

SV TEHS SIA

Development Tools for Java™

IPVES Application Note 09:

FTP server

SV TEHS SIA

IPJV-ES Application Note 09: FTP server

V 1.0

© SV TEHS SIA

Ruses 14-24 • LV1029 • Riga • Latvia

Phone: +371-9237495 +371-9223895 • Fax: +371-7332773

Email: info@svtehs.com • Web: <http://www.svtehs.com>

IPJVM and IPJV-ES are trademarks of SV TEHS SIA. ipStack, ipOS are trademarks of Ubicom, Inc. Java™ and all Java™-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. All other trademarks are property of their respective owners.



Introduction

IPJV-ES Development Board can be used in different applications.

The IPJV-ES Development Board with embedded virtual machine for Java™ offers an Ethernet based connection to the Internet and numerous interface possibilities to other equipment, include serial RS-232 DTE interface, serializer module with UART, SPI, GPSI and 10BASE-T Ethernet support, 6-channel 10-bit A/D inputs, analog comparator and 16 I/O pins.

The IPJVM virtual machine for Java is a clean room implementation, that has been specially optimized to run on device with limited amount of internal memory and designed for Java™ 2 Platform, Micro Edition (J2ME™) Connected Device Configuration (CDC) Foundation Profile.

A complete development toolkit available for application development with IPJVM platform. The IPJVM platform provide system designers and software developers simple, flexible and cost-effective solution for embedded Internet application rapid development and prototyping. The platform is combination of Uvicom IP2022 Internet Processor and a Java programmable runtime environment.

The IPJV-ES Development Board based on Uvicom IP2022 Internet Processor, optimized for Internet-edge applications. It handles protocol processing in software instead of in hard-wired logic, making the whole solution more adaptable to evolving standards and allow designer to use the same solution across a wide variety of internet-edge products simply by changing the software, thereby significantly reducing nonrecurring engineering (NRE) costs.

Typical IPJV-ES applications include Includes HTTP/FTP/SMTP/SNMP/Telnet servers, PPP support on embedded UARTs, encryption, security and authentication tools, reporting and alarming via e-mail, remote monitoring, control, management and maintenance.

Updates

New versions of the IPJV-ES software and applications can be obtained from the manufacturer's web site at:

<http://www.svtehs.com/ipjv.htm>

FTP server

Simple FTP server on IPJV Development Board.

For many different applications FTP server is the easiest way to remotely upload and download files. IPJV board have build-in FTP server, available only in the configuration mode. For more details about this server check IPJV User's Guide. This application describe FTP server, available in working mode as `FTPdServer`. For connection to the IPJV FTP server use IP address, set for the board and login credentials, set in the `ftpd.conf` configuration file. Simple test, running `FTFdServer`:

```
import jbvm.ip2k.*;
import java.text.*;
import java.util.*;
import java.io.*;
import java.net.*;
import com.svtehs.jftpd.*;
public class TestFtp
{
    public static void main (String[] aArg)
    {
        try { new VMJPinger().start(); } catch (Throwable e) {
e.printStackTrace(System.err); }
        try {
            FTPdServer.main();
        } catch (Throwable e2) { e2.printStackTrace(System.err); }
    }
}
```

FTP server support the following FTP requests:

| Request: | Description: |
|----------|---|
| ABOR | Abort previous command |
| ALLO | Allocate storage space for the file |
| CDUP | Change to parent of current working directory |
| CWD | Change working directory |
| DELE | Delete a file |
| HELP | Give help information |
| LIST | List files in a directory (ls -l) |
| MKD | Make a directory |
| MDTM | Show last modification time of file |
| NLST | Give name list of files in directory |
| NOOP | Do nothing |
| PASS | Specify password |
| PASV | Prepare for server-to-server transfer |
| PORT | Specify data connection port |
| PWD | Print the current working directory |
| QUIT | Terminate a session |
| REIN | Reinitialize |
| REST | Restart incomplete transfer |
| RETR | Retrieve a file |
| RMD | Remove a directory |
| RNFR | Rename from |
| RNTO | Rename to |
| SIZE | Return size of file |
| STOR | Store a file |
| SYST | Show operating system type of server system |
| TYPE | Specify data transfer type |
| USER | Specify username |
| XCWD | Change working directory |
| XMKD | Make a directory |
| XPWD | Print the current working directory |
| XRMD | Remove a directory |

FTP server authenticates users according to the user name and password, set in the the `ftpd.conf` configuration file. Several user name/password combinations with different permissions allowed to the FTP access.

