COVER Book Cover

IBM TCP/IP for OS/2:

User's Guide

Document Number SC31-6076-05

This softcopy edition supersedes the hardcopy book, SC31-6075-04. The hardcopy book has not been updated to include the changes herein.

NOTICES Notices

Note

Before using this information and the product it supports, be sure to read the general information under "Notices" in topic FRONT_1.

EDITION Edition Notice
Sixth Edition (August 1994)

This edition applies to the IBM Transmission Control Protocol/Internet Protocol Version 2.0 for OS/2 licensed program.

Publications are not stocked at the address given below. If you want more IBM publications, ask your IBM representative or write to the IBM branch office serving your locality.

A form for your comments is provided at the back of this document. If the form has been removed, you may address comments to:

IBM Corporation
Department E15
P.O. Box 12195
Research Triangle Park, North Carolina 27709
U.S.A.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

CONTENTS Table of Contents

FIGURES Figures

TABLES Tables

FRONT_1 Notices

References in this publication to IBM* products, programs, or services do not imply that IBM intends to make them available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any of the intellectual property rights of IBM may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, New York 10594
U.S.A.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement.

This document is not intended for production use and is furnished as is without any warranty of any kind, and all warranties are hereby disclaimed including the warranties of merchantability and fitness for a particular purpose.

IBM is required to include the following statements in order to distribute portions of this document and the software described herein to which contributions have been made by Sun Microsystems, Massachusetts Institute of Technology, Digital Equipment Corporation, and The University of California.

Portions herein ¦ Copyright 1979, 1980, 1983, 1986, Regents of the University of California. Reproduced by permission. Portions herein were developed at the Electrical Engineering and Computer Sciences Department at the Berkeley campus of the University of California under the auspices of the Regents of the University of California.

Portions of this publication relating to RPC are Copyright ¦ Sun Microsystems, Inc. 1988, 1989.

- Export of this software from the United States of America is assumed to require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting.

WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute the M.I.T. portions of this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. makes no representation about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Some portions of this publication relating to X Window System are Copyright ¦ 1987, 1988 by Digital Equipment Corporation, Maynard, Massachusetts, and the Massachusetts Institute of Technology, Cambridge, Massachusetts. All Rights Reserved.

Some portions of this publication relating to X Window System are Copyright ¦ 1986, 1987, 1988 by Hewlett-Packard Corporation.

Permission to use, copy, modify, and distribute the M.I.T., Digital Equipment Corporation, and Hewlett-Packard Corporation portions of this software and its documentation for any purpose without fee is hereby granted, provided that the above copyright notice appears in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the names of M.I.T., Digital, and Hewlett-Packard not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T., Digital, and Hewlett-Packard make no representation about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Subtopics

FRONT_1.1 Trademarks

FRONT_1.1 Trademarks

The following terms, denoted by an asterisk (*) at their first occurrences in this publication, are trademarks of IBM Corporation in the United States or other countries:

AIX	IBM	Proprinter
AnyNet	Operating System/2	PS/2
AS/400	OS/2	Ultimedia
BookManager	Presentation Manager	Workplace Shell

The following terms, denoted by a double asterisk (**) at their first occurrences in this publication, are trademarks of other companies:

Trademark	**Owned By**
DEC	Digital Equipment Corporation
IEEE	Institute of Electrical and Electronics Engineers
Microsoft	Microsoft Corporation
NDIS	3COM Corporation and Microsoft Corporation
Network Driver Interface Specification	3COM Corporation and Microsoft Corporation
Network File System	Sun Microsystems, Inc.
NFS	Sun Microsystems, Inc.
OSF/Motif	Open Software Foundation, Inc.
Portmapper	Sun Microsystems, Inc.
PostScript	Adobe Systems, Inc.
Sun	Sun Microsystems, Inc.
UNIX	UNIX System Laboratories, Inc.
VT200	Digital Equipment Corporation
VT220	Digital Equipment Corporation
Windows	Microsoft Corporation
X Window System	Massachusetts Institute of Technology

PREFACE About This Book

The *IBM Transmission Control Protocol/Internet Protocol Version 2.0 for OS/2: User's Guide* describes the functions of the IBM(*) Transmission Control Protocol/Internet Protocol Version 2.0 for OS/2 (TCP/IP for OS/2(*)) product.

Note: The abbreviation OS/2 refers to Operating System/2, Version 2.0 or higher.

This book describes how to use the applications available in TCP/IP for OS/2 to perform the following tasks:

- Inquire about network status
- Transfer files
- Print on a remote printer
- Send electronic mail
- Read electronic news
- Send interactive messages
- Log on to remote hosts
- Run programs on remote hosts

Subtopics
PREFACE.1 Who Should Use This Book
PREFACE.2 How to Use This Book
PREFACE.3 Compatibility with IBM AnyNet Product Family
PREFACE.4 Conventions Used in This Book
PREFACE.5 Where to Find More Information
PREFACE.6 Calling IBM Service
PREFACE.7 Obtaining Corrective Service Electronically

PREFACE.1 Who Should Use This Book

This book is intended for people who use a programmable workstation (hereafter referred to as a workstation) with TCP/IP for OS/2, such as end users and system administrators. You should be familiar with OS/2 and the workstation and also understand the concepts associated with multitasking operating systems.

If you are not familiar with TCP/IP concepts, see *Internetworking With TCP/IP Volume I: Principles, Protocols, and Architecture*; *Internetworking With TCP/IP Volume II: Implementation and Internals*; and the *IBM International Technical Support Centers: TCP/IP Tutorial and Technical Overview*.

PREFACE.2 How to Use This Book

Before you start using TCP/IP for OS/2, verify that the TCP/IP for OS/2 software is installed on your workstation. For information about installing TCP/IP for OS/2, see the *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration* manual.

¦You can select (double-click on) the TCP/IP Glossary icon in the TCP/IP ¦folder to obtain a list of commonly-used terms.

For comments and suggestions about the *IBM TCP/IP Version 2.0 for OS/2: User's Guide*, use the Reader's Comment Form located at the back of this book. IBM appreciates any information that might improve the book.

Subtopics
PREFACE.2.1 What Is New in This Book
PREFACE.2.2 How the Term "internet" Is Used

PREFACE.2.1 What Is New in This Book

¦Updates to the previous softcopy and hardcopy book (SC31-6076-04) are ¦indicated by revision bars (¦) in the left margin.

¦**Note:** The hardcopy book has not been updated to include these revisions.

¦Some of the major changes are shown in Table 1.

Table 1. What is New in This Softcopy Edition

Function/Topic	Description	See
CSDs	The method for obtaining Corrective Service Diskettes (CSDs) electronically is described.	PREFACE.7
FTP	An image transfer type is supported.	2.11.4, 2.11.4.4
FTPPM	The capability to trace your session activity is provided.	3.7.6
Icons	Several icons have been added to the TCP/IP folder.	1.2.1
Packet Tracing	A set of utilities used for tracing packets is provided.	13.5
REXEC	Several new parameters are supported.	11.4
RFCs	The method for obtaining Request for Comments (RFCs) is described.	PREFACE.7.1
RSH	Several new parameters are supported.	12.4
SENDMAIL	Configuring a SENDMAIL.CF file for LaMail is described. Retrieval of addressing and subject information from the header area (top) of the message rather than from the command line is supported.	5.2.2 5.5
SNMP	The following enhancements were made: □ MIB2.TBL modifications □ The search order for the **snmp** and the **snmpgrp** commands has been changed □ The **snmp** command supports new parameters	13.7.3 13.7.3.4 13.7.7.1
Telnetd	ASCII codepage support has been added.	7.2
TelnetPM	Cursor display options are available.	7.5.5.3
3270 Emulation	Default 3270 emulator keyboard mappings have been updated.	8.1

PREFACE.2.2 How the Term "internet" Is Used

In this book, an internet is a logical collection of networks supported by gateways, routers, bridges, hosts, and various layers of protocols, which permit the network to function as a large, virtual network.

Note: The term internet is used as a generic term for a TCP/IP network and should not be confused with the Internet (note capital I), which consists of large national backbone networks (such as MILNET, NSFNet, and CREN) and a myriad of regional and local campus networks all over the world.

PREFACE.3 Compatibility with IBM AnyNet Product Family

IBM TCP/IP Version 2.0 for OS/2 and its kits can coexist with, and be used in conjunction with, the IBM AnyNet(*) family of programs. For example, you can use the TCP/IP applications with the OS/2 protocol stack provided by AnyNet/2. For information about AnyNet/2, see your software distributor or call your IBM representative.

PREFACE.4 Conventions Used in This Book

The following conventions appear throughout this book:

- Commands are presented in lowercase bold, but you can enter them in either uppercase or lowercase.

- Subcommands are presented in lowercase bold, and you must enter them in lowercase.

- File names are presented in uppercase, but you can enter them in either uppercase or lowercase.

- Periods in numbers separate the whole and the decimal portions of the numerals.

- Numbers over four digits appear in metric style. A space is used, rather than a comma, to separate groups of three digits. For example, the number sixteen thousand, one hundred forty-seven is written 16 147.

Subtopics
PREFACE.4.1 How to Read a Syntax Diagram

PREFACE.4.1 How to Read a Syntax Diagram

The syntax diagram shows you how to specify a command or subcommand so that the operating system can correctly interpret what is being typed. Read the syntax diagram from left to right and from top to bottom, following the horizontal line (the main path).

Syntax diagrams use the following symbols:

Symbol	Description
	Marks the beginning of the command or subcommand syntax
	Marks the continuation of the command or subcommand
|	Marks the beginning and end of a fragment or part of the command or subcommand syntax
□	Marks the end of the command or subcommand syntax

Required parameters are displayed on the main path. Optional parameters are displayed below the main path. Default parameters are displayed above the main path.

Parameters are classified as keywords or variables. Keywords appear in uppercase or lowercase, but you must type them as they are shown in the syntax diagram. A command or a subcommand, for example, is a keyword. See "Conventions Used in This Book" in topic PREFACE.4 for the guidelines about entering commands and subcommands.

Variables are italicized, appear in lowercase letters, and represent names or values you supply. For example, a file name is a variable.

In the following example, **infile** is a variable. Replace it with the value that you want.

```
>>--rpcgen--infile--><
```

Include all punctuation such as colons, semicolons, commas, quotation marks, and minus signs shown in the diagram.

Choose One Required Item from a Stack: A stack of parameters with a parameter on the main path means that you must choose one from the stack.

```
                                        <--------------+
>>--snmp----get-----host--community_name----mib_variable----><
          +-next-+
```

Choose One Optional Item from a Stack: A stack of parameters without a parameter on the main path means that you do not have to choose any from the stack; but if you do, you cannot choose more than one.

```
>>--mode--------------><
          +-ascii--¦
          +-binary-+
```

Specify a Sequence More Than Once: An arrow above the main path that returns to a previous point means the sequence of items included by the arrow can be specified more than once.

```
              <-----------+
>>--mkfontdr----directory----><
```

PREFACE.5 Where to Find More Information

The following document is shipped with the IBM TCP/IP Version 2.0 for OS/2 Base Kit:

- *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration*, SC31-6075

The following document is shipped with the TCP/IP for OS/2 Programmer's Toolkit:

- *IBM TCP/IP Version 2.0 for OS/2: Programmer's Reference*, SC31-6077

The following TCP/IP for OS/2 documents can be ordered separately:

- *IBM TCP/IP Version 2.0 for OS/2: User's Guide*, SC31-6076
- *IBM TCP/IP Version 2.0 for OS/2: Command Reference*, SX75-0070
- *IBM TCP/IP Version 2.0 for OS/2: Network File System Guide*, SC31-7069
- *IBM TCP/IP Version 2.0 for OS/2: Extended Networking Guide*, SC31-7071
- *IBM TCP/IP Version 2.0 for OS/2: NetBIOS Guide*, SC31-6122
- *IBM TCP/IP Version 2.0 for OS/2: X Window System Client Guide*, SC31-7087
- *IBM TCP/IP Version 2.0 for OS/2: X Window System Server Guide*, SC31-7070
- *IBM TCP/IP Version 2.0 for OS/2: Domain Name Server Guide*, SC31-7174
- *IBM TCP/IP Version 2.0 for OS/2: UltiMail Installation and Configuration Guide*, SC31-7120

The TCP/IP for OS/2 documents are shipped in softcopy (BookManager*) format and can be installed during your initial product installation. You can also install them later by rerunning the installation program. Once installed, you can access the documents by clicking on the icons (one icon for each document). Features available with the softcopy documents include search facilities and hypertext capabilities.

For a listing of additional IBM and non-IBM publications that may be helpful, see the Bibliography in the *IBM TCP/IP Version 2.0 for OS/2: User's Guide*.

PREFACE.6 Calling IBM Service

The IBM Support Center provides you with telephone assistance in problem diagnosis and resolution. You can call the IBM Support Center at any time; you will receive a return call within eight business hours (Monday-Friday, 8:00 a.m.-5:00 p.m., local customer time). The number to call is 1-800-237-5511.

Outside the United States or Puerto Rico, contact your local IBM representative or your authorized IBM supplier.

PREFACE.7 Obtaining Corrective Service Electronically

You can obtain Corrective Service Diskettes (CSDs) electronically by:

- Using FTP to access SOFTWARE.WATSON.IBM.COM as Anonymous.

  Note: Fixes for TCP/IP for OS/2 are located in the pub\tcpip\os2 directory.

- Establishing a modem connection to the NCSBBS bulletin board at 1-919-517-0001.

  Note: Set your modem settings to: none,8,1.

The latest CSDs for TCP/IP Version 2.0 for OS/2 are shown in Table 2.

Table 2. Corrective Service Diskettes Available Electronically

Kit Name	**Package Name**
Applications Kit	APP20CSD.PKG
Base Kit	TCP20CSD.PKG
Domain Name Server Kit	DNS20CSD.PKG
DOS/Windows Access Kit	DOS20CSD.PKG
Extended Networking Kit	XNT20CSD.PKG
NetBIOS Kit	NET20CSD.PKG
Network File System Kit	NFS20CSD.PKG
OSF/Motif Kit	MOT20CSD.PKG
Programmer's Tool Kit	PGM20CSD.PKG
X Window System Client Kit	XCL20CSD.PKG
X Window System Server Kit	PMX20CSD.PKG

Subtopics
PREFACE.7.1 Obtaining Requests for Comments (RFCs)

PREFACE.7.1 Obtaining Requests for Comments (RFCs)

Requests For Comments (RFCs) is an ongoing set of documents that presents new protocols and establishes standards for the Internet protocol suite. For a list of RFCs that pertain to TCP/IP for OS/2, see Appendix F, "Requests for Comments (RFCs)" in topic F.0.

Hard copies of all RFCs are available from the Network Information Center (NIC), either individually or on a subscription basis. You can obtain these documents from:

    Government Systems, Inc.
    Attn: Network Information Center
    14200 Park Meadow Drive
    Suite 200
    Chantilly, VA 22021

Online copies are available from the NIC using FTP to connect to **ds.internic.net**. You can transfer the files, using the following format:

      RFC:RFC**nnnn**.TXT
      RFC:RFC**nnnn**.PS

Where:

    nnnn   Is the RFC number.
    TXT   Is the text format.
    PS   Is the PostScript** format.

The format for the RFC index is:

      RFC:RFC-INDEX.TXT

Note: Many RFCs are only available in text format. Before requesting a PostScript file, first check the RFC Index to make sure the RFC is available in that format.

You can also request online copies of the RFCs through the electronic mail, from the automated NIC mail server, by sending a message to **mailserv@ds.internic.net**. You must include the following command(s) in body of your note:

      SEND RFC**nnnn.**TXT

or

      SEND RFC**nnnn**PS

Where:

    nnnn   Is the RCF number.
    TXT   Is the text format.
    PS   Is the PostScript format.

For example, to request the text format of RFC 812, you would specify:

      SEND RFC812.TXT

in the body of your note.

To request an online copy of the RFC index, you must include the following command in the body of your note:

      SEND RFC-INDEX.TXT

1.0 Chapter 1. Getting Started

TCP/IP for OS/2 provides extensive facilities for communicating over an internet. With TCP/IP for OS/2 installed on your workstation, you can perform the following communication tasks:

- Log on to a remote host
- Transfer files between hosts
- Print files using a central print server
- Send and receive electronic mail
- Run commands on a remote host

Subtopics

1.1 Overview of TCP/IP for OS/2

You can start these tasks from a command line interface (the OS/2 prompt), or in some cases, from your OS/2 desktop.

Subtopics

1.1.1 Logging on to a Remote Host

TCP/IP for OS/2 enables you to log on to remote hosts using ASCII- based Telnet, 3270-based, or 5250-based Telnet clients. See Chapter 7, "Using ASCII-Based Telnet Clients" in topic 7.0 and Chapter 8, "Using 3270-Based and 5250-Based Telnet Clients" in topic 8.0 for more information.

You have three ASCII-based Telnet clients from which to choose:

- TelnetPM
- Telnet
- Telneto

The TelnetPM client supports ANSI, VT220**, VT100**, and NVT emulation presented in an easy-to-use Presentation Manager window. TelnetPM also provides linemode simulation to enable you to edit a line of text before sending it to the remote host.

The Telnet client supports ANSI, VT220, VT100, and NVT emulation. It runs in an OS/2 window or a full-screen OS/2 session. The Telnet client is somewhat faster than the TelnetPM client, but the TelnetPM client is easier to use. The Telnet client does not support linemode emulation.

The Telneto client supports VT220, VT100, and NVT emulation. It runs in an OS/2 window or a full-screen OS/2 session. Telneto provides true linemode support.

The TCP/IP for OS/2 Telnet clients are integrated with the OS/2 desktop. You can use the Telnet template located on your desktop to create icons that represent instances of Telnet sessions without having to learn or use the Telnet commands and parameters.

From the Telnet Settings notebook, you specify a title, an icon, and other session definitions for that instance. Using the Settings notebook, you specify the name of the remote host, the port to use, the screen dimensions of your emulator window, and other options that govern the session.

You can also log on to a remote host using a 3270-based Telnet client. You have the following 3270-based Telnet clients from which to choose:

- PMANT
- TN3270

Both PMANT and TN3270 are similar; however, PMANT is an OS/2 Presentation Manager application, whereas TN3270 is an OS/2 character-based application. TN3270 is recommended for use with a serial line internet protocol (SLIP) because it is somewhat faster than PMANT, though PMANT offers more function. Both applications support the use of translation tables.

You can also log on to an AS/400 using the 5250-based Telnet client, TN5250. TN5250 is an OS/2 Presentation Manager application that supports the use of translation tables.

1.1.2 Transferring Files between Hosts

TCP/IP for OS/2 enables you to transfer files between a local workstation and a remote host that is running an FTP or TFTP server.

You have three file transfer methods from which to choose:

- File Transfer Protocol (FTP)
- File Transfer Protocol Presentation Manager (FTPPM)
- Trivial File Transfer Protocol (TFTP)

You can use FTP to perform these functions:

- Request help
- Enter OS/2 commands from the FTP command shell
- Connect to a foreign host
- Identify yourself to a foreign host
- Convert file format during file transfer
- Delete or rename files on a foreign host
- Display a list of foreign host directories at your workstation
- Create and display directories on a foreign host

FTP supports High Performance File System's (HPFS) long file-name support, recognizing up to 256 characters. This capability enhances the interoperability with other operating systems that also support long file names. If you do not have HPFS installed on your system, FTP recognizes only an eight-character file name with a three-character extension. FTP runs in either a full-screen OS/2 session or an OS/2 window. For more information on using FTP, see Chapter 2, "Transferring Files with the File Transfer Protocol (FTP)" in topic 2.0.

You can also transfer files between your workstation and a remote host using FTPPM. FTPPM is a Presentation Manager version of FTP. FTPPM supports a maximum of 256 simultaneous connections with remote hosts. FTPPM uses standard Presentation Manager input and output conventions. For more information on using FTPPM, see Chapter 3, "Transferring Files with FTPPM" in topic 3.0.

TFTP is an alternative to FTP. TFTP is a file transfer application that provides less functionality than FTP but is required for use with hosts that have a TFTP server, but not an FTP server. TFTP uses the User Datagram Protocol (UDP) as the underlying protocol; therefore, it is an unreliable means of file transfer. For more information on using TFTP, see Chapter 4, "Transferring Files with the Trivial File Transfer Protocol" in topic 4.0.

You can use the FTPPM template, located in the Templates folder on your OS/2 desktop, to create an icon that you can use to transfer files between your workstation and a remote host. See "FTPPM Clients and the OS/2 Workplace Shell" in topic 3.2 for information about creating and configuring an FTPPM icon for a remote host.

1.1.3 Printing Files Using a Central Print Server

TCP/IP for OS/2 provides both client and server support for remote printing through the use of the Line Printer Protocol with the following commands:

- LPD
- LPQ
- LPRM
- LPR
- LPRMON
- LPRPORTD

The client commands allow you to send files to remote printer queues, query the status of those jobs and the queues to which the jobs were sent, remove jobs from remote queues, and redirect print objects and parallel devices to remote queues.

With the ability to redirect print objects and parallel devices to remote printers, you can print from any OS/2 application to a remote printer. This includes the ability to drag-and-drop files onto OS/2 printer objects, as well as the ability to print from any OS/2 or Windows application running under OS/2 and have the output sent to a remote host for printing.

The server command, LPD, enables an OS/2 workstation to act as a remote print server. Clients can then submit, query, and remove print jobs from any OS/2 print queue (printer object) defined at the server workstation.

1.1.4 Sending and Receiving Electronic Mail

TCP/IP for OS/2 provides support for sending and receiving electronic mail through the use of the Simple Mail Transfer Protocol (SMTP) using the following:

- SendMail
- LaMail
- Ultimedia* Mail/2 (UltiMail)

¦ UltiMail is not part of the base kit and not discussed in detail here.
¦ Please contact your marketing representative for more information
¦ about this product.

LaMail is an electronic mail handling system that is used with the SendMail mail delivery system. LaMail is a Presentation Manager application that enables you to view mail, write notes, and save mail in mail folders. SendMail functions as a mail router, listening for and receiving mail from the LAN and sending mail on the LAN. When SendMail receives mail from the LAN, it stores the mail in the MAIL subdirectory. You can then use LaMail to read the mail. LaMail does not have to be running to receive mail. When you use LaMail to create and send mail, LaMail automatically passes the mail to SendMail for delivery over the LAN.

You can use LaMail to work on your mail from a remote location, even when you are away from your office and do not have a TCP/IP connection. You can copy your notes to a diskette and work with the files at a remote workstation.

When working remotely, you have full access to all of the LaMail functions. You can temporarily store your notes on a diskette and send them when you return to your office workstation. For more information about LaMail remote services, see "Remote LaMail Services" in topic 5.12.11.

1.1.5 Running Commands on Remote Hosts

TCP/IP for OS/2 allows you to execute commands on remote hosts with the following:

- REXEC
- RSH

You can use REXEC to issue a single command on a remote host that is running an REXEC server (REXECD). REXEC requires that you provide an account name and a password for that account. The command that is run by the REXEC server is run under the specified account. For more information about REXEC, see Chapter 11, "Issuing Commands on a Foreign Host with REXEC" in topic 11.0.

You can also use RSH to issue a single command on a remote host. Unlike the **rexec** command, RSH neither requires nor allows you to provide a password.

RSH is similar to REXEC in that both require you to have an account on a remote machine. However, they differ in the authorization scheme. The REXEC server serves any client who correctly specifies an existing account and the account password on the server machine. RSH servers read a list that identifies the user on a specific remote client that has the same privileges as a particular server user. A password is not required, but a client user must be registered before service is provided.

For more information on using the **rsh** command to set up the client, server, and issue commands on a foreign, see Chapter 12, "Issuing Commands on a Foreign Host with the RSH Command" in topic 12.0.

1.2 TCP/IP on Your Desktop

When TCP/IP for OS/2 is started, a number of icons are displayed on your desktop. You can access many of the TCP/IP functions by selecting these icons. For example, you can:

- Configure your workstation to attach it to the network.
- Log on to a remote host. Your workstation can use the Telnet programs to emulate 3270, 5250, VT220, and several other terminals.
- Transfer files between your workstation and a remote host.
- Exchange electronic mail with other users on your network.

If you install any of the optional kits, additional icons may be added, which allow you to access the online publications for the kit and to perform other functions from the workstation. For example, if you installed the:

- NFS** Kit, you can access disk drives on remote host systems
- X System Server Kit, remote hosts can run text and graphics X applications, displaying the output on your workstation screen

During installation, the TCP/IP for OS/2 Base Kit adds the following icons:

- On the desktop:
  - TCP/IP folder
- Inside the TCP/IP folder:
  - FTP
  - FTPPM
  - LaMail
  - NewsReader/2
  - PM Ping
  - Read Me
  - REXX FTP API
  - REXX Sockets API
  - SLIP: Serial Line IP
  - SLIPTERM: Modem Setup
  - SNMPTRAP
  - TCP/IP Command Reference
  - TCP/IP Configuration
  - TCP/IP Glossary
  - TCP/IP Installation and Administration
  - TCP/IP User's Guide
  - TFTP
  - Telnet
  - Telnet Customization
  - TN5250
  - 3270 Telnet
- Inside the Startup folder:
  - TCP/IP Startup
- Inside the Templates folder:
  - Telnet
  - 3270 Telnet
  - FTPPM

You can use the Settings notebook associated with each of these icons to specify information such as parameters of the command called by the icon.

Should you inadvertently erase any of these icons, you can restore them. To restore any or all of the icons, type **tcpdsktp** and press Enter.

Subtopics

1.2.1 TCP/IP Folder

The following list describes each of the Base Kit icons contained in the TCP/IP folder.

FTP
Starts FTP using any parameters specified in the Settings notebook. For more information about FTP, read Chapter 2, "Transferring Files with the File Transfer Protocol (FTP)" in topic 2.0.

FTPPM
Starts FTPPM using any parameters specified in the Settings notebook. For more information about this PM application, see Chapter 3, "Transferring Files with FTPPM" in topic 3.0.

LaMail
Starts LaMail using any parameters specified in the Settings notebook. For more information about LaMail, read "Overview of LaMail" in topic 5.7.

NewsReader/2
Starts NewsReader/2 using any parameters specified in the Settings notebook. For more information about NewsReader/2, read Chapter 6, "Using NewsReader/2" in topic 6.0.

PM Ping
Starts PMPING using any parameters specified in the Settings notebook. For more information about PMPING, read "Monitoring the Availability of a Group of Hosts" in topic 13.2.

Read Me
Opens the Read Me file contained in the \BIN subdirectory. The Read Me file contains the latest information and news about TCP/IP for OS/2. Read this information before you begin using TCP/IP for OS/2.

REXX FTP API
Provides access to the OS/2 TCP/IP FTP APIs. You should be familiar with the basic FTP APIs. For more information, see the *Programmer's Reference*.

REXX Sockets API
Provides access to the OS/2 Socket APIs. You should be familiar with the basic socket APIs and be able to reference the OS/2-specific APIs. For more information, see the *Programmer's Reference*.

SLIP: Serial Line IP
Starts SLIP using any parameters specified in the Settings notebook. For more information about SLIP, read the *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration* manual.

SLIPTERM: Modem Setup
Starts SLIPTerm using any parameters specified in the Settings notebook. For more information about SLIPTerm, read the *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration* manual.

SNMPTRAP
Starts SNMPTRAP using any parameters specified in the Settings notebook. For more information about SNMPTRAP, read "Displaying TRAPs Received from SNMP Agents" in topic 13.7.8.

TCP/IP Command Reference
Opens the online version of the TCP/IP for OS/2 Command Reference.

TCP/IP Configuration
Starts TCPIPCFG, which opens the Configuration notebook. For more information about the Configuration notebook, read the *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration* manual.

TCP/IP Glossary
Opens the online version of the TCP/IP for OS/2 Glossary which contains a list of commonly-used terms.

TCP/IP Installation and Administration
Opens the online version of the TCP/IP for OS/2 Installation and Administration manual.

TCP/IP User's Guide
Opens the online version of the TCP/IP for OS/2 User's Guide.

Telnet

Starts Telnet using any parameters specified in the Settings notebook. For more information, see Chapter 7, "Using ASCII-Based Telnet Clients" in topic 7.0.

TFTP

Starts TFTP using any parameters specified in the Settings notebook. For more information about TFTP, read Chapter 4, "Transferring Files with the Trivial File Transfer Protocol" in topic 4.0.

TN5250

Starts TN5250 using any parameters specified in the Settings notebook. For more information about TN5250, read Chapter 8, "Using 3270-Based and 5250-Based Telnet Clients" in topic 8.0.

1.2.2 Startup Folder

The TCP/IP for OS/2 Base Kit adds the TCP/IP Startup icon to the OS/2 Startup folder. This causes TCPSTART.CMD to be initiated when your workstation is started or restarted.

1.2.3 Templates Folder

The following Base Kit icons (also in the TCP/IP Folder) are added to the Templates folder:

- FTPPM
- Telnet
- 3270 Telnet

See "TCP/IP Folder" in topic 1.2.1 was a description of these icons.

1.2.4 Requesting Help from the OS/2 Command Shell

To request online help from any OS/2 command prompt, issue the **tcphelp** command. The syntax of this command is:

```
>>--tcphelp---------------><
             +-command-+
```

command
: Specifies the name of the TCP/IP command for which you are requesting help. The online version of the TCP/IP for OS/2 Command Reference is opened, displaying information for that command.

2.0 Chapter 2. Transferring Files with the File Transfer Protocol (FTP)

This chapter describes how to transfer files between a local workstation and a remote host that is running an FTP server. FTP runs in either a full-screen OS/2 session or an OS/2 window.

You can also transfer files between your workstation and a remote host using FTPPM and TFTP. FTPPM is a Presentation Manager version of FTP. TFTP is a file transfer application that provides less functionality than FTP but is required for use with certain hosts. For more information on using FTPPM, see Chapter 3, "Transferring Files with FTPPM" in topic 3.0. For more information on using TFTP, see Chapter 4, "Transferring Files with the Trivial File Transfer Protocol" in topic 4.0.

From the FTP command shell, you can:

- Request help
- Enter OS/2 commands from the FTP command shell
- Connect to a foreign host
- Identify yourself to a foreign host
- Convert file format during file transfer
- Delete or rename files on a foreign host
- Display a list of foreign host directories at your workstation
- Create and display directories on a foreign host

FTP supports High Performance File System's (HPFS) long file-name support, recognizing up to 256 characters. This capability enhances the interoperability with other operating systems that also support long file names. If you do not have HPFS installed on your system, FTP recognizes only an eight-character file name with a three-character extension.

Note: For a concise description of commands, subcommands, and parameters, see the *IBM TCP/IP Version 2.0 for OS/2: Command Reference*.

Subtopics

2.1 Deciding When to Use FTP or TFTP

FTP and TFTP both transfer files between your workstation and a foreign host. However, FTP provides the following features that are not available with TFTP:

- Subcommands that list files and enable you to work with directories on the foreign host
- User security
- Support for many clients at a time

Use FTP to:

- Restrict directory access on a per-user basis
- Use HPFS file naming conventions
- Ensure reliable data transmission
- Achieve better performance
- Provide password-protected access
- Work with single or multiple files
- Work with directories
- Use FTP macros

Use TFTP to:

- Connect to a remote host supporting only TFTP

For more information about TFTP, see Chapter 4, "Transferring Files with the Trivial File Transfer Protocol" in topic 4.0.

2.2 Setting Up the Environment

The FTP client and server use the following files to enable or automate various functions:

TRUSERS
Defines access authorization to users on a foreign host

NETRC
Stores user ID and password values for FTP and REXEC clients

You create these files, which must reside in the ETC subdirectory or the directory specified by the ETC environment variable.

Subtopics
2.2.1 TRUSERS File
2.2.2 NETRC File

2.2.1 TRUSERS File

The TRUSERS file is used by the FTP server to determine access authorization for users on the foreign host. You can provide users with read and write access to particular directories.

The following is an example of a TRUSERS file containing multiple entries.

```
user:  chris boz
rd:    d:\ c:\
wr:    d:\tmp c:\tmp
user:  anonymous
rd:    c:\anonymous
wr:
user:  diane green
wr^:    c:\etc.
```

The TRUSERS file statements are described in the following table.

Statement	**Description**
user: chris boz	**chris** is defined as an FTP user. **boz** is the password for chris.
rd: d:\ c:\	**rd: d:\ c:** gives chris access to read files and subdirectories in the c:\ and d:\ directories.
wr: d:\tmp c:\tmp	**wr: d:\tmp c:\tmp** gives chris access to write to files and subdirectories only in the c:\tmp or d:\tmp directories.
user: anonymous rd: c:\anonymous wr:	**anonymous** is defined as the user with no password. This user name has special meaning because you are not required to define a password. This is the only user name you can define without a password. This user can read files and directories in c:\anonymous but cannot write to any files or directories.
user: diane green	**diane** is defined as an FTP user. **green** is the password for diane.
wr^: c:\etc	**wr^: c:\etc** gives diane access to read or write to any file or directory except c:\etc.

Warning: Use discretion in giving write access to other users. A remote user with write access can destroy files and directories on your workstation.

2.2.2 NETRC File

The NETRC file is used by the FTP and REXEC clients as a source for **userid** and **password** values.

The following is an example of the content of an NETRC file containing multiple entries.

```
machine raleigh login kent password baseball
machine boston login bruce password september macdef mymacro
bell
hash
prompt
binary
cd c:\mydir
get myfile.bin

machine york login jane password workday account payday
```

In this example, the password for user **kent** is defined; therefore, when user **kent** logs on to the **raleigh** host, user **kent** and password **baseball** are automatically sent to the FTP server. To allow user **kent** access to the FTP server, the password **baseball** must also be specified for user **kent** in the TRUSERS file on the host running the FTP server.

If you use FTP to open a connection to foreign host **boston**, the user name **bruce** and the password **september** are automatically passed to the FTP server on the other end of the connection. Also, the macro called **mymacro** is defined by the lines following **macdef mymacro**, until a null line is encountered.

To issue the macro **mymacro**, type **$mymacro** at an FTP prompt.

Warning: If you have a Telnet, REXEC, TFTP, RSH, or FTP server running on your workstation, be aware that a NETRC file provides users with **user** and **password** information that can allow them access to other users' files.

2.3 Setting Up the Server

You can start FTPD using INETD. INETD allows you to start multiple servers from a single OS/2 session. For more information about INETD, see the *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration* manual.

A server is required on one of the hosts involved in the transfer of files. You can type **ftpd** at an OS/2 command prompt and press the Enter key to start the server on your local host. The **ftpd** command starts the FTPD.EXE program, which runs as a task until you shut down the server.

For more information about the **ftpd** command, see the *IBM TCP/IP Version 2.0 for OS/2: Command Reference*.

2.4 Entering the FTP Command Shell

To enter the FTP command shell from an OS/2 command prompt, issue the **ftp** command. The **ftp** subcommands are issued from the FTP command shell.

The syntax of the **ftp** command is:

```
>>--ftp-------------------------------------------------------------------->
        +- -d-+  +- -g-+  +- -n-+  +- -i-+  +- -v-+  +- -c codepage-+

>------------------------><
   +- host------------+
            +- port-+
```

ftp Help

```
>>--ftp-- -?--><
```

-d Enables debugging.

-g Disables file name globbing (extension).

-n Specifies that FTP does not look in the NETRC file. You must enter the user ID and password for the remote host.

-i Disables interactive prompting.

-v Toggles verbose mode on. When verbose mode is on, FTP displays all responses from the remote server.

-c **codepage**
Specifies the ASCII codepage used by the remote host. If you do not specify a codepage, the data is sent untranslated.

Some possible selections are TCPDECMU for the DEC** Multinational codepage, TCP8859 for ISO 8859 codepage, or NONE for PC codepage 850.

host Specifies the remote host to which you are connecting.

port Specifies the destination port to which you are connecting. If **port** is not specified, you are connected by default to the well-known FTP port as specified in the services file.

-? Displays help information.

Note: When you use the -d parameter to establish an FTP connection, your password is echoed back to the screen.

You can enter multiple parameters with the **ftp** command. The following example shows the format for using multiple parameters.

```
ftp -gi
```

The **ftp -gi** command disables file-name expansion and interactive prompting.

If you specify a **host**, FTP tries to establish a connection to an FTP server on that host system. The foreign host prompts you for your name, which is your **userid**. After successfully identifying yourself, the foreign host prompts you for a password, if it requires one. After you enter the password correctly, you are connected to the foreign host. As an alternative, the **userid** and **password** values can be automatically sent from the NETRC file, if you are using a NETRC file.

If you cannot open a connection to a foreign host by specifying a **host** with the **ftp** command, or if you enter an incorrect **host** name, you can use the **open** subcommand to open this connection after you enter the FTP command shell.

When FTP transfers files in ASCII format, there must sometimes be a translation between the ASCII representation used on the workstation and the version recognized by other hosts on the network. This translation is most important for special characters used in languages other than English. Two codepages are provided for use with TCP/IP for OS/2:

- TCPDECMU for the DEC** multinational codepage
- TCP8859 for the ISO codepage 8859-1
- NONE to indicate that translation is not required. (The text is passed on to the network in the workstation representation.)

Notes:

1. When FTP is started, the -c option permits you to choose which codepage should be used.

2. If you do not use the -c option, no translation is done unless a default codepage has been set using the TCP/IP configuration notebook, TCPIPCFG.

3. If you always want the TCPDECMU or TCP8859 codepage to be used, without having to specify the -c option, you can set the default codepage with the configuration notebook. The default codepage is set in the Services section of the TCP/IP configuration notebook. If you do set a default codepage, it is used not only by FTP, but by FTPPM, Telnet, and TelnetPM.

The following sections contain examples of how to enter the FTP command shell.

Subtopics

2.4.1 Entering the Command Shell without Specifying a Host

To enter the FTP command shell without specifying a host, you can issue the **ftp** command without parameters:

```
ftp
```

After the FTP command shell is active, the FTP prompt is displayed:

```
ftp>
```

2.4.2 Entering the Command Shell with a Host Specified

The following examples show entry into the FTP command shell with a host specified.

In the first example, an FTP connection is opened between your workstation and the host 129.33.254.5.

```
ftp 129.33.254.5
```

In the second example, an FTP connection is opened between your workstation and the foreign host, whose host name is specified as **bingo** in the HOSTS file in the ETC subdirectory of the local host. FTP uses a name server or the HOSTS file to translate **bingo** to a valid network address.

```
ftp bingo
```

The following is displayed:

```
Connected to Bingo.
220 Bingo ftp server (Version x.x mm/dd/yy) ready.
Name (Bingo) :
```

Enter your user ID. If the foreign host is running a TCP/IP for OS/2 FTP server, the user ID is contained in the TRUSERS file.

After you enter your user ID, you are prompted to enter a password.

```
331 Password required for bingo.
Password:
```

The password must match the password specified for the user ID in the TRUSERS file on the remote server's machine.

2.5 ftp Subcommands

You can enter **ftp** subcommands at the FTP prompt. Subcommands must be entered in lowercase.

Subcommand	Function	Page
?	Displays help information	2.6
!	Invokes the OS/2 command processor	2.7
$	Invokes FTP macros	2.8
account	Sends the account name to the foreign host	2.9.3
append	Appends a file on your workstation to a file on the foreign host	2.12.1
ascii	Sets the file transfer type to ASCII	2.11.4.1
bell	Toggles the bell mode	2.11.6.1
binary	Sets the file transfer type to binary (image)	2.11.4.2
bye	Ends the FTP session and exits the FTP command shell; synonymous with the **quit** command	2.16.2
cd	Changes the working directory on the foreign host	2.15.1
cdup	Changes the current working directory on the foreign host to the parent of the current directory on the foreign host	2.15.1
close	Ends the FTP connection but does not leave the FTP command shell; synonymous with the **disconnect** subcommand.	2.16.1
cr	Sets carriage return stripping on or off	2.11.4.3
debug	Sets debugging on or off	2.11.6.2
delete	Deletes a file on the foreign host	2.12.2
dir	Displays a listing of the files in the current working directory on the foreign host	2.12.3
disconnect	Disconnects the foreign host; synonymous with the **close** subcommand	2.16.1
form	Sets the file transfer form to file	2.11.7
get	Transfers a file from a foreign host to your workstation	2.12.4
glob	Toggles the file name expansion for **mdelete**, **mget**, and **mput**	2.11.12
hash	Toggles hashmark printing	2.11.6.3
help	Displays help information; synonymous with the **?** subcommand	2.6
lcd	Changes or displays the name of the current working directory on your workstation	2.15.2
ls	Provides a short listing of the files in the working directory on the foreign host	2.12.3
macdef	Defines a macro name	2.8
mdelete	Deletes multiple files on the foreign host	2.12.2
mget	Transfers multiple files from the foreign host	2.12.4
mkdir	Creates a directory on the foreign host	2.15.3
mode	Sets or displays the file transfer mode	2.11.8

	(only stream mode is supported)	
mput	Transfers multiple files from your workstation	2.12.6
nmap	Toggles the file name mapping mechanism	2.11.9
ntrans	Toggles the file name character translation mechanism	2.11.10
open	Establishes a connection to the foreign host	2.9.1
prompt	Toggles interactive mode	2.11.11
proxy	Forwards commands to another server to allow logical connections between two servers, which allows them to transfer files between them	2.14
put	Transfers a file from your workstation to a foreign host; synonymous with the **send** command	2.12.6
pwd	Displays the name of the current working directory on the foreign host	2.15.4
quit	Ends the FTP session and exits the FTP command shell; synonymous with the **bye** subcommand	2.16.2
quote	Sends the text following **quote** to the foreign host verbatim	2.13
recv	Copies a remote file to the local host; synonymous with the **get** subcommand	2.12.4
remotehelp	Displays the help information that is provided by the foreign host	2.6
rename	Renames a file on the foreign host	2.12.8
reset	Clears the reply queue and resets the command/reply sequencing between the local process and the foreign server	2.11.3
rmdir	Removes a directory on the foreign host	2.15.5
runique	Toggles the creation of unique file names for local destination files during **get** and **myget** operations	2.12.5
send	Transfers a file from your workstation to a foreign host; synonymous with the **put** subcommand	2.12.6
sendport	Enables automatic transmission of the FTP server **port** command	2.11.1
site	Sends information to the foreign host that is used to provide services specific to that host	2.11.2
status	Displays the current status of the FTP connection	2.10
struct	Specifies the file transfer structure	2.11.5
sunique	Toggles the creation of unique file names for foreign destination files during **put** and **mput** operations	2.12.7
trace	Toggles the packet tracing flag	2.11.6.4
type	Sets the file transfer type	2.11.4
user	Identifies you to the foreign host	2.9.2
verbose	Toggles the verbose flag	2.11.6.5

2.6 Requesting Help

To display help while you are in the FTP command shell:

- Type **?** or **help** by itself to display a list of the FTP subcommands, or type **?** or **help** with a subcommand to display help for a specific FTP subcommand.

- Use the **remotehelp** subcommand to send a request for help to an FTP server on a foreign host.

The syntax of the **?** subcommand is:

```
>>--?------------------><
       +- subcommand-+
```

subcommand Specifies the subcommand for which you are requesting help.

The syntax of the **help** subcommand is:

```
>>--help------------------><
          +- subcommand-+
```

subcommand Specifies the subcommand for which you are requesting help.

The syntax of the **remotehelp** subcommand is:

```
>>--remotehelp----------------><
               +- command-+
```

command Identifies the host command for which you want to view help information.

The **remotehelp** subcommand identifies the commands available on the remote host. You can issue the **remotehelp** subcommand with the abbreviation **rem**.

2.7 Using OS/2 Commands

You can invoke the OS/2 command shell to issue a single command and immediately return to the FTP shell. To issue a single command, include the OS/2 command and its required arguments following the **!** subcommand:

```
>>--!---------------------------------><
       +- command------------------+
                    +- parameters-+
```

command Specifies the OS/2 command that you want to issue
parameters Specifies any parameters required by the OS/2 command

If you enter the **!** subcommand without parameters, you will enter an OS/2 command shell. To return to the FTP command shell from the OS/2 command shell, type **exit**.

2.8 Defining a Macro with the macdef Subcommand

You can define a macro by using the **macdef** subcommand with the parameter **macro_name**. Subsequent lines are stored as the macro **macro_name**. A null line, which consists of consecutive new line characters in a file or a carriage return from the terminal, terminates the macro input mode. Macros can also be defined in the NETRC file.

You can create up to 16 macros with a total of 4096 characters. Macros remain defined until you issue the **close** command.

The syntax of the **macdef** subcommand is:

```
>>--macdef-------------------><
           +- macro_name-+
```

macro_name Specifies the macro name.

The **$** subcommand is used to execute a macro.

The macro processor interprets **$** and **** as special characters. A **$** followed by a number, or numbers, is replaced by the corresponding argument on the macro invocation command line. A **$** followed by an **i** signals the macro processor that the executing macro is to be looped. On the first pass, **$i** is replaced by the first argument on the macro invocation command line. On the second pass, it is replaced by the second argument, and so on. A **** followed by any character is replaced by that character. The **** prevents special treatment of the **$**.

The syntax of the **$** subcommand is:

```
>>--$-----------------------------------><
      +- macro_name------------------+
                    +- parameters-+
```

macro_name Specifies the macro name.
parameters Specifies any parameters to be passed to the macro. You can specify more than one parameter.

2.9 Establishing a Connection to a Foreign Host

If you did not specify a foreign host upon entering the FTP shell, you must specify one in the FTP command shell before you can transfer files.

The following steps establish a connection to a foreign host's FTP server:

1. Open the connection.
2. Identify yourself to the foreign host.
3. Supply a password (if required) to the foreign host.
4. Supply account information (if required) to the foreign host.

Subtopics

2.9.1 Opening a Connection to the Foreign Host

The **open** subcommand opens a connection to a specified foreign host. Use the **open** subcommand in the following situations:

- If you did not specify a host with the **ftp** command.
- If you want to open another connection after closing a connection, without leaving the FTP shell.
- If you were unable to open a connection when you specified a **host** with the **ftp** command. This situation can occur if you did not enter a host correctly or if you experienced network problems when you issued the **ftp** command.

An optional port number can be supplied; FTP then attempts to contact an FTP server at that port.

The syntax of the **open** subcommand is:

```
>>--open------------------------><
          +- host------------+
                   +- port-+
```

host Specifies the remote host to which you want to connect. If you do not specify *host*, FTP prompts you for a host.
port Specifies the destination port to which you are connecting. If you do not specify a port, you are connected by default to the well-known FTP port as specified in the services file.

If an FTP server is not running on the foreign host when you try to connect, the following message is displayed:

```
ftp: connect: Connection refused
ftp>
```

If an FTP server is running on the foreign host when you try to connect to FTP, messages similar to the following are displayed:

```
ftp> open
(to) {address}
Connected to {address}
220 bingo (Version x.x mm/dd/yy) ready.
Name (address): Chris
331 Password required for Chris.
Password:
230 User Chris Logged in.
ftp>
```

2.9.2 Identifying Yourself to the Foreign Host

After opening a connection to a foreign host, if you are not prompted to identify yourself, or if you have entered an invalid password, use the **user** subcommand to identify yourself to the foreign host. The syntax of this subcommand is:

```
>>--user------------------------------><
          +- userid----------------+
                     +- password-+
```

userid Specifies your name to the remote host.
password Specifies the password associated with your user ID.

If you omit the user ID or password, FTP prompts you for it.

You can provide for automatic logon by using the NETRC file in the ETC subdirectory. For additional information about NETRC, see "Setting Up the Environment" in topic 2.2.

2.9.3 Specifying Account Information to the Foreign Host

Some foreign hosts require account information before you can access resources. If account information is required, use the **account** subcommand to send this information to the foreign host. **account_name** can also be specified in the NETRC file.

The syntax of the **account** subcommand is:

```
>>--account---------------------><
             +- account_name-+
```

account_name Specifies the account name on the remote host. If you do not specify the account name, FTP prompts you for it in non-echo mode.

2.10 Obtaining Status Information

Use the **status** subcommand to display the following status information:

- Connection status
- Transfer mode
- Transfer type
- Form
- Structure
- Flags

The syntax of the **status** subcommand is:

```
>>--status--><
```

The **status** subcommand has no parameters.

The following is a sample response:

```
Not connected.
No proxy connection.
Mode: stream; Type: ascii; Form: nonprint; Structure; file
Verbose; on; Bell: off; Prompting: on; Globbing: on
Store unique: off; Receive unique: off
Case: off; CR stripping: on
Mtrans: off
Nmap: off
Hashmark printing: off; Use of PORT cmds: off
```

2.11 Setting Up Your System Environment

Before transferring files with FTP, you can set up specific conditions for your system environment with certain subcommands:

- **sendport** enables automatic transmission of the FTP server **port** command.
- **site** sends information to the foreign host that provides services specific to the foreign host.
- **reset** clears the reply queue and resets the command/reply sequencing between the local process and the foreign server.
- **type, ascii, binary,** and **image** specify the transfer type used during the file transfer.
- **struct** specifies the file transfer structure (the only parameter supported is file).
- **bell** toggles the sounding of a bell that signals the completion of a file transfer.
- **debug** toggles the debugging mode.
- **hash** toggles hash sign (#) printing.
- **trace** toggles the packet tracing flag.
- **verbose** toggles the verbose flag.
- **form** specifies the file transfer format (only nonprint is supported).
- **mode** specifies the file transfer mode (only stream is supported).
- **nmap** toggles file name mapping.
- **ntrans** toggles file name character translation.
- **prompt** toggles interactive mode.
- **glob** toggles file name expansion (globbing) for **mdelete**, **mget**, and **mput**.

Remember to enter the subcommands in lowercase characters.

Subtopics

2.11.1 Specifying the Use of Port Commands

The **sendport** subcommand toggles the use of FTP port instructions.

By default, FTP attempts to use port instructions. When port instructions are disabled, FTP does not use port instructions for data transfers.

The syntax of the **sendport** subcommand is:

```
>>--sendport--><
```

The **sendport** subcommand has no parameters.

2.11.2 Sending Site-Dependent Information

The **site** subcommand sends information to a foreign host that provides services specific to the foreign host system. The syntax of this subcommand is:

>>--**site**-- **parameters**--><

parameters Specifies the service-specific information. To identify services and their respective syntax specifications, issue the **remotehelp** subcommand.

2.11.3 Resynchronizing Command/Reply Sequencing

If the foreign host server has violated the **ftp** command/reply rules, you must resynchronize the command/reply sequencing between the local FTP process and the foreign FTP server.

The **reset** subcommand clears the queue and resynchronizes command/reply sequencing with the foreign FTP server. The syntax of this subcommand is:

```
>>--reset--><
```

The **reset** subcommand has no parameters.

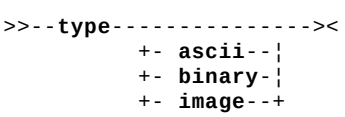
2.11.4 Setting the File Transfer Type

Use the **type** subcommand to set the file transfer type.

Set the file transfer type to match the data representations supported by the foreign host to which you are transferring files. FTP supports the following file transfer types:

- ASCII
- Binary
- Image

The syntax of the **type** subcommand is:



The setting that you specify will remain in effect until you either change it or quit FTP. If you specify **type** without a parameter, FTP will display a message indicating the current transfer type.

ascii — Specifies the file transfer type as ASCII. ASCII is the initial setting for FTP's file transfer type. Specifying the **type** subcommand with the ascii parameter is the same as issuing the **ascii** subcommand.

binary — Specifies the file transfer type as binary (image). Specifying the **type** subcommand with the binary parameter is the same as issuing the **binary** subcommand.

image — Specifies the file transfer type as image (binary). Specifying the **type** subcommand with the image parameter is the same as issuing the **image** subcommand.

Subtopics

2.11.4.1 Changing the File Transfer Type to ASCII

Use the **ascii** subcommand to change the file transfer type to ASCII. ASCII is the default file transfer type and is useful for transferring files that contain text characters. Using the **ascii** subcommand is the same as issuing the **type** subcommand with the ascii parameter.

FTP terminates each ASCII-type record with a carriage return/line feed during file transfers.

The syntax of the **ascii** subcommand is:

>>--**ascii**--><
The **ascii** subcommand has no parameters.

2.11.4.2 Changing the File Transfer Type to Binary

Use the **binary** subcommand to change the file transfer type to Binary. ASCII is the default file transfer type. Binary is useful for transferring images and executable files. Using the **binary** subcommand is the same as issuing the **type** subcommand with the binary parameter.

The syntax of the **binary** subcommand is:

>>--**binary**--><
The **binary** subcommand has no parameters.

2.11.4.3 Stripping Carriage Return Characters

The **cr** subcommand strips the carriage return character from a carriage return/line feed sequence when receiving records during ASCII-type file transfers. This conforms with the AIX(*) single-line-feed record delimiter.

Records on non-AIX foreign hosts can have single line feeds imbedded in records. The **cr** subcommand sets carriage return stripping to off to distinguish these imbedded line feeds from record delimiters. The **cr** subcommand default is on. The syntax of this subcommand is:

>>--**cr**--><
The **cr** subcommand has no parameters.

2.11.4.4 Changing the File Transfer Type to Image

Use the **image** subcommand to change the file transfer type to image. The image file transfer type is useful for binary transfers, such as transferring executable files. Using the **image** subcommand is the same as issuing the **type** subcommand with the image parameter.

The syntax of the **image** subcommand is:

The **image** subcommand has no parameters.

2.11.5 Setting the File Transfer Structure

Use the **struct** subcommand to set the file transfer structure. The syntax of this subcommand is:

```
>>--struct---- file------><
             +- record-+
```

file Specifies a file structure that is a continuous sequence of data bytes. This structure is supported for both ASCII and binary file transfer types.

record Specifies a file structure that is not currently supported by TCP/IP.

2.11.6 Setting the File Transfer Options

Use the following subcommands to set the options available while transferring a file:

- **bell**
- **debug**
- **hash**
- **trace**
- **verbose**

Subtopics
2.11.6.1 bell Subcommand
2.11.6.2 debug Subcommand
2.11.6.3 hash Subcommand
2.11.6.4 trace Subcommand
2.11.6.5 verbose Subcommand

2.11.6.1 bell Subcommand

Use the **bell** subcommand to toggle a bell sound. The bell sounds after each file transfer has completed. The default is off. The syntax of this subcommand is:

>>--**bell**--><
The **bell** subcommand has no parameters.

2.11.6.2 debug Subcommand

Use the **debug** subcommand to display the commands sent to the foreign host.

When **debug** is on, FTP displays each command sent to the foreign host, preceded by the following string:

```
      - - ->
```

Use the **debug** subcommand to toggle the debug record keeping. The default for **debug** is off.

The syntax of the **debug** subcommand is:

```
>>--debug--><
```

The **debug** subcommand has no parameters.

2.11.6.3 hash Subcommand

Use the **hash** subcommand to toggle the display of hash signs (#). When **hash** is set to on, FTP displays hashmarks to indicate data transfer progress. The syntax of this subcommand is:

>>--**hash**--><
The **hash** subcommand has no parameters.

2.11.6.4 trace Subcommand

Use the **trace** subcommand to toggle a flag that determines whether transmitted packets are traced.

If the packet tracing flag is on, packets are traced during the file transfer. If the packet tracing flag is off, packets are not traced and packet tracing messages are not displayed. The default mode for **trace** is off.

The syntax of the **trace** subcommand is:

>>--**trace**--><
The **trace** subcommand has no parameters.

2.11.6.5 verbose Subcommand

Use the **verbose** subcommand to toggle a flag that determines whether all responses from the FTP server are displayed to the user.

If the verbose flag is on, statistics regarding the efficiency of the transfer are reported when a file transfer is completed. If the verbose flag is off, statistics are not displayed about the file transfer. The default for **verbose** is on.

The syntax of the **verbose** subcommand is:

>>--**verbose**--><
The **verbose** subcommand has no parameters.

2.11.7 Specifying the File Transfer Format

Use the **form** subcommand to specify the file transfer format. The syntax of this subcommand is:

```
>>--form-- format--><
```

format Specifies the file transfer format (only nonprint is supported).

2.11.8 Specifying the File Transfer Mode

Use the **mode** subcommand to specify the file transfer mode. The syntax of this subcommand is:

```
>>--mode-- mode_name--><
```

mode_name Specifies the file transfer mode (only stream is supported).

2.11.9 Setting File Name Mapping

Use the **nmap** subcommand to toggle file name mapping. If parameters are not specified, file name mapping is turned off. The default is off.

The **nmap** subcommand is useful when you connect to a foreign host that uses file-naming conventions or practices that are different from OS/2. The syntax of this subcommand is:

```
>>--nmap------------------------------><
          +- inpattern-- outpattern-+
```

inpattern Specifies the character pattern of the file names.
outpattern Specifies the character pattern of the remote host file names.

If you specify parameters and set the mapping feature to on, characters in outgoing file names are mapped when the **put** and **mput** commands are issued without a remote destination file name, and characters in local file names are mapped when the **get** and **mget** commands are issued without a local destination file name.

inpattern is a template to be used for incoming file names. The file names may have already been processed by the **ntrans** setting. Variable templating is accomplished by including the sequences $1, $2,...,$9 in **inpattern**. All other characters are treated literally. Use the \ character to prevent special treatment of the $ character **inpattern**. For example, if you have given the **inpattern** $1.$2 and the remote file name MYFILE.DAT, $1 has the value MYFILE and $2 has the value DAT.

outpattern determines the resulting mapped file name. The sequence $1, $2,...,$9 is replaced by the value resulting from the **inpattern** template. The sequence $0 is replaced by the original file name. Additionally, the sequence [**seq1, seq2**] is replaced by **seq1** if **seq1** is not null, and replaced by **seq2** if **seq1** is null. Use the \ character to prevent special treatment of the $, [,], and , characters **outpattern**. For example, the command **nmap** $1.$2.$3 [$1,$2] .[$2,FILE] yields the output file name MYFILE.DAT for input file names MYFILE.DAT and MYFILE.DAT.OLD. The same **nmap** command yields the output file name MYFILE.FILE for the input file name MYFILE and DAT.DAT for the input file name .DAT.

2.11.10 Setting File Name Character Translation

Use the **ntrans** subcommand to toggle file name character translation. If parameters are not specified, file name character translation is turned off. The default is off.

The **ntrans** subcommand is useful when you connect to a foreign host that uses file-naming conventions or practices that are different from OS/2.

If parameters are specified, characters in outgoing file names are translated during **put** and **mput** commands that are issued without a remote destination name. Characters in incoming file names are translated for **get** and **mget** commands that are issued without a specified local destination file name.

The syntax of the **ntrans** subcommand is:

```
>>--ntrans-------------------------------><
            +- inchars----------------+
                        +- outchars-+
```

inchars Specifies the character for the file name on the workstation
outchars Specifies the character for the remote file name

Characters in a file name matching a character in **inchars** are replaced with the corresponding character in **outchars**. If the character's position in **inchars** is greater than the length of **outchars**, the character is deleted from the file name. For example, if you are transferring files from a machine that accepts the character ":" to a machine that does not accept that character, you can substitute another character ($) by entering:

```
ntrans : $
```

2.11.11 Interactive Prompting

Use the **prompt** subcommand to toggle interactive prompting. Interactive prompting occurs during multiple file transfers to allow you to selectively retrieve, store, or delete files. If prompting is on (the default), FTP prompts for verification before retrieving, sending, or deleting multiple files during **mget**, **mput**, and **mdelete** operations. If prompting is off, any **mget** or **mput** operation transfers all files; the **mdelete** operation deletes all files.

The syntax of the **prompt** subcommand is:

>>--**prompt**--><
The **prompt** subcommand has no parameters.

2.11.12 Expanding File Names on Multiple Files

Use the **glob** subcommand to toggle the expansion (translation) of file names. Globbing is the term used to designate expansion of the file names into the actual file names of the deleted or copied files. The **glob** subcommand expands file names when using **mdelete**, **mget**, or **mput**.

File-name expansion (globbing) allows you to use patterns to specify file names. The default is on. If the file-name expansion feature is turned off, you cannot use patterns with **mdelete**, **mget** or **mput**. Also, if file-name expansion is turned off, the asterisk and question mark (?) characters are treated literally. This allows FTP to transfer files with these patterns in their file names to other systems that use different file-naming conventions.

The syntax of the **glob** subcommand is:

```
>>--glob--><
```

The **glob** subcommand has no parameters.

2.12 Working with Files

The following **ftp** subcommands enable you to work with files on your workstation and the foreign host.

- **append** performs append functions on files on the foreign host.
- **delete** and **mdelete** delete the files on the foreign host.
- **dir** and **ls** display lists of files on the foreign host.
- **get, mget,** and **recv** transfer files from the foreign host to your workstation.
- **runique** toggles the renaming of a file with a unique name. If the name of the file being received is the same as a file that already exists on your host, your existing file is overwritten by the incoming file, unless **runique** is on. If **runique** is on, a unique file name is created for the incoming file and your existing file is unchanged. **runique** is used with **get** and **mget.**
- **put, mput,** and **send** transfer files from your workstation to the foreign host.
- **sunique** toggles the renaming of a file with a unique name. If the name of the file sent is the same as a file that already exists on the destination host, the existing file is overwritten by the transferred file, unless **sunique** is on. If **sunique** is on, a unique file name is created for the transferred file and the existing file is unchanged. **sunique** is used with **put** and **mput.**
- **rename** allows you to rename files on the foreign host.

Remember to enter the subcommands in lowercase characters.

Subtopics

2.12.1 Appending a File

The **append** subcommand transfers a file from your workstation to the foreign host and appends the file to a specified file or to a file of the same name.

To append data to a file on the foreign host, you must have a defined working directory on that foreign host, and you must have write privileges to this working directory.

The syntax of the **append** subcommand is:

```
>>--append-------------------------------------------><
            +- source_file------------------------+
                           +- destination_file-+
```

source_file Specifies the name of the file on your workstation that is to be transferred and appended to a file on the remote host. If you do not specify this value, FTP will prompt you for it.

destination_file
Specifies the name of the remote host file to which you want to append the **source_file**.

If neither **source_file** nor **destination_file** is specified with the **append** subcommand, FTP prompts you to supply both parameters. If only **source_file** is specified with the **append** subcommand, the source file is transferred and appended to a file of the same name on the foreign host.

In the following example, the file named AUGUST.RPT is transferred from your workstation to the foreign host, where it is appended to the file named REPORT90, which resides in the current working directory.

```
append august.rpt report90
```

When using **append** with a foreign host running a TCP/IP for OS/2 FTP server, if you specify a file that does not exist, the file is created on the foreign host and the data is transferred.

2.12.2 Deleting a File on the Foreign Host

Two subcommands, **delete** and **mdelete**, delete files on the foreign host.

Use the **delete** subcommand to delete a single file on the foreign host. The syntax of this subcommand is:

```
>>--delete------------------><
            +- file_name-+
```

file_name Specifies the name of the file to be deleted from the remote host. If you do not specify this value, FTP will prompt you for it.

If you omit **filename**, FTP prompts you for the name.

In the following example, the file named AUGUST.RPT is deleted on the foreign host.

```
delete august.rpt
```

Use the **mdelete** subcommand to delete multiple files on the foreign host. Globbing (**glob**) must be on before you can use **mdelete**. The syntax of this subcommand is:

```
             <------------+
>>--mdelete------------------><
              +- pattern-+
```

pattern Specifies the name pattern of the files to be deleted from the remote host. Patterns are any combination of ASCII characters. The following two characters have special meaning:

- `*` The asterisk means that any character or group of characters can occupy that position in the pattern.
- `?` The question mark means that any single character can occupy that position in the pattern.

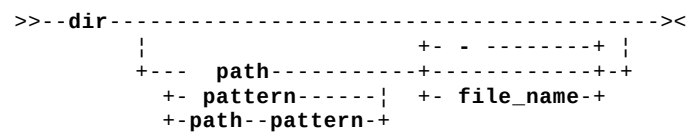
You can specify more than one pattern with the **mdelete** subcommand. If you do not specify this value, FTP will prompt you for it.

The following example deletes all files with names that have five characters, begin with ab, have any character in the third position followed by de, and have an extension of xyz, assuming file-name expansion (globbing) is on.

```
mdelete ab?de.xyz
```

2.12.3 Displaying a List of Files

Use the **dir** subcommand to display a list of the files and directories in the directory with auxiliary information. The syntax of this subcommand is:



path Specifies a path to a different directory, a specific file, or both.

pattern Specifies the pattern of the file names to be listed. Patterns are any combination of ASCII characters. The following two characters have special meaning:

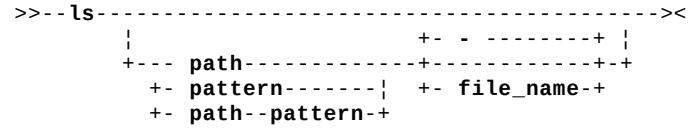
- `*` The asterisk means that any character or group of characters can occupy that position in the pattern.
- `?` The question mark means that any single character can occupy that position in the pattern.

file_name Specifies the name of a file on your workstation to which you want to write the output. If you specify a file name, you must also specify a path or pattern. If you do not specify a file name or if you specify a hyphen (-), the output is displayed on the screen.

The following example shows the command that lists the files and subdirectories in the directory **mydir** and writes that information to the file **myfiles.txt** on the workstation.

```
dir c:\mydir myfiles.txt
```

Use the **ls** subcommand to display a list containing only the file names in the directory. The syntax of this subcommand is:



path Specifies a path to a different directory, specific file, or both.

pattern Specifies the pattern of the file names to be listed. Patterns are any combination of ASCII characters. The following two characters have special meaning:

- `*` The asterisk means that any character or group of characters can occupy that position in the pattern.
- `?` The question mark means that any single character can occupy that position in the pattern.

file_name Specifies the name of a file on your workstation to which you want to write the output. If you specify a file name, you must also specify a path or pattern. If you do not specify a file name, or if you specify a hyphen (-), the output is displayed on the screen.

The following example shows the command that lists only the files in the directory **mydir** beginning with **my** and having a **txt.**extension. In the example, **my*.txt** is the file name pattern.

```
ls c:\mydir\my*.txt
```

2.12.4 Copying Files to Your Workstation

Use the **get** and **recv** subcommands to receive a file from a foreign host on to your workstation. The current settings for **type** and **struct** are used with the **get** subcommand.

The syntax of the **get** subcommand is:

```
>>--get-----------------------------------------><
         +- source_file-----------------------+
                        +- destination_file-¦
                        +- con--------------¦
                        +- prn--------------+
```

source_file Specifies the name of the file on the remote host that is to be transferred to your workstation. If you do not specify this value, FTP will prompt you for it.

destination_file
Specifies the name given to the source file when it is stored on your workstation. If **destination_file** is not specified, the **source_file** name is used and changed, if necessary, to conform to OS/2 file-naming conventions. If the name of the file being received is the same as a file that already exists on your workstation, your existing file is overwritten by the incoming file, unless **runique** is set to on. If **runique** is set to on, a unique file name is created for the incoming file, and your existing file is unchanged. If you are unsure whether **runique** is set to on, use the **status** subcommand to check the setting.

con Specifies that the file is to be displayed on your screen.

prn Specifies that the file is to be sent to a printer or special device.

You can use the **recv** subcommand to receive a file from a foreign host. The syntax of this subcommand is:

```
>>--recv----------------------------------------><
          +- source_file-----------------------+
                         +- destination_file-¦
                         +- con--------------¦
                         +- prn--------------+
```

source_file Specifies the name of the file on the remote host that is to be transferred to your workstation. If you do not specify this value, FTP will prompt you for it.

destination_file
Specifies the name given to the **source_file** when it is stored on your workstation. If **destination_file** is not specified, the **source_file** name is used and changed, if necessary, to conform to OS/2 file-naming conventions. If the name of the file being received is the same as a file that already exists on your workstation, your existing file is overwritten by the incoming file, unless **runique** is set to on. If **runique** is set to on, a unique file name is created for the incoming file, and your existing file is unchanged. If you are unsure whether **runique** is set to on, use the **status** subcommand to check the setting.

con Specifies that the file is to be displayed on your screen.

prn Specifies that the file is to be sent to a printer or special device.

You can use the **mget** subcommand to copy multiple files from the foreign host to your workstation. Globbing (**glob**) must be on before you can use **mget**.

The syntax of the **mget** subcommand is:

```
          <------------+
>>--mget-----------------><
           +- pattern-+
```

pattern Specifies the name pattern of the files to be transferred from the remote host to your workstation. Patterns are any combination of ASCII characters. The following two characters have special meaning:

`*` The asterisk means that any character or group of characters can occupy that position in the pattern.

`?` The question mark means that any single character can occupy that position in the pattern.

You can specify more than one pattern with the **mget** subcommand. If you do not specify this value, FTP will prompt you for it.

If the prompt flag is toggled on by the **prompt** subcommand, the **mget** subcommand prompts you before transferring any files.

Any response to the **mget** subcommand prompt that begins with N means No, and the file is not transferred. **mget** prompts you for the next file to be transferred. Any other response means Yes, and the file is transferred and stored with the current host name. You are prompted for the next file to match the pattern.

The **quit** subcommand ends the **mget** operations without transferring any more files.

2.12.5 Assigning Unique File Names on Your workstation

Use the **runique** subcommand to create unique file names for local destination files during **get** and **mget** operations.

The syntax of the **runique** subcommand is:

>>--**runique**--><
The **runique** subcommand has no parameters.

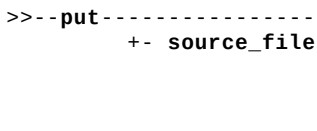
If **runique** is off (the default) and if the name of the file received is the same as a file that already exists on your host, FTP overwrites local files.

If **runique** is on, a unique file name is created for the incoming file and your existing file is unchanged.

2.12.6 Copying Files to the Foreign Host

Use the **put** or **send** subcommand to transfer a file from your workstation to the foreign host.

The syntax of the **put** subcommand is:



source_file Specifies the name of the file on your workstation that is to be transferred to the remote host. If you do not specify this value, FTP will prompt you for it.

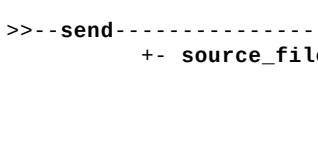
destination_file
Specifies the name given to the **source_file** when it is stored on the remote host. If **destination_file** is not specified, the **source_file** name is used and changed, if necessary, to conform to OS/2 file-naming conventions. If the name of the file being received is the same as a file that already exists on the remote host, the existing file is overwritten by the incoming file, unless **sunique** is on. If **sunique** is on, a unique file name is created for the incoming file, and the existing file is unchanged.

con Specifies that the file is to be displayed on the server screen.

prn Specifies that the file is to be sent to a destination printer or special device.

