999).

Setup

A FTP Server may reside on virtually any host or workstation computing environment. Software is available for Windows 2000, XP Pro, 2003 Server, and Unix/Linux environments. Unix/Linux, Windows 2000/2003 Servers, and XP Professional contain the service as part of the installation CD. Windows 2000 must use a third party package such as can be found at the following web links:

http://www.bpftppro.com (reference Appendix A)
http://www.serv-u.com/
http://www.candc1.com/ftpservu/index.cfm
http://www.abraxis.com/ipswitch/wsftp-server.htm
http://www.bykeyword.com/downloads/software-1/download-1977.html

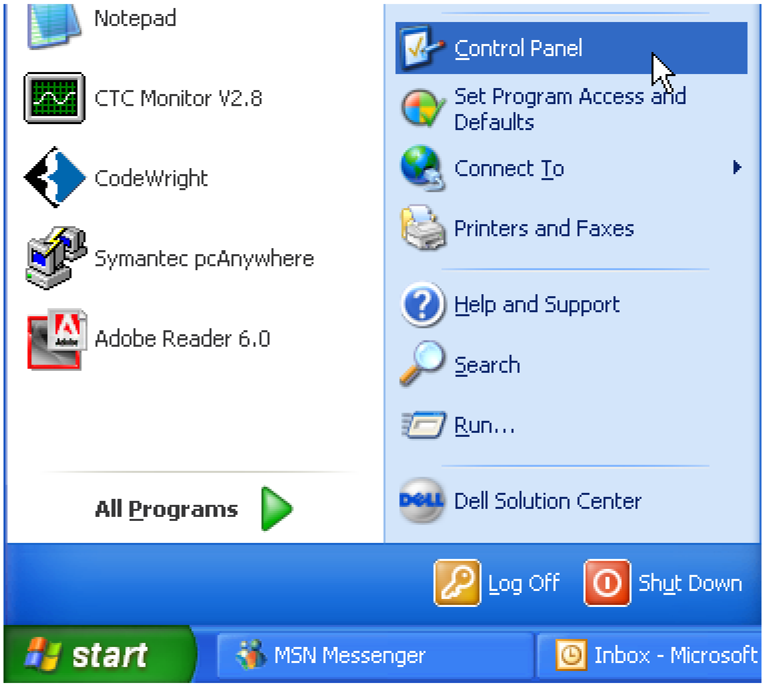
Specifications of what to use, and how to configure the environment, is beyond the scope of this document. A very good reference for installing the resident FTP Server, and its use, on a Windows XP Professional computer is available online from PCSTATS. The title of their article is called: "Beginner's Guides: Setting up a FTP Server in WinXP". It is strongly suggest that you read this article if you are not familiar with FTP Servers. The web link is:

http://www.pcstats.com/articleview.cfm?articleID=1491

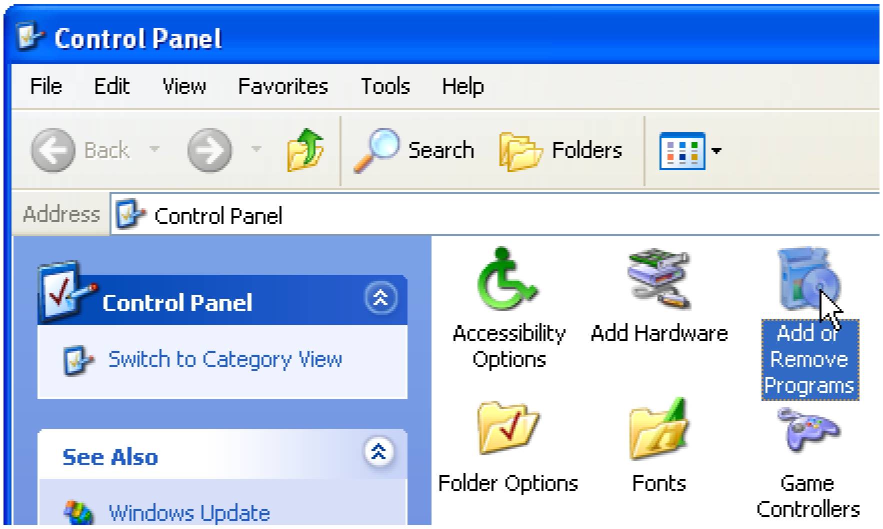
The article provides not only step by step procedures but a very good background on ftp itself, as well as security considerations.

A quick summary of the installation steps involved are as follows:

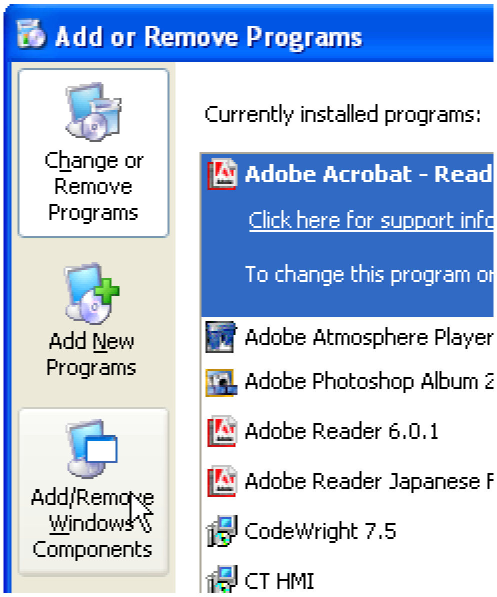
1. While logged in with administrative rights open a Control Panel Window.



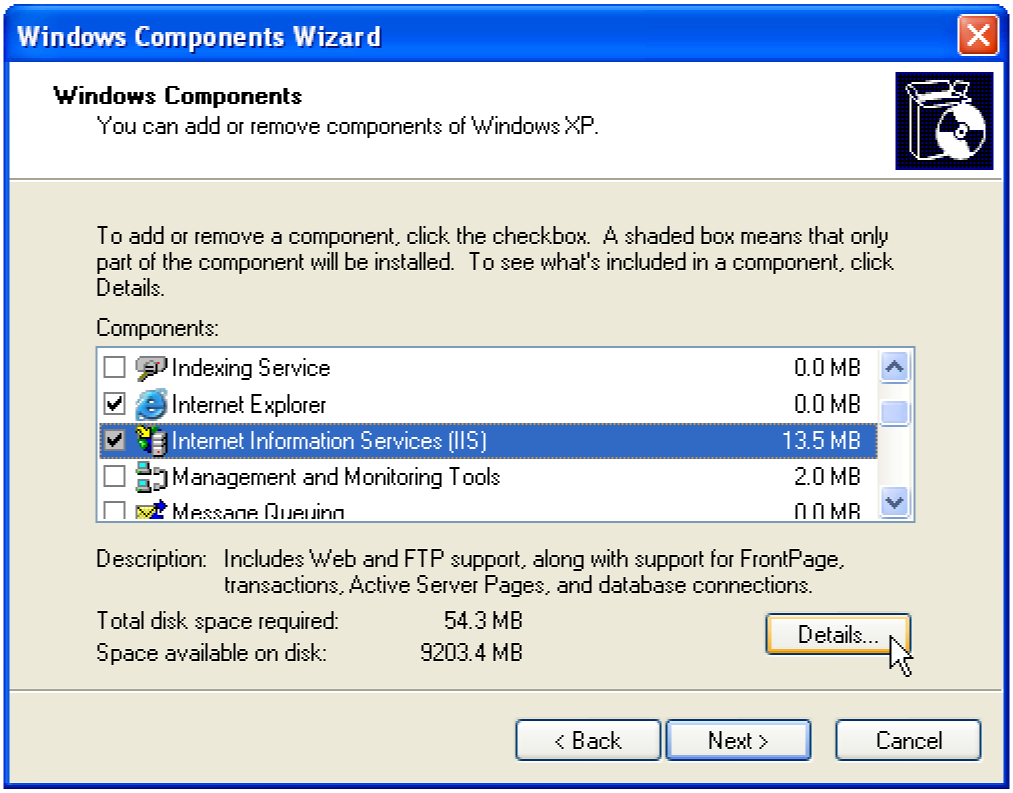
2. Double-click “Add or Remove Programs”.



3. At the left side of the window click “Add/Remove Windows Components”.



4. The Ftp Server option is listed under the IIS component. Select it then the Details button.



Windows Components Wizard

Windows Components

You can add or remove components of Windows XP.

To add or remove a component, click the checkbox. A shaded box means that only part of the component will be installed. To see what's included in a component, click Details.

Components:

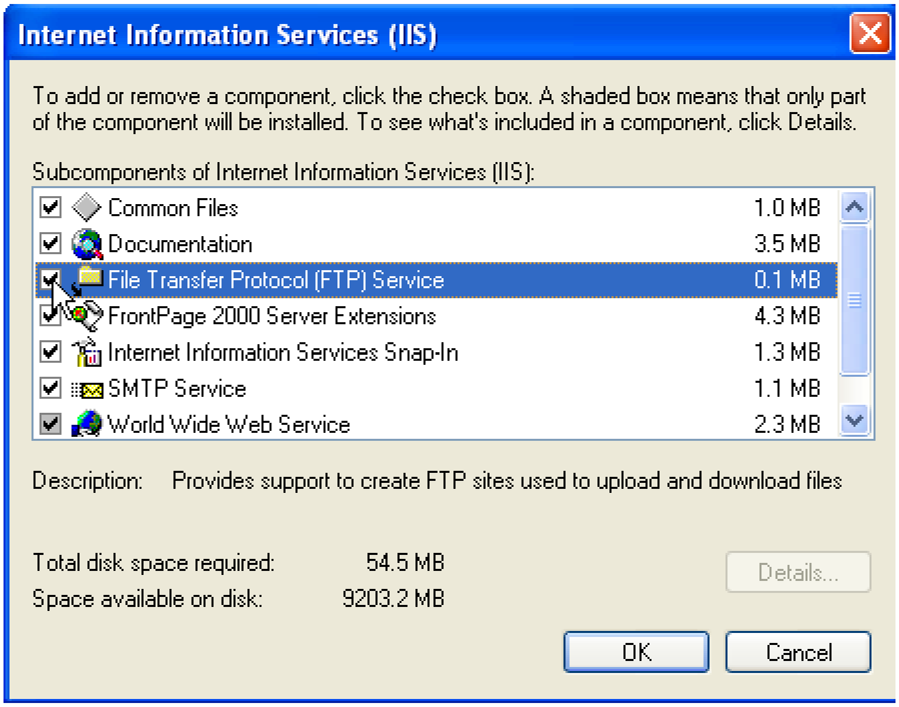
☐	Indexing Service	0.0 MB
☑	Internet Explorer	0.0 MB
☑	Internet Information Services (IIS)	13.5 MB
☐	Management and Monitoring Tools	2.0 MB
☐	Message Queuing	0.0 MB

Description: Includes Web and FTP support, along with support for FrontPage, transactions, Active Server Pages, and database connections.

Total disk space required: 54.3 MB

Space available on disk: 9203.4 MB

5. Make sure the check boxes are as below and then click ‘OK’.



Internet Information Services (IIS)

To add or remove a component, click the check box. A shaded box means that only part of the component will be installed. To see what's included in a component, click Details.

Subcomponents of Internet Information Services (IIS):

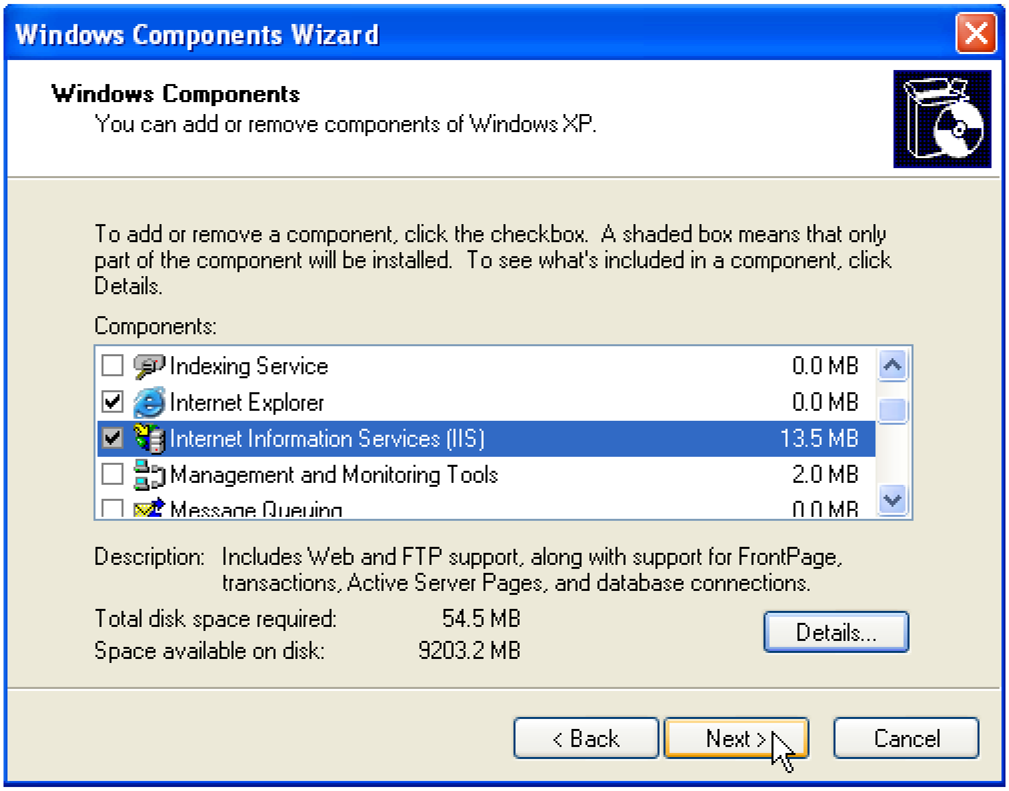
☑	Common Files	1.0 MB
☑	Documentation	3.5 MB
☑	File Transfer Protocol (FTP) Service	0.1 MB
☑	FrontPage 2000 Server Extensions	4.3 MB
☑	Internet Information Services Snap-In	1.3 MB
☑	SMTP Service	1.1 MB
☑	World Wide Web Service	2.3 MB

Description: Provides support to create FTP sites used to upload and download files

Total disk space required: 54.5 MB

Space available on disk: 9203.2 MB

6. Click ‘Next’ to proceed.



Windows Components Wizard

Windows Components

You can add or remove components of Windows XP.

To add or remove a component, click the checkbox. A shaded box means that only part of the component will be installed. To see what's included in a component, click Details.

