

TOPS-20  
ARPANET USER UTILITIES GUIDE

This manual describes the TOPS-20 utility programs available to users of the Advanced Research Project Agency Network (ARPANET). The utilities reflect the TCP/IP protocols.

OPERATING SYSTEM AND VERSION:       TOPS-20 (KL MODEL B), V6.0

SOFTWARE AND VERSION:                TELNET Version 6  
                                      FTP Version 6  
                                      TSTATS Version 6  
                                      IPHOST Version 6

Digital Equipment Corporation, Maynard, Massachusetts

First Printing, June 1978  
Updated, November 1979  
Updated, January 1980  
Revised, October 1983

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may only be used or copied in accordance with the terms of such license.

Digital Equipment Corporation assumes no responsibility for the use or reliability of its software on equipment that is not supplied by DIGITAL or its affiliated companies.

Copyright C , 1978, 1980, 1983 by Digital Equipment Corporation.  
All Rights Reserved.

The following are trademarks of Digital Equipment Corporation:

DIGITAL	DECsystem-10	MASSBUS
DEC	DEctape	OMNIBUS
PDP	DIBOL	OS/8
DECUS	EDUSYSTEM	PHA
UNIBUS	FLIP CHIP	RSTS
COMPUTER LABS	FOCAL	RSX
COMTEX	INDAC	TYPESET-8
DDT	LAB-8	TMS-11
DECCOMM	DECSYSTEM-20	TYPESET-11
	DECSYSTEM-2020	TYPESET-20

CHAPTER 1	INTRODUCTION TO THE TOPS-20 ARPANET USER UTILITIES	
1.1	USING ARPANET UTILITIES . . . . .	1-1
1.2	INVOKING THE UTILITIES . . . . .	1-2
1.3	TYPING FILE SPECIFICATIONS . . . . .	1-2
CHAPTER 2	THE TELNET PROGRAM	
2.1	INTRODUCTION . . . . .	2-1
2.2	USERS AND SERVERS . . . . .	2-1
2.3	CREATING A TELNET CONNECTION . . . . .	2-2
2.4	TERMINAL INPUT MODES . . . . .	2-4
2.5	RUNNING TELNET . . . . .	2-4
2.5.1	CONNECT Command . . . . .	2-6
2.5.2	TELNET Ports . . . . .	2-7
2.6	EXITING TELNET . . . . .	2-7
2.7	GETTING HELP WITH TELNET . . . . .	2-8
2.8	TELNET COMMAND CHARACTER . . . . .	2-8
2.9	TELNET COMMANDS . . . . .	2-9
2.10	CHANGING TELNET PARAMETERS . . . . .	2-12
2.11	CREATING A TELNET.CMD FILE . . . . .	2-14
2.12	CREATING A TELNET LOG FILE . . . . .	2-14
2.13	TAKING COMMANDS FROM FILES . . . . .	2-15
2.14	TELNET MESSAGES . . . . .	2-15
CHAPTER 3	THE FILE TRANSFER PROGRAM	
3.1	INTRODUCTION . . . . .	3-1
3.2	USING FTP . . . . .	3-1
3.3	RUNNING FTP . . . . .	3-2
3.4	EXITING FTP . . . . .	3-4
3.5	GETTING HELP WITH FTP . . . . .	3-5
3.6	FTP COMMANDS . . . . .	3-9
3.6.1	Requesting Data Transfers Or Remote File System Service . . . . .	3-9
3.6.2	FTP Information Commands . . . . .	3-15
3.6.3	Controlling FTP Transfer Parameters . . . . .	3-16
3.7	RESPONSE CODES AND MESSAGES . . . . .	3-18
3.7.1	Response-Coding System . . . . .	3-19
CHAPTER 4	TSTATS	
4.1	INTRODUCTION . . . . .	4-1
4.2	RUNNING TSTATS . . . . .	4-1
4.3	INTERPRETING TSTATS . . . . .	4-2
CHAPTER 5	IPHOST	
5.1	INTRODUCTION . . . . .	5-1
5.2	RUNNING IPHOST . . . . .	5-1
5.3	EXITING IPHOST . . . . .	5-1

5.4	GETTING HELP WITH IPHOST . . . . .	5-2
5.5	IPHOST COMMANDS . . . . .	5-2
5.6	IPHOST MESSAGES . . . . .	5-5

## PREFACE

The TOPS-20 ARPANET User Utilities Guide is written for users who communicate with remote sites on the Advanced Research Project Agency Network (ARPANET). Before using this manual, you should be familiar with the information contained in the latest version of Getting Started with TOPS-20, the TOPS-20 User's Guide, and the TOPS-20 Commands Reference Manual.

This manual contains detailed information on the following TOPS-20 ARPANET utility programs: TELNET, FTP, TSTATS, and IPHOST. Information in the manual is current as of the version of the utility stated on the title page of this manual. Tutorial and reference material is available in each chapter to accommodate inexperienced and experienced ARPANET users.

The following conventions are used in the TOPS-20 ARPANET User Utilities Guide:

<RET> Indicates that you press the RETURN key. On some terminals, this key is labeled CR.

<ESC> Indicates that you press the ESCAPE (ESC) key. This key is labeled ALT on some terminals.

<CTRL/X> Indicates that you press the CONTROL (CTRL) key and at the same time press the letter or symbol indicated after the slash (/).

file spec Indicates a file specification.

The current version of the following TOPS-20 documents are referenced in this manual:

TOPS-20 User's Guide

TOPS-20 Monitor Calls User's Guide

## CHAPTER 1

### INTRODUCTION TO THE TOPS-20 ARPANET USER UTILITIES

The TOPS-20 ARPANET user utility programs allow users to connect to remote sites on the Advanced Research Project Agency NETWORK (ARPANET) and to function as local users at the remote site. Once connected to a remote site, you have access to the resources of the remote site. You will be able to create directories; create, edit, and run programs; create and edit files; transfer files and data between the local and the remote site; communicate with users at both the local and remote site; and obtain information about sites and the status of connections between sites.

The following utility programs are described in this manual:

- o The TELNET program, which allows you to communicate with remote sites (Chapter 2)
- o The FTP program, which allows you to transfer files and data between local and remote sites (Chapter 3)
- o The TSTATS program, which allows you to check the status of host connections (Chapter 4)
- o The IPHOST program, which allows you to get information about hosts. IPHOST also allows an operator to cycle the ARPANET. (Chapter 5)

#### 1.1 USING ARPANET UTILITIES

Three manuals, Getting Started with TOPS-20, the TOPS-20 User's Guide, and the TOPS-20 Commands Reference Manual, describe the commands that are needed to use the TOPS-20 operating system. There are commands, however, that are specific to ARPANET users. The commands needed to use the TOPS-20 ARPANET utilities are described in this manual.

## 1.2 INVOKING THE UTILITIES

To invoke the ARPANET utilities, you need to be logged in to your system. Type the name of the program after the TOPS-20 prompt (@) and press the RETURN key. This invokes the utility program. The utility then prompts you for input. The general format for invoking utilities is:

```
@ Utility name <RET>
```

```
Utility prompt
```

## 1.3 TYPING FILE SPECIFICATIONS

The ARPANET utilities allow you to communicate with hosts that have operating systems that are different from the one you are working on. For this reason, each utility described in this manual documents the format for typing file specifications. The use of recognition and wildcards is also documented on a chapter by chapter basis.

## CHAPTER 2

### THE TELNET PROGRAM

#### 2.1 INTRODUCTION

TELNET is a TOPS-20 utility that provides communication between ARPANET hosts by establishing a connection between a local and a remote host. This connection makes it appear as if you are a local user at the remote site. TELNET accomplishes this by converting terminals attached to a host running TOPS-20 software into a TELNET connection.

#### 2.2 USERS AND SERVERS

ARPANET installations, including Terminal Access Controllers (TACs), are classified as users or servers. These terms are used to distinguish the manner in which resources are shared. The terms are also used to determine the roles of each host during a TELNET connection.

The way resources are shared is established by each ARPANET installation. A server installation shares its resources among all ARPANET sites. A user uses the resources of other sites, but does not share its resources. The classification of a resource as a user may be determined by security requirements or by lack of resources at the site. TACs, for example, have no resources to share. Do not attempt a TELNET connection to a host classified as a user.

The terms user and server also define the function of the two hosts during a TELNET connection. Used in this manner, the term user designates the host that is using the resources of another site. The server is providing the service. For example, two sites, Host A and Host B, are classified as servers. Host A establishes a connection to Host B. For the duration of the TELNET connection, Host A is classified as a user, and Host B acts as a server.

In addition to an understanding of the terms user and server, you must also be aware of the rules governing TELNET connections at your site and at the remote site. Check with your System Manager regarding the rules and times for TELNET connections. Also, become familiar with the requirements of the remote sites to which you make connections.



This is necessary because some sites require that you set up an account with a specific individual at their site. Some sites require a password unique to the individual user, while others allow a group password such as "guest". Some sites restrict TELNET connections to a designated time interval. Reading the ARPANET RESOURCE HANDBOOK, a publication of the Network Information Center (NIC), is helpful. At least one copy is available at each ARPANET site. The handbook is organized by sites and contains such information as:

- o Address and personnel contacts
- o Main function and interest
- o Tables containing host names and addresses, IMP number, status (Server, User, TAC)
- o Service schedules
- o Typical load and peak load time
- o Hardware and operating system

### 2.3 CREATING A TELNET CONNECTION

Creating a satisfactory TELNET connection is basic to TELNET. Therefore, understanding the principles of TELNET protocol negotiations helps you:

- o Request added (more sophisticated) capabilities
- o Request different capabilities
- o Avoid option commands that will not be honored
- o Avoid unnecessary option commands
- o Make efficient use of available commands

TELNET negotiations for assigning capabilities to the TELNET connection and the TELNET connection itself are invisible to you. A completed connection appears to be direct.

Once you type the TELNET command, a set of "current modes" for the TELNET connection is available to you and TELNET. These values, such as local echo, are a set of minimal capabilities that most hosts on the ARPANET possess. The current modes form the basis for negotiations; they are the default modes and exist for the TELNET connection except as specifically changed by TELNET negotiations.