The **put** subcommand is synonymous with the **send** subcommand.

The syntax of the **send** subcommand is:



source_file Specifies the name of the file on your workstation that is to be transferred to the remote host. If you do not specify this value, FTP will prompt you for it.

destination_file
Specifies the name given to the **source_file** when it is stored on the remote host. If **destination_file** is not specified, the **source_file** name is used and changed, if necessary, to conform to OS/2 file-naming conventions. If the name of the file being received is the same as a file that already exists on the remote host, the existing file is overwritten by the incoming file, unless **sunique** is on. If **sunique** is on, a unique file name is created for the incoming file, and the existing file is unchanged.

con Specifies that the file is to be displayed on the server screen or console.

prn Specifies that the file is to be sent to a destination printer or special device.

If you specify neither **source_file** nor **destination_file**, FTP prompts you for both parameters.

The current settings for **type** and **struct** are used with the **put** and **send** subcommands.

If you use the **put** or **send** subcommand and the foreign host already has a file with the name **destination_file** and **sunique** is off, the foreign host overwrites the existing file. If the foreign host does not have a file with the name **destination_file**, the foreign host creates a new file with that name.

Use the **mput** subcommand to copy multiple files from your workstation to a foreign host.

Note: Globbing (**glob**) must be on before you can use **mput**.

The syntax of the **mput** subcommand is:

```
          <------------+
>>--mput------------------><
           +- pattern-+
```

pattern Specifies the name pattern of the files to be copied from your workstation to the remote host. Patterns are any combination of ASCII characters. The following two characters have special meaning:

* The asterisk means that any character or group of characters can occupy that position in the pattern.

? The question mark means that any single character can occupy that position in the pattern.

You can specify more than one pattern with the **mput** subcommand. If you do not specify this value, FTP will prompt you for it.

If the prompt flag is toggled on by the **prompt** subcommand, the **mput** subcommand prompts you before transferring any files.

Any response to the **mput** subcommand prompt that begins with N means No, and the file is not transferred. **mput** prompts you for the next file to be transferred. Any other response means Yes, and the file is transferred and stored with the current host name. You are prompted for the next file to match the pattern.

The **quit** subcommand ends the **mput** operation without transferring any more files.

2.12.7 Assigning Unique File Names on the Foreign Host

Use the **sunique** subcommand to create unique file names for foreign destination files during **put** and **mput** operations. The syntax of this subcommand is:

```
>>--sunique--><
```

The **sunique** subcommand has no parameters.

If the name of the file being sent already exists on the destination host, the existing file is overwritten by the transferred file, unless **sunique** is on. If **sunique** is on, a unique file name is created for the transferred file and the existing file is unchanged. The unique file name is displayed on the screen when the transfer is complete.

2.12.8 Renaming a File on the Foreign Host

Use the **rename** subcommand to rename a file on the foreign host. The syntax of this subcommand is:

```
>>--rename------------------------------><
            +- oldname---------------+
                        +- newname-+
```

oldname Specifies the current name of a file in the working directory of the remote host.

newname Specifies the new name for the file. If the file name specified by the **newname** already exists, an error message is displayed.

If you do not specify either of these parameters, FTP will prompt you for them.

2.13 Sending Arguments to the Foreign Host

Use the **quote** subcommand to send the specified argument verbatim to the FTP server on the foreign host. The syntax of this subcommand is:

```
>>--quote-----------------><
          +- argument-+
```

argument Specifies the information to send to the remote host. If you do not specify this value, FTP will prompt you for it.

Note: Using **quote** with commands that involve data transfers can produce unpredictable results.

2.14 Opening a Connection between Two Foreign Hosts

Use the **proxy** subcommand to forward subcommands to another server to allow logical connections between two servers. Using the **proxy** subcommand enables the two servers to transfer files between them.

The syntax of the **proxy** subcommand is:

```
>>--proxy------------------><
           +- subcommand-+
```

subcommand Specifies an FTP subcommand. If you do not specify this value, FTP will prompt you for it.

Use the **open** subcommand with **proxy** to establish a secondary control connection.

Use the **?** subcommand with **proxy** to display the other **ftp** subcommands that are executable on the secondary connection.

Note: The following subcommands behave differently when used with **proxy.**

- **open** does not define new macros during the automatic login process when prefaced by **proxy**.
- **close** does not erase existing macro definitions when prefaced by **proxy**.
- **get** and **mget** transfer files from the host on the primary connection to the host on the secondary connection when prefaced by **proxy.**
- **put, mput,** and **append** transfer files from the host on the secondary connection to the host on the primary connection when prefaced by **proxy.**

Warning: If you use **proxy put** with a nonexistent file, you disconnect the FTP session. A file is created with 0 size with the file name of the primary host. You cannot erase the file unless the FTP server of the host is closed.

2.15 Working with Directories

ftp has several subcommands for working with directories on your workstation and the foreign host. The **ftp** subcommands that enable you to work with directories are:

- **cd** and **cdup** changes the working directory on the foreign host.
- **lcd** changes the current working directory on your workstation.
- **mkdir** creates a directory on the foreign host.
- **pwd** displays the name of the working directory on the foreign host.
- **rmdir** removes an existing directory on the foreign host.

Subtopics

2.15.1 Changing the Current Working Directory on the Foreign Host

Use the subcommands **cd** and **cdup** to change the current working directory on the foreign host.

Use the **cd** subcommand to change the current working directory on the foreign host. The syntax of this subcommand is:

```
>>--cd-----------------------><
        +- directory_name-+
```

directory_name Specifies the name of the file directory on the remote host that becomes the current working directory for file transfer tasks. If you do not specify this value, FTP will prompt you for it.

In the following example, the directory named **\abc\def** from the D drive becomes the current working directory when communicating with a TCP/IP for OS/2 FTP server.

```
      cd d:\abc\def
```

Use the **cdup** subcommand to change the current working directory on the foreign host to the parent directory of the current working directory on the foreign host. The parent directory is the directory directly above the current directory.

The syntax of the **cdup** subcommand is:

```
>>--cdup-->
```

The **cdup** subcommand has no parameters.

2.15.2 Changing the Current Working Directory on Your Workstation

Use the **lcd** subcommand to change the current working directory on your workstation. The syntax of this subcommand is:

```
>>--lcd-------------------><
         +- local_path-+
```

local_path Specifies the name of a directory on your workstation that you want to make your current directory. If you do not specify a local path, the name of the current working directory on your workstation is displayed.

In the following example, the directory named REPORTS on the C drive of your workstation becomes the current working directory.

```
lcd c:\reports
```

2.15.3 Creating a Directory on the Foreign Host

Use the **mkdir** subcommand to create a directory on the foreign host.

```
>>--mkdir------------------><
           +- directory-+
```

directory Specifies the path to the directory that you are creating. If you do not specify a directory, FTP prompts you for the path.

2.15.4 Displaying the Current Working Directory Name on the Foreign Host

Use the **pwd** subcommand to display the name of the current working directory on the foreign host. The syntax of this subcommand is:

```
>>--pwd--><
```

The **pwd** subcommand has no parameters.

2.15.5 Removing a Directory from the Foreign Host

Use the **rmdir** subcommand to remove a directory from a foreign host. The syntax of this subcommand is:

```
>>--rmdir------------------><
           +- directory-+
```

directory Specifies the directory that you want to remove from the remote host. If you do not specify a directory, FTP prompts you for one.

2.16 Ending the FTP Session

You can end a session with an FTP server by using one of the following subcommands:

- **close** or **disconnect** terminates the FTP session and returns control to the FTP command shell.

- **bye** or **quit** terminates the FTP session and exits the FTP command shell.

Subtopics
2.16.1 Disconnecting from the Foreign Host
2.16.2 Leaving the FTP Command Shell

2.16.1 Disconnecting from the Foreign Host

After opening a connection to the foreign host, you cannot connect to another foreign host without closing the current session. Use the **close** or **disconnect** subcommand to disconnect from the foreign host.

The syntax of the **close** subcommand is:

```
>>--close-->
```

The syntax of the **disconnect** subcommand is:

```
>>--disconnect-->
```

The **close** and **disconnect** subcommands have no parameters.

After closing a connection to the foreign host with the **close** or **disconnect** subcommand, you remain in the FTP command shell. Issue the **open** subcommand to open a connection to a different foreign host.

2.16.2 Leaving the FTP Command Shell

Use the **quit** or **bye** subcommands to close any open connection and leave the FTP environment.

The syntax of the **quit** subcommand is:

>>--**quit**--><

The syntax of the **bye** subcommand is:

>>--**bye**--><

The **quit** and **bye** subcommands have no parameters.

2.17 Ending the Server Task

Follow these steps to end the FTP server task.

1. Select the window in which the server is running.
2. Press the Ctrl and C keys at the same time.

3.0 Chapter 3. Transferring Files with FTPPM

This chapter describes how to use the File Transfer Protocol Presentation Manager (FTPPM) program. FTPPM is a Presentation Manager application that uses the File Transfer Protocol (FTP) to perform the functions of an FTP client. FTPPM supports a maximum of 256 simultaneous connections with remote hosts. FTPPM uses standard Presentation Manager input and output conventions. If you need further assistance to understand a menu selection, place the cursor on the selection and press the F1 key.

Subtopics

3.1 Overview of FTPPM

FTPPM transfers files between your workstation and a remote host that is running an FTP server. Specifically, FTPPM enables you to:

- Request help
- Establish a connection between multiple remote hosts
- Identify yourself to the remote hosts
- Transfer files between your workstation and a remote host
- Transfer files between two remote hosts to which you are connected
- Delete files on a remote host
- Work with directories on your workstation and on a remote host

3.2 FTPPM Clients and the OS/2 Workplace Shell

The TCP/IP for the OS/2 FTPPM client is integrated with the OS/2 desktop. You can use the FTPPM template located on your desktop to create icons that represent instances of FTPPM sessions without learning or using the FTPPM commands and parameters. Once you create and configure an icon for a particular session, you can double-click on it with your mouse to start the session automatically.

The OS/2 desktop includes a templates folder located in the OS/2 System folder, unless it has been moved. When TCP/IP for OS/2 is installed, it adds an FTPPM template to this folder. If you do not find the FTPPM template in the templates folder, type **tcpdsktp** on the command line of an OS/2 session and the template will be created.

Use your mouse to create an instance of an FTPPM client by following these steps:

1. Move your mouse pointer to the FTPPM template, then press and hold down mouse button two.

2. While holding down mouse button two, move the mouse pointer to a desktop location where you want to create the instance for the FTPPM client, then release the mouse button. The FTPPM Settings notebook appears.

The FTPPM Settings notebook allows you to specify a title and an icon. Using the Settings notebook, you specify the name of the remote host, the user, the password, account, and other options that govern the session. All the options in the FTPPM Settings notebook are optional, and an FTPPM instance you create will work without your having to change anything in the notebook. In most cases, however, you will want to specify at least the name of the remote host with which you want to connect.

The first page is the Host page. The hostname field specifies the remote host to which you want to connect. You can specify the host name, IP address, or an alias. If you leave this field blank, you are prompted to type the host name when you start the FTPPM instance.

The next page is the Authentication page. On this page, you can specify the user, password, and account to be used on the remote host.

The Options page lets you specify:

- The codepage

  When FTP transfers files in ASCII format, there must sometimes be a translation between the ASCII representation used on the workstation and the version recognized by other hosts on the network. This translation is most important for special characters used in languages other than English. Two codepages are provided for use with TCP/IP for OS/2:

  - TCPDECMU for the DEC multinational codepage
  - TCP8859 for the ISO codepage 8859-1
  - NONE to indicate that translation is not required. (The text is passed to the network in workstation representation.)

  If you do not specify a codepage, translation is not done unless a default codepage has been set using the TCP/IP Configuration notebook, TCPIPCFG.

  If you want the TCPDECMU or TCP8859 codepage to be used, without having to specify a codepage, you can set the default codepage with the Configuration notebook. The default codepage is set in the Services section of the notebook. If you do set a default codepage, it is used not only by FTPPM, but by FTP, Telnet, and TelnetPM.

- The name of the local directory
- The name of the remote directory
- The file mask for the local directory
- The file mask for the remote directory
- The following flags:

store unique	Whether to store remote files with unique file names
use code page	Whether to use the specified code page

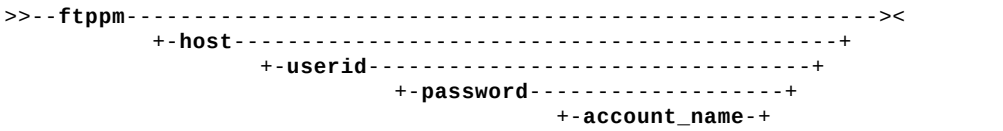
get unique	Whether to store local files with unique file names
prompt	Whether you are prompted for confirmation to transfer files

The General page of the Settings notebook lets you specify the icon title and the icon used for the FTPPM instance.

You can use the F1 key anytime to get help with the controls in the Settings notebook. After completing the Settings notebook, double-click on the notebook's system menu icon to close the FTPPM Settings notebook. You can then use this FTPPM instance to start the FTPPM session anytime by double-clicking on it.

3.3 Starting FTPPM

You can start FTPPM by entering the **ftppm** command. The syntax of this command is:



host The first remote host to which you are connecting

userid The name associated with you by the remote host to which you are establishing a connection

password Your password, which is associated with your user ID by the remote host to which you are establishing a connection

account_name
The account information required by the host

If you are using a NETRC file that contains a valid entry for the remote host, you can omit the user and password parameters.

If a connection is established, the FTPPM window appears. A file list from the specified host is displayed in the window. The remote host name is displayed at the top left corner of the file list. If FTPPM is unable to successfully connect, the Remote Logon window appears. You can alter the information or cancel the logon attempt. See "Working with Connections Using the Remote Menu" in topic 3.4 for more information about remote logon.

3.4 Working with Connections Using the Remote Menu

If you did not specify a remote host in the **ftppm** command or if you want to establish connections with additional remote hosts, select the Logon option from the Remote pull-down menu.

Figure 1 is an example of the Remote Logon window.

PICTURE 1

Figure 1. Example of the Remote Logon Window

Use the Tab key to move from each entry field to the next and from the Account entry field to the pushbuttons.

Note: For security purposes, the account and password information is not displayed as it is entered.

The following pushbuttons are displayed at the bottom of the window.

Pushbutton	**Description**
OK	Attempts to establish a connection with the information provided
Cancel	Cancels a logon attempt and returns to the FTPPM window

If you are using a NETRC file that contains a valid entry for the remote host, you can omit the userid and password parameters. FTPPM searches the NETRC file to locate these parameters. Any macros defined in the NETRC file are ignored. See Chapter 2, "Transferring Files with the File Transfer Protocol (FTP)" in topic 2.0, for more information about the NETRC file.

When a connection is established, you are returned to the FTPPM window, and the new remote host becomes the current remote host. A file list from this host replaces the file list of any previous connection. The current remote host name is displayed at the top left corner of the file list. To toggle the current remote host, see "Selecting a Current Connection" in topic 3.4.1.

Subtopics
3.4.1 Selecting a Current Connection
3.4.2 Closing All Current Connections

3.4.1 Selecting a Current Connection

If you are connected to two or more remote hosts, you can select the Select Remote Host option on the Remote pull-down menu to change the current remote host.

When you select this option, the Current Hosts window displays a list of the remote hosts to which you are currently connected. Select the remote host that you want to make current. Figure 2 is an example of the Current Hosts window.

PICTURE 2

Figure 2. Example of the Current Hosts Window

Use the mouse or the Tab key to move to the following pushbuttons at the bottom of the window.

Pushbutton	**Description**
OK	Changes the remote host to your selection and returns to the FTPPM window
Cancel	Cancels a remote host change and returns to the FTPPM window

3.4.2 Closing All Current Connections

Select the Close All Connections option from the Remote pull-down menu to disconnect from all remote hosts.

3.5 Marking and Unmarking Files or Directories

You can select the following actions for marked local files and directories:

- Delete the set of marked local files and remove empty, marked directories using the Delete local files option on the Action menu.
- Transfer the group of marked local files and all the files in a marked directory to the current remote host using the Put local files option on the Action menu.

You can select the following actions for marked remote files and directories:

- Delete the set of marked remote files and remove empty, marked directories using the Delete remote files option on the Action menu.
- Transfer the group of marked remote files and all the files in a marked directory to your workstation using the Get remote files option on the Action menu.
- Transfer the group of marked remote files and all the files in a marked directory to another remote host using the Send files between remote hosts option on the Action menu.

Subtopics

3.5.1 Marking Local Files and Directories

To mark a local file or directory, either use the mouse to select the file or directory, or use the arrow keys to position the cursor on the file or directory name and press the space bar.

3.5.2 Marking Remote Files and Directories

To mark a remote file or directory, either use the mouse to select the file or directory, or use the arrow keys to position the cursor on the file or directory and press the space bar to highlight (mark) the name.

The files and directories listing shows the file name and other information about the file. Mark **only** the file name or directory name.

Some FTP servers provide file and directory listings in which the file names are displayed with blank spaces separating the file name and the file extension. When marking files displayed with blank spaces, you must mark both parts of the file name separately. Selecting the Two Part File Names option from the Options pull-down menu enables you to mark both parts of the file name by selecting only the left part. Selecting the Free Form File Names option from the Options pull-down menu enables you to mark any portion of the line. Place the cursor over the starting character, then press and hold the left mouse button while dragging the cursor. Release the left mouse button to stop highlighting the text.

3.5.3 How to Unmark Files and Directories

If you select the Free Form File Names option, click on the right mouse button to unmark a file or directory. Otherwise, click on the left mouse button to unmark a file or directory. Another way to unmark an item is to select it a second time with the arrow keys and the space bar.

Choosing the Update option unmarks all marked files. In addition, after an action is performed on a group of marked files, they become unmarked.

3.6 Changing the Current Working Directory

This section describes how to change the current working directory on your workstation or on the remote host.

Subtopics

3.6.1 On Your Workstation

You can use the following methods to change the working directory displayed on your workstation:

- Move the cursor to the directory box. Select the subdirectory or drive that you want to be the new current working directory.

- Use the Tab key to move to the Local Directory is: entry field. Type over the displayed name with a new directory name and press the Enter key.

3.6.2 On the Remote Host

You can use the following methods to change the working directory displayed on the remote host:

- Use the Tab key to move to the directory and files list. Position the cursor on the name of the directory or drive that you want to be the new current working directory. Press the Enter key or double-click on this directory name. If you have not positioned the cursor on a directory or drive name, you will get an error message.

- Use the Tab key to move to the field labeled Remote Directory is. Type over the displayed name with the new directory name and press the Enter key.

3.6.3 Setting a Mask for Displayed Files

Use the Tab key to move to the Local Mask or Remote Mask field. Enter a pattern for the files to be displayed. (A pattern is any combination of ASCII characters.)

The following characters have special meaning:

`*` Specifies that any character or group of characters can occupy that position in the pattern

`?` Specifies that any single character can occupy that position in the pattern

3.7 Setting Options Using the Options Menu

Before transferring files with FTPPM, you can change the type of transfer and specify other environmental conditions. The following options can be changed:

- File transfer type (ASCII or binary)
- Interactive prompting
- Assignment of unique local file names
- Assignment of unique remote file names
- Marking two-part remote file names
- Marking free-form file names
- Tracing Comments and Replies

You can change these options using the Options pull-down menu. A check mark beside the option indicates that the option is set. You can save the FTPPM settings you selected in this menu when you exit FTPPM.

Subtopics

3.7.1 Setting the File Transfer Type

Select the Options pull-down menu. Under the Transfer Types heading, the following options are displayed.

Option	**Description**
ASCII	ASCII is useful for transferring files that contain text characters.
Binary	Binary is useful for image transfers, such as transferring executable files.

Select the ASCII or Binary option from the menu. A check mark appears to the left of your choice.

3.7.2 Interactive Prompting

Interactive prompting occurs during multiple file transfers and enables you to selectively send or receive marked files. If Interactive Prompting is enabled, a window appears before each marked file is transferred or deleted. Figure 3 in topic 3.7.2 shows the Get Prompt window.

Note: The Put Prompt and the Proxy Put Prompt windows contain a similar entry field and similar pushbuttons.

PICTURE 3

Figure 3. Example of the Get Prompt Window

You can specify a different target file name by typing over the default file name in the entry field.

Use the Tab key to move to the following pushbuttons displayed at the bottom of the window:

Pushbutton	**Description**
Yes	Proceeds to transfer the file to the target host, giving it the name specified in the To File entry field
No	Does not transfer the file
Cancel	Cancels transferring all the marked files and returns to the FTPPM window

If enabled, interactive prompting also occurs during multiple file deletions. The Remote Delete Prompt window (Figure 4) enables you to individually confirm the deletion of each marked file.

PICTURE 4

Figure 4. Example of the Remote Delete Prompt Window

Use the Tab key to move between the following pushbuttons.

Pushbutton	**Description**
Yes	Deletes the file
No	Does not delete the file
Cancel	Cancels deleting all the marked files and returns to the FTPPM window

Select the Interactive Prompting option from the Options pull-down menu to enable or disable interactive prompting. A check mark to the left of the option indicates that it is set. The setting you select is saved after each session.

3.7.3 Assigning Unique File Names on Your Workstation

The Assign Unique Local File Names option from the Options pull-down menu enables the creation of unique file names for local destination files during Get Remote Files operations.

If the incoming file name already exists on your host, your existing file is overwritten by the incoming file, unless you have enabled this option. If it is enabled, a unique file name is created for the incoming file, and your existing file is unchanged.

Select the Assign Unique Local File Names option from the Options pull-down menu to enable or disable the assigning of unique local file names. A check mark to the left of the option indicates that it is set.

3.7.4 Assigning Unique File Names on the Remote Host

The Assign Unique Remote File Names menu choice from the Options pull-down menu enables the creation of unique file names for foreign destination files during Put Local Files operations.

If the outgoing file name already exists on the destination host, the existing file is overwritten by the outgoing file, unless you have enabled this option. If it is enabled, a unique file name is created for the transferred file and the existing file is unchanged.

Select the Assign Unique Remote File Names option from the Options pull-down menu to enable or disable the assigning of unique remote file names. A check mark to the left of this option indicates that it is set.

3.7.5 Setting the Two Part or Free Form File Names Option

These options help you mark file names with imbedded blanks, as explained in "How to Unmark Files and Directories" in topic 3.5.3.

A check mark to the left of the option indicates that it is set.

3.7.6 Tracing Commands/Replies

If you select the Trace Commands/Replies option from the Options pull-down menu, you can trace your FTPPM session activities. The commands you issue and the replies you receive are written to a file named FTPPM.TRC in the ETC subdirectory or the directory specified by the ETC environment variable.

The trace file is overlayed (information is not appended) each time the trace option is selected.

3.8 Working with Files and Directories Using the Actions Menu

This section describes how to use the Actions menu to:

- Send, receive, and delete marked files
- Make directories
- Rename files on your workstation and on the remote host
- send site and quote commands to the remote host.

Subtopics

3.8.1 Transferring Files from Your Workstation to the Remote Host

To transfer files from your workstation to a remote host, mark the files that you want to transfer, then select the Put Local Files option from the Actions pull-down menu.

All marked files in the local files list are sent to the current remote host. If you have marked a directory from the directory list, all files in that directory are also sent to the current working directory of the remote host. (A new remote subdirectory is not created.) Files in subdirectories of the marked directory are not transferred.

See "Transferring Files Using QuickTrans" in topic 3.8.13 for an alternative method.

3.8.2 Transferring Files from the Remote Host to Your Workstation

To transfer files from a remote host to your workstation, mark the files that you want to transfer, then select the Get Remote Files option from the Actions pull-down menu.

All marked files in the remote files and directory list are sent from the current remote host to your workstation.

If you have marked a directory from the files and directories listing, all files in that directory are also sent to the working directory on your workstation. (A new local subdirectory is not created.) Files in subdirectories of the marked directory are not transferred.

See "Transferring Files Using QuickTrans" in topic 3.8.13 for an alternative method.

3.8.3 Transferring Files from the Current Remote Host to Another Remote Host

If you are connected to two or more remote hosts, you can transfer files between them. To transfer files from the current remote host to another remote host, mark the files that you want to transfer, then select the Send Files Between Remote Hosts option from the Actions pull-down menu. The Current Hosts window displays a list of all the remote hosts to which you are currently connected, as shown in Figure 5.

PICTURE 5

Figure 5. Example of the Current Hosts Window

Use the arrow keys or the mouse to select the remote host to which you want to transfer the marked files.

Use the mouse or the tab keys to move to the appropriate pushbutton at the bottom of the window.

Pushbutton	**Description**
OK	The remote host you selected becomes the destination host for the marked files, and the transfers begin.
Cancel	Cancels the transfers and returns to FTPPM window.

All marked files in the remote files and directory list are sent to the target remote host. If you have marked a directory in the files and directory list, all files in that directory are also sent to the working directory on your workstation. (A new local subdirectory is not created.) Files in subdirectories of the marked directory are not transferred.

3.8.4 Deleting Files and Directories on Your Workstation

To delete files or directories on your workstation, mark the files or directories that you want to delete, then select the Delete Local Files option from the Actions pull-down menu.

All marked files in the local files list are deleted. If you have enabled interactive prompting, you are asked to confirm the deletion of each file.

If you have marked a directory, the directory is removed only if it is empty. If files remain in the directory, an error message results.

3.8.5 Deleting Files and Directories on the Remote Host

To delete files or directories on a remote host, mark the files or directories that you want to delete, then select the Delete Remote Files option from the Actions pull-down menu.

To delete files or directories on your workstation, mark the files or directories that you want to delete, then select the Delete Local Files option from the Actions pull-down menu.

All marked files in the remote file and directory list are deleted. If you have enabled interactive prompting, you are asked to confirm the deletion of each file.

If you have marked a directory, the directory is removed only if it is empty. If you do not have the required access to that directory, or if files remain in the directory, an error message is issued.

3.8.6 Creating a New Directory on Your Workstation

To create a directory on your workstation, select the Make Local Directory option from the Actions pull-down menu. The New Local Directory window (Figure 6) is displayed. Enter the new directory name.

PICTURE 6

Figure 6. Example of the New Local Directory Window

Use the mouse or the Tab key to move to the appropriate pushbutton at the bottom of the window.

Pushbutton	**Description**
OK	Proceeds to create the directory
Cancel	Cancels the creation of a new directory on your workstation and redisplays the FTPPM window

3.8.7 Renaming Files on Your Workstation

To rename a file on your workstation, select the Rename Local File option from the Actions pull-down menu. Enter the old name and the new name of the file in the entry fields, as shown in Figure 7.

PICTURE 7

Figure 7. Example of the Rename Local File Window

Use the mouse or the Tab key to move to the appropriate pushbutton at the bottom of the window.

Pushbutton	**Description**
OK	Attempts to rename the file
Cancel	Cancels the operation

3.8.8 Creating a New Directory on the Remote Host

To create a directory on a remote host, select the Make Remote Directory option from the Actions pull-down menu. The New Remote Directory window is displayed. Enter the new directory name, as shown in Figure 8.

PICTURE 8

Figure 8. Example of the New Remote Directory Window

Use the mouse or the Tab key to move to the appropriate pushbutton at the bottom of the window.

Pushbutton	**Description**
OK	Attempts to add the foreign directory. If you do not have the correct authorization, an error message is issued.
Cancel	Cancels the creation of a new directory on the remote host and redisplays the FTPPM window.

3.8.9 Renaming Files on the Remote Host

To rename a file on a remote host, select the Rename Remote File option from the Actions pull-down menu. Enter the old name and the new name of the file in the entry fields, as shown in Figure 9.

PICTURE 9

Figure 9. Example of the Rename Remote File Window

Use the mouse or the Tab key to move to the appropriate pushbutton at the bottom of the window.

Pushbutton	**Description**
OK	Attempts to rename the file
Cancel	Cancels the operation

3.8.10 Sending a Site Command to the Remote Host

Select the Send Site Command to Remote Host option from the Actions pull-down menu.

This command sends information to the foreign host that is used to provide services that are specific to the foreign host system.

3.8.11 Sending a Quote Command to the Remote Host

Select the Send Quote command to the Remote Host option from the Actions pull-down menu.

This command sends the specified argument(s) to the FTP server on the foreign host verbatim.

Note: Using this subcommand with commands that involve data transfers can produce unpredictable results.

3.8.12 Updating Local and Remote Host Files and Directories Lists

When you select the Update option, the local or remote host's files and directories list is redisplayed on the window. You can see any changes that were made to either the local or the remote host's directories since the last update. The Update option unmarks all marked files or directories.

File and directory lists are also automatically updated after any transfer or deletion action occurs.

3.8.13 Transferring Files Using QuickTrans

When you select the QuickTrans option, all marked local files are transferred to the current remote host and all marked remote files are transferred to your workstation.

3.9 Ending the FTPPM Session

You can end an FTPPM session and disconnect from all remote hosts in one of the following ways:

- Select the Close all Connections option from the Remote pull-down menu. You will be prompted to confirm that you want to end all connections.
- Press the F3 key to exit. You will not be prompted to confirm that you want to exit the session.
- Close the FTPPM window. You will not be prompted to confirm that you want to exit the session.

Note: Options set for the current session are not saved for future sessions.

4.0 Chapter 4. Transferring Files with the Trivial File Transfer Protocol

This chapter describes the Trivial File Transfer Protocol (TFTP) and the **tftp** command. TFTP is an alternative to FTP, which is described in Chapter 2, "Transferring Files with the File Transfer Protocol (FTP)" in topic 2.0.

This chapter contains:

- An overview of the functions that TFTP provides
- Examples of how to enter the TFTP command shell
- An explanation of the **tftp** subcommands
- An explanation of how to request help and status information
- An explanation of how to set the environment for transferring files
- An explanation of how to send and receive files

Note: For a concise description of commands, subcommands, and parameters, see the *IBM TCP/IP Version 2.0 for OS/2: Command Reference*.

Subtopics

4.1 Deciding Which Command to Use

TFTP is a simple file transfer protocol that does not provide all of the features available in FTP. It uses the User Datagram Protocol (UDP) as the underlying protocol; therefore, it is an unreliable means of file transfer.

TFTP has the following additional limitations:

- It does not provide subcommands to list files or change directories on the foreign host.
- It does not provide user authentication.
- It allows the transfer of only a single file at a time.
- The TFTPD server can service only one client at a time, as opposed to FTPD, which can support many clients at once.

Despite the limitations of TFTP, certain applications may require its use. For example, TFTP may be the only file transfer method supported on the destination foreign host.

4.2 Overview of the Trivial File Transfer Command

The **tftp** command transfers files between your workstation and a foreign host.

Specifically, the **tftp** command allows you to:

- Request help for the **tftp** subcommands
- Obtain status information about the TFTP connection
- Transfer files in either binary or ASCII mode between your workstation and a foreign host

4.3 Setting Up the Server

TCP/IP for OS/2 is implemented with both client and server support for TFTP.

To start the server on your local host, type **tftpd** at an OS/2 command prompt and press the Enter key. The **tftpd** command starts the TFTPD.EXE program and runs as a task until you shut down the server.

As an alternative, you can start TFTPD using INETD. INETD allows you to start multiple servers from a single OS/2 session.

If you use INETD to autostart TFTPD, you cannot specify any parameters for this service. However, you can add the following environment variable to your CONFIG.SYS file to specify the default path:

```
SET TFTPDPATH=path
```

For more information about INETD, see the *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration* manual.

Only one TFTPD server can run on a workstation at one time. If you try to start a second TFTPD, the following message is displayed:

Warning: When the TFTPD server is running on your workstation, other users in the network can read, write, or even destroy the files on your workstation.

Subtopics
4.3.1 Restricting Access to Files

4.3.1 Restricting Access to Files

Use the **tftpd** command to restrict access to a particular path. The syntax of this command is:

```
>>--tftpd-------------><
          +- path-+
```

path The path for which you are granting access to the TFTP client. The path is used as a prefix for all file names specified by the **put** and **get** subcommands of **tftp**.

The following example shows the **tftpd** command restricting access for TFTP clients to the \TEMP directory on the C drive.

```
tftpd c:\temp\
```

The value of the parameter is used as a prefix for all file names specified by the **put** and **get** subcommands. If you request a **get** or **put** operation, specifying a file name of xxx.aaa, the resulting **get** or **put** is for the file xxx.aaa in the \TEMP directory on the C drive.

If you specify the **tftpd** command as shown in the following example and request a **get** or **put** operation for file xxx.aaa, the resulting **get** or **put** is for file \TEMP\xxx.aaa on the root directory of the C drive.

```
tftpd c:\temp
```

If you start the TFTPD server with the **tftpd path** parameter, all TFTP clients are restricted to the specified path. They do not have access to files on other paths and cannot place files on other paths.

4.4 Entering the TFTP Command Shell

You can enter **tftp** subcommands from the TFTP command shell. Enter the TFTP command shell from the OS/2 prompt by invoking the **tftp** command with one of the two following choices:

- With no host specified
- With a host specified

The syntax of the **tftp** subcommand is:

```
>>--tftp------------------------><
          +- host------------+
                   +- port-+
```

host The host to which you are connecting. The **tftp** command immediately tries to establish a connection to this host.

port The destination port to which you are connecting. If **port** is not specified, you are connected by default to the well-known TFTP port as specified in the services file.

After you start a session with the TFTP command shell, you can enter **tftp** subcommands. The following example enters the TFTP shell without specifying a host:

```
tftp
```

The following example opens a TFTP connection between your workstation and the host 129.33.254.5:

```
tftp 129.33.254.5
```

After the TFTP command shell is active, the TFTP prompt is displayed:

```
tftp>
```

4.5 tftp Subcommands

Enter the **tftp** subcommands at the TFTP prompt. The **tftp** subcommands are:

Subcommand	Function
?	Displays help information
ascii	Sets the file transfer mode to ASCII
binary	Sets the file transfer mode to binary (image)
connect	Establishes a connection to a specified foreign host
get	Transfers a specified file from a foreign host to your workstation
mode	Sets or displays the file transfer mode
put	Transfers a specified file from your workstation to a foreign host
quit	Exits the TFTP command shell
rexmt	Sets the retransmission time-out for each packet
status	Displays the current status of the TFTP connection
timeout	Sets the total retransmission time-out
trace	Toggles packet tracing
verbose	Toggles verbose mode

You must enter the subcommands in lowercase.

4.6 Requesting Help

To display help while you are in the TFTP command shell, use the **?** subcommand by itself to display a list of the **tftp** subcommands; or use the **?** subcommand with a subcommand to request help for that specific **tftp** subcommand. You must enter the **tftp** subcommand for which you are requesting help in lowercase. The syntax of this subcommand is:

```
>>--?-------------------><
       +- subcommand-+
```

subcommand The subcommand for which you are requesting help. You must enter the subcommand in lowercase.

4.7 Establishing a Connection

The **connect** subcommand with the **host** parameter establishes a predefined connection to a foreign host. The syntax of this subcommand is:

```
>>--connect------------------------><
             +- host------------+
                      +- port-+
```

host The remote host to which you want to connect.

port The destination port to which you are connecting. If **port** is not specified, you are connected by default to the well-known TFTP port as specified in the services file.

Use the **connect** subcommand in the following situations:

- If you did not specify a host when you entered the TFTP command shell.
- If you were unable to open a connection when you specified a host with the **tftp** command. This situation can occur if you did not enter the host correctly with the **tftp** command, or if you experienced network problems when you issued the **tftp** command.
- If you want to change from the current host connection to a connection with a different host.

If you do not specify a host, TFTP prompts you for it.

4.8 Obtaining Status Information

Use the **status** subcommand to display the following status information:

- Connection status
- Transfer mode
- Verbose flag setting
- Tracing flag setting
- Retransmission time-out value for one packet
- Retransmission time-out value for the file transfer

The syntax of the **status** subcommand is:

>>--**status**--><

The following is a sample response from the **status** subcommand.

```
connected to 1.1.1.2
Mode: octet Verbose: off Tracing: off
Rexmt-internal: 5 seconds, Max-timeout; 25 seconds
```

4.9 Setting the Environment

Before transferring files with TFTP, you can change the type of transfer and other aspects of the file transfer environment. The **tftp** subcommands that set the environment are:

- **ascii**
- **binary**
- **mode**
- **rexmt**
- **timeout**
- **trace**
- **verbose**

Remember to enter the subcommands in lowercase.

Subtopics

4.9.1 Setting the File Transfer Mode

Use the **mode** subcommand to set the file transfer mode. The syntax of this subcommand is:

```
>>--mode---------------><
          +- ascii--¦
          +- binary-+
```

The setting that you specify will remain in effect until you either change it or quit TFTP. If you specify **mode** without a parameter, TFTP will display a message indicating the current transfer type.

ascii — Specifies the file transfer mode as ASCII. This is the default file transfer mode and is used to transfer files that contain text characters. Specifying the ascii parameter with the **mode** subcommand is the same as issuing the **ascii** subcommand.

Note: The term netascii is synonymous with ascii.

binary — Sets the file transfer mode as binary (image). Specifying the binary parameter with the **mode** subcommand is the same as issuing the **binary** subcommand.

Note: The terms image and octet are synonymous with binary.

Set the file transfer mode to match the data representations supported by the host to which you are transferring a file. TFTP supports the following two file transfer modes:

- ASCII
- Binary (image)

The following example shows **mode** specified with the binary parameter:

```
mode binary
```

4.9.2 Changing the File Transfer Mode to ASCII

Use the **ascii** subcommand to change the file transfer mode to ASCII. The syntax of this subcommand is:

>>--**ascii**--><
ASCII is the default file transfer mode and is used to transfer files that contain text characters. Using the **ascii** subcommand is the same as issuing the **mode** subcommand with the ascii parameter.

ASCII files are often terminated by a special character (Ctrl-Z) to signify the end of the file. The special character is not part of the file and, in ASCII transfer mode, is not sent to the remote host. The remote host may or may not restore this end-of-file character, depending on its own operating system and also on the implementation of its TFTP server.

For example, the OS/2 TFTP server appends an end-of-file character to all incoming ASCII files.

4.9.3 Changing the File Transfer Mode to Binary

Use the **binary** subcommand to change the file transfer mode to binary. The syntax of this subcommand is:

```
>>--binary--><
```

The binary file transfer mode is useful for image transfers, such as transferring executable files. Using the **binary** subcommand is the same as issuing the **mode** subcommand with the binary parameter.

4.9.4 Toggling the Verbose Mode

Use the **verbose** subcommand to toggle the flag that determines whether more information regarding the transferred files should be displayed. The syntax of this subcommand is:

>>--**verbose**--><

If the verbose flag is on, a message announces that the transfer has begun and gives the file name and its destination. At the end of the file transfer, the file transfer rate is displayed. If the verbose flag is off, information is not displayed until the file transfer is complete. The following example illustrates the difference between the two flag settings.

```
tftp> verbose
Verbose mode off.
tftp> put hosts /tmp/foo
Sent 1633 bytes in 0.3 seconds
tftp> verbose
Verbose mode on.
tftp> put hosts /tmp/foo
putting hosts to rios.tcpipdev.raleigh.ibm.com:/tmp/foo [netascii]
Sent 1633 bytes in 0.3 seconds [52256 bits//sec.]
```

The default mode for verbose is off.

4.9.5 Setting the Packet Retransmission Time-Out

Use the **rexmt** subcommand to set the packet retransmission time-out. The syntax of this subcommand is:

```
>>--rexmt----------><
           +- n-+
```

n The maximum number of seconds that TFTP tries to retransmit one packet. If you do not specify this value, TFTP will prompt you for it.

The only parameter of the **rexmt** subcommand is **n**, which is the maximum number of seconds that TFTP tries to retransmit one packet.

The **rexmt** subcommand retransmits each packet for the number of seconds specified by the **n** parameter. If you do not specify the number of seconds, the **rexmt** subcommand prompts you for the number.

If a packet is not successfully transmitted in **n** seconds, the file transfer terminates. The default value for the packet retransmission timeout is 5 seconds. The acceptable range is from 0 to 32 767 seconds.

4.9.6 Setting the Total Retransmission Time-Out

Use the **timeout** subcommand to set the total retransmission time-out. The syntax of this subcommand is:

```
>>--timeout----------><
             +- n-+
```

n Specifies the maximum number of seconds that TFTP tries to retransmit the file. If you do not specify this value, TFTP will prompt you for it.

The **timeout** subcommand retransmits the file for the number of seconds for which it is set with the **n** parameter. If you do not specify the number of seconds, the **timeout** subcommand prompts you for the number. If a file is not successfully transmitted in **n** seconds, the file transfer terminates. The default value for the total retransmission timeout is 25 seconds. The acceptable range is from 0 to 32 767 seconds.

4.9.7 Toggling the Packet Tracing

Use the **trace** subcommand to toggle the flag that determines whether transmitted packets are traced. The syntax of this subcommand is:

>>--**trace**--><
The default mode for **trace** is off.

If the packet tracing flag is on, packets are traced during the file transfer. The following is a sample of the response that is displayed when packet tracing is on.

```
Sent WRQ
Received ACK <block=0>
Sent DATA <block=1, 102 bytes>
Received ACK <block 1>
Sent 102 bytes in 0.8 seconds [1165 bits/second]
```

If the packet tracing flag is off, packets are not traced and packet tracing messages are not displayed. The following is a sample of the response that is displayed when packet tracing is off:

```
Sent 102 bytes in 0.8 seconds [1165 bits/second]
```

4.10 Transferring Files

When you transfer files to or from a foreign host, the TFTP server (TFTPD) must be running on the foreign host. To set up the OS/2 TFTP server, see "Setting Up the Server" in topic 4.3.

The **tftp** command has two subcommands that transfer files between your workstation and a foreign host. The **get** subcommand transfers a file from a foreign host to your workstation. The **put** subcommand transfers a file from your workstation to a foreign host.

Subtopics
4.10.1 Receiving a File from a Foreign Host
4.10.2 Sending a File to a Foreign Host

4.10.1 Receiving a File from a Foreign Host

Use the **get** subcommand to receive a file from a foreign host. The syntax of this subcommand is:

```
>>--get-------------------------------------><
         +- remote_file------------------+
                         +- local_file-+
```

remote_file The name of the file, on the remote host, that is transferred to your workstation.

local_file The name of the file created on your workstation in which to store a copy of **remote_file**.

The **get** subcommand copies the specified file to your workstation. If you do not specify the remote file or the local file, TFTP prompts you for the remote file name. If you specify only the remote file, TFTP transfers the remote file to your workstation, assigning the remote file name to the new file.

The following example transfers the file named **hisfile.cmd** on the foreign host to your workstation, where the file is stored as **myfile.cmd.**

```
get hisfile.cmd myfile.cmd
```

4.10.2 Sending a File to a Foreign Host

Use the **put** subcommand to send a file from your workstation to a foreign host. The syntax of this subcommand is:

```
>>--put-------------------------------------><
         +- local_file-------------------+
                        +- remote_file-+
```

local_file The name of the file on your workstation that is transferred to the remote host.

remote_file The name of the **local_file** when the file is stored on the remote host.

The **put** subcommand copies the specified file to the foreign host. If you do not specify the local file or the remote file, TFTP prompts you for the local file name. If you specify only the local file, TFTP transfers the local file to the foreign host, assigning the local file name to the new file.

Because TFTP does not demand a password from the user, many foreign hosts restrict the files that can be transferred. For example, foreign hosts can impose a restriction against duplicate file names to avoid overwriting existing files. An attempt to transfer a file with the same name as an existing file can fail because of this type of restriction.

The following example transfers the file named **myfile.cmd** from your workstation to the foreign host, where the file is saved as **herfile.cmd.**

```
put myfile.cmd herfile.cmd
```

4.11 Ending the TFTP Session

Use the **quit** subcommand to end the file transfer session, disconnect any connected **host**, and leave the TFTP command shell. The syntax of this subcommand is:

4.12 Ending the Server Task

Follow these steps to end the TFTPD server task:

1. Activate the window in which the server is running by clicking in it.
2. Press the Ctrl and C keys at the same time.

The TFTPD.EXE server task is ended.

5.0 Chapter 5. Using Mail Services

This chapter describes the Simple Mail Transfer Protocol (SMTP) mail services in TCP/IP for OS/2. This chapter also describes: how to set up the SendMail server, how to use SendMail, and how to use LaMail. The commands and considerations associated with SendMail are also described.

Subtopics

5.1 Overview of the Mail Services

The SMTP mail services enable you to send and receive mail. In TCP/IP for OS/2, the SendMail program implements SMTP. SendMail uses SMTP to route mail from one host to another host, allowing you to exchange mail with other hosts that support SMTP.

There are several electronic mail handling systems used with the SendMail mail delivery system including the following IBM Presentation Manager applications:

- LaMail. You can receive, read, create and send electronic mail.

- Ultimedia Mail/2. You can receive, read, create, and send electronic mail that contains a variety of media including text, images, audio, video clips, and binary attachments such as programs, spreadsheets, and word processor documents.

  Note: Please contact your IBM marketing representative for more information about this product.

LaMail is an OS/2 Presentation Manager interface for the SMTP mail service and allows you to create, modify, and manage your mail. SMTP for TCP/IP for OS/2 mail services is implemented with the LaMail user interface and the SendMail server for mail functions.

SendMail functions as a mail router, listening for and receiving mail from the LAN (server mode) and sending mail on the LAN (client mode). For more information, see "SendMail Startup and Processing" in topic 5.3.

When SendMail receives mail from the LAN, it invokes a user mail agent which stores the mail in the specified directory. (The default installation directory is *TCPIP*\MAIL\, where *TCPIP* is the name of the base directory in which TCP/IP for OS/2 is installed on your host.) See "Setting Up the SendMail Environment" in topic 5.2 for more information about how to configure SendMail for processing incoming mail.)

You can use a mail agent to read your mail. The mail agent described here is LaMail. LaMail does not have to be running to receive mail. When you use LaMail to create and send mail, LaMail automatically passes the mail to SendMail for delivery over the LAN.

You can use LaMail to work on your mail from a remote location, even when you are away from your office and do not have a TCP/IP connection. You can copy your notes to a diskette and work with the files at a remote workstation.

When working remotely, you have full access to all of the LaMail functions. You can temporarily store your notes on a diskette and send them when you return to your office workstation. For more information about LaMail remote services, see "Remote LaMail Services" in topic 5.12.11 .

Subtopics
5.1.1 Using MX Records
5.1.2 Mail Addresses

5.1.1 Using MX Records

MX (mail exchanger) records direct the SMTP server to deliver mail to alternative hosts. MX records are obtained from the Domain Name Server. If SMTP is not using a name server, then MX records are not used.

For example, if SMTP wants to send mail to USER@HOST, it checks the name server for MX records and finds the following:

```
HOST    MX    0    HOST
HOST    MX    5    HOST-BACKUP1
HOST    MX    10   HOST-BACKUP2
```

SMTP delivers the mail to the record (host) that has the lowest count--in this example, directly to **HOST**. If **HOST** is unable to receive the mail, SMTP then tries to deliver it to **HOST-BACKUP1**. If **HOST-BACKUP1** cannot receive the mail, it tries **HOST-BACKUP2**. If none of the hosts can receive the mail, SMTP stores the mail and queues it for later delivery, at which time the process repeats.

If SMTP does not find MX records for a host, it delivers mail only to the primary host.

For more information about MX records, see RFC 974.

5.1.2 Mail Addresses

SendMail accepts mail addressed to any user on an OS/2 machine, regardless of the value of the **username** field, because the OS/2 operating system is a single-user operating system.

Parameter	**Description**
username	The user account
host_name	The host machine

Subtopics
5.1.2.1 Examples

5.1.2.1 Examples

1. A valid address format to send mail to the OS/2 machine with the host name **myps2** can be any one of the following:

   - user@myps2
   - xxx@myps2
   - **any-valid-username**@myps2

   Any **username** is allowed if it meets the requirements of the SMTP protocol.

   The value of the **username** field is not important, but it is still required. Without a **username**, mail addressed to **myps2** is not delivered correctly and is usually sent back to the originator.

2. The address for the user account **daniel** on the host **vm1** is:

   daniel@vm1

   This address requires the user account because mail is sent directly to a user account on a machine, not to the machine itself.

5.2 Setting Up the SendMail Environment

The following sections describe how to set up the SendMail environment.

Subtopics

5.2.1 Creating the Required Subdirectories

The SENDMAIL.CF file must reside in the ETC subdirectory or the directory specified by the ETC environment variable. The following subdirectories are defined in the SENDMAIL.CF file:

Subdirectory	**Description**
Mail	Incoming mail is stored in this directory. (*TCPIP*\MAIL\ is the default installation directory, where *TCPIP* is the name of the base directory in which TCP/IP for OS/2 is installed on your host.)
MQUEUE	Outgoing mail and temporary files are stored in this directory.

Ensure that the mail and MQUEUE subdirectories have been created.

5.2.2 Manually Configuring SendMail for LaMail

To configure SendMail, you must edit the SENDMAIL.CF file, which is usually installed in your TCPIP\ETC subdirectory. This is the subdirectory specified in the ETC= environment variable. To locate this file, look for the SET ETC= entry in your CONFIG.SYS file or type:

```
echo %ETC%
```

at any OS/2 command prompt.

Warning!

When you modify SENDMAIL.CF, use a text editor that preserves tab characters, such as the OS/2 System Editor (E.EXE). Other text editors might expand the tab characters into spaces and make the file unusable. It's a good idea to make a backup copy of SENDMAIL.CF before you change it.

You must make the following changes to the SENDMAIL.CF file:

- Change the DW and CW parameters to reflect your host name.
- Change the DD parameter to reflect your domain name. If you do not have a domain name, delete the line containing the DD parameter.

For more information, see "SENDMAIL.CF File Entries" in topic 5.2.2.1.

Subtopics
5.2.2.1 SENDMAIL.CF File Entries

5.2.2.1 SENDMAIL.CF File Entries

The following SENDMAIL.CF file entries determine how the SendMail program runs. Do **not** change any entries other than those recommended unless you are certain about the effect of your changes. If your SENDMAIL.CF file is incorrect, you might not be able to send or receive mail.

Cw Specifies your host (workstation) name and any aliases you may want to designate. You can use the TCP/IP **hostname** command to display your host name.

Dw Specifies your host (workstation) name only. This should be the same as the **Cw** entry. (You cannot specify alias names.)

DD Specifies your local domain name.

For instance, if your workstation is **earth.raleigh.ibm.com** and the path to your mail queue is **D:\TCPIP\MQUEUE**, you would complete these entries as follows:

```
OQD:\TCPIP\MQUEUE
Cwearth
Dwearth
DDraleigh.ibm.com
```

Mlocal Specifies aspects of the local mail handler.

A sample Mlocal entry is shown below. It assumes that:

- LaMail is installed into **D:\TCPIP\MAIL**
- Incoming mail is stored in **D:TCP\MAIL\FOLDERS\INBOX**

You would type it as one line in the SENDMAIL.CF file:

```
Mlocal, P=D:\TCPIP\MAIL\LAMAILER.EXE , F=lsm,
A=-dest D:\TCPIP\MAIL\FOLDERS\INBOX -to $u
```

These fields are used in the **Mlocal** entry to configure SendMail:

P= Specifies a mail handler to be activated. Set this field to the path and name of the LaMail mail handler (LAMAILER.EXE).

Note: Leave a blank space between the mail handler name and the comma following it.

F= Specifies (at a minimum) the following fields:

l Specifies a local mail handler for final delivery.

s Strips quotes from user IDs.

m Specifies that the mailer can handle multiple users on the same host in one transaction. If **m** is not specified, LAMAILER.EXE is called once for each user.

A= Specifies the argument to be passed to the mail handler.

For LAMAILER.EXE, it must be specified as follows:

```
A=-dest dir -to users
```

where:

dir Specifies the directory where the mail should be saved (the InBox of the LaMail server).

users Specifies a SendMail macro containing the users to whom this mail is to be delivered. This value should be set to $u.

Note: If **m** is specified in the **F=** field, the $u macro contains a list of all users to which this mail is addressed.

OT Designates how long mail is held in the mail queue until it can be delivered to a server that is down. Undelivered mail is held while SENDMAIL attempts to retransmit it.

The format of the Options Timeout (OT) entry is:

OT**xt**

Where **xt** is the time interval that undelivered mail is retained for retransmission before it's deleted from the mail queue:

- **x** is an integer
- **t** is the unit of time (**d** for days, **m** for minutes, **h** for hours, or **w** for weeks)

For example, if you add the following entry to your SENDMAIL.CF file:

OT**8h**

and your note could not be delivered because a server is down, SENDMAIL holds the note in the mail queue for eight hours (while attempting to resend it) before deleting it from the queue.

OQ Specifies the path to the mail queue. This is where SendMail temporarily holds mail sent by LaMail before delivering it to the recipient. Although you can specify any valid path, it is recommended that you create a subdirectory (such as **\TCPIP\MQUEUE**) specifically for this purpose.

OX Sets the Load Limiting variable. The format of this entry is:

OX**n**

Where **n** is an integer specifying the maximum number of concurrent SendMail instances permitted. (The default is six.) See "SendMail Startup and Processing" in topic 5.3 for more information.

5.3 SendMail Startup and Processing

You can use SendMail as a server (to receive mail) or as a Client (to send mail). If you use both modes, the server must always be started first. For more information, see:

- "Starting the SendMail Server" in topic 5.4
- "Starting the SendMail Client" in topic 5.5

When you issue the **sendmail** command:

1. A SendMail server is started to set up the requested SendMail environment, the SENDMAIL.CF file is copied to a shared memory location, and the server monitors client requests to establish connections and deliver mail.

2. The SendMail server initiates another instance (a SendMail child process) when a client requests a connection.

   Note: The number of concurrent SendMail instances is controlled by a Load Limiting variable (the default is six). You can set the OX option in the SENDMAIL.CF file to change the Load Limiting variable. For example, if you wanted to refuse client connections after three instances have started, you would modify your SENDMAIL.CF file as follows:

   ```
   OX3
   ```

   Once a SendMail instance is initialized (using the SENDMAIL.CF file information located in shared memory), the connection is established and mail is received.

3. SendMail instance(s) close the connection and exit once the mail has been processed and delivered.

4. Error messages received during SendMail transactions are recorded in the *TCPIP*\ETC\SENDMAIL.ERR file (where *TCPIP* is the name of the base directory in which TCP/IP for OS/2 is installed on your host). This file can provide valuable debugging information. See "SENDMAIL--SENDMAIL.ERR Errors" in topic D.12 for information about messages you can receive.

If you stop the SendMail program while it is sending or receiving mail, files can be stranded in the MQUEUE subdirectory. It is recommended that you periodically check the MQUEUE subdirectory and delete the old files.

5.4 Starting the SendMail Server

You can start the SendMail server by:

- Using the TCP/IP Configuration notebook program (see "Automatically Starting the Server" in topic 5.4.1)

- Issuing the **sendmail** command at an OS/2 system prompt (see "Manually Starting the Server" in topic 5.4.2)

Subtopics
5.4.1 Automatically Starting the Server
5.4.2 Manually Starting the Server

5.4.1 Automatically Starting the Server

You can use the Configuration Notebook Program (TCPIPCFG) to automatically start SendMail each time TCP/IP is started. Figure 10 shows the Configuration Notebook Program page setting to:

- Enable SendMail (minimized session) using the default parameter settings
- Automatically start a LaMail session (minimized)

PICTURE 10

Figure 10. Example of Configuration Notebook Program to Automatically Start SendMail

5.4.2 Manually Starting the Server

You can start SENDMAIL by issuing the **sendmail** command. The syntax of this command is:

Subtopics
5.4.2.1 Starting the SendMail Server
5.4.2.2 Debugging Problems

¦*5.4.2.1 Starting the SendMail Server*

¦>>--**sendmail**-- **-bd**-- **-qtime**--><

5.4.2.2 Debugging Problems

```
>>--sendmail-- -bd-- -qtime--------------><
                           +- -d----+
                           +- -dn.n-+
```

-bd Starts SENDMAIL as a server.

-q**time** Specifies how often the mail queue can be processed. Enter the time as a number and a letter, where the letter is one of the following:

- s for seconds
- m for minutes
- h for hours
- d for days
- w for weeks

For example:

-q30m specifies every 30 minutes

-q1h30m specifies every hour and 30 minutes

-d Writes detailed debugging information to the SENDMAIL console and creates a SENDMAIL.LOG file that contains the simple mail transfer protocol (SMTP) transactions between the SENDMAIL server and the remote SMTP server.

The SENDMAIL.LOG file is placed in the ETC directory. For example, to start the SENDMAIL server with detailed debug information, enter:

```
sendmail -bd -q30m -d
```

Warning!

You should use this parameter for debugging purposes only.

-d**n.n** Specifies the debugging activity supported. For example, if you specify d**1.1**, a SENDMAIL.LOG file is created in the ETC subdirectory. This file contains information about client and server interaction (per RFC 821).

See your system administrator for more information about the debugging activities supported.

Warning!

You should use this parameter for debugging purposes only.

After the SendMail program starts, a status message is displayed confirming that the program started correctly. If you press the Ctrl and Esc keys at the same time, SENDMAIL.EXE is displayed in the Task List window.

5.5 Starting the SendMail Client

You can send mail using SendMail as a foreground process without using LaMail. Follow these steps to manually send mail:

1. Create a piece of mail using any text editor.
2. File the message in one of your directories.
3. Issue the **sendmail** command.

¦**Note:** If you use both the SendMail server (to receive mail) and the ¦SendMail client (to send mail), you must always start the server first. ¦For more information about starting a SendMail server, see "Setting Up the ¦SendMail Environment" in topic 5.2.

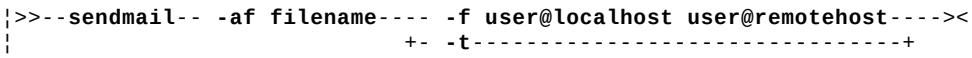
If you attempt to send mail to a host that is not up and running, SendMail:

- Stores (queues) the mail in the MQUEUE subdirectory
- Resends the mail, after the time interval specified by the -q parameter, until the mail is either:
  - Successfully delivered or
  - Deleted when the Options Time (OT) interval expires (see "SENDMAIL.CF File Entries" in topic 5.2.2.1)

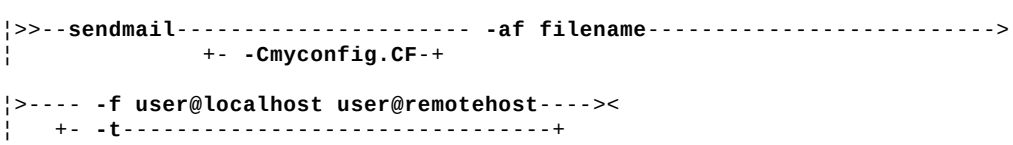
The syntax of the **sendmail** command is:

Subtopics
5.5.1 Sending Mail
5.5.2 Using An Alternate Configuration File
5.5.3 Processing the Mail Queue
5.5.4 Testing Configuration File Rules Changes
5.5.5 Debugging Problems

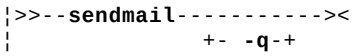
5.5.1 Sending Mail



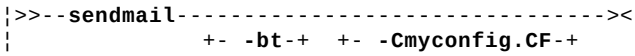
5.5.2 Using An Alternate Configuration File



5.5.3 Processing the Mail Queue



5.5.4 Testing Configuration File Rules Changes



5.5.5 Debugging Problems

```
>>--sendmail---------------------- -af filename-------------------------->
             +- -Cmyconfig.CF-+

>---- -f user@localhost user@remotehost---------------><
   +- -t--------------------------------+  +- -dn.n-+
                                         +- -d----+
```

-q
: Instructs SendMail to process messages in the mail queue.

-bt
: Instructs SendMail to run in rule-testing mode. You can use this parameter to test changes to rules in the configuration file. In rule-testing mode, SendMail prompts you to enter a rule-set number and address, and then displays how the rule-set causes the address to be rewritten. OS/2 SendMail always calls rule-set 3 first and then the entered rule-set number. In rule-testing mode, SendMail does not deliver any mail.

  Note: For more information about rule-testing mode, contact your system administrator.

-C**myconfig**.CF
: Specifies the name of a configuration file to use instead of SENDMAIL.CF

-af **filename**
: Specifies the name of the file that contains the mail message.

-f **user@localhost**
: Identifies the user and host name of the sender.

user@remotehost
: Identifies the user and host name of the receiver (or destination).

  Note: OS/2 is not a multiuser operating system. The **user** field of an address in an item of mail sent to an OS/2 system is not significant; however, it is required. The significant part of the address is the host name, with the domain name expansion if domain names are used.

-t
: Specifies that SENDMAIL retrieves the addressing and subject information from the header area (top) of the message file rather than from the command line. SendMail scans the note for the following tags:

  ```
  To:
  From:
  BCC: (blind carbon copy)
  CC: (carbon copy)
  Subject:
  ```

  Note: Addresses must comply with RFC 822 conventions.

-d**n.n**
: Specifies the debugging activity supported. For example, if you specify d**1.1**, a SENDMAIL.LOG file in the ETC subdirectory is created.

  See your system administration for information about the debugging activities supported.

  Warning!

  The SENDMAIL.LOG is not intended to provide a record of mail activity. This parameter should be used for debugging purposes only, because it records all client-server interaction (as described in RFC 821) in the SENDMAIL.LOG.

-d
: Writes detailed debugging information to the SENDMAIL console and creates a SENDMAIL.LOG file.

  Warning!

  The -d parameter produces a great deal of information, and for performance reasons, should be used only for debugging.

Notes:

1. The SendMail client uses available shared memory configuration information (see "Setting Up the SendMail Environment" in topic 5.2 for details.) If shared information is not available, the SENDMAIL.CF file is used.

2. The significant part of the address is the host name (with the domain name expansion, if domain names are used). For more information about addresses, see "Mail Addresses" in topic 5.1.2.

3. Error messages received during SendMail transactions are recorded in the *TCPIP*\ETC\SENDMAIL.ERR file (where *TCPIP* is the name of the base directory in which TCP/IP for OS/2 is installed on your host). This file can provide valuable debugging information. See "SENDMAIL--SENDMAIL.ERR Errors" in topic D.12 for information about messages you can receive.

5.6 Transmitting Messages to Multiple Users

When you are transmitting messages to multiple users:

1. It is recommended that you limit distribution to 150 recipients on the same workstation.

   Note: When you issue a message to 150 recipients or more, multiple files are generated. The number of recipients contained in a file is limited by the header buffer space. As a result, several files can be created for a single transaction when:

   - Recipients are added to a file until the header area is full.
   - Remaining recipients are processed in a new file under a new header.

   The transaction time for transmitting a message to a single user is significantly less than that required for sending a message to multiple users. If a busy server cannot service a given connection fast enough, the client that established the connection can time out. Because the connection is not closed properly when a client times out, SendMail is not aware of transmission status and requeues the message for later delivery when it was, in fact, already transmitted.

   To avoid unnecessary looping, it is imperative to use a server with sufficient processing capability to handle your mail delivery load. At a minimum, we recommend a 33 megahertz machine with 16 MB RAM, HPFS, and DISKCACHE to enhance disk performance.

   Note: The time-out period is determined by the idle time on the socket. It is typically measured from the last activity on that socket. Typical client time-out intervals are:

   Two minutes on a UNIX system
   Five minutes on an OS/2 system

2. If you send a message to 30 recipients or less, the entire file can be transmitted in one transaction.

5.7 Overview of LaMail

LaMail is an electronic mail handling system that allows you to view mail, create and send mail, and organize mail into folders.

A folder is a collection of mail files. A folder for incoming mail is automatically created and is named the In Basket folder.

5.8 Setting Up the LaMail Environment

The following sections describe how to set up the LaMail environment.

Subtopics

5.8.1 Recommended Directory Structure

SendMail places all incoming mail in the ETC\MAIL subdirectory. SendMail maintains an index file, INBOX.NDX, which describes the incoming mail. LaMail uses INBOX.NDX when it displays the In Basket folder.

The recommended directory file structure is:

ETC	The TCP/IP program subdirectory.
ETC\MAIL	LaMail places all incoming mail in the MAIL directory and uses the directory contents for the SendMail In Basket folder.

LaMail allows you to move mail from the ETC\MAIL directory to secondary directories. These subdirectories are called user folders. By default, LaMail creates a directory called LAMAIL. Individual subdirectories that represent various folders are created within the LAMAIL directory.

5.8.2 Changing the Default Directory Structure

After LaMail is running, you can follow these steps to change any of the default settings:

1. Select **Options** from the main menu action bar. The Options pull-down menu is displayed.

2. Select **Set Application Options** from the Options pull-down menu. An Applications Options window is displayed containing the various directory fields and the field default values.

The directory fields are:

*.EX files	A set of paths where MAILLIST.EX, EPMLIST.EX, PUT.EX, GET.EX, MATHLIB.EX, BOX.EX, DRAW.EX, and HELP.EX are stored. By default, the *.EX files path is set to BIN.
Configuration file	The directory where the LAMAIL.CF configuration file is stored. By default, the configuration file directory is set to LAMAIL.
Temp files	The directory where temporary files are stored. By default, the temp files directory is set to TMP.
Folders	The directory where .NDX files, the NICKNAME.NAM file, and the root of the folder subdirectories are stored. By default, the folders directory is set to LAMAIL.
In Basket	The directory where LaMail looks for incoming mail. This should be the same place where SendMail is directed to place incoming mail. By default, the In Basket directory is set to ETC\MAIL.

5.8.3 Using Nicknames

A nicknames file organizes information about users into groups. Each group is given a name called **nickname**. Each piece of information consists of a tag and a value; the tag describes what the information is, and the value is the information itself. You use nicknames in LaMail to refer to a user or list of users.

The following example shows the format of the nicknames file.

```
  :nick.ME       :userid.Your ID  :node.Your node
                       :phone.555-1234
                       :name.Your name

  :nick.LARRY    :userid.SMITH :node.Netone.Nettwo.Netthree
                       :name.Larry M. Smith

  :nick.GROUP    :list.LARRY JERRY@YORKTOWN ME
                       :name.Our Group
                       :folder.GROUP
```

For example, if you write a note to SMITH@Netone.Nettwo.Netthree, you would have to enter the long user ID and node name when prompted by the reply function. If Larry Smith is a person to whom you frequently send notes, this would be a cumbersome job. If your names file contains a nickname (LARRY) as shown above, you can simply enter the nickname when prompted by the reply command.

The LaMail names file, NICKNAME.NAM., is located in the LAMAIL subdirectory by default. To change the name or location, select **Set note options** under the Options pull-down menu on the main window.

You can modify the nicknames file at any time using the Nicknames window. Select **Edit names file** under the Options pull-down menu on the main window. For more information, see "Nicknames File Editing" in topic 5.12.7.

The following are the names file tags:

:nick. The user nickname.

:userid. The user ID.

:node. The node's host name. This name should be fully qualified.

:name. The full name of the person.

:phone. The phone number of the person.

:list. Similar to **userid**. This parameter contains a list of predefined nicknames or specific **user@node** addresses, or a combination of the two.

:folder. The folder in which mail from the user ID and node specified by the **:userid.** and **:node.** file tags is received by default.

Notes:

1. You cannot specify a blank character in the **:folder** tag field. However, an underscore character (_) is valid.

2. If an entry in the NICKNAME.NAM file has a **:folder.** tag, the specified folder becomes the default folder for all mail items for that specific entry. If an entry in the NICKNAME.NAM file does not have a **:folder.** tag, the default folder specified by Configure Note Options is used.

5.9 Using LaMail

Use LaMail to send electronic mail to other users on your network. You can start the LaMail function by selecting the LaMail icon in the TCPIPCFG folder on your OS/2 desktop, or type **lamail** at an OS/2 command prompt and press the Enter key.

Note: To send or receive mail, SendMail must be running on your workstation. LaMail can be run without SendMail; however, when you send a note from LaMail, it is queued, but not actually sent, until SendMail has been started.

Various messages are displayed informing you that LaMail is starting. The LaMail main menu is displayed, and the In Basket folder is opened. (See Figure 11 for an illustration of the LaMail Main Menu window.)

PICTURE 11

Figure 11. Example of LaMail Main Menu

The first line of the window contains a title bar, a system menu, and maximizing and minimizing arrows.

The action bar menu containing general LaMail functions appears below the title bar. The action bar menu allows you to select actions to be performed on the mail files or folders that are displayed.

There are six selections on the action bar menu of the LaMail window.

- Notes
- Folders
- Options
- Transfer
- Exit
- Help

The area below the main menu action bar is the LaMail desktop area. The desktop area is where LaMail places its mail-listing or folder windows for a particular mail folder. A folder window is a resizeable window that is clipped by the LaMail main menu. Multiple folder windows can be placed in this area.

Subtopics

5.9.1 Folder Status Information

For each folder, LaMail displays the following status information for the current mail file:

- Sender's name
- Subject
- Reference

The information for Sender, Subject, and Reference is taken from the current mail file, which is highlighted by the box cursor.

A folder listing each mail item in the file is shown below the status information.

Note: The mail file information can be changed on a folder-by-folder basis. For more information, see "Configure Window" in topic 5.12.4.

LaMail also displays the following features:

- The category headers

Header	**Purpose**
Seen	Shows an asterisk if you have already read the file
Date	Indicates the date the mail was sent
Userid	Identifies the sender of the file
Node	Identifies the place from which the file was sent
Subject	Identifies the content of the file

  Note: The contents, color, and position of the category headers can be customized by using the Configure... menu item under the Folders pull-down menu.

- The mail file listing

  The mail file listing is displayed below the category headers. The current file is highlighted by a box cursor. You can use the box cursor to select a mail file from the listing.

  The box cursor can be moved up and down in the folder, which allows you to choose a mail file for further action. You can select multiple mail files by marking each file. Files are marked or unmarked using the space bar, the right mouse button, or the Notes pull-down menu.

- The message line

  A message line appears below the mail file listings. The message line indicates the number of files in your folder along with other information, such as the number of files marked or the number of new files received. The total number of files is updated by using the refresh function.

5.9.2 Selecting an Action

Use the mouse to select an action from the LaMail main menu. The action bar options are used to issue actions on the current mail file or marked mail files. Make action bar choices by:

- Using the mouse and single clicking on the selection
- Pressing the F10 key, then using the arrow keys and the Enter key to select your choices
- Pressing the Alt key and the underlined letter in your choice

Then use the mouse or the specified function keys to select an action from the pull-down menu.

5.9.3 Accelerator Keys

Accelerator keys take you directly to a function, bypassing intermediate windows. The following accelerator keys are defined by LaMail and function as described when used from the LaMail Main window.

