UNISYS

BTOS
TCP/IP FTP and TELNET
**Operations and Programming
Guide**

UNISYS

BTOS
TCP/IP FTP and TELNET
Operations and Programming Guide

About This Guide

Purpose

This guide provides information for operation of two software applications using the Transmission Control Protocol/Internet Protocol (TCP/IP): File Transfer Protocol (FTP) and the virtual terminal protocol, TELNET.

Audience

The audience for this guide is the FTP and TELNET user who has some familiarity with the BTOS operating system and data communications concepts.

Scope

This guide describes how to send files using FTP and communicate with a TCP/IP host using TELNET. It does not include information on the administration and maintenance of TCP/IP, which is contained in the *BTOS TCP/IP Administration Guide*.

BTOS TCP/IP also supports Simple Mail Transfer Protocol (SMTP). SMTP is implemented jointly by BTOS TCP/IP and OFIS Mail (release level 1.2 or higher). The SMTP Manager (a component of BTOS TCP/IP) translates between the SMTP format and the OFIS Mail format. Users of BTOS systems use the standard OFIS Mail interface, while users of non-BTOS systems use the standard SMTP interface. The translation of the messages between the two systems is invisible to users.

See the *BTOS OFIS Mail Administration Guide* and the *BTOS OFIS Mail Operations Guide* for a description how to use OFIS Mail. See the *BTOS TCP/IP Administration Guide* for the installation of the SMTP Manager.

Prerequisites

Users of this guide must be familiar with the BTOS operating system, including how to execute commands and maintain files. In addition, TELNET users need to understand the basics of TELNET data communications.

Programmers who want to use TELNET's programmatic interface need to understand the BTOS programming environment.

How to Use This Guide

This guide is divided into three sections: an overview and a section each on FTP and TELNET operations. All users should read Section 1 for a general understanding of how the applications work. Then refer to the appropriate section for the application you are using.

Organization

This guide consists of three sections and one appendix:

Section 1. Overview

This section describes the features of FTP and TELNET.

Section 2. File Transfer Protocol

This section explains FTP concepts and terminology and provides instructions for using FTP.

Section 3. TELNET

This section explains TELNET concepts and terminology and provides instructions for using TELNET, including the programmatic interface.

Appendix A. BTOS Status Codes and FTP Protocol Codes

This section lists the BTOS status codes returned by the BTOS TCP/IP software. It lists also the TCP/IP protocol–defined status codes returned during the operation of FTP.

Appendix B. Queue Management Information

This appendix pertains to the queue management operations described in the *BTOS II System Reference Manual Volume 1* and specifically to the queue management procedures described in the *BTOS II System Procedural Interface Manual, Volumes 1 and 2.*

Appendix C. Modifying Samgen.Asm and Samgen.Mdf

This appendix gives you the information necessary to customize your Samgen.Asm and Samgen.Mdf files to operate with the TELNET bytestream interface.

A glossary, a bibliography, and an index follow Appendix C.

Related Product Information

The following documents are published by Unisys Corporation and can be ordered through Corporate Software and Publication Operations:

BTOS B-NET II Administration and Operations Guide (relative to release 1.0 or higher)

This guide contains configuration, installation, and troubleshooting information and procedures for B-NET administrators.

BTOS Context Manager II Installation and Configuration Guide (relative to release 3.0 or higher)

This guide contains introductory, procedural, and reference information about BTOS Context Manager II software.

BTOS Hardware Installation Guide

This manual describes the installation of BTOS hardware and supporting peripheral equipment.

BTOS Hardware Network Planning and Installation Guide

This guide helps you plan for connecting Unisys equipment to other Unisys equipment, or to a local area network.

BTOS Indexed Sequential Access Method (ISAM) Operations Reference Manual (relative to release 8.0)

This guide describes the features and operation of ISAM.

BTOS Intelligent Data Communications Module Systems Software (IDMSS) Operations Guide (relative to release 3.0 or higher)

This guide describes the installation and operation of the intelligent data communications module software.

BTOS OFIS Writer Operations Guide, 2 vols. (relative to release 1.0 or higher)

This two-volume guide describes the operation of BTOS OFIS Writer software, which can be used to edit the configuration files of BTOS TCP/IP.

BTOS OFIS Mail Administration Guide (relative to release 1.2 or higher)

BTOS OFIS Mail Operations Guide (relative to release 1.2 or higher)

BTOS TCP/IP FTP and TELNET Operations and Programming Guide (relative to release 1.0 or higher)

This guide contains information on how to use the BTOS TCP/IP upper layer protocols, FTP and TELNET.

BTOS X.25 Gateway Operations and Programming Guide (relative to release 9.0 or higher)

This guide contains information concerning installing and configuring the BTOS X.25 Gateway.

BTOS II Editor Operations Guide (relative to release 1.0)

This guide describes the features and operation of the BTOS II Editor.

BTOS II Standard Software Operations Guide (relative to release 1.0 or higher)

This guide contains introductory, procedural, and reference information for using the standard features of BTOS II.

BTOS II System Reference Manual, 2 vols. (relative to release 1.0 or higher)

This two-volume reference manual provides details on BTOS II concepts, system processes, functions, and services.

BTOS II System Procedural Interface Reference Manual, 2 vols. (relative to release 1.0 or higher)

BTOS II Language Development Linker and Librarian Programming Guide (relative to release 1.0)

This two-volume manual contains information on procedural interfaces available with BTOS II.

BTOS Pascal Compiler Programming Reference Manual (relative to release 6.0 or higher)

This manual contains information on the use of the BTOS Pascal Compiler.

BTOS C Compiler Programming Reference Manual (relative to release 1.1 or higher)

This manual contains information on the use of the BTOS C Compiler.

BTOS II System Status Codes Reference (relative to release 1.0 or higher)

This manual provides information on status codes you may encounter while using BTOS II.

XE500 BTOS Administration Guide (relative to release 7.1 or higher)

This guide contains information on performing system administrator functions for XE500 BTOS software.

XE500 Installation and Implementation Guide (relative to release 7.1 or higher)

This manual contains information on installing and using XEBTOS software.

XE500 BTOS Operations Guide (relative to release 7.1 or higher)

This guide contains information on executing BTOS master commands on the XE500 system.

Note: *Throughout this guide, all references to BTOS guides imply the BTOS or BTOS II guides, depending on your operating system.*

BTOS TCP/IP is based on standards described in the following documents:

- Internet Protocol MIL-STD-1777

- Transmission Control Protocol MIL-STD-1778

- Simple Mail Transfer Protocol MIL-STD-1781

- File Transfer Protocol MIL-STD-1780

- TELNET Protocol MIL-STD-1782

- Internet Protocol RFC (Request for Comment) 791

- Internet Control Message (ICMP) RFC 792

- Address Resolution Protocol (ARP) RFC 826

- Transmission Control Protocol RFC 793

- Simple Mail Transfer Protocol RFC 821

- File Transfer Protocol RFC 959

- TELNET Protocol RFC 854

- MIL-STD documents

These standards are available from the Naval Publications and Forms Center, 5801 Tabor Ave., Philadelphia, PA 19120-5099. The RFC documents are available from the DDN Information Center, SRI International-EJ 291, 333 Ravenswood Drive, Menlo Park, CA 94025.

Notation Conventions

This guide uses the following typographical conventions:

Words you are to type and that appear on the screen are shown in *italic*.

Keys are shown in uppercase letters; for example, press the GO key.

When two keys are to be used together for an operation, they are hyphenated. For example, SHIFT-RETURN means that while you hold down the SHIFT key, you press the RETURN key.

Contents

Section 3. TELNET

Section 1
Overview

Product Description

File Transfer Protocol (FTP) and TELNET are applications that use the
Transmission Control Protocol (TCP) and Internet Protocol (IP) implemented
in BTOS TCP/IP. TCP and IP prescribe the rules for the transfer of information
in a TCP/IP data communication network.

File Transfer Protocol

File Transfer Protocol (FTP) is a file transfer system that allows you to
exchange files over a TCP/IP network. You can transfer files between different
types of disk hardware subsystems on computers from different manufacturers.
FTP translates files into defined, standard formats for further processing and
transmission over the network.

FTP is implemented in accordance with Military Standard 1780, allowing you to
communicate with non-Unisys host computers that implement TCP/IP and FTP.

TELNET

TELNET is a bidirectional, eight-bit byte-oriented communications facility that
allows you to communicate with remote computers over a TCP/IP network. It
provides a standard method of interfacing terminals and terminal-oriented
processes to each other. It can also be used for terminal-to-terminal
communication and process-to-process communication.

A TELNET connection is a TCP connection that transmits data with TELNET
control information.

TELNET is implemented in accordance with Military Standard 1782, allowing
you to communicate with non-Unisys host computers that implement TCP/IP
and TELNET.

Features

This release of TCP/IP FTP and TELNET contains the features described in the following subsections.

File Transfer Protocol

FTP allows you to:

● Display host names and addresses with which you can perform operations.

● Extract details about a remote file and write them to a local file.

● Get a list of files on a remote host.

● Copy files to and from a remote host.

● Append data to a file on a remote host.

● Delete or rename a file on a remote host.

● Transfer BTOS ISAM files.

● Display the status of file transfer requests.

● Abort a request.

TELNET

TELNET has the following features:

● Provides the features of a network virtual terminal (NVT).

● Allows you to negotiate options for elegant terminal services, such as binary transmission and echo.

● Allows you to view the status of the connection, options reported by TELNET, and options reported by the remote host.

● Provides help information on using different commands and options and how to negotiate those options.

● Provides a programmatic interface that allows BTOS programs to open and use TELNET NVT-mode TCP connections.

Tasks and Commands

FTP and TELNET rely on software whose configuration and installation is described in the *BTOS TCP/IP Administration Guide*. The installation of this software requires in-depth technical knowledge and should be performed with the assistance of your system administrator. The tasks and commands specific to the operation of FTP and TELNET are as follows:

Task	Command	Location
Install FTP Service	Install TCP/IP FTP Service	Section 2
Invoke and operate FTP Client	TCP/IP FTP Client	Section 2
Invoke and operate TELNET	TCP/IP TELNET	Section 3

Section 2
File Transfer Protocol

This section describes File Transfer Protocol (FTP) features and the commands and procedures required for operations.

This section is organized as follows:

● An explanation of FTP concepts and terminology

● A description of how to install FTP Server and FTP Client

● An overview of how your workstation operates with FTP

● A description of basic features and operations

For more information on the supporting administrative tasks associated with FTP, see the *BTOS TCP/IP Adminstration Guide*.

FTP Concepts and Terminology

Figure 2-1 illustrates a typical file transfer. It is not required that you be familiar with all of these details when you use the system since menus prompt you for the necessary information, but understanding the process behind the screens will make the system easier to use.
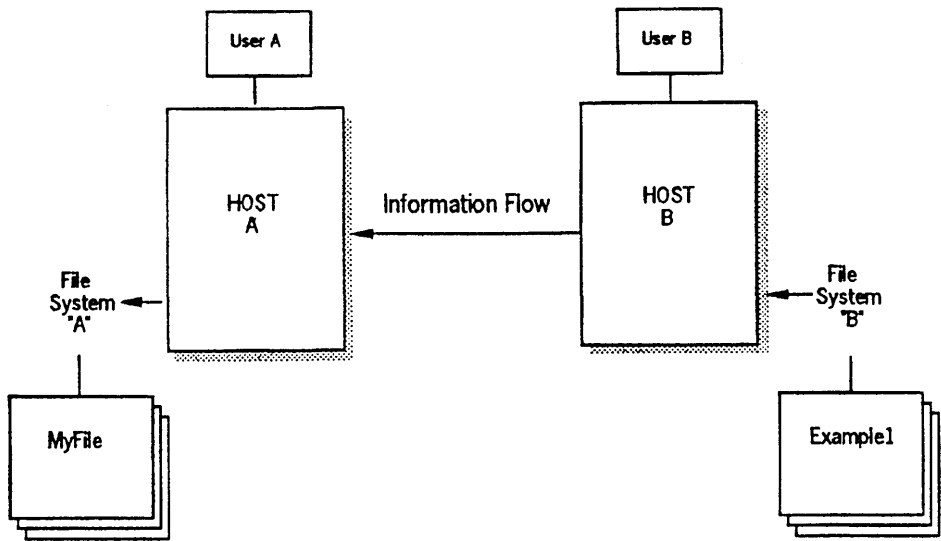
Figure 2-1. Overview of FTP

In Figure 2-1, a user at host computer *A* requests and receives a file named *EXAMPLE1* from host computer *B*, and stores it on the local system as *MYFILE*.