Following the establishment of the connection, TELNET has access to:

- o The default values
- o The actual capabilities of the user terminal
- o The actual capabilities of the server terminal
- o The preferences of user and server
  - As they exist in files (such files are helpful but are optional)
  - As given in TELNET commands by user or server

TELNET considers all the above factors and selects a set of values representing a "best fit" for the TELNET connection. TELNET resolves differences in capabilities and option requests according to the following practices:

- o A server is expected to allow a user all terminal options that the server supplies to local users.
- o Either party to a proposed connection may refuse to change a default mode. Each party must accept the other's refusal.
- o All hosts are expected to use their local terminal in such a way as to appear to be dealing with the TELNET connection. For example, if the TELNET connection has been negotiated as having local echo, both hosts for this connection must be prepared to interpret local echo.
- o Option requests may be made during the TELNET session to change the TELNET connection to suit a changed condition. Option requests may then be negotiated back to the TELNET connection.
- o A rejected option request should never be repeated until something changes. For example, wait until a different task or information is received from the other end.
- o A bidirectional 7- or 8-bit byte is the facility supported by the TOPS-20 user TELNET program. The most significant bit is controlled by the TRANSPARENT command. See Section 2.9 for a discussion of the TRANSPARENT command.

Most TELNET connections are negotiated by the program. Connections to hosts with very different hardware/software facilities, however, may require several option requests.

## 2.4 TERMINAL INPUT MODES

Because TELNET communicates with foreign hosts, TELNET distinguishes between terminal input that is a command to the program, and terminal input that is part of the user-to-server or server-to-user communication. TALK mode and EXTENDED TALK mode serve these functions in TELNET.

When a connection to a TELNET remote host is established, you enter TALK mode. In this mode, you are talking directly to the remote host. You are no longer dealing with the TELNET command protocol. The only exceptions to this rule are:

- o If you are not in TRANSPARENT mode, TALK mode will be affected. TRANSPARENT mode is discussed in Section 2.9.
- o OPAQUE and PAGE mode also affect TALK mode. These modes are discussed in Section 2.10.
- o If you hit the escape character, you are automatically returned to Command level.

TALK mode is the only mode in which many of the TELNET commands and arguments are valid. Command descriptions note whether or not commands are valid in TALK mode.

EXTENDED TALK mode allows you to return to the local host to perform certain tasks, such as logging file usage or renegotiating TELNET parameters, without interfering with the terminal function you are performing at the remote host. Once a command has been completed in EXTENDED TALK mode, you are returned to TALK mode. Enabling the EXTENDED TALK function of TELNET is discussed in Section 2.9.

Generally, EXTENDED TALK mode is entered from TALK mode. If, however, you create a TELNET.CMD file (see Section 2.11 for a discussion of TELNET.CMD files), you can enter EXTENDED TALK mode as soon as you establish a TELNET connection. Command descriptions note whether or not the command is valid in EXTENDED TALK mode.

## 2.5 RUNNING TELNET

To run the TELNET program, you must already be logged in to your system. You can determine if the ARPANET is available by using the INFORMATION (ABOUT) ARPANET-STATUS command. This command gives you information about the ARPANET status:

```
@INFORMATION (ABOUT) ARPANET <RET>
Local ARPANET host name is DEC-TOPS20, Internet address is
10.0.0.79
Network interface is down
Network interface output is disabled
Network service is enabled
Last network interface off transition: 18-Nov-83 14:16:17
```

@

To run TELNET, type TELNET in response to the TOPS-20 prompt. You are then prompted with the TELNET prompt followed by a right angle bracket(>), as in:

```
@TELNET <RET>
  TELNET>
```

Following this prompt, type the name, nickname, or host id of the host you are attempting to connect to. You then receive a message indicating the status of the connection that is being attempted. In the following example, there is a successful connection to the foreign host.

EXAMPLE:

```
Market - LCG's Timesharing System, TOPS-20 Monitor 6(5646)
@LOGIN PAETZOLD (PASSWORD)      <RET>
  Job 29 on TTY233 15-Nov-83 14:02:82
@TELNET <RET>
  TELNET>RADC-TOPS20 <RET>
  Trying...Open
```

In the next example, however, TELNET can not make a successful connection. This example assumes that you are already logged in to your system.

```
EXAMPLE: @TELNET<RET>
          TELNET>SU-SCORE<RET>
          Trying...Host dead, not scheduled up, up on Wednesday at 3:00
P.M. EST
@
```

If you are not able to make a successful connection, either choose a different host or attempt a connection at the time indicated by the TELNET message. For a list of available TELNET hosts, check the ARPANET RESOURCE HANDBOOK.

Once you are connected to an ARPANET site by means of TELNET, typically the herald for that system appears. (Some ARPANET sites have no herald. At this point, it is necessary for you to log in to the remote system. Most ARPANET sites provide any unique information needed to log in to that site. Therefore, be sure to read the herald before you attempt to log in to the foreign host.

The example below shows the procedure for logging in to your own system and then establishing a TELNET connection and logging in to the foreign host.

```
Market - LCG's Timesharing System , TOPS-20 Monitor 6(5646)
@LOGIN PAETZOLD (PASSWORD)      <RET>
  Job 29 on TTY233 15-Nov-83 14:02:38 <RET>
@TELNET <RET>
  TELNET>RADC-TOPS20 <RET>
  Trying...Open
```

```

Rome Air Development Center, TOPS-20 Monitor 6(5646)
@LOGIN PAETZOLD (PASSWORD) <RET>
Job 11 on TTY64, last login 15-Nov-83 13:58:58
15-Nov-83 14:02:50, Up 7:53:27, 9+2 jobs
@

```

### 2.5.1 CONNECT Command

The CONNECT command can be used to establish a TELNET connection to a remote site. The CONNECT command takes the form:

```
@TELNET CONNECT HOST-ID <RET>
```

where HOST-ID is the name of the remote site you want to connect to. Using the CONNECT command eliminates the TELNET> prompt.

The following is an example of the CONNECT command:

```

@TELNET CONNECT DEC-MARLBORO
Trying...Open
Market - LCG'S Timesharing System, TOPS-20 Monitor 6(5646)
@

```

The CONNECT command is also used to establish a connection to a specified host on a designated contact port. For a detailed explanation of ports, see the following section. To use the CONNECT command to establish a connection on a specified port, you must be in EXTENDED TALK mode. The CONNECT command takes the form:

```
CONNECT (TO) INTERNET (HOST) host (ON PORT) port
```

where host is either a recognized host name or a network name followed by four 8-bit decimal numbers specifying the fields of the INTERNET address. The numbers are separated from each other by periods.

The port specification for an INTERNET connection consists of two 8-bit decimal numbers from 1 to 255, separated by a period or a space. The port default is 0,23.

An example of the use of the INTERNET CONNECT command is given below:

```

TELNET>CONNECT (TO) INTERNET (HOST)DEC-MARLBORO (ON PORT)SYSTAT <RET>
Trying...Open
Wed 15-Nov-83 14:11:05 Up 19:58:14
21+12 Jobs Load av 3.04 2.05 1.87
Job      Line   Program  User           Foreign Host
 8       227   OPHIST   REHILL.OPHIST
11       33    EXEC     EIBEN
13       7     HELP     SHERWIN
15       36    MM       TITCOMB
.
.
.

```

Connection closed by foreign host

### 2.5.2 TELNET Ports

TELNET utilizes ports with the CONNECT command when establishing a connection to a particular TELNET task. Once the task has been executed by TELNET, the connection is closed by the remote host.

The argument line for the use of ports takes the form:

```
TELNET>CONNECT (TO) INTERNET (HOST) host name (ON PORT) port
```

where host name is the name of a recognized host name or host id (see the ARPANET Handbook) and port is one of the following:

DAYTIME	DISCARD	ECHO	FINGER	FTP
OLD-TELNET	SYSTAT	TELNET	TEXT	TTYTST

You can use recognition and a question mark for help in typing the correct argument line.

TELNET then establishes a connection to the remote host, performs the task, and closes the connection. The following example shows the use of ports in a TELNET session.

```
TELNET>INTERNET (HOST) DEC-MARLBORO (ON PORT) SYSTAT <RET>
Trying...Open
Wed 15-Nov-83 14:11:05 Up 19:58:14
21+12 Jobs Load av 3.04 2.05 1.87
Job Line Program User Foreign host
8 227 OPHIST REHILL.OPHIST
11 33 EXEC EIBEN
13 7 HELP SHERWIN
15 36 MM TITCOMB
.
.
.
Connection closed by foreign host
```

### 2.6 EXITING TELNET

Two commands are used to exit from TELNET: CLOSE and EXIT.

To exit from TELNET TALK mode using the CLOSE command, type <CTRL/~>C in response to the TELNET command prompt. The <CTRL/~>C is not echoed on your terminal. You are then informed that the connection has been closed:

```
Command: <CTRL/~>C
Connection closed
@
```

The CLOSE (Connection) command closes the network connection to the remote host from EXTENDED TALK mode. Using this command returns you to the TOPS-20 command level. To use this command, type:

```
TELNET>CLOSE <RET>
@
```

The EXIT or Quit (from TELNET) command returns you to command level without closing the network connection. To use this command, type:

```
TELNET>EXIT <RET>
TELNET>
```

## 2.7 GETTING HELP WITH TELNET

There are two ways of getting help in using TELNET. Typing a ? provides information on the single command or message that is currently being displayed on the terminal. After typing the ? and getting help, you are prompted for the next argument needed to complete your command:

```
TELNET>ESCAPE (CHARACTER IS)? escape character in quotes
or ASCII code for character in octal
```

To get information on running the TELNET program, use the HELP command. Typing either HELP or TELNET HELP in response to the @ prompt prints a brief help text on how to use TELNET.

## 2.8 TELNET COMMAND CHARACTER

The TELNET protocol sends data to the foreign host. This data is not interpreted by TELNET. It is only transmitted across the network.

The ^^character, which is the ASCII character 36, is the basic command character TELNET uses to control the transmission of data. On a VT100 series terminal, this character is entered as <CTRL/~>.