Function Key	**Action**
Enter	When in the mail list, brings up the current file for viewing.
F2	Updates the current folder list with the contents of the folder directory.
Tab	Activates the next folder window on the LaMail desktop.
Ctrl+Tab	Toggles the display of the LaMail desktop and the folder list box.
Alt or F10	Activates the action bar of the LaMail application window.
Alt+ Spacebar	Activates the System menu of the Mail list window.
Alt+F1	Displays the current mail item.
Alt+F2	Receives the mail item to the default folder.
Alt+F3	Discards the current item.
Shift+Alt+F1	Displays all marked items.
Shift+Alt+F2	Receives all marked items to the default folders.
Shift+Alt+F3	Discards all marked items.
Alt+M	When in mail list mode, marks all mail files in the active folder.
Alt+F	Opens an existing folder.
Alt+N	Starts a new note. See "Creating Notes" in topic 5.12.1 for more information about creating a note.
Alt+E	Edits a new or existing file.
Alt+W	Displays a list of all LaMail edit windows.
Alt+U	When in mail list mode, unmarks all mail files in the active list.
Alt+I	Marks unmarked mail files and unmarks marked mail files.
Alt+T	Tiles windows. Modifies the window sizes and displays the folders alongside one another. The leftmost folder is the active folder.
Alt+C	Cascades windows. Modifies the window sizes and displays the folders in a stacked arrangement, one behind the other. The topmost folder is the active folder.
Alt+A	Arranges folder icons on LaMail desktop.

5.9.4 Mouse Buttons

The mouse buttons have been defined in LaMail to perform the following actions:

Mouse Button	**Action**
One single click	Moves the box cursor to the pointer
One single click and drag	Moves the box cursor
One double-click	Views the current note
Two single clicks	Toggles the mark function
Two single clicks and drag	Toggles the mark of the line originally containing the cursor; then sets all lines over which the cursor is moved to the status (marked or unmarked) to which the original line was changed

When you make a selection, a pull-down menu for the selected action appears.

5.10 LaMail Main Menu

The following illustrate the pull-down menus available from the LaMail main menu. A description of the menu items associated with each of the pull-down menus is included.

Subtopics

5.10.1 Notes

The Notes pull-down menu contains items that enable you to work with notes within the current folder and display a list of edit windows.

Selection	**Description**
View	You can view the current note, all marked notes, or all the notes in an edit window.
Move	Moves notes, marked notes, or all the notes in the current folder to the default folder or to any other specified folder.
Discard	Discards the current note or all the marked notes.
Print	Prints the current note or all the marked notes.
Archive	Calls a user-supplied program to archive or restore notes.
Create note...	Starts a new note via LaMail Edit. For more information, see "LaMail Edit Menu" in topic 5.11 and "Creating Notes" in topic 5.12.1.
Mark all	Marks all notes for further action.
Unmark all	Unmarks all notes.
Inverse mark	Marks unmarked items and unmarks marked items.
List windows...	Displays a list of all LaMail edit windows.

5.10.2 Folders

The Folders pull-down menu contains items that deal with folder management.

Selection	**Description**
Sort...	Sorts the mail list in the current folder. For more information about sorting, see "Sorting Folders" in topic 5.12.2.
Configure...	Displays a window for you to configure your current folder display environment. For more information about configuring your environment, see "Configure Window" in topic 5.12.4.
Rebuild folder...	Rebuilds corrupted folders. For more information about this option, see "Rebuilding Folders" in topic 5.12.3.
Refresh	Refreshes the list of notes in the current folder.
Search folders...	Invokes the Search Folders window to search for a specified keyword in your mail folders. For more information about searching folders, see "Search Folders Window" in topic 5.12.5.
Folder Manager...	Opens a dialog for managing folders. For more information about managing folders, see "Folder Manager Window" in topic 5.12.6.
Tile Windows	Modifies the size and displays the folders alongside one another. The leftmost folder is the active folder.
Cascade Windows	Modifies the size and displays the folders in a stacked arrangement, one behind the other. The topmost folder is the active folder.

5.10.3 Options

The Options pull-down menu contains items that allow you to change LaMail configuration options and use various utilities.

Selection	**Description**
Edit names file...	Invokes the Nicknames window. For more information about editing nicknames, see "Nicknames File Editing" in topic 5.12.7 and "Viewing or Editing Nicknames Tags" in topic 5.12.8.
Set note options...	Changes note creation defaults For more information about defaults, see "LaMail Note Defaults" in topic 5.12.9.
Set application options...	Opens a window for setting LaMail options and path names. For more information about settings, see "Application Options Window" in topic 5.12.10.
Configure printers...	Opens a window for adding and removing printers from the list.
Save...	Saves the position of the main LaMail window and all the open folders (including those that are minimized). The next time LaMail is started, all of these folders are opened and positioned as they are now. This allows you to keep your most frequently used folders always available on the LaMail desktop.
Sendmail status...	Displays the status of SendMail.

5.10.4 Transfer

The Transfer pull-down menu contains items that allow you import and export information from LaMail.

Selection	**Description**
Prepare remote volume...	Prepares a diskette for using LaMail from a remote location.
Export	Exports notes to a diskette.
Import...	Imports files from a diskette into LaMail.
Configure...	Allows LaMail to be configured for importing and exporting.

5.10.5 Exit

The Exit pull-down menu contains items that allow you to exit or resume LaMail.

Selection	**Description**
Exit LaMail	Exits LaMail
Resume LaMail	Resumes LaMail

5.10.6 Help

The Help pull-down menu contains items that allow you to display help information.

Selection	**Description**
Help for Help...	Displays help on the help task manager
Extended Help...	Displays help on LaMail
Keys Help...	Displays help on LaMail keys
Help Index...	Displays an index of LaMail help
About...	Displays the copyright statement and version number

5.11 LaMail Edit Menu

When you view or create a note, the LaMail Edit window appears, as shown in Figure 12.

PICTURE 12

Figure 12. Example of LaMail Edit Window

The Edit window enables you to edit mail files, modify the basic editor options, and issue commands at the command line.

The Edit window provides the following pull-down menus:

- Mail
- File
- Edit
- Search
- Options
- Command
- Help

The LaMail Edit pull-down menus are described in the following sections. A description of the menu items associated with each of the pull-down menus is included.

Subtopics
5.11.1 Mail
5.11.2 File
5.11.3 Edit
5.11.4 Search
5.11.5 Options
5.11.6 Command
5.11.7 Help

5.11.1 Mail

The Mail pull-down menu contains items that relate specifically to mail.

Selection	**Description**
Reply...	Sends a reply to a note, uses a reply dialog, or forwards the note
Move	Moves mail from the current folder to the default folder or another folder
Discard	Discards mail from the In Basket or a folder
Create new note...	Creates a new note
Send	Sends the new note to the addressees
Add addressees...	Adds addressees to the note currently being edited
Folder...	Changes the folder in which the current note will be saved when it is sent

5.11.2 File

The File pull-down menu contains items that allow you to perform edit functions on any files.

Selection	**Description**
New	Replaces the current file with an empty .Untitled file
Open .Untitled	Opens a new window containing an .Untitled file
Open...	Loads a file into a new window
Import text file...	Gets a copy of a file and inserts it into the current file
Add file...	Loads a file into the current edit window
Rename...	Renames the file currently being edited
Save	Saves the current file
Save as...	Saves the current file under a new name
File	Saves your changes and exits the current file
Quit	Ends or exits the current file without saving data
Print file...	Prints a copy of the current file

5.11.3 Edit

The Edit pull-down menu contains items that allow you to undo changes or perform mark-related actions.

Selection	**Description**
Undo line	Restores a changed line, provided the cursor has not been moved off the line. Repeating this action toggles the change back again.
Undo...	Brings up the undo/redo window, which allows you to step through the changes made to the file and discard or retain them.
Copy	Copies text to the PM Clipboard.
Cut	Copies text to the Clipboard and deletes the marked area.
Paste	Recalls text from the Clipboard. The text is pasted in as if you were copying a character mark.
Paste lines	Recalls text from the Clipboard. The text is pasted in as new lines of text in the file.
Paste block	Recalls text from the Clipboard. The text is pasted in as if you were copying a block mark. The smallest rectangle needed to enclose all of the clipboard text is used.
Style...	Brings up the Style Dialog. You can use this to apply different styles (color and font combinations) to text for viewing and printing. The style information cannot be sent in mail.
Copy mark	Copies marked text to the current cursor location.
Move mark	Moves marked text to the current cursor location.
Overlay mark	Copies and overlays marked text onto existing text at the current cursor location
Adjust mark	Repositions the marked area. This option is similar to the Overlay marked area, except that it also blanks out the source area.
Unmark	Unmarks marked text.
Delete mark	Deletes marked text. **Note:** Deleting a block mark deletes the marked text but leaves the same number of lines in the file. To also delete the lines that the text is on, use a line mark rather than a block mark.
Print mark...	Prints the marked area of text.

5.11.4 Search

The Search pull-down menu contains items that allow you to perform search functions.

Selection	**Description**
Search...	Displays the Search window
Find next	Finds the next occurrence of text
Change next	Replaces the next occurrence of text
Bookmarks	Sets or goes to bookmarks

5.11.5 Options

The Options pull-down menu contains items that allow you to change LaMail configuration and display options.

Selection	**Description**
List ring...	Displays all file names within the current window and switches to any one of them.
Proof	Checks the spelling in the current file.
Proof word	Checks the spelling for the word at the current cursor position.
Synonym	Finds a synonym for the word at the current cursor position.
Preferences	Configures the basic editor options, toggles the stream editing mode and advanced marking mode, and adds or removes stack selections from the Edit pull-down menu.
Autosave...	Displays the active autosave values.
Messages...	Reviews old editor messages for reference.
Frame controls	Selects whether the status line, message line, scroll bars, file icon, and rotate buttons are present; whether the information area is at the top or bottom of the edit window; and whether dynamic prompting for menu selections is enabled.
Save options	Saves the status line as the default value, as well as current settings for the message line, scroll bars, and font. Each time you open an edit window, the same settings are used.
LaMail desktop	Switches control to the LaMail desktop.

5.11.6 Command

The Command pull-down menu contains items that allow you to issue commands and cancel macros.

Selection	**Description**
Command dialog...	Brings up the command line window
Halt command	Stops a macro (command) in progress

Using the Command Line Window: You can issue the following mail-related commands from the LaMail command line window.

Command	**Function**
discard	Discards the current mail file
forward addressee	Forwards the current mail item to one or more people
note addressee	Starts a note
receive [**folder_name**]	Receives the current mail file into the specified or default folder
reply	Starts a note to the sender of the current note or appends a note to the file of the sender of the current note
send	Sends the current note or append

Note: The **addressee** parameter can be specified in one or more of the following formats:

- **user** at **host**
- **user@host**
- **nickname**

See the online help windows for a complete list of all available commands.

5.11.7 Help

The Help pull-down menu contains items that allow you to display help information and start help manager functions.

Selection	**Description**
Help index	Displays an index of LaMail help
General help	Displays help on LaMail
Using help	Displays help on the help task manager
Keys help	Displays help on LaMail keys
Quick reference	Displays a quick-reference summary of the editor functions and an ASCII chart in a browse-mode window
Product information	Displays the version number of LaMail, the editor, and macros

5.12 LaMail Windows

This section describes the various special-purpose windows used in LaMail. The Create Note window is displayed, as shown in Figure 13 in topic 5.12.1, when one of the following actions is taken.

- The Create note menu item is selected from the Notes pull-down menu
- Alt + N is pressed
- The Create new note menu item is selected from the Mail pull-down menu

Subtopics
5.12.1 Creating Notes
5.12.2 Sorting Folders
5.12.3 Rebuilding Folders
5.12.4 Configure Window
5.12.5 Search Folders Window
5.12.6 Folder Manager Window
5.12.7 Nicknames File Editing
5.12.8 Viewing or Editing Nicknames Tags
5.12.9 LaMail Note Defaults
5.12.10 Application Options Window
5.12.11 Remote LaMail Services

5.12.1 Creating Notes

PICTURE 13

Figure 13. Example of LaMail Create Note Window

When you select the Add Addressees menu item while creating a note, the same dialog is displayed indicating Add rather than Create.

The following input fields are displayed in the Create Note window.

Field	**Description**
To:	This field allows you to enter a list of addresses. The list of addresses can use one or more of the following address formats: user at host user@host nickname LaMail uses the address entries to determine the destination of the note. LaMail does not verify the user or the node. If SendMail cannot send the mail, it is returned to the In Basket.
cc: *(Carbon Copy)*	This field allows you to specify a list of addresses, in any of the allowable formats, to receive copies of the note. A copy of the note is sent to each addressee. This is an optional field.
bcc: *(Blind Carbon Copy)*	This field allows you to send a note to a list of addresses, without those addresses appearing in the note file sent. This is an optional field.
Folder:	This field allows you to specify a folder where a copy of the note is placed when the note is sent. This is an optional field.

The window also contains a secondary window listing all the entries from your NAMES file. For each entry, it displays the nickname as well as the person's name. Double-clicking on an entry in the list, while the text cursor is in the To:, cc:, or bcc: fields, adds that nickname to the list.

The following example illustrates the format of the output when you type this input:

```
To:  addressee1 addressee2...
cc:  addressee3
bcc: addressee4
```

The note is created with the following format:

```
To: addressee1
    addressee2
cc: addressee3
bcc: addressee4

   This is the first line of a note...
```

LaMail uses the **addressee** entries to determine the destinations of the note. LaMail does not verify the user or the node. If SendMail cannot send the mail, it is returned to the In Basket.

The maximum length of all the **To:** entries is 255 characters. However, you can add more by assigning a nickname for a list of addresses, or by adding addresses to the **cc:** or **bcc:** fields of the note.

The following pushbuttons are also displayed:

Pushbutton	**Description**
Create	Creates a new note.
Cancel	Exits the window.
Help	Accesses the Help facility.

5.12.2 Sorting Folders

You can sort the contents of a folder by selecting **Sort** from the Folders pull-down menu. The Sort window, as shown in Figure 14, allows you to sort the contents of the folder using various fields as keys.

PICTURE 14

Figure 14. Example of LaMail Sort Window

The title bar displays the name of the current folder, to which the sort is applied. The fields from which you select a sort key are displayed in a list that you can scroll.

Field	**Description**
Nickname	Sorts all files in the current folder window by nickname, if a nickname for a mail item exists.
Userid	Sorts all files in the current folder window alphanumerically by user ID.
Node	Sorts all files in the current folder window alphanumerically by node.
Filename	Sorts all files in the current folder window alphanumerically by file name.
Size	Sorts all files in the current folder window by the size of the file numerically, by number of lines.
Date	Sorts all files in the current folder window by date received.
Time	Sorts all files in the current folder window by time received.
Seen	Specifies that notes you have seen are placed after notes you have not seen, if you select *ascending* under Direction. If you select *descending*, notes that have been seen are placed before notes not seen.
Destination	Sorts all files in the current folder window alphabetically by destination.
Full name	Sorts all files in the current folder window by the sender's name.
Subject	Sorts all files in the current folder window by the Subject: field of the note.
Reference	Sorts all files in the current folder window by the Reference: field of the note.
Full address	Sorts all files in the current folder window by the sender's full address, **userid@host**.

The Sort window allows you to select the direction of the sort:

Ascending	Sorts the field selection in ascending order (lowest to highest).
Descending	Sorts the field selection in descending order (highest to lowest).

The sort is *stable*, which means that if you sort by user ID, then by node, entries with identical nodes are still sorted by user ID.

The Sort window is displayed until you select the cancel pushbutton, allowing you to perform multiple sorts.

The following pushbuttons are also displayed:

Pushbutton	**Description**
Sort	Starts the sort for the highlighted field.
Cancel	Exits the current Sort window and allows you to perform multiple sorts.

Help	Accesses the Help facility.

5.12.3 Rebuilding Folders

You can rebuild folders by selecting the **Rebuild folder...** option from the Folders pull-down menu. The Rebuild Folder window, shown in Figure 15, allows you to specify a folder to be rebuilt if the corresponding index file is corrupted (for example, if mail directory notes are not displayed in the corresponding folder).

PICTURE 15

Figure 15. Example of LaMail Rebuild Folder Window

You can enter the folder name or press the drop-down arrow to the right of the entry field to obtain a list of existing folders from which you can select the folder to rebuild.

The following pushbuttons are also displayed:

Pushbutton	**Description**
Enter	Starts the rebuild process. LaMail requests confirmation that you want the rebuild the folder.
Cancel	Cancels your Rebuild Folder request. Once the rebuild process is started, you can cancel it at any time. The folder is returned to its initial form.
Help	Accesses the Help Facility.

5.12.4 Configure Window

The Configure window allows you to change the attributes of each folder on a per-folder window basis. When you open a folder, including the In Basket, which is opened during LaMail initialization, the corresponding *folder_name*.CFG file is read. If the .CFG file does not exist, the defaults are taken from the LAMAIL.CF configuration file.

For example, while viewing your OLDMAIL folder, if you change the *userid* field color from Black to Red and save these changes, a file named OLDMAIL.CFG is created. This file is placed in the directory defined as the configuration directory. See "Changing the Default Directory Structure" in topic 5.8.2 for more information about your configuration directory.

You can configure the attributes of a folder by selecting **Configure** from the Folders pull-down menu. The Configure window, as shown in Figure 16 in topic 5.12.4, allows you to customize a variety of attributes.

PICTURE 16

Figure 16. Example of LaMail Configure Window

The Configure window allows you to customize the following:

- **Colors**: Allows you to set the colors of the list column headings (referred to as fields) and the color of the main window areas. When you select a field or area in the left list box, the current color for that field is displayed in the right list box.

  In addition to standard colors, you can select **Hidden**, which suppresses the display of that field.

- **Position**: Allows you to set the order in which the column headings (referred to as fields) appear in the list. When you select a field in the left list box, the current position for that field is displayed in the Position entry field. The leftmost field is number 1. Fields are displayed from left to right, skipping any that have a color of **Hidden**.

  To change the field order, select the field in the left list box, then type the new position for that field in the Position entry field.

  For example, suppose you have the following field order defined:

  ```
    1     2     3      4
  Seen  Date  Time  Userid...
  ```

  If you want the **Time** field to appear before the **Date** field, select **Time** in the Field/area list, then enter **2** in the Position entry field. All fields to the right are renumbered appropriately.

  If certain fields that are not hidden are not visible, it is because these fields are shown outside the folder window. Either resize the window or use the horizontal scroll bar to see them.

- **Sort Order**: This allows you to set the order in which the list is sorted when a folder is opened or refreshed. When you select a field in the left list box, the current sort order for that field is displayed in the Sort order entry field. If the field is not included in the sort order, the Sort order field is blank. If the field has a sort order, then the Ascending and Descending pushbuttons in the Sort direction box tell you in which direction the list is sorted. The list is sorted in the specified direction for each field that has a sort order specified, starting with the field with sort order 0. The sort is stable, which means that if you sort by **Userid**, then by **Node**, entries having the same node are sorted by user ID. The last sort performed is the main order in which the folder is presented.

  For example, suppose you want your In Basket to be presented with all the unread notes first, followed by all the ones you have seen; and within each group, you want to see the newest notes first. You would set your sort fields as follows:

  ```
  Time    0    Descending
  Date    1    Descending
  Seen    2    Descending
  ```

  This first sorts by **Time**, then by **Date**, so the newer files are listed before the older ones; finally, it sorts on **Seen**, so that the files marked as having been seen appear after those that have not been seen.

Both groups are still sorted from newest to oldest within the group.

The following pushbuttons are also displayed:

Pushbutton	**Description**
Save	Saves the window settings for the current folder.
Get default	Retrieves the default settings.
Help	Displays a help window for the Configure window.
Ok	Uses the window settings for the current folder. Previously saved settings are not modified.
Set default	Uses the settings as the default for any folder that does not have custom settings.
Cancel	Returns to the settings that were used before the Configure Folder window was displayed and closes the window.

5.12.5 Search Folders Window

You can search through your folders for a specified string by selecting **Search folders** from the Options pull-down menu. The Search folders window, as shown in Figure 17, allows you to search one or more folders. Select the folders you want to search. Enter a search string in the Search for field and select Search from the action pushbuttons. The specified folders are searched for occurrences of the specified search string.

PICTURE 17

Figure 17. Example of Search Folders Window

The following check boxes are also displayed:

Check Box	**Description**
Start	Limits the search to mail dated after a specific day
End	Limits the search to mail dated before a specific day
Case sensitive	The string entered must match exactly
Search index only	Specifies a very fast search of the index file for strings that appear in the subject line or for the sender or recipient
1st occurrence only	Specifies that the search should stop with the first match in each note, rather than showing every match in each note in the result window
Mark found in folders	Specifies that you can have each note in which the search string is found marked in its folder, so you can later perform an operation on all marked notes

5.12.6 Folder Manager Window

You can manipulate your folders by selecting **Folder manager** from the Folders pull-down menu. The Folder Manager window, as shown in Figure 18, is displayed.

PICTURE 18

Figure 18. Example of LaMail Folder Manager Window

The Folder Manager window displays a list of your folders. The following field is also displayed:

Field	**Description**
View	Selects whether the list shows the icons next to the folder names, or just the text.

The following pushbuttons are also displayed:

Pushbutton	**Description**
Create	Creates a new folder.
Open/Show	Opens the selected folder if it does not exist on the LaMail desktop. The folder is restored, if it was minimized, and is brought to the top of the LaMail desktop. Folder Manager is useful for finding folders when you have many items on your desktop. You can leave Folder Manager open while doing other tasks in LaMail.
Rename	Renames the currently selected folder.
Delete	Deletes a selected folder, provided it is empty. Folder Manager does not allow you to delete a folder containing mail. To delete a folder containing mail: 1. Open the folder. 2. Mark all the mail. 3. Select *Notes/Mark All*, or press the Alt and M keys. 4. Discard the mail by selecting *Notes/Discard/All Marked*, or by pressing the Shift, Alt, and F3 keys.
Cancel	Exits the window.
Help	Displays a help window for the Folder Manager window.

The folder is a subdirectory in your Folders directory. See "Changing the Default Directory Structure" in topic 5.8.2 to set your Folder directory. By default, the icon associated with the folder is a picture of a folder. If you want a custom icon for any folder, place the icon in the Folder directory with the name *foldername*.ICO.

The entry box at the bottom of the window contains the selected folder name if you select a folder from the list. You can also type the name of an existing or new folder.

You can have LaMail automatically open your frequently used folders when it starts, with the following steps:

1. Open the folders using the Folder Manager.
2. Position the folders.
3. Select **Save...** from the Options menu.

5.12.7 Nicknames File Editing

You can add, change, and delete entries in your NAMES file by selecting **Edit names file** from the Options pull-down menu. The Nicknames window, as shown in Figure 19, allows you to search or update your NAMES file.

PICTURE 19

Figure 19. Example of LaMail Nicknames Window

A scrollable list of all the nicknames in the file is displayed at the left of the window, and the currently selected nickname is displayed at the top to the right of the list. You can type a new nickname over the selected nickname. Other commonly used NAMES tags, and any values that have been assigned to them, are listed under the nickname.

The pushbuttons on the Nicknames window are:

Pushbutton	**Description**
Add	Adds a new entry to your NAMES file.
Delete	Deletes the currently displayed entry from your NAMES file.
Cancel	Exits the window
Tags...	Displays the Other Tags window, which allows you to create or edit tags other than those that are displayed in the Nicknames window. See "Viewing or Editing Nicknames Tags" in topic 5.12.8 in the next section.
Clear	Clears the fields for the commonly used tags to specify new values.
Change	Updates the currently displayed entry in your NAMES file.
Ok	Exits the window
Help	Displays a help window for the Nicknames window.

¦**Note:** You cannot specify a blank character in the **folder** field. However, ¦an underscore character (_) is valid.

5.12.8 Viewing or Editing Nicknames Tags

You can create and edit additional nicknames tags by selecting the **Tags**... pushbutton on the Nicknames window. The Other tags window, as shown in Figure 20, allows you to view or edit all the tags in a NAMES file entry, rather than just the subset displayed in the Nicknames window.

PICTURE 20

Figure 20. Example of LaMail Other Tags Window

A list of all the tags present for the current nickname is displayed at the left of the window, and the name of the currently selected tag is displayed at the top to the right of the list. You can type a new tag over the selected tag. The tag value is displayed below the tag name.

The pushbuttons on the Other tags window are:

Pushbutton	**Description**
Add	Adds a new tag name and tag value combination to the nickname entry
Delete	Deletes the currently displayed tag name and tag value combination from the nickname entry
Help	Displays a help window for the Other tags window
Clear	Clears the fields for the tag name and value, to make it easier to type a new tag
Change	Updates the tag name and tag value combination in your NAMES file
Ok	Exits the window

5.12.9 LaMail Note Defaults

You can alter the note creation defaults by selecting **Set note options** on the Options pull-down menu. Figure 21 is displayed.

PICTURE 21

Figure 21. Example of LaMail Note Defaults Window

The LaMail Note Defaults window contains the following fields:

Personal Options Fields

Field	**Description**
Your Userid	Specifies your personal user ID.
Your Node	Specifies your host name.
Default Folder	Specifies the folder in which mail is received, if you do not specify a folder name. The default value is ALL.
Time Zone	Specifies the time zone in which you are operating.

Note Header Fields

Field	**Description**
Short Address Format	Specifies that only the user ID and node are used.
Long Address Format	Specifies that LaMail is to look for the recipient in the nicknames file. If the recipient is found, the value for the name tag is used in constructing the reply header. If the recipient is not found, only the user ID and node are used.
No Subject line	Specifies that a subject line is not added when a note header is created.
Add Subject line	Specifies that a subject line is added when a note header is created.
Use :addr. Contents	Specifies that the address of the sender is to be added if it exists in the nicknames file.
Use :phone. Contents	Specifies that the phone number of the sender is to be added if it exists in the nicknames file. **Note:** If the long address format is used, the phone numbers of all addressees found in the nicknames file are included.
Use :folder. Contents	Specifies that the folder tag in the nicknames file is to be used for the target folder when receiving mail from the SendMail In Basket.
Use :notebook. Contents	Specifies that the notebook tag in the nicknames file should be used for the target folder when receiving mail from the SendMail In Basket. Using the notebook tag allows you to use the same nicknames file for LaMail as you use on your VM host.

Note Options Fields

Field	**Description**
Names File	Specifies the location of your nicknames file. You must specify the drive, path, and file name.
Reply-To: Field	Specifies that a Reply-to line is added with the specified ID or nickname to be resolved when a note header is created.
Signature File	Specifies a file containing lines of text that are appended to the reply note. You must specify the drive, path, and file name of the signature file.

For example, a sample signature file can contain your name:

```
Larry A. Mailer
ABC Corporation
(Another Happy LaMail user.)
```

Reply in Same Window
You can answer a note from the window in which you are viewing the note.

Note: This option provides a faster method of answering a note than replying in a new window.

Reply in a New Window
Specifies that a new window is created for a reply note. This option allows you to have both the original note and the reply visible on the screen at the same time.

5.12.10 Application Options Window

You can change options and default directory settings by selecting Set application options on the Options pull-down menu. The Applications Options window, as shown in Figure 22, allows you to change the various options and directory fields.

PICTURE 22

Figure 22. Example of LaMail Application Options Window

The options in the Application Options window are:

Option	**Description**
View mail by folder in one window	You can view all mail in one window. If this option is checked, each time you view a note, it is loaded into the existing View window. If a view window does not exist, a new view window is created. **Note:** Loading a note into an existing window is faster than creating a new window. If this option is not checked, when you issue a View, the file or set of files is loaded into a new window.
Fast browse mode	If this option is checked, when the block cursor moves to a new file in the folder window, the new file is loaded into the view window. This causes the View mail by folder in one window option to be in effect.
Keep window size ratio	Specifies that any open folder windows are to be resized proportionally when the main LaMail window is resized.
Beep on new mail	Specifies that LaMail beeps when new mail is found in the In Basket.
Restore window on new mail	Specifies that the LaMail window will be restored when new mail is found in the In Basket.

The directory fields in the Application Options window are:

Field	**Description**
*.EX files	The path that should be searched for the .EX files used by LaMail. This is generally set by the TCP/IP installation program. Multiple subdirectories can be listed, separated by semicolons. The default value is TCPIP\BIN.
Configuration file	This is where LaMail saves the LAMAIL.CF and any .CFG configuration files when you select Configure from the Folders pull-down menu. The default value is LAMAIL.
Temporary files	This is where LaMail creates any temporary files it needs to use. It should either be placed on a virtual disk, defined with free space on it, or point to a \temp subdirectory. This allows you to erase all the temporary files at one time when necessary. The default value is TMP.
Folders	The directory in which the folder directories and corresponding index and icon files (except for the SendMail In Basket) are searched for and created. The default value is LAMAIL.
In Basket	The directory where new mail is located. This directory should be the subdirectory listed in the Mlocal statement in your SENDMAIL.CF file. The default value is TCPIP\ETC\MAIL.

5.12.11 Remote LaMail Services

To facilitate working with LaMail away from your office workstation, complete the following steps to export your files to a diskette:

1. Select the **Prepare Remote Volume** from the Transfer menu to prepare a diskette.

   Note: You must select **Prepare Remote Volume** the first time that you put files on your diskette and after each import function. This ensures that the notes prepared on your remote workstation have the same appearance as those prepared on your office workstation.

2. Select an **Export mode** to export the notes that you want to transfer to your remote workstation.

   Three export options are available:

   Export Current
   Exports the current note to your diskette for remote use

   Export all marked
   Exports all the marked notes to your diskette

   Export all
   Exports an entire folder to your diskette.

   Note: When exporting multiple volumes, a pop-up window displays the progress of the operation.

3. At the remote workstation, start LaMail with the /Remote option from the command line. If you are working on a different remote volume than that which you originally specified using the configuration option, you can follow this option with a path name.

   For example,

   ```
   start LaMail/remote
   start LaMail/remote b:
   ```

When you return to your office workstation, you can commit the changes by selecting the Import from the Transfer menu. This updates the folders and sends the notes you created on the remote workstation. All of the changes (such as move, discard, and send) are reflected.

6.0 Chapter 6. Using NewsReader/2

This chapter describes the NewsReader/2 program.

Subtopics

6.1 NewsReader/2 Overview

NewsReader/2 is an OS/2 application that you can use to access USENET news groups from a news server. (A news server is a workstation, in your network, that is a repository for the news database.) You can subscribe to news groups, read news articles, and post your own articles.

NewsReader/2 is a client application that requires a news server in your network from which to access the news. Ask your network administrator for the name or IP address of your news server.

6.2 USENET Overview

USENET is a logical network that provides access to a centralized database of world-wide news. Because its underlying physical network uses the client-server model and central storage, you can access large amounts of information quickly using minimal workstation resources.

The news servers that make up the USENET network communicate with each other using the **UNIX-to-UNIX cp** (UUCP) protocol. NewsReader/2 communicates with a news server using the Network News Transfer Protocol (NNTP) which provides access to the centralized news database.

6.3 Configuring NewsReader/2

You can use the Configuration notebook to configure NewsReader/2.

The third page of the Services section of the Configuration notebook, *Configure Services (TELNET, HOSTS, RHOSTS)*, contains a field for the host name or IP address of your news server. You can get this name or address from your network administrator. Figure 23 shows this page of the Configuration notebook.

PICTURE 23

Figure 23. NewsReader/2 Configuration Notebook Fields

If you enter the host name or IP address of the news server, the Configuration notebook sets the NEWSSERVER environment variable to this value in your CONFIG.SYS file. This revision takes effect when your system is restarted.

For an immediate change, use the **set** command from an OS/2 command prompt:

```
set NEWSSERVER=nrserver.raleigh.ibm.com
```

6.4 Starting NewsReader/2

You can start NewsReader/2 by either:

- Selecting the NewsReader/2 icon from the TCP/IP folder
- Entering the **nr2** command from an OS/2 command prompt.

The syntax of the **nr2** command is:

```
>>--nr2------------------------><
          +- server----------¦
          ¦          +-/nc-¦ ¦
          ¦          +-/u--+ ¦
          +-/s---------------¦
          ¦     +-/nc-+      ¦
          +-/nc--------------¦
          +-/?---------------+
```

server Specifies the host name or IP address of the news server to which you want to connect.

/s Causes NewsReader/2 to prompt you to enter the host name or IP address of a news server to access. The news server name that you specify replaces the name currently stored in the NR2.INI file.

/nc Starts NewsReader/2 without connecting to the news server. You can use this option to import a NEWSRC file from a UNIX system. For example, you could import a news file (NEWS.GRP) from a UNIX-format NEWSRC file in your ETC subdirectory.

You can also use this option if you want to learn how to use NewsReader/2 but do not have a news server.

/u Starts NewsReader/2 in update mode. In update mode, NewsReader/2 accesses the news server and obtains an updated set of all news groups for later display in your ALL GROUPS window.

/? Displays help for the **nr2** command.

When you start NewsReader/2, the NewsReader/2 window (shown in Figure 24) is displayed.

If you are starting NewsReader/2 for the first time you are prompted to specify a news server. For additional information, read "Specifying a News Server" in topic 6.4.3.

PICTURE 24

Figure 24. NewsReader/2 Main Windows

From the initial NewsReader/2 window, you can select actions and options that affect the appearance and operation of NewsReader/2.

Subtopics
6.4.1 Selecting NewsReader/2 Windows
6.4.2 Getting Help
6.4.3 Specifying a News Server

6.4.1 Selecting NewsReader/2 Windows

Several smaller windows are contained within the NewsReader/2 window. You can switch from one window to another by:

- Selecting the Window option on an action bar
- Selecting the Switch window option from the system icon in the SUBSCRIPTIONS, HEADERS, ALL GROUPS, or ARTICLE windows
- Using the Tab key to move from one window to another
- Using your mouse to select another window

6.4.2 Getting Help

Extended help and field-specific help are available from all NewsReader/2 windows. You can select the Help option from an action bar or use F1 to obtain help.

6.4.3 Specifying a News Server

You will be prompted to specify a news server if:

- You start NewsReader/2 and the server name has not been specified by one of the following:
  - The **server** option of the **nr2** command
  - The NEWSSERVER environment variable
  - The NR2.INI file from a previous session
- You start NewsReader/2 using the **nr2** command with the /s option, to use a news server other than the one that is set by the NEWSSERVER environment variable or in the NR2.INI file. The server name you specify replaces the previous server name in the NR2.INI file.

6.5 Subscribing to News Groups

When you start NewsReader/2 for the first time, NewsReader/2 reads all news groups from the news server and displays them in the ALL GROUPS window (shown in Figure 25).

You can access this window in future sessions by selecting the All Groups option from the Actions pull-down menu on the SUBSCRIPTIONS window.

PICTURE 25

Figure 25. NewsReader/2 ALL GROUPS Window

When you start NewsReader/2 thereafter, if one more new news groups have been added to the news server since you last accessed it, the NEW GROUPS window is displayed. As new groups are added to the news server, you can add them to your SUBSCRIPTIONS window.

To subscribe to news groups:

1. Double-click on each news group in the ALL GROUPS window (or NEW GROUPS window) that you want to add.

2. Select the Add Group(s) option from the Actions pull-down menu on the ALL GROUPS window (or NEW GROUPS window).

Your selections are added to the SUBSCRIPTIONS window (shown in Figure 26)

PICTURE 26

Figure 26. NewsReader/2 SUBSCRIPTIONS Window

And the list of articles from the first news group is displayed in the HEADERS window (shown in Figure 27).

PICTURE 27

Figure 27. NewsReader/2 HEADERS Window

You can also subscribe to a single news group, using the Add Group option on the Actions pull-down menu in the SUBSCRIPTIONS window.

6.6 Reading an Article from a News Group

To read an article, double-click on the article in the HEADERS window. The contents of the article is displayed in the ARTICLE window (Figure 28).

PICTURE 28

Figure 28. NewsReader/2 ARTICLE Window

When you finish reading an article, you can use the options on the File or Actions pull-down menus on the action bar of the ARTICLE window:

- To save the article in a file or a LaMail folder
- To reply to the article
- To exit the article

6.7 Posting to a News Group

You can select the Post option from the Action pull-down menu on the HEADERS window or the SUBSCRIPTIONS window to post an article to a news group. If you have not previously posted articles or configured NewsReader/2 for posting, you are prompted to provided the following information:

- Your email address
- The name of the organization to which you belong
- A text editor to be used in creating or editing the article you want to post

  Enter the name of the editor (for example, E.EXE) and any arguments. At a minimum, you must specify the % argument to represent a variable for the file name.

  Note: If you specify the EPM.EXE editor, be sure to also specify the /M argument.

After you configure the address and editor information, NewsReader/2 starts your editor, allowing you to create the article to post.

When you finish, save the file using your editor's Save function rather than a Save and Exit option.

NewsReader/2 will then ask if you want to post the article. If you select Yes, NewsReader/2 sends your article to the news server.

6.8 Exiting NewsReader/2

To exit, select the Exit NewsReader/2 option from the Actions pull-down menu on the initial NewsReader/2 window.

When you exit NewsReader/2, the following information is saved for future sessions:

- A list of the news groups to which you subscribed is saved in the NEWS.GRP file

- The name of the news server is saved in the NR2.INI file

Note: If you have resized or repositioned windows, you can save the changes for future sessions by:

1. Selecting the Actions pull-down menu on the main window
2. Selecting the Save Window Positions option

7.0 Chapter 7. Using ASCII-Based Telnet Clients

This chapter describes how to log on to a remote host using an ASCII-based Telnet client. TCP/IP for OS/2 provides three ASCII-based Telnet clients from which to choose: TelnetPM, Telnet, and Telneto.

The TelnetPM client supports ANSI, VT220, VT100, or NVT emulation presented in an easy-to-use Presentation Manager window. TelnetPM also provides line-mode simulation to enable the user to edit a line of text before sending it to the remote host.

The Telnet client supports ANSI, VT220, VT100, or NVT emulation. It runs in an OS/2 window or a full-screen OS/2 session. The Telnet client is somewhat faster than the TelnetPM client, but the TelnetPM client is more user friendly. The Telnet client does not support line-mode emulation.

The Telneto client provides true line-mode support. It runs in an OS/2 window or a full-screen OS/2 session.

Note: For a concise description of commands, subcommands, and parameters, see the *IBM TCP/IP Version 2.0 for OS/2: Command Reference*; and, for more information about line mode, see RFC 1184.

Subtopics

7.1 Setting Up the Environment

The Telnet server uses Dynamic Link Library (DLL) files to implement the supported terminal types. You must specify the path where the DLL files used with Telnet reside. This path is specified using the LIBPATH statement in your CONFIG.SYS file.

The DLL files are:

- VT100.DLL
- ANSI.DLL
- DUMB.DLL

To use the TCP/IP for OS/2 Telnet server, you must define a password for a Telnet client user to specify during logon. If a password has not been defined, the Telnet client user is not allowed to access the server.

The Telnet server uses the environment variable TELNET.PASSWORD.ID as the logon password for connecting Telnet clients. You can set this variable in your CONFIG.SYS file or in the session before starting your Telnet server.

Note: If you specify your logon password for the Telnet server in your CONFIG.SYS file, other users who have access to your CONFIG.SYS can log on remotely to your Telnet server. There are no restrictions on what a remote client can do after logging on to a TCP/IP for OS/2 workstation.

¦ For information about using Telnet with the Network Security Program ¦(NetSP), see the *IBM TCP/IP Version 2.0 for OS/2: Installation and ¦Administration*.

Once you are successfully logged on, a command shell (CMD) is created for the remote Telnet client and the file TELNETD.CMD is run automatically. The TELNETD.CMD file enables you to configure your workstation to perform a task after a Telnet client has connected to the Telnet server. There are no limitations on what you can specify in the TELNETD.CMD file. You can use standard batch file commands or REXX commands.

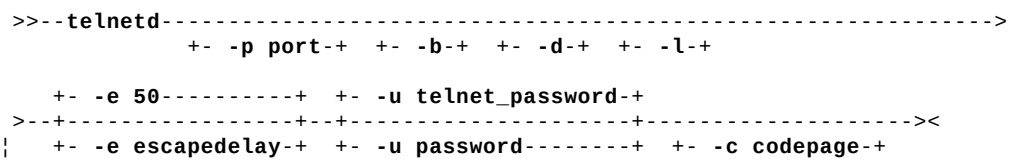
7.2 Setting Up the Server

To allow Telnet logins to your workstation using TCP/IP for OS/2, you must have a Telnet server running on your workstation.

Note: If you use Telnet to connect to an OS/2 host that is running Telnetd, you can run only OS/2 full-screen and OS/2 windowed applications. You cannot run DOS, Windows, or Presentation Manager applications remotely if you are connected to an OS/2 host that is running Telnetd.

To start the server on your local host, you can use INETD or you can type **telnetd** at an OS/2 command prompt and press the Enter key. For more information on using INETD, see *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration*, SC31-6075.

The syntax of the **telnetd** command is:



-p **port**
Specifies the port that Telnetd is to use. If no port is specified, Telnetd uses the well-known Telnet port.

-b Specifies that the copyright information is to be printed.

-d Specifies that debugging is to take place and error messages displayed.

-l Specifies that a log of the Telnetd messages is created.

This option is used in conjuction with the -d option to log debugging information.

-e **escapedelay**
Specifies the time delay for an escape from a Telnet session. The default is 50.

-u **password**
Specifies your password on the remote host. The default is the Telnet password set in the environment variables. If this parameter is not specified and the password is not set in the environment variables, you will receive an error message.

¦-c **codepage**
¦ Specifies the ASCII codepage used by the remote host. If you do not ¦ specify a codepage, the data is sent untranslated. Possible values ¦ are:

¦ TCPDECMU	DEC Multinational codepage
¦ TCP8859	ISO 8859 codepage
¦ NONE	PC codepage 850

The Telnet server listens for connecting clients on a port. Telnet enables you to switch the ports being monitored. Telnet listens to the port defined in the service file of the ETC subdirectory, as the default port.

The Telnet server can display connecting clients in the Telnet server window. This information can be redirected to a log file creating a list of clients connected to a particular server.

The Telnet server can run in debug mode, which produces a series of messages in the TELNETD.LOG file (typically created in the root directory). If the -d parameter is not specified, only critical errors are logged to this file.

As an alternative, you can start Telnetd using INETD, which allows you to start multiple servers from an OS/2 session.

The **telnetd** command starts the TELNETD.EXE program which runs as a task until you shut down the server.

A separate task, Telnet Session, is displayed in the Task List window for each client that establishes a connection with the Telnet server.

7.3 Setting Up for UNIX Clients

Along with the LOGIN.EXE program that is shipped, IBM also supplies a program called LOGINUNX.EXE, which is similar to the login that runs on many UNIX** systems. To use this version of login, you need to create a password file. The format of this file is the same as the format of the /ETC/PASSWD file that exists on any BSD 4.3 compatible UNIX system. IBM does not supply a program to create this passwords file. You must create one on a UNIX system and copy it over to the OS/2 workstation. The file should exist in the ETC subdirectory as specified by the ETC environment variable, and the file name must be PASSWD.

Note: BSD 4.3-compatible UNIX systems include AIX 3 and Sun**OS, among others.

Following is a list of the steps necessary to use the LOGINUNX login application:

1. Create a BSD 4.3-compatible passwords (PASSWD) file and copy it to the ETC subdirectory on the OS/2 workstation.

2. Create a backup for the existing login application:

   ```
   COPY TCPIP\BIN\LOGIN.EXE TCPIP\BIN\OLDLOGIN.EXE
   ```

3. Create a copy of loginunx to be run by the Telnet server:

   ```
   COPY TCPIP\BIN\LOGINUNX.EXE TCPIP\BIN\LOGIN.EXE
   ```

Note: TCPIP is the base directory where TCP/IP has been installed.

7.4 Telnet Clients and the OS/2 Workplace Shell

The TCP/IP for OS/2 Telnet clients are integrated with the OS/2 desktop. You can use the Telnet template located in your templates folder to create icons that represent instances of Telnet sessions without learning or using the Telnet commands and parameters. Once you create and configure an icon for a particular session, you can double-click on it to start the session automatically.

The OS/2 desktop includes a templates folder located in the OS/2 System folder, unless it has been moved. When TCP/IP for OS/2 is installed, it adds a Telnet template to this folder. If you do not find the Telnet template in the templates folder, type **tcpdsktp** on the command line of an OS/2 session and the template will be created.

Use your mouse to create an instance of a Telnet client by following these steps:

1. Move your mouse pointer to the Telnet template, then press and hold down mouse button 2.

2. While holding down mouse button 2, move the mouse pointer to a desktop location where you want to create the instance for the Telnet client, then release the mouse button. The Telnet settings notebook appears.

The Telnet settings notebook allows you to specify a title, an icon, and other session definitions for the instance. Using the settings notebook, you can specify the name of the remote host, the port to use, the screen dimensions of the emulator window you use, and other options that govern the session. All the options in the Telnet settings notebook are optional, and a Telnet instance you create will work without your having to change anything in the notebook. In most cases, however, you will want to specify at least the name of the remote host with which you want to connect.

The first two pages of the settings notebook, Window and General, are used to specify the window minimize behavior, the icon title, and the icon used for the Telnet instance.

The third page of the settings notebook is the Telnet page:

PICTURE 29

Figure 29. Telnet Page of the Telnet Settings Notebook

The **Host** field specifies the remote host to which you want to connect. You can specify the host name, IP address, or an alias. If you leave this field blank, you are prompted to type the host name when you start the Telnet instance.

The **Port** field specifies the specific port to which you want to connect. If you leave this field blank, the default Telnet port is used. You can type either a port number or a port name. If you type a port name, the port number corresponding to that name, as specified in the SERVICES file in the ETC subdirectory, is used.

The **Terminal type** field specifies a terminal type to emulate. Terminal types ANSI, VT220, VT100, and NVT are supported. If you leave this field blank, or if the remote host does not support the terminal type you request, Telnet will negotiate another terminal type automatically.

The **codepage** field specifies how international characters are represented.

When Telnet transfers data in ASCII format, there must sometimes be a translation between the ASCII representation used on the workstation and the version recognized by other hosts on the network. This translation is most important for special characters used in languages other than English. Two codepages are provided for use with TCP/IP for OS/2:

- TCPDECMU for the DEC multinational codepage
- TCP8859 for the ISO codepage 8859-1

You may also specify NONE. If you specify NONE, Telnet does no translation, but passes the text onto the network in the workstation representation. If you do not specify a codepage, no translation is done unless a default codepage has been set using the TCP/IP configuration notebook, TCPIPCFG.

If you always want the TCPDECMU or TCP8859 codepage to be used, without having to specify codepage option, you can set the default codepage with the configuration notebook. The default codepage is set in the Services

section of the notebook. If you do set a default codepage, it is used not only by Telnet, but by FTPPM, FTP, and TelnetPM.

The **Country keyboard** field specifies the emulation of various national keyboards. Your connection with the host may be a 7- or 8-bit connection, depending on what the remote host supports. By emulating a particular national keyboard, you will be able to type some international characters so that the remote host can recognize them, even with a 7-bit connection.

The **Configuration file** field specifies a configuration file to control keyboard mapping and other configuration options for your Telnet session. You can create a configuration file by using **setterm** For more information about **setterm**, see "Configuring Your Keyboard Remappings With SetTerm" in topic 7.6.13.

You can also specify a log file to use for a session log and for debugging information and a trace file to use for trace information.

The fourth page of the settings notebook is the Session page:

PICTURE 30

Figure 30. Session Page of the Telnet Settings Notebook

The Session page specifies whether your Telnet session is run in a full-screen session, an OS/2 window, or a Presentation Manager window. The full-screen version is fastest, but the Presentation Manager version is easiest to use. You can also specify the number of the text rows and columns in the session window.

The fifth page of the settings notebook is the Environment page:

PICTURE 31

Figure 31. Environment Page of the Telnet Settings Notebook

The Environment page contains a multiple line entry field for specifying environment settings to be exported to the remote host. Not all hosts support exporting environment settings, and in such cases, the contents of the environment field are ignored. You can enter one or more lines in the following format:

name=value

These settings specify the environment variables and values to export. Lines should not contain embedded blanks.

You can use the F1 key anytime to get help with the controls in the settings notebook. After completing the settings notebook, double-click on the notebook's system menu icon to save the changes that you have made and close the Telnet settings notebook. You can then use this Telnet instance to start the session anytime by double-clicking on it.

7.5 The Presentation Manager Telnet Client--TelnetPM

You can use the TelnetPM client to log on to a variety of remote hosts. Your workstation can emulate one of these terminals:

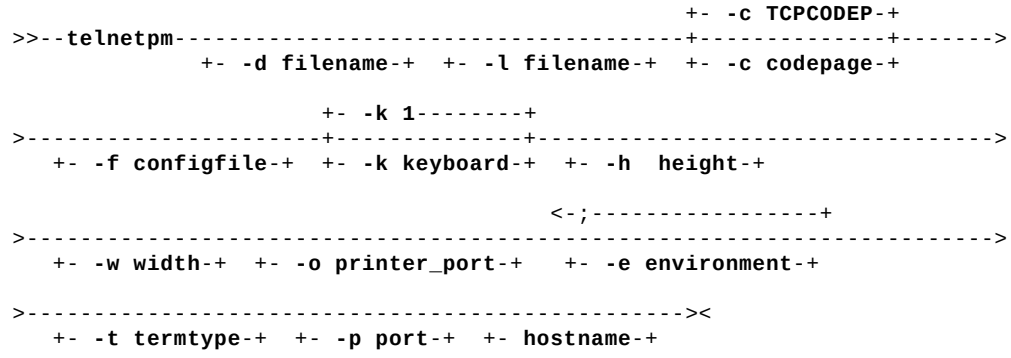
- VT220
- VT100
- ANSI
- HFT
- NVT

You can start the TelnetPM client in one of two ways:

- By creating and then double-clicking on a Telnet instance that you specify as a Presentation Manager session on the Session page of the settings notebook

- By starting the TelnetPM program from the command line of an OS/2 window or full-screen session

The syntax of the **telnetpm** command is:



-d **filename**
Sets debugging on and specifies the fully-qualified file name to be used during debugging. This parameter is useful when diagnosing problems.

-l **filename**
Sets logging on and specifies the fully-qualified file name to be used during logging. When logging is on and you specify a logfile, a log is kept of the Telnet session.

-c **codepage**
Specifies the ASCII codepage used by the remote host. The default is TCPCODEP, which uses the DEC Multinational codepage. Other possible pages are TCP8859 for ISO 8859 codepage, or NONE for PC codepage 850.

-f **configfile**
Specifies the name of a configuration file in the TCPIP\ETC subdirectory to use for establishing a customized keyboard mapping. Users who need to remap the keyboard can create this file using the **setterm** command. See "Configuring Your Keyboard Remappings With SetTerm" in topic 7.6.13.

-k **keyboard**
Specifies emulation of a VT220 national keyboard. Valid values are:

1 North American (the default)
2 British
3 Flemish
4 French Canadian
5 Danish
6 Finnish
7 German
8 Dutch
9 Italian
10 Swiss (French)
11 Swiss (German)
12 Swedish
13 Norwegian
14 French/Belgian
15 Spanish

-h **height**
Specifies the number of rows displayed on the host screen.

For VGA or BGA screens, the valid display combinations by height and width are:

25x80
43x80
50x80

TelnetPM attempts to negotiate the specified width and height with the host. If the parameters are not accepted by the host, Telnet will use the default screen size which is 24x80 for VT100 and VT220 and 25x80 for ANSI and NVT.

-w **width**
Specifies the number of columns displayed on the host screen.

The valid display combinations by height and width are:

25x80
43x80
50x80

TelnetPM attempts to negoiate the specified width and height with the host. If the parameters are not accepted by the host, Telnet will use the default screen size which is 24x80 for VT100 and VT220 and 25x80 for ANSI and NVT.

-o **printer_port**
Specifies a supported local printer port name. The supported printer port names are PRN, LPT1, LPT2, and LPT3. PRN is the default printer port name.

-e **environment**
Specifies a list of environment variables and values to be exported to the host. Specify the environment string using the following format:

name=value;name=value;name=value ...

Do not put any spaces between the entries.

-t **termtype**
Specifies a terminal type to emulate in communicating with the remote host. The following terminal types are supported:

- VT220
- VT100
- ANSI
- HFT
- NVT

If you do not specify a terminal type, Telnet will try to negotiate the terminal type with the remote host. The negotiation order is as shown previously.

-p **port**
Specifies a port number (the address of an application) or a service port name. If a port is not specified, the default Telnet port is used. Port names are mapped to port numbers by the services file.

hostname
Specifies the name, alias, or internet address of a remote host.

When TelnetPM transfers data in ASCII format, there must sometimes be a translation between the ASCII representation used on the workstation and the version recognized by other hosts on the network. This translation is most important for special characters used in languages other than English. Two codepages are provided for use with TCP/IP for OS/2:

- TCPDECMU for the DEC multinational codepage
- TCP8859 for the ISO codepage 8859-1

When TelnetPM is started, the -c option permits you to choose which codepage should be used. If you specify -c NONE, TelnetPM does no translation, but passes the text onto the network in the workstation representation. If you do not use the -c option, no translation is done unless a default codepage has been set using the TCP/IP configuration notebook.

If you always want the TCPDECMU or TCP8859 codepage to be used, without having to specify the -c option, you can set the default codepage with the configuration notebook. The default codepage is set in the Services section of the notebook. If you do set a default codepage, it is used not only by TelnetPM, but by FTP, FTPPM, and Telnet.

Subtopics
7.5.1 Using the TelnetPM Menus

7.5.1 Using the TelnetPM Menus

The TelnetPM client does not have a command shell. Instead, you perform the functions of the command shell by clicking on entries in the menu bar of the session window. The menu bar of the TelnetPM client displays these menus:

Menu	**Description**
Session	Closes a Telnet session
Edit	Copies and pastes text from your emulator window to and from the OS/2 clipboard
Commands	Issues Telnet session commands
Options	Provides options for altering some of the values that govern a Telnet session
Help	Provides help in using the Telnet client

In most Presentation Manager applications, you use the F1 key to get help, the F10 to move the cursor to the menu bar, and other keys for other functions. With TelnetPM, these function keys do not perform the same functions, because you need these keys to send F1, F10, and other keystrokes to the remote host. To access the menu bar in TelnetPM, press the Alt key or use the mouse.

7.5.2 The Session Menu

The Session pull-down menu ends a Telnet session and to log off the host to which you are connected.

Subtopics
7.5.2.1 Open
7.5.2.2 Logoff
7.5.2.3 Quit

7.5.2.1 Open

Open a host session.

7.5.2.2 Logoff

Select **Logoff** both to end a session and to log off the remote host to which you are connected.

7.5.2.3 Quit

Select **Quit** to end a Telnet session without logging off the host to which you are connected.

7.5.3 The Edit Menu

The Edit pull-down menu copies text from the emulator window to the OS/2 clipboard.

Subtopics
7.5.3.1 Copy/Paste

7.5.3.1 Copy/Paste

Use the **Copy** and **Paste** pull-down menu choices in conjunction with your mouse to copy text from your emulator window to the OS/2 clipboard. Follow these steps:

1. Move the mouse pointer to an area of text in the emulator window that you want to copy.

2. Press and hold mouse button 1.

3. While holding down mouse button 1, move the mouse pointer to draw a box around the area to be copied.

4. Release the mouse button.

5. Select the **Copy** item of the Edit pull-down menu.

6. Move the mouse pointer to an area of the emulator window where you want to paste the copied text and click.

7. Select the **Paste** item on the Edit pull-down menu.

The copied text is sent to the remote host with a carriage return following each line.

7.5.4 The Commands Menu

The Commands pull-down menu sends Telnet session commands to the remote host. These commands have varying effects, depending on the host you are using.

Subtopics

7.5.4.1 Abort output

Select **Abort output** to request that the host stop sending output. The result is that no data from the host is displayed until the host acknowledges that the abort has been processed.

7.5.4.2 Are you there?

Select **Are you there?** to request that the host respond with a message if it is still connected and active.

7.5.4.3 Break

Select **Break** to send the equivalent of a break key for those hosts that may expect or respond to a break key.

7.5.4.4 Interrupt process

Select **Interrupt process** to request that the remote host end the currently running process abnormally.

7.5.4.5 Suspend

Select **Suspend** to request that the remote host suspend the currently running process.

7.5.4.6 Synch

Select **Synch** to send a request to the host to discard data that it has received but not processed. For example, use this command to cancel the Paste function.

7.5.5 The Options Menu

The Options pull-down save alters several of the values that govern a Telnet session.

Subtopics
7.5.5.1 Backspace
7.5.5.2 CRLF
7.5.5.3 Cursor
7.5.5.4 Debug...
7.5.5.5 Fonts...
7.5.5.6 Line-mode simulation
7.5.5.7 Local Echo
7.5.5.8 Log...
7.5.5.9 Save options

7.5.5.1 Backspace

Select **Backspace** to send an ASCII backspace to the host. Backspace is the default, but this can be toggled to become an ASCII delete.

7.5.5.2 CRLF

Select **CRLF** to toggle between using the carriage return key as a carriage return followed by a null (the default) or a carriage return line feed.

7.5.5.3 Cursor

Select **Cursor** to alter the way the cursor is displayed during your TelnetPM session.

Your cursor can be defined as follows:

- Character
  - Underscore (Default)
  - Solid block
  - Frame block
- Blink
  - On (Default)
  - Off
- Intensity
  - Normal (Default)
  - Dim

7.5.5.4 Debug...

Select **Debug** to cause all network traffic to be written to a specified log file in hexadecimal and ASCII format.

7.5.5.5 Fonts...

Select **Font...** to choose an alternative font for the Telnet window.

7.5.5.6 Line-mode simulation

Select **Line-mode simulation** to edit full lines of input and send them to the host all at once. An entry field appears below the session window for this purpose. The last ten lines sent in Line-mode simulation are saved automatically so that they can be sent again.

7.5.5.7 Local Echo

Select **Local Echo** to display input on your emulator window as well as on the remote host. The default is on.

7.5.5.8 Log...

Select **Log** to log session information.

7.5.5.9 Save options

Select **Save options** to save the options chosen in this session.

7.5.6 The Help Menu

The Help pull-down menu provides on-line help when using the TelnetPM program.

Subtopics
7.5.6.1 Help Index
7.5.6.2 General Help
7.5.6.3 Using help
7.5.6.4 Keys Help
7.5.6.5 Product Information

7.5.6.1 Help Index

Select **Help index** to display the help index.

7.5.6.2 General Help

Select **General help** to display TelnetPM help information.

7.5.6.3 Using help

Select **Using help** to display help on how to use the help facility.

7.5.6.4 Keys Help

Select **Keys help** to get help in using hot keys instead of the mouse.

7.5.6.5 Product Information

Select **Product Information** to display TelnetPM version and copyright information.

7.6 The OS/2 Window Telnet Client--Telnet

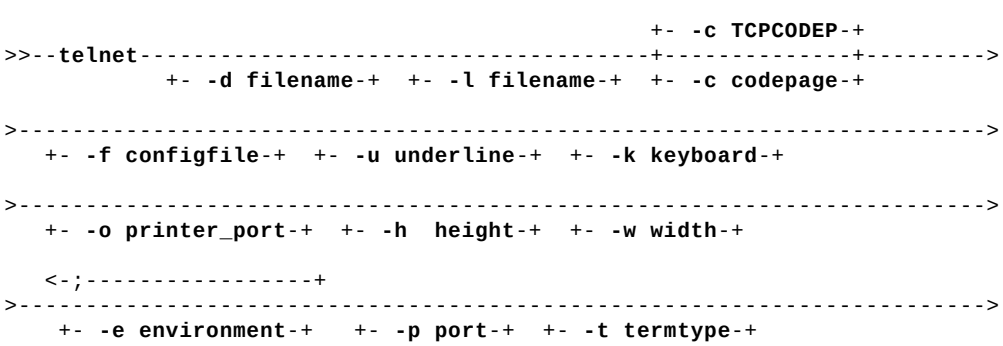
The OS/2 window Telnet client can be run either in a window or in a full-screen session. It is somewhat faster than the Presentation Manager version and may be more familiar to users of UNIX systems. The OS/2 window Telnet client does not support line-mode simulation.

You can start the OS/2 window Telnet client in one of two ways:

- By creating and then double-clicking on a Telnet instance that you specify as either an OS/2 window or full-screen on the Session page of the settings notebook

- By starting the Telnet program from the command line of an OS/2 window or full-screen session.

The syntax of the **telnet** command is:



-d **filename**
Sets debugging on and specifies the fully-qualified file name to be used during debugging. This parameter is useful when diagnosing problems.

-l **filename**
Sets logging on and specifies the fully-qualified file name to be used during logging. When logging is on and you specify a logfile, a log is kept of the Telnet session.

-c **codepage**
Specifies the ASCII codepage used by the remote host. The default is TCPCODEP, which uses the DEC Multinational codepage. Other possible pages are TCP8859 for ISO 8859 codepage, or NONE for PC codepage 850.

-f **configfile**
Specifies the name of a configuration file in the TCPIP\ETC subdirectory to use for establishing a customized keyboard mapping. Users who need to remap the keyboard can create this file using the **setterm** command. See "Configuring Your Keyboard Remappings With SetTerm" in topic 7.6.13.

-u **underline**
Specifies the screen attribute for the underscore character. This value is used only with VT100 and VT220 terminal types, and only when the Telnet program is run in a window. Valid values are:

1 Blue
2 Green (the default)
3 Cyan
4 Red
5 Magenta
6 Yellow
7 Gray

-k **keyboard**
Specifies emulation of a VT220 national keyboard. Valid values are:

1 North American (the default)
2 British
3 Flemish
4 French Canadian
5 Danish
6 Finnish
7 German
8 Dutch
9 Italian
10 Swiss (French)

```
     11  Swiss (German)
     12  Swedish
     13  Norwegian
     14  French/Belgian
     15  Spanish
```

-o **printer_port**
Specifies a supported local printer port name. The supported printer port names are PRN, LPT1, LPT2, and LPT3. PRN is the default printer port name.

-h **height**
Specifies the number of rows displayed on the host screen.

The valid values are 24, 25, 43, and 50.

-w **width**
Specifies the number of columns displayed on the host screen.

For VGA or BGA screens, the valid value is 80.

An additional valid value for XGA or SVGA screens is 132 under full-screen mode.

Telnet attempts to negotiate the specified width and height with the host. If the parameters are not accepted by the host, Telnet will use the default screen size which is 24 x 80 for VT100 and VT220 and 25 x 80 for ANSI and NVT.

-e **environment**
Specifies a list of environment variables and values to be exported to the host. Specify the environment string using the following format:

```
name=value;name=value;name=value ...
```

Do not put any spaces between the entries.

-p **port**
Specifies a port number (the address of an application) or a name.

If a port is not specified, the default Telnet port is used. Port names are mapped to port numbers by the services file.

-t **termtype**
Specifies a terminal type to emulate in communicating with the remote host. Telnet supports:

- VT220
- VT100
- ANSI
- NVT

If you do not specify a terminal type, Telnet will try to do terminal type negotiation with the remote host. The terminal type negotiation order is shown above.

hostname
Specifies the name, alias, or internet address of a remote host.

When Telnet transfers data in ASCII format, there must sometimes be a translation between the ASCII representation used on the workstation and the version recognized by other hosts on the network. This translation is most important for special characters used in languages other than English. Two codepages are provided for use with TCP/IP for OS/2:

- TCPDECMU for the DEC multinational codepage
- TCP8859 for the ISO codepage 8859-1

When Telnet is started, the -c option permits you to choose which codepage should be used. If you specify -c NONE, Telnet does no translation, but passes the text onto the network in the workstation representation. If you do not use the -c option, no translation is done unless a default codepage has been set using the TCP/IP configuration notebook, TCPIPCFG.

If you always want the TCPDECMU or TCP8859 codepage to be used without having to specify the -c option, you can set the default codepage with the configuration notebook. The default codepage is set in the Services section of the notebook. If you do set a default codepage, it is used not only by Telnet, but by FTP, FTPPM, and TelnetPM.

Subtopics
7.6.1 Telnet Subcommands

7.6.1 Telnet Subcommands

The subcommands for the **telnet** command are:

Subcommand	**Function**
? or **help**	Displays help information
close	Closes current connection
display	Displays operating parameters
emulate	Specifies an emulator type to be used before a session is opened
environ	Designates the environment variables sent to the server
open	Connects your workstation to a remote site
quit	Exits the command shell
send	Transmits special characters
set	Sets Telnet variables to a specific value or to true
status	Displays the current status of the connection
toggle	Allows you to toggle operating parameters

Remember to enter the subcommands in lowercase.

7.6.2 Requesting Help

The following shows the syntax of the **?** and **help** subcommands.

The syntax of the **?** subcommand is:

```
>>--?-------------------><
       +- subcommand-+
```

subcommand Specifies the subcommand for which you are requesting help.

The syntax of the **help** subcommand is:

```
>>--help-------------------><
          +- subcommand-+
```

subcommand Specifies the subcommand for which you are requesting help.

Using the **?** or **help** subcommand with no parameter causes Telnet to display a help summary. Using the **?** or **help** subcommand with the **subcommand** parameter causes Telnet to display help information for just that command.

7.6.3 Establishing a Connection for Telnet Client

Use the **open** subcommand with the **host** parameter to establish a connection to a foreign host. The syntax of this subcommand is:

```
>>--open---------------------------><
          +- -p port-+  +- host-+
```

-p **port**
Specifies the destination port to which you are connecting. If you do not specify a port, you are connected to the well-known Telnet port.

host Specifies the host name or an internet address in dotted decimal notation. If you do not specify this value, Telnet will prompt you for it.

7.6.4 Specifying a Terminal Emulator Type

Use the **emulate** subcommand to specify the type of terminal emulator you will use when you connect to a host. The syntax of this subcommand is:

>>--**emulate**-- **termtype**--><

Displaying emulate help

>>--**?**-- **emulate**--><

termtype
The following terminal types are supported:

- VT220
- VT100
- ANSI
- NVT

If you do not specify a terminal type, Telnet will try to do terminal type negotiation with the remote host. The terminal type negotiation order is shown above.

? Displays help information for the **emulate** subcommand.

7.6.5 Obtaining Status Information

Use the **status** subcommand to display the connection status. The syntax of this subcommand is:

```
>>--status--><
```

The following is a sample response from the **status** subcommand.

```
+-------------------------------------------------------------------------------+
¦                                                                               ¦
¦ Connected to 254.23.154.1.                                                    ¦
¦ Emulator Type = VT220                                                         ¦
¦ Escape character is '^]'.                                                     ¦
¦                                                                               ¦
```

7.6.6 Displaying the Set and Toggle Values

Use the **display** subcommand to display the **set** and **toggle** values.

The syntax of the **display** command is:

>>--**display**--><

7.6.7 Designating the Environment Variables

Use the **environ** subcommand to designate the environment variables that are sent through the Telnet ENVIRON option. The initial set of variables is taken from your environment.

Note: The variables **DISPLAY** and **PRINTER** are sent by default.

The syntax of the **environ** subcommand is:

```
>>--environ---- define option value----><
              +- name=value----------¦
              +- undefine options----¦
              +- export options------¦
              +- unexport option-----¦
              +- list----------------+
```

Displaying environ help

```
>>--?-- environ--><
```

define **option value**
: Defines the variable **option** to have a specific value. Any variables defined by this subcommand are automatically exported. The **value** can be enclosed in single or double quotes so that tabs and spaces can be included.

name=**value**
: Attempts to export an environment setting to the remote host. Not all hosts support this function.

undefine **options**
: Removes the options specified from the list of environment variables.

export **options**
: Marks the options that are to be exported to the remote host.

unexport **option**
: Marks the option that is not to be exported unless explicitly requested by the remote host.

list
: Lists the current set of environment variables. Those marked with an asterisk (*) are sent automatically. The other variables in the list are sent only if explicitly requested.

?
: Displays help information for the **environ** subcommand.

7.6.8 Setting the Operating Parameters

Use the **set** subcommand to set any Telnet variable to a specific value.

The **set** subcommand's variables can be displayed with the **display** subcommand.

The syntax of the **set** subcommand is:

```
>>--set---- erase---------- value--><
         +- escape------¦
         +- flushoutput-¦
         +- interrupt---¦
         +- kill--------¦
         +- quit--------+
```

Displaying set Help

```
>>--?--set--><
```

erase
: Erases a character (sends **ec**).

escape
: Defines the current escape character.

flushoutput
: Sends a Telnet ao (abort output) sequence to the remote host, if Telnet is in localchars mode.

interrupt
: Interrupts a character (sends **ip**).

kill
: Erases a line (sends **el**).

quit
: Sets a break character (sends **brk**).

?
: Displays help information for the **set** subcommand.

7.6.9 Toggling the Operating Parameters

Use the **toggle** subcommand to toggle (between **on** and **off**).

You can specify multiple arguments and display the state of these flags with the **display** subcommand.

The syntax of the **toggle** subcommand is:

```
>>--toggle---- autoflush-----><
            +- bs---------¦
            +- crlf-------¦
            +- crmod------¦
            +- debug------¦
            +- localchars-¦
            +- log--------¦
            +- wrap-------+
```

Displaying toggle Help

```
>>--toggle-- ?--><
```

autoflush If autoflush and localchars are both true, when the ao (abort output) or quit characters are recognized and transformed into Telnet sequences, Telnet will not display any data on the user's workstation until the remote system acknowledges that it has processed those Telnet sequences. The initial setting is true.

bs Toggles the sending of the backspace key value as an ASCII delete character (X'7F') or as a backspace character (X'08'). The initial setting sends the backspace key as a backspace character.

crlf If this is true, carriage returns (CR) are sent as carriage return line feed (CRLF). If this is false, carriage returns are sent as carriage return null (CRNUL). The initial setting is false.

crmod Toggles carriage return mode. When this mode is enabled, carriage return characters received from the remote host are mapped into a carriage return followed by a line feed. This mode affects only those characters received from the remote host. Use this mode when the remote host sends only carriage return, never line feed. The initial setting is false.

debug Toggles the recording of session debugging information. If you did not specify a debug filename when you started Telnet, you will be prompted to specify one.

localchars
If this is true, the flush, interrupt, quit, erase, and kill characters are recognized locally and transformed into appropriate Telnet control sequences.

log Toggles the recording of logging information. If you did not specify a logging filename when you started Telnet, you will be prompted to specify one.

wrap Toggles wrap on and off. The initial setting is on.

? Displays help information for the **toggle** subcommand.