Components:

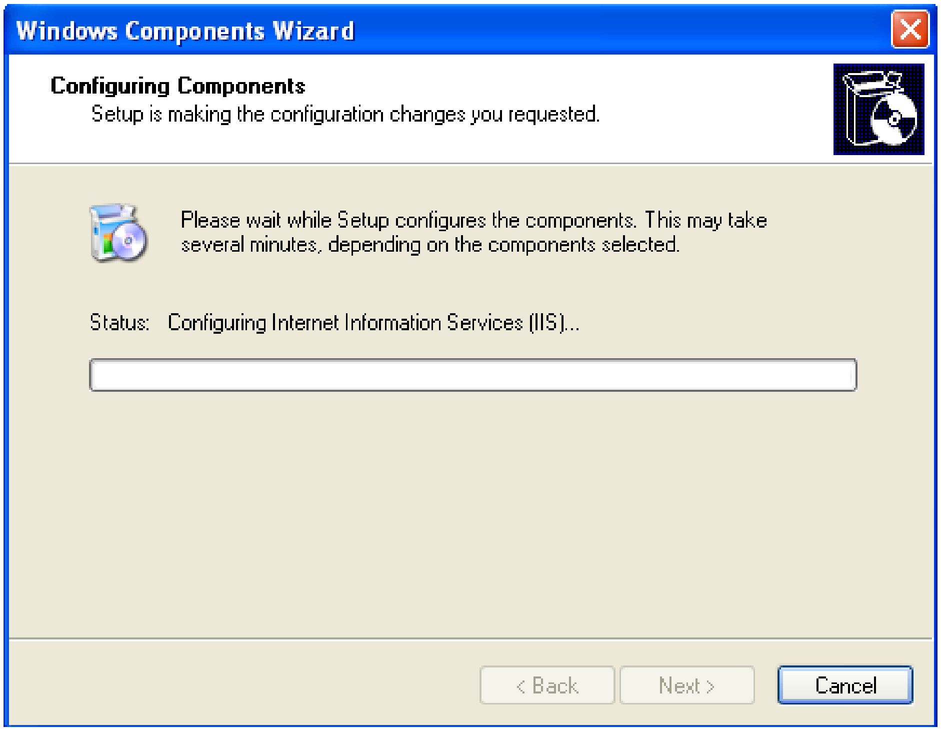
☐	Indexing Service	0.0 MB
☑	Internet Explorer	0.0 MB
☑	Internet Information Services (IIS)	13.5 MB
☐	Management and Monitoring Tools	2.0 MB
☐	Message Queuing	0.0 MB

Description: Includes Web and FTP support, along with support for FrontPage, transactions, Active Server Pages, and database connections.

Total disk space required: 54.5 MB

Space available on disk: 9203.2 MB

7. IIS components will begin to install.



Windows Components Wizard

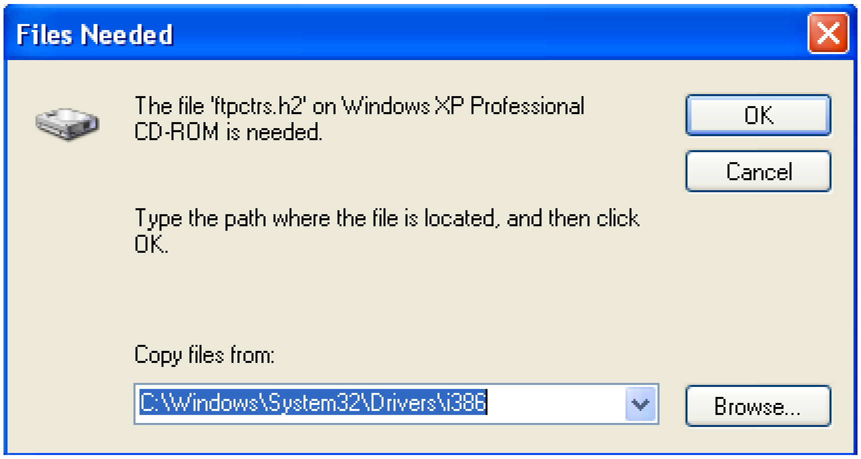
Configuring Components

Setup is making the configuration changes you requested.

Please wait while Setup configures the components. This may take several minutes, depending on the components selected.

Status: Configuring Internet Information Services (IIS)...

8. The prompt below may appear, requiring the XP Professional install CD.


Files Needed

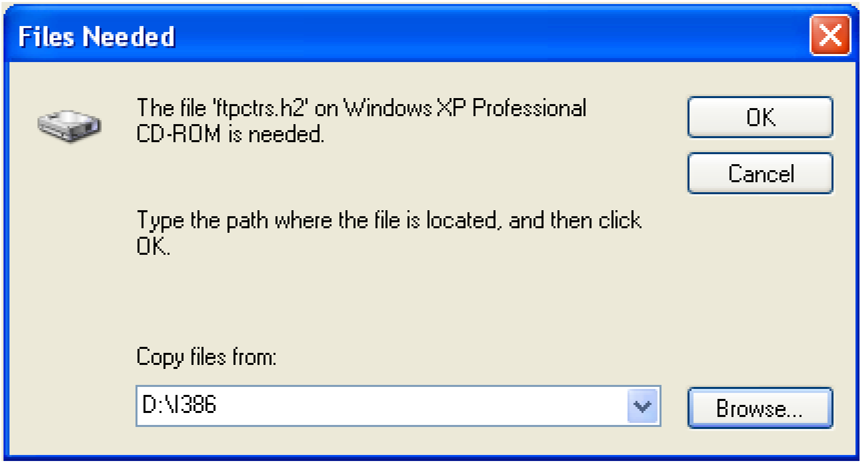
The file 'ftpctrs.h2' on Windows XP Professional CD-ROM is needed.

Type the path where the file is located, and then click OK.

Copy files from:

C:\Windows\System32\Drivers\i386

9. Insert the CD and set the directory as appropriate.


Files Needed

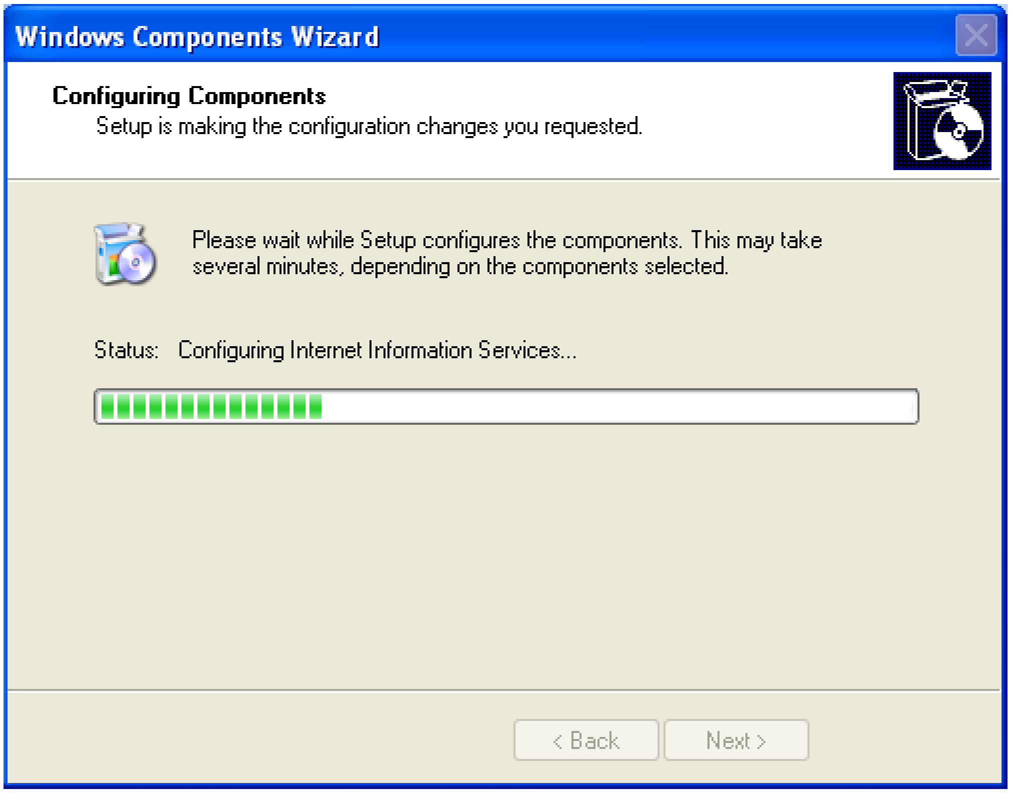
The file 'ftpctrs.h2' on Windows XP Professional CD-ROM is needed.

Type the path where the file is located, and then click OK.

Copy files from:

D:\I386

10. Click OK when the path is correct and installation will begin.



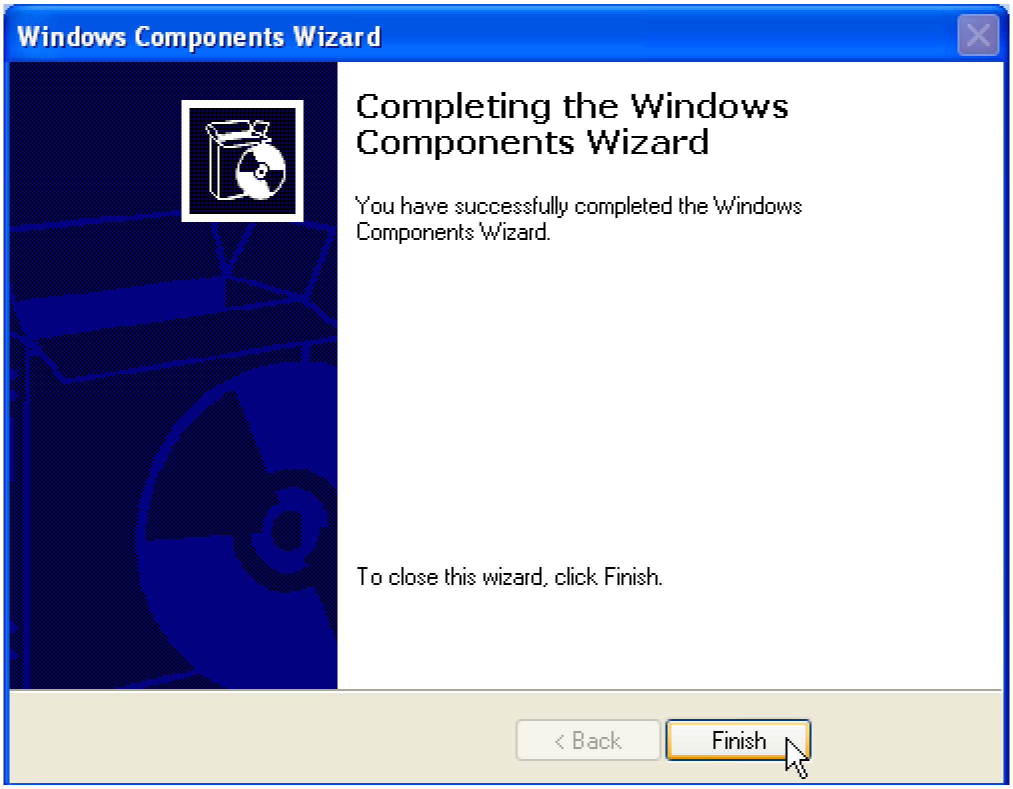
Configuring Components

Setup is making the configuration changes you requested.

Please wait while Setup configures the components. This may take several minutes, depending on the components selected.

Status: Configuring Internet Information Services...

11. Once complete click ‘Finish’



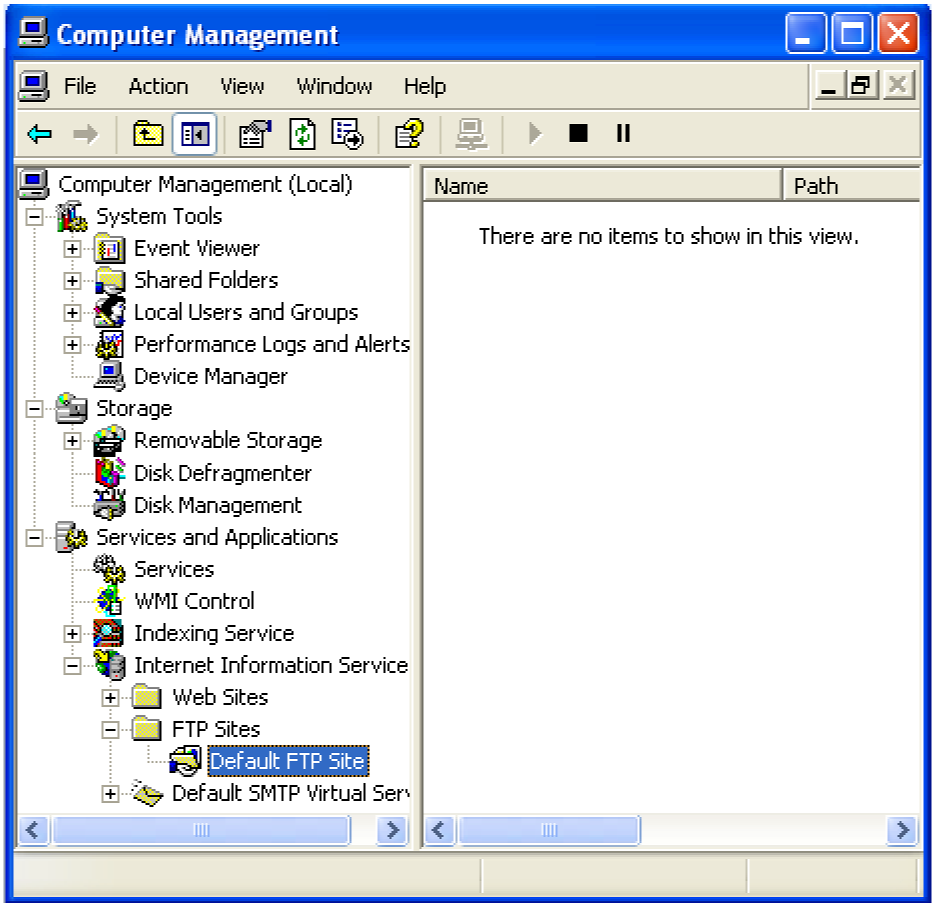
Windows Components Wizard

Completing the Windows Components Wizard

You have successfully completed the Windows Components Wizard.

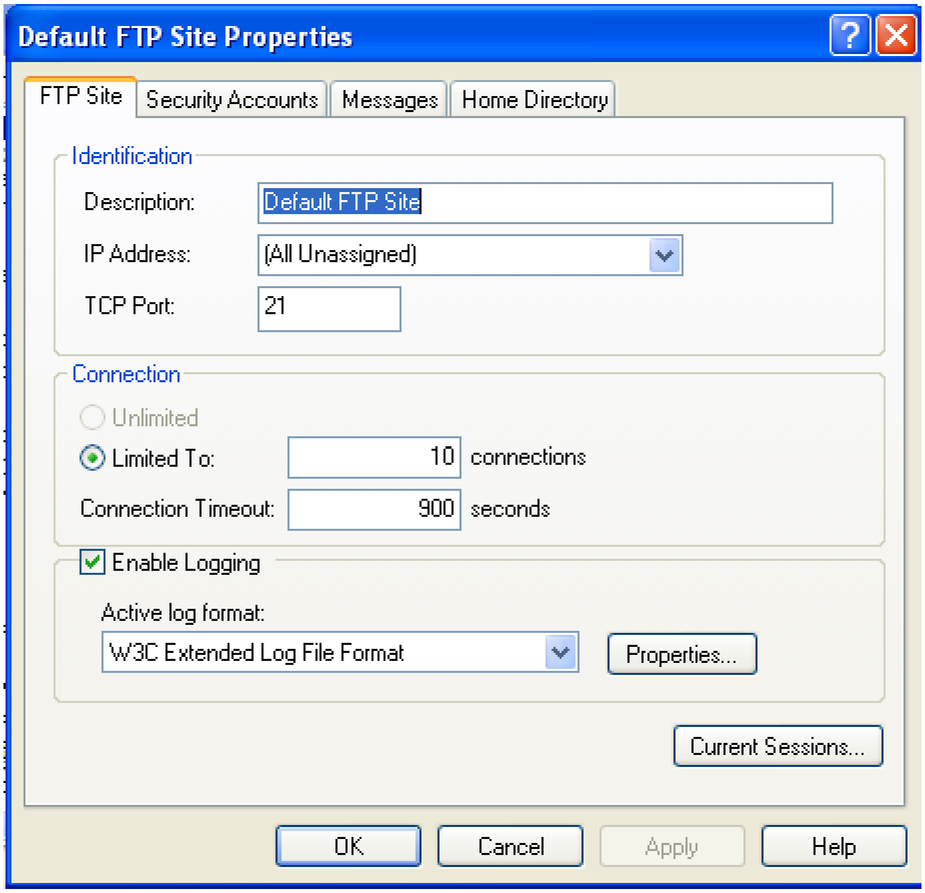
To close this wizard, click Finish.