The <CTRL/~> is the TELNET command character. This character interacts with the TELNET program. The command character is followed by an additional character to specify the action to be taken. If the command character is followed by itself, as in <CTRL/~><CTRL/~>, the command character is transmitted to the remote host.

The command character is not echoed on your terminal. However, in this manual, command characters are shown in the examples.

## 2.9 TELNET COMMANDS

TELNET commands are specified by typing the TELNET command character <CTRL/~> followed by another character that specifies the action to be taken, as in <CTRL/~>P. The command sequence that you type is not echoed on your terminal.

Some of the TELNET commands are only legal in TALK mode, while others can only be used in EXTENDED TALK mode. These commands are explained in detail below.

### QUESTION MARK (?) COMMAND:

The Question Mark command, which is input by typing a <CTRL/~>?, gives information about what is needed to complete your current task. The following example shows the use of the ? command:

```
<CTRL/~>?
Type a single character command:
? This message
A Send ATTN
B Send BREAK
C Close connection
O Abort output
P Push to inferior EXEC
S Status of TELNET connection
Q Quit
T Transparent Mode toggle
X Enter extended command mode
Typing the escape character twice sends it to the host.
Command:
```

### ATTN (KEY) COMMAND:

The ATTN command can only be used in TALK mode. Its function is to send the TELNET "interrupt process" protocol sequence for the "Attention" signal. This command is used primarily for MULTICS or IBM systems. To use this command, type <CTRL/~>A in TALK mode, as shown in the following example:

```
<CTRL/~>A
```

### BREAK COMMAND:

The BREAK command can only be used in TALK mode. Use the Break command to send the TELNET "break" sequence and to terminate whatever task is being performed at the remote host. To use this command, type CTRL/~B in TALK mode, as in the following example:

```
<CTRL/~>B
```



## CLOSE (CONNECTION) COMMAND:

As discussed previously, the CLOSE command is one method of exiting from TELNET and closing the ARPANET connection. When you are in TALK mode, typing the <CTRL/~>C returns you to the TELNET command level. When in EXTENDED TALK mode, typing CLOSE in response to the TELNET prompt closes the connection and returns you to TELNET command level. An example of the use of the CLOSE command is given below.

Using the CLOSE command from TALK mode:

```
<CTRL/~>C
Connection closed by foreign host
```

Using the CLOSE Command from EXTENDED TALK mode:

```
TELNET>Close <RET>
Connection closed by foreign host
```

## ABORT (OUTPUT) COMMAND:

The ABORT command is legal only in TALK mode. When you type a <CTRL/~>O, the TELNET "abort output" signal is sent to the remote host and any output from the remote host is halted. To use this command, type:

```
<CTRL/~>O
```

## PUSH (COMMAND LEVEL) COMMAND:

You can use the PUSH command in either TALK or EXTENDED TALK mode. The PUSH command functions in the same manner as it does at TOPS-20 command level. Use this command when you wish to perform a task at your local host. When you have completed the task, typing POP returns you to TELNET TALK mode.

Not only is the PUSH command useful for doing a temporary task at the local host and then returning to TELNET, the command also protects you against accidentally logging out while you still have an active connection. You cannot log out once you have executed the PUSH command.

To use this command in TALK mode, type <CTRL/~>P as in:

```
<CTRL/~>P
```

To use this command in EXTENDED TALK mode, type PUSH in response to the TELNET command prompt, as in:

```
TELNET>PUSH <RET>
```

## STATUS (OF TELNET CONNECTION) COMMAND:

The STATUS command allows you to ask the remote host to send a message indicating that it is still available. This command is only legal in TALK mode. Most often you receive a job/system message that identifies the remote host. The minimum response you receive is a yes.

To use the STATUS command, type <CTRL/~>S in response to the TELNET prompt. Neither the control nor the command character is echoed on your terminal. The most frequent response to this command is listed in the example given below:

```
@<CTRL/~>S <RET>
This is DEC-TOPS20, ARPANET TCP/IP Development System, TOPS-20 Monitor
6(5646)
Connected to host DEC-TOPS20 via Internet
New TELNET protocol
Remote host is echoing
Line editor disabled
GA's are suppressed
Host is not sending binary
Transparent mode disabled
Opaque mode disabled
Page mode disabled
No log file
No TAKE file
Simple talk command mode
Remote host status reply:
  09:38:55 Used 0:00:00.2 in 0:00:12, Load   0.31
```

[NO] TRANSPARENT (MODE) COMMAND:

The TRANSPARENT command controls whether or not the most significant bit of the 8-bit character (also called the edit key) is transmitted to the remote host as an editing character. This command is legal only in TALK mode. In TRANSPARENT mode, the edit key functions as a character modifier at the remote host. When the [NO] TRANSPARENT command is chosen, the edit key is suppressed and cannot be used as a character modifier.

To enable TRANSPARENT mode, type either <CTRL/~>T or TRANSPARENT. To suppress the edit key, use the NO TRANSPARENT command, which you enable by typing a double <CTRL/~>T.

The following example shows the use of the TRANSPARENT command. Remember neither the control character nor the single character is echoed on your terminal.

```
<CTRL/~>T
```

[NO] EXTENDED (COMMAND MODE AUTOMATICALLY WHEN ESCAPE CHARACTER TYPED) COMMAND:

The EXTENDED command controls entry into the EXTENDED TALK mode of TELNET. You can use the EXTENDED command in two ways: to enter

EXTENDED TALK mode immediately upon entering TELNET, or to enter EXTENDED TALK mode from TALK mode.

If you have created a TELNET.CMD file, you can place the EXTENDED command in this file. If you use the EXTENDED command, you bypass TALK mode and automatically enter EXTENDED TALK mode when you make a TELNET connection. The default for the EXTENDED command is the NO EXTENDED command. If the default is used, you enter TALK mode when you enter TELNET.

#### NOTE

If you set EXTENDED in your TELNET.CMD file, you cannot send the escape character <CTRL/~> to the remote host.

The EXTENDED command can also be used from TALK mode. When you are in TALK mode, typing <CTRL/~>X places you in EXTENDED TALK mode. This allows you to perform certain TELNET tasks at your local site without interfering with the task being performed at the remote site.

### 2.10 CHANGING TELNET PARAMETERS

Four commands are available for changing TELNET transmission parameters: ECHO, LINE, OPAQUE, and PAGE. Descriptions of each of these commands are given below.

#### [NO] ECHO (BY REMOTE HOST) COMMAND:

The ECHO command controls which host performs echoing of user input. When the NO ECHO command is enabled, echoing occurs at the local host and is under TELNET protocol control. If the ECHO command is specified, the remote host handles echoing. The ECHO parameter is part of TELNET negotiations and must be agreed to by the remote host. If the remote host does not agree to handle echoing, TELNET reverts back to local echo. On most occasions, TELNET negotiates remote echoing when establishing the TELNET connection.

Once a connection is in effect, ECHO or NO ECHO can be used only in TALK mode. To use the ECHO command, type:

```
@TELNET>ECHO<RET>
```

#### [NO] LINE (EDITOR FOR LOCAL ECHO CONNECTIONS) COMMAND:

The LINE command controls the use of the TOPS-20 editing characters rubout, CTRL/R, CTRL/W, and CTRL/U, as well as the manner in which text is sent to the remote host. The LINE command can be used only when local echoing is in effect.

When the LINE command is enabled, the rubout, CTRL/R, CTRL/W, and CTRL/U characters can be used. Also, no text is sent to the remote host until a carriage return or a line feed is typed. The LINE command is useful for connections to a line-oriented system.

To use the LINE command, type LINE in response to the TELNET prompt:

```
@TELNET>LINE <RET>
```

#### [NO] OPAQUE (MODE) COMMAND:

The OPAQUE command controls the degree of local terminal handling that is done during a TELNET session. When the OPAQUE command is enabled, the local host continues to handle such functions as padding and keeping track of the cursor's position. The OPAQUE command is generally used with terminals whose capabilities are different than the terminal you are working on.

In addition, when the OPAQUE command is enabled, nulls are ignored; and character conversion may take place. You will, in most instances, not be able to use a display editor at the remote host. The OPAQUE and PAGE commands must both be enabled in order to have automatic holding at the end of the terminal screen.

The NO OPAQUE command is the TELNET default. The NO OPAQUE command specifies that the remote host will handle all terminal functions. No local terminal handling is done.

To use the OPAQUE command, type OPAQUE in response to the TELNET prompt, as in the following example:

```
TELNET>OPAQUE<RET>
```

#### [NO] PAGE (MODE) COMMAND:

The PAGE command controls whether or not the local host handles terminal paging. When the PAGE command is enabled, the local host handles paging using the CTRL/Q and CTRL/S characters. There is no way to send these characters to the remote host. If you enable the PAGE command, you should also enable the OPAQUE command, since these commands, used jointly, tell the local host how many lines of text have been input.

Use the PAGE command when you need to have paging in effect and the remote host will not or can not handle this task. For example, VT100 terminals in smooth scroll mode require faster handling than the networks allow.

NO PAGE is the TELNET default command. When the NO PAGE command is in effect, the CTRL/Q and CTRL/S characters are passed to the remote host.

To use the PAGE command, type PAGE in response to the TELNET prompt,

as in the following example:

```
TELNET>PAGE<RET>
```

### 2.11 CREATING A TELNET.CMD FILE

A TELNET.CMD file, which is automatically read when you invoke TELNET, is used to change TELNET's default behavior. If you want your terminal to have command characters from a previous version of TELNET, you can place these commands in your TELNET.CMD file.

For example, if you want Y for the intercept character, immediate entry into EXTENDED TALK mode, and OPAQUE mode (see Section 2.10), create a TELNET.CMD file that contains the following:

```
ESCAPE 031 ;or a quoted control-y
EXTENDED
OPAQUE
PAGE
```

To determine how to define a command in your TELNET.CMD file, consult the latest version of the TOPS-20 User's Guide.

### 2.12 CREATING A TELNET LOG FILE

To create a file that contains a transcript of your TELNET session, use the LOG command. The command takes the form:

```
LOG (FILE IS) log file name
```

Where log file name is the name you choose for your file. The default file name for a log file is TELNET.LOG.