After the user signs onto the workstation and invokes FTP, a main menu screen appears. It displays the major function selections. User *A* selects the Copy function from the menu, and another form is displayed that contains the necessary fields required to transfer a file. User *A* then enters the desired remote file (*EXAMPLE1*), remote host (*B*), and local file (*MYFILE*). The screen is accepted, a transaction number is assigned, the transaction is placed in a queue, and finally it is transmitted. At the end of the file transmission, the hosts exchange acknowledgments and each PI terminates its respective DTP.

Installing FTP

Before installing FTP, it is necessary that other TCP/IP software, Queue Manager, and, in some cases, the Indexed Sequential Access Method (ISAM) be installed. The installation of this software requires a system administrator. The *BTOS TCP/IP Administration Guide* describes the procedure.

In order to install FTP you must first install the FTP system service, which performs the task of actually transferring files, and then invoke the FTP client, which performs the tasks of displaying screens and accepting your input form the keyboard.

To install the FTP service, at the Executive command line type *Install TCP/IP FTP Server* and press the GO key.

To install the FTP Client, at the Executive command line type *TCP/IP FTP Client* and press the RETURN key. The following form appears:

```
Install TCP/IP FTP Client
  [FTP Server Node Name (Default = Local)]
```

Parameter	Description
FTP Server Node Name (Default = Local)]	This parameter is the name of the B-Net node where the FTP Server is installed. The default is the local node.

Note: *When you connect the FTP Client to the FTP Server over B-Net, you can only send files that reside on the same volume as the FTP Client. Files that you receive will be stored on the volume on which you have installed the FTP Server.*

After completing the form, press the GO key to invoke the command.

Using Your Workstation with FTP

Several of the FTP screens use the functions keys, located at the top of each workstation keyboard. The function "labels" for each key appear at the bottom of the screen. To select a particular action, press the corresponding function key.

In addition to the function keys, you use these keys for basic operations:

Key	Function
GO	Transmit data to the system
RETURN, DELETE, BACK SPACE, and arrow keys	Manipulate data on the screen
CANCEL	Terminate data entry at any time and return to the previous menu
FINISH	End an FTP session. The system returns to the Executive command line. This does not abort any transaction already scheduled for processing
HELP	Obtain additional information on any menu options. The system typically returns to the previous menu when you are finished

Operational Considerations and Terminology

Transaction Numbers

A *transaction number* is a unique number FTP assigns to a user request at the time the request is made. This number is reported back to you in the screen header immediately after the request has been made, thus allowing you to track the status of your particular transaction (using the status function).

Status Information and Error Reporting

Status information is available in two forms. The last line in each screen header reports errors and general information throughout a session. In addition, you can use the Status command to obtain information about queued, active, and recently completed transactions. Information about your completed transactions remains in the status queue until you make a status request to find out if it completed successfully; at that point the information is deleted.

Host Names

Host names are parameters that identify either your host name or the host name where you wish to send a file. These names can be obtained from your system administrator. If you leave this field blank, the system assumes the user is on your local system.

Remote File Names

File names for remote systems must be in a format that the remote system can understand. FTP does not edit or interpret a file name that is to be sent to another host; it passes it along just as you entered it on·the screen. The user and password fields for remote systems may be needed to log into the remote file system, thus restricting you to files that are located only in that specific user's directory. Consult your system administrator for more information about accessing a particular remote system.

Local File Names

You must use valid BTOS file names. These names may be completely specified, such as "[D1]⟨TCP⟩ABC", or partially specified such as "ABC". In this latter case the user name specified on the request screen is used to add a [volume]⟨directory⟩ prefix to the name. For security reasons, FTP establishes a path based on the user name with which you logged onto the system, not the path established with the BTOS Path command. Wild-card names are not allowed.

Security

Security is largely the responsibility of each individual host in the network. When sending or retrieving files it is usually necessary to provide a valid user id/password for the remote system in order to access the desired file. For

example, when sending files to a UNIX system, the user id and password that you send will be interpreted by the UNIX system in its customary way. Note, however, that when sending files to a BTOS system, the user id is ignored. The password maps to the BTOS volume password on which you have installed the FTP Server. The password field that appears on several of the FTP screens is not displayed on the screen as it is typed in ("#" is displayed instead); however, for clarity in the sample screens this field is shown. As an extra security measure, this password is encrypted when the request is queued so as to prevent local security violations; it is decrypted as it passes to the remote FTP server.For more information about volume passwording, see the *BTOS Standard Software Operations Guide*

Priority

Most of the transaction screens allow the user to specify a priority in the range 0 to 9 (default is 9). This priority field is used to determine where to place the transaction in the schedule queue, with lower numbered priorities going to the top of the queue.

Basic Operations

After you invoke the FTP Client, the main menu is displayed (see Figure 2-2). You can make the following selections from the main menu, using the function keys:

Function Key	Function
Copy	Send files to or retrieve files from a remote host
ISAM	Send, retrieve, delete, or rename files between two BTOS systems
Append	Add a file to the end of another file
Rename	Change the name of a file at a remote host
Delete	Remove a file from disk storage at a remote host

Function Key	Function
Dir. List	Retrieve directory information from a remote host. It is stored locally in a file whose name is specified in the request. Be aware that the format of the list depends on the type of system on which the files reside.
File List	Retrieve file information from a remote host. It is stored locally in a file whose name is specified in the request. Be aware that the type of information returned depends on the system on which the files reside.
Abort	Terminate a specified request
Status	Report transaction status information
Hosts	Report remote host names and addresses

You can press the HELP key to obtain additional information about these functions.

Figure 2-2 illustrates the FTP main menu screen.

```
┌─────────────────────────────────────────────────────────────┐
│ User: Shaw                                    TCP/IP FTP X.Y │
│ Path: [Sys]<TCP>                    Wed Jul 19, 1989  9:16 AM │
├─────────────────────────────────────────────────────────────┤
│                                                              │
│                                                              │
│                                                              │
│                                                              │
│                                                              │
│                                                              │
│                                                              │
│                                                              │
│                                                              │
│ │Hosts│ Flist │ Dlist │ │ Copy │Appnd│ Delet │Renam│ ISAM │ Stats │Abort│ │
└─────────────────────────────────────────────────────────────┘
```
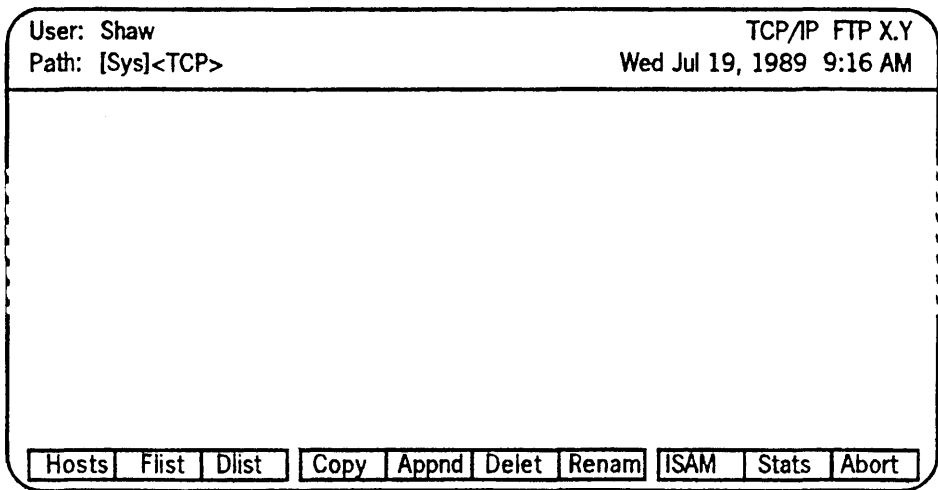
Figure 2–2. FTP Main Menu

You select all FTP commands from the main menu, except for Help, which you can invoke at any time. You choose the FTP commands displayed on the menu by pressing the corresponding function key.

Some commands prompt you to enter further selections or parameters; others ask you to press the GO key to confirm the selection.

Copying a File

Press the Copy function key to send a file to or retrieve a file from the host. The copy send screen appears.

Sending a File

Figure 2-3 illustrates a sample copy send screen. To send a file, enter the file you wish to send as the source file. Enter the file name — in a format understandable to the remote system — as the destination file.

```
/ User:  Shaw                                        TCP/IP  FTP X.Y \
  Path:  [Sys]<TCP>                        Wed Jul 19, 1989  9:16 AM
 ┌──────────────────────────────────────────────────────────────────┐
 │                                                                    │
 │   Copy                                                             │
 │      Source File                  Workfile                         │
 │            Host                                                    │
 │            User                                                    │
 │            Password                                                │
 │            Account                                                 │
 │      Destination File             Dir1/TestData                    │
 │            Host                   Phila.Com                        │
 │            User                   MASTER                           │
 │            Password               PRISM                            │
 │            Account                                                 │
 │                                                                    │
 │      Type   (Type A, E, or I)     ASCII   EBCDIC   Image           │
 │      Structure   (Type F or R)    File   Record                    │
 │      Priority (0-9)                                                │
 │                                                                    │
 \────────────────────────────────────────────────────────────────── /
```

Figure 2-3.   Completed Copy Send Screen

You can leave the fields for host, user, and account blank for the source file specification; FTP already has the information it needs to access your local file. You may have to specify a password (see your system administrator). You can leave the field for the name of your local host blank; blanks default to the local host name.

The destination file and its associated set of parameters need to be filled in according to the requirements of the remote host. *The file name format should be understandable to the remote system, not necessarily to BTOS on your host.* The user and password fields are used to log into the remote system, and account is optional for some remote systems, required for others. Your system administrator can provide you with these details.

The request shown in Figure 2-3 would result in the file [SYS]⟨TCP⟩Workfile being transferred from Shaw's host to a remote host named Phila.Com. The request would first log on at Phila.Com using the user id and password MASTER/PRISM. Phila.Com would map this code to its local file security system and then store the file under the user name Master with file name "Tranfile." Note that it is not necessary to enter any information other than the

file name for the local part of this transaction. Local user information is taken from the current path, displayed in the upper left corner of the screen. You can specify the local file completely if it is located in a directory other than the one displayed. For example, you could enter [D0]⟨TCP⟩Workfile instead of just "Workfile."

The parameters at the bottom of the screen are typically left at their default values, although the FTP protocol does support these options and you can select different values if there is a specific need. The effect of these options is somewhat dependent upon the remote host, although the general effects are as described here. Consult your system administrator regarding these parameters.

TYPE

ASCII.   You can use ASCII to copy files to any host on the network. This is the data type used on BTOS systems.

EBCDIC.   You can specify this mode to transfer a file to a system that uses EBCDIC as the native data representation format. The translation to EBCDIC occurs during the file transfer, and the data becomes immediately meaningful to the remote system. Do not use EBCDIC in retrievals.

IMAGE.   Specify this type if you need to transfer a file that contains characters not in the standard ASCII character set. The file is sent exactly as is with no translation of the data. Program code files or data files that contain 8-bit binary data are examples of files with which this attribute could be used. When transferring files between BTOS systems, the FTP program automatically sets attributes and passes additional information so that the remote system ends up with an identical copy of the file.

STRUCTURE

FILE.   This type stores files as a sequence of data bytes, with no record boundaries added to the data. This is the standard file structure for BTOS systems.

RECORD.   Specify this type to send files to foreign systems. It breaks files up into records as they are sent out. Because BTOS systems do not store files using a record format, FTP searches for the ⟨NL⟩ (next line) character as a record delimiter and substitutes the FTP end-of-record indicator for the ⟨NL⟩ as it is sent.

When receiving a file in this mode the FTP end-of-record indicator is translated into an ⟨NL⟩ character and inserted into the file at the end of each record. This mode is useful when, for example, you want to retrieve source programs from a large mainframe (such as a Unisys A-series or an IBM 308x) that has a record-oriented file system. A file retrieved in this manner could be accessed directly with the BTOS Editor or compilers, with no "fix-up" required.

MODE

STREAM.   Files sent to other systems are sent as a stream of bytes.

FORMAT

Non-print.   Files are transferred between hosts as regular data files and are not interpreted as print data with embedded carriage control information.

Retrieving a File

Figure 2-4 illustrates a sample copy retrieve screen. The process of retrieving a file is similar to that for sending one. The only difference is in the specification of the source and destination host names — the direction of the transfer is determined by which one is the local host name.