12. From Control Panel->Administrative Tools, select Computer Management and expand the Internet Information Service folders until the "Default FTP Site" appears. This verifies that ftp is installed.



There are no items to show in this view.

13. Right click the 'Default FTP Site' and select the Properties menu item to view the current settings. Reference the PCSTATS web site, previously discussed, for suggestions on how to adjust security, add user accounts for access, etc. Note that by default only **local** computer user accounts with **Administrative privileges** my access files.



Default FTP Site Properties

Identification

Description: Default FTP Site

IP Address: (All Unassigned)

TCP Port: 21

Connection

○ Unlimited

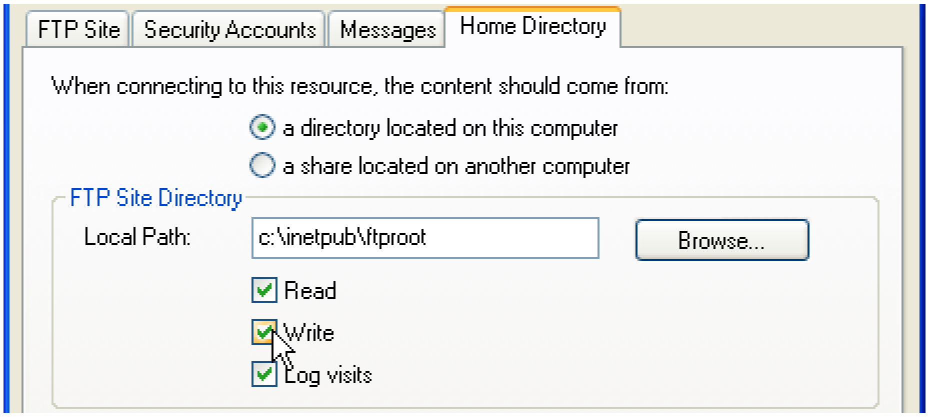
● Limited To: 10 connections

Connection Timeout: 900 seconds

☑ Enable Logging

Active log format: W3C Extended Log File Format

If both upload and download are to be done 'Write' must be enabled under the "Home Directory" tab:



When connecting to this resource, the content should come from:

● a directory located on this computer

○ a share located on another computer

FTP Site Directory

Local Path: c:\inetpub\ftproot

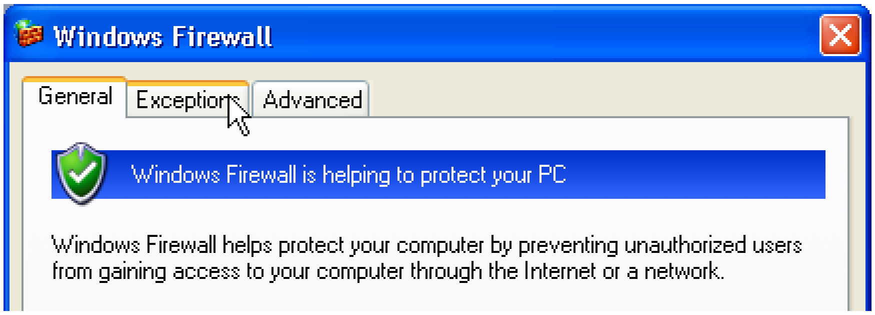
☑ Read

☑ Write

☑ Log visits

14. Windows XP has additional firewall security built into the product. The FTP Server must be specifically enabled to allow incoming connections and bypass the firewall. From the Control Panel select the Windows Firewall Icon:

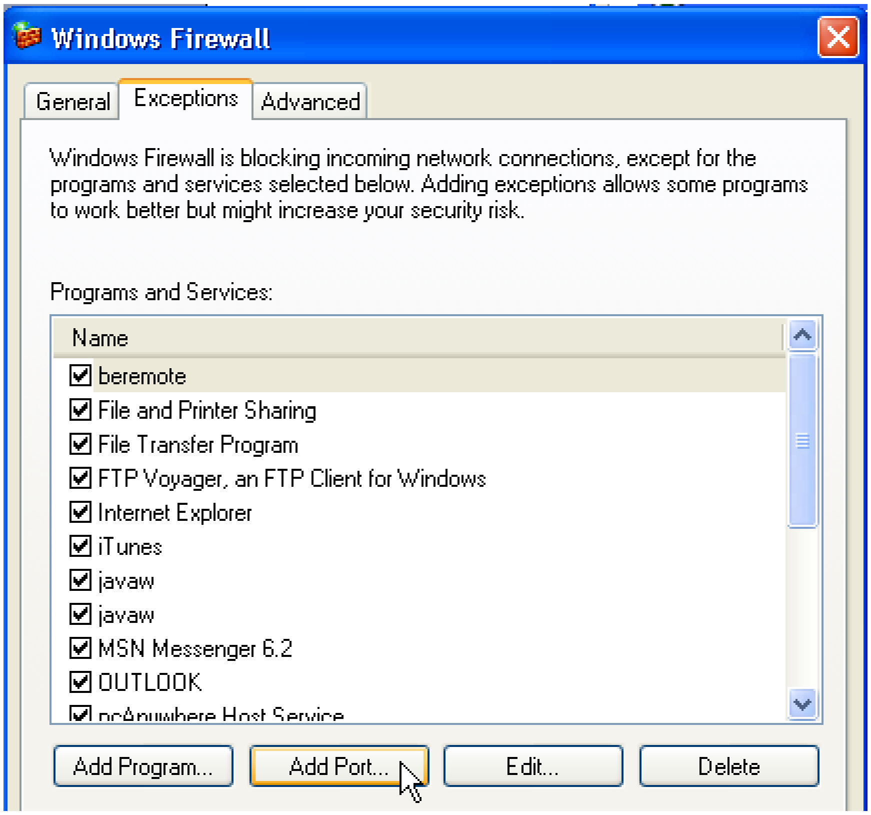
15. Select the Firewall Exceptions Tab:



Windows Firewall is helping to protect your PC

Windows Firewall helps protect your computer by preventing unauthorized users from gaining access to your computer through the Internet or a network.

16. Select Add Port:



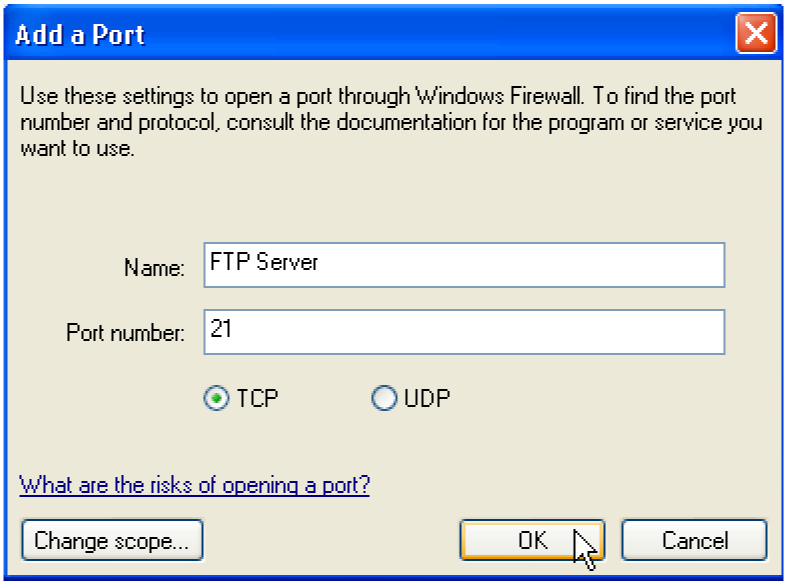
Windows Firewall is blocking incoming network connections, except for the programs and services selected below. Adding exceptions allows some programs to work better but might increase your security risk.

Programs and Services:

Name

- ☑ beremote
- ☑ File and Printer Sharing
- ☑ File Transfer Program
- ☑ FTP Voyager, an FTP Client for Windows
- ☑ Internet Explorer
- ☑ iTunes
- ☑ javaw
- ☑ javaw
- ☑ MSN Messenger 6.2
- ☑ OUTLOOK
- ☑ pcAnywhere Host Service

17. Fill out the dialog box as below and click OK:



Add a Port

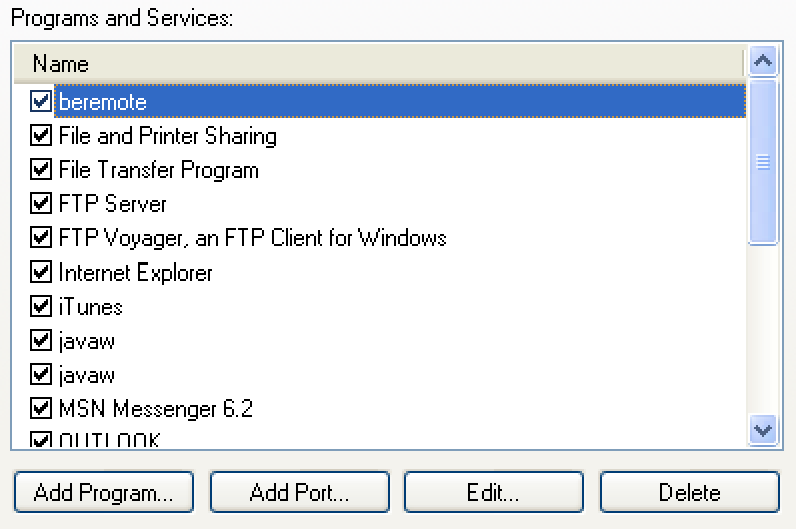
Use these settings to open a port through Windows Firewall. To find the port number and protocol, consult the documentation for the program or service you want to use.

Name: FTP Server

Port number: 21

● TCP ○ UDP

18. The FTP Server Name should appear in the list of checked ‘Programs and Services:

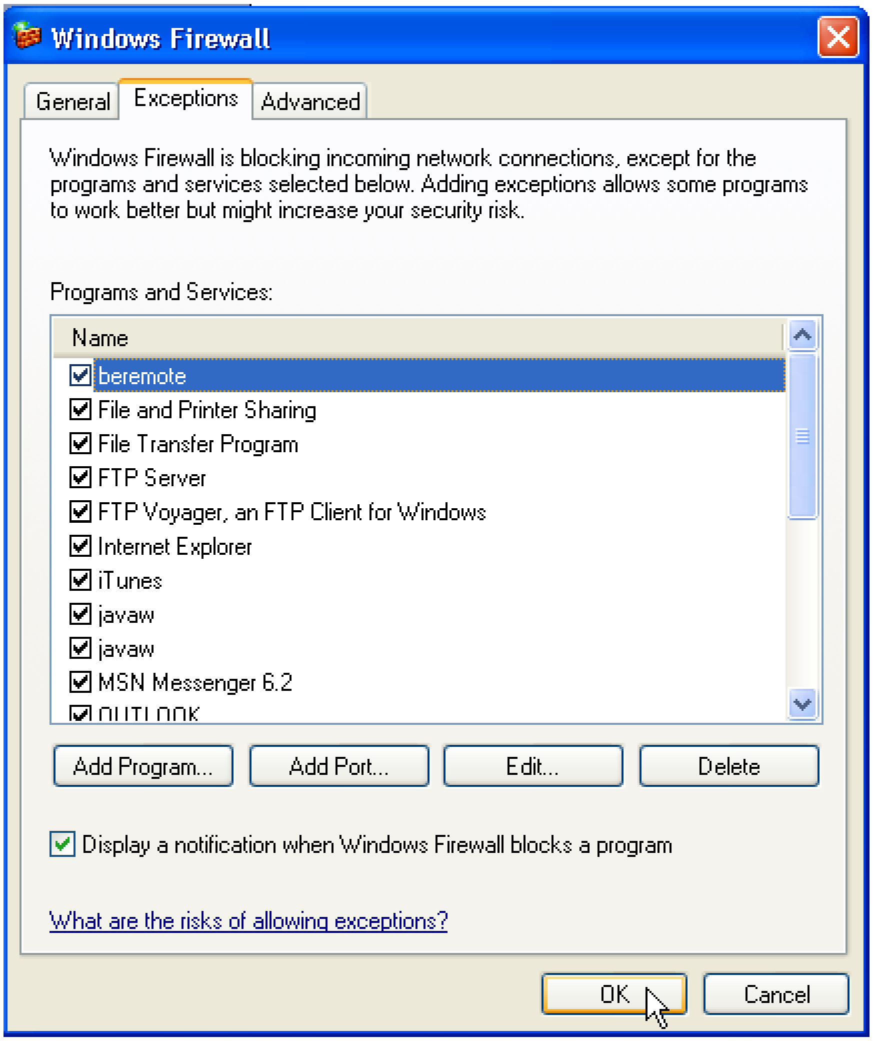


Programs and Services:

Name

☑ beremote
☑ File and Printer Sharing
☑ File Transfer Program
☑ FTP Server
☑ FTP Voyager, an FTP Client for Windows
☑ Internet Explorer
☑ iTunes
☑ javaw
☑ javaw
☑ MSN Messenger 6.2
☑ OUTLOOK

19. Click OK at the bottom of the dialog to exit:



Windows Firewall is blocking incoming network connections, except for the programs and services selected below. Adding exceptions allows some programs to work better but might increase your security risk.

Programs and Services:

Name

- ☑ beremote
- ☑ File and Printer Sharing
- ☑ File Transfer Program
- ☑ FTP Server
- ☑ FTP Voyager, an FTP Client for Windows
- ☑ Internet Explorer
- ☑ iTunes
- ☑ javaw
- ☑ javaw
- ☑ MSN Messenger 6.2
- ☑ OUTLOOK

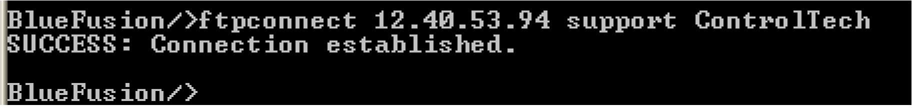
☑ Display a notification when Windows Firewall blocks a program

What are the risks of allowing exceptions?