7.6.10 Transmitting Special Characters

The **send** subcommand sends one or more special character sequences to the remote host. The syntax of this command is:

```
>>--send---- ao-----------><
          +- ayt-------¦
          +- brk-------¦
          +- interrupt-¦
          +- synch-----+
```

Displaying send Help

```
>>--?-- send-->
```

ao Sends the Telnet ao (abort output) sequence, which causes the remote system to move all output from the remote system to the user's workstation.

ayt Sends the Telnet ayt (are you there) sequence, to which the remote system can respond.

brk Sends the Telnet brk (break) sequence, which can have significance to the remote system.

interrupt Sends the Telnet abort (end processes) sequence.

synch Sends the Telnet synch sequence. This sequence causes the remote system to discard all previously typed (but not yet read) input. This sequence is sent as TCP urgent data and might not work if the remote system is a 4.2 BSD system. If it does not work, a lowercase r might be echoed on your workstation.

? Displays help information for the **send** subcommand.

7.6.11 Closing the Current Connection

You can close the current connection by using the **close** subcommand.

This subcommand disconnects you from any Telnet server to which you are connected and returns you to the command prompt.

The syntax of the **close** subcommand is:

>>--**close**--><

7.6.12 Ending the Session

Use the **quit** subcommand to close any open Telnet session and exit Telnet. The syntax of this subcommand is:

```
>>--quit--><
```

7.6.13 Configuring Your Keyboard Remappings With SetTerm

Use the SetTerm program to configure keyboard remappings for your window/full screen and TelnetPM applications.

This section describes the parameters and options in the sequence in which they are displayed by the SetTerm program. Additionally, this section provides cursor movement instruction (both keyboard and mouse) and brief explanations of each menu item.

You can select an option, item, or field and press F1 to obtain online help for that selection. SetTerm allows you to:

- Perform file level activities
- Define key mappings
- Choose keyboard types

The settings are stored in a configuration file, which is referenced when Telnet is started with -f cfg_filename option. By default, newly created configuration files will be saved to the ETC subdirectory as TELNET.CFG unless you specify a different name and path with "Save As..." option.

The syntax of the **setterm** command is:

```
>>--setterm-----------------------------><
             +- -fs -+  +- -f cfile-+
```

Displaying setterm Help

```
>>--setterm-- -?--><
```

-fs Instructs SetTerm to start execution in full-screen mode. If OS/2 is currently in window mode, SetTerm switches to full-screen mode before it starts up.

-f **cfile**
¦ Instructs SetTerm to access the **cfile**.CFG configuration file in the ¦ ETC subdirectory for the SetTerm definitions and display a menu to ¦ allow changes to the file. If you issue the **setterm** command without ¦ this parameter, you access the TELNET.CFG file in the ETC ¦ subdirectory.

-? Displays information about the **setterm** command.

Specifying the **setterm** command without any parameters creates or modifies the TELNET.CFG file in the ETC subdirectory.

7.6.14 Moving Around SetTerm Screens

Keyboard movement

To move the cursor between different fields, use the Tab or Shift-Tab keys. If the input field ends with a down-arrow symbol (□), press the Down Arrow key to access a drop-down list of selectable items. You can use the arrow keys to move the cursor within the list. To select (or implement) a choice from the list, press the Enter key.

Mouse movement

You can use the mouse to move the cursor around the screen, and select pushbuttons, options, or fields.

In a drop-down list, you can use the mouse to click on the scroll bar to scan the list, and to double-click on an item you want to select.

Pushbuttons

To select a pushbutton, use the Tab key or Shift-Tab key to move the cursor to the pushbutton. Press Enter to activate your selection. You can also use the mouse to select a pushbutton.

7.6.15 The SetTerm Program

To start SetTerm, type **setterm** at the OS/2 command prompt. SetTerm is used for both full-screen Telnet and TelnetPM.

If you did not set up your ETC environment variable, SetTerm displays an error message and terminates. Before you run SetTerm again, use the OS/2 **set** command to set your ETC environment variable.

The SetTerm program displays a menu bar from which you can choose the main functions: File, Global, Keymap, and Help. When you select one of the functions, the SetTerm program displays a pull-down menu from which you can make several selections.

- **File**

  Enables you to perform file management tasks, such as opening and saving files and changing the current directory.

- **Global**

  Enables you to select keyboard type. Fifteen different keyboard types are supported.

- **Keymap**

  Enables you to define or list key mappings of the supported terminal emulators.

- **Help**

  Provides information on how to use online help for **setterm**.

The status line at the bottom of the SetTerm window displays information about the active option, field, or window.

7.7 The File Function

The File function of the SetTerm program allows you to work with configuration files and perform file related activities to meet the needs of your application.

When you select the File function, the SetTerm program displays a pull-down menu. When you select an item from the pull-down menu, the SetTerm program displays the fields and options for that selection. You can type values in the fields, select or enable available options, or use appropriate pushbuttons to implement your selections or discard your entries and make another selection.

Some of the file functions are associated with certain hot keys that are in effect globally. You can uses these hot keys to invoke a function without going to the menu bar. For example, you can press F2 to make SetTerm save a change to the current configuration file.

You can select an option or a field and press F1 to obtain online help about the item or field.

The following section describes the parameters and options associated with each File pull-down menu selection.

Open (F3)

Allows you to open an existing SetTerm configuration file.

To open a configuration file, you can type the file name in the Name field of the Open File window, or you can select it from the list of available files. The file extension of the SetTerm configuration file is CFG.

Save (F2)

Stores new settings into the default SetTerm configuration file, ETC\TELNET.CFG, or into a previously specified file.

Save as

Allows you to save your configuration file and specify a particular file name.

Type the file name in the Save As field, or select an existing name from the Files list.

Change Directory (Change dir)

Allows you to list directory names, display the directory tree, change directories and drives, or restore previous settings.

Use the Directory name field to type the desired directory, or select from the Directory tree field. When this window is opened, the name of the current directory is displayed in the Directory name field. The Directory tree field displays the available drives and the associated file names. For example, selecting C: will display the names of files that are on your C: drive. Move through the file names and make a selection. The name of the path that you choose will be displayed in the Directory name field.

Use the pushbuttons as follows:

Change Implements the specified directory change
Revert Returns to the current working directory
OK Returns to the main SetTerm window

Save and Exit (F4)

Stores new settings into the default SetTerm configuration file, ETC\TELNET.CFG, or into a previously specified file and exits SetTerm immediately thereafter.

Exit (Alt X)

Exits SetTerm.

If you have changed your configuration, but have not saved those changes, a warning window appears. This warning window states that the SetTerm configuration file has been changed, and gives you the opportunity to save or cancel your changes.

7.8 The Global Function

The Global function of the SetTerm program enables you set parameters to be used for all activities during the session.

When you select the Global function, the SetTerm program displays a pull-down menu. When you select an item from the pull-down menu, the SetTerm program displays the fields and options for that selection. You can type values in the fields, select or enable available options, or use appropriate pushbuttons to implement your selections or discard your entries and make another selection.

You can select an option or a field and press F1 to obtain online help about the item or field.

The keyboard selection option is described below:

Keyboard Selection Allows you to specify the keyboard type. This option is a National Language Support (NLS) function.

Use this Keyboard Selection window to view a list of choices. Select the desired keyboard type; your choice is automatically displayed in the Keyboard field.

The following list shows keyboard types supported by SetTerm and Telnet.

- North American (the default)
- British
- Flemish
- French Canadian
- Danish
- Finnish
- German
- Dutch
- Italian
- Swiss (French)
- Swiss (German)
- Swedish
- Norwegian
- French/Belgian
- Spanish

7.9 Using the Keymap Function to Remap the Keyboard

The Keymap function of the SetTerm program allows you define emulator characteristics.

This choice lets you customize some of the functions for each of the following terminal types:

- VT220
- VT100
- ANSI

When you choose the terminal type, a cascaded pull-down menu appears listing the options for defining and viewing key mapping, specifying attributes, and setting video characteristics.

When you select the Keymap function, the SetTerm program displays a pull-down menu. When you select an item from the pull-down menu, the SetTerm program displays the fields and options for that selection. You can type values in the fields, select or enable available options, or use appropriate pushbuttons to implement your selections or discard your entries and make another selection.

You can select an option or a field and press F1 to obtain online help about the item or field.

The following sections describe the cascaded pull-down menu options associated with each terminal type.

Define Key Mappings

Allows you to remap a key and its defined shift status to a function, a string, a character, or a hexadecimal value. When you select this option, a keyboard diagram is displayed on which you select the keys you want to redefine.

When you select a key on the keyboard diagram, the Change Key Assignment window is displayed. It contains an input field for each of the four or five types (including AltGr) of shift status that you can use to remap a key: Base (unshifted), Shift, Alt, Ctrl, and AltGr. In each input field, use the Down Arrow to display a list of supported terminal emulator functions, then use the arrow keys to scan the list and the Enter key to select a function.

To remap a key to a function

Type the function name enclosed in square brackets ([FUNCTION]), or select the function from the drop-down list. The function list contains different sets for each emulator.

To remap a key to a string

Type the string enclosed in double quotes ("any string"). The string cannot be longer than 80 characters. If you want to include Esc or Enter in your string, enclose it in greater than and less than symbols (<ESC> or <CR>).

To remap a key to a character

Type the character enclosed in single quotes ('c').

To remap a key to a one-byte hexadecimal value

Type a one-byte hexadecimal value in the format 0xNN, where NN is any valid one-byte hexadecimal value.

Note: AltGr is supported for keyboards that support this function. AltGr maps to the right Alt key on the 102-character keyboard.

List Key Mappings

Displays keys that are currently remapped for the selected terminal type.

The Current Settings window shows selected keys, the associated shift status, and the definition. If no keys have been defined, a message is displayed.

7.10 Keys that Cannot Be Remapped with SetTerm

Certain keys and key sequences cannot be remapped using SetTerm. Both window/full screen and TelnetPM have a small number of keys that are predefined and are used by the application, by OS/2, or by Windows:

Alt	Pause
AltGr (not on US keyboard)	PrintScreen
Caps Lock	Scroll Lock
Ctrl	Shift
Num Lock	Space Bar

The following key sequences cannot be remapped:

Alt-F4	Ctrl-1
Alt-F5	Ctrl-3
Alt-F7	Ctrl-4
Alt-F8	Ctrl-5
Alt-F9	Ctrl-7
Alt-Tab	Ctrl-8
Alt-Esc	Ctrl-9
Alt-Home	Ctrl-0
Shift-Esc	Ctrl-s
Ctrl-Esc	Ctrl-p

Certain DEC key sequences and some key sequences in Auxiliary keypad and Editor keypad cannot be remapped because the key sequences are not supported by OS/2. Whether a DEC key sequence can or cannot be mapped depends on the keyboard type.

No input field of the shift status is shown on the Change Key Assignment Window if the key combination cannot be remapped.

7.11 Using the Telnet Line-mode Client--Telneto

You can use the line-mode client, Telneto, when you need true line-mode instead of the line-mode simulation provided by the Presentation Manager Telnet client.

Subtopics

7.11.1 Setting Up the Telnet Client Environment for Telneto

The Telneto client uses the environment variable TERM. The client sends the value of the TERM variable to a Telnet server during options negotiation. If the TERM environment variable is not set, the variable default VT220 is sent to the server as the client's terminal type.

If you are using the Telneto client to communicate with a TCP/IP for OS/2 Telnet server, you should set the TERM environment variable to ANSI.

You can customize session settings of the Telneto line-mode client by creating a TELNET.RC file. You cannot reconfigure the keyboard, nor can you use the SetTerm utility when using the line-mode client.

The syntax of the **telneto** command is:

```
>>--telneto-------------------------------------------------------------->
             +- -8-+  +- -E-+  +- -L-+  +- -c-+  +- -d-+

                                                    +- -u 1------+
>---------------------------------------------------+------------+------->
   +- -e escape_char-+  +- -n tracefile-+  +- -r-+  +- -u attrib-+

>---------------------------------><
   +- -N-+  +- host------------+
                    +- port-+
```

-8 Specifies an 8-bit data path. This causes an attempt to negotiate the binary option on both input and output.

-E Prevents any character from being recognized as an escape character.

-L Specifies an 8-bit data path on output. This causes the binary option to be negotiated on output.

-c Disables the reading of the user's TELNET.RC file.

-d Sets the initial value of the debugging toggle to on.

-e **escape_char**
Sets the initial escape character to **escape_char**. If this parameter is not specified, there is no escape character defined.

-n **tracefile**
Opens the trace file for recording trace information.

-r Specifies a user interface similar to rlogin. In this mode, the escape character is set to the tilde (~), unless modified by the -e parameter. When in rlogin mode, a line of the form tilde (~".") disconnects you from the remote host. This is similar to the line ~^Z, which also suspends the session.

-u **attrib** Replaces the default highlighting for data that is displayed with the underscore attribute on. The default is 1, which displays the data dark blue on a color screen and underlined on a monochrome screen. Valid values for **attrib** are 1-7.

-N Ignores the 8th bit of data when in VT220 7-bit mode. Unless you specify -N, this bit is honored when interpreted as a VT220 data stream.

host Specifies the name, alias, or internet address of a remote host.

port Specifies a port number (the address of an application) or a name. If a number is not specified, the default Telnet port is used.

Port names are mapped to port numbers by the services file. When a port number is specified, Telneto does not send out any initial Telnet option negotiation. If the port number or name is preceded by a minus sign, the initial Telnet option negotiation is sent.

7.11.2 Telneto Subcommands

The subcommands for the **telneto** command are:

Subcommand	**Function**
!	Issues an OS/2 command
?	Displays help information
close	Closes current connection
display	Displays operating parameters
environ	Designates the environment variables sent to the server
logout	Closes the current connection
mode	Changes the mode type
open	Connects your workstation to a remote site
quit	Exits the command shell
send	Transmits special characters
set	Sets Telneto variables to a specific value or to **true**
slc	Sets or changes the state of the special characters
status	Displays the current status of the connection
toggle	Allows you to toggle operating parameters
unset	Disables Telneto variables or sets them to **false**

Remember to enter the subcommands in lowercase.

7.11.3 Requesting Help

The syntax of the **?** subcommand is:

```
>>--?-------------------><
       +- subcommand-+
```

subcommand Specifies the subcommand for which you are requesting help

The syntax of the **help** subcommand is:

```
>>--help-------------------><
          +- subcommand-+
```

subcommand
Specifies the subcommand for which you are requesting help.

Using the **?** or **help** subcommand with no parameter causes Telneto to display a help summary.

Using the **?** or **help** subcommand with the **subcommand** parameter causes Telneto to display help information for just that command.

7.11.4 Establishing a Connection for Telneto Client

Use the **open** subcommand with the **host** parameter to establish a connection to a foreign host. The syntax of this subcommand is:

```
>>--open-------------------------><
          +- host-------------+
                  +- -port-¦
                  +- port--+
```

host Specifies the host name or an internet address specified in dotted decimal notation. If you do not specify this value, Telneto will prompt you for it.

-port Specifies that, when the port number is preceded by the minus sign, the initial option negotiation is done.

port Specifies the destination port to which you are connecting. If you do not specify a port, you are connected to the well-known Telnet port.

After **open** establishes a connection, if the skiprc toggle is not enabled, the TELNET.RC file in the ETC subdirectory is opened. For more information about the **skiprc** parameter, see page 7.11.9.

7.11.5 Obtaining Status Information

Use the **status** subcommand to display the following status information:

- Connection status
- Operating mode

The syntax of the **status** subcommand is:

```
>>--status--><
```

The following is a sample response from the **status** subcommand.

```
Connected to 254.23.154.1.
Operating in character-at-a-time mode.
Escape character is '^]'.
```

7.11.6 Displaying the Set and Toggle Values

Use the **display** subcommand to display the **set** and **toggle** values.

The syntax of the **display** subcommand is:

```
             <-------------+
>>--display-------------------><
              +- argument-+
```

argument Enables you to specify any of the **set** and **toggle** parameters. If you do not specify a value, Telneto displays all the values that you can specify.

You can designate more than one argument with this subcommand. To see all the **set** arguments or all the **toggle** arguments, type **display set** or **display toggle**.

7.11.7 Designating the Environment Variables

Use the **environ** subcommand to designate the environment variables that are sent through the Telnet ENVIRON option. The initial set of variables is taken from your environment.

Note: The variables **DISPLAY** and **PRINTER** are sent by default.

The syntax of the **environ** subcommand is:

```
>>--environ---- define option value----><
              +- undefine options----¦
              +- export options------¦
              +- unexport option-----¦
              +- send----------------¦
              +- list----------------+
```

Displaying environ Help

```
>>--environ-- ?--><
```

define **option value**
: Defines the variable **option** to have a specific value. Any variables defined by this subcommand are automatically exported. The **value** can be enclosed in single or double quotes so that tabs and spaces can be included.

undefine **options**
: Removes the options specified from the list of environment variables.

export **options**
: Marks the options that are to be exported to the remote host.

unexport **option**
: Marks the option that is not to be exported unless explicitly asked for by the remote host.

send
: Sends an environment variable.

list
: Lists the current set of environment variables. Those marked with an asterisk (*) are sent automatically. The other variables in the list are sent only if explicitly requested.

?
: Displays help information for the **environ** subcommand.

7.11.8 Setting the Operating Parameters

Use the **set** subcommand to set any Telneto variable to a specific value.
The syntax of this subcommand is:

```
>>--set---- autoflush-----------><
          +- autosynch---------¦
          +- ayt value---------¦
          +- binary------------¦
          +- bs----------------¦
          +- crlf--------------¦
          +- crmod-------------¦
          +- debug-------------¦
          +- ebol value--------¦
          +- echo value--------¦
          +- ecr value---------¦
          +- eeol value--------¦
          +- eof value---------¦
          +- erase value-------¦
          +- escape value------¦
          +- ewr value---------¦
          +- ewl value---------¦
          +- flushoutput value-¦
          +- forw1 value-------¦
          +- forw2 value-------¦
          +- inbinary----------¦
          +- insrt value-------¦
          +- interrupt value---¦
          +- kill value--------¦
          +- lnext value-------¦
          +- localchars--------¦
          +- mcbol value-------¦
          +- mceol value-------¦
          +- mcl value---------¦
          +- mcr value---------¦
          +- mcwl value--------¦
          +- mcwr value--------¦
          +- netdata-----------¦
          +- options-----------¦
          +- outbinary---------¦
          +- over value--------¦
          +- prettydump--------¦
          +- quit value--------¦
          +- reprint value-----¦
          +- rlogin value------¦
          +- skiprc------------¦
          +- start value-------¦
          +- stop value--------¦
          +- susp value--------¦
          +- tracefile value---¦
          +- worderase value---+
```

Displaying set Help

```
>>--set-- ?--><
```

autoflush Specifies that deleting of output occurs when sending interrupt characters.

autosynch Specifies automatic sending of interrupt characters in urgent mode.

ayt Sets the key sequence to send a Telnet ayt.

binary Specifies that the sending and receiving of data is binary.

bs Changes the backspace key from an ASCII backspace to an ASCII delete.

crlf Specifies that carriage returns are sent as <CR><LF>.

crmod Starts the mapping of received carriage returns.

debug Starts debugging in Telneto.

ebol Defines the erase-to-beginning-of-line character.

echo Sets the key sequence to toggle between local echoing of entered characters (for normal processing) and suppressing echoing of entered characters (for example, a password) when in line mode. If you do not specify this key sequence, the echo sequence is ^E.

ecr Defines the character for deleting a character to the right.

eeol Defines the character for deleting to end of the line.

eof Specifies the character to be sent to the remote system when it is the first character on a line. If you do not specify this key sequence, the eof sequence is ^Z.

erase Sets the key sequence to cause an erase character to be sent.

escape Defines the current escape character.

ewr Defines the character for deleting a character to the left.

ewl Defines the character for deleting a word to the left.

flushoutput
Sets the key sequence to send a Telnet ao (abort output) sequence to the remote host, if Telneto is in localchars mode.

forw1, forw2
Sets the key sequence to forward partial lines to the remote system.

inbinary Specifies that data being received is binary.

insrt Defines the character for entering insert mode.

interrupt Sets the key sequence to send an interrupt character (sends **ip**).

kill Sets the key sequence to send an erase line (sends **el**).

lnext Specifies the character to be taken as the terminal's next character.

localchars
Starts the local recognition of certain control characters.

mcbol Defines the character for moving the cursor to the beginning of the line.

mceol Defines the character for moving the cursor to the end of the line.

mcl Defines the character for moving the cursor to the left one character.

mcr Defines the character for moving the cursor to the right one character.

mcwl Defines the character for moving the cursor to the left one word.

mcwr Defines the character for moving the cursor to the right one word.

netdata Specifies that printing of hexadecimal representation of network data (debugging) will take place.

options Specifies that Telneto will show options processing (debugging).

outbinary Specifies that data is sent in binary.

over Defines the character for entering character replace mode.

prettydump
Specifies that Telneto will print readable output for "netdata" (debugging).

quit Defines a sequence break character (sends **brk**).

reprint Specifies the character to be taken as the terminal's reprint character.

rlogin Specifies the rlogin escape character. If set, the escape character is ignored unless it is preceded by this character. This character, followed by a period (.) at the beginning of the line, closes the connection. When followed by a ^Z, it suspends the **telneto** command. The initial setting is to have the rlogin escape character disabled.

skiprc Specifies that Telneto should not read the TELNET.RC file.

start Specifies the character to be used as the terminal's start character if the toggle-flow-control option has been enabled.

If you do not specify this key sequence, the start sequence is ^S.

stop Specifies the character to be used as the terminal's stop character if the toggle-flow-control option has been enabled. If you do not specify this key sequence, the stop sequence is ^Q.

susp Sets the key sequence to send a Telnet susp (suspend process) sequence to the remote host if Telneto is in localchars mode, or LINEMODE is enabled, and the suspend character is typed.

tracefile Specifies the file to which the output, caused by netdata, termdata, or option tracing being true, is written. If you do not specify this key sequence, the start sequence is - and tracing information is written to standard output.

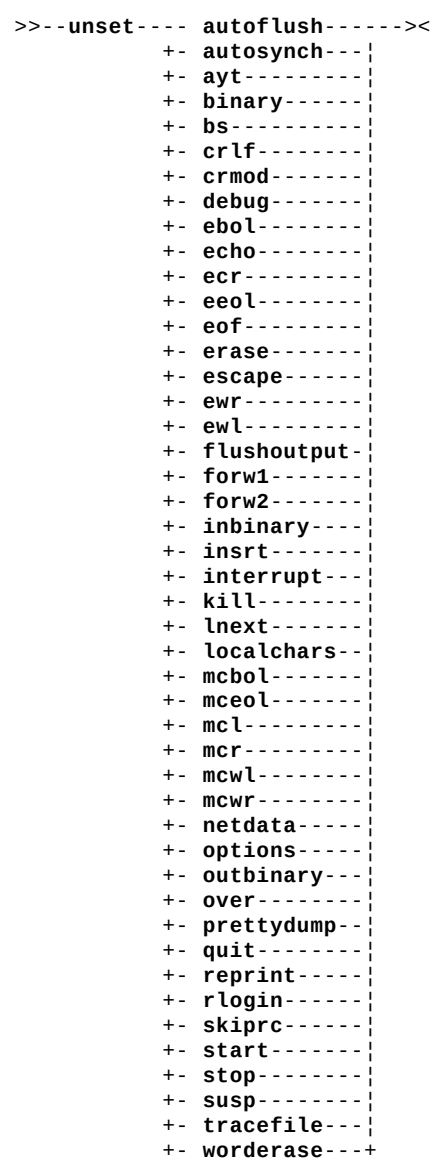
worderase Specifies the character to be used as the terminal's word erase character.

value Specifies the key sequence to use for the Telneto instruction.

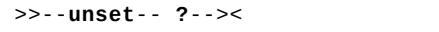
? Displays help information for the **set** subcommand.

Use the **unset** subcommand to disable a Telneto variable. The environment variables can be displayed with the **display** subcommand. The environment variables can be set or unset, but they cannot be toggled. However, any of the **toggle** subcommand variables can be set or unset using the **set** or **unset** subcommand.

The syntax of the **unset** subcommand is:



Displaying unset Help



autoflush Specifies that no deleting of output occurs when sending interrupt characters.

autosynch Specifies no automatic sending of interrupt characters in urgent mode.

ayt Cancels the key sequence defined to send a Telnet ayt.

binary Specifies that the sending and receiving of data is no longer binary.

bs Changes the backspace key from an ASCII delete to an ASCII backspace.

crlf Specifies that carriage returns are sent as <CR>.

crmod Ends the mapping of received carriage returns.

debug Ends debugging.

ebol Cancels the character defined as the erase-to-beginning-of-line character.

echo Cancels the key sequence defined to toggle between local echoing of entered characters (for normal processing) and suppressing echoing of entered characters (for example, a password) when in line mode.

ecr Cancels the character defined for deleting a character to the right.

eeol Cancels the character defined for deleting to end of the line.

eof Resets the eof sequence to ^Z.

erase Cancels the key sequence defined to cause an erase character to be sent.

escape Cancels the character defined as the escape character.

ewr Cancels the character defined for deleting a character to the left.

ewl Cancels the character defined for deleting a word to the left.

flushoutput
Cancels the key sequence defined to send a Telnet ao (abort output) sequence to the remote host, if Telneto is in localchars mode.

forw1, forw2
Cancels the key sequence defined to forward partial lines to the remote system.

inbinary Specifies that data being received is no longer binary.

insrt Cancels the character defined for entering insert mode.

interrupt Cancels the key sequence defined to send a interrupt character.

kill Cancels the key sequence defined to erase a line.

lnext Cancels the character defined as the terminal's next character.

localchars
Ends the local recognition of certain control characters.

mcbol Cancels the character defined for moving the cursor to the beginning of the line.

mceol Cancels the character defined for moving the cursor to the end of the line.

mcl Cancels the character defined for moving the cursor to the left one character.

mcr Cancels the character defined for moving the cursor to the right one character.

mcwl Cancels the character defined for moving the cursor to the left one word.

mcwr Cancels the character defined for moving the cursor to the right

one word.

netdata Specifies that printing of hexadecimal representation of network data (debugging) will not take place.

options Specifies that Telneto will no longer show options processing (debugging).

outbinary Specifies that data is no longer sent in binary.

over Cancels the key sequence defined for entering character replace mode.

prettydump
Specifies that Telneto will not print readable output for "netdata" (debugging).

quit Cancels the key sequence defined to set a break character.

reprint Cancels the character defined as the terminal's reprint character.

rlogin Cancels the character defined as the rlogin escape character.

skiprc Specifies that Telneto should not read the TELNET.RC file.

start Cancels the character defined as the terminal's start character if the toggle-flow-control option has been enabled.

stop Cancels the character defined as the terminal's stop character if the toggle-flow-control option has been enabled.

susp Cancels the key sequence defined to send a Telnet susp (suspend process) sequence to the remote host if Telneto is in localchars mode, or LINEMODE is enabled, and the suspend character is typed.

tracefile Specifies that the output caused by netdata, termdata, or option tracing is written to standard output.

worderase Cancels the character defined as the terminal's word erase character.

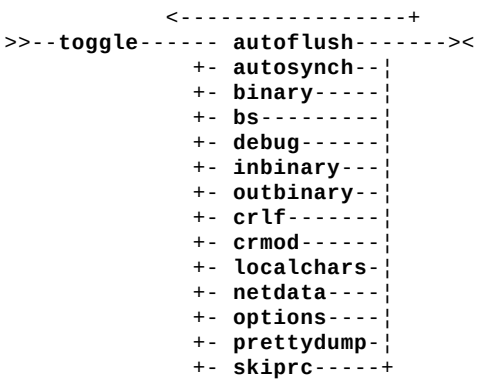
? Displays help information for the **unset** subcommand.

7.11.9 Toggling the Operating Parameters

Use the **toggle** subcommand to toggle (between **true** and **false**) various flags that can control how Telneto responds to events. You can use the **set** or **unset** subcommands to explicitly set these flags to **true** or **false**.

You can specify multiple arguments and display the state of these flags with the **display** subcommand.

The syntax of the **toggle** subcommand is:



Displaying toggle Help

>>--**toggle**-- **?**--><

autoflush If autoflush and localchars are both true when the ao (abort output) or quit characters are recognized and transformed into Telnet sequences, Telneto will not display any data on the user's workstation until the remote system acknowledges that it has processed those Telnet sequences. The initial setting is true.

autosynch If autosynch and localchars are both true when either the intr or quit characters are typed, the resulting Telnet sequence sent is followed by the Telnet synch sequence. This procedure causes the remote system to begin discarding all previously typed input until both of the Telnet sequences are read and acted upon. The initial setting is false.

binary Enables or disables the Telnet binary option on both input and output.

bs Changes the backspace key from ASCII backspace to ASCII delete.

debug Enables or disables debugging.

inbinary Enables or disables the Telnet binary option on input.

outbinary Enables or disables the Telnet binary option on output.

crlf If this is true, carriage returns (CR) are sent as carriage return line feed (CRLF). If this is false, carriage returns are sent as carriage return null (CRNUL). The initial setting is false.

crmod Toggles carriage return mode. When this mode is enabled, carriage return characters received from the remote host are mapped into a carriage return followed by a line feed. This mode affects only those characters received from the remote host. Use this mode when the remote host sends only carriage return, never line feed. The initial setting is false.

localchars
If this is true, the flush, interrupt, quit, erase, and kill characters are recognized locally and transformed into appropriate Telnet control sequences. When the linemode option is enabled, the value of localchars is ignored and assumed to be true. If linemode has been enabled, quit is sent as ends abnormally, and eof and susp are sent as eof and susp.

netdata Toggles the display of all network data in hexadecimal format. The initial setting is false.

options Toggles the display of some internal Telnet protocol processing. The initial setting is false.

prettydump
When the netdata toggle option is on, prettydump is true. The

output from the netdata and termdata options is formatted in a readable format. Spaces are put between the characters in the output, and the beginning of any Telnet escape sequence is preceded by an asterisk (*) to aid in locating it.

skiprc When the skiprc toggle is false, Telneto skips the reading of the TELNET.RC file in the ETC subdirectory when connections are opened. The initial setting is false.

? Displays help information for the **toggle** subcommand.

7.11.10 Setting or Changing the State of Special Characters

Use the **slc** (set local characters) subcommand to set or change the state of the special characters when the Telnet LINEMODE option is enabled.

Special characters are characters that get mapped to Telnet command sequences (for example, IP or quit) or line editing characters (for example, erase and kill). By default, the local special characters are exported.

The syntax of the **slc** subcommand is:

```
>>--slc---- ?--------><
          +- export-¦
          +- import-¦
          +- check--+
```

export Switches to the local defaults for the special characters. The local default characters are those of the local terminal at the time when Telnet is started.

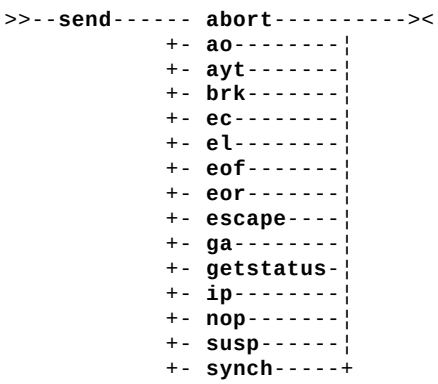
import Switches to the remote defaults for the special characters. The remote default characters are those of the remote system at the time when the Telnet connection is established.

check Verifies the current settings for the current special characters. The remote host is asked to send all the current special character settings, and if there are any discrepancies with the local host, the local host switches to the remote value.

? Displays help information for the **slc** subcommand.

7.11.11 Transmitting Special Characters

The **send** subcommand sends one or more special character sequences to the remote host. Multiple arguments can be specified. The syntax of this command is:



Displaying send Help

abort Sends the Telnet abort (end processes) sequence.

ao Sends the Telnet ao (abort output) sequence, which causes the remote system to move all output from the remote system to the user's workstation.

ayt Sends the Telnet ayt (are you there) sequence, to which the remote system can respond.

brk Sends the Telnet brk (break) sequence, which can have significance to the remote system.

ec Sends the Telnet ec (erase character) sequence, which causes the remote system to erase the last character entered.

el Sends the Telnet el (erase line) sequence, which causes the remote system to erase the line currently being entered.

eof Sends the Telnet eof (end of file) sequence.

eor Sends the Telnet eor (end of record) sequence.

escape Sends the current Telnet escape character. The initial setting of the escape character is the caret (^).

ga Sends the Telnet ga (go ahead) sequence.

getstatus Sends the subnegotiation to request that the server send its current option status if the remote host supports the Telneto **status** subcommand.

ip Sends the Telnet ip (interrupt process) sequence, which causes the remote system to end the current process abnormally.

nop Sends the Telnet nop (no operation) sequence.

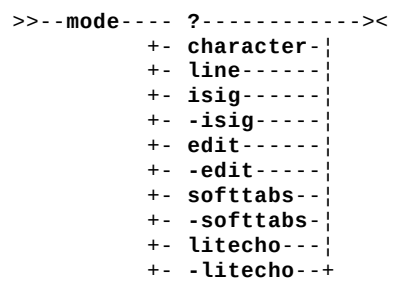
susp Sends the Telnet susp (suspend process) sequence.

synch Sends the Telnet synch sequence. This sequence causes the remote system to discard all previously typed (but not yet read) input. This sequence is sent as TCP urgent data and might not work if the remote system is a 4.2 BSD system. If it does not work, a lowercase r might be echoed on your workstation.

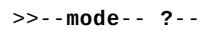
? Displays help information for the **send** subcommand.

7.11.12 Changing Mode Type

Depending on the state of the Telnet session, you can change the mode type by using the **mode** subcommand to designate the *type* of mode you want to enter. The remote host is asked for permission to go into the requested mode. If the remote host is capable of entering that mode, the permission to enter is granted.



Displaying mode Help



character Disables the Telnet LINEMODE option. If the remote host does not understand the LINEMODE option, enter the character-at-a-time mode.

line Enables the Telnet LINEMODE option.

isig Enables the TRAPSIG mode of the LINEMODE option.

-isig Disables the TRAPSIG mode of the LINEMODE option.

edit Enables the EDIT mode of the LINEMODE option.

-edit Disables the EDIT mode of the LINEMODE option.

softtabs Enables the SOFT_TAB mode of the LINEMODE option.

-softtabs Disables the SOFT_TAB mode of the LINEMODE option.

litecho Enables the LIT_ECHO mode of the LINEMODE option.

-litecho Disables the LIT_ECHO mode of the LINEMODE option.

? Displays help information for the **mode** subcommand.

7.11.13 Closing the Current Connection

You can close the current connection by using either the **logout** or **close** subcommands.

Both of these subcommands disconnect you from any Telnet server to which you are connected and return you to the command prompt.

>>--**logout**--><

>>--**close**--><

7.11.14 Ending the Session

Use the **quit** subcommand to close any open Telnet session and exit Telnet.

```
>>--quit--><
```

7.12 The TELNET.RC File

The TELNET.RC file resides in the ETC subdirectory. It is a text file used to specify commands to the Telneto line-mode client that you would normally have to enter at the command prompt. The TELNET.RC file contains command definitions used when connecting to a particular host. These commands are entered automatically for you, eliminating the need for you to enter the commands at the client command prompt.

Each TELNET.RC file section is identified by a line that begins with the name of the host with which it is connecting. The rest of the line, and successive lines in a particular section that begin with blanks, are assumed to be Telnet line-mode client commands. These commands are processed as if they had been entered manually at the Telnet command prompt. The last line of each section must be blank to signify the end of that section.

The following format rules apply:

- Lines that begin with a pound sign (#) are comment lines.
- A blank line signifies the end of a host entry section.
- Lines that begin without blanks start a new host entry section.
- For any given host, all matching host entry sections are executed.
- The host name DEFAULT matches for all hosts.

The following is an example of the TELNET.RC file:

```
#                                                                  #
# This is a DEFAULT entry that is executed                         #
# for all clients that you connect to.                             #
#                                                                  #
DEFAULT         send ayt
                toggle local
                set esc ^[
                set interrupt ^c

#                                                                  #
# This is a host specific entry that is executed                   #
# only when telneting to the host ziggy.tcpip.ibm.com              #
#                                                                  #
ziggy.tcpip.ibm.com             toggle bs
                                set esc ^t
                                toggle wrap
                                toggle crmod
```

7.13 Understanding Keyboard Mappings

For the normal keys (those defined in 7-bit ASCII), the keystroke is transmitted to the server as its ASCII definition. For keys outside of this definition, such as function keys, it is necessary to transmit a series of characters that uniquely identify that key to the server. These series of characters are known as escape sequences (the ASCII escape code followed by a series of other ASCII codes).

Subtopics
7.13.1 ASCII Telnet Emulator Keyboard Mapping

7.13.1 ASCII Telnet Emulator Keyboard Mapping

Most of the ASCII Telnet emulator keys send the value indicated on the keytop. However, there is some default keyboard mapping for these terminals, as shown in the following tables.

Workstation Key	**Hex Escape Code**	**Function**
F1	1b4f50	PF1
F2	1b4f51	PF2
F3	1b4f52	PF3
F4	1b4f53	PF4
F5	1b4f54	
F6	1b5b31377e	F6
F7	1b5b31387e	F7
F8	1b5b31397e	F8
F9	1b5b32307e	F9
F10	1b5b32317e	F10
F11	1b5b33357e	
F12	1b5b33367e	
Shift-F1	1b5b32337e	F11
Shift-F2	1b5b32347e	F12
Shift-F3	1b5b32357e	F13
Shift-F4	1b5b32367e	F14
Shift-F5	1b5b32387e	F15
Shift-F6	1b5b32397e	F16
Shift-F7	1b5b33317e	F17
Shift-F8	1b5b33327e	F18
Shift-F9	1b5b33337e	F19
Shift-F10	1b5b33347e	F20
Shift-F11	1b5b33377e	
Shift-F12	1b5b33387e	
Control-F1	1b4f61	
Control-F2	1b4f62	
Control-F3	1b4f63	
Control-F4	1b4f64	
Control-F5	1b4f65	
Control-F6	1b4f66	
Control-F7	1b4f67	
Control-F8	1b4f68	
Control-F9	1b4f69	
Control-F10	1b4f6a	
Control-F11	1b5b33397e	
Control-F12	1b5b34307e	
Alt-F1	1b4f4b	
Alt-F2	1b4f4c	

Alt-F3	1b4f4f	
Alt-F4	1b4f4e	
Alt-F5	1b4f45	
Alt-F6	1b4f46	
Alt-F7	1b4f47	
Alt-F8	1b4f48	
Alt-F9	1b4f49	
Alt-F10	1b4f4a	
Alt-F11	1b5b34317e	
Alt-F12	1b5b34327e	
Insert	1b5b327e	insert
Home	1b5b317e	Find
Pageup	1b5b357e	Previous Screen
Delete	1b5b337e	Remove
End	1b5b347e	Select
Pagedown	1b5b367e	Next Screen
Up	1b5b41	Up
Left	1b5b44	Left
Down	1b5b42	Down
Right	1b5b43	Right
Tab	09	Tab
Stab	1b5b35307e1b5045	
Backspace	7f or 8f	Delete or Backspace
Esc	1b	

By default, the numeric keypad keys send the value indicated on the keytop. However, if keypad application mode is active (for example, the host has sent Esc =), the numeric keypad keys send the following.

Workstation Key	**Hex Escape Code**	**Function**
Keypad0	1b4f70	VT220 Keypad0
Keypad1	1b4f71	VT220 Keypad1
Keypad2	1b4f72	VT220 Keypad2
Keypad3	1b4f73	VT220 Keypad3
Keypad4	1b4f74	VT220 Keypad4
Keypad5	1b4f75	VT220 Keypad5
Keypad6	1b4f76	VT220 Keypad6
Keypad7	1b4f77	VT220 Keypad7
Keypad8	1b4f78	VT220 Keypad8
Keypad9	1b4f79	VT220 Keypad9
Keypad.	1b4f6e	VT220 Keypad.
Keypad-	1b4f6d	VT220 Keypad-
Keypad+	1b4f6c	VT220 Keypad+
KeypadEnter	1b4f4d	VT220 KeypadEnter

8.0 Chapter 8. Using 3270-Based and 5250-Based Telnet Clients

This chapter describes how to log on to a remote host using a 3270-based Telnet client, and how to log on to an AS/400* using the 5250-based Telnet client, TN5250.

TCP/IP for OS/2 provides two 3270-based Telnet clients from which to choose: PMANT and TN3270. PMANT is an OS/2 Presentation Manager application, whereas TN3270 is an OS/2 character-based, full-screen application. TN3270 is recommended for use with a serial line because it is somewhat faster than PMANT, though PMANT offers more function.

Table 3 lists available functions.

Table 3. Summary of Telnet Client Functions

Function	**PMANT**	**TN3270**	**TN5250**
Select session type			
Presentation Manager	Yes	Yes	Yes
OS/2 window		Yes	Yes
OS/2 full screen			
Customize mouse actions	Yes	No	No
Customize screen size	Yes	Yes	Yes
Customize title bar and task list	Yes	No	No
Edit files			
Mark	Yes	Yes	Yes
Copy to clipboard	Yes	Yes	Yes
Paste	No	Yes	Yes
Enable square brackets	Yes	Yes	Yes
Enable extended data stream	Yes	Yes	Yes
Generate a log file for debugging purposes	Yes	No	No
Ignore Erase/Write requests to change default screen size	Yes	No	No
Maintain screen size	Yes	No	No
Remap the keyboard	Yes	Yes	Yes
Specify a translation table	Yes	Yes	Yes
Specify a nonblinking cursor	Yes	No	No
Specify a block cursor	Yes	Yes	Yes
Supress the bell	Yes	Yes	Yes
Use boxes for characters in hidden fields	No	Yes	Yes
Use old-style keyboard (Shift + PF1 = PF13)	No	Yes	Yes

Note: These functions apply to the standard PS/2 keyboard. If an OS/2 key is equivalent to a 3270 key function, and it is not listed above, the OS/2 key defaults to that function. or example, the OS/2 key insert defaults to the 3270 function insert.

For more information about:

Function	**Topic**
PMANT	"Logging On to a Foreign Host Using PMANT" in topic 8.4.
TN3270	"Logging On to a Foreign Host Using TN3270" in topic 8.5.
TN5250	"Logging On to a Foreign Host Using TN5250" in topic 8.6.

Subtopics

8.1 Default 3270 Emulator Keyboard Mappings

Table 4 lists the default keyboard mappings for 3270 telnet clients.

Table 4. Default 3270 Emulator Keyboard Mappings

Workstation Key	**3270 Function**
Alt-1	PA1
Alt-2	PA2
Alt-3	PA3
Backspace	Backspace
Control-End	Erase-EOF
Control-Delete	Kill-Word
Control-F1	PA1
Control-F2	PA2
Control-F3	PA3
Control-Left	Move backward a word
Control-Right	Move forward a word
Delete	Delete
Down Arrow	Down
End	End
Enter	Enter
Escape	Clear
F1	PF1
F2	PF2
F3	PF3
F4	PF4
F5	PF5
F6	PF6
F7	PF7
F8	PF8
F9	PF9
F10	PF10
F11	PF11
F12	PF12
Home	Home
Insert	Insert
Left Arrow	Left
LeftAlt (PMANT only)	Gives focus to the menu bar
NewLine	Enter
PageDown	PF8
PageUp	PF7
Right Arrow	Right
RightAlt (PMANT only)	Gives focus to the menu bar
RightControl	Enter

ScrollLock (PMANT only)	PA3
Shift-End	Erase-EOF
Shift-Enter	NewLine
Shift-F1	PF13
Shift-F2	PF14
Shift-F3	PF15
Shift-F4	PF16
Shift-F5	PF17
Shift-F6	PF18
Shift-F7	PF19
Shift-F8	PF20
Shift-F9	PF21
Shift-F10	PF22
Shift-F11	PF23
Shift-F12	PF24
Shift-NewLine	NewLine
Shift-Tab	Backtab
Tab	Tab
Up Arrow	Up
Note: LeftAlt, RightAlt, and ScrollLock are PMANT defaults only. They are not valid for TN3270.	

If an OS/2 key is equivalent to a 3270 key function, and it is not listed above, the OS/2 key defaults to that function. For example, the OS/2 key insert defaults to the 3270 function insert.

8.2 Translation Table Support for PMANT and TN3270

PMANT and TN3270 now support external ASCII/EBCDIC translation tables. This support is provided by the following files:

\TCPIP\BIN\TN3270.EXE
\TCPIP\BIN\PMANT.EXE
\TCPIP\ETC\3278XLT.SAM

Both PMANT and TN3270 have the following syntax for specifying a translation table parameter:

-tx translate_table_file_name

If you specify a translation table file name but do not include its full path statement, the application searches for the file in the ETC subdirectory. If you do not specify the **-tx** option when starting a session, the application uses the file 3278XLT.SAM, located in the ETC subdirectory.

The 3278XLT.SAM file is shipped as a sample and contains the US ASCII/EBCDIC translations. If you want to use a different translation table file on a regular basis, copy that file and rename it to 3278XLT.TBL in your ETC subdirectory. If the file 3278XLT.TBL does not exist, PMANT and TN3270 use the default US translation table.

8.3 3270-Based Telnet Clients and the OS/2 Workplace Shell

The TCP/IP for OS/2 Telnet clients are integrated with the OS/2 desktop. You can use the Telnet template to create icons that represent Telnet sessions without having to learn or use the Telnet commands or parameters.

The OS/2 desktop includes a Templates folder that is located in the OS/2 System folder, unless it has been moved. When TCP/IP for OS/2 is installed, it adds a 3270 Telnet template to the Templates folder. If you do not find the 3270 Telnet template in your Templates folder, type **tcpdsktp** at the command line of an OS/2 session and the template will be created.

To create an icon for a 3270 Telnet session, follow these steps:

1. Select the 3270 template, then press and hold mouse button 2.

2. Drag the 3270 Telnet template to your desktop, then release the mouse button. The 3270 Telnet settings notebook appears.

Using the 3270 Telnet settings notebook, you can specify a title, an icon, and other definitions for the session. The settings notebook also allows you to specify the name of the remote host, the port to use, the screen dimensions of your emulator window, and other options governing the session. All of the options in the 3270 Telnet settings notebook are optional, and a 3270 Telnet session will work without your having to change anything in the notebook. In most cases, however, you will want to specify at least the name of the remote host to which you want to connect.

Like other OS/2 settings notebooks, the 3270 Telnet settings notebook also includes a Window page and a General page. On these pages, you specify the window minimize behavior, the icon title, and the icon used for the 3270 Telnet session.

On the 3270 Telnet page, shown in Figure 32, you specify the host, port, and translation fields.

PICTURE 32

Figure 32. 3270 Telnet Page

In the **Host** field, type the name of the remote host to which you want to connect. You can specify the host name, IP address, or an alias. If you leave this field blank, you are prompted to type the host name when you start the 3270 Telnet session.

The **Port** field enables you to connect to a specific port. If you leave this field blank, the default Telnet port is used. You can type a port number or a port name. If you type a port name, the port number corresponding to that name, as specified in the SERVICES file in the ETC subdirectory, is used.

Figure 33 shows the Session page of the 3270 Telnet settings Notebook.

PICTURE 33

Figure 33. 3270 Session Page

Use the Session page to choose whether your 3270 session is run in a full-screen session, an OS/2 window, or a Presentation Manager window. The full screen version is fastest, but the Presentation Manager version is easiest to use. You can also specify the number of text rows and columns in the session. Some hosts may not use all of the available display area.

You can use the F1 key anytime to get help in the settings notebook. After completing the settings for your 3270 Telnet session, double-click on the notebook's system menu icon to close the 3270 Telnet settings notebook. You can then use this 3270 Telnet icon to start the session anytime by double-clicking on it.

Notes:

1. For a concise description of commands, subcommands, and parameters, see the *IBM TCP/IP Version 2.0 for OS/2: Command Reference*.

2. For information about using Telnet with the Network Security Program (NetSP), see the *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration*.

8.4 Logging On to a Foreign Host Using PMANT

To use the **pmant** command, a Telnet server must be running on a foreign host that supports a 3270 terminal emulator.

Subtopics

8.4.1 Starting a PMANT Session

Start the PMANT session from an OS/2 command prompt by issuing the **pmant** command. You must then specify the foreign host to which you are connecting.

The syntax of the **pmant** command is:

```
                                +- -rows 24-+  +- -cols 80-+
>>--pmant-----------------------+-----------+--+-----------+------------>
           +- host-+  +- -ext-+  +- -rows n--+  +- -cols n--+

>-------------------------------------------------------------------------->
   +- -nb-+  +- -b-+  +- -ew-+  +- -t text-+  +- -tx filename-+

                        +- -p 23----------+
>-----------------------+-----------------+-----------><
   +- -ncb-+  +- -bc-+  +- -p port_number-+  +- -d-+
```

Displaying pmant Help

```
>>--pmant-- -?--><
```

host Specifies the host to which you are connecting. PMANT immediately tries to establish a connection to the remote host. If you do not specify a host, PMANT displays a window that prompts you to enter the host.

-ext Starts support for extended colors and nonstandard screen sizes. Extended highlighting (reverse, blink, and underline) is supported.

-rows **n** Specifies the screen dimensions in number of rows (24-80). The default is 24.

-cols **n** Specifies the screen dimensions in number of columns (80-150). The default is 80.

Note: If -rows and -cols are specified and -ext is not specified, the resulting screen size is a standard screen size equal to or smaller than the size requested. The following are the four standard terminal screen sizes, specified in rows by columns:

```
24 x 80
32 x 80
43 x 80
27 x 132
```

-nb Suppresses the bell.

-b Changes the output translation table for code points X'AD' and X'BD' to display them as square brackets. On a 3179 terminal with the text feature, the square brackets are at code points X'BA' and X'BB'. However, many programs display the brackets using X'AD' and X'BD', because that is the EBCDIC definition.

-ew Ignores Erase/Write requests to change to the default display size (24 X 80).

-t **text** Specifies text to be displayed in the title bar and task list. When -t **text** is not used, PMANT displays the specified host in the title bar and task list. When -t is used, the text specified precedes the host. For example, if you issue the **pmant** command like this:

```
pmant host1 -t sometext
```

then **sometext: host1** is displayed in the title bar and task list. No spaces are allowed in the text.

-tx **filename** Reads the specified translation table. If the file name you specified does not contain the full path, PMANT searches the TCPIP\ETC directory for the file.

If you do not specify -tx, PMANT uses the translation file 3278XLT.SAM in the TCPIP\ETC directory. If this file does not exist in the TCPIP\ETC directory, PMANT uses the default US translate table provided with the application.

-ncb Specifies a nonblinking cursor.

Note: Even if you do not specify -ncb, the cursor will still blink on screens where there is a blinking field.

-bc Specifies a block cursor.

-p **port_number** Specifies the port number to be used. The default port is 23. Port 1023 is recommended if you are using servers that send ASCII prior to switching to transparent mode.

-d Generates a PMANT.LOG file containing debug information for terminal negotiation and all data sent to and received from the server.

-? Displays help information.

Note: On screens with many extended attributes, the screen is painted without extended attributes before the extended attributes can be painted.

The 3270 architecture provides for a terminal having two screen sizes: the default and the alternate. PMANT defines the default screen size to be 24 x 80 and the alternate to be what was defined by the -rows and -cols parameters. Most 3270 programs can use screen sizes other than 24 x 80 and notify PMANT of this by sending an Erase/Write Alternate command. 3270 programs that require a 24 x 80 screen inform PMANT by sending an Erase/Write command. When this occurs, PMANT changes to a 24 x 80 screen size. To prevent this behavior, specify the -ew parameter when starting PMANT.

If you do not specify a host when you issue the **pmant** command, PMANT prompts you for a hostname.

If you select Enter without entering a host, you are returned to the OS/2 command prompt.

If you specify an invalid host, or if an active Telnet server is not running on the host you specified, TCP/IP displays the Unknown Host message window. If this message is displayed, use the Enter key to cancel the process and clear the message.

Establishing a PMANT session can take several seconds.

If the connection takes more than two minutes to be established, a problem may exist. You can select **Cancel Connection** and verify that the destination host is operating correctly.

When a connection is established, the PMANT window is displayed. Figure 34 is an example of the PMANT window that is displayed when you connect to a VM host.

PICTURE 34

Figure 34. Example of PMANT VM Host Window

When you use PMANT, some of your keys function differently from the way they do in your regular OS/2 environment. For example, the F10 key functions as a PF key, which means that pressing the F10 key no longer selects the menu bar.

To select the PMANT menu bar without a mouse, press the Shift key and Esc key simultaneously. The System menu is displayed; you can then use the arrow keys on your keyboard to select items from the menu bar.

8.4.2 Using the PMANT menus

Select the PMANT pull-down menus from the menu bar displayed in the PMANT window.

The pull-down menus for PMANT are:

Menu	**Description**
Mark	Allows you to copy to a clipboard or unmark a selected area
Configuration	Customizes colors, mouse action definitions, and keyboard mapping
Fonts	Allows you to select any of the VIO window fonts available on your display
Exit	Exits PMANT

8.4.3 Marking a Portion of the Screen on the Foreign Host

Select the **Mark** pull-down menu choice to copy a marked portion of your host screen to the clipboard or to remove the marking from a marked section.

Subtopics
8.4.3.1 Copy to Clipboard
8.4.3.2 Unmark

8.4.3.1 Copy to Clipboard

The **Copy to Clipboard** menu item allows you to copy rectangular portions of the screen to the clipboard. The copy action is accomplished by marking the area and copying it to the clipboard.

To mark and copy a portion of the screen, do the following:

1. Move the mouse pointer to any part of the screen you want marked, then press and hold down the mouse button.

   Drag the mouse over the area you want to select and release the mouse button. The marked area is colored to delimit the area selected. The color used to delimit the marked area can be changed using the color menu item on the Configuration pull-down menu.

2. Select the **Copy to Clipboard** menu item from the Mark pull-down menu to copy the marked area to the clipboard.

Note: Although the copy function is supported, paste function is not supported.

8.4.3.2 Unmark

The **Unmark** menu item allows you to remove the marking from a marked area of the screen.

8.4.4 Configuring PMANT

Select the **Configuration** pull-down menu to configure PMANT for use with the foreign host.

Subtopics
8.4.4.1 Color
8.4.4.2 Mouse
8.4.4.3 Remapping the Keyboard

8.4.4.1 Color

Select the **Color** menu item from the Configuration pull-down menu to customize the PMANT screen colors.

Note: To enable the display to support all of the color fields, you must start PMANT with the -ext parameter, which indicates that you want the terminal emulation of PMANT to support extended color attributes. If you do not specify this parameter, extended colors received from the host default to your basic color palette.

When you select the **Color** menu item, the Color Selection is displayed. The Color Selection Window consists of the following:

- The 3270 terminal fields that can be changed are listed under the 3270 Terminal Fields heading.
- The potential color selections for the foreground and background are listed under the Fore and Back headings.

The following steps describe how to change a color:

1. Select the component to be changed from the 3270 Terminal Fields list. The current color of the selected 3270 terminal field is highlighted in the Fore and Back color lists.
2. Select the preferred color combination from the two color lists with the mouse or the arrow keys. Double clicking on a color or pressing the Enter key causes the selected field component to be temporarily colored the selected color.

The pushbuttons displayed at the bottom of the window are used for the following:

Pushbutton	**Description**
Save	Applies the changes specified in the Color Selection window to the PMANT window and saves the color assignments to be automatically used the next time PMANT is entered
Apply	Applies the changes specified in the Color Selection window to the PMANT window
Cancel	Restores the PMANT window colors to the values that were active before the Color Selection window was created

8.4.4.2 Mouse

Select the **Mouse** menu item to view or change the current mouse action definitions. The Bind Action to Mouse Event window displays the current definitions of specific mouse buttons as they relate to specific function keys.

On this window, you can assign certain actions when mouse buttons are clicked. The actions are:

Item	**Action**
Def	Moves the cursor to the location of the click.
F**n**	Moves the cursor to the location of the click, and invokes the PF keystroke that is specified. **n** represents the PF key numbers 1 through 12. For example, F1 invokes the PF1 keystroke, F2 invokes the PF2 keystroke, F3 invokes the PF3 keystroke, and so on, through F12.
Enter	Moves the cursor to the location of the click and invokes the Newline keystroke. The default key mapping for the Newline keystroke is the 3270 enter function.

A list of mouse events (items) used to initiate specific actions appears in the window. The action that applies to a mouse event appears adjacent to each event. The up and down arrow pushbuttons are used to change the definitions.

The three pushbuttons at the bottom of the window are:

Pushbutton	**Description**
Save	Applies the mouse definitions that were specified in the Mouse Event window to the PMANT terminal session, and saves the definitions to be automatically used the next time PMANT is invoked
Apply	Applies the mouse definitions that were specified in the Mouse Event window to the PMANT terminal session
Cancel	Restores the definitions to the values that were in effect before the Mouse Event window was created

To change the mouse button action definitions:

1. Using the mouse, click on the specific arrow for a desired change.
2. Using the mouse, click on the appropriate pushbutton at the bottom of the window.

8.4.4.3 Remapping the Keyboard

Use the **Refresh Key Mapping** with the PMANT.KEY file to edit the default key bindings that are activated when you start the PMANT program. Create the PMANT.KEY file using any text editor.

PMANT.KEY is a flat ASCII file that is loaded when you start the PMANT program. You use it to change default key definitions.

When PMANT is started, it first looks for PMANT.KEY in the current directory. If this fails, PMANT tries to open the file in the directory defined in the ETC environment variable. Since PMANT is started with default key bindings, PMANT.KEY need contain only those definitions that override the defaults or that provide definitions not included with the defaults.

The PMANT.KEY file is line-oriented, which means that each of its lines is either blank, contains a comment, or contains a definition. The semicolon is used in the file to delimit comments, and comments may appear anywhere on a line. A comment must begin with a semicolon and continue until the end of a line.

A line containing a binding follows this syntax:

os2_key whitespace 3270_function ; optional_comment

where the **os2_key** variable is one of the keys:

- F1 through F12
- Control-F1 through Control-F12
- Alt-F1 through Alt-F12
- Shift-F1 through Shift-F12
- Control-'unshifted_keypress'
- Alt-'unshifted_keypress'

and the **unshifted_keypress** variable is one of the keyboard's white keys, such as the letter keys, punctuation-mark keys, number keys, and foreign-language accent keys and must be enclosed in single quotes. The **unshifted_keypress** variable represents the unshifted position for that key.

For example, to bind a function to the control-'4' key combination, the **unshifted_keypress** variable is 4. It is impossible to bind control-'$', since $ is the shifted state of 4.

The **function key** variable is a keyword identifying an OS/2 function key. Possible values of the **function key** variable are:

Backspace	Newline (Enter key on alpha keypad
BackTab (same as Shift+Tab)	PgUp
Delete	PgDn
Down	Pause
End	PrtScr
Enter (Enter key on numeric keypad)	Right
Esc	Right-Alt
F1-F12	Right-Control
Home	ScrLock
Insert	Space
Left	Tab
Left-Alt	Up
Left-Control	

The **3270 function** variable specifies a 3270 function or a TN3270 application-specific function to which you can bind an OS/2 key. Possible values of the **3270 function** are:

Attn	Forward-Word
Backspace	Home
BackTab	Insert
Backward-Word	Kill-Word
Backward-Kill-Word	Left
Cent	Newline
Cursor-Move	PA1
Clear	PA2
Delete	PA3
Down	PF1-PF24
Dumpfields	Reset
Dup	Right
End	String
Enter	Sysreq
Erase-EOF	Tab

Erase-Input	Unlock
Field-Mark	Up

A sample PMANT.KEY file containing definitions looks like this:

```
  ; This is a sample PMANT.KEY file.  It should be located either
  ; in the current directory or in the directory pointed by the
  ; environment variable ETC.


  escape           clear  ; bind escape to clear (the default)
  control-'a'      home
  newline          newline; make the keyboard enter create a new line
  right-control    enter  ; make the right control key execute commands
  enter            enter  ; make numeric keypad enter execute commands
  shift-backspace  pf3    ; shift backspace invokes the pf3 function key
  backtab          tab    ; make shift-tab go to next field
```

Note: You do not have to use the PMANT.KEY file if you use the default keyboard definitions (see Table 4 in topic 8.1).

Some keys cannot be remapped at all. The following accelerator keys cannot be remapped:

Workstation Key	Function
Alt-F4	Close
Alt-F5	Restore
Alt-F7	Move
Alt-F8	Size
Alt-F9	Minimize
Alt-F10	Maximize
Shift-escape	Menu Bar
Ctrl-escape	Window List

8.4.5 Selecting Fonts

Use the Fonts pull-down menu to specify the font used in the PMANT window. The following is an example of the pull-down menu that is displayed as a result of selecting Fonts. The menu items displayed depend on the display adapter you are using.

The menu items describe the fonts (width ¦ height in pixels) that are available on your display adapter. The current font being used is checked. To change the font, select the desired font.

8.4.6 Logging Off PMANT

You should log off your session with the foreign host to exit PMANT. You can also use the Exit pull-down menu to leave PMANT.

Select **Resume** if you want to return to PMANT. Select **Exit PMANT** if you want to exit. Select **Yes** if you want to end the program.

8.5 Logging On to a Foreign Host Using TN3270

To use the **tn3270** command, a Telnet server must be running on a foreign host that supports a 3270 terminal emulator.

Subtopics

8.5.1 Starting a TN3270 Session

The format of the **tn3270** command follows:

```
>>--tn3270-------------------------------------------------------------->
            +- -b-+  +- -ext-+  +- -nb-+  +- -h-+  +- -o-+

                      +- -p 23----------+
>---------------------+-----------------+------------ hostname--><
   +- -tx filename-+  +- -p port_number-+  +- -bc-+
```

Displaying TN3270 Help

```
>>--tn3270-- -?--><
```

-b Displays brackets and backslash characters as they appear on a keyboard.

-ext Enables extended data stream support.

-nb Suppresses the bell.

-h Uses boxes for characters in hidden fields.

-o Uses old style keyboard (for example, Shift + F1 = PF13).

-tx **filename**
Reads the specified translation table. If the file name you specified does not contain the full path, TN3270 searches the TCPIP\ETC directory for the file.

If you do not specify -tx, TN3270 uses the translation file 3278XLT.TBL in the TCPIP\ETC directory. If this file does not exist in the TCPIP\ETC directory, TN3270 uses the default US translate table provided with the application.

-p **port_number**
Specifies the port number to be used. The default port is 23. Port 1023 is recommended if you are using servers that send ASCII prior to switching to transparent mode.

-bc Specifies a block cursor.

hostname
Specifies the host to which you are connecting.

8.5.2 Using the TN3270 Menus

TN3270 provides menus for modifying the keyboard and attribute mappings. Keyboard mapping allows you to define 3270 functions to specific key strokes. Attribute mapping allows you to change the colors of 3270 field types.

To get to the Main Menu, press the Alt-M keys. When the Main Menu is displayed, make your selection by pressing the Enter key.

When using the menus:

- Use the up and down arrow keys to scroll through the menu items. If you reach the bottom of the list and there are more items, then the next time you press the down arrow key, the list scrolls up one element. If you are at the top of the list and there are more items, pressing the up arrow key causes the list to scroll down one element.

- To select a menu item, press the Enter key.

- To go back to the previous menu without making any changes, press the Esc key.

  Note: The Esc key cannot cancel a change that you have already made.

- To return to the TN3270 session, press the Esc key from the Main Menu.

8.5.3 Modifying Screen Colors

To modify screen colors:

1. Select **Change Color** from the Main Menu. The **Change Color** menu is displayed.

2. Select the field type you want to modify. The Color menu appears, showing the current value of the selected field type.

3. To change the current mapping, select a different color.

4. Press Esc to return to the **Change Color** menu.

5. Select the **Save Color** menu item.

8.5.4 Remapping the Keyboard

To modify the 3270 key mappings that are activated when you start the TN3270 program, edit the file TN3270.KEY, using any text editor.

TN3270.KEY is a flat ASCII file that is loaded when you start the TN3270 program. You use it to change default key definitions.

When TN3270 is started, it first looks for TN3270.KEY in the current directory. If this fails, TN3270 tries to open the file in the directory defined in the ETC environment variable. Since TN3270 is started with default key bindings, TN3270.KEY need contain only those definitions that override the defaults or that provide definitions not included with the defaults.

The TN3270.KEY file is line-oriented, which means that each of its lines is either blank, contains a comment, or contains a definition. The semicolon is used in the file to delimit comments, and comments can appear anywhere on a line. A comment must begin with a semicolon and will continue until the end of a line.

A line containing a binding follows this syntax:

os2_key whitespace 3270_function optional_comment;

where the **os2_key** variable is one of these keys:

- F1 through F12
- Control-F1 through Control-F12
- Alt-F1 through Alt-F12
- Shift-F1 through Shift-F12
- Control-'unshifted_keypress'
- Alt-'unshifted_keypress'

and the **unshifted_keypress** variable is one of the keyboard's white keys, such as the letter keys, punctuation mark keys, number keys, and foreign language accent keys and must be enclosed in single quotes. The **unshifted_keypress** variable represents the unshifted position of that key.

For example, to bind a function to the control-'2' key combination, the **unshifted_keypress** variable is 2. It is impossible to bind the control-'@' combination, because @ is the shifted state of 2. The following key combinations cannot be bound at all:

Alt-home	Control-'-'
Control-'1'	Control-'='
Control-'3'	Control-';'
Control-'4'	Control-'\'
Control-'5'	Control-'/'
Control-'7'	Control-'.'
Control-'8'	Control-','
Control-'9'	Control-'`'
Control-'0'	

Note: TN3270 does not allow binding to the quotation-mark key, although PMANT does. Therefore, the syntax of the TN3270.KEY file must allow this binding. To bind to the quotation-mark character, the backslash is used as an "escape" to temporarily break the syntax that an unshifted keypress is surrounded by quotation marks. Since the backslash is interpreted as an escape, there must be two backslashes when binding to the backslash. For example:

```
  control-'\''     ; escape the quote so it can be bound to
                   ; ''' would be illegal syntax
  control-'\\'     ; escape the backlash so it can be bound to
                   ; '\' would be illegal syntax

  control-'\a'     ; '\a' is exactly the same as 'a'
```

The **function key** variable is a keyword identifying an OS/2 function key. Possible values of the **function key** variable are:

Backspace	Newline (Enter key on alpha keypad
BackTab (same as Shift+Tab)	PgUp
Delete	PgDn
Down	Pause
End	PrtScr
Enter (Enter key on numeric keypad)	Right
Esc	Right-Alt
F1-F12	Right-Control
Home	ScrLock
Insert	Space

Left	Tab
Left-Alt	Up
Left-Control	

The **3270 function** variable specifies a 3270 function or a TN3270 application-specific function to which you can bind an OS/2 keypress. Possible values of the **3270 function** are:

Attn	Forward-Word
Backspace	Home
BackTab	Insert
Backward-Kill-Word	Kill-Word
Backward-Word	Left
Cent	Menu
Clear	Newline
Cursor-Move	PA1
Up	PA2
Delete	PA3
Down	PF1-PF24
Dumpfields	Redraw
Dup	Reset
End	Right
Enter	String
Erase-EOF	Sysreq
Erase-INPUT	Tab
Field-Mark	Unlock

A sample TN3270.KEY file containing definitions looks like this:

```
  ; This is a sample TN3270.KEY file.  It should be located either
  ; in the current directory or in the directory pointed by the
  ; environment variable ETC.


  escape           clear  ; bind escape to clear (the default)
  control-'a'      home
  newline          newline; make the keyboard enter create a new line
  right-control    enter  ; make the right control key execute commands
  enter            enter  ; make numeric keypad enter execute commands
  shift-backspace  pf3    ; shift backspace invokes the pf3 function key
  backtab          tab    ; make shift-tab go to next field
```

Note: You do not have to use the TN3270T.KEY file if you use the default keyboard definitions (see Table 4 in topic 8.1).

8.6 Logging On to a Foreign Host Using TN5250

To use the **tn5250** command, a Telnet server must be running on a foreign host that supports a 5250 terminal emulator.

Subtopics

8.6.1 Starting a TN5250 Session

Start the TN5250 session from an OS/2 command prompt by issuing the **tn5250** command. You must then specify the foreign host to which you are connecting.

```
>>--tn5250-------------------------------------------------------------->
           +- host-+  +- -b-+  +- -bc-+  +- -d-+  +- -nb-+  +- -ncb-+

   +- -p 23----------+
>--+-----------------+------------------------------------------------><
   +- -p port_number-+  +- -t text-+  +- -tx filename-+  +- -wide-+
```

Displaying TN5250 Help

```
>>--tn5250-- -?--><
```

host Specifies the host to which you are connecting. TN5250 immediately tries to establish a connection to the remote host. If you do not specify a host, TN5250 displays a window that prompts you to enter the host.

-b Changes the output translation table for code points X'AD' and X'BD' to display them as square brackets. On a 3179 terminal with the text feature, the square brackets are at code points X'BA' and X'BB'. However, many programs display the brackets using X'AD' and X'BD', because that is the EBCDIC definition.

-bc Specifies a block cursor.

-d Generates a TN5250.LOG file containing debug information for terminal negotiation and all data sent to and received from the server.

-nb Suppresses the bell.

-ncb Specifies a nonblinking cursor.

Note: Even if you do not specify -ncb, the cursor will still blink on screens where there is a blinking field.

-p **port_number** Specifies the port number to be used. The default port is 23. Port 1023 is recommended if you are using servers that send ASCII prior to switching to transparent mode.

-t **text** Specifies text to be displayed in the title bar and task list. When -t **text** is not used, TN5250 displays the specified host in the title bar and task list. When -t is used, the text specified precedes the host. For example, if you issue the following **tn5250** command:

```
tn5250 host1 -t sometext
```

then **sometext: host1** is displayed in the title bar and task list. No spaces are allowed in the text.

-tx **filename** Reads the specified translation table. If the file name you specified does not contain the full path, TN5250 searches the TCPIP\ETC directory for the file.

If you do not specify -tx, TN5250 uses the translation file 5250XLT.TBL in the TCPIP\ETC directory. If this file does not exist in the TCPIP\ETC directory, TN5250 uses the default US translate table provided with the application.

-wide The 5250 architecture provides for a terminal having two screen sizes: the default and the alternate. TN5250 defines the default screen size to be 24x80 and the alternate to be 27x132. The alternate screen size is enabled by the **-wide** start up parameter. Most AS/400 programs can use the 24x80 screen size. The larger screen size is used to display spool files and some documents.

-? Displays help information.

Note: On screens with many extended attributes, the screen is painted without extended attributes before the extended attributes can be painted.

If you do not specify a host when issuing the **tn5250** command, tn5250 prompts you for a hostname, as shown in Figure 35.

PICTURE 35

Figure 35. Example of TN5250: Enter Host Name Window

If you select Enter without entering a host, you are returned to the OS/2 command prompt.

If you specify an invalid host, or if an active telnet server is not running on the host you specified, TCP/IP displays the Unknown Host message window, as shown in Figure 36.