Specify the remote system file you wish to retrieve in the source file field. Specify the local file name in the destination file field. Fill in all fields for the source file, but only the destination file field for the local file name. A sample retrieval is shown in Figure 2-4.

```
┌─────────────────────────────────────────────────────────────────┐
│ User:  Shaw                                      TCP/IP  FTP X.Y  │
│ Path:  [Sys]<TCP>                       Wed Jul 19, 1989  9:16 AM │
├─────────────────────────────────────────────────────────────────┤
│                                                                   │
│     Copy                                                          │
│         Source File              Dir1/TestData                    │
│              Host                Phila.Com                         │
│              User                MASTER                           │
│              Password            PRISM                             │
│              Account                                              │
│         Destination File         Workfile                         │
│              Host                                                 │
│              User                                                 │
│              Password                                             │
│              Account                                             │
│         Type   (Type A, E, or I)   ASCII   EBCDIC   Image         │
│         Structure  (Type F or R)   File   Record                  │
│         Priority (0-9)                                            │
│                                                                   │
└─────────────────────────────────────────────────────────────────┘
```

Figure 2–4.   Completed Copy Retrieval Screen

For both sending and retrieving, the source and destination host fields must be different. FTP transfers files between systems, not within a single system.

Appending a File

To add a file to the end of another file, press the Append function key. The append file screen appears.

This screen contains the same fields as the copy screen; the only difference is that the contents of the source file are added to the end of the destination file. If the destination file does not exist, it is created as part of the append operation.

Renaming a File

Press the Rename function key to change the name of a file at a remote site, subject to the limitations imposed at the remote site concerning legal names. The rename screen appears.

Figure 2-5 illustrates a sample completed rename screen. In the figure, file "Draft.wp" is renamed "Final/Document.wp".

```
User: Shaw                                          TCP/IP  FTP X.Y
Path: [Sys]<TCP>                         Wed Jul 19, 1989  9:16 AM

    Rename
       Old Filename              Draft.wp
       New Filename              Final/Document.wp
       At host                   Phila.Com
       User                      MASTER
       Password                  PRISM
       Account
       Priority  (0-9)
```

Figure 2- 5.   Completed Rename Screen

Deleting a File

Press the Delete function key to delete a file at a remote site, subject to the limitations imposed at the remote site concerning file deletions. The delete screen appears.

Figure 2-6 illustrates a sample completed delete screen. In the figure, the file Draft.wp at Phila.Com is deleted.

```
┌─────────────────────────────────────────────────────────────────┐
│  User:  Shaw                                        TCP/IP  FTP X.Y │
│  Path:  [Sys]<TCP>                          Wed Jul 19, 1989  9:16 AM │
│ ─────────────────────────────────────────────────────────────────── │
│                                                                     │
│     Delete                                                          │
│        File list                     Draft.wp                       │
│        At host                       Phila.Com                      │
│        User                          MASTER                         │
│        Password                      PRISM                          │
│        Account                                                      │
│        Priority (0-9)                                               │
│                                                                     │
│                                                                     │
│                                                                     │
│                                                                     │
│                                                                     │
└─────────────────────────────────────────────────────────────────┘
```
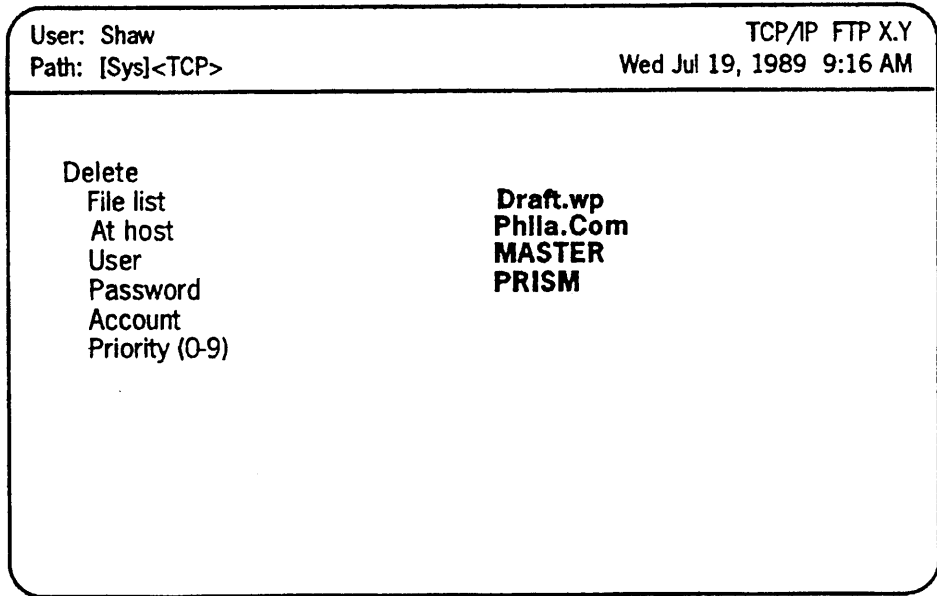
Figure 2–6.   Completed Delete Screen

Obtaining File Names

To retrieve a list of all file names under a specified directory, press the
Directory List function key. The directory list screen appears. Be aware that the
format of the list depends on the type of system on which the files reside.

Your workstation stores this information in a file that you can later print or
examine using the BTOS Editor. Figure 2-7 illustrates a sample completed
directory list screen. In the figure, a list of files under the Inventory directory is
returned to the local system and stored in a file named
[D0]⟨Remote⟩Inventor.lst. You can leave the destination directory field blank if
you want to store the file in the current user directory.

```
┌─────────────────────────────────────────────────────────────────────┐
│ User:  Shaw                                              TCP/IP  FTP X.Y  │
│ Path:  [Sys]<TCP>                                 Wed Jul 19, 1989  9:16 AM │
├─────────────────────────────────────────────────────────────────────┤
│                                                                       │
│    Directory List                                                     │
│       Source Directory          Inventory                             │
│              Host               Phila.Com                             │
│              User               MASTER                                │
│              Password           PRISM                                 │
│              Account                                                  │
│       Destination File          [D0]<Remote>Inventory.lst             │
│              Host                                                      │
│              User                                                     │
│              Password                                                 │
│              Account                                                  │
│                                                                       │
│       Type   (Type A, E, or I)   ASCII   EBCDIC   Image               │
│       Structure  (Type F or R)   File  Record                        │
│       Priority (0-9)                                                  │
│                                                                       │
└─────────────────────────────────────────────────────────────────────┘
```

Figure 2-7. Completed Directory List Screen

Obtaining File Information

To retrieve detailed information about a particular file, press the File List
function key. The file list screen appears.

Your workstation stores this information in a file that you can later print or
examine using the BTOS Editor. Figure 2-8 illustrates a sample completed file
list screen. In the figure, information about Inventory/Supply/Totals at
Phila.Com is returned to the local system and stored in a file named
⟨TCP⟩SupplyInfo. If you leave the destination file field blank or specify a
directory, the File List command works in the same way as the Directory List
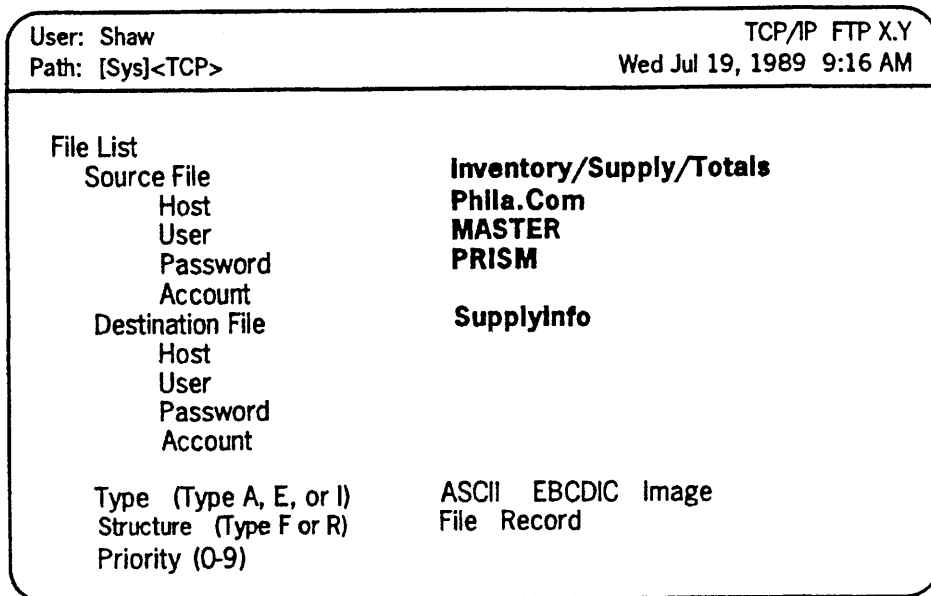command.

```
┌─────────────────────────────────────────────────────────────────────┐
│  User:  Shaw                                         TCP/IP  FTP X.Y  │
│  Path:  [Sys]<TCP>                             Wed Jul 19, 1989 9:16 AM│
├─────────────────────────────────────────────────────────────────────┤
│                                                                       │
│      File List                                                        │
│          Source File              Inventory/Supply/Totals             │
│              Host                 Phila.Com                            │
│              User                 MASTER                               │
│              Password             PRISM                                │
│              Account                                                   │
│          Destination File         SupplyInfo                          │
│              Host                                                      │
│              User                                                      │
│              Password                                                  │
│              Account                                                   │
│                                                                       │
│          Type  (Type A, E, or I)   ASCII   EBCDIC   Image             │
│          Structure  (Type F or R)  File   Record                      │
│          Priority (0-9)                                               │
│                                                                       │
└─────────────────────────────────────────────────────────────────────┘
```

Figure 2–8.   Completed File List Screen

Obtaining Status Information

Press the Status function key to retrieve information about those transactions that have been queued for later processing, those that are currently in process, those that are aborting, those that were recently completed, and those initiated by a remote system. The status report screen appears.

Because there are a limited number of connections that can be active at any one time (this value is specified by your system administrator) it is helpful to know where your request stands relative to others in the wait queue. Additionally, this report informs you of the outcome of your transaction. This information remains in the report until you examine the results; then it is deleted. Figure 2-9 illustrates a sample status report screen.

To view a status report that continues on to another screen, press any alphanumeric key. Press the CANCEL key to return to the main menu. The status screen reports the following information:

```
┌─────────────────────────────────────────────────────────────────────┐
│ User:  Shaw                                          TCP/IP  FTP X.Y   │
│ Path:  [Sys]<TCP>                              Wed Jul 19, 1989  9:16 AM│
│                       Press any key to continue                        │
├────────────────────────────────────────────────────────────────────┤
│  Status                                                               │
│                                                                       │
│  TRN#      STATUS        PR      ACTION                                │
│                                                                       │
│  0009      Done          07      Copy     <TCP>TestData                │
│                                           At HAWAII                    │
│                                           To <MASTER>Inventory         │
│                                           By Shaw                      │
│                                           Completed Successfully       │
│                                                                       │
│  0010      Queued        07      Flist    <AFPOL>Supplies              │
│                                           At HOUSTON                   │
│                                           To <TestCase1>FileInfo       │
│                                           At NEW-YORK                  │
│                                           By  Admin                    │
│  –End of Status–                                                      │
└─────────────────────────────────────────────────────────────────────┘
```

Figure 2–9. Completed Status Report Screen

TRN#	The transaction number assigned by FTP at the time the request was made. These numbers are unique throughout FTP.
STATUS	The current status of the transaction (Queued, Active, Abort(ing), Done, and Server).
PR	Priority of the request. Priority is used only to place transactions within the wait queue and has no effect once a transaction is active. Therefore, a higher priority will cause a transaction to be selected from the queue ahead of others, but it receives equal treatment once it has established a connection to a remote host.
ACTION	The type of transaction as selected from the main menu (for example, Copy, Flist, Delete).

File Data	Information identifying the source and destination files and hosts as well as the requestor, reported on separate lines. The last line reports completion status information for transactions that have finished.

Aborting a Request

To abort a request, press the Abort function key. The abort screen appears.

If the transaction is active the connection is broken and the request cancelled; a queued transaction is removed from the queue. Figure 2-10 illustrates a sample completed abort screen.

```
User:  Shaw                                    TCP/IP  FTP X.Y
Path:  [Sys]<TCP>                       Wed Jul 19, 1989  9:16 AM

  Abort

     Transaction #  .  .  .
```

Figure 2-10.   Completed Abort Screen

Processing ISAM Files Between Two BTOS Systems

Press the ISAM function key to send, retrieve, delete, or rename files to be processed between two BTOS systems.