TELNET's default is not to keep a log. If you want to have a log, you must use the LOG command. Note that the LOG command only logs information from the foreign host.

A CLOSE command automatically closes the log file. However, when another connection is opened, the log file is reopened for append.

If you decide during your session that you no longer want a log file, you can use the NO LOG command. To use this command, type:

```
@TELNET>NO LOG<RET>
```

This closes your log file.

The following example shows the creation of a log file.

```
@TELNET>LOG (FILE IS)TELNET.TXT <RET>
```

### 2.13 TAKING COMMANDS FROM FILES

TELNET accepts commands from specified input files by using the TAKE command. At the end of the file, TELNET reverts to accepting commands from the terminal. In addition, TELNET automatically reads the TELNET.CMD file at start-up.

To use the TAKE command, type TAKE in response to the TELNET prompt, followed by the name of the input file from which you want commands taken:

```
TELNET>TAKE (Commands from file)input filespec <RET>
```

where input filespec is the name of the file that you want commands taken from.

### 2.14 TELNET MESSAGES

The most common TELNET messages, with their descriptions and suggested user responses, are listed alphabetically below. Some of these messages are informational, describing that conditions are normal. These messages merely confirm that TELNET is doing what you requested. Other messages indicate an error at the remote site. These errors are not user errors. The user has no control over them.

When TELNET is unable to continue with a user command, it outputs one of the error messages listed below. Generally, any TAKE file or command that is in progress is aborted. These errors are usually errors in the user's command line. In the list below, fatal errors are preceded by a question mark (?), and warning messages are preceded by a percent sign (%).

?Abnormal fork termination interrupt

TELNET's network input fork died involuntarily. Because TELNET is supposed to check for all error conditions, this message is not expected to appear.

SUGGESTED USER RESPONSE: Contact the system people at your site if this message appears on the terminal.

?Can't continue

It is not possible to continue TELNET after a fatal error.

SUGGESTED USER RESPONSE: Restart the TELNET program.

?Character doesn't have a control form

You gave the CONTROL command with an argument that cannot be made into a control character.

SUGGESTED USER RESPONSE: Respecify an argument that can be made into a control character and issue the control command.

#### Connection Closed

The user gave a CLOSE command. TELNET is confirming that it has closed the connection.

#### Connection closed by foreign host

The remote host has closed the connection for reasons known to itself, but not to TELNET. For example, many hosts "hang up" the connection when you log out. TELNET takes the same action it does when you give a CLOSE command.

#### Foreign host sent a subnegotiation

The remote host is using the subnegotiation part of the TELNET protocol. Since TELNET does not allow subnegotiations, the host is violating protocol. Some garbage may follow this message. This message is not expected to appear.

#### Insufficient system resources

TOPS-20's network connection tables are full; there is no room for a new connection.

SUGGESTED USER RESPONSE: Try connecting at a later time.

#### ?Invalid ASCII value

The number specified for the escape characters was not between 0 and 177 octal.

SUGGESTED USER RESPONSE: Specify a number between 0 and 177 octal.

#### ?Invalid port number

The number given was out of range for the named field.

SUGGESTED USER RESPONSE: Specify a number that is within the range of the named field.

#### ?Log file error-...more details...

TELNET was unable to reopen the log file when opening the connection. The log file has been turned off, and the connection continues without it.

SUGGESTED USER RESPONSE: Correct the condition indicated in the error message, and reissue a LOG command if you still want a transcript of your session.

#### ?...message...

A miscellaneous nonfatal error occurred. This is usually a user error.

SUGGESTED USER RESPONSE: Correct the error and try the task again.

?...message...,...instruction...at PC=...PC value...

A fatal error has occurred. The PC printed is the location of the error.

SUGGESTED USER RESPONSE: Contact the system people at your site if this problem occurs.

?Must have CONTROL-C capability to run TELNET

The superior fork has denied TELNET the <CTRL/C> capability, which is necessary for TELNET to be able to trap a <CTRL/C>. TELNET needs the capability to perform properly.

SUGGESTED USER RESPONSE: This error message occurs if you try to run TELNET in a batch job. Contact the system people at your site if this occurs and you are not using batch or have not given the SET NO CONTROL-C command to the EXEC. Also, make sure not to run TELNET in batch.

?Not implemented with this protocol

The command the user gave is meaningless on the current connection because the current protocol doesn't support that function.

SUGGESTED USER RESPONSE: Use a command that is implemented by the protocol to which you are connected.

?No such host name-...more details...

TELNET was expecting a host specification, but the input given could not be interpreted. Possibly a host name was misspelled.

SUGGESTED USER RESPONSE: Correct the error and try the command again.

?TAKE command in progress

A command file run by the TAKE command tried to execute another TAKE.

SUGGESTED USER RESPONSE: Remove the second and any additional TAKE commands from the TAKE file.

?Time out

TELNET spent more than 30 seconds trying to do a single network output operation. TELNET assumes that the host at the other end



is hung and aborts the connection.

SUGGESTED USER RESPONSE: Contact the system people at your site or the remote site if this problem occurs repeatedly.

Trying...

The user gave a CONNECT command, and TELNET is attempting to open the connection. TELNET reports the result of the connection attempt with one of the following messages:

- o Host dead...more details...- The remote host is not available at the present time. If TELNET can get any details about the unavailability, it will report them. This may include such things as a message left by the remote host before it went down detailing why it went down and when it is expected to be up again.
- o Insufficient system resources- TOPS-20 network connection tables are full; there is no room for a new connection.
- o Open- TELNET has succeeded in opening a connection to the remote host. The next output on the terminal will probably be some greeting message from the remote host.
- o Refused- The remote host is up, but it is not accepting connections at the present time. This condition is usually temporary; for example, the host might be momentarily unable to accept the connection. Some hosts refuse connections when they are down for system work. TACs always refuse connections if you try to TELNET to them.

?Undefined error

A system call returned an error number that is not in SYSTEM:ERRMES.BIN.

SUGGESTED USER RESPONSE: Contact the system people at your site.

?Unrecognized command-...more details...

TELNET was expecting a command or host specification, but the input given could not be interpreted as either. Possibly a command or host name was misspelled.

SUGGESTED USER RESPONSE: Correct the error in the command line and retry the command.

## CHAPTER 3

### THE FILE TRANSFER PROGRAM

#### 3.1 INTRODUCTION

The ARPANET File Transfer Program (FTP) is a TOPS-20 function-oriented protocol that is used to transfer files between ARPANET hosts. Through controls and specifications, the FTP protocol establishes a common ground for file exchange and manipulation.

The FTP protocol uses the Transmission Control Protocol/Internet Protocol (TCP/IP) to establish a connection. This connection sends FTP commands from the User-FTP to the Server-FTP, and sends FTP replies from the Server-FTP to the User-FTP. The Server program also establishes a data connection for transferring data. Thus, two connections are required: one connection for commands and replies and one connection for transferring data. The data connection, however, is transparent to the user.

The FTP protocol is implemented by two programs: a User program and a Server program. At the local host, you interface with the FTP protocol through the User-FTP program. At the remote host, you are dependent on the Server-FTP program.

The following diagram illustrates a user request for a file at the remote site using the FTP protocol.

#### 3.2 USING FTP

The User-FTP commands are the commands that you use at your terminal. These FTP protocol commands handle communication between processes on different hosts. For this reason, FTP is more transparent and less user interactive, and has more restrictions placed on the way the program implements the protocol.

The recognition feature of TOPS-20 is not implemented for all FTP commands. If you press the escape (altmode) key after typing enough letters to make a command unique, the bell sounds if recognition is not implemented with the command. In this case, you must type the full command.

The FTP protocol may initiate transparent Host-to-Host interactions, such as establishing or refusing connections, on your behalf. Normally, you do not need to know that these transactions are occurring. If you need to act, you receive a message indicating what action you should take.

If you are a system programmer, you should be familiar with the RFC-765 ARPANET TCP/IP File Transfer Protocol Handbook and the TOPS-20 Monitor Calls User's Guide.

User-FTP commands can be classified into four categories:

- o Controlling access to remote hosts
- o Requesting a data transfer or remote file system service
- o Controlling User-FTP locally
- o Specifying data transfer parameters

Sections 3.3 through 3.6 of this manual describe these four categories and the commands that implement the program.

### 3.3 RUNNING FTP

To gain access to the FTP program, type FTP in response to the TOPS-20 Exec prompt. You are then prompted with an asterisk (\*). The \* indicates that the FTP program is ready to accept commands.

To access a remote site and to request data transfers between two sites, you must also connect to the site and log in at the site.

The two commands used to perform these functions are:

- o Connect
- o Login

#### CONNECT COMMAND

The CONNECT command establishes the connection to the remote site. The CONNECT command takes the form:

```
CONNECT (to host)<host-name>
        or <decimal-byte>.<decimal-byte>.<decimal-byte>
           <decimal-byte>
        or <octal-number>
```

where host-name is the official host name or the official host nickname, decimal-byte refers to a four-decimal octet INTERNET host number, and octal-number refers to the host's octal number.

The following example shows a completed FTP CONNECT command.

```
*CONNECT (to host) DEC-MARLBORO <RET>
```

It is recommended that you use the official host name or host nickname. If the host name or nickname is used, the User-FTP program refers to a table of Server characteristics to establish the connection. The connection parameters selected from this table are specific and efficient. If you use the host number, the User-FTP uses whatever parameters were last used for a connection. Therefore, you may need to specify options.

If the CONNECT command is successful, you receive a message indicating that the connection has been established between the local and remote sites. This message is followed by the herald of the remote host.

The following example shows a CONNECT command followed by the response of the remote host.

```
@FTP
*CONNECT (to host) DEC-MARLBORO <RET>
Connection opened
Assuming 36-bit connections, Paged transfers
*<DEC-MARLBORO FTP Server Process 6(4)-7 at Thu 3-Mar-83
08:55-EST
*
```

Once the connection has been established, you are dependent on the Server-FTP conventions. Since parameters may vary widely between ARPANET hosts, you may have to use the \*VERBOSE and \*QUOTE commands to help you at first. (These commands are discussed in Section 3.5.) This should not normally be required.