PICTURE 36

Figure 36. Unknown Host Message Window

If this message is displayed, use the Enter key to cancel the process and clear the message.

Establishing a TN5250 session can take several seconds. If you do not establish a connection within six to eight seconds after entering a valid host name, a message box is displayed, as shown in Figure 37.

PICTURE 37

Figure 37. Connecting Message Window

If the connection takes more than two minutes to be established, a problem may exist. You can select **Cancel Connection** and verify that the destination host is operating correctly.

When a connection is established, the TN5250 window is displayed. Figure 38 is an example of the TN5250 window that is displayed when you connect to an AS/400 host.

PICTURE 38

Figure 38. Example of TN5250 Window

When you use TN5250, some of your keys function differently from the way they do in your regular OS/2 environment. For example, the F10 key functions as a PF key, which means that pressing the F10 key no longer selects the menu bar.

To select the TN5250 menu bar without a mouse, press the Shift key and Esc key simultaneously. The System menu is displayed; you can then use the arrow keys on your keyboard to select items from the menu bar.

8.6.2 Using the TN5250 menus

Select the TN5250 pull-down menus from the menu bar displayed in the TN5250 window.

The pull-down menus for TN5250 are:

Menu	**Description**
Mark	Allows you to copy to a clipboard or unmark a selected area
Configuration	Customizes mouse action definitions, and keyboard refreshing
Fonts	Allows you to select any of the VIO window fonts available on your display
Exit	Exits TN5250

8.6.3 Marking a Portion of the Screen on the Foreign Host

Select the **Mark** pull-down menu choice to copy a marked portion of your host screen to the clipboard or to remove the marking from a marked section.

Subtopics
8.6.3.1 Copy to Clipboard
8.6.3.2 Unmark

8.6.3.1 Copy to Clipboard

The **Copy to Clipboard** menu item allows you to copy rectangular portions of the screen to the clipboard. The copy action is accomplished by marking the area and copying it to the clipboard.

To mark and copy a portion of the screen, do the following:

1. Move the mouse pointer to any part of the screen you want marked, then press and hold down the mouse button.

   Drag the mouse over the area you want to select and release the mouse button. The marked area is colored to delimit the area selected.

2. Select the **Copy to Clipboard** menu item from the Mark pull-down menu to copy the marked area to the clipboard.

Note: Although the copy function is supported, the paste function is not supported.

8.6.3.2 Unmark

The **Unmark** menu item allows you to remove the marking from a marked area of the screen.

8.6.4 Configuring TN5250

Select the **Configuration** pull-down menu to configure TN5250 for use with the foreign host.

Subtopics
8.6.4.1 Mouse

8.6.4.1 Mouse

Select the **Mouse** menu item to view or change the current mouse action definitions. The Bind Action to Mouse Event window displays the current definitions of specific mouse buttons as they relate to specific function keys, as shown in Figure 39.

PICTURE 39

Figure 39. Bind Action to Mouse Event Window

On this window, you can assign certain actions when mouse buttons are clicked. The actions are:

Item	**Action**
Def	Moves the cursor to the location of the click.
F**n**	Moves the cursor to the location of the click, and invokes the PF keystroke that is specified. **n** represents the PF key numbers 1 through 12. For example, F1 invokes the PF1 keystroke, F2 invokes the PF2 keystroke, F3 invokes the PF3 keystroke, and so on, through F12.
Enter	Moves the cursor to the location of the click and invokes the Newline keystroke. The default key mapping for the Newline keystroke is the 5250 enter function.

A list of mouse events (items) used to initiate specific actions appears in the window. The action that applies to a mouse event appears adjacent to each event. The up and down arrow buttons are used to change the definitions.

The three buttons at the bottom of the window are:

Button	**Description**
Save	Applies the mouse definitions that were specified in the Mouse Event window to the TN5250 terminal session, and saves the definitions to be automatically used the next time TN5250 is invoked
Apply	Applies the mouse definitions that were specified in the Mouse Event window to the TN5250 terminal session
Cancel	Restores the definitions to the values that were in effect before the Mouse Event window was created

To change the mouse button action definitions:

1. Using the mouse, click on the specific arrow for a desired change.
2. Using the mouse, click on the appropriate button at the bottom of the window.

8.7 Translation Table Support for TN5250

TN5250 supports external ASCII/EBCDIC translation tables. This support is provided by the following files:

\TCPIP\BIN\TN5250.EXE
\TCPIP\ETC\5250XLT.SMP

TN5250 uses the following syntax for specifying a translation table parameter:

-tx translate_table_file_name

If you specify a translation table file name but do not include its full path statement, the application searches for the file in the ETC subdirectory. If you do not specify the **-tx** option when starting a session, the application uses the file 5250XLT.TBL, located in the ETC subdirectory.

The 5250XLT.SMP file is shipped as a sample and contains the US ASCII/EBCDIC translations. If you want to use a different translation table file on a regular basis, copy that file to 5250XLT.TBL in your ETC subdirectory. If the file 5250XLT.TBL does not exist, TN5250 uses the default US translation table.

Subtopics
8.7.1 Remapping the Keyboard
8.7.2 Selecting Fonts
8.7.3 Logging Off TN5250

8.7.1 Remapping the Keyboard

Use **Refresh Key Mapping** with the TN5250.KEY file to edit the default key bindings that are activated when you start the TN5250 program. Edit the TN5250.KEY file using any text editor.

TN5250.KEY is a flat ASCII file that is loaded when you start the TN5250 program. You use it to change default key definitions.

When TN5250 is started, it first looks for TN5250.KEY in the current directory. If this fails, TN5250 tries to open the file in the directory defined in the ETC environment variable. Since TN5250 is started with default key bindings, TN5250.KEY need contain only those definitions that override the defaults or that provide definitions not included with the defaults.

The TN5250.KEY file is line-oriented, which means that each of its lines is either blank, contains a comment, or contains a definition. The semicolon is used in the file to delimit comments, and comments may appear anywhere on a line. A comment must begin with a semicolon and continue until the end of a line.

A line containing a binding follows this syntax:

os2_key whitespace 5250_function ;optional_comment

where the **os2_key** variable is one of the keys:

- F1 through F12
- Control-F1 through Control-F12
- Alt-F1 through Alt-F12
- Shift-F1 through Shift-F12
- Control-'unshifted_keypress'
- Alt-'unshifted_keypress'

and the **unshifted_keypress** variable is one of the keyboard's white keys, such as the letter keys, punctuation-mark keys, number keys, and foreign-language accent keys and is enclosed in single quotes. The **unshifted_keypress** variable represents the unshifted position for that key.

For example, to bind a function to the control-4 key combination, the **unshifted_keypress** variable is 4. It is impossible to bind control-$, since $ is the shifted state of 4.

The **function key** variable is a keyword identifying an OS/2 function key. Possible values of the **function key** variable are:

Backspace	Newline (Enter key on alpha keypad
BackTab (same as Shift+Tab)	PgUp
Del	PgDn
Down	Pause
End	PrtScr
Enter (Enter key on numeric keypad)	Right
Esc	Right-Alt
F1-F12	Right-Ctrl
Home	ScrLock
Ins	Space
Left	Up
Left-Alt	Tab
Left-Ctrl	

The **5250 function** variable specifies a 5250 function or a TN5250 application-specific function to which you can bind an OS/2 key. Possible values of the **5250 function** are:

Attn	Hex
Autoenter	Home
Backspace	Insert
BackTab	Kill-Word
Backward-Word	Left
Backward-Kill-Word	Newline
Cent	PA1
Cursor-Select	PA2
Clear	PA3
Delete	PF1-PF24
Down	Printscreen
DP-Backspace	Recback
Dup	Reset
End	Right
Enter	Rolldown

Erase-EOF	Rollup
Erase-Input	Selpen-attn
Field-Backspace	Spbackspace
Field-Mark	Sysreq
Fieldexit	Tab
Fieldminus	Testreq (works only on AS/400 sign n screen)
Fieldplus	Unlock
Forward-Word	Up
Help	

The following **5250 function** text functions work only in documents and mail:

Textaltd	Textnewline
Textbeginofline	Textpageend
Textbottomofpage	Textrqspace
Textcarrierreturn	Textrqtabfunction
Textcent	Textstopcodeadvance
Textdup	Textstopcodefunction
Textendattribute	Textsymbolscmdprompt
Textfastleft	Texttabadvance
Textfastright	Texttopofpage
Textintensify	Textunderscore

A sample TN5250.KEY file containing definitions looks like this:

```
  ; This is a sample TN5250.KEY file.  It should be located either
  ; in the current directory or in the directory pointed by the
  ; environment variable ETC.


  escape           clear  ; bind escape to clear (the default)
  control-'a'      home
  newline          newline; make the keyboard enter create a new line
  right-control    enter  ; make the right control key execute commands
  enter            enter  ; make numeric keypad enter execute commands
  shift-backspace  pf3    ; shift backspace invokes the pf3 function key
  backtab          tab    ; make shift-tab go to next field
```

You do not have to use the TN5250.KEY file if you use the default keyboard definitions. The default keyboard definitions are:

Workstation Key	**Function**
Enter	Field-Exit
Esc	ATTN
PgUp	PF7
PgDn	PF8
Shift-Enter	Field-Exit
Shift-Tab	Backtab
Shift-F1 through Shift-F12	PF13 through PF24
Right-Control	Enter

Note: These actions apply to the standard PS/2 keyboard. If you use another keyboard, the functions may vary.

If an OS/2 key is equivalent to a 5250 key function, and it is not listed above, the OS/2 key defaults to that function. or example, the OS/2 key insert defaults to the 5250 function insert.

Some keys cannot be remapped at all. The following accelerator keys cannot be remapped:

Workstation Key	**Function**
Alt-F4	Close
Alt-F5	Restore
Alt-F7	Move
Alt-F8	Size
Alt-F9	Minimize
Alt-F10	Maximize
Shift-escape	Menu Bar
Control-escape	Window List

8.7.2 Selecting Fonts

Use the Fonts pull-down menu to specify the font used in the TN5250 window. The following is an example of the pull-down menu that might be displayed as a result of selecting Fonts. The menu items displayed depend on the display adapter you are using.

The menu items describe the fonts (width ¦ height in pixels) that are available on your display adapter. The current font being used is checked. To change the font, select the desired font.

8.7.3 Logging Off TN5250

You should log off your session with the foreign host to exit TN5250. You can also use the Exit pull-down menu to leave TN5250.

Select **Exit TN5250** if you want to exit, then select **Yes** if you want to end the program. A window displays the message shown in Figure 40.

PICTURE 40

Figure 40. TN5250 Confirmation Window

Select **Resume** if you want to return to TN5250.

9.0 Chapter 9. Printing on a Local or Remote Printer

This chapter describes how to print files and other related functions. It describes how to:

- Set up a print server, which spools print jobs from other hosts on the network
- Print to a local or remote printer
- Redirect print jobs
- Print to a FAX machine

If you are a network administrator and need to set up a print server for other users or if you want to set up your own workstation as a print server, see "Setting Up a Print Server" in topic 9.2. If you want to use a print server that is already running, see "Setting Up the Client Environment" in topic 9.3.

Note: For a concise description of commands, subcommands, and parameters, see the *IBM TCP/IP Version 2.0 for OS/2: Command Reference*.

Subtopics

9.1 Overview of the Line Printer Commands

The commands shown in Table 5 can be used to set up the environment for remote printing and to manage a remote printer.

Table 5. Summary of Line Printer Commands

Print Command	Description	See
lpd	Starts the print server on your local host. The server spools print jobs for other hosts on the network.	9.2.1
lpr	Submits a file on your workstation to a remote printer attached to a print server.	9.4
lpq	Queries print jobs queued at a remote printer attached to a print server.	9.5
lprm	Removes jobs that are in a queue on a remote printer attached to a print server.	9.6
lprmon	Starts LPRMON, a parallel device monitor that sets up your workstation to automatically send data to a print server. This allows any application that can print to a parallel port to access a print server.	9.7.1
lprportd	Supports OS/2 workplace shell (WPS) printer objects. which allow you to redirect print jobs.	9.7.2.2

9.2 Setting Up a Print Server

The print server (also known as the LPR server, the LPR daemon and LPD) allows other hosts to use your workstation as a print server to:

- Submit print jobs to an OS/2 print queue associated with a printer object
- Query print jobs
- Remove print jobs from the printer queue

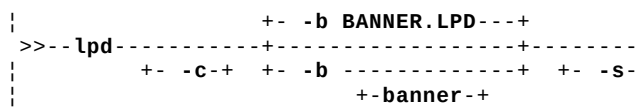
Note: The print server must be running on the workstation providing the print spooling service before you can issue Line Printer commands.

Subtopics
9.2.1 Starting the Print Server
9.2.2 Ending a Print Server

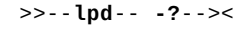
9.2.1 Starting the Print Server

To start the print server on your local host, enter the **lpd** command at an OS/2 command prompt. When you start the LPR server by issuing this command, LPD.EXE runs as a task until you shut down the server.

The syntax of the **lpd** command is:



Displaying lpd Help



-c Prevents printing of the control file.

-b **banner** Uses the information in the specified file as input for the print job **banner** (header) page. If this parameter is not specified, LPD uses the default banner page (BANNER.LPD in the \TCPIP\BIN directory). If you specify the -b parameter with no filename, a banner is not printed.

If you specify a banner file with the -b parameter, you can use the following keywords to customize the banner printed:

Keyword	**Description**
%H%	Name of the host that originated the print job
%U%	Name of the user that originated the print job
%J%	Name of the print job
%C%	Class of the print job

-s Validates client requests based on the port addresses. According to RFC 1179, all line printer requests should come from clients on a port within the range of 721 to 731 inclusive. By default, LPD does not verify that the client is connecting from a valid port within this range.

-f Changes the file format from binary (the default) to text.

-? Displays help information.

Notes:

1. As an alternative to issuing the **lpd** command from an OS/2 command prompt, you can start LPD automatically on page 2 of the Autostart tab in the TCP/IP Configuration notebook. For more information about the configuration notebook, see *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration*.

   Note: The **lpd** command does not accept any parameters when started automatically using INETD.

2. The -f parameter is used to toggle between the following file formats:

   - Binary. The file is sent directly to the specified queue for printing.
   - Text. Each line feed (LF) is converted to a carriage return line feed (CRLF) pair and the file is sent to the specified queue.

   If a client does not specify a file format, or if a format other than binary or text is specified, the default is binary.

9.2.2 Ending a Print Server

To end a print server on your local host:

1. Activate the window in which the server is running by clicking on it.
2. Hold down the Ctrl key and press the C key, or close the window.

The LPD.EXE server task is ended.

9.3 Setting Up the Client Environment

¦You must specify values for the following environment variables prior to
¦issuing the print command as described in this section.

Environment Variable	**Parameter**
LPR_PRINTER	The name of the printer that provides the output. The printer name corresponds to a queue defined at the server. If a device name is specified instead of a queue, that device's first queue is used.
LPR_SERVER	**server**. The IP address or name of the print server.
USER	**username**. An identifier of who created the print job. (The default is **PC_USER**.)

You can set the environment variables by one of the following methods:

- Completing the following fields in the Configuration notebook (Services tab, Page 2 of 3):

  - Remote Print Server
  - Remote Print Server's Printer

- Adding the following statements to your CONFIG.SYS file:

  ```
  SET LPR_PRINTER=printer
  SET LPR_SERVER=server
  SET USER=username
  ```

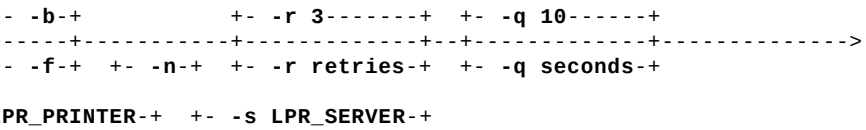
- Entering the following commands at an OS/2 prompt:

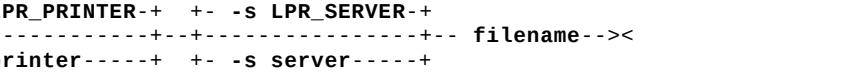
  ```
  SET LPR_PRINTER=printer
  SET LPR_SERVER=server
  SET USER=username
  ```

  Note: Environment variables set at an OS/2 command line are only valid for the current session and any sessions started by the current session.

9.4 Sending Print Jobs

The **lpr** command transfers the contents of a file from your workstation to a print server. The syntax of this command is:





Displaying lpr Help

-b Specifies that the data is interpreted as binary (the default).

-f When the print server is running on a UNIX(**) system, the -f parameter formats the file using the UNIX **pr** command. When the print server is running under OS/2, no formatting is done.

-n Displays no messages unless an error occurs.

-r **retries** Sets the number of retries (0-5). The default is 3.

-q **seconds** Sets the retry delay in seconds. The default is 10.

-p **printer** Specifies the name of the printer to which the file is sent. If the -p parameter is omitted, the LPR_PRINTER value used.

-s **server** Specifies the name or IP address of a network host with print spooling capabilities.

If a print server is not specified on the command line, the LPR_SERVER value is used.

If a print server is not specified with the **lpr** command or defined in the environment variable, the print server displays an error message and ends.

filename Specifies the name of the file to be sent to the printer.

You can specify a wildcard character (*) as part of the **filename**. For example, to print all of the TXT files in your current directory, you would enter the following command:

```
lpr *.TXT
```

-? Displays help information.

Subtopics
9.4.1 Examples

9.4.1 Examples

1. Using the Default Printer and Server

   To submit file **abc** to a printer that is:

   - Defined by the LPR_PRINTER environment variable, and
   - Connected to the print server defined by the LPR_SERVER environment variable

   you would issue the following command:

   ```
   lpr abc
   ```

2. Specifying a Print Server

   To submit file **abc** to a printer **lp**, which is connected to a print server named **prtserv.univ.edu**, you would issue the following command:

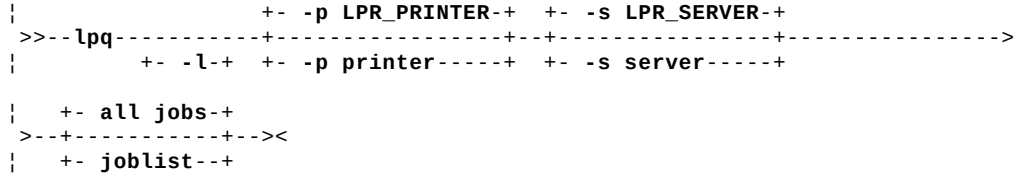
   ```
   lpr -p lp -s prtserv.univ.edu abc
   ```

   A sample response is:

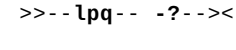
   ```
   Trying lpd print server prtserv.univ.edu(9.65.60.82), device lp.
   Sent 5192 bytes.
   The entire document was sent.
   ```

9.5 Querying the Print Queue

You can query the status of jobs queued for remote printing by issuing the **lpq** command. The syntax of this command is:



Displaying lpq help



-l Requests long output from the server. (The output depends on the operating system under which the server is running.)

Generally, short output (the default) provides printer status and the number of print jobs in the printer queue. Long ouput includes additional information about the printer queues and the print jobs in each queue, such as:

- Job identifier
- User identification
- Filename

-p printer Specifies the name of the printer to query. If the -p parameter is omitted, the LPR_PRINTER value is used.

-s server Specifies the name or IP address of the print server.

If a print server is not specified on the command line, the LPR_SERVER value is used.

If a print server is not specified with the **lpq** command or defined in the environment variable LPR_SERVER, LPQ displays an error message and ends.

joblist Specifies a list of job identifiers for the information to be displayed. Each operating system's server uses its own form of job identifier. The joblist consists of one or more of their identifier numbers separated by a single space.

The default is to return information about all jobs queued on the specified printer.

-? Displays help information.

Subtopics
9.5.1 Example

9.5.1 Example

You can obtain detailed information about the print queue by entering the following command:

```
lpq -l
```

Sample output is shown below.

```
Printer Queue Status - LONG VERSION

QUEUE NAME    STATUS    STATE    JOBS    PRINTER QUEUE DESCRIPTION
==========    ======    =====    ====    =========================
LPT1          UP        ACTIVE   3       (null)
    23    USER:  pc-user@damorris.raleigh.ibm.com
          FILE:  C:\INTERNET.TXT
    24    USER:  pc-user@janross.raleigh.ibm.com
          FILE:  C:\README.DOC
    25    USER:  pc-user@zena.raleigh.ibm.com
          FILE:  C:\TOOLS.CAT
COM1          UP        IDLE     0       (null)
    <the queue is empty>
```

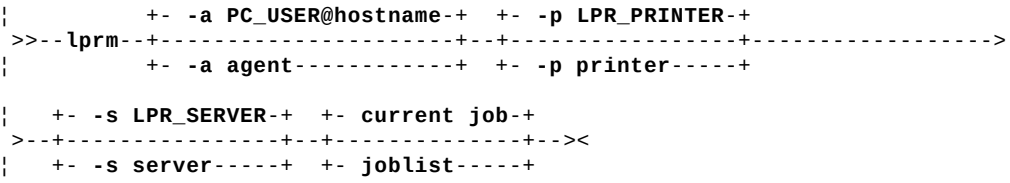
Note: The job identifier is shown to the left of the USER description. For example, the job identifier for the first job in the LPT1 printer queue is 23.

9.6 Removing Jobs from the Print Queue

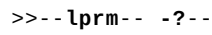
You can use the **lprm** command to delete jobs from a print queue.

Note: To use this command, you must first obtain the print job identifier from the printer queue (see "Querying the Print Queue" in topic 9.5 for more information). Because each operating system implements printer queues differently, you may need to refer to the system documentation for your remote print server to determine how to specify a job identifier.

The syntax of the **lprm** command is:



Displaying lprm Help



-a agent Specifies the agent name (user name) to send to the server. If an agent is not specified, LPRM will send the value of the USER environment variable followed by **@hostname** (where **hostname** is the symbolic name or IP address of your host). For example, pc_user@zena.raleigh.ibm.com.

-p printer Specifies the name of the printer from which you are trying to delete a job. If the -p parameter is omitted, the LPR_PRINTER value is used.

-s server Specifies the name or IP address of the print server.

If a print server is not specified on the command line, the LPR_SERVER value is used.

If a print server is neither specified with the **lprm** command nor defined in the environment variable, LPRM displays an error message and terminates.

joblist Specifies a list of job identifiers for the jobs to be removed from the queue. The default is to remove the current job in the queue.

-? Displays help information.

Note: If the server from which you are trying to remove a job does not accept the default agent name, you can override the value using the -a parameter. When you issue the **lpq** command with the -l parameter, the USER field indicates what the server expects as the agent information (see the "Example" in topic 9.5.1).

9.7 Redirecting Print Jobs

TCP/IP for OS/2 provides support for redirecting local OS/2 printers to any printer attached to an LPD server. This enables you to print from an application to a network printer without specifying any special application support. From the application perspective, it is actually printing to a local printer, but the output is then sent to a printer attached to an LPD server on your network.

¦You can redirect the output of a local printer object to a remote printer using either of the following methods:

- Using a parallel device monitor (LPRMON)
- Using the LPR port driver (LPRPORTD)

LPRMON is a parallel device monitor that redirects output from a specific parallel device to a remote printer. OS/2 currently supports three parallel devices: LPT1, LPT2, and LPT3. You can use LPRMON to redirect any of these parallel devices, as well as any OS/2 device that conforms to the Parallel Port Device Driver's Monitor interface.

¦The LPR port driver enables you to redirect any local printer object ¦instance to a remote printer. Unlike LPRMON, you can create up to 64 LPD ¦printer ports to support up to 64 remote printer destinations.

In addition to a remote server and a remote printer, printer objects enable you to configure many of the parameters that are passed to the LPD server. You can also pass free-form, server-specific parameters to the LPD server.

¦There are many advantages in using the LPR port driver instead of LPRMON, including :

- Support of up to 64 remote printers
- User configuration options
- Does not prevent the use of parallel ports for local printers

Subtopics

9.7.1 Using a Parallel Device Monitor

The **lprmon** command starts LPRMON, a Parallel Device Monitor, which is a program to intercept data sent to a parallel device (such as LPT1, LPT2, and so on). It enables you to set up your workstation to automatically send data to a remote LPR server. This allows you to print to an LPR server without an application using the Line Printer protocol directly.

Note: Use the -b option unless the remote LPD printer is strictly a text printer (the printer does not support embedded binary control characters).

```
                              +- -r 3-------+  +- -q 10------+
>>--lprmon--------------------+-------------+--+-------------+----------->
            +- -b-¦  +- -n-+  +- -r retries-+  +- -q seconds-+
            +- -f-+

¦   +- -p LPR_PRINTER-+  +- -s LPR_SERVER-+
 >--+-----------------+--+----------------+-- devicename-->
¦   +- -p printer-----+  +- -s server-----+
```

Displaying lprmon Help

```
>>--lprmon-- -?-->
```

-b Specifies that the data is interpreted as binary by the server, LPD.

-f When the print server is running on a UNIX system, the -f parameter formats the file using the UNIX **pr** command. When the print server is running under OS/2, LPD passes the file unchanged.

-n Disables the beep that occurs when there is an error.

-r **retries** Sets the number of retries (0-5). The default is 3.

-q **seconds** Sets retry delay in seconds. The default is 10.

-p **printer** Specifies the name of the printer to which the file is sent. If the -p parameter is omitted, LPRMON looks at environment variable LPR_PRINTER for the corresponding value.

-s **server** Specifies the name or internet address of a network host with print spooling capabilities.

If a print server is not specified on the command line, LPRMON looks at environment variable LPR_SERVER for the corresponding value and uses that value as the print server.

If a print server is not specified with the **lprmon** command or defined in the environment variable, LPRMON displays an error message and ends.

devicename Specifies the parallel port for LPRMON to monitor. Data sent to this port is then redirected to a remote LPR server. This must be specified as lpt**n**, where **n** is a number (1-3).

-? Displays help information.

To redirect a local printer to a remote printer using LPRMON:

1. Create a printer that supports output to one of the parallel printer objects (LPT1, LPT2, or LPT3)

¦ See "Creating and Configuring an LPD Printer Object" in topic 9.7.2.1 ¦ for more information.

2. Run LPRMON to redirect the corresponding device to a remote printer.

Note: LPRMON support is provided because you cannot use printer objects to redirect printer output from a Windows application running under OS/2. This can be accomplished only by creating a Windows printer attached to LPT1.OS2, LPT2.OS2, or LPT3.OS2. Then run LPRMON to redirect the corresponding device to a remote printer.

9.7.2 Using the LPR Port Driver

TCP/IP for OS/2 provides LPR printing capabilities which are fully integrated with the Workplace Shell*. After installing the LPR port driver, you can transmit your jobs to the server for printing by:

1. Creating a printer object (if one does not already exist)

2. Specifying a printer object as the output for that printer

3. Configuring the corresponding printer object to send output to a specific printer on a specific server

4. Printing to that printer object

Subtopics

9.7.2.1 Creating and Configuring an LPD Printer Object

Follow these steps to create and configure an LPD printer object.

1. Select an existing printer object, and click once with mouse button 2.

   Note: If you do not have an existing printer object on your desktop, you can create one if you:

   a. Select the **Templates** folder by double-clicking with mouse button 1 to display available templates.

   b. Drop and Drag the **Printer** object to your desktop area.

      The **Create a Printer Window** is displayed.

   c. Select an LPD printer port by following the instructions in Step 5 below.

2. Select the **Open** cascade menu by clicking once with mouse button 1 on the arrow next to **Open**.

3. Select **Settings** by clicking once with mouse button 1 to display the Settings notebook.

4. Select the **Output** tab from the Settings notebook. The Output page displays all of the printer objects from which you can choose, including all LPD printer ports that are available.

   Note: For more LPD printer ports, see "Changing the Number of LPD Printing Ports" in topic 9.7.2.3.

5. Double-click on an LPD printer port (for example, **\PIPE\LPD0**). The settings window for that port is displayed (see Figure 41).

   Note: Light hash marks indicate the port is already selected by another printer.

PICTURE 41

Figure 41. Example of the LPD Port Icon Window

This window contains entry fields and check boxes. Some fields are required, such as LPD Server and LPD Printer, and others are optional. Enter the appropriate information as described in the following paragraphs.

Print Destination

LPD server
Type the name or IP address of an LPD server where output is to be sent (for example, **prtserv.univ.edu** or **9.67.96.17**). **This is a required field**.

LPD printer
Type the name of the printer on an LPD server where the output is to be sent (for example, **lp**). The printer name corresponds to a queue defined at the LPD Server. If a device name is specified instead of a queue, that device's first queue is used. **This is a required field**.

Source Information

Host name Type the name sent in the control file as the source's host name. If left blank, this will default to your local host name.

User Type the name sent in the control file as the user creating the print job. The default is the user name.

Send to Server: Select the check boxes appropriate to your print job needs. These check boxes allow you to enable parameter passing for Spooler Parameters, Queue Parameters, and Network Parameters. These parameters are used for passing information to the spooler, queue processor, or the network driver processing a specific print job on the

printer server host. If you check any of these boxes in the LPD Printing Port settings window, the information associated with that job is sent in the control file.

The Prefix fields specify the string that will precede the data in the control file. The defaults that appear when you specify a new port are those expected by the OS/2 LPR server (LPD). For example, if the server recognizes an entry in the control file of the format --Q<string>, it places <string> in the new job's Queue parameters.

Display port settings on print: Select this check box to display the LPD Printing Port settings window for every print job. This gives you the opportunity to modify your print job specifications.

Enable data filter: Select this box specify or enable filters for converting data. See "Using an LPR Port Driver Filter" in topic 9.7.2.4 for more information.

Send mail on receipt: Select this check box to send mail to the specified user at the machine where the job originated.

User Type user information in the User field. If left blank, the User field from the Source Information group is used.

Note: The current OS/2 LPR server does not support sending mail when the print job is completed.

Print banner page: Select this check box to request a banner page to be printed with the job.

Class Type in banner page classification, for example **Confidential**. If left blank, the Hostname field from the Source Information group is used.

Additional Control Cards: Specify additional controls in the Additional Control Cards field. Type free-form text in this field. This text is parsed and keywords are replaced with corresponding values for the job being printed. To specify the format that you want your server to interpret data as, add a valid print request control card appropriate to your needs.

For example, if you would like the server to interpret data as a UNIX **pr** command input file, specify the following in the Additional Control Cards field:

```
p<jobfilename>
```

Note: The control card must be a valid print request control card to be accepted. In the previous example, if any blank spaces were placed after **p** or after **<jobfilename>**, this would not have been a valid print request, and the default would have been used in its place. The default control card is **l<jobfilename>**.

Table 6 shows a list of valid print requests. For more information, see RFC 1179.

Table 6. Valid Print Requests

c<jobfilename>	n<jobfilename>
d<jobfilename>	p<jobfilename>
f<jobfilename>	r<jobfilename>
g<jobfilename>	t<jobfilename>
k<jobfilename>	v<jobfilename>
l<jobfilename>	x<anydata> <jobfilename>

Note: The **x** print request is an extended request, and the <anydata> must be separated from <jobfilename> by a single blank space.

If one of the above commands is not in the Additional Control Cards field, the default is used (l<jobfilename>).

Table 7 shows a list of acceptable keywords.

Table 7. Valid Keywords

<jobid>	<position>
<username>	<submitted>
<priority>	<comment>
<size>	<notifyname>
<document>	<parms>
<datatype>	<queue>
<status>	<qprocparms>
<qprocname>	<printername>
<drivername>	<jobfilename>
<portname>	

These keywords correspond to fields in the PRJINFO3 structure for a specific print job. For example, you could add the following control card to your control file:

```
Xpriority=<priority>.
```

By default (priority default) this adds **xpriority=50** to your control file.

If you want a less-than (<)character in the data, it must be preceded by a backslash character (\). If you want a backslash character, enter two of them (\\).

For example:

```
Xpriority=\<<priority>>
```

becomes

```
Xpriority=<50>
```

If a keyword does not match one of the predefined ones listed above, the environment is checked for a matching environment variable.

Note: Environment variables are case-sensitive.

If a job fails, you will be prompted to either cancel, retry, or return to the setup window for the specified port. If you want a printer to be the default destination for all of your printing, click once with mouse button 2 on any printer object. Select **Set default** and then select the printer to be the default.

Note: LPRPORTD must be running when printing to an LPD printer object.

9.7.2.2 Starting the LPR Port Driver Support

LPRPORTD must be running before you can print to an LPD printer object. To start LPRPORTD, type **lprportd** at an OS/2 command prompt and press the Enter key.

The syntax of this command is:

>>--**lprportd**--><

The **lprportd** command has no parameters.

Note: LPRPORTD can be started automatically on page 3 of the Autostart tab in the TCP/IP Configuration notebook.

9.7.2.3 Changing the Number of LPD Printing Ports

The default number of LPD printer ports is 8. LPD printer ports can be viewed on the Output tab of a printer object Settings notebook. The LPD printer ports are labeled **\PIPE\LPDx**.

You can specify up to 64 LPD printer ports, allowing up to 64 different remote printer destinations. (You can have multiple printer objects sending output to the same port). Use the TCP/IP Configuration Notebook (TCPIPCFG) to change the number of LPD printer ports. On the Services tab (page 2), change the Maximum Number of LPD Ports field to the desired amount.

For more information about the TCP/IP Configuration Notebook, see *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration*.

Note: LPRPORTD must be running when printing to an LPD printer object.

9.7.2.4 Using an LPR Port Driver Filter

If you use an LPR port driver to redirect your print jobs, you can specify filter programs that convert data to a form required by a specific device (for example, a FAX machine). For more information, see "Sending Outbound FAXes" in topic 9.8.

A *filter* is a program used to specify a command run by the LPR port driver. The command receives print data as input (through standard input). The output produced (to standard output) is sent to the LPD server.

The filter program must contain data that the command interpreter (CMD.EXE) can execute. For example, the following REXX code adds a time and date stamp at the beginning of the data printed:

```
/* DATESTMP.CMD: Adds a time and date stamp to the beginning of a file */
   say 'Printing: 'date('W') date('N') 'at' time('C')
   do while stream('STDIN:','S') == 'READY'
           say linein()
   end
   exit 0
```

Note: Because the LPR port driver interprets a nonzero return code as an error, it will not send the job to the server to be printed unless your filter program generates a return code of zero.

You can specify an LPR port filter on the LPD Printing Port settings window (see Figure 41 in topic 9.7.2.1) as follows:

- Select the **Enable data filter** checkbox
- Specify a filter command in the **Filter** field

For example, if you want a time stamp containing the name of the file printed at the beginning of a file, you would:

1. Modify DATESTMP.CMD (the previous example) as follows.

```
/* DATESTMP.CMD: Adds a time and date stamp to the beginning of a file */
   parse arg filename
   say 'Printing: 'filename' on 'date('W') date('N') 'at' time('C')
   do while stream('STDIN:','S') == 'READY'
           say linein()
   end
   exit 0
```

2. Enter the following filter command in the **Filter** field of the LPD port icon Settings window:

```
DATESTMP <document>
```

Note: You can also use this window to enter parameters for the specified program in the **Filter** field or pass **Additional Control Cards** keywords. (See also "Additional Control Cards" in topic 9.7.2.1.)

Table 8 lists the filters provided.

Table 8. LPR Port Driver Filters

Filter Name	Description
NULFLTR.EXE	Parameters passed to this pass-through filter are added to the top of the data file.
PMFAXW.EXE	Remote FAXes are sent through a server running OS/2 TCP/IP LPD.EXE and FaxWorks for OS/2 from SofNet. It requires the IBM Proprinter* as the Printer Object's printer driver.

9.7.2.5 The PMFAXW.EXE Filter

If you specify the **pmfaxw** filter, the FaxWorks filter window is displayed (see Figure 42).

PICTURE 42

Figure 42. Example of the Initial FaxWorks Filter Window

Where:

To

Name, Company, FAX#
Prints addressing information for the recipient on the cover page

From

Name, Company, Phone #, Fax #
Prints sender identification information on the cover page

Comment Prints Comment(s) on the cover page

Header Defines and implicitly enables page headers

The following parameters can be passed to PMFAXW.EXE to specify the defaults for the corresponding fields on the dialog box.

```
>>TO=<name>,<company>,<faxnumber>
>>FROM=<name>,<company>,<phonenumber>,<faxnumber>
>>INFO=<comment>,<header>
```

If you do not specify all of these parameters, you must include a comma as a placeholder for the empty fields. A blank field is displayed for those parameters not specified.

For example, if you issue the following command at an OS/2 prompt, the FaxWorks Filter window is displayed (see Figure 43).

```
PMFAXW ">>FROM=JP Doe,Widget's,,(xxx)yyy-zzzz" ">>INFO=,<documentname>"
```

PICTURE 43

Figure 43. Example of a Completed FaxWorks Filter Window

9.8 Sending Outbound FAXes

TCP/IP for OS/2 provides outbound FAX support using an LPR port driver filter. Any application that can print to an OS/2 print object can send outbound FAXes through an OS/2 LPD server.

Notes:

1. Receiving FAXes or distributing them to clients is not supported.

2. The print resolution using the IBM Proprinter printer driver is not as fine as the resolution obtained when using the FaxWorks for OS/2 printer object directly.

Subtopics
9.8.1 Additional Software Requirements for Outbound FAX Support
9.8.2 Configuring the LPR Port Driver to Send Outbound FAXes

9.8.1 Additional Software Requirements for Outbound FAX Support

Table 9 shows the additional software required.

Table 9. Software Requirements for Fax Support

Application	**Software Required**
Client	None
Server	FaxWorks for OS/2 from SofNet (for each server)

9.8.2 Configuring the LPR Port Driver to Send Outbound FAXes

You can configure the LPR port driver to send outbound FAXes as follows:

1. Create an OS/2 Workplace Shell print object using the printer icon in the desktop folder. (See "Creating and Configuring an LPD Printer Object" in topic 9.7.2.1 for more information.)

2. Associate the IBM Proprinter Printer driver with that print object.

3. Specify an LPD Port on the Output Page of the Settings notebook.

4. Configure the LPD port icon (see Figure 43 in topic 9.7.2.5) as follows:

   - Specify an OS/2 LPD server with the FaxWorks for OS/2 software installed in the **LPD server** and **LPD printer** fields

   - Select the **Enable data filer** checkbox

   - Specify a filter command in the **Filter** field (also see "Using an LPR Port Driver Filter" in topic 9.7.2.4)

10.0 Chapter 10. Using the TALK Protocol

Use TALK to send and receive interactive electronic messages and to provide a method for exchanging simple messages between users on the network. The **talk** command uses the TALK protocol, which is a TCP/IP protocol that sends and receives electronic messages.

This chapter describes the **talk** command and associated parameters.

Note: For a concise description of commands, subcommands, and parameters, see the *IBM TCP/IP Version 2.0 for OS/2: Command Reference*.

Subtopics

10.1 Setting Up the Environment

To use the **talk** command, the originating machine:

- Must have the HOSTNAME environment variable defined
- Must have the same value (for the environment variable) defined on an accessible name server or in the HOSTS file

In addition, the originating host name must also exist in the destination host's name server or HOSTS file. If the originating host name does not exist in the destination host's name server, the following message is generated to the originating host:

```
+-------------------------------------------------------------------------------+
¦                                                                               ¦
¦ Target machine does not recognize us.                                         ¦
¦                                                                               ¦
```

10.2 Starting the Server

To use the **talk** command, the TALKD server must be running on **both** the local and foreign hosts to exchange TALK messages.

To start the server on your local host, type **talkd** at an OS/2 command prompt and press the Enter key. The **talkd** command starts the TALK program.

If you hold down the Ctrl key and press the Esc key, **TALKD.EXE** is displayed in the Window List. TALKD.EXE runs as a task until you shut down the server.

10.3 Entering the talk Command

The **talk** command is useful for sending short messages that do not need verified responses. Because the **talk** command is interactive, it cannot be used to send messages that must be filed.

```
>>--talk-- user@host----------------><
                     +- ttyname-+
```

Displaying talk Help

```
>>--talk-- -?--><
```

user Specifies the name of the user to receive the message on the remote host. For an OS/2 destination, the user name is os2user. For other destinations that support **talk**, the user is the user name of the person with whom you are attempting to communicate.

@host Specifies the host name of the remote host.

ttyname A term that is assigned to users at logon. The terminal identifier, available for display, is unique for each logon session.

-? Displays a list of the parameters.

10.4 Sending and Receiving Messages with TALK

When running the TALKD server, you are notified by a double beep when someone is attempting to send a TALK message to you. To receive a TALK message, the TALKD server must be running in the remote machine. If you use the TALK client without starting the TALKD server, the TALKD server will be started for you. The following example illustrates this:

```
[C:/] talk os2user@host
```

Note: Both parties must have the TALKD server running.

The following illustrates the sequence of actions that occur when you communicate with another OS/2 user.

In the example, your host name and user ID is os2user@normapc; the remote OS/2 user to which you are connecting is os2user@chrispc.

In OS/2, TALK messages are entered at the bottom of the screen and are displayed at the top of the remote user's screen. Enter the **talk** command as shown:

```
talk os2user@chrispc
```

The following response is displayed on your screen as a result of the connection attempt:

```
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

No connection yet
Trying to connect to your party's talk daemon
```

The OS/2 user os2user@chrispc hears a double beep, and the following messages are displayed in the TALKD.EXE window of the host chrispc:

```
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

Message from talk daemon@ at 0:38...
talk: Connection requested by os2user@normapc
talk: Respond with:talk os2user@normapc
```

You are identified to os2user@chrispc as os2user@normapc.

To reply, the OS/2 user os2user@chrispc presses Ctrl and Esc. The Window List is displayed.

The OS/2 user os2user@chrispc selects Desktop from the Window List and presses the Enter key.

From the Desktop the OS/2 user os2user@chrispc selects an OS/2 command prompt and presses the Enter key.

The OS/2 user os2user@chrispc enters the following **talk** command at an OS/2 prompt, specifying your ID and host name, os2user@normapc.

```
talk os2user@normapc
```

The following messages are displayed:

```
::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::

No connection yet
Connection established
```

```
+------------------------------------------------------------------------------+
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
+------------------------------------------------------------------------------+
```

The following messages are displayed at your os2user@normapc screen:

```
+------------------------------------------------------------------------------+
|                                                                              |
| ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::   |
|                                                                              |
| No connection yet                                                            |
| Trying to connect to your party's talk daemon                                |
| Waiting for your party to respond                                            |
| Waiting for your party to respond                                            |
| Connection established                                                       |
|                                                                              |
|                                                                              |
+------------------------------------------------------------------------------+
```

The OS/2 user os2user@chrispc enters a TALK message:

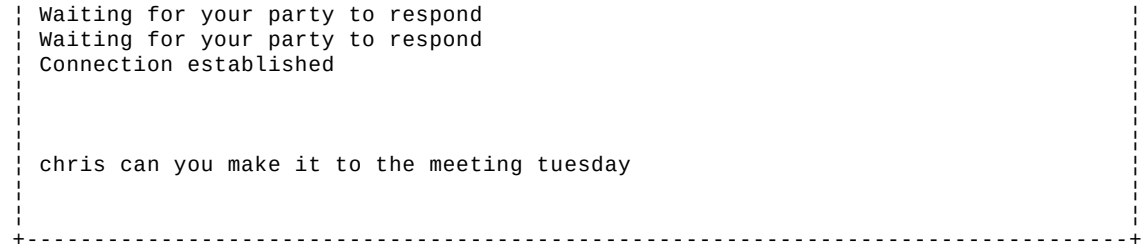
```
+------------------------------------------------------------------------------+
|                                                                              |
| ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::   |
|                                                                              |
| No connection yet                                                            |
| Trying to connect to your party's talk daemon                                |
| Waiting for your party to respond                                            |
| Waiting for your party to respond                                            |
| Connection established                                                       |
|                                                                              |
|                                                                              |
| hello this is chris                                                          |
|                                                                              |
+------------------------------------------------------------------------------+
```

Your (os2user@normapc) screen displays the TALK message from the OS/2 user os2user@chrispc:

```
+------------------------------------------------------------------------------+
|                                                                              |
| hello this is chris                                                          |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
|                                                                              |
| ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::   |
|                                                                              |
| No connection yet                                                            |
| Trying to connect to your party's talk daemon                                |
| Waiting for your party to respond                                            |
| Waiting for your party to respond                                            |
| Connection established                                                       |
|                                                                              |
|                                                                              |
|                                                                              |
+------------------------------------------------------------------------------+
```

You (os2user@normapc) enter your TALK message at the bottom of the screen:

```
+------------------------------------------------------------------------------+
|                                                                              |
| ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::   |
|                                                                              |
| No connection yet                                                            |
| Trying to connect to your party's talk daemon                                |
```



You (os2user@normapc) and the remote user (os2user@chrispc) can continue to exchange TALK messages until you choose to end the session.

10.5 Ending the TALK Session

Hold down the Ctrl key and press the D key to end the TALK session and disconnect any connected remote user.

The remote user sees the following message displayed:

```
+--------------------------------------------------------------------------------+
¦                                                                                ¦
¦ Your party has closed connection. Exiting.                                     ¦
¦                                                                                ¦
```

10.6 Ending the Server Task

Follow these steps to end the TALKD server task:

1. Activate the window in which the server is running by clicking in it.
2. Hold down the Ctrl key and press the Esc key.

The TALKD.EXE server task is ended.

11.0 Chapter 11. Issuing Commands on a Foreign Host with REXEC

This chapter describes the **rexec** command, which is used to issue a command on a foreign host. The **rexec** command sends a single command to a foreign host.

This chapter describes how to set the environment variables, set up the server, and issue commands on a foreign host.

Note: For a concise description of commands, subcommands, and parameters, see the *IBM TCP/IP Version 2.0 for OS/2: Command Reference*.

Subtopics

11.1 Setting Up the Environment

Before you activate the REXEC server, set the environment variables user and passwd. These environment variables define the user's ID and password, which a remote user specifies to log on to your workstation. The values for these environment variables are case-sensitive.

The following is an example of the user and passwd environment variables:

```
set user=userid
set passwd=password
```

You can set these environment variables in your CONFIG.SYS file, or you can type them in a command shell before starting the REXEC server. Using a command shell has the advantage that only the command shell and any functions running in it have knowledge of the user and passwd variables.

Subtopics

11.1.1 NETRC File

11.1.1 NETRC File

The NETRC file is used by the REXEC (and FTP) client as a source for user and password values. To create the NETRC file, use your system editor.

The location of the NETRC file can be customized depending upon the environment variables used. The following procedures enable you to establish the location of the NETRC file:

- If the NETRC environment already exists, the path and file names specified are used as the NETRC file. For example, **set netrc=c:\bin\etc\mynet.rc** causes the FTP and REXEC clients to look for the file MYNET.RC in the C:\BIN\ETC directory.

  This is intended to give you greater security in allowing you to hide your NETRC file. To do this, type the following line at the command prompt before doing a set of FTP or REXEC.

  ```
  set netrc=netrc_pathname
  ```

- If the NETRC environment does not exist, the ETC environment variable is used as the path and NETRC is used as the file name. For example, **set etc=c:\bin\etc** causes the FTP and REXEC clients to look for the file NETRC in the C:\BIN\ETC directory.

- If neither of the variables exist, the default file name NETRC is used with the default path C:\ETC.

The following is an example of a NETRC file containing multiple entries:

```
machine raleigh login kent password baseball
machine boston login bruce password september
machine 251.1.11.3 login jane password workday
```

In this example, when you issue the **rexec** command to host **raleigh**, the user name **kent** and password **baseball** are automatically sent to the foreign host.

In this example, the foreign host called **raleigh** has a REXEC server running and has environment variables set to the following values:

```
set user=kent
set passwd=baseball
```

Warning: If you have a Telnet, REXEC, TFTP, or FTP server running on your machine, be aware that a NETRC file provides users with user and password information that may allow them access to other users' files.

11.2 Setting Up the Server

To use the **rexec** command, the REXECD server must be running on the foreign host. To start the server on your local host, type **rexecd** at an OS/2 command prompt, and press the Enter key.

```
            +- -t 180-----+
>>--rexecd--+-------------+-><
            +- -t seconds-+
```

-t **seconds** The number of seconds before the server will timeout while servicing a client command. The default is 180. If you do not want a timeout, specify -1.

The **rexecd** command starts the REXECD.EXE program on the server and runs as a task until you shut down the server.

Security can be an issue when the REXEC server is running. If a remote user learns the user ID and password on your system, that remote user can issue commands on your workstation.

As an alternative, you can start REXECD using INETD. INETD allows you to start multiple servers from a single OS/2 session. For more information about INETD, see *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration*.

¦**Note:** If you use INETD to autostart REXECD, you cannot specify any ¦parameters for this service. However, you can add the following ¦environment variable to your CONFIG.SYS file to specify the timeout:

¦ SET REXECDTIME=**time**

Warning: Use discretion in allowing other users to learn your user and passwd variables.

11.3 Ending the Server Task

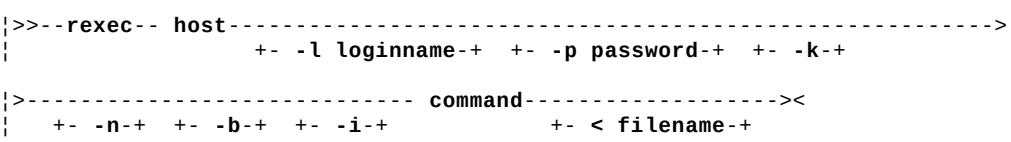
Follow these steps to end the REXEC server task on your workstation:

1. Activate the window in which the server is running by clicking in it.
2. Press down the Ctrl key and press the C key.

The REXECD.EXE server task is ended.

11.4 Using the rexec Command on a Foreign Host

Use the **rexec** command to issue commands on a foreign host.



Displaying rexec Help

host Specifies the remote host on which the command is to be issued.

-l **loginname**
Specifies the user ID on **host**. If you do not specify a login name, the values in the NETRC file are used.

-p **password**
Specifies the password that is associated with the login name. If you do not specify a password, the values in the NETRC file are used. If the NETRC file does not provide the password value, **rexec** prompts you for the password. You can enter the password in a nonecho mode.

-k Ignores the local keyboard input. This is helpful for running noninteractive input, especially from a batch file.

-n Specifies not to use the NETRC file for automatic login.

-b Transfers data to and from the remote host using binary mode. If you do not use this switch, REXEC transfers the data using ASCII mode.

If you omit this switch, REXEC will strip the CR from each CRLF pair when sending data to the remote host and will add a CR to each LF on data received from the remote host.

-i Uses interactive mode for input. REXEC reads input directly from the keyboard instead of from standard input (stdin).

command Specifies the command to be issued on the remote host. The command must be in the syntax used by the remote host.

< **filename**
Specifies the name of a file containing input to be used by the command being issued on the remote host. Your **filename** can also include the path. In cases where a command requires or accepts additional input, you can use this parameter to specify that the input comes from a file rather than from the keyboard.

-? Displays help information.

Note: Unspecified **loginname** and **password** parameters default to the values in the NETRC file, if one exists.

The **rexec** command passes the user ID, password, and commands to the REXEC server on the foreign host. The REXEC server authenticates the specified **loginname** and **password** against the values specified for the user and passwd environment variables. If the authentication fails, which means that the values do not match, an error is returned. If the authentication succeeds, the commands are issued and the remote console responses are returned to the local workstation.

Subtopics

11.4.1 Example

If you want to copy the contents of a file on your OS/2 machine to your UNIX machine, you could enter the following command:

```
rexec unix "cat >remotefile" <localfile
```

Where:

unix is the name of the remote host where the command is to be issued.

cat is the command to be issued.

remotefile is the input file (on the OS/2 machine).

localfile is the output file (on the UNIX machine).

Note: If you want the file interpreted as binary data, you must specify the -b parameter.

11.4.2 Sending a Command to a Foreign Host with a Prompted Password

The following example illustrates how to issue a command on the foreign host. This example shows you how to connect to the foreign host **host.edu** and issue the command **/usr/ucb/from**. The user ID on the foreign host is **usermike**. Because the -p parameter is not specified, and the NETRC file does not provide the password value, you are prompted to enter the password to complete the connection to **host.edu**.

```
REXEC host.edu -l usermike /usr/ucb/from
```

12.0 Chapter 12. Issuing Commands on a Foreign Host with the RSH Command

This chapter describes the RSH program, which is used to issue a command on a foreign host. The **rsh** command sends a single command to a foreign host. Unlike the **rexec** command described in the previous chapter, RSH neither requires nor allows you to provide a password.

The remote host must be set up before it can accept a service request from the local workstation. This chapter describes how to set up the server and issue commands on a foreign host using RSH.

Note: For a concise description of commands, subcommands, and parameters, see the *IBM TCP/IP Version 2.0 for OS/2: Command Reference*.

Subtopics

12.1 Setting Up the Client

RSH is similar to REXEC in that both require you to have an account on a remote machine. However, they differ in the authorization scheme. The REXEC server serves any client who can correctly specify an existing account and the account password on the server machine. RSH servers read a list that identifies the user on a specific remote client that has the same privileges as a particular server user. No password is required, but a client user has to be registered before service is provided.

To satisfy an RSH server, a single-user RSH client may supply a fictitious client user ID to the server. The value should be based on the server on which the user account is invoked and the associated client user ID. Set the user environment variable to this value. The user environment variable needs to be adjusted when a different remote user account is invoked. For example,

```
set user=mike
```

Note: The value of this variable is case-sensitive.

12.2 Setting Up the Server

Before you activate the RSH server, create the file RHOSTS in the directory pointed to by the ETC environment variable. The file defines the client hosts (and, optionally, the users on the client hosts) that are served by the RSH server. Entries in the file are case-sensitive. For example,

```
yktvmz.watson.ibm.com ralph
buzz.raleigh.ibm.com
```

Note: You must specify the authorized hosts by their full domain names.

If no user is specified on a line of the RHOSTS file, any user on that host can be served. If a user is specified, only that user can obtain RSH service. Only one user can be specified for each host, and one or more blanks must separate the user name from the host name.

To use the **rsh** command, the RSHD server must be running on the foreign host.

To start the server on your local host, type **rshd** at an OS/2 command prompt, and press the Enter key.

```
>>--rshd--><
```

The **rshd** command starts the RSHD.EXE program on the server and runs as a task until you shut down the server.

As an alternative, you can start RSHD using INETD. INETD allows you to start multiple servers from a single OS/2 session. For more information about INETD, see *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration*.

You can specify a RSHD timeout environment variable in your CONFIG.SYS file as follows:

```
set RSHD.TIMEOUT.IN.SECONDS=number
```

where **number** is the number of seconds.

12.3 Ending the Server Task

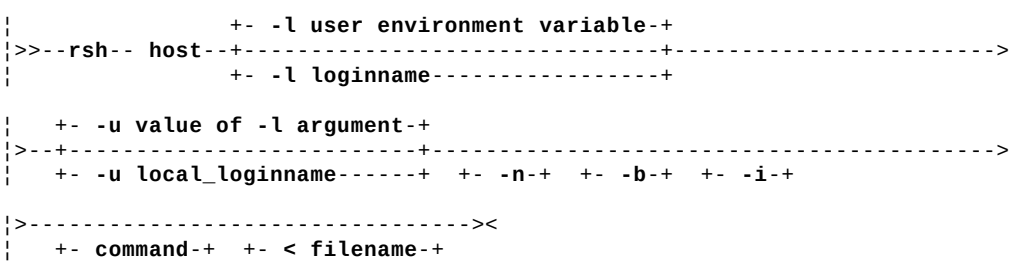
Follow these steps to end the RSH server task on your workstation:

1. Activate the window in which the server is running by clicking in it.
2. Hold down the Ctrl key and press the C key.

The RSHD.EXE server task is ended.

12.4 Using the rsh Command on a Foreign Host

Use the **rsh** command to issue commands on a foreign host.



host Specifies the remote host on which the command is to be issued.

-l **loginname**
Specifies the user ID on the remote host. If you do not specify this parameter, the value of the USER environment variable is used.

-u **local_loginname**
Specifies the user ID on the local host. If you omit this value, the value of the -l argument is used by default.

For example, to use RSH with a VM remote host, you would use -l for the VM password and -u for the VM user ID.

-n Ignores the keyboard input during the issuing of commands. This is helpful for some noninteractive commands, especially when they are run from batch files. It blocks input from the keyboard.

-b Transfers data to and from the remote host using binary mode. If you do not use this switch, RSH transfers the data using ASCII mode.

If you omit this switch, RSH will strip the CR from each CRLF pair when sending data to the remote host and will add a CR to each LF on data received from the remote host.

-i Uses interactive mode for input. RSH reads input directly from the keyboard instead of from standard input (stdin).

command
Specifies the command to be issued on the remote host. The command must conform to the syntax used by the remote host.

< **filename**
Specifies the name of a file containing input to be used by the command being issued on the remote host. Your **filename** can also include the path. In cases where a command requires or accepts additional input, you can use this parameter to specify that the input comes from a file rather than from the keyboard.

Note: The user environment variable specifies the client user ID that is to be passed to the server host to obtain RSH service.

The **rsh** command passes user, **loginname**, and **command** to the RSH server on **host**. The RSH server then authenticates the service based on user, **loginname**, and the domain name of your workstation. If authentication fails, an error is returned. Otherwise, the **command** is issued, and the server console responses are returned to the client.

Subtopics
12.4.1 Example
12.4.2 Sending a Command to a Foreign Host

12.4.1 Example

If you want to copy the contents of a file on your OS/2 machine to your UNIX machine, you could enter the following command:

```
rsh unix "cat >remotefile" <localfile
```

Where:

unix is the name of the remote host where the command is to be issued.

cat is the command to be issued.

remotefile is the input file (on the OS/2 machine).

localfile is the output file (on the UNIX machine).

Note: If you want the file interpreted as binary data, you must specify the -b parameter.

12.4.2 Sending a Command to a Foreign Host

The following example illustrates how to issue a command on the foreign host. It shows you how to connect to the foreign host **host.edu** and issue the command **/usr/ucb/from**. The user ID on the foreign host is **usermike**. The foreign host authorizes **userpam** on your workstation to log on to the account of **usermike**.

```
set user=userpam
rsh host.edu -l usermike /usr/ucb/from
```

13.0 Chapter 13. Managing Your TCP/IP Network

This chapter describes problem detection and problem determination. There are numerous programs available to perform problem detection for a TCP/IP network. TCP/IP for OS/2 provides several utilities that are useful in problem determination. These utilities are summarized in Table 10.

Table 10. Summary of TCP/IP Problem Determination Utilities

Task	**Utility**	**See**
Verifying the availability of a host	PING	13.1
Monitoring the availability of a group of hosts	PMPING	13.2
Displaying information about users on a remote host	FINGER	13.3
Displaying the status of RPC programs and services	RPCINFO	13.4
Collecting packet trace data	IPTRACE	13.5
Displaying network status information about your host	NETSTAT	13.6
Displaying network status information about other hosts	SNMP	13.7
Querying name servers	NSLOOKUP	13.8

See the *IBM TCP/IP Version 2.0 for OS/2: Command Reference* for an alphabetical listing of all network management-related commands and their parameters.

See Figure 44 for an example of the various screens used to provide network management functions.

PICTURE 44

Figure 44. Example Screens for PMPING, SNMPTRAP, SNMPGRP, SNMP GET/NEXT, NETSTAT

Subtopics

13.1 Verifying the Availability of a Single Host

You can use PING (the packet internet groper) to verify whether you can reach host B from host A. PING is a diagnostic tool that uses the services of the Internet Control Message Protocol (ICMP) to send an ICMP echo request to the specified host.

Virtually all implementations of TCP/IP can respond to a PING request; therefore PING provides a lowest-common-denominator approach to managing your network.

PING is often used in conjunction with SNMP. If an SNMP request is made to a foreign host, and no response is returned, you can use PING to verify whether the foreign host can be reached.

The syntax of the **ping** command is:

```
>>--ping----------------------------- host------------------------------>
          +- -d-+  +- -r-+  +- -v-+

>---------------------------------><
   +- data_size----------------+
                +- npackets-+
```

Displaying ping Help

```
>>--ping-- -?--><
```

-d Starts the socket-level debugging process.

-r Bypasses the routing tables and sends packets directly to a host on an attached network. If the host is not on a directly-connected network, PING cannot make a connection. This parameter can be used to ping a local host through an interface that no longer has a route through it.

-v Specifies verbose output.

host Specifies the IP address or host name of the remote host to which you want to send the echo request.

data_size Sets the number of data bytes for the echo request (the default number of data bytes is 56, with an additional 8-byte header attached).

npackets Sets the number of echo requests that are sent to the remote host.

These parameters are position dependent; you cannot specify the number of packets without specifying the data size.

Note: If you do not specify **npackets**, the echo request is sent continuously until one of the following actions stops the echo request:

- Pressing the Ctrl and C keys simultaneously
- Pressing the Ctrl and Break keys simultaneously
- Closing the task

-? Displays help information.

Subtopics
13.1.1 Sending Echo Requests to a Foreign Host Continuously
13.1.2 Sending a Specific Number of Echo Requests to a Foreign Host
13.1.3 What to do When a Foreign Host Doesn't Respond

13.1.1 Sending Echo Requests to a Foreign Host Continuously

By default, PING sends echo requests continuously to the foreign host, unless you specify a number of packets (**npackets**) in the **ping** command.

Subtopics
13.1.1.1 Example

13.1.1.1 Example

To send continuous echo requests to the foreign host **coll.univ.edu**, enter:

```
ping coll.univ.edu
```

PING displays a line of information for each PING echo request:

```
Ping coll.univ.edu: 56 data bytes
64 bytes from 129.34.128.248 icmp_seq=0.time=32.ms
64 bytes from 129.34.128.248 icmp_seq=1.time=62.ms
64 bytes from 129.34.128.248 icmp_seq=2.time=15.ms
```

You can cancel a continuous PING in one of the following ways:

- Hold down the Ctrl key and press the C key
- Hold down the Ctrl key and press the Break key
- Close the task

13.1.2 Sending a Specific Number of Echo Requests to a Foreign Host

If you do not want to send a continuous PING, you can specify the number of echo requests to be sent.

Subtopics
13.1.2.1 Example

13.1.2.1 Example

To send 4 echo requests with a data size of 512 to the foreign host 129.66.254.8:

```
ping 129.66.254.8 512 4
```

A sample response is:

```
520 bytes from 129.66.254.8: icmp_seq=0.time=31.ms
520 bytes from 129.66.254.8: icmp_seq=1.time=32.ms
520 bytes from 129.66.254.8: icmp_seq=2.time=31.ms
520 bytes from 129.66.254.8: icmp_seq=3.time=31.ms
--129.66.254.8 PING Statistics--
4 packets transmitted, 4 packets received, 0% packet loss
round-trip (ms) min/avg/max = 31/31/32
```

13.1.3 What to do When a Foreign Host Doesn't Respond

The echo request sent by the **ping** command does not guarantee delivery. A foreign host may fail to respond even after several **ping** commands, because:

- The foreign host may not be listening to the network.
- The foreign host may be inoperative, or a network or gateway leading to the foreign host may be inoperative.
- The foreign host may be slow to respond because of activity.
- The packet may be too large for the foreign host.

Therefore, you should issue the **ping** command more than once before you assume that a communication failure has occurred.

You can use additional **ping** commands to communicate with other foreign hosts in the network to determine the condition that is causing the communication failure. However, you must know the network topology to determine the location of the failure. Issue **ping** commands until the failure is located:

1. Send a **ping** command to your local host.
   - If the PING is unsuccessful, the problem is probably with your host.
   - If the PING is successful, go to step 2.
2. Send a **ping** command to a host (other than your local host) on your local network.
   - If the PING is unsuccessful, the problem is probably with your network.
   - If the PING is successful, go to step 3.
3. Send a **ping** command to the gateway leading to the network in question.
   - If the PING is unsuccessful, continue to test along the network from the target until you find the point of the communication breakdown.
   - If the PING is successful, go to step 4.
4. Send a **ping** command to another host on the same network as the original host.
   - If the PING is unsuccessful, the problem is probably with the other network.
   - If the PING is successful, the original host is either inoperative or is not listening to the network.

13.2 Monitoring the Availability of a Group of Hosts

PMPING is a Presentation Manager (PM) program that displays the status of a user-defined list of hosts using ICMP echo requests (PING). It is useful for continuous monitoring of important network elements, such as bridges or routers.

Subtopics
13.2.1 Creating a List of Hosts to be Monitored
13.2.2 Accessing PMPING

13.2.1 Creating a List of Hosts to be Monitored

PMPING uses the PINGHOST.LST file, located in the ETC subdirectory, to determine which hosts to monitor. This file contains the list of hosts to be continuously monitored and a description of each host.

During installation, a sample PINGHOST.LST file is placed in your ETC subdirectory. You can either use the TCP/IP Configuration notebook or manually edit this file to add the information for the hosts you want to monitor. You can list up to 300 hosts to be monitored.

The format of each line in the file is:

```
      host_ip_address    description
```

Each entry must start on a new line. The sequence of the entries is not important. The fields of each entry must be separated by one or more spaces. The first field is the IP address of the host being monitored. The second field is a description of up to 40 characters. Comments must begin with a "#" in column 1.

The following is an example of the PINGHOST.LST file:

```
      9.67.30.100     **Nameserver-Call_Dan
      9.67.22.1        RALVMM_via_3172-Call_IS
      # This is a comment line.
```

13.2.2 Accessing PMPING

To access PMPING, either:

- Select the **PM Ping** icon in the TCP/IP folder.

- At an OS/2 command prompt, enter:

  ```
  pmping
  ```

  The **pmping** command has no parameters.

Subtopics
13.2.2.1 Starting the Monitoring Process
13.2.2.2 Setting the Timer
13.2.2.3 Ending the Monitoring Process
13.2.2.4 Exiting PMPING

13.2.2.1 Starting the Monitoring Process

To begin monitoring, select **Ping_all** from the menu bar, then select **Start** from the pull-down menu.

Once you select **Start**, the message "Pinging..." is displayed. The results are displayed in a color-coded list (red indicates no response; black indicates that a response was received). Figure 45 shows a sample of the PMPING display.

PICTURE 45

Figure 45. Sample PMPING Display

For each host in the file PINGHOST.LST, the following is displayed:

- The IP address of the host
- Comments describing the host
- The response time in milliseconds (msec) or "No Response"

If the PMPING window is minimized to an icon, the icon changes color to indicate the current status (green if all hosts responded to the PING, or red if one or more hosts did not respond to the PING).

At the top left of the screen, the PMPING color-coded icon is displayed. When all hosts respond to the PING, the icon is green. If one or more hosts do not respond to the PING, the icon turns red.

To the right of the icon, the following text is displayed:

```
Results Displayed:Start hh:mm:ss Stop hh:mm:ss
```

You can use this information to determine both the timeliness of the status being displayed and how long it takes to PING the list of hosts you have defined.

13.2.2.2 Setting the Timer

You can specify how often the list of hosts should be monitored by changing the value of a timer. This value is the amount of time PMPING waits before reissuing the **ping** command.

For example, if you are monitoring hosts A and B and have selected a timer value of 1 minute, PMPING will query host A and host B, display the results, and then wait 1 minute. At the end of 1 minute, the cycle is repeated.

To set the timer:

1. Select the **Timer** option from the Setup pull-down menu. A cascaded pull-down is displayed.

2. Select one of the available timer values (10 seconds, 30 seconds, 1 minute, or 5 minutes).

13.2.2.3 Ending the Monitoring Process

To end the monitoring process, select the **Cancel** option from the Ping_all pull-down menu.

13.2.2.4 Exiting PMPING

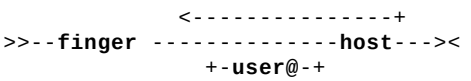
To exit PMPING, select the **Exit** option from the Ping_all pull-down menu.

13.3 Displaying Foreign Host User Information

TCP/IP for OS/2 provides a utility, FINGER, that allows you to display information about users on a foreign host. You can specify a single user or all of the users on a foreign host.

FINGER requires that the FINGER server on the foreign host be capable of responding to a request for information. In TCP/IP for OS/2, FINGER supports the client functions only.

The syntax of the **finger** command is:



Displaying finger Help

>>--**finger**-- **-?**--><

user@ Specifies the user name to be queried on the remote host. This parameter is optional; however, if you specify a user, the host must be followed by an @. Without **user@**, the **finger** command displays all users currently logged on at the host. With **user@**, only detailed information about the user will be displayed.

Note: A space is required between multiple **user@host** entries.

host Specifies a host from which you request user information. This parameter is required.

-? Displays help information.

One or more entries of the form **user@host** are required, with one or more spaces between them.

Subtopics
13.3.1 Examples

13.3.1 Examples

You can use the **finger** command to display information about users on a foreign host as shown in the following examples.

1. To display information about user **jones** at host **coll.univ.edu**, enter:

   ```
   finger jones@coll.univ.edu
   ```

   A sample response is:

   ```
   [jones@coll.univ.edu]
   Login name: jones                         In real life: Jones Beach
   Site info: E103-503,Phone (123)456-7890
   Directory: /u/jones               Shell: /bin/ksh
   On since Jul 25 13:52:27 on pts/2 from maytag.tcpipdev.
   2 minutes 1 second Idle Time
   ```

2. To display information about all users logged on to foreign host **coll.univ.edu**, enter:

   ```
   finger coll.univ.edu
   ```

   A sample response is:

   ```
   [coll.univ.edu]
   Login       Name              TTY  Idle    When           Office
   Andrew   Andy Andrews          hf      5 Fri 07:04  AVW1125      555-1234
   Jeff     Jeff Robins           p0   3:28 Thu 06:39  AVW1127      555-5678
   Norma    Norma Norman          p1     1d Sat 21:48  AVW4117      555-9012
   Bill     Bill Williams         p2     49 Fri 06:44  3215         555-3456
   Patrice  Pat Patrick           p3      1 Sun 13:43  2141         555-7890
   ```

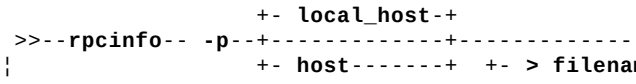
13.4 Obtaining RPC Server Status Information

TCP/IP for OS/2 provides a utility, RPCINFO, that generates a status report for RPC programs and services registered on a remote or local host.

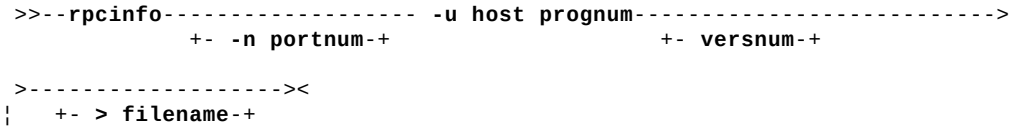
When RPCINFO starts, it reads the RPC file in the ETC subdirectory. The RPC file contains a list of server names and their corresponding RPC program numbers and aliases.