The ISAM menu screen appears. From it, you can select the following operations:

- Home
- ISAM Send
- ISAM Retrieve
- ISAM Delete
- ISAM Rename

Figure 2-11 illustrates the ISAM menu.

```
User: Shaw                                    TCP/IP  FTP X.Y
Path: [Sys]<TCP>                    Wed Jul 19, 1989  9:16 AM




  Home         ISend  IRetr  IDele  Rrenm
```
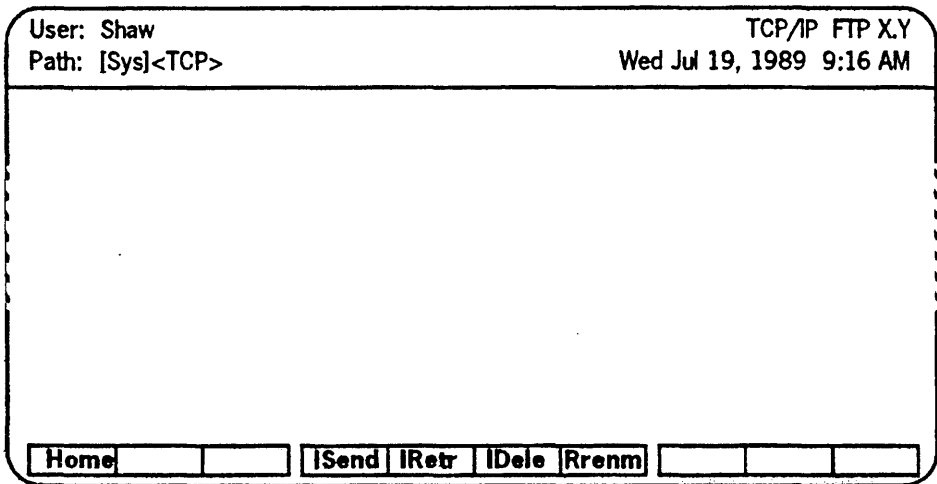
Figure 2-11. The ISAM Menu

The Home function key returns to the main menu. The screens for the other functions are similar to those previously described, although certain fields are missing. FTP uses specific options and formats to handle ISAM data, thus not allowing you to change the values of certain parameters. Figure 2-12 illustrates a sample ISAM file send screen.

```
┌────────────────────────────────────────────────────────────────┐
│  User:  Shaw                                    TCP/IP  FTP X.Y  │
│  Path:  [Sys]<TCP>                       Wed Jul 19, 1989  9:16 AM│
├────────────────────────────────────────────────────────────────┤
│                                                                  │
│    BTOS ISAM Send                                                │
│                                                                  │
│        Source ISAM Data Set          [D1]<Inventory>ABC          │
│               Password               Sailboat                    │
│        Destination  ISAM Data Set    [D2]<Inven1989>Data         │
│               Host                   Phila.Com                   │
│               User                   MASTER                      │
│               Password               PRISM                       │
│        Priority  (0-9)                                           │
│                                                                  │
└────────────────────────────────────────────────────────────────┘
```

Figure 2 – 12.   Completed ISAM File Send Screen

BTOS ISAM files require special processing because they contain embedded file name and disk drive information that must be changed during the file transfer to another BTOS system. Also, ISAM consists of multiple files, not a single file as is the case with other conventional files. Be sure to use the ISAM data file name in file name specifications, not the index file name.

In Figure 2-12, the ISAM files associated with the name [D1]⟨Inventory⟩ABC are sent from the local host to the remote BTOS host Phila.Com. The directory name is changed to ⟨Inven1989⟩ on drive [D2] in the process. These changes require some internal changes to the data that is sent, hence the main reason for the special processing. Also, this file points to the index file associated with this data file, and it is transferred automatically. If the old name for the index file was [D1]⟨Inventory⟩ABC.ind, the new name would be [D2]⟨Inven1986⟩Data.ind.

In other words, the suffix *ind* is added to the new data file name to get the new index file name.

Handling ISAM Files

Keep the following in mind when you send, retrieve, delete, or rename ISAM files:

- Use the ISAM command only for BTOS-to-BTOS operations.

  To transfer ISAM files to a non-BTOS system, transfer each file individually using the Copy screen and specify the binary option.

- An ISAM data set is usually made up of two files: One contains the data, and the other contains indices.

- To specify an ISAM file name for an FTP operation, always use the name of the data file. FTP assigns the index file name automatically.

- When you specify a new ISAM name for a send or a retrieval, the data file will be given the name that you specify and the index file will be given that same name but with an *ind* suffix.

- When you send an ISAM file, specify the volume and the directory of the source file and the destination file. Be sure you use the actual volume name for the destination ([d0], [d1], etc.) and not just [Sys]. If you specify [Sys], the receiver will need to use the ISAM REORG command after receiving the file.

- When you retrieve and ISAM file, the dataset and index file must be in the same directory; the index file should be specified as *DatasetName.ind*. If the files are not in this format, you will need to use the ISAM REORG command after receiving the files.

  As with ISAM sends, specify the volume and the directory of the remote system and your local system. Be sure you use the actual volume name of your local system ([D0], [D1], etc.) and not just [Sys]. If you specify [sys], you will need to use the ISAM REORG command after receiving the files.

Exiting FTP

Press the FINISH key to end an FTP session. The system returns to the Executive command line. This does not abort any transaction already scheduled for processing.

Section 3
TELNET

This section describes TELNET features and the commands and procedures required for operations.

This section is organized as follows:

- An explanation of TELNET concepts and terminology

- An overview of how your workstation operates with TELNET

- A description of basic features and operations

- A description of the TELNET programmatic interface

For more information on the supporting administrative tasks associated with TELNET, see the *BTOS TCP/IP Adminstration Guide*. See Military Standard 1782 for a complete discussion of the TELNET protocol.

TELNET Concepts and Terminology

Network Virtual Terminal (NVT)

TELNET allows you to access remote systems on the network as a network virtual terminal (NVT). You can log in to a remote host on the network as if your workstation were a local terminal of that system. Once the connection is established, data you enter is transmitted through the network to the remote system, and data received from the remote host is displayed on your screen.

Although your workstation can access the resources of the remote host, remote terminals cannot access your local system resources. (A local server TELNET is required in order for that to occur, which this release does not implement.)

However, TELNET does allow you to "listen" to a specific TCP port for an incoming "call" from a remote system. This feature makes it possible for a local workstation to communicate with a remote terminal or program, and a local program to communicate with a remote terminal or program.

Figure 3-1 illustrates a sample TELNET screen. In the figure, user Andrea, on local host Unisys.Com, has a connection established with remote host Phila.Com.

```
 User         : Andrea                          TELNET   X.Y
 Local Host   : Unisys.Com          Wed Jul 19, 1989    11:30  AM
 Remote Host  : Phila.Com            Press  F1 for Command mode
 Connection   : Open                    (Error/Status messages)

 (Data frame)







 (Command frame)
```

Figure 3–1. Sample TELNET Screen

At the top of the screen is the status frame, which displays information about the connection.

The data frame displays data entered from the local keyboard and data received from the remote system. (The entire screen scrolls up one line when the display reaches the bottom of the screen.)

In data mode, the data frame extends to the bottom of the screen. Whenever you press the F1 key to invoke command mode, the command frame with command menu appears and overwrites data in the data frame. However, the display is saved and will be restored when you switch back to data mode.

Negotiated Options

The standard NVT is by default an asynchronous, ASCII character–oriented, scroll–mode terminal, which is usually referred to as a dumb TTY (Teletype) terminal (see "TELNET--Dumb TTY Terminal"). Many hosts provide additional services over those available within an NVT, which users with sophisticated terminals may desire. The option negotiation feature allows users and servers to agree to use a more elaborate set of conventions for their TELNET connection. Such options include, for example, changing the character set or the echo mode. Either party (or both) can initiate a request that some option take effect. The other party may then either accept or reject that request.

Modes of Communication

TELNET supports the following modes of network communication:

● A workstation or program can access any host on the network that implements server TELNET.

● Two workstations/terminals on separate hosts can communicate with each other in "chat" mode.

● Two programs on separate hosts can communicate with each other.

● A workstation or program can communicate with another program or terminal on a different host.

Modes of Operation

There are two TELNET modes of operation: data mode and command mode. When TELNET is initially invoked, it is in command mode, so you can send commands to open a connection. After a connection is established it automatically switches to data mode. In data mode, characters typed in at the keyboard are sent to the remote system, and characters received from remote are displayed on the screen. You can switch to command mode any time by pressing the F1 function key or display the Help menu by pressing the HELP key.

When TELNET is waiting for a negotiation reply (in command mode), you can still invoke the command menu and send other commands. If you try to transmit data, the following error message appears: *Waiting for negotiation reply*.

Using Your Workstation with TELNET

The following keys have specific functions in TELNET. F1, F10, HELP, and FINISH are TELNET application keys. The remainder are TELNET terminal keys.

Key	Function
F1	Enter command mode and display command menu.
F10	Send Break key code. It causes the remote host to respond as if a local user had pressed the BREAK key or ATTENTION key on his TTY keyboard.
HELP	Display the Help file, which describes functions of special keys and TELNET commands.
FINISH	Quit TELNET and return to the BTOS Executive line.
CANCEL	Local echo mode: cancel a command (in command mode; send Bell signal (in data transfer mode). Remote echo mode: send 07 hex.
RETURN	Local echo mode: transmit message. Remote echo mode: send OD 00 hex.

CODE-⟨char⟩ | Local echo mode: "Control key" equivalent. Remote echo mode: see "Control Codes."

CODE-RETURN | Local echo mode: carriage return (without line feed); move the cursor to the beginning of current line. Remote echo mode: sends 0D 00 hex.

Down arrow | Local echo mode: line feed (without carriage return); move the cursor down one line. Remote echo mode: sends 0A hex.

BACKSPACE | Local echo mode: move the cursor one position to the left without erasing the character. Remote echo mode: sends 08 hex.

CODE-BACKSPACE | Local echo mode: backspace and erase a character at the cursor position. Remote echo mode: TELNET character function.

CODE-DELETE | Local echo mode: backspace and erase line at the cursor position. Remote echo mode: TELNET delete line function.

DELETE | Local echo mode: erase the current line. Remote echo mode: send 7F hex.

Up arrow | Local echo mode: move the cursor to the next vertical tab stop. On your workstation, the cursor moves up one position. However, it is unspecified how a remote host determines where such stops are. Remote echo mode: send 0B hex.

TAB | Local echo mode: move the cursor to the next horizontal tab stop. On your workstation, the cursor moves eight positions to the right. However, it is unspecified how a remote host determines where such tab stops are. Remote echo mode: send 09 hex.

NEXT PAGE   Local echo mode: clear the current page and move the cursor to the left top corner of the data frame.
Remote echo mode: send 0C hex.

TELNET- Dumb TTY Terminal

A BTOS workstation running TELNET appears to the remote host as a "dumb terminal." UNIX users may define the TELNET terminal capabilities to the UNIX Shell by *TERM=tty; export TERM* (in the Bourne Shell) or *setenv term tty* (in the BSD Shell). The following tables describe a BTOS workstation operating as a TELNET terminal in terms of its keyboard functionality.

TELNET Function Keys

The following keys interact with the local TELNET Application and do not transmit data to the remote host:

Key	Function
F1	Invokes the TELNET Application Menu
HELP	Invokes the TELNET HELP Function
FINISH	Only available from the F1 Menu.
CANCEL	Only available from the F1 Menu.