If your FTP connection is not successful, you receive a message from the remote host indicating the reason. You can either follow the action indicated by the message for that remote host, or you can attempt a connection to another FTP host.

#### LOGIN COMMAND

Having established a connection to a remote host, many sites require that you log in at the remote host before attempting to accomplish a data transfer. The LOGIN command performs this function.

The LOGIN command takes the form:

```
*LOGIN (user-ident) username (password)password (account)account
```

where user-ident or username is the name assigned for your login at the remote host, password is the password required by the remote host, and account is the account required by the remote host.

Since you are dependent on the requirements of the remote host during login at the remote site, you must know the requirements of the host

you are attempting to log in to. All three arguments to the LOGIN command may not be required, or they may be different arguments from those required for login at the local host. For example, TOPS-10 systems require project number and programmer number in the [n,m] as the username parameter. Some hosts require a previously assigned account number or string; some ignore the account parameter. Most remote hosts, however, do require that the LOGIN command be given immediately after a successful connection.

Documentation on login is available in the ARPANET Resource Handbook. If you require additional help in securing FTP service, contact the liaison officer for that site. The ARPANET Resource Handbook contains a list of liaison officers.

An example of a LOGIN command is given below.

```
*LOGIN (user-ident) CPERLMAN (password) (account)341 <RET>
*<User CPERLMAN logged in at Thu 3-Mar-83 08:57-EST, job 13
*
```

### 3.4 EXITING FTP

Three FTP Commands that are used to terminate an FTP connection are:

- o BYE
- o DISCONNECT
- o QUIT OR EXIT

These commands determine whether your FTP connection is closed or maintained while you return to the local host.

#### BYE COMMAND

The BYE command terminates your access to the remote host and closes your connection to the remote site. The BYE command for the TOPS-20 FTP program logs out your job at the remote host, closes the connection, and returns you to FTP command level.

The BYE command takes the form:

```
BYE closes connection
```

There is no argument after the command. Once the BYE command has been executed, you receive a message indicating that the connection has been terminated as in:

```
*BYE <RET>
*<QUIT command received. Goodbye
*
-- Connection has terminated --
```

\*

## NOTE

If you give a second CONNECT command without executing a BYE command, FTP responds with "closing first connection" before the new connection is opened. Only one connection can be active at any one time.

## DISCONNECT COMMAND

The DISCONNECT command aborts the connection to the remote host. The DISCONNECT command takes the form:

```
DISCONNECT from remote host (abort) <RET>
```

where a carriage return is used to confirm the command. Following the execution of this command, you receive a message from the remote host indicating that the connection has been aborted. A complete DISCONNECT command, therefore, appears as:

```
*DISCONNECT <RET>
<QUIT command received. Goodbye.
--Connection has terminated.--
@
```

## EXIT AND QUIT COMMANDS

The EXIT and QUIT commands allow you to return to the TOPS-20 Exec level without terminating the connection to the remote host. To use the EXIT and QUIT commands, type either EXIT or QUIT in response to the FTP prompt. After you type a carriage return, you receive a message indicating that the connection is still open, followed by the TOPS-20 prompt (@). You can then perform a task at the local host and type CONTINUE to return to the FTP program. An example of a completed EXIT command follows. The QUIT command takes the same format.

```
*EXIT <RET>
[The connection is still open.]
@
@CONTINUE <RET>
*
```

### 3.5 GETTING HELP WITH FTP

The User-FTP program has a HELP command available to help you use FTP. The HELP command types information on commands, gives an example of FTP usage, or types current status, according to your response to the HELP command. To use the HELP command, type HELP in response to the FTP \* prompt. You are then prompted with:

```
*HELP <RET>
C, E, S or ?
```

Your response to this prompt dictates what type of help you receive from the FTP program.

If you type C in response to the HELP command you are given a list of available FTP commands. The following example shows the use of the C response to the \*HELP command:

```
*HELP <RET>
C, E, S or ? C <RET>
commands are:
CONNECT to <host-name>
      or
<decimal-byte>.<decimal-byte>.<decimal-byte>.<decimal-byte>
      or #<octal-number>
LOGIN <username> <password> <optional-account>
ACCOUNT <number> or <string>
GET <remote-file> to <local-file>
SEND <local-file> to <remote-file>
MULTIPLE GET/SEND <local-file-group>
      Allows "*", TOPS20/TENEX sites only.
APPEND <local-file> to <remote-file>
RENAME <remote-file> to be <new-remote-file>
DELETE <remote-file>
DIRECTORY of <remote-user>
CWD change working directory to <directory-path>
STATUS of remote system
BYE closes connection
DISCONNECT from remote host (abort)
EXIT returns to EXEC, without closing connection
QUIT returns to EXEC, without closing connection
MODE sets transmission mode to one of (S)
STRUCTURE sets structure of data to one of [F, P]
TYPE sets data type to one of [A, N, I, L n]
ASCII a shorthand for STRU
PAGED a shorthand for STRU P, TYPE L 36
TENEX a shorthand for TYPE I
IMAGE a shorthand for TYPE I
VERBOSE typeout mode
BRIEF typeout mode
QUOTE <arbitrary-FTP-line>
STATISTICS turns on typeout of timing statistics
NOSTATISTICS turns off typeout of timing statistics
RETAIN turns on retention of generation numbers
NO RETAIN turns off retention of generation numbers
```

```
HELP types this information
Type BELL (^G) to abort a file transfer and return to command
level
Type ^O to clear typeout buffer.
Use ^V to quote characters in LOGIN.
```

If you type E in response to the FTP HELP command, you are given an example of what an FTP connection is:

```
*HELP
C, E, S or ? E <RET>
@FTP                               ;call in the subsystem
*BBN                               ;connect to host BBN
*LOG SMITH SECRET 12345            ;declare, name, password, account
                                   ;the password will not be echoed.

*DIR *.MAC <RET>
(to local file)TTY: [confirm]     ;get a partial directory
                                   listing
*GET PROGRAM.MAC <RET>
(to local file)<esc>PROGRAM.MAC   ;must end with carriage return
                                   ;escape causes the same name
                                   to be used
*BYE                               ;request server to terminate
*QUIT
@
```

If you type the S response to the HELP command, you receive the current status of connections, as follows:

```
*HELP
C, E, S or ? S <RET>
DEC-MARLBORO FTP User Process 6(13)-7
The current data transfer parameters are:
  MODE S
  STRU P
  TYPE L 36
NORETAIN generations
A connection is open to host DEC-MARLBORO
The data connection is closed.
*
```

If you type the ? in response to the HELP command, you receive a message indicating which of the responses you should make to this command:

```
*HELP <RET>
C, E, S or ? ? <RET>
Type "C" for a list of commands, "E" for an example of FTP use,
or "S" for current status.
C, E, S or ?
? Type "HELP<RETURN>" for help.
```

Three other FTP commands are useful in getting help in using the User-FTP program. These commands, VERBOSE, BRIEF, and QUOTE, provide more information on user input required to accomplish FTP tasks. These commands are discussed below.

#### VERBOSE COMMAND

The VERBOSE command causes additional informational messages to be displayed on your terminal. These messages, which are usually



transparent during your FTP session, are the User-FTP commands and Server-FTP replies. Included in these messages are the response-coding numbers. The FTP response-coding system is discussed in Section 3.7.1. The VERBOSE command takes the form:

```
VERBOSE <RET>
```

where a carriage return is used to confirm the command. Once this command is executed, you receive additional information at your terminal.

The VERBOSE command is typed prior to the command for which additional information is required. The command is especially useful when you are unfamiliar with a command that you want to use. It is also useful when attempting a connection to a remote host whose procedures you are unfamiliar with. In this case, type VERBOSE prior to attempting to log in to the host.

An example of the VERBOSE command follows:

```
*VERBOSE <RET>
*CONNECT (to host) DEC-MARLBORO <RET>
  Connection opened.
  Assuming 36-bit connections, Paged transfers.
*<220 DEC-MARLBORO FTP Server Process 6(4)-7 at Thu 3-Mar-83
  09:31-EST
*LOGIN (user-ident) CPERLMAN (password) (account)
*<331 User name ok. Password, please.
*<230 User CPERLMAN logged in at Thurs 3-Mar-83 09:31-EST, job 19.
*
```

#### BRIEF COMMAND

The BRIEF command, which is the FTP default, cancels the VERBOSE command. There is no argument for this command. Once you have received the information you require through the VERBOSE command, typing BRIEF cancels the display of the additional information. To use the BRIEF command, type:

```
*BRIEF <RET>
```

where the carriage return is used to confirm the command.

#### QUOTE COMMAND

The QUOTE command is used to send information to the remote host with no translation by the FTP program. The text is sent directly to the Server. To use the QUOTE command, type QUOTE followed by a space, followed by whatever information you wish the remote host to receive. When you press the carriage return, this information is sent exactly and directly to the remote host. The FTP protocol does not interpret this information.

In the following example, a user has requested a new password using the CWD command. (See Section 3.6.1 for a discussion of CWD.) The user wants to send this password directly to the remote host. The command, therefore, appears as:

```
*QUOTE PASS DEC-20 <RET>
```

The QUOTE command causes the string "pass DEC-20" to be sent directly to the remote host.

### 3.6 FTP COMMANDS

The following sections discuss the FTP commands used to: request data transfers or remote file system service, get local information, and specify the data transfer parameters.

#### 3.6.1 Requesting Data Transfers or Remote File System Service

Some FTP commands are used to request a service from the remote host. Included are commands that request files to be moved, deleted, and renamed. The requests that are made must conform to the User-FTP requirements. If defaults are allowed, they are Server defaults. If a name is used for a device or file presently located or to be located at the remote site, the name follows the Server-FTP naming conventions.

You cannot give any commands during a file transfer. You must wait for the \* prompt following the transfer complete message. You can type <CTRL/G>, however, if you want to abort the transfer.

#### NOTE

Some Servers are able to monitor the command connection (commands and responses) and the data connection (file transfers) simultaneously. Consult your System Manager before attempting to use this feature.

The following sections discuss the FTP Service commands.