The syntax of the **rpcinfo** command is:

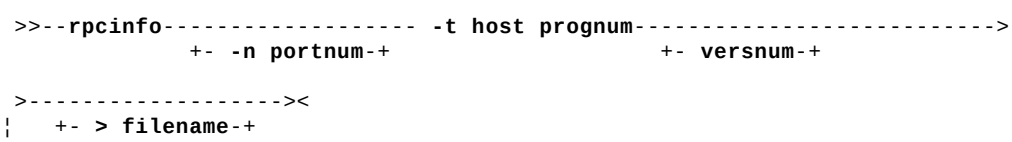
rpcinfo for a Host



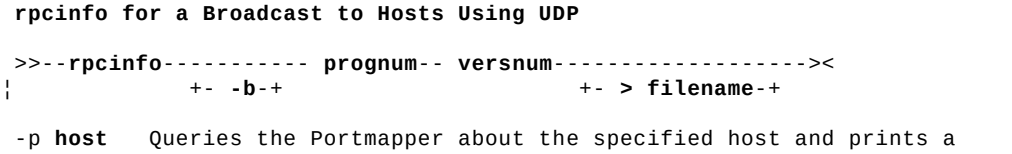
rpcinfo for a Host Using UDP



rpcinfo for a Host Using TCP



rpcinfo for a Broadcast to Hosts Using UDP



-p **host** Queries the Portmapper about the specified host and prints a list of all registered RPC programs. If the host is not specified, the system defaults to the local host name.

> filename
Specifies a file to which to redirect the list of registered RPC programs.

-n **portnum**
Specifies the port number to be used for the -t and -u parameters. This value replaces the port number that is given by the Portmapper.

-u **host prognum versnum**
Sends an RPC call to procedure 0 of **prognum** and **versnum** on the specified host using UDP and reports whether a response is received.

-t **host prognum versnum**
Sends an RPC call to procedure 0 of **prognum** and **versnum** on the specified host using TCP and reports whether a response is received.

-b **prognum versnum**
Sends an RPC broadcast to procedure 0 of the specified **prognum** and **versnum** using UDP and reports all hosts that respond.

Note: The **prognum** parameter can be either a name or a number. If you specify a **versnum**, the **rpcinfo** command attempts to call that version of the specified program. Otherwise, the **rpcinfo** command attempts to find all the registered version numbers for the program you specify by calling version 0. RPCINFO then tries to call each registered version.

Subtopics
13.4.1 Examples

13.4.1 Examples

You can use the **rpcinfo** command to obtain various information as shown in the following examples.

1. To display RPC services registered on the local host, enter:

   ```
   rpcinfo -p
   ```

2. To display the RPC services registered on the remote host named **charm**, enter:

   ```
   rpcinfo -p charm
   ```

   A sample response is:

   ```
    program vers proto   port
     100000    2   tcp    111  portmapper
     100000    2   udp    111  portmapper
     300201    1   tcp   1024
   33333332    1   tcp   1025
     100003    2   udp   2049  nfs
     100005    1   udp    793  mountd
     100005    1   tcp    795  mountd
     100024    1   udp    627  status
     100024    1   tcp    629  status
     300082    1   udp    632
     300082    1   tcp    634
     100021    1   tcp    885  nlockmgr
     100021    1   udp    887  nlockmgr
     100021    3   tcp    890  nlockmgr
     100021    3   udp    892  nlockmgr
     100020    1   udp    895  llockmgr
     100020    1   tcp    897  llockmgr
     100021    2   tcp    900  nlockmgr
   ```

3. To display the status of a particular RPC program on the remote host named **charm**. You can issue the **rpcinfo** command specifying either a program number (100003) or a program name (NFS):

   ```
   rpcinfo -u charm 100003
   ```

   or

   ```
   rpcinfo -u charm nfs
   ```

   A sample response is:

   ```
   Program 100003 Version 2 ready and waiting
   ```

   Note: The program number (100003) corresponds to the program name (NFS) as shown in Example 2.

4. To display all hosts on the local network that are running a particular version of a specific RPC server (for example, Version 2 of the NFS daemon), enter:

   ```
   rpcinfo -b 100003 2
   ```

   or

   ```
   rpcinfo -b nfs 2
   ```

   You can specify either a program number (100003) or a program name (NFS).

   A sample of the response is:

   ```
   9.67.97.166 ebbinghaus.raleigh.ibm.com
   9.67.111.121 pmx6000.tcp.raleigh.ibm.com
   9.67.97.39 kkhp1.raleigh.ibm.com
   9.67.97.166 ebbinghaus.raleigh.ibm.com
   9.67.98.247 iddaix03.raleigh.ibm.com
   9.67.96.198 jollymon.raleigh.ibm.com
   ```

```
¦ 9.67.97.98 trident.raleigh.ibm.com                                          ¦
¦ 9.67.96.196 gobraves.raleigh.ibm.com                                        ¦
¦ 9.67.99.82 bogey.raleigh.ibm.com                                            ¦
¦ 9.67.96.144 deborah.raleigh.ibm.com                                         ¦
¦ 9.67.98.247 iddaix03.raleigh.ibm.com                                        ¦
¦ 9.67.96.61 kix5.raleigh.ibm.com                                             ¦
¦ 9.67.97.98 trident.raleigh.ibm.com                                          ¦
¦ 9.67.97.161 orion.raleigh.ibm.com                                           ¦
¦                                                                             ¦
```

Note: Because this command can return extensive data, you might want to redirect the output to a file.

13.5 Using Packet Tracing Programs

TCP/IP for OS/2 provides a set of utilities that you can use to trace packets. Tracing packets is useful in diagnosing problems at your workstation. These utilities allow you to:

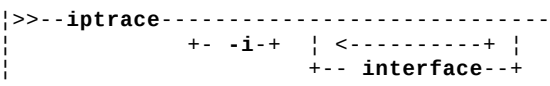
- Initiate a trace
- Convert trace information to human-readable output
- Convert trace information to network analyzer data

Subtopics
13.5.1 Initiating a Trace
13.5.2 Converting Trace Information

13.5.1 Initiating a Trace

The IPTRACE utility traces all packets received from and sent to an ethernet or token-ring adapter.

The syntax of the **iptrace** command is:



-i Specifies that only IP packets are to be traced. The default is to include all information (such as hardware type). Certain interfaces (for example, X25, snal) require this parameter.

interface Specifies an interface to be traced (for example, lan0, sl0). If not specified, all interfaces are traced.

Notes:

1. IPTRACE writes data to the IPTRACE.DMP file in the directory from which you initiated the **iptrace** command. IPTRACE records all traffic sent and received, but does not check for sufficient disk space to record that information. As a result, running IPTRACE can impact your workstation's performance as the IPTRACE.DMP file continues to grow. Therefore, you should use the IPTRACE utility in a limited fashion so that it does not impact the performance of your workstation.

2. IPTRACE is not a network monitor. It can trace only data received by and sent from the specified interfaces.

3. IPTRACE provides a time stamp, recording when the packet was sent or received.

4. To stop IPTRACE, press Enter.

5. Use IPFORMAT to convert the IPTRACE.DMP file into a user-readable format.

To end the IPTRACE, hold down the Ctrl key and press the C key.

The data is written to a file, IPTRACE.DMP, which is placed in the directory from which IPTRACE was started.

Note: After you start IPTRACE, all adapters at the workstation are traced; you cannot select one adapter if several are installed.

13.5.2 Converting Trace Information

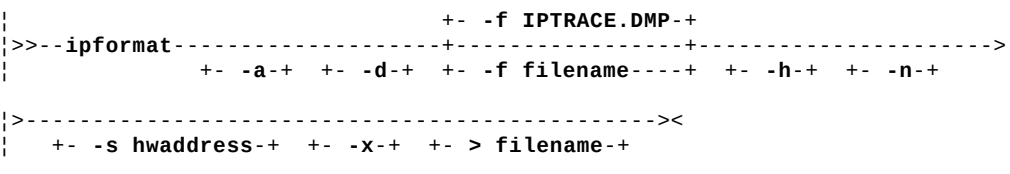
The IPFORMAT utility converts the data in the IPTRACE.DMP file and the SLIPTRC.DMP file to either:

- Human-readable format, which is displayed to the screen
- Data to be used as input to a network analyzer

If you choose to convert the data into human-readable format, you can redirect the output to a file.

IPFORMAT reads the header information in the trace to determine the type of packet received; for example, either token-ring (TRC) or ethernet (ENC). It then separates the data by the IP, TCP, UDP, and ICMP layers. The rest of the packet is displayed as hexadecimal output. If IPFORMAT cannot determine the type of packet received, it creates a hexadecimal representation of the data.

The syntax of the **ipformat** command is:



Displaying ipformat Help

-a Specifies that IPFORMAT should not format ARP or RARP packets.

-d Specifies that IPFORMAT should not display the data portion of a packet.

-f filename
Specifies the input fine name. The default is IPTRACE.DMP.

-h Specifies that IPFORMAT should display the raw data packet after the formatted information.

-s hwaddress
Format data only for the specified hardware address (**hwaddress**). Specify the hardware address as a 12-digit hexadecimal address for the Ethernet or token-ring adapters.

Note: You can use the **netstat -n** command to display the hardware address.

-n Specifies that IPFORMAT should not display hexadecimal data for unknown data types.

-x Converts IPTRACE data to a format that can be read by a Network General Sniffer**.

> filename
Redirect the output to the specified file.

-? Displays help information.

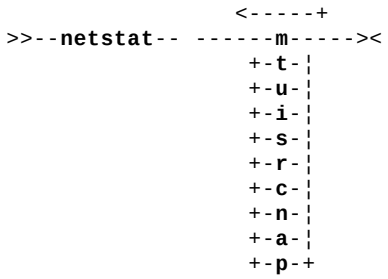
13.6 Displaying Information about Your Local Host

TCP/IP for OS/2 provides a utility, NETSTAT, that allows you to display the following information about your local host:

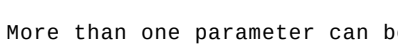
- The number of interfaces present, a description of these interfaces, and their current status
- Statistics on packets for the IP, ICMP, TCP, and UDP layers
- Routing and connection information
- Local and foreign port numbers of sockets and their state

Much of the information available from NETSTAT is also available from SNMP. However, NETSTAT provides additional information about memory buffers and sockets, while SNMP provides status information, system description information, and additional statistics for the protocol and layer information.

The syntax of the **netstat** command is:



Displaying netstat Help



More than one parameter can be specified with the **netstat** command. Do not enter spaces between multiple parameters because any entry after a space is ignored.

-m	Information about memory buffer usage
-t	Information about TCP connections
-u	Information about UDP statistics
-i	Information about IP statistics
-s	Information about sockets
-r	Routing tables and corresponding network interfaces
-c	Information about internet control message protocol (ICMP) statistics
-n	Information about LAN interfaces

¦ **Note:** If you start SLIP at 76800 baud using the SIO.SYS and ¦ VSIO.SYS COM port drivers, the **netstat -n** command will ¦ return an incorrect response of 1200 BPS for the SLIP ¦ speed.

-a	The address of the network interfaces
-p	Contents of the address resolution protocol table
-?	Help information

Subtopics
13.6.1 Examples

13.6.1 Examples

You can use the **netstat** command to obtain network status information as shown in the following examples.

1. To simultaneously display the status of memory buffers, TCP connections, and UDP statistics, you can enter multiple parameters:

   ```
   netstat -mtu
   ```

   instead of requesting specific output.

2. To display information about the TCP connections that were initiated, accepted, established, and dropped, enter:

   ```
   netstat -t
   ```

   Information about the packets, bytes, and time-outs for the TCP connections is also displayed.

   A sample response is:

   ```
   TCP STATISTICS .... TYPE ANY KEY TO CONTINUE
   connections initiated             2
   connections accepted              2
   connections established           2
   connections dropped               2
   embryonic connections dropped     2
   conn. closed (includes drops)     3
   segs where we tried to get rtt    6
   times we succeeded                2
   delayed acks sent                 3
   conn. dropped in rxmt timeout     0
   retransmit timeouts               1
   persist timeouts                  1
   keepalive timeouts                3
   keepalive probes sent             3
   connections dropped in keepalive  1
   total packets sent                8
   data packets sent                 4
   data bytes sent                   2
   data packets retransmitted        4
   data bytes retransmitted          2
   TYPE ANY KEY TO CONTINUE
   ```

3. To display information about IP statistics, enter:

   ```
   netstat -i
   ```

   A sample response is:

   ```
   total packets received 3694
   checksum bad 0
   packet too short 0
   not enough data 0
   ip header length < data size 0
   ip length < ip header length 0
   fragments received 0
   frags dropped (dups, out of space) 0
   fragments timed out 0
   packets forwarded 2027
   packets rcvd for unreachable dest 1
   packets forwarded on same net 0
   requests for transmission 3261
   output packets discarded because no route could be found 0
   input packets delivered successfully to user-protocols 1654
   input packets with an unknown protocol 12
   output packets successfully fragmented 0
   output fragments created 0
   fragmentation failed 0
   successfully assembled packets 0
   ```

4. To display routing table information, enter:

   ```
   netstat -r
   ```

You can use this command to verify your setup if you encounter problems in trying to reach networks. The table determines the networks that are connected and specifies where to send packets destined for networks that are not directly connected.

A sample response is:

```
destination        router     refcnt    use    flags     snmp     intrf
                                                        metric

    default      9.67.22.2      0        3       U        -1      lan0
   67.0.0.0     9.67.30.57      0        0       U        -1      lan0
   9.67.0.0     9.67.30.57      0       10       U        -1      lan0
```

5. To display information about network interfaces, enter:

   ```
   netstat -n
   ```

A sample response is:

```
Interface 0: IBM Token-Ring Network Adapter/A
physical address    10005a2b891d      MTU 1500

unicast packets received 810
broadcast packets received 18950
total bytes received 1560685
unicast packets sent 3242
broadcast packets sent 2
total bytes sent 427750
packets discarded on transmission 0
packets discarded on reception 0
received packets in error 0
errors trying to send 0
packets received in unsupported protocols 12




 Press any key to continue
```

Press any key to continue scrolling the network interface data:

Note: If you start SLIP at 76800 baud using the SIO.SYS and VSIO.SYS COM port drivers, the **netstat -n** command will return an incorrect response of 1200 BPS for the SLIP speed.

6. To display the contents of the Address Resolution Protocol table enter,:

   ```
   netstat -p
   ```

A sample response is:

```
          ARP table contents:

interface     hardware address         IP address   minutes since
                                                      last use
    0          50005a1a15f3         129.34.217.254     0
    1          50004a1123f3         129.34.216.254     0
```

13.7 Obtaining Network Status Information

TCP/IP for OS/2 provides a set of Simple Network Management Protocol (SNMP) utilities that you can use to obtain status information about your network.

This section provides an overview of SNMP and describes how to establish an SNMP agent, how establish an SNMP client, and how to use the SNMP commands to manage your network.

Subtopics

13.7.1 Overview of SNMP

SNMP is a protocol used by network elements (for example, hosts and bridges) to exchange information relevant to network management. It is available on a wide variety of platforms from both IBM and non-IBM sources. For example, SNMP is available from IBM for OS/2, DOS, VM, and MVS.

SNMP is defined in RFCs. The RFCs that define SNMP include RFC 1213, RFC 1157, and RFC 1155. For more information about RFCs, see "Obtaining Requests for Comments (RFCs)" in topic PREFACE.7.1.

SNMP requires one or more clients to manage the network as well as an SNMP agent on each host being managed.

¦The "Bibliography" in topic BIBLIOGRAPHY contains recommended reference ¦materials about SNMP.

Subtopics
13.7.1.1 SNMP Client
13.7.1.2 SNMP Agent
13.7.1.3 Other SNMP Clients
13.7.1.4 Other SNMP Agents

13.7.1.1 SNMP Client

TCP/IP for OS/2 provides SNMP client (manager) support that enables the client to:

- Request the following information:
  - A description of the system, where it is, and who is responsible for it
  - The number of interfaces present, a description of these interfaces, and their current status (up or down)
  - Statistics about packets for the IP, ICMP, TCP, and UDP layers
  - Routing and connection information
- Set the values of the parameters on the agent

13.7.1.2 SNMP Agent

TCP/IP for OS/2 provides SNMP agent (managed host) support that enables the agent to:

- Send management information base (MIB-II) information in response to requests from the client. The following MIB objects are supported by the OS/2 SNMP agent:

  - System Group
  - Interfaces Group
  - Address Translation Group
  - IP (Internet Protocol) Group
  - ICMP (Internet Control Message Protocol) Group
  - TCP (Transmission Control Protocol) Group
  - UDP (User Datagram Protocol) Group

- Send unsolicited notification of significant events (TRAPs) to the client. The following TRAPs are generated by the OS/2 agent:

  - coldStart
  - authenticationFailure
  - enterpriseSpecific

13.7.1.3 Other SNMP Clients

To manage your OS/2 SNMP agent from a client other than an OS/2 SNMP client, you should:

- Configure the OS/2 SNMP agent as described in "Establishing the OS/2 SNMP Agent" in topic 13.7.4.
- Review the documentation accompanying your SNMP client for setup information.

13.7.1.4 Other SNMP Agents

To manage agents other than OS/2 SNMP agents, configure those agents with a community name and set them up to send TRAPs to your OS/2 SNMP client. This procedure depends on the particular SNMP agent you are managing. You should review the documentation accompanying your SNMP agent.

For example, to send TRAPs from the IBM VM SNMP agent to your OS/2 SNMP client:

- Create the file SNMPTRAP DEST on the VM host.
- Specify your OS/2 SNMP client host name or IP address in this file.

See "Defining a List of SNMP Clients for TRAPs" in topic 13.7.4.3 for more information.

13.7.2 Establishing the OS/2 SNMP Client

To establish an OS/2 SNMP client, you must establish a database of management information for the client. This database is called a management information base (MIB).

Subtopics
13.7.2.1 Management Information Base (MIB)

13.7.2.1 Management Information Base (MIB)

The Management Information Base (MIB) defines management information obtained from SNMP agents. The MIB defines objects such as the description of the system being managed, packets received in error, and the status of an interface.

MIB objects can be described in two ways:

- Using an English-like textual notation. For example, sysDescr (system description).

- Using Abstract Syntax Notation.1 (ASN.1). For example, 1.3.6.1.2.1.1.1. is the ASN.1 equivalent of sysDescr. Requests to obtain the value of an MIB object are sent to an SNMP agent using ASN.1 notation.

The following is a list of the supported MIB objects:

- System Group
- Interfaces Group
- Address Translation Group
- IP (Internet Protocol) Group
- ICMP (Internet Control Message Protocol) Group
- TCP (Transmission Control Protocol) Group
- UDP (User Datagram Protocol) Group
- EGP (Exterior Gateway Protocol) Group
- Transmission Group
- SNMP Group

Some of these MIB objects are members of a table. For example, the Interfaces Table is a two-dimensional array of MIB objects related to the interfaces installed. This array contains information such as the description and the speed of each interface. There may be several instances of a particular object within the table. For example, there would be a description of interface number 1, interface number 2, and so on.

Some MIB objects are scalars, which means that there is only one instance of that particular object. For example, there is only one system description.

Logically related MIB objects are placed into groups. A group can contain both scalars and tables. For example, the Interfaces Group contains a scalar (the object ifNumber, which is the number of interfaces present) and a table (the Interfaces Table).

13.7.3 Customizing SNMP for Your Network

During installation, the MIB2.TBL file is placed in the directory defined by the ETC environment variable in your CONFIG.SYS file.

You can modify this file as needed for vendor-specific MIB objects, and save the new file in the directory defined by the ETC environment variable in your CONFIG.SYS file.

Note: You do not need to modify the MIB2.TBL file unless you are managing SNMP agents that have objects not defined on the command line or in the internal tables.

Before adding MIB objects to the MIB2.TBL, you should have the following information:

- The textual name for each object (for example, sysDescr)
- The ASN.1 notation for each object (for example, 1.3.6.1.2.1.1.1.)
- The syntax (data type) for each object (for example, display)

Each line in this file has the following format:

```
      textual_name          asn.1_name         syntax
```

For example:

```
      sysDescr        1.3.6.1.2.1.1.1.      display
```

Note: You do not specify instances in the MIB2.TBL file, as you would when issuing a command such as:

```
      snmp set sysDescr.0  1.3.6.1.2.1.1.1.0  display
```

You can either use the TCP/IP Configuration notebook or manually edit this file to add vendor-specific entries. Entries do not have to be in a specific order, however, each entry must start on a new line and each field in the line must be separated by one or more spaces.

The following is a sample MIB2.TBL file:

```
      sysDescr               1.3.6.1.2.1.1.1.         display
      sysObjectID            1.3.6.1.2.1.1.2.         object
      sysUpTime              1.3.6.1.2.1.1.3.         ticks
      sysContact             1.3.6.1.2.1.1.4.         display
      sysName                1.3.6.1.2.1.1.5.         display
      sysLocation            1.3.6.1.2.1.1.6.         display
      sysServices            1.3.6.1.2.1.1.7.         number
      ifNumber               1.3.6.1.2.1.2.1.         number
      ifIndex                1.3.6.1.2.1.2.2.1.1.     number
      ifInOctets             1.3.6.1.2.1.2.2.1.10.    counter
      ifInUcastPkts          1.3.6.1.2.1.2.2.1.11.    counter

                                      .
                                      .
                                      .

      ifSpeed                1.3.6.1.2.1.2.2.1.5.     gauge
      ifPhysAddress          1.3.6.1.2.1.2.2.1.6.     string
      ifAdminStatus          1.3.6.1.2.1.2.2.1.7.     number
      ifOperStatus           1.3.6.1.2.1.2.2.1.8.     number
      ifLastChange           1.3.6.1.2.1.2.2.1.9.     ticks
      atIfIndex              1.3.6.1.2.1.3.1.1.1.     number
      atPhysAddress          1.3.6.1.2.1.3.1.1.2.     string
      atNetAddress           1.3.6.1.2.1.3.1.1.3.     internet

                                      .
                                      .
                                      .
```

Subtopics

13.7.3.1 The Textual_Name Field

This field contains the name of the MIB object that is entered by the end-user. The MIB2.TBL file for the OS/2 SNMP client contains the textual names as defined in RFC 1213. You can modify the textual names of the MIB objects in your MIB2.TBL file and still have the agent respond properly. For example, you could change **sysDescr** to **systemDescription**. As long as the asn.1_name field is correct, the requested value is correct.

However, for consistency with RFC 1213 conventions, these names should not be changed.

13.7.3.2 The ASN.1_Name Field

This field represents the object identifier of the MIB object in ASN.1 notation. This value is sent to the SNMP agent as part of an **snmp get**, **snmp getnext**, **snmp set**, or **snmp walk** command.

The following rules apply to object IDs in ASN.1 notation:

- The object ID consists of 1 to 128 subIDs, which are separated by dots.
- Each subID is a positive number. No negative numbers are allowed.
- The value of each number cannot exceed 4294967295 (2 to the power of 32 minus 1).
- The valid values of the first subID are: 0, 1, or 2.
- If the first subID has a value of 0 or 1, the second subID must be in the range of 0 through 39.

13.7.3.3 The Syntax Field

The syntax of an object defines the data type of the object. It identifies the structure corresponding to object types. For a complete description of all data types and their meanings, see RFC 1155. The data types listed below are supported by the MIB2.TBL file (and used by the **snmp get**, **snmp getnext**, **snmp set**, and **snmp walk** commands. When adding new objects to the MIB2.TBL, the following values that should be used in the syntax field:

Integer or **Number**
A 32-bit numeric value. For integer objects, specify **number** in the MIB2.TBL syntax field:

```
ifNumber  1.3.6.1.2.1.2.1.  number
```

String or **Octet String**
A string of octets. Each byte in an octet string can take any value from 0 to 255. For Octet String objects, specify **string** in the MIB2.TBL syntax field:

```
ifPhysaddress  1.3.6.1.2.1.2.2.1.6  string
```

Object or **Object Identifier**
An authoritative identification of an object. This is the ASN.1 notation. For Object Identifier objects, specify **object** in the MIB2.TBL syntax field:

```
sysObjectID  1.3.6.1.2.1.1.2.  object
```

IpAddress or **Internet**
The IpAddress is a 32-bit internet address, represented as a string of eight components (octets) each with a length four bytes, in network byte order. For IpAddress objects, specify **internet** in the MIB2.TBL syntax field:

```
ipRouteDest  1.3.6.1.2.1.4.21.1.1  internet
```

Counter
A non-negative integer that increases by one until it reaches a maximum value, at which time it resets to zero and starts increasing again. For Counter objects, specify **counter** in the MIB2.TBL syntax field:

```
ifInOctets 1.3.6.1.2.1.2.2.1.10  counter
```

Gauge
A non-negative integer that can increase or decrease, but which latches at a maximum value. For Gauge objects, specify **gauge** in the MIB2.TBL syntax field:

```
iSpeed   1.3.6.1.2.1.2.2.1.5  gauge
```

Ticks or **TimeTicks**
A non-negative integer that counts the time in hundredths of a second since an event. For TimeTicks, objects specify **ticks** in the MIB2.TBL syntax field:

```
sysUptime   1.3.6.1.2.1.1.3.  ticks
```

Display
The same as an octet string, but is limited to the ASCII character set. It contains human-readable characters. For Display objects, specify **display** in the MIB2.TBL syntax field:

```
sysDescr.0  1.3.6.1.2.1.1.1.0  display
```

Opaque
A string of octets. Data is encoded and transmitted as an octet string, allowing you to pass arbitrary data. For Opaque objects, specify **opaque** in the MIB2.TBL syntax field:

```
sysUnspt.0 opaque f57dac
```

Null
A placeholder in the MIB2.TBL; it does not contain data.

13.7.3.4 How SNMP Works (Search Order)

The search order for the **snmp** and **snmpgrp** commands is:

1. The command line. Searching is not necessary if the command line contains all required information.

   Examples:

   a. A **get** command that specifies the ASN.1 notation for that object:

   ```
   snmp get 1.3.6.1.2.1.1.1.0
   ```

   b. A **set** command that specifies both the ASN.1 notation and the syntax (data type) for that object:

   ```
   snmp set 1.3.6.1.2.1.1.1.0 display foo
   ```

2. Internal MIB, defining the mapping between an object's ASN.1 notation and an object's textual notation. If the information is not specified as part of the command, the OS/2 SNMP client searches for that object in the internal table and uses the corresponding ASN.1 notation and syntax (data type) in the SNMP request to an SNMP agent.

   Examples:

   a. A **snmp get** command, specifying the object name using the textual description (instead of the ASN.1 notation):

   ```
   snmp get sysDescr.0
   ```

   b. A **set** command specifying the ASN.1 notation without a syntax (data type) for that object:

   ```
   snmp set 1.3.6.1.2.1.1.1.0 foo
   ```

3. MIB2.TBL. If the MIB variable does not exist in the internal table, SNMP searches the MIB2.TBL file for the information.

   Example:

   If you issue the following **snmp get** command specifying a MIB variable not defined in the supported RCFs:

   ```
   snmp get sysUnspt.0
   ```

See "Customizing SNMP for Your Network" in topic 13.7.3 for the more information about the MIB.2 TBL file.

The internal table contains all the textual names defined in the following RFCs:

- RFC1155
- RFC1213
- RFC1231
- RFC1285
- RFC1315
- RFC1398

13.7.4 Establishing the OS/2 SNMP Agent

This section describes how you can establish an OS/2 SNMP agent by:

1. Defining the contact information in your CONFIG.SYS file.

2. Defining a community name for this agent by creating and scrambling the PW.SRC file.

3. Defining a list of SNMP clients to which TRAPs generated by the OS/2 SNMP agent are sent, by creating the SNMPTRAP.DST file.

4. Starting the SNMP agent.

Subtopics

13.7.4.1 Defining Contact Information

The following MIB objects for an SNMP agent must be defined in your CONFIG.SYS file:

```
SET SYSCONTACT=contact_information
SET SYSLOCATION=contact_location
```

Where:

contact_information
specifies the contact person for this managed node, along with information about how to contact the person.

For example, if Bill Smith at telephone extension 389 is the contact person for this node, you would add the following line to your CONFIG.SYS file:

```
SET SYSCONTACT=B. Smith, Extension 389
```

contact_location
specifies the physical location of this node.

For example, if the node is physically located in Raleigh, Building 503, Room A145, on the floor tile K-19, you would add the following line to your CONFIG.SYS file:

```
SET SYSLOCATION=Raleigh, Bldg. 503, Room A145, Tile K-19
```

Note: You can define these statements either by using the TCP/IP Configuration notebook or by manually editing your CONFIG.SYS file.

After you define these variables, you must restart your workstation to implement the changes.

13.7.4.2 Defining a Community Name

SNMP agents respond to requests for information from SNMP clients or network management stations using a *community name*. A community name is the authentication mechanism used by SNMP to verify that a request for information is comes from a valid requestor. It is similar to a password. Each request sent to the OS/2 SNMP agent must be accompanied by the correct community name.

The community name must be defined in the PW.SRC file in your ETC subdirectory. TCP/IP for OS/2 provides a command that you can use to scramble the information contained in the PW.SRC file to prevent others from obtaining the actual community names.

The actual community names reside in an unscrambled format in a master file on a secure host.

The format of each line in the PW.SRC file is:

community_name desired_network snmp_mask

When a request is received from an SNMP client, the community name received is checked against the entries in the PW.SRC file:

- If the community name received does not match any of the entries listed in the PW.SRC file, an Authentication Failure TRAP is sent to the SNMP clients listed in the SNMPTRAP.DST file (see "Defining a List of SNMP Clients for TRAPs" in topic 13.7.4.3 for more information).

- If the community name received matches an entry in the PW.SRC file, the *snmp_mask* is applied to the originating IP address of the incoming SNMP request. The result is compared with the **desired_network**, and if they are equal, the request is accepted. This allows the agent to accept requests only from certain clients which can have different community names.

PW.SRC File Example: The following is an example of a PW.SRC file containing multiple entries:

```
passwd1   9.0.0.0       255.0.0.0
passwd2   129.34.81.22  255.255.255.255
```

Using this example (line 1), assume a request from an SNMP client at IP address **9.34.22.122** is received and the correct community name of **passwd1** is supplied:

- The subnet mask of **255.0.0.0**. is applied to the IP address of **9.34.22.122**.

- The result is **9.0.0.0**, which equals the specified **desired_network**.

- The request is accepted.

The second line in the example indicates that requests containing the community name **passwd2** are only accepted from the SNMP client at host **129.34.81.22**. If a request is received from any other client with the **passwd2** community name, an AUTHENTICATION_FAILURE TRAP is sent.

Creating a Scrambled PW.SRC File: TCP/IP for OS/2 does not provide a PW.SRC file. If you want to use the SNMP agent, you must create a PW.SRC file.

To create a protected PW.SRC file:

1. Verify that there is an entry in your CONFIG.SYS file setting the HOSTNAME environment variable. Add the entry if it is not present.

2. Create the PW.SRC file in the directory where the SNMP executable (EXE) files are installed (the default is BIN).

   A **desired_network** and **snmp_mask** of all zeros allows any host with the correct **community_name** to make requests.

   Note: You can use the TCP/IP Configuration notebook to create the PW.SRC file or you can manually create it.

3. Create the scrambled SNMP.PW file by entering the following command:

   ```
   MAKE_PW
   ```

4. Copy the SNMP.PW file to the directory defined by the ETC environment variable in your CONFIG.SYS file.

   To protect a remote SNMP agent, the SNMP.PW file should be created at a secure location and then sent to the remote host for inclusion in the ETC directory.

13.7.4.3 Defining a List of SNMP Clients for TRAPs

TRAPs are unsolicited notifications of network-significant events that are sent from an SNMP agent to an SNMP client.

The SNMP agent uses the SNMPTRAP.DST file in the ETC subdirectory to determine which SNMP clients are to be sent TRAPs. This file contains a list of SNMP clients and identifies User Datagram Protocol (UDP) as the transport protocol used to send TRAPs.

TCP/IP for OS/2 does not provide an SNMPTRAP.DST file. You can use the TCP/IP Configuration notebook to create the SNMPTRAP.DST file or you can manually create it.

The format of each line in the SNMPTRAP.DST file is:

```
hostname UDP
```

Where *hostname* is either the host name or IP address of the SNMP client.

The following is an example of an SNMPTRAP.DST file containing multiple entries:

```
124.34.216.1   UDP
Manager2       UDP
```

13.7.4.4 Starting the SNMP Agent

To start the SNMP agent, enter the following command at the OS/2 command prompt:

```
start snmpd
```

Note: You can start the SNMP agent automatically when TCP/IP starts using the TCP/IP Configuration notebook or by adding the following line to your STARTUP.CMD file:

```
START "SNMPD" SNMPD
```

SNMPD runs as a task until explicitly stopped.

13.7.5 Verifying Your Setup

To verify your setup, you can initiate a series of OS/2 SNMP client commands. These OS/2 SNMP client command exercises are provided to verify that SNMP was installed properly.

These exercises assume:

- The SNMP agent is installed and running.
- The SNMP agent supports the MIB objects **sysDescr** and **ifIndex**.
- The SNMP agent has been properly configured to send the authentication failure TRAP to your OS/2 SNMP client.

These exercises use the following parameters:

- The SNMP agent host name is **quicktest**.
- The SNMP agent community name is **green**.

Subtopics
13.7.5.1 Examples

13.7.5.1 Examples

You can use SNMP client commands to verify your installation as shown in the following examples (see "The snmpgrp Command" in topic 13.7.6.1 for more information).

1. To retrieve an entire group of MIB objects, enter:

   ```
   snmpgrp quicktest green sys
   ```

   All of the supported objects in the System group are displayed.

2. To retrieve an individual MIB object, enter:

   ```
   snmp get quicktest green sysDescr.0
   ```

   The value of **sysDescr.0**, the system description, is displayed.

3. To retrieve an individual MIB object from a table, enter:

   ```
   snmp getnext quicktest green ifIndex.0
   ```

   The value for the first instance of **ifIndex**, which is 1, is displayed.

4. To display unsolicited TRAPs from an SNMP agent, enter:

   ```
   start snmptrap
   ```

   After the SNMPTRAP window appears, select **Get_Traps** from the menu bar, and then select **Start** option.

5. To exercise a SNMPTRAP, enter:

   ```
   snmp get quicktest red sysDescr.0
   ```

   Because **red** is an incorrect community name, the SNMP agent should respond with an authentication failure TRAP, which is displayed on the SNMPTRAP window.

If the agent does not respond to queries from the OS/2 SNMP client, check the following:

- Can you reach the host? Try to PING the host.
- Is the agent operational?
- Did you specify a valid community name?
- Does the agent support the MIB objects you used?

If the authentication failure TRAP was not displayed, check the following:

- Is the agent configured to send TRAPs to your OS/2 SNMP client?
- Does the agent support the Authentication Failure TRAP?
- Is the community name you used incorrect? To generate this TRAP, you should use an incorrect community name.

13.7.6 Displaying Groups and Tables of Management Information

To retrieve logically-related groupings of information from a remote (or local) host running an SNMP agent, use the **snmpgrp** command. This command retrieves the values of the MIB objects belonging to the following groups:

- The System Group
- The Interfaces Group
- The Address Translation Group
- The IP (Internet Protocol) Group
- The ICMP (Internet Control Message Protocol) Group
- The TCP (Transmission Control Protocol) Group
- The UDP (User Datagram Protocol) Group

You can also obtain a greater level of detail by using one of the **snmpgrp** command parameters to obtain the values for tables within a group.

You can use **snmpgrp** to obtain the values of the MIB objects in the following tables:

- From the Interfaces Group
  - Interfaces table
- From the Address Translation Group
  - AT table
- From the IP Group
  - IP address table
  - IP routing table
  - IP address translation table
- From the TCP Group
  - TCP connection table
- From the UDP Group
  - UDP listener table

Notes:

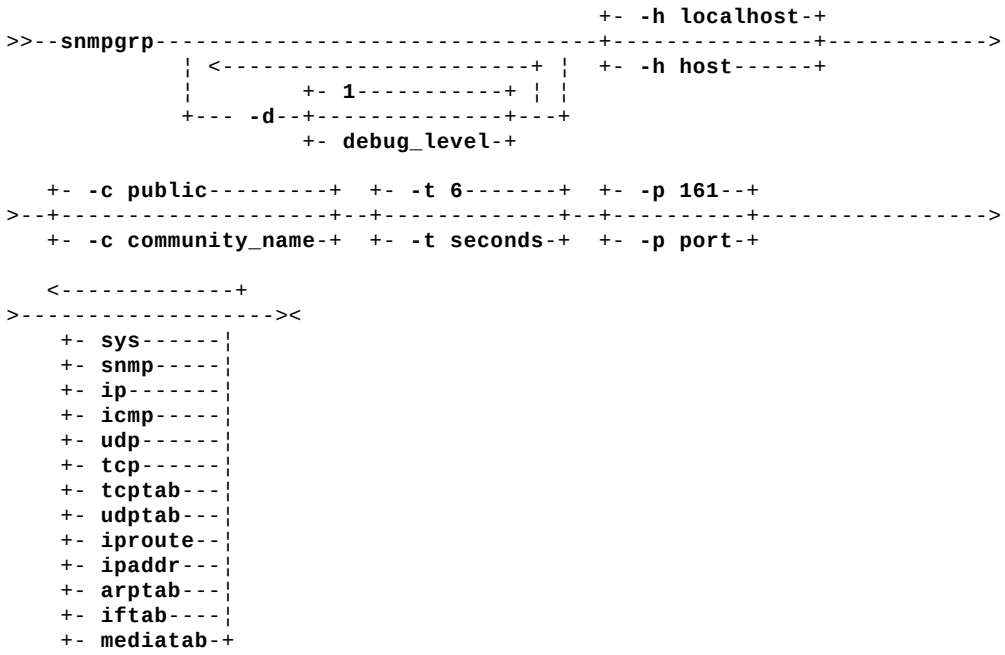
1. For a complete listing of the MIB objects within each group and a description of what each object represents, see Appendix E, "Management Information Base (MIB) Objects."

2. For more information about how the **snmpgrp** command locates the data, see "How SNMP Works (Search Order)" in topic 13.7.3.4.

Subtopics
13.7.6.1 The snmpgrp Command

13.7.6.1 The snmpgrp Command

The syntax of the **snmpgrp** command is:



Displaying snmpgrp Help

-d **debug_level**
Specifies the debug level. There are currently 2 debug levels (1 and 2). Level 2 provides more detailed debug and trace information than does level 1. If you do not specify a debug level, the default is 1.

-h **host** Specifies the host to which you want to send a request. You can specify either an internet protocol address or a host name. If you do not specify a host, the default is your local host.

-c **community_name**
Specifies the community name of the simple network management protocol agent on the destination host. If you do not specify a community name, the default is public.

Note: Community names are case sensitive.

-t **seconds**
Specifies the amount of time (in seconds) that SNMPGRP waits for a reply. The default is 6 seconds.

-p **port** Specifies the destination (agent) port. The default is 161.

Note: The following list represents the names of the groups from which you can choose. If you request more than one group, your output will be displayed in the order in which the groups appear in the list.

sys Requests the System group of MIB variables.

snmp Requests the SNMP group of MIB variables.

ip Requests the scalar MIB variables in the IP group.

icmp Requests the ICMP group of MIB variables.

udp Requests the scalar MIB variables in the UDP group.

tcp Requests the scalar MIB variables in the TCP group.

tcptab Requests the TCP connection table in the TCP group.

udptab Requests the UDP listener table in the UDP group.

iproute Requests the IP routing table in the IP group.

ipaddr Requests the IP address table in the IP group.

```
arptab    Requests the AT table in the address translation group.

iftab     Requests the Interface table in the interfaces group.

mediatab  Requests the IP address translation table in the IP group.

-?        Displays help information.


Multiple parameters can be specified on the command line.


Examples:  You can use the snmpgrp command to retrieve the values of MIB
variables from an SNMP agent.

In the following examples:

   The OS/2 snmp agent is used.
   The host name is wings.
   The community name is ok.

1.  To obtain information about the System Group, enter:

          snmpgrp -h wings -c ok sys

    A sample response is:

     +-------------------------------------------------------------------------------+
     |                                                                               |
     |                                                                               |
     | SYSTEM group --------------------------                                       |
     |                                                                               |
     |       Descr:  IBM RISC System/6000                                            |
     |  Machine Type: 0x0035 Processor id: 000013933500                              |
     |  The Base Operating System AIX version: 03.02.0000.0000                       |
     |  TCPIP Applications version: 03.02.0000.0000                                  |
     |    ObjectId:  1.3.6.1.4.1.2.3.1.2.1.1.2                                       |
     |      UpTime:  78172181 - 9 days, 1 hours, 8 minutes, 41.81 seconds            |
     |     Contact:                                                                  |
     |        Name:  wings.watson.ibm.com                                            |
     |    Location:                                                                  |
     |    Services:  72                                                              |
     |                                                                               |
     |                                                                               |
     +-------------------------------------------------------------------------------+

2.  To obtain information about the Interfaces table, enter:

          snmpgrp -h wings -c ok iftab

    A sample response is:

     +-------------------------------------------------------------------------------+
     |                                                                               |
     |                                                                               |
     |  Interface table ----------------------                                       |
     |                                                                               |
     |             Index: 1                                                          |
     |             Descr: lo0                                                        |
     |              Type: 24                                                         |
     |               Mtu: 1536                                                       |
     |             Speed: 0                                                          |
     |       PhysAddress:                                                            |
     |       AdminStatus: 1                                                          |
     |        OperStatus: 1                                                          |
     |        LastChange: 78198290 - 9 days, 1 hours, 13 minutes, 2.90 seconds       |
     |          InOctets: 55951118                                                   |
     |       InUcastPkts: 326933                                                     |
     |      InNUcastPkts: 0                                                          |
     |        InDiscards: 0                                                          |
     |          InErrors: 0                                                          |
     |   InUnknownProtos: 0                                                          |
     |         OutOctets: 55951118                                                   |
     |      OutUcastPkts: 326933                                                     |
     |     OutNUcastPkts: 0                                                          |
     |       OutDiscards: 0                                                          |
     |         OutErrors: 0                                                          |
     |           OutQLen: 0                                                          |
     |          Specific: 0                                                          |
     |             Index: 2                                                          |
     |             Descr: tr0; Product: not available! Manufacturer: not available!  |
     |  Part Number: not available! FRU Number: not available!                       |
     |              Type: 9                                                          |
     |               Mtu: 1500                                                       |
     |             Speed: 16000000                                                   |
```

```
       PhysAddress: 10:00:4f:a7:00
       AdminStatus: 1
        OperStatus: 1
        LastChange: 78198315 - 9 days, 1 hours, 13 minutes, 3.15 seconds
          InOctets: 167477448
       InUcastPkts: 1167209
      InNUcastPkts: 109571
        InDiscards: 0
          InErrors: 630379
   InUnknownProtos: 630379
         OutOctets: 97435461
      OutUcastPkts: 220088
     OutNUcastPkts: 6
       OutDiscards: 0
         OutErrors: 0
           OutQLen: 0
          Specific: 0
```

Press any key to continue scrolling the interface table data.

3. To obtain information about the UDP listener table in the UDP Group using multiple parameters, enter:

```
snmpgrp -h wings -c ok sys udp udptab
```

A sample response is:

```
UDP group------------------------------

     InDatagrams:  404196
         NoPorts:  107
        InErrors:  0
    OutDatagrams:  189378
```

13.7.7 Displaying and Setting Individual Management Information Variables

You can use the **snmp** command to retrieve and set the values of individual MIB variables from a remote or local host running an SNMP agent. This command allows the OS/2 SNMP client to obtain or set the value of any one of the defined MIB objects, as well as those MIB objects not defined by the supported RFCs.

For more information about how the **snmp** command locates the data, see "How SNMP Works (Search Order)" in topic 13.7.3.4.

There are many reasons a network manager would want to obtain or set the values of an individual MIB variable. Two common reasons are:

- Only some specific information is needed, not an entire group or table.

  The network manager might need to know the current status of a particular interface. Rather than obtaining the entire interfaces group, the value of the MIB object ifOperStatus can be quickly obtained.

- The information is not defined in the supported RFCs.

  You can extend the SNMP MIB through the use of enterprise-specific MIB objects. This allows individual vendors to define management information unique to their implementation. For example, IBM has defined a MIB object, DPI_port, which is unique to the IBM SNMP implementation.

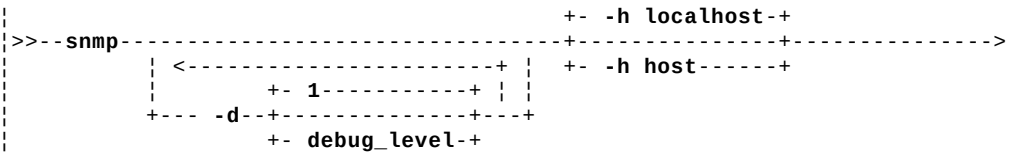
In addition to enterprise-specific MIB objects, the OS/2 Distributed Program Interface (DPI) allows an end-user to add management information objects to the OS/2 SNMP Agent. For more information about DPI, see *IBM TCP/IP Version 2.0 for OS/2: Programmer's Reference*.

Subtopics
13.7.7.1 The snmp Command

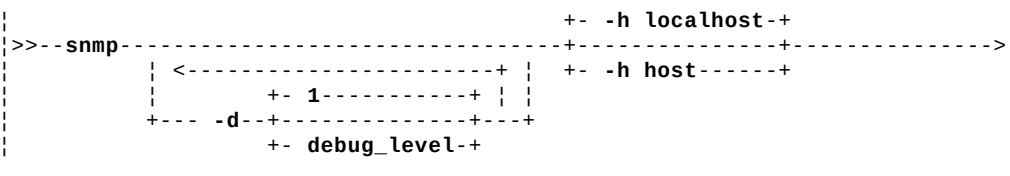
13.7.7.1 The snmp Command

The syntax of the **snmp** command is:

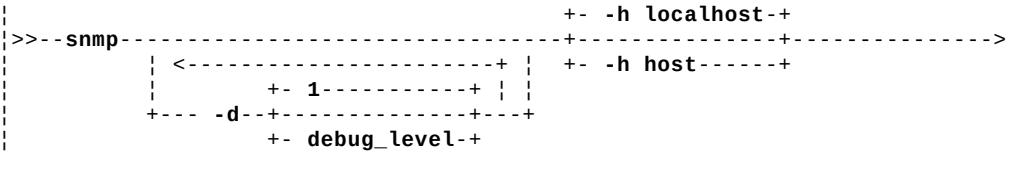
Receiving Traps



Setting MIB Variable Values



Gathering Information about MIB Variables



Displaying snmp Help

-d Starts debugging.

debug_level
Specifies the debug level. There are currently 2 debug levels (1 and 2). Level 2 provides more detailed debug and trace information than does level 1.

-h **host**
Specifies the destination host to which you want to send a request. This can be either an internet protocol address or a host name. If you do not specify a host, the default is your local host.

-c **community_name**
Specifies the community name used to access the specified variables on the destination SNMP agent. If you do not specify a community, the default name is *public*.

Note: Community names are case sensitive.

trap SNMP waits for a trap from the specified host. If no host is specified, SNMP waits for a trap from any host.

-t **seconds**
Specifies the amount of time (in seconds) that the SNMP client waits for a reply from the SNMP agent. The default is 6 seconds.

¦-p **port**
¦ Specifies the destination (agent) port. The default is 161.

¦set Sends a request to the SNMP agent to set a specific MIB variable.
¦ SNMP then waits for a response, a trap, or a timeout.

¦**mib_variable**
¦ Specifies the MIB object, using its object descriptor (textual
¦ name), object identifier using ASN.1 notation, or a combination of
¦ the two. When used with WALK, this is the MIB object prefix. A
¦ prefix can be any leading portion of the complete object identifier.

¦**type** The type of value being set. To complete an SNMP SET request, the
¦ SMI_type must be known. If no type is specified, SNMP searches the
¦ compiled MIB and the MIB2.TBL to determine the type. If the
¦ variable is not found, an error is returned. If a type is
¦ specified, the type takes precedence over any type that may be
¦ assigned in the MIB. The type and value must be compatible. For
¦ example, if you specify a type of "number" and a value of "foo," an
¦ error will be returned because "foo" is not a number. Possible
¦ variable types are:

MIB2.TBL Data Type	**ASN.1 SMI_types**
number	INTEGER
display	(none)
object	OBJECT_IDENTIFIER
internet	IPAddress
counter	(none)
gauge	(none)
ticks	TimeTicks
string	Octet_String
(none)	Opaque
(none)	NULL

¦**value** Specifies the value to be set by the SET function. If white space
¦ is needed in the value, you must enclose the value in double quotes
¦ ("). If you want to set a variable to a value that is also a type,
¦ you must specify the type.

¦get Sends a request to an SNMP agent for a specific management
¦ information base (MIB) variable. SNMP then waits for a response, a
¦ trap, or a timeout.

¦getnext
¦ Sends a request to an SNMP agent for the next MIB variable that
¦ lexicographically follows the mib_variable specified. SNMP then
¦ waits for a response, a trap, or a timeout.

¦walk Issues a GETNEXT request for a specified prefix, then continues to
¦ issue GETNEXT requests for as long as there are variables that match
¦ the specified prefix. A prefix can be any leading portion of the
¦ complete object identifier.

¦-? Displays help information.

¦Deciding Which Subcommand to Use: As a general rule, use:

¦□ **get** to obtain scalar MIB variables
¦□ **getnext** and **walk** to obtain MIB variables within a table
¦□ **set** to set the value of a MIB variable
¦□ **trap** to receive traps from a host

To understand the use of **get**, **getnext**, and the **walk** subcommands, you should understand how SNMP identifies a particular instance of a MIB object, the concept of lexicographic ordering, and ASN.1 notation.

As described in "Customizing SNMP for Your Network" in topic 13.7.3, each MIB object can be identified by:

- A textual name, called the object descriptor. For example: **sysDescr**.

- A object identifier using ASN.1 notation. For example, **1.3.6.1.2.1.1.1.** is the ASN.1 equivalent of sysDescr.

A particular instance of a MIB object is identified by appending a suffix onto the end of the object. For scalar variables (those not in a table), this suffix is 0. For example, the only instance of the MIB object sysDescr is sysDescr.**0**. This is 1.3.6.1.2.1.1.1.**0** expressed in ASN.1 notation.

For MIB variables that are part of a table, the suffix appended depends upon the table. For example, instances of variables in the interfaces table are determined by the value of the ifIndex column. Therefore, the instance of ifSpeed (the speed of an interface) for the first interface is ifSpeed.**1**, or in ASN.1 notation,

1.3.6.1.2.1.2.2.1.5.**1**

There is an ordering to the MIB objects; for example, sysObjectID (1.3.6.1.2.1.1) is lexicographically after sysDescr (1.3.6.1.2.1.1.).

Examples

Note: You must specify an instance when you issue the following commands.

1. To obtains the value of the object instance that is lexicographically next in the Agent's MIB, enter:

   ```
   snmp getnext sysdescr.0
   ```

   (The value of **sysObjectID** is displayed because sysObjectID is lexicographically after the 0 instance of sysDescr.)

2. To obtain the value of the instance specified, enter:

   ```
   snmp get sysdescr.0
   ```

   The value of the variable is displayed.

3. To obtain the value of all object instances in the Agent's MIB, enter:

   ```
   snmp walk ifindex
   ```

   A sample response is:

```
ifIndex.1 : 1
ifIndex.2 : 2
End of variables with prefix: ifIndex (1.3.6.1.2.1.2.2.1.1)

snmp walk tcpConnRemAddress
tcpConnRemAddress.0.0.0.0.0.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.7.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.9.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.13.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.19.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.21.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.23.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.25.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.37.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.79.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.111.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.199.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.512.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.513.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.514.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.540.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.608.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.666.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.697.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.702.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.710.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.713.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.714.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.715.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.728.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.811.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.865.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.870.0.0.0.0.0 : 0.0.0.0
```

```
tcpConnRemAddress.0.0.0.0.1025.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.1234.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.1305.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.2401.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.6000.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.6111.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.6680.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.7000.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.0.0.0.0.8710.0.0.0.0.0 : 0.0.0.0
tcpConnRemAddress.9.2.219.64.1045.9.67.165.107.23 : 9.67.165.107
tcpConnRemAddress.9.2.219.64.1197.9.2.197.79.23 : 9.2.197.79
tcpConnRemAddress.9.2.219.64.1871.9.2.254.247.23 : 9.2.254.247
End of variables with prefix: tcpConnRemAddress (1.3.6.1.2.1.6.13.1.4)
```

4. To set the value of the sysLocation MIB variable, enter:

```
snmp -c private set syslocation.0 35-024
```

The updated value of the variable is displayed.

```
sysLocation.0 : 35-024
```

13.7.8 Displaying TRAPs Received from SNMP Agents

To display SNMP TRAPs received from any SNMP agent in your network, use the **snmptrap** command. SNMPTRAP displays the TRAPs in a Presentation Manager window.

Subtopics
13.7.8.1 SNMP TRAPs--Format and Meaning
13.7.8.2 The snmptrap Command

13.7.8.1 SNMP TRAPs--Format and Meaning

In the majority of cases, the SNMP client makes a request of an SNMP agent, and the agent responds. At times, there is a relationship in which the SNMP agent provides an unsolicited notification of some network significant event to the SNMP client.

These unsolicited notifications from the agent to the client are referred to as TRAPs. TRAPs are asynchronous events, which means an agent sending a TRAP does not guarantee when (or if) the TRAP will be received by the client.

Types of TRAPs: RFC 1157 defines the following seven TRAPs:

coldStart
Signifies that the SNMP agent is initializing itself such that MIB objects can be altered. An example of this would be rebooting a personal computer.

warmStart
Signifies that the SNMP agent is initializing itself such that MIB objects are not altered.

linkDown
Indicates that an interface has changed from the up to the down state. This is indicated by the value of the MIB object ifOperStatus in the interfaces group.

linkUp
Indicates that an interface has changed from the down to the up state. This is indicated by the value of the MIB object ifOperStatus in the interfaces group.

authenticationFailure
Indicates that an SNMP message has been received that is not properly authenticated. For the OS/2 SNMP agent, authentication means that both the community name and source address have been verified.

egpNeighborLoss
Indicates that an EGP peer has been marked down.

enterpriseSpecific
Indicates that some other, privately defined, event has occurred.

Note: The SNMP agent DPI, which is supported by the TCP/IP for OS/2 SNMP agent, allows external processes to generate SNMP TRAPs, which allows you to define enterprise specific TRAPs. For more information about SNMP DPI, see the *IBM TCP/IP Version 2.0 for OS/2: Programmer's Reference*.

13.7.8.2 The snmptrap Command

The syntax of the **snmptrap** command is: notification of network events (TRAPs) from SNMP agents.

>>--**snmptrap**--><

The SNMPTRAP window is displayed. To begin displaying TRAPs, select **Get_Traps** from the action bar, and then select **Start** from the pull-down.

When you select **Start**, the message "Waiting for Traps..." is displayed.

Figure 46 is a sample of an SNMPTRAP display.

PICTURE 46

Figure 46. Sample SNMPTRAP Display

Each TRAP received is displayed with the following information:

- The type of TRAP (for example, authentication Failure)
- The IP Address of the host which generated the TRAP
- The value of sysObjectID for the agent generating the TRAP
- The value of sysUpTime for the agent generating the TRAP

In the last TRAP in Figure 46, host 9.67.60.162 generated an authentication Failure. The sysObjectID of this host is 1.3.6.1.4.1.2.2.1.2.2, and the value of sysUptime was 4897384.

If the window is minimized while no traps are listed, the icon shown is black. The icon turns red when a TRAP is received.

13.7.9 SNMP Diagnostics

Table 11 shows SNMP messages and codes.

Table 11. SNMP Messages and Codes

Exit Code	**Message**	**Explanation**
None	Unknown SNMP request type	The following SNMP request types are valid: □ snmp_get □ snmp_getnext □ snmp_set □ snmp_trap □ snmp_walk **Action:** Verify that you have entered a valid type.
None	arptbl_setup:cannot get memory for arptblp1	In file MIB_AT.C, the c function calloc() allocates storage for the data structure **arp_ent**. **Action:** Run fewer applications, and restart the application that failed.
None	iftable_setup:cannot get memory for ifp	In file MIB_INTE.C, the c function calloc() allocates storage for the data structure **if_ent**. **Action:** Run fewer applications, and restart the application that failed.
None	if_addrsetup:cannot get memory for ipadr1	In file MIB_IPAD.C, the c function calloc() allocates storage for the data structure **ipadr_ent**. **Action:** Run fewer applications, and restart the application that failed.
None	iproute_setup:cannot get memory for iprouteptr	In file MIB_IPRO.C, the c function calloc() allocates storage for the data structure **iproute_ent**. **Action:** Run fewer applications, and restart the application that failed.

13.8 Querying Name Servers

You can use the following commands to query servers for information about various hosts and domains.

Subtopics

13.8.1 The host Command

You can use the **host** command to translate a specified host name to its IP address or to translate a specified IP address to its host name. The syntax of this command is:

```
>>--host---- host_name----><
          +- ip_addr---+
```

host_name Host name of the host whose IP address is to be translated.
ip_addr IP address of the host whose host name is to be translated.

13.8.2 The hostname Command

You can use the **hostname** command to send a request to the domain name server to resolve your host name from your internet address and display the host name. The syntax of this command is:

```
>>--hostname--><
```

13.8.3 The nslookup Command

You can use the **nslookup** command to query servers for information about various hosts and domains or to print a list of hosts in a domain. It can be used in either interactive or noninteractive mode (command line).

Subtopics

13.8.3.1 Interactive Mode

You enter interactive mode when:

- You give no parameter and use the default name server.
- You use a hyphen (-) for the first parameter and the host name or internet address of a name server for the second parameter.

You can enter the NSLOOKUP command shell without specifying a host by using the **nslookup** command without parameters. You can also enter the NSLOOKUP command shell without specifying a host but using a specific server.

The syntax of the **nslookup** command using the default server is:

```
>>--nslookup---------------><
              +--option-+
```

The syntax of the **nslookup** command using a specific server is:

```
>>--nslookup---------------------------><
              +--option-+  +--server-+
```

The subcommands available in interactive mode allow you to issue queries and control the output. Internal state information affects the operation and results of your queries. You can change the internal state information that is maintained by NSLOOKUP with the **set** subcommand. For more information about changing the internal state information, see the **set** subcommand in "NSLOOKUP Subcommands for Interactive Queries" in topic 13.8.3.4.

13.8.3.2 Noninteractive (Command Line) Mode

You enter noninteractive mode when you use the hostname or internet address of the host to be queried as the first parameter. The optional second parameter specifies the host name or name server address.

The following two examples show how to enter the NSLOOKUP command shell with a host specified:

The following command invokes NSLOOKUP using a specific host and the default server:

```
>>--nslookup--------------host--><
              +--option-+
```

The following command invokes NSLOOKUP using a specific host and a specific name server:

```
>>--nslookup--------------host--server--><
              +--option-+
```

Note: Subcommands are case-sensitive and must be entered in lowercase. Host names are not case-sensitive.

You can also specify the options for the **set** subcommand, as shown on the command line in "NSLOOKUP Subcommands for Interactive Queries" in topic 13.8.3.4. The option must have a hyphen (-) for a prefix and must precede the arguments. For example, the format of the **nslookup** command to change the default of **querytype** to host information and the value of **timeout** to 10 seconds is:

```
nslookup -querytype=hinfo -timeout=10
```

13.8.3.3 NSLOOKUP Files

You must have the following files defined in the **TCPIP\ETC** directory:

File Name	**Description**
resolv	Contains the initial domain name and name server addresses.
nslookup	Contains the NSLOOKUP configuration and STARTUP file. Each line represents the **set** subcommand. The NSLOOKUP file is optional.

13.8.3.4 NSLOOKUP Subcommands for Interactive Queries

An **nslookup** subcommand can be up to 256 characters in length. An unrecognized, or invalid, subcommand is interpreted as a host name. Interactive subcommands can be interrupted by holding down the Ctrl key and pressing the C key. To exit NSLOOKUP, enter **exit**. The subcommands for NSLOOKUP are shown on the following pages.

The following subcommand looks up information for the specified host, using the current default server, if you do not specify a server. If you specify a server, the subcommand looks up information for the specified host, and the specified server.

```
>>--host---------------><
          +- server-+
```

host Specifies the name or internet address of a host.

If the host you specify is an internet address and the query type is A or PTR, the name of the host is returned.

If the host you specify is a name and does not have a trailing period, the default domain name is appended to the name. This behavior depends on the state of the set options:

- domain
- srchlist
- defname
- search

You can initiate a search of the domains that are not current by specifying a host name followed by a period.

server Specifies the server. If you do not specify a server, the current server will be used.

If **host** is an Internet address and the query type is A or PTR, the name of the host is returned.

If **host** is a name and does not have a trailing period, the default domain name is appended to the name. This behavior depends on the state of the SET options:

- domain
- srchlist
- defname
- search

To look up a host that is not in the current domain, append a period to the name.

This subcommand changes the default server to **name**, using the current default server. If an authoritative answer is not found, the names of servers that might provide the answer are returned.

```
>>--server-- name-->
```

name Specifies the name of the server.

This subcommand changes the default server to **name**, using the initial server. If an authoritative answer is not found, the names of servers that might provide the answer are returned.

```
>>--lserver-- name-->
```

name Specifies the name of the server.

This subcommand changes the default server to the server for the root of the domain name space. The current host is **ns.nic.ddn.mil** by default, but can be changed using the **set** subcommand.

```
>>--root-->
```

The initial setting for the root server is ns.nic.ddn.mil, but it can be changed using the **set root** subcommand.

This command is equivalent to **lserver ns.nic.ddn.mil**.

This subcommand connects with the FINGER server on the current host. The

current host is defined when a previous lookup for a host was successful and address information was returned.

```
>>--finger-------------------------------------><
            +- name-+  +--- >----- filename-+
                       +- >>-+
```

name Specifies that a list of logged-on users for the current host be returned. Information about a particular user can be found by specifying the **name** of the user as a parameter. The name is passed in the specified case to the query host. Therefore, it is necessary to specify the name in the same case in which the name appears on the host.

> Specifies that the output be directed to a file. Output goes to a file that you can view later with the **view** subcommand.

>> Specifies that the output be directed to a file. Output is appended to a file that you can view later with the **view** subcommand.

filename Specifies the name of the file where the output is to be written.

For more information on the returned address, see querytype=A on page 13.8.3.5.

The **name** parameter is optional. By default, this command returns a list of logged-in users for the current host. You can find information about a particular user by specifying the **name** of the user as a parameter. You can specify the **name** in uppercase or lowercase. The name is passed in the specified case to the query host. Therefore, you must specify the name in the same case in which the name appears on the host.

The > or >> parameter can be used to redirect output. Output can be placed in a file for later viewing with the view subcommand.

This subcommand lists the information available for **domain**, optionally creating or appending to **filename**. The default output contains host names and their Internet addresses.

```
>>--ls---------------------- domain--------------------------><
        +- -t querytype-¦           +--- >----- filename-+
        +- -a-----------¦             +- >>-+
        +- -d-----------¦
        +- -h-----------¦
        +- -s-----------+
```

-t querytype
Lists all records of the specified type.

-a Lists aliases of hosts in the domain. Using this option is the same as issuing the **set type** subcommand with the CNAME parameter.

-d Lists all records for the domain. Using this option is the same as issuing the **set type** subcommand with the ANY parameter.

-h Lists operating system information for the domain. Using this option is the same as issuing the **set type** subcommand with the HINFO parameter.

-s Lists well-known services of hosts in the domain. Using this option is the same as issuing the **set type** subcommand with the WKS parameter.

domain Specifies the domain for which information is requested.

¦> Specifies that the output be directed to a file. Output goes to a ¦ file that you can view later with the **view** subcommand. When ¦ output is directed to a file, hash marks are printed for each ¦ group of 50 records received from the server.

¦>> Specifies that the output be directed to a file. Output is ¦ appended to a file that you can view later with the **view** ¦ subcommand. When output is directed to a file, hash marks are ¦ printed for each group of 50 records received from the server.

filename
Specifies the name of the file where the output is to be written.

The options allow you to select resource records other than the default.

The following subcommand displays a summary of commands.

>>--**view**-- **filename**--><

filename Specifies the name of the file where the output was saved.

This subcommand sorts and lists the output of previous **ls** command(s).

¦>>--**help**--><

The following subcommand exits the program.

>>--**exit**--><

13.8.3.5 The Set Subcommand

This subcommand is used to change internal state information that affects query results.

```
>>--set-- all--><
```

Use this format to print the current values of the frequently used options that set the internal state variables. Information about the current default server and host also prints.

```
¦>>--set-- class--=----in--------><
                     +-chaos--¦
                     +-hesiod-¦
                     +-any----+
```

¦in	Internet class. This value is the initial setting.
chaos	Chaos class.
hesiod	MIT Athena** Hesiod class.
any	Specifies any of the above.

Use this format to set the query class. The class specifies the protocol group of the information.

Use this format to toggle the debugging mode. The debug parameter prints debugging information for each query packet sent to the server and the resulting response. The nodebug parameter specifies that the debugging information is not printed. The default is nodebug.

```
¦>>--set---- nodebug----><
¦         +- debug---+
```

Use this format to enable or disable exhaustive debugging mode. In exhaustive debugging mode, all fields of every query packet are printed. The default is nod2.

```
¦>>--set---- nod2----><
¦         +- d2---+
```

Use this format to set the default domain name to **name**.

```
>>--set-- domain=name--><
```

name	Sets the default domain name to **name**. The default domain name is appended to a lookup request, depending on the state of the defname and search options.

The default domain name is appended to a LOOKUP request depending on the state of the defname and search options. The domain search list contains the parents of the default domain if it has at least two components in its name. For example, if the default domain is **CC.Berkeley.EDU**, the search list is **CC.Berkeley.EDU** and **Berkeley.EDU**.

Use the **set srchlist** command to specify a different list. Use the **set all** command to display the list. The default is the current value for hostname, \ETC\RESOLV.

```
                     <-/----+
>>--set-- srchlist=---name-----><
```

name	Specifies a domain name. You can specify a maximum of six names, each separated by a slash (/). For example, to set the domain name to lcs.MIT.EDU, and the search list to include ai.MIT.EDU, and MIT.EDU, as well as lcs.MIT.EDU, the subcommand is: `set srchlst=lcs.MIT.EDU/ai.MIT.EDU/MIT.EDU` This subcommand overrides the default domain name and search list of the **set domain** subcommand. You can display the list with the **set all** subcommand. The initial setting is the domain name found in \ETC\RESOLV.

Use this format to change the default domain name to **name1** and the domain search list to **name1**, **name2**, and so on. You can specify a maximum of six names, each separated by a slash (/). For example, to set the domain name to lcs.MIT.EDU, and the search list to include ai.MIT.EDU and MIT.EDU, as well as lcs.MIT.EDU, the command is:

```
      set srchlst=lcs.MIT.EDU/ai.MIT.EDU/MIT.EDU
```

This command overrides the default domain name and search list of the **set domain** command. You can display the list with the **set all** command. The default is the current value for hostname, \ETC\RESOLV.

Use this format to append the default domain name to a single-component query (one that does not contain a period).

```
>>--set---- defname------><
        +- nodefname-+
```

The default is defname.

Use this format to direct NSLOOKUP to append the domain names in the domain search list to the query (reiterative searching) until an answer is received, when the LOOKUP request contains at least one period but does not end with a trailing period.

```
>>--set---- search------><
        +- nosearch-+
```

The default is search.

Use this format to specify the port to access.

```
>>--set-- port--=--value--><
```

value Specifies the port number. The initial setting is 53.

You can change the default TCP/UDP name server port to the value specified. The default port number is 53.

```
>>--set---- querytype----=----A--------><
         +- type------+     +-CNAME-¦
                            +-HNAME-¦
                            +-MINFO-¦
                            +-MX----¦
                            +-NS----¦
                            +-PTR---¦
                            +-SOA---¦
                            +-TXT---¦
                            +-UINFO-¦
                            +-WKS---+
```

Note: These parameters must be entered in uppercase.

A	Indicates the Internet address of the host. This value is the initial setting.
CNAME	Indicates the canonical name for an alias.
HINFO	Indicates operating system information.
MINFO	Indicates the mailbox or mail list information.
MX	Indicates the mail exchanger.
NS	Indicates the name server for the named zone.
PTR	Indicates a pointer to the host name if the query is an internet address. If the query is not an Internet address, this is a pointer to other information.
SOA	Indicates the domain's start of authority information.
TXT	Indicates the text information.
UINFO	Indicates the user information.
WKS	Indicates the supported well-known services.

For information about other types such as ANY, AXFR, MB, MD, MF, and NULL, see RFC 1035.

Use this format to notify the name server to perform a recursive query, querying other servers if it does not have the information.

```
>>--set---- recurse------><
        +- norecurse-+
```

The default is recurse.

Use this format to set the number of times to retry a request.

```
>>--set-- retry--=--number--><
```

number Specifies the number of times to retry a request. The initial setting is 4.

When a reply to a request is not received within the specified time-out period, the time-out period is doubled and the request is resent. The default value for retry is 4.

Use this format to change the defined name of the root server to **host**.

```
>>--set-- root--=--host--><
```

host Specifies the name of the host the root server is changed to. The initial setting is ns.nic.ddn.mil.

The default of **root** is **ns.nic.ddn.mil**.

Use this format to change the defined number of seconds to wait before timing out on a request.

```
>>--set-- timeout--=--number--><
```

number Specifies the length of time, in seconds. The initial setting is 4.

Each retry doubles the timeout period. The default is 5 seconds.

Use this format to direct NSLOOKUP to use a virtual circuit (TCP connection) to send requests to the server.

```
>>--set---- novc----><
         +- vc---+
```

The default is novc (datagrams).

Use this format to handle truncated responses.

```
>>--set---- noignoretc----><
         +- ignoretc---+
```

The default is ignoretc.

13.8.3.6 Examples

The following examples show how to use the **nslookup** command. The examples illustrate interactive subcommands entered at the NSLOOKUP prompt (>) while in the NSLOOKUP shell.