TELNET Terminal Keys

Key(s)	TELNET Key Function
F10	TELNET BREAK Function
CODE-DELETE	TELNET DELETE LINE Function
CODE-BACKSPACE	TELNET DELETE CHAR Function

These keys transmit data to the remote host:

Key	Data Transmitted	ANSI Mnemonic(s)
CANCEL	07	BEL

DOWN ARROW	0A	LF
UP ARROW	0B	VT
TAB	09	TAB
NEXT PAGE	0C	FF
RETURN	0D 0A	CR LF (two bytes of data)
ENTER	0D 0A	CR LF (two bytes of data)
CODE-RETURN	0D 00	CR NUL (two bytes of data)

ASCII Graphic keys

The following keys are the BTOS workstation keys that have printable graphics in the ASCII character set:

SPACE, and 21 thru 40 (hex) !"# $ %&'()*+' ./ 0 thru 9 :;⟨=⟩?@41 thru 7F (hex) A thru Z, a thru z, {|}~

(other graphic keys, e.g., ¢ and ½, are ignored)

Control Codes

The following keys are the BTOS workstation keys that transmit non-printable codes in the ASCII character set:

Code	Key Sequence
00	CONTROL-SPACE
01	CONTROL-A thru
1A	CONTROL-Z
1B	CONTROL-[
1C	CONTROL-\
1D	CONTROL-]
1E	CONTROL-ˣ
1F	CONTROL-_ (underscore)
7F	DELETE

Key Sequence	Action Taken
00: NUL	ignored
01: BEL	sounds BEEP
03: STX	ignored
03: ETX	ignored
04: EOT	ignored
05: ENQ	ignored
06: NUL	ignored
07: BEL	sounds BEEP
08: BS	backspaces
09: HT	ignored
0A: LF	advances to next line - scrolls if required
0B: VT	ignored
0C: FF	clears screen
0D: CR	advances to next line - scrolls if required
0E: SO	ignored
0F: SI	ignored
18: CAN	ignored
1A: SUB	ignored
1B: ESC	ignored
7F: RUB	ignored

Basic Operations

To begin a TELNET session, type *TCP/IP TELNET* at the Executive command line. Press the RETURN key and fill in the node name where TCP/IP is installed if you want to connect TELNET to TCP/IP over B-Net. Otherwise press the GO key.

The TELNET Main Menu

The main menu is the first screen you see (see Figure 3-3). You can make the following selections from the main menu:

Command	Function
Abort	Unilaterally close an open or established TELNET connection
Call	Establish an active connection to a remote host
Hang Up	Close an open connection and discontinue sending data to the remote TELNET
Listen	Establish a passive connection to a remote host
Are You There	Verify that the remote host is operating
Abort Output	Stop receiving screen output from a particular process running on remote host
Extended Options	Negotiate an additional 256 options
Host Names	Obtain all remote host names and the corresponding Internet Protocol (IP) addresses set up on the system
Interrupt Process	Suspend, interrupt, abort, or terminate a particular process running on the server, depending on remote host implementation
Status	Obtain status information about the connection, options reported by the local TELNET, and options reported by the remote TELNET

Command	Function
Synch	Synchronize the connection
Verbose	Obtain information about all option negotiation activities
Record	Open a file into which every character entered at the keyboard or received from the remote host is to be recorded
Stop Record	Close a recording file
Do	A request to the remote host that some option take effect
Don't	A demand to the remote host that some option be discontinued
Will	An offer or agreement to implement some option
Won't	A refusal to implement some option

You can press the HELP key to obtain additional information about these functions.

Figure 3-2 illustrates the TELNET main menu screen.

You select all TELNET commands from the main menu, except for Help, which you can invoke at any time. You choose the TELNET commands displayed on the menu by pressing the corresponding key; for example, press the C key (either lower or upper case) to select the Call command.

Some commands prompt you to enter further selections or parameters; others ask you to press the GO key to confirm the selection.

Establishing a TELNET Connection

You can initiate two kinds of connections to a host, active and passive. You initiate an active session using the Call command; a passive connection is initiated by the Listen command. Both commands — one initiated by the local workstation, the other by the remote — are necessary for data transmission.

Call

To initiate an active connection to a remote host, press the C key from the main

```
User        : Andrea                          TELNET  X.Y
Local Host  : Unisys.Com        Wed Jul 19, 1989   11:30 AM
Remote Host : Closed
Connection  :
─────────────────────────────────────────────────────────────
TELNET COMMANDS:   Select a key, or CANCEL to cancel
  A. Abort                          R.  Record
  C. Call                           $.  Stop Record
  H. Hang up
  L. Listen
  Y. Are you there
  O. Abort Output
  E. Extended Options
  X. Host Names
  I. Interrupt Process
  S. Status
  N. Synch
  V. Verbose

  1. Do
  2. Don't
  3. Will
  4. Won't

     FINISH
     HELP
```

Figure 3-2. TELNET Main Menu

menu. The call screen appears.

The screen text that follows illustrates a sample completed call screen. In the sample, *Phila.Com* has been specified as the remote host. This is the only field you need to complete; TELNET selects appropriate default values for the other fields.

```
CALL: Press GO to execute, or CANCEL to cancel

     Remote Host   Phila.Com
     [Remote Port]
     [Local Port]  2001
```

When you use the Call command, TELNET issues an active open message to a remote host. The active session is established only if the destination host's server TELNET has sent a Listen command to the corresponding port.

Remote Host	The remote host being called. It should be alphanumeric and defined in the ⟨sys⟩TCP.Config.sys parameter file. As a convenience, you can enter an IP address preceded by an "@" character, for example, @026.000.000.073. All 12 digits must be entered, including leading zeroes.
[Remote Port]	The remote host (or destination) port being called. It must be numeric and within the range 0–65535. The default value of 23 is used if you leave this field blank. Port 23 is the default "listen" port assigned by convention to TELNET.
[Local Port]	The local (source) port. ⟨local port⟩, if entered, must be numeric and within the range 1024–5535. An attempt to open a connection will fail if ⟨local port⟩ is in use by another connection. If you leave this field blank, an unused port will be chosen.

Listen

To initiate a passive connection to a remote host, press the L key from the main menu. The listen screen appears.

The screen text that follows illustrates a sample completed listen screen. In the sample, 2000 has been specified as the local port. You need not (but can) specify a remote host.

```
LISTEN: Press GO to execute, or CANCEL to cancel

    [Local Port]  2000
    [Remote Host]
    [Remote Port]
```

The Listen command issues a TCP passive open to a specified or unspecified remote host. That is, Listen connections are matched with active opens from remote hosts. When you issue a Listen command, TELNET "listens" for an

incoming call. The session remains inactive until a connection is established or until you abort the Listen command.

[Local Port]
The local (or source) port. It must be numeric and within the range 1024–65535. The default, however, is the standard TELNET listen port 23. This default allows operation with those hosts that do not allow the user to request other ports.

[Remote Host]
The incoming remote host at which the call originates. Complete this field if you want TELNET to accept a call only from the specified host. If you leave this field blank, TELNET will accept a call from any remote host. ⟨remote host⟩ must be alphanumeric and defined in the TCP configuration parameter file.

[Remote Port]
The incoming remote host port at which the call originates. Complete this field if you want TELNET to accept a call only from the specified port on the specified remote host. If you leave this field blank, TELNET will accept a call from any port on the specified remote host. It must be numeric and within the range 0–65535.

Negotiating TELNET Options

You begin option negotiations with one of the following commands: Will, Won't, Do, Don't. Press the corresponding key from the main menu to select the command. A screen appears that lists available TELNET options.

The basic strategy for the use of options is to have either party (or both) initiate a request that some option take effect. The other party may then either accept or reject the request. If the request is accepted the option immediately takes effect; if it is rejected the option remains as specified for an NVT.

Note: Typically, the TELNET program and server processes automatically negotiate options when a connection is established. You do not need to get involved in that process.

Will ⟨option⟩ is sent by either party to indicate that party's desire (offer) to begin performing ⟨option⟩. Do ⟨option⟩ and Don't ⟨option⟩ are its positive and negative acknowledgments.

Won't ⟨option⟩ is sent to indicate that party's desire (offer) to stop performing ⟨option⟩. Don't ⟨option⟩ is its positive acknowledgment.

Do ⟨option⟩ is sent to request that the other party begin performing ⟨option⟩. Will ⟨option⟩ and Won't ⟨option⟩ are the positive and negative replies.

Don't ⟨option⟩ is sent to request that the other party stop performing ⟨option⟩. Won't ⟨option⟩ is the positive reply.

Since all parties must support the NVT, which is what is left when no options are enabled, a party may refuse a request to enable (Do/Will) but must never refuse a request to disable some option (Don't/Won't).

For example, to request the remote TELNET to provide echoing (this option is described later) you send the request *Do Echo*. If the remote TELNET agrees to echo, it replies *Will Echo*; or it may reject your request with *Won't Echo*.

If the remote TELNET wants to stop echoing, it may send you *Won't Echo*. Upon receiving this request, you respond *Don't Echo* to allow the remote to stop echoing.

The following table describes the Do, Don't, Will, and Won't commands as they are used with the TELNET options. The subsections that follow the table give detailed descriptions of the options.

Option	Do	Don't	Will	Won't
Transmit Binary	Request for host to send data as 8–bit binary bytes	Demand for host to stop sending data as 8–bit binary bytes	Offer to send data as 8–bit binary bytes	Refusal to send data as 8–bit binary bytes
Echo	Request for host to echo received data	Demand for host to stop echoing received data	Offer to echo data received back to host	Refusal to echo data received back to host
Suppress Go–Ahead	Request for host to suppress go–heads (GAs) in the data stream	Demand for host to transmit GAs.	Offer to suppress GAs in the data stream	Refusal to suppress GAs in the data stream
Give Status	Request for host to send its perceived option status	Demand for host to stop sending its perceived option status	Offer to send host your perceived option status	Refusal to send host your perceived option status
Timing Mark	Request for a synchronized connection	Not implemented	Offer to establish a synchonized connection	Not implemented
Extended Options List	Not implemented	Demand for host to stop negotiating options beyond the standard 0–255 option values	Offer to negotiate options beyond the standard 0–255 option values	Refusal to negotiate options beyond the standard 0–255 option values

TELNET Options

A host may implement its TELNET to support the NVT only and reject any option request that it does not understand. TELNET supports the following options:

● Binary

● Suppress Go-Ahead

● Echo

● Give Status

● Timing Mark

● Extended–Options List

A description of each option follows.

Binary

The standard NVT transfers data as seven–bit ASCII characters in an eight–bit field; that is, the most significant bit is always zero. The binary transmission option allows a party to transmit data that the receiver interprets as eight–bit bytes of binary data. It is sometimes useful to have a binary transmission path within TELNET without having to use one of the more efficient protocols providing binary transmission (such as the File Transfer Protocol).

A TELNET connection is a two–way communication channel. The binary transmission mode must be negotiated separately for each direction of data flow, if desired.

Suppress Go-Ahead

Although a TELNET connection is intrinsically full–duplex, the NVT in default mode is considered to be a half–duplex device operating in line–buffered mode. *Full–duplex* means that data can flow both ways simultaneously between the two ends of the connection. *Half–duplex* means that data flows in only one direction at a time; therefore, the two sides have to take turns using the line. *Line–buffered* means that data entered should be accumulated in the host where it is generated until a complete line is ready for transmission or until some explicit signal to transmit occurs.

To implement a half–duplex environment, there must be a means to control the "turn–the–line–around" process. The TELNET Go–Ahead (GA) command provides a mechanism whereby the remote computer can signal the local computer that it is time for the local user to take over the line and to transmit data. GA signals should be transmitted at those times, and only at those times, when the user should be given control of the terminal.

Assume that you have a connection open in which the remote TELNET does not suppress go–ahead. Characters you enter at the keyboard are buffered locally until you press the RETURN key to signal the transmission. Then TELNET checks to see whether it has received a Go–Ahead command from the remote TELNET. If not, TELNET waits until it receives a Go–Ahead. Meanwhile, you still can enter more data at the keyboard until the 512–character buffer is full. When the buffer is full, data is sent automatically, regardless of the go–ahead availability.

Note that a line is not limited to a 80–character row on the display. When the RETURN key is pressed to trigger transmission, the carriage RETURN–line feed sequence is inserted in the data stream to signal the end of the line.

Now suppose you send a *Do Suppress GA* and receive a *Will Suppress GA* reply from the remote. It means the remote will begin suppressing transmission of GA signals, and you can send data any time you want to (by pressing the RETURN key), without having to wait for GA permission.

While the NVT normally follows a half–duplex protocol with the Suppress GA option disabled, there is no reason why a full–duplex connection (between a full–duplex terminal and a host that handles full–duplex terminals) should be burdened with the go–ahead signals. TELNET automatically negotiates for suppress–go–ahead by both parties when the connection is opened (by sending *Will Suppress GA* and *Do Suppress GA*).

Echo

Typically, all characters you type at the keyboard are echoed locally on your screen. In some highly interactive situations it might be preferable for the remote process (for example, command language intepreter) to control the way the characters are echoed on the local display. The TELNET Echo option allows the two parties to agree that characters typed in at the local keyboard are to be echoed by the remote party.

If you wish the remote site to do the echoing, send a *Do Echo* command. If the remote party agrees, it replies *Will Echo*. Then every character you enter is

transmitted immediately, and the remote TELNET echoes it back to you. TELNET stops local echoing and displays only data received from the remote.