#### GET COMMAND

The GET Command is used to request that a copy of a file at the remote site be transferred to a file at the local site. The GET command takes the form:

```
GET (foreign-filename) filename 1 <RET>
```

```
to (local-file) filename 2 <RET>
```

where filename 1 is the name of the remote file you are requesting and filename 2 is the name of the local file. Filename 1 uses the naming conventions of the remote host, and filename 2 uses the naming conventions of the local host. It is recommended that you always use a complete file specification for the local file.

If you do not type a directory name preceding either file name, a default is used. For the remote file, the default directory is the FTP-login directory. Or, if a CWD (Change Working Directory) command has been given (see discussion of CWD below), the directory named in the CWD command is the default. For the local file, the default directory is your current connected directory. If you press the <ESC> key for the local file, the exact name string used in the remote file name is used for the local file name.

When a GET command has been executed, you receive a message indicating that the transfer has begun. A second message appears indicating that the transfer has been completed. If the transfer cannot be completed, you receive a message indicating this. You can follow the instructions in this message to correct the problem and then reissue the \*GET command.

The following example shows a completed GET command:

```
*GET (foreign-filename) TELNET.LOG.3 <RET>
  (To local-file)TELNET.LOG.3 <RET>
<PAGED retrieve of <CPERLMAN> TELNET.LOG.3 started
<Transfer completed.
*
```

#### SEND COMMAND

The SEND command is used to transfer a copy of a file from the local host to the remote host. The SEND command takes the form:

```
SEND (local-file) filename1 <RET>
  (to remote-file) filename2 <RET>
```

where filename 1 is the name of the local file, and filename 2 is the name of the remote file.

The same naming conventions and messages that apply to the SEND command apply to the GET command.

An example of the SEND command follows:

```
*SEND (local-file) TELNET.LOG.5 <RET>
  (to remote-file)TELNET.LOG.5 <RET>
<Paged store of <CPERLMAN> TELNET.LOG.5; P777700; A341 started.
<Transfer completed.
*
```

## MULTIPLE GET/SEND COMMANDS

TOPS-20 ARPANET sites allow MULTIPLE GET/SEND commands. These commands allow the use of the TOPS-20 wildcard feature for the transfer of multiple files either to or from a remote host.

The MULTIPLE GET/SEND commands take the form:

```
MULTIPLE GET (remote-file group) group filename
  or extension <RET>
  or
MULTIPLE SEND (local-file group) group filename
  or extension <RET>
```

where in either command a wildcard can be used for the file names or the extensions, as in:

```
*MULTIPLE GET (remote-file group) FTP.* <RET>
  or
*MULTIPLE SEND (local-file group) *.MEM <RET>
```

All other naming conventions and defaults that apply to the GET and SEND commands apply to the MULTIPLE GET/SEND commands.

An example of a MULTIPLE GET command follows:

```
*MULTIPLE GET (remote-file group) *.MEM <RET>
<List started.
<Transfer completed.
ARPA.MEM.4
<Paged retrieve of <CPERLMAN> ARPA.MEM.3 started.
<Transfer completed.
<SAMPLE.MEM.13
.
.
.
*
```

## RENAME COMMAND

The RENAME command is used to rename a file at the remote host. The RENAME command takes the form:

```
RENAME (existing-file) filename 1 <RET>
  (to be) remote-filename <RET>
```

where filename 1 is the current name of the remote file and remote filename is the new file name for the file. File-naming conventions follow the requirements of the remote host.

An example of the RENAME command follows:

```
*RENAME (existing-file) ARPA.MEM.4 <RET>
  (to be) ARPA.TXT.1 <RET>
```

```
<Old file renamed <CPERLMAN> ARPA.TXT.1
*
```

#### DELETE COMMAND

The DELETE command is used to delete files at the remote host. The command takes the form:

```
DELETE (foreign file) filename <RET>
```

where filename is the name of the file you want to have deleted.

When you have issued a DELETE command, you are asked whether this is the file you want to delete. This is done to insure that you do not delete the wrong file. If you make a mistake in typing the file name, you can edit the file name before pressing the <RET> key. If you do not want to delete the file, answer N in response to the message. You then see the FTP prompt.

An example of the DELETE command follows:

```
*DELETE (foreign file) FTP.LOG.2 <RET>
Do you really want to delete FTP.LOG.2 (Y or N)? Y <RET>
*<Lowest generation deleted ok, file <CPERLMAN> FTP.LOG.2
*
```

#### CWD COMMAND

The CWD (Change Working Directory) command is used either to change your default directory at the remote site or to connect to a different directory at the remote site. This command is similar to a TOPS-20 Connect command. Access to a different directory for file storage or retrieval at the remote site is possible with this command. The command takes the form:

```
CWD (to directory) directory name <RET>
```

where directory name is the name to be given to the working directory or the name of the directory you want to access.

This command saves the overhead of repeating the log-in and accounting information. Transfer parameters remain as originally selected. The name given for the directory follows the convention of the remote host. Also, the remote host may require that you type a password or account at this time. The TOPS-20 Server-FTP program requires that you give a password if you are required to give a password when doing a connect to the named directory.

An example of the CWD command follows:

```
*CWD PS:<PAETZOLD> <RET>
*<250 Connected to PS:<PAETZOLD>.
```

## NOTE

TOPS-20 directory format is  
STRUCTURE:<DIRECTORY> where STRUCTURE,  
if not specified, defaults to the  
current connected structure and  
<DIRECTORY>, if not specified, defaults  
to the current connected directory.

TENEX and several other operating  
systems use the format USERNAME or  
username for directory.

## DIRECTORY COMMAND

The DIRECTORY command is used to request that a listing of the names of the files in the named directory at the remote host be sent to the local host. The DIRECTORY command takes the form:

```
DIRECTORY (of remote user) DIRECTORY NAME <RET>  
to local file: filename or TTY: <RET>
```

where DIRECTORY NAME is the name of the remote directory you want listed, filename is the name of the local file you want to have the directory placed in, and TTY: is used to have the directory output to your terminal.

You must type TTY: if you want output to your terminal. File-naming conventions follow the local host requirements if you want to have output placed in a file. If you do not type a directory name, the default argument is your current connected directory.

You receive a message indicating that the transfer has been completed.

An example of the DIRECTORY command follows:

```
*DIRECTORY of <CROSSLAND> <RET>  
to local-file: TTY: <RET>  
<List started.  
PS:<CROSSLAND>  
CHING.MAC.4  
DUMMY.1  
MAIL.TXT.1  
TEMP.FIL.1  
<Transfer completed.  
*
```

## STATUS COMMAND

The STATUS command is used to request information about the parameters of your current connection to the remote host. The STATUS command takes the form:

STATUS (of server) <RET>

where a carriage return <RET> is used to confirm the command. You then receive a message that shows the current parameters for your connection, as follows:

```
*STATUS <RET>
DEC-MARLBORO FTP User Process 6(13)-7
The current data transfer parameters are:
  MODE S
  STRU P
  TYPE L 36
NORETAIN generations
A connection is open to host DEC-MARLBORO
The data connection is closed.
*
```

#### ACCOUNT COMMAND

The ACCOUNT command takes the form of:

ACCOUNT <number> or <string> <RET>

The ACCOUNT command accepts either a number or a string as the argument identifying the user's account at the remote site. This command is not necessarily related to the account parameter in the LOGIN command. Some hosts require that an account be given to access specific files, or to manipulate files in a specific way. The ACCOUNT command is provided to satisfy this requirement.

#### RETAIN COMMAND

The RETAIN command causes the generation number of a file to be transferred either to or from a remote host. The generation number of the file is not changed by the FTP protocol during transmission. The RETAIN command takes the form:

RETAIN <RET>

where a carriage return is used to confirm the command. Once this command is executed, generation numbers are sent/received exactly as they appear in the name string.

#### NORETAIN COMMAND

The NORETAIN command, which is the system default, cancels the RETAIN command. This command takes the form:

NORETAIN <RET>

where a carriage return is used to confirm the command. Once the NORETAIN command is executed, generation numbers of files can change

during transmission. Therefore, if there is a file being requested that has a generation number of 4 and there is no existing file at the receiving site, the file will have a generation number of 1 when it is received, as follows:

```
*GET (foreign-filename) TELNET.TXT.4 <RET>
      (to local-file) <ESC> TELNET.TXT.1 <RET>
```

If the NORETAIN command is executed, you must type the generation number if you want to have the generation number retained.

### 3.6.2 FTP Information Commands

In addition to the VERBOSE and QUOTE commands already discussed in Section 2.5, another command, the STATISTICS command, is passed directly to the program. Neither the command connection nor the data connection are involved. All communication is accomplished at the local site.

#### [NO]STATISTICS COMMAND

The STATISTICS command and its counterpart, the NOSTATISTICS command, are used to control the terminal display of statistical information about a transfer. The information you receive displays statistics on file transfer rate, bits sent, and time consumed during data transfer. This information is displayed after the data transfer has been completed.

The STATISTICS command takes the form:

```
STATISTICS <RET>
```

where a carriage return <RET> is used to confirm the command. There is no argument to the command. The STATISTICS command is typed immediately prior to the command for which you are requesting the statistical information.

An example of the STATISTICS command follows:

```
*STATISTICS <RET>
*GET (foreign-filename) TELNET.LOG.2 <RET>
      (to local-file) TELNET.LOG <RET>
<Paged retrieve of <CPERLMAN> TELNET.LOG.5 started.
<Transfer completed.
```

```
137. bytes transferred, run time = 96. MS,
Elapsed time =3080. MS, Rate =1601. Baud
*
```

The NOSTATISTICS command cancels the STATISTICS command. There are no arguments to the NOSTATISTICS command. The NOSTATISTICS command is



the system default. Therefore, the NOSTATISTICS command is only used if you want to cancel a STATISTICS command.

The NOSTATISTICS command takes the form:

```
NOSTATISTICS <RET>
```

where a carriage return is used to confirm the command.

### 3.6.3 Controlling FTP Transfer Parameters

FTP transfer parameters specify how bits are transmitted and how the data is represented. You seldom need to use these parameter commands since the User-FTP protocol selects those parameters that provide the best fit for a connection before attempting a connection. This is done transparently. All but a few hosts accept the parameters selected by the program.