Commands

Once a server is available, the 5200 controller provides two means to access the external server via its FTP client capabilities. The first is via the command line using Telnet, the second by the use of advanced scripting detailed in the next chapter. The following demonstrates the FTP commands when using Telnet to interact with a Windows 2003 Server:

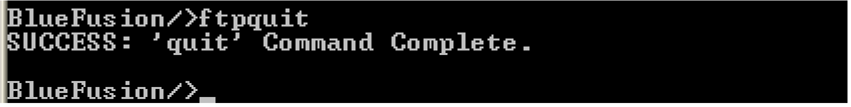
- **Ftpconnect** <ip address> <User Name> <Password>
  Provides initial connection to the remote host computer running the FTP Server.



```
BlueFusion/>ftpconnect 12.40.53.94 support ControlTech
SUCCESS: Connection established.

BlueFusion/>
```

- **Ftpquit**

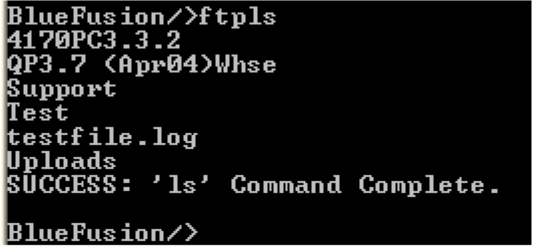
Closes an FTP session after a successful Ftpconnect.



```
BlueFusion/>ftpquit
SUCCESS: 'quit' Command Complete.

BlueFusion/>_
```

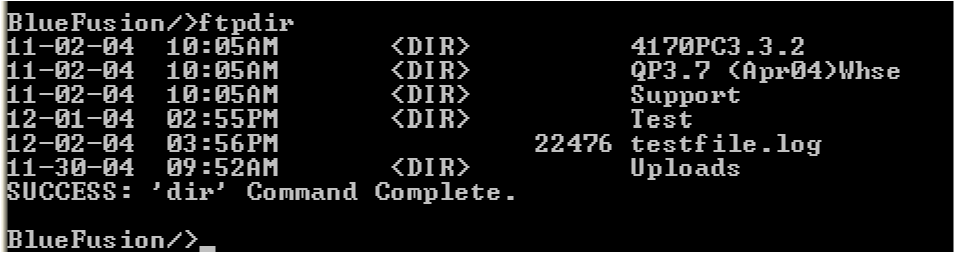
The following commands require the FTP connection to be active:

- **Ftpls** <optional path>
  Get the current directory, short format.



```
BlueFusion/>ftpls
4170PC3.3.2
QP3.7 (Apr04)Whse
Support
Test
testfile.log
Uploads
SUCCESS: 'ls' Command Complete.

BlueFusion/>
```

- **Ftpdir** <optional path>
  Get the current directory, long format.



```
BlueFusion/>ftpdir
11-02-04  10:05AM       <DIR>          4170PC3.3.2
11-02-04  10:05AM       <DIR>          QP3.7 (Apr04)Whse
11-02-04  10:05AM       <DIR>          Support
12-01-04  02:55PM       <DIR>          Test
12-02-04  03:56PM                22476 testfile.log
11-30-04  09:52AM       <DIR>          Uploads
SUCCESS: 'dir' Command Complete.

BlueFusion/>_
```

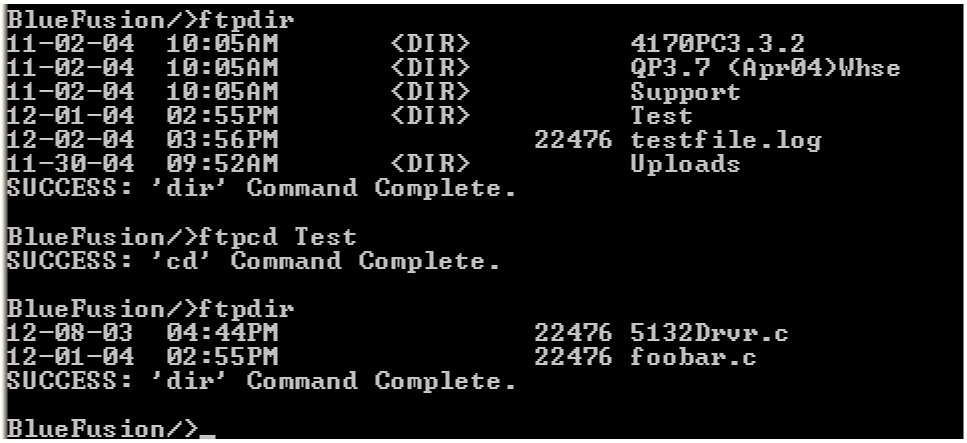
- **Ftpsend** <source path> <optional destination path/name>
  Send a file to the remote host. Paths enclosed in quotes (" ") allow the embedding of register contents using the log.ini format.

- **Ftpappend** <source path> <optional destination path/name>
  Send a file to the remote host, if it exists append to it else create a new file. Paths enclosed in quotes (" ") allow the embedding of register contents using the log.ini format.

- **Ftpget** <source path> <optional destination path/name>
  Get a file from the remote host. Firmware may be re-flashed or new programs loaded, bypassing flash storage by directing it to the root directory. Only one ftp server or client session can do this at a time since reserved SDRAM storage space is used as a temporary buffer. Example: ftpget LAN5200.sr1 /LAN5200.sr1 Paths enclosed in quotes (" ") allow the embedding of register contents using the log.ini format. Example: ftpget hostfile.fil "/mydir/Script%03dR1.ini" where the contents of R1 will be substituted into the filename.

- **Ftpcd** <path>

Change the current directory, on the host, to that specified. Paths enclosed in quotes (“ “) allow the embedding of register contents using the log.ini format.



```
BlueFusion/>ftpdir
11-02-04  10:05AM       <DIR>          4170PC3.3.2
11-02-04  10:05AM       <DIR>          QP3.7 (Apr04)Whse
11-02-04  10:05AM       <DIR>          Support
12-01-04  02:55PM       <DIR>          Test
12-02-04  03:56PM                22476 testfile.log
11-30-04  09:52AM       <DIR>          Uploads
SUCCESS: 'dir' Command Complete.

BlueFusion/>ftpcd Test
SUCCESS: 'cd' Command Complete.

BlueFusion/>ftpdir
12-08-03  04:44PM                22476 5132Drvr.c
12-01-04  02:55PM                22476 foobar.c
SUCCESS: 'dir' Command Complete.

BlueFusion/>_
```

- **Ftpmkdir** <directory name>
  Makes a directory on the host computer in the current directory. Paths enclosed in quotes (“ “) allow the embedding of register contents using the log.ini format.

- **Ftprename** <source path/file> <new name>
  Renames the specified source file to the new name. Paths enclosed in quotes (“ “) allow the embedding of register contents using the log.ini format.

- **Ftprmdir** <directory name>
  Removes the specified directory on the host computer. Paths enclosed in quotes (“ “) allow the embedding of register contents using the log.ini format.

- **Ftpdelete** <source path/file>
  Deletes a file on the host computer. Paths enclosed in quotes (“ “) allow the embedding of register contents using the log.ini format.

Telnet Error Codes

When executing ftp commands from the telnet command line the returned message will typically begin with “SUCCESS: “. At times, a failure will occur, causing an error message to appear. The message will be displayed in the Telnet session on a new line using the following format:

ERROR: <message> <code>

Where <message> is a description of the failure and <code> is detailed below:

FTP_ERROR – 0xD0
Internal ftp error.

FTP_TIMEOUT – 0xD1
Timeout occurred.

FTP_FAILED – 0xD2
General ftp failure.

FTP_NOT_CONNECTED – 0xD3
Attempted to execute a command but was not connected to the host.

FTP_NOT_DISCONNECTED – 0xD4
Host closed session during command execution.

FTP_NOT_OPEN – 0xD5
FTP failure on opening a file for transfer.

FTP_NOT_CLOSED – 0xD6
Attempted to open a file when a previous was not closed

FTP_LOGIN – 0xD9
Attempt to log onto the host failed, security violation

FTP_NOT_FOUND – 0xDA
Request was not executed (typically 550 FTP return code). Typically returned when a file/directory does not exist, or there is a security access violation blocking access.

FTP_RETURN_CODE – 0xDB
Unknown return code from host ftp server.

CHAPTER

4

Advanced Scripting

The *Model 5200 Script Language Guide (Document #951-520003)* details the operation of scripting within the 5200 controller. This section discusses more advanced features such as automated file transfers and background threaded script execution. A thorough knowledge of scripts is assumed.

Some advanced features available within scripts are:

- **Background Execution** - Scripts may not only be run as part of a step, they may also execute as separate background threads, in parallel to Quickstep execution. Writing a 1 to 999 to the *Script Execution Register* (12311) causes a script to run, within a Step, much like a 'Do' statement. An advanced feature allows a programmer to write a 1001 to 1020, executing Script001 to Script020 as a thread, much like a future 'Begin' Quickstep statement. Fully executing as an independent background thread.
- **Continuous Execution** - Scripts may execute continuously until either a fatal error or an 'end' command is executed, terminating the script.
- **Script Nesting** - Scripts may also invoke other Scripts (one call level supported).
- **Branching & Conditionals** - Scripts support branching, program labels, and 'if' conditionals.
- **Error Tracking** - Major file and communication instructions set a status error code, private to each script, which may be referenced from an 'if' conditional or 'onerror' command. The variable is referenced as ERRORCODE and allows for advanced retry and monitor operations.
- **Execution Time Control** - 'Alarm' instruction allows the script to sleep until a specific time, automatically waking. Such as every Friday afternoon at 3PM. 'Delay' instruction for pauses based on millisecond values.

Advanced Commands

Inc <Register>

Increment the contents of the reference register by 1. Typically used in counter operations such as retrying communications.

Example: inc 200

If the contents of register 200 was 99, it would become 100 after execution.

Dec <Register>

Decrement the contents of the reference register by 1. Typically used in counter operations such as retrying communications.

Example: dec 200

If the contents of register 200 was 99, it would become 98 after execution.

If <Resource> <Logic> <Resource> goto <Label>

<Resource> - Register reference (R####), decimal or hex constant (0x00000000), or ERRORCODE.

<Logic> - >, >=, <, <=, !=, ==, and & resulting in a Boolean result of true or false.

<Label> - Single line within script file containing a preceding colon ':' followed by a unique character string. Example – ':myLabel'

```
Example:     if R200 >= 55 goto tooBig
             If ERRORCODE & 0x00200000 goto fileFailure
```

:<Label>

A label consists of a single, independent line, within a script text file, to which a name is assigned, typically as a destination for a branch operation (if or goto).

Example: :myLabel

Onerror <optional error mask> goto <Label>

Set where to automatically branch should an error occur. Execution of 'Onerror' with no terms clears the branching option. Only file and communication instructions, along with syntax errors specifically set error flags. The 'optional error mask' allows you to activate individual errors. Upon return from and instruction a global ERRORCODE variable has its individual flags set should a problem occur, else ERRORCODE = 0.

```
Example:    Onerror   (clears any error branching)
            Onerror goto myLabel  (if any error occurs branch to myLabel)
            Onerror 0x00200000 goto myLabel  (branch if file error)
```

Goto <Label>

Immediately branch to the desired <Label>.

End

Exit the Script.

Delay <Register or constant – milliseconds>

Delay the designated number of milliseconds. Milliseconds may either be a decimal constant, hexadecimal notation (0x00000000), or reference the contents of a register (R200 for register 200).

```
Example:    delay 2000     (delay 2 seconds or 2000 milliseconds)
            delay R1       (delay by the number of milliseconds in register 1)
```

Alarm <TIME=HH:MM:SS> <optional day of week, DOW=Mon…>

Pause execution until the designated time occurs. A 'TIME' and optional day of week, DOW, from 0 (Monday) to 6 (Sunday) is listed. Each parameter may be contained within a register. For example the hour and/or minute could reference register 15 while the minute is a constant: R15:00. When referencing a specific day of the week enter the following: Mon, Tue, Wed, Thu, Fri, Sat, or Sun, for the desired day, or a number from 0 to 6. Seconds may also be included yielding a format of <HH:MM:SS>.

```
Example:    alarm TIME=23:00 DOW=Mon  (sleep until 11PM on Monday)
            alarm TIME=23:00    (sleep until 11PM)
            alarm TIME=R1:00    (sleep until hour contents in register 1)
            alarm TIME=23:00:05 DOW=0  (where 0 = Monday)
            alarm TIME=23:00:05 DOW=R10  (where contents R10, 0 - 6)
```

ERRORCODE

ERRORCODE is a universal variable, private to each script. Upon execution of script commands status bits are set should problems occur. 'if' conditional and 'onerror' commands can branch accordingly.

SCRIPT_ERRMASK_FATAL - 0x00800000
Fatal error, such as a memory allocation failure. Currently this can only be returned by an 'ftpconnect' command.

SCRIPT_ERRMASK_SYNTAX - 0x00400000
Syntax error, bad parameter passed as part of a command.

SCRIPT_ERRMASK_FILE - 0x00200000
General File failure. For example ‘rename’ of file failed, creation of a new directory ‘mkdir’. Effected commands are:

- mkdir
- rename
- cd
- rmdir
- format flash
- delete
- load symbols
- get symbols
- mount
- umount

SCRIPT_ERRMASK_FTP_CONNECT - 0x00100000
Returned if an ‘ftpconnect’ attempt fails or there is an attempt to use a command that requires a previous connection and not exists.

SCRIPT_ERRMASK_FTP_LOSTCONNECT - 0x00080000
Lost connection while attempting an FTP command

- Ftpls
- Ftpdir
- Ftpmkdir
- Ftprmdir
- Ftpcd
- Ftpget
- Ftpsend
- Ftpappend
- Ftpdelete

SCRIPT_ERRMASK_FTP_COMMAND - 0x00040000
Unknown FTP response code returned

- Ftpls
- Ftpdir
- Ftpmkdir
- Ftprmdir
- Ftpcd
- Ftpget
- Ftpsend
- Ftpappend
- Ftpdelete

SCRIPT_ERRMASK_FTP_NOTFOUND - 0x00020000
Operation failed, file or directory not exist on host

- Ftpls
- Ftpdir
- Ftpmkdir
- Ftprmdir
- Ftpcd
- Ftpget
- Ftpsend
- Ftpappend
- Ftpdelete

SCRIPT_ERRMASK_FTP_SECURITY - 0x00010000
User name or Password failed. This only applies to 'ftpconnect'.