Note that only one side of a connection can echo at a time. Otherwise, if both parties tried to echo characters transmitted by the other host, any character transmitted would be echoed back and forth indefinitely.

Give Status

The Give Status option allows you to verify the current status of options as viewed by the remote TELNET.

To set up the option, send the remote TELNET a *Do Status*. If the remote party responds with *Will Status*, it will transmit status information, either spontaneously or in response to your request. (See "Obtaining Status Information," later in this section, for information on how to request remote status.)

Timing Mark

It is sometimes useful for the user or process at one end of a TELNET connection to be sure that previously transmitted data has been completely processed, discarded, or otherwise disposed of. This option provides a mechanism for doing this. Suppose you discover that you have made a typing error, and would like the remote to throw it away. You can send a *Will Timing Mark* and then begin to type again. Even if the remote refuses (*Don't Timing Mark*) you are at least assured that the remote has received all previous data.

Extended–Options List

The Extended Options List option extends the option list for another 256 options. If you send *Will EXOP* and the remote responds with *Do EXOPL*, you can start using the Extended Options command to negotiate extended options. TELNET does not allow the remote to initiate EXOPL; therefore, the *Do EXOPL* is not implemented and all requests from the remote (*Will EXOPL*) are rejected.

The extended options menu lets you choose options by number, which gives you all the flexibility, but it also assumes that you know exactly what the option numbers mean to the remote TELNET. The screen text that follows illustrates a sample of the extended options menu.

```
EXTENDED OPTIONS
   Enter Action(1 = DO, 0 = DON't), Option(0 to 255)
   Press GO to execute, or CANCEL to cancel
   Action  1
   Option  123
```

Obtaining Information on Option Negotiations

Press V from the main menu to display all option negotiation activities. The screen text that follows illustrates the verbose command screen.

```
VERBOSE: Select a key, or CANCEL to cancel
   0. OFF
   1. ON
```

When the verbose function is on, TELNET reports all option negotiations activities on the screen. The verbose function is off by default when you begin a session. The same command turns the function off.

From the verbose command screen, press 1 to turn the function on. All negotiations that follow will be reported, such as:

Sent	Do Echo
Received	Won't Echo
Received	Will Binary
Sent	Do Binary

Obtaining Status Information

To obtain status information, press S from the TELNET main menu. The screen text that follows illustrates the status menu.

```
STATUS: Select a key, or CANCEL to cancel
   1. Connection status
   2. Options reported by local
   3. Options reported by remote
```

From the menu, you can select one of three functions: connection status, options reported by local TELNET, and options reported by remote TELNET.

Connection Status

The Connection Status command displays the status of the current connection as reported by TCP or the status of last session if there are no current open connections. The screen text that follows illustrates a sample connection status report.

```
STATUS: Press CANCEL to cancel
  Connection Statistics of current session:
     Connection state    Open
     Open Type           Call
     Local host          UNISYS.COM
     Local port          2001
     Remote host         Phila.Com
     Remote port         23
     Bytes sent          314
     Bytes received      590
```

Options Reported by Local TELNET

The Options Reported by Local TELNET command displays the option negotiation status of the current connection. The screen text that follows illustrates a sample local options report.

```
STATUS: Press CANCEL to cancel
  Options in effect as reported by local TELNET:
                          Local     Remote     Negotiating
     Binary               Yes       No         No
     Echo                 No        Yes·       No
     Suppress Go-Ahead    No        No         Yes
     Give Status          Yes       Yes        No
     Timing Mark          No        No         No
     Extended Options List No       No         No
```

The negotiating field indicates that TELNET has sent a Will or Do negotiation but has not yet received a reply.

Options Reported by Remote TELNET

The Options Reported by Remote TELNET command sends a subnegotiation, requesting the remote TELNET to send you its perception of the options status. For this command to work, you must have previously sent a *Do Status* request and the remote TELNET must have sent a *Will Status* in response. The remote

host then sends its option status report, which will be processed and displayed by TELNET. The screen text that follows illustrates a sample remote options report.

```
STATUS: Press CANCEL to cancel
 Options in effect as reported by remote TELNET:
                              Local    Remote    Negotiating
          Binary              No       Yes       No
          Echo                Yes      No        No
          Suppress Go-Ahead   No       No        No
          Give Status         Yes      Yes       No
          Timing Mark         No       No        No
          Extended Options List   No   No        No
```

If the remote TELNET never sent a *Will Status*, indicating its agreement to send you status information, the system displays *Remote won't give status* when you request remote status information.

Note that for the same connection, options reported by the local TELNET and the remote TELNET must match each other. That is, the local options status as viewed by TELNET should be identical with the remote options status as reported by the remote TELNET.

Verifying Remote Host Operation

Press Y from the main menu and then press the GO key to request the remote TELNET to send evidence that it is running.

For example, suppose you are running a program on the remote host that has been unexpectedly "silent" for a long time. The silence may be the result of the remote system's being bogged down because of heavy use. Or it may mean that the remote system crashed. Use the Are–You–There command to find out if the remote TELNET is running at all.

TELNET sends the Are–You–There command to the remote server TELNET. The remote server TELNET generates a response, which might be a simple message or just a beep. TELNET responds with a message such as *Remote: Yes, I'm here.*

Obtaining Host Names

Press X from the main menu to view all remote host names and the corresponding Internet Protocol (IP) addresses that have been set up on your

system by the system administrator. These are the names you can use in the Call request, for example.

Opening a Recording File

Press R from the main menu to specify a file into which every character entered at the keyboard or received from the remote host is to be recorded. If the specified file already exists, it is overwritten; otherwise, a new file is created. You can use this command before you establish a remote connection.

The screen text that follows illustrates a sample record command screen.

```
RECORD: Press GO to execute, or CANCEL to cancel.

    File name
```

Closing a Recording File

Press $ from the main menu to close a recording file. Otherwise, this file is closed automatically when you exit TELNET.

Aborting a TELNET Connection

Press A from the main menu to issue a TCP Abort message for an opening or an established connection. The following message appears:

Abort: Press ⟨GO⟩ to execute, or ⟨CANCEL⟩ to cancel.(SB)Press the GO key to confirm your selection, and TELNET aborts the connection. If you press the CANCEL key to dismiss, TELNET returns to data transfer mode.

The Abort command unilaterally closes the connection; that is, no action from the remote server TELNET is necessary or possible. The Abort command is useful when the connection "hangs" or when the remote server TELNET does not respond to a Hang Up command by closing the connection at its end.

Aborting Output to Your Screen

Press O from the main menu and then press the GO key to stop sending output to your screen from a process running on the server system.

Use this command when, for example, you are connected to a remote host and are executing a program that is producing output that you are not interested in seeing.

TELNET sends the TELNET Abort Output message to the remote server TELNET. The remote server TELNET stops sending any output from the program that you are executing on the remote host. Output resumes when the program ends.

Interrupting an Operation

Press I from the main menu to send a data message containing the TELNET Interrupt–Process command to the remote server TELNET. Depending on the remote server TELNET implementation, the remote system suspends, interrupts, aborts, or terminates the operation of a process that you have invoked on the server host.

Synchronizing the Connection

Press N from the main menu to synchronize the connection. The Synch command issues the TELNET NVT Synch command, a TCP urgent data message containing the TELNET Data Mark command. The remote TELNET discards the received data bytes until it encounters the TELNET Data Mark command.

A TELNET Synch command may be sent by the server TELNET in response to receiving a TELNET Abort Output command. When your workstation receives a Synch command, it displays the message *Connection in synch.*

Closing an Open Connection

Press H from the main menu to close an open connection. Upon hanging up, no more data can be sent to the remote TELNET. However, you can continue to receive data until the remote TELNET issues a close or the connection is aborted.

Exiting TELNET

Press the FINISH key to terminate TELNET if you have no connection open. The program terminates and returns control to the BTOS Executive. If you press the FINISH key when a connection is open, TELNET responds with the error message *please HANGUP or ABORT.*

Programmatic Interface

The TELNET programmatic interface allows user programs written in BTOS Pascal or BTOS C to access the network using TELNET connections. A program may connect to another program, terminal, or timesharing system at a remote host.

The BTOS implementation is a bytestream (sequential access method—SAM) interface that allows TELNET to emulate a conceptual, sequential, character–oriented device called the TELNET bytestream. The user program can open a TELNET connection, then read from and write to it, using simple SAM procedure calls:

● OpenByteStream

● CloseByteStream

● ReadBsRecord, ReadByte, Readbytes

● WriteBsRecord, WriteByte

● CheckPointBs

● ReleaseByteStream

For a general discussion of the sequential access method, refer to the *BTOS II System Reference Manual, Volume 1.* A detailed description of the procedures is contained in the *BTOS II System Procedural Interface Reference Manual, Volumes 1 and 2.*

This subsection consists of a sample program and procedures to link object modules and to create a bytestream TELNET configuration file.

Creating the TELNET Configuration File

In your program you will need to include a bytestream file specification (for example, '[TELNET]&[!sys]⟨Jon⟩SamConfig') in the OpenByteStream procedure call in order to open the TELNET connection. The first part of the specification always has to be stated exactly *'[TELNET]&'*. (The ampersand after the device name is only needed when a file name follows.) The second part specifies the file that contains information about the TELNET connection that you wish to open. The default file name is [Sys]⟨Sys⟩TCP.TELNETConfig.sys. No configuration file is supplied with the product. You must create the file using the following procedure:

Type *TCP/IP Make TelConfig* at the Executive commmand line. The following form appears:

```
Create TELNET bytestream configuration file
  Press GO to proceed, or CANCEL to cancel
   File name
   Open Type
   (C = Call, L = Listen)
```

Be sure to specify the name of the configuration file. Suppose you wish to call a remote host. Specify *C* in the Open Type field.

Press the GO key. The TELNET call screen appears. See "Establishing a Telnet Connection" for a description of how to fill in the parameters.

If you specify *L* in the Open Type field of the previous screen, the listen screen appears instead of the call screen.

After you have specified the parameters in the call screen and pressed the GO key, the following form appears:

```
Node Name: Press GO to execute, or CANCEL to cancel
  [TCP Node Name (Default = Local)]
```

This parameter is the name of the B-Net node where TCP is installed. The default is local node. After you have specified the parameter in the call screen and pressed the GO key, the following form appears:

```
OPTION NEGOTIATION: Select a key, or FINISH to quit
    1. Do
    2. Will
```

If you do not want any option negotiations, press the GO key. The program creates the configuration.

Suppose you want to send a DO command. Press *1* to invoke the do screen.

```
DO: Select a key, or CANCEL to cancel
    1. Transmit Binary
    2. Echo
    3. Suppress Go-Ahead
    4. Give Status
```

Press *3* to select Suppress Go-Ahead. The program returns to the option negotiation screen so that you can select other option negotiations. All option negotiations that you select will be stored in the configuration file and will be sent to the remote host right after a connection is established.

Linking Object Modules

Assume that the sample program is called *Sample.pas*, and the resulting object module is *Sample.obj*. To create a run file named *Sample.run*, the *Sample.obj* module has to be linked with all of the following modules (using the BTOS Link command):

● ⟨Sys⟩TCP.TelnetTxt.obj — the text message file

● ⟨Sys⟩TCP.SamGen.obj — the customized SamGen module

● ⟨Sys⟩TCP.SamTelnet.obj — the SAM TELNET interface module

● ⟨Sys⟩TCP.RqLabl.obj — TCP calls module

Alternately, you can use the Librarian command to create your own library that contains these modules. Refer to the *BTOS II Linker/Librarian Programming Reference Manual* for a description of the Link and Librarian commands.