If you give a data transfer parameter command that is known by the program to be "unsuitable," the command is ignored. If you are trying to receive service from a host and the host is either unwilling or unable to accept the existing data transfer parameters, the remote host generally sends you a message indicating that the parameter is not supported. If this happens, it is recommended that you try some other combination of parameter values. If it is important that you receive some service from that particular host, contact the System Manager or the liaison contact for that site.

#### WARNING

If you need to use the FTP transfer parameter commands, contact a system person at your site. Do not attempt to use these commands if you are not experienced with them.

All hosts in the FTP program must accept a set of "minimum implementation" values for data transfer parameters. All FTP hosts, therefore, accept the following values as standard defaults:

```
TYPE: ASCII NONPRINT
BYTE: 8 bits
MODE: STREAM
STRU: FILE
```

These four default values are implemented in the TOPS-20 FTP programs. The default values are assumed unless the user or, more frequently, the User-FTP program gives a command with different values. The transfer values set are normally reported in a message.

The data transfer commands supported by the User-FTP program are

discussed below.

#### TYPE COMMAND

The TYPE command specifies the data representation type. Possible arguments are A, I, and L. The TYPE command takes the form:

```
TYPE (key letter) letter (decimal) number <RET>
```

where key letter can be A, I, or L and number is 8, 32, or 36.

TYPE A specifies NVT-ASCII, that is a 7-bit ASCII string right-justified in an 8-bit byte with no vertical format information for printing. TYPE I (Image) specifies contiguous bits that, for transfer, are packed into bytes of the size specified in the BYTE command. Type L (Long) specifies a logical byte size. Theoretically, the logical byte size and the data transfer byte size may differ. Because of the restrictions described, and because the byte size is taken from the BYTE command (explicit or implied), the TYPE L command is seldom used.

An example of the TYPE command follows:

```
TYPE (key letter) L (decimal) 36 <RET>
```

#### STRUCTURE COMMAND

The STRUCTURE command specifies the file structure to be used. FTP currently recognizes file structure and paged structure. The TOPS-20 user-FTP program selects a default structure based on the type of operating system at the remote site. If the remote site is listed as TOPS-20, TOPS-10, or TENEX, PAGED structure is used as the default. Other systems use FILE structure as the default.

The STRUCTURE command takes the form:

```
STRUCTURE (key letter) letter <RET>
```

where letter is either P for page structure, F for file structure, or R for record structure.

An example of the STRUCTURE command follows:

```
*STRUCTURE (key letter) F <RET>
```

#### MODE COMMAND

Only the stream transfer mode is supported by TOPS-20. This is the default mode. If the MODE command is given, the argument must be S. The FTP protocol also recognizes block and compressed mode (represented as B and C, respectively). B and C modes are not supported by TOPS-20. Since all FTP implementations must support

stream mode, there should be no problem.

The MODE command takes the form:

```
MODE (key letter) letter <RET>
```

where the letter for TOPS-20 is S. An example of a completed MODE command follows:

```
*MODE (key letter) S <RET>
```

### 3.7 RESPONSE CODES AND MESSAGES

The FTP protocol specifies that each command received by the FTP Server generate at least one reply. Thus, the User process is aware of the current state of the Server whether the command is successful or unsuccessful. There is currently one exception. Before a connection is made, any error in the CONNECT command itself results in the ? response rather than a meaningful message. Once the connection is established, you generally receive sufficient response to commands to permit you to continue.

A 3-digit numbering system is followed for all response codes. These codes are for the User-FTP program and are interpreted by the program. The program may or may not verbalize the implied content of the code and display it for the user. If you use the VERBOSE command, reply codes with explanatory verbal messages are displayed on your terminal. Use of the VERBOSE command is recommended whenever you are engaged in unfamiliar tasks or are connecting to a host whose operating system you are not familiar with.

Command responses that appear on your terminal may originate in either the User- or Server-FTP program. Messages from the server are preceded by a left angle bracket (<). The content of responses is specified, but the specific wording is left to the implementing site. Because you may be connecting to many different Servers, it is not possible to list all possible responses. There are varying texts for the same reply code.

Server-FTP responses may sometimes appear on your terminal while you are typing commands to the User-FTP. The \* prompt may appear before the response to the previous command.

If you have any problems with coded responses from a Server, contact the liaison person at the Server site. See the current ARPANET RESOURCE HANDBOOK for both a list of liaison contact and instructions on getting an up-to-date list by means of the ARPANET. Such problems as replies not clearly worded, replies that do not appear appropriate in the context of the existing connection, or negative completion replies that you cannot understand should be reported to the appropriate Server site.

### 3.7.1 Response-Coding System

The major category of the response code is specified by the first digit. This digit indicates the general acceptability of the command. The second digit distinguishes the first digit by function, and the third digit distinguishes the second.

The first digit indicates acceptability as follows:

#### 1nn Positive Preliminary Reply.

An action has been requested and has been started; there will be another reply before the next command is considered.

Example: 150 file status okay; about to open data connection.

#### 2nn Positive Completion Reply.

An action has been requested and completed successfully. The next command can be given.

Examples: 212 Directory status.  
250 Requested file action okay, completed.

#### 3nn Positive Intermediate Reply.

A command has been given and accepted, but more information is required. This is used specifically in command sequence groups.

Example: 332 Log in first, please.

#### 4nn Transient Negative Completion Reply.

This reply encourages you to try again. Your command was received and no action was taken. The error condition can be corrected without changing the command, the arguments, the file access, or the username, and without requiring additional implementation on the part of the Server. You are most likely to get this reply code when there are problems not associated with errors in the commands.

Examples: 421 Service not available closing connection.  
This is given in reply to any command when the server knows the system is going down.  
452 Requested action not taken: insufficient storage space in system.

#### 5nn Permanent Negative Completion Reply.

A command was given but it was not acceptable. No action was taken. A reply with 5 as the first digit always indicates an error. The user must decide from the verbal message what action should be taken. Read the message and then examine the command that caused the message to be given. Perhaps a change in spelling or in your connected directory is in order. Perhaps the remote system is using a file you have requested. If so, give the BYE and QUIT commands and return later. However, if the verbal message indicates the command is not implemented, no further action is possible.

Examples: 501 Syntax error in parameters or arguments.  
502 Command not implemented.  
550 Requested action not taken: file unavailable.

The second digit is based on a functional classification as follows:

- n0n Syntax, including syntax errors or unnecessary commands
- n1n Information
- n2n Connections
- n3n Authentication (logon) and Accounting
- n4n Reserved
- n5n File system

The third digit further defines the meaning assigned in the second digit. Hosts adding to or implementing an FTP program are requested to adapt the verbal expression for a reply code if an already similar 3-digit code exists. An example of the function of the third digit is found in the 21n series. All four replies below are positive completion; all are information replies. The third digit indicates the type of information.

- 211 System status or system help
- 212 Directory status
- 213 File status
- 214 Help message

## CHAPTER 4

### TSTATS

#### 4.1 INTRODUCTION

TSTATS is an ARPANET utility that provides information about connections between hosts. TSTATS gives you the following information:

- o The existing connections
- o The state of the connections
- o The number of bytes of data you can send to a receiving host

#### 4.2 RUNNING TSTATS

To run TSTATS, type TSTATS in response to the command level prompt, as in the following example:

```
@TSTATS <RET>
```

When you press carriage return, a statistical listing of ARPANET connections is displayed on your terminal.

The following example shows the information displayed in response to the TSTATS command.

```
@TSTATS <RET>
JFN      STATE      LPORT  FPORT  FGN-HOST      R-SEQUENCE  S-SEQUENCE  SENDW
1,2  FIN.FIN.--P----   25      0  0,0,0,0        0            0            0
0,2  -3-.EST.O-PAV--   23  13591  RADC-TAC      1780306709   359662611   256
0,4  -3-.EST.O-PAV--   23   6679  RADC-TAC      1791488463   3027243164   256
0,5  -3-.EST.O-PAV--   23   4582  DEC-MARLBORO3056730154  3066953916   777
0,1  FIN.FIN.--P-V--    23      0  0,0,0,0        0            0            0
1,1  FIN.FIN.--P----   21      0  0,0,0,0        0            0            0
@
```

Once this information is displayed on your terminal, you are returned to command level. The TSTATS utility has no other commands or arguments.

### 4.3 INTERPRETING TSTATS

This section discusses the information displayed by TSTATS.

#### JOB NUMBER, JFN

The first column in TSTATS lists the job number for your job and the job file number (JFN) within the job. In the example presented above, the numbers 1,2 under JFN represent job number 1 and job file number 2.

#### STATE

The second column in TSTATS lists the state of the connection for the local port and the foreign port. The first code word indicates the state of the local port. The second code word indicates the state of the foreign port. The code words that are printed are states caused by the TSTATS protocol. The letters, called flags, represent the states caused by what the user requests. Both words and flags are displayed in the STATE column.

Connections can be in one of the following states:

CODE	MEANING
EST	Established
PND	Pending
SYN	Opened by one host
FIN	Closed by one host
NOT	Closed by both host
O	Connection open
P	Push (to send information)
A	Connection aborted
V	Send sequence number is valid
U	Urgent data to be received

An established connection (EST) is an active connection between two hosts. A pending connection (PND) exists when one host has requested a connection (SYN) and is waiting for the other host to indicate whether or not the host will accept the connection. A closed connection (NOT) exists when both hosts have closed the connection. A NOT connection may also exist when a local host has never established a connection and is open or listening for a connection. A FIN connection exists when one host has closed the connection and is waiting for the other host to close the connection.

An O flag is displayed when an active connection exists. A P (PUSH) flag is displayed when the user has issued a command that requires immediate delivery of data. An A flag is displayed when the connection has been aborted. A V flag is displayed when the send sequence number (see discussion of sequence number below) is valid. A U flag is displayed when there is a high-priority message, such as a TELNET control message, to be sent to the local host. This priority

message takes precedence over all other data.