Script Example

The following sample script, Script001, shows how to connect to an ftp server, send a file called Log001.log and download a file called anyfile.log on the host to a file called newfile.log on the controller. A maximum of 3 retries, with a 5 second interval will be attempted before aborting. Register 2 is used as an arbitrary status register, which can be monitored by Quickstep, and should be set to 0 prior to invoking the script. Its status is as follows:

1 = Complete and successful
-1 = Busy running script
-2 = General failure
-3 = File did not exist on the Controller
-4 = File did not exist on the Host

The execution of the following script would occur by writing a 1001 to the *Script Execution Register* (12311). **Adding a 1000 to the script file number causes it to execute as a background thread. Ftp Client operations should only be run from a command line within telnet or as a background thread, not as part of a Quickstep task (001 – 999).**

Script001.ini:

```
# This script tests the automated transfer of data from
# the controller to a remote ftp server.
# Register 1 is used as a retry counter
# Register 2 is used notify Quickstep program of completion status
# Writing to the Script Execution Register (12311) will cause
# execution to occur.  Script file name is currently Script001.ini
# thus writing 1001 will cause this file to run as a thread in the
# background, writing a 1 will cause it to run within a Quickstep step.
# Threaded execution (background) is the preferred method
```

```
#
:start
# Clear the retry counter
        1 = 0
# Clear our completion flag register
        2 = -1
        goto firstTime

:retry
# Delay for 5 seconds and then try again
        delay 5000
# Bump the retry counter and see if done with retries...
        inc R1
        if R1 > 3 goto abort

:firstTime
# First setup where to branch if an error should occur
        onerror goto errorOccurred

# Lets connect to the remote host
        ftpconnect 12.40.53.94 support ControlTech

# Now lets upload a file and then download a file
# If an error occurs we will exit automatically due to the 'onerror' command
# SEND to host from controller
        ftpsend /_system/Messages/data/Log001.log
# RECIEVE from host and store to controller under different name
        ftpget anyfile.log /_system/Messages/data/newfile.log
# All done so close host session gracefully
        ftpquit
# Set completion flag to 0 indicating we are done and successful
        2 = 1

# If we get here we are all done
        end

# Process error if should occur
:errorOccurred

# Lets see what type of error occurred
# First check to see if initial connection failed
        if ERRORCODE & 0x00100000 goto retry
# Next see if we failed during a transfer
        if ERRORCODE & 0x00080000 goto retry
# Next see if we failed because the file did not exist on the controller
        if ERRORCODE & 0x00200000 goto nofileController
# Next see if we failed because the file did not exist on the host
        if ERRORCODE & 0x00020000 goto nofileHost
# Was a fatal failure so give up
:abort
        2 = -2
        end
# File did not exist on the controller
:nofileController
        2 = -3
        end
```

```
# File did not exist on the host
:nofileHost
        2 = -4
        end
```

Appendix A

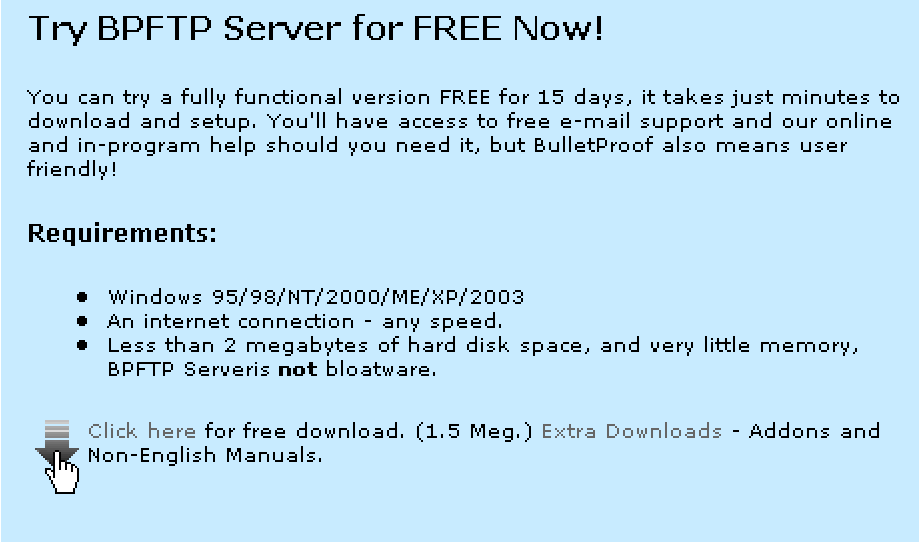
BulletProof FTP Server

BulletProof Software has available a low cost FTP Server ($34.95, 15 day free trial) which can be installed on Windows 2000 and XP systems for communications with the 5222. This appendix is provided as an initial quick start guide detailing its installation and initial setup. Detailed information, and additional support, is provided within their manual. The program may be downloaded from their web site at:

http://www.bpftpserver.com/download.php.

Installation

1. From their web site click the download Icon and save the file to a desired directory:



Try BPFTP Server for FREE Now!

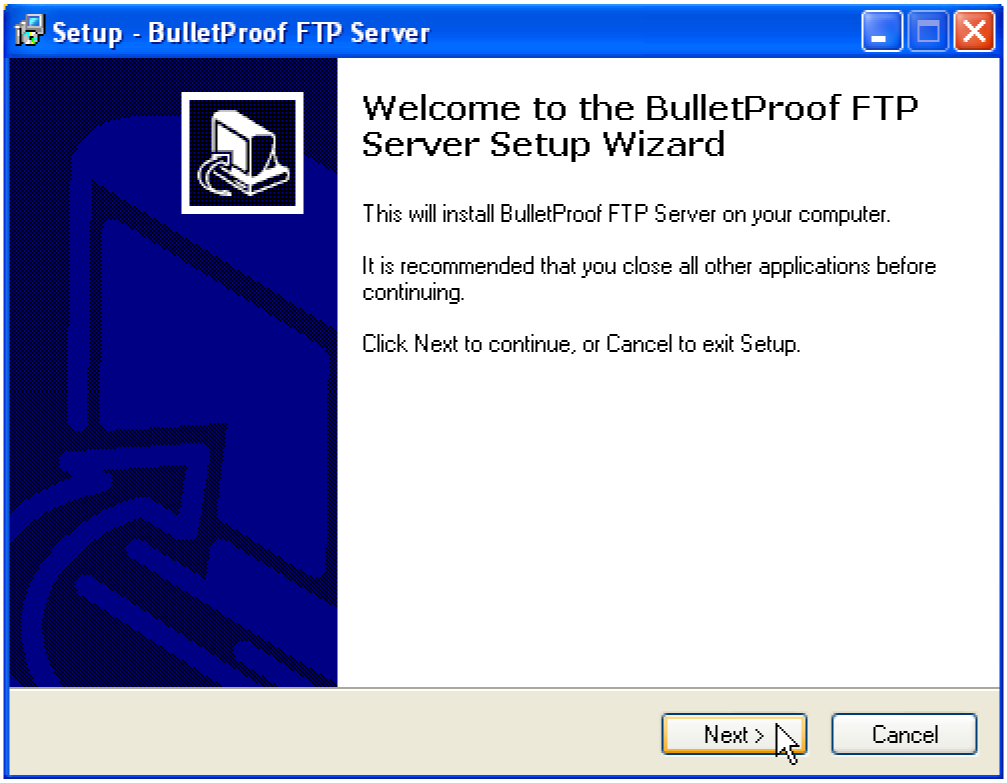
You can try a fully functional version FREE for 15 days, it takes just minutes to download and setup. You'll have access to free e-mail support and our online and in-program help should you need it, but BulletProof also means user friendly!

Requirements:

- Windows 95/98/NT/2000/ME/XP/2003
- An internet connection - any speed.
- Less than 2 megabytes of hard disk space, and very little memory, BPFTP Serveris **not** bloatware.

Click here for free download. (1.5 Meg.) Extra Downloads - Addons and Non-English Manuals.

2. Once downloaded execute their "ftpsetup.exe" file. At the welcome screen click the Next button:



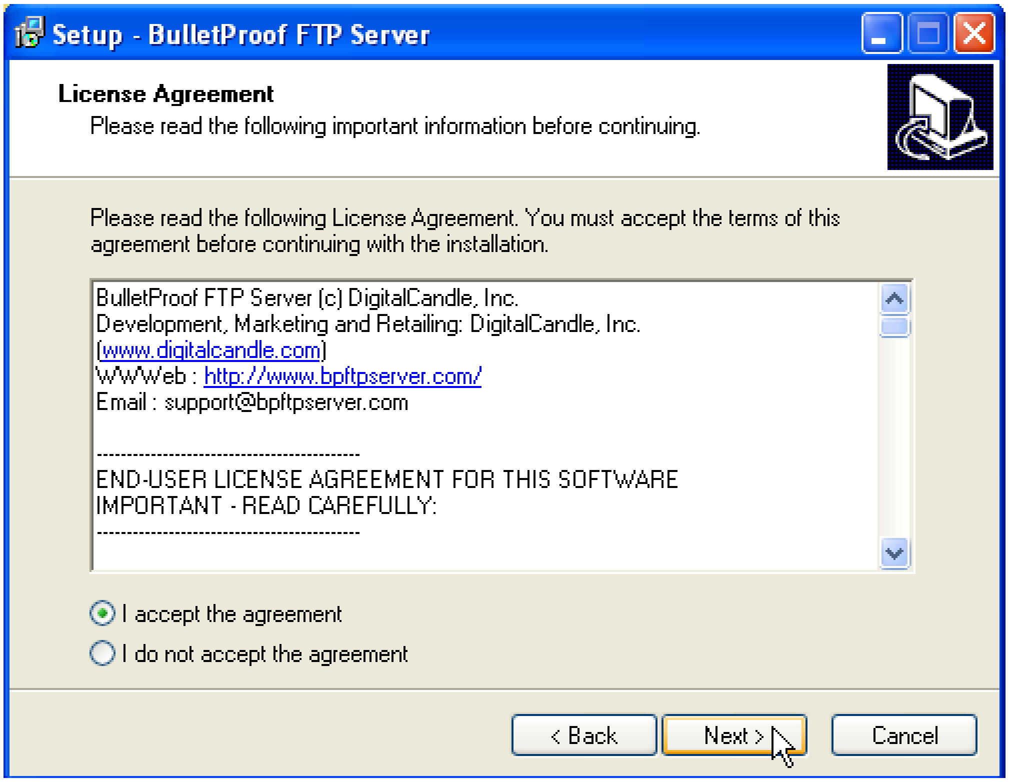
Welcome to the BulletProof FTP Server Setup Wizard

This will install BulletProof FTP Server on your computer.

It is recommended that you close all other applications before continuing.

Click Next to continue, or Cancel to exit Setup.

3. Accept their agreement and click Next:



License Agreement

Please read the following important information before continuing.

Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

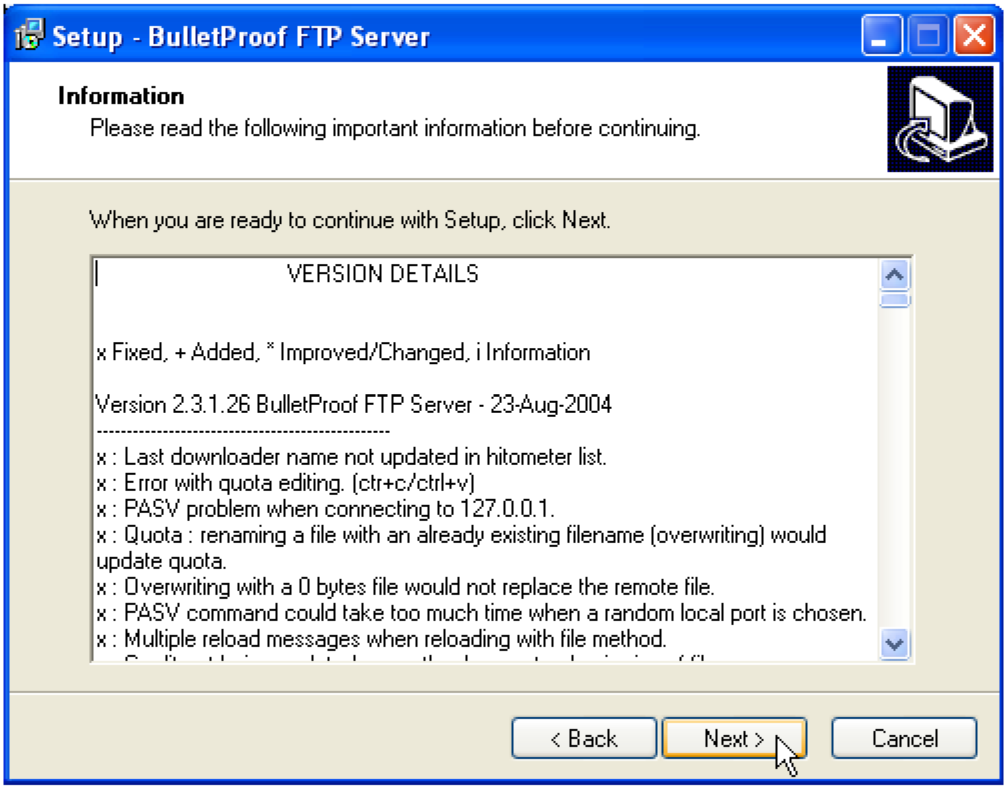
BulletProof FTP Server (c) DigitalCandle, Inc.
Development, Marketing and Retailing: DigitalCandle, Inc.
(www.digitalcandle.com)
WWWeb : http://www.bpftpserver.com/
Email : support@bpftpserver.com

END-USER LICENSE AGREEMENT FOR THIS SOFTWARE
IMPORTANT - READ CAREFULLY:

● I accept the agreement

○ I do not accept the agreement

4. Click Next at the Information screen:



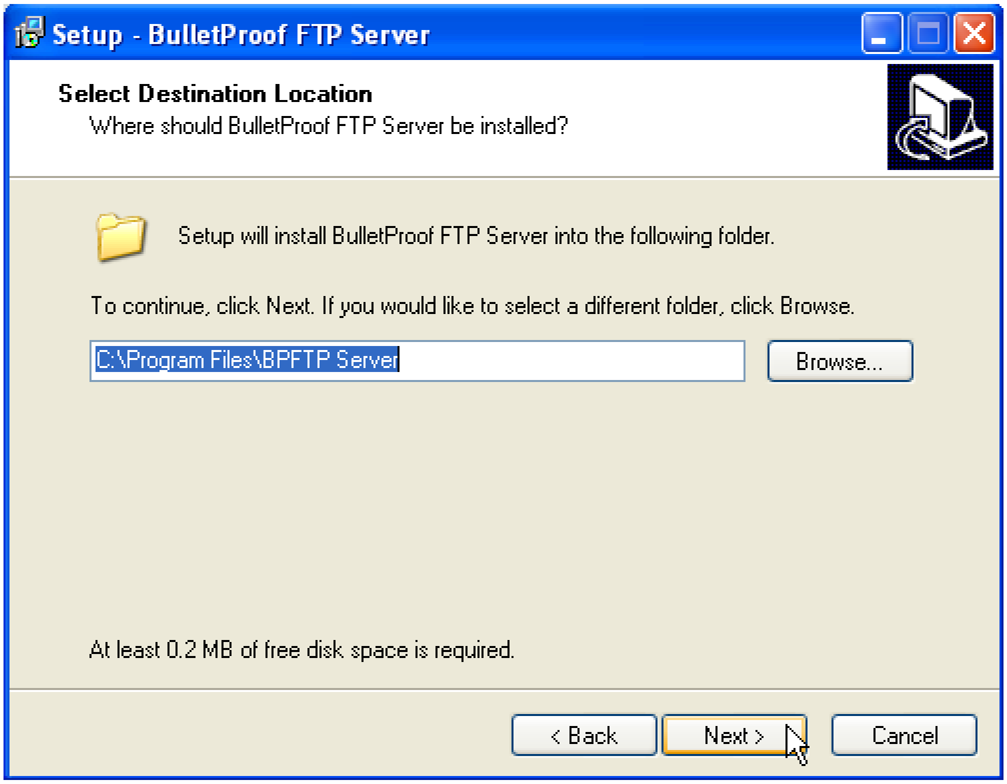
Information

Please read the following important information before continuing.