Sample Program

The following Pascal program opens a connection with a remote terminal (or another program), sends the Help file to it, then closes the connection.

```
(****************************************************************
*                    SAMPLE   PROGRAM                          *
****************************************************************)
PROGRAM Sample(input, output);
CONST   HELP_FILE_NAME  =       '[Sys] <Sys >TCP.Telnet.Help';
        HELP_FILE_LENG  =       25;
        NUL             =       0;
        ERC_OK          =       0;
        MODE_READ       =       #6D72;    (* mr *)
        MODE_MODIFY     =       #6D6D;    (* mm *)

TYPE    BSWAType        =       ARRAY [1..65] OF WORD;
        ErcType         =       WORD;
        POINTER         =       ADS OF BYTE;
VAR     bsTel           :       BSWAType;
        buf             :       ARRAY [1..512] OF WORD;
        telBuf          :       ARRAY [1..512] OF WORD;
        filHandle       :       WORD;
        nRead           :       WORD;
        nWritten        :       WORD;
        lfa             :       INTEGER4;
        fDone           :       BOOLEAN;
        erc             :       ErcType;
        password        :       ARRAY [1..12] OF CHAR;
PROCEDURE CheckErc(erc            : ErcType);        EXTERN;
FUNCTION OpenFile (pFhRet         : POINTER;
                   pbFileSpecs    : POINTER;
                   cbFileSpecs    : WORD;
                   pbPassword     : POINTER;
                   cbPassword     : WORD;
                   mode           : WORD): ErcType; EXTERN;
FUNCTION CloseFile(fh             : WORD): ErcType; EXTERN;

FUNCTION OpenByteStream (pBSWA    : POINTER;
                   pbFileSpecs    : POINTER;
                   cbFileSpecs    : WORD;
                   pbPassword     : POINTER;
                   cbPassword     : WORD;
                   mode           : WORD;
                   pBufferArea    : POINTER;
                   sBufferArea    : WORD): ErcType; EXTERN;
FUNCTION CloseByteStream (pBSWA : POINTER): ErcType; EXTERN;
FUNCTION WriteBsRecord (pBSWA    : POINTER;
                   pb             : POINTER;
                   cb             : WORD;
                   pCbRet         : POINTER): ErcType; EXTERN;
FUNCTION Read   (fh               : WORD;
                 pBufferArea      : POINTER;
                 sBufferMax       : WORD;
```

```
                    lfa             : INTEGER4;
                    psDataRet       : POINTER): ErcType; EXTERN;
(***********************************************************************
                  M A I N     R O U T I N E              *
**** ****************************************************************)
BEGIN
(* Open Telnet bytestream *)
  CheckErc(OpenByteStream(ADS bsTel,
                 ADS ('[TELNET]&[Sys]<Sys>TCP.TELNET.Config'),
                     35,
                     ADS password, 0, MODE_MODIFY,
                     ADS telBuf, 1024));
(* Write to Telnet bytestream *)
  CheckErc(OpenFile(ADS filHandle, ADS HELP_FILE_NAME,
                 HELP_FILE_LENG,
                 ADS password, 0, MODE_READ));   (*Open Help file*)
  fDone := FALSE;
  lfa   := 0;

WHILE NOT fDone DO BEGIN;
  erc := Read(filHandle, ADS buf, 512, lfa, ADS nRead);
  IF erc <> ERC_OK THEN fDone := TRUE;      (*End of Help file*)
  CheckErc(WriteBsRecord(ADS bsTel, ADS buf, nRead, ADS
                    nWritten));
  lfa := lfa + 512;                    (*Write to remote*)
                                       (*TELNET*)
END;

CheckErc(CloseFile(filHandle));             (*Close Help file*)

(* Close Telnet bytestream *)
CheckErc (CloseByteStream(ADS bsTel));

END.
```

TELNET Command Codes

You can include the following command codes in the bytestream sent to the remote system. For a detailed discussion of the use of the protocols, see the TELNET Protocol, MIL-STD 1782, August 1983.

Command	Code	Meaning
SE	240	The end of the subnegotiation parameters
NOP	241	No operation
Data Mark	242	The data stream portion of a Synch command. It should always be used with a TCP urgent notification
Break	243	NVT character BRK
Interrupt Process	244	The function IP
Abort Output	245	The function AO
Are You There	246	The function AYT
Erase Character	247	The function EC
Erase Line	248	The function EL
Go Ahead	249	The GA signal
SB	250	That what follows is a subnegotiation of the indicated options
Will (option)	251	The offer or agreement to implement the indicated option
Won't (option)	252	The refusal to implement or to continue to implement the indicated option
Do (option)	253	The request for the other party to implement or to continue to implement the indicated option
Don't (option)	254	The demand to stop implementing or confirmation that the other party is not expected to continue implementing the desired option

Command	Code	Meaning
IAC	255	The IAC byte

TELNET Option Codes

Option	Code
Transmit-Binary	0
Echo	1
Suppress Go-Ahead	3
Status	5
Timing-Mark	6
Extended Options List	255

Appendix A
BTOS Status Codes and FTP Protocol Codes

This section lists the BTOS status codes returned by the BTOS TCP/IP software. These errors can be recorded in the system log. (See the *BTOS TCP/IP Administration Guide* for information on using the logging function.) This section lists also the protocol-defined error codes returned by the network during the operation of FTP.

TCP and IP BTOS Status Codes

Decimal Value	Meaning
58950	Unable to deinstall in single partition operating system
58951	Program failed an internal check. Contact your Unisys field representative.
58952	Program failed an internal check. Contact your Unisys field representative.
58953	Program failed an internal check. Contact your Unisys field representative.
58954	Program failed an internal check. Contact your Unisys field representative.
58956	Send buffer too large.
58957	Receive message truncated.
58958	No resources for request.

Decimal Value	Meaning
58959	No connection available
58960	No such host name.
58961	No such host address.
58962	Invalid socket for open.
58963	Connection refused by remote host.
58964	No such connection handle.
58965	Program failed an internal check. Contact your Unisys field representative.
58966	Status message that a connection is closed.
58967	Program failed an internal check. Contact your Unisys field representative.
58968	Program failed an internal check. Contact your Unisys field representative.
58969	Program failed an internal check. Contact your Unisys field representative.
58970	Connection aborted by remote host.
58971	Program failed an internal check. Contact your Unisys field representative.
58972	Program failed an internal check. Contact your Unisys field representative.
58973	Bad Security range.

Decimal Value	Meaning
58974	Program failed an internal check. Contact your Unisys field representative.
58975	Security request not allowed.
58976	Connection already exists.

TELNET Status Codes

Decimal Value	Meaning
59060	Transport Service is not running.
59061	SAM TELNET configuration file not available.
59062	Help file not available.
59063	Received message of invalid size.
59064	System configuration file not available.
59065	Options already in effect.
59066	Received invalid TELNET command.
59067	Received invalid option negotiation.
59068	Subnegotiation too long.
59069	Subnegotiation too short.
59070	Invalid Subnegotation.
59071	No connection.
59072	Open failure.
59073	Close failure.
59074	Abort failure.

File Transfer Protocol Status Codes

Decimal Value	Meaning
110	Restart marker reply. In this case the text is exact and is not left to the particular implementation. It must be sent as follows: MARK yyyy = mmmm Where *yyyy* is the User–process data stream marker and *mmmm* is the Server's equivalent marker. Note the spaces betweem the markers and the equal sign.
119	Terminal is not available. Will try mailbox.
120	Service ready in **nnn** minutes.
125	Data connection already open. Transfer starting.
150	File status okay. About to open data connection.
151	User not local. Will forward to ⟨**user**⟩@⟨**host**⟩.
152	User unknown. Mail will be forwarded by the operator.
200	Command okay.
202	Command not implemented—superfluous at this site.
211	System status or system help reply.
212	Directory status.
213	File status.
214	Help message on how to use the server or the meaning of a particular non-standard command. This reply is useful only to the human user.
215	⟨**scheme**⟩ is the preferred scheme.
220	Service ready for new user.
221	Service closing TELNET connection. Logged out if appropriate.

Decimal Value	Meaning
225	Data connection open and there is no transfer in progress.
226	Closing data connection. Requested file action successful.
227	Entering passive mode. h1, h2, h3, h4, p1, p2
230	User logged in—proceed.
250	Requested file action completed successfully.
331	User name okay—need password.
332	Need account for login.
350	Requested file action pending further information.
354	Start mail input—end with ⟨CR⟩⟨LF⟩.⟨CR⟩⟨LF⟩
421	Service not available—closing TELNET connection. This may be a reply to any command if the service knows it must shut down.
425	Cannot open data connection.
426	Connection closed—transfer aborted.
450	File busy.
451	Local error in processing.
452	Insufficient storage space on system.
500	Syntax error—command unrecognized.
501	Syntax error in parameters or arguments.
502	Command not implemented.
503	Bad sequence of commands.
504	Command not implemented for that parameter.
530	Not logged in.
532	Need account for storing files.

Decimal Value	Meaning
550	File not found or access denied.
551	Page type unknown.
552	Exceeded storage allocation for current directory dataset.
553	File name not allowed.

Appendix B
Queue Manager Information

The following information pertains to the queue management operations described in the *BTOS II System Reference Manual Volume 1* and specifically to the queue management procedures described in the *BTOS II System Procedural Interface Manual, Volumes 1 and 2*. The queue management procedures allow you to write a program to accomplish the same tasks as you can accomplish using the keyboard/screen interface of the FTP Client (see the description of bFunctionCode that follows for a list of the operations you can perform).

Queue Entry Name and Queue Type

The queue entry name (pbQueueName) is *FTPQueue,*and its size (cbQueueName) is *eight*. The queue type (queueType) is *100*.

During Software Installation, this entry is added to the queue index file to create the FTP queue: FTPQueue/[sys]〈sys〉FTPQ.Queue/1/100.

Modify only the filespec portion of this entry, that is, [Sys]〈Sys〉FTPQ.Queue.

Note: You cannot send files to the FTP queue from the Executive. For example, you cannot Copy a file to the FTP queue.

Queue Entry Data Structure

The following is the queue entry data structure (pEntry, sEntry):

Offset	Field	Size (Bytes)
0	bFunctionCode	1
1	cbSrcFile	1
2	cbSrcUser	1
3	cbSrcPassword	1
4	cbSrcAccount	1
5	cbDestFile	1
6	cbDestUser	1
7	cbDestPassword	1
8	cbDestAccount	1
9	cbOriginUser	1
10	rgbSrcFile	78
88	rgbSrcHost	4
92	rgbSrcUser	30
122	rgbSrcPassword	12
134	rgbSrcAccount	10
144	rgbDestFile	78
22	rgbDestHost	4
226	rgbDestUser	30
256	rgbDestPassword	12
268	rgbDestAccount	10
278	bType	1
279	bStructure	1
280	bMode	1

Offset	Field	Size (Bytes)
281	bFormat	1
282	rgbOriginUser	30
312	cbStatus	1
313	bStatus	1
314	erc	1
315	cbTranNum	1
316	wTranNum	2
318	cbDataFile	1
319	rgbDataFile	30
349	cbIndexFile	1
350	rgbIndexFile	30

bFunctionCode Values

1	Used as a key for special queue entries
2	File information listing
3	Directory listing
4	Copy a file
5	Append a file
6	Delete a file
7	Rename a file
8	Send an ISAM data set
9	Retrieve an ISAM data set
10	Delete an ISAM data set
11	Rename an ISAM data set

rgbSrcFile	The source file name, which can be up to 78 characters
rgbSrcHost	The 4-byte hexidecimal host address in which the source file is found
rgbSrcUser	The valid user name for accessing the source file
rgbSrcPassword	The valid password that allows the user to access the source file
rgbSrcAccount	The valid account number that allows the user to access the source file
rgbDestFile	The destination file name, which can be up to 78 characters
rgbDestHost	The 4-byte hexidecimal host address in which the source file is found
rgbDestUser	The valid user name for accessing the destination file
rgbDestPassword	The valid password that allows the user to access the destination file
rgbDestAccount	The valid account number that allows the user to access the source file
bType	The three transmission types. FTP supports three: Specify *A* for ASCII, *E* for EBCDIC, or *I* for IMAGE.
bStructure	The structure. FTP supports two. Specify *F* for file structure or *R* for record structure
bMode	The mode. FTP supports only one: Specify *S* for stream

bFormat	The format. FTP supports only one format for ASCII or EBCDIC transmission:
	Specify N for nonprint
bPriority	The priority (0 to 9, with 0 the highest) assigned to the request by the local user. This is the priority at which the entry is placed in the queue entry file
cbStatus	The size in bytes of the bStatus field. It is always 1
bStatus	The status of the transaction. It can be queued, active, server, done, or abort
erc	The completion code of the transaction
cbTranNum	The size in bytes of the wTranNum field. It is always 2
wTranNum	The transaction number assigned by FTP for each transaction requested by the local user. It can be used to abort the request
rgbOriginUser	The user who initiates an FTP utility (such as copy or delete)
cbDataFile	The size in bytes of a temporary file name. This file is the actual file that will be sent to the remote host
rgbDataFile	The name of a temporary file
cbIndexFile	The size in bytes of a temporary file name. This file contains an image of the index file and is used only during an ISAM send function
rgbIndexFile	The name of a temporary index file