FTP have several numeric replies, according to RFC959. Replies to File Transfer Protocol commands are devised to ensure the synchronization of requests and actions in the process of file transfer, and to guarantee that the FTP client always knows the state of the server. Every command generate at least one reply. An FTP reply consists of a three digit number (transmitted as three alphanumeric characters). This number is intended for use by FTP client to determine what state to enter next. Usually FTP clients put also their own text messages to user, based on this number. There are five values for the first digit of the reply code:

1yz *Positive Preliminary reply.* The requested action is being initiated; expect another reply before proceeding with a new command. This type of reply used to indicate that the command was accepted and the FTP client may now pay attention to the data connections, for implementations where simultaneous monitoring is difficult.

2yz *Positive Completion reply.* The requested action has been successfully completed. A new request may be initiated.

3yz *Positive Intermediate reply.* The command has been accepted, but the requested action is being held in abeyance, pending receipt of further information. The user should send another command specifying this information. This reply is used in command sequence groups.

4yz *Transient Negative Completion reply.* The command was not accepted and the requested action did not take place, but the error condition is temporary and the action may be requested again. The user should return to the beginning of the command sequence, if any.

5yz *Permanent Negative Completion reply.* The command was not accepted and the requested action did not take place. The FTP client is discouraged from repeating the exact request (in the same sequence). Even some "permanent" error conditions can be corrected, so the human user may want to direct his FTP client to reinitiate the command sequence by direct action at some point in the future (e.g., after the spelling has been changed, or the user has altered his directory status).

The following function groupings are encoded in the second digit:

x0z *Syntax.* These replies refer to syntax errors, syntactically correct commands that don't fit any functional category, unimplemented or superfluous commands.

x1z Information. These are replies to requests for information, such as status or help.

x2z Connections. Replies referring to the control and data connections.

x3z Authentication and accounting. Replies for the login process and accounting procedures.

x5z File system. These replies indicate the status of the Server file system vis-a-vis the requested transfer or other file system action.

The third digit gives a finer gradation of meaning in each of the function categories, specified by the second digit. All replies of the FTP server in numeric order listed in the table below.

| | |
|-----|--|
| 150 | File status okay; about to open data connection. |
| 200 | Command okay. (PORT, TYPE, NOP commands) |
| 213 | File status. (MDTM, SIZE commands) |
| 214 | Help message. |
| 215 | NAME system type. |
| 220 | Welcome. Service ready for new user. |
| 221 | Quit. Service closing control connection. |
| 226 | Closing data connection. Requested file action successful (file transfer okay or file abort okay). |
| 227 | Entering Passive Mode. |
| 230 | User logged in, proceed. |
| 250 | Requested file action okay, completed (CDUP, CWD, DELE, PWD commands). |
| 257 | "PATHNAME" created (MKD and PWD commands). |
| 313 | SIZE set okay. |
| 331 | User name okay, need password. |
| 350 | Requested file action pending further information. |
| 425 | Can't open data connection. |
| 426 | Connection closed; transfer aborted. |
| 500 | Too many users; syntax error, command unrecognized. |
| 501 | Syntax error in parameters or arguments. |
| 503 | Authentication failure (USER, PASS commands) |
| 530 | Not logged in or file not exist for REIN command. |
| 550 | Requested action not taken. File or directory not exist or permission denied (CWD, DELE, LIST, MKD, NLST, RMD, SIZE, STOR commands). |
| 552 | Exceeded storage allocation. |

FTP server use configuration file `ftpd.conf`, located by default in `/etc` directory. If the configuration file located in the different place, use the following format:

```
FTPDServer.main(String arg[]); arg[0]=<config path>
```

Sample configuration file have the following parameters:

```
port=21
maxconn=2
timeout=300
group=admin
user=svtehs
passwd=svtehs
welcome=/etc/welcome.txt
root=/
path=/
# f - full d - downl u - upload l - list r - remove n - rename
# m - mkdir i - rmdir c - cd
perm=f
group=anonymous
user=anonymous
passwd=@
welcome=/etc/welcome.txt
root=/ftp
path=/
perm=dlc
path=/incoming
perm=ulc
```

Common parameters are connection port, maximum number of the simultaneous connections, and timeout (sec). Users can be divided into several groups with different permissions.

Table of Contents

| | |
|------------------------|---|
| 1. Introduction | 1 |
| Updates | 1 |
| 2. FTP server | 2 |