1. To request help, enter:

```
?
```

A sample response is:

```
Commands:   (identifiers are shown in uppercase, [] means optional)
NAME        - print info about the host/domain NAME using default server
NAME1 NAME2 - as above, but use NAME2 as server
help or ?   - print info on common commands
set OPTION  - set an option
    all     - print options, current server and host
    [no]debug   - print debugging information
    [no]d2      - print exhaustive debugging information
    [no]defname - append domain name to each query
    [no]recurse - ask for recursive answer to query
    [no]vc      - always use a virtual circuit
    domain=NAME - set default domain name to NAME
    srchlist=N1[/N2/.../N6] - set domain to N1 and search list to N1,N2, etc.
    root=NAME   - set root server to NAME
    retry=X     - set number of retries to X
    timeout=X   - set initial time-out interval to X seconds
    querytype=X - set query type, for example, A,ANY,CNAME,HINFO,MX,NS,PTR,SOA,WKS
    type=X      - synonym for querytype
    class=X     - set query class to one of IN(Internet),CHAOS,HESIOD or ANY
server NAME     - set default server to NAME, using current default server
lserver NAME    - set default server to NAME, using initial server
finger [USER]   - finger the optional NAME at the current default host
root            - set current default server to the root
ls [opt] DOMAIN [>|>>FILE] - l
ist addresses in DOMAIN(optional: output to FILE)
    -a          -  list canonical names and aliases
    -h          -  list HINFO (CPU type and operating system)
    -s          -  list well-known services
    -d          -  list all records
    -t TYPE     -  list records of the given type (for example, A,CNAME,MX, etc.)
view FILE       - sort an 'ls' output file and view it with more
exit            - exit the program
```

2. To display address information about host **charm**:

```
> charm
```

A sample response is:

```
    Server:  nameserv.raleigh.ibm.com
    Address:  9.00.40.100
    Name:    charm.raleigh.ibm.com
    Address:  9.00.40.79
```

3. To display the default option settings, enter:

```
set all
```

A sample response is:

```
Default Server:  nameserv.raleigh.ibm.com
Address:  9.00.40.100
Set options:
  nodebug         defname         search          recurse
  nod2            novc            noignoretc      port=53
  querytype=A     class=IN        timeout=4       retry=4
```

```
   root=ns.nic.ddn.mil
   domain=raleigh.ibm.com
   srchlist=raleigh.ibm.com/ibm.com
```

4. To display name server records, enter:

```
ls -d raleigh.ibm.com
```

A sample response to is:

```
[namesrv.raleigh.ibm.com]
 raleigh.ibm.com.               SOA   namesrv.raleigh.ibm.com owner.rs6k.
    raleigh.ibm.com. (92013101 86400 300 864000 3600)
 raleigh.ibm.com.               NS    namesrv.raleigh.ibm.com
 mvs20                          A     9.00.68.242
 mvs20                          HINFO IBM-3090    MVS/XA
 vm9-sna                        A     9.00.68.99
 vm9-sna                        HINFO IBM-3090    VM/ESA
 buzz                           A     9.00.32.2
 buzz                           HINFO IBM-PS/2    OS/2
 buzz                           A     9.00.70.84
 buzz                           A     9.00.70.7
 rs6k                           A     9.00.70.92
 rs6k                           HINFO IBM-RS6000  AIX3.1
 dosbox                         A     9.00.70.19
 dosbox                         HINFO IBM-PS/2    DOS
 localhost                      A     127.0.0.1
 pebbles                        A     9.00.70.80
 pebbles                        HINFO IBM-PS/2    OS/2-2.0
 boris                          CNAME pebbles.raleigh.ibm.com
 ps205                          A     9.00.70.145
 ps205                          HINFO IBM-PS/2    AIX-PS/2
 ---
 ---
```

The response displays the names of the hosts in the domain, their respective Internet addresses, the types of information, and the systems.

5. To display the debug trace for host . In the example, the trace is requested for host **charm**, enter:

```
set debug
set d2
charm
```

A sample response is:

```
Server:  namesrv.raleigh.ibm.com
Address:  9.00.40.100
------------
SendRequest(), len 42
    HEADER:
        opcode = QUERY, id = 2, rcode = NOERROR
        header flags:  query, want recursion
        questions = 1,  answers = 0,  authority records = 0,  additional = 0
    QUESTIONS:
        charm.raleigh.ibm.com, type = A, class = IN
------------
------------
Got answer (82 bytes):
  HEADER:
     opcode = QUERY, id = 2, rcode = NOERROR
     header flags:  response, auth. answer, want recursion, recursion avail.
     questions = 1,  answers = 1,  authority records = 0,  additional = 0
  QUESTIONS:
     charm.raleigh.ibm.com, type = A, class = IN
  ANSWERS:
  -> charm.raleigh.ibm.com
     type = A, class = IN, dlen = 4
     internet address = 9.00.40.79
     ttl = 86400 (1 days)
------------
```

```
¦ Name:    charm.raleigh.ibm.com                                                ¦
¦ Address:  9.00.40.79                                                          ¦
¦                                                                               ¦
+-------------------------------------------------------------------------------+
```

The response indicates the default name server as the requester and displays the request sent, the answer received, and the name of the host for which the trace was sent.

6. To display the Mail Exchanger (MX) records for host **charm**, enter:

```
set querytype=mx
charm
```

A sample response is:

```
+-------------------------------------------------------------------------------+
¦                                                                               ¦
¦                                                                               ¦
¦ Server:  nameserv.raleigh.ibm.com                                             ¦
¦ Address:  9.00.40.100                                                         ¦
¦ charm.raleigh.ibm.com preference=10, mail exchanger=charm.raleigh.ibm.com     ¦
¦ charm.raleigh.ibm.com preference=20, mail exchanger=charm1.raleigh.ibm.com    ¦
¦ charm.raleigh.ibm.com preference=20, mail exchanger=charm2.raleigh.ibm.com    ¦
¦ charm.raleigh.ibm.com internet address=9.00.40.79                             ¦
¦ charm1.raleigh.ibm.com internet address=9.00.40.81                            ¦
¦ charm2.raleigh.ibm.com internet address=9.00.40.83                            ¦
¦                                                                               ¦
+-------------------------------------------------------------------------------+
```

13.8.4 NSLOOKUP Diagnostic

If the NSLOOKUP query was not successful, an error message is printed. Possible error messages are:

Error	**Description**
Timed out	The server did not respond to a request after a certain amount of time and a certain number of retries.
No response from server	No name server is running on the server machine.
No records	The server does not have resource records of the current query type for the host, although the host name is valid. The query type is specified with the set querytype command.
Nonexistent domain	The host or domain name does not exist.
Connection refused	The connection to the name or FINGER server could not be made at the present time. This error commonly occurs with LS and FINGER requests.
Network is unreachable	The connection to the name or FINGER server could not be made at the present time. This error commonly occurs with LS and FINGER requests.
Server failure	The name server found an internal inconsistency in its database and could not return a valid answer.
Refused	The name server refused to service the request.
Format error	The name server found that the request packet was not in the proper format.

A.0 Appendix A. Introducing Computer Networks and Protocols

This chapter introduces the concepts of computer networks and of an internet environment. The protocols TCP/IP uses are listed by layer and are then described. Routing and addressing guidelines are also described.

Subtopics

A.1 Computer Networks

A computer network is a group of connected nodes that are used for data communication. A computer network configuration consists of data processing devices, software, and transmission media that are linked for information interchange.

Nodes are the functional units of the network, located at the points of connection among the data circuits. A node of the network, or end point, can be any host computer, a communication controller, a cluster controller, a video display terminal, or another peripheral device.

Computer networks can be local area networks (LANs), which provide direct communication among data stations on the user's local premises, or wide area networks (WANs), which provide communication services to a geographic area larger than that served by a LAN. Typically, WANs operate at a slower speed than LANs.

Different types of networks provide different functions, and network configurations vary, depending on the functions required by the organization. The technology used by these networks varies not only from organization to organization but also from network to network within a single organization.

Networks can differ at any or all layers. At the physical layer, networks can run over various network interfaces, such as Token Ring, Ethernet, PC Network, Fiber Distribution Data Interface (FDDI), X.25, and Integrated Services Digital Network (ISDN). Networks can also vary in the architectures they use to implement network strategies. Some of the more common architectures used today are Open Systems Interconnect (OSI), Transmission Control Protocol/Internet Protocol (TCP/IP), and System Network Architecture (SNA). Networks use different protocols to communicate over the different physical interfaces available. In addition to these differences, networks can all use different software packages to implement various functions.

In an effort to accommodate these differences, the concept of an internet emerged and developed.

A.2 The Internet Environment

An internet is a logical collection of networks supported by gateways, routers, bridges, hosts, and various layers of protocols. An internet permits different physical networks to function as a single, large, virtual network, and permits dissimilar computers to communicate, regardless of their physical connections. Processes (or applications) within gateways, routers, and hosts originate and receive packets of information. Protocols are sets of rules required to exchange these packets of information.

Understanding the following terms is central to understanding the internet environment:

A ***client*** is a computer or process that requests services on the network. A ***server*** is a computer or process that responds to a request for service from a client. A ***user*** accesses a service, which allows the use of data or some other resource.

A ***datagram*** is the basic unit of information, consisting of one data packet, which is passed across an internet at the transport level.

A ***gateway*** is a functional unit that connects two computer networks of different network architectures. A ***router*** is a device that connects networks at the International Organization for Standardization (ISO) Network Layer. Routers are protocol-dependent and connect only networks operating the same protocol. Routers do more than transmit data; they also select the best transmission paths and optimum sizes for packets. A ***bridge*** is a type of router that connects two or more networks and forwards packets among them. The operations carried out by bridges are done at the Media Access Control layer and are transparent to TCP/IP and TCP/IP routing.

A ***host*** is a computer, connected to a network, which provides an access point to that network. A host can be a client, a server, or a client and server simultaneously. In a communication network, computers are both the sources and destinations of the packets. The ***local host*** is the computer to which a user's terminal is directly connected without the use of an internet, for example, a workstation running TCP/IP. A ***foreign host*** is any host on the network including the local host. A ***remote host*** is any foreign host not including the local host. You identify hosts by their internet addresses.

An ***internet address*** is a unique 32-bit address identifying each node in an internet. An internet address consists of a network number and a local address. Internet addresses are represented in dotted-decimal notation and are used to route packets through the network.

Mapping relates internet addresses to physical hardware addresses in the network. For example, the Address Resolution Protocol (ARP) is used to map internet addresses to token ring or Ethernet physical hardware addresses.

A ***network*** is the combination of two or more nodes and the connecting branches among them. A ***physical network*** is the hardware that makes up a network. A ***logical network*** is the abstract organization overlaid on one or more physical networks. An internet is an example of a logical network.

Packet refers to the unit or block of data of one transaction between a host and its network. Packets are the exchange medium used at the internetwork layer to send and receive data through the network. A packet usually contains a network header, at least one high-level protocol header, and data blocks. Generally, the format of the data blocks does not affect how packets are handled.

A ***port*** is an end point for communication between applications, generally referring to a logical connection. A port provides queues for sending and receiving data. Each port has a port number for identification. When the port number is combined with an internet address, a ***socket*** address results.

A ***protocol*** is a set of rules for achieving communication on a network.

A.3 TCP/IP Protocols and Functions

This section categorizes the TCP/IP protocols and functions by their functional group (internetwork layer, transport layer, and application layer).

Table 12 in topic A.3 shows the functional groups and their related protocols and functions.

```
       rules=horiz
       concat=no
       frame=none
       align='l l'.
```

Table 12. Functional Groups

Group	**Protocols and Functions**
Network Layer	Network Driver Interface Specification** (NDIS**) Serial Line Internet Protocol (SLIP) X.25 Protocol
Internetwork Layer	Internet Protocol (IP) Internet Control Message Protocol (ICMP) Address Resolution Protocol (ARP)
Transport Layer	Transmission Control Protocol (TCP) User Datagram Protocol (UDP)
Application Layer	Telnet File Transfer Protocol (FTP) Trivial File Transfer Protocol (TFTP) Simple Mail Transfer Protocol (SMTP) Domain Name System (DNS) Simple Network Management Protocol (SNMP) Remote Printing (LPR and LPD) TALK FINGER ROUTED X Window System** Remote Procedure Call (RPC) Network File System** (NFS**) Remote Execution Protocol (REXEC) Socket Interfaces

Figure 47 shows the relationship of these protocols and functions within the TCP/IP layered architecture.

PICTURE 47

Figure 47. The TCP/IP Layered Architecture

A.4 Network Protocols

This section describes the protocols that comprise the network layer in TCP/IP for OS/2. Network protocols define how data is transported over a physical network, but these network protocols are not defined by TCP/IP. After a TCP/IP packet is created, the network protocol adds a transport-dependent network header before the packet is sent out on the network.

A.5 NDIS

The Network Driver Interface Specification (NDIS) is a medium access control (MAC) interface for local area network (LAN) adapter drivers (hereafter referred to as network adapter drivers) and protocol drivers. NDIS has become an industry standard, providing a common, open interface that enables network adapters and LAN software from different manufacturers to communicate with each other.

A network adapter driver provides the communication between a network adapter and a protocol, using NDIS as the interface. Network adapter drivers handle the basic transmission and reception of packets on the network.

A protocol driver provides the communication between an application and a network adapter driver, using NDIS as the interface. Protocol drivers provide a high level of communications between the data link layer and the application layer.

Subtopics
A.5.1 X.25 Protocol
A.5.2 Serial Line Internet Protocol (SLIP)

A.5.1 X.25 Protocol

You can use an X.25 network to establish a TCP/IP connection between two hosts. X.25, recommended as a communication interface standard by the International Telegraph and Telephone Consultative Committee (CCITT), defines the interface between Data Terminal Equipment (DTE) and Data Circuit-Terminating Equipment (DCE). A DTE is a computer or workstation connected to a network. A DCE is the equipment at the point of the connection to the network, such as a modem.

For more information about TCP/IP over X.25, see Request For Comments (RFC) 877.

A.5.2 Serial Line Internet Protocol (SLIP)

In TCP/IP for OS/2, the Serial Line Internet Protocol (SLIP) allows you to set up a point-to-point connection between two TCP/IP hosts over a serial line, for example, a serial cable or an RS-232 connection into a modem and over a telephone line. You can use SLIP to access a remote TCP/IP network from your local host or to route datagrams between two TCP/IP networks.

For more information about SLIP, see RFC 1055. For more information on setting up a SLIP interface, see the *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration*.

A.6 Internetwork Protocols

This section describes the internetwork protocols in TCP/IP for OS/2. Protocols in the internetwork layer provide connection services for TCP/IP. These protocols connect physical networks and transport protocols.

For more information about TCP/IP in general, see RFCs 1118, 1180, 1206, 1207, and 1208. See Appendix F, "Requests for Comments (RFCs)," for a list of other related RFCs.

Subtopics

A.6.1 Internet Protocol (IP)

The Internet Protocol (IP) provides the interface from the transport layer (host-to-host) protocols to the physical-level protocols. IP is the basic transport mechanism for routing IP packets to the next gateway, router, or destination host.

IP provides the means to transmit blocks of data (or packets of bits) from sources to destinations. Sources and destinations are hosts identified by fixed-length addresses. Outgoing packets automatically have an IP header prefixed to them, and incoming packets have their IP header removed before being sent to the higher-level protocols. This protocol provides for the universal addressing of hosts in an internet network.

IP does not ensure a reliable communication because it does not require acknowledgments from the sending host, the receiving host, or intermediate hosts. IP does not provide error control for data; it provides only a header checksum. IP treats each packet as an independent entity unrelated to any other packet. IP does not perform retransmissions or flow control. A higher-level protocol that uses IP must implement its own reliability procedures.

For more information about IP, see RFC 791.

A.6.2 Internet Control Message Protocol (ICMP)

The Internet Control Message Protocol (ICMP) passes control messages between hosts, gateways, and routers. For example, ICMP messages can be sent in any of the following situations:

- When a host checks to see if another host is available (PING).
- When a packet cannot reach its destination.
- When a gateway or router can direct a host to send traffic on a shorter route.
- When a host requests a netmask or a time stamp.
- When a gateway or router does not have the buffering capacity to forward a packet.

ICMP provides feedback about problems in the communication environment; it does not make IP reliable. The use of ICMP does not guarantee that an IP packet will be delivered or that an ICMP message will be returned to the source host when an IP packet is not delivered or is incorrectly delivered.

For more information about ICMP, see RFC 792.

A.6.3 Address Resolution Protocol (ARP)

The Address Resolution Protocol (ARP) maps internet addresses to hardware addresses. TCP/IP uses ARP to collect and distribute the information for mapping tables.

ARP is not directly available to users or applications. When an application sends an internet packet, IP requests the appropriate address mapping. If the mapping is not in the mapping table, an ARP broadcast packet is sent to all the hosts on the network requesting the physical hardware address for the host.

For more information about ARP, see RFC 826.

A.7 Transport Protocols

This section describes the transport protocols in TCP/IP for OS/2, which allow communication between application programs.

Subtopics
A.7.1 Transmission Control Protocol (TCP)
A.7.2 User Datagram Protocol (UDP)

A.7.1 Transmission Control Protocol (TCP)

The Transmission Control Protocol (TCP) provides a reliable vehicle for delivering packets between hosts on an internet. TCP takes a stream of data, breaks it into datagrams, sends each one individually using IP, and reassembles the datagrams at the destination node. If any datagrams are lost or damaged during transmission, TCP detects this and resends the missing datagrams. The received data stream is a reliable copy of the transmitted data stream.

For more information about TCP, see RFC 793.

A.7.2 User Datagram Protocol (UDP)

The user Datagram Protocol (UDP) provides an unreliable mode of communication between source and destination hosts and is an alternative to the TCP transport protocol. UDP is a datagram-level protocol built directly on the IP layer.

Like IP, UDP does not guarantee datagram delivery or duplication protection. UDP does provide checksums for both the header and data portions of a datagram. However, applications that require reliable delivery of streams of data should use TCP.

For more information about UDP, see RFC 768.

A.8 Applications, Functions, and Protocols

This section describes the applications protocols provided in TCP/IP for OS/2. Applications allow you to use network services and are included in the application layer of TCP/IP. The application layer is built on the services of the transport layer.

Subtopics

A.8.1 Telnet Protocol

The Telnet Protocol provides a standard method to interface terminal devices and terminal-oriented processes with each other. Telnet is built on the services of TCP in the transport layer. Telnet provides duplex communication and sends data either as ASCII characters or binary data.

Telnet is commonly used to establish a logon session on a foreign host. Telnet can also be used for terminal-to-terminal communication and interprocess communication.

For more information about the Telnet Protocol, see RFCs 854, 856, 857, 885, and 1091.

A.8.2 File Transfer Protocol (FTP)

The File Transfer Protocol (FTP) allows you to transfer data between local and foreign hosts or between two foreign hosts. FTP is built on the services of TCP in the transport layer. FTP transfers files as either ASCII characters or binary data. ASCII characters are used to transfer data sets that contain only text characters.

FTP provides functions, such as listing remote directories, changing the current remote directory, creating and removing remote directories, and transferring one or more files in a single request. Security is handled by passing user and account passwords to the foreign hosts.

For more information about FTP, see RFC 959.

A.8.3 Trivial File Transfer Protocol (TFTP)

The Trivial File Transfer Protocol (TFTP) is designed only to read and write files to and from a foreign host. TFTP is built upon the services of UDP in the transport layer. TFTP allows you to limit drive and directory access.

TFTP, like FTP, can transfer files as either ASCII characters or binary data. However, unlike FTP, TFTP cannot be used to list or change directories at a foreign host, and it has no provisions for user authentication.

For more information about TFTP, see RFC 783.

A.8.4 Simple Mail Transfer Protocol (SMTP)

The Simple Mail Transfer Protocol (SMTP) is an electronic mail protocol with both client (sender) and server (receiver) functions.

In the OS/2 environment, SMTP is implemented with the Sendmail program. You do not interface directly with SMTP. Instead, electronic mail software is used to create mail, which in turn uses SMTP to send the mail to its destination.

For more information about SMTP, see RFCs 821, 822, and 974.

A.8.5 Domain Name System (DNS)

The Domain Name System (DNS) uses a hierarchical system for naming hosts. Each host name is composed of domain labels separated by periods. Each label represents an increasingly higher domain level within an internet. The fully qualified domain name of a host connected to one of the larger internets generally has one or more subdomains.

For example:

host.subdomain.subdomain.rootdomain
or
host.subdomain.rootdomain

Domain names often reflect the hierarchy level used by network administrators to assign domain names. For example, the domain name **eng.mit.edu** is the lowest level domain name, which is a subdomain of **mit.edu**. The subdomain **mit.edu** is a subdomain of **edu**. Local network administrators have the authority to name local domains within an internet.

Figure 48 is an example of the DNS used in the hierarchy naming structure across an internet.

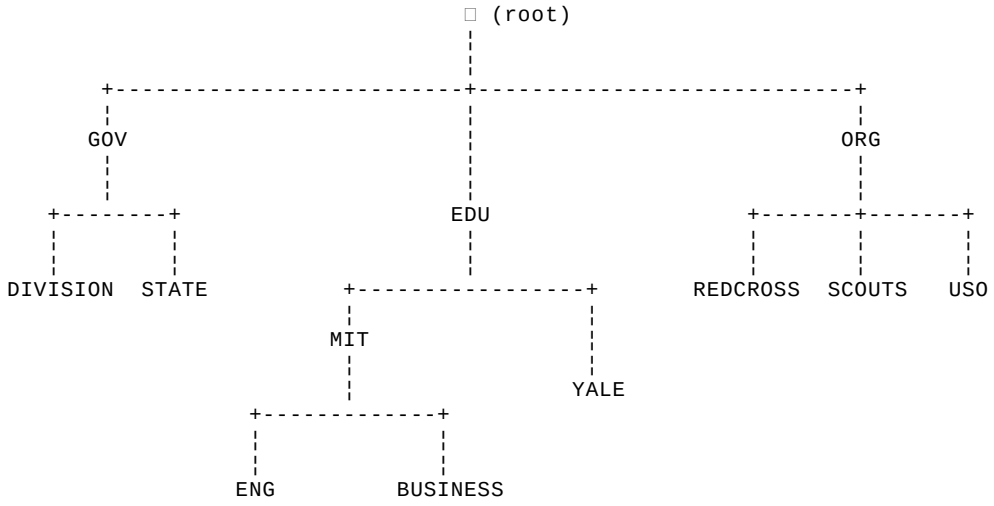


Figure 48. Hierarchical Tree

You can refer to hosts in your domain by host name only; however, a name server requires a fully qualified domain name. The local resolver combines the host name with the domain name before sending the address resolution request to the domain name server.

TCP/IP for OS/2 uses the file RESOLV in the ETC directory to specify the default domain appended to a host name and to specify the addresses of as many as five domain name servers to which address resolution queries can be sent.

TCP/IP for OS/2 can also use the local resolver functions of a local name resolution file. This file, called HOSTS, resides in the ETC directory and contains entries that allow you to map symbolic names to internet addresses. If both a RESOLV file and a HOSTS file exist in the ETC directory, the resolver sends the request to the foreign name servers listed in the RESOLV file before using the local HOSTS file.

When using the HOSTS file on a small internet, it is not necessary to use the hierarchical naming system used by the larger internets. The following example is a token ring network of three users and their entries in the HOSTS file.

```
129.5.24.1 Host1 vjsPC PC1 mathdept
129.5.24.3 PC3 normasPC Host3 # This is Norma's PC
129.5.24.4 PC4 budsPC
```

A carriage return must be entered at the end of each line.

In this example, each time the user enters the **host_name** of **Host1** or the **aliases vjsPC**, **PC1**, or **mathdept**, the local name resolver translates it to the internet address of **129.5.24.1**. For more information about the format of network addresses, see "Network Address Format" in topic A.10.1.

For more information about DNS, see RFCs 1034 and 1035.

A.8.6 Simple Network Management Protocol (SNMP)

The Simple Network Management Protocol (SNMP) provides a means for managing an internet environment. SNMP allows network management by elements, such as gateways, routers, and hosts. Network elements act as servers and contain management agents, which perform the management functions requested. Network management stations act as clients; they run the management applications, which monitor and control the network. SNMP provides a means of communicating between these elements and stations to send and receive information about network resources.

For more information about network management, see RFCs 1155, 1157, 1187, and 1213.

A.8.7 Remote Printing (LPR and LPD)

TCP/IP for OS/2 provides both client and server support for remote printing. The clients provided (LPR, LPQ, LPRM, and LPRMON) allow the user to send files or redirect printer output to a foreign host running a remote print server (LPD). These clients can also be used to query the status of a job, as well as to delegate a job.

TCP/IP for OS/2 also provides server support for remote printing (LPD). This allows any remote printing clients to submit, query, and delete print jobs from any OS/2 print queues.

For more information about remote printing, see RFC 1179.

A.8.8 TALK

TALK allows you to send interactive messages, as opposed to the batch mail capabilities of SMTP. When a local host sends a TALK request to a foreign host, the user on the foreign host is notified that there is a connection request. The user on the foreign host must respond with a TALK message to the local host. Message exchange can then occur between the local and foreign hosts.

A.8.9 FINGER Protocol (FINGER)

The FINGER Protocol (FINGER) provides an interface for querying the current status of a remote host or a user ID on a remote host. FINGER uses TCP as the underlying protocol.

For more information about FINGER, see RFC 1196.

A.8.10 Routing Information Protocol (RIP)

TCP/IP for OS/2 provides a server, ROUTED, that uses the Routing Information Protocol (RIP) to dynamically create and maintain network routing tables. RIP arranges to have gateways and routers periodically broadcast their routing tables to neighbors. Using this information, a ROUTED server can update a host's routing tables. For example, ROUTED determines if a new route has been created, if a route is temporarily unavailable, or if a more efficient route exists.

For more information about RIP, see RFC 1058.

A.8.11 X Window System

The X Window System Protocol supports network transparent windowing and graphics. TCP/IP for OS/2 provides server support to the X Window System client applications.

For more information about X Window System Protocol, see RFC 1013.

A.8.12 File Transfer Protocol Application Programming Interface (FTP API)

The File Transfer Protocol (FTP) Application Programming Interface (API) allows applications to have a client interface for file transfer. Applications written to this interface can communicate with multiple FTP servers at the same time. A maximum of 256 simultaneous connections are supported. The interface also allows third-party transfers between pairs of FTP servers. Consecutive third-party proxy transfers are allowed between any sequence of pairs of FTP servers.

The API tracks the servers to which an application is currently connected. When a new request for FTP service is requested, the API checks whether there is a connection to the server. If the connection does not exist, it is established. If the server has dropped the connection since last use, it is re-established.

FTP API provides functions, such as listing remote directories, changing the current remote directory, creating and removing remote directories, and transferring one or more files in a single request. Security is handled by passing user and account passwords to the foreign hosts.

For more information about FTP, see RFC 959.

A.8.13 Remote Procedure Call (RPC)

The Remote Procedure Call Protocol (RPC) is a programming interface that calls subroutines to be run on a foreign host. RPCs are high-level program calls, that can be used in place of the lower-level calls that are based on sockets.

For more information about RPC, see RFC 1057.

A.8.14 Network File System (NFS)

The Network File System (NFS) allows you to manipulate files on remote TCP/IP hosts as if they reside on your local host. NFS is based on the NFS protocol, and uses the Remote Procedure Call (RPC) protocol to communicate between the client and the server. The files to be accessed reside on the server host and are made available to the user on the client host.

NFS supports a hierarchical file structure. The directory and subdirectory structure can be different for individual client systems.

For more information about NFS, see RFC 1094.

A.8.15 Remote Execution Protocol (REXEC)

The Remote Execution Protocol (REXEC) allows you to issue a command on a foreign host and receive the results on the local host. REXEC provides automatic logon and user authentication, depending on the parameters set by the user.

A.8.16 Socket Interfaces

Socket interfaces allow you to write your own applications to supplement those supplied by TCP/IP. Most of these additional applications communicate with either TCP or UDP. Some applications are written to communicate directly with IP. To write applications that use the socket interfaces of TCP/IP for OS/2, you must purchase the Programmer's Toolkit.

Sockets are duplex, which means that data can be transmitted and received simultaneously. Sockets allow you to send to and receive from the socket as if you are writing to and reading from any other network device.

A.9 Routing

Routing is the process of deciding where to send a packet based on its destination address. The routing functions in an internet are performed at the internetwork layer. Two kinds of routing are involved in communication within an internet: direct and indirect.

Direct routing is used when the source and destination nodes are on the same logical network within an internet. The source node maps the destination internet address into a hardware address and sends packets to the destination node through this address. This mapping is normally performed through a translation table. If a match cannot be found for a destination internet address, ARP is invoked to determine this address.

Indirect routing is used when the source and destination nodes are on different networks within an internet. The source node sends packets to a gateway or router on the same network using direct routing. From there, the packets are forwarded through intermediate gateways or routers, as required, until they arrive at the destination network. Direct routing is then used to forward the packets to the destination host on that network. Each gateway, router, and host in an internet has a routing table that defines the address of the next gateway or router to other networks (as well as other nodes on other networks) in an internet.

A.10 Internet Addressing

Each host is assigned at least one unique internet address. This address is used by the IP and other higher-level protocols. When gateway hosts are used, more than one address may be required. Each interface to an internet is assigned its own unique address. Internet addresses are used to route packets through the network.

Addresses within an internet consist of a network number and a local address. The unique network number is assigned to each network when it connects to another internet. If a local network is not going to connect to other internets, any convenient network number is assigned. However, if you want to connect to other internet networks, it is recommended that you get an assigned network identification number from the network system administrator.

Hosts that exchange packets on the same physical network should have the same network number. If hosts have the same network number, part of the local address is used as a subnetwork number. All host interfaces attached to the same physical network are given the same subnetwork number.

An internet can provide standards for assigning addresses to networks, broadcasts, and subnetworks. Examples of these standard formats are described in the following sections.

Subtopics

A.10.1 Network Address Format

A standard internet address uses a two-part, 32-bit address field. The first part of the address field contains the network address; the second part contains the local address. The four different types of address fields are classified as A, B, C, or D, depending on the bit allocation.

Figure 49 represents a class A address. A class A address has a 7-bit network number and a 24-bit local address. The highest order bit is set to 0.

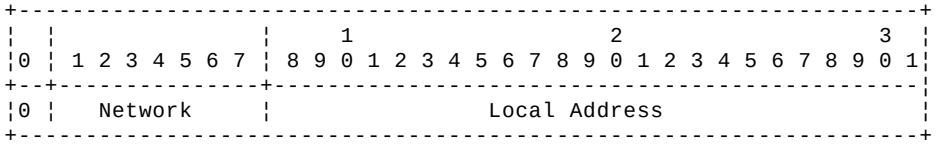


Figure 49. Class A Address

Figure 50 represents a class B address. A class B address has a 14-bit network number and a 16-bit local address with the highest order bits set to 10.

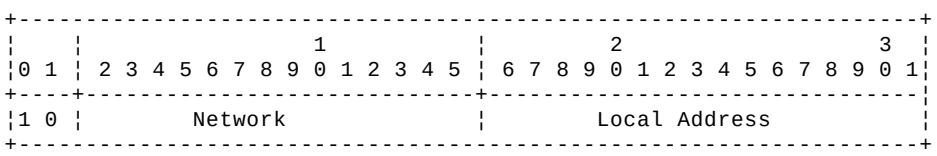


Figure 50. Class B Address

Figure 51 represents a class C address. A class C address has a 21-bit network number and an 8-bit local address with the three highest order bits set to 110.

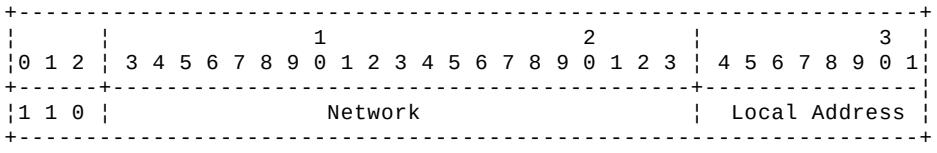


Figure 51. Class C Address

Figure 52 represents a class D address. A class D network is a multicast address that is sent to selected hosts on the network. The four highest order bits are set to 1110.

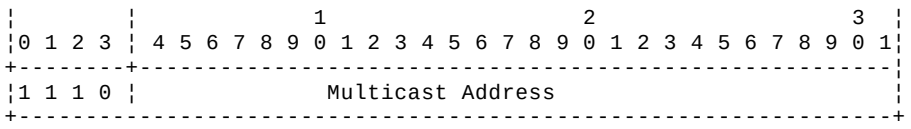


Figure 52. Class D Address

Note:

Class D addresses are not supported in TCP/IP for OS/2.

A commonly used notation for host addresses is the dotted-decimal, which divides the 32-bit address into four 8-bit fields. The value of each field is specified as a decimal number, and the fields are separated by periods.

Address examples in this book use dotted-decimal notation in the following forms:

Class A **nnn.lll.lll.lll**
Class B **nnn.nnn.lll.lll**
Class C **nnn.nnn.nnn.lll**

where:

nnn Represents part or all of a network number

lll	Represents part or all of a local address

A.10.2 Broadcast Address Format

TCP/IP uses IP broadcasting to send datagrams to all the TCP/IP hosts on a network or subnetwork. A datagram sent to the broadcast address is received by all the hosts on the network and processed as if the datagram were sent directly to the host's IP address. The IP broadcast address is formed by setting all the host bits to ones.

For more information about IP broadcasting, see RFCs 919 and 922.

A.10.3 Subnetwork Address Format

The subnetwork capability of TCP/IP divides a single network into multiple logical networks (subnets). For example, an organization can have a single internet network address that is known to users outside the organization, yet can configure its internal network into different departmental subnets. Subnetwork addresses enhance local routing capabilities while reducing the number of network numbers required.

For a subnet, the local address part of an internet address is divided into a subnet number and a host number, for example:

network_number subnet_number host_number

where:

network_number Is the network portion of the internet address
subnet_number Is a field of a constant width for a given network
host_number Is a field that is at least 1-bit wide

If the width of the **subnet_number** field is 0, the network is not organized into subnets, and addressing to the network is done with an internet network address (**network_number**).

Figure 53 represents a class B address with a 6-bit wide subnet field.

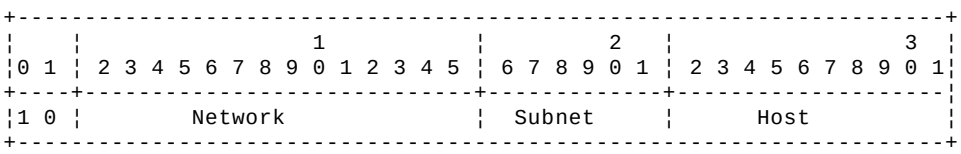


Figure 53. Class B Address with Subnet

The bits that identify the subnet are specified by a bit mask. A bit mask is a pattern of characters used to assign subnet addresses. The subnet bits are not required to be adjacent in the address. However, the subnet bits generally are contiguous and are the most significant bits of the local address.

For more information about subnetwork addresses, see RFC 950.

B.0 Appendix B. Optional Files

This appendix describes the files that you can create for certain applications in TCP/IP for OS/2. These files, with the names specified here, must reside in the ETC directory or in the directory specified by the ETC environment variable. Table 13 defines the optional files that are available.

Table 13. Usage of Optional Files for TCP/IP for OS/2

Name of File	**Used by**	**Purpose of File**
GATEWAYS	ROUTED server	Identifies gateways.
HOSTS	Any client and server	Resolves host names if a name server is unavailable.
INETD.LST	Selected servers	Defines servers to be started by the super server INETD.EXE.
MIB2.TBL	Several SNMP commands	Defines the mapping between an object's ASN.1 notation and an object's textual notation.
NETRC	FTP and REXEC clients	Alternative source for user and password.
PINGHOST.LST	PMPING	Specifies a list of hosts to be monitored.
PW.SRC	SNMP Agent	Plain-text list of community names for the SNMP agent.
RESOLV	Any client or server	Provides domain names and name server addresses.
RHOSTS	RSH server	Specifies the hosts that are authorized to use the RSH server.
SNMP.PW	SNMP Agent	The scrambled version of the PW.SRC file.
SNMPTRAP.DST	SNMP Agent	Specifies destination hosts that receive TRAP messages.
TELNET.RC	LINEMODE	Specifies commands to a Telnet client.
TRUSERS	FTP Server	Verifies user and password.

Table 14 shows example contents of each optional file.

Table 14. Contents of Optional Files for TCP/IP for OS/2

Name of File	**Sample Contents of File**
GATEWAYS	net net2 gateway host4 metric 4 passive host host3 gateway host4 metric 4 passive host host10 gateway 192.9.201.5 metric 9 active host host10 gateway 192.8.201.5 metric 8 external
HOSTS	124.34.216.1 Host1 joansPC PC1 educdept 124.34.216.3 PC3 edsPC Host3 # This is Ed's PC 124.34.216.5 PC5 janetsPC
INETD.LST	telnet tcp telnetd exec tcp rexecd ftp tcp ftpd printer tcp lpd tftp udp tftpd shell tcp rshd Where, for example: □ telnet is the service. □ tcp is the protocol. □ telnetd is the server to be activated. □ exec is the service. □ tcp is the protocol.

	□ rexecd is the server to be activated.
MIB2.TBL	sysDescr 1.3.6.1.2.1.1.1.0 display
NETRC	machine raleigh login kent password baseball machine boston login chris password boz macdef mymacro bell hash prompt binary cd c:\mydir get myfile.bin machine phoenix login writer password account payday
PINGHOST.LST	9.67.30.100 **Nameserver-Call_Dan 9.67.22.1 RALVMM_via_3172-Call_IS
PW.SRC	passwd1 9.0.0.0 255.0.0.0 passwd2 129.34.81.22 255.255.255.255
RESOLV	domain eng.mit.edu nameserver 129.34.128.245 nameserver 129.34.128.246
RHOSTS	kant.watson.ibm.com Scott jorge.raleigh.ibm.com Where: □ Scott is the only user on kant.watson.ibm.com that is served. □ All users on jorge.raleigh.ibm.com are served.
SNMPTRAP.DST	124.34.216.1 UDP Manager2 UDP
TELNET.RC	Default: send ayt toggle local set *escape_char* toggle *crmode*
TRUSERS	user: chris boz rd: d:\ c:\ wr: d:\tmp c:\tmp Where: □ chris is the user. □ boz is the password for chris. □ rd: d:\ c:\ gives chris read access to drives D: and C:. □ wr: d:\tmp c:\tmp gives chris write access to subdirectories D:\TMP and C:\TMP.

C.0 Appendix C. Problem Determination Examples

This appendix describes the processes used to determine the causes of the problems that might occur in TCP/IP for OS/2.

Table 15 gives some problem determination examples.

Table 15. Causes and Resolutions for TCP/IP Application Problems

Problem	**Resolution**
Can ping yourself but not others.	In this case, you could successfully ping your own IP address, but you cannot reach a remote host. Possible causes are: □ The internet broadcast address might be set incorrectly. □ The routing statements might be incorrect. □ The name server, if used, might not be operating. To rectify this problem, you should: □ Check that the internet address that you are specifying is correct. □ Verify that the routing statements are accurate. □ If you are using a domain name server, initiate the server or ensure that it is available. □ Leave off the broadcast parameter in the **ifconfig** command.
Cannot ping yourself.	In this case, you cannot ping your own host. This could be caused by certain adapters that do not respond to an **arp** request from their own physical address. Therefore, the **arp** request will not be answered, and the ping will fail.
TCP/IP application gives message: **Cannot bind to control socket**	The possible causes are: □ CNTRL might not be running. □ The application you are starting is already running. Check the Task Manager to determine whether the application that you requested is already operating. If it is, another copy of the application cannot be started.
FTP data transfer hangs on large-file transmission.	The TCP/IP configuration parameter **MTU size** must be at least 40 bytes less than the adapter xmit buffer size as defined in your PROTOCOL.INI file.
Cannot communicate across LAN bridge (for example, 8209).	The TCP/IP configuration parameter **MTU size** must be 1500 or smaller when traversing LAN bridges.
Having routing problems.	Make sure the correct netmask is specified in the TCP/IP configuration. If you are specifying a subnet mask into the last byte, you must use **route add subnet** (rather than **route add net**). Do not use variable-length subnetting.
FTP file received is 1 byte larger than original.	ASCII file transfer might add an EOF character. If you determine that this is the case, this is a valid transfer and no harm will be caused by this extra byte.
Long response times.	The domain name server is probably timing out or might be inoperative. Check the domain name server to determine

	whether it is operating.
SMTP (mail) problems.	Some problems can be checked by reviewing ETC\SENDMAIL.ERR file for information.
TALK does not work.	This can be caused when the HOSTNAME environment variables are not being resolved correctly. Check the specification for HOSTNAME in CONFIG.SYS.
Telnet server does not seem to be working correctly.	Check that the TERM variable is set correctly at the client.
Control keys do not work when using Telnet client.	Toggle the localchars parameters to true. For the backspace character, toggle BS.

D.0 Appendix D. Messages and Codes

This appendix describes the messages and exit codes that are displayed in TCP/IP for OS/2. The messages are arranged alphabetically, grouped by command.

Subtopics

D.1 FINGER

Table 16 shows FINGER messages and codes.

Table 16. FINGER Messages and Codes

Exit Code	**Message**	**Explanation**
N/A	Unable to connect to **host**	The foreign host can be reached, but the FINGER server is not running. **Action:** Start the FINGER server on the foreign host.

D.2 FTP

Table 17 shows FTP messages and codes.

Table 17. FTP Messages and Codes

Exit Code	**Message**	**Explanation**
N/A	Error: 2	This message covers many error situations. The most likely reason is that a file name in the subcommand does not exist. **Action:** Check the directories on the local and remote hosts, using **dir**, and verify that the file names are correct.
1	Could not create a ftpds semaphore	This is a system problem. **Action:** Restart the system. If the problem persists, contact your IBM service representative.

D.3 FTP Server FTPDC--Exit Messages

These messages are printed by the FTPDC program. This program is started by the FTP server to handle client requests. The program exits with the code listed.

Table 18 shows FTP Server messages and codes.

Table 18. FTP Server FTPDC Exit Messages

Exit Code	Message	Explanation
0	Repeated login failures from **host**.	A user on another host is trying to log on to the FTP server and has been unsuccessful. **Action:** Verify that the user attempting to log on knows the correct user name and password.
1	ftpds:ioctl (trying to set socket to nonblocking)	This is a system error that should not occur. Contact the system administrator.
1	panic: all enough memory	This is a system error. Cannot allocate enough memory to read the TRUSERS file. This might mean that the TRUSERS file is too large or that too many programs are running. **Action:** Check the size of the TRUSERS file. Unless it is several megabytes, this should not be the problem. Reduce the number of programs running, restart the system, and restart TCP/IP. If the problem persists, contact your IBM service representative.
1	FTPDS.EXE is not running (when trying to get shared segment with FTPDS)	FTPDC.EXE was started with the FTP server not running, or the FTP server died during startup of FTPDC.EXE. **Action:** Verify that the FTP server is running. If it is not running, restart it.
1	Could not open attn semaphore Could not open a mail semaphore	No message, but this problem could occur because no socket exists. This is a system problem. **Action:** Restart the system and restart TCP/IP. If the problem persists, contact your IBM service representative.
2	Could not get pid	This is a system problem. **Action:** Restart the system and restart TCP/IP. If the problem persists, contact your IBM service representative.
10	panic: FTPDS did not respond within 30 secs	The FTP server is not operating. **Action:** Restart the FTP server and FTPDC.EXE.

D.4 FTP Server FTPDC--Nonexit Messages

These messages are printed by the FTPDC.EXE. The program does not exit when these errors occur.

Table 19 shows FTPDC Nonexit messages.

Table 19. FTP Server FTPDC Nonexit Messages

Exit Code	Message	Explanation
N/A	getpeername **program_name** trying to get peer information on connection	This is a system problem. **Action:** Restart the system and restart TCP/IP. If the problem persists, contact your IBM service representative.
N/A	getsockname **program_name** trying to get socket name information	This is a system problem. **Action:** Restart the system and restart TCP/IP. If the problem persists, contact your IBM service representative.
N/A	local diskette full. aborted	The local diskette is full. **Action:** Clean up the diskette space and try the action that failed again.

D.5 IFCONFIG

Table 20 shows IFCONFIG messages and codes.

Table 20. IFCONFIG Messages and Codes

Exit Code	**Message**	**Explanation**
N/A	**illegal parameters**: bad value	The syntax of the **ifconfig** command is incorrect. **Action:** Type **ifconfig** for help, or see the *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration*.

D.6 LPD

Table 21 shows LPD messages and codes.

Table 21. LPD Messages and Codes

Exit Code	**Message**	**Explanation**
1	Unknown Option '	The option ' specified on the command line is incorrect. **Action:** Check the parameters on the command line and respecify.
N/A	Error: receiving command from client: rc=%d	An error occurred during reading of command data from the client. **Action:** Check the connecting client to verify that the configuration is correct.
N/A	LPD: error receiving data (errno=%d)	An error occurred during reading of data from the client. **Action:** Check the connecting client to verify that the configuration is correct.
N/A	Print request aborted by Client!	The client requested that any print job that is currently being created be canceled.
N/A	Error: Invalid Control file!	The client has not sent a valid control file for the current job. **Action:** Check the connecting client to verify that their configuration is correct.
N/A	Error: Invalid Data file!	The client has not sent a valid data file for the current job. **Action:** Check the connecting client to verify that the configuration is correct.
N/A	Invalid socket specified!	An incorrect parameter was passed to **lpd** on the command line. **Action:** Check the parameters on the command line and re-enter the command.

D.7 LPQ

Table 22 shows LPQ messages and codes.

Table 22. LPQ Messages and Codes

Exit Code	**Message**	**Explanation**
1	No Printer was specified!	The **lpq** command was specified without a printer, nor was an LPR_PRINTER environment variable set. **Action:** Respecify the **lpq** command with the -p **printer** parameter, or set the environment variable LPR_PRINTER.
1	No Server was specified!	The **lpq** command was specified without a server, nor was an LPR_SERVER environment variable set. **Action:** Respecify the **lpq** command with the -s **server** parameter, or set the environment variable LPR_SERVER.
N/A	Unknown Option '	The option %c specified on the **lpq** command line is incorrect. **Action:** Respecify the **lpq** command with the correct option.
2	LPQ: Unknown server %s!	The server %s is not a valid server. **Action:** Respecify the **lpq** command line with the correct server.
1	printer: printer/tcp: unknown service	LPQ was unable to determine the socket number to connect to on the server to reach the remote print server. **Action:** Check your SERVICES file to verify that an entry exists for *printer*.
2	Unable to bind socket	The LPQ Protocol states that all LPQ requests must come from a port within the range of 721 to 731. Because of a time-out on port numbers, there is a chance to run out of available ports.

D.8 LPR

Table 23 shows LPR messages and codes.

Table 23. LPR Messages and Codes

Exit Code	Message	Explanation
N/A	Can't open %s:	Unable to open the file '%s' to send to the server. **Action:** Check the file name specified on the **lpr** command line.
N/A	Early EOF found transmitting file	An imbedded end-of-file marker was found in the file. **Action:** Try the **lpr** command specifying the -b option for binary files again.
N/A	Unable to connect to %s (errno=%d)!	LPR was unable to connect to the server %s. **Action:** Verify that the server specified is correct.
N/A	printer: printer/tcp: unknown service	LPR was unable to determine the socket number to connect to on the server to reach the remote print server. **Action:** Check your SERVICES file to verify that an entry exists for *printer*.
N/A	unknown host	LPR was unable to connect to the specified server. **Action:** Verify that the specified server is correct.
N/A	Unable to bind socket	The LPR Protocol states that all LPR requests must come from a port within the range of 721 to 731. **Action:** Because of a time-out on port numbers, there is a chance to run out of available ports when sending lots of files in a short period of time. LPR will try sending the file again if this does occur. If for some reason files do not get sent, you can adjust the number of retries as well as the delay between retries.
N/A	Unable to connect to foreign host	LPR was unable to connect to the specified server because no server was running on that host. **Action:** Verify that an LPD server is running on the remote host specified as the server.
N/A	Unable to receive response:	The specified server is not responding to LPR's requests. **Action:** Check the configuration of the LPD server that is running on the remote host specified as the server.
N/A	Server closed connection prematurely	The specified server closed the connection prematurely.

		Action: Check the configuration of the LPD server that is running on the remote host specified as the server.
N/A	Server Error: Server cannot open or write to printer	An incorrect printer name was specified. **Action:** Check the printer name specified on the **lpr** command line, and respecify if necessary.
N/A	Server Error: Out of storage space	The LPD server was unable to satisfy your request because it was out of storage. **Action:** Try to free up resources at the remote host running the LPD server.
N/A	Server Error: Unknown error:	The LPD server reported an unknown error back to LPR. **Action:** Check the configuration of the LPD server that is running on the remote host specified as the server.
N/A	Unable to send file %s after %d tries!	LPR was unable to print the file %s after %d retries. **Action:** Try the **lpr** command again, increasing either or both of the -r **retries** or -q **delay** parameters.
N/A	File not found: %s	LPR was unable to find the file %s. **Action:** Verify that the file exists on the local host, and respecify the **lpr** command if necessary.
1	Cannot specify -f and -b flags together!	The user specified both the -f and -b flags on the **lpr** command line. These flags are mutually exclusive. **Action:** Respecify the **lpr** command with only the required flag.
1	Invalid delay specified!	The user specified an out-of-range value for the -q **delay** parameter. **Action:** Respecify the **lpr** command with a value for **delay** between 0 and 30.
1	Invalid Number of retries specified!	The user specified an out-of-range value for the -r **retries** parameter. **Action:** Respecify the **lpr** command with a value for **retries** between 0 and 5.
1	No Printer was specified!	The **lpr** command was specified without a printer, nor was an LPR_PRINTER environment variable set. **Action:** Respecify the **lpr** command with the -p **printer** parameter, or set the environment variable LPR_PRINTER.

1	No Server was specified!	The **lpr** command was specified without a server, nor was an LPR_SERVER environment variable set. **Action:** Respecify the **lpr** command with the -s **server** parameter, or set the environment variable LPR_SERVER.
N/A	Warning: Sending unknown options to the server!	The **lpr** command was specified with unknown options. LPR will send these options to the specified server as part of the control file.

D.9 LPRM

Table 24 shows LPRM messages and codes.

Table 24. LPRM Messages and Codes

Exit Code	**Message**	**Explanation**
1	No Printer was specified!	The **lprm** command was specified without a printer, nor was an LPR_PRINTER environment variable set. **Action:** Respecify the **lprm** command with the -p **printer** parameter, or set the environment variable LPR_PRINTER.
1	No Server was specified!	The **lprm** command was specified without a server, nor was an LPR_SERVER environment variable set. **Action:** Respecify the **lprm** command with the -s **server** parameter, or set the environment variable LPR_SERVER.
1	No User or Agent was specified!	The **lprm** command was specified without an agent, nor was a **user** environment variable set. **Action:** Respecify the **lprm** command with the -a **agent** parameter, or set the environment variable user.
N/A	Unknown Option '	The option %c specified on the **lprm** command line is incorrect. **Action:** Respecify the **lprm** command with the correct option.
2	LPRM: Unknown server %s!	The server %s is not a valid server. **Action:** Respecify the **lprm** command line with the correct server.
1	printer: printer/tcp: unknown service	LPRM was unable to determine the socket number to connect to on the server to reach the remote print server. **Action:** Check your SERVICES file to verify that an entry exists for printer.
2	Unable to bind socket	The LPRM Protocol states that all LPRM requests must come from a port within the range of 721 to 731. Because of a time-out on port numbers, there is a chance to run out of available ports.

D.10 LPRMON

Table 25 shows LPRMON messages and codes.

Table 25. LPRMON Messages and Codes

Exit Code	**Message**	**Explanation**
1	Cannot specify -f and -b flags together!	The user specified both the -f and -b flags on the **lprmon** command line. These flags are mutually exclusive. **Action:** Respecify the **lprmon** command with only the required flag.
1	Invalid delay specified!	The user specified an out-of-range value for the -q **delay** parameter. **Action:** Respecify the **lprmon** command with a value for **delay** between 0 and 30.
1	Invalid Number of retries specified!	The user specified an out-of-range value for the -r **retries** parameter. **Action:** Respecify the **lprmon** command with a value for **retries** between 0 and 5.
1	No Printer was specified!	The **lprmon** command was specified without a printer, nor was an LPR_PRINTER environment variable set. **Action:** Respecify the **lprmon** command with the -p **printer** parameter, or set the environment variable LPR_PRINTER.
1	No Server was specified!	The **lprmon** command was specified without a server, nor was an LPR_SERVER environment variable set. **Action:** Respecify the **lprmon** command with the -s **server** parameter, or set the environment variable LPR_SERVER.
N/A	Warning: Sending unknown options to the server!	The **lprmon** command was specified with unknown options. The **lprmon** command will send these options to the specified server as part of the control file.

D.11 PORTMAP

Table 26 shows PORTMAP messages and codes.

Table 26. PORTMAP Messages and Codes

Exit Code	Message	Explanation
1	Error: portmap cannot create socket	The socket() procedure did not work in the PORTMAP routine. See the socket() call in the *IBM TCP/IP Version 2.0 for OS/2: Programmer's Reference* for specific errors. The socket() call can fail because of a protocol mismatch, because the domain is incorrectly specified, or because the sockets are all in use. Sockets remain active for a time-out period after they have been closed. **Action:** Verify that TCP or UDP is operating and that memory is available. Run the **netstat** command with the sockets option for more information.
1	Error: portmap cannot bind	The sockets bind() function did not work in the PORTMAP routine. See the bind() call in the *IBM TCP/IP Version 2.0 for OS/2: Programmer's Reference* for specific errors. The PORTMAP might have been stopped with the address still in use. The socket descriptor might have been altered by another function. **Action:** Clear the PORTMAP, and reinitiate. The PORTMAP workspace might have to be completely reset. Verify that memory is available. Run the **netstat** command with the sockets option for more information.
1	Error: could not do tcp_create	The rpc svctcp_create() function did not work in the PORTMAP routine. See the svctcp_create() call description in the *IBM TCP/IP Version 2.0 for OS/2: Programmer's Reference* for information. The socket descriptor might have been altered by another function or might not be available. **Action:** Verify that memory is available and allocated. Run the **netstat** command with the sockets option for more information.
1	Error: could not do udp_create	The rpc svcudp_create() function did not work in the PORTMAP routine. See the svcudp_create() call description in the *IBM TCP/IP Version 2.0 for OS/2: Programmer's Reference* for information. The socket

		descriptor might have been altered by another function or might not be available. **Action:** Verify that memory is available and allocated. Run the **netstat** command with the sockets option for more information.
N/A	Error: svc_sendreply1 svc_sendreply1 Set a program, version to port mapping	The rpc svc_sendreply() function did not work in the PORTMAP routine when it attempted to return information to the caller. See the rpc svc_sendreply() call description in the *IBM TCP/IP Version 2.0 for OS/2: Programmer's Reference* for information. The transport handle might have been altered by another function or might not be available. An xdr error might have occurred on the transfer, or a socket error might have occurred. **Action:** Verify that memory is available and allocated. Test the path with **rpcinfo**, **ping**, and **netstat**; make changes, and try again.
N/A	Error: svc_sendreply2 svc_sendreply2 -Remove a program, version to port mapping	The rpc svc_sendreply() function did not work in the PORTMAP routine when it attempted to return information to the caller. See the rpc svc_sendreply() call description in the *IBM TCP/IP Version 2.0 for OS/2: Programmer's Reference* for information. The transport handle might not be available or might have been altered by another function. An xdr error might have occurred on the transfer, or a socket error might have occurred. **Action:** Verify that memory is available and allocated. Test the path with **rpcinfo**, **ping**, and **netstat**; make changes, and try again.
N/A	Error: svc_sendreply3 svc_sendreply3 -Lookup the mapping for a program, version	The rpc svc_sendreply() function did not work in the PORTMAP routine when it attempted to return information to the caller. See the rpc svc_sendreply() call description in the *IBM TCP/IP Version 2.0 for OS/2: Programmer's Reference* for information. The transport handle might not be available or might have been altered by another function. An xdr error might have occurred on the transfer, or a socket error might have occurred. **Action:** Verify that memory is available and allocated. Test the path with **rpcinfo**, **ping**, and **netstat**; make changes, and try again.
N/A	Error: svc_sendreply4	The rpc svc_sendreply()

	svc_sendreply4 -Return the current set of mapped program, version	function did not work in the PORTMAP routine when it attempted to return information to the caller. See the rpc svc_sendreply() call description in the *IBM TCP/IP Version 2.0 for OS/2: Programmer's Reference* for information. The transport handle might not be available or might have been altered by another function. An xdr error might have occurred on the transfer, or a socket error might have occurred. **Action:** Verify that memory is available and allocated. Test the path with **rpcinfo**, **ping**, and **netstat**; make changes, and try again.

D.12 SENDMAIL--SENDMAIL.ERR Errors

Certain errors can be found in the ETC\SENDMAIL.ERR file.

Table 27 shows SENDMAIL.ERR Errors.

Table 27. SENDMAIL.ERR Messages and Codes

Exit Code	Message	Explanation
N/A	Cannot send mail	Warning message: (The recipient host's SMTP is down or cannot be reached through the network.) **Action:** No immediate action needs to be taken. SENDMAIL automatically stores the mail in the MQUEUE and tries the delivery again, based on the time you specified using the -qtime parameter. If the problem persists for several days, verify that the note is correct by checking the data files in the MQUEUE directory. You can either make corrections to the date files or delete them.
N/A	Timed out waiting for **hostname**	Warning message: (The connection to another SMTP server was lost.) **Action:** No action needs to be taken. SENDMAIL and other SMTP servers automatically try to deliver the mail again.
N/A	Line **number**: invalid rewrite line	The rewrite rule in the ETC\SENDMAIL.CF file on the specified line is incorrectly formed. **Action:** Edit the ETC\SENDMAIL.CF configuration file and correct the line. The three fields must be separated by tabs, not spaces. If you have a large number of these errors in your file, you probably used an editor that converts tabs to spaces, and have introduced a large number of errors into your configuration file. If you are unable to recover the changes, restore an older version of the SENDMAIL.CF file from the backups in the ETC directory or from the product diskettes.
N/A	Cannot change drive	SENDMAIL is unable to access either the MAIL or MQUEUE subdirectories. **Action:** Search the ETC\SENDMAIL.CF configuration file to find the MAIL and MQUEUE subdirectories that SENDMAIL expects to find. These directories are specified by the **OQ** and **Mlocal** parameters, respectively. Verify that these directories exist and that SENDMAIL has write access to them.
N/A	getrequests: cannot bind	SENDMAIL cannot bind to port 25, the port reserved for SMTP. This occurs if SENDMAIL is already running, or if SENDMAIL has previously terminated with an error and the port was not freed. **Action:** Verify that SENDMAIL is not already running. If port 25 cannot be freed, restart the machine.
N/A	fopen to inbox.ndx failed, inbox.ndx locked	SENDMAIL cannot open the MAIL\INBOX.NDX file because it is in use by another application, usually LaMail. **Action:** Normally the problem corrects itself, as the other application finishes accessing the INBOX.NDX file and allows SENDMAIL to

		update it. If the problem persists, the INBOX.NDX file might have become corrupted and needs to be erased. If the INBOX.NDX file must be erased, manually read the remaining mail in the MAIL directory to prevent the loss of information.
N/A	Sendmail gave up trying to open inbox.ndx, mailfile delivered, inbox not updated	SENDMAIL is unable to open the MAIL\INBOX.NDX file because it is in use by another application, usually LaMail. **Action:** Normally the problem corrects itself, as the other application finishes accessing the INBOX.NDX file and allows SENDMAIL to update it. If the problem persists, the INBOX.NDX file might have become corrupted and needs to be erased. If the INBOX.NDX file needs to be erased, manually read the remaining mail in the MAIL directory to prevent the loss of information.

D.13 SENDMAIL--Exit Codes

Table 28 shows SENDMAIL exit codes.

Table 28. sendmail Exit Codes

Exit Code	Message	Explanation
61	Connection refused	The connection to the domain name server could not be made at the present time. Either the host was not available or an invalid address was specified. **Action:** Verify that the domain name server IP address specified in the ETC subdirectory RESOLV file is valid, correct if necessary, and try again.
64	Command line usage error	The command was used incorrectly. For example, the command used the wrong number of arguments, an incorrect flag, or incorrect syntax. **Action:** Verify the command you entered, correct if necessary, and try again.
65	Data format error	The input data was incorrect. The data you entered should be used only as user's data, not as system files. **Action:** The mail file you are trying to send is in the wrong format. Verify the format, correct if necessary, and try again.
66	Cannot open input	An input mail file does not exist or cannot be read. This error could also include errors such as "No message" to a mailer. **Action:** Verify the file name of the input file, correct if necessary, and try again.
67	Addressee unknown	The user you specified does not exist. **Action:** Verify the target user ID, correct if necessary, and try again.
68	Host name unknown	The host you specified does not exist. This message occurs for mail addresses or network requests. **Action:** Check the target host to see if it is defined in the HOSTS file or name server.
69	Service unavailable	A service is unavailable. This message occurs if a support program or file does not exist, or it occurs as a general message when some task you wanted to perform does not work. **Action:** Please make a note of the circumstances and contact IBM support.

70	Internal software error	An internal software error has been detected. This message should be limited to non-operating-system errors. **Action:** Please make a note of the circumstances and call IBM support.
72	Critical OS file missing	A system file does not exist or cannot be opened. **Action:** Verify that the SENDMAIL.CF and SERVICES files exist in your ETC directory.
73	Can't create (user) output file	An output file you specified cannot be created. For example, you cannot open a file when receiving mail, your diskette is full, or a file is read-only. **Action:**
75	Temp failure; user is invited to retry	This is a temporary failure. For example, a mailer could not create a connection. **Action:** You should try to send the mail again.
76	Remote error in protocol	The remote system attempted an impossible task during a protocol exchange. **Action:** Verify that the target host is working correctly.

D.14 SNMP

Table 29 shows SNMP messages and codes.

Table 29. SNMP Messages and Codes

Exit Code	Message	Explanation
N/A	Unknown SNMP request type	There are only three SNMP request types: □ snmp_get □ snmp_getnext □ snmp_set **Action:** Verify that you have entered a valid type.
N/A	arptbl_setup:cannot get memory for arptblp1	In file MIB_AT.C, the c function calloc() allocates storage for the data structure **arp_ent**. **Action:** Run fewer applications, and restart the application that failed.
N/A	iftable_setup:cannot get memory for ifp	In file MIB_INTE.C, the c function calloc() allocates storage for the data structure **if_ent**. **Action:** Run fewer applications, and restart the application that failed.
N/A	if_addrsetup:cannot get memory for ipadr1	In file MIB_IPAD.C, the c function calloc() allocates storage for the data structure **ipadr_ent**. **Action:** Run fewer applications, and restart the application that failed.
N/A	iproute_setup:cannot get memory for iprouteptr	In file MIB_IPRO.C, the c function calloc() allocates storage for the data structure **iproute_ent**. **Action:** Run fewer applications, and restart the application that failed.

D.15 TALK

Table 30 shows TALK messages and codes.

Table 30. Talk Messages and Codes

Exit Code	**Message**	**Explanation**
N/A	talk: **hostname**:	Your HOSTNAME environment variable is defined as **hostname**, which cannot be resolved. **Action:** Define **hostname** in your HOSTS file or name server.
N/A	talk: **host**: invalid host name	The host is probably valid, but it cannot be resolved. **Action:** Define the **host** in your HOSTS file or name server.

D.16 Telnet Server

Table 31 shows TELNET messages and codes.

Table 31. Telnet Server Messages and Codes

Exit Code	Message	Explanation
	Invalid password.	You entered an incorrect password. **Action:** Contact your system administrator for password information.
	Telnetd: panic state=**state**	The Telnet state machine is in an unknown state. **Action:** Verify the communication parameters between client and server. For example, **SYNC** and **FLUSH**
N/A	Unable to start up a Telnet session to service the client	The server was unable to start up a session to process the connecting client's requests. This is usually because the server host has run out of resources.
N/A	Unable to start shell specified in COMSPEC variable in config.sys!	The Telnet server was unable to start up a command processor as specified by the COMSPEC environment variable. **Action:** Check your CONFIG.SYS to verify the value of this environment variable.
N/A	Login failure: {Hard error abor Trap opera DosKillPro Unknown Fa	An unusual error occurred iduring a client login. ess l**Action:** Verify that the correct LOGIN.EXE in the TCPIP\BIN directory is the login that is being run. (Verify that this is the first executable with the name **login** in your PATH.)
	Could NOT execute login.exe command!	The server was unable to run the login program to allow clients access. **Action:** Verify that your PATH environment variable is correct.

E.0 Appendix E. Management Information Base (MIB) Objects

This appendix lists the objects for the following groups defined by the Management Information Base (MIB)-II.

- System
- Interfaces
- Address translation
- Internet Protocol (IP)
- Internet Control Message Protocol (ICMP)
- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)
- Exterior Gateway Protocol (EGP)
- Simple Network Management Protocol (SNMP)

The object types are defined using the following fields:

Object A textual string (referred to as the *object descriptor*) and the administratively assigned name (referred to as the *object identifier*).

ASN.1 Notation The abstract syntax notation that represents the object identifier.

Syntax The data type of the MIB object.

Definition A description of the MIB object.

- Read-only allows viewing of the entry.
- Read-write allows viewing and writing to the entry.
- Not-accessible does not allow reading or writing to the entry.

Subtopics

E.1 System Group

Table 32 lists the objects in the system group. The system objects identify the type of system with a text description and the vendor-assigned object-id as an identification to the type of SNMP agent.

rules=horiz frame=none headhi=2 concat=yes split=yes.

Table 32. Implementation of the System Group

Object and ASN.1 Notation	**Syntax**	**Definition**
SYSTEM GROUP 1.3.6.1.2.1.1		
sysDescr { system 1 } 1.3.6.1.2.1.1.1.0	DisplayString	A description of the entry. This value should include the full name and version identification of the system's hardware type, software operating system, and networking software. This description must contain only printable ASCII characters. Access is read-only.
sysObjectID { system 2 } 1.3.6.1.2.1.1.2.0	Object Identifier	The vendor's authorization identification of the network management subsystem contained in the entry. This value is allocated within the structure for management information (SMI) enterprise's subtree (1.3.6.1.4.1) and provides an easy and clear means for determining *what kind of box* is being managed. For example, if vendor *Stones, Inc.* was assigned the subtree 1.3.6.1.4.1.42, it could assign the identifier 1.3.6.1.4.1.42.1.1 to the router *Fred Router*. Access is read-only.
sysUpTime { system 3 } 1.3.6.1.2.1.1.3.0	TimeTicks	The time (in hundredths of a second) since the network management portion of the system was last started. Access is read-only.
sysContact { system 4 } 1.3.6.1.2.1.1.4.0	DisplayString	The textual identification of the contact person for this managed node, together with information on how to contact this person. Access is read-write.
sysName { system 5 }	DisplayString	An administratively assigned name for this managed node. By convention, this

1.3.6.1.2.1.1.5.0		is the node's fully qualified domain name. Access is read-write.
sysLocation { system 6 } 1.3.6.1.2.1.1.6.0	DisplayString	The physical location of this node (for example, telephone closet, 3rd floor). Access is read-write.
sysServices { system 7 } 1.3.6.1.2.1.1.7.0	Integer	A value that indicates the set of services that this entity primarily offers. The value is a sum. This sum initially takes the value zero, then, for each layer, L, in the range 1 through 7, for which this node performs transactions, 2 raised to (L - 1) is added to the sum. For example, a node that performs primarily routing functions would have a value of 4 (2**(3-1)). In contrast, a node that is a host offering application services would have a value of 72 (2**(4-1) + 2**(7-1)). In the context of the Internet suite of protocols, values should be calculated accordingly: **Layer Functions** 1 Physical (for example, repeaters) 2 Datalink/subnetwork (for example, bridges) 3 Internet (for example, IP gateways) 4 End-to-end (for example, IP hosts) 7 Applications (for example, mail relays) For systems including OSI protocols, layers 5 and 6 can also be counted. Access is read-only.

E.2 Interfaces Group

Table 33 lists the objects in the interfaces group. The interfaces objects are a set of entries for each network interface below the IP layer that can send and receive datagrams.