In the following example of a STATE column, both hosts are waiting for the connection to be closed. The FIN.FIN. indicates this. The P flag indicates that data has been sent and is guaranteed delivery.

```
STATE
FIN.FIN.--P----
```

#### LOCAL PORT, FOREIGN PORT

The third and fourth columns in TSTATS list the ports that the connections exist on. TCP/IP establishes connections between hosts on specific ports. Ports indicate the type of connection. Ports may be TELNET, FTP, or mail. (For a detailed discussion of ports, see Chapter 2, TELNET.) The third column of TSTATS lists the local port (LP) or the connection on the local host side.

In the following example, a TELNET connection exists for the local port. This is indicated by the number 23.

```
LPORT
23
```

The fourth column of TSTATS lists the foreign port (FP) or the connection on the foreign side. The ports for the foreign port are given as decimal numbers.

The most common ports are:

```
Port 21   FTP
Port 23   TELNET
Port 25   Mail
```

A port designated as 0 indicates that a host is waiting for a connection.

#### FOREIGN HOST (FGN-HOST)

The fifth column, FGN-HOST, lists the host name for the foreign host side of the particular connection. Most often, this column lists the host name. If the host name is unknown, the decimal octet number of the host is displayed. If zeros (0) are displayed in the foreign host column, there is no connection to a foreign host.

In the following example, the first foreign host has zeros listed. This indicates that no connection exists. In the second line under the header, the name RADC-TAC is listed. This indicates that the foreign host is RADC-TAC.

```
FGN-HOST
0,0,0,0
RADC-TAC
```



## RECEIVE SEQUENCE (R-SEQUENCE)

The sixth column in TSTATS lists the receive sequence (R-SEQUENCE). The receive sequence is the number of the next byte that the foreign port will receive.

In the following example, the receive sequence is 1780306709.

```
R-SEQUENCE
1780306709
```

## SEND SEQUENCE (S-SEQUENCE)

The seventh column in TSTATS lists the send sequence (S-SEQUENCE). The send sequence is the number of the last byte that the local host sent to the foreign host.

In the following example, the send sequence is 359662611.

```
S-SEQUENCE 359662611
```

## SEND WINDOW (SENDW)

The final column in TSTATS is the send window (SENDW). The send window in TSTATS is the number of bytes of data that a foreign port is either willing or able to receive from the local port. Most often, the send window is determined by the buffer space available at the foreign host. If the SENDW is at 0, the foreign port will not accept data transmission. The SENDW does change as space becomes available or unavailable.

In the following example, there is a send window of 0 and a send window of 256. The send window of 0 indicates that the foreign port is unwilling or unable to accept data. The send window of 256 indicates that the foreign port is able or willing to accept 256 bytes of data.

```
SENDW
  0
256
```

## CHAPTER 5

### IPHOST

#### 5.1 INTRODUCTION

IPHOST is an ARPANET utility that:

- o Gives information about ARPANET hosts.
- o Converts host numbers from one format to other formats.
- o Cycles the ARPANET up and down.

#### 5.2 RUNNING IPHOST

To run IPHOST, type IPHOST in response to the TOPS-20 prompt. You are then prompted with IPHOST followed by a right angle bracket (>), as in:

```
@IPHOST <RET>
IPHOST- TOPS-20 TCP/IP Host Utility Program - Version 6(25)
IPHOST>
```

At this point, the IPHOST program is ready to accept commands.

#### 5.3 EXITING IPHOST

To exit from IPHOST, type EXIT in response to the IPHOST prompt, as in:

```
IPHOST> EXIT <RET>
@
```

You are then returned to the Exec command level.

#### 5.4 GETTING HELP WITH IPHOST

The two commands used to get help with IPHOST are the question mark (?) and the HELP command.

The question mark typed after the IPHOST prompt gives a list of the IPHOST commands. The following example shows the use of the question mark:

```
IPHOST>? one of the following:
  ARPANET   CONVERT   EXIT    HELP    LOCAL   NAME    NUMBER
```

Escape can be used to help complete command strings. In this case, type an <ESC> following the command. The IPHOST program then prompts you for the information needed to complete the command, as in the following example:

```
IPHOST>NAME (OF HOST IS)DEC-MARLBORO ? confirm with carriage return
```

The HELP command prints a brief help text at your terminal. The following example shows the text printed in response to the HELP command.

```
IPHOST> HELP <RET>
Command      Description
-----
ARPANET      Arpanet interface control functions.
CONVERT      Convert host numbers from one format to the
              other formats.
EXIT         Return to EXEC level
HELP         Type this text.
LOCAL        Dump information about the local host.
NAME         Dump information about a specific host given
              the name.
NUMBER       Dump information about a specific host given
              the number. The number is in the
              format aaa.bbb.ccc.ddd.
```

#### 5.5 IPHOST COMMANDS

IPHOST commands fall into three categories:

- o Commands that give information about ARPANET hosts.
- o Commands that convert decimal, octal, and INTERNET numbers to other formats.
- o Commands that cycle the ARPANET up and down.

The following section discusses the IPHOST commands.

#### NAME COMMAND

The NAME command gives information about ARPANET hosts. The information includes decimal octet number, host name, status as user or server, and type of software. The NAME command takes the form:

```
IPHOST>NAME (OF HOST IS) host name <RET>
```

where host name is either the host name or host nickname. If you are unsure of the host name or nickname, typing a question mark after the NAME command displays a list of server host names, server host nicknames, user host names, and user host nicknames. The following example shows the information displayed in response to the question mark after the NAME command:

```
IPHOST>NAME (OF HOST IS) ? <RET>
```

```
Server Host Name
```

```
-----
```

```
One of the following:
```

ACC	ADA-VAX	AEROSPACE	AFGL
AFSC-AD	AFSC-DEV	AFSC-HG	AFSC-SD
AFWL	AMES-11	AMES-67	ANL
ANL-MCS	ARPA-DMS	BBN-CLXX	BBN-INOC

```
.  
.
.
```

The following example shows the information displayed in response to the NAME command.

```
IPHOST> NAME (OF HOST IS) DEC-MARLBORO <RET>  
010.001.000.079 DEC-MARLBORO, SERVER, USER, TOPS-20
```

#### LOCAL COMMAND

The LOCAL command provides the same information for the local host as the NAME command provides for a specified host. The LOCAL command takes the form:

```
IPHOST>LOCAL (HOST INFORMATION) <RET>
```

Following a carriage return, the information shown in the following example is displayed at your terminal:

```
IPHOST>LOCAL (HOST INFORMATION) <RET>  
010.001.000.079 DEC-MARLBORO, SERVER, USER, TOPS-20
```

#### NUMBER COMMAND

The NUMBER command gives the same information as the NAME and LOCAL commands, but you input a host decimal octet number instead of a host name. The NUMBER command takes the form:

```
IPHOST>NUMBER (OF HOST)DECIMAL OCTET <RET>
```

where decimal number is one of the decimal octets for the INTERNET host you want information about. The following example shows the information displayed with the NUMBER command.

```
IPHOST>NUMBER (OF HOST) 10.1.0.79 <RET>
010.001.000.079 DEC-MARLBORO, SERVER, USER, TOPS-20
```

#### CONVERT COMMAND

The CONVERT command is used to change the format of a host number from Internet to decimal or octal, from decimal to Internet or octal, and from octal to decimal or Internet. The CONVERT command takes the form:

```
IPHOST>CONVERT (A HOST NUMBER FROM)decimal,internet, octal <RET>
```

where decimal, internet, and octal followed by a number specify the address of the host you want information about.

The following examples show the three formats of the CONVERT command:

```
IPHOST>CONVERT (A HOST NUMBER FROM)INTERNET 010.001.000.079<RET>
Internet: 010.001.000.079, Decimal: 167837775, Octal: 1200200117
```

```
IPHOST>CONVERT (A HOST NUMBER FROM) DECIMAL 167837775 <RET>
Internet: 010.001.000.079, Decimal: 167837775, Octal: 1200200117
```

```
IPHOST>CONVERT (A HOST NUMBER FROM) OCTAL 1200200117 <RET>
Internet: 010.001.000.079, Decimal: 167837775, Octal: 1200200117
```

#### ARPANET COMMAND

The ARPANET command is a privileged command that is used to cycle the ARPANET up and down. Any one of three words is used to complete this command: initialize, status, and state. The ARPANET command takes the form:

```
IPHOST>ARPANET command word <RET>
```

where command word is either initialize, status, or state.

The ARPANET INITIALIZE command is used to reload host or gateway tables. This command takes the form:

```
IPHOST>ARPANET INITIALIZE HOST (TABLES)<RET>
```

or

```
IPHOST>ARPANET INITIALIZE GATEWAY (TABLES)<RET>
```

where a carriage return confirms the command.

The ARPANET STATUS command tells you whether the ARPANET is up or down. This command takes the form:

```
IPHOST>ARPANET STATUS <RET>
```

where a carriage return confirms the command. Following the execution of this command, an informational message is displayed at your terminal:

```
Network interface is down
Network interface output is disabled
Network service is enabled
Last network interface off tansition: 18-Nov-83 14:16:17
```

The ARPANET STATE command is used to bring up or bring down the ARPANET. This command takes the form:

```
IPHOST>ARPANET STATE (IS) ON OFF<RET>
```

where the arguments ON or OFF are used to complete the command. The ON argument turns the ARPANET on. The OFF argument turns the ARPANET off. An example of the ARPANET STATE command with the argument ON follows:

```
IPHOST>ARPANET STATE (IS) ON <RET>
```

## 5.6 IPHOST MESSAGES

The most common IPHOST messages, their descriptions, and suggested user responses are listed alphabetically below. When IPHOST is unable to continue with a user command, it outputs one of the error messages listed below. The errors are generally errors in the user's command line. Fatal errors are preceded by a question mark (?), and warning messages are preceded by a percent sign (%).

%Does not match switch or keyword

You specified a keyword or switch that is not valid.

SUGGESTED USER RESPONSE: Correct the keyword or switch and try the command again.

%Name for Host not found

You specified an invalid host name.

SUGGESTED USER RESPONSE: Check to see that you have specified a valid host name. Check the spelling of the host name to make sure that you have typed it correctly.

%Special Privileges are required for this command.

You have tried to execute an ARPANET command without WHEEL or OPERATOR privileges enabled.

SUGGESTED USER RESPONSE: Enable WHEEL or OPERATOR privileges before executing this command.