When you are ready to continue with Setup, click Next.

```
                    VERSION DETAILS


x Fixed, + Added, * Improved/Changed, i Information

Version 2.3.1.26 BulletProof FTP Server - 23-Aug-2004
--------------------------------------------------
x : Last downloader name not updated in hitometer list.
x : Error with quota editing. (ctr+c/ctrl+v)
x : PASV problem when connecting to 127.0.0.1.
x : Quota : renaming a file with an already existing filename (overwriting) would
update quota.
x : Overwriting with a 0 bytes file would not replace the remote file.
x : PASV command could take too much time when a random local port is chosen.
x : Multiple reload messages when reloading with file method.
```

5. Change the installation directory if required and click Next:



Select Destination Location

Where should BulletProof FTP Server be installed?

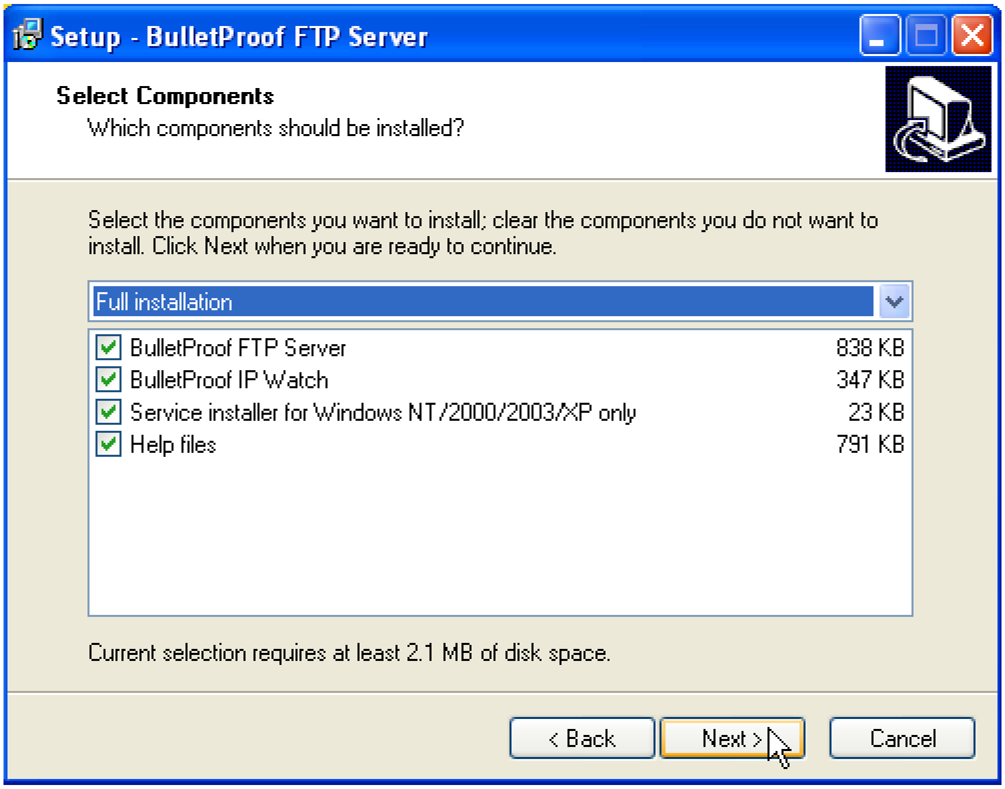
Setup will install BulletProof FTP Server into the following folder.

To continue, click Next. If you would like to select a different folder, click Browse.

C:\Program Files\BPFTP Server

At least 0.2 MB of free disk space is required.

6. Accept the defaults of the next few screens and continue as shown:



Select Components

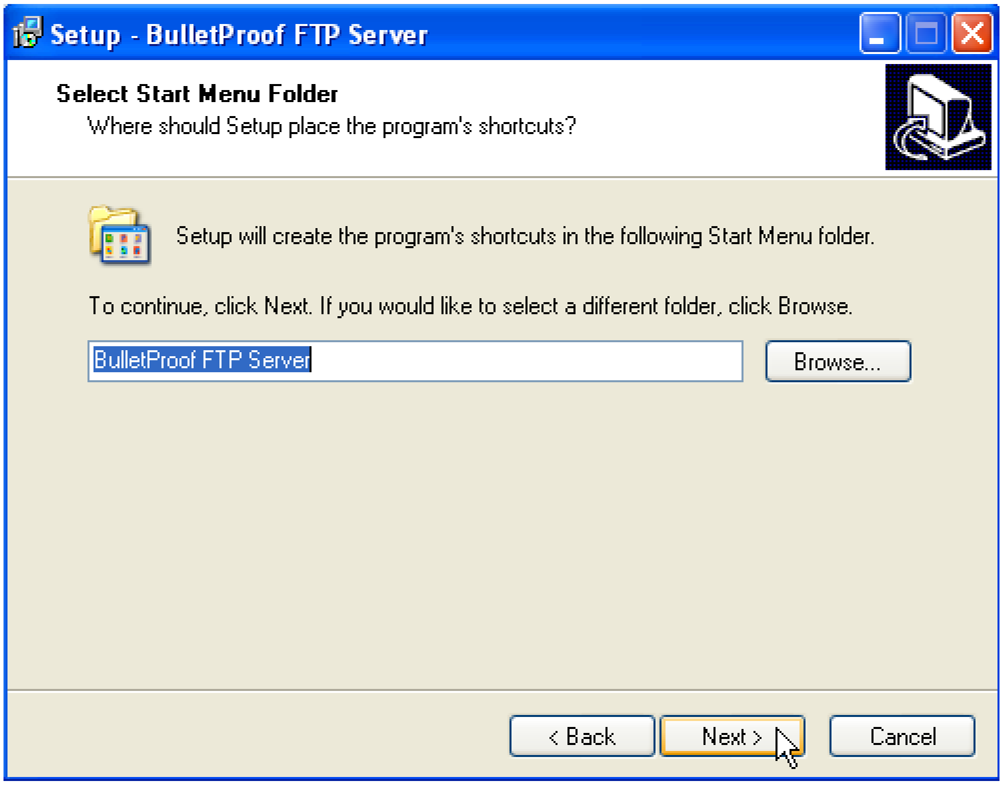
Which components should be installed?

Select the components you want to install; clear the components you do not want to install. Click Next when you are ready to continue.

Full installation

☑ BulletProof FTP Server 838 KB
☑ BulletProof IP Watch 347 KB
☑ Service installer for Windows NT/2000/2003/XP only 23 KB
☑ Help files 791 KB

Current selection requires at least 2.1 MB of disk space.



Setup - BulletProof FTP Server

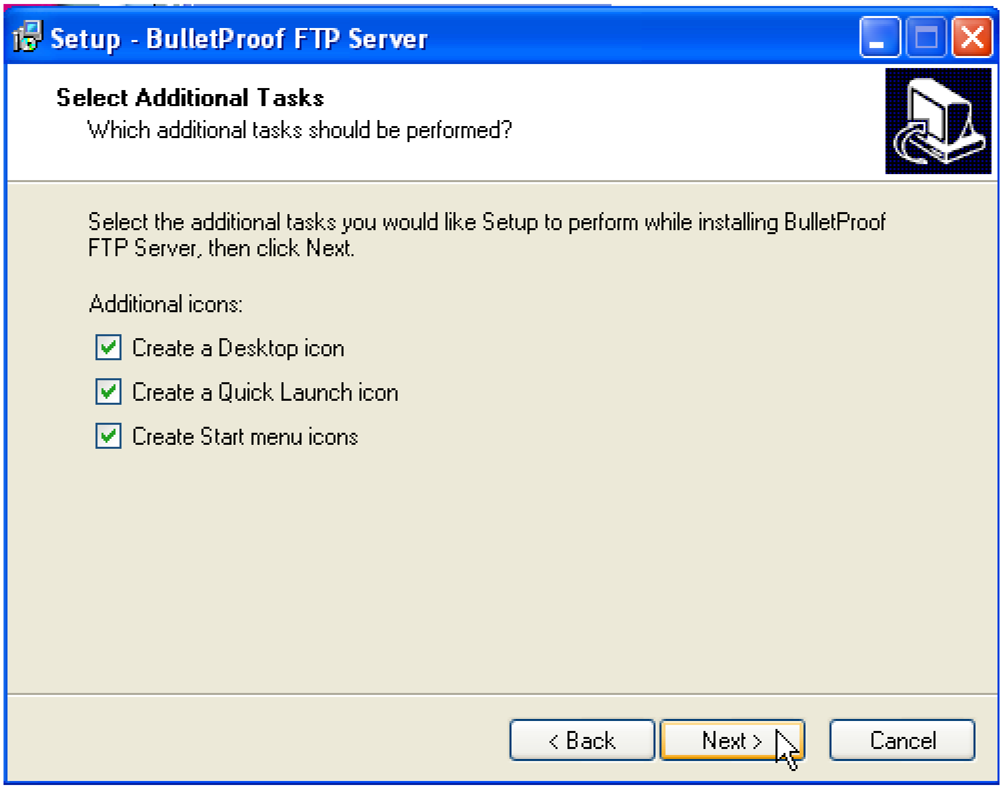
Select Start Menu Folder

Where should Setup place the program's shortcuts?

Setup will create the program's shortcuts in the following Start Menu folder.

To continue, click Next. If you would like to select a different folder, click Browse.

BulletProof FTP Server



Setup - BulletProof FTP Server

Select Additional Tasks

Which additional tasks should be performed?

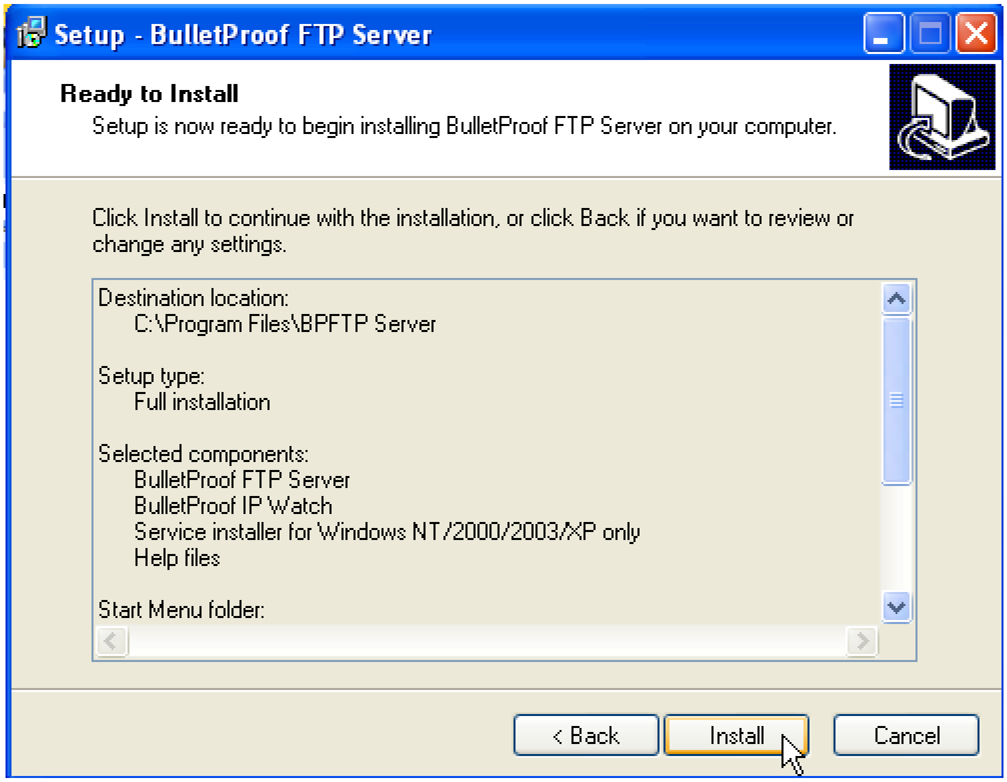
Select the additional tasks you would like Setup to perform while installing BulletProof FTP Server, then click Next.

Additional icons:

☑ Create a Desktop icon

☑ Create a Quick Launch icon

☑ Create Start menu icons



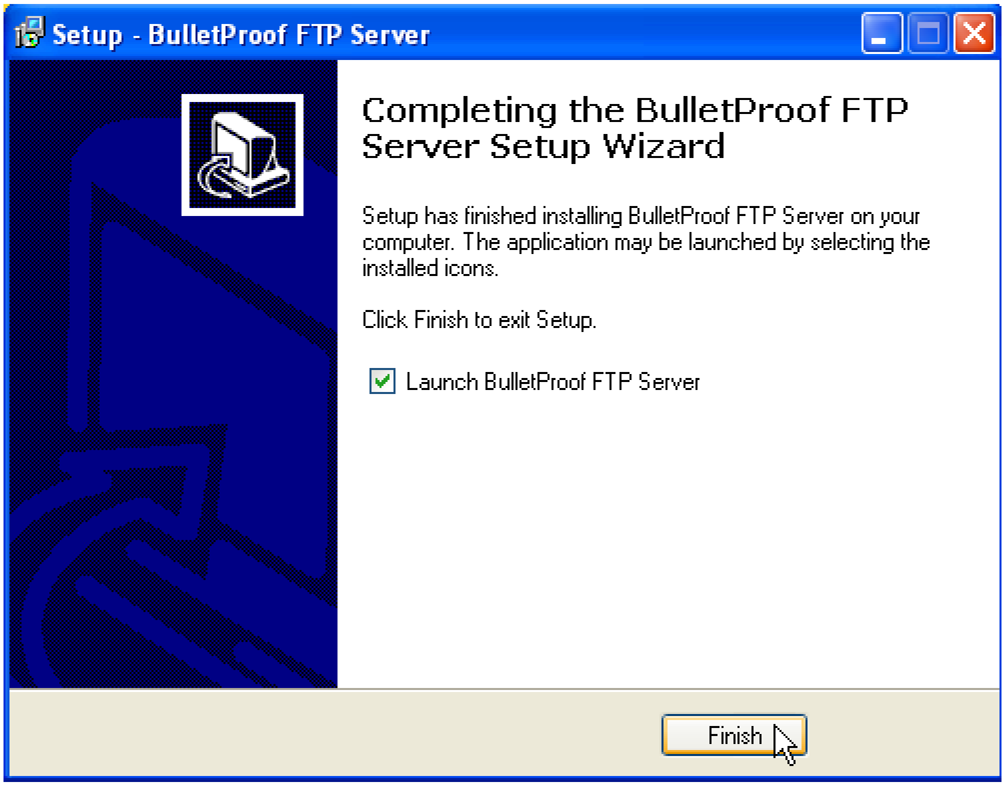
Ready to Install

Setup is now ready to begin installing BulletProof FTP Server on your computer.