Table 33. Implementation of the Interfaces Group

Object and ASN.1 Notation	**Syntax**	**Definition**
INTERFACES GROUP 1.3.6.1.2.1.2		
ifNumber { interfaces 1 } 1.3.6.1.2.1.2.1.	Integer	The number of network interfaces (regardless of their current state) present on this system. Access is read-only.
ifTable { interfaces 2 } 1.3.6.1.2.1.2.2	SEQUENCE of IfEntry	A list of interface entries. The number of entries is given by the value of ifNumber. Not accessible.
ifEntry { ifTable 1 } 1.3.6.1.2.1.2.2.1	IfEntry ::= SEQUENCE ifIndex INTEGER, ifDescr DisplayString, ifType INTEGER, ifMtu INTEGER, ifSpeed Gauge, ifPhysAddress PhysAddress, ifAdminStatus INTEGER, ifOperStatus INTEGER, ifLastChange TimeTicks, ifInOctets Counter, ifInUcastPkts Counter, ifInNUcastPkts Counter, ifInDiscards Counter, ifInErrors Counter, ifInUnkownProtos Counter,	An interface entry that contains objects at the subnetwork layer and below for a particular interface. Not accessible.
ifEntry (Cont.)	ifOutOctets Counter, ifOutUcastPkts Counter, ifOutNUcastPkts Counter, ifOutDiscards Counter, ifOutErrors Counter, ifOutQLen Gauge ifSpecific Object Identifier,	

ifIndex { ifEntry 1 } 1.3.6.1.2.1.2.2.1.1.	Integer	A unique value for each interface. Values range between 1 and the value of ifNumber. The value for each interface must remain constant for at least one start of the systems network management system to the next start. Access is read-only.
ifDescr { ifEntry 2 } 1.3.6.1.2.1.2.2.1.2.	DisplayString	A text string containing information about the interface. This string should include the name of the manufacturer, the product name, and the version of the hardware interface. Access is read-only.
ifType { ifEntry 3 } 1.3.6.1.2.1.2.2.1.3.	Integer other (1), regular 1822 (2), hdh1822 (3), ddn-x25 (4), rfc877-x25 (5), ethernet-csmacd (6), iso88023-csmacd (7), iso88024-tokenBus (8), iso88025-tokenRing (9), iso88026-kman (10), starLan (11), proteon-10Mbit (12), proteon-80Mbit (13), hyperchannel (14), fddi (15), lapb (16), sdlc (17), tl-carrier (18), cept (19), basicISDN (20), primaryISDN (21), propPointToPointSerial (22), terminalServer-asypcPort (23), softwareLoopback (24), eon (25), ethernet-3Mbit (26), nsip (27), slip (28) ultra (29), ds3 (30), sip (31), frame-relay (32),	The type of interface, distinguished according to the physical, link, and network protocols immediately *below* IP in the protocol stack. Access is read-only.
ifMtu { ifEntry 4 } 1.3.6.1.2.1.2.2.1.4.	Integer	The size of the largest datagram that can be sent or received on the interface, specified in octets. For interfaces that

		are used for transmitting IP datagrams, this is the size of the largest IP datagram that can be sent on the interface. Access is read-only.
ifSpeed { ifEntry 5 } 1.3.6.1.2.1.2.2.1.5.	Gauge	An estimate of the interface's current bandwidth in bits per second. For interfaces that do not vary in bandwidth or for those where no accurate estimate can be made, this object should contain the nominal bandwidth. Access is read-only.
ifPhysAddress { ifEntry 6 } 1.3.6.1.2.1.2.2.1.6.	PhysAddress	The interface's address at the protocol layer immediately *below* IP in the protocol stack. For interfaces that do not have such an address (for example, a serial line), this object should contain an octet string of length zero. Access is read-only.
ifAdminStatus { ifEntry 7 } 1.3.6.1.2.1.2.2.1.7.	Integer up (1), down (2), testing (3)	The desired state of the interface. The testing (3) state indicates that operational packets cannot be passed. Access is read-write.
ifOperStatus { ifEntry 8 } 1.3.6.1.2.1.2.2.1.8.	INTEGER up (1), down (2), testing (3)	The current operational state of the interface. The testing (3) state indicates that operational packets cannot be passed. Access is read-only.
ifLastChange { ifEntry 9 }	TimeTicks	The value of sysUpTime when the interface entered its

1.3.6.1.2.1.2.2.1.9.		current operational state. If the current state was entered before the last start-up of the local network management subsystem, then this object contains a value of zero. Access is read-only.
ifInOctets { ifEntry 10 } 1.3.6.1.2.1.2.2.1.10.	Counter	The total number of octets received on the interface, including framing characters. Access is read-only.
ifInUcastPkts { ifEntry 11 } 1.3.6.1.2.1.2.2.1.11.	Counter	The number of subnetwork-unicast packets delivered to a higher-layer protocol. Access is read-only.
IfInNUcastPkts { ifEntry 12 } 1.3.6.1.2.1.2.2.1.12.	Counter	The number of non-unicast (for example, subnetwork-broadcast or subnetwork-multicast) packets delivered to a higher-layer protocol. Access is read-only.
ifInDiscards { ifEntry 13 } 1.3.6.1.2.1.2.2.1.13.	Counter	The number of inbound packets that were chosen to be discarded. This occurs even though errors had not been detected to prevent their delivery to a higher-layer protocol. One possible reason for discarding such a packet could be to free buffer space. Access is read-only.
ifInErrors { ifEntry 14 } 1.3.6.1.2.1.2.2.1.14.	Counter	The number of inbound packets that contain errors that prevent delivery to a

		higher-layer protocol. Access is read-only.
ifInUnknownProtos { ifEntry 15 } 1.3.6.1.2.1.2.2.1.15.	Counter	The number of packets received through the interface that were discarded because of an unknown or unsupported protocol. Access is read-only.
ifOutOctets { ifEntry 16 } 1.3.6.1.2.1.2.2.1.16.	Counter	The total number of octets transmitted out of the interface, including framing characters. Access is read-only.
ifOutUcastPkts { ifEntry 17 } 1.3.6.1.2.1.2.2.1.17.	Counter	The total number of packets that higher-level protocols requested be transmitted to a subnetwork-unicast address. This includes those that were discarded or not sent. Access is read-only.
ifOutNUcastPkts { ifEntry 18 } 1.3.6.1.2.1.2.2.1.18.	Counter	The total number of packets that higher-level protocols request to be transmitted to a non-unicast address For example, a subnetwork-broadcast or subnetwork-multicast address, including those that were discarded or not sent. Access is read-only.
ifOutDiscards { ifEntry 19 } 1.3.6.1.2.1.2.2.1.19.	Counter	The number of outbound packets that were chosen to be discarded even though errors had not been detected to prevent their being

		transmitted. One reason for discarding such a packet could be to free buffer space. Access is read-only.
ifOutErrors { ifEntry 20 } 1.3.6.1.2.1.2.2.1.20.	Counter	The number of outbound packets that could not be transmitted because of errors. Access is read-only.
ifOutQLen { ifEntry 21 } 1.3.6.1.2.1.2.2.1.21.	Gauge	The length of the output packet queue (in packets). Access is read-only.
ifSpecific { ifEntry 22 } 1.3.6.1.2.1.2.2.1.22.	Object identifier	A reference to MIB definitions specific to the particular media being used to realize the interface. For example, if the interface is realized by an Ethernet, then the value of this object refers to a document defining objects specific to Ethernet. If this information is not present, its value should be set to the OBJECT IDENTIFIER (0 0), which is a syntactically valid object identifier, and any conformant implementation of ASN.1 and BER must be able to generate and recognize this value. Access is read-only.

E.3 Address Translation Group

Table 34 lists the objects in the address translation group. The address translation group consists of one table, which shows the mapping between IP addresses and physical addresses.

Table 34. Implementation of the Address Translation Group

Object and ASN.1 Notation	**Syntax**	**Definition**
ADDRESS TRANSLATION GROUP 1.3.6.1.2.1.3		
atTable { at 1 } 1.3.6.1.2.1.3.1	Sequence of AtEntry	The Address Translation tables contain the NetworkAddress to *physical* address equivalences. Some interfaces do not use translation tables to determine address equivalences (for example, DDN-X.25 has an algorithmic method). If all interfaces are of this type, then the Address Translation table is empty, that is, it has zero entries. Not accessible.
atEntry { atTable 1 } 1.3.6.1.2.1.3.1.1	AtEntry ::= SEQUENCE atIfIndex INTEGER, atPhysAddress PhysAddress, atNetAddress NetworkAddress	Each entry contains one NetworkAddress to the physical address equivalent. Not accessible.
atIfIndex { atEntry 1 } 1.3.6.1.2.1.3.1.1.1.	Integer	The interface on which this entry's equivalence is effective. The interface is identified by a particular value of this index and is the same interface identified by the same value of ifIndex. Access is read-write.
atPhysAddress { atEntry 2 } 1.3.6.1.2.1.3.1.1.2.	PhysAddress	The media-dependent *physical* address. Access is read-write.
atNetAddress { atEntry 3 } 1.3.6.1.2.1.3.1.1.3.	NetworkAddress	The NetworkAddress (for example, the IP address) corresponding to the media-dependent *physical* address. Access is read-write.

E.4 IP Group

Table 35 lists the objects in the IP group. The IP objects are the statistics and gateway routing tables for the IP layer.

Table 35. Implementation of the IP Group

Object and ASN.1 Notation	**Syntax**	**Definition**
IP GROUP 1.3.6.1.2.1.4		
ipForwarding { ip 1 } 1.3.6.1.2.1.4.1.	Integer gateway (1), -- entry forwards datagrams host (2) -- entry does NOT forward datagrams	Indicates if this entry is acting as an IP gateway for the forwarding of datagrams received by, but not addressed to, this entry. IP gateways forward datagrams; hosts do not, except those source-routed through the host. Access is read-write.
ipDefaultTTL { ip 2 } 1.3.6.1.2.1.4.2.	Integer	When a TTL value is not supplied by the transport layer protocol, the default value inserts into the time-to-live field of the IP header of datagrams that originate at this entry. Access is read-write.
ipInReceives { ip 3 } 1.3.6.1.2.1.4.3.	Counter	The number of input datagrams received from interfaces, including those received in error. Access is read-only.
ipInHdrErrors { ip 4 } 1.3.6.1.2.1.4.4.	Counter	The number of input datagrams discarded because of errors in their IP headers. For example, bad checksums, mismatched version number, format errors, time-to-live exceeded, and processing

		errors in IP options. Access is read-only.
ipInAddrErrors { ip 5 } 1.3.6.1.2.1.4.5.	Counter	The number of input datagrams discarded because the IP address in their IP header's destination field was not a valid address to be received at this entry. This count includes invalid addresses (for example, 0.0.0.0), addresses of unsupported classes (for example, Class E), and destination addresses that were not local (for example, IP gateways). Access is read-only.
ipForwDatagrams { ip 6 } 1.3.6.1.2.1.4.6.	Counter	The number of input datagrams for which this entry is not their final IP destination. As a result, an attempt is made to find a route to their final destination. For entries that do not act as IP gateways, this count includes only those packets that are source-routed successfully through this entry. Access is read-only.
ipInUnknownProtos { ip 7 } 1.3.6.1.2.1.4.7.	Counter	The number of locally-addressed datagrams received successfully, but discarded because of an unknown or unsupported protocol. Access is read-only.
ipInDiscards	Counter	The number of

{ ip 8 } 1.3.6.1.2.1.4.8.		input IP datagrams that are processed without problems but are discarded (for example, for lack of buffer space). This count does not include any datagrams discarded while awaiting reassembly. Access is read-only.
ipInDelivers { ip 9 } 1.3.6.1.2.1.4.9.	Counter	The number of input datagrams successfully delivered to IP user-protocols including ICMP. Access is read-only.
ipOutRequests { ip 10 } 1.3.6.1.2.1.4.10.	Counter	The number of IP datagrams that are supplied to IP and ICMP in requests for transmission. This count does not include datagrams counted in ipForwDatagrams Access is read-only.
ipOutDiscards { ip 11 } 1.3.6.1.2.1.4.11.	Counter	The number of output IP datagrams that transmit without problems but are discarded (for example, for lack of buffer space). This count includes datagrams in ipForwDatagrams that meet this discard criterion. Access is read-only.
ipOutNoRoutes { ip 12 } 1.3.6.1.2.1.4.12.	Counter	The number of IP datagrams discarded because no route can transmit them to their destination. This count includes packets in ipForwDatagrams that meet this

		no-route criterion. Access is read-only.
ipReasmTimeout { ip 13 } 1.3.6.1.2.1.4.13.	Integer	The maximum number of seconds that received fragments are held while awaiting reassembly at this entry. Access is read-only.
ipReasmReqds { ip 14 } 1.3.6.1.2.1.4.14.	Counter	The number of IP fragments that are received and need to be reassembled at this entry. Access is read-only.
ipReasmOKs { ip 15 } 1.3.6.1.2.1.4.15.	Counter	The number of IP datagrams reassembled without problems. Access is read-only.
ipReasmFails { ip 16 } 1.3.6.1.2.1.4.16.	Counter	The number of failures detected by the IP reassembly algorithm. This is not a count of discarded IP fragments because some algorithms can lose track of the number of fragments by combining them as they are received. Access is read-only.
ipFragOKs { ip 17 } 1.3.6.1.2.1.4.17.	Counter	The number of IP datagrams that have fragmented at this entry without problems. Access is read-only.
ipFragFails { ip 18 } 1.3.6.1.2.1.4.18.	Counter	The number of IP datagrams that should have been fragmented at this entry, but were not because their *Don't Fragment* flag was set. Access is

		read-only.
ipFragCreates { ip 19 } 1.3.6.1.2.1.4.19.	Counter	The number of IP datagram fragments that have been generated, because of fragmentation at this entry. Access is read-only.
ipAddrTable { ip 20 } 1.3.6.1.2.1.4.20	SEQUENCE OF IpAddrEntry	A table that contains addressing information relevant to this entry's IP addresses. Not accessible.
ipAddrEntry { ipAddrTable 1 } 1.3.6.1.2.1.4.20.1	IpAddrEntry ::= SEQUENCE ipAdEntAddr IpAddress, ipAdEntIfIndex INTEGER, ipAdEntNetMask IpAddress, ipAdEntBcastAddr INTEGER ipAdEntReasmMaxSize INTEGER (0..65535)	The addressing information for one of this entry's IP addresses. Not accessible.
ipAdEntAddr { ipAddrEntry 1 } 1.3.6.1.2.1.4.20.1.1.	IpAddress	The IP address pertaining to this entry's addressing information. Access is read-only.
ipAdEntIfIndex { ipAddrEntry 2 } 1.3.6.1.2.1.4.20.1.2.	Integer	The index value that uniquely identifies the interface to which this entry is applicable. The interface identified by a particular value of this index is the same interface that is identified by the same value of ifIndex. Access is read-only.
ipAdEntNetMask { ipAddrEntry 3 } 1.3.6.1.2.1.4.20.1.3.	IpAddress	The subnet mask associated with the IP address of this entry. The value of the mask is an IP address with all the network bits set to 1 and all the hosts bits set to 0.

		Access is read-only.
ipAdEntBcastAddr { ipAddrEntry 4 } 1.3.6.1.2.1.4.20.1.4.	Integer	The value of the least-significant bit in the IP broadcast address used for sending datagrams on the (logical) interface associated with the IP address of this entry. For example, when the internet standard all-ones broadcast address is used, the value is 1. Access is read-only.
ipAdEntReasmMaxSize { ipAddrEntry 5 } 1.3.6.1.2.1.4.20.1.5	Integer (0..65535)	The size of the largest IP datagram which this entity can reassemble from incoming IP fragmented datagrams received on this interface. Access is read-only.
ipRoutingTable { ip 21 } 1.3.6.1.2.1.4.21	SEQUENCE OF IpRouteEntry	This entry's IP routing table. Not accessible.
ipRouteEntry { ipRoutingTable 1 } 1.3.6.1.2.1.4.21.1	IpRouteEntry ::= SEQUENCE ipRouteDest IpAddress, ipRouteIfIndex INTEGER, ipRouteMetric1 INTEGER, ipRouteMetric2 INTEGER, ipRouteMetric3 INTEGER, ipRouteMetric4 INTEGER, ipRouteNextHop IpAddress, ipRouteType INTEGER, ipRouteProto INTEGER, ipRouteAge INTEGER ipRouteMask IpAddress ipRouteMetric5 INTEGER ipRouteInfo ObjectIdentifier	A route to a particular destination.
ipRouteDest { ipRouteEntry 1 }	IpAddress	The destination IP address of

1.3.6.1.2.1.4.21.1.1.		this route. An entry with a value of 0.0.0.0 is considered a default route. Multiple default routes can appear in the table, but access to these multiple entries depends on the table-access mechanisms defined by the network management protocol in use. Access is read-write.
ipRouteIfIndex { ipRouteEntry 2 } 1.3.6.1.2.1.4.21.1.2.	Integer	The index value that uniquely identifies the local interface through which the next hop of this route should be reached. The interface identified by a particular value of this index is the same interface that is identified by the same value of ifIndex. Access is read-write.
ipRouteMetric1 { ipRouteEntry 3 } 1.3.6.1.2.1.4.21.1.3.	Integer	The primary routing metric for this route. The semantics of this metric are determined by the routing protocol specified in the route's ipRouteProto value. If this metric is not used, its value should be set to -1. Access is read-write.
ipRouteMetric2 { ipRouteEntry 4 } 1.3.6.1.2.1.4.21.1.4.	Integer	An alternative routing metric for this route. The semantics of this metric are determined by the routing-protoco specified in the route's ipRouteProto value. If

		this metric is not used, its value should be set to -1. Access is read-write.
ipRouteMetric3 { ipRouteEntry 5 } 1.3.6.1.2.1.4.21.1.5.	Integer	An alternative routing metric for this route. The semantics of this metric are determined by the routing protocol specified in the route's ipRouteProto value. If this metric is not used, its value should be set to -1. Access is read-write.
ipRouteMetric4 { ipRouteEntry 6 } 1.3.6.1.2.1.4.21.1.6.	Integer	An alternative routing metric for this route. The semantics of this metric are determined by the routing-protoco specified in the route's ipRouteProto value. If this metric is not used, its value should be set to -1. Access is read-write.
ipRouteNextHop { ipRouteEntry 7 } 1.3.6.1.2.1.4.21.1.7.	IpAddress	The IP address of the next hop of this route. Access is read-write.
ipRouteType { ipRouteEntry 8 } 1.3.6.1.2.1.4.21.1.8.	Integer other (1), invalid (2), direct (3), remote (4)	The type of route. Access is read-write.
ipRouteProto { ipRouteEntry 9 } 1.3.6.1.2.1.4.21.1.9.	Integer other (1), local (2), netmgmt (3), icmp (4), egp (5), ggp (6), hello (7), rip (8), is-is (9), es-is (10), ciscoIgrp (11), bbnSpfIgp (12), ospf (13)	The routing mechanism by which this route was learned. Inclusion of values for gateway routing protocols is not intended to imply that hosts should support those protocols. Access is read-only.

ipRouteAge { ipRouteEntry 10 } 1.3.6.1.2.1.4.21.1.10.	Integer	The number of seconds since this route was last updated or otherwise determined to be correct. Note that semantics of *too old* cannot be implied, except through knowledge of the routing protocol by which the route was learned. Access is read-write.
ipRouteMask { ipRouteEntry 11 } 1.3.6.1.2.1.4.21.1.11.	IpAddress	Indicate the mask to be logical-ANDed with the destination address before being compared to the value in the ipRouteDest field. For those systems that do not support arbitrary subnet masks, an agent constructs the value of the ipRouteMask by determining whether the value of the correspondent ipRouteDest field belongs to a class-A, B, or C network, and then using one of: **ask** **etwork** 55.0.0.0 lass-A 55.255.0.0 lass-B 55.255.255.0 lass-A If the value of the ipRouteDest is 0.0.0.0 (default route), then the mask value is also 0.0.0.0. All IP routing subsystems implicitly use this mechanism. Access is read-write.
ipRouteMetric5 { ipRouteEntry 12 } 1.3.6.1.2.1.4.21.1.12.	Integer	An alternate routing metric for this route. The semantics of this metric are determined by the routing-protoco

		specified in the route's ipRouteProto value. If this metric is not used, its value should be set to -1. Access is read-write.
ipRouteInfo { ipRouteEntry 13 } 1.3.6.1.2.1.4.21.1.13.	Object Identifier	A reference to MIB definitions specific to the particular routing protocol that is responsible for this route, as determined by the value specified in the route's ipRouteProto value. If this information is not present, its value should be set to the OBJECT IDENTIFIER (0 0), which is a syntactically valid object identifier, and any conformant implementation of ASN.1 and BER must be able to generate and recognize this value. Access is read-only.
IP Address Translation Table 1.3.6.1.2.1.4.22		The IP address translation table contains the ipAddress to **physical** address equivalences. Some interfaces do not use translation tables for determining address equivalences (for example, DDN-X.25 has an algorithmic method); if all interfaces are of this type, then the Address Translation Table is empty, that is, it has zero entries. Not accessible.

ipNetToMediaTable { ip 22 } 1.3.6.1.2.1.4.22.1	SEQUENCE OF IpNetToMediaEntry	The IP Address Translation table used for mapping from IP addresses to physical addresses. Not accessible.
ipNetToMediaEntry	ipNetToMediaIfIndex INTEGER, ipNetToMediaPhysAddress PhysAddress, ipNetToMediaNetAddress IpAddress, ipNetToMediaType INTEGER	Each entry contains one ipAddress to physical address equivalence. Not accessible.
ipNetToMediaIfIndex { ipNetToMediaEntry 1 } 1.3.6.1.2.1.4.22.1.1	Integer	The interface on which this entry's equivalence is effective. The interface identified by a particular value of this index is the same interface as identified by the same value of ifIndex. Access is read-write.
ipNetToMediaPhysAddress { ipNetToMediaEntry 2 } 1.3.6.1.2.1.4.22.1.2	Octet string	The media-dependent **physical** address. Access is read-write.
ipNetToMediaNetAddress { ipNetToMediaEntry 3 } 1.3.6.1.2.1.4.22.1.3	ipAddress	The IpAddress corresponding to the media-dependent **physical** address. Access is read-write.
ipNetToMediaType { ipNetToMediaEntry 4 } 1.3.6.1.2.1.4.22.1.4	Integer	The type of mapping. Setting this object to the value invalid (2) has the effect of invalidating the corresponding entry in the ipNetToMediaTable. That is, it effectively disassociates the interface identified with that entry from the mapping identified with that entry. It is an

		implementation-specific matter as to whether the agent removes an invalidated entry from the table. Accordingly, management stations must be prepared to receive tabular information from agents that corresponds to entries not currently in use. Proper interpretation of such entries requires examination of the relevant ipNetToMediaTyp object. Access is read-write.	

E.5 ICMP Group

Table 36 lists the objects in the ICMP group. The ICMP objects are the input and output error and control message statistics for the IP layer.

rules=horiz frame=none headhi=2 concat=yes split=yes.

Table 36. Implementation of the ICMP Group

Object and ASN.1 Notation	**Syntax**	**Definition**
ICMP GROUP 1.3.6.1.2.1.5		
icmpInMsgs { icmp 1 } 1.3.6.1.2.1.5.1.0	Counter	The number of ICMP messages that the entity received. This counter includes all those counted by icmpInErrors. Access is read-only.
icmpInErrors { icmp 2 } 1.3.6.1.2.1.5.2.0	Counter	The number of ICMP messages that the entity received. and determines ICMP specific errors (bad ICMP checksums, bad length). Access is read-only.
icmpInDestUnreachs { icmp 3 } 1.3.6.1.2.1.5.3.0	Counter	The number of ICMP Destination Unreachable messages received. Access is read-only.
icmpInTimeExcds { icmp 4 } 1.3.6.1.2.1.5.4.0	Counter	The number of ICMP Time Exceeded messages received. Access is read-only.
icmpInParmProbs { icmp 5 } 1.3.6.1.2.1.5.5.0	Counter	The number of ICMP Parameter Problem messages received. Access is read-only.
icmpInSrcQuenchs { icmp 6 } 1.3.6.1.2.1.5.6.0	Counter	The number of ICMP Source Quench messages received. Access is read-only.
icmpInRedirects { icmp 7 } 1.3.6.1.2.1.5.7.0	Counter	The number of ICMP Redirect messages received. Access is read-only.
icmpInEchos { icmp 8 } 1.3.6.1.2.1.5.8.0	Counter	The number of ICMP Echo (request) messages received. Access is read-only.
icmpInEchoReps { icmp 9 } 1.3.6.1.2.1.5.9.0	Counter	The number of ICMP Echo Reply messages received. Access is read-only.
icmpInTimestamps { icmp 10 } 1.3.6.1.2.1.5.10.0	Counter	The number of ICMP Timestamp (request) messages received. Access is read-only.
icmpInTimestampReps { icmp 11 } 1.3.6.1.2.1.5.11.0	Counter	The number of ICMP Timestamp Reply messages received. Access is read-only.

icmpInAddrMasks { icmp 12 } 1.3.6.1.2.1.5.12.0	Counter	The number of ICMP Address Mask Request messages received. Access is read-only.
icmpInAddrMaskReps { icmp 13 } 1.3.6.1.2.1.5.13.0	Counter	The number of ICMP Address Mask Reply messages received. Access is read-only.
icmpOutMsgs { icmp 14 } 1.3.6.1.2.1.5.14.0	Counter	The number of ICMP messages sent. This counter includes icmpOutErrors. Access is read-only.
icmpOutErrors { icmp 15 } 1.3.6.1.2.1.5.15.0	Counter	The number of ICMP messages that this entity did not send because of problems within ICMP (for example, no buffers). This value should not include errors outside the ICMP layer--for example, the inability of IP to route the resulting datagram. In some implementations, there might not be error types that contribute to the counter's value. Access is read-only.
icmpOutDestUnreachs { icmp 16 } 1.3.6.1.2.1.5.16.0	Counter	The number of ICMP Destination Unreachable messages sent. Access is read-only.
icmpOutTimeExcds { icmp 17 } 1.3.6.1.2.1.5.17.0	Counter	The number of ICMP Time Exceeded messages sent. Access is read-only.
icmpOutParmProbs { icmp 18 } 1.3.6.1.2.1.5.18.0	Counter	The number of ICMP Parameter Problem messages sent. Access is read-only.
icmpOutSrcQuenches { icmp 19 } 1.3.6.1.2.1.5.19.0	Counter	The number of ICMP Source Quench messages sent. Access is read-only.
icmpOutRedirects { icmp 20 } 1.3.6.1.2.1.5.20.0	Counter	The number of ICMP Redirect messages sent. For a host, this object is always zero; hosts do not send redirects. Access is read-only.
icmpOutEchos { icmp 21 } 1.3.6.1.2.1.5.21.0	Counter	The number of ICMP Echo (request) messages sent. Access is read-only.
icmpOutEchoReps { icmp 22 } 1.3.6.1.2.1.5.22.0	Counter	The number of ICMP Echo Reply messages sent. Access is read-only.
icmpOutTimestamps { icmp 23 } 1.3.6.1.2.1.5.23.0	Counter	The number of ICMP Timestamp (request) messages sent. Access is read-only.
icmpOutTimestampReps	Counter	The number of ICMP Timestamp

{ icmp 24 } 1.3.6.1.2.1.5.24.0		Reply messages sent. Access is read-only.
icmpOutAddrMasks { icmp 25 } 1.3.6.1.2.1.5.25.0	Counter	The number of ICMP Address Mask Request messages sent. Access is read-only.
icmpOutAddrMasksReps { icmp 26 } 1.3.6.1.2.1.5.26.0	Counter	The number of ICMP Address Mask Reply messages sent. Access is read-only.

E.6 TCP Group

Table 37 lists the objects in the TCP group. The TCP objects are the data transmission statistics and connection data for the TCP layer.

Note: Objects that represent information about a particular TCP connection are transient; the objects exist only as long as the specified connection is in use.

Table 37. Implementation of the TCP Group

Object and ASN.1 Notation	**Syntax**	**Definition**
TCP GROUP 1.3.6.1.2.1.6		
tcpRtoAlgorithm { tcp 1 } 1.3.6.1.2.1.6.1.	Integer other (1) -- none of the following constant (2) -- a contant rto rsre (3) --MIL-STD-1778 vanj (4) -- Van Jacobson's algorithm	The algorithm used to determine the time-out value used for retransmitting unacknowledged octets. Access is read-only.
tcpRtoMin { tcp 2 } 1.3.6.1.2.1.6.2.	Integer	The minimum value allowed by a TCP implementation for the retransmission time-out, measured in milliseconds. More refined semantics for objects of this type depend upon the algorithm used to determine the retransmission time-out. For example, when the time-out algorithm is rsre (3), an object of this type has the semantics of the LBOUND quantity described in RFC 793. Access is read-only.
tcpRtoMax { tcp 3 } 1.3.6.1.2.1.6.3.	Integer	The maximum value allowed by a TCP implementation for the retransmission time-out, measured in milliseconds. More refined semantics for objects of this type depend upon the algorithm used to determine the retransmission time-out. For example, when the time-out algorithm is rsre (3), an object of this type has the semantics of the UBOUND quantity described in RFC 793. Access is read-only.
tcpMaxConn	Integer	The limit on the

{ tcp 4 } 1.3.6.1.2.1.6.4.		number of TCP connections the entry can support. In entities where the maximum number of connections is dynamic, this object should be -1. Access is read-only.
tcpActiveOpens { tcp 5 } 1.3.6.1.2.1.6.5.	Counter	The number of times TCP connections have made a direct transition to the SYN-SENT state from the CLOSED state. Access is read-only.
tcpPassiveOpens { tcp 6 } 1.3.6.1.2.1.6.6.	Counter	The number of times TCP connections have made a direct transition to the SYN-RCVD state from the LISTEN state. Access is read-only.
tcpAttemptFails { tcp 7 } 1.3.6.1.2.1.6.7.	Counter	The number of times TCP connections have made a direct transition to the CLOSED state from either the SYN-SENT state or the SYN-RCVD state, plus the number of times TCP connections have made a direct transition to the LISTEN state from the SYN-RCVD state. Access is read-only.
tcpEstabResets { tcp 8 } 1.3.6.1.2.1.6.8.	Counter	The number of times TCP connections have made a direct transition to the CLOSED state from either the ESTABLISHED or CLOSE-WAIT. Access is read-only.
tcpCurrEstab { tcp 9 } 1.3.6.1.2.1.6.9.	Gauge	The number of TCP connections of the current state that are either ESTABLISHED or CLOSE-WAIT. Access is read-only.
tcpInSegs { tcp 10 }	Counter	The total number of segments, including those

1.3.6.1.2.1.6.10.		received in error. This count includes segments received on currently established connections. Access is read-only.
tcpOutSegs { tcp 11 } 1.3.6.1.2.1.6.11.	Counter	The total number of segments sent, including those on currently established connections, but excluding those containing only retransmitted octets. Access is read-only.
tcpRetransSegs { tcp 12 } 1.3.6.1.2.1.6.12.	Counter	The total number of segments retransmitted that contain one or more previously transmitted octets. Access is read-only.
tcpConnTable { tcp 13 } 1.3.6.1.2.1.6.13	SEQUENCE OF TcpConnEntry	A table that contains TCP connection-specific information. Not accessible.
tcpConnEntry { tcpConnTable 1 } 1.3.6.1.2.1.6.13.1	TcpConnEntry :: =SEQUENCE tcpConnState INTEGER, tcpConnLocalAddress IpAddress, tcpConnLocalPort INTEGER (0..65535), tcpConnRemAddress IpAddress, tcpConnRemPort INTEGER (0..65535)	Information about a certain current TCP connection. An object of this type is transient. It does not exist when (or soon after) the connection makes the transition to the CLOSED state. Not accessible.
tcpConnState { tcpConnEntry 1 } 1.3.6.1.2.1.6.13.1.1	Integer closed(1), listen(2), synSent(3), synReceived(4), established(5), finWait1(6), finWait2(7), closeWait(8), lastAck(9), closing(10), timeWait(11) deleteTCB(12)	The TCP connection status. Access is read-write.
tcpConnLocalAddress { tcpConnEntry 2 } 1.3.6.1.2.1.6.13.1.2	IpAddress	The local IP address for this TCP connection. In the case of a connection in the LISTEN state that is willing to accept connections for any IP interface associated with the node, the

		value 0.0.0.0 is used. Access is read-only.
tcpConnLocalPort { tcpConnEntry 3 } 1.3.6.1.2.1.6.13.1.3	Integer (0..65535)	The local port number of this TCP connection. Access is read-only.
tcpConnRemAddress { tcpConnEntry 4 } 1.3.6.1.2.1.6.13.1.4	IpAddress	The remote IP address of this TCP connection. Access is read-only.
tcpConnRemPort { tcpConnEntry 5 } 1.3.6.1.2.1.6.13.1.5	Integer (0..65535)	The remote port number of this TCP connection. Access is read-only.
tcpInErrs { tcp 14 } 1.3.6.1.2.1.6.14.0	Counter	The total number of segments received in error (for example, bad TCP checksums). Access is read-only.
tcpOutRsts { tcp 15 } 1.3.6.1.2.1.6.15.0	Counter	The number of TCP segments sent containing the RST flag. Access is read-only.

E.7 UDP Group

Table 38 lists the objects in the UDP group. The UDP objects are the datagram statistics of the UDP layer.

Table 38. Implementation of the UDP Group

Object and ASN.1 Notation	**Syntax**	**Definition**
UDP GROUP 1.3.6.1.2.1.7		
udpInDatagrams { udp 1 } 1.3.6.1.2.1.7.1.0	Counter	The number of UDP datagrams delivered to UDP users. Access is read-only.
udpNoPorts { udp 2 } 1.3.6.1.2.1.7.2.0	Counter	The number of UDP datagrams received where there was no application at the destination port. Access is read-only.
udpInErrors { udp 3 } 1.3.6.1.2.1.7.3.0	Counter	The number of UDP datagrams received that could not be delivered for reasons other than the lack of an application at the destination port. Access is read-only.
udpOutDatagrams { udp 4 } 1.3.6.1.2.1.7.4.0	Counter	The number of UDP datagrams sent from this entry. Access is read-only.
UDP Listener Table 1.3.6.1.2.1.7.5		The UDP listener table contains information about this entity's UDP end-points on which a local application is currently accepting datagrams. Not accessible.
udpTable 1.3.6.1.2.1.7.5.1	SEQUENCE OF UdpEntry	A table containing UDP listener information. Not accessible.
udpEntry	udpEntry ::= SEQUENCE udpLocalAddress IpAddress, udpLocalPort INTEGER (0..65535)	Information about a particular current UDP listener. Not accessible.
udpLocalAddress 1.3.6.1.2.1.7.5.1.1	IpAddress	The local IP address for this listener. In the case of a UDP listener that can accept datagrams for any IP interface associated with the node, the value 0.0.0.0 is used. Access is read-only.
udpLocalPort 1.3.6.1.2.1.7.5.1.2	Integer	The local port number for this UDP listener.

```
¦                      ¦                       ¦ Access is read-only.  ¦
+----------------------------------------------------------------------+
```

E.8 EGP Group

Table 39 lists the objects in the EGP group.

Table 39. Implementation of the EGP Group

Object and ASN.1 Notation	**Syntax**	**Definition**
EGP GROUP 1.3.6.1.2.1.8		
egpInMsgs 1.3.6.1.2.1.8.1.0	Counter	The number of EGP messages received without error. Access is read-only.
egpInErrors 1.3.6.1.2.1.8.2.0	Counter	The number of EGP messages received that proved to be in error. Access is read-only.
egpOutMsgs 1.3.6.1.2.1.8.3.0	Counter	The total number of locally generated EGP messages. Access is read-only.
egpOutErrors 1.3.6.1.2.1.8.4.0	Counter	The number of locally generated EGP messages not sent because of resource limitations within an EGP entity. Access is read-only.
egpNeighTable 1.3.6.1.2.1.8.5	SEQUENCE OF EgpNeighEntry	Information about this entity's relationship with a particular EGP neighbor. Not accessible.
egpNeighEntry 1.3.6.1.2.1.8.5.1	EgpNeighEntry ::= SEQUENCE egpNeighState INTEGER, egpNeighAddr IpAddress, egpNeighAs INTEGER, egpNeighInMsgs Counter, egpNeighInErrs Counter, egpNeighOutMsgs Counter, egpNeighOutErrs Counter, egpNeighInErrMsgs Counter, egpNeighOutErrMsgs Counter, egpNeighStateUps Counter, egpNeighStateDowns Counter, egpNeighIntervalHello INTEGER, egpNeighIntervalPoll INTEGER, egpNeighMode INTEGER, egpNeighEventTrigger INTEGER	Information about this entity's relationship with a particular EGP neighbor. Not accessible.
egpNeighState	Integer	The EGP state of the local system with

{ egpNeighEntry 1 } 1.3.6.1.2.1.8.5.1.1		respect to this entry's EGP neighbor. Each EGP state is represented by a value that is one greater than the numerical value associated with that EGP state in RFC 904. Access is read-only.
egpNeighAddr { egpNeighEntry 2 } 1.3.6.1.2.1.8.5.1.2	IpAddress	The IP address of this entry's EGP neighbor. Access is read-only.
egpNeighAs { egpNeighEntry 3 } 1.3.6.1.2.1.8.5.1.3	Integer	The autonomous system of this EGP peer. Zero should be specified if the autonomous system number of the neighbor is not yet known. Access is read-only.
egpNeighInMsgs { egpNeighEntry 4 } 1.3.6.1.2.1.8.5.1.4	Counter	The number of EGP messages received without error from this EGP peer. Access is read-only.
egpNeighInErrs { egpNeighEntry 5 } 1.3.6.1.2.1.8.5.1.5	Counter	The number of EGP messages received from this EGP peer that proved to be in error (for example, bad EGP checksum). Access is read-only.
egpNeighOutMsgs { egpNeighEntry 6 } 1.3.6.1.2.1.8.5.1.6	Counter	The number of locally generated EGP messages to this EGP peer. Access is read-only.
egpNeighOutErrs { egpNeighEntry 7 } 1.3.6.1.2.1.8.5.1.7	Counter	The number of locally generated EGP messages not sent to this EGP peer because of resource limitations within an EGP entity. Access is read-only.
egpNeighInErrMsgs { egpNeighEntry 8 } 1.3.6.1.2.1.8.5.1.8	Counter	The number of EGP-defined error messages received from this EGP peer. Access is read-only.
egpNeighOutErrMsgs { egpNeighEntry 9 } 1.3.6.1.2.1.8.5.1.9	Counter	The number of EGP-defined error messages sent to this EGP peer. Access is read-only.
egpNeighStateUps { egpNeighEntry 10 } 1.3.6.1.2.1.8.5.1.10	Counter	The number of EGP state transitions to the UP state with this EGP peer. Access is read-only.
egpNeighStateDowns	Counter	The number of EGP

{ egpNeighEntry 11 } 1.3.6.1.2.1.8.5.1.11		state transitions from the UP state to any other state with this EGP peer. Access is read-only.
egpNeighIntervalHello { egpNeighEntry 12 } 1.3.6.1.2.1.8.5.1.12	Integer	The interval between EGP HELLO command retransmissions (in hundredths of a second). This represents the t1 timer as defined in RFC 904. Access is read-only.
egpNeighIntervalPoll { egpNeighEntry 13 } 1.3.6.1.2.1.8.5.1.13	Integer	The interval between EGP POLL command retransmissions (in hundredths of a second). This represents the t3 timer as defined in RFC 904. Access is read-only.
egpNeighMode { egpNeighEntry 14 } 1.3.6.1.2.1.8.5.1.14	Integer	The polling mode of this EGP entity, either passive or active. Access is read-only.
egpNeighEventTrigger { egpNeighEntry 15 } 1.3.6.1.2.1.8.5.1.15	Integer	A control variable used to trigger operator-initiated Start and Stop events. When read, this variable always returns the most recent value to which egpNeighEventTrigger was set. If it has not been set since the last initialization of the network management subsystem on the node, it returns a value of 'stop'. When set, this variable causes a Start or Stop event on the specified neighbor, as specified in RFC 904. A Start event causes an Idle peer to begin neighbor acquisition and a non-Idle peer to reinitiate neighbor acquisition. A Stop event causes an non-Idle peer to return to the Idle state until a Start event occurs, either by egpNeighEventTrigger or otherwise. Access is read-write.
egpAs 1.3.6.1.2.1.8.6	Integer	The autonomous system number of this EGP entity. Access is read-only.

E.9 SNMP Group

Table 40 lists the objects in the SNMP group.

Table 40. Implementation of the SNMP Group

Object and ASN.1 Notation	**Syntax**	**Definition**
SNMP GROUP 1.3.6.1.2.1.11		
SNMPInPkts { SNMP 1 } 1.3.6.1.2.1.11.1.0	Counter	The total number of Messages delivered to the SNMP entity from the transport service. Access is read-only.
SNMPOutPkts { SNMP 2 } 1.3.6.1.2.1.11.2.0	Counter	The total number of SNMP Messages that were passed from the SNMP protocol entity to the transport service. Access is read-only.
SNMPInBadVersions { SNMP 3 } 1.3.6.1.2.1.11.3.0	Counter	The total number of SNMP Messages that were delivered to the SNMP protocol entity and were for an unsupported SNMP version. Access is read-only.
snmpInBadCommunityNames { SNMP 4 } 1.3.6.1.2.1.11.4.0	Counter	The total number of SNMP Messages delivered to the SNMP protocol entity that used an SNMP community name not known to that entity. Access is read-only.
SNMPInBadCommunityUses { SNMP 5 } 1.3.6.1.2.1.11.5.0	Counter	The total number of SNMP Messages delivered to the SNMP protocol entity. These represent an SNMP operation that was not allowed by the SNMP community named in the Message. Access is read-only.
SNMPInASNParseErrs { SNMP 6 } 1.3.6.1.2.1.11.6.0	Counter	The total number of ASN.1 or BER errors encountered by the SNMP protocol entity when decoding received SNMP Messages. Access is read-only.
SNMPInTooBigs { SNMP 8 } 1.3.6.1.2.1.11.8.0	Counter	The total number of SNMP PDUs that were delivered to the SNMP protocol entity and for which the value of the error-status field is 'tooBig'. Access is read-only.
SNMPInNoSuchNames { SNMP 9 } 1.3.6.1.2.1.11.9.0	Counter	The total number of SNMP PDUs that were delivered to the SNMP protocol entity and for which the value of the error-status field is 'noSuchName'. Access is read-only.
SNMPInBadValues	Counter	The total number of SNMP PDUs that were delivered

{ SNMP 10 } 1.3.6.1.2.1.11.10.0		to the SNMP protocol entity and for which the value of the error-status field is 'badValue'. Access is read-only.
SNMPInReadOnlys { SNMP 11 } 1.3.6.1.2.1.11.11.0	Counter	The total number of valid SNMP PDUs that were delivered to the SNMP protocol entity and for which the value of the error-status field is 'readOnly'. It is a protocol error to generate an SNMP PDU that contains the value 'readOnly' in the error-status field; therefore, this object is provided as a means of detecting incorrect implementation of the SNMP. Access is read-only.
SNMPInGenErrs { SNMP 12 } 1.3.6.1.2.1.11.12.0	Counter	The total number of SNMP PDUs that were delivered to the SNMP protocol entity and for which the value of the error-status field is 'genErr'. Access is read-only.
SNMPInTotalReqVars { SNMP 13 } 1.3.6.1.2.1.11.13.0	Counter	The total number of MIB objects that have been retrieved successfully by the SNMP protocol entity as the result of receiving valid SNMP Get-Request and Get-Next PDUs. Access is read-only.
SNMPInTotalSetVars { SNMp 14 } 1.3.6.1.2.1.11.14.0	Counter	The total number of MIB objects that have been altered successfully by the SNMP protocol entity as the result of receiving valid SNMP Set-Request PDUs. Access is read-only.
SNMPInGetRequests { SNMP 15 } 1.3.6.1.2.1.11.15.0	Counter	The total number of SNMP Get-Request PDUs that have been accepted and processed by the SNMP protocol entity. Access is read-only.
SNMPInGetNexts { SNMP 16 } 1.3.6.1.2.1.11.16.0	Counter	The total number of SNMP Get-Next PDUs that have been accepted and processed by the SNMP protocol entity. Access is read-only.
SNMPInGetSetRequests { SNMP 17 } 1.3.6.1.2.1.11.17.0	Counter	Access is read-only.
SNMPInGetResponses { SNMP 18 } 1.3.6.1.2.1.11.18.0	Counter	The total number of SNMP. Get-Response PDUs that have been accepted and processed by the SNMP protocol entity. Access is read-only.

SNMPInTraps { SNMP 19 } 1.3.6.1.2.1.11.19.0	Counter	The total number of SNMP Trap PDUs that have been accepted and processed by the SNMP protocol entity. Access is read-only.
SNMPOutTooBigs { SNMP 20 } 1.3.6.1.2.1.11.20.0	Counter	The total number of SNMP PDUs that were generated by the SNMP protocol entity and for which the value of the error-status field is 'tooBig'. Access is read-only.
SNMPOutNoSuchNames { SNMP 21 } 1.3.6.1.2.1.11.21.0	Counter	The total number of SNMP PDUs that were generated by the SNMP protocol entity and for which the value of the error-status is 'noSuchName'. Access is read-only.
SNMPOutBadValues { SNMP 22 } 1.3.6.1.2.1.11.22.0	Counter	The total number of SNMP PDUs that were generated by the SNMP protocol entity and for which the value of the error-status field is 'badValue'. Access is read-only.
SNMPOutReadOnlys { SNMP 23 } 1.3.6.1.2.1.11.23.0	Counter	Access is read-only.
SNMPOutGenErrs { SNMP 24 } 1.3.6.1.2.1.11.24.0	Counter	The total number of SNMP PDUs that were generated by the SNMP protocol entity and for which the value of the error-status field is 'genErr'. Access is read-only.
SNMPOutGetRequests { SNMP 25 } 1.3.6.1.2.1.11.25.0	Counter	The total number of SNMP Get-Request PDUs that have been generated by the SNMP protocol entity. Access is read-only.
SNMPOutGetNextssnmp { SNMP 26 } 1.3.6.1.2.1.11.26.0	Counter	The total number of SNMP Get-Next PDUs that have been generated by the SNMP protocol entity. Access is read-only.
SNMPOutSetRequests { SNMP 27 } 1.3.6.1.2.1.11.27.0	Counter	The total number of SNMP Set-Request PDUs that have been generated by the SNMP protocol entity. Access is read-only.
SNMPOutGetResponses { SNMP 28 } 1.3.6.1.2.1.11.28.0	Counter	The total number of SNMP Get-Response PDUs that have been generated by the SNMP protocol entity. Access is read-only.
SNMPOutTraps { SNMP 29 } 1.3.6.1.2.1.11.29.0	Counter	The total number of SNMP Trap PDUs that have been generated by the SNMP protocol entity. Access is read-only.

SNMPEnableAuthTraps { SNMP 30 } 1.3.6.1.2.1.11.30.0	Integer	Indicates whether the SNMP agent process is permitted to generate authentication-failure traps. The value of this object overrides any configuration information; as such, it provides a means whereby all authentication-failure traps can be disabled. This object should be stored in nonvolatile memory so that it remains constant between reinitializations of the network management system. Access is read-write.

F.0 Appendix F. Requests for Comments (RFCs)

IBM is committed to industry standards. The internet protocol suite is still evolving through Requests for Comments (RFC). New protocols are being designed and implemented by researchers, and are brought to the attention of the internet community in the form of RFCs. Some of these are so useful that they become a recommended protocol. That is, all future implementations for TCP/IP are recommended to implement this particular function or protocol. These become the actual standards on which the TCP/IP protocol suite is built.

Many features of TCP/IP for OS/2 are based on the following RFCs:

RFC	Title and Author
768	*User Datagram Protocol*, J.B. Postel
783	*Trivial File Transfer Protocol*, (Revision 2), K.R. Sollins
791	*Internet Protocol*, J.B. Postel
792	*Internet Control Message Protocol*, J.B. Postel
793	*Transmission Control Protocol*, J.B. Postel
821	*Simple Mail Transfer Protocol*, J.B. Postel
822	*Standard for the Format of ARPA Internet Text Messages*, D. Crocker
823	*DARPA Internet Gateway*, R.M. Hinden, A. Sheltzer
826	*Ethernet Address Resolution Protocol: or Converting Network Protocol Addresses to 48.Bit Ethernet Address for Transmission on Ethernet Hardware*, D.C. Plummer
854	*Telnet Protocol Specification*, J.B. Postel, J.K. Reynolds
856	*Telnet Binary Transmission*, J.B. Postel, J.K. Reynolds
857	*Telnet Echo Option*, J.B. Postel, J.K. Reynolds
877	*Standard for the Transmission of IP Datagrams over Public Data Networks*, J.T. Korb
885	*Telnet End of Record Option*, J.B. Postel
919	*Broadcasting Internet Datagrams*, J.C. Mogul
922	*Broadcasting Internet Datagrams in the Presence of Subnets*, J.C. Mogul
950	*Internet Standard Subnetting Procedure*, J.C. Mogul, J.B. Postel
952	*DoD Internet Host Table Specification*, K. Harrenstien, M.K. Stahl, E.J. Feinler
959	*File Transfer Protocol*, J.B. Postel, J.K. Reynolds
974	*Mail Routing and The Domain Name System*, C. Partridge
1013	*X Window System Protocol, Version 11: Alpha Update*, R.W. Scheifler
1014	*XDR: External Data Representation Standard*, Sun Microsystems Incorporated
1034	*Domain Names--Concepts and Facilities*, P.V. Mockapetris
1035	*Domain Names--Implementation and Specification*, P.V. Mockapetris
1055	*Nonstandard for Transmission of IP Datagrams Over Serial Lines: SLIP*, J.L. Romkey
1057	*RPC: Remote Procedure Call Protocol Version 2 Specification*, Sun Microsystems Incorporated
1058	*Routing Information Protocol*, C.L. Hedrick
1060	*Assigned Numbers*, J.K. Reynolds, J.B. Postel
1091	*Telnet Terminal-Type Option*, J. VanBokkelen
1094	*NFS: Network File System Protocol Specification*, Sun Microsystems Incorporated

1118 *Hitchhikers Guide to the Internet*, E. Krol

1122 *Requirements for Internet Hosts--Communication Layers*, R.T. Braden, editor

1123 *Requirements for Internet Hosts--Application and Support*, R.T. Braden, editor

1155 *Structure and Identification of Management Information for TCP/IP-Based Internets*, M.T. Rose, K. McCloghrie

1157 *Simple Network Management Protocol (SNMP),* J.D. Case, M. Fedor, M.L. Schoffstall, C. Davin

1179 *Line Printer Daemon Protocol*, The Wollongong Group, L. McLaughlin III, editor

1180 *TCP/IP Tutorial,* T.J. Socolofsky, C.J. Kale

1187 *Bulk Table Retrieval with the SNMP*

1200 *Defense Advanced Research Projects Agency, Internet Activities Board IAB Official Protocol Standards*

1206 *FYI on Questions and Answers: Answers to Commonly Asked "New Internet User" Questions,*, G.S. Malkin, A.N. Marine

1207 *FYI on Questions and Answers: Answers to Commonly Asked "Experienced Internet User" Questions,*, G.S. Malkin, A.N. Marine, J.K. Reynolds

1208 *Glossary of Networking Terms,*, O.J. Jacobsen, D.C. Lynch

1213 *Management Information Base for Network Management of TCP/IP-Based Internets:MIB-II,* K. McCloghrie, M.T.Rose, editors

1250 *IAB Official Protocol Standards,* J. B. Postel

These documents can be obtained from:

Government Systems, Inc.
Attn: Network Information Center
14200 Park Meadow Drive
Suite 200
Chantilly, VA 22021

For more information, see "Obtaining Requests for Comments (RFCs)" in topic PREFACE.7.1.

G.0 Appendix G. Well-Known Port Assignments

This appendix lists the well-known ports supported by TCP/IP for OS/2. It provides the port number, keyword, and a description of the reserved port assignment. You can also find a list of some of these well-known port numbers in the SERVICES file.

Table 41 lists the TCP well-known ports, and Table 42 in topic G.2 lists the UDP well-known ports.

Subtopics
G.1 TCP Well-Known Port Assignments
G.2 UDP Well-Known Port Assignments

G.1 TCP Well-Known Port Assignments

Table 41. TCP Well-Known Port Assignments

Port Number	Keyword	Reserved for	Services Description
0		reserved	
5	**rje**	remote job entry	remote job entry
7	**echo**	echo	echo
9	**discard**	discard	sink null
11	**systat**	active users	active users
13	**daytime**	daytime	daytime
15	**netstat**	Netstat	who is up or Netstat
19	**chargen**	ttytst source	character generator
21	**ftp**	FTP	File Transfer Protocol
23	**telnet**	Telnet	Telnet
25	**smtp**	mail	Simple Mail Transfer Protocol
37	**time**	timeserver	timeserver
39	**rlp**	resource	Resource Location Protocol
42	**nameserver**	name	host name server
43	**nicname**	who is	who is
53	**domain**	name server	domain name server
57	**mtp**	private terminal access	private terminal access
69	**tftp**	TFTP	Trivial File Transfer Protocol
77		netrjs	any private RJE service
79	**finger**	finger	finger
87	**link**	ttylink	any private terminal link
95	**supdup**	supdup	SUPDUP Protocol
101	**hostname**	hostname	nic hostname server, usually from SRI-NIC
109	**pop**	postoffice	Post Office Protocol
111	**sunrpc**	sunrpc	Sun remote procedure call
113	**auth**	authentication	authentication service
115	**sftp**	sftp	Simple File Transfer Protocol
117	**uucp-path**	UUCP path service	UUCP path service
119	**untp**	readnews untp	USENET News Transfer Protocol
123	**ntp**	NTP	Network Time Protocol
160		reserved	
161	**snmp**	snmp Agent	snmp Agent receives packets
162	**snmptrap**	snmp Client	snmp Client receives snmp TRAPs

163-223		reserved	
712	**vexec**	vice-exec	Andrew File System authenticated service
713	**vlogin**	vice-login	Andrew File System authenticated service
714	**vshell**	vice-shell	Andrew File System authenticated service
2001	**filesrv**		Andrew File System service
2106	**venus.itc**		Andrew File System service, for the Venus process

G.2 UDP Well-Known Port Assignments

Table 42. UDP Well-Known Port Assignments

Port Number	Keyword	Reserved for	Services Description
0		reserved	
5	**rje**	remote job entry	remote job entry
7	**echo**	echo	echo
9	**discard**	discard	sink null
11	**users**	active users	active users
13	**daytime**	daytime	daytime
15	**netstat**	Netstat	Netstat
19	**chargen**	ttytst source	character generator
37	**time**	timeserver	timeserver
39	**rlp**	resource	Resource Location Protocol
42	**nameserver**	name	host name server
43	**nicname**	who is	who is
53	**domain**	name server	domain name server
67	**bootps**	bootps	bootp server
68	**bootpc**	bootpc	bootp client
69	**tftp**	TFTP	Trivial File Transfer Protocol
75			any private dial out service
77		netrjs	any private RJE service
79	**finger**	finger	finger
111	**sunrpc**	sunrpc	Sun remote procedure call
123	**ntp**	NTP	Network Time Protocol
135	**llbd**		
160-223		reserved	
531	**rvd-control**		rvd control port
2001	**rauth2**		Andrew File System service, for the Venus process
2002	**rfilebulk**		Andrew File System service, for the Venus process
2003	**rfilesrv**		Andrew File System service, for the Venus process
2018	**console**		Andrew File System service
2115	**ropcons**		Andrew File System service, for the Venus process
2131	**rupdsrv**		assigned in pairs; bulk must be **srv** +1
2132	**rupdbulk**		assigned in pairs; bulk must be **srv** +1
2133	**rupdsrv1**		assigned in pairs; bulk must be **srv** +1
2134	**rupdbulk1**		assigned in pairs; bulk must be **srv** +1

BIBLIOGRAPHY Bibliography

This bibliography is intended as a guide for identifying related publications for TCP/IP products.

For publications related specifically to IBM TCP/IP for OS/2, see "TCP/IP for OS/2 Publications."

Subtopics

BIBLIOGRAPHY.1 IBM TCP/IP Products

The following books describe the library of books associated with TCP/IP products:

Subtopics

BIBLIOGRAPHY.1.1 TCP/IP for OS/2 Publications

The following list shows books in the TCP/IP for OS/2 library:

- *IBM TCP/IP Version 2.0 for OS/2: Installation and Administration*, SC31-6075

  This book provides system programmers responsible for installing TCP/IP for OS/2 with the information required to plan and implement the installation of TCP/IP for OS/2. The topics include hardware and software requirements, pre-installation system performance considerations, instructions for defining and generating TCP/IP for OS/2, instructions for customizing the TCP/IP for OS/2 environment, and installation examples.

- *IBM TCP/IP Version 2.0 for OS/2: Programmer's Reference*, SC31-6077

  This book is written for application and system programmers in writing application programs that use TCP/IP for OS/2. Application programmers should know the OS/2 operating system.

- *IBM TCP/IP Version 2.0 for OS/2: Command Reference*, SX75-0070

  This book contains an alphabetical listing of TCP/IP for OS/2 commands, syntax diagrams, and parameter descriptions.

- *IBM TCP/IP Version 2.0 for OS/2: User's Guide*, SC31-6076

  This book is a guide for people, such as end users and system programmers, who use TCP/IP for OS/2 for data communication. The people who use this book should be familiar with OS/2.

- *IBM NetBIOS Version 2.0 for Transmission Control Protocol/Internet Protocol Version 2.0 for OS/2,* SC31-6122

  This book provides information for using IBM's NetBIOS program. This program provides a standard interface to the local area network for OS/2 applications using IBM's Transmission Control Protocol/Internet Protocol Version 2.0 for OS/2.

- *IBM Transmission Control Protocol/Internet Protocol Version 2.0 for OS/2: X Window System Client Guide*, SC31-7087

  This book describes how to install and use the X Window System Client Kit. It also lists the files included in the kit. The X Window System Client Kit enables users to develop and run applications that use the X Windows System in an OS/2 environment.

  This book also describes how to install and use the Motif kit. This kit enables users to develop and run OSF/Motif** applications in an OS/2 environment.

- *IBM TCP/IP Version 2.0 for OS/2: Domain Name Server Guide*, SC31-7174

  This book describes how to install, use, and diagnose the IBM TCP/IP Version 2.0 for OS/2: Domain Name Server Kit. The IBM TCP/IP Version 2.0 for OS/2: Domain Name Server Kit provides a domain name server and its configuration files.

- *IBM Transmission Control Protocol/Internet Protocol Version 2.0 for OS/2: UltiMail Installation and Configuration Guide,* SC31-7120

  Note: This book will be available at a later time.

BIBLIOGRAPHY.1.2 TCP/IP for VM Publications

The following paragraphs describe the library of books associated with TCP/IP for VM:

- *IBM TCP/IP Version 2 Release 2 for VM: Messages and Codes*, SC31-6151

  This book is written for system programmers for diagnosing problems.

- *IBM TCP/IP Version 2 Release 2 for VM: Planning and Customization*, SC31-6082

  This book provides system programmers responsible for installing TCP/IP for VM with the information required to plan and implement the installation of TCP/IP for VM. The topics include hardware and software requirements, pre-installation system performance considerations, instructions for defining and generating TCP/IP for VM, instructions for customizing the TCP/IP for VM environment, and installation examples.

- *IBM TCP/IP Version 2 Release 2 for VM: Programmer's Reference*, SC31-6084

  This book is written for application and system programmers in writing application programs that use TCP/IP for VM. Application programmers should know the VM operating system.

- *IBM TCP/IP Version 2 Release 2 for VM: User's Guide*, SC31-6081

  This book is a guide for people, such as end users and system programmers, who use TCP/IP for VM for data communication. The people who use this book should be familiar with the IBM Virtual Machine (VM) operating system, IBM Command Processor (CP), and IBM Conversational Monitor System (CMS).

BIBLIOGRAPHY.1.3 TCP/IP for MVS Publications

The following paragraphs describe the library of books associated with TCP/IP for MVS:

- *IBM TCP/IP Version 2 for MVS: Installation and Maintenance*, SC31-6085

  This book provides information on how to install, configure, and maintain TCP for MVS. The user of this book should be familiar with the TCP protocols, and be familiar with MVS and the IBM Time Sharing Option (TSO).

- *IBM TCP/IP Version 2 for MVS: Messages and Codes*, SC31-6142

  This is used by the system programmers for diagnosing problems.

- *IBM TCP/IP Version 2 for MVS: Programmer's Reference*, SC31-6087

  This book provides information on how to use the supplied interfaces while writing application programs that access TCP for MVS. The user of this book should be familiar with the TCP protocols, and be familiar with MVS, and the IBM Time Sharing Option (TSO).

- *IBM TCP/IP Version 2 for MVS: User's Guide,*, SC31-6088

  This book is a guide for people, such as end users and system programmers, who use TCP/IP for MVS for data communication. The people who use this book should be familiar with the IBM Multiple Virtual Storage (MVS) operating system and IBM Time Sharing Option (TSO).

BIBLIOGRAPHY.1.4 TCP/IP for DOS Publications

The following list shows the books in the TCP/IP for DOS library:

- *IBM TCP/IP V2.1 for DOS: Installation and Administration*, SC31-7047.

  This book provides system programmers, network administrators, and workstation users responsible for installing IBM TCP/IP for DOS with the information required to plan and implement the installation of IBM TCP/IP for DOS. The topics include hardware and software requirements, pre-installation system performance considerations, instructions for installing IBM TCP/IP for DOS, instructions for customizing the IBM TCP/IP for DOS environment and installation examples.

- *IBM TCP/IP V2.1 for DOS: Programmer's Reference*, SC31-7046.

  This book is written for application and system programmers to aid them in writing application programs that use IBM TCP/IP for DOS on a workstation. Application programmers should know the DOS operating system, and have knowledge of multitasking operating system concepts. Application programmers should be knowledgeable in the C programming language.

- *IBM TCP/IP V2.1 for DOS: User's Guide*, SC31-7045.

  This book is written for people who use a workstation with IBM TCP/IP for DOS, such as end users and system programmers. The people who use this book should be familiar with DOS and the workstation, and also understand DOS operating system concepts.

- *IBM TCP/IP V2.1 for DOS: Command Reference*, SX75-0083.

  This book is written for people who use a workstation with IBM TCP/IP for DOS, such as end users and system programmers. The people who use this book should be familiar with DOS and the workstation, understand DOS operating system concepts, and be familiar with the *IBM TCP/IP V2.1 for DOS: User's Guide*.

BIBLIOGRAPHY.1.5 TCP/IP for AIX (RS/6000, PS/2, RT, 370) Publications

The following list shows the books in the TCP/IP for AIX library:

- *AIX Operating System TCP/IP Users Guide*, SC23-2309
- *AIX/RT TCP/IP Version 2.2*, SC23-2005
- *IBM Advanced Interactive Executive for the Personal System/2* (AIX PS/2): TCP/IP
- *IBM AIX X-Windows Programmer's Reference*, SC23-2118
- *RT/PC Interface Program for TCP/IP*, SC23-0812
- *TCP/IP for IBM X-Window*, SC23-2349

BIBLIOGRAPHY.1.6 TCP/IP for AS/400 Publications

The following list shows the books in the TCP/IP for AS/400 library:

- *IBM AS/400 Communications: TCP/IP Guide*, SC21-9875
- *IBM AS/400 Communications: User's Guide*, SC21-9601

BIBLIOGRAPHY.1.7 Other IBM TCP/IP Publications

The following list shows other available IBM TCP/IP Publications:

- *Introducing IBM Transmission Control Protocol/Internet Protocol Products for OS/2, VM, and MVS*, GC31-6080

- *IBM International Technical Support Centers: TCP/IP Tutorial and Technical Overview*, GG24-3376

- *IBM TCP/IP Version 2 for VM and MVS: Diagnosis Guide*, LY43-0013

- *IBM Local Area Network Technical Reference*, SC30-3383

- *MVS/DFP Version 3 Release 3: Using the Network File System Server*, SC26-4732

BIBLIOGRAPHY.2 The IBM C Set/2 Library

The following books are part of the IBM C Set/2 library.

- *SAA Common Programming Interface C Reference*, SC09-1308
- *IBM C Set/2 User's Guide*, S10G-4444
- *IBM C Set/2 Migration Guide*, S10G-4445
- *IBM C Set/2 Reference Summary*, S10G-4446
- *IBM C Set/2 Debugger Tutorial*, S10G-4447
- *IBM C Set/2 Installation*, S10G-4443
- *IBM C Set/2 and WorkFrame/2: An Integrated Development Environment*, S10G-4449

BIBLIOGRAPHY.3 C Related Publications

The following book is related to the IBM C Set/2 library.

- *Portability Guide for IBM C*, SC09-1405

BIBLIOGRAPHY.4 IBM WorkFrame/2 Publication

The following book provides information about IBM WorkFrame/2.

- *IBM WorkFrame/2: Introduction*, S10G-4475

BIBLIOGRAPHY.5 IBM OS/2 Version 2.0 Publications

The following books describe the OS/2 Version 2.0 operating system and Developer's Toolkit.

- *IBM OS/2 Version 2.0 Overview Manual*, S84F-8465
- *IBM OS/2 Version 2.0 Installation Guide*, S84F-8464
- *IBM OS/2 Version 2.0 Quick Reference*, S10G-5964
- *Getting Started*, S10G-6199

BIBLIOGRAPHY.6 IBM OS/2 Version 2.0 Technical Library

The following books make up the OS/2 Version 2.0 Technical Library (10G3356).

- *Application Design Guide*, S10G-6260
- *Programming Guide*, S10G-6261
- *Information Presentation Facility Guide and Reference*, S10G-6262
- *System Object Model Guide and Reference*, S10G-6309
- *Control Program Programming Reference*, S10G-6263
- *Presentation Manager Programming Reference Volume 1*, S10G-6264
- *Presentation Manager Programming Reference Volume 2*, S10G-6265
- *Physical Device Driver Reference*, S10G-6266
- *Virtual Device Driver Reference*, S10G-6310
- *Presentation Manager Driver Reference*, S10G-6267
- *Procedures Language 2/REXX Reference*, S10G-6268
- *Procedures Language 2/REXX User's Guide*, S10G-6269
- *SAA Common User Access&asterick. Guide to User Interface Design*, SC34-4289
- *SAA Common User Access Advanced User Interface Design Guide*, SC34-4290

BIBLIOGRAPHY.7 IBM Software Products

The following publications are related specifically to IBM Software Products:

Subtopics

BIBLIOGRAPHY.7.1 X.25 NPSI Publications

BIBLIOGRAPHY.7.1 X.25 NPSI Publications

The following list shows selected X.25 NPSI publications:

- *X.25 Network Control Program Packet Switching Interface Diagnosis, Customization, and Tuning Version 3*, LY30-5610
- *X.25 Network Control Program Packet Switching Interface Host Programming*, SC30-3502
- *X.25 Network Control Program Packet Switching Interface Planning and Installation*, SC30-3470

BIBLIOGRAPHY.8 Other TCP/IP Related Publications

The following list shows other selected TCP/IP related publications:

- *Internetworking With TCP/IP Volume I: Principles, Protocols, and Architecture*, Douglas E. Comer, Prentice Hall, Englewood Cliffs, New Jersey, 1991 SC31-6144

- *Internetworking With TCP/IP Volume II: Implementation, and Internals*, Douglas E. Comer, Prentice Hall, Englewood Cliffs, New Jersey, 1991 SC31-6145

- "MIB II Extends SNMP Interoperability," C. Vanderberg, *Data Communications*, October 1990

- *"Network Management and the Design of SNMP"*, J.D. Case, J.R. Davin, M.S. Fedor, M.L. Schoffstall

- *"Network Management of TCP/IP Networks: Present and Future,"* A. Ben-Artzi, A. Chandna, V. Warrier

- "Special Issue: Network Management and Network Security," *ConneXions-The Interoperability Report*, Volume 4, No. 8, August 1990

- *IEEE Network Magazine*, July 1990

- *The Simple Book: An Introduction to Management of TCP/IP-based Internets*, Marshall T Rose, Prentice Hall, Englewood Cliffs, New Jersey, 1991

- *sendmail* Bryan Costales with Eric Allamn and Neil Rickert, O'Reilly and Associates, 1993

- *SNMP, SNMPv2, and CMIP: The Practical Guide to Network Management Standards*, William Stallings, Addison Wesley, 1993

- *TCP/IP Network Administration*, Hunt, Craig, O'Reilly and Associates, 1992

- *Interoperability Report*, Volume 3, No. 3, March 1989

- *UNIX Programmer's Reference Manual*, (4.3 Berkeley Software Distribution, Virtual VAX-11 Version). Department of Electrical Engineering and Computer Science. University of California, Berkeley, 1988

- *Unix Network Programming*, W. Richard Stevens, Prentice Hall, Englewood Cliffs, New Jersey, 1990

Subtopics

BIBLIOGRAPHY.8.1 Sun (RPC) Publications

The following list shows selected Sun Microsystems publications:

- *Networking on the Sun Workstation: Remote Procedure Call Programming Guide*, (800-1324-03), Sun Microsystems, Inc.

- *Network Programming*, (800-1779-10), Sun Microsystems, Inc.

- *The Art of Distributed Application: Programming Techniques for Remote Procedure Calls*, John R. Corbin, Springer-Verlog, 1991

BIBLIOGRAPHY.8.2 Network Transport Services/2

- *Network Transport Services/2*, S96F-8488

BIBLIOGRAPHY.8.3 OSF/Motif Publications

The following list shows selected OSF/Motif publications:

- *OSF/Motif Application Environment Specifications*, (AES), ISBN 0-13-640483-9
- *OSF/Motif Programmer's Guide*, ISBN 0-13-640509-6
- *OSF/Motif Programmer's Reference*, ISBN 0-13-640517-7
- *OSF/Motif Series (5 volumes), Open Software Foundation*, Prentice Hall, 1990
- *OSF/Motif Style Guide*, ISBN 0-13-640491-X
- *OSF/Motif User's Guide*, ISBN 0-13-640525-8

BIBLIOGRAPHY.8.4 X Window System Publications

The following list shows selected X Window System publications:

- *Introduction to the X Window System*, Oliver Jones, Prentice-Hall, 1988. ISBN 0-13-499997-5

- *The X Window System Series (6 volumes)*, O'Reilly and Associates, 1988, 1989, 1990. ISBN 0-937175-40-4, 0-937175-27-7, 0-937175-28-5, 0-937175-35-6, 0-937175-33-1, 0-937175-35-8

- *X Protocol Reference Manual*, Adrian Nye, ed. O'Reilly & Associates, Inc., 1990

- *X Window System: C Library and Protocol Reference*, Robert Scheifler, James Gettys, and Ron Newman, DEC Press, 1988. ISBN 1-55558-012-2

- *X Window System: Programming and Applications With Xt*, Douglas A. Young, Prentice-Hall, 1989. ISBN 0-13-972167-3

- *X Window System: Programming and Applications With Xt, OSF/Motif Edition*, Douglas A. Young, Prentice-Hall, 1990. ISBN 0-13-497074-8

- *X Window System Technical Reference*, Steven Mikes, Addison-Wesley, 1990. ISBN 0-201-52370-1

- *X Window System User's Guide*, Valerie Quercia & Tim O'Reilly., O'Reilly & Associates, Inc., 1990

M

S

T

U

V

BACK_1 Communicating Your Comments to IBM
IBM TCP/IP for OS/2:
User's Guide

Publication No. SC31-6076-05

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

If you are mailing a readers' comment form (RCF) from a country other than the United States, you can give the RCF to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by mail, use the RCF at the back of this book.

- If you prefer to send comments by FAX, use this number:

  United States and Canada: **1-800-227-5088**

- If you prefer to send comments electronically, use this network ID:

  - IBM Mail Exchange: **USIB2HPD at IBMMAIL**
  - IBMLink: **CIBMORCF at RALVM13**
  - Internet: **USIB2HPD@VNET.IBM.COM**

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies.

COMMENTS Help us help you!
IBM TCP/IP for OS/2:
User's Guide

Publication No. SC31-6076-05

We hope you find this publication useful, readable and technically accurate, but only you can tell us! Your comments and suggestions will help us improve our technical publications. Please take a few minutes to let us know what you think by completing this form.

Overall, how satisfied are you with the information in this book?	Satisfied	Dissatisfied
	__	__

How satisfied are you that the information in this book is:	Satisfied	Dissatisfied
Accurate	__	__
Complete	__	__
Easy to find	__	__
Easy to understand	__	__
Well organized	__	__
Applicable to your task	__	__

Specific Comments or Problems:

_____

_____

_____

Please tell us how we can improve this book:

_____

_____

_____

Thank you for your response. When you send information to IBM, you grant IBM the right to use or distribute the information without incurring any obligation to you. You of course retain the right to use the information in any way you choose.

Please complete this form and mail it to:

International Business Machines Corporation
Information Development
Department E15
PO BOX 12195
RESEARCH TRIANGLE PARK NC 27709-9990

or give it to your IBM representative.

Name . . . . . _____
Company or Organization _____
Address . . . . . _____
_____
_____
Phone No. . . . . _____