Appendix C
Modifying Samgen.Asm and Samgen.Mdf

You can customize your Samgen.Asm and Samgen.Mdf files to operate with the
TELNET bytestream interface by adding the following italic lines:

SAMGen.Asm Modifications

```
   .
   .
   .
;  SAMGEN.ASM  -  CTOS.Lib version ONLY.
;
;  MODS -
;  7/18/84  JJK  Add comments to DevDepProc's
;  12/2/84  JJK  Add mode-modify disk entry
;  02/23/85  DC  Rename "tagProc" to "tagProcs", give it a fifth
;                procedure name argument for SetImageMode.  Old
;                name remains as is to support existing SAMGEN
;                files.
;$ INCLUDE ([sys]<Telnet>samgen.mdf)
%Init

%DeviceOpen([TELNET],OpenByteStreamTel)
%DeviceOpen([Kbd],OpenByteStreamK)
%DeviceOpen([Vid],OpenByteStreamVid)
%DeviceOpen([Lpt],OpenByteStgal)

%tagProcs(tagTELNETWrite,FillBufIllegal,FlushBufferTEL,
CheckPointBsTEL,ReleaseByteStreamTEL,SetImageModeIllegal)
%tagProcs(tagTELNETModify,FillBufferTEL,FlushBufferTEL,
CheckPointBsTEL,ReleaseByteStreamTEL,SetImageModeIllegal)
   .
   .
   .
```

SAMGen.Mdf Modifications

```
        .
        .
        .
    %CheckDef(tag8251Write)
    %CheckDef(tag8251Modify)
    %CheckDef(tagPtr8251Write)
    %CheckDef(tagSplWrite)
    %CheckDef(tagGpamWrite)
    %CheckDef(tagTelnetRead)
    %CheckDef(tagTelnetWrite)
    %CheckDef(tagTelnetModify)
    PUBLIC  bsVid
    EVEN

        .
        .
        .
```

Bibliography

BTOS B-NET II Administration and Operations Guide (relative to release 1.0 or higher)

BTOS Context Manager II Installation and Configuration Guide (relative to release 3.0 or higher)

BTOS Hardware Installation Guide

BTOS Hardware Network Planning and Installation Guide

BTOS Indexed Sequential Access Method (ISAM) Operations Reference Manual (relative to release 8.0 or higher)

BTOS Intelligent Data Communications Module Systems Software (IDMSS) Operations Guide (relative to release 3.0 or higher)

BTOS OFIS Writer Operations Guide, 2 vols. (relative to release 1.0 or higher)

BTOS OFIS Mail Administration Guide (relative to release 1.2 or higher)

BTOS OFIS Mail Operations Guide (relative to release 1.2 or higher)

BTOS TCP/IP FTP and TELNET Operations and Programming Guide (relative to release 1.0 or higher)

BTOS X.25 Gateway Operations and Programming Guide (relative to release 9.0 or higher)

BTOS II Editor Operations Guide (relative to release 1.0)

BTOS II Standard Software Operations Guide (relative to release 1.0 or higher)

BTOS II System Reference Manual, 2 vols. (relative to release 1.0 or higher)

BTOS II System Procedural Interface Reference Manual, 2 vols. (relative to release 1.0 or higher)

BTOS II Language Development Linker and Librarian Programming Guide (relative to release 1.0)

BTOS Pascal Compiler Programming Reference Manual (relative to release 6.0 or higher)

BTOS C Compiler Programming Reference Manual (relative to release 1.1 or higher)

BTOS II System Status Codes Reference (relative to release 1.0 or higher)

XE500 BTOS Administration Guide (relative to release 7.1 or higher)

XE500 Installation and Implementation Guide (relative to release 7.1 or higher)

XE500 BTOS Operations Guide (relative to release 7.1 or higher)

The Military - Standard documents are available from the Naval Publications and Forms Center, 5801 Tabor Ave., Philadelphia, PA 19120-5099.

The RFC documents are available from DDN Information Center, SRI International - EJ291, 333 Ravenswood Dr., Menlo Park, CA 94025.

File Transfer Protocol. MIL-STD 1780, September 1983.

Internet Protocol. MIL-STD 1777, August 1983.

Simple Mail Transfer Protocol. MIL-STD 1781, August 1983.

TELNET Protocol. MIL-STD 1782, August 1983.

Transmission Control Protocol. MIL-STD 1778, August 1983.

Internet Protocol RFC (Request for Comment) 791

Internet Control Message (ICMP) RFC 792.

Address Resolution Protocol (ARP) RFC 826.

Transmission Control Protocol RFC 793.

Simple Mail Transfer Protocol RFC 821.

File Transfer Protocol RFC 959.

TELNET Protocol RFC 854.

Glossary

A

Abort command

A TELNET command used to unilaterally close a connection. It is particularly useful when the connection "hangs" or when the remote TELNET does not respond to a Hang Up command by closing the connection at its end.

Abort Output command

A TELNET command used to tell the server system to stop sending output from a particular program to your screen. Output resumes when the program ends.

American Standard Code for Information Interchange (ASCII)

A character set code consisting of 7-bit coded characters (8 bits including parity check) for information exchange among data processing systems, data communications systems, and related equipment.

Are You There command

A TELNET command used to verify operation of the remote host.

ASCII

See American Standard Code for Information Interchange.

C

Call command

A TELNET command used to establish an active connection with a remote system. In order for data transmission to take place, the remote system must issue a Listen command. *See also* Listen command.

command mode

The TELNET mode you use to issue commands. When the system is in command mode, the screen displays the TELNET main menu and followed by any additional screens necessary to execute a command. When the command has been completed, the system reverts back to data mode. *See also* data mode.

D

data mode

The TELNET mode you use to transmit data. When the system is in data mode, characters you type are sent to the remote system, and characters received from the remote system are displayed on the screen.

data transfer process

A software process that manages the establishment of a connection for the transfer of data to a remote host.

DDN

See Defense Data Network.

Defense Data Network

A global data communications system that serves users within the Department of Defense. It allows the sharing of information and other resources between subscriber computers.

Do command

A TELNET command used in option negotiations. *Do ⟨option⟩* requests the remote TELNET to implement that option.

Don't command

A TELNET command used in option negotiations. *Don't ⟨option⟩* demands the remote TELNET to discontinue the implementation of that option.

DTP

See data transfer process.

E

EBCDIC

See Extended Binary-Coded Decimal Interchange Code.

Echo

A TELNET option in which a remote process sends back and displays on the local screen all data it receives from the local TELNET.

Extended Binary-Coded Decimal Interchange Code (EBCDIC)

A character set code consisting of 8-bit coded characters. It is the usual code generated by synchronous IBM devices.

Extended Options command

A TELNET command that extends the number of negotiable options to 256. It

can be used only if the remote host has previously sent the local TELNET a Will EXOPL command.

F

File Transfer Protocol

A file manipulation service implemented on BTOS workstations using the Military Standard File Transfer Protocol. FTP allows the exchange of files with multivendor computer systems, shielding the user from host-dependent variations in protocol.

full duplex

A mode of data transmission situation in which data flows both ways simultaneously between the two ends of a connection.

FTP

See File Transfer Protocol.

G

GA

See go-ahead.

Give Status

A TELNET option that allows you to verify, using the Status command, the current status of options as perceived by the remote TELNET.

go-ahead (GA)

A signal sent by a remote host when it is time for the local user to take over the line and transmit data.

H

half duplex

A mode of data transmission in which data flows in only one direction at a time between the two ends in a connection, so that the two sides have to take turns using the line.

Hang Up command

A TELNET command used to terminate an open connection. Once you issue the command, your system can no longer send data to the remote system, although you will continue to receive data until the remote issues a Hang Up command or the connection is aborted.

host name
> The name of an individual computer system on the TCP/IP network. Each site in the network is identified by a mnemonic name to simplify mail and file delivery and to eliminate confusion in overall network operations.

Host Names command
> A TELNET command that lists all remote host names and the corresponding Internet Protocol addresses that have been set up on the system.

I

Indexed Sequential Access Method (ISAM)
> A method of accessing fixed-length BTOS records identified by multiple keys and stored in disk files.

Interrupt Process command
> A TELNET command used to suspend, interrupt, abort, or terminate an operation that you have invoked on the remote system.

ISAM
> *See* Indexed Sequential Access Method.

ISAM command
> An FTP command used to send, retrieve, delete, or rename files between two BTOS systems.

L

Listen command
> A TELNET command used to establish a passive connection to a remote system. In order for data transmission to take place, the remote system must issue a Call command. *See also* Call command.

N

negotiated options
> The TELNET options available in addition to those for an NVT. You use the Do, Don't, Will, and Won't commands to negotiate with the remote system for implementation of these options.

network virtual terminal (NVT)
> An asynchronous, ASCII character-oriented, scroll-mode terminal. A BTOS

workstation running TELNET is emulating an NVT, although additional options may enhance its performance. *See also* negotiated options.

NVT

See network virtual terminal.

P

PI

See protocol interpreter.

Postcard

In SMTP, a short message created in SMTP and sent to another user on the network. *Contrast with* mail file.

protocol interpreter (PI)

A software module responsible for the exchange of protocol commands that support a data transfer.

R

Record command

A TELNET command used to open a recording file. *See also* recording file.

recording file

In TELNET, a file into which is entered every keystroke you type or every character received from the remote system. You create the file using the Record command.

S

Status command

A TELNET command used to display information about your connection status, the options in effect as perceived by the local TELNET, or the options in effect as perceived by the remote TELNET.

status frame

The upper portion of the TELNET screen, which displays information about the connection.

Stop Record command

A TELNET command that closes a recording file. *See* recording file.

Suppress Go-Ahead
> A TELNET option used to suppress transmission of go-ahead signals, allowing you to send data any time you want to, without waiting for go-ahead permission.

Synch command
> A TELNET command used to synchronize the connection.

T

TCP/IP
> *See* Transmission Control Protocol/Internet Protocol.

TELNET
> A protocol that allows data communication with remote computer systems in the TCP/IP network.

Timing Mark
> A TELNET option used to ensure that previously transmitted data has been processed, discarded, or otherwise disposed of.

transaction number
> A unique number FTP assigns to a user request at the time the request is made. It allows you to track the status of your requests and abort any unwanted transactions.

Transmission Control Protocol/Internet Protocol (TCP/IP)
> The two network protocols over which run FTP and TELNET. Transmission Control Protocol is responsible for connection establishment, maintenance, data verification, and relinquishment. Internet Protocol supports the TCP protocol by sending and receiving network datagrams.

Transmit Binary
> A TELNET option in which data is transmitted as 8-bit binary bytes.

V

Verbose command
> A TELNET command that displays the details of all option negotiation activities.

W

Will command

A TELNET command used in option negotiations. *Will ⟨option⟩* is an offer or agreement to implement that option.

Won't command

A TELNET command used in option negotiations. *Won't ⟨option⟩* is a refusal to implement a particular option.

Index

A

Abort command (FTP), 2–7, 2–19
Abort command (TELNET), 3–9,
      3–30, glossary-1
Abort Output command (TELNET),
      3-9, 3–22, glossary-1
American Standard Code for
      Information Interchange
      (ASCII), 2–10, glossary-1
Append command (FTP), 2–6, 2–12
Are You There command (TELNET),
      3–9, glossary-1
ASCII (*See* American Standard Code
      for Information Interchange)
attention key, 3–4

B

B–Net, 2–3
binary option, 3–15, 3-16
break key, 3–4

C

C programming language, 3–24
Call command (TELNET), 3–9, 3–10,
      glossary–1 (*See also* Listen
      command)
chat mode, 3–3,

commands
    installation, 1–3
    FTP, 2–6
    TELNET, 3-9, 3-29
command mode, glossary–1 (*See also*
      data mode)
connection (TELNET), 3–10, 3–23
copy command (FTP), 2–6, 2–8

D

data mode, glossary–2
data transfer process, glossary–2
DDN (*See* Defense Data Network)
Defense Data Network, glossary–2
Delete command (FTP), 2–6, 2–14
Directory List command (FTP), 2–6,
      2–15
Do command (TELNET), 3–10,
      glossary–2
Don't command (TELNET), 3–10,
      glossary–2
DTP (*See* data transfer process)

E

EBCDIC (*See* Extended
      Binary–Coded Decimal
      Interchange Code)
Echo, 3–15, 3–16, 3-17, glossary–2
errors
    BTOS status codes, A–1 — A-3
    FTP protocol errors, A–3 — A-5