Click Install to continue with the installation, or click Back if you want to review or change any settings.

```
Destination location:
      C:\Program Files\BPFTP Server

Setup type:
      Full installation

Selected components:
      BulletProof FTP Server
      BulletProof IP Watch
      Service installer for Windows NT/2000/2003/XP only
      Help files

Start Menu folder:
```

7. Upon completion the final screen will appear, click Finish and the program will automatically be invoked:



Completing the BulletProof FTP Server Setup Wizard

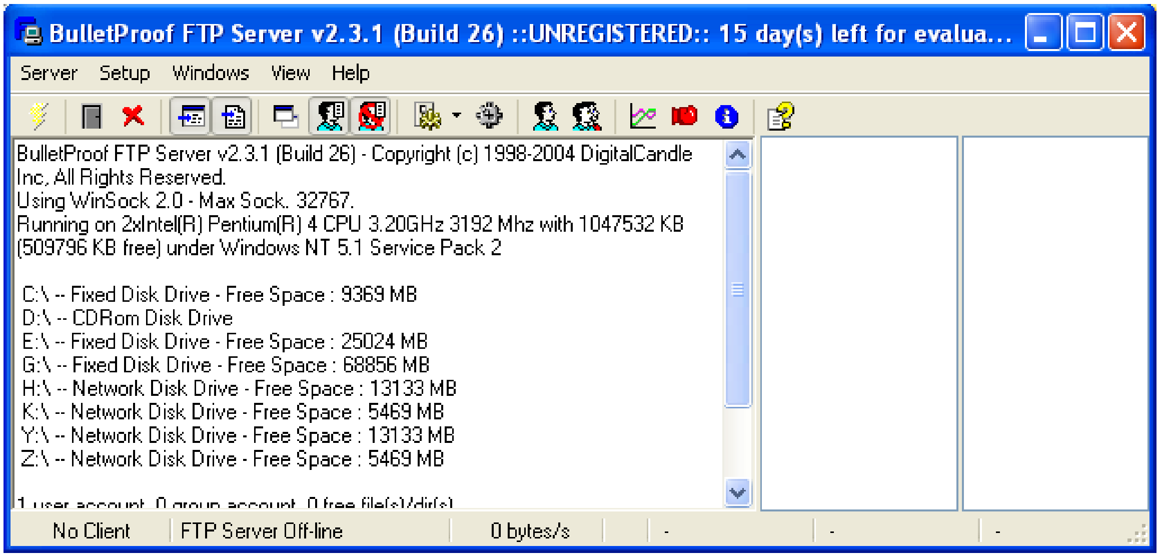
Setup has finished installing BulletProof FTP Server on your computer. The application may be launched by selecting the installed icons.

Click Finish to exit Setup.

☑ Launch BulletProof FTP Server

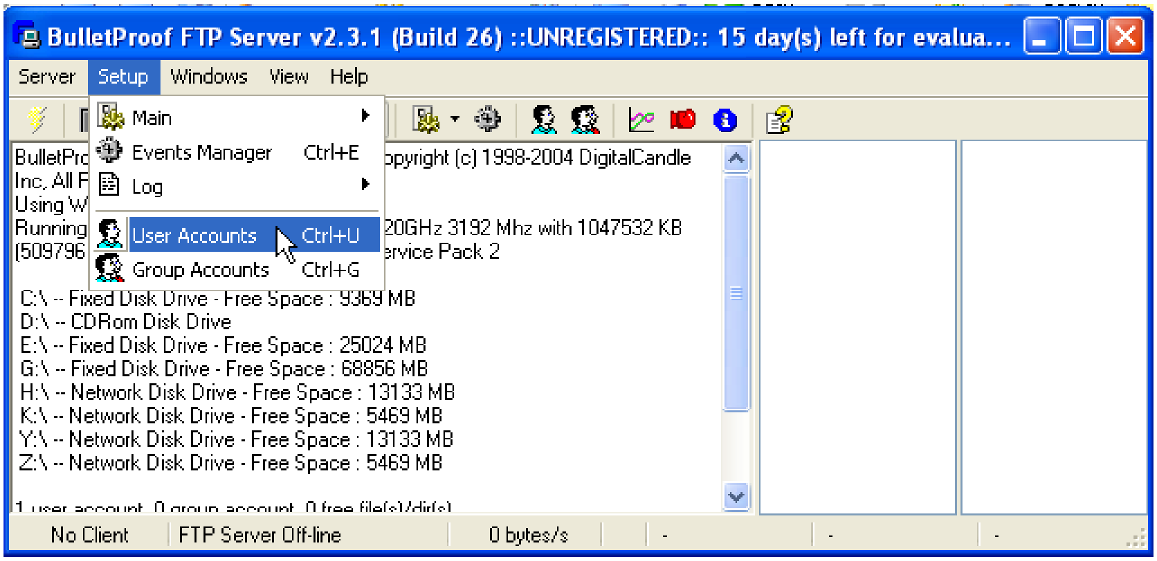
Operation

Upon initial install and execution the following will appear:


BulletProof FTP Server v2.3.1 (Build 26) ::UNREGISTERED:: 15 day(s) left for evalua...
Server
Setup
Windows
View
Help
No Client
FTP Server Off-line
0 bytes/s
-
-
-

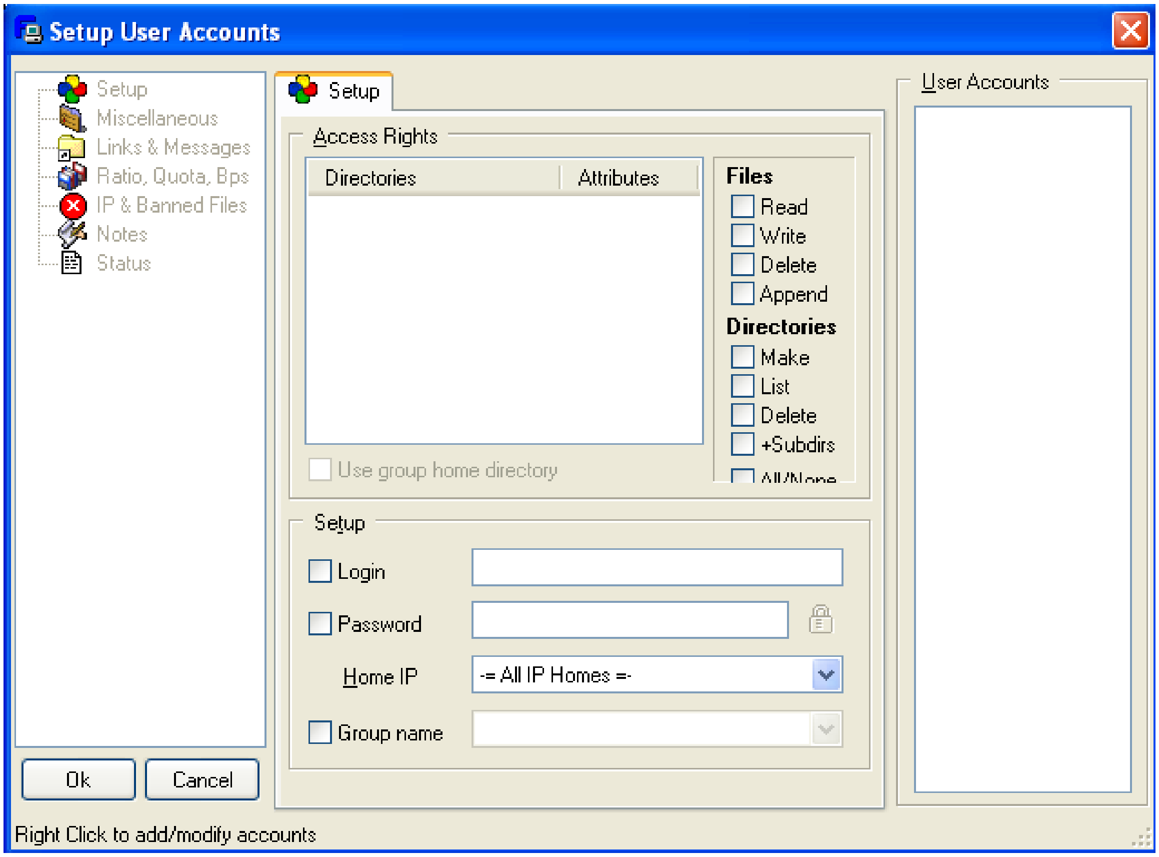
```
BulletProof FTP Server v2.3.1 (Build 26) - Copyright (c) 1998-2004 DigitalCandle
Inc, All Rights Reserved.
Using WinSock 2.0 - Max Sock. 32767.
Running on 2xIntel(R) Pentium(R) 4 CPU 3.20GHz 3192 Mhz with 1047532 KB
(509796 KB free) under Windows NT 5.1 Service Pack 2

 C:\ -- Fixed Disk Drive - Free Space : 9369 MB
 D:\ -- CDRom Disk Drive
 E:\ -- Fixed Disk Drive - Free Space : 25024 MB
 G:\ -- Fixed Disk Drive - Free Space : 68856 MB
 H:\ -- Network Disk Drive - Free Space : 13133 MB
 K:\ -- Network Disk Drive - Free Space : 5469 MB
 Y:\ -- Network Disk Drive - Free Space : 13133 MB
 Z:\ -- Network Disk Drive - Free Space : 5469 MB

1 user account, 0 group account, 0 free file(s)/dir(s)
```

An initial user account must be added. This is the ‘user name’ and ‘password’, along with access rights you will grant this person and/or controller. Click the “Setup->User Accounts” menu item:

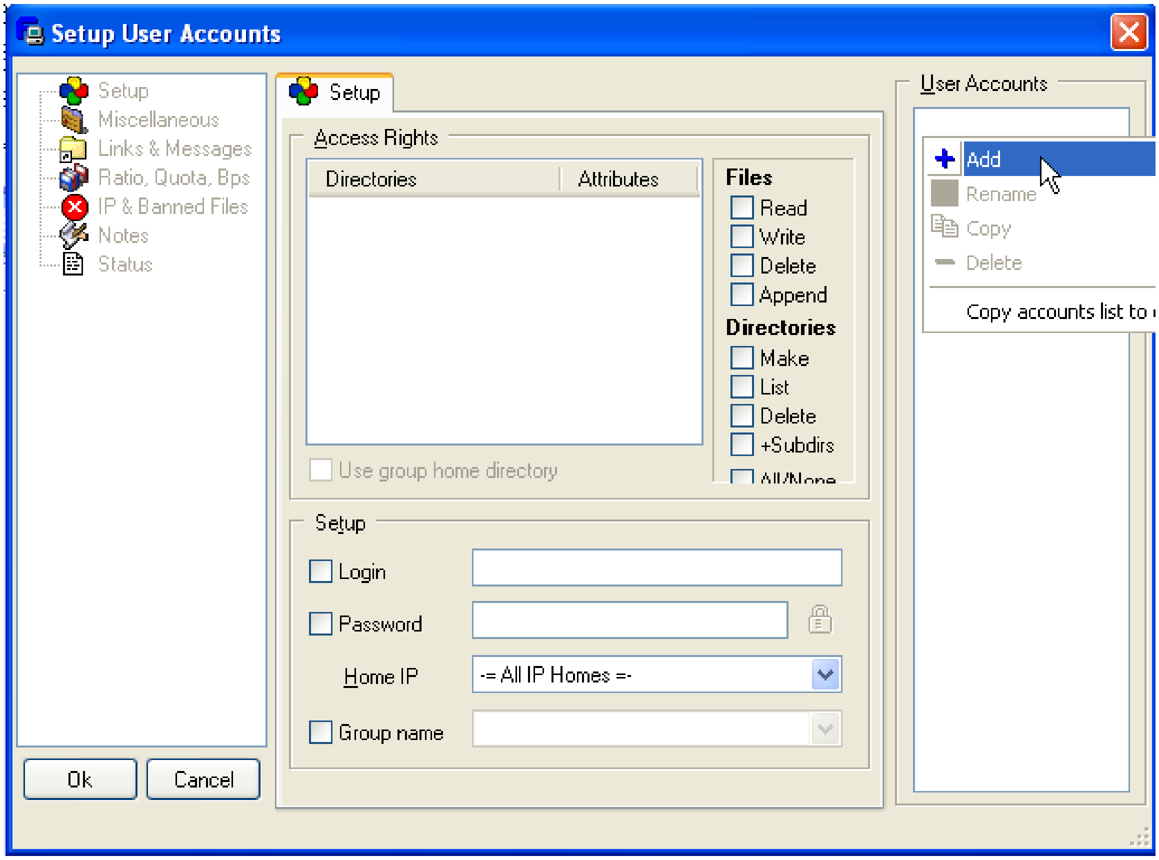


```
BulletPro                 opyright (c) 1998-2004 DigitalCandle
Inc, All F
Using W
Running                   20GHz 3192 Mhz with 1047532 KB
(509796                   ervice Pack 2
 C:\ -- Fixed Disk Drive - Free Space : 9369 MB
 D:\ -- CDRom Disk Drive
 E:\ -- Fixed Disk Drive - Free Space : 25024 MB
 G:\ -- Fixed Disk Drive - Free Space : 68856 MB
 H:\ -- Network Disk Drive - Free Space : 13133 MB
 K:\ -- Network Disk Drive - Free Space : 5469 MB
 Y:\ -- Network Disk Drive - Free Space : 13133 MB
 Z:\ -- Network Disk Drive - Free Space : 5469 MB
1 user account, 0 group account, 0 free file(s)/dir(s)
```

The following User Account screen will appear:

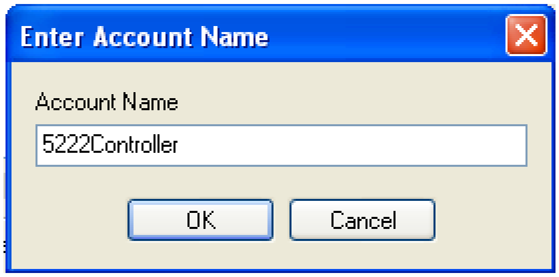


To add an account, position the mouse over the "User Accounts" window and right click the mouse, a menu will appear, select "Add":

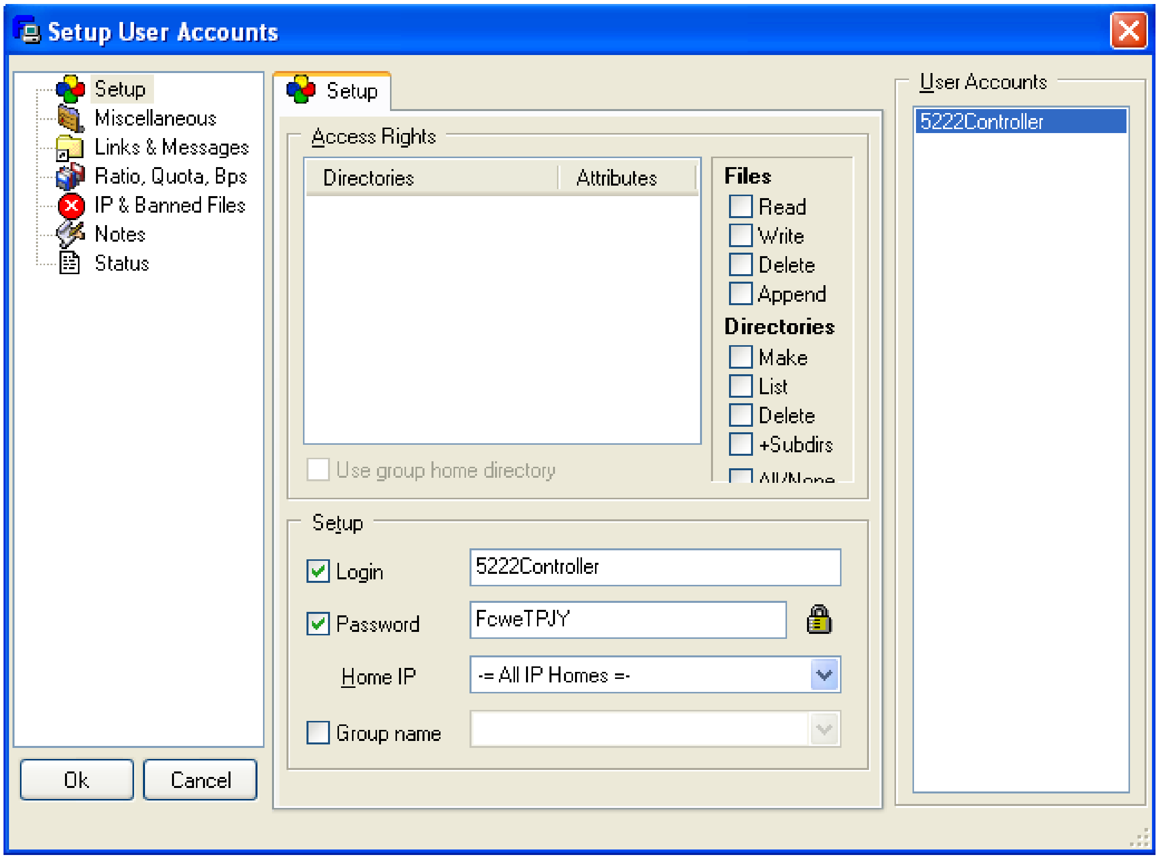


Home IP: -= All IP Homes =-

Enter the new account name, “5222Controller” is shown in the example, click OK:



Account Name: 5222Controller



Files: □ Read □ Write □ Delete □ Append

Directories: □ Make □ List □ Delete □ +Subdirs □ All/None

☑ Login: 5222Controller

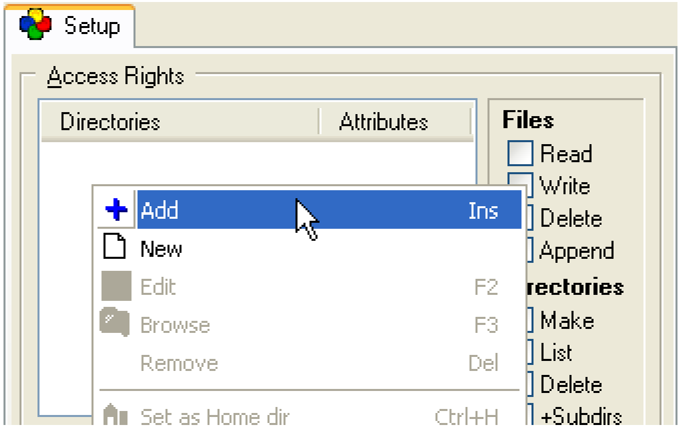
☑ Password: FcweTPJY

Home IP: -= All IP Homes =-

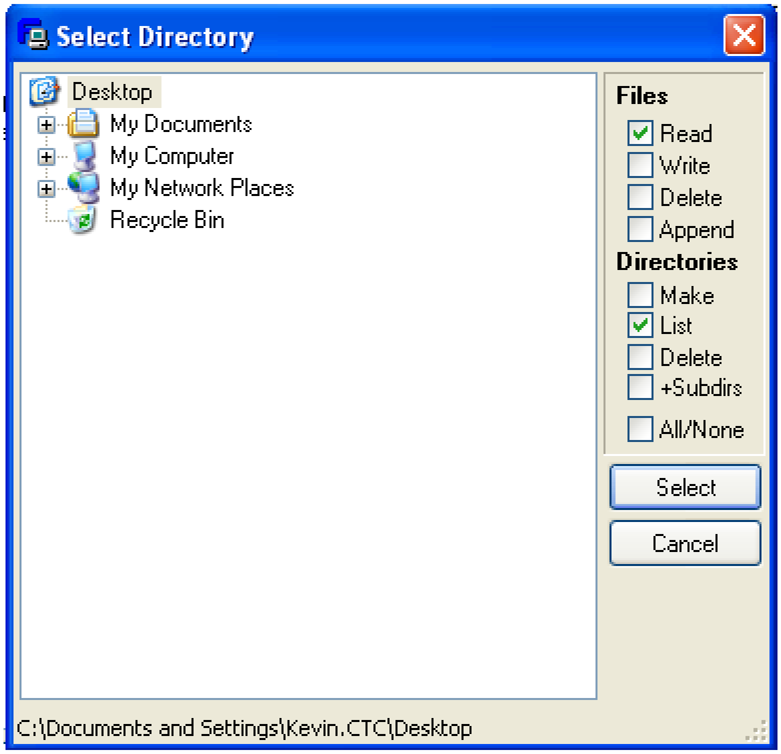
□ Group name:

User Accounts: 5222Controller

The account has now been created. A default password of “FcweTPJY” is shown. This should be changed to anything desired. By default no directory or file access is granted, only the account created. In order to grant access a directory must be referenced and access privileges granted. To add a directory right click the mouse in the Access Rights area of the screen and select “Add”:



A window will appear allowing you to select the desired directory and access permissions:



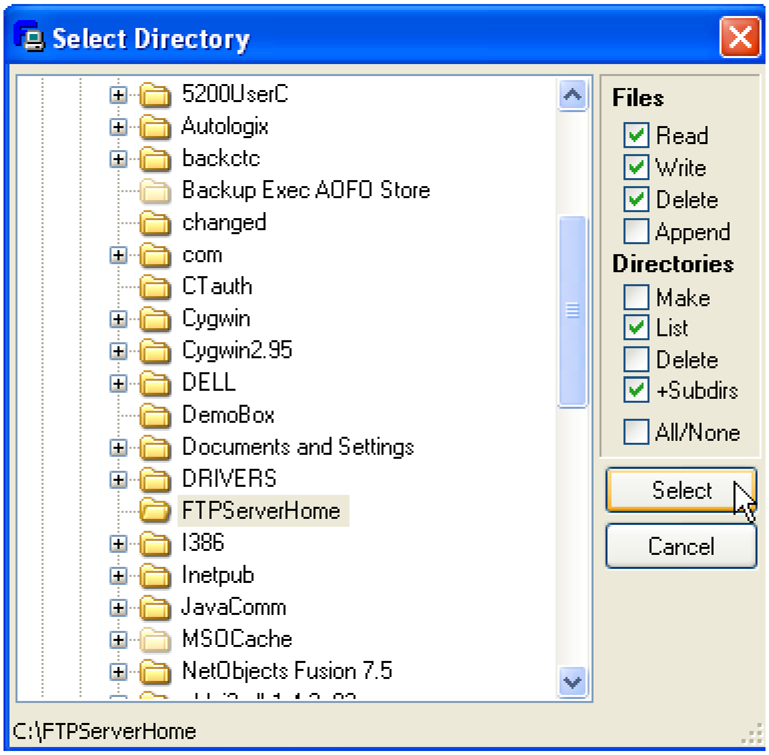
Files
☑ Read
□ Write
□ Delete
□ Append

Directories
□ Make
☑ List
□ Delete
□ +Subdirs

□ All/None

C:\Documents and Settings\Kevin.CTC\Desktop

If subdirectories are to be allowed be sure to select the +Subdirs check box. Below shows a directory on the C: drive being added called FTPServerHome. File access will allow upload and download as well as access to subdirectories.

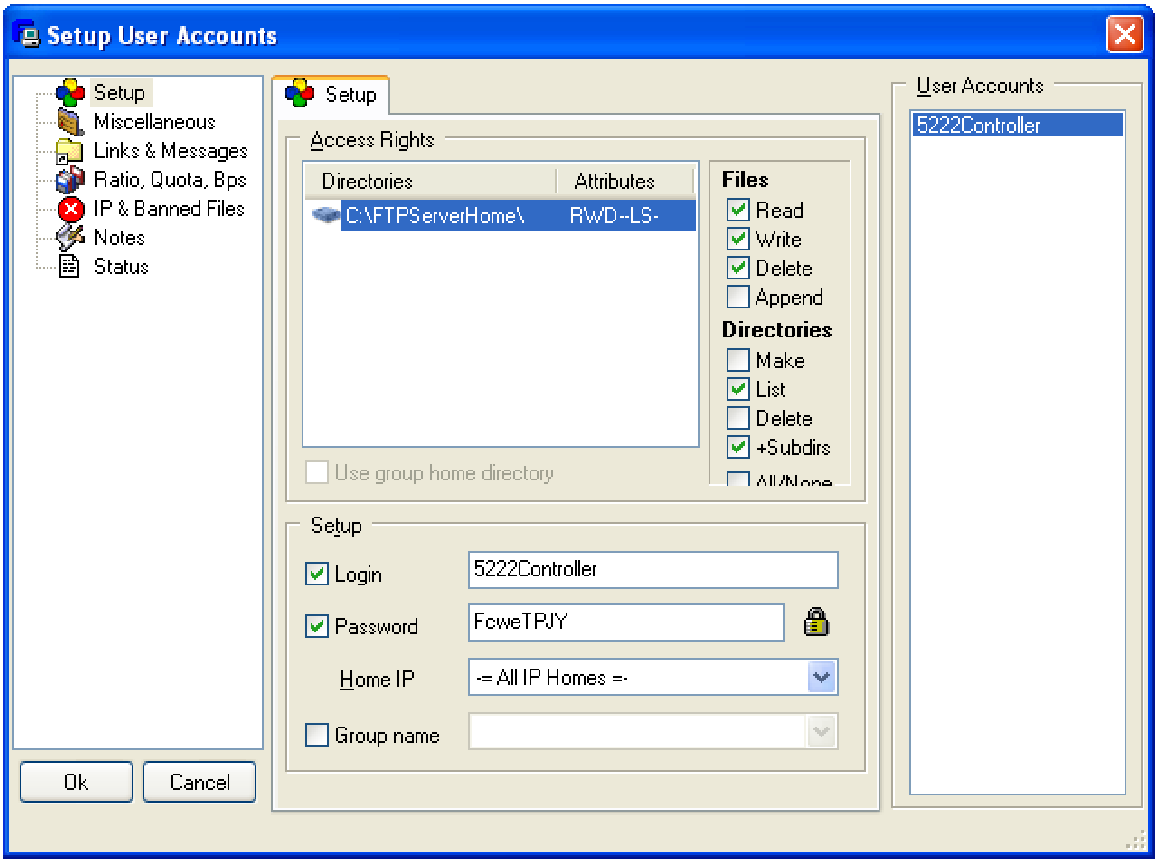


Files
☑ Read
☑ Write
☑ Delete
□ Append

Directories
□ Make
☑ List
□ Delete
☑ +Subdirs

□ All/None

C:\FTPServerHome



Directories	Attributes
C:\FTPServerHome\	RWD--LS-

Files: ☑ Read ☑ Write ☑ Delete ☐ Append

Directories: ☐ Make ☑ List ☐ Delete ☑ +Subdirs ☐ All/None

☐ Use group home directory

Setup

☑ Login: 5222Controller

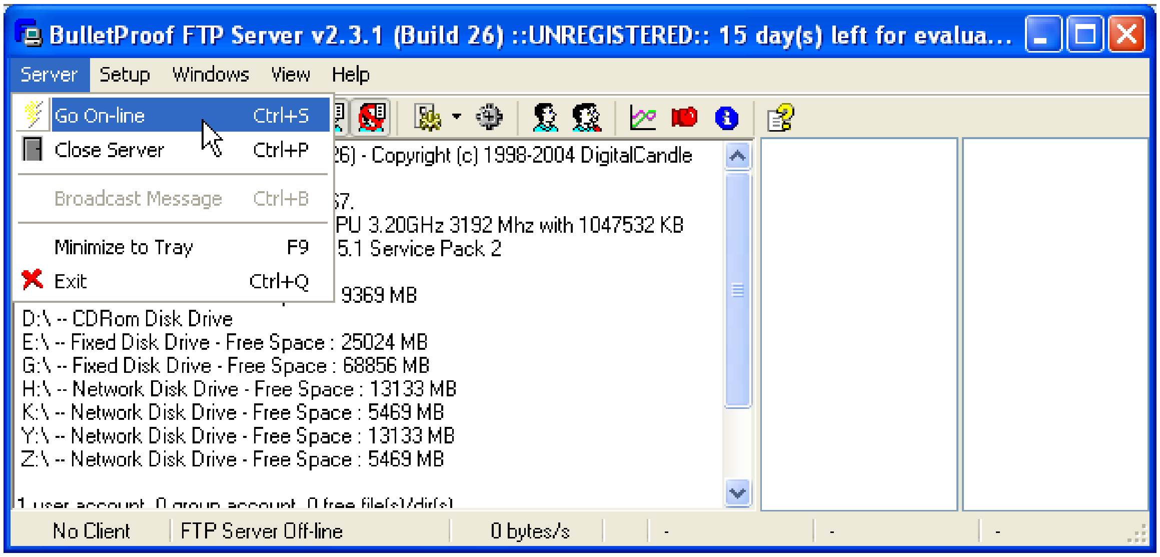
☑ Password: FcweTPJY

Home IP: -= All IP Homes =-

☐ Group name:

User Accounts: 5222Controller

To activate the server you must select OK, then "Server->Go On-line" at the main menu:



```
26) - Copyright (c) 1998-2004 DigitalCandle

57.
PU 3.20GHz 3192 Mhz with 1047532 KB
5.1 Service Pack 2

 9369 MB
D:\ -- CDRom Disk Drive
E:\ -- Fixed Disk Drive - Free Space : 25024 MB
G:\ -- Fixed Disk Drive - Free Space : 68856 MB
H:\ -- Network Disk Drive - Free Space : 13133 MB
K:\ -- Network Disk Drive - Free Space : 5469 MB
Y:\ -- Network Disk Drive - Free Space : 13133 MB
Z:\ -- Network Disk Drive - Free Space : 5469 MB

1 user account, 0 group account, 0 free file(s)/dir(s)
```

```
No Client | FTP Server Off-line | 0 bytes/s | - | - | -
```

There are numerous other features available within the BulletProof software package. It is left to the user to read their documentation available on their web site.

http://www.bpftpserver.com/help/bpftpserver.com/manual_en/

Example: Access via the telnet or script command line to this account would be:

ftpconnect 12.40.53.52 5222Controller FcweTPJY